PDF 다운로드 경험을 알려주세요.

Win32 API에 대한 프로그래밍 참조

Win32 API 참조 설명서는 여러 가지 보기로 제공됩니다. 이 페이지에서 인기 있는 기술 목록을 찾아보거나 목차의 전체 기술 목록을 찾아볼 수 있습니다. 모든 헤더를 찾아보려면 목차 맨 아래에 있는 목록을 참조하세요.

시작

개요

Win32 API를 사용하여 데스크톱 Windows 앱 빌드

다운로드

개발 도구

Windows 코드 샘플

사용자 인터페이스 및 데스크톱

참조

Windows 컨트롤

Windows 및 메시지

메뉴 및 다른 리소스

Windows 셸

접근성 기능

국제화

그래픽 및 게임

참조

Direct2D

Direct3D 11 그래픽

Direct3D 12 그래픽

DirectML

DXGI

Windows GDI

GDI+

Windows Imaging 구성 요소

오디오 및 비디오

참조

Microsoft Media Foundation

Windows 멀티미디어

XAudio2 API

핵심 오디오 API

오디오 디바이스 DDI 참조

데이터 액세스 및 스토리지

참조

데이터 액세스 및 스토리지

Backup

Background Intelligent Transfer Service

클라우드 필터 API

데이터 교환

구조적 스토리지

가상 스토리지

디바이스

참조

커널 모드 드라이버 참조

디바이스 및 드라이버 설치 참조

스토리지 드라이버 DDI 참조

USB 드라이버 참조

디바이스 참조 표시

휴먼 인터페이스 디바이스 참조

DDI 참조 인쇄

네트워킹 및 인터넷

참조

Bluetooth

HTTP Server API(HTTP 서버 API)

IP 도우미

네트워크 관리

RPC(원격 프로시저 호출)

Windows HTTP 서비스(WinHTTP)

Windows 인터넷

Windows 소켓 2

보안 및 ID

참조

보안 및 ID

네트워크 액세스 보호

네트워크 정책 서버

자녀

Windows 생체 인식 프레임워크

TPM 기본 서비스

진단

참조

이벤트 추적

네트워크 진단 프레임워크

성능 카운터

애플리케이션 복구 및 다시 시작

TraceLogging

Windows 이벤트 수집기

Windows 오류 보고

애플리케이션 설치

참조

애플리케이션 설치 및 서비스

앱 패키징 및 배포 Windows 10

개발자 라이선스

관리자 다시 시작

시스템 서비스

참조

COM(구성 요소 개체 모델)

COM+(구성 요소 서비스)

MICROSOFT 인터페이스 정의 언어(MIDL)

압축 API

활동 코디네이터

하드웨어 요구 사항 계산기(HWREQCHK)

Active Directory 도메인 서비스

아티클2023. 03. 09.

Active Directory Domain Services 기술 개요입니다.

Active Directory Domain Services 개발하려면 다음 헤더가 필요합니다.

- adsprop.h
- cmnquery.h
- dsadmin.h
- dsclient.h
- dsgetdc.h
- dsparse.h
- dsquery.h
- dsrole.h
- ntdsapi.h
- objsel.h
- schedule.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Active Directory Domain Services

열거형

DS_KCC_TASKID KCC(지식 일관성 검사기)가 실행할 수 있는 작업을 지정합니다.
DS_MANGLE_FOR DS_MANGLE_FOR 열거형은 상대 고유 이름이 손상(인코딩)되는지 여부와 맨글링이 발생하는 형태를 정의하는 데 사용됩니다.
DS_NAME_ERROR DS_NAME_ERROR 열거형은 DS_NAME_RESULT_ITEM 구조체의 상태 멤버가 반환하는 오류를 정의합니다. DsCrackNames 함수에서 이름을 변환하는 동안 발생할 수 있는 잠재적 오류입니다.

DS_NAME_FLAGS DS_NAME_FLAGS 열거형은 이름 구문을 해독하는 방법을 정의하는 데 사용됩니다. 이러한 플래그는 DsCrackNames 함수에서 사용됩니다.
DS_NAME_FORMAT DS_NAME_FORMAT 열거형은 DsCrackNames 함수의 입력 및 출력 이름에 사용할 형식을 제공합니다.
DS_REPL_INFO_TYPE DS_REPL_INFO_TYPE 열거형은 DsReplicaGetInfo 및 DsReplicaGetInfo2 함수와 함께 사용하여 검색할 복제 데이터의 형식을 지정합니다.
DS_REPL_OP_TYPE 복제 큐의 지정된 항목이 나타내는 복제 작업의 유형을 나타내는 데 사용됩니다.
DS_REPSYNCALL_ERROR DS_REPSYNCALL_ERROR 열거형은 복제 프로세스에서 오류가 발생한 위치를 나타내기 위해 DS_REPSYNCALL_ERRINFO 구조와 함께 사용됩니다.
DS_REPSYNCALL_EVENT DS_REPSYNCALL_EVENT 열거형은 DS_REPSYNCALL_UPDATE 구조가 나타내는 이벤트를 정의하기 위해 DS_REPSYNCALL_UPDATE 구조체와 함께 사용됩니다.
DS_SPN_NAME_TYPE DS_SPN_NAME_TYPE 열거형은 DsGetSPN 함수에서 SPN 작성 형식을 식별하는 데 사용됩니다.
DS_SPN_WRITE_OP DS_SPN_WRITE_OP 열거형은 DsWriteAccountSpn 함수에서 수행해야 하는 쓰기 작업의 형식을 식별합니다.
DSROLE_MACHINE_ROLE 컴퓨터 역할을 지정하기 위해 DSROLE_PRIMARY_DOMAIN_INFO_BASIC 구조체의 MachineRole 멤버와 함께 사용됩니다.
DSROLE_OPERATION_STATE 컴퓨터의 작동 상태를 나타내기 위해 DSROLE_OPERATION_STATE_INFO 구조와 함께 사용됩니다.

DSROLE_PRIMARY_DOMAIN_INFO_LEVEL DsRoleGetPrimaryDomainInformation 함수와 함께 검색할 데이터 형식을 지정하는 데 사용됩니다.
DSROLE_SERVER_STATE DSROLE_UPGRADE_STATUS_INFO 구조와 함께 서버의 역할을 나타내는 데 사용됩니다.

Functions

AddForms 쿼리 양식 확장 개체가 쿼리 대화 상자에 양식을 추가할 수 있도록 하기 위해 호출됩니다.
AddPages 쿼리 양식 개체가 기존 양식에 페이지를 추가할 수 있도록 하기 위해 호출됩니다.
AddPages IDsAdminNewObjExt::AddPages 메서드가 호출되어 개체 만들기 마법사 확장에서 원하는 페이지를 마법사에 추가할 수 있도록 합니다.
ADsPropCheckIfWritable ADsPropCheckIfWritable 함수는 특성을 쓸 수 있는지 여부를 결정합니다.
ADsPropCreateNotifyObj ADsPropCreapCreateNotifyObj 함수는 Active Directory Domain Services 속성 시트 확장에서 사용할 알림 개체를 만들거나 가져오는 데 사용됩니다.
ADsPropGetInitInfo Active Directory Domain Services 속성 시트 확장이 적용되는 디렉터리 개체 데이터를 가져오는 데 사용됩니다.
ADsPropSendErrorMessage ADsPropSendErrorMessage 함수는 ADsPropShowErrorDialog 함수를 호출하여 표시되는 오류 메시지 목록에 오류 메시지를 추가합니다.
ADsPropSetHwnd 속성 페이지 창 핸들의 알림 개체에 알리는 데 사용됩니다. (ADsPropSetHwnd)

ADsPropSetHwndWithTitle 속성 페이지 창 핸들의 알림 개체에 알리는 데 사용됩니다. (ADsPropSetHwndWithTitle)
ADsPropShowErrorDialog ADsPropShowErrorDialog 함수는 ADsPropSendErrorMessage 함수 또는 WM_ADSPROP_NOTIFY_ERROR 호출을 통해 누적된 오류 메시지가 포함된 대화 상자를 표시합니다.
시작 알림 처리기가 요청한 이벤트가 발생할 때 IDsAdminNotifyHandler::Begin 메서드가 호출됩니다. 알림 처리기는 IDsAdminNotifyHandler::Initialize가 호출될 때 알림을 받을 이벤트를 지정합니다.
BFFCALLBACK Active Directory Domain Services 컨테이너 브라우저 대화 상자에서 이벤트 알림을 받습니다.
BrowseTo IDsBrowseDomainTree::BrowseTo 메서드는 도메인을 검색하는 데 사용되는 대화 상자를 표시합니다.
지우기 쿼리 저장소의 내용을 비웁니다.
커밋 IDsAdminNewObjPrimarySite::Commit 메서드는 단일 페이지 기본 개체 생성 확장의 IDsAdminNewObjExt::WriteData 메서드를 호출하고 임시 개체를 영구 메모리에 씁니다.
CreateModal IDsAdminCreateObj::CreateModal 메서드는 개체 만들기 마법사를 표시하고 새로 만든 개체를 반환합니다. IDsAdminCreateObj::CreateModal을 호출하려면 먼저 IDsAdminCreateObj::Initialize 메서드를 호출해야 합니다.
CreateNew IDsAdminNewObjPrimarySite::CreateNew 메서드를 사용하면 기본 개체 만들기 확장 프로그램이 Active Directory Domain Services 임시 디렉터리 서비스 개체를 만들 수 있습니다.
DsAddressToSiteNamesA 지정된 주소에 해당하는 사이트 이름을 가져옵니다. (ANSI)

DsAddressToSiteNamesExA 지정된 주소에 해당하는 사이트 및 서브넷 이름을 가져옵니다. (ANSI)
DsAddressToSiteNamesExW 지정된 주소에 해당하는 사이트 및 서브넷 이름을 가져옵니다. (유니코드)
DsAddressToSiteNamesW 지정된 주소에 해당하는 사이트 이름을 가져옵니다. (유니코드)
DsAddSidHistoryA 한 도메인에서 보안 주체의 기본 계정 SID(보안 식별자)를 검색하고 다른 포리스트의 다른 도메인에 있는 보안 주체의 sIDHistory 특성에 추가합니다. (ANSI)
DsAddSidHistoryW 한 도메인에서 보안 주체의 기본 계정 SID(보안 식별자)를 검색하고 다른 포리스트의 다른 도메인에 있는 보안 주체의 sIDHistory 특성에 추가합니다. (유니코드)
DsBindA 도메인 컨트롤러에 바인딩합니다. (ANSI)
DsBindingSetTimeout DsBindingSetTimeout 함수는 지정된 바인딩 핸들을 사용하는 모든 RPC 호출에서 적용되는 시간 제한 값을 설정합니다. 시간 제한 값보다 더 많은 시간이 필요한 RPC 호출이 취소됩니다.
DsBindToISTGA 로컬 컴퓨터의 도메인에서 ISTG(Inter-Site 토폴로지 생성기) 역할을 보유하는 컴퓨터에 바인딩합니다. (ANSI)
DsBindToISTGW 로컬 컴퓨터의 도메인에서 ISTG(Inter-Site 토폴로지 생성기) 역할을 보유하는 컴퓨터에 바인딩합니다. (유니코드)
DsBindW 도메인 컨트롤러에 바인딩합니다. (유니코드)
DsBindWithCredA 지정된 자격 증명을 사용하여 도메인 컨트롤러에 바인딩합니다. (ANSI)

DsBindWithCredW 지정된 자격 증명을 사용하여 도메인 컨트롤러에 바인딩합니다. (유니코드)
DsBindWithSpnA 상호 인증을 위해 지정된 자격 증명 및 특정 SPN(서비스 사용자 이름)을 사용하여 도메인 컨트롤러에 바인딩합니다. (DsBindWithSpnA)
DsBindWithSpnExA 상호 인증을 위해 지정된 자격 증명 및 특정 SPN(서비스 사용자 이름)을 사용하여 도메인 컨트롤러에 바인딩합니다. (DsBindWithSpnExA)
DsBindWithSpnExW 상호 인증을 위해 지정된 자격 증명 및 특정 SPN(서비스 사용자 이름)을 사용하여 도메인 컨트롤러에 바인딩합니다. (DsBindWithSpnExW)
DsBindWithSpnW 상호 인증을 위해 지정된 자격 증명 및 특정 SPN(서비스 사용자 이름)을 사용하여 도메인 컨트롤러에 바인딩합니다. (DsBindWithSpnW)
DsBrowseForContainerA Active Directory Domain Services 컨테이너 개체를 찾는 데 사용되는 대화 상자를 표시합니다. (ANSI)
DsBrowseForContainerW Active Directory Domain Services 컨테이너 개체를 찾는 데 사용되는 대화 상자를 표시합니다. (유니코드)
DsClientMakeSpnForTargetServerA 인증에 사용할 특정 서버를 식별하는 SPN(서비스 사용자 이름)을 생성합니다. (ANSI)
DsClientMakeSpnForTargetServerW 인증에 사용할 특정 서버를 식별하는 SPN(서비스 사용자 이름)을 생성합니다. (유니코드)
DsCrackNamesA 디렉터리 서비스 개체 이름의 배열을 한 형식에서 다른 형식으로 변환합니다. (ANSI)
DsCrackNamesW 디렉터리 서비스 개체 이름의 배열을 한 형식에서 다른 형식으로 변환합니다. (유니코드)

DsCrackSpnA SPN(서비스 사용자 이름)을 해당 구성 요소 문자열로 구문 분석합니다. (ANSI)
DsCrackSpnW SPN(서비스 사용자 이름)을 해당 구성 요소 문자열로 구문 분석합니다. (유니코드)
DsCrackUnquotedMangledRdnA DsCrackUnquotedMangledRdn 함수는 지정된 상대 고유 이름을 unmangles(unencodes)하고 디코딩된 GUID와 사용된 mangling 형식을 모두 반환합니다. (ANSI)
DsCrackUnquotedMangledRdnW DsCrackUnquotedMangledRdn 함수는 지정된 상대 고유 이름을 unmangles(unencodes)하고 디코딩된 GUID와 사용된 mangling 형식을 모두 반환합니다. (유니코드)
DsDeregisterDnsHostRecordsA DsDeregisterDnsHostRecords 함수는 도메인 컨트롤러에 의해 등록된 A 형식 레코드를 제외하고 DNS 항목을 삭제합니다. 관리자, 계정 운영자 또는 서버 운영자만 이 함수를 호출할 수 있습니다. (ANSI)
DsDeregisterDnsHostRecordsW DsDeregisterDnsHostRecords 함수는 도메인 컨트롤러에 의해 등록된 A 형식 레코드를 제외하고 DNS 항목을 삭제합니다. 관리자, 계정 운영자 또는 서버 운영자만 이 함수를 호출할 수 있습니다. (유니코드)
DsEnumerateDomainTrustsA 지정된 도메인에 대한 도메인 신뢰 데이터를 가져옵니다. (ANSI)
DsEnumerateDomainTrustsW 지정된 도메인에 대한 도메인 신뢰 데이터를 가져옵니다. (유니코드)
DsFreeDomainControllerInfoA DsFreeDomainControllerInfo 함수는 도메인의 도메인 컨트롤러에 대한 데이터에 대해 DsGetDomainControllerInfo에서 할당한 메모리를 해제합니다. (ANSI)
DsFreeDomainControllerInfoW DsFreeDomainControllerInfo 함수는 도메인의 도메인 컨트롤러에 대한 데이터에 대해 DsGetDomainControllerInfo에서 할당한 메모리를 해제합니다. (유니코드)

DsFreeNameResultA DS_NAME_RESULT 구조체에서 보유하는 메모리를 해제합니다. (ANSI)
DsFreeNameResultW DS_NAME_RESULT 구조체에서 보유하는 메모리를 해제합니다. (유니코드)
DsFreePasswordCredentials DsMakePasswordCredentials 함수에 의해 자격 증명 구조에 할당된 메모리를 해제합니다.
DsFreeSchemaGuidMapA DsMapSchemaGuids 함수가 DS_SCHEMA_GUID_MAP 구조에 할당한 메모리를 해제합니다. (ANSI)
DsFreeSchemaGuidMapW DsMapSchemaGuids 함수가 DS_SCHEMA_GUID_MAP 구조에 할당한 메모리를 해제합니다. (유니코드)
DsFreeSpnArrayA DsGetSpn 함수에서 반환된 배열을 해제합니다. (ANSI)
DsFreeSpnArrayW DsGetSpn 함수에서 반환된 배열을 해제합니다. (유니코드)
DsGetDcCloseW 도메인 컨트롤러 열거형 작업을 닫습니다.
DsGetDcNameA 지정된 도메인에 있는 도메인 컨트롤러의 이름을 반환합니다. (ANSI)
DsGetDcNameW 지정된 도메인에 있는 도메인 컨트롤러의 이름을 반환합니다. (유니코드)
DsGetDcNextA 도메인 컨트롤러 열거형 작업에서 다음 도메인 컨트롤러를 검색합니다. (ANSI)
DsGetDcNextW 도메인 컨트롤러 열거형 작업에서 다음 도메인 컨트롤러를 검색합니다. (유니코드)

DsGetDcOpenA 새 도메인 컨트롤러 열거형 작업을 엽니다. (ANSI)
DsGetDcOpenW 새 도메인 컨트롤러 열거형 작업을 엽니다. (유니코드)
DsGetDcSiteCoverageA DsGetDcSiteCoverage 함수는 도메인 컨트롤러에서 다루는 모든 사이트의 사이트 이름을 반환합니다. (ANSI)
DsGetDcSiteCoverageW DsGetDcSiteCoverage 함수는 도메인 컨트롤러에서 다루는 모든 사이트의 사이트 이름을 반환합니다. (유니코드)
DsGetDomainControllerInfoA 도메인의 도메인 컨트롤러에 대한 데이터를 검색합니다. (ANSI)
DsGetDomainControllerInfoW 도메인의 도메인 컨트롤러에 대한 데이터를 검색합니다. (유니코드)
DsGetForestTrustInformationW 지정된 도메인에 대한 포리스트 트러스트 데이터를 가져옵니다.
DsGetFriendlyClassName 개체 클래스의 지역화된 이름을 검색합니다.
DsGetIcon 지정된 개체 클래스의 아이콘을 가져옵니다.
DsGetRdnW 첫 번째 상대 고유 이름의 키와 값을 검색하고 고유 이름 문자열에서 다음 상대 고유 이름에 대한 포인터를 검색합니다.
DsGetSiteNameA DsGetSiteName 함수는 컴퓨터가 있는 사이트의 이름을 반환합니다. (ANSI)

DsGetSiteNameW DsGetSiteName 함수는 컴퓨터가 있는 사이트의 이름을 반환합니다. (유니코드)
DsGetSpnA DsGetSpn 함수는 하나 이상의 SPN(서비스 사용자 이름)의 배열을 생성합니다. 배열의 각 이름은 서비스의 인스턴스를 식별합니다. 이러한 SPN은 DsWriteAccountSpn 함수를 사용하여 DS(디렉터리 서비스)에 등록될 수 있습니다. (ANSI)
DsGetSpnW DsGetSpn 함수는 하나 이상의 SPN(서비스 사용자 이름)의 배열을 생성합니다. 배열의 각 이름은 서비스의 인스턴스를 식별합니다. 이러한 SPN은 DsWriteAccountSpn 함수를 사용하여 DS(디렉터리 서비스)에 등록될 수 있습니다. (유니코드)
DsInheritSecurityIdentityA SrcPrincipal의 objectSid 및 sidHistory 특성을 DstPrincipal의 sidHistory에 추가한 다음 SrcPrincipal을 모두 단일 트랜잭션에서 삭제합니다. (ANSI)
DsInheritSecurityIdentityW SrcPrincipal의 objectSid 및 sidHistory 특성을 DstPrincipal의 sidHistory에 추가한 다음 SrcPrincipal을 모두 단일 트랜잭션에서 삭제합니다. (유니코드)
DsIsMangledDnA DsIsMangledDn 함수는 DN(고유 이름)의 첫 번째 RDN(상대 고유 이름)이 지정된 형식의 손상된 이름인지 확인합니다. (ANSI)
DsIsMangledDnW DsIsMangledDn 함수는 DN(고유 이름)의 첫 번째 RDN(상대 고유 이름)이 지정된 형식의 손상된 이름인지 확인합니다. (유니코드)
DsIsMangledRdnValueA 지정된 상대 고유 이름 값이 지정된 형식의 망가진 이름인지 여부를 확인합니다. (ANSI)
DsIsMangledRdnValueW 지정된 상대 고유 이름 값이 지정된 형식의 망가진 이름인지 여부를 확인합니다. (유니코드)
DsListDomainsInSiteA 사이트의 모든 도메인을 나열합니다. (ANSI)

DsListDomainsInSiteW 사이트의 모든 도메인을 나열합니다. (유니코드)
DsListInfoForServerA DsListInfoForServer 함수는 서버에 대한 기타 데이터를 나열합니다. (ANSI)
DsListInfoForServerW DsListInfoForServer 함수는 서버에 대한 기타 데이터를 나열합니다. (유니코드)
DsListRolesA DsListRoles 함수는 서버에서 인식하는 역할을 나열합니다. (ANSI)
DsListRolesW DsListRoles 함수는 서버에서 인식하는 역할을 나열합니다. (유니코드)
DsListServersForDomainInSiteA 사이트의 도메인에 있는 모든 서버를 나열합니다. (ANSI)
DsListServersForDomainInSiteW 사이트의 도메인에 있는 모든 서버를 나열합니다. (유니코드)
DsListServersInSiteA 사이트의 모든 서버를 나열합니다. (ANSI)
DsListServersInSiteW 사이트의 모든 서버를 나열합니다. (유니코드)
DsListSitesA 엔터프라이즈 포리스트의 모든 사이트를 나열합니다. (ANSI)
DsListSitesW 엔터프라이즈 포리스트의 모든 사이트를 나열합니다. (유니코드)
DsMakePasswordCredentialsA DsBindWithCred 함수에 사용하기에 적합한 자격 증명 핸들을 생성합니다. (ANSI)

DsMakePasswordCredentialsW DsBindWithCred 함수에 사용하기에 적합한 자격 증명 핸들을 생성합니다. (유니코드)
DsMakeSpnA 서비스 인스턴스를 식별하는 SPN(서비스 사용자 이름)을 생성합니다. (ANSI)
DsMakeSpnW 서비스 인스턴스를 식별하는 SPN(서비스 사용자 이름)을 생성합니다. (유니코드)
DsMapSchemaGuidsA 디렉터리 서비스 스키마 개체의 GUID를 표시 이름으로 변환합니다. (ANSI)
DsMapSchemaGuidsW 디렉터리 서비스 스키마 개체의 GUID를 표시 이름으로 변환합니다. (유니코드)
DsMergeForestTrustInformationW 새 포리스트 트러스트 데이터 구조의 변경 내용을 이전 포리스트 트러스트 데이터 구조와 병합합니다.
DsQuerySitesByCostA 한 사이트와 하나 이상의 다른 사이트 간의 통신 비용을 가져옵니다. (ANSI)
DsQuerySitesByCostW 한 사이트와 하나 이상의 다른 사이트 간의 통신 비용을 가져옵니다. (유니코드)
DsQuerySitesFree DsQuerySitesByCost 함수에 의해 할당된 메모리를 해제합니다.
DsQuoteRdnValueA RDN 값에 따옴표가 필요한 문자가 포함된 경우 RDN을 따옴표로 묶은 RDN 값으로 변환합니다. (ANSI)
DsQuoteRdnValueW RDN 값에 따옴표가 필요한 문자가 포함된 경우 RDN을 따옴표로 묶은 RDN 값으로 변환합니다. (유니코드)

DsRemoveDsDomainA 디렉터리 서비스의 전역 영역에서 도메인 명명 컨텍스트의 모든 추적을 제거합니다. (ANSI)
DsRemoveDsDomainW 디렉터리 서비스의 전역 영역에서 도메인 명명 컨텍스트의 모든 추적을 제거합니다. (유니코드)
DsRemoveDsServerA DsRemoveDsServer 함수는 디렉터리 서비스의 전역 영역에서 DSA(디렉터리 서비스 에이전트)의 모든 추적을 제거합니다. (ANSI)
DsRemoveDsServerW DsRemoveDsServer 함수는 디렉터리 서비스의 전역 영역에서 DSA(디렉터리 서비스 에이전트)의 모든 추적을 제거합니다. (유니코드)
DsReplicaAddA 대상 명명 컨텍스트에 복제 원본 참조를 추가합니다. (ANSI)
DsReplicaAddW 대상 명명 컨텍스트에 복제 원본 참조를 추가합니다. (유니코드)
DsReplicaConsistencyCheck KCC(기술 일관성 검사기)를 호출하여 복제 토폴로지 확인
DsReplicaDelA 대상 명명 컨텍스트(NC)에서 복제 원본 참조를 제거합니다. (ANSI)
DsReplicaDelW 대상 명명 컨텍스트(NC)에서 복제 원본 참조를 제거합니다. (유니코드)
DsReplicaFreeInfo DsReplicaGetInfo 또는 DsReplicaGetInfo2 함수에 의해 할당된 복제 상태 데이터 구조를 해제합니다.
DsReplicaGetInfo2W 디렉터리 서비스에서 복제 상태 데이터를 검색합니다. 이 함수를 사용하면 검색할 항목이 1,000개 이상인 경우 결과의 페이징을 허용합니다.

DsReplicaGetInfoW 디렉터리 서비스에서 복제 상태 데이터를 검색합니다.
DsReplicaModifyA 대상 명명 컨텍스트에 대한 기존 복제 원본 참조를 수정합니다. (ANSI)
DsReplicaModifyW 대상 명명 컨텍스트에 대한 기존 복제 원본 참조를 수정합니다. (유니코드)
DsReplicaSyncA NC(대상 명명 컨텍스트)를 원본 중 하나와 동기화합니다. (ANSI)
DsReplicaSyncAllA 필요에 따라 전이적 복제를 사용하여 서버를 다른 모든 서버와 동기화합니다. (ANSI)
DsReplicaSyncAllW 필요에 따라 전이적 복제를 사용하여 서버를 다른 모든 서버와 동기화합니다. (유니코드)
DsReplicaSyncW NC(대상 명명 컨텍스트)를 원본 중 하나와 동기화합니다. (유니코드)
DsReplicaUpdateRefsA 원본 명명 컨텍스트에서 대상에 대한 복제 참조를 추가하거나 제거합니다. (ANSI)
DsReplicaUpdateRefsW 원본 명명 컨텍스트에서 대상에 대한 복제 참조를 추가하거나 제거합니다. (유니코드)
DsReplicaVerifyObjectsA 원본을 사용하여 명명 컨텍스트에 대한 모든 개체를 확인합니다. (ANSI)
DsReplicaVerifyObjectsW 원본을 사용하여 명명 컨텍스트에 대한 모든 개체를 확인합니다. (유니코드)
DsRoleFreeMemory DsRoleGetPrimaryDomainInformation 함수에 의해 할당된 메모리를 해제합니다.

DsRoleGetPrimaryDomainInformation 컴퓨터의 상태 데이터를 검색합니다.
DsServerRegisterSpnA DsServerRegisterSpn 함수는 호스트 기반 서비스에 대해 두 개의 SPN을 구성합니다. (ANSI)
DsServerRegisterSpnW DsServerRegisterSpn 함수는 호스트 기반 서비스에 대해 두 개의 SPN을 구성합니다. (유니코드)
DsUnBindA DsUnBind 함수는 도메인 컨트롤러가 있는 RPC 세션을 찾고 DS(디렉터리 서비스)에 대한 핸들을 바인딩 해제합니다. (ANSI)
DsUnBindW DsUnBind 함수는 도메인 컨트롤러가 있는 RPC 세션을 찾고 DS(디렉터리 서비스)에 대한 핸들을 바인딩 해제합니다. (유니코드)
DsUnquoteRdnValueA DsUnquoteRdnValue 함수는 따옴표가 붙은 RDN 값을 따옴표가 없는 RDN 값으로 다시 변환하는 클라이언트 호출입니다. (ANSI)
DsUnquoteRdnValueW DsUnquoteRdnValue 함수는 따옴표가 붙은 RDN 값을 따옴표가 없는 RDN 값으로 다시 변환하는 클라이언트 호출입니다. (유니코드)
DsValidateSubnetNameA DsValidateSubnetName 함수는 xxx.xxx.xxx.xxx/YY 양식의 서브넷 이름의 유효성을 검사합니다. (ANSI)
DsValidateSubnetNameW DsValidateSubnetName 함수는 xxx.xxx.xxx.xxx/YY 양식의 서브넷 이름의 유효성을 검사합니다. (유니코드)
DsWriteAccountSpnA Active Directory Domain Services 지정된 사용자 또는 컴퓨터 계정 개체의 servicePrincipalName 특성에 SPN(서비스 사용자 이름) 배열을 씁니다. (ANSI)

DsWriteAccountSpnW Active Directory Domain Services 지정된 사용자 또는 컴퓨터 계정 개체의 servicePrincipalName 특성에 SPN(서비스 사용자 이름) 배열을 씁니다. (유니코드)
End 알림 이벤트가 발생한 후 IDsAdminNotifyHandler::End 메서드가 호출됩니다. 이 메서드는 알림 프로세스가 취소된 경우에도 호출됩니다.
EnumClassAttributes 지정된 개체 클래스의 특성을 열거합니다.
FlushCachedDomains IDsBrowseDomainTree::FlushCachedDomains 메서드는 캐시된 도메인 목록을 해제합니다.
FreeDomains IDsBrowseDomainTree::FreeDomains 메서드는 IDsBrowseDomainTree::GetDomains 메서드에 의해 할당된 메모리를 해제합니다.
GetAttributeADsType 지정된 특성의 특성 유형을 검색합니다.
GetClassCreationInfo 지정된 개체 클래스에 대한 클래스 만들기 마법사 개체에 대한 데이터를 검색합니다.
GetDisplaySpecifier IDsDisplaySpecifier::GetDisplaySpecifier 메서드는 Active Directory Domain Services 지정된 클래스의 표시 지정자 개체에 바인딩됩니다.
GetDomains IDsBrowseDomainTree::GetDomains 메서드는 현재 컴퓨터의 트러스트 도메인을 검색합니다. 현재 컴퓨터는 IDsBrowseDomainTree::SetComputer 메서드를 사용하여 설정됩니다.
GetFriendlyAttributeName IDsDisplaySpecifier::GetFriendlyAttributeName 메서드는 지정된 개체 클래스의 특성에 대한 지역화된 이름에서 검색합니다.
GetFriendlyClassName IDsDisplaySpecifier::GetFriendlyClassName 메서드는 개체 클래스의 지역화된 이름을 검색합니다.

GetIcon IDsDisplaySpecifier::GetIcon 메서드는 지정된 개체 클래스의 아이콘을 가져옵니다.
GetIconLocation 지정된 개체 클래스의 아이콘 위치를 가져옵니다.
GetPageCounts IDsAdminNewObj::GetPageCounts 메서드는 마법사의 총 페이지 수와 확장의 첫 번째 페이지의 인덱스를 가져옵니다.
GetSummaryInfo IDsAdminNewObjExt::GetSummaryInfo 메서드는 새 개체 마법사 확장 페이지에서 수집한 데이터의 요약을 포함하는 문자열을 가져옵니다. 이 문자열은 마법사 마침 페이지에 표시됩니다.
초기화 쿼리 양식 확장 개체를 초기화합니다.
초기화 IDsObjectPicker::Initialize 메서드는 개체 선택 대화 상자에서 사용되는 범위, 필터 및 옵션에 대한 데이터를 사용하여 개체 선택 대화 상자를 초기화합니다.
초기화 IDsAdminCreateObj::Initialize 메서드는 개체를 만들 컨테이너, 만들 개체의 클래스 및 복사할 원본 개체에 대한 데이터를 사용하여 IDsAdminCreateObj 개체를 초기화합니다.
초기화 IDsAdminNewObjExt::Initialize 메서드는 개체 만들기 마법사 확장을 초기화합니다.
초기화 알림 처리기를 초기화하기 위해 호출되었습니다.
InvokeDialog 모달 개체 선택 대화 상자를 표시하고 사용자 선택을 반환합니다.
IsClassContainer 지정된 개체 클래스가 컨테이너인지 여부를 확인합니다.

LPCQADDFORMSPROC 쿼리 폼 확장에 의해 호출되어 쿼리 대화 상자에 폼을 추가합니다.
LPCQADDPAGESPROC 쿼리 폼 확장에 의해 호출되어 쿼리 대화 상자의 쿼리 폼에 페이지를 추가합니다.
LPCQPAGEPROC 쿼리 페이지에서 발생하는 이벤트의 쿼리 양식 확장에 알리기 위해 쿼리 대화 상자에서 호출됩니다.
LPDSENUMATTRIBUTES DSEnumAttributesCallback 함수는 IDsDisplaySpecifier::EnumClassAttributes 메서드에 의해 열거된 각 특성에 대해 한 번씩 호출되는 애플리케이션 정의 콜백 함수입니다.
알림 확인 대화 상자가 표시되고 확인 대화 상자에서 알림 처리기가 선택된 후 각 개체에 대해 호출됩니다.
OnError 마법사 페이지에서 오류가 발생했을 때 호출됩니다.
OpenQueryWindow ICommonQuery::OpenQueryWindow 메서드는 디렉터리 서비스 쿼리 대화 상자를 표시합니다. 이 메서드는 사용자가 대화 상자를 닫을 때까지 반환되지 않습니다.
ReadInt 쿼리 저장소에서 정수 값을 읽습니다.
ReadString 쿼리 저장소에서 문자열을 읽습니다.
ReadStruct 쿼리 저장소에서 구조를 읽습니다.
SetButtons IDsAdminNewObj::SetButtons 메서드는 마법사에서 특정 페이지에 대해 "다음" 명령 단추를 사용하거나 사용하지 않도록 설정합니다.

SetComputer IDsBrowseDomainTree 인터페이스의 이 인스턴스에서 사용할 컴퓨터 및 자격 증명을 지정합니다.
SetCredentials 이 메서드를 사용하여 사용자 자격 증명을 재정의하고 사용할 계정 프로필에 대한 새 자격 증명을 전달합니다.
SetLanguageID IDsDisplaySpecifier 개체에서 사용하는 로캘을 지정된 언어로 변경합니다.
Setobject IDsAdminNewObjExt::SetObject 메서드는 만든 디렉터리 개체에 대한 포인터를 사용하여 개체 만들기 확장을 제공합니다.
SetServer 표시 지정자 데이터를 가져올 서버를 지정합니다.
WriteData 개체 만들기 마법사 확장이 Active Directory Domain Services 개체에 데이터를 쓸 수 있도록 합니다.
WriteInt 쿼리 저장소에 정수 값을 씁니다.
WriteString 쿼리 저장소에 문자열을 씁니다.
WriteStruct 쿼리 저장소에 구조를 씁니다.

인터페이스

ICommonQuery 프로그래밍 방식으로 시스템 제공 디렉터리 서비스 쿼리 대화 상자를 표시 하는 데 사용 합니다.

IDsAdminCreateObj 애플리케이션 또는 구성 요소에서 지정된 개체 클래스에 대한 만들기 마법사를 프로그래밍 방식으로 시작하는 데 사용됩니다.
IDsAdminNewObj IDsAdminNewObj 인터페이스는 기본 또는 보조 개체 만들기 마법사 확장에서 페이지 수 데이터를 가져오고 마법사에서 명령 단추를 제어하는 데 사용됩니다.
IDsAdminNewObjExt IDsAdminNewObjExt 인터페이스는 개체 만들기 마법사 확장에 의해 구현됩니다.
IDsAdminNewObjPrimarySite IDsAdminNewObjPrimarySite 인터페이스는 시스템에서 구현되며 기본 개체 만들기 확장에서 Active Directory Domain Services 새 임시 개체를 만든 다음 개체를 영구 메모리에 커밋하는 데 사용됩니다.
IDsAdminNotifyHandler IDsAdminNotifyHandler 인터페이스는 Active Directory 관리 알림 처리기에 의해 구현됩니다.
IDsBrowseDomainTree IDsBrowseDomainTree 인터페이스는 애플리케이션에서 도메인 브라우저 대화 상자를 표시하거나 지정된 컴퓨터와 관련된 트러스트 도메인 목록을 가져오는 데 사용됩니다.
IDsDisplaySpecifier displaySpecifier 클래스의 Active Directory 도메인 Service 개체에 대한 액세스를 제공합니다.
IDsObjectPicker IDsObjectPicker 인터페이스는 애플리케이션에서 개체 선택 대화 상자를 초기화하고 표시하는 데 사용됩니다. 이 인터페이스의 인스턴스를 만들려면 아래와 같이 CLSID_DsObjectPicker 클래스 식별자를 사용하여 CoCreateInstance를 호출합니다.
IDsObjectPickerCredentials IDsObjectPickerCredentials 인터페이스를 사용하면 이 인터페이스를 구현하는 IDsObjectPicker 개체에 대한 자격 증명을 재정의할 수 있습니다.
IPersistQuery 영구 스토리지에서 쿼리 매개 변수를 저장하고 검색하는 데 사용됩니다.

IQueryForm 양식 개체가 시스템에서 제공하는 디렉터리 서비스 쿼리 대화 상자에 양식과 페이지를 추가할 수 있도록 쿼리 폼 확장 개체에 의해 구현됩니다.

구조체

ADSPROPERROR ADSPROPERROR 구조는 ADsPropSendErrorMessage 함수 또는 WM_ADSPROP_NOTIFY_ERROR 메시지를 사용하여 알림 개체에 오류 데이터를 전달하는 데 사용됩니다.
ADSPROPINITPARAMS ADsPropGetInitInfo 함수와 함께 사용하여 표시 지정자가 적용되는 개체 데이터를 가져옵니다.
CQFORM CQAddFormsProc 콜백 함수를 사용하여 쿼리 대화 상자에 추가된 쿼리 양식을 정의하는 데 사용됩니다.
CQPAGE CQAddPagesProc 콜백 함수를 사용하여 쿼리 대화 상자의 폼에 추가된 쿼리 페이지를 정의하는 데 사용됩니다.
DOMAIN_CONTROLLER_INFOA DsGetDcName 함수와 함께 도메인 컨트롤러에 대한 데이터를 수신하는 데 사용됩니다. (ANSI)
DOMAIN_CONTROLLER_INFOW DsGetDcName 함수와 함께 도메인 컨트롤러에 대한 데이터를 수신하는 데 사용됩니다. (유니코드)
DOMAIN_DESC IDsBrowseDomainTree::GetDomains 메서드를 사용하여 가져온 도메인 트리의 요소에 대한 데이터를 포함합니다.
DOMAIN_TREE DOMAINTREE 구조에는 IDsBrowseDomainTree::GetDomains 메서드를 사용하여 가져온 도메인 트리의 노드에 대한 데이터가 포함됩니다. 트리 노드의 각 도메인은 DOMAINDESC 구조체로 표시됩니다.

DS_DOMAIN_CONTROLLER_INFO_1A DS_DOMAIN_CONTROLLER_INFO_1 구조에는 도메인 컨트롤러에 대한 데이터가 포함됩니다. 이 구조체는 DsGetDomainControllerInfo 함수에 의해 반환됩니다. (ANSI)
DS_DOMAIN_CONTROLLER_INFO_1W DS_DOMAIN_CONTROLLER_INFO_1 구조에는 도메인 컨트롤러에 대한 데이터가 포함됩니다. 이 구조체는 DsGetDomainControllerInfo 함수에 의해 반환됩니다. (유니코드)
DS_DOMAIN_CONTROLLER_INFO_2A DS_DOMAIN_CONTROLLER_INFO_2 구조에는 도메인 컨트롤러에 대한 데이터가 포함됩니다. 이 구조체는 DsGetDomainControllerInfo 함수에 의해 반환됩니다. (ANSI)
DS_DOMAIN_CONTROLLER_INFO_2W DS_DOMAIN_CONTROLLER_INFO_2 구조에는 도메인 컨트롤러에 대한 데이터가 포함됩니다. 이 구조체는 DsGetDomainControllerInfo 함수에 의해 반환됩니다. (유니코드)
DS_DOMAIN_CONTROLLER_INFO_3A DS_DOMAIN_CONTROLLER_INFO_3 구조에는 도메인 컨트롤러에 대한 데이터가 포함됩니다. 이 구조체는 DsGetDomainControllerInfo 함수에 의해 반환됩니다. (ANSI)
DS_DOMAIN_CONTROLLER_INFO_3W DS_DOMAIN_CONTROLLER_INFO_3 구조에는 도메인 컨트롤러에 대한 데이터가 포함됩니다. 이 구조체는 DsGetDomainControllerInfo 함수에 의해 반환됩니다. (유니코드)
DS_DOMAIN_TRUSTSA DsEnumerateDomainTrusts 함수와 함께 도메인에 대한 신뢰 데이터를 포함하는 데 사용됩니다. (ANSI)
DS_DOMAIN_TRUSTSW DsEnumerateDomainTrusts 함수와 함께 도메인에 대한 신뢰 데이터를 포함하는 데 사용됩니다. (유니코드)
DS_NAME_RESULT_ITEMA DS_NAME_RESULT_ITEM 구조에는 연결된 오류 및 도메인 데이터와 함께 DsCrackNames 함수로 변환된 이름이 포함됩니다. (ANSI)
DS_NAME_RESULT_ITEMW DS_NAME_RESULT_ITEM 구조에는 연결된 오류 및 도메인 데이터와 함께 DsCrackNames 함수로 변환된 이름이 포함됩니다. (유니코드)

DS_NAME_RESULTA DS_NAME_RESULT 구조체는 DsCrackNames 함수와 함께 함수로 변환된 이름을 포함하는 데 사용됩니다. (ANSI)
DS_NAME_RESULTW DS_NAME_RESULT 구조체는 DsCrackNames 함수와 함께 함수로 변환된 이름을 포함하는 데 사용됩니다. (유니코드)
DS_REPL_ATTR_META_DATA DS_REPL_ATTR_META_DATA 구조는 DsReplicaGetInfo 및 DsReplicaGetInfo2 함수와 함께 개체 특성에 대한 복제 상태 데이터를 포함하는 데 사용됩니다.
DS_REPL_ATTR_META_DATA_2 DS_REPL_ATTR_META_DATA_2 구조체는 DsReplicaGetInfo 및 DsReplicaGetInfo2 함수와 함께 개체 특성에 대한 복제 상태 데이터를 포함하는 데 사용됩니다.
DS_REPL_ATTR_META_DATA_BLOB DS_REPL_ATTR_META_DATA_BLOB 구조체는 개체 특성에 대한 복제 상태 데이터를 포함하는 데 사용됩니다.
DS_REPL_ATTR_VALUE_META_DATA DS_REPL_ATTR_VALUE_META_DATA 구조는 DsReplicaGetInfo2 함수와 함께 특성 값 컬렉션에 대한 메타데이터를 제공하는 데 사용됩니다.
DS_REPL_ATTR_VALUE_META_DATA_2 DsReplicaGetInfo2 함수와 함께 특성 값 컬렉션에 대한 메타데이터를 제공하는 데 사용됩니다.
DS_REPL_ATTR_VALUE_META_DATA_EXT 특성 복제 값 컬렉션에 대한 메타데이터를 제공합니다.
DS_REPL_CURSOR DS_REPL_CURSOR 구조에는 DsReplicaGetInfo 및 DsReplicaGetInfo2 함수에서 반환하는 지정된 명명 컨텍스트의 모든 복제본과 관련된 인바운드 복제 상태 데이터가 포함됩니다.
DS_REPL_CURSOR_2 DS_REPL_CURSOR_2 구조에는 DsReplicaGetInfo2 함수에서 반환하는 지정된 명명 컨텍스트의 모든 복제본과 관련된 인바운드 복제 상태 데이터가 포함됩니다. 이 구조체는 DS_REPL_CURSOR 구조체의 향상된 버전입니다.

DS_REPL_CURSOR_3W DS_REPL_CURSOR_3 구조에는 DsReplicaGetInfo2 함수에서 반환하는 지정된 명명 컨텍스트의 모든 복제본과 관련된 인바운드 복제 상태 데이터가 포함됩니다.
DS_REPL_CURSOR_BLOB DS_REPL_CURSOR_BLOB 구조에는 지정된 명명 컨텍스트의 모든 복제본과 관련된 인바운드 복제 상태 데이터가 포함됩니다.
DS_REPL_CURSORS DS_REPL_CURSORS 구조는 지정된 명명 컨텍스트의 모든 복제본과 관련하여 복제 상태 데이터를 제공하기 위해 DsReplicaGetInfo 및 DsReplicaGetInfo2 함수와 함께 사용됩니다.
DS_REPL_CURSORS_2 DS_REPL_CURSORS_2 구조는 지정된 명명 컨텍스트의 모든 복제본과 관련하여 복제 상태 데이터를 제공하기 위해 DsReplicaGetInfo2 함수와 함께 사용됩니다.
DS_REPL_CURSORS_3W DS_REPL_CURSORS_3 구조는 지정된 명명 컨텍스트의 모든 복제본과 관련하여 복제 상태 데이터를 제공하기 위해 DsReplicaGetInfo2 함수와 함께 사용됩니다.
DS_REPL_KCC_DSA_FAILURESW DS_REPL_KCC_DSA_FAILURES 구조체에는 DsReplicaGetInfo 및 DsReplicaGetInfo2 함수에서 반환한 대로 인바운드 복제 파트너와 관련된 복제 상태 데이터가 포함된 DS_REPL_KCC_DSA_FAILURE 구조의 배열이 포함되어 있습니다.
DS_REPL_KCC_DSA_FAILUREW DS_REPL_KCC_DSA_FAILURE 구조에는 DsReplicaGetInfo 및 DsReplicaGetInfo2 함수에서 반환한 특정 인바운드 복제 파트너에 대한 복제 상태 데이터가 포함됩니다.
DS_REPL_KCC_DSA_FAILUREW_BLOB 특정 인바운드 복제 파트너와 관련된 복제 상태 데이터를 포함합니다.
DS_REPL_NEIGHBORSW DS_REPL_NEIGHBORS 구조는 DsReplicaGetInfo 및 DsReplicaGetInfo2 함수와 함께 사용하여 명명 컨텍스트 및 원본 서버 쌍에 대한 인바운드 복제 상태 데이터를 제공합니다.
DS_REPL_NEIGHBORW DsReplicaGetInfo 및 DsReplicaGetInfo2 함수에서 반환한 특정 명명 컨텍스트 및 원본 서버 쌍에 대한 인바운드 복제 상태 데이터를 포함합니다.

DS_REPL_NEIGHBORW_BLOB 특정 명명 컨텍스트 및 원본 서버 쌍에 대한 인바운드 복제 상태 데이터를 포함합니다.
DS_REPL_OBJ_META_DATA DS_REPL_OBJ_META_DATA 구조체에는 DS_REPL_ATTR_META_DATA 구조의 배열이 포함되어 있습니다. 이러한 구조에는 지정된 개체에 대한 과거 및 현재 특성에 대한 복제 상태 데이터가 포함됩니다.
DS_REPL_OBJ_META_DATA_2 DS_REPL_OBJ_META_DATA_2 구조체에는 DsReplicaGetInfo2 함수에서 반환한 대로 지정된 개체에 대한 특성(과거 및 현재)에 대한 복제 상태 데이터가 포함된 DS_REPL_ATTR_META_DATA_2 구조의 배열이 포함되어 있습니다.
DS_REPL_OPW DS_REPL_OP 구조체는 DsReplicaGetInfo 또는 DsReplicaGetInfo2 함수에서 반환한 대로 현재 실행 중이거나 실행 보류 중인 복제 작업을 설명합니다.
DS_REPL_OPW_BLOB DS_REPL_OPW_BLOB 구조는 현재 실행 중이거나 실행 보류 중인 복제 작업을 설명합니다.
DS_REPL_PENDING_OPSW DsReplicaGetInfo 및 DsReplicaGetInfo2 함수에서 반환한 대로 현재 실행되고 큐에 대기 중인 복제 작업을 설명하는 DS_REPL_OP 구조의 배열을 포함합니다.
DS_REPL_QUEUE_STATISTICSW 복제 큐 통계를 포함하는 데 사용됩니다.
DS_REPL_VALUE_META_DATA 특성 값 복제 메타데이터를 포함하기 위해 DS_REPL_ATTR_VALUE_META_DATA 구조와 함께 사용됩니다.
DS_REPL_VALUE_META_DATA_2 특성 값 복제 메타데이터를 포함하기 위해 DS_REPL_ATTR_VALUE_META_DATA_2 구조와 함께 사용됩니다.
DS_REPL_VALUE_META_DATA_BLOB 특성 값 복제 메타데이터를 포함하는 데 사용됩니다.

DS_REPL_VALUE_META_DATA_BLOB_EXT 특성 값 복제 메타데이터를 포함합니다.
DS_REPL_VALUE_META_DATA_EXT DS_REPL_ATTR_VALUE_META_DATA_EXT 구조에 대한 특성 복제 메타 데이터를 포함합니다.
DS_REPSYNCALL_ERRINFOA DS_REPSYNCALL_ERRINFO 구조체는 복제 중에 DsReplicaSyncAll 함수에 의해 생성된 오류를 포함하기 위해 DS_REPSYNCALL_UPDATE 구조체와 함께 사용됩니다. (ANSI)
DS_REPSYNCALL_ERRINFOW DS_REPSYNCALL_ERRINFO 구조체는 복제 중에 DsReplicaSyncAll 함수에 의해 생성된 오류를 포함하기 위해 DS_REPSYNCALL_UPDATE 구조체와 함께 사용됩니다. (유니코드)
DS_REPSYNCALL_SYNCA DS_REPSYNCALL_SYNC 구조는 DsReplicaSyncAll 함수를 통해 원본 서버와 대상 서버 간에 수행되는 단일 복제 작업을 식별합니다. (ANSI)
DS_REPSYNCALL_SYNCW DS_REPSYNCALL_SYNC 구조는 DsReplicaSyncAll 함수를 통해 원본 서버와 대상 서버 간에 수행되는 단일 복제 작업을 식별합니다. (유니코드)
DS_REPSYNCALL_UPDATEA DS_REPSYNCALL_UPDATE 구조에는 DsReplicaSyncAll 함수에서 수행하는 복제에 대한 상태 데이터가 포함됩니다. (ANSI)
DS_REPSYNCALL_UPDATEW DS_REPSYNCALL_UPDATE 구조에는 DsReplicaSyncAll 함수에서 수행하는 복제에 대한 상태 데이터가 포함됩니다. (유니코드)
DS_SCHEMA_GUID_MAPA DsMapSchemaGuids에 대한 호출 결과를 포함합니다. (ANSI)
DS_SCHEMA_GUID_MAPW DsMapSchemaGuids에 대한 호출 결과를 포함합니다. (유니코드)

DS_SELECTION DS_SELECTION 구조에는 사용자가 개체 선택 대화 상자에서 선택한 개체에 대한 데이터가 포함됩니다. DS_SELECTION_LIST 구조체에는 DS_SELECTION 구조의 배열이 포함되어 있습니다.
DS_SELECTION_LIST DS_SELECTION_LIST 구조에는 사용자가 개체 선택 대화 상자에서 선택한 개체에 대한 데이터가 포함됩니다.
DS_SITE_COST_INFO DS_SITE_COST_INFO 구조는 통신 비용 데이터를 포함하기 위해 DsQuerySitesByCost 함수와 함께 사용됩니다.
DSA_NEWOBJ_DISPINFO IDsAdminNewObjExt::Initialize 메서드와 함께 사용하여 Active Directory Domain Services 개체 만들기 마법사에 대한 추가 데이터를 제공합니다.
DSBITEMA Active Directory 컨테이너 브라우저 대화 상자의 항목에 대한 데이터를 포함합니다. (ANSI)
DSBITEMW Active Directory 컨테이너 브라우저 대화 상자의 항목에 대한 데이터를 포함합니다. (유니코드)
DSBROWSEINFOA DSBROWSEINFO 구조체는 DsBrowseForContainer 함수와 함께 Active Directory 컨테이너 브라우저 대화 상자에 대한 데이터를 제공하고 반환하는 데 사용됩니다. (ANSI)
DSBROWSEINFOW DSBROWSEINFO 구조체는 DsBrowseForContainer 함수와 함께 Active Directory 컨테이너 브라우저 대화 상자에 대한 데이터를 제공하고 반환하는 데 사용됩니다. (유니코드)
DSCLASSCREATIONINFO IDsDisplaySpecifier::GetClassCreationInfo 메서드와 함께 개체 클래스에 대한 클래스 만들기 마법사 개체에 대한 데이터를 보관하는 데 사용됩니다.
DSCOLUMN DSCOLUMN 구조는 디렉터리 서비스 쿼리 대화 상자의 열을 나타냅니다. 이 구조체의 배열은 DSQUERYPARAMS 구조체에 포함됩니다.

DSDISPLAYSPECOPTIONS 사용되는 표시 지정자에 대한 컨텍스트 메뉴 또는 속성 페이지 확장에 데이터를 제공하는 데 사용됩니다.
DSOBJECT 디렉터리 개체 데이터를 포함합니다.
DSOBJECTNAMES DSOBJECTNAMES 구조체는 Active Directory 속성 시트 또는 상황에 맞는 메뉴 확장에서 사용할 디렉터리 개체 데이터를 포함하는 데 사용됩니다.
DSOP_FILTER_FLAGS 지정된 범위 또는 범위에 대해 사용자에게 표시되는 개체의 유형을 나타내는 플래그를 포함합니다.
DSOP_INIT_INFO DSOP_INIT_INFO 구조에는 개체 선택 대화 상자를 초기화하는 데 필요한 데이터가 포함되어 있습니다. 이 구조는 IDsObjectPicker::Initialize 메서드와 함께 사용됩니다.
DSOP_SCOPE_INIT_INFO DSOP_SCOPE_INIT_INFO 구조체는 특성이 동일한 하나 이상의 범위 형식을 설명합니다.
DSOP_UPLEVEL_FILTER_FLAGS DSOP_UPLEVEL_FILTER_FLAGS 구조에는 상위 수준 범위에 사용할 필터를 나타내는 플래그가 포함되어 있습니다.
DSPROPERTYPAGEINFO DSPROPERTYPAGEINFO 구조는 Active Directory 속성 시트 확장에서 확장에 대한 정적 등록 데이터를 가져오는 데 사용됩니다. 이 구조체는 CFSTR_DSPROPERTYPAGEINFO 클립보드 형식으로 제공됩니다.
DSQUERYCLASSLIST DSQUERYCLASSLIST 구조는 디렉터리 서비스 쿼리가 만들어지는 클래스 목록을 설명합니다.
DSQUERYINITPARAMS 디렉터리 서비스 쿼리에서 찾아보기 대화 상자를 초기화하는 데 사용되는 데이터에 대해 설명합니다.

DSQUERYPARAMS DSQUERYPARAMS 구조에는 디렉터리 서비스를 검색할 때 디렉터리 서비스 쿼리에서 사용하는 쿼리 데이터가 포함됩니다.
DSROLE_OPERATION_STATE_INFO DsRoleGetPrimaryDomainInformation 함수와 함께 컴퓨터의 작동 상태 데이터를 포함하는 데 사용됩니다.
DSROLE_PRIMARY_DOMAIN_INFO_BASIC DsRoleGetPrimaryDomainInformation 함수와 함께 도메인 데이터를 포함하는 데 사용됩니다.
DSROLE_UPGRADE_STATUS_INFO DsRoleGetPrimaryDomainInformation 함수와 함께 도메인 업그레이드 상태 데이터를 포함하는 데 사용됩니다.
OPENQUERYWINDOW ICommonQuery::OpenQueryWindow 메서드와 함께 디렉터리 서비스 쿼리 대화 상자를 초기화하는 데 사용됩니다.
일정 DsReplicaAdd 및 DsReplicaModify 함수와 함께 복제 원본에 대한 복제 일정 데이터를 포함하는 데 사용됩니다.
SCHEDULE_HEADER 복제 원본에 대한 복제 일정 데이터를 포함하는 데 사용됩니다.

Active Directory LDS(Lightweight Directory Services)

아티클2023. 08. 22.

Active Directory Lightweight Directory Services 기술 개요입니다.

Active Directory Lightweight Directory Services 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Active Directory LDS(Lightweight Directory Services)

Functions

DsBindByInstanceA AD LDS 또는 Active Directory instance 명시적으로 바인딩합니다. (ANSI)
DsBindByInstanceW AD LDS 또는 Active Directory instance 명시적으로 바인딩합니다. (유니코드)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Active Directory Rights Management Services SDK

Active Directory Rights Management Services SDK 기술 개요입니다.

Active Directory Rights Management Services SDK를 개발하려면 다음 헤더가 필요합니다.

- msdrm.h
- msdrmdefs.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Active Directory Rights Management Services SDK

열거 목록

테이블 확장

DRM_DISTRIBUTION_POINT_INFO DRMGetIssuanceLicenseInfo를 호출할 때 정보를 검색할 배포 지점의 유형을 지정합니다.
DRM_STATUS_MSG 사용자 지정 콜백 함수에서 콜백 함수가 호출되는 이유를 지정하는 데 사용됩니다.
DRM_USAGEPOLICY_TYPE DRMGetUsagePolicy 및 DRMSetUsagePolicy 함수와 함께 사용 정책 유형을 지정하는 데 사용됩니다.
DRMATTESTTYPE DRMATTESTTYPE 열거형은 데이터 Blob에 대해 만들 서명의 종류를 지정합니다.
DRMENCODINGTYPE DRMENCODINGTYPE 열거형은 라이선스에 사용되는 가능한 인코딩 형식을 식별합니다.
DRMGLOBALOPTIONS 전송 프로토콜에 사용되는 프로토콜 및 서버 lockbox가 사용되는지 여부를 지정하는 값을 정의합니다. 이 열거형은 DRMSetGlobalOptions 함수에서 사용됩니다.
DRMSECURITYPROVIDERTYPE

DRMSECURITYPROVIDERTYPE 열거형은 사용되는 보안 DRM 환경의 유형을 지정합니다.
DRMSPECTYPE DRMSPECTYPE 열거형은 사용되는 보안 또는 라이브러리 공급자의 유형을 나타냅니다.
DRMTIMETYPE DRMTIMETYPE 열거형은 시간 형식을 지정합니다.

Functions

테이블 확장

DRMAcquireAdvisories 제출된 라이선스에 필요한 해지 목록을 검색합니다.
DRMAcquireIssuanceLicenseTemplate 서버에서 발급 라이선스 템플릿을 비동기적으로 검색합니다.
DRMAcquireLicense 최종 사용자 라이선스 또는 클라이언트 라이선스 인증서를 비동기적으로 획득하려고 시도합니다.
DRMActivate 컴퓨터에 대한 lockbox 및 컴퓨터 인증서 또는 사용자에 대한 권한 계정 인증서를 가져옵니다.
DRMAddLicense 임시 또는 영구 라이선스 저장소에 최종 사용자 라이선스를 추가합니다.
DRMAddRightWithUser 발급 라이선스의 사용자에게 권한을 할당합니다.
DRMAttest DRMAttest 함수는 더 이상 지원되지 않으며 E_NOTIMPL 반환합니다.
DRMCALLBACK

AD RMS SDK에 포함된 일부 함수는 구현해야 하는 콜백 함수를 사용하여 애플리케이션에 상태 정보와 라이선스를 제공합니다. 콜백 구문은 다음과 같습니다.
DRMCheckSecurity 실행 중인 모든 수준의 보안 검사에 대한 S_OK 반환합니다.
DRMClearAllRights 기존 발급 라이선스에서 모든 권한을 제거합니다.
DRMCloseEnvironmentHandle 환경 핸들을 닫습니다.
DRMCloseHandle DRMLoadLibrary를 사용하여 로드된 DRMCreate* 함수 및 라이브러리로 만든 개체에 대한 핸들을 닫습니다.
DRMClosePubHandle 이전에 만든 DRMPUBHANDLE을 닫습니다.
DRMCloseQueryHandle 바인딩되지 않은 라이선스 개체에 대한 핸들을 닫습니다.
DRMCloseSession 클라이언트 세션 또는 라이선스 스토리지 세션을 닫습니다.
DRMConstructCertificateChain 임의의 수의 인증서에서 인증서 체인을 빌드합니다.
DRMCreateBoundLicense 애플리케이션에서 로컬로 저장된 라이선스에 대한 권한을 검사하거나 행사할 수 있습니다.
DRMCreateClientSession 라이선스 스토리지 세션을 호스트하고 활성화 및 기타 함수 호출에 사용되는 클라이언트 세션을 만듭니다.
DRMCreateEnablingBitsDecryptor 콘텐츠 데이터의 암호를 해독하는 데 사용되는 암호 해독 개체를 만듭니다.
DRMCreateEnablingBitsEncryptor

콘텐츠 데이터를 암호화하는 데 사용되는 AD RMS 암호화 개체를 만듭니다.
DRMCreateEnablingPrincipal 라이선스에 바인딩하는 데 필요한 사용 주체를 만듭니다.
DRMCreateIssuanceLicense 처음부터, 템플릿에서 또는 서명된 발급 라이선스에서 발급 라이선스를 만듭니다.
DRMCreateLicenseStorageSession 라이선스를 획득하거나 조작하는 데 필요한 라이선스 스토리지 세션을 만듭니다.
DRMCreateRight 사용자 또는 그룹에 부여된 권한을 정의하는 XrML 권한을 만듭니다.
DRMCreateUser 권한이 부여되는 사용자를 만듭니다.
DRMDecode base64와 같은 공통 알고리즘으로 인코딩된 문자열을 디코딩합니다.
DRMDeconstructCertificateChain 인증서 체인에서 지정된 인증서를 검색합니다.
DRMDecrypt 암호화된 콘텐츠의 암호를 해독합니다.
DRMDeleteLicense 라이선스, 클라이언트 라이선스 라이선스 인증서, 해지 목록 또는 발급 라이선스 템플릿을 삭제합니다.
DRMDuplicateEnvironmentHandle 환경 핸들의 복사본을 만듭니다.
DRMDuplicateHandle DRMHANDLE의 복사본을 만듭니다.
DRMDuplicatePubHandle DRMPUBHANDLE의 복사본을 만듭니다.

DRMDuplicateSession 클라이언트 또는 라이선스 스토리지 세션을 복제합니다.
DRMEncode base64와 같은 공용 인코딩 메서드를 사용하여 데이터를 인코딩합니다.
DRMEncrypt 데이터를 암호화합니다.
DRMEnumerateLicense 유효한 라이선스, 컴퓨터 인증서 또는 권한 계정 인증서, 현재 사용자에 대한 해지 목록 또는 발급 라이선스 템플릿을 열거합니다.
DRMGetApplicationSpecificData 임의의 애플리케이션별 정보의 이름-값 쌍을 검색합니다.
DRMGetBoundLicenseAttribute 라이선스 XrML에서 바인딩된 라이선스 특성을 검색합니다.
DRMGetBoundLicenseAttributeCount 라이선스에서 특성의 발생 횟수를 검색합니다.
DRMGetBoundLicenseObject 바인딩된 라이선스에서 개체를 반환합니다.
DRMGetBoundLicenseObjectCount 라이선스의 지정된 분기 내에서 개체의 발생 횟수를 검색합니다.
DRMGetCertificateChainCount 인증서 체인의 인증서 수를 검색합니다.
DRMGetClientVersion Active Directory Rights Management Services 클라이언트 소프트웨어의 버전 번호와 계층이 프로덕션 또는 사전 프로덕션 용도인지 여부를 반환합니다.
DRMGetEnvironmentInfo 보안 환경에 대한 정보를 반환합니다.

DRMGetInfo 개체 암호화 또는 암호 해독에 대한 정보를 검색합니다.
DRMGetIntervalTime 발급에서 종료되기까지 경과할 수 있는 일 수를 검색합니다. 사용자 라이선스를 갱신해야 합니다.
DRMGetIssuanceLicenseInfo 발급 라이선스에서 다양한 정보를 검색합니다.
DRMGetIssuanceLicenseTemplate 기존 발급 라이선스에서 발급 라이선스 템플릿을 가져옵니다.
DRMGetMetaData 발급 라이선스에서 메타데이터를 검색합니다.
DRMGetNameAndDescription 발급 라이선스에서 언어별 이름 및 설명을 검색합니다.
DRMGetOwnerLicense DRMGetSignedIssuanceLicense를 호출하여 만든 소유자 라이선스를 검색합니다.
DRMGetProcAddress 라이브러리에 있는 함수의 주소를 반환합니다. GetProcAddress 함수의 보안 버전입니다.
DRMGetRevocationPoint 발급 라이선스의 해지 지점에 대한 정보를 검색합니다.
DRMGetRightExtendedInfo 오른쪽에 연결된 사용자 지정 이름-값 쌍을 검색합니다.
DRMGetRightInfo 이전에 만든 권한에 대한 정보를 가져옵니다.
DRMGetSecurityProvider lockbox의 경로를 검색합니다.
DRMGetServiceLocation

활성화 또는 라이선스 취득과 같은 다양한 권한 관리 서비스를 수행할 수 있는 서버의 URL을 검색합니다.
DRMGetSignedIssuanceLicense 온라인 또는 오프라인에서 서명된 발급 라이선스를 획득하거나 나중에 서명할 수 있는 서명되지 않은 발급 라이선스를 생성합니다.
DRMGetSignedIssuanceLicenseEx 오프라인에서 서명된 발급 라이선스를 획득합니다.
DRMGetTime 권한 관리 시스템에서 보안 시간을 요청합니다.
DRMGetUnboundLicenseAttribute 기본 XrML에서 언바운드 라이선스 특성을 검색합니다.
DRMGetUnboundLicenseAttributeCount 언바운드 라이선스의 개체 내에서 특성이 발생하는 횟수를 검색합니다.
DRMGetUnboundLicenseObject 언바운드 라이선스에서 지정된 형식의 개체를 검색합니다.
DRMGetUnboundLicenseObjectCount 라이선스의 지정된 분기 내에 있는 개체의 인스턴스 수를 계산합니다.
DRMGetUsagePolicy 애플리케이션 이름, 버전 또는 기타 애플리케이션 특성에 따라 콘텐츠에 대한 액세스가 필요하거나 거부되는 사용 정책을 가져옵니다.
DRMGetUserInfo 사용자에 대한 정보를 가져옵니다.
DRMGetUserRights 발급 라이선스에서 사용자/오른쪽 쌍을 검색합니다.
DRMGetUsers 발급 라이선스에서 특정 사용자를 검색합니다.
DRMInitEnvironment

모든 권한 관리 호출에 대한 보안 환경을 만듭니다.
DRMIsActivated 현재 사용자 또는 컴퓨터가 활성화되었는지 여부를 나타냅니다.
DRMIsWindowProtected 창이 보호된 환경과 연결되어 있는지 여부를 나타냅니다.
DRMLoadLibrary 자격 증명에 의해 결정된 대로 승인된 라이브러리에 핸들을 로드합니다.
DRMParseUnboundLicense 애플리케이션이 개체 및 특성을 탐색할 수 있도록 언바운드 라이선스에 대한 핸들을 만듭니다.
DRMRegisterContent AD RMS(Active Directory Rights Management Services) 클라이언트에 AD RMS로 보호되는 문서가 있거나 더 이상 표시되지 않음을 알릴 수 있습니다.
DRMRegisterProtectedWindow 보호된 환경에 창을 등록합니다.
DRMRegisterRevocationList 클라이언트에 권한 해지 목록을 등록합니다.
DRMRepair 컴퓨터 또는 사용자에 대해 이전에 만든 인증서를 삭제하여 클라이언트 컴퓨터를 복구합니다.
DRMSetApplicationSpecificData 발급 라이선스가 콘텐츠 소비 애플리케이션에서 사용할 임의의 이름-값 쌍을 저장할 수 있도록 허용합니다.
DRMSetGlobalOptions 전송 프로토콜을 지정된 값으로 설정하고 필요에 따라 서버 잠금 상자가 사용되는지 여부를 지정합니다.
DRMSetIntervalTime 발급이 종료되기까지 경과할 수 있는 일 수를 지정합니다. 사용자 라이선스를 갱신해야 합니다.
DRMSetMetaData

발급 라이선스에 애플리케이션별 메타데이터를 추가합니다.
DRMSetNameAndDescription 애플리케이션에서 여러(인간) 언어로 라이선스의 이름과 설명을 지정할 수 있습니다.
DRMSetRevocationPoint 해지 목록을 가져올 새로 고침 속도 및 위치를 설정합니다.
DRMSetUsagePolicy 애플리케이션 이름, 버전 또는 기타 환경 특성에 따라 콘텐츠에 대한 액세스를 요구하거나 거부하는 사용 정책을 설정합니다.
DRMVerify 더 이상 지원되지 않으며 E_NOTIMPL 반환합니다.

구조

테이블 확장

DRM_ACTSERV_INFO DRM_ACTSERV_INFO 구조는 정품 인증 서버에 대한 정보를 저장합니다.
DRM_CLIENT_VERSION_INFO AD RMS(Active Directory Rights Management Services) 클라이언트 버전 및 프로덕션 또는 사전 프로덕션과 같은 계층 구조에 대한 정보를 받습니다.
DRM_LICENSE_ACQ_DATA 비실용 라이선스 취득 중에 라이선스 취득 데이터를 보유합니다.
DRMBOUNDLICENSEPARAMS DRMCreateBoundLicense에서 라이선스에 바인딩하는 데 사용됩니다.
DRMID 개체를 식별합니다.

Last updated on 2025. 10. 08.

Active Directory 서비스 인터페이스

아티클2023. 03. 09.

Active Directory 서비스 인터페이스 기술 개요입니다.

Active Directory 서비스 인터페이스를 개발하려면 다음 헤더가 필요합니다.

- adshlp.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Active Directory 서비스 인터페이스
- 용어

열거형

ADS_ACEFLAG_ENUM ADS_ACEFLAG_ENUM 열거형은 Active Directory 개체에 대한 ACE(Access Control Entry)의 동작을 지정하는 데 사용됩니다.
ADS_ACETYPE_ENUM Active Directory 개체에 대한 액세스 제어 항목의 형식을 지정하는 데 사용됩니다.
ADS_AUTHENTICATION_ENUM 디렉터리 서비스 개체에 바인딩하기 위해 ADSI에서 사용되는 인증 옵션을 지정합니다.
ADS_CHASE_REFERRALS_ENUM ADS_CHASE_REFERRALS_ENUM 열거형은 조회 추적이 발생하는 경우와 방법을 지정합니다.
ADS_DEREFENUM ADS_DEREFENUM 열거형은 별칭이 역참조되는 프로세스를 지정합니다.
ADS_DISPLAY_ENUM ADS_DISPLAY_ENUM 열거형은 경로를 표시하는 방법을 지정합니다.
ADS_ESCAPE_MODE_ENUM 디렉터리 경로에 이스케이프 문자가 표시되는 방법을 지정합니다.

ADS_FLAGTYPE_ENUM ADS_FLAGTYPE_ENUM 열거형은 ACE(액세스 제어 항목)에 ObjectType 또는 InheritedObjectType 필드가 있음을 나타내는 데 사용할 수 있는 값을 지정합니다.
ADS_FORMAT_ENUM IADsPathname::Retrieve 메서드에서 사용하는 사용 가능한 경로 값 형식을 지정합니다.
ADS_GROUP_TYPE_ENUM ADSI에서 그룹 개체의 형식을 지정합니다.
ADS_NAME_INITTYPE_ENUM ADS_NAME_INITTYPE_ENUM 열거형은 NameTranslate 개체에서 수행할 초기화 형식을 지정합니다. IADsNameTranslate 인터페이스에서 사용됩니다.
ADS_NAME_TYPE_ENUM 고유 이름을 나타내는 데 사용되는 형식을 지정합니다.
ADS_OPTION_ENUM IADsObjectOptions.GetOption 및 IADsObjectOptions.SetOption 메서드를 사용하여 검색하거나 설정할 수 있는 옵션을 나타내는 값을 포함합니다.
ADS_PASSWORD_ENCODING_ENUM IADsObjectOptions::GetOption 및 IADsObjectOptions::SetOption 메서드의 ADS_OPTION_PASSWORD_METHOD 옵션과 함께 사용되는 암호 인코딩 유형을 식별합니다.
ADS_PATHTYPE_ENUM ADS_PATHTYPE_ENUM 열거형은 IADsSecurityUtility 인터페이스가 보안 설명자를 추가하거나 수정할 개체의 형식을 지정합니다.
ADS_PREFERENCES_ENUM ADS_PREFERENCES_ENUM 열거형은 ADSI용 OLE DB 공급자의 쿼리 기본 설정을 지정합니다.
ADS_PROPERTY_OPERATION_ENUM 캐시에서 명명된 속성을 업데이트하는 방법을 지정합니다.
ADS_RIGHTS_ENUM Active Directory 개체에 할당된 액세스 권한을 지정합니다.

ADS_SCOPEENUM 디렉터리 검색의 범위를 지정합니다.
ADS_SD_CONTROL_ENUM ADS_SD_CONTROL_ENUM 열거형은 보안 설명자에 대한 컨트롤 플래그를 지정합니다.
ADS_SD_FORMAT_ENUM ADS_SD_FORMAT_ENUM 열거형은 개체의 보안 설명자가 IADsSecurityUtility 인터페이스로 변환되는 형식을 지정합니다.
ADS_SD_REVISION_ENUM Active Directory에 대한 ACE(액세스 제어 항목) 또는 ACL(액세스 제어 목록)의 수정 번호를 지정합니다.
ADS_SEARCHPREF_ENUM IDirectorySearch 개체에 대한 기본 설정을 지정합니다.
ADS_SECURITY_INFO_ENUM 개체의 보안 데이터를 검사하는 데 사용할 수 있는 옵션을 지정합니다.
ADS_SETTYPE_ENUM ADS_SETTYPE_ENUM 열거형은 IADsPathname::Set 메서드에서 사용하는 사용 가능한 경로 이름 형식을 지정합니다.
ADS_STATUSENUM IDirectorySearch::SetSearchPreference 메서드를 사용하여 검색 기본 설정 집합의 상태를 지정합니다.
ADS_SYSTEMFLAG_ENUM ADS_SYSTEMFLAG_ENUM 열거형은 systemFlags 특성에 할당할 수 있는 일부 값을 정의합니다. 열거형의 일부 값은 attributeSchema 개체와 관련이 있습니다. 다른 값은 모든 클래스의 개체에 설정할 수 있습니다.
ADS_USER_FLAG_ENUM 디렉터리에서 사용자 속성을 설정하는 데 사용되는 플래그를 정의합니다.
ADSI_DIALECT_ENUM ADSI_DIALECT_ENUM 열거형은 ADSI용 OLE DB 공급자에 사용되는 쿼리 언어를 지정합니다.

ADSTYPEENUM ADSI 속성 값의 데이터 형식을 식별하는 데 사용됩니다.

함수

AbandonSearch IDirectorySearch::AbandonSearch 메서드는 ExecuteSearch 메서드에 대한 이전 호출로 시작된 검색을 중단합니다.
추가 컬렉션에 명명된 항목을 추가합니다.
추가 기존 그룹에 ADSI 개체를 추가합니다.
AddAce IADsAccessControlList::AddAce 메서드는 IADsAccessControlEntry 개체를 IADsAccessControlList 개체에 추가합니다.
AddLeafElement Pathname 개체에 이미 설정된 디렉터리 경로의 끝에 요소를 추가합니다.
ADsBuildEnumerator ADsBuildEnumerator 함수는 지정된 ADSI 컨테이너 개체에 대한 열거자 개체를 만듭니다.
ADsBuildVarArrayInt ADsBuildVarArrayInt 함수는 DWORD 값 배열에서 정수의 변형 배열을 빌드합니다.
ADsBuildVarArrayStr ADsBuildVarArrayStr 함수는 유니코드 문자열 배열에서 변형 배열을 빌드합니다.
ADsEncodeBinaryData ADsEncodeBinaryData 함수는 BLOB(Binary Large Object)을 검색 필터에 포함하기에 적합한 유니코드 형식으로 변환합니다.

ADsEnumerateNext ADsEnumerateNext 함수는 열거자의 현재 커서 위치에서 지정된 개수의 요소를 열거합니다.
ADsFreeEnumerator ADsBuildEnumerator 함수를 사용하여 만든 열거자 개체를 해제합니다.
ADsGetLastError ADsGetLastError 함수는 호출 스레드의 마지막 오류 코드 값을 검색합니다.
ADsGetObject 해당 경로 및 지정된 인터페이스 식별자가 지정된 개체에 바인딩합니다.
ADsOpenObject 명시적 사용자 이름 및 암호 자격 증명을 사용하여 ADSI 개체에 바인딩합니다.
ADsSetLastError ADsSetLastError는 호출 스레드에 대한 마지막 오류 코드 값을 설정합니다.
AllocADsMem 지정된 크기의 메모리 블록을 할당합니다.
AllocADsStr 지정된 문자열에 대한 메모리를 할당하고 복사합니다.
BinarySDToSecurityDescriptor 이진 보안 설명자를 IADsSecurityDescriptor 개체로 변환합니다.
ChangePassword 사용자 암호를 지정된 이전 값에서 새 값으로 변경합니다.
지우기 속성 값 개체의 현재 값을 지웁니다.
CloseSearchHandle IDirectorySearch::CloseSearchHandle 메서드는 검색 결과에 대한 핸들을 닫고 연결된 메모리를 해제합니다.

계속 IADsServiceOperations::Continue 메서드는 IADsServiceOperations::P ause 메서드에 의해 일시 중지된 서비스 작업을 다시 시작합니다.
ConvertSecurityDescriptor 보안 설명자를 한 형식에서 다른 형식으로 변환합니다.
CopyAccessList IADsAccessControlList::CopyAccessList 메서드는 ACL(액세스 제어 목록)의 모든 ACE(액세스 제어 항목)를 호출자의 프로세스 공간에 복사합니다.
CopyAcl IADsAcl::CopyAcl 메서드는 기존 ACL의 복사본을 만듭니다.
CopyHere IADsContainer::CopyHere 메서드는 이 컨테이너에 지정된 디렉터리 개체의 복사본을 만듭니다.
CopyPath Pathname 개체의 복사본을 만듭니다.
CopySecurityDescriptor IADsSecurityDescriptor::CopySecurityDescriptor 메서드는 개체에 대한 보안 데이터를 보유하는 ADSI 보안 설명자 개체를 복사합니다.
만들기 지정된 스키마 클래스의 디렉터리 개체와 컨테이너에 지정된 이름을 만드는 요청을 설정합니다.
CreateDSObject 현재 디렉터리 서비스 개체의 자식 개체를 만듭니다.
삭제 이 컨테이너에서 지정된 디렉터리 개체를 삭제합니다.
DeleteDSObject 디렉터리 트리에서 리프 개체를 삭제합니다.

DeleteObject IADsDeleteOps::D eleteObject 메서드는 ADSI 개체를 삭제합니다.
ExecuteSearch IDirectorySearch::ExecuteSearch 메서드는 검색을 실행하고 결과를 호출자에게 전달합니다.
FreeADsMem AllocADsMem 또는 ReallocADsMem에서 할당한 메모리를 해제합니다.
FreeADsStr AllocADsStr 또는 ReallocADsStr에서 할당한 문자열의 메모리를 해제합니다.
FreeColumn IDirectorySearch::FreeColumn 메서드는 IDirectorySearch::GetColumn 메서드가 열의 데이터에 할당된 메모리를 해제합니다.
가져오기 속성 캐시에서 지정된 이름의 속성을 검색합니다.
가져오기 디렉터리 개체의 이름을 지정된 형식으로 검색합니다.
get__NewEnum IADsAccessControlList::get__NewEnum 메서드는 ACL에서 ACE를 열거하는 열거자 개체를 가져오는 데 사용됩니다.
get__NewEnum IADsCollection::get__NewEnum 메서드는 이 ADSI 컬렉션 개체에 대해 IEnumVARIANT를 구현하는 종속 열거자 개체를 가져옵니다. 함수 이름(get__NewEnum)에 두 개의 밑줄 문자가 있습니다.
get__NewEnum 컨테이너의 열거자 개체를 검색합니다.
get__NewEnum IADsMembers::get__NewEnum 메서드는 이 ADSI 컬렉션 개체에 대해 IEnumVARIANT를 구현하는 종속 열거자 개체를 가져옵니다. 함수 이름(get__NewEnum)에 두 개의 밑줄 문자가 있습니다.

get_SecurityMask 검색하거나 설정할 보안 설명자의 요소를 결정합니다. (Get)
GetAnyDCName 로컬 컴퓨터의 도메인에서 도메인 컨트롤러의 DNS 이름을 검색합니다.
GetColumn IDirectorySearch::GetColumn 메서드는 검색 결과의 명명된 열에서 데이터를 가져옵니다.
GetDCSiteName 로컬 컴퓨터가 포함된 Active Directory 사이트의 이름을 검색합니다.
GetElement 디렉터리 경로의 요소를 검색합니다.
GetEscapedElement 입력 경로에서 특수 문자를 이스케이프하는 데 사용됩니다.
GetEx 속성 캐시에서 지정된 특성의 속성 값을 검색합니다.
GetEx 지정된 형식의 개체 이름을 가져옵니다.
GetFirstRow GetFirstRow 메서드는 검색 결과의 첫 번째 행을 가져옵니다. 이 메서드는 이전에 이 메서드를 호출한 경우에도 새 검색을 실행하거나 다시 실행합니다.
GetInfo 기본 디렉터리 저장소에서 이 ADSI 개체의 지원되는 속성의 속성 캐시 값으로 로드합니다.
GetInfoEx IADs::GetInfoEx 메서드는 기본 디렉터리 저장소에서 속성 캐시로 ADSI 개체의 지정된 속성 값을 로드합니다.

GetNextColumnName IDirectorySearch::GetNextColumnName 메서드는 데이터가 포함된 검색 결과에서 다음 열의 이름을 가져옵니다.
GetNextRow 검색 결과의 다음 행을 가져옵니다.
GetNumElements 경로의 요소 수를 검색합니다.
GetObject 컬렉션의 항목을 검색합니다.
GetObject 컨테이너의 디렉터리 개체에 대한 인터페이스를 검색합니다.
GetObjectAttributes 디렉터리 서비스 개체의 지정된 특성을 하나 이상 검색합니다.
GetObjectInformation IDirectoryObject::GetObjectInformation 메서드는 디렉터리 서비스 개체의 ID 및 위치에 관한 데이터를 포함하는 ADS_OBJECT_INFO 구조체에 대한 포인터를 검색합니다.
GetObjectProperty 특성 값을 검색합니다.
GetOption 디렉터리 개체에 대한 공급자별 옵션을 가져옵니다.
GetPreviousRow IDirectorySearch::GetPreviousRow 메서드는 검색 결과의 이전 행을 가져옵니다. 공급자가 커서 지원을 제공하지 않으면 E_NOTIMPL 반환해야 합니다.
GetPropertyItem 목록에서 이름과 일치하는 항목을 검색합니다.

GetSecurityDescriptor 지정된 파일, 파일 공유 또는 레지스트리 키에 대한 보안 설명자를 검색합니다.
GetTrees 로컬 컴퓨터의 포리스트에 있는 모든 디렉터리 트리의 DNS 이름을 검색합니다.
그룹 이 사용자가 속한 ADSI 그룹 개체의 컬렉션을 가져옵니다.
Init 현재 사용자의 자격 증명을 사용하여 지정된 디렉터리 서버, 도메인 또는 글로벌 카탈로그에 바인딩하여 이름 변환 개체를 초기화합니다.
InitEx 지정된 사용자 자격 증명을 사용하여 지정된 디렉터리 서버, 도메인 또는 글로벌 카탈로그에 바인딩하여 이름 변환 개체를 초기화합니다.
IsMember 디렉터리 서비스 개체가 그룹의 직접 구성원인지 여부를 확인합니다.
Item IADsPropertyList::Item 메서드는 목록에서 지정된 속성 항목을 검색합니다.
멤버 그룹의 직행 멤버 컬렉션을 검색합니다.
MoveHere 지정된 개체를 이 인터페이스를 구현하는 컨테이너로 이동합니다.
다음 IADsPropertyList::Next 메서드는 속성 목록의 다음 항목을 가져옵니다. 반환된 항목이 속성 항목 개체입니다.
OpenDSObject 지정된 자격 증명을 사용하여 ADSI 개체에 바인딩하고 지정된 개체에 대한 IDispatch 포인터를 검색합니다.

운영 공급자의 사양에 따라 제어 코드 및 입력 매개 변수를 해석합니다.
일시 중지 IADsPrintJobOperations::P ause 메서드는 현재 인쇄 작업의 처리를 중지합니다. IADsPrintJobOperations::Resume 메서드를 호출하여 처리를 계속합니다.
일시 중지 IADsPrintQueueOperations::P ause 메서드는 인쇄 큐 서비스 내에서 인쇄 작업의 처리를 일시 중단합니다.
일시 중지 IADsServiceOperations::P ause 메서드는 IADsServiceOperations::Start 메서드로 시작된 서비스를 일시 중지합니다.
PrintJobs IADsPrintQueueOperations::P rintJobs 메서드는 이 인쇄 큐에서 처리된 인쇄 작업의 컬렉션에 대한 IADsCollection 인터페이스 포인터를 가져옵니다.
PrivateGetIDsOfNames IADsExtension::P rivateGetIDsOfNames 메서드는 ADSI 집계기에서 호출되며, ADSI는 확장이 이중 또는 디스패치 인터페이스를 지원하는 데 사용됨을 확인합니다. 이 메서드는 형식 데이터를 사용하여 IDispatch::GetIDsOfNames를 사용하여 DISPID를 가져올 수 있습니다.
PrivateInvoke IADsExtension::P rivateInvoke 메서드는 일반적으로 IADsExtension::P rivateGetIDsOfNames 메서드 이후 ADSI에서 호출됩니다. 이 메서드는 사용자 지정 구현을 사용하거나 IDispatch::D ispInvoke 메서드에 작업을 위임할 수 있습니다.
제거 IADsPrintQueueOperations::P urge 메서드는 모든 인쇄 작업의 인쇄 큐를 처리하지 않고 지웁니다.
PurgePropertyList 속성 목록에서 모든 항목을 삭제합니다.
넣어 ADSI 특성 캐시에 있는 특성의 값을 설정합니다.

put_SecurityMask 검색하거나 설정할 보안 설명자의 요소를 결정합니다. (Put)
PutEx ADSI 특성 캐시의 특성 값을 수정합니다.
PutObjectProperty 특성 값을 설정합니다.
PutPropertyItem 속성 목록의 항목 값을 업데이트.
한정자 이 스키마 클래스에 대한 추가 한정자를 설명하는 ADSI 개체의 컬렉션을 반환합니다.
한정자 이 속성의 추가 한정자를 설명하는 ADSI 개체의 컬렉션을 반환합니다.
ReallocADsMem 기존 메모리 블록을 다시 할당하고 복사합니다.
ReallocADsStr 유니코드 문자열의 복사본을 만듭니다.
RefreshSchemaCache IADsADSystemInfo::RefreshSchemaCache 메서드는 Active Directory 스키마 캐시를 새로 고칩니다.
제거 IADsCollection::Remove 메서드는 이 ADSI 컬렉션 개체에서 명명된 항목을 제거합니다.
제거 IADsGroup::Remove 메서드는 이 그룹에서 지정된 사용자 개체를 제거합니다. 이 작업은 그룹에 남아 있는 멤버가 없는 경우에도 그룹 개체 자체를 제거하지 않습니다.
RemoveAce ACL(액세스 제어 목록)에서 ACE(액세스 제어 항목)를 제거합니다.

RemoveLeafElement Pathname 개체에 설정된 디렉터리 경로에서 마지막 요소를 제거합니다.
재설정 목록을 첫 번째 항목으로 다시 설정합니다.
ResetPropertyItem 목록에서 지정된 항목을 제거합니다. 즉, 캐시에서
리소스 IADsFileServiceOperations::Resources 메서드는 이 파일 서비스의 현재 열려 있는 리소스를 나타내는 리소스 개체 컬렉션에서 IADsCollection 인터페이스에 대한 포인터를 가져옵니다.
재개 IADsPrintJobOperations::Resume 메서드는 IADsPrintJobOperations::P ause 메서드에 의해 중단된 인쇄 작업을 계속합니다.
재개 IADsPrintQueueOperations::Resume 메서드는 인쇄 큐에서 일시 중단된 인쇄 작업의 처리를 다시 시작합니다.
장치 IADsPathname::Retrieve 메서드는 형식이 다른 개체의 경로를 검색합니다.
SecurityDescriptorToBinarySD IADsSecurityDescriptor 개체를 이진 보안 설명자 형식으로 변환합니다.
세션 IADsFileServiceOperations::Sessions 메서드는 이 파일 서비스에 대한 현재 열려 있는 세션을 나타내는 세션 개체 컬렉션에서 IADsCollection 인터페이스에 대한 포인터를 가져옵니다.
설정 디렉터리 서비스에 이름 번역을 위해 지정된 개체를 설정하도록 지시합니다.
설정 디렉터리 경로를 구문 분석하기 위한 Pathname 개체를 설정합니다.

SetDisplayType 개체의 경로를 표시하는 방법을 지정합니다.
SetEx 이름 변환을 위한 개체 배열을 설정합니다.
SetInfo IADs::SetInfo 메서드는 ADSI 개체의 캐시된 속성 값을 기본 디렉터리 저장소에 저장합니다.
SetObjectAttributes IDirectoryObject::SetObjectAttributes 메서드는 ADS_ATTR_INFO 구조에 정의된 하나 이상의 지정된 개체 특성에서 데이터를 수정합니다.
Setoption 디렉터리 개체를 조작하기 위한 공급자별 옵션을 설정합니다.
SetPassword IADsServiceOperations::SetPassword 메서드는 서비스 관리자가 사용하는 계정의 암호를 설정합니다. 이 메서드는 이 서비스에 대한 보안 컨텍스트를 만들 때 호출됩니다.
SetPassword 사용자 암호를 지정된 값으로 설정합니다.
SetSearchPreference 후속 검색에서 데이터를 가져오기 위한 검색 기본 설정을 지정합니다.
SetSecurityDescriptor 지정된 파일, 파일 공유 또는 레지스트리 키에 대한 보안 설명자를 설정합니다.
종료 IADsComputerOperations::Shutdown 메서드를 사용하면 ADSI 컨트롤 아래의 컴퓨터가 선택적 재부팅으로 종료 작업을 실행합니다.
skip 속성 목록의 현재 커서 위치에서 계산하여 지정된 개수의 항목을 건너뜁니다.

시작 IADsServiceOperations::Start 메서드는 네트워크 서비스를 시작합니다.
상태 IADsComputerOperations::Status 메서드는 컴퓨터의 상태를 검색합니다.
중지 IADsServiceOperations::Stop 메서드는 현재 활성 네트워크 서비스를 중지합니다.

인터페이스

IAD IADs 인터페이스는 ADSI 개체의 기본 개체 기능, 즉 속성 및 메서드를 정의합니다.
IADsAccessControlEntry IADsAccessControlEntry 인터페이스는 디렉터리 클라이언트가 소유 개체의 개별 ACE(액세스 제어 항목)에 액세스하고 조작할 수 있도록 하는 이중 인터페이스입니다.
IADsAccessControlList IADsAccessControlList 인터페이스는 개별 ACE(액세스 제어 항목)를 관리하는 이중 인터페이스입니다.
IADsAcl IADsAcl 인터페이스는 ADSI 클라이언트가 ACL 또는 상속된 ACL 특성 값에 액세스하고 조작할 수 있는 메서드를 제공합니다. 이 인터페이스는 특성을 조작합니다.
IADsADSystemInfo IADsADSystemInfo 인터페이스는 Windows 도메인에서 Windows 운영 체제를 실행하는 경우 로컬 컴퓨터에 대한 데이터를 검색합니다. 예를 들어 로컬 컴퓨터의 도메인, 사이트 및 고유 이름을 가져올 수 있습니다.
IADsBackLink IADsBackLink 인터페이스는 ADSI 클라이언트가 Back Link 특성에 액세스하는 메서드를 제공합니다. 이 인터페이스의 속성 메서드를 호출하여 특성을 가져오고 수정할 수 있습니다.

IADsCaseIgnoreList IADsCaseIgnoreList 인터페이스는 ADSI 클라이언트가 사례 무시 목록 특성에 액세스하는 메서드를 제공합니다. 이 인터페이스의 속성 메서드를 호출하여 특성을 가져오고 수정할 수 있습니다.
IADsClass IADsClass 인터페이스는 ADSI 개체에 대한 클래스 정의를 제공하는 스키마 클래스 개체를 관리하도록 설계되었습니다. 다른 스키마 관리 인터페이스에는 특성 정의에 대한 IADsProperty 및 특성 구문에 대한 IADsSyntax가 포함됩니다.
IADsCollection IADsCollection 인터페이스는 호스팅 ADSI 개체가 디렉터리 서비스에 대해 임의의 명명된 데이터 요소 집합을 정의하고 관리할 수 있도록 하는 이중 인터페이스입니다.
IADsComputer IADsComputer 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다.
IADsComputerOperations IADsComputerOperations 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다.
IADsContainer IADsContainer 인터페이스를 사용하면 ADSI 컨테이너 개체가 포함된 ADSI 개체를 만들고, 삭제하고, 관리할 수 있습니다. 컨테이너 개체는 파일 시스템과 같은 계층적 디렉터리 트리를 나타내고 디렉터리 계층 구조를 구성합니다.
IADsDeleteOps IADsDeleteOps 인터페이스는 개체가 기본 디렉터리에서 자신을 삭제하는 데 사용할 수 있는 메서드를 지정합니다. 컨테이너 개체의 경우 메서드는 자식 및 전체 하위 트리를 삭제합니다.
IADsDNWithBinary IADsDNWithBinary 인터페이스는 ADSI 클라이언트가 DN(고유 이름)을 개체의 GUID와 연결하는 메서드를 제공합니다.
IADsDNWithString IADsDNWithString 인터페이스는 ADSI 클라이언트가 DN(고유 이름)을 문자열 값에 연결하는 메서드를 제공합니다.
IADsDomain IADsDomain 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다.

IADsEmail IADsEmail 인터페이스는 ADSI 클라이언트가 Email Address 특성에 액세스하는 메서드를 제공합니다.
IADsExtension IADsExtension 인터페이스는 ADSI 애플리케이션 확장 모델의 기초를 형성합니다.
IADsFaxNumber IADsFaxNumber 인터페이스는 ADSI 클라이언트가 Facsimile 전화 번호 특성에 액세스하는 메서드를 제공합니다.
IADsFileService IADsFileService 인터페이스는 IADsService에서 상속되는 이중 인터페이스입니다.
IADsFileServiceOperations IADsFileServiceOperations 인터페이스는 IADsServiceOperations에서 상속되는 이중 인터페이스입니다.
IADsFileShare IADsFileShare 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다. 네트워크를 통해 게시된 파일 공유를 나타내기 위해 설계되었습니다. IADsFileShare의 메서드를 호출하여 파일 공유 지점에 대한 데이터에 액세스하거나 게시합니다.
IADsGroup 디렉터리 서비스에서 그룹 멤버 자격 데이터를 관리합니다.
IADsHold IADsHold 인터페이스는 ADSI 클라이언트가 Hold 특성에 액세스하는 메서드를 제공합니다.
IADsLargeInteger LargeInteger 형식의 64비트 정수 조작에 사용됩니다.
IADsLocality IADsLocality 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다.
IADsMembers IADsMembers 인터페이스는 이중 인터페이스입니다.

IADsNamespaces IADsNamespaces 인터페이스는 ADs 공급자에 의해 구현되며 네임스페이스 개체를 관리하는 데 사용됩니다.
IADsNameTranslate IADsNameTranslateinterface는 ADS_NAME_TYPE_ENUM 열거형에 정의된 다양한 형식 간에 DN(고유 이름)을 변환합니다. 이 기능은 Active Directory의 개체에서 사용할 수 있습니다.
IADsNetAddress IADsNetAddress 인터페이스는 ADSI 클라이언트가 Net Address 특성에 액세스하는 메서드를 제공합니다.
IADSO IADSO 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다.
IADsObjectOptions ADSI 개체를 조작하기 위한 공급자별 옵션을 지정하고 가져오는 직접 메커니즘을 제공합니다.
IADsOctetList IADsOctetList 인터페이스는 ADSI 클라이언트가 Octet List 특성에 액세스하는 메서드를 제공합니다.
IADsOpenDSObject IADsOpenDSObject 인터페이스는 기본 디렉터리 저장소의 개체에 바인딩하기 위한 보안 컨텍스트를 제공하도록 설계되었습니다.
IADsOU organizationalUnit 개체를 관리하는 데 사용됩니다.
IADsPath IADsPath 인터페이스는 ADSI 클라이언트가 Path 특성에 액세스하는 메서드를 제공합니다.
IADsPathname ADSI에서 X.500 및 Windows 경로를 구문 분석합니다.
IADsPostalAddress IADsPostalAddress 인터페이스는 ADSI 클라이언트가 우편 주소 특성에 액세스하는 메서드를 제공합니다.

IADsPrintJob IADsPrintJob 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다.
IADsPrintJobOperations IADsPrintJobOperations 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다.
IADsPrintQueue IADsPrintQueue 인터페이스는 네트워크의 프린터를 나타냅니다.
IADsPrintQueueOperations 네트워크를 통해 프린터를 제어하는 데 사용됩니다.
IADsProperty IADsProperty 인터페이스는 스키마 클래스 개체에 대한 단일 특성 정의를 관리하도록 설계되었습니다.
IADsPropertyEntry IADsPropertyEntry 인터페이스는 속성 캐시에서 속성 항목을 관리하는 데 사용됩니다.
IADsPropertyList IADsPropertyList 인터페이스는 개체의 속성 캐시에 있는 속성 항목 목록을 수정, 읽기 및 업데이트하는 데 사용됩니다.
IADsPropertyValue 미리 정의된 데이터 형식에서 IADsPropertyEntry 개체의 값을 나타내는 데 사용됩니다.
IADsPropertyValue2 모든 데이터 형식으로 IADsPropertyEntry 개체의 값을 나타내는 데 사용됩니다.
IADsReplicaPointer IADsReplicaPointer 인터페이스는 ADSI 클라이언트가 복제본 포인터 특성에 액세스하는 메서드를 제공합니다.
IADsResource IADsResource 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다. 네트워크를 통해 파일 서비스에 대한 열린 리소스를 관리하도록 설계되었습니다.

IADsSecurityDescriptor ADSI 보안 설명자 개체의 속성에 대한 액세스를 제공합니다.
IADsSecurityUtility IADsSecurityUtility 인터페이스는 파일, 파일 공유 또는 레지스트리 키에서 보안 설명자를 가져오기, 설정 또는 검색하는 데 사용됩니다.
IADsService IADsService 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다.
IADsServiceOperations IADsServiceOperations 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다.
IADsSession IADsSession 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다. 네트워크를 통해 파일 서비스에 대한 활성 세션을 나타내도록 설계되었습니다.
IADsSyntax IADsSyntax 인터페이스는 해당 데이터를 나타내는 데 사용되는 사용 가능한 Automation 데이터 형식을 식별하고 수정하는 메서드를 지정합니다.
IADsTimestamp IADsTimestamp 인터페이스는 ADSI 클라이언트가 타임스탬프 특성에 액세스하는 메서드를 제공합니다.
IADsTypedName IADsTypedName 인터페이스는 ADSI 클라이언트가 Typed Name 특성에 액세스하는 메서드를 제공합니다.
IADsUser IADsUser 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다.
IADsWinNTSystemInfo IADsWinNTSystemInfo 인터페이스는 컴퓨터에 대한 WinNT 시스템 정보를 검색합니다. 이러한 시스템 정보에는 사용자 계정 이름, 사용자 도메인, 호스트 이름 및 호스트 컴퓨터의 기본 도메인 컨트롤러가 포함됩니다.

IDirectoryObject IDirectoryObject 인터페이스는 디렉터리 서비스 개체에 대한 직접 액세스를 클라이언트에 제공하는 비 자동화 COM 인터페이스입니다.
IDirectorySchemaMgmt 현재 구현되지 않으며 사용해서는 안 됩니다.
IDirectorySearch IDirectorySearch 인터페이스는 자동화가 아닌 클라이언트가 기본 디렉터리에서 쿼리를 수행하는 데 사용할 수 있는 낮은 오버헤드 메서드를 제공하는 순수 COM 인터페이스입니다.

구조체

ADS_ATTR_DEF ADS_ATTR_DEF 구조체는 사용되지 않는 인터페이스인 IDirectorySchemaMgmt의 일부로만 사용됩니다.
ADS_ATTR_INFO IDirectoryObject::CreateDSObject, IDirectoryObject::GetObjectAttributes 또는 IDirectoryObject::SetObjectAttributes 메서드와 함께 사용할 하나 이상의 특성 값을 포함하는 데 사용됩니다.
ADS_BACKLINK ADS_BACKLINK 구조체는 Back Link 특성 구문의 ADSI 표현입니다.
ADS_CASEIGNORE_LIST ADS_CASEIGNORE_LIST 구조체는 사례 무시 목록 특성 구문의 ADSI 표현입니다.
ADS_CLASS_DEF ADS_CLASS_DEF 구조체는 사용되지 않는 인터페이스인 IDirectorySchemaMgmt의 일부로만 사용됩니다. 다음 정보는 레거시 목적으로만 제공됩니다. ADS_CLASS_DEF 구조체는 개체 클래스의 정의를 보유합니다.
ADS_DN_WITH_BINARY ADSVALUE 구조체와 함께 이진 데이터를 포함하는 고유 이름 특성 값을 포함하는 데 사용됩니다.

ADS_DN_WITH_STRING 문자열 데이터도 포함하는 고유 이름 특성 값을 포함하기 위해 ADSVALUE 구조와 함께 사용됩니다.
ADS_EMAIL ADS_EMAIL 구조체는 EMail Address 특성 구문의 ADSI 표현입니다.
ADS_FAXNUMBER ADS_FAXNUMBER 구조는 Facsimile 전화 번호 특성 구문의 ADSI 표현입니다.
ADS_HOLD ADS_HOLD 구조체는 Hold 특성 구문의 ADSI 표현입니다.
ADS_NETADDRESS ADS_NETADDRESS 구조체는 Net Address 특성 구문의 ADSI 표현입니다.
ADS_NT_SECURITY_DESCRIPTOR ADS_NT_SECURITY_DESCRIPTOR 구조는 Windows용 보안 설명자의 데이터 형식을 정의합니다.
ADS_OBJECT_INFO ADS_OBJECT_INFO 구조체는 디렉터리 서비스 개체의 ID 및 위치를 포함한 데이터를 지정합니다.
ADS_OCTET_LIST ADS_OCTET_LIST 구조체는 정렬된 단일 바이트 문자열 시퀀스의 ADSI 표현입니다.
ADS_OCTET_STRING ADS_OCTET_STRING 구조는 Active Directory에서 사용되는 Octet String 특성 구문의 ADSI 표현입니다.
ADS_PATH ADS_PATH 구조체는 Path 특성 구문의 ADSI 표현입니다.
ADS_POSTALADDRESS ADS_POSTALADDRESS 구조는 우편 주소 특성의 ADSI 표현입니다.
ADS_PROV_SPECIFIC ADS_PROV_SPECIFIC 구조에는 BLOB(이진 대형 개체)로 표현되는 공급자별 데이터가 포함됩니다.

ADS_REPLICAPOINTER 복제본 포인터 특성 구문의 ADSI 표현을 나타냅니다.
ADS_SEARCH_COLUMN ADS_SEARCH_COLUMN 구조는 디렉터리 서비스 데이터베이스에서 반환된 쿼리에서 검색 열의 내용을 지정합니다.
ADS_SEARCHPREF_INFO ADS_SEARCHPREF_INFO 구조는 쿼리 기본 설정을 지정합니다.
ADS_SORTKEY ADS_SORTKEY 구조는 쿼리를 정렬하는 방법을 지정합니다.
ADS_TIMESTAMP ADS_TIMESTAMP 구조는 타임스탬프 특성 구문의 ADSI 표현입니다.
ADS_TYPEDNAME Typed Name 특성 구문의 ADSI 표현을 나타냅니다.
ADS_VLV VLV(가상 목록 보기) 검색을 수행하는 데 사용되는 메타데이터를 포함합니다.
ADSVALUE ADSI 데이터 형식으로 지정된 값을 포함합니다.

활동 코디네이터

아티클2024. 06. 12.

활동 코디네이터 API는 시스템에서 지연 가능한 작업의 실행을 조정합니다.

활동 코디네이터용으로 개발하려면 다음 헤더가 필요합니다.

- activitycoordinator.h
- activitycoordinatortypes.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 활동 코디네이터 개요
- 활동 코디네이터 API 및 용어
- 올바른 활동 코디네이터 정책 선택
- 활동 코디네이터 예제 프로젝트
- 활동 코디네이터 리소스 지원 확인

일반적인 Win32 프로그래밍 지침은 다음을 참조하세요.

- 오류 처리
- 디버그 도움말 라이브러리
- 동적 링크 라이브러리
- 이벤트 로깅
- 프로세스 및 스레드

열거형

테이블 확장

ACTIVITY_COORDINATOR_CONDITION 지원되는 조건 집합의 열거형입니다.
ACTIVITY_COORDINATOR_NOTIFICATION 구독 콜백에 대해 가능한 알림 유형의 열거형입니다.
ACTIVITY_COORDINATOR_POLICY_TEMPLATE 지원되는 템플릿 정책 집합의 열거형입니다.

ACTIVITY_COORDINATOR_RESOURCE 지원되는 리소스 집합의 열거형입니다.

Functions

테이블 확장

ACTIVITY_COORDINATOR_CALLBACK 활동에 대한 모든 알림에 대해 호출할 콜백입니다.
CreateActivityCoordinatorPolicy 새 정책을 할당하고 제공된 정책 템플릿 열거형에 따라 필드를 미리 채웁니다.
DestroyActivityCoordinatorPolicy CreateActivityCoordinatorPolicy를 통해 얻은 정책을 삭제합니다.
GetActivityCoordinatorPolicyResourceCondition 정책에 대한 리소스의 원하는 조건을 가져옵니다.
IsActivityCoordinatorResourceSupported 이 함수를 사용하면 앱이 런타임에 지원되는 리소스를 확인할 수 있습니다.
SetActivityCoordinatorPolicyResourceCondition 정책에 대해 리소스의 원하는 조건을 설정합니다.
SubscribeActivityCoordinatorPolicy 정책의 열기 및 닫기 알림을 받는 구독을 만듭니다.
UnsubscribeActivityCoordinatorPolicy 구독을 삭제합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

AllJoyn API

아티클2023. 08. 22.

AllJoyn API 기술 개요입니다.

AllJoyn API를 개발하려면 다음 헤더가 필요합니다.

- msajtransport.h
- windows.devices.alljoyn.interop.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- AllJoyn API

Functions

AllJoynCloseBusHandle 명명된 파이프 핸들을 닫습니다.
AllJoynConnectToBus AllJoyn Router Node Service 명명된 파이프를 열고 PIPE_NOWAIT 설정합니다.
AllJoynEnumEvents TCP 소켓 WSAEnumNetworkEvents 기능과 유사한 AllJoyn 전송 기능을 제공합니다.
AllJoynEventSelect TCP 소켓 WSAEventSelect 기능과 유사한 AllJoyn 전송 기능을 제공합니다.
AllJoynReceiveFromBus 명명된 파이프를 통해 버스에서 데이터를 받습니다.
AllJoynSendToBus 명명된 파이프를 통해 버스로 데이터를 보냅니다. 이 API의 호출자는 bytesTransferred가 요청된 바이트보다 작은지 검사 이 API를 다시 호출하여 나머지 데이터를 다시 보냅니다.
CreateFromWin32Handle 소유권을 가져오지 않고 기존 alljoyn_busattachment instance 대해 AllJoynBusAttachment를 생성합니다.

get_Win32Handle 기본 alljoyn_busattachment instance 대한 포인터를 반환하기만 하면 앱이 AllJoyn Core C API에서 직접 사용할 수 있는 클래식 COM 인터페이스입니다.

인터페이스

IWindowsDevicesAllJoynBusAttachmentFactoryInterop 이 인터페이스를 사용하면 참조의 소유권을 가져오지 않고 alljoyn_busattachment 만들 수 있습니다.
IWindowsDevicesAllJoynBusAttachmentInterop 이 인터페이스를 사용하면 기본 alljoyn_busattachment 개체를 검색할 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

맬웨어 방지 검사 인터페이스

아티클2023. 08. 22.

맬웨어 방지 스캔 인터페이스 기술 개요입니다.

맬웨어 방지 검사 인터페이스를 사용하여 개발하려면 이 헤더를 포함해야 합니다.

- amsi.h

이 기술에 대한 프로그래밍 지침은

- AMSI(맬웨어 방지 프로그램 검사 인터페이스)

열거형

AMSI_ATTRIBUTE IAmsiStream::GetAttribute에서 요청할 수 있는 특성 유형을 지정합니다.
AMSI_RESULT 검사에서 반환되는 결과의 유형을 지정합니다.

Functions

AmsiCloseSession AmsiOpenSession에서 연 세션을 닫습니다.
AmsiInitialize AMSI API를 초기화합니다.
AmsiNotifyOperation 맬웨어 방지 공급자에게 임의 작업에 대한 알림을 보냅니다. (AmsiNotifyOperation)
AmsiOpenSession 여러 검사 요청의 상관 관계를 지정할 수 있는 세션을 엽니다.

AmsiResultIsMalware 검사 결과가 콘텐츠를 차단해야 했음을 나타내는지 여부를 확인합니다.
AmsiScanBuffer 버퍼가 가득한 콘텐츠에서 맬웨어를 검사합니다.
AmsiScanString 문자열에서 맬웨어를 검색합니다.
AmsiUninitialize AmsiInitialize에서 원래 연 AMSI API의 instance 제거합니다.
CloseSession 세션을 닫습니다. (IAntimalware.CloseSession)
CloseSession 세션을 닫습니다. (IAntimalwareProvider.CloseSession)
표시 이름 표시할 맬웨어 방지 공급자의 이름입니다.
GetAttribute 스트림에서 요청된 특성을 반환합니다.
알림 임의 작업에 대한 알림을 맬웨어 방지 제품으로 보냅니다.
알림 맬웨어 방지 공급자에게 임의 작업에 대한 알림을 보냅니다. (IAntimalwareProvider2::Notify)
읽기 읽을 콘텐츠가 가득 찬 버퍼를 요청합니다.
검사 콘텐츠 스트림을 검사합니다. (IAntimalware.Scan)
검사

콘텐츠 스트림을 검사합니다. (IAntimalwareProvider.Scan)

인터페이스

IAmsiStream 검사할 스트림을 나타냅니다.
IAntimalware 맬웨어 방지 제품을 나타냅니다. (IAntimalware)
IAntimalware2 맬웨어 방지 제품을 나타냅니다. (IAntimalware2)
IAntimalwareProvider 맬웨어 방지 제품의 공급자를 나타냅니다. (IAntimalwareProvider)
IAntimalwareProvider2 맬웨어 방지 제품의 공급자를 나타냅니다. (IAntimalwareProvider2)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

애플리케이션 설치 및 서비스

아티클2023. 03. 16.

애플리케이션 설치 및 서비스 기술 개요입니다.

애플리케이션 설치 및 서비스를 개발하려면 다음 헤더가 필요합니다.

- evalcom2.h
- mergemod.h
- msi.h
- msiquery.h
- setupapi.h
- sfc.h
- winsxs.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 설치 API입니다.
- Windows Installer
- 격리된 애플리케이션 및 side-by-side 어셈블리
- Windows 리소스 보호

열거형

ACTCTX_COMPATIBILITY_ELEMENT_TYPE ACTCTX_COMPATIBILITY_ELEMENT_TYPE 열거형은 애플리케이션 매니페스트의 호환성 요소를 설명합니다.
ACTCTX_REQUESTED_RUN_LEVEL ACTCTX_REQUESTED_RUN_LEVEL 열거형은 활성화 컨텍스트의 요청된 실행 수준을 설명합니다.
ASM_CMP_FLAGS ASM_CMP_FLAGS 열거형의 값은 IsEqual 메서드에서 비교할 두 어셈블리 이름의 부분을 지정하는 데 사용됩니다.
ASM_DISPLAY_FLAGS ASM_DISPLAY_FLAGS 열거형의 값은 GetDisplayName 메서드에서 어셈블리 이름의 문자열 표현에 포함할 어셈블리 전체 이름의 일부를 지정하는 데 사용됩니다.

ASM_NAME ASM_NAME 열거형의 값은 side-by-side 어셈블리 이름에 포함된 이름-값 쌍의 속성 ID입니다.
CREATE_ASM_NAME_OBJ_FLAGS CREATE_ASM_NAME_OBJ_FLAGS 열거형은 CreateAssemblyNameObject 함수에서 사용됩니다.

함수

ActivateActCtx ActivateActCtx 함수는 지정된 활성화 컨텍스트를 활성화합니다.
ActivateActCtx ActivateActCtx 함수는 지정된 활성화 컨텍스트를 활성화합니다.
AddRefActCtx AddRefActCtx 함수는 지정된 활성화 컨텍스트의 참조 수를 증분합니다.
AddRefActCtx AddRefActCtx 함수는 지정된 활성화 컨텍스트의 참조 수를 증분합니다.
복제 Clone 메서드는 현재 병렬 어셈블리 이름을 IAssemblyName의 새 인스턴스에 복사합니다.
CloseCUB CloseCUB 메서드는 열려 있는 ICE(내부 일관성 계산기) .cub 파일을 닫습니다. OPENCUB 메서드를 사용하여 ICE(내부 일관성 평가기) .cub 파일을 열 수 있습니다.
CloseDatabase CloseDatabase 메서드는 현재 열려 있는 Windows Installer 패키지 또는 병합 모듈을 닫습니다. OpenDatabase 메서드를 사용하여 Windows Installer 패키지 또는 병합 모듈을 열 수 있습니다.
CloseDatabase CloseDatabase 메서드는 현재 열려 있는 Windows Installer 데이터베이스를 닫습니다. 자세한 내용은 Merge 개체의 CloseDatabase 메서드를 참조하세요.

CloseLog CloseLog 함수 메서드는 현재 로그를 닫습니다. 자세한 내용은 Merge 개체의 CloseLog 메서드를 참조하세요.
CloseModule CloseModule 메서드는 현재 열려 있는 Windows Installer 병합 모듈을 닫습니다. 자세한 내용은 Merge 개체의 CloseModule 메서드를 참조하세요.
커밋 Commit 메서드는 정보를 병렬 저장소에 복사합니다. 이 메서드가 반환되면 어셈블리가 병렬 저장소에 표시됩니다.
연결 Connect 메서드는 데이터베이스에 병합되었거나 병합된 모듈을 추가 기능에 연결합니다. 자세한 내용은 Merge 개체의 Connect 메서드를 참조하세요.
CreateActCtxA CreateActCtx 함수는 활성화 컨텍스트를 만듭니다. (ANSI)
CreateActCtxA CreateActCtx 함수는 활성화 컨텍스트를 만듭니다. (ANSI)
CreateActCtxW CreateActCtx 함수는 활성화 컨텍스트를 만듭니다. (유니코드)
CreateActCtxW CreateActCtx 함수는 활성화 컨텍스트를 만듭니다. (유니코드)
CreateAssemblyCache CreateAssemblyCache 함수는 IAssemblyCache 인터페이스의 인스턴스를 가져옵니다.
CreateAssemblyCacheItem CreateAssemblyCacheItem 메서드는 설치 중인 side-by-side 어셈블리에 해당하는 항목을 어셈블리 캐시에 만듭니다.
CreateAssemblyNameObject CreateAssemblyNameObject 함수는 IAssemblyName 인터페이스의 인스턴스를 가져옵니다.

CreateSourceImage CreateSourceImage 메서드를 사용하면 모듈 구성 중에 변경된 모듈을 고려하여 병합 후 모듈에서 디스크의 원본 이미지로 파일을 추출할 수 있습니다.
CreateStream CreateStream 메서드는 매니페스트 또는 모듈의 원본을 스트림에 복사합니다.
DeactivateActCtx DeactivateActCtx 함수는 지정된 쿠키에 해당하는 활성화 컨텍스트를 비활성화합니다.
DeactivateActCtx DeactivateActCtx 함수는 지정된 쿠키에 해당하는 활성화 컨텍스트를 비활성화합니다.
ExtractCAB ExtractCAB 메서드는 모듈에서 포함된 .cab 파일을 추출하고 지정된 파일로 저장합니다.
ExtractFiles ExtractFiles 메서드는 모듈에서 포함된 .cab 파일을 추출한 다음 해당 파일을 대상 디렉터리에 씁니다. 자세한 내용은 Merge 개체의 ExtractFiles 메서드를 참조하세요.
ExtractFilesEx ExtractFilesEx 메서드는 모듈에서 포함된 .cab 파일을 추출한 다음 해당 파일을 대상 디렉터리에 씁니다. 자세한 내용은 Merge 개체의 ExtractFilesEx 메서드를 참조하세요.
Finalize Finalize 메서드는 병렬 어셈블리 이름이 변경되지 않도록 합니다. Finalize가 호출되면 SetProperty에 대한 추가 호출이 E_UNEXPECTED 반환됩니다.
FindActCtxSectionGuid FindActCtxSectionGuid 함수는 현재 활성화 컨텍스트의 특정 GUID에 대한 정보를 검색하고 ACTCTX_SECTION_KEYED_DATA 구조를 반환합니다.
FindActCtxSectionGuid FindActCtxSectionGuid 함수는 현재 활성화 컨텍스트의 특정 GUID에 대한 정보를 검색하고 ACTCTX_SECTION_KEYED_DATA 구조를 반환합니다.

FindActCtxSectionStringA FindActCtxSectionString 함수는 현재 활성화 컨텍스트의 특정 문자열에 대한 정보를 검색하고 ACTCTX_SECTION_KEYED_DATA 구조를 반환합니다. (ANSI)
FindActCtxSectionStringA FindActCtxSectionString 함수는 현재 활성화 컨텍스트의 특정 문자열에 대한 정보를 검색하고 ACTCTX_SECTION_KEYED_DATA 구조를 반환합니다. (ANSI)
FindActCtxSectionStringW FindActCtxSectionString 함수는 현재 활성화 컨텍스트의 특정 문자열에 대한 정보를 검색하고 ACTCTX_SECTION_KEYED_DATA 구조를 반환합니다. (유니코드)
FindActCtxSectionStringW FindActCtxSectionString 함수는 현재 활성화 컨텍스트의 특정 문자열에 대한 정보를 검색하고 ACTCTX_SECTION_KEYED_DATA 구조를 반환합니다. (유니코드)
get_Attributes get_Attributes 메서드는 ConfigurableItem 개체의 Attributes 속성을 검색합니다.
get_ConfigurableItems get_ConfigurableItems 메서드는 Merge 개체의 ConfigurableItems 속성을 검색합니다.
get_Context get_Context 메서드는 ConfigurableItem 개체의 Context 속성을 검색합니다.
get_DatabaseKeys get_DatabaseKeys 메서드는 Error 개체의 DatabaseKeys 속성을 검색합니다. 이 메서드는 데이터베이스에서 행의 기본 키를 포함하는 문자열 컬렉션에 대한 포인터를 반환하여 오류가 발생하며 컬렉션의 항목당 하나의 키를 반환합니다.
get_DatabaseTable get_DatabaseTable 메서드는 Error 개체의 DatabaseTable 속성을 검색합니다. 이 메서드는 오류를 발생시킨 데이터베이스의 테이블 이름을 반환합니다.
get_DefaultValue get_DefaultValue 메서드는 ConfigurableItem 개체의 DefaultValue 속성을 검색합니다.

get_Dependencies get_Dependencies 메서드는 Merge 개체의 Dependencies 속성을 검색합니다.
get_Description get_Description 메서드는 ConfigurableItem 개체의 Description 속성을 검색합니다.
get_DisplayName get_DisplayName 메서드는 ConfigurableItem 개체의 DisplayName 속성을 검색합니다.
get_Errors get_Errors 메서드는 Merge 개체의 Errors 속성을 검색합니다. 그러면 현재 오류 컬렉션이 검색됩니다.
get_Format get_Format 메서드는 ConfigurableItem 개체의 Format 속성을 검색합니다.
get_HelpKeyword get_HelpKeyword 메서드는 ConfigurableItem 개체의 HelpKeyword 속성을 검색합니다.
get_HelpLocation get_HelpLocation 메서드는 ConfigurableItem 개체의 HelpLocation 속성을 검색합니다.
get_Language get_Language 메서드는 Dependency 개체의 Language 속성을 검색합니다. 이 메서드는 필요한 모듈의 LANGID를 반환합니다.
get_Language get_Language 메서드는 Error 개체의 Language 속성을 검색합니다. 이 함수는 오류의 LANGID를 반환합니다.
get_Module get_Module 메서드는 Dependency 개체의 Module 속성을 검색합니다. 이 메서드는 현재 문자열에 필요한 모듈의 ModuleID를 BSTR 형식으로 반환합니다. ModuleID는 ModuleSignature 테이블에 사용된 것과 동일한 형식입니다.
get_ModuleFiles get_ModuleFiles 메서드는 GetFiles 개체의 ModuleFiles 속성을 검색합니다.

get_ModuleKeys get_ModuleKeys 메서드는 Error 개체의 ModuleKeys 속성을 검색합니다. 이 메서드는 모듈에서 행의 기본 키가 들어 있는 문자열 컬렉션에 대한 포인터를 반환하여 오류를 발생시켜 컬렉션의 항목당 하나의 키를 반환합니다.
get_ModuleTable get_ModuleTable 메서드는 Error 개체의 ModuleTable 속성을 검색합니다. 이 메서드는 오류를 발생시킨 모듈의 테이블 이름을 반환합니다.
get_Name get_Name 메서드는 ConfigurableItem 개체의 Name 속성을 검색합니다.
get_Path get_Path 메서드는 Error 개체의 Path 속성을 검색합니다.
get_Type get_Type 메서드는 ConfigurableItem 개체의 Type 속성을 검색합니다.
get_Type get_Type 메서드는 Error 개체의 Type 속성을 검색합니다. 이 메서드는 이 개체가 나타내는 오류 유형을 나타내는 msmErrorType 값을 반환합니다.
get_Version get_Version 메서드는 Dependency 개체의 Version 속성을 검색합니다. 이 메서드는 필요한 모듈의 버전을 BSTR 형식으로 반환합니다.
GetCurrentActCtx GetCurrentActCtx 함수는 호출 스레드의 활성 활성화 컨텍스트에 핸들을 반환합니다.
GetCurrentActCtx GetCurrentActCtx 함수는 호출 스레드의 활성 활성화 컨텍스트에 핸들을 반환합니다.
GetDisplayName GetDisplayName 메서드는 병렬 어셈블리 이름의 문자열 표현을 가져옵니다.
GetName GetName 메서드는 어셈블리 이름의 이름 부분을 반환합니다.

Getproperty GetProperty 메서드는 어셈블리 이름에서 이름-값 쌍의 값을 가져옵니다.
InstallAssembly InstallAssembly 메서드는 어셈블리에 애플리케이션 참조를 병렬 저장소에 추가하고 어셈블리의 파일을 나란히 저장소에 복사합니다. 설치되는 어셈블리의 파일은 현재 파일 시스템에 있어야 합니다.
InstallHinfSectionA InstallHinfSection은 .inf 파일의 섹션을 실행하는 데 사용할 수 있는 Setupapi.dll 내보낸 진입점 함수입니다. InstallHinfSection은 주의 섹션에 설명된 대로 Rundll32.exe 유틸리티를 호출하여 호출할 수 있습니다. (ANSI)
InstallHinfSectionW InstallHinfSection은 .inf 파일의 섹션을 실행하는 데 사용할 수 있는 Setupapi.dll 내보낸 진입점 함수입니다. InstallHinfSection은 주의 섹션에 설명된 대로 Rundll32.exe 유틸리티를 호출하여 호출할 수 있습니다. (유니코드)
INSTALLUI_HANDLER_RECORD INSTALLUI_HANDLER_RECORD 함수 프로토타입은 설치 관리자가 진행률 알림 및 오류 메시지를 호출하는 콜백 함수를 정의합니다. MsiSetExternalUIRecord 함수를 호출하여 레코드 기반 UI(외부 사용자 인터페이스) 처리기를 사용하도록 설정합니다.
INSTALLUI_HANDLERA INSTALLUI_HANDLER 함수 프로토타입은 설치 관리자가 진행률 알림 및 오류 메시지를 호출하는 콜백 함수를 정의합니다. (ANSI)
INSTALLUI_HANDLERW INSTALLUI_HANDLER 함수 프로토타입은 설치 관리자가 진행률 알림 및 오류 메시지를 호출하는 콜백 함수를 정의합니다. (유니코드)
IsEqual IsEqual 메서드는 현재 어셈블리 이름을 다른 어셈블리 이름과 비교합니다.
Log Log 메서드는 현재 열려 있는 로그 파일에 텍스트 문자열을 씁니다. 자세한 내용은 Merge 개체의 Log 메서드를 참조하세요.

LPDISPLAYVAL LPDISPLAYVAL 함수 사양은 콜백 함수 프로토타입을 정의합니다. IValidate::SetDisplay 메서드를 사용하면 작성 도구가 등록된 콜백 함수를 통해 ICE 상태 메시지를 받을 수 있습니다.
LPEVALCOMCALLBACK LPEVALCOMCALLBACK 사양은 콜백 함수 프로토타입을 정의합니다. IValidate::SetStatus 메서드를 사용하면 작성 도구가 등록된 콜백 함수를 통해 유효성 검사 진행률에 대한 정보를 받을 수 있습니다.
병합 Merge 메서드는 현재 데이터베이스 및 현재 모듈의 병합을 실행합니다.
MergeEx MergeEx 메서드는 현재 데이터베이스와 현재 모듈의 병합을 실행합니다.
MsiAdvertiseProductA MsiAdvertiseProduct 함수는 보급 스크립트를 생성하거나 컴퓨터에 제품을 보급합니다. (ANSI)
MsiAdvertiseProductExA MsiAdvertiseProductEx 함수는 보급 스크립트를 생성하거나 컴퓨터에 제품을 보급합니다. (ANSI)
MsiAdvertiseProductExW MsiAdvertiseProductEx 함수는 보급 스크립트를 생성하거나 컴퓨터에 제품을 보급합니다. (유니코드)
MsiAdvertiseProductW MsiAdvertiseProduct 함수는 보급 스크립트를 생성하거나 컴퓨터에 제품을 보급합니다. (유니코드)
MsiAdvertiseScriptA MsiAdvertiseScript 함수는 보급된 스크립트 파일을 지정된 위치에 복사합니다. (ANSI)
MsiAdvertiseScriptW MsiAdvertiseScript 함수는 보급된 스크립트 파일을 지정된 위치에 복사합니다. (유니코드)
MsiApplyMultiplePatchesA 패치를 받을 수 있는 제품에 하나 이상의 패치를 적용합니다. (ANSI)

MsiApplyMultiplePatchesW 패치를 받을 수 있는 제품에 하나 이상의 패치를 적용합니다. (유니코드)
MsiApplyPatchA 패치 패키지를 받을 수 있는 자격으로 패치 패키지에 나열된 각 제품에 대해 MsiApplyPatch 함수는 설치를 호출하고 PATCH 속성을 패치 패키지의 경로로 설정합니다. (ANSI)
MsiApplyPatchW 패치 패키지를 받을 수 있는 자격으로 패치 패키지에 나열된 각 제품에 대해 MsiApplyPatch 함수는 설치를 호출하고 PATCH 속성을 패치 패키지의 경로로 설정합니다. (유니코드)
MsiBeginTransactionA MsiBeginTransaction 함수는 다중 패키지 설치의 트랜잭션 처리를 시작하고 트랜잭션에 대한 식별자를 반환합니다. (ANSI)
MsiBeginTransactionW MsiBeginTransaction 함수는 다중 패키지 설치의 트랜잭션 처리를 시작하고 트랜잭션에 대한 식별자를 반환합니다. (유니코드)
MsiCloseAllHandles MsiCloseAllHandles 함수는 현재 스레드에서 할당한 열려 있는 모든 설치 핸들을 닫습니다. 이 함수는 진단 함수이며 정리에 사용하면 안 됩니다.
MsiCloseHandle MsiCloseHandle 함수는 열린 설치 핸들을 닫습니다.
MsiCollectUserInfoA MsiCollectUserInfo 함수는 설치 마법사에서 사용자 정보 및 제품 ID를 가져오고 저장합니다. (ANSI)
MsiCollectUserInfoW MsiCollectUserInfo 함수는 설치 마법사에서 사용자 정보 및 제품 ID를 가져오고 저장합니다. (유니코드)
MsiConfigureFeatureA MsiConfigureFeature 함수는 제품 기능에 대해 설치된 상태를 구성합니다. (ANSI)
MsiConfigureFeatureW MsiConfigureFeature 함수는 제품 기능에 대해 설치된 상태를 구성합니다. (유니코드)

MsiConfigureProductA MsiConfigureProduct 함수는 제품을 설치하거나 제거합니다. (ANSI)
MsiConfigureProductExA 제품을 설치하거나 제거합니다. (MsiConfigureProductExA)
MsiConfigureProductExW 제품을 설치하거나 제거합니다. (MsiConfigureProductExW)
MsiConfigureProductW MsiConfigureProduct 함수는 제품을 설치하거나 제거합니다. (유니코드)
MsiCreateRecord MsiCreateRecord 함수는 지정된 수의 필드를 사용하여 새 레코드 개체를 만듭니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다.
MsiCreateTransformSummaryInfoA MsiCreateTransformSummaryInfo 함수는 유효성 검사 및 오류 조건을 포함하도록 기존 변환의 요약 정보를 만듭니다. 이 함수를 실행하면 MsiGetLastErrorRecord를 사용하여 액세스할 수 있는 오류 레코드가 설정됩니다. (ANSI)
MsiCreateTransformSummaryInfoW MsiCreateTransformSummaryInfo 함수는 유효성 검사 및 오류 조건을 포함하도록 기존 변환의 요약 정보를 만듭니다. 이 함수를 실행하면 MsiGetLastErrorRecord를 사용하여 액세스할 수 있는 오류 레코드가 설정됩니다. (유니코드)
MsiDatabaseApplyTransformA MsiDatabaseApplyTransform 함수는 데이터베이스에 변환을 적용합니다. (ANSI)
MsiDatabaseApplyTransformW MsiDatabaseApplyTransform 함수는 데이터베이스에 변환을 적용합니다. (유니코드)
MsiDatabaseCommit MsiDatabaseCommit 함수는 변경 내용을 데이터베이스에 커밋합니다.
MsiDatabaseExportA MsiDatabaseExport 함수는 열려 있는 데이터베이스에서 텍스트 보관 파일로 Microsoft Installer 테이블을 내보냅니다. (ANSI)

MsiDatabaseExportW MsiDatabaseExport 함수는 열려 있는 데이터베이스에서 텍스트 보관 파일로 Microsoft Installer 테이블을 내보냅니다. (유니코드)
MsiDatabaseGenerateTransformA MsiDatabaseGenerateTransform 함수는 두 데이터베이스 간의 차이점이 있는 변환 파일을 생성합니다. (ANSI)
MsiDatabaseGenerateTransformW MsiDatabaseGenerateTransform 함수는 두 데이터베이스 간의 차이점이 있는 변환 파일을 생성합니다. (유니코드)
MsiDatabaseGetPrimaryKeysA MsiDatabaseGetPrimaryKeys 함수는 지정된 테이블에 대한 모든 기본 키 열의 이름을 포함하는 레코드를 반환합니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다. (ANSI)
MsiDatabaseGetPrimaryKeysW MsiDatabaseGetPrimaryKeys 함수는 지정된 테이블에 대한 모든 기본 키 열의 이름을 포함하는 레코드를 반환합니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다. (유니코드)
MsiDatabaseImportA MsiDatabaseImport 함수는 설치 관리자 텍스트 보관 파일을 열린 데이터베이스 테이블로 가져옵니다. (ANSI)
MsiDatabaseImportW MsiDatabaseImport 함수는 설치 관리자 텍스트 보관 파일을 열린 데이터베이스 테이블로 가져옵니다. (유니코드)
MsiDatabaseIsTablePersistentA MsiDatabaseIsTablePersistent 함수는 특정 테이블의 상태를 설명하는 열거형을 반환합니다. (ANSI)
MsiDatabaseIsTablePersistentW MsiDatabaseIsTablePersistent 함수는 특정 테이블의 상태를 설명하는 열거형을 반환합니다. (유니코드)
MsiDatabaseMergeA MsiDatabaseMerge 함수는 두 데이터베이스를 함께 병합하여 중복 행을 허용합니다. (ANSI)

MsiDatabaseMergeW MsiDatabaseMerge 함수는 두 데이터베이스를 함께 병합하여 중복 행을 허용합니다. (유니코드)
MsiDatabaseOpenViewA MsiDatabaseOpenView 함수는 데이터베이스 쿼리를 준비하고 뷰 개체를 만듭니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다. (ANSI)
MsiDatabaseOpenViewW MsiDatabaseOpenView 함수는 데이터베이스 쿼리를 준비하고 뷰 개체를 만듭니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다. (유니코드)
MsiDetermineApplicablePatchesA MsiDetermineApplicablePatches 함수는 패치 파일, XML 파일 및 XML Blob 집합을 가져와서 지정된 Windows Installer 패키지 및 시퀀스에 적용되는 패치를 결정합니다. (ANSI)
MsiDetermineApplicablePatchesW MsiDetermineApplicablePatches 함수는 패치 파일, XML 파일 및 XML Blob 집합을 가져와서 지정된 Windows Installer 패키지 및 시퀀스에 적용되는 패치를 결정합니다. (유니코드)
MsiDeterminePatchSequenceA 패치 파일, XML 파일 및 XML Blob 집합을 가져와서 지정된 설치된 제품에 대한 패치에 가장 적합한 애플리케이션 시퀀스를 결정합니다. (ANSI)
MsiDeterminePatchSequenceW 패치 파일, XML 파일 및 XML Blob 집합을 가져와서 지정된 설치된 제품에 대한 패치에 가장 적합한 애플리케이션 시퀀스를 결정합니다. (유니코드)
MsiDoActionA MsiDoAction 함수는 기본 제공 작업, 사용자 지정 작업 또는 사용자 인터페이스 마법사 작업을 실행합니다. (ANSI)
MsiDoActionW MsiDoAction 함수는 기본 제공 작업, 사용자 지정 작업 또는 사용자 인터페이스 마법사 작업을 실행합니다. (유니코드)
MsiEnableLogA MsiEnableLog 함수는 호출 프로세스에서 시작된 모든 후속 설치에 대한 로그 모드를 설정합니다. (ANSI)

MsiEnableLogW MsiEnableLog 함수는 호출 프로세스에서 시작된 모든 후속 설치에 대한 로그 모드를 설정합니다. (유니코드)
MsiEnableUIPreview MsiEnableUIPreview 함수를 사용하면 사용자 인터페이스의 미리 보기 모드를 사용하여 사용자 인터페이스 대화 상자를 쉽게 작성할 수 있습니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다.
MsiEndTransaction MsiEndTransaction 함수는 MsiBeginTransaction 함수에서 연 트랜잭션에 속하는 모든 설치를 커밋하거나 롤백할 수 있습니다.
MsiEnumClientsA MsiEnumClients 함수는 지정된 설치된 구성 요소에 대한 클라이언트를 열거합니다. 함수는 호출할 때마다 하나의 제품 코드를 검색합니다. (ANSI)
MsiEnumClientsExA MsiEnumClientsEx 함수는 지정된 구성 요소를 사용하는 설치된 애플리케이션을 열거합니다. 함수는 호출할 때마다 애플리케이션에 대한 제품 코드를 검색합니다. (ANSI)
MsiEnumClientsExW MsiEnumClientsEx 함수는 지정된 구성 요소를 사용하는 설치된 애플리케이션을 열거합니다. 함수는 호출할 때마다 애플리케이션에 대한 제품 코드를 검색합니다. (유니코드)
MsiEnumClientsW MsiEnumClients 함수는 지정된 설치된 구성 요소에 대한 클라이언트를 열거합니다. 함수는 호출할 때마다 하나의 제품 코드를 검색합니다. (유니코드)
MsiEnumComponentCostsA MsiEnumComponentCosts 함수는 구성 요소를 설치하는 데 필요한 드라이브당 디스크 공간을 열거합니다. (ANSI)
MsiEnumComponentCostsW MsiEnumComponentCosts 함수는 구성 요소를 설치하는 데 필요한 드라이브당 디스크 공간을 열거합니다. (유니코드)

MsiEnumComponentQualifiersA MsiEnumComponentQualifiers 함수는 지정된 구성 요소에 대해 보급된 한정자를 열거합니다. 이 함수는 호출할 때마다 한정자를 검색합니다. (ANSI)
MsiEnumComponentQualifiersW MsiEnumComponentQualifiers 함수는 지정된 구성 요소에 대해 보급된 한정자를 열거합니다. 이 함수는 호출할 때마다 한정자를 검색합니다. (유니코드)
MsiEnumComponentsA MsiEnumComponents 함수는 모든 제품에 대해 설치된 구성 요소를 열거합니다. 이 함수는 호출할 때마다 하나의 구성 요소 코드를 검색합니다. (ANSI)
MsiEnumComponentsExA MsiEnumComponentsEx 함수는 설치된 구성 요소를 열거합니다. 함수는 호출할 때마다 한 구성 요소에 대한 구성 요소 코드를 검색합니다. 구성 요소 코드는 구성 요소, 버전 및 언어에 고유한 문자열 GUID입니다. (ANSI)
MsiEnumComponentsExW MsiEnumComponentsEx 함수는 설치된 구성 요소를 열거합니다. 함수는 호출할 때마다 한 구성 요소에 대한 구성 요소 코드를 검색합니다. 구성 요소 코드는 구성 요소, 버전 및 언어에 고유한 문자열 GUID입니다. (유니코드)
MsiEnumComponentsW MsiEnumComponents 함수는 모든 제품에 대해 설치된 구성 요소를 열거합니다. 이 함수는 호출할 때마다 하나의 구성 요소 코드를 검색합니다. (유니코드)
MsiEnumFeaturesA MsiEnumFeatures 함수는 지정된 제품에 대해 게시된 기능을 열거합니다. 이 함수는 호출할 때마다 하나의 기능 ID를 검색합니다. (ANSI)
MsiEnumFeaturesW MsiEnumFeatures 함수는 지정된 제품에 대해 게시된 기능을 열거합니다. 이 함수는 호출할 때마다 하나의 기능 ID를 검색합니다. (유니코드)
MsiEnumPatchesA MsiEnumPatches 함수는 제품에 적용된 모든 패치를 열거합니다. (ANSI)
MsiEnumPatchesExA 특정 컨텍스트 또는 모든 컨텍스트에서 모든 패치를 열거합니다. (ANSI)

MsiEnumPatchesExW 특정 컨텍스트 또는 모든 컨텍스트에서 모든 패치를 열거합니다. (유니코드)
MsiEnumPatchesW MsiEnumPatches 함수는 제품에 적용된 모든 패치를 열거합니다. (유니코드)
MsiEnumProductsA MsiEnumProducts 함수는 현재 보급되거나 설치된 모든 제품을 열거합니다. 사용자별 및 컴퓨터별 설치 컨텍스트와 보급 알림 모두에 설치된 제품이 열거됩니다. (ANSI)
MsiEnumProductsExA 현재 지정된 컨텍스트에 보급되거나 설치된 제품의 하나 또는 모든 인스턴스를 열거합니다. (ANSI)
MsiEnumProductsExW 현재 지정된 컨텍스트에 보급되거나 설치된 제품의 하나 또는 모든 인스턴스를 열거합니다. (유니코드)
MsiEnumProductsW MsiEnumProducts 함수는 현재 보급되거나 설치된 모든 제품을 열거합니다. 사용자별 및 컴퓨터별 설치 컨텍스트와 보급 알림 모두에 설치된 제품이 열거됩니다. (유니코드)
MsiEnumRelatedProductsA MsiEnumRelatedProducts 함수는 지정된 업그레이드 코드를 사용하여 제품을 열거합니다. 이 함수는 속성 테이블에 지정된 UpgradeCode 속성이 있는 현재 설치된 제품과 보급된 제품을 나열합니다. (ANSI)
MsiEnumRelatedProductsW MsiEnumRelatedProducts 함수는 지정된 업그레이드 코드를 사용하여 제품을 열거합니다. 이 함수는 속성 테이블에 지정된 UpgradeCode 속성이 있는 현재 설치된 제품과 보급된 제품을 나열합니다. (유니코드)
MsiEvaluateConditionA MsiEvaluateCondition 함수는 속성 이름과 값을 포함하는 조건식을 평가합니다. (ANSI)
MsiEvaluateConditionW MsiEvaluateCondition 함수는 속성 이름과 값을 포함하는 조건식을 평가합니다. (유니코드)

MsiExtractPatchXMLDataA MsiExtractPatchXMLData 함수는 패치가 대상 시스템에 적용되는지 확인하는 데 사용할 수 있는 패치에서 정보를 추출합니다. (ANSI)
MsiExtractPatchXMLDataW MsiExtractPatchXMLData 함수는 패치가 대상 시스템에 적용되는지 확인하는 데 사용할 수 있는 패치에서 정보를 추출합니다. (유니코드)
MsiFormatRecordA MsiFormatRecord 함수는 형식 문자열을 사용하여 레코드 필드 데이터 및 속성의 형식을 지정합니다. (ANSI)
MsiFormatRecordW MsiFormatRecord 함수는 형식 문자열을 사용하여 레코드 필드 데이터 및 속성의 형식을 지정합니다. (유니코드)
MsiGetActiveDatabase MsiGetActiveDatabase 함수는 설치에 대한 활성 데이터베이스를 반환합니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 읽기 전용 핸들을 반환합니다.
MsiGetComponentPathA MsiGetComponentPath 함수는 설치된 구성 요소에 대한 전체 경로를 반환합니다. 구성 요소의 키 경로가 레지스트리 키인 경우 레지스트리 키가 반환됩니다. (ANSI)
MsiGetComponentPathExA 설치된 구성 요소의 전체 경로를 반환합니다. (ANSI)
MsiGetComponentPathExW 설치된 구성 요소의 전체 경로를 반환합니다. (유니코드)
MsiGetComponentPathW MsiGetComponentPath 함수는 설치된 구성 요소에 대한 전체 경로를 반환합니다. 구성 요소의 키 경로가 레지스트리 키인 경우 레지스트리 키가 반환됩니다. (유니코드)
MsiGetComponentStateA MsiGetComponentState 함수는 구성 요소의 상태를 가져옵니다. (ANSI)

MsiGetComponentStateW MsiGetComponentState 함수는 구성 요소의 상태를 가져옵니다. (유니코드)
MsiGetDatabaseState MsiGetDatabaseState 함수는 데이터베이스의 상태를 반환합니다.
MsiGetFeatureCostA MsiGetFeatureCost 함수는 기능 및 선택한 자식 및 부모 기능에 필요한 디스크 공간을 반환합니다. (ANSI)
MsiGetFeatureCostW MsiGetFeatureCost 함수는 기능 및 선택한 자식 및 부모 기능에 필요한 디스크 공간을 반환합니다. (유니코드)
MsiGetFeatureInfoA 기능에 대한 설명 정보를 반환합니다. (ANSI)
MsiGetFeatureInfoW 기능에 대한 설명 정보를 반환합니다. (유니코드)
MsiGetFeatureStateA MsiGetFeatureState 함수는 기능의 요청된 상태를 가져옵니다. (ANSI)
MsiGetFeatureStateW MsiGetFeatureState 함수는 기능의 요청된 상태를 가져옵니다. (유니코드)
MsiGetFeatureUsageA MsiGetFeatureUsage 함수는 제품 기능에 대한 사용 메트릭을 반환합니다. (ANSI)
MsiGetFeatureUsageW MsiGetFeatureUsage 함수는 제품 기능에 대한 사용 메트릭을 반환합니다. (유니코드)
MsiGetFeatureValidStatesA MsiGetFeatureValidStates 함수는 유효한 설치 상태를 반환합니다. (ANSI)
MsiGetFeatureValidStatesW MsiGetFeatureValidStates 함수는 유효한 설치 상태를 반환합니다. (유니코드)

MsiGetFileHashA MsiGetFileHash 함수는 파일 경로를 사용하여 해당 파일의 128비트 해시를 반환합니다. 제작 도구는 MsiGetFileHash를 사용하여 MsiFileHash 테이블을 채우는 데 필요한 파일 해시를 가져올 수 있습니다. (ANSI)
MsiGetFileHashW MsiGetFileHash 함수는 파일 경로를 사용하여 해당 파일의 128비트 해시를 반환합니다. 제작 도구는 MsiGetFileHash를 사용하여 MsiFileHash 테이블을 채우는 데 필요한 파일 해시를 가져올 수 있습니다. (유니코드)
MsiGetFileSignatureInformationA MsiGetFileSignatureInformation 함수는 디지털 서명된 파일의 경로를 가져와서 파일의 서명자 인증서 및 해시를 반환합니다. (ANSI)
MsiGetFileSignatureInformationW MsiGetFileSignatureInformation 함수는 디지털 서명된 파일의 경로를 가져와서 파일의 서명자 인증서 및 해시를 반환합니다. (유니코드)
MsiGetFileVersionA MsiGetFileVersion은 설치 관리자가 데이터베이스에서 찾을 것으로 예상되는 형식으로 버전 문자열 및 언어 문자열을 반환합니다. (ANSI)
MsiGetFileVersionW MsiGetFileVersion은 설치 관리자가 데이터베이스에서 찾을 것으로 예상되는 형식으로 버전 문자열 및 언어 문자열을 반환합니다. (유니코드)
MsiGetLanguage MsiGetLanguage 함수는 현재 실행 중인 설치의 숫자 언어를 반환합니다.
MsiGetLastErrorRecord MsiGetLastErrorRecord 함수는 호출 프로세스에 대해 마지막으로 반환된 오류 레코드를 반환합니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다.
MsiGetMode MsiGetMode 함수는 테이블에 나열된 대로 설치 관리자가 현재 지정된 모드에서 실행 중인지 여부를 확인하는 데 사용됩니다.

MsiGetPatchFileListA MsiGetPatchFileList 함수는 세미콜론으로 구분된 .msp 파일 목록을 제공하고 패치로 업데이트할 수 있는 파일 목록을 검색합니다. (ANSI)
MsiGetPatchFileListW MsiGetPatchFileList 함수는 세미콜론으로 구분된 .msp 파일 목록을 제공하고 패치로 업데이트할 수 있는 파일 목록을 검색합니다. (유니코드)
MsiGetPatchInfoA MsiGetPatchInfo 함수는 패치에 대한 정보를 반환합니다. (ANSI)
MsiGetPatchInfoExA 제품의 지정된 인스턴스에 패치를 적용하는 방법에 대한 정보를 쿼리합니다. (ANSI)
MsiGetPatchInfoExW 제품의 지정된 인스턴스에 패치를 적용하는 방법에 대한 정보를 쿼리합니다. (유니코드)
MsiGetPatchInfoW MsiGetPatchInfo 함수는 패치에 대한 정보를 반환합니다. (유니코드)
MsiGetProductCodeA MsiGetProductCode 함수는 애플리케이션의 설치되거나 보급된 구성 요소의 구성 요소 코드를 사용하여 애플리케이션의 제품 코드를 반환합니다. (ANSI)
MsiGetProductCodeW MsiGetProductCode 함수는 애플리케이션의 설치되거나 보급된 구성 요소의 구성 요소 코드를 사용하여 애플리케이션의 제품 코드를 반환합니다. (유니코드)
MsiGetProductInfoA MsiGetProductInfo 함수는 게시된 제품 및 설치된 제품에 대한 제품 정보를 반환합니다. (ANSI)
MsiGetProductInfoExA 보급 및 설치된 제품에 대한 제품 정보를 반환합니다. (ANSI)
MsiGetProductInfoExW 보급 및 설치된 제품에 대한 제품 정보를 반환합니다. (유니코드)

MsiGetProductInfoFromScriptA MsiGetProductInfoFromScript 함수는 Windows Installer 스크립트 파일에 대한 제품 정보를 반환합니다. (ANSI)
MsiGetProductInfoFromScriptW MsiGetProductInfoFromScript 함수는 Windows Installer 스크립트 파일에 대한 제품 정보를 반환합니다. (유니코드)
MsiGetProductInfoW MsiGetProductInfo 함수는 게시된 제품 및 설치된 제품에 대한 제품 정보를 반환합니다. (유니코드)
MsiGetProductPropertyA MsiGetProductProperty 함수는 제품 속성을 검색합니다. 이러한 속성은 제품 데이터베이스에 있습니다. (ANSI)
MsiGetProductPropertyW MsiGetProductProperty 함수는 제품 속성을 검색합니다. 이러한 속성은 제품 데이터베이스에 있습니다. (유니코드)
MsiGetPropertyA MsiGetProperty 함수는 설치 관리자 속성의 값을 가져옵니다. (ANSI)
MsiGetPropertyW MsiGetProperty 함수는 설치 관리자 속성의 값을 가져옵니다. (유니코드)
MsiGetShortcutTargetA MsiGetShortcutTarget 함수는 바로 가기를 검사하고 제품, 기능 이름 및 구성 요소를 반환합니다(사용 가능한 경우). (ANSI)
MsiGetShortcutTargetW MsiGetShortcutTarget 함수는 바로 가기를 검사하고 제품, 기능 이름 및 구성 요소를 반환합니다(사용 가능한 경우). (유니코드)
MsiGetSourcePathA MsiGetSourcePath 함수는 디렉터리 테이블의 폴더에 대한 전체 원본 경로를 반환합니다. (ANSI)

MsiGetSourcePathW MsiGetSourcePath 함수는 디렉터리 테이블의 폴더에 대한 전체 원본 경로를 반환합니다. (유니코드)
MsiGetSummaryInformationA MsiGetSummaryInformation 함수는 설치 관리자 데이터베이스에 대한 _SummaryInformation 스트림에 대한 핸들을 가져옵니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다. (ANSI)
MsiGetSummaryInformationW MsiGetSummaryInformation 함수는 설치 관리자 데이터베이스에 대한 _SummaryInformation 스트림에 대한 핸들을 가져옵니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다. (유니코드)
MsiGetTargetPathA MsiGetTargetPath 함수는 디렉터리 테이블의 폴더에 대한 전체 대상 경로를 반환합니다. (ANSI)
MsiGetTargetPathW MsiGetTargetPath 함수는 디렉터리 테이블의 폴더에 대한 전체 대상 경로를 반환합니다. (유니코드)
MsiGetUserInfoA MsiGetUserInfo 함수는 설치된 제품에 대해 등록된 사용자 정보를 반환합니다. (ANSI)
MsiGetUserInfoW MsiGetUserInfo 함수는 설치된 제품에 대해 등록된 사용자 정보를 반환합니다. (유니코드)
MsiInstallMissingComponentA MsiInstallMissingComponent 함수는 예기치 않게 누락된 파일을 설치합니다. (ANSI)
MsiInstallMissingComponentW MsiInstallMissingComponent 함수는 예기치 않게 누락된 파일을 설치합니다. (유니코드)
MsiInstallMissingFileA MsiInstallMissingFile 함수는 예기치 않게 누락된 파일을 설치합니다. (ANSI)
MsiInstallMissingFileW MsiInstallMissingFile 함수는 예기치 않게 누락된 파일을 설치합니다. (유니코드)

MsiInstallProductA 제품을 설치하거나 제거합니다. (MsiInstallProductA)
MsiInstallProductW 제품을 설치하거나 제거합니다. (MsiInstallProductW)
MsiIsProductElevatedA MsiIsProductElevated 함수는 제품이 관리되는지 여부를 반환합니다. (ANSI)
MsiIsProductElevatedW MsiIsProductElevated 함수는 제품이 관리되는지 여부를 반환합니다. (유니코드)
MsiJoinTransaction MsiJoinTransaction 함수는 Windows Installer가 현재 프로세스를 다중 패키지 설치를 설치하는 트랜잭션의 소유자로 만들도록 요청합니다.
MsiLocateComponentA MsiLocateComponent 함수는 제품 코드 없이 설치된 구성 요소의 전체 경로를 반환합니다. (ANSI)
MsiLocateComponentW MsiLocateComponent 함수는 제품 코드 없이 설치된 구성 요소의 전체 경로를 반환합니다. (유니코드)
MsiNotifySidChangeA MsiNotifySidChange 함수는 사용자 SID에 대한 변경 내용으로 Windows Installer 내부 정보를 알리고 업데이트합니다. (ANSI)
MsiNotifySidChangeW MsiNotifySidChange 함수는 사용자 SID에 대한 변경 내용으로 Windows Installer 내부 정보를 알리고 업데이트합니다. (유니코드)
MsiOpenDatabaseA MsiOpenDatabase 함수는 데이터 액세스를 위한 데이터베이스 파일을 엽니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다. (ANSI)
MsiOpenDatabaseW MsiOpenDatabase 함수는 데이터 액세스를 위한 데이터베이스 파일을 엽니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다. (유니코드)

MsiOpenPackageA MsiOpenPackage 함수는 제품 데이터베이스에 액세스하는 함수와 함께 사용할 패키지를 엽니다. (ANSI)
MsiOpenPackageExA MsiOpenPackageEx 함수는 제품 데이터베이스에 액세스하는 함수와 함께 사용할 패키지를 엽니다. (ANSI)
MsiOpenPackageExW MsiOpenPackageEx 함수는 제품 데이터베이스에 액세스하는 함수와 함께 사용할 패키지를 엽니다. (유니코드)
MsiOpenPackageW MsiOpenPackage 함수는 제품 데이터베이스에 액세스하는 함수와 함께 사용할 패키지를 엽니다. (유니코드)
MsiOpenProductA MsiOpenProduct 함수는 제품 데이터베이스에 액세스하는 함수와 함께 사용할 제품을 엽니다. 핸들이 더 이상 필요하지 않은 경우 MsiCloseHandle 함수를 핸들로 호출해야 합니다. (ANSI)
MsiOpenProductW MsiOpenProduct 함수는 제품 데이터베이스에 액세스하는 함수와 함께 사용할 제품을 엽니다. 핸들이 더 이상 필요하지 않은 경우 MsiCloseHandle 함수를 핸들로 호출해야 합니다. (유니코드)
MsiPreviewBillboardA MsiPreviewBillboard 함수는 표시된 대화 상자에 호스트 컨트롤이 있는 광고판을 표시합니다. (ANSI)
MsiPreviewBillboardW MsiPreviewBillboard 함수는 표시된 대화 상자에 호스트 컨트롤이 있는 광고판을 표시합니다. (유니코드)
MsiPreviewDialogA MsiPreviewDialog 함수는 대화 상자를 모덜리스 및 비활성으로 표시합니다. (ANSI)
MsiPreviewDialogW MsiPreviewDialog 함수는 대화 상자를 모덜리스 및 비활성으로 표시합니다. (유니코드)

MsiProcessAdvertiseScriptA MsiProcessAdvertiseScript 함수는 지정된 위치에 보급 스크립트 파일을 처리합니다. (ANSI)
MsiProcessAdvertiseScriptW MsiProcessAdvertiseScript 함수는 지정된 위치에 보급 스크립트 파일을 처리합니다. (유니코드)
MsiProcessMessage MsiProcessMessage 함수는 처리를 위해 설치 관리자에 오류 레코드를 보냅니다.
MsiProvideAssemblyA MsiProvideAssembly 함수는 어셈블리가 포함된 Windows Installer 구성 요소의 전체 경로를 반환합니다. 이 함수는 원본을 묻는 메시지를 표시하고 필요한 설치를 수행합니다. MsiProvideAssembly는 기능에 대한 사용 횟수를 증분합니다. (ANSI)
MsiProvideAssemblyW MsiProvideAssembly 함수는 어셈블리가 포함된 Windows Installer 구성 요소의 전체 경로를 반환합니다. 이 함수는 원본을 묻는 메시지를 표시하고 필요한 설치를 수행합니다. MsiProvideAssembly는 기능에 대한 사용 횟수를 증분합니다. (유니코드)
MsiProvideComponentA MsiProvideComponent 함수는 필요한 설치를 수행하여 전체 구성 요소 경로를 반환합니다. 이 함수는 필요한 경우 원본을 묻는 메시지를 표시하고 기능에 대한 사용 횟수를 증분합니다. (ANSI)
MsiProvideComponentW MsiProvideComponent 함수는 필요한 설치를 수행하여 전체 구성 요소 경로를 반환합니다. 이 함수는 필요한 경우 원본을 묻는 메시지를 표시하고 기능에 대한 사용 횟수를 증분합니다. (유니코드)
MsiProvideQualifiedComponentA MsiProvideQualifiedComponent 함수는 정규화된 구성 요소에 대한 전체 구성 요소 경로를 반환하고 필요한 설치를 수행합니다. 이 함수는 필요한 경우 원본을 묻는 메시지를 표시하고 기능에 대한 사용 횟수를 증분합니다. (ANSI)
MsiProvideQualifiedComponentExA MsiProvideQualifiedComponentEx 함수는 제품에서 게시하고 필요한 설치를 수행하는 정규화된 구성 요소의 전체 구성 요소 경로를 반환합니다. (ANSI)
MsiProvideQualifiedComponentExW MsiProvideQualifiedComponentEx 함수는 제품에서 게시하고 필요한 설치를 수행하는 정규화된 구성 요소의 전체 구성 요소 경로를 반환합니다. (유니코드)

MsiProvideQualifiedComponentW MsiProvideQualifiedComponent 함수는 정규화된 구성 요소에 대한 전체 구성 요소 경로를 반환하고 필요한 설치를 수행합니다. 이 함수는 필요한 경우 원본을 묻는 메시지를 표시하고 기능에 대한 사용 횟수를 증분합니다. (유니코드)
MsiQueryComponentStateA MsiQueryComponentState 함수는 구성 요소에 대해 설치된 상태를 반환합니다. (ANSI)
MsiQueryComponentStateW MsiQueryComponentState 함수는 구성 요소에 대해 설치된 상태를 반환합니다. (유니코드)
MsiQueryFeatureStateA MsiQueryFeatureState 함수는 제품 기능에 대해 설치된 상태를 반환합니다. (ANSI)
MsiQueryFeatureStateExA MsiQueryFeatureStateEx 함수는 제품 기능에 대해 설치된 상태를 반환합니다. (ANSI)
MsiQueryFeatureStateExW MsiQueryFeatureStateEx 함수는 제품 기능에 대해 설치된 상태를 반환합니다. (유니코드)
MsiQueryFeatureStateW MsiQueryFeatureState 함수는 제품 기능에 대해 설치된 상태를 반환합니다. (유니코드)
MsiQueryProductStateA MsiQueryProductState 함수는 제품에 대해 설치된 상태를 반환합니다. (ANSI)
MsiQueryProductStateW MsiQueryProductState 함수는 제품에 대해 설치된 상태를 반환합니다. (유니코드)
MsiRecordClearData MsiRecordClearData 함수는 레코드의 모든 필드를 null로 설정합니다.
MsiRecordDataSize MsiRecordDataSize 함수는 레코드 필드의 길이를 반환합니다. 개수에는 종료 null 문자가 포함되지 않습니다.

MsiRecordGetFieldCount 레코드의 필드 수를 반환합니다.
MsiRecordGetInteger MsiRecordGetInteger 함수는 레코드 필드에서 정수 값을 반환합니다.
MsiRecordGetStringA MsiRecordGetString 함수는 레코드 필드의 문자열 값을 반환합니다. (ANSI)
MsiRecordGetStringW MsiRecordGetString 함수는 레코드 필드의 문자열 값을 반환합니다. (유니코드)
MsiRecordIsNull null 레코드 필드를 보고합니다.
MsiRecordReadStream MsiRecordReadStream 함수는 레코드 스트림 필드에서 버퍼로 바이트를 읽습니다.
MsiRecordSetInteger 레코드 필드를 정수 필드로 설정합니다.
MsiRecordSetStreamA MsiRecordSetStream 함수는 파일에서 레코드 스트림 필드를 설정합니다. 스트림 데이터를 임시 필드에 삽입할 수 없습니다. (ANSI)
MsiRecordSetStreamW MsiRecordSetStream 함수는 파일에서 레코드 스트림 필드를 설정합니다. 스트림 데이터를 임시 필드에 삽입할 수 없습니다. (유니코드)
MsiRecordSetStringA MsiRecordSetString 함수는 지정된 필드에 문자열을 복사합니다. (ANSI)
MsiRecordSetStringW MsiRecordSetString 함수는 지정된 필드에 문자열을 복사합니다. (유니코드)
MsiReinstallFeatureA 기능을 다시 설치합니다. (ANSI)

MsiReinstallFeatureW 기능을 다시 설치합니다. (유니코드)
MsiReinstallProductA 제품을 다시 설치합니다. (ANSI)
MsiReinstallProductW 제품을 다시 설치합니다. (유니코드)
MsiRemovePatchesA 단일 제품에서 하나 이상의 패치를 제거합니다. (ANSI)
MsiRemovePatchesW 단일 제품에서 하나 이상의 패치를 제거합니다. (유니코드)
MsiSequenceA MsiSequence 함수는 지정된 테이블에 설명된 대로 다른 작업 순서를 실행합니다. (ANSI)
MsiSequenceW MsiSequence 함수는 지정된 테이블에 설명된 대로 다른 작업 순서를 실행합니다. (유니코드)
MsiSetComponentStateA MsiSetComponentState 함수는 구성 요소를 요청된 상태로 설정합니다. (ANSI)
MsiSetComponentStateW MsiSetComponentState 함수는 구성 요소를 요청된 상태로 설정합니다. (유니코드)
MsiSetExternalUIA MsiSetExternalUI 함수는 외부 사용자 인터페이스 처리기를 사용하도록 설정합니다. (ANSI)
MsiSetExternalUIRecord MsiSetExternalUIRecord 함수는 외부 UI(사용자 인터페이스) 처리기를 사용하도록 설정합니다.
MsiSetExternalUIW MsiSetExternalUI 함수는 외부 사용자 인터페이스 처리기를 사용하도록 설정합니다. (유니코드)

MsiSetFeatureAttributesA MsiSetFeatureAttributes 함수는 런타임에 기능의 기본 특성을 수정할 수 있습니다. 기능의 기본 특성은 기능 테이블의 특성 열에 작성됩니다. (ANSI)
MsiSetFeatureAttributesW MsiSetFeatureAttributes 함수는 런타임에 기능의 기본 특성을 수정할 수 있습니다. 기능의 기본 특성은 기능 테이블의 특성 열에 작성됩니다. (유니코드)
MsiSetFeatureStateA MsiSetFeatureState 함수는 기능을 지정된 상태로 설정합니다. (ANSI)
MsiSetFeatureStateW MsiSetFeatureState 함수는 기능을 지정된 상태로 설정합니다. (유니코드)
MsiSetInstallLevel MsiSetInstallLevel 함수는 전체 제품 설치에 대한 설치 수준을 설정합니다.
MsiSetInternalUI MsiSetInternalUI 함수는 설치 관리자의 내부 사용자 인터페이스를 사용하도록 설정합니다. 그런 다음 이 사용자 인터페이스는 이 프로세스에서 사용자 인터페이스 생성 설치 관리자 함수에 대한 모든 후속 호출에 사용됩니다. 자세한 내용은 사용자 인터페이스 수준을 참조하세요.
MsiSetMode MsiSetMode 함수는 내부 엔진 부울 상태를 설정합니다.
MsiSetPropertyA MsiSetProperty 함수는 설치 속성의 값을 설정합니다. (ANSI)
MsiSetPropertyW MsiSetProperty 함수는 설치 속성의 값을 설정합니다. (유니코드)
MsiSetTargetPathA MsiSetTargetPath 함수는 디렉터리 테이블의 폴더에 대한 전체 대상 경로를 설정합니다. (ANSI)
MsiSetTargetPathW MsiSetTargetPath 함수는 디렉터리 테이블의 폴더에 대한 전체 대상 경로를 설정합니다. (유니코드)

MsiSourceListAddMediaDiskA MsiSourceListAddMediaDisk 함수는 등록된 제품 또는 패치의 미디어 원본 디스크를 추가하거나 업데이트합니다. (ANSI)
MsiSourceListAddMediaDiskW MsiSourceListAddMediaDisk 함수는 등록된 제품 또는 패치의 미디어 원본 디스크를 추가하거나 업데이트합니다. (유니코드)
MsiSourceListAddSourceA 지정된 사용자 컨텍스트에서 제품 또는 패치에 대해 지정된 유형의 원본을 포함하는 유효한 네트워크 원본 목록에 추가합니다. (ANSI)
MsiSourceListAddSourceExA 지정된 컨텍스트에서 패치 또는 제품의 원본 집합을 추가하거나 다시 정렬합니다. 지정된 컨텍스트에 없는 패치에 대한 원본 목록을 만들 수도 있습니다. (ANSI)
MsiSourceListAddSourceExW 지정된 컨텍스트에서 패치 또는 제품의 원본 집합을 추가하거나 다시 정렬합니다. 지정된 컨텍스트에 없는 패치에 대한 원본 목록을 만들 수도 있습니다. (유니코드)
MsiSourceListAddSourceW 지정된 사용자 컨텍스트에서 제품 또는 패치에 대해 지정된 유형의 원본을 포함하는 유효한 네트워크 원본 목록에 추가합니다. (유니코드)
MsiSourceListClearAllA MsiSourceListClearAll 함수는 지정된 컨텍스트에서 패치 또는 제품의 원본 목록에서 모든 네트워크 원본을 제거합니다. 자세한 내용은 원본 복원력을 참조하세요. (ANSI)
MsiSourceListClearAllExA MsiSourceListClearAllEx 함수는 지정된 제품 또는 패치 인스턴스에 대해 지정된 소스 형식의 모든 기존 원본을 제거합니다. (ANSI)
MsiSourceListClearAllExW MsiSourceListClearAllEx 함수는 지정된 제품 또는 패치 인스턴스에 대해 지정된 소스 형식의 모든 기존 원본을 제거합니다. (유니코드)
MsiSourceListClearAllW MsiSourceListClearAll 함수는 지정된 컨텍스트에서 패치 또는 제품의 원본 목록에서 모든 네트워크 원본을 제거합니다. 자세한 내용은 원본 복원력을 참조하세요. (유니코드)

MsiSourceListClearMediaDiskA MsiSourceListClearMediaDisk 함수는 특정 컨텍스트에서 제품 또는 패치에 대한 미디어 소스에서 기존의 등록된 디스크를 제거하는 기능을 제공합니다. (ANSI)
MsiSourceListClearMediaDiskW MsiSourceListClearMediaDisk 함수는 특정 컨텍스트에서 제품 또는 패치에 대한 미디어 소스에서 기존의 등록된 디스크를 제거하는 기능을 제공합니다. (유니코드)
MsiSourceListClearSourceA 지정된 컨텍스트에서 제품 또는 패치에 대한 기존 소스를 제거합니다. (ANSI)
MsiSourceListClearSourceW 지정된 컨텍스트에서 제품 또는 패치에 대한 기존 소스를 제거합니다. (유니코드)
MsiSourceListEnumMediaDisksA MsiSourceListEnumMediaDisks 함수는 패치 또는 제품의 미디어 원본에 등록된 디스크 목록을 열거합니다. (ANSI)
MsiSourceListEnumMediaDisksW MsiSourceListEnumMediaDisks 함수는 패치 또는 제품의 미디어 원본에 등록된 디스크 목록을 열거합니다. (유니코드)
MsiSourceListEnumSourcesA MsiSourceListEnumSources 함수는 지정된 패치 또는 제품의 원본 목록에 있는 원본을 열거합니다. (ANSI)
MsiSourceListEnumSourcesW MsiSourceListEnumSources 함수는 지정된 패치 또는 제품의 원본 목록에 있는 원본을 열거합니다. (유니코드)
MsiSourceListForceResolutionA MsiSourceListForceResolution 함수는 다음에 원본이 필요할 때 설치 관리자가 원본 목록에서 유효한 제품 원본을 검색하도록 합니다. (ANSI)
MsiSourceListForceResolutionExA MsiSourceListForceResolutionEx 함수는 "LastUsedSource"라는 속성의 등록을 제거합니다. (ANSI)

MsiSourceListForceResolutionExW MsiSourceListForceResolutionEx 함수는 "LastUsedSource"라는 속성의 등록을 제거합니다. (유니코드)
MsiSourceListForceResolutionW MsiSourceListForceResolution 함수는 다음에 원본이 필요할 때 설치 관리자가 원본 목록에서 유효한 제품 원본을 검색하도록 합니다. (유니코드)
MsiSourceListGetInfoA MsiSourceListGetInfo 함수는 특정 컨텍스트에서 제품 또는 패치의 원본 목록에 대한 정보를 검색합니다. (ANSI)
MsiSourceListGetInfoW MsiSourceListGetInfo 함수는 특정 컨텍스트에서 제품 또는 패치의 원본 목록에 대한 정보를 검색합니다. (유니코드)
MsiSourceListSetInfoA 특정 컨텍스트에서 제품 또는 패치의 원본 목록에 대한 정보를 설정합니다. (ANSI)
MsiSourceListSetInfoW 특정 컨텍스트에서 제품 또는 패치의 원본 목록에 대한 정보를 설정합니다. (유니코드)
MsiSummaryInfoGetPropertyA MsiSummaryInfoGetProperty 함수는 요약 정보 스트림에서 단일 속성을 가져옵니다. (ANSI)
MsiSummaryInfoGetPropertyCount MsiSummaryInfoGetPropertyCount 함수는 요약 정보 스트림의 기존 속성 수를 반환합니다.
MsiSummaryInfoGetPropertyW MsiSummaryInfoGetProperty 함수는 요약 정보 스트림에서 단일 속성을 가져옵니다. (유니코드)
MsiSummaryInfoPersist MsiSummaryInfoPersist 함수는 변경된 요약 정보를 요약 정보 스트림에 다시 씁니다.
MsiSummaryInfoSetPropertyA MsiSummaryInfoSetProperty 함수는 단일 요약 정보 속성을 설정합니다. (ANSI)

MsiSummaryInfoSetPropertyW MsiSummaryInfoSetProperty 함수는 단일 요약 정보 속성을 설정합니다. (유니코드)
MsiUseFeatureA MsiUseFeature 함수는 특정 기능의 사용 횟수를 증가시키고 해당 기능에 대한 설치 상태를 나타냅니다. 이 함수는 기능을 사용하려는 애플리케이션의 의도를 나타내는 데 사용해야 합니다. (ANSI)
MsiUseFeatureExA MsiUseFeatureEx 함수는 특정 기능의 사용 횟수를 증가시키고 해당 기능에 대한 설치 상태를 나타냅니다. 이 함수는 기능을 사용하려는 애플리케이션의 의도를 나타내는 데 사용해야 합니다. (ANSI)
MsiUseFeatureExW MsiUseFeatureEx 함수는 특정 기능의 사용 횟수를 증가시키고 해당 기능에 대한 설치 상태를 나타냅니다. 이 함수는 기능을 사용하려는 애플리케이션의 의도를 나타내는 데 사용해야 합니다. (유니코드)
MsiUseFeatureW MsiUseFeature 함수는 특정 기능의 사용 횟수를 증가시키고 해당 기능에 대한 설치 상태를 나타냅니다. 이 함수는 기능을 사용하려는 애플리케이션의 의도를 나타내는 데 사용해야 합니다. (유니코드)
MsiVerifyDiskSpace MsiVerifyDiskSpace 함수는 현재 설치에 충분한 디스크 공간이 있는지 확인합니다.
MsiVerifyPackageA MsiVerifyPackage 함수는 지정된 파일이 설치 패키지인지 확인합니다. (ANSI)
MsiVerifyPackageW MsiVerifyPackage 함수는 지정된 파일이 설치 패키지인지 확인합니다. (유니코드)
MsiViewClose MsiViewClose 함수는 실행된 뷰에 대한 결과 집합을 해제합니다.
MsiViewExecute MsiViewExecute 함수는 SQL 뷰 쿼리를 실행하고 필요한 매개 변수를 제공합니다.

MsiViewFetch MsiViewFetch 함수는 뷰에서 다음 순차 레코드를 가져옵니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다.
MsiViewGetColumnInfo MsiViewGetColumnInfo 함수는 열 이름 또는 정의가 포함된 레코드를 반환합니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다.
MsiViewGetErrorA MsiViewGetError 함수는 MsiViewModify 함수에서 발생한 오류를 반환합니다. (ANSI)
MsiViewGetErrorW MsiViewGetError 함수는 MsiViewModify 함수에서 발생한 오류를 반환합니다. (유니코드)
MsiViewModify MsiViewModify 함수는 가져온 레코드를 업데이트합니다.
OpenCUB OpenCUB 메서드는 유효성 검사에 사용할 ICE(내부 일관성 평가기) 파일을 엽니다.
OpenDatabase OpenDatabase 메서드는 유효성 검사를 위해 Windows Installer 설치 패키지 또는 병합 모듈을 엽니다.
OpenDatabase OpenDatabase 메서드는 모듈과 병합할 지정된 경로에 있는 Windows Installer 설치 데이터베이스를 엽니다. 자세한 내용은 Merge 개체의 OpenDatabase 메서드를 참조하세요.
OpenLog OpenLog 메서드는 진행률 및 오류 메시지를 수신하는 로그 파일을 엽니다.
OpenModule OpenModule 메서드는 읽기 전용 모드에서 Windows Installer 병합 모듈을 엽니다. 모듈을 설치 데이터베이스와 병합하려면 먼저 열어야 합니다. 자세한 내용은 Merge 개체의 OpenModule 메서드를 참조하세요.

ProvideIntegerData ProvideIntegerData 메서드는 클라이언트 도구에서 정수 데이터를 검색합니다. 자세한 내용은 ConfigureModule 개체의 ProvideIntegerData 메서드를 참조하세요.
ProvideTextData ProvideTextData 메서드는 클라이언트 도구에서 텍스트 데이터를 검색합니다. 자세한 내용은 ConfigureModule 개체의 ProvideTextData 메서드를 참조하세요.
PSP_FILE_CALLBACK_A FileCallback 콜백 함수는 몇 가지 설정 함수에서 사용됩니다. (ANSI)
PSP_FILE_CALLBACK_W FileCallback 콜백 함수는 몇 가지 설정 함수에서 사용됩니다. (유니코드)
QueryActCtxSettingsW QueryActCtxSettingsW 함수는 활성화 컨텍스트와 쿼리할 특성의 네임스페이스와 이름을 지정합니다.
QueryActCtxSettingsW QueryActCtxSettingsW 함수는 활성화 컨텍스트와 쿼리할 특성의 네임스페이스와 이름을 지정합니다.
QueryActCtxW QueryActCtxW 함수는 활성화 컨텍스트를 쿼리합니다.
QueryActCtxW QueryActCtxW 함수는 활성화 컨텍스트를 쿼리합니다.
QueryAssemblyInfo QueryAssemblyInfo 메서드는 어셈블리 정보를 위해 병렬 어셈블리 저장소를 쿼리하고 어셈블리 매니페스트에 대해 병렬 어셈블리 저장소의 파일의 유효성을 검사합니다.
ReleaseActCtx ReleaseActCtx 함수는 지정된 활성화 컨텍스트의 참조 수를 감소합니다.
ReleaseActCtx ReleaseActCtx 함수는 지정된 활성화 컨텍스트의 참조 수를 감소합니다.

SetDisplay SetDisplay 메서드를 사용하면 작성 도구가 콜백 함수를 통해 ICE 상태 메시지를 가져올 수 있습니다.
SetProperty SetProperty 메서드는 나란히 어셈블리 이름에 이름-값 쌍을 추가합니다. 이 메서드는 기존 이름-값 쌍의 값을 변경하거나 삭제할 수 있습니다.
SetStatus SetStatus 메서드를 사용하면 작성 도구가 등록된 콜백 함수를 통해 유효성 검사 진행률에 대한 정보를 받을 수 있습니다.
SetupAddInstallSectionToDiskSpaceListA SetupAddInstallSectionToDiskSpaceList 함수는 INF 파일의 설치 섹션에서 CopyFile 및 DelFile 줄을 검색합니다. 그런 다음, 함수는 해당 섹션에 지정된 파일 작업을 디스크 공간 목록에 추가합니다. (ANSI)
SetupAddInstallSectionToDiskSpaceListW SetupAddInstallSectionToDiskSpaceList 함수는 INF 파일의 설치 섹션에서 CopyFile 및 DelFile 줄을 검색합니다. 그런 다음, 함수는 해당 섹션에 지정된 파일 작업을 디스크 공간 목록에 추가합니다. (유니코드)
SetupAddSectionToDiskSpaceListA SetupAddSectionToDiskSpaceList 함수는 INF 파일의 파일 복사 또는 파일 삭제 섹션에 나열된 모든 파일 삭제 또는 복사 작업을 디스크 공간 목록에 추가합니다. (ANSI)
SetupAddSectionToDiskSpaceListW SetupAddSectionToDiskSpaceList 함수는 INF 파일의 파일 복사 또는 파일 삭제 섹션에 나열된 모든 파일 삭제 또는 복사 작업을 디스크 공간 목록에 추가합니다. (유니코드)
SetupAddToDiskSpaceListA SetupAddToDiskSpaceList 함수는 단일 삭제 또는 복사 작업을 디스크 공간 목록에 추가합니다. INF 파일의 섹션에 있는 모든 파일 작업을 추가하려면 SetupAddSectionToDiskSpaceList 또는 SetupAddInstallSectionToDiskSpaceList를 사용합니다. (ANSI)
SetupAddToDiskSpaceListW SetupAddToDiskSpaceList 함수는 단일 삭제 또는 복사 작업을 디스크 공간 목록에 추가합니다. INF 파일의 섹션에 있는 모든 파일 작업을 추가하려면 SetupAddSectionToDiskSpaceList 또는 SetupAddInstallSectionToDiskSpaceList를 사용합니다. (유니코드)

SetupAddToSourceListA SetupAddToSourceList 함수는 현재 사용자 또는 시스템의 설치 원본 목록에 값을 추가합니다. 값이 이미 있는 경우 중복 항목이 만들어지지 않도록 먼저 제거됩니다. (ANSI)
SetupAddToSourceListW SetupAddToSourceList 함수는 현재 사용자 또는 시스템의 설치 원본 목록에 값을 추가합니다. 값이 이미 있는 경우 중복 항목이 만들어지지 않도록 먼저 제거됩니다. (유니코드)
SetupAdjustDiskSpaceListA SetupAdjustDiskSpaceList 함수는 지정된 드라이브에 필요한 공간의 양을 조정합니다. (ANSI)
SetupAdjustDiskSpaceListW SetupAdjustDiskSpaceList 함수는 지정된 드라이브에 필요한 공간의 양을 조정합니다. (유니코드)
SetupBackupErrorA SetupBackupError 함수는 사용자에게 백업 오류를 알리는 대화 상자를 생성합니다. (ANSI)
SetupBackupErrorW SetupBackupError 함수는 사용자에게 백업 오류를 알리는 대화 상자를 생성합니다. (유니코드)
SetupCancelTemporarySourceList SetupCancelTemporarySourceList 함수는 임시 목록 및 찾아보기 없음 동작을 취소하고 표준 목록 동작을 다시 설정합니다.
SetupCloseFileQueue SetupCloseFileQueue 함수는 설치 파일 큐를 제거합니다.
SetupCloseInfFile SetupCloseInfFile 함수는 SetupOpenInfFile 호출로 열린 INF 파일을 닫습니다. 이 함수는 SetupOpenAppendInfFile을 호출하여 추가된 모든 INF 파일을 닫습니다.
SetupCloseLog SetupCloseLog 함수는 로그 파일을 닫습니다.
SetupCommitFileQueueA SetupCommitFileQueue 함수는 설치 파일 큐에 대기 중인 파일 작업을 수행합니다. (ANSI)

SetupCommitFileQueueW SetupCommitFileQueue 함수는 설치 파일 큐에 대기 중인 파일 작업을 수행합니다. (유니코드)
SetupConfigureWmiFromInfSectionA SetupConfigureWmiFromInfSection 함수는 [DDInstall.WMI] 섹션에 전달될 때 INF 파일에 의해 노출되는 WMI 데이터의 보안을 구성합니다. (ANSI)
SetupConfigureWmiFromInfSectionW SetupConfigureWmiFromInfSection 함수는 [DDInstall.WMI] 섹션에 전달될 때 INF 파일에 의해 노출되는 WMI 데이터의 보안을 구성합니다. (유니코드)
SetupCopyErrorA SetupCopyError 함수는 사용자에게 복사 파일 오류를 알리는 대화 상자를 생성합니다. (ANSI)
SetupCopyErrorW SetupCopyError 함수는 사용자에게 복사 파일 오류를 알리는 대화 상자를 생성합니다. (유니코드)
SetupCopyOEMInfA SetupCopyOEMInf 함수는 지정된 .inf 파일을 %windir%/Inf 디렉터리에 복사합니다. (ANSI)
SetupCopyOEMInfW SetupCopyOEMInf 함수는 지정된 .inf 파일을 %windir%/Inf 디렉터리에 복사합니다. (유니코드)
SetupCreateDiskSpaceListA SetupCreateDiskSpaceList 함수는 디스크 공간 목록을 만듭니다. (ANSI)
SetupCreateDiskSpaceListW SetupCreateDiskSpaceList 함수는 디스크 공간 목록을 만듭니다. (유니코드)
SetupDecompressOrCopyFileA SetupDecompressOrCopyFile 함수는 파일을 복사하여 필요한 경우 압축을 풉니다. (ANSI)
SetupDecompressOrCopyFileW SetupDecompressOrCopyFile 함수는 파일을 복사하여 필요한 경우 압축을 풉니다. (유니코드)

SetupDefaultQueueCallbackA SetupDefaultQueueCallback 함수는 설치 API에 포함된 기본 큐 콜백 루틴입니다. 이 함수를 사용하여 SetupCommitFileQueue 함수에서 보낸 알림을 처리할 수 있습니다. (ANSI)
SetupDefaultQueueCallbackW SetupDefaultQueueCallback 함수는 설치 API에 포함된 기본 큐 콜백 루틴입니다. 이 함수를 사용하여 SetupCommitFileQueue 함수에서 보낸 알림을 처리할 수 있습니다. (유니코드)
SetupDeleteErrorA SetupDeleteError 함수는 삭제 오류를 사용자에게 알리는 대화 상자를 생성합니다. (ANSI)
SetupDeleteErrorW SetupDeleteError 함수는 삭제 오류를 사용자에게 알리는 대화 상자를 생성합니다. (유니코드)
SetupDestroyDiskSpaceList SetupDestroyDiskSpaceList 함수는 디스크 공간 목록을 삭제하고 할당된 리소스를 해제합니다.
SetupDiGetClassDevsA SetupDiGetClassDevs 함수는 로컬 컴퓨터에 대해 요청된 디바이스 정보 요소가 포함된 디바이스 정보 집합에 대한 핸들을 반환합니다. (ANSI)
SetupDuplicateDiskSpaceListA SetupDuplicateDiskSpaceList 함수는 디스크 공간 목록을 새 독립 디스크 공간 목록으로 복제합니다. (ANSI)
SetupDuplicateDiskSpaceListW SetupDuplicateDiskSpaceList 함수는 디스크 공간 목록을 새 독립 디스크 공간 목록으로 복제합니다. (유니코드)
SetupEnumInfSectionsA SetupEnumInfSections 함수는 INF 파일에서 섹션 이름을 검색합니다. (ANSI)
SetupEnumInfSectionsW SetupEnumInfSections 함수는 INF 파일에서 섹션 이름을 검색합니다. (유니코드)
SetupFindFirstLineA SetupFindFirstLine 함수는 INF 파일의 지정된 섹션에서 줄을 찾습니다. 키 매개 변수가 NULL이면 SetupFindFirstLine은 섹션의 첫 번째 줄을 반환합니다. (ANSI)

SetupFindFirstLineW SetupFindFirstLine 함수는 INF 파일의 지정된 섹션에서 줄을 찾습니다. 키 매개 변수가 NULL이면 SetupFindFirstLine은 섹션의 첫 번째 줄을 반환합니다. (유니코드)
SetupFindNextLine SetupFindNextLine은 ContextIn.Line을 기준으로 INF 파일 섹션에서 다음 줄의 위치를 반환합니다.
SetupFindNextMatchLineA SetupFindNextMatchLine 함수는 지정된 키와 일치하는 ContextIn.Line을 기준으로 INF 파일에서 다음 줄의 위치를 반환합니다. (ANSI)
SetupFindNextMatchLineW SetupFindNextMatchLine 함수는 지정된 키와 일치하는 ContextIn.Line을 기준으로 INF 파일에서 다음 줄의 위치를 반환합니다. (유니코드)
SetupFreeSourceListA SetupFreeSourceList 함수는 원본 목록에 할당된 시스템 리소스를 해제합니다. (ANSI)
SetupFreeSourceListW SetupFreeSourceList 함수는 원본 목록에 할당된 시스템 리소스를 해제합니다. (유니코드)
SetupGetBinaryField SetupGetBinaryField 함수는 지정된 필드에서 줄 끝까지 INF 파일 섹션의 줄에서 이진 데이터를 검색합니다.
SetupGetFieldCount SetupGetFieldCount 함수는 INF 파일의 지정된 줄에 있는 필드 수를 검색합니다.
SetupGetFileCompressionInfoA SetupGetFileCompressionInfo 함수는 물리적 파일을 검사하여 압축되었는지 확인하고 압축되지 않은 대상 파일의 전체 경로, 크기 및 크기를 가져옵니다. (ANSI)
SetupGetFileCompressionInfoExA SetupGetFileCompressionInfoEx 함수는 잠재적으로 압축된 파일을 검사하고 압축 형식, 파일의 전체 경로(파일 이름 포함), 압축된 크기 및 압축되지 않은 대상 파일의 크기를 가져옵니다. (ANSI)

SetupGetFileCompressionInfoExW SetupGetFileCompressionInfoEx 함수는 잠재적으로 압축된 파일을 검사하고 압축 형식, 파일의 전체 경로(파일 이름 포함), 압축된 크기 및 압축되지 않은 대상 파일의 크기를 가져옵니다. (유니코드)
SetupGetFileCompressionInfoW SetupGetFileCompressionInfo 함수는 물리적 파일을 검사하여 압축되었는지 확인하고 압축되지 않은 대상 파일의 전체 경로, 크기 및 크기를 가져옵니다. (유니코드)
SetupGetFileQueueCount SetupGetFileQueueCount 함수는 설치 파일 큐에서 개수를 가져옵니다.
SetupGetFileQueueFlags SetupGetFileQueueFlags 함수는 설치 파일 큐에서 플래그를 가져옵니다.
SetupGetInfFileListA SetupGetInfFileList 함수는 호출자가 지정한 디렉터리에 있는 INF 파일 목록을 호출 제공 버퍼에 반환합니다. (ANSI)
SetupGetInfFileListW SetupGetInfFileList 함수는 호출자가 지정한 디렉터리에 있는 INF 파일 목록을 호출 제공 버퍼에 반환합니다. (유니코드)
SetupGetInfInformationA SetUpGetInfInformation 함수는 지정된 INF 파일의 SP_INF_INFORMATION 구조를 버퍼에 반환합니다. (ANSI)
SetupGetInfInformationW SetUpGetInfInformation 함수는 지정된 INF 파일의 SP_INF_INFORMATION 구조를 버퍼에 반환합니다. (유니코드)
SetupGetIntField SetupGetIntField 함수는 INF 파일의 지정된 줄 필드에서 정수 값을 검색합니다.
SetupGetLineByIndexA SetupGetLineByIndex 함수는 INF 파일의 지정된 섹션에서 인덱스 값으로 줄을 찾습니다. (ANSI)

SetupGetLineByIndexW SetupGetLineByIndex 함수는 INF 파일의 지정된 섹션에서 인덱스 값으로 줄을 찾습니다. (유니코드)
SetupGetLineCountA SetupGetLineCount 함수는 INF 파일의 지정된 섹션에 있는 줄 수를 반환합니다. (ANSI)
SetupGetLineCountW SetupGetLineCount 함수는 INF 파일의 지정된 섹션에 있는 줄 수를 반환합니다. (유니코드)
SetupGetLineTextA SetupGetLineText 함수는 INF 파일의 줄 내용을 압축 형식으로 반환합니다. (ANSI)
SetupGetLineTextW SetupGetLineText 함수는 INF 파일의 줄 내용을 압축 형식으로 반환합니다. (유니코드)
SetupGetMultiSzFieldA SetupGetMultiSzField 함수는 지정된 필드에서 줄 끝까지 INF 파일 줄에 저장된 여러 문자열을 검색합니다. (ANSI)
SetupGetMultiSzFieldW SetupGetMultiSzField 함수는 지정된 필드에서 줄 끝까지 INF 파일 줄에 저장된 여러 문자열을 검색합니다. (유니코드)
SetupGetSourceFileLocationA SetupGetSourceFileLocation 함수는 INF 파일에 나열된 원본 파일의 위치를 검색합니다. (ANSI)
SetupGetSourceFileLocationW SetupGetSourceFileLocation 함수는 INF 파일에 나열된 원본 파일의 위치를 검색합니다. (유니코드)
SetupGetSourceFileSizeA SetupGetSourceFileSize 함수는 INF 파일에 나열된 소스 파일의 압축되지 않은 크기를 읽습니다. (ANSI)
SetupGetSourceFileSizeW SetupGetSourceFileSize 함수는 INF 파일에 나열된 소스 파일의 압축되지 않은 크기를 읽습니다. (유니코드)

SetupGetSourceInfoA SetupGetSourceInfo 함수는 INF 파일에 나열된 원본에 대한 경로, 태그 파일 또는 미디어 설명을 검색합니다. (ANSI)
SetupGetSourceInfoW SetupGetSourceInfo 함수는 INF 파일에 나열된 원본에 대한 경로, 태그 파일 또는 미디어 설명을 검색합니다. (유니코드)
SetupGetStringFieldA SetupGetStringField 함수는 INF 파일의 지정된 줄 필드에서 문자열을 검색합니다. (ANSI)
SetupGetStringFieldW SetupGetStringField 함수는 INF 파일의 지정된 줄 필드에서 문자열을 검색합니다. (유니코드)
SetupGetTargetPathA SetupGetTargetPath 함수는 파일 목록 섹션의 대상 디렉터리를 결정합니다. (ANSI)
SetupGetTargetPathW SetupGetTargetPath 함수는 파일 목록 섹션의 대상 디렉터리를 결정합니다. (유니코드)
SetupInitDefaultQueueCallback SetupInitDefaultQueueCallback 함수는 설치 API에 포함된 기본 큐 콜백 루틴에서 사용하는 컨텍스트를 초기화합니다.
SetupInitDefaultQueueCallbackEx 진행률 메시지를 수락하기 위해 콜백 함수에 추가 창이 제공된다는 점을 제외하고 SetupInitDefaultQueueCallback과 동일한 방식으로 설치 API에 포함된 기본 큐 콜백 루틴에서 사용하는 컨텍스트를 초기화합니다.
SetupInitializeFileLogA SetupInitializeFileLog 함수는 설치 작업 및 결과를 기록하기 위해 파일을 초기화합니다. 시스템이 Windows의 일부로 설치된 파일 또는 다른 파일을 추적하는 시스템 로그일 수 있습니다. (ANSI)
SetupInitializeFileLogW SetupInitializeFileLog 함수는 설치 작업 및 결과를 기록하기 위해 파일을 초기화합니다. 시스템이 Windows의 일부로 설치된 파일 또는 다른 파일을 추적하는 시스템 로그일 수 있습니다. (유니코드)

SetupInstallFileA SetupInstallFile 함수는 SetupFindXXXLine에서 반환된 INFCONTEXT 또는 파일 이름 및 경로에 의해 명시적으로 지정된 대로 파일을 설치합니다. (ANSI)
SetupInstallFileExA SetupInstallFileEx 함수는 SetupFindXXXLine에서 반환된 INFCONTEXT 또는 파일 이름 및 경로 정보에 의해 명시적으로 지정된 대로 파일을 설치합니다. (ANSI)
SetupInstallFileExW SetupInstallFileEx 함수는 SetupFindXXXLine에서 반환된 INFCONTEXT 또는 파일 이름 및 경로 정보에 의해 명시적으로 지정된 대로 파일을 설치합니다. (유니코드)
SetupInstallFilesFromInfSectionA SetupInstallFilesFromInfSection 함수는 설치 섹션에 나열된 파일 복사, 파일 삭제 및 파일 이름 바꾸기 섹션에 지정된 설치에 대한 모든 파일을 큐에 대기합니다. (ANSI)
SetupInstallFilesFromInfSectionW SetupInstallFilesFromInfSection 함수는 설치 섹션에 나열된 파일 복사, 파일 삭제 및 파일 이름 바꾸기 섹션에 지정된 설치에 대한 모든 파일을 큐에 대기합니다. (유니코드)
SetupInstallFileW SetupInstallFile 함수는 SetupFindXXXLine에서 반환된 INFCONTEXT 또는 파일 이름 및 경로에 의해 명시적으로 지정된 대로 파일을 설치합니다. (유니코드)
SetupInstallFromInfSectionA SetupInstallFromInfSection 함수는 INF 파일 설치 섹션의 모든 지시문을 수행합니다. (ANSI)
SetupInstallFromInfSectionW SetupInstallFromInfSection 함수는 INF 파일 설치 섹션의 모든 지시문을 수행합니다. (유니코드)
SetupInstallServicesFromInfSectionA SetupInstallServicesFromInfSection 함수는 INF 파일의 서비스 섹션에 나열된 서비스 설치 섹션에 지정된 서비스 설치 및 삭제 작업을 수행합니다. (ANSI)
SetupInstallServicesFromInfSectionExA SetupInstallServicesFromInfSectionEx 함수는 INF 파일의 서비스 섹션에 나열된 서비스 설치 섹션에 지정된 서비스 설치 및 삭제 작업을 수행합니다. (ANSI)

SetupInstallServicesFromInfSectionExW SetupInstallServicesFromInfSectionEx 함수는 INF 파일의 서비스 섹션에 나열된 서비스 설치 섹션에 지정된 서비스 설치 및 삭제 작업을 수행합니다. (유니코드)
SetupInstallServicesFromInfSectionW SetupInstallServicesFromInfSection 함수는 INF 파일의 서비스 섹션에 나열된 서비스 설치 섹션에 지정된 서비스 설치 및 삭제 작업을 수행합니다. (유니코드)
SetupIterateCabinetA SetupIterateCabinet 함수는 캐비닛의 모든 파일을 반복하고 찾은 각 파일에 대한 콜백 함수에 알림을 보냅니다. (ANSI)
SetupIterateCabinetW SetupIterateCabinet 함수는 캐비닛의 모든 파일을 반복하고 찾은 각 파일에 대한 콜백 함수에 알림을 보냅니다. (유니코드)
SetupLogErrorA SetupLogError 함수는 로그 파일에 오류 메시지를 씁니다. (ANSI)
SetupLogErrorW SetupLogError 함수는 로그 파일에 오류 메시지를 씁니다. (유니코드)
SetupLogFileA SetupLogFile 함수는 로그 파일에 항목을 추가합니다. (ANSI)
SetupLogFileW SetupLogFile 함수는 로그 파일에 항목을 추가합니다. (유니코드)
SetupOpenAppendInfFileA SetupOpenAppendInfFile 함수는 INF 파일의 정보를 SetupOpenInfFile에서 이전에 연 INF 파일에 추가합니다. (ANSI)
SetupOpenAppendInfFileW SetupOpenAppendInfFile 함수는 INF 파일의 정보를 SetupOpenInfFile에서 이전에 연 INF 파일에 추가합니다. (유니코드)
SetupOpenFileQueue SetupOpenFileQueue 함수는 설치 파일 큐를 만듭니다.

SetupOpenInfFileA SetupOpenInfFile 함수는 INF 파일을 열고 핸들을 반환합니다. (ANSI)
SetupOpenInfFileW SetupOpenInfFile 함수는 INF 파일을 열고 핸들을 반환합니다. (유니코드)
SetupOpenLog SetupOpenLog 함수가 로그 파일을 엽니다.
SetupOpenMasterInf SetupOpenMasterInf 함수는 Windows와 함께 제공되는 파일에 대한 파일 및 레이아웃 정보를 포함하는 마스터 INF 파일을 엽니다.
SetupPromptForDiskA SetupPromptForDisk 함수는 사용자에게 디스크를 묻는 대화 상자를 표시합니다. (ANSI)
SetupPromptForDiskW SetupPromptForDisk 함수는 사용자에게 디스크를 묻는 대화 상자를 표시합니다. (유니코드)
SetupPromptReboot SetupPromptReboot 함수는 필요에 따라 커밋된 파일 큐의 파일이 파일 작업 중에 사용 중인지 여부에 따라 시스템을 다시 부팅할지 사용자에게 묻습니다.
SetupQueryDrivesInDiskSpaceListA SetupQueryDrivesInDiskSpaceList 함수는 디스크 공간 목록에 나열된 파일 작업에서 참조하는 드라이브 목록으로 버퍼를 채웁니다. (ANSI)
SetupQueryDrivesInDiskSpaceListW SetupQueryDrivesInDiskSpaceList 함수는 디스크 공간 목록에 나열된 파일 작업에서 참조하는 드라이브 목록으로 버퍼를 채웁니다. (유니코드)
SetupQueryFileLogA SetupQueryFileLog 함수는 설치 파일 로그에서 정보를 반환합니다. (ANSI)
SetupQueryFileLogW SetupQueryFileLog 함수는 설치 파일 로그에서 정보를 반환합니다. (유니코드)

SetupQueryInfFileInformationA SetupQueryInfFileInformation 함수는 SP_INF_INFORMATION 구조체에서 버퍼로 INF 파일 이름을 반환합니다. (ANSI)
SetupQueryInfFileInformationW SetupQueryInfFileInformation 함수는 SP_INF_INFORMATION 구조체에서 버퍼로 INF 파일 이름을 반환합니다. (유니코드)
SetupQueryInfOriginalFileInformationA SetupQueryInfOriginalFileInformation 함수는 OEM INF 파일의 원래 이름을 반환합니다. (ANSI)
SetupQueryInfOriginalFileInformationW SetupQueryInfOriginalFileInformation 함수는 OEM INF 파일의 원래 이름을 반환합니다. (유니코드)
SetupQueryInfVersionInformationA SetupQueryInfVersionInformation 함수는 SP_INF_INFORMATION 구조체에서 버퍼로 INF 파일 버전 정보를 반환합니다. (ANSI)
SetupQueryInfVersionInformationW SetupQueryInfVersionInformation 함수는 SP_INF_INFORMATION 구조체에서 버퍼로 INF 파일 버전 정보를 반환합니다. (유니코드)
SetupQuerySourceListA SetupQuerySourceList 함수는 현재 설치 원본 목록을 쿼리합니다. 목록은 시스템 및 사용자별 목록에서 작성되며 임시 목록에 의해 재정의될 수 있습니다(SetupSetSourceList 참조). (ANSI)
SetupQuerySourceListW SetupQuerySourceList 함수는 현재 설치 원본 목록을 쿼리합니다. 목록은 시스템 및 사용자별 목록에서 작성되며 임시 목록에 의해 재정의될 수 있습니다(SetupSetSourceList 참조). (유니코드)
SetupQuerySpaceRequiredOnDriveA SetupQuerySpaceRequiredOnDrive 함수는 디스크 공간 목록을 검사하여 특정 드라이브에 대해 나열된 모든 파일 작업을 수행하는 데 필요한 공간을 결정합니다. (ANSI)
SetupQuerySpaceRequiredOnDriveW SetupQuerySpaceRequiredOnDrive 함수는 디스크 공간 목록을 검사하여 특정 드라이브에 대해 나열된 모든 파일 작업을 수행하는 데 필요한 공간을 결정합니다. (유니코드)

SetupQueueCopyA SetupQueueCopy 함수는 설치 파일 큐에 단일 파일 복사 작업을 추가합니다. (ANSI)
SetupQueueCopyIndirectA SetupQueueCopyIndirect 함수는 추가 매개 변수를 구조체(SP_FILE_COPY_PARAMS)로 전달하는 확장된 형태의 SetupQueueCopy입니다. 이 외에는 동작이 동일합니다. (ANSI)
SetupQueueCopyIndirectW SetupQueueCopyIndirect 함수는 추가 매개 변수를 구조체(SP_FILE_COPY_PARAMS)로 전달하는 확장된 형태의 SetupQueueCopy입니다. 이 외에는 동작이 동일합니다. (유니코드)
SetupQueueCopySectionA SetupQueueCopySection 함수는 복사를 위해 INF 파일의 섹션에 있는 모든 파일을 설치 큐에 배치합니다. (ANSI)
SetupQueueCopySectionW SetupQueueCopySection 함수는 복사를 위해 INF 파일의 섹션에 있는 모든 파일을 설치 큐에 배치합니다. (유니코드)
SetupQueueCopyW SetupQueueCopy 함수는 설치 파일 큐에 단일 파일 복사 작업을 추가합니다. (유니코드)
SetupQueueDefaultCopyA SetupQueueDefaultCopy 함수는 INF 파일에 지정된 기본 원본 미디어 및 대상을 사용하여 복사를 위해 설치 파일 큐에 단일 파일을 추가합니다. (ANSI)
SetupQueueDefaultCopyW SetupQueueDefaultCopy 함수는 INF 파일에 지정된 기본 원본 미디어 및 대상을 사용하여 복사를 위해 설치 파일 큐에 단일 파일을 추가합니다. (유니코드)
SetupQueueDeleteA SetupQueueDelete 함수는 설치 파일 큐에 개별 파일 삭제 작업을 배치합니다. (ANSI)
SetupQueueDeleteSectionA SetupQueueDeleteSection 함수는 삭제를 위해 INF 파일의 섹션에 있는 모든 파일을 큐에 대기합니다. 섹션은 올바른 파일 삭제 형식이어야 하며 INF 파일에 DestinationDirs 섹션이 포함되어야 합니다. (ANSI)

SetupQueueDeleteSectionW SetupQueueDeleteSection 함수는 삭제를 위해 INF 파일의 섹션에 있는 모든 파일을 큐에 대기합니다. 섹션은 올바른 파일 삭제 형식이어야 하며 INF 파일에 DestinationDirs 섹션이 포함되어야 합니다. (유니코드)
SetupQueueDeleteW SetupQueueDelete 함수는 설치 파일 큐에 개별 파일 삭제 작업을 배치합니다. (유니코드)
SetupQueueRenameA SetupQueueRename 함수는 설치 파일 큐에 개별 파일 이름 바꾸기 작업을 배치합니다. (ANSI)
SetupQueueRenameSectionA SetupQueueRenameSection 함수는 이름을 바꾸기 위해 INF 파일의 섹션을 큐에 대기합니다. 섹션은 올바른 이름 바꾸기 목록 섹션 형식이어야 하며 INF 파일에 DestinationDirs 섹션이 포함되어야 합니다. (ANSI)
SetupQueueRenameSectionW SetupQueueRenameSection 함수는 이름을 바꾸기 위해 INF 파일의 섹션을 큐에 대기합니다. 섹션은 올바른 이름 바꾸기 목록 섹션 형식이어야 하며 INF 파일에 DestinationDirs 섹션이 포함되어야 합니다. (유니코드)
SetupQueueRenameW SetupQueueRename 함수는 설치 파일 큐에 개별 파일 이름 바꾸기 작업을 배치합니다. (유니코드)
SetupRemoveFileLogEntryA SetupRemoveFileLogEntry 함수는 파일 로그에서 항목 또는 섹션을 제거합니다. (ANSI)
SetupRemoveFileLogEntryW SetupRemoveFileLogEntry 함수는 파일 로그에서 항목 또는 섹션을 제거합니다. (유니코드)
SetupRemoveFromDiskSpaceListA SetupRemoveFromDiskSpaceList 함수는 디스크 공간 목록에서 파일 삭제 또는 복사 작업을 제거합니다. (ANSI)
SetupRemoveFromDiskSpaceListW SetupRemoveFromDiskSpaceList 함수는 디스크 공간 목록에서 파일 삭제 또는 복사 작업을 제거합니다. (유니코드)

SetupRemoveFromSourceListA SetupRemoveFromSourceList 함수는 현재 사용자 또는 시스템의 설치 원본 목록에서 값을 제거합니다. 시스템 및 사용자 목록은 런타임에 병합됩니다. (ANSI)
SetupRemoveFromSourceListW SetupRemoveFromSourceList 함수는 현재 사용자 또는 시스템의 설치 원본 목록에서 값을 제거합니다. 시스템 및 사용자 목록은 런타임에 병합됩니다. (유니코드)
SetupRemoveInstallSectionFromDiskSpaceListA SetupRemoveInstallSectionFromDiskSpaceList 함수는 INF 파일의 설치 섹션에서 CopyFiles 및 DelFiles 줄을 검색하고 디스크 공간 목록에서 해당 섹션에 지정된 파일 작업을 제거합니다. (ANSI)
SetupRemoveInstallSectionFromDiskSpaceListW SetupRemoveInstallSectionFromDiskSpaceList 함수는 INF 파일의 설치 섹션에서 CopyFiles 및 DelFiles 줄을 검색하고 디스크 공간 목록에서 해당 섹션에 지정된 파일 작업을 제거합니다. (유니코드)
SetupRemoveSectionFromDiskSpaceListA SetupRemoveSectionFromDiskSpaceList 함수는 디스크 공간 목록에서 INF 파일의 파일 복사 섹션에 나열된 파일 삭제 또는 복사 작업을 제거합니다. (ANSI)
SetupRemoveSectionFromDiskSpaceListW SetupRemoveSectionFromDiskSpaceList 함수는 디스크 공간 목록에서 INF 파일의 파일 복사 섹션에 나열된 파일 삭제 또는 복사 작업을 제거합니다. (유니코드)
SetupRenameErrorA RenameError 함수는 파일 이름 바꾸기 오류를 사용자에게 알리는 대화 상자를 생성합니다. (ANSI)
SetupRenameErrorW RenameError 함수는 파일 이름 바꾸기 오류를 사용자에게 알리는 대화 상자를 생성합니다. (유니코드)
SetupScanFileQueueA SetupScanFileQueue 함수는 설치 파일 큐를 검색하여 복사 목록의 각 노드에서 작업을 수행합니다. 작업은 플래그 집합으로 지정됩니다. 큐가 커밋되기 전이나 후에 이 함수를 호출할 수 있습니다. (ANSI)

SetupScanFileQueueW SetupScanFileQueue 함수는 설치 파일 큐를 검색하여 복사 목록의 각 노드에서 작업을 수행합니다. 작업은 플래그 집합으로 지정됩니다. 큐가 커밋되기 전이나 후에 이 함수를 호출할 수 있습니다. (유니코드)
SetupSetDirectoryIdA SetupSetDirectoryId 함수는 INF 파일의 디렉터리 식별자를 특정 디렉터리에 연결합니다. (ANSI)
SetupSetDirectoryIdExA SetupSetDirectoryIdEx 함수는 INF 파일의 디렉터리 식별자를 특정 디렉터리에 연결합니다. (ANSI)
SetupSetDirectoryIdExW SetupSetDirectoryIdEx 함수는 INF 파일의 디렉터리 식별자를 특정 디렉터리에 연결합니다. (유니코드)
SetupSetDirectoryIdW SetupSetDirectoryId 함수는 INF 파일의 디렉터리 식별자를 특정 디렉터리에 연결합니다. (유니코드)
SetupSetFileQueueAlternatePlatformA SetupSetFileQueueAlternatePlatform 함수는 파일 큐를 함수를 실행하는 플랫폼과 다른 대상 플랫폼과 연결합니다. 이 작업은 네이티브가 아닌 서명 확인을 사용하도록 설정하기 위해 수행됩니다. (ANSI)
SetupSetFileQueueAlternatePlatformW SetupSetFileQueueAlternatePlatform 함수는 파일 큐를 함수를 실행하는 플랫폼과 다른 대상 플랫폼과 연결합니다. 이 작업은 네이티브가 아닌 서명 확인을 사용하도록 설정하기 위해 수행됩니다. (유니코드)
SetupSetFileQueueFlags SetupSetFileQueueFlags 함수는 설치 파일 큐의 플래그를 설정합니다.
SetupSetPlatformPathOverrideA SetupSetPlatformPathOverride 함수는 다른 컴퓨터의 INF로 작업할 때 대상 머신에 대한 플랫폼 경로 재정의를 설정하는 데 사용됩니다. (ANSI)
SetupSetPlatformPathOverrideW SetupSetPlatformPathOverride 함수는 다른 컴퓨터의 INF로 작업할 때 대상 머신에 대한 플랫폼 경로 재정의를 설정하는 데 사용됩니다. (유니코드)

SetupSetSourceListA SetupSetSourceList 함수를 사용하면 호출자가 현재 사용자 또는 시스템(모든 사용자에게 공통)에 대한 설치 원본 목록을 설정할 수 있습니다. (ANSI)
SetupSetSourceListW SetupSetSourceList 함수를 사용하면 호출자가 현재 사용자 또는 시스템(모든 사용자에게 공통)에 대한 설치 원본 목록을 설정할 수 있습니다. (유니코드)
SetupTermDefaultQueueCallback 큐 커밋이 완료된 후 SetupTermDefaultQueueCallback 함수가 호출됩니다. SetupInitDefaultQueueCallback 또는 SetupInitDefaultQueueCallbackEx에 대한 이전 호출에 의해 할당된 리소스를 해제합니다.
SetupTerminateFileLog SetupTerminateFileLog 함수는 파일 로그와 연결된 리소스를 해제합니다.
SetupUninstallNewlyCopiedInfs SetupUninstallNewlyCopiedInfs 함수는 지정된 파일 큐의 커밋 중에 이전에 설치된 INF 파일(.inf), 미리 컴파일된 INF 파일(.pnf) 및 카탈로그 파일(.cat)을 제거합니다.
SetupUninstallOEMInfA SetupUninstallOEMInf 함수는 지정된 .inf 파일 및 연결된 모든 .pnf 파일을 제거합니다. (ANSI)
SetupUninstallOEMInfW SetupUninstallOEMInf 함수는 지정된 .inf 파일 및 연결된 모든 .pnf 파일을 제거합니다. (유니코드)
SetupVerifyInfFileA SetupVerifyInfFile 함수는 해당 카탈로그를 사용하여 지정된 INF 파일의 디지털 서명을 확인합니다. 대체 플랫폼에 대해 확인을 수행할 수 있습니다. (ANSI)
SetupVerifyInfFileW SetupVerifyInfFile 함수는 해당 카탈로그를 사용하여 지정된 INF 파일의 디지털 서명을 확인합니다. 대체 플랫폼에 대해 확인을 수행할 수 있습니다. (유니코드)
SfcGetNextProtectedFile 보호된 파일의 전체 목록을 검색합니다.

SfcIsFileProtected 지정된 파일이 보호되는지 여부를 확인합니다.
SfcIsKeyProtected 지정된 레지스트리 키가 보호되는지 여부를 확인합니다.
UninstallAssembly UnistallAssembly 메서드는 병렬 저장소에서 어셈블리에 대한 애플리케이션 참조를 제거합니다.
유효성 검사 Validate 메서드는 지정된 내부 일관성 평가기 파일을 사용하여 설치 패키지 또는 병합 모듈의 유효성 검사를 수행합니다.
ZombifyActCtx ZombifyActCtx 함수는 지정된 활성화 컨텍스트를 비활성화하지만 할당을 취소하지는 않습니다.
ZombifyActCtx ZombifyActCtx 함수는 지정된 활성화 컨텍스트를 비활성화하지만 할당을 취소하지는 않습니다.

인터페이스

IAssemblyCache IAssemblyCache 인터페이스를 사용하여 병렬 어셈블리를 설치, 제거 또는 쿼리할 수 있습니다. CreateAssemblyCache 함수를 호출하여 IAssemblyCache 인스턴스를 가져옵니다.
IAssemblyCacheItem IAssemblyCacheItem 인터페이스를 사용하여 스트림 기반 설치를 사용하여 병렬 어셈블리를 나란히 저장소에 설치할 수 있습니다.
IAssemblyName IAssemblyName 인터페이스는 나란히 어셈블리 이름을 나타냅니다.

IMsmConfigurableItem IMsmConfigurableItem 인터페이스는 ModuleConfiguration 테이블에서 단일 행을 관리합니다. 모듈에서 구성 가능한 단일 "특성"입니다. 인터페이스는 ModuleConfiguration 테이블의 각 열에 대해 하나씩 읽기 전용 속성으로 구성됩니다.
IMsmConfigureModule IMsmConfigureModule 인터페이스는 콜백 인터페이스입니다. 이렇게 하면 클라이언트가 병합 프로세스 중에 병합 구성 정보를 제공할 수 있습니다.
IMsmDependency IMsmDependency 인터페이스는 단일 모듈 종속성에 대한 세부 정보를 검색합니다.
IMsmError IMsmError 인터페이스는 단일 병합 오류에 대한 세부 정보를 검색합니다.
IMsmGetFiles IMsmGetFiles 인터페이스를 사용하면 클라이언트가 모듈의 특정 언어로 필요한 파일을 검색할 수 있습니다.
IMsmMerge IMsmMerge 인터페이스 및 IMsmMerge2 인터페이스는 Merge 개체에 대한 인터페이스를 제공합니다.
IMsmMerge2 IMsmMerge 인터페이스 및 IMsmMerge2 인터페이스는 Merge 개체에 대한 인터페이스를 제공합니다. IMsmMerge2 인터페이스는 클라이언트 병합 도구가 새 구성 가능한 모듈 기능을 활용하는 방법을 제공합니다.
IValidate IValidate 인터페이스를 사용하면 작성 도구가 내부 일관성 평가기 집합에 대해 Windows Installer 패키지의 유효성을 검사할 수 있습니다.

구조체

ACTCTX_SECTION_KEYED_DATA ACTCTX_SECTION_KEYED_DATA 구조체는 FindActCtxSectionString 및 FindActCtxSectionGuid 함수에서 GUID 또는 32비트 정수 태그가 지정된 활성화 컨텍스트 섹션과 함께 활성화 컨텍스트 정보를 반환하는 데 사용됩니다.
ACTCTX_SECTION_KEYED_DATA ACTCTX_SECTION_KEYED_DATA 구조체는 FindActCtxSectionString 및 FindActCtxSectionGuid 함수에서 GUID 또는 32비트 정수 태그가 지정된 활성화 컨텍스트 섹션과 함께 활성화 컨텍스트 정보를 반환하는 데 사용됩니다.
ACTCTXA ACTCTX 구조는 CreateActCtx 함수에서 활성화 컨텍스트를 만드는 데 사용됩니다. (ANSI)
ACTCTXA ACTCTX 구조는 CreateActCtx 함수에서 활성화 컨텍스트를 만드는 데 사용됩니다. (ANSI)
ACTCTXW ACTCTX 구조는 CreateActCtx 함수에서 활성화 컨텍스트를 만드는 데 사용됩니다. (유니코드)
ACTCTXW ACTCTX 구조는 CreateActCtx 함수에서 활성화 컨텍스트를 만드는 데 사용됩니다. (유니코드)
ACTIVATION_CONTEXT_ASSEMBLY_DETAILED_INFORMATION ACTIVATION_CONTEXT_ASSEMBLY_DETAILED_INFORMATION 구조체는 QueryActCtxW 함수에서 사용됩니다.
ACTIVATION_CONTEXT_COMPATIBILITY_INFORMATION ACTIVATION_CONTEXT_COMPATIBILITY_INFORMATION 구조체는 QueryActCtxW 함수에서 사용됩니다.
ACTIVATION_CONTEXT_DETAILED_INFORMATION ACTIVATION_CONTEXT_DETAILED_INFORMATION 구조체는 QueryActCtxW 함수에서 사용됩니다.
ACTIVATION_CONTEXT_RUN_LEVEL_INFORMATION ACTIVATION_CONTEXT_RUN_LEVEL_INFORMATION 구조체는 QueryActCtxW 함수에서 사용됩니다.

ASSEMBLY_FILE_DETAILED_INFORMATION ASSEMBLY_FILE_DETAILED_INFORMATION 구조체는 QueryActCtxW 함수에서 사용됩니다.
ASSEMBLY_INFO ASSEMBLY_INFO 구조체에는 병렬 어셈블리 저장소의 어셈블리에 대한 정보가 포함됩니다. 이 정보는 QueryAssemblyInfo 메서드에서 사용됩니다.
CABINET_INFO_A CABINET_INFO 구조체는 캐비닛 파일에 대한 정보를 저장합니다. SetupIterateCabinet 함수는 캐비닛 콜백 루틴에 SPFILENOTIFY_NEEDNEWCABINET 알림을 보낼 때 이 구조를 매개 변수로 지정합니다. (ANSI)
CABINET_INFO_W CABINET_INFO 구조체는 캐비닛 파일에 대한 정보를 저장합니다. SetupIterateCabinet 함수는 캐비닛 콜백 루틴에 SPFILENOTIFY_NEEDNEWCABINET 알림을 보낼 때 이 구조를 매개 변수로 지정합니다. (유니코드)
COMPATIBILITY_CONTEXT_ELEMENT COMPATIBILITY_CONTEXT_ELEMENT 구조체는 QueryActCtxW 함수에서 ACTIVATION_CONTEXT_COMPATIBILITY_INFORMATION 구조체의 일부로 사용됩니다.
FILE_IN_CABINET_INFO_A FILE_IN_CABINET_INFO 구조체는 캐비닛에 있는 파일에 대한 정보를 제공합니다. (ANSI)
FILE_IN_CABINET_INFO_W FILE_IN_CABINET_INFO 구조체는 캐비닛에 있는 파일에 대한 정보를 제공합니다. (유니코드)
FILEPATHS_A FILEPATHS 구조는 원본 및 대상 경로 정보를 저장합니다. 설치 함수는 FILEPATHS 구조를 콜백 루틴으로 전송된 여러 알림의 매개 변수로 보냅니다. 자세한 내용은 알림을 참조하세요. (ANSI)
FILEPATHS_SIGNERINFO_A FILEPATHS_SINGNERINFO 구조는 원본 및 대상 경로 정보와 파일 서명 정보를 저장합니다. (ANSI)
FILEPATHS_SIGNERINFO_W FILEPATHS_SINGNERINFO 구조는 원본 및 대상 경로 정보와 파일 서명 정보를 저장합니다. (유니코드)

FILEPATHS_W FILEPATHS 구조는 원본 및 대상 경로 정보를 저장합니다. 설치 함수는 FILEPATHS 구조를 콜백 루틴으로 전송된 여러 알림의 매개 변수로 보냅니다. 자세한 내용은 알림을 참조하세요. (유니코드)
FUSION_INSTALL_REFERENCE FUSION_INSTALL_REFERENCE 구조에는 병렬 어셈블리를 참조하는 애플리케이션에 대한 정보가 포함되어 있습니다.
INFCONTEXT INFCONTEXT 구조체는 SetupGetLineText와 같은 함수가 INF 파일을 탐색하는 데 사용하는 컨텍스트 정보를 저장합니다.
MSIFILEHASHINFO MSIFILEHASHINFO 구조체에는 MsiGetFileHash에서 반환되고 MsiFileHash 테이블에 사용되는 파일 해시 정보가 포함됩니다.
MSIPATCHSEQUENCEINFOA MSIPATCHSEQUENCEINFO 구조체는 MsiDeterminePatchSequence 및 MsiDetermineApplicablePatches 함수에서 사용됩니다. (ANSI)
MSIPATCHSEQUENCEINFOW MSIPATCHSEQUENCEINFO 구조체는 MsiDeterminePatchSequence 및 MsiDetermineApplicablePatches 함수에서 사용됩니다. (유니코드)
SOURCE_MEDIA_A SOURCE_MEDIA 구조는 원본 미디어 정보를 전달하기 위해 SPFILENOTIFY_NEEDMEDIA 알림과 함께 사용됩니다. (ANSI)
SOURCE_MEDIA_W SOURCE_MEDIA 구조는 원본 미디어 정보를 전달하기 위해 SPFILENOTIFY_NEEDMEDIA 알림과 함께 사용됩니다. (유니코드)
SP_ALTPLATFORM_INFO_V1 이 구조는 대체 플랫폼에 대한 정보를 SetupQueryInfOriginalFileInformation에 전달하는 데 사용됩니다.
SP_ALTPLATFORM_INFO_V2 SP_ALTPLATFORM_INFO_V2 구조체는 대체 플랫폼에 대한 정보를 SetupQueryInfOriginalFileInformation에 전달하는 데 사용됩니다.

SP_FILE_COPY_PARAMS_A SP_FILE_COPY_PARAMS 구조는 단일 파일 복사 작업을 설명합니다. (ANSI)
SP_FILE_COPY_PARAMS_W SP_FILE_COPY_PARAMS 구조는 단일 파일 복사 작업을 설명합니다. (유니코드)
SP_INF_INFORMATION SP_INF_INFORMATION 구조체는 스타일, 구성 INF 파일 수 및 버전 데이터를 포함하여 INF 파일에 대한 정보를 저장합니다.
SP_INF_SIGNER_INFO_V1_A SP_INF_SIGNER_INFO 구조체는 INF 파일의 디지털 서명에 대한 정보를 저장합니다. (sp_inf_signer_info_v1_a)
SP_INF_SIGNER_INFO_V1_W SP_INF_SIGNER_INFO 구조체는 INF 파일의 디지털 서명에 대한 정보를 저장합니다. (sp_inf_signer_info_v1_w)
SP_INF_SIGNER_INFO_V2_A SP_INF_SIGNER_INFO 구조체는 INF 파일의 디지털 서명에 대한 정보를 저장합니다. (sp_inf_signer_info_v2_a)
SP_INF_SIGNER_INFO_V2_W SP_INF_SIGNER_INFO 구조체는 INF 파일의 디지털 서명에 대한 정보를 저장합니다. (sp_inf_signer_info_v2_w)
SP_ORIGINAL_FILE_INFO_A SP_ORIGINAL_FILE_INFO 구조체는 SetupQueryInfOriginalFileInformation에서 반환된 원래 INF 파일 이름 및 카탈로그 파일 정보를 받습니다. (ANSI)
SP_ORIGINAL_FILE_INFO_W SP_ORIGINAL_FILE_INFO 구조체는 SetupQueryInfOriginalFileInformation에서 반환된 원래 INF 파일 이름 및 카탈로그 파일 정보를 받습니다. (유니코드)
SP_REGISTER_CONTROL_STATUSA SP_REGISTER_CONTROL_STATUS 구조에는 Windows 2000에서 DLL을 자체 등록하기 위해 RegisterDlls INF 지시문을 사용하여 등록되거나 등록 취소되는 파일에 대한 정보가 포함되어 있습니다. (ANSI)

SP_REGISTER_CONTROL_STATUSW

SP_REGISTER_CONTROL_STATUS 구조에는 Windows 2000에서 DLL을 자체 등록하기 위해 RegisterDlls INF 지시문을 사용하여 등록되거나 등록 취소되는 파일에 대한 정보가 포함되어 있습니다. (유니코드)

애플리케이션 복구 및 다시 시작

아티클2023. 08. 22.

Application Recovery 및 다시 시작 기술에 대한 개요입니다.

Application Recovery 및 다시 시작 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 애플리케이션 복구 및 다시 시작

Functions

ApplicationRecoveryFinished 호출하는 애플리케이션이 데이터 복구를 완료했음을 나타냅니다.
ApplicationRecoveryInProgress 호출하는 애플리케이션이 계속해서 데이터를 복구하고 있음을 나타냅니다.
GetApplicationRecoveryCallback 지정된 프로세스에 대해 등록된 콜백 루틴에 대한 포인터를 검색합니다. 반환된 주소는 프로세스의 가상 주소 공간에 있습니다.
GetApplicationRestartSettings 지정된 프로세스에 대해 등록된 다시 시작 정보를 검색합니다.
RegisterApplicationRecoveryCallback 복구를 위해 애플리케이션의 활성 instance 등록합니다.
RegisterApplicationRestart 다시 시작할 애플리케이션의 활성 instance 등록합니다.
UnregisterApplicationRecoveryCallback 복구 목록에서 애플리케이션의 활성 instance 제거합니다.
UnregisterApplicationRestart 다시 시작 목록에서 애플리케이션의 활성 instance 제거합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

오디오 디바이스 DDI 참조

아티클2023. 10. 05.

오디오 디바이스 DDI 참조 기술 개요입니다.

오디오 디바이스 DDI 참조를 개발하려면 다음 헤더가 필요합니다.

- audiomediatype.h
- audioenginebaseapo.h
- audioengineextensionapo.h
- baseaudioprocessingobject.h
- msapofxproxy.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 오디오 디바이스 DDI 참조

열거형

APO_FLAG APO_FLAG 열거형은 APO(오디오 처리 개체)에서 플래그로 사용되는 상수를 정의합니다.
APO_LOG_LEVEL IAudioProcessingObjectLoggingService::ApoLog로 기록된 APO 이벤트의 수준을 지정합니다.
APO_NOTIFICATION_TYPE APO_NOTIFICATION 형식을 지정합니다.
APO_REFERENCE_STREAM_PROPERTIES IApoAcousticEchoCancellation2::GetDesiredReferenceStreamProperties 콜백 메서드에 대한 루프백 스트림 속성을 지정합니다.
AUDIO_SYSTEMEFFECT_STATE SAPO(시스템 효과 오디오 처리 개체) 오디오 효과의 상태를 지정합니다.
DEVICE_ORIENTATION_TYPE 형식 APO_NOTIFICATION_TYPE_DEVICE_ORIENTATION 알림에 대한 디바이스 방향 값을 지정합니다.

KSPROPERTY_AUDIOEFFECTSDISCOVERY KSPROPERTY_AUDIOEFFECTSDISCOVERY 열거형은 API(오디오 처리 개체) 목록에서 사용되는 상수를 정의합니다.

Functions

AcceptInput 보조 입력 중 하나에서 APO에 대한 입력을 제공합니다.
AddAuxiliaryInput 보조 입력을 APO에 추가하고 초기화 매개 변수를 제공합니다.
AERT_Allocate AERT_Allocate 유틸리티 함수는 오디오 처리 개체에서 사용할 메모리 세그먼트를 할당하고 잠가줍니다.
AERT_Free AERT_Free 유틸리티 함수는 오디오 처리 개체가 오디오 데이터를 처리하는 데 사용할 수 있도록 AERT_Allocate 함수에 의해 잠긴 메모리를 해제합니다.
ApoLog APO 이벤트를 기록합니다.
APOProcess APOProcess 메서드를 사용하면 APO가 처리 단계를 통과합니다.
CalcInputFrames CalcInputFrames 메서드는 APO가 지정된 수의 출력 프레임을 생성하는 데 필요한 입력 프레임 수를 반환합니다.
CalcOutputFrames CalcOutputFrames 메서드는 APO가 지정된 수의 입력 프레임에 필요한 출력 프레임 수를 반환합니다.
커밋

변경한 후 이 메서드는 변경 내용을 저장합니다.
CreateAudioMediaType CreateAudioMediaType 함수는 호출자가 지정한 형식을 사용하여 오디오 형식을 설명하는 미디어 형식 개체를 만듭니다.
CreateAudioMediaTypeFromUncompressedAudioFormat CreateAudioMediaTypeFromUncompressedAudioFormat 함수는 UNCOMPRESSEDAUDIOFORMAT 구조에 제공된 정보를 사용하여 오디오 형식을 설명하는 미디어 형식 개체를 만듭니다.
GetApoNotificationRegistrationInfo 클라이언트가 APO 엔드포인트 및 시스템 효과 알림에 대한 알림 콜백을 수신하도록 등록할 수 있도록 시스템에서 호출됩니다.
GetApoNotificationRegistrationInfo2 클라이언트가 APO 엔드포인트 및 시스템 효과 알림에 대한 알림 콜백을 수신하도록 등록할 수 있도록 시스템에서 호출됩니다. 이 메서드는 현재 디바이스에서 실행 중인 Windows 버전에서 지원되는 알림 유형을 확인하는 데 사용할 수 있는 매개 변수를 추가합니다.
GetAt 항목의 속성 배열에서 속성 키를 가져옵니다.
GetAudioFormat GetAudioFormat 메서드는 오디오 데이터 형식에 대한 WAVEFORMATEX 구조를 반환합니다.
GetControllableSystemEffectsList 호출자가 현재 효과 목록을 가져올 수 있도록 시스템 효과 sAPO(오디오 처리 개체) 오디오 효과에 의해 구현됩니다.
GetCount 이 메서드는 파일에 연결된 속성 수의 개수를 반환합니다.
GetDesiredReferenceStreamProperties 연결된 오디오 엔드포인트에서 지원되는 경우 루프백 스트림에 대한 속성 집합을 요청합니다.
GetEffectsList GetEffectsList 메서드는 현재 활성 상태인 오디오 처리 효과 목록을 검색하는 데 사용되며 목록이 변경되면 신호를 받을 이벤트를 저장합니다.

Getformat GetFormat 메서드는 사용자 지정 형식의 IAudioMediaType 표현을 검색합니다.
GetFormatCount GetFormatCount 메서드는 시스템 효과 sAPO(오디오 처리 개체)에서 지원하는 사용자 지정 형식의 수를 검색합니다.
GetFormatRepresentation GetFormatRepresentation 메서드는 사용자 인터페이스에 표시될 수 있도록 사용자 지정 형식의 문자열 표현을 검색합니다.
GetInputChannelCount GetInputChannelCount는 이 APO에 대한 입력 채널 수(프레임당 샘플)를 반환합니다.
GetLatency GetLatency 메서드는 이 APO의 대기 시간을 반환합니다. 대기 시간은 APO의 처리 패스를 트래버스하는 데 프레임이 걸리는 시간입니다.
GetPreferredInputFormat API가 제공된 출력 형식에 대한 기본 입력 형식을 지정할 수 있도록 하는 콜백 함수입니다.
GetPreferredOutputFormat API가 제공된 입력 형식에 대한 기본 출력 형식을 지정할 수 있도록 하는 콜백 함수입니다.
GetRealTimeWorkQueue APO가 실시간 우선 순위로 실행해야 하는 작업을 예약하는 데 사용할 수 있는 작업 큐의 ID를 가져옵니다.
GetRegistrationProperties GetRegistrationProperties는 APO(오디오 처리 개체)의 등록 속성을 반환합니다.
GetUncompressedAudioFormat IAudioMediaType::GetUncompressedAudioFormat은 오디오 데이터 형식에 대한 정보를 반환합니다.
GetValue 이 메서드는 특정 속성에 대 한 데이터를 검색 합니다.

HandleNotification APO 엔드포인트 또는 시스템 효과에 대한 변경 내용을 클라이언트에 알리기 위해 시스템에서 호출됩니다.
초기화 Initialize 메서드는 APO를 초기화하고 가변 길이의 데이터를 지원합니다.
IsCompressedFormat IsCompressedFormat 메서드는 오디오 데이터 형식이 압축된 형식인지 여부를 결정합니다.
IsEqual IsEqual 메서드는 두 미디어 형식을 비교하고 동일한지 여부를 결정합니다.
IsInputFormatSupported 특정 보조 입력 형식이 APO에서 지원되는지 확인합니다.
IsInputFormatSupported 이 메서드는 Windows Vista 오디오 엔진과 협상하여 오디오 데이터 스트림에 대한 데이터 형식을 설정합니다.
IsOutputFormatSupported IsOutputFormatSupported 메서드는 특정 출력 형식이 지원되는지 확인하는 데 사용됩니다.
LockForProcess LockForProcess 메서드는 APO가 잠겨 있고 데이터를 처리할 준비가 되었는지 확인하는 데 사용됩니다.
RemoveAuxiliaryInput APO에서 보조 입력을 제거합니다.
Reset Reset 메서드는 APO를 원래 상태로 다시 설정합니다. 이 메서드는 APO의 입력 또는 출력에 연결된 연결 개체를 변경하지 않습니다.
SetAudioDeviceModulesManager **IAudioDeviceModulesManager**의 instance **IAudioDeviceModulesClient** 인터페이스를 구현하는 API(오디오 처리 개체)에 전달하기 위해 시스템에서 호출됩니다.

SetAudioSystemEffectState 호출자가 효과 상태를 설정할 수 있도록 SAPO(시스템 효과 오디오 처리 개체) 오디오 효과에 의해 구현됩니다.
SetValue 이 메서드는 속성 값을 설정하거나 기존 값을 바꾸거나 제거합니다.
UnlockForProcess UnlockForProcess 메서드는 LockForProcess 메서드에 의해 APO에 부과된 잠금을 해제합니다.

인터페이스

IApoAcousticEchoCancellation 이 인터페이스는 AEC(음향 에코 취소) 시나리오를 사용하도록 AMO에 의해 구현됩니다.
IApoAcousticEchoCancellation2 API가 참조 스트림의 원하는 속성을 지정할 수 있도록 IAcousticEchoCancellation을 확장합니다.
IApoAuxiliaryInputConfiguration 오디오 엔진이 보조 입력 스트림을 추가하고 제거할 수 있도록 API가 구현할 수 있는 메서드를 제공합니다.
IApoAuxiliaryInputRT APO의 보조 입력을 구동하는 데 사용되는 실시간 안전 인터페이스입니다.
IAudioDeviceModulesClient API(오디오 처리 개체)는 **IAudioDeviceModulesManager** instance 대한 참조를 얻기 위해 이 인터페이스를 구현합니다.
IAudioMediaType IAudioMediaType 인터페이스는 sAPO가 적절한 오디오 데이터 형식에 대해 오디오 엔진과 협상하는 데 사용되는 정보를 가져올 수 있도록 하는 메서드를 노출합니다.
IAudioProcessingObject

시스템 효과 SAPO(오디오 처리 개체)는 일반적으로 실시간 프로세스 스레드에서 사용되거나 호출됩니다.
IAudioProcessingObjectConfiguration IAudioProcessingObjectConfiguration 인터페이스는 APO를 구성하는 데 사용됩니다. 이 인터페이스는 메서드를 사용하여 처리를 위해 APO를 잠그고 잠금을 해제합니다.
IAudioProcessingObjectLoggingService API에 대한 로깅 서비스를 나타냅니다.
IAudioProcessingObjectNotifications APO 엔드포인트 및 시스템 효과 알림에 대한 일반적인 오디오 관련 알림을 등록하고 수신하기 위해 클라이언트에서 구현합니다.
IAudioProcessingObjectNotifications2 APO 엔드포인트 및 시스템 효과 알림에 대한 일반적인 오디오 관련 알림을 등록하고 수신하기 위해 클라이언트에서 구현합니다. 이 인터페이스는 현재 디바이스에서 실행 중인 Windows 버전에서 지원되는 알림 유형을 확인하는 기능을 추가합니다.
IAudioProcessingObjectPreferredFormatSupport 이 인터페이스는 API가 기본 입력 또는 출력 형식을 지정할 수 있도록 하기 위해 구현됩니다.
IAudioProcessingObjectRT 이 인터페이스는 실시간 모드에서 작동할 수 있으며 해당 메서드를 양식 실시간 처리 스레드라고 할 수 있습니다.
IAudioProcessingObjectRTQueueService API에 대한 실시간 작업 큐 서비스를 나타냅니다.
IAudioSystemEffects IAudioSystemEffects 인터페이스는 IUnknown에서 상속되는 기본 메서드를 사용하며 Initialize 메서드를 구현해야 합니다.
IAudioSystemEffects2 IAudioSystemEffects2 인터페이스는 지정된 모드에서 처리 개체에 대한 정보를 검색하기 위해 Windows 8.1 도입되었습니다.
IAudioSystemEffects3

APOInitSystemEffects3 구조체를 IAudioProcessingObject::Initialize 메서드에 전달해야 하는 클라이언트에 의해 구현됩니다.
IAudioSystemEffectsCustomFormats IAudioSystemEffectsCustomFormats 인터페이스는 Windows Vista 이상 버전의 Windows에서 지원됩니다.
IPropertyStore 이 인터페이스는 속성 값을 열거하고 조작하는 데 사용되는 메서드를 노출합니다.

구조체

AcousticEchoCanceller_Reference_Input 현재 AEC(Acoustic Echo Cancellation) 구성과 관련된 확장된 정보를 포함합니다.
APO_NOTIFICATION APO 엔드포인트 또는 시스템 효과에 대한 변경에 대한 알림을 나타냅니다.
APO_NOTIFICATION_DESCRIPTOR 요청된 APO 알림을 지정합니다.
APO_REG_PROPERTIES APO_REG_PROPERTIES 구조체는 IAudioProcessingObject::GetRegistrationProperties에서 APO(오디오 처리 개체)의 등록 속성을 반환하는 데 사용됩니다.
APOInitBaseStruct APOInitBaseStruct 구조체는 IAudioProcessingObject::Initialize의 다른 초기화 데이터보다 우선해야 하는 기본 초기화 헤더입니다.
APOInitSystemEffects APOInitSystemEffects 구조체는 초기화를 위해 시스템 효과 APO에 전달됩니다.
APOInitSystemEffects2 초기화를 위해 APO(오디오 처리 개체)에 추가 초기화 컨텍스트를 제공할 수 있도록 APOInitSystemEffects2 구조체가 Windows 8.1 도입되었습니다.

APOInitSystemEffects3 APO 초기화 매개 변수를 제공하여 APOInitSystemEffects2를 확장하여 로깅에 대한 서비스 공급자를 지정하는 기능을 추가합니다.
AUDIO_ENDPOINT_PROPERTY_CHANGE_APO_NOTIFICATION_DESCRIPTOR 엔드포인트 속성 변경 APO 알림을 지정합니다.
AUDIO_ENDPOINT_PROPERTY_CHANGE_NOTIFICATION 속성 변경 APO 알림을 나타냅니다.
AUDIO_ENDPOINT_VOLUME_APO_NOTIFICATION_DESCRIPTOR 엔드포인트 볼륨 APO 알림을 지정합니다.
AUDIO_ENDPOINT_VOLUME_CHANGE_NOTIFICATION 오디오 엔드포인트 볼륨 변경 APO 알림을 나타냅니다.
AUDIO_ENDPOINT_VOLUME_CHANGE_NOTIFICATION2 오디오 엔드포인트 볼륨 변경 APO 알림을 나타냅니다. 볼륨 변경 이벤트에 대한 추가 정보를 제공하는 업데이트된 AUDIO_ENDPOINT_VOLUME_CHANGE_NOTIFICATION 버전입니다.
AUDIO_ENVIRONMENT_STATE_CHANGE_NOTIFICATION 오디오 환경 변경 APO 알림을 나타냅니다.
AUDIO_MICROPHONE_BOOST_APO_NOTIFICATION_DESCRIPTOR 엔드포인트 마이크 부스트 APO 알림을 지정합니다.
AUDIO_MICROPHONE_BOOST_NOTIFICATION 오디오 마이크 부스트 APO 알림을 나타냅니다.
AUDIO_SYSTEMEFFECT SAPO(시스템 효과 오디오 처리 개체) 오디오 효과를 나타냅니다.
AUDIO_SYSTEMEFFECTS_PROPERTY_CHANGE_NOTIFICATION 시스템 오디오 효과 APO 알림을 나타냅니다.
AUDIO_VOLUME_NOTIFICATION_DATA2

볼륨 변경 알림 이벤트에 대한 정보를 나타냅니다. 이 구조체는 AUDIO_ENDPOINT_VOLUME_CHANGE_NOTIFICATION2 구조체에서 사용됩니다.
KSP_PINMODE KSP_PINMODE 구조체는 핀 팩터리에 대해 핀 속성 및 지원되는 오디오 처리 모드를 지정합니다.
MDEVICECAPSEX MDEVICECAPSEX 구조에는 PnP(플러그 앤 플레이) 디바이스 드라이버에 대한 디바이스 기능 정보가 포함되어 있습니다.
MIDIOPENDESC MIDIOPENDESC 구조체는 MIDI 디바이스를 여는 방법에 대한 정보를 제공하는 클라이언트로 채워진 구조체입니다.
UNCOMPRESSEDAUDIOFORMAT UNCOMPRESSEDAUDIOFORMAT 구조체는 압축되지 않은 오디오 데이터 형식의 프레임 속도, 채널 마스크 및 기타 특성을 지정합니다.
WAVEFORMATEX WAVEFORMATEX 구조체는 웨이브 오디오 스트림의 데이터 형식을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

자동화

아티클2023. 03. 09.

Automation 기술에 대한 개요입니다.

Automation을 개발하려면 다음 헤더가 필요합니다.

- oaidl.h
- oleauto.h
- olectl.h
- wia_xp.h
- wtypes.h

이 설명서는 OLE Automation과 관련이 있습니다. UI 자동화 대한 자세한 내용은 Windows Automation API 개요를 참조하세요.

열거형

CALLCONV METHODDATA 구조에 설명된 멤버 함수에서 사용하는 호출 규칙을 식별합니다.
DESCKIND 바인딩되는 형식 설명을 식별합니다.
FUNCFLAGS 함수 플래그를 지정합니다.
FUNCKIND 함수 형식을 지정합니다.
INVOKEKIND 함수가 호출되는 방식을 지정합니다.
LIBFLAGS 형식 라이브러리에 해당하는 플래그를 정의합니다.

REGKIND 형식 라이브러리를 등록하는 방법을 제어합니다.
SYSKIND 대상 운영 체제 플랫폼을 식별합니다.
TYPEFLAGS 형식 플래그입니다.
TYPEKIND 형식을 지정합니다.
VARENUM 변형 형식을 지정합니다.
VARFLAGS 변수 플래그를 지정합니다.
VARKIND 변수 형식을 지정합니다.

Functions

AddFuncDesc 형식 설명에 함수 설명을 추가합니다.
AddImplType 상속된 인터페이스 또는 구성 요소 개체 클래스(coclass)에 의해 구현된 인터페이스를 지정합니다.
AddRefTypeInfo 생성 중인 형식 설명에서 참조하는 형식 설명에 형식 설명을 추가합니다.
AddressOfMember DLL에 정의된 것과 같은 정적 함수나 변수의 주소를 검색합니다.

AddVarDesc 형식 설명에 변수 또는 데이터 멤버 설명을 추가합니다.
AfterTypeChange 형식이 변경된 후에 발생합니다.
바인딩 이름을 형식 멤버에 매핑하거나 형식 라이브러리에 있는 전역 변수와 함수를 바인딩합니다.
BindType 형식 라이브러리에 있는 형식 설명에 바인딩합니다.
BSTR_UserFree RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다. (BSTR_UserFree)
BSTR_UserFree64 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다. (BSTR_UserFree64)
BSTR_UserMarshal BSTR 개체를 RPC 버퍼로 마샬링합니다. (BSTR_UserMarshal)
BSTR_UserMarshal64 BSTR 개체를 RPC 버퍼로 마샬링합니다. (BSTR_UserMarshal64)
BSTR_UserSize BSTR 개체의 와이어 크기를 계산하고 해당 핸들과 데이터를 가져옵니다. (BSTR_UserSize)
BSTR_UserSize64 BSTR 개체의 와이어 크기를 계산하고 해당 핸들과 데이터를 가져옵니다. (BSTR_UserSize64)
BSTR_UserUnmarshal RPC 버퍼에서 BSTR 개체를 숨깁니다. (BSTR_UserUnmarshal)
BSTR_UserUnmarshal64 RPC 버퍼에서 BSTR 개체를 숨깁니다. (BSTR_UserUnmarshal64)

BstrFromVector BSTR을 반환하여 BSTR의 문자에 벡터의 각 요소를 할당합니다.
ClearCustData 사용자 지정 데이터 항목을 보관하는 데 사용되는 메모리를 해제합니다.
복제 현재 열거 상태의 복사본을 만듭니다.
CreateDispTypeInfo IDispatch 구현에 사용할 간소화된 형식 정보를 만듭니다.
CreateErrorInfo 제네릭 오류 개체의 인스턴스를 만듭니다.
CreateInstance 구성 요소 개체 클래스(coclass)를 설명하는 형식의 새 인스턴스를 만듭니다.
CreateStdDispatch 단일 함수 호출을 통해 IDispatch 인터페이스의 표준 구현을 만듭니다. 이렇게 하면 Automation을 통해 개체를 쉽게 노출할 수 있습니다.
CreateTypeInfo 형식 라이브러리 내에 새 형식 설명 인스턴스를 만듭니다.
CreateTypeLib ICreateTypeLib 인터페이스를 지원하는 새 개체 인스턴스에 대한 액세스를 제공합니다.
CreateTypeLib2 현재 파일 형식으로 형식 라이브러리를 만듭니다.
DefineFuncAsDllEntry DLL 진입점을 지정된 인덱스가 있는 함수와 연결합니다.
DeleteFuncDesc 인덱스 번호로 지정된 함수 설명을 삭제합니다.

DeleteFuncDescByMemId 지정된 함수 설명(FUNCDESC)을 삭제합니다.
DeleteImplType 인덱싱된 인터페이스에 대한 IMPLTYPE 플래그를 삭제합니다.
DeleteTypeInfo 형식 라이브러리에서 지정된 형식 정보를 삭제합니다.
DeleteVarDesc 지정된 VARDESC 구조를 삭제합니다. (ICreateTypeInfo2.DeleteVarDesc)
DeleteVarDescByMemId 지정된 VARDESC 구조를 삭제합니다. (ICreateTypeInfo2.DeleteVarDescByMemId)
DispCallFunc 사용자 지정된 Invoke에 대한 머신 독립성 기능을 제공하는 Invoke에 대한 하위 수준 도우미입니다. (DispCallFunc)
DispGetIDsOfNames 사용자 지정된 Invoke에 대한 머신 독립성 기능을 제공하는 Invoke에 대한 하위 수준 도우미입니다. (DispGetIDsOfNames)
DispGetParam DISPPARAMS 구조체에서 매개 변수를 검색하여 명명된 매개 변수와 위치 매개 변수를 모두 확인하고 매개 변수를 지정된 형식으로 강제 변환합니다.
DispInvoke 인터페이스에 대한 형식 정보가 제공되면 인터페이스에서 멤버 함수를 자동으로 호출합니다.
DosDateTimeToVariantTime 시간의 MS-DOS 표현을 변형에 저장된 날짜 및 시간 표현으로 변환합니다.
FindName 형식 라이브러리에 들어 있는 형식 설명의 항목을 찾습니다. 형식 라이브러리에 이름이 있는지 빠르게 확인하는 데 사용할 수 있습니다.

GetActiveObject OLE에 등록된 실행 중인 개체에 대한 포인터를 검색합니다.
GetAllCustData 라이브러리에 대한 모든 사용자 지정 데이터 항목을 가져옵니다. (ITypeInfo2.GetAllCustData)
GetAllCustData 라이브러리에 대한 모든 사용자 지정 데이터 항목을 가져옵니다. (ITypeLib2.GetAllCustData)
GetAllFuncCustData 지정한 함수에서 모든 사용자 지정 데이터를 가져옵니다.
GetAllImplTypeCustData 지정된 구현 형식에 대한 모든 사용자 지정 데이터를 가져옵니다.
GetAllParamCustData 지정된 함수 매개 변수에 대한 모든 사용자 지정 데이터를 가져옵니다.
GetAllVarCustData 사용자 지정 데이터에 대한 변수를 가져옵니다.
GetAltMonthNames 보조(대체) 월 이름을 검색합니다.
GetContainingTypeLib 포함된 형식 라이브러리 및 해당 형식 라이브러리 내의 형식 설명 인덱스를 검색합니다.
GetCustData 사용자 지정 데이터를 가져옵니다. (ITypeInfo2.GetCustData)
GetCustData 사용자 지정 데이터를 가져옵니다. (ITypeLib2.GetCustData)
GetDescription 오류에 대한 텍스트 설명을 반환합니다.

GetDllEntry DLL에 있는 함수에 대한 진입점의 설명 또는 사양을 검색합니다.
GetDocumentation 설명서 문자열, 전체 도움말 파일 이름과 경로, 지정된 형식 설명에 대한 도움말 항목의 컨텍스트 ID를 검색합니다.
GetDocumentation 라이브러리에 대한 설명서 문자열, 전체 도움말 파일 이름 및 경로 및 도움말 파일의 라이브러리 도움말 항목에 대한 컨텍스트 식별자를 검색합니다.
GetDocumentation2 설명서 문자열, 전체 도움말 파일 이름 및 경로, 사용할 지역화 컨텍스트, 도움말 파일의 라이브러리 도움말 항목에 대한 컨텍스트 ID를 검색합니다.
GetDocumentation2 라이브러리의 설명서 문자열, 전체 도움말 파일 이름 및 경로, 사용할 지역화 컨텍스트, 도움말 파일의 라이브러리 도움말 항목에 대한 컨텍스트 ID를 검색합니다.
GetErrorInfo 현재 논리 스레드에서 SetErrorInfo에 대한 이전 호출에서 설정한 오류 정보 포인터를 가져옵니다.
GetField 지정된 필드 이름의 값을 포함하는 VARIANT에 대한 포인터를 반환합니다.
GetFieldNames 레코드 필드의 이름을 가져옵니다.
GetFieldNoCopy 값을 복사하고 리소스를 할당하지 않고 지정된 필드 이름의 값에 대한 포인터를 반환합니다.
GetFuncCustData 지정된 함수에서 사용자 지정 데이터를 가져옵니다.
GetFuncDesc 지정된 함수에 대한 정보를 포함하는 FUNCDESC 구조를 검색합니다.

GetFuncIndexOfMemId 알려진 DISPID를 기준으로 특정 멤버에 바인딩합니다. 여기서는 멤버 이름이 알려져 있지 않습니다(예: 기본 멤버에 바인딩할 경우). (ITypeInfo2.GetFuncIndexOfMemId)
GetGuid 레코드 형식의 GUID를 가져옵니다.
GetGUID 오류를 정의한 인터페이스의 GUID(Globally Unique Identifier)를 반환합니다.
GetHelpContext 오류에 대한 도움말 컨텍스트 식별자(ID)를 반환합니다.
GetHelpFile 오류를 설명하는 도움말 파일의 경로를 반환합니다.
GetIDsOfNames 단일 멤버와 선택적 인수 이름 집합을 해당 정수 DISPID 집합에 매핑합니다. 이 ID는 Invoke에 대한 후속 호출에 사용할 수 있습니다.
GetIDsOfNames 멤버 이름과 멤버 ID, 매개 변수 이름과 매개 변수 ID를 매핑합니다.
GetImplTypeCustData 구현 형식의 사용자 지정 데이터를 가져옵니다.
GetImplTypeFlags 형식 설명에서 구현된 인터페이스 또는 기본 인터페이스 하나에 대한 IMPLTYPEFLAGS 열거형을 검색합니다.
GetLibAttr 라이브러리의 특성이 들어 있는 구조를 검색합니다.
GetLibStatistics 해시 테이블의 크기를 효율적으로 조정하는 데 필요한 형식 라이브러리에 대한 통계를 반환합니다.

GetMops 마샬링 정보를 검색합니다.
GetName 레코드 형식의 이름을 가져옵니다.
Getnames 지정된 멤버 ID 또는 속성 또는 메서드의 이름 및 지정된 함수 ID에 해당하는 매개 변수를 사용하여 변수를 검색합니다.
GetParamCustData 지정된 매개 변수의 사용자 지정 데이터를 가져옵니다.
GetRecordInfoFromGuids 형식 라이브러리를 로드하지 않고도 형식 정보의 GUID를 전달하여 UDT에 대한 IRecordInfo 인터페이스에 대한 포인터를 반환합니다.
GetRecordInfoFromTypeInfo 형식 정보를 전달하여 UDT의 IRecordInfo 인터페이스에 대한 포인터를 반환합니다.
GetRefTypeInfo 형식 설명이 다른 형식 설명을 참조하면 참조된 형식 설명을 검색합니다.
GetRefTypeOfImplType 형식 설명에서 COM 클래스를 설명하면 구현된 인터페이스 형식의 형식 설명을 검색합니다.
GetSize 레코드 인스턴스를 보유하는 데 필요한 메모리 바이트 수를 가져옵니다.
GetSource 오류를 발생시킨 클래스 또는 애플리케이션에 대한 언어 종속 프로그래밍 ID(ProgID)를 반환합니다.
GetTypeAttr 형식 설명의 특성을 포함하는 TYPEATTR 구조를 검색합니다.

GetTypeComp 형식 설명에 대한 ITypeComp 인터페이스를 검색하여 클라이언트 컴파일러가 형식 설명의 멤버에 바인딩할 수 있도록 합니다.
GetTypeComp 클라이언트 컴파일러가 라이브러리의 형식, 변수, 상수 및 전역 함수에 바인딩할 수 있도록 합니다.
GetTypeFlags 할당 없이 형식 플래그를 반환합니다. TYPEATTR(형식 특성)을 증가하지 않고 형식 플래그를 확장하는 플래그를 반환합니다.
GetTypeInfo 인터페이스의 형식 정보를 가져오는 데 사용할 수 있는 개체의 형식 정보를 검색합니다.
GetTypeInfo UDT 또는 UDT의 safearray를 설명하는 형식 정보를 검색합니다.
GetTypeInfo 라이브러리에서 지정된 형식 설명을 검색합니다.
GetTypeInfoCount 개체에서 제공하는 형식 정보 인터페이스의 수를 검색합니다(0 또는 1).
GetTypeInfoCount 형식 라이브러리에 있는 형식 설명의 수를 제공합니다.
GetTypeInfoOfGuid 지정된 된 GUID에 해당 하는 형식 설명을 검색 합니다.
GetTypeInfoType 형식 설명의 형식을 검색합니다.
GetTypeKind 할당을 수행하지 않고 TYPEKIND 열거형을 신속하게 반환합니다.
GetVarCustData 지정된 변수의 사용자 지정 데이터를 가져옵니다.

GetVarDesc 지정된 변수를 설명하는 VARDESC 구조를 검색합니다.
GetVarIndexOfMemId 알려진 DISPID를 기준으로 특정 멤버에 바인딩합니다. 여기서는 멤버 이름이 알려져 있지 않습니다(예: 기본 멤버에 바인딩할 경우). (ITypeInfo2.GetVarIndexOfMemId)
InterfaceSupportsErrorInfo 인터페이스가 IErrorInfo 인터페이스를 지원하는지 여부를 나타냅니다.
Invoke 개체에서 노출하는 메서드와 속성에 대한 액세스를 제공합니다.
Invoke 메서드를 호출하거나 개체의 속성에 액세스하여 형식 설명에서 설명한 인터페이스를 구현합니다.
IsMatchingType 전달된 레코드가 현재 레코드 정보의 레코드와 일치하는지 여부를 결정합니다.
IsName 라이브러리에 설명되어 있는 형식이나 멤버의 이름이 전달된 문자열에 들어 있는지 여부를 나타냅니다.
레이아웃 인스턴스별 데이터 멤버에 대한 가상 함수 및 인스턴스 오프셋에 대한 VTBL 오프셋을 할당하고 이중 인터페이스에 대한 두 가지 형식 설명을 만듭니다.
LHashValOfName 이름에 대한 해시 값을 계산합니다. (LHashValOfName)
LHashValOfNameSys 이름에 대한 해시 값을 계산합니다. (LHashValOfNameSys)
LHashValOfNameSysA 지정된 이름의 해시 값을 계산합니다.

LoadRegTypeLib 레지스트리 정보를 사용하여 형식 라이브러리를 로드합니다.
LoadTypeLib 형식 라이브러리를 로드하고 등록합니다.
LoadTypeLibEx 형식 라이브러리를 로드하고(필요에 따라) 시스템 레지스트리에 등록합니다. .
LPSAFEARRAY_UserFree RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다. (LPSAFEARRAY_UserFree)
LPSAFEARRAY_UserFree64 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다. (LPSAFEARRAY_UserFree64)
LPSAFEARRAY_UserMarshal 지정된 SAFEARRAY 개체의 데이터를 클라이언트 또는 서버 쪽의 사용자의 RPC 버퍼로 마샬링합니다. (LPSAFEARRAY_UserMarshal)
LPSAFEARRAY_UserMarshal64 지정된 SAFEARRAY 개체의 데이터를 클라이언트 또는 서버 쪽의 사용자의 RPC 버퍼로 마샬링합니다. (LPSAFEARRAY_UserMarshal64)
LPSAFEARRAY_UserSize SAFEARRAY 개체의 와이어 크기를 계산하고 핸들 및 데이터를 가져옵니다. (LPSAFEARRAY_UserSize)
LPSAFEARRAY_UserSize64 SAFEARRAY 개체의 와이어 크기를 계산하고 핸들 및 데이터를 가져옵니다. (LPSAFEARRAY_UserSize64)
LPSAFEARRAY_UserUnmarshal RPC 버퍼에서 SAFEARRAY 개체를 숨기지 않습니다. (LPSAFEARRAY_UserUnmarshal)
LPSAFEARRAY_UserUnmarshal64 RPC 버퍼에서 SAFEARRAY 개체를 숨기지 않습니다. (LPSAFEARRAY_UserUnmarshal64)

다음 열거형 시퀀스에서 지정된 항목을 검색합니다.
OaBuildVersion OLE Automation의 빌드 버전을 검색합니다.
OaEnablePerUserTLibRegistration RegisterTypeLib 함수가 Windows Vista SP1(서비스 팩 1), Windows Server 2008 이상 운영 체제 버전에서 기본 레지스트리 매핑을 재정의할 수 있도록 합니다.
OleLoadPictureFile 디스크의 그림 파일에서 IPictureDisp 개체를 만듭니다.
OleLoadPictureFileEx 파일에서 그림을 로드합니다.
OleSavePictureFile 그림을 파일에 저장합니다.
PutField 변형을 필드에 넣습니다.
PutFieldNoCopy 실제 데이터를 필드에 배치하여 할당된 필드에 데이터 소유권을 전달합니다.
QueryPathOfRegTypeLib 등록된 형식 라이브러리의 경로를 검색합니다.
RecordClear 레코드의 할당을 취소하지 않고 레코드의 개체 참조 및 기타 값을 해제합니다.
RecordCopy 전달된 버퍼에 기존 레코드를 복사합니다.
RecordCreate 새 레코드에 대한 메모리를 할당하고, 인스턴스를 초기화하고, 레코드에 대한 포인터를 반환합니다.

RecordCreateCopy 지정된 위치에 레코드 인스턴스의 복사본을 만듭니다.
RecordDestroy 리소스를 해제하고 레코드의 메모리 할당을 취소합니다.
RecordInit 레코드의 새 인스턴스를 초기화합니다.
RegisterActiveObject 개체를 해당 클래스의 활성 개체로 등록합니다.
RegisterTypeLib 형식 라이브러리에 대한 정보를 시스템 레지스트리에 추가합니다.
RegisterTypeLibForUser 호출하는 사용자가 사용할 형식 라이브러리를 등록합니다.
ReleaseFuncDesc 이전에 ITypeInfo::GetFuncDesc에서 반환한 FUNCDESC를 릴리스합니다.
ReleaseTLibAttr GetLibAttr에서 원래 가져온 TLIBATTR을 해제합니다.
ReleaseTypeAttr 이전에 ITypeInfo::GetTypeAttr에서 반환한 TYPEATTR을 해제합니다.
ReleaseVarDesc 이전에 ITypeInfo::GetVarDesc에서 반환한 VARDESC를 릴리스합니다.
RequestTypeChange 형식을 변경하라는 요청이 있을 때 발생합니다. 변경은 허용되지 않을 수 있습니다.
재설정 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumVARIANT.Reset)

RevokeActiveObject 개체의 상태를 활성 상태로 종료합니다.
SafeArrayAccessData 배열의 잠금 수를 증가시키고 배열 데이터에 대한 포인터를 검색합니다.
SafeArrayAddRef 지정된 안전 배열에 대한 설명자의 고정 참조 수를 1씩 늘리고, 안전 배열의 설명자에 의해 결정된 대로 해당 데이터가 동적으로 할당된 경우 지정된 안전 배열에 대한 데이터의 고정 참조 수를 1씩 늘릴 수 있습니다.
SafeArrayAllocData SafeArrayAllocDescriptor로 만든 설명자를 기반으로 안전한 배열에 대한 메모리를 할당합니다.
SafeArrayAllocDescriptor 안전한 배열 설명자에 대한 메모리를 할당합니다.
SafeArrayAllocDescriptorEx 배열 데이터를 할당하지 않고 VT_RECORD 포함하여 유효한 변형 형식의 배열에 대한 안전한 배열 설명자를 만듭니다.
SafeArrayCopy 기존 안전 배열의 복사본을 만듭니다.
SafeArrayCopyData 대상 배열의 리소스를 해제한 후 원본 배열을 지정된 대상 배열에 복사합니다.
SafeArrayCreate 새 배열 설명자를 만들고, 배열에 대한 데이터를 할당 및 초기화하고, 새 배열 설명자에 대한 포인터를 반환합니다.
SafeArrayCreateEx 지정된 VARTYPE, 차원 및 범위 수에서 안전한 배열 설명자를 만들고 반환합니다.
SafeArrayCreateVector 1차원 배열을 만듭니다. SafeArrayCreateVector를 사용하여 만든 안전한 배열은 고정 크기이므로 상수 FADF_FIXEDSIZE 항상 설정됩니다.

SafeArrayCreateVectorEx 지정된 VARTYPE 및 범위의 1차원 안전 배열을 만들고 반환합니다.
SafeArrayDestroy 기존 배열 설명자와 배열의 모든 데이터를 삭제합니다.
SafeArrayDestroyData 지정된 안전 배열의 모든 데이터를 삭제합니다.
SafeArrayDestroyDescriptor 지정된 안전 배열의 설명자를 삭제합니다.
SafeArrayGetDim 배열의 차수를 가져옵니다.
SafeArrayGetElement 배열의 단일 요소를 검색합니다.
SafeArrayGetElemsize 요소의 크기를 가져옵니다.
SafeArrayGetIID 지정된 안전 배열 내에 포함된 인터페이스의 GUID를 가져옵니다.
SafeArrayGetLBound 지정된 안전 배열의 모든 차원에 대한 하한을 가져옵니다.
SafeArrayGetRecordInfo 지정된 안전 배열에 포함된 UDT의 IRecordInfo 인터페이스를 검색합니다.
SafeArrayGetUBound 지정된 안전 배열의 모든 차원에 대한 상한을 가져옵니다.
SafeArrayGetVartype 지정된 안전 배열에 저장된 VARTYPE을 가져옵니다.

SafeArrayLock 배열의 잠금 수를 증가시키고 배열 데이터에 대한 포인터를 배열 설명자의 pvData에 배치합니다.
SafeArrayPtrOfIndex 배열 요소에 대한 포인터를 가져옵니다.
SafeArrayPutElement 데이터 요소를 배열의 지정된 위치에 저장합니다.
SafeArrayRedim 지정된 안전 배열의 가장 오른쪽(최적) 바인딩을 변경합니다.
SafeArrayReleaseData 지정된 안전 배열 데이터에 대한 고정 참조 수를 1씩 줄입니다. 이 개수가 0에 도달하면 해당 데이터의 메모리가 더 이상 해제되지 않습니다.
SafeArrayReleaseDescriptor 지정된 안전 배열의 설명자에 대한 고정 참조 수를 1씩 줄입니다. 해당 수가 0에 도달하면 해당 설명자에 대한 메모리가 더 이상 해제되지 않습니다.
SafeArraySetIID 지정된 안전 배열에 대한 인터페이스의 GUID를 설정합니다.
SafeArraySetRecordInfo 지정된 안전 배열의 레코드 정보를 설정합니다.
SafeArrayUnaccessData 배열의 잠금 수를 감소시키고 SafeArrayAccessData에서 검색한 포인터를 무효화합니다.
SafeArrayUnlock 배열을 해제하거나 크기를 조정할 수 있도록 배열의 잠금 수를 감소합니다.
SaveAllChanges 형식 정보의 레이아웃에 따라 ICreateTypeLib 인스턴스를 저장합니다.
SetAlignment TYPEKIND=TKIND_RECORD 항목의 데이터 맞춤을 지정합니다.

SetCustData 사용자 지정 데이터에 대한 값을 설정합니다.
SetCustData 값을 사용자 지정 데이터로 설정합니다.
SetDescription 오류에 대한 텍스트 설명을 설정합니다.
SetDocString 형식 브라우저에서 표시되는 설명서 문자열을 설정합니다.
SetDocString 라이브러리와 연결된 설명서 문자열을 설정합니다.
SetErrorInfo 실행의 현재 논리 스레드에 대한 오류 정보 개체를 설정합니다.
SetFuncAndParamNames 함수의 이름과 해당 매개 변수의 이름을 지정된 이름으로 설정합니다.
SetFuncCustData 지정된 함수에 대한 사용자 지정 데이터의 값을 설정합니다.
SetFuncDocString 지정된 인덱스로 함수의 설명서 문자열을 설정합니다.
SetFuncHelpContext 지정된 인덱스가 있는 함수의 도움말 컨텍스트 ID를 설정합니다.
SetFuncHelpStringContext 지정된 함수에 대한 도움말 컨텍스트 값을 설정합니다.
SetGuid 형식 설명과 연결된 GUID(Globally Unique Identifier)를 설정합니다.

SetGuid 형식 라이브러리와 연결된 UUID(유니버설 고유 식별자)를 설정합니다.
SetGUID 오류를 정의한 인터페이스의 GUID(Globally Unique Identifier)를 설정합니다.
SetHelpContext 오류에 대한 도움말 컨텍스트 식별자(ID)를 설정합니다.
SetHelpContext 형식 정보의 도움말 컨텍스트 ID를 설정합니다.
SetHelpContext 형식 라이브러리에 대한 일반 도움말 정보를 검색하기 위한 도움말 컨텍스트 ID를 설정합니다.
SetHelpFile 오류를 설명하는 도움말 파일의 경로를 설정합니다.
SetHelpFileName 도움말 파일의 이름을 설정합니다.
SetHelpStringContext 지정된 도움말 문자열의 컨텍스트 번호를 설정합니다.
SetHelpStringContext 도움말 문자열 컨텍스트 번호를 설정합니다.
SetHelpStringDll 도움말 문자열 조회에 사용할 DLL 이름을 설정합니다(지역화를 위해).
SetImplTypeCustData 지정된 구현 형식에 대한 사용자 지정 데이터의 값을 설정합니다.
SetImplTypeFlags 형식의 구현되거나 상속된 인터페이스에 대한 특성을 설정합니다.

SetLcid 라이브러리와 연결된 이진 Microsoft 국가 언어 ID를 설정합니다.
SetLibFlags 라이브러리 플래그를 설정합니다.
SetMops 형식 설명 또는 함수와 연결된 마샬링 opcode 문자열을 설정합니다.
SetName typeinfo의 이름을 설정합니다.
SetName 형식 라이브러리의 이름을 설정합니다.
SetParamCustData 지정된 매개 변수에 대한 사용자 지정 데이터의 값을 설정합니다.
Setsource 오류를 발생시킨 클래스 또는 애플리케이션에 대한 ProgID(언어 종속 프로그래밍 식별자)를 설정합니다.
SetTypeDescAlias TYPEKIND=TKIND_ALIAS 경우 이 형식 설명이 별칭인 형식 설명을 설정합니다.
SetTypeFlags 생성되는 형식 설명의 형식 플래그를 설정합니다.
SetVarCustData 지정된 변수에 대한 사용자 지정 데이터의 값을 설정합니다.
SetVarDocString 지정된 인덱스로 변수에 대한 설명서 문자열을 설정합니다.
SetVarHelpContext 지정된 인덱스가 있는 변수의 도움말 컨텍스트 ID를 설정합니다.

SetVarHelpStringContext 지정된 변수에 대한 도움말 컨텍스트 값을 설정합니다.
SetVarName 변수의 이름을 설정합니다.
SetVersion 형식 정보의 주 버전 및 부 버전 번호를 설정합니다.
SetVersion 형식 라이브러리의 주 버전 및 부 버전 번호를 설정합니다.
skip 열거형 시퀀스의 다음 셀트 요소를 건너뛰려고 시도합니다.
SysAddRefString 지정된 문자열에 대한 고정 참조 수를 1씩 늘입니다.
SysAllocString 새 문자열을 할당하고 전달된 문자열을 복사합니다.
SysAllocStringByteLen ANSI 문자열을 입력으로 사용하고 ANSI 문자열을 포함하는 BSTR을 반환합니다. ANSI-유니코드 변환을 수행하지 않습니다.
SysAllocStringLen 새 문자열을 할당하고, 전달된 문자열에서 지정된 수의 문자를 복사하고, null 종료 문자를 추가합니다.
SysFreeString SysAllocString, SysAllocStringByteLen, SysReAllocString, SysAllocStringLen 또는 SysReAllocStringLen에서 이전에 할당한 문자열의 할당을 취소합니다.
SysReAllocString 이전에 할당된 문자열을 두 번째 문자열의 크기로 다시 할당하고 두 번째 문자열을 다시 할당된 메모리에 복사합니다.

SysReAllocStringLen 이전 BSTR에서 지정된 수의 문자를 포함하는 새 BSTR을 만들고 이전 BSTR을 해제합니다.
SysReleaseString 지정된 문자열에 대한 고정 참조 수를 1씩 줄입니다. 이 개수가 0에 도달하면 해당 문자열의 메모리가 더 이상 해제되지 않습니다.
SysStringByteLen BSTR의 길이(바이트)를 반환합니다.
SysStringLen BSTR의 길이를 반환합니다.
SystemTimeToVariantTime 시스템 시간을 변형 표현으로 변환합니다.
UnRegisterTypeLib 시스템 레지스트리에서 형식 라이브러리 정보를 제거합니다. 이 API를 사용하여 애플리케이션이 자신을 제대로 제거할 수 있도록 합니다.
UnRegisterTypeLibForUser RegisterTypeLibForUser를 사용하여 등록된 형식 라이브러리 정보를 제거합니다.
VarAbs 변형의 절대값을 반환합니다.
VarAdd 두 변형의 합계를 반환합니다.
VarAnd 모든 정수 계열 형식의 두 변형 간에 비트 및 연산을 수행합니다.
VarBoolFromCy 통화 값을 부울 값으로 변환합니다.
VarBoolFromDate 날짜 값을 부울 값으로 변환합니다.

VarBoolFromDec 10진수 값을 부울 값으로 변환합니다.
VarBoolFromDisp IDispatch 인스턴스의 기본 속성을 부울 값으로 변환합니다.
VarBoolFromI1 char 값을 부울 값으로 변환합니다.
VarBoolFromI2 짧은 값을 부울 값으로 변환합니다.
VarBoolFromI4 긴 값을 부울 값으로 변환합니다.
VarBoolFromI8 8 바이트 정수 값을 부울 값으로 변환합니다.
VarBoolFromR4 부울 값을 부울 값으로 변환합니다.
VarBoolFromR8 Double 값을 부울 값으로 변환합니다.
VarBoolFromStr OLECHAR 문자열을 부울 값으로 변환합니다.
VarBoolFromUI1 부호 없는 문자 값을 부울 값으로 변환합니다.
VarBoolFromUI2 부호 없는 짧은 값을 부울 값으로 변환합니다.
VarBoolFromUI4 부호 없는 long 값을 부울 값으로 변환합니다.

VarBoolFromUI8 부울 값으로 8 바이트 부호 없는 정수 값을 변환합니다.
VarBstrCat BSTR 형식의 두 변형을 연결하고 결과 BSTR을 반환합니다.
VarBstrCmp BSTR 형식의 두 변형을 비교합니다.
VarBstrFromBool 부울 값을 BSTR 값으로 변환합니다.
VarBstrFromCy 통화 값을 BSTR 값으로 변환합니다.
VarBstrFromDate 날짜 값을 BSTR 값으로 변환합니다.
VarBstrFromDec 10진수 값을 BSTR 값으로 변환합니다.
VarBstrFromDisp IDispatch 인스턴스의 기본 속성을 BSTR 값으로 변환합니다.
VarBstrFromI1 char 값을 BSTR 값으로 변환합니다.
VarBstrFromI2 짧은 값을 BSTR 값으로 변환합니다.
VarBstrFromI4 긴 값을 BSTR 값으로 변환합니다.
VarBstrFromI8 부호 없는 8 바이트 정수 값을 BSTR 값으로 변환합니다. (VarBstrFromI8)

VarBstrFromR4 float 값을 BSTR 값으로 변환합니다.
VarBstrFromR8 DOUBLE 값을 BSTR 값으로 변환합니다.
VarBstrFromUI1 부호 없는 문자 값을 BSTR 값으로 변환합니다.
VarBstrFromUI2 부호 없는 짧은 값을 BSTR 값으로 변환합니다.
VarBstrFromUI4 부호 없는 long 값을 BSTR 값으로 변환합니다.
VarBstrFromUI8 부호 없는 8 바이트 정수 값을 BSTR 값으로 변환합니다. (VarBstrFromUI8)
VarCat 두 변형을 결합하고 결과를 반환합니다.
VarCmp 두 가지 변형을 비교합니다.
VarCyAbs 형식 통화 변형의 절대값을 검색합니다.
VarCyAdd 형식 통화의 두 가지 변형을 추가합니다.
VarCyCmp 형식 통화의 두 가지 변형을 비교합니다.
VarCyCmpR8 형식 통화의 변형을 double 형식 값과 비교합니다.

VarCyFix 형식 통화 변형의 정수 부분을 검색합니다. (VarCyFix)
VarCyFromBool 부울 값을 통화 값으로 변환합니다.
VarCyFromDate 날짜 값을 통화 값으로 변환합니다.
VarCyFromDec 10진수 값을 통화 값으로 변환합니다.
VarCyFromDisp IDispatch 인스턴스의 기본 속성을 통화 값으로 변환합니다.
VarCyFromI1 char 값을 통화 값으로 변환합니다.
VarCyFromI2 짧은 값을 통화 값으로 변환합니다.
VarCyFromI4 long 값을 통화 값으로 변환합니다.
VarCyFromI8 8 바이트 정수 값을 통화 값으로 변환합니다.
VarCyFromR4 float 값을 통화 값으로 변환합니다.
VarCyFromR8 double 값을 통화 값으로 변환합니다.
VarCyFromStr OLECHAR 문자열을 통화 값으로 변환합니다.

VarCyFromUI1 부호 없는 문자 값을 통화 값으로 변환합니다.
VarCyFromUI2 부호 없는 짧은 값을 통화 값으로 변환합니다.
VarCyFromUI4 부호 없는 long 값을 통화 값으로 변환합니다.
VarCyFromUI8 부호 없는 8 바이트 정수 값을 통화 값으로 변환합니다.
VarCyInt 형식 통화 변형의 정수 부분을 검색합니다. (VarCyInt)
VarCyMul 형식 통화의 두 가지 변형을 곱합니다.
VarCyMulI4 통화 값을 32비트 정수로 곱합니다.
VarCyMulI8 통화 값을 64비트 정수로 곱합니다.
VarCyNeg 형식 통화의 변형에 대해 논리적 부정을 수행합니다.
VarCyRound 형식 통화의 변형을 지정된 소수 자릿수로 반올림합니다.
VarCySub 형식 통화의 두 가지 변형을 뺍니다.
VarDateFromBool 부울 값을 날짜 값으로 변환합니다.

VarDateFromCy 통화 값을 날짜 값으로 변환합니다.
VarDateFromDec 10진수 값을 날짜 값으로 변환합니다.
VarDateFromDisp IDispatch 인스턴스의 기본 속성을 날짜 값으로 변환합니다.
VarDateFromI1 char 값을 날짜 값으로 변환합니다.
VarDateFromI2 짧은 값을 날짜 값으로 변환합니다.
VarDateFromI4 long 값을 날짜 값으로 변환합니다.
VarDateFromI8 부호 없는 8 바이트 정수 값을 날짜 값으로 변환합니다.
VarDateFromR4 float 값을 날짜 값으로 변환합니다.
VarDateFromR8 double 값을 날짜 값으로 변환합니다.
VarDateFromStr OLECHAR 문자열을 날짜 값으로 변환합니다.
VarDateFromUdate MS-DOS 형식에서 변형 형식으로 변환된 시간과 날짜를 변환합니다. (VarDateFromUdate)
VarDateFromUdateEx MS-DOS 형식에서 변형 형식으로 변환된 시간과 날짜를 변환합니다. (VarDateFromUdateEx)

VarDateFromUI1 부호 없는 문자 값을 날짜 값으로 변환합니다.
VarDateFromUI2 부호 없는 짧은 값을 날짜 값으로 변환합니다.
VarDateFromUI4 부호 없는 long 값을 날짜 값으로 변환합니다.
VarDateFromUI8 부호 없는 8 바이트 값을 날짜 값으로 변환합니다.
VarDecAbs 10진수 형식 변형의 절대값을 검색합니다.
VarDecAdd 10진수 형식의 두 변형을 추가합니다.
VarDecCmp 10진수 형식의 두 변형을 비교합니다.
VarDecCmpR8 10진수 형식의 변형을 double 형식의 값과 비교합니다.
VarDecDiv 10진수 형식의 두 변형을 나눕니다.
VarDecFix 10진수 형식 변형의 정수 부분을 검색합니다. (VarDecFix)
VarDecFromBool 부울 값을 10진수 값으로 변환합니다.
VarDecFromCy 통화 값을 10진수 값으로 변환합니다.

VarDecFromDate 날짜 값을 10진수 값으로 변환합니다.
VarDecFromDisp IDispatch 인스턴스의 기본 속성을 10진수 값으로 변환합니다.
VarDecFromI1 char 값을 10진수 값으로 변환합니다.
VarDecFromI2 짧은 값을 10진수 값으로 변환합니다.
VarDecFromI4 긴 값을 10진수 값으로 변환합니다.
VarDecFromI8 8 바이트 정수 값을 10진수 값으로 변환합니다.
VarDecFromR4 부동 소수점 값을 10진수 값으로 변환합니다.
VarDecFromR8 이중 값을 10진수 값으로 변환합니다.
VarDecFromStr OLECHAR 문자열을 10진수 값으로 변환합니다.
VarDecFromUI1 부호 없는 문자 값을 10진수 값으로 변환합니다.
VarDecFromUI2 부호 없는 짧은 값을 10진수 값으로 변환합니다.
VarDecFromUI4 부호 없는 long 값을 10진수 값으로 변환합니다.

VarDecFromUI8 부호 없는 8 바이트 정수 값을 10진수 값으로 변환합니다.
VarDecInt 10진수 형식 변형의 정수 부분을 검색합니다. (VarDecInt)
VarDecMul 10진수 형식의 두 변형을 곱합니다.
VarDecNeg 10진수 형식의 변형에 대해 논리적 부정을 수행합니다.
VarDecRound 10진수 형식의 변형을 지정된 소수 자릿수로 반올림합니다.
VarDecSub 10진수 형식의 두 변형을 뺍니다.
VarDiv 두 변형을 나눈 결과를 반환합니다.
VarEqv 두 변형에 대해 비트 동등성을 수행합니다.
VarFix 변형의 정수 부분을 반환합니다. (VarFix)
VarFormat 형식 문자열을 구문 분석하여 변형 형식을 문자열 형식으로 지정합니다.
VarFormatCurrency 통화 값이 포함된 변형의 형식을 문자열 형식으로 지정합니다.
VarFormatDateTime 명명된 날짜 및 시간 정보를 포함하는 변형의 형식을 문자열로 지정합니다.

VarFormatFromTokens 토큰화된 형식 문자열을 가져와 변형에 적용하여 서식이 지정된 출력 문자열을 생성합니다.
VarFormatNumber 숫자를 포함하는 변형의 형식을 문자열 형식으로 지정합니다.
VarFormatPercent 백분율을 포함하는 변형의 형식을 문자열 형식으로 지정합니다.
VarI1FromBool 부울 값을 char 값으로 변환합니다.
VarI1FromCy 통화 값을 char 값으로 변환합니다.
VarI1FromDate 날짜 값을 char 값으로 변환합니다.
VarI1FromDec 10진수 값을 char 값으로 변환합니다.
VarI1FromDisp IDispatch 인스턴스의 기본 속성을 char 값으로 변환합니다.
VarI1FromI2 짧은 값을 char 값으로 변환합니다.
VarI1FromI4 긴 값을 char 값으로 변환합니다.
VarI1FromI8 8 바이트 정수 값을 char 값으로 변환합니다.
VarI1FromR4 float 값을 char 값으로 변환합니다.

VarI1FromR8 double 값을 char 값으로 변환합니다.
VarI1FromStr OLECHAR 문자열을 char 값으로 변환합니다.
VarI1FromUI1 부호 없는 char 값을 char 값으로 변환합니다.
VarI1FromUI2 부호 없는 짧은 값을 char 값으로 변환합니다.
VarI1FromUI4 부호 없는 long 값을 char 값으로 변환합니다.
VarI1FromUI8 부호 없는 8 바이트 정수 값을 char 값으로 변환합니다.
VarI2FromBool 부울 값을 짧은 값으로 변환합니다.
VarI2FromCy 통화 값을 짧은 값으로 변환합니다.
VarI2FromDate 날짜 값을 짧은 값으로 변환합니다.
VarI2FromDec 10진수 값을 짧은 값으로 변환합니다.
VarI2FromDisp IDispatch 인스턴스의 기본 속성을 짧은 값으로 변환합니다.
VarI2FromI1 문자 값을 짧은 값으로 변환합니다.

VarI2FromI4 긴 값을 짧은 값으로 변환합니다.
VarI2FromI8 8 바이트 정수 값을 짧은 값으로 변환합니다.
VarI2FromR4 부동 소수자 값을 짧은 값으로 변환합니다.
VarI2FromR8 double 값을 짧은 값으로 변환합니다.
VarI2FromStr OLECHAR 문자열을 짧은 값으로 변환합니다.
VarI2FromUI1 부호 없는 문자 값을 짧은 값으로 변환합니다.
VarI2FromUI2 부호 없는 짧은 값을 짧은 값으로 변환합니다.
VarI2FromUI4 부호 없는 long 값을 짧은 값으로 변환합니다.
VarI2FromUI8 부호 없는 8 바이트 정수 값을 짧은 값으로 변환합니다.
VarI4FromBool 부울 값을 긴 값으로 변환합니다.
VarI4FromCy 통화 값을 긴 값으로 변환합니다.
VarI4FromDate 날짜 값을 긴 값으로 변환합니다.

VarI4FromDec 10진수 값을 긴 값으로 변환합니다.
VarI4FromDisp IDispatch 인스턴스의 기본 속성을 긴 값으로 변환합니다.
VarI4FromI1 char 값을 긴 값으로 변환합니다.
VarI4FromI2 짧은 값을 긴 값으로 변환합니다.
VarI4FromI4 int 값을 long 값으로 변환합니다.
VarI4FromI8 8 바이트 정수 값을 긴 값으로 변환합니다.
VarI4FromR4 부동 소수자 값을 긴 값으로 변환합니다.
VarI4FromR8 double 값을 긴 값으로 변환합니다.
VarI4FromStr OLECHAR 문자열을 긴 값으로 변환합니다.
VarI4FromUI1 부호 없는 문자 값을 긴 값으로 변환합니다.
VarI4FromUI2 부호 없는 짧은 값을 long 값으로 변환합니다.
VarI4FromUI4 부호 없는 long 값을 long 값으로 변환합니다.

VarI4FromUI8 부호 없는 8 바이트 정수 값을 long으로 변환합니다.
VarI8FromBool 부울 값을 8 바이트 정수 값으로 변환합니다.
VarI8FromCy 통화 값을 8 바이트 정수 값으로 변환합니다.
VarI8FromDate 날짜 값을 8 바이트 정수 값으로 변환합니다.
VarI8FromDec 10진수 값을 8 바이트 정수 값으로 변환합니다.
VarI8FromDisp IDispatch 인스턴스의 기본 속성을 8 바이트 정수 값으로 변환합니다.
VarI8FromI1 char 값을 8 바이트 정수 값으로 변환합니다.
VarI8FromI2 짧은 값을 8 바이트 정수 값으로 변환합니다.
VarI8FromR4 부동 소수점 값을 8 바이트 정수 값으로 변환합니다.
VarI8FromR8 이중 값을 8 바이트 정수 값으로 변환합니다.
VarI8FromStr OLECHAR 문자열을 8 바이트 정수 값으로 변환합니다.
VarI8FromUI1 부호 없는 바이트 값을 8 바이트 정수 값으로 반전합니다.

VarI8FromUI2 부호 없는 짧은 값을 8 바이트 정수 값으로 변환합니다.
VarI8FromUI4 부호 없는 long 값을 8 바이트 정수 값으로 변환합니다.
VarI8FromUI8 부호 없는 8 바이트 정수 값을 8 바이트 정수 값으로 변환합니다.
VARIANT_UserFree RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다. (VARIANT_UserFree)
VARIANT_UserFree64 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다. (VARIANT_UserFree64)
VARIANT_UserMarshal VARIANT 개체를 RPC 버퍼로 마샬링합니다. (VARIANT_UserMarshal)
VARIANT_UserMarshal64 VARIANT 개체를 RPC 버퍼로 마샬링합니다. (VARIANT_UserMarshal64)
VARIANT_UserSize VARIANT 개체의 와이어 크기를 계산하고 핸들 및 데이터를 가져옵니다. (VARIANT_UserSize)
VARIANT_UserSize64 VARIANT 개체의 와이어 크기를 계산하고 핸들 및 데이터를 가져옵니다. (VARIANT_UserSize64)
VARIANT_UserUnmarshal RPC 버퍼에서 VARIANT 개체를 숨기지 않습니다. (VARIANT_UserUnmarshal)
VARIANT_UserUnmarshal64 RPC 버퍼에서 VARIANT 개체를 숨기지 않습니다. (VARIANT_UserUnmarshal64)
VariantChangeType 변형을 한 형식에서 다른 형식으로 변환합니다.

VariantChangeTypeEx LCID를 사용하여 변형을 한 형식에서 다른 형식으로 변환합니다.
VariantClear 변형을 지웁니다.
VariantCopy 대상 변형을 해제하고 원본 변형의 복사본을 만듭니다.
VariantCopyInd 대상 변형을 해제하고 원본 변형의 복사본을 만들어 원본이 VT_BYREF 지정되는 경우 필요한 간접 참조를 수행합니다.
VariantInit 변형을 초기화합니다.
VariantTimeToDosDateTime 날짜 및 시간의 변형 표현을 MS-DOS 날짜 및 시간 값으로 변환합니다.
VariantTimeToSystemTime 시간의 변형 표현을 시스템 시간 값으로 변환합니다.
VarIdiv 모든 형식의 두 변형을 정수로 변환한 다음 결과를 나눈 결과를 반환합니다.
VarImp 두 변형에 대해 비트 의미를 수행합니다.
VarInt 변형의 정수 부분을 반환합니다. (VarInt)
VarMod 두 변형을 나누고 나머지만 반환합니다.
VarMonthName 지역화된 월 이름을 포함하는 문자열을 반환합니다.

VarMul 두 변형을 곱한 결과를 반환합니다.
VarNeg 변형에 대해 논리적 부정을 수행합니다.
VarNot 변형에서 부정이 아닌 비트 연산을 수행합니다.
VarNumFromParseNum 구문 분석된 결과를 변형으로 변환합니다.
VarOr 두 변형에 대해 논리적 분리를 수행합니다.
VarParseNumFromStr 문자열을 구문 분석하고, 문자열이 나타내는 숫자에 대한 형식 독립적 설명을 만듭니다.
VarPow 두 가지 변형으로 전원 함수를 수행한 결과를 반환합니다.
VarR4CmpR8 float 및 double 형식의 두 변형을 비교합니다.
VarR4FromBool 부울 값을 float 값으로 변환합니다.
VarR4FromCy 통화 값을 부동 소수자 값으로 변환합니다.
VarR4FromDate 날짜 값을 float 값으로 변환합니다.
VarR4FromDec 10진수 값을 부동 소수점 값으로 변환합니다.

VarR4FromDisp IDispatch 인스턴스의 기본 속성을 float 값으로 변환합니다.
VarR4FromI1 char 값을 float 값으로 반전합니다.
VarR4FromI2 짧은 값을 부동 소수자 값으로 변환합니다.
VarR4FromI4 긴 값을 float 값으로 변환합니다.
VarR4FromI8 8 바이트 정수 값을 부동 소수점 값으로 변환합니다.
VarR4FromR8 double 값을 float 값으로 변환합니다.
VarR4FromStr OLECHAR 문자열을 float 값으로 변환합니다.
VarR4FromUI1 부호 없는 char 값을 float 값으로 변환합니다.
VarR4FromUI2 부호 없는 짧은 값을 부동 소수자 값으로 변환합니다.
VarR4FromUI4 부호 없는 long 값을 float 값으로 변환합니다.
VarR4FromUI8 부호 없는 8 바이트 정수 값을 부동 소수점 값으로 변환합니다.
VarR8FromBool 부울 값을 double 값으로 변환합니다.

VarR8FromCy 통화 값을 double 값으로 변환합니다.
VarR8FromDate 날짜 값을 double 값으로 변환합니다.
VarR8FromDec 10진수 값을 double 값으로 변환합니다.
VarR8FromDisp IDispatch 인스턴스의 기본 속성을 double 값으로 변환합니다.
VarR8FromI1 char 값을 double 값으로 변환합니다.
VarR8FromI2 짧은 값을 double 값으로 변환합니다.
VarR8FromI4 long 값을 double 값으로 변환합니다.
VarR8FromI8 8 바이트 정수 값을 double 값으로 변환합니다.
VarR8FromR4 float 값을 double 값으로 변환합니다.
VarR8FromStr OLECHAR 문자열을 double 값으로 변환합니다.
VarR8FromUI1 부호 없는 문자 값을 double 값으로 변환합니다.
VarR8FromUI2 부호 없는 짧은 값을 double 값으로 변환합니다.

VarR8FromUI4 부호 없는 long 값을 double 값으로 변환합니다.
VarR8FromUI8 부호 없는 8 바이트 정수 값을 double 값으로 변환합니다.
VarR8Pow double 형식의 변형에 대해 전원 함수를 수행합니다.
VarR8Round 형식의 변형을 지정된 소수 자릿수로 반올림합니다.
VarRound 변형을 지정된 소수 자릿수로 반올림합니다.
VarSub 두 가지 변형을 뺍니다.
VarTokenizeFormatString 실제 형식 문자열을 VarFormatFromTokens를 사용하여 변형의 서식을 지정하는 데 사용할 수 있는 일련의 토큰으로 구문 분석합니다.
VarUdateFromDate 변형 형식에서 MS-DOS 형식으로 변환된 시간과 날짜를 변환합니다.
VarUI1FromBool 부울 값을 부호 없는 문자 값으로 변환합니다.
VarUI1FromCy 통화 값을 부호 없는 문자 값으로 변환합니다.
VarUI1FromDate 날짜 값을 부호 없는 문자 값으로 변환합니다.
VarUI1FromDec 10진수 값을 부호 없는 문자 값으로 변환합니다.

VarUI1FromDisp IDispatch 인스턴스의 기본 속성을 부호 없는 문자 값으로 변환합니다.
VarUI1FromI1 char 값을 부호 없는 문자 값으로 변환합니다.
VarUI1FromI2 짧은 값을 부호 없는 문자 값으로 변환합니다.
VarUI1FromI4 long 값을 부호 없는 문자 값으로 변환합니다.
VarUI1FromI8 8 바이트 정수 값을 바이트 값으로 변환합니다.
VarUI1FromR4 부동 소수자 값을 부호 없는 문자 값으로 변환합니다.
VarUI1FromR8 double 값을 부호 없는 문자 값으로 변환합니다.
VarUI1FromStr OLECHAR 문자열을 부호 없는 문자 문자열로 변환합니다.
VarUI1FromUI2 부호 없는 짧은 값을 부호 없는 문자 값으로 변환합니다.
VarUI1FromUI4 부호 없는 long 값을 부호 없는 char 값으로 변환합니다.
VarUI1FromUI8 부호 없는 8 바이트 정수 값을 바이트 값으로 변환합니다.
VarUI2FromBool 부울 값을 부호 없는 짧은 값으로 변환합니다.

VarUI2FromCy 통화 값을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromDate 날짜 값을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromDec 10진수 값을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromDisp IDispatch 인스턴스의 기본 속성을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromI1 char 값을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromI2 짧은 값을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromI4 long 값을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromI8 8 바이트 정수 값을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromR4 부동 소수자 값을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromR8 double 값을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromStr OLECHAR 문자열을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromUI1 부호 없는 문자 값을 부호 없는 짧은 값으로 변환합니다.

VarUI2FromUI4 부호 없는 long 값을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromUI8 부호 없는 8 바이트 정수 값을 부호 없는 짧은 값으로 변환합니다.
VarUI4FromBool 부울 값을 부호 없는 long 값으로 변환합니다.
VarUI4FromCy 통화 값을 부호 없는 long 값으로 변환합니다.
VarUI4FromDate 날짜 값을 부호 없는 long 값으로 변환합니다.
VarUI4FromDec 10진수 값을 부호 없는 long 값으로 변환합니다.
VarUI4FromDisp IDispatch 인스턴스의 기본 속성을 부호 없는 long 값으로 변환합니다.
VarUI4FromI1 char 값을 부호 없는 long 값으로 변환합니다.
VarUI4FromI2 짧은 값을 부호 없는 long 값으로 변환합니다.
VarUI4FromI4 긴 값을 부호 없는 long 값으로 변환합니다.
VarUI4FromI8 8 바이트 정수 값을 부호 없는 long 값으로 변환합니다.
VarUI4FromR4 부동 소수자 값을 부호 없는 long 값으로 변환합니다.

VarUI4FromR8 이중 값을 부호 없는 long 값으로 변환합니다.
VarUI4FromStr OLECHAR 문자열을 부호 없는 long 값으로 변환합니다.
VarUI4FromUI1 부호 없는 char 값을 부호 없는 long 값으로 변환합니다.
VarUI4FromUI2 부호 없는 짧은 값을 부호 없는 long 값으로 변환합니다.
VarUI4FromUI8 부호 없는 8 바이트 정수 값을 부호 없는 long 값으로 변환합니다.
VarUI8FromBool VARIANT_BOOL 값을 부호 없는 8 바이트 정수 값으로 변환합니다.
VarUI8FromCy 통화 값을 부호 없는 8 바이트 정수 값으로 변환합니다.
VarUI8FromDate 날짜 값을 부호 없는 8 바이트 정수 값으로 변환합니다.
VarUI8FromDec 10진수 값을 부호 없는 8 바이트 정수 값으로 변환합니다.
VarUI8FromDisp IDispatch 인스턴스의 기본 속성을 부호 없는 8 바이트 정수 값으로 변환합니다.
VarUI8FromI1 char 값을 부호 없는 8 바이트 정수 값으로 변환합니다.
VarUI8FromI2 짧은 값을 부호 없는 8 바이트 정수 값으로 변환합니다.

VarUI8FromI8 8 바이트 정수 값을 부호 없는 8 바이트 정수 값으로 변환합니다.
VarUI8FromR4 부동 소수점 값을 부호 없는 8 바이트 정수 값으로 변환합니다.
VarUI8FromR8 이중 값을 부호 없는 8 바이트 정수 값으로 변환합니다.
VarUI8FromStr OLECHAR 문자열을 부호 없는 8 바이트 정수 값으로 변환합니다.
VarUI8FromUI1 바이트 값을 부호 없는 8 바이트 정수 값으로 변환합니다.
VarUI8FromUI2 부호 없는 짧은 값을 부호 없는 8 바이트 정수 값으로 변환합니다.
VarUI8FromUI4 부호 없는 long 값을 부호 없는 8 바이트 정수 값으로 변환합니다.
VarWeekdayName 평일의 지역화된 이름을 포함하는 문자열을 반환합니다.
VarXor 두 변형에 대해 논리적 제외를 수행합니다.
VectorFromBstr BSTR의 각 문자를 벡터의 요소에 할당하는 벡터를 반환합니다.

인터페이스

ICreateErrorInfo 오류 정보를 반환합니다.

ICreateTypeInfo 형식 설명을 통해 정의된 형식 정보를 만들고 관리하기 위한 도구를 제공합니다. (ICreateTypeInfo)
ICreateTypeInfo2 형식 설명을 통해 정의된 형식 정보를 만들고 관리하기 위한 도구를 제공합니다. (ICreateTypeInfo2)
ICreateTypeLib 형식 정보를 포함하는 구성 요소 또는 파일을 만들고 관리하는 방법을 제공합니다. (ICreateTypeLib)
ICreateTypeLib2 형식 정보를 포함하는 구성 요소 또는 파일을 만들고 관리하는 방법을 제공합니다. (ICreateTypeLib2)
IDispatch 자동화를 지원하는 프로그래밍 도구 및 기타 애플리케이션에 개체, 메서드 및 속성을 노출합니다.
IEnumVARIANT 다른 유형의 개체 및 내장 형식 컬렉션을 포함하여 변형 컬렉션을 열거하는 메서드를 제공합니다.
IErrorInfo 자세한 상황별 오류 정보를 제공합니다.
IRecordInfo 특정 UDT의 구조를 설명합니다.
ISupportErrorInfo 오류 정보가 호출 체인을 올바르게 전파할 수 있는지 확인합니다. 오류 처리 인터페이스를 사용하는 자동화 개체는 ISupportErrorInfo를 구현해야 합니다.
ITypeChangeEvents 클라이언트가 ITypeInfo, ITypeInfo2, ICreateTypeInfo 및 ICreateTypeInfo2 인터페이스를 구현하는 개체에 대한 형식 변경 알림을 구독할 수 있도록 합니다.
Itypecomp ITypeComp 인터페이스는 구조체와 인터페이스에 바인딩하고 인스턴스화할 때 컴파일러가 필요로 하는 정보에 빠르게 액세스할 수 있는 방법을 제공합니다.

Itypeinfo 개체에 대한 정보를 읽는 데 사용됩니다. (ITypeInfo)
ITypeInfo2 개체에 대한 정보를 읽는 데 사용됩니다. (ITypeInfo2)
ITypeLib 개체 집합을 설명하는 데이터인 형식 라이브러리를 나타냅니다. (ITypeLib)
ITypeLib2 개체 집합을 설명하는 데이터인 형식 라이브러리를 나타냅니다. (ITypeLib2)

구조체

ARRAYDESC 배열, 해당 요소 형식 및 해당 차원에 대해 설명합니다.
BINDPTR 포인터를 설명합니다.
CUSTDATA 사용자 지정 데이터를 나타냅니다.
CUSTDATAITEM 사용자 지정 데이터 항목을 나타냅니다.
DISPPARAMS 메서드 또는 속성에 전달된 인수를 포함합니다.
EXCEPINFO IDispatch::Invoke 중에 발생한 예외에 대해 설명합니다.
FUNCDESC 함수에 대해 설명합니다. (FUNCDESC)

INTERFACEDATA 개체의 속성 및 메서드에 대해 설명합니다.
METHODDATA 메서드 또는 속성에 대해 설명합니다.
NUMPARSE 숫자 구문 분석 정보를 지정합니다.
PARAMDATA 메서드 또는 속성에서 허용하는 매개 변수에 대해 설명합니다.
PARAMDESC 구조체 요소, 매개 변수, 프로세스간 함수 반환 값을 전달하는데 필요한 정보가 들어 있습니다.
PARAMDESCEX 매개 변수의 기본값에 대한 정보를 포함합니다.
Safearray 안전한 배열을 나타냅니다.
SAFEARRAYBOUND 배열의 한 차원 범위를 나타냅니다.
TLIBATTR 형식 라이브러리에 대한 정보를 포함합니다. 이 구조의 정보는 형식 라이브러리를 식별하고 멤버 이름에 대한 국가별 언어 지원을 제공하는 데 사용됩니다.
TYPEATTR 형식의 특성을 포함합니다.
TYPEDESC 변수의 형식, 함수의 반환 형식 또는 함수 매개 변수의 형식에 대해 설명합니다.
VARDESC 변수, 상수 또는 데이터 멤버를 설명합니다.

변형

VARIANTARG는 DISPPARAMS 내에서 전달된 인수와 참조로 전달할 수 없는 변형 데이터를 지정하는 VARIANT에 대해 설명합니다.

BITS(Background Intelligent Transfer Service)

아티클2023. 03. 16.

Background Intelligent Transfer Service 기술에 대한 개요입니다.

Background Intelligent Transfer Service를 개발하려면 다음 헤더가 필요합니다.

- bits.h
- bits10_1.h
- bits10_2.h
- bits10_3.h
- bits1_5.h
- bits2_0.h
- bits2_5.h
- bits3_0.h
- bits4_0.h
- bits5_0.h
- bitscfg.h
- qmgr.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- BITS(Background Intelligent Transfer Service)

열거형

BG_AUTH_SCHEME 프록시 또는 서버가 사용자 인증을 요청할 때 사용할 인증 체계를 지정하는 상수를 정의합니다.
BG_AUTH_TARGET 자격 증명이 프록시 또는 서버 사용자 인증 요청에 사용되는지 여부를 지정하는 상수를 정의합니다.
BG_CERT_STORE_LOCATION 인증서 저장소의 위치를 지정하는 상수를 정의합니다.

BG_ERROR_CONTEXT 오류가 발생한 컨텍스트를 지정하는 상수를 정의합니다.
BG_JOB_PRIORITY 작업의 우선 순위 수준을 지정하는 상수를 정의합니다.
BG_JOB_PROXY_USAGE 파일 전송에 사용할 프록시를 지정하는 상수를 정의합니다. 각 작업에 대해 서로 다른 프록시 설정을 정의할 수 있습니다.
BG_JOB_STATE 작업의 다양한 상태를 지정하는 상수를 정의합니다.
BG_JOB_TYPE 다운로드와 같은 전송 작업의 유형을 지정하는 상수를 정의합니다.
BITS_FILE_PROPERTY_ID 백그라운드 복사 파일 속성에 해당하는 ID 값을 지정하는 상수를 정의합니다.
BITS_JOB_PROPERTY_ID BITS 작업에 대한 속성의 ID를 지정하는 상수를 정의합니다.
BITS_JOB_TRANSFER_POLICY BITS 속성에 해당하는 ID 값을 지정하는 상수를 정의합니다.
GROUPPROP GROUPPROP 열거형은 그룹 속성 값을 검색하고 설정하기 위한 상수 값을 정의합니다.

Functions

AddFile 작업에 단일 파일을 추가합니다.

AddFiles AddFiles 메서드를 사용하여 작업에 다운로드할 파일을 하나 이상 추가합니다.
AddFileSet 작업에 여러 파일을 추가합니다.
AddFileWithRanges 다운로드 작업에 파일을 추가하고 다운로드하려는 파일의 범위를 지정합니다.
취소 전송 큐에서 작업을 삭제하고 클라이언트(다운로드) 및 서버(업로드)에서 관련 임시 파일을 제거합니다.
CancelGroup CancelGroup 메서드를 사용하여 큐에서 그룹을 제거합니다. 이 메서드를 호출하기 전에 완전히 다운로드된 파일은 클라이언트에서 사용할 수 있습니다. 언제든지 그룹을 취소할 수 있습니다. 그러나 그룹이 취소되면 복구할 수 없습니다.
ClearHelperToken 도우미 토큰을 삭제하고 사용 플래그를 변경하지 않습니다.
ClearPeers 콘텐츠를 제공할 수 있는 피어 목록에서 모든 피어를 제거합니다.
ClearRecords 캐시에서 모든 레코드와 파일을 제거합니다.
복제 현재 열거형 상태와 동일한 열거형 상태를 포함하는 다른 IEnumBackgroundCopyFiles 열거자를 만듭니다.
복제 현재 열거형 상태와 동일한 열거형 상태를 포함하는 다른 IEnumBackgroundCopyJobs 열거자를 만듭니다.
복제 현재 열거형 상태와 동일한 열거자를 포함하는 다른 IEnumBitsPeerCacheRecords 열거자를 만듭니다.

복제 현재 열거형 상태와 동일한 열거형 상태를 포함하는 다른 IEnumBitsPeers 열거자를 만듭니다.
복제 Clone 메서드를 사용하여 현재 열거형 상태와 동일한 열거형 상태를 포함하는 다른 IEnumBackgroundCopyGroups 열거자를 만듭니다.
복제 Clone 메서드를 사용하여 현재 열거형 상태와 동일한 열거형 상태를 포함하는 다른 IEnumBackgroundCopyJobs1 열거자를 만듭니다.
완료 작업을 종료하고 전송된 파일을 클라이언트에 저장합니다.
CreateGroup CreateGroup 메서드를 사용하여 새 그룹을 만들고 다운로드 큐에 추가합니다.
CreateJob 작업을 만듭니다.
CreateJob CreateJob 메서드를 사용하여 그룹에 새 작업을 추가합니다. 그룹은 하나의 작업만 포함할 수 있습니다.
DeleteRecord 캐시에서 레코드 및 파일을 삭제합니다. 이 메서드는 레코드의 식별자를 사용하여 삭제할 레코드를 식별합니다.
DeleteUrl 지정된 URL에 대한 캐시에서 모든 캐시 레코드와 파일을 삭제합니다.
DisableBITSUploads DisableBITSUploads 메서드를 사용하여 ADSI 개체가 가리키는 가상 디렉터리에서 BITS 업로드를 사용하지 않도록 설정합니다. 이 메서드는 BITSUploadEnabled IIS 확장 속성을 설정합니다.
DiscoverPeers 콘텐츠를 제공할 수 있는 피어 목록을 생성합니다.

EnableBITSUploads EnableBITSUploads 메서드를 사용하여 ADSI 개체가 가리키는 가상 디렉터리에서 BITS 업로드를 사용하도록 설정합니다. 이 메서드는 BITSUploadEnabled IIS 확장 속성을 설정합니다.
EnumFiles 작업의 파일을 열거하는 데 사용하는 IEnumBackgroundCopyFiles 인터페이스 포인터를 검색합니다.
EnumGroups EnumGroups 메서드를 사용하여 현재 사용자가 소유한 그룹 목록을 검색합니다. 현재 사용자에게 관리자 권한이 있는 경우 메서드는 큐의 모든 그룹을 반환합니다.
EnumJobs 전송 큐에서 작업을 열거하는 데 사용하는 열거자 개체에 대한 인터페이스 포인터를 검색합니다. 열거자의 작업 순서는 임의입니다.
EnumJobs EnumJobs 메서드를 사용하여 그룹의 작업 목록을 검색합니다. 목록에는 하나의 작업만 포함됩니다.
EnumPeers 콘텐츠를 제공할 수 있는 피어를 열거하는 데 사용하는 IEnumBitsPeers 인터페이스 포인터를 가져옵니다. 열거형은 캐시에 있는 레코드의 스냅샷입니다.
EnumRecords 캐시의 레코드를 열거하는 데 사용하는 IEnumBitsPeerCacheRecords 인터페이스 포인터를 가져옵니다. 열거형은 캐시에 있는 레코드의 스냅샷입니다.
FileRangesTransferred BITS는 하나 이상의 파일 범위가 다운로드되었을 때 FileRangesTransferred 메서드의 구현을 호출합니다. 파일 범위는 IBackgroundCopyFile6::RequestFileRanges 메서드를 사용하여 작업에 추가됩니다.
FileTransferred BITS는 BITS가 파일 전송을 성공적으로 완료할 때 FileTransferred 메서드의 구현을 호출합니다.
get_GroupID get_GroupID 메서드를 사용하여 그룹의 식별자를 검색합니다.

get_JobID get_JobID 메서드를 사용하여 작업의 식별자를 검색합니다.
get_Size get_Size 메서드를 사용하여 다운로드할 그룹의 모든 파일 크기를 검색합니다.
GetCleanupTask GetCleanupTask 메서드를 사용하여 가상 디렉터리에 연결된 정리 작업에 대한 인터페이스 포인터를 검색합니다.
GetCleanupTaskName GetCleanupTaskName 메서드를 사용하여 가상 디렉터리에 연결된 정리 작업의 이름을 검색합니다.
GetClientCertificate 작업에서 클라이언트 인증서를 검색합니다.
GetConfigurationFlags 구성 플래그를 결정 하는 경우 컴퓨터 피어 컴퓨터에 콘텐츠를 제공 합니다. 피어 로부터 콘텐츠를 다운로드할 수를 가져옵니다.
GetCount 열거형의 파일 수를 검색합니다.
GetCount 열거형의 작업 수를 검색합니다.
GetCount 열거형의 캐시 레코드 수를 검색합니다.
GetCount 열거형의 피어 수 수를 검색합니다.
GetCount GetCount 메서드를 사용하여 열거형의 그룹 수를 검색합니다.

GetCount GetCount 메서드를 사용하여 열거형의 작업 수를 검색합니다.
GetCustomHeaders IBackgroundCopyJobHttpOptions::SetCustomHeaders에 대한 이전 호출에 의해 설정된 사용자 지정 헤더를 검색합니다(즉, BITS가 원격에서 수신하는 헤더가 아니라 원격으로 보낼 헤더).
GetDescription 작업에 대한 설명을 검색합니다.
GetDisplayName 작업의 표시 이름을 검색합니다. 일반적으로 표시 이름을 사용하여 사용자 인터페이스에서 작업을 식별합니다.
GetError 오류 코드를 검색하고 오류가 발생한 컨텍스트를 식별합니다.
GetError 오류가 발생한 후 오류 인터페이스를 검색합니다.
GetErrorContextDescription 오류가 발생한 컨텍스트에 대한 설명을 검색합니다.
GetErrorCount BITS가 작업을 전송하려고 시도하고 오류가 발생한 횟수를 검색합니다.
GetErrorDescription 오류와 연결된 오류 텍스트를 검색합니다.
GetErrorDescription 지정된 오류 코드에 대한 설명을 검색합니다.
GetFile 오류와 연결된 파일 개체에 대한 인터페이스 포인터를 검색합니다.
GetFile GetFile 메서드를 사용하여 작업에서 지정된 파일에 대한 원격 및 로컬 파일 이름을 검색합니다.

GetFileACLFlags SMB를 사용하여 파일을 전송할 때 유지할 소유자 및 ACL 정보를 식별하는 플래그를 검색합니다.
GetFileCount GetFileCount 메서드를 사용하여 작업의 파일 수를 검색합니다.
GetFileModificationTime 서버에서 파일이 마지막으로 수정된 날짜와 시간을 가져옵니다.
GetFileRanges 캐시에 있는 파일의 범위를 가져옵니다.
GetFileRanges 원격 파일에서 다운로드하려는 범위를 검색합니다.
GetFileSize 파일의 크기를 가져옵니다.
GetFilledFileRanges 다운로드한 파일 범위 집합을 반환합니다.
GetGroup GetGroup 메서드를 사용하여 다운로드 큐에서 그룹을 검색합니다.
GetHelperTokenFlags BITS 전송 작업과 연결된 토큰의 사용 플래그를 반환합니다.
GetHelperTokenSid 도우미 토큰이 설정된 경우 도우미 토큰의 SID를 반환합니다.
GetHttpMethod BITS 전송에 대한 HTTP 메서드 이름을 포함하는 와이드 문자열을 검색합니다. 기본적으로 다운로드 작업은 "GET"이 되고 업로드 및 업로드-회신 작업은 "BITS_POST"입니다.
GetId 큐에서 작업을 식별하는 데 사용되는 식별자를 검색합니다.

GetId 캐시의 레코드를 고유하게 식별하는 식별자를 가져옵니다.
GetJob 전송 큐에서 지정된 작업을 검색합니다. 일반적으로 애플리케이션은 작업 식별자를 유지하므로 나중에 큐에서 작업을 검색할 수 있습니다.
GetJob GetJob 메서드를 사용하여 그룹에서 작업을 검색합니다.
GetLastAccessTime 파일에 마지막으로 액세스한 날짜와 시간을 가져옵니다.
GetLocalName 파일의 로컬 이름을 검색합니다.
GetMaximumCacheSize 캐시의 최대 크기를 가져옵니다.
GetMaximumContentAge 파일이 캐시에서 제거되는 기간을 가져옵니다.
GetMaximumDownloadTime BITS가 작업에서 파일을 전송하는 데 소요되는 최대 시간을 검색합니다.
GetMinimumRetryDelay 파일을 전송하기 전에 일시적인 오류 조건이 발생한 후 서비스가 대기하는 최소 시간을 검색합니다.
GetNoProgressTimeout 일시적인 오류 조건이 발생한 후 서비스에서 파일 전송을 시도하는 시간을 검색합니다. 진행률이 있으면 타이머가 다시 설정됩니다.
GetNotifyCmdLine 작업이 오류 또는 전송된 상태가 될 때 실행할 프로그램을 검색합니다.

GetNotifyFlags 작업에 대한 이벤트 알림 플래그를 검색합니다.
GetNotifyInterface IBackgroundCopyCallback 인터페이스의 구현에 대한 인터페이스 포인터를 검색합니다.
GetObject GetObject 메서드를 사용하여 IBITSExtensionSetup 인터페이스에 대한 포인터를 검색합니다. 이 메서드는 ADsGetObject ADSI 함수가 수행하는 것과 동일한 바인딩을 수행합니다.
GetOriginUrl 캐시된 파일의 원본 URL을 가져옵니다.
GetOwner 작업 소유자의 ID를 검색합니다.
GetOwnerElevationState 소유자의 토큰이 작업을 만들거나 소유권을 가져올 때 상승되었는지 여부를 결정하는 값을 가져옵니다.
GetOwnerIntegrityLevel 작업의 소유권을 만들거나 가져온 소유자 토큰의 무결성 수준을 가져옵니다.
GetPeerCachingFlags 작업의 파일을 캐시하고 피어에 제공 할 수 있는지 여부와 BITS가 피어에서 작업에 대한 콘텐츠를 다운로드 할 수 있는지 여부를 결정하는 플래그를 검색합니다.
GetPeerDownloadStats 피어 및 원본 서버에서 다운로드한 데이터의 양에 대한 통계를 지정합니다.
GetPeerName 피어를 고유하게 식별하는 서버 보안 주체 이름을 가져옵니다.
GetPriority 작업의 우선 순위 수준을 검색합니다. 우선 순위 수준은 전송 큐의 다른 작업을 기준으로 작업이 처리되는 시기를 결정합니다.

GetProgress 파일 전송 진행률에 대한 정보를 검색합니다.
GetProgress 전송된 바이트 및 파일 수와 같은 작업 관련 진행률 정보를 검색합니다.
GetProgress GetProgress 메서드를 사용하여 다운로드 진행률을 검색합니다.
GetProgress GetProgress 메서드를 사용하여 작업의 진행률을 검색합니다.
GetProp GetProp 메서드를 사용하여 그룹에서 속성 값을 검색합니다.
Getproperty BITS 파일 전송의 제네릭 속성을 가져옵니다.
Getproperty BITS 작업 속성을 가져오기 위한 제네릭 메서드입니다.
GetProtocol 파일을 전송하는 데 사용되는 프로토콜을 검색합니다. 원격 파일 이름은 파일을 전송하는 데 사용할 프로토콜을 식별합니다.
GetProxySettings 작업이 파일을 전송하는 데 사용하는 프록시 정보를 검색합니다.
GetRecord 캐시에서 레코드를 가져옵니다.
GetRemoteName 파일의 원격 이름을 검색합니다.

GetReplyData 서버 애플리케이션에서 회신 데이터의 메모리 내 복사본을 검색합니다. 작업의 형식이 BG_JOB_TYPE_UPLOAD_REPLY 상태가 BG_JOB_STATE_TRANSFERRED 경우에만 이 메서드를 호출합니다.
GetReplyFileName 서버 애플리케이션에서 회신 데이터를 포함하는 파일의 이름을 검색합니다. 작업 유형이 BG_JOB_TYPE_UPLOAD_REPLY 경우에만 이 메서드를 호출합니다.
GetReplyProgress 업로드-회신 작업에서 회신 데이터 전송과 관련된 진행률 정보를 검색합니다.
GetSecurityFlags 인증서 해지 목록을 검사하고 특정 인증서 오류가 무시되는지 여부를 결정하는 HTTP에 대한 플래그와 서버가 HTTP 요청을 리디렉션할 때 사용할 정책을 검색합니다.
GetState 작업의 상태를 검색합니다.
GetStatus GetStatus 메서드를 사용하여 그룹의 상태를 검색합니다.
GetStatus GetStatus 메서드를 사용하여 작업의 상태를 검색합니다.
GetTemporaryName 다운로드 내용이 포함된 임시 파일의 전체 경로를 가져옵니다.
GetTimes 작업을 만들거나 마지막으로 수정한 시간과 같은 작업 관련 타임스탬프를 검색합니다.
GetType 파일 다운로드 또는 업로드와 같이 수행되는 전송 유형을 검색합니다.
GetValidationState 이 파일의 현재 유효성 검사 상태를 가져옵니다.

IsAuthenticated 피어가 인증되는지 여부를 결정합니다.
IsAvailable 피어를 콘텐츠를 제공할 수 있는지(온라인) 여부를 결정합니다.
IsDownloadedFromPeer 파일의 일부가 피어에서 다운로드되었는지 여부를 결정하는 값을 가져옵니다.
IsFileValidated 파일의 유효성을 검사했는지 여부를 확인합니다.
JobError BITS는 작업 상태가 BG_JOB_STATE_ERROR 변경되면 JobError 메서드의 구현을 호출합니다.
JobModification BITS는 작업이 수정될 때 JobModification 메서드의 구현을 호출합니다.
JobTransferred BITS는 작업의 모든 파일이 성공적으로 전송되면 JobTransferred 메서드의 구현을 호출합니다.
MakeCustomHeadersWriteOnly 이 작업에 대한 HTTP 사용자 지정 헤더를 쓰기 전용으로 설정합니다.
다음 열거형 시퀀스에서 지정된 수의 항목을 검색합니다. 시퀀스에 남아 있는 요청된 요소 수보다 적으면 나머지 요소를 검색합니다. (IEnumBackgroundCopyFiles.Next)
다음 열거형 시퀀스에서 지정된 수의 항목을 검색합니다. 시퀀스에 남아 있는 요청된 요소 수보다 적으면 나머지 요소를 검색합니다. (IEnumBackgroundCopyJobs.Next)
다음 열거형 시퀀스에서 지정된 수의 항목을 검색합니다. 시퀀스에 남아 있는 요청된 요소 수보다 적으면 나머지 요소를 검색합니다. (IEnumBitsPeerCacheRecords.Next)

다음 열거형 시퀀스에서 지정된 수의 항목을 검색합니다. 시퀀스에 남아 있는 요청된 요소 수보다 적으면 나머지 요소를 검색합니다. (IEnumBitsPeers.Next)
다음 Next 메서드를 사용하여 열거형 시퀀스에서 지정된 수의 항목을 검색합니다. 시퀀스에 남아 있는 요청된 요소 수보다 적으면 나머지 요소를 검색합니다. (IEnumBackgroundCopyGroups.Next)
다음 Next 메서드를 사용하여 열거형 시퀀스에서 지정된 수의 항목을 검색합니다. 시퀀스에 남아 있는 요청된 요소 수보다 적으면 나머지 요소를 검색합니다. (IEnumBackgroundCopyJobs1.Next)
OnStatus 그룹이 완료되거나 오류가 발생할 때 알림을 받도록 OnStatus 메서드를 구현합니다.
RemoveClientCertificate 작업에서 클라이언트 인증서를 제거합니다.
RemoveCredentials 사용에서 자격 증명을 제거합니다. 자격 증명은 IBackgroundCopyJob2::SetCredentials 메서드를 사용하여 지정한 기존 대상 및 구성표 쌍과 일치해야 합니다. 설정한 자격 증명을 검색하는 방법은 없습니다.
ReplaceRemotePrefix 다운로드 작업의 모든 원격 이름의 시작 텍스트를 지정된 문자열로 바꿉니다.
RequestFileRanges 다운로드 우선 순위를 지정하는 새 파일 범위 집합을 추가합니다.
재설정 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumBackgroundCopyFiles.Reset)
재설정 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumBackgroundCopyJobs.Reset)
재설정 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumBitsPeerCacheRecords.Reset)

재설정 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumBitsPeers.Reset)
재설정 Reset 메서드를 사용하여 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumBackgroundCopyGroups.Reset)
재설정 Reset 메서드를 사용하여 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumBackgroundCopyJobs1.Reset)
재개 새 작업을 활성화하거나 일시 중단된 작업을 다시 시작합니다.
ResumeGroup ResumeGroup 메서드를 사용하여 다운로드 큐에서 일시 중단된 그룹을 시작합니다.
SetClientCertificateByID HTTPS(SSL) 요청에서 클라이언트 인증에 사용할 클라이언트 인증서의 식별자를 지정합니다.
SetClientCertificateByName HTTPS(SSL) 요청에서 클라이언트 인증에 사용할 클라이언트 인증서의 주체 이름을 지정합니다.
SetConfigurationFlags 컴퓨터 피어 컴퓨터에 콘텐츠를 사용할 수 및 피어에서 콘텐츠를 다운로드 하는 경우 결정 하는 구성 플래그를 설정 합니다.
SetCredentials 프록시 또는 원격 서버 사용자 인증 요청에 사용할 자격 증명을 지정합니다.
SetCustomHeaders HTTP 요청에 포함할 하나 이상의 사용자 지정 HTTP 헤더를 지정합니다.
SetDescription 작업에 대한 설명을 제공합니다.

SetDisplayName 작업의 표시 이름을 지정합니다. 일반적으로 표시 이름을 사용하여 사용자 인터페이스에서 작업을 식별합니다.
SetFileACLFlags SMB를 사용하여 파일을 다운로드하거나 업로드할 때 유지 관리할 소유자 및 ACL 정보를 지정합니다.
SetHelperToken COM 클라이언트의 토큰을 가장하도록 도우미 토큰을 설정합니다.
SetHelperTokenFlags BITS 전송 작업과 연결된 토큰의 사용 플래그를 설정합니다.
SetHttpMethod BITS 전송에 사용되는 기본 HTTP 메서드를 재정의합니다.
SetMaximumCacheSize 캐시의 최대 크기를 지정합니다.
SetMaximumContentAge 기간을 기준으로 캐시에서 파일이 제거되는 시기를 지정합니다.
SetMaximumDownloadTime BITS가 작업에서 파일을 전송하는 데 소요되는 최대 시간을 설정합니다.
SetMinimumRetryDelay 파일을 전송하기 전에 일시적 오류 조건이 발생한 후 BITS가 대기하는 최소 시간을 설정합니다.
SetNoProgressTimeout 일시적 오류 조건이 발생한 후 BITS에서 파일 전송을 시도하는 시간을 설정합니다. 진행률이 있으면 타이머가 다시 설정됩니다.
SetNotifyCmdLine 작업이 BG_JOB_STATE_ERROR 또는 BG_JOB_STATE_TRANSFERRED 상태로 들어가면 실행할 프로그램을 지정합니다. BITS는 이 메서드를 호출한 사용자의 컨텍스트에서 프로그램을 실행합니다.

SetNotifyFlags 수신하려는 이벤트 알림 유형(예: 작업 전송 이벤트)을 지정합니다.
SetNotifyInterface BITS에 대한 IBackgroundCopyCallback 인터페이스의 구현을 식별합니다. IBackgroundCopyCallback 인터페이스를 사용하여 작업 관련 이벤트에 대한 알림을 받습니다.
SetPeerCachingFlags 피어에서 콘텐츠는 작업의 파일을 캐시 하 고 동료에 게 제공 하 고 작업을 다운로드할 수 하는 경우를 결정 하는 플래그를 설정 합니다.
SetPriority 작업의 우선 순위 수준을 지정합니다. 우선 순위 수준은 전송 큐의 다른 작업을 기준으로 작업이 처리되는 시기를 결정합니다.
SetProp SetProp 메서드를 사용하여 그룹 속성의 속성 값을 설정합니다.
SetProperty BITS 파일 전송의 제네릭 속성을 설정합니다.
SetProperty BITS 작업 속성을 설정하기 위한 제네릭 메서드입니다.
SetProxySettings 파일을 전송하는 데 사용할 프록시를 지정합니다.
SetRemoteName 원격 이름을 다운로드 작업의 새 URL로 변경합니다.
SetReplyFileName 서버 애플리케이션의 회신 데이터를 포함할 파일의 이름을 지정합니다. 작업의 형식이 BG_JOB_TYPE_UPLOAD_REPLY 경우에만 이 메서드를 호출합니다.
SetSecurityFlags 인증서 해지 목록을 확인하고 특정 인증서 오류가 무시되는지 여부와 서버가 HTTP 요청을 리디렉션할 때 사용할 정책을 결정하는 HTTP에 대한 플래그를 설정합니다.

SetServerCertificateValidationInterface 서버 인증서는 HTTPS 연결을 열 때 전송됩니다. 이 메서드를 사용하여 호출할 콜백을 설정하여 해당 서버 인증서의 유효성을 검사합니다.
SetValidationState 이 파일의 유효성 검사 상태를 설정합니다.
skip 열거형 시퀀스에서 지정한 다음 개수의 요소를 건너뜁니다. 시퀀스에 건너뛸 요청된 요소 수보다 적은 요소가 남아 있는 경우 시퀀스의 마지막 요소를 건너뜁니다. (IEnumBackgroundCopyFiles.Skip)
skip 열거형 시퀀스에서 지정한 다음 개수의 요소를 건너뜁니다. 시퀀스에 건너뛸 요청된 요소 수보다 적은 요소가 남아 있는 경우 시퀀스의 마지막 요소를 건너뜁니다. (IEnumBackgroundCopyJobs.Skip)
skip 열거형 시퀀스에서 지정한 다음 개수의 요소를 건너뜁니다. 시퀀스에 건너뛸 요청된 요소 수보다 적은 요소가 남아 있는 경우 시퀀스의 마지막 요소를 건너뜁니다. (IEnumBitsPeerCacheRecords.Skip)
skip 열거형 시퀀스에서 지정한 다음 개수의 요소를 건너뜁니다. 시퀀스에 건너뛸 요청된 요소 수보다 적은 요소가 남아 있는 경우 시퀀스의 마지막 요소를 건너뜁니다. (IEnumBitsPeers.Skip)
skip Skip 메서드를 사용하여 열거형 시퀀스에서 지정한 다음 개수의 요소를 건너뜁니다. 시퀀스에 건너뛸 요청된 요소 수보다 적은 요소가 남아 있는 경우 시퀀스의 마지막 요소를 건너뜁니다. (IEnumBackgroundCopyGroups.Skip)
skip Skip 메서드를 사용하여 열거형 시퀀스에서 지정한 다음 개수의 요소를 건너뜁니다. 시퀀스에 건너뛸 요청된 요소 수보다 적은 요소가 남아 있는 경우 시퀀스의 마지막 요소를 건너뜁니다. (IEnumBackgroundCopyJobs1.Skip)
Suspend 작업을 일시 중단합니다. 새 작업, 오류가 발생한 작업 및 파일 전송을 완료한 작업은 자동으로 일시 중단됩니다.

SuspendGroup SuspendGroup 메서드를 사용하여 다운로드 큐에서 그룹을 일시 중지합니다. 새 그룹, 오류가 발생한 그룹 또는 다운로드를 완료한 그룹은 자동으로 일시 중단됩니다.
SwitchToForeground SwitchToForeground 메서드를 사용하여 배경 대신 전경에서 그룹을 다운로드합니다.
TakeOwnership 작업의 소유권을 현재 사용자로 변경합니다.
UpdateDownloadPosition 누락된 데이터 다운로드의 우선 순위를 지정할 위치를 지정합니다.
ValidateServerCertificate HTTPS 연결이 열릴 때 전송되는 서버 인증서의 유효성을 검사할 수 있도록 호출되는 구현 콜백 메서드입니다.

인터페이스

IBackgroundCopyCallback IBackgroundCopyCallback 인터페이스를 구현하여 작업이 완료되었거나 수정되었거나 오류가 발생했음을 알 수 있습니다. 클라이언트는 작업 상태에 대해 폴링하는 대신 이 인터페이스를 사용합니다.
IBackgroundCopyCallback1 이벤트가 발생할 때 알림을 수신하도록 IBackgroundCopyCallback1 인터페이스를 구현합니다.
IBackgroundCopyCallback2 파일 다운로드가 완료되었다는 알림을 받도록 이 인터페이스를 구현합니다.
IBackgroundCopyCallback3 클라이언트는 IBackgroundCopyCallback3 인터페이스를 구현하여 파일 범위가 다운로드를 완료했다는 알림을 받습니다.

IBackgroundCopyError IBackgroundCopyError 인터페이스를 사용하여 오류의 원인과 전송 프로세스를 진행할 수 있는지 여부를 확인합니다.
IBackgroundCopyFile IBackgroundCopyFile에는 작업의 일부인 파일에 대한 정보가 포함되어 있습니다. 예를 들어 IBackgroundCopyFile 메서드를 사용하여 파일의 로컬 및 원격 이름을 검색하고 진행률 정보를 전송할 수 있습니다.
IBackgroundCopyFile2 IBackgroundCopyFile2 인터페이스를 사용하여 파일의 새 원격 이름을 지정하고 다운로드할 범위 목록을 검색합니다.
IBackgroundCopyFile3 이 인터페이스를 사용하여 다운로드한 콘텐츠가 포함된 임시 파일의 이름을 검색하고 피어가 콘텐츠를 요청할 수 있도록 파일의 유효성을 검사합니다.
IBackgroundCopyFile4 피어 및 원본 서버에 대한 다운로드 통계를 검색하려면 이 인터페이스를 사용합니다.
IBackgroundCopyFile5 이 인터페이스를 사용하여 BITS 파일 전송의 제네릭 속성을 얻거나 설정합니다.
IBackgroundCopyFile6 이 인터페이스를 사용하여 주문형 다운로드 작업에 대한 파일 범위를 요청합니다.
IBackgroundCopyGroup IBackgroundCopyGroup 인터페이스를 사용하여 그룹을 관리합니다. 그룹에는 다운로드 작업이 포함됩니다. 예를 들어 그룹에 작업을 추가하고, 그룹의 속성을 설정하고, 다운로드 큐에서 그룹을 시작하고 중지합니다.
IBackgroundCopyJob IBackgroundCopyJob 인터페이스를 사용하여 작업에 파일을 추가하고, 작업의 우선 순위 수준을 설정하고, 작업의 상태를 확인하고, 작업을 시작 및 중지합니다.
IBackgroundCopyJob1 IBackgroundCopyJob1 인터페이스를 사용하여 작업에 파일을 추가하고 작업의 상태를 검색합니다.

IBackgroundCopyJob2 업로드-회신 작업에서 회신 데이터를 검색하고, 클라이언트로 회신 데이터 전송의 진행률을 결정하고, 명령줄 실행을 요청하고, 프록시 및 원격 서버 인증 요청에 대한 자격 증명을 제공합니다.
IBackgroundCopyJob3 IBackgroundCopyJob3 인터페이스를 사용하여 파일 범위를 다운로드하고 원격 파일 이름의 접두사를 변경합니다.
IBackgroundCopyJob4 이 인터페이스를 사용하여 피어 캐싱을 사용하도록 설정하고, 다운로드 시간을 제한하고, 사용자 토큰 특성을 검사합니다.
IBackgroundCopyJob5 이 인터페이스를 사용하여 작업의 몇 가지 선택적 동작을 쿼리하거나 설정합니다.
IBackgroundCopyJobHttpOptions 이 인터페이스를 사용하여 인증서 기반 클라이언트 인증에 대한 클라이언트 인증서와 HTTP 요청에 대한 사용자 지정 헤더를 지정합니다.
IBackgroundCopyJobHttpOptions2 이 인터페이스를 사용하여 BITS 전송에 사용되는 HTTP 메서드를 검색 및/또는 재정의할 수 있습니다.
IBackgroundCopyJobHttpOptions3 이 인터페이스를 사용하여 HTTP 고객 헤더를 쓰기 전용으로 설정하거나 구현한 서버 인증서 유효성 검사 콜백 메서드를 설정합니다.
IBackgroundCopyManager 전송 작업을 만들고, 큐에 있는 작업이 포함된 열거자 개체를 검색하고, 큐에서 개별 작업을 검색합니다.
IBackgroundCopyQMgr IBackgroundCopyQMgr 인터페이스를 사용하여 새 그룹을 만들거나, 기존 그룹을 검색하거나, 큐의 모든 그룹을 열거합니다. 그룹에 다운로드 작업이 포함되어 있습니다.
IBackgroundCopyServerCertificateValidationCallback 서버 인증서는 HTTPS 연결을 열 때 전송됩니다. 이 메서드를 사용하여 호출할 콜백을 구현하여 해당 서버 인증서의 유효성을 검사합니다.

IBITSExtensionSetup IBITSExtensionSetup 인터페이스를 사용하여 가상 디렉터리에 대한 BITS 업로드를 사용하거나 사용하지 않도록 설정합니다.
IBITSExtensionSetupFactory IBITSExtensionSetupFactory 인터페이스를 사용하여 IBITSExtensionSetup 인터페이스에 대한 포인터를 가져옵니다.
IBitsPeer IBitsPeer를 사용하여 이웃의 피어에 대한 정보를 가져옵니다.
IBitsPeerCacheAdministration IBitsPeerCacheAdministration을 사용하여 콘텐츠를 다운로드할 수 있는 피어 풀을 관리합니다.
IBitsPeerCacheRecord IBitsPeerCacheRecord를 사용하여 캐시의 파일에 대한 정보를 가져옵니다.
IBitsTokenOptions IBitsTokenOptions를 사용하여 BITS(Background Intelligent Transfer Service) 전송 작업에 대한 보안 토큰 쌍을 연결하고 관리합니다.
IEnumBackgroundCopyFiles IEnumBackgroundCopyFiles 인터페이스를 사용하여 작업에 포함된 파일을 열거합니다. IEnumBackgroundCopyFiles 인터페이스 포인터를 얻으려면 IBackgroundCopyJob::EnumFiles 메서드를 호출합니다.
IEnumBackgroundCopyGroups IEnumBackgroundCopyGroups 인터페이스를 사용하여 다운로드 큐의 그룹 목록을 열거합니다. IEnumBackgroundCopyGroups 인터페이스 포인터를 얻으려면 IBackgroundCopyQMgr::EnumGroups 메서드를 호출합니다.
IEnumBackgroundCopyJobs IEnumBackgroundCopyJobs 인터페이스를 사용하여 전송 큐의 작업 목록을 열거합니다. IEnumBackgroundCopyJobs 인터페이스 포인터를 얻으려면 IBackgroundCopyManager::EnumJobs 메서드를 호출합니다.

IEnumBackgroundCopyJobs1 IEnumBackgroundCopyJobs1 인터페이스를 사용하여 그룹의 작업 목록을 열거합니다. IEnumBackgroundCopyJobs1 인터페이스 포인터를 얻으려면 IBackgroundCopyGroup::EnumJobs 메서드를 호출합니다.
IEnumBitsPeerCacheRecords IEnumBitsPeerCacheRecords를 사용하여 캐시의 레코드를 열거합니다.
IEnumBitsPeers IEnumBitsPeers를 사용하여 BITS에서 검색한 피어 목록을 열거합니다.

구조체

BG_AUTH_CREDENTIALS 사용자 인증 요청에 사용할 대상(프록시 또는 서버), 인증 체계 및 사용자의 자격 증명을 식별합니다. 구조체는 IBackgroundCopyJob2::SetCredentials 메서드에 전달됩니다.
BG_AUTH_CREDENTIALS_UNION BG_AUTH_CREDENTIALS 구조에 지정된 인증 체계에 사용할 자격 증명을 식별합니다.
BG_BASIC_CREDENTIALS BG_BASIC_CREDENTIALS 구조는 인증할 사용자 이름과 암호를 식별합니다.
BG_FILE_INFO 전송할 파일의 로컬 및 원격 이름을 제공합니다.
BG_FILE_PROGRESS 전송된 바이트 수와 같은 파일 관련 진행률 정보를 제공합니다.
BG_FILE_RANGE 파일에서 다운로드할 바이트 범위를 식별합니다.
BG_JOB_PROGRESS 전송된 바이트 및 파일 수와 같은 작업 관련 진행률 정보를 제공합니다.

BG_JOB_REPLY_PROGRESS 업로드-회신 작업의 회신 부분과 관련된 진행률 정보를 제공합니다.
BG_JOB_TIMES 작업 관련 타임스탬프를 제공합니다.
BITS_FILE_PROPERTY_VALUE BITS 파일의 속성 값을 제공합니다.
BITS_JOB_PROPERTY_VALUE BITS_JOB_PROPERTY_ID 열거형의 값을 기반으로 BITS 작업의 속성 값을 제공합니다.
FILESETINFO FILESETINFO 구조체는 다운로드할 파일의 원격 및 로컬 이름을 식별합니다.

Backup

아티클2023. 03. 09.

Backup 기술 개요입니다.

Backup을 개발하려면 다음 헤더가 필요합니다.

- winbase.h
- winnt.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Backup

열거형

JOB_OBJECT_NET_RATE_CONTROL_FLAGS 네트워크 속도 제어에 대한 일정 정책 유형을 지정합니다.
LOGICAL_PROCESSOR_RELATIONSHIP 해당 SYSTEM_LOGICAL_PROCESSOR_INFORMATION 또는 SYSTEM_LOGICAL_PROCESSOR_INFORMATION_EX 구조에서 식별된 프로세서 집합 간의 관계를 나타냅니다.
PROCESS_MITIGATION_POLICY 다양한 프로세스 완화 정책을 나타냅니다.
PROCESSOR_CACHE_TYPE 해당 CACHE_DESCRIPTOR 구조에서 식별되는 프로세서 캐시의 유형을 나타냅니다.

Functions

_InlineInterlockedAdd 지정된 LONG 값에 대해 원자성 추가 작업을 수행합니다. 이 작업은 메모리 주문 의미 체계를 획득하여 수행됩니다.

_InlineInterlockedAdd64 지정된 LONG64 값에 대해 원자성 추가 작업을 수행합니다.
_interlockedbittestandreset 지정된 LONG 값의 지정된 비트를 테스트하고 0으로 설정합니다. 이 작업은 원자성 작업입니다.
_interlockedbittestandreset64 지정된 LONG64 값의 지정된 비트를 테스트하고 0으로 설정합니다. 이 작업은 원자성 작업입니다.
_interlockedbittestandset 지정된 LONG 값의 지정된 비트를 테스트하고 1로 설정합니다. 이 작업은 원자성 작업입니다.
_interlockedbittestandset64 지정된 LONG64 값의 지정된 비트를 테스트하고 1로 설정합니다. 이 작업은 원자성 작업입니다.
AddIntegrityLabelToBoundaryDescriptor 지정된 경계 설명자에 새 필수 SID(보안 식별자)를 추가합니다.
BackupRead 보안 정보를 포함하여 파일 또는 디렉터리를 백업합니다.
BackupSeek BackupRead 또는 BackupWrite 함수를 사용하여 처음에 액세스한 데이터 스트림에서 전달을 추구합니다.
BackupWrite BackupRead를 사용하여 백업된 파일 또는 디렉터리를 복원합니다.
BindIoCompletionCallback 스레드 풀이 소유한 I/O 완료 포트를 지정된 파일 핸들과 연결합니다. 이 파일과 관련된 I/O 요청이 완료되면 I/O가 아닌 작업자 스레드가 지정된 콜백 함수를 실행합니다.
ConvertFiberToThread 현재 파이버를 스레드로 변환합니다.

ConvertThreadToFiber 현재 스레드를 파이버로 변환합니다. 다른 파이버를 예약하려면 먼저 스레드를 파이버로 변환해야 합니다. (ConvertThreadToFiber)
ConvertThreadToFiberEx 현재 스레드를 파이버로 변환합니다. 다른 파이버를 예약하려면 먼저 스레드를 파이버로 변환해야 합니다. (ConvertThreadToFiberEx)
CreateBoundaryDescriptorA ANSI(CreateBoundaryDescriptorA) 함수(winbase.h)는 경계 설명자를 만듭니다.
CreateFiber 파이버 개체를 할당하고, 스택을 할당하고, 지정된 시작 주소(일반적으로 파이버 함수)에서 시작하도록 실행을 설정합니다. 이 함수는 파이버를 예약하지 않습니다. (CreateFiber)
CreateFiberEx 파이버 개체를 할당하고, 스택을 할당하고, 지정된 시작 주소(일반적으로 파이버 함수)에서 시작하도록 실행을 설정합니다. 이 함수는 파이버를 예약하지 않습니다. (CreateFiberEx)
CreateJobObjectA 작업 개체를 만들거나 엽니다. (CreateJobObjectA)
CreateProcessWithLogonW 새 프로세스와 해당 기본 스레드를 만듭니다. 그런 다음, 새 프로세스는 지정된 자격 증명(사용자, 도메인 및 암호)의 보안 컨텍스트에서 지정된 실행 파일을 실행합니다. 필요에 따라 지정된 사용자에 대한 사용자 프로필을 로드할 수 있습니다.
CreateProcessWithTokenW 새 프로세스와 해당 기본 스레드를 만듭니다. 새 프로세스는 지정된 토큰의 보안 컨텍스트에서 실행됩니다. 필요에 따라 지정된 사용자에 대한 사용자 프로필을 로드할 수 있습니다.
CreateSemaphoreA 명명되거나 명명되지 않은 세마포 개체를 만들거나 엽니다. (CreateSemaphoreA)
CreateSemaphoreExA 명명되거나 명명되지 않은 세마포 개체를 만들거나 열고 개체에 대한 핸들을 반환합니다. (CreateSemaphoreExA)

CreateTapePartition 테이프를 다시 포맷합니다.
CreateUmsCompletionList UMS(사용자 모드 일정 예약) 완료 목록을 만듭니다.
CreateUmsThreadContext UMS 작업자 스레드를 나타내는 UMS(사용자 모드 예약) 스레드 컨텍스트를 만듭니다.
DeleteFiber 기존 파이버를 삭제합니다.
DeleteUmsCompletionList 지정된 UMS(사용자 모드 일정 예약) 완료 목록을 삭제합니다. 목록이 비어 있어야 합니다.
DeleteUmsThreadContext 지정된 UMS(사용자 모드 예약) 스레드 컨텍스트를 삭제합니다. 스레드를 종료해야 합니다.
DequeueUmsCompletionListItems 지정된 UMS 완료 목록에서 UMS(사용자 모드 예약) 작업자 스레드를 검색합니다.
DestroyThreadpoolEnvironment 지정된 콜백 환경을 삭제합니다. 새 스레드 풀 개체를 만드는 데 콜백 환경이 더 이상 필요하지 않은 경우 이 함수를 호출합니다. (DestroyThreadpoolEnvironment)
EnterUmsSchedulingMode 호출 스레드를 UMS(사용자 모드 예약) 스케줄러 스레드로 변환합니다.
EraseTape 테이프의 전부 또는 일부를 지웁니다.
ExecuteUmsThread 지정된 UMS 작업자 스레드를 실행합니다.
GetActiveProcessorCount 프로세서 그룹 또는 시스템의 활성 프로세서 수를 반환합니다.

GetActiveProcessorGroupCount 시스템의 활성 프로세서 그룹 수를 반환합니다.
GetCurrentFiber 현재 파이버의 주소를 검색합니다.
GetCurrentUmsThread 호출하는 UMS 스레드의 UMS(사용자 모드 예약) 스레드 컨텍스트를 반환합니다.
GetEnvironmentVariable GetEnvironmentVariable 함수(winbase.h)는 호출 프로세스의 환경 블록에서 지정된 변수의 내용을 검색합니다.
GetFiberData 현재 파이버와 연결된 파이버 데이터를 검색합니다.
GetMaximumProcessorCount 프로세서 그룹 또는 시스템에 있을 수 있는 논리 프로세서의 최대 수를 반환합니다.
GetMaximumProcessorGroupCount 시스템에 있을 수 있는 최대 프로세서 그룹 수를 반환합니다.
GetNextUmsListItem 스레드 컨텍스트 목록에서 다음 UMS(사용자 모드 예약) 스레드 컨텍스트를 반환합니다.
GetNumaAvailableMemoryNode 지정된 노드에서 사용할 수 있는 메모리 양을 검색합니다.
GetNumaAvailableMemoryNodeEx USHORT 값으로 지정된 노드에서 사용할 수 있는 메모리 양을 검색합니다.
GetNumaNodeNumberFromHandle 지정된 파일 핸들이 나타내는 파일 또는 I/O 디바이스와 연결된 NUMA 노드를 검색합니다.
GetNumaNodeProcessorMask 지정된 노드의 프로세서 마스크를 검색합니다.

GetNumaProcessorNode 지정된 프로세서의 노드 번호를 검색합니다.
GetNumaProcessorNodeEx 지정된 논리 프로세서에 대한 USHORT 값으로 노드 번호를 검색합니다.
GetNumaProximityNode 지정된 근접 도메인 식별자에 해당하는 NUMA 노드 번호를 검색합니다.
GetProcessAffinityMask 지정된 프로세스의 프로세스 선호도 마스크와 시스템의 시스템 선호도 마스크를 검색합니다.
GetProcessIoCounters 지정된 프로세스에서 수행하는 모든 I/O 작업에 대한 회계 정보를 검색합니다.
GetTapeParameters 테이프 또는 테이프 드라이브를 설명하는 정보를 검색합니다.
GetTapePosition 테이프의 현재 주소를 논리 또는 절대 블록으로 검색합니다.
GetTapeStatus 테이프 디바이스가 테이프 명령을 처리할 준비가 되었는지 여부를 결정합니다.
GetUmsCompletionListEvent 지정된 UMS(사용자 모드 예약) 완료 목록과 연결된 이벤트에 대한 핸들을 검색합니다.
GetUmsSystemThreadInformation 지정된 스레드가 UMS 스케줄러 스레드, UMS 작업자 스레드 또는 UMS가 아닌 스레드인지 여부를 쿼리합니다.
HasOverlappedIoCompleted 미해결 I/O 작업의 완료를 폴링하는 데 사용할 수 있는 고성능 테스트 작업을 제공합니다.
InitializeThreadpoolEnvironment 콜백 환경을 초기화합니다.

InterlockedAdd 지정된 LONG 값에 대해 원자성 추가 작업을 수행합니다.
InterlockedAnd 지정된 LONG 값에 대해 원자성 AND 연산을 수행합니다.
InterlockedAnd16 지정된 SHORT 값에 대해 원자성 AND 연산을 수행합니다.
InterlockedAnd64 지정된 LONGLONG 값에 대해 원자성 AND 연산을 수행합니다.
InterlockedAnd8 지정된 char 값에 대해 원자성 AND 연산을 수행합니다.
InterlockedCompareExchange 지정된 값에 대해 원자성 비교 및 교환 작업을 수행합니다. 이 함수는 두 개의 지정된 32비트 값과 교환을 비교 결과에 따라 다른 32비트 값과 비교합니다.
InterlockedCompareExchange128 지정된 값에 대해 원자성 비교 및 교환 작업을 수행합니다. 이 함수는 두 개의 지정된 128비트 값과 교환을 비교 결과에 따라 다른 128비트 값과 비교합니다.
InterlockedCompareExchange16 지정된 값에 대해 원자성 비교 및 교환 작업을 수행합니다. 이 함수는 두 개의 지정된 16비트 값과 교환을 비교 결과에 따라 다른 16비트 값과 비교합니다.
InterlockedCompareExchange64 지정된 값에 대해 원자성 비교 및 교환 작업을 수행합니다. 이 함수는 두 개의 지정된 64비트 값과 교환을 비교 결과에 따라 다른 64비트 값과 비교합니다.
InterlockedCompareExchangePointer 지정된 값에 대해 원자성 비교 및 교환 작업을 수행합니다. 이 함수는 두 개의 지정된 포인터 값과 교환을 비교 결과에 따라 다른 포인터 값과 비교합니다.
InterlockedDecrement 지정된 32비트 변수의 값을 원자성 연산으로 감소(1씩 감소)합니다.

InterlockedDecrement16 지정된 16비트 변수의 값을 원자성 연산으로 감소(1씩 감소)합니다.
InterlockedDecrement64 지정된 64비트 변수의 값을 원자성 연산으로 감소(1씩 감소)합니다.
InterlockedExchange 32비트 변수를 지정된 값으로 원자성 연산으로 설정합니다.
InterlockedExchange16 16비트 변수를 지정된 값으로 원자성 연산으로 설정합니다.
InterlockedExchange64 64비트 변수를 지정된 값으로 원자성 연산으로 설정합니다.
InterlockedExchange8 8비트 변수를 지정된 값으로 원자성 연산으로 설정합니다.
InterlockedExchangeAdd 두 개의 32비트 값에 원자성 추가를 수행합니다.
InterlockedExchangeAdd64 두 64비트 값의 원자성 추가를 수행합니다.
InterlockedExchangePointer 한 쌍의 주소를 원자 방식으로 교환합니다.
InterlockedExchangeSubtract 두 값의 원자성 빼기 수행
InterlockedIncrement 지정된 32비트 변수의 값을 원자성 연산으로 증분(1씩 증가)합니다.
InterlockedIncrement16 지정된 16비트 변수의 값을 원자성 연산으로 증분(1씩 증가)합니다.

InterlockedIncrement64 지정된 64비트 변수의 값을 원자성 연산으로 증분(1씩 증가)합니다.
InterlockedOr 지정된 LONG 값에 대해 원자성 OR 연산을 수행합니다.
InterlockedOr16 지정된 SHORT 값에 대해 원자성 OR 연산을 수행합니다.
InterlockedOr64 지정된 LONGLONG 값에 대해 원자성 OR 연산을 수행합니다.
InterlockedOr8 지정된 char 값에 대해 원자성 OR 연산을 수행합니다.
InterlockedXor 지정된 LONG 값에 대해 원자성 XOR 연산을 수행합니다.
InterlockedXor16 지정된 SHORT 값에 대해 원자성 XOR 연산을 수행합니다.
InterlockedXor64 지정된 LONGLONG 값에 대해 원자성 XOR 연산을 수행합니다.
InterlockedXor8 지정된 char 값에 대해 원자성 XOR 연산을 수행합니다.
MemoryBarrier CPU가 읽기 및 쓰기 작업을 다시 정렬하지 못하도록 하는 하드웨어 메모리 장벽(펜스)을 만듭니다. 컴파일러가 읽기 및 쓰기 작업을 다시 정렬하지 못할 수도 있습니다.
OpenJobObjectA 기존 작업 개체를 엽니다. (OpenJobObjectA)
PAPCFUNC 애플리케이션에서 정의한 완료 루틴입니다. QueueUserAPC 함수를 호출할 때 이 주소를 지정합니다.

PFIBER_START_ROUTINE CreateFiber 함수와 함께 사용되는 애플리케이션 정의 함수입니다. 파이버의 시작 주소로 사용됩니다.
PFLS_CALLBACK_FUNCTION 애플리케이션 정의 함수입니다. FLS 슬롯이 사용 중인 경우 FlsCallback은 파이버 삭제, 스레드 종료 및 FLS 인덱스가 해제되면 호출됩니다.
PreFetchCacheLine 가까운 장래에 캐시 라인이 필요할 것임을 프로세서에 나타냅니다.
PrepareTape 테이프에 액세스하거나 제거할 준비를 합니다.
PTP_CLEANUP_GROUP_CANCEL_CALLBACK 애플리케이션은 SetThreadpoolCallbackCleanupGroup 함수를 호출하여 CloseThreadpoolCleanupGroup을 호출할 때 사용할 콜백을 지정하는 경우 이 콜백을 구현합니다.
PulseEvent 지정된 이벤트 개체를 신호 상태로 설정한 다음 적절한 대기 스레드 수를 해제한 후 서명되지 않은 상태로 다시 설정합니다.
QueryFullProcessImageNameA 지정된 프로세스에 대한 실행 파일 이미지의 전체 이름을 검색합니다. (ANSI)
QueryFullProcessImageNameW 지정된 프로세스에 대한 실행 파일 이미지의 전체 이름을 검색합니다. (유니코드)
QueryUmsThreadInformation 지정된 UMS(사용자 모드 예약) 작업자 스레드에 대한 정보를 검색합니다.
RegisterWaitForSingleObject 스레드 풀의 대기 스레드가 개체를 대기하도록 지시합니다.
RTL_UMS_SCHEDULER_ENTRY_POINT UMS 완료 목록과 연결된 애플리케이션 정의 UMS(사용자 모드 예약) 스케줄러 진입점 함수입니다.

RtlFirstEntrySList 자동으로 연결된 목록의 첫 번째 항목을 검색합니다. 목록에 대한 액세스는 다중 프로세서 시스템에서 동기화됩니다.
RtlInitializeSListHead 자동으로 연결된 목록의 헤드를 초기화합니다. (RtlInitializeSListHead)
RtlInterlockedFlushSList 자동으로 연결된 목록에서 모든 항목을 제거합니다. 목록에 대한 액세스는 다중 프로세서 시스템에서 동기화됩니다. (RtlInterlockedFlushSList)
RtlInterlockedPopEntrySList 자동으로 연결된 목록의 앞면에서 항목을 제거합니다. 목록에 대한 액세스는 다중 프로세서 시스템에서 동기화됩니다. (RtlInterlockedPopEntrySList)
RtlInterlockedPushEntrySList 자동으로 연결된 목록의 맨 앞에 항목을 삽입합니다. 목록에 대한 액세스는 다중 프로세서 시스템에서 동기화됩니다. (RtlInterlockedPushEntrySList)
RtlQueryDepthSList 지정된 링크된 목록의 항목 수를 검색합니다. (RtlQueryDepthSList)
SetEnvironmentVariable SetEnvironmentVariable 함수(winbase.h)는 현재 프로세스에 대해 지정된 환경 변수의 내용을 설정합니다.
SetProcessAffinityMask 지정된 프로세스의 스레드에 대한 프로세서 선호도 마스크를 설정합니다.
SetTapeParameters 테이프의 블록 크기를 지정하거나 테이프 디바이스를 구성합니다.
SetTapePosition 지정된 디바이스에서 테이프 위치를 설정합니다.
SetThreadAffinityMask 지정된 스레드에 대한 프로세서 선호도 마스크를 설정합니다.

SetThreadpoolCallbackCleanupGroup 지정된 정리 그룹을 지정된 콜백 환경과 연결합니다. (SetThreadpoolCallbackCleanupGroup)
SetThreadpoolCallbackLibrary 미해결 콜백이 있는 한 지정된 DLL이 로드된 상태로 유지되도록 합니다. (SetThreadpoolCallbackLibrary)
SetThreadpoolCallbackPersistent 콜백이 영구 스레드에서 실행되도록 지정합니다. (SetThreadpoolCallbackPersistent)
SetThreadpoolCallbackPool 콜백을 생성할 때 사용할 스레드 풀을 설정합니다.
SetThreadpoolCallbackPriority 동일한 스레드 풀의 다른 작업 항목을 기준으로 콜백 함수의 우선 순위를 지정합니다. (SetThreadpoolCallbackPriority)
SetThreadpoolCallbackRunsLong 이 콜백 환경과 연결된 콜백이 빠르게 반환되지 않을 수 있음을 나타냅니다. (SetThreadpoolCallbackRunsLong)
SetUmsThreadInformation 지정된 UMS(사용자 모드 예약) 작업자 스레드에 대한 애플리케이션별 컨텍스트 정보를 설정합니다.
SwitchToFiber 파이버를 예약합니다. 파이버에서 함수를 호출해야 합니다.
TpDestroyCallbackEnviron 지정된 콜백 환경을 삭제합니다. 새 스레드 풀 개체를 만드는 데 콜백 환경이 더 이상 필요하지 않은 경우 이 함수를 호출합니다. (TpDestroyCallbackEnviron)
TpInitializeCallbackEnviron 스레드 풀에 대한 콜백 환경을 초기화합니다.
TpSetCallbackActivationContext 콜백 환경에 활성화 컨텍스트를 할당합니다.

TpSetCallbackCleanupGroup 지정된 정리 그룹을 지정된 콜백 환경과 연결합니다. (TpSetCallbackCleanupGroup)
TpSetCallbackFinalizationCallback 콜백 환경이 종료될 때 호출할 함수를 나타냅니다.
TpSetCallbackLongFunction 이 콜백 환경과 연결된 콜백이 빠르게 반환되지 않을 수 있음을 나타냅니다. (TpSetCallbackLongFunction)
TpSetCallbackNoActivationContext 콜백 환경에 활성화 컨텍스트가 없음을 나타냅니다.
TpSetCallbackPersistent 콜백이 영구 스레드에서 실행되도록 지정합니다. (TpSetCallbackPersistent)
TpSetCallbackPriority 동일한 스레드 풀의 다른 작업 항목을 기준으로 콜백 함수의 우선 순위를 지정합니다. (TpSetCallbackPriority)
TpSetCallbackRaceWithDll 미해결 콜백이 있는 한 지정된 DLL이 로드된 상태로 유지되도록 합니다. (TpSetCallbackRaceWithDll)
TpSetCallbackThreadpool 콜백 환경에 스레드 풀을 할당합니다.
UmsThreadYield 호출하는 UMS 작업자 스레드가 실행 중인 UMS(사용자 모드 예약) 스케줄러 스레드에 대한 제어를 생성합니다.
UnregisterWait RegisterWaitForSingleObject 함수에서 발급한 등록된 대기 작업을 취소합니다. (UnregisterWait)
WinExec 지정된 애플리케이션을 실행합니다.

WriteTapemark 지정된 수의 파일 표시, 집합 표시, 짧은 파일 표시 또는 긴 파일 표시를 테이프 디바이스에 씁니다.
YieldProcessor 대기 중인 스레드에 리소스를 제공하도록 프로세서에 신호를 전송합니다.

구조체

CACHE_DESCRIPTOR 캐시 특성을 설명합니다.
CACHE_RELATIONSHIP 캐시 특성을 설명합니다. 이 구조체는 GetLogicalProcessorInformationEx 함수와 함께 사용됩니다.
GROUP_AFFINITY 스레드의 선호도와 같은 프로세서 그룹별 선호도를 나타냅니다.
GROUP_RELATIONSHIP 프로세서 그룹에 대한 정보를 나타냅니다. 이 구조체는 GetLogicalProcessorInformationEx 함수와 함께 사용됩니다.
IO_COUNTERS 프로세스 또는 작업 개체에 대한 I/O 회계 정보를 포함합니다.
JOBOBJECT_ASSOCIATE_COMPLETION_PORT 완료 포트를 작업과 연결하는 데 사용되는 정보를 포함합니다.
JOBOBJECT_BASIC_ACCOUNTING_INFORMATION 작업 개체에 대한 기본 회계 정보를 포함합니다.
JOBOBJECT_BASIC_AND_IO_ACCOUNTING_INFORMATION 작업 개체에 대한 기본 회계 및 I/O 회계 정보를 포함합니다.

JOBOBJECT_BASIC_LIMIT_INFORMATION 작업 개체에 대한 기본 제한 정보를 포함합니다.
JOBOBJECT_BASIC_PROCESS_ID_LIST 작업 개체에 대한 프로세스 식별자 목록을 포함합니다.
JOBOBJECT_BASIC_UI_RESTRICTIONS 작업 개체에 대한 기본 사용자 인터페이스 제한을 포함합니다.
JOBOBJECT_CPU_RATE_CONTROL_INFORMATION 작업 개체에 대한 CPU 속도 제어 정보를 포함합니다. 이 구조체는 JobObjectCpuRateControlInformation 정보 클래스와 함께 SetInformationJobObject 및 QueryInformationJobObject 함수에서 사용됩니다.
JOBOBJECT_END_OF_JOB_TIME_INFORMATION 작업 종료 시간 제한을 초과할 때 시스템에서 수행할 작업을 지정합니다.
JOBOBJECT_EXTENDED_LIMIT_INFORMATION 작업 개체에 대한 기본 및 확장 제한 정보를 포함합니다.
JOBOBJECT_LIMIT_VIOLATION_INFORMATION 작업 개체에 대해 초과된 리소스 알림 제한에 대한 정보를 포함합니다. 이 구조체는 JobObjectLimitViolationInformation 정보 클래스와 함께 QueryInformationJobObject 함수와 함께 사용됩니다.
JOBOBJECT_LIMIT_VIOLATION_INFORMATION_2 작업 개체에 대해 초과된 리소스 알림 제한에 대한 확장된 정보를 포함합니다. 이 구조는 JobObjectLimitViolationInformation2 정보 클래스와 함께 QueryInformationJobObject 함수와 함께 사용됩니다.
JOBOBJECT_NET_RATE_CONTROL_INFORMATION 작업에 대한 네트워크 트래픽을 제어하는 데 사용되는 정보를 포함합니다. 이 구조체는 JobObjectNetRateControlInformation 정보 클래스와 함께 SetInformationJobObject 및 QueryInformationJobObject 함수에서 사용됩니다.
JOBOBJECT_NOTIFICATION_LIMIT_INFORMATION 작업 개체에 대한 알림 제한에 대한 정보를 포함합니다. 이 구조체는 JobObjectNotificationLimitInformation 정보 클래스와 함께 SetInformationJobObject 및 QueryInformationJobObject 함수에서 사용됩니다.

JOBOBJECT_NOTIFICATION_LIMIT_INFORMATION_2 작업 개체에 대한 알림 제한에 대한 확장된 정보를 포함합니다. 이 구조체는 JobObjectNotificationLimitInformation2 정보 클래스와 함께 SetInformationJobObject 및 QueryInformationJobObject 함수에서 사용됩니다.
JOBOBJECT_SECURITY_LIMIT_INFORMATION 작업 개체에 대한 보안 제한 사항을 포함합니다.
NUMA_NODE_RELATIONSHIP 프로세서 그룹의 NUMA 노드에 대한 정보를 나타냅니다. 이 구조체는 GetLogicalProcessorInformationEx 함수와 함께 사용됩니다.
PROCESS_MITIGATION_ASLR_POLICY ASLR(주소 공간 임의화 레이아웃)에 대한 프로세스 완화 정책 설정을 포함합니다.
PROCESS_MITIGATION_BINARY_SIGNATURE_POLICY 이미지의 서명에 따라 이미지 로드에 대한 프로세스 완화 정책 설정을 포함합니다.
PROCESS_MITIGATION_CONTROL_FLOW_GUARD_POLICY CFG(Control Flow Guard)에 대한 프로세스 완화 정책 설정을 포함합니다.
PROCESS_MITIGATION_DEP_POLICY DEP(데이터 실행 방지)에 대한 프로세스 완화 정책 설정을 포함합니다.
PROCESS_MITIGATION_DYNAMIC_CODE_POLICY 동적 코드 생성 및 수정을 제한하기 위한 프로세스 완화 정책 설정을 포함합니다.
PROCESS_MITIGATION_EXTENSION_POINT_DISABLE_POLICY 레거시 확장 지점 DLL에 대한 프로세스 완화 정책 설정을 포함합니다.
PROCESS_MITIGATION_FONT_DISABLE_POLICY 비 시스템 글꼴 로드에 대한 프로세스 완화 정책 설정을 포함합니다.
PROCESS_MITIGATION_IMAGE_LOAD_POLICY 원격 디바이스에서 이미지를 로드하기 위한 프로세스 완화 정책 설정을 포함합니다.

PROCESS_MITIGATION_REDIRECTION_TRUST_POLICY ???. 대한 프로세스 완화 정책 설정을 포함합니다.
PROCESS_MITIGATION_SIDE_CHANNEL_ISOLATION_POLICY 이 데이터 구조는 사이드 채널의 완화와 관련된 프로세스 정책의 상태를 제공합니다. 여기에는 투기적 실행 및 페이지 결합과 관련된 사이드 채널 공격이 포함될 수 있습니다.
PROCESS_MITIGATION_STRICT_HANDLE_CHECK_POLICY 유효하지 않은 핸들 참조에 새 동작을 적용하는 데 사용됩니다.
PROCESS_MITIGATION_SYSTEM_CALL_DISABLE_POLICY 프로세스에서 호출할 수 있는 시스템 호출에 제한을 적용하는 데 사용됩니다.
PROCESSOR_GROUP_INFO 프로세서 그룹의 프로세서 수와 선호도를 나타냅니다.
PROCESSOR_NUMBER 프로세서 그룹의 논리 프로세서를 나타냅니다.
PROCESSOR_RELATIONSHIP 프로세서 그룹 내의 선호도에 대한 정보를 나타냅니다. 이 구조체는 GetLogicalProcessorInformationEx 함수와 함께 사용됩니다.
SINGLE_LIST_ENTRY Singly 연결된 목록의 항목을 나타냅니다. S
SLIST_ENTRY Singly 연결된 목록의 항목을 나타냅니다. (SLIST_ENTRY)
STARTUPINFOEXA 새 프로세스의 창 스테이션, 데스크톱, 표준 핸들 및 특성을 지정합니다. CreateProcess 및 CreateProcessAsUser 함수와 함께 사용됩니다. (ANSI)
STARTUPINFOEXW 새 프로세스의 창 스테이션, 데스크톱, 표준 핸들 및 특성을 지정합니다. CreateProcess 및 CreateProcessAsUser 함수와 함께 사용됩니다. (유니코드)

SYSTEM_CPU_SET_INFORMATION 이 구조체는 GetSystemCpuSetInformation에서 반환됩니다. 시스템의 CPU 집합을 열거하고 현재 상태를 확인하는 데 사용됩니다.
SYSTEM_LOGICAL_PROCESSOR_INFORMATION 지정된 프로세서 집합 간의 관계를 설명합니다. 이 구조체는 GetLogicalProcessorInformation 함수와 함께 사용됩니다.
SYSTEM_LOGICAL_PROCESSOR_INFORMATION_EX 논리 프로세서 및 관련 하드웨어의 관계에 대한 정보를 포함합니다. GetLogicalProcessorInformationEx 함수는 이 구조를 사용합니다.
TAPE_ERASE 지울 파티션에 대해 설명합니다.
TAPE_GET_DRIVE_PARAMETERS 테이프 드라이브를 설명합니다. GetTapeParameters 함수에서 사용됩니다.
TAPE_GET_MEDIA_PARAMETERS 테이프 드라이브의 테이프에 대해 설명합니다. GetTapeParametersfunction에서 사용됩니다.
TAPE_GET_POSITION 테이프의 위치를 설명합니다.
TAPE_PREPARE 테이프를 준비하는 방법을 설명합니다.
TAPE_SET_DRIVE_PARAMETERS 테이프 드라이브를 설명합니다. SetTapeParametersfunction에서 사용됩니다.
TAPE_SET_MEDIA_PARAMETERS 테이프 드라이브의 테이프에 대해 설명합니다. SetTapeParametersfunction에서 사용됩니다.
TAPE_SET_POSITION 테이프의 위치와 위치를 설명합니다.

TAPE_WRITE_MARKS 쓸 테이프 표시의 유형과 수를 설명합니다.
UMS_CREATE_THREAD_ATTRIBUTES UMS(사용자 모드 예약) 작업자 스레드에 대한 특성을 지정합니다.
UMS_SCHEDULER_STARTUP_INFO UMS(사용자 모드 일정 예약) 스케줄러 스레드에 대한 특성을 지정합니다.
UMS_SYSTEM_THREAD_INFORMATION UMS 스케줄러 스레드, UMS 작업자 스레드 또는 비 UMS 스레드를 지정합니다. GetUmsSystemThreadInformation 함수는 이 구조를 사용합니다.
WIN32_STREAM_ID 스트림 데이터를 포함합니다.

배터리 장치 참조

아티클2024. 11. 18.

배터리 디바이스 참조 기술에 대한 개요입니다.

배터리 디바이스 참조를 개발하려면 다음 헤더가 필요합니다.

- batclass.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 배터리 장치 참조

함수

테이블 확장

BatteryClassInitializeDevice BatteryClassInitializeDevice 루틴은 클래스 드라이버에 대한 새 배터리 디바이스를 초기화합니다.
BatteryClassIoctl BatteryClassIoctl은 시스템 정의 배터리 IOCTL을 처리합니다.
BatteryClassQueryWmiDataBlock BatteryClassQueryWmiDataBlock 루틴은 배터리 클래스 드라이버가 드라이버를 대신하여 처리하는 WMI 데이터 블록 쿼리 요청을 처리할 수 있도록 DpWmiQueryDataBlock 루틴 내의 배터리 미니클래스 드라이버에서 사용됩니다.
BatteryClassStatusNotify BatteryClassStatusNotify는 배터리 클래스 드라이버에 배터리 상태 변경을 알립니다.
BatteryClassSystemControl BatteryClassSystemControl 루틴은 배터리 미니클래스 드라이버를 대신하여 WMI IRP를 처리합니다.
BatteryClassUnload BatteryClassUnload는 더 이상 사용되지 않는 배터리 디바이스에 대한 리소스를 해제합니다.

BatteryClassUpdateData BatteryClassUpdateData는 배터리 데이터가 변경될 때 클래스 드라이버에 알립니다. 이 메커니즘은 이벤트/상태를 보고하는 미니포트용입니다.
BCLASS_DISABLE_STATUS_NOTIFY_CALLBACK BatteryMiniDisableStatusNotify는 배터리 디바이스에 대한 상태 알림을 사용하지 않도록 설정합니다.
BCLASS_QUERY_INFORMATION_CALLBACK BatteryMiniQueryInformation은 지정된 배터리 디바이스에 대한 정보를 반환합니다.
BCLASS_QUERY_STATUS_CALLBACK BatteryMiniQueryStatus는 지정된 배터리 디바이스에 대한 상태 정보를 반환합니다.
BCLASS_QUERY_TAG_CALLBACK BatteryMiniQueryTag는 현재 배터리 태그를 반환합니다.
BCLASS_SET_INFORMATION_CALLBACK BatteryMiniSetInformation은 배터리가 충전 또는 방전 상태로 들어가거나 배터리에 대한 중요한 바이어스 값을 설정하라는 요청을 합니다.
BCLASS_SET_STATUS_NOTIFY_CALLBACK BatteryMiniSetStatusNotify는 클래스 드라이버에 알림이 필요한 배터리 용량 및 전원 상태 수준을 설정합니다.

구조

테이블 확장

BATTERY_MINIPORT_INFO 배터리 미니 클래스 드라이버는 배터리 클래스 드라이버의 BatteryClassInitializeDevice 루틴을 호출하기 전에이 구조를 채웁니다.
BATTERY_MINIPORT_INFO_V1_1 배터리 미니 클래스 드라이버는 배터리 클래스 드라이버의 BatteryClassInitializeDevice 루틴을 호

출하기 전에 BATTERY_MINIPORT_INFO_V1_1 구조를 채웁니다. BATTERY_MINIPORT_INFO_V1_1 이전 구조 BATTERY_MINIPORT_INFO 업데이트된 버전입니다.
BATTERY_NOTIFY 배터리 미니 클래스 드라이버는 BatteryMiniSetStatusNotify 루틴이 호출될 때 BATTERY_NOTIFY 구조를 받습니다.
BATTERY_TAG_CHANGE 이 구조는 시스템 사용을 위해 예약되어 있습니다. (BATTERY_TAG_CHANGE)
BATTERY_WMI_CYCLE_COUNT BatteryClassQueryWmiDataBlock 함수에 사용할 배터리의 충전 주기 수에 대한 정보를 정의합니다.
BATTERY_WMI_FULL_CHARGED_CAPACITY BatteryClassQueryWmiDataBlock과 함께 사용할 배터리 용량에 대한 정보를 정의합니다.)
BATTERY_WMI_RUNTIME BatteryClassQueryWmiDataBlock 함수에 사용할 배터리의 예상 런타임에 대한 정보를 정의합니다.
BATTERY_WMI_STATIC_DATA 배터리에 대한 정적 데이터를 정의합니다.
BATTERY_WMI_STATUS 배터리 상태 정보를 정의합니다.
BATTERY_WMI_STATUS_CHANGE 이 구조는 시스템 사용을 위해 예약되어 있습니다. (BATTERY_WMI_STATUS_CHANGE)
BATTERY_WMI_TEMPERATURE BatteryClassQueryWmiDataBlock 함수와 함께 사용할 배터리 온도에 대한 정보를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

Bluetooth 디바이스 참조

아티클2023. 08. 22.

Bluetooth 디바이스 참조 기술에 대한 개요입니다.

Bluetooth 디바이스 참조를 개발하려면 다음 헤더가 필요합니다.

- bluetoothleapis.h
- bthledef.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Bluetooth 디바이스 참조

GUID

헤더 파일에 다음 GUID가 `bthledef.h` 정의되어 있습니다. 페어링된 Bluetooth LE 디바이스를 열거하려면 SetupDiXxx Win32 열거형 API를 사용하여 **GUID_BLUETOOTHLE_DEVICE_INTERFACE** 디바이스 인터페이스 클래스의 디바이스를 열거할 수 있습니다.

GUID 이름, 설명	값
GUID_BLUETOOTHLE_DEVICE_INTERFACE. Bluetooth LE 디바이스 인터페이스 GUID입니다.	0x781aee18, 0x7733, 0x4ce4, 0xad, 0xd0, 0x91, 0xf4, 0x1c, 0x67, 0xb5, 0x92
GUID_BLUETOOTH_GATT_SERVICE_DEVICE_INTERFACE. Bluetooth LE 서비스 디바이스 인터페이스 GUID.	0x6e3bb679, 0x4372, 0x40c8, 0x9e, 0xaa, 0x45, 0x09, 0xdf, 0x26, 0x0c, 0xd8
BTH_LE_ATT_BLUETOOTH_BASE_GUID. Bluetooth 기본 GUID입니다.	0x00000000, 0x0000, 0x1000, 0x80, 0x00, 0x00, 0x80, 0x5F, 0x9B, 0x34, 0xFB

열거형

BTH_LE_GATT_DESCRIPTOR_TYPE

BTH_LE_GATT_DESCRIPTOR_TYPE 열거형은 다양한 유형의 Bluetooth LE GATT(제네릭 특성)를 설명합니다.
BTH_LE_GATT_EVENT_TYPE BTH_LE_GATT_EVENT_TYPE 열거형은 다양한 유형의 Bluetooth LE(저에너지) GATT(제네릭 특성) 프로필 이벤트를 설명합니다.

Functions

BluetoothGATTAbortReliableWrite 신뢰할 수 있는 쓰기 프로시저의 끝을 지정하고 쓰기를 중단해야 합니다.
BluetoothGATTBeginReliableWrite BluetoothGATTBeginReliableWrite 함수는 신뢰할 수 있는 쓰기가 시작되도록 지정합니다.
BluetoothGATTEndReliableWrite 신뢰할 수 있는 쓰기의 끝을 지정하고 쓰기를 커밋해야 합니다.
BluetoothGATTGetCharacteristics 지정된 서비스에 사용할 수 있는 모든 특성을 가져옵니다.
BluetoothGATTGetCharacteristicValue 지정된 특성의 값을 가져옵니다.
BluetoothGATTGetDescriptors 지정된 특성에 사용할 수 있는 모든 설명자를 가져옵니다.
BluetoothGATTGetDescriptorValue 지정된 설명자의 값을 가져옵니다.
BluetoothGATTGetIncludedServices 지정된 서비스에 사용할 수 있는 모든 포함된 서비스를 가져옵니다.
BluetoothGATTGetServices BluetoothGATTGetServices 함수는 서버에 사용할 수 있는 모든 기본 서비스를 가져옵니다.

BluetoothGATTRegisterEvent 특성 핸들로 식별된 지정된 특성에 대한 특성 값 변경 이벤트 중에 다시 호출할 루틴을 등록합니다.
BluetoothGATTSetCharacteristicValue 지정된 특성 값을 Bluetooth 디바이스에 씁니다.
BluetoothGATTSetDescriptorValue 지정한 설명자 값을 Bluetooth 디바이스에 씁니다.
BluetoothGATTUnregisterEvent 지정된 특성 값 변경 이벤트를 등록 취소합니다.
BluetoothSetLocalServiceInfo 특정 Bluetooth 라디오에 대한 로컬 서비스 정보를 설정합니다.
IsBthLEUuidMatch 두 UUID가 서로 일치하는지 여부를 결정합니다.
PFNBLUETOOTH_GATT_EVENT_CALLBACK 프로필 드라이버는 특정 특성의 값이 변경될 때마다 호출할 Bluetooth GATT 이벤트 콜백을 구현합니다.

구조체

BLUETOOTH_GATT_VALUE_CHANGED_EVENT BLUETOOTH_GATT_VALUE_CHANGED_EVENT 구조체는 변경된 특성 값을 설명합니다.
BLUETOOTH_GATT_VALUE_CHANGED_EVENT_REGISTRATION BLUETOOTH_GATT_VALUE_CHANGED_EVENT_REGISTRATION 구조체는 변경된 하나 이상의 특성을 설명합니다.
BTH_LE_GATT_CHARACTERISTIC BTH_LE_GATT_CHARACTERISTIC 구조는 Bluetooth LE(저에너지) GATT(제네릭 특성) 프로필 특성을 설명합니다.

BTH_LE_GATT_CHARACTERISTIC_VALUE BTH_LE_GATT_CHARACTERISTIC_VALUE 구조는 Bluetooth LE(저에너지) GATT(제네릭 특성) 프로필 특성 값을 설명합니다.
BTH_LE_GATT_DESCRIPTOR BTH_LE_GATT_DESCRIPTOR 구조는 Bluetooth LE(저에너지) GATT(제네릭 특성) 프로필 설명자를 설명합니다.
BTH_LE_GATT_DESCRIPTOR_VALUE BTH_LE_GATT_DESCRIPTOR_VALUE 구조체는 부모 특성을 설명합니다.
BTH_LE_GATT_SERVICE BTH_LE_GATT_SERVICE 구조는 Bluetooth LE(저에너지) GATT(제네릭 특성) 프로필 서비스를 설명합니다.
BTH_LE_UUID BTH_LE_UUID 구조에는 BLUEtooth LE(저에너지) UUID(유니버설 고유 식별자)에 대한 정보가 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Bluetooth

아티클2023. 03. 11.

Bluetooth 기술 개요입니다.

Bluetooth를 개발하려면 다음 헤더가 필요합니다.

- bluetoothapis.h
- bthdef.h
- bthsdpdef.h
- ws2bth.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Bluetooth

열거형

BLUETOOTH_AUTHENTICATION_METHOD BLUETOOTH_AUTHENTICATION_METHOD 열거형은 디바이스 페어링 중에 지원되는 인증 유형을 정의합니다.
BLUETOOTH_AUTHENTICATION_REQUIREMENTS BLUETOOTH_AUTHENTICATION_REQUIREMENTS 열거형은 인증에 필요한 '중간의 남자' 보호를 지정합니다.
BLUETOOTH_IO_CAPABILITY BLUETOOTH_IO_CAPABILITY 열거형은 Bluetooth 디바이스의 입력/출력 기능을 정의합니다.

Functions

BluetoothAuthenticateDevice 원격 Bluetooth 디바이스에 인증 요청을 보냅니다.

BluetoothAuthenticateDeviceEx BluetoothAuthenticateDeviceEx 함수는 원격 Bluetooth 디바이스에 인증 요청을 보냅니다.
BluetoothAuthenticateMultipleDevices 호출자가 Bluetooth 연결 마법사의 단일 인스턴스에서 여러 디바이스를 인증할지 묻는 메시지를 표시할 수 있도록 합니다.
BluetoothDisplayDeviceProperties 디바이스 정보 속성 시트를 제어판 시작합니다.
BluetoothEnableDiscovery BluetoothEnableDiscovery 함수는 로컬 Bluetooth 라디오 또는 라디오의 검색 상태를 변경합니다.
BluetoothEnableIncomingConnections BluetoothEnableIncomingConnections 함수는 로컬 Bluetooth 라디오가 들어오는 연결을 허용하는지 여부를 수정합니다.
BluetoothEnumerateInstalledServices BluetoothEnumerateInstalledServices 함수는 Bluetooth 디바이스에서 사용하도록 설정된 서비스 GUID(Globally Unique Identifiers)를 열거합니다.
BluetoothFindDeviceClose BluetoothFindDeviceClose 함수는 디바이스 쿼리와 연결된 열거형 핸들을 닫습니다.
BluetoothFindFirstDevice BluetoothFindFirstDevice 함수는 열거형 Bluetooth 디바이스를 시작합니다.
BluetoothFindFirstRadio BluetoothFindFirstRadio 함수는 로컬 Bluetooth 라디오의 열거를 시작합니다.
BluetoothFindNextDevice BluetoothFindNextDevice 함수는 다음 Bluetooth 디바이스를 찾습니다.
BluetoothFindNextRadio BluetoothFindNextRadio 함수는 다음 Bluetooth 라디오를 찾습니다.

BluetoothFindRadioClose BluetoothFindRadioClose 함수는 Bluetooth 라디오 찾기와 관련된 열거형 핸들을 닫습니다.
BluetoothGetDeviceInfo 원격 Bluetooth 디바이스에 대한 정보를 검색합니다.
BluetoothGetRadioInfo Bluetooth 라디오에 대한 정보를 가져옵니다.
BluetoothIsConnectable BluetoothIsConnectable 함수는 Bluetooth 라디오 또는 라디오를 연결할 수 있는지 여부를 결정합니다.
BluetoothIsDiscoverable BluetoothIsDiscoverable 함수는 Bluetooth 라디오 또는 라디오를 검색할 수 있는지 여부를 결정합니다.
BluetoothIsVersionAvailable BluetoothIsVersionAvailable 함수는 설치된 Bluetooth 이진 집합이 요청된 버전을 지원하는지 나타냅니다.
BluetoothRegisterForAuthentication BluetoothRegisterForAuthentication 함수는 특정 Bluetooth 디바이스가 인증을 요청할 때 호출되는 콜백 함수를 등록합니다.
BluetoothRegisterForAuthenticationEx BluetoothRegisterForAuthenticationEx 함수는 핀 요청, 숫자 비교 및 콜백 함수에 대한 애플리케이션을 등록합니다.
BluetoothRemoveDevice Bluetooth 디바이스와 컴퓨터 간의 인증을 제거하고 디바이스에 대한 캐시된 서비스 정보를 지웁니다.
BluetoothSdpEnumAttributes BluetoothSdpEnumAttributes 함수는 SDP 레코드 스트림을 통해 열거되고 레코드의 각 특성에 대한 콜백 함수를 호출합니다.

BluetoothSdpGetAttributeValue BluetoothSdpGetAttributeValue 함수는 특성 식별자에 대한 특성 값을 검색합니다.
BluetoothSdpGetContainerElementData 컨테이너 스트림을 반복하고 컨테이너 요소 내에 포함된 각 요소를 반환합니다.
BluetoothSdpGetElementData SDP 스트림에서 단일 요소를 검색하고 구문 분석합니다.
BluetoothSdpGetString SDP 레코드에 포함된 원시 문자열을 유니코드 문자열로 변환합니다.
BluetoothSelectDevices Bluetooth 디바이스를 선택할 수 있습니다.
BluetoothSelectDevicesFree BluetoothSelectDevices에 대한 이전 호출과 관련된 리소스를 해제합니다.
BluetoothSendAuthenticationResponse BluetoothSendAuthenticationResponse 함수는 암호 응답을 보내는 인증 요청을 받을 때 호출됩니다.
BluetoothSendAuthenticationResponseEx BluetoothSendAuthenticationResponseEx 함수는 암호 또는 숫자 비교 응답을 보내는 인증 요청이 수행되면 호출됩니다.
BluetoothSetServiceState Bluetooth 디바이스에 대한 서비스를 사용하거나 사용하지 않도록 설정합니다.
BluetoothUnregisterAuthentication BluetoothUnregisterAuthentication 함수는 이전에 BluetoothRegisterForAuthentication 함수에 대한 호출로 등록된 콜백 루틴에 대한 등록을 제거합니다.
BluetoothUpdateDeviceRecord Bluetooth 디바이스에 대한 로컬 컴퓨터 캐시를 업데이트.

PFN_AUTHENTICATION_CALLBACK BluetoothRegisterForAuthentication 함수와 함께 사용됩니다.
PFN_AUTHENTICATION_CALLBACK_EX PFN_AUTHENTICATION_CALLBACK_EX 함수는 BluetoothRegisterForAuthenticationEx 함수와 함께 사용되는 콜백 함수 프로토타입입니다.
PFN_BLUETOOTH_ENUM_ATTRIBUTES_CALLBACK BluetoothSdpEnumAttributes 함수 호출에 전달된 pSDPStream 매개 변수에 있는 각 특성에 대해 한 번씩 호출되는 콜백 함수 프로토타입입니다.
PFN_DEVICE_CALLBACK Bluetooth 디바이스 선택과 관련하여 사용되는 콜백 프로토타입입니다.

구조체

BLUETOOTH_ADDRESS_STRUCT BLUETOOTH_ADDRESS 구조는 Bluetooth 디바이스의 주소를 제공합니다.
BLUETOOTH_AUTHENTICATE_RESPONSE BLUETOOTH_AUTHENTICATE_RESPONSE 구조에는 BTH_REMOTE_AUTHENTICATE_REQUEST 이벤트에 대한 응답으로 전달된 정보가 포함됩니다.
BLUETOOTH_AUTHENTICATION_CALLBACK_PARAMS BLUETOOTH_AUTHENTICATION_CALLBACK_PARAMS 구조에는 인증 요청에 응답하는 Bluetooth 디바이스에 대한 특정 구성 정보가 포함됩니다.
BLUETOOTH_COD_PAIRS BLUETOOTH_COD_PAIRS 구조는 COD(Bluetooth 디바이스 클래스) 정보의 사양 및 검색을 제공합니다.
BLUETOOTH_DEVICE_INFO_STRUCT BLUETOOTH_DEVICE_INFO 구조는 Bluetooth 디바이스에 대한 정보를 제공합니다.

BLUETOOTH_DEVICE_SEARCH_PARAMS Bluetooth 디바이스 검색에 대한 검색 조건을 지정합니다.
BLUETOOTH_FIND_RADIO_PARAMS BLUETOOTH_FIND_RADIO_PARAMS 구조는 설치된 Bluetooth 라디오를 열거하는 데 도움이 됩니다.
BLUETOOTH_LOCAL_SERVICE_INFO_STRUCT Bluetooth 디바이스에 대한 로컬 서비스 정보를 포함합니다.
BLUETOOTH_NUMERIC_COMPARISON_INFO BLUETOOTH_NUMERIC_COMPARISON_INFO 구조체에는 숫자 비교를 통해 인증에 사용되는 숫자 값이 포함됩니다.
BLUETOOTH_OOB_DATA_INFO BLUETOOTH_OOB_DATA_INFO 구조에는 대역 외 디바이스 페어링을 설정하기 전에 인증하는 데 사용되는 데이터가 포함됩니다.
BLUETOOTH_PASSKEY_INFO BLUETOOTH_PASSKEY_INFO 구조에는 인증에 사용되는 암호 값이 포함되어 있습니다. 암호는 암호와 비슷하지만 암호 값은 인증에 한 번만 사용됩니다.
BLUETOOTH_PIN_INFO BLUETOOTH_PIN_INFO 구조에는 PIN을 통한 인증에 사용되는 정보가 포함되어 있습니다.
BLUETOOTH_RADIO_INFO Bluetooth 라디오에 대한 정보를 포함합니다.
BLUETOOTH_SELECT_DEVICE_PARAMS Bluetooth 디바이스 및 서비스의 표시 유형, 인증 및 선택을 용이하게 하고 관리합니다.
BTH_DEVICE_INFO Bluetooth 디바이스에 대한 정보를 저장합니다.
BTH_HCI_EVENT_INFO Bluetooth에 대한 WM_DEVICECHANGE 메시지 가져오기와 관련하여 사용됩니다.

BTH_L2CAP_EVENT_INFO L2CAP 채널과 연결된 이벤트에 대한 데이터를 포함합니다.
BTH_QUERY_DEVICE BTH_QUERY_DEVICE 구조는 Bluetooth 디바이스의 존재를 쿼리할 때 사용됩니다.
BTH_QUERY_SERVICE BTH_QUERY_SERVICE 구조는 Bluetooth 서비스를 쿼리하는 데 사용됩니다.
BTH_RADIO_IN_RANGE Bluetooth 디바이스에 대한 데이터를 통신 범위 내에 저장합니다.
BTH_SET_SERVICE 지정된 Bluetooth 서비스에 대한 서비스 정보를 제공합니다.
SDP_ELEMENT_DATA SDP_ELEMENT_DATA 구조체는 SDP 요소 데이터를 저장합니다.
SDP_STRING_TYPE_DATA SDP_STRING_TYPE_DATA 구조체는 SDP 문자열 형식에 대한 정보를 저장합니다.
SdpAttributeRange SdpAttributeRange 구조는 Bluetooth 쿼리에서 쿼리에서 반환할 특성 집합을 제한하는 데 사용됩니다.
SdpQueryUuid SdpQueryUuid 구조는 UUID를 쉽게 검색할 수 있도록 합니다.
SdpQueryUuidUnion SdpQueryUuidUnion 공용 구조체에는 SDP 쿼리를 수행할 UUID가 포함되어 있습니다. SdpQueryUuid 구조체와 함께 사용됩니다.
SOCKADDR_BTH SOCKADDR_BTH 구조는 주소 패밀리 AF_BTH 정의된 Bluetooth 소켓 작업과 함께 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

CimFS

아티클2023. 03. 09.

CIM은 WIM과 개념이 유사한 파일 지원 이미지 형식입니다.

CIM 형식은 하나 이상의 데이터 및 메타데이터 지역 파일, 하나 이상의 개체 ID 파일 및 하나 이상의 파일 시스템 설명 파일을 포함하는 플랫 파일의 작은 컬렉션으로 구성됩니다. "평탄성" CIM의 결과로 포함된 동일한 원시 디렉터리보다 더 빠르게 생성, 추출 및 삭제할 수 있습니다.

CIM은 지정된 이미지에 동일한 데이터 영역 백업 파일을 공유하면서 개별적으로 탑재할 수 있는 여러 파일 시스템 볼륨을 포함할 수 있다는 점에서 복합입니다.

구성되면 CimFS 드라이버를 지원하여 CIM을 탑재할 수 있습니다. 탑재는 이미지에 대한 읽기 전용 디스크 및 파일 시스템 볼륨 디바이스를 생성합니다. 탑재된 CIM의 콘텐츠는 표준 Win32 또는 NT API 파일 시스템 인터페이스를 사용하여 읽기 전용으로 액세스할 수 있습니다. CimFS 파일 시스템은 보안 설명자, 대체 데이터 스트림, 하드 링크 및 다시 구문 분석 지점과 같은 NTFS의 많은 구문을 지원합니다.

CIM은 파일 수준에서 중복 제거를 지원합니다. 동일한 파일의 여러 복사본이 다른 경로를 사용하여 CIM에 추가되는 경우 CIM에 저장된 파일 데이터의 복사본이 하나만 있습니다.

CIM은 원래 Windows 컨테이너 이미지 레이아웃으로 사용하도록 설계 및 최적화되었습니다.

CimFS를 사용하여 개발하려면 다음 헤더가 필요합니다.

- cimfs.h

이 라이브러리와 연결해야 합니다.

- cimfs.lib

이 샘플 에서는 CimFS(복합 이미지 파일 시스템) API를 사용하여 CimFS 이미지를 만들고 구성하고 조작하는 방법을 보여 줍니다.

열거형

CIM_MOUNT_IMAGE_FLAGS CIM_MOUNT_IMAGE_FLAGS 열거형에는 CimMountImage의 동작을 수정하는 데 사용할 수 있는 플래그가 포함되어 있습니다.

Functions

CimCloseImage CimCloseImage 함수는 이미지 핸들과 연결된 리소스를 해제합니다.
CimCloseStream CimCloseStream 함수는 스트림 핸들과 연결된 리소스를 해제합니다.
CimCommitImage CimCommitImage 함수는 이미지 핸들이 나타내는 이미지를 커밋합니다.
CimCreateAlternateStream CimCreateAlternateStream 함수는 이미지 핸들이 나타내는 이미지를 기준으로 지정된 크기의 대체 스트림을 경로에 추가합니다.
CimCreateFile CimCreateFile 함수는 이미지 핸들이 나타내는 이미지를 기준으로 경로에 지정된 메타데이터를 사용하여 새 파일 또는 디렉터리를 추가합니다.
CimCreateHardLink CimCreateHardLink 함수는 이미지 핸들이 나타내는 이미지를 기준으로 기존 경로에 하드 링크를 추가합니다.
CimCreateImage CimCreateImage 함수는 지정된 위치에 새 이미지를 나타내는 핸들을 만듭니다. 선택적으로 해당 위치의 기존 이미지를 기반으로 합니다.
CimDeletePath CimDeletePath 함수는 이미지 핸들이 나타내는 이미지를 기준으로 경로에서 파일, 스트림, 디렉터리 또는 하드링크를 제거합니다.

CimDismountImage CimDismountImage 함수는 volumeId로 탑재된 이미지를 볼륨 GUID로 분리합니다.
CimMountImage CimMountImage 함수는 cimPath에 지정된 위치에서 명명된 이미지를 volumeId로 지정된 볼륨 GUID가 있는 볼륨으로 탑재합니다.
CimWriteStream CimWriteStream 함수는 지정된 버퍼의 데이터를 스트림 핸들이 나타내는 스트림에 씁니다.

구조체

CIMFS_FILE_METADATA CIMFS_FILE_METADATA 구조체는 CimCreateFile에서 추가할 파일에 대한 파일 메타데이터를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

CIMWin32 WMI 공급자

아티클2023. 03. 16.

CIMWin32 WMI 공급자 기술 개요입니다.

CIMWin32 WMI 공급자를 개발하려면 다음 헤더가 필요합니다.

- sdoias.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- CIMWin32 WMI 공급자

클라우드 필터 API

아티클2023. 03. 09.

클라우드 필터 API 기술 개요입니다.

클라우드 필터 API를 개발하려면 다음 헤더가 필요합니다.

- cfapi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 클라우드 필터 API

열거형

CF_CALLBACK_CANCEL_FLAGS 자리 표시자 파일 또는 폴더에 대한 데이터 가져오기를 취소하기 위한 콜백 플래그입니다.
CF_CALLBACK_CLOSE_COMPLETION_FLAGS 이전에 읽기/쓰기/삭제 액세스를 위해 연 동기화 루트 중 하나에 있는 자리 표시자가 이제 닫혀 있음을 동기화 공급자에 알리기 위한 콜백 플래그입니다.
CF_CALLBACK_DEHYDRATE_COMPLETION_FLAGS 동기화 루트 중 하나에 있는 자리 표시자가 성공적으로 탈수되었음을 동기화 공급자에게 알리는 콜백 플래그입니다.
CF_CALLBACK_DEHYDRATE_FLAGS 동기화 공급자에게 동기화 루트 중 하나의 자리 표시자가 탈수될 것임을 알리는 콜백 플래그입니다.
CF_CALLBACK_DEHYDRATION_REASON 클라우드 파일이 탈수된 이유를 지정합니다.
CF_CALLBACK_DELETE_COMPLETION_FLAGS 자리 표시자가 성공적으로 삭제되었음을 동기화 공급자에 알리기 위한 콜백 플래그입니다.

CF_CALLBACK_DELETE_FLAGS 이 콜백은 동기화 공급자에게 동기화 루트 중 하나에 있는 자리 표시자 파일 또는 디렉터리가 삭제되거나 삭제되지 않을 것임을 알리는 데 사용됩니다.
CF_CALLBACK_FETCH_DATA_FLAGS 자리 표시자 파일 또는 폴더에 대한 데이터를 가져오기 위한 콜백 플래그입니다.
CF_CALLBACK_FETCH_PLACEHOLDERS_FLAGS 자리 표시자 파일 또는 디렉터리의 콘텐츠에 대한 정보를 가져오기 위한 플래그입니다.
CF_CALLBACK_OPEN_COMPLETION_FLAGS 읽기/쓰기/삭제 액세스를 위해 자리 표시자가 성공적으로 열렸다는 것을 동기화 공급자에 알리기 위한 콜백 플래그입니다.
CF_CALLBACK_RENAME_COMPLETION_FLAGS 동기화 루트 중 하나에 있는 자리 표시자의 이름이 성공적으로 변경되었음을 동기화 공급자에 알리는 콜백 플래그입니다.
CF_CALLBACK_RENAME_FLAGS 플래그를 다시 호출하여 동기화 루트 중 하나에 있는 자리 표시자의 이름을 바꾸거나 이동하려고 한다고 동기화 공급자에게 알릴 수 있습니다.
CF_CALLBACK_TYPE 자리 표시자 파일 또는 폴더에 사용되는 다양한 유형의 콜백을 포함합니다.
CF_CALLBACK_VALIDATE_DATA_FLAGS 자리 표시자 파일 또는 디렉터리의 데이터의 유효성을 검사하는 플래그입니다.
CF_CONNECT_FLAGS 콜백이 호출될 때 동기화 공급자가 요청할 수 있는 추가 정보입니다.
CF_CONVERT_FLAGS 자리 표시자 파일/디렉터리 변환 플래그에 대한 일반 파일/디렉터리입니다.
CF_CREATE_FLAGS 자리 표시자 파일 또는 디렉터리를 만들기 위한 플래그입니다.

CF_DEHYDRATE_FLAGS 자리 표시자 탈수 플래그입니다.
CF_HARDLINK_POLICY 자리 표시자 파일에서 하드 링크가 허용되는지 여부를 지정합니다.
CF_HYDRATE_FLAGS 자리 표시자 데이터 하이드레이션 플래그입니다.
CF_HYDRATION_POLICY_MODIFIER 동기화 공급자가 플랫폼에서 자리 표시자 파일을 하이드레이션하는 방법을 제어할 수 있습니다. 기본 정책:_CF_HYDRATION_POLICY_PRIMARY 사용할 수 있는 한정자입니다.
CF_HYDRATION_POLICY_PRIMARY 동기화 공급자가 플랫폼에서 자리 표시자 파일을 하이드레이션하는 방법을 제어할 수 있습니다. 이것이 기본 정책입니다.
CF_IN_SYNC_STATE 자리 표시자 파일 및 폴더의 동기화 중 상태를 지정합니다.
CF_INSYNC_POLICY 플랫폼이 자리 표시자 파일 또는 디렉터리에서 동기화 중 상태를 지워야 하는 시기를 동기화 공급자가 제어할 수 있도록 하는 정책입니다.
CF_OPEN_FILE_FLAGS 파일 열기에 대한 다양한 권한을 요청하는 플래그입니다.
CF_OPERATION_ACK_DATA_FLAGS 자리 표시자 파일 또는 폴더에 대한 데이터를 확인하고 승인하는 플래그입니다.
CF_OPERATION_ACK_DEHYDRATE_FLAGS 자리 표시자 파일 또는 디렉터리의 탈수를 승인하는 플래그입니다.
CF_OPERATION_ACK_DELETE_FLAGS 자리 표시자 파일 또는 디렉터리의 삭제를 승인하는 플래그입니다.

CF_OPERATION_ACK_RENAME_FLAGS 자리 표시자 파일 또는 디렉터리 이름 바꾸기를 승인하기 위한 플래그입니다.
CF_OPERATION_RESTART_HYDRATION_FLAGS 자리 표시자 파일 또는 폴더에서 데이터 하이드레이션을 다시 시작하는 플래그입니다.
CF_OPERATION_RETRIEVE_DATA_FLAGS 자리 표시자 파일 또는 폴더에 대한 데이터를 검색하는 플래그입니다.
CF_OPERATION_TRANSFER_DATA_FLAGS 자리 표시자 파일 또는 폴더를 수화하기 위해 데이터를 전송하는 플래그입니다.
CF_OPERATION_TRANSFER_PLACEHOLDERS_FLAGS 자리 표시자 파일 또는 디렉터리를 전송할 때 동작을 지정하는 플래그입니다.
CF_OPERATION_TYPE 자리 표시자 파일 및 디렉터리에서 수행할 수 있는 작업 유형입니다.
CF_PIN_STATE 자리 표시자 파일 또는 디렉터리의 상태를 고정합니다.
CF_PLACEHOLDER_CREATE_FLAGS 자리 표시자별로 자리 표시자를 만들기 위한 플래그입니다.
CF_PLACEHOLDER_INFO_CLASS 자리 표시자 정보에 대한 정보 클래스입니다.
CF_PLACEHOLDER_MANAGEMENT_POLICY CF_SYNC_POLICIES 구조체에 대한 자리 표시자 관리 정책을 지정합니다.
CF_PLACEHOLDER_RANGE_INFO_CLASS 자리 표시자 파일 데이터 범위의 형식입니다.
CF_PLACEHOLDER_STATE 자리 표시자 파일 또는 폴더의 상태입니다.

CF_POPULATION_POLICY_MODIFIER 모집단 정책 한정자를 정의합니다. 기본 정책:_CF_POPULATION_POLICY_PRIMARY 사용할 수 있는 한정자입니다.
CF_POPULATION_POLICY_PRIMARY 동기화 공급자가 플랫폼에서 자리 표시자 디렉터리 및 파일을 만드는 방법을 제어할 수 있습니다. 이것이 기본 정책입니다.
CF_REGISTER_FLAGS 동기화 루트를 등록하고 업데이트하기 위한 플래그입니다.
CF_REVERT_FLAGS 자리 표시자 파일을 일반 파일로 되돌리기 위한 플래그입니다.
CF_SET_IN_SYNC_FLAGS 자리 표시자 파일 및 폴더에 대한 동기화 중인 상태 플래그입니다.
CF_SET_PIN_FLAGS 자리 표시자 핀 플래그입니다.
CF_SYNC_PROVIDER_STATUS 동기화 공급자의 현재 상태입니다.
CF_SYNC_ROOT_INFO_CLASS 동기화 루트 정보의 유형입니다.
CF_UPDATE_FLAGS 자리 표시자 파일 또는 디렉터리를 업데이트하기 위한 플래그입니다.

Functions

CfCloseHandle CfOpenFileWithOplock에서 반환된 파일 또는 디렉터리 핸들을 닫습니다. CfApi.h 내에서 사용되는 핸들에서만 표준 Win32 파일 핸들과 함께 사용하면 안 됩니다.

CfConnectSyncRoot 동기화 공급자와 동기화 필터 API 간의 양방향 통신을 시작합니다.
CfConvertToPlaceholder 일반 파일/디렉터리를 자리 표시자 파일/디렉터리로 변환합니다.
CfCreatePlaceholders 동기화 루트 트리 아래에 하나 이상의 새 자리 표시자 파일 또는 디렉터리를 만듭니다.
CfDisconnectSyncRoot CfConnectSyncRoot에서 만든 통신 채널의 연결을 끊습니다.
CfExecute 모든 연결 키 기반 자리 표시자 작업의 기본 진입점입니다. 동기화 공급자가 플랫폼의 다양한 콜백에 응답하는 데 사용됩니다.
CfGetCorrelationVector 동기화 공급자가 지정된 자리 표시자 파일에 대한 현재 상관 관계 벡터를 쿼리할 수 있습니다.
CfGetPlaceholderInfo 자리 표시자 파일 또는 폴더의 다양한 특성을 가져옵니다.
CfGetPlaceholderRangeInfo 자리 표시자 파일 또는 폴더에 대한 범위 정보를 가져옵니다.
CfGetPlaceholderRangeInfoForHydration ConnectionKey, TransferKey 및 FileId를 식별자로 사용하여 자리 표시자 파일 또는 폴더에 대한 범위 정보를 가져옵니다.
CfGetPlaceholderStateFromAttributeTag 파일의 FileAttributes 및 ReparseTag 값을 기반으로 자리 표시자 상태 집합을 가져옵니다.
CfGetPlaceholderStateFromFileInfo 파일의 다양한 정보를 기반으로 자리 표시자 상태 집합을 가져옵니다.
CfGetPlaceholderStateFromFindData WIN32_FIND_DATA 구조체에 따라 자리 표시자 상태 집합을 가져옵니다.

CfGetPlatformInfo 플랫폼 버전 정보를 가져옵니다.
CfGetSyncRootInfoByHandle 파일 핸들에 지정된 지정된 파일을 포함하는 동기화 루트의 다양한 특성을 가져옵니다.
CfGetSyncRootInfoByPath 동기화 루트 아래의 파일에 지정된 다양한 동기화 루트 정보를 가져옵니다.
CfGetTransferKey 자리 표시자 파일 또는 폴더로 데이터 전송을 시작합니다.
CfGetWin32HandleFromProtectedHandle 모든 핸들 기반 Win32 API와 함께 사용할 수 있도록 보호된 핸들을 Win32 핸들로 변환합니다.
CfHydratePlaceholder 지정된 바이트 범위가 자리 표시자에 디스크에 있는지 확인하여 자리 표시자 파일을 수화합니다. 파일에만 유효합니다.
CfOpenFileWithOplock 파일 또는 디렉터리(일반 및 자리 표시자 파일 모두)에 대한 비동기 불투명 핸들을 열고 열린 플래그에 따라 적절한 oplock을 설정합니다.
CfQuerySyncProviderStatus 동기화 공급자를 쿼리하여 공급자의 상태를 가져옵니다.
CfReferenceProtectedHandle 호출자가 보호된 핸들을 비 CfApi Win32 API와 함께 사용할 수 있는 Win32 핸들에 참조할 수 있도록 허용합니다.
CfRegisterSyncRoot 일회성 동기화 루트 등록을 수행합니다.
CfReleaseProtectedHandle CfReferenceProtectedHandle에서 참조하는 보호된 핸들을 해제합니다.

CfReleaseTransferKey CfGetTransferKey에서 가져온 전송 키를 해제합니다.
CfReportProviderProgress 동기화 공급자가 대역 외 진행 상황을 보고할 수 있습니다.
CfReportProviderProgress2 동기화 공급자가 대역 외 진행 상황을 보고할 수 있습니다. 추가 매개 변수를 사용하여 CfReportProviderProgress를 확장합니다.
CfReportSyncStatus 먼저 CfConnectSyncRoot에 대한 호출과 연결하지 않고도 동기화 공급자가 지정된 동기화 루트에서 해당 상태를 플랫폼에 알릴 수 있습니다.
CfRevertPlaceholder 자리 표시자를 다시 일반 파일로 되돌리고 재분석 태그, 파일 ID 등과 같은 모든 특수 특성을 제거합니다.
CfSetCorrelationVector 동기화 공급자가 플랫폼에 자리 표시자 파일에서 원격 분석 목적으로 특정 상관 관계 벡터를 사용하도록 지시할 수 있습니다. 선택 사항입니다.
CfSetInSyncState 자리 표시자 파일 또는 폴더의 동기화 중 상태를 설정합니다.
CfSetPinState 이렇게 하면 사용자의 의도를 나타내는 데 사용되는 자리 표시자의 핀 상태가 설정됩니다. 동기화 공급자뿐만 아니라 모든 애플리케이션에서 이 함수를 호출할 수 있습니다.
CfUnregisterSyncRoot 이전에 등록된 동기화 루트의 등록을 취소합니다.
CfUpdatePlaceholder 자리 표시자 파일 또는 디렉터리의 특성을 업데이트.
CfUpdateSyncProviderStatus 동기화 공급자의 현재 상태를 업데이트.

구조체

CF_CALLBACK_INFO 일반적인 콜백 정보를 포함합니다.
CF_CALLBACK_PARAMETERS 파일 오프셋, 길이, 플래그 등과 같은 콜백 특정 매개 변수를 포함합니다.
CF_CALLBACK_REGISTRATION 동기화 공급자가 등록할 콜백입니다.
CF_FILE_RANGE 자리 표시자 파일의 데이터 범위를 지정합니다.
CF_FS_METADATA 자리 표시자 파일 또는 디렉터리 메타데이터입니다.
CF_HYDRATION_POLICY 기본 하이드레이션 정책 및 해당 한정자를 지정합니다.
CF_OPERATION_INFO 자리 표시자 파일 또는 폴더의 작업에 대한 정보입니다.
CF_OPERATION_PARAMETERS 자리 표시자 파일 또는 폴더에 대한 작업의 매개 변수입니다.
CF_PLACEHOLDER_BASIC_INFO 기본 자리 표시자 정보입니다.
CF_PLACEHOLDER_CREATE_INFO 새 자리 표시자 파일 또는 디렉터리를 만들기 위한 자리 표시자 정보를 포함합니다.
CF_PLACEHOLDER_STANDARD_INFO 표준 자리 표시자 정보입니다.

CF_PLATFORM_INFO 클라우드 파일 플랫폼에 대한 정보를 반환합니다. 이는 여러 버전의 Windows에서 실행되는 동기화 공급자를 위한 것입니다.
CF_POPULATION_POLICY 기본 채우기 정책 및 해당 한정자를 지정합니다.
CF_PROCESS_INFO 사용자 프로세스에 대한 정보를 포함합니다.
CF_SYNC_POLICIES 동기화 루트에서 사용하는 동기화 정책을 정의합니다.
CF_SYNC_REGISTRATION 등록할 동기화 공급자 및 동기화 루트의 세부 정보입니다.
CF_SYNC_ROOT_BASIC_INFO 기본 동기화 루트 정보입니다.
CF_SYNC_ROOT_PROVIDER_INFO 루트 공급자 정보를 동기화합니다.
CF_SYNC_ROOT_STANDARD_INFO 표준 동기화 루트 정보입니다.
CF_SYNC_STATUS CF_OPERATION_INFO 구조에서 지정된 동기화 루트의 상태를 설명하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

COM+(구성 요소 서비스)

아티클2023. 08. 22.

COM+(구성 요소 서비스) 기술 개요입니다.

COM+(Component Services)를 개발하려면 다음 헤더가 필요합니다.

- comadmin.h
- mtxdm.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- COM+(구성 요소 서비스)

열거형

COMAdminTxIsolationLevelOptions 트랜잭션에 사용할 격리 수준을 나타냅니다.
COMPLUS_APPTYPE 추적기 서버에서 추적하는 애플리케이션 유형을 나타냅니다.
CRMFLAGS CRM 보상자에 대한 특정 로그 레코드가 작성된 시기에 대한 정보를 제공합니다.
CRMREGFLAGS CRM 보상자가 수신해야 하는 트랜잭션 완료 단계와 복구를 시도한 후에도 의심스러운 트랜잭션이 남아 있는 경우 복구가 실패해야 하는지 여부를 제어합니다.
CrmTransactionState 트랜잭션의 현재 트랜잭션 상태를 나타냅니다.
CSC_Binding CoCreateActivity에서 반환된 작업을 통해 제출된 모든 작업을 STA(단일 스레드 아파트)에만 바인딩해야 하는지 여부를 나타냅니다. 이 열거형은 MTA(다중 스레드 아파트)에는 영향을 주지 않습니다.
CSC_COMTIIntrinsicsConfig

현재 COMTI(COM Transaction Integrator) 내장 함수가 새 컨텍스트로 전파되는지 여부를 나타냅니다.
CSC_IISIntrinsicsConfig 현재 IIS 내장 함수가 새 컨텍스트로 전파되는지 여부를 나타냅니다.
CSC_InheritanceConfig 현재 컨텍스트를 기반으로 새 컨텍스트를 만들거나 CServiceConfig의 정보만을 기반으로 새 컨텍스트를 만들 것인지를 나타냅니다.
CSC_PartitionConfig 묶은 컨텍스트가 실행되는 COM+ 파티션을 나타냅니다.
CSC_SxsConfig CServiceConfig에 대해 병렬 어셈블리를 구성하는 방법을 나타냅니다.
CSC_SynchronizationConfig CServiceConfig에 대해 동기화를 구성하는 방법을 나타냅니다.
CSC_ThreadPool CoCreateActivity에서 반환된 작업을 통해 제출된 작업을 실행하는 스레드 풀을 나타냅니다.
CSC_TrackerConfig 묶은 코드가 실행되는 컨텍스트에 추적기 속성이 추가되는지 여부를 나타냅니다.
CSC_TransactionConfig 트랜잭션이 CServiceConfig에 대해 구성된 방법을 나타냅니다.
GetAppTrackerDataFlags IGetAppTrackerData 인터페이스에 대한 호출에서 반환되는 데이터를 제어합니다.
TRACKING_COLL_TYPE 추적 정보 컬렉션의 개체 유형을 나타냅니다.
TransactionVote 현재 트랜잭션을 커밋하거나 중단할 개체의 준비 상태를 나타냅니다.

Functions

중단 현재 트랜잭션에 참여하는 모든 COM 개체의 작업을 중단합니다. 트랜잭션은 이 메서드의 반환에 종료됩니다. (ITransactionContext.Abort)
중단 현재 트랜잭션에 참여하는 모든 COM 개체의 작업을 중단합니다. 트랜잭션은 이 메서드의 반환에 종료됩니다. (ITransactionContextEx.Abort)
중단 트랜잭션을 중단합니다.
AbortRecord 중단 단계 중에 CRM Compensator에 로그 레코드를 전달합니다. (ICrmCompensator.AbortRecord)
AbortRecordVariants 중단 단계 중에 CRM Compensator에 로그 레코드를 전달합니다. (ICrmCompensatorVariants.AbortRecordVariants)
활성화 COM+ 개체가 활성화될 때마다 컨텍스트별 초기화를 수행할 수 있도록 합니다. (IObjectControl.Activate)
활성화 COM+ 개체가 활성화될 때마다 컨텍스트별 초기화를 수행할 수 있도록 합니다. (ObjectControl.Activate)
Activityid 지정된 인덱스에 대한 CRM 작업자의 활동 ID를 검색합니다.
추가 컬렉션에 이벤트 개체를 추가합니다.
추가 컬렉션에 항목을 추가하여 높은 인덱스 값을 제공합니다.
AliasComponent

기존 COM+ 구성 요소에 대한 별칭을 만듭니다.
AllocResource 인벤토리에서 리소스를 할당합니다.
AreApplicationInstancesPaused 지정된 애플리케이션 인스턴스(프로세스)가 일시 중지되는지 여부를 결정합니다.
AsyncCall 사용자 정의 작업을 비동기적으로 수행합니다. (IMTSActivity.AsyncCall)
AsynchronousCall 사용자 정의 작업을 비동기적으로 수행합니다. (IServiceActivity.AsynchronousCall)
BackupREGDB COM+ 클래스 등록 데이터베이스를 지정된 파일에 백업합니다.
BeginAbort CRM Compensator에게 트랜잭션 완료의 중단 단계와 레코드가 배달될 예정임을 알 수 있습니다. (ICrmCompensator.BeginAbort)
BeginAbortVariants CRM Compensator에게 트랜잭션 완료의 중단 단계와 레코드가 배달될 예정임을 알 수 있습니다. (ICrmCompensatorVariants.BeginAbortVariants)
BeginCommit 트랜잭션 완료의 커밋 단계를 CRM Compensator에 알리고 레코드가 배달될 예정임을 알 수 있습니다.
BeginCommitVariants 트랜잭션 완료의 커밋 단계(2단계)를 CRM Compensator에 알리고 레코드가 배달될 예정임을 알 수 있습니다.
BeginPrepare CRM Compensator에게 트랜잭션 완료 준비 단계에 대해 알리고 레코드가 배달될 예정임을 알 수 있습니다. (ICrmCompensator.BeginPrepare)
BeginPrepareVariants

CRM Compensator에게 트랜잭션 완료 준비 단계에 대해 알리고 레코드가 배달될 예정임을 알 수 있습니다. (ICrmCompensatorVariants.BeginPrepareVariants)
BindToCurrentThread IMTSActivity::AsyncCall 또는 IMTSActivity::SynchronousCall을 사용하여 제출된 일괄 처리 작업을 현재 STA(단일 스레드 아파트)에 바인딩합니다.
BindToCurrentThread 사용자 정의 일괄 처리 작업을 현재 스레드에 바인딩합니다.
BringYourOwnTransaction 제공하는 기존 트랜잭션에서 묶은 코드를 실행할 수 있습니다.
CanBePooled COM+ 런타임 환경에 개체가 비활성화될 때 다시 사용할 수 있는지 여부를 알릴 수 있습니다.
CanBePooled 비활성화될 때 개체를 다시 사용할 수 있는지 여부를 나타냅니다.
ChangedEventClass 이벤트 클래스 개체가 추가, 수정 또는 삭제되었음을 나타냅니다. (IEventObjectChange.ChangedEventClass)
ChangedEventClass 이벤트 클래스 개체가 추가, 수정 또는 삭제되었음을 나타냅니다. (IEventObjectChange2.ChangedEventClass)
ChangedPublisher 게시자 개체가 추가, 수정 또는 삭제되었음을 나타냅니다.
ChangedSubscription 구독 개체가 추가, 수정 또는 삭제되었음을 나타냅니다. (IEventObjectChange.ChangedSubscription)
ChangedSubscription 구독 개체가 추가, 수정 또는 삭제되었음을 나타냅니다. (IEventObjectChange2.ChangedSubscription)
복제

현재 열거자와 동일한 열거 상태를 포함하는 열거자를 만듭니다. (IEnumNames.Clone)
복제 현재 열거자와 동일한 열거 상태를 포함하는 열거자를 만듭니다. (IEnumEventObject.Clone)
닫기 홀더를 닫습니다.
CoCreateActivity COM+ 구성 요소를 만들지 않고 COM+ 서비스를 사용할 수 있는 동기 또는 비동기 일괄 작업을 수행하기 위한 작업을 만듭니다.
CoEnterServiceDomain COM+ 서비스를 사용할 수 있는 코드를 입력하는 데 사용됩니다.
CoGetDefaultContext 지정된 아파트의 기본 컨텍스트에 대한 참조를 검색합니다.
CoLeaveServiceDomain COM+ 서비스를 사용하는 코드를 남기는 데 사용됩니다.
커밋 현재 트랜잭션에 참여하는 모든 COM 개체의 작업을 커밋하려고 시도합니다. 트랜잭션은 이 메서드의 반환에 종료됩니다. (ITransactionContext.Commit)
커밋 현재 트랜잭션에 참여하는 모든 COM 개체의 작업을 커밋하려고 시도합니다. 트랜잭션은 이 메서드의 반환에 종료됩니다. (ITransactionContextEx.Commit)
커밋 트랜잭션을 커밋합니다.
CommitRecord 커밋 단계 중에 전달 순서에 따라 로그 레코드를 전달합니다.
CommitRecordVariants 커밋 단계 중에 CRM Compensator에 로그 레코드를 전달합니다.

ComTIIntrinsicsConfig 묶은 작업에 대한 COMTI 내장 함수를 구성합니다.
ConfigureBYOT 사용자 고유의 트랜잭션을 가져올 때 사용하는 트랜잭션을 구성할 수 있습니다.
ConfigureBYOTSysTxn 트랜잭션 프록시를 사용하여 지정한 기존 트랜잭션의 scope 묶은 코드를 실행할 수 있습니다.
ConfigureSynchronization 묶은 작업에 대한 동기화를 구성합니다.
ConfigureTransaction 묶은 작업에서 트랜잭션을 사용하는 방법을 구성합니다.
연결 지정된 원격 컴퓨터의 COM+ 카탈로그에 연결합니다.
구문 지정된 매개 변수를 사용하여 개체를 생성합니다.
ContainingContextTreatment 포함된 컨텍스트가 현재 컨텍스트를 기반으로 하는지 여부를 결정합니다.
CopyApplications 지정된 COM+ 애플리케이션을 한 파티션에서 다른 파티션으로 복사합니다.
CopyComponents 지정된 구성 요소를 한 파티션에서 다른 파티션으로 복사합니다.
Count 추적 정보 컬렉션의 개체 수를 검색합니다.
Count 컨텍스트 개체 속성의 수를 검색합니다.
Count

컨텍스트 속성의 수를 계산합니다.
CreateInstance TIP 트랜잭션 URL로 지정된 수동 트랜잭션의 scope 내에서 실행되는 COM+ 개체를 만듭니다.
CreateInstance ITransaction 인터페이스에 대한 참조로 지정된 수동 트랜잭션의 scope 내에서 실행되는 COM+ 개체를 만듭니다.
CreateInstance 현재 개체의 컨텍스트를 사용하여 개체를 만듭니다. (IObjectContext.CreateInstance)
CreateInstance 트랜잭션 컨텍스트 개체에서 시작된 트랜잭션의 scope 내에서 실행할 수 있는 COM 개체를 만듭니다. (ITransactionContext.CreateInstance)
CreateInstance 트랜잭션 컨텍스트 개체에서 시작된 트랜잭션의 scope 내에서 실행할 수 있는 COM 개체를 만듭니다. (ITransactionContextEx.CreateInstance)
CreateInstance 현재 개체의 컨텍스트를 사용하여 개체를 만듭니다. (ObjectContext.CreateInstance)
CreateInstanceWithSysTx 지정된 로컬 트랜잭션의 scope 내에서 실행되는 COM+ 개체를 만듭니다. (ICreateWithLocalTransaction.CreateInstanceWithSysTx)
CreateManagedStub 현재 COM+ 컨텍스트 내에서 관리되는 개체에 대한 스텁을 만듭니다.
CreateProperty 지정된 이름을 사용하여 새 공유 속성을 만듭니다.
CreatePropertyByPosition 지정된 인덱스로 새 공유 속성을 만듭니다.
CreatePropertyGroup 새 공유 속성 그룹을 만듭니다.

CreateResource 리소스를 만듭니다.
CreateServiceForApplication Windows 서비스로 실행되도록 COM+ 애플리케이션을 구성합니다.
CreateVoter COM+ 트랜잭션 컨텍스트가 트랜잭션에 투표할 수 있도록 투표를 제공합니다.
비활성화 COM+ 개체가 재활용되거나 제거되기 전에 필요한 정리를 수행할 수 있도록 합니다.
비활성화 COM+ 개체가 재활용되거나 제거되기 전에 필요한 정리를 수행할 수 있도록 합니다.
DeleteServiceForApplication 지정된 COM+ 애플리케이션과 연결된 Windows 서비스를 삭제합니다.
설명 지정된 인덱스에 대한 CRM Compensator의 설명을 검색합니다.
DestroyManagedStub CreateManagedStub에서 만든 스텁을 삭제합니다.
DestroyResource 리소스를 삭제합니다.
DestroyResources 리소스(문자열 리소스 버전)를 삭제합니다.
DisableCommit 개체의 트랜잭션 업데이트가 일관되지 않은 상태이며 현재 상태에서 커밋할 수 없음을 선언합니다.
DisableCommit 개체의 트랜잭션 업데이트가 일관되지 않으며 현재 상태에서 커밋할 수 없음을 선언합니다.
DoCallback

호출 개체가 바인딩되어야 하는 지정된 애플리케이션 도메인으로 전환하고, 해당 애플리케이션 도메인에서 제공된 콜백 함수를 실행한 다음, 원래 애플리케이션 도메인으로 돌아갑니다.
DumpApplicationInstance 지정된 애플리케이션 instance(프로세스)의 상태 이미지를 포함하는 덤프 파일을 만듭니다.
EnableCommit 개체의 작업이 반드시 완료된 것은 아니지만 트랜잭션 업데이트가 일관된 상태이며 현재 형식으로 커밋될 수 있음을 선언합니다.
EnableCommit 현재 개체의 작업이 반드시 완료된 것은 아니지만 트랜잭션 업데이트가 일관되고 현재 형식으로 커밋될 수 있음을 선언합니다.
EndAbort 중단 단계에서 사용할 수 있는 모든 로그 레코드를 수신했음을 CRM Compensator에 알 수 있습니다. (ICrmCompensator.EndAbort)
EndAbortVariants 중단 단계에서 사용할 수 있는 모든 로그 레코드를 수신했음을 CRM Compensator에 알 수 있습니다. (ICrmCompensatorVariants.EndAbortVariants)
Endcommit 커밋 단계에서 사용할 수 있는 모든 로그 레코드를 전달했음을 CRM Compensator에 알 수 있습니다. (ICrmCompensator.EndCommit)
EndCommitVariants 커밋 단계에서 사용할 수 있는 모든 로그 레코드를 전달했음을 CRM Compensator에 알 수 있습니다. (ICrmCompensatorVariants.EndCommitVariants)
EndPrepare 준비 단계에서 사용할 수 있는 모든 로그 레코드가 CRM Compensator에 있음을 알 수 있습니다. (ICrmCompensator.EndPrepare)
EndPrepareVariants 준비 단계에서 사용할 수 있는 모든 로그 레코드가 CRM Compensator에 있음을 알 수 있습니다. (ICrmCompensatorVariants.EndPrepareVariants)
EnlistResource

트랜잭션에 리소스를 등록합니다.
EnumNames 컨텍스트 개체 속성에 대한 열거자에 대한 참조를 검색합니다.
EnumNames 현재 컨텍스트 속성의 이름 목록을 검색합니다.
ExportApplication COM+ 애플리케이션 또는 애플리케이션 프록시를 다른 컴퓨터에 설치할 준비가 된 파일로 내보냅니다.
ExportPartition 파티션을 파일로 내보냅니다.
FinalClientRetry 클라이언트 쪽 예외 처리 구성 요소에 메시지를 서버에 배달하려는 모든 메시지 큐 시도가 거부되었음을 알릴 수 있습니다. 메시지는 클라이언트 쪽 Xact 배달 못 한 편지 큐에서 끝났습니다.
FinalServerRetry 지연된 활성화를 재생하려는 모든 시도가 실패했음을 서버 쪽 Exception_CLSID 구현에 알릴 수 있습니다. 메시지가 최종 휴식 큐로 이동하려고 합니다.
FireSubscription 단일 구독자에게 이벤트를 발생합니다.
FlushPartitionCache 사용자를 기본 파티션에 매핑하는 캐시를 비웁니다.
ForceLog 모든 로그 레코드가 디스크에서 지속되도록 합니다.
ForceTransactionToAbort 트랜잭션에 대한 즉시 중단 호출을 수행합니다.
ForgetLogRecord 인터페이스의 이 instance 작성한 마지막 로그 레코드를 잊어버리세요.

FreeResource 인벤토리에 리소스를 반환합니다.
get__NewEnum CRM 클럭의 instance CLSID에 대한 열거자를 검색합니다.
get__NewEnum 보안 호출 컨텍스트 컬렉션에 대한 열거자를 검색합니다.
get__NewEnum 보안 호출자 컬렉션에 대한 열거자를 검색합니다.
get__NewEnum 보안 ID 컬렉션에 대한 열거자를 검색합니다.
get__NewEnum 명명된 보안 호출 컨텍스트 속성에 대한 열거자를 검색합니다.
get__NewEnum 명명된 컨텍스트 개체 속성에 대한 열거자를 검색합니다.
get__NewEnum 컬렉션의 개체에 대한 열거자입니다.
get__NewEnum 컬렉션 개체를 반복하는 데 사용할 수 있는 열거자를 검색합니다.
get_AddEnabled 컬렉션에 대해 Add 메서드를 사용할 수 있는지 여부를 나타냅니다.
get_AllowInprocActivation 이벤트 클래스를 In Process에서 활성화할 수 있는지 여부를 나타냅니다. (Get)
get_AllowInprocActivation 게시자 프로세스에서 구독자를 활성화할 수 있는지 여부를 나타냅니다. (IEventControl.get_AllowInprocActivation)

get_AllowInprocActivation 게시자 프로세스에서 구독자를 활성화할 수 있는지 여부를 나타냅니다. (IMultiInterfaceEventControl.get_AllowInprocActivation)
get_ClassFactory 풀된 개체에 대한 클래스 팩터리를 검색합니다.
get_CommitBatchSize 커밋 일괄 처리 크기를 검색합니다.
get_ConstructString 개체의 생성자 문자열을 검색합니다.
get_ContextInfo 현재 개체 컨텍스트의 컨텍스트 정보 개체를 검색합니다.
get_Count 컬렉션에서 CRM 클럭 수를 검색합니다.
get_Count 이 CRM 서기가 작성한 로그 레코드 수를 검색합니다.
get_Count 개체에서 데이터 값의 수를 검색합니다.
get_Count 카탈로그에서 실행 중인 패키지 수를 검색합니다.
get_Count 보안 컨텍스트 컬렉션의 속성 수를 검색합니다.
get_Count 보안 호출자 컬렉션의 호출자 수를 검색합니다.
get_Count 보안 ID 컬렉션의 속성 수를 검색합니다.

get_Count 명명된 컨텍스트 개체 속성의 수를 검색합니다.
get_Count 컬렉션의 개체 수입니다.
get_Count 컬렉션의 항목 수를 검색합니다. (ICatalogCollection.get_Count)
get_CreationTimeout 풀링된 개체를 활성화하기 위한 제한 시간 간격을 검색합니다.
get_CurrentPartitionID 현재 파티션의 식별자를 검색합니다.
get_CurrentPartitionName 현재 파티션의 이름을 검색합니다.
get_CustomConfigCLSID 구독 개체의 속성 모음에 속성을 추가하는 데 도움이 될 수 있는 구성 요소의 CLSID입니다. 이 속성은 이전 버전과의 호환성에 대해서만 지원됩니다. (Get)
get_DataStoreMajorVersion 카탈로그 데이터 저장소의 주 버전 번호를 검색합니다.
get_DataStoreMinorVersion 카탈로그 데이터 저장소의 부 버전 번호를 검색합니다.
get_Description 이벤트 클래스 개체에 대한 표시 가능한 텍스트 설명입니다. (Get)
get_Description 구독에 대한 표시 가능한 텍스트 설명입니다. (Get)
get_DestPath 대상(출력) 큐의 경로를 검색합니다.

get_DisplayName 개체의 표시 이름을 검색합니다.
get_Enabled 구독을 사용할 수 있는지 여부를 나타냅니다. (Get)
get_EventClassID 이벤트 클래스 개체의 CLSID입니다. (Get)
get_EventClassID 구독과 연결된 이벤트 클래스의 고유 ID입니다. (Get)
get_EventClassName 이벤트 클래스 개체의 ProgID입니다. (Get)
get_EventID 개체의 이벤트 식별자를 검색합니다.
get_EventObjectChangeEventClassID 호출자에게 이벤트 저장소에 대한 변경 내용을 알 수 있도록 하는 이벤트 클래스 개체의 CLSID를 검색합니다.
get_FireEvents 이벤트 싱크에 대해 이벤트를 사용하도록 설정하거나 사용하지 않도록 설정할지 여부를 검색합니다.
get_FireInParallel 이 클래스의 이벤트를 병렬로 실행할 수 있는지 여부를 나타냅니다. (Get)
get_FireInParallel 이벤트를 두 명 이상의 구독자에게 병렬로 전달할 수 있는지 여부를 나타냅니다. (Get)
get_FiringInterfaceID 이벤트 클래스 개체의 이벤트 인터페이스 ID입니다. 이 속성은 이전 버전과의 호환성에 대해서만 지원됩니다. (Get)
get_GlobalPartitionID

전역 파티션의 식별자를 검색합니다.
get_Group 기존 공유 속성 그룹에 대한 참조를 검색합니다.
get_InterfaceID 구독자가 이벤트를 수신하려는 특정 인터페이스의 식별자입니다. (Get)
get_Is64BitCatalogServer 현재 연결된 카탈로그 서버가 64비트 컴퓨터인지 여부를 나타냅니다.
get_IsApplicationInstanceDumpSupported 애플리케이션 instance 덤프에 필요한 소프트웨어가 설치되어 있는지 여부를 나타냅니다.
get_Item 보안 호출 컨텍스트 컬렉션에서 지정된 속성을 검색합니다.
get_Item 보안 호출자 컬렉션에서 지정된 호출자를 검색합니다.
get_Item 보안 ID 컬렉션에서 지정된 속성을 검색합니다.
get_Item 명명된 속성을 검색합니다.
get_Item 컬렉션의 항목입니다.
get_Item 지정된 인덱스에 해당하는 항목을 검색합니다.
get_Key 개체의 키 속성을 검색합니다.
get_MachineName 구독자를 활성화해야 하는 컴퓨터의 이름입니다(영구 구독의 경우). (Get)

get_MajorVersion COMAdmin 라이브러리의 주 버전 번호를 검색합니다.
get_MaxPoolSize 풀의 최대 개체 수를 검색합니다.
get_MethodName 이벤트 메서드의 이름입니다. (Get)
get_MinorVersion COMAdmin 라이브러리의 부 버전 번호를 검색합니다.
get_MinPoolSize 풀의 최소 개체 수를 검색합니다.
get_MultiInterfacePublisherFilterCLSID IMultiInterfacePublisherFilter를 구현하는 개체의 CLSID입니다. (Get)
get_Name 컬렉션의 이름을 검색합니다.
get_Name 개체의 name 속성을 검색합니다.
get_Names 데이터 값의 이름에 대한 열거자를 검색합니다.
get_NewEnum IEnumEventObject를 구현하는 열거형 개체입니다.
get_OwnerSID 이벤트 클래스 개체 작성자의 보안 ID입니다. 이 속성은 이전 버전과의 호환성에 대해서만 지원됩니다. (Get)
get_OwnerSID 구독 작성자의 보안 ID입니다. (Get)

get_PackageGuid 이벤트가 발생한 패키지에 대한 GUID(Globally Unique Identifier)를 검색합니다.
get_PackageName IMtsEvents 인터페이스를 구현하는 개체의 instance 실행 중인 패키지의 이름을 검색합니다.
get_PerUser 구독 소유자가 게시자와 동일한 컴퓨터에 로그온한 경우에만 구독이 이벤트를 수신하는지 여부를 나타냅니다. (Get)
get_Property 지정된 이름의 기존 공유 속성에 대한 참조를 검색합니다.
get_PropertyByPosition 지정된 인덱스를 사용하여 기존 공유 속성에 대한 참조를 검색합니다.
get_PublisherID 이벤트 게시자에 대한 CLSID입니다. (Get)
get_PublisherID 이벤트 게시자의 고유 ID입니다. (Get)
get_RemoveEnabled 컬렉션에 대해 Remove 메서드를 사용할 수 있는지 여부를 나타냅니다.
get_Security 현재 개체 컨텍스트의 보안 개체를 검색합니다.
get_SourcePath 원본(입력) 큐의 현재 경로를 검색합니다.
get_StructuredRecords 이 CRM 서기가 작성한 로그 레코드가 구조화되었는지 여부를 나타내는 플래그를 검색합니다.
get_SubscriberCLSID 구독자 구성 요소의 CLSID입니다(영구 구독의 경우). (Get)

get_SubscriberInterface 구독자의 이벤트 인터페이스에 대한 마샬링된 포인터입니다(임시 구독의 경우). (Get)
get_SubscriptionID 구독 개체의 고유 ID입니다. (Get)
get_SubscriptionName 구독 개체의 표시 가능한 이름입니다. (Get)
get_TransactionAffinity 트랜잭션이 완료될 때까지 트랜잭션에 관련된 개체가 보관되는지 여부를 결정합니다.
get_TransactionState 트랜잭션의 현재 상태를 검색합니다.
get_TransactionUOW 로그 레코드에 트랜잭션 UOW를 기록하지 않고도 UOW(트랜잭션 작업 단위)를 검색합니다.
get_TypeLib 이벤트 인터페이스에 대한 설명을 포함하는 형식 라이브러리의 경로입니다. (Get)
get_Valid 카탈로그 데이터 저장소에서 모든 속성을 성공적으로 읽었는지 여부를 나타냅니다.
get_Value 지정된 사용자 정의 이벤트의 값을 검색합니다.
get_Value 공유 속성의 값을 검색합니다.
get_Value 이 카탈로그 개체에 의해 노출되는 지정된 속성의 값에 액세스합니다. (Get)
GetActivityId 개체 컨텍스트와 연결된 활동 식별자를 검색합니다.
GetActivityId

현재 활동과 연결된 GUID를 검색합니다.
GetActivityId 현재 활동의 식별자를 검색합니다.
GetApplicationId 현재 개체 컨텍스트의 애플리케이션 GUID를 검색합니다.
GetApplicationId 현재 개체 컨텍스트의 애플리케이션 식별자를 검색합니다.
GetApplicationInstanceId 현재 개체 컨텍스트의 애플리케이션 instance GUID를 검색합니다.
GetApplicationInstanceId 현재 개체 컨텍스트의 애플리케이션 instance 식별자를 검색합니다.
GetApplicationInstanceIDFromProcessID 지정된 프로세스 식별자에 대한 애플리케이션 instance 식별자를 검색합니다.
GetApplicationProcessDetails COM+ 애플리케이션을 호스팅하는 단일 프로세스에 대한 자세한 정보를 검색합니다.
GetApplicationProcesses COM+ 애플리케이션을 호스팅하는 모든 프로세스 또는 이러한 프로세스의 지정된 하위 집합에 대한 요약 정보를 검색합니다.
GetApplicationsInProcess 단일 프로세스에서 호스트되는 모든 COM+ 애플리케이션 또는 이러한 애플리케이션의 지정된 하위 집합에 대한 요약 정보를 검색합니다.
GetClerks 클럭의 현재 상태의 스냅샷 클럭 컬렉션 개체를 검색합니다.
GetCLSID 개체의 CLSID를 검색합니다.
GetCollection

현재 컬렉션과 관련된 COM+ 카탈로그에서 컬렉션을 검색합니다.
GetCollection COM+ 카탈로그에서 최상위 컬렉션을 검색합니다.
GetCollectionByQuery 모든 부모 항목에 대한 키 속성 값이 지정된 경우 COM+ 카탈로그에서 컬렉션을 검색합니다.
GetCollectionByQuery2 COM+ 카탈로그에서 지정된 쿼리 키 집합을 충족하는 항목의 컬렉션을 검색합니다.
GetColumn 구조화되지 않은 로그 레코드 하나를 볼 수 있는 필드 배열로 서식을 지정합니다.
GetColumnCount 이 CRM Compensator에서 사용하는 형식의 로그 레코드에 있는 필드(열)의 수를 검색합니다.
GetColumnHeaders 정보가 표시될 때 열 머리글로 사용할 수 있도록 필드(열)의 이름을 검색합니다.
GetColumnVariants 하나의 구조적 로그 레코드를 볼 수 있는 필드 배열로 서식을 지정합니다.
GetComponentDetails 프로세스에서 호스트되는 단일 COM+ 구성 요소에 대한 자세한 정보를 검색합니다.
GetComponentsInProcess 단일 프로세스에서 호스트되는 모든 COM+ 구성 요소 또는 이러한 구성 요소의 지정된 하위 집합에 대한 요약 정보를 검색합니다.
GetComponentVersionCount 지정된 구성 요소가 설치된 파티션 수를 검색합니다.
GetContext 현재 컨텍스트를 결정합니다.
GetContextId 이 개체 컨텍스트의 고유 식별자를 검색합니다.

GetContextId 현재 컨텍스트의 식별자를 검색합니다.
GetCurrentQueuedRequests 현재 큐에 대기 중인 비동기 실행 요청 수를 검색합니다.
GetCurrentThreads 현재 풀에 있는 스레드 수를 검색합니다.
GetDeactivateOnReturn 완료 플래그의 값을 검색합니다.
GetDeleteDelay 풀링된 스레드가 제거되기 전에 유휴 상태일 수 있는 시간(밀리초)을 검색합니다.
GetDirectCallerName 현재 실행 중인 메서드를 호출한 외부 프로세스와 연결된 사용자 이름을 검색합니다.
GetDirectCallerSID 현재 실행 중인 메서드를 호출한 외부 프로세스의 보안 식별자를 검색합니다.
GetDirectCreatorName 현재 개체의 즉시(out-of-process) 작성자와 연결된 사용자 이름을 검색합니다.
GetDirectCreatorSID MTS 2.0에서 이 메서드는 현재 개체를 직접 만든 외부 프로세스의 보안 식별자를 검색합니다. COM+에서는 이 메서드를 사용하지 마세요.
GetDispenserManager 디스펜서 관리자의 IDispenserManager 인터페이스를 검색합니다.
GetEventClassesForIID 지정된 인터페이스를 구현하는 컴퓨터에 등록된 이벤트 클래스 목록을 검색합니다.
GetEventDispatcher 현재 프로세스에 대한 이벤트 디스패처에 대한 포인터를 검색합니다.

GetIdentifier 비 DTC 트랜잭션의 식별자를 검색합니다.
GetIObjectControl 관리되는 개체와 연결된 IObjectControl 인터페이스를 검색합니다.
GetIsolationLevel 비 DTC 트랜잭션의 격리 수준을 검색합니다.
GetIUnknown 관리되는 개체와 연결된 IUnknown 인터페이스를 검색합니다.
GetLBServer 부하 분산 서버의 이름을 검색합니다.
GetLogRecord 숫자 인덱스가 지정된 경우 구조화되지 않은 로그 레코드를 검색합니다.
GetLogRecordVariants 숫자 인덱스가 지정된 경우 구조화된 로그 레코드를 검색합니다.
GetMachineName 부하 분산 서버의 컴퓨터 이름을 검색합니다.
GetManagedExtensions 설치된 COM+ 버전이 서비스 구성 요소(관리되는 개체)를 관리하기 위해 제공되는 특수 기능을 지원하는지 여부를 결정합니다.
GetMaxQueuedRequests 동시에 큐에 대기할 수 있는 최대 비동기 실행 요청 수를 검색합니다.
GetMaxThreads 풀에서 허용되는 최대 스레드 수를 검색합니다.
GetModules 어셈블리에 포함된 모듈의 이름을 가져오는 데 사용됩니다.

GetMultipleComponentsInfo 지정된 파일에 있는 구성 요소에 대한 정보를 검색합니다.
GetMyTransactionVote 일관된 플래그의 값을 검색합니다.
GetObject 개체 풀에서 개체를 검색합니다.
GetObjectContext 현재 COM+ 개체와 연결된 컨텍스트에 대한 참조를 검색합니다.
GetOriginalCallerName 현재 개체에 대한 호출이 시작된 호출 시퀀스를 시작한 기본 프로세스와 연결된 사용자 이름을 검색합니다.
GetOriginalCallerSID 현재 메서드가 호출된 호출 순서를 시작한 기본 프로세스의 보안 식별자를 검색합니다.
GetOriginalCreatorName 현재 개체가 실행 중인 작업을 시작한 원래 기본 프로세스와 연결된 사용자 이름을 검색합니다.
GetOriginalCreatorSID MTS 2.0에서 이 메서드는 현재 개체가 실행되는 작업을 시작한 기본 프로세스의 보안 식별자를 검색합니다. COM+에서는 이 메서드를 사용하지 마세요.
GetPartitionId 현재 개체 컨텍스트의 COM+ 파티션 GUID를 검색합니다.
GetPartitionId 현재 개체 컨텍스트의 파티션 식별자를 검색합니다.
GetPartitionID 지정된 COM+ 애플리케이션의 파티션 식별자를 검색합니다.
GetPartitionName 지정된 COM+ 애플리케이션의 이름을 검색합니다.

GetProcessID 이벤트가 발생한 프로세스의 식별자를 검색합니다.
GetProperty 컨텍스트 개체 속성을 검색합니다.
GetProperty 지정된 컨텍스트 속성의 값을 검색합니다.
GetPropName 지정된 인덱스 번호에 해당하는 속성의 이름을 검색합니다.
GetPublisherProperty 게시자 컨텍스트를 정의하기 위해 속성 모음에 저장된 속성의 값을 검색합니다.
GetPublisherPropertyCollection 게시자 속성 모음에 저장된 속성 및 값의 컬렉션을 검색합니다.
GetResource 풀된 개체 목록에서 개체를 검색합니다.
GetSecurityCallContext 현재 호출과 연결된 SecurityCallContext 클래스에서 만든 개체에 대한 참조를 검색합니다. (IGetSecurityCallContext.GetSecurityCallContext)
GetSubscriberProperty 구독자 컨텍스트를 정의하기 위해 속성 모음에 저장된 속성의 값을 검색합니다.
GetSubscriberPropertyCollection 구독자 속성 모음에 저장된 속성 및 값의 컬렉션을 검색합니다.
GetSubscriptions 이벤트 메서드와 연결된 구독의 컬렉션을 검색합니다.
GetSubscriptions 이벤트 메서드와 연결된 구독 개체의 컬렉션을 검색합니다.

GetSuggestedPollingInterval 추적기 서버에서 제안하는 폴링의 최소 간격을 검색합니다.
GetTipUrl TIP 컨텍스트의 URL을 검색합니다.
GetTrackerDataAsCollectionObject 컬렉션 개체의 형태로 모든 COM+ 애플리케이션에 대한 추적 데이터를 검색합니다.
GetTransaction 개체 컨텍스트의 트랜잭션 개체를 검색합니다.
GetTransaction 현재 트랜잭션에 대한 참조를 검색합니다.
GetTransactionId 개체 컨텍스트와 연결된 트랜잭션 식별자를 검색합니다. 동일한 트랜잭션의 개체는 동일한 트랜잭션 식별자를 공유합니다.
GetTransactionId 현재 트랜잭션의 식별자를 검색합니다.
GetTransactionResourcePool 이 컨텍스트의 트랜잭션과 연결된 리소스 풀을 검색합니다.
GetTransactionStatus 트랜잭션 상태 검색합니다.
GetUtilInterface 컬렉션에 대한 유틸리티 인터페이스를 검색합니다.
GetValue 지정된 속성의 값을 검색합니다.
HoldClerk 지정된 클럭에 대한 포인터를 검색합니다.

IISIntrinsicsConfig 묶은 작업에 대한 IIS 내장 함수를 구성합니다.
ImportComponent 프로세스 내 서버로 이미 등록된 구성 요소를 COM+ 애플리케이션으로 가져옵니다.
ImportComponents 애플리케이션에 이미 등록된 지정된 구성 요소를 가져옵니다.
ImportUnconfiguredComponents 지정된 클래스를 구성 요소로 COM+ 애플리케이션으로 가져옵니다.
Init 부하 분산 서버 개체를 초기화합니다.
초기화 호출 개체를 현재 애플리케이션 도메인에 바인딩하고 애플리케이션 도메인이 언로드될 때 실행되는 종료에 대한 콜백 함수를 제공합니다.
초기화 개체 풀을 초기화합니다.
초기화 이벤트 클래스를 게시자 필터와 연결합니다.
초기화 이벤트 메서드를 구독 개체 컬렉션과 연결합니다.
InstallApplication 지정된 파일에서 COM+ 애플리케이션 또는 애플리케이션 프록시를 설치합니다.
InstallComponent DLL 파일의 모든 구성 요소(COM 클래스)를 COM+ 애플리케이션에 설치하고 COM+ 클래스 등록 데이터베이스에 구성 요소를 등록합니다.
InstallEventClass 파일의 이벤트 클래스를 COM+ 애플리케이션에 설치합니다.

InstallMultipleComponents 여러 파일의 구성 요소를 COM+ 애플리케이션에 설치합니다.
InstallMultipleEventClasses 여러 파일의 이벤트 클래스를 COM+ 애플리케이션에 설치합니다.
InstallPartition 파일에서 파티션을 가져옵니다.
IsCallerInRole 개체의 직접 호출자가 지정된 역할(직접 또는 그룹의 일부)에 있는지 여부를 나타냅니다. (IObjectContext.IsCallerInRole)
IsCallerInRole 직접 호출자가 지정된 역할에 있는지 여부를 확인합니다.
IsCallerInRole 개체의 직접 호출자가 지정된 역할(직접 또는 그룹의 일부)에 있는지 여부를 나타냅니다. (ObjectContext.IsCallerInRole)
IsInTransaction 현재 개체가 트랜잭션에서 실행 중인지 여부를 나타냅니다. (ContextInfo.IsInTransaction)
IsInTransaction 개체가 트랜잭션 내에서 실행 중인지 여부를 나타냅니다.
IsInTransaction 현재 개체가 트랜잭션에서 실행 중인지 여부를 나타냅니다. (IObjectContextInfo.IsInTransaction)
IsInTransaction 현재 개체가 트랜잭션에서 실행 중인지 여부를 나타냅니다. (ObjectContext.IsInTransaction)
IsolationLevel 트랜잭션의 격리 수준을 설정합니다.
IsPropertyReadOnly Value를 사용하여 지정된 속성을 수정할 수 있는지 여부를 나타냅니다.

IsPropertyWriteOnly Value를 사용하여 지정된 속성을 읽을 수 있는지 여부를 나타냅니다.
IsReusable 비 DTC 트랜잭션 컨텍스트를 여러 트랜잭션에 다시 사용할 수 있는지 여부를 나타냅니다.
IsSafeToDelete COM+ 카탈로그 또는 레지스트리에서 지정된 DLL을 사용하고 있는지 여부를 확인합니다.
IsSameSxsConfig 병렬 어셈블리에 지정된 구성이 있는지 여부를 확인합니다.
IsSecurityEnabled 현재 개체에 대한 보안을 사용할 수 있는지 여부를 나타냅니다. 개체가 클라이언트 프로세스에서 실행되고 있지 않으면 COM+ 보안을 사용할 수 있습니다.
IsSecurityEnabled 개체에 대한 보안을 사용할 수 있는지 여부를 결정합니다.
IsSecurityEnabled 현재 개체에 대한 보안을 사용할 수 있는지 여부를 나타냅니다.
IsUserInRole 지정된 사용자가 지정된 역할에 있는지 여부를 확인합니다.
Item 추적 정보 컬렉션의 지정된 멤버에서 지정된 인터페이스를 검색합니다.
Item 지정된 인덱스에 대한 CRM 클럭의 instance CLSID를 검색합니다.
Item 지정된 패키지에 대한 IUnknown 포인터를 검색합니다.
LastRelease COM+ 개체 풀이 구현하는 개체에 대한 마지막 참조를 삭제할 때 호출됩니다.

MoveComponents 지정된 구성 요소를 한 애플리케이션에서 다른 애플리케이션으로 이동합니다.
MoveMessages 원본 큐에서 대상 큐로 모든 메시지를 이동합니다.
MTSCreateActivity 동기 또는 비동기 일괄 처리를 수행하는 단일 스레드 아파트에 활동을 만듭니다.
NewTransactionDescription 트랜잭션 통계가 표시될 때 사용되는 이름을 설정합니다.
다음 열거형 시퀀스에서 지정된 수의 항목을 검색합니다. (IEnumNames.Next)
다음 열거형 시퀀스에서 지정된 수의 항목을 검색합니다. (IEnumEventObject.Next)
OnActivityCreate 작업이 시작될 때 생성됩니다.
OnActivityDestroy 작업이 완료되면 생성됩니다.
OnActivityEnter 활동 스레드를 입력할 때 생성됩니다.
OnActivityLeave 작업 스레드가 남아 있을 때 생성됩니다.
OnActivityLeaveSame 재귀적으로 입력된 후 활동 스레드가 남아 있을 때 생성됩니다.
OnActivityReenter 활동 스레드가 재귀적으로 다시 입력될 때 생성됩니다.
OnActivityTimeout

활동에 대한 호출 시간이 초과될 때 생성됩니다.
OnAppActivation 애플리케이션 서버가 시작될 때 생성됩니다.
OnAppActivation2 서버 애플리케이션 프로세스가 로드될 때 생성됩니다.
OnAppForceShutdown 애플리케이션 서버를 강제로 종료할 때 생성됩니다.
OnAppForceShutdown2 서버 애플리케이션을 강제로 종료할 때 생성됩니다.
OnAppPaused2 서버 애플리케이션이 일시 중지되거나 초기 상태로 다시 시작될 때 생성됩니다.
OnAppRecycle2 서버 애플리케이션 프로세스가 재활용 종료로 표시될 때 생성됩니다.
OnAppShutdown 애플리케이션 서버가 종료될 때 생성됩니다.
OnAppShutdown2 서버 애플리케이션이 종료될 때 생성됩니다.
OnAuthenticate 메서드 호출 수준 인증이 성공하면 생성됩니다.
OnAuthenticateFail 메서드 호출 수준 인증에 실패할 때 생성됩니다.
OnCall 이 메서드에 구현된 일괄 처리 작업의 실행을 트리거합니다. (IMTSCall.OnCall)
OnCall 이 메서드에 구현된 일괄 처리 작업의 실행을 트리거합니다. (IServiceCall.OnCall)

OnCRMAbort CRM 클럭이 CRM 보상자에 전달할 중단 알림을 수신할 때 생성됩니다.
OnCRMAnalyze CRM 클럭이 복구의 분석 단계에서 레코드를 수신할 때 생성됩니다.
OnCRMBegin 클라이언트가 보상자를 등록하거나 복구하는 동안 CRM 클럭이 시작될 때 생성됩니다.
OnCRMCheckpoint CRM 검사점이 발생할 때 생성됩니다.
OnCRMCommit CRM 클럭이 CRM 보상자에 전달할 커밋 알림을 수신할 때 생성됩니다.
OnCRMDeliver CRM 클럭이 CRM 보상자에 레코드를 전달할 때 생성됩니다.
OnCRMDone CRM 클럭이 트랜잭션 결과 알림 처리를 완료할 때 생성됩니다.
OnCRMForce CRM 클럭이 CRM 작업자 또는 CRM 보상자에서 로그 레코드를 디스크에 강제로 적용하라는 요청을 수신할 때 생성됩니다.
OnCRMForget CRM 작업자 또는 CRM 보상자에서 로그 레코드를 잊어버리라는 요청을 받을 때 생성됩니다.
OnCRMIndoubt CRM 클럭이 CRM 보상자에게 전달할 의심스러운 알림을 수신할 때 생성됩니다.
OnCRMPrepare CRM 클럭이 CRM 보상자에 전달할 준비 알림을 수신할 때 생성됩니다.
OnCRMRecoveryDone CRM 복구가 완료될 때 생성됩니다.

OnCRMRecoveryStart CRM 복구가 시작될 때 생성됩니다.
OnCRMRelease CRM 클럭이 완료되고 리소스 잠금을 해제할 때 생성됩니다.
OnCRMWrite CRM 작업자 또는 CRM 보상자에서 로그 레코드 작성 요청을 받을 때 생성됩니다.
OnDataChanged COM+ 애플리케이션 instance 구성이 변경될 때 생성됩니다.
OnDisableCommit 클라이언트가 컨텍스트에서 DisableCommit을 호출할 때 생성됩니다.
OnEnableCommit 클라이언트가 컨텍스트에서 EnableCommit을 호출할 때 생성됩니다.
OnError 비동기 일괄 처리 작업에서 오류가 발생할 때 COM+에서 호출됩니다.
OnExceptionUser 사용자 코드에서 처리되지 않은 예외가 발생할 때 트랜잭션 구성 요소에 대해 생성됩니다.
OnIISRequestInfo 작업이 ASP 페이지의 일부일 때 생성됩니다.
OnLtxTransactionAbort 트랜잭션이 중단될 때 생성됩니다.
OnLtxTransactionCommit 트랜잭션이 커밋될 때 생성됩니다.
OnLtxTransactionPrepare COM+가 트랜잭션에 대한 준비 알림을 받을 때 생성됩니다.
OnLtxTransactionPromote

트랜잭션이 승격될 때 생성됩니다.
OnLtxTransactionStart 트랜잭션이 시작될 때 생성됩니다.
OnMethodCall 개체의 메서드가 호출될 때 생성됩니다. (IComMethodEvents.OnMethodCall)
OnMethodCall2 개체의 메서드가 호출될 때 생성됩니다. (IComMethod2Events.OnMethodCall2)
OnMethodException 개체의 메서드가 예외를 생성할 때 생성됩니다. (IComMethodEvents.OnMethodException)
OnMethodException2 개체의 메서드가 예외를 생성할 때 생성됩니다. (IComMethod2Events.OnMethodException2)
OnMethodReturn 개체의 메서드가 반환될 때 생성됩니다. (IComMethodEvents.OnMethodReturn)
OnMethodReturn2 개체의 메서드가 반환될 때 생성됩니다. (IComMethod2Events.OnMethodReturn2)
OnNewTrackingInfo 컬렉션에 대한 추적 정보가 변경될 때 생성됩니다.
OnObjectActivate 개체가 새 JIT 활성화 개체의 instance 가져오면 생성됩니다.
OnObjectConstruct 생성된 개체를 만들 때 생성됩니다. (IComObjectConstructionEvents.OnObjectConstruct)
OnObjectConstruct2 생성된 개체를 만들 때 생성됩니다. (IComObjectConstruction2Events.OnObjectConstruct2)
OnObjectCreate 클라이언트에서 개체를 만들 때 생성됩니다.

OnObjectCreate2 클라이언트가 개체를 만들 때 생성됩니다.
OnObjectDeactivate SETComplete 또는 SetAbort에서 JIT 활성화 개체를 해제할 때 생성됩니다.
OnObjectDestroy 클라이언트에서 개체를 해제할 때 생성됩니다.
OnObjectDestroy2 클라이언트가 개체를 해제할 때 생성됩니다.
OnObjPoolCreateDecision 풀이 요청 클라이언트에 기존 개체를 제공하거나 새 개체를 만들 때 생성됩니다.
OnObjPoolCreateObject 풀에 대한 개체를 만들 때 생성됩니다.
OnObjPoolCreatePool 새 풀을 만들 때 생성됩니다.
OnObjPoolDestroyObject 개체가 풀에서 영구적으로 제거될 때 생성됩니다.
OnObjPoolGetFromTx 트랜잭션 개체를 풀에서 가져올 때 생성됩니다. (IComObjectPoolEvents.OnObjPoolGetFromTx)
OnObjPoolGetFromTx2 트랜잭션 개체를 풀에서 가져올 때 생성됩니다. (IComObjectPool2Events.OnObjPoolGetFromTx2)
OnObjPoolGetObject 비 트랜잭션 개체를 풀에서 가져올 때 생성됩니다. (IComObjectPoolEvents.OnObjPoolGetObject)
OnObjPoolGetObject2 비 트랜잭션 개체를 풀에서 가져올 때 생성됩니다. (IComObjectPool2Events.OnObjPoolGetObject2)

OnObjPoolPutObject 새 개체가 풀에 추가될 때 생성됩니다.
OnObjPoolPutObject2 개체가 풀에 추가될 때 생성됩니다.
OnObjPoolRecycleToTx 트랜잭션 개체가 풀에 반환될 때 생성됩니다. (IComObjectPoolEvents.OnObjPoolRecycleToTx)
OnObjPoolRecycleToTx2 트랜잭션 개체가 풀에 반환될 때 생성됩니다. (IComObjectPool2Events.OnObjPoolRecycleToTx2)
OnObjPoolTimeout 풀된 개체에 대한 요청 시간이 초과될 때 생성됩니다.
OnQCMoveToDeadQueue 메시지가 배달 못 한 편지 큐로 이동되어 배달할 수 없는 경우에 생성됩니다.
OnQCMoveToReTryQueue 메시지가 큐에 대기 중인 구성 요소로 이동될 때 생성되며 큐를 다시 시도합니다.
OnQCPlayback 메시지 내용이 재생될 때 생성됩니다.
OnQCQueueOpen 큐에 대기 중인 구성 요소 큐가 열릴 때 생성됩니다.
OnQCReceive 큐에 대기 중인 구성 요소 서비스에서 내용에 문제가 있을 수 있는 경우에도 메시지가 큐에서 성공적으로 제거될 때 생성됩니다.
OnQCReceiveFail 수신 메시지가 실패할 때 생성됩니다.
OnQCRecord 큐에 대기된 구성 요소 레코더가 큐에 대기된 메시지를 만들 때 생성됩니다.

OnResourceAllocate 기존 리소스가 할당될 때 생성됩니다.
OnResourceCreate 새 리소스를 만들고 할당할 때 생성됩니다.
OnResourceDestroy 리소스가 제거될 때 생성됩니다.
OnResourceRecycle 개체가 리소스로 완료되면 생성됩니다.
OnResourceTrack 리소스를 추적할 때 생성됩니다.
OnSetComplete 클라이언트가 컨텍스트에서 SetComplete를 호출할 때 생성됩니다.
OnThreadAssignApartment 활동이 아파트 스레드에 할당될 때 생성됩니다.
OnThreadBindToApartment 아파트 스레드가 실행할 아파트 스레드가 없는 STA(단일 스레드 아파트) 스레드에 할당될 때 생성됩니다.
OnThreadStart STA(단일 스레드 아파트) 스레드가 시작될 때 생성됩니다.
OnThreadTerminate STA(단일 스레드 아파트) 스레드가 종료될 때 생성됩니다.
OnThreadUnassignApartment 아파트 스레드에서 활동을 할당 취소할 때 생성됩니다.
OnThreadUnBind 구성된 구성 요소의 수명이 초과되고 아파트 스레드의 작업 수를 감소할 수 있을 때 생성됩니다.

OnTransactionAbort 트랜잭션이 중단될 때 생성됩니다. (IComTransactionEvents.OnTransactionAbort)
OnTransactionAbort2 트랜잭션이 중단될 때 생성됩니다. (IComTransaction2Events.OnTransactionAbort2)
OnTransactionCommit 트랜잭션이 커밋될 때 생성됩니다. (IComTransactionEvents.OnTransactionCommit)
OnTransactionCommit2 트랜잭션이 커밋될 때 생성됩니다. (IComTransaction2Events.OnTransactionCommit2)
OnTransactionPrepare 트랜잭션의 2단계 커밋 프로토콜 준비 단계가 완료될 때 생성됩니다.
OnTransactionPrepare2 트랜잭션이 커밋 프로토콜의 준비 단계에 있을 때 생성됩니다.
OnTransactionStart Microsoft DTC(Distributed Transaction Coordinator) 트랜잭션이 시작될 때 생성됩니다. (IComTransactionEvents.OnTransactionStart)
OnTransactionStart2 Microsoft DTC(Distributed Transaction Coordinator) 트랜잭션이 시작될 때 생성됩니다. (IComTransaction2Events.OnTransactionStart2)
OnUserEvent 사용자 정의 이벤트를 생성하기 위해 사용자 구성 요소에 대해 제공됩니다.
PartitionConfig 묶은 작업에 파티션을 사용하는 방법을 구성합니다.
PartitionID 묶은 작업에 사용되는 파티션의 GUID를 설정합니다.
PauseApplicationInstances 지정된 애플리케이션 서버 프로세스를 일시 중지합니다.

채우기 컬렉션에 포함된 모든 항목에 대한 데이터로 컬렉션을 채웁니다.
PopulateByKey 지정된 키에 따라 COM+ 카탈로그에서 컬렉션의 선택한 항목 목록을 채웁니다.
PopulateByQuery 다음에 사용하도록 예약됩니다. (ICatalogCollection.PopulateByQuery)
PostEvent 사용자 정의 이벤트를 이벤트 싱크에 게시합니다.
PrepareRecord 준비 단계 중에 전달 순서에 따라 로그 레코드를 전달합니다.
PrepareRecordVariants 준비 단계 중에 CRM Compensator에 로그 레코드를 전달합니다.
PrepareToFire 제공된 실행 컨트롤을 사용하여 필터링된 구독 목록을 실행하기 시작하도록 게시자 필터를 준비합니다. 실행 컨트롤은 이벤트 클래스 개체에 포함됩니다.
PrepareToFire 제공된 실행 컨트롤을 사용하여 필터링된 구독 목록을 실행하기 시작하도록 게시자 필터를 준비합니다. 실행 컨트롤은 이벤트 클래스 개체에 포함됩니다.
ProgIdCompensator 지정된 인덱스에 대한 CRM Compensator의 ProgId를 검색합니다.
Promote DTC가 아닌 트랜잭션을 DTC 트랜잭션으로 승격합니다.
PromoteUnconfiguredComponents 구성되지 않은 구성 요소에서 구성된 구성 요소로 지정된 클래스를 승격합니다.
PropCount 추적 정보 개체에 대해 정의된 속성 수를 검색합니다.

put_AllowInprocActivation 이벤트 클래스를 프로세스에서 활성화할 수 있는지 여부를 나타냅니다. (Put)
put_AllowInprocActivation 게시자의 프로세스에서 구독자를 활성화할 수 있는지 여부를 나타냅니다. (IEventControl.put_AllowInprocActivation)
put_AllowInprocActivation 게시자의 프로세스에서 구독자를 활성화할 수 있는지 여부를 나타냅니다. (IMultiInterfaceEventControl.put_AllowInprocActivation)
put_ClassFactory 풀된 개체에 대한 클래스 팩터리를 설정합니다.
put_CommitBatchSize 커밋 일괄 처리 크기를 설정합니다. 커밋 작업 간에 원본 큐에서 대상 큐로 이동해야 하는 메시지 수입니다.
put_CreationTimeout 풀링된 개체를 활성화하기 위한 제한 시간 간격을 설정합니다.
put_CurrentPartition 현재 대상 파티션을 설정합니다.
put_CustomConfigCLSID 구독 개체의 속성 모음에 속성을 추가하는 데 도움이 될 수 있는 구성 요소의 CLSID입니다. 이 속성은 이전 버전과의 호환성에 대해서만 지원됩니다. (Put)
put_Description 이벤트 클래스 개체의 표시 가능한 텍스트 설명입니다. (Put)
put_Description 구독에 대한 표시 가능한 텍스트 설명입니다. (Put)
put_DestPath 대상(출력) 큐의 경로를 설정합니다.

put_Enabled 구독을 사용할 수 있는지 여부를 나타냅니다. (Put)
put_EventClassID 이벤트 클래스 개체의 CLSID입니다. (Put)
put_EventClassID 구독과 연결된 이벤트 클래스의 고유 ID입니다. (Put)
put_EventClassName 이벤트 클래스 개체의 ProgID입니다. (Put)
put_FireInParallel 이 클래스의 이벤트를 병렬로 실행할 수 있는지 여부를 나타냅니다. (Put)
put_FireInParallel 이벤트를 두 명 이상의 구독자에게 병렬로 전달할 수 있는지 여부를 나타냅니다. (Put)
put_FiringInterfaceID 이벤트 클래스 개체의 이벤트 인터페이스 ID입니다. 이 속성은 이전 버전과의 호환성에 대해서만 지원됩니다. (Put)
put_InterfaceID 구독자가 이벤트를 수신하려는 특정 인터페이스의 식별자입니다. (Put)
put_MachineName 구독자를 활성화해야 하는 컴퓨터의 이름입니다(영구 구독의 경우). (Put)
put_MaxPoolSize 풀의 최대 개체 수를 설정합니다.
put_MethodName 이벤트 메서드의 이름입니다. (Put)
put_MinPoolSize 풀의 최소 개체 수를 설정합니다.

put_MultiInterfacePublisherFilterCLSID IMultiInterfacePublisherFilter를 구현하는 개체의 CLSID입니다. (Put)
put_OwnerSID 이벤트 클래스 개체 작성자의 보안 ID입니다. 이 속성은 이전 버전과의 호환성에 대해서만 지원됩니다. (Put)
put_OwnerSID 구독 작성자의 보안 ID입니다. (Put)
put_PerUser 구독 소유자가 게시자와 동일한 컴퓨터에 로그온한 경우에만 구독이 이벤트를 수신하는지 여부를 나타냅니다. (Put)
put_PublisherID 이벤트 게시자에 대한 CLSID입니다. (Put)
put_PublisherID 이벤트 게시자의 고유 ID입니다. (Put)
put_SourcePath 원본(입력) 큐의 경로를 설정합니다.
put_SubscriberCLSID 구독자 구성 요소의 CLSID입니다(영구 구독의 경우). (Put)
put_SubscriberInterface 구독자의 이벤트 인터페이스에 대한 마샬링된 포인터입니다(임시 구독의 경우). (Put)
put_SubscriptionID 구독 개체의 고유 ID입니다. (Put)
put_SubscriptionName 구독 개체의 표시 가능한 이름입니다. (Put)
put_TransactionAffinity 트랜잭션이 완료될 때까지 트랜잭션에 관련된 개체를 보유할지 여부를 설정합니다.

put_TypeLib 이벤트 인터페이스에 대한 설명을 포함하는 형식 라이브러리의 경로입니다. (Put)
put_Value 공유 속성의 값을 설정합니다.
put_Value 이 카탈로그 개체에 의해 노출되는 지정된 속성의 값에 액세스합니다. (Put)
PutEndTx 트랜잭션이 종료되면 풀된 개체를 삭제합니다.
PutPublisherProperty 게시자 컨텍스트를 정의하기 위해 속성 모음에 속성 및 해당 값을 씁니다.
PutResource 풀된 개체 목록에 개체를 추가합니다.
PutSubscriberProperty 구독자 컨텍스트를 정의하기 위해 속성 모음에 속성 및 해당 값을 씁니다.
쿼리 이벤트 데이터 저장소에서 구독 또는 이벤트 개체의 컬렉션을 검색합니다. (IEventSystem.Query)
QueryApplicationFile 애플리케이션 파일에서 COM+ 애플리케이션에 대한 정보를 검색합니다.
QueryApplicationFile2 설치할 애플리케이션에 대한 정보를 검색합니다.
쿼리 이벤트 데이터 저장소에서 구독 또는 이벤트 개체의 컬렉션을 검색합니다. (IEventSystem.QueryS)
RateResource 후보 리소스가 얼마나 잘 일치하는지 평가합니다.
RecycleApplicationInstances

지정된 애플리케이션 서버 프로세스를 재활용(종료 및 다시 시작)합니다.
RecycleSurrogate 호출 프로세스를 재활용합니다.
새로 고침 개체를 만들 때 채워진 IUnknown 포인터 목록을 업데이트.
RefreshComponents 레지스트리에서 구성 요소 등록 정보를 업데이트.
RefreshRouter 이 메서드는 사용되지 않습니다.
RegisterCompensator CRM 작업자는 이 메서드를 사용하여 CRM 인프라에 CRM Compensator를 등록합니다.
RegisterDispenser 리소스 디스펜서를 디스펜서 관리자에 등록합니다.
ReleaseSID 다른 ISecurityProperty 메서드 중 하나에서 반환된 보안 식별자를 해제합니다.
제거 컬렉션에서 이벤트 개체를 제거합니다.
제거 이벤트 데이터 저장소에서 하나 이상의 구독 또는 이벤트 개체를 제거합니다. (IEventSystem.Remove)
제거 인덱스가 지정된 경우 컬렉션에서 항목을 제거하고 인덱스 값이 더 높은 항목을 다시 인덱싱합니다.
RemoveProperty 컨텍스트 개체 속성을 제거합니다.
RemovePublisherProperty

게시자 컨텍스트를 정의하는 속성 모음에서 속성 및 해당 값을 제거합니다.
제거 이벤트 데이터 저장소에서 하나 이상의 구독 또는 이벤트 개체를 제거합니다. (IEventSystem.RemoveS)
RemoveSubscriberProperty 구독자 컨텍스트를 정의하는 속성 모음에서 속성 및 해당 값을 제거합니다.
RequestDestroyResource 리소스를 삭제하고 소멸자를 호출하여 메모리 및 기타 연결된 시스템 리소스를 해제합니다.
Reset 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumNames.Reset)
Reset 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumEventObject.Reset)
ResetResource 일반 또는 인벤토리에 다시 넣을 리소스를 준비합니다.
RestoreREGDB 지정된 파일에서 COM+ 클래스 등록 데이터베이스(RegDB)를 복원합니다. 이를 적용하려면 시스템 다시 부팅이 필요합니다.
ResumeApplicationInstances 지정된 애플리케이션 서버 프로세스를 다시 시작합니다.
SafeRef SafeRef 함수
SaveChanges 컬렉션 및 컬렉션에 포함된 항목에 대한 보류 중인 모든 변경 내용을 COM+ 카탈로그 데이터 저장소에 저장합니다.
SelectThreadPool 작업을 통해 제출된 작업이 실행될 스레드 풀을 선택합니다.

SendMethodCall 구성 요소 인터페이스를 통해 메서드를 호출할 때 생성됩니다.
SendMethodReturn 구성 요소 인터페이스를 통해 호출된 메서드가 호출자에게 컨트롤을 반환할 때 생성됩니다.
ServiceCheck 지정된 COM+ 서비스의 현재 상태 검색합니다.
SetAbort 개체가 실행 중인 트랜잭션을 중단해야 하며 현재 실행 중인 메서드 호출에서 반환될 때 개체가 비활성화되어야 한다고 선언합니다.
SetAbort 개체가 실행 중인 트랜잭션을 중단해야 하며 반환할 때 개체가 비활성화되어야 한다고 선언합니다.
SetBindingInfo 활동에서 제출한 모든 작업을 단일 스레드 아파트로 바인딩합니다.
SetCLSID 개체의 CLSID를 설정합니다.
SetComplete 개체가 실행 중인 트랜잭션을 커밋할 수 있으며 현재 실행 중인 메서드 호출에서 반환될 때 개체가 비활성화되어야 한다고 선언합니다.
SetComplete 개체가 실행 중인 트랜잭션을 커밋할 수 있고 반환할 때 개체를 비활성화해야 한다고 선언합니다.
SetDeactivateOnReturn 개체가 메서드 반환에서 비활성화되는지 여부를 제어하는 완료 플래그를 설정합니다.
SetDefaultQuery 구독자를 결정하는 기본 쿼리를 설정합니다.
SetDefaultQuery 게시자 필터가 이벤트 메서드와 연결되지 않은 경우 사용할 기본 쿼리를 설정합니다.

SetDeleteDelay 풀링된 스레드가 삭제되기 전에 유휴 상태일 수 있는 시간(밀리초)을 설정합니다.
SetHeld 관리되는 개체가 COM+ 개체 풀로 돌아갈지 여부를 설정합니다.
SetInPool 관리되는 개체가 COM+ 개체 풀에 속하는지 여부를 설정합니다.
SetLogControl 트랜잭션이 완료되는 동안 추가 로그 레코드를 작성할 수 있도록 CRM Compensator에 ICrmLogControl 인터페이스를 제공합니다.
SetLogControlVariants CRM Compensator에 ICrmLogControl 인터페이스를 제공합니다.
SetMachineName 부하 분산 서버의 컴퓨터 이름을 설정합니다.
SetMaxQueuedRequests 동시에 큐에 대기할 수 있는 비동기 실행 요청의 최대 수를 설정합니다.
SetMaxThreads 풀에서 허용되는 최대 스레드 수를 설정합니다.
SetMinThreads 풀에서 유지 관리할 최소 스레드 수를 설정합니다.
SetMultiInterfacePublisherFilter 런타임에 이벤트 메서드에 게시자 필터를 할당합니다.
SetMyTransactionVote 일관된 플래그를 설정합니다.
SetProperty 컨텍스트 개체 속성을 설정합니다.

SetPublisherFilter 이벤트 메서드에 게시자 필터를 할당합니다.
SetQueueDepth 풀에 새 스레드가 추가되는 위의 실행 요청 임계값을 설정합니다.
SetTransactionStatus 트랜잭션 상태 커밋되거나 중단됨으로 설정합니다. 이 메서드를 사용하지 마세요. COM+에서 내부적으로만 사용됩니다.
SetWrapperStrength 관리되는 개체가 COM+ 컨텍스트에 대한 강력한 참조 또는 약한 참조를 보유하는지 여부를 설정합니다.
종료 Dllhost.exe 종료할 때 호출됩니다.
종료 개체 풀을 종료합니다.
ShutdownApplication COM+ 서버 애플리케이션 프로세스의 종료를 시작합니다.
ShutdownApplicationInstances 지정된 애플리케이션 서버 프로세스의 종료를 시작합니다.
ShutdownPool 개체 풀을 종료합니다.
Skip 열거형 시퀀스에서 지정된 수의 항목을 건너뜁니다. (IEnumNames.Skip)
Skip 열거형 시퀀스에서 지정된 수의 항목을 건너뜁니다. (IEnumEventObject.Skip)
StartApplication

지정된 COM+ 서버 애플리케이션을 시작합니다. 애플리케이션 구성 요소는 전용 서버 프로세스에서 시작됩니다.
StartRouter 서비스가 현재 설치된 경우 구성 요소 부하 분산 서비스를 시작합니다.
Startup 클래스 Dllhost.exe 시작할 때 호출됩니다.
Startup 클래스 COM+ 애플리케이션 instance 만들 때 호출됩니다.
StopRouter 서비스가 현재 설치된 경우 구성 요소 부하 분산 서비스를 중지합니다.
Store 이벤트 시스템 내에서 이벤트 또는 구독 개체를 만들거나 수정합니다.
SxsConfig 묶은 작업에 대한 병렬 어셈블리를 구성합니다.
SxsDirectory 묶은 작업의 병렬 어셈블리에 대한 디렉터리를 설정합니다.
SxsName 묶은 작업에 대한 병렬 어셈블리의 파일 이름을 설정합니다.
SynchronousCall 사용자 정의 작업을 동기적으로 수행합니다. (IMTSActivity.SynchronousCall)
SynchronousCall 사용자 정의 작업을 동기적으로 수행합니다. (IServiceActivity.SynchronousCall)
TrackerConfig 묶은 작업에 대한 추적기 속성을 구성합니다.
TrackResource 리소스를 추적합니다.

TrackResources 리소스(문자열 버전)를 추적합니다.
TransactionTimeout 새 트랜잭션에 대한 트랜잭션 제한 시간을 설정합니다.
TransactionUOW 지정된 인덱스에 대한 트랜잭션의 작업 단위(UOW)를 검색합니다.
형식 추적 정보 컬렉션의 형식을 검색합니다.
UnbindFromThread 실행 중인 스레드에서 IMTSActivity::AsyncCall 또는 IMTSActivity::SynchronousCall을 사용하여 제출된 일괄 처리 작업을 바인딩 해제합니다.
UnbindFromThread 실행 중인 스레드에서 사용자 정의 일괄 처리 작업을 바인딩 해제합니다.
UntrackResource 리소스 추적을 중지합니다.
UntrackResources 리소스(문자열 버전) 추적을 중지합니다.
WriteLogRecord CRM 작업자 및 CRM Compensator는 이 메서드를 사용하여 비정형 로그 레코드를 로그에 씁니다.
WriteLogRecordVariants CRM 작업자 및 CRM Compensator는 이 메서드를 사용하여 구조적 로그 레코드를 로그에 씁니다.

인터페이스

ContextInfo

현재 컨텍스트 개체에 대한 트랜잭션, 작업 및 컨텍스트 정보를 검색합니다. 이 인터페이스의 메서드를 사용하여 개체 컨텍스트 내에 포함된 관련 정보를 검색할 수 있습니다.
ContextInfo2 ContextInfo 인터페이스를 통해 사용할 수 있는 정보를 보완하여 개체의 컨텍스트에 대한 추가 정보를 제공합니다.
IAppDomainHelper 애플리케이션이 실행되는 격리된 환경인 애플리케이션 도메인에 관리되는 개체를 바인딩합니다.
IAssemblyLocator .NET Framework 공용 언어 런타임에서 관리 코드를 사용할 때 어셈블리에 대한 정보를 검색합니다.
IAsyncErrorNotify CoCreateActivity에서 만든 작업을 통해 제출되는 비동기 일괄 처리 작업에서 오류 트래핑을 구현하는 데 사용됩니다.
ICatalogCollection COM+ 카탈로그의 모든 컬렉션을 나타냅니다. ICatalogCollection을 사용하면 컬렉션의 항목을 열거, 추가, 제거 및 검색하고 관련 컬렉션에 액세스할 수 있습니다.
ICatalogObject COM+ 카탈로그의 컬렉션에 있는 항목을 나타냅니다. ICatalogObject를 사용하면 카탈로그의 개체에 의해 노출되는 속성을 가져와 배치할 수 있습니다.
ICheckSxsConfig 현재 병렬 어셈블리의 구성을 검사 데 사용됩니다.
IComActivityEvents 활동이 생성, 제거 또는 시간 초과된 경우 구독자에게 알립니다.
ICOMAdminCatalog 프로그래밍 방식의 COM+ 관리를 수행하고, 카탈로그의 컬렉션에 액세스하고, COM+ 애플리케이션 및 구성 요소를 설치하고, 서비스를 시작 및 중지하고, 원격 서버에 연결하는 세션을 시작합니다.
ICOMAdminCatalog2 ICOMAdminCatalog 인터페이스의 확장입니다.

IComApp2Events COM+ 서버 애플리케이션이 로드, 종료 또는 일시 중지된 경우 구독자에게 알합니다.
IComAppEvents COM+ 서버 애플리케이션이 시작, 종료 또는 강제로 종료될 경우 구독자에게 알합니다.
IComCRMEvents 구성 요소 서비스의 CRM(보상 Resource Manager) 기능의 활동에 대해 구독자에게 알합니다.
IComExceptionEvents 사용자 코드에서 처리되지 않은 예외가 발생할 때 구독자에게 알린 것입니다.
IComIdentityEvents IIS(인터넷 정보 서비스) ASP(활성 서버 페이지) 페이지의 일부인 활동에 대해 구독자에게 알깁니다. 예를 들어 ASP 페이지에서 COM+ 개체가 호출되면 사용자에게 이 작업에 대한 알림이 표시됩니다.
IComInstance2Events 클라이언트에서 개체를 만들거나 해제하는 경우 구독자에게 알 수 있습니다.
IComInstanceEvents 개체의 생성 또는 릴리스를 구독자에게 알릴 수 있습니다.
ICOMLBArguments COM+ 구성 요소 부하 분산 서비스를 활성화하는 데 사용됩니다.
IComLTxEvents COM+ 트랜잭션과 관련된 이벤트를 구독자에게 알 수 있습니다.
IComMethod2Events 개체의 메서드가 호출, 반환 또는 예외를 생성했는지 구독자에게 알립니다. (IComMethod2Events)
IComMethodEvents 개체의 메서드가 호출, 반환 또는 예외를 생성했는지 구독자에게 알립니다. (IComMethodEvents)
IComObjectConstruction2Events 생성된 개체가 만들어지면 구독자에게 알 수 있습니다.

IComObjectConstructionEvents 생성된 개체가 개체 풀에 만들어지는 경우 구독자에게 알 수 있습니다.
IComObjectEvents JIT(Just-In-Time) 활성화 개체의 instance 만들거나 해제했는지 구독자에게 알합니다.
IComObjectPool2Events 개체 풀에서 트랜잭션 또는 비 트랜잭션 개체가 추가되거나 가져올 경우 구독자에게 알 수 있습니다.
IComObjectPoolEvents 새 개체가 풀에 추가되면 구독자에게 알 수 있습니다.
IComObjectPoolEvents2 풀에서 새 개체를 만들거나 제거할 때 구독자에게 알합니다.
IComQCEvents 큐에 대기 중인 메시지가 만들어지거나, 큐에서 해제되거나, 재시도 또는 배달 못한 편지 큐로 이동되면 구독자에게 알 수 있습니다.
IComResourceEvents 리소스가 생성, 할당, 추적 또는 제거되면 구독자에게 알립니다.
IComSecurityEvents 메서드 호출의 인증이 성공했거나 실패한 경우 구독자에게 알합니다.
IComThreadEvents STA(단일 스레드 아파트)가 만들어지거나 종료되고 아파트 스레드가 할당될 때 구독자에게 알 수 있습니다.
IComTrackingInfoCollection 추적 정보 수집에 대한 정보를 검색합니다.
IComTrackingInfoEvents 컬렉션에 대한 추적 정보가 변경되면 구독자에게 알 수 있습니다.
IComTrackingInfoObject

추적 정보 개체의 속성을 검색합니다.
IComTrackingInfoProperties 추적 정보 개체 및 해당 이름과 연결된 총 속성 수를 검색합니다.
IComTransaction2Events Microsoft DTC(Distributed Transaction Coordinator) 트랜잭션이 시작, 커밋 또는 중단되면 구독자에게 알합니다. 트랜잭션이 2단계 커밋 프로토콜의 준비 단계에 있을 때 구독자에게도 알림이 표시됩니다.
IComTransactionEvents Microsoft DTC(Distributed Transaction Coordinator) 트랜잭션이 시작, 커밋 또는 중단되면 구독자에게 알합니다.
IComUserEvent 지정된 사용자 정의 메트릭의 구독자에게 알 수 있습니다.
IContextProperties 컨텍스트 개체 속성에 대한 액세스를 제공합니다.
IContextState 컨텍스트 상태 플래그를 조작하여 개체 비활성화 및 트랜잭션 투표를 제어합니다.
ICreateWithLocalTransaction 지정된 로컬 트랜잭션의 scope 내에서 실행되는 COM+ 개체를 만듭니다. (ICreateWithLocalTransaction)
ICreateWithTipTransactionEx TIP(Transaction Internet Protocol)를 사용하는 수동 트랜잭션 내에 참여하는 개체를 만듭니다.
ICreateWithTransactionEx 수동 트랜잭션 내에 참여하는 개체를 만듭니다.
ICrmCompensator Microsoft Visual C++ 사용할 때 구조화되지 않은 로그 레코드를 CRM Compensator에 전달합니다.
ICrmCompensatorVariants Microsoft Visual Basic을 사용할 때 CRM Compensator에 구조화된 로그 레코드를 전달합니다.

ICrmFormatLogRecords 로그 레코드를 일반 모니터링 도구를 사용하여 표시할 수 있도록 볼 수 있는 형식으로 변환합니다.
ICrmLogControl CRM 작업자 및 CRM Compensator가 로그에 레코드를 쓰고 지속성으로 만드는 수단입니다.
ICrmMonitor CRM의 현재 상태 스냅샷 캡처하고 특정 CRM 클럭을 보유합니다.
ICrmMonitorClerks 서기의 상태에 대한 정보를 검색합니다.
ICrmMonitorLogRecords 특정 CRM 서기가 지정된 트랜잭션에 대해 유지 관리하는 개별 로그 레코드를 모니터링합니다.
IDispenserDriver COM+ 리소스 디스펜서의 소유자에 의해 호출되어 리소스를 만들고, 참여시키고, 평가하고, 준비하고, 삭제합니다.
IDispenserManager 디스펜서 관리자에 연결합니다.
IEnumEventObject COM+ 이벤트 저장소에 등록된 이벤트 개체를 열거합니다.
IEnumNames 이름을 열거합니다.
IEventClass 이벤트 개체의 클래스를 해당 개체가 구현하는 이벤트 인터페이스와 연결합니다.
IEventClass2 이벤트 클래스 개체에 대한 데이터를 설정하고 가져오는 데 사용됩니다. 이 인터페이스는 IEventClass 인터페이스를 확장합니다.
IEventControl 이벤트를 구독자에게 발생시키는 개체인 이벤트 개체의 동작을 제어합니다. (IEventControl)

IEventObjectChange 구독자에게 이벤트 저장소에 대한 변경 내용을 알 수 있습니다.
IEventObjectChange2 파티션 및 애플리케이션 ID 정보를 포함하면서 구독자에게 이벤트 저장소에 대한 변경 내용을 알 수 있습니다.
IEventObjectCollection 이벤트 개체 컬렉션의 개체를 관리합니다.
IEventSubscription 이벤트 구독자와 구독 중인 이벤트 간의 관계에 대한 정보를 지정합니다. 게시자 필터에서 사용됩니다.
IEventSystem 이벤트 데이터 저장소에 대한 액세스를 제공합니다.
IFiringControl 단일 구독에 이벤트를 발생합니다.
IGetAppTrackerData 관리 애플리케이션이 COM+ 애플리케이션 실행에 대한 통계 정보를 검색할 수 있도록 합니다.
IGetContextProperties 호출자가 현재 개체의 컨텍스트와 연결된 속성을 가져올 수 있도록 합니다.
IGetSecurityCallContext 현재 호출과 연결된 SecurityCallContext 클래스에서 만든 개체에 대한 참조를 검색합니다. (IGetSecurityCallContext)
IHolder 설치된 리소스 디스펜서에 대한 리소스를 할당하거나 해제합니다.
IManagedActivationEvents 현재 COM+ 컨텍스트 내에서 관리되는 개체에 대한 스텁을 만들고 삭제하는 데 사용됩니다.
IManagedObjectInfo

관리되는 개체의 스텁에 대해 설명합니다.
IManagedPoolAction COM+ 개체 풀에서 개체를 해제하기 전에 알림을 받을 수 있도록 합니다.
IManagedPooledObj COM+ 개체 풀에서 관리되는 개체를 사용하는 방법을 설명합니다.
IMessageMover 메시지를 한 큐에서 다른 큐로 이동합니다.
IMTSActivity MTSCreateActivity 함수에서 만든 작업을 통해 일괄 처리 작업을 제출합니다.
IMTSCall MTSCreateActivity 함수에서 만든 작업을 통해 제출되는 일괄 처리 작업을 구현합니다.
IMtsEventInfo 사용자 정의 이벤트를 설명합니다.
IMtsEvents 실행 중인 패키지에 대한 정보를 가져오고 이벤트 싱크를 설정하는 메서드를 제공합니다.
IMtsGrp 실행 중인 패키지를 열거하는 메서드를 제공합니다.
IMTSLocator 현재 프로세스에 대한 이벤트 디스패처의 IMtsEvents 인터페이스에 대한 액세스를 제공하는 단일 이벤트를 설명합니다.
IMultiInterfaceEventControl 이벤트를 구독자에게 발생시키는 개체인 이벤트 개체의 동작을 제어합니다. (IMultiInterfaceEventControl)
IMultiInterfacePublisherFilter 이벤트 메서드에 대한 필터링된 구독 캐시를 관리합니다.
IObjectConstruct

다른 메서드 또는 개체의 매개 변수를 전달하여 개체 생성 프로세스를 제어합니다.
IObjectConstructString 생성자 문자열에 대한 액세스를 제공합니다. 개체를 생성하는 동안 매개 변수를 지정하려는 경우 사용합니다.
IObjectContext 현재 개체의 컨텍스트에 대한 액세스를 제공합니다. 개체의 컨텍스트는 트랜잭션을 사용하거나 개체의 보안을 처리할 때 주로 사용됩니다. (IObjectContext)
IObjectContextActivity 현재 개체 컨텍스트와 연결된 활동 식별자를 검색합니다.
IObjectContextInfo 현재 컨텍스트 개체에 대한 트랜잭션, 작업 및 컨텍스트 정보를 검색합니다.
IObjectContextInfo2 개체의 컨텍스트에 대한 추가 정보를 제공합니다. 이 인터페이스는 IObjectContextInfo 인터페이스를 확장합니다.
IObjectContextTip TIP(트랜잭션 인터넷 프로토콜) 트랜잭션 컨텍스트를 설명하는 속성을 검색합니다.
IObjectControl COM+ 개체에 대한 컨텍스트별 초기화 및 정리 프로시저를 정의하고 개체를 재활용할 수 있는지 여부를 지정합니다.
IObjPool 트랜잭션 리소스 풀의 각 개체에 대한 키를 나타냅니다.
IPlaybackControl 메시지 큐 배달 메커니즘의 서버 쪽 재생 오류 및 클라이언트 쪽 오류의 비정상적인 처리에 참여할 수 있습니다.
IPoolManager 호출자가 개체 풀을 제어할 수 있도록 합니다.
IProcessInitializer

Dllhost.exe 시작하거나 종료할 때마다 호출할 수 있는 메서드를 제공합니다.
IPublisherFilter 이벤트 게시자가 이벤트 알림을 받는 구독자 또는 구독자에게 알림을 받는 순서를 제어할 수 있도록 콜백 인터페이스 역할을 합니다.
ISecurityCallContext 현재 호출의 보안 호출 컨텍스트에 대한 보안 메서드 및 정보에 대한 액세스를 제공합니다.
ISecurityCallersColl 호출자 컬렉션의 개별 호출자에 대한 정보에 대한 액세스를 제공합니다.
ISecurityIdentityColl 호출자의 ID를 나타내는 보안 정보 컬렉션에 대한 액세스를 제공합니다. 이 컬렉션에서 사용할 수 있는 항목은 SID, 계정 이름, 인증 서비스, 인증 수준 및 가장 수준입니다.
ISecurityProperty 현재 개체의 원래 호출자 또는 직접 호출자의 보안 식별자를 결정합니다. 그러나 개체의 호출자에 대한 정보를 가져오는 기본 방법은 ISecurityCallContext 인터페이스를 사용하는 것입니다.
ISelectCOMLBServer COM+ 구성 요소 부하 분산 서비스를 활성화합니다.
ISendMethodEvents 구현하는 개체의 메서드가 호출되거나 호출에서 반환할 때마다 구독자에게 알립니다.
IServiceActivity CoCreateActivity에서 만든 작업을 통해 제출되는 일괄 처리 작업을 호출하는 데 사용됩니다.
IServiceCall CoCreateActivity에서 만든 작업을 통해 제출되는 일괄 처리 작업을 구현하는 데 사용됩니다.
IServiceComTIIntrinsicsConfig CoCreateActivity 또는 CoEnterServiceDomain 함수를 호출할 때 수행되는 작업에 대한 COMTI(COM Transaction Integrator) 내장 함수를 구성합니다.
IServiceIISIntrinsicsConfig

CoCreateActivity 또는 CoEnterServiceDomain 함수를 호출할 때 수행되는 작업에 대한 IIS 내장 함수를 구성합니다.
IServiceInheritanceConfig 현재 컨텍스트를 기반으로 새 컨텍스트를 생성할지 또는 CServiceConfig의 정보만을 기반으로 새 컨텍스트를 만들 것인지 결정합니다.
IServicePartitionConfig CoCreateActivity 또는 CoEnterServiceDomain을 호출할 때 수행되는 작업에 파티션을 사용하는 방법을 구성합니다.
IServicePool COM+ 개체 풀을 관리하는 데 사용됩니다.
IServicePoolConfig 개체 풀을 구성하는 데 사용됩니다.
IServiceSxsConfig CoCreateActivity 또는 CoEnterServiceDomain을 호출할 때 수행되는 작업에 대해 병렬 어셈블리를 구성합니다.
IServiceSynchronizationConfig CoCreateActivity 또는 CoEnterServiceDomain을 호출할 때 수행되는 작업에 대한 동기화를 구성합니다.
IServiceSysTxnConfig 트랜잭션 프록시를 사용하여 지정한 기존 트랜잭션의 scope 코드 집합을 실행할 수 있습니다.
IServiceThreadPoolConfig CoCreateActivity를 호출하여 반환되는 활동 개체의 스레드 풀을 구성합니다.
IServiceTrackerConfig CoCreateActivity 또는 CoEnterServiceDomain을 호출할 때 수행되는 작업에 대한 추적기 속성을 구성합니다.
IServiceTransactionConfig CoCreateActivity 또는 CoEnterServiceDomain을 호출할 때 수행되는 작업에 대한 트랜잭션 서비스를 구성합니다. (IServiceTransactionConfig)

IServiceTransactionConfigBase CoCreateActivity 또는 CoEnterServiceDomain을 호출할 때 수행되는 작업에 대한 트랜잭션 서비스를 구성합니다. (IServiceTransactionConfigBase)
ISharedProperty 공유 속성의 값을 설정하거나 검색하는 데 사용할 수 있는 속성 메서드를 노출합니다.
ISharedPropertyGroup 공유 속성 그룹의 공유 속성을 만들고 액세스하는 데 사용됩니다.
ISharedPropertyGroupManager 공유 속성 그룹을 만들고 기존 공유 속성 그룹에 대한 액세스 권한을 얻는 데 사용됩니다.
ISystemAppEventData COM+ 애플리케이션 instance 만들거나 다시 구성할 때 구독자에게 알 수 있습니다.
IThreadPoolKnobs 스레드 풀의 동작을 제어하는 데 사용됩니다.
ITransactionContext 단일 트랜잭션에서 여러 COM+ 개체의 작업을 작성하고 트랜잭션을 명시적으로 커밋하거나 중단할 수 있습니다.
ITransactionContextEx 트랜잭션을 시작하는 제네릭 트랜잭션 개체에 대한 기본 메서드를 제공합니다. 이 인터페이스의 메서드를 호출하면 단일 트랜잭션에서 여러 COM+ 개체의 작업을 작성하고 트랜잭션을 명시적으로 커밋하거나 중단할 수 있습니다.
ITransactionProperty 트랜잭션 리소스 풀을 가져오는 데 사용됩니다.
ITransactionProxy COM+ 트랜잭션 컨텍스트가 DTC가 아닌 트랜잭션에서 작동하는 방법을 제공합니다.
ITransactionResourcePool 트랜잭션이 완료될 때까지 사용되는 풀된 개체의 목록을 유지 관리합니다. 이 개체는 IObjPool에서 키로 지정합니다.

ITransactionStatus CServiceConfig가 CoEnterServiceDomain 호출에서 트랜잭션을 사용하도록 구성된 경우 CoLeaveServiceDomain 호출로 완료된 트랜잭션의 상태 검색하는 데 사용됩니다.
Objectcontext 현재 개체의 컨텍스트에 대한 액세스를 제공합니다. 개체의 컨텍스트는 트랜잭션을 사용하거나 개체의 보안을 처리할 때 주로 사용됩니다. (ObjectContext)
ObjectControl 구성 요소에서 이 인터페이스를 구현하는 경우 COM+ 런타임 환경은 적절한 시간에 개체에서 해당 메서드를 자동으로 호출합니다.
SecurityProperty 현재 개체의 원래 호출자 및 직접 호출자에 대한 정보를 검색합니다.

구조체

ApplicationProcessRecycleInfo COM+ 애플리케이션을 호스팅하는 프로세스의 재활용에 대한 세부 정보를 나타냅니다.
ApplicationProcessStatistics COM+ 애플리케이션을 호스팅하는 프로세스에 대한 통계 정보를 나타냅니다.
ApplicationProcessSummary COM+ 애플리케이션을 호스팅하는 프로세스에 대한 요약 정보를 나타냅니다.
ApplicationSummary 특정 프로세스에서 호스트되는 COM+ 애플리케이션을 나타냅니다. 프로세스의 모든 SWC(서비스 구성 요소 없음) 컨텍스트에 대한 의사 애플리케이션 항목을 나타낼 수도 있습니다.
COMEVENTSYSCHANGEINFO 이벤트가 시작된 파티션 및 애플리케이션 ID를 포함하는 시스템 이벤트 구조를 나타냅니다.
ComponentHangMonitorInfo COM+ 구성 요소에 대한 중단 모니터링 구성을 나타냅니다.

ComponentStatistics 특정 프로세스에서 호스트되는 COM+ 구성 요소에 대한 통계 정보를 나타냅니다.
ComponentSummary 특정 프로세스에서 호스트되는 COM+ 구성 요소에 대한 요약 정보를 나타냅니다. SWC(구성 요소 없는 서비스) 컨텍스트를 나타낼 수도 있습니다.
COMSVCSEVENTINFO 이벤트가 생성된 시간 및 생성한 프로세스 서버 및 COM+ 애플리케이션과 같은 이벤트에 대한 컨텍스트 정보를 나타냅니다.
CrmLogRecordRead CRM(보상 Resource Manager)에 대한 구조화되지 않은 로그 레코드를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

COM(구성 요소 개체 모델)

아티클2023. 03. 11.

COM(구성 요소 개체 모델) 기술 개요입니다.

COM(구성 요소 개체 모델)을 개발하려면 다음 헤더가 필요합니다.

- accctrl.h
- callobj.h
- combaseapi.h
- comcat.h
- ctxtcall.h
- dmerror.h
- docobj.h
- eventsys.h
- guiddef.h
- iaccess.h
- imessagedispatcher.h
- messagedispatcherapi.h
- objbase.h
- objidl.h
- objidlbase.h
- ocidl.h
- ole.h
- ole2.h
- oledlg.h
- oleidl.h
- roapi.h
- rpcdce.h
- servprov.h
- shobjidl.h
- txlogpub.h
- unknwn.h
- unknwnbase.h
- urlmon.h
- vbinterf.h
- winddi.h
- winerror.h
- wtypesbase.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- COM(구성 요소 개체 모델)

열거형

ACTIVATEFLAGS 개체가 창 없는 개체로 활성화되는지 여부를 나타냅니다. IOleInPlaceSiteEx::OnInPlaceActivateEx에서 사용됩니다.
ADVF 데이터 변경 내용의 캐싱 및 알림을 제어하는 플래그입니다.
APTTYPE APTTYPE 열거형(objidl.h)은 다양한 유형의 아파트를 지정합니다.
APTTYPE APTTYPE(objidlbase.h) 열거형은 다양한 유형의 아파트를 지정합니다.
APTTYPEQUALIFIER APTTYPEQUALIFIER 열거형(objidl.h)은 가능한 COM 아파트 형식 한정자 집합을 지정합니다.
APTTYPEQUALIFIER APTTYPEQUALIFIER(objidlbase.h) 열거형은 가능한 COM 아파트 형식 한정자 집합을 지정합니다.
BIND_FLAGS 모니커 바인딩 작업의 여러 측면을 제어합니다.
BINDSPEED 호출자가 개체에 바인딩될 때까지 대기하는 시간을 대략적으로 나타냅니다.
CALLFRAME_COPY 부모 프레임에 대한 수명 종속성을 결정하여 복사된 호출 프레임 데이터를 부모 프레임의 데이터와 공유할 수 있는지 여부를 결정합니다.
CALLFRAME_FREE 해제할 매개 변수 형식을 결정합니다. (CALLFRAME_FREE)

CALLFRAME_NULL 해제할 매개 변수 형식을 결정합니다. (CALLFRAME_NULL)
CALLFRAME_WALK 연습할 매개 변수 형식을 결정합니다.
CALLTYPE IMessageFilter::HandleInComingCall에서 사용하는 호출 형식을 지정합니다.
CLSCTX 개체를 실행할 실행 컨텍스트를 나타내기 위해 활성화 호출에 사용되는 값입니다.
CO_MARSHALING_CONTEXT_ATTRIBUTES CO_MARSHALING_CONTEXT_ATTRIBUTES 열거형(objidl.h)은 특정 메서드를 사용하여 쿼리할 수 있는 마샬링 컨텍스트 특성 중 하나를 식별합니다.
CO_MARSHALING_CONTEXT_ATTRIBUTES CO_MARSHALING_CONTEXT_ATTRIBUTES(objidlbase.h) 열거형은 GetMarshalingContextAttribute 메서드를 사용하여 쿼리할 수 있는 특성 중 하나를 식별합니다.
COINIT 이 스레드에서 만든 개체에 대한 들어오는 호출에 사용되는 동시성 모델을 결정합니다. 이 동시성 모델은 아파트 스레드 또는 다중 스레드일 수 있습니다.
COMSD CoGetSystemSecurityPermissions를 호출할 때 가져올 COM 보안 설명자의 유형을 결정합니다.
COWAIT_FLAGS CoWaitForMultipleHandles 함수의 동작을 지정합니다.
CWMO_FLAGS CoWaitForMultipleObjects 함수에 대한 플래그를 제공합니다.
DATADIR 데이터 흐름의 방향을 지정합니다. 이 메서드는 결과 열거자가 열거할 수 있는 형식을 결정합니다.

DISCARDCACHE 더티 비트가 설정된 경우 메모리에서 삭제할 캐시로 수행할 작업을 지정합니다.
DOCMISC 문서 개체에 대한 기타 속성 정보를 제공합니다.
DVASPECT 데이터를 그리거나 가져올 때 개체의 원하는 데이터 또는 뷰 모양을 지정합니다.
DVASPECT2 그리기 프로세스를 최적화하는 데 사용되는 새 그리기 측면을 지정합니다.
DVASPECTINFOFLAG 개체가 자체의 최적화된 그리기를 지원할 수 있는지 여부를 나타냅니다.
DVEXTENTMODE 크기 조정 모드가 콘텐츠인지 정수 크기 조정인지 여부를 나타냅니다.
EOC_ChangeType 이벤트 개체에 대한 변경의 특성을 나타냅니다.
EOLE_AUTHENTICATION_CAPABILITIES EOLE_AUTHENTICATION_CAPABILITIES 열거형(objidl.h)은 CoInitializeSecurity 및 IClientSecurity::SetBlanket의 다양한 기능을 지정합니다.
EOLE_AUTHENTICATION_CAPABILITIES EOLE_AUTHENTICATION_CAPABILITIES(objidlbase.h) 열거형은 CoInitializeSecurity 및 IClientSecurity::SetBlanket의 다양한 기능을 지정합니다.
EXTCONN EXTCONN 열거형(objidl.h)은 포함된 개체에 존재하는 외부 연결의 형식을 지정합니다.
EXTCONN EXTCONN(objidlbase.h) 열거형은 포함된 개체에 존재하는 외부 연결의 형식을 지정합니다.

GLOBALOPT_PROPERTIES GLOBALOPT_PROPERTIES 열거형(objidl.h)은 IGlobalOptions 인터페이스를 사용하여 설정하거나 쿼리할 수 있는 프로세스 전역 옵션을 식별합니다.
GLOBALOPT_PROPERTIES GLOBALOPT_PROPERTIES(objidlbase.h) 열거형은 IGlobalOptions 인터페이스를 사용하여 설정하거나 쿼리할 수 있는 프로세스 전역 옵션을 식별합니다.
GLOBALOPT_UNMARSHALING_POLICY_VALUES GLOBALOPT_UNMARSHALING_POLICY_VALUES 열거형(objidl.h)은 COM unmarshaling 정책 전역 옵션에 대한 값을 제공합니다.
GLOBALOPT_UNMARSHALING_POLICY_VALUES GLOBALOPT_UNMARSHALING_POLICY_VALUES(objidlbase.h) 열거형은 COM 경계 해제 정책 전역 옵션에 대한 값을 제공합니다.
GUIDKIND IProvideClassInfo2의 개체에서 요청된 정보의 종류를 지정하는 데 사용되는 플래그입니다.
HITRESULT 위치가 개체 이미지 내에 있는지 여부를 나타냅니다.
MKRREDUCE 모니커를 줄일 정도를 지정합니다.
MKSYS 모니커의 클래스를 나타냅니다.
MSHCTX unmarshaling을 수행할 프로세스인 대상 컨텍스트를 지정합니다.
MSHLFLAGS 마샬링을 수행하는 이유를 지정합니다.
OLECLOSE 닫기 전에 개체를 저장해야 하는지 여부를 나타냅니다.

OLECMDEXECOPT 명령 실행 옵션을 지정합니다.
OLECMDF OLECMD 구조체에 지정된 명령에 대해 개체에서 제공하는 지원 유형을 지정합니다.
OLECMDID 실행할 표준 명령을 지정합니다. 이 열거형의 단일 값은 IOleCommandTarget::Exec의 nCmdID 인수에 전달됩니다.
OLECMDID_WINDOWSTATE_FLAG 창 상태를 지정합니다.
OLECMDTEXTF IOleCommandTarget::QueryStatus에 전달된 OLECMDTEXT 구조체에 개체가 저장해야 하는 정보의 형식을 지정합니다.
OLECONTF 열거할 개체의 형식을 나타냅니다.
OLEDCFLAGS 개체가 요청한 디바이스 컨텍스트에 대한 추가 정보를 컨테이너에 지정합니다.
OLEGETMONIKER IOleObject::GetMoniker 및 IOleClientSite::GetMoniker 메서드의 동작 측면을 제어합니다.
OLELINKBIND 링크 원본에 대한 바인딩 작업을 제어합니다.
OLEMISC 개체 또는 개체 클래스의 기타 특성을 설명합니다.
OLERENDER 새로 만든 개체에 대해 요청된 캐싱 유형을 나타냅니다.

OLEUIPASTEFLAG 이 형식을 붙여넣을 때 사용자가 사용할 수 있는 사용자 옵션과 선택 항목 그룹 또는 목록(붙여넣기, 링크 붙여넣기 등)을 나타냅니다.
OLEUPDATE 연결된 개체가 연결된 개체에 대해 캐시된 데이터를 자동으로 업데이트하는지 또는 컨테이너가 IOleObject::Update 또는 IOleLink::Update 메서드를 호출하는 경우에만 업데이트하는지 여부를 나타냅니다. 상수는 IOleLink 인터페이스에서 사용됩니다.
OLEVERBATTRIB 개체에 대해 지정된 동사의 특성을 설명합니다.
OLEWHICHMK 개체의 모니커가 설정되거나 검색되는 부분을 나타냅니다.
PENDINGMSG IMessageFilter::MessagePending 메서드의 반환 값을 지정합니다.
PENDINGTYPE IMessageFilter::MessagePending 메서드의 중첩 수준을 나타냅니다.
PICTUREATTRIBUTES IPicture::get_Attributes 메서드를 통해 반환되는 그림 개체의 특성을 지정합니다.
POINTERINACTIVE 개체의 활성화 정책을 나타내며 IPointerInactive::GetActivationPolicy 메서드에 사용됩니다.
QACONTAINERFLAGS 컨테이너에서 제공하는 앰비언트 속성을 나타냅니다. QACONTAINER 구조체의 dwAmbientFlags 멤버에서 사용됩니다.
RECORD_READING_POLICY 로그에서 레코드를 읽을 순서에 대한 힌트를 지정합니다.
REGCLS 클래스 개체에 대한 연결 유형을 제어합니다.

SERVERCALL 서버 호출의 상태를 나타냅니다.
THDTYPE THDTYPE 열거형(objidl.h)은 특정 스레드가 메시지 루프를 지원하는지 여부를 나타냅니다.
THDTYPE THDTYPE(objidlbase.h) 열거형은 특정 스레드가 메시지 루프를 지원하는지 여부를 나타냅니다.
TYMED 데이터 전송에 사용되는 스토리지 매체의 유형을 나타냅니다. STGMEDIUM 또는 FORMATETC 구조체에서 사용됩니다.
TYSPEC 클래스 ID에 대한 매핑을 지정합니다.
UASFLAGS 부모 실행 취소 단위에 대한 정보를 제공합니다.
USERCLASSTYPE 개체 클래스와 연결된 표시 이름의 다양한 변형을 나타냅니다.
VIEWSTATUS 개체의 불투명도와 개체에서 지원하는 그리기 측면을 지정합니다.

Functions

활성화 속성 페이지의 대화 상자 창을 만듭니다.
ActivateMe 문서 사이트에 현재 위치 활성 개체가 아닌 문서 개체로 호출하는 문서를 활성화하도록 요청하고 필요에 따라 활성화할 개체 문서의 보기를 지정합니다.

추가 컬렉션에 간단한 실행 취소 단위를 추가합니다.
추가 컬렉션에 간단한 실행 취소 단위를 추가합니다. 부모 실행 취소 단위가 열려 있는 동안 실행 취소 관리자는 IOleParentUndoUnit::Add를 호출하여 실행 취소 단위를 추가합니다.
AddConnection IExternalConnection::AddConnection 메서드(objidl.h)는 개체의 강력한 외부 연결 수를 증가합니다.
AddConnection IExternalConnection::AddConnection(objidlbase.h) 메서드는 개체의 강력한 외부 연결 수를 증가합니다.
AddRef COM 개체에 대한 인터페이스 포인터에 대한 참조 수를 증분합니다. 인터페이스 포인터의 복사본을 만들 때마다 이 메서드를 호출해야 합니다.
AddRefHfont ReleaseHfont가 호출되거나 글꼴 개체 자체가 완전히 해제될 때까지 hFont로 식별된 이전에 실현된 글꼴이 유효한 상태로 유지되어야 한다는 것을 글꼴 개체에 알립니다.
AddRefOnProcess 프로세스의 참조 수를 증분합니다.
AddSynchronize ISynchronizeContainer::AddSynchronize 메서드(objidl.h)는 컨테이너에 동기화 개체를 추가합니다.
AddSynchronize ISynchronizeContainer::AddSynchronize(objidlbase.h) 메서드는 컨테이너에 동기화 개체를 추가합니다.
AdjustRect 겹치는 불투명 개체로 완전히 또는 부분적으로 적용되는 경우 지정된 사각형을 조정합니다.
조언 연결 지점 개체와 클라이언트의 싱크 간에 연결을 설정합니다.

조언 OLE 개체와 호출 개체의 advise 싱크 간에 권고 연결을 설정합니다. 해당 싱크를 통해 OLE 개체의 이름을 바꾸거나 저장하거나 닫을 때 호출 개체가 알림을 받을 수 있습니다.
조언 복합 문서 개체와 호출 개체의 advise 싱크 간에 권고 연결을 설정합니다. 이를 통해 복합 문서 개체의 이름이 변경, 저장 또는 닫힐 때 호출 개체가 알림을 받습니다.
조언 알림 수신을 위해 advise 싱크와 데이터 개체 간에 연결을 만듭니다.
Alloc 메모리 블록을 할당합니다. (IMalloc.Alloc)
AllowForegroundTransfer COM 서버 프로세스에 대한 포그라운드 창을 생성합니다.
AppendRecord 로그 끝에 새 레코드를 씁니다.
ApplicationFree 애플리케이션을 해제합니다.
ApplicationLaunch 애플리케이션을 시작합니다.
적용 이전에 IPropertyPage::SetObjects에 전달된 속성 페이지와 연결된 기본 개체에 현재 값을 적용합니다.
ApplyViewState 이전에 IOleDocumentView::SaveViewState 호출에 저장된 보기 상태를 사용하여 보기를 초기화합니다.
BindIfRunning 링크 원본이 이미 실행 중인 경우 연결된 개체와 링크 원본 간의 연결을 활성화합니다.

BindMoniker 모니커를 통해 개체를 찾고, 비활성 상태이면 개체를 활성화하고, 해당 개체의 지정된 인터페이스에 대한 포인터를 검색합니다.
BindToObject 지정된 개체에 바인딩합니다. 바인딩 프로세스에는 개체를 찾고, 필요한 경우 실행 상태로 전환하고, 호출자에게 식별된 개체의 지정된 인터페이스에 대한 포인터를 제공하는 작업이 포함됩니다.
BindToSource 연결된 개체 내에 저장된 모니커를 바인딩하여 링크 원본에 대한 연결을 활성화합니다.
BindToStorage 지정된 개체의 스토리지에 바인딩합니다. IMoniker::BindToObject 메서드와 달리 이 메서드는 모니커로 식별된 개체를 활성화하지 않습니다.
Cache 포함된 개체 내에 캐시할 형식 및 기타 데이터를 지정합니다.
CallIndirect 호출 인수에 대한 간접 참조를 사용하여 인터페이스의 메서드 중 하나를 호출합니다.
취소 ICancelMethodCalls::Cancel 메서드(objidl.h)는 메서드 호출을 취소하도록 요청합니다.
취소 ICancelMethodCalls::Cancel(objidlbase.h) 메서드는 메서드 호출을 취소하도록 요청합니다.
CancelLink 선택한 링크의 연결을 끊습니다. (ANSI)
CancelLink 선택한 링크의 연결을 끊습니다. (유니코드)
CanInPlaceActivate 컨테이너가 개체를 활성화할 수 있는지 여부를 결정합니다.

CanWindowlessActivate 컨테이너가 현재 위치에서 활성화할 수 있는 창 없는 개체로 지원할 수 있는지 개체에 알립니다.
CatalogRefresh 카탈로그를 새로 고칩니다.
CLIPFORMAT_UserFree CLIPFORMAT_UserFree 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
CLIPFORMAT_UserFree64 CLIPFORMAT_UserFree64 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
CLIPFORMAT_UserMarshal CLIPFORMAT_UserMarshal 함수(oleidl.h)는 CLIPFORMAT 개체를 RPC 버퍼로 마샬링합니다.
CLIPFORMAT_UserMarshal64 CLIPFORMAT_UserMarshal64 함수(oleidl.h)는 CLIPFORMAT 개체를 RPC 버퍼로 마샬링합니다.
CLIPFORMAT_UserSize CLIPFORMAT_UserSize 함수(oleidl.h)는 CLIPFORMAT 개체의 와이어 크기를 계산하고 해당 핸들과 데이터를 가져옵니다.
CLIPFORMAT_UserSize64 CLIPFORMAT_UserSize64 함수(oleidl.h)는 CLIPFORMAT 개체의 와이어 크기를 계산하고 해당 핸들과 데이터를 가져옵니다.
CLIPFORMAT_UserUnmarshal CLIPFORMAT_UserUnmarshal 함수(oleidl.h)는 RPC 버퍼에서 CLIPFORMAT 개체를 숨깁니다.
CLIPFORMAT_UserUnmarshal64 CLIPFORMAT_UserUnmarshal64 함수(oleidl.h)는 RPC 버퍼에서 CLIPFORMAT 개체를 숨깁니다.
복제 현재 열거자와 열거 상태가 같은 새 열거자를 만듭니다. (IEnumConnectionPoints.Clone)

복제 현재 열거자와 열거 상태가 같은 새 열거자를 만듭니다. (IEnumConnections.Clone)
복제 현재 열거자와 열거 상태가 같은 새 열거자를 만듭니다. (IEnumOleUndoUnits.Clone)
복제 중복 글꼴 개체를 만듭니다.
복제 현재 열거자와 열거 상태가 같은 새 열거자를 만듭니다. (IEnumOleDocumentViews.Clone)
복제 현재 뷰와 동일한 내부 상태를 사용하여 중복 보기 개체를 만듭니다.
복제 현재 열거자와 열거 상태가 같은 새 열거자를 만듭니다. (IEnumOLEVERB.Clone)
복제 IEnumContextProps::Clone 메서드(objidl.h)는 현재 열거형 상태와 동일한 열거자를 포함하는 새 열거자를 만듭니다.
복제 현재 열거자와 열거 상태가 같은 새 열거자를 만듭니다. (IEnumFORMATETC.Clone)
복제 현재 열거자와 열거 상태가 같은 새 열거자를 만듭니다. (IEnumMoniker.Clone)
복제 현재 열거자와 열거 상태가 같은 새 열거자를 만듭니다. (IEnumSTATDATA.Clone)
복제 IEnumString::Clone 메서드(objidl.h)는 현재 열거형 상태와 동일한 열거자를 포함하는 새 열거자를 만듭니다.

복제 IEnumUnknown::Clone 메서드(objidl.h)는 현재 열거형 상태와 동일한 열거자를 포함하는 새 열거자를 만듭니다.
복제 IEnumContextProps::Clone(objidlbase.h) 메서드는 현재 열거형 상태와 동일한 열거자를 포함하는 새 열거자를 만듭니다.
복제 IEnumString::Clone(objidlbase.h) 메서드는 현재 열거형 상태와 동일한 열거자를 포함하는 새 열거자를 만듭니다.
복제 IEnumUnknown::Clone(objidlbase.h) 메서드는 현재 열거형 상태와 동일한 열거자를 포함하는 새 열거자를 만듭니다.
복제 현재 열거자와 열거 상태가 같은 새 열거자를 만듭니다. (IEnumCATEGORYINFO.Clone)
복제 현재 열거자와 열거 상태가 같은 새 열거자를 만듭니다. (IEnumGUID.Clone)
닫기 지정된 부모 실행 취소 단위를 닫습니다. (IOleParentUndoUnit.Close)
닫기 지정된 부모 실행 취소 단위를 닫습니다. (IOleUndoManager.Close)
닫기 포함된 개체를 실행 중에서 로드된 상태로 변경합니다. 링크 원본에서 연결된 개체의 연결을 끊습니다.
CloseView 문서 보기에 자체적으로 닫고 IOleInPlaceSite 포인터를 해제하도록 지시합니다.
CLSIDFromProgID ProgID가 지정된 경우 레지스트리에서 CLSID를 조회합니다.

CLSIDFromProgIDEx COMClassStore 정책을 사용하는 경우 CLSIDFromProgIDEx 함수(objbase.h)가 자동 설치를 트리거합니다.
CLSIDFromProgIDEx COMClassStore 정책을 사용하도록 설정하면 CLSIDFromProgIDEx 함수(combaseapi.h)가 자동 설치를 트리거합니다.
CLSIDFromString StringFromCLSID 함수에서 생성된 문자열을 원래 CLSID로 다시 변환합니다.
CoAddRefServerProcess 프로세스별 전역 참조 수를 증분합니다.
CoAllowSetForegroundWindow 이 함수는 전경 권한(포그라운드 창을 설정하는 권한)을 한 프로세스에서 다른 프로세스로 전달합니다. 포그라운드 권한이 있는 프로세스는 이 함수를 호출하여 해당 권한을 로컬 COM 서버 프로세스에 전달할 수 있습니다.
CoAllowUnmarshalerCLSID 호출 프로세스에 대해서만 허용된 목록에 unmarshaler CLSID를 추가합니다.
CoCancelCall 지정된 스레드에서 보류 중인 아웃바운드 DCOM 메서드 호출의 취소를 요청합니다.
CoCopyProxy 지정된 프록시의 프라이빗 복사본을 만듭니다.
CoCreateFreeThreadedMarshaler 컨텍스트 종속 마샬링이 가능한 집계 가능한 개체를 만듭니다.
CoCreateGuid CLSID 및 인터페이스 식별자에 사용되는 고유한 128비트 정수인 GUID를 만듭니다.
CoCreateInstance 지정된 CLSID와 연결된 클래스의 초기화되지 않은 단일 개체를 만듭니다.

CoCreateInstanceEx 특정 컴퓨터에 특정 클래스의 인스턴스를 만듭니다.
CoCreateInstanceFromApp 앱 컨테이너 내에서 특정 컴퓨터에 특정 클래스의 인스턴스를 만듭니다.
CoDecrementMTAUsage CoIncrementMTAUsage 함수에 대한 이전 호출에서 수행한 증분을 해제합니다.
CoDisableCallCancellation CoEnableCallCancellation에 대한 호출 작업을 실행 취소합니다. CoEnableCallCancellation에 대한 모든 호출이 CoDisableCallCancellation 호출에 의해 균형을 이루면 호출 스레드에서 동기 호출의 취소를 사용하지 않도록 설정합니다.
CoDisconnectContext 현재 컨텍스트의 개체를 가리키는 모든 인터페이스 포인터를 대신하여 유지 관리되는 모든 프록시 연결의 연결을 끊습니다.
CoDisconnectObject 지정된 개체를 가리키는 모든 인터페이스 포인터를 대신하여 유지 관리되는 모든 원격 프로세스 연결을 끊습니다.
CoDosDateTimeToFileTime 시간과 날짜의 MS-DOS 표현을 Windows에서 사용하는 FILETIME 구조로 변환합니다.
CoEnableCallCancellation 호출 스레드에서 동기 호출을 취소할 수 있습니다.
CoFileTimeNow CoFileTimeNow 함수(objbase.h)는 현재 시간을 FILETIME 구조체로 반환합니다.
CoFileTimeNow CoFileTimeNow 함수(combaseapi.h)는 현재 시간을 FILETIME 구조체로 반환합니다.
CoFileTimeToDosDateTime FILETIME을 MS-DOS 날짜 및 시간 값으로 변환합니다.

CoFreeAllLibraries 현재 사용 중인지 여부에 관계없이 CoLoadLibrary 함수(CoGetClassObject에서 내부적으로 호출)로 로드된 모든 DLL을 해제합니다.
CoFreeLibrary 로드될 때 명시적으로 해제되도록 지정된 라이브러리를 해제합니다.
CoFreeUnusedLibraries DLL에 더 이상 인스턴스화되지 않은 COM 개체가 없기 때문에 더 이상 사용되지 않는 DLL을 언로드합니다.
CoFreeUnusedLibrariesEx 더 이상 사용되지 않고 언로드 지연이 만료된 DLL을 언로드합니다.
CoGetApartmentType 현재 아파트 형식 및 형식 한정자를 반환합니다.
CoGetCallContext 현재 스레드에서 현재 호출의 컨텍스트를 검색합니다.
CoGetCallerTID 호출자 스레드의 아파트 ID를 포함하는 DWORD에 대한 포인터를 반환합니다.
CoGetCancelObject 동일한 또는 다른 클라이언트 스레드에서 보류 중인 아웃바운드 COM 메서드 호출에 해당하는 cancel 개체에서 호출 제어 인터페이스(일반적으로 ICancelMethodCalls)에 대한 포인터를 가져옵니다.
CoGetClassObject 지정된 CLSID와 연결된 클래스 개체의 인터페이스에 대한 포인터를 제공합니다.
CoGetContextToken 현재 컨텍스트에 대한 IObjContext 구현에 대한 포인터를 반환합니다.
CoGetCurrentLogicalThreadId 현재 실제 스레드의 논리적 스레드 식별자를 반환합니다.

CoGetCurrentProcess 현재 스레드에 고유한 값을 반환합니다. CoGetCurrentProcess를 사용하여 스레드 ID 재사용 문제를 방지할 수 있습니다.
CoGetInstanceFromFile 새 개체를 만들고 IPersistFile::Load를 사용하여 파일에서 초기화합니다.
CoGetInstanceFromIStorage 새 개체를 만들고 IPersistFile::Load에 대한 내부 호출을 통해 스토리지 개체에서 초기화합니다.
CoGetInterceptor 지정된 인터페이스를 가로챌 적절한 인터셉터를 인스턴스화하고 새로 만든 인터셉터를 반환합니다.
CoGetInterfaceAndReleaseStream 인터페이스 포인터가 포함된 버퍼를 언마샬링하고 인터페이스 포인터가 다른 스레드에서 호출 스레드로 마샬링되면 스트림을 해제합니다.
CoGetMalloc 애플리케이션이 메서드를 호출하여 메모리를 관리할 수 있도록 기본 OLE 작업 메모리 할당자(IMalloc 인터페이스의 시스템 구현을 지원)에 대한 포인터를 검색합니다.
CoGetMarshalSizeMax 지정된 인터페이스 포인터를 지정된 개체에 마샬링하는 데 필요한 바이트 수의 상한을 반환합니다.
CoGetObject 표시 이름을 라는 개체를 식별하는 모니커로 변환한 다음 모니커로 식별된 개체에 바인딩합니다.
CoGetObjectContext 현재 개체의 컨텍스트를 반환합니다.
CoGetPSClsid 지정된 인터페이스에 대한 프록시 및 스텁을 구현하는 DLL의 CLSID를 반환합니다.
CoGetStandardMarshal 호출자에 따라 클라이언트 프로세스 또는 서버 프로세스에서 기본 또는 표준 마샬링 개체를 만들고 해당 개체의 IMarshal 구현에 대한 포인터를 반환합니다.

CoGetStdMarshalEx 경량 클라이언트 쪽 처리기에 사용할 집계된 표준 마샬러를 만듭니다.
CoGetSystemSecurityPermissions 머신 전체 시작 및 액세스 권한의 보안 설명자의 기본값과 시작 및 액세스 제한을 반환합니다.
CoGetTreatAsClass 지정된 개체를 에뮬레이트할 수 있는 개체의 CLSID를 반환합니다.
CoHandlePriorityEventsFromMessagePump WINDOWS 메시지를 디스패치한 후 ASTA 스레드에서 메시지 디스패처가 호출하여 짧은 실행 인프라 COM 호출 및 메시지 간에 디스패치할 기타 높은 우선 순위 또는 짧은 실행 COM 작업을 위한 기회를 제공합니다.
CoImpersonateClient 서버가 호출 기간 동안 현재 호출의 클라이언트를 가장할 수 있도록 합니다.
CoIncrementMTAUsage MTA 스레드가 실행되고 있지 않은 경우 MTA 지원을 활성으로 유지합니다.
CoInitialize 현재 스레드에서 COM 라이브러리를 초기화하고 동시성 모델을 STA(단일 스레드 아파트)로 식별합니다.
CoInitializeEx 호출 스레드에서 사용할 COM 라이브러리를 초기화하고, 스레드의 동시성 모델을 설정하고, 필요한 경우 스레드에 대한 새 아파트를 만듭니다.
CoInitializeSecurity 보안을 등록하고 프로세스의 기본 보안 값을 설정합니다.
CoInvalidateRemoteMachineBindings 서비스 제어 관리자에게 지정된 컴퓨터에 대해 캐시된 RPC 바인딩 핸들을 플러시하도록 지시합니다.
CoIsHandlerConnected 원격 개체가 해당 in-process 개체에 연결되어 있는지 여부를 확인합니다.

CoIsOle1Class 지정된 CLSID가 OLE 1 개체를 나타내는지 여부를 결정합니다.
CoLoadLibrary 호출자의 프로세스에 특정 DLL을 로드합니다.
CoLockObjectExternal 개체가 메모리에 유지되도록 개체를 잠그거나 이러한 잠금을 해제하기 위해 호출됩니다.
CoMarshalHresult COUnmarshalHresult 함수를 사용하여 HRESULT를 지정된 스트림으로 마샬링합니다.
CoMarshalInterface 일부 클라이언트 프로세스에서 프록시 개체를 초기화하는 데 필요한 데이터를 스트림에 씁니다.
CoMarshalInterThreadInterfaceInStream 동일한 프로세스에서 한 스레드에서 다른 스레드로 인터페이스 포인터를 마샬링합니다.
CommonPrefixWith 이 모니커가 지정된 모니커와 공통적으로 갖는 접두사를 기반으로 새 모니커를 만듭니다.
ComposeWith 현재 모니커와 지정된 모니커를 결합하여 새 복합 모니커를 만듭니다.
연결 IRpcProxyBuffer::Connect 메서드(objidl.h)는 클라이언트 프록시를 초기화하여 지정된 RPC 채널에 바인딩합니다.
연결 IRpcStubBuffer::Connect 메서드(objidl.h)는 서버 스텁을 초기화하여 지정된 인터페이스에 바인딩합니다.
연결 IRpcProxyBuffer::Connect(objidlbase.h) 메서드는 클라이언트 프록시를 초기화하여 지정된 RPC 채널에 바인딩합니다.

연결 IRpcStubBuffer::Connect(objidlbase.h) 메서드는 서버 스텁을 초기화하여 지정된 인터페이스에 바인딩합니다.
ContextCallback 개체 컨텍스트를 입력하고, 지정된 함수를 실행하고, 를 반환합니다.
ContextSensitiveHelp 현재 위치 활성화 세션 중에 상황에 맞는 도움말 모드를 입력해야 하는지 여부를 결정합니다.
ConvertObject 개체를 지정된 CLSID의 형식으로 변환합니다. (ANSI)
ConvertObject 개체를 지정된 CLSID의 형식으로 변환합니다. (유니코드)
복사 이 호출 프레임 및 연결된 모든 데이터의 복사본을 만듭니다.
CopyProxy 지정된 인터페이스에 대한 프록시의 프라이빗 복사본을 만듭니다.
CoQueryAuthenticationServices CoInitializeSecurity라는 프로세스가 있을 때 등록된 인증 서비스 목록을 검색합니다.
CoQueryClientBlanket 서버에서 호출하여 현재 스레드에서 실행되는 메서드를 호출한 클라이언트에 대해 알아봅니다.
CoQueryProxyBlanket 클라이언트가 지정된 프록시에서 호출하는 데 사용하는 인증 정보를 검색합니다.
CoRegisterActivationFilter 프로세스 전체 필터를 등록하여 활성화 요청을 처리합니다.
CoRegisterChannelHook 채널 후크를 등록합니다.

CoRegisterClassObject EXE 클래스 개체를 OLE에 등록하여 다른 애플리케이션이 연결할 수 있도록 합니다.
CoRegisterDeviceCatalog 마샬링 코드가 해당 인터페이스를 마샬링할 수 있도록 다운로드한 DLL이 실행 중인 프로세스 내에 디바이스 카탈로그 인터페이스를 등록할 수 있도록 합니다.
CoRegisterInitializeSpy IInitializeSpy 인터페이스의 구현을 등록합니다. 개발자가 COM 아파트에서 초기화 및 정리를 수행할 수 있도록 IInitializeSpy 인터페이스는 무시됩니다.
CoRegisterMallocSpy IMallocSpy 인터페이스의 구현을 등록한 후 OLE가 해당 IMalloc 메서드를 호출할 때마다 래퍼 메서드를 호출하도록 요구합니다.
CoRegisterMessageFilter 현재 스레드에서 동시성 문제를 처리하는 데 사용되는 IMessageFilter 인터페이스의 인스턴스를 OLE에 등록합니다.
CoRegisterPSClsid 마샬링 코드가 해당 인터페이스를 마샬링할 수 있도록 다운로드한 DLL이 실행 중인 프로세스 내에 사용자 지정 인터페이스를 등록할 수 있도록 합니다.
CoRegisterSurrogate 해당 ISurrogate 인터페이스 포인터를 통해 서로게이트 프로세스를 등록합니다.
CoReleaseMarshalData 이전에 마샬링된 데이터 패킷을 제거합니다.
CoReleaseServerProcess 전역 프로세스별 참조 수를 감소합니다.
CoResumeClassObjects 등록된 모든 클래스에 대해 SCM에 알리기 위해 여러 클래스 개체를 등록하고 해당 클래스 개체에 대한 활성화 요청을 허용하는 서버에서 호출됩니다.
CoRevertToSelf 실행 스레드에 대한 인증 정보를 복원합니다.

CoRevokeClassObject 이전에 CoRegisterClassObject 함수에 등록된 클래스 개체를 더 이상 사용할 수 없음을 OLE에 알릴 수 있습니다.
CoRevokeDeviceCatalog CoRegisterDeviceCatalog에 대한 이전 호출에서 수행한 디바이스 카탈로그 등록을 취소합니다.
CoRevokeInitializeSpy IInitializeSpy 인터페이스의 등록된 구현을 취소합니다.
CoRevokeMallocSpy 등록된 IMallocSpy 개체를 취소합니다.
CoSetCancelObject 현재 스레드에서 후속 취소 작업 중에 사용할 취소 개체를 설정(등록) 또는 다시 설정(등록 취소)합니다.
CoSetMessageDispatcher ASTA 스레드의 COM 대기 API 내에서 디스패치할 수 있는 창 메시지가 있을 때 호출할 스레드별 메시지 디스패처를 등록하거나 등록 취소합니다.
CoSetProxyBlanket 지정된 프록시에서 호출하는 데 사용할 인증 정보를 설정합니다.
CoSuspendClassObjects 프로세스 내에 등록된 모든 클래스 개체에서 SCM의 새 활성화 요청을 방지합니다.
CoSwitchCallContext CoGetCallContext에서 사용하는 호출 컨텍스트 개체를 전환합니다.
CoTaskMemAlloc IMalloc::Alloc와 동일한 방식으로 작업 메모리 블록을 할당합니다.
CoTaskMemFree CoTaskMemAlloc 또는 CoTaskMemRealloc 함수에 대한 호출을 통해 이전에 할당된 작업 메모리 블록을 해제합니다.

CoTaskMemRealloc 이전에 할당된 작업 메모리 블록의 크기를 변경합니다.
CoTestCancel 서버에서 실행 중인 호출이 클라이언트에 의해 취소되었는지 여부를 확인합니다.
CoTreatAsClass 한 클래스의 개체가 다른 클래스의 개체로 처리되는 에뮬레이션을 설정하거나 제거합니다.
CoUninitialize 현재 스레드에서 COM 라이브러리를 닫고, 스레드에 의해 로드된 모든 DLL을 언로드하고, 스레드가 유지 관리하는 다른 모든 리소스를 해제하고, 스레드의 모든 RPC 연결을 강제로 닫습니다.
CoUnmarshalHresult 지정된 스트림에서 HRESULT 형식을 언마샬링합니다.
CoUnmarshalInterface CoMarshalInterface 함수에 대한 이전 호출에 의해 스트림에 기록된 데이터를 사용하여 새로 만든 프록시를 초기화하고 해당 프록시에 대한 인터페이스 포인터를 반환합니다.
개수 IEnumContextProps::Count 메서드(objidl.h)는 컨텍스트의 컨텍스트 속성 수를 검색합니다.
개수 IEnumContextProps::Count(objidlbase.h) 메서드는 컨텍스트의 컨텍스트 속성 수를 검색합니다.
CountRefs IRpcStubBuffer::CountRefs 메서드(objidl.h)는 스텁이 연결된 서버 개체에 있는 총 참조 수를 검색합니다.
CountRefs IRpcStubBuffer::CountRefs(objidlbase.h) 메서드는 스텁이 연결된 서버 개체에 있는 총 참조 수를 검색합니다.
CoWaitForMultipleHandles 지정된 핸들이 신호를 받을 때까지 또는 지정된 시간 제한 기간이 경과될 때까지 기다립니다.

CoWaitForMultipleObjects CoWaitForMultipleHandles를 대체합니다. 이 대체 API는 ASTA에서 지원되지 않는 CoWaitForMultipleHandles에 대한 옵션을 숨깁니다.
CreateAntiMoniker 새 안티 모니커를 만들고 반환합니다.
CreateAsyncBindCtx 비동기 모니커에 사용할 비동기 바인딩 컨텍스트를 만듭니다.
CreateBindCtx IBindCtx(바인딩 컨텍스트 개체)의 구현에 대한 포인터를 반환합니다. 이 개체는 특정 모니커 바인딩 작업에 대한 정보를 저장합니다.
CreateCall ICallFactory::CreateCall 메서드(objidl.h)는 지정된 비동기 인터페이스에 해당하는 호출 개체의 인스턴스를 만듭니다.
CreateCall ICallFactory::CreateCall(objidlbase.h) 메서드는 지정된 비동기 인터페이스에 해당하는 호출 개체의 인스턴스를 만듭니다.
CreateClassMoniker 지정된 클래스를 참조하는 클래스 모니커를 만듭니다.
CreateDataAdviseHolder CreateDataAdviseHolder 함수(ole2.h)는 데이터 조언 소유자 개체에서 IDataAdviseHolder의 OLE 구현에 대한 포인터를 검색합니다.
CreateDataAdviseHolder CreateDataAdviseHolder 함수(objbase.h)는 데이터 조언 소유자 개체에서 IDataAdviseHolder의 OLE 구현에 대한 포인터를 검색합니다.
CreateDataCache 데이터 캐시의 OLE 제공 구현의 새 인스턴스에 대한 포인터를 검색합니다.
CreateFileMoniker 지정된 경로를 기반으로 파일 모니커를 만듭니다.

CreateFormatEnumerator FORMATETC 구조체의 정적 배열을 통해 IEnumFORMATETC를 구현하는 개체를 만듭니다.
CreateGenericComposite 두 모니커의 제네릭 컴퍼지션을 수행하고 결과 복합 모니커에 대한 포인터를 제공합니다.
CreateInstance 초기화되지 않은 개체를 만듭니다.
CreateInstanceLic 지정된 라이선스 키에 대해 라이선스가 부여된 개체의 인스턴스를 만듭니다. 이 메서드는 사용 허가되지 않은 컴퓨터에서 개체를 만들 수 있는 유일한 방법입니다.
CreateItemMoniker 포함하는 개체(일반적으로 복합 문서) 내에서 개체를 식별하는 항목 모니커를 만듭니다.
CreateObjrefMoniker 개체에 대한 포인터를 기반으로 OBJREF 모니커를 만듭니다.
CreateOleAdviseHolder 복합 문서 알림을 관리하기 위한 조언자 개체를 만듭니다. IOleAdviseHolder 인터페이스의 개체 OLE 구현에 대한 포인터를 반환합니다.
CreatePointerMoniker 개체에 대한 포인터를 기반으로 포인터 모니커를 만듭니다.
CreateProxy IPSFactoryBuffer::CreateProxy 메서드는 지정된 원격 인터페이스에 대한 프록시를 만듭니다.
CreateProxy IPSFactoryBuffer::CreateProxy(objidlbase.h) 메서드는 지정된 원격 인터페이스에 대한 프록시를 만듭니다.
CreateStub IPSFactoryBuffer::CreateStub 메서드(objidl.h)는 지정된 인터페이스의 원격 사용을 위한 스텁을 만듭니다.

CreateStub IPSFactoryBuffer::CreateStub(objidlbase.h) 메서드는 지정된 인터페이스의 원격 사용을 위한 스텁을 만듭니다.
CreateView 호출자 프로세스에서 문서 뷰 개체를 만들고 해당 개체의 IOleDocumentView 인터페이스에 대한 포인터를 가져옵니다.
DAdvise 데이터 개체와 advise 싱크 간의 연결을 만들기 위해 advise 싱크를 지원하는 개체에 의해 호출됩니다. 이렇게 하면 advise 싱크에 개체의 데이터 변경 내용에 대한 알림을 받을 수 있습니다.
비활성화 IPropertyPage::Activate에서 만든 창을 삭제합니다.
DeactivateAndUndo 개체를 비활성화하고, 현재 위치 세션을 종료하고, 컨테이너의 저장된 실행 취소 상태로 되돌려 놓습니다.
DebugServerQueryInterface IRpcStubBuffer::D ebugServerQueryInterface 메서드(objidl.h)는 스텁이 나타내는 인터페이스에 대한 포인터를 검색합니다.
DebugServerQueryInterface IRpcStubBuffer::D ebugServerQueryInterface(objidlbase.h) 메서드는 스텁이 나타내는 인터페이스에 대한 포인터를 검색합니다.
DebugServerRelease IRpcStubBuffer::D ebugServerRelease 메서드(objidl.h)는 이전에 DebugServerQueryInterface에서 반환된 인터페이스 포인터를 해제합니다.
DebugServerRelease IRpcStubBuffer::D ebugServerRelease(objidlbase.h) 메서드는 이전에 DebugServerQueryInterface에서 반환된 인터페이스 포인터를 해제합니다.
DidAlloc IMalloc::D idAlloc 메서드(objidl.h)는 이 할당자가 지정된 메모리 블록을 할당하는 데 사용되었는지 여부를 결정합니다.

DidAlloc IMalloc::D idAlloc(objidlbase.h) 메서드는 이 할당자가 지정된 메모리 블록을 할당하는 데 사용되었는지 여부를 결정합니다.
DiscardCache 메모리에 있는 캐시를 삭제합니다.
DiscardFrom 실행 취소 관리자에게 실행 취소 또는 다시 실행 스택에서 지정한 실행 취소 단위와 그 아래의 모든 실행 취소 단위를 취소하도록 지시합니다.
DiscardUndoState 컨테이너에 실행 취소 상태를 취소하도록 지시합니다. 컨테이너는 IOleInPlaceObject::ReActivateAndUndo를 호출하면 안 됩니다.
연결 끊기 IRpcProxyBuffer::D isconnect 메서드(objidl.h)는 클라이언트 프록시가 연결된 모든 RPC 채널에서 클라이언트 프록시의 연결을 끊습니다.
연결 끊기 IRpcStubBuffer::D isconnect 메서드(objidl.h)는 서버 스텁이 연결된 모든 인터페이스에서 서버 스텁의 연결을 끊습니다.
연결 끊기 IRpcProxyBuffer::D isconnect(objidlbase.h) 메서드는 클라이언트 프록시가 연결된 모든 RPC 채널에서 클라이언트 프록시의 연결을 끊습니다.
연결 끊기 IRpcStubBuffer::D isconnect(objidlbase.h) 메서드는 서버 스텁이 연결된 모든 인터페이스에서 서버 스텁의 연결을 끊습니다.
DisconnectObject IMarshal::D isconnectObject 메서드(objidl.h)는 종료하기 전에 개체에 대한 모든 연결을 해제합니다.
DisconnectObject IMarshal::D isconnectObject(objidlbase.h) 메서드는 서버에서 이 메서드의 개체 구현을 호출하는 개체에 대한 모든 연결을 해제합니다.

DllCanUnloadNow 이 함수를 구현하는 DLL이 사용 중인지 여부를 확인합니다. 그렇지 않은 경우 호출자는 메모리에서 DLL을 언로드할 수 있습니다.
DllGetClassObject DLL 개체 처리기 또는 개체 애플리케이션에서 클래스 개체를 검색합니다.
DllRegisterServer 이 서버 모듈에서 지원되는 모든 클래스에 대한 레지스트리 항목을 만들도록 In-Process 서버에 지시합니다.
DllUnregisterServer In Process 서버에 DllRegisterServer를 통해 만든 항목만 제거하도록 지시합니다.
해야 할 일 실행 취소 단위에 해당 작업을 수행하도록 지시합니다. 자식 실행 취소 단위가 포함된 경우 Do 메서드도 호출해야 합니다.
DockWindow 지정된 창 핸들을 지정된 모니터 핸들에 도킹합니다.
Dodragdrop OLE 끌어서 놓기 작업을 수행합니다.
Doverb 개체가 최종 사용자의 작업에 대한 응답으로 작업을 수행하도록 요청합니다. 가능한 작업은 IOleObject::EnumVerbs의 개체에 대해 열거됩니다.
DragEnter 낙하를 수락할 수 있는지 여부와 드롭의 효과를 나타냅니다.
DragEnterTarget OLE는 사용자가 마우스 커서를 잠재적인 놓기 대상 창으로 끌 때 이 메서드를 호출합니다.
DragLeave 대상 피드백을 제거하고 데이터 개체를 해제합니다.

DragLeaveTarget OLE는 사용자가 잠재적인 놓기 대상 창에서 마우스 커서를 끌 때 이 메서드를 호출합니다.
DragOver 사용자에게 대상 피드백을 제공하고 드롭 효과를 DoDragDrop 함수에 전달하여 드롭백의 효과를 원본으로 전달할 수 있습니다.
그리기 개체의 표현을 지정된 디바이스 컨텍스트에 그립니다.
Drop 원본 데이터를 대상 창에 통합하고, 대상 피드백을 제거하고, 데이터 개체를 해제합니다.
DUnadvise 이전에 설정된 알림 연결을 삭제합니다.
EditProperty 속성 페이지가 활성화될 때 포커스를 받을 필드를 지정합니다.
사용 실행 취소 관리자를 사용하거나 사용하지 않도록 설정합니다.
EnableModeless 컨테이너가 모달 대화 상자를 만들거나 삭제할 때 모덜리스 대화 상자를 사용하거나 사용하지 않도록 설정합니다.
EnableModeless 프레임의 모덜리스 대화 상자를 사용하거나 사용하지 않도록 설정합니다.
Enum 복합 모니커의 구성 요소에 대한 열거자에 대한 포인터를 검색합니다.
EnumAdvise 개체에 대해 현재 설정된 권고 연결을 열거하는 데 사용할 수 있는 열거자를 만듭니다.

EnumAdvise 개체에 등록된 권고 연결을 열거하는 데 사용할 수 있는 열거자에 대한 포인터를 검색하므로 컨테이너는 종료하기 전에 해제할 항목을 알 수 있습니다.
EnumAdvise 현재 권고 연결을 열거하는 데 사용할 수 있는 개체를 반환합니다.
EnumCache 현재 캐시 연결을 열거하는 데 사용할 수 있는 열거자를 만듭니다.
EnumCategories 시스템에 등록된 구성 요소 범주에 대한 열거자를 검색합니다.
EnumClassesOfCategories 하나 이상의 지정된 범주 식별자를 구현하는 클래스의 열거자를 검색합니다.
EnumConnectionPoints 연결 가능한 개체에서 지원되는 모든 연결 지점을 반복하는 열거자 개체를 만듭니다(나가는 IID당 하나의 연결점).
EnumConnections 이 연결점에 대한 현재 연결을 반복할 열거자 개체를 만듭니다.
EnumContextProps IContext::EnumContextProps 메서드(objidl.h)는 이 컨텍스트의 컨텍스트 속성을 열거할 수 있는 IEnumContextProps 인터페이스 포인터를 반환합니다.
EnumContextProps IContext::EnumContextProps(objidlbase.h) 메서드는 컨텍스트 속성을 열거하는 데 사용할 수 있는 IEnumContextProps 인터페이스 포인터를 반환합니다.
EnumControls 폼의 컨트롤을 열거합니다.
EnumDAdvise 현재 advise 연결을 열거하는 데 사용할 수 있는 개체를 만듭니다.

EnumFormatEtc 데이터 개체에서 지원하는 형식을 열거하는 개체를 만듭니다.
EnumImplCategoriesOfClass 지정된 클래스에 의해 구현된 CATID의 열거자를 검색합니다.
EnumObjectParam 바인딩 컨텍스트의 문자열 키 테이블 포인터의 키를 열거하는 데 사용할 수 있는 인터페이스에 대한 포인터를 검색합니다.
EnumObjects 현재 컨테이너의 개체를 열거합니다.
EnumRedoable 호출자가 다시 실행 스택에서 일련의 최상위 실행 취소 단위를 반복하는 데 사용할 수 있는 열거자 개체를 만듭니다.
EnumReqCategoriesOfClass 지정된 클래스에 필요한 CATID의 열거자를 검색합니다.
EnumRunning 현재 ROT(실행 중인 개체 테이블)에 등록된 모든 개체의 모니커를 나열할 수 있는 열거자에 대한 포인터를 만들고 반환합니다.
EnumUndoable 호출자가 실행 취소 스택에서 일련의 최상위 실행 취소 단위를 반복하는 데 사용할 수 있는 열거자 개체를 만듭니다.
EnumVerbs 개체에 사용할 수 있는 동사를 동사 번호별로 오름차순으로 나열하는 풀다운 메뉴를 노출합니다.
EnumViews 문서 개체에서 지원하는 뷰를 열거하는 개체를 만들거나 하나의 보기만 지원되는 경우 해당 뷰에 대한 포인터를 반환합니다.
Exec 지정된 명령을 실행하거나 명령에 대한 도움말을 표시합니다.

실패 상태 값에 대한 오류에 대한 일반 테스트를 제공합니다.
FContinue 제네릭 작업을 계속할지 여부를 나타냅니다.
FContinuePrinting 긴 인쇄 작업을 계속할지 여부를 나타냅니다.
FindConnectionPoint IID가 지원되는 나가는 인터페이스를 설명하는 경우 지정된 IID에 대한 연결점의 IConnectionPoint 인터페이스에 대한 포인터를 반환합니다.
FindUnit 지정된 단위가 이 실행 취소 단위의 자식인지 또는 해당 자식 중 하나인지, 즉 지정된 단위가 이 부모 단위의 계층 구조에 속하는지 여부를 나타냅니다.
Force 적어도 지정된 LSN을 통해 로그의 내용을 디스크에 강제로 적용합니다.
Format 패턴에 따라 문자열의 서식을 지정합니다.
Free IMalloc::Free 메서드(objidl.h)는 이전에 할당된 메모리 블록을 해제합니다.
Free IMalloc::Free(objidlbase.h) 메서드는 이전에 할당된 메모리 블록을 해제합니다.
Free 메모리 누수 방지를 위해 프레임 복사본을 해제합니다.
FreeBuffer IRpcChannelBuffer::FreeBuffer 메서드(objidl.h)는 이전에 할당된 RPC 채널 버퍼를 해제합니다.
FreeBuffer IRpcChannelBuffer::FreeBuffer(objidlbase.h) 메서드는 이전에 할당된 RPC 채널 버퍼를 해제합니다.

FreeParam 프레임에서 지정된 매개 변수를 해제합니다.
FreeSurrogate ISurrogate::FreeSurrogate 메서드(objidl.h)는 DLL 서버를 언로드합니다.
FreeSurrogate ISurrogate::FreeSurrogate(objidlbase.h) 메서드는 DLL 서버를 언로드합니다.
중지 IViewObject::Unfreeze 메서드가 호출될 때까지 개체가 변경되지 않도록 개체의 그리기 표현을 고정합니다. 이 메서드의 가장 일반적인 용도는 묶은 인쇄입니다.
FreezeEvents 컨테이너가 컨트롤의 이벤트를 무시하거나 수락하는지 여부를 나타냅니다.
get_Attributes 그림의 비트 특성의 현재 집합을 검색합니다.
get_Bold 글꼴의 현재 Bold 속성을 가져옵니다.
get_Charset 글꼴에 사용되는 문자 집합을 검색합니다.
get_CurDC 현재 디바이스 컨텍스트의 핸들을 검색합니다. 이 속성은 비트맵 그림에만 유효합니다.
get_Description 이벤트 게시자의 표시 텍스트입니다. (Get)
get_Handle 이 그림 개체 내에서 관리되는 그림의 핸들을 지정된 위치로 검색합니다.
get_Height 그림 개체에서 그림의 현재 높이를 검색합니다.

get_hFont 이 글꼴 개체에서 설명하는 글꼴에 대한 핸들을 검색합니다.
get_hPal 그림 개체에서 현재 사용하는 팔레트의 복사본을 검색합니다.
get_Italic 글꼴의 현재 기울임꼴 속성을 가져옵니다.
get_KeepOriginalFormat 그림의 KeepOriginalFormat 속성의 현재 값을 검색합니다.
get_Name 글꼴 패밀리의 이름을 검색합니다.
get_Name 이벤트 속성의 이름입니다. (Get)
get_OwnerSID 이벤트 게시자 작성자의 보안 식별자입니다. (Get)
get_PublisherName 이벤트 게시자의 표시 이름입니다. (Get)
get_PublisherType 이벤트 게시자에 대한 선택적 관리 그룹입니다. (Get)
get_Size 글꼴의 점 크기를 검색합니다.
get_Strikethrough 글꼴의 현재 Strikethrough 속성을 가져옵니다.
get_Type 그림 개체에 포함된 그림의 현재 형식을 검색합니다.

get_Underline 글꼴의 현재 밑줄 속성을 가져옵니다.
get_Value 이벤트 속성의 값입니다. (Get)
get_Weight 글꼴의 현재 Weight 속성을 가져옵니다.
get_Width 그림 개체에서 그림의 현재 너비를 검색합니다.
GetActivationPolicy 개체에 대한 현재 활성화 정책을 검색합니다. 이 메서드는 비활성 개체가 마우스 포인터 아래에 있을 때 WM_SETCURSOR 또는 WM_MOUSEMOVE 메시지를 수신할 때 컨테이너에서 호출됩니다.
GetAdvise IViewObject::SetAdvise에 대한 최신 호출에 사용된 개체에 대한 권고 연결을 검색합니다.
GetAllAccessRights 지정된 개체에 대한 액세스 권한 및/또는 소유자 및 그룹의 전체 목록을 가져옵니다.
GetBindOptions 이 바인딩 컨텍스트에 저장된 바인딩 옵션을 검색합니다.
GetBorder 개체가 활성 상태인 동안 도구 모음 및 컨트롤에 대한 외부 사각형을 검색합니다.
GetBoundSource 연결이 활성 상태인 경우 링크 원본에 대한 포인터를 검색합니다.
GetBuffer IRpcChannelBuffer::GetBuffer 메서드(objidl.h)는 전송을 위해 데이터를 마샬링할 수 있는 버퍼를 검색합니다.

GetBuffer IRpcChannelBuffer::GetBuffer(objidlbase.h) 메서드는 전송을 위해 데이터를 마샬링할 수 있는 버퍼를 검색합니다.
GetCanonicalFormatEtc 잠재적으로 다르지만 논리적으로 동등한 FORMATETC 구조를 제공합니다. 이 메서드를 사용하여 두 개의 다른 FORMATETC 구조가 동일한 데이터를 반환할지 여부를 확인하여 중복 렌더링이 필요하지 않습니다.
GetCapture 현재 위치 활성 창 없는 개체에 의해 호출되어 마우스 캡처가 여전히 있는지 여부를 확인합니다.
GetCategoryDesc 특정 범주 ID에 대한 지역화된 설명 문자열을 검색합니다.
GetClassFile 지정된 파일 이름과 연결된 CLSID를 반환합니다.
GetClassForHandler IStdMarshalInfo::GetClassForHandler 메서드(objidl.h)는 표준 마샬링 중에 대상 프로세스에 사용할 개체 처리기의 CLSID를 검색합니다.
GetClassForHandler IStdMarshalInfo::GetClassForHandler(objidlbase.h) 메서드는 표준 마샬링 중에 대상 프로세스에 사용할 개체 처리기의 CLSID를 검색합니다.
GetClassID 개체의 CLSID(클래스 식별자)를 검색합니다.
GetClassInfo 개체의 형식 정보에 대한 ITypeInfo 인터페이스에 대한 포인터를 검색합니다. 개체의 형식 정보는 형식 라이브러리에 있는 개체의 coclass 항목에 해당합니다.
GetClassObject 클래스 개체를 검색합니다.
GetClientSite 포함된 개체의 클라이언트 사이트에 대한 포인터를 검색합니다.

GetClipboardData 이 메서드가 호출되는 포함된 개체의 현재 내용을 포함하는 데이터 개체를 검색합니다. 이 데이터 개체에 대한 포인터를 사용하면 원본과 동일한 데이터를 사용하여 포함된 새 개체를 만들 수 있습니다.
GetColorSet 해당 매개 변수를 사용하여 개체가 IViewObject::D raw 메서드에서 그리는 데 사용할 논리 팔레트를 반환합니다.
GetComparisonData 모니커에서 다른 모니커와 같은지 테스트하는 데 사용할 수 있는 데이터를 검색합니다.
GetConnectionInterface 이 연결 지점에서 관리하는 나가는 인터페이스의 IID를 검색합니다.
GetConnectionPointContainer 부모 연결 가능한 개체에 대한 IConnectionPointContainer 인터페이스 포인터를 검색합니다.
GetContainer 개체의 컨테이너에 대한 포인터를 검색합니다.
GetContentExtent 컨트롤의 콘텐츠 범위를 가져옵니다.
GetControlInfo 컨트롤의 키보드 니모닉 및 동작에 대한 정보를 검색합니다.
GetConvertInfo 지정된 개체와 연결된 변환 정보를 가져옵니다. (ANSI)
GetConvertInfo 지정된 개체와 연결된 변환 정보를 가져옵니다. (유니코드)
GetCurFile 개체와 연결된 파일의 현재 이름을 검색합니다. 현재 작업 파일이 없는 경우 이 메서드는 개체에 대한 기본 저장 프롬프트를 검색합니다.

GetCurrentApartmentType IComThreadingInfo::GetCurrentApartmentType 메서드(objidl.h)는 호출자가 실행 중인 아파트 유형을 검색합니다.
GetCurrentApartmentType IComThreadingInfo::GetCurrentApartmentType(objidlbase.h) 메서드는 호출자가 실행 중인 아파트 유형을 검색합니다.
GetCurrentLogicalThreadId IComThreadingInfo::GetCurrentLogicalThreadId 메서드(objidl.h)는 호출자가 실행 중인 스레드의 GUID를 검색합니다.
GetCurrentLogicalThreadId IComThreadingInfo::GetCurrentLogicalThreadId(objidlbase.h) 메서드는 호출자가 실행 중인 스레드의 GUID를 검색합니다.
GetCurrentThreadType IComThreadingInfo::GetCurrentThreadType 메서드(objidl.h)는 호출자가 실행 중인 스레드 유형을 검색합니다.
GetCurrentThreadType IComThreadingInfo::GetCurrentThreadType(objidlbase.h) 메서드는 호출자가 실행 중인 스레드 유형을 검색합니다.
GetData 원본 데이터 개체에서 데이터를 가져오기 위해 데이터 소비자가 호출합니다.
GetDataHere 원본 데이터 개체에서 데이터를 가져오기 위해 데이터 소비자가 호출합니다. 이 메서드는 호출자가 지정된 스토리지 매체를 할당하고 해제해야 한다는 점에서 GetData 메서드와 다릅니다.
Getdc 컨테이너에서 화면 또는 호환되는 디바이스에 대한 디바이스 컨텍스트에 대한 핸들이 있는 개체를 제공합니다.
GetDefaultProperty 이벤트 게시자와 연결된 속성 모음에서 명명된 속성 및 해당 값을 검색합니다.

GetDefaultPropertyCollection 이벤트 게시자 개체와 연결된 속성 모음에 포함된 속성을 열거하는 컬렉션 개체를 만듭니다.
GetDescription 실행 취소 또는 다시 실행 사용자 인터페이스에서 사용할 수 있는 실행 취소 단위에 대한 설명을 검색합니다.
GetDestCtx IRpcChannelBuffer::GetDestCtx 메서드(objidl.h)는 RPC 채널의 대상 컨텍스트를 검색합니다.
GetDestCtx IRpcChannelBuffer::GetDestCtx(objidlbase.h) 메서드는 RPC 채널의 대상 컨텍스트를 검색합니다.
GetDisplayName 모니커의 표시 이름을 검색합니다.
GetDisplayString 지정된 속성을 설명하는 텍스트 문자열을 검색합니다.
GetDocMiscStatus 문서 개체에 대한 상태 정보를 검색합니다.
GetDocument 이 뷰를 소유하는 문서 개체에서 IUnknown 인터페이스 포인터를 가져옵니다.
GetDropTarget 끌어서 놓기를 지원하는 현재 위치 활성 창 없는 개체에 대한 IDropTarget 인터페이스를 검색합니다.
GetExtendedControl 컨테이너가 실제 컨트롤을 래핑하는 데 사용하는 확장 컨트롤에 대한 IDispatch 포인터를 검색합니다.
GetExtent 실행 중인 개체의 현재 표시 크기를 검색합니다.

GetExtent 지정된 대상 디바이스에서 지정된 뷰 개체를 그릴 크기를 검색합니다.
GetFocus 현재 위치 활성 창 없는 개체에 의해 호출되어 키보드 포커스가 여전히 있는지 여부를 확인합니다.
GetGUID 개체에 대해 지정된 GUID를 검색합니다.
GetHandle ISynchronizeHandle::GetHandle 메서드(objidl.h)는 동기화 개체에 대한 핸들을 검색합니다.
GetHandle ISynchronizeHandle::GetHandle(objidlbase.h) 메서드는 동기화 개체에 대한 핸들을 검색합니다.
GetIID 이 ICallIndirect 구현에서 지원하는 인터페이스 ID를 검색합니다.
GetIIDAndMethod 인터페이스 ID 또는 메서드 번호를 검색합니다.
GetInfo 호출 프레임에 대한 정보를 검색합니다.
GetInfoOfIndex 지정된 인덱스에서 형식 정보를 검색합니다.
GetInPlaceSite 이 뷰 개체와 연결된 뷰 사이트를 검색합니다.
GetInterfaceFromGlobal 호출하는 아파트에서 사용할 수 있는 개체의 인터페이스에 대한 포인터를 검색합니다. 이 인터페이스는 현재 전역 인터페이스 테이블에 등록되어 있어야 합니다.
GetLastRedoDescription 다시 실행 스택 위에 있는 최상위 실행 취소 단위에 대한 설명을 검색합니다.

GetLastUndoDescription 실행 취소 스택 위에 있는 최상위 실행 취소 단위에 대한 설명을 검색합니다.
GetLastUpdate 개체가 마지막으로 업데이트된 시간을 확인합니다. (ANSI)
GetLastUpdate 개체가 마지막으로 업데이트된 시간을 확인합니다. (유니코드)
GetLicInfo 이 클래스 팩터리의 라이선스 기능에 대한 정보를 검색합니다.
GetLinkSource 링크 대화 상자에 표시할 수 있는 링크에 대한 정보를 검색합니다. (ANSI)
GetLinkSource 링크 대화 상자에 표시할 수 있는 링크에 대한 정보를 검색합니다. (유니코드)
GetLinkUpdateOptions 링크에 대한 현재 업데이트 옵션을 결정합니다. (ANSI)
GetLinkUpdateOptions 링크에 대한 현재 업데이트 옵션을 결정합니다. (유니코드)
GetLocaleID 속성 페이지에서 로캘별 설정을 조정하는 데 사용할 수 있는 로캘 식별자(LCID)를 검색합니다.
GetLogLimits 로그의 현재 경계에 대한 정보를 검색합니다.
GetMarshalingContextAttribute IMarshalingStream::GetMarshalingContextAttribute 메서드(objidl.h)는 마샬링 컨텍스트에 대한 정보를 가져옵니다.
GetMarshalingContextAttribute IMarshalingStream::GetMarshalingContextAttribute(objidlbase.h) 메서드는 마샬링 컨텍스트에 대한 정보를 가져옵니다.

GetMarshalSizeMax 마샬링하는 동안 필요한 버퍼의 최대 크기를 검색합니다.
GetMarshalSizeMax 호출 프레임을 마샬링하는 데 필요한 바이트 수의 상한을 검색합니다.
GetMethodInfo 호출 프레임에서 인터페이스 메서드에 대한 정보를 검색합니다.
GetMiscStatus 생성 및 로드 시 개체의 상태를 검색합니다.
GetMoniker 개체의 클라이언트 사이트에 대한 모니커를 검색합니다. 개체는 dwAssign에 대한 값을 지정하여 자체 또는 컨테이너의 모니커를 강제로 할당할 수 있습니다.
GetMoniker 호출자가 개체에 연결하는 데 사용할 수 있는 포함된 개체의 모니커를 검색합니다.
GetMultiTypeInfoCount 이 개체가 제공해야 하는 형식 정보 블록의 수를 검색합니다.
Getnames 이 호출의 메서드 또는 인터페이스 이름을 검색합니다.
GetNaturalExtent 사용자가 크기를 조정할 때 사용할 개체에 대한 컨테이너의 크기 조정 힌트를 제공합니다.
GetNextLink 컨테이너의 링크를 열거합니다. (ANSI)
GetNextLink 컨테이너의 링크를 열거합니다. (유니코드)
GetObject VBA 개체의 인터페이스에 대한 포인터를 가져옵니다.

GetObject 지정된 개체에 대한 포인터를 검색합니다.
GetObject 지정한 모니커로 식별된 개체가 실행 중인지 여부를 확인하고, 실행 중인 경우 해당 개체에 대한 포인터를 검색합니다.
GetObjectInfo 개체의 크기, 형식, 이름 및 위치 정보를 가져옵니다. (ANSI)
GetObjectInfo 개체의 크기, 형식, 이름 및 위치 정보를 가져옵니다. (유니코드)
GetObjectParam 바인딩 컨텍스트의 문자열 키 테이블 포인터에서 지정된 키와 연결된 개체에 대한 인터페이스 포인터를 검색합니다.
GetObjectStorage 지정된 개체의 스토리지에 대한 포인터를 검색합니다.
GetOleObject Visual Basic 컨테이너의 OLE 컨트롤에 대한 포인터를 가져옵니다.
GetOpenParentState 가장 안쪽에 열려 있는 부모 실행 취소 단위에 대한 상태 정보를 검색합니다. (IOleUndoManager.GetOpenParentState)
GetPageContainer 모든 페이지를 포함하는 전체 속성 프레임 대화 상자를 나타내는 개체에 대한 포인터를 검색합니다. 이 메서드를 호출하면 한 페이지가 다른 페이지로 이동할 수 있습니다.
GetPageInfo 속성 페이지에 대한 정보를 검색합니다.
GetPageInfo 문서의 첫 번째 페이지 수와 총 페이지 수를 검색합니다.

GetPages 이 개체의 속성 시트에 표시할 수 있는 속성 페이지의 목록을 검색합니다.
GetParam 호출 프레임에서 지정된 매개 변수의 값을 검색합니다.
GetParamInfo 지정된 매개 변수에 대한 정보를 검색합니다.
GetParentState 가장 안쪽에 열려 있는 부모 실행 취소 단위에 대한 상태 정보를 검색합니다. (IOleParentUndoUnit.GetParentState)
GetPredefinedStrings 지정된 속성이 허용할 수 있는 허용 값에 대한 배열 설명 문자열을 검색합니다.
GetPredefinedValue 미리 정의된 문자열 이름과 연결된 지정된 속성의 값을 검색합니다.
GetProperty IContext::GetProperty 메서드(objidl.h)는 컨텍스트에서 지정된 컨텍스트 속성을 검색합니다.
GetProperty IContext::GetProperty(objidlbase.h) 메서드는 컨텍스트에서 지정된 컨텍스트 속성을 검색합니다.
GetRect 요청된 그리기 측면을 설명하는 사각형을 검색합니다.
GetRect 뷰가 활성화되거나 활성화될 뷰포트의 사각형 좌표를 검색합니다.
GetRegisteredSink 등록된 이벤트 싱크를 검색합니다.
GetReturnValue 호출 프레임에 저장된 반환 값을 검색합니다.

GetRunningClass 실행 중인 개체의 CLSID를 검색합니다.
GetRunningObjectTable 이 바인딩 컨텍스트가 실행 중인 컴퓨터의 ROT(실행 중인 개체 테이블)에 대한 인터페이스 포인터를 검색합니다.
GetRunningObjectTable ROT(로컬 실행 개체 테이블)의 IRunningObjectTable 인터페이스에 대한 포인터를 반환합니다.
GetSite SetSite를 사용하여 전달된 최신 사이트를 검색합니다.
GetSize IMalloc::GetSize 메서드(objidl.h)는 이전에 할당된 메모리 블록의 크기를 검색합니다.
GetSize IMalloc::GetSize(objidlbase.h) 메서드는 이전에 할당된 메모리 블록의 크기를 검색합니다.
GetSizeMax 개체를 저장하는 데 필요한 스트림의 크기를 검색합니다. (IPersistStreamInit.GetSizeMax)
GetSizeMax 개체를 저장하는 데 필요한 스트림의 크기를 검색합니다. (IPersistStream.GetSizeMax)
GetSourceDisplayName 연결된 개체의 링크 원본 표시 이름을 검색합니다.
GetSourceMoniker 연결된 개체의 링크 원본을 식별하는 모니커를 검색합니다.
GetStackLocation 이 호출 프레임이 바인딩된 스택 위치를 검색합니다.
GetStackSize 메서드 호출에서 반환하기 위해 스택에서 팝되어야 하는 바이트 수를 검색합니다.

GetTimeOfLastChange 이 모니커로 식별된 개체가 마지막으로 변경된 시간을 검색합니다.
GetTimeOfLastChange 개체가 마지막으로 수정된 시간을 검색합니다.
GetUnitType 실행 취소 단위에 대한 CLSID 및 형식 식별자를 검색합니다.
GetUnmarshalClass 경계 해제 코드의 CLSID를 검색합니다.
GetUpdateOptions 연결된 개체가 캐시된 데이터를 업데이트하는 빈도를 나타내는 값을 검색합니다.
GetUserClassID 최종 사용자에게 개체를 식별하는 문자열에 해당하는 CLSID인 개체의 클래스 식별자를 검색합니다.
GetUserType 메뉴, 목록 상자 및 대화 상자와 같은 사용자 인터페이스 요소에 표시할 개체의 사용자 유형 이름을 검색합니다.
GetViewInfo 개체와 연결된 뷰 정보를 가져옵니다. (ANSI)
GetViewInfo 개체와 연결된 뷰 정보를 가져옵니다. (유니코드)
GetViewStatus 개체의 불투명도 및 지원되는 그리기 측면에 대한 정보를 검색합니다.
GetWindow 현재 위치 활성화(프레임, 문서, 부모 또는 현재 위치 개체 창)에 참여하는 창 중 하나에 대한 핸들을 검색합니다.

GetWindowContext 현재 위치 개체가 창 개체 계층 구조를 구성하는 창 인터페이스와 개체의 현재 위치 활성화 창이 있어야 하는 부모 창의 위치를 검색할 수 있도록 합니다.
Givefeedback DoDragDrop 함수에 시각적 효과를 지정하는 열거형 값을 제공하여 끌어서 놓기 작업 중에 원본 애플리케이션에서 최종 사용자에게 시각적 피드백을 제공할 수 있습니다.
GrantAccessRights 새 액세스 권한 목록을 개체의 기존 액세스 권한과 병합합니다.
HACCEL_UserFree HACCEL_UserFree 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HACCEL_UserFree64 HACCEL_UserFree64 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HACCEL_UserMarshal HACCEL을 RPC 버퍼로 마샬링합니다.
HACCEL_UserMarshal64 HACCEL 개체를 RPC 버퍼로 마샬링합니다.
HACCEL_UserSize HACCEL_UserSize 함수(oleidl.h)는 HACCEL 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
HACCEL_UserSize64 HACCEL_UserSize64 함수(oaild.h)는 HACCEL 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
HACCEL_UserUnmarshal HACCEL_UserUnmarshal 함수(oleidl.h)는 RPC 버퍼에서 HACCEL 개체를 숨기지 않습니다.
HACCEL_UserUnmarshal64 HACCEL_UserUnmarshal64 함수(oleidl.h)는 RPC 버퍼에서 HACCEL 개체를 숨기지 않습니다.

HandleInComingCall 들어오는 호출에 대한 단일 진입점을 제공합니다.
HandsOffStorage 개체가 컨테이너에 의해 전달된 모든 스토리지 개체를 해제하고 HandsOff 모드로 전환하도록 지시합니다.
해시 모니커의 내부 상태를 사용하여 해시 값을 만듭니다.
HBITMAP_UserFree RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HBITMAP_UserFree64 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HBITMAP_UserMarshal HBITMAP 개체를 RPC 버퍼로 마샬링합니다.
HBITMAP_UserMarshal64 HBITMAP 개체를 RPC 버퍼로 마샬링합니다.
HBITMAP_UserSize HBITMAP 개체의 와이어 크기를 계산하고 해당 핸들과 데이터를 가져옵니다.
HBITMAP_UserSize64 HBITMAP 개체의 와이어 크기를 계산하고 해당 핸들과 데이터를 가져옵니다.
HBITMAP_UserUnmarshal RPC 버퍼에서 HBITMAP 개체의 숨을 풉니다.
HBITMAP_UserUnmarshal64 RPC 버퍼에서 HBITMAP 개체의 숨을 풉니다.
HDC_UserFree HDC_UserFree 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.

HDC_UserFree64 HDC_UserFree64 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HDC_UserMarshal HDC_UserMarshal 함수(oleidl.h)는 HDC 개체를 RPC 버퍼로 마샬링합니다.
HDC_UserMarshal64 HDC_UserMarshal64 함수(oleidl.h)는 HDC 개체를 RPC 버퍼로 마샬링합니다.
HDC_UserSize HDC_UserSize 함수(oleidl.h)는 HDC 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
HDC_UserSize64 HDC_UserSize64 함수(oleidl.h)는 HDC 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
HDC_UserUnmarshal HDC_UserUnmarshal 함수(oleidl.h)는 RPC 버퍼에서 HDC 개체를 숨기지 않습니다.
HDC_UserUnmarshal64 HDC_UserUnmarshal64 함수(oleidl.h)는 RPC 버퍼에서 HDC 개체를 숨기지 않습니다.
HeapMinimize IMalloc::HeapMinimize 메서드(objidl.h)는 사용되지 않는 메모리를 운영 체제에 해제하고, 인접한 사용 가능한 블록을 병합하고, 사용 가능한 페이지를 커밋하여 힙을 최소화합니다.
HeapMinimize IMalloc::HeapMinimize(objidlbase.h) 메서드는 사용되지 않는 메모리를 운영 체제에 해제하여 힙을 최소화합니다.
도움말 최종 사용자 요청에 대한 응답으로 속성 페이지 도움말을 호출합니다.
HGLOBAL_UserFree HGLOBAL_UserFree 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.

HGLOBAL_UserFree64 HGLOBAL_UserFree64 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HGLOBAL_UserMarshal HGLOBAL을 RPC 버퍼로 마샬링합니다.
HGLOBAL_UserMarshal64 HGLOBAL 개체를 RPC 버퍼로 마샬링합니다.
HGLOBAL_UserSize HGLOBAL_UserSize 함수(oleidl.h)는 HGLOBAL 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
HGLOBAL_UserSize64 HGLOBAL_UserSize64 함수(oleidl.h)는 HGLOBAL 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
HGLOBAL_UserUnmarshal HGLOBAL_UserUnmarshal 함수(oleidl.h)는 RPC 버퍼에서 HGLOBAL 개체를 숨기지 않습니다.
HGLOBAL_UserUnmarshal64 HGLOBAL_UserUnmarshal64 함수(oleidl.h)는 RPC 버퍼에서 HGLOBAL 개체를 숨기지 않습니다.
HMENU_UserFree HMENU_UserFree 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HMENU_UserFree64 HMENU_UserFree64 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HMENU_UserMarshal HMENU_UserMarshal 함수(oleidl.h)는 HMENU를 RPC 버퍼로 마샬링합니다.
HMENU_UserMarshal64 HMENU_UserMarshal64 함수(oleidl.h)는 HMENU를 RPC 버퍼로 마샬링합니다.

HMENU_UserSize HMENU_UserSize 함수(oleidl.h)는 HMENU 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
HMENU_UserSize64 HMENU_UserSize64 함수(oleidl.h)는 HMENU 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
HMENU_UserUnmarshal HMENU_UserUnmarshal 함수(oleidl.h)는 RPC 버퍼에서 HMENU 개체를 숨기지 않습니다.
HMENU_UserUnmarshal64 HMENU_UserUnmarshal64 함수(oleidl.h)는 RPC 버퍼에서 HMENU 개체를 숨기지 않습니다.
HPALETTE_UserFree RPC 스텁 파일(HPALETTE_UserFree)에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HPALETTE_UserFree64 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다. (HPALETTE_UserFree64)
HPALETTE_UserMarshal HPALETTE 개체를 RPC 버퍼로 마샬링합니다. (HPALETTE_UserMarshal)
HPALETTE_UserMarshal64 HPALETTE 개체를 RPC 버퍼로 마샬링합니다. (HPALETTE_UserMarshal64)
HPALETTE_UserSize HPALETTE 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다. (HPALETTE_UserSize)
HPALETTE_UserSize64 HPALETTE 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다. (HPALETTE_UserSize64)
HPALETTE_UserUnmarshal RPC 버퍼에서 HPALETTE 개체를 언마샬링합니다. (HPALETTE_UserUnmarshal)
HPALETTE_UserUnmarshal64 RPC 버퍼에서 HPALETTE 개체를 언마샬링합니다. (HPALETTE_UserUnmarshal64)

HRESULT_CODE 지정된 HRESULT의 코드 부분을 추출합니다.
HRESULT_FACILITY 지정된 HRESULT의 기능을 추출합니다.
HRESULT_FROM_NT NT 상태 값을 HRESULT 값에 매핑합니다.
HRESULT_FROM_WIN32 시스템 오류 코드를 HRESULT 값에 매핑합니다.
HRESULT_SEVERITY 지정된 HRESULT의 심각도 필드를 추출합니다.
HWND_UserFree HWND_UserFree 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HWND_UserFree64 HWND_UserFree64 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HWND_UserMarshal HWND_UserMarshal 함수(oleidl.h)는 HWND 개체를 RPC 버퍼로 마샬링합니다.
HWND_UserMarshal64 HWND_UserMarshal64 함수(oleidl.h)는 HWND 개체를 RPC 버퍼로 마샬링합니다.
HWND_UserSize HWND_UserSize 함수(oleidl.h)는 HWND 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
HWND_UserSize64 HWND_UserSize64 함수(oleidl.h)는 HWND 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.

HWND_UserUnmarshal HWND_UserUnmarshal 함수(oleidl.h)는 RPC 버퍼에서 HWND 개체를 숨기지 않습니다.
HWND_UserUnmarshal64 HWND_UserUnmarshal64 함수(oleidl.h)는 RPC 버퍼에서 HWND 개체를 숨기지 않습니다.
IIDFromString StringFromIID 함수에서 생성된 문자열을 원래 IID(인터페이스 식별자)로 다시 변환합니다.
ImpersonateClient IServerSecurity::ImpersonateClient 메서드(objidl.h)를 사용하면 서버가 호출 기간 동안 클라이언트를 가장할 수 있습니다.
ImpersonateClient IServerSecurity::ImpersonateClient(objidlbase.h) 메서드를 사용하면 서버가 호출 기간 동안 클라이언트를 가장할 수 있습니다.
Init 프로세스 서버를 초기화합니다.
InitCache 지정된 데이터 개체에서 제공하는 데이터를 사용하여 필요에 따라 캐시를 채웁니다.
InitFromData 동일한 컨테이너 또는 클립보드에 상주할 수 있는 지정된 데이터 개체의 데이터를 사용하여 새로 만든 개체를 초기화합니다.
초기화 Windows 런타임 API를 사용하도록 스레드를 초기화합니다.
InitNew 개체가 새로 만든 개체로 초기화되고 있음을 알릴 수 있습니다.
InitNew 개체를 기본 상태로 초기화합니다. 이 메서드는 IPersistStreamInit::Load 대신 호출됩니다.

InitNew 새 스토리지 개체를 초기화합니다.
InitNew 지정된 파일에 새 로그 인스턴스를 만듭니다. 해당 이름의 파일이 이미 있는 경우 덮어씁니다.
InPlaceDeactivate 활성 현재 위치 개체를 비활성화하고 개체의 실행 취소 상태를 삭제합니다.
InsertMenus 컨테이너가 현재 위치 세션 중에 사용할 복합 메뉴에 메뉴 그룹을 삽입할 수 있도록 합니다.
InvalidateRect 개체가 화면에서 현재 위치 이미지의 지정된 사각형을 무효화할 수 있도록 합니다.
InvalidateRgn 개체가 화면에서 현재 위치 이미지의 지정된 영역을 무효화할 수 있도록 합니다.
Inverse 이 모니커의 역방향 모니커를 만듭니다. 이 모니커의 오른쪽 또는 유사한 구조 중 하나로 구성된 경우 모니커는 아무 것도 구성하지 않습니다.
Invoke IRpcStubBuffer::Invoke 메서드(objidl.h)는 스텁이 나타내는 인터페이스를 호출합니다.
Invoke IRpcStubBuffer::Invoke(objidlbase.h) 메서드는 스텁이 나타내는 인터페이스를 호출합니다.
Invoke 개체에 이 활성화 레코드를 적용합니다. 마샬링 상황에서는 일반적으로 서버 쪽에서 수행되며 실제 개체의 작업이 수행되는 수단입니다.
IS_ERROR 상태 값에 대한 오류에 대한 일반 테스트를 제공합니다.
IsAccelerator 지정된 키 입력이 지정된 가속기 테이블의 액셀러레이터에 매핑되는지 여부를 결정합니다.

IsAccessAllowed 지정한 트러스티에 개체 또는 속성에 대한 액세스 권한이 있는지 여부를 확인합니다.
IsClassOfCategories 클래스가 하나 이상의 범주를 구현하는지 여부를 결정합니다.
IsConnected IRpcChannelBuffer::IsConnected 메서드(objidl.h)는 RPC 채널이 연결되어 있는지 여부를 결정합니다.
IsConnected IRpcChannelBuffer::IsConnected(objidlbase.h) 메서드는 RPC 채널이 연결되어 있는지 여부를 결정합니다.
IsDirty 개체가 스트림에 마지막으로 저장된 이후 변경되었는지 여부를 확인합니다. (IPersistStreamInit.IsDirty)
IsDirty 개체가 현재 파일에 마지막으로 저장된 이후 변경되었는지 여부를 확인합니다.
IsDirty 개체가 현재 스토리지에 마지막으로 저장된 이후 변경되었는지 여부를 확인합니다.
IsDirty 개체가 스트림에 마지막으로 저장된 이후 변경되었는지 여부를 확인합니다. (IPersistStream.IsDirty)
IsEqual 동등성을 위해 이 글꼴 개체를 다른 글꼴 개체와 비교합니다.
IsEqual 이 모니커가 지정된 모니커와 동일한지 여부를 확인합니다.
IsEqualCLSID 두 CLSID가 같은지 여부를 나타내는 부울 값으로 평가됩니다.

IsEqualGUID IsEqualGUID 매크로(objbase.h)는 두 GUID가 같은지 여부를 결정합니다.
IsEqualGUID 두 GUID가 같은지 여부를 나타내는 부울 값으로 평가됩니다.
IsEqualGUID IsEqualGUID 매크로(winddi.h)는 두 GUID가 같은지 여부를 결정합니다.
IsEqualIID 두 IID가 같은지 여부를 나타내는 부울 값으로 계산됩니다.
IsEvaluatingEdpPolicy 삭제 대상이 엔터프라이즈 보호 정책의 평가를 처리할 것인지 여부를 나타냅니다.
IsIIDSupported IRpcStubBuffer::IsIIDSupported 메서드(objidl.h)는 스텁이 특정 인터페이스의 경계 해제를 처리하도록 설계되었는지 여부를 결정합니다.
IsIIDSupported IRpcStubBuffer::IsIIDSupported(objidlbase.h) 메서드는 스텁이 특정 인터페이스의 분리를 처리하도록 설계되었는지 여부를 결정합니다.
IsImpersonating IServerSecurity::IsImpersonating 메서드(objidl.h)는 서버가 현재 클라이언트를 가장하고 있는지 여부를 나타냅니다.
IsImpersonating IServerSecurity::IsImpersonating(objidlbase.h) 메서드는 서버가 현재 클라이언트를 가장하고 있는지 여부를 나타냅니다.
IsPageDirty 속성 페이지가 활성화된 이후 또는 적용에 대한 가장 최근 호출 이후 변경되었는지 여부를 나타냅니다.
IsRunning 지정된 개체가 실행 중인지 여부를 확인합니다.

IsRunning 이 모니커로 식별된 개체가 현재 로드되고 실행 중인지 여부를 확인합니다.
IsRunning 개체가 현재 실행 중 상태인지 여부를 확인합니다.
IsRunning 지정된 모니커로 식별된 개체가 현재 실행 중인지 여부를 확인합니다.
IsSystemMoniker 이 모니커가 시스템에서 제공하는 모니커 클래스 중 하나인지 여부를 확인합니다.
IsUpToDate 개체가 최신 상태인지 여부를 확인합니다.
로드 속성 모음에서 사용할 수 있는 속성을 사용하여 개체를 초기화하고 오류가 발생할 때 제공된 오류 로그 개체에 알리도록 개체에 지시합니다.
로드 이전에 저장된 스트림에서 개체를 초기화합니다. (IPersistStreamInit.Load)
로드 지정된 파일을 열고 파일 내용에서 개체를 초기화합니다.
로드 기존 스토리지에서 개체를 로드합니다.
로드 이전에 저장된 스트림에서 개체를 초기화합니다. (IPersistStream.Load)
LoadDllServer ISurrogate::LoadDllServer 메서드(objidl.h)는 DLL 서버를 구현하는 서로게이트에 로드합니다.
LoadDllServer ISurrogate::LoadDllServer(objidlbase.h) 메서드는 DLL 서버를 구현하는 서로게이트에 로드합니다.

잠금 공유 리소스에 대한 잠금을 요청합니다.
LockContainer 명시적으로 해제될 때까지 포함된 개체에 대한 컨테이너를 계속 실행합니다.
LockInPlaceActive 컨트롤이 활성 상태로 유지되어야 하는지 여부를 나타냅니다. 이 메서드에 대한 호출은 일반적으로 이벤트를 중첩하여 개체의 활성화 상태가 이벤트 처리 내내 안정적으로 유지되도록 합니다.
LockRunning 이미 실행 중인 개체를 실행 중 상태로 잠그거나 실행 상태에서 잠금을 해제합니다. (IRunnableObject.LockRunning)
LockServer IClassFactory::LockServer(unknwnbase.h) 메서드는 메모리에서 열린 개체 애플리케이션을 잠급니다. 이렇게 하면 인스턴스를 더 빠르게 만들 수 있습니다.
LockServer IClassFactory::LockServer 메서드는 메모리에서 열린 개체 애플리케이션을 잠급 수 있습니다. 이렇게 하면 인스턴스를 더 빠르게 만들 수 있습니다.
MAKE_HRESULT MAKE_HRESULT 매크로(winerror.h)는 구성 요소 조각에서 HRESULT 값을 만듭니다.
MAKE_HRESULT MAKE_HRESULT 매크로(dmerror.h)는 구성 요소 조각에서 HRESULT 값을 만듭니다.
MAKE_SCODE 구성 요소 조각에서 SCODE 값을 만듭니다.
MapPropertyToPage 지정된 속성과 연결된 속성 페이지의 CLSID를 검색합니다.
마샬링하 프레임을 방해하지 않고 연결 가능한 데이터를 플랫 버퍼로 전환하여 호출 프레임을 마샬링합니다.

MarshalInterface IMarshal::MarshalInterface 메서드(objidl.h)는 인터페이스 포인터를 마샬링합니다.
MarshalInterface IMarshal::MarshalInterface(objidlbase.h) 메서드는 인터페이스 포인터를 마샬링합니다.
MessagePending COM이 원격 호출에 응답하기 위해 대기하는 동안 메시지가 도착했음을 나타냅니다.
MkParseDisplayName 문자열을 문자열로 명명된 개체를 식별하는 모니커로 변환합니다.
MonikerCommonPrefixWith 이 모니커(이 모니커 개체의 데이터로 구성된 모니커)가 다른 모니커와 공유하는 공통 접두사를 기반으로 새 모니커를 만듭니다.
MonikerRelativePathTo 첫 번째 지정된 모니커(또는 구조가 비슷한 모니커)의 끝에 구성될 때 두 번째 지정된 모니커를 생성하는 모니커를 제공합니다.
이동 프레임 내에서 속성 페이지 대화 상자의 위치를 지정하고 크기를 조정합니다.
다음 열거형 시퀀스에서 지정된 수의 항목을 검색합니다. (IEnumConnectionPoints.Next)
다음 열거형 시퀀스에서 지정된 수의 항목을 검색합니다. (IEnumConnections.Next)
다음 열거형 시퀀스에서 지정된 수의 항목을 검색합니다. (IEnumOleUndoUnits.Next)
다음 열거형 시퀀스에서 지정된 수의 항목을 검색합니다. (IEnumOleDocumentViews.Next)
다음 열거형 시퀀스에서 지정된 수의 항목을 검색합니다. (IEnumOLEVERB.Next)

다음 IEnumContextProps::Next 메서드(objidl.h)는 열거형 시퀀스에서 지정된 수의 항목을 검색합니다.
다음 열거형 시퀀스에서 지정된 수의 항목을 검색합니다. (IEnumFORMATETC.Next)
다음 열거형 시퀀스에서 지정된 수의 항목을 검색합니다. (IEnumMoniker.Next)
다음 열거형 시퀀스에서 지정된 수의 항목을 검색합니다. (IEnumSTATDATA.Next)
다음 IEnumString::Next 메서드(objidl.h)는 열거형 시퀀스에서 지정된 수의 항목을 검색합니다.
다음 IEnumUnknown::Next 메서드(objidl.h)는 열거형 시퀀스에서 지정된 수의 항목을 검색합니다.
다음 IEnumContextProps::Next(objidlbase.h) 메서드는 열거형 시퀀스에서 지정된 수의 항목을 검색합니다.
다음 IEnumString::Next(objidlbase.h) 메서드는 열거형 시퀀스에서 지정된 수의 항목을 검색합니다.
다음 IEnumUnknown::Next(objidlbase.h) 메서드는 열거형 시퀀스에서 지정된 수의 항목을 검색합니다.
다음 열거형 시퀀스에서 지정된 수의 항목을 검색합니다. (IEnumCATEGORYINFO.Next)
다음 열거형 시퀀스에서 지정된 수의 항목을 검색합니다. (IEnumGUID.Next)
NoteChangeTime 실행 중인 개체가 마지막으로 수정된 시간을 기록합니다. 개체는 이전에 ROT(실행 중인 개체 테이블)에 등록되어 있어야 합니다. 이 메서드는 ROT에서 마지막 변경 시간을 저장합니다.

OleBuildVersion 이 함수는 사용되지 않습니다.
OleCreate OleCreate 함수(ole.h)는 CLSID로 식별되는 포함된 개체를 만듭니다. 새 개체의 삽입을 허용하는 메뉴 항목을 구현할 수 있습니다.
OleCreate OleCreate 함수(ole2.h)는 CLSID로 식별되는 포함된 개체를 만듭니다. 최종 사용자가 개체를 삽입할 수 있도록 하는 메뉴 항목을 구현할 수 있습니다.
OleCreateDefaultHandler 기본 포함 처리기의 새 인스턴스를 만듭니다. 이 인스턴스는 포함된 개체가 실행 중 상태가 될 때 로컬 서버를 만들도록 초기화됩니다.
OleCreateEmbeddingHelper OLE 기본 개체 처리기의 조각으로 집계된 애플리케이션 제공 코드를 사용하여 OLE 포함 도우미 개체를 만듭니다. 이 도우미 개체는 호출자가 결정한 특정 컨텍스트 및 역할에서 만들고 사용할 수 있습니다.
OleCreateEx OleCreate에서 지원하는 단일 형식 대신 여러 프레젠테이션 형식 또는 데이터를 캐싱해야 하는 컨테이너의 개체를 보다 효율적으로 인스턴스화할 수 있도록 지원하여 OleCreate 기능을 확장합니다.
OleCreateFontIndirect FONTDESC 구조체의 글꼴 속성에 대한 초기 설명을 사용하여 표준 글꼴 개체를 만들고 초기화합니다.
OleCreateFromData 클립보드 또는 OLE 끌어서 놓기 작업의 일부로 검색된 데이터 전송 개체에서 포함된 개체를 만듭니다. OLE 끌어서 놓기 작업에서 붙여넣기를 구현하는 데 사용됩니다.
OleCreateFromDataEx OleCreateFromData에서 지원하는 단일 형식 대신 여러 형식의 프레젠테이션 또는 데이터를 캐싱해야 하는 컨테이너의 개체를 보다 효율적으로 인스턴스화할 수 있도록 지원하여 OleCreateFromData 기능을 확장합니다.

OleCreateFromFile OleCreateFromFile 함수(ole.h)는 명명된 파일의 내용에서 포함된 개체를 만듭니다.
OleCreateFromFile OleCreateFromFile 함수(ole2.h)는 명명된 파일의 내용에서 포함된 개체를 만듭니다.
OleCreateFromFileEx OleCreateFromFile에서 지원하는 단일 형식 대신 여러 프레젠테이션 형식 또는 데이터를 캐싱해야 하는 컨테이너의 개체를 보다 효율적으로 인스턴스화할 수 있도록 지원하여 OleCreateFromFile 기능을 확장합니다.
OleCreateLink OLE 복합 문서 연결된 개체를 만듭니다.
OleCreateLinkEx OleCreateLink에서 지원하는 단일 형식 대신 여러 형식의 프레젠테이션 또는 데이터를 캐싱해야 하는 컨테이너의 개체를 보다 효율적으로 인스턴스화할 수 있도록 지원하여 OleCreateLink 기능을 확장합니다.
OleCreateLinkFromData 클립보드 또는 OLE 끌어서 놓기 작업의 일부로 검색된 데이터 전송 개체에서 연결된 개체를 만듭니다.
OleCreateLinkFromDataEx OleCreateLinkFromData에서 지원하는 단일 형식 대신 여러 형식의 프레젠테이션 또는 데이터를 캐싱해야 하는 컨테이너의 개체를 보다 효율적으로 인스턴스화할 수 있도록 지원하여 OleCreateLinkFromData 기능을 확장합니다.
OleCreateLinkToFile 파일에 연결된 개체를 만듭니다.
OleCreateLinkToFileEx OleCreateLinkToFile에서 지원하는 단일 형식 대신 여러 형식의 프레젠테이션 또는 데이터를 캐싱해야 하는 컨테이너의 개체를 보다 효율적으로 인스턴스화할 수 있도록 지원하여 OleCreateLinkToFile 기능을 확장합니다.
OleCreateMenuDescriptor 메뉴 메시지 및 명령을 디스패치할 때 사용할 OLE 메뉴 설명자(즉, 메뉴를 설명하는 OLE 제공 데이터 구조)를 만들고 반환합니다.

OleCreatePictureIndirect PICTDESC 구조체에 따라 초기화된 새 그림 개체를 만듭니다.
OleCreatePropertyFrame 새 속성 프레임, 즉 부모가 hwndOwner인 속성 시트 대화 상자를 호출합니다. 여기서 대화 상자는 부모 창의 점(x,y)에 배치되고 캡션 lpszCaption이 있습니다.
OleCreatePropertyFrameIndirect OleCreatePropertyFrame을 호출할 때와 같이 별도의 매개 변수를 지정하지 않고 매개 변수가 포함된 구조체(OCPFIPARAMS)를 기반으로 속성 시트 대화 상자를 만듭니다.
OleCreateStaticFromData 데이터 전송 개체에서 네이티브 데이터가 없는 표현만 포함하는 정적 개체를 만듭니다.
OleDestroyMenuDescriptor OleCreateMenuDescriptor 함수에 의해 할당된 공유 메뉴 설명자를 해제하기 위해 컨테이너에서 호출됩니다.
OleDoAutoConvert 레지스트리에서 해당 개체 클래스에 대한 자동 변환이 설정된 경우 개체를 새 클래스로 자동으로 변환합니다.
OleDraw OleDraw 함수(ole.h)를 사용하면 개체를 더 쉽게 그릴 수 있습니다. IViewObject::D raw를 직접 호출하는 대신 사용할 수 있습니다.
OleDraw OleDraw 함수(ole2.h)를 사용하면 개체를 더 쉽게 그릴 수 있습니다. IViewObject::D raw를 직접 호출하는 대신 사용할 수 있습니다.
OleDuplicateData 지정된 핸들에 있는 데이터를 복제하고 중복된 데이터에 대한 핸들을 반환합니다. 원본 데이터는 클립보드 형식입니다. 이 함수를 사용하여 IDataObject와 같은 일부 데이터 전송 인터페이스를 구현할 수 있습니다.
OleFlushClipboard 클립보드 종료 시퀀스를 수행합니다. 또한 OleSetClipboard 함수에 의해 클립보드에 배치된 IDataObject 포인터를 해제합니다.

OleGetAutoConvert 지정된 CLSID의 개체에 대해 레지스트리가 다른 CLSID로 자동으로 변환되도록 설정되었는지 여부를 확인하고, 이 경우 새 CLSID를 검색합니다.
OleGetClipboard 클립보드의 내용에 액세스하는 데 사용할 수 있는 데이터 개체를 검색합니다.
OleGetClipboardWithEnterpriseInfo Windows Information Protection 지원 애플리케이션이 데이터 및 원본 애플리케이션에 대한 Windows Information Protection 정보와 함께 OLE 클립보드에서 IDataObject를 검색할 수 있도록 합니다.
OleGetIconOfClass 지정된 CLSID에 대한 아이콘 및 문자열 레이블이 포함된 메타파일의 핸들을 반환합니다.
OleGetIconOfFile 지정된 파일 이름에 대한 아이콘 및 문자열 레이블이 포함된 메타파일의 핸들을 반환합니다.
OleIconToCursor 아이콘을 커서로 변환합니다.
OleInitialize 현재 아파트에서 COM 라이브러리를 초기화하고, 동시성 모델을 STA(단일 스레드 아파트)로 식별하고, 아래 설명 섹션에 설명된 추가 기능을 사용하도록 설정합니다.
OleIsCurrentClipboard OleSetClipboard 함수에 의해 이전에 클립보드에 배치된 데이터 개체 포인터가 여전히 클립보드에 있는지 여부를 확인합니다.
OleIsRunning 복합 문서 개체가 현재 실행 중 상태인지 여부를 확인합니다.
OleLoad 지정된 스토리지 개체 내에 중첩된 개체를 메모리에 로드합니다.
OleLoadFromStream OleLoadFromStream 함수(ole.h)는 스트림에서 개체를 로드합니다.

OleLoadFromStream OleLoadFromStream 함수(ole2.h)는 스트림에서 개체를 로드합니다.
OleLoadPicture 새 그림 개체를 만들고 스트림의 내용에서 초기화합니다. 이는 NULL을 첫 번째 매개 변수로 사용하여 OleCreatePictureIndirect를 호출한 다음 IPersistStream::Load를 호출하는 것과 같습니다. (OleLoadPicture)
OleLoadPictureEx 새 그림 개체를 만들고 스트림의 내용에서 초기화합니다. 이는 NULL을 첫 번째 매개 변수로 사용하여 OleCreatePictureIndirect를 호출한 다음 IPersistStream::Load를 호출하는 것과 같습니다. (OleLoadPictureEx)
OleLoadPicturePath 새 그림 개체를 만들고 스트림의 내용에서 초기화합니다. 이는 OleCreatePictureIndirect(NULL, ...) 뒤에 IPersistStream::Load를 호출하는 것과 같습니다.
OleLockRunning 이미 실행 중인 개체를 실행 중 상태로 잠그거나 실행 상태에서 잠금을 해제합니다. (OleLockRunning)
OleMetafilePictFromIconAndLabel 지정된 아이콘과 레이블이 그려지는 메타파일을 만듭니다.
OleNoteObjectVisible 개체를 실행 중 상태로 유지하는 외부 참조를 증가 또는 감소합니다.
OleQueryCreateFromData 데이터 개체에 OleCreateFromData 또는 OleCreateStaticFromData 함수를 호출하여 포함된 개체가 될 수 있는 형식 중 하나가 있는지 확인합니다.
OleQueryLinkFromData OLE 포함 개체가 아닌 OLE 연결된 개체를 클립보드 데이터 개체에서 만들 수 있는지 여부를 결정합니다.
OleRegEnumFormatEtc OLE 개체 서버가 시스템 레지스트리에 등록한 데이터 형식을 열거하는 데 사용할 수 있는 열거형 개체를 만듭니다.

OleRegEnumVerbs 지정된 클래스에 등록된 동사의 열거형을 제공합니다. 사용자 지정 DLL 개체 애플리케이션 개발자는 이 함수를 사용하여 기본 개체 처리기의 동작을 에뮬레이트합니다.
OleRegGetMiscStatus 레지스트리에서 지정된 CLSID에서 지원하는 프레젠테이션 및 동작에 대한 기타 정보를 반환합니다.
OleRegGetUserType 레지스트리에서 지정된 클래스의 사용자 유형을 가져옵니다.
OleRun OLE 복합 문서 개체를 실행 중 상태로 전환합니다.
OleSave 트랜잭션 모드에서 열린 개체를 지정된 스토리지 개체에 저장합니다.
OleSaveToStream OleSaveToStream 함수(ole.h)는 IPersistStream 인터페이스가 있는 개체를 지정된 스트림에 저장합니다.
OleSaveToStream OleSaveToStream 함수(ole2.h)는 IPersistStream 인터페이스가 있는 개체를 지정된 스트림에 저장합니다.
OleSetAutoConvert 해당 클래스의 개체가 로드될 때 다른 클래스로 자동 변환하기 위한 CLSID를 지정합니다.
OleSetClipboard 특정 데이터 개체에 대한 포인터를 클립보드에 배치합니다. 이렇게 하면 OleGetClipboard 함수에서 데이터 개체에 액세스할 수 있습니다.
OleSetContainedObject 포함된 개체에 대한 링크를 지원하는 컨테이너에 대해 참조 계산이 올바르게 수행되도록 개체가 OLE 컨테이너에 포함되어 있음을 알 수 있습니다. (OleSetContainedObject)
OleSetMenuDescriptor 컨테이너의 프레임 창에서 OLE 디스패치 코드를 설치하거나 제거합니다.

OLESTR 문자열 리터럴을 유니코드 문자열로 변환합니다.
OleTranslateAccelerator 개체 애플리케이션에서 호출하면 개체의 컨테이너가 컨테이너의 가속기 테이블에 따라 가속기를 변환할 수 있습니다.
OleTranslateColor OLE_COLOR 형식을 COLORREF로 변환합니다.
OleUIAddVerbMenuA 지정된 개체의 동사 메뉴를 지정된 메뉴에 추가합니다. (ANSI)
OleUIAddVerbMenuW 지정된 개체의 동사 메뉴를 지정된 메뉴에 추가합니다. (유니코드)
OleUIBusyA 사용자가 동시성을 관리할 수 있도록 표준 사용 중 대화 상자를 호출합니다. (ANSI)
OleUIBusyW 사용자가 동시성을 관리할 수 있도록 표준 사용 중 대화 상자를 호출합니다. (유니코드)
OleUICanConvertOrActivateAs 레지스트리에 지정된 CLSID를 변환하거나 활성화하는 데 사용할 수 있는 OLE 개체 클래스가 있는지 확인합니다.
OleUIChangeIconA 사용자가 아이콘 파일, 실행 파일 또는 DLL에서 아이콘을 선택할 수 있는 표준 변경 아이콘 대화 상자를 호출합니다. (ANSI)
OleUIChangeIconW 사용자가 아이콘 파일, 실행 파일 또는 DLL에서 아이콘을 선택할 수 있는 표준 변경 아이콘 대화 상자를 호출합니다. (유니코드)
OleUIChangeSourceA 사용자가 링크의 원본을 변경할 수 있도록 원본 변경 대화 상자를 호출합니다. (ANSI)

OleUIChangeSourceW 사용자가 링크의 원본을 변경할 수 있도록 원본 변경 대화 상자를 호출합니다. (유니코드)
OleUIConvertA 사용자가 지정된 단일 개체의 형식 또는 지정된 개체 클래스의 모든 OLE 개체 형식을 변경할 수 있도록 표준 변환 대화 상자를 호출합니다. (ANSI)
OleUIConvertW 사용자가 지정된 단일 개체의 형식 또는 지정된 개체 클래스의 모든 OLE 개체 형식을 변경할 수 있도록 표준 변환 대화 상자를 호출합니다. (유니코드)
OleUIEditLinksA 표준 링크 대화 상자를 호출하여 사용자가 컨테이너의 연결된 개체를 수정할 수 있도록 합니다. (ANSI)
OleUIEditLinksW 표준 링크 대화 상자를 호출하여 사용자가 컨테이너의 연결된 개체를 수정할 수 있도록 합니다. (유니코드)
OleUIInsertObjectA 사용자가 개체 원본 및 클래스 이름과 개체를 자체 또는 아이콘으로 표시하는 옵션을 선택할 수 있는 표준 개체 삽입 대화 상자를 호출합니다. (ANSI)
OleUIInsertObjectW 사용자가 개체 원본 및 클래스 이름과 개체를 자체 또는 아이콘으로 표시하는 옵션을 선택할 수 있는 표준 개체 삽입 대화 상자를 호출합니다. (유니코드)
OleUIObjectPropertiesA 개체에 대한 일반, 보기 및 링크 정보를 표시하는 개체 속성 대화 상자를 호출합니다. (ANSI)
OleUIObjectPropertiesW 개체에 대한 일반, 보기 및 링크 정보를 표시하는 개체 속성 대화 상자를 호출합니다. (유니코드)
OleUIPasteSpecialA 사용자가 붙여넣거나 붙여넣을 클립보드 개체의 형식을 선택할 수 있도록 표준 붙여넣기 특수 대화 상자를 호출합니다. (ANSI)

OleUIPasteSpecialW 사용자가 붙여넣거나 붙여넣을 클립보드 개체의 형식을 선택할 수 있도록 표준 붙여넣기 특수 대화 상자를 호출합니다. (유니코드)
OleUIPromptUserA 지정된 템플릿이 있는 대화 상자를 표시하고 사용자의 응답(단추 식별자)을 반환합니다. 이 함수는 OLE 경고 메시지(예: 등록되지 않은 클래스)를 표시하는 데 사용됩니다. (ANSI)
OleUIPromptUserW 지정된 템플릿이 있는 대화 상자를 표시하고 사용자의 응답(단추 식별자)을 반환합니다. 이 함수는 OLE 경고 메시지(예: 등록되지 않은 클래스)를 표시하는 데 사용됩니다. (유니코드)
OleUIUpdateLinksA 링크 컨테이너의 모든 링크를 업데이트 업데이트 프로세스의 진행률을 보여 주는 대화 상자를 표시합니다. 사용자가 중지 단추를 누르거나 모든 링크가 처리될 때 프로세스가 중지됩니다. (ANSI)
OleUIUpdateLinksW 링크 컨테이너의 모든 링크를 업데이트 업데이트 프로세스의 진행률을 보여 주는 대화 상자를 표시합니다. 사용자가 중지 단추를 누르거나 모든 링크가 처리될 때 프로세스가 중지됩니다. (유니코드)
OleUninitialize 아파트의 COM 라이브러리를 닫고, 아파트가 보유한 클래스 팩터리, 기타 COM 개체 또는 서버를 해제하고, 아파트에서 RPC를 사용하지 않도록 설정하고, 아파트가 유지 관리하는 모든 리소스를 해제합니다.
OnAmbientPropertyChange 컨테이너의 앰비언트 속성 중 하나 이상이 변경되었음을 컨트롤에 알릴 수 있습니다.
OnCall 인터셉터에서 메서드 호출을 수신하는 경우 이벤트 싱크에 알릴 수 있습니다. 싱크는 가로채는 들어오는 메서드 호출에 바인딩된 ICallFrame 인스턴스와 함께 제공됩니다. 이 싱크를 통해 호출 프레임을 다양한 방식으로 조작할 수 있습니다.
OnChanged 바인딩 가능한 속성이 변경되었음을 싱크에 알 보냅니다.
OnClose 서버에서 호출하여 등록된 모든 권고 싱크에 개체가 실행 중에서 로드된 상태로 변경되었음을 알립니다.

OnControlInfoChanged 컨트롤의 CONTROLINFO 구조가 변경되었으며 컨테이너가 업데이트를 위해 컨트롤의 IOleControl::GetControlInfo를 호출해야 한다고 컨테이너에 알릴 수 있습니다.
OnDataChange 서버에서 호출하여 데이터 개체의 현재 등록된 알림 싱크에 개체의 데이터가 변경되었음을 알립니다.
OnDefWindowMessage 개체에 전달된 모든 메시지에 대한 기본 처리를 호출합니다.
OnDocWindowActivate 컨테이너의 문서 창이 활성화되거나 비활성화되면 현재 위치 개체에 알립니다.
Onfocus 이 컨트롤 사이트에서 관리하는 컨트롤이 포커스를 얻었거나 손실되었는지 여부를 나타냅니다.
OnFrameWindowActivate 컨테이너의 최상위 프레임 창이 활성화되거나 비활성화될 때 개체에 알립니다.
OnInactiveMouseMove 개체가 마우스 이벤트를 발생시키는 데 사용할 수 있도록 마우스 포인터를 개체 위로 이동했음을 알 수 있습니다. 이 메서드는 비활성 개체가 마우스 포인터 아래에 있을 때 WM_MOUSEMOVE 메서드를 수신할 때 컨테이너에서 호출됩니다.
OnInactiveSetCursor 비활성 개체의 마우스 포인터를 설정합니다. 이 메서드는 비활성 개체가 마우스 포인터 아래에 있을 때 WM_SETCURSOR 메서드를 수신할 때 컨테이너에서 호출됩니다.
OnInPlaceActivate 컨테이너에 해당 개체 중 하나가 활성화되고 있음을 알 수 있습니다.
OnInPlaceActivateEx 활성화 시 자신을 다시 그릴 필요가 있는지 여부를 확인하기 위해 포함된 개체에서 호출됩니다.
OnInPlaceDeactivate 개체가 더 이상 활성 상태가 아님을 컨테이너에 알깁니다.

OnInPlaceDeactivateEx 비활성화 시 개체를 다시 그려야 하는 경우 컨테이너에 알립니다.
OnLinkSrcChange 링크 원본(이름 또는 위치)이 변경되었음을 advise 싱크를 등록한 컨테이너에 알리고 컨테이너가 링크의 모니커를 업데이트할 수 있도록 합니다.
OnMnemonic 사용자가 키보드 니모닉을 나타내는 키 입력을 눌렀다는 것을 컨트롤에 알릴 수 있습니다.
OnNextAdd 컬렉션의 마지막 실행 취소 단위에 새 단위가 추가되었음을 알 수 있습니다.
OnPosRectChange 개체 범위가 변경되었다는 것을 컨테이너에 알 수 있습니다.
OnProgress 등록된 개체 및 애플리케이션에 다운로드 작업의 진행률을 알 수 있습니다.
OnRename 등록된 모든 권고 싱크에 개체 이름이 변경되었음을 알리기 위해 서버에서 호출합니다.
OnRequestEdit requestedit 속성이 변경하려고 한다는 것을 싱크에 알 보냅니다.
OnRun 캐시 개체가 필요에 따라 조언 싱크를 설정할 수 있도록 데이터 원본 개체가 실행 중 상태로 들어갔다는 것을 캐시에 알 수 있습니다.
Onsave 등록된 모든 권고 싱크에 개체가 저장되었음을 알리기 위해 서버에서 호출합니다.
OnShowWindow 포함된 개체의 창이 표시되거나 보이지 않을 때 컨테이너에 알립니다. 이 메서드는 현재 위치에서 활성화되어 컨테이너와 별도의 창이 없는 개체에는 적용되지 않습니다.

OnStatusChange 이 사이트에서 관리하는 속성 페이지가 해당 상태를 변경했음을 프레임에 알릴 수 있습니다. 즉, 페이지에서 하나 이상의 속성 값이 변경되었습니다. 속성 페이지는 대화 상자에서 변경이 발생할 때마다 이 메서드를 호출해야 합니다.
OnStop 캐시에 기존 advise 싱크를 종료해야 한다고 알 수 있습니다. 연결이 실제로 존재하는지 여부에 대한 표시는 제공되지 않습니다.
OnUIActivate 개체가 현재 위치에서 활성화될 예정이며 개체가 컨테이너의 주 메뉴를 현재 위치 복합 메뉴로 바꿀 것임을 컨테이너에 알립니다.
OnUIDeactivate 컨테이너에 사용자 인터페이스를 다시 설치하고 포커스를 가져와야 하며 개체에 실행 취소할 수 있는 상태가 있는지 여부를 알릴 수 있습니다.
OnViewChange 개체의 등록된 advise 싱크에 보기가 변경되었음을 알 수 있습니다.
OnViewStatusChange 개체의 보기 상태가 변경되었음을 싱크에 알 보냅니다.
OnWalkInterface 호출 프레임을 안내하여 호출 프레임에서 지정된 인터페이스를 찾습니다. 참조 횟수에 세심한 주의를 기울여 인터페이스를 조작하거나 바꿀 수 있습니다.
OnWindowMessage 컨테이너에서 현재 위치 활성 상태인 창 없는 개체로 메시지를 디스패치합니다.
열기 포함하는 단위의 실행 취소 스택의 일부가 되는 새 부모 실행 취소 단위를 엽니다.
열기 포함하는 단위의 실행 취소 스택의 일부가 되는 새 부모 실행 취소 단위를 엽니다.

열기 별도의 팝업 창에 문서 보기를 표시합니다. 의미 체계는 OLEIVERB_OPEN IOleObject::D oVerb과 동일합니다.
OpenLinkSource 링크의 원본을 엽니다. (ANSI)
OpenLinkSource 링크의 원본을 엽니다. (유니코드)
ParseDisplayName 지정된 표시 이름을 구문 분석하고 해당 모니커를 만듭니다.
ParseDisplayName 표시 이름을 모니커로 변환합니다.
PictureChanged 그림 개체에 그림 리소스가 변경되었음을 알 수 있습니다. 이 메서드는 연결된 싱크에 대해 DISPID_PICT_HANDLE IPropertyNotifySink::OnChanged만 호출합니다.
PostAlloc IMalloc::Alloc를 호출한 후 필요한 작업을 수행합니다.
PostDidAlloc IMalloc::D idAlloc를 호출한 후 필요한 작업을 수행합니다.
PostFree IMalloc::Free를 호출한 후 필요한 작업을 수행합니다.
PostGetSize IMalloc::GetSize를 호출한 후 필요한 작업을 수행합니다.
PostHeapMinimize IMalloc::HeapMinimize를 호출한 후 필요한 작업을 수행합니다.
PostInitialize CoInitializeEx 함수를 호출한 후 필요한 초기화 단계를 수행합니다.

PostMessageFilter 컨트롤이 메시지를 처리한 후 컨트롤의 자체 창에서 받은 메시지를 간단한 프레임 사이트에 보냅니다.
PostRealloc IMalloc::Realloc을 호출한 후 필요한 작업을 수행합니다.
PostUninitialize CoUninitialize 함수를 호출한 후 필요한 정리 단계를 수행합니다.
PreAlloc IMalloc::Alloc를 호출하기 전에 필요한 작업을 수행합니다.
PreDidAlloc IMalloc::D idAlloc를 호출하기 전에 필요한 작업을 수행합니다.
프리프리(PreFree) IMalloc::Free를 호출하기 전에 필요한 작업을 수행합니다. 이 메서드는 Free에 전달된 포인터가 실제 할당의 시작 부분을 가리키도록 합니다.
PreGetSize IMalloc::GetSize를 호출하기 전에 필요한 작업을 수행합니다.
PreHeapMinimize IMalloc::HeapMinimize를 호출하기 전에 필요한 작업을 수행합니다.
미리 초기화 CoInitializeEx 함수를 호출하기 전에 필요한 초기화 단계를 수행합니다.
PreMessageFilter 컨트롤 자체가 처리를 수행하기 전에 컨트롤의 자체 창에서 받은 메시지를 처리할 수 있는 기회를 사이트에 제공합니다.
PreRealloc IMalloc::Realloc을 호출하기 전에 필요한 작업을 수행합니다.

PreUninitialize CoUninitialize 함수를 호출하기 전에 필요한 정리 단계를 수행합니다.
인쇄 지정된 작업 요구 사항을 사용하여 지정된 프린터에 개체를 인쇄합니다.
ProcessShutdown 프로세스를 종료합니다.
ProgIDFromCLSID 지정된 CLSID에 대한 ProgID를 검색합니다.
끌어오기 IPipeByte::P ull 메서드(objidl.h)는 파이프 소스에서 바이트 형식의 데이터를 검색합니다.
끌어오기 IPipeDouble::P ull 메서드(objidl.h)는 파이프 소스에서 double 정수 형식의 데이터를 검색합니다.
끌어오기 IPipeLong::P ull 메서드(objidl.h)는 파이프 소스에서 긴 정수 형식의 데이터를 검색합니다.
끌어오기 IPipeByte::P ull(objidlbase.h) 메서드는 파이프 소스에서 바이트 형식의 데이터를 검색합니다.
끌어오기 IPipeDouble::P ull(objidlbase.h) 메서드는 파이프 소스에서 double 정수 형식의 데이터를 검색합니다.
끌어오기 IPipeLong::P ull(objidlbase.h) 메서드는 파이프 소스에서 긴 정수 형식의 데이터를 검색합니다.
PumpMessages ASTA 스레드에서 창 메시지를 디스패치할 수 있는 경우 사용자 지정 디스패치를 수행합니다.
Push IPipeByte::P ush 메서드(objidl.h)는 바이트 형식의 데이터를 파이프 소스로 보냅니다.

Push IPipeDouble::P ush 메서드(objidl.h)는 이중 정수 형식의 데이터를 파이프 소스로 보냅니다.
Push IPipeLong::P ush 메서드(objidl.h)는 긴 정수 형식의 데이터를 파이프 소스로 보냅니다.
Push IPipeByte::P ush(objidlbase.h) 메서드는 바이트 형식의 데이터를 파이프 소스로 보냅니다.
Push IPipeDouble::P ush(objidlbase.h) 메서드는 이중 정수 형식의 데이터를 파이프 소스로 보냅니다.
Push IPipeLong::P ush(objidlbase.h) 메서드는 긴 정수 형식의 데이터를 파이프 소스로 보냅니다.
put_Bold 글꼴의 Bold 속성을 설정합니다.
put_Charset 글꼴의 문자 집합을 설정합니다.
put_Description 이벤트 게시자의 표시 텍스트입니다. (Put)
put_Italic 글꼴의 기울임꼴 속성을 설정합니다.
put_KeepOriginalFormat 그림의 KeepOriginalFormat 속성 값을 설정합니다.
put_Name 글꼴 패밀리의 새 이름을 지정합니다.
put_Name 이벤트 속성의 이름입니다. (Put)

put_OwnerSID 이벤트 게시자 작성자의 보안 식별자입니다. (Put)
put_PublisherID 이벤트 게시자의 식별자입니다.
put_PublisherName 이벤트 게시자의 표시 이름입니다. (Put)
put_PublisherType 이벤트 게시자에 대한 선택적 관리 그룹입니다. (Put)
put_Size 글꼴의 점 크기를 설정합니다.
put_Strikethrough 글꼴의 Strikethrough 속성을 설정합니다.
put_Underline 글꼴의 Underline 속성을 설정합니다.
put_Value 이벤트 속성의 값입니다. (Put)
put_Weight 글꼴의 Weight 속성을 설정합니다.
PutDefaultProperty 명명된 속성과 해당 값을 이벤트 게시자와 연결된 속성 모음에 씁니다.
쿼리 IGlobalOptions::Query 메서드(objidl.h)는 COM 런타임의 지정된 전역 속성을 쿼리합니다.
쿼리 IRpcOptions::Query 메서드(objidl.h)는 RPC 바인딩 옵션 속성의 값을 검색합니다.

쿼리 IGlobalOptions::Query(objidlbase.h) 메서드는 COM 런타임의 지정된 전역 속성을 쿼리합니다.
쿼리 IRpcOptions::Query(objidlbase.h) 메서드는 RPC 바인딩 옵션 속성의 값을 검색합니다.
QueryBlanket 클라이언트가 지정된 프록시에서 호출하는 데 사용하는 인증 정보를 검색합니다.
QueryBlanket IServerSecurity::QueryBlanket 메서드(objidl.h)는 서버의 메서드 중 하나를 호출한 클라이언트에 대한 정보를 검색합니다.
QueryBlanket IServerSecurity::QueryBlanket(objidlbase.h) 메서드는 서버의 메서드 중 하나를 호출한 클라이언트에 대한 정보를 검색합니다.
QueryContinueDrag 끌어서 놓기 작업을 계속, 취소 또는 완료해야 하는지 여부를 결정합니다. 이 메서드는 직접 호출하지 않습니다. OLE DoDragDrop 함수는 끌어서 놓기 작업 중에 이 메서드를 호출합니다.
QueryGetData 데이터 개체가 지정된 대로 데이터를 렌더링할 수 있는지 여부를 결정합니다. 붙여넣기 또는 삭제 작업을 시도하는 개체는 IDataObject::GetData를 호출하기 전에 이 메서드를 호출하여 작업이 성공할 수 있는지 여부를 표시할 수 있습니다.
QueryHitPoint 점이 개체의 지정된 측면 내에 있는지 여부를 나타냅니다.
QueryHitRect 사각형의 점이 개체의 지정된 그리기 측면 내에 있는지 여부를 나타냅니다.
QueryInterface 인터페이스 식별자를 유추하고 QueryInterface(REFIID,void)를 호출하는 도우미 함수 템플릿입니다.
QueryInterface 개체에서 지원되는 인터페이스에 대한 포인터를 검색합니다.

QueryInternalInterface IInternalUnknown::QueryInternalInterface 메서드(objidl.h)는 개체에서 지원되는 내부 인터페이스에 대한 포인터를 검색합니다.
QueryInternalInterface IInternalUnknown::QueryInternalInterface(objidlbase.h) 메서드는 개체에서 지원되는 내부 인터페이스에 대한 포인터를 검색합니다.
QueryMultipleInterfaces IMultiQI::QueryMultipleInterfaces 메서드(objidl.h)는 개체에서 지원되는 여러 인터페이스에 대한 포인터를 검색합니다.
QueryMultipleInterfaces IMultiQI::QueryMultipleInterfaces(objidlbase.h) 메서드는 개체에서 지원되는 여러 인터페이스에 대한 포인터를 검색합니다.
QueryService IServiceProvider 구현을 통해 노출되는 모든 서비스에 대한 팩터리 메서드 역할을 합니다.
QueryService IServiceProvider 구현을 통해 노출되는 모든 서비스에 대한 팩터리 메서드 역할을 합니다. CLSID 매개 변수를 허용합니다.
QueryStatus 사용자 인터페이스 이벤트를 통해 생성된 하나 이상의 명령의 상태를 개체에 쿼리합니다.
QueryTextMetrics 호출자가 할당한 구조체를 글꼴에 대한 정보로 채웁니다.
QuickActivate 컨트롤을 빠르게 활성화합니다.
ReactivateAndUndo 이전에 비활성화된 개체를 다시 활성화하여 개체의 마지막 상태를 실행 취소합니다.
ReadRecord 로그에서 레코드를 읽습니다.

ReadRecordPrefix 로그에서 레코드의 초기 부분을 읽습니다.
Realloc IMalloc::Realloc 메서드(objidl.h)는 이전에 할당된 메모리 블록의 크기를 변경합니다.
Realloc IMalloc::Realloc(objidlbase.h) 메서드는 이전에 할당된 메모리 블록의 크기를 변경합니다.
RedoTo 실행 취소 관리자가 다시 실행 스택을 통해 지정된 실행 취소 단위까지 실행 취소 작업을 다시 호출하도록 지시합니다.
감소 모니커를 가장 간단한 형태로 줄입니다.
등록 ROT(실행 중인 개체 테이블)에 개체 및 식별 모니커를 등록합니다.
RegisterCategories 하나 이상의 구성 요소 범주를 등록합니다. 각 구성 요소 범주는 CATID 및 로캘 종속 설명 문자열 목록으로 구성됩니다.
RegisterClassImplCategories 하나 이상의 구성 요소 범주를 구현하는 것으로 클래스를 등록합니다.
RegisterClassReqCategories 클래스를 하나 이상의 구성 요소 범주가 필요한 것으로 등록합니다.
RegisterDragDrop 지정된 창을 OLE 끌어서 놓기 작업의 대상이 될 수 있는 창으로 등록하고 삭제 작업에 사용할 IDropTarget 인스턴스를 지정합니다.
RegisterInterfaceInGlobal 프로세스의 한 아파트에 있는 개체에 지정된 인터페이스를 전역 인터페이스로 등록하여 다른 아파트에서 해당 인터페이스에 액세스할 수 있도록 합니다.

RegisterObjectBound 바인딩 컨텍스트를 사용하여 개체를 등록하여 바인딩 컨텍스트가 해제될 때까지 개체가 활성 상태로 유지되도록 합니다.
RegisterObjectParam 개체를 바인딩 컨텍스트의 문자열 키 포인터 테이블에서 문자열 키와 연결합니다.
RegisterSink 메서드 호출 알림을 수신하기 위한 이벤트 싱크를 등록합니다.
RelativePathTo 이 모니커와 지정된 모니커 사이에 상대 모니커를 만듭니다.
릴리스 COM 개체의 인터페이스에 대한 참조 수를 감소합니다.
ReleaseBoundObjects RegisterObjectBound 호출로 이전에 등록된 모든 개체에 대한 모든 포인터를 해제합니다.
ReleaseConnection IExternalConnection::ReleaseConnection 메서드(objidl.h)는 개체의 강력한 외부 연결 수를 감소합니다.
ReleaseConnection IExternalConnection::ReleaseConnection(objidlbase.h) 메서드는 개체의 강력한 외부 연결 수를 감소합니다.
ReleaseDC IOleInPlaceSiteWindowless::GetDC 호출로 이전에 가져온 디바이스 컨텍스트를 해제합니다.
ReleaseHfont 이전에 IFont::AddRefHfont를 사용하여 캐시에서 이 글꼴을 잠근 호출자에게 더 이상 잠금이 필요하지 않음을 글꼴 개체에 알릴 수 있습니다.
ReleaseMarshalData IMarshal::ReleaseMarshalData 메서드(objidl.h)는 마샬링된 데이터 패킷을 삭제합니다.

ReleaseMarshalData IMarshal::ReleaseMarshalData(objidlbase.h) 메서드는 마샬링된 데이터 패킷을 삭제합니다.
ReleaseMarshalData 마샬링된 데이터 패킷에 있는 인터페이스 포인터에 의해 유지되는 리소스를 해제합니다. 이 메서드는 패킷의 모든 인터페이스 포인터를 찾아 각각에 대해 CoReleaseMarshalData 함수를 호출합니다.
ReleaseMarshalData 마샬링된 데이터 패킷에 있는 인터페이스 포인터가 보유할 수 있는 리소스를 해제합니다. 이 메서드는 패킷의 모든 인터페이스 포인터를 찾아 각 인터페이스 포인터에서 CoReleaseMarshalData 함수를 호출합니다.
ReleaseRefOnProcess 프로세스의 참조 수를 감소합니다.
ReleaseStgMedium 지정된 스토리지 매체를 해제합니다.
RemoveDefaultProperty 이벤트 게시자 개체와 연결된 속성 모음에서 명명된 속성과 해당 값을 제거합니다.
RemoveMenus 복합 메뉴에서 컨테이너의 메뉴 요소를 제거합니다.
RemoveProperty IContext::RemoveProperty 메서드(objidl.h)는 컨텍스트에서 지정된 컨텍스트 속성을 제거합니다.
RemoveProperty IContext::RemoveProperty(objidlbase.h) 메서드는 컨텍스트에서 지정된 컨텍스트 속성을 제거합니다.
렌더링 원본 그림의 오프셋(xSrc,ySrc)과 복사할 차원(cxSrc,xySrc)으로 정의된 그림의 지정된 부분을 렌더링(그리기)합니다.
RequestBorderSpace 개체가 활성 상태인 동안 개체의 창 프레임 주위에 도구를 설치할 수 있는 공간이 있는지 여부를 결정합니다.

RequestLicKey 호출자가 저장하고 나중에 사용이 허가된 개체의 인스턴스를 만드는 데 사용할 수 있는 라이선스 키를 만듭니다.
RequestNewObjectLayout 컨테이너에 포함된 개체의 표시 사이트 크기를 조정하도록 요청합니다.
RequestUIActivate 개체가 UI 활성 상태로 들어가려고 했음을 컨테이너에 알 수 있습니다.
Reset 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumConnectionPoints.Reset)
Reset 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumConnections.Reset)
Reset 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumOleUndoUnits.Reset)
Reset 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumOleDocumentViews.Reset)
Reset 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumOLEVERB.Reset)
Reset IEnumContextProps::Reset 메서드(objidl.h)는 열거형 시퀀스를 시작 부분으로 다시 설정합니다.
Reset 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumFORMATETC.Reset)
Reset 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumMoniker.Reset)
Reset 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumSTATDATA.Reset)

Reset IEnumString::Reset 메서드(objidl.h)는 열거형 시퀀스를 시작 부분으로 다시 설정합니다.
Reset IEnumUnknown::Reset 메서드(objidl.h)는 열거형 시퀀스를 시작 부분으로 다시 설정합니다.
Reset ISynchronize::Reset 메서드(objidl.h)는 동기화 개체를 서명되지 않은 상태로 설정합니다.
Reset IEnumContextProps::Reset(objidlbase.h) 메서드는 열거형 시퀀스를 시작 부분으로 다시 설정합니다.
Reset IEnumString::Reset(objidlbase.h) 메서드는 열거형 시퀀스를 시작 부분으로 다시 설정합니다.
Reset IEnumUnknown::Reset(objidlbase.h) 메서드는 열거형 시퀀스를 시작 부분으로 다시 설정합니다.
Reset ISynchronize::Reset(objidlbase.h) 메서드는 동기화 개체를 서명되지 않은 상태로 설정합니다.
Reset 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumCATEGORYINFO.Reset)
Reset 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumGUID.Reset)
ResetInitializerTimeout IProcessInitControl::ResetInitializerTimeout 메서드(objidl.h)는 프로세스 초기화 시간 제한을 설정합니다.
ResetInitializerTimeout IProcessInitControl::ResetInitializerTimeout(objidlbase.h) 메서드는 프로세스 초기화 시간 제한을 설정합니다.

ResizeBorder 테두리 공간의 크기를 조정해야 하는 개체에 경고합니다.
RetryRejectedCall 다시 시도, 취소 또는 작업 전환 옵션을 제공하는 대화 상자를 표시할 수 있는 기회를 애플리케이션에 제공합니다.
RevertToSelf IServerSecurity::RevertToSelf 메서드(objidl.h)는 가장이 시작되기 전의 스레드 인증 정보를 복원합니다.
RevertToSelf IServerSecurity::RevertToSelf(objidlbase.h) 메서드는 가장이 시작되기 전의 스레드 인증 정보를 복원합니다.
취소 IRunningObjectTable::Register 호출로 이전에 등록된 ROT(실행 중인 개체 테이블)에서 항목을 제거합니다.
RevokeAccessRights 수탁자 목록에 대한 명시적 항목을 제거합니다.
RevokeDragDrop 지정된 애플리케이션 창의 등록을 OLE 끌어서 놓기 작업의 잠재적 대상으로 취소합니다.
RevokeInterfaceFromGlobal 전역 인터페이스 테이블에서 인터페이스 등록을 취소합니다.
RevokeObjectBound Bind 컨텍스트에서 개체를 제거하여 RegisterObjectBound에 대한 이전 호출을 실행 취소합니다.
RevokeObjectParam 바인딩 컨텍스트의 문자열 키 테이블 개체에서 지정된 키와 관련 포인터를 제거합니다. 이전에 RegisterObjectParam을 호출하여 테이블에 키를 삽입해야 합니다.
실행 개체를 강제로 실행합니다.

저장 지정된 속성 모음에 속성을 저장하고 필요에 따라 개체의 더티 플래그를 지우도록 개체에 지시합니다.
저장 개체를 지정된 스트림에 저장합니다. (IPersistStreamInit.Save)
저장 개체의 복사본을 지정된 파일에 저장합니다.
저장 개체와 개체에 포함된 중첩된 개체를 지정된 스토리지 개체에 저장합니다. 개체가 NoScribble 모드로 전환됩니다.
저장 개체를 지정된 스트림에 저장합니다. (IPersistStream.Save)
SaveAsFile 그림의 데이터를 파일에 저장하는 것과 동일한 형식으로 스트림에 저장합니다. 비트맵은 BMP 파일 형식을 사용하고, WMF 형식을 메타파일하고, ICO 형식을 아이콘으로 표시합니다.
SaveCompleted 개체가 해당 파일에 쓸 수 있다는 것을 개체에 알립니다.
SaveCompleted 개체가 스토리지 개체에 쓸 수 있음을 알 수 있습니다.
SaveObject 클라이언트 사이트와 연결된 포함된 개체를 저장합니다. 이 함수는 동기식입니다. 반환될 때까지 저장이 완료됩니다.
SaveViewState 뷰 상태를 지정된 스트림에 저장합니다.
SCODE_CODE 지정된 SCODE의 코드 부분을 추출합니다.

SCODE_FACILITY 지정된 SCODE의 기능을 추출합니다.
SCODE_SEVERITY 지정된 SCODE의 심각도 필드를 추출합니다.
스크롤 지정된 픽셀 수만큼 개체의 뷰를 스크롤하도록 컨테이너에 지시합니다.
ScrollRect 개체가 화면의 현재 위치 활성 이미지 내에서 영역을 스크롤할 수 있도록 합니다.
SelectPicture 지정된 디바이스 컨텍스트로 비트맵 그림을 선택하고 그림이 이전에 선택된 디바이스 컨텍스트와 그림의 GDI 핸들을 반환합니다. 이 메서드는 IPicture::get_CurDC 함께 작동합니다.
SendOnClose 개체가 닫혔다는 알림 소유자에 현재 등록된 모든 권고 싱크에 알림을 보냅니다.
SendOnDataChange 현재 이 자문 소유자 개체 인스턴스에서 처리 중인 각 조언 싱크에 대해 IAdviseSink::OnDataChange 메서드를 호출하여 설정된 연결이 있는 각 조언 싱크에 알림을 보냅니다.
SendOnRename 현재 개체 이름이 변경되었다는 알림 소유자에 등록된 모든 권고 싱크에 알림을 보냅니다.
SendOnSave 개체가 저장되었다는 알림 소유자에게 현재 등록된 모든 권고 싱크에 알림을 보냅니다.
SendReceive IRpcChannelBuffer::SendReceive 메서드(objidl.h)는 RPC 채널을 통해 메서드 호출을 서버 스텁으로 보냅니다.
SendReceive IRpcChannelBuffer::SendReceive(objidlbase.h) 메서드는 RPC 채널을 통해 메서드 호출을 서버 스텁으로 보냅니다.

설정 IGlobalOptions::Set 메서드(objidl.h)는 COM 런타임의 지정된 전역 속성을 설정합니다.
설정 IRpcOptions::Set 메서드(objidl.h)는 RPC 바인딩 옵션 속성의 값을 설정합니다.
설정 IGlobalOptions::Set(objidlbase.h) 메서드는 COM 런타임의 지정된 전역 속성을 설정합니다.
설정 IRpcOptions::Set(objidlbase.h) 메서드는 RPC 바인딩 옵션 속성의 값을 설정합니다.
set_hPal 그림 개체에 포함된 그림에 GDI 팔레트를 할당합니다.
SetAccessPolicyHint 레코드를 읽을 패턴에 대한 구현에 대한 힌트를 제공합니다.
SetAccessRights 개체의 기존 액세스 권한을 지정된 목록으로 바꿉니다.
SetActiveObject 개체와 각 프레임 및 문서 창 간의 직접 통신 채널을 제공합니다.
SetAdvise 개체 보기의 변경 내용에 대해 조언 싱크에 알 수 있도록 뷰 개체와 조언 싱크 간의 연결을 설정합니다.
SetBindOptions 바인딩 컨텍스트에 저장된 바인딩 매개 변수에 대한 새 값을 설정합니다.
SetBlanket 지정된 프록시에서 호출하는 데 사용할 인증 정보(보안 담요)를 설정합니다.
SetBorderSpace IOleInPlaceUIWindow::RequestBorderSpace 호출에서 요청된 테두리의 공간을 할당합니다.

SetCapture 현재 위치 활성 창 없는 개체가 모든 마우스 메시지를 캡처할 수 있도록 합니다.
SetClientSite 컨테이너 내에서 "클라이언트 사이트"라고 하는 표시 위치를 포함된 개체에 알립니다.
SetColorScheme 개체 애플리케이션이 지정된 개체를 편집할 때 사용해야 하는 색상표를 지정합니다.
SetContainedObject 포함된 개체에 대한 링크를 지원하는 컨테이너에 대해 참조 계산이 올바르게 수행되도록 하는 OLE 컨테이너에 포함된 개체를 알 수 있습니다. (IRunnableObject.SetContainedObject)
SetContentExtent 컨트롤의 콘텐츠 범위를 설정합니다.
SetCurrentLogicalThreadId IComThreadingInfo::SetCurrentLogicalThreadId 메서드(objidl.h)는 호출자가 실행 중인 스레드의 GUID를 설정합니다.
SetCurrentLogicalThreadId IComThreadingInfo::SetCurrentLogicalThreadId(objidlbase.h) 메서드는 호출자가 실행 중인 스레드의 GUID를 설정합니다.
Setdata 지정된 형식의 데이터와 지정된 매체의 데이터를 사용하여 캐시를 초기화합니다.
Setdata 이 메서드를 구현하는 개체로 데이터를 전송하기 위해 데이터 원본이 포함된 개체에서 호출됩니다.
SetDropSourceEnterpriseId 드롭 대상에 드롭 원본의 엔터프라이즈 ID를 제공합니다.
SetEventHandle ISynchronizeEvent::SetEventHandle 메서드(objidl.h)는 동기화 개체에 이벤트 핸들을 할당합니다.

SetEventHandle ISynchronizeEvent::SetEventHandle(objidlbase.h) 메서드는 동기화 개체에 이벤트 핸들을 할당합니다.
SetExtent 컨테이너에 할당된 디스플레이 공간의 양을 개체에 알릴 수 있습니다.
SetFocus 창이 없는 UI 활성 개체의 키보드 포커스를 설정합니다.
SetHdc 논리 매핑 모드를 설명하는 글꼴에 디바이스 컨텍스트를 제공합니다.
SetHostNames 개체에 컨테이너 애플리케이션의 이름과 포함된 복합 문서를 제공합니다.
SetInitialPageNum 문서의 첫 번째 페이지의 페이지 번호를 설정합니다.
SetInPlaceSite 컨테이너의 문서 보기 사이트를 문서의 뷰 개체와 연결합니다.
SetLinkSource 링크의 원본을 변경합니다. (ANSI)
SetLinkSource 링크의 원본을 변경합니다. (유니코드)
SetLinkUpdateOptions 링크의 업데이트 옵션을 자동 또는 수동으로 설정합니다. (ANSI)
SetLinkUpdateOptions 링크의 업데이트 옵션을 자동 또는 수동으로 설정합니다. (유니코드)
SetMenu 활성화되는 개체가 포함된 창 프레임에 복합 메뉴를 추가합니다.

SetMoniker 개체의 컨테이너 모니커, 컨테이너를 기준으로 하는 개체의 모니커 또는 개체의 전체 모니커를 알 수 있습니다.
SetObjectRects 표시할 현재 위치 개체의 양을 지정합니다.
SetObjects 속성 페이지에 이 속성 페이지와 연결된 개체에 대한 포인터 배열을 제공합니다.
SetOwner 항목의 소유자 또는 그룹을 설정합니다.
SetPageSite 속성 페이지를 초기화하고 속성 페이지가 속성 프레임과 통신하는 IPropertyPageSite 인터페이스에 대한 포인터를 페이지에 제공합니다.
SetParam 호출 프레임에서 지정된 매개 변수의 값을 설정합니다.
SetProperty IContext::SetProperty 메서드(objidl.h)는 지정된 컨텍스트 속성을 개체 컨텍스트에 추가합니다.
SetProperty IContext::SetProperty(objidlbase.h) 메서드는 지정된 컨텍스트 속성을 개체 컨텍스트에 추가합니다.
SetRatio 이 글꼴의 배율 인수를 논리 단위와 HIMETRIC 단위 간에 변환합니다.
SetRect 뷰를 활성화할 뷰포트의 사각형 좌표를 설정하거나 보기가 현재 활성화된 뷰포트의 좌표를 다시 설정합니다.
SetRectComplex 뷰포트, 스크롤 막대 및 크기 상자의 사각형 좌표를 설정합니다.

SetReturnValue 호출 프레임 내에서 반환 값을 설정합니다.
SetSite 컨테이너가 개체에 해당 사이트의 인터페이스에 대한 포인터를 전달할 수 있도록 합니다.
SetSourceDisplayName 링크 원본의 표시 이름을 설정합니다.
SetSourceMoniker 링크 원본의 모니커를 설정합니다.
SetStackLocation 이 호출 프레임이 바인딩된 스택 위치를 설정합니다.
SetStatusText 컨테이너의 프레임 창 상태 줄에 있는 현재 위치 개체에 대한 상태 텍스트를 설정하고 표시합니다.
SetUpdateOptions 연결된 개체가 캐시된 데이터를 업데이트하는 빈도를 지정합니다.
SetViewInfo 개체와 연결된 뷰 정보를 설정합니다. (ANSI)
SetViewInfo 개체와 연결된 뷰 정보를 설정합니다. (유니코드)
표시 속성 페이지 대화 상자를 표시하거나 보이지 않게 만듭니다. 페이지가 표시되면 페이지가 포커스를 그 자체로 설정해야 하며, 특히 페이지의 첫 번째 속성으로 설정해야 합니다.
표시 보기를 활성화하거나 비활성화합니다.
ShowObject 컨테이너에 해당 개체를 사용자에게 표시하도록 요청합니다. 이 메서드는 컨테이너 자체가 표시되고 최소화되지 않도록 합니다.

ShowPropertyFrame 컨테이너에 이 사이트에 포함된 컨트롤의 속성 시트를 표시하도록 지시합니다.
신호 ISynchronize::Signal 메서드(objidl.h)는 동기화 개체를 신호된 상태로 설정하고 보류 중인 대기 작업이 S_OK 반환하도록 합니다.
신호 ISynchronize::Signal(objidlbase.h) 메서드는 동기화 개체를 신호된 상태로 설정하고 보류 중인 대기 작업이 S_OK 반환하도록 합니다.
Skip 열거형 시퀀스에서 지정된 수의 항목을 건너뜁니다. (IEnumConnectionPoints.Skip)
Skip 열거형 시퀀스에서 지정된 수의 항목을 건너뜁니다. (IEnumConnections.Skip)
Skip 열거형 시퀀스에서 지정된 수의 항목을 건너뜁니다. (IEnumOleUndoUnits.Skip)
Skip 열거형 시퀀스에서 지정된 수의 항목을 건너뜁니다. (IEnumOleDocumentViews.Skip)
Skip 열거형 시퀀스에서 지정된 수의 항목을 건너뜁니다. (IEnumOLEVERB.Skip)
Skip IEnumContextProps::Skip 메서드(objidl.h)는 열거형 시퀀스에서 지정된 수의 항목을 건너뜁니다.
Skip 열거형 시퀀스에서 지정된 수의 항목을 건너뜁니다. (IEnumFORMATETC.Skip)
Skip 열거형 시퀀스에서 지정된 수의 항목을 건너뜁니다. (IEnumMoniker.Skip)
Skip 열거형 시퀀스에서 지정된 수의 항목을 건너뜁니다. (IEnumSTATDATA.Skip)

Skip IEnumString::Skip 메서드(objidl.h)는 열거형 시퀀스에서 지정된 수의 항목을 건너뜁니다.
Skip IEnumUnknown::Skip 메서드(objidl.h)는 열거형 시퀀스에서 지정된 수의 항목을 건너뜁니다.
Skip IEnumContextProps::Skip(objidlbase.h) 메서드는 열거형 시퀀스에서 지정된 수의 항목을 건너뜁니다.
Skip IEnumString::Skip(objidlbase.h) 메서드는 열거형 시퀀스에서 지정된 수의 항목을 건너뜁니다.
Skip IEnumUnknown::Skip(objidlbase.h) 메서드는 열거형 시퀀스에서 지정된 수의 항목을 건너뜁니다.
Skip 열거형 시퀀스에서 지정된 수의 항목을 건너뜁니다. (IEnumCATEGORYINFO.Skip)
Skip 열거형 시퀀스에서 지정된 수의 항목을 건너뜁니다. (IEnumGUID.Skip)
STGMEDIUM_UserFree STGMEDIUM_UserFree 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
STGMEDIUM_UserFree64 STGMEDIUM_UserFree64 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
STGMEDIUM_UserMarshal STGMEDIUM_UserMarshal 함수(oleidl.h)는 STGMEDIUM 개체를 RPC 버퍼로 마샬링합니다.
STGMEDIUM_UserMarshal64 STGMEDIUM_UserMarshal64 함수(oleidl.h)는 STGMEDIUM 개체를 RPC 버퍼로 마샬링합니다.

STGMEDIUM_UserSize STGMEDIUM_UserSize 함수(oleidl.h)는 STGMEDIUM 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
STGMEDIUM_UserSize64 STGMEDIUM_UserSize64 함수(oleidl.h)는 STGMEDIUM 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
STGMEDIUM_UserUnmarshal STGMEDIUM_UserUnmarshal 함수(oleidl.h)는 RPC 버퍼에서 STGMEDIUM 개체를 숨기지 않습니다.
STGMEDIUM_UserUnmarshal64 STGMEDIUM_UserUnmarshal64 함수(oleidl.h)는 RPC 버퍼에서 STGMEDIUM 개체를 숨기지 않습니다.
StringFromCLSID CLSID를 인쇄 가능한 문자 문자열로 변환합니다. CLSID는 항상 다른 문자열로 변환됩니다.
StringFromGUID2 GUID(Globally Unique Identifier)를 인쇄 가능한 문자 문자열로 변환합니다.
StringFromIID 인터페이스 식별자를 인쇄 가능한 문자 문자열로 변환합니다.
성공 모든 상태 값에 대한 성공에 대한 일반 테스트를 제공합니다.
TestCancel ICancelMethodCalls::TestCancel 메서드(objidl.h)는 호출이 취소되었는지 여부를 결정합니다.
TestCancel ICancelMethodCalls::TestCancel(objidlbase.h) 메서드는 호출이 취소되었는지 여부를 결정합니다.
TransformCoords HIMETRIC 단위로 표현된 좌표(OLE의 표준)를 컨테이너에서 지정한 단위로 변환합니다.

TranslateAccelerator 처리를 위해 컨트롤 사이트에 키 입력을 전달합니다.
TranslateAccelerator 처리를 위해 속성 페이지에 키 입력을 전달합니다.
TranslateAccelerator 처리를 위해 속성 프레임에 키 입력을 전달합니다.
TranslateAccelerator 컨테이너의 메시지 큐에서 메뉴 가속기 키 메시지를 처리합니다. 이 메서드는 DLL 개체 애플리케이션에서 만든 개체에만 사용해야 합니다.
TranslateAccelerator 개체가 활성 상태인 동안 컨테이너의 프레임에 대한 액셀러레이터 키 입력을 변환합니다.
TruncatePrefix 로그의 지정된 접두사를 버리면 더 이상 검색할 수 없습니다.
UIActivate 메뉴, 도구 모음 및 액셀러레이터와 같은 문서 보기의 사용자 인터페이스 요소를 활성화하거나 비활성화합니다.
UI디활성화 활성 현재 위치 개체의 사용자 인터페이스를 비활성화하고 제거합니다.
Unadvise 연결 지점 개체와 클라이언트의 싱크 간에 이전에 설정된 권고 연결을 종료합니다.
Unadvise 이전에 설정된 권고 연결을 삭제합니다. (IOleAdviseHolder.Unadvise)
Unadvise 이전에 설정된 권고 연결을 삭제합니다. (IOleObject.Unadvise)

Unadvise IDataAdviseHolder::Advise에 대한 이전 호출을 통해 설정된 데이터 개체와 권고 싱크 간의 연결을 제거합니다. 이 메서드는 일반적으로 IDataObject::D Unadvise 구현에서 호출됩니다.
UnbindSource 연결된 개체와 해당 링크 원본 간의 연결을 끊습니다.
Uncache 이전에 IOleCache::Cache를 사용하여 만든 캐시 연결을 제거합니다.
도킹 해제할 접근성 창이 이전 위치로 자동으로 이동되지 않도록 도킹 해제합니다.
UndockWindow 지정된 창 핸들이 현재 도킹된 경우 도킹 해제합니다.
UndoTo 실행 취소 관리자에게 지정한 실행 취소 단위까지 실행 취소 스택을 통해 실행 취소 작업을 다시 호출하도록 지시합니다.
고정 취소 IViewObject::Freeze를 사용하여 이전에 고정된 드로잉을 해제합니다. 이 메서드의 가장 일반적인 용도는 대역폭 인쇄입니다.
Unlock 공유 리소스에 대한 잠금을 해제합니다.
언마르샬 () 이 기존 활성화 레코드에 대한 호출의 이전에 마샬링된 [out] 매개 변수를 포함하는 데이터 패킷을 마스할 수 없습니다.
언마르샬 () 마샬링된 데이터 패킷을 활성화 레코드로 되돌립니다. 그러면 다른 방법으로 호출하거나 조작할 수 있습니다.
UnmarshalInterface IMarshal::UnmarshalInterface 메서드(objidl.h)는 인터페이스 포인터를 숨기지 않습니다.

UnmarshalInterface IMarshal::UnmarshalInterface(objidlbase.h) 메서드는 인터페이스 포인터를 unmarshalshals합니다.
UnRegisterCategories 하나 이상의 구성 요소 범주 등록을 제거합니다. 각 구성 요소 범주는 CATID 및 로캘 종속 설명 문자열 목록으로 구성됩니다.
UnRegisterClassImplCategories 클래스에서 하나 이상의 구현된 범주 식별자를 제거합니다.
UnRegisterClassReqCategories 클래스에서 하나 이상의 필수 범주 식별자를 제거합니다.
업데이트 연결된 개체에 대해 복합 문서의 캐시된 데이터를 업데이트. 링크 원본이 아직 바인딩되지 않은 경우 링크 원본에 바인딩하는 작업이 포함됩니다.
업데이트 개체 처리기의 또는 링크 개체의 데이터 또는 뷰 캐시를 업데이트.
UpdateCache 지정된 캐시를 업데이트. 이 메서드는 애플리케이션이 캐싱을 정밀하게 제어해야 하는 경우에 사용됩니다.
UpdateLink 선택한 링크를 강제로 원본에 연결하고 현재 정보를 검색합니다. (ANSI)
UpdateLink 선택한 링크를 강제로 원본에 연결하고 현재 정보를 검색합니다. (유니코드)
Wait ISynchronize::Wait 메서드(objidl.h)는 동기화 개체가 신호를 받을 때까지 대기하거나 지정된 시간 제한 기간이 경과될 때까지 기다립니다(중 먼저 제공됨).
Wait ISynchronize::Wait(objidlbase.h) 메서드는 동기화 개체가 신호를 받을 때까지 대기하거나 지정된 시간 제한 기간이 경과될 때까지 기다립니다.

WaitMultiple ISynchronizeContainer::WaitMultiple 메서드(objidl.h)는 동기화 개체가 신호를 받을 때까지 대기하거나 지정된 시간 제한 기간이 경과될 때까지 대기합니다(중 먼저 제공됨).
WaitMultiple ISynchronizeContainer::WaitMultiple(objidlbase.h) 메서드는 컨테이너의 동기화 개체가 신호를 받을 때까지 기다립니다.
워크프레임 프레임의 [in], [in, out] 또는 [out] 매개 변수에서 연결할 수 있는 인터페이스 포인터를 검색합니다.

인터페이스

IAccessControl 개체의 개체 및 속성에 대한 액세스를 관리할 수 있습니다.
IAccessibilityDockingService Windows 스토어 앱이 표시되고 스냅되지 않거나 시작 관리자가 표시될 때 애플리케이션 창을 모니터 아래쪽에 도킹합니다.
IAccessibilityDockingServiceCallback 접근성 창 도킹 이벤트를 수신합니다.
IAdviseSink 컨테이너 및 기타 개체가 데이터 변경에 대한 알림을 받고, 변경 내용을 보고, 관심 있는 개체에서 발생하는 복합 문서 변경 내용을 수신할 수 있도록 합니다.
IAdviseSink2 IAdviseSink2 인터페이스는 IAdviseSink 인터페이스의 확장이며 연결된 개체의 모니커 변경을 처리하기 위해 OnLinkSrcChange 메서드를 계약에 추가합니다.
IAdviseSinkEx 이 인터페이스는 IAdviseSink에서 파생되어 개체의 뷰 상태 변경 내용을 싱크에 알리기 위한 확장을 제공합니다.

IAgileObject IAgileObject 인터페이스(objidl.h)는 아파트 전체에서 인터페이스를 민첩하게 표시합니다.
IAgileObject IAgileObject(objidlbase.h) 인터페이스는 아파트 전체에서 인터페이스를 민첩하게 표시합니다.
IBindCtx 특정 모니커 바인딩 작업에 대한 정보를 저장하는 개체인 바인딩 컨텍스트에 대한 액세스를 제공합니다.
IBlockingLock 파일과 같은 공유 리소스에 일시적으로 배타적인 액세스를 제공하는 데 사용할 수 있는 세마포를 제공합니다.
ICallFactory ICallFactory 인터페이스(objidl.h)는 비동기 인터페이스의 메서드에 대한 호출을 처리하기 위한 호출 개체를 만듭니다.
ICallFactory ICallFactory(objidlbase.h) 인터페이스는 비동기 인터페이스의 메서드에 대한 호출을 처리하기 위한 호출 개체를 만듭니다.
ICallFrame 스택 프레임과 같은 호출 프레임을 조작할 수 있습니다.
ICallFrameEvents 메서드 호출 알림을 제공합니다.
ICallFrameWalker 흥미로운 값을 찾는 스택 프레임을 안내합니다.
ICallIndirect 기존의 직접 호출이 아닌 호출 인수에 대한 간접 참조를 사용하여 개체를 호출합니다.
ICallInterceptor 인터페이스에서 직접 호출을 알리려는 이벤트 싱크의 등록 및 등록 취소를 지원합니다.

ICallUnmarshal 원격 호출의 서버(수신) 쪽에서 사용됩니다.
ICancelMethodCalls ICancelMethodCalls::Cancel 메서드(objidl.h)는 아웃바운드 메서드 호출에 대한 취소 요청을 관리하고 서버 스레드에서 해당 메서드 호출의 현재 상태를 모니터링합니다.
ICancelMethodCalls ICancelMethodCalls(objidlbase.h) 인터페이스는 아웃바운드 메서드 호출에 대한 취소 요청을 관리하고 해당 호출의 현재 상태를 모니터링합니다.
ICatInformation 특정 클래스에서 구현되거나 필요한 범주에 대한 정보와 지정된 컴퓨터에 등록된 범주에 대한 정보를 가져옵니다.
ICatRegister 레지스트리에서 구성 요소 범주 정보를 등록 및 등록 취소하는 메서드를 제공합니다. 여기에는 사람이 읽을 수 있는 범주 이름과 지정된 구성 요소 또는 클래스에서 구현/필요한 범주가 모두 포함됩니다.
IClassActivator 클래스 개체를 검색하는 메서드를 지정합니다.
IClassFactory IClassFactory 인터페이스(unknwnbase.h)를 사용하면 개체 클래스를 만들 수 있습니다.
IClassFactory IClassFactory 인터페이스를 사용하면 개체 클래스를 만들 수 있습니다.
IClassFactory2 모든 종류의 개체 서버에서 클래스 팩터리 개체가 라이선스를 통해 개체 생성을 제어할 수 있도록 합니다.
IClientSecurity 개체의 각 개별 인터페이스 프록시에 대한 보안 설정을 클라이언트에 제어할 수 있습니다.

IComThreadingInfo IComThreadingInfo 인터페이스(objidl.h)를 사용하면 호출자가 실행 중인 아파트 및 스레드에 대한 아파트 유형, 스레드 유형 및 스레드 GUID를 가져올 수 있습니다.
IComThreadingInfo IComThreadingInfo(objidlbase.h) 인터페이스를 사용하면 호출자가 실행 중인 아파트 및 스레드에 대한 다음 정보를 얻을 수 있습니다.
IConnectionPoint 연결 가능한 개체에 대한 연결점을 지원합니다. (IConnectionPoint)
IConnectionPointContainer 연결 가능한 개체에 대한 연결점을 지원합니다. (IConnectionPointContainer)
IContext IContext 인터페이스(objidl.h)는 COM+ 컨텍스트 속성 설정을 지원합니다.
IContext IContext(objidlbase.h) 인터페이스는 COM+ 컨텍스트 속성 설정을 지원합니다.
IContextCallback 특정 COM+ 개체 컨텍스트 내에서 함수를 실행하는 메커니즘을 제공합니다.
IContinueCallback 개체에 계속할지 여부를 주기적으로 요청해야 하는 인터럽트 가능한 프로세스에 대한 일반 콜백 메커니즘을 제공합니다.
IDataAdviseHolder 데이터 개체와 하나 이상의 조언 싱크 간의 권고 연결을 만들고 관리합니다.
Idataobject 데이터 전송 및 데이터 변경 내용 알림을 사용하도록 설정합니다.
IDropSource IDropSource 인터페이스는 애플리케이션에서 끌어서 놓기 작업을 제공하기 위해 구현하는 인터페이스 중 하나입니다.

IDropSourceNotify IDropSourceNotify 인터페이스는 IDropSource 개체에서 구현되어 사용자가 잠재적인 놓기 대상 창으로 또는 밖으로 마우스를 끌 때 OLE에서 알림을 받습니다.
Idroptarget IDropTarget 인터페이스는 애플리케이션에서 끌어서 놓기 작업을 제공하기 위해 구현하는 인터페이스 중 하나입니다.
IEnterpriseDropTarget 놓기 대상 애플리케이션에서 구현하는 경우 이 인터페이스는 OLE 끌어서 놓기 엔진에 드롭 대상 애플리케이션이 엔터프라이즈 보호 정책을 평가할 것인지 여부를 확인하는 기능을 제공하고 OLE 끌어서 놓기 엔진에 드롭 소스 애플리케이션의 엔터프라이즈 ID를 드롭 대상 애플리케이션에 제공하는 방법을 제공합니다.
IEnumCATEGORYINFO 시스템에 등록된 구성 요소 범주를 열거합니다.
IEnumConnectionPoints 연결점을 열거합니다.
IEnumConnections 연결 가능한 개체에 대한 현재 연결을 열거합니다.
IEnumContextProps IEnumContextProps 인터페이스(objidl.h)는 COM+ 개체 컨텍스트와 연결된 컨텍스트 속성을 열거하는 메커니즘을 제공합니다.
IEnumContextProps IEnumContextProps(objidlbase.h) 인터페이스는 COM+ 개체 컨텍스트와 연결된 컨텍스트 속성을 열거하는 메커니즘을 제공합니다.
IEnumFORMATETC 지정된 데이터 개체에서 지원하는 형식 및 미디어를 정의하는 FORMATETC 구조를 열거합니다.
IEnumGUID 클라이언트가 COM 클래스에 대한 클래스 ID 컬렉션을 열거할 수 있도록 합니다.

IEnumMoniker 모니커 테이블의 모니커 또는 모니커 구성 요소를 열거합니다.
IEnumOleDocumentViews 문서 개체에서 지원하는 뷰를 열거합니다.
IEnumOleUndoUnits 실행 취소 또는 다시 실행 스택에서 실행 취소 단위를 열거합니다.
IEnumOLEVERB 개체에 사용할 수 있는 다른 동사를 오름차순으로 열거합니다. IEnumOLEVERB 인터페이스를 구현하는 열거자는 IOleObject::EnumVerbs에서 반환됩니다.
IEnumSTATDATA 데이터 개체에 대한 권고 연결 정보를 열거합니다.
IEnumString IEnumString 인터페이스(objidl.h)는 문자열을 열거합니다. LPWSTR은 와이드 또는 유니코드 문자의 0으로 끝나는 문자열에 대한 포인터를 나타내는 형식입니다.
IEnumString IEnumString(objidlbase.h) 인터페이스는 문자열을 열거합니다. LPWSTR은 와이드 또는 유니코드 문자의 0으로 끝나는 문자열에 대한 포인터를 나타내는 형식입니다.
IEnumUnknown IEnumUnknown 인터페이스(objidl.h)는 IUnknown 인터페이스를 사용하여 개체를 열거합니다.
IEnumUnknown IEnumUnknown(objidlbase.h) 인터페이스는 IUnknown 인터페이스를 사용하여 개체를 열거하고 여러 개체가 포함된 구성 요소의 개체를 열거할 수 있습니다.
IEventProperty 명명된 이벤트 속성을 해당 값과 연결합니다.
IEventPublisher 이벤트 게시자에 대한 정보를 등록, 수정, 제거 및 제공합니다.

IExternalConnection IExternalConnection 인터페이스(objidl.h)는 서버 개체의 마샬링된 연결 및 외부 연결 수를 관리합니다.
IExternalConnection IExternalConnection(objidlbase.h) 인터페이스는 서버 개체의 마샬링된 연결 또는 외부 연결 수를 관리합니다.
IFastRundown IFastRundown 인터페이스(objidl.h)는 인터페이스를 빠른 런다운 동작에 적합한 것으로 표시합니다.
IFastRundown IFastRundown(objidlbase.h) 인터페이스는 인터페이스를 빠른 런다운 동작에 적합한 것으로 표시합니다.
IFileBasedLogInit ILog의 파일 기반 구현 인스턴스를 초기화합니다.
IFont Windows 글꼴 개체 주위에 래퍼를 제공합니다.
IFontDisp Automation을 통해 글꼴 개체의 속성을 노출합니다. IFont 메서드의 하위 집합을 제공합니다.
IForegroundTransfer 전경 창을 COM 서버를 호스트하는 프로세스로 전송합니다.
IGetOleObject Visual Basic 컨테이너의 OLE 컨트롤에 대한 액세스를 가져옵니다.
IGetVBAObject IVBFormat 인터페이스에 대한 액세스를 가져옵니다.
IGlobalInterfaceTable 프로세스의 모든 아파트가 프로세스의 다른 아파트의 개체에 구현된 인터페이스에 액세스할 수 있도록 합니다.

IGlobalOptions IGlobalOptions 인터페이스(objidl.h)는 COM(구성 요소 개체 모델) 런타임의 전역 속성을 설정하고 쿼리합니다.
IGlobalOptions IGlobalOptions(objidlbase.h) 인터페이스는 COM(구성 요소 개체 모델) 런타임의 전역 속성을 설정하고 쿼리합니다.
IInitializeSpy COM 아파트를 입력하거나 종료할 때 초기화 또는 정리를 수행합니다.
IInternalUnknown IInternalUnknown 인터페이스(objidl.h)는 프록시의 내부 인터페이스에 액세스해야 하는 경량 클라이언트 쪽 처리기에서만 사용됩니다.
IInternalUnknown IInternalUnknown(objidlbase.h) 인터페이스는 프록시의 내부 인터페이스에 액세스해야 하는 경량 클라이언트 쪽 처리기에 사용됩니다.
Ilog 일반 하위 수준 로깅 기능을 제공합니다.
IMalloc IMalloc 인터페이스(objidl.h)는 메모리를 할당, 해제 및 관리합니다.
IMalloc IMalloc(objidlbase.h) 인터페이스는 메모리를 할당, 해제 및 관리합니다.
IMallocSpy 애플리케이션 개발자가 메모리 할당을 모니터링(감시)하고, 메모리 누수 검색, IMalloc 메서드 호출에서 메모리 오류를 시뮬레이트할 수 있습니다.
IMarshal IMarshal 인터페이스(objidl.h)를 사용하면 COM 개체가 해당 인터페이스 포인터의 마샬링을 정의하고 관리할 수 있습니다.

IMarshal IMarshal(objidlbase.h) 인터페이스를 사용하면 COM 개체가 해당 인터페이스 포인터의 마샬링을 정의하고 관리할 수 있습니다.
IMarshalingStream IMarshalingStream 인터페이스(objidl.h)는 사용자 지정 마샬링된 개체 및 unmarshalers에 대한 마샬링 컨텍스트에 대한 추가 정보를 제공합니다.
IMarshalingStream IMarshalingStream(objidlbase.h) 인터페이스는 사용자 지정 마샬링된 개체 및 언마샬러에 대한 마샬링 컨텍스트에 대한 추가 정보를 제공합니다.
IMessageDispatcher ASTA 스레드에서 창 메시지를 특수하게 처리해야 하는 구성 요소에 의해 구현된 콜백 인터페이스입니다.
IMessageFilter 동기 호출의 응답을 기다리는 동안 수신 및 발신 COM 메시지를 선택적으로 처리할 수 있는 기능을 COM 서버 및 애플리케이션에 제공합니다.
Imoniker COM 개체를 고유하게 식별하는 정보가 포함된 모니커 개체를 사용할 수 있습니다.
IMultiQI IMultiQI 인터페이스(objidl.h)를 사용하면 클라이언트가 단일 RPC 호출을 사용하여 여러 인터페이스에 대한 개체 프록시 또는 처리기를 쿼리할 수 있습니다.
IMultiQI IMultiQI(objidlbase.h) 인터페이스를 사용하면 클라이언트가 단일 RPC 호출을 사용하여 여러 인터페이스에 대해 개체 프록시 또는 처리기를 쿼리할 수 있습니다.
INoMarshal INoMarshal 인터페이스(objidl.h)는 전역 인터페이스 테이블에 마샬링되거나 저장되는 것을 지원하지 않는 개체를 표시합니다.
INoMarshal INoMarshal(objidlbase.h) 인터페이스는 전역 인터페이스 테이블에 마샬링되거나 저장되는 것을 지원하지 않는 개체를 표시합니다.

IObjContext IObjContext 인터페이스(objidl.h)는 컨텍스트에서 다양한 작업을 수행합니다.
IObjContext IObjContext(objidlbase.h) 인터페이스는 컨텍스트에서 다양한 작업을 수행합니다.
IObjectWithSite 컨테이너에서 개체와 해당 사이트 간의 통신을 지원하는 간단한 방법을 제공합니다.
IOleAdviseHolder 개체 서버에서 권고 연결 및 복합 문서 알림을 관리합니다.
IOleCache 개체 내부에 캐시되는 프레젠테이션 데이터를 제어합니다. 캐시된 프레젠테이션 데이터는 서버 애플리케이션이 실행되고 있지 않거나 사용할 수 없는 경우에도 개체의 컨테이너에서 사용할 수 있습니다.
IOleCache2 개체 클라이언트가 IOleCache::Cache를 사용하여 만든 각 캐시를 선택적으로 업데이트할 수 있습니다.
IOleCacheControl 캐시의 적절한 유지 관리를 제공합니다. 실행 중인 개체의 IDataObject 구현을 캐시에 연결하여 캐시를 유지 관리하므로 캐시가 실행 중인 개체에서 알림을 받을 수 있습니다.
IOleClientSite 포함된 개체가 표시 사이트의 위치 및 범위, 모니커, 사용자 인터페이스 및 컨테이너에서 제공하는 기타 리소스에 대한 정보를 가져오는 기본 수단을 제공합니다.
IOleCommandTarget 개체와 해당 컨테이너가 서로 명령을 디스패치할 수 있도록 합니다. 예를 들어 개체의 도구 모음에는 인쇄, 인쇄 미리 보기, 저장, 새로 만들기 및 확대/축소와 같은 명령에 대한 단추가 포함될 수 있습니다.
IOleContainer 복합 문서의 개체를 열거하거나 실행 중인 상태에서 컨테이너를 잠급니다. 컨테이너 및 개체 애플리케이션은 모두 이 인터페이스를 구현합니다.

IOleControl 컨트롤 개체에서 키보드 니모닉, 주변 속성 및 이벤트를 지원하기 위한 기능을 제공합니다.
IOleControlSite 사이트 개체가 컨테이너 내의 각 포함된 컨트롤을 관리할 수 있도록 하는 메서드를 제공합니다.
IOleDocument 문서 개체가 해당 데이터의 뷰를 만드는 기능을 컨테이너와 통신할 수 있도록 합니다.
IOleDocumentSite 문서 개체로 구현된 문서가 현재 위치 활성 개체의 일반 활성화 시퀀스를 무시하고 클라이언트 사이트에 문서 개체로 활성화하도록 직접 지시할 수 있습니다.
IOleDocumentView IOleDocumentView 인터페이스를 사용하면 컨테이너가 문서 개체에서 지원하는 각 보기와 통신할 수 있습니다.
IOleInPlaceActiveObject 현재 위치 개체와 연결된 애플리케이션의 가장 바깥 프레임 창과 포함된 개체가 포함된 애플리케이션 내의 문서 창 간에 직접 통신 채널을 제공합니다.
IOleInPlaceFrame 컨테이너의 최상위 프레임 창을 제어합니다.
IOleInPlaceObject 내부 개체의 활성화 및 비활성화를 관리하고 표시할 내부 개체 크기를 결정합니다.
IOleInPlaceObjectWindowless 창 없는 개체가 창 메시지를 처리하고 끌어서 놓기 작업에 참여할 수 있도록 합니다. IOleInPlaceObject 인터페이스에서 파생되고 확장됩니다.
IOleInPlaceSite 컨테이너와 개체의 현재 위치 클라이언트 사이트 간의 상호 작용을 관리합니다. 클라이언트 사이트는 포함된 개체의 표시 사이트이며 개체에 대한 위치 및 개념 정보를 제공합니다.
IOleInPlaceSiteEx 개체가 활성화되고 비활성화될 때 개체가 화면에서 불필요한 깜박임을 방지할 수 있도록 하는 추가 활성화 및 비활성화 알림 메서드 집합을 제공합니다.

IOleInPlaceSiteWindowless IOleInPlaceSiteEx 인터페이스를 확장합니다.
IOleInPlaceUIWindow 컨테이너 애플리케이션에 의해 구현되고 개체 애플리케이션에서 문서 또는 프레임 창의 테두리 공간을 협상하는 데 사용됩니다.
IOleItemContainer 항목 모니커가 식별하는 개체에 바인딩할 때 사용됩니다.
IOleLink 연결된 개체가 연결과 관련된 함수를 컨테이너에 제공할 수 있도록 합니다.
IOleObject 포함된 개체가 컨테이너에 기본 기능을 제공하고 해당 컨테이너와 통신하는 주체 수단으로 사용됩니다.
IOleParentUndoUnit 실행 취소 단위가 자식 실행 취소 단위를 포함하도록 설정합니다.
IOleUILinkContainerA 컨테이너에 의해 구현되고 OLE 공통 대화 상자에서 사용됩니다. 컨테이너의 링크를 관리하는 데 필요한 메서드를 제공하여 이러한 대화 상자를 지원합니다. (ANSI)
IOleUILinkContainerW 컨테이너에 의해 구현되고 OLE 공통 대화 상자에서 사용됩니다. 컨테이너의 링크를 관리하는 데 필요한 메서드를 제공하여 이러한 대화 상자를 지원합니다. (유니코드)
IOleUILinkInfoA IOleUILinkContainer 인터페이스의 확장입니다. IOleUILinkContainer에서 제공하지 않는 링크 정보인 개체가 마지막으로 업데이트된 시간을 반환합니다. (ANSI)
IOleUILinkInfoW IOleUILinkContainer 인터페이스의 확장입니다. IOleUILinkContainer에서 제공하지 않는 링크 정보인 개체가 마지막으로 업데이트된 시간을 반환합니다. (유니코드)

IOleUIObjInfoA 컨테이너에 의해 구현되고 컨테이너의 개체 속성 대화 상자 및 변환 대화 상자에서 사용됩니다. (ANSI)
IOleUIObjInfoW 컨테이너에 의해 구현되고 컨테이너의 개체 속성 대화 상자 및 변환 대화 상자에서 사용됩니다. (유니코드)
IOleUndoManager IOleUndoManager 인터페이스를 사용하면 컨테이너가 포함된 컨트롤 내에서 발생하는 작업에 대해 다단계 실행 취소 및 다시 실행 작업을 구현할 수 있습니다.
IOleUndoUnit 실행 취소 단위의 기본 인터페이스 역할을 합니다. 실행 취소 단위는 단일 작업을 실행 취소하거나 다시 실행하는 데 필요한 정보를 캡슐화합니다.
IOleWindow IOleWindow 인터페이스는 애플리케이션이 현재 위치 활성화에 참여하는 다양한 창에 대한 핸들을 가져오고 상황에 맞는 도움말 모드를 입력하고 종료할 수 있는 메서드를 제공합니다.
IParseDisplayName 표시 가능한 이름 문자열을 구문 분석하여 사용자 지정 모니커 구현을 위한 모니커로 변환합니다.
IPerPropertyBrowsing 개체에서 제공하는 속성 페이지의 정보를 검색합니다.
IPersist 시스템에 영구적으로 저장할 수 있는 개체의 CLSID를 제공합니다. 개체가 마샬링의 기본 구현에 사용되므로 클라이언트 프로세스에서 사용할 개체 처리기를 지정할 수 있습니다.
IPersistFile 스토리지 개체 또는 스트림이 아닌 디스크 파일에서 개체를 로드하거나 디스크 파일에 저장할 수 있도록 합니다.
IPersistPropertyBag IPropertyBag 및 IErrorlog를 사용하여 개별 속성 기반 지속성 메커니즘을 정의합니다.

IPersistStorage 컨테이너 애플리케이션이 포함된 개체 중 하나에 스토리지 개체를 전달하고 스토리지 개체를 로드하고 저장할 수 있도록 합니다.
IPersistStream 스토리지 요구 사항에 맞게 간단한 직렬 스트림을 사용하는 개체를 저장하고 로드할 수 있습니다.
IPersistStreamInit 초기화 메서드를 추가하는 IPersistStream을 대체합니다.
IPicture 그림 개체 및 해당 속성을 관리합니다. 그림 개체는 비트맵, 아이콘 및 메타파일의 언어 중립적 추상화를 제공합니다.
IPictureDisp Automation을 통해 그림 개체의 속성을 노출합니다. IPicture 메서드를 통해 사용할 수 있는 기능의 하위 집합을 제공합니다.
IPipeByte IPipeByte 인터페이스(objidl.h)는 8비트 너비의 바이트 형식의 데이터를 전송합니다.
IPipeByte IPipeByte(objidlbase.h) 인터페이스는 바이트 형식(너비 8비트)의 데이터를 전송합니다.
IPipeDouble IPipeDouble 인터페이스(objidl.h)는 너비가 64비트인 double 형식의 데이터를 전송합니다.
IPipeDouble IPipeDouble(objidlbase.h) 인터페이스는 이중 형식(너비 64비트)의 데이터를 전송합니다.
IPipeLong IPipeLong 인터페이스(objidl.h)는 너비가 32비트인 긴 정수 형식의 데이터를 전송합니다.
IPipeLong IPipeLong(objidlbase.h) 인터페이스는 긴 정수 형식(너비 32비트)의 데이터를 전송합니다.

IPointerInactive 개체가 대부분의 시간 동안 비활성 상태로 유지되지만 끌어서 놓기를 포함하여 마우스와의 상호 작용에 계속 참여할 수 있습니다.
IPrint 특히 일반 및 활성 문서에서 복합 문서를 사용하여 프로그래밍 방식 인쇄를 지원할 수 있습니다.
IProcessInitControl IProcessInitControl 인터페이스(objidl.h)는 프로세스 초기화 시간 제한 간격을 지정합니다.
IProcessInitControl IProcessInitControl(objidlbase.h) 인터페이스는 프로세스 초기화 시간 제한 간격을 지정합니다.
IProcessLock ISurrogateService에서 시간 제한으로 인해 프로세스가 종료되지 않도록 하는 데 사용됩니다.
IProgressNotify 애플리케이션 및 기타 개체가 다운로드 작업 진행 중 변경 내용에 대한 알림을 받을 수 있도록 합니다.
IPropertyNotifySink 송신 인터페이스로 IPropertyNotifySink를 지원하는 개체에서 속성 변경에 대한 알림을 수신하기 위해 싱크 개체에 의해 구현됩니다.
IPropertyPage 속성 시트 내의 특정 페이지를 관리하는 속성 페이지 개체의 주요 기능을 제공합니다.
IPropertyPage2 페이지에서 속성의 초기 선택을 지원하기 위한 IPropertyPage 확장입니다.
IPropertyPageSite 속성 페이지 사이트 개체의 주요 기능을 제공합니다.
IProvideClassInfo 형식 라이브러리에서 개체의 coclass 항목에 대한 형식 정보에 대한 액세스를 제공합니다.

IProvideClassInfo2 기본 이벤트 집합에 대한 개체의 나가는 인터페이스 IID를 더 빠르고 쉽게 검색할 수 있도록 하는 IProvideClassInfo 확장입니다.
IProvideMultipleClassInfo 동작을 결정하는 여러 코클래스가 있을 수 있는 구성 요소에서 형식 정보를 더 빠르고 쉽게 검색할 수 있도록 하는 IProvideClassInfo2 확장입니다.
IPSFactoryBuffer IPSFactoryBuffer 인터페이스(objidl.h)는 COM 개체 프록시 및 스텁을 만들기 위한 사용자 지정 메서드를 제공합니다. 이 인터페이스는 마샬링할 수 없습니다.
IPSFactoryBuffer IPSFactoryBuffer(objidlbase.h) 인터페이스는 COM 개체 프록시 및 스텁을 만들기 위한 사용자 지정 메서드를 제공합니다. 이 인터페이스는 마샬링할 수 없습니다.
IQuickActivate 컨트롤 및 컨테이너를 사용하여 컨트롤 로드에서 성능 병목 현상을 방지할 수 있습니다. 컨트롤과 컨테이너 간의 로드 시간 또는 초기화 시간 핸드셰이크를 단일 호출로 결합합니다.
IROTData 모니커에 의해 구현되어 ROT(실행 중인 개체 테이블)가 모니커를 서로 비교할 수 있도록 합니다.
IRpcChannelBuffer IRpcChannelBuffer 인터페이스(objidl.h)는 COM 클라이언트 프록시와 COM 서버 스텁 간에 데이터를 마샬링합니다.
IRpcChannelBuffer IRpcChannelBuffer(objidlbase.h) 인터페이스는 COM 클라이언트 프록시와 COM 서버 스텁 간에 데이터를 마샬링합니다.
IRpcOptions IRpcOptions 인터페이스(objidl.h)를 사용하면 호출자가 COM이 RPC(원격 프로시저 호출)를 처리하는 방법을 제어하는 다양한 속성의 값을 설정하거나 쿼리할 수 있습니다.
IRpcOptions IRpcOptions(objidlbase.h) 인터페이스를 사용하면 호출자가 COM이 RPC(원격 프로시저 호출)를 처리하는 방법을 제어하는 다양한 속성의 값을 설정하거나 쿼리할 수 있습니다.

IRpcProxyBuffer IRpcProxyBuffer(objidl.h)는 COM 구성 요소 간에 데이터를 마샬링하는 데 사용되는 RPC 프록시를 제어합니다.
IRpcProxyBuffer IRpcProxyBuffer(objidlbase.h) 인터페이스는 COM 구성 요소 간에 데이터를 마샬링하는 데 사용되는 RPC 프록시를 제어합니다.
IRpcStubBuffer IRpcStubBuffer 인터페이스(objidl.h)는 COM 구성 요소 간에 데이터를 마샬링하는 데 사용되는 RPC 스텁을 제어합니다.
IRpcStubBuffer IRpcStubBuffer(objidlbase.h) 인터페이스는 COM 구성 요소 간에 데이터를 마샬링하는 데 사용되는 RPC 스텁을 제어합니다.
IRunnableObject 컨테이너가 포함된 개체의 실행을 제어할 수 있도록 합니다.
IRunningObjectTable 각 워크스테이션에서 전역적으로 액세스할 수 있는 조회 테이블인 ROT(실행 중인 개체 테이블)에 대한 액세스를 관리합니다.
IServerSecurity IServerSecurity(objidl.h) 인터페이스는 서버에서 클라이언트를 인증하고 클라이언트의 가장을 관리하는 데 사용됩니다.
IServerSecurity IServerSecurity(objidlbase.h) 인터페이스는 서버에서 클라이언트를 인증하고 클라이언트의 가장을 관리하는 데 사용됩니다.
IServiceProvider IServiceProvider 인터페이스는 GUID 식별 서비스를 찾는 일반 액세스 메커니즘을 제공합니다.
ISimpleFrameSite 다른 중첩된 컨트롤에 대한 간단한 컨테이너 역할을 하는 간단한 프레임 컨트롤을 제공합니다.

ISpecifyPropertyPages 개체가 속성 페이지를 지원한다는 것을 나타냅니다.
IStdMarshalInfo IStdMarshalInfo(objidl.h) 인터페이스는 표준 마샬링 중에 대상 프로세스에서 사용할 처리기를 식별하는 CLSID를 검색합니다.
IStdMarshalInfo IStdMarshalInfo(objidlbase.h) 인터페이스는 표준 마샬링 중에 대상 프로세스에서 사용할 처리기를 식별하는 CLSID를 검색합니다.
ISurrogate ISurrogate(objidl.h) 인터페이스는 새 DLL 서버를 기존 서로게이트에 동적으로 로드하고 더 이상 필요하지 않은 경우 서로게이트를 해제하는 데 사용됩니다.
ISurrogate ISurrogate(objidlbase.h) 인터페이스는 새 DLL 서버를 기존 서로게이트에 동적으로 로드하고 더 이상 필요하지 않은 경우 서로게이트를 해제하는 데 사용됩니다.
ISurrogateService COM+ 애플리케이션을 초기화, 시작 및 릴리스하는 데 사용됩니다. 카탈로그를 새로 고치고 프로세스를 종료할 수도 있습니다.
ISynchronize ISynchronize(objidl.h) 인터페이스는 이벤트 발생에 대한 개체 간의 비동기 통신을 제공합니다.
ISynchronize ISynchronize(objidlbase.h) 인터페이스는 이벤트 발생에 대한 개체 간의 비동기 통신을 제공합니다.
ISynchronizeContainer ISynchronizeContainer(objidl.h) 인터페이스는 서명되지 않은 동기화 개체 그룹을 관리합니다.
ISynchronizeContainer ISynchronizeContainer(objidlbase.h) 인터페이스는 서명되지 않은 동기화 개체 그룹을 관리합니다.
ISynchronizeEvent ISynchronizeEvent(objidl.h) 인터페이스는 동기화 개체에 이벤트 핸들을 할당합니다.

ISynchronizeEvent ISynchronizeEvent(objidlbase.h) 인터페이스는 동기화 개체에 이벤트 핸들을 할당합니다.
ISynchronizeHandle ISynchronizeHandle(objidl.h) 인터페이스는 동기화 개체와 연결된 핸들을 검색합니다.
ISynchronizeHandle ISynchronizeHandle(objidlbase.h) 인터페이스는 동기화 개체와 연결된 핸들을 검색합니다.
IUnknown 클라이언트가 QueryInterface 메서드를 통해 지정된 개체의 다른 인터페이스에 대한 포인터를 가져와 AddRef 및 Release 메서드를 통해 개체의 존재를 관리할 수 있도록 합니다.
IVBFormat Visual Basic 컨테이너의 컨트롤에 대한 서식 지원을 제공합니다.
IVBGetControl Visual Basic 컨테이너의 모든 컨트롤에 대한 액세스를 제공합니다.
IViewObject 호출자에게 데이터 개체를 전달하지 않고 개체가 직접 표시되도록 설정합니다. 또한 이 인터페이스는 뷰 개체의 변경 내용을 호출자에게 알 수 있도록 advise 싱크와의 연결을 만들고 관리할 수 있습니다.
IViewObject2 개체의 지정된 뷰에 대한 드로잉 크기를 반환하는 IViewObject 인터페이스의 확장입니다. IOleObject::GetExtent 대신 이 메서드를 호출하여 개체가 아직 실행되고 있지 않은 경우 개체가 실행되지 않도록 방지할 수 있습니다.
IViewObjectEx 직사각형이 아닌 개체 및 투명 개체에 대한 향상된 깜박임 없는 그리기 지원, 직사각형이 아닌 개체에 대한 적중 테스트 및 컨트롤 크기 조정을 지원하기 위해 IViewObject2에서 파생된 확장입니다.

구조체

ACTRL_ACCESS_ENTRY_LISTA 액세스 항목 목록을 포함합니다. (ANSI)
ACTRL_ACCESS_ENTRY_LISTW 액세스 항목 목록을 포함합니다. (유니코드)
ACTRL_ACCESS_ENTRYA 지정된 트러스티에 대한 액세스 제어 정보를 포함합니다. 이 구조체는 ACE에 저장된 액세스 제어 정보에 해당하는 정보를 저장합니다. (ANSI)
ACTRL_ACCESS_ENTRYW 지정된 트러스티에 대한 액세스 제어 정보를 포함합니다. 이 구조체는 ACE에 저장된 액세스 제어 정보에 해당하는 정보를 저장합니다. (유니코드)
ACTRL_ACCESSA 개체 및 해당 속성에 대한 액세스 제어 목록의 배열을 포함합니다. (ANSI)
ACTRL_ACCESSW 개체 및 해당 속성에 대한 액세스 제어 목록의 배열을 포함합니다. (유니코드)
ACTRL_PROPERTY_ENTRYA 개체 또는 개체의 지정된 속성에 대한 액세스 제어 항목 목록을 포함합니다. (ANSI)
ACTRL_PROPERTY_ENTRYW 개체 또는 개체의 지정된 속성에 대한 액세스 제어 항목 목록을 포함합니다. (유니코드)
BIND_OPTS 모니커 바인딩 작업 중에 사용되는 매개 변수를 포함합니다.
CADWORD 속성에 대해 미리 정의된 문자열 중 하나에 해당하는 값을 가져오는 데 사용할 수 있는 값의 계산된 배열을 지정합니다.
CALLFRAME_MARSHALCONTEXT 마샬링을 수행해야 하는 컨텍스트에 대한 정보를 제공합니다.

CALLFRAMEINFO 호출 프레임의 메서드와 같은 호출 프레임에 대한 정보를 제공합니다(매개 변수가 in, out 또는 in/out 매개 변수가 있는 경우, [in], [out] 또는 [in, out] 인터페이스 수, 인터페이스 ID, 인터페이스의 메서드 수 및 이 메서드의 매개 변수 수입니다.
CALLFRAMEPARAMINFO 스택의 매개 변수에 대한 정보를 제공합니다.
CALPOLESTR 속성이 허용할 수 있는 미리 정의된 문자열을 지정하는 데 사용되는 개수 지정된 문자열 배열을 지정합니다.
CATEGORYINFO 구성 요소 범주를 설명합니다.
CAUUID 개체가 표시하려는 속성 페이지에 대한 CLSID 배열을 수신하는 데 사용되는 UUID 또는 GUID 형식의 계산된 배열을 지정합니다.
COAUTHIDENTITY 사용자 이름 및 암호를 포함합니다.
COAUTHINFO 클라이언트 컴퓨터에서 서버 컴퓨터로 원격 활성화 요청을 수행하는 동안 사용되는 인증 설정을 포함합니다.
CONNECTDATA 지정된 연결점에 대해 존재하는 연결을 설명합니다.
CONTROLINFO 컨트롤의 키보드 니모닉 및 키보드 동작을 설명하는 매개 변수를 포함합니다. 구조체는 IOleControl::GetControlInfo 메서드 중에 채워집니다.
COSERVERINFO COSERVERINFO(objidl.h) 구조는 활성화 함수에 대한 원격 컴퓨터 리소스를 식별합니다.
COSERVERINFO COSERVERINFO(objidlbase.h) 구조는 활성화 함수에 대한 원격 컴퓨터 리소스를 식별합니다.

CSPLATFORM 운영 체제 플랫폼 및 프로세서 아키텍처를 포함합니다.
DVASPECTINFO GDI를 보다 효율적으로 사용하여 비활성 개체의 렌더링을 최적화하기 위해 IViewObject::D raw 메서드에서 사용하는 정보를 포함합니다.
DVEXTENTINFO IViewObjectEx::GetNaturalExtent에 사용되는 크기 조정 데이터를 나타냅니다.
DVTARGETDEVICE 데이터를 구성하는 대상 디바이스에 대한 정보를 지정합니다. DVTARGETDEVICE에는 Windows 대상 디바이스에 대한 충분한 정보가 포함되어 있으므로 CreateDC 함수를 사용하여 HDC(디바이스 컨텍스트)에 대한 핸들을 만들 수 있습니다.
FONTDESC OleCreateFontIndirect 함수를 통해 글꼴 개체를 만드는 데 사용되는 매개 변수를 포함합니다.
Formatetc 일반화된 클립보드 형식을 나타냅니다.
INTERFACEINFO 들어오는 호출에 대한 정보를 포함합니다.
LICINFO 라이선스를 지원하는 클래스 팩터리의 라이선스 동작을 설명하는 매개 변수를 포함합니다. 구조체는 IClassFactory2::GetLicInfo 메서드를 호출하여 채워집니다.
MULTI_QI MULTI_QI(objidl.h) 구조체는 여러 인터페이스에 대한 쿼리의 인터페이스를 나타냅니다.
MULTI_QI MULTI_QI(objidlbase.h) 구조체는 여러 인터페이스에 대한 쿼리의 인터페이스를 나타냅니다.
OBJECTDESCRIPTOR CF_OBJECTDESRIPTOR 및 CF_LINKSRCDESCRIPTOR 파일 형식에 사용되는 데이터 구조를 나타냅니다.

OCPFIPARAMS OleCreatePropertyFrameIndirect 함수를 통해 속성 시트 대화 상자를 호출하는 데 사용되는 매개 변수를 포함합니다.
OLECMD IOleCommandTarget::QueryStatus 호출을 통해 OLECMDF 열거형의 명령 플래그를 명령 식별자와 연결합니다.
OLECMDTEXT 단일 명령 식별자에 대한 텍스트 이름 또는 상태 문자열을 지정합니다.
OLEINPLACEFRAMEINFO 현재 위치 세션 동안 컨테이너에서 지원하는 가속기에 대한 정보를 포함합니다. 구조체는 IOleInPlaceSite::GetWindowContext 메서드 및 OleTranslateAccelerator 함수에 사용됩니다.
OLEMENUGROUPWIDTHS 현재 위치 편집 세션 동안 컨테이너와 개체 서버 간에 공유되는 메뉴의 6개 메뉴 그룹 각각에 있는 메뉴 항목 수를 나타냅니다. 공유 메뉴를 빌드하기 위한 메커니즘입니다.
OLEUIBUSYA OLE 사용자 인터페이스 라이브러리가 사용 중인 대화 상자를 초기화하는 데 사용하는 정보와 대화 상자가 해제될 때 라이브러리가 정보를 반환할 공간을 포함합니다. (ANSI)
OLEUIBUSYW OLE 사용자 인터페이스 라이브러리가 사용 중인 대화 상자를 초기화하는 데 사용하는 정보와 대화 상자가 해제될 때 라이브러리가 정보를 반환할 공간을 포함합니다. (유니코드)
OLEUICHANGEICONA OLE 사용자 인터페이스 라이브러리가 변경 아이콘 대화 상자를 초기화하는 데 사용하는 정보가 포함되어 있으며, 대화 상자가 해제될 때 라이브러리가 정보를 반환할 수 있는 공간이 포함되어 있습니다. (ANSI)
OLEUICHANGEICONW OLE 사용자 인터페이스 라이브러리가 변경 아이콘 대화 상자를 초기화하는 데 사용하는 정보가 포함되어 있으며, 대화 상자가 해제될 때 라이브러리가 정보를 반환할 수 있는 공간이 포함되어 있습니다. (유니코드)
OLEUICHANGESOURCEA 표준 원본 변경 대화 상자를 초기화하는 데 사용되는 정보를 포함합니다. (ANSI)

OLEUICHANGESOURCEW 표준 원본 변경 대화 상자를 초기화하는 데 사용되는 정보를 포함합니다. (유니코드)
OLEUICONVERTA OLE 사용자 인터페이스 라이브러리가 변환 대화 상자를 초기화하는 데 사용하는 정보와 대화 상자가 해제될 때 라이브러리가 정보를 반환할 수 있는 공간이 포함되어 있습니다. (ANSI)
OLEUICONVERTW OLE 사용자 인터페이스 라이브러리가 변환 대화 상자를 초기화하는 데 사용하는 정보와 대화 상자가 해제될 때 라이브러리가 정보를 반환할 수 있는 공간이 포함되어 있습니다. (유니코드)
OLEUIEDITLINKSA OLE 사용자 인터페이스 라이브러리가 링크 편집 대화 상자를 초기화하는 데 사용하는 정보를 포함하고, 대화 상자가 해제될 때 라이브러리가 정보를 반환할 공간을 포함합니다. (ANSI)
OLEUIEDITLINKSW OLE 사용자 인터페이스 라이브러리가 링크 편집 대화 상자를 초기화하는 데 사용하는 정보를 포함하고, 대화 상자가 해제될 때 라이브러리가 정보를 반환할 공간을 포함합니다. (유니코드)
OLEUIGNRLPROPSA 개체 속성 대화 상자의 일반 탭을 초기화합니다. (ANSI)
OLEUIGNRLPROPSW 개체 속성 대화 상자의 일반 탭을 초기화합니다. (유니코드)
OLEUIINSERTOBJECTA OLE 사용자 인터페이스 라이브러리가 개체 삽입 대화 상자를 초기화하는 데 사용하는 정보와 대화 상자가 해제될 때 라이브러리가 정보를 반환할 수 있는 공간이 포함되어 있습니다. (ANSI)
OLEUIINSERTOBJECTW OLE 사용자 인터페이스 라이브러리가 개체 삽입 대화 상자를 초기화하는 데 사용하는 정보와 대화 상자가 해제될 때 라이브러리가 정보를 반환할 수 있는 공간이 포함되어 있습니다. (유니코드)
OLEUILINKPROPSA 개체 속성 대화 상자의 링크 탭을 초기화하는 데 사용되는 정보를 포함합니다. (ANSI)
OLEUILINKPROPSW 개체 속성 대화 상자의 링크 탭을 초기화하는 데 사용되는 정보를 포함합니다. (유니코드)

OLEUIOBJECTPROPSA 표준 개체 속성 대화 상자를 초기화하는 데 사용되는 정보를 포함합니다. (ANSI)
OLEUIOBJECTPROPSW 표준 개체 속성 대화 상자를 초기화하는 데 사용되는 정보를 포함합니다. (유니코드)
OLEUIPASTEENTRYA 특수 붙여넣기 대화 상자의 OLEUIPASTESPECIAL 구조에 지정할 항목의 배열입니다. (ANSI)
OLEUIPASTEENTRYW 특수 붙여넣기 대화 상자의 OLEUIPASTESPECIAL 구조에 지정할 항목의 배열입니다. (유니코드)
OLEUIPASTESPECIALA OLE 사용자 인터페이스 라이브러리가 특수 붙여넣기 대화 상자를 초기화하는 데 사용하는 정보와 대화 상자가 해제될 때 라이브러리가 정보를 반환할 수 있는 공간이 포함되어 있습니다. (ANSI)
OLEUIPASTESPECIALW OLE 사용자 인터페이스 라이브러리가 특수 붙여넣기 대화 상자를 초기화하는 데 사용하는 정보와 대화 상자가 해제될 때 라이브러리가 정보를 반환할 수 있는 공간이 포함되어 있습니다. (유니코드)
OLEUIVIEWPROPSA 개체 속성 대화 상자의 보기 탭을 초기화하는 데 사용되는 정보를 포함합니다. (ANSI)
OLEUIVIEWPROPSW 개체 속성 대화 상자의 보기 탭을 초기화하는 데 사용되는 정보를 포함합니다. (유니코드)
OLEVERB 개체가 지원하는 동사를 정의합니다. IOleObject::EnumVerbs 메서드는 개체에 대해 이러한 구조를 열거할 수 있는 열거자를 만들고 열거자의 IEnumOLEVERB에 대한 포인터를 제공합니다.
Pagerange 페이지 범위를 지정합니다.
PAGESET 인쇄할 하나 이상의 페이지 범위를 식별하고 필요에 따라 짝수 또는 홀수 페이지만 페이지 세트의 일부로 식별합니다.

PICTDESC OleCreatePictureIndirect 함수를 통해 그림 개체를 만드는 매개 변수를 포함합니다.
Pointf HIMETRIC으로 표현되는 부동 소수점 및 제어 단위로 표현되는 컨테이너 단위 간에 변환하는 데 사용되는 정보를 포함합니다.
PROPPAGEINFO 속성 프레임에 대한 속성 페이지를 설명하는 데 사용되는 매개 변수를 포함합니다. 속성 페이지는 IPropertyPage::GetPageInfo 메서드에서 호출자 제공 구조를 채웁니다.
QACONTAINER IQuickActivate::QuickActivate에 대한 컨테이너 정보를 지정합니다.
QACONTROL IQuickActivate::QuickActivate에 대한 컨트롤 정보를 지정합니다.
QUERYCONTEXT 클래스 구현을 조회하는 데 사용되는 특성 목록을 포함합니다.
RPCOLEMESSAGE RPCOLEMESSAGE(objidl.h) 구조체에는 COM 구성 요소 간의 호출 인수 및 반환 값 마샬링이 포함됩니다.
RPCOLEMESSAGE RPCOLEMESSAGE(objidlbase.h) 구조체에는 COM 구성 요소 간의 호출 인수 및 반환 값 마샬링이 포함됩니다.
SECURITY_ATTRIBUTES SECURITY_ATTRIBUTES 구조체는 개체에 대한 보안 설명자를 포함하고 이 구조를 지정하여 검색된 핸들을 상속할 수 있는지 여부를 지정합니다.
SOLE_AUTHENTICATION_INFO SOLE_AUTHENTICATION_INFO(objidl.h) 구조는 인증 서비스, 권한 부여 서비스 및 지정된 서비스에 대한 정보를 식별합니다.

SOLE_AUTHENTICATION_INFO SOLE_AUTHENTICATION_INFO(objidlbase.h) 구조는 지정된 인증 서비스에 대한 인증 서비스, 권한 부여 서비스 및 인증 정보를 식별합니다.
SOLE_AUTHENTICATION_LIST SOLE_AUTHENTICATION_LIST(objidl.h) 구조는 각 인증 서비스에 사용할 기본 인증 정보를 나타냅니다.
SOLE_AUTHENTICATION_LIST SOLE_AUTHENTICATION_LIST(objidlbase.h) 구조는 각 인증 서비스에 사용할 기본 인증 정보를 나타냅니다.
SOLE_AUTHENTICATION_SERVICE SOLE_AUTHENTICATION_SERVICE(objidl.h) 구조는 서버가 클라이언트와 통신하는 데 사용할 인증 서비스를 식별합니다.
SOLE_AUTHENTICATION_SERVICE SOLE_AUTHENTICATION_SERVICE(objidlbase.h) 구조는 서버가 클라이언트와 통신하는 데 사용할 인증 서비스를 식별합니다.
STATDATA 각 권고 연결을 지정하는 데 사용되는 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

컴퍼지션 스왑 체인

아티클2023. 08. 15.

컴퍼지션 스왑 체인 기술의 개요입니다.

컴퍼지션 스왑 체인을 사용하여 개발하려면 다음 헤더가 필요합니다.

- presentation.h
- presentationtypes.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 컴퍼지션 스왑 체인

열거형

CompositionFrameInstanceKind 컴퍼지션 프레임에서 콘텐츠가 사용된 방법을 나타내는 상수를 정의합니다.
PresentStatisticsKind 통계가 적용되는 프레젠테이션 표면의 종류를 나타내는 상수를 정의합니다.
PresentStatus 현재 상태 지정하는 상수를 정의합니다.

Functions

CancelPresentsFrom 아직 표시되지 않고 현재 ID가 적어도 에 전달된 것으로 지정된 이전에 발급된 `presentIdToCancelFrom` 프레젠테이션과 진행 중인 프레젠테이션을 취소합니다.
CreatePresentationFactory 프레젠테이션 팩터리를 만듭니다.
CreatePresentationManager

프레젠테이션 관리자를 만듭니다.
CreatePresentationSurface 시각적 트리에서 호스트되고 단일 전면 버퍼를 할당할 수 있는 콘텐츠 조각에 대한 프레젠테이션 화면을 만듭니다.
DECLSPEC_XFGVIRT DECLSPEC_XFGVIRT 함수는 presentation.h 헤더와 함께 컴퍼지션 스왑 체인으로 개발하는 데 사용됩니다.
DECLSPEC_XFGVIRT DECLSPEC_XFGVIRT 함수는 컴퍼지션 스왑 체인에서 사용하는 presentationtypes.h 헤더의 일부입니다.
EnablePresentStatisticsKind 지정된 현재 통계 종류를 사용하거나 사용하지 않도록 설정합니다.
ForceVSyncInterrupt 가 표시되면 GPU에서 항상 VSync 인터럽트 실행 여부를 나타내는 값을 설정합니다.
GetAvailableEvent 버퍼를 사용할 수 있을 때 신호를 전송하는 이벤트에 대한 핸들을 가져옵니다.
GetCompositionFrameId 다른 API에서 더 자세한 정보를 가져오는 데 사용할 컴퍼지션 프레임의 ID를 가져옵니다.
GetCompositionFrameId 현재 가 처리, 건너뛰기 또는 취소된 컴퍼지션 프레임의 ID를 가져옵니다.
GetContentTag 통계가 보고되는 콘텐츠의 태그를 가져옵니다. (ICompositionFramePresentStatistics::GetContentTag)
GetContentTag 통계가 보고되는 콘텐츠의 태그를 가져옵니다. (IIndependentFlipFramePresentStatistics::GetContentTag)
GetDisplayedTime

현재가 표시된 시간을 가져옵니다.
GetDisplayInstanceArray 표시 인스턴스 및 해당 수의 배열을 가져옵니다.
GetKind 이 데이터가 해당하는 특정 종류의 현재 통계를 가져옵니다.
GetLostEvent 프레젠테이션 관리자가 복구할 수 없는 오류에 도달하면 신호를 표시하는 이벤트에 대한 핸들을 가져옵니다.
GetNextPresentId 다음 현재에 대한 식별자를 가져옵니다. 모든 동기화 펜스는 해당 현재를 참조할 때 이 값에 신호를 보냅니다.
GetNextPresentStatistics 큐의 다음 현재 통계 항목을 가져옵니다.
GetOutputAdapterLUID 이 독립 대칭 이동이 발생한 디스플레이 어댑터를 참조하는 LUID(로컬 고유 ID)를 가져옵니다.
GetOutputVidPnSourceId 디스플레이 어댑터의 비디오 현재 원본을 식별하는 정수 를 가져옵니다.
GetPresentDuration 현재가 표시된 실제 시간을 가져옵니다.
GetPresentId 이 통계가 해당하는 현재 의 식별자를 가져옵니다.
GetPresentRetiringFence 현재가 사용 중지될 때 신호를 표시하는 펜스를 가져옵니다.
GetPresentStatisticsAvailableEvent 현재 통계를 보고할 수 있을 때 신호를 전송하는 이벤트에 대한 핸들을 가져옵니다.
GetPresentStatus

프레임의 상태 가져옵니다.
IIndependentFlipFramePresentStatistics_GetDisplayedTime_Proxy IIndependentFlipFramePresentStatistics_GetDisplayedTime_Proxy 함수(presentation.h)는 현재가 표시된 시간을 가져옵니다.
IIndependentFlipFramePresentStatistics_GetDisplayedTime_Stub IIndependentFlipFramePresentStatistics_GetDisplayedTime_Stub 함수는 현재가 표시된 시간을 가져옵니다.
IIndependentFlipFramePresentStatistics_GetOutputAdapterLUID_Proxy IIndependentFlipFramePresentStatistics_GetOutputAdapterLUID_Proxy 함수는 독립 대칭 이동이 발생한 디스플레이 어댑터를 나타냅니다.
IIndependentFlipFramePresentStatistics_GetOutputAdapterLUID_Stub IIndependentFlipFramePresentStatistics_GetOutputAdapterLUID_Stub 함수는 독립 대칭 이동이 발생한 디스플레이 어댑터를 나타냅니다.
IIndependentFlipFramePresentStatistics_GetPresentDuration_Proxy IIndependentFlipFramePresentStatistics_GetPresentDuration_Proxy 함수는 현재가 표시된 실제 시간을 가져옵니다.
IIndependentFlipFramePresentStatistics_GetPresentDuration_Stub IIndependentFlipFramePresentStatistics_GetPresentDuration_Stub 함수는 현재가 표시된 실제 시간을 가져옵니다.
IsAvailable 생산자가 이 버퍼를 사용할 수 있는지 여부를 나타내는 값을 가져옵니다.
IsPresentationSupported 백업 D3D 디바이스에서 모든 종류의 프레젠테이션(독립적인 대칭 이동 포함 또는 제외)이 지원되는지 여부를 나타내는 값을 가져옵니다.
IsPresentationSupportedWithIndependentFlip 지원 D3D 디바이스에서 독립 대칭 이동 사용 프레젠테이션이 지원되는지 여부를 나타내는 값을 가져옵니다.
있음 이 프레젠테이션 관리자를 표시합니다.

RestrictToOutput 프레젠테이션을 특정 디스플레이 어댑터 출력으로 제한합니다.
SetAlphaMode 프레젠테이션 화면의 투명도 동작을 설정합니다.
SetBuffer 이 프레젠테이션 화면과 연결된 프레젠테이션 버퍼를 설정합니다.
SetColorSpace 프레젠테이션 화면에서 사용하는 색 공간의 유형을 설정합니다.
SetDisableReadback 버퍼 읽기를 사용하지 않도록 설정하거나 다시 사용하도록 설정하는 플래그를 설정합니다.
SetLetterboxingMargins 각 레터박스 영역에서 사용하는 크기를 시각적 공간으로 설정합니다.
SetPreferredPresentDuration 기본 설정 프레임 기간을 설정합니다.
SetSourceRect 원본 프레젠테이션 버퍼의 영역을 샘플로 설정합니다.
SetTag 이 콘텐츠와 연결할 사용자 정의 태그를 설정합니다. 이 태그는 통계에서 콘텐츠를 참조하는 방법입니다.
SetTargetTime 다음 현재에 대한 대상 시간을 설정합니다.
SetTransform 원본 버퍼 영역에 적용된 변환을 설정하여 버퍼가 표시되는 화면 영역을 정의합니다.

인터페이스

ICompositionFramePresentStatistics 시스템이 특정 현재 내에 특정 콘텐츠의 instance 표시하는 방법을 설명합니다.
IIndependentFlipFramePresentStatistics 시스템이 특정 독립 플립 프레젠테이션 내에서 특정 콘텐츠의 instance 표시하는 방법을 설명합니다.
IPresentationBuffer 프레젠테이션 관리자에 등록된 프레젠테이션 버퍼에 대해 설명합니다.
IPresentationContent 모든 프레젠테이션 콘텐츠의 기본 표현을 설명합니다.
IPresentationFactory 이 인터페이스는 프레젠테이션에 대한 시스템 지원을 쿼리하고 프레젠테이션 관리자를 만드는 데 사용됩니다.
IPresentationManager 프레젠테이션을 관리하기 위한 메서드를 정의합니다.
IPresentationSurface `IPresentationContent` 시스템 작성기와 공유하고 전역 시각적 트리에 콘텐츠로 표시할 수 있는 단일 바인딩된 버퍼가 있는 을 설명합니다.
IPresentStatistics 현재 통계 큐의 단일 항목을 나타냅니다.
IPresentStatusPresentStatistics DWM으로 구성된 프레임 중에 현재가 처리된 방법을 설명합니다.

구조체

CompositionFrameDisplayInstance 단일 출력에 표시되는 콘텐츠의 단일 instance 나타냅니다.

PresentationTransform 3-by-2 행렬에 의해 정의된 임의의 아핀 2D 변환을 나타냅니다. (PresentationTransform)
SystemInterruptTime 시스템이 마지막으로 시작된 이후의 시간(100ns 간격)을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

압축 API

아티클2023. 08. 22.

압축 API 기술 개요입니다.

압축 API를 개발하려면 다음 헤더가 필요합니다.

- compressapi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 압축 API

열거형

COMPRESS_INFORMATION_CLASS 이 열거형의 값은 설정되거나 검색되는 정보 클래스의 유형을 식별합니다.

Functions

CloseCompressor 를 호출하여 열린 COMPRESSOR_HANDLE 닫습니다.
CloseDecompressor 를 호출하여 열린 DECOMPRESSOR_HANDLE 닫습니다.
압축 정보 블록을 가져와서 압축합니다.
CreateCompressor 새 COMPRESSOR_HANDLE 생성합니다.
CreateDecompressor 새 DECOMPRESSOR_HANDLE 생성합니다.

압축 압축된 정보 블록을 가져와 압축을 풉니다.
QueryCompressorInformation 압축기에서 특정 압축 알고리즘에 대한 정보를 쿼리합니다.
QueryDecompressorInformation 이 함수를 사용하여 특정 압축 알고리즘에 대한 정보를 쿼리합니다.
ResetCompressor 새 스트림의 압축을 위해 압축기를 준비합니다.
ResetDecompressor 새 스트림의 압축 해제를 위해 압축 해제기를 준비합니다.
SetCompressorInformation 특정 압축 알고리즘에 대한 압축기 정보를 설정합니다.
SetDecompressorInformation 특정 압축 알고리즘에 대한 압축 해제기에서 정보를 설정합니다.

구조체

COMPRESS_ALLOCATION_ROUTINES 선택적 메모리 할당 및 할당 취소 루틴을 포함하는 구조체입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

핵심 오디오 API

아티클2023. 06. 23.

핵심 오디오 API 기술 개요입니다.

핵심 오디오 API를 개발하려면 다음 헤더가 필요합니다.

- audioclient.h
- audioendpoints.h
- audioenginebaseapo.h
- audioengineendpoint.h
- audiopolicy.h
- audiosessiontypes.h
- audiostatemonitorapi.h
- devicetopology.h
- endpointvolume.h
- mmdeviceapi.h
- spatialaudioclient.h
- spatialaudiohrtf.h
- spatialaudiometadata.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 핵심 오디오 API

열거형

_AUDCLNT_BUFFERFLAGS _AUDCLNT_BUFFERFLAGS 열거형은 오디오 엔드포인트 버퍼의 상태 나타내는 플래그를 정의합니다.
AUDCLNT_SHAREMODE AUDCLNT_SHAREMODE 열거형은 오디오 스트림이 공유 모드에서 실행되는지 아니면 배타적 모드에서 실행되는지 여부를 나타내는 상수를 정의합니다.
AUDCLNT_STREAMOPTIONS 오디오 스트림의 특성을 설명하는 값을 정의합니다.

AUDIO_DUCKING_OPTIONS 오디오 더킹 옵션을 지정합니다. IAudioClientDuckingControl::SetDuckingOptionsForCurrentStream을 호출할 때 이 열거형의 값을 사용합니다.
AUDIO_EFFECT_STATE 오디오 효과의 상태를 지정합니다.
AUDIO_STREAM_CATEGORY 오디오 스트림의 범주를 지정합니다.
AUDIOCLIENT_ACTIVATION_TYPE ActivateAudioInterfaceAsync 호출에 전달된 AUDIOCLIENT_ACTIVATION_PARAMS 구조체의 활성화 유형을 지정합니다.
AudioObjectType ISpatialAudioObject의 형식을 지정합니다.
AudioSessionState AudioSessionState 열거형은 오디오 세션의 현재 상태를 나타내는 상수를 정의합니다.
AudioStateMonitorSoundLevel
ConnectorType ConnectorType 열거형은 커넥터가 속한 연결 유형을 나타냅니다.
Dataflow DataFlow 열거형은 커넥터를 통해 오디오 스트림의 데이터 흐름 방향을 나타냅니다.
EDataFlow EDataFlow 열거형은 오디오 엔드포인트 디바이스와 애플리케이션 간에 오디오 데이터가 흐르는 방향을 나타내는 상수를 정의합니다.
EndpointFormFactor EndpointFormFactor 열거형은 오디오 엔드포인트 디바이스의 일반적인 물리적 특성을 나타내는 상수를 정의합니다.

ERole ERole 열거형은 시스템이 오디오 엔드포인트 디바이스에 할당한 역할을 나타내는 상수를 정의합니다.
KSJACK_SINK_CONNECTIONTYPE KSJACK_SINK_CONNECTIONTYPE 열거형은 연결 유형을 지정하는 상수를 정의합니다. 이러한 값은 오디오 잭 싱크에 대한 정보를 저장하는 KSJACK_SINK_INFORMATION 구조에 사용됩니다.
PartType PartType 열거형은 디바이스 토폴로지의 부분이 커넥터인지 하위 단위인지를 나타내는 상수를 정의합니다.
PROCESS_LOOPBACK_MODE ActivateAudioInterfaceAsync 호출에 전달된 AUDIOCLIENT_ACTIVATION_PARAMS 구조체에 대한 루프백 모드를 지정합니다.
SPATIAL_AUDIO_STREAM_OPTIONS ActivateSpatialAudioStream 호출에 대한 오디오 스트림 옵션을 지정합니다.
SpatialAudioHrtfDirectivityType ISpatialAudioObjectForHrtf에서 소리를 내보낸 셰이프를 지정합니다.
SpatialAudioHrtfDistanceDecayType ISpatialAudioObjectForHrtf의 위치에서 수신기 위치까지의 거리에 적용된 감쇠 유형을 지정합니다.
SpatialAudioHrtfEnvironmentType ISpatialAudioObjectForHrtf에 대해 오디오가 처리될 때 시뮬레이션되는 음향 환경의 유형을 지정합니다.
SpatialAudioMetadataCopyMode ISpatialAudioMetadataCopier::CopyMetadataForFrames를 호출할 때 사용되는 복사 모드를 지정합니다.
SpatialAudioMetadataWriterOverflowMode ISpatialAudioMetadataWriter가 클라이언트를 초기화할 때 지정한 것보다 더 많은 항목을 메타데이터 버퍼에 쓰려고 할 때 원하는 동작을 지정합니다.

Functions

활성화 Activate 메서드는 지정된 인터페이스를 사용하여 COM 개체를 만듭니다.
활성화 Activate 메서드는 커넥터 또는 하위 단위에서 함수별 인터페이스를 활성화합니다.
ActivateAudioInterfaceAsync Windows 스토어 앱이 WASAPI 제품군의 기존 COM(구성 요소 개체 모델) 인터페이스에 액세스할 수 있도록 합니다.
ActivateCompleted WASAPI 인터페이스의 활성화가 완료되었으며 결과를 사용할 수 있음을 나타냅니다.
ActivateSpatialAudioMetadataCopier 한 ISpatialAudioMetadataItems 개체에서 다른 ISpatialAudioMetadataItems 개체로 공간 오디오 메타데이터 항목을 복사하기 위한 ISpatialAudioMetadataWriter 개체를 만듭니다.
ActivateSpatialAudioMetadataItems 공간 오디오 메타데이터 항목을 저장하기 위한 ISpatialAudioMetadataItems 개체를 만듭니다.
ActivateSpatialAudioMetadataReader ISpatialAudioMetadataItems 개체에서 공간 오디오 메타데이터 항목을 읽기 위한 ISpatialAudioMetadataWriter 개체를 만듭니다.
ActivateSpatialAudioMetadataWriter ISpatialAudioMetadataItems 개체에 공간 오디오 메타데이터 항목을 쓰기 위한 ISpatialAudioMetadataWriter 개체를 만듭니다.
ActivateSpatialAudioObject 오디오 렌더링을 위해 ISpatialAudioObject를 활성화합니다.
ActivateSpatialAudioObjectForHrtf 오디오 렌더링을 위해 ISpatialAudioObjectForHrtf를 활성화합니다.

ActivateSpatialAudioObjectForMetadataCommands 렌더링을 위해 ISpatialAudioObjectForMetadataCommands를 활성화합니다.
ActivateSpatialAudioObjectForMetadataItems 렌더링을 위해 ISpatialAudioObjectForMetadataItems를 활성화합니다.
ActivateSpatialAudioStream 공간 오디오 스트림 활성화 구조 중 하나를 사용하여 공간 오디오 스트림을 활성화하고 초기화합니다.
AttachToBuffer ISpatialAudioMetadataItems 개체의 스토리지에 대해 호출자가 제공한 메모리를 연결합니다.
AttachToPopulatedBuffer 이전에 채워진 버퍼를 ISpatialAudioMetadataItems 개체의 스토리지에 연결합니다. 버퍼에 이미 있는 메타데이터 항목은 보존됩니다.
AudioStateMonitorCallback 시스템이 IAudioStreamStateMonitor에서 모니터링하는 오디오 스트림의 사운드 수준을 변경할 때 발생합니다.
BeginUpdatingAudioObjects 처리를 위해 오디오 개체 데이터를 제출할 수 있고 ISpatialAudioObject 상태를 수정할 수 있는 상태로 시스템을 배치합니다.
닫기 SpatialAudioMetadataItems 개체에 필요한 작업을 완료하고 개체를 해제합니다. (ISpatialAudioMetadataCopier.Close)
닫기 SpatialAudioMetadataItems 개체에 필요한 작업을 완료하고 개체를 해제합니다. (ISpatialAudioMetadataReader.Close)
닫기 메타데이터 버퍼에 필요한 작업을 완료하고 지정된 ISpatialAudioMetadataItems 개체를 해제합니다.

ConnectTo ConnectTo 메서드는 이 커넥터를 다른 디바이스 토폴로지 개체의 커넥터에 연결합니다.
CopyMetadataForFrames Open 메서드에 제공된 원본 ISpatialAudioMetadataItems의 메타데이터 항목을 dstMetadataItems 매개 변수로 지정된 대상 ISpatialAudioMetadataItems 개체에 복사합니다.
CreateCaptureAudioStateMonitor 캡처 스트림에 대한 IAudioStateMonitor의 새 instance 만듭니다.
CreateCaptureAudioStateMonitorForCategory 지정된 오디오 범주를 사용하여 캡처 스트림에 대한 IAudioStateMonitor의 새 instance 만듭니다.
CreateCaptureAudioStateMonitorForCategoryAndDeviceId 지정된 오디오 범주 및 오디오 디바이스 ID를 사용하여 캡처 스트림에 대한 IAudioStateMonitor의 새 instance 만듭니다.
CreateCaptureAudioStateMonitorForCategoryAndDeviceRole 지정된 오디오 범주 및 오디오 디바이스 역할을 사용하여 캡처 스트림에 대한 IAudioStateMonitor의 새 instance 만듭니다.
CreateRenderAudioStateMonitor 렌더링 스트림에 대한 IAudioStateMonitor의 새 instance 만듭니다.
CreateRenderAudioStateMonitorForCategory 지정된 오디오 범주를 사용하여 렌더링 스트림에 대한 IAudioStateMonitor의 새 instance 만듭니다.
CreateRenderAudioStateMonitorForCategoryAndDeviceId 지정된 오디오 범주 및 오디오 디바이스 ID를 사용하여 렌더링 스트림에 대한 IAudioStateMonitor의 새 instance 만듭니다.
CreateRenderAudioStateMonitorForCategoryAndDeviceRole 지정된 오디오 범주 및 오디오 디바이스 역할을 사용하여 렌더링 스트림에 대한 IAudioStateMonitor의 새 instance 만듭니다.
DetachBuffer 버퍼를 분리합니다. 메모리는 한 번에 하나의 메타데이터 항목에만 연결할 수 있습니다.

연결 끊기 Disconnect 메서드는 다른 커넥터에서 이 커넥터의 연결을 끊습니다.
EndUpdatingAudioObjects ActivateSpatialAudioObject를 사용하여 활성화된 공간 오디오 개체에 대한 오디오 데이터 제공을 완료했음을 시스템에 알 수 있습니다.
EnumAudioEndpoints EnumAudioEndpoints 메서드는 지정된 조건을 충족하는 오디오 엔드포인트 디바이스 컬렉션을 생성합니다.
EnumPartsIncoming EnumPartsIncoming 메서드는 들어오는 모든 부분, 즉 이 부분에서 업스트림 데이터 경로에 있는 파트의 목록을 가져옵니다.
EnumPartsOutgoing EnumPartsOutgoing 메서드는 이 부분의 다운스트림에 있는 데이터 경로에 있는 모든 나가는 부분의 목록을 검색합니다.
Get4BRange Get4BRange 메서드는 디바이스별 속성 값의 4바이트 범위를 가져옵니다.
GetActivateResult ActivateAudioInterfaceAsync 함수를 호출하는 애플리케이션에서 시작한 WASAPI 인터페이스의 비동기 활성화 결과를 가져옵니다.
GetAllVolumes GetAllVolumes 메서드는 오디오 스트림의 모든 채널에 대한 볼륨 수준을 검색합니다.
GetAllVolumes GetAllVolumes 메서드는 오디오 세션의 모든 채널에 대한 볼륨 수준을 검색합니다.
GetAudioEffects 연결된 오디오 스트림에 대한 오디오 효과의 현재 목록을 가져옵니다.
GetAudioObjectType ISpatialAudioObject로 표시되는 오디오 개체의 형식을 지정하는 값을 가져옵니다.

GetAudioSessionControl GetAudioSessionControl 메서드는 오디오 세션 컨트롤을 검색합니다.
GetAvailableDynamicObjectCount 현재 사용할 수 있는 동적 공간 오디오 개체의 수를 가져옵니다.
GetAvailableOffloadConnectorCount GetAvailableOffloadConnectorCount 메서드는 하드웨어 오디오 엔진에서 오프로드된 스트림을 처리할 수 있는 사용 가능한 엔드포인트 수를 검색합니다.
GetBuffer ISpatialAudioObject에 대한 오디오 데이터를 제공하는 데 사용되는 버퍼를 가져옵니다.
GetBuffer 캡처 엔드포인트 버퍼에서 사용 가능한 다음 데이터 패킷에 대한 포인터를 검색합니다.
GetBuffer 호출자가 데이터 패킷을 작성할 수 있는 렌더링 엔드포인트 버퍼에서 사용 가능한 다음 공간에 대한 포인터를 검색합니다.
GetBufferSize GetBufferSize 메서드는 엔드포인트 버퍼의 크기(최대 용량)를 검색합니다.
GetBufferSizeLimits GetBufferSizeLimits 메서드는 하드웨어 오디오 엔진의 버퍼 크기 제한을 100나노초 단위로 반환합니다.
GetChannelConfig GetChannelConfig 메서드는 채널 구성 컨트롤에서 현재 채널 구성 마스크를 가져옵니다.
GetChannelCount GetChannelCount 메서드는 오디오 스트림의 채널 수를 가져옵니다. (IAudioPeakMeter.GetChannelCount)
GetChannelCount GetChannelCount 메서드는 오디오 스트림의 채널 수를 가져옵니다. (IPerChannelDbLevel.GetChannelCount)

GetChannelCount GetChannelCount 메서드는 오디오 스트림의 채널 수를 검색합니다.
GetChannelCount GetChannelCount 메서드는 오디오 세션의 스트림 형식으로 채널 수를 검색합니다.
GetChannelCount GetChannelCount 메서드는 오디오 엔드포인트 디바이스를 입력하거나 종료하는 오디오 스트림의 채널 수를 가져옵니다.
GetChannelsPeakValues GetChannelsPeakValues 메서드는 오디오 스트림의 모든 채널에 대한 최대 샘플 값을 가져옵니다.
GetChannelVolume GetChannelVolume 메서드는 오디오 스트림에서 지정된 채널의 볼륨 수준을 검색합니다.
GetChannelVolume GetChannelVolume 메서드는 오디오 세션에서 지정된 채널의 볼륨 수준을 검색합니다.
GetChannelVolumeLevel GetChannelVolumeLevel 메서드는 오디오 엔드포인트 디바이스를 입력하거나 종료하는 오디오 스트림에서 지정된 채널의 볼륨 수준(데시벨)을 가져옵니다.
GetChannelVolumeLevelScalar GetChannelVolumeLevelScalar 메서드는 오디오 엔드포인트 디바이스를 입력하거나 종료하는 오디오 스트림의 지정된 채널의 정규화된 오디오 테이퍼 볼륨 수준을 가져옵니다.
GetChannelVolumes GetChannelVolumes 메서드는 오프로드된 스트림의 다양한 오디오 채널에 대한 볼륨 수준을 검색합니다.
GetCharacteristics GetCharacteristics 메서드는 나중에 사용할 수 있도록 예약되어 있습니다.
GetConnectedTo GetConnectedTo 메서드는 이 커넥터가 연결된 커넥터를 가져옵니다.

GetConnector GetConnector 메서드는 커넥터 번호로 지정된 커넥터를 가져옵니다.
GetConnectorCount GetConnectorCount 메서드는 디바이스 토폴로지 개체의 커넥터 수를 가져옵니다.
GetConnectorIdConnectedTo GetConnectorIdConnectedTo 메서드는 이 커넥터가 연결된 커넥터의 전역 ID(있는 경우)를 가져옵니다.
GetControlInterface GetControlInterface 메서드는 이 파트가 지원하는 경우 지정된 컨트롤 인터페이스에 대한 참조를 가져옵니다.
GetControlInterfaceCount GetControlInterfaceCount 메서드는 이 파트에서 지원하는 컨트롤 인터페이스 수를 가져옵니다.
GetCount 목록에서 지원되는 오디오 형식의 수를 가져옵니다.
GetCount GetCount 메서드는 디바이스 컬렉션의 디바이스 수를 검색합니다.
GetCount GetCount 메서드는 파트 목록의 파트 수를 가져옵니다.
GetCount GetCount 메서드는 오디오 디바이스에서 열려 있는 총 오디오 세션 수를 가져옵니다.
GetCurrentPadding GetCurrentPadding 메서드는 엔드포인트 버퍼의 패딩 프레임 수를 검색합니다.
GetCurrentSharedModeEnginePeriod 오디오 엔진의 현재 형식 및 주기를 반환합니다.

GetDataFlow GetDataFlow 메서드는 오디오 엔드포인트 디바이스가 렌더링 디바이스인지 또는 캡처 디바이스인지를 나타냅니다.
GetDataFlow GetDataFlow 메서드는 이 커넥터를 통해 데이터 흐름의 방향을 가져옵니다.
GetDefaultAudioEndpoint GetDefaultAudioEndpoint 메서드는 지정된 데이터 흐름 방향 및 역할에 대한 기본 오디오 엔드포인트를 검색합니다.
GetDevice GetDevice 메서드는 엔드포인트 ID 문자열로 식별되는 오디오 엔드포인트 디바이스를 검색합니다.
GetDeviceId GetDeviceId 메서드는 디바이스 토폴로지 개체로 표시되는 디바이스의 디바이스 식별자를 가져옵니다.
GetDeviceIdConnectedTo GetDeviceIdConnectedTo 메서드는 이 커넥터가 연결된 오디오 디바이스(있는 경우)의 디바이스 식별자를 가져옵니다.
GetDevicePeriod GetDevicePeriod 메서드는 엔드포인트 버퍼의 데이터에 대한 오디오 엔진의 연속 처리 패스를 구분하는 주기적 간격의 길이를 검색합니다.
GetDevicePosition GetDevicePosition 메서드는 하드웨어에서 직접 프레임에서 현재 디바이스 위치를 가져옵니다.
GetDevicePreferredFormat GetDevicePreferredFormat 메서드는 연결에 대한 기본 오디오 스트림 형식을 가져옵니다.
GetDisplayName GetDisplayName 메서드는 오디오 세션의 표시 이름을 검색합니다.
GetEnabled GetEnabled 메서드는 AGC의 현재 상태(사용 또는 사용 안 함)를 가져옵니다.

GetEnabled GetEnabled 메서드는 음의도 컨트롤의 현재 상태(사용 또는 사용 안 함)를 가져옵니다.
GetEngineFormat GetEngineFormat 메서드는 오프로드된 오디오 스트림의 현재 데이터 형식을 검색합니다.
Getformat 목록에서 지정된 인덱스가 있는 형식을 가져옵니다. 형식은 중요도 순서대로 나열됩니다. 가장 바람직한 형식은 목록의 첫 번째 형식입니다.
GetFrameCount 유효한 항목 오프셋을 정의하는 ISpatialAudioMetadataItems의 총 프레임 수를 가져옵니다.
GetFrequency GetFrequency 메서드는 디바이스 빈도를 가져옵니다.
GetGfxState GetGfxState 메서드는 오프로드된 오디오 스트림의 GFX 상태를 검색합니다.
GetGlobalId GetGlobalId 메서드는 이 부분의 전역 ID를 가져옵니다.
GetGroupingParam GetGroupingParam 메서드는 오디오 세션의 그룹화 매개 변수를 검색합니다.
GetIconPath GetIconPath 메서드는 오디오 세션의 표시 아이콘 경로를 검색합니다.
GetId GetId 메서드는 오디오 엔드포인트 디바이스를 식별하는 엔드포인트 ID 문자열을 검색합니다.
GetIID GetIID 메서드는 파트의 함수별 컨트롤 인터페이스의 인터페이스 ID를 가져옵니다.
GetInfo 유효한 항목 오프셋을 정의하는 ISpatialAudioMetadataItems의 총 프레임 수를 가져옵니다.

GetItemCount ISpatialAudioMetadataItems에 의해 저장된 현재 항목 수입니다.
GetJackCount GetJackCount 메서드는 오디오 엔드포인트 디바이스에 연결하는 데 필요한 잭 수를 가져옵니다.
GetJackCount GetJackCount 메서드는 엔드포인트 디바이스에 연결하는 데 필요한 커넥터의 잭 수를 가져옵니다.
GetJackDescription GetJackDescription 메서드는 오디오 잭에 대한 설명을 가져옵니다.
GetJackDescription2 GetJackDescription2 메서드는 지정된 오디오 잭에 대한 설명을 가져옵니다.
GetJackSinkInformation GetJackSinkInformation 메서드는 지정된 잭에 대한 싱크 정보를 검색합니다.
GetLevel GetLevel 메서드는 해당 채널의 최고 수준을 이전에 읽은 이후 지정된 채널에 대해 기록된 피크 미터의 최고 수준을 가져옵니다.
GetLevel GetLevel 메서드는 지정된 채널의 볼륨 수준(데시벨)을 가져옵니다.
GetLevelRange GetLevelRange 메서드는 지정된 채널의 볼륨 수준 범위를 데시벨 단위로 가져옵니다.
GetLocalEffectsState GetLocalEffectsState 메서드는 현재 오프로드된 오디오 스트림에 적용되는 로컬 효과 상태를 검색합니다.
GetLocalId GetLocalId 메서드는 이 파트의 로컬 ID를 가져옵니다.
GetMasterVolume GetMasterVolume 메서드는 오디오 세션에 대한 클라이언트 볼륨 수준을 검색합니다.

GetMasterVolumeLevel GetMasterVolumeLevel 메서드는 오디오 엔드포인트 디바이스에 들어가거나 나가는 오디오 스트림의 master 볼륨 수준(데시벨)을 가져옵니다.
GetMasterVolumeLevelScalar GetMasterVolumeLevelScalar 메서드는 오디오 엔드포인트 디바이스에 들어가거나 나가는 오디오 스트림의 master 볼륨 수준을 가져옵니다. 볼륨 수준은 0.0에서 1.0 범위의 정규화된 오디오 테이퍼 값으로 표현됩니다.
GetMaxDynamicObjectCount 공간 오디오 클라이언트의 최대 동적 오디오 개체 수를 가져옵니다.
GetMaxFrameCount 처리 패스당 가능한 최대 프레임 수를 가져옵니다. 이 메서드는 각 처리 패스에 대한 오디오 데이터를 전달하기 위해 할당해야 하는 원본 버퍼의 크기를 결정하는 데 사용할 수 있습니다.
GetMaxFrameCountForCategory 처리 단계당 지원되는 최대 프레임 수를 가져옵니다.
GetMaxItemCount 개체를 만들 때 정의된 ISpatialAudioMetadataItems에서 허용되는 최대 항목 수입니다.
GetMaxValueBufferLength ISpatialAudioMetadataItems의 메타데이터 형식으로 정의된 가장 큰 명령 값의 크기입니다.
GetMeterChannelCount 계량할 수 있는 오프로드된 스트림에서 사용 가능한 오디오 채널 수를 가져옵니다.
GetMeteringChannelCount GetMeteringChannelCount 메서드는 최대 미터로 모니터링되는 오디오 스트림의 채널 수를 가져옵니다.
GetMeteringData GetMeteringData 메서드는 오프로드된 스트림에서 사용 가능한 오디오 채널에 대한 일반 정보를 검색합니다.

GetMixFormat GetMixFormat 메서드는 오디오 엔진이 공유 모드 스트림의 내부 처리에 사용하는 스트림 형식을 검색합니다.
GetMute GetMute 메서드는 오프로드된 오디오 스트림의 음소거 상태 검색합니다.
GetMute GetMute 메서드는 음소거 컨트롤의 현재 상태(사용 또는 사용 안 함)를 가져옵니다.
GetMute GetMute 메서드는 오디오 세션의 현재 음소거 상태를 검색합니다.
GetMute GetMute 메서드는 오디오 엔드포인트 디바이스를 입력하거나 종료하는 오디오 스트림의 음소거 상태를 가져옵니다.
GetName GetName 메서드는 컨트롤 인터페이스가 캡슐화하는 오디오 함수의 이름을 가져옵니다.
GetName GetName 메서드는 이 파트의 이름을 가져옵니다.
GetNativeStaticObjectTypeMask 현재 렌더링 엔진에 네이티브 정적 스피커 베드 채널의 하위 집합을 나타내는 채널 마스크를 가져옵니다.
GetNextPacketSize GetNextPacketSize 메서드는 캡처 엔드포인트 버퍼의 다음 데이터 패킷에 있는 프레임 수를 검색합니다.
GetPart GetPart 메서드는 파트 목록에서 파트를 가져옵니다.
GetPartById GetPartById 메서드는 로컬 ID로 식별되는 파트를 가져옵니다.

GetPartType GetPartType 메서드는 이 파트의 파트 형식을 가져옵니다.
GetPeakValue GetPeakValue 메서드는 오디오 스트림의 채널에 대한 최대 샘플 값을 가져옵니다.
GetPosition GetPosition 메서드는 현재 디바이스 위치를 가져옵니다.
GetProcessId GetProcessId 메서드는 오디오 세션의 프로세스 식별자를 검색합니다.
GetSelection GetSelection 메서드는 현재 선택된 선택기 입력에 연결된 파트의 로컬 ID를 가져옵니다.
GetSelection GetSelection 메서드는 현재 선택된 선택기 출력에 연결된 파트의 로컬 ID를 가져옵니다.
GetService ISpatialAudioObjectRenderStream에서 추가 서비스를 가져옵니다.
GetService GetService 메서드는 오디오 클라이언트 개체에서 추가 서비스에 액세스합니다.
GetSession GetSession 메서드는 오디오 세션 번호로 지정된 오디오 세션을 가져옵니다.
GetSessionEnumerator GetSessionEnumerator 메서드는 오디오 세션 열거자 개체에 대한 포인터를 가져옵니다.
GetSessionIdentifier GetSessionIdentifier 메서드는 오디오 세션 식별자를 검색합니다.
GetSessionInstanceIdentifier GetSessionInstanceIdentifier 메서드는 오디오 세션 instance 식별자를 검색합니다.

GetSharedModeEnginePeriod 지정된 스트림 형식에 대해 엔진에서 지원하는 주기 범위 를 반환합니다.
GetSignalPath GetSignalPath 메서드는 경로가 있는 경우 두 부분을 연결하는 신호 경로의 파트 목록을 가져옵니다.
GetSimpleAudioVolume GetSimpleAudioVolume 메서드는 간단한 오디오 볼륨 컨트롤을 검색합니다.
GetSoundLevel IAudioStateMonitor와 연결된 오디오 스트림의 현재 사운드 수준을 가져옵니다.
GetSpatialAudioMetadataItems ISpatialAudioObjectForMetadataItems에 대한 메타데이터 항목을 저장하는 ISpatialAudioMetadataItems 개체에 대한 포인터를 가져옵니다.
GetSpatialAudioMetadataItemsBufferLength 지정된 수의 공간 오디오 메타데이터 항목을 저장하는 데 필요한 버퍼의 길이를 가져옵니다.
GetState GetState 메서드는 현재 디바이스 상태를 검색합니다.
GetState GetState 메서드는 오디오 세션의 현재 상태를 검색합니다.
GetStaticObjectPosition 지정된 정적 공간 오디오 채널의 3D 공간에서 위치를 가져옵니다.
GetStreamLatency GetStreamLatency 메서드는 현재 스트림의 최대 대기 시간을 검색하고 스트림이 초기화된 후 언제든지 호출할 수 있습니다.
GetSubType GetSubType 메서드는 이 파트의 파트 하위 형식을 가져옵니다.

GetSubunit GetSubunit 메서드는 하위 단위 번호로 지정된 하위 단위를 가져옵니다.
GetSubunitCount GetSubunitCount 메서드는 디바이스 토폴로지의 하위 단위 수를 가져옵니다.
GetSupportedAudioObjectFormatEnumerator 공간 오디오 개체에 대해 지원되는 모든 오디오 형식을 포함하는 IAudioFormatEnumerator를 가져옵니다. 목록의 첫 번째 항목은 가장 바람직한 형식을 나타냅니다.
GetTopologyObject GetTopologyObject 메서드는 이 부분을 포함하는 디바이스 토폴로지 개체의 IDeviceTopology 인터페이스에 대한 참조를 가져옵니다.
GetType GetType 메서드는 이 커넥터의 형식을 가져옵니다.
GetType GetType 메서드는 디바이스별 속성 값의 데이터 형식을 가져옵니다.
GetValue GetValue 메서드는 디바이스별 속성의 현재 값을 가져옵니다.
GetVolumeChannelCount GetVolumeChannelCount 메서드는 오프로드된 스트림에서 사용 가능한 오디오 채널 수를 검색합니다.
GetVolumeRange GetVolumeRange 메서드는 오디오 엔드포인트 디바이스에 들어오거나 나가는 오디오 스트림의 볼륨 범위(데시벨)를 가져옵니다.
GetVolumeRangeChannel GetVolumeRangeChannel 메서드는 지정된 채널의 볼륨 범위를 가져옵니다.
GetVolumeStepInfo GetVolumeStepInfo 메서드는 볼륨 범위의 현재 단계에 대한 정보를 가져옵니다.

초기화 Initialize 메서드는 오디오 스트림을 초기화합니다.
InitializeSharedAudioStream 지정된 주기를 사용하여 공유 스트림을 초기화합니다.
IsActive ISpatialAudioObject가 유효한지 여부를 나타내는 부울 값을 가져옵니다.
IsAudioObjectFormatSupported ISpatialAudioObjectRenderStream이 지정된 형식을 지원하는지 여부를 나타내는 값을 가져옵니다.
IsConnected IsConnected 메서드는 이 커넥터가 다른 커넥터에 연결되어 있는지 여부를 나타냅니다.
IsFormatSupported IsFormatSupported 메서드는 오디오 엔드포인트 디바이스가 지정된 오디오 스트림 형식을 지원하는지 여부를 나타냅니다.
IsFormatSupported IsFormatSupported 메서드는 오디오 엔드포인트 디바이스가 특정 스트림 형식을 지원하는지 여부를 나타냅니다.
IsLastBufferControlSupported 마지막 버퍼 컨트롤이 지원되는지를 나타냅니다.
IsOffloadCapable ISpatialAudioClient2가 만들어진 오디오 렌더링 엔드포인트가 하드웨어 오프로드 오디오 처리를 지원하는지 여부를 쿼리합니다.
IsOffloadCapable IsOffloadCapable 메서드는 스트림이 만들어진 엔드포인트가 오프로드된 오디오 스트림을 지원할 수 있는지 여부에 대한 정보를 검색합니다.
IsSpatialAudioStreamAvailable 성공하면 현재 활성 공간 렌더링 엔진이 지정된 공간 오디오 렌더링 스트림을 지원하는지 여부를 나타내는 값을 가져옵니다.

IsSystemSoundsSession IsSystemSoundsSession 메서드는 세션이 시스템 사운드 세션인지 여부를 나타냅니다.
Item Item 메서드는 디바이스 컬렉션에서 지정된 항목에 대한 포인터를 검색합니다.
OnAudioEffectsChanged 오디오 효과 목록이 변경되거나 효과 변경을 사용하도록 설정하는 데 필요한 리소스가 변경될 때 시스템에서 호출됩니다.
OnAvailableDynamicObjectCountChange ISpatialAudioObjectRenderStream의 렌더링 용량이 변경될 때 공간 오디오 클라이언트에 알리고, 변경이 수행되는 시간을 지정하고, 변경 후 사용할 수 있는 동적 오디오 개체 수를 지정합니다.
OnChannelVolumeChanged OnChannelVolumeChanged 메서드는 세션 서브믹스의 오디오 채널 볼륨 수준이 변경되었음을 클라이언트에 알릴 수 있습니다.
OnDefaultDeviceChanged OnDefaultDeviceChanged 메서드는 클라이언트에 특정 디바이스 역할에 대한 기본 오디오 엔드포인트 디바이스가 변경되었음을 알 수 있습니다.
OnDevice추가됨 OnDeviceAdded 메서드는 새 오디오 엔드포인트 디바이스가 추가되었음을 나타냅니다.
OnDeviceRemoved OnDeviceRemoved 메서드는 오디오 엔드포인트 디바이스가 제거되었음을 나타냅니다.
OnDeviceStateChanged OnDeviceStateChanged 메서드는 오디오 엔드포인트 디바이스의 상태가 변경되었음을 나타냅니다.
OnDisplayNameChanged OnDisplayNameChanged 메서드는 클라이언트에 세션의 표시 이름이 변경되었음을 알 수 있습니다.

OnGroupingParamChanged OnGroupingParamChanged 메서드는 세션의 그룹화 매개 변수가 변경되었음을 클라이언트에 알 수 있습니다.
OnIconPathChanged OnIconPathChanged 메서드는 세션의 표시 아이콘이 변경되었음을 클라이언트에 알 수 있습니다.
OnNotify OnNotify 메서드는 커넥터 또는 하위 단위의 상태 변경될 때 클라이언트에 알 수 있습니다.
OnNotify OnNotify 메서드는 오디오 엔드포인트 디바이스의 볼륨 수준 또는 음소거 상태가 변경되었음을 클라이언트에 알 수 있습니다.
OnPropertyValueChanged OnPropertyValueChanged 메서드는 오디오 엔드포인트 디바이스에 속하는 속성의 값이 변경되었음을 나타냅니다.
OnSessionCreated OnSessionCreated 메서드는 등록된 프로세스에 오디오 세션이 생성되었음을 알 수 있습니다.
OnSessionDisconnected OnSessionDisconnected 메서드는 클라이언트에 오디오 세션의 연결이 끊어졌는지 알 수 있습니다.
OnSimpleVolumeChanged OnSimpleVolumeChanged 메서드는 오디오 세션의 볼륨 수준 또는 음소거 상태가 변경되었음을 클라이언트에 알릴 수 있습니다.
OnStateChanged OnStateChanged 메서드는 세션의 스트림 작업 상태가 변경되었음을 클라이언트에 알깁니다.
OnVolumeDuckNotification OnVolumeDuckNotification 메서드는 보류 중인 시스템 더킹 이벤트에 대한 알림을 보냅니다.
OnVolumeUnduckNotification OnVolumeUnduckNotification 메서드는 보류 중인 시스템 언덕트 이벤트에 대한 알림을 보냅니다.

열기 복사할 ISpatialAudioMetadataItems 개체를 엽니다.
열기 읽을 ISpatialAudioMetadataItems 개체를 엽니다.
열기 쓰기를 위해 ISpatialAudioMetadataItems 개체를 엽니다.
OpenPropertyStore OpenPropertyStore 메서드는 디바이스의 속성 저장소에 대한 인터페이스를 검색합니다.
QueryHardwareSupport QueryHardwareSupport 메서드는 오디오 엔드포인트 디바이스에서 하드웨어 지원 함수를 쿼리합니다. (IAudioEndpointVolume.QueryHardwareSupport)
QueryHardwareSupport QueryHardwareSupport 메서드는 오디오 엔드포인트 디바이스에서 하드웨어 지원 함수를 쿼리합니다. (IAudioMeterInformation.QueryHardwareSupport)
ReadNextItem 읽는 메타데이터 항목의 명령 수와 샘플 오프셋을 가져옵니다.
ReadNextItemCommand 현재 항목에 대한 메타데이터 명령 및 값 데이터를 읽습니다.
RegisterAudioEffectsChangedNotificationCallback AudioEffectsChangedNotificationClient 인터페이스를 등록합니다.
RegisterAudioSessionNotification RegisterAudioSessionNotification 메서드는 스트림 상태의 변경 내용을 포함하여 세션 이벤트의 알림을 수신하도록 클라이언트를 등록합니다.
RegisterCallback 시스템이 IAudioStateMonitor에서 모니터링하는 오디오 스트림의 사운드 수준을 변경할 때 호출되는 AudioStateMonitorCallback의 구현을 등록합니다.

RegisterControlChangeCallback RegisterControlChangeCallback 메서드는 IControlChangeNotify 인터페이스를 등록합니다. 이 인터페이스는 클라이언트가 이 부분에서 상태 변경 내용에 대한 알림을 수신하기 위해 구현합니다.
RegisterControlChangeNotify RegisterControlChangeNotify 메서드는 클라이언트의 알림 콜백 인터페이스를 등록합니다.
RegisterDuckNotification RegisterDuckNotification 메서드는 더킹 알림을 받기 위해 애플리케이션을 세션 관리자에 등록합니다.
RegisterEndpointNotificationCallback RegisterEndpointNotificationCallback 메서드는 클라이언트의 알림 콜백 인터페이스를 등록합니다.
RegisterSessionNotification RegisterSessionNotification 메서드는 세션을 만들 때 알림을 받을 애플리케이션을 등록합니다.
ReleaseBuffer ReleaseBuffer 메서드는 버퍼를 해제합니다.
ReleaseBuffer ReleaseBuffer 메서드는 IAudioRenderClient::GetBuffer 메서드에 대한 이전 호출에서 획득한 버퍼 공간을 해제합니다.
ReleaseOutputDataPointerForLastBuffer 마지막 버퍼에 대한 출력 데이터 포인터를 해제합니다.
Reset 중지된 오디오 스트림을 다시 설정합니다.
Reset Reset 메서드는 오디오 스트림을 다시 설정합니다.
ResetToDefault 형식을 디바이스 제조업체에서 제공하는 기본 설정으로 다시 설정합니다.

SetAllVolumes SetAllVolumes 메서드는 오디오 스트림의 모든 채널에 대한 개별 볼륨 수준을 설정합니다.
SetAllVolumes SetAllVolumes 메서드는 오디오 세션의 모든 채널에 대한 개별 볼륨 수준을 설정합니다.
SetAudioStreamWindow 지정된 HWND 창 핸들을 오디오 스트림과 연결합니다.
SetChannelConfig SetChannelConfig 메서드는 채널 구성 컨트롤에서 채널 구성 마스크를 설정합니다.
SetChannelVolume SetChannelVolume 메서드는 오디오 스트림에서 지정된 채널의 볼륨 수준을 설정합니다.
SetChannelVolume SetChannelVolume 메서드는 오디오 세션에서 지정된 채널의 볼륨 수준을 설정합니다.
SetChannelVolumeLevel SetChannelVolumeLevel 메서드는 오디오 엔드포인트 디바이스를 입력하거나 종료하는 오디오 스트림의 지정된 채널의 볼륨 수준을 데시벨 단위로 설정합니다.
SetChannelVolumeLevelScalar SetChannelVolumeLevelScalar 메서드는 오디오 엔드포인트 디바이스를 입력하거나 종료하는 오디오 스트림에서 지정된 채널의 정규화된 오디오 테이퍼 볼륨 수준을 설정합니다.
SetChannelVolumes SetChannelVolumes 메서드는 오프로드된 스트림의 다양한 오디오 채널에 대한 볼륨 수준을 설정합니다.
SetClientProperties AudioClientProperties 구조를 채워 오디오 스트림의 속성을 설정합니다.
SetDirectivity ISpatialAudioObjectForHrtf에 대한 공간 오디오 지향성 모델을 설정합니다.

SetDisplayName SetDisplayName 메서드는 현재 세션에 표시 이름을 할당합니다.
SetDistanceDecay ISpatialAudioObjectForHrtf의 위치에서 수신기의 위치까지 멀리서 적용되는 감쇠 모델을 설정합니다.
SetDuckingOptionsForCurrentStream 오디오 렌더링 스트림에 대한 오디오 더킹 옵션을 설정합니다.
SetDuckingPreference SetDuckingPreference 메서드는 시스템에서 제공하는 기본 스트림 감쇠 환경(자동 더킹)을 사용하거나 사용하지 않도록 설정합니다.
SetEchoCancellationRenderEndpoint AEC(음향 에코 취소)의 참조 스트림으로 사용해야 하는 오디오 렌더링 엔드포인트를 설정합니다.
SetEnabled SetEnabled 메서드는 AGC를 사용하거나 사용하지 않도록 설정합니다.
SetEnabled SetEnabled 메서드는 음의도 컨트롤을 사용하거나 사용하지 않도록 설정합니다.
SetEndOfStream 개체를 비활성화하고 해당 리소스를 재사용할 수 있도록 ISpatialAudioObject에 대해 오디오 데이터의 마지막 블록이 제출되었음을 시스템에 지시합니다.
SetEngineDeviceFormat SetEngineDeviceFormat 메서드는 하드웨어 오디오 엔진의 파형 오디오 형식을 설정합니다.
SetEnvironment ISpatialAudioObjectForHrtf에 대해 오디오가 처리될 때 시뮬레이션되는 음향 환경의 유형을 설정합니다.
SetEventHandle SetEventHandle 메서드는 클라이언트에서 오디오 버퍼를 처리할 준비가 되면 시스템에서 신호하는 이벤트 핸들을 설정합니다.

SetGain ISpatialAudioObjectForHrtf에 대한 게인을 설정합니다.
SetGfxState SetGfxState 메서드는 오프로드된 오디오 스트림의 GFX 상태를 설정합니다.
SetGroupingParam SetGroupingParam 메서드는 세션을 세션 그룹화에 할당합니다.
SetIconPath SetIconPath 메서드는 현재 세션에 표시 아이콘을 할당합니다.
SetLevel SetLevel 메서드는 지정된 채널의 볼륨 수준(데시벨)을 설정합니다.
SetLevelAllChannels SetLevelAllChannels 메서드는 오디오 스트림에 있는 모든 채널의 볼륨 수준을 데시벨 단위로 설정합니다.
SetLevelUniform SetLevelUniform 메서드는 오디오 스트림의 모든 채널을 동일한 균일 볼륨 수준(데시벨)으로 설정합니다.
SetLocalEffectsState SetLocalEffectsState 메서드는 오프로드된 오디오 스트림에 적용할 로컬 효과 상태를 설정합니다.
SetMasterVolume SetMasterVolume 메서드는 오디오 세션의 master 볼륨 수준을 설정합니다.
SetMasterVolumeLevel SetMasterVolumeLevel 메서드는 오디오 엔드포인트 디바이스에 들어오거나 나가는 오디오 스트림의 master 볼륨 수준(데시벨)을 설정합니다.
SetMasterVolumeLevelScalar SetMasterVolumeLevelScalar 메서드는 오디오 엔드포인트 디바이스에 들어오거나 나가는 오디오 스트림의 master 볼륨 수준을 설정합니다. 볼륨 수준은 0.0에서 1.0까지의 범위에서 정규화된 오디오 테이퍼 값으로 표현됩니다.

SetMute SetMute 메서드는 오프로드된 오디오 스트림의 음소거 상태 설정합니다.
SetMute SetMute 메서드는 음소거 컨트롤을 사용하거나 사용하지 않도록 설정합니다.
SetMute SetMute 메서드는 오디오 세션의 음소거 상태를 설정합니다.
SetMute SetMute 메서드는 오디오 엔드포인트 디바이스로 들어오거나 나가는 오디오 스트림의 음소거 상태를 설정합니다.
SetOrientation ISpatialAudioObjectForHrtf 오디오 데이터가 렌더링될 수신기의 참조 프레임을 기준으로 방향을 3D 공간으로 설정합니다.
SetPosition ISpatialAudioObjectForHrtf 오디오 데이터가 렌더링될 수신기를 기준으로 3D 공간의 위치를 설정합니다.
SetPosition ISpatialAudioObject 오디오 데이터가 렌더링될 수신기를 기준으로 3D 공간의 위치를 설정합니다.
SetSampleRate SetSampleRate 메서드는 스트림의 샘플 속도를 설정합니다.
SetSelection SetSelection 메서드는 입력 선택기의 입력 중 하나를 선택합니다.
SetSelection SetSelection 메서드는 출력 선택기의 출력 중 하나를 선택합니다.
SetValue SetValue 메서드는 디바이스별 속성의 값을 설정합니다.

SetVolume ISpatialAudioObject에서 제공하는 오디오 데이터에 적용된 오디오 진폭 승수를 오디오 렌더링 엔진에 제출하기 전에 설정합니다.
시작 공간 오디오 스트림을 시작합니다.
시작 Start 메서드는 오디오 스트림을 시작합니다.
중지 실행 중인 오디오 스트림을 중지합니다.
중지 Stop 메서드는 오디오 스트림을 중지합니다.
UnregisterAudioEffectsChangedNotificationCallback IAudioEffectsChangedNotificationClient 인터페이스를 등록 취소합니다.
UnregisterAudioSessionNotification UnregisterAudioSessionNotification 메서드는 알림을 받기 위해 클라이언트에서 이전 등록을 삭제합니다.
UnregisterCallback 이전에 IAudioStateMonitor::RegisterCallback에 대한 호출로 등록된 AudioStateMonitorCallback의 등록을 취소합니다.
UnregisterControlChangeCallback UnregisterControlChangeCallback 메서드는 IPart::RegisterControlChangeCallback 메서드를 호출하여 클라이언트가 이전에 등록한 IControlChangeNotify 인터페이스의 등록을 제거합니다.
UnregisterControlChangeNotify UnregisterControlChangeNotify 메서드는 IAudioEndpointVolume::RegisterControlChangeNotify 메서드에 대한 이전 호출에서 클라이언트가 등록한 클라이언트의 알림 콜백 인터페이스 등록을 삭제합니다.

UnregisterDuckNotification UnregisterDuckNotification 메서드는 알림을 받기 위해 애플리케이션에서 이전 등록을 삭제합니다.
UnregisterEndpointNotificationCallback UnregisterEndpointNotificationCallback 메서드는 IMMDeviceEnumerator::RegisterEndpointNotificationCallback 메서드에 대한 이전 호출에서 클라이언트가 등록한 알림 인터페이스의 등록을 삭제합니다.
UnregisterSessionNotification UnregisterSessionNotification 메서드는 세션을 만들 때 알림을 받기 위해 등록을 삭제합니다.
VolumeStepDown VolumeStepDown 메서드는 오디오 엔드포인트 디바이스에 들어오거나 나가는 오디오 스트림의 볼륨 수준을 한 단계씩 감소합니다.
VolumeStepUp VolumeStepUp 메서드는 오디오 엔드포인트 디바이스에 들어오거나 나가는 오디오 스트림의 볼륨 수준을 한 단계씩 증가합니다.
WriteNextItem 지정된 오프셋에서 새 메타데이터 항목을 시작합니다.
WriteNextItemCommand 메타데이터 명령 및 값 데이터를 현재 항목에 씁니다.
WriteNextMetadataCommand 메타데이터 명령을 공간 오디오 개체에 씁니다. 각 명령은 처리 주기당 개체당 한 번만 추가할 수 있습니다.

인터페이스

IAcousticEchoCancellationControl 오디오 캡처 엔드포인트가 AEC(음향 에코 취소)를 지원하는지 여부를 결정하는 메커니즘을 제공하며, 이 경우 클라이언트가 참조 스트림으로 사용해야 하는 오디오 렌더링 엔드포인트를 설정할 수 있도록 합니다.
IActivateAudioInterfaceAsyncOperation WASAPI 인터페이스를 활성화하는 비동기 작업을 나타내며 활성화 결과를 검색하는 메서드를 제공합니다.
IActivateAudioInterfaceCompletionHandler WASAPI 인터페이스의 활성화가 완료되었음을 나타내는 콜백을 제공합니다.
IAudioAutoGainControl IAudioAutoGainControl 인터페이스는 하드웨어 AGC(자동 게인 제어)에 대한 액세스를 제공합니다.
IAudioBass IAudioBass 인터페이스는 하드웨어 베이스 수준 컨트롤에 대한 액세스를 제공합니다.
IAudioCaptureClient IAudioCaptureClient 인터페이스를 사용하면 클라이언트가 캡처 엔드포인트 버퍼에서 입력 데이터를 읽을 수 있습니다.
IAudioChannelConfig IAudioChannelConfig 인터페이스는 하드웨어 채널 구성 컨트롤에 대한 액세스를 제공합니다.
IAudioClient IAudioClient 인터페이스를 사용하면 클라이언트가 오디오 애플리케이션과 오디오 엔진(공유 모드 스트림용) 또는 오디오 엔드포인트 디바이스의 하드웨어 버퍼(전용 모드 스트림용) 간에 오디오 스트림을 만들고 초기화할 수 있습니다.
IAudioClient2 IAudioClient2 인터페이스는 IAudioClient 인터페이스에서 파생되며, WASAPI(Windows Audio Session API) 오디오 _opt 클라이언트가 오프로드, 쿼리 스트림 속성 및 오프로드를 처리하는 하드웨어에서 정보를 가져올 수 있도록 하는 추가 메서드 집합이 있습니다. 기본 엔드포인트가 하드웨어 오디오 엔진을 지원하고, 엔드포인트가 오디오 시스템에서 열거되고 검색되었으며, 엔드포인트에서 사용할 수 있는 오프로드 핀 인스턴스가 여전히 있는 경우 오디오 클라이언트가 오프로드된 스트림을 성공적으로 만들 수 있습니다.

IAudioClient3 IAudioClient3 인터페이스는 IAudioClient2 인터페이스에서 파생되며, WASAPI(Windows Audio Session API) 오디오 클라이언트가 오디오 엔진의 지원되는 주기 및 현재 주기를 쿼리하고 지정된 주기로 공유 오디오 스트림의 초기화를 요청할 수 있는 추가 메서드 집합이 있습니다.
IAudioClientDuckingControl 앱의 오디오 렌더링 스트림이 활성 상태일 때 시스템에서 다른 스트림의 오디오를 더킹하지 않도록 앱이 지정할 수 있도록 하는 SetDuckingOptionsForCurrentStream 메서드를 제공합니다.
IAudioClock IAudioClock 인터페이스를 사용하면 클라이언트가 스트림의 데이터 속도 및 스트림의 현재 위치를 모니터링할 수 있습니다.
IAudioClock2 IAudioClock2 인터페이스는 현재 디바이스 위치를 가져오는 데 사용됩니다.
IAudioClockAdjustment IAudioClockAdjustment 인터페이스는 스트림의 샘플 속도를 조정하는 데 사용됩니다.
IAudioEffectsChangedNotificationClient 콜백 인터페이스를 사용하면 오디오 효과 목록이 변경되거나 효과 변경을 사용하도록 설정하는 데 필요한 리소스가 변경될 때 애플리케이션에서 알림을 받을 수 있습니다.
IAudioEffectsManager 오디오 효과 파이프라인에 대한 관리 기능을 제공합니다.
IAudioEndpointFormatControl 현재 오디오 엔드포인트 디바이스 형식을 다시 설정하는 데 사용됩니다.
IAudioEndpointLastBufferControl 오프로드 스트림 클라이언트가 마지막 버퍼가 부분적으로만 전송되었음을 엔드포인트에 알릴 수 있도록 하는 기능을 제공합니다.
IAudioEndpointOffloadStreamMeter IAudioEndpointOffloadStreamMeter 인터페이스는 오프로드된 오디오 스트림의 오디오 채널에 대한 일반 정보를 검색합니다.

IAudioEndpointOffloadStreamMute IAudioEndpointOffloadStreamMute 인터페이스를 사용하면 클라이언트가 오프로드된 오디오 스트림의 음소거 상태 조작할 수 있습니다.
IAudioEndpointOffloadStreamVolume IAudioEndpointOffloadStreamVolume 인터페이스를 사용하면 클라이언트 애플리케이션이 오프로드된 오디오 스트림의 볼륨 수준을 조작할 수 있습니다.
IAudioEndpointVolume IAudioEndpointVolume 인터페이스는 오디오 엔드포인트 디바이스를 오가는 오디오 스트림의 볼륨 컨트롤을 나타냅니다.
IAudioEndpointVolumeCallback IAudioEndpointVolumeCallback 인터페이스는 볼륨 수준의 변경 내용과 오디오 엔드포인트 디바이스의 음소거 상태에 대한 알림을 제공합니다.
IAudioEndpointVolumeEx IAudioEndpointVolumeEx 인터페이스는 디바이스 엔드포인트를 오가는 오디오 스트림의 볼륨 컨트롤을 제공합니다.
IAudioFormatEnumerator 지원되는 오디오 형식 목록을 제공합니다. 가장 선호되는 형식은 목록의 첫 번째 형식입니다. ISpatialAudioClient::GetSupportedAudioObjectFormatEnumerator를 호출하여 이 인터페이스에 대한 참조를 가져옵니다.
IAudioInputSelector IAudioInputSelector 인터페이스는 하드웨어 멀티플렉서 컨트롤(입력 선택기)에 대한 액세스를 제공합니다.
IAudioLfxControl IAudioLfxControl 인터페이스를 사용하면 클라이언트가 오프로드된 오디오 스트림에서 로컬 효과를 적용하거나 제거할 수 있습니다.
IAudioLoudness IAudioLoudness 인터페이스는 "음음" 보정 컨트롤에 대한 액세스를 제공합니다.
IAudioMeterInformation IAudioMeterInformation 인터페이스는 오디오 엔드포인트 디바이스를 오가는 오디오 스트림의 최대 측정기를 나타냅니다.

IAudioMidrange IAudioMidrange 인터페이스는 하드웨어 미드레인지 수준 컨트롤에 대한 액세스를 제공합니다.
IAudioMute IAudioMute 인터페이스는 하드웨어 음소거 컨트롤에 대한 액세스를 제공합니다.
IAudioOutputSelector IAudioOutputSelector 인터페이스는 하드웨어 demultiplexer 컨트롤(출력 선택기)에 대한 액세스를 제공합니다.
IAudioPeakMeter IAudioPeakMeter 인터페이스는 하드웨어 피크 미터 컨트롤에 대한 액세스를 제공합니다.
IAudioRenderClient IAudioRenderClient 인터페이스를 사용하면 클라이언트가 렌더링 엔드포인트 버퍼에 출력 데이터를 쓸 수 있습니다.
IAudioSessionControl IAudioSessionControl 인터페이스를 사용하면 클라이언트가 오디오 세션에 대한 제어 매개 변수를 구성하고 세션의 이벤트를 모니터링할 수 있습니다.
IAudioSessionControl2 IAudioSessionControl2 인터페이스는 클라이언트에서 오디오 세션에 대한 정보를 가져오는 데 사용할 수 있습니다.
IAudioSessionEnumerator IAudioSessionEnumerator 인터페이스는 오디오 디바이스의 오디오 세션을 열거합니다.
IAudioSessionEvents IAudioSessionEvents 인터페이스는 볼륨 수준 변경, 표시 이름 및 세션 상태와 같은 세션 관련 이벤트에 대한 알림을 제공합니다.
IAudioSessionManager IAudioSessionManager 인터페이스를 사용하면 클라이언트가 교차 프로세스 및 프로세스별 오디오 세션 모두에 대한 세션 컨트롤 및 볼륨 컨트롤에 액세스할 수 있습니다.

IAudioSessionManager2 IAudioSessionManager2 인터페이스를 사용하면 애플리케이션이 오디오 디바이스에 대한 서브믹스를 관리할 수 있습니다.
IAudioSessionNotification IAudioSessionNotification 인터페이스는 오디오 세션을 만들 때 알림을 제공합니다.
IAudioStateMonitor 오디오 스트림의 소리 수준을 쿼리하고 소리 수준이 변경되면 알림을 수신하기 위한 API를 제공합니다.
IAudioStreamVolume IAudioStreamVolume 인터페이스를 사용하면 클라이언트가 오디오 스트림의 모든 채널에 대한 볼륨 수준을 제어하고 모니터링할 수 있습니다.
IAudioTreble IAudioTreble 인터페이스는 하드웨어 고음 수준 컨트롤에 대한 액세스를 제공합니다.
IAudioViewManagerService HWND를 오디오 스트림과 연결하기 위한 API를 제공합니다.
IAudioVolumeDuckNotification IAudioVolumeDuckNotification 인터페이스는 시스템에서 스트림 감쇠 변경에 대한 알림을 보내는 데 사용됩니다. 스트림 감쇠 또는 더킹은 Windows 7에서 도입된 기능으로, 시스템이 새 통신 스트림을 열 때 비 통신 스트림의 볼륨을 조정합니다. 이 기능에 대한 자세한 내용은 기본 더킹 환경을 참조하세요.
IAudioVolumeLevel IAudioVolumeLevel 인터페이스는 하드웨어 볼륨 제어에 대한 액세스를 제공합니다.
IChannelAudioVolume IChannelAudioVolume 인터페이스를 사용하면 클라이언트가 스트림이 속한 오디오 세션의 모든 채널에 대한 볼륨 수준을 제어하고 모니터링할 수 있습니다.
IConnector IConnector 인터페이스는 구성 요소 간의 연결 지점을 나타냅니다.

IControlChangeNotify IControlChangeNotify 인터페이스는 파트(커넥터 또는 하위 단위)의 상태 변경될 때 알림을 제공합니다.
IControlInterface IControlInterface 인터페이스는 디바이스 토폴로지의 파트(커넥터 또는 하위 단위)의 제어 인터페이스를 나타냅니다. 클라이언트는 IPart::GetControlInterface 메서드를 호출하여 파트의 IControlInterface 인터페이스에 대한 참조를 가져옵니다.
IDeviceSpecificProperty IDeviceSpecificProperty 인터페이스는 디바이스별 하드웨어 컨트롤의 제어 값에 대한 액세스를 제공합니다.
IDeviceTopology IDeviceTopology 인터페이스는 오디오 디바이스의 토폴로지 액세스 권한을 제공합니다.
IHardwareAudioEngineBase IHardwareAudioEngineBase 인터페이스는 하드웨어 오디오 엔진에 대한 정보를 구성하고 검색하는 데 사용할 오디오 스택의 오디오 엔드포인트에 의해 구현됩니다.
IKsFormatSupport IKsFormatSupport 인터페이스는 오디오 어댑터 디바이스와 시스템 메모리 간의 소프트웨어 구성 I/O 연결(일반적으로 DMA 채널)에서 지원하는 오디오 데이터 형식에 대한 정보를 제공합니다.
IKsJackDescription IKsJackDescription 인터페이스는 오디오 어댑터의 디바이스와 외부 또는 내부 엔드포인트 디바이스(예: 마이크 또는 CD 플레이어) 간의 물리적 연결을 제공하는 잭 또는 내부 커넥터에 대한 정보를 제공합니다.
IKsJackDescription2 IKsJackDescription2 인터페이스는 오디오 어댑터의 디바이스와 외부 또는 내부 엔드포인트 디바이스(예: 마이크 또는 CD 플레이어) 간의 물리적 연결을 제공하는 잭 또는 내부 커넥터에 대한 정보를 제공합니다.
IKsJackSinkInformation IKsJackSinkInformation 인터페이스는 잭이 하드웨어에서 지원되는 경우 잭 싱크 정보에 대한 액세스를 제공합니다.

IMMDevice IMMDevice 인터페이스는 멀티미디어 디바이스 리소스의 일반 기능을 캡슐화합니다.
IMMDeviceCollection IMMDeviceCollection 인터페이스는 멀티미디어 디바이스 리소스의 컬렉션을 나타냅니다.
IMMDeviceEnumerator IMMDeviceEnumerator 인터페이스는 멀티미디어 디바이스 리소스를 열거하는 메서드를 제공합니다.
IMMEndpoint IMMEndpoint 인터페이스는 오디오 엔드포인트 디바이스를 나타냅니다.
IMMNotificationClient IMMNotificationClient 인터페이스는 오디오 엔드포인트 디바이스가 추가 또는 제거되거나, 엔드포인트 디바이스의 상태 또는 속성이 변경되거나, 엔드포인트 디바이스에 할당된 기본 역할이 변경될 때 알림을 제공합니다.
IPart IPart 인터페이스는 디바이스 토폴로지의 파트(커넥터 또는 하위 단위)를 나타냅니다.
IPartsList IPartsList 인터페이스는 각각 커넥터 또는 하위 단위를 나타내는 IPart 인터페이스가 있는 개체인 파트 목록을 나타냅니다.
IPerChannelDbLevel IPerChannelDbLevel 인터페이스는 오디오 스트림 또는 오디오 스트림의 주파수 대역의 볼륨 수준(데시벨)에 대한 채널별 제어를 제공하는 제네릭 하위 단위 컨트롤 인터페이스를 나타냅니다.
ISimpleAudioVolume ISimpleAudioVolume 인터페이스를 사용하면 클라이언트가 오디오 세션의 master 볼륨 수준을 제어할 수 있습니다.
ISpatialAudioClient ISpatialAudioClient 인터페이스를 사용하면 클라이언트가 3D 공간의 위치에서 오디오를 내보내는 오디오 스트림을 만들 수 있습니다.

ISpatialAudioClient2 **ISpatialAudioClient2** 인터페이스는 ISpatialAudioClient에서 상속되며 큰 오디오 버퍼 오프로드에 대한 지원을 쿼리하는 메서드를 추가합니다.
ISpatialAudioMetadataClient ISpatialAudioMetadataItems, ISpatialAudioMetadataWriter, ISpatialAudioMetadataReader 및 ISpatialAudioMetadataCopier 개체를 만들기 위한 클래스 팩터리를 제공합니다.
ISpatialAudioMetadataCopier 원본 SpatialAudioMetadataItems에서 대상 SpatialAudioMetadataItems로 메타데이터 항목의 전체 또는 하위 집합을 복사하는 메서드를 제공합니다.
ISpatialAudioMetadataItems 공간 오디오 메타데이터 항목의 버퍼를 나타냅니다.
ISpatialAudioMetadataItemsBuffer 데이터의 현재 위치 스토리지를 위해 SpatialAudioMetadataItems에 버퍼를 연결하는 메서드를 제공합니다.
ISpatialAudioMetadataReader ISpatialAudioMetadataItems 개체에서 공간 오디오 메타데이터 항목 및 항목 명령 값 쌍을 추출하는 메서드를 제공합니다.
ISpatialAudioMetadataWriter 해당 오디오 프레임 범위 내에 배치된 공간 오디오 메타데이터 항목을 저장하는 메서드를 제공합니다.
ISpatialAudioObject 사용자를 기준으로 3D 공간의 위치에서 렌더링할 오디오 데이터를 제공하는 개체를 나타냅니다.
ISpatialAudioObjectBase 사용자를 기준으로 3D 공간의 위치에서 렌더링할 오디오 데이터를 제공하는 개체를 나타내는 기본 인터페이스입니다.
ISpatialAudioObjectForHrtf HRTF(헤드 상대 전송 함수)를 기준으로 3D 공간의 위치에서 렌더링할 오디오 데이터를 제공하는 개체를 나타냅니다.

ISpatialAudioObjectForMetadataCommands 공간 오디오에 대한 메타데이터 명령을 작성하는 데 사용됩니다.
ISpatialAudioObjectForMetadataItems 프레임 정확도 배치를 사용하여 버퍼당 여러 메타데이터 항목이 필요한 애플리케이션에 대한 공간 오디오 메타데이터를 작성하는 데 사용됩니다.
ISpatialAudioObjectRenderStream 스트림 시작, 중지 및 초기화를 포함하여 공간 오디오 개체 렌더링 스트림을 제어하는 메서드를 제공합니다.
ISpatialAudioObjectRenderStreamBase 스트림 시작, 중지 및 초기화를 포함하여 공간 오디오 개체 렌더링 스트림을 제어하는 메서드를 제공하는 기본 인터페이스입니다.
ISpatialAudioObjectRenderStreamForHrtf 스트림 시작, 중지 및 초기화를 포함하여 Hrtf 공간 오디오 개체 렌더링 스트림을 제어하는 메서드를 제공합니다.
ISpatialAudioObjectRenderStreamForMetadata 스트림 시작, 중지 및 초기화를 포함하여 메타데이터에 대한 공간 오디오 개체 렌더링 스트림을 제어하는 메서드를 제공합니다.
ISpatialAudioObjectRenderStreamNotify 공간 오디오 클라이언트가 ISpatialAudioObjectRenderStream의 상태 변경에 응답하도록 알림을 제공합니다.
ISubunit ISubunit 인터페이스는 클라이언트와 오디오 엔드포인트 디바이스 간의 데이터 경로에 있는 하드웨어 하위 단위(예: 볼륨 컨트롤)를 나타냅니다.

구조체

AUDIO_EFFECT 오디오 효과를 나타냅니다.

AUDIO_VOLUME_NOTIFICATION_DATA AUDIO_VOLUME_NOTIFICATION_DATA 구조는 오디오 엔드포인트 디바이스의 볼륨 수준 또는 음소거 상태의 변화를 설명합니다.
AUDIOCLIENT_ACTIVATION_PARAMS ActivateAudioInterfaceAsync 호출에 대한 활성화 매개 변수를 지정합니다.
AUDIOCLIENT_PROCESS_LOOPBACK_PARAMS 루프백 활성화가 요청되는 ActivateAudioInterfaceAsync 호출에 대한 매개 변수를 지정합니다.
AudioExtensionParams 이 구조체는 IShellPropSheetExt::AddPages를 통해 제어판 엔드포인트 확장 속성 페이지로 전달되며 엔드포인트 PropertyPages를 만드는 데 사용됩니다.
DIRECTX_AUDIO_ACTIVATION_PARAMS DIRECTX_AUDIO_ACTIVATION_PARAMS 구조체는 DirectSound 스트림의 초기화 매개 변수를 지정합니다.
KSJACK_DESCRIPTION KSJACK_DESCRIPTION 구조는 오디오 잭을 설명합니다.
KSJACK_DESCRIPTION2 KSJACK_DESCRIPTION2 구조는 오디오 잭을 설명합니다. 커넥터의 오디오 잭에 대한 설명을 얻으려면 IKsJackDescription2::GetJackDescription2를 호출합니다.
KSJACK_SINK_INFORMATION KSJACK_SINK_INFORMATION 구조체는 오디오 잭 싱크에 대한 정보를 저장합니다.
Luid LUID 구조체는 비디오 포트 식별자를 저장합니다. 이 구조체는 KSJACK_SINK_INFORMATION 구조체의 PortId 멤버에 저장됩니다.
SpatialAudioClientActivationParams 공간 오디오 렌더링 스트림에 대한 선택적 활성화 매개 변수를 나타냅니다. ISpatialAudioClient 인터페이스를 활성화할 때 이 구조를 ActivateAudioInterfaceAsync에 전달합니다.
SpatialAudioHrtfActivationParams ISpatialAudioRenderStreamForHrtf에 대한 활성화 매개 변수를 지정합니다.

SpatialAudioHrtfActivationParams2 공간 오디오 렌더링 스트림에 대한 활성화 매개 변수를 나타내며 스트림 옵션을 지정할 수 있는 기능으로 SpatialAudioHrtfActivationParams를 확장합니다.
SpatialAudioHrtfDirectivity ISpatialAudioObjectForHrtf에 대한 전방향 모델을 나타냅니다. 전방향 방출은 크기 조정 필드의 값을 기반으로 형식 필드에 지정된 직접성 모델과 선형으로 보간됩니다.
SpatialAudioHrtfDirectivityCardioid ISpatialAudioObjectForHrtf에 대한 심장 모양의 직접성 모델을 나타냅니다.
SpatialAudioHrtfDirectivityCone ISpatialAudioObjectForHrtf에 대한 원뿔 모양의 직접성 모델을 나타냅니다.
SpatialAudioHrtfDirectivityUnion ISpatialAudioObjectForHrtf에 대한 공간 오디오 직접성 모델을 정의합니다.
SpatialAudioHrtfDistanceDecay ISpatialAudioObjectForHrtf의 위치에서 수신기 위치까지의 거리에 적용되는 감쇠 모델을 나타냅니다.
SpatialAudioMetadataItemsInfo ISpatialAudioMetadataItems 개체에 대한 정보를 제공합니다. GetInfo를 호출하여 이 구조체의 복사본을 가져옵니다.
SpatialAudioObjectRenderStreamActivationParams 공간 오디오 렌더링 스트림에 대한 활성화 매개 변수를 나타냅니다. 스트림을 활성화할 때 이 구조를 ISpatialAudioClient::ActivateSpatialAudioStream에 전달합니다.
SpatialAudioObjectRenderStreamActivationParams2 공간 오디오 렌더링 스트림에 대한 활성화 매개 변수를 나타내며 스트림 옵션을 지정할 수 있는 기능으로 SpatialAudioObjectRenderStreamActivationParams를 확장합니다.
SpatialAudioObjectRenderStreamForMetadataActivationParams 메타데이터에 대한 공간 오디오 렌더링 스트림에 대한 활성화 매개 변수를 나타냅니다. 스트림을 활성화할 때 이 구조를 ISpatialAudioClient::ActivateSpatialAudioStream에 전달합니다.

SpatialAudioObjectRenderStreamForMetadataActivationParams2 스트림 옵션을 지정할 수 있는 기능으로 SpatialAudioObjectRenderStreamForMetadataActivationParams를 확장하는 메타데이터에 대한 공간 오디오 렌더링 스트림에 대한 활성화 매개 변수를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

데이터 액세스 및 스토리지

아티클2023. 03. 09.

데이터 액세스 및 스토리지 기술에 대한 개요입니다.

데이터 액세스 및 스토리지를 개발하려면 다음 헤더가 필요합니다.

- clfs.h
- clfsmgmt.h
- clfsmgmtw32.h
- clfsw32.h
- dskquota.h
- fileapi.h
- fileapifromapp.h
- ioringapi.h
- ktmtypes.h
- ktmw32.h
- lmshare.h
- lmstats.h
- lzexpand.h
- ntioring_x.h
- ntmsapi.h
- ntmsmli.h
- nvme.h
- txfw32.h
- winefs.h
- wofapi.h
- wow64apiset.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 이동식 저장소 관리자
- 일반 로그 파일 시스템
- 로컬 파일 시스템
- 커널 트랜잭션 관리자
- 네트워크 공유 관리

IOCTLs

FSCTL_CREATE_OR_GET_OBJECT_ID 지정된 파일 또는 디렉터리에 대한 개체 식별자를 검색합니다. 개체 식별자가 없으면 FSCTL_CREATE_OR_GET_OBJECT_ID를 사용하여 만듭니다.
FSCTL_CSV_CONTROL CSV 제어 작업의 결과를 검색합니다.
FSCTL_CSV_QUERY_DOWN_LEVEL_FILE_SYSTEM_CHARACTERISTICS CSVFS가 프록시인 파일 시스템에 대한 정보를 검색합니다.
FSCTL_DELETE_OBJECT_ID 지정된 파일 또는 디렉터리에서 개체 식별자를 제거합니다.
FSCTL_DELETE_REPARSE_POINT 지정된 파일 또는 디렉터리에서 재분석 지점을 삭제합니다.
FSCTL_DUPLICATE_EXTENTS_TO_FILE 애플리케이션을 대신하여 파일 바이트 범위를 복사하도록 파일 시스템에 지시합니다.
FSCTL_EXTEND_VOLUME 탑재된 볼륨의 크기를 늘립니다.
FSCTL_FILE_LEVEL_TRIM 스토리지 시스템에서 저장할 필요가 없는 지정된 파일 내의 범위를 나타냅니다.
FSCTL_FILESYSTEM_GET_STATISTICS 다양한 파일 시스템 성능 카운터에서 정보를 검색합니다.
FSCTL_FILESYSTEM_GET_STATISTICS_EX 다양한 파일 시스템 성능 카운터에서 정보를 검색합니다. 이 제어 코드에 대한 지원은 Windows 10 시작했습니다.
FSCTL_FIND_FILES_BY_SID 디렉터리에서 작성자 소유자가 지정된 SID와 일치하는 파일을 검색합니다.
FSCTL_GET_BOOT_AREA_INFO 볼륨의 부팅 섹터 위치를 검색합니다.

FSCTL_GET_COMPRESSION 파일 시스템이 스트림별 압축을 지원하는 볼륨에서 파일 또는 디렉터리의 현재 압축 상태를 검색합니다.
FSCTL_GET_INTEGRITY_INFORMATION ReFS 볼륨에서 파일 또는 디렉터리의 무결성 상태를 검색합니다. (FSCTL_GET_INTEGRITY_INFORMATION)
FSCTL_GET_NTFS_FILE_RECORD 사용 중이고 서수 값이 요청된 파일 참조 번호보다 작거나 같은 첫 번째 파일 레코드를 검색합니다.
FSCTL_GET_NTFS_VOLUME_DATA 지정된 NTFS 파일 시스템 볼륨에 대한 정보를 검색합니다.
FSCTL_GET_OBJECT_ID 지정된 파일 또는 디렉터리에 대한 개체 식별자를 검색합니다.
FSCTL_GET_REPAIR NTFS 파일 시스템의 자동 복구 메커니즘에 대한 정보를 검색합니다.
FSCTL_GET_REPARSE_POINT 지정된 핸들로 식별되는 파일 또는 디렉터리와 연결된 재분석 지점 데이터를 검색합니다.
FSCTL_GET_RETRIEVAL_POINTER_BASE 볼륨의 시작을 기준으로 파일 시스템의 첫 번째 LCN(논리 클러스터 번호)에 대한 섹터 오프셋을 반환합니다.
FSCTL_GET_RETRIEVAL_POINTERS 파일 핸들이 지정된 경우 디스트에서 특정 파일의 할당 및 위치를 설명하는 데이터 구조를 검색하거나 볼륨 핸들이 지정된 경우 볼륨에서 잘못된 클러스터의 위치를 검색합니다.
FSCTL_GET_VOLUME_BITMAP 볼륨에서 점유된 클러스터 및 사용 가능한 클러스터의 비트맵을 검색합니다.
FSCTL_INITIATE_REPAIR 단일 파일에서 자동 복구 주기를 시작하도록 NTFS 파일 시스템을 트리거합니다.

FSCTL_IS_CSV_FILE 파일이 CSVFS 볼륨에 저장되는지 여부를 확인하거나 네임스페이스 정보를 검색합니다. (FSCTL_IS_CSV_FILE)
FSCTL_IS_FILE_ON_CSV_VOLUME 파일이 CSVFS 볼륨에 저장되는지 여부를 확인하거나 네임스페이스 정보를 검색합니다. (FSCTL_IS_FILE_ON_CSV_VOLUME)
FSCTL_IS_VOLUME_MOUNTED 지정된 볼륨이 탑재되는지 또는 지정된 파일 또는 디렉터리가 탑재된 볼륨에 있는지 확인합니다.
FSCTL_IS_VOLUME_OWNED_BYCSVFS 볼륨이 CSVFS에 의해 잠겨 있는지 여부를 확인합니다.
FSCTL_LOOKUP_STREAM_FROM_CLUSTER NTFS 볼륨 또는 NTFS 볼륨의 파일에 대한 핸들이 지정된 경우 지정된 클러스터를 차지하는 스트림을 설명하는 데이터 구조 체인을 반환합니다.
FSCTL_MAKE_MEDIA_COMPATIBLE 미디어 ROM이 호환되도록 한 번 쓰기 미디어에서 열린 UDF 세션을 닫습니다.
FSCTL_MARK_HANDLE 지정된 파일 또는 디렉터리와 해당 변경 저널 레코드를 해당 파일 또는 디렉터리의 변경 내용에 대한 정보로 표시합니다.
FSCTL_MOVE_FILE 파일의 가상 클러스터 하나 이상을 하나의 논리적 클러스터에서 동일한 볼륨 내의 다른 클러스터로 재배치합니다. 이 작업은 조각 모음 중에 사용됩니다.
FSCTL_OPBATCH_ACK_CLOSE_PENDING 클라이언트 애플리케이션이 파일을 닫을 준비가 되었음을 서버에 알립니다.
FSCTL_OPLOCK_BREAK_ACK_NO_2 파일에 대한 편의적 잠금이 곧 중단된다는 알림에 응답합니다. 이 작업을 사용하여 파일에 대한 모든 편의적 잠금을 해제하고 파일을 열어 둡니다.

FSCTL_OPLOCK_BREAK_ACKNOWLEDGE 파일에 대한 배타적인 편의적 잠금이 곧 중단된다는 알림에 응답합니다. 이 작업을 사용하여 파일이 수준 2 편의적 잠금을 수신해야 함을 나타냅니다.
FSCTL_OPLOCK_BREAK_NOTIFY 호출 애플리케이션이 편의적 잠금 중단이 완료될 때까지 기다릴 수 있도록 합니다.
FSCTL_QUERY_ALLOCATED_RANGES 파일 또는 대체 스트림을 스캔하여 0이 아닌 데이터를 포함할 수 있는 범위를 찾습니다.
FSCTL_QUERY_FILE_SYSTEM_RECOGNITION 볼륨에 대한 파일 시스템 인식 정보를 쿼리합니다.
FSCTL_QUERY_ON_DISK_VOLUME_INFO UDF 관련 볼륨 정보를 요청합니다.
FSCTL_QUERY_REGION_INFO 데이터 계층화를 지원하는 볼륨에 대해 정의된 스토리지 계층 영역을 검색합니다.
FSCTL_QUERY_SPARING_INFO 볼륨의 결함 관리 속성을 검색합니다. UDF 파일 시스템에 사용됩니다.
FSCTL_QUERY_STORAGE_CLASSES 데이터 계층화를 지원하는 볼륨에 대해 정의된 스토리지 계층을 검색합니다.
FSCTL_READ_FROM_PLEX 지정된 플렉스에서 읽습니다.
FSCTL_RECALL_FILE 계층적 스토리지 관리 소프트웨어인 원격 스토리지가 관리하는 스토리지 미디어에서 파일을 다시 호출합니다.
FSCTL_REPAIR_COPIES 사용할 적절한 복사본을 선택하여 데이터 손상을 복구합니다.
FSCTL_REQUEST_BATCH_OPLOCK 파일에 대한 일괄 편의적 잠금을 요청합니다.

FSCTL_REQUEST_FILTER_OPLOCK 파일에 대한 필터 편의적 잠금을 요청합니다.
FSCTL_REQUEST_OPLOCK 파일에 대한 편의적 잠금(oplock)을 요청하고 oplock 중단이 발생했음을 확인합니다.
FSCTL_REQUEST_OPLOCK_LEVEL_1 파일에 대해 수준 1 편의적 잠금을 요청합니다.
FSCTL_REQUEST_OPLOCK_LEVEL_2 파일에 대해 수준 2 편의적 잠금을 요청합니다.
FSCTL_SET_COMPRESSION 파일 시스템이 파일별 및 디렉터리별 압축을 지원하는 볼륨에서 파일 또는 디렉터리의 압축 상태를 설정합니다.
FSCTL_SET_DEFECT_MANAGEMENT 지정된 파일에 대한 소프트웨어 결함 관리 상태를 설정합니다. UDF 파일 시스템에 사용됩니다.
FSCTL_SET_INTEGRITY_INFORMATION ReFS 볼륨에서 파일 또는 디렉터리의 무결성 상태를 검색합니다. (FSCTL_SET_INTEGRITY_INFORMATION)
FSCTL_SET_OBJECT_ID 지정된 파일 또는 디렉터리에 대한 개체 식별자를 설정합니다.
FSCTL_SET_OBJECT_ID_EXTENDED 지정된 파일 또는 디렉터리에 대한 개체 식별자와 연결된 사용자 데이터를 수정합니다.
FSCTL_SET_REPAIR NTFS 파일 시스템의 자동 복구 기능 모드를 설정합니다.
FSCTL_SET_REPARSE_POINT 파일 또는 디렉터리에 재분석 지점을 설정합니다.

FSCTL_SET_SPARSE 표시된 파일을 스파스 또는 스파스 아님으로 표시합니다. 스파스 파일에서 큰 범위의 0에는 디스크 할당이 필요하지 않을 수 있습니다.
FSCTL_SET_ZERO_DATA 파일의 지정된 범위를 0으로 채웁니다.
FSCTL_SET_ZERO_ON_DEALLOCATION NTFS 파일 시스템 파일 핸들이 할당 취소될 때 클러스터가 0으로 채워져야 함을 나타냅니다.
FSCTL_SHRINK_VOLUME 축소 작업을 수행하기 위해 볼륨을 준비하거나, 축소 작업을 커밋하거나, 축소 작업을 종료해야 한다는 신호를 보냅니다.
FSCTL_TXFS_CREATE_MINIVERSION 지정한 파일에 새 miniversion을 만듭니다. 미니버전에서는 트랜잭션 중에 파일의 스냅샷을 참조할 수 있습니다. 트랜잭션이 커밋되거나 롤백되면 미니버전이 삭제됩니다.
FSCTL_TXFS_GET_METADATA_INFO 지정된 파일을 잠근 트랜잭션의 GUID 및 파일에 대한 TxF(트랜잭션 NTFS) 메타데이터를 검색합니다(파일이 잠긴 경우).
FSCTL_TXFS_GET_TRANSACTED_VERSION TXFS_GET_TRANSACTED_VERSION 구조체를 반환합니다. 구조체는 지정된 파일의 가장 최근에 커밋된 버전인 핸들의 버전 번호를 식별합니다.
FSCTL_TXFS_LIST_TRANSACTION_LOCKED_FILES 지정된 트랜잭션에 의해 현재 잠겨 있는 모든 파일의 목록을 반환합니다. 반환 값이 ERROR_MORE_DATA인 경우 이 호출 시 전체 파일 목록을 보관하는 데 필요한 버퍼의 길이를 반환합니다.
FSCTL_TXFS_LIST_TRANSACTIONS 지정된 리소스 관리자에 현재 관련된 모든 트랜잭션의 목록을 반환합니다.
FSCTL_TXFS_MODIFY_RM RM(보조 리소스 관리자)에 대한 로그 모드 및 로그 매개 변수 정보를 설정합니다.

FSCTL_TXFS_QUERY_RM_INFORMATION RM(리소스 관리자)에 대한 정보를 검색합니다.
FSCTL_TXFS_READ_BACKUP_INFORMATION 지정된 파일에 대한 TxF(트랜잭션 NTFS)별 정보를 반환합니다.
FSCTL_TXFS_SAVEPOINT_INFORMATION FSCTL_TXFS_SAVEPOINT_INFORMATION 제어 코드는 지정된 저장점으로의 설정, 지우기 및 롤백을 제어합니다.
FSCTL_TXFS_TRANSACTION_ACTIVE 스냅샷을 만들 때 연결된 볼륨에서 활성 상태인 트랜잭션이 있는지 여부를 나타내는 부울 값을 반환합니다. 이 호출은 읽기 전용 스냅샷 볼륨에만 유효합니다.
FSCTL_TXFS_WRITE_BACKUP_INFORMATION TxF(트랜잭션 NTFS)별 정보를 지정된 파일에 씁니다.
FSCTL_USN_TRACK_MODIFIED_RANGES 대상 볼륨에서 USN(업데이트 시퀀스 번호) 변경 저널 스트림에 범위 추적 기능을 사용하도록 설정하거나 이미 활성화된 범위 추적 매개 변수를 수정합니다.
FSCTL_WAIT_FOR_REPAIR 지정된 복구가 완료되면 반환됩니다.
IOCTL_DISK_CREATE_DISK CREATE_DISK 구조체의 정보를 사용하여 지정된 디스크 및 디스크 파티션 테이블을 초기화합니다.
IOCTL_DISK_DELETE_DRIVE_LAYOUT 디스크가 섹터 0에서 디스크 끝까지 포맷되도록 마스터 부팅 레코드에서 부팅 서명을 제거합니다.
IOCTL_DISK_FORMAT_TRACKS 플로피 디스크에서 지정된 연속 트랙 세트를 포맷합니다. 추가 매개 변수를 제공하려면 IOCTL_DISK_FORMAT_TRACKS_EXinstead 사용합니다.
IOCTL_DISK_FORMAT_TRACKS_EX 플로피 디스크에서 지정된 연속 트랙 세트를 포맷합니다.

IOCTL_DISK_GET_CACHE_INFORMATION 디스크 캐시 구성 데이터를 검색합니다.
IOCTL_DISK_GET_DISK_ATTRIBUTES 지정된 디스크 디바이스의 특성을 검색합니다.
IOCTL_DISK_GET_DRIVE_GEOMETRY 실제 디스크의 geometry:_type, 실린더 수, 실린더당 트랙 수, 트랙당 섹터 및 섹터당 바이트에 대한 정보를 검색합니다.
IOCTL_DISK_GET_DRIVE_GEOMETRY_EX 실제 디스크의 geometry:_type, 실린더 수, 실린더당 트랙 수, 트랙당 섹터 및 섹터당 바이트에 대한 확장 정보를 검색합니다.
IOCTL_DISK_GET_DRIVE_LAYOUT 디스크의 파티션 테이블에 있는 각 항목에 대한 정보를 검색합니다.
IOCTL_DISK_GET_DRIVE_LAYOUT_EX 디스크의 파티션 테이블에 있는 각 항목에 대한 확장 정보를 검색합니다.
IOCTL_DISK_GET_LENGTH_INFO 지정된 디스크, 볼륨 또는 파티션의 길이를 검색합니다.
IOCTL_DISK_GET_PARTITION_INFO 디스크 파티션의 형식, 크기 및 특성에 대한 정보를 검색합니다.
IOCTL_DISK_GET_PARTITION_INFO_EX 디스크 파티션의 형식, 크기 및 특성에 대한 확장 정보를 검색합니다.
IOCTL_DISK_GROW_PARTITION 지정된 파티션을 확대합니다.
IOCTL_DISK_IS_WRITABLE 지정한 디스크가 쓰기 가능한지 여부를 확인합니다.
IOCTL_DISK_PERFORMANCE 디스크 성능 정보를 제공하는 성능 카운터를 사용하도록 설정합니다.

IOCTL_DISK_PERFORMANCE_OFF 디스크 성능 정보를 제공하는 성능 카운터를 사용하지 않도록 설정합니다.
IOCTL_DISK_REASSIGN_BLOCKS 디스크 디바이스가 하나 이상의 블록을 예비 블록 풀에 매핑하도록 지시합니다. (IOCTL_DISK_REASSIGN_BLOCKS)
IOCTL_DISK_REASSIGN_BLOCKS_EX 디스크 디바이스가 하나 이상의 블록을 예비 블록 풀에 매핑하도록 지시합니다. (IOCTL_DISK_REASSIGN_BLOCKS_EX)
IOCTL_DISK_RESET_SNAPSHOT_INFO 디스크에서 VSS(볼륨 섀도 복사본 서비스) 하드웨어 기반 섀도 복사본("스냅샷"이라고도 함) 정보를 모두 지웁니다.
IOCTL_DISK_SET_CACHE_INFORMATION 디스크 구성 데이터를 설정합니다.
IOCTL_DISK_SET_DISK_ATTRIBUTES 지정된 디스크 디바이스의 특성을 설정합니다.
IOCTL_DISK_SET_DRIVE_LAYOUT 드라이브 레이아웃 및 파티션 정보 데이터에 지정된 대로 디스크를 분할합니다.
IOCTL_DISK_SET_DRIVE_LAYOUT_EX 지정된 드라이브 레이아웃 및 파티션 정보 데이터에 따라 디스크를 분할합니다.
IOCTL_DISK_SET_PARTITION_INFO 지정된 디스크 파티션에 대한 파티션 정보를 설정합니다.
IOCTL_DISK_SET_PARTITION_INFO_EX AT 및 EFI(Extensible Firmware Interface) 파티션에 대한 레이아웃 정보를 포함하여 지정된 디스크 파티션에 대한 파티션 정보를 설정합니다.
IOCTL_DISK_UPDATE_PROPERTIES 캐시된 파티션 테이블을 무효화하고 디바이스를 다시 열거합니다.

IOCTL_DISK_VERIFY 고정 디스크에서 지정된 익스텐트를 확인합니다.
IOCTL_STORAGE_DEVICE_POWER_CAP Windows 애플리케이션은 이 제어 코드를 사용하여 스토리지 디바이스의 최대 작동 전력 소비 수준을 지정할 수 있습니다.
IOCTL_STORAGE_FIRMWARE_ACTIVATE Windows 애플리케이션은 이 제어 코드를 사용하여 지정된 디바이스에서 펌웨어 이미지를 활성화할 수 있습니다.
IOCTL_STORAGE_FIRMWARE_DOWNLOAD Windows 애플리케이션은 이 제어 코드를 사용하여 대상 디바이스에 펌웨어 이미지를 다운로드할 수 있지만 활성화할 수는 없습니다.
IOCTL_STORAGE_FIRMWARE_GET_INFO Windows 애플리케이션은 이 제어 코드를 사용하여 스토리지 디바이스에서 자세한 펌웨어 정보를 쿼리할 수 있습니다.
IOCTL_STORAGE_PROTOCOL_COMMAND Windows 애플리케이션은 이 제어 코드를 사용하여 스토리지 디바이스 또는 어댑터의 속성을 반환할 수 있습니다. 요청은 디바이스에 대한 조회 데이터 또는 어댑터의 기능 및 제한 사항과 같이 검색할 정보의 종류를 나타냅니다.
IOCTL_STORAGE_QUERY_PROPERTY Windows 애플리케이션은 이 제어 코드를 사용하여 스토리지 디바이스 또는 어댑터의 속성을 반환할 수 있습니다.
IOCTL_STORAGE_SET_TEMPERATURE_THRESHOLD Windows 애플리케이션은 이 제어 코드를 사용하여 디바이스의 온도 임계값을 설정할 수 있습니다(디바이스에서 지원하는 경우).
IOCTL_VOLUME_GET_GPT_ATTRIBUTES 볼륨의 특성을 검색합니다.
IOCTL_VOLUME_GET_VOLUME_DISK_EXTENTS 하나 이상의 디스크에서 지정된 볼륨의 물리적 위치를 검색합니다.

IOCTL_VOLUME_IS_CLUSTERED 지정된 볼륨이 클러스터되었는지 여부를 확인합니다.
IOCTL_VOLUME_OFFLINE 볼륨을 오프라인 상태로 전환합니다.
IOCTL_VOLUME_ONLINE 볼륨을 온라인 상태로 전환합니다.

열거형

CLFS_CONTEXT_MODE 클라이언트가 로그를 검사하는 데 사용하는 방향 및 액세스 방법을 나타내는 컨텍스트 모드 형식을 지정합니다.
CLFS_IOSTATS_CLASS CLFS에서 보고한 I/O 통계 유형을 정의하며 클라이언트가 GetLogIoStatistics를 호출할 때 사용됩니다.
CLFS_LOG_ARCHIVE_MODE 로그가 임시인지 여부를 지정합니다.
CLFS_MGMT_POLICY_TYPE CLFS_MGMT_POLICY_TYPE 열거형에는 유효한 정책 유형이 나열됩니다.
COPYFILE2_COPY_PHASE 오류 발생 시 복사본의 단계를 나타냅니다.
COPYFILE2_MESSAGE_ACTION CopyFile2ProgressRoutine 콜백 함수에 의해 반환되어 보류 중인 복사 작업에 대해 수행해야 하는 작업을 나타냅니다.
COPYFILE2_MESSAGE_TYPE COPYFILE2_MESSAGE 구조에서 CopyFile2ProgressRoutine 콜백 함수에 전달된 메시지의 유형을 나타냅니다.

CSV_CONTROL_OP FSCTL_CSV_CONTROL 제어 코드와 함께 사용할 CSV 제어 작업의 유형을 지정합니다.
FILE_ID_TYPE FILE_ID_DESCRIPTOR 구조체에서의 통합을 위한 판별자입니다.
FILE_INFO_BY_HANDLE_CLASS GetFileInformationByHandleEx가 검색하거나 SetFileInformationByHandle이 설정해야 하는 파일 정보의 유형을 식별합니다.
FILE_STORAGE_TIER_MEDIA_TYPE 스토리지 미디어 유형을 지정합니다.
FINDEX_INFO_LEVELS 반환된 데이터의 정보 수준을 지정하기 위해 FindFirstFileEx 함수와 함께 사용되는 값을 정의합니다.
FINDEX_SEARCH_OPS 수행할 필터링의 유형을 지정하기 위해 FindFirstFileEx 함수와 함께 사용할 값을 정의합니다.
GET_FILEEX_INFO_LEVELS 반환된 데이터의 정보 수준을 지정하기 위해 GetFileAttributesEx 및 GetFileAttributesTransacted 함수와 함께 사용할 값을 정의합니다.
IORING_CREATE_ADVISORY_FLAGS CreateIoRing을 호출하여 I/O 링을 만들기 위한 권고 플래그를 지정합니다.
IORING_CREATE_REQUIRED_FLAGS CreateIoRing을 호출하여 I/O 링을 만드는 데 필요한 플래그를 지정합니다.
IORING_FEATURE_FLAGS I/O 링 API 버전에 대한 기능 지원을 나타냅니다.
IORING_OP_CODE I/O 링 작업의 형식을 지정합니다.

IORING_REF_KIND IORING_HANDLE_REF 구조체의 형식을 지정합니다.
IORING_SQE_FLAGS I/O 링 제출 큐 항목에 대한 커널 동작 옵션을 지정합니다.
IORING_VERSION IORING API 버전을 지정합니다.
MEDIA_TYPE 다양한 형태의 디바이스 미디어를 나타냅니다.
NtmsObjectsTypes NtmsObjectsTypes 열거형 형식은 RSM 개체의 형식을 지정합니다.
NVME_ACCESS_FREQUENCIES LBA(논리 블록 주소 지정) 범위에 대한 읽기 및 쓰기 액세스 빈도를 나타내는 값을 정의합니다.
NVME_ACCESS_LATENCIES 읽기 및 쓰기 작업의 대기 시간을 나타내는 값을 정의합니다.
NVME_ADMIN_COMMANDS 관리 명령 집합에서 명령을 지정하는 값을 정의합니다. 관리 명령 집합에는 관리 제출 큐에 제출할 수 있는 명령이 포함되어 있습니다.
NVME_ASYNC_EVENT_ERROR_STATUS_CODES 일반적인 오류 이벤트 유형을 나타내는 값을 포함합니다.
NVME_ASYNC_EVENT_HEALTH_STATUS_CODES SMART/Health 상태 이벤트 유형을 나타내는 값을 포함합니다.
NVME_ASYNC_EVENT_IO_COMMAND_SET_STATUS_CODES I/O 명령 집합 이벤트 유형을 나타내는 값을 포함합니다.
NVME_ASYNC_EVENT_NOTICE_CODES 알림 이벤트 유형을 나타내는 값을 포함합니다.

NVME_ASYNC_EVENT_TYPES 비동기 이벤트 유형을 나타내는 값을 포함합니다.
NVME_CC_SHN_SHUTDOWN_NOTIFICATIONS CC(컨트롤러 구성) 종료 알림을 나타내는 값을 포함합니다.
NVME_CMBSZ_SIZE_UNITS 컨트롤러 메모리 버퍼의 크기를 나타내는 크기 단위를 지정하는 값을 포함합니다.
NVME_COMMAND_EFFECT_SBUMISSION_EXECUTION_LIMITS 연결된 명령에 대한 명령 제출 및 실행 권장 사항을 나타내는 값을 포함합니다.
NVME_CSTS_SHST_SHUTDOWN_STATUS NVME_CONTROLLER_CONFIGURATION 구조에서 **SHN** 필드를 설정하는 호스트에 의해 시작된 종료 처리의 상태를 나타내는 값을 포함합니다.
NVME_DIRECTIVE_RECEIVE_IDENTIFY_OPERATIONS 식별 작업에 대한 지시문 형식을 지정하는 값을 포함합니다.
NVME_DIRECTIVE_RECEIVE_STREAMS_OPERATIONS Streams 작업에 대한 지시문 형식을 나타내는 값을 포함합니다.
NVME_DIRECTIVE_SEND_IDENTIFY_OPERATIONS 식별 작업으로 전송되는 지시문 형식을 지정하는 값을 포함합니다.
NVME_DIRECTIVE_SEND_STREAMS_OPERATIONS Streams 작업으로 전송된 지시문 형식을 나타내는 값을 포함합니다.
NVME_DIRECTIVE_TYPES 지시문의 형식을 나타내는 값을 포함합니다.
NVME_FEATURE_VALUE_CODES **NVME_ADMIN_COMMAND_GET_FEATURES 및 NVME_ADMIN_COMMAND_SET_FEATURES 관리** 명령으로 검색하거나 구성해야 하는 기능 값을 나타내는 값을 포함합니다.

NVME_FEATURES NVME_ADMIN_COMMAND_GET_FEATURES 검색하거나 구성해야 하는 기능을 나타내고 **관리** 명령을 **NVME_ADMIN_COMMAND_SET_FEATURES** 값을 포함합니다.
NVME_FIRMWARE_ACTIVATE_ACTIONS 펌웨어 이미지 다운로드 명령 또는 이전에 다운로드하고 배치한 이미지에서 다운로드한 이미지에서 수행되는 작업을 나타내는 값을 포함합니다.
NVME_FUSED_OPERATION_CODES 명령이 융합 작업의 첫 번째 또는 두 번째 명령인지 여부를 나타내는 값을 포함합니다.
NVME_IDENTIFY_CNS_CODES NVME_CDW10_IDENTIFY 구조체의 컨트롤러 또는 CNS(네임스페이스 구조) 멤버에서 반환될 컨트롤러 또는 네임스페이스 정보의 유형을 나타내는 값을 포함합니다.
NVME_LBA_RANGE_TYPES NVME_LBA_RANGET_TYPE_ENTRY 구조체의 LBA(논리 블록 주소 지정) 범위 유형을 나타내는 값을 포함합니다.
NVME_LOG_PAGES 로그 페이지 가져오기 NVME_ADMIN_COMMAND_GET_LOG_PAGE **관리** 명령에서 검색할 수 있는 로그 페이지를 나타내는 값을 포함합니다.
NVME_NVM_COMMANDS NVM 명령 집합의 명령을 나타내는 값을 포함합니다.
NVME_NVM_QUEUE_PRIORITIES 컨트롤러에서 지원하는 경우 중재 메커니즘에서 고려할 I/O 제출 큐에 할당할 수 있는 우선 순위를 나타내는 값을 포함합니다.
NVME_PROTECTION_INFORMATION_TYPES 엔드 투 엔드 데이터 보호를 사용할 수 있는지 여부를 나타내는 값을 포함하며, 이 경우 보호 정보 유형을 지정합니다.
NVME_RESERVATION_NOTIFICATION_TYPES 예약 알림 로그 페이지의 예약 알림 유형을 나타내는 값을 포함합니다.

NVME_SECURE_ERASE_SETTINGS NVM 형식 명령의 일부로 보안 지우기 작업의 형식을 수행할지 여부를 지정하는 값을 포함합니다.
NVME_STATUS_COMMAND_SPECIFIC_CODES 특정 명령과 관련된 상태를 나타내는 값을 포함합니다.
NVME_STATUS_GENERIC_COMMAND_CODES 다양한 유형의 명령에서 제네릭인 명령과 연결된 상태를 나타내는 값을 포함합니다.
NVME_STATUS_MEDIA_ERROR_CODES NVM 미디어와 연결된 오류를 나타내거나 데이터 무결성 유형 오류를 나타내는 값을 포함합니다.
NVME_STATUS_TYPES 명령이 완료될 때 컨트롤러가 완료 큐 항목에 게시하는 상태 코드의 유형을 나타내는 값을 포함합니다.
NVME_TEMPERATURE_THRESHOLD_TYPES 전체 디바이스(컨트롤러 및 NVM 포함)의 온도에 대한 임계값 유형을 나타내는 값을 포함합니다.
PARTITION_STYLE 파티션 형식을 나타냅니다.
PRIORITY_HINT 파일 I/O 작업에 대한 우선 순위 힌트를 지정하기 위해 FILE_IO_PRIORITY_HINT_INFO 구조체와 함께 사용할 값을 정의합니다.
READ_DIRECTORY_NOTIFY_INFORMATION_CLASS ReadDirectoryChangesExW 함수를 호출하는 애플리케이션이 요청할 수 있는 정보 유형을 나타냅니다.
STORAGE_COMPONENT_HEALTH_STATUS 스토리지 구성 요소의 상태를 지정합니다.
STORAGE_DEVICE_FORM_FACTOR 디바이스의 폼 팩터를 지정합니다.

STORAGE_DEVICE_POWER_CAP_UNITS 최대 전원 임계값의 단위입니다.
STORAGE_PORT_CODE_SET 시스템에서 사용하도록 예약되었습니다. (STORAGE_PORT_CODE_SET)
STORAGE_PROPERTY_ID 스토리지 디바이스 또는 어댑터의 속성을 검색하기 위해 IOCTL_STORAGE_QUERY_PROPERTY 요청에 입력으로 전달된 STORAGE_PROPERTY_QUERY 구조체의 PropertyId 멤버에 가능한 값을 열거합니다.
STORAGE_PROTOCOL_ATA_DATA_TYPE ATA 프로토콜 데이터 형식입니다.
STORAGE_PROTOCOL_NVME_DATA_TYPE IOCTL_STORAGE_QUERY_PROPERTY 요청 중에 쿼리할 NVMe 프로토콜별 데이터의 형식을 설명합니다.
STORAGE_PROTOCOL_TYPE 스토리지 디바이스의 프로토콜을 지정합니다.
STORAGE_QUERY_TYPE IOCTL_STORAGE_QUERY_PROPERTY 제어 코드에 전달된 STORAGE_PROPERTY_QUERY 구조체에서 스토리지 디바이스 또는 어댑터의 속성에 대해 반환되는 정보를 나타내는 데 사용됩니다.
STREAM_INFO_LEVELS 반환된 데이터의 정보 수준을 지정하기 위해 FindFirstStreamW 함수와 함께 사용할 값을 정의합니다.
TRANSACTION_OUTCOME KTM이 트랜잭션에 할당할 수 있는 결과(결과)를 정의합니다.
WRITE_CACHE_CHANGE 디바이스의 쓰기 캐시 기능을 변경할 수 있는지 여부를 나타냅니다.
WRITE_CACHE_ENABLE 쓰기 캐시의 활성화 여부를 나타냅니다.

WRITE_CACHE_TYPE 캐시 유형을 지정합니다.
WRITE_THROUGH 스토리지 디바이스가 write-through 캐싱을 지원하는지 여부를 지정합니다.

Functions

AccessNtmsLibraryDoor AccessNtmsLibraryDoor 함수는 지정된 라이브러리의 문을 잠금 해제합니다. 라이브러리가 사용 중인 경우 RSM은 요청을 큐에 대기하고 성공적으로 반환합니다.
추가 일괄 처리 목록에 IDiskQuotaUser 포인터를 추가합니다.
AddLogContainer 호출 프로세스에 .blf 파일에 대한 쓰기 액세스 권한이 있고 컨테이너의 대상 디렉터리에 파일을 만드는 기능이 있는 경우 로그 핸들과 연결된 실제 로그에 컨테이너를 추가합니다.
AddLogContainerSet 호출 프로세스에서 로그 핸들에 액세스할 수 있는 경우 로그 핸들과 연결된 실제 로그에 여러 로그 컨테이너를 추가합니다.
AddNtmsMediaType AddNtmsMediaType 함수는 현재 라이브러리 개체에 관계가 없는 경우 지정된 미디어 형식을 지정된 라이브러리에 추가합니다. 그런 다음, 함수가 존재하지 않는 경우 시스템 미디어 풀을 만듭니다.
AddUserName 지정된 사용자의 볼륨에 새 할당량 항목을 추가합니다. 사용자는 도메인 및 계정 이름으로 식별됩니다.
AddUserSid 지정된 사용자의 볼륨에 새 할당량 항목을 추가합니다. 사용자는 SID(보안 식별자)로 식별됩니다.

AddUsersToEncryptedFile 지정된 암호화된 파일에 사용자를 추가합니다.
AdvanceLogBase 로그 스트림의 LSN(기본 로그 시퀀스 번호)을 지정된 LSN으로 이동합니다.
AlignReservedLog 예약 레코드 집합에 대한 섹터 정렬 예약 크기를 계산합니다.
AllocateNtmsMedia AllocateNtmsMedia 함수는 사용 가능한 미디어를 할당합니다.
AllocReservedLog 예약된 레코드 집합에 대한 섹터 맞춤 공간을 할당합니다.
AreFileApisANSI 파일 I/O 함수가 ANSI 또는 OEM 문자 집합 코드 페이지를 사용하는지 여부를 결정합니다.
AreShortNamesEnabled AreShortNamesEnabled 함수는 지정된 볼륨에 짧은 이름을 사용할 수 있는지 여부를 결정합니다.
BeginNtmsDeviceChangeDetection BeginNtmsDeviceChangeDetection 함수를 사용하면 애플리케이션이 디바이스 변경 검색 세션을 시작할 수 있습니다.
BuildIoRingCancelRequest 이전에 제출된 I/O 링 작업을 취소하려고 시도합니다.
BuildIoRingReadFile I/O 링을 사용하여 파일에서 비동기 읽기를 수행합니다.
BuildIoRingRegisterBuffers 향후 I/O 링 작업을 위해 버퍼 배열을 시스템에 등록합니다.
BuildIoRingRegisterFileHandles 이후 I/O 링 작업을 위해 시스템에 파일 핸들 배열을 등록합니다.

CancelIo 지정된 파일에 대해 호출 스레드에서 발급한 보류 중인 모든 I/O(입력 및 출력) 작업을 취소합니다.
CancelIoEx 지정된 파일 핸들에 대해 처리 중인 I/O 작업을 표시합니다. 이 함수는 I/O 작업을 만든 스레드에 관계없이 현재 프로세스의 I/O 작업만 취소합니다.
CancelNtmsLibraryRequest CancelNtmsLibraryRequest 함수는 CleanNtmsDrive 함수 호출과 같은 미해결 RSM 요청을 취소합니다. 라이브러리가 사용 중인 경우 RSM은 취소를 큐에 대기하고 성공을 반환합니다.
CancelNtmsOperatorRequest CancelNtmsOperatorRequest 함수는 지정된 RSM 연산자 요청을 취소합니다.
CancelSynchronousIo 지정된 스레드에서 실행한 보류 중인 동기 I/O 작업을 취소됨으로 표시합니다.
ChangeNtmsMediaType ChangeNtmsMediaType 함수는 지정된 PMID를 지정된 대상 미디어 풀로 이동하고 PMID의 미디어 형식 식별자를 대상 미디어 풀의 미디어 형식으로 설정합니다.
CheckNameLegalDOS8Dot3A 지정된 이름을 사용하여 FAT 파일 시스템에 파일을 만들 수 있는지 여부를 결정합니다. (ANSI)
CheckNameLegalDOS8Dot3W 지정된 이름을 사용하여 FAT 파일 시스템에 파일을 만들 수 있는지 여부를 결정합니다. (유니코드)
CLAIMMEDIALABEL ClaimMediaLabel 콜백 함수는 미디어의 연결된 애플리케이션에서 지정된 미디어 레이블을 만들었는지 여부를 결정합니다.
CleanNtmsDrive CleanNtmsDrive 함수는 정리를 위해 지정된 드라이브에 대한 정리 요청을 큐에 대기합니다.
ClfsLsnEqual 동일한 스트림의 두 LSN이 같은지 여부를 확인합니다.

ClfsLsnGreater 한 LSN이 다른 LSN보다 큰지 여부를 확인합니다. 두 LSN은 동일한 스트림에서 온 것이어야 합니다.
ClfsLsnLess 한 LSN이 다른 LSN보다 작은지 여부를 확인합니다. 두 LSN은 동일한 스트림에서 온 것이어야 합니다.
ClfsLsnNull 지정된 LSN이 CLFS_LSN_NULL 가능한 가장 작은 LSN과 같은지 여부를 확인합니다.
복제 현재 항목과 동일한 열거형 상태를 포함하는 사용자 할당량 항목의 또 다른 열거자를 만듭니다.
CloseAndResetLogFile 로그 파일을 다시 설정한 다음 로그를 종료합니다.
CloseEncryptedFileRaw 백업 또는 복원 작업 후 암호화된 파일을 닫고 연결된 시스템 리소스를 해제합니다.
CloseIoRing CreateIoRing 호출을 사용하여 이전에 연 **HIORING** 핸들을 닫습니다.
CloseNtmsNotification CloseNtmsNotification 함수는 지정된 열린 알림 채널을 닫습니다.
CloseNtmsSession CloseNtmsSession 함수는 지정된 RSM 세션을 닫습니다.
CommitComplete RM(리소스 관리자)이 TM(트랜잭션 관리자)이 요청한 트랜잭션 커밋을 완료했음을 나타냅니다.
CommitEnlistment 이 인리스트먼트 핸들과 연결된 트랜잭션을 커밋합니다. 이 함수는 통신 리소스 관리자(우수한 트랜잭션 관리자라고도 함)에서 사용됩니다.
CommitTransaction 지정된 트랜잭션을 커밋할 것을 요청합니다. (CommitTransaction)

CommitTransactionAsync 지정된 트랜잭션을 커밋할 것을 요청합니다. (CommitTransactionAsync)
CopyFile CopyFile 함수(winbase.h)는 기존 파일을 새 파일에 복사합니다.
CopyFile2 기존 파일을 새 파일에 복사하고, 콜백 함수를 통해 애플리케이션에 진행 상황을 알립니다. (CopyFile2)
CopyFileA 새 파일에 기존 파일을 복사합니다. (CopyFileA)
CopyFileExA 기존 파일을 새 파일에 복사하고, 콜백 함수를 통해 애플리케이션에 진행 상황을 알립니다. (CopyFileExA)
CopyFileExW 기존 파일을 새 파일에 복사하고, 콜백 함수를 통해 애플리케이션에 진행 상황을 알립니다. (CopyFileExW)
CopyFileFromAppW 새 파일에 기존 파일을 복사합니다. 이 함수는 유니버설 Windows 플랫폼 앱 보안 모델을 준수한다는 점을 제외하고 이 함수의 동작은 CopyFile과 동일합니다.
CopyFileTransactedA 트랜잭션 작업으로 기존 파일을 새 파일에 복사하고, 콜백 함수를 통해 애플리케이션에 진행 상황을 알립니다. (ANSI)
CopyFileTransactedW 트랜잭션 작업으로 기존 파일을 새 파일에 복사하고, 콜백 함수를 통해 애플리케이션에 진행 상황을 알립니다. (유니코드)
CopyFileW CopyFileW(유니코드) 함수(winbase.h)는 기존 파일을 새 파일에 복사합니다.
CreateDirectory CreateDirectory 함수(winbase.h)는 새 디렉터리를 만듭니다.

CreateDirectoryA 새 디렉터리를 만듭니다. (CreateDirectoryA)
CreateDirectoryExA 지정된 템플릿 디렉터리의 특성을 사용하여 새 디렉터리를 만듭니다. (ANSI)
CreateDirectoryExW 지정된 템플릿 디렉터리의 특성을 사용하여 새 디렉터리를 만듭니다. (유니코드)
CreateDirectoryFromAppW 새 디렉터리를 만듭니다. 이 함수는 유니버설 Windows 플랫폼 앱 보안 모델을 준수한다는 점을 제외하고 이 함수의 동작은 CreateDirector와 동일합니다.
CreateDirectoryTransactedA 지정된 템플릿 디렉터리의 특성을 사용하여 트랜잭션 작업으로 새 디렉터리를 만듭니다. (ANSI)
CreateDirectoryTransactedW 지정된 템플릿 디렉터리의 특성을 사용하여 트랜잭션 작업으로 새 디렉터리를 만듭니다. (유니코드)
CreateDirectoryW 새 디렉터리를 만듭니다. (CreateDirectoryW)
CreateEnlistment 인리스트먼트를 만들고, 초기 상태를 설정하고, 지정된 액세스 권한으로 인리스트먼트에 대한 핸들을 엽니다.
CreateEnumUsers 볼륨에서 할당량 사용자를 열거하기 위한 열거자 개체를 만듭니다.
CreateFile2 파일 또는 I/O 디바이스를 만들거나 엽니다.
CreateFile2FromAppW 파일 또는 I/O 디바이스를 만들거나 엽니다. 이 함수는 유니버설 Windows 플랫폼 앱 보안 모델을 준수한다는 점을 제외하고 이 함수의 동작은 CreateFile2와 동일합니다.

CreateFileA 파일 또는 I/O 디바이스를 만들거나 엽니다. 가장 일반적으로 사용되는 I/O 디바이스는 다음과 같습니다. _file, 파일 스트림, 디렉터리, 실제 디스크, 볼륨, 콘솔 버퍼, 테이프 드라이브, 통신 리소스, mailslot 및 파이프. (ANSI)
CreateFileFromAppW 파일 또는 I/O 디바이스를 만들거나 엽니다. 이 함수는 유니버설 Windows 플랫폼 앱 보안 모델을 준수한다는 점을 제외하고 이 함수의 동작은 CreateFile과 동일합니다.
CreateFileTransactedA 트랜잭션 작업으로 파일, 파일 스트림 또는 디렉터리를 만들거나 엽니다. (ANSI)
CreateFileTransactedW 트랜잭션 작업으로 파일, 파일 스트림 또는 디렉터리를 만들거나 엽니다. (유니코드)
CreateFileW 파일 또는 I/O 디바이스를 만들거나 엽니다. 가장 일반적으로 사용되는 I/O 디바이스는 다음과 같습니다. _file, 파일 스트림, 디렉터리, 실제 디스크, 볼륨, 콘솔 버퍼, 테이프 드라이브, 통신 리소스, mailslot 및 파이프. (유니코드)
CreateHardLinkA 기존 파일과 새 파일 간에 하드 링크를 설정합니다. (ANSI)
CreateHardLinkTransactedA 트랜잭션 작업으로 기존 파일과 새 파일 간에 하드 링크를 설정합니다. (ANSI)
CreateHardLinkTransactedW 트랜잭션 작업으로 기존 파일과 새 파일 간에 하드 링크를 설정합니다. (유니코드)
CreateHardLinkW 기존 파일과 새 파일 간에 하드 링크를 설정합니다. (유니코드)
CreateIoCompletionPort I/O(입출력) 완료 포트를 만들고 지정된 파일 핸들에 연결하거나, 아직 파일 핸들에 연결되지 않은 I/O 완료 포트를 만들어 나중에 연결할 수 있습니다.
CreateIoRing I/O 링 제출/완료 큐 쌍의 새 인스턴스를 만들고 I/O 링을 참조하기 위한 핸들을 반환합니다.

CreateLogContainerScanContext ScanLogContainers와 함께 사용하여 로그와 연결된 모든 로그 컨테이너를 열거하는 데 사용할 검사 컨텍스트를 만들고 첫 번째 검사를 수행합니다.
CreateLogFile 로그를 만들거나 엽니다.
CreateLogMarshallingArea 로그에 대한 마샬링 영역을 만들고 성공하면 마샬링 컨텍스트를 반환합니다. 마샬링 영역을 만들기 전에 로그에 하나 이상의 컨테이너가 있어야 합니다.
CreateNtmsMediaA CreateNtmsMedia 함수는 오프라인 미디어의 새 부분에 대한 PMID 및 측면(또는 측면)을 만듭니다. 미디어는 lpPhysicalMedia에 지정된 미디어 풀에 배치됩니다. (ANSI)
CreateNtmsMediaPool CreateNtmsMediaPool 함수는 새 애플리케이션 미디어 풀을 만듭니다. (CreateNtmsMediaPool)
CreateNtmsMediaPoolA CreateNtmsMediaPool 함수는 새 애플리케이션 미디어 풀을 만듭니다. (CreateNtmsMediaPoolA)
CreateNtmsMediaPoolW CreateNtmsMediaPoolW(유니코드) 함수는 새 애플리케이션 미디어 풀을 만듭니다. (CreateNtmsMediaPoolW)
CreateNtmsMediaW CreateNtmsMedia 함수는 오프라인 미디어의 새 부분에 대한 PMID 및 측면(또는 측면)을 만듭니다. 미디어는 lpPhysicalMedia에 지정된 미디어 풀에 배치됩니다. (유니코드)
CreateResourceManager 새 RM(리소스 관리자) 개체를 만들고 RM을 TM(트랜잭션 관리자)과 연결합니다.
CreateSymbolicLinkA 기호화 된 링크를 만듭니다. (ANSI)
CreateSymbolicLinkTransactedA 트랜잭션된 작업으로 기호 링크를 만듭니다. (ANSI)

CreateSymbolicLinkTransactedW 트랜잭션된 작업으로 기호 링크를 만듭니다. (유니코드)
CreateSymbolicLinkW 기호화 된 링크를 만듭니다. (유니코드)
CreateTransaction 새 트랜잭션 개체를 만듭니다.
CreateTransactionManager 새 TM(트랜잭션 관리자) 개체를 만들고 지정된 액세스 권한이 있는 핸들을 반환합니다.
CreateUserBatch 여러 사용자의 할당량 설정에 대한 업데이트를 동시에 최적화하기 위한 일괄 처리 개체를 만듭니다.
DeallocateNtmsMedia DeallocateNtmsMedia 함수는 지정된 논리 미디어와 연결된 쪽의 할당을 취소합니다.
DecommissionNtmsMedia DecommissionNtmsMedia 함수는 사용 가능한 상태에서 서비스 해제됨 상태로 한 쪽을 이동합니다.
DecryptFileA 암호화된 파일 또는 디렉터리를 암호 해독합니다. (ANSI)
DecryptFileW 암호화된 파일 또는 디렉터리를 암호 해독합니다. (유니코드)
DefineDosDeviceA MS-DOS 디바이스 이름을 정의, 재정의 또는 삭제합니다. (DefineDosDeviceA)
DefineDosDeviceW MS-DOS 디바이스 이름을 정의, 재정의 또는 삭제합니다. (DefineDosDeviceW)
DeleteFile DeleteFile 함수(winbase.h)는 기존 파일을 삭제합니다.

DeleteFileA 기존 파일을 삭제합니다. (DeleteFileA)
DeleteFileFromAppW 기존 파일을 삭제합니다. 이 함수는 유니버설 Windows 플랫폼 앱 보안 모델을 준수한다는 점을 제외하고 이 함수의 동작은 DeleteFile과 동일합니다.
DeleteFileTransactedA 트랜잭션 작업으로 기존 파일을 삭제합니다. (ANSI)
DeleteFileTransactedW 트랜잭션 작업으로 기존 파일을 삭제합니다. (유니코드)
DeleteFileW 기존 파일을 삭제합니다. (DeleteFileW)
DeleteLogByHandle 삭제할 지정된 로그를 표시합니다. 로그에 대한 모든 핸들, 마샬링 영역 및 읽기 컨텍스트가 닫혀 있으면 로그가 실제로 삭제됩니다. 로그가 실제 로그인 경우 해당 기본 컨테이너가 삭제됩니다.
DeleteLogFile 삭제할 로그를 표시합니다. 로그에 대한 모든 핸들, 마샬링 영역 및 읽기 컨텍스트가 닫혀 있으면 로그가 실제로 삭제됩니다. 로그가 실제 로그인 경우 해당 기본 컨테이너가 삭제됩니다.
DeleteLogMarshallingArea CreateLogMarshallingArea를 성공적으로 호출하여 만든 마샬링 영역을 삭제합니다.
DeleteNtmsDrive DeleteNtmsDrive 함수는 RSM 데이터베이스에서 드라이브를 삭제합니다. 드라이브에는 NTMS_NOT_PRESENT dwOperationalState가 있어야 합니다.
DeleteNtmsLibrary DeleteNtmsLibrary 함수는 RSM 데이터베이스에서 라이브러리 및 라이브러리에 포함된 모든 디바이스를 삭제합니다. 라이브러리의 모든 미디어가 오프라인 라이브러리로 이동됩니다.
DeleteNtmsMedia DeleteNtmsMedia 함수는 데이터베이스에서 지정된 미디어에 대한 모든 참조를 제거하여 RSM에서 오프라인 미디어의 물리적 부분을 삭제합니다.

DeleteNtmsMediaPool DeleteNtmsMediaPool 함수는 지정된 애플리케이션 미디어 풀을 삭제합니다.
DeleteNtmsMediaType DeleteNtmsMediaType 함수는 지정된 미디어 형식의 물리적 미디어 개체가 라이브러리에 없는 경우 지정된 라이브러리에서 지정된 미디어 형식 관계를 삭제합니다.
DeleteNtmsRequests DeleteNtmsRequests 함수는 RSM 데이터베이스에서 요청 또는 요청 목록을 삭제합니다.
DeleteUser 볼륨 할당량 정보 파일에서 사용자 항목을 제거합니다.
DeleteVolumeMountPointA 드라이브 문자 또는 탑재된 폴더를 삭제합니다. (DeleteVolumeMountPointA)
DeleteVolumeMountPointW 드라이브 문자 또는 탑재된 폴더를 삭제합니다. (DeleteVolumeMountPointW)
DeregisterManageableLogClient 로그 관리자를 사용하여 클라이언트를 등록 취소합니다.
DisableNtmsObject DisableNtmsObject 함수는 지정된 RSM 개체를 사용하지 않도록 설정합니다.
DismountNtmsDrive DismountNtmsDrive 함수는 지정된 드라이브의 미디어를 스토리지 슬롯으로 이동하는 명령을 큐에 대기합니다. 이 함수는 MountNtmsMedia 함수와 페어링되어야 합니다.
DismountNtmsMedia DismountNtmsMedia 함수는 드라이브의 지정된 미디어를 스토리지로 이동하는 명령을 큐에 대기합니다. 이 함수는 MountNtmsMedia 함수와 페어링되어야 합니다.
DumpLogRecords 지정된 로그를 검사합니다. 레코드 형식에 따라 로그 레코드를 필터링합니다. 및 는 호출자가 여는 출력 파일 스트림에 레코드를 배치합니다.

DuplicateEncryptionInfoFile 한 파일 또는 디렉터리에서 다른 파일로 EFS 메타데이터를 복사합니다.
EjectDiskFromSADriveA EjectDiskFromSADrive 함수는 독립 실행형 이동식 드라이브에 있는 미디어를 배출합니다. (ANSI)
EjectDiskFromSADriveW EjectDiskFromSADrive 함수는 독립 실행형 이동식 드라이브에 있는 미디어를 배출합니다. (유니코드)
EjectNtmsCleaner EjectNtmsCleaner 함수는 현재 예약된 클리너 슬롯에서 정리 카트리지를 배출합니다.
EjectNtmsMedia EjectNtmsMedia 함수는 현재 라이브러리의 포트에서 지정된 매체를 배출합니다. 라이브러리가 사용 중인 경우 RSM은 EjectNtmsMedia를 큐에 대기하고 성공을 반환합니다.
EnableNtmsObject EnableNtmsObject 함수는 지정된 개체를 사용하도록 설정합니다.
EncryptFileA 파일 또는 디렉터리를 암호화합니다. (ANSI)
EncryptFileW 파일 또는 디렉터리를 암호화합니다. (유니코드)
EncryptionDisable 지정된 디렉터리와 해당 디렉터리에 암호화를 사용하거나 사용하지 않도록 설정합니다.
EndNtmsDeviceChangeDetection EndNtmsDeviceChangeDetection 함수는 SetNtmsDeviceChangeDetection 함수를 사용하여 지정된 대상 디바이스에 대한 디바이스 변경 검색을 종료하고 변경 검색 핸들을 닫습니다.
EnumerateNtmsObject EnumerateNtmsObject 함수는 lpContainerId 매개 변수에 포함된 RSM 개체를 열거합니다.

ExportNtmsDatabase ExportNtmsDatabase 함수는 RSM 데이터베이스 디렉터리에 일관된 데이터베이스 파일 집합을 만듭니다.
FileEncryptionStatusA 지정된 파일의 암호화 상태를 검색합니다. (ANSI)
FileEncryptionStatusW 지정된 파일의 암호화 상태를 검색합니다. (유니코드)
FindClose FindFirstFile, FindFirstFileEx, FindFirstFileNameW, FindFirstFileNameTransactedW, FindFirstFileTransacted, FindFirstStreamTransactedW 또는 FindFirstStreamW 함수가 연 파일 검색 핸들을 닫습니다.
FindCloseChangeNotification 변경 알림 핸들 모니터링을 중지합니다.
FindFirstChangeNotificationA 변경 알림 핸들을 만들고 초기 변경 알림 필터 조건을 설정합니다. (ANSI)
FindFirstChangeNotificationW 변경 알림 핸들을 만들고 초기 변경 알림 필터 조건을 설정합니다. (유니코드)
FindFirstFileA 디렉터리를 검색하여 특정 이름(와일드카드를 사용하는 경우 부분 이름)과 일치하는 이름을 가진 파일 또는 하위 디렉터리를 찾습니다. (ANSI)
FindFirstFileExA 디렉터리를 검색하여 지정된 이름 및 특성과 일치하는 파일 또는 하위 디렉터리를 찾습니다. (FindFirstFileExA)
FindFirstFileExFromAppW 디렉터리를 검색하여 지정된 이름 및 특성과 일치하는 파일 또는 하위 디렉터리를 찾습니다.
FindFirstFileExW 디렉터리를 검색하여 지정된 이름 및 특성과 일치하는 파일 또는 하위 디렉터리를 찾습니다. (FindFirstFileExW)

FindFirstFileNameTransactedW 트랜잭션 작업으로 지정된 파일에 대한 모든 하드 링크의 열거형을 만듭니다. 이 함수는 후속 FindNextFileNameW 함수 호출에 사용할 수 있는 열거형에 대한 핸들을 반환합니다.
FindFirstFileNameW 지정된 파일에 대한 모든 하드 링크의 열거형을 만듭니다. FindFirstFileNameW 함수는 후속 FindNextFileNameW 함수 호출에 사용할 수 있는 열거형에 대한 핸들을 반환합니다.
FindFirstFileTransactedA 트랜잭션 작업으로 디렉터리를 검색하여 특정 이름과 일치하는 이름을 가진 파일 또는 하위 디렉터리를 찾습니다. (ANSI)
FindFirstFileTransactedW 트랜잭션 작업으로 디렉터리를 검색하여 특정 이름과 일치하는 이름을 가진 파일 또는 하위 디렉터리를 찾습니다. (유니코드)
FindFirstFileW 디렉터리를 검색하여 특정 이름(와일드카드를 사용하는 경우 부분 이름)과 일치하는 이름을 가진 파일 또는 하위 디렉터리를 찾습니다. (유니코드)
FindFirstStreamTransactedW 트랜잭션 작업으로 지정된 파일 또는 디렉터리의 첫 번째 스트림을 열거합니다.
FindFirstStreamW 첫 번째 스트림을 지정된 파일 또는 디렉터리에 ::$DATA 스트림 형식으로 열거합니다.
FindFirstVolumeA 컴퓨터에 있는 볼륨의 이름을 검색합니다. (FindFirstVolumeA)
FindFirstVolumeMountPointA 지정된 볼륨에 있는 탑재된 폴더의 이름을 검색합니다. (ANSI)
FindFirstVolumeMountPointW 지정된 볼륨에 있는 탑재된 폴더의 이름을 검색합니다. (유니코드)
FindFirstVolumeW 컴퓨터에 있는 볼륨의 이름을 검색합니다. (FindFirstVolumeW)

FindNextChangeNotification 운영 체제가 다음에 적절한 변경 사항을 감지할 때 변경 알림 핸들에 신호를 보내도록 요청합니다.
FindNextFileA 이전 FindFirstFile, FindFirstFileEx 또는 FindFirstFileTransacted 함수 호출에서 파일을 계속 검색합니다 (ANSI)
FindNextFileNameW FindFirstFileNameW 함수를 성공적으로 호출하여 반환된 핸들을 사용하여 파일에 대한 하드 링크를 계속 열거합니다.
FindNextFileW 이전 FindFirstFile, FindFirstFileEx 또는 FindFirstFileTransacted 함수 호출에서 파일을 계속 검색합니다 (유니코드)
FindNextStreamW 이전 FindFirstStreamW 함수 호출로 시작된 스트림 검색을 계속합니다.
FindNextVolumeA FindFirstVolume 함수 호출로 시작된 볼륨 검색을 계속합니다. (FindNextVolumeA)
FindNextVolumeMountPointA FindFirstVolumeMountPoint 함수 호출로 시작된 탑재된 폴더 검색을 계속합니다. (ANSI)
FindNextVolumeMountPointW FindFirstVolumeMountPoint 함수 호출로 시작된 탑재된 폴더 검색을 계속합니다. (유니코드)
FindNextVolumeW FindFirstVolume 함수 호출로 시작된 볼륨 검색을 계속합니다. (FindNextVolumeW)
FindUserName 볼륨 할당량 정보에서 특정 항목을 찾습니다.
FindUserSid 볼륨 할당량 정보에서 특정 사용자 항목을 찾습니다.

FindVolumeClose 지정된 볼륨 검색 핸들을 닫습니다.
FindVolumeMountPointClose 지정된 탑재된 폴더 검색 핸들을 닫습니다.
FlushLogBuffers 이 마샬링 영역에 추가된 모든 레코드를 디스크에 플러시하도록 합니다.
FlushLogToLsn 지정된 LSN(로그 시퀀스 번호)이 있는 레코드까지 이 마샬링 영역에 추가된 모든 레코드를 디스크에 플러시하도록 합니다. 이 작업 중에 지정된 것보다 많은 레코드가 플러시될 수 있습니다.
FlushToDisk 기본 파일 시스템에 대한 단일 호출에서 디스크에 사용자 개체 변경 내용을 씁니다.
FreeEncryptionCertificateHashList 인증서 해시 목록을 해제합니다.
FreeReservedLog ReserveAndAppendLog, ReserveAndAppendLogAligned 또는 AllocReservedLog를 호출하여 만든 마샬링 영역에서 예약된 로그 레코드 수를 줄입니다.
GetAccountStatus 사용자 개체 계정의 상태를 검색합니다.
GetBinaryTypeA 파일이 실행 파일(.exe)인지, 실행 파일이라면 어떤 하위 시스템이 실행 파일을 실행하는지 확인합니다. (ANSI)
GetBinaryTypeW 파일이 실행 파일(.exe)인지, 실행 파일이라면 어떤 하위 시스템이 실행 파일을 실행하는지 확인합니다. (유니코드)
GetCompressedFileSizeA 지정된 파일을 저장하는 데 사용되는 디스크 스토리지의 실제 바이트 수를 검색합니다. (ANSI)

GetCompressedFileSizeTransactedA 트랜잭션 작업으로 지정된 파일을 저장하는 데 사용되는 디스크 스토리지의 실제 바이트 수를 검색합니다. (ANSI)
GetCompressedFileSizeTransactedW 트랜잭션 작업으로 지정된 파일을 저장하는 데 사용되는 디스크 스토리지의 실제 바이트 수를 검색합니다. (유니코드)
GetCompressedFileSizeW 지정된 파일을 저장하는 데 사용되는 디스크 스토리지의 실제 바이트 수를 검색합니다. (유니코드)
GetCurrentClockTransactionManager 트랜잭션 관리자에서 가상 클록 값을 가져옵니다.
GetCurrentDirectory 현재 프로세스의 현재 디렉터리를 검색합니다.
GetDefaultQuotaLimit 볼륨의 기본 할당량 제한을 검색합니다.
GetDefaultQuotaLimitText 볼륨의 기본 할당량 제한을 검색합니다. 제한은 텍스트 문자열로 표현됩니다. 예를 들어 10.5MB입니다.
GetDefaultQuotaThreshold 볼륨의 기본 할당량 경고 임계값을 검색합니다.
GetDefaultQuotaThresholdText 볼륨의 기본 경고 임계값을 검색합니다.
GetDiskFreeSpaceA 디스크의 사용 가능한 공간을 포함하여 지정된 디스크에 대한 정보를 검색합니다. (ANSI)
GetDiskFreeSpaceExA 디스크 볼륨에서 사용할 수 있는 공간의 양(총 공간 양, 사용 가능한 공간의 총 양 및 호출 스레드와 연결된 사용자가 사용할 수 있는 총 사용 가능한 공간 양)에 대한 정보를 검색합니다. (ANSI)

GetDiskFreeSpaceExW 디스크 볼륨에서 사용할 수 있는 공간의 양(총 공간 양, 사용 가능한 공간의 총 양 및 호출 스레드와 연결된 사용자가 사용할 수 있는 총 사용 가능한 공간 양)에 대한 정보를 검색합니다. (유니코드)
GetDiskFreeSpaceW 디스크의 사용 가능한 공간을 포함하여 지정된 디스크에 대한 정보를 검색합니다. (유니코드)
GetDiskSpaceInformationA 지정된 루트 경로의 볼륨에 대한 디스크 공간 정보를 가져옵니다.
GetDiskSpaceInformationW 지정된 루트 경로의 볼륨에 대한 디스크 공간 정보를 가져옵니다.
GetDriveTypeA 디스크 드라이브가 이동식, 고정식, CD-ROM, RAM 디스크 또는 네트워크 드라이브인지 여부를 확인합니다. (ANSI)
GetDriveTypeW 디스크 드라이브가 이동식, 고정식, CD-ROM, RAM 디스크 또는 네트워크 드라이브인지 여부를 확인합니다. (유니코드)
GetEnlistmentId 지정된 인리스트먼트의 식별자(ID)를 가져옵니다.
GetEnlistmentRecoveryInformation KTM에서 불투명한 복구 데이터 구조를 검색합니다.
GetExpandedNameA 파일이 Lempel-Ziv 알고리즘을 통해 압축된 경우 압축된 파일의 원래 이름을 검색합니다. (ANSI)
GetExpandedNameW 파일이 Lempel-Ziv 알고리즘을 통해 압축된 경우 압축된 파일의 원래 이름을 검색합니다. (유니코드)
GetFileAttributesA 지정된 파일 또는 디렉터리의 파일 시스템 특성을 검색합니다. (ANSI)

GetFileAttributesExA 지정된 파일 또는 디렉터리의 특성을 검색합니다. (ANSI)
GetFileAttributesExFromAppW 지정된 파일 또는 디렉터리의 특성을 검색합니다. 이 함수는 유니버설 Windows 플랫폼 앱 보안 모델을 준수한다는 점을 제외하고 이 함수의 동작은 GetFileAttributesEx와 동일합니다.
GetFileAttributesExW 지정된 파일 또는 디렉터리의 특성을 검색합니다. (유니코드)
GetFileAttributesTransactedA 트랜잭션 작업으로 지정된 파일 또는 디렉터리의 파일 시스템 특성을 검색합니다. (ANSI)
GetFileAttributesTransactedW 트랜잭션 작업으로 지정된 파일 또는 디렉터리의 파일 시스템 특성을 검색합니다. (유니코드)
GetFileAttributesW 지정된 파일 또는 디렉터리의 파일 시스템 특성을 검색합니다. (유니코드)
GetFileBandwidthReservation 지정된 파일이 상주하는 볼륨의 대역폭 예약 속성을 검색합니다.
GetFileInformationByHandle 지정된 파일의 파일 정보를 검색합니다. (GetFileInformationByHandle)
GetFileInformationByHandleEx 지정된 파일의 파일 정보를 검색합니다. (GetFileInformationByHandleEx)
GetFileSize 지정된 파일의 크기(바이트)를 검색합니다.
GetFileSizeEx 지정된 파일의 크기를 검색합니다.
GetFileType은 지정된 파일의 파일 유형을 검색합니다.

GetFinalPathNameByHandleA 지정된 파일의 최종 경로를 검색합니다. (ANSI)
GetFinalPathNameByHandleW 지정된 파일의 최종 경로를 검색합니다. (유니코드)
GetFullPathNameA 지정된 파일의 전체 경로 및 파일 이름을 검색합니다. (ANSI)
GetFullPathNameTransactedA 트랜잭션 작업으로 지정된 파일의 전체 경로 및 파일 이름을 검색합니다. (ANSI)
GetFullPathNameTransactedW 트랜잭션 작업으로 지정된 파일의 전체 경로 및 파일 이름을 검색합니다. (유니코드)
GetFullPathNameW 지정된 파일의 전체 경로 및 파일 이름을 검색합니다. (유니코드)
GetID DiskQuotaUser 개체의 고유 식별자(ID) 번호를 검색합니다.
GetIoRingInfo I/O 링의 API 버전 및 큐 크기에 대한 정보를 가져옵니다.
GetLogContainerName 지정된 컨테이너의 전체 경로 이름을 검색합니다.
GetLogFileInformation 지정된 로그 및 해당 현재 상태에 대한 메타데이터가 포함된 버퍼를 반환하며, 이는 CLFS_INFORMATION 구조체에 의해 정의됩니다.
GetLogicalDrives 현재 사용 가능한 디스크 드라이브를 나타내는 비트 마스크를 검색합니다.
GetLogicalDriveStringsA 버퍼를 시스템에서 유효한 드라이브를 지정하는 문자열로 채웁니다. (GetLogicalDriveStringsA)

GetLogicalDriveStringsW 버퍼를 시스템에서 유효한 드라이브를 지정하는 문자열로 채웁니다. (GetLogicalDriveStringsW)
GetLogIoStatistics 지정된 핸들과 연결된 전용 또는 멀티플렉싱 로그에 대한 로그 I/O 통계를 검색합니다.
GetLongPathNameA 지정된 경로를 긴 형식으로 변환합니다. (ANSI)
GetLongPathNameTransactedA 트랜잭션 작업으로 지정된 경로를 긴 형식으로 변환합니다. (ANSI)
GetLongPathNameTransactedW 트랜잭션 작업으로 지정된 경로를 긴 형식으로 변환합니다. (유니코드)
GetLongPathNameW 지정된 경로를 긴 형식으로 변환합니다. (유니코드)
GetName 디스크 할당량 사용자와 연결된 이름 문자열을 검색합니다.
GetNextLogArchiveExtent 로그 보관 컨텍스트에서 다음 보관 익스텐트 집합을 검색합니다.
GetNotificationResourceManager RM(리소스 관리자)에 대한 알림을 요청하고 받습니다. 이 함수는 트랜잭션 상태가 변경될 때 RM 레지스터에서 알림을 받는 데 사용됩니다.
GetNotificationResourceManagerAsync RM(리소스 관리자)에 대한 비동기 알림을 요청하고 받습니다. 이 함수는 트랜잭션 상태가 변경될 때 RM 레지스터에서 알림을 받는 데 사용됩니다.
GetNtmsMediaPoolNameA GetNtmsMediaPoolName 함수는 지정된 미디어 풀의 전체 이름 계층 구조를 검색합니다. (ANSI)

GetNtmsMediaPoolNameW GetNtmsMediaPoolName 함수는 지정된 미디어 풀의 전체 이름 계층 구조를 검색합니다. (유니코드)
GetNtmsObjectAttributeA GetNtmsObjectAttribute 함수는 지정된 RSM 개체에서 확장 특성(명명된 프라이빗 데이터)을 검색합니다. (ANSI)
GetNtmsObjectAttributeW GetNtmsObjectAttribute 함수는 지정된 RSM 개체에서 확장 특성(명명된 프라이빗 데이터)을 검색합니다. (유니코드)
GetNtmsObjectInformation GetNtmsObjectInformation 함수는 지정된 개체에 대한 개체의 정보 구조를 반환합니다. (GetNtmsObjectInformation)
GetNtmsObjectInformationA GetNtmsObjectInformation 함수는 지정된 개체에 대한 개체의 정보 구조를 반환합니다. (GetNtmsObjectInformationA)
GetNtmsObjectInformationW GetNtmsObjectInformationW(유니코드) 함수는 지정된 개체에 대한 개체의 정보 구조를 반환합니다. (GetNtmsObjectInformationW)
GetNtmsObjectSecurity GetNtmsObjectSecurity 함수는 지정된 RSM 개체에 대한 보안 설명자를 읽습니다.
GetNtmsRequestOrder GetNtmsRequestOrder 함수는 지정된 요청이 라이브러리 큐에서 처리되는 순서를 가져옵니다.
GetNtmsUIOptionsA GetNtmsUIOptions 함수는 지정된 형식의 사용자 인터페이스가 지정된 개체에 대해 전달되는 컴퓨터 이름 목록을 가져옵니다. (ANSI)
GetNtmsUIOptionsW GetNtmsUIOptions 함수는 지정된 형식의 사용자 인터페이스가 지정된 개체에 대해 전달되는 컴퓨터 이름 목록을 가져옵니다. (유니코드)

GetQueuedCompletionStatus I/O 완료 패킷을 지정된 I/O 완료 포트의 큐에서 제거하려고 시도합니다.
GetQueuedCompletionStatusEx 여러 완료 포트 항목을 동시에 검색합니다.
GetQuotaInformation 사용자의 경고 임계값, 하드 할당량 제한 및 사용된 할당량에 대한 값을 검색합니다.
GetQuotaLimit 볼륨에서 사용자의 할당량 제한 값을 검색합니다.
GetQuotaLimitText 볼륨에 대한 사용자의 할당량 제한을 검색합니다.
GetQuotaLogFlags 볼륨에서 사용자 관련 할당량 이벤트의 로깅을 제어하는 플래그를 검색합니다.
GetQuotaState 할당량 시스템의 상태를 설명하는 플래그 집합을 검색합니다.
GetQuotaThreshold 볼륨에서 사용자의 경고 임계값을 검색합니다.
GetQuotaThresholdText 볼륨에 대한 사용자의 경고 임계값을 검색합니다.
GetQuotaUsed 볼륨에서 사용자의 할당량 사용 값을 검색합니다.
GetQuotaUsedText 볼륨에 대한 사용자의 할당량 사용 값을 검색합니다.
GetShortPathNameA 지정된 경로의 짧은 경로 형식을 검색합니다. (GetShortPathNameA)

GetShortPathNameW 지정된 경로의 짧은 경로 형식을 검색합니다. (GetShortPathNameW)
GetSid 사용자의 SID(보안 식별자)를 검색합니다. (IDiskQuotaUser.GetSid)
GetSidLength 사용자 SID(보안 식별자)의 길이를 바이트 단위로 검색합니다.
GetSystemWow64Directory2A 지정된 이미지 파일 컴퓨터 형식을 사용하여 WOW64에서 사용하는 시스템 디렉터리의 경로를 검색합니다. (ANSI)
GetSystemWow64Directory2W 지정된 이미지 파일 컴퓨터 형식을 사용하여 WOW64에서 사용하는 시스템 디렉터리의 경로를 검색합니다. (유니코드)
GetSystemWow64DirectoryA WOW64에서 사용하는 시스템 디렉터리의 경로를 검색합니다. (ANSI)
GetSystemWow64DirectoryW WOW64에서 사용하는 시스템 디렉터리의 경로를 검색합니다. (유니코드)
GetTempFileName GetTempFileName 함수(winbase.h)는 임시 파일의 이름을 만듭니다. 고유한 파일 이름이 생성되면 빈 파일이 만들어지고 해당 파일에 대한 핸들이 해제됩니다. 그렇지 않으면 파일 이름만 생성됩니다.
GetTempFileNameA 임시 파일의 이름을 만듭니다. 고유한 파일 이름이 생성되면 빈 파일이 만들어지고 해당 파일에 대한 핸들이 해제됩니다. 그렇지 않으면 파일 이름만 생성됩니다. (GetTempFileNameA)
GetTempFileNameW 임시 파일의 이름을 만듭니다. 고유한 파일 이름이 생성되면 빈 파일이 만들어지고 해당 파일에 대한 핸들이 해제됩니다. 그렇지 않으면 파일 이름만 생성됩니다. (GetTempFileNameW)
GetTempPath2A 호출 프로세스의 권한에 따라 임시 파일에 대해 지정된 디렉터리의 경로를 검색합니다. (ANSI)

GetTempPath2W 호출 프로세스의 권한에 따라 임시 파일에 대해 지정된 디렉터리의 경로를 검색합니다. (유니코드)
GetTempPathA 임시 파일에 대해 지정된 디렉터리의 경로를 검색합니다. (ANSI)
GetTempPathW 임시 파일에 대해 지정된 디렉터리의 경로를 검색합니다. (유니코드)
GetTransactionId 지정된 트랜잭션의 식별자(ID)를 가져옵니다.
GetTransactionInformation 지정된 트랜잭션에 대해 요청된 정보를 반환합니다.
GetTransactionManagerId 지정된 트랜잭션 관리자의 식별자를 가져옵니다.
GetVolumeInformationA 지정된 루트 디렉터리와 연결된 파일 시스템 및 볼륨 관련 정보를 검색합니다. (ANSI)
GetVolumeInformationByHandleW 지정된 파일과 연결된 파일 시스템 및 볼륨 관련 정보를 검색합니다.
GetVolumeInformationW 지정된 루트 디렉터리와 연결된 파일 시스템 및 볼륨 관련 정보를 검색합니다. (유니코드)
GetVolumeNameForVolumeMountPointA 지정된 볼륨 탑재 지점(드라이브 문자, 볼륨 GUID 경로 또는 탑재된 폴더)과 연결된 볼륨의 볼륨 GUID 경로를 검색합니다. (GetVolumeNameForVolumeMountPointA)
GetVolumeNameForVolumeMountPointW 지정된 볼륨 탑재 지점(드라이브 문자, 볼륨 GUID 경로 또는 탑재된 폴더)과 연결된 볼륨의 볼륨 GUID 경로를 검색합니다. (GetVolumeNameForVolumeMountPointW)
GetVolumePathNameA 지정된 경로가 탑재되는 볼륨 탑재 지점을 검색합니다. (GetVolumePathNameA)

GetVolumePathNamesForVolumeNameA 지정된 볼륨에 대한 드라이브 문자 및 탑재된 폴더 경로 목록을 검색합니다. (GetVolumePathNamesForVolumeNameA)
GetVolumePathNamesForVolumeNameW 지정된 볼륨에 대한 드라이브 문자 및 탑재된 폴더 경로 목록을 검색합니다. (GetVolumePathNamesForVolumeNameW)
GetVolumePathNameW 지정된 경로가 탑재되는 볼륨 탑재 지점을 검색합니다. (GetVolumePathNameW)
GetVolumesFromDriveA GetVolumesFromDrive 함수는 지정된 이동식 스토리지 미디어 드라이브에 대한 볼륨 및 드라이브 문자를 검색합니다. (ANSI)
GetVolumesFromDriveW GetVolumesFromDrive 함수는 지정된 이동식 스토리지 미디어 드라이브에 대한 볼륨 및 드라이브 문자를 검색합니다. (유니코드)
GiveUserNameResolutionPriority 지정된 사용자 개체를 큐의 머리로 승격하여 다음 줄에서 해결하도록 합니다.
HandleLogFull 로그를 예약하거나 로그에 추가하려고 하면 로그 전체 오류 메시지와 함께 실패할 때 관리되는 로그 클라이언트에서 호출됩니다. 로그 관리자는 클라이언트에 대한 로그 전체 조건을 확인하려고 시도하고 결과가 알려지면 클라이언트에 알 수 있습니다.
IdentifyNtmsSlot IdentifiNtmsSlot 함수는 라이브러리의 지정된 슬롯에 있는 미디어를 식별합니다. 식별이 완료되면 명령이 반환됩니다.
ImportNtmsDatabase ImportNtmsDatabase 함수를 사용하면 RSM이 다음에 RSM을 다시 시작할 때 데이터베이스 내보내기 디렉터리에서 데이터베이스 파일을 가져옵니다.
초기화 요청된 액세스 권한으로 NTFS 파일 시스템 볼륨을 열어 새 DiskQuotaControl 개체를 초기화합니다.

InjectNtmsCleaner InjectNtmsCleaner 함수를 사용하면 지정된 라이브러리 단위에 클리너 카트리지를 삽입할 수 있습니다.
InjectNtmsMedia InjectNtmsMedia 함수를 사용하면 지정된 라이브러리의 포트에 미디어를 삽입할 수 있습니다. 라이브러리가 사용 중인 경우 RSM은 InjectNtmsMedia를 큐에 대기하고 성공을 반환합니다.
InstallLogPolicy 로그에 대한 정책을 설치(설정)합니다.
Invalidate 할당량 사용자 개체에 저장된 할당량 정보를 무효화합니다.
InvalidateSidNameCache 새 사용자 개체(IEnumDiskQuotaUsers::Next, IDiskQuotaControl::FindUserSid 및 IDiskQuotaControl::FindUserName)에 대한 후속 요청이 도메인 컨트롤러에서 사용자 이름을 가져와야 하므로 시스템의 SID-이름 캐시 콘텐츠를 무효화합니다.
InventoryNtmsLibrary InventoryNtmsLibrary 함수는 지정된 라이브러리의 인벤토리를 큐에 대기합니다. 라이브러리가 사용 중인 경우 RSM은 InventoryNtmsLibrary를 큐에 대기하고 성공을 반환합니다.
IoRingBufferRefFromIndexAndOffset 제공된 버퍼 인덱스 및 오프셋을 사용하여 IORING_BUFFER_REF 구조체의 인스턴스를 만듭니다.
IoRingBufferRefFromPointer 제공된 포인터에서 IORING_BUFFER_REF 구조체의 인스턴스를 만듭니다.
IoRingHandleRefFromHandle 제공된 파일 핸들에서 IORING_HANDLE_REF 구조체의 인스턴스를 만듭니다.
IoRingHandleRefFromIndex 제공된 인덱스에서 IORING_HANDLE_REF 구조체의 인스턴스를 만듭니다.
IsIoRingOpSupported 지정된 I/O 링에 대해 지정된 작업의 지원을 쿼리합니다.

IsReparseTagMicrosoft 재분석 지점 태그가 Microsoft 재분석 지점을 나타내는지 여부를 결정합니다.
IsReparseTagNameSurrogate 태그의 연결된 재분석 지점이 다른 명명된 엔터티(예: 탑재된 폴더)의 서로게이트인지 여부를 결정합니다.
IsWow64GuestMachineSupported 지정된 컴퓨터 아키텍처에서 지원되는 아키텍처(WOW64 아래)를 결정합니다.
IsWow64Process 지정된 프로세스가 WOW64 또는 x64 프로세서의 Intel64에서 실행되고 있는지 여부를 확인합니다.
IsWow64Process2 지정된 프로세스가 WOW64에서 실행 중인지 여부를 확인합니다. 또한 추가 머신 프로세스 및 아키텍처 정보를 반환합니다.
LockFile 호출 프로세스의 배타적 액세스를 위해 지정된 파일을 잠급니다.
LockFileEx 호출 프로세스의 배타적 액세스를 위해 지정된 파일을 잠급니다. 이 함수는 동기적으로 또는 비동기적으로 작동할 수 있으며 배타적 잠금 또는 공유 잠금을 요청할 수 있습니다.
LogTailAdvanceFailure LogTailAdvanceFailure 함수는 로그 클라이언트가 로그 관리 요청을 준수하여 꼬리를 진행할 수 없음을 나타내기 위해 호출됩니다.
LPOVERLAPPED_COMPLETION_ROUTINE ReadFileEx 및 WriteFileEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. 비동기 입출력(I/O) 작업이 완료 또는 취소되고 호출 스레드가 경고 가능한 상태일 때 호출됩니다.
LPPROGRESS_ROUTINE CopyFileEx, MoveFileTransacted 및 MoveFileWithProgress 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
LsnBlockOffset 지정된 LSN에 포함된 섹터 맞춤 블록 오프셋을 반환합니다.

LsnContainer 지정된 LSN에 포함된 논리 컨테이너 ID를 검색합니다.
LsnCreate 컨테이너 ID, 블록 오프셋 및 레코드 시퀀스 번호가 지정된 LSN(로그 시퀀스 번호)을 만듭니다.
LsnRecordSequence 지정된 LSN에 포함된 레코드 시퀀스 번호를 검색합니다.
LZClose LZOpenFile 함수를 사용하여 연 파일을 닫습니다.
LZCopy 원본 파일을 대상 파일에 복사합니다.
LZInit 파일 압축을 해제하는 데 필요한 내부 데이터 구조에 사용할 메모리를 할당한 다음, 파일을 만들고 초기화합니다.
LZOpenFileA 지정된 파일을 만들거나, 열거나, 다시 열거나, 삭제합니다. (ANSI)
LZOpenFileW 지정된 파일을 만들거나, 열거나, 다시 열거나, 삭제합니다. (유니코드)
LZRead 파일에서 지정된 바이트 수(최대)를 읽고 버퍼에 복사합니다.
LZSeek 파일 포인터를 시작 위치에서 지정된 바이트 수만큼 이동합니다.
MAXMEDIALABEL MaxMediaLabel 콜백 함수는 미디어 레이블 라이브러리에서 지원하는 애플리케이션에 대한 미디어 레이블의 최대 크기를 결정합니다.
MountNtmsMedia MountNtmsMedia 함수는 하나 이상의 미디어를 동기적으로 탑재합니다.

MoveFile MoveFile 함수(winbase.h)는 자식을 포함하여 기존 파일 또는 디렉터리를 이동합니다.
MoveFileA 자식을 포함한 기존 파일 또는 디렉터리를 이동합니다. (MoveFileA)
MoveFileFromAppW 자식을 포함한 기존 파일 또는 디렉터리를 이동합니다. 이 함수는 유니버설 Windows 플랫폼 앱 보안 모델을 준수한다는 점을 제외하고 이 함수의 동작은 MoveFile과 동일합니다.
MoveFileW MoveFileW(유니코드) 함수(winbase.h)는 자식을 포함하여 기존 파일 또는 디렉터리를 이동합니다.
MoveFileWithProgressA 자식을 포함한 파일 또는 디렉터리를 이동합니다. 진행률 알림을 받는 콜백 함수를 제공할 수 있습니다. (ANSI)
MoveFileWithProgressW 자식을 포함한 파일 또는 디렉터리를 이동합니다. 진행률 알림을 받는 콜백 함수를 제공할 수 있습니다. (유니코드)
MoveToNtmsMediaPool MoveToNtmsMediaPool 함수는 지정된 매체를 현재 미디어 풀에서 지정된 미디어 풀로 이동합니다.
NetConnectionEnum 서버의 공유 리소스에 대한 모든 연결 또는 특정 컴퓨터에서 설정된 모든 연결을 나열합니다.
NetFileClose 리소스를 강제로 닫습니다. 이 함수는 오류가 다른 수단으로 닫을 수 없는 경우에 사용할 수 있습니다. NetFileClose는 파일을 닫기 전에 클라이언트 시스템에 캐시된 데이터를 파일에 쓰지 않으므로 주의해서 사용해야 합니다.
NetFileEnum 지정된 매개 변수에 따라 서버의 열려 있는 파일 중 일부 또는 모두에 대한 정보를 반환합니다.
NetFileGetInfo 서버 리소스의 특정 열기에 대한 정보를 검색합니다.

NetSessionDel 서버와 워크스테이션 간의 네트워크 세션을 종료합니다.
NetSessionEnum 서버에 설정된 세션에 대한 정보를 제공합니다.
NetSessionGetInfo 특정 서버와 워크스테이션 간에 설정된 세션에 대한 정보를 검색합니다.
NetShareAdd 서버 리소스를 공유합니다.
NetShareCheck 서버가 디바이스를 공유하고 있는지 여부를 확인합니다.
NetShareDel 공유 리소스에 대한 모든 연결이 끊어지도록 서버의 공유 리소스 목록에서 공유 이름을 삭제합니다.
NetShareDelEx 서버의 공유 리소스 목록에서 공유 이름을 삭제하여 해당 공유에 대한 모든 연결을 끊습니다.
NetShareEnum 서버의 각 공유 리소스에 대한 정보를 검색합니다.
NetShareGetInfo 서버의 특정 공유 리소스에 대한 정보를 검색합니다.
NetShareSetInfo 공유 리소스의 매개 변수를 설정합니다.
NetStatisticsGet 서비스에 대한 운영 통계를 검색합니다. 현재는 워크스테이션 및 서버 서비스만 지원됩니다.
다음 열거형 시퀀스의 다음 사용자 할당량 항목 수를 검색합니다.

OnUserNameChanged 사용자의 SID가 비동기적으로 해결될 때마다 클라이언트의 연결 싱크를 알 수 있습니다.
OpenEncryptedFileRawA 파일을 백업(내보내기) 또는 복원(가져오기)하기 위해 암호화된 파일을 엽니다. (ANSI)
OpenEncryptedFileRawW 파일을 백업(내보내기) 또는 복원(가져오기)하기 위해 암호화된 파일을 엽니다. (유니코드)
OpenEnlistment 기존 인리스트먼트 개체를 열고 인리스트먼트에 대한 핸들을 반환합니다.
OpenFile 파일을 만들거나, 열거나, 다시 열거나, 삭제합니다.
OpenFileById 지정된 식별자와 일치하는 파일을 엽니다.
OpenNtmsNotification OpenNtmsNotification 함수는 지정된 형식의 개체에 대한 RSM 개체 변경 알림을 수신하는 채널을 엽니다.
OpenNtmsSessionA OpenNtmsSession 함수는 RSM 서버로 세션을 설정합니다. (ANSI)
OpenNtmsSessionW OpenNtmsSession 함수는 RSM 서버로 세션을 설정합니다. (유니코드)
OpenResourceManager 기존 RM(리소스 관리자)을 엽니다.
OpenTransaction 기존 트랜잭션을 엽니다.
OpenTransactionManager 기존 트랜잭션 관리자를 엽니다. (OpenTransactionManager)

OpenTransactionManagerById 기존 트랜잭션 관리자를 엽니다. (OpenTransactionManagerById)
PCOPYFILE2_PROGRESS_ROUTINE CopyFile2 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PFE_EXPORT_FUNC ReadEncryptedFileRaw와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PFE_IMPORT_FUNC WriteEncryptedFileRaw와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. 시스템은 백업 파일의 일부 데이터를 검색할 때마다 ImportCallback을 한 번 이상 호출합니다.
PLOG_FULL_HANDLER_CALLBACK LOG_FULL_HANDLER_CALLBACK 함수는 HandleLogFull에 대한 호출이 완료되었다는 알림을 받는 애플리케이션 정의 콜백 함수입니다.
PLOG_TAIL_ADVANCE_CALLBACK LOG_TAIL_ADVANCE_CALLBACK 함수는 로그 테일을 진행하는 애플리케이션 정의 콜백 함수입니다. 콜백은 로그 관리를 위해 등록하는 스레드에서 APC(비동기 프로시저 호출)의 컨텍스트에서 호출됩니다.
PLOG_UNPINNED_CALLBACK LOG_UNPINNED_CALLBACK 함수는 로그가 고정 해제되었다는 알림을 받는 애플리케이션 정의 콜백 함수입니다.
PopIoRingCompletion 사용할 수 있는 경우 완료 큐에서 단일 항목을 팝합니다.
PostQueuedCompletionStatus I/O 완료 패킷을 I/O 완료 포트에 게시합니다.
PrepareComplete RM(리소스 관리자)이 지정된 트랜잭션에 대해 커밋 또는 중단 작업이 성공하도록 보장하는 데 필요한 모든 처리를 완료했음을 나타냅니다.

PrepareEnlistment 이 인리스트먼트 핸들과 연결된 트랜잭션을 준비합니다. 이 함수는 통신 리소스 관리자(뛰어난 트랜잭션 관리자라고도 함)에서 사용됩니다.
PrepareLogArchive 보관을 위한 물리적 로그를 준비합니다.
PrePrepareComplete 이제 다른 리소스 관리자가 준비 작업을 시작할 수 있도록 이 리소스 관리자가 준비 작업을 완료했음을 표시합니다.
PrePrepareEnlistment 이 인리스트먼트 핸들과 연결된 트랜잭션을 미리 준비합니다. 이 함수는 통신 리소스 관리자(뛰어난 트랜잭션 관리자라고도 함)에서 사용됩니다.
QueryDosDeviceA MS-DOS 디바이스 이름에 대한 정보를 검색합니다. (QueryDosDeviceA)
QueryDosDeviceW MS-DOS 디바이스 이름에 대한 정보를 검색합니다. (QueryDosDeviceW)
QueryIoRingCapabilities OS에서 IORING에 지원되는 기능을 쿼리합니다.
QueryLogPolicy QueryLogPolicy 함수를 사용하면 지정된 로그에 대해 설치된 정책을 가져올 수 있습니다.
QueryRecoveryAgentsOnEncryptedFile 복구 에이전트 목록을 검색하여 지정된 파일을 찾습니다.
QueryUsersOnEncryptedFile 사용자 목록을 검색하여 지정된 파일을 찾습니다.
ReadDirectoryChangesExW 지정된 디렉터리 내의 변경 내용을 설명하는 정보를 검색합니다. 이 정보에는 해당 정보 유형이 지정된 경우 확장된 정보가 포함될 수 있습니다.

ReadDirectoryChangesW 지정된 디렉터리 내의 변경 내용을 설명하는 정보를 검색합니다.
ReadEncryptedFileRaw 암호화된 파일을 백업(내보내기)합니다.
ReadFile 지정된 파일 또는 I/O(입력/출력) 디바이스에서 데이터를 읽습니다. 디바이스에서 파일 포인터를 지원하는 경우 파일 포인터가 지정한 위치에서 읽기가 발생합니다.
ReadFileEx 지정된 파일 또는 I/O(입력/출력) 디바이스에서 데이터를 읽습니다. 읽기가 완료 또는 취소되고 호출 스레드가 경고 대기 상태에 있을 때 완료 상태를 비동기적으로 보고하고, 지정된 완료 루틴을 호출합니다.
ReadFileScatter 파일에서 데이터를 읽고 버퍼 배열에 저장합니다.
ReadLogArchiveMetadata 메타데이터의 보관 보기 범위를 지정된 버퍼에 복사합니다.
ReadLogNotification 로그 관리자에서 알림을 검색합니다. 알림을 사용할 수 있는 경우 로그 관리자에서 대기 중인 알림을 즉시 검색합니다. 그렇지 않으면 알림이 생성될 때까지 요청이 보류 중인 상태로 유지됩니다.
ReadLogRecord 세 가지 모드 중 하나에서 지정된 LSN(로그 시퀀스 번호)에서 읽기 시퀀스를 시작하고 지정된 로그 레코드의 첫 번째 및 읽기 컨텍스트를 반환합니다.
ReadLogRestartArea WriteLogRestartArea의 마샬링 영역과 연결된 로그에 성공적으로 기록된 마지막 다시 시작 영역을 반환합니다.
ReadNextLogRecord ReadLogRecord 또는 ReadLogRestartArea를 호출하여 시작되는 시퀀스에서 다음 레코드를 읽습니다.

ReadOnlyEnlistment 지정된 인리스트먼트를 읽기 전용 인리스트먼트로 변환하도록 요청합니다. 읽기 전용 인리스트먼트는 트랜잭션 결과에 참여할 수 없으며 복구를 위해 지속적으로 기록되지 않습니다.
ReadPreviousLogRestartArea 읽기 컨텍스트 pvReadContext에 지정된 현재 다시 시작 레코드를 기준으로 하는 이전 로그 다시 시작 영역을 읽습니다. 이 읽기 컨텍스트는 ReadLogRestartArea를 호출하여 이전에 만든 컨텍스트입니다.
RecoverEnlistment 인리스트먼트의 상태를 복구합니다.
RecoverResourceManager 로그 파일에서 리소스 관리자의 상태를 복구합니다.
RecoverTransactionManager 로그 파일에서 트랜잭션 관리자의 상태를 복구합니다.
RegisterForLogWriteNotification RegisterForLogWriteNotification 함수는 로그 쓰기 알림을 사용하거나 사용하지 않도록 설정하기 위해 관리 로그 클라이언트에서 호출됩니다.
RegisterManageableLogClient 로그 관리자에 클라이언트를 등록합니다. 클라이언트는 콜백을 사용하여 알림을 받을지 또는 ReadLogNotification을 사용하여 검색을 위해 알림을 큐에 대기할지를 지정할 수 있습니다.
ReleaseNtmsCleanerSlot ReleaseNtmsCleanerSlot 함수는 정리 카트리지에 대한 기존 슬롯 예약을 제거합니다. 그런 다음 슬롯을 데이터 카트리지에 사용할 수 있습니다.
Remove 일괄 처리 목록에서 IDiskQuotaUser 포인터를 제거합니다.
Removeall 일괄 처리 목록에서 모든 IDiskQuotaUser 포인터를 제거합니다.
RemoveDirectoryA 기존 빈 디렉터리를 삭제합니다. (ANSI)

RemoveDirectoryFromAppW 기존 빈 디렉터리를 삭제합니다. 이 함수는 유니버설 Windows 플랫폼 앱 보안 모델을 준수한다는 점을 제외하고 이 함수의 동작은 RemoveDirectory와 동일합니다.
RemoveDirectoryTransactedA 트랜잭션 작업으로 기존 빈 디렉터리를 삭제합니다. (ANSI)
RemoveDirectoryTransactedW 트랜잭션 작업으로 기존 빈 디렉터리를 삭제합니다. (유니코드)
RemoveDirectoryW 기존 빈 디렉터리를 삭제합니다. (유니코드)
RemoveLogContainer 전용 또는 멀티플렉싱된 로그 핸들과 연결된 로그에서 하나의 컨테이너를 제거합니다.
RemoveLogContainerSet 전용 또는 멀티플렉싱된 로그 핸들과 연결된 로그에서 여러 컨테이너를 제거합니다.
RemoveLogPolicy 지정된 정책을 기본 동작으로 다시 설정합니다.
RemoveUsersFromEncryptedFile 지정된 파일에서 지정된 인증서 해시를 제거합니다.
RenameTransactionManager TM(트랜잭션 관리자) 개체의 이름을 바꿉니다. 이 함수는 명명된 TM 핸들에서만 사용할 수 있습니다.
ReOpenFile 액세스 권한, 공유 모드 및 플래그가 다른 지정된 파일 시스템 개체를 다시 엽니다.
ReplaceFileA 한 파일을 다른 파일로 바꾸며, 원본 파일의 백업 복사본을 만드는 옵션을 제공합니다. (ANSI)

ReplaceFileFromAppW 한 파일을 다른 파일로 바꾸며, 원본 파일의 백업 복사본을 만드는 옵션을 제공합니다. 이 함수는 유니버설 Windows 플랫폼 앱 보안 모델을 준수한다는 점을 제외하고 이 함수의 동작은 ReplaceFile과 동일합니다.
ReplaceFileW 한 파일을 다른 파일로 바꾸며, 원본 파일의 백업 복사본을 만드는 옵션을 제공합니다. (유니코드)
ReserveAndAppendLog 로그 버퍼의 공간을 예약하거나 로그 레코드를 로그에 추가하거나 둘 다 수행합니다. 함수는 atomic입니다.
ReserveAndAppendLogAligned 로그 버퍼의 공간을 예약하거나 로그 레코드 또는 둘 다에 로그 레코드를 추가합니다.
ReserveNtmsCleanerSlot ReserveNtmsCleanerSlot 함수는 드라이브 클리너 카트리지에 대한 라이브러리 단위의 단일 슬롯을 예약합니다.
Reset 열거형 시퀀스를 사용자 할당량 항목 집합의 시작 부분으로 다시 설정합니다.
RollbackComplete RM(리소스 관리자)이 트랜잭션 롤백을 성공적으로 완료했음을 나타냅니다.
RollbackEnlistment 인리스트먼트와 연결된 지정된 트랜잭션을 롤백합니다. 이 함수는 읽기 전용 인리스트먼트에 대해 호출할 수 없습니다.
RollbackTransaction 지정된 트랜잭션을 롤백할 것을 요청합니다.
RollbackTransactionAsync 지정된 트랜잭션을 롤백할 것을 요청합니다. 이 함수는 비동기적으로 반환됩니다.
RollforwardTransactionManager 지정된 가상 클록 값으로만 정보를 복구합니다.

SatisfyNtmsOperatorRequest SatisfyNtmsOperatorRequest 함수는 지정된 RSM 연산자 요청을 완료합니다.
ScanLogContainers 로그 컨테이너를 열거합니다. 이 함수를 반복적으로 호출하여 모든 로그 컨테이너를 반복합니다.
SearchPathA 지정된 경로에서 지정된 파일을 검색합니다. (ANSI)
SearchPathW 지정된 경로에서 지정된 파일을 검색합니다. (유니코드)
SetCurrentDirectory 현재 프로세스의 현재 디렉터리를 변경합니다.
SetDefaultQuotaLimit 기본 할당량 제한을 수정합니다.
SetDefaultQuotaThreshold 기본 경고 임계값을 수정합니다.
SetEndOfFile 지정된 파일의 실제 파일 크기를 파일 포인터의 현재 위치로 설정합니다.
SetEndOfLog 이 함수는 더 이상 사용되지 않습니다. 대신 TruncateLog를 사용합니다.
SetEnlistmentRecoveryInformation KTM에서 복구 데이터의 불투명한 사용자 정의 구조를 설정합니다.
SetFileApisToANSI 파일 I/O 함수가 현재 프로세스에 ANSI 문자 집합 코드 페이지를 사용하게 합니다.
SetFileApisToOEM 프로세스에 대한 파일 I/O 함수가 OEM 문자 집합 코드 페이지를 사용하게 합니다.

SetFileAttributesA 파일 또는 디렉터리의 특성을 설정합니다. (ANSI)
SetFileAttributesFromAppW 파일 또는 디렉터리의 특성을 설정합니다. 이 함수는 유니버설 Windows 플랫폼 앱 보안 모델을 준수한다는 점을 제외하고 이 함수의 동작은 SetFileAttributes와 동일합니다.
SetFileAttributesTransactedA 트랜잭션 작업으로 파일 또는 디렉터리의 특성을 설정합니다. (ANSI)
SetFileAttributesTransactedW 트랜잭션 작업으로 파일 또는 디렉터리의 특성을 설정합니다. (유니코드)
SetFileAttributesW 파일 또는 디렉터리의 특성을 설정합니다. (유니코드)
SetFileBandwidthReservation 지정된 파일 스트림의 대역폭을 예약하도록 요청합니다. 예약은 지정된 파일 핸들의 I/O 요청에 대한 밀리초 단위 바이트 수로 지정됩니다.
SetFileCompletionNotificationModes 파일 핸들의 알림 모드를 설정하여 완료 알림이 지정된 파일에 대해 작동하는 방식을 지정할 수 있습니다.
SetFileInformationByHandle 지정된 파일의 파일 정보를 설정합니다.
SetFileIoOverlappedRange 가상 주소 범위를 지정된 파일 핸들과 연결합니다.
SetFilePointer 지정된 파일의 파일 포인터를 이동합니다. (SetFilePointer)
SetFilePointerEx 지정된 파일의 파일 포인터를 이동합니다. (SetFilePointerEx)

SetFileShortNameA 지정된 파일의 짧은 이름을 설정합니다. (ANSI)
SetFileShortNameW 지정된 파일의 짧은 이름을 설정합니다. (유니코드)
SetFileValidData 지정된 파일의 유효한 데이터 길이를 설정합니다. 이 함수는 매우 제한된 시나리오에서 유용합니다. 자세한 내용은 주의 섹션을 참조하세요.
SetIoRingCompletionEvent 완료 큐 이벤트를 IORING에 등록합니다.
SetLogArchiveMode 지정된 로그에 대한 로그 보관 지원을 사용하거나 사용하지 않도록 설정합니다.
SetLogArchiveTail 보관 가능한 로그의 마지막 LSN(보관 로그 시퀀스 번호) 또는 보관 꼬리를 설정합니다.
SetLogFileSizeWithPolicy 설치된 정책의 상태에 따라 로그에서 컨테이너를 추가하거나 삭제합니다.
SetNtmsDeviceChangeDetection SetNtmsDeviceChangeDetection 함수는 변경 검색을 위해 하나 이상의 대상 디바이스를 설정합니다.
SetNtmsMediaComplete SetNtmsMediaComplete 함수는 논리 미디어의 일부를 완료로 표시합니다.
SetNtmsObjectAttributeA SetNtmsObjectAttribute 함수는 지정된 RSM 개체에 확장 특성(명명된 프라이빗 데이터)을 만듭니다. (ANSI)
SetNtmsObjectAttributeW SetNtmsObjectAttribute 함수는 지정된 RSM 개체에 확장 특성(명명된 프라이빗 데이터)을 만듭니다. (유니코드)

SetNtmsObjectInformation SetNtmsObjectInformation 함수는 지정된 개체의 정보 구조를 변경합니다. (SetNtmsObjectInformation)
SetNtmsObjectInformationA SetNtmsObjectInformation 함수는 지정된 개체의 정보 구조를 변경합니다. (SetNtmsObjectInformationA)
SetNtmsObjectInformationW SetNtmsObjectInformationW(유니코드) 함수는 지정된 개체의 정보 구조를 변경합니다. (SetNtmsObjectInformationW)
SetNtmsObjectSecurity SetNtmsObjectSecurity 함수는 지정된 RSM 개체에 대한 보안 설명자를 씁니다.
SetNtmsRequestOrder SetNtmsRequestOrder 함수는 지정된 요청이 라이브러리 큐에서 처리되는 순서를 설정합니다.
SetNtmsUIOptionsA SetNtmsUIOptions 함수는 지정된 형식의 UI가 지정된 개체에 대해 전달되는 컴퓨터 이름 목록을 수정합니다. (ANSI)
SetNtmsUIOptionsW SetNtmsUIOptions 함수는 지정된 형식의 UI가 지정된 개체에 대해 전달되는 컴퓨터 이름 목록을 수정합니다. (유니코드)
SetQuotaLimit 볼륨에 대한 사용자의 할당량 제한 값을 설정합니다.
SetQuotaLogFlags 볼륨에 대한 사용자 관련 할당량 이벤트의 로깅을 제어합니다.
SetQuotaState 할당량 시스템의 상태를 설정합니다.
SetQuotaThreshold 볼륨에 대한 사용자의 경고 임계값을 설정합니다.

SetResourceManagerCompletionPort 지정된 I/O 완료 포트를 지정된 RM(리소스 관리자)과 연결합니다. 이 포트는 RM에 대한 모든 알림을 받습니다.
SetSearchPathMode 파일을 찾을 때 SearchPath 함수가 사용하는 프로세스별 모드를 설정합니다.
SetTransactionInformation 지정된 트랜잭션에 대한 트랜잭션 정보를 설정합니다.
SetUserFileEncryptionKey 사용자의 현재 키를 지정된 인증서로 설정합니다.
SetVolumeLabelA 파일 시스템 볼륨의 레이블을 설정합니다. (ANSI)
SetVolumeLabelW 파일 시스템 볼륨의 레이블을 설정합니다. (유니코드)
SetVolumeMountPointA 볼륨을 드라이브 문자 또는 다른 볼륨의 디렉터리에 연결합니다. (ANSI)
SetVolumeMountPointW 볼륨을 드라이브 문자 또는 다른 볼륨의 디렉터리에 연결합니다. (유니코드)
ShutdownNameResolution SID(사용자 보안 식별자)를 사용자 이름으로 변환합니다.
SinglePhaseReject RM(리소스 관리자)이 단일 단계 요청을 거부했음을 나타냅니다. TM(트랜잭션 관리자)이 이 호출을 수신하면 2단계 커밋을 시작하고 등록된 모든 RM에 준비 요청을 보냅니다.
Skip 열거형 시퀀스의 다음 사용자 할당량 항목 수를 건너뜁니다.

SubmitIoRing 생성되었지만 아직 제출되지 않은 항목을 커널의 큐에 제출하고 필요에 따라 작업 집합이 완료될 때까지 기다립니다.
SubmitNtmsOperatorRequestA SubmitNtmsOperatorRequest 함수는 RSM 연산자 요청을 제출합니다. (ANSI)
SubmitNtmsOperatorRequestW SubmitNtmsOperatorRequest 함수는 RSM 연산자 요청을 제출합니다. (유니코드)
SwapNtmsMedia SwapNtmsMedia 함수는 지정된 두 LMID와 연결된 측면을 교환합니다. 지정된 LMID는 동일한 미디어 풀에 있어야 합니다.
TerminateLogArchive PrepareLogArchive에서 로그 보관 컨텍스트에 대해 원래 할당된 시스템 리소스의 할당을 취소합니다.
TerminateReadLog 읽기 컨텍스트를 종료합니다. 이 함수는 지정된 읽기 컨텍스트와 연결된 시스템 할당 리소스를 해제합니다. 이 함수를 호출한 후 로그 레코드를 읽으려고 시도하지 마세요. 확정되지 않은 결과를 받게 됩니다.
TruncateLog 로그를 자립니다. 함수는 로그의 끝을 지정된 값으로 설정합니다.
TxfGetThreadMiniVersionForCreate 후속 만들기가 열리도록 설정된 MiniVersion을 반환합니다.
TxfLogCreateFileReadContext 복제 레코드를 읽는 데 사용할 컨텍스트를 만듭니다.
TxfLogCreateRangeReadContext 복제 레코드를 읽는 데 필요한 컨텍스트를 만듭니다.
TxfLogDestroyReadContext TxfLogCreateFileReadContext 함수에 의해 만들어진 읽기 컨텍스트를 닫습니다.

TxfLogReadRecords 로그에서 다시 실행 레코드를 읽습니다.
TxfSetThreadMiniVersionForCreate 후속 만들기를 열어야 하는 MiniVersion을 설정합니다.
UnlockFile 열려 있는 파일에서 지역 잠금을 해제합니다.
UnlockFileEx 지정된 파일에서 영역 잠금을 해제합니다. 이 함수는 동기적으로 또는 비동기적으로 작동할 수 있습니다.
UpdateNtmsOmidInfo UpdateNtmsOmidInfo 함수는 새로 할당된 매체에 쓴 직후 레이블 정보로 RSM 데이터베이스를 업데이트합니다.
ValidateLog 로그 보관 전과 로그 복원 후 로그 메타데이터 및 데이터의 일관성을 확인합니다.
WaitForNtmsNotification WaitForNtmsNotification 함수는 다음 개체 변경 알림을 기다립니다.
WaitForNtmsOperatorRequest WaitForNtmsOperatorRequest 함수는 지정된 RSM 연산자 요청을 기다립니다.
WofEnumEntries 지정된 볼륨에 대해 지정된 공급자의 모든 데이터 원본을 열거합니다.
WofEnumEntryProc WofEnumEntries 호출에 대한 응답으로 각 데이터 원본에 대해 호출되는 콜백 함수입니다.
WofEnumFilesProc WIM 파일과 같은 외부 데이터 원본이 지원하는 각 파일에 대해 호출되는 콜백 함수입니다.
WofFileEnumFiles 지정된 볼륨에서 지정된 압축 알고리즘을 사용하여 압축되는 모든 파일을 열거합니다.

WofGetDriverVersion 특정 공급자를 지원하는 데 사용되는 드라이버 버전을 쿼리하는 데 사용됩니다.
WofIsExternalFile 실제 파일 또는 시스템 데이터 공급자가 파일을 백업하는지 확인하는 데 사용되며, 필요에 따라 파일에 대한 공급자 또는 추가 데이터를 나타냅니다.
WofSetFileDataLocation 파일을 물리적 파일 백업에서 시스템 데이터 공급자 백업으로 변경하는 데 사용됩니다.
WofShouldCompressBinaries 특정 볼륨에서 압축을 사용해야 하는지, 사용해야 한다면 어떤 압축 알고리즘을 사용해야 하는지 나타냅니다.
WofWimAddEntry WIM 내에 저장되는 볼륨에 파일을 만들 수 있도록 단일 WIM 데이터 원본을 볼륨에 추가합니다.
WofWimEnumFiles 지정된 볼륨의 지정된 WIM 데이터 원본이 백업하는 모든 파일을 열거합니다.
WofWimRemoveEntry 볼륨의 백업 파일에서 단일 WIM 데이터 원본을 제거합니다.
WofWimSuspendEntry 볼륨이 다시 탑재되거나 데이터 원본이 WofWimUpdateEntry로 업데이트될 때까지 볼륨의 백업 파일에서 WIM 데이터 원본을 일시적으로 제거합니다.
WofWimUpdateEntry 다른 WIM 파일 위치를 가리키도록 WIM 항목을 업데이트합니다.
Wow64DisableWow64FsRedirection 호출 스레드에 파일 시스템 리디렉션을 사용하지 않도록 설정합니다. 파일 시스템 리디렉션은 기본적으로 사용하도록 설정됩니다.
Wow64EnableWow64FsRedirection 호출 스레드에 파일 시스템 리디렉션을 사용하도록 설정합니다.

Wow64GetThreadContext 스레드 컨텍스트를 검색합니다.
Wow64RevertWow64FsRedirection 호출 스레드에 대한 파일 시스템 리디렉션을 복원합니다.
Wow64SetThreadContext 스레드 컨텍스트를 설정합니다.
Wow64SetThreadDefaultGuestMachine 스레드를 지정된 컴퓨터 아키텍처로 설정합니다.
Wow64SuspendThread 지정된 WOW64 스레드를 일시 중단합니다.
WriteEncryptedFileRaw 암호화된 파일을 복원(가져오기)합니다.
WriteFile 지정된 파일 또는 I/O(입출력) 디바이스에 데이터를 씁니다.
WriteFileEx 지정된 파일 또는 I/O(입출력) 디바이스에 데이터를 씁니다. 쓰기가 완료 또는 취소되고 호출 스레드가 경고 대기 상태에 있을 때 완료 상태를 비동기적으로 보고하고, 지정된 완료 루틴을 호출합니다.
WriteFileGather 버퍼 배열에서 데이터를 검색하고 데이터를 파일에 씁니다.
WriteLogRestartArea 새 클라이언트 다시 시작 영역을 로그에 추가하고 필요에 따라 로그의 LSN(기본 로그 시퀀스 번호)을 앞으로 이동합니다.

인터페이스

IDiskQuotaControl 단일 NTFS 파일 시스템 볼륨의 디스크 할당량 기능을 제어합니다.
IDiskQuotaEvents 할당량 관련 이벤트 알림을 받습니다.
IDiskQuotaUser 볼륨 할당량 정보 파일의 단일 사용자 할당량 항목을 나타냅니다.
IDiskQuotaUserBatch 컨테이너에 여러 할당량 사용자 개체를 추가한 다음, 단일 호출에서 업데이트를 위해 제출됩니다.
IEnumDiskQuotaUsers 볼륨의 사용자 할당량 항목을 열거합니다.

구조체

BOOT_AREA_INFO FSCTL_GET_BOOT_AREA_INFO 제어 코드에 대한 출력을 포함합니다.
BY_HANDLE_FILE_INFORMATION GetFileInformationByHandle 함수가 검색하는 정보를 포함합니다.
CLFS_MGMT_NOTIFICATION CLFS_MGMT_NOTIFICATION 구조는 클라이언트가 수신하는 알림에 대한 정보를 지정합니다.
CLFS_MGMT_POLICY CLFS_MGMT_POLICY 구조는 CLFS(Common Log File System) 관리 정책을 지정합니다. PolicyType 멤버는 정책에 사용되는 멤버를 지정합니다.
CLFS_NODE_ID 노드 식별자를 나타냅니다.

CLS_ARCHIVE_DESCRIPTOR GetNextLogArchiveExtent 함수에서 로그 보관 익스텐트에 대한 정보를 반환하는 데 사용됩니다.
CLS_CONTAINER_INFORMATION 컨테이너에 대한 일반적인 정보를 설명합니다.
CLS_INFORMATION 로그에 대한 일반적인 정보를 설명합니다.
CLS_IO_STATISTICS GetLogIoStatistics에서 보고하는 통계를 정의합니다.
CLS_IO_STATISTICS_HEADER 로그의 I/O 성능 카운터를 정의하는 GetLogIoStatistics 함수에서 검색한 정보에 대한 헤더입니다.
CLS_LSN 유효한 로그 주소를 나타냅니다.
CLS_WRITE_ENTRY 로그 레코드의 일부가 될 사용자 버퍼와 해당 길이를 포함합니다.
CONNECTION_INFO_0 연결의 ID 번호를 포함합니다.
CONNECTION_INFO_1 연결의 ID 번호, 열려 있는 파일 수, 연결 시간, 연결의 사용자 수 및 연결 유형을 포함합니다.
COPYFILE2_EXTENDED_PARAMETERS CopyFile2 함수에 대한 확장 매개 변수를 포함합니다.
COPYFILE2_MESSAGE 보류 중인 복사 작업에 대한 정보와 함께 CopyFile2ProgressRoutine 콜백 함수에 전달됩니다.
CREATE_DISK IOCTL_DISK_CREATE_DISK 제어 코드에서 GPT(GUID 파티션 테이블), MBR(마스터 부트레코드) 또는 원시 디스크를 초기화하는 데 사용하는 정보를 포함합니다.

CREATE_DISK_GPT IOCTL_DISK_CREATE_DISK 제어 코드에서 GPT(GUID 파티션 테이블) 디스크를 초기화하는 데 사용하는 정보를 포함합니다.
CREATE_DISK_MBR IOCTL_DISK_CREATE_DISK 제어 코드dptj MBR(마스터 부팅 레코드) 디스크를 초기화하는 데 사용하는 정보를 포함합니다.
CREATE_USN_JOURNAL_DATA USN(업데이트 시퀀스 번호) 변경 저널을 설명하는 정보를 포함합니다.
CREATEFILE2_EXTENDED_PARAMETERS CreateFile2에 대한 선택적 확장 매개 변수를 포함합니다.
CSV_CONTROL_PARAM CSV 제어 작업의 형식을 나타냅니다.
CSV_IS_OWNED_BY_CSVFS 볼륨을 CSVFS에서 소유하는지 여부를 결정하는 FSCTL_IS_VOLUME_OWNED_BYCSVFS 제어 코드에 대한 출력을 포함합니다.
CSV_NAMESPACE_INFO 파일에 대한 네임스페이스 정보를 가져오는 FSCTL_IS_CSV_FILE 제어 코드에 대한 출력을 포함합니다.
CSV_QUERY_FILE_REVISION 스트림의 파일이 수정되었는지 여부에 대한 정보를 포함합니다.
CSV_QUERY_MDS_PATH CSV에서 MDS와 통신하는 데 사용하는 경로를 포함합니다.
CSV_QUERY_REDIRECT_STATE 스트림의 파일이 리디렉션되었는지 여부에 대한 정보를 포함합니다.
CSV_QUERY_VETO_FILE_DIRECT_IO_OUTPUT 볼륨이 리디렉션 모드인 이유에 대한 문제 해결 정보를 포함합니다.

DELETE_USN_JOURNAL_DATA FSCTL_DELETE_USN_JOURNAL 제어 코드를 사용한 USN(업데이트 시퀀스 번호) 변경 저널 삭제에 대한 정보를 포함합니다.
DEVICE_COPY_OFFLOAD_DESCRIPTOR 스토리지 디바이스에 대한 복사 오프로드 기능을 포함합니다.
DEVICE_LB_PROVISIONING_DESCRIPTOR 스토리지 디바이스에 대한 씬 프로비저닝 기능을 포함합니다.
DEVICE_POWER_DESCRIPTOR DEVICE_POWER_DESCRIPTOR 구조는 스토리지 디바이스의 전원 기능을 설명합니다.
DEVICE_SEEK_PENALTY_DESCRIPTOR IOCTL_STORAGE_QUERY_PROPERTY 요청과 함께 디바이스에 대한 검색 페널티 설명자 데이터를 검색하는 데 사용됩니다.
DEVICE_TRIM_DESCRIPTOR IOCTL_STORAGE_QUERY_PROPERTY 요청과 함께 디바이스에 대한 트리밍 설명자 데이터를 검색하는 데 사용됩니다.
DEVICE_WRITE_AGGREGATION_DESCRIPTOR 시스템에서 사용하도록 예약되었습니다. (DEVICE_WRITE_AGGREGATION_DESCRIPTOR)
DISK_CACHE_INFORMATION 디스크 캐시에 대한 정보를 제공합니다.
DISK_DETECTION_INFO 검색된 드라이브 매개 변수를 포함합니다.
DISK_EX_INT13_INFO 확장된 Int13 드라이브 매개 변수를 포함합니다.
DISK_EXTENT 디스크 범위를 나타냅니다.

DISK_GEOMETRY 디스크 디바이스 및 미디어의 기하학을 설명합니다.
DISK_GEOMETRY_EX 디스크 디바이스 및 미디어의 확장된 기하학을 설명합니다.
DISK_GROW_PARTITION 파티션 크기를 늘리는 데 사용되는 정보를 포함합니다.
DISK_INT13_INFO 표준 Int13 드라이브 기하학 매개 변수를 포함합니다.
DISK_PARTITION_INFO 디스크 파티션 정보를 포함합니다.
DISK_PERFORMANCE 디스크 성능 정보를 제공합니다.
DISK_SPACE_INFORMATION DISK_SPACE_INFORMATION 구조에는 특정 볼륨의 디스크 공간에 대한 정보가 포함됩니다.
DISKQUOTA_USER_INFORMATION 사용자별 할당량 정보를 나타냅니다.
DRIVE_LAYOUT_INFORMATION 드라이브의 파티션에 대한 정보를 포함합니다.
DRIVE_LAYOUT_INFORMATION_EX 드라이브의 파티션에 대한 확장된 정보를 포함합니다.
DRIVE_LAYOUT_INFORMATION_GPT 드라이브의 GPT(GUID 파티션 테이블) 파티션에 대한 정보를 포함합니다.
DRIVE_LAYOUT_INFORMATION_MBR 드라이브의 MBR(마스터 부트 레코드) 파티션에 대한 정보를 제공합니다.

DUPLICATE_EXTENTS_DATA 블록 복제 작업을 수행하는 FSCTL_DUPLICATE_EXTENTS 제어 코드에 대한 매개 변수를 포함합니다.
EFS_CERTIFICATE_BLOB 인증서를 포함합니다.
EFS_HASH_BLOB 인증서 해시를 포함합니다.
ENCRYPTION_CERTIFICATE 인증서와 해당 소유자의 SID를 포함합니다.
ENCRYPTION_CERTIFICATE_HASH 인증서 해시를 포함하고 인증서에 대한 정보를 표시합니다.
ENCRYPTION_CERTIFICATE_HASH_LIST 인증서 해시 목록을 포함합니다.
ENCRYPTION_CERTIFICATE_LIST 인증서 목록을 포함합니다.
EXFAT_STATISTICS exFAT 파일 시스템의 통계 정보를 포함합니다.
FAT_STATISTICS FAT 파일 시스템의 통계 정보를 포함합니다.
FILE_ALIGNMENT_INFO 파일에 대한 맞춤 정보를 포함합니다.
FILE_ALLOCATED_RANGE_BUFFER 파일에서 바이트 범위를 나타냅니다.
FILE_ALLOCATION_INFO 파일에 할당되어야 하는 총 바이트 수를 포함합니다.

FILE_ATTRIBUTE_TAG_INFO 요청한 파일 특성 정보를 수신합니다. 모든 핸들에 사용됩니다.
FILE_BASIC_INFO 파일에 대한 기본 정보를 포함합니다. 파일 핸들에 사용됩니다.
FILE_COMPRESSION_INFO 파일 압축 정보를 수신합니다.
FILE_DISPOSITION_INFO 파일을 삭제해야 하는지 여부를 나타냅니다. 모든 핸들에 사용됩니다.
FILE_END_OF_FILE_INFO 파일의 끝을 설정해야 하는 지정된 값을 포함합니다.
FILE_FULL_DIR_INFO 파일에 대한 디렉터리 정보를 포함합니다. (FILE_FULL_DIR_INFO)
FILE_ID_128 128비트 파일 식별자를 정의합니다.
FILE_ID_BOTH_DIR_INFO 지정된 디렉터리의 파일에 대한 정보를 포함합니다.
FILE_ID_DESCRIPTOR 사용 중인 ID 유형을 지정합니다.
FILE_ID_EXTD_DIR_INFO 파일에 대한 식별 정보를 포함합니다. (FILE_ID_EXTD_DIR_INFO)
FILE_ID_INFO 파일에 대한 식별 정보를 포함합니다. (FILE_ID_INFO)
FILE_INFO_2 파일, 디바이스 또는 파이프의 ID 번호를 포함합니다.

FILE_INFO_3 파일, 디바이스 및 파이프에 대한 식별 번호 및 기타 관련 정보를 포함합니다.
FILE_IO_PRIORITY_HINT_INFO 파일 I/O 작업의 우선 순위 힌트를 지정합니다.
FILE_LEVEL_TRIM FSCTL_FILE_LEVEL_TRIM 제어 코드에 대한 입력으로 사용됩니다.
FILE_LEVEL_TRIM_OUTPUT FSCTL_FILE_LEVEL_TRIM 제어 코드에 대한 출력으로 사용됩니다.
FILE_LEVEL_TRIM_RANGE 잘라낼 파일의 범위를 지정합니다.
FILE_MAKE_COMPATIBLE_BUFFER 현재 세션을 닫을 디스크를 지정합니다. 이 제어 코드는 UDF 파일 시스템에 사용됩니다. 이 구조체는 FSCTL_MAKE_MEDIA_COMPATIBLE을 호출할 때 입력에 사용됩니다.
FILE_NAME_INFO 파일 이름을 수신합니다.
FILE_NOTIFY_EXTENDED_INFORMATION ReadDirectoryChangesExW 함수를 통해 찾은 변경 내용에 대해 설명합니다.
FILE_NOTIFY_INFORMATION ReadDirectoryChangesW 함수를 통해 찾은 변경 내용에 대해 설명합니다.
FILE_OBJECTID_BUFFER 개체 식별자 및 개체 식별자와 연결된 사용자 정의 메타데이터를 포함합니다.
FILE_QUERY_ON_DISK_VOL_INFO_BUFFER FSCTL_QUERY_ON_DISK_VOLUME_INFO 호출에서 볼륨 정보를 수신합니다.
FILE_QUERY_SPARING_BUFFER 결함 관리 속성을 포함합니다.

FILE_REMOTE_PROTOCOL_INFO 파일 원격 프로토콜 정보를 포함합니다.
FILE_RENAME_INFO 파일의 이름을 바꿀 대상 이름을 포함합니다.
FILE_SET_DEFECT_MGMT_BUFFER 설정할 결함 관리 상태를 지정합니다.
FILE_SET_SPARSE_BUFFER 설정할 스파스 상태를 지정합니다.
FILE_STANDARD_INFO 파일에 대한 확장 정보를 수신합니다.
FILE_STORAGE_INFO 파일에 대한 디렉터리 정보를 포함합니다. (FILE_STORAGE_INFO)
FILE_STORAGE_TIER 볼륨을 기준으로 스토리지 계층의 식별자를 나타냅니다.
FILE_STORAGE_TIER_REGION 단일 스토리지 계층 영역을 설명합니다.
FILE_STREAM_INFO 지정된 파일에 대한 파일 스트림 정보를 수신합니다.
FILE_SYSTEM_RECOGNITION_INFORMATION FSCTL_QUERY_FILE_SYSTEM_RECOGNITION 제어 코드를 통해 가져온 파일 시스템 인식 정보를 포함합니다.
FILE_ZERO_DATA_INFORMATION 0으로 설정할 파일의 범위를 포함합니다.
FILESYSTEM_STATISTICS 파일 시스템의 통계 정보를 포함합니다.

FILESYSTEM_STATISTICS_EX 파일 시스템의 통계 정보를 포함합니다. 이 구조체에 대한 지원은 Windows 10 시작했습니다.
FIND_BY_SID_DATA FSCTL_FIND_FILES_BY_SID 제어 코드에 대한 데이터를 포함합니다.
FIND_BY_SID_OUTPUT 파일 이름을 나타냅니다.
FORMAT_EX_PARAMETERS 인접한 디스크 트랙 집합의 서식을 지정하는 데 사용되는 정보를 포함합니다. IOCTL_DISK_FORMAT_TRACKS_EX 제어 코드에서 사용됩니다.
FORMAT_PARAMETERS 인접한 디스크 트랙 집합의 서식을 지정하는 데 사용되는 정보를 포함합니다.
FSCTL_GET_INTEGRITY_INFORMATION_BUFFER 파일 또는 디렉터리에 대한 무결성 정보를 포함합니다.
FSCTL_QUERY_REGION_INFO_INPUT 특정 볼륨에 대한 스토리지 스택의 스토리지 계층 영역을 포함합니다.
FSCTL_QUERY_REGION_INFO_OUTPUT 하나 이상의 영역에 대한 정보를 포함합니다.
FSCTL_QUERY_STORAGE_CLASSES_OUTPUT 특정 볼륨의 모든 계층에 대한 정보를 포함합니다.
FSCTL_SET_INTEGRITY_INFORMATION_BUFFER FSCTL_SET_INTEGRITY_INFORMATION 제어 코드를 사용하여 전달된 입력 버퍼입니다.
GET_DISK_ATTRIBUTES 디스크 디바이스의 특성을 포함합니다.
GET_LENGTH_INFORMATION IOCTL_DISK_GET_LENGTH_INFO 제어 코드에서 사용하는 디스크, 볼륨 또는 파티션 길이 정보를 포함합니다.

IORING_BUFFER_INFO I/O 링에 등록할 수 있는 데이터 버퍼를 나타냅니다.
IORING_BUFFER_REF IORING_BUFFER_REF I/O 링 작업에 사용되는 버퍼에 대한 참조를 나타냅니다.
IORING_CAPABILITIES IORING API 기능을 나타냅니다.
IORING_CQE 완료된 I/O 링 큐 항목을 나타냅니다.
IORING_CREATE_FLAGS CreateIoRing을 호출하여 I/O 링을 만들기 위한 플래그를 지정합니다.
IORING_HANDLE_REF I/O 링 작업에 사용되는 파일 핸들에 대한 참조를 나타냅니다.
IORING_INFO 지정된 I/O 링의 세이프 및 버전 정보를 나타냅니다.
IORING_REGISTERED_BUFFER BuildIoRingRegisterBuffers를 호출하여 I/O 링에 등록된 버퍼를 나타냅니다.
LOG_MANAGEMENT_CALLBACKS LOG_MANAGEMENT_CALLBACKS 구조는 클라이언트 프로그램에 정보가 필요한 콜백에 대해 CLFS(Common Log File System)에 등록하는 데 사용됩니다.
LOOKUP_STREAM_FROM_CLUSTER_ENTRY FSCTL_LOOKUP_STREAM_FROM_CLUSTER 제어 코드에서 반환됩니다.
LOOKUP_STREAM_FROM_CLUSTER_INPUT FSCTL_LOOKUP_STREAM_FROM_CLUSTER 제어 코드에 입력으로 전달됩니다.
LOOKUP_STREAM_FROM_CLUSTER_OUTPUT FSCTL_LOOKUP_STREAM_FROM_CLUSTER 제어 코드에서 출력으로 수신됩니다.

MARK_HANDLE_INFO 지정된 파일 또는 디렉터리를 표시하는 데 사용되는 정보 및 변경 내용에 대한 데이터가 있는 USN(업데이트 시퀀스 번호) 변경 저널 레코드를 포함합니다.
MARK_HANDLE_INFO32 지정된 파일 또는 디렉터리를 표시하는 데 사용되는 정보 및 변경 내용에 대한 데이터가 있는 USN(업데이트 시퀀스 번호) 변경 저널 레코드를 포함합니다.
MediaLabelInfo MediaLabelInfo 구조체는 테이프 OMID에 대한 정보를 RSM 데이터베이스에 전달합니다. 미디어 레이블 라이브러리는 라이브러리가 인식하는 모든 미디어 레이블에 대해 이 구조를 채웁니다.
MFT_ENUM_DATA_V0 USN(업데이트 시퀀스 번호) 변경 저널 레코드 열거의 경계 및 시작 위치를 정의하는 정보를 포함합니다.
MFT_ENUM_DATA_V1 ReFS 볼륨에 대한 USN(업데이트 시퀀스 번호) 변경 저널 레코드 열거의 경계 및 시작 위치를 정의하는 정보를 포함합니다.
MOVE_FILE_DATA FSCTL_MOVE_FILE 제어 코드의 입력 데이터를 포함합니다.
NTFS_EXTENDED_VOLUME_DATA 볼륨 데이터를 나타냅니다. N
NTFS_FILE_RECORD_INPUT_BUFFER FSCTL_GET_NTFS_FILE_RECORD 제어 코드에 대한 데이터를 포함합니다.
NTFS_FILE_RECORD_OUTPUT_BUFFER FSCTL_GET_NTFS_FILE_RECORD 제어 코드에서 출력 데이터를 수신합니다.
NTFS_STATISTICS NTFS 파일 시스템의 통계 정보를 포함합니다.
NTFS_STATISTICS_EX NTFS 파일 시스템의 통계 정보를 포함합니다. 이 구조체에 대한 지원은 Windows 10 시작했습니다.

NTFS_VOLUME_DATA_BUFFER 볼륨 데이터를 나타냅니다. (NTFS_VOLUME_DATA_BUFFER)
NTMS_ALLOCATION_INFORMATION NTMS_ALLOCATION_INFORMATION 구조에는 미디어를 가져온 원본 미디어 풀에 대한 정보가 포함됩니다.
NTMS_CHANGERINFORMATIONA NTMS_CHANGERINFORMATION 구조체는 로봇 체인저 개체와 관련된 속성을 정의합니다. (ANSI)
NTMS_CHANGERINFORMATIONW NTMS_CHANGERINFORMATION 구조체는 로봇 체인저 개체와 관련된 속성을 정의합니다. (유니코드)
NTMS_CHANGERTYPEINFORMATIONA NTMS_CHANGERTYPEINFORMATION 구조는 RSM에서 지원하는 로봇 체인저 유형과 관련된 속성을 정의합니다. (ANSI)
NTMS_CHANGERTYPEINFORMATIONW NTMS_CHANGERTYPEINFORMATION 구조는 RSM에서 지원하는 로봇 체인저 유형과 관련된 속성을 정의합니다. (유니코드)
NTMS_COMPUTERINFORMATION NTMS_COMPUTERINFORMATION 구조체는 RSM 서버와 관련된 속성을 정의합니다.
NTMS_DRIVEINFORMATIONA NTMS_DRIVEINFORMATION 구조체는 드라이브 개체와 관련된 속성을 정의합니다. (ANSI)
NTMS_DRIVEINFORMATIONW NTMS_DRIVEINFORMATION 구조체는 드라이브 개체와 관련된 속성을 정의합니다. (유니코드)
NTMS_DRIVETYPEINFORMATIONA NTMS_DRIVETYPEINFORMATION 구조체는 RSM에서 지원하는 드라이브 유형과 관련된 속성을 정의합니다. (ANSI)
NTMS_DRIVETYPEINFORMATIONW NTMS_DRIVETYPEINFORMATION 구조체는 RSM에서 지원하는 드라이브 유형과 관련된 속성을 정의합니다. (유니코드)

NTMS_IEDOORINFORMATION NTMS_IEDOORINFORMATION 구조체는 insert/eject door 개체와 관련된 속성을 정의합니다.
NTMS_IEPORTINFORMATION NTMS_IEPORTINFORMATION 구조체는 insert/eject 포트 개체와 관련된 속성을 정의합니다.
NTMS_LIBRARYINFORMATION NTMS_LIBRARYINFORMATION 구조체는 라이브러리 개체와 관련된 속성을 정의합니다.
NTMS_LIBREQUESTINFORMATIONA NTMS_LIBREQUESTINFORMATION 구조는 RSM에 큐에 대기 중인 작업 요청과 관련된 속성을 정의합니다. (ANSI)
NTMS_LIBREQUESTINFORMATIONW NTMS_LIBREQUESTINFORMATION 구조는 RSM에 큐에 대기 중인 작업 요청과 관련된 속성을 정의합니다. (유니코드)
NTMS_LMIDINFORMATION NTMS_LMIDINFORMATION 구조체는 논리 미디어 개체와 관련된 속성을 정의합니다.
NTMS_MEDIAPOOLINFORMATION NTMS_MEDIAPOOLINFORMATION 구조체는 미디어 풀 개체와 관련된 속성을 정의합니다.
NTMS_MEDIATYPEINFORMATION NTMS_MEDIATYPEINFORMATION 구조체는 RSM에서 지원하는 미디어 유형과 관련된 속성을 정의합니다.
NTMS_NOTIFICATIONINFORMATION NTMS_NOTIFICATIONINFORMATION 구조체는 RSM 데이터베이스에서 발생한 개체 및 작업을 정의합니다.
NTMS_OBJECTINFORMATIONA NTMS_OBJECTINFORMATION 구조는 애플리케이션이 RSM 디바이스, 미디어 및 시스템 컨트롤(예: 라이브러리, 드라이브, 미디어, 운영자 요청)에 대해 가져와서 설정할 수 있는 속성을 정의합니다. RSM 데이터베이스에 있는 개체의 일반적인 구조입니다. (ANSI)

NTMS_OBJECTINFORMATIONW NTMS_OBJECTINFORMATION 구조는 애플리케이션이 RSM 디바이스, 미디어 및 시스템 컨트롤(예: 라이브러리, 드라이브, 미디어, 운영자 요청)에 대해 가져와서 설정할 수 있는 속성을 정의합니다. RSM 데이터베이스에 있는 개체의 일반적인 구조입니다. (유니코드)
NTMS_OPREQUESTINFORMATIONA NTMS_OPREQUESTINFORMATION 구조체는 RSM에 대한 연산자 요청 시스템 제어와 관련된 속성을 정의합니다. (ANSI)
NTMS_OPREQUESTINFORMATIONW NTMS_OPREQUESTINFORMATION 구조체는 RSM에 대한 연산자 요청 시스템 제어와 관련된 속성을 정의합니다. (유니코드)
NTMS_PARTITIONINFORMATIONA NTMS_PARTITIONINFORMATION 구조체는 사이드 개체와 관련된 속성을 정의합니다. (ANSI)
NTMS_PARTITIONINFORMATIONW NTMS_PARTITIONINFORMATION 구조체는 사이드 개체와 관련된 속성을 정의합니다. (유니코드)
NTMS_PMIDINFORMATIONA NTMS_PMIDINFORMATION 구조체는 물리적 미디어 개체와 관련된 속성을 정의합니다. (ANSI)
NTMS_PMIDINFORMATIONW NTMS_PMIDINFORMATION 구조체는 물리적 미디어 개체와 관련된 속성을 정의합니다. (유니코드)
NTMS_STORAGESLOTINFORMATION NTMS_STORAGESLOTINFORMATION 구조체는 스토리지 슬롯 개체와 관련된 속성을 정의합니다.
NVM_SET_LIST NVME 특성 설정 명령에 대한 항목 배열을 포함합니다.
NVME_ADMIN_COMPLETION_QUEUE_BASE_ADDRESS 관리 완료 큐의 기본 메모리 주소를 포함합니다.
NVME_ADMIN_QUEUE_ATTRIBUTES 관리 제출 큐 및 관리 완료 큐에 대한 AQA(관리 큐 특성)를 포함합니다.

NVME_ADMIN_SUBMISSION_QUEUE_BASE_ADDRESS 관리 제출 큐의 기본 메모리 주소를 포함합니다.
NVME_AUTO_POWER_STATE_TRANSITION_ENTRY 허용되는 32개 전원 상태 각각에 대한 유휴 시간 및 전원 상태 전환에 대한 정보를 지정하는 64비트 항목을 포함합니다.
NVME_CDW10_ABORT 이전에 관리 제출 큐 또는 I/O 제출 큐에 제출된 특정 명령을 중단하는 데 사용되는 Abort 명령에 대한 매개 변수를 포함합니다.
NVME_CDW10_CREATE_IO_QUEUE I/O 완료 큐 만들기 및 IO 제출 큐 만들기 명령에 사용되는 매개 변수를 포함합니다.
NVME_CDW10_DATASET_MANAGEMENT NVME_CDW10_DATASET_MANAGEMENT 구조에는 논리 블록 범위에 대한 특성을 나타내는 데이터 세트 관리 명령에 대한 매개 변수가 포함되어 있습니다.
NVME_CDW10_DIRECTIVE_RECEIVE 지시문 수신 명령에 대한 매개 변수를 포함합니다.
NVME_CDW10_DIRECTIVE_SEND 지시문 보내기 명령에 대한 매개 변수를 포함합니다.
NVME_CDW10_FIRMWARE_ACTIVATE 펌웨어 커밋 명령에 사용되는 매개 변수를 포함합니다.
NVME_CDW10_FIRMWARE_DOWNLOAD NVME_CDW10_FIRMWARE_DOWNLOAD 구조에는 컨트롤러에 새 펌웨어 이미지를 복사하는 펌웨어 이미지 다운로드 명령에 대한 매개 변수가 포함되어 있습니다.
NVME_CDW10_FORMAT_NVM NVM 미디어의 형식을 낮게 지정하는 데 사용되는 NVM 형식 명령에 대한 매개 변수를 포함합니다.
NVME_CDW10_GET_FEATURES 지정된 기능의 특성을 검색하는 기능 가져오기 명령에 대한 매개 변수를 포함합니다.

NVME_CDW10_GET_LOG_PAGE NVME_CDW10_GET_LOG_PAGE 구조에는 요청된 로그 페이지가 포함된 데이터 버퍼를 반환하는 로그 페이지 가져오기 명령에 대한 매개 변수가 포함되어 있습니다.
NVME_CDW10_GET_LOG_PAGE_V13 NVME_CDW10_GET_LOG_PAGE_V13 구조에는 요청된 로그 페이지가 포함된 데이터 버퍼를 반환하는 로그 페이지 가져오기 명령에 대한 매개 변수가 포함되어 있습니다.
NVME_CDW10_IDENTIFY NVM 하위 시스템, 컨트롤러 또는 네임스페이스에 대한 정보를 설명하는 데이터 버퍼를 반환하는 Identify 명령에 대한 매개 변수를 포함합니다.
NVME_CDW10_SECURITY_SEND_RECEIVE 관리 명령 보안 보내기 및 보안 수신에 사용되는 매개 변수를 포함합니다.
NVME_CDW10_SET_FEATURES NVME_CDW10_SET_FEATURES 구조체에는 지정된 기능의 특성을 설정하는 기능 설정 명령에 대한 매개 변수가 포함되어 있습니다.
NVME_CDW11_CREATE_IO_CQ 관리 완료 큐를 제외한 모든 I/O 완료 큐를 만드는 데 사용되는 I/O 완료 큐 만들기 명령에 대한 매개 변수를 포함합니다.
NVME_CDW11_CREATE_IO_SQ IO 제출 큐를 만드는 데 사용되는 IO 제출 큐 만들기 명령에 대한 매개 변수를 포함합니다.
NVME_CDW11_DATASET_MANAGEMENT NVME_CDW11_DATASET_MANAGEMENT 구조에는 논리 블록 범위에 대한 특성을 나타내는 데이터 세트 관리 명령에 대한 매개 변수가 포함되어 있습니다.
NVME_CDW11_DIRECTIVE_RECEIVE 지시문 수신 명령에 대한 매개 변수를 포함합니다.
NVME_CDW11_DIRECTIVE_SEND 지시문 보내기 명령에 대한 매개 변수를 포함합니다.
NVME_CDW11_FEATURE_ARBITRATION 명령 중재를 제어하는 중재 기능에 대한 값을 포함합니다.

NVME_CDW11_FEATURE_ASYNC_EVENT_CONFIG 호스트에 대한 비동기 이벤트 알림을 트리거하는 이벤트를 제어하는 비동기 이벤트 구성 기능에 대한 매개 변수를 포함합니다.
NVME_CDW11_FEATURE_AUTO_POWER_STATE_TRANSITION 자율 전원 상태 전환에 대한 설정을 구성하는 자율 전원 상태 전환 기능에 대한 매개 변수를 포함합니다.
NVME_CDW11_FEATURE_HOST_MEMORY_BUFFER 호스트가 호스트 메모리의 일부를 할당하여 컨트롤러에서 단독으로 사용할 수 있도록 하는 메커니즘을 제공하는 호스트 메모리 버퍼 기능에 대한 매개 변수를 포함합니다.
NVME_CDW11_FEATURE_INTERRUPT_COALESCING 인터럽트 병합 설정을 구성하는 인터럽트 병합 기능에 대한 매개 변수를 포함합니다.
NVME_CDW11_FEATURE_INTERRUPT_VECTOR_CONFIG 특정 인터럽트 벡터와 관련된 설정을 구성하는 인터럽트 벡터 구성 기능에 대한 매개 변수를 포함합니다.
NVME_CDW11_FEATURE_LBA_RANGE_TYPE 기능 설정 명령에서 LBA 범위 유형 기능의 LBA 범위 수를 지정하는 매개 변수를 포함합니다.
NVME_CDW11_FEATURE_NON_OPERATIONAL_POWER_STATE 비작동 전원 상태에 대해 허용 모드를 사용할 수 있는지 여부를 나타내는 비운전 전원 상태 기능에 대한 매개 변수를 포함합니다.
NVME_CDW11_FEATURE_NUMBER_OF_QUEUES 호스트가 이 컨트롤러에 대해 요청하는 I/O 완료 큐 및 I/O 제출 큐 수를 나타내는 큐 수 기능에 대한 매개 변수를 포함합니다.
NVME_CDW11_FEATURE_POWER_MANAGEMENT 호스트가 전원 상태를 구성할 수 있도록 하는 전원 관리 기능에 대한 값을 포함합니다.
NVME_CDW11_FEATURE_SUPPORTED_CAPABILITY 지정된 기능의 지원되는 기능을 설명하는 기능 가져오기 명령에서 반환되는 값을 포함합니다.

NVME_CDW11_FEATURE_TEMPERATURE_THRESHOLD 온도 임계값 초과 및 최대 9개의 온도 값에 대한 온도 임계값 미만을 설정하는 데 사용되는 온도 임계값 기능에 대한 매개 변수를 포함합니다.
NVME_CDW11_FEATURE_VOLATILE_WRITE_CACHE 컨트롤러에서 지원되고 존재하는 경우 휘발성 쓰기 캐시를 제어하는 휘발성 쓰기 캐시 기능에 대한 매개 변수를 포함합니다.
NVME_CDW11_FEATURE_WRITE_ATOMICITY_NORMAL 원자성 작업에 대한 컨트롤러의 지원을 정의하는 **AWUN**(Atomic Write Unit Normal) 및 네임스페이스 **NAWUN**(Atomic Write Unit Normal) 매개 변수의 작업을 제어하는 쓰기 원자성 일반 기능에 대한 매개 변수를 포함합니다.
NVME_CDW11_FEATURES 지정된 기능의 특성을 검색하거나 설정하는 기능 가져오기 및 기능 설정 명령에 대한 매개 변수를 포함합니다.
NVME_CDW11_FIRMWARE_DOWNLOAD 새 펌웨어 이미지(전체 또는 부분적으로)를 컨트롤러에 복사하는 데 사용되는 펌웨어 이미지 다운로드 명령에 대한 매개 변수를 포함합니다.
NVME_CDW11_GET_LOG_PAGE NVME_CDW11_GET_LOG_PAGE 구조에는 요청된 로그 페이지가 포함된 데이터 버퍼를 반환하는 로그 페이지 가져오기 명령에 대한 매개 변수가 포함되어 있습니다.
NVME_CDW11_IDENTIFY Identify 명령에 대한 매개 변수를 포함합니다.
NVME_CDW11_SECURITY_RECEIVE 보안 수신 명령에 사용되는 매개 변수를 포함합니다.
NVME_CDW11_SECURITY_SEND 보안 보내기 명령에 사용되는 매개 변수를 포함합니다.
NVME_CDW12_DIRECTIVE_RECEIVE 지시문 수신 명령에 대한 스트림 리소스를 할당하기 위한 매개 변수를 포함합니다.

NVME_CDW12_DIRECTIVE_RECEIVE_STREAMS_ALLOCATE_RESOURCES 지시문 수신 명령에서 스트림 리소스를 할당하는 데 사용되는 네임스페이스 스트림을 요청하기 위한 매개 변수를 포함합니다.
NVME_CDW12_DIRECTIVE_SEND 지시문 보내기 명령에 지시문을 사용하도록 설정하기 위한 매개 변수를 포함합니다.
NVME_CDW12_DIRECTIVE_SEND_IDENTIFY_ENABLE_DIRECTIVE 지시문 보내기 명령에서 지시문을 지정하고 사용하도록 설정하기 위한 매개 변수를 포함합니다.
NVME_CDW12_FEATURE_HOST_MEMORY_BUFFER 호스트 메모리 버퍼의 크기를 지정하는 호스트 메모리 버퍼 기능에 대한 매개 변수를 포함합니다.
NVME_CDW12_FEATURES 지정된 기능의 특성을 설정하는 기능 설정 명령에 대한 매개 변수를 포함합니다.
NVME_CDW12_GET_LOG_PAGE
NVME_CDW12_READ_WRITE 지정된 LBA(논리 블록 주소)에 대해 NVM 컨트롤러에서 데이터 및 메타데이터(해당하는 경우)를 읽거나 쓰는 NVME 읽기 및 NVME 쓰기 명령에 대한 매개 변수를 포함합니다.
NVME_CDW13_FEATURE_HOST_MEMORY_BUFFER 호스트 메모리 설명자 목록의 실제 위치의 하위 32비트를 지정하는 호스트 메모리 버퍼 기능에 대한 매개 변수를 포함합니다.
NVME_CDW13_FEATURES 지정된 기능의 특성을 설정하는 기능 설정 명령에 대한 매개 변수를 포함합니다.
NVME_CDW13_GET_LOG_PAGE
NVME_CDW13_READ_WRITE 지정된 LBA(논리 블록 주소)에 대해 NVM 컨트롤러에서 데이터 및 메타데이터(해당하는 경우)를 읽거나 쓰는 NVME 읽기 및 NVME 쓰기 명령에 대한 매개 변수를 포함합니다.

NVME_CDW14_FEATURE_HOST_MEMORY_BUFFER 호스트 메모리 설명자 목록의 실제 위치의 상위 32비트를 지정하는 호스트 메모리 버퍼 기능에 대한 매개 변수를 포함합니다.
NVME_CDW14_FEATURES NVME_CDW14_FEATURES 구조체에는 지정된 기능의 특성을 설정하는 기능 설정 명령에 대한 매개 변수가 포함되어 있습니다.
NVME_CDW15_FEATURE_HOST_MEMORY_BUFFER 호스트 메모리 설명자 목록의 항목 수를 지정하는 호스트 메모리 버퍼 기능에 대한 매개 변수를 포함합니다.
NVME_CDW15_FEATURES NVME_CDW15_FEATURES 구조체에는 지정된 기능의 특성을 설정하는 기능 설정 명령에 대한 매개 변수가 포함되어 있습니다.
NVME_CDW15_READ_WRITE 지정된 LBA(논리 블록 주소)에 대해 NVM 컨트롤러에서 데이터 및 메타데이터(해당하는 경우)를 읽거나 쓰는 NVME 읽기 및 NVME 쓰기 명령에 대한 매개 변수를 포함합니다.
NVME_CHANGED_NAMESPACE_LIST_LOG 로그 페이지를 마지막으로 읽은 이후 네임스페이스 식별 정보를 변경한 컨트롤러의 네임스페이스를 설명하는 변경된 네임스페이스 목록 로그 페이지에 대한 데이터를 포함합니다.
NVME_COMMAND 관리 명령 및 NVM 명령 집합의 모든 명령에 대한 매개 변수를 포함합니다.
NVME_COMMAND_DWORD0 모든 관리 명령 및 NVM 명령에 공통적인 매개 변수를 포함합니다.
NVME_COMMAND_EFFECTS_DATA 명령의 선택적 기능을 포함하여 관리 또는 I/O 명령의 전반적인 가능한 효과를 설명하는 정보를 포함합니다.
NVME_COMMAND_EFFECTS_LOG 컨트롤러가 지원하는 명령과 이러한 명령이 NVM 하위 시스템의 상태에 미치는 영향을 설명하는 정보를 포함합니다.

NVME_COMMAND_STATUS 명령의 상태에 대한 정보를 포함합니다.
NVME_COMPLETION_DW0_ASYNC_EVENT_REQUEST 완료 큐 항목의 DWord 0에서 관리 완료 큐에 게시되는 비동기 이벤트에 대한 정보를 포함합니다. 비동기 이벤트는 호스트 소프트웨어에 상태, 오류 및 상태 정보를 알리는 데 사용됩니다.
NVME_COMPLETION_DW0_DIRECTIVE_RECEIVE_STREAMS_ALLOCATE_RESOURCES 지시문 수신 명령의 할당된 스트림 리소스 수에 대한 정보를 포함합니다.
NVME_COMPLETION_ENTRY 완료 큐의 크기가 16바이트인 항목을 지정합니다.
NVME_COMPLETION_QUEUE_HEAD_DOORBELL 완료 큐 y에 대한 헤드 항목 포인터를 업데이트하는 초인종 레지스터를 정의 *합니다*.
NVME_CONTEXT_ATTRIBUTES LBA 범위(논리 블록 범위)에 대한 선택적 컨텍스트 특성을 지정합니다.
NVME_CONTROLLER_CAPABILITIES 소프트웨어를 호스트하는 컨트롤러의 기본 기능을 지정하는 읽기 전용 값을 포함합니다.
NVME_CONTROLLER_CONFIGURATION 컨트롤러에 대한 읽기/쓰기 구성 설정을 포함합니다.
NVME_CONTROLLER_LIST 컨트롤러 식별자의 순서가 지정된 목록을 포함합니다.
NVME_CONTROLLER_MEMORY_BUFFER_LOCATION NVME_CONTROLLER_REGISTERS 구조체의 **CMBLOC** 필드에 있는 선택적 컨트롤러 메모리 버퍼 위치 레지스터의 위치를 정의합니다.
NVME_CONTROLLER_MEMORY_BUFFER_SIZE 선택적 컨트롤러 메모리 버퍼 레지스터의 크기를 정의하고 NVME_CONTROLLER_REGISTERS 구조체의 **CMBSZ** 필드에 사용됩니다.

NVME_CONTROLLER_REGISTERS 컨트롤러에 대한 레지스터 맵을 지정합니다.
NVME_CONTROLLER_STATUS 컨트롤러 상태를 나타내는 값을 포함합니다.
NVME_DEVICE_SELF_TEST_LOG 디바이스 자체 테스트의 상태, 완료율 및 결과를 설명하는 디바이스 자체 테스트 로그 페이지의 정보를 지정하는 필드가 포함되어 있습니다.
NVME_DEVICE_SELF_TEST_RESULT_DATA 디바이스 Self-Test 작업의 결과에 대한 데이터를 포함합니다.
NVME_DIRECTIVE_IDENTIFY_RETURN_PARAMETERS Identify 지시문에 대한 반환 매개 변수를 설명하는 필드를 포함합니다.
NVME_DIRECTIVE_IDENTIFY_RETURN_PARAMETERS_DESCRIPTOR Identify 지시문에 대한 반환 매개 변수를 설명하는 값을 포함합니다.
NVME_DIRECTIVE_STREAMS_GET_STATUS_DATA 현재 열려 있는 스트림의 식별자를 포함합니다.
NVME_DIRECTIVE_STREAMS_RETURN_PARAMETERS Streams 지시문에 대한 반환 매개 변수를 포함합니다.
NVME_ENDURANCE_GROUP_LOG 지구력 그룹 정보 로그 페이지에서 지구력 그룹에서 읽고 쓰는 데이터의 양을 나타내는 정보를 지정하는 필드를 포함합니다.
NVME_ERROR_INFO_LOG 오류 정보 로그 페이지의 정보를 지정하는 필드가 포함되어 있습니다.
NVME_FIRMWARE_SLOT_INFO_LOG 펌웨어 슬롯 정보 로그 페이지의 정보를 지정하는 필드가 포함되어 있습니다.
NVME_HEALTH_INFO_LOG SMART/Health Information Log 페이지에 포함된 정보를 지정하는 필드가 포함되어 있습니다.

NVME_HOST_MEMORY_BUFFER_DESCRIPTOR_ENTRY 호스트 메모리 설명자 목록에서 단일 항목의 매개 변수를 정의합니다.
NVME_IDENTIFY_CONTROLLER_DATA 컨트롤러 기능, 기능, 명령 집합 특성 및 전원 상태 설명자를 나타내는 값을 포함합니다.
NVME_IDENTIFY_NAMESPACE_DATA 특정 네임스페이스와 관련된 기능 및 설정을 나타내는 값을 포함합니다.
NVME_LBA_FORMAT NVM 형식 명령의 일부로 NVM 미디어에 적용할 LBA 형식을 지정하는 매개 변수를 포함합니다.
NVME_LBA_RANGE 시작 LBA에서 지정한 연속 논리 블록의 컬렉션과 논리 블록 수를 정의하는 매개 변수를 포함합니다.
NVME_LBA_RANGET_TYPE_ENTRY 기능 설정 명령의 LBA 범위 유형 기능에 대한 LBA(논리 블록 주소) 범위 목록에서 단일 항목을 지정하는 매개 변수를 포함합니다.
NVME_NVM_SUBSYSTEM_RESET NVM 하위 시스템 재설정을 시작하는 기능을 호스트 소프트웨어에 제공하는 매개 변수를 지정합니다.
NVME_POWER_STATE_DESC 각 전원 상태의 특성을 설명하는 필드를 포함합니다.
NVME_PRP_ENTRY 실제 메모리 페이지에 대한 포인터의 PBAO(페이지 기준 주소 및 오프셋)를 지정하는 필드가 포함되어 있습니다.
NVME_RESERVATION_NOTIFICATION_LOG 예약 알림 로그 페이지의 정보를 지정하는 필드가 포함되어 있습니다.
NVME_SCSI_NAME_STRING SCSI 이름 문자열 식별자를 생성하는 데 사용되는 정보를 포함합니다.

NVME_SET_ATTRIBUTES_ENTRY 특성을 설정하기 위한 정보를 지정하는 필드가 포함되어 있습니다.
NVME_SUBMISSION_QUEUE_TAIL_DOORBELL 제출 큐 y에 대한 Tail 항목 포인터를 업데이트하는 초인종 레지스터를 정의 *합니다*.
NVME_TELEMETRY_HOST_INITIATED_LOG 원격 분석 Host-Initiated 로그 페이지에서 정보를 지정하는 필드를 포함합니다.
NVME_VERSION 컨트롤러 구현에서 지원하는 NVM Express 사양의 버전 번호를 지정하는 필드가 포함되어 있습니다.
OFSTRUCT OpenFile 함수가 열었거나 열려고 시도한 파일에 대한 정보를 포함합니다.
OVERLAPPED_ENTRY GetQueuedCompletionStatusEx 함수 호출로 반환된 정보를 포함합니다.
PARTITION_INFORMATION 디스크 파티션에 대한 정보를 포함합니다.
PARTITION_INFORMATION_EX 표준 AT 스타일 MBR(마스터 부트 레코드) 및 EFI(Extensible Firmware Interface) 디스크에 대한 파티션 정보를 포함합니다.
PARTITION_INFORMATION_GPT GPT(GUID 파티션 테이블) 파티션 정보를 포함합니다.
PARTITION_INFORMATION_MBR MBR(마스터 부트 레코드) 디스크와 관련된 파티션 정보를 포함합니다.
PLEX_READ_DATA_REQUEST 수행할 읽기 작업의 범위와 읽을 플렉스를 나타냅니다.
READ_FILE_USN_DATA 애플리케이션에서 지원하는 USN(업데이트 시퀀스 번호) 변경 저널의 버전을 지정합니다.

READ_USN_JOURNAL_DATA_V0 호출 프로세스에 반환할 USN(업데이트 시퀀스 번호) 변경 저널 레코드 세트를 정의하는 정보를 포함합니다. (READ_USN_JOURNAL_DATA_V0)
READ_USN_JOURNAL_DATA_V1 호출 프로세스로 돌아가기 위해 USN(업데이트 시퀀스 번호) 변경 저널 레코드 집합을 정의하는 정보를 포함합니다. R
REASSIGN_BLOCKS 디스크 블록 재할당 데이터를 포함합니다. (REASSIGN_BLOCKS)
REASSIGN_BLOCKS_EX 디스크 블록 재할당 데이터를 포함합니다. (REASSIGN_BLOCKS_EX)
REPAIR_COPIES_INPUT FSCTL_REPAIR_COPIES 제어 코드에 대한 입력 구조입니다.
REPAIR_COPIES_OUTPUT FSCTL_REPAIR_COPIES 제어 코드에서 반환된 복사본 복구 작업의 출력을 포함합니다.
REPARSE_GUID_DATA_BUFFER 재분석 지점에 대한 정보를 포함합니다.
REQUEST_OPLOCK_INPUT_BUFFER 편의적 잠금(oplock)을 요청하거나 FSCTL_REQUEST_OPLOCK 제어 코드로 oplock 중단을 확인하는 정보를 포함합니다.
REQUEST_OPLOCK_OUTPUT_BUFFER FSCTL_REQUEST_OPLOCK 제어 코드에서 반환된 편의적 잠금(oplock) 정보를 포함합니다.
RETRIEVAL_POINTER_BASE FSCTL_GET_RETRIEVAL_POINTER_BASE 제어 코드에 대한 출력을 포함합니다.
RETRIEVAL_POINTERS_BUFFER FSCTL_GET_RETRIEVAL_POINTERS 제어 코드에 대한 출력을 포함합니다.

SESSION_INFO_0 세션을 설정한 컴퓨터의 이름을 포함합니다.
SESSION_INFO_1 컴퓨터의 이름을 포함하여 세션에 대한 정보를 포함합니다. 사용자의 이름입니다. 컴퓨터에서 파일, 파이프 및 디바이스를 엽니다.
SESSION_INFO_10 컴퓨터의 이름을 포함하여 세션에 대한 정보를 포함합니다. 사용자의 이름입니다. 세션에 대한 활성 및 유휴 시간입니다.
SESSION_INFO_2 컴퓨터의 이름을 포함하여 세션에 대한 정보를 포함합니다. 사용자의 이름입니다. 컴퓨터에서 파일, 파이프 및 디바이스 열기 및 세션을 설정한 클라이언트의 유형입니다.
SESSION_INFO_502 컴퓨터의 이름을 포함하여 세션에 대한 정보를 포함합니다. 사용자의 이름입니다. 컴퓨터에서 파일, 파이프 및 디바이스 열기 및 클라이언트가 사용하는 전송의 이름입니다.
SET_DISK_ATTRIBUTES 디스크 디바이스에서 설정할 특성을 지정합니다.
SET_PARTITION_INFORMATION 디스크 파티션의 형식을 설정하는 데 사용되는 정보를 포함합니다.
SHARE_INFO_0 공유 리소스의 이름을 포함합니다.
SHARE_INFO_1 리소스의 이름 및 유형 및 리소스와 연결된 주석을 포함하여 공유 리소스에 대한 정보를 포함합니다.
SHARE_INFO_1004 공유 리소스와 연결된 주석을 포함합니다.
SHARE_INFO_1005 공유 리소스에 대한 정보를 포함합니다.

SHARE_INFO_1006 공유 리소스가 수용할 수 있는 최대 동시 연결 수를 지정합니다.
SHARE_INFO_1501 지정된 공유와 연결된 보안 설명자를 포함합니다. 자세한 내용은 보안 설명자를 참조하세요.
SHARE_INFO_2 리소스 이름, 유형 및 권한, 현재 연결 수를 포함하여 공유 리소스에 대한 정보를 포함합니다.
SHARE_INFO_501 리소스의 이름 및 유형 및 리소스와 연결된 주석을 포함하여 공유 리소스에 대한 정보를 포함합니다.
SHARE_INFO_502 리소스 이름, 유형 및 권한, 연결 수 및 기타 관련 정보를 포함하여 공유 리소스에 대한 정보를 포함합니다.
SHARE_INFO_503 공유 리소스에 대한 정보를 포함합니다. 서버 이름도 포함되어 있다는 점을 제외하고 SHARE_INFO_502 구조와 동일합니다.
SHRINK_VOLUME_INFORMATION 수행할 볼륨 축소 작업을 지정합니다.
STARTING_LCN_INPUT_BUFFER FSCTL_GET_VOLUME_BITMAP 제어 코드에 대한 시작 LCN을 포함합니다.
STARTING_VCN_INPUT_BUFFER FSCTL_GET_RETRIEVAL_POINTERS 제어 코드에 대한 시작 VCN을 포함합니다.
STAT_SERVER_0 서버에 대한 통계 정보를 포함합니다.
STORAGE_ACCESS_ALIGNMENT_DESCRIPTOR IOCTL_STORAGE_QUERY_PROPERTY 제어 코드와 함께 사용하여 디바이스에 대한 스토리지 액세스 맞춤 설명자 데이터를 검색합니다.

STORAGE_ADAPTER_DESCRIPTOR IOCTL_STORAGE_QUERY_PROPERTY 제어 코드와 함께 사용하여 디바이스에 대한 스토리지 어댑터 설명자 데이터를 검색합니다.
STORAGE_ADAPTER_SERIAL_NUMBER STORAGE_PROPERTY_ID 정의된 StorageAdapterSerialNumberProperty에 대한 어댑터 일련 번호의 NULL로 끝나는 유니코드 문자열입니다.
STORAGE_DESCRIPTOR_HEADER IOCTL_STORAGE_QUERY_PROPERTY 제어 코드와 함께 사용하여 스토리지 디바이스 또는 어댑터의 속성을 검색합니다.
STORAGE_DEVICE_ATTRIBUTES_DESCRIPTOR 다음에 사용하도록 예약됩니다. (STORAGE_DEVICE_ATTRIBUTES_DESCRIPTOR)
STORAGE_DEVICE_DESCRIPTOR IOCTL_STORAGE_QUERY_PROPERTY 제어 코드와 함께 사용하여 디바이스에 대한 스토리지 디바이스 설명자 데이터를 검색합니다.
STORAGE_DEVICE_ID_DESCRIPTOR IOCTL_STORAGE_QUERY_PROPERTY 제어 코드 요청과 함께 사용하여 디바이스에 대한 디바이스 ID 설명자 데이터를 검색합니다.
STORAGE_DEVICE_IO_CAPABILITY_DESCRIPTOR STORAGE_PROPERTY_ID 정의된 StorageDeviceIoCapabilityProperty에 대한 출력 버퍼입니다.
STORAGE_DEVICE_POWER_CAP 이 구조체는 IOCTL_STORAGE_DEVICE_POWER_CAP 대한 입력 및 출력 버퍼로 사용됩니다.
STORAGE_DEVICE_RESILIENCY_DESCRIPTOR 시스템에서 사용하도록 예약되었습니다. (STORAGE_DEVICE_RESILIENCY_DESCRIPTOR)
STORAGE_HW_FIRMWARE_ACTIVATE 이 구조체에는 활성화할 다운로드한 펌웨어에 대한 정보가 포함되어 있습니다.
STORAGE_HW_FIRMWARE_DOWNLOAD 이 구조에는 대상에 다운로드할 펌웨어 이미지 페이로드가 포함되어 있습니다.

STORAGE_MEDIUM_PRODUCT_TYPE_DESCRIPTOR IOCTL_STORAGE_QUERY_PROPERTY 요청과 함께 사용되어 스토리지 디바이스의 제품 유형을 설명합니다.
STORAGE_MINIPORT_DESCRIPTOR 시스템에서 사용하도록 예약되었습니다. (STORAGE_MINIPORT_DESCRIPTOR)
STORAGE_PHYSICAL_ADAPTER_DATA 실제 스토리지 어댑터를 설명합니다.
STORAGE_PHYSICAL_DEVICE_DATA 실제 스토리지 디바이스를 설명합니다.
STORAGE_PHYSICAL_NODE_DATA 스토리지 노드의 물리적 디바이스 데이터를 지정합니다.
STORAGE_PHYSICAL_TOPOLOGY_DESCRIPTOR STORAGE_PHYSICAL_TOPOLOGY_DESCRIPTOR 구조체는 IOCTL_STORAGE_QUERY_PROPERTY 요청에서 반환된 쿼리 결과 구조체 중 하나입니다.
STORAGE_PROPERTY_QUERY IOCTL_STORAGE_QUERY_PROPERTY 제어 코드에 전달된 입력 버퍼로 검색할 스토리지 디바이스 또는 어댑터의 속성을 나타냅니다.
STORAGE_PROTOCOL_COMMAND 이 구조체는 통과 메커니즘을 사용하여 IOCTL_STORAGE_PROTOCOL_COMMAND를 통해 공급업체별 명령을 스토리지 디바이스에 발급할 때 입력 버퍼로 사용됩니다.
STORAGE_PROTOCOL_DATA_DESCRIPTOR 이 구조체는 스토리지 디바이스 또는 어댑터에서 프로토콜별 데이터를 반환하기 위해 IOCTL_STORAGE_QUERY_PROPERTY와 함께 사용됩니다.
STORAGE_PROTOCOL_SPECIFIC_DATA IOCTL_STORAGE_QUERY_PROPERTY 요청의 입력 및 출력 버퍼에 제공되는 프로토콜별 디바이스 데이터를 설명합니다.
STORAGE_SPEC_VERSION 스토리지 사양 버전입니다.

STORAGE_TEMPERATURE_DATA_DESCRIPTOR 이 구조체는 스토리지 디바이스 또는 어댑터에서 온도 데이터를 반환하기 위해 IOCTL_STORAGE_QUERY_PROPERTY와 함께 사용됩니다.
STORAGE_TEMPERATURE_INFO 디바이스 온도 데이터를 설명합니다. IOCTL_STORAGE_QUERY_PROPERTY 요청으로 온도 데이터를 쿼리할 때 STORAGE_TEMPERATURE_DATA_DESCRIPTOR의 일부로 반환됩니다.
STORAGE_TEMPERATURE_THRESHOLD 이 구조체는 IOCTL_STORAGE_SET_TEMPERATURE_THRESHOLD를 통해 스토리지 디바이스의 온도 상한 또는 하한 임계값을 설정하는 데 사용됩니다.
STORAGE_WRITE_CACHE_PROPERTY IOCTL_STORAGE_QUERY_PROPERTY 제어 코드와 함께 사용하여 디바이스의 쓰기 캐시 속성에 대한 정보를 검색합니다.
TRANSACTION_NOTIFICATION 트랜잭션 알림과 연결된 데이터를 포함합니다.
TRANSACTION_NOTIFICATION_RECOVERY_ARGUMENT 복구할 트랜잭션을 나타냅니다. 이 구조는 복구 알림과 함께 전송됩니다.
TXF_ID 리소스 관리자의 컨텍스트 내의 고유 식별자를 나타냅니다.
TXF_LOG_RECORD_AFFECTED_FILE 트랜잭션의 영향을 받은 파일에 대한 정보를 포함합니다.
TXF_LOG_RECORD_BASE 기본 레코드 정보를 포함합니다.
TXF_LOG_RECORD_TRUNCATE 잘린 작업에 대한 레코드를 포함합니다.
TXF_LOG_RECORD_WRITE 쓰기 작업에 대한 레코드를 포함합니다.

TXFS_CREATE_MINIVERSION_INFO FSCTL_TXFS_CREATE_MINIVERSION에 의해 만들어진 미니버전에 대한 버전 정보를 포함합니다.
TXFS_GET_METADATA_INFO_OUT 생성된 미니버전에 대한 버전 정보를 포함합니다.
TXFS_GET_TRANSACTED_VERSION 지정된 파일의 기본 및 최신 버전에 대한 정보를 포함합니다.
TXFS_LIST_TRANSACTION_LOCKED_FILES 트랜잭션된 기록기에 의해 잠긴 파일의 목록을 포함합니다.
TXFS_LIST_TRANSACTION_LOCKED_FILES_ENTRY 잠긴 트랜잭션에 대한 정보를 포함합니다.
TXFS_LIST_TRANSACTIONS 트랜잭션 목록을 포함합니다.
TXFS_LIST_TRANSACTIONS_ENTRY 트랜잭션에 대한 정보를 포함합니다.
TXFS_MODIFY_RM 보조 리소스 관리자에 대한 로그 매개 변수 및 로깅 모드를 수정할 때 필요한 정보를 포함합니다.
TXFS_QUERY_RM_INFORMATION 리소스 관리자(RM)에 대한 정보를 포함합니다.
TXFS_READ_BACKUP_INFORMATION_OUT 트랜잭션 NTFS(TxF) 특정 구조체를 포함합니다. 이 정보는 TXFS_WRITE_BACKUP_INFORMATION을 호출할 때만 사용해야 합니다. (TXFS_READ_BACKUP_INFORMATION_OUT)
TXFS_SAVEPOINT_INFORMATION FSCTL_TXFS_SAVEPOINT_INFORMATION 구조체는 수행할 작업과 트랜잭션을 지정합니다.
TXFS_TRANSACTION_ACTIVE_INFO 스냅샷을 만들 때 트랜잭션이 활성 상태인지를 나타내는 플래그를 포함합니다.

TXFS_WRITE_BACKUP_INFORMATION 트랜잭션 NTFS(TxF) 특정 구조체를 포함합니다. 이 정보는 TXFS_WRITE_BACKUP_INFORMATION을 호출할 때만 사용해야 합니다. (TXFS_WRITE_BACKUP_INFORMATION)
USN_JOURNAL_DATA_V0 USN(업데이트 시퀀스 번호) 변경 저널, 해당 레코드 및 용량을 나타냅니다. (USN_JOURNAL_DATA_V0)
USN_JOURNAL_DATA_V1 USN(업데이트 시퀀스 번호) 변경 저널, 해당 레코드 및 용량을 나타냅니다. U
USN_JOURNAL_DATA_V2 USN(업데이트 시퀀스 번호) 변경 저널, 해당 레코드 및 용량을 나타냅니다. 이 구조는 FSCTL_QUERY_USN_JOURNAL 제어 코드의 출력 버퍼입니다.
USN_RANGE_TRACK_OUTPUT FSCTL_USN_TRACK_MODIFIED_RANGES 제어 코드에서 반환된 USN(업데이트 시퀀스 번호)을 포함합니다.
USN_RECORD_COMMON_HEADER USN_RECORD_V2, USN_RECORD_V3 및 USN_RECORD_V4에서 공통되는 USN(업데이트 시퀀스 번호) 공통 헤더에 대한 정보를 포함합니다.
USN_RECORD_EXTENT USN(업데이트 시퀀스 번호) 레코드 범위에 대한 오프셋과 길이를 포함합니다.
USN_RECORD_V2 USN(업데이트 시퀀스 번호) 변경 저널 버전 2.0 레코드에 대한 정보를 포함합니다.
USN_RECORD_V3 USN(업데이트 시퀀스 번호) 변경 저널 버전 3.0 레코드에 대한 정보를 포함합니다.
USN_RECORD_V4 USN(업데이트 시퀀스 번호) 변경 저널 버전 4.0 레코드에 대한 정보를 포함합니다. 버전 2.0 및 3.0 레코드는 각각 USN_RECORD_V2(USN_RECORD라고도 함) 및 USN_RECORD_V3 구조로 정의됩니다.

USN_TRACK_MODIFIED_RANGES FSCTL_USN_TRACK_MODIFIED_RANGES 제어 코드를 사용하는 USN(업데이트 시퀀스 번호) 변경 저널의 범위 추적 매개 변수에 대한 정보를 포함합니다.
VERIFY_INFORMATION 디스크 익스텐트 확인에 사용되는 정보를 포함합니다.
VOLUME_BITMAP_BUFFER 디스크를 차지하고 있으며 사용 가능한 클러스터를 나타냅니다.
VOLUME_DISK_EXTENTS 디스크에서의 실제 위치를 나타냅니다.
VOLUME_GET_GPT_ATTRIBUTES_INFORMATION IOCTL_VOLUME_GET_GPT_ATTRIBUTES 제어 코드를 사용하여 가져온 볼륨 특성을 포함합니다.
WIM_ENTRY_INFO 볼륨에서 호스트되는 각 WIM 데이터 원본과 관련된 메타데이터를 정의합니다.
WIM_EXTERNAL_FILE_INFO WOF_PROVIDER_WIM에서 제공하는 파일과 관련된 메타데이터를 정의합니다.
WIN32_FILE_ATTRIBUTE_DATA 파일 또는 디렉터리에 대한 특성 정보를 포함합니다.
WIN32_FIND_DATAA FindFirstFile, FindFirstFileEx 또는 FindNextFile 함수에서 찾은 파일에 대한 정보를 포함합니다. (ANSI)
WIN32_FIND_DATAW FindFirstFile, FindFirstFileEx 또는 FindNextFile 함수에서 찾은 파일에 대한 정보를 포함합니다. (유니코드)
WIN32_FIND_STREAM_DATA FindFirstStreamW 또는 FindNextStreamW 함수에서 찾은 스트림에 대한 정보를 포함합니다.

WOF_FILE_COMPRESSION_INFO_V1 WOF_PROVIDER_FILE에서 제공하는 파일과 관련된 메타데이터를 정의합니다.

데이터 중복 제거 API

아티클2023. 08. 22.

데이터 중복 제거 API 기술에 대한 개요입니다.

데이터 중복 제거 API를 개발하려면 다음 헤더가 필요합니다.

- ddpbackup.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 데이터 중복 제거 API

열거형

DEDUP_BACKUP_SUPPORT_PARAM_TYPE 데이터 중복 제거가 최적화되지 않은 복원 또는 최적화된 복원을 수행해야 하는지 여부를 나타냅니다.

Functions

OrderContainersRestore 이 메서드는 대상 파일을 검색하는 데 필요한 보류 중인 읽기의 순서에 영향을 주는 기능을 애플리케이션에 제공합니다.
PreviewContainerRead 지정된 컨테이너 파일 익스텐트에서 보류 중인 읽기 시퀀스의 미리 보기를 애플리케이션에 제공합니다.
ReadBackupFile 백업 저장소에 있는 데이터 중복 제거 저장소 메타데이터 또는 컨테이너 파일에서 데이터를 읽습니다.
RestoreFiles 완전히 최적화된 버전의 파일(재분석 지점) 및 데이터 중복 제거 저장소가 포함된 백업 저장소의 파일 집합을 다시 구성합니다.

인터페이스

IDedupBackupSupport 데이터 중복 제거 재분석 지점, 메타데이터 및 컨테이너 파일의 복사본을 포함하는 백업 저장소에서 파일을 복원하는 방법을 제공합니다.
IDedupReadFileCallback 백업 애플리케이션에서 구현되는 콜백 인터페이스로, 데이터 중복 제거가 백업 저장소에 있는 메타데이터 및 컨테이너 파일에서 콘텐츠를 읽고 필요에 따라 복원 효율성을 향상시킬 수 있습니다.

구조체

DDP_FILE_EXTENT DDP_FILE_EXTENT ReadBackupFile에 대한 보류 중인 호출에서 읽을 파일의 데이터 범위를 나타냅니다.
DEDUP_CONTAINER_EXTENT 논리 컨테이너 파일은 백업 저장소의 단일 세그먼트 또는 여러 세그먼트에 저장될 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

데이터 교환

아티클2023. 08. 22.

데이터 교환 기술 개요입니다.

데이터 교환을 개발하려면 다음 헤더가 필요합니다.

- dde.h
- ddeml.h
- wingdi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 데이터 교환

Functions

AddAtomA 로컬 원자 테이블에 문자열을 추가하고 문자열을 식별하는 고유 값(원자)을 반환합니다. (ANSI)
AddAtomW 로컬 원자 테이블에 문자열을 추가하고 문자열을 식별하는 고유 값(원자)을 반환합니다. (유니코드)
AddClipboardFormatListener 지정된 창을 시스템 유지 관리 클립보드 형식 수신기 목록에 배치합니다.
ChangeClipboardChain 클립보드 뷰어 체인에서 지정된 창을 제거합니다.
CloseClipboard 클립보드를 닫습니다.
CountClipboardFormats 현재 클립보드에 있는 다양한 데이터 형식의 수를 검색합니다.
DdeAbandonTransaction 지정된 비동기 트랜잭션을 포기하고 트랜잭션과 연결된 모든 리소스를 해제합니다.

DdeAccessData 지정된 DDE(동적 데이터 교환) 개체의 데이터에 대한 액세스를 제공합니다. 애플리케이션이 개체의 데이터에 대한 액세스를 완료한 경우 DdeUnaccessData 함수를 호출해야 합니다.
DdeAddData 지정된 DDE(동적 데이터 교환) 개체에 데이터를 추가합니다.
DdeClientTransaction 클라이언트와 서버 간의 데이터 트랜잭션을 시작합니다. DDE(동적 데이터 교환) 클라이언트 애플리케이션만 이 함수를 호출할 수 있으며 애플리케이션은 서버와의 대화를 설정한 후에만 사용할 수 있습니다.
DdeCmpStringHandles 두 문자열 핸들의 값을 비교합니다. 문자열 핸들의 값은 연결된 문자열의 경우와 관련이 없습니다.
DdeConnect 지정된 서비스 이름 및 토픽 이름 쌍을 지원하는 서버 애플리케이션과의 대화를 설정합니다. 이러한 서버가 두 개 이상 있는 경우 시스템은 하나만 선택합니다.
DdeConnectList 지정된 서비스 이름 및 토픽 이름 쌍을 지원하는 모든 서버 애플리케이션과 대화를 설정합니다.
DdeCreateDataHandle DDE(동적 데이터 교환) 개체를 만들고 지정된 버퍼의 데이터로 개체를 채웁니다. DDE 애플리케이션은 파트너 애플리케이션에 데이터를 전달하는 트랜잭션 중에 이 함수를 사용합니다.
DdeCreateStringHandleA 지정된 문자열을 식별하는 핸들을 만듭니다. DDE(동적 데이터 교환) 클라이언트 또는 서버 애플리케이션은 문자열 핸들을 매개 변수로 다른 DDEML(동적 데이터 교환 관리 라이브러리) 함수에 전달할 수 있습니다. (ANSI)
DdeCreateStringHandleW 지정된 문자열을 식별하는 핸들을 만듭니다. DDE(동적 데이터 교환) 클라이언트 또는 서버 애플리케이션은 문자열 핸들을 매개 변수로 다른 DDEML(동적 데이터 교환 관리 라이브러리) 함수에 전달할 수 있습니다. (유니코드)
DdeDisconnect DdeConnect 또는 DdeConnectList 함수에서 시작한 대화를 종료하고 지정된 대화 핸들을 무효화합니다.

DdeDisconnectList 지정된 대화 목록을 삭제하고 목록과 연결된 모든 대화를 종료합니다.
DdeEnableCallback 호출 애플리케이션에서 현재 설정한 특정 대화 또는 모든 대화에 대해 트랜잭션을 사용하거나 사용하지 않도록 설정합니다.
DdeFreeDataHandle DDE(동적 데이터 교환) 개체를 해제하고 개체와 연결된 데이터 핸들을 삭제합니다.
DdeFreeStringHandle 호출 애플리케이션에서 문자열 핸들을 해제합니다.
DdeGetData 지정된 DDE(동적 데이터 교환) 개체의 데이터를 지정된 로컬 버퍼에 복사합니다.
DdeGetLastError DDEML(동적 데이터 교환 관리 라이브러리) 함수의 실패로 설정된 가장 최근의 오류 코드를 검색하고 오류 코드를 DMLERR_NO_ERROR 다시 설정합니다.
DdeImpersonateClient DDE 클라이언트 대화에서 DDE(동적 데이터 교환) 클라이언트 애플리케이션을 가장합니다.
DdeInitializeA DDEML(동적 데이터 교환 관리 라이브러리)에 애플리케이션을 등록합니다. 애플리케이션은 다른 DDEML(동적 데이터 교환 관리 라이브러리) 함수를 호출하기 전에 이 함수를 호출해야 합니다. (ANSI)
DdeInitializeW DDEML(동적 데이터 교환 관리 라이브러리)에 애플리케이션을 등록합니다. 애플리케이션은 다른 DDEML(동적 데이터 교환 관리 라이브러리) 함수를 호출하기 전에 이 함수를 호출해야 합니다. (유니코드)
DdeKeepStringHandle 지정된 핸들과 연결된 사용 횟수를 증분합니다.
DdeNameService DDE(동적 데이터 교환) 서버에서 지원하는 서비스 이름을 등록하거나 등록 취소합니다.

DdePostAdvise 시스템이 지정된 토픽 및 항목에 대한 활성 조언 루프를 사용하여 각 클라이언트에 대한 호출(서버) 애플리케이션의 DDE(동적 데이터 교환) 콜백 함수에 XTYP_ADVREQ 트랜잭션을 보내도록 합니다.
DdeQueryConvInfo DDE(동적 데이터 교환) 트랜잭션 및 트랜잭션이 발생하는 대화에 대한 정보를 검색합니다.
DdeQueryNextServer 지정된 대화 목록에서 다음 대화 핸들을 검색합니다.
DdeQueryStringA 문자열 핸들과 연결된 텍스트를 버퍼에 복사합니다. (ANSI)
DdeQueryStringW 문자열 핸들과 연결된 텍스트를 버퍼에 복사합니다. (유니코드)
DdeReconnect 클라이언트 DDEML(동적 데이터 교환 관리 라이브러리) 애플리케이션이 클라이언트와의 대화를 종료한 서비스와의 대화를 다시 설정하려고 시도할 수 있도록 합니다.
DdeSetQualityOfService 향후 DDE 대화에 필요한 원시 DDE(동적 데이터 교환) 애플리케이션의 QOS(서비스 품질)를 지정합니다.
DdeSetUserHandle 애플리케이션 정의 값을 대화 핸들 또는 트랜잭션 식별자와 연결합니다. 이는 비동기 트랜잭션 처리를 간소화하는 데 유용합니다. 애플리케이션은 DdeQueryConvInfo 함수를 사용하여 이 값을 검색할 수 있습니다.
DdeUnaccessData DDE(동적 데이터 교환) 개체에 액세스하지 않습니다. 애플리케이션이 개체에 대한 액세스를 완료한 후 이 함수를 호출해야 합니다.
DdeUninitialize 호출 애플리케이션과 연결된 모든 DDEML(동적 데이터 교환 관리 라이브러리) 리소스를 해제합니다.
DeleteAtom

로컬 문자열 원자의 참조 수를 줄입니다. 원자의 참조 수가 0으로 줄면 DeleteAtom은 로컬 원자 테이블에서 원자와 연결된 문자열을 제거합니다.
EmptyClipboard 클립보드를 비우고 클립보드의 데이터에 대한 핸들을 해제합니다. 그런 다음 함수는 클립보드가 열려 있는 창에 클립보드의 소유권을 할당합니다.
EnumClipboardFormats 클립보드에서 현재 사용할 수 있는 데이터 형식을 열거합니다.
FindAtomA 로컬 원자 테이블에서 지정된 문자열을 검색하고 해당 문자열과 연결된 원자를 검색합니다. (ANSI)
FindAtomW 로컬 원자 테이블에서 지정된 문자열을 검색하고 해당 문자열과 연결된 원자를 검색합니다. (유니코드)
FreeDDElParam 게시된 DDE(동적 데이터 교환) 메시지의 lParam 매개 변수로 지정된 메모리를 해제합니다. 게시된 DDE 메시지를 수신하는 애플리케이션은 UnpackDDElParam 함수를 사용하여 lParam 값의 압축을 풉니다.
GetAtomNameA 지정된 로컬 원자와 연결된 문자열의 복사본을 검색합니다. (ANSI)
GetAtomNameW 지정된 로컬 원자와 연결된 문자열의 복사본을 검색합니다. (유니코드)
GetClipboardData 클립보드에서 지정된 형식으로 데이터를 검색합니다. 클립보드가 이전에 열렸어야 합니다.
GetClipboardFormatNameA 클립보드에서 지정된 등록된 형식의 이름을 검색합니다. 함수는 지정된 버퍼에 이름을 복사합니다. (ANSI)
GetClipboardFormatNameW 클립보드에서 지정된 등록된 형식의 이름을 검색합니다. 함수는 지정된 버퍼에 이름을 복사합니다. (유니코드)

GetClipboardOwner 클립보드의 현재 소유자 창 핸들을 검색합니다.
GetClipboardSequenceNumber 현재 창 스테이션의 클립보드 시퀀스 번호를 검색합니다.
GetClipboardViewer 클립보드 뷰어 체인의 첫 번째 창에 대한 핸들을 검색합니다.
GetOpenClipboardWindow 현재 클립보드가 열려 있는 창에 대한 핸들을 검색합니다.
GetPriorityClipboardFormat 지정된 목록에서 사용 가능한 첫 번째 클립보드 형식을 검색합니다.
GetUpdatedClipboardFormats 현재 지원되는 클립보드 형식을 검색합니다.
GlobalAddAtomA 전역 원자 테이블에 문자열을 추가하고 문자열을 식별하는 고유 값(원자)을 반환합니다. (GlobalAddAtomA)
GlobalAddAtomExA 전역 원자 테이블에 문자열을 추가하고 문자열을 식별하는 고유 값(원자)을 반환합니다. (GlobalAddAtomExA)
GlobalAddAtomExW 전역 원자 테이블에 문자열을 추가하고 문자열을 식별하는 고유 값(원자)을 반환합니다. (GlobalAddAtomExW)
GlobalAddAtomW 전역 원자 테이블에 문자열을 추가하고 문자열을 식별하는 고유 값(원자)을 반환합니다. (GlobalAddAtomW)
GlobalDeleteAtom 전역 문자열 원자의 참조 수를 줄입니다. 원자의 참조 수가 0에 도달하면 GlobalDeleteAtom은 글로벌 원자 테이블에서 원자와 연결된 문자열을 제거합니다.

GlobalFindAtomA 전역 원자 테이블에서 지정된 문자열을 검색하고 해당 문자열과 연결된 전역 원자를 검색합니다. (ANSI)
GlobalFindAtomW 전역 원자 테이블에서 지정된 문자열을 검색하고 해당 문자열과 연결된 전역 원자를 검색합니다. (유니코드)
GlobalGetAtomNameA 지정된 전역 원자와 연결된 문자열의 복사본을 검색합니다. (ANSI)
GlobalGetAtomNameW 지정된 전역 원자와 연결된 문자열의 복사본을 검색합니다. (유니코드)
ImpersonateDdeClientWindow DDE(동적 데이터 교환) 서버 애플리케이션이 DDE 클라이언트 애플리케이션의 보안 컨텍스트를 가장할 수 있도록 합니다. 이렇게 하면 권한이 없는 DDE 클라이언트로부터 보안 서버 데이터가 보호됩니다.
InitAtomTable 로컬 원자 테이블을 초기화하고 해시 버킷 수를 지정된 크기로 설정합니다.
IsClipboardFormatAvailable 클립보드에 지정된 형식의 데이터가 포함되어 있는지 여부를 확인합니다.
MAKEINTATOM 지정된 원자를 문자열로 변환하므로 원자 또는 문자열을 허용하는 함수에 전달할 수 있습니다.
OpenClipboard 검사를 위해 클립보드를 열고 다른 애플리케이션이 클립보드 콘텐츠를 수정하지 못하도록 합니다.
PackDDElParam DDE(동적 데이터 교환) lParam 값을 프로세스 간에 DDE 데이터를 공유하는 데 사용되는 내부 구조로 압축합니다.
PFNCALLBACK DDEML(동적 데이터 교환 관리 라이브러리) 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.

RegisterClipboardFormatA 새 클립보드 형식을 등록합니다. 그런 다음 이 형식을 유효한 클립보드 형식으로 사용할 수 있습니다. (ANSI)
RegisterClipboardFormatW 새 클립보드 형식을 등록합니다. 그런 다음 이 형식을 유효한 클립보드 형식으로 사용할 수 있습니다. (유니코드)
RemoveClipboardFormatListener 시스템이 유지 관리하는 클립보드 형식 수신기 목록에서 지정된 창을 제거합니다.
ReuseDDElParam 애플리케이션이 압축된 새 lParam을 할당하는 대신 압축된 DDE(동적 데이터 교환) lParam 매개 변수를 다시 사용할 수 있도록 합니다. 이 함수를 사용하면 압축된 DDE 메시지를 전달하는 애플리케이션의 재할당이 줄어듭니다.
SetClipboardData 지정된 클립보드 형식으로 클립보드에 데이터를 배치합니다.
SetClipboardViewer 클립보드 뷰어 체인에 지정된 창을 추가합니다. 클립보드 뷰어 창은 클립보드의 콘텐츠가 변경될 때마다 WM_DRAWCLIPBOARD 메시지를 받습니다. 이 함수는 이전 버전의 Windows와 이전 버전과의 호환성을 위해 사용됩니다.
UnpackDDElParam 게시된 DDE 메시지에서 받은 DDE(동적 데이터 교환) lParam 값의 압축을 풉니다.

구조체

CONVCONTEXT DDE(동적 데이터 교환) 클라이언트 애플리케이션에서 제공하는 정보를 포함합니다. 이 정보는 특수 언어 또는 언어 간 DDE 대화에 유용합니다.
CONVINFO DDE(동적 데이터 교환) 대화에 대한 정보를 포함합니다.

COPYDATASTRUCT WM_COPYDATA 메시지에서 다른 애플리케이션에 전달할 데이터를 포함합니다.
DDEACK DDE 애플리케이션이 WM_DDE_ACK 메시지의 일부로 파트너에게 전달하는 상태 플래그를 포함합니다.
DDEADVISE DDE 서버 애플리케이션이 조언 루프 중에 클라이언트 애플리케이션에 데이터를 보내는 방법을 지정하는 플래그를 포함합니다. 클라이언트는 WM_DDE_ADVISE 메시지의 일부로 DDEADVISE 구조체에 대한 핸들을 서버에 전달합니다.
DDEDATA WM_DDE_DATA 메시지의 일부로 전송된 데이터 및 데이터에 대한 정보를 포함합니다.
DDEML_MSG_HOOK_DATA DDE(동적 데이터 교환) 메시지에 대한 정보를 포함하고 메시지에서 참조하는 데이터에 대한 읽기 권한을 제공합니다. 이 구조는 DDEML(동적 데이터 교환 관리 라이브러리) 모니터링 애플리케이션에서 사용됩니다.
DDEPOKE WM_DDE_POKE 메시지의 일부로 전송된 데이터 및 데이터에 대한 정보를 포함합니다.
HSZPAIR DDE 서비스 이름 및 토픽 이름을 포함합니다. DDE 서버 애플리케이션은 XTYP_WILDCONNECT 트랜잭션 중에 이 구조를 사용하여 지원하는 서비스 토픽 쌍을 열거할 수 있습니다.
METAFILEPICT 클립보드를 통해 메타파일 데이터를 교환하는 데 사용되는 메타파일 그림 형식을 정의합니다.
MONCBSTRUCT 현재 DDE(동적 데이터 교환) 트랜잭션에 대한 정보를 포함합니다. DDE 디버깅 애플리케이션은 시스템이 다른 애플리케이션의 DDE 콜백 함수에 전달하는 트랜잭션을 모니터링할 때 이 구조를 사용할 수 있습니다.
MONCONVSTRUCT DDE(동적 데이터 교환) 대화에 대한 정보를 포함합니다. DDE 모니터링 애플리케이션은 이 구조를 사용하여 설정되었거나 종료된 대화에 대한 정보를 가져올 수 있습니다.

MONERRSTRUCT 현재 DDE(동적 데이터 교환) 오류에 대한 정보를 포함합니다. DDE 모니터링 애플리케이션은 이 구조를 사용하여 DDE 관리 라이브러리 함수에서 반환된 오류를 모니터링할 수 있습니다.
MONHSZSTRUCTA DDE(동적 데이터 교환) 문자열 핸들에 대한 정보를 포함합니다. DDE 모니터링 애플리케이션은 DDE 관리 라이브러리의 문자열 관리자 구성 요소의 활동을 모니터링할 때 이 구조를 사용할 수 있습니다. (ANSI)
MONHSZSTRUCTW DDE(동적 데이터 교환) 문자열 핸들에 대한 정보를 포함합니다. DDE 모니터링 애플리케이션은 DDE 관리 라이브러리의 문자열 관리자 구성 요소의 활동을 모니터링할 때 이 구조를 사용할 수 있습니다. (유니코드)
MONLINKSTRUCT DDE(동적 데이터 교환) 조언 루프에 대한 정보를 포함합니다. DDE 모니터링 애플리케이션은 이 구조를 사용하여 시작되거나 종료된 조언 루프에 대한 정보를 얻을 수 있습니다.
MONMSGSTRUCT DDE(동적 데이터 교환) 메시지에 대한 정보를 포함합니다. DDE 모니터링 애플리케이션은 이 구조를 사용하여 전송되거나 게시된 DDE 메시지에 대한 정보를 가져올 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

디버깅 및 작성 지원

아티클2024. 02. 05.

디버깅 및 작성 지원 기술에 대한 개요입니다.

디버깅 및 제작 지원을 개발하려면 다음 헤더가 필요합니다.

- webapplication.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 디버깅 및 작성 지원

Functions

테이블 확장

조언 클라이언트가 이벤트를 수신할 수 있도록 연결을 설정합니다.
BeforeNavigate 지정된 호스트(창 또는 프레임 세트 요소)에서 탐색이 발생하기 전에 발생합니다.
CancelPendingActivation 보류 중인 활성화를 취소합니다.
DocumentComplete 탐색 중인 문서가 ReadyState_Complete 도달하면 발생합니다.
DownloadBegin 페이지 다운로드가 시작되었습니다.
DownloadComplete 페이지 다운로드가 완료되었습니다.
get_AuthoringClientBinary WWAHost 프로세스에 로드할 DLL의 전체 로컬 경로를 가져옵니다.

get_Document 현재 최상위 문서의 HTML 문서 개체 모델을 가져옵니다.
get_HWND 현재 WWAHost 창의 핸들을 가져옵니다.
NavigateComplete 탐색 중인 문서가 표시되고 탐색 스택에 들어갈 때 발생합니다.
NavigateError 바인딩 오류가 발생할 때 발생합니다(창 또는 프레임 세트 요소).
OnCssChanged CSS(Cascading Style Sheets)가 변경되었음을 제작 앱에 알 수 있습니다.
OnPaint 제작 앱에 앱의 일부가 그려졌다는 것을 알 수 있습니다.
새로 고침 서버에 'Pragma:no-cache' HTTP 헤더를 보내지 않고 현재 문서를 새로 고칩니다.
RegisterAuthoringClientFunctionType 작성 이진 파일로 사용할 DLL(동적 연결 라이브러리)에서 애플리케이션 정의 함수에 대한 포인터를 정의합니다. 앱 호스트가 제작 모드에서 시작되면 이 함수가 호출되어 제작 이진 파일을 초기화합니다.
ScriptError 처리되지 않은 스크립트 오류가 발생할 때 발생합니다.
SecurityProblem 인증 문제에 대해 제작 앱에 알합니다.
Unadvise 이전에 설정된 연결을 제거합니다.
UnregisterAuthoringClientFunctionType

RegisterAuthoringClientFunctionType 함수에 등록된 애플리케이션 정의 함수의 등록을 취소합니다. 이 함수는 앱 호스트가 종료되면 호출됩니다.

인터페이스

테이블 확장

IWebApplicationActivation 디버깅 애플리케이션이 활성화를 관리할 수 있도록 합니다.
IWebApplicationAuthoringMode WWAHost 프로세스에 로드할 작성 이진 파일에 대한 전체 로컬 경로를 제공합니다.
IWebApplicationHost WWAHost에서 구현하는 메서드 및 속성을 노출합니다.
IWebApplicationNavigationEvents 디버깅 또는 제작 앱이 탐색 이벤트 알림을 받을 수 있도록 합니다.
IWebApplicationScriptEvents 디버깅 또는 제작 앱이 스크립팅 엔진 이벤트에 대한 알림을 받을 수 있도록 합니다.
IWebApplicationUIEvents 디버깅 또는 제작 앱이 사용자 인터페이스 이벤트에 대한 알림을 받고 사용자 상호 작용이 필요한 이벤트에 응답할 수 있도록 합니다.
IWebApplicationUpdateEvents 작성 앱이 디자이너 이벤트에 대한 알림을 받고 해당 이벤트에 응답할 수 있도록 합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

배달 최적화

아티클2023. 08. 22.

배달 최적화 기술 개요입니다.

배달 최적화를 개발하려면 다음 헤더가 필요합니다.

- deliveryoptimization.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 배달 최적화

열거형

DODownloadCostPolicy **DODownloadProperty_CostPolicy** 속성과 연결된 비용 정책 옵션의 ID를 지정합니다.
DODownloadProperty 배달 최적화 다운로드 작업의 속성 ID를 지정합니다.
DODownloadState **DO_DOWNLOAD_STATUS** 구조의 일부인 현재 다운로드 상태의 ID를 지정합니다.

Functions

중단 다운로드를 중단합니다.
CreateDownload 새 다운로드를 만듭니다.
EnumDownloads 기존 다운로드를 열거하는 데 사용되는 열거자 개체에 대한 인터페이스 포인터를 검색합니다.

Finalize 다운로드를 완료합니다.
GetProperty 특정 다운로드 속성이 포함된 **VARIANT** 에 대한 포인터를 검색합니다.
GetStatus IDODownload::GetStatus는 다운로드의 현재 상태 반영하는 DO_DOWNLOAD_STATUS 구조체에 대한 포인터를 검색합니다.
OnStatusChange 배달 최적화는 다운로드 상태 변경되면 언제든지 이 메서드의 구현을 호출합니다.
일시 중지 다운로드를 일시 중지합니다.
SetProperty 다운로드 속성을 설정합니다.
시작 다운로드를 시작하거나 다시 시작합니다.

인터페이스

IDO다운로드 다운로드를 시작하고 관리하는 데 사용됩니다.
IDODownloadStatusCallback 다운로드에 대한 알림을 받는 데 사용됩니다.
IDOManager 새 다운로드를 만들고 기존 다운로드를 열거하는 데 사용됩니다.

구조체

DO_DOWNLOAD_ENUM_CATEGORY **IDOManager::EnumDownloads**에서 특정 속성 값으로 다운로드 열거형을 필터링하는 데 사용됩니다.
DO_DOWNLOAD_RANGE 파일에서 다운로드할 단일 바이트 범위를 식별합니다.
DO_DOWNLOAD_RANGES_INFO 파일에서 다운로드할 바이트 범위의 배열을 식별합니다.
DO_DOWNLOAD_STATUS 특정 다운로드의 상태 가져오는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

데스크톱 창 관리자(DWM)

아티클2023. 08. 22.

DWM(데스크톱 창 관리자) 기술 개요입니다.

DWM(데스크톱 창 관리자)을 개발하려면 다음 헤더가 필요합니다.

- dwmapi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 데스크톱 창 관리자(DWM)

열거형

DWM_SHOWCONTACT 시각적 피드백을 제공해야 하는 입력 작업을 지정합니다.
DWM_SOURCE_FRAME_SAMPLING DwmSetPresentParameters 함수에서 프레임 샘플링 유형을 지정하는 데 사용하는 플래그입니다.
DWM_SYSTEMBACKDROP_TYPE 클라이언트가 아닌 영역 뒤를 포함하여 창의 시스템 그리기 배경 자료를 지정하기 위한 플래그입니다.
DWM_TAB_WINDOW_REQUIREMENTS 이 열거형은 Windows에서 구현되지 않은 실험적 기능을 지원했습니다. 영향을 주지 않으며 사용해서는 안 됩니다.
DWM_WINDOW_CORNER_PREFERENCE DWM_WINDOW_CORNER_PREFERENCE 열거형(dwmapi.h)은 창의 둥근 모서리 기본 설정을 지정합니다.
DWMFLIP3DWINDOWPOLICY DwmSetWindowAttribute 함수에서 Flip3D 창 정책을 지정하는 데 사용하는 플래그입니다.
DWMNCRENDERINGPOLICY

DwmSetWindowAttribute 함수에서 비 클라이언트 영역 렌더링 정책을 지정하는 데 사용하는 플래그입니다.
DWMTRANSITION_OWNEDWINDOW_TARGET 대상을 식별합니다.
DWMWINDOWATTRIBUTE DwmGetWindowAttribute 및 DwmSetWindowAttribute 함수에서 사용하는 플래그입니다.
GESTURE_TYPE DwmRenderGesture에 지정된 제스처 유형을 식별합니다.

Functions

DwmAttachMilContent 이 함수는 구현되지 않습니다. (DwmAttachMilContent)
DwmDefWindowProc DWM(데스크톱 창 관리자)에 대한 기본 창 프로시저는 클라이언트가 아닌 영역 내에서 테스트를 적중합니다.
DwmDetachMilContent 이 함수는 구현되지 않습니다. (DwmDetachMilContent)
DwmEnableBlurBehindWindow 지정된 창에서 흐림 효과를 사용하도록 설정합니다.
DwmEnableComposition DWM(데스크톱 창 관리자) 컴퍼지션을 사용하거나 사용하지 않도록 설정합니다.
DwmEnableMMCSS 호출 프로세스가 활성 상태인 동안 DWM(데스크톱 창 관리자)에 MMCSS(멀티미디어 클래스 일정 서비스) 예약을 옵트인하거나 옵트아웃하도록 알립니다.
DwmExtendFrameIntoClientArea

창 프레임을 클라이언트 영역으로 확장합니다.
DwmFlush 현재 미해결 상태인 모든 Microsoft DirectX 표면 업데이트가 수행될 때까지 호출자를 차단하는 플러시 호출을 실행합니다. 이렇게 하면 매우 복잡한 장면 또는 우선 순위가 매우 낮은 호출 프로세스를 보상합니다.
DwmGetColorizationColor DWM(데스크톱 창 관리자) 유리 컴퍼지션에 사용되는 현재 색을 검색합니다.
DwmGetCompositionTimingInfo 지정된 창에 대한 현재 컴퍼지션 타이밍 정보를 검색합니다.
DwmGetGraphicsStreamClient 이 함수는 구현되지 않습니다. (DwmGetGraphicsStreamClient)
DwmGetGraphicsStreamTransformHint 이 함수는 구현되지 않습니다. (DwmGetGraphicsStreamTransformHint)
DwmGetTransportAttributes 전송 특성을 검색합니다.
DwmGetUnmetTabRequirements 이 함수는 Windows에서 구현되지 않은 실험적 기능의 일부였습니다. 영향을 주지 않으며 사용해서는 안 됩니다.
DwmGetWindowAttribute 창에 적용된 지정된 DWM(데스크톱 창 관리자) 특성의 현재 값을 검색합니다.
DwmInvalidateIconicBitmaps 미리 보기와 피킹 표현을 모두 창에서 이전에 제공한 모든 아이콘 비트맵을 새로 고쳐야 함을 나타내기 위해 애플리케이션에서 호출됩니다.
DwmIsCompositionEnabled DWM(데스크톱 창 관리자) 컴퍼지션을 사용할 수 있는지 여부를 나타내는 값을 가져옵니다. Windows 7 이하를 실행하는 컴퓨터의 애플리케이션은 WM_DWMCOMPOSITIONCHANGED 알림을 처리하여 컴퍼지션 상태 변경을 수신 대기할 수 있습니다.
DwmModifyPreviousDxFrameDuration

이전 프레임이 표시될 모니터 새로 고침 수를 변경합니다. DwmModifyPreviousDxFrameDuration은 더 이상 지원되지 않습니다. Windows 8.1 시작해서 DwmModifyPreviousDxFrameDuration에 대한 호출은 항상 E_NOTIMPL 반환합니다.
DwmQueryThumbnailSourceSize DWM(데스크톱 창 관리자) 축소판 그림의 원본 크기를 검색합니다.
DwmRegisterThumbnail 대상 창과 원본 창 간에 DWM(데스크톱 창 관리자) 썸네일 관계를 만듭니다.
DwmRenderGesture 터치 접촉이 제스처로 인식되었으며 DWM이 해당 제스처에 대한 피드백을 작성해야 한다는 것을 DWM(데스크톱 창 관리자)에 알립니다.
DwmSetDxFrameDuration 표시된 프레임을 표시할 모니터 새로 고침 수를 설정합니다. DwmSetDxFrameDuration은 더 이상 지원되지 않습니다. Windows 8.1 DwmSetDxFrameDuration 호출은 항상 E_NOTIMPL 반환합니다.
DwmSetIconicLivePreviewBitmap 창 또는 탭의 라이브 미리 보기(미리 보기 미리 보기라고도 함)를 표시하도록 정적 아이콘 비트맵을 설정합니다. 작업 표시줄은 이 비트맵을 사용하여 창 또는 탭의 전체 크기 미리 보기를 표시할 수 있습니다.
DwmSetIconicThumbnail 썸네일 표현으로 사용할 창 또는 탭의 정적 아이콘 비트맵을 설정합니다. 작업 표시줄은 이 비트맵을 창 또는 탭의 축소판 그림 스위치 대상으로 사용할 수 있습니다.
DwmSetPresentParameters 프레임 컴퍼지션에 대한 현재 매개 변수를 설정합니다. DwmSetPresentParameters는 더 이상 지원되지 않습니다. Windows 8.1 DwmSetPresentParameters에 대한 호출은 항상 E_NOTIMPL 반환합니다.
DwmSetWindowAttribute 창에 대한 클라이언트가 아닌 렌더링 특성인 DWM(Desktop Window Manager) 값을 설정합니다.
DwmShowContact 특정 터치 또는 펜 접촉에 대한 응답으로 그릴 시각적 피드백 유형을 지정하기 위해 앱 또는 프레임워크에서 호출됩니다.
DwmTetherContact

터치 및 끌기 상호 작용의 그래픽 피드백을 사용자에게 사용하도록 설정합니다.
DwmTransitionOwnedWindow DWM(데스크톱 창 관리자)을 사용하여 도구 창의 애니메이션을 조정합니다.
DwmUnregisterThumbnail DwmRegisterThumbnail 함수에서 만든 DWM(데스크톱 창 관리자) 썸네일 관계를 제거합니다.
DwmUpdateThumbnailProperties DWM(데스크톱 창 관리자) 썸네일의 속성을 업데이트.

구조체

DWM_BLURBEHIND DWM(데스크톱 창 관리자) 흐림 효과 속성을 지정합니다. DwmEnableBlurBehindWindow 함수에서 사용됩니다.
DWM_PRESENT_PARAMETERS 프레임 컴퍼지션에 대한 DWM(데스크톱 창 관리자) 비디오 프레임 매개 변수를 지정합니다. DwmSetPresentParameters 함수에서 사용됩니다.
DWM_THUMBNAIL_PROPERTIES DWM(데스크톱 창 관리자) 썸네일 속성을 지정합니다. DwmUpdateThumbnailProperties 함수에서 사용됩니다.
DWM_TIMING_INFO DWM(데스크톱 창 관리자) 컴퍼지션 타이밍 정보를 지정합니다. DwmGetCompositionTimingInfo 함수에서 사용됩니다.
MilMatrix3x2D 변환을 설명하는 3x2 행렬을 지정합니다.
UNSIGNED_RATIO DWM(데스크톱 창 관리자) API에서 사용하는 데이터 형식을 정의합니다. 제네릭 비율을 나타내며 단일 API 내에서도 다양한 용도 및 단위로 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

개발자 라이선스

아티클2023. 08. 22.

개발자 라이선스 기술 개요입니다.

개발자 라이선스를 개발하려면 다음 헤더가 필요합니다.

- wsdevlicensing.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 개발자 라이선스

Functions

AcquireDeveloperLicense 개발자 라이선스를 획득합니다.
CheckDeveloperLicense 개발자 라이선스가 있는지 확인합니다.
RemoveDeveloperLicense 개발자 라이선스를 제거합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

개발자 정보

아티클2023. 03. 09.

개발자 노트 기술 개요입니다.

개발자 노트를 개발하려면 다음 헤더가 필요합니다.

- advpub.h
- apiquery2.h
- appcompatapi.h
- aux_ulib.h
- avrfsdk.h
- camerauicontrol.h
- capi.h
- comppkgsup.h
- dciman.h
- ddrawgdi.h
- editionupgradehelper.h
- exdisp.h
- exposeenums2managed.h
- fci.h
- fdi.h
- fdi_fci_types.h
- featurestagingapi.h
- fhcfg.h
- fhsvcctl.h
- filehc.h
- icwcfg.h
- ime.h
- isolatedapplauncher.h
- iwscapi.h
- lmaccess.h
- loadperf.h
- msxml.h
- ntsecpkg.h
- rpcndr.h
- rtlsupportapi.h
- stralign.h
- tcpioctl.h
- tdiinfo.h

- vdmdbg.h
- wininet.h
- winnls32.h
- winreg.h
- wscapi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 개발자 정보
- Windows 헤더 사용

IOCTLs

IOCTL_TCP_QUERY_INFORMATION_EX TCP/IP 드라이버에서 정보를 검색합니다.

열거형

eAvrfResourceTypes VerifierEnumerateResource 함수를 사용하여 열거할 수 있는 리소스 유형을 지정합니다.
eHANDLE_TRACE_OPERATIONS 발생한 핸들 작업의 유형을 식별합니다.
eHeapAllocationState 현재 힙 할당 상태를 지정합니다.
eHeapEnumerationLevel 열거형 작업을 계속할지 중지할지 여부를 결정합니다.
eUserAllocationState 애플리케이션의 현재 힙 할당 상태를 지정합니다.
FEATURE_CHANGE_TIME 이 열거형은 인프라 전용입니다. (FEATURE_CHANGE_TIME)

FEATURE_ENABLED_STATE 이 열거형은 인프라 전용입니다. (FEATURE_ENABLED_STATE)
FH_BACKUP_STATUS 파일 기록 백업을 사용할 수 있는지 여부를 지정합니다.
FH_DEVICE_VALIDATION_RESULT 스토리지 디바이스 또는 네트워크 공유를 파일 기록 백업 대상으로 사용할 수 있는지 여부를 나타냅니다.
FH_LOCAL_POLICY_TYPE 파일 기록 기능에 대한 로컬 정책의 형식을 지정합니다. 각 로컬 정책에는 연결된 숫자 매개 변수가 있습니다.
FH_PROTECTED_ITEM_CATEGORY 포함 또는 제외 목록의 형식을 지정합니다.
FH_RETENTION_TYPES 백업 대상에서 이전 버전의 파일 및 폴더를 삭제할 수 있는 조건에 따라 지정합니다.
FH_TARGET_DRIVE_TYPES 파일 기록 백업 대상의 형식을 지정합니다.
FH_TARGET_PROPERTY_TYPE 백업 대상의 속성 형식을 지정합니다.
FIRMWARE_TYPE 펌웨어 유형을 지정합니다.
WSC_SECURITY_PRODUCT_STATE Windows 보안 센터에서 사용할 수 있는 보안 제품의 현재 상태를 정의합니다.
WSC_SECURITY_PROVIDER WSC(Windows 보안 Center)에서 모니터링하는 모든 서비스를 정의합니다.
WSC_SECURITY_PROVIDER_HEALTH WSC(Windows 보안 Center)에서 모니터링하는 모든 서비스에 대해 가능한 상태를 정의합니다.

WSC_SECURITY_SIGNATURE_STATUS 보안 제품의 현재 버전 상태를 Windows 보안 Center에 보고합니다.

Functions

_lclose _lclose 함수는 지정된 파일을 더 이상 읽거나 쓸 수 없도록 닫습니다. 이 함수는 16비트 버전의 Windows와의 호환성을 위해 제공됩니다. Win32 기반 애플리케이션은 CloseHandle 함수를 사용해야 합니다.
_lcreat 지정된 파일을 만들거나 엽니다.
_llseek 지정된 파일의 파일 포인터 위치를 변경합니다.
_lopen _lopen 함수는 기존 파일을 열고 파일 포인터를 파일의 시작 부분으로 설정합니다. 이 함수는 16비트 버전의 Windows와의 호환성을 위해 제공됩니다. Win32 기반 애플리케이션은 CreateFile 함수를 사용해야 합니다.
_lread _lread 함수는 지정된 파일에서 데이터를 읽습니다. 이 함수는 16비트 버전의 Windows와의 호환성을 위해 제공됩니다. Win32 기반 애플리케이션은 ReadFile 함수를 사용해야 합니다.
_lwrite 지정된 파일에 데이터를 씁니다.
AddRemoveExcludeRule 제외 목록에 제외 규칙을 추가하거나 목록에서 규칙을 제거합니다.
ApphelpCheckShellObject 애플리케이션에서 잘못된 확장 개체를 검색하고 실행하지 못하도록 차단하거나 수정할 수 있습니다.

AreDvdCodecsEnabled 현재 장치에서 DVD 코덱을 사용할 수 있는지 여부를 나타내는 값을 반환합니다.
AssociateContextWithName 이름 캐시에 이름을 삽입하여 지정된 FIO_CONTEXT 구조를 찾습니다.
AssociateFileEx 파일을 비시도 컨텍스트와 연결합니다.
AuxUlibInitialize Aux_ulib 라이브러리를 초기화합니다.
AuxUlibIsDLLSynchronizationHeld 호출자가 동기화 기본 형식을 보유하고 있는지 여부를 확인합니다.
AuxUlibSetSystemFileCacheSize 현재 파일 시스템 캐시 크기를 설정합니다.
AVRF_HANDLEOPERATION_ENUMERATE_CALLBACK 핸들 추적의 열거와 관련된 정보를 받습니다.
AVRF_HEAPALLOCATION_ENUMERATE_CALLBACK 힙 할당과 관련된 정보를 받습니다.
AVRF_RESOURCE_ENUMERATE_CALLBACK 힙 할당 또는 추적 정보 처리의 열거를 위해 특수 콜백 함수 중 하나에 대한 액세스를 제공합니다.
CACHE_DESTROY_CALLBACK 이름 캐시의 항목이 제거 될 때마다 호출 되는 함수입니다.
CACHE_READ_CALLBACK 캐시 내의 항목을 검사하는 데 도움이 되도록 캐시에 제공되는 콜백입니다.
CacheCreateFile 캐시에 파일을 만들거나 기존 파일을 찾습니다.

CacheRichCreateFile 캐시에 파일을 만들거나 기존 파일을 찾아서 캐시에 속성을 추가할 수 있습니다.
CanUpgrade 사용자에게 운영 체제를 업그레이드할 수 있는 충분한 권한이 있는지 확인하고 필요한 경우 관리자 권한으로 실행하라는 메시지를 표시합니다.
ChangeDefaultTargetRecommendation 현재 할당된 백업 대상이 컴퓨터가 속한 홈 그룹의 다른 구성원에게 권장되거나 권장되지 않도록 합니다.
CheckConnectionWizard CheckConnectionWizard 함수는 ICW(인터넷 연결 마법사)가 설치되어 있고 이전에 실행되지 않은지 확인합니다.
닫기 사용자 인터페이스 컨트롤을 닫습니다.
CoInstall 요청된 COM 서버 애플리케이션을 설치합니다.
CreateDefaultConfiguration 현재 사용자에 대한 기본 설정을 사용하여 파일 기록 구성 파일을 만들고 FhConfigMgr 개체에 로드합니다.
DCIBeginAccess 지정된 사각형을 기반으로 프레임 버퍼를 표시하는 액세스 포인터를 가져옵니다.
DCICloseProvider 디스플레이의 디바이스 컨텍스트를 닫습니다.
DCICreatePrimary 기본 표면을 만들고 표면 정보를 가져옵니다.
DCIDestroy 디스플레이 디바이스의 기본 표면을 제거합니다.

DCIEndAccess 디스플레이 프레임 버퍼에 대한 액세스를 해제합니다.
DCIOpenProvider 디스플레이의 디바이스 컨텍스트 핸들을 가져옵니다.
DdAttachSurface DdAttachSurface 함수는 두 개의 커널 모드 표면 표현을 연결합니다. GdiEntry11은 이 함수의 별칭으로 정의됩니다.
DdCreateDIBSection 해당 색 테이블을 디바이스와 공유하는 DIBSECTION 구조를 만듭니다. GdiEntry9는 이 함수의 별칭으로 정의됩니다.
DdCreateDirectDrawObject NtGdiDdCreateDirectDrawObject 함수에 대한 래퍼이며 Microsoft DirectDraw 개체의 커널 쪽 표현을 만듭니다.
DdCreateSurfaceObject NtGdiDdCreateSurfaceObject 함수에 대한 래퍼이며 커널 모드 표면 개체를 만듭니다. GdiEntry4는 이 함수의 별칭으로 정의됩니다.
DdDeleteDirectDrawObject NtGdiDdDeleteDirectDrawObject 함수에 대한 래퍼이며 이전에 DdCreateDirectDrawObject를 사용하여 만든 커널 모드 Microsoft DirectDraw 개체를 삭제합니다. GdiEntry3은 이 함수의 별칭으로 정의됩니다.
DdDeleteSurfaceObject NtGdiDdDeleteSurfaceObject 함수에 대한 래퍼이며 이전에 NtGdiDdCreateSurfaceObject에서 만든 커널 모드 표면 개체를 삭제합니다. GdiEntry5는 이 함수의 별칭으로 정의됩니다.
DdGetDC NtGdiDdGetDC 함수에 대한 래퍼이며 표시된 Microsoft DirectDraw 표면을 나타내는 GDI(Windows Graphics Device Interface) DC(디바이스 컨텍스트)를 반환합니다. GdiEntry7은 이 함수의 별칭으로 정의됩니다.
DdGetDxHandle DirectX API 메커니즘을 제어하는 커널 모드 진입점에 대한 후속 호출에 사용할 커널 모드 Microsoft DirectX API 핸들을 반환합니다.

DdQueryDirectDrawObject NtGdiDdQueryDirectDrawObject 함수에 대한 래퍼이며 이전에 만든 커널 모드 표현에서 기능을 쿼리합니다. GdiEntry2는 이 함수의 별칭으로 정의됩니다.
DdQueryDisplaySettingsUniqueness 데스크톱 스위치, 빠른 사용자 스위치 또는 전체 화면 Microsoft MS-DOS 상자가 있는 경우와 같이 모드 스위치가 발생할 때마다 증가되는 정수의 현재 값을 반환합니다.
DdReenableDirectDrawObject NtGdiDdReenableDirectDrawObject 함수에 대한 래퍼입니다.
DdReleaseDC NtGdiDdReleaseDC 함수에 대한 래퍼이며 이전에 DdGetDC 또는 GdiEntry7을 통해 가져온 디바이스 컨텍스트(DC)를 해제합니다. GdiEntry8은 이 함수의 별칭으로 정의됩니다.
DdResetVisrgn NtGdiDdResetVisrgn 함수에 대한 래퍼이며 데스크톱의 창에 대한 클리핑 영역에서 적시에 사용자 모드 정보를 사용하도록 설정합니다.
DdSetGammaRamp DdSetGammaRamp 함수는 디바이스의 감마 램프를 설정합니다.
DdSwapTextureHandles Microsoft DirectDraw 7.0 이전의 DDI(디바이스 드라이버 인터페이스)용으로 개발되었으며 Microsoft Windows NT 시스템에서는 아무 작업도 수행하지 않습니다. 모든 매개 변수는 무시됩니다. GdiEntry16은 이 함수의 별칭으로 정의됩니다.
DdUnattachSurface DdUnattachSurface 함수는 두 커널 모드 표면 개체 간에 DdAttachSurface로 만든 첨부 파일을 제거합니다.
DisableProcessWindowsGhosting 호출 GUI 프로세스에 대해 창 고스팅 기능을 사용하지 않도록 설정합니다. 창 고스팅은 사용자가 응답하지 않는 애플리케이션의 주 창을 최소화, 이동 또는 닫을 수 있는 Windows 관리자 기능입니다.
DnsHostnameToComputerNameA DNS 스타일 호스트 이름을 NetBIOS 스타일 컴퓨터 이름으로 변환합니다. (ANSI)

DnsHostnameToComputerNameW DNS 스타일 호스트 이름을 NetBIOS 스타일 컴퓨터 이름으로 변환합니다. (유니코드)
DosDateTimeToFileTime MS-DOS 날짜 및 시간 값을 파일 시간으로 변환합니다.
FCACHE_CREATE_CALLBACK 캐시에 항목을 만드는 데 사용되는 콜백 함수입니다. (FCACHE_CREATE_CALLBACK)
FCACHE_RICHCREATE_CALLBACK 캐시에 항목을 만드는 데 사용되는 콜백 함수입니다. (FCACHE_RICHCREATE_CALLBACK)
FCIAddFile FCIAddFile은 생성 중인 캐비닛에 파일을 추가합니다.
FCICreate FCICreate 함수는 FCI 컨텍스트를 만듭니다.
FCIDestroy FCIDestroy 함수는 열려 있는 FCI 컨텍스트를 삭제하여 컨텍스트와 연결된 모든 메모리 및 임시 파일을 해제합니다.
FCIFlushCabinet FCIFlushCabinet 함수는 현재 캐비닛을 완료합니다.
FCIFlushFolder FCIFlushFolder 함수는 생성 중인 현재 폴더를 즉시 완료하도록 합니다.
FDICopy FDICopy 함수는 캐비닛에서 파일을 추출합니다.
FDICreate FDICreate 함수는 FDI 컨텍스트를 만듭니다.
FDIDestroy FDIDestroy 함수는 열려 있는 FDI 컨텍스트를 삭제합니다.

FDIIsCabinet FDIIsCabinet 함수는 파일이 캐비닛인지 여부를 확인하고, 파일인 경우 해당 파일에 대한 정보를 반환합니다.
FDITruncateCabinet FDITruncateCabinet 함수는 지정된 폴더 번호에서 시작하는 캐비닛 파일을 자릅니다.
FhServiceBlockBackup 이 함수는 현재 사용자에 대한 백업을 일시적으로 차단합니다.
FhServiceClosePipe FhServiceOpenPipe로 열린 파일 기록 서비스에 대한 통신 채널을 닫습니다.
FhServiceOpenPipe 파일 기록 서비스에 대한 통신 채널을 엽니다.
FhServiceReloadConfiguration 이 함수를 사용하면 파일 기록 서비스가 현재 사용자의 파일 기록 구성 파일을 다시 로드합니다.
FhServiceStartBackup 이 함수는 현재 사용자에 대한 즉시 백업을 시작합니다.
FhServiceStopBackup 이 함수는 현재 사용자에 대한 지속적인 백업 주기를 중지합니다.
FhServiceUnblockBackup 이 함수는 FhServiceBlockBackup을 통해 차단된 백업의 차단을 해제합니다.
FileTimeToDosDateTime 파일 시간을 MS-DOS 날짜 및 시간 값으로 변환합니다.
FindOrCreateNameCache 이름 캐시를 찾거나 만듭니다.
FindSyncContextFromName 지정된 사용자 이름과 연결된 FIO_CONTEXT 구조를 검색합니다.

FNALLOC FNALLOC는 FDI 컨텍스트에서 메모리를 할당하기 위해 애플리케이션 정의 콜백 함수에 대한 선언을 제공합니다.
FNCLOSE FNCLOSE 매크로는 FDI 컨텍스트에서 파일을 닫기 위해 애플리케이션 정의 콜백 함수에 대한 선언을 제공합니다.
FNFCIALLOC FNFCIALLOC는 애플리케이션 정의 콜백 함수에 대한 선언을 제공하여 FCI 컨텍스트 내에서 메모리를 할당합니다.
FNFCICLOSE FNFCICLOSE 매크로는 FCI 컨텍스트에서 파일을 닫기 위해 애플리케이션 정의 콜백 함수에 대한 선언을 제공합니다.
FNFCIDELETE FNFCIDELETE 매크로는 FCI 컨텍스트에서 파일을 삭제하는 애플리케이션 정의 콜백 함수에 대한 선언을 제공합니다.
FNFCIFILEPLACED FNFCIFILEPLACED 매크로는 파일이 캐비닛에 배치될 때 알릴 애플리케이션 정의 콜백 함수에 대한 선언을 제공합니다.
FNFCIFREE FNFCIFREE 매크로는 애플리케이션 정의 콜백 함수에 대한 선언을 제공하여 FCI 컨텍스트에서 이전에 할당된 메모리를 해제합니다.
FNFCIGETNEXTCABINET FNFCIGETNEXTCABINET 매크로는 애플리케이션 정의 콜백 함수에 대한 선언을 제공하여 다음 캐비닛에 대한 정보를 요청합니다.
FNFCIGETOPENINFO FNFCIGETOPENINFO 매크로는 애플리케이션 정의 콜백 함수에 대한 선언을 제공하여 파일을 열고 파일 날짜, 시간 및 특성을 검색합니다.
FNFCIGETTEMPFILE FNFCIGETTEMPFILE 매크로는 임시 파일 이름을 얻기 위해 애플리케이션 정의 콜백 함수에 대한 선언을 제공합니다.

FNFCIOPEN FNFCIOPEN 매크로는 애플리케이션 정의 콜백 함수에 대한 선언을 제공하여 FCI 컨텍스트에서 파일을 엽니다.
FNFCIREAD FNFCIREAD 매크로는 애플리케이션 정의 콜백 함수가 FCI 컨텍스트의 파일에서 데이터를 읽는 선언을 제공합니다.
FNFCISEEK FNFCISEEK 매크로는 파일 포인터를 FCI 컨텍스트의 지정된 위치로 이동하는 애플리케이션 정의 콜백 함수에 대한 선언을 제공합니다.
FNFCISTATUS FNFCISTATUS 매크로는 애플리케이션 정의 콜백 함수에 대한 선언을 제공하여 사용자를 업데이트합니다.
FNFCIWRITE FNFCIWRITE 매크로는 애플리케이션 정의 콜백 함수가 FCI 컨텍스트의 파일에 데이터를 쓰는 선언을 제공합니다.
FNFDINOTIFY FNFDINOTIFY 매크로는 애플리케이션 정의 콜백 알림 함수에 대한 선언을 제공하여 디코더의 상태에 따라 애플리케이션을 업데이트합니다.
FNFREE FNFREE 매크로는 FDI 컨텍스트에서 이전에 할당된 메모리를 해제하기 위해 애플리케이션 정의 콜백 함수에 대한 선언을 제공합니다.
FNOPEN FNOPEN 매크로는 FDI 컨텍스트에서 파일을 열기 위해 애플리케이션 정의 콜백 함수에 대한 선언을 제공합니다.
FNREAD FNREAD 매크로는 애플리케이션 정의 콜백 함수가 FDI 컨텍스트의 파일에서 데이터를 읽는 선언을 제공합니다.
FNSEEK FNSEEK 매크로는 FDI 컨텍스트에서 지정된 위치로 파일 포인터를 이동하는 애플리케이션 정의 콜백 함수에 대한 선언을 제공합니다.

FNWRITE FNWRITE 매크로는 애플리케이션 정의 콜백 함수가 FDI 컨텍스트의 파일에 데이터를 쓰는 선언을 제공합니다.
get_Count 컴퓨터에서 지정된 유형의 모든 보안 제품 공급자의 총 수를 수집합니다.
get_Item 컴퓨터의 공급자 유형 중 하나를 반환합니다.
get_length 컬렉션의 요소 수를 검색합니다. (IXMLElementCollection.get_length)
get_ProductName 보안 제품의 현재 제품 정보를 반환합니다.
get_ProductState 보안 제품에 대한 서명 데이터의 현재 상태를 반환합니다.
get_ProductStateTimestamp 보안 제품의 현재 타임스탬프를 반환합니다.
get_RemediationPath 보안 제품의 현재 수정 경로를 반환합니다.
get_SignatureStatus 보안 제품에 대한 서명 데이터의 현재 상태를 반환합니다.
get_TheaterMode 개체가 극장 모드인지 여부를 설정하거나 가져옵니다. (Get)
GetActiveItem 활성 캡처된 항목을 가져옵니다.
GetBackupStatus FhConfigMgr 개체의 백업 상태 값을 검색합니다.

GetComputerNameA 로컬 컴퓨터의 NetBIOS 이름을 검색합니다. 이 이름은 시스템이 레지스트리에서 읽을 때 시스템 시작 시 설정됩니다. (ANSI)
GetComputerNameW 로컬 컴퓨터의 NetBIOS 이름을 검색합니다. 이 이름은 시스템이 레지스트리에서 읽을 때 시스템 시작 시 설정됩니다. (유니코드)
GetConfigurationDetails 이 메서드는 IFhReassociation::ScanTargetForConfigurations 메서드를 통해 스토리지 디바이스 또는 네트워크 공유에서 검색된 파일 기록 구성을 열거하고 검색된 각 구성에 대한 추가 정보를 반환합니다.
GetCurrentHwProfileA 로컬 컴퓨터의 현재 하드웨어 프로필에 대한 정보를 검색합니다. (ANSI)
GetCurrentHwProfileW 로컬 컴퓨터의 현재 하드웨어 프로필에 대한 정보를 검색합니다. (유니코드)
GetCurrentViewType 현재 뷰의 형식을 가져옵니다.
GetDefaultTarget 현재 할당된 백업 대상에 대한 정보를 쿼리하는 데 사용할 수 있는 IFhTarget 인터페이스에 대한 포인터를 반환합니다.
GetDotStuffState 점 스터핑 메커니즘이 켜져 있을 때 점이 파일에 추가되는지 여부를 결정합니다.
GetFeatureEnabledState 이 함수는 인프라 전용입니다. (GetFeatureEnabledState)
GetFeatureVariant 이 함수는 인프라 전용입니다. (GetFeatureVariant)
GetFileSizeFromContext 핸들을 사용하여 캐시된 파일 크기를 보고합니다.

GetFirmwareEnvironmentVariableA 지정된 펌웨어 환경 변수의 값을 검색합니다. (ANSI)
GetFirmwareEnvironmentVariableExA 지정된 펌웨어 환경 변수 및 해당 특성의 값을 검색합니다. (ANSI)
GetFirmwareEnvironmentVariableExW 지정된 펌웨어 환경 변수 및 해당 특성의 값을 검색합니다. (유니코드)
GetFirmwareEnvironmentVariableW 지정된 펌웨어 환경 변수의 값을 검색합니다. (유니코드)
GetFirmwareType 로컬 컴퓨터의 펌웨어 유형을 검색합니다.
GetGenuineLocalStatus 현재 설치된 운영 체제가 활성화되었는지 여부를 검색합니다.
GetIncludeExcludeRules 현재 FhConfigMgr 개체에 저장된 포함 및 제외 규칙을 검색합니다.
GetItem 포함 또는 제외 목록에서 현재 항목을 검색합니다.
GetLocalPolicy 파일 기록 기능에 대한 로컬 정책에 대한 숫자 매개 변수를 검색합니다.
GetMediaComponentPackageInfo 지정된 요구 사항을 충족하는 시스템에 설치된 모든 미디어 코덱의 속성 목록을 반환합니다.
GetMediaExtensionCommunicationFactory 미디어 확장을 등록하기 위한 통신 팩터리를 만듭니다.
GetNumericalProperty IFhTarget 인터페이스로 표시되는 파일 기록 백업 대상의 숫자 속성을 검색합니다.

GetOsProductContentId 운영 체제의 현재 설치에 해당하는 콘텐츠 식별자를 검색합니다. 콘텐츠 식별자는 스토어 카탈로그에서 운영 체제 제품을 조회하는 데 사용됩니다.
GetPrivateProfileInt GetPrivateProfileInt 함수(winbase.h)는 초기화 파일의 지정된 섹션에서 키와 연결된 정수를 검색합니다.
GetPrivateProfileIntA 초기화 파일의 지정된 섹션에서 키와 연결된 정수 검색 (GetPrivateProfileIntA)
GetPrivateProfileIntW GetPrivateProfileIntW(유니코드) 함수(winbase.h)는 초기화 파일의 지정된 섹션에서 키와 연결된 정수를 검색합니다.
GetPrivateProfileSection GetPrivateProfileSection 함수(winbase.h)는 초기화 파일의 지정된 섹션에 대한 모든 키와 값을 검색합니다.
GetPrivateProfileSectionA 초기화 파일의 지정된 섹션에 대한 모든 키와 값을 검색합니다. (GetPrivateProfileSectionA)
GetPrivateProfileSectionNames GetPrivateProfileSectionNames 함수(winbase.h)는 초기화 파일에서 모든 섹션의 이름을 검색합니다.
GetPrivateProfileSectionNamesA 초기화 파일에서 모든 섹션의 이름을 검색합니다. (GetPrivateProfileSectionNamesA)
GetPrivateProfileSectionNamesW GetPrivateProfileSectionNamesW(유니코드) 함수(winbase.h)는 초기화 파일에서 모든 섹션의 이름을 검색합니다.
GetPrivateProfileSectionW GetPrivateProfileSectionW(유니코드) 함수(winbase.h)는 초기화 파일의 지정된 섹션에 대한 모든 키와 값을 검색합니다.

GetPrivateProfileString GetPrivateProfileString 함수(winbase.h)는 초기화 파일의 지정된 섹션에서 문자열을 검색합니다.
GetPrivateProfileStringA 초기화 파일의 지정된 섹션에서 문자열을 검색합니다. (GetPrivateProfileStringA)
GetPrivateProfileStringW GetPrivateProfileStringW(유니코드) 함수(winbase.h)는 초기화 파일의 지정된 섹션에서 문자열을 검색합니다.
GetPrivateProfileStruct GetPrivateProfileStruct 함수(winbase.h)는 초기화 파일의 지정된 섹션에서 키와 연결된 데이터를 검색합니다.
GetPrivateProfileStructA 초기화 파일의 지정된 섹션에서 키와 연결된 데이터를 검색합니다. (GetPrivateProfileStructA)
GetPrivateProfileStructW GetPrivateProfileStructW(유니코드) 함수(winbase.h)는 초기화 파일의 지정된 섹션에서 키와 연결된 데이터를 검색합니다.
GetProfileIntA Win.ini 파일의 지정된 섹션에 있는 키에서 정수 검색 (ANSI)
GetProfileIntW Win.ini 파일의 지정된 섹션에 있는 키에서 정수 검색 (유니코드)
GetProfileSectionA Win.ini 파일의 지정된 섹션에 대한 모든 키와 값을 검색합니다. (ANSI)
GetProfileSectionW Win.ini 파일의 지정된 섹션에 대한 모든 키와 값을 검색합니다. (유니코드)
GetProfileStringA Win.ini 파일의 지정된 섹션에서 키와 연결된 문자열을 검색합니다. (ANSI)

GetProfileStringW Win.ini 파일의 지정된 섹션에서 키와 연결된 문자열을 검색합니다. (유니코드)
GetSelectedItems 선택한 항목을 가져옵니다.
GetServerForPMP RegisterServerForPMP에 대한 이전 호출을 사용하여 PMP(보호된 미디어 프로세스) 사용에 등록된 COM 서버를 가져옵니다.
GetStringProperty IFhTarget 인터페이스로 표시되는 파일 기록 백업 대상의 문자열 속성을 검색합니다.
GetSystemRegistryQuota 레지스트리의 현재 크기와 레지스트리가 시스템에서 얻을 수 있는 최대 크기를 검색합니다.
GetUserNameA 현재 스레드와 연결된 사용자의 이름을 검색합니다. (ANSI)
GetUserNameW 현재 스레드와 연결된 사용자의 이름을 검색합니다. (유니코드)
I_NetLogonControl2 Netlogon 서비스의 다양한 측면을 제어합니다.
초기화 컴퓨터에서 지정된 형식의 모든 공급자에 대한 정보를 수집합니다.
InitializeCache 파일 핸들 캐시의 초기화 값을 제공합니다.
InsertFile 캐시에 파일을 삽입합니다.
InstallPerfDllA 입력 .ini 파일에 정의된 성능 카운터 문자열을 시스템 레지스트리에 설치합니다. (ANSI)

InstallPerfDllW 입력 .ini 파일에 정의된 성능 카운터 문자열을 시스템 레지스트리에 설치합니다. (유니코드)
InstantiateComponentFromPackage 애플리케이션 패키지에 클래스의 인스턴스를 만듭니다.
Int32x32To64 부 서명된 32비트 정수 2개를 곱하여 서명된 64비트 정수 결과를 반환합니다.
Int64ShllMod32 부호 없는 64비트 정수 값에 대해 왼쪽 논리 시프트 작업을 수행합니다. 이 함수는 시프트 수가 0-31 범위인 왼쪽 논리 교대조에 대해 향상된 시프트 코드를 제공합니다.
Int64ShraMod32 서명된 64비트 정수 값에 대해 올바른 산술 시프트 작업을 수행합니다. 이 함수는 시프트 수가 0-31 범위인 오른쪽 산술 교대조에 대해 향상된 시프트 코드를 제공합니다.
Int64ShrlMod32 부호 없는 64비트 정수 값에 대해 올바른 논리 시프트 작업을 수행합니다. 이 함수는 시프트 수가 0-31 범위인 오른쪽 논리 교대조에 대해 향상된 시프트 코드를 제공합니다.
InvalidateName 사용자가 이름 캐시에서 단일 이름 및 모든 연결된 데이터를 제거할 수 있도록 합니다.
IsApiSetImplemented IsApiSetImplemented 함수는 지정된 API 집합이 컴퓨터에 있는지 테스트합니다.
IsMediaBehaviorEnabled 지정된 GUID와 연결된 미디어 동작을 사용할 수 있는지 여부를 나타내는 값을 가져옵니다.
IsNativeVhdBoot OS가 VHD 컨테이너에서 부팅되었는지를 나타냅니다.
item 인덱스, 이름 또는 둘 다를 사용하여 컬렉션에서 자식 요소를 검색합니다.

LoadConfiguration 현재 사용자의 파일 기록 구성 정보를 FhConfigMgr 개체로 로드합니다.
MoveToNextItem 포함 또는 제외 목록의 다음 항목으로 이동합니다.
MulDiv 두 개의 32비트 값을 곱한 다음 64비트 결과를 세 번째 32비트 값으로 나눕니다.
Multiply128 64비트 정수 2개를 곱하여 128비트 정수 생성
MultiplyExtract128 64비트 정수 2개를 곱하여 128비트 정수를 생성하고, 제품을 오른쪽으로 지정된 비트 수만큼 이동하고, 결과의 낮은 64비트를 반환합니다.
MultiplyHigh 64비트 정수 2개를 곱하여 128비트 정수와 높은 64비트를 가져옵니다.
NdrComplexArrayBufferSize NdrComplexArrayBufferSize 함수는 복잡한 배열을 마샬링하는 데 필요한 버퍼 크기를 바이트 단위로 계산합니다.
NdrComplexArrayMarshall NdrComplexArrayMarshall 함수는 복잡한 배열을 네트워크 버퍼로 마샬링합니다.
NdrComplexArrayUnmarshall NdrComplexArrayUnmarshall 함수는 복잡한 배열을 네트워크 버퍼에서 메모리로 분리합니다.
NdrComplexStructBufferSize NdrComplexStructBufferSize 함수는 복잡한 구조를 마샬링하는 데 필요한 버퍼 크기를 바이트 단위로 계산합니다.
NdrComplexStructMarshall NdrComplexStructMarshall 함수는 복잡한 구조를 네트워크 버퍼로 마샬링합니다.

NdrComplexStructUnmarshall NdrComplexStructUnmarshall 함수는 복잡한 구조를 네트워크 버퍼에서 메모리로 분리합니다.
NdrConformantArrayBufferSize NdrConformantArrayBufferSize 함수는 규칙적인 배열을 마샬링하는 데 필요한 버퍼 크기를 바이트 단위로 계산합니다.
NdrConformantArrayMarshall NdrConformantArrayMarshall 함수는 규칙적인 배열을 네트워크 버퍼로 마샬링합니다.
NdrSimpleStructBufferSize NdrSimpleStructBufferSize 함수는 간단한 구조를 마샬링하는 데 필요한 버퍼 크기를 바이트 단위로 계산합니다.
NdrSimpleStructMarshall NdrSimpleStructMarshall 함수는 단순 구조를 네트워크 버퍼로 마샬링합니다.
NdrSimpleStructUnmarshall NdrSimpleStructUnmarshall 함수는 단순한 구조를 네트워크 버퍼에서 메모리로 분리합니다.
NdrUserMarshalUnmarshall NdrUserMarshalUnmarshall 함수는 사용자 정의 unmarshal 루틴을 호출하여 특성으로 데이터를 unmarshal합니다.
NtClose 더 이상 사용되지 않습니다. 지정된 핸들을 닫습니다. NtClose는 CloseHandle로 대체됩니다.
NtCreateFile 새 파일 또는 디렉터리를 만들거나 기존 파일, 디바이스, 디렉터리 또는 볼륨을 엽니다.
NtDeviceIoControlFile 더 이상 사용되지 않습니다. 제공된 버퍼에 대한 설명자를 빌드하고 형식화되지 않은 데이터를 파일 핸들과 연결된 디바이스 드라이버에 전달합니다. NtDeviceIoControlFile은 DeviceIoControl로 대체됩니다.
NtNotifyChangeMultipleKeys 레지스트리 키 또는 해당 하위 키가 변경되면 알림을 요청합니다.

NtOpenFile 기존 파일, 디바이스, 디렉터리 또는 볼륨을 열고 파일 개체에 대한 핸들을 반환합니다.
NtQueryMultipleValueKey 지정된 다중 값 키에 대한 값을 검색합니다.
NtQueryObject 다양한 종류의 개체 정보를 검색합니다.
NtQuerySystemInformation 지정된 시스템 정보를 검색합니다.
NtQuerySystemTime 현재 시스템 시간을 검색합니다.
NtRenameKey 지정된 레지스트리 키의 이름을 변경합니다. (NtRenameKey)
NtSetInformationKey 지정된 레지스트리 키에 대한 정보를 설정합니다.
NtWaitForSingleObject 더 이상 사용되지 않습니다. 지정된 개체가 신호 상태가 될 때까지 기다립니다. NtWaitForSingleObject는 WaitForSingleObject로 대체됩니다.
OnClosed 카메라 UI 컨트롤이 닫혀 있을 때 발생합니다.
OnItemCaptured 항목이 캡처될 때 발생합니다.
OnItemDeleted 항목이 삭제될 때 발생합니다.
OnStartupComplete 카메라 UI 컨트롤의 시작이 완료되면 발생합니다.

OnSuspendComplete 카메라 UI 컨트롤이 일시 중단될 때 발생합니다.
PerformReassociation 이 메서드는 현재 사용자와 IFhReassociation::SelectConfiguration 메서드를 통해 이전에 선택한 구성 간의 관계를 다시 설정하고 현재 컴퓨터에서 백업 데이터를 수락하기 위한 대상 디바이스를 준비합니다.
PopulationCount64 부호 없는 64비트 정수의 1비트(모집단 수)를 계산합니다.
PROCESSENUMPROC VDMEnumProcessWOW가 열거하는 각 VDM(가상 DOS 머신)에 대한 정보를 수신하도록 이 함수를 구현합니다.
ProduceDotStuffedContext 요청된 상태의 FIO_CONTEXT 구조를 검색합니다.
ProvisionAndSetNewTarget 특정 스토리지 디바이스 또는 네트워크 공유를 파일 기록 백업 대상으로 프로비전하고 현재 사용자의 기본 백업 대상으로 할당합니다.
PSYMBOLSERVERCALLBACKPROC 기호 서버 DLL에 대한 진입점입니다.
put_TheaterMode 개체가 극장 모드인지 여부를 설정하거나 가져옵니다. (Put)
QueryProtectionStatus 현재 파일 기록 보호 상태를 검색합니다.
RecordFeatureError 이 함수는 인프라 전용입니다. (RecordFeatureError)
RecordFeatureUsage 이 함수는 인프라 전용입니다. (RecordFeatureUsage)

RegCloseKey 지정된 레지스트리 키에 대한 핸들을 닫습니다.
RegConnectRegistryA 다른 컴퓨터에서 미리 정의된 레지스트리 키에 대한 연결을 설정합니다. (ANSI)
RegConnectRegistryW 다른 컴퓨터에서 미리 정의된 레지스트리 키에 대한 연결을 설정합니다. (유니코드)
RegCopyTreeA 지정된 레지스트리 키를 해당 값 및 하위 키와 함께 지정된 대상 키에 복사합니다. (ANSI)
RegCopyTreeW 지정된 레지스트리 키를 해당 값 및 하위 키와 함께 지정된 대상 키에 복사합니다. (유니코드)
RegCreateKeyA 지정된 레지스트리 키를 만듭니다. 레지스트리에 키가 이미 있는 경우 함수가 키를 엽니다. (ANSI)
RegCreateKeyExA 지정된 레지스트리 키를 만듭니다. 키가 이미 있는 경우 함수가 키를 엽니다. 키 이름은 대/소문자를 구분하지 않습니다. (ANSI)
RegCreateKeyExW 지정된 레지스트리 키를 만듭니다. 키가 이미 있는 경우 함수가 키를 엽니다. 키 이름은 대/소문자를 구분하지 않습니다. (유니코드)
RegCreateKeyTransactedA 지정된 레지스트리 키를 만들고 트랜잭션에 연결합니다. (ANSI)
RegCreateKeyTransactedW 지정된 레지스트리 키를 만들고 트랜잭션에 연결합니다. (유니코드)
RegCreateKeyW 지정된 레지스트리 키를 만듭니다. 레지스트리에 키가 이미 있는 경우 함수가 키를 엽니다. (유니코드)

RegDeleteKeyA 하위 키와 해당 값을 삭제합니다. (ANSI)
RegDeleteKeyExA 레지스트리의 지정된 플랫폼별 보기에서 하위 키와 해당 값을 삭제합니다. (ANSI)
RegDeleteKeyExW 레지스트리의 지정된 플랫폼별 보기에서 하위 키와 해당 값을 삭제합니다. (유니코드)
RegDeleteKeyTransactedA 레지스트리의 지정된 플랫폼별 보기에서 하위 키와 해당 값을 트랜잭션 작업으로 삭제합니다. (ANSI)
RegDeleteKeyTransactedW 레지스트리의 지정된 플랫폼별 보기에서 하위 키와 해당 값을 트랜잭션 작업으로 삭제합니다. (유니코드)
RegDeleteKeyValueA 지정된 레지스트리 키 및 하위 키에서 지정된 값을 제거합니다. (ANSI)
RegDeleteKeyValueW 지정된 레지스트리 키 및 하위 키에서 지정된 값을 제거합니다. (유니코드)
RegDeleteKeyW 하위 키와 해당 값을 삭제합니다. (유니코드)
RegDeleteTreeA 지정된 키의 하위 키와 값을 재귀적으로 삭제합니다. (ANSI)
RegDeleteTreeW 지정된 키의 하위 키와 값을 재귀적으로 삭제합니다. (유니코드)
RegDeleteValueA 지정된 레지스트리 키에서 명명된 값을 제거합니다. (ANSI)
RegDeleteValueW 지정된 레지스트리 키에서 명명된 값을 제거합니다. (유니코드)

RegDisablePredefinedCache 현재 프로세스에 대한 HKEY_CURRENT_USER 대해 미리 정의된 레지스트리 핸들의 캐싱 처리를 사용하지 않도록 설정합니다.
RegDisablePredefinedCacheEx 현재 프로세스에 대해 미리 정의된 모든 레지스트리 핸들에 대한 핸들 캐싱을 사용하지 않도록 설정합니다.
RegDisableReflectionKey 지정된 키에 대한 레지스트리 리플렉션을 사용하지 않도록 설정합니다. 키에 대한 리플렉션을 사용하지 않도록 설정해도 하위 키의 리플렉션에는 영향을 주지 않습니다.
RegEnableReflectionKey 지정된 비활성화된 키에 대한 레지스트리 리플렉션을 복원합니다. 키에 대한 리플렉션을 복원해도 하위 키의 리플렉션에는 영향을 주지 않습니다.
RegEnumKeyA 지정된 열린 레지스트리 키의 하위 키를 열거합니다. (RegEnumKeyA)
RegEnumKeyExA 지정된 열린 레지스트리 키의 하위 키를 열거합니다. 함수는 호출할 때마다 하나의 하위 키에 대한 정보를 검색합니다. (ANSI)
RegEnumKeyExW 지정된 열린 레지스트리 키의 하위 키를 열거합니다. 함수는 호출할 때마다 하나의 하위 키에 대한 정보를 검색합니다. (유니코드)
RegEnumKeyW 지정된 열린 레지스트리 키의 하위 키를 열거합니다. (RegEnumKeyW)
RegEnumValueA 지정된 열린 레지스트리 키의 값을 열거합니다. 함수는 호출될 때마다 키의 인덱싱된 값 이름 및 데이터 블록 하나를 복사합니다. (ANSI)
RegEnumValueW 지정된 열린 레지스트리 키의 값을 열거합니다. 함수는 호출될 때마다 키의 인덱싱된 값 이름 및 데이터 블록 하나를 복사합니다. (유니코드)

RegFlushKey 지정된 열린 레지스트리 키의 특성을 모두 레지스트리에 기록합니다.
RegGetValueA 지정된 레지스트리 값에 대한 형식 및 데이터를 검색합니다. (ANSI)
RegGetValueW 지정된 레지스트리 값에 대한 형식 및 데이터를 검색합니다. (유니코드)
RegInstallA 제공된 테이블의 문자열 레지스트리 값을 업데이트. (ANSI)
RegInstallW 제공된 테이블의 문자열 레지스트리 값을 업데이트. (유니코드)
RegisterServerForPMP COM 서버 CLSID 및 PMP(보호된 미디어 프로세스) 사용을 위한 클래스 팩터리를 등록합니다.
RegLoadAppKeyA 지정된 레지스트리 하이브를 애플리케이션 하이브로 로드합니다. (ANSI)
RegLoadAppKeyW 지정된 레지스트리 하이브를 애플리케이션 하이브로 로드합니다. (유니코드)
RegLoadKeyA HKEY_USERS 또는 HKEY_LOCAL_MACHINE 아래에 하위 키를 만들고 지정된 레지스트리 하이브의 데이터를 해당 하위 키로 로드합니다. (ANSI)
RegLoadKeyW HKEY_USERS 또는 HKEY_LOCAL_MACHINE 아래에 하위 키를 만들고 지정된 레지스트리 하이브의 데이터를 해당 하위 키로 로드합니다. (유니코드)
RegLoadMUIStringA 지정된 키와 하위 키에서 지정된 문자열을 로드합니다. (ANSI)
RegLoadMUIStringW 지정된 키와 하위 키에서 지정된 문자열을 로드합니다. (유니코드)

RegNotifyChangeKeyValue 지정된 레지스트리 키의 특성 또는 내용에 대한 변경 내용에 대해 호출자에게 알합니다.
RegOpenCurrentUser 현재 스레드가 가장하는 사용자의 HKEY_CURRENT_USER 키에 대한 핸들을 검색합니다.
RegOpenKeyA 지정된 레지스트리 키를 엽니다. (ANSI)
RegOpenKeyExA 지정된 레지스트리 키를 엽니다. 키 이름은 대/소문자를 구분하지 않습니다. (ANSI)
RegOpenKeyExW 지정된 레지스트리 키를 엽니다. 키 이름은 대/소문자를 구분하지 않습니다. (유니코드)
RegOpenKeyTransactedA 지정된 레지스트리 키를 열고 트랜잭션에 연결합니다. (ANSI)
RegOpenKeyTransactedW 지정된 레지스트리 키를 열고 트랜잭션에 연결합니다. (유니코드)
RegOpenKeyW 지정된 레지스트리 키를 엽니다. (유니코드)
RegOpenUserClassesRoot 지정된 사용자의 HKEY_CLASSES_ROOT 키에 대한 핸들을 검색합니다. 사용자는 액세스 토큰으로 식별됩니다.
RegOverridePredefKey 미리 정의된 레지스트리 키를 지정된 레지스트리 키에 매핑합니다.
RegQueryInfoKeyA 지정된 레지스트리 키에 대한 정보를 검색합니다. (ANSI)
RegQueryInfoKeyW 지정된 레지스트리 키에 대한 정보를 검색합니다. (유니코드)

RegQueryMultipleValuesA 열린 레지스트리 키와 연결된 값 이름 목록에 대한 형식 및 데이터를 검색합니다. (ANSI)
RegQueryMultipleValuesW 열린 레지스트리 키와 연결된 값 이름 목록에 대한 형식 및 데이터를 검색합니다. (유니코드)
RegQueryReflectionKey 지정된 키에 대해 리플렉션을 사용하지 않도록 설정했는지 여부를 결정합니다.
RegQueryValueA 지정된 레지스트리 키의 기본값 또는 명명되지 않은 값과 연결된 데이터를 검색합니다. 데이터는 null로 끝나는 문자열이어야 합니다. (ANSI)
RegQueryValueExA 열린 레지스트리 키와 연결된 지정된 값 이름에 대한 형식 및 데이터를 검색합니다. (ANSI)
RegQueryValueExW 열린 레지스트리 키와 연결된 지정된 값 이름에 대한 형식 및 데이터를 검색합니다. (유니코드)
RegQueryValueW 지정된 레지스트리 키의 기본값 또는 명명되지 않은 값과 연결된 데이터를 검색합니다. 데이터는 null로 끝나는 문자열이어야 합니다. (유니코드)
RegRenameKey 지정된 레지스트리 키의 이름을 변경합니다. (RegRenameKey)
RegReplaceKeyA 레지스트리 키와 모든 하위 키를 지원하는 파일을 다른 파일로 바꿉니다. 따라서 시스템이 다음에 시작될 때 키와 하위 키는 새 파일에 저장된 값을 갖게 됩니다. (ANSI)
RegReplaceKeyW 레지스트리 키와 모든 하위 키를 지원하는 파일을 다른 파일로 바꿉니다. 따라서 시스템이 다음에 시작될 때 키와 하위 키는 새 파일에 저장된 값을 갖게 됩니다. (유니코드)
RegRestoreKeyA 지정된 파일에서 레지스트리 정보를 읽고 지정된 키 위에 복사합니다. 이 레지스트리 정보는 키와 여러 수준의 하위 키 형식일 수 있습니다. (ANSI)

RegRestoreKeyW 지정된 파일에서 레지스트리 정보를 읽고 지정된 키 위에 복사합니다. 이 레지스트리 정보는 키와 여러 수준의 하위 키 형식일 수 있습니다. (유니코드)
RegSaveKeyA 지정된 키와 모든 하위 키와 값을 표준 형식으로 새 파일에 저장합니다. (ANSI)
RegSaveKeyExA 지정된 키와 모든 하위 키 및 값을 레지스트리 파일에 지정된 형식으로 저장합니다. (ANSI)
RegSaveKeyExW 지정된 키와 모든 하위 키 및 값을 레지스트리 파일에 지정된 형식으로 저장합니다. (유니코드)
RegSaveKeyW 지정된 키와 모든 하위 키와 값을 표준 형식으로 새 파일에 저장합니다. (유니코드)
RegSetKeyValueA 지정된 레지스트리 키 및 하위 키에서 지정된 값의 데이터를 설정합니다. (ANSI)
RegSetKeyValueW 지정된 레지스트리 키 및 하위 키에서 지정된 값의 데이터를 설정합니다. (유니코드)
RegSetValueA 지정된 레지스트리 키의 기본값 또는 명명되지 않은 값에 대한 데이터를 설정합니다. 데이터는 텍스트 문자열이어야 합니다. (ANSI)
RegSetValueExA 레지스트리 키 아래에 지정된 값의 데이터 및 형식을 설정합니다. (ANSI)
RegSetValueExW 레지스트리 키 아래에 지정된 값의 데이터 및 형식을 설정합니다. (유니코드)
RegSetValueW 지정된 레지스트리 키의 기본값 또는 명명되지 않은 값에 대한 데이터를 설정합니다. 데이터는 텍스트 문자열이어야 합니다. (유니코드)

RegUnLoadKeyA 레지스트리에서 지정된 레지스트리 키와 해당 하위 키를 언로드합니다. (ANSI)
RegUnLoadKeyW 레지스트리에서 지정된 레지스트리 키와 해당 하위 키를 언로드합니다. (유니코드)
ReleaseNameCache 이름 캐시를 해제합니다.
RemoveCapturedItem 캡처된 항목을 제거합니다.
RequireNetworkDuringMediaTaskCompletion 미디어 작업 완료에 필요한 네트워크 연결 수를 증가 또는 감소합니다.
재개 사용자 인터페이스 컨트롤의 다시 시작을 시뮬레이션합니다.
RtlAnsiStringToUnicodeString 지정된 ANSI 원본 문자열을 유니코드 문자열로 변환합니다.
RtlCharToInteger 문자열을 정수로 변환합니다.
RtlFreeAnsiString RtlUnicodeStringToAnsiString에서 할당한 문자열 버퍼를 해제합니다.
RtlFreeOemString RtlUnicodeStringToOemString에서 할당한 문자열 버퍼를 해제합니다.
RtlFreeUnicodeString RtlAnsiStringToUnicodeString 또는 RtlUpcaseUnicodeString에 의해 할당된 문자열 버퍼를 해제합니다.
RtlInitString 계산된 문자열을 초기화합니다.

RtlInitUnicodeString 계산된 유니코드 문자열을 초기화합니다.
RtlIsNameLegalDOS8Dot3 지정된 이름을 사용하여 FAT 파일 시스템에 파일을 만들 수 있는지 여부를 결정합니다.
RtlLocalTimeToSystemTime 지정된 현지 시간을 시스템 시간으로 변환합니다.
RtlRaiseException 예외를 발생합니다.
RtlTimeToSecondsSince1970 지정된 64비트 시스템 시간을 1970년 1월 1일 이후의 시간(초)으로 변환합니다.
RtlUnicodeStringToAnsiString 지정된 유니코드 소스 문자열을 ANSI 문자열로 변환합니다.
RtlUnicodeStringToOemString 지정된 유니코드 소스 문자열을 OEM 문자열로 변환합니다. 변환은 OEM 코드 페이지(OCP)와 관련하여 수행됩니다.
RtlUnicodeToMultiByteSize 유니코드 문자열을 ANSI 문자열로 나타내는 데 필요한 바이트 수를 결정합니다.
RtlUniform D.H. Lehmer의 1948 알고리즘을 사용하여 균일한 난수를 생성합니다.
SaveConfiguration 현재 사용자의 파일 기록 구성 파일에 대해 LoadConfiguration, CreateDefaultConfiguration 또는 SaveConfiguration 메서드를 마지막으로 호출한 이후 FhConfigMgr 개체에서 변경된 모든 변경 내용을 디스크에 저장합니다.
ScanTargetForConfigurations 지정된 스토리지 디바이스 또는 네트워크 공유의 네임스페이스를 검사하여 현재 컴퓨터에서 다시 연결하고 계속 사용할 수 있는 파일 기록 구성을 검색합니다.

SelectConfiguration 이후 다시 연결하기 위해 IFhReassociation::ScanTargetForConfigurations 메서드가 스토리지 디바이스 또는 네트워크 공유에서 검색한 파일 기록 구성 중 하나를 선택합니다.
SendIMEMessageExA 지정된 하위 함수를 통해 IME(입력 메서드 편집기)에 대한 작업 또는 처리를 지정합니다. (ANSI)
SendIMEMessageExW 지정된 하위 함수를 통해 IME(입력 메서드 편집기)에 대한 작업 또는 처리를 지정합니다. (유니코드)
SetBackupStatus FhConfigMgr 개체의 백업 상태 값을 변경합니다.
SetDotStuffingOnWrites 메시지의 파일 핸들 캐시의 쓰기 경로에서 점 스터핑 속성을 사용하도록 설정합니다.
SetDotStuffState 점 스터핑을 FIO_CONTEXT 구조에서 설정할 수 있습니다.
SetFirmwareEnvironmentVariableA 지정된 펌웨어 환경 변수의 값을 설정합니다. (ANSI)
SetFirmwareEnvironmentVariableExA 지정된 펌웨어 환경 변수의 값을 이 변수를 저장하고 유지 관리하는 방법을 나타내는 특성으로 설정합니다.
SetFirmwareEnvironmentVariableExW 지정된 펌웨어 환경 변수의 값과 이 변수를 저장하고 유지 관리하는 방법을 나타내는 특성을 설정합니다.
SetFirmwareEnvironmentVariableW 지정된 펌웨어 환경 변수의 값을 설정합니다. (유니코드)
SetLocalPolicy FhConfigMgr 개체에서 로컬 정책의 숫자 매개 변수 값을 변경합니다.

SetShellNext ShellNext 레지스트리 키를 지정된 값으로 설정합니다.
ShiftLeft128 128비트 왼쪽으로 이동합니다.
ShiftRight128 128비트 오른쪽으로 이동합니다.
표시 카메라의 사용자 인터페이스 컨트롤을 표시합니다.
ShowProductKeyUI 사용자가 제품 키를 제공하여 운영 체제의 정품 복사본을 업그레이드하거나 가져올 수 있는 사용자 인터페이스를 표시합니다.
SubscribeFeatureStateChangeNotification 이 함수는 인프라 전용입니다. (SubscribeFeatureStateChangeNotification)
Suspend 사용자 인터페이스 컨트롤의 일시 중단을 시뮬레이션합니다.
태그 shim 데이터베이스의 항목을 식별합니다.
TASKENUMPROCEX VDMEnumTaskWOWEx가 열거하는 각 작업에 대한 정보를 수신하도록 이 함수를 구현합니다.
TCOMPfromLZXWindow TCOMPfromLZXWindow 매크로는 창 크기를 FCIAddFile의 LXZTCOMP 값으로 변환합니다.
TerminateCache 캐시를 제거하고 DLL을 사용하여 비동기 I/O를 종료합니다.
uaw_wcslen null로 종료된 유니코드 문자열의 문자 수를 검색합니다.

UInt32x32To64 부호 없는 32비트 정수 2개를 곱하여 부호 없는 64비트 정수 결과를 반환합니다.
UnregisterServerForPMP 이전에 PMP(보호된 미디어 프로세스) 사용을 위해 등록된 COM 서버 CLSID 및 클래스 팩터리를 등록합니다.
UnsignedMultiply128 부호 없는 64비트 정수 2개를 곱하여 부호 없는 128비트 정수 생성
UnsignedMultiplyExtract128 부호 없는 64비트 정수 2개를 곱하여 부호 없는 128비트 정수를 생성하고, 지정된 비트 수만큼 제품을 오른쪽으로 이동하고, 결과의 낮은 64비트를 반환합니다.
UnsignedMultiplyHigh 2개의 64비트 정수에 곱하여 128비트 정수와 부호 없는 상위 64비트를 가져옵니다.
UnsubscribeFeatureStateChangeNotification 이 함수는 인프라 전용입니다. (UnsubscribeFeatureStateChangeNotification)
UpdateOperatingSystem 설치된 운영 체제 버전을 사용자가 Windows 스토어에서 구입한 버전으로 업그레이드하거나 운영 체제의 정품 복사본을 가져옵니다.
ValidateTarget 특정 스토리지 디바이스 또는 네트워크 공유를 파일 기록 백업 대상으로 사용할 수 있는지 여부를 확인합니다.
ValidateTarget 이 메서드는 특정 스토리지 디바이스 또는 네트워크 공유를 파일 기록 기본 대상으로 사용할 수 있는지 여부와 다시 연결이 가능한지 여부를 확인합니다.
VDMEnumProcessWOW 16비트 Windows 작업을 실행하는 모든 가상 DOS 머신을 열거합니다.
VDMEnumTaskWOWEx 특정 VDM(가상 DOS 머신) 내에서 작업을 열거합니다.

VER_SET_CONDITION 지정된 운영 체제 버전 특성에 사용할 비교 연산자를 나타내도록 64비트 값의 비트를 설정합니다. 이 매크로는 VerifyVersionInfo 함수의 dwlConditionMask 매개 변수를 빌드하는 데 사용됩니다.
VerifierEnumerateResource 디버깅 및 지원 도구에서 사용할 운영 체제 리소스를 열거합니다.
VerifierIsPerUserSettingsEnabled 애플리케이션 검증 도구가 사용자별 설정을 사용할 수 있는지 여부를 결정합니다.
VerifyVersionInfoA 운영 체제 버전 요구 사항 집합을 현재 실행 중인 시스템 버전의 해당 값과 비교합니다. (ANSI)
VerifyVersionInfoW 운영 체제 버전 요구 사항 집합을 현재 실행 중인 시스템 버전의 해당 값과 비교합니다. (유니코드)
VerSetConditionMask 지정된 운영 체제 버전 특성에 사용할 비교 연산자를 나타내도록 64비트 값의 비트를 설정합니다. 이 함수는 VerifyVersionInfo 함수의 dwlConditionMask 매개 변수를 빌드하는 데 사용됩니다.
WINNLSEnableIME IME(입력 방법 편집기)를 일시적으로 사용하거나 사용하지 않도록 설정하고 동시에 IME가 소유한 모든 창의 표시를 켜거나 끕니다.
WritePrivateProfileSectionA 초기화 파일에서 지정된 섹션의 키와 값을 바꿉니다. (ANSI)
WritePrivateProfileSectionW 초기화 파일에서 지정된 섹션의 키와 값을 바꿉니다. (유니코드)
WritePrivateProfileStringA 초기화 파일의 지정된 섹션에 문자열을 복사합니다. (ANSI)
WritePrivateProfileStringW 초기화 파일의 지정된 섹션에 문자열을 복사합니다. (유니코드)

WritePrivateProfileStructA 초기화 파일의 지정된 섹션에 있는 키에 데이터를 복사합니다. 데이터를 복사할 때 함수는 체크섬을 계산하고 데이터 끝에 추가합니다. (ANSI)
WritePrivateProfileStructW 초기화 파일의 지정된 섹션에 있는 키에 데이터를 복사합니다. 데이터를 복사할 때 함수는 체크섬을 계산하고 데이터 끝에 추가합니다. (유니코드)
WriteProfileSectionA Win.ini 파일의 지정된 섹션 내용을 지정된 키와 값으로 바꿉니다. (ANSI)
WriteProfileSectionW Win.ini 파일의 지정된 섹션 내용을 지정된 키와 값으로 바꿉니다. (유니코드)
WriteProfileStringA 문자열을 Win.ini 파일의 지정된 섹션에 복사합니다. (ANSI)
WriteProfileStringW 문자열을 Win.ini 파일의 지정된 섹션에 복사합니다. (유니코드)
WscGetSecurityProviderHealth 지정된 WSC_SECURITY_PROVIDER 열거형 값으로 표시되는 보안 공급자 범주의 집계 상태를 가져옵니다.
WscRegisterForChanges WSC(Windows 보안 Center)가 보안 공급자 중 하나의 상태에 영향을 줄 수 있는 변경 사항을 검색할 때 실행할 콜백 함수를 등록합니다.
WscUnRegisterChanges WscRegisterForChanges 함수를 호출하여 만든 콜백 등록을 취소합니다.

인터페이스

ICameraUIControl 카메라 디바이스에 대한 사용자 인터페이스 컨트롤을 사용하도록 설정합니다.

ICameraUIControlEventCallback 카메라 사용자 인터페이스 컨트롤에서 이벤트를 수신하기 위한 콜백 인터페이스입니다.
IEditionUpgradeHelper Windows 스토어에서 사용자가 구매한 Windows 제품을 설치하거나, 사용자가 현재 설치한 Windows 버전으로 업그레이드를 수행하거나, Windows의 정품이 아닌 복사본을 Windows 정품 복사본으로 바꿀 수 있습니다.
IFhConfigMgr IFhConfigMgr 인터페이스를 사용하면 클라이언트 애플리케이션이 이 인터페이스의 메서드가 호출되는 사용자 계정에 대한 파일 기록 구성을 읽고 수정할 수 있습니다.
IFhReassociation 이 인터페이스를 사용하면 클라이언트 애플리케이션이 파일 기록 대상의 파일 기록 구성을 현재 사용자와 다시 연결할 수 있습니다.
IFhScopeIterator IFhScopeIterator 인터페이스를 사용하면 클라이언트 애플리케이션이 포함 또는 제외 목록의 개별 항목을 열거할 수 있습니다. 포함 및 제외 목록을 검색하려면 IFhConfigMgr::GetIncludeExcludeRules 메서드를 호출합니다.
IFhTarget IFhTarget 인터페이스를 사용하면 클라이언트 애플리케이션이 파일 기록 백업 대상의 숫자 및 문자열 속성을 읽을 수 있습니다.
IIsolatedAppLauncher 앱이 Windows Defender Application Guard 컨테이너(VM 컨테이너 환경)에서 실행 중인지 여부를 확인할 수 있습니다.
IWebBrowser2 WebBrowser 컨트롤(Microsoft ActiveX 컨트롤)에 의해 구현되거나 OLE 자동화(InternetExplorer 애플리케이션)의 인스턴스에 의해 구현되는 메서드를 노출합니다.
IWscProduct 개별 공급자가 Windows 보안 Center와 상호 작용할 수 있도록 제품 정보를 가져오는 방법을 제공합니다.
IWSCProductList 컴퓨터에 설치된 선택한 공급자 유형에 대한 제품 정보를 수집하는 메서드를 제공합니다.

IXMLElementCollection 인덱싱된 액세스에 대한 XML 요소 컬렉션을 지원합니다.

구조체

AVRF_BACKTRACE_INFORMATION 수행된 백트레이스에 대한 정보를 포함합니다.
AVRF_HANDLE_OPERATION 핸들 추적 정보를 수집하는 데 필요한 정보를 포함합니다.
AVRF_HEAP_ALLOCATION 힙 할당에 대한 메타데이터 정보를 저장합니다.
CCAB CCAB 구조에는 캐비닛 정보가 포함됩니다.
Erf ERF 구조에는 FCI/FDI의 오류 정보가 포함되어 있습니다. 호출자는 이 구조를 수정하면 안 됩니다.
FDICABINETINFO FDICABINETINFO 구조체에는 특정 캐비닛 파일에 대한 세부 정보가 포함되어 있습니다.
FDINOTIFICATION FNFDINOTIFY에 정보를 제공하는 FDINOTIFICATION 구조체입니다.
FEATURE_ERROR 이 구조는 인프라 전용입니다.
HW_PROFILE_INFOA 하드웨어 프로필에 대한 정보를 포함합니다. (ANSI)
HW_PROFILE_INFOW 하드웨어 프로필에 대한 정보를 포함합니다. (유니코드)

IFEntry 16~23페이지의 IETF(Internet Engineering Task Force) Request for Comments(RFC) 1213에 지정된 대로 TCP/IP 네트워크(MIB-II)의 네트워크 관리를 위한 관리 정보 베이스용 SNMPv2(Simple Network Management Protocol, 버전 2) 표준에 따라 TCP/IP 인터페이스를 설명합니다.
IMESTRUCT SendIMEMessageEx에서 IME(입력 메서드 편집기) 메시지 및 해당 매개 변수에서 실행할 하위 기능을 지정하는 데 사용됩니다. 이 구조체는 해당 하위 기능에서 반환 값을 받는 데도 사용됩니다.
IPAddrEntry IETF(Internet Engineering Task Force) REQUEST for Comments(RFC) 2011에 지정된 대로 IP(인터넷 프로토콜)에 대한 MIB-II(관리 정보 베이스) 정보 그룹의 일부를 구현합니다. (IPAddrEntry)
IPInterfaceInfo IPInterfaceInfo 구조체에는 IOCTL_TCP_QUERY_INFORMATION_EX 제어 코드에서 반환된 특정 IP 인터페이스에 대한 정보가 포함됩니다.
IPSNMPInfo IETF(Internet Engineering Task Force) REQUEST for Comments(RFC) 2011에 지정된 대로 IP(인터넷 프로토콜)에 대한 MIB-II(관리 정보 베이스) 정보 그룹의 일부를 구현합니다. (IPSNMPInfo)
JAVA_TRUST 신뢰 정보를 포함합니다.
NAME_CACHE_CONTEXT 이름 캐시를 나타냅니다.
NETLOGON_INFO_1 도메인 컨트롤러에서 수준 1 제어 쿼리 응답을 정의합니다.
NETLOGON_INFO_2 도메인 컨트롤러에서 수준 2 제어 쿼리 응답을 정의합니다.
NETLOGON_INFO_3 도메인 컨트롤러에서 수준 3 제어 쿼리 응답을 정의합니다.
NETLOGON_INFO_4 도메인 컨트롤러에서 수준 4 제어 쿼리 응답을 정의합니다.

OSVERSIONINFOA 운영 체제 버전 정보를 포함합니다. (ANSI)
OSVERSIONINFOEXA 운영 체제 버전 정보를 포함합니다. 이 정보에는 주 버전 및 부 버전 번호, 빌드 번호, 플랫폼 식별자, 제품 제품군 및 시스템에 설치된 최신 서비스 팩에 대한 정보가 포함됩니다. (ANSI)
OSVERSIONINFOEXW 운영 체제 버전 정보를 포함합니다. 이 정보에는 주 버전 및 부 버전 번호, 빌드 번호, 플랫폼 식별자, 제품 제품군 및 시스템에 설치된 최신 서비스 팩에 대한 정보가 포함됩니다. (유니코드)
OSVERSIONINFOW 운영 체제 버전 정보를 포함합니다. (유니코드)
STRENTRYA 레지스트리 문자열 대체를 나타냅니다. (ANSI)
STRENTRYW 레지스트리 문자열 대체를 나타냅니다. (유니코드)
STRING RtlUnicodeStringToOemString 함수와 함께 사용됩니다.
STRTABLEA 레지스트리 문자열 대체 테이블을 나타냅니다. (ANSI)
STRTABLEW 레지스트리 문자열 대체 테이블을 나타냅니다. (유니코드)
TCP_REQUEST_QUERY_INFORMATION_EX_W2K TCP_REQUEST_QUERY_INFORMATION_EX_W2K 구조체(tdiinfo.h)는 IOCTL_TCP_QUERY_INFORMATION_EX 제어 코드에 대한 입력을 포함합니다.
TCP_REQUEST_QUERY_INFORMATION_EX_XP TCP_REQUEST_QUERY_INFORMATION_EX_XP 구조체(tdiinfo.h)는 IOCTL_TCP_QUERY_INFORMATION_EX 제어 코드에 대한 입력을 포함합니다.

TDIEntityID IOCTL_TCP_QUERY_INFORMATION_EX 제어 코드를 사용하여 검색된 TDI 드라이버에 대한 정보를 나타내는 TDIObjectID 구조체의 일부를 포함합니다.
TDIObjectID TCP 드라이버에서 요청되는 정보의 종류를 지정하기 위해 IOCTL_TCP_QUERY_INFORMATION_EX 제어 코드와 함께 사용되는 TCP_REQUEST_QUERY_INFORMATION_EX 구조체의 일부를 포함합니다.
VALENTA 레지스트리 값에 대한 정보를 포함합니다. RegQueryMultipleValues 함수는 이 구조를 사용합니다. (ANSI)
VALENTW 레지스트리 값에 대한 정보를 포함합니다. RegQueryMultipleValues 함수는 이 구조를 사용합니다. (유니코드)

디바이스 액세스 API

아티클2023. 08. 22.

디바이스 액세스 API 기술 개요입니다.

디바이스 액세스 API를 개발하려면 다음 헤더가 필요합니다.

- deviceaccess.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 디바이스 액세스 API

Functions

취소 Cancel 메서드는 진행 중인 비동기 작업을 취소하려고 시도합니다.
CancelOperation CancelOperation 메서드는 DeviceIoControlAsync 메서드를 사용하여 이전에 발급한 호출을 취소하려고 시도합니다.
닫기 Close 메서드는 비동기 작업이 완료된 후 정리를 수행하고 결과를 검색합니다.
CreateDeviceAccessInstance 디바이스에 액세스하는 데 사용되는 개체를 만듭니다. 인스턴스화된 개체는 IDeviceIoControl 및 ICreateDeviceAccessAsync 인터페이스를 구현합니다.
DeviceIoControlAsync DeviceIoControlAsync 메서드는 CreateDeviceAccessInstance 함수에 대한 호출이 지정된 디바이스 인터페이스에 비동기 디바이스 입력/출력(I/O) 제어 요청을 보냅니다.
DeviceIoControlSync DeviceIoControlSync 메서드는 CreateDeviceAccessInstance 함수에 대한 호출이 지정된 디바이스 인터페이스에 동기 디바이스 입력/출력(I/O) 제어 요청을 보냅니다.
GetResult

CreateDeviceAccessInstance 함수 호출에 지정된 디바이스 인터페이스에 바인딩된 IDeviceIoControl 개체를 검색합니다.
RequestCompletion RequestCompletion 메서드를 구현하여 DeviceIoControlAsyncmethod에 대한 호출 완료를 처리합니다.
Wait Wait 메서드는 진행 중인 비동기 바인딩 작업이 완료될 때까지 지정된 시간 길이를 기다립니다.

인터페이스

ICreateDeviceAccessAsync ICreateDeviceAccessAsync 인터페이스는 CreateDeviceAccessInstance 호출에서 반환됩니다.
IDeviceIoControl 디바이스 드라이버에 제어 코드를 보냅니다. 이 작업을 수행하면 디바이스가 해당 작업을 수행합니다.
IDeviceRequestCompletionCallback DeviceIoControlAsyncmethod에 대한 호출 완료를 처리하는 메서드를 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

디바이스 및 드라이버 설치 참조

아티클2024. 11. 15.

디바이스 및 드라이버 설치 참조 기술 개요입니다.

디바이스 및 드라이버 설치 참조를 개발하려면 다음 헤더가 필요합니다.

- cfg.h
- cfgmgr32.h
- devfiltertypes.h
- devquery.h
- devquerydef.h
- newdev.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 디바이스 및 드라이버 설치 참조

열거형

테이블 확장

CM_NOTIFY_ACTION 이 열거형은 플러그 앤 플레이 디바이스 이벤트 유형을 식별합니다.
DEV_OBJECT_TYPE DEV_OBJECT 형식을 지정합니다.
DEV_QUERY_FLAGS 디바이스 쿼리 동작을 변경하는 플래그를 지정합니다.
DEV_QUERY_RESULT_ACTION DEV_QUERY_RESULT_ACTION_DATA 구조체와 연결된 작업의 형식을 지정합니다.
DEV_QUERY_STATE DEV_QUERY_RESULT_ACTION_DATA 구조와 연결된 쿼리의 상태를 지정합니다.
DEVPROP_OPERATOR

DEVPROP_FILTER_EXPRESSION 사용할 수 있는 작업을 지정합니다.
PNP_VETO_TYPE PnP 관리자가 작업 수행 요청을 거부하는 경우 PNP_VETO_TYPE 열거형을 사용하여 거부 이유를 식별합니다.

함수

테이블 확장

CM_Add_Empty_Log_Conf CM_Add_Empty_Log_Conf 함수는 로컬 컴퓨터에서 지정된 구성 유형 및 지정된 디바이스 인스턴스에 대해 빈 논리 구성을 만듭니다.
CM_Add_Empty_Log_Conf_Ex CM_Add_Empty_Log_Conf_Ex 함수는 로컬 또는 원격 컴퓨터에서 지정된 구성 유형 및 지정된 디바이스 인스턴스에 대해 빈 논리 구성을 만듭니다.
CM_Add_ID_ExA CM_Add_ID_Ex 함수는 디바이스 ID(아직 없는 경우)를 로컬 또는 원격 컴퓨터의 디바이스 인스턴스의 하드웨어 ID 목록 또는 호환 ID 목록에 추가합니다. (ANSI)
CM_Add_ID_ExW CM_Add_ID_Ex 함수는 디바이스 ID(아직 없는 경우)를 로컬 또는 원격 컴퓨터의 디바이스 인스턴스의 하드웨어 ID 목록 또는 호환 ID 목록에 추가합니다. (유니코드)
CM_Add_IDA CM_Add_ID 함수는 지정된 디바이스 ID(아직 없는 경우)를 디바이스 인스턴스의 하드웨어 ID 목록 또는 호환 ID 목록에 추가합니다.
CM_Add_IDW CM_Add_ID 함수는 지정된 디바이스 ID(아직 없는 경우)를 디바이스 인스턴스의 하드웨어 ID 목록 또는 호환 ID 목록에 추가합니다.
CM_Add_Range

CM_Add_Res_Des CM_Add_Res_Des 함수는 논리 구성에 리소스 설명자를 추가합니다.
CM_Add_Res_Des_Ex CM_Add_Res_Des_Ex 함수는 논리 구성에 리소스 설명자를 추가합니다. 논리 구성은 로컬 또는 원격 컴퓨터에 있을 수 있습니다.
CM_Connect_MachineA CM_Connect_Machine 함수는 원격 컴퓨터에 대한 연결을 만듭니다. (ANSI)
CM_Connect_MachineW CM_Connect_Machine 함수는 원격 컴퓨터에 대한 연결을 만듭니다. (유니코드)
CM_Create_DevNode_ExA
CM_Create_DevNode_ExW
CM_Create_DevNodeA
CM_Create_DevNodeW
CM_Create_Range_List
CM_Delete_Class_Key CM_Delete_Class_Key 함수는 시스템에서 지정된 설치된 디바이스 클래스를 제거합니다.
CM_Delete_Class_Key_Ex
CM_Delete_Device_Interface_Key_ExA CM_Delete_Device_Interface_Key_ExA 함수는 애플리케이션 및 드라이버가 인터페이스 관련 정보를 저장하는 데 사용하는 레지스트리 하위 키를 삭제합니다.
CM_Delete_Device_Interface_Key_ExW CM_Delete_Device_Interface_Key_ExW 함수는 애플리케이션 및 드라이버가 인터페이스 관련 정보를 저장하는 데 사용하는 레지스트리 하위 키를 삭제합니다.

CM_Delete_Device_Interface_KeyA CM_Delete_Device_Interface_Key 함수는 애플리케이션 및 드라이버가 인터페이스 관련 정보를 저장하는 데 사용하는 레지스트리 하위 키를 삭제합니다. (ANSI)
CM_Delete_Device_Interface_KeyW CM_Delete_Device_Interface_Key 함수는 애플리케이션 및 드라이버가 인터페이스 관련 정보를 저장하는 데 사용하는 레지스트리 하위 키를 삭제합니다. (유니코드)
CM_Delete_DevNode_Key CM_Delete_DevNode_Key 함수는 디바이스와 연결된 지정된 사용자 액세스 가능 레지스트리 키를 삭제합니다.
CM_Delete_DevNode_Key_Ex
CM_Delete_Range
CM_Detect_Resource_Conflict
CM_Detect_Resource_Conflict_Ex
CM_Disable_DevNode CM_Disable_DevNode 함수는 디바이스를 사용하지 않도록 설정합니다.
CM_Disable_DevNode_Ex
CM_Disconnect_Machine CM_Disconnect_Machine 함수는 원격 컴퓨터에 대한 연결을 제거합니다.
CM_Dup_Range_List
CM_Enable_DevNode CM_Enable_DevNode 함수는 디바이스를 사용하도록 설정합니다.
CM_Enable_DevNode_Ex
CM_Enumerate_Classes

CM_Enumerate_Classes 함수는 반복적으로 호출될 때 각 클래스의 GUID를 제공하여 로컬 머신의 설치된 디바이스 클래스를 열거합니다.
CM_Enumerate_Classes_Ex CM_Enumerate_Classes_Ex 함수는 반복적으로 호출될 때 각 클래스의 GUID를 제공하여 로컬 또는 원격 머신의 설치된 디바이스 클래스를 열거합니다.
CM_Enumerate_Enumerators_ExA CM_Enumerate_Enumerators_Ex 함수는 각 열거자의 이름을 제공하여 로컬 또는 원격 컴퓨터의 디바이스 열거자를 열거합니다. (ANSI)
CM_Enumerate_Enumerators_ExW CM_Enumerate_Enumerators_Ex 함수는 각 열거자의 이름을 제공하여 로컬 또는 원격 컴퓨터의 디바이스 열거자를 열거합니다. (유니코드)
CM_Enumerate_EnumeratorsA CM_Enumerate_Enumerators 함수는 각 열거자의 이름을 제공하여 로컬 컴퓨터의 디바이스 열거자를 열거합니다. (ANSI)
CM_Enumerate_EnumeratorsW CM_Enumerate_Enumerators 함수는 각 열거자의 이름을 제공하여 로컬 컴퓨터의 디바이스 열거자를 열거합니다. (유니코드)
CM_Find_Range
CM_First_Range
CM_Free_Log_Conf CM_Free_Log_Conf 함수는 로컬 컴퓨터에서 논리 구성 및 연결된 모든 리소스 설명자를 제거합니다.
CM_Free_Log_Conf_Ex CM_Free_Log_Conf_Ex 함수는 로컬 또는 원격 컴퓨터에서 논리 구성 및 연결된 모든 리소스 설명자를 제거합니다.
CM_Free_Log_Conf_Handle CM_Free_Log_Conf_Handle 함수는 논리 구성 핸들을 무효화하고 연결된 메모리 할당을 해제합니다.

CM_Free_Range_List
CM_Free_Res_Des CM_Free_Res_Des 함수는 로컬 컴퓨터의 논리적 구성에서 리소스 설명자를 제거합니다.
CM_Free_Res_Des_Ex CM_Free_Res_Des_Ex 함수는 로컬 또는 원격 컴퓨터의 논리 구성에서 리소스 설명자를 제거합니다.
CM_Free_Res_Des_Handle CM_Free_Res_Des_Handle 함수는 리소스 설명 핸들을 무효화하고 연결된 메모리 할당을 해제합니다.
CM_Free_Resource_Conflict_Handle CM_Free_Resource_Conflict_Handle 함수는 리소스 충돌 목록에 대한 핸들을 무효화하고 핸들의 연결된 메모리 할당을 해제합니다.
CM_Get_Child CM_Get_Child 함수는 로컬 컴퓨터의 디바이스 트리에서 지정된 디바이스 노드(devnode)의 첫 번째 자식 노드에 대한 디바이스 인스턴스 핸들을 검색하는 데 사용됩니다.
CM_Get_Child_Ex CM_Get_Child_Ex 함수는 로컬 또는 원격 머신의 디바이스 트리에서 지정된 디바이스 노드(devnode)의 첫 번째 자식 노드에 대한 디바이스 인스턴스 핸들을 검색하는 데 사용됩니다.
CM_Get_Class_Key_Name_ExA
CM_Get_Class_Key_Name_ExW
CM_Get_Class_Key_NameA
CM_Get_Class_Key_NameW
CM_Get_Class_Name_ExA
CM_Get_Class_Name_ExW

CM_Get_Class_NameA
CM_Get_Class_NameW
CM_Get_Class_Property_ExW CM_Get_Class_Property_ExW 함수는 디바이스 인터페이스 클래스 또는 디바이스 설정 클래스에 대해 설정된 디바이스 속성을 검색합니다.
CM_Get_Class_Property_Keys CM_Get_Class_Property_Keys 함수는 디바이스 인터페이스 클래스 또는 디바이스 설정 클래스에 대해 설정된 디바이스 속성을 나타내는 디바이스 속성 키의 배열을 검색합니다.
CM_Get_Class_Property_Keys_Ex CM_Get_Class_Property_Keys_Ex 함수는 디바이스 인터페이스 클래스 또는 디바이스 설정 클래스에 대해 설정된 디바이스 속성을 나타내는 디바이스 속성 키의 배열을 검색합니다.
CM_Get_Class_PropertyW CM_Get_Class_Property 함수는 디바이스 인터페이스 클래스 또는 디바이스 설정 클래스에 대해 설정된 디바이스 속성을 검색합니다.
CM_Get_Class_Registry_PropertyA CM_Get_Class_Registry_Property 함수는 디바이스 설정 클래스 속성을 검색합니다. (ANSI)
CM_Get_Class_Registry_PropertyW CM_Get_Class_Registry_Property 함수는 디바이스 설정 클래스 속성을 검색합니다. (유니코드)
CM_Get_Depth CM_Get_Depth 함수는 로컬 컴퓨터의 디바이스 트리 내에서 지정된 디바이스 노드(devnode)의 깊이를 가져오는 데 사용됩니다.
CM_Get_Depth_Ex CM_Get_Depth_Ex 함수는 로컬 또는 원격 머신의 디바이스 트리 내에서 지정된 디바이스 노드(devnode)의 깊이를 가져오는 데 사용됩니다.
CM_Get_Device_ID_ExA CM_Get_Device_ID_Ex 함수는 로컬 또는 원격 컴퓨터에서 지정된 디바이스 인스턴스에 대한 디바이스 인스턴스 ID를 검색합니다. (ANSI)

CM_Get_Device_ID_ExW CM_Get_Device_ID_Ex 함수는 로컬 또는 원격 컴퓨터에서 지정된 디바이스 인스턴스에 대한 디바이스 인스턴스 ID를 검색합니다. (유니코드)
CM_Get_Device_ID_List_ExA CM_Get_Device_ID_List_Ex 함수는 로컬 또는 원격 컴퓨터의 디바이스 인스턴스에 대한 디바이스 인스턴스 ID 목록을 검색합니다. (ANSI)
CM_Get_Device_ID_List_ExW CM_Get_Device_ID_List_Ex 함수는 로컬 또는 원격 컴퓨터의 디바이스 인스턴스에 대한 디바이스 인스턴스 ID 목록을 검색합니다. (유니코드)
CM_Get_Device_ID_List_Size_ExA CM_Get_Device_ID_List_Size_Ex 함수는 로컬 또는 원격 컴퓨터의 디바이스 인스턴스에 대한 디바이스 인스턴스 ID 목록을 보유하는 데 필요한 버퍼 크기를 검색합니다. (ANSI)
CM_Get_Device_ID_List_Size_ExW CM_Get_Device_ID_List_Size_Ex 함수는 로컬 또는 원격 컴퓨터의 디바이스 인스턴스에 대한 디바이스 인스턴스 ID 목록을 보유하는 데 필요한 버퍼 크기를 검색합니다. (유니코드)
CM_Get_Device_ID_List_SizeA CM_Get_Device_ID_List_Size 함수는 로컬 컴퓨터의 디바이스 인스턴스에 대한 디바이스 인스턴스 ID 목록을 보유하는 데 필요한 버퍼 크기를 검색합니다. (ANSI)
CM_Get_Device_ID_List_SizeW CM_Get_Device_ID_List_Size 함수는 로컬 컴퓨터의 디바이스 인스턴스에 대한 디바이스 인스턴스 ID 목록을 보유하는 데 필요한 버퍼 크기를 검색합니다. (유니코드)
CM_Get_Device_ID_ListA CM_Get_Device_ID_List 함수는 로컬 컴퓨터의 디바이스 인스턴스에 대한 디바이스 인스턴스 ID 목록을 검색합니다. (ANSI)
CM_Get_Device_ID_ListW CM_Get_Device_ID_List 함수는 로컬 컴퓨터의 디바이스 인스턴스에 대한 디바이스 인스턴스 ID 목록을 검색합니다. (유니코드)
CM_Get_Device_ID_Size CM_Get_Device_ID_Size 함수는 로컬 컴퓨터의 디바이스 인스턴스에 대한 디바이스 인스턴스 ID를 보유하는 데 필요한 버퍼 크기를 검색합니다.

CM_Get_Device_ID_Size_Ex CM_Get_Device_ID_Size_Ex 함수는 로컬 또는 원격 컴퓨터에서 디바이스 인스턴스에 대한 디바이스 인스턴스 ID를 보유하는 데 필요한 버퍼 크기를 검색합니다.
CM_Get_Device_IDA CM_Get_Device_ID 함수는 로컬 컴퓨터에서 지정된 디바이스 인스턴스에 대한 디바이스 인스턴스 ID를 검색합니다. (ANSI)
CM_Get_Device_IDW CM_Get_Device_ID 함수는 로컬 컴퓨터에서 지정된 디바이스 인스턴스에 대한 디바이스 인스턴스 ID를 검색합니다. (유니코드)
CM_Get_Device_Interface_Alias_ExA
CM_Get_Device_Interface_Alias_ExW
CM_Get_Device_Interface_AliasA CM_Get_Device_Interface_Alias 함수는 별칭이 있는 경우 지정된 디바이스 인터페이스 인스턴스의 별칭을 반환합니다. (ANSI)
CM_Get_Device_Interface_AliasW CM_Get_Device_Interface_Alias 함수는 별칭이 있는 경우 지정된 디바이스 인터페이스 인스턴스의 별칭을 반환합니다. (유니코드)
CM_Get_Device_Interface_List_ExA
CM_Get_Device_Interface_List_ExW
CM_Get_Device_Interface_List_Size_ExA
CM_Get_Device_Interface_List_Size_ExW
CM_Get_Device_Interface_List_SizeA CM_Get_Device_Interface_List_Size 함수는 CM_Get_Device_Interface_List 함수에 전달해야 하는 버퍼 크기를 검색합니다. (ANSI)
CM_Get_Device_Interface_List_SizeW

CM_Get_Device_Interface_List_Size 함수는 CM_Get_Device_Interface_List 함수에 전달해야 하는 버퍼 크기를 검색합니다. (유니코드)
CM_Get_Device_Interface_ListA CM_Get_Device_Interface_List 함수는 지정된 디바이스 인터페이스 클래스에 속하는 디바이스 인터페이스 인스턴스 목록을 검색합니다. (ANSI)
CM_Get_Device_Interface_ListW CM_Get_Device_Interface_List 함수는 지정된 디바이스 인터페이스 클래스에 속하는 디바이스 인터페이스 인스턴스 목록을 검색합니다. (유니코드)
CM_Get_Device_Interface_Property_ExW CM_Get_Device_Interface_Property_ExW 함수는 디바이스 인터페이스에 대해 설정된 디바이스 속성을 검색합니다.
CM_Get_Device_Interface_Property_Keys_ExW CM_Get_Device_Interface_Property_Keys_ExW 함수는 디바이스 인터페이스에 대해 설정된 디바이스 속성을 나타내는 디바이스 속성 키 배열을 검색합니다.
CM_Get_Device_Interface_Property_KeysW CM_Get_Device_Interface_Property_Keys 함수는 디바이스 인터페이스에 대해 설정된 디바이스 속성을 나타내는 디바이스 속성 키 배열을 검색합니다.
CM_Get_Device_Interface_PropertyW CM_Get_Device_Interface_Property 함수는 디바이스 인터페이스에 대해 설정된 디바이스 속성을 검색합니다.
CM_Get_DevNode_Custom_Property_ExA
CM_Get_DevNode_Custom_Property_ExW
CM_Get_DevNode_Custom_PropertyA
CM_Get_DevNode_Custom_PropertyW
CM_Get_DevNode_Property_ExW CM_Get_DevNode_Property_ExW 함수는 디바이스 인스턴스 속성을 검색합니다.

CM_Get_DevNode_Property_Keys CM_Get_DevNode_Property_Keys 함수는 디바이스 인스턴스에 대해 설정된 디바이스 속성을 나타내는 디바이스 속성 키의 배열을 검색합니다.
CM_Get_DevNode_Property_Keys_Ex CM_Get_DevNode_Property_Keys_Ex 함수는 디바이스 인스턴스에 대해 설정된 디바이스 속성을 나타내는 디바이스 속성 키의 배열을 검색합니다.
CM_Get_DevNode_PropertyW CM_Get_DevNode_Property 함수는 디바이스 인스턴스 속성을 검색합니다.
CM_Get_DevNode_Registry_Property_ExA
CM_Get_DevNode_Registry_Property_ExW
CM_Get_DevNode_Registry_PropertyA CM_Get_DevNode_Registry_Property 함수는 레지스트리에서 지정된 디바이스 속성을 검색합니다. (ANSI)
CM_Get_DevNode_Registry_PropertyW CM_Get_DevNode_Registry_Property 함수는 레지스트리에서 지정된 디바이스 속성을 검색합니다. (유니코드)
CM_Get_DevNode_Status CM_Get_DevNode_Status 함수는 로컬 컴퓨터의 디바이스 트리에 있는 디바이스 노드(devnode)에서 디바이스 인스턴스의 상태를 가져옵니다.
CM_Get_DevNode_Status_Ex CM_Get_DevNode_Status_Ex 함수는 로컬 또는 원격 머신의 디바이스 트리에 있는 디바이스 노드(devnode)에서 디바이스 인스턴스의 상태를 가져옵니다.
CM_Get_First_Log_Conf CM_Get_First_Log_Conf 함수는 로컬 컴퓨터의 지정된 디바이스 인스턴스와 연결된 지정된 구성 유형의 첫 번째 논리 구성을 가져옵니다.
CM_Get_First_Log_Conf_Ex CM_Get_First_Log_Conf_Ex 함수는 로컬 또는 원격 컴퓨터에서 지정된 디바이스 인스턴스와 연결된 첫 번째 논리 구성을 가져옵니다.

CM_Get_Global_State
CM_Get_Global_State_Ex
CM_Get_Hardware_Profile_Info_ExA
CM_Get_Hardware_Profile_Info_ExW
CM_Get_Hardware_Profile_InfoA
CM_Get_Hardware_Profile_InfoW
CM_Get_HW_Prof_Flags_ExA CM_Get_HW_Prof_Flags_Ex 함수는 원격 컴퓨터 또는 로컬 컴퓨터의 디바이스 인스턴스에 대한 하드웨어 프로필별 구성 플래그를 검색합니다. (ANSI)
CM_Get_HW_Prof_Flags_ExW CM_Get_HW_Prof_Flags_Ex 함수는 원격 컴퓨터 또는 로컬 컴퓨터의 디바이스 인스턴스에 대한 하드웨어 프로필별 구성 플래그를 검색합니다. (유니코드)
CM_Get_HW_Prof_FlagsA CM_Get_HW_Prof_Flags 함수는 로컬 컴퓨터의 디바이스 인스턴스에 대한 하드웨어 프로필별 구성 플래그를 검색합니다. (ANSI)
CM_Get_HW_Prof_FlagsW CM_Get_HW_Prof_Flags 함수는 로컬 컴퓨터의 디바이스 인스턴스에 대한 하드웨어 프로필별 구성 플래그를 검색합니다. (유니코드)
CM_Get_Log_Conf_Priority CM_Get_Log_Conf_Priority 함수는 로컬 컴퓨터에서 지정된 논리 구성의 구성 우선 순위를 가져옵니다.
CM_Get_Log_Conf_Priority_Ex CM_Get_Log_Conf_Priority_Ex 함수는 로컬 또는 원격 컴퓨터에서 지정된 논리 구성의 구성 우선 순위를 가져옵니다.
CM_Get_Next_Log_Conf

CM_Get_Next_Log_Conf 함수는 로컬 컴퓨터의 특정 디바이스 인스턴스와 연결된 다음 논리 구성을 가져옵니다.

CM_Get_Next_Log_Conf_Ex

CM_Get_Next_Log_Conf_Ex 함수는 로컬 또는 원격 컴퓨터의 특정 디바이스 인스턴스와 연결된 다음 논리 구성을 가져옵니다.

CM_Get_Next_Res_Des

CM_Get_Next_Res_Des 함수는 로컬 컴퓨터의 논리적 구성에 대해 지정된 리소스 유형의 다음 리소스 설명자에 대한 핸들을 가져옵니다.

CM_Get_Next_Res_Des_Ex

CM_Get_Next_Res_Des_Ex 함수는 로컬 또는 원격 컴퓨터의 논리적 구성에 대해 지정된 리소스 유형의 다음 리소스 설명자에 대한 핸들을 가져옵니다.

CM_Get_Parent

CM_Get_Parent 함수는 로컬 컴퓨터의 디바이스 트리에서 지정된 디바이스 노드(devnode)의 부모 노드에 대한 디바이스 인스턴스 핸들을 가져옵니다.

CM_Get_Parent_Ex

CM_Get_Parent_Ex 함수는 로컬 또는 원격 머신의 디바이스 트리에서 지정된 디바이스 노드(devnode)의 부모 노드에 대한 디바이스 인스턴스 핸들을 가져옵니다.

CM_Get_Res_Des_Data

CM_Get_Res_Des_Data 함수는 로컬 컴퓨터의 리소스 설명자에 저장된 정보를 검색합니다.

CM_Get_Res_Des_Data_Ex

CM_Get_Res_Des_Data_Ex 함수는 로컬 또는 원격 컴퓨터의 리소스 설명자에 저장된 정보를 검색합니다.

CM_Get_Res_Des_Data_Size

CM_Get_Res_Des_Data_Size 함수는 로컬 컴퓨터의 지정된 리소스 설명자에 포함된 정보를 보관하는 데 필요한 버퍼 크기를 가져옵니다.

CM_Get_Res_Des_Data_Size_Ex

CM_Get_Res_Des_Data_Size_Ex 함수는 로컬 또는 원격 컴퓨터의 지정된 리소스 설명자에 포함된 정보를 보관하는 데 필요한 버퍼 크기를 가져옵니다.

CM_Get_Resource_Conflict_Count

CM_Get_Resource_Conflict_Count 함수는 지정된 리소스 충돌 목록에 포함된 충돌 횟수를 가져옵니다.
CM_Get_Resource_Conflict_DetailsA CM_Get_Resource_Conflict_Details 함수는 충돌 목록의 리소스 충돌 중 하나에 대한 세부 정보를 가져옵니다. (ANSI)
CM_Get_Resource_Conflict_DetailsW CM_Get_Resource_Conflict_Details 함수는 충돌 목록의 리소스 충돌 중 하나에 대한 세부 정보를 가져옵니다. (유니코드)
CM_Get_Sibling CM_Get_Sibling 함수는 로컬 컴퓨터의 디바이스 트리에서 지정된 디바이스 노드(devnode)의 다음 형제 노드에 대한 디바이스 인스턴스 핸들을 가져옵니다.
CM_Get_Sibling_Ex CM_Get_Sibling_Ex 함수는 로컬 또는 원격 컴퓨터의 디바이스 트리에서 지정된 디바이스 노드의 다음 형제 노드에 대한 디바이스 인스턴스 핸들을 가져옵니다.
CM_Get_Version CM_Get_Version 함수는 로컬 컴퓨터에 대한 PnP(플러그 앤 플레이) Configuration Manager DLL(Cfgmgr32.dll)의 버전 4.0을 반환합니다.
CM_Get_Version_Ex CM_Get_Version_Ex 함수는 로컬 또는 원격 컴퓨터에 대한 PnP(플러그 앤 플레이) Configuration Manager DLL(Cfgmgr32.dll)의 버전 4.0을 반환합니다.
CM_Intersect_Range_List
CM_Invert_Range_List
CM_Is_Dock_Station_Present CM_Is_Dock_Station_Present 함수는 도킹 스테이션이 로컬 컴퓨터에 있는지 여부를 식별합니다.
CM_Is_Dock_Station_Present_Ex CM_Is_Dock_Station_Present_Ex 함수는 도킹 스테이션이 로컬 또는 원격 컴퓨터에 있는지 여부를 식별합니다.
CM_Is_Version_Available

CM_Is_Version_Available 함수는 지정된 버전의 PnP(플러그 앤 플레이) Configuration Manager DLL(Cfgmgr32.dll)이 로컬 컴퓨터에서 지원되는지 여부를 나타냅니다.
CM_Is_Version_Available_Ex CM_Is_Version_Available_Ex 함수는 지정된 버전의 PNP(플러그 앤 플레이) Configuration Manager DLL(Cfgmgr32.dll)이 로컬 또는 원격 컴퓨터에서 지원되는지 여부를 나타냅니다.
CM_Locate_DevNode_ExA CM_Locate_DevNode_Ex 함수는 로컬 컴퓨터 또는 원격 컴퓨터에서 지정된 디바이스 인스턴스 ID와 연결된 디바이스 노드에 대한 디바이스 인스턴스 핸들을 가져옵니다. (ANSI)
CM_Locate_DevNode_ExW CM_Locate_DevNode_Ex 함수는 로컬 컴퓨터 또는 원격 컴퓨터에서 지정된 디바이스 인스턴스 ID와 연결된 디바이스 노드에 대한 디바이스 인스턴스 핸들을 가져옵니다. (유니코드)
CM_Locate_DevNodeA CM_Locate_DevNode 함수는 로컬 컴퓨터에서 지정된 디바이스 인스턴스 ID와 연결된 디바이스 노드에 대한 디바이스 인스턴스 핸들을 가져옵니다. (ANSI)
CM_Locate_DevNodeW CM_Locate_DevNode 함수는 로컬 컴퓨터에서 지정된 디바이스 인스턴스 ID와 연결된 디바이스 노드에 대한 디바이스 인스턴스 핸들을 가져옵니다. (유니코드)
CM_MapCrToWin32Err 지정된 CONFIGRET 코드를 해당하는 시스템 오류 코드로 변환합니다.
CM_Merge_Range_List
CM_Modify_Res_Des CM_Modify_Res_Des 함수는 로컬 컴퓨터에서 지정된 리소스 설명자를 수정합니다.
CM_Modify_Res_Des_Ex CM_Modify_Res_Des_Ex 함수는 로컬 또는 원격 컴퓨터에서 지정된 리소스 설명자를 수정합니다.
CM_Move_DevNode
CM_Move_DevNode_Ex

CM_Next_Range
CM_Open_Class_Key_ExA
CM_Open_Class_Key_ExW
CM_Open_Class_KeyA CM_Open_Class_Key 함수는 디바이스 설정 클래스 레지스트리 키, 디바이스 인터페이스 클래스 레지스트리 키 또는 클래스의 특정 하위 키를 엽니다. (ANSI)
CM_Open_Class_KeyW CM_Open_Class_Key 함수는 디바이스 설정 클래스 레지스트리 키, 디바이스 인터페이스 클래스 레지스트리 키 또는 클래스의 특정 하위 키를 엽니다. (유니코드)
CM_Open_Device_Interface_Key_ExA CM_Open_Device_Interface_Key_ExA 함수는 애플리케이션 및 드라이버가 디바이스 인터페이스와 관련된 정보를 저장하는 데 사용하는 레지스트리 하위 키를 엽니다.
CM_Open_Device_Interface_Key_ExW CM_Open_Device_Interface_Key_ExW 함수는 애플리케이션 및 드라이버가 디바이스 인터페이스와 관련된 정보를 저장하는 데 사용하는 레지스트리 하위 키를 엽니다.
CM_Open_Device_Interface_KeyA CM_Open_Device_Interface_Key 함수는 애플리케이션 및 드라이버가 디바이스 인터페이스와 관련된 정보를 저장하는 데 사용하는 레지스트리 하위 키를 엽니다. (ANSI)
CM_Open_Device_Interface_KeyW CM_Open_Device_Interface_Key 함수는 애플리케이션 및 드라이버가 디바이스 인터페이스와 관련된 정보를 저장하는 데 사용하는 레지스트리 하위 키를 엽니다. (유니코드)
CM_Open_DevNode_Key CM_Open_DevNode_Key 함수는 디바이스별 구성 정보에 대한 레지스트리 키를 엽니다.
CM_Open_DevNode_Key_Ex
CM_Query_And_Remove_SubTree_ExA

CM_Query_And_Remove_SubTree_Ex 함수는 디바이스 인스턴스와 해당 자식을 제거할 수 있는지 여부를 확인하고, 제거하면 제거합니다. (ANSI)
CM_Query_And_Remove_SubTree_ExW CM_Query_And_Remove_SubTree_Ex 함수는 디바이스 인스턴스와 해당 자식을 제거할 수 있는지 여부를 확인하고, 제거하면 제거합니다. (유니코드)
CM_Query_And_Remove_SubTreeA CM_Query_And_Remove_SubTree 함수는 디바이스 인스턴스와 해당 자식을 제거할 수 있는지 여부를 확인하고, 제거하면 제거합니다. (ANSI)
CM_Query_And_Remove_SubTreeW CM_Query_And_Remove_SubTree 함수는 디바이스 인스턴스와 해당 자식을 제거할 수 있는지 여부를 확인하고, 제거하면 제거합니다. (유니코드)
CM_Query_Arbitrator_Free_Data
CM_Query_Arbitrator_Free_Data_Ex
CM_Query_Arbitrator_Free_Size
CM_Query_Arbitrator_Free_Size_Ex
CM_Query_Remove_SubTree
CM_Query_Remove_SubTree_Ex
CM_Query_Resource_Conflict_List CM_Query_Resource_Conflict_List 함수는 지정된 디바이스 인스턴스의 리소스 설명과 충돌하는 리소스 요구 사항이 있는 디바이스 인스턴스를 식별합니다.
CM_Reenumerate_DevNode CM_Reenumerate_DevNode 함수는 지정된 디바이스 노드 및 모든 자식으로 식별되는 디바이스를 열거합니다.
CM_Reenumerate_DevNode_Ex CM_Reenumerate_DevNode_Ex 함수는 지정된 디바이스 노드 및 모든 자식으로 식별되는 디바이스를 열거합니다.

CM_Register_Device_Driver
CM_Register_Device_Driver_Ex
CM_Register_Device_Interface_ExA
CM_Register_Device_Interface_ExW
CM_Register_Device_InterfaceA
CM_Register_Device_InterfaceW
CM_Register_Notification CM_Register_Notification 함수는 지정된 형식의 PnP 이벤트가 발생할 때 호출할 애플리케이션 콜백 루틴을 등록합니다.
CM_Remove_SubTree
CM_Remove_SubTree_Ex
CM_Request_Device_Eject_ExA CM_Request_Device_Eject_Ex 함수는 디바이스가 이동식인 경우 안전한 제거를 위해 로컬 또는 원격 디바이스 인스턴스를 준비합니다. 디바이스를 물리적으로 배출할 수 있으면 해당 디바이스가 됩니다. (ANSI)
CM_Request_Device_Eject_ExW CM_Request_Device_Eject_Ex 함수는 디바이스가 이동식인 경우 안전한 제거를 위해 로컬 또는 원격 디바이스 인스턴스를 준비합니다. 디바이스를 물리적으로 배출할 수 있으면 해당 디바이스가 됩니다. (유니코드)
CM_Request_Device_EjectA CM_Request_Device_Eject 함수는 디바이스가 이동식인 경우 안전한 제거를 위해 로컬 디바이스 인스턴스를 준비합니다. 디바이스를 물리적으로 배출할 수 있으면 해당 디바이스가 됩니다. (ANSI)
CM_Request_Device_EjectW CM_Request_Device_Eject 함수는 디바이스가 이동식인 경우 안전한 제거를 위해 로컬 디바이스 인

스턴스를 준비합니다. 디바이스를 물리적으로 배출할 수 있으면 해당 디바이스가 됩니다. (유니코드)
CM_Request_Eject_PC CM_Request_Eject_PC 함수는 로컬 도킹 스테이션에 삽입된 휴대용 PC를 배출할 것을 요청합니다.
CM_Request_Eject_PC_Ex CM_Request_Eject_PC_Ex 함수는 로컬 또는 원격 도킹 스테이션에 삽입된 휴대용 PC를 배출할 것을 요청합니다.
CM_Run_Detection
CM_Run_Detection_Ex
CM_Set_Class_Property_ExW CM_Set_Class_Property_ExW 함수는 디바이스 설정 클래스 또는 디바이스 인터페이스 클래스에 대한 클래스 속성을 설정합니다.
CM_Set_Class_PropertyW CM_Set_Class_Property 함수는 디바이스 설정 클래스 또는 디바이스 인터페이스 클래스에 대한 클래스 속성을 설정합니다.
CM_Set_Class_Registry_PropertyA CM_Set_Class_Registry_Property 함수는 디바이스 설정 클래스의 속성을 설정하거나 삭제합니다. (ANSI)
CM_Set_Class_Registry_PropertyW CM_Set_Class_Registry_Property 함수는 디바이스 설정 클래스의 속성을 설정하거나 삭제합니다. (유니코드)
CM_Set_Device_Interface_Property_ExW CM_Set_Device_Interface_Property_ExW 함수는 디바이스 인터페이스의 디바이스 속성을 설정합니다.
CM_Set_Device_Interface_PropertyW CM_Set_Device_Interface_Property 함수는 디바이스 인터페이스의 디바이스 속성을 설정합니다.
CM_Set_DevNode_Problem

CM_Set_DevNode_Problem 함수는 로컬 컴퓨터에 설치된 디바이스에 대한 문제 코드를 설정합니다.
CM_Set_DevNode_Problem_Ex CM_Set_DevNode_Problem_Ex 함수는 로컬 또는 원격 컴퓨터에 설치된 디바이스에 대한 문제 코드를 설정합니다.
CM_Set_DevNode_Property_ExW CM_Set_DevNode_Property_ExW 함수는 디바이스 인스턴스 속성을 설정합니다.
CM_Set_DevNode_PropertyW CM_Set_DevNode_Property 함수는 디바이스 인스턴스 속성을 설정합니다.
CM_Set_DevNode_Registry_Property_ExA
CM_Set_DevNode_Registry_Property_ExW
CM_Set_DevNode_Registry_PropertyA CM_Set_DevNode_Registry_Property 함수는 레지스트리에서 지정된 디바이스 속성을 설정합니다. (ANSI)
CM_Set_DevNode_Registry_PropertyW CM_Set_DevNode_Registry_Property 함수는 레지스트리에서 지정된 디바이스 속성을 설정합니다. (유니코드)
CM_Set_HW_Prof
CM_Set_HW_Prof_Ex
CM_Set_HW_Prof_Flags_ExA
CM_Set_HW_Prof_Flags_ExW
CM_Set_HW_Prof_FlagsA
CM_Set_HW_Prof_FlagsW

CM_Setup_DevNode CM_Setup_DevNode 함수는 디바이스 구성에 문제가 있으므로 실행되지 않는 디바이스 인스턴스를 다시 시작합니다.
CM_Setup_DevNode_Ex
CM_Test_Range_Available
CM_Uninstall_DevNode CM_Uninstall_DevNode 함수는 디바이스 인스턴스와 연결된 모든 영구 상태를 제거합니다.
CM_Uninstall_DevNode_Ex
CM_Unregister_Device_Interface_ExA
CM_Unregister_Device_Interface_ExW
CM_Unregister_Device_InterfaceA
CM_Unregister_Device_InterfaceW
CM_Unregister_Notification 코드가 Windows 7 또는 이전 버전의 Windows를 대상으로 하는 경우 CM_Unregister_Notification 대신 UnregisterDeviceNotification을 사용합니다.
CM_WaitNoPendingInstallEvents CMP_WaitNoPendingInstallEvents(CM_WaitNoPendingInstallEvents) 함수는 PnP 관리자가 수행할 보류 중인 디바이스 설치 작업이 없을 때까지 기다립니다.
DevCloseObjectQuery 디바이스 쿼리 핸들을 닫습니다.
DevCreateObjectQuery 지정된 쿼리 매개 변수를 기반으로 속성을 검색하는 디바이스 쿼리를 만듭니다.
DevCreateObjectQueryEx

지정된 쿼리 매개 변수 및 확장 매개 변수를 기반으로 속성을 검색하는 디바이스 쿼리를 만듭니다.
devCreateObjectQueryFromId 지정된 쿼리 매개 변수 및 개체 ID를 기반으로 속성을 검색하는 디바이스 쿼리를 만듭니다.
DevCreateObjectQueryFromIdEx 지정된 쿼리 매개 변수, 확장 매개 변수 및 개체 ID를 기반으로 속성을 검색하는 디바이스 쿼리를 만듭니다.
devCreateObjectQueryFromIds 지정된 쿼리 매개 변수 및 개체 ID 목록을 기반으로 속성을 검색하는 디바이스 쿼리를 만듭니다.
DevCreateObjectQueryFromIdsEx 지정된 쿼리 매개 변수, 확장 매개 변수 및 개체 ID 목록을 기반으로 속성을 검색하는 디바이스 쿼리를 만듭니다.
devFindProperty DEVPROPERTY 구조체 배열 내의 특정 속성에 해당하는 DEVPROPERTY를 찾습니다.
DevFreeObjectProperties DevGetObjectProperties 또는 DevGetObjectPropertiesEx 호출에 의해 할당된 DEVPROPERTY 구조를 해제합니다.
devFreeObjects DevGetObjects 또는 DevGetObjectsEx를 호출하여 할당된 DEV_OBJECT 구조를 해제합니다.
DevGetObjectProperties 지정된 개체에 대한 속성 집합을 동기적으로 검색합니다.
DevGetObjectPropertiesEx 지정된 확장 매개 변수에 따라 필요에 따라 지정된 개체에 대한 속성 집합을 동기적으로 검색합니다.
devGetObjects 제공된 요청된 속성 및 필터 조건에 따라 DEV_OBJECT 구조 집합을 동기적으로 검색합니다.
DevGetObjectsEx

제공된 요청된 속성, 확장 매개 변수 및 필터 조건에 따라 DEV_OBJECT 구조 집합을 동기적으로 검색합니다.
DiInstallDevice DiInstallDevice 함수는 시스템에 있는 지정된 디바이스의 드라이버 저장소에 미리 설치된 지정된 드라이버를 설치합니다.
DiInstallDriverA DiInstallDriver 함수는 드라이버 저장소에 드라이버를 미리 설치한 다음 드라이버가 지원하는 시스템에 있는 디바이스에 드라이버를 설치합니다. (ANSI)
DiInstallDriverW DiInstallDriver 함수는 드라이버 저장소에 드라이버를 미리 설치한 다음 드라이버가 지원하는 시스템에 있는 디바이스에 드라이버를 설치합니다. (유니코드)
DiRollbackDriver DiRollbackDriver 함수는 지정된 디바이스에 설치된 드라이버를 롤백합니다.
DiShowUpdateDevice DiShowUpdateDevice 함수는 지정된 디바이스에 대한 하드웨어 업데이트 마법사를 표시합니다.
DiUninstallDevice DiUninstallDevice 함수는 디바이스를 제거하고 시스템에서 디바이스 노드(devnode)를 제거합니다.
DiUninstallDriverA DiUninstallDriver 함수는 일치하는 다른 드라이버 패키지(사용 가능한 경우)를 사용하여 해당 디바이스를 설치하여 설치된 모든 장치에서 드라이버 패키지를 제거하거나, 일치하는 다른 드라이버 패키지를 사용할 수 없는 경우 null 드라이버를 제거합니다. 그런 다음 지정된 드라이버 패키지가 드라이버 저장소에서 제거됩니다. (ANSI)
DiUninstallDriverW DiUninstallDriver 함수는 일치하는 다른 드라이버 패키지(사용 가능한 경우)를 사용하여 해당 디바이스를 설치하여 설치된 모든 장치에서 드라이버 패키지를 제거하거나, 일치하는 다른 드라이버 패키지를 사용할 수 없는 경우 null 드라이버를 제거합니다. 그런 다음 지정된 드라이버 패키지가 드라이버 저장소에서 제거됩니다. (유니코드)
PDEV_QUERY_RESULT_CALLBACK DevQuery 쿼리 결과를 수신하는 콜백 함수에 필요한 함수 프로토타입입니다.
SetupDiAskForOEMDisk

SetupDiAskForOEMDisk 함수는 사용자에게 OEM 설치 디스크의 경로를 요청하는 대화 상자를 표시합니다.
SetupDiBuildClassInfoList SetupDiBuildClassInfoList 함수는 로컬 컴퓨터에 설치된 클래스를 식별하는 설치 클래스 GUID 목록을 반환합니다.
SetupDiBuildClassInfoListExA SetupDiBuildClassInfoListEx 함수는 로컬 시스템 또는 원격 시스템에 설치된 모든 클래스를 포함하는 설치 클래스 GUID 목록을 반환합니다. (ANSI)
SetupDiBuildClassInfoListExW SetupDiBuildClassInfoListEx 함수는 로컬 시스템 또는 원격 시스템에 설치된 모든 클래스를 포함하는 설치 클래스 GUID 목록을 반환합니다. (유니코드)
SetupDiBuildDriverInfoList SetupDiBuildDriverInfoList 함수는 특정 디바이스 또는 디바이스 정보 집합에 대한 전역 클래스 드라이버 목록과 연결된 드라이버 목록을 빌드합니다.
SetupDiCallClassInstaller SetupDiCallClassInstaller 함수는 지정된 설치 요청(DIF 코드)을 사용하여 적절한 클래스 설치 관리자 및 등록된 모든 공동 설치 관리자를 호출합니다.
SetupDiCancelDriverInfoSearch SetupDiCancelDriverInfoSearch 함수는 현재 다른 스레드에서 진행 중인 드라이버 목록 검색을 취소합니다.
SetupDiChangeState SetupDiChangeState 함수는 DIF_PROPERTYCHANGE 설치 요청에 대한 기본 처리기입니다.
SetupDiClassGuidsFromNameA SetupDiClassGuidsFromName 함수는 지정된 클래스 이름과 연결된 GUID를 검색합니다. 이 목록은 현재 시스템에 설치된 클래스를 기반으로 빌드됩니다. (ANSI)
SetupDiClassGuidsFromNameExA SetupDiClassGuidsFromNameEx 함수는 지정된 클래스 이름과 연결된 GUID를 검색합니다. 이 결과 목록에는 현재 로컬 또는 원격 컴퓨터에 설치된 클래스가 포함됩니다. (ANSI)
SetupDiClassGuidsFromNameExW

SetupDiClassGuidsFromNameEx 함수는 지정된 클래스 이름과 연결된 GUID를 검색합니다. 이 결과 목록에는 현재 로컬 또는 원격 컴퓨터에 설치된 클래스가 포함됩니다. (유니코드)
SetupDiClassGuidsFromNameW SetupDiClassGuidsFromName 함수는 지정된 클래스 이름과 연결된 GUID를 검색합니다. 이 목록은 현재 시스템에 설치된 클래스를 기반으로 빌드됩니다. (유니코드)
SetupDiClassNameFromGuidA SetupDiClassNameFromGuid 함수는 클래스 GUID와 연결된 클래스 이름을 검색합니다. (ANSI)
setupDiClassNameFromGuidExA SetupDiClassNameFromGuidEx 함수는 클래스 GUID와 연결된 클래스 이름을 검색합니다. 클래스는 로컬 또는 원격 컴퓨터에 설치할 수 있습니다. (ANSI)
SetupDiClassNameFromGuidExW SetupDiClassNameFromGuidEx 함수는 클래스 GUID와 연결된 클래스 이름을 검색합니다. 클래스는 로컬 또는 원격 컴퓨터에 설치할 수 있습니다. (유니코드)
SetupDiClassNameFromGuidW SetupDiClassNameFromGuid 함수는 클래스 GUID와 연결된 클래스 이름을 검색합니다. (유니코드)
SetupDiCreateDeviceInfoA SetupDiCreateDeviceInfo 함수는 새 디바이스 정보 요소를 만들고 지정된 디바이스 정보 집합에 새 멤버로 추가합니다. (ANSI)
SetupDiCreateDeviceInfoList SetupDiCreateDeviceInfoList 함수는 빈 디바이스 정보 집합을 만들고 필요에 따라 집합을 디바이스 설정 클래스 및 최상위 창에 연결합니다.
setupDiCreateDeviceInfoListExA SetupDiCreateDeviceInfoList 함수는 원격 또는 로컬 컴퓨터에 설정된 빈 디바이스 정보를 만들고 필요에 따라 집합을 디바이스 설정 클래스와 연결합니다. (ANSI)
SetupDiCreateDeviceInfoListExW SetupDiCreateDeviceInfoList 함수는 원격 또는 로컬 컴퓨터에 설정된 빈 디바이스 정보를 만들고 필요에 따라 집합을 디바이스 설정 클래스와 연결합니다. (유니코드)
SetupDiCreateDeviceInfoW

SetupDiCreateDeviceInfo 함수는 새 디바이스 정보 요소를 만들고 지정된 디바이스 정보 집합에 새 멤버로 추가합니다. (유니코드)
SetupDiCreateDeviceInterfaceA SetupDiCreateDeviceInterface 함수는 로컬 시스템 또는 원격 시스템에 디바이스 인터페이스를 등록합니다. (ANSI)
SetupDiCreateDeviceInterfaceRegKeyA SetupDiCreateDeviceInterfaceRegKey 함수는 디바이스 인터페이스에 대한 정보를 저장하기 위한 레지스트리 키를 만들고 키에 대한 핸들을 반환합니다. (ANSI)
SetupDiCreateDeviceInterfaceRegKeyW SetupDiCreateDeviceInterfaceRegKey 함수는 디바이스 인터페이스에 대한 정보를 저장하기 위한 레지스트리 키를 만들고 키에 대한 핸들을 반환합니다. (유니코드)
SetupDiCreateDeviceInterfaceW SetupDiCreateDeviceInterface 함수는 로컬 시스템 또는 원격 시스템에 디바이스 인터페이스를 등록합니다. (유니코드)
SetupDiCreateDevRegKeyA SetupDiCreateDevRegKey 함수는 디바이스별 구성 정보에 대한 레지스트리 키를 만들고 키에 대한 핸들을 반환합니다. (ANSI)
SetupDiCreateDevRegKeyW SetupDiCreateDevRegKey 함수는 디바이스별 구성 정보에 대한 레지스트리 키를 만들고 키에 대한 핸들을 반환합니다. (유니코드)
SetupDiDeleteDeviceInfo SetupDiDeleteDeviceInfo 함수는 디바이스 정보 집합에서 디바이스 정보 요소를 삭제합니다. 이 함수는 실제 디바이스를 삭제하지 않습니다.
SetupDiDeleteDeviceInterfaceData SetupDiDeleteDeviceInterfaceData 함수는 디바이스 정보 집합에서 디바이스 인터페이스를 삭제합니다.
SetupDiDeleteDeviceInterfaceRegKey SetupDiDeleteDeviceInterfaceRegKey 함수는 애플리케이션 및 드라이버에서 인터페이스 관련 정보를 저장하는 데 사용하는 레지스트리 하위 키를 삭제합니다.
SetupDiDeleteDevRegKey

SetupDiDeleteDevRegKey 함수는 디바이스 정보 요소와 연결된 지정된 사용자 액세스 가능 레지스트리 키를 삭제합니다.
setupDiDestroyClassImageList SetupDiDestroyClassImageList 함수는 SetupDiGetClassImageList 또는 SetupDiGetClassImageListEx를 호출하여 빌드된 클래스 이미지 목록을 삭제합니다.
SetupDiDestroyDeviceInfoList SetupDiDestroyDeviceInfoList 함수는 디바이스 정보 집합을 삭제하고 연결된 모든 메모리를 해제합니다.
SetupDiDestroyDriverInfoList SetupDiDestroyDriverInfoList 함수는 드라이버 목록을 삭제합니다.
SetupDiDrawMiniIcon SetupDiDrawMiniIcon 함수는 요청된 위치에 지정된 미니 아이콘을 그립니다.
SetupDiEnumDeviceInfo SetupDiEnumDeviceInfo 함수는 디바이스 정보 집합의 디바이스 정보 요소를 지정하는 SP_DEVINFO_DATA 구조를 반환합니다.
SetupDiEnumDeviceInterfaces SetupDiEnumDeviceInterfaces 함수는 디바이스 정보 집합에 포함된 디바이스 인터페이스를 열거합니다.
SetupDiEnumDriverInfoA SetupDiEnumDriverInfo 함수는 드라이버 목록의 멤버를 열거합니다. (ANSI)
SetupDiEnumDriverInfoW SetupDiEnumDriverInfo 함수는 드라이버 목록의 멤버를 열거합니다. (유니코드)
SetupDiGetActualModelsSectionA SetupDiGetActualModelsSection 함수는 디바이스 INF 파일에서 디바이스를 설치할 때 사용할 적절한 데코레이팅된 INF 모델 섹션을 검색합니다. (ANSI)
SetupDiGetActualModelsSectionW SetupDiGetActualModelsSection 함수는 디바이스 INF 파일에서 디바이스를 설치할 때 사용할 적절한 데코레이팅된 INF 모델 섹션을 검색합니다. (유니코드)

SetupDiGetActualSectionToInstallA SetupDiGetActualSectionToInstall 함수는 로컬 컴퓨터의 디바이스 INF 파일에서 디바이스를 설치할 때 사용할 적절한 INF DDInstall 섹션을 검색합니다. (ANSI)
SetupDiGetActualSectionToInstallExA SetupDiGetActualSectionToInstallEx 함수는 지정된 운영 체제 및 프로세서 아키텍처에 대한 디바이스를 설치하는 INF DDInstall 섹션의 이름을 검색합니다. (ANSI)
SetupDiGetActualSectionToInstallExW SetupDiGetActualSectionToInstallEx 함수는 지정된 운영 체제 및 프로세서 아키텍처에 대한 디바이스를 설치하는 INF DDInstall 섹션의 이름을 검색합니다. (유니코드)
SetupDiGetActualSectionToInstallW SetupDiGetActualSectionToInstall 함수는 로컬 컴퓨터의 디바이스 INF 파일에서 디바이스를 설치할 때 사용할 적절한 INF DDInstall 섹션을 검색합니다. (유니코드)
SetupDiGetClassBitmapIndex SetupDiGetClassBitmapIndex 함수는 지정된 클래스에 대해 제공된 미니 아이콘의 인덱스를 검색합니다.
SetupDiGetClassDescriptionA SetupDiGetClassDescription 함수는 지정된 설치 클래스 GUID와 연결된 클래스 설명을 검색합니다. (ANSI)
SetupDiGetClassDescriptionExA SetupDiGetClassDescriptionEx 함수는 로컬 또는 원격 컴퓨터에 설치된 설치 클래스에 대한 설명을 검색합니다. (ANSI)
SetupDiGetClassDescriptionExW SetupDiGetClassDescriptionEx 함수는 로컬 또는 원격 컴퓨터에 설치된 설치 클래스에 대한 설명을 검색합니다. (유니코드)
SetupDiGetClassDescriptionW SetupDiGetClassDescription 함수는 지정된 설치 클래스 GUID와 연결된 클래스 설명을 검색합니다. (유니코드)
SetupDiGetClassDevPropertySheetsA SetupDiGetClassDevPropertySheets 함수는 디바이스 정보 요소의 속성 시트 또는 디바이스 정보 집합의 디바이스 설정 클래스에 대한 핸들을 검색합니다. (ANSI)

SetupDiGetClassDevPropertySheetsW SetupDiGetClassDevPropertySheets 함수는 디바이스 정보 요소의 속성 시트 또는 디바이스 정보 집합의 디바이스 설정 클래스에 대한 핸들을 검색합니다. (유니코드)
SetupDiGetClassDevsExA SetupDiGetClassDevsEx 함수는 로컬 또는 원격 컴퓨터에 대해 요청된 디바이스 정보 요소가 포함된 디바이스 정보 집합에 대한 핸들을 반환합니다. (ANSI)
SetupDiGetClassDevsExW SetupDiGetClassDevsEx 함수는 로컬 또는 원격 컴퓨터에 대해 요청된 디바이스 정보 요소가 포함된 디바이스 정보 집합에 대한 핸들을 반환합니다. (유니코드)
setupDiGetClassDevsW SetupDiGetClassDevs 함수는 로컬 컴퓨터에 대해 요청된 디바이스 정보 요소가 포함된 디바이스 정보 집합에 대한 핸들을 반환합니다. (유니코드)
setupDiGetClassImageIndex SetupDiGetClassImageIndex 함수는 지정된 클래스의 클래스 이미지 목록 내에서 인덱스를 검색합니다.
SetupDiGetClassImageList SetupDiGetClassImageList 함수는 설치된 모든 클래스에 대한 비트맵을 포함하는 이미지 목록을 빌드하고 데이터 구조의 목록을 반환합니다.
SetupDiGetClassImageListExA SetupDiGetClassImageListEx 함수는 로컬 또는 원격 시스템에 설치된 모든 클래스에 대한 비트맵의 이미지 목록을 빌드합니다. (ANSI)
SetupDiGetClassImageListExW SetupDiGetClassImageListEx 함수는 로컬 또는 원격 시스템에 설치된 모든 클래스에 대한 비트맵의 이미지 목록을 빌드합니다. (유니코드)
SetupDiGetClassInstallParamsA SetupDiGetClassInstallParams 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대한 클래스 설치 매개 변수를 검색합니다. (ANSI)
SetupDiGetClassInstallParamsW SetupDiGetClassInstallParams 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대한 클래스 설치 매개 변수를 검색합니다. (유니코드)

setupDiGetClassPropertyExW SetupDiGetClassPropertyEx 함수는 로컬 또는 원격 컴퓨터의 디바이스 설정 클래스 또는 디바이스 인터페이스 클래스에 대한 클래스 속성을 검색합니다.
setupDiGetClassPropertyKeys SetupDiGetClassPropertyKeys 함수는 디바이스 설정 클래스 또는 디바이스 인터페이스 클래스에 대해 설정된 디바이스 속성을 나타내는 디바이스 속성 키의 배열을 검색합니다.
SetupDiGetClassPropertyKeysExW SetupDiGetClassPropertyKeysEx 함수는 로컬 또는 원격 컴퓨터에서 디바이스 설정 클래스 또는 디바이스 인터페이스 클래스에 대해 설정된 디바이스 속성을 나타내는 디바이스 속성 키의 배열을 검색합니다.
SetupDiGetClassPropertyW SetupDiGetClassProperty 함수는 디바이스 설정 클래스 또는 디바이스 인터페이스 클래스에 대해 설정된 디바이스 속성을 검색합니다.
SetupDiGetClassRegistryPropertyA SetupDiGetClassRegistryProperty 함수는 레지스트리에서 지정된 디바이스 설정 클래스에 대한 속성을 검색합니다. (ANSI)
SetupDiGetClassRegistryPropertyW SetupDiGetClassRegistryProperty 함수는 레지스트리에서 지정된 디바이스 설정 클래스에 대한 속성을 검색합니다. (유니코드)
SetupDiGetCustomDevicePropertyA SetupDiGetCustomDeviceProperty 함수는 레지스트리에서 지정된 사용자 지정 디바이스 속성을 검색합니다. (ANSI)
SetupDiGetCustomDevicePropertyW SetupDiGetCustomDeviceProperty 함수는 레지스트리에서 지정된 사용자 지정 디바이스 속성을 검색합니다. (유니코드)
setupDiGetDeviceInfoListClass SetDiGetDeviceInfoListClass 함수는 집합에 연결된 클래스가 있는 경우 디바이스 정보 집합과 연결된 디바이스 설정 클래스의 GUID를 검색합니다.
SetupDiGetDeviceInfoListDetailA

SetupDiGetDeviceInfoListDetail 함수는 클래스 GUID, 원격 컴퓨터 핸들 및 원격 컴퓨터 이름을 포함하여 디바이스 정보 집합과 관련된 정보를 검색합니다. (ANSI)
setupDiGetDeviceInfoListDetailW SetupDiGetDeviceInfoListDetail 함수는 클래스 GUID, 원격 컴퓨터 핸들 및 원격 컴퓨터 이름을 포함하여 디바이스 정보 집합과 관련된 정보를 검색합니다. (유니코드)
SetupDiGetDeviceInstallParamsA SetupDiGetDeviceInstallParams 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대한 디바이스 설치 매개 변수를 검색합니다. (ANSI)
SetupDiGetDeviceInstallParamsW SetupDiGetDeviceInstallParams 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대한 디바이스 설치 매개 변수를 검색합니다. (유니코드)
SetupDiGetDeviceInstanceIdA SetupDiGetDeviceInstanceId 함수는 디바이스 정보 요소와 연결된 디바이스 인스턴스 ID를 검색합니다. (ANSI)
SetupDiGetDeviceInstanceIdW SetupDiGetDeviceInstanceId 함수는 디바이스 정보 요소와 연결된 디바이스 인스턴스 ID를 검색합니다. (유니코드)
SetupDiGetDeviceInterfaceAlias SetupDiGetDeviceInterfaceAlias 함수는 지정된 디바이스 인터페이스의 별칭을 반환합니다.
SetupDiGetDeviceInterfaceDetailA SetupDiGetDeviceInterfaceDetail 함수는 디바이스 인터페이스에 대한 세부 정보를 반환합니다. (ANSI)
SetupDiGetDeviceInterfaceDetailW SetupDiGetDeviceInterfaceDetail 함수는 디바이스 인터페이스에 대한 세부 정보를 반환합니다. (유니코드)
setupDiGetDeviceInterfacePropertyKeys SetupDiGetDeviceInterfacePropertyKeys 함수는 디바이스 인터페이스에 대해 설정된 디바이스 속성을 나타내는 디바이스 속성 키 배열을 검색합니다.
SetupDiGetDeviceInterfacePropertyW

SetupDiGetDeviceInterfaceProperty 함수는 디바이스 인터페이스에 대해 설정된 디바이스 속성을 검색합니다.
SetupDiGetDevicePropertyKeys SetupDiGetDevicePropertyKeys 함수는 디바이스 인스턴스에 대해 설정된 디바이스 속성을 나타내는 디바이스 속성 키의 배열을 검색합니다.
setupDiGetDevicePropertyW SetupDiGetDeviceProperty 함수는 디바이스 인스턴스 속성을 검색합니다.
SetupDiGetDeviceRegistryPropertyA SetupDiGetDeviceRegistryProperty 함수는 지정된 플러그 앤 플레이 디바이스 속성을 검색합니다. (ANSI)
SetupDiGetDeviceRegistryPropertyW SetupDiGetDeviceRegistryProperty 함수는 지정된 플러그 앤 플레이 디바이스 속성을 검색합니다. (유니코드)
SetupDiGetDriverInfoDetailA SetupDiGetDriverInfoDetail 함수는 디바이스 정보 집합 또는 디바이스 정보 집합의 특정 디바이스 정보 요소에 대한 드라이버 정보 세부 정보를 검색합니다. (ANSI)
SetupDiGetDriverInfoDetailW SetupDiGetDriverInfoDetail 함수는 디바이스 정보 집합 또는 디바이스 정보 집합의 특정 디바이스 정보 요소에 대한 드라이버 정보 세부 정보를 검색합니다. (유니코드)
SetupDiGetDriverInstallParamsA SetupDiGetDriverInstallParams 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대한 드라이버 설치 매개 변수를 검색합니다. (ANSI)
SetupDiGetDriverInstallParamsW SetupDiGetDriverInstallParams 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대한 드라이버 설치 매개 변수를 검색합니다. (유니코드)
SetupDiGetHwProfileFriendlyNameA SetupDiGetHwProfileFriendlyName 함수는 하드웨어 프로필 ID와 연결된 이름을 검색합니다. (ANSI)
SetupDiGetHwProfileFriendlyNameExA

SetupDiGetHwProfileFriendlyNameEx 함수는 로컬 또는 원격 컴퓨터의 하드웨어 프로필 ID와 연결된 이름을 검색합니다. (ANSI)
SetupDiGetHwProfileFriendlyNameExW SetupDiGetHwProfileFriendlyNameEx 함수는 로컬 또는 원격 컴퓨터의 하드웨어 프로필 ID와 연결된 이름을 검색합니다. (유니코드)
SetupDiGetHwProfileFriendlyNameW SetupDiGetHwProfileFriendlyName 함수는 하드웨어 프로필 ID와 연결된 이름을 검색합니다. (유니코드)
SetupDiGetHwProfileList SetupDiGetHwProfileList 함수는 현재 정의된 모든 하드웨어 프로필 ID의 목록을 검색합니다.
setupDiGetHwProfileListExA SetupDiGetHwProfileListEx 함수는 로컬 또는 원격 컴퓨터에서 현재 정의된 모든 하드웨어 프로필 ID 목록을 검색합니다. (ANSI)
setupDiGetHwProfileListExW SetupDiGetHwProfileListEx 함수는 로컬 또는 원격 컴퓨터에서 현재 정의된 모든 하드웨어 프로필 ID 목록을 검색합니다. (유니코드)
SetupDiGetINFClassA SetupDiGetINFClass 함수는 지정된 디바이스 INF 파일의 클래스를 반환합니다. (ANSI)
setupDiGetINFClassW SetupDiGetINFClass 함수는 지정된 디바이스 INF 파일의 클래스를 반환합니다. (유니코드)
SetupDiGetSelectedDevice SetupDiGetSelectedDevice 함수는 디바이스 정보 집합에서 선택한 디바이스 정보 요소를 검색합니다.
SetupDiGetSelectedDriverA SetupDiGetSelectedDriver 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대해 선택한 드라이버를 검색합니다. (ANSI)
SetupDiGetSelectedDriverW SetupDiGetSelectedDriver 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대해 선택한 드라이버를 검색합니다. (유니코드)

SetupDiInstallClassA SetupDiInstallClass 함수는 지정된 INF 파일의 ClassInstall32 섹션을 설치합니다. (ANSI)
SetupDiInstallClassExA SetupDiInstallClassEx 함수는 클래스 설치 관리자 또는 인터페이스 클래스를 설치합니다. (ANSI)
SetupDiInstallClassExW SetupDiInstallClassEx 함수는 클래스 설치 관리자 또는 인터페이스 클래스를 설치합니다. (유니코드)
SetupDiInstallClassW SetupDiInstallClass 함수는 지정된 INF 파일의 ClassInstall32 섹션을 설치합니다. (유니코드)
SetupDiInstallDevice SetupDiInstallDevice 함수는 DIF_INSTALLDEVICE 설치 요청에 대한 기본 처리기입니다.
SetupDiInstallDeviceInterfaces SetupDiInstallDeviceInterfaces 함수는 DIF_INSTALLINTERFACES 설치 요청에 대한 기본 처리기입니다.
SetupDiInstallDriverFiles SetupDiInstallDriverFiles 함수는 DIF_INSTALLDEVICEFILES 설치 요청에 대한 기본 처리기입니다.
setupDiLoadClassIcon SetupDiLoadClassIcon 함수는 지정된 클래스의 큰 아이콘과 미니 아이콘을 모두 로드합니다.
SetupDiLoadDeviceIcon SetupDiLoadDeviceIcon 함수는 지정된 디바이스에 대한 아이콘을 검색합니다.
SetupDiOpenClassRegKey SetupDiOpenClassRegKey 함수는 설치 클래스 레지스트리 키 또는 특정 클래스의 하위 키를 엽니다.
SetupDiOpenClassRegKeyExA SetupDiOpenClassRegKeyEx 함수는 디바이스 설정 클래스 레지스트리 키, 디바이스 인터페이스 클래스 레지스트리 키 또는 특정 클래스의 하위 키를 엽니다. 이 함수는 로컬 컴퓨터 또는 원격 컴퓨터에서 지정된 키를 엽니다. (ANSI)

SetupDiOpenClassRegKeyExW SetupDiOpenClassRegKeyEx 함수는 디바이스 설정 클래스 레지스트리 키, 디바이스 인터페이스 클래스 레지스트리 키 또는 특정 클래스의 하위 키를 엽니다. 이 함수는 로컬 컴퓨터 또는 원격 컴퓨터에서 지정된 키를 엽니다. (유니코드)
SetupDiOpenDeviceInfoA SetupDiOpenDeviceInfo 함수는 디바이스 정보 집합에 아직 없는 경우 디바이스 인스턴스에 대한 디바이스 정보 요소를 디바이스 정보 집합에 추가하고 디바이스 정보 집합에서 디바이스 인스턴스의 디바이스 정보 요소를 식별하는 정보를 검색합니다. (ANSI)
SetupDiOpenDeviceInfoW SetupDiOpenDeviceInfo 함수는 디바이스 정보 집합에 아직 없는 경우 디바이스 인스턴스에 대한 디바이스 정보 요소를 디바이스 정보 집합에 추가하고 디바이스 정보 집합에서 디바이스 인스턴스의 디바이스 정보 요소를 식별하는 정보를 검색합니다. (유니코드)
SetupDiOpenDeviceInterfaceA SetupDiOpenDeviceInterface 함수는 디바이스 인터페이스에 대한 정보를 검색하고 로컬 시스템 또는 원격 시스템에 대해 설정된 지정된 디바이스 정보에 인터페이스를 추가합니다. (ANSI)
setupDiOpenDeviceInterfaceRegKey SetupDiOpenDeviceInterfaceRegKey 함수는 애플리케이션 및 드라이버가 디바이스 인터페이스와 관련된 정보를 저장하는 데 사용하는 레지스트리 하위 키를 엽니다.
SetupDiOpenDeviceInterfaceW SetupDiOpenDeviceInterface 함수는 디바이스 인터페이스에 대한 정보를 검색하고 로컬 시스템 또는 원격 시스템에 대해 설정된 지정된 디바이스 정보에 인터페이스를 추가합니다. (유니코드)
SetupDiOpenDevRegKey SetupDiOpenDevRegKey 함수는 디바이스별 구성 정보에 대한 레지스트리 키를 엽니다.
SetupDiRegisterCoDeviceInstallers SetupDiRegisterCoDeviceInstallers 함수는 DIF_REGISTER_COINSTALLERS 기본 처리기입니다.
SetupDiRegisterDeviceInfo SetupDiRegisterDeviceInfo 함수는 DIF_REGISTERDEVICE 요청에 대한 기본 처리기입니다.
SetupDiRemoveDevice SetupDiRemoveDevice 함수는 DIF_REMOVE 설치 요청에 대한 기본 처리기입니다.

SetupDiRemoveDeviceInterface SetupDiRemoveDeviceInterface 함수는 등록된 디바이스 인터페이스를 시스템에서 제거합니다.
SetupDiRestartDevices SetupDiRestartDevices 함수는 지정된 디바이스를 다시 시작하거나 필요한 경우 동일한 함수에서 작동하는 모든 디바이스를 다시 시작하고 지정된 디바이스를 작동하는 드라이버를 필터링합니다.
SetupDiSelectBestCompatDrv SetupDiSelectBestCompatDrv 함수는 DIF_SELECTBESTCOMPATDRV 설치 요청에 대한 기본 처리기입니다.
SetupDiSelectDevice SetupDiSelectDevice 함수는 DIF_SELECTDEVICE 요청에 대한 기본 처리기입니다.
SetupDiSelectOEMDrv SetupDiSelectOEMDrv 함수는 디바이스 정보 집합에 대한 드라이버 또는 사용자가 제공한 OEM 경로를 사용하는 특정 디바이스 정보 요소를 선택합니다.
SetupDiSetClassInstallParamsA SetupDiSetClassInstallParams 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대한 클래스 설치 매개 변수를 설정하거나 지웁니다. (ANSI)
SetupDiSetClassInstallParamsW SetupDiSetClassInstallParams 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대한 클래스 설치 매개 변수를 설정하거나 지웁니다. (유니코드)
SetupDiSetClassPropertyExW SetupDiSetClassPropertyEx 함수는 로컬 또는 원격 컴퓨터의 디바이스 설정 클래스 또는 디바이스 인터페이스 클래스에 대한 디바이스 속성을 설정합니다.
SetupDiSetClassPropertyW SetupDiSetClassProperty 함수는 디바이스 설정 클래스 또는 디바이스 인터페이스 클래스에 대한 클래스 속성을 설정합니다.
setupDiSetClassRegistryPropertyA SetupDiSetClassRegistryProperty 함수는 레지스트리에서 지정된 디바이스 클래스 속성을 설정합니다. (ANSI)

SetupDiSetClassRegistryPropertyW SetupDiSetClassRegistryProperty 함수는 레지스트리에서 지정된 디바이스 클래스 속성을 설정합니다. (유니코드)
SetupDiSetDeviceInstallParamsA SetupDiSetDeviceInstallParams 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대한 디바이스 설치 매개 변수를 설정합니다. (ANSI)
setupDiSetDeviceInstallParamsW SetupDiSetDeviceInstallParams 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대한 디바이스 설치 매개 변수를 설정합니다. (유니코드)
SetupDiSetDeviceInterfaceDefault SetupDiSetDeviceInterfaceDefault 함수는 디바이스 인터페이스를 디바이스 인터페이스 클래스의 기본 인터페이스로 설정합니다.
SetupDiSetDeviceInterfacePropertyW SetupDiSetDeviceInterfaceProperty 함수는 디바이스 인터페이스의 디바이스 속성을 설정합니다.
setupDiSetDevicePropertyW SetupDiSetDeviceProperty 함수는 디바이스 인스턴스 속성을 설정합니다.
SetupDiSetDeviceRegistryPropertyA SetupDiSetDeviceRegistryProperty 함수는 디바이스에 대한 플러그 앤 플레이 디바이스 속성을 설정합니다. (ANSI)
SetupDiSetDeviceRegistryPropertyW SetupDiSetDeviceRegistryProperty 함수는 디바이스에 대한 플러그 앤 플레이 디바이스 속성을 설정합니다. (유니코드)
SetupDiSetDriverInstallParamsA SetupDiSetDriverInstallParams 함수는 드라이버 정보 요소에 대한 드라이버 설치 매개 변수를 설정합니다. (ANSI)
SetupDiSetDriverInstallParamsW SetupDiSetDriverInstallParams 함수는 드라이버 정보 요소에 대한 드라이버 설치 매개 변수를 설정합니다. (유니코드)

SetupDiSetSelectedDevice SetupDiSetSelectedDevice 함수는 디바이스 정보 집합의 선택된 멤버로 디바이스 정보 요소를 설정합니다. 이 함수는 일반적으로 설치 마법사에서 사용됩니다.
SetupDiSetSelectedDriverA SetupDiSetSelectedDriver 함수는 디바이스 정보 요소에 대해 선택한 드라이버 또는 디바이스 정보 집합에 대해 선택한 클래스 드라이버를 설정하거나 다시 설정합니다. (ANSI)
SetupDiSetSelectedDriverW SetupDiSetSelectedDriver 함수는 디바이스 정보 요소에 대해 선택한 드라이버 또는 디바이스 정보 집합에 대해 선택한 클래스 드라이버를 설정하거나 다시 설정합니다. (유니코드)
SetupDiUnremoveDevice SetupDiUnremoveDevice 함수는 DIF_UNREMOVE 설치 요청에 대한 기본 처리기입니다.
SetupGetInfDriverStoreLocationA SetupGetInfDriverStoreLocation 함수는 드라이버 저장소에서 시스템 INF 파일 디렉터리의 지정된 INF 파일 또는 드라이버 저장소의 지정된 INF 파일에 해당하는 INF 파일의 정규화된 파일 이름(디렉터리 경로 및 파일 이름)을 검색합니다. (ANSI)
SetupGetInfDriverStoreLocationW SetupGetInfDriverStoreLocation 함수는 드라이버 저장소에서 시스템 INF 파일 디렉터리의 지정된 INF 파일 또는 드라이버 저장소의 지정된 INF 파일에 해당하는 INF 파일의 정규화된 파일 이름(디렉터리 경로 및 파일 이름)을 검색합니다. (유니코드)
SetupGetInfPublishedNameA SetupGetInfPublishedName 함수는 시스템 INF 파일 디렉터리에서 드라이버 저장소의 지정된 INF 파일 또는 시스템 INF 파일 디렉터리의 지정된 INF 파일에 해당하는 INF 파일의 정규화된 파일 이름(디렉터리 경로 및 파일 이름)을 검색합니다. (ANSI)
SetupGetInfPublishedNameW SetupGetInfPublishedName 함수는 시스템 INF 파일 디렉터리에서 드라이버 저장소의 지정된 INF 파일 또는 시스템 INF 파일 디렉터리의 지정된 INF 파일에 해당하는 INF 파일의 정규화된 파일 이름(디렉터리 경로 및 파일 이름)을 검색합니다. (유니코드)
setupGetNonInteractiveMode SetupGetNonInteractiveMode 함수는 호출자의 프로세스가 대화 상자와 같은 사용자 인터페이스 구성 요소를 통해 사용자와 상호 작용할 수 있는지 여부를 나타내는 SetupAPI 비대화형 플래그 값을 반환합니다.

SetupGetThreadLogToken SetupGetThreadLogToken 함수는 이 함수가 호출된 스레드에 대한 로그 토큰을 검색합니다.
SetupSetNonInteractiveMode SetupSetNonInteractiveMode 함수는 SetupAPI가 호출자의 컨텍스트에서 사용자와 상호 작용할 수 있는지 여부를 결정하는 비대화형 SetupAPI 플래그를 설정합니다.
setupSetThreadLogToken SetupSetThreadLogToken 함수는 이 함수가 호출된 스레드에 대해 로그 토큰으로 표시되는 로그 컨텍스트를 설정합니다.
SetupWriteTextLog SetupWriteTextLog 함수는 SetupAPI 텍스트 로그에 로그 항목을 씁니다.
SetupWriteTextLogError SetupWriteTextLogError 함수는 SetupAPI 관련 오류 또는 Win32 시스템 오류에 대한 정보를 SetupAPI 텍스트 로그에 씁니다.
SetupWriteTextLogInfLine SetupWriteTextLogInfLine 함수는 지정된 INF 파일 줄의 텍스트가 포함된 SetupAPI 텍스트 로그에 로그 항목을 씁니다.
UpdateDriverForPlugAndPlayDevicesA INF 파일 및 하드웨어 ID가 지정된 UpdateDriverForPlugAndPlayDevices 함수는 하드웨어 ID와 일치하는 디바이스에 대해 업데이트된 드라이버를 설치합니다. (ANSI)
UpdateDriverForPlugAndPlayDevicesW INF 파일 및 하드웨어 ID가 지정된 UpdateDriverForPlugAndPlayDevices 함수는 하드웨어 ID와 일치하는 디바이스에 대해 업데이트된 드라이버를 설치합니다. (유니코드)

구조

테이블 확장

BUSNUMBER_DES

BUSNUMBER_DES 구조는 디바이스 인스턴스의 버스 번호 사용량을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다.
BUSNUMBER_RANGE BUSNUMBER_RANGE 구조는 디바이스 인스턴스의 버스 번호 사용량을 설명하는 리소스 요구 사항 목록을 지정합니다. 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.
BUSNUMBER_RESOURCE BUSNUMBER_RESOURCE 구조는 디바이스 인스턴스의 버스 번호 사용량을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정합니다. 리소스 목록 및 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.
CM_NOTIFY_EVENT_DATA 디바이스 알림 이벤트 데이터 구조입니다.
CM_NOTIFY_FILTER 디바이스 알림 필터 구조입니다.
CONFLICT_DETAILS_A CONFLICT_DETAILS 구조체는 CM_Get_Resource_Conflict_Details 함수에 대한 매개 변수로 사용됩니다. (ANSI)
CONFLICT_DETAILS_W CONFLICT_DETAILS 구조체는 CM_Get_Resource_Conflict_Details 함수에 대한 매개 변수로 사용됩니다. (유니코드)
CS_DES CS_DES 구조는 디바이스 인스턴스에 대한 디바이스 클래스별 리소스 사용을 설명하는 리소스 목록을 지정하는 데 사용됩니다. 리소스 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.
CS_RESOURCE CS_RESOURCE 구조는 디바이스 인스턴스에 대한 디바이스 클래스별 리소스 사용을 설명하는 리소스 목록을 지정하는 데 사용됩니다. 리소스 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.
DEV_OBJECT 디바이스 개체를 나타내는 정보를 포함합니다.
DEV_QUERY_RESULT_ACTION_DATA

PDEV_QUERY_RESULT_CALLBACK 콜백 함수에 정보를 제공합니다.
DEVPROP_FILTER_EXPRESSION 필터 결과를 쿼리하는 데 사용되는 필터 식을 설명합니다.
DMA_DES DMA_DES 구조는 디바이스 인스턴스에 대한 DMA(직접 메모리 액세스) 채널 사용을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다.
DMA_RANGE DMA_RANGE 구조는 디바이스 인스턴스에 대한 DMA 채널 사용을 설명하는 리소스 요구 사항 목록을 지정합니다. 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.
DMA_RESOURCE DMA_RESOURCE 구조는 디바이스 인스턴스에 대한 DMA 채널 사용량을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다.
IO_DES IO_DES 구조는 디바이스 인스턴스의 I/O 포트 사용량을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다. 리소스 목록 및 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.
IO_RANGE IO_RANGE 구조는 디바이스 인스턴스의 I/O 포트 사용을 설명하는 리소스 요구 사항 목록을 지정합니다. 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.
IO_RESOURCE IO_RESOURCE 구조는 디바이스 인스턴스의 I/O 포트 사용량을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다.
IRQ_DES_32 IRQ_DES 구조는 디바이스 인스턴스에 대한 IRQ 줄 사용을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다. 리소스 목록 및 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요. (32비트)
IRQ_DES_64 IRQ_DES 구조는 디바이스 인스턴스에 대한 IRQ 줄 사용을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다. 리소스 목록 및 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요. (64비트)

IRQ_RANGE IRQ_RANGE 구조는 디바이스 인스턴스에 대한 IRQ 줄 사용을 설명하는 리소스 요구 사항 목록을 지정합니다. 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.
IRQ_RESOURCE_32 IRQ_RESOURCE 구조는 디바이스 인스턴스의 IRQ 줄 사용을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다. (32비트)
IRQ_RESOURCE_64 IRQ_RESOURCE 구조는 디바이스 인스턴스의 IRQ 줄 사용을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다. (64비트)
MEM_DES MEM_DES 구조는 디바이스 인스턴스의 메모리 사용량을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다. 리소스 목록 및 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.
MEM_RANGE MEM_RANGE 구조는 디바이스 인스턴스의 메모리 사용량을 설명하는 리소스 요구 사항 목록을 지정합니다. 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.
MEM_RESOURCE MEM_RESOURCE 구조는 디바이스 인스턴스의 메모리 사용량을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다. 리소스 목록 및 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.
MFCARD_DES MFCARD_DES 구조는 다기능 디바이스 인스턴스에서 제공하는 하드웨어 함수 중 하나에서 리소스 사용량을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다.
MFCARD_RESOURCE MFCARD_RESOURCE 구조는 다기능 디바이스 인스턴스에서 제공하는 하드웨어 함수 중 하나에서 리소스 사용량을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다.
PCCARD_DES PCCARD_DES 구조는 PC 카드 인스턴스의 리소스 사용량을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다. 리소스 목록 및 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.

PCCARD_RESOURCE PCCARD_RESOURCE 구조는 PC 카드 인스턴스의 리소스 사용량을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다.
SP_CLASSIMAGELIST_DATA SP_CLASSIMAGELIST_DATA 구조체는 클래스 이미지 목록을 설명합니다.
SP_CLASSINSTALL_HEADER SP_CLASSINSTALL_HEADER 클래스 설치 매개 변수 구조의 첫 번째 멤버입니다. 여기에는 나머지 설치 매개 변수 구조의 형식을 정의하는 디바이스 설치 요청 코드가 포함되어 있습니다.
SP_DETECTDEVICE_PARAMS SP_DETECTDEVICE_PARAMS 구조는 DIF_DETECT 설치 요청에 해당합니다.
SP_DEVICE_INTERFACE_DATA SP_DEVICE_INTERFACE_DATA 구조는 디바이스 정보 집합에서 디바이스 인터페이스를 정의합니다.
SP_DEVICE_INTERFACE_DETAIL_DATA_A SP_DEVICE_INTERFACE_DETAIL_DATA 구조체에는 디바이스 인터페이스에 대한 경로가 포함됩니다. (ANSI)
SP_DEVICE_INTERFACE_DETAIL_DATA_W SP_DEVICE_INTERFACE_DETAIL_DATA 구조체에는 디바이스 인터페이스에 대한 경로가 포함됩니다. (유니코드)
SP_DEVINFO_DATA SP_DEVINFO_DATA 구조체는 디바이스 정보 집합의 멤버인 디바이스 인스턴스를 정의합니다.
SP_DEVINFO_LIST_DETAIL_DATA_A SP_DEVINFO_LIST_DETAIL_DATA 구조에는 연결된 설치 클래스 GUID(연결된 설치 클래스가 있는 경우)와 같은 디바이스 정보 집합에 대한 정보가 포함됩니다. (ANSI)
SP_DEVINFO_LIST_DETAIL_DATA_W SP_DEVINFO_LIST_DETAIL_DATA 구조에는 연결된 설치 클래스 GUID(연결된 설치 클래스가 있는 경우)와 같은 디바이스 정보 집합에 대한 정보가 포함됩니다. (유니코드)
SP_DEVINSTALL_PARAMS_A

SP_DEVINSTALL_PARAMS 구조에는 특정 디바이스 정보 요소와 연결되거나 디바이스 정보 집합과 전역적으로 연결된 디바이스 설치 매개 변수가 포함됩니다. (ANSI)
SP_DEVINSTALL_PARAMS_W SP_DEVINSTALL_PARAMS 구조에는 특정 디바이스 정보 요소와 연결되거나 디바이스 정보 집합과 전역적으로 연결된 디바이스 설치 매개 변수가 포함됩니다. (유니코드)
SP_DRVINFO_DATA_V1_A SP_DRVINFO_DATA 구조체에는 드라이버에 대한 정보가 포함됩니다. 이 구조는 특정 디바이스 인스턴스 또는 전역적으로 디바이스 정보 집합과 연결할 수 있는 드라이버 정보 목록의 멤버입니다. (sp_drvinfo_data_v1_a)
SP_DRVINFO_DATA_V1_W SP_DRVINFO_DATA 구조체에는 드라이버에 대한 정보가 포함됩니다. 이 구조는 특정 디바이스 인스턴스 또는 전역적으로 디바이스 정보 집합과 연결할 수 있는 드라이버 정보 목록의 멤버입니다. (sp_drvinfo_data_v1_w)
SP_DRVINFO_DATA_V2_A SP_DRVINFO_DATA 구조체에는 드라이버에 대한 정보가 포함됩니다. 이 구조는 특정 디바이스 인스턴스 또는 전역적으로 디바이스 정보 집합과 연결할 수 있는 드라이버 정보 목록의 멤버입니다. (sp_drvinfo_data_v2_a)
SP_DRVINFO_DATA_V2_W SP_DRVINFO_DATA 구조체에는 드라이버에 대한 정보가 포함됩니다. 이 구조는 특정 디바이스 인스턴스 또는 전역적으로 디바이스 정보 집합과 연결할 수 있는 드라이버 정보 목록의 멤버입니다. (sp_drvinfo_data_v2_w)
SP_DRVINFO_DETAIL_DATA_A SP_DRVINFO_DETAIL_DATA 구조에는 특정 드라이버 정보 구조에 대한 자세한 정보가 포함됩니다. (ANSI)
SP_DRVINFO_DETAIL_DATA_W SP_DRVINFO_DETAIL_DATA 구조에는 특정 드라이버 정보 구조에 대한 자세한 정보가 포함됩니다. (유니코드)
SP_DRVINSTALL_PARAMS SP_DRVINSTALL_PARAMS 구조에는 특정 드라이버 정보 요소와 연결된 드라이버 설치 매개 변수가 포함됩니다.
SP_NEWDEVICEWIZARD_DATA

SP_NEWDEVICEWIZARD_DATA 구조는 설치 관리자가 사용자 지정 페이지를 추가하여 하드웨어 설치 마법사의 작업을 확장하는 데 사용됩니다. DIF_NEWDEVICEWIZARD_XXX 설치 요청과 함께 사용됩니다.
SP_POWERMESSAGEWAKE_PARAMS_A SP_POWERMESSAGEWAKE_PARAMS 구조는 DIF_POWERMESSAGEWAKE 설치 요청에 해당합니다. (ANSI)
SP_POWERMESSAGEWAKE_PARAMS_W SP_POWERMESSAGEWAKE_PARAMS 구조는 DIF_POWERMESSAGEWAKE 설치 요청에 해당합니다. (유니코드)
SP_PROPCHANGE_PARAMS SP_PROPCHANGE_PARAMS 구조는 DIF_PROPERTYCHANGE 설치 요청에 해당합니다.
SP_PROPSHEETPAGE_REQUEST SP_PROPSHEETPAGE_REQUEST 구조체는 SetupAPI DLL의 ExtensionPropSheetPageProc 진입점에 첫 번째 매개 변수(lpv)로 전달될 수 있습니다.
SP_REMOVEDEVICE_PARAMS SP_REMOVEDEVICE_PARAMS 구조는 DIF_REMOVE 설치 요청에 해당합니다.
SP_SELECTDEVICE_PARAMS_A SP_SELECTDEVICE_PARAMS 구조는 DIF_SELECTDEVICE 설치 요청에 해당합니다. (ANSI)
SP_SELECTDEVICE_PARAMS_W SP_SELECTDEVICE_PARAMS 구조는 DIF_SELECTDEVICE 설치 요청에 해당합니다. (유니코드)
SP_TROUBLESHOOTER_PARAMS_A SP_TROUBLESHOOTER_PARAMS 구조는 DIF_TROUBLESHOOTER 설치 요청에 해당합니다. (ANSI)
SP_TROUBLESHOOTER_PARAMS_W SP_TROUBLESHOOTER_PARAMS 구조는 DIF_TROUBLESHOOTER 설치 요청에 해당합니다. (유니코드)
SP_UNREMOVEDEVICE_PARAMS SP_UNREMOVEDEVICE_PARAMS 구조는 DIF_UNREMOVE 설치 요청에 해당합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

대화 상자

아티클2023. 03. 14.

대화 상자 기술 개요입니다.

대화 상자를 개발하려면 다음 헤더가 필요합니다.

- commdlg.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 대화 상자

Functions

AddRef IPrintDialogCallback::AddRef 메서드(commdlg.h)는 COM 개체의 인터페이스에 대한 참조 수를 감소시키는 IUnknown의 AddRef 메서드입니다.
AddRef IPrintDialogServices::AddRef 메서드(commdlg.h)는 COM 개체의 인터페이스에 대한 참조 수를 감소시키는 IUnknown의 AddRef 메서드입니다.
CDSIZEOF_STRUCT 지정된 멤버까지 구조체의 크기를 가져옵니다.
ColorW 선택 ChooseColor의 와이드 문자열 버전
선택FontA ChooseFont의 ASCII 버전
ChooseFontW ChooseFont의 와이드 문자열 버전
CommDlg_OpenSave_GetFilePathA ASCII 버전의 CommDlg_OpenSave_GetFilePath

CommDlg_OpenSave_GetFilePathW CommDlg_OpenSave_GetFilePath 와이드 문자열 버전
CommDlg_OpenSave_GetFolderIDList 탐색기 스타일 열기 또는 다른 이름으로 저장 대화 상자가 현재 열려 있는 폴더에 해당하는 항목 식별자 목록의 주소를 검색합니다.
CommDlg_OpenSave_GetFolderPathA ASCII 버전의 CommDlg_OpenSave_GetFolderPath
CommDlg_OpenSave_GetFolderPathW CommDlg_OpenSave_GetFolderPath 와이드 문자열 버전
CommDlg_OpenSave_GetSpecA ASCII 버전의 CommDlg_OpenSave_GetSpec
CommDlg_OpenSave_GetSpecW CommDlg_OpenSave_GetSpec 와이드 문자열 버전
CommDlg_OpenSave_HideControl 탐색기 스타일 열기 또는 다른 이름으로 저장 대화 상자에서 지정된 컨트롤을 숨깁니다.
CommDlg_OpenSave_SetControlText 탐색기 스타일 열기 또는 다른 이름으로 저장 대화 상자에서 지정된 컨트롤의 텍스트를 설정합니다.
CommDlg_OpenSave_SetDefExt 탐색기 스타일 열기 또는 다른 이름으로 저장 대화 상자의 기본 파일 이름 확장명을 설정합니다.
CommDlgExtendedError CommDlgExtendedError 함수(commdlg.h)는 일반적인 대화 상자 오류 코드를 반환합니다.
CreateDialogA 대화 상자 템플릿 리소스에서 모덜리스 대화 상자를 만듭니다. CreateDialog 매크로는 CreateDialogParam 함수를 사용합니다. (ANSI)

CreateDialogIndirectA 메모리의 대화 상자 템플릿에서 모덜리스 대화 상자를 만듭니다. CreateDialogIndirect 매크로는 CreateDialogIndirectParam 함수를 사용합니다. (ANSI)
CreateDialogIndirectParamA 메모리의 대화 상자 템플릿에서 모덜리스 대화 상자를 만듭니다. (ANSI)
CreateDialogIndirectParamW 메모리의 대화 상자 템플릿에서 모덜리스 대화 상자를 만듭니다. (유니코드)
CreateDialogIndirectW 메모리의 대화 상자 템플릿에서 모덜리스 대화 상자를 만듭니다. CreateDialogIndirect 매크로는 CreateDialogIndirectParam 함수를 사용합니다. (유니코드)
CreateDialogParamA 대화 상자 템플릿 리소스에서 모덜리스 대화 상자를 만듭니다. (ANSI)
CreateDialogParamW 대화 상자 템플릿 리소스에서 모덜리스 대화 상자를 만듭니다. (유니코드)
CreateDialogW 대화 상자 템플릿 리소스에서 모덜리스 대화 상자를 만듭니다. CreateDialog 매크로는 CreateDialogParam 함수를 사용합니다. (유니코드)
DefDlgProcA 기본 대화 상자 창 프로시저를 호출하여 프라이빗 창 클래스가 있는 대화 상자가 처리하지 않는 모든 창 메시지에 대한 기본 처리를 제공합니다. (ANSI)
DefDlgProcW 기본 대화 상자 창 프로시저를 호출하여 프라이빗 창 클래스가 있는 대화 상자가 처리하지 않는 모든 창 메시지에 대한 기본 처리를 제공합니다. (유니코드)
DialogBoxA 대화 상자 템플릿 리소스에서 모달 대화 상자를 만듭니다. DialogBox는 지정된 콜백 함수가 EndDialog 함수를 호출하여 모달 대화 상자를 종료할 때까지 컨트롤을 반환하지 않습니다. (ANSI)

DialogBoxIndirectA 메모리의 대화 상자 템플릿에서 모달 대화 상자를 만듭니다. DialogBoxIndirect는 지정된 콜백 함수가 EndDialog 함수를 호출하여 모달 대화 상자를 종료할 때까지 컨트롤을 반환하지 않습니다. (ANSI)
DialogBoxIndirectParamA 메모리의 대화 상자 템플릿에서 모달 대화 상자를 만듭니다. (ANSI)
DialogBoxIndirectParamW 메모리의 대화 상자 템플릿에서 모달 대화 상자를 만듭니다. (유니코드)
DialogBoxIndirectW 메모리의 대화 상자 템플릿에서 모달 대화 상자를 만듭니다. DialogBoxIndirect는 지정된 콜백 함수가 EndDialog 함수를 호출하여 모달 대화 상자를 종료할 때까지 컨트롤을 반환하지 않습니다. (유니코드)
DialogBoxParamA 대화 상자 템플릿 리소스에서 모달 대화 상자를 만듭니다. (ANSI)
DialogBoxParamW 대화 상자 템플릿 리소스에서 모달 대화 상자를 만듭니다. (유니코드)
DialogBoxW 대화 상자 템플릿 리소스에서 모달 대화 상자를 만듭니다. DialogBox는 지정된 콜백 함수가 EndDialog 함수를 호출하여 모달 대화 상자를 종료할 때까지 컨트롤을 반환하지 않습니다. (유니코드)
DLGPROC CreateDialog 및 DialogBox 함수 패밀리와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
EndDialog 모달 대화 상자를 삭제하여 시스템에서 대화 상자에 대한 모든 처리를 종료합니다.
FindTextA 사용자가 검색할 문자열과 문서에서 텍스트를 검색할 때 사용할 옵션을 지정할 수 있는 시스템 정의 모덜리스 찾기 대화 상자를 만듭니다. (ANSI)

FindTextW 사용자가 검색할 문자열과 문서에서 텍스트를 검색할 때 사용할 옵션을 지정할 수 있는 시스템 정의 모덜리스 찾기 대화 상자를 만듭니다. (유니코드)
GetCurrentDevMode PrintDlgEx에 사용할 현재 선택한 프린터에 대한 정보로 DEVMODE 구조를 채웁니다.
GetCurrentPortName PrintDlgEx에 사용할 현재 포트의 이름을 검색합니다.
GetCurrentPrinterName PrintDlgEx에 사용할 현재 선택한 프린터의 이름을 검색합니다.
GetDialogBaseUnits 시스템 글꼴에서 문자의 평균 너비와 높이인 시스템의 대화 상자 기본 단위를 검색합니다.
GetDlgCtrlID 지정된 컨트롤의 식별자를 검색합니다.
GetDlgItem 지정된 대화 상자에서 컨트롤에 대한 핸들을 검색합니다.
GetDlgItemInt 대화 상자에서 지정된 컨트롤의 텍스트를 정수 값으로 변환합니다.
GetDlgItemTextA 대화 상자에서 컨트롤과 연결된 제목 또는 텍스트를 검색합니다. (ANSI)
GetDlgItemTextW 대화 상자에서 컨트롤과 연결된 제목 또는 텍스트를 검색합니다. (유니코드)
GetFileTitleA 지정된 파일의 이름을 검색합니다. (ANSI)
GetFileTitleW 지정된 파일의 이름을 검색합니다. (유니코드)

GetNextDlgGroupItem 대화 상자에서 지정된 컨트롤 앞에 오거나 뒤에 오는 컨트롤 그룹의 첫 번째 컨트롤에 대한 핸들을 검색합니다.
GetNextDlgTabItem 지정된 컨트롤 앞에 오거나 뒤에 오는 WS_TABSTOP 스타일이 있는 첫 번째 컨트롤에 대한 핸들을 검색합니다.
GetOpenFileNameA 사용자가 열 파일 또는 파일 집합의 드라이브, 디렉터리 및 이름을 지정할 수 있는 열기 대화 상자를 만듭니다. (ANSI)
GetOpenFileNameW 사용자가 열 파일 또는 파일 집합의 드라이브, 디렉터리 및 이름을 지정할 수 있는 열기 대화 상자를 만듭니다. (유니코드)
GetSaveFileNameA 사용자가 저장할 파일의 드라이브, 디렉터리 및 이름을 지정할 수 있는 저장 대화 상자를 만듭니다. (ANSI)
GetSaveFileNameW 사용자가 저장할 파일의 드라이브, 디렉터리 및 이름을 지정할 수 있는 저장 대화 상자를 만듭니다. (유니코드)
HandleMessage PrintDlgEx에서 호출되어 애플리케이션이 인쇄 속성 시트의 일반 페이지 아래쪽에 있는 자식 대화 상자로 전송된 메시지를 처리할 수 있는 기회를 제공합니다.
InitDone 시스템에서 Print 속성 시트의 일반 페이지 초기화를 완료할 때 PrintDlgEx에서 호출됩니다.
IsDialogMessageA 메시지가 지정된 대화 상자의 용도인지 여부를 확인하고, 메시지인 경우 메시지를 처리합니다. (ANSI)
IsDialogMessageW 메시지가 지정된 대화 상자의 용도인지 여부를 확인하고, 메시지인 경우 메시지를 처리합니다. (유니코드)

LPCCHOOKPROC 색 대화 상자의 기본 대화 상자 프로시저에 대한 메시지 또는 알림을 받습니다. ChooseColor 함수와 함께 사용되는 애플리케이션 정의 또는 라이브러리 정의 콜백 함수입니다.
LPCFHOOKPROC 글꼴 대화 상자의 기본 대화 상자 프로시저에 대한 메시지 또는 알림을 받습니다. 이것은 ChooseFont 함수와 함께 사용되는 애플리케이션 정의 또는 라이브러리 정의 콜백 프로시저입니다.
LPFRHOOKPROC 찾기 또는 바꾸기 대화 상자의 기본 대화 상자 프로시저에 대한 메시지 또는 알림을 받습니다.
LPOFNHOOKPROC 대화 상자에서 보낸 알림 메시지를 받습니다.
LPPAGEPAINTHOOK 페이지 설정 대화 상자에서 샘플 페이지의 드로잉을 사용자 지정할 수 있는 메시지를 받습니다. PagePaintHook 후크 프로시저는 PageSetupDlg 함수와 함께 사용되는 애플리케이션 정의 또는 라이브러리 정의 콜백 함수입니다.
LPPAGESETUPHOOK 페이지 설정 대화 상자의 기본 대화 상자 프로시저에 대한 메시지 또는 알림을 받습니다. PageSetupHook 후크 프로시저는 PageSetupDlg 함수와 함께 사용되는 애플리케이션 정의 또는 라이브러리 정의 콜백 함수입니다.
LPPRINTHOOKPROC 인쇄 대화 상자의 기본 대화 상자 프로시저에 대한 메시지 또는 알림을 받습니다. PrintDlg 함수와 함께 사용되는 애플리케이션 정의 또는 라이브러리 정의 콜백 함수입니다.
LPSETUPHOOKPROC PrintDlg 함수와 함께 사용되는 애플리케이션 정의 또는 라이브러리 정의 콜백 함수입니다. 후크 프로시저는 인쇄 설정 대화 상자의 기본 대화 상자 프로시저에 대한 메시지 또는 알림을 받습니다.
MapDialogRect 지정된 대화 상자 단위를 화면 단위(픽셀)로 변환합니다.
MessageBox 시스템 아이콘, 단추 집합 및 상태 또는 오류 정보와 같은 간단한 애플리케이션 관련 메시지가 포함된 모달 대화 상자를 표시합니다. 메시지 상자는 사용자가 클릭한 단추를 나타내는 정수 값을 반환합니다. (MessageBoxW)

MessageBoxA 시스템 아이콘, 단추 집합 및 상태 또는 오류 정보와 같은 간단한 애플리케이션 관련 메시지가 포함된 모달 대화 상자를 표시합니다. 메시지 상자는 사용자가 클릭한 단추를 나타내는 정수 값을 반환합니다. (MessageBoxA)
MessageBoxExA 메시지 상자를 만들고, 표시하고, 작동합니다. (ANSI)
MessageBoxExW 메시지 상자를 만들고, 표시하고, 작동합니다. (유니코드)
MessageBoxIndirectA 메시지 상자를 만들고, 표시하고, 작동합니다. 메시지 상자에는 애플리케이션 정의 메시지 텍스트와 제목, 모든 아이콘 및 미리 정의된 푸시 단추 조합이 포함됩니다. (ANSI)
MessageBoxIndirectW 메시지 상자를 만들고, 표시하고, 작동합니다. 메시지 상자에는 애플리케이션 정의 메시지 텍스트와 제목, 모든 아이콘 및 미리 정의된 푸시 단추 조합이 포함됩니다. (유니코드)
MessageBoxW 시스템 아이콘, 단추 집합 및 상태 또는 오류 정보와 같은 간단한 애플리케이션 관련 메시지가 포함된 모달 대화 상자를 표시합니다. 메시지 상자는 사용자가 클릭한 단추를 나타내는 정수 값을 반환합니다. (MessageBoxW)
MSGBOXCALLBACK 애플리케이션에서 정의한 콜백 함수는 메시지 상자에 대한 도움말 이벤트를 처리합니다.
PageSetupDlgA PageSetupDlg의 ASCII 버전
PageSetupDlgW PageSetupDlg의 와이드 문자열 버전
PrintDlgA PrintDlg의 ASCII 버전
PrintDlgExA PrintDlgEx의 ASCII 버전

PrintDlgExW PrintDlgEx의 와이드 문자열 버전
PrintDlgW PrintDlg의 와이드 문자열 버전
QueryInterface IPrintDialogCallback::QueryInterface 메서드(commdlg.h)는 포인터에 대한 COM 개체를 쿼리하는 IUnknown의 QueryInterface 메서드입니다.
QueryInterface IPrintDialogServices::QueryInterface 메서드(commdlg.h)는 포인터에 대한 COM 개체를 쿼리하는 IUnknown의 QueryInterface 메서드입니다.
릴리스 IPrintDialogCallback::Release 메서드(commdlg.h)는 COM 개체의 인터페이스에 대한 참조 수를 감소시키는 IUnknown의 Release 메서드입니다.
릴리스 IPrintDialogServices::Release 메서드(commdlg.h)는 COM 개체의 인터페이스에 대한 참조 수를 감소시키는 IUnknown의 Release 메서드입니다.
ReplaceTextA 사용자가 검색할 문자열과 대체 문자열을 지정할 수 있는 시스템 정의 모델리스 대화 상자와 찾기 및 바꾸기 작업을 제어하는 옵션을 만듭니다. (ANSI)
ReplaceTextW 사용자가 검색할 문자열과 대체 문자열을 지정할 수 있는 시스템 정의 모델리스 대화 상자와 찾기 및 바꾸기 작업을 제어하는 옵션을 만듭니다. (유니코드)
SelectionChange 인쇄 속성 시트의 일반 페이지에 설치된 프린터 목록에서 다른 프린터를 선택할 때 PrintDlgEx에서 호출됩니다.
SendDlgItemMessageA 대화 상자에서 지정된 컨트롤에 메시지를 보냅니다. (ANSI)

SendDlgItemMessageW 대화 상자에서 지정된 컨트롤에 메시지를 보냅니다. (유니코드)
SetDlgItemInt 대화 상자에서 컨트롤의 텍스트를 지정된 정수 값의 문자열 표현으로 설정합니다.
SetDlgItemTextA 대화 상자에서 컨트롤의 제목 또는 텍스트를 설정합니다. (ANSI)
SetDlgItemTextW 대화 상자에서 컨트롤의 제목 또는 텍스트를 설정합니다. (유니코드)

인터페이스

IPrintDialogCallback Print 속성 시트가 표시되는 동안 애플리케이션이 PrintDlgEx 함수에서 알림 및 메시지를 받을 수 있도록 하는 메서드를 제공합니다.
IPrintDialogServices PrintDlgEx 함수를 사용하여 애플리케이션에서 현재 선택된 프린터에 대한 정보를 검색할 수 있도록 하는 메서드를 제공합니다.

구조체

CHOOSEFONTA ChooseFont 함수가 글꼴 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. 사용자가 대화 상자를 닫은 후 시스템은 이 구조에서 사용자의 선택에 대한 정보를 반환합니다. (ANSI)
CHOOSEFONTW ChooseFont 함수가 글꼴 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. 사용자가 대화 상자를 닫은 후 시스템은 이 구조에서 사용자의 선택에 대한 정보를 반환합니다. (유니코드)

DEVNAMES 프린터의 드라이버, 디바이스 및 출력 포트 이름을 식별하는 문자열을 포함합니다.
DLGITEMTEMPLATE 대화 상자에서 컨트롤의 차원과 스타일을 정의합니다. 이러한 구조체 중 하나 이상이 DLGTEMPLATE 구조체와 결합되어 대화 상자의 표준 템플릿을 형성합니다.
DLGTEMPLATE 대화 상자의 차원과 스타일을 정의합니다.
FINDREPLACEA FindText 및 ReplaceText 함수가 찾기 및 바꾸기 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. (ANSI)
FINDREPLACEW FindText 및 ReplaceText 함수가 찾기 및 바꾸기 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. (유니코드)
MSGBOXPARAMSA 메시지 상자를 표시하는 데 사용되는 정보를 포함합니다. MessageBoxIndirect 함수는 이 구조를 사용합니다. (ANSI)
MSGBOXPARAMSW 메시지 상자를 표시하는 데 사용되는 정보를 포함합니다. MessageBoxIndirect 함수는 이 구조를 사용합니다. (유니코드)
OFNOTIFYA 열기 또는 다른 이름으로 저장 대화 상자의 OFNHookProc 후크 프로시저로 전송된 WM_NOTIFY 메시지에 대한 정보를 포함합니다. WM_NOTIFY 메시지의 lParam 매개 변수는 OFNOTIFY 구조체에 대한 포인터입니다. (ANSI)
OFNOTIFYEXA CDN_INCLUDEITEM 알림 메시지에 대한 정보를 포함합니다. (ANSI)
OFNOTIFYEXW CDN_INCLUDEITEM 알림 메시지에 대한 정보를 포함합니다. (유니코드)

OFNOTIFYW 열기 또는 다른 이름으로 저장 대화 상자의 OFNHookProc 후크 프로시저로 전송된 WM_NOTIFY 메시지에 대한 정보를 포함합니다. WM_NOTIFY 메시지의 lParam 매개 변수는 OFNOTIFY 구조체에 대한 포인터입니다. (유니코드)
OPENFILENAME_NT4A OPENFILENAME_NT4 구조는 _WIN32_WINNT 0x0400 설정된 OPENFILENAME과 동일합니다. (ANSI)
OPENFILENAME_NT4W OPENFILENAME_NT4 구조는 _WIN32_WINNT 0x0400 설정된 OPENFILENAME과 동일합니다. (유니코드)
OPENFILENAMEA GetOpenFileName 및 GetSaveFileName 함수가 다른 이름으로 열기 또는 다른 이름으로 저장 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. 사용자가 대화 상자를 닫은 후 시스템은 이 구조에서 사용자의 선택에 대한 정보를 반환합니다. (ANSI)
OPENFILENAMEW GetOpenFileName 및 GetSaveFileName 함수가 다른 이름으로 열기 또는 다른 이름으로 저장 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. 사용자가 대화 상자를 닫은 후 시스템은 이 구조에서 사용자의 선택에 대한 정보를 반환합니다. (유니코드)
PAGESETUPDLGA PageSetupDlg 함수가 페이지 설정 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. 사용자가 대화 상자를 닫은 후 시스템은 이 구조의 사용자 정의 페이지 매개 변수에 대한 정보를 반환합니다. (ANSI)
PAGESETUPDLGW PageSetupDlg 함수가 페이지 설정 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. 사용자가 대화 상자를 닫은 후 시스템은 이 구조의 사용자 정의 페이지 매개 변수에 대한 정보를 반환합니다. (유니코드)
PRINTDLGA PrintDlg 함수가 인쇄 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. 사용자가 대화 상자를 닫은 후 시스템은 이 구조를 사용하여 사용자의 선택 항목에 대한 정보를 반환합니다. (ANSI)

PRINTDLGEXA PrintDlgEx 함수가 Print 속성 시트를 초기화하는 데 사용하는 정보를 포함합니다. 사용자가 속성 시트를 닫은 후 시스템은 이 구조를 사용하여 사용자의 선택 항목에 대한 정보를 반환합니다. (ANSI)
PRINTDLGEXW PrintDlgEx 함수가 Print 속성 시트를 초기화하는 데 사용하는 정보를 포함합니다. 사용자가 속성 시트를 닫은 후 시스템은 이 구조를 사용하여 사용자의 선택 항목에 대한 정보를 반환합니다. (유니코드)
PRINTDLGW PrintDlg 함수가 인쇄 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. 사용자가 대화 상자를 닫은 후 시스템은 이 구조를 사용하여 사용자의 선택 항목에 대한 정보를 반환합니다. (유니코드)
PRINTPAGERANGE 인쇄 작업의 페이지 범위를 나타냅니다. 인쇄 작업에는 둘 이상의 페이지 범위가 있을 수 있습니다. 이 정보는 PrintDlgEx 함수를 호출할 때 PRINTDLGEX 구조체에 제공됩니다.

직접 조작

아티클2023. 08. 22.

직접 조작 기술에 대한 개요입니다.

직접 조작을 개발하려면 다음 헤더가 필요합니다.

- directmanipulation.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 직접 조작

열거형

DIRECTMANIPULATION_AUTOSCROLL_CONFIGURATION 적용할 자동 스크롤 애니메이션의 유형과 방향을 결정합니다.
DIRECTMANIPULATION_CONFIGURATION 직접 조작에서 사용할 수 있는 상호 작용 구성 상태를 정의합니다.
DIRECTMANIPULATION_DRAG_DROP_CONFIGURATION 끌어서 놓기 상호 작용에 대한 동작을 정의합니다.
DIRECTMANIPULATION_DRAG_DROP_STATUS 뷰포트에 대한 끌어서 놓기 상호 작용 상태를 정의합니다.
DIRECTMANIPULATION_GESTURE_CONFIGURATION SetManualGesture에 전달할 수 있는 제스처를 정의합니다.
DIRECTMANIPULATION_HITTEST_TYPE RegisterHitTestTarget을 통해 등록된 전용 적중 테스트 스레드를 사용할 때 직접 조작에서 적중 테스트를 처리하는 방법을 정의합니다.
DIRECTMANIPULATION_HORIZONTALALIGNMENT 뷰포트 내의 콘텐츠에 대한 가로 맞춤 옵션을 정의합니다.

DIRECTMANIPULATION_INPUT_MODE SetInputMode 또는 SetUpdateMode에 대한 스레딩 동작을 정의합니다. 각 상수의 정확한 의미는 호출된 메서드에 따라 달라집니다.
DIRECTMANIPULATION_INTERACTION_TYPE 직접 조작으로 인식되는 제스처를 정의합니다.
DIRECTMANIPULATION_MOTION_TYPES 직접 조작 동작 유형을 정의합니다.
DIRECTMANIPULATION_SNAPPOINT_COORDINATE 끌기 지점 컬렉션의 좌표계를 정의합니다.
DIRECTMANIPULATION_SNAPPOINT_TYPE 최종 관성 끝 위치를 계산하는 방법을 수정합니다.
DIRECTMANIPULATION_STATUS 직접 조작의 가능한 상태를 정의합니다.
DIRECTMANIPULATION_VERTICALALIGNMENT 뷰포트 내의 콘텐츠에 대한 세로 맞춤 설정을 정의합니다.
DIRECTMANIPULATION_VIEWPORT_OPTIONS 뷰포트에 대한 입력 동작 옵션을 정의합니다.

Functions

중단 뷰포트에서 사용되는 모든 리소스를 해제하고 메모리에서 소멸되도록 준비합니다.
활성화 지정된 창에서 입력 처리 및 콜백 처리를 위한 직접 조작을 활성화합니다.
ActivateConfiguration

입력 상호 작용에 대한 구성을 설정합니다.
AddBehavior 뷰포트에 동작을 추가하고 호출자에게 쿠키를 반환합니다.
AddConfiguration 뷰포트에 대한 상호 작용 구성을 추가합니다.
AddContent 콘텐츠(호출자가 소유)를 작성자와 연결하고, 컴퍼지션 디바이스를 콘텐츠에 할당하고, 다른 컴퍼지션 시각적 개체를 기준으로 컴퍼지션 트리의 콘텐츠 위치를 지정합니다.
AddContent 이동 표시기 같은 보조 콘텐츠를 뷰포트에 추가합니다.
AddContentWithCrossProcessChaining 구성 요소 호스트가 소유한 콘텐츠를 작성자와 연결하고, 컴퍼지션 디바이스를 콘텐츠에 할당하고, 다른 컴퍼지션 시각적 개체를 기준으로 컴퍼지션 트리의 콘텐츠 위치를 지정합니다.
AddEventHandler 뷰포트 이벤트를 수신 대기하는 새 이벤트 처리기를 추가합니다.
CancelContact 이 pointerId에 대해 SetContact에 예약된 모든 호출을 취소합니다.
CancelDeferral DeferContact에서 지연 집합을 취소하고 이 pointerId에 대해 예약된 SetContact 호출을 처리합니다.
CreateBehavior 동작을 만드는 팩터리 메서드입니다.
CreateContent 뷰포트 내부에 보조 콘텐츠(예: 이동 표시기)의 instance 만드는 데 사용되는 팩터리 메서드입니다.
CreateViewport 새 IDirectManipulationViewport 개체를 만드는 데 사용되는 팩터리 메서드입니다.

비활성화 지정된 창에서 입력 처리 및 콜백 처리를 위한 직접 조작을 비활성화합니다.
DeferContact 이 pointerId에 대한 SetContact 호출 실행을 연기할 시간을 지정합니다.
사용 안 함 뷰포트에서 입력 처리를 중지합니다.
사용 뷰포트에서 입력 처리를 시작하거나 다시 시작합니다.
플러시 작성기에서 보류 중인 모든 업데이트를 렌더링을 위해 시스템에 커밋합니다.
GetCenterPoint 콘텐츠 좌표에서 조작의 중심점을 검색합니다.
GetContentRect 뷰포트의 경계 사각형(정의된 경우)을 기준으로 콘텐츠의 경계 사각형을 검색합니다.
GetContentTransform 콘텐츠에 적용된 변환을 검색합니다.
GetInertiaEndTransform 주 콘텐츠의 관성을 포함한 최종 변환을 가져옵니다.
GetNextFrameInfo 작성자에서 컴퍼지션 타이밍 정보를 검색합니다.
GetOutputTransform 콘텐츠에 적용된 최종 변환을 가져옵니다.
GetPrimaryContent IDirectManipulationContent 및 IDirectManipulationPrimaryContent를 구현하는 뷰포트의 기본 콘텐츠를 가져옵니다.

GetService IDirectManipulationDeferContactService 개체를 검색합니다.
GetStatus 이 동작이 연결된 뷰포트에 대한 끌어서 놓기 상호 작용의 상태 가져옵니다.
GetStatus 뷰포트의 상태를 가져옵니다.
GetTag 이 콘텐츠에 설정된 태그 개체를 검색합니다.
GetTag 뷰포트의 태그 값을 가져옵니다.
GetUpdateManager 작성기 업데이트를 수신하는 IDirectManipulationUpdateManager 개체에 대한 포인터를 가져옵니다.
GetViewport 콘텐츠가 포함된 뷰포트를 검색합니다.
GetViewportRect SetViewportRect에서 지정한 뷰포트 좌표계의 원본을 기준으로 뷰포트의 사각형을 검색합니다.
OnContentUpdated 뷰포트 내의 콘텐츠가 업데이트되면 호출됩니다.
OnDragDropStatusChange 끌어서 놓기 동작이 연결된 뷰포트에서 상태 변경이 발생할 때 호출됩니다.
OnInteraction 상호 작용이 감지될 때 호출됩니다.
OnViewportStatusChanged 뷰포트의 상태 변경되면 호출합니다.

OnViewportUpdated 뷰포트의 모든 콘텐츠가 업데이트된 후 호출됩니다.
ProcessInput 앱의 UI 스레드에서 조작 관리자에게 키보드 및 마우스 메시지를 전달합니다.
RegisterHitTestTarget 적중 테스트를 위해 전용 스레드를 등록합니다.
RegisterWaitHandleCallback 핸들에 의해 트리거되는 콜백을 등록합니다.
ReleaseAllContacts 뷰포트와 연결된 모든 연락처를 제거합니다. 뷰포트가 관성(inertia)을 지원하는 경우 관성은 시작됩니다.
ReleaseContact 뷰포트와 연결된 연락처를 제거합니다.
RemoveAllBehaviors 뷰포트에 추가된 모든 동작을 제거합니다.
RemoveBehavior 뷰포트에서 지정된 쿠키와 일치하는 동작을 제거합니다.
RemoveConfiguration 뷰포트에 대한 상호 작용 구성을 제거합니다.
RemoveContent 작성자에서 콘텐츠를 제거합니다.
RemoveContent 뷰포트에서 보조 콘텐츠를 제거합니다.
RemoveEventHandler 뷰포트에서 기존 이벤트 처리기를 제거합니다.

SetChaining 부모 뷰포트에 연결할 수 있는 뷰포트에서 지원되는 동작 유형을 지정합니다.
SetConfiguration 이 동작이 연결된 뷰포트에 대한 자동 스크롤 애니메이션을 수행합니다.
SetConfiguration 이 동작이 연결된 뷰포트에 대한 끌어서 놓기 상호 작용의 구성을 설정합니다.
SetContact 연락처와 뷰포트 간의 연결을 지정합니다.
SetContentRect 뷰포트를 기준으로 콘텐츠의 경계 사각형을 지정합니다.
SetHorizontalAlignment 뷰포트를 기준으로 기본 콘텐츠의 가로 맞춤을 설정합니다.
SetInputMode 입력이 UI 스레드에 표시되는지 지정합니다.
SetManualGesture 직접 조작에서 무시되는 제스처를 설정합니다.
SetSnapCoordinate 끌기 지점 또는 끌기 간격에 대한 좌표계를 지정합니다.
SetSnapInterval 균일한 간격으로 관성 끝 위치에 대한 끌기 지점을 지정합니다.
SetSnapPoints 관성 휴지 위치의 끌기 지점을 지정합니다.
SetSnapType 끌기 지점의 유형을 지정합니다.
SetTag

콘텐츠의 태그 개체를 지정합니다.
SetTag 뷰포트 태그를 설정합니다.
SetUpdateManager 작성기 업데이트를 직접 조작으로 보내는 데 사용되는 업데이트 관리자를 설정합니다.
SetUpdateMode 뷰포트가 입력 이벤트 중에 대신 수동으로 콘텐츠를 업데이트할지 여부를 지정합니다.
SetVerticalAlignment 뷰포트에서 기본 콘텐츠의 세로 맞춤을 지정합니다.
SetViewportOptions 뷰포트가 입력 및 출력을 처리하는 방법을 설정합니다.
SetViewportRect 뷰포트 좌표계의 원점과 관련된 뷰포트의 경계 사각형을 설정합니다.
SetViewportTransform 뷰포트 좌표계에서 창 클라이언트 좌표계로 변환을 지정합니다.
SetZoomBoundaries 확대/축소의 최소 및 최대 경계를 지정합니다.
중지 조작을 중지하고 뷰포트를 준비 상태로 반환합니다.
SyncContentTransform 출력 변환을 유지하면서 콘텐츠 변환을 수정합니다.
SyncDisplayTransform 뷰포트에 대한 표시 변환을 지정하고 출력 변환을 디스플레이 변환의 새 값과 동기화합니다.
UnregisterWaitHandleCallback 콜백을 등록 취소합니다.

업데이트 관성 애니메이션을 업데이트할 때 작성자에게 알 수 있습니다.
업데이트 현재 시간에 직접 조작을 업데이트.
ZoomToRect 뷰포트를 기본 콘텐츠의 특정 영역으로 이동하고 전환에 애니메이션 효과를 주는지 여부를 지정합니다.

인터페이스

IDirectManipulationAutoScrollBehavior 지정된 축 또는 축의 경계에 접근할 때 콘텐츠의 자동 스크롤 애니메이션 동작을 나타냅니다.
IDirectManipulationCompositor 조작된 콘텐츠를 그리기 화면(예: 캔버스(JavaScript를 사용하는 Windows 앱) 또는 Canvas(C++, C# 또는 Visual Basic을 사용하는 Windows 스토어 앱)와 연결하는 작성자 개체를 나타냅니다.
IDirectManipulationCompositor2 조작된 콘텐츠를 여러 프로세스에서 그리기 표면과 연결하는 작성자 개체를 나타냅니다.
IDirectManipulationContent 콘텐츠가 뷰포트 내에서 잘린 시각적 표면을 나타내는 뷰포트 내의 콘텐츠를 캡슐화합니다.
IDirectManipulationDeferContactService 연락처와 뷰포트 간의 연결을 관리하기 위한 서비스를 나타냅니다.
IDirectManipulationDragDropBehavior 교차 슬라이드 또는 길게 누르기 제스처에 의해 트리거되는 끌어서 놓기 상호 작용에 대한 동작을 나타냅니다.
IDirectManipulationDragDropEventHandler 끌어서 놓기 동작 이벤트를 처리하는 메서드를 정의합니다.

IDirectManipulationFrameInfoProvider 애플리케이션에서 사용하는 컴퍼지션 인프라의 대기 시간을 측정하고 이 데이터를 직접 조작에 제공하는 시간 유지 개체를 나타냅니다.
IDirectManipulationInteractionEventHandler 검색될 때 상호 작용을 처리하는 메서드를 정의합니다.
IDirectManipulationManager 클라이언트 애플리케이션에서 사용할 수 있는 모든 직접 조작 기능 및 API에 대한 액세스를 제공합니다.
IDirectManipulationManager2 클라이언트 애플리케이션에서 사용할 수 있는 모든 직접 조작 기능 및 API에 대한 액세스를 제공하는 IDirectManipulationManager 인터페이스를 확장합니다.
IDirectManipulationManager3 클라이언트 애플리케이션에서 사용할 수 있는 모든 직접 조작 기능 및 API에 대한 액세스를 제공하는 IDirectManipulationManager2 인터페이스를 확장합니다.
IDirectManipulationPrimaryContent 뷰포트 내의 기본 콘텐츠를 캡슐화합니다.
IDirectManipulationUpdateHandler 조작 업데이트 이벤트를 처리하는 메서드를 정의합니다.
IDirectManipulationUpdateManager 작성기 업데이트를 직접 조작으로 보내는 방법을 관리합니다.
IDirectManipulationViewport 사용자 상호 작용에서 입력을 수신하고 처리할 수 있는 창(뷰포트라고 함) 내의 지역을 정의합니다.
IDirectManipulationViewport2 뷰포트에서 동작 관리를 제공합니다. 동작은 직접 조작 워크플로의 특정 부분의 기능에 영향을 줍니다.
IDirectManipulationViewportEventHandler 뷰포트에 대한 상태 및 업데이트 이벤트를 처리하는 메서드를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Direct2D

아티클2023. 03. 11.

Direct2D 기술 개요입니다.

Direct2D를 개발하려면 다음 헤더가 필요합니다.

- d2d1.h
- d2d1_1.h
- d2d1_1helper.h
- d2d1_2.h
- d2d1_3.h
- d2d1_3helper.h
- d2d1effectauthor.h
- d2d1effectauthor_1.h
- d2d1effecthelpers.h
- d2d1effecthelpers.hlsli
- d2d1effects.h
- d2d1effects_1.h
- d2d1effects_2.h
- d2d1helper.h
- d2d1svg.h
- dcommon.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Direct2D

클래스

ColorF 알려진 색을 나타내는 상수 집합을 정의하고 다양한 색을 인스턴스화하기 위한 편리한 메서드를 제공합니다.
Matrix3x2F Matrix3x2F 클래스는 3x2 행렬을 나타내며 행렬을 만들기 위한 편리한 메서드를 제공합니다.

Matrix4x4F Matrix4x4F 클래스는 4x4 행렬을 나타내며 행렬을 만들기 위한 편리한 메서드를 제공합니다.

열거형

D2D1_2DAFFINETRANSFORM_INTERPOLATION_MODE 이미지 크기를 조정하기 위해 2D 아핀 변환 효과와 함께 사용할 보간 모드입니다. 품질과 속도의 범위는 6개의 배율 모드가 있습니다.
D2D1_2DAFFINETRANSFORM_PROP 2D 아핀 변환 효과의 속성에 대한 식별자입니다.
D2D1_3DPERSPECTIVETRANSFORM_INTERPOLATION_MODE 3D 큐브 뷰 변환 효과가 이미지에 사용하는 보간 모드입니다. 품질과 속도의 범위는 5개의 배율 모드가 있습니다.
D2D1_3DPERSPECTIVETRANSFORM_PROP 3D 원근 변환 효과의 속성에 대한 식별자입니다.
D2D1_3DTRANSFORM_INTERPOLATION_MODE 3D 변환 효과가 이미지에 사용하는 보간 모드입니다. 품질과 속도의 범위는 5개의 배율 모드가 있습니다.
D2D1_3DTRANSFORM_PROP 3D 변환 효과의 속성에 대한 식별자입니다.
D2D1_ALPHA_MODE 비트맵 또는 렌더링 대상의 알파 값을 처리하는 방법을 지정합니다.
D2D1_ANTIALIAS_MODE 텍스트가 아닌 기본 형식의 가장자리를 렌더링하는 방법을 지정합니다.
D2D1_ARC_SIZE 호가 180도보다 커야 하는지 여부를 지정합니다.

D2D1_ARITHMETICCOMPOSITE_PROP 산술 복합 효과의 속성에 대한 식별자입니다.
D2D1_ATLAS_PROP Atlas 효과의 속성에 대한 식별자입니다.
D2D1_BITMAP_INTERPOLATION_MODE 이미지의 크기를 조정하거나 회전할 때 사용되는 알고리즘을 지정합니다.
D2D1_BITMAP_OPTIONS 비트맵을 사용하는 방법을 지정합니다.
D2D1_BITMAPSOURCE_ALPHA_MODE 비트맵 소스 효과 출력의 알파 모드를 지정합니다.
D2D1_BITMAPSOURCE_INTERPOLATION_MODE 비트맵 소스 효과에서 이미지의 크기를 조정하는 데 사용되는 보간 모드입니다.
D2D1_BITMAPSOURCE_ORIENTATION 비트맵 소스 효과에 의해 대칭 이동 및/또는 회전 작업을 수행할지 여부를 지정합니다.
D2D1_BITMAPSOURCE_PROP 비트맵 소스 효과의 속성에 대한 식별자입니다.
D2D1_BLEND 색 원본 중 하나를 파생하는 방법을 지정하고 선택적으로 색 원본에 대한 사전 피운산자를 지정합니다.
D2D1_BLEND_MODE Blend 효과에 사용되는 혼합 모드입니다.
D2D1_BLEND_OPERATION 두 색 원본에 대한 혼합 연산을 지정합니다.
D2D1_BLEND_PROP Blend 효과의 속성에 대한 식별자입니다.

D2D1_BORDER_EDGE_MODE 테두리 효과의 에지 모드입니다.
D2D1_BORDER_MODE 자르기 효과가 소수 픽셀 좌표에 떨어지는 자르기 사각형을 처리하는 방법을 지정합니다.
D2D1_BORDER_PROP 테두리 효과의 속성에 대한 식별자입니다.
D2D1_BRIGHTNESS_PROP 밝기 효과의 속성에 대한 식별자입니다.
D2D1_BUFFER_PRECISION Direct2D에서 이미징 파이프라인의 비트 깊이를 나타냅니다.
D2D1_CAP_STYLE 선 또는 세그먼트 끝의 셰이프를 설명합니다.
D2D1_CHANGE_TYPE 렌더러가 사용자 지정 꼭짓점 셰이더와 상호 작용하는 방식에 영향을 주는 플래그를 설명합니다. (D2D1_CHANGE_TYPE)
D2D1_CHANNEL_DEPTH 호출자가 렌더링 파이프라인에서 스테이지의 채널 깊이를 제어할 수 있습니다.
D2D1_CHANNEL_SELECTOR 변위 맵이 강도를 영향을 미치는 색 채널을 지정하고 이를 사용하여 이미지를 X 또는 Y 방향으로 공간적으로 대체합니다.
D2D1_CHROMAKEY_PROP Chroma 키 효과의 속성에 대한 식별자입니다.
D2D1_COLOR_BITMAP_GLYPH_SNAP_OPTION 색 비트맵 문자 모양을 렌더링할 때 픽셀 맞춤 정책을 지정합니다.
D2D1_COLOR_CONTEXT_TYPE 색 프로필이 정의되는 방법을 지정합니다.

D2D1_COLOR_INTERPOLATION_MODE 색을 보간하는 방법을 정의합니다.
D2D1_COLOR_SPACE 색 공간에 적용해야 하는 옵션을 정의합니다.
D2D1_COLORMANAGEMENT_ALPHA_MODE 색 관리 효과가 입력 이미지에 포함된 알파 데이터를 해석하는 방법을 나타냅니다.
D2D1_COLORMANAGEMENT_PROP 색 관리 효과의 속성에 대한 식별자입니다.
D2D1_COLORMANAGEMENT_QUALITY 색 관리 효과에 대한 변환의 품질 수준입니다.
D2D1_COLORMANAGEMENT_RENDERING_INTENT 색 관리 효과가 사용해야 하는 ICC 렌더링 의도를 지정합니다.
D2D1_COLORMATRIX_ALPHA_MODE 색 행렬 효과 출력의 알파 모드입니다.
D2D1_COLORMATRIX_PROP 색 행렬 효과의 속성에 대한 식별자입니다.
D2D1_COMBINE_MODE 두 기하 도형을 결합하는 데 사용할 수 있는 여러 메서드를 지정합니다.
D2D1_COMPATIBLE_RENDER_TARGET_OPTIONS 호환되는 렌더링 대상을 만들 때 지원되는 추가 기능을 지정합니다. 이 열거형은 멤버 값의 비트 조합을 허용합니다.
D2D1_COMPOSITE_MODE 모든 Direct2D 혼합 작업에 대한 혼합 모드를 지정하는 데 사용됩니다.
D2D1_COMPOSITE_PROP 복합 효과의 속성에 대한 식별자입니다.

D2D1_CONTRAST_PROP 대비 효과의 속성에 대한 식별자입니다.
D2D1_CONVOLVEMATRIX_PROP Convolve 행렬 효과의 속성에 대한 식별자입니다.
D2D1_CONVOLVEMATRIX_SCALE_MODE Convolve 행렬 효과가 이미지를 해당 커널 단위 길이로 크기 조정하는 데 사용하는 보간 모드입니다. 품질과 속도의 범위는 여섯 가지 배율 모드가 있습니다.
D2D1_CROP_PROP 자르기 효과의 속성에 대한 식별자입니다.
D2D1_DASH_STYLE 스트로크의 대시 및 간격 시퀀스를 설명합니다.
D2D1_DC_INITIALIZE_MODE 디바이스 컨텍스트가 렌더링 대상에서 검색될 때 GDI 렌더링을 위해 초기화되는 방법을 지정합니다.
D2D1_DEBUG_LEVEL Direct2D 디버그 계층에서 제공하는 정보의 형식을 나타냅니다.
D2D1_DEVICE_CONTEXT_OPTIONS 수명 동안 디바이스 컨텍스트에 적용되는 옵션을 지정합니다.
D2D1_DIRECTIONALBLUR_OPTIMIZATION 방향 흐림 효과에 대한 최적화 모드를 지정합니다.
D2D1_DIRECTIONALBLUR_PROP 방향 흐림 효과의 속성에 대한 식별자입니다.
D2D1_DISCRETETRANSFER_PROP 불연속 전송 효과의 속성에 대한 식별자입니다.
D2D1_DISPLACEMENTMAP_PROP 변위 맵 효과의 속성에 대한 식별자입니다.

D2D1_DISTANTDIFFUSE_PROP 원거리 확산 조명 효과의 속성에 대한 식별자입니다.
D2D1_DISTANTDIFFUSE_SCALE_MODE 효과를 사용하여 이미지를 해당 커널 단위 길이로 조정하는 데 사용하는 보간 모드입니다. 품질과 속도의 범위는 여섯 가지 배율 모드가 있습니다.
D2D1_DISTANTSPECULAR_PROP 원거리 반사 조명 효과의 속성에 대한 식별자입니다.
D2D1_DISTANTSPECULAR_SCALE_MODE 원거리 반사 조명 효과가 이미지를 해당 커널 단위 길이로 조정하는 데 사용하는 보간 모드입니다. 품질과 속도의 범위는 여섯 가지 배율 모드가 있습니다.
D2D1_DPICOMPENSATION_INTERPOLATION_MODE DPI 보정 효과가 이미지 크기를 조정하는 데 사용하는 보간 모드입니다.
D2D1_DPICOMPENSATION_PROP DPI 보정 효과의 속성에 대한 식별자입니다.
D2D1_DRAW_TEXT_OPTIONS 텍스트 맞춤을 표시하지 않거나 레이아웃 사각형에 클리핑을 사용할지 여부를 지정합니다. 이 열거형은 멤버 값의 비트 조합을 허용합니다.
D2D1_EDGEDETECTION_MODE 에지 검색 효과의 D2D1_EDGEDETECTION_PROP_MODE 속성 값입니다.
D2D1_EDGEDETECTION_PROP 에지 검색 효과의 속성에 대한 식별자입니다.
D2D1_EMBOSS_PROP Emboss 효과의 속성에 대한 식별자입니다.
D2D1_EXPOSURE_PROP 노출 효과의 속성에 대한 식별자입니다.

D2D1_EXTEND_MODE 브러시가 일반 콘텐츠 영역 외부의 영역을 그리는 방법을 지정합니다.
D2D1_FACTORY_TYPE Direct2D가 여러 스레드에서 안전하게 액세스할 수 있도록 ID2D1Factory 및 생성된 리소스에 대한 동기화를 제공하는지 여부를 지정합니다.
D2D1_FEATURE ID2D1EffectContext::CheckFeatureSupport를 사용하여 쿼리할 수 있는 기본 Direct3D 디바이스의 기능을 정의합니다.
D2D1_FEATURE_LEVEL 렌더링 대상에서 하드웨어 렌더링에 필요한 최소 DirectX 지원에 대해 설명합니다.
D2D1_FIGURE_BEGIN 특정 ID2D1SimplifiedGeometrySink 그림이 채워져 있는지 빈지 여부를 나타냅니다.
D2D1_FIGURE_END 특정 ID2D1SimplifiedGeometrySink 그림이 열려 있는지 닫혀 있는지 여부를 나타냅니다.
D2D1_FILL_MODE 기하 도형 또는 그림의 교차 영역을 결합하여 복합 기하 도형의 영역을 형성하는 방법을 지정합니다.
D2D1_FILTER 변환이 입력 텍스처에 사용하도록 선택할 수 있는 필터링 모드를 나타냅니다.
D2D1_FLOOD_PROP 홍수 효과의 속성에 대한 식별자입니다.
D2D1_GAMMA 보간에 사용되는 감마를 지정합니다.
D2D1_GAMMA1 보간 및 혼합에 사용되는 감마를 결정합니다.

D2D1_GAMMATRANSFER_PROP 감마 전송 효과의 속성에 대한 식별자입니다.
D2D1_GAUSSIANBLUR_OPTIMIZATION 가우스 흐림 효과에 대한 최적화 모드입니다.
D2D1_GAUSSIANBLUR_PROP 가우스 흐림 효과의 속성에 대한 식별자입니다.
D2D1_GEOMETRY_RELATION 한 기하 도형 개체가 다른 기하 도형 개체와 공간적으로 관련되는 방법을 설명합니다.
D2D1_GEOMETRY_SIMPLIFICATION_OPTION ID2D1SimplifiedGeometrySink로 기하 도형을 단순화하는 방법을 지정합니다.
D2D1_HDRTONEMAP_DISPLAY_MODE HDR 톤 맵 효과의 D2D1_HDRTONEMAP_PROP_DISPLAY_MODE 속성 값을 지정하는 상수를 정의합니다.
D2D1_HDRTONEMAP_PROP HDR 톤 맵 효과의 최상위 속성을 식별하는 상수 정의
D2D1_HIGHLIGHTSANDSHADOWS_INPUT_GAMMA 강조 표시 및 그림자 효과의 D2D1_HIGHLIGHTSANDSHADOWS_PROP_INPUT_GAMMA 속성 값입니다.
D2D1_HIGHLIGHTSANDSHADOWS_PROP 강조 표시 및 그림자 효과의 속성에 대한 식별자입니다.
D2D1_HISTOGRAM_PROP 히스토그램 효과의 속성에 대한 식별자입니다.
D2D1_HUEROTATION_PROP Hue 회전 효과의 속성에 대한 식별자입니다.
D2D1_HUETORGB_INPUT_COLOR_SPACE Hue-RGB 효과의 D2D1_HUETORGB_PROP_INPUT_COLOR_SPACE 속성 값입니다.

D2D1_HUETORGB_PROP Hue에서 RGB 효과의 속성에 대한 식별자입니다.
D2D1_IMAGE_SOURCE_FROM_DXGI_OPTIONS CreateImageSourceFromDxgi(있는 경우)에서 수행하는 기본 변환을 제어하는 옵션 플래그입니다.
D2D1_IMAGE_SOURCE_LOADING_OPTIONS 새 ID2D1ImageSource를 만들 때 옵션 플래그를 제어합니다.
D2D1_INK_NIB_SHAPE 잉크 닙(펜 팁)의 모양을 D2D1_INK_STYLE_PROPERTIES 구조체의 일부로 지정합니다.
D2D1_INTERPOLATION_MODE ID2D1DeviceContext::D rawImage 및 2D 아핀 변환 효과를 사용하여 이미지 크기 조정 품질을 지정하는 데 사용됩니다.
D2D1_LAYER_OPTIONS 레이어 리소스를 적용하여 레이어를 만들 때 적용할 수 있는 옵션을 지정합니다.
D2D1_LAYER_OPTIONS1 레이어 콘텐츠를 준비하는 방법을 지정합니다.
D2D1_LINE_JOIN 두 선 또는 세그먼트를 연결하는 모양을 설명합니다.
D2D1_LINEARTRANSFER_PROP 선형 전송 효과의 속성에 대한 식별자입니다.
D2D1_LOOKUPTABLE3D_PROP 3D 조회 테이블 효과의 속성에 대한 식별자입니다.
D2D1_MAP_OPTIONS 해당 ID2D1Bitmap1에서 매핑할 메모리를 처리하는 방법을 지정합니다.
D2D1_MORPHOLOGY_MODE 형태 효과의 모드입니다.

D2D1_MORPHOLOGY_PROP 형태 효과의 속성에 대한 식별자입니다.
D2D1_OPACITY_MASK_CONTENT 불투명 마스크에 그래픽 또는 텍스트가 포함되어 있는지 여부를 설명합니다. Direct2D는 이 정보를 사용하여 불투명 마스크를 혼합할 때 사용할 감마 공간을 결정합니다.
D2D1_OPACITYMETADATA_PROP 불투명 메타데이터 효과의 속성에 대한 식별자입니다.
D2D1_ORIENTATION 이미지가 표시되는 대칭 이동 및 회전을 지정합니다.
D2D1_PATCH_EDGE_MODE 그라데이션 메시 가장자리를 렌더링하는 방법을 지정합니다.
D2D1_PATH_SEGMENT 세그먼트를 스트로크해야 하는지 여부와 이 세그먼트와 이전 세그먼트 간의 조인이 원활해야 하는지 여부를 나타냅니다. 이 열거형은 멤버 값의 비트 조합을 허용합니다.
D2D1_PIXEL_OPTIONS 픽셀 셰이더 샘플링을 제한하는 방법을 나타냅니다.
D2D1_POINTDIFFUSE_PROP 점 확산 조명 효과의 속성에 대한 식별자입니다.
D2D1_POINTDIFFUSE_SCALE_MODE 점 확산 조명 효과가 이미지 크기를 해당 커널 단위 길이로 조정하는 데 사용하는 보간 모드입니다. 품질과 속도의 범위는 여섯 배율 모드가 있습니다.
D2D1_POINTSPECULAR_PROP 점 반사 조명 효과의 속성에 대한 식별자입니다.
D2D1_POINTSPECULAR_SCALE_MODE 점 반사 조명 효과가 이미지 크기를 해당 커널 단위 길이로 조정하는 데 사용하는 보간 모드입니다. 품질과 속도의 범위는 여섯 배율 모드가 있습니다.

D2D1_POSTERIZE_PROP Posterize 효과의 속성에 대한 식별자입니다.
D2D1_PRESENT_OPTIONS 렌더링 대상이 콘텐츠를 표시할 때 동작하는 방법을 설명합니다. 이 열거형은 멤버 값의 비트 조합을 허용합니다.
D2D1_PRIMITIVE_BLEND 모든 Direct2D 기본 형식에 대한 기하학적 혼합 모드를 지정하는 데 사용됩니다.
D2D1_PRINT_FONT_SUBSET_MODE 인쇄하는 동안 글꼴 리소스를 하위 집합으로 설정해야 하는 시기를 정의합니다.
D2D1_PROPERTY ID2D1Effect에 대한 ID2D1Properties 인터페이스에 있는 시스템 속성의 인덱스를 지정합니다.
D2D1_PROPERTY_TYPE Direct2D 속성 인터페이스에서 지원하는 속성의 형식을 지정합니다.
D2D1_RENDER_TARGET_TYPE 렌더링 대상에서 하드웨어 또는 소프트웨어 렌더링을 사용하는지 또는 Direct2D가 렌더링 모드를 선택해야 하는지에 대해 설명합니다.
D2D1_RENDER_TARGET_USAGE 렌더링 대상이 원격으로 지정되는 방법 및 GDI와 호환되어야 하는지 여부를 설명합니다. 이 열거형은 멤버 값의 비트 조합을 허용합니다.
D2D1_RENDERING_PRIORITY 렌더링 우선 순위는 Direct2D가 렌더링 워크로드를 제한하는 정도에 영향을 줍니다.
D2D1_RGBTOHUE_OUTPUT_COLOR_SPACE RGB to Hue 효과의 D2D1_RGBTOHUE_PROP_OUTPUT_COLOR_SPACE 속성 값입니다.
D2D1_RGBTOHUE_PROP RGB to Hue 효과의 속성에 대한 식별자입니다.

D2D1_SATURATION_PROP 채도 효과의 속성에 대한 식별자입니다.
D2D1_SCALE_INTERPOLATION_MODE 눈금 효과가 이미지 크기를 조정하는 데 사용하는 보간 모드입니다. 품질과 속도의 범위는 6개의 배율 모드가 있습니다.
D2D1_SCALE_PROP 눈금 효과의 속성에 대한 식별자입니다.
D2D1_SEPIA_PROP 세피아 효과의 속성에 대한 식별자입니다.
D2D1_SHADOW_OPTIMIZATION 그림자 효과에 대한 성능 최적화 수준입니다.
D2D1_SHADOW_PROP 그림자 효과의 속성에 대한 식별자입니다.
D2D1_SHARPEN_PROP 선명도 효과의 속성에 대한 식별자입니다.
D2D1_SPOTDIFFUSE_PROP 스폿 확산 조명 효과의 속성에 대한 식별자입니다.
D2D1_SPOTDIFFUSE_SCALE_MODE 스폿 확산 조명 효과가 이미지를 해당 커널 단위 길이로 조정하는 데 사용하는 보간 모드입니다. 품질과 속도의 범위는 여섯 가지 배율 모드가 있습니다.
D2D1_SPOTSPECULAR_PROP 스폿 반사 조명 효과의 속성에 대한 식별자입니다.
D2D1_SPOTSPECULAR_SCALE_MODE 스폿 반사 조명 효과가 이미지를 해당 커널 단위 길이로 조정하는 데 사용하는 보간 모드입니다. 품질과 속도의 범위는 여섯 가지 배율 모드가 있습니다.

D2D1_SPRITE_OPTIONS ID2D1DeviceContext3::D rawSpriteBatch 호출의 일부로 스프라이트 일괄 처리를 그리는 방법의 추가 측면을 지정합니다.
D2D1_STRAIGHTEN_PROP 똑바른 효과의 속성에 대한 식별자입니다.
D2D1_STRAIGHTEN_SCALE_MODE 똑바른 효과의 D2D1_STRAIGHTEN_PROP_SCALE_MODE 속성에 대한 값입니다.
D2D1_STROKE_TRANSFORM_TYPE 월드 변환, 인치당 점 수(dpi) 및 스트로크 너비가 원형 스트로크에 사용되는 펜의 모양에 미치는 영향을 정의합니다.
D2D1_SUBPROPERTY 모든 속성에 있을 수 있는 시스템 하위 속성의 인덱스를 지정합니다.
D2D1_SVG_ASPECT_ALIGN SVG preserveAspectRatio 특성의 맞춤 부분입니다.
D2D1_SVG_ASPECT_SCALING SVG preserveAspectRatio 특성의 meetOrSlice 부분입니다.
D2D1_SVG_ATTRIBUTE_POD_TYPE 설정하거나 가져올 SVG POD 특성의 형식을 정의합니다.
D2D1_SVG_ATTRIBUTE_STRING_TYPE 설정하거나 가져올 SVG 문자열 특성의 형식을 정의합니다.
D2D1_SVG_DISPLAY SVG 표시 속성의 값을 지정합니다.
D2D1_SVG_LENGTH_UNITS SVG 길이의 단위를 지정합니다.
D2D1_SVG_LINE_CAP SVG stroke-linecap 속성의 값을 지정합니다.

D2D1_SVG_LINE_JOIN SVG stroke-linejoin 속성의 값을 지정합니다.
D2D1_SVG_OVERFLOW SVG 오버플로 속성의 값을 지정합니다.
D2D1_SVG_PAINT_TYPE SVG 채우기 또는 스트로크에 대한 페인트 유형을 지정합니다.
D2D1_SVG_PATH_COMMAND 명령된 경로를 나타냅니다. 각 명령은 세그먼트 데이터의 부동 소수자를 참조할 수 있습니다. _ABSOLUTE 끝나는 명령은 데이터를 절대 좌표로 해석합니다. _RELATIVE 끝나는 명령은 데이터를 이전 지점에 상대적인 것으로 해석합니다.
D2D1_SVG_UNIT_TYPE SVG 그라데이션 또는 clipPath 요소에 사용되는 좌표계를 정의합니다.
D2D1_SVG_VISIBILITY SVG 표시 유형 속성의 값을 지정합니다.
D2D1_SWEEP_DIRECTION 타원형 호가 그려지는 방향을 정의합니다.
D2D1_TABLETRANSFER_PROP 테이블 전송 효과의 속성에 대한 식별자입니다.
D2D1_TEMPERATUREANDTINT_PROP 온도 및 색조 효과의 속성에 대한 식별자입니다.
D2D1_TEXT_ANTIALIAS_MODE 텍스트를 그리는 데 사용되는 앤티앨리어싱 모드를 설명합니다.
D2D1_THREADING_MODE 디바이스, 팩터리 및 디바이스 컨텍스트를 동시에 만드는 동안 사용되는 스레딩 모드를 지정합니다.

D2D1_TILE_PROP 타일 효과의 속성에 대한 식별자입니다.
D2D1_TRANSFORMED_IMAGE_SOURCE_OPTIONS 변환된 이미지 원본에 대한 옵션 플래그입니다.
D2D1_TURBULENCE_NOISE 난기류 효과에 대한 난기류 노이즈 모드입니다. 프랙탈 노이즈 또는 난기류 함수를 기반으로 비트맵을 생성할지 여부를 나타냅니다.
D2D1_TURBULENCE_PROP 난기류 효과의 속성에 대한 식별자입니다.
D2D1_UNIT_MODE Direct2D의 단위를 해석하는 방법을 지정합니다.
D2D1_VERTEX_OPTIONS 렌더러가 사용자 지정 꼭짓점 셰이더와 상호 작용하는 방식에 영향을 주는 플래그를 설명합니다. (D2D1_VERTEX_OPTIONS)
D2D1_VERTEX_USAGE 꼭짓점 버퍼가 자주 변경되지 않는지 또는 자주 변경되는지를 나타냅니다.
D2D1_VIGNETTE_PROP 비네팅 효과의 속성에 대한 식별자입니다.
D2D1_WHITELEVELADJUSTMENT_PROP 흰색 수준 조정 효과의 최상위 속성을 식별하는 상수 정의
D2D1_WINDOW_STATE 창이 폐색되었는지 여부를 설명합니다.
D2D1_YCBCR_CHROMA_SUBSAMPLING YCbCr 효과에 사용되는 입력 크로마 이미지의 크로마 하위 샘플링을 지정합니다.
D2D1_YCBCR_INTERPOLATION_MODE YCbCr 효과의 보간 모드를 지정합니다.

D2D1_YCBCR_PROP YCbCr 효과의 속성에 대한 식별자입니다.

Functions

AddArc 경로 기하 도형에 단일 호를 추가합니다.
AddArc 단일 호를 만들어 경로 기하 도형에 추가합니다.
AddBezier 현재 점과 지정된 끝점 간에 입방형 3차원 곡선을 만듭니다.
AddBezier 현재 지점과 지정된 엔드포인트 사이에 입방형 3차원 곡선을 만듭니다.
AddBeziers 입방형 베지어 곡선의 시퀀스를 만들고 기하 도형 싱크에 추가합니다.
AddLine 현재 점과 지정된 끝점 사이에 선 세그먼트를 만들고 기하 도형 싱크에 추가합니다.
AddLines 지정된 점을 사용하여 선 시퀀스를 만들고 기하 도형 싱크에 추가합니다.
AddNode 제공된 노드를 변환 그래프에 추가합니다.
Addpage 전달된 명령 목록의 Direct2D 기본 형식을 인쇄 하위 시스템에 사용할 고정된 페이지 표현으로 변환합니다.

AddQuadraticBezier 현재 점과 지정된 끝점 간에 정방형 3차원 곡선을 만듭니다.
AddQuadraticBezier 현재 점과 지정된 끝점 사이에 이차 베지어 곡선을 만들고 기하 도형 싱크에 추가합니다.
AddQuadraticBeziers 단일 호출에서 4차원 Bezier 세그먼트 시퀀스를 배열로 추가합니다.
AddSegments 지정된 세그먼트를 이 잉크 개체의 끝에 추가합니다.
AddSprites 지정된 스프라이트를 이 스프라이트 일괄 처리의 끝에 추가합니다.
AddTriangles 지정된 삼각형을 싱크에 복사합니다.
AppendChild 자식 목록에 요소를 추가합니다.
ArcSegment D2D1_ARC_SEGMENT 구조를 만듭니다.
BeginDraw 이 렌더링 대상에 그리기를 시작합니다.
BeginDraw 그리기가 시작된다는 것을 명령 싱크의 구현에 알립니다.
BeginFigure 지정된 지점에서 새 그림을 시작합니다.
BezierSegment D2D1_BEZIER_SEGMENT 구조를 만듭니다.

BindDC 렌더링 대상을 그리기 명령을 발급하는 디바이스 컨텍스트에 바인딩합니다.
BitmapBrushProperties D2D1_BITMAP_BRUSH_PROPERTIES 구조를 만듭니다.
BitmapBrushProperties1 D2D1_BITMAP_BRUSH_PROPERTIES1 구조를 만듭니다.
BitmapProperties D2D1_BITMAP_PROPERTIES 구조를 만듭니다.
BitmapProperties1 D2D1_BITMAP_PROPERTIES1 구조를 만듭니다.
BlendImage 지정된 혼합 모드를 사용하여 디바이스 컨텍스트에 이미지를 그립니다. 결과는 Direct2D의 기본 제공 Blend 효과를 사용하는 것과 같습니다. (ID2D1CommandSink5.BlendImage)
BlendImage 지정된 혼합 모드를 사용하여 디바이스 컨텍스트에 이미지를 그립니다. 결과는 Direct2D의 기본 제공 Blend 효과를 사용하는 것과 같습니다. (ID2D1DeviceContext6.BlendImage)
BrushProperties D2D1_BRUSH_PROPERTIES 구조를 만듭니다.
CalculateThreadgroups 이 메서드를 사용하면 컴퓨팅 셰이더 기반 변환에서 채워야 하는 출력 픽셀 수에 따라 실행할 스레드 그룹 수를 선택할 수 있습니다.
CheckFeatureSupport 이는 선택적 기능이 D3D 디바이스에서 지원되는지 여부를 나타냅니다.
CheckWindowState 이 렌더링 대상과 연결된 HWND가 폐색되었는지 여부를 나타냅니다.

지우기 드로잉 영역을 지정된 색으로 지웁니다. (오버로드 2/3)
지우기 드로잉 영역을 지정된 색으로 지웁니다. (오버로드 1/3)
지우기 드로잉 영역을 지정된 색으로 지웁니다. (ID2D1CommandSink.Clear)
지우기 변환 그래프에서 변환 노드 및 모든 연결을 지웁니다.
지우기 이 스프라이트 일괄 처리에서 모든 스프라이트를 제거합니다.
ClearResources Direct2D에서 사용하는 모든 렌더링 리소스를 지웁니다.
복제 이 특성 값의 복제본을 만듭니다. 만들 때 복제된 특성은 어떤 요소에도 설정되지 않습니다.
닫기 기하 도형 싱크를 닫고, 오류 상태인지 여부를 나타내고, 싱크의 오류 상태를 다시 설정합니다.
닫기 싱크를 닫고 해당 오류 상태를 반환합니다.
닫기 명령 목록에 명령 허용을 중지하여 효과에 대한 입력 또는 ID2D1DeviceContext::D rawImage 호출에 사용할 수 있도록 지시합니다.
닫기 나머지 모든 리소스를 인쇄 하위 시스템에 전달한 다음 현재 인쇄 작업을 정리하고 닫습니다.
ColorF 지정된 빨강, 녹색, 파랑 및 알파 값을 포함하는 ColorF 클래스의 새 인스턴스를 인스턴스화합니다.

ColorF 지정된 RGB 및 알파 값을 포함하는 ColorF 클래스의 새 인스턴스를 인스턴스화합니다.
CombineWithGeometry 이 기하 도형을 지정된 기하 도형과 결합하고 결과를 ID2D1SimplifiedGeometrySink에 저장합니다. (오버로드 3/4)
CombineWithGeometry 이 기하 도형을 지정된 기하 도형과 결합하고 결과를 ID2D1SimplifiedGeometrySink에 저장합니다. (오버로드 2/4)
CombineWithGeometry 이 기하 도형을 지정된 기하 도형과 결합하고 결과를 ID2D1SimplifiedGeometrySink에 저장합니다. (오버로드 4/4)
CombineWithGeometry 이 기하 도형을 지정된 기하 도형과 결합하고 결과를 ID2D1SimplifiedGeometrySink에 저장합니다. (오버로드 1/4)
CompareWithGeometry 이 기하 도형과 지정된 기하 도형 간의 교차점에 대해 설명합니다. 비교는 기본 평면화 허용 오차를 사용하여 수행됩니다. (오버로드 1/2)
CompareWithGeometry 이 기하 도형과 지정된 기하 도형 간의 교차점에 대해 설명합니다. 비교는 지정된 평면화 허용 오차를 사용하여 수행됩니다.
CompareWithGeometry 이 기하 도형과 지정된 기하 도형 간의 교차점에 대해 설명합니다. 비교는 기본 평면화 허용 오차를 사용하여 수행됩니다. (오버로드 2/2)
CompareWithGeometry 이 기하 도형과 지정된 기하 도형 간의 교차점에 대해 설명합니다. 비교는 지정된 평면화 허용 오차를 사용하여 수행됩니다.
ComputeArea 지정된 행렬에 의해 변형되고 기본 허용 오차를 사용하여 평면화된 후 기하 도형의 영역을 계산합니다.

ComputeArea 지정된 행렬에 의해 변형되고 지정된 허용 오차를 사용하여 평면화된 후 기하 도형의 영역을 계산합니다. (오버로드 2/3)
ComputeArea 지정된 행렬에 의해 변형되고 지정된 허용 오차를 사용하여 평면화된 후 기하 도형의 영역을 계산합니다. (오버로드 3/3)
ComputeArea 지정된 행렬에 의해 변형되고 지정된 허용 오차를 사용하여 평면화된 후 기하 도형의 영역을 계산합니다. (오버로드 1/3)
ComputeLength 각 세그먼트가 선으로 언롤된 것처럼 기하 도형의 길이를 계산합니다. (오버로드 1/4)
ComputeLength 각 세그먼트가 선으로 언롤된 것처럼 기하 도형의 길이를 계산합니다. (오버로드 3/4)
ComputeLength 각 세그먼트가 선으로 언롤된 것처럼 기하 도형의 길이를 계산합니다. (오버로드 2/4)
ComputeLength 각 세그먼트가 선으로 언롤된 것처럼 기하 도형의 길이를 계산합니다. (오버로드 4/4)
ComputePointAndSegmentAtLength 경로 기하 도형을 따라 지정된 거리에 있는 점을 점이 있는 세그먼트의 인덱스 및 해당 지점의 방향 벡터와 함께 계산합니다. (오버로드 1/4)
ComputePointAndSegmentAtLength 경로 기하 도형을 따라 지정된 거리에 있는 점을 점이 있는 세그먼트의 인덱스 및 해당 지점의 방향 벡터와 함께 계산합니다. (오버로드 3/4)
ComputePointAndSegmentAtLength 경로 기하 도형을 따라 지정된 거리에 있는 점을 점이 있는 세그먼트의 인덱스 및 해당 지점의 방향 벡터와 함께 계산합니다. (오버로드 2/4)

ComputePointAndSegmentAtLength 경로 기하 도형을 따라 지정된 거리에 있는 점을 점이 있는 세그먼트의 인덱스 및 해당 지점의 방향 벡터와 함께 계산합니다. (오버로드 4/4)
ComputePointAtLength 지정된 행렬에 의해 변환되고 기본 허용 오차를 사용하여 평면화된 후 기하 도형을 따라 지정된 거리에 있는 점 및 탄젠트 벡터를 계산합니다. (오버로드 1/2)
ComputePointAtLength 지정된 행렬에 의해 변환되고 지정된 허용 오차를 사용하여 평면화된 후 기하 도형을 따라 지정된 거리에 있는 점 및 탄젠트 벡터를 계산합니다. (오버로드 2/2)
ComputePointAtLength 지정된 행렬에 의해 변환되고 기본 허용 오차를 사용하여 평면화된 후 기하 도형을 따라 지정된 거리에 있는 점 및 탄젠트 벡터를 계산합니다. (오버로드 2/2)
ComputePointAtLength 지정된 행렬에 의해 변환되고 지정된 허용 오차를 사용하여 평면화된 후 기하 도형을 따라 지정된 거리에 있는 점 및 탄젠트 벡터를 계산합니다. (오버로드 1/2)
ConnectNode 변환 그래프 내에 두 개의 노드를 연결합니다.
ConnectToEffectInput 그래프 내의 변환 노드를 캡슐화 효과의 해당 효과 입력에 연결합니다.
ConvertColorSpace 한 색 공간에서 다른 색 공간으로 D2D1_COLOR_F 변환합니다.
CopyFromBitmap 지정된 영역을 지정된 비트맵에서 현재 비트맵으로 복사합니다.
CopyFromMemory 메모리에서 현재 비트맵으로 지정된 영역을 복사합니다.
CopyFromRenderTarget 지정된 렌더링 대상에서 현재 비트맵으로 지정된 영역을 복사합니다.

CreateBitmap 초기화되지 않은 Direct2D 비트맵을 만듭니다.
CreateBitmap 포인터에서 메모리 내 원본 데이터에 대한 Direct2D 비트맵을 만듭니다.
CreateBitmap 지정된 비트맵에서 ID2D1BitmapBrush를 만듭니다. (오버로드 2/3)
CreateBitmap 대상 표면, CPU로 다시 읽기 또는 DrawBitmap 및 ID2D1BitmapBrush API의 원본으로 사용할 수 있는 비트맵을 만듭니다. 또한 색 컨텍스트 정보를 비트맵에 전달할 수 있습니다. (오버로드 1/2)
CreateBitmap 대상 표면, CPU로 다시 읽기 또는 DrawBitmap 및 ID2D1BitmapBrush API의 원본으로 사용할 수 있는 비트맵을 만듭니다. 또한 색 컨텍스트 정보를 비트맵에 전달할 수 있습니다. (오버로드 2/2)
CreateBitmapBrush 지정된 비트맵에서 ID2D1BitmapBrush를 만듭니다. (오버로드 1/3)
CreateBitmapBrush 지정된 비트맵에서 ID2D1BitmapBrush를 만듭니다. 브러시는 불투명도 및 변환에 기본값을 사용합니다.
CreateBitmapBrush 지정된 비트맵에서 ID2D1BitmapBrush를 만듭니다. (오버로드 3/3)
CreateBitmapBrush 지정된 비트맵에서 ID2D1BitmapBrush를 만듭니다. 브러시는 확장 모드, 보간 모드, 불투명도 및 변환에 기본값을 사용합니다.
CreateBitmapBrush 비트맵 브러시를 만들고 입력 이미지는 Direct2D 비트맵 개체입니다. (오버로드 4/4)
CreateBitmapBrush 비트맵 브러시를 만들고 입력 이미지는 Direct2D 비트맵 개체입니다. (오버로드 3/4)

CreateBitmapBrush 비트맵 브러시를 만들고 입력 이미지는 Direct2D 비트맵 개체입니다. (오버로드 2/4)
CreateBitmapBrush 비트맵 브러시를 만들고 입력 이미지는 Direct2D 비트맵 개체입니다. (오버로드 1/4)
CreateBitmapFromDxgiSurface 대상 표면으로 설정하거나 추가 색 컨텍스트 정보를 지정할 수 있는 DXGI 표면에서 비트맵을 만듭니다. (오버로드 2/2)
CreateBitmapFromDxgiSurface 대상 표면으로 설정하거나 추가 색 컨텍스트 정보를 지정할 수 있는 DXGI 표면에서 비트맵을 만듭니다. (오버로드 1/2)
CreateBitmapFromWicBitmap 지정된 WIC(Microsoft Windows 이미징 구성 요소) 비트맵을 복사하여 ID2D1Bitmap을 만듭니다. (오버로드 2/4)
CreateBitmapFromWicBitmap 지정된 WIC(Microsoft Windows 이미징 구성 요소) 비트맵을 복사하여 ID2D1Bitmap을 만듭니다. (오버로드 4/4)
CreateBitmapFromWicBitmap 지정된 WIC(Microsoft Windows 이미징 구성 요소) 비트맵을 복사하여 ID2D1Bitmap을 만듭니다. (오버로드 3/4)
CreateBitmapFromWicBitmap WIC 비트맵을 복사하여 Direct2D 비트맵을 만듭니다. (오버로드 1/2)
CreateBitmapFromWicBitmap WIC 비트맵을 복사하여 Direct2D 비트맵을 만듭니다. (오버로드 2/2)
CreateBitmapFromWicBitmap 지정된 WIC(Microsoft Windows 이미징 구성 요소) 비트맵을 복사하여 ID2D1Bitmap을 만듭니다. (오버로드 1/4)
CreateBlendTransform 이렇게 하면 변환 그래프에 삽입할 수 있는 혼합 변환이 만들어집니다.

CreateBorderTransform 전달된 확장 모드에 따라 모든 방향으로 입력을 무한히 확장하는 변환을 만듭니다.
CreateBoundsAdjustmentTransform 경계 조정 변환을 만들고 반환합니다.
CreateChild 태그 이름에서 요소를 만듭니다. 자식 목록에 요소가 추가됩니다.
CreateColorContext 색 컨텍스트를 만듭니다.
CreateColorContext 색 공간에서 색 컨텍스트를 만듭니다.
CreateColorContextFromDxgiColorSpace DXGI 색 공간 형식에서 색 컨텍스트를 만듭니다. '최적' 모드에서 색 관리 효과에만 사용할 수 있습니다.
CreateColorContextFromFilename 지정된 파일 이름에서 색 컨텍스트를 로드하여 만듭니다. 프로필 바이트는 파일 이름으로 지정된 파일의 내용입니다.
CreateColorContextFromFilename 지정된 파일 이름에서 색 컨텍스트를 로드하여 만듭니다. 프로필 바이트는 파일 이름으로 지정된 파일의 내용입니다.
CreateColorContextFromSimpleColorProfile 간단한 색 프로필에서 색 컨텍스트를 만듭니다. '최적' 모드에서 색 관리 효과에만 사용할 수 있습니다. (오버로드 2/2)
CreateColorContextFromSimpleColorProfile 간단한 색 프로필에서 색 컨텍스트를 만듭니다. '최적' 모드에서 색 관리 효과에만 사용할 수 있습니다. (오버로드 1/2)

CreateColorContextFromWicColorContext IWICColorContext에서 색 컨텍스트를 만듭니다. 결과 컨텍스트의 D2D1ColorContext 공간은 다양합니다. 자세한 내용은 비고 섹션을 참조하세요. (ID2D1DeviceContext.CreateColorContextFromWicColorContext)
CreateColorContextFromWicColorContext IWICColorContext에서 색 컨텍스트를 만듭니다. 결과 컨텍스트의 D2D1ColorContext 공간은 다양합니다. 자세한 내용은 비고 섹션을 참조하세요. (ID2D1EffectContext.CreateColorContextFromWicColorContext)
CreateCommandList ID2D1CommandList 개체를 만듭니다.
CreateCompatibleRenderTarget 현재 렌더링 대상과 호환되는 중간 오프스크린 그리기 중에 사용할 새 비트맵 렌더링 대상을 만듭니다.
CreateCompatibleRenderTarget 현재 렌더링 대상과 호환되는 중간 오프스크린 그리기 중에 사용할 비트맵 렌더링 대상을 만듭니다. (오버로드 2/2)
CreateCompatibleRenderTarget 현재 렌더링 대상과 호환되는 중간 오프스크린 그리기 중에 사용할 비트맵 렌더링 대상을 만듭니다. (오버로드 1/2)
CreateCompatibleRenderTarget 현재 렌더링 대상과 호환되는 중간 오프스크립 드로잉 중에 사용할 비트맵 렌더링 대상을 만듭니다. 새 비트맵 렌더링 대상은 현재 렌더링 대상과 동일한 픽셀 형식(알파 모드는 아님)을 가집니다.
CreateCompatibleRenderTarget 현재 렌더링 대상과 호환되고 현재 렌더링 대상과 동일한 픽셀 형식(알파 모드 아님)을 갖는 중간 오프스크린 그리기 중에 사용할 새 비트맵 렌더링 대상을 만듭니다.
CreateCompatibleRenderTarget 현재 렌더링 대상과 호환되며 현재 렌더링 대상과 크기, DPI 및 픽셀 형식(알파 모드는 아님)을 갖는 중간 오프스크린 그리기 중에 사용할 새 비트맵 렌더링 대상을 만듭니다.
CreateDCRenderTarget Windows GDI(그래픽 디바이스 인터페이스) 디바이스 컨텍스트에 그리는 렌더링 대상을 만듭니다.

CreateDevice ID2D1Device 개체를 만듭니다.
CreateDevice ID2D1Device2 개체를 만듭니다.
CreateDevice ID2D1Device3 개체를 만듭니다.
CreateDevice ID2D1Device4 개체를 만듭니다.
CreateDevice 지정된 IDXGIDevice에서 새 Direct2D 디바이스를 만듭니다. (ID2D1Factory6.CreateDevice)
CreateDevice 지정된 IDXGIDevice에서 새 Direct2D 디바이스를 만듭니다. (ID2D1Factory7.CreateDevice)
CreateDevice ID2D1Device1 개체를 만듭니다.
CreateDeviceContext Direct2D 디바이스에서 새 디바이스 컨텍스트를 만듭니다.
CreateDeviceContext Direct2D 디바이스에서 새 ID2D1DeviceContext2를 만듭니다.
CreateDeviceContext 이 Direct2D 디바이스에서 새 ID2D1DeviceContext3을 만듭니다.
CreateDeviceContext 이 Direct2D 디바이스에서 새 ID2D1DeviceContext4를 만듭니다.
CreateDeviceContext 처음에 할당된 대상이 없는 새 디바이스 컨텍스트를 만듭니다. (ID2D1Device5.CreateDeviceContext)

CreateDeviceContext 처음에 할당된 대상이 없는 새 디바이스 컨텍스트를 만듭니다. (ID2D1Device6.CreateDeviceContext)
CreateDrawingStateBlock 렌더링 대상의 SaveDrawingState 및 RestoreDrawingState 메서드와 함께 사용할 수 있는 ID2D1DrawingStateBlock을 만듭니다. (오버로드 1/3)
CreateDrawingStateBlock 렌더링 대상의 SaveDrawingState 및 RestoreDrawingState 메서드와 함께 사용할 수 있는 ID2D1DrawingStateBlock을 만듭니다. (오버로드 2/3)
CreateDrawingStateBlock 렌더링 대상의 SaveDrawingState 및 RestoreDrawingState 메서드와 함께 사용할 수 있는 ID2D1DrawingStateBlock을 만듭니다. (오버로드 3/3)
CreateDrawingStateBlock 새 그리기 상태 블록을 만듭니다. 이 블록은 렌더링 대상의 후속 SaveDrawingState 및 RestoreDrawingState 작업에서 사용할 수 있습니다. (오버로드 1/3)
CreateDrawingStateBlock 새 그리기 상태 블록을 만듭니다. 이 블록은 렌더링 대상의 후속 SaveDrawingState 및 RestoreDrawingState 작업에서 사용할 수 있습니다. (오버로드 3/3)
CreateDrawingStateBlock 새 그리기 상태 블록을 만듭니다. 이 블록은 렌더링 대상의 후속 SaveDrawingState 및 RestoreDrawingState 작업에서 사용할 수 있습니다. (오버로드 2/3)
CreateDxgiSurfaceRenderTarget DXGI(DirectX Graphics Infrastructure) 화면에 그리는 렌더링 대상을 만듭니다. (오버로드 1/2)
CreateDxgiSurfaceRenderTarget DXGI(DirectX Graphics Infrastructure) 화면에 그리는 렌더링 대상을 만듭니다. (오버로드 2/2)
CreateEffect 지정된 클래스 ID에 대한 효과를 만듭니다.

CreateEffect 지정된 클래스 ID에 대한 Direct2D 효과를 만듭니다.
CreateEllipseGeometry ID2D1EllipseGeometry를 만듭니다. (오버로드 2/2)
CreateEllipseGeometry ID2D1EllipseGeometry를 만듭니다. (오버로드 1/2)
CreateFilledGeometryRealization 이후에 렌더링할 수 있는 기하 도형 채우기의 디바이스 종속 표현을 만듭니다.
CreateGdiMetafile 메타파일 콘텐츠를 재생하는 데 사용할 수 있는 새 ID2D1GdiMetafile 개체를 만듭니다.
CreateGeometryGroup 다른 기하 도형을 보유하는 개체인 ID2D1GeometryGroup을 만듭니다.
CreateGradientMesh 지정된 패치 배열을 사용하여 새 ID2D1GradientMesh 인스턴스를 만듭니다.
CreateGradientStopCollection 지정된 D2D1_GRADIENT_STOP 구조체 배열에서 ID2D1GradientStopCollection을 만듭니다.
CreateGradientStopCollection D2D1_GAMMA_2_2 색 보간 감마 및 클램프 확장 모드를 사용하는 지정된 그라데이션 정지점에서 ID2D1GradientStopCollection을 만듭니다.
CreateGradientStopCollection 그라데이션 중지 컬렉션을 만들어 그라데이션에 [0,1] 이외의 값이 있는 색 채널을 포함할 수 있도록 하고 sRGB 공간에서 보간이 있는 고색 렌더링 대상에 렌더링할 수도 있습니다.
CreateHwndRenderTarget 창에 렌더링되는 렌더링 대상인 ID2D1HwndRenderTarget을 만듭니다. (오버로드 2/2)
CreateHwndRenderTarget 창에 렌더링되는 렌더링 대상인 ID2D1HwndRenderTarget을 만듭니다. (오버로드 1/2)

CreateImageBrush 이미지 브러시를 만듭니다. 입력 이미지는 비트맵, 효과 또는 명령 목록을 포함하여 모든 유형의 이미지일 수 있습니다. (오버로드 2/3)
CreateImageBrush 이미지 브러시를 만듭니다. 입력 이미지는 비트맵, 효과 또는 명령 목록을 포함하여 모든 유형의 이미지일 수 있습니다. (오버로드 3/3)
CreateImageBrush 이미지 브러시를 만듭니다. 입력 이미지는 비트맵, 효과 또는 명령 목록을 포함하여 모든 유형의 이미지일 수 있습니다. (오버로드 1/3)
CreateImageSourceFromDxgi DXGI 표면 집합에서 이미지 원본을 만듭니다. YCbCr 표면은 후속 그리기 중에 자동으로 RGBA로 변환됩니다.
CreateImageSourceFromWic 이미지 원본 내의 모든 픽셀 메모리를 채우는 동안 WIC 비트맵 원본에서 이미지 원본 개체를 만듭니다. 최소한의 메모리를 사용하는 동안 이미지가 로드되고 저장됩니다. (오버로드 3/3)
CreateImageSourceFromWic 이미지 원본 내의 모든 픽셀 메모리를 채우는 동안 WIC 비트맵 원본에서 이미지 원본 개체를 만듭니다. 최소한의 메모리를 사용하는 동안 이미지가 로드되고 저장됩니다. (오버로드 2/3)
CreateImageSourceFromWic 이미지 원본 내의 모든 픽셀 메모리를 채우는 동안 WIC 비트맵 원본에서 이미지 원본 개체를 만듭니다. 최소한의 메모리를 사용하는 동안 이미지가 로드되고 저장됩니다. (오버로드 1/3)
CreateInk 지정된 지점에서 시작하는 새 ID2D1Ink 개체를 만듭니다. (오버로드 1/2)
CreateInk 지정된 지점에서 시작하는 새 ID2D1Ink 개체를 만듭니다. (오버로드 2/2)
CreateInkStyle DrawInk와 같은 잉크 렌더링 메서드에 사용할 새 ID2D1InkStyle 개체를 만듭니다. (오버로드 1/2)

CreateInkStyle DrawInk와 같은 잉크 렌더링 메서드에 사용할 새 ID2D1InkStyle 개체를 만듭니다. (오버로드 2/2)
CreateLayer 이 렌더링 대상 및 호환되는 렌더링 대상과 함께 사용할 수 있는 레이어 리소스를 만듭니다. (오버로드 2/2)
CreateLayer 이 렌더링 대상 및 호환되는 렌더링 대상과 함께 사용할 수 있는 레이어 리소스를 만듭니다. 새 계층의 초기 크기가 지정됩니다.
CreateLayer 이 렌더링 대상 및 호환되는 렌더링 대상과 함께 사용할 수 있는 레이어 리소스를 만듭니다. (오버로드 1/2)
CreateLinearGradientBrush 지정된 그라데이션 정지를 포함하고 지정된 변환 및 기본 불투명도를 포함하는 ID2D1LinearGradientBrush를 만듭니다.
CreateLinearGradientBrush 지정된 그라데이션 정지를 포함하고 변환이 없으며 기본 불투명도가 1.0인 ID2D1LinearGradientBrush를 만듭니다.
CreateLinearGradientBrush 선형 그라데이션을 사용하여 영역을 그리기 위한 ID2D1LinearGradientBrush 개체를 만듭니다.
CreateLookupTable3D 3 채널 입력을 3 채널 출력에 매핑하기 위한 3D 조회 테이블을 만듭니다. 테이블 데이터는 4 채널 형식으로 제공되어야 합니다. (ID2D1DeviceContext2.CreateLookupTable3D)
CreateLookupTable3D 3 채널 입력을 3 채널 출력에 매핑하기 위한 3D 조회 테이블을 만듭니다. 테이블 데이터는 4 채널 형식으로 제공되어야 합니다. (ID2D1EffectContext1.CreateLookupTable3D)
CreateMesh 삼각형을 사용하여 도형을 설명하는 메시를 만듭니다.

CreateOffsetTransform 오프셋 변환을 만들고 반환합니다.
CreatePaint 'fill' 또는 'stroke' 속성을 설정하는 데 사용할 수 있는 페인트 개체를 만듭니다. (오버로드 2/2)
CreatePaint 'fill' 또는 'stroke' 속성을 설정하는 데 사용할 수 있는 페인트 개체를 만듭니다. (오버로드 1/2)
CreatePathData 'path' 요소에 'd' 특성을 설정하는 데 사용할 수 있는 경로 데이터 개체를 만듭니다.
CreatePathGeometry 빈 ID2D1PathGeometry를 만듭니다.
CreatePathGeometry ID2D1PathGeometry1 개체를 만듭니다.
CreatePathGeometry 경로 데이터를 나타내는 경로 기하 도형 개체를 만듭니다.
CreatePointCollection 다각형 또는 다각형 요소에 점 특성을 설정하는 데 사용할 수 있는 points 개체를 만듭니다.
CreatePrintControl ID2D1CommandList에 저장된 Direct2D 기본 형식을 고정 페이지 표현으로 변환하는 ID2D1PrintControl 개체를 만듭니다. 그런 다음 인쇄 하위 시스템에서 기본 형식을 사용합니다. (오버로드 2/2)
CreatePrintControl ID2D1CommandList에 저장된 Direct2D 기본 형식을 고정 페이지 표현으로 변환하는 ID2D1PrintControl 개체를 만듭니다. 그런 다음 인쇄 하위 시스템에서 기본 형식을 사용합니다. (오버로드 1/2)
CreateRadialGradientBrush 지정된 그라데이션 정지를 포함하고 지정된 변환 및 기본 불투명도를 포함하는 ID2D1RadialGradientBrush를 만듭니다.

CreateRadialGradientBrush 지정된 그라데이션 정지를 포함하고 변환이 없으며 기본 불투명도가 1.0인 ID2D1RadialGradientBrush를 만듭니다.
CreateRadialGradientBrush 방사형 그라데이션으로 영역을 그리는 데 사용할 수 있는 ID2D1RadialGradientBrush 개체를 만듭니다.
CreateRectangleGeometry ID2D1RectangleGeometry를 만듭니다. (오버로드 2/2)
CreateRectangleGeometry ID2D1RectangleGeometry를 만듭니다. (오버로드 1/2)
CreateResourceTexture 리소스 ID가 지정되었는지 여부에 따라 지정된 리소스 텍스처를 만들거나 찾습니다.
CreateRoundedRectangleGeometry ID2D1RoundedRectangleGeometry를 만듭니다. (오버로드 2/2)
CreateRoundedRectangleGeometry ID2D1RoundedRectangleGeometry를 만듭니다. (오버로드 1/2)
CreateSharedBitmap 데이터가 다른 리소스와 공유되는 ID2D1Bitmap을 만듭니다.
CreateSolidColorBrush 지정한 색과 불투명도가 있는 새 ID2D1SolidColorBrush를 만듭니다.
CreateSolidColorBrush 지정된 색과 기본 불투명도가 1.0f인 새 ID2D1SolidColorBrush를 만듭니다.
CreateSolidColorBrush 단색으로 영역을 그리는 데 사용할 수 있는 새 ID2D1SolidColorBrush를 만듭니다.

CreateSpriteBatch 비어 있는 새 스프라이트 일괄 처리를 만듭니다. 스프라이트 일괄 처리를 만든 후 ID2D1SpriteBatch::AddSprites를 사용하여 스프라이트를 추가한 다음 ID2D1DeviceContext3::DrawSpriteBatch를 사용하여 그립니다.
CreateStrokeDashArray stroke-dasharray 속성을 설정하는 데 사용할 수 있는 대시 배열 개체를 만듭니다.
CreateStrokedGeometryRealization 이후에 렌더링할 수 있는 기하 도형 스트로크의 디바이스 종속 표현을 만듭니다.
CreateStrokeStyle 시작 캡, 대시 패턴 및 스트로크의 기타 기능을 설명하는 ID2D1StrokeStyle을 만듭니다. (오버로드 2/2)
CreateStrokeStyle 시작 캡, 대시 패턴 및 스트로크의 기타 기능을 설명하는 ID2D1StrokeStyle을 만듭니다. (오버로드 1/2)
CreateStrokeStyle ID2D1StrokeStyle1 개체를 만듭니다. (오버로드 1/2)
CreateStrokeStyle ID2D1StrokeStyle1 개체를 만듭니다. (오버로드 2/2)
CreateSvgDocument 스트림에서 SVG 문서를 만듭니다.
CreateSvgGlyphStyle SVG 문자 모양 스타일 개체를 만듭니다.
CreateTransformedGeometry 지정된 기하 도형을 변환하고 결과를 ID2D1TransformedGeometry 개체로 저장합니다. (오버로드 2/2)
CreateTransformedGeometry 지정된 기하 도형을 변환하고 결과를 ID2D1TransformedGeometry 개체로 저장합니다. (오버로드 1/2)

CreateTransformedImageSource 원본과 리소스를 공유하는 이미지 원본을 만듭니다.
CreateTransformNodeFromEffect 효과 그래프를 단일 변환 노드로 래핑한 다음 변환 그래프에 삽입합니다. 이렇게 하면 효과가 다른 효과를 집계할 수 있습니다.
CreateVertexBuffer 꼭짓점 버퍼를 만들거나 표준 꼭짓점 버퍼를 찾고 필요에 따라 꼭짓점을 사용하여 초기화합니다.
CreateWicBitmapRenderTarget Microsoft WIC(Windows 이미징 구성 요소) 비트맵으로 렌더링되는 렌더링 대상을 만듭니다. (오버로드 2/2)
CreateWicBitmapRenderTarget Microsoft WIC(Windows 이미징 구성 요소) 비트맵으로 렌더링되는 렌더링 대상을 만듭니다. (오버로드 1/2)
CreationProperties 루트 수준 만들기 세부 정보를 설명하는 D2D1_CREATION_PROPERTIES 반환합니다.
D2D1ComputeMaximumScaleFactor 지정된 변환이 벡터를 확장할 수 있는 최대 인수를 계산합니다.
D2D1ConvertColorSpace 지정된 색을 한 색 영역에서 다른 색으로 변환합니다.
D2D1CreateDevice 제공된 DXGI 디바이스와 연결된 새 Direct2D 디바이스를 만듭니다.
D2D1CreateDeviceContext DXGI 표면과 연결된 새 Direct2D 디바이스 컨텍스트를 만듭니다.
D2D1CreateFactory D2D1CreateFactory(D2D1_FACTORY_TYPE,REFIID,D2D1_FACTORY_OPTIONS*,void**)는 Direct2D 리소스를 만드는 데 사용할 수 있는 팩터리 개체를 만듭니다.

D2D1CreateFactory D2D1CreateFactory(D2D1_FACTORY_TYPE,REFIID,void**) 함수는 Direct2D 리소스를 만드는 데 사용할 수 있는 팩터리 개체를 만듭니다.
D2D1GetGradientMeshInteriorPointsFromCoonsPatch Coons 패치를 정의하는 점을 기반으로 그라데이션 메시 패치의 내부 점을 반환합니다.
D2D1InvertMatrix 지정된 행렬을 반전하려고 시도합니다.
D2D1IsMatrixInvertible 지정된 행렬을 반전할 수 있는지 여부를 나타냅니다.
D2D1MakeRotateMatrix 지정된 지점에 대해 지정된 각도로 회전하는 회전 변환을 만듭니다.
D2D1MakeSkewMatrix 지정된 x축 각도, y축 각도 및 중심점을 갖는 기울이기 변환을 만듭니다.
D2D1SinCos 각도의 사인과 코사인을 반환합니다.
D2D1Tan 각도의 탄젠트를 반환합니다.
D2D1Vec3Length 3차원 벡터의 길이를 반환합니다.
역직렬화 스트림에서 하위 트리를 역직렬화합니다. 스트림에는 루트 요소가 하나만 있어야 하지만 해당 루트 요소는 'svg' 요소가 될 필요가 없습니다. 출력 요소가 이 문서 트리에 삽입되지 않습니다.
결정 행렬의 결정자를 계산합니다. (Matrix3x2F.Determinant)
결정 행렬의 결정자를 계산합니다. (Matrix4x4F.Determinant)

그릴 GPU(그래픽 처리 장치) 기반 Direct2D 파이프라인으로 변환을 그립니다.
DrawBitmap 지정된 사각형의 크기로 크기를 조정한 후 지정된 비트맵을 그립니다. (오버로드 1/3)
DrawBitmap 지정된 사각형의 크기로 크기를 조정한 후 지정된 비트맵을 그립니다. (오버로드 2/3)
DrawBitmap 지정된 사각형의 크기로 크기를 조정한 후 지정된 비트맵을 그립니다. (오버로드 3/3)
DrawBitmap 렌더링 대상에 비트맵을 그립니다. (ID2D1CommandSink.DrawBitmap)
DrawBitmap 렌더링 대상에 비트맵을 그립니다. (오버로드 2/5)
DrawBitmap 렌더링 대상에 비트맵을 그립니다. (오버로드 4/5)
DrawBitmap 렌더링 대상에 비트맵을 그립니다. (오버로드 3/5)
DrawBitmap 렌더링 대상에 비트맵을 그립니다. (오버로드 1/5)
DrawColorBitmapGlyphRun 비트맵 형식 중 하나를 사용하여 색 비트맵 문자 모양 실행을 그립니다.
DrawEllipse 지정한 스트로크 스타일을 사용하여 지정한 타원의 윤곽선을 그립니다.
DrawEllipse 지정된 차원과 스트로크를 사용하여 타원의 윤곽선을 그립니다.

DrawGdiMetafile 디바이스 컨텍스트에 메타파일을 그립니다. (ID2D1CommandSink.DrawGdiMetafile)
DrawGdiMetafile 디바이스 컨텍스트에 메타파일을 그립니다. (오버로드 1/3)
DrawGdiMetafile 디바이스 컨텍스트에 메타파일을 그립니다. (오버로드 2/3)
DrawGdiMetafile 지정된 원본 및 대상 사각형을 사용하여 명령 싱크에 메타파일을 그립니다.
DrawGdiMetafile 지정된 원본 및 대상 사각형을 사용하여 디바이스 컨텍스트에 메타파일을 그립니다. (오버로드 1/3)
DrawGdiMetafile 지정된 원본 및 대상 사각형을 사용하여 디바이스 컨텍스트에 메타파일을 그립니다. (오버로드 3/3)
DrawGdiMetafile 지정된 원본 및 대상 사각형을 사용하여 디바이스 컨텍스트에 메타파일을 그립니다. (오버로드 2/3)
DrawGeometry 지정된 스트로크 스타일을 사용하여 지정된 기하 도형의 윤곽선을 그립니다.
DrawGeometry 명령 싱크에 그릴 기하 도형을 나타냅니다.
DrawGeometryRealization 지정된 브러시를 사용하여 지정된 기하 도형 실현을 대상에 렌더링합니다.
DrawGlyphRun 지정된 문자 모양을 그립니다.

DrawGlyphRun 그릴 문자 모양을 나타냅니다.
DrawGlyphRun 디바이스 컨텍스트에 일련의 문자 모양을 그립니다.
DrawGradientMesh 지정된 그라데이션 메시를 대상에 렌더링합니다. (ID2D1CommandSink2.DrawGradientMesh)
DrawGradientMesh 지정된 그라데이션 메시를 대상에 렌더링합니다. (ID2D1DeviceContext2.DrawGradientMesh)
Drawimage 제공된 이미지를 명령 싱크에 그립니다.
Drawimage 디바이스 컨텍스트에 이미지를 그립니다. (오버로드 8/8)
Drawimage 디바이스 컨텍스트에 이미지를 그립니다. (오버로드 1/8)
Drawimage 디바이스 컨텍스트에 이미지를 그립니다. (오버로드 7/8)
Drawimage 디바이스 컨텍스트에 이미지를 그립니다. (오버로드 2/8)
Drawimage 디바이스 컨텍스트에 이미지를 그립니다. (오버로드 6/8)
Drawimage 디바이스 컨텍스트에 이미지를 그립니다. (오버로드 4/8)
Drawimage 디바이스 컨텍스트에 이미지를 그립니다. (오버로드 3/8)

DrawImage 디바이스 컨텍스트에 이미지를 그립니다. (오버로드 5/8)
DrawingStateDescription D2D1_DRAWING_STATE_DESCRIPTION 구조를 만듭니다.
DrawingStateDescription1 D2D1_DRAWING_STATE_DESCRIPTION1 구조를 만듭니다.
DrawInk 지정된 브러시 및 잉크 스타일을 사용하여 지정된 잉크 개체를 렌더링합니다. (ID2D1CommandSink2.DrawInk)
DrawInk 지정된 브러시 및 잉크 스타일을 사용하여 지정된 잉크 개체를 렌더링합니다. (ID2D1DeviceContext2.DrawInk)
DrawLine 지정한 스트로크 스타일을 사용하여 지정된 점 사이에 선을 그립니다.
DrawLine 두 점 사이에 그려진 선을 그립니다.
DrawRectangle 지정한 차원과 스트로크 스타일이 있는 사각형의 윤곽선을 그립니다. (오버로드 2/2)
DrawRectangle 지정된 차원 및 스트로크 스타일이 있는 사각형의 윤곽선을 그립니다. (오버로드 1/2)
DrawRectangle 사각형을 그립니다.
DrawRoundedRectangle 지정된 스트로크 스타일을 사용하여 지정된 둥근 사각형의 윤곽선을 그립니다. (오버로드 1/2)
DrawRoundedRectangle 지정된 스트로크 스타일을 사용하여 지정된 둥근 사각형의 윤곽선을 그립니다. (오버로드 2/2)

DrawSpriteBatch 지정된 그리기 옵션을 사용하여 지정된 스프라이트 일괄 처리의 일부 또는 전체를 디바이스 컨텍스트에 렌더링합니다. (ID2D1CommandSink3.DrawSpriteBatch)
DrawSpriteBatch 지정된 그리기 옵션을 사용하여 지정된 스프라이트 일괄 처리의 모든 스프라이트를 디바이스 컨텍스트에 렌더링합니다.
DrawSpriteBatch 지정된 그리기 옵션을 사용하여 지정된 스프라이트 일괄 처리의 일부 또는 전체를 디바이스 컨텍스트에 렌더링합니다. (오버로드 1/2)
DrawSvgDocument SVG 문서를 그립니다.
DrawSvgGlyphRun DWRITE_GLYPH_IMAGE_FORMATS_SVG 형식의 색 문자 모양 실행을 그립니다.
Drawtext IDWriteTextFormat 개체에서 제공하는 서식 정보를 사용하여 지정된 텍스트를 그립니다. (오버로드 1/2)
Drawtext IDWriteTextFormat 개체에서 제공하는 서식 정보를 사용하여 지정된 텍스트를 그립니다. (오버로드 2/2)
Drawtext 지정된 레이아웃 사각형 내에 텍스트를 그립니다. (오버로드 2/2)
Drawtext 지정된 레이아웃 사각형 내에 텍스트를 그립니다. (오버로드 1/2)
DrawTextLayout 지정된 IDWriteTextLayout 개체에서 설명하는 서식이 지정된 텍스트를 그립니다.
DrawTextLayout 텍스트 레이아웃 개체를 그립니다. 레이아웃이 이후에 변경되지 않으면 동일한 레이아웃을 반복적으로 그릴 때 DrawText보다 더 효율적일 수 있습니다.

EffectInputDescription D2D1_EFFECT_INPUT_DESCRIPTION 구조를 만듭니다.
타원 D2D1_ELLIPSE 구조를 만듭니다.
EndDraw 렌더링 대상에 대한 그리기 작업을 종료하고 현재 오류 상태 및 관련 태그를 나타냅니다.
EndDraw ID2D1CommandSink 처리가 완료된 시기를 나타냅니다.
EndFigure 현재 그림을 종료합니다. 필요에 따라 닫습니다.
EnsureCached 이미지 원본 캐시의 지정된 지역이 채워지도록 합니다. (오버로드 1/2)
EnsureCached 이미지 원본 캐시의 지정된 지역이 채워지도록 합니다. (오버로드 2/2)
Enter Direct2D API 중요 섹션(있는 경우)을 입력합니다.
FillContainsPoint 기하 도형으로 채워진 영역에 지정된 점이 포함되는지 여부를 나타냅니다.
FillContainsPoint 지정된 평면화 허용 오차가 지정된 경우 기하 도형으로 채워진 영역에 지정된 점이 포함되는지 여부를 나타냅니다. (오버로드 1/2)
FillContainsPoint 이 기하 도형으로 채워진 영역에 지정된 점이 포함되는지 여부를 나타냅니다.
FillContainsPoint 지정된 평면화 허용 오차가 지정된 경우 기하 도형으로 채워진 영역에 지정된 점이 포함되는지 여부를 나타냅니다. (오버로드 2/2)

FillEllipse 지정된 타원의 내부를 그립니다. (오버로드 1/2)
FillEllipse 지정된 타원의 내부를 그립니다. (오버로드 2/2)
FillGeometry 지정된 기하 도형의 내부를 그립니다.
FillGeometry 입력할 기하 도형을 명령 싱크에 나타냅니다.
FillMesh 지정된 메시의 내부를 그립니다.
FillMesh 명령 싱크에서 채울 메시를 나타냅니다.
FillOpacityMask 지정된 비트맵에서 설명하는 불투명 마스크를 브러시에 적용하고 해당 브러시를 사용하여 렌더링 대상 영역을 그립니다. (오버로드 1/2)
FillOpacityMask 지정된 비트맵에서 설명하는 불투명 마스크를 브러시에 적용하고 해당 브러시를 사용하여 렌더링 대상 영역을 그립니다. (오버로드 2/2)
FillOpacityMask 명령 싱크에서 불투명 마스크를 채웁니다.
FillOpacityMask 제공된 불투명 마스크 비트맵의 알파 채널을 사용하여 채웁니다. 브러시 불투명도는 마스크에 의해 변조됩니다. 렌더링 대상 앤티앨리어싱 모드를 별칭으로 설정해야 합니다. (오버로드 2/3)
FillOpacityMask 제공된 불투명 마스크 비트맵의 알파 채널을 사용하여 채웁니다. 브러시 불투명도는 마스크에 의해 변조됩니다. 렌더링 대상 앤티앨리어싱 모드를 별칭으로 설정해야 합니다. (오버로드 1/3)

FillOpacityMask 제공된 불투명 마스크 비트맵의 알파 채널을 사용하여 채웁니다. 브러시 불투명도는 마스크에 의해 변조됩니다. 렌더링 대상 앤티앨리어싱 모드를 별칭으로 설정해야 합니다. (오버로드 3/3)
FillRectangle 지정된 사각형의 내부를 그립니다. (오버로드 1/2)
FillRectangle 지정된 사각형의 내부를 그립니다. (오버로드 2/2)
FillRectangle 입력할 사각형을 명령 싱크에 나타냅니다.
FillRoundedRectangle 지정된 둥근 사각형의 내부를 그립니다. (오버로드 1/2)
FillRoundedRectangle 지정된 둥근 사각형의 내부를 그립니다. (오버로드 2/2)
FindElementById 지정된 ID를 가진 SVG 요소를 가져옵니다.
FindResourceTexture 동일한 GUID를 사용하여 ID2D1EffectContext::CreateResourceTexture를 사용하여 이미 만들어진 경우 지정된 리소스 텍스처를 찾습니다.
FindVertexBuffer 동일한 GUID를 사용하여 ID2D1EffectContext::CreateVertexBuffer를 사용하여 이미 만들어진 경우 지정된 꼭짓점 버퍼를 찾습니다.
FloatMax 최대 부동 소수점 값을 반환합니다.
플러시 보류 중인 모든 그리기 명령을 실행합니다.

FlushDeviceContexts 지정된 비트맵을 참조하는 모든 디바이스 컨텍스트를 플러시합니다.
GetAntialiasMode 텍스트가 아닌 그리기 작업에 대한 현재 앤티앨리어싱 모드를 검색합니다.
GetAttributeValue 이 요소의 특성을 문자열로 가져옵니다. (오버로드 2/2)
GetAttributeValue 이 요소의 특성을 확장 모드 값으로 가져옵니다. 이 메서드는 spreadMethod 특성의 값을 가져오는 데 사용할 수 있습니다.
GetAttributeValue 채우기 모드로 이 요소의 특성을 가져옵니다. 이 메서드는 채우기 규칙 또는 클립 규칙 속성의 값을 가져오는 데 사용할 수 있습니다.
GetAttributeValue 이 요소의 특성을 행렬 값으로 가져옵니다. 이 메서드는 변환 또는 gradientTransform 특성의 값을 가져오는 데 사용할 수 있습니다.
GetAttributeValue 이 요소의 특성을 POD 형식으로 가져옵니다.
GetAttributeValue 이 요소의 특성을 문자열로 가져옵니다. (오버로드 1/2)
GetAttributeValue 이 요소의 특성을 표시 값으로 가져옵니다. 이 메서드는 표시 속성의 값을 가져오는 데 사용할 수 있습니다. (오버로드 1/2)
GetAttributeValue 이 요소의 특성을 길이 값으로 가져옵니다.
GetAttributeValue 이 요소의 특성을 줄 상한 값으로 가져옵니다. 이 메서드는 stroke-linecap 속성의 값을 가져오는 데 사용할 수 있습니다.

GetAttributeValue 이 요소의 특성을 줄 조인 값으로 가져옵니다. 이 메서드는 stroke-linejoin 속성의 값을 가져오는 데 사용할 수 있습니다.
GetAttributeValue 이 요소의 특성을 오버플로 값으로 가져옵니다. 이 메서드는 오버플로 속성의 값을 가져오는 데 사용할 수 있습니다.
GetAttributeValue 가로 세로 비율 유지 값으로 이 요소의 특성을 가져옵니다. 이 메서드는 preserveAspectRatio 특성의 값을 가져오는 데 사용할 수 있습니다.
GetAttributeValue 이 요소의 특성을 단위 형식 값으로 가져옵니다. 이 메서드는 gradientUnits 또는 clipPathUnits 특성의 값을 가져오는 데 사용할 수 있습니다.
GetAttributeValue 이 요소의 특성을 표시 유형 값으로 가져옵니다. 이 메서드는 표시 유형 속성의 값을 가져오는 데 사용할 수 있습니다.
GetAttributeValue 이 요소의 특성을 float로 가져옵니다.
GetAttributeValue 이 요소의 특성을 가져옵니다.
GetAttributeValue 이 요소의 특성을 페인트로 가져옵니다. 이 메서드는 채우기 또는 스트로크 속성의 값을 가져오는 데 사용할 수 있습니다.
GetAttributeValue 이 요소의 특성을 경로 데이터로 가져옵니다. 이 메서드는 경로 요소에서 d 특성의 값을 가져오는 데 사용할 수 있습니다.
GetAttributeValue 이 요소의 특성을 포인트로 가져옵니다. 이 메서드는 다각형 또는 다각형 요소에 점 특성의 값을 가져오는 데 사용할 수 있습니다.

GetAttributeValue 이 요소의 특성을 스트로크 대시 배열로 가져옵니다. 이 메서드는 stroke-dasharray 속성의 값을 가져오는 데 사용할 수 있습니다.
GetAttributeValue 인터페이스 형식으로 이 요소의 특성을 가져옵니다.
GetAttributeValueLength 이 요소의 특성에 대한 문자열 길이를 가져옵니다.
GetBitmap 이 브러시가 그리는 데 사용하는 비트맵 소스를 가져옵니다.
GetBitmap 이 렌더링 대상의 비트맵을 검색합니다. 반환된 비트맵은 그리기 작업에 사용할 수 있습니다.
GetBounds 기하 도형의 경계를 검색합니다. (오버로드 2/2)
GetBounds 기하 도형의 경계를 검색합니다. (오버로드 1/2)
GetBounds 메타파일의 헤더에 보고된 메타파일의 범위를 DIP(디바이스 독립적 픽셀)로 가져옵니다.
GetBounds 선택적 적용된 변환을 사용하여 기하 도형의 경계를 검색합니다.
GetBufferPrecision 그라데이션 버퍼의 전체 자릿수를 가져옵니다.
GetCenter 그라데이션 타원의 중심을 검색합니다.
GetColor 단색 브러시의 색을 검색합니다.

GetColor 페인트 형식이 D2D1_SVG_PAINT_TYPE_COLOR 경우에 사용되는 페인트 색을 가져옵니다.
GetColorBitmapGlyphImage 색 문자 모양 캐시에서 색 비트맵 문자 모양 이미지를 검색합니다.
GetColorContext 비트맵과 연결된 색 컨텍스트 정보를 가져옵니다.
GetColorContextType 색 컨텍스트 형식을 검색합니다. (ID2D1ColorContext1.GetColorContextType)
GetColorInterpolationGamma 그라데이션 중지점이 보간되는 감마 공간을 나타냅니다.
GetColorInterpolationMode 그라데이션 중지 컬렉션에서 사용하는 색 보간 모드를 검색합니다.
GetColorSpace 색 컨텍스트의 색 공간을 가져옵니다.
GetCommands 명령 배열에서 명령을 가져옵니다.
GetCommandsCount 명령 배열의 크기를 가져옵니다.
GetDashCap 각 대시의 끝을 그리는 방법을 지정하는 값을 가져옵니다.
GetDashes 대시 패턴을 지정된 배열에 복사합니다.
GetDashes 배열에서 대시를 가져옵니다. (오버로드 2/2)

GetDashes 배열에서 대시를 가져옵니다. (오버로드 1/2)
GetDashesCount 대시 배열의 항목 수를 검색합니다.
GetDashesCount 배열의 대시 수를 가져옵니다.
GetDashOffset 스트로크가 시작되는 대시 시퀀스의 간격을 지정하는 값을 검색합니다.
GetDashStyle 스트로크의 대시 패턴을 설명하는 값을 가져옵니다.
Getdc 이 렌더링 대상과 연결된 디바이스 컨텍스트를 검색합니다.
GetDescription 그리기 상태의 앤티앨리어싱 모드, 변환 및 태그 부분을 검색합니다.
GetDescription 그리기 상태의 앤티앨리어싱 모드, 변환, 태그, 기본 혼합 및 단위 모드 부분을 가져옵니다.
GetDescription 해당 blend 변환 개체의 혼합 설명을 가져옵니다.
GetDesktopDpi 현재 데스크톱 DPI(인치당 점 수)를 검색합니다. 이 값을 새로 고치려면 ReloadSystemMetrics를 호출합니다.
GetDevice 디바이스 컨텍스트와 연결된 디바이스를 가져옵니다.
GetDocument 이 요소가 포함된 문서를 가져옵니다.

GetDpi 비트맵의 DPI(인치당 점)를 반환합니다.
GetDpi 렌더링 대상의 DPI(인치당 점)를 반환합니다.
GetDpi 효과가 dpi(인치당 도트 수) 또는 픽셀에 있을 수 있는 속성에 사용할 단위 매핑을 가져옵니다.
GetDpi 메타파일에서 보고한 DPI를 가져옵니다.
GetDXGIColorSpace 이 컨텍스트의 DXGI 색 공간을 검색합니다. 색 컨텍스트 형식이 ICC인 경우 DXGI_COLOR_SPACE_CUSTOM 반환합니다.
GetDxgiDevice 이 Direct2D 디바이스와 연결된 DXGI 디바이스를 반환합니다.
GetEffectInvalidRectangleCount 효과에 누적된 잘못된 출력 사각형의 수를 가져옵니다.
GetEffectInvalidRectangles 마지막으로 효과가 그려지고 EndDraw가 디바이스 컨텍스트에서 호출된 이후 누적된 잘못된 사각형을 가져옵니다.
GetEffectProperties 효과의 속성을 검색합니다.
GetEffectRequiredInputRectangles 지정된 출력 사각형을 생성하기 위해 호출자가 제공하는 데 필요한 입력 사각형을 반환합니다.
GetElement 이 특성이 설정된 요소를 반환합니다. 특성이 요소에 설정되지 않은 경우 null을 반환합니다.
GetEllipse 이 타원 기하 도형을 설명하는 D2D1_ELLIPSE 구조를 가져옵니다.

GetEndCap 스트로크의 끝에 사용되는 셰이프의 형식을 검색합니다.
GetEndPoint 선형 그라데이션의 끝 좌표를 검색합니다.
GetExtendMode 정규화된 그라데이션 범위를 벗어난 그라데이션의 동작을 나타냅니다.
GetExtendModeX 브러시가 비트맵을 지나 확장되는 영역을 가로로 타일로 배열하는 메서드를 가져옵니다.
GetExtendModeX x축에 있는 이미지 브러시의 확장 모드를 가져옵니다.
GetExtendModeX x 방향으로 확장 모드를 가져옵니다.
GetExtendModeY 브러시가 비트맵을 지나 확장되는 영역을 세로로 타일하는 메서드를 가져옵니다.
GetExtendModeY 이미지의 y축에 있는 이미지 브러시의 확장 모드를 가져옵니다.
GetExtendModeY y 방향으로 확장 모드를 가져옵니다.
GetFactory 이 리소스와 연결된 팩터리를 검색합니다.
GetFigureCount 경로 기하 도형의 그림 수를 검색합니다.
GetFill 요청된 채우기 매개 변수를 반환합니다.

GetFillMode 이 기하 도형 그룹에 포함된 기하 도형의 교차 영역을 결합하는 방법을 나타냅니다.
GetFirstChild 이 요소의 첫 번째 자식 요소를 가져옵니다.
GetGlyphRunWorldBounds 디바이스 컨텍스트 DPI를 사용하여 문자 모양 실행의 DIP에서 세계 공간 범위를 가져옵니다.
GetGradientMeshWorldBounds 지정된 그라데이션 메시의 월드 경계를 반환합니다.
GetGradientOriginOffset 그라데이션 타원의 중심을 기준으로 그라데이션 원점의 오프셋을 검색합니다.
GetGradientStopCollection 이 선형 그라데이션 브러시와 연결된 ID2D1GradientStopCollection을 검색합니다.
GetGradientStopCollection 이 방사형 그라데이션 브러시 개체와 연결된 ID2D1GradientStopCollection을 검색합니다.
GetGradientStopCount 컬렉션의 그라데이션 중지점 수를 검색합니다.
GetGradientStops 컬렉션의 그라데이션 정지를 D2D1_GRADIENT_STOP 구조체 배열로 복사합니다.
GetGradientStops1 컬렉션에서 메모리로 그라데이션 중지를 복사합니다.
GetHwnd 이 렌더링 대상과 연결된 HWND를 반환합니다.
GetId 페인트 서버 역할을 하는 요소 ID를 가져옵니다. 이 ID는 페인트 형식이 D2D1_SVG_PAINT_TYPE_URI 경우에 사용됩니다.

GetIdLength 페인트 서버 역할을 하는 요소 ID의 문자열 길이를 가져옵니다. 이 ID는 페인트 형식이 D2D1_SVG_PAINT_TYPE_URI 경우에 사용됩니다.
GetImage 이미지 브러시와 연결된 이미지를 가져옵니다.
GetImageLocalBounds 컨텍스트의 월드 변환이 적용되지 않은 이미지의 경계를 가져옵니다.
GetImageWorldBounds 컨텍스트의 월드 변환이 적용된 이미지의 경계를 가져옵니다.
GetInput 인덱스별로 지정된 입력 이미지를 가져옵니다.
GetInputCount 효과에 대한 입력 수를 가져옵니다.
GetInputCount 변환 그래프에 대한 입력 수를 반환합니다.
GetInputCount 변환 노드에 대한 입력 수를 가져옵니다.
GetInterpolationMode 브러시 비트맵의 크기를 조정하거나 회전할 때 사용되는 보간 메서드를 가져옵니다.
GetInterpolationMode 이미지 브러시의 보간 모드를 가져옵니다.
GetInterpolationMode1 브러시의 현재 보간 모드를 반환합니다.
GetLastChild 이 요소의 마지막 자식 요소를 가져옵니다.

GetLineJoin 셰이프 윤곽선의 꼭짓점에서 사용되는 조인트 유형을 검색합니다.
GetMaximumBitmapSize 렌더링 대상에서 지원하는 하나의 비트맵 차원의 최대 크기(픽셀)를 가져옵니다.
GetMaximumColorGlyphCacheMemory 색 문자 모양 캐시의 최대 용량을 가져옵니다.
GetMaximumSupportedFeatureLevel 이는 디바이스에서 지원하는 제공된 목록의 최대 기능 수준을 나타냅니다.
GetMaximumTextureMemory Direct2D가 이미지 캐시 및 캐시된 텍스처 할당을 제거하기 전에 누적되는 최대 텍스처 메모리 양을 설정합니다. (ID2D1Device.GetMaximumTextureMemory)
GetMiterLimit 미터 길이에서 스트로크 두께의 절반까지의 비율에 대한 제한을 검색합니다.
GetMultithreadProtected D2D1_FACTORY_TYPE_MULTI_THREADED 플래그를 사용하여 Direct2D 팩터리를 만들었는지 여부를 반환합니다.
GetNextChild referenceChild 요소의 다음 형제를 가져옵니다.
GetNibShape 이 스타일의 미리 변환된 nib 셰이프를 검색합니다.
GetNibTransform 이 스타일의 nib 셰이프에 적용할 변환을 검색합니다.
GetOffset 현재 오프셋 변환에 있는 오프셋을 가져옵니다.
GetOpacity 이 브러시의 불투명도를 가져옵니다.

GetOptions 비트맵을 만드는 데 사용되는 옵션을 가져옵니다.
GetOutput 효과에서 출력 이미지를 가져옵니다.
GetOutputBounds 지원 변환의 출력 사각형을 반환합니다.
GetPaintType 페인트 형식을 가져옵니다.
GetParent 부모 요소를 가져옵니다.
GetPatchCount 이 그라데이션 메시를 구성하는 패치 수를 반환합니다.
GetPatches 이 그라데이션 메시를 구성하는 패치의 하위 집합을 반환합니다.
GetPixelFormat 비트맵의 픽셀 형식 및 알파 모드를 검색합니다.
GetPixelFormat 렌더링 대상의 픽셀 형식 및 알파 모드를 검색합니다.
GetPixelSize 비트맵의 크기를 디바이스 종속 단위(픽셀)로 반환합니다.
GetPixelSize 렌더링 대상의 크기를 디바이스 픽셀 단위로 반환합니다.
GetPoints 점 배열에서 점을 가져옵니다.

GetPointsCount 배열의 점 수를 가져옵니다.
GetPostInterpolationSpace 보간이 발생한 후 색 공간을 가져옵니다.
GetPreInterpolationSpace 입력 색의 색 공간과 그라데이션 중지점이 보간되는 공간을 가져옵니다.
GetPreviousChild referenceChild 요소의 이전 형제를 가져옵니다.
GetPrimitiveBlend 디바이스 컨텍스트에서 사용되는 현재 설정된 기본 혼합을 반환합니다.
Getprofile ID2D1ColorContext의 색 프로필 바이트를 가져옵니다.
GetProfileSize 비트맵과 연결된 색 프로필의 크기를 가져옵니다.
GetProperties 변환된 이미지 원본을 만들 때 지정된 속성을 검색합니다.
GetPropertyCount 최상위 속성 수를 가져옵니다.
GetPropertyIndex 지정된 속성 이름에 해당하는 인덱스입니다.
GetPropertyName 지정된 인덱스에 해당하는 속성 이름을 가져옵니다. 템플릿 오버로드입니다. 설명 부분을 참조하세요.
GetPropertyName 지정된 인덱스에 해당하는 속성 이름을 가져옵니다.

GetPropertyNameLength 지정된 속성 이름의 문자 수를 가져옵니다. 템플릿 오버로드입니다. 설명 부분을 참조하세요.
GetPropertyNameLength 지정된 속성 이름의 문자 수를 가져옵니다.
GetRadiusX 그라데이션 타원의 x 반경을 검색합니다.
GetRadiusY 그라데이션 타원의 y 반경을 검색합니다.
GetRect 사각형 기하 도형의 차원을 설명하는 사각형을 검색합니다.
GetRegisteredEffects 이 팩터리에 대해 현재 등록된 효과 및 전역 효과의 클래스 ID를 반환합니다.
GetRenderingControls 컨텍스트에 적용된 렌더링 컨트롤을 가져옵니다.
GetRenderingPriority 디바이스의 현재 렌더링 우선 순위를 검색합니다.
GetRoot 문서의 루트 요소를 가져옵니다.
GetRoundedRect 이 둥근 사각형 기하 도형을 설명하는 둥근 사각형을 검색합니다.
GetSegmentCount 경로 기하 도형의 세그먼트 수를 검색합니다.
GetSegmentCount 이 잉크 개체의 세그먼트 수를 반환합니다.

GetSegmentData 세그먼트 데이터 배열에서 데이터를 가져옵니다.
GetSegmentDataCount 세그먼트 데이터 배열의 크기를 가져옵니다.
GetSegments 이 잉크 개체에 저장된 세그먼트의 지정된 하위 집합을 검색합니다.
GetSimpleColorProfile 집합 단순 색 프로필을 검색합니다.
GetSize 비트맵의 크기를 DIP(디바이스 독립적 픽셀)로 반환합니다.
GetSize 디바이스 독립적 픽셀의 레이어 크기를 가져옵니다.
GetSize 렌더링 대상의 크기를 디바이스 독립적 픽셀로 반환합니다.
GetSource WIC(Windows 이미징 구성 요소)에서 기본 비트맵 이미지 원본을 검색합니다.
GetSource 변환된 이미지 원본을 만드는 데 사용되는 원본 이미지를 검색합니다.
GetSourceBounds DIP의 소스 공간에서 메타파일의 범위를 가져옵니다. EMF/EMF+의 프레임 사각형에 해당합니다.
GetSourceGeometries 기하 도형 그룹의 기하 도형을 검색합니다.
GetSourceGeometry 변환된 이 기하 도형 개체의 원본 기하 도형을 검색합니다.

GetSourceGeometryCount 기하 도형 그룹의 기하 도형 개체 수를 나타냅니다.
GetSourceRectangle 이미지 브러시로 그릴 때 이미지의 범위로 사용할 사각형을 가져옵니다.
GetSpecifiedAttributeCount 이 요소의 지정된 특성 수를 반환합니다.
GetSpecifiedAttributeName 지정된 인덱스에서 특성의 이름을 가져옵니다.
GetSpecifiedAttributeNameLength 지정된 인덱스에서 지정된 특성 이름의 문자열 길이를 가져옵니다.
GetSpriteCount 이 스프라이트 일괄 처리의 스프라이트 수를 검색합니다.
GetSprites 이 스프라이트 일괄 처리에서 지정한 스프라이트 하위 집합을 검색합니다. 최상의 성능을 위해 검색할 필요가 없는 속성에 nullptr을 사용합니다.
GetStartCap 스트로크의 시작 부분에 사용되는 셰이프의 형식을 검색합니다.
GetStartPoint 선형 그라데이션의 시작 좌표를 검색합니다.
GetStartPoint 이 잉크 개체의 시작점을 검색합니다.
GetStroke 요청된 스트로크 매개 변수를 반환합니다.
GetStrokeDashesCount 대시 배열의 대시 수를 반환합니다.

GetStrokeTransformType 스트로크 변환 형식을 가져옵니다.
GetSubProperties 인덱스별로 제공된 속성의 하위 속성을 가져옵니다. 템플릿 오버로드입니다.
GetSubProperties 인덱스별로 제공된 속성의 하위 속성을 가져옵니다.
GetSurface 비트맵을 만들 때 지정한 서피스 또는 비트맵을 만들 때 생성된 기본 표면을 가져옵니다.
GetSvgGlyphImage 색 문자 모양 캐시에서 SVG 문자 모양 이미지를 검색합니다.
GetTagName 태그 이름을 가져옵니다.
GetTagNameLength 태그 이름의 문자열 길이를 가져옵니다.
GetTags 후속 그리기 작업에 대한 레이블을 가져옵니다.
GetTarget 현재 디바이스 컨텍스트와 연결된 대상을 가져옵니다.
GetTextAntialiasMode 텍스트 및 문자 모양 그리기 작업에 대한 현재 앤티앨리어싱 모드를 가져옵니다.
GetTextRenderingParams 그리기 상태의 텍스트 렌더링 구성을 검색합니다.
GetTextRenderingParams 렌더링 대상의 현재 텍스트 렌더링 옵션을 검색합니다.

GetTextValue 텍스트 콘텐츠 요소의 값을 가져옵니다.
GetTextValueLength 텍스트 콘텐츠 값의 길이를 가져옵니다.
GetTransform 이 브러시에 적용된 변환을 가져옵니다.
GetTransform 렌더링 대상의 현재 변환을 가져옵니다.
GetTransform ID2D1TransformedGeometry 개체의 원본 기하 도형을 변환하는 데 사용되는 행렬을 검색합니다.
GetType 선택한 속성의 D2D1_PROPERTY_TYPE 가져옵니다. 템플릿 오버로드입니다. 설명 부분을 참조하세요.
GetType 선택한 속성의 D2D1_PROPERTY_TYPE 가져옵니다.
GetUnitMode 디바이스 컨텍스트에서 값을 해석하는 데 사용되는 모드를 가져옵니다.
GetValue 인덱스별로 속성 값을 가져옵니다. 템플릿 오버로드입니다. 설명 부분을 참조하세요. (오버로드 1/2)
GetValue 인덱스별로 지정된 속성의 값을 가져옵니다. 템플릿 오버로드입니다. 설명 부분을 참조하세요.
GetValue 인덱스별로 속성 값을 가져옵니다. 템플릿 오버로드입니다. 설명 부분을 참조하세요. (오버로드 2/2)

GetValue 인덱스별로 지정된 속성의 값을 가져옵니다. (오버로드 1/2)
GetValue 인덱스별로 지정된 속성의 값을 가져옵니다. (오버로드 2/2)
GetValueByName 속성 값을 이름으로 가져옵니다. 템플릿 오버로드입니다. 설명 부분을 참조하세요. (오버로드 2/2)
GetValueByName 속성 값을 이름으로 가져옵니다. (오버로드 2/2)
GetValueByName 속성 값을 이름으로 가져옵니다. (오버로드 1/2)
GetValueByName 속성 값을 이름으로 가져옵니다. 템플릿 오버로드입니다. 설명 부분을 참조하세요. (오버로드 1/2)
GetValueSize 속성 인덱스로 속성 값의 크기를 바이트 단위로 가져옵니다. 템플릿 오버로드입니다. 설명 부분을 참조하세요.
GetValueSize 속성 인덱스로 속성 값의 크기를 바이트 단위로 가져옵니다.
GetViewportSize 초기 뷰포트의 크기를 반환합니다.
GetWidenedBounds 지정한 스트로크 너비와 스타일로 확장되고 지정된 행렬에 의해 변형된 후 기하 도형의 경계를 가져옵니다. (오버로드 4/4)
GetWidenedBounds 지정한 스트로크 너비와 스타일로 확장되고 지정된 행렬에 의해 변형된 후 기하 도형의 경계를 가져옵니다. (오버로드 3/4)

GetWidenedBounds 지정한 스트로크 너비와 스타일로 확장되고 지정된 행렬에 의해 변형된 후 기하 도형의 경계를 가져옵니다. (오버로드 1/4)
GetWidenedBounds 지정한 스트로크 너비와 스타일로 확장되고 지정된 행렬에 의해 변형된 후 기하 도형의 경계를 가져옵니다. (오버로드 2/4)
GradientMeshPatch 지정된 제어점, 색 및 경계 플래그를 포함하는 D2D1_GRADIENT_MESH_PATCH 구조를 만듭니다.
GradientMeshPatchFromCoonsPatch 지정된 Coons 패치 설명에서 D2D1_GRADIENT_MESH_PATCH 만듭니다.
Gradientstop D2D1_GRADIENT_STOP 구조를 만듭니다.
HasChildren 이 요소에 자식이 있는지 여부를 나타내는 부울 값을 반환합니다.
HwndRenderTargetProperties D2D1_HWND_RENDER_TARGET_PROPERTIES 구조를 만듭니다.
ID ID 행렬을 만듭니다. (Matrix3x2F.Identity)
IdentityMatrix ID 행렬을 만듭니다. (IdentityMatrix)
ImageBrushProperties D2D1_IMAGE_BRUSH_PROPERTIES 구조를 만듭니다.
InfiniteRect 왼쪽 위 모서리가 음수 무한대, 음수 무한대)로 설정되고 오른쪽 아래 모서리가 (무한대, 무한대)로 설정된 사각형을 만듭니다.

초기화 이 효과는 이 메서드를 사용하여 한 번 초기화 작업을 수행할 수 있습니다.
InkBezierSegment D2D1_INK_BEZIER_SEGMENT 구조를 만듭니다.
InkPoint D2D1_INK_POINT 구조를 만듭니다.
InkStyleProperties D2D1_INK_STYLE_PROPERTIES 구조를 만듭니다.
InsertChildBefore referenceChild 요소 앞에 newChild를 이 요소의 자식으로 삽입합니다.
InvalidateEffectInputRectangle 이는 효과 입력의 일부가 유효하지 않음을 나타냅니다. 이 메서드를 여러 번 호출할 수 있습니다.
반전 반전할 수 없는 경우 행렬을 반전합니다.
IsAttributeSpecified 특성이 요소에 명시적으로 설정되어 있는지 여부를 나타내는 부울 값을 반환합니다.
IsBufferPrecisionSupported 기본 Direct3D 디바이스에서 버퍼 전체 자릿수를 지원하는지 여부를 나타냅니다.
IsBufferPrecisionSupported 기본 Direct2D 디바이스에서 버퍼 전체 자릿수를 지원하는지 여부를 나타냅니다.
IsDxgiFormatSupported 형식이 디바이스 컨텍스트에서 지원되는지 여부를 나타냅니다.
IsIdentity 이 행렬이 ID 행렬인지 여부를 나타냅니다. (Matrix3x2F.IsIdentity)

IsIdentity 이 행렬이 ID 행렬인지 여부를 나타냅니다. (Matrix4x4F.IsIdentity)
IsInvertible 행렬을 반전할 수 있는지 여부를 나타냅니다.
IsShaderLoaded 이 테스트는 지정된 셰이더가 로드되었는지 확인합니다.
Issupported 렌더링 대상이 지정된 속성을 지원하는지 여부를 나타냅니다. (오버로드 1/2)
Issupported 렌더링 대상이 지정된 속성을 지원하는지 여부를 나타냅니다. (오버로드 2/2)
IsTextContent 이 요소가 텍스트 콘텐츠를 나타내는지 여부를 나타내는 부울 값을 반환합니다.
LayerParameters D2D1_LAYER_PARAMETERS 구조를 만듭니다.
LayerParameters1 레이어 리소스에 대한 콘텐츠 범위, 마스크 정보, 불투명도 설정 및 기타 옵션을 포함하는 D2D1_LAYER_PARAMETERS1 구조체를 반환합니다.
나가기 Direct2D API 중요 섹션(있는 경우)을 둡니다.
LinearGradientBrushProperties D2D1_LINEAR_GRADIENT_BRUSH_PROPERTIES 구조를 만듭니다.
LoadComputeShader 지정된 셰이더를 고유 ID로 로드합니다. (ID2D1EffectContext.LoadComputeShader)
LoadPixelShader 지정된 셰이더를 고유 ID로 로드합니다. (ID2D1EffectContext.LoadPixelShader)

LoadVertexShader 지정된 셰이더를 고유 ID로 로드합니다. (ID2D1EffectContext.LoadVertexShader)
Map 지정된 비트맵을 메모리에 매핑합니다.
Map 제공된 데이터를 사용자 메모리에 매핑합니다.
MapInputRectsToOutputRect MapOutputRectToInputRects에 대한 역 매핑을 수행합니다.
MapInvalidRect 이 렌더링 패스의 입력 사각형을 변환으로 설정합니다.
MapOutputRectToInputRects 변환에서 출력에 요청된 사각형을 입력의 샘플 사각형 집합에 매핑하는 방법을 명시할 수 있습니다.
Matrix3x2F 지정된 값을 포함하는 Matrix3x2F 클래스의 새 인스턴스를 인스턴스화합니다.
Matrix3x2F 행렬 값을 초기화하지 않고 Matrix3x2F 클래스의 새 인스턴스를 인스턴스화합니다.
OfferResources 운영 체제에서 콘텐츠를 삭제하여 리소스의 비디오 메모리를 해제할 수 있습니다. (ID2D1ImageSource.OfferResources)
열기 모집단에 대한 메시를 엽니다.
열기 경로 기하 도형을 그림 및 세그먼트로 채우는 데 사용되는 기하 도형 싱크를 검색합니다.

연산자* Matrix3x2F::operator-mult(d2d1helper.h) 메서드는 이 행렬을 지정된 행렬과 곱하고 결과를 반환합니다.
연산자* Matrix4x4F::operator-mult(d2d1_1helper.h) 인터페이스는 이 행렬을 지정된 행렬과 곱하고 결과를 반환합니다.
윤곽선 기하 도형의 윤곽선을 계산하고 결과를 ID2D1SimplifiedGeometrySink에 씁니다. (오버로드 3/4)
윤곽선 기하 도형의 윤곽선을 계산하고 결과를 ID2D1SimplifiedGeometrySink에 씁니다. (오버로드 2/4)
윤곽선 기하 도형의 윤곽선을 계산하고 결과를 ID2D1SimplifiedGeometrySink에 씁니다. (오버로드 1/4)
윤곽선 기하 도형의 윤곽선을 계산하고 결과를 ID2D1SimplifiedGeometrySink에 씁니다. (오버로드 4/4)
PD2D1_EFFECT_FACTORY 효과의 구현에 대해 설명합니다.
PD2D1_PROPERTY_GET_FUNCTION 효과에서 속성을 가져옵니다.
PD2D1_PROPERTY_SET_FUNCTION 효과에 대한 속성을 설정합니다.
PerspectiveProjection 깊이 값이 지정된 큐브 뷰 변환입니다.
PixelFormat D2D1_PIXEL_FORMAT 구조를 만듭니다.
Point2F 지정된 x 좌표와 y 좌표를 포함하는 D2D1_POINT_2F 구조를 만듭니다.

Point2L 긴 정수의 점 구조체를 반환합니다.
Point2U 지정된 x 좌표와 y 좌표를 포함하는 D2D1_POINT_2U 구조를 만듭니다.
PopAxisAlignedClip 렌더링 대상에서 마지막 축 맞춤 클립을 제거합니다. 이 메서드를 호출한 후에는 클립이 후속 그리기 작업에 더 이상 적용되지 않습니다.
PopAxisAlignedClip 레이어 및 클립 스택에서 축 맞춤 클립을 제거합니다.
PopLayer 마지막 PushLayer 호출에서 지정한 레이어로 그리기 작업 리디렉션을 중지합니다.
PopLayer 레이어 및 클립 스택에서 레이어를 제거합니다.
PrepareForRender 렌더링 프로세스에 대한 효과를 준비합니다.
PrintControlProperties 채워진 D2D1_PRINT_CONTROL_PROPERTIES 구조를 반환합니다.
ProcessAnalysisResults 분석 데이터를 분석 변환에 제공합니다.
ProcessRecord 이 메서드는 메타파일에 저장된 각 레코드에 대해 한 번 호출됩니다.
ProcessRecord 형식, 데이터 및 플래그를 포함하여 메타파일 레코드에 대한 액세스를 제공합니다.
PushAxisAlignedClip 모든 후속 그리기 작업이 잘린 사각형을 지정합니다. (오버로드 2/2)

PushAxisAlignedClip 모든 후속 그리기 작업이 잘린 사각형을 지정합니다. (오버로드 1/2)
PushAxisAlignedClip 클리핑 사각형을 클립 및 레이어 스택에 푸시합니다.
PushLayer PopLayer가 호출될 때까지 모든 후속 그리기 작업을 받을 수 있도록 지정된 레이어를 렌더링 대상에 추가합니다. (오버로드 2/2)
PushLayer PopLayer가 호출될 때까지 모든 후속 그리기 작업을 받을 수 있도록 지정된 레이어를 렌더링 대상에 추가합니다. (오버로드 1/2)
PushLayer 레이어를 클립 및 레이어 스택에 푸시합니다.
PushLayer 계층을 디바이스 컨텍스트의 클립 및 레이어 스택에 푸시합니다. (오버로드 2/2)
PushLayer 계층을 디바이스 컨텍스트의 클립 및 레이어 스택에 푸시합니다. (오버로드 1/2)
QuadraticBezierSegment D2D1_QUADRATIC_BEZIER_SEGMENT 구조를 만듭니다.
RadialGradientBrushProperties D2D1_RADIAL_GRADIENT_BRUSH_PROPERTIES 구조를 만듭니다.
RectF 지정된 차원을 포함하는 D2D1_RECT_F 구조를 만듭니다.
RectL 채워진 D2D1_RECT_L 구조를 반환합니다.
RectU 지정된 차원을 포함하는 D2D1_RECT_U 구조를 만듭니다.

RegisterEffectFromStream 스트림으로 지정된 속성 XML을 사용하여 팩터리 인스턴스 내의 효과를 등록합니다.
RegisterEffectFromString 문자열로 지정된 속성 XML을 사용하여 팩터리 인스턴스 내에서 효과를 등록합니다.
ReinterpretBaseType 복사본을 만들지 않고 지정된 D2D1_MATRIX_3X2_F 행렬을 Matrix3x2F로 변환합니다. (오버로드 1/2)
ReinterpretBaseType 복사본을 만들지 않고 지정된 D2D1_MATRIX_3X2_F 행렬을 Matrix3x2F로 변환합니다. (오버로드 2/2)
ReinterpretBaseType 복사본을 만들지 않고 지정된 D2D1_MATRIX_4X4_F 행렬을 Matrix4x4F로 변환합니다. (오버로드 1/2)
ReinterpretBaseType 복사본을 만들지 않고 지정된 D2D1_MATRIX_4X4_F 행렬을 Matrix4x4F로 변환합니다. (오버로드 2/2)
ReleaseDC GetDC 메서드를 사용하여 검색된 디바이스 컨텍스트를 사용하여 그리기를 완료했음을 나타냅니다.
ReloadSystemMetrics 팩터리 생성 이후 변경되었을 수 있는 시스템 기본값을 팩터리에서 새로 고치도록 합니다.
RemoveAttribute 이 요소에서 특성을 제거합니다.
RemoveChild 트리에서 oldChild를 제거합니다. oldChild의 자녀는 oldChild의 자녀로 남아 있습니다.
RemoveCommandsAtEnd 명령 배열의 끝에서 명령을 제거합니다.

RemoveDashesAtEnd 배열의 끝에서 대시를 제거합니다.
RemoveNode 변환 그래프에서 제공된 노드를 제거합니다.
RemovePointsAtEnd 배열의 끝에서 점을 제거합니다.
RemoveSegmentDataAtEnd 세그먼트 데이터 배열의 끝에서 데이터를 제거합니다.
RemoveSegmentsAtEnd 이 잉크 개체의 끝에서 지정된 세그먼트 수를 제거합니다.
RenderingControls 채워진 D2D1_RENDERING_CONTROLS 구조를 반환합니다.
RenderTargetProperties D2D1_RENDER_TARGET_PROPERTIES 구조를 만듭니다.
ReplaceChild oldChild 요소를 newChild로 바꿉니다.
크기 조정 렌더링 대상의 크기를 지정된 픽셀 크기로 변경합니다. (오버로드 2/2)
크기 조정 렌더링 대상의 크기를 지정된 픽셀 크기로 변경합니다. (오버로드 1/2)
RestoreDrawingState 렌더링 대상의 그리기 상태를 지정된 ID2D1DrawingStateBlock의 그리기 상태로 설정합니다.
회전 지정된 각도와 중심점이 있는 회전 변환을 만듭니다.

RotationArbitraryAxis 임의의 축에 대한 3차원 회전 행렬을 결정합니다.
RotationX X축을 중심으로 변환 행렬을 회전합니다.
회전(Rotationy) 변환 행렬을 Y축을 중심으로 회전합니다.
RotationZ Z축을 중심으로 변환 행렬을 회전합니다.
RoundedRect D2D1_ROUNDED_RECT 구조를 만듭니다.
SaveDrawingState 현재 그리기 상태를 지정된 ID2D1DrawingStateBlock에 저장합니다.
배율 지정된 배율 요소와 중심점이 있는 배율 변환을 만듭니다. (오버로드 2/2)
배율 지정된 배율 요소와 중심점이 있는 배율 변환을 만듭니다. (오버로드 1/2)
배율 행렬의 원근 평면 크기를 조정합니다.
직렬화 요소와 해당 하위 트리를 XML로 직렬화합니다. 출력 XML은 UTF-8로 인코딩됩니다.
SetAntialiasMode 렌더링 대상의 앤티앨리어싱 모드를 설정합니다. 앤티앨리어싱 모드는 텍스트 및 문자 모양 그리기 작업을 제외한 모든 후속 그리기 작업에 적용됩니다.
SetAntialiasMode 후속 기하 도형을 렌더링하는 데 사용할 앤티앨리어싱 모드를 설정합니다.

SetAttributeValue 이 요소의 특성을 색으로 설정합니다.
SetAttributeValue 이 요소의 특성을 행렬 값으로 설정합니다. 이 메서드를 사용하여 변환 또는 gradientTransform 특성의 값을 설정할 수 있습니다.
SetAttributeValue 이 요소의 특성을 길이 값으로 설정합니다.
SetAttributeValue 이 요소의 특성을 가로 세로 비율 유지 값으로 설정합니다. 이 메서드를 사용하여 preserveAspectRatio 특성의 값을 설정할 수 있습니다.
SetAttributeValue 이 요소의 특성을 확장 모드 값으로 설정합니다. 이 메서드를 사용하여 spreadMethod 특성의 값을 설정할 수 있습니다.
SetAttributeValue 이 요소의 특성을 채우기 모드로 설정합니다. 이 메서드는 'fill-rule' 또는 'clip-rule' 속성의 값을 설정하는 데 사용할 수 있습니다.
SetAttributeValue POD 형식을 사용하여 이 요소의 특성을 설정합니다.
SetAttributeValue 문자열을 사용하여 이 요소의 특성을 설정합니다.
SetAttributeValue 이 요소의 특성을 표시 값으로 가져옵니다. 이 메서드는 표시 속성의 값을 가져오는 데 사용할 수 있습니다. (오버로드 2/2)
SetAttributeValue 이 요소의 특성을 줄 바꿈 값으로 설정합니다. 이 메서드를 사용하여 stroke-linecap 속성의 값을 설정할 수 있습니다.

SetAttributeValue 이 요소의 특성을 선 조인 값으로 설정합니다. 이 메서드를 사용하여 stroke-linejoin 속성의 값을 설정할 수 있습니다.
SetAttributeValue 이 요소의 특성을 오버플로 값으로 설정합니다. 이 메서드를 사용하여 오버플로 속성의 값을 설정할 수 있습니다.
SetAttributeValue 이 요소의 특성을 단위 형식 값으로 설정합니다. 이 메서드를 사용하여 gradientUnits 또는 clipPathUnits 특성의 값을 설정할 수 있습니다.
SetAttributeValue 이 요소의 특성을 표시 유형 값으로 설정합니다. 이 메서드를 사용하여 표시 유형 속성의 값을 설정할 수 있습니다.
SetAttributeValue float를 사용하여 이 요소의 특성을 설정합니다.
SetAttributeValue 인터페이스를 사용하여 이 요소의 특성을 설정합니다.
SetBitmap 이 브러시가 그리는 데 사용하는 비트맵 소스를 지정합니다.
SetCached 지정된 변환의 출력이 캐시되는지 여부를 설정합니다.
SetCached 렌더링 정보가 캡슐화되는 변환의 출력이 캐시되거나 캐시되지 않도록 지정합니다.
SetCenter 브러시의 좌표 공간에서 그라데이션 타원의 중심을 지정합니다.
SetColor 이 단색 브러시의 색을 지정합니다.

SetColor 이 단색 브러시의 색을 지정합니다.
SetColor 페인트 유형이 D2D1_SVG_PAINT_TYPE_COLOR 경우 사용되는 페인트 색을 설정합니다. (오버로드 2/2)
SetColor 페인트 유형이 D2D1_SVG_PAINT_TYPE_COLOR 경우 사용되는 페인트 색을 설정합니다. (오버로드 1/2)
SetComputeInfo 컴퓨팅 셰이더 패스를 지정하는 데 사용되는 렌더링 정보를 설정합니다.
SetComputeShader 컴퓨팅 셰이더를 지정된 셰이더 리소스로 설정합니다. 이 호출을 수행하려면 먼저 리소스를 로드해야 합니다.
SetComputeShaderConstantBuffer 이 변환에 대한 상수 버퍼 데이터를 설정하거나 변경합니다.
SetDescription 그리기 상태의 앤티앨리어싱 모드, 변환 및 태그 부분을 지정합니다. (오버로드 1/2)
SetDescription 그리기 상태의 앤티앨리어싱 모드, 변환 및 태그 부분을 지정합니다. (오버로드 2/2)
SetDescription 이 그리기 상태 블록과 연결된 D2D1_DRAWING_STATE_DESCRIPTION1 설정합니다.
SetDescription 해당 blend 변환 개체의 혼합 설명을 변경합니다.
SetDpi 렌더링 대상의 DPI(인치당 점)를 설정합니다.

SetDpiCompensatedEffectInput 디바이스 컨텍스트의 DPI가 변경됨에 따라 시각적 모양을 유지하기 위해 DPI 보정 효과를 삽입하는 동안 비트맵을 효과 입력으로 설정합니다.
SetDrawInfo 변환 구현에 GPU 렌더링 정보 인터페이스를 제공합니다.
SetEndPoint 브러시의 좌표 공간에서 선형 그라데이션의 끝 좌표를 설정합니다.
SetExtendModeX 브러시가 비트맵을 지나 확장되는 영역을 가로로 타일하는 방법을 지정합니다.
SetExtendModeX 이미지 브러시의 원본 사각형 내 콘텐츠가 x축에서 확장되는 방식을 설정합니다.
SetExtendModeX 확장 모드를 x 방향으로 설정합니다.
SetExtendModeY 브러시가 비트맵을 지나 확장되는 영역을 세로로 타일하는 방법을 지정합니다.
SetExtendModeY y축의 확장 모드를 설정합니다.
SetExtendModeY 확장 모드를 y 방향으로 설정합니다.
SetFill 채우기를 위해 SVG 문자 모양에 값을 제공합니다.
SetFillMode 이 기하 도형 싱크에서 설명하는 기하 도형 내부에 있는 점과 외부에 있는 점을 결정하는 데 사용되는 메서드를 지정합니다.
SetGradientOriginOffset 그라데이션 타원의 중심을 기준으로 그라데이션 원점의 오프셋을 지정합니다.

SetGraph 렌더러는 이 메서드를 호출하여 효과 구현에 해당 변환 그래프를 지정하고 그래프 변경 내용을 변환하는 방법을 제공합니다.
SetId 페인트 서버 역할을 하는 요소 ID를 설정합니다. 이 ID는 페인트 유형이 D2D1_SVG_PAINT_TYPE_URI 경우에 사용됩니다.
SetImage 제공된 이미지 브러시와 연결된 이미지를 설정합니다.
SetInput 지정된 입력 이미지를 인덱스별로 설정합니다.
SetInputCount 애플리케이션에서 입력 수를 효과로 변경할 수 있습니다.
SetInputDescription 샘플링 측면에서 렌더러가 변환에 대한 특정 입력을 처리하는 방법을 설정합니다.
SetInputEffect 인덱스별로 지정된 입력 효과를 설정합니다.
SetInstructionCountHint D2D에 대한 셰이더 실행 비용에 대한 예상 힌트를 제공합니다.
SetInterpolationMode 브러시 비트맵의 크기를 조정하거나 회전할 때 사용되는 보간 모드를 지정합니다.
SetInterpolationMode 이미지 브러시의 보간 모드를 설정합니다.
SetInterpolationMode1 브러시의 보간 모드를 설정합니다.
SetMaximumColorGlyphCacheMemory 색 문자 모양 캐시의 최대 용량을 설정합니다.

SetMaximumTextureMemory Direct2D가 이미지 캐시 및 캐시된 텍스처 할당을 제거하기 전에 누적되는 최대 텍스처 메모리 양을 설정합니다. (ID2D1Device.SetMaximumTextureMemory)
SetNibShape 이 스타일에 대한 미리 변환된 nib 도형을 설정합니다.
SetNibTransform 이 스타일의 nib 셰이프에 적용할 변환을 설정합니다. (오버로드 1/2)
SetNibTransform 이 스타일의 nib 셰이프에 적용할 변환을 설정합니다. (오버로드 2/2)
SetOffset 현재 오프셋 변환의 오프셋을 설정합니다.
SetOpacity 이 브러시의 불투명도를 설정합니다.
SetOutputBounds 이렇게 하면 지원 변환에 대한 출력 범위가 설정됩니다.
SetOutputBuffer 지정된 변환 노드의 출력 버퍼 속성을 설정합니다.
SetOutputBuffer 호출자가 렌더링 정보가 캡슐화되는 변환의 출력 정밀도 및 채널 깊이를 제어할 수 있습니다.
SetOutputNode 변환 그래프의 출력 노드를 설정합니다.
SetPaintType 페인트 유형을 설정합니다.
SetPassthroughGraph 지정된 입력을 효과 출력으로 사용합니다.

SetPixelShader 이 변환에 대한 셰이더 지침을 설정합니다.
SetPixelShaderConstantBuffer 이 변환의 픽셀 셰이더에 대한 상수 버퍼를 설정합니다.
SetPrimitiveBlend 새 기본 혼합 모드를 설정합니다. (ID2D1CommandSink.SetPrimitiveBlend)
SetPrimitiveBlend 디바이스 컨텍스트의 모든 렌더링 작업에 사용되는 기본 혼합 모드를 변경합니다.
SetPrimitiveBlend1 새 기본 혼합 모드를 설정합니다. (ID2D1CommandSink1.SetPrimitiveBlend1)
SetPrimitiveBlend2 새 기본 혼합 모드를 설정합니다. MAX 기본 혼합 모드에 액세스할 수 있습니다.
SetProduct 두 행렬을 곱하고 결과를 이 행렬에 저장합니다. (Matrix3x2F.SetProduct)
SetProduct 두 행렬을 곱하고 결과를 이 행렬에 저장합니다. (Matrix4x4F.SetProduct)
SetRadiusX 브러시의 좌표 공간에서 그라데이션 타원의 x 반경을 지정합니다.
SetRadiusY 브러시의 좌표 공간에서 그라데이션 타원의 y 반경을 지정합니다.
SetRenderInfo 변환에 대한 렌더링 정보를 설정합니다.
SetRenderingControls 지정된 디바이스 컨텍스트에 대한 렌더링 컨트롤을 설정합니다. (오버로드 1/2)

SetRenderingControls 지정된 디바이스 컨텍스트에 대한 렌더링 컨트롤을 설정합니다. (오버로드 2/2)
SetRenderingPriority 디바이스와 연결된 모든 디바이스 컨텍스트에서 수행되는 Direct2D 렌더링 작업의 우선 순위를 설정합니다.
SetResourceTexture 지정된 셰이더 텍스처 인덱싱에 해당하는 리소스 텍스처를 지정된 텍스처 리소스로 설정합니다.
SetResourceTexture 지정된 셰이더 텍스처 인덱스로 해당하는 리소스 텍스처를 설정합니다.
SetRoot 문서의 루트 요소를 설정합니다.
SetSegmentAtEnd 새 제어점을 사용하여 이 잉크 개체의 마지막 세그먼트를 업데이트. (오버로드 2/2)
SetSegmentAtEnd 새 제어점을 사용하여 이 잉크 개체의 마지막 세그먼트를 업데이트. (오버로드 1/2)
SetSegmentFlags 기하 도형 싱크에 추가된 새 세그먼트에 적용할 스트로크 및 조인 옵션을 지정합니다.
SetSegments 이 잉크 개체의 지정된 세그먼트를 새 제어점으로 업데이트.
SetSingleTransformNode 단일 변환 노드를 전체 그래프와 동일한 것으로 설정합니다.
SetSourceRectangle 이미지 브러시에서 원본 사각형을 설정합니다.
SetSprites 이 스프라이트 일괄 처리에서 지정된 스프라이트의 속성을 업데이트.

SetStartPoint 브러시의 좌표 공간에서 선형 그라데이션의 시작 좌표를 설정합니다.
SetStartPoint 이 잉크 개체의 시작점을 설정합니다. 그러면 이 잉크 개체가 렌더링을 시작할 위치가 결정됩니다. (오버로드 2/2)
SetStartPoint 이 잉크 개체의 시작점을 설정합니다. 그러면 이 잉크 개체가 렌더링을 시작할 위치가 결정됩니다. (오버로드 1/2)
SetStroke 스트로크 속성에 대한 SVG 문자 모양에 값을 제공합니다. 불투명도가 1로 설정된 브러시는 '컨텍스트 스트로크'로 사용됩니다. 브러시의 불투명도는 '컨텍스트 스트로크 불투명도' 값으로 사용됩니다.
SetTags 후속 그리기 작업에 대한 레이블을 지정합니다.
SetTags 명령 싱크의 태그에 해당하는 태그를 설정합니다.
SetTarget 이제 Direct2D 디바이스 컨텍스트가 렌더링될 비트맵 또는 명령 목록입니다.
SetTextAntialiasMode 후속 텍스트 및 문자 모양 그리기 작업에 사용할 앤티앨리어싱 모드를 지정합니다.
SetTextAntialiasMode 텍스트에 대한 새 기본 앤티앨리어싱 모드를 나타냅니다.
SetTextRenderingParams 그리기 상태의 텍스트 렌더링 구성을 지정합니다.
SetTextRenderingParams 모든 후속 텍스트 및 문자 모양 그리기 작업에 적용할 텍스트 렌더링 옵션을 지정합니다.

SetTextRenderingParams 자세한 텍스트 렌더링 매개 변수를 나타냅니다.
SetTextValue 텍스트 콘텐츠 요소의 값을 설정합니다.
SetTransform 브러시에 적용된 변환을 설정합니다. (오버로드 2/2)
SetTransform 브러시에 적용된 변환을 설정합니다. (오버로드 1/2)
SetTransform 지정된 변환을 렌더링 대상에 적용하여 기존 변환을 대체합니다. 모든 후속 그리기 작업은 변환된 공간에서 발생합니다. (오버로드 2/2)
SetTransform 지정된 변환을 렌더링 대상에 적용하여 기존 변환을 대체합니다. 모든 후속 그리기 작업은 변환된 공간에서 발생합니다. (오버로드 1/2)
SetTransform 새 변환을 설정합니다.
SetUnitMode 단위 모드는 후속 단위의 의미를 디바이스 독립적 픽셀(DIP)에서 픽셀 또는 다른 방식으로 변경합니다. 명령 싱크는 DPI를 기록하지 않습니다. 이는 재생 컨텍스트 또는 ID2D1PrintControl과 같은 다른 재생 인터페이스에 의해 암시됩니다.
SetUnitMode 디바이스 컨텍스트에 전달된 값을 해석하는 데 사용할 단위를 설정합니다.
SetValue 인덱스별로 해당 속성을 설정합니다. 템플릿 오버로드입니다. (오버로드 2/2)
SetValue 인덱스별로 해당 속성을 설정합니다. 템플릿 오버로드입니다. (오버로드 1/2)

SetValue 인덱스별로 해당 속성을 설정합니다. (오버로드 1/2)
SetValue 인덱스별로 해당 속성을 설정합니다. (오버로드 2/2)
SetValueByName 명명된 속성을 지정된 값으로 설정합니다. (오버로드 1/2)
SetValueByName 명명된 속성을 지정된 값으로 설정합니다. 템플릿 오버로드입니다. 설명 부분을 참조하세요.
SetValueByName 명명된 속성을 지정된 값으로 설정합니다. (오버로드 2/2)
SetVertexProcessing 꼭짓점 버퍼, 해당 꼭짓점 셰이더 및 Direct2D 컨텍스트에서 꼭짓점을 처리하는 방법을 제어하는 옵션을 설정합니다.
SetVertexShaderConstantBuffer 이 변환의 꼭짓점 셰이더에 대한 상수 버퍼를 설정합니다.
SetViewportSize 초기 뷰포트의 크기를 설정합니다.
단순화 선 및(선택 사항) 입방형 베지어 곡선만 포함하는 간소화된 기하 도형 버전을 만들고 결과를 ID2D1SimplifiedGeometrySink에 씁니다. (오버로드 4/4)
단순화 선 및(선택 사항) 입방형 베지어 곡선만 포함하는 간소화된 기하 도형 버전을 만들고 결과를 ID2D1SimplifiedGeometrySink에 씁니다. (오버로드 1/4)
단순화 선 및(선택 사항) 입방형 베지어 곡선만 포함하는 간소화된 기하 도형 버전을 만들고 결과를 ID2D1SimplifiedGeometrySink에 씁니다. (오버로드 2/4)

단순화 선 및(선택 사항) 입방형 베지어 곡선만 포함하는 간소화된 기하 도형 버전을 만들고 결과를 ID2D1SimplifiedGeometrySink에 씁니다. (오버로드 3/4)
Sizef 지정된 너비와 높이를 포함하는 D2D1_SIZE_F 구조를 만듭니다.
SizeU 지정된 너비와 높이를 포함하는 D2D1_SIZE_U 구조를 만듭니다.
기울이기 지정된 x축 및 y축 값과 중심점이 있는 기울이기 변환을 만듭니다.
SkewX 행렬을 X 방향으로 기울입니다.
기울이기 행렬을 Y 방향으로 기울입니다.
스트림 경로 기하 도형의 내용을 지정된 ID2D1GeometrySink에 복사합니다.
스트림 명령 목록의 내용을 지정된 명령 싱크로 스트리밍합니다.
스트림 이 메서드는 명령의 내용을 지정된 메타파일 싱크로 스트리밍합니다.
StreamAsGeometry 이 잉크 개체의 기하학적 표현을 검색합니다. (오버로드 4/4)
StreamAsGeometry 이 잉크 개체의 기하학적 표현을 검색합니다. (오버로드 1/4)
StreamAsGeometry 이 잉크 개체의 기하학적 표현을 검색합니다. (오버로드 2/4)

StreamAsGeometry 이 잉크 개체의 기하학적 표현을 검색합니다. (오버로드 3/4)
StringGetter 문자열 형식 속성에 대한 멤버 함수 속성 getter 콜백을 호출합니다.
StringGetter 문자열 형식 속성에 대한 멤버 함수 속성 getter 콜백을 호출합니다.
StringSetter 문자열 형식 속성에 대한 멤버 함수 속성 setter 콜백을 호출합니다.
StringSetter 문자열 형식 속성에 대한 멤버 함수 속성 setter 콜백을 호출합니다.
StrokeContainsPoint 지정된 스트로크 두께, 스타일 및 변환이 지정된 경우 기하 도형의 스트로크에 지정된 점이 포함되어 있는지 여부를 결정합니다. (오버로드 4/4)
StrokeContainsPoint 지정된 스트로크 두께, 스타일 및 변환이 지정된 경우 기하 도형의 스트로크에 지정된 점이 포함되어 있는지 여부를 결정합니다. (오버로드 2/4)
StrokeContainsPoint 지정된 스트로크 두께, 스타일 및 변환이 지정된 경우 기하 도형의 스트로크에 지정된 점이 포함되어 있는지 여부를 결정합니다. (오버로드 1/4)
StrokeContainsPoint 지정된 스트로크 두께, 스타일 및 변환이 지정된 경우 기하 도형의 스트로크에 지정된 점이 포함되어 있는지 여부를 결정합니다. (오버로드 3/4)
StrokeStyleProperties D2D1_STROKE_STYLE_PROPERTIES 구조를 만듭니다.
StrokeStyleProperties1 채워진 D2D1_STROKE_STYLE_PROPERTIES1 구조를 반환합니다.

테셀레이트 () 지정한 행렬을 사용하여 변환되고 지정한 허용 오차를 사용하여 평면화된 기하 도형을 포괄하는 시계 방향으로 돌아가는 삼각형 집합을 만듭니다. (오버로드 1/2)
테셀레이트 () 지정된 행렬을 사용하여 변형되고 기본 허용 오차를 사용하여 평면화된 후 기하 도형을 덮는 시계 방향으로 감겨진 삼각형 집합을 만듭니다. (오버로드 2/2)
테셀레이트 () 지정한 행렬을 사용하여 변환되고 지정한 허용 오차를 사용하여 평면화된 기하 도형을 포괄하는 시계 방향으로 돌아가는 삼각형 집합을 만듭니다. (오버로드 2/2)
테셀레이트 () 지정된 행렬을 사용하여 변형되고 기본 허용 오차를 사용하여 평면화된 후 기하 도형을 덮는 시계 방향으로 감겨진 삼각형 집합을 만듭니다. (오버로드 1/2)
TransformPoint 이 행렬을 사용하여 지정된 점을 변환하고 결과를 반환합니다.
번역 지정된 x 및 y 변위가 있는 변환 변환을 만듭니다. (오버로드 1/2)
번역 지정된 x 및 y 변위가 있는 변환 변환을 만듭니다. (오버로드 2/2)
TrimCache 이 메서드는 이미지 원본 캐시의 채워진 영역을 지정된 사각형으로만 트리밍합니다. (오버로드 2/2)
TrimCache 이 메서드는 이미지 원본 캐시의 채워진 영역을 지정된 사각형으로만 트리밍합니다. (오버로드 1/2)
TryReclaimResources OfferResources를 호출하여 이전에 제공된 리소스에 대한 액세스를 복원합니다.
Unmap 메모리에서 비트맵의 매핑을 해제합니다.

Unmap 꼭짓점 버퍼의 매핑을 해제합니다.
UnregisterEffect 제공된 classId에 해당하는 팩터리 인스턴스 내에서 효과를 등록 취소합니다.
업데이트 제공된 데이터를 사용하여 특정 범위 또는 상자 내의 특정 리소스 텍스처를 업데이트.
UpdateCommands 명령 배열을 업데이트. 이 메서드에 의해 업데이트되지 않은 기존 명령은 유지됩니다. 새 명령을 수용하기 위해 필요한 경우 배열의 크기가 더 큽니다.
UpdateDashes 배열을 업데이트. 이 메서드에서 업데이트하지 않는 기존 대시는 유지됩니다. 새 대시를 수용하기 위해 필요한 경우 배열의 크기가 더 큽니다. (오버로드 2/2)
UpdateDashes 배열을 업데이트. 이 메서드에서 업데이트하지 않는 기존 대시는 유지됩니다. 새 대시를 수용하기 위해 필요한 경우 배열의 크기가 더 큽니다. (오버로드 1/2)
UpdatePoints 점 배열을 업데이트. 이 메서드에 의해 업데이트되지 않은 기존 지점은 유지됩니다. 새 점을 수용하기 위해 필요한 경우 배열의 크기가 더 큽니다.
UpdateSegmentData 세그먼트 데이터 배열을 업데이트. 이 메서드에 의해 업데이트되지 않은 기존 세그먼트 데이터는 유지됩니다. 새 세그먼트 데이터를 수용하기 위해 필요한 경우 배열의 크기가 더 큽니다.
ValueGetter 값 형식 속성에 대한 멤버 함수 속성 setter 콜백을 호출합니다. (ValueGetter)
ValueGetter 값 형식 속성에 대한 멤버 함수 속성 setter 콜백을 호출합니다. (ValueGetter)
ValueSetter 값 형식 속성에 대한 멤버 함수 속성 setter 콜백을 호출합니다. (ValueSetter)

ValueSetter 값 형식 속성에 대한 멤버 함수 속성 setter 콜백을 호출합니다. (ValueSetter)
Vector2F 채워진 D2D1_VECTOR_2F 구조를 반환합니다.
Vector3F 채워진 D2D1_VECTOR_3F 구조를 반환합니다.
Vector4F 채워진 D2D1_VECTOR_4F 구조를 반환합니다.
넓힐 지정된 스트로크로 기하 도형을 넓히고 지정된 행렬에 의해 변환되고 지정된 허용 오차를 사용하여 평면화된 후 ID2D1SimplifiedGeometrySink에 결과를 씁니다. (오버로드 1/2)
넓힐 지정된 스트로크를 기준으로 기하 도형을 넓히고, ID2D1SimplifiedGeometrySink가 지정된 행렬에 의해 변환되고 기본 허용 오차를 사용하여 평면화된 후 결과를 ID2D1SimplifiedGeometrySink에 씁니다. (오버로드 2/2)
넓힐 지정된 스트로크로 기하 도형을 넓히고 지정된 행렬에 의해 변환되고 지정된 허용 오차를 사용하여 평면화된 후 ID2D1SimplifiedGeometrySink에 결과를 씁니다. (오버로드 2/2)
넓힐 지정된 스트로크를 기준으로 기하 도형을 넓히고, ID2D1SimplifiedGeometrySink가 지정된 행렬에 의해 변환되고 기본 허용 오차를 사용하여 평면화된 후 결과를 ID2D1SimplifiedGeometrySink에 씁니다. (오버로드 1/2)

인터페이스

ID2D1AnalysisTransform 분석 효과에 데이터를 제공합니다.

ID2D1Bitmap ID2D1RenderTarget에 바인딩된 비트맵을 나타냅니다.
ID2D1Bitmap1 ID2D1DeviceContext의 표면으로 사용하거나 시스템 메모리에 매핑할 수 있고 추가 색 컨텍스트 정보를 포함할 수 있는 비트맵을 나타냅니다.
ID2D1BitmapBrush 비트맵으로 영역을 그립니다. (ID2D1BitmapBrush)
ID2D1BitmapBrush1 비트맵으로 영역을 그립니다. (ID2D1BitmapBrush1)
ID2D1BitmapRenderTarget CreateCompatibleRenderTarget 메서드에서 만든 중간 텍스처로 렌더링합니다.
ID2D1BlendTransform 혼합 연산을 변환 그래프에 삽입할 수 있도록 하는 메서드를 제공합니다.
ID2D1BorderTransform 지정된 확장 모드를 사용하여 입력 사각형을 무한대로 확장합니다.
ID2D1BoundsAdjustmentTransform 이전 효과 또는 비트맵의 출력 사각형을 수정하는 효과에 대한 지원 변환입니다.
ID2D1Brush 영역을 그리는 개체를 정의합니다. ID2D1Brush에서 파생되는 인터페이스는 영역이 그려지는 방법을 설명합니다.
ID2D1ColorContext ID2D1Bitmap1 개체와 함께 사용할 수 있는 색 컨텍스트를 나타냅니다.
ID2D1ColorContext1 색 관리 효과와 함께 사용할 색 컨텍스트를 나타냅니다.
ID2D1CommandList 기록하고 재생할 수 있는 명령 시퀀스를 나타냅니다.

ID2D1CommandSink 명령 목록에 기록된 명령의 재생을 수신하려는 경우 애플리케이션에 대해 명령 싱크가 구현됩니다.
ID2D1CommandSink1 이 인터페이스는 기존 ID2D1CommandSink 인터페이스와 동일한 모든 함수를 수행합니다. 또한 SetPrimitiveBlend1 메서드를 통해 새로운 기본 혼합 모드인 MIN 및 ADD에 액세스할 수 있습니다.
ID2D1CommandSink2 이 인터페이스는 기존 ID2D1CommandSink1 인터페이스와 동일한 모든 함수를 수행합니다. 또한 잉크 렌더링 및 그라데이션 메시 렌더링에 액세스할 수 있습니다.
ID2D1CommandSink3 이 인터페이스는 기존 ID2D1CommandSink2 인터페이스와 동일한 모든 함수를 수행합니다. 또한 스프라이트 일괄 처리 렌더링에 액세스할 수 있습니다.
ID2D1CommandSink4 이 인터페이스는 기존 ID2D1CommandSink3 인터페이스와 동일한 모든 함수를 수행합니다. 또한 SetPrimitiveBlend2 메서드를 통해 새로운 기본 혼합 모드 MAX에 액세스할 수 있습니다.
ID2D1CommandSink5 이 인터페이스는 기존 ID2D1CommandSink4 인터페이스와 동일한 모든 함수를 수행하고 BlendImage 메서드에 액세스할 수 있도록 합니다.
ID2D1ComputeInfo 컴퓨팅 셰이더 렌더링 패스에 대한 정보 사양을 사용하도록 설정합니다.
ID2D1ComputeTransform 컴퓨팅 셰이더를 사용하는 변환을 정의합니다.
ID2D1ConcreteTransform 고정 기능을 제공하는 효과 렌더링 시스템에서 구현하는 변환 집합을 나타냅니다.
ID2D1DCRenderTarget GDI 디바이스 컨텍스트에 그리기 명령을 발급합니다.
ID2D1Device 개체와 디바이스 컨텍스트를 함께 사용할 수 있는 리소스 도메인을 나타냅니다. (ID2D1Device)

ID2D1Device1 개체와 디바이스 컨텍스트를 함께 사용할 수 있는 리소스 도메인을 나타냅니다. (ID2D1Device1)
ID2D1Device2 개체와 디바이스 컨텍스트를 함께 사용할 수 있는 리소스 도메인을 나타냅니다. 이 인터페이스는 기존 ID2D1Device1 인터페이스와 동일한 모든 함수를 수행합니다. 또한 ID2D1DeviceContext2 개체를 만들 수 있습니다.
ID2D1Device3 개체와 디바이스 컨텍스트를 함께 사용할 수 있는 리소스 도메인을 나타냅니다. 이 인터페이스는 ID2D1Device2 인터페이스와 동일한 모든 함수를 수행합니다. 또한 ID2D1DeviceContext3 개체를 만들 수 있습니다.
ID2D1Device4 개체와 디바이스 컨텍스트를 함께 사용할 수 있는 리소스 도메인을 나타냅니다. 이 인터페이스는 ID2D1Device3 인터페이스와 동일한 모든 함수를 수행합니다. 또한 ID2D1DeviceContext4 개체를 만들 수 있습니다.
ID2D1Device5 개체와 디바이스 컨텍스트를 함께 사용할 수 있는 리소스 도메인을 나타냅니다. 이 인터페이스는 ID2D1Device4 인터페이스와 동일한 모든 함수를 수행합니다. 또한 ID2D1DeviceContext5 개체를 만들 수 있습니다.
ID2D1Device6 개체와 디바이스 컨텍스트를 함께 사용할 수 있는 리소스 도메인을 나타냅니다. 이 인터페이스는 ID2D1Device5 인터페이스와 동일한 모든 함수를 수행하고 ID2D1DeviceContext6 개체를 만들 수 있습니다.
ID2D1DeviceContext 대상에 렌더링하는 데 사용되는 상태 및 명령 버퍼 집합을 나타냅니다.
ID2D1DeviceContext1 기하 도형 실현 개체를 만들고 그릴 수 있습니다.
ID2D1DeviceContext2 이 인터페이스는 ID2D1DeviceContext1 인터페이스와 동일한 모든 기능을 수행하며 잉크 렌더링, 그라데이션 메시 렌더링 및 향상된 이미지 로드와 같은 기능을 사용할 수 있습니다.

ID2D1DeviceContext3 이 인터페이스는 ID2D1DeviceContext2 인터페이스와 동일한 모든 기능을 수행하며 스프라이트 일괄 처리를 만들고 그리는 기능을 사용할 수 있습니다.
ID2D1DeviceContext4 이 인터페이스는 ID2D1DeviceContext3 인터페이스와 동일한 모든 함수를 수행하며 새로운 유형의 색 글꼴 문자 모양을 처리하는 기능을 사용할 수 있습니다.
ID2D1DeviceContext5 이 인터페이스는 ID2D1DeviceContext4 인터페이스와 동일한 모든 함수를 수행하며 색 컨텍스트 및 Svg 문서를 만들 수 있습니다.
ID2D1DeviceContext6 이 인터페이스는 기존 ID2D1DeviceContext5 인터페이스와 동일한 모든 함수를 수행하고 BlendImage 메서드에 액세스할 수 있도록 합니다.
ID2D1DrawInfo 이 인터페이스는 꼭짓점 또는 픽셀 셰이더에서 GPU 렌더링 패스를 설명하는 데 사용됩니다. ID2D1DrawTransform에 전달됩니다.
ID2D1DrawingStateBlock 렌더링 대상:_the 앤티앨리어싱 모드, 변환, 태그 및 텍스트 렌더링 옵션의 그리기 상태를 나타냅니다.
ID2D1DrawingStateBlock1 기존 앤티앨리어스 모드, 변환, 태그 및 텍스트 렌더링 모드 외에도 기본 혼합의 기능을 추가하는 그리기 상태 블록의 구현입니다.
ID2D1DrawTransform GPU에 구현된 변환에 대한 Shantzis 계산의 특수한 구현입니다.
ID2D1Effect Direct2D의 기본 이미지 처리 구문을 나타냅니다.
ID2D1EffectContext 효과 및 변환 작성자용 팩터리 메서드 및 기타 상태 관리를 제공합니다. (ID2D1EffectContext)

ID2D1EffectContext1 효과 및 변환 작성자용 팩터리 메서드 및 기타 상태 관리를 제공합니다. (ID2D1EffectContext1)
ID2D1EffectImpl 효과 작성자가 사용자 지정 효과의 인터페이스 및 동작을 지정할 수 있도록 허용합니다.
ID2D1EllipseGeometry 줄임표를 나타냅니다.
ID2D1Factory Direct2D 리소스를 만듭니다. (ID2D1Factory)
ID2D1Factory1 Direct2D 리소스를 만듭니다. (ID2D1Factory1)
ID2D1Factory2 Direct2D 리소스를 만듭니다. (ID2D1Factory2)
ID2D1Factory3 Direct2D 리소스를 만듭니다. 또한 이 인터페이스를 사용하면 ID2D1Device2 개체를 만들 수 있습니다.
ID2D1Factory4 Direct2D 리소스를 만듭니다. 또한 이 인터페이스를 사용하면 ID2D1Device3 개체를 만들 수 있습니다.
ID2D1Factory5 Direct2D 리소스를 만듭니다. 또한 이 인터페이스를 사용하면 ID2D1Device4 개체를 만들 수 있습니다.
ID2D1Factory6 Direct2D 리소스를 만듭니다. 또한 이 인터페이스를 사용하면 ID2D1Device5 개체를 만들 수 있습니다.
ID2D1Factory7 Direct2D 리소스를 만듭니다. 이 인터페이스는 ID2D1Factory6 인터페이스와 동일한 모든 기능을 수행하고 ID2D1Device6 개체를 만들 수 있습니다.

ID2D1GdiInteropRenderTarget GDI 그리기 명령을 수락할 수 있는 디바이스 컨텍스트에 대한 액세스를 제공합니다.
ID2D1GdiMetafile WMF, EMF 또는 EMF+ 메타파일을 래핑하는 Direct2D 리소스입니다.
ID2D1GdiMetafile1 이 인터페이스는 기존 ID2D1GdiMetafile 인터페이스와 동일한 모든 함수를 수행합니다. 또한 메타파일 DPI 및 경계에 액세스할 수 있습니다.
ID2D1GdiMetafileSink 메타파일을 재생할 수 있는 개발자 구현 인터페이스입니다.
ID2D1GdiMetafileSink1 이 인터페이스는 기존 ID2D1GdiMetafileSink 인터페이스와 동일한 모든 함수를 수행합니다. 또한 메타파일 레코드에 액세스할 수 있습니다.
ID2D1Geometry 기하 도형 리소스를 나타내고 기하 도형을 조작하고 측정하기 위한 도우미 메서드 집합을 정의합니다. ID2D1Geometry에서 상속되는 인터페이스는 특정 셰이프를 정의합니다.
ID2D1GeometryGroup 다른 ID2D1Geometry 개체로 구성된 복합 기하 도형을 나타냅니다.
ID2D1GeometryRealization 채워지거나 스트로크된 기하 도형의 디바이스 및 변환 종속 표현을 캡슐화합니다.
ID2D1GeometrySink 선, 호, 입방형 베지어 곡선 및 4차원 베지어 곡선을 포함할 수 있는 기하학적 경로를 설명합니다.
ID2D1GradientMesh 패치로 구성된 그라데이션 메시의 디바이스 종속 표현을 나타냅니다. ID2D1DeviceContext2::CreateGradientMesh 메서드를 사용하여 ID2D1GradientMesh 인스턴스를 만듭니다.
ID2D1GradientStopCollection 선형 및 방사형 그라데이션 브러시에 대한 D2D1_GRADIENT_STOP 개체의 컬렉션을 나타냅니다.

ID2D1GradientStopCollection1 선형 및 방사형 그라데이션 브러시에 대한 D2D1_GRADIENT_STOP 개체의 컬렉션을 나타냅니다. 그라데이션 중지 컬렉션에 추가된 모든 새 매개 변수에 대한 get 메서드를 제공합니다.
ID2D1HwndRenderTarget 드로잉 지침을 창에 렌더링합니다.
ID2D1Image 임의의 2D 평면을 채울 수 있는 픽셀의 생산자를 나타냅니다. (ID2D1Image)
ID2D1ImageBrush ID2D1Image를 기반으로 하는 브러시를 나타냅니다.
ID2D1ImageSource 임의의 2D 평면을 채울 수 있는 픽셀의 생산자를 나타냅니다. (ID2D1ImageSource)
ID2D1ImageSourceFromWic WIC에서 원본으로 제공되는 2D 픽셀 데이터를 생성합니다.
ID2D1Ink 일련의 Bezier 세그먼트 및 너비로 정의된 가변 너비 잉크의 단일 연속 스트로크를 나타냅니다.
ID2D1InkStyle 잉크를 렌더링할 때 ID2D1DeviceContext2::D rawInk와 같은 메서드에서 사용할 스타일 속성의 컬렉션을 나타냅니다. 잉크 스타일은 펜 팁(펜 팁) 모양과 변환을 정의합니다.
ID2D1Layer 레이어를 렌더링하는 데 필요한 백업 저장소를 나타냅니다.
ID2D1LinearGradientBrush 선형 그라데이션으로 영역을 그립니다.
ID2D1LookupTable3D LookupTable3D 효과에 전달할 수 있는 3D 조회 테이블 데이터에 대한 컨테이너입니다.
ID2D1Mesh 삼각형 목록을 구성하는 꼭짓점 집합을 나타냅니다.

ID2D1Multithread Direct2D가 여러 스레드를 사용하는 앱에서 전용 리소스 액세스를 제어하는 데 사용하는 Direct2D 팩터리의 잠금 메커니즘입니다.
ID2D1OffsetTransform 렌더링 패스를 삽입하지 않고 입력 비트맵을 오프셋하도록 효과 렌더링 시스템에 지시합니다.
ID2D1PathGeometry 호, 곡선 및 선으로 구성될 수 있는 복합 셰이프를 나타냅니다.
ID2D1PathGeometry1 ID2D1PathGeometry1 인터페이스는 ID2D1PathGeometry에 기능을 추가합니다. 특히 경로 기하 도형별 ComputePointAndSegmentAtLength 메서드를 제공합니다.
ID2D1PrintControl ID2D1CommandList에 저장된 Direct2D 기본 형식을 고정 페이지 표현으로 변환합니다. 그런 다음 인쇄 하위 시스템에서 기본 형식을 사용합니다.
ID2D1Properties 데이터 기반 애플리케이션이 Direct2D 효과의 상태를 수정할 수 있도록 하는 런타임 바인딩 가능 및 검색 가능한 속성 집합을 나타냅니다.
ID2D1RadialGradientBrush 방사형 그라데이션으로 영역을 그립니다.
ID2D1RectangleGeometry 2차원 사각형을 설명합니다.
ID2D1RenderInfo 모든 다양한 변환 구현에 공통적인 렌더링 정보를 설명합니다.
ID2D1RenderTarget 그리기 명령을 받을 수 있는 개체를 나타냅니다. ID2D1RenderTarget에서 상속되는 인터페이스는 다양한 방법으로 수신하는 그리기 명령을 렌더링합니다.
ID2D1Resource Direct2D 그리기 리소스를 나타냅니다.

ID2D1ResourceTexture 변환으로 만든 리소스 텍스처를 추적합니다.
ID2D1RoundedRectangleGeometry 둥근 사각형을 설명합니다.
ID2D1SimplifiedGeometrySink 이차 베지어 곡선 또는 호를 포함하지 않는 기하학적 경로를 설명합니다.
ID2D1SolidColorBrush 단색으로 영역을 그립니다.
ID2D1SourceTransform 변환 파이프라인 그래프의 CPU 기반 래스터화 단계를 나타냅니다.
ID2D1SpriteBatch 연결된 그리기 속성이 있는 스프라이트의 단일 그룹을 나타냅니다.
ID2D1StrokeStyle 스트로크에 대한 대문자, 마이터 제한, 선 조인 및 대시 정보를 설명합니다. (ID2D1StrokeStyle)
ID2D1StrokeStyle1 스트로크에 대한 대문자, 마이터 제한, 선 조인 및 대시 정보를 설명합니다. (ID2D1StrokeStyle1)
ID2D1SvgAttribute SVG 특성을 설명하는 인터페이스입니다.
ID2D1SvgDocument SVG 문서를 나타냅니다.
ID2D1SvgElement 모든 SVG 요소에 대한 인터페이스입니다.
ID2D1SvgglyphStyle 이 개체는 SVG 문자 모양을 렌더링할 때 사용되는 컨텍스트 채우기, 컨텍스트 스트로크 및 컨텍스트-값에 대한 값을 제공합니다.

ID2D1SvgPaint SVG 채우기 또는 스트로크 값을 설명하는 인터페이스입니다.
ID2D1SvgPathData SVG 경로 데이터를 설명하는 인터페이스입니다. 경로 데이터는 'path' 요소에서 'd' 특성으로 설정할 수 있습니다.
ID2D1SvgPointCollection 다각형 또는 다각형 요소의 SVG 점 값을 설명하는 인터페이스입니다.
ID2D1SvgStrokeDashArray SVG stroke-dasharray 값을 설명하는 인터페이스입니다.
ID2D1TessellationSink ID2D1Mesh 개체를 삼각형으로 채웁니다.
ID2D1Transform 변환 작성자가 구현하는 모든 변환의 기본 인터페이스를 나타냅니다.
ID2D1TransformedGeometry 변환된 기하 도형을 나타냅니다.
ID2D1TransformedImageSource 원본 이미지 원본과 리소스를 공유하는 이미지 원본을 나타냅니다.
ID2D1TransformGraph 변환 노드의 그래프를 나타냅니다.
ID2D1TransformNode 변환 토폴로지의 노드에 대해 설명합니다.
ID2D1VertexBuffer 매핑 가능한 1차원 꼭짓점 버퍼를 정의합니다.

구조체

D2D_MATRIX_3X2_F 3-by-2 행렬을 나타냅니다.
D2D_MATRIX_4X3_F 4-by-3 부동 소수점 행렬을 설명합니다.
D2D_MATRIX_4X4_F 4-by-4 부동 소수점 행렬을 설명합니다.
D2D_MATRIX_5X4_F 5-4 부동 소수점 행렬을 설명합니다.
D2D_POINT_2F 2차원 공간에서 부동 소수점 값으로 표현되는 x 좌표 및 y 좌표 쌍을 나타냅니다.
D2D_POINT_2U 2차원 공간에서 부호 없는 32비트 정수 값으로 표현되는 x 좌표 및 y 좌표 쌍을 나타냅니다.
D2D_RECT_F 왼쪽 위 모퉁이(왼쪽, 위쪽)의 좌표와 오른쪽 아래 모서리(오른쪽, 아래쪽)의 좌표로 정의된 사각형을 나타냅니다.
D2D_RECT_U 좌표의 왼쪽 위 모퉁이 쌍(왼쪽, 위쪽) 및 오른쪽 아래 모서리 좌표 쌍(오른쪽, 아래쪽)으로 정의된 사각형을 나타냅니다. 이러한 좌표는 32비트 정수 값으로 표현됩니다.
D2D_SIZE_F 정렬된 부동 소수점 값 쌍(일반적으로 사각형의 너비 및 높이)을 저장합니다.
D2D_SIZE_U 보통 사각형의 너비와 높이의 순서로 정렬된 정수 쌍을 저장합니다.
D2D_VECTOR_2F FLOAT 값 2개(x, y)의 벡터입니다.
D2D_VECTOR_3F FLOAT 값 3개(x, y, z)의 벡터입니다.

D2D_VECTOR_4F FLOAT 값 4개(x, y, z, w)의 벡터입니다.
D2D1_ARC_SEGMENT 두 점 사이의 타원형 호를 설명합니다.
D2D1_BEZIER_SEGMENT 두 점 사이에 그려진 입방형 베지어 세그먼트를 나타냅니다.
D2D1_BITMAP_BRUSH_PROPERTIES ID2D1BitmapBrush의 확장 모드 및 보간 모드에 대해 설명합니다. (D2D1_BITMAP_BRUSH_PROPERTIES)
D2D1_BITMAP_BRUSH_PROPERTIES1 ID2D1BitmapBrush의 확장 모드 및 보간 모드에 대해 설명합니다. (D2D1_BITMAP_BRUSH_PROPERTIES1)
D2D1_BITMAP_PROPERTIES 비트맵의 픽셀 형식 및 dpi를 설명합니다.
D2D1_BITMAP_PROPERTIES1 이 구조를 사용하면 사용할 수 있는 비트맵 옵션 및 색 컨텍스트 정보를 사용하여 ID2D1Bitmap1을 만들 수 있습니다.
D2D1_BLEND_DESCRIPTION 특정 혼합 변환에 사용할 혼합 설명을 정의합니다.
D2D1_BRUSH_PROPERTIES 브러시의 불투명도 및 변환에 대해 설명합니다.
D2D1_CREATION_PROPERTIES Direct2D 디바이스, 팩터리 및 디바이스 컨텍스트를 만드는 옵션을 지정합니다.
D2D1_CUSTOM_VERTEX_BUFFER_PROPERTIES 꼭짓점 셰이더 및 입력 요소 설명을 정의하여 입력 레이아웃을 정의합니다.

D2D1_DRAWING_STATE_DESCRIPTION 렌더링 대상의 그리기 상태를 설명합니다.
D2D1_DRAWING_STATE_DESCRIPTION1 디바이스 컨텍스트의 그리기 상태를 설명합니다.
D2D1_EFFECT_INPUT_DESCRIPTION 효과의 기능을 설명합니다.
D2D1_ELLIPSE 타원의 중심점, x-반경 및 y 반경을 포함합니다.
D2D1_FACTORY_OPTIONS ID2D1Factory 개체의 디버깅 수준을 포함합니다.
D2D1_FEATURE_DATA_D3D10_X_HARDWARE_OPTIONS D3D10 기능 수준의 옵션인 컴퓨팅 셰이더 지원에 대해 설명합니다.
D2D1_FEATURE_DATA_DOUBLES 셰이더에서 double에 대한 지원을 설명합니다.
D2D1_GRADIENT_MESH_PATCH 제어점 16개, 모서리 색 4개 및 경계 플래그가 있는 텐서 패치를 나타냅니다. ID2D1GradientMesh는 1개 이상의 그라데이션 메시 패치로 구성됩니다. GradientMeshPatch 함수 또는 GradientMeshPatchFromCoonsPatch 함수를 사용하여 만듭니다.
D2D1_GRADIENT_STOP 그라데이션 중지점의 위치와 색을 포함합니다.
D2D1_HWND_RENDER_TARGET_PROPERTIES ID2D1HwndRenderTarget에 대한 HWND, 픽셀 크기 및 프레젠테이션 옵션을 포함합니다.
D2D1_IMAGE_BRUSH_PROPERTIES 이미지 브러시 기능을 설명합니다.

D2D1_INK_BEZIER_SEGMENT ID2D1Ink 개체를 만드는 데 사용할 Bezier 세그먼트를 나타냅니다. 이 구조는 x 및 y 좌표 외에도 반경을 포함하는 D2D1_INK_POINTs 구성된다는 점에서 D2D1_BEZIER_SEGMENT 다릅니다.
D2D1_INK_POINT D2D1_INK_BEZIER_SEGMENT 일부를 구성하는 점 반경 쌍을 나타냅니다.
D2D1_INK_STYLE_PROPERTIES ID2D1InkStyle 개체에 사용되는 일반 펜 팁 셰이프 및 변환을 정의합니다.
D2D1_INPUT_DESCRIPTION 변환이 입력 텍스처에 설정할 수 있는 옵션에 대해 설명합니다.
D2D1_INPUT_ELEMENT_DESC 꼭짓점 레이아웃에 대한 단일 요소에 대한 설명입니다.
D2D1_LAYER_PARAMETERS 콘텐츠 범위, 마스크 정보, 불투명도 설정 및 계층 리소스에 대한 기타 옵션을 포함합니다. (D2D1_LAYER_PARAMETERS)
D2D1_LAYER_PARAMETERS1 콘텐츠 범위, 마스크 정보, 불투명도 설정 및 계층 리소스에 대한 기타 옵션을 포함합니다. (D2D1_LAYER_PARAMETERS1)
D2D1_LINEAR_GRADIENT_BRUSH_PROPERTIES ID2D1LinearGradientBrush에 대한 그라데이션 축의 시작점과 엔드포인트를 포함합니다.
D2D1_MAPPED_RECT ID2D1Bitmap1::Map API에서 매핑된 메모리에 대해 설명합니다.
D2D1_PIXEL_FORMAT 비트맵 또는 렌더링 대상에 대한 데이터 형식 및 알파 모드를 포함합니다.
D2D1_POINT_DESCRIPTION 경로 기하 도형의 점을 설명합니다.

D2D1_PRINT_CONTROL_PROPERTIES ID2D1PrintControl 개체의 생성 속성입니다.
D2D1_PROPERTY_BINDING 해당 속성을 가져와서 설정하는 함수 쌍에 대한 속성 바인딩을 정의합니다.
D2D1_QUADRATIC_BEZIER_SEGMENT 이차 베지어 세그먼트의 제어점과 끝점을 포함합니다.
D2D1_RADIAL_GRADIENT_BRUSH_PROPERTIES ID2D1RadialGradientBrush의 그라데이션 원점 오프셋과 그라데이션 타원의 크기와 위치를 포함합니다.
D2D1_RENDER_TARGET_PROPERTIES 렌더링 옵션(하드웨어 또는 소프트웨어), 픽셀 형식, DPI 정보, 원격 옵션 및 렌더링 대상에 대한 Direct3D 지원 요구 사항을 포함합니다.
D2D1_RENDERING_CONTROLS 이미징 효과 렌더러에 적용할 제한 사항에 대해 설명합니다.
D2D1_RESOURCE_TEXTURE_PROPERTIES 원래 리소스 텍스처를 만들 때 리소스 텍스처를 정의합니다.
D2D1_ROUNDED_RECT 둥근 사각형의 치수와 모서리 반경을 포함합니다.
D2D1_SIMPLE_COLOR_PROFILE 색 공간에 대한 간단한 설명입니다.
D2D1_STROKE_STYLE_PROPERTIES 도형을 윤곽선으로 표시하는 스트로크를 설명합니다. (D2D1_STROKE_STYLE_PROPERTIES)
D2D1_STROKE_STYLE_PROPERTIES1 도형을 윤곽선으로 표시하는 스트로크를 설명합니다. (D2D1_STROKE_STYLE_PROPERTIES1)
D2D1_SVG_LENGTH SVG 길이를 나타냅니다.

D2D1_SVG_PRESERVE_ASPECT_RATIO 모든 SVG preserveAspectRatio 설정을 나타냅니다.
D2D1_SVG_VIEWBOX SVG viewBox를 나타냅니다.
D2D1_TRANSFORMED_IMAGE_SOURCE_PROPERTIES 변환된 이미지 원본의 속성입니다.
D2D1_TRIANGLE 삼각형을 설명하는 세 가지 꼭짓점을 포함합니다.
D2D1_VERTEX_BUFFER_PROPERTIES 모든 꼭짓점 셰이더 정의에 대한 표준인 꼭짓점 버퍼의 속성을 정의합니다.
D2D1_VERTEX_RANGE 꼭짓점 버퍼의 전체 콘텐츠보다 적게 렌더링할 때 사용되는 꼭짓점 범위를 정의합니다.
TypeTraits TypeTraits(d2d1helper.h) 구조체에는 지정된 형식을 사용하여 데이터를 저장하는 Point, Size 및 Rect의 구현이 포함되어 있습니다.
TypeTraits TypeTraits(d2d1_1helper.h) 구조체에는 지정된 형식을 사용하여 데이터를 저장하는 Point, Size 및 Rect의 구현이 포함됩니다.

Direct3D 10 그래픽

아티클2023. 03. 09.

Direct3D 10 그래픽 기술 개요입니다.

Direct3D 10 그래픽을 개발하려면 다음 헤더가 필요합니다.

- d3d10.h
- d3d10_1.h
- d3d10_1shader.h
- d3d10effect.h
- d3d10misc.h
- d3d10sdklayers.h
- d3d10shader.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Direct3D 10 그래픽

열거형

D3D10_ASYNC_GETDATA_FLAG ID3D10Asynchronous::GetData의 동작을 제어하는 선택적 플래그입니다.
D3D10_BIND_FLAG 파이프라인에 리소스를 바인딩하는 방법을 식별합니다. (D3D10_BIND_FLAG)
D3D10_BLEND 혼합 옵션입니다. 혼합 옵션은 데이터 원본과 선택적 사전 혼합 작업을 식별합니다.
D3D10_BLEND_OP RGB 또는 알파 혼합 작업입니다. (D3D10_BLEND_OP)
D3D10_CLEAR_FLAG 지울 깊이 스텐실의 부분을 지정합니다. 일반적으로 ID3D10Device::ClearDepthStencilView와 함께 사용됩니다.

D3D10_COLOR_WRITE_ENABLE 혼합하는 동안 쓰기 가능한 렌더링 대상의 각 픽셀 구성 요소를 식별합니다. (D3D10_COLOR_WRITE_ENABLE)
D3D10_COMPARISON_FUNC 비교 옵션입니다. (D3D10_COMPARISON_FUNC)
D3D10_COUNTER 성능 카운터 유형입니다.
D3D10_COUNTER_TYPE 성능 카운터의 데이터 형식입니다. (D3D10_COUNTER_TYPE)
D3D10_CPU_ACCESS_FLAG 리소스에 허용되는 CPU 액세스 유형을 지정합니다. (D3D10_CPU_ACCESS_FLAG)
D3D10_CREATE_DEVICE_FLAG 디바이스 만들기 플래그입니다.
D3D10_CULL_MODE 특정 방향을 향한 삼각형이 그려지지 않음을 나타냅니다. (D3D10_CULL_MODE)
D3D10_DEPTH_WRITE_MASK 깊이 데이터를 쓰기 위한 깊이 스텐실 버퍼 부분을 식별합니다. (D3D10_DEPTH_WRITE_MASK)
D3D10_DEVICE_STATE_TYPES 효과 디바이스 상태 유형입니다.
D3D10_DRIVER_TYPE 디바이스 드라이버 유형입니다.
D3D10_DSV_DIMENSION 깊이 스텐실 보기에서 사용되는 리소스에 액세스하는 방법을 지정합니다. (D3D10_DSV_DIMENSION)
D3D10_FEATURE_LEVEL1 요청된 하드웨어 가속의 버전입니다.

D3D10_FILL_MODE 삼각형을 렌더링할 때 사용할 채우기 모드를 결정합니다. (D3D10_FILL_MODE)
D3D10_FILTER 텍스처 샘플링 중 필터링 옵션입니다. (D3D10_FILTER)
D3D10_FILTER_TYPE 배율 또는 축소 샘플러 필터의 형식입니다. (D3D10_FILTER_TYPE)
D3D10_FORMAT_SUPPORT 지정된 형식 및 지정된 디바이스에 대해 지원되는 리소스(ID3D10Device::CheckFormatSupport 참조).
D3D10_INPUT_CLASSIFICATION 입력 슬롯에 포함된 데이터 형식입니다. (D3D10_INPUT_CLASSIFICATION)
D3D10_MAP CPU에서 읽고 쓰기 위해 액세스할 리소스를 식별합니다. 애플리케이션은 이러한 플래그 중 하나 이상을 결합할 수 있습니다. (D3D10_MAP)
D3D10_MAP_FLAG GPU에서 사용하는 리소스에서 맵이 호출될 때 CPU가 응답하는 방법을 지정합니다.
D3D10_MESSAGE_CATEGORY 디버그 메시지의 범주입니다. (D3D10_MESSAGE_CATEGORY)
D3D10_MESSAGE_ID 정보 큐 필터를 설정하기 위한 메시지 디버그(D3D10_INFO_QUEUE_FILTER 참조) 이러한 메시지를 사용하여 스토리지 및 검색 필터를 통과하는 메시지 범주를 허용하거나 거부합니다.
D3D10_MESSAGE_SEVERITY 정보 큐에 대한 메시지 심각도 수준을 디버그합니다. (D3D10_MESSAGE_SEVERITY)
D3D10_QUERY 쿼리 형식입니다. (D3D10_QUERY)

D3D10_QUERY_MISC_FLAG 기타 쿼리 동작을 설명하는 플래그입니다. (D3D10_QUERY_MISC_FLAG)
D3D10_RAISE_FLAG 연속할 수 없는 예외에 대한 오류를 발생시키는 옵션입니다. (D3D10_RAISE_FLAG)
D3D10_RESOURCE_DIMENSION 사용 중인 리소스의 유형을 식별합니다. (D3D10_RESOURCE_DIMENSION)
D3D10_RESOURCE_MISC_FLAG 리소스에 대한 덜 일반적인 다른 옵션을 식별합니다.
D3D10_RTV_DIMENSION 렌더링 대상 보기에서 사용되는 리소스에 액세스하는 방법을 지정합니다.
D3D10_SHADER_DEBUG_REGTYPE 셰이더 레지스터 형식입니다.
D3D10_SHADER_DEBUG_SCOPETYPE 범위 유형입니다.
D3D10_SHADER_DEBUG_VARTYPE 범위의 함수와 변수를 구분합니다.
D3D10_STENCIL_OP 깊이 스텐실 테스트 중에 수행할 수 있는 스텐실 작업입니다. (D3D10_STENCIL_OP)
D3D10_TEXTURE_ADDRESS_MODE 텍스처 경계 밖에 있는 텍스처 좌표를 확인하는 기술을 식별합니다. (D3D10_TEXTURE_ADDRESS_MODE)
D3D10_TEXTURECUBE_FACE 큐브 텍스처의 다른 얼굴입니다. (D3D10_TEXTURECUBE_FACE)
D3D10_USAGE 렌더링하는 동안 예상되는 리소스 사용을 식별합니다. 사용량은 CPU 및/또는 GPU에서 리소스에 액세스할 수 있는지 여부를 직접 반영합니다.

함수

AddApplicationMessage 사용자 정의 메시지를 메시지 큐에 추가하고 해당 메시지를 전송하여 출력을 디버그합니다. (ID3D10InfoQueue.AddApplicationMessage)
AddMessage Direct3D 10 디버그 메시지를 메시지 큐에 추가하고 해당 메시지를 전송하여 출력을 디버그합니다.
AddRetrievalFilterEntries 검색 필터 스택의 맨 위에 스토리지 필터를 추가합니다. (ID3D10InfoQueue.AddRetrievalFilterEntries)
AddStorageFilterEntries 스토리지 필터 스택의 맨 위에 스토리지 필터를 추가합니다. (ID3D10InfoQueue.AddStorageFilterEntries)
적용 디바이스에 대한 패스에 포함된 상태를 설정합니다.
적용 현재 디바이스 상태에 상태 블록을 적용합니다. (ID3D10StateBlock.Apply)
AsBlend 효과 혼합 변수를 가져옵니다.
AsConstantBuffer 상수 버퍼를 가져옵니다. (ID3D10EffectVariable.AsConstantBuffer)
AsDepthStencil 깊이 스텐실 변수를 가져옵니다.
AsDepthStencilView 깊이 스텐실 보기 변수를 가져옵니다.
AsEffect 효과 풀을 만든 효과를 가져옵니다.

AsMatrix 행렬 변수를 가져옵니다.
AsRasterizer 래스터라이저 변수를 가져옵니다.
AsRenderTargetView render-target-view 변수를 가져옵니다.
AsSampler 샘플러 변수를 가져옵니다.
AsScalar 스칼라 변수를 가져옵니다.
AsShader 셰이더 변수를 가져옵니다.
AsShaderResource 셰이더 리소스 변수를 가져옵니다.
AsString 문자열 변수를 가져옵니다.
AsVector 벡터 변수를 가져옵니다.
시작 GPU 데이터 컬렉션을 시작합니다.
캡처 상태 블록에 포함된 상태의 현재 값을 캡처합니다. (ID3D10StateBlock.Capture)
CheckCounter 형식, 이름, 측정 단위 및 기존 카운터에 대한 설명을 가져옵니다. (ID3D10Device.CheckCounter)

CheckCounterInfo 카운터의 정보를 가져옵니다. (ID3D10Device.CheckCounterInfo)
CheckFormatSupport 설치된 비디오 디바이스에서 지정된 형식의 지원을 받습니다. (ID3D10Device.CheckFormatSupport)
CheckMultisampleQualityLevels 다중 샘플링 중에 사용할 수 있는 품질 수준 수를 가져옵니다. (ID3D10Device.CheckMultisampleQualityLevels)
ClearDepthStencilView 깊이 스텐실 리소스를 지웁니다. (ID3D10Device.ClearDepthStencilView)
ClearRenderTargetView 렌더링 대상의 모든 요소를 하나의 값으로 설정합니다. (ID3D10Device.ClearRenderTargetView)
ClearRetrievalFilter 검색 필터 스택의 맨 위에서 검색 필터를 제거합니다. (ID3D10InfoQueue.ClearRetrievalFilter)
ClearState 모든 기본 디바이스 설정을 복원합니다. 는 디바이스를 만들 때의 상태로 반환합니다.
ClearStorageFilter 스토리지 필터 스택의 맨 위에서 스토리지 필터를 제거합니다. (ID3D10InfoQueue.ClearStorageFilter)
ClearStoredMessages 메시지 큐에서 모든 메시지를 지웁니다. (ID3D10InfoQueue.ClearStoredMessages)
ComputeStateBlockMask 상태 변경을 허용/방지하기 위한 마스크를 생성합니다.
ComputeStateBlockMask 상태 변경 허용/방지를 위해 상태 블록 마스크를 계산합니다.

CopyResource GPU를 사용하여 원본 리소스의 전체 내용을 대상 리소스에 복사합니다. (ID3D10Device.CopyResource)
CopySubresourceRegion 원본 리소스에서 대상 리소스로 지역을 복사합니다. (ID3D10Device.CopySubresourceRegion)
CreateBlendState 출력-병합 단계에 대한 혼합 상태를 캡슐화하는 혼합 상태 개체를 만듭니다. (ID3D10Device.CreateBlendState)
CreateBlendState1 출력-병합 단계에 대한 혼합 상태를 캡슐화하는 혼합 상태 개체를 만듭니다. (ID3D10Device1.CreateBlendState1)
CreateBuffer 버퍼(꼭짓점 버퍼, 인덱스 버퍼 또는 셰이더 상수 버퍼)를 만듭니다.
CreateCounter GPU 성능을 측정하기 위한 카운터 개체를 만듭니다. (ID3D10Device.CreateCounter)
CreateDepthStencilState 출력-병합 단계에 대한 깊이 스텐실 테스트 정보를 캡슐화하는 깊이 스텐실 상태 개체를 만듭니다. (ID3D10Device.CreateDepthStencilState)
CreateDepthStencilView 리소스 데이터에 액세스하기 위한 깊이 스텐실 보기를 만듭니다. (ID3D10Device.CreateDepthStencilView)
CreateGeometryShader 기하 도형 셰이더를 만듭니다. (ID3D10Device.CreateGeometryShader)
CreateGeometryShaderWithStreamOutput 스트리밍 출력 버퍼에 쓸 수 있는 기하 도형 셰이더를 만듭니다. (ID3D10Device.CreateGeometryShaderWithStreamOutput)

CreateInputLayout 입력-어셈블러 단계에 대한 입력 버퍼 데이터를 설명하는 입력 레이아웃 개체를 만듭니다. (ID3D10Device.CreateInputLayout)
CreatePixelShader 픽셀 셰이더를 만듭니다. (ID3D10Device.CreatePixelShader)
CreatePredicate 조건자를 만듭니다. (ID3D10Device.CreatePredicate)
CreateQuery 이 인터페이스는 GPU에서 정보를 쿼리하는 메서드를 캡슐화합니다. (ID3D10Device.CreateQuery)
CreateRasterizerState 래스터라이저 단계에 동작 방법을 알려주는 래스터라이저 상태 개체를 만듭니다. (ID3D10Device.CreateRasterizerState)
CreateRenderTargetView 리소스 데이터에 액세스하기 위한 렌더링 대상 뷰를 만듭니다.
CreateSamplerState 텍스처에 대한 샘플링 정보를 캡슐화하는 샘플러 상태 개체를 만듭니다. (ID3D10Device.CreateSamplerState)
CreateShaderResourceView 리소스의 데이터에 액세스하기 위한 셰이더 리소스 뷰를 만듭니다. (ID3D10Device.CreateShaderResourceView)
CreateShaderResourceView1 리소스의 데이터에 액세스하기 위한 셰이더 리소스 뷰를 만듭니다. (ID3D10Device1.CreateShaderResourceView1)
CreateTexture1D 1D 텍스처 배열을 만듭니다(Texture1D 참조).
CreateTexture2D 2D 텍스처 배열을 만듭니다(Texture2D 참조).

CreateTexture3D 단일 3D 텍스처를 만듭니다(Texture3D 참조).
CreateVertexShader 컴파일된 셰이더에서 꼭짓점 셰이더 개체를 만듭니다. (ID3D10Device.CreateVertexShader)
D3D10CalcSubresource 텍스처에 대한 하위 리소스 인덱스 계산
D3D10CompileEffectFromMemory 효과를 컴파일합니다.
D3D10CompileShader HLSL 셰이더를 컴파일합니다.
D3D10CreateBlob 버퍼를 만듭니다. 이 함수를 사용하는 대신 D3DCreateBlob API를 사용하는 것이 좋습니다.
D3D10CreateDevice 디스플레이 어댑터를 나타내는 Direct3D 10.0 디바이스를 만듭니다.
D3D10CreateDevice1 디스플레이 어댑터를 나타내는 Direct3D 10.1 디바이스를 만듭니다.
D3D10CreateDeviceAndSwapChain Direct3D 10.0 디바이스 및 스왑 체인을 만듭니다.
D3D10CreateDeviceAndSwapChain1 Direct3D 10.1 디바이스 및 스왑 체인을 만듭니다.
D3D10CreateEffectFromMemory 컴파일된 효과를 포함하는 버퍼에서 ID3D10Effect를 만듭니다.
D3D10CreateEffectPoolFromMemory 효과 풀(또는 공유 메모리 위치)을 만들어 효과 간에 변수를 공유할 수 있도록 합니다.

D3D10CreateStateBlock 상태 블록을 만듭니다.
D3D10DisassembleEffect 컴파일된 효과를 어셈블리 명령 및 등록 할당이 포함된 텍스트 문자열로 디스어셈블하는 이 함수는 더 이상 사용되지 않습니다. 대신 D3DDisassemble10Effect를 사용합니다.
D3D10DisassembleShader 컴파일된 셰이더를 어셈블리 지침 및 등록 할당이 포함된 텍스트 문자열로 디스어셈블하는 이 함수는 더 이상 사용되지 않습니다. 대신 D3DDisassemble을 사용합니다.
D3D10GeometryShaderProfile 지정된 디바이스에 가장 적합한 기하 도형 셰이더 프로필을 가져옵니다.
D3D10GetInputAndOutputSignatureBlob 셰이더 서명이 포함된 버퍼를 가져옵니다.
D3D10GetInputSignatureBlob 셰이더 입력 서명이 포함된 버퍼를 가져옵니다.
D3D10GetOutputSignatureBlob 셰이더 출력 서명이 포함된 버퍼를 가져옵니다.
D3D10GetPixelShaderProfile 지정된 디바이스에 가장 적합한 픽셀 셰이더 프로필을 가져옵니다.
D3D10GetShaderDebugInfo 셰이더 디버그 정보를 가져옵니다. 디버그 정보는 D3D10CompileShader에서 생성되며 셰이더 본문에 포함됩니다.
D3D10GetVertexShaderProfile 지정된 디바이스에 가장 적합한 꼭짓점 셰이더 프로필을 가져옵니다.
D3D10PreprocessShader 컴파일된 셰이더에서 찾을 수 있는 셰이더 토큰이 포함된 셰이더 텍스트 문자열을 생성합니다.

D3D10ReflectShader 컴파일된 셰이더에 대한 정보를 검색하기 위해 셰이더 리플렉션 개체를 만드는 이 함수는 더 이상 사용되지 않습니다. 대신 D3DReflect를 사용합니다.
D3D10StateBlockMaskDifference 두 개의 상태 블록 마스크를 비트 XOR와 결합합니다.
D3D10StateBlockMaskDisableAll 상태 블록 마스크를 사용하여 모든 상태 캡처를 사용하지 않도록 설정합니다.
D3D10StateBlockMaskDisableCapture 상태 블록 마스크를 사용하여 상태 캡처를 사용하지 않도록 설정합니다.
D3D10StateBlockMaskEnableAll 상태 블록 마스크를 사용하여 모든 상태 변수를 캡처하고 적용합니다.
D3D10StateBlockMaskEnableCapture 상태 블록 마스크에서 상태 값 범위를 사용하도록 설정합니다.
D3D10StateBlockMaskGetSetting 상태 블록 마스크의 요소를 가져옵니다. 는 요소를 캡처하고 적용하기 위해 마스크에서 허용하는지 여부를 결정합니다.
D3D10StateBlockMaskIntersect 두 개의 상태 블록 마스크를 비트 AND와 결합합니다.
D3D10StateBlockMaskUnion 두 개의 상태 블록 마스크를 비트 OR과 결합합니다.
그릴 인덱싱되지 않은 인스턴스가 아닌 기본 형식을 그립니다. (ID3D10Device.Draw)
DrawAuto 기하 도형 셰이더 단계에서 만든 알 수 없는 크기의 기하 도형을 그립니다. 설명을 참조하세요.
DrawIndexed 인덱싱된 인스턴스가 아닌 기본 형식을 그립니다. (ID3D10Device.DrawIndexed)

DrawIndexedInstanced 인덱싱된 인스턴스화된 기본 형식을 그립니다. (ID3D10Device.DrawIndexedInstanced)
DrawInstanced 인덱싱되지 않은 인스턴스화된 기본 형식을 그립니다. (ID3D10Device.DrawInstanced)
End GPU 데이터 컬렉션을 종료합니다.
Enter 디바이스의 중요 섹션을 입력합니다. (ID3D10Multithread.Enter)
플러시 명령 버퍼의 큐에 대기 중인 명령을 GPU로 보냅니다.
GenerateMips 지정된 셰이더 리소스에 대한 mipmap을 생성합니다. (ID3D10Device.GenerateMips)
GetAnnotationByIndex ID3D10EffectPass::GetAnnotationByIndex(d3d10effect.h) 메서드는 인덱스별로 주석을 가져옵니다.
GetAnnotationByIndex ID3D10EffectTechnique::GetAnnotationByIndex(d3d10effect.h) 메서드는 인덱스별로 주석을 가져옵니다.
GetAnnotationByIndex ID3D10EffectVariable::GetAnnotationByIndex(d3d10effect.h) 메서드는 인덱스별로 주석을 가져옵니다.
GetAnnotationByName 이름으로 주석을 가져옵니다. (ID3D10EffectPass.GetAnnotationByName)
GetAnnotationByName 이름으로 주석을 가져옵니다. (ID3D10EffectTechnique.GetAnnotationByName)
GetAnnotationByName 이름으로 주석을 가져옵니다. (ID3D10EffectVariable.GetAnnotationByName)

GetBackingStore 혼합 상태 변수에 대한 포인터를 가져옵니다.
GetBackingStore 깊이 스텐실 상태가 포함된 변수에 대한 포인터를 가져옵니다.
GetBackingStore 래스터라이저 상태가 포함된 변수에 대한 포인터를 가져옵니다.
GetBackingStore 샘플러 상태가 포함된 변수에 대한 포인터를 가져옵니다.
GetBitwiseInstructionCount 비트 명령 수를 가져옵니다. (ID3D10ShaderReflection1.GetBitwiseInstructionCount)
GetBlendState 혼합 상태 인터페이스에 대한 포인터를 가져옵니다.
GetBool 부울 변수를 가져옵니다.
GetBoolArray 부울 변수 배열을 가져옵니다.
GetBoolVector 부울 데이터가 포함된 4개 구성 요소 벡터를 가져옵니다.
GetBoolVectorArray 부울 데이터가 포함된 4개 구성 요소 벡터의 배열을 가져옵니다.
GetBreakOnCategory 해당 범주의 메시지가 스토리지 필터를 통과할 때 중단되는 메시지 범주를 가져옵니다. (ID3D10InfoQueue.GetBreakOnCategory)
GetBreakOnID 해당 식별자가 있는 메시지가 스토리지 필터를 통과할 때 중단되는 메시지 식별자를 가져옵니다. (ID3D10InfoQueue.GetBreakOnID)

GetBreakOnSeverity 심각도 수준이 있는 메시지가 스토리지 필터를 통과할 때 중단되는 메시지 심각도 수준을 가져옵니다. (ID3D10InfoQueue.GetBreakOnSeverity)
GetConstantBuffer 상수 버퍼를 가져옵니다.
GetConstantBufferByIndex ID3D10ShaderReflection::GetConstantBufferByIndex(d3d10shader.h) 메서드는 인덱스별로 상수 버퍼를 가져옵니다.
GetConstantBufferByIndex ID3D10Effect::GetConstantBufferByIndex(d3d10effect.h) 메서드는 인덱스별로 상수 버퍼를 가져옵니다.
GetConstantBufferByName 이름으로 상수 버퍼를 가져옵니다. (ID3D10ShaderReflection.GetConstantBufferByName)
GetConstantBufferByName 이름으로 상수 버퍼를 가져옵니다. (ID3D10Effect.GetConstantBufferByName)
GetConversionInstructionCount 셰이더에 사용되는 변환 명령 수를 가져옵니다.
GetCreationFlags D3D10CreateDevice를 사용하여 디바이스를 만들기 위해 호출하는 동안 사용되는 플래그를 가져옵니다.
GetData GPU에서 비동기적으로 데이터를 가져옵니다.
GetDataSize ID3D10Asynchronous::GetData를 호출할 때 출력되는 데이터 크기(바이트)를 가져옵니다.
GetDepthStencil 깊이 스텐실 보기 리소스를 가져옵니다.

GetDepthStencilArray 깊이 스텐실 뷰 리소스 배열을 가져옵니다.
GetDepthStencilState 깊이 스텐실 인터페이스에 대한 포인터를 가져옵니다.
GetDesc 혼합 상태를 가져옵니다. (ID3D10BlendState.GetDesc)
GetDesc 버퍼 리소스의 속성을 가져옵니다. (ID3D10Buffer.GetDesc)
GetDesc 카운터 설명을 가져옵니다. (ID3D10Counter.GetDesc)
GetDesc 깊이 스텐실 상태를 가져옵니다.
GetDesc 깊이 스텐실 보기를 가져옵니다. (ID3D10DepthStencilView.GetDesc)
GetDesc 쿼리 설명을 가져옵니다. (ID3D10Query.GetDesc)
GetDesc 래스터라이저 상태 개체의 속성을 가져옵니다.
GetDesc 렌더링 대상 뷰의 속성을 가져옵니다. (ID3D10RenderTargetView.GetDesc)
GetDesc 샘플러 상태를 가져옵니다.
GetDesc 셰이더 리소스 뷰의 설명을 가져옵니다. (ID3D10ShaderResourceView.GetDesc)

GetDesc 텍스처 리소스의 속성을 가져옵니다. (ID3D10Texture1D.GetDesc)
GetDesc 텍스처 리소스의 속성을 가져옵니다. (ID3D10Texture2D.GetDesc)
GetDesc 텍스처 리소스의 속성을 가져옵니다. (ID3D10Texture3D.GetDesc)
GetDesc 셰이더 설명을 가져옵니다. (ID3D10ShaderReflection.GetDesc)
GetDesc 상수 버퍼 설명을 가져옵니다. (ID3D10ShaderReflectionConstantBuffer.GetDesc)
GetDesc 셰이더-리플렉션 변수 형식에 대한 설명을 가져옵니다. (ID3D10ShaderReflectionType.GetDesc)
GetDesc 셰이더 변수 설명을 가져옵니다. (ID3D10ShaderReflectionVariable.GetDesc)
GetDesc 효과 설명을 가져옵니다.
GetDesc 패스 설명을 가져옵니다.
GetDesc 기술 설명을 가져옵니다.
GetDesc 효과 유형 설명을 가져옵니다.
GetDesc 설명을 가져옵니다.

GetDesc1 혼합 상태를 가져옵니다. (ID3D10BlendState1.GetDesc1)
GetDesc1 셰이더 리소스 뷰의 설명을 가져옵니다. (ID3D10ShaderResourceView1.GetDesc1)
GetDevice 이 인터페이스를 만든 디바이스에 대한 포인터를 가져옵니다. (ID3D10DeviceChild.GetDevice)
GetDevice 효과를 만든 디바이스를 가져옵니다.
GetDevice 디바이스를 가져옵니다.
GetDeviceRemovedReason 디바이스가 제거된 이유를 확인합니다. (ID3D10Device.GetDeviceRemovedReason)
GetElement 배열 요소를 가져옵니다.
GetEvictionPriority 리소스의 제거 우선 순위를 가져옵니다. (ID3D10Resource.GetEvictionPriority)
GetExceptionMode 예외 모드 플래그를 가져옵니다. (ID3D10Device.GetExceptionMode)
GetFeatureLevel 하드웨어 디바이스의 기능 수준을 가져옵니다. (ID3D10Device1.GetFeatureLevel)
GetFeatureMask 어떤 디버그 기능이 켜지거나 꺼져 있는지 나타내는 플래그의 비트 필드를 가져옵니다. (ID3D10Debug.GetFeatureMask)
GetFloat 부동 소수점 변수를 가져옵니다.

GetFloatArray 부동 소수점 변수의 배열을 가져옵니다.
GetFloatVector 부동 소수점 데이터가 포함된 4개 구성 요소 벡터를 가져옵니다.
GetFloatVectorArray 부동 소수점 데이터가 포함된 4개 구성 요소 벡터의 배열을 가져옵니다.
GetGeometryShader 기하 도형 셰이더를 가져옵니다.
GetGeometryShaderDesc 기하 도형 셰이더 설명을 가져옵니다.
GetGSInputPrimitive geometry-shader input-primitive 설명을 가져옵니다. (ID3D10ShaderReflection1.GetGSInputPrimitive)
GetInputParameterDesc 셰이더에 대한 입력 매개 변수 설명을 가져옵니다. (ID3D10ShaderReflection.GetInputParameterDesc)
GetInputSignatureElementDesc 입력 서명 설명을 가져옵니다.
GetInt 정수 변수를 가져옵니다.
GetIntArray 정수 변수 배열을 가져옵니다.
GetIntVector 정수 데이터가 포함된 4개 구성 요소 벡터를 가져옵니다.
GetIntVectorArray 정수 데이터가 포함된 4개 구성 요소 벡터의 배열을 가져옵니다.

GetMatrix 행렬을 가져옵니다.
GetMatrixArray 행렬 배열을 가져옵니다.
GetMatrixTranspose 부동 소수점 행렬을 변환하고 가져옵니다.
GetMatrixTransposeArray 부동 소수점 행렬의 배열을 변환하고 가져옵니다.
GetMemberByIndex 인덱스별로 구조체 멤버를 가져옵니다.
GetMemberByName 이름으로 구조체 멤버를 가져옵니다.
GetMemberBySemantic 의미 체계로 구조체 멤버를 가져옵니다.
GetMemberName 멤버의 이름을 가져옵니다.
GetMemberSemantic 멤버에 연결된 의미 체계를 가져옵니다.
GetMemberTypeByIndex ID3D10ShaderReflectionType::GetMemberTypeByIndex(d3d10shader.h) 메서드는 인덱스별로 셰이더-리플렉션 변수 형식을 가져옵니다.
GetMemberTypeByIndex 인덱스별로 멤버 형식을 가져옵니다.
GetMemberTypeByName 이름으로 셰이더-리플렉션 변수 형식을 가져옵니다. (ID3D10ShaderReflectionType.GetMemberTypeByName)

GetMemberTypeByName 이름으로 멤버 형식을 가져옵니다.
GetMemberTypeBySemantic 의미 체계를 사용하여 멤버 형식을 가져옵니다.
GetMemberTypeName 셰이더-리플렉션 변수 형식을 가져옵니다. (ID3D10ShaderReflectionType.GetMemberTypeName)
GetMessage 메시지 큐에서 메시지를 가져옵니다. (ID3D10InfoQueue.GetMessage)
GetMessageCountLimit 메시지 큐에 추가할 수 있는 최대 메시지 수를 가져옵니다. (ID3D10InfoQueue.GetMessageCountLimit)
GetMovcInstructionCount Movc 명령 수를 가져옵니다. (ID3D10ShaderReflection1.GetMovcInstructionCount)
GetMovInstructionCount Mov 명령 수를 가져옵니다. (ID3D10ShaderReflection1.GetMovInstructionCount)
GetMultithreadProtected 다중 스레딩이 켜져 있는지 확인합니다.
GetMuteDebugOutput 디버그 출력을 켜거나 끄는 부울을 가져옵니다. (ID3D10InfoQueue.GetMuteDebugOutput)
GetNumMessagesAllowedByStorageFilter 스토리지 필터를 통과하도록 허용된 메시지 수를 가져옵니다. (ID3D10InfoQueue.GetNumMessagesAllowedByStorageFilter)
GetNumMessagesDeniedByStorageFilter 스토리지 필터를 통해 전달이 거부된 메시지 수를 가져옵니다. (ID3D10InfoQueue.GetNumMessagesDeniedByStorageFilter)

GetNumMessagesDiscardedByMessageCountLimit 메시지 수 제한으로 인해 삭제된 메시지 수를 가져옵니다. (ID3D10InfoQueue.GetNumMessagesDiscardedByMessageCountLimit)
GetNumStoredMessages 현재 메시지 큐에 저장된 메시지 수를 가져옵니다. (ID3D10InfoQueue.GetNumStoredMessages)
GetNumStoredMessagesAllowedByRetrievalFilter 검색 필터를 통과할 수 있는 메시지 수를 가져옵니다. (ID3D10InfoQueue.GetNumStoredMessagesAllowedByRetrievalFilter)
GetOutputParameterDesc 셰이더에 대한 출력 매개 변수 설명을 가져옵니다. (ID3D10ShaderReflection.GetOutputParameterDesc)
GetOutputSignatureElementDesc 출력 서명 설명을 가져옵니다.
GetParentConstantBuffer 상수 버퍼를 가져옵니다. (ID3D10EffectVariable.GetParentConstantBuffer)
GetPassByIndex 인덱스로 패스를 가져옵니다.
GetPassByName 이름으로 패스를 가져옵니다.
GetPixelShader 픽셀 셰이더를 가져옵니다.
GetPixelShaderDesc 픽셀 셰이더 설명을 가져옵니다.
GetPredication 렌더링 조건자 상태를 가져옵니다. (ID3D10Device.GetPredication)

GetPresentPerRenderOpDelay Present가 호출된 후 절전 모드로의 시간(밀리초)을 가져옵니다.
GetPrivateData GUID와 연결된 디바이스에서 데이터를 가져옵니다.
GetPrivateData 디바이스 자식에서 애플리케이션 정의 데이터를 가져옵니다. (ID3D10DeviceChild.GetPrivateData)
GetRasterizerState 래스터라이저 인터페이스에 대한 포인터를 가져옵니다.
GetRawValue 데이터를 가져옵니다.
GetRenderTarget 렌더링 대상을 가져옵니다.
GetRenderTargetArray 렌더링 대상 배열을 가져옵니다.
GetResource 이 보기를 통해 액세스되는 리소스를 가져옵니다. (ID3D10View.GetResource)
GetResource 셰이더 리소스를 가져옵니다.
GetResourceArray 셰이더 리소스 배열을 가져옵니다.
GetResourceBindingDesc 셰이더에 바인딩된 리소스에 대한 설명을 가져옵니다.
GetResourceBindingDescByName 이름으로 리소스 바인딩 설명을 가져옵니다.

GetRetrievalFilter 검색 필터 스택의 맨 위에 있는 검색 필터를 가져옵니다. (ID3D10InfoQueue.GetRetrievalFilter)
GetRetrievalFilterStackSize 검색 필터 스택의 크기를 바이트 단위로 가져옵니다. (ID3D10InfoQueue.GetRetrievalFilterStackSize)
GetSampler 샘플러 인터페이스에 대한 포인터를 가져옵니다.
GetShaderDesc 셰이더 설명을 가져옵니다. (ID3D10EffectShaderVariable.GetShaderDesc)
GetStorageFilter 스토리지 필터 스택의 맨 위에 있는 스토리지 필터를 가져옵니다. (ID3D10InfoQueue.GetStorageFilter)
GetStorageFilterStackSize 스토리지 필터 스택의 크기를 바이트 단위로 가져옵니다. (ID3D10InfoQueue.GetStorageFilterStackSize)
GetString 문자열을 가져옵니다.
GetStringArray 문자열 배열을 가져옵니다.
GetSwapChain 런타임에서 Present를 자동으로 호출하는 데 사용할 스왑 체인을 가져옵니다.
GetTechniqueByIndex 인덱스별로 기술을 가져옵니다.
GetTechniqueByName 이름으로 기술을 가져옵니다.

GetTextFilterSize 이 메서드가 구현되지 않은 경우 (ID3D10Device.GetTextFilterSize)
GetTextureBuffer 텍스처 버퍼를 가져옵니다.
GetType 리소스의 형식을 가져옵니다. (ID3D10Resource.GetType)
GetType 셰이더 변수 형식을 가져옵니다. (ID3D10ShaderReflectionVariable.GetType)
GetType 형식 정보를 가져옵니다.
GetUseRef 사용 중인 디바이스의 유형을 나타내는 부울 값을 가져옵니다.
GetVariableByIndex ID3D10ShaderReflectionConstantBuffer::GetVariableByIndex(d3d10shader.h) 메서드는 인덱스별로 셰이더 리플렉션 변수를 가져옵니다.
GetVariableByIndex 인덱스별로 변수를 가져옵니다.
GetVariableByName 이름으로 셰이더 리플렉션 변수를 가져옵니다. (ID3D10ShaderReflectionConstantBuffer.GetVariableByName)
GetVariableByName 이름으로 변수를 가져옵니다.
GetVariableByName 이름으로 변수를 가져옵니다. (ID3D10ShaderReflection1.GetVariableByName)
GetVariableBySemantic 의미 체계로 변수를 가져옵니다.

GetVertexShader 꼭짓점 셰이더를 가져옵니다.
GetVertexShaderDesc 꼭짓점 셰이더 설명을 가져옵니다.
GSGetConstantBuffers 기하 도형 셰이더 파이프라인 단계에서 사용하는 상수 버퍼를 가져옵니다. (ID3D10Device.GSGetConstantBuffers)
GSGetSamplers 기하 도형 셰이더 파이프라인 단계에서 샘플러 상태 배열을 가져옵니다.
GSGetShader 디바이스에서 현재 설정된 기하 도형 셰이더를 가져옵니다. (ID3D10Device.GSGetShader)
GSGetShaderResources 기하 도형 셰이더 리소스를 가져옵니다. (ID3D10Device.GSGetShaderResources)
GSSetConstantBuffers 기하 도형 셰이더 파이프라인 단계에서 사용되는 상수 버퍼를 설정합니다.
GSSetSamplers 샘플러 상태 배열을 기하 도형 셰이더 파이프라인 단계로 설정합니다. (ID3D10Device.GSSetSamplers)
GSSetShader 기하 도형 셰이더를 디바이스로 설정합니다. (ID3D10Device.GSSetShader)
GSSetShaderResources 셰이더 리소스 배열을 기하 도형 셰이더 단계에 바인딩합니다. (ID3D10Device.GSSetShaderResources)
IAGetIndexBuffer 입력 어셈블러 단계에 바인딩된 인덱스 버퍼에 대한 포인터를 가져옵니다. (ID3D10Device.IAGetIndexBuffer)

IAGetInputLayout 입력 어셈블러 단계에 바인딩된 입력 레이아웃 개체에 대한 포인터를 가져옵니다. (ID3D10Device.IAGetInputLayout)
IAGetPrimitiveTopology 기본 형식 및 입력 어셈블러 단계에 대한 입력 데이터를 설명하는 데이터 순서에 대한 정보를 가져옵니다. (ID3D10Device.IAGetPrimitiveTopology)
IAGetVertexBuffers 입력 어셈블러 단계에 바인딩된 꼭짓점 버퍼를 가져옵니다. (ID3D10Device.IAGetVertexBuffers)
IASetIndexBuffer 인덱스 버퍼를 입력 어셈블러 단계에 바인딩합니다. (ID3D10Device.IASetIndexBuffer)
IASetInputLayout 입력 레이아웃 개체를 입력 어셈블러 단계에 바인딩합니다. (ID3D10Device.IASetInputLayout)
IASetPrimitiveTopology 기본 형식 및 입력 어셈블러 단계에 대한 입력 데이터를 설명하는 데이터 순서에 대한 정보를 바인딩합니다. (ID3D10Device.IASetPrimitiveTopology)
IASetVertexBuffers 꼭짓점 버퍼 배열을 입력 어셈블러 단계에 바인딩합니다. (ID3D10Device.IASetVertexBuffers)
IsLevel9Shader 셰이더가 Direct3D 9 모드에서 Direct3D 10에서 컴파일되었는지 여부를 나타냅니다.
IsOptimized 효과를 테스트하여 리플렉션 메타데이터가 메모리에서 제거되었는지 확인합니다.
IsPool 효과를 테스트하여 메모리 풀의 일부인지 확인합니다.
IsSampleFrequencyShader 픽셀 셰이더가 픽셀 빈도 또는 샘플 빈도를 실행할 것인지 여부를 나타냅니다.

IsValid 효과를 테스트하여 유효한 구문이 포함되어 있는지 확인합니다.
IsValid 패스를 테스트하여 유효한 구문이 포함되어 있는지 확인합니다.
IsValid 기술을 테스트하여 유효한 구문이 포함되어 있는지 확인합니다.
IsValid 효과 유형이 유효한지 테스트합니다.
IsValid 데이터 형식을 저장된 데이터와 비교합니다.
나가기 디바이스의 중요 섹션을 그대로 둡니다. (ID3D10Multithread.Leave)
Map 리소스에 포함된 데이터에 대한 포인터를 가져와서 리소스에 대한 GPU 액세스를 거부합니다.
Map 하위 리소스에 포함된 데이터에 대한 포인터를 가져와서 해당 하위 리소스에 대한 GPU 액세스를 거부합니다.
Map 하위 리소스에 포함된 데이터에 대한 포인터를 가져와서 해당 하위 리소스에 대한 GPU 액세스를 거부합니다. (ID3D10Texture2D.Map)
Map 하위 리소스에 포함된 데이터에 대한 포인터를 가져와서 해당 하위 리소스에 대한 GPU 액세스를 거부합니다. (ID3D10Texture3D.Map)
OMGetBlendState 출력 병합 단계의 혼합 상태를 가져옵니다. (ID3D10Device.OMGetBlendState)

OMGetDepthStencilState 출력 병합 단계의 깊이 스텐실 상태를 가져옵니다. (ID3D10Device.OMGetDepthStencilState)
OMGetRenderTargets 출력-병합 단계에서 사용할 수 있는 렌더링 대상 및 깊이 스텐실 버퍼에 대한 포인터를 가져옵니다.
OMSetBlendState 출력 병합 단계의 혼합 상태를 설정합니다. (ID3D10Device.OMSetBlendState)
OMSetDepthStencilState 출력 병합 단계의 깊이 스텐실 상태를 설정합니다. (ID3D10Device.OMSetDepthStencilState)
OMSetRenderTargets 하나 이상의 렌더링 대상과 깊이 스텐실 버퍼를 출력 병합 단계에 바인딩합니다.
OpenSharedResource 디바이스에 다른 Direct3d 디바이스에서 만든 공유 리소스에 대한 액세스 권한을 부여합니다.
Optimize 효과에 필요한 메모리 양을 최소화합니다.
PopRetrievalFilter 검색 필터 스택의 맨 위에서 검색 필터를 팝합니다. (ID3D10InfoQueue.PopRetrievalFilter)
PopStorageFilter 스토리지 필터 스택의 맨 위에서 스토리지 필터를 팝합니다. (ID3D10InfoQueue.PopStorageFilter)
PSGetConstantBuffers 픽셀 셰이더 파이프라인 단계에서 사용하는 상수 버퍼를 가져옵니다. (ID3D10Device.PSGetConstantBuffers)
PSGetSamplers 픽셀 셰이더 파이프라인 단계에서 샘플러 상태 배열을 가져옵니다. (ID3D10Device.PSGetSamplers)
PSGetShader 디바이스에서 현재 설정된 픽셀 셰이더를 가져옵니다. (ID3D10Device.PSGetShader)

PSGetShaderResources 픽셀 셰이더 리소스를 가져옵니다. (ID3D10Device.PSGetShaderResources)
PSSetConstantBuffers 픽셀 셰이더 파이프라인 단계에서 사용하는 상수 버퍼를 설정합니다.
PSSetSamplers 샘플러 상태 배열을 픽셀 셰이더 파이프라인 단계로 설정합니다. (ID3D10Device.PSSetSamplers)
PSSetShader 픽셀 셰이더를 디바이스로 설정합니다. (ID3D10Device.PSSetShader)
PSSetShaderResources 셰이더 리소스 배열을 픽셀 셰이더 단계에 바인딩합니다. (ID3D10Device.PSSetShaderResources)
PushCopyOfRetrievalFilter 검색 필터 스택의 맨 위에 있는 검색 필터의 복사본을 검색 필터 스택에 푸시합니다. (ID3D10InfoQueue.PushCopyOfRetrievalFilter)
PushCopyOfStorageFilter 현재 스토리지 필터 스택의 맨 위에 있는 스토리지 필터 복사본을 스토리지 필터 스택에 푸시합니다. (ID3D10InfoQueue.PushCopyOfStorageFilter)
PushEmptyRetrievalFilter 빈 검색 필터를 검색 필터 스택에 푸시합니다. (ID3D10InfoQueue.PushEmptyRetrievalFilter)
PushEmptyStorageFilter 빈 스토리지 필터를 스토리지 필터 스택에 푸시합니다. (ID3D10InfoQueue.PushEmptyStorageFilter)
PushRetrievalFilter 검색 필터를 검색 필터 스택에 푸시합니다. (ID3D10InfoQueue.PushRetrievalFilter)
PushStorageFilter 스토리지 필터를 스토리지 필터 스택에 푸시합니다. (ID3D10InfoQueue.PushStorageFilter)

ReleaseAllDeviceObjects 디바이스 개체에 대한 모든 참조를 해제합니다.
ResolveSubresource 다중 샘플링된 리소스를 다중 샘플링되지 않은 리소스에 복사합니다. 이 API는 한 렌더링 패스의 결과 렌더링 대상을 두 번째 렌더링 패스에 대한 입력으로 다시 사용할 때 가장 유용합니다.
RSGetScissorRects 래스터라이저 단계에 바인딩된 가위 사각형 배열을 가져옵니다. (ID3D10Device.RSGetScissorRects)
RSGetState 파이프라인의 래스터라이저 단계에서 래스터라이저 상태를 가져옵니다. (ID3D10Device.RSGetState)
RSGetViewports 래스터라이저 단계에 바인딩된 뷰포트 배열 가져오기
RSSetScissorRects 가위 사각형 배열을 래스터라이저 단계에 바인딩합니다. (ID3D10Device.RSSetScissorRects)
RSSetState 파이프라인의 래스터라이저 단계에 대한 래스터라이저 상태를 설정합니다. (ID3D10Device.RSSetState)
RSSetViewports 뷰포트 배열을 파이프라인의 래스터라이저 단계에 바인딩합니다. (ID3D10Device.RSSetViewports)
SetBool 부울 변수를 설정합니다.
SetBoolArray 부울 변수 배열을 설정합니다.
SetBoolVector 부울 데이터가 포함된 4개 구성 요소 벡터를 설정합니다.

SetBoolVectorArray 부울 데이터를 포함하는 4개 구성 요소 벡터의 배열을 설정합니다.
SetBreakOnCategory 해당 범주의 메시지가 스토리지 필터를 통과할 때 중단되도록 메시지 범주를 설정합니다. (ID3D10InfoQueue.SetBreakOnCategory)
SetBreakOnID 해당 식별자가 있는 메시지가 스토리지 필터를 통과할 때 중단되도록 메시지 식별자를 설정합니다. (ID3D10InfoQueue.SetBreakOnID)
SetBreakOnSeverity 해당 심각도 수준의 메시지가 스토리지 필터를 통과할 때 중단되도록 메시지 심각도 수준을 설정합니다. (ID3D10InfoQueue.SetBreakOnSeverity)
SetConstantBuffer 상수 버퍼를 설정합니다.
SetDepthStencil depth-stencil-view 리소스를 설정합니다.
SetDepthStencilArray depth-stencil-view 리소스의 배열을 설정합니다.
SetEvictionPriority 리소스의 제거 우선 순위를 설정합니다. (ID3D10Resource.SetEvictionPriority)
SetExceptionMode 예외 모드 플래그를 가져옵니다. (ID3D10Device.SetExceptionMode)
SetFeatureMask 디버그 기능을 켜고 끄는 플래그의 비트 필드를 설정합니다.
SetFloat 부동 소수점 변수를 설정합니다.

SetFloatArray 부동 소수점 변수의 배열을 설정합니다.
SetFloatVector 부동 소수점 데이터가 포함된 4개 구성 요소 벡터를 설정합니다.
SetFloatVectorArray 부동 소수점 데이터가 포함된 4개 구성 요소 벡터의 배열을 설정합니다.
SetInt 정수 변수를 설정합니다.
SetIntArray 정수 변수 배열을 설정합니다.
SetIntVector 정수 데이터가 포함된 4개 구성 요소 벡터를 설정합니다.
SetIntVectorArray 정수 데이터가 포함된 4개 구성 요소 벡터의 배열을 설정합니다.
SetMatrix 부동 소수점 행렬을 설정합니다.
SetMatrixArray 부동 소수점 행렬의 배열을 설정합니다.
SetMatrixTranspose 부동 소수점 행렬을 변환하고 설정합니다.
SetMatrixTransposeArray 부동 소수점 행렬의 배열을 변환하고 설정합니다.
SetMessageCountLimit 메시지 큐에 추가할 수 있는 최대 메시지 수를 설정합니다. (ID3D10InfoQueue.SetMessageCountLimit)

SetMultithreadProtected 다중 스레딩을 켜거나 끕니다.
SetMuteDebugOutput 디버그 출력을 켜거나 끄는 부울을 설정합니다. (ID3D10InfoQueue.SetMuteDebugOutput)
SetPredication 렌더링 조건자를 설정합니다. (ID3D10Device.SetPredication)
SetPresentPerRenderOpDelay Present가 호출된 후 절전 모드로 설정된 시간(밀리초)입니다.
SetPrivateData 디바이스에 데이터를 설정하고 해당 데이터를 GUID와 연결합니다. (ID3D10Device.SetPrivateData)
SetPrivateData 애플리케이션 정의 데이터를 디바이스 자식으로 설정하고 해당 데이터를 애플리케이션 정의 GUID와 연결합니다. (ID3D10DeviceChild.SetPrivateData)
SetPrivateDataInterface IUnknown 파생 인터페이스를 이 디바이스와 연결하고 해당 인터페이스를 애플리케이션 정의 GUID와 연결합니다.
SetPrivateDataInterface IUnknown 파생 인터페이스를 이 디바이스 자식과 연결하고 해당 인터페이스를 애플리케이션 정의 GUID와 연결합니다. (ID3D10DeviceChild.SetPrivateDataInterface)
SetRawValue 데이터를 설정합니다.
SetRenderTarget 렌더링 대상을 설정합니다.
SetRenderTargetArray 렌더링 대상 배열을 설정합니다.

SetResource 셰이더 리소스를 설정합니다.
SetResourceArray 셰이더 리소스 배열을 설정합니다.
SetSwapChain 런타임에서 Present를 자동으로 호출하는 데 사용할 스왑 체인을 설정합니다.
SetTextFilterSize 이 메서드가 구현되지 않은 경우 (ID3D10Device.SetTextFilterSize)
SetTextureBuffer 텍스처 버퍼를 설정합니다.
SetUseRef 하드웨어와 소프트웨어 디바이스 간에 전환합니다.
SOGetTargets 파이프라인의 StreamOutput 단계에 대한 대상 출력 버퍼를 가져옵니다.
SOSetTargets StreamOutput 단계에 대한 대상 출력 버퍼를 설정하여 파이프라인이 데이터를 스트림아웃하도록 설정/해제합니다.
Unmap ID3D10Buffer::Map 및 리소스에 대한 다시 활성화 가능한 GPU 액세스로 검색된 리소스에 대한 포인터를 무효화합니다.
Unmap ID3D10Texture1D::Map에서 검색된 리소스에 대한 포인터를 무효화하고 해당 리소스에 대한 GPU의 액세스를 다시 사용하도록 설정합니다.
Unmap ID3D10Texture2D::Map에서 검색한 리소스에 대한 포인터를 무효화하고 리소스에 대한 GPU 액세스를 다시 사용하도록 설정합니다.

Unmap ID3D10Texture3D::Map에서 검색한 리소스에 대한 포인터를 무효화하고 리소스에 대한 GPU의 액세스를 다시 사용하도록 설정합니다.
UpdateSubresource CPU는 메모리에서 mappable이 아닌 메모리에서 만든 하위 리소스로 데이터를 복사합니다. 설명을 참조하세요.
유효성 검사 파이프라인 상태의 유효성을 확인합니다.
VSGetConstantBuffers 꼭짓점 셰이더 파이프라인 단계에서 사용되는 상수 버퍼를 가져옵니다. (ID3D10Device.VSGetConstantBuffers)
VSGetSamplers 꼭짓점 셰이더 파이프라인 단계에서 샘플러 상태 배열을 가져옵니다. (ID3D10Device.VSGetSamplers)
VSGetShader 현재 디바이스에 설정된 꼭짓점 셰이더를 가져옵니다. (ID3D10Device.VSGetShader)
VSGetShaderResources 꼭짓점 셰이더 리소스를 가져옵니다. (ID3D10Device.VSGetShaderResources)
VSSetConstantBuffers 꼭짓점 셰이더 파이프라인 단계에서 사용하는 상수 버퍼를 설정합니다.
VSSetSamplers 샘플러 상태 배열을 꼭짓점 셰이더 파이프라인 단계로 설정합니다. (ID3D10Device.VSSetSamplers)
VSSetShader 꼭짓점 셰이더를 디바이스로 설정합니다. (ID3D10Device.VSSetShader)
VSSetShaderResources 셰이더 리소스 배열을 꼭짓점 셰이더 단계에 바인딩합니다.

인터페이스

ID3D10Asynchronous 이 인터페이스는 GPU에서 데이터를 비동기적으로 검색하는 메서드를 캡슐화합니다. (ID3D10Asynchronous)
ID3D10BlendState 이 혼합 상태 인터페이스는 출력 병합 단계의 Direct3D 10.0 디바이스에 대한 혼합 상태에 액세스합니다.
ID3D10BlendState1 이 혼합 상태 인터페이스는 출력 병합 단계의 Direct3D 10.1 디바이스에 대한 혼합 상태에 액세스합니다.
ID3D10Buffer 버퍼 인터페이스는 비정형 메모리인 버퍼 리소스에 액세스합니다. 버퍼는 일반적으로 꼭짓점 또는 인덱스 데이터를 저장합니다. (ID3D10Buffer)
ID3D10Counter 이 인터페이스는 GPU 성능을 측정하기 위한 메서드를 캡슐화합니다. (ID3D10Counter)
ID3D10Debug 디버그 인터페이스는 디버그 설정을 제어하고 파이프라인 상태의 유효성을 검사하며 디버그 계층이 켜져 있는 경우에만 사용할 수 있습니다. (ID3D10Debug)
ID3D10DepthStencilState 깊이 스텐실 상태 인터페이스는 출력-병합 단계에 대한 깊이 스텐실 테스트를 설정하는 깊이 스텐실 상태에 액세스합니다.
ID3D10DepthStencilView 깊이 스텐실 보기 인터페이스는 깊이 스텐실 테스트 중에 텍스처 리소스에 액세스합니다. (ID3D10DepthStencilView)
ID3D10Device 디바이스 인터페이스는 Direct3D 10.0에 대한 가상 어댑터를 나타냅니다. 렌더링을 수행하고 Direct3D 리소스를 만드는 데 사용됩니다.

ID3D10Device1 디바이스 인터페이스는 Direct3D 10.1에 대한 가상 어댑터를 나타냅니다. 렌더링을 수행하고 Direct3D 리소스를 만드는 데 사용됩니다.
ID3D10DeviceChild 디바이스-자식 인터페이스는 디바이스에서 사용하는 데이터에 액세스합니다. (ID3D10DeviceChild)
ID3D10Effect ID3D10Effect 인터페이스는 렌더링 효과를 구현하기 위한 상태 개체, 리소스 및 셰이더 집합을 관리합니다.
ID3D10EffectBlendVariable blend-variable 인터페이스는 혼합 상태에 액세스합니다.
ID3D10EffectConstantBuffer 상수 버퍼 인터페이스는 상수 버퍼 또는 텍스처 버퍼에 액세스합니다.
ID3D10EffectDepthStencilVariable 깊이 스텐실 변수 인터페이스는 깊이 스텐실 상태에 액세스합니다.
ID3D10EffectDepthStencilViewVariable depth-stencil-view-variable 인터페이스는 깊이 스텐실 보기에 액세스합니다.
ID3D10EffectMatrixVariable 행렬 변수 인터페이스는 행렬에 액세스합니다.
ID3D10EffectPass 통과 인터페이스는 기술 내에서 상태 할당을 캡슐화합니다.
ID3D10EffectPool 풀 인터페이스는 효과 간에 변수를 공유하기 위한 공통 메모리 공간(또는 풀)을 나타냅니다.
ID3D10EffectRasterizerVariable 래스터라이저 변수 인터페이스는 래스터라이저 상태에 액세스합니다.

ID3D10EffectRenderTargetViewVariable render-target-view 인터페이스는 렌더링 대상에 액세스합니다.
ID3D10EffectSamplerVariable 샘플러 인터페이스는 샘플러 상태에 액세스합니다.
ID3D10EffectScalarVariable 효과 스칼라 변수 인터페이스는 스칼라 값에 액세스합니다.
ID3D10EffectShaderResourceVariable 셰이더 리소스 인터페이스는 셰이더 리소스에 액세스합니다.
ID3D10EffectShaderVariable 셰이더 변수 인터페이스는 셰이더 변수에 액세스합니다.
ID3D10EffectStringVariable 문자열 변수 인터페이스는 문자열 변수에 액세스합니다.
ID3D10EffectTechnique ID3D10EffectTechnique 인터페이스는 패스의 컬렉션입니다.
ID3D10EffectType ID3D10EffectType 인터페이스는 형식별로 효과 변수에 액세스합니다.
ID3D10EffectVariable ID3D10EffectVariable 인터페이스는 모든 효과 변수의 기본 클래스입니다.
ID3D10EffectVectorVariable 벡터 변수 인터페이스는 4개 구성 요소 벡터에 액세스합니다.
ID3D10GeometryShader 기하 도형 셰이더 인터페이스는 기하 도형 셰이더 단계를 제어하는 실행 프로그램(기하 도형 셰이더)을 관리합니다. (ID3D10GeometryShader)
ID3D10InfoQueue 정보 큐 인터페이스는 디버그 메시지를 저장, 검색 및 필터링합니다. 큐는 메시지 큐, 선택적 스토리지 필터 스택 및 선택적 검색 필터 스택으로 구성됩니다. (ID3D10InfoQueue)

ID3D10InputLayout 입력 레이아웃 인터페이스는 입력 어셈블러 단계의 입력 데이터에 액세스합니다.
ID3D10Multithread 다중 스레드 인터페이스는 다중 스레드 설정에 액세스하며 스레드로부터 안전한 계층이 켜져 있는 경우에만 사용할 수 있습니다.
ID3D10PixelShader 픽셀 셰이더 인터페이스는 픽셀 셰이더 단계를 제어하는 실행 프로그램(픽셀 셰이더)을 관리합니다. (ID3D10PixelShader)
ID3D10Predicate 조건자 인터페이스는 이전 그리기 호출의 결과에 따라 기하 도형을 처리해야 하는지 여부를 결정합니다. (ID3D10Predicate)
ID3D10Query 쿼리 인터페이스는 GPU의 정보를 쿼리합니다. (ID3D10Query)
ID3D10RasterizerState 래스터라이저 상태 인터페이스는 래스터라이저 단계에 대한 래스터라이저 상태에 액세스합니다.
ID3D10RenderTargetView render-target-view 인터페이스는 렌더링 중에 액세스할 수 있는 렌더링 대상 하위 리소스를 식별합니다. (ID3D10RenderTargetView)
ID3D10Resource 리소스 인터페이스는 모든 리소스에 대한 일반적인 작업을 제공합니다. (ID3D10Resource)
ID3D10SamplerState 샘플러 상태 인터페이스는 텍스처에 대한 샘플러 상태에 액세스합니다.
ID3D10ShaderReflection 셰이더 리플렉션 인터페이스는 셰이더 정보에 액세스합니다. (ID3D10ShaderReflection)
ID3D10ShaderReflection1 셰이더 리플렉션 인터페이스는 셰이더 정보에 액세스합니다. (ID3D10ShaderReflection1)

ID3D10ShaderReflectionConstantBuffer 이 셰이더 리플렉션 인터페이스는 상수 버퍼에 대한 액세스를 제공합니다. (ID3D10ShaderReflectionConstantBuffer)
ID3D10ShaderReflectionType 이 셰이더 리플렉션 인터페이스는 변수 형식에 대한 액세스를 제공합니다. (ID3D10ShaderReflectionType)
ID3D10ShaderReflectionVariable 이 셰이더 리플렉션 인터페이스는 변수에 대한 액세스를 제공합니다. (ID3D10ShaderReflectionVariable)
ID3D10ShaderResourceView 셰이더-리소스 뷰 인터페이스는 렌더링 중에 셰이더가 액세스할 수 있는 하위 리소스를 지정합니다. 셰이더 리소스의 예로는 상수 버퍼, 텍스처 버퍼, 텍스처 또는 샘플러가 있습니다. (ID3D10ShaderResourceView)
ID3D10ShaderResourceView1 셰이더-리소스 뷰 인터페이스는 렌더링 중에 셰이더가 액세스할 수 있는 하위 리소스를 지정합니다. 셰이더 리소스의 예로는 상수 버퍼, 텍스처 버퍼, 텍스처 또는 샘플러가 있습니다. (ID3D10ShaderResourceView1)
ID3D10StateBlock 상태 블록 인터페이스는 렌더링 상태를 캡슐화합니다.
ID3D10SwitchToRef 참조로 전환 인터페이스(참조로 전환 계층 참조)를 사용하면 애플리케이션이 하드웨어와 소프트웨어 디바이스 간에 전환할 수 있습니다.
ID3D10Texture1D 1D 텍스처 인터페이스는 구조화된 메모리인 텍셀 데이터에 액세스합니다. (ID3D10Texture1D)
ID3D10Texture2D 2D 텍스처 인터페이스는 구조화된 메모리인 텍셀 데이터를 관리합니다. (ID3D10Texture2D)
ID3D10Texture3D 3D 텍스처 인터페이스는 구조화된 메모리인 텍셀 데이터에 액세스합니다. (ID3D10Texture3D)

ID3D10VertexShader 꼭짓점 셰이더 인터페이스는 꼭짓점 셰이더 단계를 제어하는 실행 프로그램(꼭짓점 셰이더)을 관리합니다. (ID3D10VertexShader)
ID3D10View 뷰 인터페이스는 렌더링 중에 파이프라인이 액세스할 수 있는 리소스 부분을 지정합니다(보기 참조).

구조체

CD3D10_BUFFER_DESC 버퍼 리소스에 대해 설명합니다. D
CD3D10_TEXTURE1D_DESC 1D 텍스처를 설명합니다. D
CD3D10_TEXTURE2D_DESC 2D 텍스처를 설명합니다. D
CD3D10_TEXTURE3D_DESC CD3D10_TEXTURE3D_DESC(d3d10.h) 구조체는 3D 텍스처를 설명합니다.
D3D10_BLEND_DESC 혼합 상태를 설명합니다. (D3D10_BLEND_DESC)
D3D10_BLEND_DESC1 Direct3D 10.1 디바이스의 혼합 상태를 설명합니다.
D3D10_BOX 3D 상자를 정의합니다. (D3D10_BOX)
D3D10_BUFFER_DESC 버퍼 리소스에 대해 설명합니다. (D3D10_BUFFER_DESC)

D3D10_BUFFER_RTV 렌더링 대상 뷰에서 사용할 버퍼 리소스의 요소를 지정합니다.
D3D10_BUFFER_SRV 셰이더-리소스 뷰에서 사용할 버퍼 리소스의 요소를 지정합니다. (D3D10_BUFFER_SRV)
D3D10_COUNTER_DESC 카운터를 설명합니다. (D3D10_COUNTER_DESC)
D3D10_COUNTER_INFO 비디오 카드의 성능 카운터 기능에 대한 정보입니다. (D3D10_COUNTER_INFO)
D3D10_DEPTH_STENCIL_DESC 깊이 스텐실 상태를 설명합니다. (D3D10_DEPTH_STENCIL_DESC)
D3D10_DEPTH_STENCIL_VIEW_DESC 깊이 스텐실 보기를 사용하여 액세스할 수 있는 텍스처의 하위 리소스를 지정합니다.
D3D10_DEPTH_STENCILOP_DESC 스텐실 테스트 결과에 따라 수행할 수 있는 스텐실 작업에 대해 설명합니다.
D3D10_EFFECT_DESC 효과를 설명합니다.
D3D10_EFFECT_SHADER_DESC 효과 셰이더를 설명합니다.
D3D10_EFFECT_TYPE_DESC 효과 변수 형식에 대해 설명합니다.
D3D10_EFFECT_VARIABLE_DESC 효과 변수에 대해 설명합니다.
D3D10_INFO_QUEUE_FILTER 디버그 메시지 필터; 에는 허용하거나 거부할 메시지 형식 목록이 포함되어 있습니다. (D3D10_INFO_QUEUE_FILTER)

D3D10_INFO_QUEUE_FILTER_DESC 특정 유형의 메시지가 필터를 통과하도록 허용하거나 거부합니다. (D3D10_INFO_QUEUE_FILTER_DESC)
D3D10_INPUT_ELEMENT_DESC 입력 어셈블러 단계에 대한 단일 요소에 대한 설명입니다. (D3D10_INPUT_ELEMENT_DESC)
D3D10_MAPPED_TEXTURE2D 2D 텍스처의 하위 리소스 데이터에 대한 액세스를 제공합니다.
D3D10_MAPPED_TEXTURE3D 3D 텍스처의 하위 리소스 데이터에 대한 액세스를 제공합니다.
D3D10_MESSAGE 정보 큐의 디버그 메시지입니다. (D3D10_MESSAGE)
D3D10_PASS_DESC 파이프라인 상태를 포함하는 효과 전달에 대해 설명합니다.
D3D10_PASS_SHADER_DESC 셰이더를 포함하는 효과 변수에 대해 설명합니다.
D3D10_QUERY_DATA_PIPELINE_STATISTICS ID3D10Asynchronous::Begin 및 ID3D10Asynchronous::End 호출 간의 그래픽 파이프라인 활동에 대한 정보를 쿼리합니다.
D3D10_QUERY_DATA_SO_STATISTICS ID3D10Asynchronous::Begin 및 ID3D10Asynchronous::End 사이에 스트림 출력 버퍼로 스트리밍되는 데이터의 양에 대한 정보를 쿼리합니다.
D3D10_QUERY_DATA_TIMESTAMP_DISJOINT 타임스탬프 쿼리의 안정성에 대한 쿼리 정보입니다. (D3D10_QUERY_DATA_TIMESTAMP_DISJOINT)
D3D10_QUERY_DESC 쿼리를 설명합니다. (D3D10_QUERY_DESC)

D3D10_RASTERIZER_DESC 래스터라이저 상태를 설명합니다.
D3D10_RENDER_TARGET_BLEND_DESC1 Direct3D 10.1 디바이스의 렌더링 대상에 대한 혼합 상태를 설명합니다.
D3D10_RENDER_TARGET_VIEW_DESC 렌더링 대상 보기를 사용하여 액세스할 수 있는 리소스의 하위 리소스를 지정합니다.
D3D10_SAMPLER_DESC 샘플러 상태를 설명합니다. (D3D10_SAMPLER_DESC)
D3D10_SHADER_BUFFER_DESC 셰이더 상수 버퍼에 대해 설명합니다. (D3D10_SHADER_BUFFER_DESC)
D3D10_SHADER_DEBUG_FILE_INFO 셰이더에 포함된 파일에 대해 설명합니다.
D3D10_SHADER_DEBUG_INFO D3D10GetShaderDebugInfo에서 반환된 ID3D10Blob 인터페이스의 형식을 설명합니다.
D3D10_SHADER_DEBUG_INPUT_INFO 셰이더 입력에 대해 설명합니다.
D3D10_SHADER_DEBUG_INST_INFO 명령 데이터를 포함합니다.
D3D10_SHADER_DEBUG_OUTPUTREG_INFO 셰이더 출력 레지스터에 대해 설명합니다.
D3D10_SHADER_DEBUG_OUTPUTVAR 셰이더 출력 변수에 대해 설명합니다.
D3D10_SHADER_DEBUG_SCOPE_INFO 변수 이름을 디버그 변수에 매핑하는 범위 데이터를 포함합니다.

D3D10_SHADER_DEBUG_SCOPEVAR_INFO 셰이더 범위 변수에 대해 설명합니다.
D3D10_SHADER_DEBUG_TOKEN_INFO 셰이더 요소의 원본 위치를 제공합니다.
D3D10_SHADER_DEBUG_VAR_INFO 셰이더 원본 변수에 대한 정보를 나타냅니다.
D3D10_SHADER_DESC 셰이더에 대해 설명합니다. (D3D10_SHADER_DESC)
D3D10_SHADER_INPUT_BIND_DESC 셰이더 리소스가 셰이더 입력에 바인딩되는 방법을 설명합니다. (D3D10_SHADER_INPUT_BIND_DESC)
D3D10_SHADER_RESOURCE_VIEW_DESC 셰이더 리소스 뷰에 대해 설명합니다. (D3D10_SHADER_RESOURCE_VIEW_DESC)
D3D10_SHADER_RESOURCE_VIEW_DESC1 셰이더 리소스 뷰에 대해 설명합니다. (D3D10_SHADER_RESOURCE_VIEW_DESC1)
D3D10_SHADER_TYPE_DESC 셰이더 변수 형식에 대해 설명합니다. (D3D10_SHADER_TYPE_DESC)
D3D10_SHADER_VARIABLE_DESC 셰이더 변수에 대해 설명합니다. (D3D10_SHADER_VARIABLE_DESC)
D3D10_SIGNATURE_PARAMETER_DESC 셰이더 서명을 설명합니다. (D3D10_SIGNATURE_PARAMETER_DESC)
D3D10_SO_DECLARATION_ENTRY 출력 슬롯의 꼭짓점 버퍼에 있는 꼭짓점 요소에 대한 설명입니다. (D3D10_SO_DECLARATION_ENTRY)
D3D10_STATE_BLOCK_MASK 디바이스 상태를 나타냅니다.

D3D10_SUBRESOURCE_DATA 하위 리소스를 초기화하기 위한 데이터를 지정합니다. (D3D10_SUBRESOURCE_DATA)
D3D10_TECHNIQUE_DESC 효과 기술을 설명합니다.
D3D10_TEX1D_ARRAY_DSV 깊이 스텐실 보기에서 사용할 1D 텍스처 배열의 하위 리소스를 지정합니다.
D3D10_TEX1D_ARRAY_RTV 렌더링 대상 보기에서 사용할 1D 텍스처 배열의 하위 리소스를 지정합니다.
D3D10_TEX1D_ARRAY_SRV 셰이더 리소스 뷰에서 사용할 1D 텍스처 배열의 하위 리소스를 지정합니다.
D3D10_TEX1D_DSV 깊이 스텐실 보기에 액세스할 수 있는 1D 텍스처의 하위 리소스를 지정합니다. (D3D10_TEX1D_DSV)
D3D10_TEX1D_RTV 렌더링 대상 보기에서 사용할 1D 텍스처의 하위 리소스를 지정합니다. (D3D10_TEX1D_RTV)
D3D10_TEX1D_SRV 셰이더 리소스 뷰에서 사용할 1D 텍스처의 하위 리소스를 지정합니다. (D3D10_TEX1D_SRV)
D3D10_TEX2D_ARRAY_DSV 깊이 스텐실 보기에 액세스할 수 있는 배열 2D 텍스처의 하위 리소스를 지정합니다.
D3D10_TEX2D_ARRAY_RTV 렌더링 대상 보기에서 사용할 2D 텍스처 배열의 하위 리소스를 지정합니다.
D3D10_TEX2D_ARRAY_SRV 셰이더-리소스 뷰에서 사용할 2D 텍스처 배열의 하위 리소스를 지정합니다.
D3D10_TEX2D_DSV 깊이 스텐실 보기에 액세스할 수 있는 2D 텍스처의 하위 리소스를 지정합니다. (D3D10_TEX2D_DSV)

D3D10_TEX2D_RTV 렌더링 대상 보기에서 사용할 2D 텍스처의 하위 리소스를 지정합니다. (D3D10_TEX2D_RTV)
D3D10_TEX2D_SRV 셰이더-리소스 뷰에서 사용할 2D 텍스처의 하위 리소스를 지정합니다. (D3D10_TEX2D_SRV)
D3D10_TEX2DMS_ARRAY_DSV 깊이 스텐실 보기에 대해 다중 샘플링된 2D 텍스처 배열의 하위 리소스를 지정합니다.
D3D10_TEX2DMS_ARRAY_RTV 렌더링 대상 보기에서 사용할 다중 샘플링된 2D 텍스처 배열의 하위 리소스를 지정합니다.
D3D10_TEX2DMS_ARRAY_SRV 셰이더 리소스 뷰에서 사용할 다중 샘플링된 2D 텍스처 배열의 하위 리소스를 지정합니다.
D3D10_TEX2DMS_DSV 깊이 스텐실 보기에 액세스할 수 있는 다중 샘플링된 2D 텍스처의 하위 리소스를 지정합니다. (D3D10_TEX2DMS_DSV)
D3D10_TEX2DMS_RTV 렌더링 대상 보기에서 사용할 다중 샘플링된 2D 텍스처의 하위 리소스를 지정합니다. (D3D10_TEX2DMS_RTV)
D3D10_TEX2DMS_SRV 셰이더-리소스 뷰에서 사용할 다중 샘플링된 2D 텍스처의 하위 리소스를 지정합니다.
D3D10_TEX3D_RTV 렌더링 대상 보기에서 사용할 3D 텍스처의 하위 리소스를 지정합니다.
D3D10_TEX3D_SRV 셰이더 리소스 뷰에서 사용할 3D 텍스처의 하위 리소스를 지정합니다.
D3D10_TEXCUBE_ARRAY_SRV1 셰이더 리소스 뷰에서 사용할 큐브 텍스처 배열의 하위 리소스를 지정합니다.
D3D10_TEXCUBE_SRV 셰이더 리소스 뷰에서 사용할 큐브 텍스처의 하위 리소스를 지정합니다. (D3D10_TEXCUBE_SRV)

D3D10_TEXTURE1D_DESC 1D 텍스처를 설명합니다. (D3D10_TEXTURE1D_DESC)
D3D10_TEXTURE2D_DESC 2D 텍스처를 설명합니다. (D3D10_TEXTURE2D_DESC)
D3D10_TEXTURE3D_DESC 3D 텍스처를 설명합니다. (D3D10_TEXTURE3D_DESC)
D3D10_VIEWPORT 뷰포트의 차원을 정의합니다. (D3D10_VIEWPORT)

Direct3D 11 그래픽

아티클2023. 03. 11.

Direct3D 11 그래픽 기술 개요입니다.

Direct3D 11 그래픽을 개발하려면 다음 헤더가 필요합니다.

- d3d11.h
- d3d11_1.h
- d3d11_2.h
- d3d11_3.h
- d3d11_4.h
- d3d11sdklayers.h
- d3d11shader.h
- d3d11shadertracing.h
- d3dcommon.h
- d3dcsx.h
- windows.graphics.directx.direct3d11.interop.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Direct3D 11 그래픽

열거형

D3D_CBUFFER_TYPE 상수 버퍼 데이터의 의도된 사용을 식별하는 값입니다.
D3D_DRIVER_TYPE 드라이버 유형 옵션입니다.
D3D_FEATURE_LEVEL Direct3D 디바이스를 대상으로 하는 기능 집합에 대해 설명합니다.
D3D_INCLUDE_TYPE 셰이더의 위치를 나타내는 값

D3D_INTERPOLATION_MODE 래스터화 중에 값을 계산하는 방법에 영향을 미치는 보간 모드를 지정합니다.
D3D_MIN_PRECISION 원하는 최소 보간 정밀도를 나타내는 값입니다.
D3D_NAME 시스템 값 의미 체계를 사용하는 셰이더 매개 변수를 식별하는 값입니다.
D3D_PARAMETER_FLAGS 함수 매개 변수에 대한 의미 체계 플래그를 나타냅니다.
D3D_PRIMITIVE 파이프라인이 기하 도형 또는 헐 셰이더 입력 기본 형식을 해석하는 방법을 나타냅니다.
D3D_PRIMITIVE_TOPOLOGY 파이프라인이 입력 어셈블러 단계에 바인딩된 꼭짓점 데이터를 해석하는 방법을 나타내는 값입니다. 이러한 기본 토폴로지 값은 꼭짓점 데이터가 화면에 렌더링되는 방법을 결정합니다.
D3D_REGISTER_COMPONENT_TYPE 레지스터에 저장할 수 있는 데이터 형식을 식별하는 값입니다.
D3D_RESOURCE_RETURN_TYPE 반환 값 형식을 나타냅니다.
D3D_SHADER_CBUFFER_FLAGS 상수 데이터 버퍼의 의도된 사용을 식별하는 값입니다.
D3D_SHADER_INPUT_FLAGS 셰이더 입력 옵션을 식별하는 값입니다.
D3D_SHADER_INPUT_TYPE 셰이더에 바인딩할 수 있고 셰이더에 대한 리소스 설명의 일부로 반영되는 리소스 종류를 식별하는 값입니다.
D3D_SHADER_VARIABLE_CLASS 셰이더 변수의 클래스를 식별하는 값입니다.

D3D_SHADER_VARIABLE_FLAGS 셰이더 변수에 대한 정보를 식별하는 값입니다.
D3D_SHADER_VARIABLE_TYPE 셰이더 변수에 할당할 수 있는 다양한 데이터, 텍스처 및 버퍼 형식을 식별하는 값입니다.
D3D_SRV_DIMENSION 셰이더 리소스로 볼 리소스의 유형을 식별하는 값입니다.
D3D_TESSELLATOR_DOMAIN 테셀레이터 데이터에 대한 도메인 옵션입니다.
D3D_TESSELLATOR_OUTPUT_PRIMITIVE 출력 기본 형식입니다.
D3D_TESSELLATOR_PARTITIONING 분할 옵션.
D3D11_1_CREATE_DEVICE_CONTEXT_STATE_FLAG ID3D11Device1::CreateDeviceContextState 메서드를 사용하여 디바이스 컨텍스트 상태 개체 (ID3DDeviceContextState)를 만드는 데 사용되는 플래그에 대해 설명합니다.
D3D11_ASYNC_GETDATA_FLAG ID3D11DeviceContext::GetData의 동작을 제어하는 선택적 플래그입니다.
D3D11_BIND_FLAG 파이프라인에 리소스를 바인딩하는 방법을 식별합니다. (D3D11_BIND_FLAG)
D3D11_BLEND 픽셀 셰이더 및 렌더링 대상에 대한 값을 변조하는 혼합 요소입니다.
D3D11_BLEND_OP RGB 또는 알파 혼합 작업. (D3D11_BLEND_OP)
D3D11_BUFFER_UAV_FLAG 버퍼 리소스에 대한 순서가 지정되지 않은 액세스 보기 옵션을 식별합니다. (D3D11_BUFFER_UAV_FLAG)

D3D11_BUFFEREX_SRV_FLAG 버퍼 리소스를 보는 방법을 식별합니다. (D3D11_BUFFEREX_SRV_FLAG)
D3D11_CHECK_MULTISAMPLE_QUALITY_LEVELS_FLAG 다중 샘플 품질 수준을 확인하는 방법을 식별합니다.
D3D11_CLEAR_FLAG 지울 깊이 스텐실의 부분을 지정합니다.
D3D11_COLOR_WRITE_ENABLE 혼합 중에 쓸 수 있는 렌더링 대상의 각 픽셀 구성 요소를 식별합니다. (D3D11_COLOR_WRITE_ENABLE)
D3D11_COMPARISON_FUNC 비교 옵션. (D3D11_COMPARISON_FUNC)
D3D11_CONSERVATIVE_RASTERIZATION_MODE 보수적인 래스터화가 켜지거나 꺼져 있는지 여부를 식별합니다. (D3D11_CONSERVATIVE_RASTERIZATION_MODE)
D3D11_CONSERVATIVE_RASTERIZATION_TIER 하드웨어 및 드라이버가 보수적인 래스터화를 지원하는지와 계층 수준에서 지원하는지 지정합니다.
D3D11_CONTEXT_TYPE 쿼리가 발생하는 컨텍스트를 지정합니다.
D3D11_COPY_FLAGS 해당 리소스 내 지역의 복사 또는 업데이트 작업 중에 리소스의 기존 콘텐츠를 처리하는 방법을 지정합니다.
D3D11_COUNTER 성능 카운터에 대한 옵션입니다.
D3D11_COUNTER_TYPE 성능 카운터의 데이터 형식입니다. (D3D11_COUNTER_TYPE)

D3D11_CPU_ACCESS_FLAG 리소스에 허용되는 CPU 액세스 유형을 지정합니다. (D3D11_CPU_ACCESS_FLAG)
D3D11_CREATE_DEVICE_FLAG 디바이스를 만드는 데 사용되는 매개 변수에 대해 설명합니다.
D3D11_CULL_MODE 특정 방향을 향한 삼각형이 그려지지 않음을 나타냅니다. (D3D11_CULL_MODE)
D3D11_DEPTH_WRITE_MASK 깊이 데이터를 작성하기 위한 깊이 스텐실 버퍼의 부분을 식별합니다. (D3D11_DEPTH_WRITE_MASK)
D3D11_DEVICE_CONTEXT_TYPE 디바이스 컨텍스트 옵션.
D3D11_DSV_DIMENSION 깊이 스텐실 보기에서 사용되는 리소스에 액세스하는 방법을 지정합니다. (D3D11_DSV_DIMENSION)
D3D11_DSV_FLAG 깊이 스텐실 보기 옵션입니다.
D3D11_FEATURE Direct3D 11 기능 옵션.
D3D11_FENCE_FLAG 펜스 옵션을 지정합니다. (D3D11_FENCE_FLAG)
D3D11_FILL_MODE 삼각형을 렌더링할 때 사용할 채우기 모드를 결정합니다. (D3D11_FILL_MODE)
D3D11_FILTER 텍스처 샘플링 중 필터링 옵션입니다. (D3D11_FILTER)
D3D11_FILTER_REDUCTION_TYPE 샘플러 필터 감소 유형을 지정합니다.

D3D11_FILTER_TYPE 배율 또는 축소 샘플러 필터의 유형입니다. (D3D11_FILTER_TYPE)
D3D11_FORMAT_SUPPORT 지정된 형식 및 지정된 디바이스에 대해 지원되는 리소스(ID3D11Device::CheckFormatSupport 및 ID3D11Device::CheckFeatureSupport 참조).
D3D11_FORMAT_SUPPORT2 컴퓨팅 셰이더 리소스에 대한 순서가 지정되지 않은 리소스 지원 옵션입니다 (ID3D11Device::CheckFeatureSupport 참조).
D3D11_INPUT_CLASSIFICATION 입력 슬롯에 포함된 데이터 형식입니다. (D3D11_INPUT_CLASSIFICATION)
D3D11_LOGIC_OP 렌더링 대상에 대해 구성할 논리 작업을 지정합니다. (D3D11_LOGIC_OP)
D3D11_MAP CPU에서 읽고 쓰기 위해 액세스할 리소스를 식별합니다. 애플리케이션은 이러한 플래그 중 하나 이상을 결합할 수 있습니다. (D3D11_MAP)
D3D11_MAP_FLAG 애플리케이션이 GPU에서 사용 중인 리소스에서 ID3D11DeviceContext::Map 메서드를 호출할 때 CPU가 응답하는 방법을 지정합니다.
D3D11_MESSAGE_CATEGORY 디버그 메시지의 범주입니다. (D3D11_MESSAGE_CATEGORY)
D3D11_MESSAGE_ID 정보 큐 필터를 설정하기 위한 메시지 디버그(D3D11_INFO_QUEUE_FILTER 참조) 이러한 메시지를 사용하여 스토리지 및 검색 필터를 통과하도록 메시지 범주를 허용하거나 거부합니다.
D3D11_MESSAGE_SEVERITY 정보 큐에 대한 메시지 심각도 수준을 디버그합니다. (D3D11_MESSAGE_SEVERITY)
D3D11_QUERY 쿼리 형식입니다. (D3D11_QUERY)

D3D11_QUERY_MISC_FLAG 기타 쿼리 동작을 설명하는 플래그입니다. (D3D11_QUERY_MISC_FLAG)
D3D11_RAISE_FLAG 연속 불가능한 예외로 오류를 발생시키는 옵션입니다. (D3D11_RAISE_FLAG)
D3D11_RESOURCE_DIMENSION 사용 중인 리소스의 유형을 식별합니다. (D3D11_RESOURCE_DIMENSION)
D3D11_RESOURCE_MISC_FLAG 리소스에 대한 옵션을 식별합니다.
D3D11_RLDO_FLAGS 디바이스 개체의 수명에 대해 보고할 정보의 양에 대한 옵션입니다.
D3D11_RTV_DIMENSION 이러한 플래그는 렌더링 대상으로 볼 리소스 유형을 식별합니다.
D3D11_SHADER_CACHE_SUPPORT_FLAGS 현재 그래픽 드라이버의 셰이더 캐싱 지원 수준을 설명합니다. (D3D11_SHADER_CACHE_SUPPORT_FLAGS)
D3D11_SHADER_MIN_PRECISION_SUPPORT 셰이더 단계에서 최소 정밀도 수준을 지정하는 값입니다.
D3D11_SHADER_TRACKING_OPTIONS 셰이더 디버그 추적을 수행하는 방법을 지정하는 옵션입니다.
D3D11_SHADER_TRACKING_RESOURCE_TYPE 추적할 리소스 종류를 나타냅니다.
D3D11_SHADER_TYPE 추적을 위한 셰이더 유형을 식별합니다.
D3D11_SHADER_VERSION_TYPE 셰이더 유형을 나타냅니다.

D3D11_SHARED_RESOURCE_TIER 현재 그래픽 드라이버에서 공유 리소스에 대한 지원 수준을 지정하는 상수를 정의합니다.
D3D11_STANDARD_MULTISAMPLE_QUALITY_LEVELS 다중 샘플 패턴 형식을 지정합니다.
D3D11_STENCIL_OP 깊이 스텐실 테스트 중에 수행할 수 있는 스텐실 작업입니다. (D3D11_STENCIL_OP)
D3D11_TEXTURE_ADDRESS_MODE 텍스처 경계 밖에 있는 텍스처 좌표를 확인하는 기술을 식별합니다. (D3D11_TEXTURE_ADDRESS_MODE)
D3D11_TEXTURE_LAYOUT 텍스처 레이아웃 옵션을 지정합니다. (D3D11_TEXTURE_LAYOUT)
D3D11_TEXTURECUBE_FACE 큐브 텍스처의 다른 얼굴입니다. (D3D11_TEXTURECUBE_FACE)
D3D11_TILE_COPY_FLAG 타일을 복사하는 방법을 식별합니다.
D3D11_TILE_MAPPING_FLAG 타일 매핑 작업을 수행하는 방법을 식별합니다.
D3D11_TILE_RANGE_FLAG ID3D11DeviceContext2::UpdateTiles와 함께 사용할 타일 매핑 범위를 지정합니다.
D3D11_TILED_RESOURCES_TIER 타일형 리소스가 지원되는 계층 수준을 나타냅니다.
D3D11_TRACE_GS_INPUT_PRIMITIVE 기하 도형 셰이더 입력 기본 형식을 식별합니다.
D3D11_TRACE_REGISTER_TYPE 추적 레지스터 유형을 식별합니다.

D3D11_UAV_DIMENSION 순서가 지정되지 않은 액세스 보기 옵션입니다.
D3D11_USAGE 렌더링하는 동안 예상되는 리소스 사용을 식별합니다. 사용량은 CPU 및/또는 GPU(그래픽 처리 장치)에서 리소스에 액세스할 수 있는지 여부를 직접 반영합니다.
D3D11_VIDEO_DECODER_HISTOGRAM_COMPONENT 구성 요소별 히스토그램 정보의 배열에 대한 인덱스를 지정합니다. (D3D11_VIDEO_DECODER_HISTOGRAM_COMPONENT)
D3D11_VIDEO_DECODER_HISTOGRAM_COMPONENT_FLAGS 비디오 디코딩 히스토그램과 함께 사용되는 구성 요소의 하위 집합을 나타내는 플래그입니다. (D3D11_VIDEO_DECODER_HISTOGRAM_COMPONENT_FLAGS)
D3DX11_FFT_CREATE_FLAG FFT 생성 플래그입니다.
D3DX11_FFT_DATA_TYPE FFT 데이터 형식.
D3DX11_FFT_DIM_MASK FFT 데이터의 차원 수입니다.
D3DX11_SCAN_DATA_TYPE 스캔 데이터를 입력합니다.
D3DX11_SCAN_DIRECTION 검색을 수행할 방향입니다.
D3DX11_SCAN_OPCODE opcode를 검사합니다.

Functions

~CD3D11_BLEND_DESC CD3D11_BLEND_DESC 구조체의 인스턴스를 삭제합니다.
~CD3D11_BOX CD3D11_BOX 구조체의 인스턴스를 삭제합니다.
~CD3D11_BUFFER_DESC CD3D11_BUFFER_DESC 구조체의 인스턴스를 삭제합니다.
~CD3D11_COUNTER_DESC CD3D11_COUNTER_DESC 구조체의 인스턴스를 삭제합니다.
~CD3D11_DEPTH_STENCIL_DESC CD3D11_DEPTH_STENCIL_DESC 구조체의 인스턴스를 삭제합니다.
~CD3D11_DEPTH_STENCIL_VIEW_DESC CD3D11_DEPTH_STENCIL_VIEW_DESC 구조체의 인스턴스를 삭제합니다.
~CD3D11_QUERY_DESC CD3D11_QUERY_DESC 구조체의 인스턴스를 삭제합니다.
~CD3D11_RASTERIZER_DESC CD3D11_RASTERIZER_DESC 구조체의 인스턴스를 삭제합니다.
~CD3D11_RECT CD3D11_RECT 구조체의 인스턴스를 삭제합니다.
~CD3D11_RENDER_TARGET_VIEW_DESC CD3D11_RENDER_TARGET_VIEW_DESC 구조체의 인스턴스를 삭제합니다.
~CD3D11_SAMPLER_DESC CD3D11_SAMPLER_DESC 구조체의 인스턴스를 삭제합니다.
~CD3D11_SHADER_RESOURCE_VIEW_DESC CD3D11_SHADER_RESOURCE_VIEW_DESC 구조체의 인스턴스를 삭제합니다.

~CD3D11_TEXTURE1D_DESC CD3D11_TEXTURE1D_DESC 구조체의 인스턴스를 삭제합니다.
~CD3D11_TEXTURE2D_DESC CD3D11_TEXTURE2D_DESC 구조체의 인스턴스를 삭제합니다.
~CD3D11_TEXTURE3D_DESC CD3D11_TEXTURE3D_DESC 구조체의 인스턴스를 삭제합니다.
~CD3D11_UNORDERED_ACCESS_VIEW_DESC CD3D11_UNORDERED_ACCESS_VIEW_DESC 구조체의 인스턴스를 삭제합니다.
~CD3D11_VIEWPORT CD3D11_VIEWPORT 구조체의 인스턴스를 삭제합니다.
AddApplicationMessage 사용자 정의 메시지를 메시지 큐에 추가하고 해당 메시지를 전송하여 출력을 디버그합니다. (ID3D11InfoQueue.AddApplicationMessage)
AddClipPlaneFromCBuffer 10Level9 셰이더에 대한 cbuffer 항목에서 가져온 평면 계수가 있는 클립 평면을 추가합니다.
AddMessage 메시지 큐에 디버그 메시지를 추가하고 해당 메시지를 전송하여 출력을 디버그합니다.
AddRetrievalFilterEntries 검색 필터 스택의 맨 위에 스토리지 필터를 추가합니다. (ID3D11InfoQueue.AddRetrievalFilterEntries)
AddStorageFilterEntries 스토리지 필터 스택의 맨 위에 스토리지 필터를 추가합니다. (ID3D11InfoQueue.AddStorageFilterEntries)
AttachBuffersAndPrecompute FFT 컨텍스트에 버퍼를 연결하고 필요한 사전 계산을 수행합니다.

시작 일련의 명령의 시작을 표시합니다.
BeginEvent 이벤트 코드 섹션의 시작을 표시합니다.
BeginEventInt 애플리케이션에서 다양한 그래픽 명령의 시작 부분에 주석을 달 수 있습니다.
BindConstantBuffer 원본 슬롯에서 대상 슬롯으로 상수 버퍼를 다시 바인딩합니다.
BindConstantBufferByName 이름별로 상수 버퍼를 대상 슬롯에 다시 바인딩합니다.
BindResource 원본 슬롯에서 대상 슬롯으로 텍스처 또는 버퍼를 다시 바인딩합니다.
BindResourceAsUnorderedAccessView 리소스를 원본 슬롯에서 대상 슬롯으로 정렬되지 않은 액세스 보기(UAV)로 다시 바인딩합니다.
BindResourceAsUnorderedAccessViewByName 이름으로 리소스를 UAV(정렬되지 않은 액세스 보기)로 대상 슬롯에 다시 바인딩합니다.
BindResourceByName 이름으로 텍스처 또는 버퍼를 대상 슬롯에 다시 바인딩합니다.
BindSampler 샘플러를 원본 슬롯에서 대상 슬롯으로 다시 바인딩합니다.
BindSamplerByName 이름으로 샘플러를 대상 슬롯에 다시 바인딩합니다.
BindUnorderedAccessView 원본 슬롯에서 대상 슬롯으로 UAV(정렬되지 않은 액세스 뷰)를 다시 바인딩합니다.

BindUnorderedAccessViewByName 이름으로 UAV(정렬되지 않은 액세스 뷰)를 대상 슬롯에 다시 바인딩합니다.
CallFunction function-linking-graph에 사용할 호출 함수 연결 노드를 만듭니다.
CD3D11_BLEND_DESC 초기화되지 않은 **CD3D11_BLEND_DESC** 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_BLEND_DESC 기본 혼합 상태 값으로 초기화된 CD3D11_BLEND_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_BLEND_DESC CD3D11_BLEND_DESC 구조체로 초기화된 **CD3D11_BLEND_DESC구조체** 의 새 인스턴스를 인스턴스화합니다.
CD3D11_BOX 초기화되지 않은 CD3D11_BOX 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_BOX CD3D11_BOX 구조체로 초기화된 **CD3D11_BOX구조체** 의 새 인스턴스를 인스턴스화합니다.
CD3D11_BOX 상자의 차원으로 초기화된 CD3D11_BOX 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_BUFFER_DESC 초기화되지 않은 CD3D11_BUFFER_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_BUFFER_DESC D3D11_BUFFER_DESC 구조체로 초기화된 CD3D11_BUFFER_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_BUFFER_DESC D3D11_BUFFER_DESC 값으로 초기화된 CD3D11_BUFFER_DESC 구조체의 새 인스턴스를 인스턴스화합니다.

CD3D11_COUNTER_DESC 초기화되지 않은 CD3D11_COUNTER_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_COUNTER_DESC D3D11_COUNTER_DESC 구조체로 초기화된 CD3D11_COUNTER_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_COUNTER_DESC 카운터에 대한 정보를 사용하여 초기화된 CD3D11_COUNTER_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_DEPTH_STENCIL_DESC 초기화되지 않은 CD3D11_DEPTH_STENCIL_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_DEPTH_STENCIL_DESC 기본 깊이 스텐실 상태 값으로 초기화된 CD3D11_DEPTH_STENCIL_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_DEPTH_STENCIL_DESC D3D11_DEPTH_STENCIL_DESC 구조체로 초기화된 **CD3D11_DEPTH_STENCIL_DESC 구조체** 의 새 인스턴스를 인스턴스화합니다.
CD3D11_DEPTH_STENCIL_VIEW_DESC 초기화되지 않은 CD3D11_DEPTH_STENCIL_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_DEPTH_STENCIL_VIEW_DESC D3D11_DEPTH_STENCIL_VIEW_DESC 구조체로 초기화된 CD3D11_DEPTH_STENCIL_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_DEPTH_STENCIL_VIEW_DESC D3D11_DEPTH_STENCIL_VIEW_DESC 값으로 초기화된 CD3D11_DEPTH_STENCIL_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_DEPTH_STENCIL_VIEW_DESC D3D11_TEX1D_DSV 또는 D3D11_TEX1D_ARRAY_DSV 값으로 초기화된 CD3D11_DEPTH_STENCIL_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.

CD3D11_DEPTH_STENCIL_VIEW_DESC 2D 텍스처 값으로 초기화된 CD3D11_DEPTH_STENCIL_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_QUERY_DESC 초기화되지 않은 **CD3D11_QUERY_DESC** 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_QUERY_DESC D3D11_QUERY_DESC 구조체로 초기화된 CD3D11_QUERY_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_QUERY_DESC 쿼리에 대한 정보를 사용하여 초기화된 CD3D11_QUERY_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_RASTERIZER_DESC 초기화되지 않은 CD3D11_RASTERIZER_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_RASTERIZER_DESC 기본 래스터라이저 상태 값으로 초기화된 CD3D11_RASTERIZER_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_RASTERIZER_DESC D3D11_RASTERIZER_DESC 구조체로 초기화된 CD3D11_RASTERIZER_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_RASTERIZER_DESC D3D11_RASTERIZER_DESC 값으로 초기화된 CD3D11_RASTERIZER_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_RECT 초기화되지 않은 CD3D11_RECT 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_RECT D3D11_RECT 구조체로 초기화된 CD3D11_RECT 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_RECT 사각형의 차원으로 초기화된 CD3D11_RECT 구조체의 새 인스턴스를 인스턴스화합니다.

CD3D11_RENDER_TARGET_VIEW_DESC 초기화되지 않은 CD3D11_RENDER_TARGET_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_RENDER_TARGET_VIEW_DESC D3D11_RENDER_TARGET_VIEW_DESC 구조체로 초기화된 CD3D11_RENDER_TARGET_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_RENDER_TARGET_VIEW_DESC D3D11_RENDER_TARGET_VIEW_DESC 값으로 초기화된 CD3D11_RENDER_TARGET_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_RENDER_TARGET_VIEW_DESC D3D11_RTV_DIMENSION 값으로 초기화된 CD3D11_RENDER_TARGET_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_RENDER_TARGET_VIEW_DESC D3D11_TEX1D_RTV 또는 D3D11_TEX1D_ARRAY_RTV 값으로 초기화된 CD3D11_RENDER_TARGET_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_RENDER_TARGET_VIEW_DESC 2D 텍스처 값으로 초기화된 CD3D11_RENDER_TARGET_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_RENDER_TARGET_VIEW_DESC 3D 텍스처 값으로 초기화된 CD3D11_RENDER_TARGET_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_SAMPLER_DESC 초기화되지 않은 CD3D11_SAMPLER_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_SAMPLER_DESC 기본 샘플러 상태 값으로 초기화된 CD3D11_SAMPLER_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_SAMPLER_DESC D3D11_SAMPLER_DESC 구조체로 초기화된 CD3D11_SAMPLER_DESC 구조체의 새 인스턴스를 인스턴스화합니다. (오버로드 1/2)

CD3D11_SAMPLER_DESC D3D11_SAMPLER_DESC 구조체로 초기화된 CD3D11_SAMPLER_DESC 구조체의 새 인스턴스를 인스턴스화합니다. (오버로드 2/2)
CD3D11_SHADER_RESOURCE_VIEW_DESC 초기화되지 않은 CD3D11_SHADER_RESOURCE_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_SHADER_RESOURCE_VIEW_DESC D3D11_SHADER_RESOURCE_VIEW_DESC 구조체로 초기화된 CD3D11_SHADER_RESOURCE_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_SHADER_RESOURCE_VIEW_DESC D3D11_SHADER_RESOURCE_VIEW_DESC 값으로 초기화된 CD3D11_SHADER_RESOURCE_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_SHADER_RESOURCE_VIEW_DESC D3D11_BUFFEREX_SRV 값으로 초기화된 CD3D11_SHADER_RESOURCE_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_SHADER_RESOURCE_VIEW_DESC D3D11_TEX1D_SRV 또는 D3D11_TEX1D_ARRAY_SRV 값으로 초기화된 CD3D11_SHADER_RESOURCE_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_SHADER_RESOURCE_VIEW_DESC 2D 텍스처 값으로 초기화된 CD3D11_SHADER_RESOURCE_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_SHADER_RESOURCE_VIEW_DESC 3D 텍스처 값으로 초기화된 CD3D11_SHADER_RESOURCE_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_TEXTURE1D_DESC 초기화되지 않은 CD3D11_TEXTURE1D_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_TEXTURE1D_DESC D3D11_TEXTURE1D_DESC 구조체로 초기화된 CD3D11_TEXTURE1D_DESC 구조체의 새 인스턴스를 인스턴스화합니다.

CD3D11_TEXTURE1D_DESC D3D11_TEXTURE1D_DESC 값으로 초기화된 CD3D11_TEXTURE1D_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_TEXTURE2D_DESC 초기화되지 않은 CD3D11_TEXTURE2D_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_TEXTURE2D_DESC D3D11_TEXTURE2D_DESC 구조체로 초기화된 CD3D11_TEXTURE2D_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_TEXTURE2D_DESC D3D11_TEXTURE2D_DESC 값으로 초기화된 CD3D11_TEXTURE2D_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_TEXTURE3D_DESC 초기화되지 않은 CD3D11_TEXTURE3D_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_TEXTURE3D_DESC D3D11_TEXTURE3D_DESC 구조체로 초기화된 CD3D11_TEXTURE3D_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_TEXTURE3D_DESC D3D11_TEXTURE3D_DESC 값으로 초기화된 CD3D11_TEXTURE3D_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_UNORDERED_ACCESS_VIEW_DESC 초기화되지 않은 CD3D11_UNORDERED_ACCESS_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_UNORDERED_ACCESS_VIEW_DESC D3D11_UNORDERED_ACCESS_VIEW_DESC 구조체로 초기화된 CD3D11_UNORDERED_ACCESS_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_UNORDERED_ACCESS_VIEW_DESC D3D11_UNORDERED_ACCESS_VIEW_DESC 값으로 초기화된 CD3D11_UNORDERED_ACCESS_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.

CD3D11_UNORDERED_ACCESS_VIEW_DESC D3D11_BUFFER_UAV 값으로 초기화된 CD3D11_UNORDERED_ACCESS_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_UNORDERED_ACCESS_VIEW_DESC D3D11_TEX1D_UAV 또는 D3D11_TEX1D_ARRAY_UAV 값으로 초기화된 CD3D11_UNORDERED_ACCESS_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_UNORDERED_ACCESS_VIEW_DESC 2D 텍스처 값으로 초기화된 CD3D11_UNORDERED_ACCESS_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_UNORDERED_ACCESS_VIEW_DESC 3D 텍스처 값으로 초기화된 CD3D11_UNORDERED_ACCESS_VIEW_DESC 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_VIEWPORT 초기화되지 않은 CD3D11_VIEWPORT 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_VIEWPORT D3D11_VIEWPORT 구조체로 초기화된 CD3D11_VIEWPORT 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_VIEWPORT D3D11_VIEWPORT 값으로 초기화된 CD3D11_VIEWPORT 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_VIEWPORT D3D11_BUFFER_RTV 값으로 초기화된 CD3D11_VIEWPORT 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_VIEWPORT D3D11_TEX1D_RTV 또는 D3D11_TEX1D_ARRAY_RTV 값으로 초기화된 CD3D11_VIEWPORT 구조체의 새 인스턴스를 인스턴스화합니다.
CD3D11_VIEWPORT 2D 텍스처 값으로 초기화된 CD3D11_VIEWPORT 구조체의 새 인스턴스를 인스턴스화합니다.

CD3D11_VIEWPORT 3D 텍스처 값으로 초기화된 CD3D11_VIEWPORT 구조체의 새 인스턴스를 인스턴스화합니다.
CheckCounter 형식, 이름, 측정 단위 및 기존 카운터에 대한 설명을 가져옵니다. (ID3D11Device.CheckCounter)
CheckCounterInfo 카운터의 정보를 가져옵니다. (ID3D11Device.CheckCounterInfo)
CheckFeatureSupport 현재 그래픽 드라이버에서 지원하는 기능에 대한 정보를 가져옵니다. (ID3D11Device.CheckFeatureSupport)
CheckFeatureSupport 현재 비디오 드라이버에서 지원하는 기능에 대한 정보를 가져옵니다. (ID3D11VideoDevice2::CheckFeatureSupport)
CheckFormatSupport 설치된 비디오 디바이스에서 지정된 형식의 지원을 받습니다. (ID3D11Device.CheckFormatSupport)
CheckMultisampleQualityLevels 다중 샘플링 중에 사용할 수 있는 품질 수준 수를 가져옵니다. (ID3D11Device.CheckMultisampleQualityLevels)
CheckMultisampleQualityLevels1 다중 샘플링 중에 사용할 수 있는 품질 수준 수를 가져옵니다. (ID3D11Device2.CheckMultisampleQualityLevels1)
ClearDepthStencilView 깊이 스텐실 리소스를 지웁니다. (ID3D11DeviceContext.ClearDepthStencilView)
ClearRenderTargetView 렌더링 대상의 모든 요소를 하나의 값으로 설정합니다. (ID3D11DeviceContext.ClearRenderTargetView)
ClearRetrievalFilter 검색 필터 스택의 맨 위에서 검색 필터를 제거합니다. (ID3D11InfoQueue.ClearRetrievalFilter)

ClearState 모든 기본 설정을 복원합니다.
ClearStorageFilter 스토리지 필터 스택의 맨 위에서 스토리지 필터를 제거합니다. (ID3D11InfoQueue.ClearStorageFilter)
ClearStoredMessages 메시지 큐에서 모든 메시지를 지웁니다. (ID3D11InfoQueue.ClearStoredMessages)
ClearUnorderedAccessViewFloat 부동 소수점 값을 사용하여 순서가 지정되지 않은 액세스 리소스를 지웁니다.
ClearUnorderedAccessViewUint 비트 정밀 값을 사용하여 정렬되지 않은 액세스 리소스를 지웁니다.
ClearView 리소스 뷰의 모든 요소를 하나의 값으로 설정합니다.
닫기 셰이더를 닫기 위한 사용자 구현 메서드
CopyResource GPU를 사용하여 원본 리소스의 전체 콘텐츠를 대상 리소스에 복사합니다. (ID3D11DeviceContext.CopyResource)
CopyStructureCount 가변 길이 데이터를 보유하는 버퍼에서 데이터를 복사합니다.
CopySubresourceRegion 원본 리소스에서 대상 리소스로 지역을 복사합니다. (ID3D11DeviceContext.CopySubresourceRegion)
CopySubresourceRegion1 원본 리소스에서 대상 리소스로 지역을 복사합니다.

CopyTileMappings 매핑을 원본 타일식 리소스에서 대상 타일식 리소스로 복사합니다.
CopyTiles 버퍼에서 타일 리소스로 또는 그 반대로 타일을 복사합니다. (ID3D11DeviceContext2.CopyTiles)
CreateBlendState 출력 병합기 단계에 대한 혼합 상태를 캡슐화하는 혼합 상태 개체를 만듭니다. (ID3D11Device.CreateBlendState)
CreateBlendState1 출력 병합기 단계에 대한 혼합 상태를 캡슐화하고 논리 작업의 구성을 허용하는 혼합 상태 개체를 만듭니다.
CreateBuffer 버퍼(꼭짓점 버퍼, 인덱스 버퍼 또는 셰이더 상수 버퍼)를 만듭니다.
CreateClassInstance HLSL 클래스 인스턴스를 나타내는 클래스 인스턴스 개체를 초기화합니다.
CreateClassLinkage 클래스 연결 라이브러리를 만들어 동적 셰이더 연결을 사용하도록 설정합니다.
CreateComputeShader 컴퓨팅 셰이더를 만듭니다.
CreateCounter GPU 성능을 측정하기 위한 카운터 개체를 만듭니다. (ID3D11Device.CreateCounter)
CreateDeferredContext 명령 목록을 기록할 수 있는 지연된 컨텍스트를 만듭니다. (ID3D11Device.CreateDeferredContext)
CreateDeferredContext1 명령 목록을 기록할 수 있는 지연된 컨텍스트를 만듭니다. (ID3D11Device1.CreateDeferredContext1)

CreateDeferredContext2 명령 목록을 기록할 수 있는 지연된 컨텍스트를 만듭니다. (ID3D11Device2.CreateDeferredContext2)
CreateDeferredContext3 명령 목록을 기록할 수 있는 지연된 컨텍스트를 만듭니다. (ID3D11Device3.CreateDeferredContext3)
CreateDepthStencilState 출력 병합 단계에 대한 깊이 스텐실 테스트 정보를 캡슐화하는 깊이 스텐실 상태 개체를 만듭니다. (ID3D11Device.CreateDepthStencilState)
CreateDepthStencilView 리소스 데이터에 액세스하기 위한 깊이 스텐실 보기를 만듭니다. (ID3D11Device.CreateDepthStencilView)
CreateDeviceContextState 모든 Microsoft Direct3D 상태와 일부 Direct3D 동작을 보유하는 컨텍스트 상태 개체를 만듭니다.
CreateDomainShader 도메인 셰이더를 만듭니다.
CreateFence 펜스 개체를 만듭니다. (ID3D11Device5.CreateFence)
CreateGeometryShader 기하 도형 셰이더를 만듭니다. (ID3D11Device.CreateGeometryShader)
CreateGeometryShaderWithStreamOutput 스트리밍 출력 버퍼에 쓸 수 있는 기하 도형 셰이더를 만듭니다. (ID3D11Device.CreateGeometryShaderWithStreamOutput)
CreateHullShader 헐 셰이더를 만듭니다.
CreateInputLayout 입력-어셈블러 단계에 대한 입력 버퍼 데이터를 설명하는 입력 레이아웃 개체를 만듭니다. (ID3D11Device.CreateInputLayout)

CreateInstance 리소스 다시 바인딩에 사용되는 셰이더 모듈의 인스턴스를 초기화합니다.
CreateModuleInstance function-linking-graph 개체에서 셰이더 모듈을 초기화합니다.
CreatePixelShader 픽셀 셰이더를 만듭니다. (ID3D11Device.CreatePixelShader)
CreatePredicate 조건자를 만듭니다. (ID3D11Device.CreatePredicate)
CreateQuery 이 인터페이스는 GPU에서 정보를 쿼리하는 메서드를 캡슐화합니다. (ID3D11Device.CreateQuery)
CreateQuery1 GPU(그래픽 처리 장치)에서 정보를 쿼리하기 위한 쿼리 개체를 만듭니다.
CreateRasterizerState 래스터라이저 단계에 동작 방법을 알려주는 래스터라이저 상태 개체를 만듭니다. (ID3D11Device.CreateRasterizerState)
CreateRasterizerState1 UAV 렌더링 또는 래스터화 중 래스터라이저 스테이지에서 동작하는 방법을 알리고 샘플 수를 강제로 적용하는 래스터라이저 상태 개체를 만듭니다. (ID3D11Device1.CreateRasterizerState1)
CreateRasterizerState2 UAV 렌더링 또는 래스터화 중 래스터라이저 스테이지에서 동작하는 방법을 알리고 샘플 수를 강제로 적용하는 래스터라이저 상태 개체를 만듭니다. (ID3D11Device3.CreateRasterizerState2)
CreateRenderTargetView 리소스 데이터에 액세스하기 위한 렌더링 대상 뷰를 만듭니다. (ID3D11Device.CreateRenderTargetView)
CreateRenderTargetView1 리소스 데이터에 액세스하기 위한 렌더링 대상 뷰를 만듭니다. (ID3D11Device3.CreateRenderTargetView1)

CreateSamplerState 텍스처에 대한 샘플링 정보를 캡슐화하는 샘플러 상태 개체를 만듭니다. (ID3D11Device.CreateSamplerState)
CreateShaderResourceView 리소스의 데이터에 액세스하기 위한 셰이더 리소스 뷰를 만듭니다. (ID3D11Device.CreateShaderResourceView)
CreateShaderResourceView1 리소스의 데이터에 액세스하기 위한 셰이더-리소스 뷰를 만듭니다. (ID3D11Device3.CreateShaderResourceView1)
CreateShaderTrace 셰이더 추적 정보 개체에 대한 셰이더 추적 인터페이스를 만듭니다.
CreateSharedHandle 펜스 개체에 대한 공유 핸들을 만듭니다.
CreateTexture1D 1D 텍스처의 배열을 만듭니다.
CreateTexture2D 2D 텍스처 배열을 만듭니다.
CreateTexture2D1 2D 텍스처를 만듭니다.
CreateTexture3D 단일 3D 텍스처를 만듭니다.
CreateTexture3D1 3D 텍스처를 만듭니다.
CreateUnorderedAccessView 순서가 지정되지 않은 액세스 리소스에 액세스하기 위한 뷰를 만듭니다. (ID3D11Device.CreateUnorderedAccessView)

CreateUnorderedAccessView1 순서가 지정되지 않은 액세스 리소스에 액세스하기 위한 뷰를 만듭니다. (ID3D11Device3.CreateUnorderedAccessView1)
CreateVertexShader 컴파일된 셰이더에서 꼭짓점 셰이더 개체를 만듭니다. (ID3D11Device.CreateVertexShader)
CSGetConstantBuffers 컴퓨팅 셰이더 단계에서 사용하는 상수 버퍼를 가져옵니다.
CSGetConstantBuffers1 컴퓨팅 셰이더 단계에서 사용하는 상수 버퍼를 가져옵니다.
CSGetSamplers 컴퓨팅 셰이더 단계에서 샘플러 상태 인터페이스의 배열을 가져옵니다.
CSGetShader 디바이스에 현재 설정된 컴퓨팅 셰이더를 가져옵니다.
CSGetShaderResources 컴퓨팅 셰이더 리소스를 가져옵니다.
CSGetUnorderedAccessViews 순서가 지정되지 않은 리소스에 대한 뷰 배열을 가져옵니다.
CSSetConstantBuffers 컴퓨팅 셰이더 단계에서 사용하는 상수 버퍼를 설정합니다.
CSSetConstantBuffers1 컴퓨팅 셰이더 단계에서 사용하는 상수 버퍼를 설정합니다.
CSSetSamplers 샘플러 상태 배열을 컴퓨팅 셰이더 단계로 설정합니다.
CSSetShader 컴퓨팅 셰이더를 디바이스로 설정합니다.

CSSetShaderResources 셰이더 리소스 배열을 컴퓨팅 셰이더 단계에 바인딩합니다.
CSSetUnorderedAccessViews 순서가 지정되지 않은 리소스에 대한 뷰 배열을 설정합니다.
D3D11CalcSubresource 텍스처에 대한 하위 리소스 인덱스를 계산합니다.
D3D11CreateDevice 디스플레이 어댑터를 나타내는 디바이스를 만듭니다. (D3D11CreateDevice)
D3D11CreateDeviceAndSwapChain 디스플레이 어댑터와 렌더링에 사용되는 스왑 체인을 나타내는 디바이스를 만듭니다.
D3DDisassemble11Trace 셰이더 추적 단계에서 지정한 컴파일된 Microsoft HLSL(High Level Shader Language) 코드의 섹션을 디스어셈블합니다.
D3DX11CreateFFT ID3DX11FFT COM 인터페이스 개체를 만듭니다. (D3DX11CreateFFT)
D3DX11CreateFFT1DComplex ID3DX11FFT COM 인터페이스 개체를 만듭니다. (D3DX11CreateFFT1DComplex)
D3DX11CreateFFT1DReal ID3DX11FFT COM 인터페이스 개체를 만듭니다. (D3DX11CreateFFT1DReal)
D3DX11CreateFFT2DComplex ID3DX11FFT COM 인터페이스 개체를 만듭니다. (D3DX11CreateFFT2DComplex)
D3DX11CreateFFT2DReal ID3DX11FFT COM 인터페이스 개체를 만듭니다. (D3DX11CreateFFT2DReal)
D3DX11CreateFFT3DComplex ID3DX11FFT COM 인터페이스 개체를 만듭니다. (D3DX11CreateFFT3DComplex)

D3DX11CreateFFT3DReal ID3DX11FFT COM 인터페이스 개체를 만듭니다. (D3DX11CreateFFT3DReal)
D3DX11CreateScan 검사 컨텍스트를 만듭니다.
D3DX11CreateSegmentedScan 분할된 검사 컨텍스트를 만듭니다.
DecoderBeginFrame1 비디오 프레임을 디코딩하는 디코딩 작업을 시작합니다. (ID3D11VideoContext3::DecoderBeginFrame1)
DiscardResource 디바이스 컨텍스트에서 리소스를 삭제합니다.
DiscardView 디바이스 컨텍스트에서 리소스 보기를 삭제합니다.
DiscardView1 디바이스 컨텍스트에서 리소스 뷰의 지정된 요소를 삭제합니다.
Dispatch 스레드 그룹에서 명령 목록을 실행합니다.
DispatchIndirect 하나 이상의 스레드 그룹에 대해 명령 목록을 실행합니다.
그릴 인덱싱되지 않은 인스턴스가 아닌 기본 형식을 그립니다. (ID3D11DeviceContext.Draw)
DrawAuto 알 수 없는 크기의 기하 도형을 그립니다.
DrawIndexed 인덱싱된 인스턴스가 아닌 기본 형식을 그립니다. (ID3D11DeviceContext.DrawIndexed)

DrawIndexedInstanced 인덱싱된 인스턴스화된 기본 형식을 그립니다. (ID3D11DeviceContext.DrawIndexedInstanced)
DrawIndexedInstancedIndirect 인덱싱된 인스턴스화된 GPU 생성 기본 형식을 그립니다.
DrawInstanced 인덱싱되지 않은 인스턴스화된 기본 형식을 그립니다. (ID3D11DeviceContext.DrawInstanced)
DrawInstancedIndirect 인스턴스화된 GPU 생성 기본 형식을 그립니다.
DSGetConstantBuffers 도메인 셰이더 단계에서 사용하는 상수 버퍼를 가져옵니다.
DSGetConstantBuffers1 도메인 셰이더 단계에서 사용하는 상수 버퍼를 가져옵니다.
DSGetSamplers 도메인 셰이더 단계에서 샘플러 상태 인터페이스 배열을 가져옵니다.
DSGetShader 디바이스에서 현재 설정된 도메인 셰이더를 가져옵니다.
DSGetShaderResources 도메인 셰이더 리소스를 가져옵니다.
DSSetConstantBuffers 도메인 셰이더 단계에서 사용하는 상수 버퍼를 설정합니다.
DSSetConstantBuffers1 도메인 셰이더 단계에서 사용하는 상수 버퍼를 설정합니다.
DSSetSamplers 샘플러 상태 배열을 도메인 셰이더 단계로 설정합니다.

DSSetShader 도메인 셰이더를 디바이스로 설정합니다.
DSSetShaderResources 셰이더 리소스 배열을 도메인 셰이더 단계에 바인딩합니다.
End 일련의 명령의 끝을 표시합니다.
EndEvent 이벤트 코드 섹션의 끝을 표시합니다.
EndEvent 애플리케이션에서 다양한 그래픽 명령의 끝에 주석을 달 수 있습니다.
Enter 디바이스의 중요 섹션을 입력합니다. (ID3D11Multithread.Enter)
ExecuteCommandList 명령 목록에서 디바이스로 명령을 큐에 추가합니다.
FinishCommandList 명령 목록을 만들고 그래픽 명령을 기록합니다.
플러시 명령 버퍼의 큐에 대기 중인 명령을 GPU(그래픽 처리 장치)로 보냅니다.
Flush1 지정된 컨텍스트 형식과 이벤트 쿼리를 만들기 위한 선택적 이벤트 핸들을 사용하여 명령 버퍼의 큐에 대기 중인 명령을 GPU(그래픽 처리 장치)로 보냅니다.
ForwardTransform 앞으로 FFT를 수행합니다.
GenerateHlsl function-linking-graph를 나타내는 Microsoft HLSL(High Level Shader Language) 셰이더 코드를 생성합니다.

GenerateMips 지정된 셰이더 리소스에 대한 mipmap을 생성합니다. (ID3D11DeviceContext.GenerateMips)
GetBaseClass 변수 기본 클래스 형식을 포함하는 ID3D11ShaderReflectionType 인터페이스 인터페이스를 가져옵니다.
GetBitwiseInstructionCount 비트 명령 수를 가져옵니다. (ID3D11ShaderReflection.GetBitwiseInstructionCount)
GetBreakOnCategory 해당 범주의 메시지가 스토리지 필터를 통과할 때 중단되는 메시지 범주를 가져옵니다. (ID3D11InfoQueue.GetBreakOnCategory)
GetBreakOnID 해당 식별자가 있는 메시지가 스토리지 필터를 통과할 때 중단할 메시지 식별자를 가져옵니다. (ID3D11InfoQueue.GetBreakOnID)
GetBreakOnSeverity 심각도 수준이 있는 메시지가 스토리지 필터를 통과할 때 중단할 메시지 심각도 수준을 가져옵니다. (ID3D11InfoQueue.GetBreakOnSeverity)
GetBuffer 이 메서드는 현재 ID3D11ShaderReflectionVariable의 버퍼를 반환합니다.
GetBufferPointer 데이터에 대한 포인터를 가져옵니다.
GetBufferSize 크기를 가져옵니다.
GetClassInstance 지정된 HLSL 클래스를 나타내는 클래스 인스턴스 개체를 가져옵니다.
GetClassLinkage 현재 HLSL 클래스와 연결된 ID3D11ClassLinkage 개체를 가져옵니다.

GetCompletedValue 펜스의 현재 값을 가져옵니다. (ID3D11Fence.GetCompletedValue)
GetConstantBufferByIndex ID3D11FunctionReflection::GetConstantBufferByIndex(d3d11shader.h) 메서드는 함수에 대한 인덱스별로 상수 버퍼를 가져옵니다.
GetConstantBufferByIndex ID3D11ShaderReflection::GetConstantBufferByIndex(d3d11shader.h) 메서드는 인덱스별로 상수 버퍼를 가져옵니다.
GetConstantBufferByName 함수의 이름으로 상수 버퍼를 가져옵니다. (ID3D11FunctionReflection.GetConstantBufferByName)
GetConstantBufferByName 이름별로 상수 버퍼를 가져옵니다. (ID3D11ShaderReflection.GetConstantBufferByName)
GetContextFlags 명령 목록을 만든 지연된 컨텍스트와 연결된 초기화 플래그를 가져옵니다.
GetContextFlags 현재 지연된 컨텍스트와 연결된 초기화 플래그를 가져옵니다.
GetConversionInstructionCount 변환 명령 수를 가져옵니다. (ID3D11ShaderReflection.GetConversionInstructionCount)
GetCreationFlags D3D11CreateDevice를 사용하여 디바이스를 만들기 위해 호출하는 동안 사용되는 플래그를 가져옵니다.
GetData GPU(그래픽 처리 장치)에서 비동기적으로 데이터를 가져옵니다.
GetDataSize ID3D11DeviceContext::GetData를 호출할 때 출력되는 데이터 크기(바이트)를 가져옵니다.

GetDesc 혼합 상태 개체를 만드는 데 사용한 혼합 상태에 대한 설명을 가져옵니다. (ID3D11BlendState.GetDesc)
GetDesc 버퍼 리소스의 속성을 가져옵니다. (ID3D11Buffer.GetDesc)
GetDesc 현재 HLSL 클래스에 대한 설명을 가져옵니다.
GetDesc 카운터 설명을 가져옵니다. (ID3D11Counter.GetDesc)
GetDesc depth-stencil-state 개체를 만드는 데 사용한 깊이 스텐실 상태에 대한 설명을 가져옵니다.
GetDesc 깊이 스텐실 보기를 가져옵니다. (ID3D11DepthStencilView.GetDesc)
GetDesc 쿼리 설명을 가져옵니다. (ID3D11Query.GetDesc)
GetDesc 래스터라이저 상태 개체를 만드는 데 사용한 래스터라이저 상태에 대한 설명을 가져옵니다. (ID3D11RasterizerState.GetDesc)
GetDesc 렌더링 대상 뷰의 속성을 가져옵니다. (ID3D11RenderTargetView.GetDesc)
GetDesc 샘플러 상태 개체를 만드는 데 사용한 샘플러 상태에 대한 설명을 가져옵니다.
GetDesc 셰이더 리소스 뷰의 설명을 가져옵니다. (ID3D11ShaderResourceView.GetDesc)
GetDesc 텍스처 리소스의 속성을 가져옵니다. (ID3D11Texture1D.GetDesc)

GetDesc 텍스처 리소스의 속성을 가져옵니다. (ID3D11Texture2D.GetDesc)
GetDesc 텍스처 리소스의 속성을 가져옵니다. (ID3D11Texture3D.GetDesc)
GetDesc 리소스에 대한 설명을 가져옵니다.
GetDesc 함수의 매개 변수에 대한 매개 변수 설명자 구조를 채웁니다. (ID3D11FunctionParameterReflection.GetDesc)
GetDesc 함수에 대한 함수 설명자 구조를 채웁니다. (ID3D11FunctionReflection.GetDesc)
GetDesc 라이브러리 리플렉션에 대한 라이브러리 설명자 구조를 채웁니다. (ID3D11LibraryReflection.GetDesc)
GetDesc 셰이더 설명을 가져옵니다. (ID3D11ShaderReflection.GetDesc)
GetDesc 상수 버퍼 설명을 가져옵니다. (ID3D11ShaderReflectionConstantBuffer.GetDesc)
GetDesc 셰이더-리플렉션 변수 형식에 대한 설명을 가져옵니다. (ID3D11ShaderReflectionType.GetDesc)
GetDesc 셰이더 변수 설명을 가져옵니다. (ID3D11ShaderReflectionVariable.GetDesc)
GetDesc1 혼합 상태 개체를 만드는 데 사용한 혼합 상태에 대한 설명을 가져옵니다. (ID3D11BlendState1.GetDesc1)

GetDesc1 래스터라이저 상태 개체를 만드는 데 사용한 래스터라이저 상태에 대한 설명을 가져옵니다. (ID3D11RasterizerState1.GetDesc1)
GetDesc1 쿼리 설명을 가져옵니다.
GetDesc1 렌더링 대상 뷰의 속성을 가져옵니다.
GetDesc1 셰이더 리소스 뷰의 설명을 가져옵니다.
GetDesc1 텍스처 리소스의 속성을 가져옵니다. (ID3D11Texture2D1.GetDesc1)
GetDesc1 텍스처 리소스의 속성을 가져옵니다. (ID3D11Texture3D1.GetDesc1)
GetDesc1 리소스에 대한 설명을 가져옵니다.
GetDesc2 래스터라이저 상태 개체를 만드는 데 사용한 래스터라이저 상태에 대한 설명을 가져옵니다. (ID3D11RasterizerState2.GetDesc2)
GetDevice 이 인터페이스를 만든 디바이스에 대한 포인터를 가져옵니다. (ID3D11DeviceChild.GetDevice)
GetDeviceRemovedReason 디바이스가 제거된 이유를 확인합니다. (ID3D11Device.GetDeviceRemovedReason)
GetEvictionPriority 리소스의 제거 우선 순위를 가져옵니다. (ID3D11Resource.GetEvictionPriority)
GetExceptionMode 예외 모드 플래그를 가져옵니다. (ID3D11Device.GetExceptionMode)

GetFeatureLevel 하드웨어 디바이스의 기능 수준을 가져옵니다. (ID3D11Device.GetFeatureLevel)
GetFeatureMask 어떤 디버그 기능이 켜지거나 꺼져 있는지 나타내는 플래그의 비트 필드를 가져옵니다. (ID3D11Debug.GetFeatureMask)
GetForwardScale 앞으로 변환할 크기를 가져옵니다.
GetFunctionByIndex ID3D11LibraryReflection::GetFunctionByIndex(d3d11shader.h) 메서드는 함수 리플렉터를 가져옵니다.
GetFunctionParameter 함수 매개 변수 리플렉터를 가져옵니다. (ID3D11FunctionReflection.GetFunctionParameter)
GetGSInputPrimitive geometry-shader input-primitive 설명을 가져옵니다. (ID3D11ShaderReflection.GetGSInputPrimitive)
GetHardwareProtectionState 하드웨어 보호를 사용할 수 있는지 여부를 가져옵니다.
GetImmediateContext 명령 목록을 재생할 수 있는 즉각적인 컨텍스트를 가져옵니다. (ID3D11Device.GetImmediateContext)
GetImmediateContext1 명령 목록을 재생할 수 있는 즉각적인 컨텍스트를 가져옵니다. (ID3D11Device1.GetImmediateContext1)
GetImmediateContext2 명령 목록을 재생할 수 있는 즉각적인 컨텍스트를 가져옵니다. (ID3D11Device2.GetImmediateContext2)

GetImmediateContext3 명령 목록을 재생할 수 있는 즉각적인 컨텍스트를 가져옵니다. (ID3D11Device3.GetImmediateContext3)
GetInitialRegisterContents 지정된 입력 레지스터의 초기 콘텐츠를 검색합니다.
GetInputParameterDesc 셰이더에 대한 입력 매개 변수 설명을 가져옵니다. (ID3D11ShaderReflection.GetInputParameterDesc)
GetInstanceName 현재 HLSL 클래스의 인스턴스 이름을 가져옵니다.
GetInterfaceByIndex 인덱스별로 인터페이스를 가져옵니다.
GetInterfaceSlot 인터페이스 포인터를 나타내는 변수의 해당 인터페이스 슬롯을 가져옵니다. (ID3D11ShaderReflectionVariable.GetInterfaceSlot)
GetInverseScale 역 변환의 배율을 가져옵니다.
Getlasterror function-linking-graph의 마지막 함수 호출에서 오류를 가져옵니다.
GetMemberTypeByIndex ID3D11ShaderReflectionType::GetMemberTypeByIndex(d3d11shader.h) 메서드는 인덱스별로 셰이더-리플렉션 변수 형식을 가져옵니다.
GetMemberTypeByName 이름으로 셰이더-리플렉션 변수 형식을 가져옵니다. (ID3D11ShaderReflectionType.GetMemberTypeByName)
GetMemberTypeName 셰이더 리플렉션 변수 형식을 가져옵니다. (ID3D11ShaderReflectionType.GetMemberTypeName)

GetMessage 메시지 큐에서 메시지를 가져옵니다. (ID3D11InfoQueue.GetMessage)
GetMessageCountLimit 메시지 큐에 추가할 수 있는 최대 메시지 수를 가져옵니다. (ID3D11InfoQueue.GetMessageCountLimit)
GetMinFeatureLevel 최소 기능 수준을 가져옵니다. (ID3D11ShaderReflection.GetMinFeatureLevel)
GetMovcInstructionCount Movc 명령 수를 가져옵니다. (ID3D11ShaderReflection.GetMovcInstructionCount)
GetMovInstructionCount Mov 명령 수를 가져옵니다. (ID3D11ShaderReflection.GetMovInstructionCount)
GetMultithreadProtected 다중 스레드 보호가 켜져 있는지 확인합니다.
GetMuteDebugOutput 디버그 출력을 켜거나 끄는 부울을 가져옵니다. (ID3D11InfoQueue.GetMuteDebugOutput)
GetNumInterfaces 인터페이스 수를 가져옵니다. (ID3D11ShaderReflectionType.GetNumInterfaces)
GetNumInterfaceSlots 셰이더의 인터페이스 슬롯 수를 가져옵니다. (ID3D11ShaderReflection.GetNumInterfaceSlots)
GetNumMessagesAllowedByStorageFilter 스토리지 필터를 통과하도록 허용된 메시지 수를 가져옵니다. (ID3D11InfoQueue.GetNumMessagesAllowedByStorageFilter)
GetNumMessagesDeniedByStorageFilter 스토리지 필터를 통해 전달이 거부된 메시지 수를 가져옵니다. (ID3D11InfoQueue.GetNumMessagesDeniedByStorageFilter)

GetNumMessagesDiscardedByMessageCountLimit 메시지 수 제한으로 인해 삭제된 메시지 수를 가져옵니다. (ID3D11InfoQueue.GetNumMessagesDiscardedByMessageCountLimit)
GetNumStoredMessages 현재 메시지 큐에 저장된 메시지 수를 가져옵니다. (ID3D11InfoQueue.GetNumStoredMessages)
GetNumStoredMessagesAllowedByRetrievalFilter 검색 필터를 통해 전달할 수 있는 메시지 수를 가져옵니다. (ID3D11InfoQueue.GetNumStoredMessagesAllowedByRetrievalFilter)
GetOutputParameterDesc 셰이더에 대한 출력 매개 변수 설명을 가져옵니다. (ID3D11ShaderReflection.GetOutputParameterDesc)
GetPatchConstantParameterDesc 셰이더에 대한 패치 상수 매개 변수 설명을 가져옵니다.
GetPredication 렌더링 조건자 상태를 가져옵니다. (ID3D11DeviceContext.GetPredication)
GetPresentPerRenderOpDelay IDXGISwapChain::P resent가 호출된 후 절전 모드로 이동할 시간(밀리초)을 가져옵니다.
GetPrivateData 디바이스에서 애플리케이션 정의 데이터를 가져옵니다.
GetPrivateData 디바이스 자식에서 애플리케이션 정의 데이터를 가져옵니다. (ID3D11DeviceChild.GetPrivateData)
GetReadRegister 추적의 단계에서 읽은 레지스터에 대한 정보를 검색합니다.
GetRequiresFlags 셰이더의 요구 사항을 나타내는 플래그 그룹을 가져옵니다. (ID3D11ShaderReflection.GetRequiresFlags)

GetResource 이 보기를 통해 액세스되는 리소스를 가져옵니다. (ID3D11View.GetResource)
GetResourceBindingDesc 리소스가 함수에 바인딩되는 방법에 대한 설명을 가져옵니다. (ID3D11FunctionReflection.GetResourceBindingDesc)
GetResourceBindingDesc 리소스가 셰이더에 바인딩되는 방법에 대한 설명을 가져옵니다. (ID3D11ShaderReflection.GetResourceBindingDesc)
GetResourceBindingDescByName 리소스가 함수에 바인딩되는 방법에 대한 설명을 가져옵니다. (ID3D11FunctionReflection.GetResourceBindingDescByName)
GetResourceBindingDescByName 리소스가 셰이더에 바인딩되는 방법에 대한 설명을 가져옵니다. (ID3D11ShaderReflection.GetResourceBindingDescByName)
GetResourceMinLOD 최소 LOD(세부 수준)를 가져옵니다.
GetResourceTiling 타일형 리소스가 타일로 분할되는 방법에 대한 정보를 가져옵니다. (ID3D11Device2.GetResourceTiling)
GetRetrievalFilter 검색 필터 스택의 맨 위에 있는 검색 필터를 가져옵니다. (ID3D11InfoQueue.GetRetrievalFilter)
GetRetrievalFilterStackSize 검색 필터 스택의 크기를 바이트 단위로 가져옵니다. (ID3D11InfoQueue.GetRetrievalFilterStackSize)
GetStatus 호출 애플리케이션이 Microsoft Direct3D 프로파일링 도구에서 실행 중인지 여부를 확인합니다.
GetStep 추적에서 지정된 단계에 대한 정보를 검색합니다.

GetStorageFilter 스토리지 필터 스택의 맨 위에 있는 스토리지 필터를 가져옵니다. (ID3D11InfoQueue.GetStorageFilter)
GetStorageFilterStackSize 스토리지 필터 스택의 크기를 바이트 단위로 가져옵니다. (ID3D11InfoQueue.GetStorageFilterStackSize)
GetSubType 클래스의 기본 클래스를 가져옵니다. (ID3D11ShaderReflectionType.GetSubType)
GetSwapChain 런타임이 IDXGISwapChain::P resent를 자동으로 호출하는 데 사용할 스왑 체인을 가져옵니다.
GetThreadGroupSize 셰이더 스레드 그룹 그리드의 X, Y 및 Z 차원의 크기(스레드 단위)를 검색합니다. (ID3D11ShaderReflection.GetThreadGroupSize)
GetTraceStats 추적에 대한 통계를 반환합니다.
GetType 디바이스 컨텍스트의 유형을 가져옵니다.
GetType 리소스의 형식을 가져옵니다. (ID3D11Resource.GetType)
GetType 셰이더 변수 형식을 가져옵니다. (ID3D11ShaderReflectionVariable.GetType)
GetTypeName 현재 HLSL 클래스의 형식을 가져옵니다.
GetUseRef ID3D11SwitchToRef::GetUseRef 메서드

GetVariableByIndex ID3D11ShaderReflectionConstantBuffer::GetVariableByIndex(d3d11shader.h) 메서드는 인덱스별로 셰이더 리플렉션 변수를 가져옵니다.
GetVariableByName 이름으로 변수를 가져옵니다. (ID3D11FunctionReflection.GetVariableByName)
GetVariableByName 이름으로 변수를 가져옵니다. (ID3D11ShaderReflection.GetVariableByName)
GetVariableByName 이름으로 셰이더 리플렉션 변수를 가져옵니다. (ID3D11ShaderReflectionConstantBuffer.GetVariableByName)
GetWrittenRegister 추적의 단계에서 작성한 레지스터에 대한 정보를 검색합니다.
GSGetConstantBuffers 기하 도형 셰이더 파이프라인 단계에서 사용하는 상수 버퍼를 가져옵니다. (ID3D11DeviceContext.GSGetConstantBuffers)
GSGetConstantBuffers1 기하 도형 셰이더 파이프라인 단계에서 사용하는 상수 버퍼를 가져옵니다.
GSGetSamplers 기하 도형 셰이더 파이프라인 단계에서 샘플러 상태 인터페이스의 배열을 가져옵니다.
GSGetShader 현재 디바이스에 설정된 기하 도형 셰이더를 가져옵니다. (ID3D11DeviceContext.GSGetShader)
GSGetShaderResources 기하 도형 셰이더 리소스를 가져옵니다. (ID3D11DeviceContext.GSGetShaderResources)
GSSetConstantBuffers 기하 도형 셰이더 파이프라인 단계에서 사용하는 상수 버퍼를 설정합니다.

GSSetConstantBuffers1 기하 도형 셰이더 파이프라인 단계에서 사용하는 상수 버퍼를 설정합니다.
GSSetSamplers 샘플러 상태의 배열을 기하 도형 셰이더 파이프라인 단계로 설정합니다. (ID3D11DeviceContext.GSSetSamplers)
GSSetShader 기하 도형 셰이더를 디바이스로 설정합니다. (ID3D11DeviceContext.GSSetShader)
GSSetShaderResources 셰이더 리소스 배열을 기하 도형 셰이더 단계에 바인딩합니다. (ID3D11DeviceContext.GSSetShaderResources)
HSGetConstantBuffers 헐 셰이더 단계에서 사용하는 상수 버퍼를 가져옵니다.
HSGetConstantBuffers1 헐 셰이더 단계에서 사용하는 상수 버퍼를 가져옵니다.
HSGetSamplers 헐 셰이더 단계에서 샘플러 상태 인터페이스 배열을 가져옵니다.
HSGetShader 디바이스에서 현재 설정된 헐 셰이더를 가져옵니다.
HSGetShaderResources 헐 셰이더 리소스를 가져옵니다.
HSSetConstantBuffers 헐 셰이더 단계에서 사용하는 상수 버퍼를 설정합니다.
HSSetConstantBuffers1 파이프라인의 헐 셰이더 단계에서 사용하는 상수 버퍼를 설정합니다.
HSSetSamplers 샘플러 상태 배열을 헐 셰이더 단계로 설정합니다.

HSSetShader 헐 셰이더를 디바이스로 설정합니다.
HSSetShaderResources 셰이더 리소스 배열을 헐 셰이더 단계에 바인딩합니다.
IAGetIndexBuffer 입력 어셈블러 단계에 바인딩된 인덱스 버퍼에 대한 포인터를 가져옵니다. (ID3D11DeviceContext.IAGetIndexBuffer)
IAGetInputLayout 입력 어셈블러 단계에 바인딩된 입력 레이아웃 개체에 대한 포인터를 가져옵니다. (ID3D11DeviceContext.IAGetInputLayout)
IAGetPrimitiveTopology 기본 형식 및 입력 어셈블러 단계의 입력 데이터를 설명하는 데이터 순서에 대한 정보를 가져옵니다. (ID3D11DeviceContext.IAGetPrimitiveTopology)
IAGetVertexBuffers 입력 어셈블러 단계에 바인딩된 꼭짓점 버퍼를 가져옵니다. (ID3D11DeviceContext.IAGetVertexBuffers)
IASetIndexBuffer 인덱스 버퍼를 입력 어셈블러 단계에 바인딩합니다. (ID3D11DeviceContext.IASetIndexBuffer)
IASetInputLayout 입력 레이아웃 개체를 input-assembler 단계에 바인딩합니다. (ID3D11DeviceContext.IASetInputLayout)
IASetPrimitiveTopology 기본 형식 및 입력 어셈블러 단계의 입력 데이터를 설명하는 데이터 순서에 대한 정보를 바인딩합니다. (ID3D11DeviceContext.IASetPrimitiveTopology)
IASetVertexBuffers 꼭짓점 버퍼 배열을 입력 어셈블러 단계에 바인딩합니다. (ID3D11DeviceContext.IASetVertexBuffers)

ImplementsInterface 클래스 형식이 인터페이스를 구현하는지 여부를 나타냅니다. (ID3D11ShaderReflectionType.ImplementsInterface)
InverseTransform 역 FFT를 수행합니다.
IsAnnotationEnabled 앱에서 캡처 또는 프로파일링 요청이 사용되는 시기를 확인할 수 있습니다.
IsEqual 두 ID3D11ShaderReflectionType 인터페이스 포인터의 기본 형식이 같은지 여부를 나타냅니다.
IsOfType 변수가 지정된 형식인지 여부를 나타냅니다. (ID3D11ShaderReflectionType.IsOfType)
IsSampleFrequencyShader 셰이더가 샘플 빈도 셰이더인지 여부를 나타냅니다. (ID3D11ShaderReflection.IsSampleFrequencyShader)
나가기 디바이스의 중요 섹션을 그대로 둡니다. (ID3D11Multithread.Leave)
링크 셰이더를 연결하고 Direct3D 런타임에서 사용할 수 있는 셰이더 Blob을 생성합니다.
Map 하위 리소스에 포함된 데이터에 대한 포인터를 가져오고 해당 하위 리소스에 대한 GPU 액세스를 거부합니다.
Multiscan 시퀀스의 멀티스캔을 수행합니다.
OMGetBlendState 출력 병합기 단계의 혼합 상태를 가져옵니다. (ID3D11DeviceContext.OMGetBlendState)

OMGetDepthStencilState 출력 병합기 단계의 깊이 스텐실 상태를 가져옵니다. (ID3D11DeviceContext.OMGetDepthStencilState)
OMGetRenderTargets 출력 병합기에 바인딩된 리소스에 대한 포인터를 가져옵니다. (ID3D11DeviceContext.OMGetRenderTargets)
OMGetRenderTargetsAndUnorderedAccessViews 출력 병합기에 바인딩된 리소스에 대한 포인터를 가져옵니다. (ID3D11DeviceContext.OMGetRenderTargetsAndUnorderedAccessViews)
OMSetBlendState 출력 병합기 단계의 혼합 상태를 설정합니다. (ID3D11DeviceContext.OMSetBlendState)
OMSetDepthStencilState 출력 병합기 단계의 깊이 스텐실 상태를 설정합니다. (ID3D11DeviceContext.OMSetDepthStencilState)
OMSetRenderTargets 하나 이상의 렌더링 대상을 원자성으로 바인딩하고 깊이 스텐실 버퍼를 출력 병합 단계에 바인딩합니다.
OMSetRenderTargetsAndUnorderedAccessViews 출력 병합기에 리소스를 바인딩합니다.
열기 셰이더의 내용을 열고 읽기 위한 사용자 구현 메서드
OpenSharedFence HANDLE 및 REFIID를 사용하여 공유 펜스에 대한 핸들을 엽니다.
OpenSharedResource 디바이스에 다른 디바이스에서 만든 공유 리소스에 대한 액세스 권한을 부여합니다.
OpenSharedResource1 핸들에서 참조하고 다른 디바이스에서 만든 공유 리소스에 대한 액세스 권한을 디바이스에 제공합니다.

OpenSharedResourceByName 이름으로 참조되고 다른 디바이스에서 만든 공유 리소스에 대한 액세스 권한을 디바이스에 제공합니다.
PassValue 원본 연결 노드에서 대상 연결 노드로 값을 전달합니다.
PassValueWithSwizzle 원본 연결 노드에서 대상 연결 노드로 스위즐이 있는 값을 전달합니다.
PopRetrievalFilter 검색 필터 스택의 맨 위에서 검색 필터를 팝합니다. (ID3D11InfoQueue.PopRetrievalFilter)
PopStorageFilter 스토리지 필터 스택의 맨 위에서 스토리지 필터를 팝합니다. (ID3D11InfoQueue.PopStorageFilter)
PSGetConstantBuffers 픽셀 셰이더 파이프라인 단계에서 사용하는 상수 버퍼를 가져옵니다. (ID3D11DeviceContext.PSGetConstantBuffers)
PSGetConstantBuffers1 픽셀 셰이더 파이프라인 단계에서 사용하는 상수 버퍼를 가져옵니다.
PSGetSamplers 픽셀 셰이더 파이프라인 단계에서 샘플러 상태 배열을 가져옵니다. (ID3D11DeviceContext.PSGetSamplers)
PSGetShader 디바이스에서 현재 설정된 픽셀 셰이더를 가져옵니다. (ID3D11DeviceContext.PSGetShader)
PSGetShaderResources 픽셀 셰이더 리소스를 가져옵니다. (ID3D11DeviceContext.PSGetShaderResources)
PSSelectStamp 지정된 픽셀 셰이더 스탬프를 설정합니다.

PSSetConstantBuffers 픽셀 셰이더 파이프라인 단계에서 사용하는 상수 버퍼를 설정합니다.
PSSetConstantBuffers1 픽셀 셰이더 파이프라인 단계에서 사용하는 상수 버퍼를 설정하고 셰이더가 버퍼의 다른 부분에 액세스할 수 있도록 합니다.
PSSetSamplers 샘플러 상태 배열을 픽셀 셰이더 파이프라인 단계로 설정합니다. (ID3D11DeviceContext.PSSetSamplers)
PSSetShader 디바이스에 픽셀 셰이더를 설정합니다. (ID3D11DeviceContext.PSSetShader)
PSSetShaderResources 셰이더 리소스 배열을 픽셀 셰이더 단계에 바인딩합니다. (ID3D11DeviceContext.PSSetShaderResources)
PushCopyOfRetrievalFilter 검색 필터 스택의 맨 위에 있는 검색 필터 복사본을 검색 필터 스택에 푸시합니다. (ID3D11InfoQueue.PushCopyOfRetrievalFilter)
PushCopyOfStorageFilter 현재 스토리지 필터 스택의 맨 위에 있는 스토리지 필터 복사본을 스토리지 필터 스택에 푸시합니다. (ID3D11InfoQueue.PushCopyOfStorageFilter)
PushEmptyRetrievalFilter 빈 검색 필터를 검색 필터 스택에 푸시합니다. (ID3D11InfoQueue.PushEmptyRetrievalFilter)
PushEmptyStorageFilter 빈 스토리지 필터를 스토리지 필터 스택에 푸시합니다. (ID3D11InfoQueue.PushEmptyStorageFilter)
PushRetrievalFilter 검색 필터를 검색 필터 스택에 푸시합니다. (ID3D11InfoQueue.PushRetrievalFilter)
PushStorageFilter 스토리지 필터를 스토리지 필터 스택에 푸시합니다. (ID3D11InfoQueue.PushStorageFilter)

ReadFromSubresource NULL D3D11_MAPPED_SUBRESOURCEparameter 제공하는 ID3D11DeviceContext3::Mapwhile를 사용하여 매핑된 D3D11_USAGE_DEFAULTtexture 데이터를 복사합니다.
RegisterDestructionCallback 이 ID3DDestructionNotifier 가 만들어진 개체의 파기 시 호출할 사용자 정의 콜백을 등록합니다.
RegisterDeviceRemovedEvent "디바이스 제거됨" 이벤트를 등록하고 비동기 알림 메커니즘을 사용하여 어떤 이유로든 Direct3D 디바이스가 제거된 시기를 나타냅니다.
ReportLiveDeviceObjects 디바이스 개체의 수명에 대한 정보를 보고합니다.
ResetTrace 셰이더 추적 개체를 다시 설정합니다.
ResizeTilePool 타일 풀의 크기를 조정합니다.
ResolveSubresource 다중 샘플링된 리소스를 다중 샘플링되지 않은 리소스에 복사합니다.
RSGetScissorRects 래스터라이저 단계에 바인딩된 가위 사각형의 배열을 가져옵니다. (ID3D11DeviceContext.RSGetScisorRects)
RSGetState 파이프라인의 래스터라이저 단계에서 래스터라이저 상태를 가져옵니다. (ID3D11DeviceContext.RSGetState)
RSGetViewports 래스터라이저 단계에 바인딩된 뷰포트의 배열을 가져옵니다.
RSSetScissorRects 가위 사각형 배열을 래스터라이저 단계에 바인딩합니다. (ID3D11DeviceContext.RSSetScisorRects)

RSSetState 파이프라인의 래스터라이저 단계에 대한 래스터라이저 상태를 설정합니다. (ID3D11DeviceContext.RSSetState)
RSSetViewports 뷰포트 배열을 파이프라인의 래스터라이저 단계에 바인딩합니다. (ID3D11DeviceContext.RSSetViewports)
검사 시퀀스의 분리되지 않은 검사를 수행합니다.
SegScan 시퀀스의 분할된 검사를 수행합니다.
SetBreakOnCategory 해당 범주의 메시지가 스토리지 필터를 통과할 때 중단되도록 메시지 범주를 설정합니다. (ID3D11InfoQueue.SetBreakOnCategory)
SetBreakOnID 해당 식별자가 있는 메시지가 스토리지 필터를 통과할 때 중단되도록 메시지 식별자를 설정합니다. (ID3D11InfoQueue.SetBreakOnID)
SetBreakOnSeverity 심각도 수준이 있는 메시지가 스토리지 필터를 통과할 때 중단되도록 메시지 심각도 수준을 설정합니다. (ID3D11InfoQueue.SetBreakOnSeverity)
SetEventOnCompletion 펜스가 특정 값에 도달할 때 발생해야 하는 이벤트를 지정합니다. (ID3D11Fence.SetEventOnCompletion)
SetEvictionPriority 리소스의 제거 우선 순위를 설정합니다. (ID3D11Resource.SetEvictionPriority)
SetExceptionMode 예외 모드 플래그를 가져옵니다. (ID3D11Device.SetExceptionMode)
SetFeatureMask 디버그 기능을 켜고 끄는 플래그의 비트 필드를 설정합니다. (ID3D11Debug.SetFeatureMask)

SetForwardScale 정방향 변환에 사용되는 배율을 설정합니다.
SetHardwareProtectionState 하드웨어 보호 상태를 설정합니다.
SetInputSignature function-linking-graph의 입력 서명을 설정합니다.
SetInverseScale 역 변환에 사용되는 배율을 설정합니다.
SetMarker 코드에서 단일 실행 지점을 표시합니다.
SetMarkerInt 애플리케이션에서 그래픽 명령에 주석을 달 수 있습니다.
SetMessageCountLimit 메시지 큐에 추가할 수 있는 최대 메시지 수를 설정합니다. (ID3D11InfoQueue.SetMessageCountLimit)
SetMultithreadProtected 다중 스레드 보호를 켜거나 끕니다.
SetMuteDebugOutput 디버그 출력을 켜거나 끄는 부울을 설정합니다. (ID3D11InfoQueue.SetMuteDebugOutput)
SetOutputSignature function-linking-graph의 출력 서명을 설정합니다.
SetPredication 렌더링 조건자를 설정합니다. (ID3D11DeviceContext.SetPredication)
SetPresentPerRenderOpDelay IDXGISwapChain::P resent가 호출된 후 절전 모드로 설정할 시간(밀리초)을 설정합니다.

SetPrivateData 데이터를 디바이스로 설정하고 해당 데이터를 GUID와 연결합니다. (ID3D11Device.SetPrivateData)
SetPrivateData 애플리케이션 정의 데이터를 디바이스 자식으로 설정하고 해당 데이터를 애플리케이션 정의 GUID와 연결합니다. (ID3D11DeviceChild.SetPrivateData)
SetPrivateDataInterface IUnknown 파생 인터페이스를 이 디바이스 자식과 연결하고 해당 인터페이스를 애플리케이션 정의 GUID와 연결합니다. (ID3D11Device.SetPrivateDataInterface)
SetPrivateDataInterface IUnknown 파생 인터페이스를 이 디바이스 자식과 연결하고 해당 인터페이스를 애플리케이션 정의 GUID와 연결합니다. (ID3D11DeviceChild.SetPrivateDataInterface)
SetResourceMinLOD 리소스에 대한 최소 LOD(세부 수준)를 설정합니다.
SetScanDirection 검사를 수행할 방향을 설정합니다. (ID3DX11Scan.SetScanDirection)
SetScanDirection 검사를 수행할 방향을 설정합니다. (ID3DX11SegmentedScan.SetScanDirection)
SetShaderTrackingOptions 특정 셰이더에 대한 참조 래스터라이저의 경합 상태 추적 옵션을 설정합니다.
SetShaderTrackingOptionsByType 지정된 리소스 종류에 대한 참조 래스터라이저의 기본 경합 조건 추적 옵션을 설정합니다.
SetSwapChain 런타임이 IDXGISwapChain::P resent를 자동으로 호출하는 데 사용할 스왑 체인을 설정합니다.
SetTrackingOptions 특정 리소스 종류에 대한 GPU(그래픽 처리 장치) 디버그 참조 기본 추적 옵션을 설정합니다.

SetTrackingOptions GPU(그래픽 처리 장치) 디버그 참조 추적 옵션을 설정합니다.
SetUseRef ID3D11SwitchToRef::SetUseRef 메서드
신호 이전 작업이 모두 완료된 후 지정된 값으로 펜스를 업데이트.
SOGetTargets 파이프라인의 스트림 출력 단계에 대한 대상 출력 버퍼를 가져옵니다.
SOSetTargets 파이프라인의 스트림 출력 단계에 대한 대상 출력 버퍼를 설정합니다.
SwapDeviceContextState 지정된 컨텍스트 상태 개체를 활성화하고 현재 디바이스 동작을 Direct3D 11, Direct3D 10.1 또는 Direct3D 10으로 변경합니다.
TiledResourceBarrier 여러 타일식 리소스 간의 데이터 액세스 순서 제약 조건을 지정합니다.
TraceReady 셰이더 추적이 기록되고 사용할 준비가 되도록 지정합니다.
Unmap 리소스에 대한 포인터를 무효화하고 해당 리소스에 대한 GPU의 액세스를 다시 활성화합니다.
UnregisterDestructionCallback RegisterDestructionCallback에 등록된 콜백을 등록 취소합니다.
UnregisterDeviceRemoved "디바이스 제거됨" 이벤트를 등록 취소합니다.
UpdateSubresource CPU는 메모리에서 mappable이 아닌 메모리에서 만든 하위 리소스로 데이터를 복사합니다. (ID3D11DeviceContext.UpdateSubresource)

UpdateSubresource1 CPU는 메모리에서 mappable이 아닌 메모리에서 만든 하위 리소스로 데이터를 복사합니다. (ID3D11DeviceContext1.UpdateSubresource1)
UpdateTileMappings 타일 풀의 메모리 위치에 타일 리소스의 타일 위치 매핑을 업데이트.
UpdateTiles 앱 메모리에서 바둑판형 리소스로 복사하여 타일을 업데이트.
UseLibrary 연결에 사용할 라이브러리 모듈의 인스턴스를 추가합니다.
ValidateContext 그리기 파이프라인 상태가 유효한지 확인합니다.
ValidateContextForDispatch 디스패치 파이프라인 상태가 유효한지 확인합니다.
VSGetConstantBuffers 꼭짓점 셰이더 파이프라인 단계에서 사용하는 상수 버퍼를 가져옵니다. (ID3D11DeviceContext.VSGetConstantBuffers)
VSGetConstantBuffers1 꼭짓점 셰이더 파이프라인 단계에서 사용하는 상수 버퍼를 가져옵니다.
VSGetSamplers 꼭짓점 셰이더 파이프라인 단계에서 샘플러 상태 배열을 가져옵니다. (ID3D11DeviceContext.VSGetSamplers)
VSGetShader 현재 디바이스에 설정된 꼭짓점 셰이더를 가져옵니다. (ID3D11DeviceContext.VSGetShader)
VSGetShaderResources 꼭짓점 셰이더 리소스를 가져옵니다. (ID3D11DeviceContext.VSGetShaderResources)

VSSetConstantBuffers 꼭짓점 셰이더 파이프라인 단계에서 사용하는 상수 버퍼를 설정합니다.
VSSetConstantBuffers1 꼭짓점 셰이더 파이프라인 단계에서 사용하는 상수 버퍼를 설정합니다.
VSSetSamplers 샘플러 상태 배열을 꼭짓점 셰이더 파이프라인 단계로 설정합니다. (ID3D11DeviceContext.VSSetSamplers)
VSSetShader 꼭짓점 셰이더를 디바이스로 설정합니다. (ID3D11DeviceContext.VSSetShader)
VSSetShaderResources 셰이더 리소스 배열을 꼭짓점 셰이더 단계에 바인딩합니다.
Wait 지정된 펜스가 지정된 값에 도달하거나 초과할 때까지 기다린 후 이후 작업을 시작할 수 있습니다.
WriteToSubresource NULL D3D11_MAPPED_SUBRESOURCEparameter 제공하는 ID3D11DeviceContext3::Mapwhile를 사용하여 매핑된 D3D11_USAGE_DEFAULTtexture 데이터를 복사합니다.

인터페이스

ID3D10Blob 이 인터페이스는 임의 길이 데이터를 반환하는 데 사용됩니다.
ID3D11Asynchronous 이 인터페이스는 GPU에서 데이터를 비동기적으로 검색하는 메서드를 캡슐화합니다. (ID3D11Asynchronous)
ID3D11BlendState 혼합 상태 인터페이스에는 출력 병합기 단계에 바인딩할 수 있는 혼합 상태에 대한 설명이 있습니다.

ID3D11BlendState1 혼합 상태 인터페이스에는 출력 병합기 단계에 바인딩할 수 있는 혼합 상태에 대한 설명이 있습니다. 이 혼합 상태 인터페이스는 논리 작업과 혼합 작업을 지원합니다.
ID3D11Buffer 버퍼 인터페이스는 비정형 메모리인 버퍼 리소스에 액세스합니다. 버퍼는 일반적으로 꼭짓점 또는 인덱스 데이터를 저장합니다. (ID3D11Buffer)
ID3D11ClassInstance 이 인터페이스는 HLSL 클래스를 캡슐화합니다.
ID3D11ClassLinkage 이 인터페이스는 HLSL 동적 연결을 캡슐화합니다.
ID3D11CommandList ID3D11CommandList 인터페이스는 재생을 위한 그래픽 명령 목록을 캡슐화합니다.
ID3D11ComputeShader 컴퓨팅 셰이더 인터페이스는 컴퓨팅 셰이더 단계를 제어하는 실행 프로그램(컴퓨팅 셰이더)을 관리합니다.
ID3D11Counter 이 인터페이스는 GPU 성능을 측정하기 위한 메서드를 캡슐화합니다. (ID3D11Counter)
ID3D11Debug 디버그 인터페이스는 디버그 설정을 제어하고 파이프라인 상태의 유효성을 검사하며 디버그 계층이 켜져 있는 경우에만 사용할 수 있습니다. (ID3D11Debug)
ID3D11DepthStencilState depth-stencil-state 인터페이스는 출력 병합기에 바인딩할 수 있는 깊이 스텐실 상태에 대한 설명을 포함합니다.
ID3D11DepthStencilView 깊이 스텐실 뷰 인터페이스는 깊이 스텐실 테스트 중에 텍스처 리소스에 액세스합니다. (ID3D11DepthStencilView)
ID3D11Device 디바이스 인터페이스는 가상 어댑터를 나타냅니다. 리소스를 만드는 데 사용됩니다.

ID3D11Device1 디바이스 인터페이스는 가상 어댑터를 나타냅니다. 리소스를 만드는 데 사용됩니다. ID3D11Device1은 ID3D11Device의 메서드에 새 메서드를 추가합니다.
ID3D11Device2 디바이스 인터페이스는 가상 어댑터를 나타냅니다. 리소스를 만드는 데 사용됩니다. ID3D11Device2는 ID3D11Device1의 메서드에 새 메서드를 추가합니다.
ID3D11Device3 디바이스 인터페이스는 가상 어댑터를 나타냅니다. 리소스를 만드는 데 사용됩니다. ID3D11Device3은 ID3D11Device2의 메서드에 새 메서드를 추가합니다.
ID3D11Device4 디바이스 인터페이스는 가상 어댑터를 나타냅니다. 리소스를 만드는 데 사용됩니다. ID3D11Device4는 RegisterDeviceRemovedEvent 및 UnregisterDeviceRemoved와 같은 ID3D11Device3의 메서드에 새 메서드를 추가합니다.
ID3D11Device5 디바이스 인터페이스는 가상 어댑터를 나타냅니다. 리소스를 만드는 데 사용됩니다. ID3D11Device5는 ID3D11Device4의 메서드에 새 메서드를 추가합니다.
ID3D11DeviceChild 디바이스-자식 인터페이스는 디바이스에서 사용하는 데이터에 액세스합니다. (ID3D11DeviceChild)
ID3D11DeviceContext ID3D11DeviceContext 인터페이스는 렌더링 명령을 생성하는 디바이스 컨텍스트를 나타냅니다.
ID3D11DeviceContext1 디바이스 컨텍스트 인터페이스는 디바이스 컨텍스트를 나타냅니다. 명령을 렌더링하는 데 사용됩니다. ID3D11DeviceContext1은 ID3D11DeviceContext의 메서드에 새 메서드를 추가합니다.
ID3D11DeviceContext2 디바이스 컨텍스트 인터페이스는 디바이스 컨텍스트를 나타냅니다. 명령을 렌더링하는 데 사용됩니다. ID3D11DeviceContext2는 ID3D11DeviceContext1의 메서드에 새 메서드를 추가합니다.
ID3D11DeviceContext3 디바이스 컨텍스트 인터페이스는 디바이스 컨텍스트를 나타냅니다. 명령을 렌더링하는 데 사용됩니다. ID3D11DeviceContext3은 ID3D11DeviceContext2의 메서드에 새 메서드를 추가합니다.

ID3D11DeviceContext4 디바이스 컨텍스트 인터페이스는 디바이스 컨텍스트를 나타냅니다. 명령을 렌더링하는 데 사용됩니다. ID3D11DeviceContext4는 ID3D11DeviceContext3의 메서드에 새 메서드를 추가합니다.
ID3D11DomainShader 도메인 셰이더 인터페이스는 도메인 셰이더 단계를 제어하는 실행 프로그램(도메인 셰이더)을 관리합니다.
ID3D11Fence 펜스, CPU 동기화에 사용되는 개체 및 하나 이상의 GPU를 나타냅니다. (ID3D11Fence)
ID3D11FunctionLinkingGraph 함수 연결-그래프 인터페이스는 서로 값을 전달하는 미리 컴파일된 함수 호출 시퀀스로 구성된 셰이더를 생성하는 데 사용됩니다.
ID3D11FunctionParameterReflection function-parameter-reflection 인터페이스는 function-parameter 정보에 액세스합니다. (ID3D11FunctionParameterReflection)
ID3D11FunctionReflection 함수-리플렉션 인터페이스는 함수 정보에 액세스합니다. (ID3D11FunctionReflection)
ID3D11GeometryShader 기하 도형 셰이더 인터페이스는 기하 도형 셰이더 단계를 제어하는 실행 프로그램(기하 도형 셰이더)을 관리합니다. (ID3D11GeometryShader)
ID3D11HullShader 헐 셰이더 인터페이스는 헐 셰이더 단계를 제어하는 실행 프로그램(헐 셰이더)을 관리합니다.
ID3D11InfoQueue 정보 큐 인터페이스는 디버그 메시지를 저장, 검색 및 필터링합니다. 큐는 메시지 큐, 선택적 스토리지 필터 스택 및 선택적 검색 필터 스택으로 구성됩니다. (ID3D11InfoQueue)
ID3D11InputLayout 입력 레이아웃 인터페이스에는 메모리에 배치된 꼭짓점 데이터를 그래픽 파이프라인의 입력 어셈블러 단계에 공급하는 방법에 대한 정의가 있습니다.

ID3D11LibraryReflection 라이브러리 리플렉션 인터페이스는 라이브러리 정보에 액세스합니다. (ID3D11LibraryReflection)
ID3D11Linker 링커 인터페이스는 셰이더 모듈을 연결하는 데 사용됩니다.
ID3D11LinkingNode 연결 노드 인터페이스는 셰이더 연결에 사용됩니다.
ID3D11모듈 모듈 인터페이스는 리소스 다시 바인딩에 사용되는 모듈의 인스턴스를 만듭니다.
ID3D11ModuleInstance 모듈 인스턴스 인터페이스는 리소스 다시 바인딩에 사용됩니다.
ID3D11Multithread 다중 스레드 애플리케이션의 중요한 섹션에 대한 스레딩 보호를 제공합니다.
ID3D11PixelShader 픽셀 셰이더 인터페이스는 픽셀 셰이더 단계를 제어하는 실행 프로그램(픽셀 셰이더)을 관리합니다. (ID3D11PixelShader)
ID3D11Predicate 조건자 인터페이스는 이전 그리기 호출의 결과에 따라 기하 도형을 처리해야 하는지 여부를 결정합니다. (ID3D11Predicate)
ID3D11Query 쿼리 인터페이스는 GPU의 정보를 쿼리합니다. (ID3D11Query)
ID3D11Query1 GPU(그래픽 처리 장치)에서 정보를 쿼리하기 위한 쿼리 개체를 나타냅니다.
ID3D11RasterizerState 래스터라이저 상태 인터페이스는 래스터라이저 단계에 바인딩할 수 있는 래스터라이저 상태에 대한 설명을 포함합니다.

ID3D11RasterizerState1 래스터라이저 상태 인터페이스는 래스터라이저 단계에 바인딩할 수 있는 래스터라이저 상태에 대한 설명을 포함합니다. 이 래스터라이저 상태 인터페이스는 강제 샘플 수를 지원합니다.
ID3D11RasterizerState2 래스터라이저 상태 인터페이스는 래스터라이저 단계에 바인딩할 수 있는 래스터라이저 상태에 대한 설명을 포함합니다. 이 래스터라이저 상태 인터페이스는 강제 샘플 수 및 보수적 래스터화 모드를 지원합니다.
ID3D11RefDefaultTrackingOptions 기본 추적 인터페이스 집합은 기본 추적 옵션을 참조합니다.
ID3D11RefTrackingOptions 추적 인터페이스는 참조 추적 옵션을 설정합니다.
ID3D11RenderTargetView render-target-view 인터페이스는 렌더링 중에 액세스할 수 있는 렌더링 대상 하위 리소스를 식별합니다. (ID3D11RenderTargetView)
ID3D11RenderTargetView1 render-target-view 인터페이스는 렌더링 중에 액세스할 수 있는 렌더링 대상 하위 리소스를 나타냅니다.
ID3D11Resource 리소스 인터페이스는 모든 리소스에 대한 일반적인 작업을 제공합니다. (ID3D11Resource)
ID3D11SamplerState 샘플러 상태 인터페이스는 텍스처 샘플 작업에서 참조하기 위해 파이프라인의 셰이더 단계에 바인딩할 수 있는 샘플러 상태에 대한 설명을 포함합니다.
ID3D11ShaderReflection 셰이더 리플렉션 인터페이스는 셰이더 정보에 액세스합니다. (ID3D11ShaderReflection)
ID3D11ShaderReflectionConstantBuffer 이 셰이더 리플렉션 인터페이스는 상수 버퍼에 대한 액세스를 제공합니다. (ID3D11ShaderReflectionConstantBuffer)

ID3D11ShaderReflectionType 이 셰이더 리플렉션 인터페이스는 변수 형식에 대한 액세스를 제공합니다. (ID3D11ShaderReflectionType)
ID3D11ShaderReflectionVariable 이 셰이더 리플렉션 인터페이스는 변수에 대한 액세스를 제공합니다. (ID3D11ShaderReflectionVariable)
ID3D11ShaderResourceView 셰이더 리소스 뷰 인터페이스는 렌더링 중에 셰이더가 액세스할 수 있는 하위 리소스를 지정합니다. 셰이더 리소스의 예로는 상수 버퍼, 텍스처 버퍼 및 텍스처가 있습니다.
ID3D11ShaderResourceView1 셰이더 리소스 뷰 인터페이스는 렌더링 중에 셰이더가 액세스할 수 있는 하위 리소스를 나타냅니다. 셰이더 리소스의 예로는 상수 버퍼, 텍스처 버퍼 및 텍스처가 있습니다.
ID3D11ShaderTrace ID3D11ShaderTrace 인터페이스는 셰이더 실행 추적을 가져오는 메서드를 구현합니다.
ID3D11ShaderTraceFactory ID3D11ShaderTraceFactory 인터페이스는 셰이더 추적 정보 개체를 생성하는 메서드를 구현합니다.
ID3D11SwitchToRef ID3D11SwitchToRef 인터페이스
ID3D11Texture1D 1D 텍스처 인터페이스는 구조화된 메모리인 텍셀 데이터에 액세스합니다. (ID3D11Texture1D)
ID3D11Texture2D 2D 텍스처 인터페이스는 구조화된 메모리인 텍셀 데이터를 관리합니다. (ID3D11Texture2D)
ID3D11Texture2D1 2D 텍스처 인터페이스는 구조화된 메모리인 텍셀 데이터를 나타냅니다.
ID3D11Texture3D 3D 텍스처 인터페이스는 구조화된 메모리인 텍셀 데이터에 액세스합니다. (ID3D11Texture3D)

ID3D11Texture3D1 3D 텍스처 인터페이스는 구조화된 메모리인 텍셀 데이터를 나타냅니다.
ID3D11TracingDevice 추적 디바이스 인터페이스는 셰이더 추적 정보를 설정하여 셰이더 실행의 정확한 로깅 및 재생을 가능하게 합니다.
ID3D11UnorderedAccessView 뷰 인터페이스는 렌더링 중에 파이프라인이 액세스할 수 있는 리소스 부분을 지정합니다. (ID3D11UnorderedAccessView)
ID3D11UnorderedAccessView1 unordered-access-view 인터페이스는 렌더링 중에 파이프라인이 액세스할 수 있는 리소스 부분을 나타냅니다.
ID3D11VertexShader 꼭짓점 셰이더 인터페이스는 꼭짓점 셰이더 단계를 제어하는 실행 프로그램(꼭짓점 셰이더)을 관리합니다. (ID3D11VertexShader)
ID3D11VideoContext3 Microsoft Direct3D 11 디바이스의 비디오 기능을 제공합니다. (ID3D11VideoContext3)
ID3D11VideoDevice2 Microsoft Direct3D 11 디바이스의 비디오 디코딩 및 비디오 처리 기능을 제공합니다.
ID3D11View 뷰 인터페이스는 렌더링 중에 파이프라인이 액세스할 수 있는 리소스 부분을 지정합니다. (ID3D11View)
ID3DDestructionNotifier **ID3DDestructionNotifier** 는 Direct3D nano-COM 개체가 제거될 때 콜백을 등록하는 데 사용할 수 있는 인터페이스입니다.
ID3DDeviceContextState ID3DDeviceContextState 인터페이스는 Microsoft Direct3D 디바이스에 대한 상태 및 동작 정보를 포함하는 컨텍스트 상태 개체를 나타냅니다.

ID3DInclude ID3DInclude는 애플리케이션이 셰이더를 열고 닫기 위해 사용자 재정의 가능한 메서드를 호출할 수 있도록 사용자가 구현하는 포함 인터페이스입니다.
ID3DUserDefinedAnnotation ID3DUserDefinedAnnotation 인터페이스를 사용하면 애플리케이션이 애플리케이션의 코드 흐름 내에서 개념적 섹션 및 표식을 설명할 수 있습니다.
ID3DX11FFT 앞으로 및 역FFT를 캡슐화합니다.
ID3DX11Scan 컨텍스트를 검사합니다.
ID3DX11SegmentedScan 분할된 검사 컨텍스트입니다.

구조체

CD3D11_BLEND_DESC 혼합 상태 구조를 나타내며 혼합 상태 구조를 만들기 위한 편리한 메서드를 제공합니다.
CD3D11_BLEND_DESC1 ID3D11Device1::CreateBlendState1을 호출하여 혼합 상태 개체를 만드는 데 사용하는 혼합 상태를 설명합니다. D
CD3D11_BOX 상자를 나타내고 상자를 만들기 위한 편리한 메서드를 제공합니다.
CD3D11_BUFFER_DESC 버퍼를 나타내며 버퍼를 만들기 위한 편리한 메서드를 제공합니다.
CD3D11_COUNTER_DESC 카운터를 나타내며 카운터를 만들기 위한 편리한 메서드를 제공합니다.

CD3D11_DEPTH_STENCIL_DESC 깊이 스텐실 상태 구조를 나타내며 깊이 스텐실 상태 구조를 만들기 위한 편리한 메서드를 제공합니다.
CD3D11_DEPTH_STENCIL_VIEW_DESC 깊이 스텐실 보기를 나타내며 깊이 스텐실 뷰를 만들기 위한 편리한 방법을 제공합니다.
CD3D11_QUERY_DESC 쿼리를 나타내며 쿼리를 만들기 위한 편리한 메서드를 제공합니다.
CD3D11_QUERY_DESC1 쿼리를 설명합니다. D
CD3D11_RASTERIZER_DESC 래스터라이저 상태 구조를 나타내며 래스터라이저 상태 구조를 만들기 위한 편리한 메서드를 제공합니다.
CD3D11_RASTERIZER_DESC1 CD3D11_RASTERIZER_DESC1(d3d11_1.h) 구조체는 래스터라이저 상태를 설명합니다.
CD3D11_RASTERIZER_DESC2 CD3D11_RASTERIZER_DESC2(d3d11_3.h) 구조체는 래스터라이저 상태를 설명합니다.
CD3D11_RECT 사각형을 나타내며 사각형을 만들기 위한 편리한 메서드를 제공합니다.
CD3D11_RENDER_TARGET_VIEW_DESC 렌더링 대상 뷰를 나타내며 렌더링 대상 뷰를 만들기 위한 편리한 메서드를 제공합니다.
CD3D11_RENDER_TARGET_VIEW_DESC1 렌더링 대상 뷰를 사용하여 액세스할 수 있는 리소스의 하위 리소스에 대해 설명합니다. D
CD3D11_SAMPLER_DESC 샘플러 상태를 나타내며 샘플러 상태를 만들기 위한 편리한 메서드를 제공합니다.
CD3D11_SHADER_RESOURCE_VIEW_DESC 셰이더 리소스 뷰를 나타내며 셰이더 리소스 뷰를 만들기 위한 편리한 메서드를 제공합니다.

CD3D11_SHADER_RESOURCE_VIEW_DESC1 셰이더 리소스 뷰를 설명합니다. D
CD3D11_TEXTURE1D_DESC 1D 텍스처를 나타내며 1D 텍스처를 만들기 위한 편리한 방법을 제공합니다.
CD3D11_TEXTURE2D_DESC 2D 텍스처를 나타내며 2D 텍스처를 만들기 위한 편리한 방법을 제공합니다.
CD3D11_TEXTURE2D_DESC1 CD3D11_TEXTURE2D_DESC1(d3d11_3.h) 구조체는 2D 텍스처를 설명합니다.
CD3D11_TEXTURE3D_DESC 3D 텍스처를 나타내며 3D 텍스처를 만들기 위한 편리한 방법을 제공합니다.
CD3D11_TEXTURE3D_DESC1 CD3D11_TEXTURE3D_DESC1(d3d11_3.h) 구조체는 3D 텍스처를 설명합니다.
CD3D11_UNORDERED_ACCESS_VIEW_DESC 순서가 지정되지 않은 액세스 보기를 나타내며 순서가 지정되지 않은 액세스 뷰를 만들기 위한 편리한 메서드를 제공합니다.
CD3D11_UNORDERED_ACCESS_VIEW_DESC1 순서가 지정되지 않은 액세스 보기를 사용하여 액세스할 수 있는 리소스의 하위 리소스에 대해 설명합니다. D
CD3D11_VIEWPORT 뷰포트를 나타내며 뷰포트를 만들기 위한 편리한 메서드를 제공합니다.
D3D_SHADER_MACRO 셰이더 매크로를 정의합니다.
D3D11_AUTHENTICATED_PROTECTION_FLAGS 비디오 콘텐츠에 대한 보호 수준을 지정합니다.

D3D11_BLEND_DESC ID3D11Device::CreateBlendState를 호출하여 혼합 상태 개체를 만드는 데 사용하는 혼합 상태를 설명합니다.
D3D11_BLEND_DESC1 ID3D11Device1::CreateBlendState1을 호출하여 혼합 상태 개체를 만드는 데 사용하는 혼합 상태를 설명합니다. (D3D11_BLEND_DESC1)
D3D11_BOX 3D 상자를 정의합니다. (D3D11_BOX)
D3D11_BUFFER_DESC 버퍼 리소스에 대해 설명합니다. (D3D11_BUFFER_DESC)
D3D11_BUFFER_RTV 렌더링 대상 뷰에서 사용할 버퍼 리소스의 요소를 지정합니다.
D3D11_BUFFER_SRV 셰이더 리소스 뷰에서 사용할 버퍼 리소스의 요소를 지정합니다. (D3D11_BUFFER_SRV)
D3D11_BUFFER_UAV 순서가 지정되지 않은 액세스 뷰에서 사용할 버퍼의 요소에 대해 설명합니다. (D3D11_BUFFER_UAV)
D3D11_BUFFEREX_SRV 셰이더 리소스 뷰에서 사용할 원시 버퍼 리소스의 요소를 설명합니다.
D3D11_CLASS_INSTANCE_DESC HLSL 클래스 인스턴스에 대해 설명합니다.
D3D11_COMPUTE_SHADER_TRACE_DESC 추적할 컴퓨팅 셰이더의 인스턴스를 설명합니다.
D3D11_COUNTER_DESC 카운터를 설명합니다. (D3D11_COUNTER_DESC)

D3D11_COUNTER_INFO 비디오 카드의 성능 카운터 기능에 대한 정보입니다. (D3D11_COUNTER_INFO)
D3D11_DEPTH_STENCIL_DESC 깊이 스텐실 상태를 설명합니다. (D3D11_DEPTH_STENCIL_DESC)
D3D11_DEPTH_STENCIL_VIEW_DESC 깊이 스텐실 보기에서 액세스할 수 있는 텍스처의 하위 리소스를 지정합니다.
D3D11_DEPTH_STENCILOP_DESC 스텐실 테스트의 결과에 따라 수행할 수 있는 스텐실 작업입니다.
D3D11_DOMAIN_SHADER_TRACE_DESC 추적할 도메인 셰이더의 인스턴스를 설명합니다.
D3D11_DRAW_INDEXED_INSTANCED_INDIRECT_ARGS 인덱싱된 인스턴스 간접 그리기 인수입니다.
D3D11_DRAW_INSTANCED_INDIRECT_ARGS 그리기 인스턴스 간접 인수입니다.
D3D11_FEATURE_DATA_ARCHITECTURE_INFO Direct3D 11.1 어댑터 아키텍처에 대한 정보를 설명합니다.
D3D11_FEATURE_DATA_D3D10_X_HARDWARE_OPTIONS 현재 그래픽 드라이버의 컴퓨팅 셰이더 및 원시 및 구조화된 버퍼 지원에 대해 설명합니다.
D3D11_FEATURE_DATA_D3D11_OPTIONS 현재 그래픽 드라이버의 Direct3D 11.1 기능 옵션에 대해 설명합니다.
D3D11_FEATURE_DATA_D3D11_OPTIONS1 현재 그래픽 드라이버의 Direct3D 11.2 기능 옵션에 대해 설명합니다.
D3D11_FEATURE_DATA_D3D11_OPTIONS2 현재 그래픽 드라이버의 Direct3D 11.3 기능 옵션에 대해 설명합니다. (D3D11_FEATURE_DATA_D3D11_OPTIONS2)

D3D11_FEATURE_DATA_D3D11_OPTIONS3 현재 그래픽 드라이버의 Direct3D 11.3 기능 옵션에 대해 설명합니다. (D3D11_FEATURE_DATA_D3D11_OPTIONS3)
D3D11_FEATURE_DATA_D3D11_OPTIONS4 현재 그래픽 드라이버의 Direct3D 11.4 기능 옵션에 대해 설명합니다.
D3D11_FEATURE_DATA_D3D11_OPTIONS5 현재 그래픽 드라이버의 공유 리소스에 대한 지원 수준을 설명합니다.
D3D11_FEATURE_DATA_D3D9_OPTIONS 현재 그래픽 드라이버의 Direct3D 9 기능 옵션에 대해 설명합니다. (D3D11_FEATURE_DATA_D3D9_OPTIONS)
D3D11_FEATURE_DATA_D3D9_OPTIONS1 현재 그래픽 드라이버의 Direct3D 9 기능 옵션에 대해 설명합니다. (D3D11_FEATURE_DATA_D3D9_OPTIONS1)
D3D11_FEATURE_DATA_D3D9_SHADOW_SUPPORT 현재 그래픽 드라이버의 Direct3D 9 섀도 지원에 대해 설명합니다.
D3D11_FEATURE_DATA_D3D9_SIMPLE_INSTANCING_SUPPORT 간단한 인스턴싱이 지원되는지 여부를 설명합니다.
D3D11_FEATURE_DATA_DISPLAYABLE 현재 그래픽 드라이버에서 지원되는 표시 가능한 표면의 수준을 설명합니다.
D3D11_FEATURE_DATA_DOUBLES 현재 그래픽 드라이버의 이중 데이터 형식 지원에 대해 설명합니다.
D3D11_FEATURE_DATA_FORMAT_SUPPORT 지정된 형식에 대해 현재 그래픽 드라이버에서 지원하는 리소스를 설명합니다. (D3D11_FEATURE_DATA_FORMAT_SUPPORT)
D3D11_FEATURE_DATA_FORMAT_SUPPORT2 지정된 형식에 대해 현재 그래픽 드라이버에서 지원하는 순서가 지정되지 않은 리소스 옵션을 설명합니다.

D3D11_FEATURE_DATA_GPU_VIRTUAL_ADDRESS_SUPPORT 리소스당 최대 주소 비트 및 프로세스당 최대 주소 비트를 포함하여 기능 데이터 GPU 가상 주소 지원에 대해 설명합니다.
D3D11_FEATURE_DATA_MARKER_SUPPORT GPU 프로파일링 기술이 지원되는지 여부를 설명합니다.
D3D11_FEATURE_DATA_SHADER_CACHE 현재 그래픽 드라이버에서 지원되는 셰이더 캐싱 수준을 설명합니다. (D3D11_FEATURE_DATA_SHADER_CACHE)
D3D11_FEATURE_DATA_SHADER_MIN_PRECISION_SUPPORT 현재 그래픽 드라이버의 셰이더에 대한 정밀도 지원 옵션을 설명합니다.
D3D11_FEATURE_DATA_THREADING 현재 그래픽 드라이버에서 지원하는 다중 스레딩 기능에 대해 설명합니다.
D3D11_FEATURE_DATA_VIDEO_DECODER_HISTOGRAM 지정된 기능이 D3D11_FEATURE_VIDEO_DECODER_HISTOGRAM 때 ID3D11VideoDevice2::CheckFeatureSupport 호출에 대한 데이터를 제공합니다.
D3D11_FUNCTION_DESC 함수를 설명합니다. (D3D11_FUNCTION_DESC)
D3D11_GEOMETRY_SHADER_TRACE_DESC 추적할 기하 도형 셰이더의 인스턴스를 설명합니다.
D3D11_HULL_SHADER_TRACE_DESC 추적할 헐 셰이더의 인스턴스를 설명합니다.
D3D11_INFO_QUEUE_FILTER 디버그 메시지 필터; 에는 허용하거나 거부할 메시지 형식 목록이 포함되어 있습니다. (D3D11_INFO_QUEUE_FILTER)
D3D11_INFO_QUEUE_FILTER_DESC 필터를 통과하는 특정 유형의 메시지를 허용하거나 거부합니다. (D3D11_INFO_QUEUE_FILTER_DESC)

D3D11_INPUT_ELEMENT_DESC 입력 어셈블러 단계의 단일 요소에 대한 설명입니다. (D3D11_INPUT_ELEMENT_DESC)
D3D11_LIBRARY_DESC 라이브러리에 대해 설명합니다. (D3D11_LIBRARY_DESC)
D3D11_MAPPED_SUBRESOURCE 하위 리소스 데이터에 대한 액세스를 제공합니다.
D3D11_MESSAGE 정보 큐의 디버그 메시지입니다. (D3D11_MESSAGE)
D3D11_PACKED_MIP_DESC 밉맵이 있는 타일식 리소스의 타일 구조에 대해 설명합니다. (D3D11_PACKED_MIP_DESC)
D3D11_PARAMETER_DESC 함수 매개 변수를 설명합니다. (D3D11_PARAMETER_DESC)
D3D11_PIXEL_SHADER_TRACE_DESC 추적할 픽셀 셰이더의 인스턴스를 설명합니다.
D3D11_QUERY_DATA_PIPELINE_STATISTICS ID3D11DeviceContext::Begin 및 ID3D11DeviceContext::End 호출 간의 그래픽 파이프라인 활동에 대한 정보를 쿼리합니다.
D3D11_QUERY_DATA_SO_STATISTICS ID3D11DeviceContext::Begin 및 ID3D11DeviceContext::End 사이에 스트림 출력 버퍼로 스트리밍되는 데이터의 양에 대한 정보를 쿼리합니다.
D3D11_QUERY_DATA_TIMESTAMP_DISJOINT 타임스탬프 쿼리의 안정성에 대한 쿼리 정보입니다. (D3D11_QUERY_DATA_TIMESTAMP_DISJOINT)
D3D11_QUERY_DESC 쿼리를 설명합니다. (D3D11_QUERY_DESC)
D3D11_QUERY_DESC1 쿼리를 설명합니다. (D3D11_QUERY_DESC1)

D3D11_RASTERIZER_DESC 래스터라이저 상태를 설명합니다. (D3D11_RASTERIZER_DESC)
D3D11_RASTERIZER_DESC1 래스터라이저 상태를 설명합니다. (D3D11_RASTERIZER_DESC1)
D3D11_RASTERIZER_DESC2 래스터라이저 상태를 설명합니다. (D3D11_RASTERIZER_DESC2)
D3D11_RENDER_TARGET_BLEND_DESC 렌더링 대상의 혼합 상태를 설명합니다. (D3D11_RENDER_TARGET_BLEND_DESC)
D3D11_RENDER_TARGET_BLEND_DESC1 렌더링 대상의 혼합 상태를 설명합니다. (D3D11_RENDER_TARGET_BLEND_DESC1)
D3D11_RENDER_TARGET_VIEW_DESC 렌더링 대상 뷰를 사용하여 액세스할 수 있는 리소스의 하위 리소스를 지정합니다.
D3D11_RENDER_TARGET_VIEW_DESC1 렌더링 대상 뷰를 사용하여 액세스할 수 있는 리소스의 하위 리소스에 대해 설명합니다. (D3D11_RENDER_TARGET_VIEW_DESC1)
D3D11_SAMPLER_DESC 샘플러 상태를 설명합니다. (D3D11_SAMPLER_DESC)
D3D11_SHADER_BUFFER_DESC 셰이더 상수 버퍼를 설명합니다. (D3D11_SHADER_BUFFER_DESC)
D3D11_SHADER_DESC 셰이더에 대해 설명합니다. (D3D11_SHADER_DESC)
D3D11_SHADER_INPUT_BIND_DESC 셰이더 리소스가 셰이더 입력에 바인딩되는 방법을 설명합니다. (D3D11_SHADER_INPUT_BIND_DESC)
D3D11_SHADER_RESOURCE_VIEW_DESC 셰이더 리소스 뷰에 대해 설명합니다. (D3D11_SHADER_RESOURCE_VIEW_DESC)

D3D11_SHADER_RESOURCE_VIEW_DESC1 셰이더 리소스 뷰에 대해 설명합니다. (D3D11_SHADER_RESOURCE_VIEW_DESC1)
D3D11_SHADER_TRACE_DESC 셰이더 추적 개체에 대해 설명합니다.
D3D11_SHADER_TYPE_DESC 셰이더 변수 형식에 대해 설명합니다. (D3D11_SHADER_TYPE_DESC)
D3D11_SHADER_VARIABLE_DESC 셰이더 변수에 대해 설명합니다. (D3D11_SHADER_VARIABLE_DESC)
D3D11_SIGNATURE_PARAMETER_DESC 셰이더 서명을 설명합니다. (D3D11_SIGNATURE_PARAMETER_DESC)
D3D11_SO_DECLARATION_ENTRY 출력 슬롯의 꼭짓점 버퍼에 있는 꼭짓점 요소에 대한 설명입니다. (D3D11_SO_DECLARATION_ENTRY)
D3D11_SUBRESOURCE_DATA 하위 리소스를 초기화하기 위한 데이터를 지정합니다. (D3D11_SUBRESOURCE_DATA)
D3D11_SUBRESOURCE_TILING 타일형 하위 리소스 볼륨에 대해 설명합니다. (D3D11_SUBRESOURCE_TILING)
D3D11_TEX1D_ARRAY_DSV 깊이 스텐실 보기에서 사용할 1D 텍스처 배열의 하위 리소스를 지정합니다.
D3D11_TEX1D_ARRAY_RTV 렌더링 대상 보기에서 사용할 1D 텍스처 배열의 하위 리소스를 지정합니다.
D3D11_TEX1D_ARRAY_SRV 셰이더 리소스 뷰에서 사용할 1D 텍스처 배열의 하위 리소스를 지정합니다.
D3D11_TEX1D_ARRAY_UAV 순서가 지정되지 않은 액세스 1D 텍스처 리소스의 배열에 대해 설명합니다. (D3D11_TEX1D_ARRAY_UAV)

D3D11_TEX1D_DSV 깊이 스텐실 보기에 액세스할 수 있는 1D 텍스처의 하위 리소스를 지정합니다. (D3D11_TEX1D_DSV)
D3D11_TEX1D_RTV 렌더링 대상 보기에서 사용할 1D 텍스처의 하위 리소스를 지정합니다. (D3D11_TEX1D_RTV)
D3D11_TEX1D_SRV 셰이더 리소스 뷰에서 사용할 1D 텍스처의 하위 리소스를 지정합니다. (D3D11_TEX1D_SRV)
D3D11_TEX1D_UAV 순서가 지정되지 않은 액세스 1D 텍스처 리소스에 대해 설명합니다. (D3D11_TEX1D_UAV)
D3D11_TEX2D_ARRAY_DSV 깊이 스텐실 보기에 액세스할 수 있는 배열 2D 텍스처의 하위 리소스를 지정합니다.
D3D11_TEX2D_ARRAY_RTV 렌더링 대상 보기에서 사용할 2D 텍스처 배열의 하위 리소스를 지정합니다.
D3D11_TEX2D_ARRAY_RTV1 렌더링 대상 보기에서 사용할 2D 텍스처 배열의 하위 리소스에 대해 설명합니다. (D3D11_TEX2D_ARRAY_RTV1)
D3D11_TEX2D_ARRAY_SRV 셰이더 리소스 뷰에서 사용할 2D 텍스처 배열의 하위 리소스를 지정합니다.
D3D11_TEX2D_ARRAY_SRV1 셰이더 리소스 뷰에서 사용할 2D 텍스처 배열의 하위 리소스에 대해 설명합니다. (D3D11_TEX2D_ARRAY_SRV1)
D3D11_TEX2D_ARRAY_UAV 순서가 지정되지 않은 액세스 2D 텍스처 리소스의 배열을 설명합니다. (D3D11_TEX2D_ARRAY_UAV)
D3D11_TEX2D_ARRAY_UAV1 순서가 지정되지 않은 액세스 2D 텍스처 리소스의 배열을 설명합니다. (D3D11_TEX2D_ARRAY_UAV1)

D3D11_TEX2D_DSV 깊이 스텐실 보기에 액세스할 수 있는 2D 텍스처의 하위 리소스를 지정합니다. (D3D11_TEX2D_DSV)
D3D11_TEX2D_RTV 렌더링 대상 보기에서 사용할 2D 텍스처의 하위 리소스를 지정합니다. (D3D11_TEX2D_RTV)
D3D11_TEX2D_RTV1 렌더링 대상 보기에서 사용할 2D 텍스처의 하위 리소스에 대해 설명합니다. (D3D11_TEX2D_RTV1)
D3D11_TEX2D_SRV 셰이더 리소스 뷰에서 사용할 2D 텍스처의 하위 리소스를 지정합니다. (D3D11_TEX2D_SRV)
D3D11_TEX2D_SRV1 셰이더 리소스 뷰에서 사용할 2D 텍스처의 하위 리소스에 대해 설명합니다. (D3D11_TEX2D_SRV1)
D3D11_TEX2D_UAV 순서가 지정되지 않은 액세스 2D 텍스처 리소스에 대해 설명합니다. (D3D11_TEX2D_UAV)
D3D11_TEX2D_UAV1 순서가 지정되지 않은 액세스 2D 텍스처 리소스에 대해 설명합니다. (D3D11_TEX2D_UAV1)
D3D11_TEX2DMS_ARRAY_DSV 깊이 스텐실 보기에 대한 다중 샘플링된 2D 텍스처 배열의 하위 리소스를 지정합니다.
D3D11_TEX2DMS_ARRAY_RTV 렌더링 대상 보기에서 사용할 다중 샘플링된 2D 텍스처 배열의 하위 리소스를 지정합니다.
D3D11_TEX2DMS_ARRAY_SRV 셰이더 리소스 뷰에서 사용할 다중 샘플링된 2D 텍스처 배열의 하위 리소스를 지정합니다.
D3D11_TEX2DMS_DSV 깊이 스텐실 보기에 액세스할 수 있는 다중 샘플링된 2D 텍스처의 하위 리소스를 지정합니다. (D3D11_TEX2DMS_DSV)

D3D11_TEX2DMS_RTV 렌더링 대상 보기에서 사용할 다중 샘플링된 2D 텍스처의 하위 리소스를 지정합니다. (D3D11_TEX2DMS_RTV)
D3D11_TEX2DMS_SRV 셰이더 리소스 뷰에서 사용할 다중 샘플링된 2D 텍스처의 하위 리소스를 지정합니다.
D3D11_TEX3D_RTV 렌더링 대상 보기에서 사용할 3D 텍스처의 하위 리소스를 지정합니다.
D3D11_TEX3D_SRV 셰이더 리소스 뷰에서 사용할 3D 텍스처의 하위 리소스를 지정합니다.
D3D11_TEX3D_UAV 순서가 지정되지 않은 액세스 3D 텍스처 리소스에 대해 설명합니다. (D3D11_TEX3D_UAV)
D3D11_TEXCUBE_ARRAY_SRV 셰이더 리소스 뷰에서 사용할 큐브 텍스처 배열의 하위 리소스를 지정합니다.
D3D11_TEXCUBE_SRV 셰이더 리소스 뷰에서 사용할 큐브 텍스처의 하위 리소스를 지정합니다. (D3D11_TEXCUBE_SRV)
D3D11_TEXTURE1D_DESC 1D 텍스처를 설명합니다. (D3D11_TEXTURE1D_DESC)
D3D11_TEXTURE2D_DESC 2D 텍스처를 설명합니다. (D3D11_TEXTURE2D_DESC)
D3D11_TEXTURE2D_DESC1 2D 텍스처를 설명합니다. (D3D11_TEXTURE2D_DESC1)
D3D11_TEXTURE3D_DESC 3D 텍스처를 설명합니다. (D3D11_TEXTURE3D_DESC)
D3D11_TEXTURE3D_DESC1 3D 텍스처를 설명합니다. (D3D11_TEXTURE3D_DESC1)

D3D11_TILE_REGION_SIZE 타일형 영역의 크기를 설명합니다. (D3D11_TILE_REGION_SIZE)
D3D11_TILE_SHAPE 크기를 지정하여 타일의 모양을 설명합니다. (D3D11_TILE_SHAPE)
D3D11_TILED_RESOURCE_COORDINATE 타일형 리소스의 좌표를 설명합니다. (D3D11_TILED_RESOURCE_COORDINATE)
D3D11_TRACE_REGISTER 추적 레지스터에 대해 설명합니다.
D3D11_TRACE_STATS 추적에 대한 통계를 지정합니다.
D3D11_TRACE_STEP 명령인 추적 단계를 설명합니다.
D3D11_TRACE_VALUE 추적 값을 설명합니다.
D3D11_UNORDERED_ACCESS_VIEW_DESC 순서가 지정되지 않은 액세스 보기를 사용하여 액세스할 수 있는 리소스의 하위 리소스를 지정합니다.
D3D11_UNORDERED_ACCESS_VIEW_DESC1 순서가 지정되지 않은 액세스 보기를 사용하여 액세스할 수 있는 리소스의 하위 리소스에 대해 설명합니다. (D3D11_UNORDERED_ACCESS_VIEW_DESC1)
D3D11_VERTEX_SHADER_TRACE_DESC 추적할 꼭짓점 셰이더의 인스턴스를 설명합니다.
D3D11_VIEWPORT 뷰포트의 차원을 정의합니다. (D3D11_VIEWPORT)
D3DX11_FFT_BUFFER_INFO FFT에 대한 버퍼 요구 사항을 설명합니다.

D3DX11_FFT_DESC FFT에 대해 설명합니다.

Direct3D 12 그래픽

아티클2023. 03. 09.

Direct3D 12 그래픽 기술 개요입니다.

Direct3D 12 그래픽을 개발하려면 다음 헤더가 필요합니다.

- d3d11on12.h
- d3d12.h
- d3d12sdklayers.h
- d3d12shader.h
- windows.graphics.holographic.interop.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Direct3D 12 그래픽

열거형

D3D_ROOT_SIGNATURE_VERSION 루트 서명 레이아웃의 버전을 지정합니다.
D3D_SHADER_MODEL 셰이더 모델을 지정합니다.
D3D12_AUTO_BREADCRUMB_OP 렌더링/컴퓨팅 GPU 작업을 지정하는 상수를 정의합니다. (D3D12_AUTO_BREADCRUMB_OP)
D3D12_AXIS_SHADING_RATE 가로 또는 세로 축을 따라 음영 속도(가변 속도 음영 또는 VRS의 경우)를 지정하는 상수를 정의합니다.
D3D12_BACKGROUND_PROCESSING_MODE 이후에 제출되는 GPU 작업에 적용할 동적 최적화 수준을 지정하는 상수를 정의합니다.
D3D12_BLEND 픽셀 셰이더 및 렌더링 대상의 값을 변조하는 혼합 요소를 지정합니다.

D3D12_BLEND_OP RGB 또는 알파 혼합 작업을 지정합니다.
D3D12_BUFFER_SRV_FLAGS 버퍼 리소스를 보는 방법을 식별합니다. (D3D12_BUFFER_SRV_FLAGS)
D3D12_BUFFER_UAV_FLAGS 버퍼 리소스에 대한 순서가 지정되지 않은 액세스 보기 옵션을 식별합니다. (D3D12_BUFFER_UAV_FLAGS)
D3D12_CLEAR_FLAGS 깊이 스텐실 보기에서 지울 내용을 지정합니다.
D3D12_COLOR_WRITE_ENABLE 혼합 중에 쓸 수 있는 렌더링 대상의 각 픽셀에 대한 구성 요소를 식별합니다.
D3D12_COMMAND_LIST_FLAGS D3D12_COMMAND_LIST_FLAGS 열거형은 명령 목록을 만들 때 사용할 플래그를 지정합니다.
D3D12_COMMAND_LIST_SUPPORT_FLAGS 다양한 작업을 지원할 수 있는 명령 목록의 종류를 결정하는 데 사용됩니다.
D3D12_COMMAND_LIST_TYPE 명령 목록의 형식을 지정합니다.
D3D12_COMMAND_QUEUE_FLAGS 명령 큐를 만들 때 사용할 플래그를 지정합니다.
D3D12_COMMAND_QUEUE_PRIORITY 명령 큐의 우선 순위 수준을 정의합니다.
D3D12_COMPARISON_FUNC 비교 옵션을 지정합니다.
D3D12_CONSERVATIVE_RASTERIZATION_MODE 보수적인 래스터화가 켜지거나 꺼져 있는지 여부를 식별합니다. (D3D12_CONSERVATIVE_RASTERIZATION_MODE)

D3D12_CONSERVATIVE_RASTERIZATION_TIER 보수적 래스터화의 계층 수준을 식별합니다.
D3D12_CPU_PAGE_PROPERTY 힙의 CPU 페이지 속성을 지정합니다.
D3D12_CROSS_NODE_SHARING_TIER 계층 1 에뮬레이트, 계층 1 또는 계층 2와 같은 어댑터의 노드 간에 공유 수준을 지정합니다.
D3D12_CULL_MODE 특정 방향을 향한 삼각형이 그려지지 않도록 지정합니다.
D3D12_DEBUG_COMMAND_LIST_PARAMETER_TYPE ID3D12DebugCommandList1::SetDebugParameter 및 ID3D12DebugCommandList1::GetDebugParameter에서 사용하는 디버그 매개 변수 형식을 나타냅니다.
D3D12_DEBUG_DEVICE_PARAMETER_TYPE ID3D12DebugDevice1::SetDebugParameter 및 ID3D12DebugDevice1::GetDebugParameter의 pData 매개 변수가 가리키는 메모리의 데이터 형식을 지정합니다.
D3D12_DEBUG_FEATURE 선택적 D3D12 디버그 계층 기능에 대한 플래그입니다.
D3D12_DEPTH_WRITE_MASK 깊이 데이터를 쓰기 위한 깊이 스텐실 버퍼의 부분을 식별합니다.
D3D12_DESCRIPTOR_HEAP_FLAGS 힙에 대한 옵션을 지정합니다.
D3D12_DESCRIPTOR_HEAP_TYPE 설명자 힙의 형식을 지정합니다.
D3D12_DESCRIPTOR_RANGE_FLAGS 일부 드라이버 최적화를 사용하도록 설정할 수 있는 루트 서명 1.1 설명에서 참조하는 설명자와 데이터의 변동성을 지정합니다.

D3D12_DESCRIPTOR_RANGE_TYPE 예를 들어 설명자 테이블의 일부에 100개의 SRV(셰이더 리소스 뷰)가 있는 경우 범위를 100이 아닌 하나의 항목으로 선언할 수 있도록 범위를 지정합니다.
D3D12_DRED_ALLOCATION_TYPE 3D12DDI_HANDLETYPE 열거형 값과 일치합니다.
D3D12_DRED_ENABLEMENT 렌더링/컴퓨팅 GPU 작업을 지정하는 상수를 정의합니다. (D3D12_DRED_ENABLEMENT)
D3D12_DRED_FLAGS D3D12_DEVICE_REMOVED_EXTENDED_DATA 구조에서 Direct3D 런타임에 대한 컨트롤 플래그를 지정하는 데 사용되는 상수를 정의합니다.
D3D12_DRED_VERSION D3D12_VERSIONED_DEVICE_REMOVED_EXTENDED_DATA 구조에서 사용하는 DRED(Device Removed Extended Data) 버전을 지정하는 상수를 정의합니다.
D3D12_DRIVER_MATCHING_IDENTIFIER_STATUS 직렬화된 데이터가 현재 디바이스 및 드라이버 버전과 호환되는지 여부를 쿼리하는 ID3D12Device5::CheckDriverMatchingIdentifier 호출의 결과를 지정합니다.
D3D12_DSV_DIMENSION 깊이 스텐실 보기에서 사용되는 리소스에 액세스하는 방법을 지정합니다. (D3D12_DSV_DIMENSION)
D3D12_DSV_FLAGS 깊이 스텐실 보기 옵션을 지정합니다.
D3D12_ELEMENTS_LAYOUT 요소의 위치를 식별하는 방법을 설명합니다.
D3D12_EXPORT_FLAGS 상태 하위 개체에서 기호를 내보낼 때 적용할 플래그입니다.
D3D12_FEATURE 쿼리할 Direct3D 12 기능 또는 기능 집합을 지정하는 상수를 정의합니다.

D3D12_FENCE_FLAGS 펜스 옵션을 지정합니다. (D3D12_FENCE_FLAGS)
D3D12_FILL_MODE 삼각형을 렌더링할 때 사용할 채우기 모드를 지정합니다.
D3D12_FILTER 텍스처 샘플링 중 필터링 옵션을 지정합니다.
D3D12_FILTER_REDUCTION_TYPE 필터 감소 유형을 지정합니다.
D3D12_FILTER_TYPE 배율 또는 축소 샘플러 필터의 유형을 지정합니다.
D3D12_FORMAT_SUPPORT1 제공된 형식에 대해 지원되는 리소스를 지정합니다.
D3D12_FORMAT_SUPPORT2 제공된 형식에 대해 지원되는 순서가 지정되지 않은 리소스 옵션을 지정합니다.
D3D12_GPU_BASED_VALIDATION_FLAGS 런타임에 수행할 GPU 기반 유효성 검사 수준을 설명합니다.
D3D12_GPU_BASED_VALIDATION_PIPELINE_STATE_CREATE_FLAGS GPU-Based ID3D12Device::CreateGraphicsPipelineState 및 ID3D12Device::CreateComputePipelineState 중에 유효성 검사에서 패치된 파이프라인 상태를 처리하는 방법을 지정합니다.
D3D12_GPU_BASED_VALIDATION_SHADER_PATCH_MODE 디바이스 또는 명령 목록 수준에서 유효성 검사 GPU-Based 사용하는 셰이더 패치 유형을 지정합니다.
D3D12_GRAPHICS_STATES 그래픽 명령 목록과 관련된 상태를 지정하는 플래그를 정의합니다. 값은 비트 OR일 수 있습니다.

D3D12_HEAP_FLAGS 힙에 텍스처를 포함할 수 있는지 여부 및 리소스가 어댑터 간에 공유되는지 여부와 같은 힙 옵션을 지정합니다.
D3D12_HEAP_SERIALIZATION_TIER 힙 serialization 지원을 지정하는 상수를 정의합니다.
D3D12_HEAP_TYPE 힙의 형식을 지정합니다. 상주하는 경우 힙은 특정 CPU 캐시 속성이 있는 특정 실제 메모리 풀에 상주합니다.
D3D12_HIT_GROUP_TYPE 광선 추적 적중 그룹 상태 하위 개체의 유형을 지정합니다. 이 열거형의 값을 D3D12_HIT_GROUP_DESC 구조체와 함께 사용합니다.
D3D12_INDEX_BUFFER_STRIP_CUT_VALUE 삼각형 스트립 기본 토폴로지를 사용하는 경우 꼭짓점 위치는 연속 삼각형 "스트립"의 꼭짓점으로 해석됩니다.
D3D12_INDIRECT_ARGUMENT_TYPE 간접 매개 변수의 형식을 지정합니다.
D3D12_INPUT_CLASSIFICATION 입력 슬롯에 포함된 데이터의 형식을 식별합니다.
D3D12_LIFETIME_STATE 수명 추적 개체의 수명 상태를 지정하는 상수를 정의합니다.
D3D12_LOGIC_OP 렌더링 대상에 대해 구성할 논리 작업을 지정합니다. (D3D12_LOGIC_OP)
D3D12_MEASUREMENTS_ACTION 이전 워크로드 계측의 결과로 수행해야 하는 작업을 지정하는 상수를 정의합니다.
D3D12_MEMORY_POOL 힙의 메모리 풀을 지정합니다.

D3D12_MESH_SHADER_TIER 메시 및 증폭 셰이더 지원을 지정하는 상수를 정의합니다.
D3D12_MESSAGE_CATEGORY 디버그 메시지의 범주를 지정합니다.
D3D12_MESSAGE_ID 정보 큐 필터를 설정하기 위한 디버그 메시지 ID를 지정합니다(D3D12_INFO_QUEUE_FILTER 참조). 이러한 메시지를 사용하여 메시지 범주가 스토리지 및 검색 필터를 통과하도록 허용하거나 거부합니다.
D3D12_MESSAGE_SEVERITY 정보 큐에 대한 메시지 심각도 수준을 디버그합니다. (D3D12_MESSAGE_SEVERITY)
D3D12_META_COMMAND_PARAMETER_FLAGS 메타 명령에 대한 매개 변수의 플래그를 지정하는 상수를 정의합니다. 값은 비트 OR일 수 있습니다.
D3D12_META_COMMAND_PARAMETER_STAGE 메타 명령에 대한 매개 변수의 단계를 지정하는 상수를 정의합니다.
D3D12_META_COMMAND_PARAMETER_TYPE 메타 명령에 대한 매개 변수의 데이터 형식을 지정하는 상수를 정의합니다.
D3D12_MULTIPLE_FENCE_WAIT_FLAGS 여러 펜스에 대해 여러 대기 플래그를 지정합니다.
D3D12_MULTISAMPLE_QUALITY_LEVEL_FLAGS 품질 수준을 결정하는 옵션을 지정합니다.
D3D12_PIPELINE_STATE_FLAGS 파이프라인 상태를 제어하는 플래그입니다.
D3D12_PIPELINE_STATE_SUBOBJECT_TYPE 파이프라인 상태 스트림 설명에서 하위 개체의 형식을 지정합니다.

D3D12_PREDICATION_OP 적용할 조건자 작업을 지정합니다.
D3D12_PRIMITIVE_TOPOLOGY_TYPE 파이프라인이 기하 도형 또는 헐 셰이더 입력 기본 형식을 해석하는 방법을 지정합니다.
D3D12_PROGRAMMABLE_SAMPLE_POSITIONS_TIER 어댑터에서 제공하는 프로그래밍 가능한 샘플 위치에 대한 지원 수준을 지정합니다.
D3D12_PROTECTED_RESOURCE_SESSION_FLAGS 보호된 리소스 세션 플래그를 지정하는 상수를 정의합니다.
D3D12_PROTECTED_RESOURCE_SESSION_SUPPORT_FLAGS 보호된 리소스 세션 지원을 지정하는 상수를 정의합니다.
D3D12_PROTECTED_SESSION_STATUS 보호된 세션 상태를 지정하는 상수를 정의합니다.
D3D12_QUERY_HEAP_TYPE 만들 쿼리 힙의 형식을 지정합니다.
D3D12_QUERY_TYPE 쿼리 유형을 지정합니다.
D3D12_RAY_FLAGS 투명도, 컬링 및 조기 아웃 동작을 재정의하기 위해 TraceRay 함수에 전달된 플래그입니다.
D3D12_RAYTRACING_ACCELERATION_STRUCTURE_BUILD_FLAGS 광선 추적 가속 구조의 빌드에 대한 플래그를 지정합니다. 가속 구조 빌드 작업에 대한 입력을 제공하는 D3D12_BUILD_RAYTRACING_ACCELERATION_STRUCTURE_INPUTS 구조와 함께 이 열거형의 값을 사용합니다.
D3D12_RAYTRACING_ACCELERATION_STRUCTURE_COPY_MODE CopyRaytracingAccelerationStructure를 호출할 때 수행되는 복사 작업의 유형을 지정합니다.

D3D12_RAYTRACING_ACCELERATION_STRUCTURE_POSTBUILD_INFO_TYPE EmitRaytracingAccelerationStructurePostbuildInfo 및 BuildRaytracingAccelerationStructure를 호출하여 검색할 수 있는 가속 구조 사후 빌드 정보의 유형을 지정합니다.
D3D12_RAYTRACING_ACCELERATION_STRUCTURE_TYPE 광선 추적 가속 구조체의 유형을 지정합니다.
D3D12_RAYTRACING_GEOMETRY_FLAGS D3D12_RAYTRACING_GEOMETRY_DESC 구조체에서 광선 추적 기하 도형에 대한 플래그를 지정합니다.
D3D12_RAYTRACING_GEOMETRY_TYPE 광선 추적에 사용되는 기하 도형의 유형을 지정합니다. 이 열거형의 값을 사용하여 D3D12_RAYTRACING_GEOMETRY_DESC 기하 도형 형식을 지정합니다.
D3D12_RAYTRACING_INSTANCE_FLAGS 광선 추적 가속 구조 인스턴스에 대한 플래그입니다. 이러한 플래그는 개별 인스턴스에 대한 D3D12_RAYTRACING_GEOMETRY_FLAGS 재정의하는 데 사용할 수 있습니다.
D3D12_RAYTRACING_PIPELINE_FLAGS 광선 추적 파이프라인에 대한 구성 플래그를 지정하는 상수를 정의합니다.
D3D12_RAYTRACING_TIER 그래픽 디바이스에서 광선 추적 지원 수준을 지정합니다.
D3D12_RENDER_PASS_BEGINNING_ACCESS_TYPE 렌더링 패스로 전환할 때 애플리케이션이 지정된 리소스에 부여되는 액세스 유형을 지정합니다.
D3D12_RENDER_PASS_ENDING_ACCESS_TYPE 렌더링 패스에서 전환할 때 애플리케이션이 지정된 리소스에 부여되는 액세스 유형을 지정합니다.
D3D12_RENDER_PASS_FLAGS 렌더링 패스의 특성을 지정합니다. 예를 들어 일시 중단 또는 다시 시작할 렌더링 패스인지 여부입니다.
D3D12_RENDER_PASS_TIER 그래픽 디바이스에서 렌더링 패스에 대한 지원 수준을 지정합니다.

D3D12_RESIDENCY_FLAGS EnqueuMakeResident 함수와 함께 사용하여 메모리 예산을 초과할 때 상주 작업이 진행되는 방식을 선택합니다.
D3D12_RESIDENCY_PRIORITY 애플리케이션 우선 순위 체계를 신속하게 설정하는 데 유용한 광범위한 상주 우선 순위 버킷을 지정합니다.
D3D12_RESOLVE_MODE 확인 작업을 지정합니다.
D3D12_RESOURCE_BARRIER_FLAGS 분할 리소스 장벽을 설정하기 위한 플래그입니다.
D3D12_RESOURCE_BARRIER_TYPE 리소스 장벽 유형(리소스 사용의 전환) 설명을 지정합니다.
D3D12_RESOURCE_BINDING_TIER 사용 중인 리소스 바인딩의 계층을 식별합니다.
D3D12_RESOURCE_DIMENSION 사용 중인 리소스의 유형을 식별합니다. (D3D12_RESOURCE_DIMENSION)
D3D12_RESOURCE_FLAGS 리소스 작업에 대한 옵션을 지정합니다.
D3D12_RESOURCE_HEAP_TIER 하드웨어 및 드라이버에서 지원하는 리소스 힙 계층을 지정합니다.
D3D12_RESOURCE_STATES 리소스 사용 방법에 대한 리소스의 상태를 지정하는 상수를 정의합니다.
D3D12_RLDO_FLAGS 라이브 디바이스 개체의 수명에 대해 보고할 정보의 양에 대한 옵션을 지정합니다.

D3D12_ROOT_DESCRIPTOR_FLAGS 일부 드라이버 최적화를 사용하도록 설정할 수 있는 루트 서명 1.1 설명에서 설명자가 참조하는 데이터의 변동성을 지정합니다.
D3D12_ROOT_PARAMETER_TYPE 루트 서명 슬롯의 형식을 지정합니다.
D3D12_ROOT_SIGNATURE_FLAGS 루트 서명 레이아웃에 대한 옵션을 지정합니다.
D3D12_RTV_DIMENSION 렌더링 대상으로 볼 리소스의 유형을 식별합니다.
D3D12_SAMPLER_FEEDBACK_TIER 샘플러 피드백 지원을 지정하는 상수를 정의합니다.
D3D12_SERIALIZED_DATA_TYPE 직렬화된 데이터의 형식을 지정합니다. ID3D12Device5::CheckDriverMatchingIdentifier를 호출할 때 이 열거형의 값을 사용합니다.
D3D12_SHADER_CACHE_CONTROL_FLAGS 셰이더 캐시 제어 옵션을 지정하는 상수를 정의합니다.
D3D12_SHADER_CACHE_FLAGS 셰이더 캐시 플래그를 지정하는 상수를 정의합니다.
D3D12_SHADER_CACHE_KIND_FLAGS 셰이더 캐시의 종류를 지정하는 상수를 정의합니다.
D3D12_SHADER_CACHE_MODE 셰이더 캐시 모드를 지정하는 상수를 정의합니다.
D3D12_SHADER_CACHE_SUPPORT_FLAGS 현재 그래픽 드라이버의 셰이더 캐싱 지원 수준을 설명합니다. (D3D12_SHADER_CACHE_SUPPORT_FLAGS)

D3D12_SHADER_COMPONENT_MAPPING 메모리가 SRV(셰이더 리소스 뷰)에 의해 라우팅되는 방법을 지정합니다.
D3D12_SHADER_MIN_PRECISION_SUPPORT 현재 그래픽 드라이버의 셰이더에 대한 최소 정밀도 지원 옵션을 설명합니다.
D3D12_SHADER_VERSION_TYPE Direct3D에서 인식하는 셰이더 유형을 열거합니다. D3D12_SHADER_DESC 구조체의 Version 멤버를 인코딩하는 데 사용됩니다.
D3D12_SHADER_VISIBILITY 지정된 루트 서명 슬롯의 내용에 액세스할 수 있는 셰이더를 지정합니다.
D3D12_SHADING_RATE 음영 속도를 지정하는 상수를 정의합니다(가변 속도 음영 또는 VRS의 경우).
D3D12_SHADING_RATE_COMBINER 음영 속도 결합자를 지정하는 상수를 정의합니다(가변 속도 음영 또는 VRS용).
D3D12_SHARED_RESOURCE_COMPATIBILITY_TIER API 간 공유 지원 계층을 지정하는 상수를 정의합니다.
D3D12_SRV_DIMENSION 셰이더 리소스로 볼 리소스의 유형을 식별합니다.
D3D12_STATE_OBJECT_FLAGS 상태 개체에 대한 제약 조건을 지정합니다. D3D12_STATE_OBJECT_CONFIG 구조체에서 이 열거형의 값을 사용합니다.
D3D12_STATE_OBJECT_TYPE 상태 개체의 형식을 지정합니다. D3D12_STATE_OBJECT_DESC 함께 사용합니다.
D3D12_STATE_SUBOBJECT_TYPE 상태 하위 개체의 형식입니다. D3D12_STATE_SUBOBJECT 함께 사용합니다.
D3D12_STATIC_BORDER_COLOR 정적 샘플러의 테두리 색을 지정합니다.

D3D12_STENCIL_OP 깊이 스텐실 테스트 중에 수행할 수 있는 스텐실 작업을 식별합니다.
D3D12_TEXTURE_ADDRESS_MODE 텍스처 경계 밖에 있는 텍스처 좌표를 확인하는 기술을 식별합니다.
D3D12_TEXTURE_COPY_TYPE 수행되는 텍스처 복사본 유형을 지정합니다.
D3D12_TEXTURE_LAYOUT 텍스처 레이아웃 옵션을 지정합니다. (D3D12_TEXTURE_LAYOUT)
D3D12_TILE_COPY_FLAGS 타일을 복사하는 방법을 지정합니다.
D3D12_TILE_MAPPING_FLAGS 타일 매핑 작업을 수행하는 방법을 지정합니다.
D3D12_TILE_RANGE_FLAGS 타일 매핑 범위를 지정합니다.
D3D12_TILED_RESOURCES_TIER 타일식 리소스가 지원되는 계층 수준을 식별합니다.
D3D12_TRI_STATE TBD
D3D12_UAV_DIMENSION 순서가 지정되지 않은 액세스 보기 옵션을 식별합니다.
D3D12_VARIABLE_SHADING_RATE_TIER 음영 속도 계층(가변 속도 음영 또는 VRS의 경우)을 지정하는 상수를 정의합니다.
D3D12_VIEW_INSTANCING_FLAGS 뷰 인스턴스화 옵션을 지정합니다.

D3D12_VIEW_INSTANCING_TIER 뷰 인스턴스화가 지원되는 계층 수준을 나타냅니다.
D3D12_WAVE_MMA_TIER waveMMA(wave_matrix) 작업에 대한 지원 수준을 지정하는 상수를 정의합니다.
D3D12_WRITEBUFFERIMMEDIATE_MODE WriteBufferImmediate 작업에 사용되는 모드를 지정합니다.

Functions

AcquireDirect3D12BufferResource IHolographicCameraInterop::AcquireDirect3D12BufferResource 함수는 Direct3D 12 버퍼 리소스를 획득합니다.
AcquireDirect3D12BufferResource IHolographicQuadLayerInterop::AcquireDirect3D12BufferResource 함수는 Direct3D 12 버퍼 리소스를 획득합니다.
AcquireDirect3D12BufferResourceWithTimeout IHolographicCameraInterop::AcquireDirect3D12BufferResourceWithTimeout 함수는 선택적 시간 제한과 함께 Direct3D 12 버퍼 리소스를 획득합니다.
AcquireDirect3D12BufferResourceWithTimeout IHolographicQuadLayerInterop::AcquireDirect3D12BufferResourceWithTimeout 함수는 선택적 시간 제한과 함께 Direct3D 12 버퍼 리소스를 획득합니다.
AcquireWrappedResources D3D 11on12에 사용할 D3D11 리소스를 획득합니다. 래핑된 리소스에 대한 렌더링을 다시 시작할 수 있음을 나타냅니다.
AddApplicationMessage 사용자 정의 메시지를 메시지 큐에 추가하고 해당 메시지를 전송하여 출력을 디버그합니다.

AddMessage 메시지 큐에 디버그 메시지를 추가하고 해당 메시지를 전송하여 출력을 디버그합니다.
AddRetrievalFilterEntries 검색 필터 스택의 맨 위에 스토리지 필터를 추가합니다. (ID3D12InfoQueue.AddRetrievalFilterEntries)
AddStorageFilterEntries 스토리지 필터 스택의 맨 위에 스토리지 필터를 추가합니다. (ID3D12InfoQueue.AddStorageFilterEntries)
AddToStateObject 기존 상태 개체에 증분 방식으로 추가합니다. 이렇게 하면 기존 개체의 상위 집합인 상태 개체를 처음부터 만드는 것보다 CPU 오버헤드가 낮아집니다.
AssertResourceState 리소스 또는 하위 리소스가 지정된 상태인지 여부를 확인합니다. (ID3D12DebugCommandList.AssertResourceState)
AssertResourceState 지정된 상태가 명령 목록을 기록하는 동안 지정된 하위 리소스의 상태를 알고 있다고 가정하여 하위 리소스의 상태와 일치하는지 확인합니다(예:
AssertResourceState 리소스 또는 하위 리소스가 지정된 상태인지 여부를 확인합니다. (ID3D12DebugCommandQueue.AssertResourceState)
AtomicCopyBufferUINT 선택적 종속 리소스와 함께 UINT 형식의 기본 데이터 요소를 한 리소스에서 다른 리소스로 원자적으로 복사합니다.
AtomicCopyBufferUINT64 선택적 종속 리소스와 함께 UINT64 형식의 기본 데이터 요소를 한 리소스에서 다른 리소스로 원자적으로 복사합니다.
장벽 그래픽 명령 목록 기록에 장벽 컬렉션을 추가합니다.

BeginEvent 직접 호출할 수 없습니다. PIX 이벤트 런타임을 사용하여 명령 큐에 이벤트를 삽입합니다. (ID3D12CommandQueue.BeginEvent)
BeginEvent 직접 호출할 수 없습니다. PIX 이벤트 런타임을 사용하여 명령 목록에 이벤트를 삽입합니다. (ID3D12GraphicsCommandList.BeginEvent)
BeginQuery 쿼리 실행을 시작합니다. (ID3D12GraphicsCommandList.BeginQuery)
BeginRenderPass 렌더링 패스 기간 동안 출력 리소스 집합을 바인딩하여 렌더링 패스의 시작을 표시합니다. 이러한 바인딩은 하나 이상의 RTV(렌더링 대상 뷰) 및/또는 DSV(깊이 스텐실 뷰)에 대한 것입니다.
BuildRaytracingAccelerationStructure GPU에서 광선 추적 가속 구조 빌드를 수행하고 필요에 따라 빌드 직후 빌드 후 정보를 출력합니다.
CheckDriverMatchingIdentifier 현재 디바이스/드라이버와 함께 모드 D3D12_RAYTRACING_ACCELERATION_STRUCTURE_COPY_MODE_SERIALIZE CopyRaytracingAccelerationStructure 호출로 인해 발생하는 직렬화된 광선 추적 가속 구조와 같은 직렬화된 데이터의 호환성을 보고합니다.
CheckFeatureSupport 현재 그래픽 드라이버에서 지원하는 기능에 대한 정보를 가져옵니다. (ID3D12Device.CheckFeatureSupport)
ClearDepthStencilView 깊이 스텐실 리소스를 지웁니다. (ID3D12GraphicsCommandList.ClearDepthStencilView)
ClearRenderTargetView 렌더링 대상의 모든 요소를 하나의 값으로 설정합니다.
ClearRetrievalFilter 검색 필터 스택의 맨 위에서 검색 필터를 제거합니다. (ID3D12InfoQueue.ClearRetrievalFilter)

ClearState 직접 명령 목록의 상태를 명령 목록을 만들 때의 상태로 다시 설정합니다. (ID3D12GraphicsCommandList.ClearState)
ClearStorageFilter 스토리지 필터 스택의 맨 위에서 스토리지 필터를 제거합니다. (ID3D12InfoQueue.ClearStorageFilter)
ClearStoredMessages 메시지 큐에서 모든 메시지를 지웁니다. (ID3D12InfoQueue.ClearStoredMessages)
ClearUnorderedAccessViewFloat 순서가 지정되지 않은 액세스 뷰의 모든 요소를 지정된 float 값으로 설정합니다.
ClearUnorderedAccessViewUint UAV(정렬되지 않은 액세스 뷰)의 모든 요소를 지정된 정수 값으로 설정합니다.
닫기 명령 목록에 대한 기록이 완료되었음을 나타냅니다. (ID3D12GraphicsCommandList.Close)
CommitDirect3D12Resource IHolographicCameraRenderingParametersInterop::CommitDirect3D12Resource 함수는 HolographicCamera 출력에 대한 프레젠테이션을 위해 Direct3D 12 버퍼를 커밋합니다.
CommitDirect3D12Resource 쿼드 계층이 연결된 HolographicCamera 와 연결된 출력에 대한 프레젠테이션을 위해 Direct3D 12 버퍼를 커밋합니다.
CommitDirect3D12ResourceWithDepthData IHolographicCameraRenderingParametersInterop::CommitDirect3D12ResourceWithDepthData 함수는 HolographicCamera 출력에 대한 Direct3D 12 버퍼를 커밋합니다.
CopyBufferRegion 버퍼의 영역을 한 리소스에서 다른 리소스로 복사합니다.
CopyDescriptors 원본에서 대상으로 설명자를 복사합니다. (ID3D12Device.CopyDescriptors)

CopyDescriptorsSimple 원본에서 대상으로 설명자를 복사합니다. (ID3D12Device.CopyDescriptorsSimple)
CopyRaytracingAccelerationStructure 지정된 변환을 적용하는 동안 원본 가속 구조를 대상 메모리에 복사합니다.
CopyResource 원본 리소스의 전체 내용을 대상 리소스에 복사합니다.
CopyTextureRegion 이 메서드는 GPU를 사용하여 두 위치 간에 텍스처 데이터를 복사합니다. 원본과 대상 모두 버퍼 리소스 또는 텍스처 리소스 내에 있는 텍스처 데이터를 참조할 수 있습니다.
CopyTileMappings 원본 예약 리소스에서 대상 예약 리소스로 매핑을 복사합니다.
CopyTiles 버퍼에서 타일 리소스로 또는 그 반대로 타일을 복사합니다. (ID3D12GraphicsCommandList.CopyTiles)
CreateCommandAllocator 명령 할당자 개체를 만듭니다.
CreateCommandList 명령 목록을 만듭니다.
CreateCommandList1 닫힌 상태에서 명령 목록을 만듭니다.
CreateCommandQueue 명령 큐를 만듭니다.
CreateCommandQueue1 작성자 ID를 사용하여 명령 큐를 만듭니다.
CreateCommandSignature 이 메서드는 명령 서명을 만듭니다.

CreateCommittedResource 힙이 전체 리소스를 포함할 만큼 크고 리소스가 힙에 매핑되도록 리소스와 암시적 힙을 모두 만듭니다.
CreateCommittedResource1 힙이 전체 리소스를 포함할 만큼 크고 리소스가 힙에 매핑되도록 리소스와 암시적 힙(선택 사항)을 모두 만듭니다. (ID3D12Device4::CreateCommittedResource1)
CreateCommittedResource2 힙이 전체 리소스를 포함할 만큼 크고 리소스가 힙에 매핑되도록 리소스와 암시적 힙(선택 사항)을 모두 만듭니다.
CreateCommittedResource3 초기 상태가 아닌 초기 레이아웃으로 커밋된 리소스를 만듭니다.
CreateComputePipelineState 컴퓨팅 파이프라인 상태 개체를 만듭니다.
CreateConstantBufferView 리소스 데이터에 액세스하기 위한 상수 버퍼 보기를 만듭니다.
CreateDepthStencilView 리소스 데이터에 액세스하기 위한 깊이 스텐실 보기를 만듭니다.
CreateDescriptorHeap 설명자 힙 개체를 만듭니다.
CreateDirect3D12BackBufferResource 카메라의 콘텐츠 버퍼로 사용할 Direct3D 12 리소스를 만듭니다.
CreateDirect3D12ContentBufferResource 레이어의 콘텐츠 버퍼로 사용할 Direct3D 12 리소스를 만듭니다.
CreateDirect3D12HardwareProtectedBackBufferResource IHolographicCameraInterop::CreateDirect3D12HardwareProtectedBackBufferResource는 카메라의 콘텐츠 버퍼로 사용할 Direct3D 12 리소스를 만듭니다.

CreateDirect3D12HardwareProtectedContentBufferResource IHolographicQuadLayerInterop::CreateDirect3D12HardwareProtectedContentBufferResource 함수는 카메라에 대한 Direct3D 12 리소스 콘텐츠 버퍼를 만듭니다.
CreateFence 펜스 개체를 만듭니다. (ID3D12Device.CreateFence)
CreateFenceFd TBD
CreateGraphicsPipelineState 그래픽 파이프라인 상태 개체를 만듭니다.
CreateHeap 배치된 리소스 및 예약된 리소스와 함께 사용할 수 있는 힙을 만듭니다.
CreateHeap1 배치된 리소스 및 예약된 리소스와 함께 사용할 수 있는 힙(선택적으로 보호된 세션의 경우)을 만듭니다.
CreateLifetimeTracker 애플리케이션 정의 콜백과 연결된 수명 추적기를 만듭니다. 콜백은 추적된 개체의 수명이 변경될 때 알림을 받습니다.
CreateMetaCommand 지정된 메타 명령의 인스턴스를 만듭니다.
CreatePipelineLibrary 캐시된 파이프라인 라이브러리를 만듭니다.
CreatePipelineState 파이프라인 상태 스트림 설명에서 파이프라인 상태 개체를 만듭니다.
CreatePlacedResource 특정 힙에 배치되는 리소스를 만듭니다. 배치된 리소스는 사용 가능한 가장 가벼운 리소스 개체이며 가장 빠르게 만들고 삭제할 수 있습니다.

CreatePlacedResource1 특정 힙에 배치되는 리소스를 만듭니다. 배치된 리소스는 사용 가능한 가장 가벼운 리소스 개체이며 가장 빠르게 만들고 삭제할 수 있습니다.
CreatePlacedResource2 특정 힙에 배치되는 리소스를 만듭니다. 배치된 리소스는 사용 가능한 가장 가벼운 리소스 개체이며 가장 빠르게 만들고 삭제할 수 있습니다.
CreateProtectedResourceSession 콘텐츠 보호를 위한 세션을 나타내는 개체를 만듭니다.
CreateProtectedResourceSession1 보호된 리소스 세션의 유형을 나타내는 프로비저닝 **GUID**를 사용하여 **ID3D12Device4::CreateProtectedResourceSession** 메서드를 수정합니다.
CreateQueryHeap 쿼리 힙을 만듭니다. 쿼리 힙에는 쿼리 배열이 포함됩니다.
CreateRenderTargetView 리소스 데이터에 액세스하기 위한 렌더링 대상 뷰를 만듭니다. (ID3D12Device.CreateRenderTargetView)
CreateReservedResource 예약되어 있고 아직 힙의 페이지에 매핑되지 않은 리소스를 만듭니다.
CreateReservedResource1 힙의 페이지에 아직 매핑되지 않은 리소스(선택적으로 보호된 세션의 경우)를 만듭니다.
CreateReservedResource2 예약되어 있고 아직 힙의 페이지에 매핑되지 않은 리소스를 만듭니다.
CreateRootSignature 루트 서명 레이아웃을 만듭니다.
CreateSampler 텍스처에 대한 샘플링 정보를 캡슐화하는 샘플러 개체를 만듭니다.

CreateSamplerFeedbackUnorderedAccessView 샘플러 피드백을 위해 은 바인딩에 적합한 설명자를 만듭니다.
CreateShaderCacheSession 셰이더 캐시에 대한 액세스 권한을 부여하여 기존 캐시를 열거나 새 캐시를 만드는 개체를 만듭니다.
CreateShaderResourceView 리소스의 데이터에 액세스하기 위한 셰이더-리소스 뷰를 만듭니다. (ID3D12Device.CreateShaderResourceView)
CreateSharedHandle 힙, 리소스 또는 펜스 개체에 대한 공유 핸들을 만듭니다.
CreateStateObject ID3D12StateObject를 만듭니다.
CreateUnorderedAccessView 순서가 지정되지 않은 액세스를 위한 뷰를 만듭니다.
CreateWrappedResource 이 메서드는 D3D 11on12와 함께 사용할 D3D11 리소스를 만듭니다.
D3D11On12CreateDevice Direct3D 12에서 Direct3D 11 기능을 사용하는 디바이스를 만들고 Direct3D 11 interop에 사용할 기존 Direct3D 12 디바이스를 지정합니다.
D3D12CreateDevice 디스플레이 어댑터를 나타내는 디바이스를 만듭니다. (D3D12CreateDevice)
D3D12CreateRootSignatureDeserializer 레이아웃 정의(D3D12_ROOT_SIGNATURE_DESC)를 확인할 수 있도록 루트 서명을 역직렬화합니다.
D3D12CreateVersionedRootSignatureDeserializer GetUnconvertedRootSignatureDesc를 통해 역직렬화된 데이터 구조를 반환할 수 있는 인터페이스를 생성합니다.

D3D12EnableExperimentalFeatures 실험적 기능 목록을 사용하도록 설정합니다.
D3D12GetDebugInterface 디버그 인터페이스를 가져옵니다.
D3D12GetInterface 시스템이 Windows 개발자 모드에 있을 때 런타임에 SDK 버전을 선택합니다.
D3D12SerializeRootSignature ID3D12Device::CreateRootSignature에 전달할 수 있는 루트 서명 버전 1.0을 직렬화합니다.
D3D12SerializeVersionedRootSignature ID3D12Device::CreateRootSignature에 전달할 수 있는 모든 버전의 루트 서명을 직렬화합니다.
DestroyOwnedObject 수명 추적 개체를 삭제합니다.
DisableDebugLayer 디버그 계층을 사용하지 않도록 설정합니다.
DiscardResource 리소스를 삭제합니다.
Dispatch 스레드 그룹에서 컴퓨팅 셰이더를 실행합니다.
DispatchRays 광선 생성 셰이더의 스레드를 시작합니다.
DrawIndexedInstanced 인덱싱된 인스턴스화된 기본 형식을 그립니다.
DrawInstanced 인덱싱되지 않은 인스턴스화된 기본 형식을 그립니다.

EmitRaytracingAccelerationStructurePostbuildInfo 가속 구조 집합에 대한 빌드 후 속성을 내보낸다. 이를 통해 애플리케이션은 ID3D12GraphicsCommandList4::CopyRaytracingAccelerationStructure를 통해 가속 구조 작업을 수행하기 위한 출력 리소스 요구 사항을 알 수 있습니다.
EnableDebugLayer 디버그 계층을 사용하도록 설정합니다. (ID3D12Debug.EnableDebugLayer)
EnableDebugLayer 디버그 계층을 사용하도록 설정합니다. (ID3D12Debug1.EnableDebugLayer)
EnableShaderInstrumentation 이 메서드를 사용하면 PIX와 같은 도구가 셰이더를 계측할 수 있습니다.
EndEvent 직접 호출할 수 없습니다. PIX 이벤트 런타임을 사용하여 명령 큐에 이벤트를 삽입합니다. (ID3D12CommandQueue.EndEvent)
EndEvent 직접 호출할 수 없습니다. PIX 이벤트 런타임을 사용하여 명령 목록에 이벤트를 삽입합니다. (ID3D12GraphicsCommandList.EndEvent)
EndQuery 실행 중인 쿼리를 종료합니다.
EndRenderPass 렌더링 패스의 끝을 표시합니다.
EnqueueMakeResident 비동기적으로 개체를 디바이스에 상주하게 만듭니다.
EnumerateMetaCommandParameters 지정된 메타 명령의 매개 변수에 대한 리플렉션 메타데이터를 쿼리합니다.
EnumerateMetaCommands 사용 가능한 메타 명령에 대한 리플렉션 메타데이터를 쿼리합니다.

Evict 해당 데이터의 GPU 액세스를 배제하는 데이터 페이지 아웃을 사용하도록 설정합니다.
ExecuteBundle 번들을 실행합니다.
ExecuteCommandLists 실행을 위해 명령 목록 배열을 제출합니다.
ExecuteIndirect 앱은 ExecuteIndirect 메서드를 사용하여 간접 그리기/디스패치를 수행합니다.
ExecuteMetaCommand 지정한 메타 명령의 실행(또는 호출)을 그래픽 명령 목록에 기록합니다.
FindValue 해당 키가 제공된 키와 정확히 일치하는 캐시의 항목을 조회합니다.
GetAdapterLuid 현재 디바이스(어댑터)에 대한 로컬 고유 식별자를 가져옵니다.
GetAutoBreadcrumbsOutput 디바이스 제거 후 DRED(디바이스 제거 확장 데이터) 자동 이동 경로 출력을 검색합니다.
GetBaseClass 변수 기본 클래스 형식을 포함하는 ID3D12ShaderReflectionType 인터페이스 인터페이스를 가져옵니다.
GetBitwiseInstructionCount 비트 명령 수를 가져옵니다. (ID3D12ShaderReflection.GetBitwiseInstructionCount)
GetBreakOnCategory 해당 범주의 메시지가 스토리지 필터를 통과할 때 중단되는 메시지 범주를 가져옵니다. (ID3D12InfoQueue.GetBreakOnCategory)

GetBreakOnID 해당 식별자가 있는 메시지가 스토리지 필터를 통과할 때 중단할 메시지 식별자를 가져옵니다. (ID3D12InfoQueue.GetBreakOnID)
GetBreakOnSeverity 심각도 수준이 있는 메시지가 스토리지 필터를 통과할 때 중단할 메시지 심각도 수준을 가져옵니다. (ID3D12InfoQueue.GetBreakOnSeverity)
GetBuffer 현재 ID3D12ShaderReflectionVariable의 ID3D12ShaderReflectionConstantBuffer를 반환합니다.
GetCachedBlob 파이프라인 상태를 나타내는 캐시된 Blob을 가져옵니다.
GetClockCalibration 이 메서드는 동시에 CPU 및 GPU 타임스탬프 카운터를 샘플합니다.
GetCompletedValue 펜스의 현재 값을 가져옵니다. (ID3D12Fence.GetCompletedValue)
GetConstantBufferByIndex ID3D12FunctionReflection::GetConstantBufferByIndex 메서드(d3d12shader.h)는 함수에 대한 인덱스별로 상수 버퍼를 가져옵니다.
GetConstantBufferByIndex 인덱스별로 상수 버퍼를 가져옵니다.
GetConstantBufferByName 함수의 이름으로 상수 버퍼를 가져옵니다. (ID3D12FunctionReflection.GetConstantBufferByName)
GetConstantBufferByName 이름별로 상수 버퍼를 가져옵니다.
GetConversionInstructionCount 변환 명령 수를 가져옵니다. (ID3D12ShaderReflection.GetConversionInstructionCount)

GetCopyableFootprints 복사할 수 있는 리소스 레이아웃을 가져옵니다. 업로드 힙에서 공간을 하위 할당할 때 앱이 D3D12_PLACED_SUBRESOURCE_FOOTPRINT 및 D3D12_SUBRESOURCE_FOOTPRINT 채우는 데 도움이 됩니다.
GetCopyableFootprints1 복사할 수 있는 리소스 레이아웃을 가져옵니다. 업로드 힙에서 공간을 하위 할당할 때 앱이 D3D12_PLACED_SUBRESOURCE_FOOTPRINT 채우고 D3D12_SUBRESOURCE_FOOTPRINT 수 있도록 도와줍니다.
GetCPUDescriptorHandleForHeapStart 힙의 시작을 나타내는 CPU 설명자 핸들을 가져옵니다.
GetCreationFlags 현재 인스턴스가 나타내는 펜스를 만드는 데 사용되는 플래그를 가져옵니다.
GetCustomHeapProperties 어댑터의 아키텍처 속성에 따라 사용자 지정이 아닌 힙 형식에 사용되는 동일한 사용자 지정 힙 속성을 공개합니다.
GetD3D12Device 상호 운용되는 Direct3D 12 디바이스 를 검색합니다.
GetDebugParameter 선택적 명령 목록 디버그 계층 설정을 가져옵니다.
GetDebugParameter 선택적 디바이스 전체 디버그 계층 설정을 가져옵니다.
GetDesc 명령 큐에 대한 설명을 가져옵니다.
GetDesc 설명자 힙 설명을 가져옵니다.
GetDesc 힙 설명을 가져옵니다.

GetDesc 보호된 리소스 세션에 대한 설명을 검색합니다. (ID3D12ProtectedResourceSession.GetDesc)
GetDesc 리소스 설명을 가져옵니다.
GetDesc 캐시 세션을 만드는 데 사용되는 설명을 검색합니다.
GetDesc 함수의 매개 변수에 대한 매개 변수 설명자 구조를 채웁니다. (ID3D12FunctionParameterReflection.GetDesc)
GetDesc 함수에 대한 함수 설명자 구조를 채웁니다. (ID3D12FunctionReflection.GetDesc)
GetDesc 라이브러리 리플렉션에 대한 라이브러리 설명자 구조를 채웁니다. (ID3D12LibraryReflection.GetDesc)
GetDesc 셰이더 설명을 가져옵니다.
GetDesc 상수 버퍼 설명을 가져옵니다.
GetDesc 셰이더-리플렉션-변수 형식에 대한 설명을 가져옵니다.
GetDesc 셰이더 변수 설명을 가져옵니다.
GetDesc1 보호된 리소스 세션에 대한 설명을 검색합니다. (ID3D12ProtectedResourceSession1::GetDesc1)

GetDescriptorHandleIncrementSize 지정된 형식의 설명자 힙에 대한 핸들 증분 크기를 가져옵니다. 이 값은 일반적으로 올바른 양만큼 설명자 배열로 핸들을 증분하는 데 사용됩니다.
GetDevice 이 인터페이스를 만든 디바이스에 대한 포인터를 가져옵니다.
GetDeviceRemovedReason 디바이스가 제거된 이유를 가져옵니다.
GetFeatureMask 명령 목록에 설정된 디버그 기능 플래그를 반환합니다.
GetFeatureMask 어떤 디버그 기능이 켜지거나 꺼져 있는지를 나타내는 플래그의 비트 필드를 가져옵니다.
GetFunctionByIndex ID3D12LibraryReflection::GetFunctionByIndex 메서드(d3d12shader.h)는 함수 리플렉터를 가져옵니다.
GetFunctionParameter 함수 매개 변수 리플렉터를 가져옵니다. (ID3D12FunctionReflection.GetFunctionParameter)
GetGPUDescriptorHandleForHeapStart 힙의 시작을 나타내는 GPU 설명자 핸들을 가져옵니다.
GetGPUVirtualAddress 이 메서드는 버퍼 리소스의 GPU 가상 주소를 반환합니다.
GetGSInputPrimitive geometry-shader input-primitive 설명을 가져옵니다. (ID3D12ShaderReflection.GetGSInputPrimitive)
GetHeapProperties 배치 및 커밋된 리소스에 대한 리소스 힙의 속성을 검색합니다.

GetInputParameterDesc 셰이더에 대한 입력 매개 변수 설명을 가져옵니다.
GetInterfaceByIndex 인덱스별로 인터페이스를 가져옵니다.
GetInterfaceSlot 인터페이스 포인터를 나타내는 변수의 해당 인터페이스 슬롯을 가져옵니다. (ID3D12ShaderReflectionVariable.GetInterfaceSlot)
GetMemberTypeByIndex 인덱스별로 셰이더-리플렉션 변수 형식을 가져옵니다.
GetMemberTypeByName 이름으로 셰이더-리플렉션 변수 형식을 가져옵니다.
GetMemberTypeName 셰이더 리플렉션 변수 형식을 가져옵니다.
GetMessage 메시지 큐에서 메시지를 가져옵니다. (ID3D12InfoQueue.GetMessage)
GetMessageCountLimit 메시지 큐에 추가할 수 있는 최대 메시지 수를 가져옵니다. (ID3D12InfoQueue.GetMessageCountLimit)
GetMinFeatureLevel 최소 기능 수준을 가져옵니다. (ID3D12ShaderReflection.GetMinFeatureLevel)
GetMovcInstructionCount Movc 명령 수를 가져옵니다. (ID3D12ShaderReflection.GetMovcInstructionCount)
GetMovInstructionCount Mov 명령 수를 가져옵니다. (ID3D12ShaderReflection.GetMovInstructionCount)
GetMuteDebugOutput 디버그 출력이 켜지거나 꺼져 있는지 확인하는 부울을 가져옵니다.

GetNodeCount 이 디바이스와 연결된 실제 어댑터(노드) 수를 보고합니다.
GetNumInterfaces 인터페이스 수를 가져옵니다. (ID3D12ShaderReflectionType.GetNumInterfaces)
GetNumInterfaceSlots 셰이더의 인터페이스 슬롯 수를 가져옵니다. (ID3D12ShaderReflection.GetNumInterfaceSlots)
GetNumMessagesAllowedByStorageFilter 스토리지 필터를 통과하도록 허용된 메시지 수를 가져옵니다. (ID3D12InfoQueue.GetNumMessagesAllowedByStorageFilter)
GetNumMessagesDeniedByStorageFilter 스토리지 필터를 통해 전달이 거부된 메시지 수를 가져옵니다. (ID3D12InfoQueue.GetNumMessagesDeniedByStorageFilter)
GetNumMessagesDiscardedByMessageCountLimit 메시지 수 제한으로 인해 삭제된 메시지 수를 가져옵니다. (ID3D12InfoQueue.GetNumMessagesDiscardedByMessageCountLimit)
GetNumStoredMessages 메시지 큐에 현재 저장된 메시지 수를 가져옵니다. (ID3D12InfoQueue.GetNumStoredMessages)
GetNumStoredMessagesAllowedByRetrievalFilter 검색 필터를 통과할 수 있는 메시지 수를 가져옵니다. (ID3D12InfoQueue.GetNumStoredMessagesAllowedByRetrievalFilter)
GetOutputParameterDesc 셰이더에 대한 출력 매개 변수 설명을 가져옵니다.
GetPageFaultAllocationOutput DRED(디바이스 제거된 확장 데이터) 페이지 오류 데이터를 검색합니다.
GetPatchConstantParameterDesc 셰이더에 대한 패치 상수 매개 변수 설명을 가져옵니다.

GetPipelineStackSize 현재 파이프라인 스택 크기를 가져옵니다.
GetPrivateData 디바이스 개체에서 애플리케이션 정의 데이터를 가져옵니다.
GetRaytracingAccelerationStructurePrebuildInfo 드라이버에서 리소스 요구 사항을 쿼리하여 가속 구조를 빌드합니다.
GetRequiredParameterResourceSize 지정된 단계에 대해 메타 명령에 대해 지정된 런타임 매개 변수 리소스에 필요한 메모리 양을 검색합니다.
GetRequiresFlags 셰이더의 요구 사항을 나타내는 플래그 그룹을 가져옵니다. (ID3D12ShaderReflection.GetRequiresFlags)
GetResourceAllocationInfo 이 어댑터의 리소스 컬렉션에 필요한 메모리의 크기와 맞춤을 가져옵니다.
GetResourceAllocationInfo1 이 어댑터의 리소스 컬렉션에 필요한 메모리의 크기 및 맞춤에 대한 다양한 정보를 가져옵니다. (ID3D12Device4::GetResourceAllocationInfo1)
GetResourceAllocationInfo2 이 어댑터의 리소스 컬렉션에 필요한 메모리의 크기와 맞춤에 대한 풍부한 정보를 가져옵니다. (ID3D12Device8::GetResourceAllocationInfo2)
GetResourceBindingDesc 리소스가 함수에 바인딩되는 방법에 대한 설명을 가져옵니다. (ID3D12FunctionReflection.GetResourceBindingDesc)
GetResourceBindingDesc 리소스가 셰이더에 바인딩되는 방법에 대한 설명을 가져옵니다. (ID3D12ShaderReflection.GetResourceBindingDesc)

GetResourceBindingDescByName 리소스가 함수에 바인딩되는 방법에 대한 설명을 가져옵니다. (ID3D12FunctionReflection.GetResourceBindingDescByName)
GetResourceBindingDescByName 리소스가 셰이더에 바인딩되는 방법에 대한 설명을 가져옵니다. (ID3D12ShaderReflection.GetResourceBindingDescByName)
GetResourceTiling 타일형 리소스가 타일로 분할되는 방법에 대한 정보를 가져옵니다. (ID3D12Device.GetResourceTiling)
GetRetrievalFilter 검색 필터 스택의 맨 위에 있는 검색 필터를 가져옵니다. (ID3D12InfoQueue.GetRetrievalFilter)
GetRetrievalFilterStackSize 검색 필터 스택의 크기를 바이트 단위로 가져옵니다. (ID3D12InfoQueue.GetRetrievalFilterStackSize)
GetRootSignatureDesc 루트 서명의 레이아웃을 가져옵니다.
GetRootSignatureDescAtVersion 루트 서명 설명 구조를 요청된 버전으로 변환합니다.
GetSerializedSize 데이터베이스의 현재 콘텐츠를 serialize하는 데 필요한 메모리 양을 반환합니다.
GetSessionStatus 보호된 세션의 상태를 가져옵니다.
GetShaderIdentifier 셰이더 레코드에서 사용할 수 있는 셰이더의 고유 식별자를 검색합니다.
GetShaderStackSize HLSL에서 광선 추적 셰이더를 호출하는 데 필요한 스택 메모리의 양을 가져옵니다.

GetStatusFence 보호된 세션의 펜스를 검색합니다. 펜스에서 현재 고유성 유효성 값 (ID3D12Fence::GetCompletedValue 사용)을 검색하고 해당 값의 변경 내용에 대한 모니터를 추가할 수 있습니다. 읽기 전용 펜스입니다.
GetStorageFilter 스토리지 필터 스택의 맨 위에 있는 스토리지 필터를 가져옵니다. (ID3D12InfoQueue.GetStorageFilter)
GetStorageFilterStackSize 스토리지 필터 스택의 크기를 바이트 단위로 가져옵니다. (ID3D12InfoQueue.GetStorageFilterStackSize)
GetSubType 클래스의 기본 클래스를 가져옵니다. (ID3D12ShaderReflectionType.GetSubType)
GetThreadGroupSize 셰이더 스레드 그룹 그리드의 X, Y 및 Z 차원의 크기(스레드 단위)를 검색합니다. (ID3D12ShaderReflection.GetThreadGroupSize)
GetTimestampFrequency 이 메서드는 GPU 타임스탬프 카운터가 증가되는 속도를 결정하는 데 사용됩니다.
GetType 직접, 번들, 컴퓨팅 또는 복사와 같은 명령 목록의 형식을 가져옵니다.
GetType 셰이더 변수 형식을 가져옵니다.
GetUnconvertedRootSignatureDesc 루트 서명 버전 간에 변환하지 않고 루트 서명의 레이아웃을 가져옵니다.
GetVariableByIndex 인덱스별로 셰이더 리플렉션 변수를 가져옵니다.
GetVariableByName 이름으로 변수를 가져옵니다. (ID3D12FunctionReflection.GetVariableByName)

GetVariableByName 이름으로 변수를 가져옵니다. (ID3D12ShaderReflection.GetVariableByName)
GetVariableByName 이름으로 셰이더 리플렉션 변수를 가져옵니다.
IASetIndexBuffer 인덱스 버퍼에 대한 뷰를 설정합니다.
IASetPrimitiveTopology 기본 형식 및 입력 어셈블러 단계의 입력 데이터를 설명하는 데이터 순서에 대한 정보를 바인딩합니다. (ID3D12GraphicsCommandList.IASetPrimitiveTopology)
IASetVertexBuffers 꼭짓점 버퍼에 대한 CPU 설명자 핸들을 설정합니다.
ImplementsInterface 클래스 형식이 인터페이스를 구현하는지 여부를 나타냅니다. (ID3D12ShaderReflectionType.ImplementsInterface)
InitializeMetaCommand 지정된 메타 명령을 초기화합니다.
IsEqual 두 ID3D12ShaderReflectionType 인터페이스 포인터의 기본 형식이 같은지 여부를 나타냅니다.
IsOfType 변수가 지정된 형식인지 여부를 나타냅니다. (ID3D12ShaderReflectionType.IsOfType)
IsSampleFrequencyShader 셰이더가 샘플 빈도 셰이더인지 여부를 나타냅니다. (ID3D12ShaderReflection.IsSampleFrequencyShader)
LifetimeStateUpdated 수명 추적 개체의 수명 상태가 변경될 때 호출됩니다.

LoadComputePipeline 라이브러리에서 요청된 PSO를 검색합니다. 입력 desc는 현재 라이브러리 데이터베이스의 데이터와 일치하며 PSO 콘텐츠의 중복을 방지하기 위해 기억됩니다.
LoadGraphicsPipeline 라이브러리에서 요청된 PSO를 검색합니다.
LoadPipeline 라이브러리에서 요청된 PSO를 검색합니다. 파이프라인 스트림 설명은 라이브러리 데이터베이스와 일치하며 PSO 콘텐츠의 중복을 방지하기 위해 기억됩니다.
MakeResident 개체를 디바이스에 상주하게 만듭니다.
Map 리소스에서 지정된 하위 리소스에 대한 CPU 포인터를 가져오지만 애플리케이션에 포인터 값을 공개하지 않을 수 있습니다. 또한 맵은 필요한 경우 CPU 캐시를 무효화하여 CPU가 이 주소로 읽도록 GPU에서 수정한 내용을 반영합니다.
OMSetBlendFactor 픽셀 셰이더, 렌더링 대상 또는 둘 다에 대한 값을 변조하는 혼합 요소를 설정합니다.
OMSetDepthBounds 이 메서드를 사용하면 깊이 범위를 동적으로 변경할 수 있습니다.
OMSetRenderTargets 렌더링 대상 및 깊이 스텐실에 대한 CPU 설명자 핸들을 설정합니다.
OMSetStencilRef 깊이 스텐실 테스트에 대한 참조 값을 설정합니다.
OpenExistingHeapFromAddress 주소에서 시스템 메모리에 특수 용도의 진단 힙을 만듭니다. 생성된 힙은 GPU 오류 또는 디바이스 제거 시나리오의 경우에도 지속될 수 있습니다.
OpenExistingHeapFromFileMapping 파일 매핑 개체에서 시스템 메모리에 특수 용도의 진단 힙을 만듭니다. 생성된 힙은 GPU 오류 또는 디바이스 제거 시나리오의 경우에도 지속될 수 있습니다.

OpenSharedHandle HANDLE 및 REFIID를 사용하여 공유 리소스, 공유 힙 및 공유 펜스에 대한 핸들을 엽니다.
OpenSharedHandleByName 이름 및 액세스를 사용하여 공유 리소스, 공유 힙 및 공유 펜스에 대한 핸들을 엽니다.
PopRetrievalFilter 검색 필터 스택의 맨 위에서 검색 필터를 팝합니다. (ID3D12InfoQueue.PopRetrievalFilter)
PopStorageFilter 스토리지 필터 스택의 맨 위에서 스토리지 필터를 팝합니다. (ID3D12InfoQueue.PopStorageFilter)
있음 D3D 계층과 진단 도구 간에 리소스(또는 하위 리소스)를 공유합니다.
PushCopyOfRetrievalFilter 검색 필터 스택의 맨 위에 있는 검색 필터 복사본을 검색 필터 스택에 푸시합니다. (ID3D12InfoQueue.PushCopyOfRetrievalFilter)
PushCopyOfStorageFilter 현재 스토리지 필터 스택의 맨 위에 있는 스토리지 필터 복사본을 스토리지 필터 스택에 푸시합니다. (ID3D12InfoQueue.PushCopyOfStorageFilter)
PushEmptyRetrievalFilter 빈 검색 필터를 검색 필터 스택에 푸시합니다. (ID3D12InfoQueue.PushEmptyRetrievalFilter)
PushEmptyStorageFilter 빈 스토리지 필터를 스토리지 필터 스택에 푸시합니다. (ID3D12InfoQueue.PushEmptyStorageFilter)
PushRetrievalFilter 검색 필터를 검색 필터 스택에 푸시합니다. (ID3D12InfoQueue.PushRetrievalFilter)
PushStorageFilter 스토리지 필터를 스토리지 필터 스택에 푸시합니다. (ID3D12InfoQueue.PushStorageFilter)

ReadFromSubresource CPU를 사용하여 하위 리소스에서 데이터를 복사하여 CPU가 정의되지 않은 레이아웃으로 대부분의 텍스처의 내용을 읽을 수 있도록 합니다.
ReleaseWrappedResources D3D 11on12용으로 래핑된 D3D11 리소스를 릴리스합니다.
RemoveDevice **RemoveDevice**를 호출하여 GPU 디바이스에 문제가 발생하여 더 이상 사용할 수 없음을 Direct3D 12 런타임에 나타낼 수 있습니다.
ReportLiveDeviceObjects 디바이스 개체의 수명에 대한 정보를 보고합니다.
ReportLiveDeviceObjects 디바이스 개체의 수명에 보고할 정보의 양을 지정합니다.
Reset 명령 할당자와 연결된 메모리를 다시 사용하도록 나타냅니다.
Reset 새 명령 목록을 방금 만든 것처럼 명령 목록을 초기 상태로 다시 설정합니다. (ID3D12GraphicsCommandList.Reset)
ResolveQueryData 쿼리에서 데이터를 추출합니다. ResolveQueryData는 모든 힙 형식(기본, 업로드 및 다시 읽기)에 작동합니다. ResolveQueryData는 모든 힙 형식(기본, 업로드 및 다시 읽기)에 작동합니다. .
ResolveSubresource 다중 샘플링된 리소스를 다중 샘플링이 아닌 리소스에 복사합니다.
ResolveSubresourceRegion 다중 샘플링 또는 압축된 리소스의 영역을 다중 샘플링되지 않은 리소스 또는 비압축 리소스에 복사합니다.
ResourceBarrier 드라이버에 리소스에 대한 여러 액세스를 동기화해야 한다는 것을 알 수 있습니다. (ID3D12GraphicsCommandList.ResourceBarrier)

ReturnUnderlyingResource 이 메서드를 사용하면 Direct3D 11 리소스 개체를 Direct3D11On12로 반환하고 리소스를 사용할 준비가 된 시기를 나타낼 수 있습니다.
RSSetScissorRects 가위 사각형 배열을 래스터라이저 단계에 바인딩합니다.
RSSetShadingRate ID3D12GraphicsCommandList5::RSSetShadingRate 메서드(d3d12.h)는 VRS(가변 속도 음영)에 대한 기본 음영 속도 및 결합자를 설정합니다.
RSSetShadingRateImage ID3D12GraphicsCommandList5::RSSetShadingRateImage 메서드(d3d12.h)는 VRS(가변 속도 음영)에 대한 화면 공간 음영 속도 이미지를 설정합니다.
RSSetViewports 뷰포트 배열을 파이프라인의 래스터라이저 단계에 바인딩합니다. (ID3D12GraphicsCommandList.RSSetViewports)
직렬화 나중에 런타임에 다시 제공될 라이브러리의 내용을 제공된 메모리에 씁니다.
SetAutoBreadcrumbsEnablement DRED(디바이스 제거된 확장 데이터) 자동 이동 경로에 대한 사용 설정을 구성합니다.
SetBackgroundProcessingMode 드라이버 백그라운드 처리 최적화 모드를 설정합니다.
SetBreakOnCategory 해당 범주의 메시지가 스토리지 필터를 통과할 때 중단되도록 메시지 범주를 설정합니다. (ID3D12InfoQueue.SetBreakOnCategory)
SetBreakOnID 해당 식별자가 있는 메시지가 스토리지 필터를 통과할 때 중단되도록 메시지 식별자를 설정합니다. (ID3D12InfoQueue.SetBreakOnID)

SetBreakOnSeverity 심각도 수준이 있는 메시지가 스토리지 필터를 통과할 때 중단되도록 메시지 심각도 수준을 설정합니다. (ID3D12InfoQueue.SetBreakOnSeverity)
SetComputeRoot32BitConstant 컴퓨팅 루트 서명에서 상수를 설정합니다.
SetComputeRoot32BitConstants 컴퓨팅 루트 서명에서 상수 그룹을 설정합니다.
SetComputeRootConstantBufferView 컴퓨팅 루트 서명의 상수 버퍼에 대한 CPU 설명자 핸들을 설정합니다.
SetComputeRootDescriptorTable 설명자 테이블을 컴퓨팅 루트 서명으로 설정합니다.
SetComputeRootShaderResourceView 컴퓨팅 루트 서명의 셰이더 리소스에 대한 CPU 설명자 핸들을 설정합니다.
SetComputeRootSignature 컴퓨팅 루트 서명의 레이아웃을 설정합니다.
SetComputeRootUnorderedAccessView 컴퓨팅 루트 서명에서 순서가 지정되지 않은 액세스 보기 리소스에 대한 CPU 설명자 핸들을 설정합니다.
SetDebugParameter 명령 목록의 선택적 디버그 계층 설정을 수정합니다.
SetDebugParameter D3D12 선택적 디바이스 전체 디버그 계층 설정을 수정합니다.
SetDeleteOnDestroy 지정된 캐시에 해당하는 모든 캐시 세션 개체가 제거되면 캐시가 지워집니다.
SetDescriptorHeaps 명령 목록과 연결된 현재 바인딩된 설명자 힙을 변경합니다.

SetEnableAutoName 개체의 자동 이름을 구성합니다.
SetEnableGPUBasedValidation 이 메서드는 디버그 계층이 사용하도록 설정된 디바이스를 만들기 전에 GBV(GPU-Based 유효성 검사)를 사용하거나 사용하지 않도록 설정합니다.
SetEnableGPUBasedValidation 이 메서드는 디버그 계층이 사용하도록 설정된 디바이스를 만들기 전에 GPU 기반 유효성 검사(GBV)를 사용하거나 사용하지 않도록 설정합니다.
SetEnableSynchronizedCommandQueueValidation 디버그 계층이 활성화된 D3D12 디바이스를 사용할 때 종속 명령 큐 동기화를 사용하거나 사용하지 않도록 설정합니다.
SetEnableSynchronizedCommandQueueValidation 디버그 계층이 활성화된 Direct3D 12 디바이스를 사용할 때 종속 명령 큐 동기화를 사용하거나 사용하지 않도록 설정합니다.
SetEventOnCompletion 펜스가 특정 값에 도달할 때 발생해야 하는 이벤트를 지정합니다. (ID3D12Fence.SetEventOnCompletion)
SetEventOnMultipleFenceCompletion 하나 이상의 펜스 컬렉션이 특정 값에 도달할 때 발생해야 하는 이벤트를 지정합니다.
SetFeatureMask 명령 목록의 디버그 기능을 켜거나 끕니다.
SetFeatureMask 디버그 기능을 켜고 끄는 플래그의 비트 필드를 설정합니다. (ID3D12DebugDevice.SetFeatureMask)
SetForceLegacyBarrierValidation TBD

SetGPUBasedValidationFlags 이 메서드는 디버그 디바이스가 런타임에 수행할 GPU 기반 유효성 검사 수준을 구성합니다. (ID3D12Debug2.SetGPUBasedValidationFlags)
SetGPUBasedValidationFlags 이 메서드는 디버그 디바이스가 런타임에 수행할 GPU 기반 유효성 검사 수준을 구성합니다. (ID3D12Debug3.SetGPUBasedValidationFlags)
SetGraphicsRoot32BitConstant 그래픽 루트 서명에서 상수를 설정합니다.
SetGraphicsRoot32BitConstants 그래픽 루트 서명에서 상수 그룹을 설정합니다.
SetGraphicsRootConstantBufferView 그래픽 루트 서명의 상수 버퍼에 대한 CPU 설명자 핸들을 설정합니다.
SetGraphicsRootDescriptorTable 설명자 테이블을 그래픽 루트 서명으로 설정합니다.
SetGraphicsRootShaderResourceView 그래픽 루트 서명의 셰이더 리소스에 대한 CPU 설명자 핸들을 설정합니다.
SetGraphicsRootSignature 그래픽 루트 서명의 레이아웃을 설정합니다.
SetGraphicsRootUnorderedAccessView 그래픽 루트 서명에서 순서가 지정되지 않은 액세스 보기 리소스에 대한 CPU 설명자 핸들을 설정합니다.
SetMarker 직접 호출할 수 없습니다. PIX 이벤트 런타임을 사용하여 명령 큐에 이벤트를 삽입합니다. (ID3D12CommandQueue.SetMarker)
SetMarker 직접 호출할 수 없습니다. PIX 이벤트 런타임을 사용하여 명령 목록에 이벤트를 삽입합니다. (ID3D12GraphicsCommandList.SetMarker)

SetMessageCountLimit 메시지 큐에 추가할 수 있는 최대 메시지 수를 설정합니다. (ID3D12InfoQueue.SetMessageCountLimit)
SetMuteDebugOutput 디버그 출력을 켜거나 끄는 부울을 설정합니다. (ID3D12InfoQueue.SetMuteDebugOutput)
SetName 이름을 디바이스 개체와 연결합니다. 이 이름은 디버그 진단 및 도구에 사용하기 위한 것입니다.
SetPageFaultEnablement DRED(디바이스 제거된 확장 데이터) 페이지 오류 보고에 대한 사용 설정을 구성합니다.
SetPipelineStackSize 현재 파이프라인 스택 크기를 설정합니다.
SetPipelineState 모든 셰이더를 설정하고 GPU(그래픽 처리 장치) 파이프라인의 고정 함수 상태 대부분을 프로그래밍합니다.
SetPipelineState1 명령 목록에 상태 개체를 설정합니다.
SetPredication 렌더링 조건자를 설정합니다.
SetPrivateData 애플리케이션 정의 데이터를 디바이스 개체로 설정하고 해당 데이터를 애플리케이션 정의 GUID와 연결합니다.
SetPrivateDataInterface IUnknown 파생 인터페이스를 디바이스 개체와 연결하고 해당 인터페이스를 애플리케이션 정의 GUID와 연결합니다.
SetProtectedResourceSession 명령 목록의 후속 명령으로 보호된 리소스에 액세스할 수 있는지 여부를 지정합니다.

SetResidencyPriority 이 메서드는 지정된 개체 목록의 상주 우선 순위를 설정합니다.
SetSamplePositions 이 메서드는 후속 그리기, 복사, 확인 및 유사한 작업에 사용되는 샘플 위치를 구성합니다.
SetSDKVersion 사용할 SDK 버전을 구성합니다.
SetStablePowerState 특정 유형의 프로파일링 및 실험적 프로토타입에 대한 개발 시간 지원.
SetViewInstanceMask 후속 그리기에서 사용할 수 있는 보기 인스턴스를 제어하는 마스크를 설정합니다.
SetWatsonDumpEnablement DRED(디바이스 제거 확장 데이터) Watson 덤프 만들기에 대한 사용 설정을 구성합니다.
ShaderCacheControl Direct3D 또는 드라이버에서 내부적으로 사용되는 캐시의 동작을 수정합니다.
ShaderInstrumentationEnabled 셰이더 계측을 사용할지 여부를 결정합니다.
SharedFenceSignal D3D 계층과 진단 도구 간의 공유 펜스에 신호를 보냅니다.
ShareWithHost TBD
신호 펜스를 지정된 값으로 업데이트.
신호 펜스를 지정된 값으로 설정합니다.

SOSetTargets 스트림 출력 버퍼 뷰를 설정합니다.
StorePipeline 해당 이름의 내부 데이터베이스에 입력 PSO를 추가합니다.
StoreValue 캐시에 항목을 추가합니다.
UnacquireDirect3D12BufferResource IHolographicCameraInterop::UnacquireDirect3D12BufferResource 함수는 Direct3D 12 버퍼 리소스를 획득하지 않습니다.
UnacquireDirect3D12BufferResource IHolographicQuadLayerInterop::UnacquireDirect3D12BufferResource 함수는 Direct3D 12 버퍼 리소스를 획득하지 않습니다.
Unmap 리소스에서 지정된 하위 리소스에 대한 CPU 포인터를 무효화합니다.
UnwrapUnderlyingResource Direct3D 11 리소스 개체를 래프 해제하고 Direct3D 12 리소스 개체로 검색합니다.
UpdateTileMappings 예약된 리소스의 타일 위치를 리소스 힙의 메모리 위치에 매핑하는 업데이트.
Wait GPU 쪽 대기를 큐에 대기하고 즉시 반환합니다. GPU 쪽 대기는 지정된 펜스가 지정된 값에 도달하거나 초과할 때까지 GPU가 대기하는 위치입니다.
WriteBufferImmediate 명령 스트림에서 직접 지정된 버퍼 위치에 32비트 직접 값을 씁니다. (ID3D12GraphicsCommandList2.WriteBufferImmediate)
WriteToSubresource CPU를 사용하여 데이터를 하위 리소스에 복사하여 CPU가 정의되지 않은 레이아웃으로 대부분의 텍스처의 콘텐츠를 수정할 수 있도록 합니다.

인터페이스

ID3D11On12Device Direct3D11on12에 대한 D3D11 리소스 만들기, 래핑 및 릴리스를 처리합니다.
ID3D11On12Device1 Direct3D 11 디바이스를 전달하지만 Direct3D 12를 대신 활용하려는 구성 요소와의 상호 운용성을 향상할 수 있습니다.
ID3D11On12Device2 Direct3D 11 API를 통해 만든 리소스를 가져와 Direct3D 12에서 사용할 수 있습니다.
ID3D12CommandAllocator GPU(그래픽 처리 장치) 명령에 대한 스토리지 할당을 나타냅니다.
ID3D12CommandList ID3D12GraphicsCommandList가 상속하는 인터페이스입니다. GPU가 실행하는 순서가 지정된 명령 집합을 나타내며 확장이 그래픽(예: 컴퓨팅 및 복사)에 대한 명령 목록보다 다른 명령 목록을 지원할 수 있도록 허용합니다.
ID3D12CommandQueue 명령 목록을 제출하고, 명령 목록 실행을 동기화하고, 명령 큐를 계측하고, 리소스 타일 매핑을 업데이트하는 메서드를 제공합니다.
ID3D12CommandSignature 명령 서명 개체를 사용하면 앱에서 사용할 버퍼 형식, 명령 유형 및 리소스 바인딩을 포함하여 간접 그리기를 지정할 수 있습니다.
ID3D12Debug 디버그 계층을 켜는 데 사용되는 인터페이스입니다.
ID3D12Debug1 GPU-Based 유효성 검사 및 종속 명령 큐 동기화를 디버그 계층에 추가합니다.
ID3D12Debug2 구성 가능한 수준의 GPU 기반 유효성 검사를 디버그 계층에 추가합니다. (ID3D12Debug2)

ID3D12Debug3 구성 가능한 수준의 GPU 기반 유효성 검사를 디버그 계층에 추가합니다. (ID3D12Debug3)
ID3D12Debug4 디버그 계층을 사용하지 않도록 설정하는 기능을 추가합니다.
ID3D12Debug5 개체의 자동 명명을 구성하는 기능을 디버그 계층에 추가합니다.
ID3D12Debug6 디버그 인터페이스는 디버그 설정을 제어합니다.
ID3D12DebugCommandList 명령 목록을 모니터링하고 디버그하는 메서드를 제공합니다.
ID3D12DebugCommandList1 이 인터페이스를 사용하면 추가 명령 목록 디버그 계층 설정을 수정할 수 있습니다.
ID3D12DebugCommandQueue 명령 큐를 모니터링하고 디버그하는 메서드를 제공합니다.
ID3D12DebugDevice 이 인터페이스는 디버깅을 위한 그래픽 디바이스를 나타냅니다.
ID3D12DebugDevice1 디바이스 전체 디버그 계층 설정을 지정합니다.
ID3D12DescriptorHeap 설명자 힙은 설명자의 연속 할당 컬렉션으로, 모든 설명자에 대한 하나의 할당입니다.
ID3D12Device 가상 어댑터를 나타냅니다. 명령 할당자, 명령 목록, 명령 큐, 펜스, 리소스, 파이프라인 상태 개체, 힙, 루트 서명, 샘플러 및 여러 리소스 뷰를 만드는 데 사용됩니다.
ID3D12Device1 가상 어댑터를 나타내고 ID3D12Device에서 제공하는 메서드 범위를 확장합니다.

ID3D12Device10 TBD
ID3D12Device2 가상 어댑터를 나타냅니다. 이 인터페이스는 ID3D12Device1을 확장하여 파이프라인 상태 스트림 설명에서 파이프라인 상태 개체를 만듭니다.
ID3D12Device3 가상 어댑터를 나타냅니다. 이 인터페이스는 ID3D12Device2를 확장하여 GPU 오류 또는 디바이스 제거 시나리오의 경우에도 지속되는 시스템 메모리에 특수한 용도의 진단 힙을 만들 수 있도록 지원합니다.
ID3D12Device4 가상 어댑터를 나타냅니다. 이 인터페이스는 ID3D12Device3을 확장합니다.
ID3D12Device5 가상 어댑터를 나타냅니다. 이 인터페이스는 ID3D12Device4를 확장합니다.
ID3D12Device6 가상 어댑터를 나타냅니다. 이 인터페이스는 ID3D12Device5를 확장합니다.
ID3D12Device7 가상 어댑터를 나타냅니다. 이 인터페이스는 ID3D12Device6을 확장합니다.
ID3D12Device8 가상 어댑터를 나타냅니다. 이 인터페이스는 ID3D12Device7을 확장합니다.
ID3D12Device9 가상 어댑터를 나타냅니다. 이 인터페이스는 ID3D12Device8 을 확장하여 셰이더 캐시를 관리하는 메서드를 추가합니다.
ID3D12DeviceChild ID3D12PipelineLibrary, ID3D12CommandList, ID3D12Pageable 및 ID3D12RootSignature를 포함하여 다른 핵심 인터페이스가 상속하는 인터페이스입니다. 생성된 디바이스 개체로 돌아가는 메서드를 제공합니다.
ID3D12DeviceRemovedExtendedData 디바이스 DRED(확장 데이터 제거) 데이터에 대한 런타임 액세스를 제공합니다.

ID3D12DeviceRemovedExtendedDataSettings 이 인터페이스는 DRED(디바이스 제거된 확장 데이터) 설정을 제어합니다.
ID3D12Fence 펜스, CPU 동기화에 사용되는 개체 및 하나 이상의 GPU를 나타냅니다. (ID3D12Fence)
ID3D12Fence1 펜스를 나타냅니다. 이 인터페이스는 ID3D12Fence를 확장하고 원래 펜스를 만드는 데 사용되는 플래그의 검색을 지원합니다.
ID3D12FunctionParameterReflection function-parameter-reflection 인터페이스는 function-parameter 정보에 액세스합니다. (ID3D12FunctionParameterReflection)
ID3D12FunctionReflection 함수-리플렉션 인터페이스는 함수 정보에 액세스합니다. (ID3D12FunctionReflection)
ID3D12GraphicsCommandList 렌더링을 위한 그래픽 명령 목록을 캡슐화합니다. 명령 목록 실행을 계측하고 파이프라인 상태를 설정 및 지우기 위한 API를 포함합니다.
ID3D12GraphicsCommandList1 렌더링을 위한 그래픽 명령 목록을 캡슐화하고, 인터페이스를 확장하여 프로그래밍 가능한 샘플 위치, 지연 래치 기술 구현을 위한 원자성 복사본 및 선택적 깊이 경계 테스트를 지원합니다.
ID3D12GraphicsCommandList2 렌더링을 위한 그래픽 명령 목록을 캡슐화하여 버퍼에 직접 직접 값을 쓸 수 있도록 인터페이스를 확장합니다.
ID3D12GraphicsCommandList3 렌더링을 위한 그래픽 명령 목록을 캡슐화합니다.
ID3D12GraphicsCommandList4 렌더링을 위한 그래픽 명령 목록을 캡슐화하고, 광선 추적 및 렌더링 패스를 지원하도록 인터페이스를 확장합니다.

ID3D12GraphicsCommandList5 렌더링을 위한 그래픽 명령 목록을 캡슐화하여 VRS(가변 속도 음영)를 지원하도록 인터페이스를 확장합니다.
ID3D12GraphicsCommandList7 TBD
ID3D12Heap 힙은 실제 메모리를 관리하는 데 사용되는 연속 메모리 할당의 추상화입니다. 이 힙은 ID3D12Resource 개체와 함께 사용하여 배치된 리소스 또는 예약된 리소스를 지원할 수 있습니다.
ID3D12InfoQueue 정보 큐 인터페이스는 디버그 메시지를 저장, 검색 및 필터링합니다. 큐는 메시지 큐, 선택적 스토리지 필터 스택 및 선택적 검색 필터 스택으로 구성됩니다. (ID3D12InfoQueue)
ID3D12LibraryReflection 라이브러리 리플렉션 인터페이스는 라이브러리 정보에 액세스합니다. (ID3D12LibraryReflection)
ID3D12LifetimeOwner 개체의 수명 변경에 대한 알림을 받는 데 사용되는 애플리케이션 정의 콜백을 나타냅니다.
ID3D12LifetimeTracker 수명 추적 개체의 수명을 제어하기 위한 기능을 나타냅니다.
ID3D12MetaCommand 메타 명령을 나타냅니다. 메타 명령은 IHV(독립 하드웨어 공급업체)에 의해 가속화되는 알고리즘을 나타내는 Direct3D 12 개체입니다. 드라이버에서 구현하는 명령 생성기에 대한 불투명 참조입니다.
ID3D12Object ID3D12Device 및 ID3D12DeviceChild가 상속하는 인터페이스입니다. 프라이빗 데이터를 연결하고 개체 이름에 주석을 추가하는 메서드를 제공합니다.
ID3D12Pageable 다른 많은 핵심 인터페이스가 상속하는 인터페이스입니다. 개체 형식이 GPU에 액세스할 수 있는 메모리의 양을 캡슐화한다는 것을 나타냅니다. 하지만 애플리케이션이 개체의 상주를 조작할 수 있는지 여부를 강력하게 나타내지는 않습니다.

ID3D12PipelineLibrary 파이프라인 라이브러리, 특히 개별 PSO 로드 및 검색을 관리합니다.
ID3D12PipelineLibrary1 파이프라인 라이브러리를 관리합니다. 이 인터페이스는 ID3D12PipelineLibrary를 확장하여 파이프라인 상태 스트림 설명에서 PSO를 로드합니다.
ID3D12PipelineState 현재 설정된 모든 셰이더와 특정 고정 함수 상태 개체의 상태를 나타냅니다.
ID3D12ProtectedResourceSession 보호된 리소스 세션의 유효성을 모니터링합니다. (ID3D12ProtectedResourceSession)
ID3D12ProtectedResourceSession1 보호된 리소스 세션의 유효성을 모니터링합니다. (ID3D12ProtectedResourceSession1)
ID3D12ProtectedSession 다양한 유형의 세션에서 세션의 유효성을 일관되게 모니터링할 수 있는 기본 기능을 제공합니다.
ID3D12QueryHeap 쿼리 힙을 관리합니다. 쿼리 힙은 인덱스에서 참조하는 쿼리 배열을 보유합니다.
ID3D12Resource CPU 및 GPU의 일반화된 기능을 캡슐화하여 실제 메모리 또는 힙을 읽고 씁니다. 셰이더 샘플링에 최적화된 다차원 데이터뿐만 아니라 간단한 데이터 배열을 구성하고 조작하기 위한 추상화가 포함되어 있습니다.
ID3D12RootSignature 루트 서명은 그래픽 파이프라인에 바인딩되는 리소스를 정의합니다. 루트 서명은 앱에서 구성되고 명령 목록을 셰이더에 필요한 리소스에 연결합니다. 현재 앱당 하나의 그래픽과 하나의 컴퓨팅 루트 서명이 있습니다.
ID3D12RootSignatureDeserializer 직렬화된 루트 서명 버전 1.0의 역직렬화된 D3D12_ROOT_SIGNATURE_DESC 데이터 구조를 반환하는 메서드를 포함합니다.
ID3D12SDKConfiguration SDK 구성 메서드를 제공합니다.

ID3D12ShaderCacheSession 셰이더 캐시 세션을 나타냅니다.
ID3D12ShaderReflection 셰이더 리플렉션 인터페이스는 셰이더 정보에 액세스합니다. (ID3D12ShaderReflection)
ID3D12ShaderReflectionConstantBuffer 이 셰이더 리플렉션 인터페이스는 상수 버퍼에 대한 액세스를 제공합니다. (ID3D12ShaderReflectionConstantBuffer)
ID3D12ShaderReflectionType 이 셰이더 리플렉션 인터페이스는 변수 형식에 대한 액세스를 제공합니다. (ID3D12ShaderReflectionType)
ID3D12ShaderReflectionVariable 이 셰이더 리플렉션 인터페이스는 변수에 대한 액세스를 제공합니다. (ID3D12ShaderReflectionVariable)
ID3D12SharingContract D3D11On12 진단 계층과 그래픽 진단 도구 간의 계약의 일부입니다.
ID3D12StateObject 애플리케이션이 단일 단위로 관리하고 컴파일 또는 최적화와 같이 처리하도록 드라이버에 원자성으로 제공되는 셰이더를 포함한 구성 상태의 가변 양을 나타냅니다.
ID3D12StateObjectProperties ID3D12StateObject의 속성을 가져오고 설정하는 메서드를 제공합니다.
ID3D12Tools 이 인터페이스는 PIX와 같은 도구에 대한 런타임을 구성하는 데 사용됩니다. 다른 시나리오에서 의도하거나 지원되지 않습니다.
ID3D12VersionedRootSignatureDeserializer 직렬화된 루트 서명의 모든 버전의 역직렬화된 D3D12_ROOT_SIGNATURE_DESC1 데이터 구조를 반환하는 메서드를 포함합니다.
ID3D12VirtualizationGuestDevice TBD

IHolographicCameraInterop HolographicCamera를 확장하여 Direct3D 12에서 홀로그램 렌더링을 위한 백 버퍼로 2D 텍스처 리소스를 만들고 사용할 수 있도록 합니다.
IHolographicCameraRenderingParametersInterop 홀로그램 렌더링에 Direct3D 12를 사용하는 애플리케이션의 경우 HOLographicCameraRenderingParameters 클래스와 COM 상호 운용을 허용하는 nano-COM 인터페이스입니다.
IHolographicQuadLayerInterop 홀로그램 렌더링에 Direct3D 12를 사용하는 앱의 경우 HolographicQuadLayer Windows 런타임 클래스와 COM 상호 운용을 허용하는 nano-COM 인터페이스입니다.
IHolographicQuadLayerUpdateParametersInterop 홀로그램 렌더링에 Direct3D 12를 사용하는 애플리케이션에 대한 HolographicQuadLayerUpdateParameters 클래스와 COM 상호 운용을 허용하는 nano-COM 인터페이스입니다.

구조체

D3D11_RESOURCE_FLAGS ID3D11On12Device::CreateWrappedResource와 함께 사용되어 바인딩 플래그, 기타 플래그 및 CPU 액세스 플래그를 포함하여 리소스 속성 또는 힙 속성에서 유추되는 플래그를 재정의합니다.
D3D12_AUTO_BREADCRUMB_NODE 연결된 목록의 노드로 디바이스 DRED(제거된 확장 데이터) 자동 이동 경로 데이터를 나타냅니다.
D3D12_BARRIER_GROUP 지정된 형식의 장벽 그룹을 설명합니다.
D3D12_BARRIER_SUBRESOURCE_RANGE 논리적으로 인접한 하위 리소스 범위를 전환할 수 있습니다.
D3D12_BLEND_DESC 혼합 상태를 설명합니다. (D3D12_BLEND_DESC)

D3D12_BOX 3D 상자를 설명합니다.
D3D12_BUFFER_BARRIER 버퍼 메모리 액세스 장벽에 대해 설명합니다. 버퍼 장벽에서 리소스 메모리를 특정 액세스 유형에 대해 표시해야 하는 시기를 나타내는 데 사용됩니다.
D3D12_BUFFER_RTV 렌더링 대상 뷰에서 사용할 버퍼 리소스의 요소에 대해 설명합니다.
D3D12_BUFFER_SRV 셰이더 리소스 뷰에서 사용할 버퍼 리소스의 요소에 대해 설명합니다.
D3D12_BUFFER_UAV 순서가 지정되지 않은 액세스 뷰에서 사용할 버퍼의 요소에 대해 설명합니다. (D3D12_BUFFER_UAV)
D3D12_BUILD_RAYTRACING_ACCELERATION_STRUCTURE_DESC 광선 추적 가속 구조를 설명합니다. 이 구조를 ID3D12GraphicsCommandList4::BuildRaytracingAccelerationStructure에 전달하여 빌드할 가속 구조를 설명합니다.
D3D12_BUILD_RAYTRACING_ACCELERATION_STRUCTURE_INPUTS 광선 추적 가속 구조 빌드 작업에 대한 입력을 정의합니다. 이 구조체는 ID3D12GraphicsCommandList4::BuildRaytracingAccelerationStructure 및 ID3D12Device5::GetRaytracingAccelerationStructurePrebuildInfo에서 사용됩니다.
D3D12_BUILD_RAYTRACING_ACCELERATION_STRUCTURE_TOOLS_VISUALIZATION_HEADER 가속 구조 시각화의 GPU 메모리 레이아웃에 대해 설명합니다.
D3D12_CACHED_PIPELINE_STATE 파이프라인 상태를 저장합니다.
D3D12_CLEAR_VALUE 특정 리소스에 대한 명확한 작업을 최적화하는 데 사용되는 값을 설명합니다.
D3D12_COMMAND_QUEUE_DESC 명령 큐에 대해 설명합니다.

D3D12_COMMAND_SIGNATURE_DESC 명령 서명의 인수(매개 변수)를 설명합니다.
D3D12_COMPUTE_PIPELINE_STATE_DESC 컴퓨팅 파이프라인 상태 개체를 설명합니다.
D3D12_CONSTANT_BUFFER_VIEW_DESC 볼 상수 버퍼에 대해 설명합니다.
D3D12_CPU_DESCRIPTOR_HANDLE CPU 설명자 핸들을 설명합니다.
D3D12_DEBUG_COMMAND_LIST_GPU_BASED_VALIDATION_SETTINGS GPU-Based 유효성 검사에서 사용하는 명령 목록별 설정에 대해 설명합니다.
D3D12_DEBUG_DEVICE_GPU_BASED_VALIDATION_SETTINGS GPU-Based 유효성 검사에서 사용하는 설정에 대해 설명합니다.
D3D12_DEBUG_DEVICE_GPU_SLOWDOWN_PERFORMANCE_FACTOR 성능이 낮은 그래픽 어댑터를 시뮬레이션하기 위해 디버그 디바이스에서 삽입한 인공 속도 저하의 양을 설명합니다.
D3D12_DEPTH_STENCIL_DESC 깊이 스텐실 상태를 설명합니다. (D3D12_DEPTH_STENCIL_DESC)
D3D12_DEPTH_STENCIL_DESC1 깊이 스텐실 상태를 설명합니다. (D3D12_DEPTH_STENCIL_DESC1)
D3D12_DEPTH_STENCIL_VALUE 깊이 및 스텐실 값을 지정합니다.
D3D12_DEPTH_STENCIL_VIEW_DESC 깊이 스텐실 보기에서 액세스할 수 있는 텍스처의 하위 리소스에 대해 설명합니다.
D3D12_DEPTH_STENCILOP_DESC 스텐실 테스트 결과에 따라 수행할 수 있는 스텐실 작업에 대해 설명합니다.

D3D12_DESCRIPTOR_HEAP_DESC 설명자 힙에 대해 설명합니다.
D3D12_DESCRIPTOR_RANGE 설명자 범위를 설명합니다.
D3D12_DESCRIPTOR_RANGE1 플래그가 있는 설명자 범위를 설명하여 변동성을 확인합니다.
D3D12_DEVICE_REMOVED_EXTENDED_DATA DRED(디바이스 제거 확장 데이터) 버전 1.0 데이터를 나타냅니다.
D3D12_DEVICE_REMOVED_EXTENDED_DATA1 디바이스 DRED(제거된 확장 데이터) 버전 1.1 데이터를 나타냅니다.
D3D12_DISCARD_REGION 무시 리소스 작업에 대한 세부 정보를 설명합니다.
D3D12_DISPATCH_ARGUMENTS 컴퓨팅 셰이더에서 사용할 디스패치 매개 변수에 대해 설명합니다.
D3D12_DISPATCH_RAYS_DESC ID3D12GraphicsCommandList4::D ispatchRays 호출로 시작된 광선 디스패치 작업의 속성을 설명합니다.
D3D12_DRAW_ARGUMENTS 그리기 인스턴스에 대한 매개 변수를 설명합니다.
D3D12_DRAW_INDEXED_ARGUMENTS 인덱싱된 인스턴스를 그리기 위한 매개 변수를 설명합니다.
D3D12_DRED_ALLOCATION_NODE 연결된 목록의 노드로서 DRED(디바이스 제거 확장 데이터)에서 추적한 할당에 대한 데이터를 설명합니다.
D3D12_DRED_AUTO_BREADCRUMBS_OUTPUT 연결된 D3D12_AUTO_BREADCRUMB_NODE 개체 목록의 헤드에 대한 포인터를 포함합니다.

D3D12_DRED_PAGE_FAULT_OUTPUT 지정된 VA(가상 주소)의 GPU 페이지 오류와 관련된 할당 데이터에 대해 설명합니다.
D3D12_DXIL_LIBRARY_DESC 상태 개체에 포함할 수 있는 DXIL 라이브러리 상태 하위 개체에 대해 설명합니다.
D3D12_DXIL_SUBOBJECT_TO_EXPORTS_ASSOCIATION 이 하위 개체는 현재 릴리스에서 지원되지 않습니다.
D3D12_EXISTING_COLLECTION_DESC 상태 개체에 포함할 수 있는 기존 컬렉션을 설명하는 상태 하위 개체입니다.
D3D12_EXPORT_DESC DXIL 라이브러리 또는 컬렉션 상태 개체와 같은 상태 하위 개체에서 내보내기를 설명합니다.
D3D12_FEATURE_DATA_ARCHITECTURE 애플리케이션이 특정 어댑터 속성에 더 잘 최적화할 수 있도록 어댑터 아키텍처에 대한 세부 정보를 제공합니다.
D3D12_FEATURE_DATA_ARCHITECTURE1 애플리케이션이 특정 어댑터 속성에 더 잘 최적화할 수 있도록 각 어댑터의 아키텍처 세부 정보에 대한 세부 정보를 제공합니다.
D3D12_FEATURE_DATA_COMMAND_QUEUE_PRIORITY 다양한 명령 큐 유형의 우선 순위 지정에 대한 어댑터의 지원을 자세히 설명합니다.
D3D12_FEATURE_DATA_CROSS_NODE 여러 어댑터 간의 리소스 공유에 대한 지원 수준을 나타냅니다(예: 여러 GPU).
D3D12_FEATURE_DATA_D3D12_OPTIONS 현재 그래픽 드라이버의 Direct3D 12 기능 옵션에 대해 설명합니다.
D3D12_FEATURE_DATA_D3D12_OPTIONS1 HLSL 6.0 웨이브 작업에 대한 지원 수준을 설명합니다.

D3D12_FEATURE_DATA_D3D12_OPTIONS10 SUM 결합자를 사용할 수 있는지 여부와 메시 셰이더에서 *SV_ShadingRate* 설정할 수 있는지 여부를 나타냅니다.
D3D12_FEATURE_DATA_D3D12_OPTIONS11 설명자 힙의 리소스에서 64비트 정수 원자성 지원 여부를 나타냅니다.
D3D12_FEATURE_DATA_D3D12_OPTIONS12 향상된 장벽이 지원되는지 여부를 나타냅니다.
D3D12_FEATURE_DATA_D3D12_OPTIONS13 TBD
D3D12_FEATURE_DATA_D3D12_OPTIONS2 어댑터가 깊이 경계 테스트 및 프로그래밍 가능한 샘플 위치에 대해 제공하는 지원 수준을 나타냅니다.
D3D12_FEATURE_DATA_D3D12_OPTIONS3 어댑터가 타임스탬프 쿼리, 형식 캐스팅, 즉시 쓰기, 뷰 인스턴스화 및 barycentrics에 대해 제공하는 지원 수준을 나타냅니다.
D3D12_FEATURE_DATA_D3D12_OPTIONS4 64KB 정렬 MSAA 텍스처, 교차 API 공유 및 네이티브 16비트 셰이더 작업에 대한 지원 수준을 나타냅니다.
D3D12_FEATURE_DATA_D3D12_OPTIONS5 어댑터가 렌더링 패스, 광선 추적 및 셰이더 리소스 뷰 계층 3 타일식 리소스에 대해 제공하는 지원 수준을 나타냅니다.
D3D12_FEATURE_DATA_D3D12_OPTIONS6 어댑터가 VRS(가변 속도 음영)에 대해 제공하는 지원 수준을 나타내고 백그라운드 처리가 지원되는지 여부를 나타냅니다.
D3D12_FEATURE_DATA_D3D12_OPTIONS7 어댑터가 메시 및 증폭 셰이더 및 샘플러 피드백에 제공하는 지원 수준을 나타냅니다.
D3D12_FEATURE_DATA_D3D12_OPTIONS8 정렬되지 않은 블록 압축 텍스처가 지원되는지 여부를 나타냅니다.

D3D12_FEATURE_DATA_D3D12_OPTIONS9 메시 셰이더, 8개 이상의 *SV_RenderTargetArrayIndex* 값, 형식화된 리소스 64비트 정수 원자, 파생 및 파생 종속 텍스처 샘플 작업 및 WaveMMA(wave_matrix) 작업에 대한 지원 수준이 있는지 여부를 나타냅니다.
D3D12_FEATURE_DATA_DISPLAYABLE 이 기능은 현재 미리 보기로 제공됩니다.
D3D12_FEATURE_DATA_EXISTING_HEAPS 어댑터가 기존 시스템 메모리에서 힙 만들기를 지원하는지 여부에 대한 세부 정보를 제공합니다.
D3D12_FEATURE_DATA_FEATURE_LEVELS 현재 그래픽 드라이버에서 지원하는 기능 수준에 대한 정보를 설명합니다.
D3D12_FEATURE_DATA_FORMAT_INFO DXGI 데이터 형식 및 평면 수를 설명합니다.
D3D12_FEATURE_DATA_FORMAT_SUPPORT 지정된 형식에 대해 현재 그래픽 드라이버에서 지원하는 리소스를 설명합니다. (D3D12_FEATURE_DATA_FORMAT_SUPPORT)
D3D12_FEATURE_DATA_GPU_VIRTUAL_ADDRESS_SUPPORT 리소스당 최대 주소 비트 및 프로세스당 최대 주소 비트를 포함하여 어댑터의 GPU 가상 주소 공간 제한을 자세히 설명합니다.
D3D12_FEATURE_DATA_MULTISAMPLE_QUALITY_LEVELS 지정된 형식 및 샘플 수에 대한 다중 샘플링 이미지 품질 수준을 설명합니다.
D3D12_FEATURE_DATA_PROTECTED_RESOURCE_SESSION_SUPPORT 보호된 리소스 세션에 대한 지원 수준을 나타냅니다.
D3D12_FEATURE_DATA_PROTECTED_RESOURCE_SESSION_TYPE_COUNT 보호된 리소스 세션 유형의 수를 나타냅니다.
D3D12_FEATURE_DATA_PROTECTED_RESOURCE_SESSION_TYPES 보호된 리소스 세션 유형의 목록을 나타냅니다.

D3D12_FEATURE_DATA_QUERY_META_COMMAND 어댑터가 메타 명령에 대해 제공하는 지원 수준을 나타냅니다.
D3D12_FEATURE_DATA_ROOT_SIGNATURE 루트 서명 버전 지원을 나타냅니다.
D3D12_FEATURE_DATA_SERIALIZATION 힙 serialization에 대한 지원 수준을 나타냅니다.
D3D12_FEATURE_DATA_SHADER_CACHE 현재 그래픽 드라이버에서 지원되는 셰이더 캐싱 수준을 설명합니다. (D3D12_FEATURE_DATA_SHADER_CACHE)
D3D12_FEATURE_DATA_SHADER_MODEL 지원되는 셰이더 모델을 포함합니다.
D3D12_FUNCTION_DESC 함수를 설명합니다. (D3D12_FUNCTION_DESC)
D3D12_GLOBAL_BARRIER 리소스 메모리 액세스 장벽에 대해 설명합니다. 전역, 텍스처 및 버퍼 장벽에서 리소스 메모리를 특정 액세스 유형에 대해 표시해야 하는 시기를 나타내는 데 사용됩니다.
D3D12_GLOBAL_ROOT_SIGNATURE 연결된 셰이더와 함께 사용할 전역 루트 서명 상태 하위 개체를 정의합니다.
D3D12_GPU_DESCRIPTOR_HANDLE GPU 설명자 핸들을 설명합니다.
D3D12_GPU_VIRTUAL_ADDRESS_AND_STRIDE GPU 가상 주소 및 인덱싱 진행을 나타냅니다.
D3D12_GPU_VIRTUAL_ADDRESS_RANGE GPU 가상 주소 범위를 나타냅니다.
D3D12_GPU_VIRTUAL_ADDRESS_RANGE_AND_STRIDE GPU 가상 주소 범위 및 보폭을 나타냅니다.

D3D12_GRAPHICS_PIPELINE_STATE_DESC 그래픽 파이프라인 상태 개체를 설명합니다.
D3D12_HEAP_DESC 힙에 대해 설명합니다.
D3D12_HEAP_PROPERTIES 힙 속성에 대해 설명합니다.
D3D12_HIT_GROUP_DESC 상태 개체에 포함할 수 있는 광선 추적 적중 그룹 상태 하위 개체에 대해 설명합니다.
D3D12_INDEX_BUFFER_VIEW 볼 인덱스 버퍼에 대해 설명합니다.
D3D12_INDIRECT_ARGUMENT_DESC 명령 서명과 함께 사용할 간접 인수(간접 매개 변수)를 설명합니다.
D3D12_INFO_QUEUE_FILTER 디버그 메시지 필터; 에는 허용하거나 거부할 메시지 형식 목록이 포함되어 있습니다. (D3D12_INFO_QUEUE_FILTER)
D3D12_INFO_QUEUE_FILTER_DESC 필터를 통과하는 특정 유형의 메시지를 허용하거나 거부합니다. (D3D12_INFO_QUEUE_FILTER_DESC)
D3D12_INPUT_ELEMENT_DESC 그래픽 파이프라인의 입력 어셈블러 단계에 대한 단일 요소를 설명합니다.
D3D12_INPUT_LAYOUT_DESC 입력 어셈블러 단계의 입력 버퍼 데이터를 설명합니다.
D3D12_LIBRARY_DESC 라이브러리에 대해 설명합니다. (D3D12_LIBRARY_DESC)
D3D12_LOCAL_ROOT_SIGNATURE 연결된 셰이더와 함께 사용할 로컬 루트 서명 상태 하위 개체를 정의합니다.

D3D12_MEMCPY_DEST 메모리 복사 작업의 대상을 설명합니다.
D3D12_MESSAGE 정보 큐의 디버그 메시지입니다. (D3D12_MESSAGE)
D3D12_META_COMMAND_DESC 메타 명령에 대해 설명합니다.
D3D12_META_COMMAND_PARAMETER_DESC 메타 명령에 대한 매개 변수를 설명합니다.
D3D12_MIP_REGION 밉 영역의 크기를 설명합니다.
D3D12_NODE_MASK 상태 개체가 적용되는 GPU 노드를 식별하는 상태 하위 개체입니다.
D3D12_PACKED_MIP_INFO 밉맵이 있는 타일식 리소스의 타일 구조에 대해 설명합니다. (D3D12_PACKED_MIP_INFO)
D3D12_PARAMETER_DESC 함수 매개 변수를 설명합니다. (D3D12_PARAMETER_DESC)
D3D12_PIPELINE_STATE_STREAM_DESC 파이프라인 상태 스트림에 대해 설명합니다.
D3D12_PLACED_SUBRESOURCE_FOOTPRINT 오프셋 및 D3D12_SUBRESOURCE_FOOTPRINT 포함하여 배치된 하위 리소스의 공간을 설명합니다.
D3D12_PROTECTED_RESOURCE_SESSION_DESC 어댑터당 보호된 리소스 세션에 대한 플래그를 설명합니다.
D3D12_PROTECTED_RESOURCE_SESSION_DESC1 어댑터당 보호된 리소스 세션에 대한 플래그 및 보호 유형을 설명합니다.

D3D12_QUERY_DATA_PIPELINE_STATISTICS BeginQuery 및 EndQuery 호출 간의 그래픽 파이프라인 활동에 대한 정보를 쿼리합니다.
D3D12_QUERY_DATA_SO_STATISTICS 스트림 출력에 대한 쿼리 데이터를 설명합니다.
D3D12_QUERY_HEAP_DESC 쿼리 힙의 용도를 설명합니다. 쿼리 힙에는 개별 쿼리의 배열이 포함됩니다.
D3D12_RANGE 메모리 범위를 설명합니다.
D3D12_RANGE_UINT64 64비트 주소 공간의 메모리 범위를 설명합니다.
D3D12_RASTERIZER_DESC 래스터라이저 상태를 설명합니다. (D3D12_RASTERIZER_DESC)
D3D12_RAYTRACING_AABB 광선 추적 기하 도형으로 사용되는 AABB(축 맞춤 경계 상자)를 나타냅니다.
D3D12_RAYTRACING_ACCELERATION_STRUCTURE_POSTBUILD_INFO_COMPACTED_SIZE_DESC 압축 후 가속 구조에 대한 공간 요구 사항을 설명합니다.
D3D12_RAYTRACING_ACCELERATION_STRUCTURE_POSTBUILD_INFO_CURRENT_SIZE_DESC 현재 가속 구조에서 사용되는 공간을 설명합니다.
D3D12_RAYTRACING_ACCELERATION_STRUCTURE_POSTBUILD_INFO_DESC 가속 구조에서 생성할 빌드 후 정보에 대한 설명입니다. EmitRaytracingAccelerationStructurePostbuildInfo 및 BuildRaytracingAccelerationStructure 호출에서 이 구조를 사용합니다.
D3D12_RAYTRACING_ACCELERATION_STRUCTURE_POSTBUILD_INFO_SERIALIZATION_DESC 직렬화된 가속 구조체 및 헤더의 크기와 레이아웃을 설명합니다.
D3D12_RAYTRACING_ACCELERATION_STRUCTURE_POSTBUILD_INFO_TOOLS_VISUALIZATION_DESC 가속 구조를 도구로 시각화할 수 있는 폼으로 디코딩하기 위한 공간 요구 사항을 설명합니다.

D3D12_RAYTRACING_ACCELERATION_STRUCTURE_PREBUILD_INFO 광선 추적 가속 구조에 대한 사전 빌드 정보를 나타냅니다. GetRaytracingAccelerationStructurePrebuildInfo를 호출하여 이 구조체의 인스턴스를 가져옵니다.
D3D12_RAYTRACING_ACCELERATION_STRUCTURE_SRV 광선 추적 가속 구조를 저장하기 위한 SRV(셰이더 리소스 뷰) 구조체입니다.
D3D12_RAYTRACING_GEOMETRY_AABBS_DESC D3D12_BUILD_RAYTRACING_ACCELERATION_STRUCTURE_INPUTS 구조에서 광선 추적 가속 구조 빌드 작업에 입력 데이터를 제공하는 데 사용되는 축 맞춤 경계 상자 집합에 대해 설명합니다.
D3D12_RAYTRACING_GEOMETRY_DESC 광선 추적 가속 구조 빌드 작업에 입력 데이터를 제공하기 위해 D3D12_BUILD_RAYTRACING_ACCELERATION_STRUCTURE_INPUTS 구조에 사용되는 기하 도형 집합을 설명합니다.
D3D12_RAYTRACING_GEOMETRY_TRIANGLES_DESC 광선 추적 기하 도형으로 사용되는 삼각형 집합을 설명합니다. 이 구조체가 가리키는 기하 도형은 항상 삼각형 목록 형식으로 인덱싱되거나 인덱싱되지 않습니다. 삼각형 스트립은 지원되지 않습니다.
D3D12_RAYTRACING_INSTANCE_DESC 가속 구조 빌드 프로세스 중에 GPU 메모리에 사용되는 광선 추적 가속 구조의 인스턴스를 설명합니다.
D3D12_RAYTRACING_PIPELINE_CONFIG 광선 추적 파이프라인 구성을 나타내는 상태 하위 개체입니다.
D3D12_RAYTRACING_PIPELINE_CONFIG1 플래그가 있는 광선 추적 파이프라인 구성을 나타내는 상태 하위 개체입니다.
D3D12_RAYTRACING_SHADER_CONFIG 셰이더 구성을 나타내는 상태 하위 개체입니다.
D3D12_RENDER_PASS_BEGINNING_ACCESS 렌더링 패스로 전환할 때 애플리케이션에서 요청한 리소스에 대한 액세스를 설명합니다.
D3D12_RENDER_PASS_BEGINNING_ACCESS_CLEAR_PARAMETERS 렌더링 패스의 시작 부분에서 리소스를 지워야 하는 명확한 값을 설명합니다.

D3D12_RENDER_PASS_DEPTH_STENCIL_DESC DSV(깊이 스텐실 뷰)에 대한 바인딩(렌더링 패스 기간 동안 고정됨)과 시작 및 끝 액세스 특성에 대해 설명합니다.
D3D12_RENDER_PASS_ENDING_ACCESS 렌더링 패스에서 전환할 때 애플리케이션에서 요청한 리소스에 대한 액세스를 설명합니다.
D3D12_RENDER_PASS_ENDING_ACCESS_RESOLVE_PARAMETERS 렌더링 패스가 끝날 때 확인할 리소스에 대해 설명합니다.
D3D12_RENDER_PASS_ENDING_ACCESS_RESOLVE_SUBRESOURCE_PARAMETERS 렌더링 패스가 끝날 때 해결에 관련된 하위 리소스에 대해 설명합니다.
D3D12_RENDER_PASS_RENDER_TARGET_DESC 하나 이상의 RTV(렌더링 대상 뷰)에 대한 바인딩(렌더링 패스 기간 동안 고정됨)과 시작 및 끝 액세스 특성에 대해 설명합니다.
D3D12_RENDER_TARGET_BLEND_DESC 렌더링 대상의 혼합 상태를 설명합니다. (D3D12_RENDER_TARGET_BLEND_DESC)
D3D12_RENDER_TARGET_VIEW_DESC 렌더링 대상 뷰를 사용하여 액세스할 수 있는 리소스의 하위 리소스에 대해 설명합니다.
D3D12_RESOURCE_ALIASING_BARRIER 동일한 힙에 매핑된 두 개의 서로 다른 리소스 사용 간의 전환에 대해 설명합니다.
D3D12_RESOURCE_ALLOCATION_INFO 리소스를 할당하는 데 필요한 매개 변수에 대해 설명합니다.
D3D12_RESOURCE_ALLOCATION_INFO1 오프셋을 포함하여 리소스를 할당하는 데 필요한 매개 변수에 대해 설명합니다.
D3D12_RESOURCE_BARRIER 리소스 장벽(리소스 사용 전환)을 설명합니다.
D3D12_RESOURCE_DESC 텍스처와 같은 리소스를 설명합니다. 이 구조체는 광범위하게 사용됩니다.

D3D12_RESOURCE_DESC1 밉 영역을 포함하여 텍스처와 같은 리소스에 대해 설명합니다. 이 구조체는 여러 메서드에서 사용됩니다.
D3D12_RESOURCE_TRANSITION_BARRIER 서로 다른 사용량 간의 하위 리소스 전환에 대해 설명합니다.
D3D12_RESOURCE_UAV_BARRIER 향후 UAV 액세스를 시작하기 전에 모든 UAV 액세스가 완료되어야 하는 리소스를 나타냅니다.
D3D12_ROOT_CONSTANTS 셰이더에 하나의 상수 버퍼로 표시되는 루트 서명의 상수 인라인을 설명합니다.
D3D12_ROOT_DESCRIPTOR 셰이더에 표시되는 루트 서명 버전 1.0의 설명자를 인라인으로 설명합니다.
D3D12_ROOT_DESCRIPTOR_TABLE 설명자 테이블의 루트 서명 1.0 레이아웃을 단일 기본 설명자 핸들을 기준으로 하는 설명자 범위의 컬렉션으로 설명합니다.
D3D12_ROOT_DESCRIPTOR_TABLE1 설명자 테이블의 루트 서명 1.1 레이아웃을 단일 기본 설명자 핸들을 기준으로 하는 설명자 범위의 컬렉션으로 설명합니다.
D3D12_ROOT_DESCRIPTOR1 셰이더에 표시되는 루트 서명 버전 1.1의 설명자를 인라인으로 설명합니다.
D3D12_ROOT_PARAMETER 루트 서명 버전 1.0의 슬롯에 대해 설명합니다.
D3D12_ROOT_PARAMETER1 루트 서명 버전 1.1의 슬롯에 대해 설명합니다.
D3D12_ROOT_SIGNATURE_DESC 루트 서명 버전 1.0의 레이아웃을 설명합니다.

D3D12_ROOT_SIGNATURE_DESC1 루트 서명 버전 1.1의 레이아웃을 설명합니다.
D3D12_RT_FORMAT_ARRAY 렌더링 대상 형식의 배열을 래핑합니다.
D3D12_SAMPLE_POSITION 프로그래밍 가능한 샘플 위치와 함께 사용할 하위 픽셀 샘플 위치를 설명합니다.
D3D12_SAMPLER_DESC 샘플러 상태를 설명합니다. (D3D12_SAMPLER_DESC)
D3D12_SERIALIZED_DATA_DRIVER_MATCHING_IDENTIFIER 직렬화된 가속 구조에 대한 드라이버 버전 관리를 설명하는 불투명 데이터 구조입니다.
D3D12_SERIALIZED_RAYTRACING_ACCELERATION_STRUCTURE_HEADER 직렬화된 광선 추적 가속 구조체의 헤더를 정의합니다.
D3D12_SHADER_BUFFER_DESC 셰이더 상수 버퍼에 대해 설명합니다. (D3D12_SHADER_BUFFER_DESC)
D3D12_SHADER_BYTECODE 셰이더 데이터를 설명합니다. (D3D12_SHADER_BYTECODE)
D3D12_SHADER_CACHE_SESSION_DESC 셰이더 캐시 세션에 대해 설명합니다.
D3D12_SHADER_DESC 셰이더에 대해 설명합니다. (D3D12_SHADER_DESC)
D3D12_SHADER_INPUT_BIND_DESC 셰이더 리소스가 셰이더 입력에 바인딩되는 방법을 설명합니다. (D3D12_SHADER_INPUT_BIND_DESC)
D3D12_SHADER_RESOURCE_VIEW_DESC 셰이더 리소스 뷰에 대해 설명합니다. (D3D12_SHADER_RESOURCE_VIEW_DESC)

D3D12_SHADER_TYPE_DESC 셰이더 변수 형식에 대해 설명합니다. (D3D12_SHADER_TYPE_DESC)
D3D12_SHADER_VARIABLE_DESC 셰이더 변수에 대해 설명합니다. (D3D12_SHADER_VARIABLE_DESC)
D3D12_SIGNATURE_PARAMETER_DESC 셰이더 서명을 설명합니다. (D3D12_SIGNATURE_PARAMETER_DESC)
D3D12_SO_DECLARATION_ENTRY 출력 슬롯의 꼭짓점 버퍼에 있는 꼭짓점 요소에 대해 설명합니다.
D3D12_STATE_OBJECT_CONFIG 상태 개체의 일반 속성을 정의합니다.
D3D12_STATE_OBJECT_DESC 상태 개체에 대한 설명입니다. 이 구조를 ID3D12Device::CreateStateObject에 전달합니다.
D3D12_STATE_SUBOBJECT 상태 개체 설명 내의 하위 개체를 나타냅니다. D3D12_STATE_OBJECT_DESC 함께 사용합니다.
D3D12_STATIC_SAMPLER_DESC 정적 샘플러에 대해 설명합니다.
D3D12_STREAM_OUTPUT_BUFFER_VIEW 스트림 출력 버퍼에 대해 설명합니다.
D3D12_STREAM_OUTPUT_DESC 스트리밍 출력 버퍼에 대해 설명합니다.
D3D12_SUBOBJECT_TO_EXPORTS_ASSOCIATION 상태 개체에 직접 정의된 하위 개체를 셰이더 내보내기와 연결합니다.
D3D12_SUBRESOURCE_DATA 하위 리소스 데이터를 설명합니다. (D3D12_SUBRESOURCE_DATA)

D3D12_SUBRESOURCE_FOOTPRINT 부모 리소스에 대한 하위 리소스의 형식, 너비, 높이, 깊이 및 행 피치에 대해 설명합니다.
D3D12_SUBRESOURCE_INFO 하위 리소스 데이터를 설명합니다. (D3D12_SUBRESOURCE_INFO)
D3D12_SUBRESOURCE_RANGE_UINT64 하위 리소스 메모리 범위에 대해 설명합니다.
D3D12_SUBRESOURCE_TILING 타일형 하위 리소스 볼륨에 대해 설명합니다. (D3D12_SUBRESOURCE_TILING)
D3D12_TEX1D_ARRAY_DSV 깊이 스텐실 보기에서 사용할 1D 텍스처 배열의 하위 리소스에 대해 설명합니다.
D3D12_TEX1D_ARRAY_RTV 렌더링 대상 보기에서 사용할 1D 텍스처 배열의 하위 리소스에 대해 설명합니다.
D3D12_TEX1D_ARRAY_SRV 셰이더 리소스 뷰에서 사용할 1D 텍스처 배열의 하위 리소스에 대해 설명합니다.
D3D12_TEX1D_ARRAY_UAV 순서가 지정되지 않은 액세스 1D 텍스처 리소스의 배열에 대해 설명합니다. (D3D12_TEX1D_ARRAY_UAV)
D3D12_TEX1D_DSV 깊이 스텐실 보기에 액세스할 수 있는 1D 텍스처의 하위 리소스에 대해 설명합니다.
D3D12_TEX1D_RTV 렌더링 대상 보기에서 사용할 1D 텍스처의 하위 리소스에 대해 설명합니다.
D3D12_TEX1D_SRV 셰이더 리소스 뷰에서 사용할 1D 텍스처의 하위 리소스를 지정합니다. (D3D12_TEX1D_SRV)
D3D12_TEX1D_UAV 순서가 지정되지 않은 액세스 1D 텍스처 리소스에 대해 설명합니다. (D3D12_TEX1D_UAV)

D3D12_TEX2D_ARRAY_DSV 깊이 스텐실 보기에 액세스할 수 있는 2D 텍스처 배열의 하위 리소스에 대해 설명합니다.
D3D12_TEX2D_ARRAY_RTV 렌더링 대상 보기에서 사용할 2D 텍스처 배열의 하위 리소스에 대해 설명합니다. (D3D12_TEX2D_ARRAY_RTV)
D3D12_TEX2D_ARRAY_SRV 셰이더 리소스 뷰에서 사용할 2D 텍스처 배열의 하위 리소스에 대해 설명합니다. (D3D12_TEX2D_ARRAY_SRV)
D3D12_TEX2D_ARRAY_UAV 순서가 지정되지 않은 액세스 2D 텍스처 리소스의 배열을 설명합니다. (D3D12_TEX2D_ARRAY_UAV)
D3D12_TEX2D_DSV 깊이 스텐실 보기에 액세스할 수 있는 2D 텍스처의 하위 리소스에 대해 설명합니다.
D3D12_TEX2D_RTV 렌더링 대상 보기에서 사용할 2D 텍스처의 하위 리소스에 대해 설명합니다. (D3D12_TEX2D_RTV)
D3D12_TEX2D_SRV 셰이더 리소스 뷰에서 사용할 2D 텍스처의 하위 리소스에 대해 설명합니다. (D3D12_TEX2D_SRV)
D3D12_TEX2D_UAV 순서가 지정되지 않은 액세스 2D 텍스처 리소스에 대해 설명합니다. (D3D12_TEX2D_UAV)
D3D12_TEX2DMS_ARRAY_DSV 깊이 스텐실 보기에 대한 다중 샘플링된 2D 텍스처 배열의 하위 리소스에 대해 설명합니다.
D3D12_TEX2DMS_ARRAY_RTV 렌더링 대상 보기에서 사용할 다중 샘플링된 2D 텍스처 배열의 하위 리소스에 대해 설명합니다.
D3D12_TEX2DMS_ARRAY_SRV 셰이더 리소스 뷰에서 사용할 다중 샘플링된 2D 텍스처 배열의 하위 리소스에 대해 설명합니다.
D3D12_TEX2DMS_DSV 깊이 스텐실 보기에 액세스할 수 있는 다중 샘플링된 2D 텍스처의 하위 리소스에 대해 설명합니다.

D3D12_TEX2DMS_RTV 렌더링 대상 보기에서 사용할 다중 샘플링된 2D 텍스처의 하위 리소스에 대해 설명합니다.
D3D12_TEX2DMS_SRV 셰이더 리소스 뷰에서 사용할 다중 샘플링된 2D 텍스처의 하위 리소스에 대해 설명합니다.
D3D12_TEX3D_RTV 렌더링 대상 보기에서 사용할 3D 텍스처의 하위 리소스에 대해 설명합니다.
D3D12_TEX3D_SRV 셰이더 리소스 뷰에서 사용할 3D 텍스처의 하위 리소스에 대해 설명합니다.
D3D12_TEX3D_UAV 순서가 지정되지 않은 액세스 3D 텍스처 리소스에 대해 설명합니다. (D3D12_TEX3D_UAV)
D3D12_TEXCUBE_ARRAY_SRV 셰이더 리소스 뷰에서 사용할 큐브 텍스처 배열의 하위 리소스에 대해 설명합니다.
D3D12_TEXCUBE_SRV 셰이더 리소스 뷰에서 사용할 큐브 텍스처의 하위 리소스에 대해 설명합니다.
D3D12_TEXTURE_BARRIER 텍스처에 대한 액세스 전환을 표현합니다.
D3D12_TEXTURE_COPY_LOCATION 텍스처 복사본을 위해 텍스처의 일부를 설명합니다.
D3D12_TILE_REGION_SIZE 타일 영역의 크기를 설명합니다. (D3D12_TILE_REGION_SIZE)
D3D12_TILE_SHAPE 해당 차원을 지정하여 타일의 모양을 설명합니다. (D3D12_TILE_SHAPE)
D3D12_TILED_RESOURCE_COORDINATE 타일형 리소스의 좌표를 설명합니다. (D3D12_TILED_RESOURCE_COORDINATE)

D3D12_UNORDERED_ACCESS_VIEW_DESC 순서가 지정되지 않은 액세스 보기를 사용하여 액세스할 수 있는 리소스의 하위 리소스에 대해 설명합니다.
D3D12_VERSIONED_DEVICE_REMOVED_EXTENDED_DATA 버전이 지정된 DRED(디바이스 제거 확장 데이터) 데이터를 나타냅니다.
D3D12_VERSIONED_ROOT_SIGNATURE_DESC 루트 서명 설명의 모든 버전을 보유하며 serialization/deserialization 함수와 함께 사용하도록 설계되었습니다.
D3D12_VERTEX_BUFFER_VIEW 꼭짓점 버퍼 뷰에 대해 설명합니다.
D3D12_VIEW_INSTANCE_LOCATION 뷰 인스턴스와 연결된 뷰포트/스텐실 및 렌더링 대상을 지정합니다.
D3D12_VIEW_INSTANCING_DESC 뷰 인스턴스화 구성 중에 사용되는 매개 변수를 지정합니다.
D3D12_VIEWPORT 뷰포트의 차원에 대해 설명합니다.
D3D12_WRITEBUFFERIMMEDIATE_PARAMETER ID3D12CommandList2::WriteBufferImmediate를 사용하여 작성된 즉각적인 값 및 대상 주소를 지정합니다.

Direct3D 9 그래픽

아티클2023. 08. 22.

Direct3D 9 그래픽 기술 개요입니다.

Direct3D 9 그래픽을 개발하려면 다음 헤더가 필요합니다.

- d3d9.h
- d3d9caps.h
- d3d9helper.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Direct3D 9 그래픽

Functions

AddDirtyBox IDirect3DVolumeTexture9::AddDirtyBox 메서드(d3d9helper.h)는 볼륨 텍스처 리소스에 더티 영역을 추가합니다.
AddDirtyBox IDirect3DVolumeTexture9::AddDirtyBox(d3d9.h) 메서드는 볼륨 텍스처 리소스에 더티 영역을 추가합니다.
AddDirtyRect IDirect3DCubeTexture9::AddDirtyRect 메서드(d3d9.h)는 큐브 텍스처 리소스에 더티 영역을 추가합니다.
AddDirtyRect IDirect3DTexture9::AddDirtyRect 메서드(d3d9.h)는 텍스처 리소스에 더티 영역을 추가합니다.
AddDirtyRect IDirect3DCubeTexture9::AddDirtyRect는 큐브 텍스처 리소스에 더티 영역을 추가합니다.
AddDirtyRect IDirect3DTexture9::AddDirtyRect(d3d9.h) 메서드는 텍스처 리소스에 더티 영역을 추가합니다.

적용 IDirect3DStateBlock9::Apply 메서드(d3d9helper.h)는 상태 블록을 현재 디바이스 상태에 적용합니다.
적용 IDirect3DStateBlock9::Apply(d3d9.h) 메서드는 상태 블록을 현재 디바이스 상태에 적용합니다.
BeginScene IDirect3DDevice9::BeginScene 메서드(d3d9.h)가 장면을 시작합니다.
BeginScene IDirect3DDevice9::BeginScene 메서드(d3d9.h)가 장면을 시작합니다.
BeginStateBlock IDirect3DDevice9::BeginStateBlock 메서드(d3d9.h)는 Direct3D에 디바이스 상태 블록 기록을 시작하도록 신호를 보냅니다.
BeginStateBlock IDirect3DDevice9::BeginStateBlock 메서드(d3d9.h)는 Direct3D에 디바이스 상태 블록 기록을 시작하도록 신호를 보냅니다.
캡처 IDirect3DStateBlock9::Capture 메서드(d3d9helper.h)는 stateblock에 포함된 상태의 현재 값을 캡처합니다.
캡처 IDirect3DStateBlock9::Capture(d3d9.h) 메서드는 stateblock에 포함된 상태의 현재 값을 캡처합니다.
CheckDepthStencilMatch IDirect3D9::CheckDepthStencilMatch(d3d9helper.h) 메서드는 깊이 스텐실 형식이 렌더링 대상 형식과 호환되는지 여부를 결정합니다.
CheckDepthStencilMatch IDirect3D9::CheckDepthStencilMatch 메서드(d3d9helper.h)는 깊이 스텐실 형식이 특정 표시 모드의 렌더링 대상 형식과 호환되는지 여부를 결정합니다.
CheckDeviceFormat

IDirect3D9::CheckDeviceFormat(d3d9helper.h) 메서드는 표면 형식을 지정된 리소스 유형으로 사용할 수 있는지 여부를 결정합니다.
CheckDeviceFormat IDirect3D9::CheckDeviceFormat 메서드(d3d9helper.h)는 표면 형식을 지정된 리소스 유형으로 사용할 수 있는지 여부를 결정합니다.
CheckDeviceFormatConversion IDirect3D9::CheckDeviceFormatConversion(d3d9helper.h) 메서드는 디바이스를 테스트하여 한 디스플레이 형식에서 다른 디스플레이 형식으로의 변환을 지원하는지 확인합니다.
CheckDeviceFormatConversion IDirect3D9::CheckDeviceFormatConversion 메서드(d3d9helper.h)는 디바이스를 테스트하여 한 디스플레이 형식에서 다른 디스플레이 형식으로의 변환을 지원하는지 확인합니다.
CheckDeviceMultiSampleType IDirect3D9::CheckDeviceMultiSampleType(d3d9helper.h) 메서드는 이 디바이스에서 다중 샘플링 기술을 사용할 수 있는지 여부를 결정합니다.
CheckDeviceMultiSampleType IDirect3D9::CheckDeviceMultiSampleType 메서드(d3d9.h)는 이 디바이스에서 다중 샘플링 기술을 사용할 수 있는지 여부를 결정합니다.
CheckDeviceState 창 또는 전체 화면 애플리케이션에 대한 Direct3D 디바이스의 현재 협력 수준 상태 보고합니다. (IDirect3DDevice9Ex.CheckDeviceState)
CheckDeviceType IDirect3D9::CheckDeviceType(d3d9helper.h) 메서드는 이 어댑터에서 하드웨어 가속 디바이스 유형을 사용할 수 있는지 여부를 확인합니다.
CheckDeviceType IDirect3D9::CheckDeviceType 메서드(d3d9.h)는 이 어댑터에서 하드웨어 가속 디바이스 유형을 사용할 수 있는지 여부를 확인합니다.
CheckResourceResidency 리소스 배열을 확인하여 시스템이 리소스 GPU에 액세스할 수 있도록 해야 하기 때문에 그리기 시 큰 중단이 발생할 가능성이 있는지 확인합니다.
지우기

IDirect3DDevice9::Clear 메서드(d3d9.h)는 렌더링 대상, 여러 렌더링 대상, 스텐실 버퍼 또는 깊이 버퍼와 같은 하나 이상의 표면을 지웁니다.
지우기 IDirect3DDevice9::Clear 메서드(d3d9.h)는 렌더링 대상, 여러 렌더링 대상, 스텐실 버퍼 또는 깊이 버퍼와 같은 하나 이상의 표면을 지웁니다.
ColorFill IDirect3DDevice9::ColorFill 메서드(d3d9.h)를 사용하면 애플리케이션이 지정된 색으로 D3DPOOL_DEFAULT 표면의 사각형 영역을 채울 수 있습니다.
ColorFill IDirect3DDevice9::ColorFill 메서드(d3d9.h)를 사용하면 애플리케이션이 지정된 색으로 D3DPOOL_DEFAULT 표면의 사각형 영역을 채울 수 있습니다.
ComposeRects 다른 표면의 문자 모양 알파벳을 사용하여 텍스트 문자열을 한 화면에 복사합니다. 컴퍼지션은 비트 연산을 사용하여 GPU에서 수행됩니다.
CreateAdditionalSwapChain IDirect3DDevice9::CreateAdditionalSwapChain 메서드(d3d9.h)는 여러 보기를 렌더링하기 위한 추가 스왑 체인을 만듭니다.
CreateAdditionalSwapChain IDirect3DDevice9::CreateAdditionalSwapChain 메서드(d3d9.h)는 여러 보기를 렌더링하기 위한 추가 스왑 체인을 만듭니다.
CreateCubeTexture IDirect3DDevice9::CreateCubeTexture 메서드(d3d9helper.h)는 큐브 텍스처 리소스를 만듭니다.
CreateCubeTexture IDirect3DDevice9::CreateCubeTexture 메서드(d3d9.h)는 큐브 텍스처 리소스를 만듭니다.
CreateDepthStencilSurface IDirect3DDevice9::CreateDepthStencilSurface 메서드(d3d9helper.h)는 깊이 스텐실 리소스를 만듭니다.
CreateDepthStencilSurface IDirect3DDevice9::CreateDepthStencilSurface 메서드(d3d9.h)는 깊이 스텐실 리소스를 만듭니다.

CreateDepthStencilSurfaceEx 깊이 스텐실 표면을 만듭니다.
CreateDevice IDirect3D9::CreateDevice(d3d9helper.h) 메서드는 디스플레이 어댑터를 나타내는 디바이스를 만듭니다.
CreateDevice IDirect3D9::CreateDevice 메서드(d3d9.h)는 디스플레이 어댑터를 나타내는 디바이스를 만듭니다.
CreateDeviceEx 디스플레이 어댑터를 나타내는 디바이스를 만듭니다. (IDirect3D9Ex.CreateDeviceEx)
CreateIndexBuffer IDirect3DDevice9::CreateIndexBuffer 메서드(d3d9helper.h)는 인덱스 버퍼를 만듭니다.
CreateIndexBuffer IDirect3DDevice9::CreateIndexBuffer 메서드(d3d9.h)는 인덱스 버퍼를 만듭니다.
CreateOffscreenPlainSurface IDirect3DDevice9::CreateOffscreenPlainSurface 메서드(d3d9helper.h)는 화면 외부 화면을 만듭니다.
CreateOffscreenPlainSurface IDirect3DDevice9::CreateOffscreenPlainSurface 메서드(d3d9.h)는 화면 외부 화면을 만듭니다.
CreateOffscreenPlainSurfaceEx 화면 외부 화면을 만듭니다. (IDirect3DDevice9Ex.CreateOffscreenPlainSurfaceEx)
CreatePixelShader IDirect3DDevice9::CreatePixelShader 메서드(d3d9.h)는 픽셀 셰이더를 만듭니다.
CreatePixelShader IDirect3DDevice9::CreatePixelShader 메서드(d3d9.h)는 픽셀 셰이더를 만듭니다.
CreateQuery IDirect3DDevice9::CreateQuery 메서드(d3d9helper.h)는 상태 쿼리를 만듭니다.

CreateQuery IDirect3DDevice9::CreateQuery 메서드(d3d9.h)는 상태 쿼리를 만듭니다.
CreateRenderTarget IDirect3DDevice9::CreateRenderTarget 메서드(d3d9helper.h)는 렌더링 대상 표면을 만듭니다.
CreateRenderTarget IDirect3DDevice9::CreateRenderTarget 메서드(d3d9.h)는 렌더링 대상 표면을 만듭니다.
CreateRenderTargetEx 렌더링 대상 표면을 만듭니다. (IDirect3DDevice9Ex.CreateRenderTargetEx)
CreateStateBlock IDirect3DDevice9::CreateStateBlock 메서드(d3d9helper.h)는 모든 디바이스 상태, 꼭짓점 관련 상태 또는 픽셀 관련 상태에 대한 값을 포함하는 새 상태 블록을 만듭니다.
CreateStateBlock IDirect3DDevice9::CreateStateBlock 메서드(d3d9.h)는 모든 디바이스 상태, 꼭짓점 관련 상태 또는 픽셀 관련 상태에 대한 값을 포함하는 새 상태 블록을 만듭니다.
CreateTexture IDirect3DDevice9::CreateTexture 메서드(d3d9helper.h)는 텍스처 리소스를 만듭니다.
CreateTexture IDirect3DDevice9::CreateTexture 메서드(d3d9.h)는 텍스처 리소스를 만듭니다.
CreateVertexBuffer IDirect3DDevice9::CreateVertexBuffer 메서드(d3d9helper.h)는 꼭짓점 버퍼를 만듭니다.
CreateVertexBuffer IDirect3DDevice9::CreateVertexBuffer 메서드(d3d9.h)는 꼭짓점 버퍼를 만듭니다.
CreateVertexDeclaration IDirect3DDevice9::CreateVertexDeclaration 메서드(d3d9helper.h)는 디바이스 및 꼭짓점 요소에서 꼭짓점 셰이더 선언을 만듭니다.
CreateVertexDeclaration

IDirect3DDevice9::CreateVertexDeclaration 메서드(d3d9.h)는 디바이스 및 꼭짓점 요소에서 꼭짓점 셰이더 선언을 만듭니다.
CreateVertexShader IDirect3DDevice9::CreateVertexShader 메서드(d3d9helper.h)는 꼭짓점 셰이더를 만듭니다.
CreateVertexShader IDirect3DDevice9::CreateVertexShader 메서드(d3d9.h)는 꼭짓점 셰이더를 만듭니다.
CreateVolumeTexture IDirect3DDevice9::CreateVolumeTexture 메서드(d3d9helper.h)는 볼륨 텍스처 리소스를 만듭니다.
CreateVolumeTexture IDirect3DDevice9::CreateVolumeTexture 메서드(d3d9.h)는 볼륨 텍스처 리소스를 만듭니다.
DeletePatch IDirect3DDevice9::D eletePatch 메서드(d3d9helper.h)는 캐시된 고차 패치를 해제합니다.
DeletePatch IDirect3DDevice9::D eletePatch 메서드(d3d9.h)는 캐시된 고차 패치를 해제합니다.
Direct3DCreate9 Direct3DCreate9 함수(d3d9helper.h)는 IDirect3D9 개체를 만들고 인터페이스를 반환합니다.
Direct3DCreate9 Direct3DCreate9Ex 함수(d3d9.h)는 IDirect3D9 개체를 만들고 인터페이스를 반환합니다.
Direct3DCreate9Ex IDirect3D9Ex 개체를 만들고 해당 개체에 대한 인터페이스를 반환합니다.
DrawIndexedPrimitive IDirect3DDevice9::D rawIndexedPrimitive 메서드(d3d9helper.h)는 지정된 기하학적 기본 형식을 꼭짓점 배열로 렌더링합니다.
DrawIndexedPrimitive IDirect3DDevice9::D rawIndexedPrimitive 메서드(d3d9.h)는 지정된 기하학적 기본 형식을 꼭짓점 배열로 렌더링합니다.

DrawIndexedPrimitiveUP IDirect3DDevice9::D rawIndexedPrimitiveUP 메서드(d3d9helper.h)는 사용자 메모리 포인터로 지정된 데이터로 지정된 기하학적 기본 형식을 렌더링합니다.
DrawIndexedPrimitiveUP IDirect3DDevice9::D rawIndexedPrimitiveUP 메서드(d3d9.h)는 사용자 메모리 포인터로 지정된 데이터로 지정된 기하학적 기본 형식을 렌더링합니다.
DrawPrimitive IDirect3DDevice9::D rawPrimitive 메서드(d3d9helper.h)는 현재 데이터 입력 스트림 집합에서 지정된 형식의 인덱싱되지 않은 기하학적 기본 형식 시퀀스를 렌더링합니다.
DrawPrimitive IDirect3DDevice9::D rawPrimitive 메서드(d3d9.h)는 현재 데이터 입력 스트림 집합에서 지정된 형식의 인덱싱되지 않은 기하학적 기본 형식 시퀀스를 렌더링합니다.
DrawPrimitiveUP IDirect3DDevice9::D rawPrimitiveUP 메서드(d3d9helper.h)는 사용자 메모리 포인터가 지정한 데이터를 지정된 형식의 기하학적 기본 형식 시퀀스로 렌더링합니다.
DrawPrimitiveUP IDirect3DDevice9::D rawPrimitiveUP 메서드(d3d9.h)는 사용자 메모리 포인터에 지정된 데이터를 지정된 형식의 기하학적 기본 형식 시퀀스로 렌더링합니다.
DrawRectPatch IDirect3DDevice9::D rawRectPatch 메서드(d3d9helper.h)는 현재 설정된 스트림을 사용하여 사각형 패치를 그립니다.
DrawRectPatch IDirect3DDevice9::D rawRectPatch 메서드(d3d9.h)는 현재 설정된 스트림을 사용하여 사각형 패치를 그립니다.
DrawTriPatch IDirect3DDevice9::D rawTriPatch 메서드(d3d9.h)는 현재 설정된 스트림을 사용하여 삼각형 패치를 그립니다.
DrawTriPatch IDirect3DDevice9::D rawTriPatch 메서드(d3d9.h)는 현재 설정된 스트림을 사용하여 삼각형 패치를 그립니다.

EndScene IDirect3DDevice9::EndScene 메서드(d3d9.h)는 IDirect3DDevice9::BeginScene을 호출하여 시작된 장면을 종료합니다.
EndScene IDirect3DDevice9::EndScene 메서드(d3d9.h)는 IDirect3DDevice9::BeginScene을 호출하여 시작된 장면을 종료합니다.
EndStateBlock IDirect3DDevice9::EndStateBlock 메서드(d3d9.h)는 Direct3D에 디바이스 상태 블록 기록을 중지하고 상태 블록 인터페이스에 대한 포인터를 검색하도록 신호를 보냅니다.
EndStateBlock IDirect3DDevice9::EndStateBlock 메서드(d3d9.h)는 Direct3D에 디바이스 상태 블록 기록을 중지하고 상태 블록 인터페이스에 대한 포인터를 검색하도록 신호를 보냅니다.
EnumAdapterModes IDirect3D9::EnumAdapterModes(d3d9helper.h) 메서드는 디바이스를 쿼리하여 지정된 어댑터가 요청된 형식 및 디스플레이 모드를 지원하는지 여부를 확인합니다.
EnumAdapterModes IDirect3D9::EnumAdapterModes 메서드(d3d9.h)는 디바이스를 쿼리하여 지정된 어댑터가 요청된 형식 및 표시 모드를 지원하는지 여부를 확인합니다.
EnumAdapterModesEx 이 메서드는 지정된 모드 인덱스 기반의 실제 표시 모드 정보를 반환합니다.
EvictManagedResources IDirect3DDevice9::EvictManagedResources 메서드(d3d9.h)는 Direct3D 및 드라이버 관리 리소스를 포함하여 모든 관리되는 리소스를 제거합니다.
EvictManagedResources IDirect3DDevice9::EvictManagedResources 메서드(d3d9.h)는 Direct3D 및 드라이버 관리 리소스를 포함하여 모든 관리되는 리소스를 제거합니다.
FreePrivateData IDirect3DResource9::FreePrivateData 메서드(d3d9helper.h)는 이 리소스와 연결된 지정된 프라이빗 데이터를 해제합니다.

FreePrivateData IDirect3DVolume9::FreePrivateData 메서드(d3d9.h)는 이 볼륨과 연결된 지정된 프라이빗 데이터를 해제합니다.
FreePrivateData IDirect3DResource9::FreePrivateData(d3d9.h) 메서드는 이 리소스와 연결된 지정된 프라이빗 데이터를 해제합니다.
FreePrivateData IDirect3DVolume9::FreePrivateData(d3d9.h) 메서드는 이 볼륨과 연결된 지정된 프라이빗 데이터를 해제합니다.
GenerateMipSubLevels IDirect3DBaseTexture9::GenerateMipSubLevels(d3d9helper.h) 메서드는 mipmap sublevels를 생성합니다.
GenerateMipSubLevels IDirect3DBaseTexture9::GenerateMipSubLevels 메서드(d3d9helper.h)는 mipmap sublevels를 생성합니다.
GetAdapterCount IDirect3D9::GetAdapterCount(d3d9helper.h) 메서드는 시스템의 어댑터 수를 반환합니다.
GetAdapterCount IDirect3D9::GetAdapterCount 메서드(d3d9.h)는 시스템의 어댑터 수를 반환합니다.
GetAdapterDisplayMode IDirect3D9::GetAdapterDisplayMode(d3d9helper.h) 메서드는 어댑터의 현재 표시 모드를 검색합니다.
GetAdapterDisplayMode IDirect3D9::GetAdapterDisplayMode 메서드(d3d9.h)는 어댑터의 현재 표시 모드를 검색합니다.
GetAdapterDisplayModeEx 어댑터의 현재 디스플레이 모드 및 회전 설정을 검색합니다.
GetAdapterIdentifier

IDirect3D9::GetAdapterIdentifier(d3d9helper.h) 메서드는 IDirect3D9 인터페이스가 인스턴스화되었을 때 시스템에 있는 물리적 디스플레이 어댑터를 설명합니다.
GetAdapterIdentifier IDirect3D9::GetAdapterIdentifier 메서드(d3d9.h)는 IDirect3D9 인터페이스가 인스턴스화되었을 때 시스템에 있는 물리적 디스플레이 어댑터를 설명합니다.
GetAdapterLUID 이 메서드는 어댑터 하드웨어와 관련된 어댑터의 고유 식별자를 반환합니다. 애플리케이션은 이 식별자를 사용하여 다양한 API(Direct3D 9, DXGI)에서 강력한 매핑을 정의할 수 있습니다.
GetAdapterModeCount IDirect3D9::GetAdapterModeCount(d3d9helper.h) 메서드는 이 어댑터에서 사용할 수 있는 표시 모드 수를 반환합니다.
GetAdapterModeCount IDirect3D9::GetAdapterModeCount 메서드(d3d9.h)는 이 어댑터에서 사용할 수 있는 표시 모드 수를 반환합니다.
GetAdapterModeCountEx 사용 가능한 표시 모드 수를 반환합니다.
GetAdapterMonitor IDirect3D9::GetAdapterMonitor(d3d9helper.h) 메서드는 Direct3D 개체와 연결된 모니터의 핸들을 반환합니다.
GetAdapterMonitor IDirect3D9::GetAdapterMonitor 메서드(d3d9.h)는 Direct3D 개체와 연결된 모니터의 핸들을 반환합니다.
GetAutoGenFilterType IDirect3DBaseTexture9::GetAutoGenFilterType(d3d9helper.h) 메서드는 자동으로 생성된 mipmap sublevels에 사용되는 필터 형식을 가져옵니다.
GetAutoGenFilterType IDirect3DBaseTexture9::GetAutoGenFilterType 메서드(d3d9helper.h)는 자동으로 생성된 mipmap sublevels에 사용되는 필터 형식을 가져옵니다.
GetAvailableTextureMem

IDirect3DDevice9::GetAvailableTextureMem 메서드(d3d9.h)는 사용 가능한 텍스처 메모리의 예상량을 반환합니다.
GetAvailableTextureMem IDirect3DDevice9::GetAvailableTextureMem 메서드(d3d9.h)는 사용 가능한 텍스처 메모리의 예상량을 반환합니다.
GetBackBuffer IDirect3DDevice9::GetBackBuffer 메서드(d3d9.h)는 디바이스의 스왑 체인에서 백 버퍼를 검색합니다.
GetBackBuffer IDirect3DSwapChain9::GetBackBuffer 메서드(d3d9helper.h)는 디바이스의 스왑 체인에서 백 버퍼를 검색합니다.
GetBackBuffer IDirect3DDevice9::GetBackBuffer 메서드(d3d9.h)는 디바이스의 스왑 체인에서 백 버퍼를 검색합니다.
GetBackBuffer IDirect3DSwapChain9::GetBackBuffer(d3d9.h) 메서드는 디바이스의 스왑 체인에서 백 버퍼를 검색합니다.
GetClipPlane IDirect3DDevice9::GetClipPlane 메서드(d3d9.h)는 디바이스에 대한 사용자 정의 클리핑 평면의 계수를 검색합니다.
GetClipPlane IDirect3DDevice9::GetClipPlane 메서드(d3d9.h)는 디바이스에 대한 사용자 정의 클리핑 평면의 계수를 검색합니다.
GetClipStatus IDirect3DDevice9::GetClipStatus 메서드(d3d9.h)는 클립 상태 검색합니다.
GetClipStatus IDirect3DDevice9::GetClipStatus 메서드(d3d9.h)는 클립 상태 검색합니다.
GetContainer

IDirect3DSurface9::GetContainer 메서드(d3d9helper.h)는 부모 큐브 텍스처 또는 텍스처(mipmap) 개체에 대한 액세스를 제공합니다.
GetContainer IDirect3DVolume9::GetContainer 메서드(d3d9.h)는 표면이 볼륨 텍스처의 자식 수준인 경우 부모 볼륨 텍스처 개체에 대한 액세스를 제공합니다.
GetContainer IDirect3DSurface9::GetContainer(d3d9.h) 메서드는 부모 큐브 텍스처 또는 텍스처(mipmap) 개체에 대한 액세스를 제공합니다.
GetContainer IDirect3DVolume9::GetContainer(d3d9.h) 메서드는 이 표면이 볼륨 텍스처의 자식 수준인 경우 부모 볼륨 텍스처 개체에 대한 액세스를 제공합니다.
GetCreationParameters IDirect3DDevice9::GetCreationParameters 메서드(d3d9.h)는 디바이스의 생성 매개 변수를 검색합니다.
GetCreationParameters IDirect3DDevice9::GetCreationParameters 메서드(d3d9.h)는 디바이스의 생성 매개 변수를 검색합니다.
GetCubeMapSurface IDirect3DCubeTexture9::GetCubeMapSurface 메서드(d3d9.h)는 큐브 텍스처 맵 표면을 검색합니다.
GetCubeMapSurface IDirect3DCubeTexture9::GetCubeMapSurface 메서드(d3d9.h)는 큐브 텍스처 맵 표면을 검색합니다.
GetCurrentTexturePalette IDirect3DDevice9::GetCurrentTexturePalette 메서드(d3d9.h)는 현재 텍스처 팔레트를 검색합니다.
GetCurrentTexturePalette IDirect3DDevice9::GetCurrentTexturePalette 메서드(d3d9.h)는 현재 텍스처 팔레트를 검색합니다.
GetData IDirect3DQuery9::GetData 메서드(d3d9helper.h)는 쿼리된 리소스를 폴링하여 쿼리 상태 또는 쿼리 결과를 가져옵니다.

GetData IDirect3DQuery9::GetData 메서드(d3d9.h)는 쿼리된 리소스를 폴링하여 쿼리 상태 또는 쿼리 결과를 가져옵니다.
GetDataSize IDirect3DQuery9::GetDataSize 메서드(d3d9helper.h)는 쿼리 데이터의 바이트 수를 가져옵니다.
GetDataSize IDirect3DQuery9::GetDataSize(d3d9.h) 메서드는 쿼리 데이터의 바이트 수를 가져옵니다.
Getdc IDirect3DSurface9::GetDC 메서드(d3d9helper.h)는 디바이스 컨텍스트를 검색합니다.
Getdc IDirect3DSurface9::GetDC(d3d9.h) 메서드는 디바이스 컨텍스트를 검색합니다.
GetDeclaration IDirect3DVertexDeclaration9::GetDeclaration 메서드(d3d9helper.h)는 꼭짓점 셰이더 선언을 가져옵니다.
GetDeclaration IDirect3DVertexDeclaration9::GetDeclaration(d3d9.h) 메서드는 꼭짓점 셰이더 선언을 가져옵니다.
GetDepthStencilSurface IDirect3DDevice9::GetDepthStencilSurface 메서드(d3d9.h)는 Direct3DDevice 개체가 소유한 깊이 스텐실 표면을 가져옵니다.
GetDepthStencilSurface IDirect3DDevice9::GetDepthStencilSurface 메서드(d3d9.h)는 Direct3DDevice 개체가 소유한 깊이 스텐실 표면을 가져옵니다.
GetDesc IDirect3DIndexBuffer9::GetDesc 메서드(d3d9helper.h)는 인덱스 버퍼 리소스에 대한 설명을 검색합니다.
GetDesc IDirect3DSurface9::GetDesc 메서드(d3d9helper.h)는 표면에 대한 설명을 검색합니다.

GetDesc IDirect3DVertexBuffer9::GetDesc 메서드(d3d9.h)는 꼭짓점 버퍼 리소스에 대한 설명을 검색합니다.
GetDesc IDirect3DVolume9::GetDesc 메서드(d3d9.h)는 볼륨에 대한 설명을 검색합니다.
GetDesc IDirect3DIndexBuffer9::GetDesc 메서드(d3d9helper.h)는 인덱스 버퍼 리소스에 대한 설명을 검색합니다.
GetDesc IDirect3DSurface9::GetDesc(d3d9.h) 메서드는 표면에 대한 설명을 검색합니다.
GetDesc IDirect3DVertexBuffer9::GetDesc(d3d9.h) 메서드는 꼭짓점 버퍼 리소스에 대한 설명을 검색합니다.
GetDesc IDirect3DVolume9::GetDesc(d3d9.h) 메서드는 볼륨에 대한 설명을 검색합니다.
GetDevice IDirect3DPixelShader9::GetDevice 메서드(d3d9helper.h)는 디바이스를 가져옵니다.
GetDevice IDirect3DQuery9::GetDevice 메서드(d3d9helper.h)는 쿼리 중인 디바이스를 가져옵니다.
GetDevice IDirect3DResource9::GetDevice 메서드(d3d9helper.h)는 리소스와 연결된 디바이스를 검색합니다.
GetDevice IDirect3DStateBlock9::GetDevice 메서드(d3d9helper.h)는 디바이스를 가져옵니다.
GetDevice IDirect3DSwapChain9::GetDevice 메서드(d3d9helper.h)는 스왑 체인과 연결된 디바이스를 검색합니다.
GetDevice IDirect3DVertexDeclaration9::GetDevice 메서드(d3d9helper.h)는 현재 디바이스를 가져옵니다.

GetDevice IDirect3DVertexShader9::GetDevice 메서드(d3d9.h)는 디바이스를 가져옵니다.
GetDevice IDirect3DVolume9::GetDevice 메서드(d3d9.h)는 볼륨과 연결된 디바이스를 검색합니다.
GetDevice IDirect3DPixelShader9::GetDevice 메서드(d3d9helper.h)는 디바이스를 가져옵니다.
GetDevice IDirect3DQuery9::GetDevice(d3d9.h) 메서드는 쿼리 중인 디바이스를 가져옵니다.
GetDevice IDirect3DResource9::GetDevice(d3d9.h) 메서드는 리소스와 연결된 디바이스를 검색합니다.
GetDevice IDirect3DStateBlock9::GetDevice(d3d9.h) 메서드는 디바이스를 가져옵니다.
GetDevice IDirect3DSwapChain9::GetDevice(d3d9.h) 메서드는 스왑 체인과 연결된 디바이스를 검색합니다.
GetDevice IDirect3DVertexDeclaration9::GetDevice(d3d9.h) 메서드는 현재 디바이스를 가져옵니다.
GetDevice IDirect3DVertexShader9::GetDevice(d3d9.h) 메서드는 디바이스를 가져옵니다.
GetDevice IDirect3DVolume9::GetDevice(d3d9.h) 메서드는 볼륨과 연결된 디바이스를 검색합니다.
GetDeviceCaps IDirect3D9::GetDeviceCaps(d3d9helper.h) 메서드는 디바이스에 대한 디바이스별 정보를 검색합니다.
GetDeviceCaps IDirect3DDevice9::GetDeviceCaps 메서드(d3d9.h)는 렌더링 디바이스의 기능을 검색합니다.

GetDeviceCaps IDirect3D9::GetDeviceCaps 메서드(d3d9.h)는 디바이스에 대한 디바이스별 정보를 검색합니다.
GetDeviceCaps IDirect3DDevice9::GetDeviceCaps 메서드(d3d9.h)는 렌더링 디바이스의 기능을 검색합니다.
GetDirect3D IDirect3DDevice9::GetDirect3D 메서드(d3d9.h)는 디바이스를 만든 Direct3D 개체의 instance 대한 인터페이스를 반환합니다.
GetDirect3D IDirect3DDevice9::GetDirect3D 메서드(d3d9.h)는 디바이스를 만든 Direct3D 개체의 instance 대한 인터페이스를 반환합니다.
GetDisplayMode IDirect3DDevice9::GetDisplayMode 메서드(d3d9.h)는 디스플레이 모드의 공간 해상도, 색 해상도 및 새로 고침 빈도를 검색합니다.
GetDisplayMode IDirect3DSwapChain9::GetDisplayMode 메서드(d3d9helper.h)는 디스플레이 모드의 공간 해상도, 색 해상도 및 새로 고침 빈도를 검색합니다.
GetDisplayMode IDirect3DDevice9::GetDisplayMode 메서드(d3d9.h)는 디스플레이 모드의 공간 해상도, 색 해상도 및 새로 고침 빈도를 검색합니다.
GetDisplayMode IDirect3DSwapChain9::GetDisplayMode(d3d9.h) 메서드는 디스플레이 모드의 공간 해상도, 색 해상도 및 새로 고침 빈도를 검색합니다.
GetDisplayModeEx 디스플레이 모드의 공간 해상도, 색 해상도, 새로 고침 빈도 및 회전 설정을 검색합니다. (IDirect3DDevice9Ex.GetDisplayModeEx)
GetDisplayModeEx 디스플레이 모드의 공간 해상도, 색 해상도, 새로 고침 빈도 및 회전 설정을 검색합니다. (IDirect3DSwapChain9Ex.GetDisplayModeEx)

GetFrontBufferData IDirect3DDevice9::GetFrontBufferData 메서드(d3d9.h)는 디바이스의 전면 버퍼 복사본을 생성하고 애플리케이션에서 제공하는 시스템 메모리 버퍼에 복사하는 위치를 생성합니다.
GetFrontBufferData IDirect3DSwapChain9::GetFrontBufferData 메서드(d3d9helper.h)는 스왑 체인의 프런트 버퍼 복사본을 생성하고 애플리케이션에서 제공하는 시스템 메모리 버퍼에 복사하는 위치를 생성합니다.
GetFrontBufferData IDirect3DDevice9::GetFrontBufferData 메서드(d3d9.h)는 디바이스의 프런트 버퍼를 애플리케이션에서 제공하는 시스템 메모리 버퍼에 복사합니다.
GetFrontBufferData IDirect3DSwapChain9::GetFrontBufferData(d3d9.h) 메서드는 스왑 체인의 프런트 버퍼 복사본을 생성하고 시스템 메모리 버퍼에 복사하는 위치를 생성합니다.
GetFunction IDirect3DPixelShader9::GetFunction 메서드(d3d9helper.h)는 셰이더 데이터에 대한 포인터를 가져옵니다.
GetFunction IDirect3DVertexShader9::GetFunction 메서드(d3d9.h)는 셰이더 데이터에 대한 포인터를 가져옵니다.
GetFunction IDirect3DPixelShader9::GetFunction 메서드(d3d9helper.h)는 셰이더 데이터에 대한 포인터를 가져옵니다.
GetFunction IDirect3DVertexShader9::GetFunction(d3d9.h) 메서드는 셰이더 데이터에 대한 포인터를 가져옵니다.
GetFVF IDirect3DDevice9::GetFVF 메서드(d3d9.h)는 고정 꼭짓점 함수 선언을 가져옵니다.
GetFVF IDirect3DDevice9::GetFVF 메서드(d3d9.h)는 고정 꼭짓점 함수 선언을 가져옵니다.

GetGammaRamp IDirect3DDevice9::GetGammaRamp 메서드(d3d9.h)는 스왑 체인에 대한 감마 수정 램프를 검색합니다.
GetGammaRamp IDirect3DDevice9::GetGammaRamp 메서드(d3d9.h)는 스왑 체인에 대한 감마 수정 램프를 검색합니다.
GetGPUThreadPriority GPU 스레드의 우선 순위를 가져옵니다.
GetIndices IDirect3DDevice9::GetIndices 메서드(d3d9.h)는 인덱스 데이터를 검색합니다.
GetIndices IDirect3DDevice9::GetIndices 메서드(d3d9.h)는 인덱스 데이터를 검색합니다.
GetLastPresentCount 스왑 체인이 처리된 횟수를 반환합니다.
GetLevelCount IDirect3DBaseTexture9::GetLevelCount(d3d9helper.h) 메서드는 다단계 텍스처의 텍스처 수준 수를 반환합니다.
GetLevelCount IDirect3DBaseTexture9::GetLevelCount 메서드(d3d9helper.h)는 다단계 텍스처의 텍스처 수준 수를 반환합니다.
GetLevelDesc IDirect3DCubeTexture9::GetLevelDesc 메서드(d3d9.h)는 지정된 큐브 텍스처 수준의 한 면에 대한 설명을 검색합니다.
GetLevelDesc IDirect3DTexture9::GetLevelDesc 메서드(d3d9.h)는 텍스처 리소스에 대한 수준 설명을 검색합니다.
GetLevelDesc IDirect3DVolumeTexture9::GetLevelDesc 메서드(d3d9helper.h)는 볼륨 텍스처 리소스에 대한 수준 설명을 검색합니다.

GetLevelDesc IDirect3DCubeTexture9::GetLevelDesc 메서드(d3d9.h)는 지정된 큐브 텍스처 수준의 한 면에 대한 설명을 검색합니다.
GetLevelDesc IDirect3DTexture9::GetLevelDesc(d3d9.h) 메서드는 텍스처 리소스에 대한 수준 설명을 검색합니다.
GetLevelDesc IDirect3DVolumeTexture9::GetLevelDesc(d3d9.h) 메서드는 볼륨 텍스처 리소스에 대한 수준 설명을 검색합니다.
GetLight IDirect3DDevice9::GetLight 메서드(d3d9.h)는 이 디바이스에서 사용하는 조명 속성 집합을 검색합니다.
GetLight IDirect3DDevice9::GetLight 메서드(d3d9.h)는 이 디바이스에서 사용하는 조명 속성 집합을 검색합니다.
GetLightEnable IDirect3DDevice9::GetLightEnable 메서드(d3d9.h)는 디바이스 내의 조명 매개 변수 집합에 대해 사용 또는 사용 안 함의 활동 상태 검색합니다.
GetLightEnable IDirect3DDevice9::GetLightEnable 메서드(d3d9.h)는 디바이스 내의 조명 매개 변수 집합에 대한 활동 상태 검색합니다.
GetLOD IDirect3DBaseTexture9::GetLOD(d3d9helper.h) 메서드는 관리되는 텍스처에 대한 최대 세부 수준 집합으로 고정된 값을 반환합니다.
GetLOD IDirect3DBaseTexture9::GetLOD 메서드(d3d9helper.h)는 관리되는 텍스처에 대한 최대 세부 수준 집합으로 고정된 값을 반환합니다.
GetMaterial IDirect3DDevice9::GetMaterial 메서드(d3d9.h)는 디바이스의 현재 재질 속성을 검색합니다.

GetMaterial IDirect3DDevice9::GetMaterial 메서드(d3d9.h)는 디바이스의 현재 재질 속성을 검색합니다.
GetMaximumFrameLatency 시스템에서 큐에 대기할 수 있는 데이터 프레임 수를 검색합니다.
GetNPatchMode IDirect3DDevice9::GetNPatchMode 메서드(d3d9.h)는 N 패치 모드 세그먼트를 가져옵니다.
GetNPatchMode IDirect3DDevice9::GetNPatchMode 메서드(d3d9.h)는 N 패치 모드 세그먼트를 가져옵니다.
GetNumberOfSwapChains IDirect3DDevice9::GetNumberOfSwapChains 메서드(d3d9.h)는 암시적 스왑 체인의 수를 가져옵니다.
GetNumberOfSwapChains IDirect3DDevice9::GetNumberOfSwapChains 메서드(d3d9.h)는 암시적 스왑 체인의 수를 가져옵니다.
GetPaletteEntries IDirect3DDevice9::GetPaletteEntries 메서드(d3d9.h)는 팔레트 항목을 검색합니다.
GetPaletteEntries IDirect3DDevice9::GetPaletteEntries 메서드(d3d9.h)는 팔레트 항목을 검색합니다.
GetPixelShader IDirect3DDevice9::GetPixelShader 메서드(d3d9.h)는 현재 설정된 픽셀 셰이더를 검색합니다.
GetPixelShader IDirect3DDevice9::GetPixelShader 메서드(d3d9.h)는 현재 설정된 픽셀 셰이더를 검색합니다.
GetPixelShaderConstantB IDirect3DDevice9::GetPixelShaderConstantB 메서드(d3d9.h)는 부울 셰이더 상수를 가져옵니다.
GetPixelShaderConstantB IDirect3DDevice9::GetPixelShaderConstantB 메서드(d3d9.h)는 부울 셰이더 상수를 가져옵니다.

GetPixelShaderConstantF IDirect3DDevice9::GetPixelShaderConstantF 메서드(d3d9.h)는 부동 소수점 셰이더 상수를 가져옵니다.
GetPixelShaderConstantF IDirect3DDevice9::GetPixelShaderConstantF 메서드(d3d9.h)는 부동 소수점 셰이더 상수를 가져옵니다.
GetPixelShaderConstantI IDirect3DDevice9::GetPixelShaderConstantI 메서드(d3d9.h)는 정수 셰이더 상수를 가져옵니다.
GetPixelShaderConstantI IDirect3DDevice9::GetPixelShaderConstantI 메서드(d3d9.h)는 정수 셰이더 상수를 가져옵니다.
GetPresentParameters IDirect3DSwapChain9::GetPresentParameters 메서드(d3d9helper.h)는 스왑 체인과 연결된 프레젠테이션 매개 변수를 검색합니다.
GetPresentParameters IDirect3DSwapChain9::GetPresentParameters(d3d9.h) 메서드는 스왑 체인과 연결된 프레젠테이션 매개 변수를 검색합니다.
GetPriority IDirect3DResource9::GetPriority 메서드(d3d9helper.h)는 이 리소스의 우선 순위를 검색합니다.
GetPriority IDirect3DResource9::GetPriority(d3d9.h) 메서드는 이 리소스의 우선 순위를 검색합니다.
GetPrivateData IDirect3DResource9::GetPrivateData 메서드(d3d9helper.h)는 리소스와 연결된 프라이빗 데이터를 제공된 버퍼에 복사합니다.
GetPrivateData IDirect3DVolume9::GetPrivateData 메서드(d3d9.h)는 볼륨과 연결된 개인 데이터를 제공된 버퍼에 복사합니다.
GetPrivateData

IDirect3DResource9::GetPrivateData(d3d9.h) 메서드는 리소스와 연결된 프라이빗 데이터를 제공된 버퍼에 복사합니다.
GetPrivateData IDirect3DVolume9::GetPrivateData(d3d9.h) 메서드는 볼륨과 연결된 개인 데이터를 제공된 버퍼에 복사합니다.
GetRasterStatus IDirect3DDevice9::GetRasterStatus 메서드(d3d9.h)는 스왑 체인이 표시되는 모니터의 래스터를 설명하는 정보를 반환합니다.
GetRasterStatus IDirect3DSwapChain9::GetRasterStatus 메서드(d3d9helper.h)는 스왑 체인이 표시되는 모니터의 래스터를 설명하는 정보를 반환합니다.
GetRasterStatus IDirect3DDevice9::GetRasterStatus 메서드(d3d9.h)는 스왑 체인이 표시되는 모니터의 래스터를 설명하는 정보를 반환합니다.
GetRasterStatus IDirect3DSwapChain9::GetRasterStatus(d3d9.h) 메서드는 스왑 체인이 표시되는 모니터의 래스터를 설명하는 정보를 반환합니다.
GetRenderState IDirect3DDevice9::GetRenderState 메서드(d3d9.h)는 디바이스의 렌더링 상태 값을 검색합니다.
GetRenderState IDirect3DDevice9::GetRenderState 메서드(d3d9.h)는 디바이스의 렌더링 상태 값을 검색합니다.
GetRenderTarget IDirect3DDevice9::GetRenderTarget 메서드(d3d9.h)는 렌더링 대상 표면을 검색합니다.
GetRenderTarget IDirect3DDevice9::GetRenderTarget 메서드(d3d9.h)는 렌더링 대상 표면을 검색합니다.
GetRenderTargetData IDirect3DDevice9::GetRenderTargetData 메서드(d3d9.h)는 렌더링 대상 데이터를 디바이스 메모리에서 시스템 메모리로 복사합니다.

GetRenderTargetData IDirect3DDevice9::GetRenderTargetData 메서드(d3d9.h)는 렌더링 대상 데이터를 디바이스 메모리에서 시스템 메모리로 복사합니다.
GetSamplerState IDirect3DDevice9::GetSamplerState 메서드(d3d9.h)는 샘플러 상태 값을 가져옵니다.
GetSamplerState IDirect3DDevice9::GetSamplerState 메서드(d3d9.h)는 샘플러 상태 값을 가져옵니다.
GetScissorRect IDirect3DDevice9::GetScissorRect 메서드(d3d9.h)는 가위 사각형을 가져옵니다.
GetScissorRect IDirect3DDevice9::GetScissorRect 메서드(d3d9.h)는 가위 사각형을 가져옵니다.
GetSoftwareVertexProcessing IDirect3DDevice9::GetSoftwareVertexProcessing 메서드(d3d9.h)는 꼭짓점 하드웨어 또는 소프트웨어 처리 모드를 가져옵니다.
GetSoftwareVertexProcessing IDirect3DDevice9::GetSoftwareVertexProcessing 메서드(d3d9.h)는 꼭짓점 처리(하드웨어 또는 소프트웨어) 모드를 가져옵니다.
GetStreamSource IDirect3DDevice9::GetStreamSource 메서드(d3d9.h)는 지정된 데이터 스트림에 바인딩된 꼭짓점 버퍼를 검색합니다.
GetStreamSource IDirect3DDevice9::GetStreamSource 메서드(d3d9.h)는 지정된 데이터 스트림에 바인딩된 꼭짓점 버퍼를 검색합니다.
GetStreamSourceFreq IDirect3DDevice9::GetStreamSourceFreq 메서드(d3d9.h)는 스트림 원본 빈도 구분선 값을 가져옵니다.
GetStreamSourceFreq

IDirect3DDevice9::GetStreamSourceFreq 메서드(d3d9.h)는 스트림 원본 빈도 구분선 값을 가져옵니다.
GetSurfaceLevel IDirect3DTexture9::GetSurfaceLevel 메서드(d3d9.h)는 지정된 텍스처 표면 수준을 검색합니다.
GetSurfaceLevel IDirect3DTexture9::GetSurfaceLevel(d3d9.h) 메서드는 지정된 텍스처 표면 수준을 검색합니다.
GetSwapChain IDirect3DDevice9::GetSwapChain 메서드(d3d9.h)는 스왑 체인에 대한 포인터를 가져옵니다.
GetSwapChain IDirect3DDevice9::GetSwapChain 메서드(d3d9.h)는 스왑 체인에 대한 포인터를 가져옵니다.
GetTexture IDirect3DDevice9::GetTexture 메서드(d3d9.h)는 디바이스의 스테이지에 할당된 텍스처를 검색합니다.
GetTexture IDirect3DDevice9::GetTexture 메서드(d3d9.h)는 디바이스의 스테이지에 할당된 텍스처를 검색합니다.
GetTextureStageState IDirect3DDevice9::GetTextureStageState 메서드(d3d9.h)는 할당된 텍스처에 대한 상태 값을 검색합니다.
GetTextureStageState IDirect3DDevice9::GetTextureStageState 메서드(d3d9.h)는 할당된 텍스처에 대한 상태 값을 검색합니다.
GetTransform IDirect3DDevice9::GetTransform 메서드(d3d9.h)는 변환 상태를 설명하는 행렬을 검색합니다.
GetTransform IDirect3DDevice9::GetTransform 메서드(d3d9.h)는 변환 상태를 설명하는 행렬을 검색합니다.
GetType

IDirect3DQuery9::GetType 메서드(d3d9helper.h)는 쿼리 형식을 가져옵니다.
GetType IDirect3DResource9::GetType 메서드(d3d9helper.h)는 리소스의 형식을 반환합니다.
GetType IDirect3DQuery9::GetType(d3d9.h) 메서드는 쿼리 형식을 가져옵니다.
GetType IDirect3DResource9::GetType(d3d9.h) 메서드는 리소스의 형식을 반환합니다.
GetVertexDeclaration IDirect3DDevice9::GetVertexDeclaration 메서드(d3d9.h)는 꼭짓점 셰이더 선언을 가져옵니다.
GetVertexDeclaration IDirect3DDevice9::GetVertexDeclaration 메서드(d3d9.h)는 꼭짓점 셰이더 선언을 가져옵니다.
GetVertexShader IDirect3DDevice9::GetVertexShader 메서드(d3d9.h)는 현재 설정된 꼭짓점 셰이더를 검색합니다.
GetVertexShader IDirect3DDevice9::GetVertexShader 메서드(d3d9.h)는 현재 설정된 꼭짓점 셰이더를 검색합니다.
GetVertexShaderConstantB IDirect3DDevice9::GetVertexShaderConstantB 메서드(d3d9.h)는 부울 꼭짓점 셰이더 상수를 가져옵니다.
GetVertexShaderConstantB IDirect3DDevice9::GetVertexShaderConstantB 메서드(d3d9.h)는 부울 꼭짓점 셰이더 상수를 가져옵니다.
GetVertexShaderConstantF IDirect3DDevice9::GetVertexShaderConstantF 메서드(d3d9.h)는 부동 소수점 꼭짓점 셰이더 상수를 가져옵니다.
GetVertexShaderConstantF IDirect3DDevice9::GetVertexShaderConstantF 메서드(d3d9.h)는 부동 소수점 꼭짓점 셰이더 상수를 가져옵니다.

GetVertexShaderConstantI IDirect3DDevice9::GetVertexShaderConstantI 메서드(d3d9.h)는 정수 꼭짓점 셰이더 상수를 가져옵니다.
GetVertexShaderConstantI IDirect3DDevice9::GetVertexShaderConstantI 메서드(d3d9.h)는 정수 꼭짓점 셰이더 상수를 가져옵니다.
GetViewport IDirect3DDevice9::GetViewport 메서드(d3d9.h)는 디바이스에 대해 현재 설정된 뷰포트 매개 변수를 검색합니다.
GetViewport IDirect3DDevice9::GetViewport 메서드(d3d9.h)는 디바이스에 대해 현재 설정된 뷰포트 매개 변수를 검색합니다.
GetVolumeLevel IDirect3DVolumeTexture9::GetVolumeLevel 메서드(d3d9helper.h)는 지정된 볼륨 텍스처 수준을 검색합니다.
GetVolumeLevel IDirect3DVolumeTexture9::GetVolumeLevel(d3d9.h) 메서드는 지정된 볼륨 텍스처 수준을 검색합니다.
문제점 IDirect3DQuery9::Issue 메서드(d3d9helper.h)는 쿼리를 실행합니다.
문제점 IDirect3DQuery9::Issue(d3d9.h) 메서드는 쿼리를 실행합니다.
LightEnable IDirect3DDevice9::LightEnable 메서드(d3d9.h)는 디바이스 내에서 조명 매개 변수 집합을 사용하거나 사용하지 않도록 설정합니다.
LightEnable IDirect3DDevice9::LightEnable 메서드(d3d9.h)는 디바이스 내에서 조명 매개 변수 집합을 사용하거나 사용하지 않도록 설정합니다.

잠금 IDirect3DIndexBuffer9::Lock 메서드(d3d9helper.h)는 인덱스 데이터의 범위를 잠그고 인덱스 버퍼 메모리에 대한 포인터를 가져옵니다.
잠금 IDirect3DVertexBuffer9::Lock 메서드(d3d9helper.h)는 꼭짓점 데이터의 범위를 잠그고 꼭짓점 버퍼 메모리에 대한 포인터를 가져옵니다.
잠금 IDirect3DIndexBuffer9::Lock 메서드(d3d9helper.h)는 인덱스 데이터의 범위를 잠그고 인덱스 버퍼 메모리에 대한 포인터를 가져옵니다.
잠금 IDirect3DVertexBuffer9::Lock(d3d9.h) 메서드는 꼭짓점 데이터의 범위를 잠그고 꼭짓점 버퍼 메모리에 대한 포인터를 가져옵니다.
Lockbox IDirect3DVolume9::LockBox 메서드(d3d9.h)는 볼륨 리소스에 상자를 잠급니다.
Lockbox IDirect3DVolumeTexture9::LockBox 메서드(d3d9helper.h)는 볼륨 텍스처 리소스에 상자를 잠급니다.
Lockbox IDirect3DVolume9::LockBox(d3d9.h) 메서드는 볼륨 리소스에 상자를 잠급니다.
Lockbox IDirect3DVolumeTexture9::LockBox(d3d9.h) 메서드는 볼륨 텍스처 리소스에 상자를 잠급니다.
LockRect IDirect3DCubeTexture9::LockRect 메서드(d3d9.h)는 큐브 텍스처 리소스에 사각형을 잠급니다.
LockRect IDirect3DSurface9::LockRect 메서드(d3d9helper.h)는 표면에 사각형을 잠급니다.
LockRect IDirect3DTexture9::LockRect 메서드(d3d9.h)는 텍스처 리소스에 사각형을 잠급니다.

LockRect IDirect3DCubeTexture9::LockRect 메서드(d3d9.h)는 큐브 텍스처 리소스에 사각형을 잠급니다.
LockRect IDirect3DSurface9::LockRect(d3d9.h) 메서드는 표면에 사각형을 잠급니다.
LockRect IDirect3DTexture9::LockRect(d3d9.h) 메서드는 텍스처 리소스에 사각형을 잠급니다.
MultiplyTransform IDirect3DDevice9::MultiplyTransform 메서드(d3d9.h)는 디바이스의 월드, 뷰 또는 프로젝션 행렬을 지정된 행렬에 곱합니다.
MultiplyTransform IDirect3DDevice9::MultiplyTransform 메서드(d3d9.h)는 디바이스의 월드, 뷰 또는 프로젝션 행렬을 지정된 행렬에 곱합니다.
미리 IDirect3DResource9::P reLoad 메서드(d3d9helper.h)는 관리되는 리소스를 미리 로드합니다.
미리 IDirect3DResource9::P reLoad(d3d9.h) 메서드는 관리되는 리소스를 미리 로드합니다.
있음 IDirect3DDevice9::P resent 메서드(d3d9.h)는 디바이스가 소유한 백 버퍼 시퀀스에서 다음 버퍼의 내용을 표시합니다.
있음 IDirect3DSwapChain9::P resent 메서드(d3d9helper.h)는 스왑 체인이 소유한 백 버퍼 시퀀스에서 다음 버퍼의 내용을 표시합니다.
있음 IDirect3DDevice9::P resent 메서드(d3d9.h)는 디바이스가 소유한 백 버퍼 시퀀스에서 다음 버퍼의 내용을 표시합니다.
있음 IDirect3DSwapChain9::P resent(d3d9.h) 메서드는 스왑 체인이 소유한 백 버퍼 시퀀스에서 다음 버퍼의 내용을 제공합니다.

PresentEx 스왑 체인의 다음 버퍼를 프런트 버퍼로 교환합니다.
ProcessVertices IDirect3DDevice9::P rocessVertices 메서드(d3d9.h)는 꼭짓점 셰이더에서 정의한 꼭짓점 처리를 입력 데이터 스트림 집합에 적용합니다.
ProcessVertices IDirect3DDevice9::P rocessVertices 메서드(d3d9.h)는 꼭짓점 셰이더에서 정의한 꼭짓점 처리를 입력 데이터 스트림 집합에 적용합니다.
RegisterSoftwareDevice IDirect3D9::RegisterSoftwareDevice(d3d9helper.h) 메서드는 애플리케이션이 소프트웨어 래스터라이저에 액세스할 수 있도록 소프트웨어 래스터화를 제공하는 플러그형 소프트웨어 디바이스를 등록합니다.
RegisterSoftwareDevice IDirect3D9::RegisterSoftwareDevice 메서드(d3d9.h)는 플러그형 소프트웨어 디바이스를 등록합니다.
ReleaseDC IDirect3DSurface9::ReleaseDC 메서드(d3d9helper.h)는 디바이스 컨텍스트 핸들을 해제합니다.
ReleaseDC IDirect3DSurface9::ReleaseDC(d3d9.h) 메서드는 디바이스 컨텍스트 핸들을 해제합니다.
Reset IDirect3DDevice9::Reset 메서드(d3d9.h)는 스왑 체인의 형식, 크기 및 형식을 다시 설정합니다.
Reset IDirect3DDevice9::Reset 메서드(d3d9.h)는 스왑 체인의 형식, 크기 및 형식을 다시 설정합니다.
ResetEx 다른 모든 표면을 영구적으로 사용하여 스왑 체인의 형식, 크기 및 형식을 다시 설정합니다.
SetAutoGenFilterType IDirect3DBaseTexture9::SetAutoGenFilterType(d3d9helper.h) 메서드는 자동으로 생성된 Mipmap sublevels에 사용되는 필터 형식을 설정합니다.

SetAutoGenFilterType IDirect3DBaseTexture9::SetAutoGenFilterType 메서드(d3d9helper.h)는 자동으로 생성된 mipmap sublevels에 사용되는 필터 형식을 설정합니다.
SetClipPlane IDirect3DDevice9::SetClipPlane 메서드(d3d9.h)는 디바이스에 대한 사용자 정의 클리핑 평면의 계수를 설정합니다.
SetClipPlane IDirect3DDevice9::SetClipPlane 메서드(d3d9.h)는 디바이스에 대한 사용자 정의 클리핑 평면의 계수를 설정합니다.
SetClipStatus IDirect3DDevice9::SetClipStatus 메서드(d3d9.h)는 클립을 상태 설정합니다.
SetClipStatus IDirect3DDevice9::SetClipStatus 메서드(d3d9.h)는 클립을 상태 설정합니다.
SetConvolutionMonoKernel 단일 색 텍스처에서 단색 컨볼루션 필터링을 위해 텍스처 샘플러를 준비합니다.
SetCurrentTexturePalette IDirect3DDevice9::SetCurrentTexturePalette 메서드(d3d9.h)는 현재 텍스처 팔레트를 설정합니다.
SetCurrentTexturePalette IDirect3DDevice9::SetCurrentTexturePalette 메서드(d3d9.h)는 현재 텍스처 팔레트를 설정합니다.
SetCursorPosition IDirect3DDevice9::SetCursorPosition 메서드(d3d9.h)는 커서 위치 및 업데이트 옵션을 설정합니다.
SetCursorPosition IDirect3DDevice9::SetCursorPosition 메서드(d3d9.h)는 커서 위치 및 업데이트 옵션을 설정합니다.
SetCursorProperties IDirect3DDevice9::SetCursorProperties 메서드(d3d9.h)는 커서의 속성을 설정합니다.
SetCursorProperties

IDirect3DDevice9::SetCursorProperties 메서드(d3d9.h)는 커서의 속성을 설정합니다.
SetDepthStencilSurface IDirect3DDevice9::SetDepthStencilSurface 메서드(d3d9.h)는 깊이 스텐실 표면을 설정합니다.
SetDepthStencilSurface IDirect3DDevice9::SetDepthStencilSurface 메서드(d3d9.h)는 깊이 스텐실 표면을 설정합니다.
SetDialogBoxMode IDirect3DDevice9::SetDialogBoxMode 메서드(d3d9.h)를 사용하면 전체 화면 모드 애플리케이션에서 GDI 대화 상자를 사용할 수 있습니다.
SetDialogBoxMode IDirect3DDevice9::SetDialogBoxMode 메서드(d3d9.h)를 사용하면 전체 화면 모드 애플리케이션에서 GDI 대화 상자를 사용할 수 있습니다.
SetFVF IDirect3DDevice9::SetFVF 메서드(d3d9.h)는 현재 꼭짓점 스트림 선언을 설정합니다.
SetFVF IDirect3DDevice9::SetFVF 메서드(d3d9.h)는 현재 꼭짓점 스트림 선언을 설정합니다.
SetGammaRamp IDirect3DDevice9::SetGammaRamp 메서드(d3d9helper.h)는 암시적 스왑 체인에 대한 감마 수정 램프를 설정합니다.
SetGammaRamp IDirect3DDevice9::SetGammaRamp 메서드(d3d9.h)는 암시적 스왑 체인에 대한 감마 수정 램프를 설정합니다.
SetGPUThreadPriority GPU 스레드의 우선 순위를 설정합니다.
SetIndices IDirect3DDevice9::SetIndices 메서드(d3d9helper.h)는 인덱스 데이터를 설정합니다.
SetIndices IDirect3DDevice9::SetIndices 메서드(d3d9.h)는 인덱스 데이터를 설정합니다.

SetLight IDirect3DDevice9::SetLight 메서드(d3d9helper.h)는 이 디바이스에 대한 조명 속성 집합을 할당합니다.
SetLight IDirect3DDevice9::SetLight 메서드(d3d9.h)는 이 디바이스에 대한 조명 속성 집합을 할당합니다.
SetLOD IDirect3DBaseTexture9::SetLOD 메서드(d3d9helper.h)는 관리되는 텍스처에 대해 가장 자세한 세부 수준을 설정합니다.
SetLOD IDirect3DBaseTexture9::SetLOD는 관리되는 텍스처에 대해 가장 자세한 세부 수준을 설정합니다.
SetMaterial IDirect3DDevice9::SetMaterial 메서드(d3d9helper.h)는 디바이스의 재질 속성을 설정합니다.
SetMaterial IDirect3DDevice9::SetMaterial 메서드(d3d9.h)는 디바이스의 재질 속성을 설정합니다.
SetMaximumFrameLatency 시스템에서 렌더링을 위해 큐에 대기할 수 있는 프레임 수를 설정합니다.
SetNPatchMode IDirect3DDevice9::SetNPatchMode 메서드(d3d9helper.h)는 N 패치를 사용하거나 사용하지 않도록 설정합니다.
SetNPatchMode IDirect3DDevice9::SetNPatchMode 메서드(d3d9.h)는 N 패치를 사용하거나 사용하지 않도록 설정합니다.
SetPaletteEntries IDirect3DDevice9::SetPaletteEntries 메서드(d3d9helper.h)는 팔레트 항목을 설정합니다.
SetPaletteEntries IDirect3DDevice9::SetPaletteEntries 메서드(d3d9.h)는 팔레트 항목을 설정합니다.

SetPixelShader IDirect3DDevice9::SetPixelShader 메서드(d3d9helper.h)는 현재 픽셀 셰이더를 이전에 만든 픽셀 셰이더로 설정합니다.
SetPixelShader IDirect3DDevice9::SetPixelShader 메서드(d3d9.h)는 현재 픽셀 셰이더를 이전에 만든 픽셀 셰이더로 설정합니다.
SetPixelShaderConstantB IDirect3DDevice9::SetPixelShaderConstantB 메서드(d3d9helper.h)는 부울 셰이더 상수를 설정합니다.
SetPixelShaderConstantB IDirect3DDevice9::SetPixelShaderConstantB 메서드(d3d9.h)는 부울 셰이더 상수를 설정합니다.
SetPixelShaderConstantF IDirect3DDevice9::SetPixelShaderConstantF 메서드(d3d9helper.h)는 부동 소수점 셰이더 상수를 설정합니다.
SetPixelShaderConstantF IDirect3DDevice9::SetPixelShaderConstantF 메서드(d3d9helper.h)는 부동 소수점 셰이더 상수를 설정합니다.
SetPixelShaderConstantI IDirect3DDevice9::SetPixelShaderConstantI 메서드(d3d9helper.h)는 정수 셰이더 상수를 설정합니다.
SetPixelShaderConstantI IDirect3DDevice9::SetPixelShaderConstantI 메서드(d3d9helper.h)는 정수 셰이더 상수를 설정합니다.
SetPriority IDirect3DResource9::SetPriority 메서드(d3d9helper.h)는 예약을 위해 리소스의 우선 순위를 할당합니다.
SetPriority IDirect3DResource9::SetPriority(d3d9.h) 메서드는 일정 예약을 위해 리소스의 우선 순위를 할당합니다.

SetPrivateData IDirect3DResource9::SetPrivateData 메서드(d3d9helper.h)는 Direct3D가 아닌 애플리케이션에서 사용하기 위한 리소스와 데이터를 연결합니다.
SetPrivateData IDirect3DVolume9::SetPrivateData 메서드(d3d9.h)는 Direct3D가 아닌 애플리케이션에서 사용하기 위한 볼륨과 데이터를 연결합니다.
SetPrivateData IDirect3DResource9::SetPrivateData(d3d9.h) 메서드는 Direct3D가 아닌 애플리케이션에서 사용하기 위한 리소스와 데이터를 연결합니다.
SetPrivateData IDirect3DVolume9::SetPrivateData(d3d9.h) 메서드는 Direct3D가 아닌 애플리케이션에서 사용하기 위한 볼륨과 데이터를 연결합니다.
SetRenderState IDirect3DDevice9::SetRenderState 메서드(d3d9helper.h)는 단일 디바이스 렌더링 상태 매개 변수를 설정합니다.
SetRenderState IDirect3DDevice9::SetRenderState 메서드(d3d9helper.h)는 단일 디바이스 렌더링 상태 매개 변수를 설정합니다.
SetRenderTarget IDirect3DDevice9::SetRenderTarget 메서드(d3d9helper.h)는 디바이스에 대한 새 색 버퍼를 설정합니다.
SetRenderTarget IDirect3DDevice9::SetRenderTarget 메서드(d3d9helper.h)는 디바이스에 대한 새 색 버퍼를 설정합니다.
SetSamplerState IDirect3DDevice9::SetSamplerState 메서드(d3d9helper.h)는 샘플러 상태 값을 설정합니다.
SetSamplerState IDirect3DDevice9::SetSamplerState 메서드(d3d9helper.h)는 샘플러 상태 값을 설정합니다.

SetScissorRect IDirect3DDevice9::SetScissorRect 메서드(d3d9helper.h)는 가위 사각형을 설정합니다.
SetScissorRect IDirect3DDevice9::SetScissorRect 메서드(d3d9helper.h)는 가위 사각형을 설정합니다.
SetSoftwareVertexProcessing IDirect3DDevice9::SetSoftwareVertexProcessing 메서드(d3d9helper.h)는 소프트웨어와 하드웨어 꼭짓점 처리 간에 전환됩니다.
SetSoftwareVertexProcessing IDirect3DDevice9::SetSoftwareVertexProcessing 메서드(d3d9helper.h)를 사용하면 소프트웨어와 하드웨어 꼭짓점 처리 모드 간에 전환할 수 있습니다.
SetStreamSource IDirect3DDevice9::SetStreamSource 메서드(d3d9helper.h)는 꼭짓점 버퍼를 디바이스 데이터 스트림에 바인딩합니다.
SetStreamSource IDirect3DDevice9::SetStreamSource 메서드(d3d9helper.h)는 꼭짓점 버퍼를 디바이스 데이터 스트림에 바인딩합니다.
SetStreamSourceFreq IDirect3DDevice9::SetStreamSourceFreq 메서드(d3d9helper.h)는 스트림 소스 빈도 구분선 값을 설정합니다.
SetStreamSourceFreq IDirect3DDevice9::SetStreamSourceFreq 메서드(d3d9helper.h)는 스트림 소스 빈도 구분선 값을 설정합니다.
SetTexture IDirect3DDevice9::SetTexture 메서드(d3d9helper.h)는 디바이스의 스테이지에 텍스처를 할당합니다.
SetTexture IDirect3DDevice9::SetTexture 메서드(d3d9helper.h)는 디바이스의 스테이지에 텍스처를 할당합니다.

SetTextureStageState IDirect3DDevice9::SetTextureStageState 메서드(d3d9helper.h)는 현재 할당된 텍스처의 상태 값을 설정합니다.
SetTextureStageState IDirect3DDevice9::SetTextureStageState 메서드(d3d9helper.h)는 현재 할당된 텍스처의 상태 값을 설정합니다.
SetTransform IDirect3DDevice9::SetTransform 메서드(d3d9helper.h)는 단일 디바이스 변환 관련 상태를 설정합니다.
SetTransform IDirect3DDevice9::SetTransform 메서드(d3d9helper.h)는 단일 디바이스 변환 관련 상태를 설정합니다.
SetVertexDeclaration IDirect3DDevice9::SetVertexDeclaration 메서드(d3d9helper.h)는 꼭짓점 선언(Direct3D 9)을 설정합니다.
SetVertexDeclaration IDirect3DDevice9::SetVertexDeclaration 메서드(d3d9helper.h)는 꼭짓점 선언(Direct3D 9)을 설정합니다.
SetVertexShader IDirect3DDevice9::SetVertexShader 메서드(d3d9helper.h)는 꼭짓점 셰이더를 설정합니다.
SetVertexShader IDirect3DDevice9::SetVertexShader 메서드(d3d9helper.h)는 꼭짓점 셰이더를 설정합니다.
SetVertexShaderConstantB IDirect3DDevice9::SetVertexShaderConstantB 메서드(d3d9helper.h)는 부울 꼭짓점 셰이더 상수를 설정합니다.
SetVertexShaderConstantB IDirect3DDevice9::SetVertexShaderConstantB 메서드(d3d9helper.h)는 부울 꼭짓점 셰이더 상수를 설정합니다.

SetVertexShaderConstantF IDirect3DDevice9::SetVertexShaderConstantF 메서드(d3d9helper.h)는 부동 소수점 꼭짓점 셰이더 상수를 설정합니다.
SetVertexShaderConstantF IDirect3DDevice9::SetVertexShaderConstantF 메서드(d3d9helper.h)는 부동 소수점 꼭짓점 셰이더 상수를 설정합니다.
SetVertexShaderConstantI IDirect3DDevice9::SetVertexShaderConstantI 메서드(d3d9helper.h)는 정수 꼭짓점 셰이더 상수를 설정합니다.
SetVertexShaderConstantI IDirect3DDevice9::SetVertexShaderConstantI 메서드(d3d9helper.h)는 정수 꼭짓점 셰이더 상수를 설정합니다.
SetViewport IDirect3DDevice9::SetViewport 메서드(d3d9helper.h)는 디바이스에 대한 뷰포트 매개 변수를 설정합니다.
SetViewport IDirect3DDevice9::SetViewport 메서드(d3d9helper.h)는 디바이스에 대한 뷰포트 매개 변수를 설정합니다.
ShowCursor IDirect3DDevice9::ShowCursor 메서드(d3d9helper.h)는 커서를 표시하거나 숨깁니다.
ShowCursor IDirect3DDevice9::ShowCursor 메서드(d3d9helper.h)는 커서를 표시하거나 숨깁니다.
StretchRect IDirect3DDevice9::StretchRect 메서드(d3d9helper.h)는 원본 사각형의 내용을 대상 사각형에 복사합니다.
StretchRect IDirect3DDevice9::StretchRect 메서드(d3d9helper.h)를 사용하면 원본 사각형의 내용을 대상 사각형에 복사할 수 있습니다. 원본은 복사본으로 확장 및 필터링할 수 있습니다.

TestCooperativeLevel IDirect3DDevice9::TestCooperativeLevel 메서드(d3d9helper.h)는 창 또는 전체 화면 애플리케이션에 대한 Direct3D 디바이스의 현재 협력 수준 상태 보고합니다.
TestCooperativeLevel IDirect3DDevice9::TestCooperativeLevel 메서드(d3d9.h)는 창 또는 전체 화면 애플리케이션에 대한 Direct3D 디바이스의 현재 협력 수준 상태 보고합니다.
TestCooperativeLevel 창 또는 전체 화면 애플리케이션에 대한 Direct3D 디바이스의 현재 협력 수준 상태 보고합니다. (IDirect3DDevice9Ex.TestCooperativeLevel)
Unlock IDirect3DIndexBuffer9::Unlock 메서드(d3d9helper.h)는 인덱스 데이터의 잠금을 해제합니다.
Unlock IDirect3DVertexBuffer9::Unlock 메서드(d3d9helper.h)는 꼭짓점 데이터를 잠금 해제합니다.
Unlock IDirect3DIndexBuffer9::Unlock 메서드(d3d9helper.h)는 인덱스 데이터의 잠금을 해제합니다.
Unlock IDirect3DVertexBuffer9::Unlock(d3d9.h) 메서드는 꼭짓점 데이터의 잠금을 해제합니다.
UnlockBox IDirect3DVolume9::UnlockBox 메서드(d3d9.h)는 볼륨 리소스의 상자를 잠금 해제합니다.
UnlockBox IDirect3DVolumeTexture9::UnlockBox 메서드(d3d9helper.h)는 볼륨 텍스처 리소스의 상자를 잠금 해제합니다.
UnlockBox IDirect3DVolume9::UnlockBox(d3d9.h) 메서드는 볼륨 리소스의 상자 잠금을 해제합니다.
UnlockBox IDirect3DVolumeTexture9::UnlockBox(d3d9.h) 메서드는 볼륨 텍스처 리소스의 상자를 잠금 해제합니다.

UnlockRect IDirect3DCubeTexture9::UnlockRect 메서드(d3d9.h)는 큐브 텍스처 리소스에서 사각형의 잠금을 해제합니다.
UnlockRect IDirect3DSurface9::UnlockRect 메서드(d3d9helper.h)는 표면의 사각형을 잠금 해제합니다.
UnlockRect IDirect3DTexture9::UnlockRect 메서드(d3d9.h)는 텍스처 리소스에서 사각형의 잠금을 해제합니다.
UnlockRect IDirect3DCubeTexture9::UnlockRect 메서드(d3d9.h)는 큐브 텍스처 리소스에서 사각형의 잠금을 해제합니다.
UnlockRect IDirect3DSurface9::UnlockRect(d3d9.h) 메서드는 표면에서 사각형의 잠금을 해제합니다.
UnlockRect IDirect3DTexture9::UnlockRect(d3d9.h) 메서드는 텍스처 리소스에서 사각형의 잠금을 해제합니다.
UpdateSurface IDirect3DDevice9::UpdateSurface 메서드(d3d9helper.h)는 한 표면에서 다른 표면으로 픽셀의 사각형 하위 집합을 복사합니다.
UpdateSurface IDirect3DDevice9::UpdateSurface 메서드(d3d9.h)는 픽셀의 사각형 하위 집합을 한 표면에서 다른 표면으로 복사합니다.
UpdateTexture IDirect3DDevice9::UpdateTexture 메서드(d3d9helper.h)는 텍스처의 더티 부분을 업데이트합니다.
UpdateTexture IDirect3DDevice9::UpdateTexture 메서드(d3d9.h)는 텍스처의 더티 부분을 업데이트합니다.
ValidateDevice IDirect3DDevice9::ValidateDevice 메서드(d3d9helper.h)는 패스에서 현재 텍스처 혼합 작업 및 인수를 렌더링하는 디바이스의 기능을 보고합니다.

ValidateDevice IDirect3DDevice9::ValidateDevice 메서드(d3d9.h)는 단일 패스에서 현재 텍스처 혼합 작업 및 인수를 렌더링하는 디바이스의 기능을 보고합니다.
WaitForVBlank 다음 세로 빈 신호까지 호출 스레드 실행을 일시 중단합니다.

인터페이스

IDirect3D9 IDirect3D9 인터페이스(d3d9helper.h)는 Microsoft Direct3D 개체를 만들고 환경을 설정하는 메서드를 제공합니다.
IDirect3D9 IDirect3D9(d3d9.h) 인터페이스 애플리케이션은 IDirect3D9 인터페이스의 메서드를 사용하여 Microsoft Direct3D 개체를 만들고 환경을 설정합니다.
IDirect3D9Ex 애플리케이션은 IDirect3D9Ex 인터페이스(IDirect3D9에서 상속됨)의 메서드를 사용하여 Microsoft Direct3D 9Ex 개체를 만들고 환경을 설정합니다.
IDirect3DBaseTexture9 IDirect3DBaseTexture9 인터페이스(d3d9helper.h)는 큐브 및 볼륨 텍스처를 비롯한 텍스처 리소스를 조작하는 메서드를 제공합니다.
IDirect3DBaseTexture9 IDirect3DBaseTexture9(d3d9.h) 인터페이스 애플리케이션은 IDirect3DBaseTexture9 인터페이스의 메서드를 사용하여 텍스처 리소스를 조작합니다.
IDirect3DCubeTexture9 IDirect3DCubeTexture9 인터페이스(d3d9helper.h)는 큐브 텍스처 리소스를 조작하는 메서드를 제공합니다. (IDirect3DCubeTexture9)
IDirect3DCubeTexture9

IDirect3DCubeTexture9(d3d9.h) 인터페이스 애플리케이션은 IDirect3DCubeTexture9 인터페이스의 메서드를 사용하여 큐브 텍스처 리소스를 조작합니다.
IDirect3DDevice9 IDirect3DDevice9 인터페이스(d3d9helper.h)는 DrawPrimitive 기반 렌더링, 리소스 및 셰이더 만들기 등을 수행하는 메서드를 제공합니다.
IDirect3DDevice9 IDirect3DDevice9(d3d9.h) 애플리케이션은 IDirect3DDevice9 인터페이스의 메서드를 사용하여 DrawPrimitive 기반 렌더링을 수행하고 리소스를 만듭니다.
IDirect3DDevice9Ex 애플리케이션은 IDirect3DDevice9Ex 인터페이스의 메서드를 사용하여 기본 형식을 렌더링하고, 리소스를 만들고, 시스템 수준 변수로 작업하고, 감마 램프 수준을 조정하고, 색상표를 사용하고, 셰이더를 만듭니다.
IDirect3DIndexBuffer9 IDirect3DIndexBuffer9 인터페이스(d3d9helper.h)는 인덱스 버퍼 리소스를 조작하는 메서드를 제공합니다.
IDirect3DIndexBuffer9 IDirect3DIndexBuffer9(d3d9.h) 인터페이스 애플리케이션은 IDirect3DIndexBuffer9 인터페이스의 메서드를 사용하여 인덱스 버퍼 리소스를 조작합니다.
IDirect3DPixelShader9 IDirect3DPixelShader9 인터페이스(d3d9helper.h)는 픽셀 셰이더의 기능을 캡슐화하는 메서드를 제공합니다.
IDirect3DPixelShader9 IDirect3DPixelShader9(d3d9.h) 인터페이스 애플리케이션은 IDirect3DPixelShader9 인터페이스의 메서드를 사용하여 픽셀 셰이더의 기능을 캡슐화합니다.
IDirect3DQuery9 IDirect3DQuery9 인터페이스(d3d9helper.h)는 드라이버에서 비동기 쿼리를 수행하는 메서드를 제공합니다.
IDirect3DQuery9 IDirect3DQuery9(d3d9.h) 인터페이스 애플리케이션은 IDirect3DQuery9 인터페이스의 메서드를 사용하여 드라이버에서 비동기 쿼리를 수행합니다.

IDirect3DResource9 IDirect3DResource9 인터페이스(d3d9helper.h)는 리소스를 쿼리하고 준비하는 메서드를 제공합니다.
IDirect3DResource9 IDirect3DResource9(d3d9.h) 인터페이스는 애플리케이션에서 리소스를 쿼리하고 준비하는 데 사용됩니다.
IDirect3DStateBlock9 IDirect3DStateBlock9 인터페이스(d3d9helper.h)는 렌더링 상태를 캡슐화하는 메서드를 제공합니다.
IDirect3DStateBlock9 IDirect3DStateBlock9(d3d9.h) 인터페이스는 애플리케이션에서 렌더링 상태를 캡슐화하는 데 사용됩니다.
IDirect3DSurface9 IDirect3DSurface9 인터페이스(d3d9helper.h)는 표면을 쿼리하고 준비하는 메서드를 제공합니다.
IDirect3DSurface9 IDirect3DSurface9(d3d9.h) 인터페이스는 애플리케이션에서 표면을 쿼리하고 준비하는 데 사용됩니다.
IDirect3DSwapChain9 IDirect3DSwapChain9 인터페이스(d3d9helper.h)는 스왑 체인을 조작하는 메서드를 제공합니다.
IDirect3DSwapChain9 IDirect3DSwapChain9(d3d9.h) 인터페이스는 애플리케이션에서 스왑 체인을 조작하는 데 사용됩니다.
IDirect3DSwapChain9Ex 애플리케이션은 IDirect3DSwapChain9Ex 인터페이스의 메서드를 사용하여 스왑 체인을 조작합니다.
IDirect3DTexture9 IDirect3DTexture9 인터페이스(d3d9helper.h)는 텍스처 리소스를 조작하는 메서드를 제공합니다.
IDirect3DTexture9

IDirect3DTexture9(d3d9.h) 인터페이스는 애플리케이션에서 텍스처 리소스를 조작하는 데 사용됩니다.
IDirect3DVertexBuffer9 IDirect3DVertexBuffer9 인터페이스(d3d9helper.h)는 꼭짓점 버퍼 리소스를 조작하는 메서드를 제공합니다.
IDirect3DVertexBuffer9 IDirect3DVertexBuffer9(d3d9.h) 인터페이스는 애플리케이션에서 꼭짓점 버퍼 리소스를 조작하는 데 사용됩니다.
IDirect3DVertexDeclaration9 IDirect3DVertexDeclaration9 인터페이스(d3d9helper.h)는 꼭짓점 셰이더 선언을 캡슐화하는 메서드를 제공합니다.
IDirect3DVertexDeclaration9 IDirect3DVertexDeclaration9(d3d9.h) 인터페이스는 애플리케이션에서 꼭짓점 셰이더 선언을 캡슐화하는 데 사용됩니다.
IDirect3DVertexShader9 IDirect3DVertexShader9 인터페이스(d3d9helper.h)는 꼭짓점 셰이더의 기능을 캡슐화하는 메서드를 제공합니다.
IDirect3DVertexShader9 IDirect3DVertexShader9(d3d9.h) 인터페이스는 애플리케이션에서 꼭짓점 셰이더의 기능을 캡슐화하는 데 사용됩니다.
IDirect3DVolume9 IDirect3DVolume9 인터페이스(d3d9helper.h)는 볼륨 리소스를 조작하는 메서드를 제공합니다.
IDirect3DVolume9 IDirect3DVolume9(d3d9.h) 인터페이스는 애플리케이션에서 볼륨 리소스를 조작하는 데 사용됩니다.
IDirect3DVolumeTexture9 IDirect3DVolumeTexture9 인터페이스(d3d9helper.h)는 볼륨 텍스처 리소스를 조작하는 메서드를 제공합니다.
IDirect3DVolumeTexture9

IDirect3DVolumeTexture9(d3d9.h) 인터페이스는 애플리케이션에서 볼륨 텍스처 리소스를 조작하는 데 사용됩니다.

구조체

D3DCAPS9 Direct3D 개체를 통해 노출되는 하드웨어의 기능을 나타냅니다.
D3DPSHADERCAPS2_0 픽셀 셰이더 드라이버 대문자입니다.
D3DVSHADERCAPS2_0 꼭짓점 셰이더 대문자입니다.
PALETTEENTRY 논리적 팔레트에서 항목의 색과 사용량을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

DirectComposition

아티클2023. 06. 16.

DirectComposition 기술 개요입니다.

DirectComposition을 개발하려면 다음 헤더가 필요합니다.

- dcomp.h
- dcompanimation.h
- dcomptypes.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- DirectComposition

열거형

COMPOSITION_FRAME_ID_TYPE 작성기 프레임의 상태 지정하는 상수를 정의합니다.
DCOMPOSITION_BACKFACE_VISIBILITY 시각적 개체에 적용할 백페이스 표시 유형을 지정합니다.
DCOMPOSITION_BITMAP_INTERPOLATION_MODE 비트맵의 픽셀이 화면의 픽셀과 정확히 일대일로 정렬되지 않는 변환으로 비트맵이 구성될 때 사용할 보간 모드를 지정합니다.
DCOMPOSITION_BORDER_MODE 비트맵 또는 클립의 가장자리가 정수 좌표와 축으로 정렬되지 않도록 비트맵을 작성하거나 변환으로 클립을 적용할 때 사용할 테두리 모드를 지정합니다.
DCOMPOSITION_COMPOSITE_MODE 시각적 개체의 비트맵 콘텐츠를 렌더링 대상과 혼합하는 데 사용할 모드입니다.
DCOMPOSITION_OPACITY_MODE 시각적 개체의 유효 불투명도 값을 해당 시각적 개체의 콘텐츠 및 자식에 적용하는 방법을 지정합니다.

Functions

AddCubic 애니메이션 함수에 입방형 다항 세그먼트를 추가합니다.
AddRepeat 애니메이션 함수의 지정된 부분이 반복되도록 하는 반복 세그먼트를 추가합니다.
AddSinusoidal 애니메이션 함수에 부비동 세그먼트를 추가합니다.
AddVisual 이 시각적 개체의 자식 목록에 새 자식 시각적 개체를 추가합니다.
BeginDraw 이 Microsoft DirectComposition 표면 개체에서 그리기를 시작합니다.
CheckCompositionTextureSupport 렌더링에 사용하는 Direct3D 디바이스가 컴퍼지션 텍스처를 지원하는지 여부를 쿼리합니다.
CheckDeviceState DirectComposition 디바이스 개체가 여전히 유효한지 여부를 확인합니다.
커밋 이 디바이스에서 보류 중인 모든 DirectComposition 명령을 커밋합니다. (IDCompositionDevice.Commit)
커밋 이 디바이스에서 보류 중인 모든 DirectComposition 명령을 커밋합니다. (IDCompositionDevice2.Commit)
CreateAffineTransform2DEffect IDCompositionAffineTransform2DEffect의 instance 만듭니다.
CreateAnimation 하나 이상의 Microsoft DirectComposition 개체의 하나 이상의 스칼라 속성에 애니메이션 효과를 주는 데 사용되는 애니메이션 개체를 만듭니다. (IDCompositionDevice.CreateAnimation)

CreateAnimation 하나 이상의 Microsoft DirectComposition 개체의 하나 이상의 스칼라 속성에 애니메이션 효과를 주는 데 사용되는 애니메이션 개체를 만듭니다. (IDCompositionDevice2.CreateAnimation)
CreateArithmeticCompositeEffect IDCompositionArithmeticCompositeEffect의 instance 만듭니다.
CreateBlendEffect IDCompositionBlendEffect의 instance 만듭니다.
CreateBrightnessEffect IDCompositionBrightnessEffect의 instance 만듭니다.
CreateColorMatrixEffect IDCompositionColorMatrixEffect의 instance 만듭니다.
CreateCompositeEffect IDCompositionCompositeEffect의 instance 만듭니다.
CreateCompositionTexture 전달된 Direct3D 텍스처를 참조하는 컴퍼지션 텍스처를 만듭니다.
CreateEffectGroup 시각적 하위 트리에 적용할 여러 효과를 나타내는 개체를 만듭니다. (IDCompositionDevice.CreateEffectGroup)
CreateEffectGroup 시각적 하위 트리에 적용할 여러 효과를 나타내는 개체를 만듭니다. (IDCompositionDevice2.CreateEffectGroup)
CreateGaussianBlurEffect IDCompositionGaussianBlurEffect의 instance 만듭니다.
CreateHueRotationEffect IDCompositionHueRotationEffect의 instance 만듭니다.

CreateLinearTransferEffect IDCompositionLinearTransferEffect의 instance 만듭니다.
CreateMatrixTransform 2D 3-by-2 행렬 변환 개체를 만듭니다. (IDCompositionDevice.CreateMatrixTransform)
CreateMatrixTransform 2D 3-by-2 행렬 변환 개체를 만듭니다. (IDCompositionDevice2.CreateMatrixTransform)
CreateMatrixTransform3D 3D 4-by-4 행렬 변환 개체를 만듭니다. (IDCompositionDevice.CreateMatrixTransform3D)
CreateMatrixTransform3D 3D 4-by-4 행렬 변환 개체를 만듭니다. (IDCompositionDevice2.CreateMatrixTransform3D)
CreateRectangleClip 시각적 하위 트리의 렌더링을 사각형 영역으로 제한하는 데 사용할 수 있는 클립 개체를 만듭니다. (IDCompositionDevice.CreateRectangleClip)
CreateRectangleClip 시각적 하위 트리의 렌더링을 사각형 영역으로 제한하는 데 사용할 수 있는 클립 개체를 만듭니다. (IDCompositionDevice2.CreateRectangleClip)
CreateRotateTransform 2D 회전 변환 개체를 만듭니다. (IDCompositionDevice.CreateRotateTransform)
CreateRotateTransform 2D 회전 변환 개체를 만듭니다. (IDCompositionDevice2.CreateRotateTransform)
CreateRotateTransform3D IDCompositionDevice 인터페이스의 CreateRotateTransform3D 메서드는 3D 회전 변환 개체를 만듭니다.
CreateRotateTransform3D 3D 회전 변환 개체를 만듭니다.

CreateSaturationEffect IDCompositionSaturationEffect의 instance 만듭니다.
CreateScaleTransform IDCompositionDevice::CreateScaleTransform 메서드는 2D 배율 변환 개체를 만듭니다.
CreateScaleTransform 2D 배율 변환 개체를 만듭니다.
CreateScaleTransform3D IDCompositionDevice::CreateScaleTransform3D 메서드는 3D 배율 변환 개체를 만듭니다.
CreateScaleTransform3D 3D 배율 변환 개체를 만듭니다.
CreateShadowEffect IDCompositionShadowEffect의 instance 만듭니다.
CreateSkewTransform IDCompositionDevice::CreateSkewTransform 메서드는 2D 기울이기 변환 개체를 만듭니다.
CreateSkewTransform 2D 기울이기 변환 개체를 만듭니다.
CreateSurface IDCompositionDevice::CreateSurface 메서드는 컴퍼지션을 위해 하나 이상의 시각적 개체와 연결할 수 있는 업데이트 가능한 Surface 개체를 만듭니다.
CreateSurface 컴퍼지션을 위해 하나 이상의 시각적 개체와 연결할 수 있는 업데이트 가능한 Surface 개체를 만듭니다.
CreateSurface 컴퍼지션을 위해 하나 이상의 시각적 개체와 연결할 수 있는 Surface 개체를 만듭니다.

CreateSurfaceFactory 다른 DirectComposition 표면 또는 가상 표면 개체를 만드는 데 사용할 수 있는 Microsoft DirectComposition Surface 팩터리 개체를 만듭니다.
CreateSurfaceFromHandle 기존 컴퍼지션 표면을 래핑하는 새 컴퍼지션 표면 개체를 만듭니다.
CreateSurfaceFromHandle IDCompositionDevice::CreateSurfaceFromHandle 메서드는 기존 컴퍼지션 표면을 래핑하는 새 컴퍼지션 표면 개체를 만듭니다.
CreateSurfaceFromHwnd 계층화된 창의 래스터화를 나타내고 컴퍼지션을 위해 시각적 개체와 연결할 수 있는 래퍼 개체를 만듭니다.
CreateSurfaceFromHwnd IDCompositionDevice::CreateSurfaceFromHwnd 메서드는 계층화된 창의 래스터화를 나타내며 컴퍼지션에 대한 시각적 개체와 연결할 수 있는 래퍼 개체를 만듭니다.
CreateTableTransferEffect IDCompositionTableTransferEffect의 instance 만듭니다.
CreateTargetForHwnd 지정된 창 핸들로 표시되는 창에 바인딩된 컴퍼지션 대상 개체를 만듭니다.
CreateTargetForHwnd 지정된 창 핸들(HWND)로 표시되는 창에 바인딩된 컴퍼지션 대상 개체를 만듭니다.
CreateTransform3DGroup IDCompositionDevice::CreateTransform3DGroup 메서드는 3D 변환 개체의 배열을 포함하는 3D 변환 그룹 개체를 만듭니다.
CreateTransform3DGroup 3D 변환 개체의 배열을 포함하는 3D 변환 그룹 개체를 만듭니다.
CreateTransformGroup IDCompositionDevice::CreateTransformGroup 메서드는 2D 변환 개체의 배열을 포함하는 2D 변환 그룹 개체를 만듭니다.

CreateTransformGroup 2D 변환 개체의 배열을 포함하는 2D 변환 그룹 개체를 만듭니다.
CreateTranslateTransform IDCompositionDevice::CreateTranslateTransform 메서드는 2D 변환 변환 개체를 만듭니다.
CreateTranslateTransform 2D 변환 변환 개체를 만듭니다.
CreateTranslateTransform3D IDCompositionDevice::CreateTranslateTransform3D 메서드는 3D 변환 변환 개체를 만듭니다.
CreateTranslateTransform3D 3D 변환 변환 개체를 만듭니다.
CreateTurbulenceEffect IDCompositionTurbulenceEffect의 instance 만듭니다.
CreateVirtualSurface IDCompositionDevice::CreateVirtualSurface 메서드는 컴퍼지션을 위해 하나 이상의 시각적 개체와 연결할 수 있는 드물게 채워진 표면을 만듭니다.
CreateVirtualSurface 컴퍼지션을 위해 하나 이상의 시각적 개체와 연결할 수 있는 채우기가 거의 없는 표면을 만듭니다. (IDCompositionDevice2.CreateVirtualSurface)
CreateVirtualSurface 컴퍼지션을 위해 하나 이상의 시각적 개체와 연결할 수 있는 채우기가 거의 없는 표면을 만듭니다. (IDCompositionSurfaceFactory.CreateVirtualSurface)
CreateVisual IDCompositionDevice::CreateVisual 메서드는 새 시각적 개체를 만듭니다.
CreateVisual 새 시각적 개체를 만듭니다.

DCompositionAttachMouseDragToHwnd Interaction/InputSink를 만들어 마우스 단추를 아래로 라우팅하고 이후의 이동 및 위로 이벤트를 지정된 HWND로 라우팅합니다.
DCompositionAttachMouseWheelToHwnd Interaction/InputSink를 만들어 마우스 휠 메시지를 지정된 HWND로 라우팅합니다.
DCompositionBoostCompositorClock 시스템에서 대기 시간에 민감한 콘텐츠를 향상시키기 위해 더 높은 새로 고침 속도로 동적으로 전환하도록 요청합니다.
DCompositionCreateDevice 다른 Microsoft DirectComposition 개체를 만드는 데 사용할 수 있는 새 디바이스 개체를 만듭니다. (DCompositionCreateDevice)
DCompositionCreateDevice2 다른 Microsoft DirectComposition 개체를 만드는 데 사용할 수 있는 새 디바이스 개체를 만듭니다. (DCompositionCreateDevice2)
DCompositionCreateDevice3 다른 DirectComposition 개체를 만드는 데 사용할 수 있는 새 DirectComposition 디바이스 개체를 만듭니다.
DCompositionCreateSurfaceHandle Microsoft DirectX 스왑 체인 또는 스왑 버퍼에 바인딩되고 시각적 개체와 연결할 수 있는 새 컴퍼지션 표면 개체를 만듭니다.
DCompositionGetFrameId 지정된 형식의 가장 최근 작성자 프레임의 식별자를 가져옵니다.
DCompositionGetStatistics 컴퍼지션 프레임에 대한 기본 정보와 프레임의 일부인 렌더링 대상 ID 목록을 가져옵니다.
DCompositionGetTargetStatistics
DCompositionWaitForCompositorClock 작성자 클록의 다음 신호가 발생할 때까지 스레드를 중지합니다.

DisableDebugCounters 성능 디버깅 카운터 표시를 사용하지 않도록 설정합니다.
DisableHeatMap 시각적 열 지도를 사용하지 않도록 설정합니다.
DisableRedrawRegions 시각적 개체 다시 그리기 영역을 사용하지 않도록 설정합니다.
EnableDebugCounters 성능 디버깅 카운터를 표시할 수 있습니다.
EnableHeatMap 초과 그리기 영역을 나타내는 시각적 열 지도를 사용하도록 설정합니다.
EnableRedrawRegions 콘텐츠를 다시 그릴 때 시각적 개체를 강조 표시할 수 있습니다.
End 애니메이션 함수의 끝을 표시하는 끝 세그먼트를 추가합니다.
EndDraw 이 Microsoft DirectComposition 표면 개체에 그리기의 끝을 표시합니다.
GetAvailableFence 해당 정보가 알려진 경우 컴퍼지션 텍스처를 사용할 수 있게 되는 시기를 나타내는 Direct3D 동기화 펜스/값 쌍을 검색합니다.
GetFrameStatistics IDCompositionDevice::GetFrameStatistics 메서드는 컴퍼지션 엔진에서 컴퍼지션 시간 및 프레임 속도에 대한 정보를 검색합니다.
GetFrameStatistics 컴퍼지션 엔진에서 컴퍼지션 시간 및 프레임 속도에 대한 정보를 검색합니다.
operator!= 두 개체의 값을 비교하여 서로 `COMPOSITION_TARGET_ID` 다른 값이 포함되어 있는지 확인합니다.

operator== 두 `COMPOSITION_TARGET_ID` 개체의 값을 비교하여 동일한 값이 포함되어 있는지 확인합니다.
RemoveAllVisuals 이 시각적 개체의 자식 목록에서 모든 시각적 개체를 제거합니다.
RemoveVisual 이 시각적 개체의 자식 목록에서 자식 시각적 개체를 제거합니다.
Reset 세그먼트가 포함되지 않도록 애니메이션 함수를 다시 설정합니다.
크기 조정 이 가상 표면 개체의 논리적 크기를 변경합니다.
ResumeDraw 이 Microsoft DirectComposition 표면 개체에서 그리기를 다시 시작합니다.
스크롤 Microsoft DirectComposition 논리 표면의 사각형 영역을 스크롤합니다.
SetAbsoluteBeginTime 애니메이션 함수가 시작되는 절대 시간을 설정합니다.
SetAlpha 효과의 알파 값을 설정합니다. (오버로드 2/2)
SetAlpha 효과의 알파 값을 설정합니다. (오버로드 1/2)
SetAlphaDisable 알파 채널에 전송 함수를 적용할지 여부를 지정합니다.
SetAlphaDisable 알파 채널에 전송 함수를 적용할지 여부를 지정합니다.

SetAlphaMode 색 행렬 효과에 대한 출력의 알파 모드를 설정합니다.
SetAlphaMode TBD
SetAlphaSlope IDCompositionLinearTransferEffect::SetAlphaSlope 메서드는 알파 채널에 대한 선형 함수의 기울기를 설정합니다.
SetAlphaSlope IDCompositionLinearTransferEffect::SetAlphaSlope(IDCompositionAnimation) 메서드는 알파 채널에 대한 선형 함수의 기울기를 설정합니다.
SetAlphaTable 알파 채널의 전송 함수를 정의하는 데 사용되는 값 목록을 설정합니다.
SetAlphaTableValue 알파 테이블의 값을 설정합니다. (오버로드 1/2)
SetAlphaTableValue 알파 테이블의 값을 설정합니다. (오버로드 2/2)
SetAlphaYIntercept 알파 채널에 대한 선형 함수의 Y 절편을 설정합니다.
SetAlphaYIntercept 알파 채널에 대한 선형 함수의 Y 절편을 설정합니다.
SetAngle IDCompositionHueRotationEffect::SetAngle(float) 메서드는 색조를 회전하는 각도를 설정합니다.
SetAngle IDCompositionHueRotationEffect::SetAngle(IDCompositionAnimation) 메서드는 색조를 회전하도록 각도를 설정합니다.

SetAngle 2D 회전 변환의 Angle 속성 값을 변경합니다. Angle 속성은 회전 각도를 지정합니다.
SetAngle 2D 회전 변환의 Angle 속성 값에 애니메이션 효과를 옵니다. Angle 속성은 회전 각도를 지정합니다.
SetAngle 3D 회전 변환의 Angle 속성 값을 변경합니다. Angle 속성은 회전 각도를 지정합니다. 기본값은 영입니다.
SetAngle 3D 회전 변환의 Angle 속성 값에 애니메이션 효과를 옵니다. Angle 속성은 회전 각도를 지정합니다. 기본값은 영입니다.
SetAngleX 2D 기울이기 변환의 AngleX 속성 값을 변경합니다.
SetAngleX 2D 기울이기 변환의 AngleX 속성 값에 애니메이션 효과를 주었습니다.
SetAngleY 2D 기울이기 변환의 AngleY 속성 값을 변경합니다.
SetAngleY 2D 기울이기 변환의 AngleY 속성 값에 애니메이션 효과를 주었습니다.
SetAxisX 3D 회전 변환의 AxisX 속성 값을 변경합니다. AxisX 속성은 회전 축 벡터의 x 좌표를 지정합니다. 기본값은 영입니다.
SetAxisX 3D 회전 변환의 AxisX 속성 값에 애니메이션 효과를 옵니다. AxisX 속성은 회전 축 벡터의 x 좌표를 지정합니다. 기본값은 영입니다.
SetAxisy 3D 회전 변환의 AxisY 속성 값을 변경합니다. AxisY 속성은 회전 축 벡터의 y 좌표를 지정합니다. 기본값은 영입니다.

SetAxisy 3D 회전 변환의 AxisY 속성 값에 애니메이션 효과를 옵니다. AxisY 속성은 회전 축 벡터의 y 좌표를 지정합니다. 기본값은 영입니다.
SetAxisZ 3D 회전 변환의 AxisZ 속성 값을 변경합니다. AxisZ 속성은 회전 축 벡터의 z 좌표를 지정합니다. 기본값은 1.0입니다.
SetAxisZ 3D 회전 변환의 AxisZ 속성 값에 애니메이션 효과를 옵니다. AxisZ 속성은 회전 축 벡터의 z 좌표를 지정합니다. 기본값은 1.0입니다.
SetBackFaceVisibility 관찰자를 향할 때 3D 변환이 적용된 표면을 표시할지 여부를 지정합니다.
SetBaseFrequency X 및 Y 방향으로 기본 주파수를 설정합니다.
SetBitmapInterpolationMode 축 맞춤이 아니거나 정확히 크기 조정되지 않은 비트맵에서 픽셀을 보간할 때 사용할 Microsoft DirectComposition 모드를 지정하는 BitmapInterpolationMode 속성을 설정합니다.
SetBlackPoint 밝기 효과에 대한 밝기 전송 곡선의 아래쪽 부분을 지정합니다.
SetBlackPointX 검은색 점의 x 값을 설정합니다. (오버로드 2/2)
SetBlackPointX 검은색 점의 x 값을 설정합니다. (오버로드 1/2)
SetBlackPointY 검은색 점의 y 값을 설정합니다. (오버로드 1/2)
SetBlackPointY 검은색 점의 y 값을 설정합니다. (오버로드 2/2)

SetBlue 그림자 색의 파란색 값을 설정합니다. (오버로드 1/2)
SetBlue 그림자 색의 파란색 값을 설정합니다. (오버로드 2/2)
SetBlueDisable IDCompositionLinearTransferEffect::SetBlueDisable 메서드는 전송 함수를 파란색 채널에 적용할지 여부를 지정합니다.
SetBlueDisable 파란색 채널에 전송 함수를 적용할지 여부를 지정합니다.
SetBlueSlope IDCompositionLinearTransferEffect::SetBlueSlope 메서드는 파란색 채널에 대한 선형 함수의 기울기를 설정합니다.
SetBlueSlope IDCompositionLinearTransferEffect::SetBlueSlope(IDCompositionAnimation) 메서드는 파란색 채널에 대한 선형 함수의 기울기를 설정합니다.
SetBlueTable 파란색 채널의 전송 함수를 정의하는 데 사용되는 값 목록을 설정합니다.
SetBlueTableValue 파란색 테이블의 값을 설정합니다. (오버로드 2/2)
SetBlueTableValue 파란색 테이블의 값을 설정합니다. (오버로드 1/2)
SetBlueYIntercept IDCompositionLinearTransferEffect::SetBlueYIntercept 메서드는 파란색 채널에 대한 선형 함수의 Y 절편을 설정합니다.
SetBlueYIntercept IDCompositionLinearTransferEffect::SetBlueYIntercept(IDCompositionAnimation) 메서드는 파란색 채널에 대한 선형 함수의 Y 절편을 설정합니다.

SetBorderMode 효과와 함께 사용할 테두리 모드를 설정합니다.
SetBorderMode 이미지 테두리를 계산하는 데 사용되는 모드를 설정합니다.
SetBorderMode 이 시각적 개체와 연결된 비트맵 및 클립의 가장자리 또는 이 시각적 개체에 루팅된 하위 트리의 시각적 개체를 구성하는 방법을 지정하는 BorderMode 속성을 설정합니다.
SetBottom 클립 개체의 Bottom 속성 값을 변경합니다.
SetBottom 클립 사각형의 Bottom 속성 값에 애니메이션 효과를 냅니다.
SetBottomLeftRadiusX 이 클립의 BottomLeftRadiusX 속성 값을 변경하거나 애니메이션합니다. BottomLeftRadiusX 속성은 클립의 왼쪽 아래 모서리를 반올림하는 타원의 x 반경을 지정합니다.
SetBottomLeftRadiusX 이 클립의 BottomLeftRadiusX 속성 값에 애니메이션 효과를 냅니다. BottomLeftRadiusX 속성은 클립의 왼쪽 아래 모서리를 반올림하는 타원의 x 반경을 지정합니다.
SetBottomLeftRadiusY 이 클립의 BottomLeftRadiusY 속성 값을 변경하거나 애니메이션합니다. BottomLeftRadiusY 속성은 클립의 왼쪽 아래 모서리를 반올림하는 타원의 y 반경을 지정합니다.
SetBottomLeftRadiusY 이 클립의 BottomLeftRadiusY 속성 값에 애니메이션 효과를 냅니다. BottomLeftRadiusY 속성은 클립의 왼쪽 아래 모서리를 반올림하는 타원의 y 반경을 지정합니다.
SetBottomRightRadiusX 이 클립의 BottomRightRadiusX 속성 값을 변경하거나 애니메이션합니다. BottomRightRadiusX 속성은 클립의 오른쪽 아래 모서리를 반올림하는 타원의 x 반경을 지정합니다.

SetBottomRightRadiusX 이 클립의 BottomRightRadiusX 속성 값에 애니메이션 효과를 냅니다. BottomRightRadiusX 속성은 클립의 오른쪽 아래 모서리를 반올림하는 타원의 x 반경을 지정합니다.
SetBottomRightRadiusY 이 클립의 BottomRightRadiusY 속성 값을 변경하거나 애니메이션합니다. BottomRightRadiusY 속성은 클립의 오른쪽 아래 모서리를 반올림하는 타원의 y 반경을 지정합니다.
SetBottomRightRadiusY 이 클립의 BottomRightRadiusY 속성 값에 애니메이션 효과를 냅니다. BottomRightRadiusY 속성은 클립의 오른쪽 아래 모서리를 반올림하는 타원의 y 반경을 지정합니다.
SetCenterX 2D 회전 변환의 CenterX 속성 값을 변경합니다.
SetCenterX 2D 회전 변환의 CenterX 속성 값에 애니메이션 효과를 옵니다.
SetCenterX 3D 회전 변환의 CenterX 속성 값을 변경합니다. CenterX 속성은 회전이 수행되는 지점의 x 좌표를 지정합니다. 기본값은 영입니다.
SetCenterX 3D 회전 변환의 CenterX 속성 값에 애니메이션 효과를 옵니다. CenterX 속성은 회전이 수행되는 지점의 x 좌표를 지정합니다. 기본값은 영입니다.
SetCenterX 2D 배율 변환의 CenterX 속성 값을 변경합니다.
SetCenterX 2D 배율 변환의 CenterX 속성 값에 애니메이션 효과를 제공합니다.
SetCenterX 3D 배율 변환의 CenterX 속성 값을 변경합니다.
SetCenterX 3D 배율 변환의 CenterX 속성 값에 애니메이션 효과를 제공합니다.

SetCenterX 2D 기울이기 변환의 CenterX 속성 값을 변경합니다.
SetCenterX 2D 기울이기 변환의 CenterX 속성 값에 애니메이션 효과를 시킵니다.
SetCentery 2D 회전 변환의 CenterY 속성 값을 변경합니다.
SetCentery 2D 회전 변환의 CenterY 속성 값에 애니메이션 효과를 옵니다.
SetCentery 3D 회전 변환의 CenterY 속성 값을 변경합니다. CenterY 속성은 회전이 수행되는 지점의 y 좌표를 지정합니다. 기본값은 영입니다.
SetCentery 3D 회전 변환의 CenterY 속성 값에 애니메이션 효과를 옵니다. CenterY 속성은 회전이 수행되는 지점의 y 좌표를 지정합니다. 기본값은 영입니다.
SetCentery 2D 배율 변환의 CenterY 속성 값을 변경합니다.
SetCentery 2D 배율 변환의 CenterY 속성 값에 애니메이션 효과를 제공합니다.
SetCentery 3D 배율 변환의 CenterY 속성 값을 변경합니다.
SetCentery 3D 배율 변환의 CenterY 속성 값에 애니메이션 효과를 제공합니다.
SetCentery 2D 기울이기 변환의 CenterY 속성 값을 변경합니다.
SetCentery 2D 기울이기 변환의 CenterY 속성 값에 애니메이션 효과를 시킵니다.

SetCenterZ 3D 회전 변환의 CenterZ 속성 값을 변경합니다. CenterZ 속성은 회전이 수행되는 지점의 z 좌표를 지정합니다. 기본값은 영입니다.
SetCenterZ 3D 회전 변환의 CenterZ 속성 값에 애니메이션 효과를 옵니다. CenterZ 속성은 회전이 수행되는 지점의 z 좌표를 지정합니다. 기본값은 영입니다.
SetCenterZ 3D 배율 변환의 CenterZ 속성 값을 변경합니다.
SetCenterZ 3D 배율 변환의 CenterZ 속성 값에 애니메이션 효과를 제공합니다.
SetClampOutput 효과가 그래프의 다음 효과에 값을 전달하기 전에 색 값을 클램핑할지 여부를 지정합니다.
SetClampOutput 효과가 체인의 다음 효과에 값을 전달하기 전에 효과가 색 값을 0에서 1 사이로 고정하는지 여부를 지정합니다.
SetClampOutput IDCompositionLinearTransferEffect::SetClampOutput 메서드는 효과가 그래프의 다음 효과에 값을 전달하기 전에 색 값을 0에서 1 사이로 고정하는지 여부를 지정합니다.
SetClampOutput 효과가 그래프의 다음 효과에 값을 전달하기 전에 효과가 색 값을 0에서 1 사이로 고정할지 여부를 지정합니다.
SetClip 이 시각적 개체의 Clip 속성을 지정된 사각형으로 설정합니다.
SetClip 이 시각적 개체의 Clip 속성을 지정된 클립 개체로 설정합니다.
SetCoefficient1 두 입력 이미지를 합성하는 데 사용되는 수식의 첫 번째 계수를 설정합니다. (오버로드 2/2)

SetCoefficient1 두 입력 이미지를 합성하는 데 사용되는 수식의 첫 번째 계수를 설정합니다. (오버로드 1/2)
SetCoefficient2 두 입력 이미지를 합성하는 데 사용되는 수식의 두 번째 계수를 설정합니다. (오버로드 1/2)
SetCoefficient2 두 입력 이미지를 합성하는 데 사용되는 수식의 두 번째 계수를 설정합니다. (오버로드 2/2)
SetCoefficient3 두 입력 이미지를 합성하는 데 사용되는 수식의 세 번째 계수를 설정합니다. (오버로드 1/2)
SetCoefficient3 두 입력 이미지를 합성하는 데 사용되는 수식의 세 번째 계수를 설정합니다. (오버로드 2/2)
SetCoefficient4 두 입력 이미지를 합성하는 데 사용되는 수식의 네 번째 계수를 설정합니다. (오버로드 2/2)
SetCoefficient4 두 입력 이미지를 합성하는 데 사용되는 수식의 네 번째 계수를 설정합니다. (오버로드 1/2)
SetCoefficients 두 입력 이미지를 합성하는 데 사용되는 수식의 계수를 설정합니다.
SetColor 그림자의 색을 설정합니다.
SetColorSpace TBD
SetCompositeMode 이 시각적 개체의 혼합 모드를 설정합니다.
SetContent 이 시각적 개체의 Content 속성을 지정된 비트맵 또는 창 래퍼로 설정합니다.

SetDepthMode 이 시각적 개체와 연결된 깊이 모드 속성을 설정합니다.
SetEffect 이 시각적 개체의 Effect 속성을 설정합니다.
SetGreen 그림자 색의 녹색 값을 설정합니다. (오버로드 1/2)
SetGreen 그림자 색의 녹색 값을 설정합니다. (오버로드 2/2)
SetGreenDisable IDCompositionLinearTransferEffect::SetGreenDisable 메서드는 전송 함수를 녹색 채널에 적용할지 여부를 지정합니다.
SetGreenDisable 전송 함수를 녹색 채널에 적용할지 여부를 지정합니다.
SetGreenSlope IDCompositionLinearTransferEffect::SetGreenSlope 메서드는 녹색 채널에 대한 선형 함수의 기울기를 설정합니다.
SetGreenSlope IDCompositionLinearTransferEffect::SetGreenSlope(IDCompositionAnimation) 메서드는 녹색 채널에 대한 선형 함수의 기울기를 설정합니다.
SetGreenTable 녹색 채널의 전송 함수를 정의하는 데 사용되는 값 목록을 설정합니다.
SetGreenTableValue 녹색 테이블의 값을 설정합니다. (오버로드 2/2)
SetGreenTableValue 녹색 테이블의 값을 설정합니다. (오버로드 1/2)

SetGreenYIntercept IDCompositionLinearTransferEffect::SetGreenYIntercept 메서드는 녹색 채널에 대한 선형 함수의 Y 절편을 설정합니다.
SetGreenYIntercept 녹색 채널에 대한 선형 함수의 Y 절편을 설정합니다.
SetInput 인덱스에서 입력을 지정된 필터 효과로 설정합니다.
SetInterpolationMode 효과의 보간 모드를 설정합니다.
SetLeft 클립 사각형의 Left 속성 값을 변경합니다.
SetLeft 클립 사각형의 Left 속성 값에 애니메이션 효과를 냅니다.
SetMatrix 효과에서 이미지의 RGBA 값을 곱하기 위해 사용하는 행렬을 설정합니다.
SetMatrix 이 2D 변환의 행렬 값을 모두 변경합니다.
SetMatrix 이 3D 변환 효과의 행렬 값을 모두 변경합니다.
SetMatrixElement 색 행렬의 요소를 설정합니다. (오버로드 1/2)
SetMatrixElement 색 행렬의 요소를 설정합니다. (오버로드 2/2)
SetMatrixElement 이 변환의 행렬에 있는 한 요소의 값을 변경합니다.

SetMatrixElement 이 2D 변환의 행렬에 있는 한 요소의 값에 애니메이션 효과를 적용합니다.
SetMatrixElement 이 3D 변환 행렬의 한 요소 값을 변경합니다.
SetMatrixElement 이 3D 변환의 행렬에 있는 한 요소의 값에 애니메이션 효과를 적용합니다.
SetMode 혼합 효과가 두 이미지를 결합할 때 사용할 혼합 모드를 설정합니다.
SetMode 복합 효과에 대한 모드를 설정합니다.
SetNoise 난기류 노이즈 모드를 설정합니다.
SetNumOctaves 노이즈 함수의 옥타브 수를 설정합니다.
SetOffset 난기류 출력이 생성되는 좌표를 설정합니다.
SetOffsetX 2D 변환 변환의 OffsetX 속성 값을 변경합니다.
SetOffsetX 2D 변환 변환의 OffsetX 속성 값에 애니메이션 효과를 주세요.
SetOffsetX 3D 변환 변환 효과의 OffsetX 속성 값을 변경합니다. OffsetX 속성은 x축을 따라 변환할 거리를 지정합니다.
SetOffsetX 3D 변환 변환 효과의 OffsetX 속성 값에 애니메이션 효과를 적용합니다. OffsetX 속성은 x축을 따라 변환할 거리를 지정합니다.

SetOffsetX 이 시각적 개체의 OffsetX 속성 값을 변경합니다. (오버로드 1/2)
SetOffsetX 이 시각적 개체의 OffsetX 속성 값을 변경합니다. (오버로드 2/2)
SetOffsetY 2D 변환 변환의 OffsetY 속성 값을 변경합니다.
SetOffsetY 2D 변환 변환의 OffsetY 속성 값에 애니메이션 효과를 주세요.
SetOffsetY 3D 변환 변환 효과의 OffsetY 속성 값을 변경합니다. OffsetY 속성은 y축을 따라 변환할 거리를 지정합니다.
SetOffsetY 3D 변환 변환 효과의 OffsetY 속성 값에 애니메이션 효과를 적용합니다. OffsetY 속성은 y축을 따라 변환할 거리를 지정합니다.
SetOffsetY 이 시각적 개체의 OffsetY 속성 값을 변경합니다.
SetOffsetY 이 시각적 개체의 OffsetY 속성 값에 애니메이션 효과를 옵니다.
SetOffsetZ 3D 변환 변환 효과의 OffsetZ 속성 값을 변경합니다. OffsetZ 속성은 z축을 따라 변환할 거리를 지정합니다.
SetOffsetZ 3D 변환 변환 효과의 OffsetZ 속성 값에 애니메이션 효과를 적용합니다. OffsetZ 속성은 z축을 따라 변환할 거리를 지정합니다.
SetOffsetZ OffsetZ 속성의 값을 변경합니다.

SetOffsetZ OffsetZ 속성의 값에 애니메이션 효과를 옵니다.
SetOpacity Opacity 속성의 값을 변경합니다.
SetOpacity Opacity 속성의 값에 애니메이션 효과를 불어넣습니다.
SetOpacity 시각적 개체의 불투명도 속성 값을 설정합니다.
SetOpacity 시각적 개체의 불투명도 속성 값에 애니메이션 효과를 옵니다.
SetOpacityMode 이 시각적 개체의 불투명도 모드를 설정합니다.
SetRed 그림자 색의 빨간색 값을 설정합니다. (오버로드 2/2)
SetRed 그림자 색의 빨간색 값을 설정합니다. (오버로드 1/2)
SetRedDisable 전송 함수를 빨간색 채널에 적용할지 여부를 지정합니다. (IDCompositionLinearTransferEffect.SetRedDisable)
SetRedDisable 전송 함수를 빨간색 채널에 적용할지 여부를 지정합니다. (IDCompositionTableTransferEffect.SetRedDisable)
SetRedSlope 빨간색 채널에 대한 선형 함수의 기울기를 설정합니다. (오버로드 2/2)
SetRedSlope 빨간색 채널에 대한 선형 함수의 기울기를 설정합니다. (오버로드 1/2)

SetRedTable 빨간색 채널의 전송 함수를 정의하는 데 사용되는 값 목록을 설정합니다.
SetRedTableValue 빨간색 테이블의 값을 설정합니다. (오버로드 1/2)
SetRedTableValue 빨간색 테이블의 값을 설정합니다. (오버로드 2/2)
SetRedYIntercept 빨간색 채널에 대한 선형 함수의 Y 절편을 설정합니다. (오버로드 1/2)
SetRedYIntercept 빨간색 채널에 대한 선형 함수의 Y 절편을 설정합니다. (오버로드 2/2)
SetRight 클립 사각형의 Right 속성 값을 변경합니다.
SetRight 클립 사각형의 Right 속성 값에 애니메이션 효과를 냅니다.
SetRoot 시각적 개체를 시각적 트리의 새 루트 개체로 설정합니다.
SetSaturation 이미지의 채도를 설정합니다. (오버로드 2/2)
SetSaturation 이미지의 채도를 설정합니다. (오버로드 1/2)
SetScaleX 2D 배율 변환의 ScaleX 속성 값을 변경합니다.
SetScaleX 2D 배율 변환의 ScaleX 속성 값에 애니메이션 효과를 제공합니다.

SetScaleX 3D 배율 변환의 ScaleX 속성 값을 변경합니다.
SetScaleX 배율 변환의 ScaleX 속성 값에 애니메이션 효과를 제공합니다.
SetScaley 2D 배율 변환의 ScaleY 속성 값을 변경합니다.
SetScaley 2D 배율 변환의 ScaleY 속성 값에 애니메이션 효과를 제공합니다.
SetScaley 3D 배율 변환의 ScaleY 속성 값을 변경합니다.
SetScaley 배율 변환의 ScaleY 속성 값에 애니메이션 효과를 제공합니다.
SetScaleZ 3D 배율 변환의 ScaleZ 속성 값을 변경합니다.
SetScaleZ 배율 변환의 ScaleZ 속성 값에 애니메이션 효과를 제공합니다.
SetSeed 의사 임의 생성기의 시드를 설정합니다.
SetSharpness 효과의 선명도를 설정합니다. (오버로드 2/2)
SetSharpness 효과의 선명도를 설정합니다. (오버로드 1/2)
SetSize 난기류 출력의 크기를 설정합니다.

SetSourceRect 컴퍼지션 텍스처가 나타내는 Direct3D 텍스처의 영역을 지정합니다.
SetStandardDeviation IDCompositionGaussianBlurEffect::SetStandardDeviation(float) 메서드는 이미지에 적용할 흐림 효과를 설정합니다.
SetStandardDeviation IDCompositionGaussianBlurEffect::SetStandardDeviation(IDCompositionAnimation) 메서드는 이미지에 적용할 흐림 효과를 설정합니다.
SetStandardDeviation 이미지의 알파 채널에 적용할 흐림 효과를 설정합니다. (오버로드 1/2)
SetStandardDeviation 이미지의 알파 채널에 적용할 흐림 효과를 설정합니다. (오버로드 2/2)
SetStitchable 연결이 켜지거나 꺼져 있는지 여부를 지정합니다.
SetTop 클립 사각형의 Top 속성 값을 변경합니다.
SetTop 클립 사각형의 Top 속성 값에 애니메이션 효과를 냅니다.
SetTopLeftRadiusX 이 클립의 TopLeftRadiusX 속성 값을 변경하거나 애니메이션합니다. TopLeftRadiusX 속성은 클립의 왼쪽 위 모서리를 반올림하는 타원의 x 반경을 지정합니다.
SetTopLeftRadiusX 이 클립의 TopLeftRadiusX 속성 값에 애니메이션 효과를 냅니다. TopLeftRadiusX 속성은 클립의 왼쪽 위 모서리를 반올림하는 타원의 x 반경을 지정합니다.
SetTopLeftRadiusY 이 클립의 TopLeftRadiusY 속성 값을 변경하거나 애니메이션합니다. TopLeftRadiusY 속성은 클립의 왼쪽 위 모서리를 반올림하는 타원의 y 반경을 지정합니다.

SetTopLeftRadiusY 이 클립의 TopLeftRadiusY 속성 값에 애니메이션 효과를 냅니다. TopLeftRadiusY 속성은 클립의 왼쪽 위 모서리를 반올림하는 타원의 y 반경을 지정합니다.
SetTopRightRadiusX 이 클립의 TopRightRadiusX 속성 값을 변경하거나 애니메이션합니다. TopRightRadiusX 속성은 클립의 오른쪽 위 모서리를 반올림하는 타원의 x 반경을 지정합니다.
SetTopRightRadiusX 이 클립의 TopRightRadiusX 속성 값에 애니메이션 효과를 냅니다. TopRightRadiusX 속성은 클립의 오른쪽 위 모서리를 반올림하는 타원의 x 반경을 지정합니다.
SetTopRightRadiusY 이 클립의 TopRightRadiusY 속성 값을 변경하거나 애니메이션합니다. TopRightRadiusY 속성은 클립의 오른쪽 위 모서리를 반올림하는 타원의 y 반경을 지정합니다.
SetTopRightRadiusY 이 클립의 TopRightRadiusY 속성 값에 애니메이션 효과를 냅니다. TopRightRadiusY 속성은 클립의 오른쪽 위 모서리를 반올림하는 타원의 y 반경을 지정합니다.
SetTransform 이 시각적 개체의 Transform 속성을 지정된 3-by-2 변환 행렬로 설정합니다.
SetTransform 이 시각적 개체의 Transform 속성을 지정된 2D 변환 개체로 설정합니다.
SetTransform 이 시각적 개체의 Transform 속성을 지정된 4-by-4 변환 행렬로 설정합니다.
SetTransform 이 시각적 개체의 Transform 속성을 지정된 3D 변환 개체로 설정합니다.
SetTransform3D 이 효과 그룹이 적용되는 시각적 개체의 래스터화를 수정하는 3D 변환 효과 개체를 설정합니다.
SetTransformMatrix 효과의 변환 행렬을 설정합니다.

SetTransformMatrixElement 효과의 변환 행렬 요소를 설정합니다. (오버로드 1/2)
SetTransformMatrixElement 효과의 변환 행렬 요소를 설정합니다. (오버로드 2/2)
SetTransformParent 이 시각적 개체의 TransformParent 속성을 설정합니다. TransformParent 속성은 이 시각적 개체가 구성된 상대적인 좌표계를 설정합니다.
SetVisible 시각적 개체의 Visible 속성 값을 변경합니다.
SetWhitePoint 밝기 전송 곡선의 위쪽 부분을 설정합니다.
SetWhitePointX 흰색 점의 x 값을 설정합니다. (오버로드 2/2)
SetWhitePointX 흰색 점의 x 값을 설정합니다. (오버로드 1/2)
SetWhitePointy 흰색 점의 y 값을 설정합니다. (오버로드 2/2)
SetWhitePointy 흰색 점의 y 값을 설정합니다. (오버로드 1/2)
SuspendDraw 이 Microsoft DirectComposition 표면 개체에서 그리기를 일시 중단합니다.
자르기 지정된 트리밍 사각형 외부에 있는 픽셀을 삭제합니다.
WaitForCommitCompletion 컴퍼지션 엔진이 IDCompositionDevice::Commit 메서드에 대한 이전 호출 처리를 완료할 때까지 기다립니다.

WaitForCommitCompletion 컴퍼지션 엔진이 IDCompositionDevice2::Commit 메서드에 대한 이전 호출 처리를 완료할 때까지 기다립니다.

인터페이스

IDCompositionAffineTransform2DEffect 산술 복합 효과는 입력 이미지에서 픽셀의 가중치 합계를 사용하여 2개의 이미지를 결합하는 데 사용됩니다. (IDCompositionAffineTransform2DEffect)
IDCompositionAnimation 하나 이상의 Microsoft DirectComposition 개체 속성에 애니메이션 효과를 주는 함수를 나타냅니다.
IDCompositionArithmeticCompositeEffect 산술 복합 효과는 입력 이미지에서 픽셀의 가중치 합계를 사용하여 2개의 이미지를 결합하는 데 사용됩니다. (IDCompositionArithmeticCompositeEffect)
IDCompositionBlendEffect 혼합 효과는 2개 이미지를 결합하는 데 사용됩니다.
IDCompositionBrightnessEffect 밝기 효과는 이미지의 밝기를 제어합니다.
IDCompositionClip 시각적 하위 트리의 렌더링을 사각형 영역으로 제한하는 데 사용되는 클립 개체를 나타냅니다.
IDCompositionColorMatrixEffect 색 행렬 효과는 비트맵의 RGBA 값을 변경합니다.
IDCompositionCompositeEffect 복합 효과는 2개 이상의 이미지를 결합하는 데 사용됩니다.

IDCompositionDesktopDevice Win32 데스크톱 애플리케이션에서 DirectComposition을 사용하려면 애플리케이션에서 IDCompositionDesktopDevice 인터페이스를 사용해야 합니다.
IDCompositionDevice 다른 모든 Microsoft DirectComposition 개체에 대한 팩터리 역할을 하며 트랜잭션 컴퍼지션을 제어하는 메서드를 제공합니다. (IDCompositionDevice)
IDCompositionDevice2 다른 모든 Microsoft DirectComposition 개체에 대한 팩터리 역할을 하며 트랜잭션 컴퍼지션을 제어하는 메서드를 제공합니다. (IDCompositionDevice2)
IDCompositionDevice3 다른 모든 Microsoft DirectComposition 개체에 대한 팩터리 역할을 하며 트랜잭션 컴퍼지션을 제어하는 메서드를 제공합니다. (IDCompositionDevice3)
IDCompositionDevice4 컴퍼지션 텍스처의 루트 팩터리 역할을 합니다.
IDCompositionDeviceDebug 애플리케이션 디버깅 및 성능 튜닝에 도움이 되는 렌더링 기능에 대한 액세스를 제공합니다. 이 인터페이스는 DirectComposition 디바이스 인터페이스에서 쿼리할 수 있습니다.
IDCompositionEffect 시각적 개체 하위 트리의 래스터화를 수정하는 비트맵 효과를 나타냅니다.
IDCompositionEffectGroup 시각적 개체 하위 트리의 래스터화를 수정하기 위해 함께 적용되는 비트맵 효과 그룹을 나타냅니다.
IDCompositionFilterEffect 필터 효과를 나타냅니다.
IDCompositionGaussianBlurEffect Gaussian 흐림 효과는 일반적으로 이미지 노이즈를 줄이고 세부 정보를 줄이기 위해 Gaussian 함수에 의해 이미지를 흐리게 하는 데 사용됩니다.

IDCompositionHueRotationEffect 색조 회전 효과는 회전 각도에 따라 색 행렬을 적용하여 이미지의 색조를 변경합니다.
IDCompositionLinearTransferEffect 선형 전송 효과는 각 채널에 대해 제공하는 값 목록에서 만든 선형 함수를 사용하여 이미지의 색 강도를 매핑하는 데 사용됩니다.
IDCompositionMatrixTransform 3-by-2 행렬로 정의된 임의의 아핀 2D 변환을 나타냅니다. (IDCompositionMatrixTransform)
IDCompositionMatrixTransform3D 4-by-4 행렬로 정의된 임의의 3D 변환을 나타냅니다.
IDCompositionRectangleClip 시각적 하위 트리의 렌더링을 지정된 사각형 영역으로 제한하는 클립 개체를 나타냅니다. 필요에 따라 클립 개체에 둥근 모서리가 지정되어 있을 수 있습니다.
IDCompositionRotateTransform z축 주위의 시각적 개체 회전에 영향을 주는 2D 변환을 나타냅니다. 좌표계는 지정된 중심점을 중심으로 회전됩니다.
IDCompositionRotateTransform3D 3D 공간에서 임의의 축을 따라 시각적 개체의 회전에 영향을 주는 3D 변환을 나타냅니다. 좌표계는 지정된 중심점을 중심으로 회전됩니다.
IDCompositionSaturationEffect 이 효과는 이미지의 채도를 변경하는 데 사용됩니다. 채도 효과는 색 행렬 효과의 전문화입니다.
IDCompositionScaleTransform x축 및 y축을 따라 시각적 개체의 배율에 영향을 주는 2D 변환을 나타냅니다. 좌표계는 지정된 중심점에서 크기가 조정됩니다.
IDCompositionScaleTransform3D x축, y축 및 z축을 따라 시각적 개체의 눈금에 영향을 주는 3D 변환 효과를 나타냅니다. 좌표계는 지정된 중심점에서 크기가 조정됩니다.

IDCompositionShadowEffect 그림자 효과는 이미지의 알파 채널에서 그림자를 생성하는 데 사용됩니다. 그림자는 알파 값이 높을수록 불투명하고 알파 값이 낮을수록 투명합니다. 흐림 효과의 양과 그림자의 색을 설정할 수 있습니다.
IDCompositionSkewTransform x축 및 y축을 따라 시각적 개체의 기울이기 에 영향을 주는 2D 변환을 나타냅니다. 좌표계가 지정된 중심점 주위로 기울어져 있습니다.
IDCompositionSurface 시각적 트리에서 컴퍼지션을 위해 시각적 개체와 연결할 수 있는 물리적 비트맵을 나타냅니다. 이 인터페이스를 사용하여 비트맵 콘텐츠를 업데이트할 수도 있습니다.
IDCompositionSurfaceFactory 애플리케이션 제공 렌더링 디바이스와 연결된 표면 및 가상 표면 개체를 만듭니다.
IDCompositionTableTransferEffect 테이블 전송 효과는 제공하는 값 목록을 보간하여 만든 전송 함수를 사용하여 이미지의 색 강도를 매핑하는 데 사용됩니다.
IDCompositionTarget Microsoft DirectComposition 시각적 트리와 시각적 트리를 구성해야 하는 대상 간의 바인딩을 나타냅니다.
IDCompositionTexture DComp 시각적 개체에 콘텐츠로 바인딩할 수 있는 원시 Direct3D 텍스처를 나타내는 개체에 대한 인터페이스입니다.
IDCompositionTransform 시각적 하위 트리의 좌표 공간을 수정하는 데 사용할 수 있는 2D 변환을 나타냅니다.
IDCompositionTransform3D 시각적 하위 트리의 래스터화를 수정하는 데 사용할 수 있는 3D 변환 효과를 나타냅니다.
IDCompositionTranslateTransform x축 및 y축을 따라 시각적 개체의 오프셋에만 영향을 주는 2D 변환을 나타냅니다.

IDCompositionTranslateTransform3D x축, y축 및 z축을 따라 시각적 개체의 오프셋에 영향을 주는 3D 변환을 나타냅니다.
IDCompositionTurbulenceEffect 난기류 효과는 Perlin 노이즈 함수를 기반으로 비트맵을 생성하는 데 사용됩니다. 난기류 효과에는 입력 이미지가 없습니다.
IDCompositionVirtualSurface 시각적 트리에서 컴퍼지션에 대한 시각적 개체와 연결할 수 있는 드물게 할당된 비트맵을 나타냅니다.
IDCompositionVisual Microsoft DirectComposition 시각적 개체를 나타냅니다.
IDCompositionVisual2 시각적 트리의 DirectComposition 시각적 개체 하나를 나타냅니다. (IDCompositionVisual2)
IDCompositionVisual3 시각적 트리의 DirectComposition 시각적 개체 하나를 나타냅니다. (IDCompositionVisual3)
IDCompositionVisualDebug 디버그 시각적 개체를 나타냅니다.

구조체

COMPOSITION_FRAME_STATS 작성자 프레임에 대한 타이밍 및 컴퍼지션 통계를 설명합니다.
COMPOSITION_STATS 타이밍 및 컴퍼지션 정보를 설명합니다.
COMPOSITION_TARGET_ID 컴퍼지션 렌더링 대상에 대한 정보를 포함합니다.

COMPOSITION_TARGET_STATS 컴퍼지션 프레임 및 렌더링 대상에 대한 대상별 정보를 포함합니다.
DCOMPOSITION_FRAME_STATISTICS 프레임의 타이밍 및 컴퍼지션 통계를 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Directdraw

아티클2023. 03. 16.

DirectDraw 기술 개요입니다.

DirectDraw를 개발하려면 다음 헤더가 필요합니다.

- ddraw.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Directdraw

함수

AddAttachedSurface 지정된 z 버퍼 표면을 이 표면에 연결합니다.
AddOverlayDirtyRect IDirectDrawSurface7::AddOverlayDirtyRect 메서드가 현재 구현되지 않았습니다.
샌드위치 비트 블록 전송(비트블렛)을 수행합니다. 이 메서드는 비트blt 작업 중에 z 버퍼링 또는 알파 혼합을 지원하지 않습니다.
BltBatch IDirectDrawSurface7::BltBatch 메서드가 현재 구현되지 않았습니다.
BltFast 원본 색 키 또는 대상 색 키를 사용하여 원본 복사 비트blt 또는 투명한 비트블렛을 수행합니다.
ChangeUniquenessValue 이 표면의 고유성 값을 수동으로 업데이트합니다.
컴팩트 현재 이 메서드는 구현되지 않습니다. (IDirectDraw7.Compact)

CreateClipper DirectDrawClipper 개체를 만듭니다.
CreatePalette 이 DirectDraw 개체에 대한 DirectDrawPalette 개체를 만듭니다.
CreateSurface 이 DirectDraw 개체에 대한 DirectDrawSurface 개체를 만듭니다.
DeleteAttachedSurface 하나 이상의 연결된 표면을 분리합니다.
DirectDrawCreate DirectDraw 개체의 인스턴스를 만듭니다.
DirectDrawCreateClipper DirectDraw 개체와 연결되지 않은 DirectDrawClipper 개체의 인스턴스를 만듭니다.
DirectDrawCreateEx DirectX 7.0에서 Direct3D 인터페이스 집합을 지원하는 DirectDraw 개체의 인스턴스를 만듭니다. DirectX 7.0에서 Direct3D의 기능을 사용하려면 이 함수를 사용하여 DirectDraw 개체를 만듭니다.
DirectDrawEnumerateA 이 함수는 DirectDrawEnumerateEx 함수로 대체됩니다. (ANSI)
DirectDrawEnumerateExA 컴퓨터에 설치된 모든 DirectDraw 디바이스를 열거합니다. NULL 항목은 항상 GDI와 공유되는 기본 디스플레이 디바이스를 식별합니다. (ANSI)
DirectDrawEnumerateExW 컴퓨터에 설치된 모든 DirectDraw 디바이스를 열거합니다. NULL 항목은 항상 GDI와 공유되는 기본 디스플레이 디바이스를 식별합니다. (유니코드)
DirectDrawEnumerateW 이 함수는 DirectDrawEnumerateEx 함수로 대체됩니다. (유니코드)

DuplicateSurface DirectDrawSurface 개체를 복제합니다.
EnumAttachedSurfaces 이 표면에 연결된 모든 표면을 열거합니다.
EnumDisplayModes 하드웨어가 DirectDraw 개체를 통해 노출하고 제공된 표면 설명과 호환되는 모든 표시 모드를 열거합니다.
EnumOverlayZOrders 지정된 대상의 오버레이 표면을 열거합니다. 오버레이를 앞뒤 또는 백투백 순서로 열거할 수 있습니다.
EnumSurfaces 지정된 표면 설명을 충족하는 기존 또는 가능한 모든 표면을 열거합니다.
EvaluateMode IDirectDraw7::StartModeTest를 호출한 후 테스트가 제공하는 각 모드를 통과하거나 실패하고 테스트가 완료될 때까지 모드를 단계별로 진행하는 데 사용됩니다.
Flip DDSCAPS_BACKBUFFER 표면과 연결된 표면 메모리를 전면 버퍼 표면과 연결합니다.
FlipToGDISurface GDI가 기본 화면에 쓰는 표면을 만듭니다.
FreePrivateData 이 표면과 연결된 지정된 프라이빗 데이터를 해제합니다.
GetAttachedSurface 지정된 기능이 있는 연결된 표면을 가져오고 검색된 인터페이스의 참조 수를 증분합니다.
GetAvailableVidMem 사용 가능한 디스플레이 메모리의 총 양과 지정된 유형의 화면에 대해 현재 사용 가능한 디스플레이 메모리 양을 검색합니다.

GetBltStatus 비트 블록 전송(비트blt) 작업에 대한 상태를 가져옵니다.
GetCaps 하드웨어 및 HEL(하드웨어 에뮬레이션 계층)에 대한 디바이스 드라이버의 기능을 검색합니다.
GetCaps 색상표 개체의 기능을 검색합니다.
GetCaps 이 표면의 기능을 검색합니다. 이러한 기능은 반드시 디스플레이 디바이스의 기능과 관련이 있는 것은 아닙니다.
GetClipList DirectDrawClipper 개체와 연결된 클립 목록의 복사본을 검색합니다. 클립 목록의 하위 집합을 선택하려면 클립 목록을 클립하는 사각형을 전달할 수 있습니다.
GetClipper 이 표면과 연결된 DirectDrawClipper 개체를 검색하고 반환된 클리퍼의 참조 수를 증분합니다.
GetColorControls 오버레이 또는 기본 표면과 연결된 현재 색 제어 설정을 검색합니다.
GetColorKey 이 표면의 색 키 값을 검색합니다.
Getdc 이 화면에 대한 디바이스 컨텍스트의 GDI 호환 핸들을 만듭니다.
GetDDInterface 이 화면을 만드는 데 사용된 DirectDraw 개체에 대한 인터페이스를 검색합니다.
GetDeviceIdentifier 디바이스 드라이버에 대한 정보를 가져옵니다. 이 방법은 드라이버 또는 칩셋 동작 불량에 대한 해결 방법을 구현하기 위해 특정 하드웨어 설치를 인식하기 위해 주의해서 사용할 수 있습니다.

GetDisplayMode 현재 표시 모드를 검색합니다.
GetEntries DirectDrawPalette 개체에서 팔레트 값을 검색합니다.
GetFlipStatus 이 표면이 대칭 이동 프로세스를 완료했는지 여부에 대한 상태를 검색합니다.
GetFourCCCodes DirectDraw 개체에서 지원하는 4자 코드(FOURCC)를 검색합니다. 이 메서드는 지원되는 코드 수를 검색할 수도 있습니다.
GetGammaRamp 기본 표면의 빨간색, 녹색 및 파란색 감마 램프를 검색합니다.
GetGDISurface 현재 GDI가 주 표면으로 처리하는 표면 메모리를 나타내는 DirectDrawSurface 개체를 검색합니다.
GetHWnd IDirectDrawClipper::SetHWnd 메서드를 사용하여 이전에 이 DirectDrawClipper 개체와 연결된 창 핸들을 검색합니다.
GetLOD 관리되는 Mipmap 표면에 대해 현재 설정된 LOD(최대 세부 수준)를 검색합니다. 이 메서드는 관리되는 텍스처에서만 성공합니다.
GetMonitorFrequency DirectDraw 개체가 제어하는 모니터의 빈도를 검색합니다.
GetOverlayPosition 이 표면의 표시 좌표를 검색합니다. 이 메서드는 표시되는 활성 오버레이 표면(즉, DDSCAPS_OVERLAY 플래그가 설정된 표면)에서 사용됩니다.
GetPalette 이 표면과 연결된 DirectDrawPalette 개체를 검색하고 반환된 팔레트의 참조 수를 증분합니다.

GetPixelFormat 이 표면의 색 및 픽셀 형식을 검색합니다.
GetPriority 이 텍스처의 텍스처 관리 우선 순위를 검색합니다. 이 메서드는 관리되는 텍스처에서만 성공합니다.
GetPrivateData 이 표면과 연결된 개인 데이터를 제공된 버퍼에 복사합니다.
GetScanLine 현재 모니터에 그려지고 있는 검사 선을 검색합니다.
GetSurfaceDesc 현재 조건에서 이 표면에 대한 설명을 검색합니다.
GetSurfaceFromDC 해당 GDI 디바이스 컨텍스트 핸들을 기반으로 표면에 대한 IDirectDrawSurface7 인터페이스를 검색합니다.
GetUniquenessValue 이 표면의 현재 고유성 값을 검색합니다.
GetVerticalBlankStatus 세로 공백의 상태를 검색합니다.
초기화 CoCreateInstance COM 함수를 사용하여 만든 DirectDraw 개체를 초기화합니다.
초기화 CoCreateInstance COM 함수를 사용하여 만든 DirectDrawClipper 개체를 초기화합니다.
초기화 DirectDrawPalette 개체를 초기화합니다.
초기화 DirectDrawSurface 개체를 초기화합니다.

IsClipListChanged 창 핸들이 DirectDrawClipper 개체와 연결된 경우 클립 목록의 상태를 검색합니다.
IsLost DirectDrawSurface 개체와 연결된 표면 메모리가 해제되었는지 여부를 확인합니다.
잠금 표면 메모리에 대한 포인터를 가져옵니다.
LPDDENUMCALLBACKA DDEnumCallback 함수는 DirectDrawEnumerate 함수에 대한 애플리케이션 정의 콜백 함수입니다. (ANSI)
LPDDENUMCALLBACKEXA DDEnumCallbackEx 함수는 DirectDrawEnumerateEx 함수에 대한 애플리케이션 정의 콜백 함수입니다. (ANSI)
LPDDENUMCALLBACKEXW DDEnumCallbackEx 함수는 DirectDrawEnumerateEx 함수에 대한 애플리케이션 정의 콜백 함수입니다. (유니코드)
LPDDENUMCALLBACKW DDEnumCallback 함수는 DirectDrawEnumerate 함수에 대한 애플리케이션 정의 콜백 함수입니다. (유니코드)
LPDDENUMMODESCALLBACK 사용하지 마십시오. 이 콜백 함수는 IDirectDraw7::EnumDisplayModes 메서드와 함께 사용되는 EnumModesCallback2 함수로 대체됩니다.
LPDDENUMMODESCALLBACK2 EnumModesCallback2 함수는 IDirectDraw7::EnumDisplayModes 메서드에 대한 애플리케이션 정의 콜백 함수입니다.
LPDDENUMSURFACESCALLBACK LPDDENUMSURFACESCALLBACK 함수(ddraw.h)는 EnumSurfacesCallback7 함수로 대체됩니다.
LPDDENUMSURFACESCALLBACK2 LPDDENUMSURFACESCALLBACK2 함수(ddraw.h)는 EnumSurfacesCallback7 함수로 대체됩니다.

LPDDENUMSURFACESCALLBACK7 EnumSurfacesCallback7 함수는 IDirectDrawSurface7::EnumAttachedSurfaces 및 IDirectDrawSurface7::EnumOverlayZOrders 메서드에 대한 애플리케이션 정의 콜백 함수입니다.
PageLock 시스템 메모리 간에 직접 메모리 액세스(DMA) 전송을 사용하는 비트 블록 전송(비트블렛) 작업이 진행 중인 동안 시스템 메모리 표면이 페이징되지 않도록 합니다.
PageUnlock 시스템 메모리 표면의 잠금을 해제한 다음 페이징할 수 있습니다.
ReleaseDC IDirectDrawSurface7::GetDC 메서드를 사용하여 이전에 가져온 디바이스 컨텍스트의 핸들을 해제합니다.
복원 손실된 표면을 복원합니다. DirectDrawSurface 개체와 연결된 표면 메모리가 해제된 경우에 발생합니다.
RestoreAllSurfaces DirectDraw 개체에 대해 만들어진 모든 표면을 만든 순서대로 복원합니다.
RestoreDisplayMode 기본 표면의 디스플레이 디바이스 하드웨어 모드를 IDirectDraw7::SetDisplayMode 메서드가 호출되기 전의 모드로 다시 설정합니다. 이 메서드를 사용하려면 단독 수준 액세스가 필요합니다.
SetClipList 부모 DirectDrawClipper 개체가 연결된 표면에서 IDirectDrawSurface7::Blt, IDirectDrawSurface7::BltBatch 및 IDirectDrawSurface7::UpdateOverlay 메서드에 사용되는 클립 목록을 설정하거나 삭제합니다.
SetClipper clipper 개체를 이 표면에 연결하거나 삭제합니다.
SetColorControls 오버레이 또는 기본 표면의 색 제어 옵션을 설정합니다.

SetColorKey 하드웨어가 표면별로 색 키를 지원하는 경우 DirectDrawSurface 개체의 색 키 값을 설정합니다.
SetCooperativeLevel 애플리케이션의 최상위 동작을 결정합니다.
SetDisplayMode 디스플레이 디바이스 하드웨어의 모드를 설정합니다.
SetEntries DirectDrawPalette 개체의 항목을 즉시 변경합니다.
SetGammaRamp 기본 표면의 빨간색, 녹색 및 파란색 감마 램프를 설정합니다.
SetHWnd clipper 개체가 클리핑 정보를 가져오는 데 사용하는 창 핸들을 설정합니다.
SetLOD 관리되는 Mipmap 표면의 최대 LOD(세부 수준)를 설정합니다. 이 메서드는 관리되는 텍스처에서만 성공합니다.
SetOverlayPosition 오버레이 표면의 표시 좌표를 변경합니다.
SetPalette 색상표 개체를 표면에 연결하거나 표면에서 분리합니다. 표면은 모든 후속 작업에 이 팔레트를 사용합니다. 색상표 변경은 새로 고침 타이밍과 관계없이 즉시 수행됩니다.
SetPriority 이 텍스처에 대한 텍스처 관리 우선 순위를 할당합니다. 이 메서드는 관리되는 텍스처에서만 성공합니다.
SetPrivateData DirectDraw가 아닌 애플리케이션에서 사용하기 위한 표면과 데이터를 연결합니다. 데이터는 값으로 전달되고 여러 데이터 집합을 단일 표면과 연결할 수 있습니다.

SetSurfaceDesc 기존 표면의 특성을 설정합니다.
StartModeTest 현재 디스플레이 어댑터 및 모니터 조합에 대한 새로 고침 속도 정보로 시스템 레지스트리를 업데이트하는 테스트를 시작합니다.
TestCooperativeLevel 창 또는 전체 화면 애플리케이션에 대한 DirectDraw 디바이스의 현재 협력 수준 상태를 보고합니다.
잠금을 해제 직접 표면 조작이 완료되었다는 것을 DirectDraw에 알립니다.
UpdateOverlay 오버레이 표면의 시각적 특성을 변경하거나 수정합니다. 이러한 표면에는 DDSCAPS_OVERLAY 플래그가 설정되어 있어야 합니다.
UpdateOverlayDisplay IDirectDrawSurface7::UpdateOverlayDisplay 메서드가 현재 구현되지 않았습니다.
UpdateOverlayZOrder 오버레이의 z 순서를 설정합니다.
WaitForVerticalBlank 애플리케이션이 세로 공백 간격과 동기화하는 데 도움이 됩니다.

인터페이스

IDirectDraw7 애플리케이션은 IDirectDraw7 인터페이스의 메서드를 사용하여 DirectDraw 개체를 만들고 시스템 수준 변수를 사용합니다. 이 섹션은 IDirectDraw7 인터페이스의 메서드에 대한 참조입니다.

IDirectDrawClipper 애플리케이션은 IDirectDrawClipper 인터페이스의 메서드를 사용하여 클립 목록을 관리합니다. 이 섹션은 이 인터페이스의 메서드에 대한 참조입니다.
IDirectDrawColorControl 애플리케이션은 IDirectDrawColorControl 인터페이스의 메서드를 사용하여 색 컨트롤을 가져와서 설정합니다.
IDirectDrawGammaControl 애플리케이션은 IDirectDrawGammaControl 인터페이스의 메서드를 사용하여 기본 표면의 빨간색, 녹색 및 파란색 감마 램프 수준을 조정합니다. 이 섹션은 이 인터페이스의 메서드에 대한 참조입니다.
IDirectDrawPalette 애플리케이션은 IDirectDrawPalette 인터페이스의 메서드를 사용하여 DirectDrawPalette 개체를 만들고 시스템 수준 변수를 사용합니다. 이 섹션은 이 인터페이스의 메서드에 대한 참조입니다.
IDirectDrawSurface7 애플리케이션은 IDirectDrawSurface7 인터페이스의 메서드를 사용하여 DirectDrawSurface 개체를 만들고 시스템 수준 변수를 사용합니다. 이 섹션은 이 인터페이스의 메서드에 대한 참조입니다.

구조체

DDCOLORCONTROL DDCOLORCONTROL 구조체는 Microsoft *DirectDraw VPE(비디오 포트 확장)* 개체와 연결된 색 컨트롤을 정의합니다.
DDSURFACEDESC DDSURFACEDESC 구조체에는 드라이버에서 만들 서피스에 대한 설명이 포함되어 있습니다.
DDSURFACEDESC2 DDSURFACEDESC2 구조체에는 드라이버에서 만들 서피스에 대한 설명이 포함되어 있습니다.

DDBLTBATCH DDBLTBATCH 구조체는 비트 블록 전송(비트blt) 작업을 IDirectDrawSurface7::BltBatch 메서드에 전달합니다.
DDBLTFX DDBLTFX 구조체는 래스터 작업(ROP), 효과 및 재정의 정보를 IDirectDrawSurface7::Blt 메서드에 전달합니다. 이 구조체는 IDirectDrawSurface7::BltBatch 메서드와 함께 사용되는 DDBLTBATCH 구조체의 일부이기도 합니다.
DDCAPS_DX3 DDCAPS_DX3 구조체(ddraw.h)는 DirectDraw 개체를 통해 노출되는 하드웨어의 기능을 나타냅니다.
DDCAPS_DX5 DDCAPS_DX5 구조체(ddraw.h)는 DirectDraw 개체를 통해 노출되는 하드웨어의 기능을 나타냅니다.
DDCAPS_DX6 DDCAPS_DX6 구조체(ddraw.h)는 DirectDraw 개체를 통해 노출되는 하드웨어의 기능을 나타냅니다.
DDCAPS_DX7 DDCAPS_DX7 구조체(ddraw.h)는 DirectDraw 개체를 통해 노출되는 하드웨어의 기능을 나타냅니다.
DDCOLORKEY DDCOLORKEY 구조는 원본 색 키, 대상 색 키 또는 색 공간을 설명합니다.
DDDEVICEIDENTIFIER2 DDDEVICEIDENTIFIER2 구조체는 IDirectDraw7::GetDeviceIdentifier 메서드에 전달되어 디바이스에 대한 정보를 가져옵니다.
DDGAMMARAMP DDGAMMARAMP 구조체에는 IDirectDrawGammaControl::GetGammaRamp 및 IDirectDrawGammaControl::SetGammaRamp 메서드에 대한 빨간색, 녹색 및 파란색 램프 데이터가 포함됩니다.

DDOVERLAYFX DDOVERLAYFX 구조체는 오버레이 정보를 IDirectDrawSurface7::UpdateOverlay 메서드에 전달합니다.
DDPIXELFORMAT DDPIXELFORMAT 구조체는 IDirectDrawSurface7::GetPixelFormat 메서드에 대한 DirectDrawSurface 개체의 픽셀 형식을 설명합니다.
DDSCAPS DDSCAPS 구조체는 Microsoft DirectDraw Surface 개체의 기능을 정의합니다.
DDSCAPS2 DDSCAPS2 구조체는 Microsoft DirectDraw 표면 개체의 추가 기능을 정의합니다.

DirectML

아티클2024. 11. 13.

DirectML 기술 개요입니다.

DirectML을 개발하려면 다음 헤더가 필요합니다.

- directml.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- DirectML

열거형

테이블 확장

DML_AXIS_DIRECTION 연산자에 대해 지정된 축을 따라 연산의 방향을 지정하는 상수를 정의합니다(예: 합계, top-k 요소 선택, 최소 요소 선택).
DML_BINDING_TYPE 바인딩 설명(DML_BINDING_DESC 구조)에서 참조하는 리소스의 특성을 지정하는 상수를 정의합니다.
DML_CONVOLUTION_DIRECTION DirectML 컨볼루션 연산자의 방향을 지정하는 상수를 정의합니다 (DML_CONVOLUTION_OPERATOR_DESC 구조에 설명된 대로).
DML_CONVOLUTION_MODE DirectML 구성 연산자에 대한 모드를 지정하는 상수를 정의합니다 (DML_CONVOLUTION_OPERATOR_DESC 구조에 설명된 대로).
DML_CREATE_DEVICE_FLAGS DMLCreateDevice에 추가 디바이스 만들기 옵션을 제공합니다. 값은 비트 OR를 함께 사용할 수 있습니다.
DML_DEPTH_SPACE_ORDER

DirectML 연산자 DML_OPERATOR_DEPTH_TO_SPACE1 및 **DML_OPERATOR_SPACE_TO_DEPTH1**적용된 변환을 제어하는 상수 정의
DML_EXECUTION_FLAGS DirectML에 연산자 실행을 제어하는 옵션을 제공합니다. 이러한 플래그는 한 번에 여러 플래그를 지정하기 위해 함께 비트 OR일 수 있습니다.
DML_FEATURE DirectML 디바이스에서 쿼리할 수 있는 선택적 기능 집합을 정의합니다.
DML_FEATURE_LEVEL DirectML *기능 수준*지정하는 상수를 정의합니다. 기능 수준은 DirectML에서 지원하는 광범위한 기능을 정의합니다.
DML_GRAPH_EDGE_TYPE 그래프 가장자리의 형식을 지정하는 상수를 정의합니다. 이 열거형의 사용은 DML_GRAPH_EDGE_DESC 참조하세요.
DML_GRAPH_NODE_TYPE 그래프 노드의 형식을 지정하는 상수를 정의합니다. 이 열거형의 사용은 DML_GRAPH_NODE_DESC 참조하세요.
DML_INTERPOLATION_MODE DirectML upsample 2-D 연산자의 모드를 지정하는 상수를 정의합니다 (DML_UPSAMPLE_2D_OPERATOR_DESC 구조에 설명된 대로).
DML_IS_INFINITY_MODE 무한대 기호를 지정하는 상수를 정의합니다.
DML_MATRIX_TRANSFORM DirectML 텐서에 적용할 행렬 변환을 지정하는 상수를 정의합니다.
DML_OPERATOR_TYPE 연산자 설명의 형식을 정의합니다.
DML_PADDING_MODE DirectML 패드 연산자의 모드를 지정하는 상수를 정의합니다(DML_PADDING_OPERATOR_DESC 구조에 설명된 대로).

DML_RANDOM_GENERATOR_TYPE 난수 생성기의 형식을 지정하는 상수를 정의합니다.
DML_RECURRENT_NETWORK_DIRECTION 되풀이 DirectML 연산자에 대한 방향을 지정하는 상수를 정의합니다.
DML_REDUCE_FUNCTION DirectML reduce 연산자에 사용할 특정 감소 알고리즘을 지정하는 상수를 정의합니다 (DML_REDUCE_OPERATOR_DESC 구조에 설명된 대로).
DML_ROUNDING_MODE 반올림 모드를 지정하는 상수를 정의합니다.
DML_TENSOR_DATA_TYPE 텐서에 있는 값의 데이터 형식을 지정합니다. DirectML 연산자는 모든 데이터 형식을 지원하지 않을 수 있습니다. 지원하는 데이터 형식을 찾으려면 각 특정 연산자에 대한 설명서를 참조하세요.
DML_TENSOR_FLAGS 텐서 설명에 추가 옵션을 지정합니다. 값은 비트 OR를 함께 사용할 수 있습니다.
DML_TENSOR_TYPE 텐서 설명의 형식을 식별합니다.

함수

테이블 확장

BindInputs 리소스 집합을 입력 텐서로 바인딩합니다.
BindOutputs 리소스 집합을 출력 텐서로 바인딩합니다.
BindPersistentResource

버퍼를 영구 리소스로 바인딩합니다. IDMLDispatchable::GetBindingProperties를 호출하여 이 버퍼 범위의 필요한 크기를 확인할 수 있습니다.
BindTemporaryResource 임시 스크래치 메모리로 사용할 버퍼를 바인딩합니다. IDMLDispatchable::GetBindingProperties를 호출하여 이 버퍼 범위의 필요한 크기를 확인할 수 있습니다.
CheckFeatureSupport DirectML 디바이스에서 지원하는 선택적 기능 및 기능에 대한 정보를 가져옵니다.
CompileGraph DirectML 연산자의 그래프를 GPU에 디스패치할 수 있는 개체로 컴파일합니다.
CompileOperator GPU에 디스패치할 수 있는 개체로 연산자를 컴파일합니다.
CreateBindingTable 파이프라인에 리소스(예: 텐서)를 바인딩하는 데 사용할 수 있는 개체인 바인딩 테이블을 만듭니다.
CreateCommandRecorder DirectML 명령 레코더를 만듭니다.
CreateOperator DirectML 연산자를 만듭니다.
createOperatorInitializer 컴파일된 연산자를 초기화하는 데 사용할 수 있는 개체를 만듭니다.
DMLCreateDevice 지정된 Direct3D 12 디바이스에 대한 DirectML 디바이스를 만듭니다. (DMLCreateDevice)
DMLCreateDevice1 지정된 Direct3D 12 디바이스에 대한 DirectML 디바이스를 만듭니다. (DMLCreateDevice1)
제거 GPU 메모리에서 하나 이상의 페이즈 가능 개체를 제거합니다. IDMLDevice::MakeResident도 참조하세요.

GetBindingProperties 디스패치 가능한 개체(연산자 이니셜라이저 또는 컴파일된 연산자)에 대한 바인딩 속성을 검색합니다.
GetDevice 이 개체를 만드는 데 사용된 DirectML 디바이스를 검색합니다.
GetDeviceRemovedReason DirectML 디바이스가 제거된 이유를 검색합니다.
GetParentDevice 이 DirectML 디바이스를 만드는 데 사용된 Direct3D 12 디바이스를 검색합니다.
GetPrivateData DirectML 디바이스 개체에서 애플리케이션 정의 데이터를 가져옵니다.
메이크리던트 하나 이상의 페이지 가능한 개체가 GPU 메모리에 상주하게 합니다. IDMLDevice::Evict도 참조하세요.
RecordDispatch 디스패치 가능한 개체(연산자 이니셜라이저 또는 컴파일된 연산자)의 실행을 명령 목록에 기록합니다.
다시 설정 바인딩 테이블을 다시 설정하여 다른 연산자 또는 이니셜라이저에 대해 새 설명자 범위를 래핑합니다. 이렇게 하면 바인딩 테이블을 동적으로 다시 사용할 수 있습니다.
다시 설정 새 연산자 집합의 초기화를 처리하도록 이니셜라이저를 다시 설정합니다.
setMuteDebugOutput DirectML에서 ID3D12InfoQueue로 메시지를 보내는 것을 음소거할지 여부를 결정합니다.
SetName 이름을 DirectML 디바이스 개체와 연결합니다. 이 이름은 디버그 진단 및 도구에 사용하기 위한 것입니다.

SetPrivateData 애플리케이션 정의 데이터를 DirectML 디바이스 개체로 설정하고 해당 데이터를 애플리케이션 정의 GUID와 연결합니다.
SetPrivateDataInterface IUnknown 파생 인터페이스를 DirectML 디바이스 개체와 연결하고 해당 인터페이스를 애플리케이션 정의 GUID와 연결합니다.

인터페이스

테이블 확장

IDMLBindingTable 애플리케이션 관리 설명자 힙의 범위를 래핑하고 DirectML에서 리소스에 대한 바인딩을 만드는 데 사용됩니다. 이 개체를 만들려면 IDMLDevice::CreateBindingTable을 호출합니다.
IDMLCommandRecorder DirectML의 디스패치를 Direct3D 12 명령 목록으로 기록합니다.
IDMLCompiledOperator GPU에서 실행하기에 적합한 컴파일된 효율적인 연산자 형식을 나타냅니다. 이 개체를 만들려면 IDMLDevice::CompileOperator를 호출합니다.
IDMLDebugDevice DirectML 디버그 계층을 제어합니다.
IDMLDevice 연산자, 바인딩 테이블, 명령 레코더 및 기타 개체를 만드는 데 사용되는 DirectML 디바이스를 나타냅니다. (IDMLDevice)
IDMLDevice1 연산자, 바인딩 테이블, 명령 레코더 및 기타 개체를 만드는 데 사용되는 DirectML 디바이스를 나타냅니다. (IDMLDevice1)
IDMLDeviceChild

DirectML 디바이스에서 만든 모든 개체에 의해 구현되는 인터페이스입니다.
IDMLDispatchable IDMLCommandRecorder::RecordDispatch를 사용하여 GPU에서 디스패치를 위해 명령 목록에 기록할 수 있는 개체에 의해 구현됩니다.
IDMLObject IDMLDevice 및 IDMLDeviceChild가 직접 상속하는 인터페이스(및 다른 모든 인터페이스는 간접적으로).
IDMLOperator DirectML 연산자를 나타냅니다.
IDMLOperatorInitializer 컴파일된 연산자를 초기화하는 특수한 개체를 나타냅니다. 이 개체의 인스턴스를 만들려면 IDMLDevice::CreateOperatorInitializer를 호출합니다.
IDMLPageable GPU 메모리에서 제거할 수 있으므로 IDMLDevice::Evict 및 IDMLDevice::MakeResident에 제공할 수 있는 개체에 의해 구현됩니다.

구조

테이블 확장

DML_ACTIVATION_CELU_OPERATOR_DESC *InputTensor*모든 요소에 대해 지속적으로 다른 CELU(지수 선형 단위) 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_ELU_OPERATOR_DESC *InputTensor*모든 요소에 대해 ELU(지수 선형 단위) 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_HARD_SIGMOID_OPERATOR_DESC *InputTensor*모든 요소에 대해 하드 시그모이드 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.

DML_ACTIVATION_HARDMAX_OPERATOR_DESC *InputTensor*각 요소에 대해 하드맥스 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_IDENTITY_OPERATOR_DESC id 활성화를 수행하여 *InputTensor* 모든 요소를 *OutputTensor*해당 요소로 효과적으로 복사합니다.
DML_ACTIVATION_LEAKY_RELU_OPERATOR_DESC *InputTensor*모든 요소에 대해 새는 ReLU(정정 선형 단위) 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_LINEAR_OPERATOR_DESC *InputTensor*모든 요소에 대해 선형 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_LOG_SOFTMAX_OPERATOR_DESC *InputTensor*각 요소에 대해 (자연) 소프트맥스 로그 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_PARAMETERIZED_RELU_OPERATOR_DESC *InputTensor*모든 요소에 대해 매개 변수가 있는 ReLU(정정 선형 단위) 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_PARAMETRIC_SOFTPLUS_OPERATOR_DESC *InputTensor*모든 요소에 대해 매개 변수 소프트플러스 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다. (DML_ACTIVATION_PARAMETRIC_SOFTPLUS_OPERATOR_DESC)
DML_ACTIVATION_RELU_GRAD_OPERATOR_DESC ReLU(정정 선형 단위)에 대한 백프로포지션 그라데이션을 계산합니다.
DML_ACTIVATION_RELU_OPERATOR_DESC *InputTensor*모든 요소에 대해 ReLU(정정 선형 단위) 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_SCALED_ELU_OPERATOR_DESC *InputTensor*모든 요소에 대해 확장된 ELU(지수 선형 단위) 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.

DML_ACTIVATION_SCALED_TANH_OPERATOR_DESC *InputTensor*모든 요소에 대해 확장된 하이퍼볼릭 탄젠트 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_SHRINK_OPERATOR_DESC *InputTensor*모든 요소에 대해 축소 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_SIGMOID_OPERATOR_DESC *InputTensor*모든 요소에 대해 시그모이드 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_SOFTMAX_OPERATOR_DESC *InputTensor*대해 softmax 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_SOFTPLUS_OPERATOR_DESC *InputTensor*모든 요소에 대해 매개 변수 소프트플러스 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다. (DML_ACTIVATION_SOFTPLUS_OPERATOR_DESC)
DML_ACTIVATION_SOFTSIGN_OPERATOR_DESC *InputTensor*모든 요소에 대해 softsign 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_TANH_OPERATOR_DESC *InputTensor*모든 요소에 대해 하이퍼볼릭 탄젠트 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_THRESHOLDED_RELU_OPERATOR_DESC *InputTensor*모든 요소에 대해 임계값이 지정된 ReLU(정정 선형 단위) 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ADAM_OPTIMIZER_OPERATOR_DESC Adam(**ADA**ptive **M**오멘트 추정) 알고리즘을 기반으로 제공된 그라데이션을 사용하여 업데이트된 가중치(매개 변수)를 계산합니다. 이 연산자는 최적화 프로그램이며 일반적으로 학습 루프의 가중치 업데이트 단계에서 그라데이션 하강을 수행하는 데 사용됩니다.
DML_ARGMAX_OPERATOR_DESC 입력 텐서의 하나 이상의 차원 내에서 최대값 요소의 인덱스를 출력합니다.

DML_ARGMIN_OPERATOR_DESC 입력 텐서의 하나 이상의 차원 내에서 최소값 요소의 인덱스를 출력합니다.
DML_AVERAGE_POOLING_GRAD_OPERATOR_DESC 평균 풀링에 대한 백프로포지션 그라데이션을 계산합니다 (DML_AVERAGE_POOLING_OPERATOR_DESC참조).
DML_AVERAGE_POOLING_OPERATOR_DESC 입력 텐서의 슬라이딩 윈도우 내 요소에 대한 값의 평균을 계산합니다.
DML_BATCH_NORMALIZATION_GRAD_OPERATOR_DESC 일괄 정규화대한 백프로포지션 그라데이션을 계산합니다.
DML_BATCH_NORMALIZATION_OPERATOR_DESC DML_BATCH_NORMALIZATION_OPERATOR_DESC 구조체(directml.h)는 입력에 대한 일괄 처리 정규화를 수행합니다.
DML_BATCH_NORMALIZATION_TRAINING_GRAD_OPERATOR_DESC 일괄 처리 정규화 학습대한 백프로포지션 그라데이션을 계산합니다.
DML_BATCH_NORMALIZATION_TRAINING_OPERATOR_DESC DML_BATCH_NORMALIZATION_TRAINING_OPERATOR_DESC 구조체(directml.h)는 입력에 대한 일괄 처리 정규화를 수행합니다.
DML_BINDING_DESC IDMLBindingTable 메서드 중 하나에 대한 호출을 통해 바인딩 테이블에 추가할 수 있도록 바인딩에 대한 설명을 포함합니다.
DML_BINDING_PROPERTIES 컴파일된 특정 연산자 또는 연산자 이니셜라이저의 바인딩 요구 사항에 대한 정보를 포함합니다. 이 구조체는 IDMLDispatchable::GetBindingProperties에서 검색됩니다.
DML_BINDING_TABLE_DESC IDMLDevice::CreateBindingTable 및 IDMLBindingTable::Reset에 대한 매개 변수를 지정합니다.
DML_BUFFER_ARRAY_BINDING 개별 버퍼 바인딩의 배열인 리소스 바인딩을 지정합니다.

DML_BUFFER_BINDING ID3D12Resource에 대한 오프셋 및 크기로 표현되는 Direct3D 12 버퍼의 바이트 범위로 설명된 리소스 바인딩을 지정합니다.
DML_BUFFER_TENSOR_DESC Direct3D 12 버퍼 리소스에 저장될 텐서에 대해 설명합니다.
DML_CAST_OPERATOR_DESC 입력의 각 요소를 출력 텐서의 데이터 형식으로 캐스팅하고 결과를 출력의 해당 요소에 저장합니다.
DML_CONVOLUTION_INTEGER_OPERATOR_DESC *InputTensor*사용하여 *FilterTensor* 구성을 수행합니다. 이 연산자는 정수 데이터에 대해 정방향 연결을 수행합니다.
DML_CONVOLUTION_OPERATOR_DESC *InputTensor*사용하여 *FilterTensor* 구성을 수행합니다. 이 연산자는 다양한 표준 나선형 구성을 지원합니다.
DML_CUMULATIVE_PRODUCT_OPERATOR_DESC 축을 따라 텐서의 요소를 곱하여 제품의 실행 중인 집계를 출력 텐서에 기록합니다.
DML_CUMULATIVE_SUMMATION_OPERATOR_DESC 축을 따라 텐서의 요소를 합산하여 합계의 실행 집계를 출력 텐서에 기록합니다.
DML_DEPTH_TO_SPACE_OPERATOR_DESC 깊이에서 공간 데이터 블록으로 데이터를 다시 정렬(permutes)합니다. 연산자는 깊이 차원의 값이 공간 블록에서 높이 및 너비 차원으로 이동되는 입력 텐서의 복사본을 출력합니다. (DML_DEPTH_TO_SPACE_OPERATOR_DESC)
DML_DEPTH_TO_SPACE1_OPERATOR_DESC 깊이에서 공간 데이터 블록으로 데이터를 다시 정렬(permutes)합니다. 연산자는 깊이 차원의 값이 공간 블록에서 높이 및 너비 차원으로 이동되는 입력 텐서의 복사본을 출력합니다. (DML_DEPTH_TO_SPACE1_OPERATOR_DESC)
DML_DIAGONAL_MATRIX_OPERATOR_DESC 주 대각선에 있는 ID와 같은 행렬(또는 기타 명시적 값)을 생성하고 다른 곳에서는 0을 생성합니다.

DML_DYNAMIC_QUANTIZE_LINEAR_OPERATOR_DESC InputTensor정량화하는 데 필요한 수량화 배율과 0포인트 값을 계산한 다음, 해당 양자화를 적용하여 결과를 outputTensor.
DML_ELEMENT_WISE_ABS_OPERATOR_DESC *InputTensor*각 요소에 대한 절대값을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_ACOS_OPERATOR_DESC *InputTensor*각 요소에 대한 아크코사인을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_ACOSH_OPERATOR_DESC *InputTensor*각 요소에 대한 하이퍼볼릭 아크코사인을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_ADD_OPERATOR_DESC *ATensor* 모든 요소를 *BTensor*해당 요소에 추가하여 결과를 *OutputTensor*해당 요소에 배치합니다. (DML_ELEMENT_WISE_ADD_OPERATOR_DESC)
DML_ELEMENT_WISE_ADD1_OPERATOR_DESC *ATensor* 모든 요소를 *BTensor* 해당 요소에 추가하고 결과를 *OutputTensor*해당 요소에 배치하고 융합 활성화 옵션을 사용합니다.
DML_ELEMENT_WISE_ASIN_OPERATOR_DESC *InputTensor*각 요소에 대한 아크사인을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_ASINH_OPERATOR_DESC *InputTensor*각 요소에 대한 하이퍼볼릭 아크사인을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_ATAN_OPERATOR_DESC *InputTensor*각 요소에 대한 아크탄젠트를 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_ATAN_YX_OPERATOR_DESC *ATensor* 및 *BTensor*각 요소에 대해 2 인수 아크탄젠트를 계산합니다. 여기서 *ATensorY*축 및 *BTensorX*축결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_ATANH_OPERATOR_DESC

*InputTensor*각 요소에 대한 하이퍼볼릭 아크탄젠트를 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_BIT_AND_OPERATOR_DESC 입력 텐서의 각 해당 요소 간에 비트 AND를 계산하고 결과를 출력 텐서에 씁니다.
DML_ELEMENT_WISE_BIT_COUNT_OPERATOR_DESC 입력 텐서의 각 요소에 대한 비트 모집단 수(1로 설정된 비트 수)를 계산하고 결과를 출력 텐서에 씁니다.
DML_ELEMENT_WISE_BIT_NOT_OPERATOR_DESC 입력 텐서의 각 요소에 대해 비트 NOT을 계산하고 결과를 출력 텐서에 씁니다.
DML_ELEMENT_WISE_BIT_OR_OPERATOR_DESC 입력 텐서의 각 해당 요소 간에 비트 OR를 계산하고 결과를 출력 텐서에 씁니다.
DML_ELEMENT_WISE_BIT_SHIFT_LEFT_OPERATOR_DESC *BTensor*해당 요소에서 제공하는 비트 수만큼 *ATensor* 각 요소의 논리적 왼쪽 시프트를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_BIT_SHIFT_RIGHT_OPERATOR_DESC *BTensor*해당 요소에서 제공하는 비트 수만큼 *ATensor* 각 요소의 논리적 오른쪽 이동을 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_BIT_XOR_OPERATOR_DESC 입력 텐서의 각 해당 요소 간에 비트 XOR(eXclusive OR)를 계산하고 결과를 출력 텐서에 씁니다.
DML_ELEMENT_WISE_CEIL_OPERATOR_DESC *InputTensor*각 요소의 최대값을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다. x의 최대값은 x보다 크거나 같은 가장 작은 정수입니다.
DML_ELEMENT_WISE_CLIP_GRAD_OPERATOR_DESC 요소별 클립백프로포지션 그라데이션을 계산합니다.
DML_ELEMENT_WISE_CLIP_GRAD1_OPERATOR_DESC 요소별 클립백프로포지션 그라데이션을 계산합니다.
DML_ELEMENT_WISE_CLIP_OPERATOR_DESC

*InputTensor*각 요소에 대해 다음 작업을 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다. 이 연산자는 닫힌 간격 [최소, 최대] 내에서 입력의 모든 요소를 클램프(또는 제한)합니다.
DML_ELEMENT_WISE_CLIP1_OPERATOR_DESC *InputTensor*각 요소에 대해 클램핑(또는 제한) 작업을 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_CONSTANT_POW_OPERATOR_DESC *InputTensor* 각 요소를 *지수OutputTensor*해당 요소에 넣습니다.
DML_ELEMENT_WISE_COS_OPERATOR_DESC *InputTensor*각 요소의 삼각 코사인을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_COSH_OPERATOR_DESC *InputTensor*각 요소의 하이퍼볼릭 코사인을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_DEQUANTIZE_LINEAR_OPERATOR_DESC *ScaleTensor* 및 `ZeroPointTensor`해당 요소와 관련하여 *InputTensor* 모든 요소에 대해 다음 선형 정수화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_DIFFERENCE_SQUARE_OPERATOR_DESC *ATensor*해당 요소에서 *BTensor* 각 요소를 빼고 결과 자체를 곱하고 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_DIVIDE_OPERATOR_DESC *BTensor*해당 요소에 대해 *ATensor* 각 요소의 몫을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_ERF_OPERATOR_DESC *InputTensor*각 요소에 대해 erf(Gaussian 오류 함수)를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_EXP_OPERATOR_DESC *InputTensor*각 요소에 자연 지수 함수를 적용하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_FLOOR_OPERATOR_DESC *InputTensor*각 요소의 바닥을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_IDENTITY_OPERATOR_DESC

*InputTensor*각 요소에 대한 ID를 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_IF_OPERATOR_DESC *ConditionTensor*해당 요소의 값에 따라 *ATensor* 또는 *BTensor*요소를 선택합니다. *ConditionTensor의 0이 아닌 요소는 ATensor선택할* 있지만 값이 0인 요소는 *BTensor*선택합니다.
DML_ELEMENT_WISE_IS_INFINITY_OPERATOR_DESC 지정된 *InfinityMode*따라 IEEE-754 -inf, inf 또는 둘 다에 대해 *InputTensor* 각 요소를 확인하고 결과(true의 경우 1, false의 경우 0)를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_IS_NAN_OPERATOR_DESC 입력 텐서의 각 요소에 대해 입력이 NaN(IEEE-754에 정의된 대로)이면 1을 반환하고, 그렇지 않으면 0을 반환합니다. 결과는 출력 텐서의 해당 요소에 배치됩니다.
DML_ELEMENT_WISE_LOG_OPERATOR_DESC *InputTensor*각 요소의 기본 e(자연) 로그를 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_LOGICAL_AND_OPERATOR_DESC 입력 텐서의 해당 요소 쌍마다 논리 AND를 수행하여 결과(true의 경우 1, false의 경우 0)를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_LOGICAL_EQUALS_OPERATOR_DESC 입력 텐서의 해당 요소 쌍마다 논리 수행하여 결과(true의 경우 1, false의 경우 0)를 OutputTensor해당 요소에 배치합니다.
DML_ELEMENT_WISE_LOGICAL_GREATER_THAN_OPERATOR_DESC 입력 텐서의 해당 요소 쌍마다보다 큰 논리 수행하여 결과(true의 경우 1, false의 경우 0)를 OutputTensor해당 요소에 배치합니다.
DML_ELEMENT_WISE_LOGICAL_GREATER_THAN_OR_EQUAL_OPERATOR_DESC 입력 텐서의 해당 요소 쌍에서 보다 크거나 같은 논리 수행하여 결과(true의 경우 1, false의 경우 0)를 OutputTensor해당 요소에 배치합니다.
DML_ELEMENT_WISE_LOGICAL_LESS_THAN_OPERATOR_DESC 입력 텐서의 해당 요소 쌍마다 미만의 논리 수행하여 결과(true의 경우 1, false의 경우 0)를 OutputTensor해당 요소에 배치합니다.
DML_ELEMENT_WISE_LOGICAL_LESS_THAN_OR_EQUAL_OPERATOR_DESC

입력 텐서의 해당 요소 쌍마다 보다 작거나 같은 논리 수행하여 결과(true의 경우 1, false의 경우 0)를 OutputTensor해당 요소에 배치합니다.
DML_ELEMENT_WISE_LOGICAL_NOT_OPERATOR_DESC *InputTensor*각 요소에 대해 논리 NOT을 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_LOGICAL_OR_OPERATOR_DESC 입력 텐서의 해당 요소 쌍마다 논리 OR을 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_LOGICAL_XOR_OPERATOR_DESC 입력 텐서의 해당 요소 쌍마다 논리 XOR(배타적 또는)를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_MAX_OPERATOR_DESC 입력 텐서에서 두 개의 해당 요소 중 더 큰 요소를 사용하여 결과를 출력 텐서의 해당 요소에 배치합니다.
DML_ELEMENT_WISE_MEAN_OPERATOR_DESC 입력 텐서의 해당 요소 쌍의 각 쌍을 평균하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_MIN_OPERATOR_DESC 입력 텐서에서 해당하는 두 요소 중 더 작은 요소를 가져와 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_MODULUS_FLOOR_OPERATOR_DESC 입력 텐서에서 해당하는 각 요소 쌍에 대해 Python 모듈러스와 동일한 결과를 사용하여 모듈러스를 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_MODULUS_TRUNCATE_OPERATOR_DESC 입력 텐서의 해당 요소 쌍마다 C 모듈러스 연산자를 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_MULTIPLY_OPERATOR_DESC 입력 텐서의 해당 요소 쌍의 각 쌍의 곱을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_NEGATE_OPERATOR_DESC *InputTensor*각 요소를 부정하고 결과를 *OutputTensor*해당 요소에 저장합니다.
DML_ELEMENT_WISE_POW_OPERATOR_DESC

ExponentTensor해당 요소의 성능으로 InputTensor의 각 요소를 계산하여 결과를 OutputTensor해당 요소에 배치합니다.
DML_ELEMENT_WISE_QUANTIZE_LINEAR_OPERATOR_DESC *ScaleTensor* 및 `ZeroPointTensor`해당 요소와 관련하여 *InputTensor* 모든 요소에 대해 다음 선형 양자화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_QUANTIZED_LINEAR_ADD_OPERATOR_DESC *ATensor* 모든 요소를 *BTensor*해당 요소에 추가하여 결과를 *OutputTensor*해당 요소에 배치합니다. (DML_ELEMENT_WISE_QUANTIZED_LINEAR_ADD_OPERATOR_DESC)
DML_ELEMENT_WISE_RECIP_OPERATOR_DESC 입력 텐서의 각 요소에 대한 상호 계산을 수행하여 결과를 출력 텐서의 해당 요소에 배치합니다.
DML_ELEMENT_WISE_ROUND_OPERATOR_DESC *InputTensor* 각 요소를 정수 값으로 반올림하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_SIGN_OPERATOR_DESC *InputTensor*각 요소의 기호를 나타내는 값을 반환하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_SIN_OPERATOR_DESC *InputTensor*각 요소의 삼각 사인을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_SINH_OPERATOR_DESC *InputTensor*각 요소의 하이퍼볼릭 사인을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_SQRT_OPERATOR_DESC *InputTensor*각 요소의 제곱근을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_SUBTRACT_OPERATOR_DESC *ATensor*해당 요소에서 *BTensor* 각 요소를 뺍니다. 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_TAN_OPERATOR_DESC *InputTensor*각 요소의 삼각 탄젠트를 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_TANH_OPERATOR_DESC *InputTensor*요소의 하이퍼볼릭 탄젠트를 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.

DML_ELEMENT_WISE_THRESHOLD_OPERATOR_DESC 지정된 임계값 *Min*아래의 *InputTensor* 모든 요소를 *Min*바꿉니다. 결과는 *OutputTensor*해당 요소에 배치됩니다.
DML_FEATURE_DATA_FEATURE_LEVELS DirectML 디바이스에서 지원하는 기능 수준에 대한 세부 정보를 제공합니다.
DML_FEATURE_DATA_TENSOR_DATA_TYPE_SUPPORT DirectML 디바이스가 텐서 내의 특정 데이터 형식을 지원하는지 여부에 대한 세부 정보를 제공합니다.
DML_FEATURE_QUERY_FEATURE_LEVELS 하나 이상의 기능 수준에 대한 지원을 위해 DirectML 디바이스를 쿼리하는 데 사용됩니다.
DML_FEATURE_QUERY_TENSOR_DATA_TYPE_SUPPORT 텐서 내의 특정 데이터 형식에 대한 지원을 위해 DirectML 디바이스를 쿼리하는 데 사용됩니다.
DML_FILL_VALUE_CONSTANT_OPERATOR_DESC 지정된 상수 *값*텐서를 채웁니다.
DML_FILL_VALUE_SEQUENCE_OPERATOR_DESC 텐서를 시퀀스로 채웁니다.
DML_GATHER_ELEMENTS_OPERATOR_DESC 인덱스 텐서를 사용하여 지정된 축을 따라 입력 텐서에서 요소를 수집하여 입력에 다시 매핑합니다.
DML_GATHER_ND_OPERATOR_DESC 인덱스 텐서를 사용하여 입력 텐서에서 요소를 수집하여 인덱스를 입력의 전체 하위 블록에 다시 매핑합니다. (DML_GATHER_ND_OPERATOR_DESC)
DML_GATHER_ND1_OPERATOR_DESC 인덱스 텐서를 사용하여 입력 텐서에서 요소를 수집하여 인덱스를 입력의 전체 하위 블록에 다시 매핑합니다. (DML_GATHER_ND1_OPERATOR_DESC)
DML_GATHER_OPERATOR_DESC **IndicesTensor** 사용하여 **Axis**따라 입력 텐서에서 요소를 수집하여 인덱스를 다시 매핑합니다.

DML_GEMM_OPERATOR_DESC `Output = FusedActivation(Alpha * TransA(A) x TransB(B) + Beta * C)`양식의 일반 행렬 곱하기 함수를 수행합니다. 여기서 `x` 행렬 곱셈을 표시하고 `*` 스칼라를 곱한 값을 나타냄
DML_GRAPH_DESC 최적화된 결합 연산자를 컴파일하는 데 사용되는 DirectML 연산자의 그래프를 설명합니다.
DML_GRAPH_EDGE_DESC DML_GRAPH_DESC 정의하고 IDMLDevice1::CompileGraph전달된 DirectML 연산자의 그래프 내 연결에 대한 일반 컨테이너입니다.
DML_GRAPH_NODE_DESC DML_GRAPH_DESC 정의하고 IDMLDevice1::CompileGraph전달된 DirectML 연산자의 그래프 내 노드에 대한 제네릭 컨테이너입니다.
DML_GRU_OPERATOR_DESC 입력에서 (표준 계층) 1층 제어 GRU(되풀이 단위) 함수를 수행합니다. 이 연산자는 여러 게이트를 사용하여 이 계층을 수행합니다. 이러한 게이트는 시퀀스 길이 차원과 *SequenceLengthsTensor*따라 루프에서 여러 번 수행됩니다.
DML_INPUT_GRAPH_EDGE_DESC DML_GRAPH_DESC 정의하고 IDMLDevice1::CompileGraph전달된 DirectML 연산자의 그래프 내에서 연결을 설명합니다. 이 구조체는 그래프 입력에서 내부 노드의 입력으로의 연결을 정의하는 데 사용됩니다.
DML_INTERMEDIATE_GRAPH_EDGE_DESC DML_GRAPH_DESC 정의하고 IDMLDevice1::CompileGraph전달된 DirectML 연산자의 그래프 내에서 연결을 설명합니다. 이 구조체는 내부 노드 간의 연결을 정의하는 데 사용됩니다.
DML_JOIN_OPERATOR_DESC 지정된 축을 따라 입력 텐서 배열을 연결합니다.
DML_LOCAL_RESPONSE_NORMALIZATION_GRAD_OPERATOR_DESC 로컬 응답 정규화대한 백프로포지션 그라데이션을 계산합니다.
DML_LOCAL_RESPONSE_NORMALIZATION_OPERATOR_DESC 입력에서 LRN(로컬 응답 정규화) 함수를 수행합니다.

DML_LP_NORMALIZATION_OPERATOR_DESC 입력 텐서의 지정된 축을 따라 Lp 정규화 함수를 수행합니다.
DML_LP_POOLING_OPERATOR_DESC 입력 텐서를 통해 슬라이딩 윈도우 내의 요소에서 Lp 정규화된 값을 계산합니다.
DML_LSTM_OPERATOR_DESC 입력에서 LSTM(1층 장기 메모리) 함수를 수행합니다. 이 연산자는 여러 게이트를 사용하여 이 계층을 수행합니다. 이러한 게이트는 시퀀스 길이 차원과 *SequenceLengthsTensor*따라 루프에서 여러 번 수행됩니다.
DML_MATRIX_MULTIPLY_INTEGER_OPERATOR_DESC 정수 데이터에 대해 행렬 곱하기 함수를 수행합니다.
DML_MAX_POOLING_GRAD_OPERATOR_DESC 최대 풀링에 대한 백프로포지션 그라데이션을 계산합니다 (DML_MAX_POOLING2_OPERATOR_DESC참조).
DML_MAX_POOLING_OPERATOR_DESC 입력 텐서를 통해 슬라이딩 윈도우 내의 요소에서 최대값을 계산합니다.
DML_MAX_POOLING1_OPERATOR_DESC 입력 텐서를 통해 슬라이딩 윈도우 내 요소의 최대값을 계산하고 선택적으로 선택한 최대값의 인덱스를 반환합니다. (DML_MAX_POOLING1_OPERATOR_DESC)
DML_MAX_POOLING2_OPERATOR_DESC 입력 텐서를 통해 슬라이딩 윈도우 내 요소의 최대값을 계산하고 선택적으로 선택한 최대값의 인덱스를 반환합니다. (DML_MAX_POOLING2_OPERATOR_DESC)
DML_MAX_UNPOOLING_OPERATOR_DESC *IndicesTensor*제공된 인덱스 값에 따라 출력 텐서 *OutputTensor* 입력 텐서 *InputTensor*값을 입력 풀링 작업에서 얻은 값으로 채워서 최대 풀링 작업(자세한 내용은 DML_MAX_POOLING_OPERATOR1_DESC 참조)을 반전합니다. 이 프로세스의 영향을 받지 않은 출력 텐서의 요소는 0 값으로 남습니다.
DML_MEAN_VARIANCE_NORMALIZATION_OPERATOR_DESC 입력 텐서에서 평균 분산 정규화 함수를 수행합니다. 이 연산자는 정규화를 수행하기 위해 입력 텐서의 평균 및 분산을 계산합니다. (DML_MEAN_VARIANCE_NORMALIZATION_OPERATOR_DESC)

DML_MEAN_VARIANCE_NORMALIZATION1_OPERATOR_DESC 입력 텐서에서 평균 분산 정규화 함수를 수행합니다. 이 연산자는 정규화를 수행하기 위해 입력 텐서의 평균 및 분산을 계산합니다. (DML_MEAN_VARIANCE_NORMALIZATION1_OPERATOR_DESC)
DML_NONZERO_COORDINATES_OPERATOR_DESC 입력 텐서의 0이 아닌 모든 요소의 N차원 좌표를 계산합니다.
DML_ONE_HOT_OPERATOR_DESC *원 핫 인코딩된* 값으로 채워진 텐서를 생성합니다. 이 연산자는 선택한 축의 모든 시퀀스에 대해 해당 시퀀스의 한 요소를 제외한 모든 요소가 OffValue설정되고 나머지 단일 요소가 OnValue설정되는 출력 텐서를 생성합니다.
DML_OPERATOR_DESC 연산자 설명에 대한 제네릭 컨테이너입니다. 이 구조체에 지정된 매개 변수를 사용하여 DirectML 연산자를 생성합니다. 자세한 내용은 IDMLDevice::CreateOperator를 참조하세요.
DML_OPERATOR_GRAPH_NODE_DESC DML_GRAPH_DESC 정의하고 IDMLDevice1::CompileGraph전달된 DirectML 연산자의 그래프 내에서 노드를 디크리브합니다.
DML_OUTPUT_GRAPH_EDGE_DESC DML_GRAPH_DESC 정의하고 IDMLDevice1::CompileGraph전달된 DirectML 연산자의 그래프 내에서 연결을 설명합니다. 이 구조체는 내부 노드의 출력에서 그래프 출력으로의 연결을 정의하는 데 사용됩니다.
DML_PADDING_OPERATOR_DESC DML_PADDING_OPERATOR_DESC 구조체(directml.h)는 입력 텐서를 가장자리의 상수 또는 미러 값으로 팽창시키고 결과를 출력에 씁니다.
DML_PADDING1_OPERATOR_DESC DML_PADDING1_OPERATOR_DESC 구조체(directml.h)는 입력 텐서를 가장자리에 상수 또는 미러 값으로 팽창시키고 결과를 출력에 씁니다.
DML_QUANTIZED_LINEAR_CONVOLUTION_OPERATOR_DESC *InputTensor*사용하여 *FilterTensor* 구성을 수행합니다. 이 연산자는 정량화된 데이터에 대해 정방향 연결을 수행합니다. 이 연산자는 입력을 정량화하고 출력을 계산한 다음 정량화하는 것과 수학적으로 동일합니다.
DML_QUANTIZED_LINEAR_MATRIX_MULTIPLY_OPERATOR_DESC

정량화된 데이터에 대해 행렬 곱하기 함수를 수행합니다. 이 연산자는 입력을 기준으로 한 다음 행렬을 곱한 다음 출력을 정량화하는 것과 수학적으로 동일합니다.
DML_RANDOM_GENERATOR_OPERATOR_DESC 출력 텐서를 결정적으로 생성된 의사 임의 균일하게 분산된 비트로 채웁니다. 이 연산자는 필요에 따라 업데이트된 내부 생성기 상태를 출력할 수도 있습니다. 이 상태는 연산자의 후속 실행 중에 사용할 수 있습니다.
DML_REDUCE_OPERATOR_DESC 입력 텐서의 하나 이상의 차원 내에서 요소의 감소(합계, 제품, 최소값 등)를 출력합니다.
DML_RESAMPLE_GRAD_OPERATOR_DESC Resample에 대한 백프로포지션 그라데이션을 계산합니다(DML_RESAMPLE1_OPERATOR_DESC참조).
DML_RESAMPLE_OPERATOR_DESC 배율 인수를 사용하여 원본에서 대상 텐서로 요소를 다시 샘플링하여 대상 텐서 크기를 계산합니다. 선형 또는 가장 가까운 인접 보간 모드를 사용할 수 있습니다. (DML_RESAMPLE_OPERATOR_DESC)
DML_RESAMPLE1_OPERATOR_DESC 배율 인수를 사용하여 원본에서 대상 텐서로 요소를 다시 샘플링하여 대상 텐서 크기를 계산합니다. 선형 또는 가장 가까운 인접 보간 모드를 사용할 수 있습니다. (DML_RESAMPLE1_OPERATOR_DESC)
DML_REVERSE_SUBSEQUENCES_OPERATOR_DESC 텐서의 하나 이상의 하위 시퀀스의 요소를 반대로 바뀝니다. 역방향으로 사용할 하위 시퀀스 집합은 제공된 축 및 시퀀스 길이에 따라 선택됩니다.
DML_RNN_OPERATOR_DESC 입력에서 1계층 단순 RNN(되풀이 신경망) 함수를 수행합니다. 이 함수를 입력 게이트라고도 합니다. 이 연산자는 시퀀스 길이 차원과 *SequenceLengthsTensor*따라 루프에서 이 함수를 여러 번 수행합니다.
DML_ROI_ALIGN_GRAD_OPERATOR_DESC ROI_ALIGN 및 ROI_ALIGN1대한 백프로포지션 그라데이션을 계산합니다.
DML_ROI_ALIGN_OPERATOR_DESC 마스크 R-CNN 용지에 설명된 대로 ROI 맞춤 작업을 수행합니다. 요약하자면, 이 작업은 입력 이

미지 텐서에서 작물을 추출하고 지정된 *InterpolationMode*사용하여 *OutputTensor* 마지막 2차원으로 지정된 일반 출력 크기로 크기를 조정합니다.
DML_ROI_ALIGN1_OPERATOR_DESC 마스크 R-CNN 용지에 설명된 대로 ROI 맞춤 작업을 수행합니다. 요약하자면, 이 작업은 입력 이미지 텐서에서 잘린 창을 추출하고 지정된 *InterpolationMode*사용하여 *OutputTensor* 마지막 2차원으로 지정된 공통 출력 크기로 크기를 조정합니다.
DML_ROI_POOLING_OPERATOR_DESC 입력 텐서에서 MaxPool 함수를 수행합니다(관심 영역 또는 UI에 따라).
DML_SCALAR_UNION 스칼라 형식의 통합입니다.
DML_SCALE_BIAS DirectML 연산자에 제공된 배율 및 바이어스 용어의 값을 포함합니다. 배율 및 바이어스는 g(x) = x * 배율 + 바이어스 함수를 적용하는 효과가 있습니다.
DML_SCATTER_ND_OPERATOR_DESC 전체 입력 텐서를 출력에 복사한 다음, 선택한 인덱스를 업데이트 텐서의 해당 값으로 덮어씁니다. (DML_SCATTER_ND_OPERATOR_DESC)
DML_SCATTER_OPERATOR_DESC 전체 입력 텐서를 출력에 복사한 다음, 선택한 인덱스를 업데이트 텐서의 해당 값으로 덮어씁니다. (DML_SCATTER_OPERATOR_DESC)
DML_SIZE_2D 텐서 내 요소의 2차원 평면이나 2차원 배율 또는 2차원 너비/높이 값의 크기(DirectML 연산자에 제공됨)를 나타낼 수 있는 값을 포함합니다.
DML_SLICE_GRAD_OPERATOR_DESC Slice에 대한 백프로포지션 그라데이션을 계산합니다(DML_SLICE1_OPERATOR_DESC참조).
DML_SLICE_OPERATOR_DESC 입력 텐서의 단일 하위 영역("조각")을 추출합니다. (DML_SLICE_OPERATOR_DESC)
DML_SLICE1_OPERATOR_DESC 입력 텐서의 단일 하위 영역("조각")을 추출합니다. (DML_SLICE1_OPERATOR_DESC)

DML_SPACE_TO_DEPTH_OPERATOR_DESC 공간 데이터 블록을 깊이로 다시 정렬합니다. 연산자는 높이 및 너비 차원의 값이 깊이 차원으로 이동되는 입력 텐서의 복사본을 출력합니다. (DML_SPACE_TO_DEPTH_OPERATOR_DESC)
DML_SPACE_TO_DEPTH1_OPERATOR_DESC 공간 데이터 블록을 깊이로 다시 정렬합니다. 연산자는 높이 및 너비 차원의 값이 깊이 차원으로 이동되는 입력 텐서의 복사본을 출력합니다. (DML_SPACE_TO_DEPTH1_OPERATOR_DESC)
DML_SPLIT_OPERATOR_DESC 축을 따라 입력 텐서를 여러 출력 텐서로 분할합니다.
DML_TENSOR_DESC DirectML 텐서 설명에 대한 제네릭 컨테이너입니다.
DML_TILE_OPERATOR_DESC 입력 텐서를 바둑판식으로 배열하여 출력 텐서를 생성합니다. 입력 텐서의 각 차원에 있는 요소는 *반복* 배열의 배수로 반복됩니다.
DML_TOP_K_OPERATOR_DESC *InputTensor*축을 따라 각 시퀀스에서 가장 큰 *K* 요소를 선택하고 *OutputValueTensor* 및 *OutputIndexTensor*해당 요소의 값과 인덱스를 각각 반환합니다.
DML_TOP_K1_OPERATOR_DESC *InputTensor*축을 따라 각 시퀀스에서 가장 크거나 작은 *K* 요소를 선택하고 *OutputValueTensor* 및 *OutputIndexTensor*해당 요소의 값과 인덱스를 각각 반환합니다.
DML_UPSAMPLE_2D_OPERATOR_DESC 입력 이미지를 업샘플링하여 결과를 출력 텐서에 기록합니다. 차원의 순서는 NCHW(BatchSize, ChannelCount, Height, Width) 또는 NCDHW(BatchSize, ChannelCount, Depth, Height, Width)여야 하지만 데이터가 다른 형식으로 저장되는 경우 보폭을 사용할 수 있습니다.
DML_VALUE_SCALE_2D_OPERATOR_DESC 요소별 배율 및 바이어스 함수 `Output = Scale * Input + Bias`수행합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

DirectWrite

아티클2023. 03. 16.

DirectWrite 기술에 대한 개요입니다.

DirectWrite 개발하려면 다음 헤더가 필요합니다.

- dwrite.h
- dwrite_1.h
- dwrite_2.h
- dwrite_3.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- DirectWrite

열거형

DWRITE_AUTOMATIC_FONT_AXES 글꼴을 선택하는 동안 레이아웃에 자동으로 적용할 수 있는 특정 축을 지정하는 상수를 정의합니다.
DWRITE_BASELINE DWRITE_BASELINE 열거형에는 텍스트 맞춤 기준을 지정하는 값이 포함됩니다.
DWRITE_BREAK_CONDITION 줄 바꿈 동작을 결정하는 데 사용되는 인라인 개체 또는 텍스트 가장자리의 조건을 나타냅니다.
DWRITE_CONTAINER_TYPE 글꼴 리소스의 컨테이너 형식을 지정합니다. 컨테이너는 기본 글꼴 파일이 패키지되는 컨테이너를 설명하므로 컨테이너 형식은 글꼴 파일 형식(DWRITE_FONT_FILE_TYPE)과 다릅니다.
DWRITE_FACTORY_TYPE DirectWrite 팩터리 개체의 형식을 지정합니다.
DWRITE_FLOW_DIRECTION 텍스트 줄이 서로 상대적으로 배치되는 방식의 방향을 나타냅니다.

DWRITE_FONT_AXIS_ATTRIBUTES 글꼴 축의 특성을 지정하는 상수를 정의합니다.
DWRITE_FONT_AXIS_TAG 글꼴 축에 4자 식별자를 지정하는 상수를 정의합니다.
DWRITE_FONT_FACE_TYPE 전체 글꼴 면의 파일 형식을 나타냅니다.
DWRITE_FONT_FAMILY_MODEL 글꼴 패밀리를 함께 그룹화하는 방법을 지정하는 상수를 정의합니다.
DWRITE_FONT_FEATURE_TAG 글꼴에서 제공하는 텍스트의 입력 기능을 나타내는 값입니다.
DWRITE_FONT_FILE_TYPE 단일 글꼴 파일로 표시되는 글꼴의 형식입니다. 여러 파일(예: 형식 1)로 구성된 글꼴 형식입니다. PFM 및 . PFB에는 각 파일 형식에 대해 별도의 열거형 값이 있습니다.
DWRITE_FONT_LINE_GAP_USAGE DWRITE_FONT_METRICS::lineGap 값이 줄 메트릭의 일부여야 하는지 여부를 지정합니다.
DWRITE_FONT_PROPERTY_ID 글꼴에서 문자열을 식별합니다.
DWRITE_FONT_SIMULATIONS 글꼴 면에 적용할 알고리즘 스타일 시뮬레이션을 지정합니다. 굵게 및 비스듬한 시뮬레이션은 비트 OR 연산을 통해 결합할 수 있습니다.
DWRITE_FONT_SOURCE_TYPE 글꼴 집합에 글꼴이 포함되는 메커니즘을 지정하는 상수를 정의합니다.
DWRITE_FONT_STRETCH 글꼴의 일반 가로 세로 비율에 비해 글꼴이 확장된 정도를 나타냅니다.
DWRITE_FONT_STYLE 글꼴 얼굴의 스타일을 보통, 기울임꼴 또는 경사로 나타냅니다.

DWRITE_FONT_WEIGHT 스트로크의 가벼움 또는 무거움 측면에서 서체의 밀도를 나타냅니다.
DWRITE_GLYPH_IMAGE_FORMATS 글꼴 전체 수준 또는 문자 모양당 글꼴에서 지원되는 형식을 지정합니다.
DWRITE_GLYPH_ORIENTATION_ANGLE DWRITE_GLYPH_ORIENTATION_ANGLE 열거형에는 문자 모양이 x축을 지향하는 방법을 지정하는 값이 포함됩니다.
DWRITE_GRID_FIT_MODE 문자 모양 윤곽선의 그리드 맞춤을 사용할지 여부를 지정합니다(힌트라고도 함).
DWRITE_INFORMATIONAL_STRING_ID 글꼴 파일에 포함된 문자열을 식별하는 정보 문자열 열거형입니다.
DWRITE_LINE_SPACING_METHOD 텍스트 레이아웃의 줄 간격에 사용되는 메서드입니다.
DWRITE_LOCALITY 리소스의 위치를 지정합니다.
DWRITE_MEASURING_MODE 텍스트 레이아웃에 사용되는 측정 방법을 나타냅니다.
DWRITE_NUMBER_SUBSTITUTION_METHOD 숫자 및 관련 문장 부호에 숫자 대체를 적용하는 방법을 지정합니다.
DWRITE_OPTICAL_ALIGNMENT 광학 여백 맞춤 모드입니다.
DWRITE_OUTLINE_THRESHOLD DWRITE_OUTLINE_THRESHOLD 열거형에는 개요 모드에서 문자 모양을 렌더링할지 여부를 결정하기 위해 IDWriteFontFace1::GetRecommendedRenderingMode 메서드에서 사용하는 정책을 지정하는 값이 포함되어 있습니다.

DWRITE_PANOSE_ARM_STYLE DWRITE_PANOSE_ARM_STYLE 열거형에는 텍스트에 대한 줄기 및 둥근 문자 형식의 종료 스타일을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_ASPECT DWRITE_PANOSE_ASPECT 열거형에는 문자 면의 너비와 높이 사이의 비율을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_ASPECT_RATIO DWRITE_PANOSE_ASPECT_RATIO 열거형에는 문자 면의 너비와 높이 사이의 비율에 대한 정보를 지정하는 값이 포함됩니다.
DWRITE_PANOSE_CHARACTER_RANGES DWRITE_PANOSE_CHARACTER_RANGES 열거형에는 글꼴에서 사용할 수 있는 문자 유형을 지정하는 값이 포함되어 있습니다.
DWRITE_PANOSE_CONTRAST DWRITE_PANOSE_CONTRAST 열거형에는 대문자 'O'와 같은 문자에 대해 스트로크의 가장 두꺼운 점과 가장 얇은 점 사이의 비율을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_DECORATIVE_CLASS DWRITE_PANOSE_DECORATIVE_CLASS 열거형에는 문자 얼굴의 일반적인 모양을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_DECORATIVE_TOPOLOGY DWRITE_PANOSE_DECORATIVE_TOPOLOGY 열거형에는 글꼴의 전체 도형 특성을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_FAMILY DWRITE_PANOSE_FAMILY 열거형에는 서체 분류의 종류를 지정하는 값이 포함됩니다.
DWRITE_PANOSE_FILL DWRITE_PANOSE_FILL 열거형에는 채우기 및 선 처리 유형을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_FINIALS DWRITE_PANOSE_FINIALS 열거형에는 문자 끝과 소수 오름차순 처리 방법을 지정하는 값이 포함됩니다.

DWRITE_PANOSE_LETTERFORM DWRITE_PANOSE_LETTERFORM 열거형에는 텍스트에 대한 문자 폼의 원형을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_LINING DWRITE_PANOSE_LINING 열거형에는 장식 서체에 대한 윤곽선 처리를 지정하는 값이 포함되어 있습니다.
DWRITE_PANOSE_MIDLINE DWRITE_PANOSE_MIDLINE 열거형에는 대문자 간에 중간선 배치 및 대각선 줄기 정점 처리에 대한 정보를 지정하는 값이 포함되어 있습니다.
DWRITE_PANOSE_PROPORTION DWRITE_PANOSE_PROPORTION 열거형에는 표준 문자에 대한 추가 세부 정보를 고려하여 문자 모양 비율을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_SCRIPT_FORM DWRITE_PANOSE_SCRIPT_FORM 열거형에는 기울기와 꼬리를 고려하여 문자 얼굴의 일반적인 모양을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_SCRIPT_TOPOLOGY DWRITE_PANOSE_SCRIPT_TOPOLOGY 열거형에는 문자 폼의 토폴로지를 지정하는 값이 포함되어 있습니다.
DWRITE_PANOSE_SERIF_STYLE DWRITE_PANOSE_SERIF_STYLE 열거형에는 serif 텍스트의 모양을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_SPACING DWRITE_PANOSE_SPACING 열거형에는 문자 간격(모노스페이스 및 비례)을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_STROKE_VARIATION DWRITE_PANOSE_STROKE_VARIATION 열거형에는 텍스트 문자의 얇고 두꺼운 줄기 사이의 관계를 지정하는 값이 포함되어 있습니다.
DWRITE_PANOSE_SYMBOL_ASPECT_RATIO DWRITE_PANOSE_SYMBOL_ASPECT_RATIO 열거형에는 기호 문자의 가로 세로 비율을 지정하는 값이 포함됩니다.

DWRITE_PANOSE_SYMBOL_KIND DWRITE_PANOSE_SYMBOL_KIND 열거형에는 기호 집합의 종류를 지정하는 값이 포함됩니다.
DWRITE_PANOSE_TOOL_KIND DWRITE_PANOSE_TOOL_KIND 열거형에는 문자 양식을 만드는 데 사용되는 도구 종류를 지정하는 값이 포함되어 있습니다.
DWRITE_PANOSE_WEIGHT DWRITE_PANOSE_WEIGHT 열거형에는 문자의 가중치를 지정하는 값이 포함됩니다.
DWRITE_PANOSE_XASCENT DWRITE_PANOSE_XASCENT 열거형에는 소문자의 상대 크기를 지정하는 값이 포함됩니다.
DWRITE_PANOSE_XHEIGHT DWRITE_PANOSE_XHEIGHT 열거형에는 소문자의 상대적 크기와 분음 부호(xheight)의 처리에 대한 정보를 지정하는 값이 포함되어 있습니다.
DWRITE_PARAGRAPH_ALIGNMENT 흐름 레이아웃 상자의 위쪽과 아래쪽을 기준으로 흐름 방향 축을 따라 단락 텍스트의 맞춤을 지정합니다.
DWRITE_PIXEL_GEOMETRY 텍스트 렌더링을 위해 가정되는 디바이스 픽셀(즉, 빨간색, 녹색 및 파란색 구성 요소의 물리적 배열)의 내부 구조를 나타냅니다.
DWRITE_READING_DIRECTION 읽기가 진행되는 방향을 지정합니다.
DWRITE_RENDERING_MODE 문자 모양을 렌더링하는 메서드를 나타냅니다.
DWRITE_RENDERING_MODE1 문자 모양이 렌더링되는 방법을 지정합니다.
DWRITE_SCRIPT_SHAPES 텍스트에 대한 추가 셰이핑 요구 사항을 나타냅니다.

DWRITE_TEXT_ALIGNMENT 레이아웃 상자의 선행 및 후행 가장자리를 기준으로 읽기 방향 축을 따라 단락 텍스트의 맞춤을 지정합니다.
DWRITE_TEXT_ANTIALIAS_MODE DWRITE_TEXT_ANTIALIAS_MODE 열거형에는 렌더링 모드에서 앤티앨리어싱을 호출할 때 텍스트에 사용할 앤티앨리어싱 유형을 지정하는 값이 포함되어 있습니다.
DWRITE_TEXTURE_TYPE 알파 텍스처의 형식을 식별합니다.
DWRITE_TRIMMING_GRANULARITY 레이아웃 상자가 오버플로되는 텍스트를 자르는 데 사용되는 텍스트 세분성을 지정합니다.
DWRITE_VERTICAL_GLYPH_ORIENTATION DWRITE_VERTICAL_GLYPH_ORIENTATION 열거형에는 텍스트에 대해 원하는 종류의 문자 모양 방향을 지정하는 값이 포함되어 있습니다.
DWRITE_WORD_WRAPPING 특정 여러 줄 단락에서 사용할 단어 줄 바꿈을 지정합니다.

Functions

Addfont 빌드 중인 집합에 글꼴을 추가합니다.
AddFontFaceReference 빌드 중인 집합에 글꼴에 대한 참조를 추가합니다. 필요한 메타데이터는 CreateFontSet를 호출할 때 글꼴에서 자동으로 추출됩니다. (오버로드 1/2)
AddFontFaceReference 빌드 중인 집합에 글꼴에 대한 참조를 추가합니다. 필요한 메타데이터는 CreateFontSet를 호출할 때 글꼴에서 자동으로 추출됩니다. (오버로드 2/2)

AddFontFeature OpenType 글꼴 기능을 추가합니다.
AddFontFile 지정된 글꼴 파일의 모든 글꼴에 대한 참조를 추가합니다.
AddFontFile 지정된 글꼴 파일의 모든 글꼴에 대한 참조를 추가합니다. 메서드는 글꼴 파일을 구문 분석하여 글꼴 및 해당 속성을 확인합니다.
AddFontSet 기존 글꼴 집합을 빌드 중인 글꼴에 추가하여 두 집합을 집계하거나 기본적으로 기존 글꼴 집합을 확장할 수 있도록 합니다.
Addlistener 다운로드 알림을 받는 클라이언트 정의 수신기 개체를 등록합니다. 등록된 모든 수신기의 DownloadCompleted는 BeginDownloadcompletes 후에 호출됩니다.
AddMapping 목록에 단일 매핑을 추가합니다. 각 추가 매핑에 대해 이를 한 번 호출합니다.
AddMappings 기존 글꼴 대체 개체의 모든 매핑을 추가합니다.
분석 파일을 분석하고 글꼴을 나타내는지 여부와 글꼴 시스템에서 글꼴 유형을 지원하는지 여부를 반환합니다.
AnalyzeBidi 스크립트 방향성에 대한 텍스트 범위를 분석하여 원본 및 보고 수준에서 싱크 콜백 SetBidiLevel로 특성을 읽습니다.
AnalyzeContainerType AnalyzeContainerType 메서드는 지정된 파일 데이터를 분석하여 알려진 글꼴 컨테이너 형식(예: WOFF 또는 WOFF2)인지 여부를 확인합니다.

AnalyzeLineBreakpoints 잠재적인 중단점 기회를 위해 텍스트 범위를 분석하고, 원본에서 특성을 읽고, 중단점 기회를 싱크 콜백 SetLineBreakpoints로 보고합니다.
AnalyzeNumberSubstitution 숫자 대체가 적용되는 범위에 대한 텍스트 범위를 분석하여 원본에서 특성을 읽고 대체 가능한 범위를 싱크 콜백 SetNumberSubstitution로 보고합니다.
AnalyzeScript 스크립트 경계에 대한 텍스트 범위를 분석하고 원본에서 텍스트 특성을 읽고 유니코드 스크립트 ID를 싱크 콜백 SetScript에 보고합니다.
AnalyzeVerticalGlyphOrientation 스크립트 방향에 대한 텍스트 범위를 분석하고 원본에서 텍스트 및 특성을 읽고 결과를 싱크 콜백 SetGlyphOrientation에 보고합니다.
ApplyCharacterSpacing 문자 사이의 간격을 적용하여 문자 모양 클러스터와 분음 부호를 적절하게 조정합니다.
AreCharactersLocal 지정된 문자가 로컬인지 여부를 확인합니다.
AreGlyphsLocal 지정된 문자 모양이 로컬인지 여부를 확인합니다.
BeginDownload 비동기 다운로드 작업을 시작합니다. 다운로드 작업은 CancelDownload 호출이 완료되거나 취소될 때까지 백그라운드에서 실행됩니다.
BeginDownload 글꼴 파일의 전체 또는 일부를 다운로드하기 시작합니다.
CancelDownload 큐에서 모든 다운로드 요청을 제거하고 활성 다운로드 작업을 취소합니다.
CheckTypographicFeature 문자 모양 또는 문자 모양 집합에 입력 기능을 사용할 수 있는지 확인합니다.

ComputeGlyphOrigins 문자 모양 실행 배치를 문자 모양 원본으로 변환합니다. (오버로드 2/2)
ComputeGlyphOrigins 문자 모양 실행 배치를 문자 모양 원본으로 변환합니다. (오버로드 1/2)
ConvertFontFaceToLOGFONT 지정된 글꼴의 GDI 호환 속성을 기반으로 LOGFONT 구조를 초기화합니다. (IDWriteGdiInterop.ConvertFontFaceToLOGFONT)
ConvertFontToLOGFONT 지정된 글꼴의 GDI 호환 속성을 기반으로 LOGFONT 구조를 초기화합니다. (IDWriteGdiInterop.ConvertFontToLOGFONT)
ConvertWeightStretchStyleToFontAxisValues 지정된 글꼴 두께, 늘이기, 스타일 및 크기에서 파생된 글꼴 축 값을 계산합니다.
CreateAlphaTexture 지정된 경계 사각형 내에서 문자 모양에 대해 지정된 형식의 알파 텍스처를 만듭니다.
CreateBitmapRenderTarget 문자 모양을 렌더링하는 데 사용할 수 있는 비트맵 및 메모리 DC(디바이스 컨텍스트)를 캡슐화하는 개체를 만듭니다.
CreateCustomFontCollection 사용자 지정 글꼴 컬렉션 로더를 사용하여 글꼴 컬렉션을 만듭니다.
CreateCustomFontFileReference 애플리케이션별 글꼴 파일 리소스에 대한 참조를 만듭니다.
CreateCustomRenderingParams 지정된 속성을 사용하여 렌더링 매개 변수 개체를 만듭니다. (IDWriteFactory.CreateCustomRenderingParams)
CreateCustomRenderingParams 지정된 속성을 사용하여 렌더링 매개 변수 개체를 만듭니다. (IDWriteFactory1.CreateCustomRenderingParams)

CreateCustomRenderingParams 지정된 속성을 사용하여 렌더링 매개 변수 개체를 만듭니다. (IDWriteFactory3.CreateCustomRenderingParams)
CreateCustomRenderingParams 지정된 속성을 사용하여 렌더링 매개 변수 개체를 만듭니다. (IDWriteFactory2.CreateCustomRenderingParams)
CreateEllipsisTrimmingSign 생략 기호로 줄임표를 사용하여 트리밍을 위한 인라인 개체를 만듭니다.
CreateEnumeratorFromKey 글꼴 파일 컬렉션을 캡슐화하는 글꼴 파일 열거자 개체를 만듭니다. 글꼴 시스템은 이 인터페이스를 다시 호출하여 글꼴 컬렉션을 만듭니다.
CreateFontCollectionFromFontSet 글꼴 집합에서 두께/너비/기울기 트리를 만듭니다.
CreateFontCollectionFromFontSet 글꼴 집합에서 패밀리로 그룹화된 글꼴 컬렉션을 만듭니다.
CreateFontFace 글꼴 얼굴을 나타내는 개체를 만듭니다.
CreateFontFace 글꼴에 대한 글꼴 얼굴 개체를 만듭니다. (IDWriteFont.CreateFontFace)
CreateFontFace 글꼴에 대한 글꼴 얼굴 개체를 만듭니다. (IDWriteFont3.CreateFontFace)
CreateFontFace 레이아웃, 셰이핑 또는 렌더링에 사용할 참조에서 글꼴 얼굴을 만듭니다.
CreateFontFace 참조를 사용하여 레이아웃, 셰이핑 또는 렌더링에 사용할 글꼴 얼굴을 만듭니다.

CreateFontFace 특정 축 값을 사용하여 글꼴 얼굴 인스턴스를 만듭니다.
CreateFontFace 글꼴 얼굴 참조를 거치지 않고 단일 항목에 대한 글꼴 얼굴을 만듭니다.
CreateFontFaceFromHdc 지정된 HDC의 현재 선택된 HFONT에 해당하는 IDWriteFontFace 개체를 만듭니다.
CreateFontFaceReference 전체 경로가 지정된 글꼴에 대한 참조를 만듭니다. (오버로드 1/2)
CreateFontFaceReference 전체 경로가 지정된 글꼴에 대한 참조를 만듭니다. (오버로드 2/2)
CreateFontFaceReference 파일 내의 특정 글꼴 인스턴스에 대한 참조를 만듭니다.
CreateFontFaceReference 특정 축 값을 사용하여 글꼴 얼굴 참조를 만듭니다.
CreateFontFaceWithSimulations 예를 들어 일반 변형에서 굵은 글꼴 얼굴을 명시적으로 시뮬레이션하기 위해 대체 글꼴 시뮬레이션을 사용하여 글꼴 얼굴을 만듭니다.
CreateFontFallback 추가된 매핑에서 최종 대체 개체를 만듭니다.
CreateFontFallbackBuilder 글꼴 대체 작성기 개체를 만듭니다.
CreateFontFileReference 로컬 글꼴 파일에서 글꼴 파일 참조 개체를 만듭니다.
CreateFontFileReferenceFromUrl 로더가 이 기능을 지원하는 경우 URL에서 글꼴 파일 참조를 만듭니다.

CreateFontFromLOGFONT LOGFONT 구조체에 지정된 속성과 일치하는 글꼴 개체를 만듭니다. (IDWriteGdiInterop.CreateFontFromLOGFONT)
CreateFontFromLOGFONT LOGFONT 구조체에 지정된 속성과 일치하는 글꼴 개체를 만듭니다. (IDWriteGdiInterop1.CreateFontFromLOGFONT)
CreateFontResource 글꼴 파일과 얼굴 인덱스가 지정된 글꼴 리소스를 만듭니다.
CreateFontResource 단일 항목의 글꼴 리소스를 만듭니다.
CreateFontSet AddFontFaceReference를 사용하여 지금까지 추가된 모든 글꼴 얼굴 참조에서 글꼴 집합을 만듭니다.
CreateFontSetBuilder 빈 글꼴 집합 작성기를 만들어 글꼴 얼굴 참조를 추가하고 사용자 지정 글꼴 집합을 만듭니다. (IDWriteFactory3.CreateFontSetBuilder)
CreateFontSetBuilder 빈 글꼴 집합 작성기를 만들어 글꼴 얼굴 참조를 추가하고 사용자 지정 글꼴 집합을 만듭니다. (IDWriteFactory5.CreateFontSetBuilder)
CreateFontSetBuilder 글꼴 인스턴스를 추가할 준비가 된 빈 글꼴 집합 작성기를 만들고 사용자 지정 글꼴 집합을 만듭니다.
CreateGdiCompatibleTextLayout 문자열, 형식 및 관련 제약 조건을 사용하고 결과를 나타내는 개체를 생성하며, 특정 디스플레이 해상도 및 측정 모드에 대한 형식이 지정됩니다.
CreateGlyphRunAnalysis 문자 모양 실행을 렌더링하는 데 사용되는 정보를 캡슐화하는 문자 모양 실행 분석 개체를 만듭니다. (IDWriteFactory.CreateGlyphRunAnalysis)

CreateGlyphRunAnalysis DirectWrite 문자 모양 실행을 렌더링하는 데 사용하는 정보를 캡슐화하는 문자 모양 실행 분석 개체를 만듭니다.
CreateGlyphRunAnalysis 문자 모양 실행을 렌더링하는 데 사용되는 정보를 캡슐화하는 문자 모양 실행 분석 개체를 만듭니다. (IDWriteFactory2.CreateGlyphRunAnalysis)
CreateHttpFontFileLoader HTTP 또는 HTTPS URL에서 글꼴 파일 참조를 만들 수 있는 원격 글꼴 파일 로더를 만듭니다. 호출자는 로더 등록 및 등록 취소를 담당합니다.
CreateInMemoryFontFileLoader 메모리 내 글꼴에 대한 글꼴 파일 참조를 만드는 데 사용할 수 있는 로더 개체를 만듭니다. 호출자는 로더 등록 및 등록 취소를 담당합니다.
CreateInMemoryFontFileReference 바이트 배열에서 글꼴 파일 참조(IDWriteFontFile 개체)를 만듭니다.
CreateMonitorRenderingParams 지정된 모니터에 대한 기본 설정을 사용하여 렌더링 매개 변수 개체를 만듭니다. 대부분의 경우 렌더링 매개 변수 개체를 만드는 것이 좋습니다.
CreateNumberSubstitution 로캘 이름, 대체 방법 및 사용자 재정의를 무시할지 여부를 나타내는 표시기를 사용하여 숫자 대체 개체를 만듭니다(지정된 문화권에 대해 NLS 기본값 사용).
CreateRemoteStreamFromKey 열린 파일 리소스를 캡슐화하고 원격 파일 데이터를 다운로드하는 데 사용할 수 있는 원격 글꼴 파일 스트림 개체를 만듭니다.
CreateRenderingParams 기본 모니터에 대한 기본 설정을 사용하여 렌더링 매개 변수 개체를 만듭니다. 여러 모니터에 다른 렌더링 매개 변수가 있을 수 있습니다. 자세한 내용은 여러 모니터에 대한 지원을 추가하는 방법 항목을 참조하세요.
CreateStreamFromKey 열려 있는 파일 리소스를 캡슐화하는 글꼴 파일 스트림 개체를 만듭니다.

CreateTextAnalyzer 텍스트 분석을 수행하기 위한 인터페이스를 반환합니다.
CreateTextFormat 텍스트 레이아웃에 사용되는 텍스트 형식 개체를 만듭니다. (IDWriteFactory.CreateTextFormat)
CreateTextFormat 텍스트 레이아웃에 사용되는 텍스트 형식 개체를 만듭니다. 나
CreateTextLayout 문자열, 텍스트 형식 및 관련 제약 조건을 사용하고 완전히 분석되고 형식이 지정된 결과를 나타내는 개체를 생성합니다.
CreateTypography 텍스트 레이아웃에 사용할 입력 체계 개체를 만듭니다.
DetermineMinWidth 전체 단어의 문자 간에 비상 중단 없이 레이아웃을 설정할 수 있는 최소 너비를 결정합니다.
DownloadCompleted DownloadCompleted 메서드는 다운로드 작업이 종료될 때 임의의 스레드에서 다시 호출됩니다.
그리기 애플리케이션이 구현한 렌더링 콜백(IDWriteTextRenderer::D rawInlineObject)은 개체 형식을 캐스팅하거나 쿼리할 필요 없이 인라인 개체를 그리는 데 사용할 수 있습니다. 텍스트 레이아웃은 이 메서드를 직접 호출하지 않습니다.
그리기 지정된 클라이언트 그리기 컨텍스트를 사용하여 텍스트를 그립니다.
DrawGlyphRun 지정된 위치에 있는 비트맵 대상에 문자 모양 실행을 그립니다.
DrawGlyphRun IDWriteTextLayout::D raw는 이 함수를 호출하여 클라이언트에 문자 모양 실행을 렌더링하도록 지시합니다. (IDWriteTextRenderer.DrawGlyphRun)

DrawGlyphRun IDWriteTextLayout::D raw는 이 함수를 호출하여 클라이언트에 문자 모양 실행을 렌더링하도록 지시합니다. (IDWriteTextRenderer1.DrawGlyphRun)
DrawInlineObject IDWriteTextLayout::D raw는 인라인 개체를 그려야 할 때 이 애플리케이션 콜백을 호출합니다. (IDWriteTextRenderer.DrawInlineObject)
DrawInlineObject IDWriteTextLayout::D raw는 인라인 개체를 그려야 할 때 이 애플리케이션 콜백을 호출합니다. (IDWriteTextRenderer1.DrawInlineObject)
DrawStrikethrough IDWriteTextLayout::D raw는 이 함수를 호출하여 클라이언트에 취소선을 그리도록 지시합니다. (IDWriteTextRenderer.DrawStrikethrough)
DrawStrikethrough IDWriteTextLayout::D raw는 이 함수를 호출하여 클라이언트에 취소선을 그리도록 지시합니다. (IDWriteTextRenderer1.DrawStrikethrough)
DrawUnderline IDWriteTextLayout::D raw는 이 함수를 호출하여 클라이언트에 밑줄을 그리도록 지시합니다. (IDWriteTextRenderer.DrawUnderline)
DrawUnderline IDWriteTextLayout::D raw는 이 함수를 호출하여 클라이언트에 밑줄을 그리도록 지시합니다. (IDWriteTextRenderer1.DrawUnderline)
DWRITE_MAKE_FONT_AXIS_TAG 글꼴 축에 대한 OpenType 태그를 만듭니다.
DWRITE_MAKE_OPENTYPE_TAG OpenType 태그를 32비트 정수로 만들어 태그의 첫 번째 문자가 글꼴 파일의 태그와 비교하는 데 사용할 수 있는 가장 낮은 바이트(작은 엔디안 아키텍처에서 가장 중요하지 않음)입니다.
DWriteCreateFactory 개별 DirectWrite 개체의 후속 생성에 사용되는 DirectWrite 팩터리 개체를 만듭니다.

EnqueueCharacterDownloadRequest 글꼴 다운로드 큐에 요청을 추가합니다(IDWriteFontDownloadQueue). (IDWriteFontFaceReference.EnqueueCharacterDownloadRequest)
EnqueueFileFragmentDownloadRequest 글꼴 다운로드 큐에 요청을 추가합니다(IDWriteFontDownloadQueue). (IDWriteFontFaceReference.EnqueueFileFragmentDownloadRequest)
EnqueueFontDownloadRequest 글꼴 다운로드 큐에 요청을 추가합니다(IDWriteFontDownloadQueue). (IDWriteFontFaceReference.EnqueueFontDownloadRequest)
EnqueueGlyphDownloadRequest 글꼴 다운로드 큐에 요청을 추가합니다(IDWriteFontDownloadQueue). (IDWriteFontFaceReference.EnqueueGlyphDownloadRequest)
같음 같음을 위해 글꼴 참조의 두 인스턴스를 비교합니다.
같음 **Equals**가 호출되는 글꼴 얼굴 개체와 매개 변수로 전달된 글꼴 얼굴 개체 간의 같음 비교를 수행합니다.
FindFamilyName 지정된 패밀리 이름을 사용하여 글꼴 패밀리를 찾습니다.
FindFontFace 동일한 파일, 얼굴 인덱스 및 시뮬레이션을 사용하여 글꼴 집합에서 일치하는 글꼴 얼굴 참조의 인덱스를 가져옵니다. (IDWriteFontSet.FindFontFace)
FindFontFaceReference 동일한 파일, 얼굴 인덱스 및 시뮬레이션을 사용하여 글꼴 집합에서 일치하는 글꼴 얼굴 참조의 인덱스를 가져옵니다. (IDWriteFontSet.FindFontFaceReference)
FindLocaleName 지정된 로캘 이름을 가진 로캘 이름/문자열 쌍의 인덱스(0부터 시작)를 가져옵니다.

GetAlphaBlendParams ClearType 혼합에 필요한 알파 혼합 속성을 가져옵니다.
GetAlphaTextureBounds 문자 모양 실행의 영향을 받는 실제 픽셀의 경계 사각형을 가져옵니다.
GetAutomaticFontAxes 자동 축 옵션을 검색합니다.
GetAxisNames 글꼴 축의 지역화된 이름을 검색합니다.
GetAxisValueNameCount 특정 축의 명명된 값 수를 검색합니다.
GetAxisValueNames 글꼴 축에 대한 특정 값의 지역화된 이름을 검색합니다.
GetBaseline 글꼴에서 지정된 기준선을 검색합니다.
GetBreakConditions 레이아웃은 이를 사용하여 텍스트 중 인라인 개체의 줄 바꿈 동작을 결정합니다.
GetCaretMetrics 디자인 단위의 글꼴에 대한 캐리트 메트릭을 가져옵니다.
GetCharacterSpacing 문자 사이의 간격을 가져옵니다.
GetClearTypeLevel 렌더링 매개 변수 개체의 ClearType 수준을 가져옵니다.
GetClusterMetrics 각 문자 모양 클러스터의 논리적 속성 및 측정값을 검색합니다.

GetColorPaletteCount 글꼴로 정의된 색상표 수를 가져옵니다.
GetCount 언어/문자열 쌍의 수를 가져옵니다.
GetCount 문자열 목록의 문자열 수를 가져옵니다.
GetCurrentFontFile 현재 글꼴 파일에 대한 참조를 가져옵니다.
GetCurrentRun 현재 색 문자 모양 실행을 가져옵니다.
GetCurrentRun 열거자의 현재 문자 모양 실행을 반환합니다.
GetCurrentTransform 추상 좌표를 DIP에 매핑하는 변환을 가져옵니다. 기본적으로 ID 변환입니다. 이는 기본 디바이스 컨텍스트의 월드 변환과 관련이 없습니다.
GetCurrentTransform 추상 좌표를 DIP에 매핑하는 변환을 가져옵니다.
GetDefaultFontAxisValues 글꼴 리소스에서 지원하는 모든 축의 기본값을 검색합니다.
GetDesignGlyphAdvances 문자 모양 시퀀스에 대한 디자인 단위의 고급을 검색합니다.
GetDesignGlyphMetrics 글꼴 디자인 단위에서 이상적인(해상도 독립적) 문자 모양 메트릭을 가져옵니다.
GetDrawingEffect 지정된 텍스트 위치에서 애플리케이션 정의 그리기 효과를 가져옵니다.

GetEnhancedContrast 렌더링 매개 변수 개체의 향상된 대비 속성을 가져옵니다. 유효한 값이 0보다 크거나 같습니다.
GetEudcFontCollection EUDC(최종 사용자 정의 문자) 글꼴 집합을 나타내는 글꼴 컬렉션을 가져옵니다.
GetExpirationEvent 글꼴 집합에 대한 만료 이벤트(있는 경우)를 검색합니다. 만료 이벤트는 설치, 제거 또는 업데이트되는 글꼴로 인해 만료된 경우 시스템 글꼴 집합 개체에 설정됩니다. (IDWriteFontCollection3::GetExpirationEvent)
GetExpirationEvent 글꼴 집합에 대한 만료 이벤트(있는 경우)를 검색합니다. 만료 이벤트는 설치, 제거 또는 업데이트되는 글꼴로 인해 만료된 경우 시스템 글꼴 집합 개체에 설정됩니다. (IDWriteFontSet2::GetExpirationEvent)
GetFaceNames 로캘 이름으로 인덱싱된 글꼴의 얼굴 이름(예: 일반 또는 굵게)이 포함된 지역화된 문자열 컬렉션을 가져옵니다.
GetFaceNames 로캘 이름으로 인덱싱된 글꼴의 얼굴 이름(예: 일반 또는 굵게)이 포함된 지역화된 문자열 개체를 만듭니다.
GetFamilyNames 로캘 이름으로 인덱싱된 글꼴 패밀리의 패밀리 이름을 포함하는 지역화된 문자열 개체를 만듭니다. (IDWriteFontFamily.GetFamilyNames)
GetFamilyNames 로캘 이름으로 인덱싱된 글꼴 패밀리의 패밀리 이름을 포함하는 지역화된 문자열 개체를 만듭니다. (IDWriteFontFace3.GetFamilyNames)
GetFileCount 이 로더 인스턴스를 사용하여 만든 글꼴 파일 참조 수를 반환합니다.
GetFileFragmentLocality 글꼴 파일 스트림 내의 바이트 범위(예: 글꼴 조각)의 지역성에 대한 정보를 반환합니다.

GetFilePathFromKey 글꼴 파일 참조 키에서 절대 글꼴 파일 경로를 가져옵니다.
GetFilePathLengthFromKey 글꼴 파일 참조 키에서 절대 파일 경로의 길이를 가져옵니다.
GetFiles 글꼴 얼굴을 나타내는 글꼴 파일을 가져옵니다.
GetFileSize 파일의 총 크기를 가져옵니다.
GetFileSize 글꼴 얼굴의 총 크기를 바이트 단위로 가져옵니다.
GetFileTime 마지막으로 수정한 날짜를 가져옵니다.
GetFilteredFontIndices 지정된 범위로 필터링된 모든 항목 인덱스를 검색합니다.
GetFilteredFonts 지정된 범위(엔드포인트 포함)로 필터링된 글꼴의 하위 집합을 검색합니다.
GetFilteredFonts 지정된 속성으로 필터링된 글꼴의 하위 집합을 검색합니다.
GetFilteredFonts 지정된 인덱스로 필터링된 글꼴의 하위 집합을 검색합니다.
GetFirstFontResources 집합에서 처음 발생하는 각 글꼴 리소스만 포함하는 새 글꼴 집합을 검색합니다.
GetFirstMatchingFont 지정된 속성과 가장 일치하는 글꼴을 가져옵니다.

GetFlowDirection 텍스트 줄이 흐르는 방향을 가져옵니다.
GetFont 0부터 시작하는 인덱스가 지정된 글꼴을 가져옵니다. (IDWriteFontList.GetFont)
GetFont 0부터 시작하는 인덱스가 지정된 글꼴을 가져옵니다. (IDWriteFontFamily1.GetFont)
GetFont 0부터 시작하는 인덱스가 지정된 글꼴을 가져옵니다. (IDWriteFontList1.GetFont)
GetFontAxisAttributes 글꼴 작성자가 사용자 인터페이스에서 축을 숨기도록 권장하는지 여부와 같이 지정된 축을 설명하는 특성을 검색합니다.
GetFontAxisCount 글꼴 리소스에서 지원하는 축 수를 검색합니다.
GetFontAxisRanges 각 축의 값 범위를 검색합니다.
GetFontAxisRanges 단일 항목의 축 범위를 검색합니다.
GetFontAxisValueCount 글꼴로 정의된 축 수를 검색합니다. 여기에는 정적 축과 가변 축이 모두 포함됩니다.
GetFontAxisValueCount 참조로 지정된 축 수를 검색합니다.
GetFontAxisValueCount 형식에 설정된 축 수를 검색합니다.
GetFontAxisValues 글꼴에서 사용하는 축 값 목록을 검색합니다.

GetFontAxisValues 참조로 지정된 글꼴 축 값 목록을 검색합니다.
GetFontAxisValues 형식의 글꼴 축 값 목록을 검색합니다.
GetFontCollection 글꼴 목록의 글꼴이 포함된 글꼴 컬렉션을 가져옵니다.
GetFontCollection 현재 글꼴 컬렉션을 가져옵니다.
GetFontCollection 지정된 위치에 있는 텍스트와 연결된 글꼴 컬렉션을 가져옵니다.
GetFontCount 글꼴 목록의 글꼴 수를 가져옵니다.
GetFontCount 집합의 총 글꼴 수를 가져옵니다.
GetFontDownloadQueue 이 팩터리 개체와 연결된 글꼴 다운로드 큐를 가져옵니다.
GetFontFaceIndex 글꼴 파일 또는 파일에서 글꼴 면의 인덱스(0부터 시작)를 가져옵니다. 글꼴 파일에 단일 면이 포함된 경우 반환 값은 0입니다.
GetFontFaceIndex 글꼴 파일 내에서 글꼴 면의 인덱스(0부터 시작)를 검색합니다.
GetFontFaceReference 이 글꼴을 식별하는 글꼴 얼굴 참조를 가져옵니다. (IDWriteFont3.GetFontFaceReference)
GetFontFaceReference 이 글꼴을 식별하는 글꼴 얼굴 참조를 가져옵니다. (IDWriteFontFace3.GetFontFaceReference)

GetFontFaceReference 0부터 시작하는 인덱스가 지정된 글꼴 얼굴 참조를 가져옵니다. (IDWriteFontFamily1.GetFontFaceReference)
GetFontFaceReference 0부터 시작하는 인덱스가 지정된 글꼴 얼굴 참조를 가져옵니다. (IDWriteFontList1.GetFontFaceReference)
GetFontFaceReference 로컬 또는 원격일 수 있는 지정된 인덱스에서 글꼴에 대한 참조를 가져옵니다.
GetFontFaceReference 단일 항목의 글꼴 얼굴 참조를 검색합니다.
GetFontFallback 현재 대체를 가져옵니다. 레이아웃을 만든 이후 설정되지 않은 경우 nullptr이 됩니다.
GetFontFallback 현재 글꼴 대체 개체를 가져옵니다.
GetFontFamily 지정된 글꼴이 속한 글꼴 패밀리를 가져옵니다.
GetFontFamily 0부터 시작하는 글꼴 패밀리 인덱스가 지정된 글꼴 패밀리 개체를 만듭니다.
GetFontFamily 0부터 시작하는 글꼴 패밀리 인덱스가 지정된 글꼴 패밀리 개체를 만듭니다.
GetFontFamilyCount 컬렉션의 글꼴 패밀리 수를 가져옵니다.
GetFontFamilyModel 글꼴 컬렉션에서 패밀리를 그룹화하기 위해 사용하는 글꼴 패밀리 모델을 검색합니다.
GetFontFamilyName 글꼴 패밀리 이름의 복사본을 가져옵니다.

GetFontFamilyName 텍스트의 글꼴 패밀리 이름을 지정된 위치에 복사합니다.
GetFontFamilyNameLength 글꼴 패밀리 이름의 길이를 가져옵니다.
GetFontFamilyNameLength 현재 위치에 있는 글꼴 패밀리 이름의 길이를 가져옵니다.
GetFontFeature 지정된 인덱스에서 글꼴 기능을 가져옵니다.
GetFontFeatureCount 현재 글꼴의 OpenType 글꼴 기능 수를 가져옵니다.
GetFontFile 글꼴 얼굴을 나타내는 글꼴 파일을 가져옵니다.
GetFontFile 리소스의 글꼴 파일을 검색합니다.
GetFontFromFontFace 지정된 글꼴 얼굴 개체와 동일한 실제 글꼴에 해당하는 글꼴 개체를 가져옵니다. 지정한 실제 글꼴이 글꼴 컬렉션에 속해야 합니다.
GetFontLocality 0부터 시작하는 인덱스가 지정된 글꼴의 현재 위치를 가져옵니다. (IDWriteFontFamily1.GetFontLocality)
GetFontLocality 0부터 시작하는 인덱스가 지정된 글꼴의 현재 위치를 가져옵니다. (IDWriteFontList1.GetFontLocality)
GetFontLocality 단일 항목의 지역성을 검색합니다.

GetFontResource 이 글꼴 면의 기본 글꼴 리소스를 검색합니다.
GetFontSet 이 컬렉션에서 사용하는 기본 글꼴 집합을 가져옵니다.
GetFontSet 이 컬렉션에서 사용하는 기본 글꼴 집합을 검색합니다.
GetFontSet 이 패밀리에서 사용하는 기본 글꼴 집합을 검색합니다.
GetFontSet 이 목록에서 사용하는 기본 글꼴 집합을 검색합니다.
GetFontSignature 지정된 글꼴에서 글꼴 서명을 읽습니다. (오버로드 2/2)
GetFontSignature 지정된 글꼴에서 글꼴 서명을 읽습니다. (오버로드 1/2)
GetFontSize DIP 통합의 글꼴 크기를 가져옵니다.
GetFontSize 지정된 위치에 있는 텍스트의 글꼴 em 높이를 가져옵니다.
GetFontSourceName 글꼴 원본 이름(지정된 글꼴의 경우)을 출력 배열에 복사합니다.
GetFontSourceNameLength 지정된 글꼴의 글꼴 원본 이름 길이를 검색합니다.
GetFontSourceType 지정된 글꼴의 글꼴 원본 형식을 검색합니다.

GetFontStretch 텍스트의 글꼴 스트레치를 가져옵니다.
GetFontStretch 지정된 위치에 있는 텍스트의 글꼴 스트레치를 가져옵니다.
GetFontStyle 텍스트의 글꼴 스타일을 가져옵니다.
GetFontStyle 지정된 위치에 있는 텍스트의 글꼴 스타일(기울기라고도 함)을 가져옵니다.
GetFontWeight 텍스트의 글꼴 두께를 가져옵니다.
GetFontWeight 지정된 위치에 있는 텍스트의 글꼴 두께를 가져옵니다.
GetGamma 감마 보정에 사용되는 감마 값을 가져옵니다. 유효한 값은 0보다 커야 하며 256을 초과할 수 없습니다.
GetGdiCompatibleGlyphAdvances 문자 모양 시퀀스에 대한 픽셀 맞춤 어드밴스를 반환합니다.
GetGdiCompatibleGlyphMetrics GDI에서 생성하는 값과 호환되는 반환 값을 사용하여 글꼴 디자인 단위의 문자 모양 메트릭을 가져옵니다.
GetGdiCompatibleGlyphPlacements 글꼴 및 쓰기 시스템의 렌더링 규칙에 따라 GetGlyphs 메서드의 문자 모양 출력을 배치합니다.
GetGdiCompatibleMetrics 글꼴 면에 대한 디자인 단위 및 공통 메트릭을 가져옵니다. 이러한 메트릭은 글꼴면 내의 모든 문자 모양에 적용할 수 있으며 레이아웃 계산을 위해 애플리케이션에서 사용됩니다. (IDWriteFontFace.GetGdiCompatibleMetrics)

GetGdiCompatibleMetrics 글꼴 면에 대한 디자인 단위 및 공통 메트릭을 가져옵니다. 이러한 메트릭은 글꼴면 내의 모든 문자 모양에 적용할 수 있으며 레이아웃 계산을 위해 애플리케이션에서 사용됩니다. (IDWriteFontFace1.GetGdiCompatibleMetrics)
GetGdiInterop GDI와의 상호 운용성에 사용되는 개체를 만듭니다.
GetGenerationCount 다운로드가 완료된 후 실패 또는 성공 여부에 관계없이 매번 증가하는 다운로드 큐의 현재 생성 번호를 가져옵니다. 이 쿠키 값을 캐시된 데이터와 비교하여 부실인지 확인할 수 있습니다.
GetGlyphCount 글꼴 면의 문자 모양 수를 가져옵니다.
GetGlyphImageData 원하는 이미지 형식에 따라 문자 모양 데이터에 대한 포인터를 가져옵니다.
GetGlyphImageFormats 전체 글꼴에서 지원하는 모든 문자 모양 이미지 형식을 가져옵니다. (IDWriteFontFace4.GetGlyphImageFormats)
GetGlyphImageFormats 전체 글꼴에서 지원하는 모든 문자 모양 이미지 형식을 가져옵니다. (오버로드 1/2)
GetGlyphIndices 글꼴 'CMAP' 테이블에 정의된 대로 UCS4 유니코드 코드 포인트의 명목 매핑을 문자 모양 인덱스에 반환합니다.
GetGlyphOrientationTransform 문자 모양 실행을 그리는 각 각도에 대해 2x3 변환 행렬을 반환합니다. (IDWriteTextAnalyzer1.GetGlyphOrientationTransform)
GetGlyphOrientationTransform 문자 모양 실행을 그리는 각 각도에 대해 2x3 변환 행렬을 반환합니다. (IDWriteTextAnalyzer2.GetGlyphOrientationTransform)

GetGlyphPlacements 글꼴 및 쓰기 시스템의 렌더링 규칙에 따라 GetGlyphs 메서드의 문자 모양 출력을 배치합니다.
GetGlyphRunOutline 개요 싱크 인터페이스로 다시 호출하여 문자 모양 실행의 개요를 계산합니다.
GetGlyphs 입력 텍스트 문자열을 구문 분석하고 글꼴 및 쓰기 시스템의 렌더링 규칙에 따라 문자 모양 및 연결된 문자 모양 데이터 집합에 매핑합니다.
GetGrayscaleEnhancedContrast 회색조 앤티앨리어싱에 사용할 대비 향상의 양을 가져옵니다.
GetGridFitMode 그리드 맞춤 모드를 가져옵니다.
GetIncrementalTabStop 증분 탭 중지 위치를 가져옵니다.
GetIndex 글꼴 파일의 컨텍스트에서 글꼴 면의 인덱스를 가져옵니다.
GetInformationalStrings 로캘 이름으로 인덱싱된 지정된 정보 문자열을 포함하는 지역화된 문자열 컬렉션을 가져옵니다.
GetInformationalStrings 로캘 이름으로 인덱싱된 지정된 정보 문자열을 포함하는 지역화된 문자열 컬렉션을 가져옵니다.
GetInlineObject 지정된 위치에 있는 인라인 개체를 가져옵니다.
GetJustificationOpportunities 텍스트 및 셰이핑 문자 모양 속성이 지정된 각 문자 모양에 대한 근거 기회 정보를 검색합니다.
GetJustifiedGlyphs 정당화가 카시다 아랍어와 같은 문자 모양의 발전을 증가 복잡한 스크립트에 대한 새로운 문자 모양을 채웁니다.

GetKerningPairAdjustments 글꼴의 커닝 테이블에서 커닝 쌍 조정을 검색합니다.
GetLastLineWrapping 마지막 줄의 래핑 모드를 가져옵니다.
GetLastLineWrapping 마지막 줄의 마지막 단어가 래핑되는지 여부를 가져옵니다.
GetLastWriteTime 파일의 마지막으로 수정된 시간을 가져옵니다.
GetLastWriteTimeFromKey 글꼴 파일 참조 키에서 파일의 마지막 쓰기 시간을 가져옵니다.
GetLineMetrics 텍스트 문자열의 각 개별 텍스트 줄에 대한 정보를 검색합니다.
GetLineMetrics 각 줄의 속성을 검색합니다.
GetLineSpacing 여러 줄 텍스트 단락의 줄 간격 조정 집합을 가져옵니다. (IDWriteTextFormat.GetLineSpacing)
GetLineSpacing 여러 줄 텍스트 단락의 줄 간격 조정 집합을 가져옵니다. (IDWriteTextFormat2.GetLineSpacing)
GetLineSpacing 줄 간격 정보를 가져옵니다.
GetLoader 글꼴 파일 개체와 연결된 파일 로더를 가져옵니다.
GetLocaleName 지정된 인덱스가 있는 로캘 이름을 지정된 배열에 복사합니다. (IDWriteLocalizedStrings.GetLocaleName)

GetLocaleName 텍스트 분석의 영향을 받는 범위의 로캘 이름을 가져옵니다.
GetLocaleName 로캘 이름의 복사본을 가져옵니다.
GetLocaleName 지정된 위치에 있는 텍스트의 로캘 이름을 가져옵니다.
GetLocaleName 지정된 인덱스가 있는 로캘 이름을 지정된 배열에 복사합니다. (IDWriteStringList.GetLocaleName)
GetLocaleNameLength 지정된 인덱스가 있는 로캘 이름의 문자 길이(null 종결자를 포함하지 않음)를 가져옵니다. (IDWriteLocalizedStrings.GetLocaleNameLength)
GetLocaleNameLength 로캘 이름의 길이를 가져옵니다.
GetLocaleNameLength 지정된 위치에 있는 텍스트의 로캘 이름 길이를 가져옵니다.
GetLocaleNameLength 지정된 인덱스가 있는 로캘 이름의 문자 길이(null 종결자를 포함하지 않음)를 가져옵니다. (IDWriteStringList.GetLocaleNameLength)
GetLocalFileSize 항상 GetFullSize보다 작거나 같은 글꼴 얼굴의 로컬 크기를 바이트 단위로 가져옵니다. 지역성이 원격인 경우 이 값은 0입니다. 가득 차면 이 값은 GetFileSize와 같습니다.
GetLocalFileSize GetLocalFileSize는 현재 로컬인 글꼴 파일의 바이트 수를 반환하며, GetFileSize에서 반환하는 전체 파일 크기보다 작거나 같아야 합니다.
GetLocality 글꼴의 현재 지역성을 가져옵니다.

GetLocality 이 글꼴 얼굴 참조의 지역성을 가져옵니다.
GetLocality 파일의 현재 지역성을 가져옵니다.
GetLocalityFromKey 고유 키로 식별된 파일 리소스의 지역성을 가져옵니다.
GetMatchingFonts 지정된 속성과 얼마나 잘 일치하는지 순서대로 순위가 지정된 글꼴 패밀리의 글꼴 목록을 가져옵니다.
GetMatchingFonts 지정된 축 값과 얼마나 일치하는지 순서대로 순위가 지정된 글꼴 패밀리의 글꼴 목록을 검색합니다.
GetMatchingFonts 지정된 축 값과 얼마나 일치하는지 순서대로 순위가 지정된 글꼴 패밀리의 글꼴 목록을 검색합니다.
GetMatchingFonts 지정된 속성으로 필터링된 글꼴의 하위 집합을 반환합니다. (오버로드 2/2)
GetMatchingFonts 지정된 속성으로 필터링된 글꼴의 하위 집합을 반환합니다. (오버로드 1/2)
GetMatchingFonts 요청된 입력에 따라 일치하는 글꼴 집합을 검색하여 더 가까운 일치 항목이 더 일찍 정렬되도록 정렬합니다.
GetMatchingFonts 요청된 입력에 따라 일치하는 글꼴 집합을 생성하고, 더 가까운 일치 항목이 더 일찍 정렬되도록 정렬됩니다.
GetMatchingFontsByLOGFONT 지정된 LOGFONT 값을 기반으로 일치하는 글꼴 목록을 가져옵니다. 해당 패밀리 이름의 글꼴만 반환됩니다.

GetMaxHeight 레이아웃 최대 높이를 가져옵니다.
GetMaxWidth 레이아웃 최대 너비를 가져옵니다.
GetMemoryDC 메모리 디바이스 컨텍스트에 대한 핸들을 가져옵니다.
GetMetrics 글꼴 면에 대한 디자인 단위 및 공통 메트릭을 가져옵니다. 이러한 메트릭은 글꼴 면 내의 모든 문자 모양에 적용할 수 있으며 레이아웃 계산을 위해 애플리케이션에서 사용됩니다. (IDWriteFont.GetMetrics)
GetMetrics 글꼴 면에 대한 디자인 단위 및 공통 메트릭을 가져옵니다. 이러한 메트릭은 글꼴 면 내의 모든 문자 모양에 적용할 수 있으며 레이아웃 계산을 위해 애플리케이션에서 사용됩니다. (IDWriteFontFace.GetMetrics)
GetMetrics IDWriteTextLayout은 이 콜백 함수를 호출하여 인라인 개체의 측정값을 가져옵니다.
GetMetrics 형식이 지정된 문자열에 대한 전체 메트릭을 검색합니다. (IDWriteTextLayout.GetMetrics)
GetMetrics 글꼴 면에 대한 디자인 단위 및 공통 메트릭을 가져옵니다. 이러한 메트릭은 글꼴 면 내의 모든 문자 모양에 적용할 수 있으며 레이아웃 계산을 위해 애플리케이션에서 사용됩니다. (IDWriteFont1.GetMetrics)
GetMetrics 글꼴 면에 대한 디자인 단위 및 공통 메트릭을 가져옵니다. 이러한 메트릭은 글꼴 면 내의 모든 문자 모양에 적용할 수 있으며 레이아웃 계산을 위해 애플리케이션에서 사용됩니다. (IDWriteFontFace1.GetMetrics)
GetMetrics 형식이 지정된 문자열에 대한 전체 메트릭을 검색합니다. (IDWriteTextLayout2.GetMetrics)

GetNumberSubstitution 텍스트 분석의 영향을 받는 텍스트 범위에서 숫자 대체를 가져옵니다.
GetOpticalAlignment 텍스트 형식의 광학 여백 맞춤을 가져옵니다.
GetOpticalAlignment 문자 모양이 여백 가장자리와 정렬되는 방식을 가져옵니다.
GetOverhangMetrics IDWriteTextLayout은 이 콜백 함수를 호출하여 인라인 개체의 표시되는 익스텐트(DIP)를 가져옵니다. 패딩이 없고 오버행이 없는 간단한 비트맵의 경우 모든 오버행은 단순히 0이 됩니다.
GetOverhangMetrics 레이아웃의 오버행과 텍스트 문자 모양 및 인라인 개체를 포함하여 레이아웃에 포함된 모든 개체를 반환합니다.
GetPairKerning 지정된 위치에서 쌍 커닝을 사용할 수 있는지 여부를 가져옵니다.
GetPaletteEntries 글꼴의 색상표에서 색 값을 가져옵니다.
GetPaletteEntryCount 각 색상표의 항목 수를 가져옵니다.
GetPanose 글꼴에서 PANOSE 값을 가져오고 글꼴 선택 및 일치에 사용됩니다.
GetPanose 글꼴 선택 및 일치에 사용되는 글꼴에서 PANOSE 값을 가져옵니다.
GetParagraphAlignment 레이아웃 상자의 위쪽 및 아래쪽 가장자리를 기준으로 하는 단락의 맞춤 옵션을 가져옵니다.
GetParagraphReadingDirection 단락 읽기 방향을 가져옵니다.

GetPixelGeometry 렌더링 매개 변수 개체의 픽셀 기하 도형을 가져옵니다.
GetPixelsPerDip DIP당 비트맵 픽셀 수를 가져옵니다.
GetPixelsPerDip DIP당 실제 픽셀 수를 가져옵니다.
GetPropertyOccurrenceCount 집합에서 지정된 속성 값이 발생하는 횟수를 반환합니다.
GetPropertyValues 특정 글꼴 항목 인덱스의 속성 값을 반환합니다. (오버로드 2/3)
GetPropertyValues 특정 글꼴 항목 인덱스의 속성 값을 반환합니다. (오버로드 1/3)
GetPropertyValues 특정 글꼴 항목 인덱스의 속성 값을 반환합니다. (오버로드 3/3)
GetReadingDirection 단락의 텍스트에 대한 현재 읽기 방향을 가져옵니다.
GetRecommendedRenderingMode 지정된 크기 및 렌더링 매개 변수를 사용하여 글꼴에 권장되는 렌더링 모드를 결정합니다. (IDWriteFontFace.GetRecommendedRenderingMode)
GetRecommendedRenderingMode 지정된 크기 및 렌더링 매개 변수를 사용하여 글꼴에 권장되는 렌더링 모드를 결정합니다. (IDWriteFontFace1.GetRecommendedRenderingMode)
GetRecommendedRenderingMode 글꼴, 크기, 월드 변환 및 측정 모드에 따라 사용할 권장 텍스트 렌더링 및 그리드 맞춤 모드를 결정합니다. (IDWriteFontFace3.GetRecommendedRenderingMode)

GetRecommendedRenderingMode 글꼴, 크기, 월드 변환 및 측정 모드에 따라 사용할 권장 텍스트 렌더링 및 그리드 맞춤 모드를 결정합니다. (IDWriteFontFace2.GetRecommendedRenderingMode)
GetReferenceKey 글꼴 파일의 참조 키에 대한 포인터를 가져옵니다. 반환된 포인터는 글꼴 파일 개체가 해제될 때까지 유효합니다.
GetRenderingMode 렌더링 매개 변수 개체의 렌더링 모드를 가져옵니다.
GetRenderingMode1 렌더링 모드를 가져옵니다.
GetResult 비동기 작업의 결과를 반환합니다. 작업이 아직 완료되지 않은 경우 반환 값은 E_PENDING.
GetScriptProperties 지정된 스크립트의 속성을 검색합니다.
GetSimulations 지정된 글꼴에 적용되는 시뮬레이션을 나타내는 값을 가져옵니다.
GetSimulations 글꼴 면의 알고리즘 스타일 시뮬레이션 플래그를 가져옵니다. (IDWriteFontFace.GetSimulations)
GetSimulations 글꼴 면의 알고리즘 스타일 시뮬레이션 플래그를 가져옵니다. (IDWriteFontFaceReference.GetSimulations)
GetSize 대상 비트맵의 차원을 가져옵니다.
GetStretch 지정된 글꼴의 늘이기 또는 너비를 가져옵니다.

GetStretch 이 글꼴의 스트레치(너비라고도 함)를 가져옵니다.
GetStrikethrough 지정된 위치에 있는 텍스트의 취소선 존재를 가져옵니다.
GetString 지정된 인덱스가 있는 문자열을 지정된 배열에 복사합니다. (IDWriteLocalizedStrings.GetString)
GetString 지정된 인덱스가 있는 문자열을 지정된 배열에 복사합니다. (IDWriteStringList.GetString)
GetStringLength 지정된 인덱스가 있는 문자열의 길이(null 종결자를 포함하지 않음)를 가져옵니다. (IDWriteLocalizedStrings.GetStringLength)
GetStringLength 지정된 인덱스가 있는 문자열의 길이(null 종결자를 포함하지 않음)를 가져옵니다. (IDWriteStringList.GetStringLength)
GetStyle 지정된 글꼴의 스타일 또는 기울기를 가져옵니다.
GetStyle 이 글꼴의 스타일(기울기라고도 함)을 가져옵니다.
GetSystemFontCollection 설치된 글꼴 집합을 나타내는 개체를 가져옵니다.
GetSystemFontCollection 시스템 글꼴의 두께/너비/기울기 트리를 검색합니다.
GetSystemFontCollection 패밀리로 그룹화된 글꼴 컬렉션을 검색합니다. (IDWriteFactory6::GetSystemFontCollection)
GetSystemFontCollection 패밀리로 그룹화된 글꼴 컬렉션을 검색합니다. (IDWriteFactory7::GetSystemFontCollection)

GetSystemFontFallback 시스템 글꼴 대체 목록에서 글꼴 대체 개체를 만듭니다.
GetSystemFontSet 시스템 글꼴 목록을 검색합니다.
GetSystemFontSet 시스템 글꼴 집합을 검색합니다. (IDWriteFactory6::GetSystemFontSet)
GetSystemFontSet 시스템 글꼴 집합을 검색합니다. (IDWriteFactory7::GetSystemFontSet)
GetTextAlignment 레이아웃 상자의 선행 및 후행 가장자리를 기준으로 텍스트의 맞춤 옵션을 가져옵니다.
GetTextAntialiasMode 비트맵 렌더링 대상의 현재 텍스트 앤티앨리어싱 모드를 가져옵니다.
GetTextAtPosition 지정된 텍스트 위치에서 시작하는 텍스트 블록을 가져옵니다.
GetTextBeforePosition 지정한 위치 바로 앞에 있는 텍스트 블록을 가져옵니다.
GetTextComplexity 텍스트의 복잡성과 전체 스크립트 셰이핑을 위해 IDWriteTextAnalyzer::GetGlyphs를 호출해야 하는지 여부를 결정합니다.
GetTrimming 레이아웃 상자를 오버플로하는 텍스트에 대한 트리밍 옵션을 가져옵니다.
GetType 글꼴 면의 파일 형식 형식을 가져옵니다.
GetTypographicFeatures 스크립트 또는 글꼴에 사용할 수 있는 OpenType 기능의 전체 목록을 반환합니다.

GetTypography 지정된 위치에 있는 텍스트의 입력 체계 설정을 가져옵니다.
GetUnderline 지정된 위치에 있는 텍스트의 밑줄 존재를 가져옵니다.
GetUnicodeRanges 글꼴에서 지원하는 문자 범위 목록을 검색합니다.
GetUnicodeRanges 글꼴에서 지원하는 문자 범위 목록을 검색합니다.
GetVerticalGlyphOrientation 텍스트 분석기에서 원하는 문자 모양 방향 및 확인된 bidi 수준을 가져오는 데 사용됩니다.
GetVerticalGlyphOrientation 세로 읽기 방향을 사용할 때 문자 모양의 기본 방향을 가져옵니다. (IDWriteTextFormat1.GetVerticalGlyphOrientation)
GetVerticalGlyphOrientation 세로 읽기 방향을 사용할 때 문자 모양의 기본 방향을 가져옵니다. (IDWriteTextLayout2.GetVerticalGlyphOrientation)
GetVerticalGlyphVariants GetGlyphIndices에서 검색된 명목 문자 모양의 세로 형식을 검색합니다.
GetWaitHandle 비동기 작업이 완료될 때까지 기다리는 데 사용할 수 있는 핸들을 반환합니다. 인터페이스가 해제될 때까지 핸들은 유효한 상태로 유지됩니다.
GetWeight 지정된 글꼴의 두께 또는 스트로크 두께를 가져옵니다.
GetWeight 이 글꼴의 가중치를 가져옵니다.

GetWordWrapping 단어 줄 바꿈 옵션을 가져옵니다.
HasCharacter 글꼴이 지정된 문자를 지원하는지 여부를 결정합니다.
HasCharacter 글꼴이 지정된 문자를 지원하는지 여부를 결정합니다.
HasKerningPairs 글꼴이 쌍 커닝을 지원하는지 여부를 결정합니다.
HasVariations 이 글꼴 얼굴의 리소스가 변수 축을 지원하는지 여부를 결정합니다. (IDWriteFontFace5::HasVariations)
HasVariations 이 글꼴 얼굴의 리소스가 변수 축을 지원하는지 여부를 결정합니다. (IDWriteFontResource::HasVariations)
HasVerticalGlyphVariants 글꼴에 세로 문자 모양 변형이 있는지 여부를 확인합니다.
HitTestPoint 애플리케이션은 레이아웃 상자의 왼쪽 위 위치를 기준으로 특정 픽셀 위치를 전달하는 이 함수를 호출하고 적중 테스트가 발생한 텍스트 문자열의 특파원 적중 테스트 메트릭에 대한 정보를 가져옵니다.
HitTestTextPosition 애플리케이션은 텍스트 위치와 위치의 논리적 측면이 지정된 레이아웃 상자의 왼쪽 위를 기준으로 픽셀 위치를 가져오기 위해 이 함수를 호출합니다.
HitTestTextRange 애플리케이션은 이 함수를 호출하여 텍스트 위치 범위에 해당하는 적중 테스트 메트릭 집합을 가져옵니다. 주요 사용법 중 하나는 텍스트 문자열의 강조 표시 선택을 구현하는 것입니다.

InvalidateLayout 레이아웃을 무효화하여 메트릭 또는 그리기 함수를 호출하기 전에 레이아웃을 강제로 다시 측정합니다. 이는 글꼴의 지역성이 변경되고 레이아웃을 다시 그려야 하거나 클라이언트가 구현한 IDWriteInlineObject의 크기가 변경되는 경우에 유용합니다.
IsCharacterLocal 글꼴에서 문자를 로컬로 다운로드할지 여부를 결정합니다.
IsColorFont 색 렌더링 경로가 잠재적으로 필요한지 여부를 결정할 수 있습니다.
IsColorFont 색 렌더링 경로가 잠재적으로 필요한지 여부를 확인할 수 있습니다.
IsEmpty 다운로드 큐가 비어 있는지 여부를 확인합니다. 큐에는 이미 다운로드 중인 요청이 포함되지 않습니다. BeginDownload를 호출하면 큐가 종료됩니다.
IsGlyphLocal 글꼴에서 문자 모양이 로컬로 다운로드되는지 여부를 확인합니다.
IsMonospacedFont 글꼴이 모노스페이스인지, 즉 문자가 동일한 고정 피치 너비(비례하지 않음)인지 확인합니다.
IsMonospacedFont 텍스트 범위의 글꼴이 모노스페이스인지, 즉 글꼴 문자가 동일한 고정 피치 너비인지 여부를 결정합니다.
IsSymbolFont 글꼴이 기호 글꼴인지 여부를 확인합니다. (IDWriteFont.IsSymbolFont)
IsSymbolFont 글꼴이 기호 글꼴인지 여부를 확인합니다. (IDWriteFontFace.IsSymbolFont)
JustifyGlyphAdvances 선 너비에 맞게 문자 모양 어드밴스 배열을 양쪽에 맞춥니다.

MapCharacters 텍스트의 시작 범위를 렌더링하는 데 사용할 적절한 글꼴을 결정합니다.
MoveNext 컬렉션의 다음 글꼴 파일로 진행합니다. 열거자가 처음 만들어지면 컬렉션의 첫 번째 요소 앞에 열거자가 배치되고 MoveNext에 대한 첫 번째 호출이 첫 번째 파일로 이동합니다.
MoveNext 열거자에서 다음 문자 모양 실행으로 이동합니다.
ReadFileFragment 글꼴 파일에서 조각을 읽습니다.
RegisterFontCollectionLoader 팩터리 개체에 사용자 지정 글꼴 컬렉션 로더를 등록합니다.
RegisterFontFileLoader 글꼴 파일 로더를 DirectWrite 등록합니다.
ReleaseFileFragment 파일에서 조각을 해제합니다.
ReleaseFontTable TryGetFontTable에서 이전에 가져온 테이블을 해제합니다.
ReleaseGlyphImageData ReadGlyphData에서 가져온 테이블 데이터를 해제합니다.
Removelistener 이전에 AddListener를 사용하여 등록한 알림 처리기를 등록 취소합니다.
크기 조정 비트맵의 크기를 조정합니다.
SetAutomaticFontAxes 자동 글꼴 축 옵션을 설정합니다.

SetBidiLevel 실행 변경당 한 번 호출되는 범위의 양방향 수준을 설정합니다(명시적 또는 해결된 암시적).
SetCharacterSpacing 문자 사이의 간격을 설정합니다.
SetCurrentTransform 추상 좌표를 DIP(디바이스 독립적 픽셀)에 매핑하는 변환을 설정합니다. 이는 기본 디바이스 컨텍스트의 월드 변환에 영향을 주지 않습니다.
SetDrawingEffect 애플리케이션 정의 그리기 효과를 설정합니다.
SetFlowDirection 단락 흐름 방향을 설정합니다.
SetFontAxisValues 형식의 글꼴 축에 대한 값을 설정합니다.
SetFontCollection 글꼴 컬렉션을 설정합니다.
SetFontFallback 레이아웃에 사용자 지정 글꼴 대체를 적용합니다. 설정되지 않은 경우 기본 시스템 대체 목록을 사용합니다.
SetFontFallback 레이아웃에 사용자 지정 글꼴 대체를 적용합니다.
SetFontFamilyName 지정된 텍스트 범위 내의 텍스트에 대해 null로 끝나는 글꼴 패밀리 이름을 설정합니다.
SetFontSize 지정된 텍스트 범위 내의 텍스트에 대한 글꼴 크기를 DIP 단위로 설정합니다.
SetFontStretch 지정된 텍스트 범위 내에서 텍스트의 글꼴 늘이기를 설정합니다.

SetFontStyle DWRITE_TEXT_RANGE 구조체로 지정된 텍스트 범위 내에서 텍스트의 글꼴 스타일을 설정합니다.
SetFontWeight DWRITE_TEXT_RANGE 구조체로 지정된 텍스트 범위 내에서 텍스트의 글꼴 두께를 설정합니다.
SetGlyphOrientation 텍스트 분석기는 이를 다시 호출하여 셰이핑 및 그리기에 대한 각 문자의 실제 방향을 보고합니다.
SetIncrementalTabStop 인접한 두 탭 정지 사이의 고정 거리를 설정합니다.
SetInlineObject 인라인 개체를 설정합니다.
SetLastLineWrapping 마지막 줄의 래핑 모드를 설정합니다.
SetLastLineWrapping 마지막 줄의 마지막 단어를 래핑할지 여부를 설정합니다.
SetLineBreakpoints 지정된 위치에서 시작하여 각 문자에 대한 줄 바꿈 기회를 설정합니다.
SetLineSpacing 줄 간격을 설정합니다.
SetLineSpacing 줄 간격을 설정합니다. (IDWriteTextFormat2.SetLineSpacing)
SetLineSpacing 줄 간격을 설정합니다. (IDWriteTextLayout3.SetLineSpacing)
SetLocaleName 지정된 텍스트 범위 내에서 텍스트의 로캘 이름을 설정합니다.

SetMaxHeight 레이아웃 최대 높이를 설정합니다.
SetMaxWidth 레이아웃 최대 너비를 설정합니다.
SetNumberSubstitution 텍스트 분석의 영향을 받는 텍스트 범위의 숫자 대체를 설정합니다.
SetOpticalAlignment 텍스트 형식의 광학 여백 맞춤을 설정합니다.
SetOpticalAlignment 문자 모양이 여백 가장자리와 정렬되는 방식을 설정합니다.
SetPairKerning 지정된 텍스트 범위에서 쌍 커닝을 사용하거나 사용하지 않도록 설정합니다.
SetParagraphAlignment 레이아웃 상자의 위쪽 및 아래쪽 가장자리를 기준으로 단락의 맞춤 옵션을 설정합니다.
SetPixelsPerDip DIP당 비트맵 픽셀 수(디바이스 독립적 픽셀)를 설정합니다. DIP는 1/96인치이므로 인치당 픽셀을 96으로 나눈 경우 이 값은 숫자입니다.
SetReadingDirection 단락 읽기 방향을 설정합니다.
SetScriptAnalysis 지정된 텍스트 범위에 대한 스크립트 분석을 보고합니다.
SetStrikethrough 지정된 텍스트 범위 내의 텍스트에 대한 취소선 을 설정합니다.
SetTextAlignment IDWriteTextFormat 인터페이스에 대한 레이아웃 상자의 선행 및 후행 가장자리를 기준으로 단락의 텍스트 맞춤을 설정합니다.

SetTextAntialiasMode 비트맵 렌더링 대상의 현재 텍스트 앤티앨리어싱 모드를 설정합니다.
SetTrimming 레이아웃 너비를 오버플로하는 텍스트에 대한 트리밍 옵션을 설정합니다.
SetTypography 지정된 텍스트 범위 내의 텍스트에 대한 글꼴 입력 체계 기능을 설정합니다.
SetUnderline 지정된 텍스트 범위 내의 텍스트에 대한 밑면을 설정합니다.
SetVerticalGlyphOrientation 텍스트 형식의 방향을 설정합니다.
SetVerticalGlyphOrientation 세로 읽기 방향을 사용할 때 문자 모양에 대한 기본 방향을 설정합니다.
SetWordWrapping 단어 줄 바꿈 옵션을 설정합니다.
TranslateColorGlyphRun 문자 모양 실행을 원래 "기본" 실행의 색 표현을 생성하도록 렌더링할 수 있는 색 문자 모양 실행 시퀀스로 변환합니다.
TranslateColorGlyphRun 이 메서드는 여러 색 문자 모양 실행으로 변환하기 위해 문자 모양 실행에서 호출됩니다.
TryGetFontTable 지정한 OpenType 글꼴 테이블이 있으면 찾아 포인터를 반환합니다. 함수는 글꼴 파일 로더에 의해 구현된 IDWriteFontFileStream 인터페이스를 통해 기본 글꼴 데이터에 액세스합니다.
UnpackFontFile UnpackFontFile 메서드는 컨테이너 파일(WOFF 또는 WOFF2)에서 글꼴 데이터의 압축을 풀고 패키지되지 않은 글꼴 데이터를 글꼴 파일 스트림 형식으로 반환합니다.

UnregisterFontCollectionLoader RegisterFontCollectionLoader를 사용하여 이전에 등록된 사용자 지정 글꼴 컬렉션 로더의 등록을 취소합니다.
UnregisterFontFileLoader RegisterFontFileLoader를 사용하여 이전에 DirectWrite 글꼴 시스템에 등록된 글꼴 파일 로더의 등록을 취소합니다.

인터페이스

IDWriteAsyncResult 비동기 작업의 결과를 나타냅니다. 클라이언트는 인터페이스를 사용하여 작업이 완료되기를 기다렸다가 결과를 가져올 수 있습니다.
IDWriteBitmapRenderTarget 문자 모양을 렌더링하는 데 사용할 수 있는 32비트 디바이스 독립적 비트맵 및 디바이스 컨텍스트를 캡슐화합니다.
IDWriteBitmapRenderTarget1 문자 모양을 렌더링하는 데 사용할 수 있는 32비트 디바이스 독립적 비트맵 및 디바이스 컨텍스트를 캡슐화합니다.
IDWriteColorGlyphRunEnumerator 이 인터페이스를 사용하면 애플리케이션이 색 문자 모양 실행을 통해 열거할 수 있습니다.
IDWriteColorGlyphRunEnumerator1 색 문자 모양 실행의 정렬된 컬렉션에 대한 열거자입니다.
IDWriteFactory 모든 후속 DirectWrite 개체를 만드는 데 사용됩니다. 이 인터페이스는 모든 DirectWrite 개체에 대한 루트 팩터리 인터페이스입니다.
IDWriteFactory1 모든 DirectWrite 개체에 대한 루트 팩터리 인터페이스입니다. (IDWriteFactory1)

IDWriteFactory2 모든 DirectWrite 개체에 대한 루트 팩터리 인터페이스입니다. 나
IDWriteFactory3 모든 DirectWrite 개체에 대한 루트 팩터리 인터페이스입니다. (IDWriteFactory3)
IDWriteFactory4 모든 DirectWrite 개체에 대한 루트 팩터리 인터페이스입니다. (IDWriteFactory4)
IDWriteFactory5 모든 DirectWrite 개체에 대한 루트 팩터리 인터페이스입니다. (IDWriteFactory5)
IDWriteFactory6 이 인터페이스는 모든 DirectWrite 개체가 만들어지는 팩터리 개체를 나타냅니다. **IDWriteFactory6**은 글꼴 및 글꼴 리소스 작업을 위한 새로운 기능을 추가합니다.
IDWriteFactory7 이 인터페이스는 모든 DirectWrite 개체가 만들어지는 팩터리 개체를 나타냅니다. **IDWriteFactory7**은 시스템 글꼴 작업을 위한 새로운 기능을 추가합니다.
IDWriteFont 글꼴 컬렉션의 실제 글꼴을 나타냅니다. 이 인터페이스는 실제 글꼴에서 글꼴 얼굴을 만들거나 기존 글꼴 얼굴에서 글꼴 얼굴 메트릭 또는 얼굴 이름과 같은 정보를 검색하는 데 사용됩니다.
IDWriteFont1 글꼴 컬렉션의 실제 글꼴을 나타냅니다. (IDWriteFont1)
IDWriteFont2 글꼴 컬렉션의 실제 글꼴을 나타냅니다. (IDWriteFont2)
IDWriteFont3 글꼴 컬렉션의 글꼴을 나타냅니다.
IDWriteFontCollection 시스템에 설치된 글꼴 집합 또는 특정 디렉터리의 글꼴 집합과 같은 글꼴 집합을 캡슐화하는 개체입니다. (IDWriteFontCollection)

IDWriteFontCollection1 시스템에 설치된 글꼴 집합 또는 특정 디렉터리의 글꼴 집합과 같은 글꼴 집합을 캡슐화하는 개체입니다. (IDWriteFontCollection1)
IDWriteFontCollection2 이 인터페이스는 시스템에 설치된 글꼴 집합 또는 특정 디렉터리의 글꼴 집합과 같은 글꼴 집합을 캡슐화합니다. 글꼴 컬렉션 API를 사용하여 사용할 수 있는 글꼴 패밀리 및 글꼴을 검색하고 글꼴에 대한 메타데이터를 가져올 수 있습니다. (IDWriteFontCollection2)
IDWriteFontCollection3 이 인터페이스는 시스템에 설치된 글꼴 집합 또는 특정 디렉터리의 글꼴 집합과 같은 글꼴 집합을 캡슐화합니다. 글꼴 컬렉션 API를 사용하여 사용할 수 있는 글꼴 패밀리 및 글꼴을 검색하고 글꼴에 대한 메타데이터를 가져올 수 있습니다. (IDWriteFontCollection3)
IDWriteFontCollectionLoader 특정 유형의 키가 지정된 글꼴 컬렉션을 생성하는 데 사용됩니다.
IDWriteFontDownloadListener 글꼴 다운로드 큐(IDWriteFontDownloadQueue 인터페이스)에서 알림을 받는 애플리케이션 정의 콜백 인터페이스입니다.
IDWriteFontDownloadQueue 원격 글꼴, 문자, 문자 모양 및 글꼴 조각에 대한 다운로드 요청을 큐에 추가하는 인터페이스입니다.
IDWriteFontFace 이 인터페이스는 메트릭, 이름 및 문자 모양 윤곽선과 같은 다양한 글꼴 데이터를 노출합니다. 글꼴 얼굴 유형, 적절한 파일 참조 및 얼굴 식별 데이터가 포함됩니다.
IDWriteFontFace1 글꼴 얼굴 유형, 적절한 파일 참조 및 얼굴 식별 데이터를 포함합니다. (IDWriteFontFace1)
IDWriteFontFace2 글꼴 얼굴 유형, 적절한 파일 참조 및 얼굴 식별 데이터를 포함합니다. (IDWriteFontFace2)
IDWriteFontFace3 글꼴 얼굴 유형, 적절한 파일 참조 및 얼굴 식별 데이터를 포함합니다. (IDWriteFontFace3)

IDWriteFontFace4 글꼴 얼굴 유형, 적절한 파일 참조 및 얼굴 식별 데이터를 포함합니다. (IDWriteFontFace4)
IDWriteFontFace5 글꼴 얼굴 유형, 적절한 파일 참조 및 얼굴 식별 데이터를 포함합니다. (IDWriteFontFace5)
IDWriteFontFaceReference 글꼴 면에 대한 참조를 나타냅니다.
IDWriteFontFaceReference1 글꼴 면에 대한 참조를 나타냅니다. 글꼴에 대한 고유 식별 참조로, 글꼴 얼굴을 만들어 글꼴 메트릭을 쿼리하고 렌더링에 사용할 수 있습니다.
IDWriteFontFallback 글꼴 목록에서 대체 글꼴에 액세스할 수 있습니다.
IDWriteFontFallback1 글꼴 목록에서 대체 글꼴에 액세스할 수 있습니다.
IDWriteFontFallbackBuilder 유니코드 글꼴 대체 매핑을 만들고 해당 매핑에서 글꼴 대체 개체를 만들 수 있습니다.
IDWriteFontFamily 관련 글꼴 패밀리를 나타냅니다. (IDWriteFontFamily)
IDWriteFontFamily1 관련 글꼴 패밀리를 나타냅니다. (IDWriteFontFamily1)
IDWriteFontFamily2 관련 글꼴 패밀리를 나타냅니다. **IDWriteFontFamily2** 는 글꼴 축 값으로 글꼴 검색을 포함하여 새로운 기능을 추가합니다.
IDWriteFontFile 글꼴 파일을 나타냅니다. 글꼴 관리자 또는 글꼴 뷰어와 같은 애플리케이션은 IDWriteFontFile::Analyze를 호출하여 특정 파일이 글꼴 파일인지 여부와 글꼴 시스템에서 지원하는 글꼴 형식인지 여부를 확인할 수 있습니다.

IDWriteFontFileEnumerator 글꼴 파일의 컬렉션을 캡슐화합니다. 글꼴 시스템은 이 인터페이스를 사용하여 글꼴 컬렉션을 빌드할 때 글꼴 파일을 열거합니다.
IDWriteFontFileLoader 글꼴 파일 참조 키에서 글꼴 파일 스트림 개체로 특정 형식의 글꼴 파일 리소스 로드를 처리합니다.
IDWriteFontFileStream 사용자 지정 글꼴 파일 로더에서 글꼴 파일 데이터를 로드합니다.
IDWriteFontList 글꼴 목록을 나타냅니다. (IDWriteFontList)
IDWriteFontList1 글꼴 목록을 나타냅니다. (IDWriteFontList1)
IDWriteFontList2 글꼴 목록을 나타냅니다. **IDWriteFontList2** 는 목록에서 사용하는 기본 글꼴 집합을 검색하는 것을 포함하여 새로운 기능을 추가합니다.
IDWriteFontResource 글꼴 리소스에 대한 축 정보를 제공하며 특정 글꼴 얼굴 인스턴스를 만드는 데 사용됩니다.
IDWriteFontSet 글꼴 집합을 나타냅니다. (IDWriteFontSet)
IDWriteFontSet1 글꼴 집합을 나타냅니다. (IDWriteFontSet1)
IDWriteFontSet2 글꼴 집합을 나타냅니다. (IDWriteFontSet2)
IDWriteFontSet3 글꼴 집합을 나타냅니다. (IDWriteFontSet3)
IDWriteFontSet4 글꼴 집합을 나타냅니다. (IDWriteFontSet4)

IDWriteFontSetBuilder 글꼴 집합을 빌드하는 메서드를 포함합니다. (IDWriteFontSetBuilder)
IDWriteFontSetBuilder1 글꼴 집합을 빌드하는 메서드를 포함합니다. (IDWriteFontSetBuilder1)
IDWriteFontSetBuilder2 글꼴 집합을 빌드하는 메서드를 포함합니다. (IDWriteFontSetBuilder2)
IDWriteGdiInterop 글꼴 얼굴을 LOGFONT 구조로 변환하거나 GDI 글꼴 설명을 글꼴 면으로 변환하는 메서드와 같은 GDI와의 상호 운용성을 제공합니다. 비트맵 렌더링 대상 개체를 만드는 데도 사용됩니다. (IDWriteGdiInterop)
IDWriteGdiInterop1 글꼴 얼굴을 LOGFONT 구조로 변환하거나 GDI 글꼴 설명을 글꼴 면으로 변환하는 메서드와 같은 GDI와의 상호 운용성을 제공합니다. 비트맵 렌더링 대상 개체를 만드는 데도 사용됩니다. (IDWriteGdiInterop1)
IDWriteGlyphRunAnalysis 문자 모양 실행을 렌더링하는 데 사용되는 하위 수준 정보를 포함합니다.
IDWriteInlineObject 애플리케이션 정의 인라인 그래픽을 래핑하여 DWrite에서 그래픽이 텍스트가 있는 문자 모양 인라인인 것처럼 메트릭을 쿼리할 수 있도록 합니다.
IDWriteInMemoryFontFileLoader 메모리 내 글꼴에 액세스할 수 있는 글꼴 파일 로더를 나타냅니다.
IDWriteLocalFontFileLoader 로컬 글꼴 파일에서 작동하고 글꼴 파일 참조 키에서 로컬 글꼴 파일 정보를 노출하는 IDWriteFontFileLoader 인터페이스의 기본 제공 구현입니다.
IDWriteLocalizedStrings 로캘 이름으로 인덱싱된 문자열의 컬렉션을 나타냅니다.
IDWriteNumberSubstitution 지정된 로캘에 대한 적절한 숫자와 숫자 문장 부호를 보유합니다.

IDWritePixelSnapping DIP당 픽셀(디바이스 독립적 픽셀) 및 텍스트 렌더러의 현재 변환 매트릭스와 같은 픽셀 맞춤 속성을 정의합니다.
IDWriteRemoteFontFileLoader 원격(즉, 다운로드 가능한) 글꼴에 액세스할 수 있는 글꼴 파일 로더를 나타냅니다.
IDWriteRemoteFontFileStream 글꼴 파일 스트림을 나타내며, 그 중 일부는 로컬이 아닐 수 있습니다.
IDWriteRenderingParams 문자 모양 래스터화 및 필터링을 위한 ClearType 수준, 향상된 대비 및 감마 수정과 같은 텍스트 렌더링 설정을 나타냅니다.
IDWriteRenderingParams1 문자 모양 래스터화 및 필터링에 대한 텍스트 렌더링 설정을 나타냅니다. (IDWriteRenderingParams1)
IDWriteRenderingParams2 문자 모양 래스터화 및 필터링에 대한 텍스트 렌더링 설정을 나타냅니다. (IDWriteRenderingParams2)
IDWriteRenderingParams3 문자 모양 래스터화 및 필터링에 대한 텍스트 렌더링 설정을 나타냅니다. (IDWriteRenderingParams3)
IDWriteStringList 숫자로 인덱싱된 문자열의 컬렉션을 나타냅니다.
IDWriteTextAnalysisSink 이 인터페이스는 지정된 텍스트 분석의 출력을 수신하기 위해 텍스트 분석기의 클라이언트에 의해 구현됩니다.
IDWriteTextAnalysisSink1 텍스트 분석기의 출력을 수신하기 위해 구현하는 인터페이스입니다.
IDWriteTextAnalysisSource 분석기에서 텍스트를 제공하기 위해 텍스트 분석기 클라이언트에 의해 구현됩니다.

IDWriteTextAnalysisSource1 텍스트 및 연결된 텍스트 속성과 같은 텍스트 분석기에서 필요한 정보를 제공하기 위해 구현하는 인터페이스입니다.
IDWriteTextAnalyzer 아랍어, 줄 바꿈 기회 결정, 문자 모양 배치 및 숫자 대체와 같은 언어에 대한 양방향(양방향) 지원과 같은 복잡한 스크립트 처리를 위한 다양한 텍스트 속성을 분석합니다.
IDWriteTextAnalyzer1 복잡한 스크립트 처리를 위해 다양한 텍스트 속성을 분석합니다. (IDWriteTextAnalyzer1)
IDWriteTextAnalyzer2 복잡한 스크립트 처리를 위해 다양한 텍스트 속성을 분석합니다. (IDWriteTextAnalyzer2)
IDWriteTextFormat IDWriteTextFormat 인터페이스는 텍스트 서식을 지정하는 데 사용되는 글꼴 및 단락 속성을 설명하고 로캘 정보를 설명합니다.
IDWriteTextFormat1 텍스트 서식을 지정하는 데 사용되는 글꼴 및 단락 속성을 설명하고 로캘 정보를 설명합니다. (IDWriteTextFormat1)
IDWriteTextFormat2 텍스트 서식을 지정하는 데 사용되는 글꼴 및 단락 속성을 설명하고 로캘 정보를 설명합니다. (IDWriteTextFormat2)
IDWriteTextFormat3 텍스트 서식을 지정하는 데 사용되는 글꼴 및 단락 속성을 설명하고 로캘 정보를 설명합니다. (IDWriteTextFormat3)
IDWriteTextLayout IDWriteTextLayout 인터페이스는 완전히 분석되고 서식이 지정된 후 텍스트 블록을 나타냅니다.
IDWriteTextLayout1 완전히 분석되고 서식이 지정된 후의 텍스트 블록을 나타냅니다. (IDWriteTextLayout1)
IDWriteTextLayout2 완전히 분석되고 서식이 지정된 후의 텍스트 블록을 나타냅니다. (IDWriteTextLayout2)

IDWriteTextLayout3 완전히 분석되고 서식이 지정된 후의 텍스트 블록을 나타냅니다. (IDWriteTextLayout3)
IDWriteTextLayout4 완전히 분석되고 서식이 지정된 후의 텍스트 블록을 나타냅니다.
IDWriteTextRenderer 텍스트, 인라인 개체 및 밑줄과 같은 장식의 렌더링을 수행하는 애플리케이션 정의 콜백 집합을 나타냅니다. (IDWriteTextRenderer)
IDWriteTextRenderer1 텍스트, 인라인 개체 및 밑줄과 같은 장식의 렌더링을 수행하는 애플리케이션 정의 콜백 집합을 나타냅니다. (IDWriteTextRenderer1)
IDWriteTypography 글꼴 입력 체계 설정을 나타냅니다.

구조체

DWRITE_CARET_METRICS DWRITE_CARET_METRICS 구조체는 글꼴에서 caret 배치에 대한 메트릭을 지정합니다.
DWRITE_CLUSTER_METRICS 문자 모양 클러스터에 대한 정보를 포함합니다.
DWRITE_COLOR_GLYPH_RUN 문자 모양 색 정보를 사용하여 문자 모양 실행을 그리는 데 렌더러가 필요한 정보를 포함합니다.
DWRITE_COLOR_GLYPH_RUN1 색 문자 모양 실행을 나타냅니다. IDWriteFactory4::TranslateColorGlyphRun 메서드는 글꼴이 지원하는 항목에 따라 다양한 형식의 정렬된 색 문자 모양 실행 컬렉션을 반환합니다.
DWRITE_FILE_FRAGMENT 글꼴 파일의 바이트 범위를 나타냅니다.

DWRITE_FONT_AXIS_RANGE 글꼴 축에 사용할 수 있는 값의 최소 및 최대 범위를 나타냅니다.
DWRITE_FONT_AXIS_VALUE 글꼴 축의 값을 나타냅니다. 글꼴 인스턴스를 쿼리하고 만들 때 사용됩니다.
DWRITE_FONT_FEATURE 현재 글꼴 얼굴에서 입력 기능을 식별하고 실행하는 데 사용되는 속성을 지정합니다.
DWRITE_FONT_METRICS DWRITE_FONT_METRICS 구조체는 글꼴 면 내의 모든 문자 모양에 적용할 수 있는 메트릭을 지정합니다.
DWRITE_FONT_METRICS1 DWRITE_FONT_METRICS1 구조체는 글꼴 면 내의 모든 문자 모양에 적용할 수 있는 메트릭을 지정합니다.
DWRITE_FONT_PROPERTY 글꼴 집합을 필터링하고 명시적 속성을 사용하여 글꼴 집합을 빌드하는 데 사용되는 Font 속성입니다.
DWRITE_GLYPH_IMAGE_DATA GetGlyphImageData의 단일 문자 모양에 대한 데이터입니다.
DWRITE_GLYPH_METRICS 개별 문자 모양의 메트릭을 지정합니다.
DWRITE_GLYPH_OFFSET 문자 모양 위치에 대한 선택적 조정입니다.
DWRITE_GLYPH_RUN 렌더러가 문자 모양 실행을 그리는 데 필요한 정보를 포함합니다.
DWRITE_GLYPH_RUN_DESCRIPTION DWRITE_GLYPH_RUN 속성과 관련된 추가 속성을 포함합니다.

DWRITE_HIT_TEST_METRICS 적중 테스트에서 얻은 지역을 설명합니다.
DWRITE_INLINE_OBJECT_METRICS 애플리케이션 정의 인라인 개체의 기하학적 측정을 설명하는 속성을 포함합니다.
DWRITE_JUSTIFICATION_OPPORTUNITY DWRITE_JUSTIFICATION_OPPORTUNITY 구조체는 문자 모양당 근거 정보를 지정합니다.
DWRITE_LINE_BREAKPOINT 문자의 줄 중단점 특성입니다.
DWRITE_LINE_METRICS 서식이 지정된 텍스트 줄에 대한 정보를 포함합니다. (DWRITE_LINE_METRICS)
DWRITE_LINE_METRICS1 서식이 지정된 텍스트 줄에 대한 정보를 포함합니다. (DWRITE_LINE_METRICS1)
DWRITE_LINE_SPACING 텍스트 줄 사이의 세로 간격을 설정합니다.
DWRITE_MATRIX DWRITE_MATRIX 구조체는 렌더링된 문자 모양에 적용할 그래픽 변환을 지정합니다.
DWRITE_OVERHANG_METRICS 표시되는 DIP(디바이스 독립적 픽셀)가 레이아웃 또는 인라인 개체의 각 측면을 오버슈트하는 정도를 나타냅니다.
DWRITE_PANOSE DWRITE_PANOSE 공용 구조체는 IDWriteFont1::GetPanose에서 글꼴을 선택하고 일치시키는 데 사용하는 서체 분류 값을 설명합니다.
DWRITE_SCRIPT_ANALYSIS 텍스트와 해당 쓰기 시스템 스크립트의 연결과 일부 표시 특성을 저장합니다.
DWRITE_SCRIPT_PROPERTIES DWRITE_SCRIPT_PROPERTIES 구조는 캐리트 탐색 및 근거에 대한 스크립트 속성을 지정합니다.

DWRITE_SHAPING_GLYPH_PROPERTIES 출력 문자 모양에 대한 셰이핑 출력 속성을 포함합니다.
DWRITE_SHAPING_TEXT_PROPERTIES 출력 문자 모양에 대한 출력 속성 셰이핑
DWRITE_STRIKETHROUGH 취소선의 크기 및 배치에 대한 정보를 포함합니다.
DWRITE_TEXT_METRICS 레이아웃 후 텍스트와 연결된 메트릭을 포함합니다. (DWRITE_TEXT_METRICS)
DWRITE_TEXT_METRICS1 레이아웃 후 텍스트와 연결된 메트릭을 포함합니다. (DWRITE_TEXT_METRICS1)
DWRITE_TEXT_RANGE IDWriteTextLayout 개체가 나타내는 텍스트에 서식이 적용되는 텍스트 위치의 범위를 지정합니다.
DWRITE_TRIMMING 레이아웃 상자를 오버플로하는 텍스트에 대한 트리밍 옵션을 지정합니다.
DWRITE_TYPOGRAPHIC_FEATURES 텍스트 셰이핑 중에 적용할 입력 기능 집합을 포함합니다.
DWRITE_UNDERLINE 밑줄의 너비, 두께, 오프셋, 실행 높이, 읽기 방향 및 흐름 방향에 대한 정보를 포함합니다.
DWRITE_UNICODE_RANGE DWRITE_UNICODE_RANGE 구조체는 유니코드 코드 포인트의 범위를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

DirectXMath

아티클2023. 08. 22.

DirectXMath 기술 개요입니다.

DirectXMath를 개발하려면 다음 헤더가 필요합니다.

- directxcollision.h
- directxmath.h
- directxpackedvector.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- DirectXMath

열거형

ContainmentType 개체에 다른 개체가 포함되어 있는지 여부를 나타냅니다.
PlaneIntersectionType 개체가 평면과 교차하는지 여부를 나타냅니다.

Functions

BoundingBox BoundingBox 클래스의 instance 만듭니다. (BoundingBox.BoundingBox)
BoundingBox BoundingBox 클래스의 instance 만듭니다. (오버로드 2/4)
BoundingBox BoundingBox 클래스의 instance 만듭니다. (오버로드 3/4)
BoundingBox

BoundingBox 클래스의 instance 만듭니다. (오버로드 1/4)
BoundingFrustum BoundingFrustum의 instance 만듭니다. (BoundingFrustum.BoundingFrustum)
BoundingFrustum BoundingFrustum의 instance 만듭니다. (오버로드 3/4)
BoundingFrustum BoundingFrustum의 instance 만듭니다. (오버로드 1/4)
BoundingFrustum BoundingFrustum의 instance 만듭니다. (오버로드 2/4)
BoundingFrustum 왼손 프로젝션 행렬에서 BoundingFrustum의 instance 만듭니다.
BoundingOrientedBox BoundingOrientedBox의 instance 만듭니다. (BoundingOrientedBox.BoundingOrientedBox)
BoundingOrientedBox BoundingOrientedBox의 instance 만듭니다. (오버로드 1/4)
BoundingOrientedBox BoundingOrientedBox의 instance 만듭니다. (오버로드 2/4)
BoundingOrientedBox BoundingOrientedBox의 instance 만듭니다. (오버로드 3/4)
BoundingSphere BoundingSphere 클래스의 instance 만듭니다. (BoundingSphere::BoundingSphere)
BoundingSphere BoundingSphere 클래스의 instance 만듭니다. (오버로드 2/3)
BoundingSphere BoundingSphere 클래스의 instance 만듭니다. (오버로드 1/3)

ContainedBy BoundingBox가 지정된 frustum에 포함되어 있는지 여부를 테스트합니다.
ContainedBy BoundingFrustum이 지정된 frustum에 포함되어 있는지 여부를 테스트합니다.
ContainedBy BoundingOrientedBox가 frustum에 포함되어 있는지 여부를 테스트합니다.
ContainedBy BoundingSphere가 지정된 frustum에 포함되어 있는지 여부를 테스트합니다.
ContainedBy 삼각형이 6개의 평면(일반적으로 frustum) 내에 포함되어 있는지 여부를 테스트합니다.
포함 BoundingBox에 지정된 지점이 포함되어 있는지 여부를 테스트합니다.
포함 BoundingBox에 다른 BoundingBox가 포함되어 있는지 테스트합니다.
포함 BoundingBox에 지정된 BoundingFrustum이 포함되어 있는지 테스트합니다.
포함 BoundingBox에 지정된 BoundingOrientedBox가 포함되어 있는지 테스트합니다.
포함 BoundingBox에 지정된 BoundingSphere가 포함되어 있는지 여부를 테스트합니다.
포함 BoundingBox에 지정된 삼각형이 포함되어 있는지 테스트합니다.
포함 BoundingFrustum에 지정된 점이 포함되어 있는지 여부를 테스트합니다.
포함

BoundingFrustum에 지정된 BoundingBox가 포함되어 있는지 테스트합니다.
포함 BoundingFrustum에 지정된 BoundingFrustum이 포함되어 있는지 테스트합니다.
포함 BoundingFrustum에 지정된 BoundingOrientedBox가 포함되어 있는지 테스트합니다.
포함 BoundingFrustum에 지정된 BoundingSphere가 포함되어 있는지 여부를 테스트합니다.
포함 BoundingFrustum에 지정된 삼각형이 포함되어 있는지 테스트합니다.
포함 BoundingOrientedBox에 지정된 점이 포함되어 있는지 여부를 테스트합니다.
포함 BoundingOrientedBox에 BoundingBox가 포함되어 있는지 테스트합니다.
포함 BoundingOrientedBox에 BoundingFrustum이 포함되어 있는지 여부를 테스트합니다.
포함 BoundingOrientedBox에 BoundingOrientedBox가 포함되어 있는지 테스트합니다.
포함 BoundingOrientedBox에 BoundingSphere가 포함되어 있는지 테스트합니다.
포함 BoundingOrientedBox에 삼각형이 포함되어 있는지 여부를 테스트합니다.
포함 BoundingSphere에 지정된 점이 포함되어 있는지 여부를 테스트합니다.
포함 BoundingSphere에 지정된 BoundingBox가 포함되어 있는지 테스트합니다.

포함 BoundingSphere에 지정된 BoundingFrustum이 포함되어 있는지 테스트합니다.
포함 BoundingSphere에 지정된 BoundingOrientedBox가 포함되어 있는지 테스트합니다.
포함 BoundingSphere에 지정된 BoundingSphere가 포함되어 있는지 여부를 테스트합니다.
포함 BoundingSphere에 지정된 삼각형이 포함되어 있는지 테스트합니다.
CreateFromBoundingBox BoundingBox에서 BoundingOrientedBox를 만듭니다.
CreateFromBoundingBox 지정된 BoundingBox를 포함하는 BoundingSphere를 만듭니다. (BoundingSphere.CreateFromBoundingBox)
CreateFromBoundingBox 지정된 BoundingBox를 포함하는 BoundingSphere를 만듭니다. (오버로드 1/2)
CreateFromMatrix 지정된 프로젝션 행렬에서 BoundingFrustum을 만듭니다.
CreateFromPoints 두 지점에서 BoundingBox를 만듭니다.
CreateFromPoints 포인트 목록에서 BoundingBox를 만듭니다.
CreateFromPoints 점 컬렉션에서 BoundingOrientedBox를 만듭니다.
CreateFromPoints 점 목록에서 새 BoundingSphere를 만듭니다.

CreateFromSphere 지정된 BoundingSphere를 포함할 수 있을 만큼 큰 BoundingBox를 만듭니다.
CreateMerged 지정된 두 개의 BoundBox 인스턴스를 포함할 수 있을 만큼 큰 BoundingBox를 만듭니다.
CreateMerged 지정된 두 BoundingSphere 개체를 포함하는 BoundingSphere를 만듭니다.
GetCorners BoundingBox의 모서리를 검색합니다.
GetCorners BoundingFrustum을 구성하는 모서리를 가져옵니다.
GetCorners BoundingOrientedBox의 모서리를 검색합니다.
GetPlanes BoundingFrustum을 구성하는 평면을 가져옵니다.
교차 BoundingBox::Intersects 함수(directxcollision.h)는 BoundingSphere와의 교집합에 대해 BoundingBox를 테스트합니다.
교차 BoundingBox가 다른 BoundingBox와 교차하는지 테스트합니다.
교차 BoundingFrustum과의 교집합에 대한 BoundingBox를 테스트합니다.
교차 BoundingOrientedBox와 교집합할 수 있는 BoundingBox를 테스트합니다.
교차 경계 상자에서 평면과의 교집합을 테스트합니다.

교차 경계 상자에서 광선과의 교집합을 테스트합니다.
교차 삼각형과의 교집합에 대해 BoundingBox를 테스트합니다.
교차 BoundingSphere와의 교집합에 대해 BoundingFrustum을 테스트합니다.
교차 BoundingBox와의 교집합에 대해 BoundingFrustum을 테스트합니다.
교차 BoundingFrustum에서 다른 BoundingFrustum과의 교집합을 테스트합니다.
교차 BoundingOrientedBox와의 교집합에 대해 BoundingFrustum을 테스트합니다.
교차 평면과의 교집합에 대해 BoundingFrustum을 테스트합니다.
교차 경계Frustum에서 광선과의 교차를 테스트합니다.
교차 삼각형과의 교차점에 대해 BoundingFrustum을 테스트합니다.
교차 BoundingSphere와의 교집합에 대한 BoundingOrientedBox를 테스트합니다.
교차 BoundingBox와 교집합할 수 있는 BoundingOrientedBox를 테스트합니다.
교차 BoundingFrustum과의 교집합에 대해 BoundingOrientedBox를 테스트합니다.
교차

BoundingOrientedBox와의 교집합에 대해 BoundingOrientedBox를 테스트합니다.
교차 평면과의 교차점에 대한 BoundingOrientedBox를 테스트합니다.
교차 BoundingOrientedBox에서 광선과 교차하는지 테스트합니다.
교차 삼각형과의 교집합에 대한 BoundingOrientedBox를 테스트합니다.
교차 BoundingSphere와 교집합을 위해 BoundingSphere를 테스트합니다.
교차 BoundingBox와의 교집합에 대한 BoundingSphere를 테스트합니다.
교차 BoundingFrustum과의 교집합에 대한 BoundingSphere를 테스트합니다.
교차 BoundingOrientedBox와의 교집합에 대한 BoundingSphere를 테스트합니다.
교차 BoundingSphere에서 평면과의 교집합을 테스트합니다.
교차 경계 구에서 광선과의 교집합을 테스트합니다.
교차 삼각형과의 교집합에 대한 BoundingSphere를 테스트합니다.
교차 삼각형이 광선과 교차하는지 테스트합니다.
교차 두 삼각형이 교차하는지 테스트합니다.

교차 삼각형과 평면이 교차하는지 여부를 테스트합니다.
연산자- XMVECTOR instance 부정을 계산합니다.
연산자- 두 번째 instance XMVECTOR의 한 instance 빼고 결과를 XMVECTOR의 새 instance 반환합니다.
연산자 uint16_t XMU555 instance 구성 요소가 포함된 uint16_t instance 압축된 형식으로 반환합니다.
연산자 uint16_t XMU555instance의 구성 요소를 포함하는 uint16_t instance 압축된 형식으로 반환합니다.
연산자 uint16_t XMUNIBBLE4 instance 구성 요소를 포함하는 uint16_t instance 압축된 형식으로 반환합니다.
연산자 uint16_t XMU555 instance 구성 요소가 포함된 uint16_t instance 압축된 형식으로 반환합니다.
연산자 uint16_t XMU555instance의 구성 요소를 포함하는 uint16_t instance 압축된 형식으로 반환합니다.
연산자 uint16_t XMUNIBBLE4 instance 구성 요소를 포함하는 uint16_t instance 압축된 형식으로 반환합니다.
연산자 uint32_t XMCOLORinstance의 구성 요소가 포함된 uint32_t instance 압축된 형식으로 반환합니다.
연산자 uint32_t uint32_t instance 압축된 벡터 구성 요소 데이터를 XMDEC4의 현재 instance 할당합니다.
연산자 uint32_t XMDECN4instance의 구성 요소를 포함하는 uint32_t instance 압축된 형식으로 반환합니다.
연산자 uint32_t

XMFLOAT3PK instance 구성 요소를 포함하는 uint32_t instance 압축된 형식으로 반환합니다.
연산자 uint32_t XMFLOAT3SE instance 구성 요소를 포함하는 uint32_t instance 압축된 형식으로 반환합니다.
연산자 uint32_t XMUDEC4instance의 구성 요소가 포함된 uint32_t instance 압축된 형식으로 반환합니다.
연산자 uint32_t XMUDECN4instance의 구성 요소를 포함하는 uint32_t instance 압축된 형식으로 반환합니다.
연산자 uint32_t XMXDEC4instance의 구성 요소를 포함하는 uint32_t instance 압축된 형식으로 반환합니다.
연산자 uint32_t XMXDECN4instance의 구성 요소를 포함하는 uint32_t instance 압축된 형식으로 반환합니다.
연산자 uint32_t XMCOLORinstance의 구성 요소가 포함된 uint32_t instance 압축된 형식으로 반환합니다.
연산자 uint32_t uint32_t instance 압축된 벡터 구성 요소 데이터를 XMDEC4의 현재 instance 할당합니다.
연산자 uint32_t XMDECN4instance의 구성 요소를 포함하는 uint32_t instance 압축된 형식으로 반환합니다.
연산자 uint32_t XMFLOAT3PK instance 구성 요소를 포함하는 uint32_t instance 압축된 형식으로 반환합니다.
연산자 uint32_t XMFLOAT3SE instance 구성 요소를 포함하는 uint32_t instance 압축된 형식으로 반환합니다.
연산자 uint32_t XMUDEC4instance의 구성 요소가 포함된 uint32_t instance 압축된 형식으로 반환합니다.
연산자 uint32_t XMUDECN4instance의 구성 요소를 포함하는 uint32_t instance 압축된 형식으로 반환합니다.

연산자 uint32_t XMXDEC4instance의 구성 요소를 포함하는 uint32_t instance 압축된 형식으로 반환합니다.
연산자 uint32_t XMXDECN4instance의 구성 요소를 포함하는 uint32_t instance 압축된 형식으로 반환합니다.
operator() 행 및 열 인수에 지정된 대로 instance XMFLOAT3X3의 행렬 요소에 대한 참조를 반환합니다.
operator() 행 및 열 인수로 지정된 **XMFLOAT3X4**의 행렬 요소 복사본을 반환합니다.
operator() 행 및 열 인수로 지정된 **XMFLOAT3X4**의 행렬 요소에 대한 참조를 반환합니다.
operator() 행 및 열 인수에 지정된 대로 instance XMFLOAT4X3의 행렬 요소에 대한 참조를 반환합니다.
operator() 행 및 열 인수에 지정된 대로 instance XMFLOAT4X4의 행렬 요소에 대한 참조를 반환합니다.
operator() 행 및 열 인수에 지정된 대로 instance XMMATRIX의 행렬 요소에 대한 참조를 반환합니다.
연산자* XMMATRIX의 현재 instance XMMATRIX의 다른 instance 행렬 곱셈을 수행합니다.
연산자* XMVECTOR의 한 instance 두 번째 instance 곱하여 결과를 세 번째 instance 반환합니다.
연산자* XMVECTOR의 instance 부동 소수점 값으로 곱하여 결과를 XMVECTOR의 새 instance 반환합니다.
연산자* 부동 소수점 값을 XMVECTOR의 instance 곱하여 결과를 XMVECTOR의 새 instance 반환합니다.
operator*=

XMMATRIX의 현재 instance XMMATRIX의 다른 instance 행렬 곱셈을 수행하고 업데이트된 현재 instance 대한 참조를 반환합니다.
operator*= 하나의 XMVECTOR instance 두 번째 instance 곱하여 업데이트된 초기 instance 대한 참조를 반환합니다.
operator*= XMVECTOR instance 부동 소수점 값을 곱하고 업데이트된 instance 대한 참조를 반환합니다.
연산자/ XMVECTOR의 한 instance 두 번째 instance 나눠서 결과를 세 번째 instance.o로 반환합니다.
연산자/ XMVECTOR instance 부동 소수점 값으로 나누어 결과를 XMVECTOR의 새 instance 반환합니다.
operator/= 하나의 XMVECTOR instance 두 번째 instance 나누어 업데이트된 초기 instance 대한 참조를 반환합니다.
operator/= XMVECTOR instance 부동 소수점 값으로 나누고 업데이트된 instance 대한 참조를 반환합니다.
operator+ XMVECTOR instance ID 작업을 수행합니다.
operator+ **XMVECTOR**의 두 인스턴스를 추가하여 결과를 새 instance 반환합니다.
operator+= XMVECTOR instance 부동 소수점 값을 추가하고 업데이트된 instance 대한 참조를 반환합니다.
operator= XMFLOAT2의 한 instance 벡터 구성 요소 데이터를 XMFLOAT2의 현재 instance 할당합니다.
operator= XMFLOAT3의 한 instance 벡터 구성 요소 데이터를 XMFLOAT3의 현재 instance 할당합니다.

operator= XMFLOAT3X3의 한 instance 벡터 구성 요소 데이터를 XMFLOAT3X3의 현재 instance 할당합니다.
operator= **XMFLOAT3X4**에 대한 복사 할당 연산자입니다. 인수의 벡터 구성 요소 데이터를 **XMFLOAT3X4**의 현재 instance 복사합니다.
operator= **XMFLOAT3X4**에 대한 할당 연산자 이동 인수의 벡터 구성 요소 데이터를 **XMFLOAT3X4**의 현재 instance 이동합니다.
operator= XMFLOAT4의 한 instance 벡터 구성 요소 데이터를 XMFLOAT4의 현재 instance 할당합니다. (오버로드 1/2)
operator= XMFLOAT4의 한 instance 벡터 구성 요소 데이터를 XMFLOAT4의 현재 instance 할당합니다. (오버로드 2/2)
operator= XMFLOAT4X3의 한 instance 벡터 구성 요소 데이터를 XMFLOAT4X3의 현재 instance 할당합니다.
operator= XMFLOAT4X4의 한 instance 벡터 구성 요소 데이터를 XMFLOAT4X4의 현재 instance 할당합니다.
operator= XMINT2의 한 instance 벡터 구성 요소 데이터를 XMINT2의 현재 instance 할당합니다.
operator= XMINT3의 한 instance 벡터 구성 요소 데이터를 XMINT3의 현재 instance 할당합니다.
operator= XMINT4의 한 instance 벡터 구성 요소 데이터를 XMINT4의 현재 instance 할당합니다.
operator= XMMATRIX의 한 instance 행렬 데이터를 XMMATRIX의 현재 instance 할당하고 현재 instance 대한 참조를 반환합니다.

operator= XMUINT2의 한 instance 벡터 구성 요소 데이터를 XMUINT2의 현재 instance 할당합니다.
operator= XMUINT3의 한 instance 벡터 구성 요소 데이터를 XMUINT3의 현재 instance 할당합니다.
operator= XMUINT4의 한 instance 벡터 구성 요소 데이터를 XMUINT4의 현재 instance 할당합니다.
operator= XMBYTE2의 한 instance 벡터 구성 요소 데이터를 XMBYTE2의 현재 instance 할당합니다.
operator= XMBYTE4의 한 instance 벡터 구성 요소 데이터를 XMBYTE4의 현재 instance 할당합니다.
operator= XMBYTEN2의 한 instance 벡터 구성 요소 데이터를 XMBYTEN2의 현재 instance 할당합니다.
operator= XMBYTEN4의 한 instance 벡터 구성 요소 데이터를 XMBYTEN4의 현재 instance 할당합니다.
operator= uint32_t instance 압축된 벡터 구성 요소 데이터를 XMCOLOR의 현재 instance 할당합니다.
operator= 이 연산자는 XMCOLOR의 한 instance 벡터 구성 요소 데이터를 XMCOLOR의 현재 instance 할당합니다.
operator= XMDEC4의 한 instance 벡터 구성 요소 데이터를 XMDEC4의 현재 instance 할당합니다.
operator= 이 연산자는 XMDECN4의 현재 instance uint32_tto instance 압축된 벡터 구성 요소 데이터를 할당합니다.
operator= XMDECN4의 한 instance 벡터 구성 요소 데이터를 XMDECN4의 현재 instance 할당합니다.

operator= 이 연산자는 uint32_t instance 압축된 벡터 구성 요소 데이터를 XMFLOAT3PK의 현재 instance 할당합니다.
operator= XMFLOAT3PK의 한 instance 벡터 구성 요소 데이터를 XMFLOAT3PK의 현재 instance 할당합니다.
operator= 이 연산자는 uint32_t instance 압축된 벡터 구성 요소 데이터를 XMFLOAT3SE의 현재 instance 할당합니다.
operator= XMHALF2의 한 instance 벡터 구성 요소 데이터를 XMHALF2의 현재 instance 할당합니다.
operator= XMHALF4의 한 instance 벡터 구성 요소 데이터를 XMHALF4의 현재 instance 할당합니다.
operator= XMSHORT2의 한 instance 벡터 구성 요소 데이터를 XMSHORT2의 현재 instance 할당합니다.
operator= XMSHORT4의 한 instance 벡터 구성 요소 데이터를 XMSHORT4의 현재 instance 할당합니다.
operator= XMSHORTN2의 한 instance 벡터 구성 요소 데이터를 XMSHORTN2의 현재 instance 할당합니다.
operator= XMSHORTN4의 한 instance 벡터 구성 요소 데이터를 XMSHORTN4의 현재 instance 할당합니다.
operator= uint16_t instance 압축된 벡터 구성 요소 데이터를 XMU555의 현재 instance 할당합니다.
operator= XMU555의 한 instance 벡터 구성 요소 데이터를 XMU555의 현재 instance 할당합니다.
operator= uint16_t instance 압축된 벡터 구성 요소 데이터를 XMU565의 현재 instance 할당합니다.

operator= XMU565의 한 instance 벡터 구성 요소 데이터를 XMU565의 현재 instance 할당합니다.
operator= XMUBYTE2의 한 instance 벡터 구성 요소 데이터를 XMUBYTE2의 현재 instance 할당합니다.
operator= XMUBYTE4의 한 instance 벡터 구성 요소 데이터를 XMUBYTE4의 현재 instance 할당합니다.
operator= XMUBYTEN2의 한 instance 벡터 구성 요소 데이터를 XMUBYTEN2의 현재 instance 할당합니다.
operator= XMUBYTEN4의 한 instance 벡터 구성 요소 데이터를 XMUBYTEN4의 현재 instance 할당합니다.
operator= uint32_t instance 압축된 벡터 구성 요소 데이터를 XMUDEC4의 현재 instance 할당합니다.
operator= uint32_t instance 압축된 벡터 구성 요소 데이터를 XMUDECN4의 현재 instance 할당합니다.
operator= XMUDECN4의 한 instance 벡터 구성 요소 데이터를 XMUDECN4의 현재 instance 할당합니다.
operator= uint16_t instance 압축된 벡터 구성 요소 데이터를 XMUNIBBLE4의 현재 instance 할당합니다.
operator= XMUNIBBLE4의 한 instance 벡터 구성 요소 데이터를 XMUNIBBLE4의 현재 instance 할당합니다.
operator= XMUSHORT2의 한 instance 벡터 구성 요소 데이터를 XMUSHORT2의 현재 instance 할당합니다.
operator= XMUSHORT4의 한 instance 벡터 구성 요소 데이터를 XMUSHORT4의 현재 instance 할당합니다.
operator=

XMUSHORTN2의 한 instance 벡터 구성 요소 데이터를 XMUSHORTN2의 현재 instance 할당합니다.
operator= XMUSHORTN4의 한 instance 벡터 구성 요소 데이터를 XMUSHORTN4의 현재 instance 할당합니다.
operator= uint32_t instance 압축된 벡터 구성 요소 데이터를 XMXDEC4의 현재 instance 할당합니다.
operator= XMXDEC4의 한 instance 벡터 구성 요소 데이터를 XMXDEC4의 현재 instance 할당합니다.
operator= uint32_t instance 압축된 벡터 구성 요소 데이터를 XMXDECN4의 현재 instance 할당합니다.
operator= XMXDECN4의 한 instance 벡터 구성 요소 데이터를 XMXDECN4의 현재 instance 할당합니다.
operator= 다른 BoundingBox에서 값을 복사합니다.
operator= 다른 BoundingFrustum에서 값을 복사합니다. (오버로드 1/2)
operator= 다른 BoundingFrustum에서 값을 복사합니다. (오버로드 2/2)
operator= 다른 BoundingOrientedBox에서 값을 복사합니다. (오버로드 2/2)
operator= 다른 BoundingOrientedBox에서 값을 복사합니다. (오버로드 1/2)
operator= 지정된 BoundingSphere의 값을 사용하여 BoundingSphere를 초기화합니다. (오버로드 2/2)
operator= 지정된 BoundingSphere의 값을 사용하여 BoundingSphere를 초기화합니다. (오버로드 1/2)

operator= XMBYTE2의 한 instance 벡터 구성 요소 데이터를 XMBYTE2의 현재 instance 할당합니다.
operator= XMBYTE4의 한 instance 벡터 구성 요소 데이터를 XMBYTE4의 현재 instance 할당합니다.
operator= XMBYTEN2의 한 instance 벡터 구성 요소 데이터를 XMBYTEN2의 현재 instance 할당합니다.
operator= XMBYTEN4의 한 instance 벡터 구성 요소 데이터를 XMBYTEN4의 현재 instance 할당합니다.
operator= uint32_t instance 압축된 벡터 구성 요소 데이터를 XMCOLOR의 현재 instance 할당합니다.
operator= 이 연산자는 XMCOLOR의 한 instance 벡터 구성 요소 데이터를 XMCOLOR의 현재 instance 할당합니다.
operator= XMDEC4의 한 instance 벡터 구성 요소 데이터를 XMDEC4의 현재 instance 할당합니다.
operator= 이 연산자는 XMDECN4의 현재 instance uint32_tto instance 압축된 벡터 구성 요소 데이터를 할당합니다.
operator= XMDECN4의 한 instance 벡터 구성 요소 데이터를 XMDECN4의 현재 instance 할당합니다.
operator= 이 연산자는 uint32_t instance 압축된 벡터 구성 요소 데이터를 XMFLOAT3PK의 현재 instance 할당합니다.
operator= XMFLOAT3PK의 한 instance 벡터 구성 요소 데이터를 XMFLOAT3PK의 현재 instance 할당합니다.
operator=

이 연산자는 uint32_t instance 압축된 벡터 구성 요소 데이터를 XMFLOAT3SE의 현재 instance 할당합니다.
operator= XMHALF2의 한 instance 벡터 구성 요소 데이터를 XMHALF2의 현재 instance 할당합니다.
operator= XMHALF4의 한 instance 벡터 구성 요소 데이터를 XMHALF4의 현재 instance 할당합니다.
operator= XMSHORT2의 한 instance 벡터 구성 요소 데이터를 XMSHORT2의 현재 instance 할당합니다.
operator= XMSHORT4의 한 instance 벡터 구성 요소 데이터를 XMSHORT4의 현재 instance 할당합니다.
operator= XMSHORTN2의 한 instance 벡터 구성 요소 데이터를 XMSHORTN2의 현재 instance 할당합니다.
operator= XMSHORTN4의 한 instance 벡터 구성 요소 데이터를 XMSHORTN4의 현재 instance 할당합니다.
operator= uint16_t instance 압축된 벡터 구성 요소 데이터를 XMU555의 현재 instance 할당합니다.
operator= XMU555의 한 instance 벡터 구성 요소 데이터를 XMU555의 현재 instance 할당합니다.
operator= uint16_t instance 압축된 벡터 구성 요소 데이터를 XMU565의 현재 instance 할당합니다.
operator= XMU565의 한 instance 벡터 구성 요소 데이터를 XMU565의 현재 instance 할당합니다.
operator= XMUBYTE2의 한 instance 벡터 구성 요소 데이터를 XMUBYTE2의 현재 instance 할당합니다.
operator= XMUBYTE4의 한 instance 벡터 구성 요소 데이터를 XMUBYTE4의 현재 instance 할당합니다.

operator= XMUBYTEN2의 한 instance 벡터 구성 요소 데이터를 XMUBYTEN2의 현재 instance 할당합니다.
operator= XMUBYTEN4의 한 instance 벡터 구성 요소 데이터를 XMUBYTEN4의 현재 instance 할당합니다.
operator= uint32_t instance 압축된 벡터 구성 요소 데이터를 XMUDEC4의 현재 instance 할당합니다.
operator= uint32_t instance 압축된 벡터 구성 요소 데이터를 XMUDECN4의 현재 instance 할당합니다.
operator= XMUDECN4의 한 instance 벡터 구성 요소 데이터를 XMUDECN4의 현재 instance 할당합니다.
operator= uint16_t instance 압축된 벡터 구성 요소 데이터를 XMUNIBBLE4의 현재 instance 할당합니다.
operator= XMUNIBBLE4의 한 instance 벡터 구성 요소 데이터를 XMUNIBBLE4의 현재 instance 할당합니다.
operator= XMUSHORT2의 한 instance 벡터 구성 요소 데이터를 XMUSHORT2의 현재 instance 할당합니다.
operator= XMUSHORT4의 한 instance 벡터 구성 요소 데이터를 XMUSHORT4의 현재 instance 할당합니다.
operator= XMUSHORTN2의 한 instance 벡터 구성 요소 데이터를 XMUSHORTN2의 현재 instance 할당합니다.
operator= XMUSHORTN4의 한 instance 벡터 구성 요소 데이터를 XMUSHORTN4의 현재 instance 할당합니다.
operator= uint32_t instance 압축된 벡터 구성 요소 데이터를 XMXDEC4의 현재 instance 할당합니다.
operator=

XMXDEC4의 한 instance 벡터 구성 요소 데이터를 XMXDEC4의 현재 instance 할당합니다.
operator= uint32_t instance 압축된 벡터 구성 요소 데이터를 XMXDECN4의 현재 instance 할당합니다.
operator= XMXDECN4의 한 instance 벡터 구성 요소 데이터를 XMXDECN4의 현재 instance 할당합니다.
operator-= XMVECTOR의 현재 instance 부동 소수점 값을 빼고 업데이트된 현재 instance 결과를 반환합니다.
변환 BoundingBox를 변환합니다. (BoundingBox::Transform)
변환 BoundingBox를 변환합니다. (오버로드 1/2)
변환 지정된 변환 매트릭스를 통해 BoundingFrustum을 변환합니다.
변환 지정된 배율, 회전 및 변환 벡터를 사용하여 BoundingFrustum을 변환합니다.
변환 BoundingOrientedBox를 변환합니다. (BoundingOrientedBox::Transform)
변환 BoundingOrientedBox를 변환합니다. (오버로드 1/2)
변환 BoundingSphere를 변환합니다.
변환 지정된 배율, 회전 및 변환 벡터를 사용하여 BoundingSphere를 변환합니다.
XMBYTE2 XMBYTE2의 기본 생성자입니다.

XMBYTE2 2개 요소 float 배열 인수에서 XMBYTE2의 새 instance 초기화합니다.
XMBYTE2 2개 요소 int8_t 배열 인수에서 XMBYTE2의 새 instance 초기화합니다.
XMBYTE2 XMBYTE2에 대한 생성자입니다. (오버로드 3/3)
XMBYTE2 두 float 인수에서 XMBYTE2의 새 instance 초기화합니다.
XMBYTE2 XMBYTE2 int8_t 인수에서 XMBYTE2의 새 instance 초기화합니다.
XMBYTE2 XMBYTE2에 대한 생성자입니다. (오버로드 2/3)
XMBYTE2 XMBYTE2에 대한 생성자입니다. (오버로드 1/3)
XMBYTE2 XMBYTE2의 기본 생성자입니다.
XMBYTE2 2개 요소 float 배열 인수에서 XMBYTE2의 새 instance 초기화합니다.
XMBYTE2 2개 요소 int8_t 배열 인수에서 XMBYTE2의 새 instance 초기화합니다.
XMBYTE2 XMBYTE2에 대한 생성자입니다. (오버로드 3/3)
XMBYTE2 두 float 인수에서 XMBYTE2의 새 instance 초기화합니다.
XMBYTE2

XMBYTE2 int8_t 인수에서 XMBYTE2의 새 instance 초기화합니다.
XMBYTE2 XMBYTE2에 대한 생성자입니다. (오버로드 2/3)
XMBYTE2 XMBYTE2에 대한 생성자입니다. (오버로드 1/3)
XMBYTE4 XMBYTE4의 기본 생성자입니다.
XMBYTE4 4개의 요소 float 배열 인수에서 XMBYTE4의 새 instance 초기화합니다.
XMBYTE4 4개 요소 int8_t 배열 인수에서 XMBYTE4의 새 instance 초기화합니다.
XMBYTE4 XMBYTE4의 생성자입니다. (오버로드 2/2)
XMBYTE4 네 개의 float 인수에서 XMBYTE4의 새 instance 초기화합니다.
XMBYTE4 4개의 int8_t 인수에서 XMBYTE4의 새 instance 초기화합니다.
XMBYTE4 압축된 형식의 구성 요소 데이터를 포함하는 uint32_tvariable XMBYTE4의 새 instance 초기화합니다.
XMBYTE4 XMBYTE4의 생성자입니다. (오버로드 1/2)
XMBYTE4 XMBYTE4의 기본 생성자입니다.
XMBYTE4 4개 요소 float 배열 인수에서 XMBYTE4의 새 instance 초기화합니다.

XMBYTE4 4개 요소 int8_t 배열 인수에서 XMBYTE4의 새 instance 초기화합니다.
XMBYTE4 XMBYTE4에 대한 생성자입니다. (오버로드 2/2)
XMBYTE4 네 개의 float 인수에서 XMBYTE4의 새 instance 초기화합니다.
XMBYTE4 4개의 int8_t 인수에서 XMBYTE4의 새 instance 초기화합니다.
XMBYTE4 압축된 형식의 구성 요소 데이터를 포함하는 uint32_tvariable XMBYTE4의 새 instance 초기화합니다.
XMBYTE4 XMBYTE4에 대한 생성자입니다. (오버로드 1/2)
XMBYTEN2 XMBYTEN2의 기본 생성자입니다.
XMBYTEN2 2개 요소 float 배열 인수에서 XMBYTEN2의 새 instance 초기화합니다.
XMBYTEN2 두 요소 int8_t 배열 인수에서 XMBYTEN2의 새 instance 초기화합니다.
XMBYTEN2 XMBYTEN2에 대한 생성자입니다. (오버로드 3/3)
XMBYTEN2 두 float 인수에서 XMBYTEN2의 새 instance 초기화합니다.
XMBYTEN2 두 개의 int8_t 인수에서 XMBYTEN2의 새 instance 초기화합니다.

XMBYTEN2 XMBYTEN2에 대한 생성자입니다. (오버로드 1/3)
XMBYTEN2 XMBYTEN2에 대한 생성자입니다. (오버로드 2/3)
XMBYTEN2 XMBYTEN2의 기본 생성자입니다.
XMBYTEN2 2개 요소 float 배열 인수에서 XMBYTEN2의 새 instance 초기화합니다.
XMBYTEN2 두 요소 int8_t 배열 인수에서 XMBYTEN2의 새 instance 초기화합니다.
XMBYTEN2 XMBYTEN2에 대한 생성자입니다. (오버로드 3/3)
XMBYTEN2 두 float 인수에서 XMBYTEN2의 새 instance 초기화합니다.
XMBYTEN2 두 개의 int8_t 인수에서 XMBYTEN2의 새 instance 초기화합니다.
XMBYTEN2 XMBYTEN2에 대한 생성자입니다. (오버로드 1/3)
XMBYTEN2 XMBYTEN2에 대한 생성자입니다. (오버로드 2/3)
XMBYTEN4 XMBYTEN4의 기본 생성자입니다.
XMBYTEN4 4개의 요소 float 배열 인수에서 XMBYTEN4의 새 instance 초기화합니다.
XMBYTEN4

4개 요소 int8_t 배열 인수에서 XMBYTEN4의 새 instance 초기화합니다.
XMBYTEN4 XMBYTEN4에 대한 생성자입니다. (오버로드 2/2)
XMBYTEN4 네 개의 float 인수에서 XMBYTEN4의 새 instance 초기화합니다.
XMBYTEN4 4개의 int8_t 인수에서 XMBYTEN4의 새 instance 초기화합니다.
XMBYTEN4 압축된 형식의 구성 요소 데이터를 포함하는 uint32_tvariable XMBYTEN4의 새 instance 초기화합니다.
XMBYTEN4 XMBYTEN4에 대한 생성자입니다. (오버로드 1/2)
XMBYTEN4 XMBYTEN4의 기본 생성자입니다.
XMBYTEN4 4개의 요소 float 배열 인수에서 XMBYTEN4의 새 instance 초기화합니다.
XMBYTEN4 4개 요소 int8_t 배열 인수에서 XMBYTEN4의 새 instance 초기화합니다.
XMBYTEN4 XMBYTEN4에 대한 생성자입니다. (오버로드 2/2)
XMBYTEN4 네 개의 float 인수에서 XMBYTEN4의 새 instance 초기화합니다.
XMBYTEN4 4개의 int8_t 인수에서 XMBYTEN4의 새 instance 초기화합니다.
XMBYTEN4

압축된 형식의 구성 요소 데이터를 포함하는 uint32_tvariable XMBYTEN4의 새 instance 초기화합니다.
XMBYTEN4 XMBYTEN4에 대한 생성자입니다. (오버로드 1/2)
XMCOLOR XMCOLOR의 기본 생성자입니다.
XMCOLOR 4개의 요소 float 배열 인수에서 XMCOLOR의 새 instance 초기화합니다.
XMCOLOR XMCOLOR에 대한 생성자입니다. (오버로드 2/2)
XMCOLOR 네 개의 float 인수에서 XMCOLOR의 새 instance 초기화합니다.
XMCOLOR 압축된 형식의 구성 요소 데이터를 포함하는 uint32_t 변수에서 XMCOLOR의 새 instance 초기화합니다.
XMCOLOR XMCOLOR에 대한 생성자입니다. (오버로드 1/2)
XMCOLOR XMCOLOR의 기본 생성자입니다.
XMCOLOR 4개의 요소 float 배열 인수에서 XMCOLOR의 새 instance 초기화합니다.
XMCOLOR XMCOLOR에 대한 생성자입니다. (오버로드 2/2)
XMCOLOR 네 개의 float 인수에서 XMCOLOR의 새 instance 초기화합니다.
XMCOLOR

압축된 형식의 구성 요소 데이터를 포함하는 uint32_t 변수에서 XMCOLOR의 새 instance 초기화합니다.
XMCOLOR XMCOLOR에 대한 생성자입니다. (오버로드 1/2)
XMColorAdjustContrast 색의 대비 값을 조정합니다.
XMColorAdjustSaturation 색의 채도 값을 조정합니다.
XMColorEqual 두 색의 같음을 테스트합니다.
XMColorGreater 첫 번째 색의 모든 구성 요소가 두 번째 색의 해당 구성 요소보다 큰지 테스트합니다.
XMColorGreaterOrEqual 첫 번째 색의 모든 구성 요소가 두 번째 색의 해당 구성 요소보다 크거나 같은지 테스트합니다.
XMColorHSLToRGB HSL 색 값을 RGB 색 값으로 변환합니다.
XMColorHSVToRGB HSV 색 값을 RGB 색 값으로 변환합니다.
XMColorIsInfinite 색의 구성 요소가 양수 또는 음의 무한대인지 여부를 테스트합니다.
XMColorIsNaN 색의 구성 요소가 숫자(NaN)가 아닌지 여부를 테스트합니다.
XMColorLess 첫 번째 색의 모든 구성 요소가 두 번째 색의 해당 구성 요소보다 작은지 테스트합니다.
XMColorLessOrEqual 첫 번째 색의 모든 구성 요소가 두 번째 색의 해당 구성 요소보다 작거나 같은지 테스트합니다.

XMColorModulate 해당 구성 요소를 함께 곱하여 두 색을 혼합합니다.
XMColorNegative 색의 음수 RGB 색 값을 결정합니다.
XMColorNotEqual 두 색이 같지 않은지 여부를 테스트합니다.
XMColorrGBToHSL RGB 색 값을 HSL 색 값으로 변환합니다.
XMColorrGBToHSV RGB 색 값을 HSV 색 값으로 변환합니다.
XMColorRGBToSRGB RGB 색 벡터를 sRGB로 변환합니다.
XMColorrGBToXYZ RGB 색 값을 XYZ 색 값으로 변환합니다.
XMColorrGBToYUV RGB 색 값을 YUV 색 값으로 변환합니다.
XMColorRGBToYUV_HD RGB 색 값을 YUV HD 색 값으로 변환합니다.
XMColorRGBToYUV_UHD RGB 색 값을 YUV UHD 색 값으로 변환합니다.
XMColorSRGBToRGB sRGB 색 벡터를 RGB로 변환합니다.
XMColorSRGBToXYZ SRGB 색 값을 XYZ 색 값으로 변환합니다.
XMColorXYZToRGB

XYZ 색 값을 RGB 색 값으로 변환합니다.
XMColorXYZToSRGB XYZ 색 값을 SRGB 색 값으로 변환합니다.
XMColorYUVToRGB YUV 색 값을 RGB 색 값으로 변환합니다.
XMColorYUVToRGB_HD YUV 색 값을 RGB HD 색 값으로 변환합니다.
XMColorYUVToRGB_UHD YUV 색 값을 RGB UHD 색 값으로 변환합니다.
XMComparisonAllFalse 비교 값을 테스트하여 비교된 모든 구성 요소가 false인지 확인합니다.
XMComparisonAllInBounds 비교 값을 테스트하여 비교된 모든 구성 요소가 설정된 범위 내에 있는지 확인합니다.
XMComparisonAllTrue 비교 값을 테스트하여 비교된 모든 구성 요소가 true인지 확인합니다.
XMComparisonAnyFalse 비교 값을 테스트하여 비교된 구성 요소가 false인지 확인합니다.
XMComparisonAnyOutOfBounds 비교 값을 테스트하여 비교된 구성 요소가 설정된 범위를 벗어나는지 확인합니다.
XMComparisonAnyTrue 비교 값을 테스트하여 비교된 구성 요소가 true인지 확인합니다.
XMComparisonMixed 비교 값을 테스트하여 비교된 구성 요소에 혼합된 결과가 있는지 확인합니다(일부 true 및 일부 false).
XMConvertFloatToHalf 단정밀도 부동 소수점 값을 반정밀도 부동 소수점 값으로 변환합니다.

XMConvertFloatToHalf 단정밀도 부동 소수점 값을 반정밀도 부동 소수점 값으로 변환합니다.
XMConvertFloatToHalfStream 단정밀도 부동 소수점 값의 스트림을 반정밀도 부동 소수점 값의 스트림으로 변환합니다.
XMConvertFloatToHalfStream 단정밀도 부동 소수점 값의 스트림을 반정밀도 부동 소수점 값의 스트림으로 변환합니다.
XMConvertHalfToFloat 반정밀도 부동 소수점 값을 단정밀도 부동 소수점 값으로 변환합니다.
XMConvertHalfToFloat 반정밀도 부동 소수점 값을 단정밀도 부동 소수점 값으로 변환합니다.
XMConvertHalfToFloatStream 반정밀도 부동 소수점 값의 스트림을 단정밀도 부동 소수점 값의 스트림으로 변환합니다.
XMConvertHalfToFloatStream 반정밀도 부동 소수점 값의 스트림을 단정밀도 부동 소수점 값의 스트림으로 변환합니다.
XMConvertToDegrees 라디안으로 측정된 각도를 도 단위로 측정된 각도로 변환합니다.
XMConvertToRadians 도 단위로 측정된 각도를 라디안 단위로 측정된 각도로 변환합니다.
XMConvertVectorFloatToInt float 구성 요소가 있는 XMVECTOR를 int32_t 구성 요소가 있는 XMVECTOR로 변환하고 균일한 바이어스를 적용합니다.
XMConvertVectorFloatToUInt float 구성 요소가 있는 XMVECTOR를 uint32_t 구성 요소가 있는 XMVECTOR로 변환하고 균일한 바이어스를 적용합니다.
XMConvertVectorIntToFloat

int32_t 구성 요소가 있는 XMVECTOR를 float 구성 요소가 있는 XMVECTOR로 변환하고 균일한 바이어스를 적용합니다.
XMConvertVectorUIntToFloat uint32_t 구성 요소가 있는 XMVECTOR를 float 구성 요소가 있는 XMVECTOR로 변환하고 균일한 바이어스를 적용합니다.
XMDEC4 XMDEC4의 기본 생성자입니다.
XMDEC4 4개의 요소 float 배열 인수에서 XMDEC4의 새 instance 초기화합니다.
XMDEC4 XMDEC4의 생성자입니다. (오버로드 1/2)
XMDEC4 네 개의 float 인수에서 XMDEC4의 새 instance 초기화합니다.
XMDEC4 압축된 형식의 구성 요소 데이터를 포함하는 uint32_t 변수에서 XMDEC4의 새 instance 초기화합니다.
XMDEC4 XMDEC4의 생성자입니다. (오버로드 2/2)
XMDEC4 XMDEC4의 기본 생성자입니다.
XMDEC4 4개의 요소 float 배열 인수에서 XMDEC4의 새 instance 초기화합니다.
XMDEC4 XMDEC4의 생성자입니다. (오버로드 1/2)
XMDEC4 네 개의 float 인수에서 XMDEC4의 새 instance 초기화합니다.

XMDEC4 압축된 형식의 구성 요소 데이터를 포함하는 uint32_t 변수에서 XMDEC4의 새 instance 초기화합니다.
XMDEC4 XMDEC4의 생성자입니다. (오버로드 2/2)
XMDECN4 XMDECN4의 기본 생성자입니다.
XMDECN4 4개 요소 float 배열 인수에서 XMDECN4의 새 instance 초기화합니다.
XMDECN4 XMDECN4에 대한 생성자입니다. (오버로드 2/2)
XMDECN4 정규화된 4개의 부동 소수점에서 XMDECN4의 새 instance 초기화합니다.
XMDECN4 압축된 형식의 구성 요소 데이터를 포함하는 uint32_t 변수에서 XMDECN4의 새 instance 초기화합니다.
XMDECN4 XMDECN4에 대한 생성자입니다. (오버로드 1/2)
XMDECN4 XMDECN4의 기본 생성자입니다.
XMDECN4 4개 요소 float 배열 인수에서 XMDECN4의 새 instance 초기화합니다.
XMDECN4 XMDECN4에 대한 생성자입니다. (오버로드 2/2)
XMDECN4 정규화된 4개의 부동 소수점에서 XMDECN4의 새 instance 초기화합니다.

XMDECN4 압축된 형식의 구성 요소 데이터를 포함하는 uint32_t 변수에서 XMDECN4의 새 instance 초기화합니다.
XMDECN4 XMDECN4에 대한 생성자입니다. (오버로드 1/2)
XMFLOAT2 XMFLOAT2의 기본 생성자입니다.
XMFLOAT2 두 요소 float 배열 인수에서 XMFLOAT2의 새 instance 초기화합니다.
XMFLOAT2 XMFLOAT2에 대한 생성자입니다. (오버로드 1/2)
XMFLOAT2 두 float 인수에서 XMFLOAT2의 새 instance 초기화합니다.
XMFLOAT2 XMFLOAT2에 대한 생성자입니다. (오버로드 2/2)
XMFLOAT3 XMFLOAT3의 기본 생성자입니다.
XMFLOAT3 XMFLOAT3의 생성자입니다. (오버로드 2/3)
XMFLOAT3 XMFLOAT3의 생성자입니다. (오버로드 3/3)
XMFLOAT3 세 개의 float 인수에서 XMFLOAT3의 새 instance 초기화합니다.
XMFLOAT3 XMFLOAT3의 생성자입니다. (오버로드 1/3)

XMFLOAT3PK XMFLOAT3PK의 기본 생성자입니다.
XMFLOAT3PK 3개 요소 float 배열 인수에서 XMFLOAT3PK의 새 instance 초기화합니다.
XMFLOAT3PK XMFLOAT3PK에 대한 생성자입니다.
XMFLOAT3PK 세 개의 float 인수에서 XMFLOAT3PK의 새 instance 초기화합니다.
XMFLOAT3PK 압축된 형식의 구성 요소 데이터를 포함하는 uint32_tvariable XMFLOAT3PK의 새 instance 초기화합니다.
XMFLOAT3PK XMFLOAT3SE의 한 instance 벡터 구성 요소 데이터를 XMFLOAT3SE의 현재 instance 할당합니다.
XMFLOAT3PK XMFLOAT3PK의 기본 생성자입니다.
XMFLOAT3PK 세 요소 float 배열 인수에서 XMFLOAT3PK의 새 instance 초기화합니다.
XMFLOAT3PK XMFLOAT3PK에 대한 생성자입니다.
XMFLOAT3PK 세 개의 float 인수에서 XMFLOAT3PK의 새 instance 초기화합니다.
XMFLOAT3PK 압축된 형식의 구성 요소 데이터를 포함하는 uint32_tvariable XMFLOAT3PK의 새 instance 초기화합니다.
XMFLOAT3PK XMFLOAT3SE의 한 instance 벡터 구성 요소 데이터를 XMFLOAT3SE의 현재 instance 할당합니다.

XMFLOAT3SE XMFLOAT3SE의 기본 생성자입니다.
XMFLOAT3SE 3개의 요소 float 배열 인수에서 XMFLOAT3SE의 새 instance 초기화합니다.
XMFLOAT3SE XMFLOAT3SE에 대한 생성자입니다. (오버로드 1/2)
XMFLOAT3SE 세 개의 float 인수에서 XMFLOAT3SE의 새 instance 초기화합니다.
XMFLOAT3SE 압축된 형식의 구성 요소 데이터를 포함하는 uint32_tvariable XMFLOAT3SE의 새 instance 초기화합니다.
XMFLOAT3SE XMFLOAT3SE에 대한 생성자입니다. (오버로드 2/2)
XMFLOAT3SE XMFLOAT3SE의 기본 생성자입니다.
XMFLOAT3SE 3개의 요소 float 배열 인수에서 XMFLOAT3SE의 새 instance 초기화합니다.
XMFLOAT3SE XMFLOAT3SE에 대한 생성자입니다. (오버로드 1/2)
XMFLOAT3SE 세 개의 float 인수에서 XMFLOAT3SE의 새 instance 초기화합니다.
XMFLOAT3SE 압축된 형식의 구성 요소 데이터를 포함하는 uint32_tvariable XMFLOAT3SE의 새 instance 초기화합니다.
XMFLOAT3SE XMFLOAT3SE에 대한 생성자입니다. (오버로드 2/2)

XMFLOAT3X3 XMFLOAT3X3의 기본 생성자입니다. (XMFLOAT3X3. XMFLOAT3X3)
XMFLOAT3X3 9개 요소 float 배열에서 XMFLOAT3X3 구조체의 새 instance 초기화합니다.
XMFLOAT3X3 XMFLOAT3X3의 기본 생성자입니다. (오버로드 2/3)
XMFLOAT3X3 9개의 스칼라 부동 소수점 값에서 XMFLOAT3X3 구조체의 새 instance 초기화합니다.
XMFLOAT3X3 XMFLOAT3X3의 기본 생성자입니다. (오버로드 1/3)
XMFLOAT3X4 **XMFLOAT3X4**의 기본 생성자입니다.
XMFLOAT3X4 float 값의 12개 요소 배열에서 **XMFLOAT3X4** 를 생성합니다.
XMFLOAT3X4 **XMFLOAT3X4**에 대한 복사 생성자입니다.
XMFLOAT3X4 12개의 스칼라 float 값에서 **XMFLOAT3X4** 를 생성합니다.
XMFLOAT3X4 **XMFLOAT3X4**에 대한 이동 생성자입니다.
XMFLOAT4 XMFLOAT4의 기본 생성자입니다.
XMFLOAT4 XMFLOAT4에 대한 생성자입니다. (오버로드 3/3)
XMFLOAT4

XMFLOAT4에 대한 생성자입니다. (오버로드 2/3)
XMFLOAT4 네 개의 float 인수에서 XMFLOAT4의 새 instance 초기화합니다.
XMFLOAT4 XMFLOAT4에 대한 생성자입니다. (오버로드 1/3)
XMFLOAT4X3 XMFLOAT4X3의 기본 생성자입니다. (XMFLOAT4X3. XMFLOAT4X3)
XMFLOAT4X3 12개 요소 float 배열에서 XMFLOAT4X3 구조체의 새 instance 초기화합니다.
XMFLOAT4X3 XMFLOAT4X3의 기본 생성자입니다. (오버로드 1/3)
XMFLOAT4X3 12개의 스칼라 float 값에서 XMFLOAT4X3 구조체의 새 instance 초기화합니다.
XMFLOAT4X3 XMFLOAT4X3의 기본 생성자입니다. (오버로드 2/3)
XMFLOAT4X4 XMFLOAT4X4의 기본 생성자입니다. (XMFLOAT4X4. XMFLOAT4X4)
XMFLOAT4X4 16개 요소 float 배열에서 XMFLOAT4X4 구조체의 새 instance 초기화합니다.
XMFLOAT4X4 XMFLOAT4X4의 기본 생성자입니다. (오버로드 1/3)
XMFLOAT4X4 16개의 스칼라 부동 소수점 값에서 XMFLOAT4X4 구조체의 새 instance 초기화합니다.
XMFLOAT4X4 XMFLOAT4X4의 기본 생성자입니다. (오버로드 2/3)

XMFresnelTerm 극명화되지 않은 광원에 대한 프레넬 용어를 계산합니다.
XMHALF2 XMHALF2의 기본 생성자입니다.
XMHALF2 두 요소 float 배열 인수에서 XMHALF2의 새 instance 초기화합니다.
XMHALF2 두 요소 HALF 배열 인수에서 XMHALF2의 새 instance 초기화합니다.
XMHALF2 XMHALF2에 대한 생성자입니다. (오버로드 2/3)
XMHALF2 두 float 인수에서 XMHALF2의 새 instance 초기화합니다.
XMHALF2 두 개의 HALF 인수에서 XMHALF2의 새 instance 초기화합니다.
XMHALF2 XMHALF2에 대한 생성자입니다. (오버로드 3/3)
XMHALF2 XMHALF2에 대한 생성자입니다. (오버로드 1/3)
XMHALF2 XMHALF2의 기본 생성자입니다.
XMHALF2 두 요소 float 배열 인수에서 XMHALF2의 새 instance 초기화합니다.
XMHALF2 두 요소 HALF 배열 인수에서 XMHALF2의 새 instance 초기화합니다.
XMHALF2

XMHALF2에 대한 생성자입니다. (오버로드 2/3)
XMHALF2 두 float 인수에서 XMHALF2의 새 instance 초기화합니다.
XMHALF2 두 개의 HALF 인수에서 XMHALF2의 새 instance 초기화합니다.
XMHALF2 XMHALF2에 대한 생성자입니다. (오버로드 3/3)
XMHALF2 XMHALF2에 대한 생성자입니다. (오버로드 1/3)
XMHALF4 XMHALF4의 기본 생성자입니다. (XMHALF4. XMHALF4)
XMHALF4 4개의 요소 float 배열 인수에서 XMHALF4의 새 instance 초기화합니다.
XMHALF4 4개 요소 HALF 배열 인수에서 XMHALF4의 새 instance 초기화합니다.
XMHALF4 XMHALF4의 기본 생성자입니다. (오버로드 2/3)
XMHALF4 네 개의 float 인수에서 XMHALF4의 새 instance 초기화합니다.
XMHALF4 네 개의 HALF 인수에서 XMHALF4의 새 instance 초기화합니다.
XMHALF4 XMHALF4에 대한 생성자입니다.
XMHALF4 XMHALF4의 기본 생성자입니다. (오버로드 1/3)

XMHALF4 XMHALF4의 기본 생성자입니다. (XMHALF4. XMHALF4)
XMHALF4 4개의 요소 float 배열 인수에서 XMHALF4의 새 instance 초기화합니다.
XMHALF4 4개 요소 HALF 배열 인수에서 XMHALF4의 새 instance 초기화합니다.
XMHALF4 XMHALF4의 기본 생성자입니다. (오버로드 2/3)
XMHALF4 네 개의 float 인수에서 XMHALF4의 새 instance 초기화합니다.
XMHALF4 네 개의 HALF 인수에서 XMHALF4의 새 instance 초기화합니다.
XMHALF4 XMHALF4에 대한 생성자입니다.
XMHALF4 XMHALF4의 기본 생성자입니다. (오버로드 1/3)
XMINT2 XMINT2의 기본 생성자입니다.
XMINT2 배열 인수를 int32_t2 두 요소에서 XMINT2의 새 instance 초기화합니다.
XMINT2 XMINT2에 대한 생성자입니다. (오버로드 2/2)
XMINT2 두 개의 int32_t 인수에서 XMINT2의 새 instance 초기화합니다.
XMINT2

XMINT2에 대한 생성자입니다. (오버로드 1/2)
XMINT3 XMINT3의 기본 생성자입니다.
XMINT3 배열 인수를 int32_t 세 요소에서 XMINT3의 새 instance 초기화합니다.
XMINT3 XMINT3의 생성자입니다. (오버로드 1/2)
XMINT3 세 개의 int32_t 인수에서 XMINT3의 새 instance 초기화합니다.
XMINT3 XMINT3의 생성자입니다. (오버로드 2/2)
XMINT4 XMINT4의 기본 생성자입니다.
XMINT4 4개 요소 int32_t 배열 인수에서 XMINT4의 새 instance 초기화합니다.
XMINT4 XMINT4의 생성자입니다. (오버로드 1/2)
XMINT4 4개의 int32_t 인수에서 XMINT4의 새 instance 초기화합니다.
XMINT4 XMINT4의 생성자입니다. (오버로드 2/2)
XMLoadByte2 XMBYTE2를 XMVECTOR에 로드합니다.
XMLoadByte2 XMBYTE2를 XMVECTOR에 로드합니다.

XMLoadByte4 XMBYTE4를 XMVECTOR에 로드합니다.
XMLoadByte4 XMBYTE4를 XMVECTOR에 로드합니다.
XMLoadByteN2 XMBYTEN2를 XMVECTOR에 로드합니다.
XMLoadByteN2 XMBYTEN2를 XMVECTOR에 로드합니다.
XMLoadByteN4 XMBYTEN4를 XMVECTOR에 로드합니다.
XMLoadByteN4 XMBYTEN4를 XMVECTOR에 로드합니다.
XMLoadColor XMCOLOR를 XMVECTOR에 로드합니다.
XMLoadColor XMCOLOR를 XMVECTOR에 로드합니다.
XMLoadDec4 XMDEC4를 XMVECTOR에 로드합니다.
XMLoadDec4 XMDEC4를 XMVECTOR에 로드합니다.
XMLoadDecN4 XMDECN4를 XMVECTOR에 로드합니다.
XMLoadDecN4 XMDECN4를 XMVECTOR에 로드합니다.
XMLoadFloat

부동 소수점 스칼라 값을 XMVECTOR에 로드합니다.
XMLoadFloat2 XMFLOAT2를 XMVECTOR에 로드합니다.
XMLoadFloat2A XMFLOAT2A를 XMVECTOR에 로드합니다.
XMLoadFloat3 XMFLOAT3을 XMVECTOR에 로드합니다.
XMLoadFloat3A XMFLOAT3A를 XMVECTOR에 로드합니다.
XMLoadFloat3PK XMFLOAT3PK를 XMVECTOR에 로드합니다.
XMLoadFloat3PK XMFLOAT3PK를 XMVECTOR에 로드합니다.
XMLoadFloat3SE XMFLOAT3SE를 XMVECTOR에 로드합니다.
XMLoadFloat3SE XMFLOAT3SE를 XMVECTOR에 로드합니다.
XMLoadFloat3x3 XMFLOAT3X3을 XMMATRIX에 로드합니다.
XMLoadFloat3x4 **XMFLOAT3X4**를 **XMMATRIX**에 로드합니다.
XMLoadFloat3x4A **XMFLOAT3X4A**를 **XMMATRIX**에 로드합니다.
XMLoadFloat4 XMFLOAT4를 XMVECTOR에 로드합니다.

XMLoadFloat4A XMFLOAT4A를 XMVECTOR에 로드합니다.
XMLoadFloat4x3 XMFLOAT4X3을 XMMATRIX에 로드합니다.
XMLoadFloat4x3A XMFLOAT4X3A를 XMVECTOR에 로드합니다.
XMLoadFloat4x4 XMFLOAT4X4를 XMMATRIX에 로드합니다.
XMLoadFloat4x4A XMFLOAT4X4A를 XMVECTOR에 로드합니다.
XMLoadHalf2 XMHALF2를 XMVECTOR에 로드합니다.
XMLoadHalf2 XMHALF2를 XMVECTOR에 로드합니다.
XMLoadHalf4 XMHALF4를 XMVECTOR에 로드합니다.
XMLoadHalf4 XMHALF4를 XMVECTOR에 로드합니다.
XMLoadInt 스칼라 값을 XMVECTOR에 로드합니다.
XMLoadInt2 XMVECTOR의 x 및 y 구성 요소에 데이터를 로드합니다.
XMLoadInt2A XMVECTOR의 x 및 y 구성 요소에 16 바이트 정렬 데이터를 로드합니다.
XMLoadInt3

형식 검사 없이 XMVECTOR의 x, y 및 z 구성 요소에 데이터를 로드합니다.
XMLoadInt3A 형식 검사 없이 XMVECTOR의 x, y 및 z 구성 요소에 16바이트의 정렬된 데이터를 로드합니다.
XMLoadInt4 형식 검사 없이 XMVECTOR에 데이터를 로드합니다.
XMLoadInt4A 형식 검사 없이 16비트 정렬 데이터를 XMVECTOR에 로드합니다.
XMLoadShort2 XMSHORT2를 XMVECTOR에 로드합니다.
XMLoadShort2 XMSHORT2를 XMVECTOR에 로드합니다.
XMLoadShort4 XMSHORT4를 XMVECTOR에 로드합니다.
XMLoadShort4 XMSHORT4를 XMVECTOR에 로드합니다.
XMLoadShortN2 XMSHORTN2를 XMVECTOR에 로드합니다.
XMLoadShortN2 XMSHORTN2를 XMVECTOR에 로드합니다.
XMLoadShortN4 XMSHORTN4를 XMVECTOR에 로드합니다.
XMLoadShortN4 XMSHORTN4를 XMVECTOR에 로드합니다.
XMLoadSInt2 서명된 정수 데이터를 XMVECTOR의 x 및 y 구성 요소에 로드합니다.

XMLoadSInt3 XMVECTOR의 x, y 및 z 구성 요소에 부호 있는 정수 데이터를 로드합니다.
XMLoadSInt4 서명된 정수 데이터를 XMVECTOR에 로드합니다.
XMLoadU555 XMU555를 XMVECTOR에 로드합니다.
XMLoadU555 XMU555를 XMVECTOR에 로드합니다.
XMLoadU565 XMU565를 XMVECTOR에 로드합니다.
XMLoadU565 XMU565를 XMVECTOR에 로드합니다.
XMLoadUByte2 XMVECTOR에 XMUBYTE2를 로드합니다.
XMLoadUByte2 XMVECTOR에 XMUBYTE2를 로드합니다.
XMLoadUByte4 XMVECTOR에 XMUBYTE4를 로드합니다.
XMLoadUByte4 XMVECTOR에 XMUBYTE4를 로드합니다.
XMLoadUByteN2 XMVECTOR에 XMUBYTEN2를 로드합니다.
XMLoadUByteN2 XMVECTOR에 XMUBYTEN2를 로드합니다.
XMLoadUByteN4

XMVECTOR에 XMUBYTEN4를 로드합니다.
XMLoadUByteN4 XMVECTOR에 XMUBYTEN4를 로드합니다.
XMLoadUDec4 XMUDEC4를 XMVECTOR에 로드합니다.
XMLoadUDec4 XMUDEC4를 XMVECTOR에 로드합니다.
XMLoadUDecN4 XMUDECN4를 XMVECTOR에 로드합니다.
XMLoadUDecN4 XMUDECN4를 XMVECTOR에 로드합니다.
XMLoadUDecN4_XR 확장된 범위 XMUDECN4를 XMVECTOR에 로드합니다.
XMLoadUDecN4_XR 확장된 범위 XMUDECN4를 XMVECTOR에 로드합니다.
XMLoadUInt2 부호 없는 정수 데이터를 XMVECTOR의 x 및 y 구성 요소에 로드합니다.
XMLoadUInt3 형식 검사 없이 부호 없는 정수 데이터를 XMVECTOR의 x, y 및 z 구성 요소에 로드합니다.
XMLoadUInt4 부호 없는 정수 데이터를 XMVECTOR에 로드합니다.
XMLoadUNibble4 XMVECTOR에 XMUNIBBLE4를 로드합니다.
XMLoadUNibble4 XMVECTOR에 XMUNIBBLE4를 로드합니다.

XMLoadUShort2 XMUSHORT2를 XMVECTOR에 로드합니다.
XMLoadUShort2 XMUSHORT2를 XMVECTOR에 로드합니다.
XMLoadUShort4 XMUSHORT4를 XMVECTOR에 로드합니다.
XMLoadUShort4 XMUSHORT4를 XMVECTOR에 로드합니다.
XMLoadUShortN2 XMUSHORTN2를 XMVECTOR에 로드합니다.
XMLoadUShortN2 XMUSHORTN2를 XMVECTOR에 로드합니다.
XMLoadUShortN4 XMVECTOR에 XMUSHORTN4를 로드합니다.
XMLoadUShortN4 XMVECTOR에 XMUSHORTN4를 로드합니다.
XMLoadXDec4 XMXDEC4를 XMVECTOR에 로드합니다.
XMLoadXDec4 XMXDEC4를 XMVECTOR에 로드합니다.
XMLoadXDecN4 XMXDECN4를 XMVECTOR에 로드합니다.
XMLoadXDecN4 XMXDECN4를 XMVECTOR에 로드합니다.
XMMATRIX

XMMATRIX의 기본 생성자입니다.
XMMATRIX 16개의 요소 float 배열에서 XMMATRIX 구조체의 새 instance 초기화합니다.
XMMATRIX XMMATRIX에 대한 생성자입니다. (오버로드 2/2)
XMMATRIX 16개의 스칼라 부동 소수점 값에서 XMMATRIX 구조체의 새 instance 초기화합니다.
XMMATRIX XMVECTOR의 4개 인스턴스에서 XMMATRIX 구조체의 새 instance 초기화합니다.
XMMATRIX XMMATRIX에 대한 생성자입니다. (오버로드 1/2)
XMMatrixAffineTransformation 아핀 변환 매트릭스를 빌드합니다.
XMMatrixAffineTransformation2D xy 평면에서 2D 아핀 변환 매트릭스를 빌드합니다.
XMMatrixDecompose 일반 3D 변환 매트릭스를 스칼라, 회전 및 변환 구성 요소로 구분합니다.
XMMatrixDeterminant 행렬의 결정자를 계산합니다.
XMMatrixIdentity ID 매트릭스를 빌드합니다.
XMMatrixInverse 행렬의 역방향을 계산합니다.
XMMatrixIsIdentity 행렬이 ID 행렬인지 여부를 테스트합니다.

XMMatrixIsInfinite 행렬의 요소가 양수 또는 음의 무한대인지 테스트합니다.
XMMatrixIsNaN 행렬의 요소가 NaN인지 여부를 테스트합니다.
XMMatrixLookAtLH 카메라 위치, 위쪽 방향 및 초점을 사용하여 왼손 좌표계용 보기 행렬을 작성합니다.
XMMatrixLookAtRH 카메라 위치, 위쪽 방향 및 초점을 사용하여 오른손 좌표계용 보기 행렬을 작성합니다.
XMMatrixLookToLH 카메라 위치, 위쪽 방향 및 카메라 방향을 사용하여 왼손 좌표계용 보기 행렬을 작성합니다.
XMMatrixLookToRH 카메라 위치, 위쪽 방향 및 카메라 방향을 사용하여 오른손 좌표계용 보기 행렬을 작성합니다.
XMMatrixMultiply 두 행렬의 곱을 계산합니다.
XMMatrixMultiplyTranspose 두 행렬의 곱의 전이를 계산합니다.
XMMatrixOrthographicLH 왼손 좌표계용 직교 투영 행렬을 작성합니다.
XMMatrixOrthographicOffCenterLH 왼손 좌표계용 사용자 지정 직교 투영 행렬을 작성합니다.
XMMatrixOrthographicOffCenterRH 오른손 좌표계용 사용자 지정 직교 투영 행렬을 작성합니다.
XMMatrixOrthographicRH 오른손 좌표계용 직교 투영 행렬을 작성합니다.
XMMatrixPerspectiveFovLH

보기의 필드를 기준으로 왼손 원근 투영 행렬을 작성합니다.
XMMatrixPerspectiveFovRH 보기의 필드를 기준으로 오른손 원근 투영 행렬을 작성합니다.
XMMatrixPerspectiveLH 왼손 원근 투영 행렬을 작성합니다.
XMMatrixPerspectiveOffCenterLH 왼손 원근 투영 행렬의 사용자 지정 버전을 작성합니다.
XMMatrixPerspectiveOffCenterRH 오른손 원근 투영 행렬의 사용자 지정 버전을 작성합니다.
XMMatrixPerspectiveRH 오른손 원근 투영 행렬을 작성합니다.
XMMatrixReflect 지정된 평면을 통해 벡터를 반영하도록 설계된 변환 매트릭스를 빌드합니다.
XMMatrixRotationAxis 임의의 축을 중심으로 회전하는 행렬을 빌드합니다.
XMMatrixRotationNormal 임의의 일반 벡터를 중심으로 회전하는 행렬을 빌드합니다.
XMMatrixRotationQuaternion 쿼터니언에서 회전 행렬을 작성합니다.
XMMatrixRotationRollPitchYaw 지정된 피치, 요, 롤(오일러 각도)을 기반으로 회전 행렬을 작성합니다.
XMMatrixRotationRollPitchYawFromVector Euler 각도(피치, 요 및 롤)를 포함하는 벡터를 기반으로 회전 행렬을 빌드합니다.
XMMatrixRotationX x축을 중심으로 회전하는 행렬을 빌드합니다.

XMMatrixRotationY y축을 중심으로 회전하는 행렬을 빌드합니다.
XMMatrixRotationZ z축을 중심으로 회전하는 행렬을 빌드합니다.
XMMatrixScaling x축, y축 및 z축을 따라 크기가 조정되는 행렬을 빌드합니다.
XMMatrixScalingFromVector 3D 벡터에서 크기 조정 매트릭스를 빌드합니다.
XMMatrixSet float 값을 사용하여 행렬을 만듭니다.
XMMatrixShadow 기하 도형을 평면으로 평면화하는 변환 행렬을 빌드합니다.
XMMatrixTransformation 변환 매트릭스를 빌드합니다.
XMMatrixTransformation2D xy 평면에서 2D 변환 매트릭스를 빌드합니다.
XMMatrixTranslation 지정된 오프셋에서 변환 행렬을 작성합니다.
XMMatrixTranslationFromVector 벡터에서 변환 행렬을 빌드합니다.
XMMatrixTranspose 행렬의 전이를 계산합니다.
XMMatrixVectorTensorProduct 두 벡터에서 행렬을 빌드합니다.
XMPlaneDot

입력 평면과 4D 벡터 사이의 점 곱을 계산합니다.
XMPlaneDotCoord 입력 평면과 3D 벡터 사이의 점 곱을 계산합니다.
XMPlaneDotNormal 평면의 일반 벡터와 3D 벡터 사이의 점 곱을 계산합니다.
XMPlaneEqual 두 평면이 같은지 여부를 확인합니다.
XMPlaneFromPointNormal 평면의 지점과 일반 벡터에서 생성된 평면의 수식을 계산합니다.
XMPlaneFromPoints 평면의 세 지점에서 생성된 평면의 수식을 계산합니다.
XMPlaneIntersectLine 평면과 선 간의 교차를 찾습니다.
XMPlaneIntersectPlane 두 평면의 교집합을 찾습니다.
XMPlaneIsInfinite 평면의 계수가 양수인지 음의 무한대인지 테스트합니다.
XMPlaneIsNaN 평면 계수가 NaN인지 여부를 테스트합니다.
XMPlaneNearEqual 두 평면이 거의 같은지 여부를 결정합니다.
XMPlaneNormalize x, y 및 z 계수가 단위 일반 벡터를 형성하도록 평면의 계수를 정규화합니다.
XMPlaneNormalizeEst x, y 및 z 계수가 단위 일반 벡터를 형성할 수 있도록 평면의 계수를 추정합니다.

XMPlaneNotEqual 두 평면이 같지 않은지 확인합니다.
XMPlaneTransform 지정된 행렬로 평면을 변환합니다.
XMPlaneTransformStream 지정된 행렬로 평면 스트림을 변환합니다.
XMQuaternionBaryCentric 지정된 쿼터니언을 사용하여 바리센트릭 좌표의 점을 반환합니다. (XMQuaternionBaryCentric)
XMQuaternionBaryCentricV 지정된 쿼터니언을 사용하여 바리센트릭 좌표의 점을 반환합니다. (XMQuaternionBaryCentricV)
XMQuaternionConjugate 쿼터니언의 결합을 계산합니다.
XMQuaternionDot 두 개의 쿼터니언의 점 곱을 계산합니다.
XMQuaternionEqual 두 쿼터니언이 같은지 테스트합니다.
XMQuaternionExp 지정된 순수 쿼터니언의 지수를 계산합니다.
XMQuaternionIdentity ID 쿼터니언을 반환합니다.
XMQuaternionInverse 쿼터니언의 역을 계산합니다.
XMQuaternionIsIdentity 특정 쿼터니언이 ID 쿼터니언인지 테스트합니다.
XMQuaternionIsInfinite

쿼터니언의 구성 요소가 양수 또는 음의 무한대인지 테스트합니다.
XMQuaternionIsNaN 쿼터니언의 구성 요소가 NaN인지 테스트합니다.
XMQuaternionLength 쿼터니언의 크기를 계산합니다.
XMQuaternionLengthSq 쿼터니언 크기의 제곱을 계산합니다.
XMQuaternionLn 지정된 단위 쿼터니언의 자연 로그를 계산합니다.
XMQuaternionMultiply 두 개의 쿼터니언의 곱을 계산합니다.
XMQuaternionNormalize 쿼터니언을 정규화합니다.
XMQuaternionNormalizeEst 정규화된 쿼터니언 버전을 예측합니다.
XMQuaternionNotEqual 두 쿼터니언이 같지 않은지 테스트합니다.
XMQuaternionReciprocalLength 쿼터니언 크기의 상호를 계산합니다.
XMQuaternionRotationAxis 축에 대한 회전 쿼터니언을 계산합니다.
XMQuaternionRotationMatrix 회전 행렬에서 회전 쿼터니언을 계산합니다.
XMQuaternionRotationNormal 일반 벡터에 대한 회전 쿼터니언을 계산합니다.

XMQuaternionRotationRollPitchYaw 피치, 요, 롤(오일러 각도)에 따라 회전 쿼터니언을 계산합니다.
XMQuaternionRotationRollPitchYawFromVector Euler 각도(피치, 요 및 롤)를 포함하는 벡터를 기반으로 회전 쿼터니언을 계산합니다.
XMQuaternionSlerp 구형 선형 보간을 사용하여 두 단위 쿼터니언 사이를 보간합니다. (XMQuaternionSlerp)
XMQuaternionSlerpV 구형 선형 보간을 사용하여 두 단위 쿼터니언 사이를 보간합니다. (XMQuaternionSlerpV)
XMQuaternionSquad 구형 사분면 보간을 사용하여 4개의 단위 쿼터니언 사이를 보간합니다. (XMQuaternionSquad)
XMQuaternionSquadSetup 구형 사분면 보간을 위한 설정 제어 지점의 주소를 제공합니다.
XMQuaternionSquadV 구형 사분면 보간을 사용하여 4개의 단위 쿼터니언 사이를 보간합니다. (XMQuaternionSquadV)
XMQuaternionToAxisAngle 지정된 쿼터니언에 대해 해당 축에 대한 축 및 회전 각도를 계산합니다.
XMScalarACos 부동 소수점 숫자의 아크코사인을 계산합니다.
XMScalarACosEst 부동 소수점 숫자의 아크코사인을 추정합니다.
XMScalarASin 부동 소수점 숫자의 아크사인을 계산합니다.
XMScalarASinEst 부동 소수점 숫자의 아크사인을 추정합니다.
XMScalarCos

라디안 각도의 코사인을 계산합니다.
XMScalarCosEst 라디안 각도의 코사인을 추정합니다.
XMScalarModAngle -XM_PI 및 XM_PI 사이의 각도를 계산합니다.
XMScalarNearEqual 두 부동 소수점 값이 거의 같은지 확인합니다.
XMScalarSin 라디안 각도의 사인을 계산합니다.
XMScalarSinCos 라디안 각도의 사인과 코사인을 모두 계산합니다.
XMScalarSinCosEst 라디안 각도의 사인과 코사인을 모두 추정합니다.
XMScalarSinEst 라디안 각도의 사인을 추정합니다.
XMSHORT2 XMSHORT2의 기본 생성자입니다.
XMSHORT2 두 요소 float 배열 인수에서 XMSHORT2의 새 instance 초기화합니다.
XMSHORT2 배열 인수를 int16_t 두 요소에서 XMSHORT2의 새 instance 초기화합니다.
XMSHORT2 XMSHORT2에 대한 생성자입니다. (오버로드 1/3)
XMSHORT2 두 float 인수에서 XMSHORT2의 새 instance 초기화합니다.

XMSHORT2 두 개의 int16_t 인수에서 XMSHORT2의 새 instance 초기화합니다.
XMSHORT2 XMSHORT2에 대한 생성자입니다. (오버로드 3/3)
XMSHORT2 XMSHORT2에 대한 생성자입니다. (오버로드 2/3)
XMSHORT2 XMSHORT2의 기본 생성자입니다.
XMSHORT2 두 요소 float 배열 인수에서 XMSHORT2의 새 instance 초기화합니다.
XMSHORT2 배열 인수를 int16_t 두 요소에서 XMSHORT2의 새 instance 초기화합니다.
XMSHORT2 XMSHORT2에 대한 생성자입니다. (오버로드 1/3)
XMSHORT2 두 float 인수에서 XMSHORT2의 새 instance 초기화합니다.
XMSHORT2 두 개의 int16_t 인수에서 XMSHORT2의 새 instance 초기화합니다.
XMSHORT2 XMSHORT2에 대한 생성자입니다. (오버로드 3/3)
XMSHORT2 XMSHORT2에 대한 생성자입니다. (오버로드 2/3)
XMSHORT4 XMSHORT4의 기본 생성자입니다.
XMSHORT4

4개 요소 float 배열 인수에서 XMSHORT4의 새 instance 초기화합니다.
XMSHORT4 4개 요소 int16_t 배열 인수에서 XMSHORT4의 새 instance 초기화합니다.
XMSHORT4 XMSHORT4의 생성자입니다. (오버로드 3/3)
XMSHORT4 네 개의 float 인수에서 XMSHORT4의 새 instance 초기화합니다.
XMSHORT4 4개의 int16_t 인수에서 XMSHORT4의 새 instance 초기화합니다.
XMSHORT4 XMSHORT4의 생성자입니다. (오버로드 1/3)
XMSHORT4 XMSHORT4의 생성자입니다. (오버로드 2/3)
XMSHORT4 XMSHORT4의 기본 생성자입니다.
XMSHORT4 4개 요소 float 배열 인수에서 XMSHORT4의 새 instance 초기화합니다.
XMSHORT4 4개 요소 int16_t 배열 인수에서 XMSHORT4의 새 instance 초기화합니다.
XMSHORT4 XMSHORT4의 생성자입니다. (오버로드 3/3)
XMSHORT4 네 개의 float 인수에서 XMSHORT4의 새 instance 초기화합니다.
XMSHORT4 4개의 int16_t 인수에서 XMSHORT4의 새 instance 초기화합니다.

XMSHORT4 XMSHORT4의 생성자입니다. (오버로드 1/3)
XMSHORT4 XMSHORT4의 생성자입니다. (오버로드 2/3)
XMSHORTN2 XMSHORTN2의 기본 생성자입니다.
XMSHORTN2 두 요소 float 배열 인수에서 XMSHORTN2의 새 instance 초기화합니다.
XMSHORTN2 두 요소 int16_t 배열 인수에서 XMSHORTN2의 새 instance 초기화합니다.
XMSHORTN2 XMSHORTN2의 생성자입니다. (오버로드 3/3)
XMSHORTN2 정규화된 두 부동 소수점에서 XMSHORTN2의 새 instance 초기화합니다.
XMSHORTN2 두 개의 int16_t 인수에서 XMSHORTN2의 새 instance 초기화합니다.
XMSHORTN2 XMSHORTN2의 생성자입니다. (오버로드 1/3)
XMSHORTN2 XMSHORTN2의 생성자입니다. (오버로드 2/3)
XMSHORTN2 XMSHORTN2의 기본 생성자입니다.
XMSHORTN2 두 요소 float 배열 인수에서 XMSHORTN2의 새 instance 초기화합니다.
XMSHORTN2

두 요소 int16_t 배열 인수에서 XMSHORTN2의 새 instance 초기화합니다.
XMSHORTN2 XMSHORTN2의 생성자입니다. (오버로드 3/3)
XMSHORTN2 정규화된 두 부동 소수점에서 XMSHORTN2의 새 instance 초기화합니다.
XMSHORTN2 두 개의 int16_t 인수에서 XMSHORTN2의 새 instance 초기화합니다.
XMSHORTN2 XMSHORTN2의 생성자입니다. (오버로드 1/3)
XMSHORTN2 XMSHORTN2의 생성자입니다. (오버로드 2/3)
XMSHORTN4 XMSHORTN4의 기본 생성자입니다.
XMSHORTN4 4개 요소 float 배열 인수에서 XMSHORTN4의 새 instance 초기화합니다.
XMSHORTN4 4개 요소 int16_t 배열 인수에서 XMSHORTN4의 새 instance 초기화합니다.
XMSHORTN4 XMSHORTN4에 대한 생성자입니다. (오버로드 3/3)
XMSHORTN4 정규화된 4개의 부동 소수점에서 XMSHORTN4의 새 instance 초기화합니다.
XMSHORTN4 4개의 int16_t 인수에서 XMSHORTN4의 새 instance 초기화합니다.
XMSHORTN4 XMSHORTN4에 대한 생성자입니다. (오버로드 2/3)

XMSHORTN4 XMSHORTN4에 대한 생성자입니다. (오버로드 1/3)
XMSHORTN4 XMSHORTN4의 기본 생성자입니다.
XMSHORTN4 4개 요소 float 배열 인수에서 XMSHORTN4의 새 instance 초기화합니다.
XMSHORTN4 4개 요소 int16_t 배열 인수에서 XMSHORTN4의 새 instance 초기화합니다.
XMSHORTN4 XMSHORTN4에 대한 생성자입니다. (오버로드 3/3)
XMSHORTN4 정규화된 4개의 부동 소수점에서 XMSHORTN4의 새 instance 초기화합니다.
XMSHORTN4 4개의 int16_t 인수에서 XMSHORTN4의 새 instance 초기화합니다.
XMSHORTN4 XMSHORTN4에 대한 생성자입니다. (오버로드 2/3)
XMSHORTN4 XMSHORTN4에 대한 생성자입니다. (오버로드 1/3)
XMStoreByte2 XMBYTE2에 XMVECTOR를 저장합니다.
XMStoreByte2 XMBYTE2에 XMVECTOR를 저장합니다.
XMStoreByte4 XMBYTE4에 XMVECTOR를 저장합니다.
XMStoreByte4

XMBYTE4에 XMVECTOR를 저장합니다.
XMStoreByteN2 XMBYTEN2에 XMVECTOR를 저장합니다.
XMStoreByteN2 XMBYTEN2에 XMVECTOR를 저장합니다.
XMStoreByteN4 XMBYTEN4에 XMVECTOR를 저장합니다.
XMStoreByteN4 XMBYTEN4에 XMVECTOR를 저장합니다.
XMStoreColor XMCOLOR에 XMVECTOR를 저장합니다.
XMStoreColor XMCOLOR에 XMVECTOR를 저장합니다.
XMStoreDec4 XMDEC4에 XMVECTOR를 저장합니다.
XMStoreDec4 XMDEC4에 XMVECTOR를 저장합니다.
XMStoreDecN4 XMDECN4에 XMVECTOR를 저장합니다.
XMStoreDecN4 XMDECN4에 XMVECTOR를 저장합니다.
XMStoreFloat XMVECTOR를 float에 저장합니다.
XMStoreFloat2 XMFLOAT2에 XMVECTOR를 저장합니다.

XMStoreFloat2A XMFLOAT2A에 XMVECTOR를 저장합니다.
XMStoreFloat3 XMFLOAT3에 XMVECTOR를 저장합니다.
XMStoreFloat3A XMFLOAT3A에 XMVECTOR를 저장합니다.
XMStoreFloat3PK XMFLOAT3PK에 XMVECTOR를 저장합니다.
XMStoreFloat3PK XMFLOAT3PK에 XMVECTOR를 저장합니다.
XMStoreFloat3SE XMFLOAT3SE에 XMVECTOR를 저장합니다.
XMStoreFloat3SE XMFLOAT3SE에 XMVECTOR를 저장합니다.
XMStoreFloat3x3 XMFLOAT3X3에 XMMATRIX를 저장합니다.
XMStoreFloat3x4 **XMFLOAT3X4**에 **XMMATRIX**를 저장합니다.
XMStoreFloat3x4A **XMFLOAT3X4A에 XMMATRIX**를 저장합니다.
XMStoreFloat4 XMFLOAT4에 XMVECTOR를 저장합니다.
XMStoreFloat4A XMFLOAT4A에 XMVECTOR를 저장합니다.
XMStoreFloat4x3

XMFLOAT4X3에 XMMATRIX를 저장합니다.
XMStoreFloat4x3A XMFLOAT4X3A에 XMVECTOR를 저장합니다.
XMStoreFloat4x4 XMFLOAT4X4에 XMMATRIX를 저장합니다.
XMStoreFloat4x4A XMFLOAT4X4A에 XMVECTOR를 저장합니다.
XMStoreHalf2 XMHALF2에 XMVECTOR를 저장합니다.
XMStoreHalf2 XMHALF2에 XMVECTOR를 저장합니다.
XMStoreHalf4 XMHALF4에 XMVECTOR를 저장합니다.
XMStoreHalf4 XMHALF4에 XMVECTOR를 저장합니다.
XMStoreInt XMVECTOR를 uint32_t 저장합니다.
XMStoreInt2 XMVECTOR를 2개 요소 uint32_t 배열에 저장합니다.
XMStoreInt2A XMVECTOR를 16 바이트 정렬된 2개 요소 uint32_t 배열에 저장합니다.
XMStoreInt3 XMVECTOR를 3개 요소 uint32_t 배열에 저장합니다.
XMStoreInt3A XMVECTOR를 16 바이트 정렬된 3개 요소 uint32_t 배열에 저장합니다.

XMStoreInt4 XMVECTOR를 4개 요소 uint32_t 배열에 저장합니다.
XMStoreInt4A XMVECTOR를 16 바이트 정렬된 4개 요소 uint32_t 배열에 저장합니다.
XMStoreShort2 XMSHORT2에 XMVECTOR를 저장합니다.
XMStoreShort2 XMSHORT2에 XMVECTOR를 저장합니다.
XMStoreShort4 XMSHORT4에 XMVECTOR를 저장합니다.
XMStoreShort4 XMSHORT4에 XMVECTOR를 저장합니다.
XMStoreShortN2 XMSHORTN2에 XMVECTOR를 저장합니다.
XMStoreShortN2 XMSHORTN2에 XMVECTOR를 저장합니다.
XMStoreShortN4 XMSHORTN4에 XMVECTOR를 저장합니다.
XMStoreShortN4 XMSHORTN4에 XMVECTOR를 저장합니다.
XMStoreSInt2 XMVECTOR의 부록 정수 데이터를 XMINT2 구조체에 저장합니다.
XMStoreSInt3 XMVECTOR의 부록 정수 데이터를 XMINT3 구조체에 저장합니다.
XMStoreSInt4

XMVECTOR의 부록된 정수 데이터를 XMINT4 구조체에 저장합니다.
XMStoreU555 XMU555에 XMVECTOR를 저장합니다.
XMStoreU555 XMU555에 XMVECTOR를 저장합니다.
XMStoreU565 XMU565에 XMVECTOR를 저장합니다.
XMStoreU565 XMU565에 XMVECTOR를 저장합니다.
XMStoreUByte2 XMVECTOR를 XMUBYTE2에 저장합니다.
XMStoreUByte2 XMVECTOR를 XMUBYTE2에 저장합니다.
XMStoreUByte4 XMUBYTE4에 XMVECTOR를 저장합니다.
XMStoreUByte4 XMUBYTE4에 XMVECTOR를 저장합니다.
XMStoreUByteN2 XMUBYTEN2에 XMVECTOR를 저장합니다.
XMStoreUByteN2 XMUBYTEN2에 XMVECTOR를 저장합니다.
XMStoreUByteN4 XMUBYTEN4에 XMVECTOR를 저장합니다.
XMStoreUByteN4 XMUBYTEN4에 XMVECTOR를 저장합니다.

XMStoreUDec4 XMUDEC4에 XMVECTOR를 저장합니다.
XMStoreUDec4 XMUDEC4에 XMVECTOR를 저장합니다.
XMStoreUDecN4 XMUDECN4에 XMVECTOR를 저장합니다.
XMStoreUDecN4 XMUDECN4에 XMVECTOR를 저장합니다.
XMStoreUDecN4_XR 확장된 범위 XMUDECN4를 XMVECTOR에 저장합니다.
XMStoreUDecN4_XR 확장된 범위 XMUDECN4를 XMVECTOR에 저장합니다.
XMStoreUInt2 XMVECTOR의 부호 없는 정수 데이터를 XMUINT2 구조체에 저장합니다.
XMStoreUInt3 XMVECTOR의 부호 없는 정수 데이터를 XMUINT3 구조체에 저장합니다.
XMStoreUInt4 XMVECTOR의 부호 없는 정수 데이터를 XMUINT4 구조체에 저장합니다.
XMStoreUNibble4 XMVECTOR를 XMUNIBBLE4에 저장합니다.
XMStoreUNibble4 XMVECTOR를 XMUNIBBLE4에 저장합니다.
XMStoreUShort2 XMUSHORT2에 XMVECTOR를 저장합니다.
XMStoreUShort2

XMUSHORT2에 XMVECTOR를 저장합니다.
XMStoreUShort4 XMUSHORT4에 XMVECTOR를 저장합니다.
XMStoreUShort4 XMUSHORT4에 XMVECTOR를 저장합니다.
XMStoreUShortN2 XMUSHORTN2에 XMVECTOR를 저장합니다.
XMStoreUShortN2 XMUSHORTN2에 XMVECTOR를 저장합니다.
XMStoreUShortN4 XMUSHORTN4에 XMVECTOR를 저장합니다.
XMStoreUShortN4 XMUSHORTN4에 XMVECTOR를 저장합니다.
XMStoreXDec4 XMXDEC4에 XMVECTOR를 저장합니다.
XMStoreXDec4 XMXDEC4에 XMVECTOR를 저장합니다.
XMStoreXDecN4 XMXDECN4에 XMVECTOR를 저장합니다.
XMStoreXDecN4 XMXDECN4에 XMVECTOR를 저장합니다.
XMU555 XMU555의 기본 생성자입니다.
XMU555 3개의 요소 float 배열과 1개의 bool 인수에서 XMU555의 새 instance 초기화합니다.

XMU555 세 요소 int8_t 배열과 하나의 bool 인수에서 XMU555의 새 instance 초기화합니다.
XMU555 XMU555의 생성자입니다. (오버로드 1/2)
XMU555 부동 소수점 인수 3개와 부울 인수 1개에서 XMU555의 새 instance 초기화합니다.
XMU555 압축된 형식의 구성 요소 데이터를 포함하는 uint16_t 변수에서 XMU555의 새 instance 초기화합니다.
XMU555 세 개의 int8_t 및 하나의 bool 인수에서 XMU555의 새 instance 초기화합니다.
XMU555 XMU555의 생성자입니다. (오버로드 2/2)
XMU555 XMU555의 기본 생성자입니다.
XMU555 3개의 요소 float 배열과 1개의 bool 인수에서 XMU555의 새 instance 초기화합니다.
XMU555 세 요소 int8_t 배열과 하나의 bool 인수에서 XMU555의 새 instance 초기화합니다.
XMU555 XMU555의 생성자입니다. (오버로드 1/2)
XMU555 부동 소수점 인수 3개와 부울 인수 1개에서 XMU555의 새 instance 초기화합니다.
XMU555 압축된 형식의 구성 요소 데이터를 포함하는 uint16_t 변수에서 XMU555의 새 instance 초기화합니다.

XMU555 세 개의 int8_t 및 하나의 bool 인수에서 XMU555의 새 instance 초기화합니다.
XMU555 XMU555의 생성자입니다. (오버로드 2/2)
XMU565 XMU565의 기본 생성자입니다.
XMU565 3개 요소 float 배열에서 XMU565의 새 instance 초기화합니다.
XMU565 세 요소 int8_t 배열에서 XMU565의 새 instance 초기화합니다.
XMU565 XMU565에 대한 생성자입니다. (오버로드 2/2)
XMU565 세 개의 float 인수에서 XMU565의 새 instance 초기화합니다.
XMU565 압축된 형식의 구성 요소 데이터를 포함하는 uint16_t 변수에서 XMU565의 새 instance 초기화합니다.
XMU565 세 개의 int8_t 인수에서 XMU565의 새 instance 초기화합니다.
XMU565 XMU565에 대한 생성자입니다. (오버로드 1/2)
XMU565 XMU565의 기본 생성자입니다.
XMU565 3개 요소 float 배열에서 XMU565의 새 instance 초기화합니다.

XMU565 세 요소 int8_t 배열에서 XMU565의 새 instance 초기화합니다.
XMU565 XMU565에 대한 생성자입니다. (오버로드 2/2)
XMU565 세 개의 float 인수에서 XMU565의 새 instance 초기화합니다.
XMU565 압축된 형식의 구성 요소 데이터를 포함하는 uint16_t 변수에서 XMU565의 새 instance 초기화합니다.
XMU565 세 개의 int8_t 인수에서 XMU565의 새 instance 초기화합니다.
XMU565 XMU565에 대한 생성자입니다. (오버로드 1/2)
XMUBYTE2 XMUBYTE2의 기본 생성자입니다.
XMUBYTE2 2개 요소 float 배열 인수에서 XMUBYTE2의 새 instance 초기화합니다.
XMUBYTE2 2개 요소 int8_t 배열 인수에서 **XMUBYTE2**의 새 instance 초기화합니다.
XMUBYTE2 XMUBYTE2에 대한 생성자입니다. (오버로드 2/3)
XMUBYTE2 두 float 인수에서 XMUBYTE2의 새 instance 초기화합니다.
XMUBYTE2 XMUBYTE2에 대한 생성자입니다. (오버로드 3/3)

XMUBYTE2 두 개의 int8_t 인수에서 XMUBYTE2의 새 instance 초기화합니다.
XMUBYTE2 XMUBYTE2에 대한 생성자입니다. (오버로드 1/3)
XMUBYTE2 XMUBYTE2의 기본 생성자입니다.
XMUBYTE2 2개 요소 float 배열 인수에서 XMUBYTE2의 새 instance 초기화합니다.
XMUBYTE2 2개 요소 int8_t 배열 인수에서 **XMUBYTE2**의 새 instance 초기화합니다.
XMUBYTE2 XMUBYTE2에 대한 생성자입니다. (오버로드 2/3)
XMUBYTE2 두 float 인수에서 XMUBYTE2의 새 instance 초기화합니다.
XMUBYTE2 XMUBYTE2에 대한 생성자입니다. (오버로드 3/3)
XMUBYTE2 두 개의 int8_t 인수에서 XMUBYTE2의 새 instance 초기화합니다.
XMUBYTE2 XMUBYTE2에 대한 생성자입니다. (오버로드 1/3)
XMUBYTE4 XMUBYTE4의 기본 생성자입니다. (XMUBYTE4. XMUBYTE4)
XMUBYTE4 4개 요소 int8_t 배열 인수에서 XMUBYTE4의 새 instance 초기화합니다.
XMUBYTE4

4개 요소 float 배열 인수에서 XMUBYTE4의 새 instance 초기화합니다.
XMUBYTE4 XMUBYTE4의 기본 생성자입니다. (오버로드 2/3)
XMUBYTE4 네 개의 float 인수에서 XMUBYTE4의 새 instance 초기화합니다.
XMUBYTE4 압축된 형식의 구성 요소 데이터를 포함하는 Uuint32_t 변수에서 <XMUBYTE4의 새 instance 초기화합니다.
XMUBYTE4 4개의 int8_t 인수에서 XMUBYTE4의 새 instance 초기화합니다.
XMUBYTE4 XMUBYTE4의 기본 생성자입니다. (오버로드 1/3)
XMUBYTE4 XMUBYTE4의 기본 생성자입니다. (XMUBYTE4. XMUBYTE4)
XMUBYTE4 4개 요소 int8_t 배열 인수에서 XMUBYTE4의 새 instance 초기화합니다.
XMUBYTE4 4개 요소 float 배열 인수에서 XMUBYTE4의 새 instance 초기화합니다.
XMUBYTE4 XMUBYTE4의 기본 생성자입니다. (오버로드 2/3)
XMUBYTE4 네 개의 float 인수에서 XMUBYTE4의 새 instance 초기화합니다.
XMUBYTE4 압축된 형식의 구성 요소 데이터를 포함하는 Uuint32_t 변수에서 <XMUBYTE4의 새 instance 초기화합니다.
XMUBYTE4

4개의 int8_t 인수에서 XMUBYTE4의 새 instance 초기화합니다.
XMUBYTE4 XMUBYTE4의 기본 생성자입니다. (오버로드 1/3)
XMUBYTEN2 XMUBYTEN2의 기본 생성자입니다. (XMUBYTEN2. XMUBYTEN2)
XMUBYTEN2 2개 요소 float 배열 인수에서 XMUBYTEN2의 새 instance 초기화합니다.
XMUBYTEN2 XMUBYTEN2의 기본 생성자입니다. (오버로드 1/2)
XMUBYTEN2 XMUBYTEN2에 대한 생성자입니다. (오버로드 2/3)
XMUBYTEN2 두 float 인수에서 XMUBYTEN2의 새 instance 초기화합니다.
XMUBYTEN2 XMUBYTEN2에 대한 생성자입니다. (오버로드 1/3)
XMUBYTEN2 두 개의 uint8_t 인수에서 XMUBYTEN2의 새 instance 초기화합니다.
XMUBYTEN2 XMUBYTEN2에 대한 생성자입니다. (오버로드 3/3)
XMUBYTEN2 XMUBYTEN2의 기본 생성자입니다. (XMUBYTEN2. XMUBYTEN2)
XMUBYTEN2 2개 요소 float 배열 인수에서 XMUBYTEN2의 새 instance 초기화합니다.
XMUBYTEN2 XMUBYTEN2의 기본 생성자입니다. (오버로드 1/2)

XMUBYTEN2 XMUBYTEN2에 대한 생성자입니다. (오버로드 2/3)
XMUBYTEN2 두 float 인수에서 XMUBYTEN2의 새 instance 초기화합니다.
XMUBYTEN2 XMUBYTEN2에 대한 생성자입니다. (오버로드 1/3)
XMUBYTEN2 두 개의 uint8_t 인수에서 XMUBYTEN2의 새 instance 초기화합니다.
XMUBYTEN2 XMUBYTEN2에 대한 생성자입니다. (오버로드 3/3)
XMUBYTEN4 XMUBYTEN4의 기본 생성자입니다. (XMUBYTEN4. XMUBYTEN4)
XMUBYTEN4 4개의 요소 float 배열 인수에서 XMUBYTEN4의 새 instance 초기화합니다.
XMUBYTEN4 4개 요소 uint8_t 배열 인수에서 XMUBYTEN4의 새 instance 초기화합니다.
XMUBYTEN4 XMUBYTEN4에 대한 생성자입니다.
XMUBYTEN4 네 개의 float 인수에서 XMUBYTEN4의 새 instance 초기화합니다.
XMUBYTEN4 압축된 형식의 구성 요소 데이터를 포함하는 uint32_tvariable XMUBYTEN4의 새 instance 초기화합니다.
XMUBYTEN4 4개의 uint8_t 인수에서 XMUBYTEN4의 새 instance 초기화합니다.

XMUBYTEN4 XMUBYTEN4의 기본 생성자입니다. (오버로드 1/2)
XMUBYTEN4 XMUBYTEN4의 기본 생성자입니다. (XMUBYTEN4. XMUBYTEN4)
XMUBYTEN4 4개의 요소 float 배열 인수에서 XMUBYTEN4의 새 instance 초기화합니다.
XMUBYTEN4 4개 요소 uint8_t 배열 인수에서 XMUBYTEN4의 새 instance 초기화합니다.
XMUBYTEN4 XMUBYTEN4에 대한 생성자입니다.
XMUBYTEN4 네 개의 float 인수에서 XMUBYTEN4의 새 instance 초기화합니다.
XMUBYTEN4 압축된 형식의 구성 요소 데이터를 포함하는 uint32_tvariable XMUBYTEN4의 새 instance 초기화합니다.
XMUBYTEN4 4개의 uint8_t 인수에서 XMUBYTEN4의 새 instance 초기화합니다.
XMUBYTEN4 XMUBYTEN4의 기본 생성자입니다. (오버로드 1/2)
XMUDEC4 XMUDEC4의 기본 생성자입니다.
XMUDEC4 4개의 요소 float 배열 인수에서 XMUDEC4의 새 instance 초기화합니다.
XMUDEC4 XMUDEC4에 대한 생성자입니다. (오버로드 1/2)

XMUDEC4 네 개의 float 인수에서 XMUDEC4의 새 instance 초기화합니다.
XMUDEC4 압축된 형식의 구성 요소 데이터를 포함하는 uint32_t 변수에서 XMUDEC4의 새 instance 초기화합니다.
XMUDEC4 XMUDEC4에 대한 생성자입니다. (오버로드 2/2)
XMUDEC4 XMUDEC4의 기본 생성자입니다.
XMUDEC4 4개의 요소 float 배열 인수에서 XMUDEC4의 새 instance 초기화합니다.
XMUDEC4 XMUDEC4에 대한 생성자입니다. (오버로드 1/2)
XMUDEC4 네 개의 float 인수에서 XMUDEC4의 새 instance 초기화합니다.
XMUDEC4 압축된 형식의 구성 요소 데이터를 포함하는 uint32_t 변수에서 XMUDEC4의 새 instance 초기화합니다.
XMUDEC4 XMUDEC4에 대한 생성자입니다. (오버로드 2/2)
XMUDECN4 XMUDECN4의 기본 생성자입니다.
XMUDECN4 4개의 요소 float 배열 인수에서 XMUDECN4의 새 instance 초기화합니다.
XMUDECN4 XMUDECN4에 대한 생성자입니다. (오버로드 2/2)

XMUDECN4 이 생성자는 정규화된 네 개의 float 인수에서 XMUDECN4의 새 instance 초기화합니다.
XMUDECN4 압축된 형식의 구성 요소 데이터를 포함하는 uint32_t 변수에서 XMUDECN4의 새 instance 초기화합니다.
XMUDECN4 XMUDECN4에 대한 생성자입니다. (오버로드 1/2)
XMUDECN4 XMUDECN4의 기본 생성자입니다.
XMUDECN4 4개의 요소 float 배열 인수에서 XMUDECN4의 새 instance 초기화합니다.
XMUDECN4 XMUDECN4에 대한 생성자입니다. (오버로드 2/2)
XMUDECN4 이 생성자는 정규화된 네 개의 float 인수에서 XMUDECN4의 새 instance 초기화합니다.
XMUDECN4 압축된 형식의 구성 요소 데이터를 포함하는 uint32_t 변수에서 XMUDECN4의 새 instance 초기화합니다.
XMUDECN4 XMUDECN4에 대한 생성자입니다. (오버로드 1/2)
XMUINT2 XMUINT2의 기본 생성자입니다.
XMUINT2 배열 인수를 uint32_t 두 요소에서 XMUINT2의 새 instance 초기화합니다.
XMUINT2 XMUINT2에 대한 생성자입니다. (오버로드 1/2)

XMUINT2 두 개의 uint32_t 인수에서 XMUINT2의 새 instance 초기화합니다.
XMUINT2 XMUINT2에 대한 생성자입니다. (오버로드 2/2)
XMUINT3 XMUINT3의 기본 생성자입니다.
XMUINT3 배열 인수를 uint32_t 세 요소에서 XMUINT3의 새 instance 초기화합니다.
XMUINT3 XMUINT3의 생성자입니다. (오버로드 2/2)
XMUINT3 세 개의 uint32_t 인수에서 XMUINT3의 새 instance 초기화합니다.
XMUINT3 XMUINT3의 생성자입니다. (오버로드 1/2)
XMUINT4 XMUINT4의 기본 생성자입니다.
XMUINT4 4개 요소 uint32_t 배열 인수에서 XMUINT4의 새 instance 초기화합니다.
XMUINT4 XMUINT4에 대한 생성자입니다. (오버로드 2/2)
XMUINT4 4개의 uint32_t 인수에서 XMUINT4의 새 instance 초기화합니다.
XMUINT4 XMUINT4에 대한 생성자입니다. (오버로드 1/2)
XMUNIBBLE4

XMUNIBBLE4의 기본 생성자입니다.
XMUNIBBLE4 4개의 요소 float 배열 인수에서 **XMUNIBBLE4**의 새 instance 초기화합니다.
XMUNIBBLE4 4개 요소 int8_t 배열 인수에서 XMUNIBBLE4의 새 instance 초기화합니다.
XMUNIBBLE4 XMUNIBBLE4에 대한 생성자입니다. (오버로드 2/2)
XMUNIBBLE4 네 개의 float 인수에서 XMUNIBBLE4의 새 instance 초기화합니다.
XMUNIBBLE4 압축된 형식의 구성 요소 데이터를 포함하는 uint16_t 변수에서 XMUNIBBLE의 새 instance 초기화합니다.
XMUNIBBLE4 4개의 int8_t 인수에서 XMUNIBBLE4의 새 instance 초기화합니다.
XMUNIBBLE4 XMUNIBBLE4에 대한 생성자입니다. (오버로드 1/2)
XMUNIBBLE4 XMUNIBBLE4의 기본 생성자입니다.
XMUNIBBLE4 4개의 요소 float 배열 인수에서 **XMUNIBBLE4**의 새 instance 초기화합니다.
XMUNIBBLE4 4개 요소 int8_t 배열 인수에서 XMUNIBBLE4의 새 instance 초기화합니다.
XMUNIBBLE4 XMUNIBBLE4에 대한 생성자입니다. (오버로드 2/2)
XMUNIBBLE4 네 개의 float 인수에서 XMUNIBBLE4의 새 instance 초기화합니다.

XMUNIBBLE4 압축된 형식의 구성 요소 데이터를 포함하는 uint16_t 변수에서 XMUNIBBLE의 새 instance 초기화합니다.
XMUNIBBLE4 4개의 int8_t 인수에서 XMUNIBBLE4의 새 instance 초기화합니다.
XMUNIBBLE4 XMUNIBBLE4의 생성자입니다. (오버로드 1/2)
XMUSHORT2 XMUSHORT2의 기본 생성자입니다.
XMUSHORT2 두 요소 float 배열 인수에서 XMUSHORT2의 새 instance 초기화합니다.
XMUSHORT2 uint16_t 인수 배열에서 XMUSHORT2의 새 instance 초기화합니다.
XMUSHORT2 XMUSHORT2의 생성자입니다. (오버로드 1/3)
XMUSHORT2 두 float 인수에서 XMUSHORT2의 새 instance 초기화합니다.
XMUSHORT2 두 개의 uint16_t 인수에서 XMUSHORT2의 새 instance 초기화합니다.
XMUSHORT2 XMUSHORT2의 생성자입니다. (오버로드 2/3)
XMUSHORT2 XMUSHORT2의 생성자입니다. (오버로드 3/3)
XMUSHORT2 XMUSHORT2의 기본 생성자입니다.

XMUSHORT2 두 요소 float 배열 인수에서 XMUSHORT2의 새 instance 초기화합니다.
XMUSHORT2 uint16_t 인수 배열에서 XMUSHORT2의 새 instance 초기화합니다.
XMUSHORT2 XMUSHORT2의 생성자입니다. (오버로드 1/3)
XMUSHORT2 두 float 인수에서 XMUSHORT2의 새 instance 초기화합니다.
XMUSHORT2 두 개의 uint16_t 인수에서 XMUSHORT2의 새 instance 초기화합니다.
XMUSHORT2 XMUSHORT2의 생성자입니다. (오버로드 2/3)
XMUSHORT2 XMUSHORT2의 생성자입니다. (오버로드 3/3)
XMUSHORT4 XMUSHORT4의 기본 생성자입니다.
XMUSHORT4 4개 요소 float 배열 인수에서 XMUSHORT4의 새 instance 초기화합니다.
XMUSHORT4 4개 요소 uint16_t 배열 인수에서 XMUSHORT4의 새 instance 초기화합니다.
XMUSHORT4 XMUSHORT4에 대한 생성자입니다. (오버로드 1/3)
XMUSHORT4 네 개의 float 인수에서 XMUSHORT4의 새 instance 초기화합니다.
XMUSHORT4

4개의 uint16_t 인수에서 XMUSHORT4의 새 instance 초기화합니다.
XMUSHORT4 XMUSHORT4에 대한 생성자입니다. (오버로드 2/3)
XMUSHORT4 XMUSHORT4에 대한 생성자입니다. (오버로드 3/3)
XMUSHORT4 XMUSHORT4의 기본 생성자입니다.
XMUSHORT4 4개 요소 float 배열 인수에서 XMUSHORT4의 새 instance 초기화합니다.
XMUSHORT4 4개 요소 uint16_t 배열 인수에서 XMUSHORT4의 새 instance 초기화합니다.
XMUSHORT4 XMUSHORT4에 대한 생성자입니다. (오버로드 1/3)
XMUSHORT4 네 개의 float 인수에서 XMUSHORT4의 새 instance 초기화합니다.
XMUSHORT4 4개의 uint16_t 인수에서 XMUSHORT4의 새 instance 초기화합니다.
XMUSHORT4 XMUSHORT4에 대한 생성자입니다. (오버로드 2/3)
XMUSHORT4 XMUSHORT4에 대한 생성자입니다. (오버로드 3/3)
XMUSHORTN2 XMUSHORTN2의 기본 생성자입니다.
XMUSHORTN2 XMUSHORTN2의 생성자입니다. (오버로드 4/4)

XMUSHORTN2 두 요소 uint16_t 배열 인수에서 XMUSHORTN2의 새 instance 초기화합니다.
XMUSHORTN2 XMUSHORTN2의 생성자입니다. (오버로드 1/4)
XMUSHORTN2 정규화된 두 부동 소수점에서 XMUSHORTN2의 새 instance 초기화합니다.
XMUSHORTN2 두 개의 uint16_t 인수에서 XMUSHORTN2의 새 instance 초기화합니다.
XMUSHORTN2 XMUSHORTN2에 대한 생성자입니다. (오버로드 2/4)
XMUSHORTN2 XMUSHORTN2에 대한 생성자입니다. (오버로드 3/4)
XMUSHORTN2 XMUSHORTN2의 기본 생성자입니다.
XMUSHORTN2 XMUSHORTN2에 대한 생성자입니다. (오버로드 4/4)
XMUSHORTN2 배열 인수를 uint16_t 두 요소에서 XMUSHORTN2의 새 instance 초기화합니다.
XMUSHORTN2 XMUSHORTN2에 대한 생성자입니다. (오버로드 1/4)
XMUSHORTN2 정규화된 두 부동 소수점에서 XMUSHORTN2의 새 instance 초기화합니다.
XMUSHORTN2 두 개의 uint16_t 인수에서 XMUSHORTN2의 새 instance 초기화합니다.
XMUSHORTN2

XMUSHORTN2에 대한 생성자입니다. (오버로드 2/4)
XMUSHORTN2 XMUSHORTN2에 대한 생성자입니다. (오버로드 3/4)
XMUSHORTN4 XMUSHORTN4의 기본 생성자입니다.
XMUSHORTN4 4개의 요소 float 배열 인수에서 XMUSHORTN4의 새 instance 초기화합니다.
XMUSHORTN4 4개 요소 uint16_t 배열 인수에서 XMUSHORTN4의 새 instance 초기화합니다.
XMUSHORTN4 XMUSHORTN4에 대한 생성자입니다. (오버로드 1/3)
XMUSHORTN4 정규화된 네 개의 부동 소수점에서 XMUSHORTN4의 새 instance 초기화합니다.
XMUSHORTN4 4개의 uint16_t 인수에서 XMUSHORTN4의 새 instance 초기화합니다.
XMUSHORTN4 XMUSHORTN4에 대한 생성자입니다. (오버로드 2/3)
XMUSHORTN4 XMUSHORTN4에 대한 생성자입니다. (오버로드 3/3)
XMUSHORTN4 XMUSHORTN4의 기본 생성자입니다.
XMUSHORTN4 4개의 요소 float 배열 인수에서 XMUSHORTN4의 새 instance 초기화합니다.
XMUSHORTN4 4개 요소 uint16_t 배열 인수에서 XMUSHORTN4의 새 instance 초기화합니다.

XMUSHORTN4 XMUSHORTN4에 대한 생성자입니다. (오버로드 1/3)
XMUSHORTN4 정규화된 네 개의 부동 소수점에서 XMUSHORTN4의 새 instance 초기화합니다.
XMUSHORTN4 4개의 uint16_t 인수에서 XMUSHORTN4의 새 instance 초기화합니다.
XMUSHORTN4 XMUSHORTN4에 대한 생성자입니다. (오버로드 2/3)
XMUSHORTN4 XMUSHORTN4에 대한 생성자입니다. (오버로드 3/3)
XMVector2AngleBetweenNormals 정규화된 두 2D 벡터 사이의 라디안 각도를 계산합니다.
XMVector2AngleBetweenNormalsEst 정규화된 두 2D 벡터 사이의 라디안 각도를 예측합니다.
XMVector2AngleBetweenVectors 두 2D 벡터 사이의 라디안 각도를 계산합니다.
XMVector2ClampLength 2D 벡터의 길이를 지정된 범위로 고정합니다. (XMVector2ClampLength)
XMVector2ClampLengthV 2D 벡터의 길이를 지정된 범위로 고정합니다. (XMVector2ClampLengthV)
XMVector2Cross 2D 교차 제품을 계산합니다.
XMVector2Dot 2D 벡터 사이의 점 제품을 계산합니다.
XMVector2Equal

두 2D 벡터가 같은지 테스트합니다.
XMVector2EqualInt 두 개의 2D 벡터가 같은지 테스트하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVector2EqualIntR 두 개의 2D 벡터가 같은지 테스트하여 각 구성 요소를 부호 없는 정수로 처리합니다. 또한 이 함수는 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector2EqualR 두 2D 벡터가 같은지 테스트합니다. 또한 이 함수는 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector2Greater 한 2D 벡터가 다른 2D 벡터보다 큰지 테스트합니다.
XMVector2GreaterOrEqual 한 2D 벡터가 다른 2D 벡터보다 크거나 같은지 테스트합니다.
XMVector2GreaterOrEqualR 한 2D 벡터가 다른 2D 벡터보다 크거나 같은지 테스트하고 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector2GreaterR 한 2D 벡터가 다른 2D 벡터보다 큰지 테스트하고 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector2InBounds 2D 벡터의 구성 요소가 설정된 범위 내에 있는지 테스트합니다.
XMVector2IntersectLine 두 줄의 교집합을 찾습니다.
XMVector2IsInfinite 2D 벡터의 구성 요소가 양수인지 음의 무한대인지 테스트합니다.
XMVector2IsNaN 2D 벡터의 구성 요소가 NaN인지 테스트합니다.

XMVector2Length 2D 벡터의 길이를 계산합니다.
XMVector2LengthEst 2D 벡터의 길이를 예측합니다.
XMVector2LengthSq 2D 벡터 길이의 제곱을 계산합니다.
XMVector2Less 하나의 2D 벡터가 다른 2D 벡터보다 작은지 테스트합니다.
XMVector2LessOrEqual 한 2D 벡터가 다른 2D 벡터보다 작거나 같은지 테스트합니다.
XMVector2LinePointDistance 선과 점 사이의 최소 거리를 계산합니다. (XMVector2LinePointDistance)
XMVector2NearEqual 하나의 2D 벡터가 다른 2D 벡터 근처에 있는지 테스트합니다.
XMVector2Normalize 정규화된 2D 벡터 버전을 반환합니다.
XMVector2NormalizeEst 정규화된 2D 벡터 버전을 예측합니다.
XMVector2NotEqual 두 2D 벡터가 같지 않은지 테스트합니다.
XMVector2NotEqualInt 두 벡터가 같지 않은지 테스트하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVector2Orthogonal 2D 벡터에 수직인 벡터를 계산합니다.
XMVector2ReciprocalLength

2D 벡터 길이의 상호 계산
XMVector2ReciprocalLengthEst 2D 벡터 길이의 상호를 추정합니다.
XMVector2Reflect 2D 일반 벡터에서 인시던트 2D 벡터를 반영합니다.
XMVector2Refract 2D 일반 벡터에서 인시던트 2D 벡터를 굴절합니다. (XMVector2Refract)
XMVector2RefractV 2D 일반 벡터에서 인시던트 2D 벡터를 굴절합니다. (XMVector2RefractV)
XMVector2Transform 행렬별로 2D 벡터를 변환합니다.
XMVector2TransformCoord 지정된 행렬에 의해 2D 벡터를 변환하여 결과를 w = 1로 다시 프로젝션합니다.
XMVector2TransformCoordStream 지정된 행렬을 통해 2D 벡터 스트림을 변환하여 w 좌표가 1.0과 같도록 결과 벡터를 프로젝션합니다.
XMVector2TransformNormal 지정된 행렬에 의해 2D 벡터 정규를 변환합니다.
XMVector2TransformNormalStream 지정된 행렬을 통해 2D 일반 벡터의 스트림을 변환합니다.
XMVector2TransformStream 지정된 행렬을 통해 2D 벡터의 스트림을 변환합니다.
XMVector3AngleBetweenNormals 정규화된 두 3D 벡터 사이의 라디안 각도를 계산합니다.
XMVector3AngleBetweenNormalsEst 정규화된 두 3D 벡터 사이의 라디안 각도를 예측합니다.

XMVector3AngleBetweenVectors 두 3D 벡터 사이의 라디안 각도를 계산합니다.
XMVector3ClampLength 3D 벡터의 길이를 지정된 범위로 고정합니다. (XMVector3ClampLength)
XMVector3ClampLengthV 3D 벡터의 길이를 지정된 범위로 고정합니다. (XMVector3ClampLengthV)
XMVector3ComponentsFromNormal 참조 일반 벡터를 사용하여 3D 벡터를 일반과 병렬 및 수직인 구성 요소로 분할합니다.
XMVector3Cross 두 3D 벡터 간의 교차 제품을 계산합니다.
XMVector3Dot 3D 벡터 간에 점 제품을 계산합니다.
XMVector3Equal 두 3D 벡터가 같은지 테스트합니다.
XMVector3EqualInt 두 3D 벡터가 같은지 테스트하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVector3EqualIntR 두 3D 벡터가 같은지 테스트하여 각 구성 요소를 부호 없는 정수로 처리합니다. 또한 이 함수는 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector3EqualR 두 3D 벡터가 같은지 테스트합니다. 또한 이 함수는 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector3Greater 하나의 3D 벡터가 다른 3D 벡터보다 큰지 테스트합니다.
XMVector3GreaterOrEqual 한 3D 벡터가 다른 3D 벡터보다 크거나 같은지 테스트합니다.

XMVector3GreaterOrEqualR 한 3D 벡터가 다른 3D 벡터보다 크거나 같은지 테스트하고 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector3GreaterR 한 3D 벡터가 다른 3D 벡터보다 큰지 테스트하고 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector3InBounds 3D 벡터의 구성 요소가 설정된 범위 내에 있는지 테스트합니다.
XMVector3InverseRotate 쿼터니언의 역방향을 사용하여 3D 벡터를 회전합니다.
XMVector3IsInfinite 3D 벡터의 구성 요소가 양수인지 음의 무한대인지 테스트합니다.
XMVector3IsNaN 3D 벡터의 구성 요소가 NaN인지 테스트합니다.
XMVector3Length 3D 벡터의 길이를 계산합니다.
XMVector3LengthEst 3D 벡터의 길이를 예측합니다.
XMVector3LengthSq 3D 벡터 길이의 제곱을 계산합니다.
XMVector3Less 한 3D 벡터가 다른 3D 벡터보다 작은지 테스트합니다.
XMVector3LessOrEqual 한 3D 벡터가 다른 3D 벡터보다 작거나 같은지 테스트합니다.
XMVector3LinePointDistance 선과 점 사이의 최소 거리를 계산합니다. (XMVector3LinePointDistance)

XMVector3NearEqual 하나의 3D 벡터가 다른 3D 벡터 근처에 있는지 테스트합니다.
XMVector3Normalize 3D 벡터의 정규화된 버전을 반환합니다.
XMVector3NormalizeEst 3D 벡터의 정규화된 버전을 예측합니다.
XMVector3NotEqual 두 개의 3D 벡터가 같지 않은지 테스트합니다.
XMVector3NotEqualInt 두 개의 3D 벡터가 같지 않은지 테스트하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVector3Orthogonal 3D 벡터에 수직인 벡터를 계산합니다.
XMVector3Project 개체 공간에서 화면 공간으로 3D 벡터를 프로젝터합니다.
XMVector3ProjectStream 개체 공간에서 화면 공간으로 3D 벡터 스트림을 프로젝터합니다.
XMVector3ReciprocalLength 3D 벡터 길이의 상호 계산
XMVector3ReciprocalLengthEst 3D 벡터 길이의 상호를 추정합니다.
XMVector3Reflect 3D 일반 벡터에서 인시던트 3D 벡터를 반영합니다.
XMVector3Refract 3D 일반 벡터에서 인시던트 3D 벡터를 굴절합니다. (XMVector3Refract)
XMVector3RefractV

3D 일반 벡터에서 인시던트 3D 벡터를 굴절합니다. (XMVector3RefractV)
XMVector3Rotate 쿼터니언을 사용하여 3D 벡터를 회전합니다.
XMVector3Transform 행렬별로 3D 벡터를 변환합니다.
XMVector3TransformCoord 지정된 행렬에 의해 3D 벡터를 변환하여 결과를 w = 1로 다시 프로젝션합니다.
XMVector3TransformCoordStream 지정된 행렬에 따라 3D 벡터 스트림을 변환하여 w 좌표가 1.0과 같도록 결과 벡터를 프로젝션합니다.
XMVector3TransformNormal 지정된 행렬에 의해 3D 벡터 정규를 변환합니다.
XMVector3TransformNormalStream 지정된 행렬을 통해 3D 일반 벡터의 스트림을 변환합니다.
XMVector3TransformStream 지정된 행렬을 통해 3D 벡터 스트림을 변환합니다.
XMVector3Unproject 화면 공간에서 개체 공간으로 3D 벡터를 프로젝터합니다.
XMVector3UnprojectStream 3D 벡터 스트림을 화면 공간에서 개체 공간으로 변환합니다.
XMVector4AngleBetweenNormals 정규화된 두 4D 벡터 사이의 방사형 각도를 계산합니다.
XMVector4AngleBetweenNormalsEst 정규화된 두 4D 벡터 사이의 라디안 각도를 예측합니다.
XMVector4AngleBetweenVectors 두 4D 벡터 사이의 방사형 각도를 계산합니다.

XMVector4ClampLength 4D 벡터의 길이를 지정된 범위로 고정합니다. (XMVector4ClampLength)
XMVector4ClampLengthV 4D 벡터의 길이를 지정된 범위로 고정합니다. (XMVector4ClampLengthV)
XMVector4Cross 4D 교차 제품을 계산합니다.
XMVector4Dot 4D 벡터 간에 점 제품을 계산합니다.
XMVector4Equal 두 4D 벡터가 같은지 테스트합니다.
XMVector4EqualInt 두 개의 4D 벡터가 같은지 테스트하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVector4EqualIntR 두 개의 4D 벡터가 같은지 테스트하여 각 구성 요소를 부호 없는 정수로 처리합니다. 또한 이 함수는 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector4EqualR 두 4D 벡터가 같은지 테스트합니다. 또한 이 함수는 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector4Greater 하나의 4D 벡터가 다른 4D 벡터보다 큰지 테스트합니다.
XMVector4GreaterOrEqual 한 4D 벡터가 다른 4D 벡터보다 크거나 같은지 테스트합니다.
XMVector4GreaterOrEqualR 한 4D 벡터가 다른 4D 벡터보다 크거나 같은지 테스트하고 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector4GreaterR

한 4D 벡터가 다른 4D 벡터보다 큰지 테스트하고 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector4InBounds 4D 벡터의 구성 요소가 설정된 범위 내에 있는지 테스트합니다.
XMVector4IsInfinite 4D 벡터의 구성 요소가 양수인지 음의 무한대인지 테스트합니다.
XMVector4IsNaN 4D 벡터의 구성 요소가 NaN인지 테스트합니다.
XMVector4Length 4D 벡터의 길이를 계산합니다.
XMVector4LengthEst 4D 벡터의 길이를 예측합니다.
XMVector4LengthSq 4D 벡터 길이의 제곱을 계산합니다.
XMVector4Less 하나의 4D 벡터가 다른 4D 벡터보다 작은지 테스트합니다.
XMVector4LessOrEqual 한 4D 벡터가 다른 4D 벡터보다 작거나 같은지 테스트합니다.
XMVector4NearEqual 하나의 4D 벡터가 다른 4D 벡터 근처에 있는지 테스트합니다.
XMVector4Normalize 정규화된 4D 벡터 버전을 반환합니다.
XMVector4NormalizeEst 4D 벡터의 정규화된 버전을 예측합니다.
XMVector4NotEqual 두 4D 벡터가 같지 않은지 테스트합니다.

XMVector4NotEqualInt 두 개의 4D 벡터가 같지 않은지 테스트하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVector4Orthogonal 4D 벡터에 수직인 벡터를 계산합니다.
XMVector4ReciprocalLength 4D 벡터 길이의 상호를 계산합니다.
XMVector4ReciprocalLengthEst 4D 벡터 길이의 상호를 추정합니다.
XMVector4Reflect 4D 일반 벡터에서 인시던트 4D 벡터를 반영합니다.
XMVector4Refract 4D 일반 벡터에서 인시던트 4D 벡터를 굴절합니다. (XMVector4Refract)
XMVector4RefractV 4D 일반 벡터에서 인시던트 4D 벡터를 굴절합니다. (XMVector4RefractV)
XMVector4Transform 행렬별로 4D 벡터를 변환합니다.
XMVector4TransformStream 지정된 행렬을 통해 4D 벡터 스트림을 변환합니다.
XMVectorAbs XMVECTOR의 각 구성 요소에 대한 절대값을 계산합니다.
XMVectorACos XMVECTOR의 각 구성 요소의 아크코사인을 계산합니다.
XMVectorACosEst XMVECTOR의 각 구성 요소의 아크코사인을 추정합니다.
XMVectorAdd

두 벡터의 합계를 계산합니다.
XMVectorAddAngles 각도를 나타내는 두 개의 벡터를 추가합니다.
XMVectorAndCInt 두 번째 벡터의 부정을 사용하여 한 벡터의 논리적 AND를 계산하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVectorAndInt 두 벡터의 논리적 AND를 계산하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVectorASin XMVECTOR의 각 구성 요소의 아크사인을 계산합니다.
XMVectorASinEst XMVECTOR의 각 구성 요소의 아크사인을 추정합니다.
XMVectorATan XMVECTOR의 각 구성 요소의 아크탄젠트를 계산합니다.
XMVectorATan2 Y/X의 아크탄젠트를 계산합니다.
XMVectorATan2Est Y/X의 아크탄젠트를 추정합니다.
XMVectorATanEst XMVECTOR의 각 구성 요소의 아크탄젠트를 예측합니다.
XMVectorBaryCentric 지정된 위치 벡터를 사용하여 Barycentric 좌표의 점을 반환합니다. (XMVectorBaryCentric)
XMVectorBaryCentricV 지정된 위치 벡터를 사용하여 Barycentric 좌표의 점을 반환합니다. (XMVectorBaryCentricV)
XMVectorCatmullRom 지정된 위치 벡터를 사용하여 Catmull-Rom 보간을 수행합니다. (XMVectorCatmullRom)

XMVectorCatmullRomV 지정된 위치 벡터를 사용하여 Catmull-Rom 보간을 수행합니다. (XMVectorCatmullRomV)
XMVectorCeiling XMVECTOR의 각 구성 요소의 최대값을 계산합니다.
XMVectorClamp 벡터의 구성 요소를 지정된 최소 및 최대 범위로 고정합니다.
XMVectorCos XMVECTOR의 각 구성 요소의 코사인을 계산합니다.
XMVectorCosEst XMVECTOR의 각 구성 요소의 코사인을 추정합니다.
XMVectorCosH XMVECTOR의 각 구성 요소에 대한 쌍곡선 코사인을 계산합니다.
XMVectorDivide XMVECTOR의 한 instance 두 번째 instance 나누어 결과를 세 번째 instance 반환합니다. (XMVectorDivide)
XMVectorEqual 두 벡터의 같음을 위해 구성 요소별 테스트를 수행합니다.
XMVectorEqualInt 두 벡터의 같음을 위해 구성 요소별 테스트를 수행하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVectorEqualIntR 두 벡터의 같음을 위해 구성 요소별 테스트를 수행하여 각 구성 요소를 부호 없는 정수로 처리합니다. 또한 이 함수는 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 설정합니다.
XMVectorEqualR 두 벡터의 같음에 대한 구성 요소별 테스트를 수행하고 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 설정합니다.

XMVectorExp 각 구성 요소의 전원에 대해 발생된 두 개를 계산합니다. 참고 이 함수는 기존 Windows 8 코드의 XMVectorExp2에 대한 호환성 별칭입니다. 이 함수는 Windows 8.1 더 이상 사용되지 않습니다. 사용하지 말고 대신 XMVectorExp2 또는 XMVectorExpE를 사용합니다. .
XMVectorExp10 각 구성 요소의 전원으로 10개를 계산합니다.
XMVectorExp2 각 구성 요소의 전원에 대해 두 개를 계산합니다.
XMVectorExpE 각 구성 요소의 전원으로 발생하는 e(~2.71828)를 계산합니다.
XMVectorFalseInt 0(false) 벡터를 반환합니다.
XMVectorFloor XMVECTOR의 각 구성 요소의 바닥을 계산합니다.
XMVectorGetByIndex 인덱스별 부동 소수점 데이터를 포함하는 XMVECTOR 데이터 형식의 네 가지 구성 요소 중 하나의 값을 검색합니다.
XMVectorGetByIndexPtr 포인터에서 참조하는 부동 소수점의 instance 인덱스에서 참조하는 부동 소수점 데이터를 포함하는 XMVECTOR 데이터 형식의 4개 구성 요소 중 하나의 값을 검색합니다.
XMVectorGetIntByIndex 인덱스별 정수 데이터를 포함하는 XMVECTOR 데이터 형식의 네 가지 구성 요소 중 하나의 값을 검색합니다.
XMVectorGetIntByIndexPtr 포인터에서 참조하는 정수의 instance 인덱스별 정수 데이터를 포함하는 XMVECTOR 데이터 형식의 4개 구성 요소 중 하나의 값을 검색합니다.
XMVectorGetIntW XMVECTOR 데이터 형식의 w 구성 요소를 검색합니다. (XMVectorGetIntW)

XMVectorGetIntWPtr 정수 데이터가 포함된 XMVECTOR 데이터 형식의 w 구성 요소를 검색하고 포인터에서 참조하는 uint32_t instance 해당 구성 요소의 값을 저장합니다.
XMVectorGetIntX XMVECTOR 데이터 형식의 x 구성 요소를 검색합니다. (XMVectorGetIntX)
XMVectorGetIntXPtr 정수 데이터가 포함된 XMVECTOR 데이터 형식의 x 구성 요소를 검색하고 포인터에서 참조하는 uint32_t instance 해당 구성 요소의 값을 저장합니다.
XMVectorGetIntY XMVECTOR 데이터 형식의 y 구성 요소를 검색합니다. (XMVectorGetIntY)
XMVectorGetIntYPtr 정수 데이터를 포함하는 XMVECTOR 데이터 형식의 y 구성 요소를 검색하고 포인터에서 참조하는 uint32_t instance 해당 구성 요소의 값을 저장합니다.
XMVectorGetIntZ XMVECTOR 데이터 형식의 z 구성 요소를 검색합니다. (XMVectorGetIntZ)
XMVectorGetIntZPtr 정수 데이터를 포함하는 XMVECTOR 데이터 형식의 z 구성 요소를 검색하고 포인터에서 참조하는 uint32_t instance 해당 구성 요소의 값을 저장합니다.
XMVectorGetW XMVECTOR 데이터 형식의 w 구성 요소를 검색합니다. (XMVectorGetW)
XMVectorGetWPtr 부동 소수점 데이터를 포함하는 XMVECTOR 데이터 형식의 w 구성 요소를 검색하고 포인터에서 참조하는 부동 소수점 instance 해당 구성 요소의 값을 저장합니다.
XMVectorGetX XMVECTOR 데이터 형식의 x 구성 요소를 검색합니다. (XMVectorGetX)
XMVectorGetXPtr 부동 소수점 데이터를 포함하는 XMVECTOR 데이터 형식의 x 구성 요소를 검색하고 포인터에서 참조하는 부동 소수점 instance 해당 구성 요소의 값을 저장합니다.

XMVectorGetY XMVECTOR 데이터 형식의 y 구성 요소를 검색합니다. (XMVectorGetY)
XMVectorGetYPtr 부동 소수점 데이터가 포함된 XMVECTOR 데이터 형식의 y 구성 요소를 검색하고 포인터에서 참조하는 부동 소수점 instance 해당 구성 요소의 값을 저장합니다.
XMVectorGetZ XMVECTOR 데이터 형식의 z 구성 요소를 검색합니다. (XMVectorGetZ)
XMVectorGetZPtr 부동 소수점 데이터를 포함하는 XMVECTOR 데이터 형식의 z 구성 요소를 검색하고 포인터에서 참조하는 부동 소수점 instance 해당 구성 요소의 값을 저장합니다.
XMVectorGreater 두 벡터 사이의 보다 큰 에 대한 구성 요소별 테스트를 수행합니다.
XMVectorGreaterOrEqual 두 벡터 간에 크거나 같은 구성 요소별 테스트를 수행합니다.
XMVectorGreaterOrEqualR 두 벡터 간에 크거나 같은 구성 요소별 테스트를 수행하고 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 설정합니다.
XMVectorGreaterR 두 벡터 간보다 큰 구성 요소별 테스트를 수행하고 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 설정합니다.
XMVectorHermite 지정된 벡터를 사용하여 Hermite 스플라인 보간을 수행합니다. (XMVectorHermite)
XMVectorHermiteV 지정된 벡터를 사용하여 Hermite 스플라인 보간을 수행합니다. (XMVectorHermiteV)
XMVectorInBounds 지정된 벡터의 구성 요소가 설정된 범위 내에 있는지 여부를 테스트합니다.

XMVectorInBoundsR 지정된 벡터의 구성 요소가 특정 범위 내에 있는지 테스트하고 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 설정합니다.
XMVectorInsert 지정된 수의 32비트 구성 요소에서 왼쪽으로 벡터를 회전하고 해당 결과의 선택한 요소를 다른 벡터에 삽입합니다.
XMVectorIsInfinite 벡터에서 +/- 무한대에 대한 구성 요소별 테스트를 수행합니다.
XMVectorIsNaN 벡터에서 구성 요소별 NaN 테스트를 수행합니다.
XMVectorLerp 두 벡터 간의 선형 보간을 수행합니다. (XMVectorLerp)
XMVectorLerpV 두 벡터 간의 선형 보간을 수행합니다. (XMVectorLerpV)
XMVectorLess 두 벡터 간보다 작을 경우 구성 요소별 테스트를 수행합니다.
XMVectorLessOrEqual 두 벡터 간의 작거나 같음에 대한 구성 요소별 테스트를 수행합니다.
XMVectorLog 벡터의 각 구성 요소에 대한 기본 두 로그를 계산합니다. 참고 이 함수는 기존 Windows 8 코드의 XMVectorLog2에 대한 호환성 별칭입니다.
XMVectorLog10 벡터의 각 구성 요소에 대한 기본 10개의 로그를 계산합니다.
XMVectorLog2 벡터의 각 구성 요소에 대한 기본 두 로그를 계산합니다.
XMVectorLoge

벡터의 각 구성 요소에 대한 기본 e 로그를 계산합니다.
XMVectorMax 두 벡터를 구성 요소별로 비교하고 가장 큰 구성 요소를 포함하는 벡터를 반환합니다.
XMVectorMergeXY 두 벡터의 x 및 y 구성 요소를 결합하여 새 벡터를 만듭니다.
XMVectorMergeZW 두 벡터의 z 및 w 구성 요소를 결합하여 새 벡터를 만듭니다.
XMVectorMin 두 벡터를 구성 요소별로 비교하고 가장 작은 구성 요소를 포함하는 벡터를 반환합니다.
XMVectorMod 두 벡터의 몫의 구성 요소별 부동 소수점 나머지를 계산합니다.
XMVectorModAngles 구성 요소별 각도 모듈로 2PI를 계산합니다.
XMVectorMultiply 두 벡터의 구성 요소별 제품을 계산합니다.
XMVectorMultiplyAdd 세 번째 벡터에 추가된 처음 두 벡터의 곱을 계산합니다.
XMVectorNearEqual 지정된 임계값 내에서 두 벡터의 같음을 위해 구성 요소별 테스트를 수행합니다.
XMVectorNegate 벡터의 부정을 계산합니다.
XMVectorNegativeMultiplySubtract 세 번째 벡터와 처음 두 벡터의 곱의 차이를 계산합니다.
XMVectorNorInt 두 벡터의 논리적 NOR을 계산하여 각 구성 요소를 부호 없는 정수로 처리합니다.

XMVectorNotEqual 두 벡터의 같지 않음에 대한 구성 요소별 테스트를 수행합니다.
XMVectorNotEqualInt 두 벡터의 같지 않음에 대한 구성 요소별 테스트를 수행하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVectorOrInt 두 벡터의 논리적 OR을 계산하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVectorPermute 두 벡터의 구성 요소를 permutes하여 새 벡터를 만듭니다.
XMVectorPow V1을 V2의 강력한 성능으로 계산합니다.
XMVectorReciprocal 벡터의 구성 요소별 상호를 계산합니다.
XMVectorReciprocalEst 벡터의 구성 요소별 상호를 추정합니다.
XMVectorReciprocalSqrt 벡터의 구성 요소별 상호 제곱근을 계산합니다.
XMVectorReciprocalSqrtEst 벡터의 구성 요소별 상호 제곱근을 예측합니다.
XMVectorReplicate 부동 소수점 값을 벡터의 네 가지 구성 요소 모두에 복제합니다.
XMVectorReplicateInt 정수 값을 벡터의 네 가지 구성 요소 모두에 복제합니다.
XMVectorReplicateIntPtr 포인터에서 참조하는 정수 값을 벡터의 네 가지 구성 요소 모두에 복제합니다.

XMVectorReplicatePtr 포인터에서 참조하는 부동 소수점 값을 벡터의 네 가지 구성 요소 모두에 복제합니다.
XMVectorRotateLeft 지정된 수의 32비트 요소로 왼쪽 벡터를 회전합니다.
XMVectorRotateRight 지정된 수의 32비트 요소로 벡터를 오른쪽으로 회전합니다.
XMVectorRound 벡터의 각 구성 요소를 가장 가까운 정수("은행 반올림"라고 함)로 반올림합니다.
XMVectorSaturate 벡터의 각 구성 요소를 0.0f에서 1.0f 범위로 포화합니다.
XMVectorScale 스칼라는 벡터를 부동 소수점 값으로 곱합니다.
XMVectorSelect 두 입력 벡터 간에 구성 요소별 선택을 수행하고 결과 벡터를 반환합니다.
XMVectorSelectControl XMVectorSelect에서 사용할 컨트롤 벡터를 정의합니다.
XMVectorSet 4개의 부동 소수점 값을 사용하여 벡터를 만듭니다.
XMVectorSetBinaryConstant 각 구성 요소가 0.0f 또는 1.0f인 벡터를 만듭니다.
XMVectorSetByIndex 부동 소수점 개체를 사용하여 인덱스에서 참조하는 정수 데이터를 포함하는 XMVECTOR 데이터 형식의 네 가지 구성 요소 중 하나의 값을 설정합니다.
XMVectorSetByIndexPtr 부동 소수점 instance 대한 포인터를 사용하여 인덱스에서 참조하는 부동 소수점 데이터를 포함하는 XMVECTOR 데이터 형식의 네 가지 구성 요소 중 하나의 값을 설정합니다.

XMVectorSetInt 부호 없는 정수 구성 요소를 사용하여 벡터를 만듭니다.
XMVectorSetIntByIndex 정수 instance 사용하여 인덱스에서 참조하는 정수 데이터를 포함하는 XMVECTOR 데이터 형식의 네 가지 구성 요소 중 하나의 값을 설정합니다.
XMVectorSetIntByIndexPtr 정수 instance 대한 포인터를 사용하여 인덱스에서 참조하는 XMVECTOR 데이터 형식의 네 가지 구성 요소 중 하나의 값을 설정합니다.
XMVectorSetIntW XMVECTOR 데이터 형식의 w 구성 요소 값을 설정합니다. (XMVectorSetIntW)
XMVectorSetIntWPtr 정수 데이터가 포함된 XMVECTOR의 w 구성 요소를 포인터에서 참조하는 uint32_t instance 포함된 값을 설정합니다.
XMVectorSetIntX XMVECTOR 데이터 형식의 x 구성 요소 값을 설정합니다. (XMVectorSetIntX)
XMVectorSetIntXPtr 정수 데이터를 포함하는 XMVECTOR의 x 구성 요소를 포인터에서 참조하는 uint32_t instance 포함된 값으로 설정합니다.
XMVectorSetInty XMVECTOR 데이터 형식의 y 구성 요소 값을 설정합니다. (XMVectorSetIntY)
XMVectorSetIntYPtr 정수 데이터를 포함하는 XMVECTOR의 y 구성 요소를 포인터에서 참조하는 uint32_t instance 포함된 값으로 설정합니다.
XMVectorSetIntZ XMVECTOR 데이터 형식의 z 구성 요소 값을 설정합니다. (XMVectorSetIntZ)
XMVectorSetIntZPtr 정수 데이터가 포함된 XMVECTOR의 z 구성 요소를 포인터에서 참조하는 uint32_t instance 포함된 값으로 설정합니다.

XMVectorSetW XMVECTOR 데이터 형식의 w 구성 요소 값을 설정합니다. (XMVectorSetW)
XMVectorSetWPtr 부동 소수점 데이터를 포함하는 XMVECTOR의 w 구성 요소를 포인터가 참조하는 부동 소수점의 instance 포함된 값으로 설정합니다.
XMVectorSetX XMVECTOR 데이터 형식의 x 구성 요소 값을 설정합니다. (XMVectorSetX)
XMVectorSetXPtr 부동 소수점 데이터를 포함하는 XMVECTOR의 x 구성 요소를 포인터가 참조하는 부동 소수점 instance 포함된 값으로 설정합니다.
XMVectorSety XMVECTOR 데이터 형식의 y 구성 요소 값을 설정합니다. (XMVectorSety)
XMVectorSetYPtr 부동 소수점 데이터를 포함하는 XMVECTOR의 y 구성 요소를 포인터가 참조하는 부동 소수점 instance 포함된 값으로 설정합니다.
XMVectorSetZ XMVECTOR 데이터 형식의 z 구성 요소 값을 설정합니다. (XMVectorSetZ)
XMVectorSetZPtr 부동 소수점 데이터를 포함하는 XMVECTOR의 z 구성 요소를 포인터에서 참조하는 부동 소수점의 instance 포함된 값으로 설정합니다.
XMVectorShiftLeft 지정된 수의 32비트 요소에서 왼쪽으로 벡터를 이동하여 비어 있는 요소를 두 번째 벡터의 요소로 채웁니다.
XMVectorSin XMVECTOR의 각 구성 요소의 사인을 계산합니다.
XMVectorSinCos XMVECTOR의 각 구성 요소에 대한 사인 및 코사인을 계산합니다.

XMVectorSinCosEst XMVECTOR의 각 구성 요소에 대한 사인 및 코사인을 추정합니다.
XMVectorSinEst XMVECTOR의 각 구성 요소의 사인을 추정합니다.
XMVectorSinH XMVECTOR의 각 구성 요소에 대한 쌍곡선 사인을 계산합니다.
XMVectorSplatConstant 동일한 부동 소수점 구성 요소를 사용하여 벡터를 만듭니다. 각 구성 요소는 정수 지수로 2로 나눈 상수입니다.
XMVectorSplatConstantInt 동일한 정수 구성 요소를 사용하여 벡터를 만듭니다.
XMVectorSplatEpsilon 각 구성 요소가 epsilon인 벡터를 반환합니다(1.192092896e-7).
XMVectorSplatInfinity 각 구성 요소가 무한대(0x7F800000)인 벡터를 반환합니다.
XMVectorSplatOne 각 구성 요소가 하나인 벡터를 반환합니다.
XMVectorSplatQNaN 각 구성 요소가 QNaN(0x7CF00000)인 벡터를 반환합니다.
XMVectorSplatSignMask 각 구성 요소가 기호 마스크(0x80000000)인 벡터를 반환합니다.
XMVectorSplatW 벡터의 w 구성 요소를 모든 구성 요소에 복제합니다.
XMVectorSplatX 벡터의 x 구성 요소를 모든 구성 요소에 복제합니다.

XMVectorSplatY 벡터의 y 구성 요소를 모든 구성 요소에 복제합니다.
XMVectorSplatZ 벡터의 z 구성 요소를 모든 구성 요소에 복제합니다.
XMVectorSqrt 벡터의 구성 요소별 제곱근을 계산합니다.
XMVectorSqrtEst 벡터의 구성 요소별 제곱근을 추정합니다.
XMVectorSubtract 두 벡터의 차이를 계산합니다.
XMVectorSubtractAngles 각도를 나타내는 두 개의 벡터를 뺍니다.
XMVectorSum XMVECTOR 구성 요소의 가로 합계를 계산합니다. 가로 합계는 벡터의 각 구성 요소를 함께 추가한 결과입니다.
XMVectorSwizzle 벡터를 스위즐합니다.
XMVectorTan XMVECTOR의 각 구성 요소의 탄젠트를 계산합니다.
XMVectorTanEst XMVECTOR의 각 구성 요소의 탄젠트를 추정합니다.
XMVectorTanH XMVECTOR의 각 구성 요소에 대한 쌍곡선 탄젠트를 계산합니다.
XMVectorTrueInt 각 구성 요소가 true(0xFFFFFFFF)를 나타내는 벡터를 반환합니다.

XMVectorTruncate 벡터의 각 구성 요소를 0 방향으로 가장 가까운 정수 값으로 반올림합니다.
XMVectorXorInt 두 벡터의 논리 XOR을 계산하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVectorZero 0 벡터를 만듭니다.
XMVerifyCPUSupport DirectXMath 라이브러리가 현재 플랫폼을 지원하는지 나타냅니다.
XMXDEC4 XMXDEC4의 기본 생성자입니다.
XMXDEC4 4개 요소 float 배열 인수에서 XMXDEC4의 새 instance 초기화합니다.
XMXDEC4 XMXDEC4에 대한 생성자입니다. (오버로드 1/2)
XMXDEC4 네 개의 float 인수에서 XMXDEC4의 새 instance 초기화합니다.
XMXDEC4 압축된 형식의 구성 요소 데이터를 포함하는 uint32_t6 변수에서 XMXDEC4의 새 instance 초기화합니다.
XMXDEC4 XMXDEC4에 대한 생성자입니다. (오버로드 2/2)
XMXDEC4 XMXDEC4의 기본 생성자입니다.
XMXDEC4 4개 요소 float 배열 인수에서 XMXDEC4의 새 instance 초기화합니다.

XMXDEC4 XMXDEC4에 대한 생성자입니다. (오버로드 1/2)
XMXDEC4 네 개의 float 인수에서 XMXDEC4의 새 instance 초기화합니다.
XMXDEC4 압축된 형식의 구성 요소 데이터를 포함하는 uint32_t6 변수에서 XMXDEC4의 새 instance 초기화합니다.
XMXDEC4 XMXDEC4에 대한 생성자입니다. (오버로드 2/2)
XMXDECN4 XMXDECN4의 기본 생성자입니다.
XMXDECN4 4개의 요소 float 배열 인수에서 XMXDECN4의 새 instance 초기화합니다.
XMXDECN4 XMXDECN4에 대한 생성자입니다. (오버로드 1/2)
XMXDECN4 정규화된 네 개의 부동 소수점에서 XMXDECN4의 새 instance 초기화합니다.
XMXDECN4 압축된 형식의 구성 요소 데이터를 포함하는 uint32_t 변수에서 XMXDECN4의 새 instance 초기화합니다.
XMXDECN4 XMXDECN4에 대한 생성자입니다. (오버로드 2/2)
XMXDECN4 XMXDECN4의 기본 생성자입니다.
XMXDECN4 4개의 요소 float 배열 인수에서 XMXDECN4의 새 instance 초기화합니다.

XMXDECN4 XMXDECN4에 대한 생성자입니다. (오버로드 1/2)
XMXDECN4 정규화된 네 개의 부동 소수점에서 XMXDECN4의 새 instance 초기화합니다.
XMXDECN4 압축된 형식의 구성 요소 데이터를 포함하는 uint32_t 변수에서 XMXDECN4의 새 instance 초기화합니다.
XMXDECN4 XMXDECN4에 대한 생성자입니다. (오버로드 2/2)

구조체

BoundingBox 경계 축 맞춤 개체입니다.
BoundingFrustum 경계 frustum 개체입니다.
BoundingOrientedBox 지향 경계 상자 개체입니다.
BoundingSphere 경계 구 개체입니다.
XMBYTE2 각 구성 요소가 부호 있는 정수인 8비트(1 바이트) 길이의 2D 벡터입니다.
XMBYTE2 각 구성 요소가 부호 있는 정수인 8비트(1 바이트) 길이의 2D 벡터입니다.
XMBYTE4

각 구성 요소가 부호 있는 정수인 8비트(1 바이트) 길이의 4D 벡터입니다.
XMBYTE4 각 구성 요소가 부호 있는 정수인 8비트(1 바이트) 길이의 4D 벡터입니다.
XMBYTEN2 부호 있는 정규화된 값을 부호 있는 8비트(1 바이트) 정수로 저장하기 위한 2D 벡터입니다.
XMBYTEN2 부호 있는 정규화된 값을 부호 있는 8비트(1 바이트) 정수로 저장하기 위한 2D 벡터입니다.
XMBYTEN4 부호 있는 정규화된 값을 부호 있는 8비트(1 바이트) 정수로 저장하기 위한 3D 벡터입니다.
XMBYTEN4 부호 있는 정규화된 값을 부호 있는 8비트(1 바이트) 정수로 저장하기 위한 3D 벡터입니다.
XMCOLOR 각 색 채널이 부호 없는 8비트 정수로 지정되는 32비트 ARGB(알파 레드 그린 블루) 색 벡터입니다.
XMCOLOR 각 색 채널이 부호 없는 8비트 정수로 지정되는 32비트 ARGB(알파 레드 그린 블루) 색 벡터입니다.
XMDEC4 x-,y-및 z- 구성 요소가 10비트 부호 있는 정수 값으로 표시되고 w 구성 요소가 부호 있는 2비트 정수 값으로 표시되는 4D 벡터입니다.
XMDEC4 x-,y-및 z- 구성 요소가 10비트 부호 있는 정수 값으로 표시되고 w 구성 요소가 부호 있는 2비트 정수 값으로 표시되는 4D 벡터입니다.
XMDECN4 부호 있는 정규화된 값을 10비트 부호 있는 x-,y-및 z- 구성 요소와 2비트 부호 있는 w 구성 요소로 저장하기 위한 4D 벡터입니다.
XMDECN4 부호 있는 정규화된 값을 10비트 부호 있는 x-,y-및 z- 구성 요소와 2비트 부호 있는 w 구성 요소로 저장하기 위한 4D 벡터입니다.

XMFLOAT2 두 개의 단정밀도 부동 소수점 값으로 구성된 2D 벡터입니다.
XMFLOAT2A 16비트 경계에 정렬된 XMFLOAT2 구조체에 대해 설명합니다.
XMFLOAT3 세 개의 단정밀도 부동 소수점 값으로 구성된 3D 벡터를 설명합니다.
XMFLOAT3A 16비트 경계에 정렬된 XMFLOAT3 구조체에 대해 설명합니다.
XMFLOAT3PK X 및 Y 구성 요소가 11비트 부동 소수점 숫자로 저장되고 Z 구성 요소가 10비트 부동 소수점 값으로 저장된 3D 벡터를 설명합니다.
XMFLOAT3PK X 및 Y 구성 요소가 11비트 부동 소수점 숫자로 저장되고 Z 구성 요소가 10비트 부동 소수점 값으로 저장된 3D 벡터를 설명합니다.
XMFLOAT3SE 각각 동일한 5비트 지수를 공유하는 9비트 가수가 있는 3개의 부동 소수점 구성 요소로 구성된 3D 벡터를 설명합니다.
XMFLOAT3SE 각각 동일한 5비트 지수를 공유하는 9비트 가수가 있는 3개의 부동 소수점 구성 요소로 구성된 3D 벡터를 설명합니다.
XMFLOAT3X3 3x3 부동 소수점 행렬입니다.
XMFLOAT3X4 32비트 부동 소수점 구성 요소를 포함하는 3x4 열 주 행렬입니다.
XMFLOAT3X4A 16비트 경계에 맞춰 정렬된 32비트 부동 소수점 구성 요소를 포함하는 3x4 열 주 행렬입니다.

XMFLOAT4 4개의 단정밀도 부동 소수점 값으로 구성된 4D 벡터에 대해 설명합니다.
XMFLOAT4A 16 바이트 경계에 맞춰진 XMFLOAT4 구조체에 대해 설명합니다.
XMFLOAT4X3 4*3 부동 소수점 행렬입니다.
XMFLOAT4X3A 16비트 경계에 맞춰진 XMFLOAT4X3 구조체에 대해 설명합니다.
XMFLOAT4X4 4*4 부동 소수점 행렬입니다.
XMFLOAT4X4A 16 바이트 경계에 맞춰진 XMFLOAT4X4 구조체에 대해 설명합니다.
XMHALF2 2개의 반정밀도(16비트) 부동 소수점 값으로 구성된 2D 벡터입니다.
XMHALF2 2개의 반정밀도(16비트) 부동 소수점 값으로 구성된 2D 벡터입니다.
XMHALF4 4개의 반정밀도(16비트) 부동 소수점 값으로 구성된 4D 벡터에 대해 설명합니다.
XMHALF4 4개의 반정밀도(16비트) 부동 소수점 값으로 구성된 4D 벡터에 대해 설명합니다.
XMINT2 각 구성 요소가 부록 정수인 2D 벡터입니다.
XMINT3 각 구성 요소가 부록 정수인 3D 벡터입니다.
XMINT4

각 구성 요소가 부록 정수인 4D 벡터입니다.
XMMATRIX 4개의 하드웨어 벡터 레지스터에 매핑되는 16비트 경계에 맞춰진 4*4 행렬을 설명합니다.
XMSHORT2 부록 및 정규화된 16비트 정수 구성 요소로 구성된 2D 벡터에 대해 설명합니다.
XMSHORT2 부록 및 정규화된 16비트 정수 구성 요소로 구성된 2D 벡터에 대해 설명합니다.
XMSHORT4 16비트 부록 정수 구성 요소로 구성된 4D 벡터입니다.
XMSHORT4 16비트 부록 정수 구성 요소로 구성된 4D 벡터입니다.
XMSHORTN2 부호 있는 정규화된 값을 부호 있는 16비트 정수(int16_t 형식)로 저장하기 위한 2D 벡터입니다.
XMSHORTN2 부호 있는 정규화된 값을 부호 있는 16비트 정수(int16_t 형식)로 저장하기 위한 2D 벡터입니다.
XMSHORTN4 부호 있는 정규화된 값을 부호 있는 16비트 정수(int16_t 형식)로 저장하기 위한 4D 벡터입니다.
XMSHORTN4 부호 있는 정규화된 값을 부호 있는 16비트 정수(int16_t 형식)로 저장하기 위한 4D 벡터입니다.
XMU555 x,y-및 z-구성 요소가 5비트 부호 없는 정수 값으로 표현되고 w-component가 1비트 정수 값으로 표시되는 4D 벡터입니다.
XMU555 x,y-및 z-구성 요소가 5비트 부호 없는 정수 값으로 표현되고 w-component가 1비트 정수 값으로 표시되는 4D 벡터입니다.
XMU565

x-및 z-구성 요소가 5비트 부호 없는 정수 값으로 표현되고 y- 구성 요소가 6비트 부호 없는 정수 값으로 표시되는 3D 벡터입니다.
XMU565 x-및 z-구성 요소가 5비트 부호 없는 정수 값으로 표현되고 y- 구성 요소가 6비트 부호 없는 정수 값으로 표시되는 3D 벡터입니다.
XMUBYTE2 각 구성 요소가 부호 없는 정수인 8비트(1 바이트)인 2D 벡터를 설명합니다.
XMUBYTE2 각 구성 요소가 부호 없는 정수인 8비트(1 바이트)인 2D 벡터를 설명합니다.
XMUBYTE4 각 구성 요소가 부호 없는 정수인 8비트(1 바이트)인 4D 벡터를 설명합니다.
XMUBYTE4 각 구성 요소가 부호 없는 정수인 8비트(1 바이트)인 4D 벡터를 설명합니다.
XMUBYTEN2 부호 없는 정규화된 값을 부호 있는 8비트(1 바이트) 정수로 저장하기 위한 2D 벡터입니다.
XMUBYTEN2 부호 없는 정규화된 값을 부호 있는 8비트(1 바이트) 정수로 저장하기 위한 2D 벡터입니다.
XMUBYTEN4 부호 없는 정규화된 값을 부호 있는 8비트(1 바이트) 정수로 저장하기 위한 3D 벡터입니다.
XMUBYTEN4 부호 없는 정규화된 값을 부호 있는 8비트(1 바이트) 정수로 저장하기 위한 3D 벡터입니다.
XMUDEC4 x,y-및 z-구성 요소가 10비트 부호 없는 정수 값으로 표현되고 w-component가 2비트 부호 없는 정수 값으로 표시되는 4D 벡터입니다.
XMUDEC4 x,y-및 z-구성 요소가 10비트 부호 없는 정수 값으로 표현되고 w-component가 2비트 부호 없는 정수 값으로 표시되는 4D 벡터입니다.

XMUDECN4 부호 없는 정규화된 정수 값을 10비트 부호 없는 x-, y 및 z 구성 요소와 2비트 부호 없는 w-component로 저장하기 위한 4D 벡터입니다.
XMUDECN4 부호 없는 정규화된 정수 값을 10비트 부호 없는 x-, y 및 z 구성 요소와 2비트 부호 없는 w-component로 저장하기 위한 4D 벡터입니다.
XMUINT2 각 구성 요소가 부호 없는 정수인 2D 벡터입니다.
XMUINT3 각 구성 요소가 부호 없는 정수인 3D 벡터입니다.
XMUINT4 각 구성 요소가 부호 없는 정수인 4D 벡터입니다.
XMUNIBBLE4 부호 없는 4비트 정수 구성 요소 4개가 있는 4D 벡터입니다.
XMUNIBBLE4 부호 없는 4비트 정수 구성 요소 4개가 있는 4D 벡터입니다.
XMUSHORT2 16비트 부호 없는 정수 구성 요소로 구성된 2D 벡터에 대해 설명합니다.
XMUSHORT2 16비트 부호 없는 정수 구성 요소로 구성된 2D 벡터에 대해 설명합니다.
XMUSHORT4 16비트 부호 없는 정수 구성 요소로 구성된 4D 벡터입니다.
XMUSHORT4 16비트 부호 없는 정수 구성 요소로 구성된 4D 벡터입니다.
XMUSHORTN2 부호 없는 정규화된 값을 부호 없는 16비트 정수(uint16_t 형식)로 저장하기 위한 2D 벡터입니다.

XMUSHORTN2 부호 없는 정규화된 값을 부호 없는 16비트 정수(uint16_t 형식)로 저장하기 위한 2D 벡터입니다.
XMUSHORTN4 부호 있는 16비트 정수(uint16_t 형식)로 부호 없는 정규화된 값을 저장하기 위한 4D 벡터입니다.
XMUSHORTN4 부호 있는 16비트 정수(uint16_t 형식)로 부호 없는 정규화된 값을 저장하기 위한 4D 벡터입니다.
XMXDEC4 x-,y-및 z- 구성 요소가 10비트 부호 있는 정수 값으로 표현되고 w-component가 부호 없는 2비트 정수 값으로 표시되는 4D 벡터입니다.
XMXDEC4 x-,y-및 z- 구성 요소가 10비트 부호 있는 정수 값으로 표현되고 w-component가 부호 없는 2비트 정수 값으로 표시되는 4D 벡터입니다.
XMXDECN4 부호 있는 정규화된 값을 10비트 부호 있는 x-,y-및 z- 구성 요소로 저장하고 부호 없는 정규화된 값을 2비트 부호 없는 w-component로 저장하기 위한 4D 벡터입니다.
XMXDECN4 부호 있는 정규화된 값을 10비트 부호 있는 x-,y-및 z- 구성 요소로 저장하고 부호 없는 정규화된 값을 2비트 부호 없는 w-component로 저장하기 위한 4D 벡터입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

디바이스 참조 표시

아티클2023. 03. 09.

디스플레이 디바이스 참조 기술에 대한 개요입니다.

디스플레이 디바이스 참조를 개발하려면 다음 헤더가 필요합니다.

- cloneviewhelper.h
- ddkernel.h
- ddkmapi.h
- ddrawi.h
- ddrawint.h
- dmemmgr.h
- dvp.h
- dxmini.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 디바이스 참조 표시

열거형

DISPLAYCONFIG_DEVICE_INFO_TYPE DISPLAYCONFIG_DEVICE_INFO_TYPE 열거형은 DisplayConfigSetDeviceInfo 또는 DisplayConfigGetDeviceInfo 함수를 통해 구성하거나 가져올 표시 디바이스 정보의 유형을 지정합니다.
DISPLAYCONFIG_MODE_INFO_TYPE DISPLAYCONFIG_MODE_INFO_TYPE 열거형은 DISPLAYCONFIG_MODE_INFO 구조에 포함된 정보가 원본 또는 대상 모드임을 지정합니다.
DISPLAYCONFIG_PIXELFORMAT DISPLAYCONFIG_PIXELFORMAT 열거형은 BPP(픽셀당 다양한 비트) 값의 픽셀 형식을 지정합니다.
DISPLAYCONFIG_ROTATION DISPLAYCONFIG_ROTATION 열거형은 디스플레이의 시계 방향 회전을 지정합니다.

DISPLAYCONFIG_SCALING DISPLAYCONFIG_SCALING 열거형은 비디오 프레젠테이션 네트워크(VidPN) 현재 경로에 표시되는 콘텐츠에 적용되는 크기 조정 변환을 지정합니다.
DISPLAYCONFIG_SCANLINE_ORDERING DISPLAYCONFIG_SCANLINE_ORDERING 열거형은 디스플레이가 화면에 이미지를 만드는 데 사용하는 메서드를 지정합니다.
DISPLAYCONFIG_TOPOLOGY_ID DISPLAYCONFIG_TOPOLOGY_ID 열거형은 표시 토폴로지의 유형을 지정합니다.
DISPLAYCONFIG_VIDEO_OUTPUT_TECHNOLOGY DISPLAYCONFIG_VIDEO_OUTPUT_TECHNOLOGY 열거형은 대상의 커넥터 유형을 지정합니다.

Functions

BRUSHOBJ_hGetColorTransform BRUSHOBJ_hGetColorTransform 함수는 지정된 브러시의 색 변환을 검색합니다.
BRUSHOBJ_pvAllocRbrush BRUSHOBJ_pvAllocRbrush 함수는 드라이버가 지정된 브러시를 실현하기 위해 메모리를 할당합니다.
BRUSHOBJ_pvGetRbrush BRUSHOBJ_pvGetRbrush 함수는 드라이버의 지정된 브러시 실현에 대한 포인터를 검색합니다.
BRUSHOBJ_ulGetBrushColor BRUSHOBJ_ulGetBrushColor 함수는 지정된 단색 브러시의 RGB 색을 반환합니다.
CLIPOBJ_bEnum CLIPOBJ_bEnum 함수는 지정된 클립 영역에서 사각형 일괄 처리를 열거합니다. CLIPOBJ_cEnumStart 대한 이전 호출은 열거 순서를 결정합니다.

CLIPOBJ_cEnumStart CLIPOBJ_cEnumStart 함수는 지정된 클립 영역에서 사각형을 열거하기 위한 매개 변수를 설정합니다.
CLIPOBJ_ppoGetPath CLIPOBJ_ppoGetPath 함수는 지정된 클립 영역의 윤곽선을 포함하는 PATHOBJ 구조를 만듭니다.
DisplayConfigGetDeviceInfo DisplayConfigGetDeviceInfo 함수는 디바이스에 대한 표시 구성 정보를 검색합니다.
DisplayConfigSetDeviceInfo DisplayConfigSetDeviceInfo 함수는 대상의 속성을 설정합니다.
DrvAlphaBlend DrvAlphaBlend 함수는 알파 혼합을 사용하여 비트 블록 전송 기능을 제공합니다.
DrvAssertMode DrvAssertMode 함수는 지정된 물리적 디바이스의 모드를 PDEV가 초기화될 때 지정된 모드 또는 하드웨어의 기본 모드로 설정합니다.
DrvBitBlt DrvBitBlt 함수는 디바이스 관리 표면 간, GDI 관리 표준 형식 비트맵 간 또는 디바이스 관리형 표면과 GDI 관리 표준 형식 비트맵 간에 일반적인 비트 블록 전송 기능을 제공합니다.
DrvCompletePDEV DrvCompletePDEV 함수는 생성되는 물리적 디바이스의 GDI 핸들을 저장합니다.
DrvCopyBits DrvCopyBits 함수는 디바이스 관리 래스터 표면과 GDI 표준 형식 비트맵 간에 변환됩니다.
DrvCreateDeviceBitmap DrvCreateDeviceBitmap 함수는 비트맵을 만들고 관리합니다.
DrvDeleteDeviceBitmap DrvDeleteDeviceBitmap 함수는 DrvCreateDeviceBitmap에서 만든 디바이스 비트맵을 삭제합니다.

DrvDeriveSurface DrvDeriveSurface 함수는 지정된 DirectDraw 표면에서 GDI 표면을 파생합니다.
DrvDescribePixelFormat DrvDescribePixelFormat 함수는 PIXELFORMATDESCRIPTOR 구조에 픽셀 형식 설명을 작성하여 디바이스에서 지정한 PDEV의 픽셀 형식을 설명합니다.
DrvDestroyFont DrvDestroyFont 함수는 글꼴 실현이 더 이상 필요하지 않으며 이제 드라이버가 할당한 연결된 모든 데이터 구조를 해제할 수 있음을 드라이버에 알깁니다.
DrvDisableDirectDraw DrvDisableDirectDraw 함수는 DirectDraw 사용을 위해 하드웨어를 사용하지 않도록 설정합니다.
DrvDisableDriver DrvDisableDriver 함수는 GDI에서 드라이버가 더 이상 필요하지 않으며 언로드할 준비가 되었음을 드라이버에 알리는 데 사용됩니다.
DrvDisablePDEV DrvDisablePDEV 함수는 지정된 PDEV가 더 이상 필요하지 않음을 드라이버에 알리기 위해 GDI에서 사용됩니다.
DrvDisableSurface DrvDisableSurface 함수는 GDI에서 현재 디바이스에 대해 DrvEnableSurface에서 만든 표면이 더 이상 필요하지 않음을 드라이버에 알리는 데 사용됩니다.
DrvDitherColor DrvDitherColor 함수는 디바이스 팔레트에 대해 디더링된 브러시를 만들도록 디바이스에 요청합니다.
DrvDrawEscape DrvDrawEscape 함수는 둘 이상의 함수 호출을 제공하는 진입점입니다. 특정 함수는 iEsc 매개 변수의 값에 따라 달라집니다.
DrvEnableDirectDraw DrvEnableDirectDraw 함수를 사용하면 하드웨어에서 DirectDraw를 사용할 수 있습니다.

DrvEnableDriver DrvEnableDriver 함수는 드라이버 DLL에서 내보낸 초기 드라이버 진입점입니다.
DrvEnablePDEV DrvEnablePDEV 함수는 물리적 디바이스의 특성에 대한 설명을 GDI에 반환합니다.
DrvEnableSurface DrvEnableSurface 함수는 그릴 표면을 설정하고 지정된 물리적 디바이스와 연결합니다.
DrvEndDoc DrvEndDoc 함수는 렌더링을 위해 드라이버에 문서 보내기를 마쳤을 때 GDI에서 호출됩니다.
DrvEscape DrvEscape 함수는 디바이스 독립적 디바이스 드라이버 인터페이스에서 사용할 수 없는 디바이스에서 정보를 검색하는 데 사용됩니다. 특정 쿼리는 iEsc 매개 변수의 값에 따라 달라집니다.
DrvFillPath DrvFillPath 함수는 닫힌 경로의 채우기를 처리하는 선택적 진입점입니다.
DrvFontManagement DrvFontManagement 함수는 PostScript 디바이스에 제공되는 선택적 진입점입니다.
DrvFree DrvFree 함수는 지정된 구조가 더 이상 필요하지 않음을 드라이버에 알리는 데 사용됩니다.
DrvGetDirectDrawInfo DrvGetDirectDrawInfo 함수는 그래픽 하드웨어의 기능을 반환합니다.
DrvGetGlyphMode DrvGetGlyphMode 함수는 GDI에 문자 모양 정보를 캐시하는 방법을 알려줍니다.
DrvGetModes DrvGetModes 함수는 지정된 디바이스에서 지원하는 모드를 나열합니다.
DrvGetTrueTypeFile DrvGetTrueTypeFile 함수는 메모리 매핑 TrueType 글꼴 파일에 액세스합니다.

DrvGradientFill DrvGradientFill 함수는 지정된 기본 형식을 음영으로 지정합니다.
DrvIcmCheckBitmapBits DrvIcmCheckBitmapBits 함수는 지정된 비트맵의 픽셀이 지정된 변환의 디바이스 영역 내에 있는지 여부를 확인합니다.
DrvIcmCreateColorTransform DrvIcmCreateColorTransform 함수는 ICM 색 변환을 만듭니다.
DrvIcmDeleteColorTransform DrvIcmDeleteColorTransform 함수는 지정된 색 변환을 삭제합니다.
DrvIcmSetDeviceGammaRamp DrvIcmSetDeviceGammaRamp 함수는 지정된 디스플레이 디바이스의 하드웨어 감마 램프를 설정합니다.
DrvLineTo DrvLineTo 함수는 단일 단색 정수 전용 화장품 선을 그립니다.
DrvLoadFontFile DrvLoadFontFile 함수는 글꼴 파일 로드 및 매핑과 관련된 정보를 GDI로부터 받습니다.
DrvMovePointer DrvMovePointer 함수는 포인터를 새 위치로 이동하고 GDI가 포인터 표시를 방해하지 않도록 합니다.
DrvNextBand DrvNextBand 함수는 물리적 페이지에 대한 밴드 그리기를 마쳤을 때 GDI에 의해 호출되므로 드라이버는 다음 밴드를 프린터로 보낼 수 있습니다.
DrvNotify DrvNotify 함수를 사용하면 디스플레이 드라이버에 GDI의 특정 정보에 대한 알림을 받을 수 있습니다.

DrvPaint DrvPaint 함수는 사용되지 않으며 더 이상 Windows 2000 이상에서 GDI에 의해 호출되지 않습니다. 새 드라이버는 하나 이상의 DrvFillPath, DrvStrokePath 또는 DrvStrokeAndFillPath를 구현해야 합니다.
DrvPlgBlt DrvPlgBlt 함수는 디바이스 관리형 표면과 GDI 관리형 표면의 조합 간에 비트 블록 전송 회전 기능을 제공합니다.
DrvQueryAdvanceWidths DrvQueryAdvanceWidths 함수는 지정된 문자 모양 집합에 대한 문자 이동 너비를 반환합니다.
DrvQueryDeviceSupport DrvQueryDeviceSupport 함수는 요청된 디바이스 관련 정보를 반환합니다.
DrvQueryDriverInfo DrvQueryDriverInfo 함수는 요청된 드라이버 관련 정보를 반환합니다.
DrvQueryFont DrvQueryFont 함수는 GDI에서 지정된 글꼴에 대한 IFIMETRICS 구조를 가져오는 데 사용됩니다.
DrvQueryFontCaps DrvQueryFontCaps 함수는 글꼴 드라이버의 기능을 정의합니다.
DrvQueryFontData DrvQueryFontData 함수는 실현된 글꼴에 대한 정보를 검색합니다.
DrvQueryFontFile DrvQueryFontFile 함수는 글꼴 파일 정보를 제공합니다.
DrvQueryFontTree DrvQueryFontTree 함수는 GDI에 다음 중 하나를 정의하는 구조체에 대한 포인터를 제공합니다. 유니코드에서 문자 모양 핸들로의 매핑(glyph variantsA 커닝 핸들에 대한 커닝 쌍 매핑 포함)
DrvQueryPerBandInfo 프린터 그래픽 DLL의 DrvQueryPerBandInfo 함수는 물리적 페이지에 대역을 그리기 시작하기 전에 GDI에서 호출되므로 드라이버는 GDI에 대역별 정보를 제공할 수 있습니다.

DrvQueryTrueTypeOutline DrvQueryTrueTypeOutline 함수는 네이티브 TrueType 형식의 문자 모양 윤곽선을 검색합니다.
DrvQueryTrueTypeTable DrvQueryTrueTypeTable 함수는 TrueType 글꼴 설명 파일의 특정 테이블에 액세스합니다.
DrvRealizeBrush DrvRealizeBrush 함수는 드라이버가 지정된 표면에 대해 지정된 브러시를 인식하도록 요청합니다.
DrvResetDevice DrvResetDevice 함수는 작동하지 않거나 응답하지 않는 디바이스를 다시 설정합니다.
DrvResetPDEV DrvResetPDEV 함수를 사용하면 Win32 애플리케이션이 ResetDC를 호출할 때 그래픽 드라이버가 이전 PDEV 구조에서 새 PDEV 구조로 드라이버의 상태를 전송할 수 있습니다.
DrvSaveScreenBits DrvSaveScreenBits 함수를 사용하면 디스플레이 드라이버가 표시된 이미지의 지정된 사각형을 저장하거나 복원합니다.
DrvSendPage 프린터 그래픽 DLL의 DrvSendPage 함수는 실제 페이지 그리기를 마쳤을 때 GDI에서 호출되므로 드라이버가 페이지를 프린터로 보낼 수 있습니다.
DrvSetPalette DrvSetPalette 함수는 드라이버가 지정된 디바이스에 대한 팔레트를 실현할 것을 요청합니다.
DrvSetPixelFormat DrvSetPixelFormat 함수는 창의 픽셀 형식을 설정합니다.
DrvSetPointerShape DrvSetPointerShape 함수는 드라이버가 포인터를 디스플레이에서 끌어온 경우 드라이버에 포인터를 표시에서 제거하도록 요청하는 데 사용됩니다. 새 포인터 셰이프를 설정하려고 합니다. 지정한 위치에 새 포인터를 표시합니다.
DrvStartBanding DrvStartBanding 함수는 렌더링을 위해 물리적 페이지의 밴드를 드라이버에 보낼 준비가 되면 GDI에서 호출됩니다.

DrvStartDoc DrvStartDoc 함수는 렌더링을 위해 드라이버에 문서 보내기를 시작할 준비가 되면 GDI에서 호출됩니다.
DrvStartPage DrvStartPage 함수는 렌더링을 위해 드라이버에 실제 페이지의 콘텐츠를 보낼 준비가 되면 GDI에 의해 호출됩니다.
DrvStretchBlt DrvStretchBlt 함수는 디바이스 관리 표면과 GDI 관리 표면의 조합 간에 확장 비트 블록 전송 기능을 제공합니다.
DrvStretchBltROP DrvStretchBltROP 함수는 ROP를 사용하여 스트레치 비트 블록 전송을 수행합니다.
DrvStrokeAndFillPath DrvStrokeAndFillPath 함수는 경로를 동시에 스트로크(윤곽선)하고 채웁니다.
DrvStrokePath DrvStrokePath 함수는 경로를 스트로크(윤곽선)합니다.
DrvSwapBuffers DrvSwapBuffers 함수는 지정된 표면에 창과 연결된 숨겨진 버퍼의 내용을 표시합니다.
DrvSynchronize DrvSynchronize 함수는 GDI가 디바이스 관리 표면에 액세스해야 한다고 드라이버에 알립니다. 이 함수를 사용하면 디바이스의 공동 프로세서에서 수행하는 비동기 그리기 작업을 GDI 액세스와 조율할 수 있습니다.
DrvSynchronizeSurface DrvSynchronizeSurface 함수는 지정된 표면에 GDI를 작성해야 한다고 드라이버에 알립니다. 이 함수를 사용하면 디바이스의 공동 프로세서에서 수행하는 그리기 작업을 GDI와 조율할 수 있습니다.
DrvTextOut DrvTextOut 함수는 드라이버가 지정된 위치에서 문자 모양 집합을 렌더링하도록 요구하는 GDI의 진입점입니다.

DrvTransparentBlt DrvTransparentBlt 함수는 투명도를 갖춘 비트 블록 전송 기능을 제공합니다.
DrvUnloadFontFile DrvUnloadFontFile 함수는 지정된 글꼴 파일이 더 이상 필요하지 않음을 글꼴 드라이버에 알릴 수 있습니다.
EngAcquireSemaphore EngAcquireSemaphore 함수는 호출 스레드에서 단독 액세스를 위해 세마포와 연결된 리소스를 획득합니다.
EngAllocMem EngAllocMem 함수는 메모리 블록을 할당하고 할당 전에 호출자 제공 태그를 삽입합니다.
EngAllocPrivateUserMem EngAllocPrivateUserMem 함수는 지정된 프로세스의 주소 공간에서 사용자 메모리 블록을 할당하고 할당 전에 호출자 제공 태그를 삽입합니다.
EngAllocUserMem EngAllocUserMem 함수는 현재 프로세스의 주소 공간에서 메모리 블록을 할당하고 할당 전에 호출자 제공 태그를 삽입합니다.
EngAlphaBlend EngAlphaBlend 함수는 알파 혼합이 포함된 비트 블록 전송 기능을 제공합니다.
EngAssociateSurface EngAssociateSurface 함수는 지정된 표면을 지정된 디바이스에 속하는 것으로 표시합니다.
EngBitBlt EngBitBlt 함수는 디바이스 관리 표면 간 또는 디바이스 관리 표면과 GDI 관리 표준 형식 비트맵 간에 일반적인 비트 블록 전송 기능을 제공합니다.
EngBugCheckEx EngBugCheckEx 함수는 호출자가 복구할 수 없는 오류를 발견하여 호출자가 계속 실행될 경우 시스템을 손상시킬 수 있는 제어된 방식으로 시스템을 다운합니다.

EngCheckAbort EngCheckAbort 함수를 사용하면 프린터 그래픽 DLL에서 인쇄 작업을 종료해야 하는지 여부를 결정할 수 있습니다.
EngClearEvent EngClearEvent 함수는 지정된 이벤트 개체를 서명되지 않은 상태로 설정합니다.
EngComputeGlyphSet EngComputeGlyphSet 함수는 디바이스에서 지원되는 문자 모양 집합을 계산합니다.
EngControlSprites EngControlSprites 함수는 지정된 WNDOBJ 영역에서 스프라이트를 아래로 흘리거나 다시 그리습니다.
EngCopyBits EngCopyBits 함수는 디바이스 관리 래스터 표면과 GDI 표준 형식 비트맵 간에 변환됩니다.
EngCreateBitmap EngCreateBitmap 함수는 GDI가 비트맵을 만들고 관리해 줄 것을 요청합니다.
EngCreateClip EngCreateClip 함수는 드라이버가 콜백에 사용하는 CLIPOBJ 구조를 만듭니다.
EngCreateDeviceBitmap EngCreateDeviceBitmap 함수는 GDI에 디바이스 비트맵에 대한 핸들을 만들도록 요청합니다.
EngCreateDeviceSurface EngCreateDeviceSurface 함수는 드라이버가 관리할 디바이스 표면에 대한 핸들을 만들고 반환합니다.
EngCreateDriverObj EngCreateDriverObj 함수는 DRIVEROBJ 구조를 만듭니다.
EngCreateEvent EngCreateEvent 함수는 디스플레이 드라이버와 비디오 미니포트 드라이버 간의 하드웨어 액세스를 동기화하는 데 사용할 수 있는 동기화 이벤트 개체를 만듭니다.

EngCreatePalette EngCreatePalette 함수는 RGB 팔레트를 만들기 위한 요청을 GDI에 보냅니다.
EngCreatePath EngCreatePath 함수는 드라이버의 임시 사용을 위한 경로를 할당합니다.
EngCreateSemaphore EngCreateSemaphore 함수는 세마포 개체를 만듭니다.
EngCreateWnd EngCreateWnd 함수는 hwnd에서 참조하는 창에 대한 WNDOBJ 구조를 만듭니다.
EngDebugBreak EngDebugBreak 함수를 사용하면 현재 프로세스의 중단점이 발생합니다.
EngDebugPrint EngDebugPrint 함수는 지정된 디버그 메시지를 커널 디버거에 출력합니다.
EngDeleteClip EngDeleteClip 함수는 EngCreateClip에 의해 할당된 CLIPOBJ 구조를 삭제합니다.
EngDeleteDriverObj EngDeleteDriverObj 함수는 디바이스 관리 리소스를 추적하는 데 사용되는 핸들을 해제합니다.
EngDeleteEvent EngDeleteEvent 함수는 지정된 이벤트 개체를 삭제합니다.
EngDeleteFile EngDeleteFile 함수는 파일을 삭제합니다.
EngDeletePalette EngDeletePalette 함수는 지정된 색상표를 삭제하라는 요청을 GDI에 보냅니다.
EngDeletePath EngDeletePath 함수는 이전에 EngCreatePath에 의해 할당된 경로를 삭제합니다.

EngDeleteSafeSemaphore EngDeleteSafeSemaphore 함수는 지정된 안전 세마포에 대한 참조를 제거합니다.
EngDeleteSemaphore EngDeleteSemaphore 함수는 시스템의 리소스 목록에서 세마포 개체를 삭제합니다.
EngDeleteSurface EngDeleteSurface 함수는 지정된 표면을 삭제합니다.
EngDeleteWnd EngDeleteWnd 함수는 WNDOBJ 구조를 삭제합니다.
EngDeviceIoControl EngDeviceIoControl 함수는 지정된 비디오 미니포트 드라이버에 제어 코드를 보내 디바이스가 지정된 작업을 수행하도록 합니다.
EngDitherColor EngDitherColor 함수는 지정된 RGB 색과 근사치를 나타내는 표준 8x8 디더를 반환합니다.
EngEnumForms EngEnumForms 함수는 지정된 프린터에서 지원하는 양식을 열거합니다.
EngEraseSurface EngEraseSurface 함수는 GDI를 호출하여 표면을 지웁니다. 표면에 지정된 사각형이 지정된 색으로 채워집니다.
EngFillPath EngFillPath 함수는 경로를 채웁니다.
EngFindImageProcAddress EngFindImageProcAddress 함수는 실행 가능한 모듈 내에서 함수의 주소를 반환합니다.
EngFindResource EngFindResource 함수는 모듈에서 리소스의 위치를 결정합니다.
EngFntCacheAlloc EngFntCacheAlloc 함수는 캐시된 메모리에 저장할 글꼴에 대한 스토리지를 할당합니다.

EngFntCacheFault 글꼴 드라이버에서 글꼴 데이터 캐시를 읽거나 쓰는 동안 오류가 발생하면 EngFntCacheFault 함수는 글꼴 엔진에 오류를 보고합니다.
EngFntCacheLookUp EngFntCacheLookUp 함수는 캐시된 글꼴 파일 데이터의 주소를 검색합니다.
EngFreeMem EngFreeMem 함수는 시스템 메모리 블록을 할당 취소합니다.
EngFreeModule EngFreeModule 함수는 시스템 메모리에서 파일의 매핑을 해제합니다.
EngFreePrivateUserMem EngFreePrivateUserMem 함수는 개인 사용자 메모리 블록을 할당 취소합니다.
EngFreeUserMem EngFreeUserMem 함수는 사용자 메모리 블록을 할당 취소합니다.
EngGetCurrentCodePage EngGetCurrentCodePage 함수는 시스템의 기본 OEM 및 ANSI 코드 페이지를 반환합니다.
EngGetCurrentProcessId EngGetCurrentProcessId 함수는 애플리케이션의 현재 프로세스를 식별합니다.
EngGetCurrentThreadId EngGetCurrentThreadId 함수는 애플리케이션의 현재 스레드를 식별합니다.
EngGetDriverName EngGetDriverName 함수는 드라이버의 DLL 이름을 반환합니다.
EngGetFileChangeTime EngGetFileChangeTime 함수는 파일의 마지막 쓰기 시간을 검색합니다.
EngGetFilePath EngGetFilePath 함수는 지정된 글꼴 파일과 연결된 파일 경로를 결정합니다.

EngGetForm EngGetForm 함수는 지정된 양식에 대한 FORM_INFO_1 세부 정보를 가져옵니다.
EngGetLastError EngGetLastError 함수는 호출 스레드에 대해 GDI에서 기록한 마지막 오류 코드를 반환합니다.
EngGetPrinter EngGetPrinter 함수는 지정된 프린터에 대한 정보를 검색합니다.
EngGetPrinterData EngGetPrinterData 함수는 지정된 프린터에 대한 구성 데이터를 검색합니다.
EngGetPrinterDataFileName EngGetPrinterDataFileName 함수는 프린터 데이터 파일의 문자열 이름을 검색합니다.
EngGetPrinterDriver EngGetPrinterDriver 함수는 지정된 프린터에 대한 드라이버 데이터를 검색합니다.
EngGetProcessHandle EngGetProcessHandle 함수는 현재 클라이언트 프로세스에 대한 핸들을 검색합니다.
EngGetType1FontList EngGetType1FontList 함수는 로컬 및 원격으로 설치된 PostScript 유형 1 글꼴 목록을 검색합니다.
EngGradientFill EngGradientFill 함수는 지정된 기본 형식에 음영을 적용합니다.
EngHangNotification EngHangNotification 함수는 지정된 디바이스가 작동하지 않거나 응답하지 않음을 시스템에 알린다.
EngInitializeSafeSemaphore EngInitializeSafeSemaphore 함수는 지정된 안전 세마포를 초기화합니다.
EngIsSemaphoreOwned EngIsSemaphoreOwned 함수는 스레드가 지정된 세마포를 보유하는지 여부를 결정합니다.

EngIsSemaphoreOwnedByCurrentThread EngIsSemaphoreOwnedByCurrentThread 함수는 현재 실행 중인 스레드가 지정된 세마포를 보유하는지 여부를 결정합니다.
EngLineTo EngLineTo 함수는 단일 단색 정수 전용 화장품 선을 그립니다.
EngLoadImage EngLoadImage 함수는 지정된 실행 파일을 커널 모드 메모리에 로드합니다.
EngLoadModule EngLoadModule 함수는 읽기 위해 지정된 데이터 모듈을 시스템 메모리에 로드합니다.
EngLoadModuleForWrite EngLoadModuleForWrite 함수는 지정된 실행 모듈을 쓰기 위해 시스템 메모리에 로드합니다.
EngLockDirectDrawSurface EngLockDirectDrawSurface 함수는 DirectDraw 표면의 커널 모드 핸들을 잠급니다.
EngLockDriverObj EngLockDriverObj 함수는 호출 스레드에 대해 이 개체에 대한 배타적 잠금을 만듭니다.
EngLockSurface EngLockSurface 함수는 지정된 표면에 대한 사용자 개체를 만듭니다. 이 함수는 드라이버가 만드는 표면에 대한 액세스 권한을 제공합니다.
EngLpkInstalled EngLpkInstalled 함수는 언어 팩이 시스템에 설치되어 있는지 여부를 결정합니다.
EngMapEvent EngMapEvent 함수는 사용자 모드 이벤트 개체를 커널 모드에 매핑합니다.
EngMapFile EngMapFile 함수는 파일을 만들거나 열고 시스템 공간에 매핑합니다.
EngMapFontFile EngMapFontFile 함수는 사용되지 않습니다. 대신 EngMapFontFileFD를 사용합니다.

EngMapFontFileFD 필요한 경우 EngMapFontFileFD 함수는 글꼴 파일을 시스템 메모리에 매핑하고 파일의 글꼴 데이터의 기본 위치에 대한 포인터를 반환합니다.
EngMapModule EngMapModule 함수는 EngLoadModule, EngLoadModuleForWrite, EngLoadImage 또는 EngMapFile에 의해 로드된 파일의 주소와 크기를 반환합니다.
EngMarkBandingSurface EngMarkBandingSurface 함수는 지정된 표면을 밴딩 표면으로 표시합니다.
EngModifySurface EngModifySurface 함수는 드라이버에서 만든 표면의 특성에 대해 GDI에 알립니다.
EngMovePointer EngMovePointer 함수는 디바이스에서 엔진 관리 포인터를 이동합니다.
EngMulDiv EngMulDiv 함수는 두 개의 32비트 값을 곱한 다음 64비트 결과를 세 번째 32비트 값으로 나눕니다.
EngMultiByteToUnicodeN EngMultiByteToUnicodeN 함수는 현재 ANSI 코드 페이지를 사용하여 지정된 ANSI 소스 문자열을 유니코드 문자열로 변환합니다.
EngMultiByteToWideChar EngMultiByteToWideChar 함수는 지정된 코드 페이지를 사용하여 ANSI 소스 문자열을 와이드 문자열로 변환합니다.
EngPaint EngPaint 함수를 사용하면 GDI가 지정된 영역을 그립니다.
EngPlgBlt EngPlgBlt 함수는 GDI가 회전 비트 블록 전송을 수행하도록 합니다.
EngProbeForRead EngProbeForRead 함수는 읽기 접근성을 위해 구조를 검색합니다.

EngProbeForReadAndWrite EngProbeForReadAndWrite 함수는 읽기 및 쓰기 접근성을 위해 구조를 검색합니다.
EngQueryDeviceAttribute EngQueryDeviceAttribute 함수를 사용하면 드라이버가 디바이스의 특정 특성에 대해 시스템을 쿼리할 수 있습니다.
EngQueryFileTimeStamp EngQueryFileTimeStamp 함수는 파일의 타임스탬프를 반환합니다.
EngQueryLocalTime EngQueryLocalTime 함수는 현지 시간을 쿼리합니다.
EngQueryPalette EngQueryPalette 함수는 특성에 대해 지정된 팔레트를 쿼리합니다.
EngQueryPerformanceCounter EngQueryPerformanceCounter 함수는 성능 카운터를 쿼리합니다.
EngQueryPerformanceFrequency EngQueryPerformanceFrequency 함수는 성능 카운터의 빈도를 쿼리합니다.
EngQuerySystemAttribute EngQuerySystemAttribute 함수는 프로세서별 또는 시스템별 기능을 쿼리합니다.
EngReadStateEvent EngReadStateEvent 함수는 지정된 이벤트 개체:_signaled 또는 부호 없는 이벤트의 현재 상태를 반환합니다.
EngReleaseSemaphore EngReleaseSemaphore 함수는 지정된 세마포를 해제합니다.
EngRestoreFloatingPointState 드라이버에서 부동 소수점 또는 MMX 하드웨어 지침을 사용한 후 EngRestoreFloatingPointState 함수는 Windows 2000 이상 커널 부동 소수점 상태를 복원합니다.

EngSaveFloatingPointState EngSaveFloatingPointState 함수는 현재 Windows 2000 이상 커널 부동 소수점 상태를 저장합니다.
EngSecureMem EngSecureMem 함수는 메모리에서 지정된 주소 범위를 잠급니다.
EngSetEvent EngSetEvent 함수는 지정된 이벤트 개체를 신호 상태로 설정하고 이벤트 개체의 이전 상태를 반환합니다.
EngSetLastError EngSetLastError 함수를 사용하면 GDI가 애플리케이션에서 검색할 수 있는 오류 코드를 보고합니다.
EngSetPointerShape EngSetPointerShape 함수는 호출 드라이버에 대한 포인터 셰이프를 설정합니다.
EngSetPointerTag EngSetPointerTag 함수는 Windows 2000 이상 운영 체제 버전에서는 사용되지 않습니다.
EngSetPrinterData EngSetPrinterData 함수는 Windows 2000 이상에서 사용되지 않습니다. 이전 버전의 Windows EngSetPrinterData에서는 지정된 프린터의 구성 데이터를 설정합니다.
EngSort EngSort 함수는 지정된 목록에서 빠른 정렬을 수행합니다.
EngStretchBlt EngStretchBlt 함수는 GDI가 확장 비트 블록 전송을 수행하도록 합니다.
EngStretchBltROP EngStretchBltROP 함수는 ROP를 사용하여 스트레치 비트 블록 전송을 수행합니다.
EngStrokeAndFillPath EngStrokeAndFillPath 함수는 GDI가 경로를 채우고 동시에 스트로크하도록 합니다.

EngStrokePath EngStrokePath 함수는 GDI가 지정된 경로를 스트로크할 것을 요청합니다.
EngTextOut EngTextOut 함수를 사용하면 GDI가 지정된 위치에서 문자 모양 집합을 렌더링합니다.
EngTransparentBlt EngTransparentBlt 함수는 투명도를 갖춘 비트 블록 전송 기능을 제공합니다.
EngUnicodeToMultiByteN EngUnicodeToMultiByteN 함수는 현재 ANSI 코드 페이지를 사용하여 지정된 유니코드 문자열을 ANSI 문자열로 변환합니다.
EngUnloadImage EngUnloadImage 함수는 EngLoadImage에서 로드한 이미지를 언로드합니다.
EngUnlockDirectDrawSurface EngUnlockDirectDrawSurface 함수는 지정된 표면에서 잠금을 해제합니다.
EngUnlockDriverObj EngUnlockDriverObj 함수를 사용하면 GDI가 드라이버 개체의 잠금을 해제합니다.
EngUnlockSurface EngUnlockSurface 함수는 GDI가 표면의 잠금을 해제하도록 합니다.
EngUnmapEvent EngUnmapEvent 함수는 매핑된 사용자 모드 이벤트에 할당된 커널 모드 리소스를 정리합니다.
EngUnmapFile EngUnmapFile 함수는 시스템 공간에서 파일 뷰의 매핑을 해제합니다.
EngUnmapFontFile EngUnmapFontFile 함수는 사용되지 않습니다. 대신 EngUnmapFontFileFD를 사용합니다.
EngUnmapFontFileFD EngUnmapFontFileFD 함수는 시스템 메모리에서 지정된 글꼴 파일의 매핑을 해제합니다.

EngUnsecureMem EngUnsecureMem 함수는 메모리에 잠긴 주소 범위를 잠금 해제합니다.
EngWaitForSingleObject EngWaitForSingleObject 함수는 지정된 이벤트 개체가 신호 상태로 설정되거나 대기 시간이 초과될 때까지 디스플레이 드라이버의 현재 스레드를 대기 상태로 전환합니다.
EngWideCharToMultiByte EngWideCharToMultiByte 함수는 지정된 코드 페이지를 사용하여 와이드 문자열을 ANSI 소스 문자열로 변환합니다.
EngWritePrinter EngWritePrinter 함수를 사용하면 프린터 그래픽 DLL이 데이터 스트림을 프린터 하드웨어로 보낼 수 있습니다.
FLOATOBJ_Add FLOATOBJ_Add 함수는 두 FLOATOBJ를 추가하고 첫 번째 매개 변수의 결과와 함께 반환합니다.
FLOATOBJ_AddFloat FLOATOBJ_AddFloat 함수는 FLOATL 형식의 값을 FLOATOBJ에 추가하고 첫 번째 매개 변수의 결과와 함께 반환합니다.
FLOATOBJ_AddLong FLOATOBJ_AddLong 함수는 FLOATOBJ에 LONG 형식의 값을 추가하고 첫 번째 매개 변수의 결과와 함께 반환합니다.
FLOATOBJ_Div FLOATOBJ_Div 함수는 두 FLOATOBJ를 나누고 첫 번째 매개 변수의 결과와 함께 반환합니다.
FLOATOBJ_DivFloat FLOATOBJ_DivFloat 함수는 FLOATOBJ를 FLOATL 형식의 값으로 나누고 첫 번째 매개 변수의 결과와 함께 반환합니다.
FLOATOBJ_DivLong FLOATOBJ_DivLong 함수는 FLOATOBJ를 LONG 형식의 값으로 나누고 첫 번째 매개 변수의 결과와 함께 반환합니다.

FLOATOBJ_Equal FLOATOBJ_Equal 함수는 두 FLOATOBJ가 같은지 여부를 결정합니다.
FLOATOBJ_EqualLong FLOATOBJ_EqualLong 함수는 FLOATOBJ와 LONG 형식의 값이 같은지 여부를 결정합니다.
FLOATOBJ_GetFloat FLOATOBJ_GetFloat 함수는 지정된 FLOATOBJ의 FLOAT 등가 값을 계산하고 반환합니다.
FLOATOBJ_GetLong FLOATOBJ_GetLong 함수는 지정된 FLOATOBJ의 LONG과 동등한 값을 계산하고 반환합니다.
FLOATOBJ_GreaterThan FLOATOBJ_GreaterThan 함수는 첫 번째 FLOATOBJ가 두 번째 FLOATOBJ보다 큰지 여부를 결정합니다.
FLOATOBJ_GreaterThanLong FLOATOBJ_GreaterThanLong 함수는 FLOATOBJ가 LONG 형식의 값보다 큰지 여부를 결정합니다.
FLOATOBJ_LessThan FLOATOBJ_LessThan 함수는 첫 번째 FLOATOBJ가 두 번째 FLOATOBJ보다 작은지 여부를 결정합니다.
FLOATOBJ_LessThanLong FLOATOBJ_LessThanLong 함수는 FLOATOBJ가 LONG 형식의 값보다 작은지 여부를 결정합니다.
FLOATOBJ_Mul FLOATOBJ_Mul 함수는 두 FLOATOBJ를 곱하고 첫 번째 매개 변수의 결과와 함께 반환합니다.
FLOATOBJ_MulFloat FLOATOBJ_MulFloat 함수는 FLOATOBJ를 FLOATL 형식의 값으로 곱하고 첫 번째 매개 변수의 결과와 함께 반환합니다.
FLOATOBJ_MulLong FLOATOBJ_MulLong 함수는 FLOATOBJ를 LONG 형식의 값으로 곱하고 첫 번째 매개 변수의 결과와 함께 반환합니다.

FLOATOBJ_Neg FLOATOBJ_Neg 함수는 FLOATOBJ를 부정합니다.
FLOATOBJ_SetFloat FLOATOBJ_SetFloat 함수는 FLOATL 형식의 값을 FLOATOBJ에 할당합니다.
FLOATOBJ_SetLong FLOATOBJ_SetLong 함수는 FLOATOBJ에 LONG 형식의 값을 할당합니다.
FLOATOBJ_Sub FLOATOBJ_Sub 함수는 첫 번째에서 두 번째 FLOATOBJ를 빼고 첫 번째 매개 변수의 결과와 함께 반환합니다.
FLOATOBJ_SubFloat FLOATOBJ_SubFloat 함수는 FLOATOBJ에서 FLOATL 형식의 값을 빼고 첫 번째 매개 변수의 결과와 함께 반환합니다.
FLOATOBJ_SubLong FLOATOBJ_SubLong 함수는 FLOATOBJ에서 LONG 형식의 값을 빼고 첫 번째 매개 변수의 결과와 함께 반환합니다.
FONTOBJ_cGetAllGlyphHandles FONTOBJ_cGetAllGlyphHandles 함수를 사용하면 디바이스 드라이버가 GDI 글꼴의 모든 문자 모양 핸들을 찾을 수 있습니다.
FONTOBJ_cGetGlyphs FONTOBJ_cGetGlyphs 함수는 문자 모양 핸들을 문자 모양 데이터에 대한 포인터로 변환하는 글꼴 소비자에 대한 서비스로, FONTOBJ_cGetGlyphs 다음 호출까지 유효합니다.
FONTOBJ_pfdg FONTOBJ_pfdg 함수는 지정된 글꼴과 연결된 FD_GLYPHSET 구조체에 대한 포인터를 검색합니다.
FONTOBJ_pifi FONTOBJ_pifi 함수는 지정된 글꼴과 연결된 IFIMETRICS 구조체에 대한 포인터를 검색합니다.
FONTOBJ_pjOpenTypeTablePointer FONTOBJ_pjOpenTypeTablePointer 함수는 OpenType 테이블의 뷰에 대한 포인터를 반환합니다.

FONTOBJ_pQueryGlyphAttrs FONTOBJ_pQueryGlyphAttrs 함수는 글꼴의 문자 모양에 대한 정보를 반환합니다.
FONTOBJ_pvTrueTypeFontFile FONTOBJ_pvTrueTypeFontFile 함수는 TrueType, OpenType 또는 Type1 글꼴 파일의 보기에 대한 사용자 모드 포인터를 검색합니다.
FONTOBJ_pwszFontFilePaths FONTOBJ_pwszFontFilePaths 함수는 글꼴과 연결된 파일 경로를 검색합니다.
FONTOBJ_pxoGetXform FONTOBJ_pxoGetXform 함수는 지정된 글꼴에 대한 개념-디바이스 변환을 검색합니다.
FONTOBJ_vGetInfo FONTOBJ_vGetInfo 함수는 연결된 글꼴에 대한 정보를 검색합니다.
GetCaps IDirectDrawKernel::GetCaps 메서드는 이 커널 모드 디바이스의 기능을 반환합니다.
GetDisplayConfigBufferSizes GetDisplayConfigBufferSizes 함수는 QueryDisplayConfig 함수를 호출하는 데 필요한 버퍼의 크기를 검색합니다.
GetKernelHandle IDirectDrawKernel::GetKernelHandle 메서드는 DirectDraw 개체에 커널 모드 핸들을 반환합니다.
GetKernelHandle IDirectDrawSurfaceKernel::GetKernelHandle 메서드는 DirectDraw 화면에 커널 모드 핸들을 반환합니다.
HeapVidMemAllocAligned HeapVidMemAllocAligned 함수는 DirectDraw 비디오 메모리 힙 관리자를 사용하여 디스플레이 드라이버에 대한 off_screen_memory 할당합니다.
HT_ComputeRGBGammaTable HT_ComputeRGBGammaTable 함수를 사용하면 GDI가 감마 번호에 따라 디바이스 빨강, 녹색 및 파랑 강도를 계산합니다.

HT_Get8BPPFormatPalette HT_Get8BPPFormatPalette 함수는 픽셀 디바이스 유형당 표준 8비트에서 사용할 하프톤 팔레트를 반환합니다.
HT_Get8BPPMaskPalette HT_Get8BPPMaskPalette 함수는 픽셀당 8비트 디바이스 유형에 대한 마스크 팔레트를 반환합니다.
HTUI_DeviceColorAdjustment 그래픽 디바이스 드라이버에서 HTUI_DeviceColorAdjustment 함수를 사용하여 사용자가 디바이스의 하프톤 속성을 조정할 수 있는 대화 상자를 표시할 수 있습니다.
LPDD_NOTIFYCALLBACK NotifyCallback 콜백 함수는 발생한 이벤트와 관련된 작업을 수행합니다.
PALOBJ_cGetColors PALOBJ_cGetColors 함수는 인덱싱된 팔레트에서 RGB 색을 복사합니다.
PATHOBJ_bCloseFigure PATHOBJ_bCloseFigure 함수는 현재 위치에서 그림의 첫 번째 지점까지 선을 그려 경로의 열린 그림을 닫습니다.
PATHOBJ_bEnum PATHOBJ_bEnum 함수는 지정된 경로에서 다음 PATHDATA 레코드를 검색하고 경로의 곡선을 열거합니다.
PATHOBJ_bEnumClipLines PATHOBJ_bEnumClipLines 함수는 지정된 경로에서 잘린 선 세그먼트를 열거합니다.
PATHOBJ_bMoveTo PATHOBJ_bMoveTo 함수는 지정된 경로에서 현재 위치를 설정합니다.
PATHOBJ_bPolyBezierTo PATHOBJ_bPolyBezierTo 함수는 경로에 베지어 곡선을 그립니다.
PATHOBJ_bPolyLineTo PATHOBJ_bPolyLineTo 함수는 지정된 지점을 통해 경로의 현재 위치에서 선을 그립니다.

PATHOBJ_vEnumStart PATHOBJ_vEnumStart 함수는 지정된 PATHOBJ 구조에 드라이버가 PATHOBJ_bEnum 호출하여 경로의 선 및/또는 곡선을 열거하도록 알립니다.
PATHOBJ_vEnumStartClipLines PATHOBJ_vEnumStartClipLines 함수를 사용하면 드라이버가 지정된 클립 영역에 대해 줄을 잘리도록 요청할 수 있습니다.
PATHOBJ_vGetBounds PATHOBJ_vGetBounds 함수는 지정된 경로에 대한 경계 사각형을 검색합니다.
PDD_CANCREATESURFACE CanCreateD3DBuffer 콜백 함수는 드라이버가 지정된 설명의 드라이버 수준 명령 또는 꼭짓점 버퍼를 만들 수 있는지 여부를 결정합니다.
PDD_COLORCB_COLORCONTROL DdControlColor 콜백 함수는 오버레이 표면의 광도 및 밝기 컨트롤을 제어합니다.
PDD_CREATEPALETTE DdCreatePalette 콜백 함수는 지정된 DirectDraw 개체에 대한 DirectDrawPalette 개체를 만듭니다.
PDD_CREATESURFACE CreateD3DBuffer 콜백 함수는 지정된 설명의 드라이버 수준 명령 또는 꼭짓점 버퍼를 만드는 데 사용됩니다.
PDD_CREATESURFACEEX D3dCreateSurfaceEx 함수는 Direct3D 렌더링을 위해 표면을 설정할 수 있도록 Microsoft DirectDraw 표면과 Microsoft Direct3D 핸들 값의 연결에 대해 알립니다.
PDD_DESTROYDDLOCAL D3dDestroyDDDLocal 함수는 동일한 지정된 로컬 Microsoft DirectDraw 개체에 속하는 D3dCreateSurfaceEx 함수에서 이전에 만든 모든 Microsoft Direct3D 표면을 제거합니다.
PDD_FLIPTOGDISURFACE DdFlipToGDISurface 콜백 함수는 DirectDraw가 GDI 표면에서 대칭 이동 중일 때 드라이버에 알립니다.

PDD_FREEDRIVERMEMORY DdFreeDriverMemory 콜백 함수는 오프스크린 또는 비로컬 표시 메모리를 해제하여 새 할당 요청을 충족합니다.
PDD_GETAVAILDRIVERMEMORY DdGetAvailDriverMemory 콜백 함수는 드라이버 관리 메모리 힙의 사용 가능한 메모리 양을 쿼리합니다.
PDD_GETDRIVERINFO DdGetDriverInfo 함수는 드라이버에서 지원하는 추가 DirectDraw 및 Direct3D 기능을 드라이버에 쿼리합니다.
PDD_GETDRIVERSTATE D3dGetDriverState 함수는 Microsoft DirectDraw 및 Microsoft Direct3D 런타임 모두 드라이버에서 현재 상태에 대한 정보를 가져오는 데 사용됩니다.
PDD_GETSCANLINE DdGetScanLine 콜백 함수는 현재 실제 검사 줄의 수를 반환합니다.
PDD_KERNELCB_SYNCSURFACE DdSyncSurfaceData 콜백 함수는 비디오 미니포트 드라이버에 전달되기 전에 표면 데이터를 설정하고 수정합니다.
PDD_KERNELCB_SYNCVIDEOPORT DdSyncVideoPortData 콜백 함수는 비디오 미니포트 드라이버에 전달되기 전에 VPE 개체 데이터를 설정하고 수정합니다.
PDD_MAPMEMORY DdMapMemory 콜백 함수는 프레임 버퍼의 애플리케이션 수정 가능한 부분을 지정된 프로세스의 사용자 모드 주소 공간에 매핑하거나 메모리를 매핑 해제합니다.
PDD_MOCOMPCB_BEGINFRAME DdMoCompBeginFrame 콜백 함수는 새 프레임 디코딩을 시작합니다.
PDD_MOCOMPCB_CREATE DdMoCompCreate 콜백 함수는 지정된 GUID를 사용하여 소프트웨어 디코더가 동작 보정을 사용하기 시작할 것임을 드라이버에 알합니다.

PDD_MOCOMPCB_DESTROY DdMoCompDestroy 콜백 함수는 드라이버에 이 동작 보정 개체가 더 이상 사용되지 않음을 알 수 있습니다. 이제 드라이버는 필요한 정리를 수행해야 합니다.
PDD_MOCOMPCB_ENDFRAME DdMoCompEndFrame 콜백 함수는 디코딩된 프레임을 완료합니다.
PDD_MOCOMPCB_GETCOMPBUFFINFO DDMoCompGetBuffInfo 콜백 함수를 사용하면 드라이버는 지정된 GUID를 지원하는 데 필요한 중간 표면 수와 이러한 각 표면의 크기, 위치 및 형식을 지정할 수 있습니다.
PDD_MOCOMPCB_GETFORMATS DdMoCompGetFormats 콜백 함수는 하드웨어에서 데이터를 디코딩할 수 있는 압축되지 않은 형식을 나타냅니다.
PDD_MOCOMPCB_GETGUIDS DdMoCompGetGuids 콜백 함수는 드라이버에서 지원하는 GUID 수를 검색합니다.
PDD_MOCOMPCB_GETINTERNALINFO DdMoCompGetInternalInfo 콜백 함수를 사용하면 드라이버가 디스플레이 메모리를 내부적으로 할당하여 동작 보정을 수행한다고 보고할 수 있습니다.
PDD_MOCOMPCB_QUERYSTATUS DdMoCompQueryStatus 콜백 함수는 가장 최근의 렌더링 작업의 상태를 지정된 표면에 쿼리합니다.
PDD_MOCOMPCB_RENDER DdMoCompRender 콜백 함수는 매크로 블록이 포함된 표면, 매크로 블록이 있는 각 표면의 오프셋 및 렌더링할 macroblock 데이터의 크기를 지정하여 렌더링할 매크로 블록의 크기를 드라이버에 알려줍니다.
PDD_PALCB_DESTROYPALETTE DdDestroyPalette 콜백 함수는 지정된 팔레트를 삭제합니다.
PDD_PALCB_SETENTRIES DdSetEntries 콜백 함수는 지정된 색상표의 색상표 항목을 업데이트합니다.

PDD_SETEXCLUSIVEMODE DdSetExclusiveMode 콜백 함수는 DirectDraw 애플리케이션이 배타적 모드로 전환될 때 드라이버에 알립니다.
PDD_SURFCB_ADDATTACHEDSURFACE DdAddAttachedSurface 콜백 함수는 표면을 다른 표면에 연결합니다.
PDD_SURFCB_BLT DdBlt 콜백 함수는 비트 블록 전송을 수행합니다.
PDD_SURFCB_DESTROYSURFACE DdDestroySurface 콜백 함수는 DirectDraw 표면을 제거합니다.
PDD_SURFCB_FLIP DdFlip 콜백 함수를 사용하면 대상 표면과 연결된 표면 메모리가 기본 표면이 되고 현재 표면이 기본 표면이 되지 않습니다.
PDD_SURFCB_GETBLTSTATUS DdGetBltStatus 콜백 함수는 지정된 표면의 블릿 상태를 쿼리합니다.
PDD_SURFCB_GETFLIPSTATUS DdGetFlipStatus 콜백 함수는 표면에서 가장 최근에 요청된 대칭 이동이 발생했는지 여부를 결정합니다.
PDD_SURFCB_LOCK DdLock 콜백 함수는 지정된 표면 메모리 영역을 잠그고 표면과 연결된 메모리 블록에 대한 유효한 포인터를 제공합니다.
PDD_SURFCB_SETCOLORKEY DdSetColorKey 콜백 함수는 지정된 표면의 색 키 값을 설정합니다.
PDD_SURFCB_SETOVERLAYPOSITION DdSetOverlayPosition 콜백 함수는 오버레이의 위치를 설정합니다.
PDD_SURFCB_SETPALETTE DdSetPalette 콜백 함수는 지정된 표면에 색상표를 연결합니다.

PDD_SURFCB_UNLOCK DdUnLock 콜백 함수는 지정된 표면에 보관된 잠금을 해제합니다.
PDD_SURFCB_UPDATEOVERLAY DdUpdateOverlay 콜백 함수는 오버레이 표면의 시각적 특성을 변경하거나 수정합니다.
PDD_VPORTCB_CANCREATEVIDEOPORT DdVideoPortCanCreate 콜백 함수는 드라이버가 지정된 설명의 DirectDraw VPE 개체를 지원할 수 있는지 여부를 결정합니다.
PDD_VPORTCB_COLORCONTROL DdVideoPortColorControl 콜백 함수는 VPE 개체 색 컨트롤을 가져오거나 설정합니다.
PDD_VPORTCB_CREATEVIDEOPORT DdVideoPortCreate 콜백 함수는 DirectDraw가 VPE 개체를 만들었다는 사실을 드라이버에 알합니다.
PDD_VPORTCB_DESTROYVPORT DdVideoPortDestroy 콜백 함수는 DirectDraw가 지정된 VPE 개체를 제거했음을 드라이버에 알합니다.
PDD_VPORTCB_FLIP DdVideoPortFlip 콜백 함수는 물리적 대칭 이동 기능을 수행하여 VPE 개체가 새 표면에 데이터를 쓰기 시작합니다.
PDD_VPORTCB_GETBANDWIDTH DdVideoPortGetBandwidth 콜백 함수는 지정된 VPE 개체 출력 형식을 기반으로 디바이스의 프레임 버퍼 메모리에 대한 대역폭 제한을 보고합니다.
PDD_VPORTCB_GETFIELD DdVideoPortGetField 콜백 함수는 인터레이스된 신호의 현재 필드가 짝수인지 아니면 홀수인지를 결정합니다.
PDD_VPORTCB_GETFLIPSTATUS DdVideoPortGetFlipStatus 콜백 함수는 표면에서 가장 최근에 요청한 대칭 이동이 발생했는지 여부를 결정합니다.

PDD_VPORTCB_GETINPUTFORMATS DdVideoPortGetInputFormats 콜백 함수는 DirectDraw VPE 개체가 허용할 수 있는 입력 형식을 결정합니다.
PDD_VPORTCB_GETLINE DdVideoPortGetLine 콜백 함수는 하드웨어 비디오 포트의 현재 줄 번호를 반환합니다.
PDD_VPORTCB_GETOUTPUTFORMATS DdVideoPortGetOutputFormats 콜백 함수는 VPE 개체가 지원하는 출력 형식을 결정합니다.
PDD_VPORTCB_GETSIGNALSTATUS DdVideoPortGetSignalStatus 콜백 함수는 하드웨어 비디오 포트에 현재 표시되는 비디오 신호의 상태를 검색합니다.
PDD_VPORTCB_GETVPORTCONNECT DdVideoPortGetConnectInfo 콜백 함수는 지정된 VPE 개체에서 지원하는 연결을 반환합니다.
PDD_VPORTCB_UPDATE DdVideoPortUpdate 콜백 함수는 VPE 개체를 시작 및 중지하고 VPE 개체 데이터 스트림을 수정합니다.
PDD_VPORTCB_WAITFORSYNC DdVideoPortWaitForSync 콜백 함수는 다음 세로 동기화가 발생할 때까지 대기합니다.
PDD_WAITFORVERTICALBLANK DdWaitForVerticalBlank 콜백 함수는 디바이스의 세로 빈 상태를 반환합니다.
PDX_BOBNEXTFIELD DxBobNextField 콜백 함수는 인터리브된 데이터의 다음 필드를 표시합니다.
PDX_ENABLEIRQ DxEnableIRQ 콜백 함수는 비디오 미니포트 드라이버에 IRQ를 사용하거나 사용하지 않도록 설정해야 하는 것을 나타냅니다.
PDX_FLIPOVERLAY DxFlipOverlay 콜백 함수는 비디오 미니포트 드라이버의 클라이언트가 오버레이를 대칭 이동하려고 하거나 자동 이동이 활성화될 때 호출됩니다.

PDX_FLIPVIDEOPORT DxFlipVideoPort 콜백 함수는 비디오 미니포트 드라이버의 클라이언트가 VPE(비디오 포트 확장) 개체를 대칭 이동하려고 하거나 자동 이동이 설정된 경우에 호출됩니다.
PDX_GETCURRENTAUTOFLIP DxGetCurrentAutoflip 콜백 함수는 디바이스가 하드웨어 자동 매핑이고 비디오 미니포트 드라이버의 클라이언트가 캡처를 위해 비디오 데이터의 현재 필드를 수신하는 화면을 알고자 할 때 호출됩니다.
PDX_GETIRQINFO DxGetIRQInfo 콜백 함수는 드라이버가 인터럽트 요청을 관리한다는 것을 나타냅니다.
PDX_GETPOLARITY DxGetPolarity 콜백 함수는 VPE(비디오 포트 확장) 개체에 의해 작성되는 현재 필드의 극성(짝수 또는 홀수)을 반환합니다.
PDX_GETPREVIOUSAUTOFLIP DxGetPreviousAutoflip 콜백 함수는 디바이스가 하드웨어 자동 매핑이고 비디오 미니포트 드라이버의 클라이언트가 캡처를 위해 비디오 데이터의 이전 필드를 수신한 화면을 알고자 할 때 호출됩니다.
PDX_GETTRANSFERSTATUS DxGetTransferStatus 콜백 함수는 DirectDraw에서 어떤 하드웨어 버스 마스터가 완료되었는지 확인하는 데 사용됩니다.
PDX_IRQCALLBACK IRQCallback 함수는 발생한 IRQ와 관련된 작업을 수행합니다.
PDX_LOCK DxLock 콜백 함수는 비디오 미니포트 드라이버의 클라이언트가 프레임 버퍼에 액세스하려고 할 때 호출됩니다.
PDX_SETSTATE DxSetState 콜백 함수는 비디오 미니포트 드라이버의 클라이언트가 bob 모드에서 직조 모드로 전환하려는 경우 호출되며 그 반대의 경우도 마찬가지입니다.
PDX_SKIPNEXTFIELD DxSkipNextField 콜백 함수는 다음 필드를 건너뛰거나 다시 사용하도록 설정해야 할 때 호출됩니다.

PDX_TRANSFER DxTransfer 콜백 함수는 드라이버에 표면의 마스터 데이터를 MDL(메모리 설명자 목록)에 지정된 버퍼로 전달합니다.
PFN_DrvQueryGlyphAttrs DrvQueryGlyphAttrs 함수는 글꼴의 문자 모양에 대한 정보를 반환합니다.
QueryDisplayConfig QueryDisplayConfig 함수는 현재 설정에서 모든 디스플레이 디바이스 또는 보기에 대해 가능한 모든 표시 경로에 대한 정보를 검색합니다.
ReleaseKernelHandle IDirectDrawKernel::ReleaseKernelHandle 메서드는 DirectDraw 개체에 커널 모드 핸들을 해제합니다.
ReleaseKernelHandle IDirectDrawSurfaceKernel::ReleaseKernelHandle 메서드는 커널 모드 핸들을 DirectDraw 표면으로 해제합니다.
SetDisplayConfig SetDisplayConfig 함수는 현재 세션에서 지정된 경로를 단독으로 사용하도록 설정하여 표시 토폴로지, 원본 및 대상 모드를 수정합니다.
STROBJ_bEnum STROBJ_bEnum 함수는 문자 모양 ID 및 위치를 열거합니다.
STROBJ_bEnumPositionsOnly STROBJ_bEnumPositionsOnly 함수는 지정된 텍스트 문자열의 문자 모양 ID와 위치를 열거하지만 캐시된 문자 모양 비트맵은 만들지 않습니다.
STROBJ_bGetAdvanceWidths STROBJ_bGetAdvanceWidths 함수는 지정된 문자열을 구성하는 문자 모양의 가능한 너비를 지정하는 벡터 배열을 검색합니다.
STROBJ_dwGetCodePage STROBJ_dwGetCodePage 함수는 지정된 STROBJ 구조체와 연결된 코드 페이지를 반환합니다.

STROBJ_fxBreakExtra STROBJ_fxBreakExtra 함수는 양쪽 맞춤 텍스트를 표시 및/또는 인쇄할 때 문자열의 각 공백 문자에 추가할 추가 공간의 양을 검색합니다.
STROBJ_fxCharacterExtra STROBJ_fxCharacterExtra 함수는 표시 및/또는 인쇄할 때 문자열에서 각 문자의 너비를 보강할 추가 공간의 양을 검색합니다.
STROBJ_vEnumStart STROBJ_vEnumStart 함수는 STROBJ_bEnum 대한 후속 호출에서 GDI에서 반환되는 데이터의 폼 또는 형식을 정의합니다.
VidMemFree VidMemFree 함수는 HeapVidMemAllocAligned에 의해 디스플레이 드라이버에 할당된 오프스크림 메모리를 해제합니다.
WNDOBJ_bEnum WNDOBJ_bEnum 함수는 창의 표시되는 영역에서 사각형 일괄 처리를 가져옵니다.
WNDOBJ_cEnumStart WNDOBJ_cEnumStart 함수는 창의 표시되는 영역에서 사각형을 열거하기 위한 매개 변수를 설정하는 콜백 함수입니다.
WNDOBJ_vSetConsumer WNDOBJ_vSetConsumer 함수는 지정된 WNDOBJ 구조체의 pvConsumer 필드에서 드라이버 정의 값을 설정합니다.
WNDOBJCHANGEPROC **WNDOBJCHANGEPROC** 함수는 GDI가 드라이버에 해당 창의 변경 내용을 알리는 데 사용하는 드라이버 정의 콜백 함수입니다.
XFORMOBJ_bApplyXform XFORMOBJ_bApplyXform 함수는 지정된 변환 또는 해당 역을 지정된 점 배열에 적용합니다.
XFORMOBJ_iGetFloatObjXform XFORMOBJ_iGetFloatObjXform 함수는 FLOATOBJ 변환을 드라이버에 다운로드합니다.

XFORMOBJ_iGetXform XFORMOBJ_iGetXform 함수는 드라이버에 변환을 다운로드합니다.
XLATEOBJ_cGetPalette XLATEOBJ_cGetPalette 함수는 지정된 색상표에서 RGB 색 또는 비트 필드 형식을 검색합니다.
XLATEOBJ_hGetColorTransform XLATEOBJ_hGetColorTransform 함수는 지정된 변환 개체의 색 변환을 반환합니다.
XLATEOBJ_iXlate XLATEOBJ_iXlate 함수는 원본 팔레트의 색 인덱스를 대상 팔레트에서 가장 가까운 인덱스로 변환합니다.
XLATEOBJ_piVector XLATEOBJ_piVector 함수는 드라이버가 원본 인덱스를 대상 인덱스로 변환하는 데 사용할 수 있는 변환 벡터를 검색합니다.

인터페이스

IDirectDrawKernel IDirectDrawKernel 인터페이스
IDirectDrawSurfaceKernel IDirectDrawSurfaceKernel 인터페이스

구조체

DD_ADDATTACHEDSURFACEDATA DD_ADDATTACHEDSURFACEDATA 구조체에는 표면을 다른 표면에 연결하는 데 필요한 정보가 포함되어 있습니다.

DD_ATTACHLIST DD_ATTACHLIST 구조체는 Microsoft DirectDraw에 대한 연결된 표면 목록을 유지 관리합니다.
DD_BEGINMOCOMPFRAMEDATA DDHAL_BEGINMOCOMPFRAMEDATA 구조체에는 디코딩을 시작하는 데 필요한 프레임 정보가 포함됩니다.
DD_BLTDATA DD_BLTDATA 구조에는 비트 블록 전송을 수행하는 드라이버와 관련된 정보가 포함됩니다.
DD_CANCREATESURFACEDATA DD_CANCREATESURFACEDATA 구조체에는 표면(CanCreateD3DBuffer의 경우 버퍼)을 만들 수 있는지 여부를 나타내는 데 필요한 정보가 포함되어 있습니다.
DD_CANCREATEVPORTDATA DD_CANCREATEVPORTDATA 구조체에는 VPE(비디오 포트 확장) 개체를 만들 수 있는지 여부를 결정하는 데 드라이버에 필요한 정보가 포함되어 있습니다.
DD_CLIPPER_GLOBAL DD_CLIPPER_GLOBAL 구조체에는 개체 인스턴스 간에 공유할 수 있는 전역 DirectDrawClipper 데이터가 포함됩니다.
DD_CLIPPER_LOCAL DD_CLIPPER_LOCAL 구조체에는 각 개별 DirectDrawClipper 개체에 대한 로컬 데이터가 포함됩니다.
DD_COLORCONTROLDATA DD_COLORCONTROLDATA 구조체에는 지정된 오버레이에 대한 색 제어 정보가 포함됩니다.
DD_CREATEMOCOMPDATA DD_CREATEMOCOMPDATA 구조에는 동작 보정 사용을 시작하는 데 필요한 데이터가 포함됩니다.
DD_CREATEPALETTEDATA DD_CREATEPALETTEDATA 구조체에는 이 Microsoft DirectDraw 개체에 대한 DirectDrawPalette 개체를 만드는 데 필요한 정보가 포함되어 있습니다.

DD_CREATESURFACEDATA DD_CREATESURFACEDATA 구조체는 CreateD3DBuffer, 명령 또는 꼭짓점 버퍼의 경우 표면을 만드는 데 필요한 정보를 포함합니다.
DD_CREATESURFACEEXDATA DD_CREATESURFACEEXDATA 구조체에는 드라이버가 표면을 만들고 제공된 텍스처 핸들과 연결하는 데 필요한 정보가 포함되어 있습니다.
DD_CREATEVPORTDATA DD_CREATEVPORTDATA 구조에는 생성되는 VPE(비디오 포트 확장) 개체를 설명하는 데 필요한 정보가 포함되어 있습니다.
DD_DESTROYMOCOMPDATA DD_DESTROYMOCOMPDATA 구조에는 동작 보정 수행을 완료하는 데 필요한 정보가 포함되어 있습니다.
DD_DESTROYPALETTEDATA DD_DESTROYPALETTEDATA 구조체에는 지정된 팔레트를 삭제하는 데 필요한 정보가 포함되어 있습니다.
DD_DESTROYSURFACEDATA DD_DESTROYSURFACEDATA 구조체에는 지정된 표면(DestroyD3DBuffer의 경우 명령 또는 꼭짓점 버퍼)을 삭제하는 데 필요한 정보가 포함됩니다.
DD_DESTROYVPORTDATA DD_DESTROYVPORTDATA 구조에는 드라이버가 정리하는 데 필요한 정보가 포함되어 있습니다.
DD_DIRECTDRAW_GLOBAL DD_DIRECTDRAW_GLOBAL 구조에는 드라이버의 디바이스를 설명하는 드라이버 정보가 포함되어 있습니다.
DD_DIRECTDRAW_LOCAL DD_DIRECTDRAW_LOCAL 구조에는 현재 DirectDraw 프로세스와 관련된 드라이버 정보만 포함됩니다.
DD_ENDMOCOMPFRAMEDATA DD_ENDMOCOMPFRAMEDATA 구조체에는 디코딩된 프레임을 완료하는 데 필요한 정보가 포함되어 있습니다.

DD_FLIPDATA DD_FLIPDATA 구조체에는 대칭 이동에 필요한 정보가 포함되어 있습니다.
DD_FLIPTOGDISURFACEDATA DD_FLIPTOGDISURFACEDATA 구조에는 GDI 표면 알림 정보가 포함됩니다.
DD_FLIPVPORTDATA DD_FLIPVPORTDATA 구조체에는 VPE(비디오 포트 확장) 개체가 대칭 이동 작업을 수행하는 데 필요한 정보가 포함되어 있습니다.
DD_FREEDRIVERMEMORYDATA DD_FREEDRIVERMEMORYDATA 구조에는 무료 요청의 세부 정보가 포함됩니다.
DD_GETAVAILDRIVERMEMORYDATA DD_GETAVAILDRIVERMEMORYDATA 구조에는 드라이버가 사용 가능한 메모리 양을 쿼리하고 반환하는 데 필요한 정보가 포함되어 있습니다.
DD_GETBLTSTATUSDATA DD_GETBLTSTATUSDATA 구조체는 음역 상태 정보를 반환합니다.
DD_GETDRIVERINFODATA DD_GETDRIVERINFODATA 구조는 DdGetDriverInfo 콜백 루틴에서 데이터를 전달하는 데 사용됩니다.
DD_GETDRIVERSTATEDATA DD_GETDRIVERSTATEDATA 구조체는 드라이버의 상태를 설명합니다.
DD_GETFLIPSTATUSDATA DD_GETFLIPSTATUSDATA 구조체는 대칭 이동 상태 정보를 반환합니다.
DD_GETINTERNALMOCOMPDATA DD_GETINTERNALMOCOMPDATA 구조에는 내부 메모리 요구 사항이 포함됩니다.
DD_GETMOCOMPCOMPBUFFDATA DD_GETMOCOMPCOMPBUFFDATA 구조에는 압축된 버퍼 정보가 포함됩니다.

DD_GETMOCOMPFORMATSDATA DD_GETMOCOMPFORMATSDATA 구조체에는 압축되지 않은 형식 정보가 포함됩니다.
DD_GETMOCOMPGUIDSDATA DD_GETMOCOMPGUIDSDATA 구조에는 동작 보정 GUID 정보가 포함됩니다.
DD_GETSCANLINEDATA DD_GETSCANLINEDATA 구조에는 현재 검사 줄의 수를 쿼리하고 반환하는 데 필요한 멤버가 포함됩니다.
DD_GETVPORTBANDWIDTHDATA DD_GETVPORTBANDWIDTHDATA 구조체에는 지정된 형식에 대한 대역폭 정보가 포함됩니다.
DD_GETVPORTCONNECTDATA DD_GETVPORTCONNECTDATA 구조에는 지정된 VPE(비디오 포트 확장) 개체에서 지원하는 연결 조합이 포함됩니다.
DD_GETVPORTFIELDDATA DD_GETVPORTFIELDDATA 구조체에는 드라이버가 인터레이스된 신호의 현재 필드가 짝수인지 아니면 홀수인지를 확인하는 데 필요한 정보가 포함되어 있습니다.
DD_GETVPORTFLIPSTATUSDATA DD_GETVPORTFLIPSTATUSDATA 구조체에는 지정된 표면에 대한 대칭 이동 상태 정보가 포함됩니다.
DD_GETVPORTINPUTFORMATDATA DD_GETVPORTINPUTFORMATDATA 구조체에는 드라이버가 VPE(비디오 포트 확장) 개체가 허용할 수 있는 입력 형식을 반환하는 데 필요한 정보가 포함되어 있습니다.
DD_GETVPORTLINEDATA DD_GETVPORTLINEDATA 구조체에는 하드웨어 비디오 포트의 현재 줄 번호가 포함됩니다.
DD_GETVPORTOUTPUTFORMATDATA DD_GETVPORTOUTPUTFORMATDATA 구조체에는 드라이버가 VPE(비디오 포트 확장) 개체가 지정된 입력 형식에 대해 지원하는 모든 출력 형식을 반환하는 데 필요한 정보가 포함되어 있습니다.
DD_GETVPORTSIGNALDATA DD_GETVPORTSIGNALDATA 구조에는 하드웨어 비디오 포트의 신호 상태가 포함됩니다.

DD_LOCKDATA DD_LOCKDATA 구조체에는 Microsoft DirectDraw 매개 변수 구조에 정의된 잠금을 수행하는 데 필요한 정보가 포함되어 있습니다.
DD_MAPMEMORYDATA DD_MAPMEMORYDATA 구조에는 프레임 버퍼를 사용자 모드 메모리에 매핑하거나 매핑 해제하는 데 필요한 정보가 포함되어 있습니다.
DD_MORESURFACECAPS DD_MORESURFACECAPS 구조는 DDCORECAPS에 설명된 기능 외에도 더 많은 드라이버 표면 기능을 정의합니다.
DD_MOTIONCOMP_LOCAL DD_MOTIONCOMP_LOCAL 구조에는 각 개별 Microsoft DirectDraw 동작 보정 개체에 대한 로컬 데이터가 포함됩니다.
DD_PALETTE_GLOBAL DD_PALETTE_GLOBAL 구조에는 개체 인스턴스 간에 공유할 수 있는 전역 DirectDrawPalette 데이터가 포함됩니다.
DD_PALETTE_LOCAL DD_PALETTE_LOCAL 구조체에는 개별 색상표 개체에 고유한 색상표 관련 데이터가 포함되어 있습니다.
DD_QUERYMOCOMPSTATUSDATA DD_QUERYMOCOMPSTATUSDATA 구조에는 이전 프레임의 상태를 쿼리하는 데 필요한 정보가 포함되어 있습니다.
DD_RENDERMOCOMPDATA DD_RENDERMOCOMPDATA 구조체에는 프레임을 렌더링하는 데 필요한 정보가 포함됩니다.
DD_SETCOLORKEYDATA DD_SETCOLORKEYDATA 구조체에는 지정된 표면의 색 키 값을 설정하는 데 필요한 정보가 포함되어 있습니다.
DD_SETENTRIESDATA DD_SETENTRIESDATA 구조체에는 팔레트 항목을 설정하는 데 필요한 정보가 포함되어 있습니다.

DD_SETEXCLUSIVEMODEDATA DD_SETEXCLUSIVEMODEDATA 구조에는 배타적 모드 알림 정보가 포함됩니다.
DD_SETOVERLAYPOSITIONDATA DD_SETOVERLAYPOSITIONDATA 구조체에는 오버레이 표면의 표시 좌표를 변경하는 데 필요한 정보가 포함되어 있습니다.
DD_SETPALETTEDATA DD_SETPALETTEDATA 구조체에는 특정 표면의 색상표를 설정하는 데 필요한 정보가 포함되어 있습니다.
DD_STEREOMODE DD_STEREOMODE 구조는 런타임에서 DdGetDriverInfo 호출에서 GUID_DDStereoMode 사용하여 드라이버가 지정된 비디오 디스플레이 모드에 대한 스테레오를 지원하는지 여부를 쿼리합니다.
DD_SURFACE_GLOBAL DD_SURFACE_GLOBAL 구조에는 여러 표면 간에 공유할 수 있는 전역 표면 관련 데이터가 포함되어 있습니다.
DD_SURFACE_INT DD_SURFACE_INT 구조체에는 DirectDrawSurface 개체의 인터페이스 정보가 포함됩니다.
DD_SURFACE_LOCAL DD_SURFACE_LOCAL 구조에는 개별 표면 개체에 고유한 표면 관련 데이터가 포함됩니다.
DD_SURFACE_MORE DD_SURFACE_MORE 구조체에는 각 개별 DirectDrawSurface 개체에 대한 추가 로컬 데이터가 포함됩니다.
DD_SYNCSURFACEDATA DD_SYNCSURFACEDATA 구조체에는 표면 정보가 포함됩니다.
DD_SYNCVIDEOPORTDATA DD_SYNCVIDEOPORTDATA 구조에는 VPE(비디오 포트 확장) 개체 정보가 포함됩니다.
DD_UNLOCKDATA DD_UNLOCKDATA 구조에는 Microsoft DirectDraw 매개 변수 구조에 정의된 대로 잠금 해제를 수행하는 데 필요한 정보가 포함되어 있습니다.

DD_UPDATENONLOCALHEAPDATA DD_UPDATENONLOCALHEAPDATA 구조에는 필요한 힙 정보가 포함됩니다.
DD_UPDATEOVERLAYDATA DD_UPDATEOVERLAYDATA 구조체에는 오버레이 표면을 업데이트하는 데 필요한 정보가 포함되어 있습니다.
DD_UPDATEVPORTDATA DD_UPDATEVPORTDATA 구조에는 VPE(비디오 포트 확장) 개체를 시작, 중지 및 변경하는 데 필요한 정보가 포함되어 있습니다.
DD_VIDEOPORT_LOCAL DD_VIDEOPORT_LOCAL 구조에는 개별 Microsoft DirectDraw VPE 개체에 고유한 VPE(비디오 포트 확장) 관련 데이터가 포함되어 있습니다.
DD_VPORTCOLORDATA DD_VPORTCOLORDATA 구조에는 VPE(비디오 포트 확장) 개체 색 제어 정보가 포함됩니다.
DD_WAITFORVERTICALBLANKDATA DD_WAITFORVERTICALBLANKDATA 구조에는 모니터의 세로 빈 정보를 가져오는 데 필요한 정보가 포함되어 있습니다. D
DD_WAITFORVPORTSYNCDATA DD_WAITFORVPORTSYNCDATA 구조에는 드라이버가 VPE(비디오 포트 확장) 개체를 동기화하는 데 필요한 정보가 포함되어 있습니다.
DDHAL_DESTROYDDLOCALDATA DDHAL_DESTROYDDLOCALDATA 드라이버가 지정된 로컬 DirectDraw 개체에 연결된 표면 집합을 삭제하는 데 필요한 정보를 포함합니다.
DDHAL_WAITFORVERTICALBLANKDATA DD_WAITFORVERTICALBLANKDATA 구조에는 모니터의 세로 빈 정보를 가져오는 데 필요한 정보가 포함되어 있습니다. (DDHAL_WAITFORVERTICALBLANKDATA)
DDKERNELCAPS DDKERNELCAPS 구조는 커널 모드 비디오 전송을 위한 미니포트 드라이버에 존재하는 지원(있는 경우)을 클라이언트에 알릴 수 있습니다.

DDVIDEOPORTBANDWIDTH DDVIDEOPORTBANDWIDTH 구조체는 특정 VPE(비디오 포트 확장) 개체/픽셀 형식 구성과 함께 사용할 때 오버레이의 대역폭 특성을 설명합니다.
DDVIDEOPORTCAPS DDVIDEOPORTCAPS 구조체는 하드웨어 비디오 포트의 기능 및 맞춤 제한을 설명합니다.
DDVIDEOPORTDESC DDVIDEOPORTDESC 구조는 생성되는 VPE(비디오 포트 확장) 개체에 대해 설명합니다.
DDVIDEOPORTINFO DDVIDEOPORTINFO 구조는 드라이버가 비디오 데이터를 표면(또는 표면)으로 전송하는 방법을 설명합니다. DDVIDEOPORTINFO는 DD_VIDEOPORT_LOCAL 구조체의 멤버입니다.
어댑터 어댑터 구조는 그래픽 어댑터를 설명합니다.
어댑터 어댑터 구조에는 그래픽 어댑터 목록이 포함되어 있습니다.
BLENDOBJ BLENDOBJ 구조체는 소스 및 대상 비트맵에 대한 혼합 함수를 지정하여 혼합을 제어합니다.
BRUSHOBJ BRUSHOBJ 구조체에는 브러시 개체를 설명하는 세 개의 공용 멤버가 포함되어 있습니다.
CIECHROMA CIECHROMA 구조는 CIE 색 공간에서 색도 좌표, x 및 y 및 광도 Y를 설명하는 데 사용됩니다.
클립라인 CLIPLINE 구조는 드라이버가 그리기에 사용되는 두 클립 영역 사이의 선 부분에 액세스할 수 있도록 합니다.
CLIPOBJ CLIPOBJ 구조체는 그리기 시 사용되는 클립 영역을 설명합니다.

COLORINFO COLORINFO 구조는 CIE 좌표 공간에서 디바이스의 색을 정의합니다.
DD_CALLBACKS DD_CALLBACKS 구조에는 디바이스 드라이버가 지원하는 콜백 함수에 대한 진입 포인터가 포함됩니다.
DD_COLORCONTROLCALLBACKS DD_COLORCONTROLCALLBACKS 구조에는 디바이스 드라이버에서 지원하는 Microsoft DirectDraw 색 제어 콜백에 대한 진입 포인터가 포함되어 있습니다.
DD_D3DBUFCALLBACKS DD_D3DBUFCALLBACKS 구조는 명령 및 꼭짓점 버퍼의 드라이버 수준 할당을 구현하는 드라이버에서만 사용됩니다.
DD_GETHEAPALIGNMENTDATA DD_GETHEAPALIGNMENTDATA 구조체에는 특정 힙의 필수 맞춤에 대한 데이터가 포함됩니다.
DD_HALINFO DD_HALINFO 구조는 하드웨어 및 드라이버의 기능을 설명합니다.
DD_KERNELCALLBACKS DD_KERNELCALLBACKS 구조에는 드라이버가 지원하는 DirectDraw 커널 모드 콜백 함수에 대한 진입 포인터가 포함되어 있습니다.
DD_MISCELLANEOUS2CALLBACKS DD_MISCELLANEOUS2CALLBACKS 구조체는 기타 콜백 루틴의 주소를 반환하는 데 사용됩니다.
DD_MISCELLANEOUSCALLBACKS DD_MISCELLANEOUSCALLBACKS 구조에는 디바이스 드라이버가 지원하는 메모리 쿼리 콜백에 대한 항목 포인터가 포함되어 있습니다.
DD_MOTIONCOMPCALLBACKS DD_MOTIONCOMPCALLBACKS 구조에는 디바이스 드라이버가 지원하는 동작 보정 콜백 함수에 대한 진입 포인터가 포함되어 있습니다.
DD_NONLOCALVIDMEMCAPS DD_NONLOCALVIDMEMCAPS 구조에는 비로컬 표시 메모리에 대한 기능이 포함되어 있습니다.

DD_NTCALLBACKS DD_NTCALLBACKS 구조에는 디바이스 드라이버에서 지원하는 Microsoft Windows 2000 이상 Microsoft DirectDraw 콜백 함수에 대한 진입 포인터가 포함되어 있습니다.
DD_NTPRIVATEDRIVERCAPS DD_NTPRIVATEDRIVERCAPS 구조를 사용하면 DirectDraw에서 표면을 만들 때 드라이버가 Microsoft DirectDraw의 동작을 변경할 수 있습니다.
DD_PALETTECALLBACKS DD_PALETTECALLBACKS 구조에는 디바이스 드라이버가 지원하는 DirectDraw 팔레트 콜백 함수에 대한 진입 포인터가 포함되어 있습니다.
DD_SURFACECALLBACKS DD_SURFACECALLBACKS 구조에는 디바이스 드라이버가 지원하는 Microsoft DirectDraw 표면 콜백 함수에 대한 진입 포인터가 포함되어 있습니다.
DD_VIDEOPORTCALLBACKS DD_VIDEOPORTCALLBACKS 구조에는 디바이스 드라이버가 지원하는 Microsoft DirectDraw VPE(비디오 포트 확장) 콜백 함수에 대한 진입 포인터가 포함되어 있습니다.
DDADDVPCAPTUREBUFF DDADDVPCAPTUREBUFF 구조에는 내부 캡처 큐에 새 버퍼를 추가하는 데 필요한 정보가 포함되어 있습니다.
DDBOBNEXTFIELDINFO DDBOBNEXTFIELDINFO 구조체에는 표면에 대한 bob 정보가 포함됩니다.
DDCAPBUFFINFO DDCAPBUFFINFO 구조체에는 캡처 정보가 포함됩니다.
DDCLOSEHANDLE DDCLOSEHANDLE 구조체에는 Microsoft DirectDraw 개체, 표면, VPE(비디오 포트 확장) 개체 또는 VPE 캡처 핸들이 포함됩니다.
DDCOMPBUFFERINFO DDCOMPBUFFERINFO 구조체에는 압축 버퍼에 대한 드라이버 제공 정보가 포함되어 있습니다.

DDCORECAPS DDCORECAPS 구조는 DirectDraw 개체를 통해 애플리케이션에 노출되는 Microsoft DirectDraw 드라이버 및 해당 디바이스의 핵심 기능을 지정합니다.
DDENABLEIRQINFO DDENABLEIRQINFO 구조체에는 인터럽트 사용 설정에 필요한 정보가 포함되어 있습니다.
DDFLIPOVERLAY DDFLIPOVERLAY 구조체에는 대칭 이동에 필요한 표면 정보가 포함됩니다.
DDFLIPOVERLAYINFO DDFLIPOVERLAYINFO 구조체는 표면에 대한 대칭 이동 정보를 포함합니다.
DDFLIPVIDEOPORT DDFLIPVIDEOPORT 구조체에는 하드웨어 비디오 포트를 대칭 이동하는 데 필요한 정보가 포함되어 있습니다.
DDFLIPVIDEOPORTINFO DDFLIPVIDEOPORTINFO 구조체에는 VPE(비디오 포트 확장) 개체 및 표면 정보가 포함됩니다.
DDGETAUTOFLIPIN DDGETAUTOFLIPIN 구조체에는 핸들 정보가 포함됩니다.
DDGETAUTOFLIPOUT DDGETAUTOFLIPOUT 구조체에는 DxApi 함수의 DD_DXAPI_GET_CURRENT_VP_AUTOFLIP_SURFACE 및 DD_DXAPI_GET_LAST_VP_AUTOFLIP_SURFACE 함수 식별자에서 반환된 핸들 및 극성 정보가 포함됩니다.
DDGETCURRENTAUTOFLIPINFO DDGETCURRENTAUTOFLIPININFO 구조체에는 VPE(비디오 포트 확장) 개체 정보가 포함됩니다.
DDGETCURRENTAUTOFLIPOUTINFO DDGETCURRENTAUTOFLIPOUTINFO 구조체는 표면 정보를 제공합니다.
DDGETFIELDNUMIN DDGETFIELDNUMIN 구조에는 Microsoft DirectDraw 및 VPE(비디오 포트 확장) 개체 핸들 정보가 포함됩니다.

DDGETFIELDNUMOUT DDGETFIELDNUMOUT 구조체에는 하드웨어 비디오 포트의 필드 번호가 포함됩니다.
DDGETIRQINFO DDGETIRQINFO 구조체에는 비디오 미니포트 드라이버에 대한 인터럽트 정보가 포함됩니다.
DDGETKERNELCAPSOUT DDGETKERNELCAPSOUT 구조체에는 Microsoft DirectDraw 개체의 기능이 포함되어 있습니다.
DDGETPOLARITYIN DDGETPOLARITYIN 구조체에는 Microsoft DirectDraw 및 VPE(비디오 포트 확장) 개체 핸들이 포함됩니다.
DDGETPOLARITYININFO DDGETPOLARITYININFO 구조체에는 VPE(비디오 포트 확장) 개체 정보가 포함됩니다.
DDGETPOLARITYOUT DDGETPOLARITYOUT 구조체에는 요청된 극성 정보가 포함됩니다.
DDGETPOLARITYOUTINFO DDGETPOLARITYOUTINFO 구조체는 VPE(비디오 포트 확장) 개체의 극성 정보를 포함합니다.
DDGETPREVIOUSAUTOFLIPININFO DDGETPREVIOUSAUTOFLIPININFO 구조체는 VPE(비디오 포트 확장) 개체 정보를 제공합니다.
DDGETPREVIOUSAUTOFLIPOUTINFO DDGETPREVIOUSAUTOFLIPOUTINFO 구조체는 표면 데이터를 제공합니다.
DDGETSURFACESTATEIN DDGETSURFACESTATEIN 구조체에는 Microsoft DirectDraw 및 DirectDraw 표면 핸들 정보가 포함됩니다.
DDGETSURFACESTATEOUT DDGETSURFACESTATEOUT 구조체에는 지정된 표면의 기능과 상태가 포함됩니다.
DDGETTRANSFERSTATUSOUTINFO DDGETTRANSFERSTATUSOUTINFO 구조체에는 전송 상태 정보가 포함됩니다.

DDGETVERSIONNUMBER DDGETVERSIONNUMBER 구조체에는 비디오 미니포트 드라이버의 DxApi 인터페이스에서 지원하는 Microsoft DirectDraw의 커널 모드 비디오 전송 구성 요소의 버전 번호가 포함되어 있습니다.
DDLOCKIN DDLOCKIN 구조에는 Microsoft DirectDraw 개체 및 DirectDraw 표면 핸들 정보가 포함됩니다.
DDLOCKININFO DDLOCKININFO 구조체에는 표면 정보가 포함됩니다.
DDLOCKOUT DDLOCKOUT 구조체에는 표면에 대한 설명이 포함되어 있습니다.
DDLOCKOUTINFO DDLOCKOUTINFO 구조체에는 DxLock 함수의 표면 정보 출력이 포함됩니다.
DDMOCOMPBUFFERINFO DDMOCOMPBUFFERINFO 구조체는 프레임을 렌더링하는 데 필요한 매크로 블록 정보를 포함하고 이 정보를 DD_RENDERMOCOMPDATA 구조체에 전달합니다.
DDOPENDIRECTDRAWIN DDOPENDIRECTDRAWIN 구조체에는 Microsoft DirectDraw 개체 정보가 포함됩니다.
DDOPENDIRECTDRAWOUT DDOPENDIRECTDRAWOUT의 ddRVal 멤버가 DD_OK 설정된 경우 DDOPENDIRECTDRAWOUT 구조체에는 DxApi 함수의 DD_DXAPI_OPENDIRECTDRAW 함수 식별자에 대한 새 Microsoft DirectDraw 핸들이 포함됩니다.
DDOPENSURFACEIN DDOPENSURFACEIN 구조체에는 DirectDrawSurface 개체 정보가 포함됩니다.
DDOPENSURFACEOUT DDOPENSURFACEOUT의 ddRVal 멤버가 DD_OK 설정된 경우 DDOPENSURFACEOUT 구조체에는 새 DirectDrawSurface 핸들이 포함됩니다. 이 새 핸들은 DirectDrawSurface 핸들이 필요한 모든 후속 호출에서 사용해야 합니다.
DDOPENVIDEOPORTIN DDOPENVIDEOPORTIN 구조체에는 VPE(비디오 포트 확장) 개체 정보가 포함됩니다.

DDOPENVIDEOPORTOUT ddRVal이 DD_OK 설정된 경우 DDOPENVIDEOPORTOUT 구조에는 Microsoft DirectDraw 반환 코드와 새 표면 핸들이 포함됩니다. 이 새 핸들은 VPE(비디오 포트 확장) 개체 핸들이 필요한 모든 후속 호출에서 사용해야 합니다.
DDOPENVPCAPTUREDEVICEIN DDOPENVPCAPTUREDEVICEIN 구조에는 VPE(비디오 포트 확장) 캡처 정보가 포함됩니다.
DDOPENVPCAPTUREDEVICEOUT DDOPENVPCAPTUREDEVICEOUT 구조체에는 VPE(비디오 포트 확장) 캡처 핸들이 포함됩니다.
DDREGISTERCALLBACK DDREGISTERCALLBACK 구조에는 레지스터 콜백 정보가 포함됩니다. 이 구조체는 DxApi 함수의 DD_DXAPI_REGISTER_CALLBACK 및 DD_DXAPI_UNREGISTER_CALLBACK 함수 식별자 모두에서 사용됩니다.
DDSETFIELDNUM DDSETFIELDNUM 구조체에는 핸들과 필드 번호가 포함됩니다.
DDSETSKIPFIELD DDSETSKIPFIELD 구조체에는 시작 필드 정보가 포함됩니다.
DDSETSTATEININFO DDSETSTATEININFO 구조체에는 표면 및 VPE(비디오 포트 확장) 개체 정보가 포함됩니다.
DDSETSTATEOUTINFO DDSETSTATEOUTINFO 구조체에는 VPE(비디오 포트 확장) 개체에 대한 상태 정보가 포함됩니다.
DDSETSURFACESTATE DDSETSURFACESTATE 구조체에는 표면 상태 정보가 포함됩니다.
DDSKIPNEXTFIELDINFO DDSKIPNEXTFIELDINFO 구조체에는 VPE(비디오 포트 확장) 개체에 대한 건너뛰기 정보가 포함됩니다.
DDSURFACEDATA DDSURFACEDATA 구조체는 DirectDraw에서 커널 모드 미니포트 드라이버의 표면을 나타내는 데 사용됩니다.

DDTRANSFERININFO DDTRANSFERININFO 구조체는 표면에 대한 전송 정보를 포함합니다.
DDTRANSFEROUTINFO DDTRANSFEROUTINFO 구조체는 캡처되는 필드의 극성을 반환합니다.
DDVIDEOPORTDATA DDVIDEOPORTDATA 구조는 DirectDraw에서 커널 모드 비디오 미니포트 드라이버에 대한 VPE(비디오 포트 확장) 개체를 나타내는 데 사용됩니다.
DEVHTADJDATA DEVHTADJDATA 구조체는 HTUI_DeviceColorAdjustment 함수에 대한 입력으로 사용됩니다.
DEVHTINFO DEVHTINFO 구조체는 HTUI_DeviceColorAdjustment 함수에 대한 입력으로 사용됩니다.
DEVINFO DEVINFO 구조체는 그래픽 엔진에 드라이버 및 해당 개인 PDEV에 대한 정보를 제공합니다.
DEVMODEW DEVMODEW 구조는 유니코드(와이드) 문자 집합에서 디스플레이 및 인쇄 디바이스의 특성을 지정하는 데 사용됩니다.
DISPLAYCONFIG_2DREGION DISPLAYCONFIG_2DREGION 구조체는 2차원 공간의 점 또는 오프셋을 나타냅니다.
DISPLAYCONFIG_ADAPTER_NAME DISPLAYCONFIG_ADAPTER_NAME 구조에는 디스플레이 어댑터에 대한 정보가 포함되어 있습니다.
DISPLAYCONFIG_DESKTOP_IMAGE_INFO DISPLAYCONFIG_DESKTOP_IMAGE_INFO 구조체에는 바탕 화면에 표시되는 이미지에 대한 정보가 포함됩니다.
DISPLAYCONFIG_DEVICE_INFO_HEADER DISPLAYCONFIG_DEVICE_INFO_HEADER 구조에는 디바이스에 대한 표시 정보가 포함됩니다.

DISPLAYCONFIG_MODE_INFO DISPLAYCONFIG_MODE_INFO 구조에는 원본 모드 또는 대상 모드 정보가 포함됩니다.
DISPLAYCONFIG_PATH_INFO DISPLAYCONFIG_PATH_INFO 구조체는 대상에서 원본으로의 단일 경로를 설명하는 데 사용됩니다.
DISPLAYCONFIG_PATH_SOURCE_INFO DISPLAYCONFIG_PATH_SOURCE_INFO 구조에는 단일 경로에 대한 원본 정보가 포함됩니다.
DISPLAYCONFIG_PATH_TARGET_INFO DISPLAYCONFIG_PATH_TARGET_INFO 구조에는 단일 경로에 대한 대상 정보가 포함됩니다.
DISPLAYCONFIG_RATIONAL DISPLAYCONFIG_RATIONAL 구조체는 비디오 모드(즉, 세로 동기화 및 가로 동기화)의 세로 및 가로 주파수를 나타내는 소수 자릿수 값을 설명합니다.
DISPLAYCONFIG_SET_TARGET_PERSISTENCE DISPLAYCONFIG_SET_TARGET_PERSISTENCE 구조에는 디스플레이 설정에 대한 정보가 포함되어 있습니다.
DISPLAYCONFIG_SOURCE_DEVICE_NAME DISPLAYCONFIG_SOURCE_DEVICE_NAME 구조에는 원본 또는 뷰에 대한 GDI 디바이스 이름이 포함됩니다.
DISPLAYCONFIG_SOURCE_MODE DISPLAYCONFIG_SOURCE_MODE 구조체는 2차원 공간의 점 또는 오프셋을 나타냅니다.
DISPLAYCONFIG_SUPPORT_VIRTUAL_RESOLUTION DISPLAYCONFIG_SUPPORT_VIRTUAL_RESOLUTION 구조에는 모니터에 대한 가상 해상도 지원 상태에 대한 정보가 포함되어 있습니다.
DISPLAYCONFIG_TARGET_BASE_TYPE 지정된 대상 ID에 대한 기본 출력 기술 정보를 지정합니다.
DISPLAYCONFIG_TARGET_DEVICE_NAME DISPLAYCONFIG_TARGET_DEVICE_NAME 구조에는 대상에 대한 정보가 포함됩니다.

DISPLAYCONFIG_TARGET_DEVICE_NAME_FLAGS DISPLAYCONFIG_TARGET_DEVICE_NAME_FLAGS 구조에는 대상 디바이스에 대한 정보가 포함됩니다.
DISPLAYCONFIG_TARGET_MODE DISPLAYCONFIG_TARGET_MODE 구조체는 표시 경로 대상 모드를 설명합니다.
DISPLAYCONFIG_TARGET_PREFERRED_MODE DISPLAYCONFIG_TARGET_PREFERRED_MODE 구조체에는 디스플레이의 기본 설정 모드에 대한 정보가 포함되어 있습니다.
DISPLAYCONFIG_VIDEO_SIGNAL_INFO DISPLAYCONFIG_VIDEO_SIGNAL_INFO 구조에는 디스플레이의 비디오 신호에 대한 정보가 포함되어 있습니다.
DisplayMode DisplayMode 구조는 디스플레이 디바이스를 설명합니다.
DisplayModes DisplayModes 구조에는 디스플레이 모드 목록이 포함되어 있습니다.
DRIVEROBJ DRIVEROBJ 구조는 GDI 서비스를 사용해야 하는 드라이버에서 할당한 리소스를 추적하는 데 사용됩니다.
DRVENABLEDATA DRVENABLEDATA 구조체에는 DRVFN 구조의 배열과 NT 기반 운영 체제의 그래픽 DDI 버전 번호에 대한 포인터가 포함되어 있습니다.
DRVFN DRVFN 구조는 드라이버에서 정의한 그래픽 DDI 함수에 대한 포인터를 GDI에 제공하기 위해 그래픽 드라이버에서 사용됩니다.
DX_IRQDATA DX_IRQDATA 구조에는 드라이버에서 제공하는 IRQ 정보가 포함됩니다.

DXAPI_INTERFACE DXAPI_INTERFACE 구조에는 비디오 미니포트 드라이버가 Kernel-Mode 비디오 전송을 지원하기 위해 구현하는 인터페이스 콜백 함수가 포함되어 있습니다.
ENG_TIME_FIELDS ENG_TIME_FIELDS 구조체는 EngQueryLocalTime 함수에서 현지 시간을 반환하는 데 사용됩니다.
ENGSAFESEMAPHORE ENGSAFESEMAPHORE 구조체는 드라이버에 스레드로부터 안전한 세마포를 제공합니다.
열거형 ENUMRECTS 구조체는 CLIPOBJ_cEnumStart 함수에서 CLIPOBJ_bEnum 함수에 대한 클립 영역의 사각형에 대한 정보를 제공하는 데 사용됩니다.
FD_DEVICEMETRICS FD_DEVICEMETRICS 구조체는 드라이버 제공 DrvQueryFontData 함수의 iMode 매개 변수가 QFD_MAXEXTENTS 경우 GDI에 디바이스별 글꼴 정보를 제공하는 데 사용됩니다.
FD_GLYPHATTR FD_GLYPHATTR 구조체는 FONTOBJ_pQueryGlyphAttrs 및 DrvQueryGlyphAttrs 함수의 반환 값을 지정하는 데 사용됩니다.
FD_GLYPHSET FD_GLYPHSET 구조체는 유니코드 문자에서 문자 모양 핸들로의 매핑을 정의하는 데 사용됩니다.
FD_KERNINGPAIR FD_KERNINGPAIR 구조체는 커닝 쌍에 대한 정보를 저장하는 데 사용됩니다.
FD_XFORM FD_XFORM 구조체는 임의의 2차원 글꼴 변환을 설명합니다.
FLOATOBJ FLOATOBJ 구조체는 부동 소수점 숫자를 에뮬레이트하는 데 사용됩니다.
FLOATOBJ_XFORM FLOATOBJ_XFORM 구조체는 기하학적 와이드 라인과 같은 임의의 선형 2차원 변환을 설명합니다. (FLOATOBJ_XFORM)

FONTDIFF FONTDIFF 구조는 기본 글꼴과 시뮬레이션 중 하나 간에 다른 모든 특성을 설명합니다.
Fontinfo FONTINFO 구조체에는 특정 글꼴에 대한 정보가 포함됩니다.
FONTOBJ FONTOBJ 구조체는 드라이버가 글꼴의 특정 인스턴스에 대한 정보에 액세스할 수 있도록 하는 데 사용됩니다.
FONTSIM FONTSIM 구조체에는 굵게, 기울임꼴 및 굵은 기울임꼴 글꼴 시뮬레이션을 설명하는 하나 이상의 FONTDIFF 구조에 대한 오프셋이 포함되어 있습니다.
GAMMARAMP GAMMARAMP 구조체는 DrvIcmSetDeviceGammaRamp에서 특정 디스플레이 디바이스의 하드웨어 감마 램프를 설정하는 데 사용됩니다.
GDIINFO GDIINFO 구조는 지정된 디바이스의 그래픽 기능을 설명합니다.
문자 모양 GLYPHBITS 구조체는 문자 모양 비트맵을 정의하는 데 사용됩니다.
GLYPHDATA GLYPHDATA 구조체에는 개별 문자 모양에 대한 정보가 포함되어 있습니다.
GLYPHDEF GLYPHDEF 공용 구조체는 개별 문자 모양을 식별하고 GLYPHBITS 구조체에 대한 포인터 또는 PATHOBJ 구조체에 대한 포인터를 제공합니다.
문자 모양 GLYPHPOS 구조체는 GDI에서 문자 모양의 설명과 위치를 그래픽 드라이버에 제공하는 데 사용됩니다.
HEAPALIGNMENT HEAPALIGNMENT 구조에는 지정된 디스플레이 메모리 힙에 대한 맞춤 요구 사항을 지정하는 데이터가 포함됩니다.

IFIEXTRA IFIEXTRA 구조체는 GDI에서 사용할 수 있는 지정된 서체에 대한 추가 정보를 정의합니다.
IFIMETRICS IFIMETRICS 구조체는 GDI에서 사용할 수 있는 지정된 서체에 대한 정보를 정의합니다.
LINEATTRS LINEATTRS 구조체는 드라이버의 선 그리기 함수에서 선 특성을 확인하는 데 사용됩니다.
PALOBJ PALOBJ 구조체는 인덱싱된 색상표를 나타내는 사용자 개체입니다.
PATHDATA PATHDATA 구조체는 하위 경로의 전부 또는 일부를 설명합니다.
PATHOBJ PATHOBJ 구조체는 스트로크 또는 채워질 선 및 베지어 곡선 집합을 설명하는 데 사용됩니다.
PERBANDINFO PERBANDINFO 구조체는 프린터 그래픽 DLL의 DrvQueryPerBandInfo 함수에 대한 입력으로 사용됩니다.
지점 POINT 구조는 점의 x 및 y 좌표를 정의합니다.
POINTL POINTL 구조체는 점의 x 및 y 좌표를 정의합니다.
포인트 POINTS 구조는 점의 x 및 y 좌표를 정의합니다.
Rect RECT 구조체는 왼쪽 위와 오른쪽 아래 모서리의 좌표로 사각형을 정의합니다.
RECTL RECTL 구조체는 왼쪽 위와 오른쪽 아래 모서리의 좌표로 사각형을 정의합니다.

실행 RUN 구조체는 CLIPLINE 구조에 의해 잘리지 않는 선형 픽셀 집합을 설명하는 데 사용됩니다.
SIZE SIZE 구조는 사각형의 너비와 높이를 정의합니다.
Sources 원본 구조에는 VidPN(Video Present Network) 토폴로지가 포함되어 있습니다.
STROBJ STROBJ 클래스 또는 텍스트 문자열 개체는 디바이스 드라이버에 대한 문자 모양 핸들 및 위치의 열거형을 포함합니다.
SURFACEALIGNMENT SURFACEALIGNMENT 구조는 디스플레이 드라이버에서 HeapVidMemAllocAligned에 의해 할당되는 표면에 대한 맞춤 제한을 설명하는 데 사용됩니다.
SURFOBJ SURFOBJ 구조체는 표면의 사용자 개체입니다. 디바이스 드라이버는 일반적으로 Surface 개체가 GDI 비트맵 또는 디바이스 관리형 표면을 나타내는 경우에만 Surface 개체에서 메서드를 호출합니다.
TYPE1_FONT TYPE1_FONT 구조에는 PostScript 드라이버가 GDI를 통해 Type1 글꼴에 액세스하는 데 필요한 정보가 포함되어 있습니다.
VIDEOMEMORY VIDEOMEMORY 구조를 사용하면 드라이버가 디스플레이 메모리를 힙으로 관리할 수 있습니다.
VIDEOMEMORYINFO VIDEOMEMORYINFO 구조체는 디스플레이 메모리의 일반적인 형식을 설명합니다.
VIDEOPARAMETERS 비디오 미니포트 드라이버는 IOCTL 요청이 IOCTL_VIDEO_HANDLE_VIDEOPARAMETERS 때 VIDEO_REQUEST_PACKET InputBuffer 멤버의 VIDEOPARAMETERS 구조체에 대한 포인터를 받습니다.

VMEMHEAP VMEMHEAP 구조체에는 힙에 대한 정보가 포함됩니다.
WCRUN WCRUN 구조는 유니코드 문자의 실행을 설명합니다.
WNDOBJ WNDOBJ 구조를 사용하면 드라이버가 창의 위치, 크기 및 표시되는 클라이언트 영역 변경 내용을 추적할 수 있습니다.
XFORML FLOATOBJ_XFORM 구조체는 기하학적 와이드 선과 같은 임의의 선형 2차원 변환을 설명합니다. F
XFORMOBJ XFORMOBJ 구조체는 기하학적 와이드 선과 같은 임의의 선형 2차원 변환을 설명합니다.
XLATEOBJ XLATEOBJ 구조체는 색상표에서 다른 색상표로 색 인덱스를 변환하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

분산 파일 시스템

아티클2023. 03. 09.

분산 파일 시스템 기술 개요입니다.

분산 파일 시스템을 개발하려면 다음 헤더가 필요합니다.

- lmdfs.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 분산 파일 시스템

열거형

DFS_NAMESPACE_VERSION_ORIGIN DFS 네임스페이스 버전 정보의 원본을 식별합니다.

Functions

NetDfsAdd 새 DFS(분산 파일 시스템) 링크를 만들거나 DFS 네임스페이스의 기존 링크에 대상을 추가합니다.
NetDfsAddFtRoot 새 도메인 기반 DFS(분산 파일 시스템) 네임스페이스를 만듭니다. 네임스페이스가 이미 있는 경우 함수는 지정된 루트 대상을 추가합니다.
NetDfsAddRootTarget 도메인 기반 또는 독립 실행형 DFS 네임스페이스를 만들거나 기존 도메인 기반 네임스페이스에 새 루트 대상을 추가합니다.
NetDfsAddStdRoot 새 독립 실행형 DFS(분산 파일 시스템) 네임스페이스를 만듭니다.

NetDfsEnum 서버에서 호스트하는 네임스페이스의 DFS 링크 또는 서버에서 호스트되는 DFS(분산 파일 시스템) 네임스페이스를 열거합니다.
NetDfsGetClientInfo DFS 클라이언트에서 유지 관리하는 캐시에서 DFS(분산 파일 시스템) 루트 또는 링크에 대한 정보를 검색합니다.
NetDfsGetFtContainerSecurity 지정된 Active Directory 도메인의 도메인 기반 DFS 네임스페이스에 대한 컨테이너 개체의 보안 설명자를 검색합니다.
NetDfsGetInfo DFS 네임스페이스에서 지정된 DFS(분산 파일 시스템) 루트 또는 링크에 대한 정보를 검색합니다.
NetDfsGetSecurity 지정된 DFS 네임스페이스의 루트 개체에 대한 보안 설명자를 검색합니다.
NetDfsGetStdContainerSecurity 지정된 독립 실행형 DFS 네임스페이스의 컨테이너 개체에 대한 보안 설명자를 검색합니다.
NetDfsGetSupportedNamespaceVersion 지원되는 메타데이터 버전 번호를 결정합니다.
NetDfsMove DFS 링크의 이름을 바꾸거나 이동합니다.
NetDfsRemove DFS(분산 파일 시스템) 링크 또는 DFS 네임스페이스에서 DFS 링크의 특정 링크 대상을 제거합니다. 특정 링크 대상을 제거할 때 링크의 마지막 링크 대상이 제거되면 링크 자체가 제거됩니다.
NetDfsRemoveFtRoot 도메인 기반 DFS(분산 파일 시스템) 네임스페이스에서 지정된 루트 대상을 제거합니다.
NetDfsRemoveFtRootForced 루트 대상 서버가 오프라인인 경우에도 도메인 기반 DFS(분산 파일 시스템) 네임스페이스에서 지정된 루트 대상을 제거합니다.

NetDfsRemoveRootTarget 도메인 기반 DFS 네임스페이스에서 DFS 루트 대상을 제거합니다. 루트 대상이 DFS 네임스페이스의 마지막 루트 대상인 경우 이 함수는 DFS 네임스페이스를 제거합니다. 이 함수를 사용하여 독립 실행형 DFS 네임스페이스를 제거할 수도 있습니다.
NetDfsRemoveStdRoot 독립 실행형 DFS(분산 파일 시스템) 네임스페이스를 삭제합니다.
NetDfsSetClientInfo DFS 클라이언트에서 유지 관리하는 캐시의 DFS(분산 파일 시스템) 루트 또는 링크에 대한 정보를 수정합니다.
NetDfsSetFtContainerSecurity 지정된 Active Directory 도메인의 도메인 기반 DFS 네임스페이스에 대한 컨테이너 개체의 보안 설명자를 설정합니다.
NetDfsSetInfo 특정 DFS(분산 파일 시스템) 루트, 루트 대상, 링크 또는 링크 대상에 대한 정보를 설정하거나 수정합니다.
NetDfsSetSecurity 지정된 DFS 네임스페이스의 루트 개체에 대한 보안 설명자를 설정합니다.
NetDfsSetStdContainerSecurity 지정된 독립 실행형 DFS 네임스페이스의 컨테이너 개체에 대한 보안 설명자를 설정합니다.

구조체

DFS_GET_PKT_ENTRY_STATE_ARG FSCTL_DFS_GET_PKT_ENTRY_STATE 제어 코드와 함께 사용되는 입력 버퍼입니다.
DFS_INFO_1 DFS(분산 파일 시스템) 루트 또는 링크의 이름을 포함합니다.

DFS_INFO_100 DFS(분산 파일 시스템) 루트 또는 링크와 연결된 주석을 포함합니다.
DFS_INFO_101 DFS 루트, 링크, 루트 대상 또는 링크 대상의 스토리지 상태에 대해 설명합니다.
DFS_INFO_102 DFS(분산 파일 시스템) 루트 또는 명명된 DFS 루트의 링크와 연결할 제한 시간 값을 포함합니다.
DFS_INFO_103 DFS 루트 또는 링크에 대한 특정 동작을 설정하는 속성을 포함합니다.
DFS_INFO_104 DFS 루트 대상 또는 링크 대상의 우선 순위를 포함합니다.
DFS_INFO_105 속성 플래그로 지정된 주석, 상태, 제한 시간 및 DFS 동작을 포함하여 DFS 루트 또는 링크에 대한 정보를 포함합니다.
DFS_INFO_106 DFS 루트 대상 또는 링크 대상에 대한 스토리지 상태 및 우선 순위를 포함합니다. 이 구조체는 NetDfsSetInfo 함수에만 사용됩니다.
DFS_INFO_107 주석, 상태, 제한 시간, 속성 플래그 및 링크 재문 분석 지점 보안 설명자를 포함하여 DFS 루트 또는 링크에 대한 정보를 포함합니다.
DFS_INFO_150 DFS 링크의 재문 분석 지점에 대한 보안 설명자를 포함합니다.
DFS_INFO_2 DFS(분산 파일 시스템) 루트 또는 링크에 대한 정보를 포함합니다. 이 구조에는 루트 또는 링크의 이름, 상태 및 DFS 대상 수가 포함됩니다.
DFS_INFO_200 도메인 기반 DFS(분산 파일 시스템) 네임스페이스의 이름을 포함합니다.

DFS_INFO_3 DFS(분산 파일 시스템) 루트 또는 링크에 대한 정보를 포함합니다. 이 구조에는 이름, 상태, DFS 대상 수 및 루트 또는 링크의 각 대상에 대한 정보가 포함됩니다.
DFS_INFO_300 DFS 네임스페이스의 이름 및 형식(도메인 기반 또는 독립 실행형)을 포함합니다.
DFS_INFO_4 DFS(분산 파일 시스템) 루트 또는 링크에 대한 정보를 포함합니다. 이 구조에는 이름, 상태, GUID, 제한 시간, 대상 수 및 루트 또는 링크의 각 대상에 대한 정보가 포함됩니다.
DFS_INFO_5 DFS(분산 파일 시스템) 루트 또는 링크에 대한 정보를 포함합니다. 이 구조체에는 이름, 상태, GUID, 제한 시간, 네임스페이스/루트/링크 속성, 메타데이터 크기 및 루트 또는 링크의 대상 수가 포함됩니다.
DFS_INFO_50 기존 DFS 네임스페이스의 DFS 메타데이터 버전 및 기능을 포함합니다.
DFS_INFO_6 DFS(분산 파일 시스템) 루트 또는 링크에 대한 정보를 포함합니다. 이 구조에는 이름, 상태, GUID, 제한 시간, 네임스페이스/루트/링크 속성, 메타데이터 크기, 대상 수 및 루트 또는 링크의 각 대상에 대한 정보가 포함됩니다.
DFS_INFO_7 DFS 네임스페이스에 대한 정보를 포함합니다. 이 구조에는 네임스페이스에 대한 메타데이터에 대한 버전 GUID가 포함됩니다.
DFS_INFO_8 이름, 상태, GUID, 제한 시간, 속성 플래그, 메타데이터 크기, DFS 대상 정보 및 루트 또는 링크에 대한 링크 재분석 지점 보안 설명자를 포함합니다.
DFS_INFO_9 이름, 상태, GUID, 제한 시간, 속성 플래그, 메타데이터 크기, DFS 대상 정보, 링크 재분석 지점 보안 설명자 및 루트 또는 링크에 대한 DFS 대상 목록을 포함합니다.
DFS_STORAGE_INFO DFS 네임스페이스 또는 DFS 클라이언트에서 유지 관리하는 캐시의 DFS 루트 또는 링크 대상에 대한 정보를 포함합니다.

DFS_STORAGE_INFO_1 DFS 대상 서버 이름 및 공유 이름과 대상의 상태 및 우선 순위를 포함하여 DFS 대상에 대한 정보를 포함합니다.
DFS_SUPPORTED_NAMESPACE_VERSION_INFO DFS 네임스페이스에 대한 버전 정보를 포함합니다.
DFS_TARGET_PRIORITY 특정 DFS 대상의 우선 순위 클래스 및 순위를 포함합니다.

분산 트랜잭션 코디네이터 WMI 공급자

아티클2023. 03. 09.

분산 트랜잭션 코디네이터 WMI 공급자 기술 개요입니다.

분산 트랜잭션 코디네이터 WMI 공급자를 개발하려면 다음 헤더가 필요합니다.

- wbemcli.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 분산 트랜잭션 코디네이터 WMI 공급자

DNS(도메인 이름 시스템)

아티클2025. 01. 07.

DNS(도메인 이름 시스템) 기술 개요입니다.

DNS(도메인 이름 시스템)를 개발하려면 다음 헤더가 필요합니다.

- windns.h
- windnsdef.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- DNS(도메인 이름 시스템)

열거형

테이블 확장

DNS_CHARSET DNS_CHARSET 열거형은 사용된 문자 집합을 지정합니다.
DNS_CONFIG_TYPE DNS_CONFIG_TYPE 열거형은 DNS 구성 형식 정보를 제공합니다.
DNS_FREE_TYPE DNS_FREE_TYPE 열거형은 해제할 데이터 형식을 지정합니다.
DNS_NAME_FORMAT DNS_NAME_FORMAT 열거형은 DNS의 이름 형식 정보를 지정합니다.
DNS_PROXY_INFORMATION_TYPE DNS_PROXY_INFORMATION_TYPE 열거형은 DNS_PROXY_INFORMATION 구조체의 프록시 정보 형식을 정의합니다.
DNS_SECTION DNS_SECTION 열거형은 레코드 플래그 및 DNS 와이어 메시지 헤더 섹션 개수에 대한 인덱스로 사용됩니다.

함수

테이블 확장

DNS_QUERY_COMPLETION_ROUTINE DNS_QUERY_COMPLETION_ROUTINE 콜백은 DNS 쿼리의 결과를 비동기적으로 반환하는 데 사용됩니다.
DNS_QUERY_RAW_COMPLETION_ROUTINE 구현하는 비동기 콜백 함수의 함수 서명입니다. 시스템은 DnsQueryRaw를 호출하여 시작한 쿼리 결과를 사용하여 구현을 호출합니다.
DNS_SERVICE_BROWSE_CALLBACK DNS-SD 쿼리의 결과를 비동기적으로 반환하는 데 사용됩니다.
DNS_SERVICE_REGISTER_COMPLETE 서비스 등록이 완료되었음을 애플리케이션에 알리는 데 사용됩니다.
DNS_SERVICE_RESOLVE_COMPLETE 서비스 확인 작업의 결과를 비동기적으로 반환하는 데 사용됩니다.
DnsAcquireContextHandle_A DnsAcquireContextHandle 함수 형식은 자격 증명 집합에 대한 컨텍스트 핸들을 획득합니다. (ANSI)
DnsAcquireContextHandle_W DnsAcquireContextHandle 함수 형식은 자격 증명 집합에 대한 컨텍스트 핸들을 획득합니다. (유니코드)
dnsCancelQuery DnsCancelQuery 함수를 사용하여 DNS 네임스페이스에 대한 보류 중인 쿼리를 취소할 수 있습니다.
dnsCancelQueryRaw DnsQueryRaw를 호출하여 시작된 쿼리를 취소합니다.
DnsExtractRecordsFromMessage_UTF8

DnsExtractRecordsFromMessage 함수 형식은 DNS 메시지에서 RR(리소스 레코드)을 추출하고 해당 레코드를 DNS_RECORD 구조에 저장합니다. (DnsExtractRecordsFromMessage_UTF8)
DnsExtractRecordsFromMessage_W DnsExtractRecordsFromMessage 함수 형식은 DNS 메시지에서 RR(리소스 레코드)을 추출하고 해당 레코드를 DNS_RECORD 구조에 저장합니다. (DnsExtractRecordsFromMessage_W)
dnsFree DnsQuery 함수를 사용하여 가져온 DNS 레코드에 할당된 메모리를 해제합니다.
dnsFreeCustomServers DnsGetApplicationSettings이전 호출에서 반환된 사용자 지정 서버의 배열을 해제합니다.
dnsFreeProxyName DnsGetProxyInformation 함수를 사용하여 가져온 DNS_PROXY_INFORMATION 구조체의 proxyName 멤버에 할당된 메모리를 해제합니다.
dnsGetApplicationSettings 애플리케이션별 DNS 설정을 검색합니다.
dnsGetProxyInformation DnsGetProxyInformation 함수는 DNS 서버의 이름 확인 정책 테이블에 대한 프록시 정보를 반환합니다.
DnsModifyRecordsInSet_A 함수는 이전에 DNS 서버에 등록되었을 수 있는 RR(리소스 레코드) 집합을 추가, 수정 또는 제거합니다. (DnsModifyRecordsInSet_A)
DnsModifyRecordsInSet_UTF8 함수는 이전에 DNS 서버에 등록되었을 수 있는 RR(리소스 레코드) 집합을 추가, 수정 또는 제거합니다. (DnsModifyRecordsInSet_UTF8)
DnsModifyRecordsInSet_W 함수는 이전에 DNS 서버에 등록되었을 수 있는 RR(리소스 레코드) 집합을 추가, 수정 또는 제거합니다. (DnsModifyRecordsInSet_W)
dnsNameCompare DnsNameCompare 함수는 두 개의 DNS 이름을 비교합니다. (DnsNameCompare_UTF8)

DnsNameCompare_A DnsNameCompare 함수는 두 개의 DNS 이름을 비교합니다. (DnsNameCompare_A)
DnsNameCompare_W DnsNameCompare 함수는 두 개의 DNS 이름을 비교합니다. (DnsNameCompare_W)
DnsQuery_A DNS 네임스페이스에 대한 일반 쿼리 인터페이스이며 애플리케이션 개발자에게 DNS 쿼리 확인 인터페이스를 제공합니다. (DnsQuery_A)
DnsQuery_UTF8 DNS 네임스페이스에 대한 일반 쿼리 인터페이스이며 애플리케이션 개발자에게 DNS 쿼리 확인 인터페이스를 제공합니다. (DnsQuery_UTF8)
DnsQuery_W DNS 네임스페이스에 대한 일반 쿼리 인터페이스이며 애플리케이션 개발자에게 DNS 쿼리 확인 인터페이스를 제공합니다. (DnsQuery_W)
dnsQueryConfig DnsQueryConfig 함수를 사용하면 애플리케이션 프로그래머가 로컬 컴퓨터 또는 특정 어댑터의 구성을 쿼리할 수 있습니다.
dnsQueryEx DNS 네임스페이스에 대한 비동기 제네릭 쿼리 인터페이스이며 애플리케이션 개발자에게 DNS 쿼리 확인 인터페이스를 제공합니다.
dnsQueryRaw DNS 쿼리를 포함하는 원시 패킷이나 쿼리 이름 및 형식을 허용하는 DNS 쿼리를 수행할 수 있습니다.
dnsQueryRawResultFree DNS_QUERY_RAW_RESULT 구조체 개체에 할당된 메모리를 해제합니다.
dnsRecordCompare DnsRecordCompare 함수는 두 개의 RR(DNS 리소스 레코드)을 비교합니다.
dnsRecordCopyEx

DnsRecordCopyEx 함수는 지정된 RR(리소스 레코드)의 복사본을 만듭니다. DnsRecordCopyEx 함수는 복사 작업 중에 문자 인코딩을 변환할 수도 있습니다.
DnsRecordListFree DnsQuery 함수를 사용하여 가져온 DNS 레코드에 할당된 메모리를 해제합니다.
dnsRecordSetCompare DnsRecordSetCompare 함수는 두 개의 RR 집합을 비교합니다.
dnsRecordSetCopyEx DnsRecordSetCopyEx 함수는 지정된 리소스 레코드 집합의 복사본을 만듭니다. DnsRecordSetCopyEx 함수는 복사 작업 중에 문자 인코딩을 변환할 수도 있습니다.
dnsRecordSetDetach DnsRecordSetDetach 함수는 지정된 DNS 레코드 목록에서 첫 번째 레코드 집합을 분리합니다.
dnsReleaseContextHandle DnsReleaseContextHandle 함수는 특정 계정의 자격 증명을 저장하는 데 사용되는 메모리를 해제합니다.
dnsReplaceRecordSetA 기존 RR(리소스 레코드) 집합을 대체합니다. (DnsReplaceRecordSetA)
dnsReplaceRecordSetUTF8 기존 RR(리소스 레코드) 집합을 대체합니다. (DnsReplaceRecordSetUTF8)
dnsReplaceRecordSetW 기존 RR(리소스 레코드) 집합을 대체합니다. (DnsReplaceRecordSetW)
dnsServiceBrowse 로컬 네트워크에서 실행되는 서비스에 대한 DNS-SD 검색을 시작하는 데 사용됩니다.
dnsServiceBrowseCancel 실행 중인 DNS-SD 검색 쿼리를 취소하는 데 사용됩니다.
dnsServiceConstructInstance 이를 설명하는 데이터에서 DNS_SERVICE_INSTANCE 구조를 빌드하는 데 사용됩니다.

dnsServiceCopyInstance DNS_SERVICE_INSTANCE 구조를 복사하는 데 사용됩니다.
dnsServiceDeRegister 등록된 서비스를 제거하는 데 사용됩니다.
dnsServiceFreeInstance DNS_SERVICE_INSTANCE 구조와 연결된 리소스를 해제하는 데 사용됩니다.
dnsServiceRegister 이 디바이스에서 검색 가능한 서비스를 등록하는 데 사용됩니다. (DnsServiceRegister)
dnsServiceRegisterCancel 보류 중인 등록 작업을 취소하는 데 사용됩니다.
dnsServiceResolve 로컬 네트워크에 보급된 서비스에 대한 자세한 정보를 가져오는 데 사용됩니다.
dnsServiceResolveCancel 실행 중인 DNS-SD 쿼리를 취소하는 데 사용됩니다.
dnsSetApplicationSettings 애플리케이션별 DNS 설정을 구성합니다. 여기에는 애플리케이션별 DNS 서버를 시스템 구성 서버에 대한 대체(fallback)로 설정하거나 단독으로 설정하는 기능이 포함됩니다.
dnsStartMulticastQuery 이 디바이스에서 검색 가능한 서비스를 등록하는 데 사용됩니다. (DnsStartMulticastQuery)
dnsStopMulticastQuery DnsStartMulticastQuery 작업을 실행을 중지하는 데 사용됩니다.
dnsValidateName DnsValidateName 매크로(windns.h) 함수는 지정된 DNS 이름의 상태를 확인합니다.
DnsValidateName_A DnsValidateName 함수는 지정된 DNS 이름의 상태를 확인합니다. (DnsValidateName_A)

DnsValidateName_UTF8 DnsValidateName_UTF8 함수(windns.h) 함수는 지정된 DNS 이름의 상태를 확인합니다.
DnsValidateName_W DnsValidateName 함수는 지정된 DNS 이름의 상태를 확인합니다. (DnsValidateName_W)
dnsValidateServerStatus DnsValidateServerStatus 함수는 IP 주소를 적합한 DNS 서버로 유효성을 검사합니다.
DnsWriteQuestionToBuffer_UTF8 DnsWriteQuestionToBuffer 함수 형식은 DNS 쿼리 메시지를 만들고 DNS_MESSAGE_BUFFER 구조에 저장합니다. (DnsWriteQuestionToBuffer_UTF8)
DnsWriteQuestionToBuffer_W DnsWriteQuestionToBuffer 함수 형식은 DNS 쿼리 메시지를 만들고 DNS_MESSAGE_BUFFER 구조에 저장합니다. (DnsWriteQuestionToBuffer_W)
MDNS_QUERY_CALLBACK mDNS 쿼리의 결과를 비동기적으로 반환하는 데 사용됩니다.

구조

테이블 확장

DNS_A_DATA DNS_A_DATA 구조체는 RFC 1035의 섹션 3.4.1에 지정된 DNS 주소(A) 레코드를 나타냅니다.
DNS_AAAA_DATA DNS_AAAA_DATA 구조체는 RFC 3596에 지정된 대로 AAAA(DNS IPv6) 레코드를 나타냅니다.
DNS_ADDR DNS_ADDR 구조체는 IPv4 또는 IPv6 주소를 저장합니다.
DNS_ADDR_ARRAY

IPv4 또는 IPv6 주소의 배열을 저장합니다.
DNS_APPLICATION_SETTINGS 애플리케이션별 DNS 설정을 나타냅니다.
DNS_ATMA_DATA DNS_ATMA_DATA 구조체는 ATMA(DNS ATM 주소) RR(리소스 레코드)을 나타냅니다.
DNS_CUSTOM_SERVER DNS 사용자 지정 서버를 나타냅니다. DNS_CUSTOM_SERVER 개체는 DNS_QUERY_REQUEST3 구조를 통해 dnsQueryEx 전달됩니다.
DNS_DHCID_DATA RFC 4701 섹션 3에 지정된 DNS DHCID(동적 호스트 구성 프로토콜 정보) RR(리소스 레코드)을 나타냅니다.
DNS_DS_DATA RFC 4034의 섹션 2에 지정된 DS RR(리소스 레코드)을 나타내며 DNS_DNSKEY_DATA 내용을 확인하는 데 사용됩니다.
DNS_HEADER DNS_HEADER 구조에는 RFC 1035의 섹션 4.1.1에 지정된 대로 DNS 메시지를 보낼 때 사용되는 DNS 헤더 정보가 포함됩니다.
DNS_KEY_DATA DNS_KEY_DATA 구조체는 RFC 3445에 지정된 DNS 키(KEY) RR(리소스 레코드)을 나타냅니다.
DNS_LOC_DATA DNS_LOC_DATA 구조체는 RFC 1876에 지정된 대로 RR(DNS 위치) 리소스 레코드를 나타냅니다.
DNS_MESSAGE_BUFFER DNS_MESSAGE_BUFFER 구조체는 DNS 쿼리에 대한 메시지 정보를 저장합니다.
DNS_MINFO_DATAA DNS_MINFO_DATA 구조는 RFC 1035의 섹션 3.3.7에 지정된 대로 MINFO(DNS 메일 정보) 레코드를 나타냅니다. (ANSI)
DNS_MINFO_DATAW

DNS_MINFO_DATA 구조는 RFC 1035의 섹션 3.3.7에 지정된 대로 MINFO(DNS 메일 정보) 레코드를 나타냅니다. (유니코드)
DNS_MX_DATAA DNS_MX_DATA 구조는 RFC 1035의 섹션 3.3.9에 지정된 대로 DNS MX(메일 교환기) 레코드를 나타냅니다. (ANSI)
DNS_MX_DATAW DNS_MX_DATA 구조는 RFC 1035의 섹션 3.3.9에 지정된 대로 DNS MX(메일 교환기) 레코드를 나타냅니다. (유니코드)
DNS_NAPTR_DATAA DNS_NAPTR_DATA 구조체는 RFC 2915에 지정된 대로 NAPTR(명명 기관 포인터) RR(DNS 리소스 레코드)을 나타냅니다. (ANSI)
DNS_NAPTR_DATAW DNS_NAPTR_DATA 구조체는 RFC 2915에 지정된 대로 NAPTR(명명 기관 포인터) RR(DNS 리소스 레코드)을 나타냅니다. (유니코드)
DNS_NSEC_DATAA RFC 4034 섹션 4에 지정된 NSEC RR(리소스 레코드)을 나타냅니다. (ANSI)
DNS_NSEC_DATAW RFC 4034 섹션 4에 지정된 NSEC RR(리소스 레코드)을 나타냅니다. (유니코드)
DNS_NULL_DATA DNS_NULL_DATA 구조체는 RFC 1035 섹션 3.3.10에 지정된 대로 DNS 리소스 레코드에 대한 NULL 데이터를 나타냅니다.
DNS_NXT_DATAA DNS_NXT_DATA 구조체는 RFC 2535 섹션 5에 지정된 대로 DNS 다음(NXT) RR(리소스 레코드)을 나타냅니다. (ANSI)
DNS_NXT_DATAW DNS_NXT_DATA 구조체는 RFC 2535 섹션 5에 지정된 대로 DNS 다음(NXT) RR(리소스 레코드)을 나타냅니다. (유니코드)
DNS_OPT_DATA RFC 2671의 섹션 4에 지정된 대로 DNS 옵션(OPT) RR(리소스 레코드)을 나타냅니다.

DNS_PROXY_INFORMATION DNS 서버의 이름 확인 정책 테이블에 대한 프록시 정보를 포함합니다.
DNS_PTR_DATAA DNS_PTR_DATA 구조체는 RFC 1035의 섹션 3.3.12에 지정된 대로 PTR(DNS 포인터) 레코드를 나타냅니다. (ANSI)
DNS_PTR_DATAW DNS_PTR_DATA 구조체는 RFC 1035의 섹션 3.3.12에 지정된 대로 PTR(DNS 포인터) 레코드를 나타냅니다. (유니코드)
DNS_QUERY_CANCEL DNS_QUERY_CANCEL 구조를 사용하여 비동기 DNS 쿼리를 취소할 수 있습니다.
DNS_QUERY_RAW_CANCEL DNS 원시 쿼리 취소 핸들을 나타냅니다.
DNS_QUERY_RAW_REQUEST DNS 원시 쿼리 요청을 나타냅니다(dnsQueryRaw참조).
DNS_QUERY_RAW_RESULT DNS 원시 쿼리 결과를 나타냅니다(DNS_QUERY_RAW_COMPLETION_ROUTINE참조).
DNS_QUERY_REQUEST DNS_QUERY_REQUEST 구조에는 DnsQueryEx 호출에 사용되는 DNS 쿼리 매개 변수가 포함됩니다.
DNS_QUERY_REQUEST3 DnsQueryEx호출에 사용되는 DNS 쿼리 매개 변수를 포함합니다.
DNS_QUERY_RESULT DNS_QUERY_RESULT 구조에는 DnsQueryEx 호출에서 반환된 DNS 쿼리 결과가 포함됩니다.
DNS_RECORD_FLAGS DNS_RECORD_FLAGS 구조체는 DNS_RECORD 구조체에서 사용할 플래그를 설정하는 데 사용됩니다.
DNS_RECORDA

DNS RR(리소스 레코드)을 저장합니다. (ANSI)
DNS_RECORDW DNS RR(리소스 레코드)을 저장합니다. (유니코드)
DNS_RRSET DNS_RRSET 구조에는 RR(DNS 리소스 레코드) 집합에 대한 정보가 포함됩니다.
DNS_SERVICE_BROWSE_REQUEST DnsServiceBrowse호출에 사용되는 쿼리 매개 변수를 포함합니다.
DNS_SERVICE_CANCEL 비동기 DNS-SD 작업을 취소하는 데 사용됩니다.
DNS_SERVICE_INSTANCE 네트워크에서 실행되는 DNS 서비스를 나타냅니다.
DNS_SERVICE_REGISTER_REQUEST DnsServiceRegister사용하여 서비스를 보급하거나 DnsServiceDeRegister사용하여 광고를 중지하는 데 필요한 정보를 포함합니다.
DNS_SERVICE_RESOLVE_REQUEST DnsServiceResolve호출에 사용되는 쿼리 매개 변수를 포함합니다.
DNS_SIG_DATAA 구조체는 RFC 4034에 지정된 대로 DNSSEC(DNS 보안 확장) SIG(암호화 서명) RR(리소스 레코드)을 나타냅니다. (ANSI)
DNS_SIG_DATAW 구조체는 RFC 4034에 지정된 대로 DNSSEC(DNS 보안 확장) SIG(암호화 서명) RR(리소스 레코드)을 나타냅니다. (유니코드)
DNS_SOA_DATAA DNS_SOA_DATA 구조는 RFC 1035의 섹션 3.3.13에 지정된 대로 SOA(DNS 권한 시작) 레코드를 나타냅니다. (ANSI)
DNS_SOA_DATAW

DNS_SOA_DATA 구조는 RFC 1035의 섹션 3.3.13에 지정된 대로 SOA(DNS 권한 시작) 레코드를 나타냅니다. (유니코드)
DNS_SRV_DATAA DNS_SRV_DATA 구조체는 RFC 2782에 지정된 대로 SRV(DNS 서비스) 레코드를 나타냅니다. (ANSI)
DNS_SRV_DATAW DNS_SRV_DATA 구조체는 RFC 2782에 지정된 대로 SRV(DNS 서비스) 레코드를 나타냅니다. (유니코드)
DNS_TKEY_DATAA DNS_TKEY_DATA 구조는 RFC 2930에 지정된 대로 DNS 확인자와 서버 간에 알고리즘의 공유 비밀 키를 설정하고 삭제하는 데 사용되는 DNS TKEY 리소스 레코드를 나타냅니다. (ANSI)
DNS_TKEY_DATAW DNS_TKEY_DATA 구조는 RFC 2930에 지정된 대로 DNS 확인자와 서버 간에 알고리즘의 공유 비밀 키를 설정하고 삭제하는 데 사용되는 DNS TKEY 리소스 레코드를 나타냅니다. (유니코드)
DNS_TSIG_DATAA DNS_TSIG_DATA 구조는 RFC 2845 및 RFC 3645에 지정된 대로 TSIG(비밀 키 트랜잭션 인증) RR(리소스 레코드)을 나타냅니다. (ANSI)
DNS_TSIG_DATAW DNS_TSIG_DATA 구조는 RFC 2845 및 RFC 3645에 지정된 대로 TSIG(비밀 키 트랜잭션 인증) RR(리소스 레코드)을 나타냅니다. (유니코드)
DNS_TXT_DATAA DNS_TXT_DATA 구조체는 RFC 1035의 섹션 3.3.14에 지정된 대로 TXT(DNS 텍스트) 레코드를 나타냅니다. (ANSI)
DNS_TXT_DATAW DNS_TXT_DATA 구조체는 RFC 1035의 섹션 3.3.14에 지정된 대로 TXT(DNS 텍스트) 레코드를 나타냅니다. (유니코드)
DNS_WINS_DATA DNS_WINS_DATA 구조는 DNS WINS(Windows Internet Name Service) 레코드를 나타냅니다.
DNS_WINSR_DATAA

DNS_WINSR_DATA 구조는 DNS WINDOWS WINDOWS 인터넷 이름 서비스 WINSR(역방향 조회) 레코드를 나타냅니다. (ANSI)
DNS_WINSR_DATAW DNS_WINSR_DATA 구조는 DNS WINDOWS WINDOWS 인터넷 이름 서비스 WINSR(역방향 조회) 레코드를 나타냅니다. (유니코드)
DNS_WIRE_QUESTION DNS_WIRE_QUESTION 구조에는 RFC 1035의 섹션 4.1.2에 지정된 대로 네트워크를 통해 전송되는 DNS 질문에 대한 정보가 포함됩니다.
DNS_WIRE_RECORD DNS_WIRE_RECORD 구조에는 RFC 1035의 섹션 4.1.3에 지정된 대로 네트워크를 통해 전송되는 DNS 와이어 레코드에 대한 정보가 포함됩니다.
DNS_WKS_DATA DNS_WKS_DATA 구조는 RFC 1035의 섹션 3.4.2에 지정된 대로 DNS WKS(잘 알려진 서비스) 레코드를 나타냅니다.
IP4_ARRAY IP4_ARRAY 구조체는 IPv4 주소 배열을 저장합니다.
IP6_ADDRESS IP6_ADDRESS 구조체는 IPv6 주소를 저장합니다. (IP6_ADDRESS)
MDNS_QUERY_HANDLE 진행 중인 MDNS 쿼리와 관련된 정보를 포함합니다. 애플리케이션에서 해당 콘텐츠를 수정해서는 안 됩니다.
MDNS_QUERY_REQUEST mDNS 쿼리를 수행하는 데 필요한 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

드라이버 개발 도구 참조

아티클2023. 08. 22.

드라이버 개발 도구 참조 기술 개요입니다.

드라이버 개발 도구 참조 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 드라이버 개발 도구 참조

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

DXCore

아티클2024. 11. 08.

DXCore 기술 개요입니다.

DXCore를 사용하여 개발하려면 다음 헤더가 필요합니다.

- dxcore.h
- dxcore_interface.h

이 기술에 대한 프로그래밍 지침 및 코드 예제는 다음을 참조하세요.

- DXCore

열거형

테이블 확장

DXCoreAdapterPreference 목록 정렬 조건으로 사용할 DXCore 어댑터 기본 설정을 지정하는 상수를 정의합니다.
DXCoreAdapterProperty DXCore 어댑터 속성을 지정하는 상수를 정의합니다.
DXCoreAdapterState DXCore 어댑터 상태의 종류를 지정하는 상수를 정의합니다.
DXCoreHardwareTypeFilterFlags 어댑터를 열거할 때 지정된 하드웨어 유형을 지정하는 상수를 정의합니다.
DXCoreNotificationType IDXCoreAdapter 또는 IDXCoreAdapterList 개체에서 발생하는 알림 유형을 지정하는 상수를 정의합니다.
DXCoreSegmentGroup 어댑터의 메모리 세그먼트 그룹화를 지정하는 상수를 정의합니다.

함수

테이블 확장

DXCoreCreateAdapterFactory 추가 DXCore 개체를 생성하는 데 사용할 수 있는 DXCore 어댑터 팩터리를 만듭니다.
PFN_DXCORE_NOTIFICATION_CALLBACK 알림 이벤트에 대해 DXCore 개체에서 호출하는 콜백 함수(애플리케이션에서 구현)입니다.

구조

테이블 확장

DXCoreAdapterMemoryBudget 어댑터의 메모리 예산을 설명합니다.
DXCoreAdapterMemoryBudgetNodeSegmentGroup 어댑터의 메모리 세그먼트 그룹에 대해 설명합니다.
DXCoreHardwareID 어댑터의 PnP 하드웨어 ID 부분을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

DXGI

아티클2023. 03. 16.

DXGI 기술 개요입니다.

DXGI를 개발하려면 다음 헤더가 필요합니다.

- dxgi.h
- dxgi1_2.h
- dxgi1_3.h
- dxgi1_4.h
- dxgi1_5.h
- dxgi1_6.h
- dxgicommon.h
- dxgidebug.h
- dxgiformat.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- DXGI

열거형

DXGI_ADAPTER_FLAG DXGI 어댑터의 유형을 식별합니다. (DXGI_ADAPTER_FLAG)
DXGI_ADAPTER_FLAG3 DXGI 어댑터의 유형을 식별합니다. (DXGI_ADAPTER_FLAG3)
DXGI_ALPHA_MODE 표면의 알파 값, 투명도 동작을 식별합니다.
DXGI_COLOR_SPACE_TYPE 색 공간 유형을 지정합니다.
DXGI_COMPUTE_PREEMPTION_GRANULARITY 현재 컴퓨팅 작업을 수행할 때 GPU(그래픽 처리 장치)를 선점할 수 있는 세분성을 식별합니다.

DXGI_DEBUG_RLO_FLAGS ReportLiveObjects와 함께 개체의 수명에 대해 보고할 정보의 양을 지정하는 데 사용되는 플래그입니다.
DXGI_FEATURE 기능 지원을 확인할 때 사용할 다양한 하드웨어 기능을 지정합니다.
DXGI_FORMAT 완전히 형식화되고 형식이 없는 형식을 포함한 리소스 데이터 형식입니다. 페이지 아래쪽에 있는 한정자 목록은 각 형식 형식에 대해 자세히 설명합니다.
DXGI_FRAME_PRESENTATION_MODE 스왑 체인에 프레임을 표시하는 옵션을 나타냅니다.
DXGI_GPU_PREFERENCE 실행할 앱에 대한 GPU의 기본 설정입니다.
DXGI_GRAPHICS_PREEMPTION_GRANULARITY 현재 그래픽 렌더링 작업을 수행할 때 GPU(그래픽 처리 장치)를 선점할 수 있는 세분성을 식별합니다.
DXGI_HARDWARE_COMPOSITION_SUPPORT_FLAGS 지원되는 하드웨어 구성 수준을 설명합니다.
DXGI_HDR_METADATA_TYPE 헤더 메타데이터 유형을 지정합니다.
DXGI_INFO_QUEUE_MESSAGE_CATEGORY 디버그 메시지의 범주를 지정하는 값입니다.
DXGI_INFO_QUEUE_MESSAGE_SEVERITY 정보 큐에 대한 디버그 메시지 심각도 수준을 지정하는 값입니다.
DXGI_MEMORY_SEGMENT_GROUP 사용할 메모리 세그먼트 그룹을 지정합니다.

DXGI_MULTIPLANE_OVERLAY_YCbCr_FLAGS 스왑 체인 색 공간에 대한 옵션입니다.
DXGI_OFFER_RESOURCE_FLAGS OfferResources1 메서드에 대한 플래그를 지정합니다.
DXGI_OFFER_RESOURCE_PRIORITY IDXGIDevice2::OfferResources 메서드를 호출하여 리소스를 제공할 때 리소스 콘텐츠의 중요도를 식별합니다.
DXGI_OUTDUPL_POINTER_SHAPE_TYPE 포인터 셰이프의 형식을 식별합니다.
DXGI_OVERLAY_COLOR_SPACE_SUPPORT_FLAG 오버레이 색 공간에 대한 지원을 지정합니다.
DXGI_OVERLAY_SUPPORT_FLAG IDXGIOutput3::CheckOverlaySupport에 대한 호출에서 확인할 오버레이 지원을 지정합니다.
DXGI_RECLAIM_RESOURCE_RESULTS ReclaimResources1 메서드의 결과 플래그를 지정합니다.
DXGI_RESIDENCY 리소스의 메모리 위치를 나타내는 플래그입니다.
DXGI_SCALING 백 버퍼 크기가 대상 출력의 크기와 일치하지 않는 경우 크기 조정 동작을 식별합니다.
DXGI_SWAP_CHAIN_COLOR_SPACE_SUPPORT_FLAG 스왑 체인에 대한 색 공간 지원을 지정합니다.
DXGI_SWAP_CHAIN_FLAG 스왑 체인 동작에 대한 옵션입니다.
DXGI_SWAP_EFFECT IDXGISwapChain1::P resent1을 호출한 후 디스플레이 화면에서 픽셀을 처리하는 옵션입니다.

함수

AcquireNextFrame 애플리케이션이 다음 데스크톱 이미지를 처리할 준비가 되었음을 나타냅니다.
AcquireSync 키를 사용하여 공유 리소스에 대한 단독 렌더링 액세스를 가져옵니다.
AddApplicationMessage 사용자 정의 메시지를 메시지 큐에 추가하고 해당 메시지를 디버그 출력에 보냅니다.
AddMessage 메시지 큐에 디버그 메시지를 추가하고 해당 메시지를 디버그 출력에 보냅니다.
AddRetrievalFilterEntries 검색 필터 스택의 맨 위에 검색 필터를 추가합니다.
AddStorageFilterEntries 스토리지 필터 스택의 맨 위에 스토리지 필터를 추가합니다.
CheckColorSpaceSupport 스왑 체인의 색 공간 지원을 확인합니다.
CheckFeatureSupport 하드웨어 기능 지원을 확인하는 데 사용됩니다.
CheckHardwareCompositionSupport 애플리케이션에 하드웨어 확대가 지원됨을 알립니다.
CheckInterfaceSupport 시스템에서 그래픽 구성 요소에 대한 디바이스 인터페이스를 지원하는지 여부를 확인합니다.
CheckOverlayColorSpaceSupport 오버레이 색 공간 지원을 확인합니다.

CheckOverlaySupport 오버레이 지원을 확인합니다.
CheckPresentDurationSupport 그래픽 드라이버에서 사용자 지정 새로 고침 속도에 해당하는 지원되는 프레임 현재 기간을 쿼리합니다.
ClearRetrievalFilter 검색 필터 스택의 맨 위에서 검색 필터를 제거합니다.
ClearStorageFilter 스토리지 필터 스택의 맨 위에서 스토리지 필터를 제거합니다.
ClearStoredMessages 메시지 큐에서 모든 메시지를 지웁니다.
CreateDecodeSwapChainForCompositionSurfaceHandle 기존 DirectComposition 표면 핸들에 대한 YUV 스왑 체인을 만듭니다. (IDXGIFactoryMedia.CreateDecodeSwapChainForCompositionSurfaceHandle)
CreateDXGIFactory 다른 DXGI 개체를 생성하는 데 사용할 수 있는 DXGI 1.0 팩터리를 만듭니다.
CreateDXGIFactory1 다른 DXGI 개체를 생성하는 데 사용할 수 있는 DXGI 1.1 팩터리를 만듭니다.
CreateDXGIFactory2 다른 DXGI 개체를 생성하는 데 사용할 수 있는 DXGI 1.3 팩터리를 만듭니다.
CreateSharedHandle 공유 리소스에 대한 핸들을 만듭니다. 그런 다음 여러 Direct3D 디바이스에서 반환된 핸들을 사용할 수 있습니다.
CreateSoftwareAdapter 소프트웨어 어댑터를 나타내는 어댑터 인터페이스를 만듭니다.

CreateSubresourceSurface 하위 리소스 표면 개체를 만듭니다.
CreateSurface 표면을 반환합니다. 이 메서드는 내부적으로 사용되며 애플리케이션에서 직접 호출하면 안 됩니다.
CreateSwapChain 스왑 체인을 만듭니다.
CreateSwapChainForComposition DirectComposition API 또는 창에서 작성할 Xaml 프레임워크로 Direct3D 콘텐츠를 보내는 데 사용할 수 있는 스왑 체인을 만듭니다.
CreateSwapChainForCompositionSurfaceHandle 기존 DirectComposition 표면 핸들에 대한 YUV 스왑 체인을 만듭니다. (IDXGIFactoryMedia.CreateSwapChainForCompositionSurfaceHandle)
CreateSwapChainForCoreWindow 스왑 체인의 출력 창에 대한 CoreWindow 개체와 연결된 스왑 체인을 만듭니다.
CreateSwapChainForHwnd 스왑 체인의 출력 창에 대한 HWND 핸들과 연결된 스왑 체인을 만듭니다.
DisableLeakTrackingForThread 현재 스레드에 대한 누수 추적을 중지합니다.
DuplicateOutput 어댑터 출력을 나타내는 IDXGIOutput1 인터페이스에서 데스크톱 중복 인터페이스를 만듭니다.
DuplicateOutput1 IDXGIOutputDuplication 개체에서 반환할 수 있는 전체 화면 화면에 지원되는 형식 목록을 지정할 수 있습니다.
DXGIDeclareAdapterRemovalSupport 프로세스가 제거되는 그래픽 디바이스에 복원력이 있음을 나타낼 수 있습니다.

DXGIDisableVBlankVirtualization 프로세스에 대해 v-blank 가상화를 사용하지 않도록 설정합니다. 이 가상화는 IDXGIOutput::WaitForVBlank에서 안정적인 가상화된 현재 속도 및 v-blank 주기를 유지하기 위해 기본적으로 모든 스왑 체인에 대해 DRR(동적 새로 고침 속도) 기능에 사용됩니다. 가상화를 사용하지 않도록 설정하면 이러한 API에 변경된 새로 고침 속도가 표시됩니다.
DXGIGetDebugInterface 디버깅 인터페이스를 검색합니다.
DXGIGetDebugInterface1 Windows 스토어 앱이 Microsoft DXGI(DirectX Graphics Infrastructure)를 디버깅하는 데 사용하는 인터페이스를 검색합니다.
EnableLeakTrackingForThread 현재 스레드에 대한 누수 추적을 시작합니다.
EnqueueSetEvent 이전에 제출한 렌더링 명령이 모두 완료된 후 미해결 렌더링 명령을 플러시하고 지정된 이벤트 개체를 신호 상태로 설정합니다.
EnumAdapterByGpuPreference 지정된 GPU 기본 설정에 따라 그래픽 어댑터를 열거합니다.
EnumAdapterByLuid 지정된 LUID에 대한 IDXGIAdapter를 출력합니다.
EnumAdapters 어댑터(비디오 카드)를 열거합니다.
EnumAdapters1 출력이 있거나 없는 두 어댑터(비디오 카드)를 열거합니다.
EnumOutputs 어댑터(비디오 카드) 출력을 열거합니다.
EnumWarpAdapter WARP 렌더러를 사용하기 위해 D3D12CreateDevice에 제공할 수 있는 어댑터를 제공합니다.

FindClosestMatchingMode 요청된 표시 모드와 가장 일치하는 표시 모드를 찾습니다. (IDXGIOutput.FindClosestMatchingMode)
FindClosestMatchingMode1 요청된 표시 모드와 가장 일치하는 표시 모드를 찾습니다. (IDXGIOutput1.FindClosestMatchingMode1)
GetAdapter 지정된 디바이스의 어댑터를 반환합니다.
GetBackgroundColor 스왑 체인의 배경색을 검색합니다.
GetBreakOnCategory 메시지 범주의 중단이 켜져 있는지 여부를 결정합니다.
GetBreakOnID 메시지 식별자의 중단이 켜져 있는지 여부를 확인합니다.
GetBreakOnSeverity 메시지 심각도 수준의 중단이 켜져 있는지 여부를 결정합니다.
GetBuffer 스왑 체인의 백 버퍼 중 하나에 액세스합니다.
GetColorSpace 스왑 체인에서 사용하는 색 공간을 가져옵니다.
GetContainingOutput 대상 창의 클라이언트 영역 대부분을 포함하는 출력(디스플레이 모니터)을 가져옵니다.
GetCoreWindow 이 스왑 체인 개체의 기본 CoreWindow 개체를 검색합니다.
GetCreationFlags Microsoft DXGI(DirectX Graphics Infrastructure) 개체를 만들 때 사용된 플래그를 가져옵니다.

GetCurrentBackBufferIndex 스왑 체인의 현재 백 버퍼 인덱스 가져옵니다.
Getdc GDI(Windows Graphics Device Interface)를 사용하여 Microsoft DXGI(DirectX Graphics Infrastructure) 화면에 렌더링할 수 있는 DC(디바이스 컨텍스트)를 반환합니다.
GetDesc 어댑터(또는 비디오 카드)에 대한 DXGI 1.0 설명을 가져옵니다.
GetDesc 출력에 대한 설명을 가져옵니다.
GetDesc 표면에 대한 설명을 가져옵니다.
GetDesc 스왑 체인에 대한 설명을 가져옵니다.
GetDesc 중복된 출력에 대한 설명을 검색합니다. 이 설명은 데스크톱 이미지를 포함하는 표면의 크기를 지정합니다.
GetDesc1 어댑터(또는 비디오 카드)에 대한 DXGI 1.1 설명을 가져옵니다.
GetDesc1 스왑 체인에 대한 설명을 가져옵니다.
GetDesc1 색 특성 및 연결 형식을 포함하는 출력에 대한 확장된 설명을 가져옵니다.
GetDesc2 어댑터 또는 비디오 카드에 대한 Microsoft DXGI(DirectX Graphics Infrastructure) 1.2 설명을 가져옵니다.

GetDesc3 어댑터 또는 비디오 카드에 대한 Microsoft DXGI(DirectX Graphics Infrastructure) 1.6 설명을 가져옵니다. 이 설명에는 ACG 호환성에 대한 정보가 포함됩니다.
GetDestSize 비디오 처리 블릿 작업에 사용할 대상 표면의 크기를 가져옵니다.
GetDevice 디바이스를 검색합니다.
GetDisplayModeList 요청된 형식 및 기타 입력 옵션과 일치하는 표시 모드를 가져옵니다. (IDXGIOutput.GetDisplayModeList)
GetDisplayModeList1 요청된 형식 및 기타 입력 옵션과 일치하는 표시 모드를 가져옵니다. (IDXGIOutput1.GetDisplayModeList1)
GetDisplaySurfaceData 현재 디스플레이 화면의 복사본을 가져옵니다.
GetDisplaySurfaceData1 디스플레이 화면(전면 버퍼)을 사용자가 제공한 리소스에 복사합니다.
GetEvictionPriority 제거 우선 순위를 가져옵니다.
GetFrameDirtyRects 현재 데스크톱 프레임의 더티 사각형에 대한 정보를 가져옵니다.
GetFrameLatencyWaitableObject DXGI 어댑터에서 새 프레임 표시가 완료된 시점을 알리는 대기 가능 핸들을 반환합니다.
GetFrameMoveRects 현재 데스크톱 프레임의 이동된 사각형에 대한 정보를 가져옵니다.

GetFramePointerShape 현재 데스크톱 프레임의 새 포인터 셰이프에 대한 정보를 가져옵니다.
GetFrameStatistics 최근에 렌더링된 프레임에 대한 통계를 가져옵니다.
GetFrameStatistics 마지막 렌더링 프레임에 대한 성능 통계를 가져옵니다.
GetFrameStatisticsMedia 사용자 지정 새로 고침 빈도가 현재 시스템에서 승인되었는지 여부를 나타내는 DXGI_FRAME_STATISTICS_MEDIA 구조에 대해 시스템을 쿼리합니다.
GetFullscreenDesc 전체 화면 스왑 체인에 대한 설명을 가져옵니다.
GetFullscreenState 전체 화면 모드와 연결된 상태를 가져옵니다.
GetGammaControl 감마 컨트롤 설정을 가져옵니다.
GetGammaControlCapabilities 감마 제어 기능에 대한 설명을 가져옵니다.
GetGPUThreadPriority GPU 스레드 우선 순위를 가져옵니다.
GetHwnd 이 스왑 체인 개체의 기본 HWND를 검색합니다.
GetLastPresentCount IDXGISwapChain::P resent 또는 IDXGISwapChain1::P resent1이 호출된 횟수를 가져옵니다.
GetMatrixTransform 다음에 컴퍼지션 스왑 체인에 적용할 변환 행렬을 가져옵니다.

GetMaximumFrameLatency 시스템에서 렌더링을 위해 큐에 대기할 수 있는 프레임 수를 가져옵니다.
GetMaximumFrameLatency 스왑 체인이 렌더링을 위해 큐에 대기할 수 있는 프레임 수를 가져옵니다.
GetMessage 메시지 큐에서 메시지를 가져옵니다.
GetMessageCountLimit 메시지 큐에 추가할 수 있는 최대 메시지 수를 가져옵니다.
GetMuteDebugOutput 디버그 출력이 켜져 있는지 여부를 결정합니다.
GetNumMessagesAllowedByStorageFilter 스토리지 필터가 통과하도록 허용된 메시지 수를 가져옵니다.
GetNumMessagesDeniedByStorageFilter 스토리지 필터를 통해 통과가 거부된 메시지 수를 가져옵니다.
GetNumMessagesDiscardedByMessageCountLimit 메시지 수 제한으로 인해 삭제된 메시지 수를 가져옵니다.
GetNumStoredMessages 메시지 큐에 현재 저장된 메시지 수를 가져옵니다.
GetNumStoredMessagesAllowedByRetrievalFilters 검색 필터를 통과할 수 있는 메시지 수를 가져옵니다.
GetParent 개체의 부모를 가져옵니다.
GetPrivateData 개체의 데이터에 대한 포인터를 가져옵니다.

GetResource 하위 리소스 표면을 지원하는 부모 리소스 및 하위 리소스 인덱스 가져오기
GetRestrictToOutput 현재 작업의 내용을 제한할 수 있는 출력(디스플레이 모니터)을 가져옵니다.
GetRetrievalFilter 검색 필터 스택의 맨 위에 있는 검색 필터를 가져옵니다.
GetRetrievalFilterStackSize 검색 필터 스택의 크기(바이트)를 가져옵니다.
GetRotation 스왑 체인에 대한 백 버퍼의 회전을 가져옵니다.
GetSharedHandle 공유 리소스에 대한 핸들을 가져옵니다.
GetSharedResourceAdapterLuid 공유 리소스 개체가 만들어진 어댑터를 식별합니다.
GetSourceRect 스왑 체인에 사용되는 원본 영역을 가져옵니다.
GetSourceSize 스왑 체인에 사용되는 원본 영역을 가져옵니다.
GetStorageFilter 스토리지 필터 스택의 맨 위에 있는 스토리지 필터를 가져옵니다.
GetStorageFilterStackSize 스토리지 필터 스택의 크기(바이트)를 가져옵니다.
GetTargetRect 비디오 처리 블릿 작업의 대상 영역을 정의하는 사각형을 가져옵니다.

GetUsage 예상 리소스 사용량을 가져옵니다.
GetWindowAssociation 사용자가 전체 화면으로의 전환을 제어하는 창을 가져옵니다.
IsCurrent 어댑터를 다시 열거해야 하는 경우 애플리케이션에 알릴 수 있습니다.
IsLeakTrackingEnabledForThread 현재 스레드에 대해 누수 추적이 켜져 있는지 여부를 나타내는 값을 가져옵니다.
IsStereoEnabled 운영 체제의 스테레오스코픽 3D 표시 동작이 사용되는지 여부를 나타내는 부울 값을 검색합니다.
IsTemporaryMonoSupported 스왑 체인이 "임시 모노"를 지원하는지 여부를 확인합니다.
IsWindowedStereoEnabled 스테레오 모드를 사용할지 여부를 결정합니다.
MakeWindowAssociation DXGI에서 alt-enter 키 시퀀스에 대한 애플리케이션의 메시지 큐를 모니터링할 수 있습니다(이로 인해 애플리케이션이 창에서 전체 화면으로 전환되거나 그 반대로 전환됨).
Map 표면에 포함된 데이터에 대한 포인터를 가져와서 표면에 대한 GPU 액세스를 거부합니다.
MapDesktopSurface 데스크톱 이미지가 시스템 메모리에 이미 있는 경우 CPU에 데스크톱 이미지에 대한 효율적인 액세스를 제공합니다.
OfferResources 운영 체제에서 콘텐츠를 삭제하여 리소스의 비디오 메모리를 해제할 수 있습니다. (IDXGIDevice2.OfferResources)

OfferResources1 운영 체제에서 콘텐츠 삭제 및 메모리 커밋 해제를 포함하여 리소스의 비디오 메모리를 해제할 수 있습니다.
PopRetrievalFilter 검색 필터 스택의 맨 위에서 검색 필터를 팝합니다.
PopStorageFilter 스토리지 필터 스택의 맨 위에서 스토리지 필터를 팝합니다.
있음 렌더링된 이미지를 사용자에게 표시합니다.
Present1 디스플레이 화면에 프레임을 표시합니다.
PresentBuffer 출력 어댑터에 프레임을 표시합니다.
PushCopyOfRetrievalFilter 검색 필터 스택의 맨 위에 있는 검색 필터의 복사본을 검색 필터 스택에 푸시합니다.
PushCopyOfStorageFilter 현재 스토리지 필터 스택의 맨 위에 있는 스토리지 필터의 복사본을 스토리지 필터 스택에 푸시합니다.
PushDenyAllRetrievalFilter 거부-모든 검색 필터를 검색 필터 스택에 푸시합니다.
PushDenyAllStorageFilter 거부-모든 스토리지 필터를 스토리지 필터 스택에 푸시합니다.
PushEmptyRetrievalFilter 빈 검색 필터를 검색 필터 스택에 푸시합니다.
PushEmptyStorageFilter 빈 스토리지 필터를 스토리지 필터 스택에 푸시합니다.

PushRetrievalFilter 검색 필터를 검색 필터 스택에 푸시합니다.
PushStorageFilter 스토리지 필터를 스토리지 필터 스택에 푸시합니다.
QueryResourceResidency 리소스 배열의 상주 상태를 가져옵니다.
QueryVideoMemoryInfo 이 메서드는 현재 예산 및 프로세스 사용의 프로세스를 알려줍니다.
ReclaimResources IDXGIDevice2::OfferResources를 호출하여 이전에 제공된 리소스에 대한 액세스를 복원합니다.
ReclaimResources1 IDXGIDevice4::OfferResources1을 호출하여 이전에 제공된 리소스에 대한 액세스를 복원합니다.
RegisterAdaptersChangedEvent 어댑터 열거 상태가 변경될 때마다 변경 알림을 받도록 등록합니다.
RegisterHardwareContentProtectionTeardownStatusEvent 하드웨어 콘텐츠 보호 중단 이벤트의 알림을 수신하도록 등록합니다.
RegisterOcclusionStatusEvent 이벤트 신호를 사용하여 폐색 상태 변경 알림을 수신하도록 등록합니다.
RegisterOcclusionStatusWindow 폐색 상태 변경에 대한 알림 메시지를 수신하도록 애플리케이션 창을 등록합니다.
RegisterStereoStatusEvent 이벤트 신호를 사용하여 스테레오 상태 변경 알림을 받도록 등록합니다.
RegisterStereoStatusWindow 스테레오 상태 변경에 대한 알림 메시지를 수신하도록 애플리케이션 창을 등록합니다.

RegisterVideoMemoryBudgetChangeNotificationEvent 이 메서드는 CPU 동기화 개체와 예산 변경 이벤트 간에 상관 관계를 설정합니다.
ReleaseDC 현재 표면과 연결된 GDI DC(디바이스 컨텍스트)를 해제하고 Direct3D를 사용하여 렌더링할 수 있습니다.
ReleaseFrame 애플리케이션이 프레임 처리를 완료했음을 나타냅니다.
ReleaseOwnership 출력의 소유권을 해제합니다.
ReleaseSync 키를 사용하여 공유 리소스에 대한 단독 렌더링 액세스를 해제합니다.
ReportLiveObjects 개체 또는 개체의 수명에 대한 정보를 보고합니다.
ResizeBuffers 스왑 체인의 백 버퍼 크기, 형식 및 버퍼 수를 변경합니다. 애플리케이션 창의 크기를 조정할 때 호출해야 합니다.
ResizeBuffers1 D3D12 명령 큐를 입력 디바이스로 사용하여 스왑 체인을 만든 스왑 체인의 백 버퍼 크기, 형식 및 버퍼 수를 변경합니다. 애플리케이션 창의 크기를 조정할 때 호출해야 합니다.
ResizeTarget 출력 대상의 크기를 조정합니다.
SetBackgroundColor 스왑 체인의 배경색을 변경합니다.
SetBreakOnCategory 해당 범주의 메시지가 스토리지 필터를 통과할 때 중단되도록 메시지 범주를 설정합니다.

SetBreakOnID 해당 식별자가 있는 메시지가 스토리지 필터를 통과할 때 중단되도록 메시지 식별자를 설정합니다.
SetBreakOnSeverity 해당 심각도 수준의 메시지가 스토리지 필터를 통과할 때 중단되도록 메시지 심각도 수준을 설정합니다.
SetColorSpace 스왑 체인에서 사용하는 색 공간을 설정합니다. (IDXGIDecodeSwapChain.SetColorSpace)
SetColorSpace1 스왑 체인에서 사용하는 색 공간을 설정합니다. (IDXGISwapChain3.SetColorSpace1)
SetDestSize 비디오 처리 블릿 작업에 사용할 대상 표면의 크기를 설정합니다.
SetDisplaySurface 표시 모드를 변경합니다.
SetEvictionPriority 메모리에서 리소스를 제거하는 우선 순위를 설정합니다.
SetFullscreenState 표시 상태를 창 또는 전체 화면으로 설정합니다.
SetGammaControl 감마 컨트롤을 설정합니다.
SetGPUThreadPriority GPU 스레드 우선 순위를 설정합니다.
SetHDRMetaData 이 메서드는 HDR(High Dynamic Range) 및 WCG(Wide Color Gamut) 헤더 메타데이터를 설정합니다.
SetMatrixTransform 다음에 컴퍼지션 스왑 체인에 적용될 변환 행렬을 설정합니다.

SetMaximumFrameLatency 시스템에서 렌더링을 위해 큐에 대기할 수 있는 프레임 수를 설정합니다.
SetMaximumFrameLatency 스왑 체인이 렌더링을 위해 큐에 대기할 수 있는 프레임 수를 설정합니다.
SetMessageCountLimit 메시지 큐에 추가할 수 있는 최대 메시지 수를 설정합니다.
SetMuteDebugOutput 디버그 출력을 켜거나 끕니다.
SetPresentDuration 사용자 지정 프레젠테이션 기간(사용자 지정 새로 고침 속도)을 요청합니다.
SetPrivateData 애플리케이션 정의 데이터를 개체로 설정하고 해당 데이터를 GUID와 연결합니다.
SetPrivateDataInterface 개체의 개인 데이터에 인터페이스를 설정합니다.
SetRotation 스왑 체인에 대한 백 버퍼의 회전을 설정합니다.
SetSourceRect 비디오 처리 블릿 작업의 원본 영역을 정의하는 사각형을 설정합니다.
SetSourceSize 스왑 체인에 사용할 원본 영역을 설정합니다.
SetStereoEnabled 운영 체제의 스테레오스코픽 3D 표시 동작을 사용하거나 사용하지 않도록 설정하려면 부울 값을 설정합니다.
SetTargetRect 비디오 처리 블릿 작업의 대상 영역을 정의하는 사각형을 설정합니다.

SetVideoMemoryReservation 이 메서드는 애플리케이션에 필요한 최소 실제 메모리를 OS로 보냅니다.
SupportsOverlays 다중 평면 오버레이 지원을 위해 어댑터 출력을 쿼리합니다.
TakeOwnership 출력의 소유권을 가져옵니다.
자르기 앱 대신 IDXGIDevice3 DXGI 디바이스에 의해 할당된 그래픽 메모리를 트리밍합니다.
Unmap IDXGISurface::Map에서 검색된 표면에 대한 포인터를 무효화하고 리소스에 대한 GPU 액세스를 다시 사용하도록 설정합니다.
UnMapDesktopSurface IDXGIOutputDuplication::MapDesktopSurface를 사용하여 검색된 데스크톱 이미지에 대한 포인터를 무효화합니다.
UnregisterAdaptersChangedEvent 어댑터 열거 상태가 변경되면 알림 수신을 중지하도록 이벤트를 등록 취소합니다.
UnregisterHardwareContentProtectionTeardownStatus 하드웨어 콘텐츠 보호 중단 이벤트의 알림을 받지 못하도록 이벤트를 등록 취소합니다.
UnregisterOcclusionStatus 폐색 상태가 변경되면 알림을 받지 못하도록 창 또는 이벤트를 등록 취소합니다.
UnregisterStereoStatus 스테레오 상태가 변경되면 알림을 받지 못하도록 창 또는 이벤트를 등록 취소합니다.
UnregisterVideoMemoryBudgetChangeNotification 이 메서드는 예산 변경이 발생할 때마다 CPU 동기화 개체에 대한 알림을 중지합니다. 애플리케이션은 정보를 정기적으로 폴링하도록 다시 전환할 수 있습니다.

WaitForVBlank 다음 세로 공백이 발생할 때까지 스레드를 중지합니다.

인터페이스

IDXGIAdapter IDXGIAdapter 인터페이스는 디스플레이 하위 시스템(하나 이상의 GPU, DAC 및 비디오 메모리 포함)을 나타냅니다.
IDXGIAdapter1 IDXGIAdapter1 인터페이스는 디스플레이 하위 시스템(하나 이상의 GPU, DAC 및 비디오 메모리 포함)을 나타냅니다.
IDXGIAdapter2 IDXGIAdapter2 인터페이스는 하나 이상의 GPU, DAC 및 비디오 메모리를 포함하는 디스플레이 하위 시스템을 나타냅니다.
IDXGIAdapter3 이 인터페이스는 실제 메모리의 예산을 책정하고 예약하기 위한 몇 가지 메모리 상주 메서드를 추가합니다.
IDXGIAdapter4 이 인터페이스는 표시 하위 시스템을 나타내며 이 인터페이스 제품군을 확장하여 어댑터와 ACG(임의 코드 가드)의 호환성을 확인하는 메서드를 노출합니다.
IDXGIDebug 이 인터페이스는 디버그 설정을 제어하며 디버그 계층이 켜져 있는 경우에만 사용할 수 있습니다.
IDXGIDebug1 Microsoft DXGI(DirectX Graphics Infrastructure)에 대한 디버그 설정을 제어합니다. Windows 스토어 앱에서 IDXGIDebug1 인터페이스를 사용할 수 있습니다.
IDXGIDecodeSwapChain 데스크톱 미디어 앱에서 비디오 데이터를 디코딩하고 DirectComposition 화면에 표시하는 데 사용되는 스왑 체인을 나타냅니다.

IDXGIDevice IDXGIDevice 인터페이스는 이미지 데이터를 생성하는 DXGI 개체에 대한 파생 클래스를 구현합니다.
IDXGIDevice1 IDXGIDevice1 인터페이스는 이미지 데이터를 생성하는 DXGI 개체에 대한 파생 클래스를 구현합니다.
IDXGIDevice2 IDXGIDevice2 인터페이스는 이미지 데이터를 생성하는 DXGI 개체에 대한 파생 클래스를 구현합니다. 인터페이스는 GPU가 처리를 완료할 때까지 CPU 처리를 차단하고 운영 체제에 리소스를 제공하는 메서드를 노출합니다.
IDXGIDevice3 IDXGIDevice3 인터페이스는 이미지 데이터를 생성하는 DXGI 개체에 대한 파생 클래스를 구현합니다. 인터페이스는 DXGI 디바이스에서 그래픽 메모리 사용량을 자르는 메서드를 노출합니다.
IDXGIDevice4 이 인터페이스는 리소스를 제공하고 회수하는 업데이트된 메서드를 제공합니다.
IDXGIDeviceSubObject 디바이스에 연결된 개체에서 상속되어 포인터를 검색할 수 있습니다.
IDXGIDisplayControl IDXGIDisplayControl 인터페이스는 운영 체제의 입체 3D 표시 동작에 대한 사용자 기본 설정을 나타내고 스테레오스코픽 3D 표시 상태를 설정하여 사용하거나 사용하지 않도록 설정하는 메서드를 노출합니다.
IDXGIFactory IDXGIFactory 인터페이스는 전체 화면 전환을 처리하는 DXGI 개체를 생성하는 메서드를 구현합니다.
IDXGIFactory1 IDXGIFactory1 인터페이스는 DXGI 개체를 생성하는 메서드를 구현합니다.
IDXGIFactory2 IDXGIFactory2 인터페이스에는 IDXGISwapChain보다 더 많은 기능을 갖춘 최신 버전 스왑 체인을 만들고 스테레오스코픽 3D 기능을 모니터링하는 메서드가 포함되어 있습니다.

IDXGIFactory3 Microsoft DirectX Graphics Infrastructure(DXGI) 개체를 만들 수 있습니다. (IDXGIFactory3)
IDXGIFactory4 Microsoft DirectX Graphics Infrastructure(DXGI) 개체를 만들 수 있습니다. (IDXGIFactory4)
IDXGIFactory5 이 인터페이스를 사용하면 단일 메서드가 가변 새로 고침 빈도 표시를 지원할 수 있습니다.
IDXGIFactory6 이 인터페이스를 사용하면 지정된 GPU 기본 설정에 따라 그래픽 어댑터를 열거하는 단일 메서드를 사용할 수 있습니다.
IDXGIFactory7 이 인터페이스를 사용하면 알림을 등록하여 어댑터 열거 상태 변경을 검색할 수 있습니다.
IDXGIFactoryMedia DirectComposition 화면을 사용하여 비디오를 디코딩하고 표시하는 데스크톱 미디어 앱에 대한 스왑 체인을 만듭니다.
IDXGIInfoQueue 이 인터페이스는 디버그 정보 큐를 제어하며 디버그 계층이 켜져 있는 경우에만 사용할 수 있습니다.
IDXGIKeyedMutex 여러 디바이스에서 사용되는 공유 리소스에 대한 단독 액세스를 허용하는 키 뮤텍스를 나타냅니다.
IDXGIObject IDXGIObject 인터페이스는 모든 DXGI 개체에 대한 기본 인터페이스입니다. IDXGIObject는 호출자 정의(프라이빗 데이터)를 개체와 연결하고 부모 개체에 대한 인터페이스 검색을 지원합니다.
IDXGIOutput IDXGIOutput 인터페이스는 어댑터 출력(예: 모니터)을 나타냅니다.
IDXGIOutput1 IDXGIOutput1 인터페이스는 어댑터 출력(예: 모니터)을 나타냅니다.

IDXGIOutput2 어댑터 출력(예: 모니터)을 나타냅니다. IDXGIOutput2 인터페이스는 기본 출력 어댑터에서 다중 평면 오버레이 지원을 확인하는 메서드를 노출합니다.
IDXGIOutput3 어댑터 출력(예: 모니터)을 나타냅니다. IDXGIOutput3 인터페이스는 오버레이 지원을 확인하는 메서드를 노출합니다.
IDXGIOutput4 어댑터 출력(예: 모니터)을 나타냅니다. IDXGIOutput4 인터페이스는 오버레이 색 공간 지원을 확인하는 메서드를 노출합니다.
IDXGIOutput5 어댑터 출력(예: 모니터)을 나타냅니다. IDXGIOutput5 인터페이스는 전체 화면 화면에 지원되는 형식 목록을 지정하는 단일 메서드를 노출합니다.
IDXGIOutput6 어댑터 출력(예: 모니터)을 나타냅니다. IDXGIOutput6 인터페이스는 특정 모니터 기능을 제공하는 메서드를 노출합니다.
IDXGIOutputDuplication IDXGIOutputDuplication 인터페이스는 중복된 데스크톱 이미지에 액세스하고 조작합니다.
IDXGIResource IDXGIResource 인터페이스는 리소스 공유를 허용하고 리소스가 상주하는 메모리를 식별합니다.
IDXGIResource1 IDXGIResource1 인터페이스는 하위 리소스 표면 개체를 만들고 공유 리소스에 대한 핸들을 만들기 위한 지원을 추가하여 IDXGIResource 인터페이스를 확장합니다.
IDXGISurface IDXGISurface 인터페이스는 이미지 데이터 개체에 대한 메서드를 구현합니다.
IDXGISurface1 IDXGISurface1 인터페이스는 GDI(Windows Graphics Device Interface)를 사용하여 Microsoft DXGI(DirectX Graphics Infrastructure) 화면에 렌더링하는 지원을 추가하여 IDXGISurface를 확장합니다.

IDXGISurface2 IDXGISurface2 인터페이스는 하위 리소스 화면에 대한 지원을 추가하고 공유 리소스에 대한 핸들을 얻어 IDXGISurface1 인터페이스를 확장합니다.
IDXGISwapChain IDXGISwapChain 인터페이스는 렌더링된 데이터를 출력에 표시하기 전에 저장하기 위해 하나 이상의 표면을 구현합니다.
IDXGISwapChain1 IDXGISwapChain에서 향상된 프레젠테이션 기능을 제공합니다. 이러한 프레젠테이션 기능은 프레젠테이션을 최적화하기 위해 더티 사각형 및 스크롤 사각형을 지정하는 것으로 구성됩니다.
IDXGISwapChain2 스왑 백 버퍼 크기 조정 및 짧은 대기 시간 스왑 체인을 지원하는 메서드를 사용하여 IDXGISwapChain1을 확장합니다.
IDXGISwapChain3 스왑 체인의 현재 백 버퍼 인덱스 가져오기 및 색 공간 지원을 지원하는 메서드를 사용하여 IDXGISwapChain2를 확장합니다.
IDXGISwapChain4 이 인터페이스는 비디오 메타데이터를 설정하는 단일 메서드를 노출합니다.
IDXGISwapChainMedia 이 스왑 체인 인터페이스를 사용하면 데스크톱 미디어 애플리케이션이 특정 새로 고침 빈도에 대한 원활한 변경을 요청할 수 있습니다.

구조체

DXGI_ADAPTER_DESC DXGI 1.0을 사용하여 어댑터(또는 비디오 카드)를 설명합니다.
DXGI_ADAPTER_DESC1 DXGI 1.1을 사용하는 어댑터(또는 비디오 카드)에 대해 설명합니다.

DXGI_ADAPTER_DESC2 Microsoft DXGI(DirectX Graphics Infrastructure) 1.2를 사용하는 어댑터(또는 비디오 카드)에 대해 설명합니다.
DXGI_ADAPTER_DESC3 Microsoft DXGI(DirectX Graphics Infrastructure) 1.6을 사용하는 어댑터(또는 비디오 카드)에 대해 설명합니다.
DXGI_DECODE_SWAP_CHAIN_DESC IDXGIFactoryMedia::CreateDecodeSwapChainForCompositionSurfaceHandle과 함께 디코드 스왑 체인을 설명하는 데 사용됩니다.
DXGI_DISPLAY_COLOR_SPACE 이 구조를 사용하지 마세요. 지원되지 않으며 향후 릴리스에서 헤더에서 제거됩니다.
DXGI_FRAME_STATISTICS 프레임의 타이밍 및 프레젠테이션 통계를 설명합니다.
DXGI_FRAME_STATISTICS_MEDIA 앱의 사용자 지정 현재 기간(사용자 지정 새로 고침 속도)에 대한 시스템 승인을 확인하는 데 사용됩니다.
DXGI_HDR_METADATA_HDR10 HEVC(고효율 비디오 코딩)를 사용하여 비디오를 압축할 때 사용되는 HDR10에 대한 메타데이터를 설명합니다.
DXGI_INFO_QUEUE_FILTER 허용 및 거부할 메시지 유형의 목록을 포함하는 디버그 메시지 필터에 대해 설명합니다.
DXGI_INFO_QUEUE_FILTER_DESC 필터를 통과하도록 허용하거나 거부할 메시지 유형에 대해 설명합니다.
DXGI_INFO_QUEUE_MESSAGE 정보 큐의 디버그 메시지를 설명합니다.
DXGI_MAPPED_RECT 표면에 액세스하는 데 사용되는 매핑된 사각형에 대해 설명합니다.

DXGI_MATRIX_3X2_F 3x2 행렬을 나타냅니다. GetMatrixTransform 및 SetMatrixTransform과 함께 사용하여 SwapChainPanel 스왑 체인의 크기 조정 및 변환 변환을 나타냅니다.
DXGI_MODE_DESC1 디스플레이 모드 및 디스플레이 모드에서 스테레오를 지원하는지 여부를 설명합니다.
DXGI_OUTDUPL_DESC DXGI_OUTDUPL_DESC 구조체는 출력의 차원과 데스크톱 이미지를 포함하는 표면을 설명합니다. 데스크톱 이미지의 형식은 항상 DXGI_FORMAT_B8G8R8A8_UNORM.
DXGI_OUTDUPL_FRAME_INFO DXGI_OUTDUPL_FRAME_INFO 구조는 현재 데스크톱 이미지를 설명합니다.
DXGI_OUTDUPL_MOVE_RECT DXGI_OUTDUPL_MOVE_RECT 구조체는 사각형의 움직임을 설명합니다.
DXGI_OUTDUPL_POINTER_POSITION DXGI_OUTDUPL_POINTER_POSITION 구조는 하드웨어 커서의 위치를 설명합니다.
DXGI_OUTDUPL_POINTER_SHAPE_INFO DXGI_OUTDUPL_POINTER_SHAPE_INFO 구조체는 커서 모양에 대한 정보를 설명합니다.
DXGI_OUTPUT_DESC 어댑터(비디오 카드)와 디바이스 간의 출력 또는 물리적 연결을 설명합니다.
DXGI_OUTPUT_DESC1 색 기능 및 연결 유형에 대한 추가 정보를 포함하여 어댑터(비디오 카드)와 디바이스 간의 출력 또는 물리적 연결을 설명합니다.
DXGI_PRESENT_PARAMETERS 운영 체제에서 프레젠테이션을 최적화하는 데 도움이 되는 현재에 대한 정보를 설명합니다.
DXGI_QUERY_VIDEO_MEMORY_INFO 현재 비디오 메모리 예산 매개 변수를 설명합니다.

DXGI_RATIONAL 합리적인 숫자를 나타냅니다.
DXGI_SAMPLE_DESC 리소스에 대한 다중 샘플링 매개 변수에 대해 설명합니다.
DXGI_SHARED_RESOURCE 공유 리소스에 대한 핸들을 나타냅니다.
DXGI_SURFACE_DESC 표면을 설명합니다.
DXGI_SWAP_CHAIN_DESC 스왑 체인에 대해 설명합니다. (DXGI_SWAP_CHAIN_DESC)
DXGI_SWAP_CHAIN_DESC1 스왑 체인에 대해 설명합니다. (DXGI_SWAP_CHAIN_DESC1)
DXGI_SWAP_CHAIN_FULLSCREEN_DESC 스왑 체인의 전체 화면 모드를 설명합니다.
Luid 어댑터의 로컬 식별자를 설명합니다. (LUID)

DHCP(Dynamic Host Configuration Protocol)

아티클2023. 03. 09.

DHCP(동적 호스트 구성 프로토콜) 기술에 대한 개요입니다.

DHCP(동적 호스트 구성 프로토콜)를 개발하려면 다음 헤더가 필요합니다.

- dhcpcsdk.h
- dhcpsapi.h
- dhcpssdk.h
- dhcpv6csdk.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- DHCP(Dynamic Host Configuration Protocol)

열거형

DHCP_FAILOVER_MODE DHCP_FAILOVER_MODE 열거형은 장애 조치(failover) 관계에서 DHCPv4 서버 모드 작업을 정의합니다.
DHCP_FAILOVER_SERVER DHCP_FAILOVER_SERVER 열거형은 DHCP 서버가 DHCPv4 장애 조치(failover) 관계의 주 서버인지 또는 보조 서버인지를 정의합니다.
DHCP_FILTER_LIST_TYPE DHCP 서버에서 사용할 수 있는 필터 목록의 유형을 지정합니다.
DHCP_FORCE_FLAG DHCP_FORCE_FLAG 열거형은 DHCP 서브넷 요소 삭제 작업의 힘 수준을 설명하는 플래그 집합을 정의합니다.
DHCP_OPTION_DATA_TYPE DHCP_OPTION_DATA_TYPE 열거형은 DHCP 옵션 데이터를 나타내는 형식 집합을 정의합니다.

DHCP_OPTION_SCOPE_TYPE DHCP_OPTION_SCOPE_TYPE 열거형은 가능한 DHCP 옵션 범위 집합을 정의합니다.
DHCP_OPTION_SCOPE_TYPE6 DHCP 옵션에 대해 가능한 범위 집합을 정의합니다.
DHCP_OPTION_TYPE DHCP_OPTION_TYPE 열거형은 가능한 DHCP 옵션 형식 집합을 정의합니다.
DHCP_POL_ATTR_TYPE DHCP_POL_ATTR_TYPE 열거형은 DHCP 서버 정책의 조건에 대한 특성 형식을 정의합니다.
DHCP_POL_COMPARATOR DHCP_POL_COMPARATOR 열거형은 DHCP 서버 정책을 빌드할 때 조건에 대한 비교 연산자를 정의합니다.
DHCP_POL_LOGIC_OPER DHCP_POL_LOGIC_OPER 열거형은 DHCP 서버 정책에서 식의 구성 조건 및 하위 식을 그룹화하는 방법을 정의합니다.
DHCP_POLICY_FIELDS_TO_UPDATE DHCP_POLICY_FIELDS_TO_UPDATE 열거형은 업데이트해야 하는 DHCP 서버 정책의 속성을 정의합니다.
DHCP_SCAN_FLAG DHCP_SCAN_FLAG 열거형은 데이터베이스 검색 작업 중에 동기화할 수 있는 대상 집합을 정의합니다.
DHCP_SEARCH_INFO_TYPE DHCP 클라이언트 정보 레코드를 검색하는 데 사용할 수 있는 특성 집합을 정의합니다.
DHCP_SEARCH_INFO_TYPE_V6 DHCPv6 클라이언트 정보 레코드를 검색하는 데 사용할 수 있는 특성 집합을 정의합니다.
DHCP_SUBNET_ELEMENT_TYPE DHCP_SUBNET_ELEMENT_TYPE 열거형은 가능한 서브넷 요소 형식의 집합을 정의합니다.

DHCP_SUBNET_STATE DHCP_SUBNET_STATE 열거형은 서브넷에 대해 가능한 상태 집합을 정의합니다.
DHCPV6_STATELESS_PARAM_TYPE DHCPV6_STATELESS_PARAM_TYPE 열거형은 DHCPv6 상태 비저장 클라이언트 인벤토리 구성 매개 변수 형식을 정의합니다.
FSM_STATE FSM_STATE 열거형은 DHCPv4 서버에서 가능한 장애 조치(failover) 관계 상태 집합을 정의합니다.
QuarantineStatus QuarantineStatus 열거형은 NAP 서버에서 유효성이 검사된 대로 DHCPv4 클라이언트에 대해 가능한 상태 값을 지정합니다.
StatusCode StatusCode 열거형에는 IPv6 작업에 대한 상태 코드가 포함되어 있습니다.

Functions

DhcpAddFilterV4 허용/거부 목록에 링크 계층 주소 또는 주소 패턴을 추가합니다.
DhcpAddServer DhcpAddServer 함수는 도메인 디렉터리 서비스에서 유지 관리되는 기존 DHCP 서버 목록에 새 서버를 추가하려고 합니다. 지정된 DHCP 서버가 디렉터리 서비스에 이미 있는 경우 오류가 반환됩니다.
DhcpAddSubnetElement DHCP 데이터베이스의 서브넷 항목에 서브넷의 기능 또는 측면을 설명하는 요소를 추가합니다.
DhcpAddSubnetElementV4 DHCP 데이터베이스의 서브넷 항목에 서브넷의 기능 또는 측면을 설명하는 요소를 추가합니다. 이 함수는 클라이언트 유형을 고려하는 서브넷 요소를 통합하여 DhcpAddSubnetElement를 확장합니다.

DhcpAddSubnetElementV5 DhcpAddSubnetElementV5 함수는 DHCP 데이터베이스의 서브넷 항목에 서브넷의 기능 또는 측면을 설명하는 요소를 추가합니다. Windows 2000 이하: 이 함수를 사용할 수 없습니다.
DhcpAddSubnetElementV6 DhcpAddSubnetElementV6 함수는 DHCP 데이터베이스의 서브넷 항목에 서브넷의 기능 또는 측면을 설명하는 요소를 추가합니다.
DhcpAuditLogGetParams DHCP 서버에서 감사 로그 구성 설정을 반환합니다.
DhcpAuditLogSetParams DHCP 서버에서 감사 로그 생성에 대한 매개 변수를 설정합니다.
DhcpCApiCleanup DhcpCApiCleanup 함수를 사용하면 DHCP 함수 호출을 사용하는 동안 할당된 리소스를 올바르게 정리할 수 있습니다. DhcpCApiCleanup 함수는 DhcpCApiInitialize에 대한 이전 호출이 성공적으로 실행된 경우에만 호출되어야 합니다.
DhcpCApiInitialize DhcpCApiInitialize 함수는 DHCP 사용자가 수행한 첫 번째 함수 호출이어야 합니다. 다른 모든 DHCP 함수 호출에 대한 시스템을 준비합니다. 다른 DHCP 함수는 DhcpCApiInitialize 함수가 성공적으로 실행되는 경우에만 호출되어야 합니다.
DhcpCreateClass 사용자 지정 옵션 클래스를 만듭니다.
DhcpCreateClassV6 사용자 지정 DHCPv6 옵션 클래스를 만듭니다.
DhcpCreateClientInfo DhcpCreateClientInfo 함수는 DHCP 서버에 클라이언트 정보 레코드를 만듭니다.
DhcpCreateClientInfoV4 DHCP 서버에 클라이언트 정보 레코드를 만들고, 레코드에 클라이언트 유형(DHCP 또는 BOOTP)을 포함하여 DhcpCreateClientInfo의 기능을 확장합니다.

DhcpCreateClientInfoVQ DHCP 서버 데이터베이스에 제공된 DHCP 클라이언트 임대 레코드를 만듭니다.
DhcpCreateOption 기본 옵션 수준에서 기본 사용자 및 공급업체 클래스에 대한 옵션 정의를 만듭니다.
DhcpCreateOptionV5 DHCP 옵션을 만듭니다.
DhcpCreateOptionV6 DhcpCreateOptionV6 함수는 DHCP 옵션을 만듭니다.
DhcpCreateSubnet DhcpCreateSubnet 함수는 DHCP 서버에 새 서브넷을 만듭니다.
DhcpCreateSubnetV6 DhcpCreateSubnetV6 함수는 DHCP 서버에 새 서브넷을 만듭니다.
DhcpCreateSubnetVQ DhcpCreateSubnetVQ 함수는 DHCP 서버에 새 IPv4 서브넷 및 관련 NAP 상태 정보를 만듭니다.
DhcpDeleteClass DHCP 서버에서 DHCP 클래스를 삭제합니다.
DhcpDeleteClassV6 DHCPv6 서버에서 DHCP 클래스를 삭제합니다.
DhcpDeleteClientInfo DhcpDeleteClientInfo 함수는 DHCP 서버에서 클라이언트 정보 레코드를 삭제합니다.
DhcpDeleteClientInfoV6 DHCPv6 서버 데이터베이스에서 지정된 DHCPv6 클라이언트 주소 릴리스 레코드를 삭제합니다.
DhcpDeleteFilterV4 DHCP 서버의 허용/거부 목록에서 링크 계층 주소 또는 주소 패턴을 삭제합니다.

DhcpDeleteServer DhcpDeleteServer 함수는 디렉터리 서비스에서 DHCP 서버 및 관련 개체(예: 서브넷 정보 및 IP 예약)를 삭제하려고 시도합니다.
DhcpDeleteSubnet DhcpDeleteSubnet 함수는 DHCP 서버에서 서브넷을 삭제합니다.
DhcpDeleteSubnetV6 DhcpDeleteSubnetV6 함수는 DHCP 서버에서 서브넷을 삭제합니다.
DhcpDeleteSuperScopeV4 DHCP 서버에서 대범위가 삭제됩니다.
DhcpDeRegisterParamChange DhcpDeRegisterParamChange 함수는 이전에 등록된 이벤트 알림과 연결된 리소스를 해제하고 연결된 이벤트 핸들을 닫습니다.
DhcpDsCleanup DhcpDsCleanup 함수는 DhcpDsInit에서 DHCP 서비스에 할당된 디렉터리 서비스 리소스를 해제합니다. 이 함수는 해당하는 각 DHCP 서비스 프로세스에 대해 정확히 한 번 호출되어야 하며 프로세스가 종료될 때만 호출되어야 합니다.
DhcpDsInit DhcpDsInit 함수는 새 DHCP 서버 프로세스에 대한 디렉터리 서비스 내의 메모리를 초기화합니다.
DhcpEnumClasses DHCP 서버에 대해 구성된 사용자 또는 공급업체 클래스를 열거합니다.
DhcpEnumClassesV6 DHCPv6 서버에 대해 구성된 사용자 또는 공급업체 클래스를 열거합니다.
DhcpEnumFilterV4 DHCP 서버의 허용 또는 거부 목록에서 모든 필터 레코드를 열거합니다.
DhcpEnumOptions DHCPv4 서버에 저장된 열거형 옵션 집합을 반환합니다.

DhcpEnumOptionsV5 DhcpEnumOptionsV5 함수는 지정된 사용자 또는 공급업체 클래스에 대한 DHCP 옵션의 열거 목록을 반환합니다.
DhcpEnumOptionsV6 DhcpEnumOptionsV6 함수는 지정된 클래스 및/또는 공급업체에 대한 DHCP 옵션의 열거 목록을 반환합니다.
DhcpEnumOptionValues 지정된 범위에 대해 열거된 옵션 값 목록(옵션 데이터 및 연결된 ID 번호만)을 반환합니다.
DhcpEnumOptionValuesV5 DhcpEnumOptionValuesV5 함수는 지정된 사용자 또는 공급업체 클래스 내의 특정 범위에 대해 열거된 옵션 값 목록(옵션 데이터 및 연결된 ID 번호만)을 반환합니다.
DhcpEnumOptionValuesV6 DhcpEnumOptionValuesV6 함수는 지정된 클래스 내의 특정 범위에 대한 옵션 값(옵션 데이터 및 관련 ID 번호)의 열거 목록을 반환합니다.
DhcpEnumServers DhcpEnumServers 함수는 디렉터리 서비스에 있는 DHCP 서버의 열거 목록을 반환합니다.
DhcpEnumSubnetClients DhcpEnumSubnetClients 함수는 지정된 서브넷에 제공된 IP 주소가 있는 열거된 클라이언트 목록을 반환합니다.
DhcpEnumSubnetClientsFilterStatusInfo 지정된 서브넷에서 서비스되는 모든 DHCP 클라이언트를 열거하고 각 클라이언트에 대한 링크 계층 필터 상태를 포함합니다.
DhcpEnumSubnetClientsV4 지정된 서브넷에 제공된 IP 주소가 있는 클라이언트 임대 레코드의 열거 목록을 반환합니다.
DhcpEnumSubnetClientsV5 지정된 서브넷에 제공된 IP 주소가 있는 클라이언트의 열거 목록을 반환합니다.

DhcpEnumSubnetClientsV6 DhcpEnumSubnetClientsV6 함수는 지정된 서브넷에 제공된 IP 주소가 있는 열거된 클라이언트 목록을 반환합니다.
DhcpEnumSubnetClientsVQ 지정된 IPv4 서브넷에서 서비스되는 모든 DHCP 클라이언트를 검색합니다.
DhcpEnumSubnetElements 특정 DHCP 서브넷에 대한 열거형 요소 목록을 반환합니다.
DhcpEnumSubnetElementsV4 특정 DHCP 서브넷에 대한 열거형 요소 목록을 반환합니다. 이 함수는 클라이언트 유형에 따라 IP 예약을 포함할 수 있는 DHCP_SUBNET_ELEMENT_DATA_V4 구조 목록을 반환하여 DhcpEnumSubnetElements를 확장합니다.
DhcpEnumSubnetElementsV5 DhcpEnumSubnetElementsV5 함수는 특정 DHCP 서브넷에 대해 열거된 요소 목록을 반환합니다.
DhcpEnumSubnetElementsV6 DhcpEnumSubnetElementsV6 함수는 특정 DHCP 서브넷에 대해 열거된 요소 목록을 반환합니다.
DhcpEnumSubnets DhcpEnumSubnets 함수는 DHCP 서버에 정의된 서브넷의 열거 목록을 반환합니다.
DhcpEnumSubnetsV6 DhcpEnumSubnetsV6 함수는 DHCP 서버에 정의된 서브넷의 열거 목록을 반환합니다.
DhcpGetAllOptions DHCP 서버에 정의된 모든 옵션이 포함된 배열을 반환합니다.
DhcpGetAllOptionsV6 DhcpGetAllOptionsV6 함수는 DHCP 서버에 정의된 모든 옵션이 포함된 배열을 반환합니다.
DhcpGetAllOptionValues DHCP 서버의 특정 범위에 대해 정의된 모든 옵션 값이 포함된 배열을 반환합니다.

DhcpGetAllOptionValuesV6 DhcpGetAllOptionValuesV6 함수는 DHCP 서버의 특정 범위에 대해 정의된 모든 옵션 값이 포함된 배열을 반환합니다.
DhcpGetClassInfo DhcpGetClassInfo 함수는 특정 DHCP 서버에 구성된 사용자 또는 공급업체 클래스 정보를 반환합니다.
DhcpGetClientInfo DhcpGetClientInfo 함수는 특정 DHCP 클라이언트에 대한 정보를 반환합니다.
DhcpGetClientInfoV4 특정 DHCP 클라이언트에 대한 정보를 반환합니다. 이 함수는 클라이언트 형식 정보를 포함하는 DHCP_CLIENT_INFO_V4 구조를 반환하여 DhcpGetClientInfo를 확장합니다.
DhcpGetClientInfoV6 DHCPv6 서버에서 특정 IPv6 클라이언트 예약에 대한 IPv6 주소 임대 정보를 검색합니다.
DhcpGetClientInfoVQ DHCP 서버 데이터베이스에서 DHCP 클라이언트 임대 레코드 정보를 검색합니다. (DhcpGetClientInfoVQ)
DhcpGetClientOptions 사용되거나 지원되지 않으므로 ERROR_NOT_IMPLEMENTED만 반환합니다.
DhcpGetFilterV4 DHCPv4 서버의 허용/거부 목록에 대한 사용/사용 안 함 설정을 검색합니다.
DhcpGetMibInfoV5 지정된 DHCP 서버에 대한 현재 통계를 포함하는 MIB 데이터 구조를 가져옵니다.
DhcpGetMibInfoV6 DHCP 서버의 IPv6 카운터 값을 검색합니다.
DhcpGetOptionInfo 기본 사용자 및 공급업체 클래스에 대한 특정 DHCP 옵션에 대한 정보를 반환합니다.

DhcpGetOptionInfoV5 DhcpGetOptionInfoV5 함수는 특정 DHCP 옵션에 대한 정보를 반환합니다.
DhcpGetOptionInfoV6 DhcpGetOptionInfoV6 함수는 특정 DHCP 옵션에 대한 정보를 반환합니다.
DhcpGetOptionValue DhcpGetOptionValue 함수는 특정 범위에 대한 DHCP 옵션 값(옵션 코드 및 관련 데이터)을 검색합니다.
DhcpGetOptionValueV5 DhcpGetOptionValueV5 함수는 특정 범위에 대한 DHCP 옵션 값(옵션 코드 및 관련 데이터)을 검색합니다.
DhcpGetOptionValueV6 특정 사용자 또는 공급업체 클래스에 대해 DHCPv6 서버에 정의된 특정 옵션의 옵션 값을 검색합니다.
DhcpGetServerBindingInfo DhcpGetServerBindingInfo 함수는 DHCP 서버에 설정된 엔드포인트 바인딩을 반환합니다.
DhcpGetServerBindingInfoV6 DHCPv6 서버와 관련된 IPv6 인터페이스 바인딩 정보의 배열을 검색합니다.
DhcpGetServerSpecificStrings 기본 공급업체 클래스 및 사용자 클래스의 이름을 검색합니다.
DhcpGetSubnetDelayOffer DISCOVER 메시지를 받은 후 DHCP OFFER 메시지의 지연 기간을 가져옵니다.
DhcpGetSubnetInfo DhcpGetSubnetInfo 함수는 특정 서브넷에 대한 정보를 반환합니다.
DhcpGetSubnetInfoV6 DhcpGetSubnetInfoV6 함수는 특정 서브넷에 대한 정보를 반환합니다.

DhcpGetSubnetInfoVQ DHCP 서버에 정의된 특정 IPv4 서브넷에 대한 정보를 검색합니다.
DhcpGetSuperScopeInfoV4 DHCP 서버의 대범위 정보를 반환합니다.
DhcpGetThreadOptions DhcpGetThreadOptions 함수는 DhcpSetThreadOptions에서 설정한 현재 스레드 옵션을 검색합니다.
DhcpGetVersion DhcpGetVersion 함수는 DHCP 서버의 주 버전 및 부 버전 번호를 반환합니다.
DhcpHlprAddV4PolicyCondition DHCP 서버 정책 조건을 DHCP 서버 정책에 할당, 초기화 및 추가합니다.
DhcpHlprAddV4PolicyExpr DHCP 서버 정책 식을 할당, 초기화 및 DHCP 서버 정책에 추가합니다.
DhcpHlprAddV4PolicyRange DHCP 서버 정책에 DHCP IPv4 범위를 추가합니다.
DhcpHlprCreateV4Policy DHCP 서버 정책 구조를 할당하고 초기화합니다.
DhcpHlprFreeV4Policy DHCP 서버 정책 구조 내의 모든 데이터 구조의 메모리를 해제합니다.
DhcpHlprIsV4PolicySingleUC DHCP 서버 정책이 단일 사용자 클래스를 기반으로 하는지 확인합니다.
DhcpHlprIsV4PolicyValid DHCP 서버 정책을 확인합니다.
DhcpHlprIsV4PolicyWellFormed DHCP 서버 정책 구조가 올바른 형식인지 확인합니다.

DhcpHlprModifyV4PolicyExpr DHCP 서버 정책 구조에서 DHCP 서버 정책 식을 수정합니다.
DhcpHlprResetV4PolicyExpr DHCP 서버 정책 구조에서 DHCP 서버 정책 식을 다시 설정합니다.
DhcpModifyClass 서버에 정의된 DHCP 클래스를 수정합니다.
DhcpModifyClassV6 서버에 정의된 DHCPv6 사용자 또는 공급업체 클래스를 수정합니다.
DhcpRegisterParamChange DhcpRegisterParamChange 함수를 사용하면 클라이언트가 DHCP 구성 매개 변수의 변경 내용 알림을 등록할 수 있습니다.
DhcpRemoveDNSRegistrations DhcpRemoveDNSRegistrations 함수는 클라이언트에 대한 모든 DHCP 시작 DNS 등록을 제거합니다.
DhcpRemoveOption DHCP 서버의 기본 옵션 수준에서 기본 사용자 클래스 및 공급업체 클래스에 대한 특정 옵션의 정의를 제거합니다.
DhcpRemoveOptionV5 DHCP 서버의 기본 옵션 수준에서 특정 사용자 클래스 및 공급업체 클래스에 대한 특정 옵션의 정의를 제거합니다. 이렇게 하면 특정 클래스 및 공급업체 이름을 지원하는 DhcpRemoveOption의 기능이 확장됩니다.
DhcpRemoveOptionV6 DhcpRemoveOptionV6 함수는 DHCP 서버에 정의된 옵션을 제거합니다.
DhcpRemoveOptionValue 지정된 범위에 대한 기본 사용자 클래스 및 공급업체 클래스에 대한 DHCP4 서버의 특정 옵션에 대한 옵션 값을 제거합니다.
DhcpRemoveOptionValueV5 DhcpRemoveOptionValueV5 함수는 DHCP 서버에 정의된 범위에서 옵션 값을 제거합니다.

DhcpRemoveOptionValueV6 DhcpRemoveOptionValueV6 함수는 DHCP 서버에 정의된 범위에서 옵션 값을 제거합니다.
DhcpRemoveSubnetElement DHCPv4 서버에 정의된 IPv4 서브넷에서 IPv4 서브넷 요소를 제거합니다.
DhcpRemoveSubnetElementV4 DHCPv4 서버에 정의된 IPv4 서브넷에서 IPv4 서브넷 요소를 제거합니다. 이 함수는 클라이언트 유형(DHCP 또는 BOOTP) 정보를 포함하는 서브넷의 사양을 허용하여 DhcpRemoveSubnetElement에서 제공하는 기능을 확장합니다.
DhcpRemoveSubnetElementV5 DhcpRemoveSubnetElementV5 함수는 DHCP 서버에 정의된 서브넷에서 요소를 제거합니다.
DhcpRemoveSubnetElementV6 DhcpRemoveSubnetElementV6 함수는 DHCP 서버에 정의된 서브넷에서 요소를 제거합니다.
DhcpRequestParams DhcpRequestParams 함수를 사용하면 호출자가 DHCP 서버에서 DHCP 데이터를 동기적으로 또는 동기적으로 지속적으로 가져올 수 있습니다.
DhcpRpcFreeMemory DhcpRpcFreeMemory 함수는 매개 변수로 반환된 버퍼 공간 블록을 해제합니다.
DhcpScanDatabase 메모리 내 캐시와 서버 데이터베이스 간에 동기화되지 않은 임대된 DHCPv4 클라이언트 IPv4 주소를 열거합니다.
DhcpServerBackupDatabase DHCP 서버 데이터베이스 구성, 설정 및 DHCP 클라이언트 임대 레코드를 지정된 파일 위치에 백업합니다.
DhcpServerGetConfig DHCP 서버의 특정 구성 설정을 반환합니다. (DhcpServerGetConfig)
DhcpServerGetConfigV4 DHCP 서버의 특정 구성 설정을 반환합니다. (DhcpServerGetConfigV4)

DhcpServerGetConfigV6 DHCPv6 서버에 대한 구성 정보를 검색합니다.
DhcpServerGetConfigVQ 현재 DHCP 서버 구성 설정을 검색합니다.
DhcpServerQueryAttribute DHCP 서버에서 특정 특성 정보를 반환합니다.
DhcpServerQueryAttributes DHCP 서버에 설정된 특성 배열을 반환합니다.
DhcpServerQueryDnsRegCredentials 클라이언트 동적 DNS 등록을 위해 DHCP 서버에서 사용하는 현재 DNS(도메인 이름 시스템) 자격 증명을 검색합니다.
DhcpServerRedoAuthorization DhcpServerRedoAuthorization 함수는 DHCP 서버의 권한이 부여되었는지 여부를 확인하고, 그렇지 않은 경우 임대 작업을 복원합니다.
DhcpServerRestoreDatabase 특정 백업 위치(경로)에서 클라이언트 임대 데이터베이스에 대한 설정, 구성 및 레코드를 복원합니다.
DhcpServerSetConfig 서브넷 및 클라이언트 임대 정보를 저장하는 데 사용되는 JET 데이터베이스에 대한 정보 및 지원되는 프로토콜을 포함하여 특정 설정을 사용하여 DHCPv4 서버를 구성합니다.
DhcpServerSetConfigV4 DhcpServerSetConfigV4 함수는 서브넷 및 클라이언트 임대 정보를 저장하는 데 사용되는 JET 데이터베이스에 대한 정보 및 지원되는 프로토콜을 포함하여 특정 설정으로 DHCP 서버를 구성합니다.
DhcpServerSetConfigV6 범위 또는 서버 수준에서 DHCPv6 서버 구성 데이터를 설정합니다.
DhcpServerSetConfigVQ DHCP 서버 설정을 설정하거나 업데이트합니다.

DhcpServerSetDnsRegCredentialsV5 DHCP 서버에서 DHCP 클라이언트 임대 레코드에 대한 DNS(도메인 이름 시스템) 등록을 만드는 데 사용하는 자격 증명을 설정합니다.
DhcpSetClientInfo DhcpSetClientInfo 함수는 IP 주소 임대가 DHCP 서버에서 관리되는 클라이언트에 대한 정보를 설정합니다.
DhcpSetClientInfoV4 IP 주소 임대가 DHCP 서버에서 관리되는 클라이언트에 대한 정보를 설정합니다. 이 함수는 호출자가 클라이언트 유형(DHCP 또는 BOOTP)을 지정할 수 있도록 하여 DhcpSetClientInfo에서 제공하는 기능을 확장합니다.
DhcpSetClientInfoV6 DHCPv6 서버 데이터베이스에서 예약된 DHCPv6 클라이언트 임대 레코드를 설정하거나 수정합니다.
DhcpSetClientInfoVQ DHCP 서버 레코드 데이터베이스에서 기존 DHCP 클라이언트 임대 레코드를 설정하거나 수정합니다.
DhcpSetFilterV4 DHCP 서버에서 허용 및 거부 목록을 사용하거나 사용하지 않도록 설정합니다.
DhcpSetOptionInfo 기본 옵션 수준에서 기본 사용자 클래스 및 공급업체 클래스에 대해 지정된 옵션의 옵션 정의를 수정합니다.
DhcpSetOptionInfoV5 특정 DHCP 옵션에 대한 정보를 설정합니다.
DhcpSetOptionInfoV6 DhcpSetOptionInfoV6 함수는 특정 DHCP 옵션에 대한 정보를 설정합니다.
DhcpSetOptionValue DHCP 서버의 특정 옵션 값에 대한 정보를 설정합니다.

DhcpSetOptionValues DHCP 서버에 정의된 특정 범위에 대한 옵션 코드 및 관련 데이터 값을 설정합니다.
DhcpSetOptionValuesV5 DHCP 서버에 정의된 특정 범위에 대한 옵션 코드 및 관련 데이터 값을 설정합니다. 이 함수는 호출자가 옵션에 대한 클래스 및/또는 공급업체를 지정할 수 있도록 하여 DhcpSetOptionValues에서 제공하는 기능을 확장합니다.
DhcpSetOptionValueV5 DhcpSetOptionValueV5 함수는 DHCP 서버의 특정 옵션 값에 대한 정보를 설정합니다. 이 함수는 호출자가 옵션에 대한 클래스 및/또는 공급업체를 지정할 수 있도록 하여 DhcpSetOptionValue에서 제공하는 기능을 확장합니다.
DhcpSetOptionValueV6 DhcpSetOptionValueV6 함수는 DHCP 서버의 특정 옵션 값에 대한 정보를 설정합니다.
DhcpSetServerBindingInfo DhcpSetServerBindingInfo 함수는 DHCP 서버에 대한 엔드포인트 바인딩을 설정합니다.
DhcpSetServerBindingInfoV6 DHCPv6 서버에 대한 IPv6 인터페이스 바인딩을 설정하거나 수정합니다.
DhcpSetSubnetDelayOffer 특정 DHCP 범위에 대해 DISCOVER 메시지를 받은 후 DHCP OFFER 메시지의 지연 기간을 설정합니다.
DhcpSetSubnetInfo DhcpSetSubnetInfo 함수는 DHCP 서버에 정의된 서브넷에 대한 정보를 설정합니다.
DhcpSetSubnetInfoV6 DHCPv6 서버에 정의된 IPv6 서브넷에 대한 정보를 설정하거나 업데이트합니다.
DhcpSetSubnetInfoVQ DHCP 서버에 정의된 서브넷에 대한 정보를 설정합니다.
DhcpSetSuperScopeV4 서브넷을 DHCP 서버의 대범위로 설정합니다.

DhcpSetThreadOptions DhcpSetThreadOptions 함수는 현재 실행 중인 DHCP 스레드에 대한 옵션을 설정합니다.
DhcpUndoRequestParams DhcpUndoRequestParams 함수는 이전에 DhcpRequestParams 함수 호출을 사용하여 수행한 영구 요청을 제거합니다.
DhcpV4AddPolicyRange 정책에 IP 주소 범위를 추가합니다.
DhcpV4CreateClientInfo DHCP 서버 데이터베이스에 DHCPv4 클라이언트 임대 레코드를 만듭니다.
DhcpV4CreatePolicy DHCP 서버에 새 정책을 만듭니다.
DhcpV4DeletePolicy DHCP 서버에서 기존 정책을 삭제합니다.
DhcpV4EnumPolicies DHCP 서버에 구성된 정책을 열거합니다.
DhcpV4EnumSubnetClients 지정된 IPv4 서브넷에서 서비스된 모든 DHCP 클라이언트 레코드를 열거합니다.
DhcpV4EnumSubnetReservations 특정 DHCP IPv4 서브넷에 대한 예약을 열거합니다.
DhcpV4FailoverAddScopeToRelationship 지정된 장애 조치(failover) 관계에 DHCPv4 범위를 추가합니다.
DhcpV4FailoverCreateRelationship 두 서버 간에 새 DHCPv4 장애 조치(failover) 관계를 만듭니다.
DhcpV4FailoverDeleteRelationship 두 서버 간의 DHCPv4 장애 조치(failover) 관계를 삭제합니다.

DhcpV4FailoverDeleteScopeFromRelationship 지정된 장애 조치(failover) 관계에서 DHCPv4 범위를 삭제합니다.
DhcpV4FailoverEnumRelationship 서버에 있는 모든 장애 조치(failover) 관계를 열거합니다.
DhcpV4FailoverGetAddressStatus IPv4 주소의 상태를 반환합니다.
DhcpV4FailoverGetClientInfo DHCPv4 클라이언트 임대 정보를 검색합니다.
DhcpV4FailoverGetRelationship 특정 관계 이름에 대한 관계 세부 정보를 검색합니다.
DhcpV4FailoverGetScopeRelationship 지정된 DHCPv4 범위에 구성된 장애 조치(failover) 관계를 검색합니다.
DhcpV4FailoverGetScopeStatistics 장애 조치(failover) 관계의 일부인 특정 범위의 주소 사용 통계를 검색합니다.
DhcpV4FailoverGetSystemTime DHCP 서버의 현재 시간을 반환합니다.
DhcpV4FailoverSetRelationship DHCPv4 서버 장애 조치(failover) 관계의 매개 변수를 설정하거나 수정합니다.
DhcpV4FailoverTriggerAddrAllocation 장애 조치(failover) 관계의 일부인 주 서버와 보조 서버 간에 사용 가능한 주소를 재배포합니다.
DhcpV4GetAllOptionValues 특정 범위에 대한 DHCP 옵션 값(옵션 코드 및 관련 데이터)의 배열을 검색합니다.
DhcpV4GetClientInfo DHCP 서버 데이터베이스에서 DHCP 클라이언트 임대 레코드 정보를 검색합니다. (DhcpV4GetClientInfo)

DhcpV4GetFreeIPAddress 클라이언트에 임대할 수 있는 사용 가능한 IPv4 주소 목록을 검색합니다.
DhcpV4GetOptionValue 특정 범위에 대한 DHCP 옵션 값(옵션 코드 및 관련 데이터)을 검색합니다. 이 함수는 호출자가 옵션에 대한 정책을 지정할 수 있도록 하여 DhcpGetOptionValueV5에서 제공하는 기능을 확장합니다.
DhcpV4GetPolicy DHCP 서버에서 정책을 검색합니다.
DhcpV4QueryPolicyEnforcement 서버의 정책 적용 상태 또는 DHCP 서버에서 지정된 IPv4 서브넷을 검색합니다.
DhcpV4RemoveOptionValue DHCP 서버에 정의된 범위에서 옵션 값을 제거합니다. 이 함수는 호출자가 옵션에 대한 정책을 지정할 수 있도록 하여 DhcpRemoveOptionValueV5에서 제공하는 기능을 확장합니다.
DhcpV4RemovePolicyRange 정책의 IP 주소 범위 목록에서 지정된 IP 주소 범위를 제거합니다.
DhcpV4SetOptionValue DHCP 서버의 특정 옵션 값에 대한 정보를 설정합니다. 이 함수는 호출자가 옵션에 대한 정책을 지정할 수 있도록 하여 DhcpSetOptionValueV5에서 제공하는 기능을 확장합니다.
DhcpV4SetOptionValues DHCP 서버에 정의된 특정 범위에 대한 옵션 코드 및 관련 데이터 값을 설정합니다. 이 함수는 호출자가 옵션에 대한 정책을 지정할 수 있도록 하여 DhcpSetOptionValuesV5에서 제공하는 기능을 확장합니다.
DhcpV4SetPolicy 기존 정책의 매개 변수를 하나 이상 업데이트.
DhcpV4SetPolicyEnforcement DHCP 서버에서 서버 또는 지정된 IPv4 서브넷의 정책 적용 상태를 설정합니다.

Dhcpv6CApiCleanup Dhcpv6CApiCleanup 함수를 사용하면 DHCPv6에서 DHCPv6 함수 호출을 사용하는 동안 할당된 리소스를 제대로 정리할 수 있습니다. Dhcpv6CApiCleanup 함수는 Dhcpv6CApiInitialize에 대한 이전 호출이 성공적으로 실행된 경우에만 호출되어야 합니다.
Dhcpv6CApiInitialize Dhcpv6CApiInitialize 함수는 DHCPv6 사용자가 수행한 첫 번째 함수 호출이어야 합니다.
DhcpV6CreateClientInfo DHCP 서버 데이터베이스에 DHCPv6 클라이언트 임대 레코드를 만듭니다.
DhcpV6GetFreeIPAddress 클라이언트에 임대할 수 있는 사용 가능한 IPv6 주소 목록을 검색합니다.
DhcpV6GetStatelessStatistics 상태 비저장 서버 IPv6 서브넷 통계를 검색합니다.
DhcpV6GetStatelessStoreParams 서버 또는 범위 수준에서 현재 DHCPv6 상태 비스테이션 클라이언트 인벤토리 구성 설정을 검색합니다.
Dhcpv6ReleasePrefix 접두사 릴리스
Dhcpv6RenewPrefix 접두사 갱신
Dhcpv6RequestParams DHCPv6 클라이언트 캐시 또는 DHCPv6 서버에서 직접 옵션을 요청합니다.
Dhcpv6RequestPrefix 특정 접두사 요청
DhcpV6SetStatelessStoreParams 서버 또는 범위 수준에서 DHCPv6 상태 비스테이션 클라이언트 인벤토리 구성 설정을 지정합니다.

LPDHCP_CONTROL DhcpControlHook 함수는 DHCP 서버 서비스가 시작, 중지, 일시 중지 또는 계속될 때 Microsoft DHCP 서버에서 호출됩니다.
LPDHCP_DELETE_CLIENT DhcpDeleteClientHook 함수는 클라이언트 임대가 활성 임대 데이터베이스에서 삭제되기 직전에 Microsoft DHCP 서버에 의해 호출됩니다.
LPDHCP_DROP_SEND LPDHCP_DROP_SEND 콜백 함수
LPDHCP_ENTRY_POINT_FUNC 타사 DLL을 초기화하고 타사 DLL에서 알림을 원하는 이벤트를 검색하기 위해 Microsoft DHCP 서버에서 DhcpServerCalloutEntry 함수를 호출합니다. DhcpServerCalloutEntry 함수는 타사 DLL에 의해 구현됩니다.
LPDHCP_GIVE_ADDRESS DhcpAddressOfferHook 함수는 Microsoft DHCP 서버가 DHCP REQUEST 메시지에 ACK(승인)를 보내기 직전에 Microsoft DHCP 서버에서 호출됩니다.
LPDHCP_HANDLE_OPTIONS DhcpHandleOptionsHook 함수를 사용하면 타사 DLL이 DHCP 패킷에서 일반적으로 사용되는 옵션을 가져올 수 있으므로 전체 DHCP 패킷을 처리할 필요가 없습니다. DhcpHandleOptionsHook 함수는 차단하지 않아야 합니다.
LPDHCP_NEWPKT DhcpNewPktHook 함수는 처리를 위해 예정된 DHCP 패킷을 받은 직후 Microsoft DHCP 서버에서 호출됩니다.
LPDHCP_PROB DhcpAddressDelHook 함수는 다음 네 개의 정의된 이벤트 중 하나가 발생할 때 Microsoft DHCP 서버에서 호출됩니다.

구조체

DATE_TIME DATE_TIME 구조체는 날짜/시간을 포함하는 64비트 정수 값을 정의하며, 그레고리오력의 1월 1일 자정 12:00 이후 틱 수(100나노초 증분)로 표현됩니다.
DHCP_ADDR_PATTERN 링크 계층 주소/패턴에 대한 정보를 포함합니다.
DHCP_ALL_OPTION_VALUES 클래스/공급업체 페어링에 따라 구성된 DHCP 서버에 정의된 모든 옵션 값 집합을 정의합니다.
DHCP_ALL_OPTION_VALUES_PB DHCP_ALL_OPTION_VALUES_PB 구조체는 범위 내의 DHCP 서버에 대한 모든 옵션 값 집합을 정의합니다.
DHCP_ALL_OPTIONS DHCP 서버에서 사용할 수 있는 모든 옵션 집합을 정의합니다.
DHCP_ATTRIB DHCP 서버에서 설정된 특성을 정의합니다.
DHCP_ATTRIB_ARRAY DHCP 서버 특성 집합을 정의합니다.
DHCP_BINARY_DATA DHCP_BINARY_DATA 구조체는 이진 데이터의 불투명 Blob을 정의합니다.
DHCP_BIND_ELEMENT DHCP 서버에 대한 개별 네트워크 바인딩을 정의합니다. 단일 DHCP 서버는 여러 바인딩을 포함하고 여러 네트워크를 제공할 수 있습니다.
DHCP_BIND_ELEMENT_ARRAY DHCP_BIND_ELEMENT_ARRAY 구조는 DHCP 서버에서 사용하는 네트워크 바인딩 요소의 배열을 정의합니다.
DHCP_BOOTP_IP_RANGE DHCP_BOOTP_IP_RANGE 구조는 BOOTP 관련 클라이언트에 임대할 IP 제품군을 정의합니다.

DHCP_CALLOUT_TABLE DHCP_CALLOUT_TABLE 구조는 Microsoft DHCP 서버 및 타사 DLL에서 DHCP 서버 이벤트에 대한 알림 요청을 보내는 데 사용됩니다.
DHCP_CLASS_INFO DHCP 옵션 클래스를 정의합니다.
DHCP_CLASS_INFO_ARRAY DHCP 클래스 정보를 포함하는 요소의 배열을 정의합니다.
DHCP_CLASS_INFO_ARRAY_V6 DHCP_CLASS_INFO_ARRAY_V6 구조에는 사용자 클래스 또는 공급업체 클래스에 대한 정보 목록이 포함되어 있습니다.
DHCP_CLASS_INFO_V6 특정 DHCPv6 사용자 클래스 또는 공급업체 클래스에 대한 정보를 포함합니다.
DHCP_CLIENT_FILTER_STATUS_INFO 필터 상태 정보를 포함하여 DHCPv4 클라이언트에 대한 정보를 정의합니다.
DHCP_CLIENT_FILTER_STATUS_INFO_ARRAY DHCPv4 클라이언트에 대한 정보 요소의 배열을 포함합니다.
DHCP_CLIENT_INFO DHCP_CLIENT_INFO 구조는 DHCP 서버에서 사용하는 클라이언트 정보 레코드를 정의합니다.
DHCP_CLIENT_INFO_ARRAY DHCP_CLIENT_INFO_ARRAY 구조체는 열거형 함수에 사용할 DHCP_CLIENT_INFO 구조의 배열을 정의합니다.
DHCP_CLIENT_INFO_ARRAY_V4 열거형 함수에 사용할 DHCP_CLIENT_INFO_V4 구조의 배열을 정의합니다.
DHCP_CLIENT_INFO_ARRAY_V5 열거형 함수에 사용할 DHCP_CLIENT_INFO_V5 구조체의 배열을 정의합니다.

DHCP_CLIENT_INFO_ARRAY_V6 DHCPv6 클라이언트 열거형 함수에 사용할 DHCP_CLIENT_INFO_V6 구조의 배열을 정의합니다.
DHCP_CLIENT_INFO_ARRAY_VQ DHCP_CLIENT_INFO_VQ 구조체의 배열을 지정합니다.
DHCP_CLIENT_INFO_PB DHCP_CLIENT_INFO_PB 구조는 필터 상태 정보 및 IPv4 주소 할당을 초래한 정책을 포함하여 DHCPv4 클라이언트에 대한 정보를 정의합니다.
DHCP_CLIENT_INFO_PB_ARRAY DHCP_CLIENT_INFO_PB_ARRAY 구조는 DHCPv4 클라이언트 정보 요소의 배열을 정의합니다.
DHCP_CLIENT_INFO_V4 DHCP 서버에서 사용하는 클라이언트 정보 레코드를 정의하여 클라이언트 유형 정보를 포함하여 DHCP_CLIENT_INFO 제공된 정의를 확장합니다.
DHCP_CLIENT_INFO_V5 DHCP 서버에서 사용하는 클라이언트 정보 레코드를 정의하여 클라이언트 유형 및 주소 상태 정보를 포함하여 DHCP_CLIENT_INFO 제공된 정의를 확장합니다.
DHCP_CLIENT_INFO_V6 DHCP_CLIENT_INFO_V6 구조에는 DHCPv6 클라이언트에 대한 정보가 포함됩니다.
DHCP_CLIENT_INFO_VQ DHCPv4 클라이언트에 대한 정보를 정의합니다.
DHCP_FAILOVER_RELATIONSHIP DHCP_FAILOVER_RELATIONSHIP 구조는 DHCPv4 서버 장애 조치(failover) 관계에 대한 정보를 정의합니다.
DHCP_FAILOVER_RELATIONSHIP_ARRAY DHCP_FAILOVER_RELATIONSHIP_ARRAY 구조는 파트너 서버 간의 DHCPv4 장애 조치(failover) 관계 배열을 정의합니다.
DHCP_FAILOVER_STATISTICS DHCP_FAILOVER_STATISTICS 구조는 장애 조치(failover) 관계의 일부인 DHCP 서버 범위 통계를 정의합니다.

DHCP_FILTER_ADD_INFO 허용 및 거부 필터 목록에 추가할 링크 계층 필터에 대한 정보를 포함합니다.
DHCP_FILTER_ENUM_INFO 링크 계층 필터 레코드 수에 대한 정보를 포함합니다.
DHCP_FILTER_GLOBAL_INFO 허용 및 거부 필터 목록의 사용 및 비활성화에 대한 정보를 포함합니다.
DHCP_FILTER_RECORD 특정 링크 계층 필터에 대한 정보를 포함합니다.
DHCP_HOST_INFO DHCP_HOST_INFO 구조는 DHCP 서버(호스트)에 대한 정보를 정의합니다.
DHCP_HOST_INFO_V6 IPv6 주소 및 이름과 같은 DHCPv6 서버(호스트)에 대한 네트워크 정보를 포함합니다.
DHCP_IP_ARRAY DHCP_IP_ARRAY 구조체는 IP 주소 배열을 정의합니다.
DHCP_IP_CLUSTER DHCP_IP_CLUSTER 구조는 네트워크 클러스터의 주소와 마스트를 정의합니다.
DHCP_IP_RANGE DHCP_IP_RANGE 구조는 IP 주소 범위를 정의합니다.
DHCP_IP_RANGE_ARRAY DHCP_IP_RANGE_ARRAY 구조체는 DHCP IPv4 범위의 배열을 정의합니다.
DHCP_IP_RANGE_V6 DHCPv6 서버에서 사용할 IPv6 주소 범위를 지정합니다.
DHCP_IP_RESERVATION DHCP_IP_RESERVATION 구조는 클라이언트 IP 예약을 정의합니다.

DHCP_IP_RESERVATION_INFO DHCP_IP_RESERVATION_INFO 구조는 DHCPv4 클라이언트에 대한 IPv4 예약을 정의합니다.
DHCP_IP_RESERVATION_V4 DHCP_IP_RESERVATION_V4 구조는 클라이언트 IP 예약을 정의합니다. 이 구조는 예약을 보유하는 클라이언트 유형(DHCP 또는 BOOTP)을 포함하여 IP 예약을 확장합니다.
DHCP_IP_RESERVATION_V6 특정 IPv6 접두사에서 DHCPv6 클라이언트에 대한 IPv6 예약을 정의합니다.
DHCP_IPV6_ADDRESS DHCP_IPV6_ADDRESS 구조체에는 IPv6 주소가 포함되어 있습니다.
DHCP_MIB_INFO 현재 DHCP 서비스에 대한 DHCP 관련 SNMP 관리 정보 블록(MIB)에서 반환된 정보를 정의합니다.
DHCP_MIB_INFO_V5 DHCP 서버에 대한 통계 정보를 포함합니다.
DHCP_MIB_INFO_V6 DHCPv6 서버에 대한 통계를 포함합니다.
DHCP_OPTION DHCP_OPTION 구조체는 단일 DHCP 옵션 및 연결된 모든 데이터 요소를 정의합니다.
DHCP_OPTION_ARRAY DHCP_OPTION_ARRAY 구조는 DHCP 서버 옵션의 배열을 정의합니다.
DHCP_OPTION_DATA DHCP_OPTION_DATA 구조는 DHCP 옵션과 연결된 하나 이상의 데이터 요소에 대한 데이터 컨테이너를 정의합니다.
DHCP_OPTION_DATA_ELEMENT DHCP_OPTION_DATA_ELEMENT 구조체는 DHCP_OPTION_DATA 구조체 내에 있는 데이터 요소(배열의 멤버 또는 배열의 멤버)를 정의합니다.

DHCP_OPTION_LIST DHCP 옵션 값 목록을 정의합니다(연결된 ID 태그가 있는 옵션 데이터만).
DHCP_OPTION_SCOPE_INFO DHCP_OPTION_SCOPE_INFO 구조체는 특정 DHCP 범위에 제공된 옵션에 대한 정보를 정의합니다.
DHCP_OPTION_SCOPE_INFO6 DHCP 옵션 범위와 연결된 데이터를 정의합니다.
DHCP_OPTION_VALUE DHCP_OPTION_VALUE 구조체는 DHCP 옵션 값(연결된 ID 태그가 있는 옵션 데이터만)을 정의합니다.
DHCP_OPTION_VALUE_ARRAY DHCP_OPTION_VALUE_ARRAY 구조는 DHCP 옵션 값 목록(연결된 ID 태그가 있는 옵션 데이터만)을 정의합니다.
DHCP_POL_COND DHCP_POL_COND 구조는 DHCP 서버 정책 조건을 정의합니다.
DHCP_POL_COND_ARRAY DHCP_POL_COND_ARRAY 구조는 DHCP 서버 정책 조건의 배열을 정의합니다.
DHCP_POL_EXPR DHCP_POL_EXP 구조는 DHCP 서버 정책 식을 정의합니다.
DHCP_POL_EXPR_ARRAY DHCP_POL_EXPR_ARRAY 구조는 DHCP 서버 정책 식의 배열을 정의합니다.
DHCP_POLICY DHCP_POLICY 구조는 DHCP 서버 정책을 정의합니다.
DHCP_POLICY_ARRAY DHCP_POLICY_ARRAY 구조는 DHCP 서버 정책의 배열을 정의합니다.

DHCP_RESERVATION_INFO_ARRAY DHCP_RESERVATION_INFO_ARRAY 구조는 DHCPv4 클라이언트에 대한 IPv4 예약 배열을 정의합니다.
DHCP_RESERVED_SCOPE DHCP_RESERVED_SCOPE 구조체는 예약된 DHCP 범위를 정의합니다.
DHCP_SCAN_ITEM DHCP_SCAN_ITEM 구조는 DHCPv4 서버에 저장된 비동기화된 클라이언트 임대 주소와 고정해야 하는 위치(메모리 내 캐시 또는 데이터베이스)를 정의합니다.
DHCP_SCAN_LIST 수정해야 하는 DHCPv4 서버의 모든 비동기 클라이언트 임대 IP 주소 목록을 정의합니다.
DHCP_SEARCH_INFO DHCP_SEARCH_INFO 구조는 특정 서버 작업을 검색하는 데 사용되는 DHCP 클라이언트 레코드 데이터를 정의합니다.
DHCP_SEARCH_INFO_V6 DHCPv6 서버 데이터베이스를 검색할 용어 또는 값을 포함합니다.
DHCP_SERVER_CONFIG_INFO DHCP_SERVER_CONFIG_INFO 구조는 DHCP 서버를 구성하는 데 사용되는 데이터를 정의합니다.
DHCP_SERVER_CONFIG_INFO_V4 DHCP 서버를 구성하는 데 사용되는 데이터를 정의합니다.
DHCP_SERVER_CONFIG_INFO_V6 DHCPv6 서버에 대한 설정을 포함합니다.
DHCP_SERVER_CONFIG_INFO_VQ DHCP 서버에 대한 설정을 정의합니다.
DHCP_SERVER_OPTIONS DHCP_SERVER_OPTIONS 구조는 요청된 DHCP 서버 옵션을 지정합니다.

DHCP_SERVER_SPECIFIC_STRINGS 사용자 및 공급업체 클래스 이름에 대한 기본 문자열 값을 포함합니다.
DHCP_SUBNET_ELEMENT_DATA 서브넷의 기능 또는 제한을 설명하는 요소를 정의합니다. (DHCP_SUBNET_ELEMENT_DATA)
DHCP_SUBNET_ELEMENT_DATA_V4 서브넷의 기능 또는 제한을 설명하는 요소를 정의합니다. (DHCP_SUBNET_ELEMENT_DATA_V4)
DHCP_SUBNET_ELEMENT_DATA_V5 DHCP_SUBNET_ELEMENT_DATA_V5 구조체는 서브넷의 기능 또는 제한을 설명하는 요소를 정의합니다.
DHCP_SUBNET_ELEMENT_DATA_V6 IPv6 예약, IPv6 제외 범위 및 IPv6 범위와 같은 IPv6 접두사 요소에 대한 정의를 포함합니다.
DHCP_SUBNET_ELEMENT_INFO_ARRAY 서브넷 요소 데이터의 배열을 정의합니다.
DHCP_SUBNET_ELEMENT_INFO_ARRAY_V4 서브넷 요소 데이터의 배열을 정의합니다. V4 구조의 요소 데이터에는 클라이언트 형식 정보가 포함됩니다.
DHCP_SUBNET_ELEMENT_INFO_ARRAY_V5 DHCP_SUBNET_ELEMENT_INFO_ARRAY_V5 구조체는 서브넷 요소 데이터의 배열을 정의합니다. V5 구조의 요소 데이터는 BOOTP에 따라 다릅니다.
DHCP_SUBNET_ELEMENT_INFO_ARRAY_V6 DHCP_SUBNET_ELEMENT_DATA_V6 IPv6 접두사 요소의 배열을 정의하는 데이터를 포함합니다.
DHCP_SUBNET_INFO DHCP_SUBNET_INFO 구조는 서브넷을 설명하는 정보를 정의합니다.
DHCP_SUBNET_INFO_V6 IPv6 서브넷에 대한 정보를 포함합니다.

DHCP_SUBNET_INFO_VQ 서브넷을 설명하는 정보를 정의합니다.
DHCP_SUPER_SCOPE_TABLE DHCP 서버의 대범위 정의
DHCP_SUPER_SCOPE_TABLE_ENTRY 상위 범위 테이블 내에서 서브넷 항목을 정의합니다.
DHCPAPI_PARAMS DHCPAPI_PARAMS 구조체는 DHCP 매개 변수를 요청하는 데 사용됩니다.
DHCPCAPI_CLASSID DHCPCAPI_CLASSID 구조는 클라이언트 클래스 ID를 정의합니다.
DHCPCAPI_PARAMS_ARRAY DHCPCAPI_PARAMS_ARRAY 구조체는 DHCP 매개 변수를 쿼리하는 데 사용되는 DHCPAPI_PARAMS 구조의 배열을 저장합니다.
DHCPDS_SERVER DHCPDS_SERVER 구조는 디렉터리 서비스의 컨텍스트에서 DHCP 서버에 대한 정보를 정의합니다.
DHCPDS_SERVERS DHCPDS_SERVERS 구조는 디렉터리 서비스의 컨텍스트에서 DHCP 서버 목록을 정의합니다.
DHCPV4_FAILOVER_CLIENT_INFO DHCPV4_FAILOVER_CLIENT_INFO 구조는 장애 조치(failover) 관계의 일부인 DHCP 서버 범위 통계를 정의합니다.
DHCPV4_FAILOVER_CLIENT_INFO_ARRAY DHCPV4_FAILOVER_CLIENT_INFO_ARRAY 구조는 장애 조치(failover) 관계의 일부인 DHCP 서버 범위 통계의 배열을 정의합니다.
DHCPV6_BIND_ELEMENT DHCPv6 패킷을 수신하는 DHCP 서버에 대한 IPv6 인터페이스 바인딩을 정의합니다.

DHCPV6_BIND_ELEMENT_ARRAY DHCPv6 인터페이스 바인딩을 포함하는 DHCPV6_BIND_ELEMENT 구조체의 배열을 지정합니다.
DHCPV6_IP_ARRAY DHCPV6_IP_ARRAY 구조에는 DHCP IPv6 주소 구조의 배열이 포함되어 있습니다.
DHCPV6_STATELESS_PARAMS DHCPV6_STATELESS_PARAMS 구조는 서버 및 범위 수준에서 DHCPv6 상태 비국적 클라이언트 인벤토리 구성 설정을 정의합니다.
DHCPV6_STATELESS_SCOPE_STATS DHCPV6_STATELESS_SCOPE_STATS 구조는 특정 IPv6 상태 비저장 서브넷에 대한 주소 카운터를 정의합니다. 상태 비국적 클라이언트 인벤토리에서 추가 및 제거된 상태 비 상태 비지방 IPv6 클라이언트 수가 이 구조에 저장됩니다.
DHCPV6_STATELESS_STATS DHCPV6_STATELESS_STATS 구조는 상태 비저장 IPv6 서브넷 통계의 배열을 정의합니다.
DHCPV6CAPI_CLASSID IPv6 클라이언트 클래스 ID를 정의합니다.
DHCPV6CAPI_PARAMS 요청된 매개 변수를 포함합니다.
DHCPV6CAPI_PARAMS_ARRAY 요청된 매개 변수 배열을 포함합니다.
DHCPV6Prefix DHCPV6Prefix에는 IPv6 접두사가 포함되어 있습니다.
DHCPV6PrefixLeaseInformation 접두사 임대에 대한 정보입니다.
DWORD_DWORD DWORD_DWORD 구조체는 64비트 정수 값을 정의합니다.

SCOPE_MIB_INFO 반환된 DHCP 관련 SNMP 관리 정보 블록(MIB) 데이터 내에서 사용할 수 있는 범위에 대한 정보를 정의합니다.
SCOPE_MIB_INFO_V5 특정 DHCP 범위에 대한 정보를 포함합니다.

강화된 스토리지

아티클2023. 03. 09.

고급 스토리지 기술 개요입니다.

고급 스토리지를 개발하려면 다음 헤더가 필요합니다.

- ehstorapi.h
- ehstorextensions.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 강화된 스토리지

함수

권한을 부여 ACT(주소 지정 가능 명령 대상)를 ACT_AUTHORIZATION_STATE 정의된 권한 있는 상태와 연결하고 ACT에 대한 액세스 권한을 부여하는 데 필요한 필요한 시퀀스 및 논리적 조합에 따라 각 개별 사일로의 인증을 보장합니다.
GetActions 사일로 개체에 사용할 수 있는 모든 작업의 열거형을 반환합니다.
GetACT 현재 시스템에 연결된 모든 ACT(주소 지정 가능 명령 대상)의 열거형을 반환합니다. 하나 이상의 ACT가 있는 경우 고급 스토리지 API는 1개 이상의 IEnumEnhancedStorageACT 포인터 배열을 할당합니다.
GetAuthorizationState ACT의 현재 권한 부여 상태를 반환합니다.
GetDescription IEnhancedStorageSiloAction 개체에 지정된 작업에 대한 설명 문자열을 반환합니다.
GetDeviceName IEnhancedStorageACT2::GetDeviceName 메서드는 ACT(Addressable Command Target)와 연결된 디바이스 이름을 반환합니다.

GetDevicePath 사일로 디바이스 노드의 경로를 검색합니다. 반환된 문자열은 CreateFile 또는 SetupDiOpenDeviceInterface와 같은 Windows 시스템 API에 전달하는 데 적합합니다.
GetInfo 사일로 개체와 연결된 설명 정보를 반환합니다.
GetMatchingACT 클라이언트에서 제공하는 문자열을 통해 지정된 볼륨과 연결된 ACT(주소 지정 가능 명령 대상)를 반환합니다.
GetMatchingVolume ACT(주소 지정 가능 명령 대상)와 연결된 볼륨을 반환합니다.
GetName IEnhancedStorageSiloAction 개체에 지정된 작업의 이름에 대한 문자열을 반환합니다.
GetPortableDevice 해당 Enhanced Storage 사일로 드라이버에 명령을 발급하는 데 사용되는 IPortableDevice 포인터를 가져옵니다.
GetSilos ACT(Addressable Command Target)와 연결된 모든 사일로의 열거형을 반환합니다.
GetUniqueIdentity ACT(주소 지정 가능 명령 대상)의 고유 ID를 검색합니다.
Invoke IEnhancedStorageSiloAction 개체로 지정된 작업을 수행합니다.
IsDeviceRemovable IEnhancedStorageACT2::IsDeviceRemovable 메서드는 ACT와 연결된 디바이스가 이동식인지를 나타내는 정보를 반환합니다.
SendCommand 원시 사일로 명령을 사일로 개체로 보냅니다. 이 메서드는 드라이버가 나타내지 않는 사일로와 통신하는 데 활용됩니다.

권한 없음 ACT(주소 지정 가능 명령 대상)를 ACT_AUTHORIZATION_STATE 정의한 권한 없는 상태와 연결하고 ACT에 대한 액세스를 제한하는 데 필요한 필요한 시퀀스 및 논리적 조합에 따라 각 개별 사일로의 인증을 보장합니다.

인터페이스

IEnhancedStorageACT 이 인터페이스는 정보를 얻고 1667 ACT(주소 지정 가능한 연락처 대상)에 대한 작업을 수행합니다.
IEnhancedStorageACT2 IEnhancedStorageACT2 인터페이스는 1667 ACT(주소 지정 가능 연락처 대상)에 대한 정보를 가져오는 데 사용됩니다.
IEnhancedStorageSilo IEnhancedStorageSilo 인터페이스는 IEEE 1667 사일로에 대한 액세스 지점이며 정보를 얻고 사일로 수준에서 작업을 수행하는 데 사용됩니다.
IEnhancedStorageSiloAction 이 인터페이스를 IEEE 1667 사일로와 관련된 작업에 대한 액세스 지점으로 사용합니다.
IEnumEnhancedStorageACT 이 인터페이스를 모든 IEEE 1667 ACT(주소 지정 가능한 연락처 대상)의 최상위 열거자로 사용합니다.

구조체

ACT_AUTHORIZATION_STATE ACT_AUTHORIZATION_STATE 구조에는 ACT(Addressable Command Target)의 현재 권한 부여 상태를 설명하는 데이터가 포함되어 있습니다.

ENHANCED_STORAGE_PASSWORD_SILO_INFORMATION ENHANCED_STORAGE_PASSWORD_SILO_INFORMATION 구조에는 암호 사일로의 기능 및 요구 사항을 정의하는 데이터가 포함되어 있습니다.
SILO_INFO SILO_INFO 구조에는 사일로를 식별하고 설명하는 정보가 포함되어 있습니다.

이벤트 주적

2023. 03. 11.

이벤트 추적 기술에 대한 개요입니다.

이벤트 추적을 개발하려면 다음 헤더가 필요합니다.

- evntcons.h
- evntprov.h
- evntrace.h
- relogger.h
- securitybaseapi.h
- tdh.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 이벤트 추적

열거형

테이블 확장

_TDH_IN_TYPE 추적 데이터 도우미(TDH)에 대해 지원되는 [in] 형식을 정의합니다.
_TDH_OUT_TYPE 추적 데이터 도우미(TDH)에 대해 지원되는 [out] 형식을 정의합니다.
DECODING_SOURCE 이벤트 데이터의 원본을 정의합니다.
ETW_CONTEXT_REGISTER_TYPES 컨텍스트 레지스터 추적을 사용할 때 수집할 레지스터 집합을 지정합니다.
ETW_PROCESS_HANDLE_INFO_TYPE 추적 처리 세션에서 수행할 작업을 지정합니다.
ETW_PROCESS_TRACE_MODES

지원되는 프로세스 추적 모드를 지정합니다.
ETW_PROVIDER_TRAIT_TYPE ETW(Windows용 이벤트 추적)에서 지원하는 공급자 특성 유형을 지정합니다.
EVENT_FIELD_TYPE 검색할 공급자 정보를 정의합니다.
EVENT_INFO_CLASS EVENT_INFO_CLASS 열거형 형식은 EventSetInformation 함수와 함께 ETW 이벤트 공급자 등록에서 수행할 구성 작업을 지정하는 데 사용됩니다.
EVENTSECURITYOPERATION EventAccessControl 함수가 수정하는 보안 설명자의 구성 요소를 정의합니다.
MAP_FLAGS 맵이 값 맵, 비트맵 또는 패턴 맵인지 여부를 나타내는 상수 값을 정의합니다.
MAP_VALUETYPE 값 맵 값이 ULONG 데이터 형식 또는 문자열에 있는지를 정의합니다.
PAYLOAD_OPERATOR TDH(추적 데이터 도우미)에 대해 지원되는 페이로드 연산자를 정의합니다.
PROPERTY_FLAGS 속성이 구조체 또는 배열에 포함되어 있는지 여부를 정의합니다.
TDH_CONTEXT_TYPE 컨텍스트 형식을 정의합니다.
TEMPLATE_FLAGS 이벤트 데이터의 레이아웃을 나타내는 상수 값을 정의합니다.
TRACE_QUERY_INFO_CLASS EnumerateTraceGuidsEx 및 TraceSetInformation과 함께 추적 정보 유형을 지정하는 데 사용됩니다.

Functions

테이블 확장

AddLogfileTraceStream 새 로그 파일 기반 ETW 추적 스트림을 다시 로그에 추가합니다.
AddRealtimeTraceStream 재로거에 새 실시간 ETW 추적 스트림을 추가합니다.
취소 다시 로깅 프로세스를 종료합니다.
복제 이벤트의 중복 복사본을 만듭니다.
CloseTrace CloseTrace 함수는 OpenTrace를 사용하여 만든 추적 처리 세션을 닫습니다.
ControlTraceA CONTROLTraceA(ANSI) 함수(evntrace.h)는 지정된 이벤트 추적 세션을 플러시, 쿼리, 업데이트 또는 중지합니다.
ControlTraceW ControlTraceW(유니코드) 함수(evntrace.h)는 지정된 이벤트 추적 세션을 플러시, 쿼리, 업데이트 또는 중지합니다.
CreateEventInstance 새 이벤트를 생성합니다.
CreateTraceInstanceId RegisterTraceGuids 기반("클래식") 이벤트 공급자는 CreateTraceInstanceId 함수를 사용하여 고유한 트랜잭션 식별자를 만들고 등록 핸들에 매핑합니다. 그런 다음, 공급자는 TraceEventInstance 함수를 호출할 때 트랜잭션 식별자를 사용할 수 있습니다.
CveEventWrite 사용자 모드 애플리케이션에서 보안 취약성 악용 시도가 감지될 때 이벤트를 게시하기 위한 추적 함수입니다.
DECLSPEC_XFGVIRT

EMI_MAP_FORMAT 이벤트 맵 형식을 검색하는 매크로입니다.
EMI_MAP_INPUT 이벤트 맵 입력을 검색하는 매크로입니다.
EMI_MAP_NAME 이벤트 맵 이름을 검색하는 매크로입니다.
EMI_MAP_OUTPUT 이벤트 맵 출력을 검색하는 매크로입니다.
EnableTrace 추적 세션 컨트롤러는 EnableTrace를 호출하여 ETW 이벤트 공급자가 추적 세션에 이벤트를 기록하는 방법을 구성합니다. EnableTraceEx2 함수는 이 함수를 대체합니다.
EnableTraceEx 추적 세션 컨트롤러는 EnableTraceEx를 호출하여 ETW 이벤트 공급자가 추적 세션에 이벤트를 기록하는 방법을 구성합니다. EnableTraceEx2 함수는 이 함수를 대체합니다.
EnableTraceEx2 추적 세션 컨트롤러는 EnableTraceEx2를 호출하여 ETW 이벤트 공급자가 추적 세션에 이벤트를 기록하는 방법을 구성합니다.
EnumerateTraceGuids 현재 컴퓨터에서 실행 중인 이벤트 추적 공급자에 대한 정보를 검색합니다. EnumerateTraceGuidsEx 함수는 이 함수를 대체합니다.
EnumerateTraceGuidsEx 현재 컴퓨터에서 실행 중인 이벤트 추적 공급자에 대한 정보를 검색합니다.
EtwGetTraitFromProviderTraits
EventAccessControl 지정된 공급자 또는 세션의 권한을 추가하거나 수정합니다.
EventAccessQuery

지정된 컨트롤러 또는 공급자에 대한 권한을 검색합니다.
EventAccessRemove 지정된 공급자 또는 세션에 대한 레지스트리에 정의된 권한을 제거합니다.
EventActivityIdControl ETW 이벤트에서 사용할 활동 식별자를 만들고, 쿼리하고, 설정합니다.
EventDataDescCreate EVENT_DATA_DESCRIPTOR 값을 설정합니다.
EventDescCreate 이벤트 설명자의 값을 설정합니다.
EventDescGetChannel 이벤트 설명자에서 채널을 검색합니다.
EventDescGetId 이벤트 설명자에서 이벤트 식별자를 검색합니다.
EventDescGetKeyword 이벤트 설명자에서 키워드를 검색합니다.
EventDescGetLevel 이벤트 설명자에서 심각도 수준을 검색합니다.
EventDescGetOpcode 이벤트 설명자에서 작업 코드를 검색합니다.
EventDescGetTask 이벤트 설명자에서 작업을 검색합니다.
EventDescGetVersion 이벤트 설명자에서 버전을 검색합니다.
EventDescOrKeyword 이벤트 설명자에 다른 키워드를 추가합니다.

EventDescSetChannel 이벤트 설명자의 채널 멤버를 설정합니다.
EventDescSetId 이벤트 설명자의 ID 멤버를 설정합니다.
EventDescSetKeyword 이벤트 설명자의 Keyword 멤버를 설정합니다.
EventDescSetLevel 이벤트 설명자의 Level 멤버를 설정합니다.
EventDescSetOpcode 이벤트 설명자의 Opcode 멤버를 설정합니다.
EventDescSetTask 이벤트 설명자의 Task 멤버를 설정합니다.
EventDescSetVersion 이벤트 설명자의 Version 멤버를 설정합니다.
EventDescZero 이벤트 설명자를 0으로 초기화합니다.
EventEnabled 이벤트 공급자가 이벤트의 EVENT_DESCRIPTOR 따라 특정 이벤트를 생성해야 하는지 여부를 결정합니다.
EventProviderEnabled 이벤트 공급자가 이벤트의 수준 및 키워드를 기반으로 특정 이벤트를 생성해야 하는지 여부를 결정합니다.
EventRegister ETW 이벤트 공급자를 등록하여 ETW 이벤트를 작성하는 데 사용할 수 있는 핸들을 만듭니다.
EventSetInformation ETW 이벤트 공급자를 구성합니다.
EventUnregister

ETW 이벤트 공급자의 등록을 취소합니다.
EventWrite 현재 스레드의 작업 ID를 사용하는 ETW 이벤트를 씁니다.
EventWriteEx 활동 ID, 선택적 관련 활동 ID, 세션 필터 및 특수 옵션을 사용하여 ETW 이벤트를 작성합니다.
EventWriteString 문자열을 데이터로 포함하는 ETW 이벤트를 씁니다. 이 함수를 사용하면 안 됩니다.
EventWriteTransfer 활동 ID 및 선택적 관련 활동 ID를 사용하여 ETW 이벤트를 씁니다.
FlushTraceA FLUSHTraceA(ANSI) 함수(evntrace.h)를 사용하면 이벤트 추적 세션이 지정된 세션에 대해 버퍼링된 이벤트를 즉시 전달합니다.
FlushTraceW FlushTraceW(유니코드) 함수(evntrace.h)를 사용하면 이벤트 추적 세션이 지정된 세션에 대해 버퍼링된 이벤트를 즉시 전달합니다.
GetEventProcessorIndex
GetEventRecord 이벤트를 설명하는 이벤트 레코드를 검색합니다.
GetTraceEnableFlags RegisterTraceGuids 기반("클래식") 이벤트 공급자는 GetTraceEnableFlags 함수를 사용하여 추적 컨트롤러에서 지정한 사용 플래그를 검색하여 추적할 이벤트의 범주를 나타냅니다. 공급자는 ControlCallback 함수에서 이 함수를 호출합니다.
GetTraceEnableLevel RegisterTraceGuids 기반("클래식") 이벤트 공급자는 GetTraceEnableLevel 함수를 사용하여 추적 컨트롤러에서 지정한 사용 수준을 검색하여 추적할 이벤트의 수준을 나타냅니다. 공급자는 ControlCallback 함수에서 이 함수를 호출합니다.
GetTraceLoggerHandle RegisterTraceGuids 기반("클래식") 이벤트 공급자는 GetTraceLoggerHandle 함수를 사용하여 이벤트를 작

성해야 하는 이벤트 추적 세션의 핸들을 검색합니다. 공급자는 ControlCallback 함수에서 이 함수를 호출합니다.
GetUserContext 이벤트가 속한 스트림과 연결된 사용자 컨텍스트를 검색합니다.
주입하다 시스템에서 생성되지 않은 이벤트를 출력 추적 로그 파일에 기록되는 이벤트 스트림에 삽입합니다.
OnBeginProcessTrace 다시 로깅을 시작할 수 있도록 추적이 시작되려고 했음을 나타냅니다.
Onevent 재로거와 연결된 추적 스트림에서 이벤트가 수신되었음을 나타냅니다.
OnFinalizeProcessTrace 다시 로깅을 완료할 수 있도록 추적이 끝나려고 했음을 나타냅니다.
OpenTraceA ANSI(OpenTraceA) 함수(evntrace.h)는 ETW 실시간 추적 세션 또는 ETW 로그 파일에서 이벤트를 소비하기 위한 ETW 추적 처리 핸들을 엽니다.
OpenTraceFromBufferStream 파일 또는 활성 세션에 직접 연결되지 않은 추적 처리 세션을 만듭니다.
OpenTraceFromFile Tracelog .etl 파일을 처리하는 추적 처리 세션을 만듭니다.
OpenTraceFromRealTimeLogger ETW 실시간 추적 세션 또는 ETW 로그 파일에서 이벤트를 소비하기 위한 ETW 추적 처리 핸들을 엽니다.
OpenTraceFromRealTimeLoggerWithAllocationOptions 활성 실시간 ETW 세션에 연결된 추적 처리 세션을 만듭니다.
OpenTraceW OpenTraceW(유니코드) 함수(evntrace.h)는 ETW 실시간 추적 세션 또는 ETW 로그 파일에서 이벤트를 소비하기 위한 ETW 추적 처리 핸들을 엽니다.

PEI_PROVIDER_NAME PEI(공급자 이벤트 정보) 이름을 검색하는 매크로입니다.
PENABLECALLBACK ETW 이벤트 공급자는 필요에 따라 구성 변경 알림을 수신하도록 EnableCallback 함수를 정의합니다. PENABLECALLBACK 형식은 이 콜백 함수에 대한 포인터를 정의합니다. EnableCallback은 애플리케이션 정의 함수 이름의 자리 표시자입니다.
PETW_BUFFER_CALLBACK ProcessTrace에서 호출할 BufferCallback에 대한 함수 정의입니다.
PETW_BUFFER_COMPLETION_CALLBACK ProcessTraceAddBufferToBufferStream이 버퍼로 완료될 때 발생할 콜백에 대한 함수 정의입니다. 이 콜백은 일반적으로 버퍼를 적절하게 해제해야 합니다.
PEVENT_CALLBACK ETW 이벤트 소비자는 추적 처리 세션에서 이벤트를 수신하기 위해 이 콜백을 구현합니다. EventRecordCallback 콜백은 이 콜백을 대체합니다.
PEVENT_RECORD_CALLBACK ETW 이벤트 소비자는 추적 처리 세션에서 이벤트를 수신하기 위해 이 콜백을 구현합니다. PEVENT_RECORD_CALLBACK 형식은 이 콜백 함수에 대한 포인터를 정의합니다. EventRecordCallback은 애플리케이션 정의 함수 이름의 자리 표시자입니다.
PEVENT_TRACE_BUFFER_CALLBACKA ANSI(PEVENT_TRACE_BUFFER_CALLBACKA)(evntrace.h) 함수는 추적 처리 세션 중에 ETW가 보내는 이벤트의 각 버퍼에 대한 통계를 가져옵니다.
PEVENT_TRACE_BUFFER_CALLBACKW PEVENT_TRACE_BUFFER_CALLBACKW(유니코드)(evntrace.h) 함수는 추적 처리 세션 중에 ETW가 보내는 이벤트의 각 버퍼에 대한 통계를 가져옵니다.
PFI_FIELD_MESSAGE PFI(공급자 필드 정보) 필드 메시지를 검색하는 매크로입니다.
PFI_FIELD_NAME PFI(공급자 필드 정보) 필드 이름을 검색하는 매크로입니다.

PFI_FILTER_MESSAGE PFI(공급자 필드 정보) 필드 메시지를 필터링하는 매크로입니다.
PFI_PROPERTY_NAME PFI(공급자 필드 정보) 속성 이름을 검색하는 매크로입니다.
ProcessTrace 하나 이상의 추적 처리 세션에서 소비자에게 이벤트를 전달합니다.
ProcessTrace 연결된 추적 스트림의 이벤트를 소비자에게 전달합니다.
ProcessTraceAddBufferToBufferStream OpenTraceFromBufferStream에서 만든 처리 세션에 ETW 추적 버퍼를 제공합니다.
ProcessTraceBufferDecrementReference ProcessTraceBufferIncrementReference에서 추가한 버퍼에 대한 참조를 해제합니다.
ProcessTraceBufferIncrementReference 제공된 버퍼에서 BufferCallback 중에 호출되어 호출자가 이를 수행할 때까지 해제되지 않도록 합니다.
QueryAllTracesA ANSI(QueryAllTracesA) 함수(evntrace.h) 함수는 호출자가 쿼리할 수 있는 모든 이벤트 추적 세션에 대한 속성과 통계를 검색합니다.
QueryAllTracesW QueryAllTracesW(유니코드) 함수(evntrace.h) 함수는 호출자가 쿼리할 수 있는 모든 이벤트 추적 세션에 대한 속성과 통계를 검색합니다.
QueryTraceA ANSI(QueryTraceA) 함수(evntrace.h)는 지정된 이벤트 추적 세션에 대한 속성 설정 및 세션 통계를 검색합니다.
QueryTraceProcessingHandle OpenTrace에서 연 ETW 추적 처리 세션에 대한 정보를 검색합니다.
QueryTraceW

QueryTraceW(유니코드) 함수(evntrace.h)는 지정된 이벤트 추적 세션에 대한 속성 설정 및 세션 통계를 검색합니다.
RegisterCallback 추적 작업(새 이벤트 시작, 중지 및 로깅)을 알리기 위해 IEventCallback의 구현을 relogger에 등록합니다.
RegisterTraceGuidsA ANSI(RegisterTraceGuidsA) 함수(evntrace.h)는 사용되지 않는 함수이며 새 코드는 제공된 대안을 사용해야 합니다.
RegisterTraceGuidsW RegisterTraceGuidsW(유니코드) 함수(evntrace.h)는 사용되지 않는 함수이며 새 코드는 제공된 대안을 사용해야 합니다.
RemoveTraceCallback RemoveTraceCallback 함수는 EventCallback 함수가 이벤트 추적 클래스에 대한 이벤트를 수신하는 것을 중지합니다. 이 함수는 사용되지 않습니다.
SetActivityId 현재 스레드의 활동 ID를 설정합니다.
SetCompressionMode 다시 로그된 추적에서 압축을 사용하거나 사용하지 않도록 설정합니다.
SetEventDescriptor 이벤트에 대한 이벤트 설명자를 설정합니다.
SetOutputFilename ETW가 다시 기록된 새 추적을 작성해야 하는 파일을 나타냅니다.
SetPayload 이벤트에 대한 페이로드를 설정합니다.
SetProcessId 특정 프로세스에 이벤트를 할당합니다.
SetProcessorIndex 현재 스레드에서 프로세서 인덱스 를 설정합니다.

SetProviderId 이벤트를 추적한 공급자의 GUID를 설정합니다.
SetThreadId 이벤트를 생성하는 스레드의 식별자를 설정합니다.
SetThreadTimes 현재 스레드의 스레드 시간을 설정합니다.
SetTimeStamp 이벤트가 발생한 시간을 설정합니다.
SetTraceCallback SetTraceCallback 함수는 지정된 이벤트 추적 클래스에 대한 이벤트를 처리하는 EventCallback 함수를 지정합니다. 이 함수는 사용되지 않습니다.
StartTraceA StartTrace 함수는 이벤트 추적 세션을 시작합니다. (ANSI)
StartTraceW StartTrace 함수는 이벤트 추적 세션을 시작합니다. (유니코드)
StopTraceA STOPTraceA(ANSI) 함수(evntrace.h)는 지정된 이벤트 추적 세션을 중지합니다. ControlTrace 함수는 이 함수를 대체합니다.
StopTraceW StopTraceW(유니코드) 함수(evntrace.h)는 지정된 이벤트 추적 세션을 중지합니다. ControlTrace 함수는 이 함수를 대체합니다.
TdhAggregatePayloadFilters EnableTraceEx2 함수에 사용할 단일 데이터 구조로 단일 공급자에 대한 여러 페이로드 필터를 집계합니다.
TdhCleanupPayloadEventFilterDescriptor TdhAggregatePayloadFilters 함수를 사용하여 만든 페이로드 필터의 집계된 구조를 해제합니다.
TdhCloseDecodingHandle

입력 디코딩 핸들과 연결된 모든 리소스를 해제합니다.
TdhCreatePayloadFilter EnableTraceEx2 함수와 함께 사용할 단일 페이로드에 대한 단일 필터를 만듭니다.
TdhDeletePayloadFilter TdhCreatePayloadFilter 함수에 의해 단일 페이로드 필터에 할당된 메모리를 해제합니다.
TdhEnumerateManifestProviderEvents 공급자 매니페스트에 있는 이벤트 목록을 검색합니다.
TdhEnumerateProviderFieldInformation 지정된 공급자에 대해 지정된 필드 메타데이터를 검색합니다.
TdhEnumerateProviderFilters 매니페스트에 정의된 지정된 공급자가 지정한 필터를 열거합니다.
TdhEnumerateProviders 컴퓨터에 MOF 클래스 또는 매니페스트 파일을 등록한 공급자 목록을 검색합니다.
TdhEnumerateProvidersForDecodingSource 컴퓨터에 MOF 클래스 또는 매니페스트 파일을 등록한 공급자 목록을 검색합니다.
TdhFormatProperty 표시할 속성 값의 형식을 지정합니다.
TdhGetDecodingParameter 디코딩 매개 변수의 값을 검색합니다.
TdhGetEventInformation 이벤트에 대한 메타데이터를 검색합니다.
TdhGetEventMapInformation 이벤트에 포함된 이벤트 맵에 대한 정보를 검색합니다.
TdhGetManifestEventInformation 매니페스트의 이벤트에 대한 메타데이터를 검색합니다.

TdhGetProperty 이벤트 데이터에서 속성 값을 검색합니다.
TdhGetPropertySize 이벤트 데이터에서 하나 이상의 속성 값의 크기를 검색합니다.
TdhGetWppMessage EVENT_RECORD 구조에 포함된 형식의 WPP 메시지를 검색합니다.
TdhGetWppProperty WPP 메시지와 연결된 특정 속성을 검색합니다.
TdhLoadManifest 로그 파일을 디코딩하는 데 사용되는 매니페스트를 로드합니다.
TdhLoadManifestFromBinary NULL로 끝나는 경로를 특정 이벤트 공급자를 디코딩하는 데 필요한 메타데이터 리소스가 포함된 이진 파일로 이동합니다.
TdhLoadManifestFromMemory 메모리에서 매니페스트를 로드합니다.
TdhOpenDecodingHandle 디코딩 핸들을 엽니다.
TdhQueryProviderFieldInformation 지정된 값과 일치하는 필드 값에 대한 이벤트 설명에서 지정된 필드에 대한 정보를 검색합니다.
TdhSetDecodingParameter 디코딩 매개 변수의 값을 설정합니다.
TdhUnloadManifest TdhLoadManifest 함수에 의해 로드된 매니페스트를 언로드합니다.
TdhUnloadManifestFromMemory 메모리에서 매니페스트를 언로드합니다.

TEI_ACTIVITYID_NAME TEI(추적 이벤트 정보) 활동 ID 이름을 검색하는 매크로입니다.
TEI_CHANNEL_NAME TEI(추적 이벤트 정보) 채널 이름을 검색하는 매크로입니다.
TEI_EVENT_MESSAGE TEI(추적 이벤트 정보) 메시지를 검색하는 매크로입니다.
TEI_KEYWORDS_NAME TEI(추적 이벤트 정보) 키워드 이름을 검색하는 매크로입니다.
TEI_LEVEL_NAME TEI(추적 이벤트 정보) 수준 이름을 검색하는 매크로입니다.
TEI_MAP_NAME TEI(추적 이벤트 정보) 맵 이름을 검색하는 매크로입니다.
TEI_OPCODE_NAME TEI(추적 이벤트 정보) opcode 이름을 검색하는 매크로입니다.
TEI_PROPERTY_NAME TEI(추적 이벤트 정보) 속성 이름을 검색하는 매크로입니다.
TEI_PROVIDER_MESSAGE TEI(추적 이벤트 정보) 공급자 메시지를 검색하는 매크로입니다.
TEI_PROVIDER_NAME TEI(추적 이벤트 정보) 공급자 이름을 검색하는 매크로입니다.
TEI_RELATEDACTIVITYID_NAME TEI(추적 이벤트 정보) 관련 활동 ID 이름을 검색하는 매크로입니다.
TEI_TASK_NAME TEI(추적 이벤트 정보) 작업 이름을 검색하는 매크로입니다.
TraceEvent

RegisterTraceGuids 기반("클래식") 이벤트 공급자는 TraceEvent 함수를 사용하여 구조적 이벤트를 이벤트 추적 세션으로 보냅니다.
TraceEventInstance RegisterTraceGuids 기반("클래식") 이벤트 공급자는 TraceEventInstance 함수를 사용하여 인스턴스 식별자가 있는 이벤트 추적 세션에 구조적 이벤트를 보냅니다.
TraceMessage RegisterTraceGuids 기반("클래식") 이벤트 공급자는 TraceMessage 함수를 사용하여 TMF 기반 WPP(메시지 기반) 이벤트를 이벤트 추적 세션으로 보냅니다.
TraceMessageVa RegisterTraceGuids 기반("클래식") 이벤트 공급자는 TraceMessageVa 함수를 사용하여 va_list 매개 변수를 사용하여 TMF 기반 WPP(메시지 기반) 이벤트를 이벤트 추적 세션으로 보냅니다.
TraceQueryInformation 이벤트 추적 세션에 대한 정보를 제공합니다.
TraceSetInformation 이벤트 추적 세션 설정을 구성합니다.
UnregisterTraceGuids RegisterTraceGuids를 사용하여 등록된 "클래식"(Windows 2000 스타일) ETW 이벤트 추적 공급자를 등록 취소합니다.
UpdateTraceA ANSI(UpdateTraceA) 함수(evntrace.h)는 지정된 이벤트 추적 세션의 속성 설정을 업데이트합니다.
UpdateTraceW UpdateTraceW(유니코드) 함수(evntrace.h)는 지정된 이벤트 추적 세션의 속성 설정을 업데이트합니다.
WMIDPREQUEST RegisterTraceGuids 기반("클래식") 이벤트 공급자는 컨트롤러로부터 알림을 받도록 이 함수를 구현합니다. WMIDPREQUEST 형식은 이 콜백 함수에 대한 포인터를 정의합니다. ControlCallback은 애플리케이션 정의 함수 이름의 자리 표시자입니다.

인터페이스

테이블 확장

ITraceEvent 특정 이벤트와 관련된 데이터에 대한 액세스를 제공합니다.
ITraceEventCallback ETW에서 추적 프로세스가 시작, 종료 및 로그 이벤트를 시작할 때 재로거에 정보를 제공하는 데 사용됩니다.
ITraceRelogger ETW 추적 스트림에서 이벤트를 조작하고 다시 로깅할 수 있도록 다시 로깅 기능에 대한 액세스를 제공합니다.

구조체

테이블 확장

CLASSIC_EVENT_ID 호출 스택 추적을 사용하도록 설정할 커널 이벤트를 식별합니다.
ENABLE_TRACE_PARAMETERS EnableTraceEx2를 통해 공급자를 사용하도록 설정하는 데 사용되는 정보를 포함합니다.
ENABLE_TRACE_PARAMETERS_V1 EnableTraceEx2를 통해 공급자를 사용하도록 설정하는 데 사용되는 정보를 포함합니다. 이 구조체는 사용되지 않습니다.
ETW_BUFFER_CALLBACK_INFORMATION BufferCallback에 *ConsumerInfo* 매개 변수로 제공되며 현재 처리 세션에 대한 세부 정보를 제공합니다.
ETW_BUFFER_CONTEXT 이벤트에 대한 컨텍스트 정보를 제공합니다.
ETW_BUFFER_CONTEXT 이벤트에 대한 컨텍스트 정보를 제공합니다. (ETW_BUFFER_CONTEXT)

ETW_BUFFER_HEADER ETW 버퍼의 헤더 구조입니다.
ETW_OPEN_TRACE_OPTIONS OpenTraceFromBufferStream, OpenTraceFromFile, OpenTraceFromRealTimeLogger, OpenTraceFromRealTimeLoggerWithAllocationOptions 함수에 대한 구성 매개 변수를 제공합니다.
ETW_TRACE_PARTITION_INFORMATION ETW 추적에서 가져온 파티션 정보를 포함합니다.
EVENT_DATA_DESCRIPTOR EVENT_DATA_DESCRIPTOR 구조체는 ETW 이벤트에 사용될 데이터 블록을 정의합니다.
EVENT_DESCRIPTOR EVENT_DESCRIPTOR 구조에는 ETW 이벤트에 대한 정보(메타데이터)가 포함됩니다.
EVENT_DESCRIPTOR 이벤트를 정의하는 메타데이터를 포함합니다.
EVENT_EXTENDED_ITEM_EVENT_KEY
EVENT_EXTENDED_ITEM_INSTANCE TraceEventInstance를 사용하여 관련 이벤트를 기록하는 경우 이벤트 간의 관계를 정의합니다.
EVENT_EXTENDED_ITEM_PEBS_INDEX
EVENT_EXTENDED_ITEM_PMC_COUNTERS
EVENT_EXTENDED_ITEM_PROCESS_START_KEY
EVENT_EXTENDED_ITEM_RELATED_ACTIVITYID 이 이벤트의 부모 이벤트를 정의합니다.
EVENT_EXTENDED_ITEM_STACK_KEY32
EVENT_EXTENDED_ITEM_STACK_KEY64

EVENT_EXTENDED_ITEM_STACK_TRACE32 32비트 컴퓨터에서 호출 스택을 정의합니다.
EVENT_EXTENDED_ITEM_STACK_TRACE64 64비트 컴퓨터에서 호출 스택을 정의합니다.
EVENT_EXTENDED_ITEM_TS_ID 이벤트를 기록한 터미널 세션을 정의합니다.
EVENT_FILTER_DESCRIPTOR 세션이 공급자의 콜백 사용 함수에 전달하는 필터 데이터를 정의합니다.
EVENT_FILTER_EVENT_ID 이벤트 ID 또는 스택 워크 필터의 EVENT_FILTER_DESCRIPTOR 구조에 사용되는 이벤트 ID를 정의합니다.
EVENT_FILTER_EVENT_NAME 이벤트 이름 또는 줄기 워크 이름 필터의 EVENT_FILTER_DESCRIPTOR 구조에 사용되는 이벤트 ID를 정의합니다.
EVENT_FILTER_HEADER 계측 매니페스트에 정의된 필터 데이터 앞에 있어야 하는 헤더 데이터를 정의합니다.
EVENT_FILTER_LEVEL_KW 스택 워크 수준 키워드 필터의 EVENT_FILTER_DESCRIPTOR 구조에 사용되는 이벤트 ID를 정의합니다.
EVENT_HEADER EVENT_HEADER 구조체(evntcons.h)는 이벤트에 대한 정보를 정의합니다.
EVENT_HEADER EVENT_HEADER 구조체(relogger.h)는 이벤트에 대한 정보를 정의합니다.
EVENT_HEADER_EXTENDED_DATA_ITEM EVENT_HEADER_EXTENDED_DATA_ITEM 구조체(evntcons.h)는 ETW가 이벤트 데이터의 일부로 수집하는 확장 데이터를 정의합니다.
EVENT_HEADER_EXTENDED_DATA_ITEM

EVENT_HEADER_EXTENDED_DATA_ITEM 구조체(relogger.h)는 ETW가 이벤트 데이터의 일부로 수집하는 확장 데이터를 정의합니다.
EVENT_INSTANCE_HEADER EVENT_INSTANCE_HEADER 구조에는 TraceEventInstance에서 작성한 모든 이벤트에 공통적인 표준 이벤트 추적 정보가 포함되어 있습니다.
EVENT_INSTANCE_INFO EVENT_INSTANCE_INFO 구조체는 TraceEventInstance에 대해 등록된 이벤트 추적 클래스에 고유한 트랜잭션 식별자를 매핑합니다.
EVENT_MAP_ENTRY 단일 값 맵 항목을 정의합니다.
EVENT_MAP_INFO 이벤트 맵에 대한 메타데이터를 정의합니다.
EVENT_PROPERTY_INFO 이벤트 또는 필터의 단일 속성에 대한 정보를 제공합니다.
EVENT_RECORD EVENT_RECORD 구조체(evntcons.h)는 ETW가 제공하는 이벤트의 레이아웃을 정의합니다.
EVENT_RECORD EVENT_RECORD 구조체(relogger.h)는 ETW가 제공하는 이벤트의 레이아웃을 정의합니다.
EVENT_TRACE EVENT_TRACE 구조는 이벤트 추적 소비자에게 이벤트 정보를 전달하는 데 사용됩니다.
EVENT_TRACE_HEADER EVENT_TRACE_HEADER 구조에는 TraceEvent에서 작성한 모든 이벤트에 공통적인 표준 이벤트 추적 정보가 포함되어 있습니다.
EVENT_TRACE_LOGFILEA ANSI(EVENT_TRACE_LOGFILEA) 구조체(evntrace.h)는 추적 데이터 원본에 대한 정보를 저장합니다.
EVENT_TRACE_LOGFILEW EVENT_TRACE_LOGFILEW(유니코드) 구조체(evntrace.h)는 추적 데이터 원본에 대한 정보를 저장합니다.

EVENT_TRACE_PROPERTIES EVENT_TRACE_PROPERTIES 구조는 이벤트 추적 세션에 대한 정보를 포함하며 StartTrace 및 ControlTrace와 같은 API와 함께 사용됩니다.
EVENT_TRACE_PROPERTIES_V2 EVENT_TRACE_PROPERTIES_V2 구조는 이벤트 추적 세션에 대한 정보를 포함하며 StartTrace 및 ControlTrace와 같은 API와 함께 사용됩니다.
MOF_FIELD MOF_FIELD 구조를 사용하여 이벤트 데이터를 EVENT_TRACE_HEADER 또는 EVENT_INSTANCE_HEADER 구조에 추가할 수 있습니다.
PAYLOAD_FILTER_PREDICATE 추적 세션의 단일 필드를 필터링하는 방법을 설명하는 이벤트 페이로드 필터 조건자를 정의합니다.
PROPERTY_DATA_DESCRIPTOR 검색할 속성을 정의합니다.
PROVIDER_ENUMERATION_INFO 컴퓨터에 MOF 또는 매니페스트를 등록한 공급자의 배열을 정의합니다.
PROVIDER_EVENT_INFO 공급자 매니페스트에서 이벤트 배열을 정의합니다.
PROVIDER_FIELD_INFO 필드 정보를 정의합니다.
PROVIDER_FIELD_INFOARRAY 요청된 필드에 대한 메타데이터 정보를 정의합니다.
PROVIDER_FILTER_INFO 필터 및 해당 데이터를 정의합니다.
TDH_CONTEXT 이벤트를 구문 분석하는 데 필요한 추가 정보를 정의합니다.
TRACE_CONTEXT_REGISTER_INFO

컨텍스트 등록 추적을 사용하도록 설정할 때 기록할 레지스터 집합을 식별합니다.
TRACE_ENABLE_INFO 세션과 세션이 공급자를 사용하도록 설정하는 데 사용한 정보를 정의합니다.
TRACE_EVENT_INFO 이벤트에 대한 정보를 정의합니다.
TRACE_GUID_INFO EnumerateTraceGuidsEx에서 반환됩니다. 공급자를 사용하도록 설정한 세션 목록에 대한 헤더를 정의합니다.
TRACE_GUID_PROPERTIES EnumerateTraceGuids에서 반환됩니다. 이벤트 추적 공급자에 대한 정보를 포함합니다.
TRACE_GUID_REGISTRATION RegisterTraceGuids와 함께 이벤트 추적 클래스를 등록하는 데 사용됩니다.
TRACE_LOGFILE_HEADER TRACE_LOGFILE_HEADER 구조에는 이벤트 추적 세션 및 해당 이벤트에 대한 정보가 포함됩니다.
TRACE_PERIODIC_CAPTURE_STATE_INFO TraceQueryInformation 및 TraceSetInformation과 함께 사용하여 주기적인 캡처 상태와 관련된 정보를 얻거나 설정합니다.
TRACE_PROVIDER_INFO 공급자의 GUID 및 이름을 정의합니다.
TRACE_PROVIDER_INSTANCE_INFO 공급자 GUID의 인스턴스를 정의합니다.
TRACE_VERSION_INFO TraceLogging 세션의 버전 정보를 결정합니다.

eventlogprov

아티클2023. 08. 22.

이벤트 로그 공급자 기술 개요입니다.

이벤트 로그 공급자 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 이벤트 로그 공급자

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

확장 가능한 인증 프로토콜 호스트

아티클2023. 03. 09.

확장 가능한 인증 프로토콜 호스트 기술에 대한 개요입니다.

Extensible Authentication 프로토콜 호스트를 개발하려면 다음 헤더가 필요합니다.

- eapauthenticatoractiondefine.h
- eapauthenticatortypes.h
- eaphostpeerconfigapis.h
- eaphostpeertypes.h
- eapmethodauthenticatorapis.h
- eapmethodpeerapis.h
- eapmethodtypes.h
- eappapis.h
- eaptypes.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 확장 가능한 인증 프로토콜 호스트

열거형

EAP_ATTRIBUTE_TYPE 인증 엔터티에서 사용할 수 있는 가능한 EAP 특성 형식 집합을 정의합니다.
EAP_AUTHENTICATOR_SEND_TIMEOUT 패킷이 전송된 후 사용자 입력을 기다리는 시간을 인증자 메서드에 나타냅니다. 제한 시간 값은 없음으로 설정할 수 있습니다.
EAP_CONFIG_INPUT_FIELD_TYPE 사용자 자격 증명을 쿼리할 때 사용할 수 있는 가능한 입력 필드 형식 집합을 정의합니다.
EAP_INTERACTIVE_UI_DATA_TYPE 특정 유연한 API 호출에 제공되는 대화형 UI 컨텍스트 데이터 형식의 집합을 지정합니다.
EAP_METHOD_AUTHENTICATOR_RESPONSE_ACTION 인증자가 지원자 또는 EAP 피어 메서드로 보낸 응답 명령 집합을 정의합니다.

EAP_METHOD_PROPERTY_TYPE 가능한 EAP 메서드 속성 집합을 지정합니다.
EAP_METHOD_PROPERTY_VALUE_TYPE EAP 메서드 속성 값에 대해 가능한 데이터 형식 집합을 정의합니다.
EapCode EAP 패킷 유형 집합을 정의합니다.
EapCredentialType EapPeerGetConfigBlobAndUserBlob 함수에 전달할 수 있는 가능한 EAP 자격 증명 집합을 정의합니다.
EAPHOST_AUTH_STATUS 인증 프로세스 중에 가능한 EAP 인증 세션 상태 값 집합을 정의합니다.
EapHostPeerAuthParams 가능한 인증 매개 변수 값 집합을 정의합니다.
EapHostPeerMethodResultReason EAP 메서드가 지원인에게 반환한 결과를 설명하는 가능한 이유 집합을 정의합니다.
EapHostPeerResponseAction EAP 인증자 또는 피어 메서드가 인증 중에 지원자에 나타낼 수 있는 작업 집합을 정의합니다.
EapPeerMethodResponseAction 인증 중에 EAP 인증자가 최적 또는 EAP 피어 메서드에 나타낼 수 있는 작업 집합을 정의합니다.
EapPeerMethodResultReason EAP 인증자 메서드에서 EAP 피어 메서드로 반환된 EAP 인증 세션의 결과 집합을 정의합니다.
ISOLATION_STATE 컴퓨터의 가능한 격리 상태 값 집합을 정의합니다.

Functions

EapHostPeerBeginSession EAP 인증 세션을 시작합니다.
EapHostPeerClearConnection 인증 세션 연결을 지웁니다.
EapHostPeerConfigBlob2Xml 구성 BLOB을 XML로 변환합니다. (EapHostPeerConfigBlob2Xml)
EapHostPeerConfigXml2Blob XML을 구성 BLOB으로 변환합니다. (EapHostPeerConfigXml2Blob)
EapHostPeerCredentialsXml2Blob 자격 증명 BLOB을 생성합니다.
EapHostPeerEndSession EAPHost와 호출 지원자 간의 현재 EAP 인증 세션을 종료하고 세션에 대해 저장된 데이터를 지웁니다.
EapHostPeerFreeEapError EAPHost 런타임 API에서 반환된 EAP_ERROR 구조를 해제합니다.
EapHostPeerFreeErrorMemory EAP_ERROR 구조에 할당된 메모리를 해제합니다.
EapHostPeerFreeMemory 구성 API에서 반환된 메모리를 해제합니다.
EapHostPeerFreeRuntimeMemory 런타임 중에 사용되는 메모리 공간을 해제합니다.
EapHostPeerGetAuthStatus EAPHost에서 지원자의 현재 EAP 인증 상태를 가져옵니다.

EapHostPeerGetDataToUnplumbCredentials EAPHost에서 SSO에 대한 자격 증명을 저장하는 데 사용하는 연결 ID, 사용자 가장 토큰 및 Eaphost 프로세스 ID를 반환합니다. 이 데이터는 이전에 배관된 자격 증명의 삭제를 해제하는 데 필요합니다.
EapHostPeerGetIdentity 이 함수는 내부 메서드에서 ID 정보를 요청하기 위해 터널 메서드에 의해 호출됩니다. 이 함수는 ID 및 사용자 자격 증명 정보를 반환합니다.
EapHostPeerGetMethodProperties 연결 및 사용자 데이터가 지정된 경우 EAP 메서드의 속성을 검색하는 데 사용됩니다.
EapHostPeerGetMethods 레거시 EAP 메서드를 포함하여 설치되고 사용할 수 있는 모든 EAP 메서드를 열거합니다.
EapHostPeerGetResponseAttributes EAPHost에서 EAP 인증 특성 배열을 가져옵니다.
EapHostPeerGetResult 지정된 EAP 인증 세션에 대한 인증 결과를 가져옵니다.
EapHostPeerGetSendPacket 지원자가 인증자로 보내기 위해 EAPHost에서 패킷을 가져와야 하는 경우 지원자가 호출합니다.
EapHostPeerGetUIContext UI를 발생시킬 경우 EAPHost에서 지원자의 사용자 인터페이스 컨텍스트를 가져옵니다.
EapHostPeerInitialize EAPHost 인증 세션을 초기화합니다.
EapHostPeerInvokeConfigUI 지정된 EAP 메서드의 구성 사용자 인터페이스를 시작합니다.
EapHostPeerInvokeIdentityUI 이 함수는 터널 메서드에 의해 호출되어 내부 메서드의 ID UI를 호출합니다. 이 함수는 인증을 시작하기 위해 사용할 자격 증명뿐만 아니라 ID를 반환합니다.

EapHostPeerInvokeInteractiveUI 사용자로부터 자격 증명을 가져오는 데 사용되는 대화형 사용자 인터페이스를 발생합니다.
EapHostPeerProcessReceivedPacket 지원자가 EAPHost에서 처리해야 하는 패킷을 받을 때마다 지원자가 호출합니다.
EapHostPeerQueryCredentialInputFields 사용자가 SSO(Single Sign-On) 시나리오에서 인증을 수행하는 데 필요한 자격 증명 종류를 확인할 수 있습니다.
EapHostPeerQueryInteractiveUIInputFields 지원자에서 발생할 대화형 UI 구성 요소에 대한 입력 필드를 가져옵니다.
EapHostPeerQueryUIBlobFromInteractiveUIInputFields 사용자 정보를 EAPHost 런타임 함수에서 사용할 수 있는 사용자 BLOB으로 변환합니다. (EapHostPeerQueryUIBlobFromInteractiveUIInputFields)
EapHostPeerQueryUserBlobFromCredentialInputFields SSO(Single Sign-On) UI에서 받은 사용자 입력에서 인증을 시작하는 데 사용할 수 있는 자격 증명 BLOB을 가져옵니다.
EapHostPeerSetResponseAttributes EAPHost에 업데이트된 EAP 인증 특성을 제공합니다.
EapHostPeerSetUIContext UI가 발생한 후 EAPHost에 로드된 EAP 피어 메서드에 새 사용자 인터페이스 컨텍스트 또는 업데이트된 사용자 인터페이스 컨텍스트를 제공합니다.
EapHostPeerUninitialize 모든 EAPHost 인증 세션을 초기화하지 않습니다.
EapMethodAuthenticatorBeginSession 서버 EAPHost에 새 EAP 인증 세션을 만듭니다.
EapMethodAuthenticatorEndSession 서버 EAPHost에서 EAP 인증 세션을 닫습니다.

EapMethodAuthenticatorFreeErrorMemory EAP 인증자 메서드에 의해 할당된 오류 관련 메모리를 해제합니다.
EapMethodAuthenticatorFreeMemory 불투명 사용자 인터페이스 컨텍스트 데이터 버퍼와 연결된 모든 메모리를 해제합니다. (EapMethodAuthenticatorFreeMemory)
EapMethodAuthenticatorGetAttributes EAP 인증자 메서드에서 EAP 인증 특성 배열을 가져옵니다.
EapMethodAuthenticatorGetInfo 로드된 EAP 인증자 메서드의 구현을 위한 함수 포인터 집합을 가져옵니다. EapMethodAuthenticatorGetInfo는 함수 프로토타입입니다.
EapMethodAuthenticatorGetResult EAP 인증자 메서드에서 인증 결과를 가져옵니다.
EapMethodAuthenticatorInitialize 서버 EAPHost에 대한 EAP 인증자 메서드를 초기화합니다.
EapMethodAuthenticatorInvokeConfigUI 클라이언트에서 EAP 메서드의 연결 구성 사용자 인터페이스 대화 상자를 발생시키는 함수를 정의합니다.
EapMethodAuthenticatorReceivePacket 서버 EAPHost에서 받은 EAP 인증 패킷을 처리하고 응답 작업을 반환합니다.
EapMethodAuthenticatorSendPacket EAP 인증자 메서드에서 인증 패킷을 가져와서 지원자에 보냅니다.
EapMethodAuthenticatorSetAttributes EAP 인증자 메서드에 설정할 업데이트된 EAP 인증 특성을 제공합니다.
EapMethodAuthenticatorShutdown EAP 인증자 메서드를 종료하고 서버 EAPHost에서 언로드할 준비를 합니다.

EapMethodAuthenticatorUpdateInnerMethodParams EAPHost 서버에서 EapMethodAuthenticatorBeginSession을 호출하여 이전에 설정한 EAP 인증 세션 설정을 업데이트.
EapPeerBeginSession EAP 메서드를 사용하여 피어 EAPHost에서 EAP 인증 세션을 시작합니다.
EapPeerConfigBlob2Xml 구성 BLOB을 XML로 변환합니다. (EapPeerConfigBlob2Xml)
EapPeerConfigXml2Blob XML을 구성 BLOB으로 변환합니다. (EapPeerConfigXml2Blob)
EapPeerCredentialsXml2Blob XML을 구성 BLOB으로 변환합니다. (EapPeerCredentialsXml2Blob)
EapPeerEndSession EAP 메서드에 대한 EAP 인증 세션을 종료합니다.
EapPeerFreeErrorMemory EAP 피어 메서드에서 할당한 오류 관련 메모리를 해제합니다.
EapPeerFreeMemory 불투명 사용자 인터페이스 컨텍스트 데이터 버퍼와 연결된 모든 메모리를 해제합니다. (EapPeerFreeMemory)
EapPeerGetConfigBlobAndUserBlob EAP 메서드 개발자가 메서드에서 지원하는 다양한 연결 속성 및 사용자 속성을 제공할 수 있습니다. EAPHost는 이 함수를 호출하여 EAP 메서드의 연결 속성 및 사용자 속성을 만듭니다.
EapPeerGetIdentity EAPHost에서 호출된 후 사용자 데이터 및 사용자 ID를 반환합니다.
EapPeerGetInfo 현재 EAPHost 서비스에 로드된 EAP 피어 메서드 EapPeerGetInfo 구현을 위한 함수 포인터 집합을 가져옵니다.

EapPeerGetMethodProperties 연결 및 사용자 데이터가 지정된 경우 EAP 메서드의 속성을 검색하는 EAP 메서드 관련 함수입니다.
EapPeerGetResponseAttributes EAP 메서드에서 EAP 응답 특성 배열을 가져옵니다.
EapPeerGetResponsePacket EAP 메서드에서 응답 패킷을 가져옵니다.
EapPeerGetResult EAP 메서드에서 인증 세션의 결과를 가져옵니다.
EapPeerGetUIContext EAP 메서드에서 사용자 인터페이스 컨텍스트를 가져옵니다.
EapPeerInitialize EAPHost에 대한 EAP 피어 메서드를 초기화합니다.
EapPeerInvokeConfigUI 클라이언트에서 EAP 메서드의 특정 연결 구성 사용자 인터페이스 대화 상자를 발생합니다.
EapPeerInvokeIdentityUI 사용자 지정 대화형 사용자 인터페이스 대화 상자를 발생시켜 클라이언트의 EAP 메서드에 대한 사용자 ID 정보를 가져옵니다.
EapPeerInvokeInteractiveUI 클라이언트에서 EAP 메서드에 대한 사용자 지정 대화형 사용자 인터페이스 대화 상자를 발생합니다.
EapPeerProcessRequestPacket EAPHost가 지원자로부터 받은 패킷을 처리합니다.
EapPeerQueryCredentialInputFields EAP 메서드에 대한 EAP SSO(Single Sign-On) 자격 증명 입력 필드를 가져오는 EAP 메서드 관련 함수의 구현을 정의합니다.

EapPeerQueryInteractiveUIInputFields 지원자에서 발생시킬 대화형 UI 구성 요소에 대한 입력 필드를 제공하는 EAP 메서드 API의 구현을 정의합니다.
EapPeerQueryUIBlobFromInteractiveUIInputFields 사용자 정보를 EAPHost 런타임 함수에서 사용할 수 있는 사용자 BLOB으로 변환합니다. (EapPeerQueryUIBlobFromInteractiveUIInputFields)
EapPeerQueryUserBlobFromCredentialInputFields 지원자에서 발생한 대화형 SSO(Single Sign-On) UI에 제공된 사용자 BLOB 데이터를 가져오는 EAP 메서드 함수의 구현을 정의합니다.
EapPeerSetCredentials EAP 메서드에 새 인증 또는 업데이트된 인증 자격 증명을 제공합니다.
EapPeerSetResponseAttributes EAP 메서드에 대한 EAP 응답 특성의 업데이트된 배열을 제공합니다.
EapPeerSetUIContext EAP 메서드에 사용자 인터페이스 컨텍스트를 제공합니다. 이 함수는 EapPeerGetUIContext 함수를 통해 UI가 발생한 후에 호출됩니다.
EapPeerShutdown EAP 메서드를 종료하고 해당 DLL을 언로드할 준비를 합니다.
NotificationHandler SoH(Health 문)가 변경되고 NAP(네트워크 액세스 보호) 시스템 연결의 재인증이 필요하다는 사실을 사용자에게 알릴 수 있습니다.

구조체

EAP_ATTRIBUTE EAP 특성을 포함합니다.

EAP_ATTRIBUTES EAP 특성의 배열을 포함합니다.
EAP_AUTHENTICATOR_METHOD_ROUTINES EAPHost Authenticator 메서드 API에 대한 함수 포인터 집합을 포함합니다.
EAP_CONFIG_INPUT_FIELD_ARRAY 사용자로부터 가져온 사용자 입력 필드 데이터를 전체적으로 포함하는 EAP_CONFIG_INPUT_FIELD_DATA 구조 집합을 포함합니다.
EAP_CONFIG_INPUT_FIELD_DATA 단일 입력 필드와 연결된 데이터를 포함합니다.
EAP_CRED_EXPIRY_REQ 자격 증명 만료 작업에 대한 이전 및 새 EAP 자격 증명을 모두 포함합니다.
EAP_ERROR EAPHost 작업 중에 발생한 오류에 대한 정보를 포함합니다.
EAP_INTERACTIVE_UI_DATA EAP 지원자에서 발생하는 대화형 UI 구성 요소에 대한 구성 정보를 포함합니다.
EAP_METHOD_AUTHENTICATOR_RESULT EAP 인증자 메서드에서 반환된 인증 결과를 포함합니다.
EAP_METHOD_INFO EAP 메서드에 대한 정보를 포함합니다. (EAP_METHOD_INFO)
EAP_METHOD_INFO_ARRAY 클라이언트 컴퓨터에 설치된 EAP 메서드에 대한 정보를 포함합니다.
EAP_METHOD_INFO_ARRAY_EX 클라이언트 컴퓨터에 설치된 모든 EAP 메서드에 대한 정보를 포함합니다.
EAP_METHOD_INFO_EX EAP 메서드에 대한 정보를 포함합니다. (EAP_METHOD_INFO_EX)

EAP_METHOD_PROPERTY EAP 메서드 속성을 포함합니다.
EAP_METHOD_PROPERTY_ARRAY EAP 메서드 속성의 배열을 포함합니다.
EAP_METHOD_PROPERTY_VALUE EAP 메서드 속성의 값을 포함합니다.
EAP_METHOD_PROPERTY_VALUE_BOOL EAP 메서드 속성의 부울 값을 포함합니다.
EAP_METHOD_PROPERTY_VALUE_DWORD EAP 메서드 속성의 DWORD 값을 포함합니다.
EAP_METHOD_PROPERTY_VALUE_STRING EAP 메서드 속성의 문자열 값을 포함합니다.
EAP_METHOD_TYPE EAP 메서드에 대한 형식, 식별 및 작성자 정보를 포함합니다.
EAP_PEER_METHOD_ROUTINES EAPHost 피어 메서드 API에 대한 함수 포인터 집합을 포함합니다.
EAP_TYPE EAP 메서드에 대한 유형 및 공급업체 식별 정보를 포함합니다.
EAP_UI_DATA_FORMAT EAP_UI_DATA_FORMAT 공용 구조체는 EAP_INTERACTIVE_UI_DATA 구조체의 pbUiData 멤버에 저장된 특성의 값을 지정합니다.
EapCertificateCredential EAP 메서드가 인증에 사용하는 인증서에 대한 정보를 포함합니다.
EapCredential 자격 증명 유형 및 적절한 자격 증명에 대한 정보를 포함합니다. EapPeerGetConfigBlobAndUserBlob API에 대한 입력으로 전달됩니다.

EAPHOST_AUTH_INFO EAP 인증 프로세스의 여러 단계에서 현재 인증 정보를 설명합니다.
EapHostPeerMethodResult EAP 메서드에 전달되는 인증 세션 중에 EAPHost에서 생성된 결과 데이터를 포함합니다.
EapPacket EAP 인증 세션 중에 전송된 불투명 데이터의 패킷을 포함합니다.
EapPeerMethodOutput EAP 피어 메서드에서 반환된 작업 정보를 포함합니다.
EapPeerMethodResult 인증 중에 EAP 메서드에서 생성된 결과 데이터를 포함합니다.
EapSimCredential 인증을 위해 EAP 메서드에서 사용하는 SIM에 대한 정보를 포함합니다.
EapUsernamePasswordCredential 사용자를 인증하기 위해 EAP 메서드에서 사용하는 사용자 이름 및 암호를 포함합니다.

확장할 수 있는 인증 프로토콜

아티클2023. 08. 22.

확장 가능한 인증 프로토콜 기술 개요입니다.

확장 가능한 인증 프로토콜을 개발하려면 다음 헤더가 필요합니다.

- raseapif.h
- rrascfg.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 확장할 수 있는 인증 프로토콜

열거형

PPP_EAP_ACTION PPP_EAP_ACTION 열거형 형식은 연결 관리자 인증 프로토콜을 대신하여 수행해야 하는 작업을 지정합니다.
RAS_AUTH_ATTRIBUTE_TYPE RAS_AUTH_ATTRIBUTE_TYPE 열거형 형식은 세션 인증에 사용되는 특성 값을 지정합니다.

Functions

초기화 시스템은 Initialize 메서드를 호출하여 지정된 컴퓨터를 사용하여 EAP 구성 세션을 초기화합니다.
RasEapFreeMemory RAS 연결 관리자는 RasEapFreeMemory를 호출하여 RasEapInvokeConfigUI, RasEapGetIdentity 및 RasEapInvokeInteractiveUI에서 반환한 메모리 버퍼를 해제합니다.
RasEapGetIdentity RAS 연결 관리자는 RasEapGetIdentity 함수를 호출하여 인증을 요청하는 사용자에 대한 ID 정보를 가져옵니다.

RasEapGetInfo RAS 연결 관리자는 RasEapGetInfo를 호출하여 지정된 인증 프로토콜에 대한 함수 포인터 집합을 가져옵니다.
RasEapInvokeConfigUI RAS 연결 관리자는 RasEapInvokeConfigUI 함수를 호출하여 사용자로부터 구성 정보를 가져오는 대화 상자를 표시합니다.
RasEapInvokeInteractiveUI RAS 연결 관리자는 RasEapInvokeInteractiveUI 함수를 호출하여 사용자로부터 인증 데이터를 가져오는 대화 상자를 표시합니다.
RouterInvokeConfigUI 시스템은 RouterInvokeConfigUI 메서드를 호출하여 두 라우터 간의 EAP 인증에 대한 구성 사용자 인터페이스를 호출합니다.
RouterInvokeCredentialsUI 시스템은 RouterInvokeCredentialsUI 메서드를 호출하여 두 라우터 간의 EAP 인증을 위한 자격 증명 사용자 인터페이스를 호출합니다.
ServerInvokeConfigUI 시스템은 ServerInvokeConfigUI 메서드를 호출하여 원격 액세스 클라이언트와 서버 간의 EAP 인증에 대한 구성 사용자 인터페이스를 호출합니다.
Uninitialize 시스템은 Uninitialize 메서드를 호출하여 지정된 EAP 구성 세션을 종료합니다.

인터페이스

IEAPProviderConfig EAP 공급자에 대한 UI입니다.

구조체

PPP_EAP_INFO PPP_EAP_INFO 구조는 EAP DLL에 있는 함수에 대한 포인터를 포함하여 인증 프로토콜에 대한 정보를 연결 관리자 제공합니다.
PPP_EAP_INPUT PPP_EAP_INPUT 구조는 RAS 연결 관리자 서비스 PPP 구현과 EAP 간의 상호 작용에 사용됩니다.
PPP_EAP_OUTPUT 인증 프로토콜은 PPP_EAP_OUTPUT 구조를 사용하여 RasEapMakeMessage 호출에서 반환할 때 요청을 전달하고 정보를 연결 관리자 상태.
PPP_EAP_PACKET PPP_EAP_PACKET 구조는 인증 프로토콜에서 처리되는 패킷에 대한 정보를 지정합니다.
RAS_AUTH_ATTRIBUTE RAS_AUTH_ATTRIBUTE 구조는 EAP 세션 중에 RAS_AUTH_ATTRIBUTE_TYPE 형식의 인증 특성을 전달하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

장애 조치(failover) 클러스터 WMI 공급자

아티클2023. 03. 09.

장애 조치(failover) 클러스터 WMI 공급자 기술 개요입니다.

장애 조치(failover) 클러스터 WMI 공급자 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 장애 조치(failover) 클러스터 WMI 공급자

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Hyper-V WMI 공급자 클러스터링 장애 조치(failover)

아티클2023. 08. 22.

Hyper-V WMI 공급자 기술을 클러스터링하는 장애 조치(failover) 개요입니다.

Hyper-V WMI 공급자를 클러스터링하는 장애 조치(failover) 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Hyper-V WMI 공급자 클러스터링 장애 조치(failover)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

팩스 서비스

아티클2023. 03. 09.

팩스 서비스 기술 개요입니다.

팩스 서비스를 개발하려면 다음 헤더가 필요합니다.

- faxcom.h
- faxcomex.h
- faxdev.h
- faxext.h
- faxroute.h
- fxsutility.h
- winfax.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 팩스 서비스

열거형

FAX_ACCESS_RIGHTS_ENUM FAX_ACCESS_RIGHTS_ENUM 열거형은 팩스 서버에 대한 액세스 권한을 정의합니다.
FAX_ACCESS_RIGHTS_ENUM_2 팩스 서버에 대한 액세스 권한을 정의합니다.
FAX_ACCOUNT_EVENTS_TYPE_ENUM 서버가 수신 대기 클라이언트에 보내는 특정 계정의 이벤트 알림 유형을 지정합니다.
FAX_COVERPAGE_TYPE_ENUM FAX_COVERPAGE_TYPE_ENUM 열거형은 표지 템플릿 파일이 로컬 컴퓨터 표지인지 아니면 서버 기반 표지인지를 정의합니다. 또한 파일이 사용되지 않음을 지정할 수도 있습니다.
FAX_DEVICE_RECEIVE_MODE_ENUM FAX_DEVICE_RECEIVE_MODE_ENUM 열거형은 디바이스가 들어오는 호출에 응답하는 방법을 정의합니다.

FAX_GROUP_STATUS_ENUM FAX_GROUP_STATUS_ENUM 열거형은 아웃바운드 라우팅 그룹에 대한 상태 유형을 정의합니다.
FAX_JOB_EXTENDED_STATUS_ENUM FAX_JOB_EXTENDED_STATUS_ENUM 열거형은 팩스 작업의 확장 상태 값을 정의합니다.
FAX_JOB_OPERATIONS_ENUM FAX_JOB_OPERATIONS_ENUM 열거형은 팩스 작업에서 수행할 수 있는 작업을 정의합니다. 이 열거형의 멤버는 비트 값이며 함께 사용할 수 있습니다.
FAX_JOB_STATUS_ENUM FAX_JOB_STATUS_ENUM 열거형은 팩스 작업의 상태 값을 정의합니다. 참고 멤버 fjsPAUSED 및 fjsNOLINE은 한정자입니다. 이 열거형의 다른 멤버와 함께 사용할 수 있습니다.
FAX_JOB_TYPE_ENUM FAX_JOB_TYPE_ENUM 열거형은 팩스 작업 유형을 정의합니다.
FAX_LOG_LEVEL_ENUM FAX_LOG_LEVEL_ENUM 열거형은 로깅 범주에 대한 이벤트 로깅 수준을 정의합니다.
FAX_PRIORITY_TYPE_ENUM FAX_PRIORITY_TYPE_ENUM 열거형은 아웃바운드 팩스의 우선 순위 유형을 정의합니다.
FAX_PROVIDER_STATUS_ENUM FAX_PROVIDER_STATUS_ENUM 열거형은 팩스 확장(팩스 서비스 공급자(FSP) 또는 팩스 인바운드 라우팅 확장)의 상태 값을 정의합니다.
FAX_RECEIPT_TYPE_ENUM FAX_RECEIPT_TYPE_ENUM 열거형은 아웃바운드 팩스에 대한 배달 보고서 유형(배달 확인 형식)을 정의합니다. 이 열거형의 멤버는 비트 값이며 함께 사용할 수 있습니다.
FAX_ROUTING_RULE_CODE_ENUM FAX_ROUTING_RULE_CODE_ENUM 열거형은 아웃바운드 라우팅에 대한 규칙을 정의합니다.
FAX_RULE_STATUS_ENUM FAX_RULE_STATUS_ENUM 열거형은 아웃바운드 라우팅 규칙에 대한 상태 유형을 정의합니다.

FAX_SCHEDULE_TYPE_ENUM FAX_SCHEDULE_TYPE_ENUM 열거형은 아웃바운드 팩스에 대한 예약 유형을 정의합니다.
FAX_SERVER_APIVERSION_ENUM FAX_SERVER_APIVERSION_ENUM 열거형은 팩스 API의 버전을 정의합니다. 아래 값은 지정한 것보다 이전 버전의 팩스 서비스에서 지원되지 않습니다.
FAX_SERVER_EVENTS_TYPE_ENUM FAX_SERVER_EVENTS_TYPE_ENUM 열거형은 팩스 서비스가 이벤트를 수신 대기하는 클라이언트 애플리케이션에 보내는 이벤트 유형을 정의합니다. 이 열거형의 멤버는 비트 값이며 함께 사용할 수 있습니다.
FAX_SMTP_AUTHENTICATION_TYPE_ENUM FAX_SMTP_AUTHENTICATION_TYPE_ENUM 열거형은 전자 메일을 통해 전송된 배달 확인에 대한 구성 옵션을 정의합니다.
SendToMode 애플리케이션 내에서 파일을 팩스로 전송하는 방법을 정의합니다. Windows Vista에서는 가능한 값이 하나뿐입니다.

Functions

추가 IFaxDeviceIds::Add 메서드는 디바이스의 ID를 사용하여 팩스 디바이스를 FaxDeviceIds 컬렉션에 추가합니다.
추가 IFaxOutboundRoutingGroups::Add 메서드는 IFaxOutboundRoutingGroups 인터페이스가 나타내는 컬렉션에 아웃바운드 라우팅 그룹을 추가합니다.
추가 IFaxOutboundRoutingRules::Add 메서드는 아웃바운드 라우팅 규칙(IFaxOutboundRoutingRule 인터페이스)을 IFaxOutboundRoutingRules 인터페이스로 정의된 컬렉션에 추가합니다.

추가 IFaxRecipients::Add 메서드는 FaxRecipients 컬렉션에 새 FaxRecipient 개체를 추가합니다.
AddAccount 팩스 서버에 팩스 계정을 추가하고 새 IFaxAccount 개체를 반환합니다.
AnswerCall IFaxDevice::AnswerCall 메서드를 사용하면 팩스 디바이스가 들어오는 전화에 응답합니다.
취소 Cancel 메서드는 들어오는 팩스 작업을 취소합니다.
취소 IFaxOutgoingJob::Cancel 메서드는 아웃바운드 팩스 작업을 취소합니다.
CanSendToFaxRecipient Windows Vista 함수 SendToFaxRecipient를 호출하는 메뉴 항목 또는 기타 UI를 사용할 수 있도록 할지 여부를 결정하기 위해 애플리케이션에서 호출합니다.
연결 IFaxServer::Connect 메서드는 팩스 클라이언트 애플리케이션을 지정된 팩스 서버에 연결합니다.
연결 Connect 메서드는 팩스 클라이언트 애플리케이션을 지정된 팩스 서버에 연결합니다.
ConnectedSubmit IFaxDocument::ConnectedSubmit 메서드는 단일 팩스 문서를 연결된 IFaxServer에 제출합니다. 이 메서드는 팩스의 받는 사람마다 하나씩 팩스 작업 ID 문자열 배열을 반환합니다.
ConnectedSubmit2 하나 이상의 팩스 문서를 연결된 FaxServer에 제출합니다.

CopyTiff CopyTiff 메서드는 인바운드 팩스 작업과 연결된 TIFF 클래스 F(태그가 지정된 이미지 파일 형식 클래스 F) 파일을 로컬 컴퓨터의 파일에 복사합니다.
CopyTiff CopyTiff 메서드는 인바운드 팩스 메시지와 연결된 TIFF 클래스 F(태그가 지정된 이미지 파일 형식 클래스 F) 파일을 로컬 컴퓨터의 파일에 복사합니다.
CopyTiff IFaxOutgoingJob::CopyTiff 메서드는 아웃바운드 팩스 작업과 연결된 태그가 지정된 이미지 파일 형식 클래스 F(TIFF 클래스 F) 파일을 로컬 컴퓨터의 파일에 복사합니다.
CopyTiff IFaxOutgoingMessage::CopyTiff 메서드는 아웃바운드 팩스 메시지와 연결된 TIFF 클래스 F(태그가 지정된 이미지 파일 형식 클래스 F) 파일을 로컬 컴퓨터의 파일에 복사합니다.
CreateDocument IFaxServer::CreateDocument 메서드는 지정된 FaxServer 개체에 대한 FaxDoc 개체를 만듭니다. FaxDoc 개체를 사용하면 사용자가 문서를 만들고 하나 이상의 팩스 받는 사람에게 전송할 수 있습니다.
삭제 Delete 메서드는 인바운드 팩스 보관함에서 지정된 팩스 메시지를 삭제합니다.
삭제 IFaxOutgoingMessage::D 삭제 메서드는 아웃바운드 보관함에서 팩스 메시지를 삭제합니다.
연결 끊기 IFaxServer::D isconnect 메서드는 팩스 서버에 대한 팩스 클라이언트 애플리케이션의 연결을 종료합니다. 클라이언트가 활성 팩스 서버에 연결되지 않은 경우 메서드가 실패합니다.
연결 끊기 IFaxServer::D isconnect 메서드는 팩스 서버에 대한 팩스 클라이언트 애플리케이션의 연결을 종료합니다.

FaxCompleteJobParamsA FaxCompleteJobParams 함수는 팩스 클라이언트 애플리케이션에 대한 FAX_COVERPAGE_INFO 구조와 FAX_JOB_PARAM 구조를 모두 만듭니다. (ANSI)
FaxCompleteJobParamsW FaxCompleteJobParams 함수는 팩스 클라이언트 애플리케이션에 대한 FAX_COVERPAGE_INFO 구조와 FAX_JOB_PARAM 구조를 모두 만듭니다. (유니코드)
FaxConnectFaxServerA FaxConnectFaxServer 함수는 팩스 클라이언트 애플리케이션을 로컬 팩스 서버에 연결합니다. 이 함수는 작업, 디바이스, 구성 및 문서 관리를 용이하게 하는 다른 팩스 클라이언트 함수를 호출하는 데 필요한 팩스 서버 핸들을 반환합니다. (ANSI)
FaxConnectFaxServerW FaxConnectFaxServer 함수는 팩스 클라이언트 애플리케이션을 로컬 팩스 서버에 연결합니다. 이 함수는 작업, 디바이스, 구성 및 문서 관리를 용이하게 하는 다른 팩스 클라이언트 함수를 호출하는 데 필요한 팩스 서버 핸들을 반환합니다. (유니코드)
FaxDevAbortOperation 팩스 서비스는 FaxDevAbortOperation 함수를 호출하여 팩스 서비스 공급자(FSP)가 FaxHandle 매개 변수로 지정된 팩스 작업에 대한 활성 팩스 작업을 종료하도록 요청합니다. 각 FSP는 FaxDevAbortOperation 함수를 내보내야 합니다.
FaxDevEndJob 팩스 서비스는 팩스 작업의 마지막 팩스 작업 후에 FaxDevEndJob 함수를 호출합니다. 각 팩스 서비스 공급자(FSP)는 FaxDevEndJob 함수를 내보내야 합니다.
FaxDevInitialize 팩스 서비스는 FSP(팩스 서비스 공급자) DLL을 로드한 후 서비스가 시작될 때마다 FaxDevInitialize 함수를 호출합니다. 각 FSP는 FaxDevInitialize 함수를 내보내야 합니다.
FaxDevReceive 팩스 서비스는 FaxDevReceive 함수를 호출하여 들어오는 팩스 전송을 팩스 서비스 공급자(FSP)에 신호를 전송합니다. 각 FSP는 FaxDevReceive 함수를 내보내야 합니다.

FaxDevReportStatus 팩스 서비스는 FaxDevReportStatus 함수를 호출하여 팩스 서비스 공급자(FSP)에게 개별 활성 팩스 작업에 대한 상태 정보 또는 팩스 작업 실패 후 상태 정보를 쿼리합니다.
FaxDevSend 팩스 서비스는 FaxDevSend 함수를 호출하여 팩스 서비스 공급자(FSP)에게 송신 팩스 전송을 시작해야 한다는 신호를 보냅니다. 각 FSP는 FaxDevSend 함수를 내보내야 합니다.
FaxDevShutdown 팩스 서비스는 FaxDevShutdown 함수를 호출하여 FSP(팩스 서비스 공급자)에게 서비스가 FSP의 DLL을 언로드하려고 했음을 알립니다. FaxDevShutdown은 FaxDevInitialize 함수에 의해 할당된 전역 리소스를 해제합니다.
FaxDevStartJob 팩스 서비스는 FaxDevStartJob 함수를 호출하여 새 팩스 작업을 초기화합니다. 또한 팩스 서비스는 FaxDevStartJob을 호출하여 각 팩스 작업의 시작을 팩스 서비스 공급자(FSP)에게 신호를 전송합니다. 각 FSP는 FaxDevStartJob 함수를 내보내야 합니다.
FaxDevVirtualDeviceCreation 팩스 서비스는 초기화 중에 FaxDevVirtualDeviceCreation 함수를 호출하여 FSP(팩스 서비스 공급자)가 가상 팩스 디바이스를 표시할 수 있도록 합니다.
FaxEnableRoutingMethodA FaxEnableRoutingMethod 함수는 특정 팩스 디바이스에 대한 팩스 라우팅 방법을 사용하거나 사용하지 않도록 설정합니다. 팩스 관리 애플리케이션은 일반적으로 디바이스 관리를 위해 이 함수를 호출합니다. (ANSI)
FaxEnableRoutingMethodW FaxEnableRoutingMethod 함수는 특정 팩스 디바이스에 대한 팩스 라우팅 방법을 사용하거나 사용하지 않도록 설정합니다. 팩스 관리 애플리케이션은 일반적으로 디바이스 관리를 위해 이 함수를 호출합니다. (유니코드)
FaxEnumGlobalRoutingInfoA FaxEnumGlobalRoutingInfo 함수는 특정 팩스 서버와 연결된 모든 팩스 라우팅 메서드를 열거합니다. (ANSI)

FaxEnumGlobalRoutingInfoW FaxEnumGlobalRoutingInfo 함수는 특정 팩스 서버와 연결된 모든 팩스 라우팅 메서드를 열거합니다. (유니코드)
FaxEnumJobsA FaxEnumJobs 함수는 클라이언트가 연결된 팩스 서버에서 큐에 대기 중인 모든 활성 팩스 작업을 열거합니다. 이 함수는 팩스 클라이언트 애플리케이션에 각 팩스 작업에 대한 자세한 정보를 반환합니다. (ANSI)
FaxEnumJobsW FaxEnumJobs 함수는 클라이언트가 연결된 팩스 서버에서 큐에 대기 중인 모든 활성 팩스 작업을 열거합니다. 이 함수는 팩스 클라이언트 애플리케이션에 각 팩스 작업에 대한 자세한 정보를 반환합니다. (유니코드)
FaxEnumPortsA FaxEnumPorts 함수는 클라이언트가 연결된 팩스 서버에 현재 연결된 모든 팩스 디바이스를 열거합니다. 이 함수는 팩스 클라이언트 애플리케이션에 각 팩스 포트에 대한 자세한 정보를 반환합니다. (ANSI)
FaxEnumPortsW FaxEnumPorts 함수는 클라이언트가 연결된 팩스 서버에 현재 연결된 모든 팩스 디바이스를 열거합니다. 이 함수는 팩스 클라이언트 애플리케이션에 각 팩스 포트에 대한 자세한 정보를 반환합니다. (유니코드)
FaxEnumRoutingMethodsA FaxEnumRoutingMethods 함수는 특정 팩스 디바이스에 대한 모든 팩스 라우팅 방법을 열거합니다. 함수는 팩스 클라이언트 애플리케이션에 각 라우팅 방법에 대한 정보를 반환합니다. (ANSI)
FaxEnumRoutingMethodsW FaxEnumRoutingMethods 함수는 특정 팩스 디바이스에 대한 모든 팩스 라우팅 방법을 열거합니다. 함수는 팩스 클라이언트 애플리케이션에 각 라우팅 방법에 대한 정보를 반환합니다. (유니코드)
FaxExtConfigChange FaxExtConfigChange 콜백 함수는 팩스 확장 DLL로 정의된 함수 이름의 자리 표시자입니다. 팩스 확장 DLL은 이 함수를 노출해서는 안 됩니다.

FaxExtFreeBuffer FaxExtFreeBuffer 콜백 함수는 이전에 FaxExtGetData 함수를 성공적으로 호출하여 할당한 메모리의 할당을 취소합니다.
FaxExtGetData FaxExtGetData 콜백 함수는 특정 디바이스 및 GUID에 대한 구성 데이터를 검색합니다.
FaxExtInitializeConfig 팩스 서비스는 FaxExtInitializeConfig 함수를 호출하여 팩스 확장 DLL을 초기화합니다. 서비스는 다른 팩스 확장 초기화 함수를 호출하기 전에 이 함수를 호출합니다.
FaxExtRegisterForEvents FaxExtRegisterForEvents 콜백 함수는 특정 디바이스 및 GUID와 관련된 구성 데이터 변경에 대한 알림을 위해 팩스 확장 DLL을 등록합니다.
FaxExtSetData FaxExtSetData 함수는 특정 디바이스 및 GUID에 대한 구성 데이터를 설정합니다.
FaxExtUnregisterForEvents FaxExtUnregisterForEvents 콜백 함수는 특정 디바이스 및 GUID와 관련된 구성 데이터 변경에 대한 알림을 위해 팩스 확장 DLL의 등록을 취소합니다.
FaxGetConfigurationA FaxGetConfiguration 함수는 클라이언트가 연결된 팩스 서버에 대한 전역 구성 설정을 팩스 클라이언트 애플리케이션으로 반환합니다. (ANSI)
FaxGetConfigurationW FaxGetConfiguration 함수는 클라이언트가 연결된 팩스 서버에 대한 전역 구성 설정을 팩스 클라이언트 애플리케이션으로 반환합니다. (유니코드)
FaxGetDeviceStatusA FaxGetDeviceStatus 함수는 팩스 클라이언트 애플리케이션의 관심 있는 팩스 디바이스에 대한 현재 상태 정보로 돌아갑니다. (ANSI)
FaxGetDeviceStatusW FaxGetDeviceStatus 함수는 팩스 클라이언트 애플리케이션의 관심 있는 팩스 디바이스에 대한 현재 상태 정보로 돌아갑니다. (유니코드)

FaxGetJobA 팩스 클라이언트 애플리케이션은 FaxGetJob 함수를 호출하여 지정된 큐 또는 활성 팩스 작업에 대한 자세한 정보를 검색합니다. 이 함수는 FAX_JOB_ENTRY 구조체의 정보를 반환합니다. (ANSI)
FaxGetJobW 팩스 클라이언트 애플리케이션은 FaxGetJob 함수를 호출하여 지정된 큐 또는 활성 팩스 작업에 대한 자세한 정보를 검색합니다. 이 함수는 FAX_JOB_ENTRY 구조체의 정보를 반환합니다. (유니코드)
FaxGetLoggingCategoriesA FaxGetLoggingCategories 함수는 클라이언트가 연결된 팩스 서버에 대한 현재 로깅 범주를 팩스 클라이언트 애플리케이션으로 반환합니다. (ANSI)
FaxGetLoggingCategoriesW FaxGetLoggingCategories 함수는 클라이언트가 연결된 팩스 서버에 대한 현재 로깅 범주를 팩스 클라이언트 애플리케이션으로 반환합니다. (유니코드)
FaxGetPortA FaxGetPort 함수는 지정된 팩스 포트에 대한 정보를 팩스 클라이언트 애플리케이션에 반환합니다. (ANSI)
FaxGetPortW FaxGetPort 함수는 지정된 팩스 포트에 대한 정보를 팩스 클라이언트 애플리케이션에 반환합니다. (유니코드)
FaxGetRoutingInfoA FaxGetRoutingInfo 함수는 특정 팩스 디바이스와 연결된 팩스 라우팅 방법에 대한 팩스 클라이언트 애플리케이션 라우팅 정보로 돌아갑니다. (ANSI)
FaxGetRoutingInfoW FaxGetRoutingInfo 함수는 특정 팩스 디바이스와 연결된 팩스 라우팅 방법에 대한 팩스 클라이언트 애플리케이션 라우팅 정보로 돌아갑니다. (유니코드)
FaxPrintCoverPageA FaxPrintCoverPage 함수는 팩스 전송 표지 페이지를 팩스 클라이언트 애플리케이션의 지정된 디바이스 컨텍스트로 인쇄합니다. (ANSI)

FaxPrintCoverPageW FaxPrintCoverPage 함수는 팩스 전송 표지 페이지를 팩스 클라이언트 애플리케이션의 지정된 디바이스 컨텍스트로 인쇄합니다. (유니코드)
FaxRegisterRoutingExtensionW FaxRegisterRoutingExtension 함수는 팩스 서비스에 팩스 라우팅 확장 DLL을 등록합니다. 이 함수는 새 라우팅 확장 DLL을 사용하도록 팩스 서비스 레지스트리를 구성합니다.
FaxRegisterServiceProviderW FaxRegisterServiceProvider 함수는 팩스 서비스 공급자 DLL을 팩스 서비스에 등록합니다. 이 함수는 팩스 서비스가 다시 시작될 때 새 팩스 서비스 공급자 DLL을 쿼리하고 사용하도록 팩스 서비스 레지스트리를 구성합니다.
FaxRouteDeviceChangeNotification 팩스 서비스는 FaxRouteDeviceChangeNotification 함수를 호출하여 팩스 라우팅 확장 DLL에 팩스 디바이스가 팩스 서버에서 제거되었거나 새 팩스 장치가 설치되었음을 알릴 수 있습니다.
FaxRouteDeviceEnable FaxRouteDeviceEnable 함수를 사용하면 팩스 라우팅 확장 DLL이 특정 팩스 디바이스에 대한 팩스 라우팅 방법을 쿼리, 사용 또는 사용하지 않도록 설정할 수 있습니다. 각 팩스 라우팅 확장 프로그램은 FaxRouteDeviceEnable 함수를 내보내야 합니다.
FaxRouteGetRoutingInfo FaxRouteGetRoutingInfo 함수는 팩스 라우팅 확장 DLL을 쿼리하여 특정 팩스 디바이스에 대한 라우팅 구성 데이터를 가져옵니다. 각 팩스 라우팅 확장 프로그램 DLL은 FaxRouteGetRoutingInfo 함수를 내보내야 합니다.
FaxRouteInitialize 팩스 서비스는 서비스가 시작될 때마다 FaxRouteInitialize 함수를 한 번 호출하여 팩스 라우팅 확장 DLL을 초기화합니다. 각 팩스 라우팅 확장 프로그램 DLL은 FaxRouteInitialize 함수를 내보내야 합니다.
FaxRouteSetRoutingInfo FaxRouteSetRoutingInfo 함수는 특정 팩스 디바이스에 대한 라우팅 구성 데이터를 수정합니다. 각 팩스 라우팅 확장 프로그램 DLL은 FaxRouteSetRoutingInfo 함수를 내보내야 합니다.

FaxSendDocumentA 팩스 클라이언트 애플리케이션은 FaxSendDocument 함수를 호출하여 송신 팩스 전송을 전송하는 팩스 작업을 대기합니다. (ANSI)
FaxSendDocumentForBroadcastA 팩스 클라이언트 애플리케이션은 FaxSendDocumentForBroadcast 함수를 호출하여 여러 받는 사람에게 동일한 송신 팩스 전송을 전송하는 여러 팩스 작업을 큐에 대기합니다. (ANSI)
FaxSendDocumentForBroadcastW 팩스 클라이언트 애플리케이션은 FaxSendDocumentForBroadcast 함수를 호출하여 여러 받는 사람에게 동일한 송신 팩스 전송을 전송하는 여러 팩스 작업을 큐에 대기합니다. (유니코드)
FaxSendDocumentW 팩스 클라이언트 애플리케이션은 FaxSendDocument 함수를 호출하여 송신 팩스 전송을 전송하는 팩스 작업을 대기합니다. (유니코드)
FaxSetConfigurationA 팩스 클라이언트 애플리케이션은 FaxSetConfiguration 함수를 호출하여 클라이언트가 연결된 팩스 서버에 대한 전역 구성 설정을 변경합니다. (ANSI)
FaxSetConfigurationW 팩스 클라이언트 애플리케이션은 FaxSetConfiguration 함수를 호출하여 클라이언트가 연결된 팩스 서버에 대한 전역 구성 설정을 변경합니다. (유니코드)
FaxSetGlobalRoutingInfoA 팩스 관리 애플리케이션은 FaxSetGlobalRoutingInfo 함수를 호출하여 팩스 서버에 전역적으로 적용되는 라우팅 우선 순위와 같은 팩스 라우팅 방법 데이터를 수정합니다. (ANSI)
FaxSetGlobalRoutingInfoW 팩스 관리 애플리케이션은 FaxSetGlobalRoutingInfo 함수를 호출하여 팩스 서버에 전역적으로 적용되는 라우팅 우선 순위와 같은 팩스 라우팅 방법 데이터를 수정합니다. (유니코드)
FaxSetJobA 팩스 클라이언트 애플리케이션은 FaxSetJob 함수를 호출하여 지정된 팩스 작업을 일시 중지, 다시 시작, 취소 또는 다시 시작합니다. (ANSI)

FaxSetJobW 팩스 클라이언트 애플리케이션은 FaxSetJob 함수를 호출하여 지정된 팩스 작업을 일시 중지, 다시 시작, 취소 또는 다시 시작합니다. (유니코드)
FaxSetLoggingCategoriesA 팩스 클라이언트 애플리케이션은 FaxSetLoggingCategories 함수를 호출하여 클라이언트가 연결된 팩스 서버에 대한 현재 로깅 범주를 수정합니다. (ANSI)
FaxSetLoggingCategoriesW 팩스 클라이언트 애플리케이션은 FaxSetLoggingCategories 함수를 호출하여 클라이언트가 연결된 팩스 서버에 대한 현재 로깅 범주를 수정합니다. (유니코드)
FaxSetPortA 팩스 클라이언트 애플리케이션은 FaxSetPort 함수를 호출하여 관심 있는 팩스 포트의 구성을 변경합니다. (ANSI)
FaxSetPortW 팩스 클라이언트 애플리케이션은 FaxSetPort 함수를 호출하여 관심 있는 팩스 포트의 구성을 변경합니다. (유니코드)
FaxSetRoutingInfoA 팩스 관리 애플리케이션은 FaxSetRoutingInfo 함수를 호출하여 특정 팩스 디바이스와 연결된 팩스 라우팅 방법에 대한 라우팅 정보를 수정합니다. (ANSI)
FaxSetRoutingInfoW 팩스 관리 애플리케이션은 FaxSetRoutingInfo 함수를 호출하여 특정 팩스 디바이스와 연결된 팩스 라우팅 방법에 대한 라우팅 정보를 수정합니다. (유니코드)
FaxStartPrintJobA 팩스 클라이언트 애플리케이션은 FaxStartPrintJob 함수를 호출하여 지정된 팩스 프린터에서 아웃바운드 팩스 전송 인쇄를 시작합니다. (ANSI)
FaxStartPrintJobW 팩스 클라이언트 애플리케이션은 FaxStartPrintJob 함수를 호출하여 지정된 팩스 프린터에서 아웃바운드 팩스 전송 인쇄를 시작합니다. (유니코드)
get__NewEnum IFaxDeviceIds::get__NewEnum 메서드는 FaxDeviceIds 컬렉션을 반복하는 데 사용할 수 있는 열거자 개체에 대한 참조를 반환합니다.

get__NewEnum IFaxDeviceProviders::get__NewEnum 메서드는 FaxDeviceProviders 컬렉션을 반복하는 데 사용할 수 있는 열거자 개체에 대한 참조를 반환합니다.
get__NewEnum IFaxDevices::get__NewEnum 메서드는 FaxDevices 컬렉션을 반복하는 데 사용할 수 있는 열거자 개체에 대한 참조를 반환합니다.
get__NewEnum IFaxInboundRoutingExtensions::get__NewEnum 메서드는 IFaxInboundRoutingExtensions 컬렉션을 반복하는 데 사용할 수 있는 열거자 개체에 대한 참조를 반환합니다.
get__NewEnum IFaxInboundRoutingMethods::get__NewEnum 메서드는 IFaxInboundRoutingMethods 컬렉션을 반복하는 데 사용할 수 있는 열거자 개체에 대한 참조를 반환합니다.
get__NewEnum get__NewEnum 메서드는 FaxIncomingJobs 컬렉션을 반복하는 데 사용할 수 있는 열거자 개체에 대한 참조를 반환합니다.
get__NewEnum IFaxOutboundRoutingGroups::get__NewEnum 메서드는 FaxOutboundRoutingGroups 컬렉션을 반복하는 데 사용할 수 있는 열거자 개체에 대한 참조를 반환합니다.
get__NewEnum IFaxOutboundRoutingRules::get__NewEnum 메서드는 FaxOutboundRoutingRules 컬렉션을 반복하는 데 사용할 수 있는 열거자 개체에 대한 참조를 반환합니다.
get__NewEnum IFaxOutgoingJobs::get__NewEnum 메서드는 IFaxOutgoingJobs 컬렉션을 반복하는 데 사용할 수 있는 열거자 개체에 대한 참조를 반환합니다.
get__NewEnum IFaxRecipients::get__NewEnum 메서드는 FaxRecipients 컬렉션을 반복하는 데 사용할 수 있는 열거자 개체에 대한 참조를 반환합니다.

get_AccountName 서버에서 특정 팩스 계정의 이름을 검색합니다.
get_Activity IFaxServer::get_Activity 속성은 IFaxActivity 인터페이스 개체를 만듭니다. 이 인터페이스는 팩스 클라이언트 애플리케이션이 연결된 팩스 서버의 활동 및 팩스 서버 상태에 대한 정보에 액세스할 수 있도록 허용합니다.
get_ActivityLogging ActivityLogging 속성은 FaxActivityLogging 구성 개체를 검색합니다.
get_Address 부모 FaxPort 개체의 FaxStatus 개체에 대한 Address 속성을 검색합니다. Address 속성은 팩스 작업의 대상을 포함하는 null로 끝나는 문자열입니다.
get_AgeLimit AgeLimit 속성은 팩스 서비스가 인바운드 팩스 보관에 팩스 메시지를 보관하는 일 수를 나타내는 값입니다. (Get)
get_AgeLimit IFaxOutgoingArchive::get_AgeLimit 속성은 팩스 서비스가 아웃바운드 팩스 보관함에서 팩스 메시지를 보존하는 일 수를 나타내는 값입니다. (Get)
get_AgeLimit IFaxOutgoingQueue::get_AgeLimit 속성은 팩스 서비스가 팩스 작업 큐에서 전송되지 않은 작업을 유지하는 일 수를 나타내는 값입니다. (Get)
get_AllowedReceipts IFaxReceiptOptions::get_AllowedReceipts 속성은 허용되는 배달 영수증 유형을 지정하는 값입니다. (Get)
get_AllowPersonalCoverPages 개인 표지가 허용되는지 여부를 나타내는 값을 설정하거나 검색합니다. (Get)
get_AllowPersonalCoverPages AllowPersonalCoverPages 속성은 팩스 클라이언트 애플리케이션에 팩스 전송이 있는 사용자가 디자인한 표지를 포함할 수 있는지 여부를 나타내는 부울 값입니다. (Get)

get_APIVersion IFaxServer::get_APIVersion 속성은 팩스 서버 API의 버전을 나타내는 값입니다.
get_ArchiveAgeLimit 팩스 메시지가 서버에 보관되는 기간을 나타내는 값을 설정하거나 검색합니다. (Get)
get_ArchiveDirectory IFaxServer::get_ArchiveDirectory 메서드는 FaxServer 개체에 대한 ArchiveDirectory 속성을 검색합니다. ArchiveDirectory 속성은 팩스 서버에서 보관된 아웃바운드 팩스를 저장하는 위치를 포함하는 null로 종료된 문자열입니다. (Get)
get_ArchiveFolder ArchiveFolder 속성은 보관된 인바운드 팩스에 대한 팩스 서버의 폴더 위치를 지정하는 null로 종료된 문자열입니다. (Get)
get_ArchiveFolder IFaxOutgoingArchive::get_ArchiveFolder 속성은 보관된 아웃바운드 팩스에 대한 팩스 서버의 폴더 위치를 지정하는 null로 끝나는 문자열입니다. (Get)
get_ArchiveLocation 서버에서 보관 파일의 위치를 나타내는 값을 설정하거나 검색합니다. (Get)
get_ArchiveOutboundFaxes FaxServer 개체에 대한 ArchiveOutboundFaxes 속성을 설정하거나 검색합니다. ArchiveOutboundFaxes 속성은 팩스 서버에서 보내는 팩스 전송을 보관하는지 여부를 나타내는 부울 값입니다. (Get)
get_ArchiveSizeHigh 팩스 메시지 보관 파일의 크기에 대한 상위 32비트 값(바이트)을 지정하는 값입니다.
get_ArchiveSizeLow 팩스 메시지 보관 파일의 크기에 대한 낮은 순서의 32비트 값(바이트)을 지정하는 값입니다.

get_AreaCode IFaxOutboundRoutingRule::get_AreaCode 속성은 아웃바운드 라우팅 규칙이 적용되는 영역 코드를 지정합니다.
get_AtEOF AtEOF 속성은 인바운드 팩스 메시지 보관에 대한 파일 표식의 끝입니다.
get_AtEOF AtEOF 속성은 아웃바운드 팩스 메시지 보관에 대한 파일 끝 표식입니다.
get_AttachFaxToReceipt IFaxDocument::get_AttachFaxToReceipt 속성은 영수증에 팩스를 첨부할지 여부를 나타냅니다. (Get)
get_AuthenticationType IFaxReceiptOptions::get_AuthenticationType 속성은 SMTP(Simple Mail Transport Protocol) 서버에 연결할 때 팩스 서비스에서 사용하는 인증 유형을 지정합니다. (Get)
get_AutoCreateAccountOnConnect 연결이 시작되면 서버에서 팩스 계정을 자동으로 만드는지 여부를 나타내는 값을 설정하거나 검색합니다. (Get)
get_AvailableOperations FaxIncomingJob 개체의 AvailableOperations 속성을 검색합니다. AvailableOperations 속성은 현재 상태가 지정된 경우 팩스 작업에서 수행할 수 있는 유효한 작업의 조합을 나타냅니다.
get_AvailableOperations AvailableOperations 속성은 현재 상태를 고려하여 팩스 작업에서 수행할 수 있는 유효한 작업의 조합을 나타냅니다.
get_AvailableOperations IFaxOutgoingJob::get_AvailableOperations 속성은 현재 상태를 고려하여 팩스 작업에서 수행할 수 있는 유효한 작업의 조합을 나타냅니다.
get_BillingCode IFaxSender::get_BillingCode 속성은 보낸 사람과 연결된 청구 코드를 포함하는 null로 끝나는 문자열입니다. (Get)

get_BillingCode FaxDoc 개체의 BillingCode 속성을 설정하거나 검색합니다. BillingCode 속성은 팩스 전송에 적용되는 선택적 청구 코드를 포함하는 null로 끝나는 문자열입니다. (Get)
get_BillingCode IFaxJob::get_BillingCode 속성은 팩스 작업에 적용되는 선택적 청구 코드를 포함하는 null로 종료된 문자열입니다.
get_Blocked Blocked 속성은 들어오는 팩스의 작업 큐가 차단되었는지 여부를 나타내는 부울 값입니다. (Get)
get_Blocked IFaxOutgoingQueue::get_Blocked 속성은 보내는 팩스의 작업 큐가 차단되었는지 여부를 나타내는 부울 값입니다. (Get)
get_Bodies 팩스 문서에 하나 이상의 문서 컬렉션을 제공합니다. (Get)
get_Body IFaxDocument::get_Body 속성은 팩스 본문을 구성하는 파일의 경로를 제공합니다. 팩스 본문은 표지가 아닌 팩스 페이지로 구성됩니다. (Get)
get_Branding 팩스 서버에서 보내는 팩스에 브랜딩 표시를 생성하는지 여부를 나타내는 값을 설정하거나 검색합니다. (Get)
get_Branding IFaxOutgoingQueue::get_Branding 속성은 팩스 서비스가 송신 팩스 전송 맨 위에 브랜드(배너)를 생성하는지 여부를 나타내는 부울 값입니다. (Get)
get_Branding FaxServer 개체의 브랜딩 속성을 설정하거나 검색합니다. 브랜딩 속성은 팩스 서버가 팩스 전송 맨 위에 브랜딩 정보를 생성하는지 여부를 나타내는 부울 값입니다. (Get)

get_CallerId CallerId 속성은 인바운드 팩스 작업을 보낸 호출 디바이스를 식별하는 문자열입니다.
get_CallerId CallerId 속성은 인바운드 팩스 메시지와 연결된 호출 디바이스를 식별하는 null로 종료된 문자열입니다.
get_CallerId CallerId 속성은 팩스 작업과 연결된 호출 디바이스를 식별하는 null로 종료된 문자열입니다.
get_CallerId 부모 FaxPort 개체의 FaxStatus 개체에 대한 CallerId 속성을 검색합니다. CallerId 속성은 인바운드 팩스 작업을 보낸 호출 디바이스를 식별하는 문자열입니다.
get_CallerId FaxTiff 개체의 CallerId 속성을 검색합니다. CallerId 속성은 지정된 팩스 파일을 보낸 호출 디바이스를 식별하는 문자열입니다.
get_CanModify IFaxPort::get_CanModify 속성은 사용자에게 팩스 포트에 대한 구성 정보를 수정할 수 있는 권한이 있는지 여부를 나타내는 부울 값입니다.
get_Company IFaxSender::get_Company 속성은 보낸 사람과 연결된 회사 이름을 포함하는 null로 끝나는 문자열입니다. (Get)
get_Configuration IFaxServer2::Configuration 속성은 IFaxConfiguration 개체를 보유합니다. 이 개체는 팩스 클라이언트 애플리케이션이 연결된 팩스 서버의 구성에 대한 정보에 액세스할 수 있도록 허용합니다.
get_Count IFaxAccounts 컬렉션의 항목 수를 보유합니다.
get_Count IFaxDeviceIds::get_Count 속성은 FaxDeviceIds 컬렉션의 개체 수를 나타냅니다. 팩스 서버와 연결된 총 디바이스 ID 수입니다.

get_Count Count 속성은 FaxDeviceProviders 컬렉션의 개체 수를 나타냅니다. 팩스 서버와 연결된 총 팩스 디바이스 공급자 수입니다.
get_Count IFaxDevices::get_Count 속성은 FaxDevices 컬렉션의 개체 수를 나타냅니다. 팩스 서버에서 사용하는 총 디바이스 수입니다.
get_Count IFaxInboundRoutingExtensions::get_Count 속성은 IFaxInboundRoutingExtensions 컬렉션의 개체 수를 나타냅니다. 팩스 서버와 연결된 인바운드 라우팅 확장의 총 수입니다.
get_Count IFaxInboundRoutingMethods::get_Count 속성은 IFaxInboundRoutingMethods 컬렉션의 개체 수를 나타냅니다. 팩스 서버와 연결된 인바운드 라우팅 방법의 총 수입니다.
get_Count Count 속성은 FaxIncomingJobs 컬렉션의 개체 수를 나타냅니다. 팩스 서버에 대해 들어오는 작업의 총 수입니다.
get_Count Count 속성은 FaxOutboundRoutingGroups 컬렉션의 개체 수를 나타냅니다. 팩스 서버와 연결된 아웃바운드 라우팅 그룹의 총 수입니다.
get_Count IFaxOutboundRoutingRules::get_Count 속성은 FaxOutboundRoutingRules 컬렉션의 개체 수를 나타냅니다. 팩스 서버와 연결된 아웃바운드 라우팅 규칙의 총 수입니다.
get_Count IFaxOutgoingJobs::get_Count 속성은 FaxOutgoingJobs 컬렉션의 개체 수를 나타냅니다. 팩스 서버에 대한 나가는 작업의 총 수입니다.
get_Count IFaxRecipients::get_Count 속성은 FaxRecipients 컬렉션의 개체 수를 나타냅니다. 팩스 서버 또는 팩스 문서와 연결된 받는 사람의 총 수입니다.

get_Count IFaxJobs::get_Count 메서드는 연결된 팩스 서버와 연결된 대기 중 팩스 작업 수를 반환합니다.
get_Count IFaxPorts::get_Count 메서드는 연결된 팩스 서버에 연결된 팩스 포트 수를 검색합니다.
get_Count IFaxRoutingMethods::get_Count FaxPort 개체와 연결된 팩스 라우팅 메서드의 수를 반환합니다.
get_CountryCode IFaxOutboundRoutingRule::get_CountryCode 속성은 아웃바운드 라우팅 규칙이 적용되는 국가/지역 코드를 지정합니다.
get_CoverPage IFaxDocument::get_CoverPage 속성은 팩스 문서와 연결할 표지 서식 파일(.cov)의 이름을 포함하는 null로 끝나는 문자열입니다. (Get)
get_CoverpageName FaxDoc 개체의 CoverpageName 속성을 설정하거나 검색합니다. CoverpageName 속성은 개체와 연결된 표지 템플릿 파일(.cov)의 이름을 포함하는 null로 끝나는 문자열입니다. (Get)
get_CoverpageNote FaxDoc 개체의 CoverpageNote 속성을 설정하거나 검색합니다. CoverpageNote 속성은 팩스 전송과 관련된 보낸 사람의 메시지 또는 메모 텍스트를 포함하는 null로 끝나는 문자열입니다. (Get)
get_CoverpageSubject FaxDoc 개체의 CoverpageSubject 속성을 설정하거나 검색합니다. CoverpageSubject 속성은 팩스 전송의 제목 줄을 포함하는 null로 끝나는 문자열입니다. (Get)
get_CoverPageType IFaxDocument::get_CoverPageType 속성은 지정된 표지 템플릿 파일(.cov)이 서버 기반 표지 파일인지 아니면 로컬 컴퓨터 기반 표지 파일인지 여부를 나타내는 열거형의 값입니다. (Get)

get_Csid IFaxPort::get_Csid 속성은 팩스 포트와 연결된 CSID(호출된 스테이션 식별자)를 포함하는 null로 종료된 문자열입니다. (Get)
get_Csid 부모 FaxPort 개체의 FaxStatus 개체에 대한 Csid 속성을 검색합니다. Csid 속성은 일반적으로 수신 디바이스의 팩스 번호인 CSID(스테이션 식별자) 정보를 포함하는 문자열입니다.
get_Csid FaxTiff 개체의 Csid 속성을 검색합니다. Csid 속성은 일반적으로 지정된 팩스 파일을 받은 디바이스의 팩스 번호인 CSID(스테이션 식별자) 정보를 포함하는 문자열입니다.
get_CSID IFaxDevice::get_CSID 속성은 디바이스에 대한 CSID(호출된 스테이션 식별자)를 포함하는 null로 끝나는 문자열입니다. (Get)
get_CSID CSID 속성은 작업에 대해 호출된 CSID(스테이션 식별자)를 포함하는 null로 끝나는 문자열입니다. (IFaxIncomingJob.get_CSID)
get_CSID CSID 속성은 인바운드 팩스 메시지에 대한 CSID(호출된 스테이션 식별자)를 포함하는 null로 종료된 문자열입니다.
get_CSID CSID 속성은 작업에 대해 호출된 CSID(스테이션 식별자)를 포함하는 null로 끝나는 문자열입니다. (IFaxJobStatus.get_CSID)
get_CSID IFaxOutgoingJob::get_CSID 속성은 팩스 아웃바운드 작업과 연결된 CSID(호출된 스테이션 식별자)를 포함하는 null로 종료된 문자열입니다.
get_CSID IFaxOutgoingMessage::get_CSID 속성은 팩스 메시지에 대한 CSID(호출된 스테이션 식별자)를 포함하는 null로 끝나는 문자열입니다.

get_CurrentAccount IFaxServer2::CurrentAccount 속성은 팩스 서버에 연결된 사용자 계정의 팩스 계정을 반환합니다.
get_CurrentPage CurrentPage 속성은 인바운드 팩스 작업에서 팩스 서비스가 적극적으로 수신하는 페이지를 식별하는 숫자입니다.
get_CurrentPage CurrentPage 속성은 팩스 서비스가 현재 처리 중인 페이지를 식별하는 숫자입니다.
get_CurrentPage IFaxOutgoingJob::get_CurrentPage 속성은 팩스 서비스가 아웃바운드 팩스 작업에서 적극적으로 전송하는 페이지를 식별하는 숫자입니다.
get_CurrentPage 부모 FaxPort 개체의 FaxStatus 개체에 대한 CurrentPage 속성을 검색합니다. CurrentPage 속성은 특정 포트에서 활성 아웃바운드 팩스 작업의 현재 페이지를 식별하는 숫자입니다.
get_DatabasePath IFaxActivityLogging::get_DatabasePath 속성은 활동 로그 데이터베이스 파일의 경로를 포함하는 null로 끝나는 문자열입니다. (Get)
get_Debug IFaxDeviceProvider::get_Debug 속성은 디버그 환경에서 팩스 서비스 공급자(FSP)Â DLL이 만들어졌는지 여부를 나타내는 부울 값입니다.
get_Debug IFaxInboundRoutingExtension::get_Debug 속성은 디버그 환경에서 팩스 라우팅 확장 DLL이 만들어졌는지 여부를 나타내는 부울 값입니다.
get_Debug IFaxServer::get_Debug 속성은 디버그 환경에서 팩스 서버가 만들어졌는지 여부를 나타내는 부울 값입니다.

get_Department IFaxSender::get_Department 속성은 보낸 사람과 연결된 부서를 포함하는 null로 종료된 문자열입니다. (Get)
get_Description IFaxDevice::get_Description 속성은 팩스 디바이스에 대한 사용자 친화적인 설명을 포함하는 null로 끝나는 문자열입니다. 이 문자열은 사용자에게 표시하기에 적합합니다. (Get)
get_Description 부모 FaxPort 개체의 FaxStatus 개체에 대한 Description 속성을 검색합니다. Description 속성은 지정된 포트의 현재 상태를 설명하는 null로 끝나는 문자열입니다.
get_Descriptor 설명자 속성은 IFaxServer 개체의 보안 설명자를 나타냅니다. (Get)
get_Descriptor IFaxServer2 개체의 보안 설명자를 나타냅니다. (Get)
get_DeviceId DeviceId 속성은 인바운드 팩스 작업을 수신하는 디바이스의 디바이스 ID를 나타냅니다.
get_DeviceId DeviceId 속성은 팩스 작업과 연결된 디바이스의 디바이스 ID를 나타냅니다.
get_DeviceId IFaxOutboundRoutingRule::get_DeviceId 속성은 아웃바운드 라우팅 규칙이 단일 팩스 디바이스를 가리키는 경우 디바이스 ID를 지정합니다. (Get)
get_DeviceId IFaxOutgoingJob::get_DeviceId 속성은 아웃바운드 팩스 작업을 전송하는 디바이스의 디바이스 ID를 나타냅니다.
get_DeviceId IFaxPort::get_DeviceId 속성은 팩스 포트의 영구 회선 식별자를 나타내는 숫자입니다.

get_DeviceId IFaxRoutingMethod::get_DeviceId 속성은 팩스 포트의 줄 식별자를 나타내는 숫자입니다.	
get_DeviceId 부모 FaxPort 개체의 FaxStatus 개체에 대한 DeviceId 속성을 검색합니다. DeviceId 속성은 팩스 포트의 영구 회선 식별자를 나타내는 숫자입니다.	
get_DeviceIds IFaxDeviceProvider::get_DeviceIds 속성은 긴 형식의 안전 배열을 반환합니다(VT_I4	VT_ARRAY). 배열의 각 긴 값은 디바이스 ID입니다.
get_DeviceIds DeviceIds 속성은 아웃바운드 라우팅 그룹에 참여하는 디바이스 ID의 정렬된 컬렉션을 나타내는 인터페이스를 검색합니다.	
get_DeviceName IFaxDevice::get_DeviceName 속성은 팩스 디바이스의 이름을 포함하는 null로 끝나는 문자열입니다.	
get_DeviceName DeviceName 속성은 인바운드 팩스 메시지를 받은 디바이스의 이름을 포함하는 null로 끝나는 문자열입니다.	
get_DeviceName IFaxOutgoingMessage::get_DeviceName 속성은 팩스 메시지가 전송된 디바이스의 이름을 포함하는 null로 끝나는 문자열입니다.	
get_DeviceName IFaxRoutingMethod::get_DeviceName 속성은 팩스 포트에 대한 사용자에게 친숙한 표시 이름을 포함하는 null로 끝나는 문자열입니다.	
get_DeviceName 부모 FaxPort 개체의 FaxStatus 개체에 대한 DeviceName 속성을 검색합니다. DeviceName 속성은 팩스 포트에 대한 사용자에게 친숙한 표시 이름을 포함하는 null로 끝나는 문자열입니다.	

get_DeviceStatus IFaxJob::get_DeviceStatus 속성은 팩스 작업과 연결된 포트의 상태를 설명하는 null로 종료된 문자열입니다.
get_DirtyDays FaxServer 개체의 DirtyDays 속성을 설정하거나 검색합니다. DirtyDays 속성은 팩스 서버가 팩스 작업 큐에서 전송되지 않은 작업을 유지하는 일 수입니다. (Get)
get_DiscountRateEnd 할인율 기간이 종료되는 시간을 나타내는 값을 설정하거나 검색합니다. (Get)
get_DiscountRateEnd IFaxOutgoingQueue::get_DiscountRateEnd 속성은 팩스 전송 할인 기간이 끝나는 시간을 나타내는 값입니다. 할인 기간은 발신 팩스에 적용됩니다. (Get)
get_DiscountRateEndHour FaxServer 개체에 대한 DiscountRateEndHour 속성을 설정하거나 검색합니다. DiscountRateEndHour 속성은 할인 기간이 종료되는 시간을 나타내는 숫자입니다. 할인 기간은 발신 팩스 전송에만 적용됩니다. (Get)
get_DiscountRateEndMinute FaxServer 개체에 대한 DiscountRateEndMinute 속성을 설정하거나 검색합니다. DiscountRateEndMinute 속성은 할인 기간이 종료되는 분을 나타내는 숫자입니다. 할인 기간은 발신 팩스 전송에만 적용됩니다. (Get)
get_DiscountRateStart 할인율 기간이 시작되는 시간을 나타내는 값을 설정하거나 검색합니다. (Get)
get_DiscountRateStart IFaxOutgoingQueue::get_DiscountRateStart 속성은 팩스 전송 할인 기간이 시작되는 시간을 나타내는 값입니다. 할인 기간은 발신 팩스에 적용됩니다. (Get)
get_DiscountRateStartHour FaxServer 개체에 대한 DiscountRateStartHour 속성을 설정하거나 검색합니다. DiscountRateStartHour 속성은 할인 기간이 시작되는 시간을 나타내는 숫자입니다. 할인 기간은 발신 팩스 전송에만 적용됩니다. (Get)

get_DiscountRateStartMinute FaxServer 개체에 대한 DiscountRateStartMinute 속성을 설정하거나 검색합니다. DiscountRateStartMinute 속성은 할인 기간이 시작되는 분을 나타내는 숫자입니다. 할인 기간은 발신 팩스 전송에만 적용됩니다. (Get)
get_DiscountSend FaxDoc 개체의 DiscountSend 속성을 설정하거나 검색합니다. DiscountSend 속성은 팩스 서버가 할인 기간 동안 팩스를 전송하는지 여부를 나타내는 부울 값입니다. (Get)
get_DiscountSend IFaxJob::get_DiscountSend 속성은 팩스 서버가 할인율 기간 동안 팩스 작업을 전송할지 여부를 나타내는 부울 값입니다. 할인 기간은 발신 팩스 전송에만 적용됩니다.
get_DisplayName FaxDoc 개체의 DisplayName 속성을 설정하거나 검색합니다. DisplayName 속성은 팩스 문서와 연결할 이름을 포함하는 null로 끝나는 문자열입니다. (Get)
get_DisplayName IFaxJob::get_DisplayName 속성은 팩스 작업과 연결할 사용자 친화적인 이름을 포함하는 null로 종료된 문자열입니다.
get_DocumentName IFaxDocument::get_DocumentName 속성은 팩스 문서에 표시할 사용자 친화적인 이름을 포함하는 null로 끝나는 문자열입니다. 이 값은 표시 목적으로만 사용됩니다. (Get)
get_DocumentName IFaxOutgoingJob::get_DocumentName 속성은 팩스 문서에 표시할 사용자 친화적인 이름을 포함하는 null로 끝나는 문자열입니다.
get_DocumentName IFaxOutgoingMessage::get_DocumentName 속성은 팩스 메시지에 표시할 사용자 친화적인 이름을 포함하는 null로 끝나는 문자열입니다.

get_DocumentName 부모 FaxPort 개체의 FaxStatus 개체에 대한 DocumentName 속성을 검색합니다. DocumentName 속성은 활성 팩스 문서와 연결된 사용자에게 친숙한 이름을 포함하는 null로 끝나는 문자열입니다.
get_DocumentSize 부모 FaxPort 개체의 FaxStatus 개체에 대한 DocumentSize 속성을 검색합니다. DocumentSize 속성은 특정 포트에서 활성 아웃바운드 작업과 연결된 팩스 문서의 크기입니다.
get_ElapsedTime 부모 FaxPort 개체의 FaxStatus 개체에 대한 ElapsedTime 속성을 검색합니다. ElapsedTime 속성은 활성 팩스 작업의 경과 시간을 나타내는 숫자입니다.
get_Email IFaxSender::get_Email 속성은 보낸 사람과 연결된 전자 메일 주소를 포함하는 null로 끝나는 문자열입니다. (Get)
get_EmailAddress FaxDoc 개체의 EmailAddress 속성을 설정하거나 검색합니다. EmailAddress 속성은 팩스 전송 보낸 사람의 이메일 주소를 포함하는 null로 끝나는 문자열입니다. (Get)
get_Enable IFaxRoutingMethod::get_Enable 속성은 특정 팩스 포트에서 팩스 라우팅 방법을 사용할 수 있는지 여부를 나타내는 부울 값입니다. (Get)
get_EventLogging EventLogging 속성은 FaxEventLogging 구성 개체를 검색합니다.
get_ExtendedStatus ExtendedStatus 속성은 작업의 확장 상태를 설명하는 null로 끝나는 문자열입니다. (IFaxIncomingJob.get_ExtendedStatus)
get_ExtendedStatus ExtendedStatus 속성은 작업의 확장 상태를 설명하는 null로 끝나는 문자열입니다. (IFaxJobStatus.get_ExtendedStatus)

get_ExtendedStatus IFaxOutgoingJob::get_ExtendedStatus 속성은 작업의 확장 상태를 설명하는 null로 끝나는 문자열입니다.
get_ExtendedStatusCode FaxIncomingJob 개체의 ExtendedStatusCode 속성을 검색합니다. ExtendedStatusCode 속성은 작업의 확장 상태를 설명하는 코드를 지정합니다.
get_ExtendedStatusCode ExtendedStatusCode 속성은 작업의 확장 상태를 설명하는 코드를 지정합니다.
get_ExtendedStatusCode IFaxOutgoingJob::get_ExtendedStatusCode 속성은 작업의 확장 상태를 설명하는 코드를 지정합니다.
get_ExtensionFriendlyName IFaxInboundRoutingMethod::get_ExtensionFriendlyName 속성은 인바운드 팩스 라우팅 방법을 내보내는 팩스 라우팅 확장의 사용자에게 친숙한 이름입니다.
get_ExtensionImageName IFaxInboundRoutingMethod::get_ExtensionImageName 속성은 팩스 라우팅 메서드를 내보내는 팩스 라우팅 확장 프로그램의 실행 가능한 이미지 이름(DLL 경로 및 파일 이름)을 포함하는 null로 끝나는 문자열입니다.
get_ExtensionName IFaxRoutingMethod::get_ExtensionName 속성은 지정된 팩스 라우팅 방법을 구현하는 팩스 라우팅 확장 DLL의 사용자 친화적인 이름을 포함하는 null로 끝나는 문자열입니다.
get_FaxAccountSet IFaxServer2::FaxAccountSet 속성은 팩스 서버에서 팩스 계정을 관리하는 데 사용되는 IFaxAccountSet 개체를 반환합니다.
get_FaxNumber IFaxRecipient::get_FaxNumber 속성은 받는 사람과 연결된 팩스 번호를 포함하는 null로 끝나는 문자열입니다. (Get)

get_FaxNumber IFaxSender::get_FaxNumber 속성은 보낸 사람과 연결된 팩스 번호를 포함하는 null로 끝나는 문자열입니다. (Get)
get_FaxNumber FaxDoc 개체의 FaxNumber 속성을 설정하거나 검색합니다. FaxNumber 속성은 팩스 서버에서 팩스 전송을 보낼 팩스 번호를 포함하는 null로 끝나는 문자열입니다. (Get)
get_FaxNumber IFaxJob::get_FaxNumber 속성은 팩스 서버에서 팩스 작업을 전송할 팩스 번호를 포함하는 null로 끝나는 문자열입니다. IFaxJob::get_FaxNumber 속성은 송신 팩스 전송에만 적용됩니다.
get_FileName FaxDoc 개체의 FileName 속성을 설정하거나 검색합니다. FileName 속성은 개체와 연결된 문서 파일의 이름을 포함하는 null로 끝나는 문자열입니다. (Get)
get_Folders 들어오고 나가는 보관 파일 및 들어오고 나가는 큐를 포함하여 계정의 폴더를 나타냅니다.
get_Folders IFaxServer::get_Folders 속성은 IFaxFolders 구성 인터페이스에 액세스합니다. 인터페이스를 사용하여 연결된 팩스 서버의 폴더, 작업 및 메시지에 액세스할 수 있습니다.
get_FriendlyName IFaxDeviceProvider::get_FriendlyName 속성은 FSP(팩스 서비스 공급자)의 사용자에게 친숙한 이름을 포함하는 null로 끝나는 문자열입니다. 이 문자열은 사용자에게 표시하기에 적합합니다.
get_FriendlyName IFaxInboundRoutingExtension::get_FriendlyName 속성은 팩스 라우팅 확장 프로그램에 대한 사용자에게 친숙한 이름을 포함하는 null로 끝나는 문자열입니다. 이 문자열은 사용자에게 표시하기에 적합합니다.

get_FriendlyName IFaxRoutingMethod::get_FriendlyName 속성은 팩스 라우팅 메서드의 사용자 친화적인 이름을 포함하는 null로 끝나는 문자열입니다.
get_FunctionName IFaxInboundRoutingMethod::get_FunctionName 속성은 특정 팩스 라우팅 프로시저를 실행하는 함수의 이름을 포함하는 null로 끝나는 문자열입니다.
get_FunctionName IFaxRoutingMethod::get_FunctionName 속성은 특정 팩스 라우팅 프로시저를 실행하는 함수의 이름을 포함하는 null로 끝나는 문자열입니다.
get_GeneralEventsLevel IFaxEventLogging::get_GeneralEventsLevel 속성은 팩스 서비스가 애플리케이션 로그의 일반 이벤트를 기록하는 세부 정보 수준을 나타냅니다. (Get)
get_GrantedRights IFaxSecurity::get_GrantedRights 속성은 이 속성을 참조하는 사용자에게 부여된 팩스 서버 액세스 권한의 조합입니다.
get_GrantedRights 이 속성을 참조하는 사용자에게 부여된 팩스 서버 액세스 권한의 조합을 검색합니다.
get_GroupBroadcastReceipts IFaxDocument::get_GroupBroadcastReceipts 속성은 브로드캐스트의 각 받는 사람에 대해 개별 배달 영수증을 보낼지 또는 모든 받는 사람에 대한 요약 영수증을 보낼지 여부를 나타내는 부울 값입니다. (Get)
get_GroupBroadcastReceipts IFaxOutgoingJob::get_GroupBroadcastReceipts 속성은 브로드캐스트의 각 받는 사람에 대해 개별 배달 영수증을 보낼지 또는 모든 받는 사람에 대한 요약 영수증을 보낼지 여부를 나타내는 부울 값입니다.
get_GroupName IFaxOutboundRoutingRule::get_GroupName 속성은 아웃바운드 라우팅 규칙이 팩스 디바이스 그룹을 가리키는 경우 그룹 이름을 지정합니다. (Get)

get_Guid IFaxRoutingMethod::get_Guid 속성은 팩스 라우팅 방법을 고유하게 식별하는 GUID를 포함하는 null로 끝나는 문자열입니다.
get_GUID IFaxInboundRoutingMethod::get_GUID 속성은 팩스 라우팅 메서드를 고유하게 식별하는 GUID를 지정하는 null로 끝나는 문자열입니다.
get_HasCoverPage 팩스에 표지가 있는지 여부를 나타내는 플래그입니다. (Get)
get_HasCoverPage 팩스에 표지가 있는지를 지정합니다.
get_HasCoverPage 팩스에 표지가 있는지를 나타냅니다.
get_HighQuotaWaterMark 워터마크의 최대 할당 크기를 나타내는 값을 설정하거나 검색합니다. (Get)
get_HighQuotaWaterMark HighQuotaWaterMark 속성은 인바운드 팩스 메시지 보관의 크기에 대한 상위 경고 임계값(MB)을 지정하는 값입니다. (Get)
get_HighQuotaWaterMark IFaxOutgoingArchive::get_HighQuotaWaterMark 속성은 인바운드 팩스 메시지 보관의 크기에 대한 상한 임계값(MB)을 지정하는 값입니다. (Get)
get_HomePhone IFaxSender::get_HomePhone 속성은 보낸 사람과 연결된 집 전화 번호를 포함하는 null로 끝나는 문자열입니다. (Get)
get_Id ID IFaxDevice::get_Id 팩스 디바이스를 고유하게 식별하는 숫자 값입니다.
get_Id Id 속성은 인바운드 팩스 작업에 대한 고유 ID를 포함하는 null로 끝나는 문자열입니다.

get_Id Id 속성은 인바운드 팩스 메시지에 대한 고유 ID를 포함하는 null로 끝나는 문자열입니다.
get_Id IFaxOutgoingJob::get_Id 속성은 아웃바운드 팩스 작업에 대한 고유 식별자를 포함하는 null로 끝나는 문자열입니다. 작업이 성공적으로 완료된 후 식별자를 사용하여 보관된 팩스 메시지를 검색할 수 있습니다.
get_Id IFaxOutgoingMessage::get_Id 속성은 아웃바운드 팩스 메시지에 대한 고유 식별자를 포함하는 null로 끝나는 문자열입니다.
get_Image FaxTiff 개체의 Image 속성을 설정하거나 검색합니다. (Get)
get_ImageName IFaxDeviceProvider::get_ImageName 속성은 FSP(팩스 서비스 공급자)의 실행 파일 이미지 이름(DLL 경로 및 파일 이름)을 포함하는 null로 끝나는 문자열입니다.
get_ImageName IFaxInboundRoutingExtension::get_ImageName 속성은 팩스 라우팅 확장 프로그램의 실행 파일 이미지 이름(DLL 경로 및 파일 이름)을 포함하는 null로 끝나는 문자열입니다.
get_ImageName IFaxRoutingMethod::get_ImageName 속성은 팩스 라우팅 메서드를 구현하는 팩스 라우팅 확장 DLL의 실행 파일 이미지 이름을 포함하는 null로 끝나는 문자열입니다.
get_InboundEventsLevel IFaxEventLogging::get_InboundEventsLevel 속성은 팩스 서비스가 애플리케이션 로그의 인바운드 팩스 전송에 대한 이벤트를 기록하는 세부 수준을 나타냅니다. (Get)
get_InboundRouting IFaxServer::get_InboundRouting 속성은 IFaxInboundRouting 구성 인터페이스를 만듭니다. 인터페이스는 인바운드 팩스 라우팅 확장 및 해당 메서드에 대한 액세스를 허용합니다.

get_IncomingArchive 특정 팩스 계정에 대해 들어오는 팩스의 보관 파일을 나타냅니다. 수신된 팩스입니다.
get_IncomingArchive IFaxFolders::get_IncomingArchive 속성은 들어오는 팩스의 보관 파일을 나타냅니다.
get_IncomingFaxesArePublic 들어오는 팩스를 모든 사람이 볼 수 있는지 또는 비공개로 볼 수 있는지 여부를 나타냅니다. (Get)
get_IncomingMessages IFaxActivity::get_IncomingMessages 속성은 팩스 서비스가 현재 수신 중인 수신 팩스 작업의 총 수를 나타내는 숫자입니다.
get_IncomingQueue 특정 팩스 계정에 대해 들어오는 팩스의 큐를 나타냅니다. 아직 완전히 처리되지 않은 수신 팩스입니다.
get_IncomingQueue IFaxFolders::get_IncomingQueue 속성은 들어오는 팩스의 큐를 나타냅니다.
get_IncomingQueueBlocked 들어오는 팩스의 팩스 서버 큐가 차단되었는지 여부를 나타내는 값을 설정하거나 검색합니다. (Get)
get_InformationType IFaxSecurity::InformationType 속성은 보안 정보 유형을 나타냅니다. (Get)
get_InformationType 보안 정보 유형을 검색합니다. (Get)
get_InitErrorCode IFaxDeviceProvider::get_InitErrorCode 속성은 팩스 서비스가 FSPÂ DLL을 로드하고 초기화하는 동안 FSP(팩스 서비스 공급자)가 반환한 마지막 오류 코드를 지정하는 값입니다.

get_InitErrorCode IFaxInboundRoutingExtension::get_InitErrorCode 속성은 팩스 서비스가 팩스 라우팅 확장의 DLL을 로드하고 초기화하는 동안 팩스 라우팅 확장이 반환한 마지막 오류 코드를 지정하는 값입니다.
get_InitEventsLevel IFaxEventLogging::get_InitEventsLevel 속성은 팩스 서비스가 애플리케이션 로그에서 초기화(서버 시작) 및 종료(서버 종료) 이벤트를 기록하는 세부 수준을 나타냅니다. (Get)
get_Item FaxAccounts 컬렉션에서 FaxAccount 개체를 반환합니다.
get_Item IFaxDeviceIds::get_Item 메서드는 FaxDeviceIds 컬렉션의 디바이스 ID를 나타냅니다.
get_Item IFaxDeviceProviders::get_Item 속성은 FaxDeviceProviders 컬렉션에서 FaxDeviceProvider 개체를 반환합니다.
get_Item IFaxDevices::get_Item 메서드는 인덱스를 사용하여 FaxDevices 컬렉션에서 FaxDevice 개체를 반환합니다.
get_Item IFaxInboundRoutingExtensions::get_Item 메서드는 IFaxInboundRoutingExtensions 컬렉션에서 IFaxInboundRoutingExtension 인터페이스를 반환합니다.
get_Item IFaxInboundRoutingMethods::get_Item 메서드는 IFaxInboundRoutingMethods 컬렉션에서 IFaxInboundRoutingMethod 개체를 반환합니다.
get_Item FaxIncomingJobs 컬렉션에서 FaxIncomingJob 개체를 검색합니다.

get_Item IFaxOutboundRoutingGroups::get_Item 메서드는 컬렉션에서 IFaxOutboundRoutingGroup 인터페이스를 반환합니다.
get_Item IFaxOutboundRoutingRules::get_Item 메서드는 라우팅 규칙의 인덱스를 사용하여 IFaxOutboundRoutingRules 인터페이스에서 IFaxOutboundRoutingRule 인터페이스를 반환합니다.
get_Item IFaxOutgoingJobs::get_Item 메서드는 IFaxOutgoingJobs 인터페이스에서 IFaxOutgoingJob 인터페이스를 반환합니다.
get_Item Item 메서드는 FaxRecipients 컬렉션에서 FaxRecipient 개체를 반환합니다.
get_Item IFaxJobs::get_Item 메서드는 지정된 팩스 작업에 대한 새 FaxJob 개체를 반환합니다. 이 개체를 사용하면 연결된 팩스 서버와 연결된 팩스 작업을 열거할 수 있습니다.
get_Item IFaxPorts::get_Item 메서드는 지정된 팩스 포트에 대한 팩스 포트 개체를 만듭니다. 이 개체를 사용하면 팩스 서버에 대한 특정 연결에 대한 포트 구성 정보를 열거할 수 있습니다.
get_Item IFaxRoutingMethods::get_Item 메서드는 지정된 팩스 라우팅 메서드에 대한 FaxRoutingMethod 개체를 만듭니다. 이 개체를 사용하면 지정된 FaxPort 개체에 대한 팩스 라우팅 정보를 열거할 수 있습니다.
get_ItemById IFaxDevices::get_ItemById 메서드는 디바이스 ID를 사용하여 FaxDevices 컬렉션에서 FaxDevice 개체를 반환합니다.
get_JobId IFaxJob::get_JobId 속성은 지정된 팩스 작업을 고유하게 식별하는 숫자입니다.

get_JobType FaxIncomingJob 개체의 JobType 속성을 검색합니다. JobType 속성은 팩스 작업의 유형을 설명합니다. 예를 들어 작업은 수신 작업, 송신 작업 또는 라우팅 작업일 수 있습니다.
get_JobType JobType 속성은 팩스 작업의 유형을 설명합니다. 예를 들어 작업은 수신 작업, 송신 작업 또는 라우팅 작업일 수 있습니다.
get_LoggingOptions IFaxServer::get_LoggingOptions 속성은 IFaxLoggingOptions 구성 인터페이스를 만듭니다. 인터페이스는 활동 로깅 옵션과 팩스 서비스에서 사용하는 이벤트 로깅 범주를 모두 구성할 수 있습니다.
get_LogIncoming IFaxActivityLogging::get_LogIncoming 속성은 팩스 서비스가 활동 로그 데이터베이스에서 들어오는 팩스의 항목을 기록하는지 여부를 나타내는 부울 값입니다. (Get)
get_LogOutgoing IFaxActivityLogging::get_LogOutgoing 속성은 팩스 서비스가 활동 로그 데이터베이스에서 보내는 팩스에 대한 항목을 기록하는지 여부를 나타내는 부울 값입니다. (Get)
get_LowQuotaWaterMark 워터마크의 최소 크기를 나타내는 값을 설정하거나 검색합니다. (Get)
get_LowQuotaWaterMark LowQuotaWaterMark 속성은 인바운드 팩스 메시지 보관에 대한 낮은 경고 임계값(MB)을 지정하는 값입니다. (Get)
get_LowQuotaWaterMark IFaxOutgoingArchive::get_LowQuotaWaterMark 속성은 아웃바운드 팩스 메시지 보관에 대한 낮은 임계값(MB)을 지정하는 값입니다. (Get)
get_MajorBuild IFaxDeviceProvider::get_MajorBuild 속성은 FSP(팩스 서비스 공급자)에 대한 빌드 번호의 주요 부분을 지정하는 값입니다.

get_MajorBuild IFaxInboundRoutingExtension::get_MajorBuild 속성은 팩스 라우팅 확장의 DLL에 대한 빌드 번호의 주요 부분을 지정하는 값입니다.
get_MajorBuild IFaxServer::get_MajorBuild 속성은 팩스 서비스에 대한 빌드 번호의 주요 부분을 지정하는 값입니다.
get_MajorVersion IFaxDeviceProvider::get_MajorVersion 속성은 FSP(팩스 서비스 공급자)에 대한 버전 번호의 주요 부분을 지정하는 값입니다.
get_MajorVersion IFaxInboundRoutingExtension::get_MajorVersion 속성은 팩스 라우팅 확장의 DLL에 대한 버전 번호의 주요 부분을 지정하는 값입니다.
get_MajorVersion IFaxServer::get_MajorVersion 속성은 팩스 서비스에 대한 버전 번호의 주요 부분을 지정하는 값입니다.
get_Message Message 속성은 보관 커서 아래의 인바운드 팩스 메시지를 검색합니다.
get_Message IFaxOutgoingMessageIterator::get_Message 속성은 보관 커서 아래에서 아웃바운드 팩스 메시지를 검색합니다.
get_Methods IFaxInboundRoutingExtension::get_Methods 속성은 팩스 라우팅 확장에 의해 노출되는 인바운드 라우팅 메서드를 고유하게 식별하는 GUID 배열입니다.
get_MinorBuild IFaxDeviceProvider::get_MinorBuild 속성은 FSP(팩스 서비스 공급자)에 대한 빌드 번호의 부 부분을 지정하는 값입니다.
get_MinorBuild IFaxInboundRoutingExtension::get_MinorBuild 속성은 팩스 라우팅 확장의 DLL에 대한 빌드 번호의 부 부분을 지정하는 값입니다.

get_MinorBuild IFaxServer::get_MinorBuild 속성은 팩스 서비스에 대한 빌드 번호의 부 부분을 지정하는 값입니다.
get_MinorVersion IFaxDeviceProvider::get_MinorVersion 속성은 FSP(팩스 서비스 공급자)에 대한 버전 번호의 부 부분을 지정하는 값입니다.
get_MinorVersion IFaxInboundRoutingExtension::get_MinorVersion 속성은 팩스 라우팅 확장의 DLL에 대한 버전 번호의 부 부분을 지정하는 값입니다.
get_MinorVersion IFaxServer::get_MinorVersion 속성은 팩스 서비스에 대한 버전 번호의 부 부분을 지정하는 값입니다.
get_Name IFaxInboundRoutingMethod::get_Name 속성은 인바운드 팩스 라우팅 메서드와 연결된 사용자 친화적인 이름을 포함하는 null로 끝나는 문자열입니다. 이 문자열은 사용자에게 표시하기에 적합합니다.
get_Name Name 속성은 아웃바운드 라우팅 그룹의 이름을 지정하는 null로 끝나는 문자열입니다.
get_Name IFaxRecipient::get_Name 속성은 받는 사람의 이름을 포함하는 null로 끝나는 문자열입니다. (Get)
get_Name IFaxSender::get_Name 속성은 보낸 사람의 이름을 포함하는 null로 끝나는 문자열입니다. (Get)
get_Name IFaxPort::get_Name 속성은 팩스 포트에 대한 사용자에게 친숙한 표시 이름을 포함하는 null로 끝나는 문자열입니다.

get_Note IFaxDocument::get_Note 속성은 팩스의 표지에 있는 메모 필드의 내용을 포함하는 null로 끝나는 문자열입니다. (Get)
get_OfficeLocation IFaxSender::get_OfficeLocation 속성은 보낸 사람의 사무실 위치를 포함하는 null로 끝나는 문자열입니다. (Get)
get_OfficePhone IFaxSender::get_OfficePhone 속성은 보낸 사람과 연결된 사무실 전화 번호를 포함하는 null로 끝나는 문자열입니다. (Get)
get_OriginalScheduledTime IFaxOutgoingJob::get_OriginalScheduledTime 속성은 팩스 작업이 원래 전송되도록 예약된 시간을 지정합니다.
get_OriginalScheduledTime IFaxOutgoingMessage::get_OriginalScheduledTime 속성은 팩스 메시지가 원래 전송되도록 예약된 시간을 지정합니다.
get_OutboundEventsLevel IFaxEventLogging::get_OutboundEventsLevel 속성은 팩스 서비스가 애플리케이션 로그의 아웃바운드 팩스 전송에 대한 이벤트를 기록하는 세부 수준을 나타냅니다. (Get)
get_OutboundRouting IFaxServer::get_OutboundRouting 속성은 IFaxOutboundRouting 구성 인터페이스를 만듭니다. 인터페이스를 사용하면 사용자가 아웃바운드 라우팅 그룹 및 규칙을 구성할 수 있습니다.
get_OutgoingArchive 전송된 특정 팩스 계정에 대한 송신 팩스의 보관 파일을 나타냅니다.
get_OutgoingArchive IFaxFolders::get_OutgoingArchive 속성은 보내는 팩스의 보관 파일을 나타냅니다.
get_OutgoingMessages IFaxActivity::get_OutgoingMessages 속성은 팩스 서비스가 보내는 동안 보내는 팩스 작업의 총 수를 나타내는 숫자입니다.

get_OutgoingQueue 특정 팩스 계정에 대해 보내는 팩스의 큐를 나타냅니다. 아직 전송되지 않은 팩스입니다.
get_OutgoingQueue IFaxFolders::get_OutgoingQueue 속성은 보내는 팩스의 큐를 나타냅니다.
get_OutgoingQueueAgeLimit 배달할 수 없는 팩스 메시지가 삭제되기 전에 팩스 서버에 보관되는 시간을 나타내는 값을 설정하거나 검색합니다. (Get)
get_OutgoingQueueBlocked 보내는 팩스의 팩스 서버 큐가 차단되었는지 여부를 나타내는 값을 설정하거나 검색합니다. (Get)
get_OutgoingQueuePaused 나가는 큐가 일시 중지되었는지 여부를 나타내는 값을 설정하거나 검색합니다. (Get)
get_PageCount IFaxJob::get_PageCount 속성은 팩스 전송의 총 페이지 수를 나타내는 숫자입니다. IFaxJob::get_PageCount 속성은 보내는 팩스 전송에만 적용됩니다.
get_PageCount 부모 FaxPort 개체의 FaxStatus 개체에 대한 PageCount 속성을 검색합니다. PageCount 속성은 아웃바운드 팩스 전송의 총 페이지 수를 나타냅니다.
get_Pages Pages 속성은 인바운드 팩스 메시지의 총 페이지 수를 나타내는 값입니다.
get_Pages Pages 속성은 팩스 전송에서 지금까지 받은 총 페이지 수를 나타내는 숫자입니다.
get_Pages IFaxOutgoingJob::get_Pages 속성은 아웃바운드 팩스 작업의 총 페이지 수를 나타내는 숫자입니다.

get_Pages IFaxOutgoingMessage::get_Pages 속성은 아웃바운드 팩스 메시지의 총 페이지 수를 나타내는 숫자입니다.
get_Paused IFaxOutgoingQueue::get_Paused 속성은 보내는 팩스의 작업 큐가 일시 중지되었는지 여부를 나타내는 부울 값입니다. (Get)
get_PauseServerQueue FaxServer 개체에 대한 PauseServerQueue 속성을 설정하거나 검색합니다. PauseServerQueue 속성은 팩스 서버에서 팩스 작업 큐를 일시 중지했는지 여부를 나타내는 부울 값입니다. (Get)
get_PoweredOff IFaxDevice::get_PoweredOff 속성은 팩스 디바이스가 현재 팩스를 보내고 받을 수 있는지 여부를 나타내는 부울 값입니다.
get_PrefetchSize PrefetchSize 속성은 프리페치(미리 읽기) 버퍼의 크기를 나타냅니다. (Get)
get_PrefetchSize IFaxOutgoingMessageIterator::get_PrefetchSize 속성은 프리페치(미리 읽기) 버퍼의 크기를 나타냅니다. 그러면 개체가 콘텐츠를 새로 고쳐야 할 때 반복기 개체가 팩스 서버에서 검색하는 팩스 메시지 수를 결정합니다. (Get)
get_Priority IFaxDocument::get_Priority 속성은 팩스를 보낼 때 사용할 우선 순위를 지정합니다. 예를 들어 보통, 낮음 또는 높은 우선 순위입니다. (Get)
get_Priority Priority 속성은 서비스가 팩스 작업을 받을 때 팩스 서비스가 라우팅 메서드를 호출하는 순서와 연결된 값입니다. (Get)
get_Priority IFaxOutgoingJob::get_Priority 속성은 팩스를 보낼 때 사용할 우선 순위를 지정합니다. 예를 들어 보통, 낮음 또는 높은 우선 순위입니다.

get_Priority IFaxOutgoingMessage::get_Priority 속성은 팩스를 보낼 때 사용되는 우선 순위를 지정합니다. 예를 들어 보통, 낮음 또는 높은 우선 순위입니다.
get_Priority IFaxPort::get_Priority 속성은 지정된 팩스 포트에 지정된 전송 우선 순위를 나타내는 숫자입니다. 우선 순위는 사용 가능한 팩스 장치가 송신 전송을 보내는 상대 순서를 결정합니다. (Get)
get_ProviderUniqueName IFaxDevice::get_ProviderUniqueName 속성은 디바이스와 연결된 팩스 서비스 공급자(FSP)의 고유한 이름을 포함하는 null로 끝나는 문자열입니다.
get_QueuedMessages IFaxActivity::get_QueuedMessages 속성은 처리 보류 중인 팩스 작업 큐의 총 팩스 작업 수를 나타내는 숫자입니다. 재시도 횟수가 초과된 작업은 포함되지 않습니다.
get_QueueStatus IFaxJob::get_QueueStatus 속성은 팩스 작업의 작업 큐 상태를 설명하는 null로 종료된 문자열입니다.
get_RawReceiveTime FaxTiff 개체의 RawReceiveTime 속성을 검색합니다.
get_Read 팩스를 읽었는지를 나타내는 플래그입니다. (Get)
get_Read 팩스를 읽었는지를 나타냅니다. (Get)
get_ReceiptAddress IFaxDocument::get_ReceiptAddress 속성은 팩스 작업이 최종 상태에 도달할 때 팩스 서비스가 배달 영수증을 보내야 하는 전자 메일 주소를 나타내는 null로 종료된 문자열입니다. (Get)
get_ReceiptAddress 배달 보고서를 보낼 주소를 포함하는 null로 종료된 문자열로, 성공 또는 실패를 나타냅니다.

get_ReceiptAddress 배달 보고서를 보낼 주소를 지정합니다.
get_ReceiptOptions IFaxServer::get_ReceiptOptions 속성은 IFaxReceiptOptions 구성 인터페이스를 만듭니다. 이 개체는 팩스 클라이언트 애플리케이션이 팩스 서비스가 팩스 영수증을 보내는 데 사용하는 영수증 구성을 설정하고 검색할 수 있도록 허용합니다.
get_ReceiptType IFaxDocument::get_ReceiptType 속성은 팩스 작업이 최종 상태에 도달할 때 배달할 배달 영수증의 유형을 지정합니다. (Get)
get_ReceiptType IFaxOutgoingJob::get_ReceiptType 속성은 팩스 메시지가 최종 상태에 도달할 때 배달할 배달 확인 유형을 지정하는 값입니다. 영수증 유형은 SMTP(Simple Mail Transport Protocol) 메일, 메시지 상자 또는 영수증이 될 수 없습니다.
get_ReceiptType 전송 시도 후 전송되는 배달 보고서의 유형을 지정합니다.
get_Receive IFaxPort::get_Receive 속성은 지정된 팩스 포트에서 팩스를 받을 수 있는지 여부를 나타내는 부울 값입니다. (Get)
get_Receive 부모 FaxPort 개체의 FaxStatus 개체에 대한 Receive 속성을 검색합니다. Receive 속성은 지정된 팩스 포트가 현재 팩스를 받고 있는지 여부를 나타내는 부울 값입니다.
get_ReceiveMode ReceiveMode 속성은 디바이스가 들어오는 호출에 응답하는 방법을 정의하는 FAX_DEVICE_RECEIVE_MODE_ENUM 열거형의 값입니다. (Get)
get_ReceiveTime FaxTiff 개체의 ReceiveTime 속성을 검색합니다.
get_ReceivingNow IFaxDevice::get_ReceivingNow 속성은 팩스 디바이스가 속성을 검색할 때 팩스를 수신하는지 여부를 나타내는 부울 값입니다(상태는 그 후 즉시 변경될 수 있음).

get_Recipient IFaxOutgoingJob::get_Recipient 속성은 팩스 작업의 받는 사람에 대한 정보를 포함하는 개체에 대한 인터페이스를 검색합니다.
get_Recipient IFaxOutgoingMessage::get_Recipient 속성은 팩스 메시지 수신자에 대한 정보가 포함된 인터페이스를 검색합니다.
get_RecipientAddress FaxDoc 개체의 RecipientAddress 속성을 설정하거나 검색합니다. RecipientAddress 속성은 팩스 전송 수신자의 주소가 포함된 null로 끝나는 문자열입니다. (Get)
get_RecipientCity FaxDoc 개체의 RecipientCity 속성을 설정하거나 검색합니다. RecipientCity 속성은 팩스 전송 수신자의 도시 이름을 포함하는 null로 끝나는 문자열입니다. (Get)
get_RecipientCompany FaxDoc 개체의 RecipientCompany 속성을 설정하거나 검색합니다. RecipientCompany 속성은 팩스 전송 수신자의 회사 이름을 포함하는 null로 종료된 문자열입니다. (Get)
get_RecipientCountry FaxDoc 개체의 RecipientCountry 속성을 설정하거나 검색합니다. RecipientCountry 속성은 팩스 전송 수신자의 국가/지역을 포함하는 null로 끝나는 문자열입니다. (Get)
get_RecipientDepartment FaxDoc 개체의 RecipientDepartment 속성을 설정하거나 검색합니다. RecipientDepartment 속성은 팩스 전송 수신자의 부서를 포함하는 null로 끝나는 문자열입니다. (Get)
get_RecipientHomePhone FaxDoc 개체의 RecipientHomePhone 속성을 설정하거나 검색합니다. RecipientHomePhone 속성은 팩스 전송 수신자의 집 전화 번호를 포함하는 null로 끝나는 문자열입니다. (Get)

get_RecipientName FaxDoc 개체의 RecipientName 속성을 설정하거나 검색합니다. RecipientName 속성은 팩스 전송 수신자의 이름을 포함하는 null로 끝나는 문자열입니다. (Get)
get_RecipientName IFaxJob::get_RecipientName 속성은 팩스 작업의 받는 사람의 이름을 포함하는 null로 끝나는 문자열입니다.
get_RecipientName FaxStatus 개체의 RecipientName 속성을 검색합니다. RecipientName 속성은 인바운드 팩스 전송 수신자의 이름을 포함하는 null로 끝나는 문자열입니다.
get_RecipientName FaxTiff 개체의 RecipientName 속성을 검색합니다. RecipientName 속성은 지정된 팩스 파일의 받는 사람 이름을 포함하는 null로 끝나는 문자열입니다.
get_RecipientNumber FaxTiff 개체의 RecipientNumber 속성을 검색합니다. RecipientNumber 속성은 팩스가 전송된 팩스 번호를 포함하는 null로 끝나는 문자열입니다.
get_RecipientOffice FaxDoc 개체의 RecipientOffice 속성을 설정하거나 검색합니다. RecipientOffice 속성은 팩스 전송 수신자의 사무실이 포함된 null로 종료된 문자열입니다. (Get)
get_RecipientOfficePhone FaxDoc 개체의 RecipientOfficePhone 속성을 설정하거나 검색합니다. RecipientOfficePhone 속성은 팩스 전송 수신자의 사무실 전화 번호를 포함하는 null로 종료된 문자열입니다. (Get)
get_Recipients IFaxDocument::get_Recipients 속성은 팩스 문서에 대한 하나 이상의 받는 사람 컬렉션을 검색합니다.
get_Recipients 인바운드 팩스 메시지와 연결된 받는 사람을 포함합니다. 이 속성은 null로 끝나는 문자열입니다. (Get)

get_RecipientState FaxDoc 개체의 RecipientState 속성을 설정하거나 검색합니다. RecipientState 속성은 팩스 전송 수신자의 상태를 포함하는 null로 끝나는 문자열입니다. (Get)
get_RecipientTitle FaxDoc 개체의 RecipientTitle 속성을 설정하거나 검색합니다. RecipientTitle 속성은 팩스 전송 수신자의 제목을 포함하는 null로 끝나는 문자열입니다. (Get)
get_RecipientZip FaxDoc 개체의 RecipientZip 속성을 설정하거나 검색합니다. RecipientZip 속성은 팩스 전송 수신자의 우편 번호를 포함하는 null로 끝나는 문자열입니다. (Get)
get_RegisteredEvents 계정이 수신 대기하는 이벤트의 유형을 나타내는 플래그 집합입니다.
get_RegisteredEvents IFaxServer::get_RegisteredEvents 속성은 클라이언트 애플리케이션이 수신 대기 중인 팩스 서비스 이벤트의 유형을 나타내는 열거형의 값입니다.
get_Retries 지정된 팩스 작업에 대한 다시 시도 횟수를 나타내는 값을 설정하거나 검색합니다. (Get)
get_Retries Retries 속성은 초기 라우팅 시도가 실패했을 때 팩스 서비스가 들어오는 팩스를 라우팅하려고 시도한 횟수를 나타내는 값입니다.
get_Retries Retries 속성은 초기 라우팅 시도가 실패한 후 팩스 서비스가 인바운드 팩스 메시지를 라우팅하려고 시도한 횟수를 나타내는 값입니다.
get_Retries Retries 속성은 초기 시도가 실패했을 때 팩스 서비스가 팩스 작업을 전송하려고 시도한 횟수를 나타내는 값입니다.
get_Retries IFaxOutgoingJob::get_Retries 속성은 초기 전송 시도가 실패한 후 팩스 서비스가 송신 팩스를 전송하려고 시도한 횟수를 나타내는 값입니다.

get_Retries IFaxOutgoingMessage::get_Retries 속성은 초기 전송 시도가 실패한 후 팩스 서비스가 송신 팩스를 전송하려고 시도한 횟수를 나타내는 값입니다.
get_Retries IFaxOutgoingQueue::get_Retries 속성은 초기 전송이 실패할 때 팩스 서비스가 송신 팩스를 재전송하려고 시도하는 횟수를 나타내는 값입니다. (Get)
get_Retries FaxServer 개체에 대한 Retries 속성을 설정하거나 검색합니다. Retries 속성은 초기 전송이 실패할 때 팩스 서버에서 송신 팩스를 다시 전송하려고 시도하는 횟수를 나타내는 값입니다. (Get)
get_RetryDelay 팩스 서비스가 실패한 팩스 전송을 다시 시도하기 전에 대기해야 하는 시간을 나타내는 값을 설정하거나 검색합니다. (Get)
get_RetryDelay IFaxOutgoingQueue::get_RetryDelay 속성은 팩스 서비스가 아웃바운드 팩스 작업을 다시 전송하기 전에 대기하는 시간 간격(분)을 나타내는 값입니다. (Get)
get_RetryDelay FaxServer 개체에 대한 RetryDelay 속성을 설정하거나 검색합니다. RetryDelay 속성은 시간 간격(분)을 나타내는 값으로, 팩스 서버는 아웃바운드 팩스 작업을 다시 전송하기 전에 대기합니다. (Get)
get_RingingNow IFaxDevice::get_RingingNow 속성은 속성이 검색되는 시점에 팩스 디바이스가 울리는지 여부를 나타내는 부울 값입니다(상태는 그 후 즉시 변경될 수 있음).
get_Rings IFaxPort::get_Rings 속성은 팩스 포트가 통화에 응답하기 전에 수신 팩스 호출이 대기해야 하는 링 수를 나타냅니다. (Get)
get_RingsBeforeAnswer IFaxDevice::get_RingsBeforeAnswer 속성은 팩스 장치가 들어오는 팩스 호출에 응답하기 전에 발생하는 링 수를 지정하는 숫자입니다. (Get)

get_Routing FaxTiff 개체의 라우팅 속성을 검색합니다. 라우팅 속성은 지정된 팩스 파일에 대한 인바운드 라우팅 문자열을 포함하는 null로 끝나는 문자열입니다.
get_RoutingData IFaxRoutingMethod::get_RoutingData 속성은 들어오는 팩스 전송에 대한 라우팅 문자열을 포함하는 null로 끝나는 문자열입니다.
get_RoutingInformation RoutingInformation 속성은 인바운드 팩스 작업에 대한 라우팅 정보를 지정하는 null 종료 문자열입니다.
get_RoutingInformation RoutingInformation 속성은 팩스 메시지에 대한 인바운드 라우팅 정보를 나타내는 null로 끝나는 문자열입니다.
get_RoutingInformation RoutingInformation 속성은 팩스 작업에 대한 라우팅 정보를 지정하는 null로 끝나는 문자열입니다.
get_RoutingMessages IFaxActivity::get_RoutingMessages 속성은 팩스 서비스가 현재 라우팅하는 들어오는 팩스 작업의 총 수를 나타내는 숫자입니다.
get_RoutingString 부모 FaxPort 개체의 FaxStatus 개체에 대한 RoutingString 속성을 검색합니다. RoutingString 속성은 팩스 서비스 공급자와 관련된 인바운드 팩스 전송에 대한 라우팅 정보를 포함하는 null로 종료된 문자열입니다.
get_ScheduledTime ScheduledTime 속성은 팩스 작업이 전송되도록 예약된 시간을 나타냅니다.
get_ScheduledTime IFaxOutgoingJob::get_ScheduledTime 속성은 팩스 서비스에 처리하기 위해 팩스를 제출할 시간을 나타냅니다.
get_ScheduleTime IFaxDocument::get_ScheduleTime 속성은 팩스 서비스에 처리하기 위해 팩스를 제출하는 시간을 나타냅니다. (Get)

get_ScheduleType IFaxDocument::get_ScheduleType 속성은 팩스 작업을 예약할 시기를 나타냅니다. 예를 들어 팩스 서비스가 지정된 시간에 또는 미리 정의된 할인 기간 동안 즉시 팩스를 보내도록 지정할 수 있습니다. (Get)
get_ScheduleType 전송에 사용된 일정 유형을 지정합니다.
get_Security IFaxServer::get_Security 속성은 IFaxSecurity 구성 인터페이스를 만듭니다. 이 인터페이스를 사용하면 호출 애플리케이션이 팩스 서버에 대한 보안 설명자를 설정하고 검색할 수 있습니다.
get_Security2 IFaxServer2::Security2 속성은 팩스 서버에서 보안을 구성하는 데 사용되는 IFaxSecurity2 개체를 반환합니다.
get_Send IFaxPort::get_Send 속성은 팩스 포트에서 팩스를 보낼 수 있는지 여부를 나타내는 부울 값입니다. (Get)
get_Send 부모 FaxPort 개체의 FaxStatus 개체에 대한 Send 속성을 검색합니다. Send 속성은 지정된 팩스 포트가 현재 팩스를 보내고 있는지 여부를 나타내는 부울 값입니다.
get_SendCoverpage FaxDoc 개체의 SendCoverpage 속성을 설정하거나 검색합니다. SendCoverpage 속성은 지정된 표지 파일이 팩스 서버에 저장되는지 여부를 나타내는 부울 값입니다. (Get)
get_SendEnabled IFaxDevice::get_SendEnabled 속성은 팩스 디바이스에서 팩스를 보낼 수 있는지 여부를 나타내는 부울 값입니다. (Get)
get_Sender 팩스 문서의 보낸 사람 정보를 포함하는 개체를 검색합니다.

get_Sender IFaxOutgoingJob::get_Sender 속성은 팩스 보낸 사람 정보를 포함하는 개체를 검색합니다.
get_Sender IFaxOutgoingMessage::get_Sender 속성은 팩스 메시지의 보낸 사람 정보를 포함하는 인터페이스를 검색합니다.
get_SenderAddress FaxDoc 개체의 SenderAddress 속성을 설정하거나 검색합니다. SenderAddress 속성은 팩스 전송 발신자의 주소가 포함된 null로 끝나는 문자열입니다. (Get)
get_SenderCompany FaxDoc 개체의 SenderCompany 속성을 설정하거나 검색합니다. SenderCompany 속성은 팩스 전송 보낸 사람의 회사 이름을 포함하는 null로 끝나는 문자열입니다. (Get)
get_SenderCompany IFaxJob::get_SenderCompany 속성은 팩스 작업 보낸 사람의 회사 이름을 포함하는 null로 종료된 문자열입니다. IFaxJob::get_SenderCompany 속성은 보내는 팩스 전송에만 적용됩니다.
get_SenderDepartment FaxDoc 개체의 SenderDepartment 속성을 설정하거나 검색합니다. SenderDepartment 속성은 팩스 전송 보낸 사람의 부서를 포함하는 null로 끝나는 문자열입니다. (Get)
get_SenderDept IFaxJob::get_SenderDept 속성은 팩스 작업 보낸 사람의 부서 식별자를 포함하는 null로 종료된 문자열입니다. IFaxJob::get_SenderDept 속성은 보내는 팩스 전송에만 적용됩니다.
get_SenderFax FaxDoc 개체의 SenderFax 속성을 설정하거나 검색합니다. SenderFax 속성은 아웃바운드 팩스 전송 보낸 사람의 팩스 번호를 포함하는 null로 끝나는 문자열입니다. (Get)

get_SenderFaxNumber 인바운드 팩스 메시지와 연결된 보낸 사람의 팩스 번호를 포함합니다. 이 속성은 null로 끝나는 문자열입니다. (Get)
get_SenderHomePhone FaxDoc 개체의 SenderHomePhone 속성을 설정하거나 검색합니다. SenderHomePhone 속성은 팩스 전송 발신자의 집 전화 번호를 포함하는 null로 끝나는 문자열입니다. (Get)
get_SenderName 인바운드 팩스 메시지와 연결된 보낸 사람의 이름을 포함합니다. 이 속성은 null로 끝나는 문자열입니다. (Get)
get_SenderName FaxDoc 개체의 SenderName 속성을 설정하거나 검색합니다. SenderName 속성은 팩스 전송 보낸 사람의 이름을 포함하는 null로 끝나는 문자열입니다. (Get)
get_SenderName IFaxJob::get_SenderName 속성은 팩스 작업을 시작한 보낸 사람의 이름을 포함하는 null로 끝나는 문자열입니다. IFaxJob::get_SenderName 속성은 보내는 팩스 전송에만 적용됩니다.
get_SenderName 부모 FaxPort 개체의 FaxStatus 개체에 대한 SenderName 속성을 검색합니다. SenderName 속성은 팩스 전송을 보낸 사용자의 이름을 포함하는 null로 종료된 문자열입니다.
get_SenderName FaxTiff 개체의 SenderName 속성을 검색합니다. SenderName 속성은 팩스 전송을 대기한 사용자의 이름을 포함하는 null로 종료된 문자열입니다.
get_SenderOffice FaxDoc 개체의 SenderOffice 속성을 설정하거나 검색합니다. SenderOffice 속성은 팩스 전송 발신자의 사무실이 포함된 null로 종료된 문자열입니다. (Get)
get_SenderOfficePhone FaxDoc 개체의 SenderOfficePhone 속성을 설정하거나 검색합니다. SenderOfficePhone 속성은 팩스 전송 발신자의 사무실 전화 번호를 포함하는 null로 끝나는 문자열입니다. (Get)

get_SenderTitle FaxDoc 개체의 SenderTitle 속성을 설정하거나 검색합니다. SenderTitle 속성은 팩스 전송 발신자의 제목을 포함하는 null로 끝나는 문자열입니다. (Get)
get_SendingNow IFaxDevice::get_SendingNow 속성은 속성이 검색되는 시점에 팩스 디바이스가 팩스를 보내는지 여부를 나타내는 부울 값입니다(상태는 그 후 즉시 변경될 수 있음).
get_ServerCoverpage FaxDoc 개체의 ServerCoverpage 속성을 설정하거나 검색합니다. ServerCoverpage 속성은 지정된 표지 파일이 팩스 서버에 저장되는지 여부를 나타내는 부울 값입니다. (Get)
get_ServerCoverpage FaxServer 개체의 ServerCoverpage 속성을 설정하거나 검색합니다. ServerCoverpage 속성은 팩스 서버에서 일반 표지만 사용할 수 있는지 여부를 나타내는 부울 값입니다. (Get)
get_ServerMapiProfile FaxServer 개체의 ServerMapiProfile 속성을 설정하거나 검색합니다. ServerMapiProfile 속성은 팩스 서버가 들어오는 팩스 전송을 라우팅하는 데 사용하는 MAPI 사용자 프로필을 포함하는 null로 종료된 문자열입니다. (Get)
get_ServerName IFaxServer::get_ServerName 속성은 팩스 클라이언트가 연결된 활성 팩스 서버의 이름을 검색합니다.
get_Size Size 속성은 인바운드 팩스 작업과 연결된 TIFF 클래스 F(태그가 지정된 이미지 파일 형식 클래스 F) 파일의 크기를 나타내는 값입니다.
get_Size Size 속성은 인바운드 팩스 메시지와 연결된 TIFF 클래스 F(태그가 지정된 이미지 파일 형식 클래스 F) 파일의 크기를 나타내는 값입니다.
get_Size Size 속성은 팩스 작업에 대해 지금까지 받은 TIFF 클래스 F(태그가 지정된 이미지 파일 형식 클래스 F) 파일의 바이트 수를 나타내는 값입니다.

get_Size IFaxOutgoingJob::get_Size 속성은 아웃바운드 팩스 작업과 연결된 TIFF 클래스 F(태그가 지정된 이미지 파일 형식 클래스 F) 파일의 크기를 나타내는 값입니다.
get_Size Size 속성은 아웃바운드 팩스 메시지와 연결된 TIFF 클래스 F(태그가 지정된 이미지 파일 형식 클래스 F) 파일의 크기를 나타내는 값입니다.
get_SizeHigh 특정 팩스 계정에 대한 인바운드 팩스 메시지 보관의 크기에 대한 높은 32비트 값(바이트)을 지정합니다.
get_SizeHigh 특정 팩스 계정에 대한 아웃바운드 팩스 메시지 보관 파일의 크기(바이트)의 상위 32비트 값을 지정합니다.
get_SizeHigh SizeHigh 속성은 인바운드 팩스 메시지 보관의 크기에 대해 높은 32비트 값(바이트)을 지정하는 값입니다.
get_SizeHigh IFaxOutgoingArchive::get_SizeHigh 속성은 보내는 팩스 메시지의 보관 크기에 대해 높은 32비트 값(바이트)을 지정하는 값입니다.
get_SizeLow 특정 팩스 계정의 인바운드 팩스 메시지 보관 크기에 대한 낮은 32비트 값(바이트)을 지정합니다.
get_SizeLow 특정 팩스 계정에 대한 아웃바운드 팩스 메시지 보관 파일의 크기(바이트)의 하위 순서 32비트 값을 지정합니다.
get_SizeLow SizeLow 속성은 인바운드 팩스 메시지 보관의 크기에 대해 낮은 32비트 값(바이트)을 지정하는 값입니다.
get_SizeLow IFaxOutgoingArchive::get_SizeLow 속성은 보내는 팩스 메시지의 보관 크기에 대해 낮은 32비트 값(바이트)을 지정하는 값입니다.

get_SizeQuotaWarning 크기 할당량 경고가 켜져 있는지 여부를 나타내는 값을 설정하거나 검색합니다. (Get)
get_SizeQuotaWarning SizeQuotaWarning 속성은 인바운드 보관 파일의 크기가 HighQuotaWaterMark 속성에 정의된 제한을 초과할 때 팩스 서비스가 이벤트 로그에 경고를 발생시키는지 여부를 나타내는 부울 값입니다. (Get)
get_SizeQuotaWarning IFaxOutgoingArchive::get_SizeQuotaWarning 속성은 아웃바운드 보관 파일의 크기가 IFaxOutgoingArchive::get_HighQuotaWaterMark 속성에 정의된 제한을 초과할 때 팩스 서비스가 이벤트 로그에 경고를 발생시키는지 여부를 나타내는 부울 값입니다. (Get)
get_SMTPPassword IFaxReceiptOptions::get_SMTPPassword 속성은 인증된 연결에 사용되는 SMTP(Simple Mail Transport Protocol) 암호를 포함하는 null로 끝나는 문자열입니다. (Get)
get_SMTPPort IFaxReceiptOptions::get_SMTPPort 속성은 SMTP(Simple Mail Transport Protocol) 포트 번호를 지정하는 값입니다. (Get)
get_SMTPSender IFaxReceiptOptions::get_SMTPSender 속성은 메일 메시지 확인 보낸 사람의 SMTP(Simple Mail Transport Protocol) 전자 메일 주소를 포함하는 null로 끝나는 문자열입니다. (Get)
get_SMTPServer IFaxReceiptOptions::get_SMTPServer 속성은 SMTP(Simple Mail Transport Protocol) 서버의 이름을 포함하는 null로 끝나는 문자열입니다. (Get)
get_SMTPUser IFaxReceiptOptions::get_SMTPUser 속성은 인증된 연결에 사용되는 SMTP(Simple Mail Transport Protocol) 사용자 이름을 포함하는 null로 끝나는 문자열입니다. (Get)

get_StartTime 부모 FaxPort 개체의 FaxStatus 개체에 대한 StartTime 속성을 검색합니다. StartTime 속성은 활성 팩스 작업의 시작 시간을 나타내는 숫자입니다.
get_Status IFaxDeviceProvider::get_Status 속성은 FSP(팩스 서비스 공급자)가 성공적으로 로드되고 초기화되었는지 여부를 나타내는 숫자입니다.
get_Status IFaxInboundRoutingExtension::get_Status 속성은 팩스 라우팅 확장이 로드되고 성공적으로 초기화되었는지 여부를 나타내는 값입니다.
get_Status FaxIncomingJob 개체의 Status 속성을 검색합니다. Status 속성은 작업 큐에 있는 인바운드 팩스 작업의 현재 상태를 나타내는 숫자입니다.
get_Status Status 속성은 작업 큐에서 팩스 작업의 현재 상태를 나타내는 숫자입니다.
get_Status Status 속성은 아웃바운드 라우팅 그룹에 있는 팩스 디바이스의 집합적 상태를 나타냅니다.
get_Status IFaxOutboundRoutingRule::get_Status 속성은 아웃바운드 라우팅 규칙의 현재 상태를 나타냅니다. 예를 들어 규칙이 유효한지 여부와 팩스 작업에 적용할 수 있는지 여부입니다.
get_Status IFaxOutgoingJob::get_Status 속성은 작업 큐에서 아웃바운드 팩스 작업의 현재 상태를 나타내는 숫자입니다.
get_StreetAddress IFaxSender::get_StreetAddress 속성은 보낸 사람과 연결된 거리 주소를 포함하는 null로 끝나는 문자열입니다. (Get)
get_Subject IFaxDocument::get_Subject 속성은 팩스의 표지에 있는 제목 필드의 내용을 포함하는 null로 끝나는 문자열입니다. (Get)

get_Subject Subject 속성에는 인바운드 팩스 메시지와 연결된 제목이 포함됩니다. 이 속성은 null로 끝나는 문자열입니다. (Get)
get_Subject IFaxOutgoingJob::get_Subject 속성은 팩스의 표지에 있는 제목 필드의 내용을 포함하는 null로 끝나는 문자열입니다.
get_Subject IFaxOutgoingMessage::get_Subject 속성은 팩스의 표지에 있는 제목 필드의 내용을 포함하는 null로 끝나는 문자열입니다.
get_SubmissionId 팩스 문서의 제출 식별자를 검색합니다.
get_SubmissionId IFaxOutgoingJob::get_SubmissionId 속성은 제출 프로세스 중에 팩스 작업에 할당된 고유 식별자를 포함하는 null로 종료된 문자열입니다.
get_SubmissionId IFaxOutgoingMessage::get_SubmissionId 속성은 제출 프로세스 중에 팩스 메시지에 할당된 고유 식별자를 포함하는 null로 끝나는 문자열입니다.
get_SubmissionTime IFaxOutgoingJob::get_SubmissionTime 속성은 처리를 위해 아웃바운드 팩스 작업이 제출된 시간을 나타냅니다.
get_SubmissionTime IFaxOutgoingMessage::get_SubmissionTime 속성은 처리를 위해 아웃바운드 팩스 메시지가 제출된 시간을 나타냅니다.
get_SubmittedTime 부모 FaxPort 개체의 FaxStatus 개체에 대한 SubmittedTime 속성을 검색합니다. SubmittedTime 속성은 사용자가 활성 팩스 작업을 제출한 시간을 나타내는 숫자입니다.

get_TapiProviderName IFaxDeviceProvider::get_TapiProviderName 속성은 FSP(팩스 서비스 공급자) 팩스 디바이스와 연결된 TSP(전화 통신 서비스 공급자)의 이름을 포함하는 null로 끝나는 문자열입니다.
get_TiffTagString FaxTiff 개체의 TiffTagString 속성을 검색합니다. TiffTagString 속성은 지정된 TIFF(태그가 지정된 이미지 파일 형식) 태그(필드) 값을 포함하는 null로 끝나는 문자열입니다.
get_Title IFaxSender::get_Title 속성은 보낸 사람과 연결된 제목을 포함하는 null로 끝나는 문자열입니다. (Get)
get_TransmissionEnd TransmissionEnd 속성은 인바운드 팩스 작업이 전송을 완료한 시간을 나타냅니다.
get_TransmissionEnd TransmissionEnd 속성은 인바운드 팩스 메시지가 전송을 완료한 시간을 나타냅니다.
get_TransmissionEnd TransmissionEnd 속성은 팩스 작업이 전송을 완료한 시간을 나타냅니다.
get_TransmissionEnd IFaxOutgoingJob::get_TransmissionEnd 속성은 아웃바운드 팩스 작업이 전송을 완료한 시간을 나타냅니다.
get_TransmissionEnd IFaxOutgoingMessage::get_TransmissionEnd 속성은 팩스 아웃바운드 메시지 전송이 완료된 시간을 나타냅니다.
get_TransmissionStart TransmissionStart 속성은 팩스 인바운드 작업이 전송되기 시작한 시간을 나타냅니다.
get_TransmissionStart TransmissionStart 속성은 인바운드 팩스 메시지가 전송되기 시작한 시간을 나타냅니다.

get_TransmissionStart TransmissionStart 속성은 팩스 작업이 전송되기 시작한 시간을 나타냅니다.
get_TransmissionStart IFaxOutgoingJob::get_TransmissionStart 속성은 팩스 아웃바운드 작업이 전송되기 시작한 시간을 나타냅니다. 이 속성은 전송이 시작된 후에만 값을 갖습니다.
get_TransmissionStart IFaxOutgoingMessage::get_TransmissionStart 속성은 팩스 아웃바운드 메시지가 전송되기 시작한 시간을 나타냅니다.
get_Tsid FaxDoc 개체의 Tsid 속성을 설정하거나 검색합니다. Tsid 속성은 사용자 정의 TSID(전송 스테이션 식별자)를 포함하는 null로 끝나는 문자열입니다. (Get)
get_Tsid IFaxJob::get_Tsid 속성은 팩스 작업과 연결된 TSID(전송 스테이션 식별자)를 포함하는 null로 종료된 문자열입니다.
get_Tsid IFaxPort::get_Tsid 속성은 팩스 포트와 연결된 TSID(전송 스테이션 식별자)를 포함하는 null로 종료된 문자열입니다. (Get)
get_Tsid 부모 FaxPort 개체의 FaxStatus 개체에 대한 Tsid 속성을 검색합니다. Tsid 속성은 팩스 포트와 연결된 TSID(전송 스테이션 식별자)를 포함하는 null로 끝나는 문자열입니다.
get_Tsid FaxTiff 개체의 Tsid 속성을 검색합니다. Tsid 속성은 일반적으로 지정된 팩스 파일을 보낸 디바이스의 팩스 번호인 TSID(송신 스테이션 식별자) 정보를 포함하는 null로 종료된 문자열입니다.
get_TSID IFaxDevice::get_TSID 속성은 디바이스에 대한 TSID(전송 스테이션 식별자)를 포함하는 null로 종료된 문자열입니다. (Get)

get_TSID TSID 속성은 팩스 인바운드 작업과 연결된 TSID(전송 스테이션 식별자)를 포함하는 null로 종료된 문자열입니다.
get_TSID TSID 속성은 인바운드 팩스 메시지와 연결된 TSID(전송 스테이션 식별자)를 포함하는 null로 끝나는 문자열입니다.
get_TSID TSID 속성은 팩스 작업과 연결된 TSID(전송 스테이션 식별자)를 포함하는 null로 끝나는 문자열입니다.
get_TSID IFaxOutgoingJob::get_TSID 속성은 팩스 아웃바운드 작업과 연결된 TSID(전송 스테이션 식별자)를 포함하는 null로 종료된 문자열입니다.
get_TSID IFaxOutgoingMessage::get_TSID 속성은 팩스 아웃바운드 메시지와 연결된 TSID(전송 스테이션 식별자)를 포함하는 null로 끝나는 문자열입니다.
get_TSID IFaxSender::get_TSID 속성은 보낸 사람의 디바이스에 대한 TSID(전송 스테이션 식별자)를 포함하는 null로 끝나는 문자열입니다. (Get)
get_Type IFaxJob::get_Type 속성은 개체가 나타내는 팩스 작업의 형식을 설명하는 숫자입니다.
get_UniqueName IFaxDeviceProvider::get_UniqueName 속성은 팩스 서비스 공급자(FSP)를 식별하는 고유한 이름을 포함하는 null로 끝나는 문자열입니다.
get_UniqueName IFaxInboundRoutingExtension::get_UniqueName 속성은 팩스 라우팅 확장 프로그램에 대한 고유한 이름을 포함하는 null로 끝나는 문자열입니다. 팩스 서비스는 이 이름을 내부적으로 사용하여 팩스 라우팅 확장을 식별합니다.
get_UseArchive 팩스를 보관해야 하는지 여부를 나타내는 값을 설정하거나 검색합니다. (Get)

get_UseArchive UseArchive 속성은 팩스 서비스가 인바운드 팩스 메시지를 보관하는지 여부를 나타내는 부울 값입니다. (Get)
get_UseArchive IFaxOutgoingArchive::get_UseArchive 속성은 팩스 서비스가 아웃바운드 팩스 메시지를 보관하는지 여부를 나타내는 부울 값입니다. (Get)
get_UseDevice IFaxOutboundRoutingRule::get_UseDevice 속성은 아웃바운드 라우팅 규칙이 단일 팩스 디바이스를 가리키는지 여부를 나타내는 부울 값입니다. (Get)
get_UseDeviceTsid FaxServer 개체의 UseDeviceTsid 속성을 설정하거나 검색합니다. UseDeviceTsid 속성은 팩스 서버가 사용자가 지정한 TSID 대신 디바이스의 TSID(전송 스테이션 식별자)를 사용하는지 여부를 나타내는 부울 값입니다. (Get)
get_UseDeviceTSID TSID(전송 스테이션 식별자)가 사용되는지 여부를 나타내는 값을 설정하거나 검색합니다. (Get)
get_UseDeviceTSID IFaxOutgoingQueue::get_UseDeviceTSID 속성은 팩스 서비스가 보낸 사람 TSID 대신 TSID(디바이스 전송 스테이션 식별자)를 사용하는지 여부를 나타내는 부울 값입니다. (Get)
get_UsedRoutingMethods IFaxDevice::get_UsedRoutingMethods 속성은 디바이스에서 사용하는 라우팅 메서드와 연결된 GUID를 포함하는 문자열 배열입니다. 여기서 각 GUID는 인바운드 라우팅 메서드(FaxInboundRoutingMethod)를 나타냅니다.
get_UseForInboundRouting IFaxReceiptOptions::get_UseForInboundRouting 속성은 들어오는 팩스를 전자 메일 주소로 라우팅할 수 있도록 하는 Microsoft 라우팅 확장에 대한 FaxReceiptOptions 설정을 사용할지 여부를 설정하거나 검색합니다. (Get)

get_UserName IFaxJob::get_UserName 속성은 작업 큐에 팩스 작업을 제출한 사용자의 이름을 포함하는 null로 종료된 문자열입니다. IFaxJob::get_UserName 속성은 보내는 팩스 전송에만 적용됩니다.
get_WasReAssigned 팩스가 재할당되었는지를 나타냅니다.
GetAccount 계정 이름을 사용하여 IFaxAccount 개체를 반환합니다.
GetAccounts 팩스 서버의 모든 팩스 계정을 나타내는 IFaxAccounts 개체를 반환합니다.
GetDeviceProviders IFaxServer::GetDeviceProviders 메서드는 현재 팩스 서비스에 등록된 팩스 서비스 공급자(FSP)의 컬렉션인 IFaxDeviceProviders 인터페이스를 만듭니다.
GetDevices IFaxServer::GetDevices 메서드는 현재 팩스 서비스에 등록된 모든 팩스 서비스 공급자(FSP)가 노출하는 모든 팩스 디바이스 컬렉션인 IFaxDevices 인터페이스를 만듭니다.
GetExtensionProperty IFaxDevice::get_GetExtensionProperty 메서드는 디바이스 수준에 저장된 확장 구성 속성을 검색합니다.
GetExtensionProperty IFaxServer::GetExtensionProperty 메서드는 서버 수준에 저장된 확장 구성 속성을 검색합니다.
GetExtensions GetExtensions 메서드는 팩스 서비스에 등록된 인바운드 라우팅 확장의 컬렉션을 검색합니다.
GetGroups IFaxOutboundRouting::GetGroups 메서드는 아웃바운드 라우팅 그룹의 컬렉션을 나타내는 인터페이스를 검색합니다.

GetJob 작업 ID에 따라 현재 팩스 계정의 작업 큐에 들어오는 팩스 작업을 반환합니다.
GetJob 작업 ID에 따라 현재 팩스 계정의 작업 큐에서 나가는 팩스 작업을 반환합니다.
GetJob GetJob 메서드는 해당 ID에 따라 작업 큐에서 들어오는 팩스 작업을 반환합니다.
GetJob IFaxOutgoingQueue::GetJob 메서드는 ID에 따라 작업 큐에서 아웃바운드 팩스 작업을 반환합니다.
GetJobs 현재 팩스 계정에 대한 큐의 인바운드 팩스 작업 컬렉션을 반환합니다.
GetJobs 현재 팩스 계정에 대한 큐의 아웃바운드 팩스 작업 컬렉션을 반환합니다.
GetJobs GetJobs 메서드는 큐에 있는 인바운드 팩스 작업의 컬렉션을 반환합니다.
GetJobs IFaxOutgoingQueue::GetJobs 메서드는 작업 큐에 있는 아웃바운드 팩스 작업의 컬렉션을 반환합니다.
GetJobs GetJobs 메서드는 지정된 FaxServer 개체에 대한 FaxJobs 개체를 만들고 초기화합니다. FaxJobs 개체를 사용하면 연결된 팩스 서버에 대해 현재 큐에 대기 중인 작업을 열거할 수 있습니다.
GetMessage 팩스 메시지 ID를 사용하여 특정 팩스 계정에 대한 인바운드 팩스 보관 파일에서 팩스 메시지를 반환합니다.
GetMessage 팩스 메시지 ID를 사용하여 특정 팩스 계정에 대한 아웃바운드 팩스 보관 파일에서 팩스 메시지를 반환합니다.

GetMessage GetMessage 메서드는 팩스 메시지 ID를 사용하여 인바운드 팩스 보관 파일에서 팩스 메시지를 반환합니다.
GetMessage IFaxOutgoingArchive::GetMessage 메서드는 팩스 메시지 ID를 사용하여 아웃바운드 팩스 보관 파일에서 팩스 메시지를 반환합니다.
GetMessages 특정 팩스 계정에 대한 인바운드 팩스 메시지 보관에 대한 새 반복기(보관 커서)를 반환합니다.
GetMessages 특정 팩스 계정에 대한 아웃바운드 팩스 메시지 보관에 대한 새 반복기(보관 커서)를 반환합니다.
GetMessages GetMessages 메서드는 인바운드 팩스 메시지 보관에 대한 새 반복기(보관 커서)를 반환합니다.
GetMessages IFaxOutgoingArchive::GetMessages 메서드는 아웃바운드 팩스 메시지 보관에 대한 새 반복기(보관 커서)를 반환합니다. 자세한 내용은 IFaxOutgoingMessageIterator를 참조하세요.
GetMethods IFaxInboundRouting::GetMethods 메서드는 현재 팩스 서비스에 등록된 모든 인바운드 라우팅 확장 프로그램에서 노출하는 모든 인바운드 라우팅 메서드의 정렬된 컬렉션을 검색합니다.
GetPorts GetPorts 메서드는 지정된 FaxServer 개체에 대한 FaxPorts 개체를 만들고 초기화합니다. FaxPorts 개체를 사용하면 연결된 팩스 서버에 대한 팩스 포트 구성 정보를 열거할 수 있습니다.
GetRoutingMethods IFaxPort::GetRoutingMethods 인터페이스 메서드는 부모 FaxPort 개체에 대한 FaxRoutingMethods 개체를 만듭니다.

GetRules IFaxOutboundRouting::GetRules 메서드는 아웃바운드 라우팅 그룹의 컬렉션을 나타내는 인터페이스를 검색합니다.
GetStatus IFaxPort::GetStatus 메서드는 부모 FaxPort 개체에 대한 FaxStatus 개체를 만듭니다. FaxStatus 개체에는 팩스 포트의 현재 상태가 포함됩니다.
ItemByCountryAndArea IFaxOutboundRoutingRules::get_ItemByCountryAndArea 메서드는 라우팅 규칙의 국가/지역 코드 및 지역 코드를 사용하여 컬렉션에서 아웃바운드 라우팅 규칙(FaxOutboundRoutingRule 개체)을 반환합니다.
ListenToAccountEvents 계정이 수신 대기하는 이벤트를 나타내는 FAX_ACCOUNT_EVENTS_TYPE_ENUM 변수의 플래그를 설정합니다.
ListenToServerEvents IFaxServer::ListenToServerEvents 메서드는 하나 이상의 서버 이벤트 유형에 대한 알림을 받거나 이러한 알림을 중지하도록 FaxServer 개체를 등록합니다.
LoadDefaultSender IFaxSender::get_LoadDefaultSender 메서드는 FaxSender 개체를 기본 보낸 사람 정보로 채웁니다.
MoveFirst MoveFirst 메서드는 보관 커서를 인바운드 팩스 보관 파일의 첫 번째 팩스 메시지로 이동합니다.
MoveFirst IFaxOutgoingMessageIterator::MoveFirst 메서드는 보관 커서를 아웃바운드 보관 파일의 첫 번째 팩스 메시지로 이동합니다.
MoveNext MoveNext 메서드는 인바운드 팩스 보관의 다음 메시지로 보관 커서를 이동합니다.

MoveNext IFaxOutgoingMessageIterator::MoveNext 메서드는 보관 커서를 아웃바운드 보관 파일의 다음 팩스 메시지로 이동합니다.
OnActivityLoggingConfigChange 활동 로깅과 관련된 구성 변경이 있는 경우 팩스 서비스는 IFaxServerNotify2::OnActivityLoggingConfigChange 메서드를 호출합니다.
OnDevicesConfigChange 팩스 서비스는 팩스 디바이스 구성이 변경될 때 IFaxServerNotify2::OnDevicesConfigChange 메서드를 호출합니다.
OnDeviceStatusChange 팩스 서비스는 팩스 디바이스 상태가 변경되면 IFaxServerNotify2::OnDeviceStatusChange 메서드를 호출합니다.
OnEventLoggingConfigChange 이벤트 로깅과 관련된 구성 변경이 있는 경우 팩스 서비스는 IFaxServerNotify2::OnEventLoggingConfigChange 메서드를 호출합니다.
OnGeneralServerConfigChanged IFaxServer2::Configuration 속성이 변경되면 팩스 서비스에서 호출됩니다.
OnIncomingArchiveConfigChange 들어오는 팩스 보관 파일과 관련된 구성 변경 내용이 있는 경우 팩스 서비스는 IFaxServerNotify2::OnIncomingArchiveConfigChange 메서드를 호출합니다.
OnIncomingJobAdded 들어오는 팩스 작업이 특정 팩스 계정의 작업 큐에 추가되면 팩스 서비스에서 호출됩니다.
OnIncomingJobAdded 들어오는 팩스 작업이 작업 큐에 추가되면 팩스 서비스는 IFaxServerNotify2::OnIncomingJobAdded 메서드를 호출합니다.
OnIncomingJobChanged 특정 팩스 계정에 대해 들어오는 팩스 작업의 상태가 변경되면 팩스 서비스에서 호출합니다.

OnIncomingJobChanged 팩스 서비스는 들어오는 팩스 작업의 상태가 변경될 때 IFaxServerNotify2::OnIncomingJobChanged 메서드를 호출합니다.
OnIncomingJobRemoved 특정 팩스 계정의 작업 큐에서 들어오는 팩스 작업이 제거될 때 팩스 서비스에서 호출됩니다.
OnIncomingJobRemoved 팩스 서비스는 들어오는 팩스 작업이 작업 큐에서 제거될 때 IFaxServerNotify2::OnIncomingJobRemoved 메서드를 호출합니다.
OnIncomingMessageAdded 들어오는 메시지가 인바운드 팩스 보관에 추가되면 팩스 서비스에서 호출됩니다.
OnIncomingMessageAdded 들어오는 메시지가 인바운드 팩스 보관 파일에 추가되면 팩스 서비스는 IFaxServerNotify2::OnIncomingMessageAdded 메서드를 호출합니다.
OnIncomingMessageRemoved 들어오는 메시지가 인바운드 팩스 보관 파일에서 제거될 때 팩스 서비스에서 호출됩니다.
OnIncomingMessageRemoved 들어오는 메시지가 인바운드 팩스 보관 파일에서 제거되면 팩스 서비스는 IFaxServerNotify2::OnIncomingMessageRemoved 메서드를 호출합니다.
OnNewCall 팩스 서비스는 새 수신 팩스 호출이 있을 때 IFaxServerNotify2::OnNewCall 메서드를 호출합니다.
OnOutboundRoutingGroupsConfigChange 아웃바운드 팩스 라우팅 그룹과 관련된 구성 변경이 있는 경우 팩스 서비스는 IFaxServerNotify2::OnOutboundRoutingGroupsConfigChange 메서드를 호출합니다.

OnOutboundRoutingRulesConfigChange 아웃바운드 팩스 라우팅 규칙과 관련된 구성 변경이 있는 경우 팩스 서비스는 IFaxServerNotify2::OnOutboundRoutingRulesConfigChange 메서드를 호출합니다.
OnOutgoingArchiveConfigChange 송신 팩스 보관 파일과 관련된 구성 변경 내용이 있는 경우 팩스 서비스는 IFaxServerNotify2::OnOutgoingArchiveConfigChange 메서드를 호출합니다.
OnOutgoingJobAdded 송신 팩스 작업이 특정 팩스 계정의 작업 큐에 추가되면 팩스 서비스에서 호출됩니다.
OnOutgoingJobAdded 팩스 서비스는 나가는 팩스 작업이 작업 큐에 추가될 때 IFaxServerNotify2::OnOutgoingJobAdded 메서드를 호출합니다.
OnOutgoingJobChanged 특정 팩스 계정에 대한 송신 팩스 작업의 상태가 변경되면 팩스 서비스에서 호출합니다.
OnOutgoingJobChanged 팩스 서비스는 송신 팩스 작업의 상태가 변경될 때 IFaxServerNotify2::OnOutgoingJobChanged 메서드를 호출합니다.
OnOutgoingJobRemoved 송신 팩스 작업이 특정 팩스 계정의 작업 큐에서 제거될 때 팩스 서비스에 의해 호출됩니다.
OnOutgoingJobRemoved 팩스 서비스는 나가는 팩스 작업이 작업 큐에서 제거될 때 IFaxServerNotify2::OnOutgoingJobRemoved 메서드를 호출합니다.
OnOutgoingMessageAdded 송신 메시지가 아웃바운드 팩스 보관에 추가되면 팩스 서비스는 IFaxAccountNotify::OnOutgoingMessageAdded 메서드를 호출합니다.

OnOutgoingMessageAdded 송신 메시지가 아웃바운드 팩스 보관 파일에 추가되면 팩스 서비스는 IFaxServerNotify2::OnOutgoingMessageAdded 메서드를 호출합니다.
OnOutgoingMessageRemoved 보내는 메시지가 아웃바운드 팩스 보관 파일에서 제거될 때 팩스 서비스에서 호출됩니다.
OnOutgoingMessageRemoved 팩스 서비스는 팩스 아웃바운드 보관 파일에서 보내는 메시지가 제거될 때 IFaxServerNotify2::OnOutgoingMessageRemoved 메서드를 호출합니다.
OnOutgoingQueueConfigChange 송신 팩스 작업 큐와 관련된 구성 변경이 있는 경우 팩스 서비스는 IFaxServerNotify2::OnOutgoingQueueConfigChange 메서드를 호출합니다.
OnQueuesStatusChange 팩스 서비스는 팩스 작업 큐 상태가 변경되면 IFaxServerNotify2::OnQueuesStatusChange 메서드를 호출합니다.
OnReceiptOptionsChange 영수증 구성과 관련된 변경 내용이 있는 경우 팩스 서비스는 IFaxServerNotify2::OnReceiptOptionsChange 메서드를 호출합니다.
OnSecurityConfigChange 보안과 관련된 구성 변경이 있는 경우 팩스 서비스는 IFaxServerNotify2::OnSecurityConfigChange 메서드를 호출합니다.
OnServerActivityChange 팩스 서비스는 팩스 서비스 활동 및 상태가 변경될 때 IFaxServerNotify2::OnServerActivityChange 메서드를 호출합니다.
OnServerShutDown 팩스 서비스가 종료되면 호출됩니다.
OnServerShutDown 팩스 서비스는 팩스 서비스가 종료될 때 IFaxServerNotify2::OnServerShutDown 메서드를 호출합니다.

일시 중지 IFaxOutgoingJob::P ause 메서드는 아웃바운드 팩스 작업을 일시 중지합니다.
PFAX_LINECALLBACK FaxLineCallback 함수는 팩스 서비스가 팩스 서비스 공급자(FSP)에 TAPI(전화 통신 애플리케이션 프로그래밍 인터페이스) 이벤트를 전달하기 위해 호출하는 애플리케이션 정의 또는 라이브러리 정의 콜백 함수입니다.
PFAX_RECIPIENT_CALLBACKA FAX_RECIPIENT_CALLBACK 함수는 FaxSendDocumentForBroadcast 함수가 전송에 대한 사용자별 정보를 검색하기 위해 호출하는 애플리케이션 정의 또는 라이브러리 정의 콜백 함수입니다. (ANSI)
PFAX_RECIPIENT_CALLBACKW FAX_RECIPIENT_CALLBACK 함수는 FaxSendDocumentForBroadcast 함수가 전송에 대한 사용자별 정보를 검색하기 위해 호출하는 애플리케이션 정의 또는 라이브러리 정의 콜백 함수입니다. (유니코드)
PFAX_ROUTING_INSTALLATION_CALLBACKW FaxRoutingInstallationCallback 함수는 FaxRegisterRoutingExtension 함수가 팩스 라우팅 확장 DLL을 설치하기 위해 호출하는 라이브러리 정의 콜백 함수입니다.
PFAX_SEND_CALLBACK FaxSendCallback 함수는 송신 팩스 호출이 진행 중임을 팩스 서비스에 알리기 위해 FSP(팩스 서비스 공급자)가 호출하는 애플리케이션 정의 또는 라이브러리 정의 콜백 함수입니다.
PFAXABORT 팩스 클라이언트 애플리케이션은 팩스 작업을 종료하기 위해 FaxAbort 함수를 호출합니다.
PFAXACCESSCHECK 팩스 클라이언트 애플리케이션은 FaxAccessCheck 함수를 호출하여 사용자의 팩스 액세스 권한을 쿼리합니다.
PFAXCLOSE FaxClose 함수는 팩스 핸들을 닫습니다.

PFAXFREEBUFFER FaxFreeBuffer 함수는 팩스 클라이언트 애플리케이션에서 함수 호출의 결과로 이전에 할당된 버퍼와 연결된 리소스를 해제합니다.
PFAXGETPAGEDATA FaxGetPageData 함수는 팩스 작업에 대한 데이터의 첫 번째 페이지를 팩스 클라이언트 애플리케이션으로 반환합니다.
PFAXINITIALIZEEVENTQUEUE FaxInitializeEventQueue 함수는 호출하는 팩스 클라이언트 애플리케이션에 대한 팩스 이벤트 큐를 만듭니다. 큐를 사용하면 애플리케이션이 팩스 서버에서 비동기 이벤트의 알림을 받을 수 있습니다.
PFAXOPENPORT FaxOpenPort 함수는 팩스 포트 핸들을 팩스 클라이언트 애플리케이션에 반환합니다. 포트 핸들은 애플리케이션이 디바이스 관리 및 팩스 문서 라우팅을 용이하게 하는 다른 팩스 클라이언트 함수를 호출할 때 필요합니다.
PFAXROUTEADDFILE 팩스 라우팅 메서드는 FaxRouteAddFile 콜백 함수를 호출하여 받은 팩스 문서와 연결된 팩스 파일 목록에 파일을 추가합니다.
PFAXROUTEDELETEFILE 팩스 라우팅 메서드는 FaxRouteDeleteFile 콜백 함수를 호출하여 받은 팩스 문서와 연결된 팩스 파일 목록에서 파일을 삭제합니다.
PFAXROUTEENUMFILE FaxRouteEnumFile 콜백 함수는 받은 팩스 문서와 연결된 팩스 파일 목록에서 파일 이름을 받습니다.
PFAXROUTEENUMFILES 팩스 라우팅 메서드는 FaxRouteEnumFiles 콜백 함수를 호출하여 받은 팩스 문서와 연결된 팩스 파일 목록의 파일을 열거합니다.
PFAXROUTEGETFILE 팩스 라우팅 메서드는 FaxRouteGetFile 콜백 함수를 호출하여 받은 팩스 문서와 연결된 팩스 파일 목록에서 특정 파일의 이름을 검색합니다.

PFAXROUTEMETHOD FaxRouteMethod 함수는 팩스 라우팅 확장 DLL에서 정의한 함수 이름의 자리 표시자입니다. 이 함수는 정의된 팩스 라우팅 프로시저를 실행합니다.
PFAXROUTEMODIFYROUTINGDATA 팩스 라우팅 메서드는 FaxRouteModifyRoutingData 콜백 함수를 호출하여 후속 팩스 라우팅 메서드에 대한 라우팅 데이터를 수정합니다.
put_AgeLimit AgeLimit 속성은 팩스 서비스가 인바운드 팩스 보관에 팩스 메시지를 보관하는 기간(일)을 나타내는 값입니다. (Put)
put_AgeLimit IFaxOutgoingArchive::get_AgeLimit 속성은 팩스 서비스가 아웃바운드 팩스 보관에 팩스 메시지를 보관하는 기간(일)을 나타내는 값입니다. (Put)
put_AgeLimit IFaxOutgoingQueue::get_AgeLimit 속성은 팩스 서비스가 팩스 작업 큐에서 전송되지 않은 작업을 유지하는 일 수를 나타내는 값입니다. (Put)
put_AllowedReceipts IFaxReceiptOptions::get_AllowedReceipts 속성은 허용되는 배달 확인 유형을 지정하는 값입니다. (Put)
put_AllowPersonalCoverPages 개인 표지가 허용되는지 여부를 나타내는 값을 설정하거나 검색합니다. (Put)
put_AllowPersonalCoverPages AllowPersonalCoverPages 속성은 팩스 클라이언트 애플리케이션에 팩스 전송이 있는 사용자 디자인 표지를 포함할 수 있는지 여부를 나타내는 부울 값입니다. (Put)
put_ArchiveAgeLimit 팩스 메시지가 서버에 보관되는 기간을 나타내는 값을 설정하거나 검색합니다. (Put)

put_ArchiveDirectory IFaxServer::get_ArchiveDirectory 메서드는 FaxServer 개체에 대한 ArchiveDirectory 속성을 검색합니다. ArchiveDirectory 속성은 팩스 서버에서 보관된 아웃바운드 팩스를 저장하는 위치를 포함하는 null로 끝나는 문자열입니다. (Put)
put_ArchiveFolder ArchiveFolder 속성은 보관된 인바운드 팩스에 대한 팩스 서버의 폴더 위치를 지정하는 null로 끝나는 문자열입니다. (Put)
put_ArchiveFolder IFaxOutgoingArchive::get_ArchiveFolder 속성은 보관된 아웃바운드 팩스에 대한 팩스 서버의 폴더 위치를 지정하는 null로 끝나는 문자열입니다. (Put)
put_ArchiveLocation 서버에서 보관 파일의 위치를 나타내는 값을 설정하거나 검색합니다. (Put)
put_ArchiveOutboundFaxes FaxServer 개체에 대한 ArchiveOutboundFaxes 속성을 설정하거나 검색합니다. ArchiveOutboundFaxes 속성은 팩스 서버가 송신 팩스 전송을 보관하는지 여부를 나타내는 부울 값입니다. (Put)
put_AttachFaxToReceipt IFaxDocument::get_AttachFaxToReceipt 속성은 팩스를 영수증에 첨부할지 여부를 나타냅니다. (Put)
put_AuthenticationType IFaxReceiptOptions::get_AuthenticationType 속성은 SMTP(Simple Mail Transport Protocol) 서버에 연결할 때 팩스 서비스에서 사용하는 인증 유형을 지정합니다. (Put)
put_AutoCreateAccountOnConnect 연결이 시작되면 서버에서 팩스 계정을 자동으로 만드는지 여부를 나타내는 값을 설정하거나 검색합니다. (Put)
put_BillingCode IFaxSender::get_BillingCode 속성은 보낸 사람과 연결된 청구 코드를 포함하는 null로 끝나는 문자열입니다. (Put)

put_BillingCode FaxDoc 개체의 BillingCode 속성을 설정하거나 검색합니다. BillingCode 속성은 팩스 전송에 적용되는 선택적 청구 코드를 포함하는 null로 끝나는 문자열입니다. (Put)
put_Blocked Blocked 속성은 들어오는 팩스의 작업 큐가 차단되는지 여부를 나타내는 부울 값입니다. (Put)
put_Blocked IFaxOutgoingQueue::get_Blocked 속성은 보내는 팩스의 작업 큐가 차단되었는지 여부를 나타내는 부울 값입니다. (Put)
put_Bodies 팩스 문서에 하나 이상의 문서 컬렉션을 제공합니다. (Put)
put_Body IFaxDocument::get_Body 속성은 팩스 본문을 구성하는 파일의 경로를 제공합니다. 팩스 본문은 표지 이외의 팩스 페이지로 구성됩니다. (Put)
put_Branding 팩스 서버에서 보내는 팩스에 브랜딩 표시를 생성하는지 여부를 나타내는 값을 설정하거나 검색합니다. (Put)
put_Branding IFaxOutgoingQueue::get_Branding 속성은 팩스 서비스가 송신 팩스 전송 맨 위에 브랜드(배너)를 생성하는지 여부를 나타내는 부울 값입니다. (Put)
put_Branding FaxServer 개체의 브랜딩 속성을 설정하거나 검색합니다. Branding 속성은 팩스 서버가 팩스 전송 맨 위에 브랜딩 정보를 생성하는지 여부를 나타내는 부울 값입니다. (Put)
put_Company IFaxSender::get_Company 속성은 보낸 사람과 연결된 회사 이름을 포함하는 null로 끝나는 문자열입니다. (Put)

put_CoverPage IFaxDocument::get_CoverPage 속성은 팩스 문서와 연결할 표지 서식 파일(.cov)의 이름을 포함하는 null로 끝나는 문자열입니다. (Put)
put_CoverpageName FaxDoc 개체의 CoverpageName 속성을 설정하거나 검색합니다. CoverpageName 속성은 개체와 연결된 표지 템플릿 파일(.cov)의 이름을 포함하는 null로 끝나는 문자열입니다. (Put)
put_CoverpageNote FaxDoc 개체의 CoverpageNote 속성을 설정하거나 검색합니다. CoverpageNote 속성은 팩스 전송과 관련된 보낸 사람의 메시지 또는 메모 텍스트가 포함된 null로 끝나는 문자열입니다. (Put)
put_CoverpageSubject FaxDoc 개체의 CoverpageSubject 속성을 설정하거나 검색합니다. CoverpageSubject 속성은 팩스 전송의 제목 줄을 포함하는 null로 끝나는 문자열입니다. (Put)
put_CoverPageType IFaxDocument::get_CoverPageType 속성은 지정된 표지 템플릿 파일(.cov)이 서버 기반 표지 파일인지 아니면 로컬 컴퓨터 기반 표지 파일인지 여부를 나타내는 열거형의 값입니다. (Put)
put_Csid IFaxPort::get_Csid 속성은 팩스 포트와 연결된 호출된 스테이션 식별자(CSID)를 포함하는 null로 끝나는 문자열입니다. (Put)
put_CSID IFaxDevice::get_CSID 속성은 디바이스에 대해 호출된 스테이션 식별자(CSID)를 포함하는 null로 끝나는 문자열입니다. (Put)
put_DatabasePath IFaxActivityLogging::get_DatabasePath 속성은 활동 로그 데이터베이스 파일의 경로를 포함하는 null로 끝나는 문자열입니다. (Put)
put_Department IFaxSender::get_Department 속성은 보낸 사람과 연결된 부서를 포함하는 null로 끝나는 문자열입니다. (Put)

put_Description IFaxDevice::get_Description 속성은 팩스 디바이스에 대한 사용자 친화적인 설명을 포함하는 null로 끝나는 문자열입니다. 이 문자열은 사용자에게 표시하기에 적합합니다. (Put)
put_Descriptor Descriptor 속성은 IFaxServer 개체의 보안 설명자를 나타냅니다. (Put)
put_Descriptor IFaxServer2 개체의 보안 설명자를 나타냅니다. (Put)
put_DeviceId IFaxOutboundRoutingRule::get_DeviceId 속성은 아웃바운드 라우팅 규칙이 단일 팩스 디바이스를 가리키는 경우 디바이스 ID를 지정합니다. (Put)
put_DirtyDays FaxServer 개체의 DirtyDays 속성을 설정하거나 검색합니다. DirtyDays 속성은 팩스 서버가 팩스 작업 큐에서 전송되지 않은 작업을 유지하는 일 수입니다. (Put)
put_DiscountRateEnd 할인율 기간이 종료되는 시간을 나타내는 값을 설정하거나 검색합니다. (Put)
put_DiscountRateEnd IFaxOutgoingQueue::get_DiscountRateEnd 속성은 팩스 전송 할인 기간이 끝나는 시간을 나타내는 값입니다. 할인 기간은 발신 팩스에 적용됩니다. (Put)
put_DiscountRateEndHour FaxServer 개체에 대한 DiscountRateEndHour 속성을 설정하거나 검색합니다. DiscountRateEndHour 속성은 할인 기간이 종료되는 시간을 나타내는 숫자입니다. 할인 기간은 발신 팩스 전송에만 적용됩니다. (Put)
put_DiscountRateEndMinute FaxServer 개체에 대한 DiscountRateEndMinute 속성을 설정하거나 검색합니다. DiscountRateEndMinute 속성은 할인 기간이 종료되는 분을 나타내는 숫자입니다. 할인 기간은 발신 팩스 전송에만 적용됩니다. (Put)
put_DiscountRateStart 할인율 기간이 시작되는 시간을 나타내는 값을 설정하거나 검색합니다. (Put)

put_DiscountRateStart IFaxOutgoingQueue::get_DiscountRateStart 속성은 팩스 전송 할인 기간이 시작되는 시간을 나타내는 값입니다. 할인 기간은 발신 팩스에 적용됩니다. (Put)
put_DiscountRateStartHour FaxServer 개체에 대한 DiscountRateStartHour 속성을 설정하거나 검색합니다. DiscountRateStartHour 속성은 할인 기간이 시작되는 시간을 나타내는 숫자입니다. 할인 기간은 발신 팩스 전송에만 적용됩니다. (Put)
put_DiscountRateStartMinute FaxServer 개체에 대한 DiscountRateStartMinute 속성을 설정하거나 검색합니다. DiscountRateStartMinute 속성은 할인 기간이 시작되는 분을 나타내는 숫자입니다. 할인 기간은 발신 팩스 전송에만 적용됩니다. (Put)
put_DiscountSend FaxDoc 개체에 대한 DiscountSend 속성을 설정하거나 검색합니다. DiscountSend 속성은 할인 기간 동안 팩스 서버에서 팩스를 전송하는지 여부를 나타내는 부울 값입니다. (Put)
put_DisplayName FaxDoc 개체의 DisplayName 속성을 설정하거나 검색합니다. DisplayName 속성은 팩스 문서와 연결할 이름을 포함하는 null로 끝나는 문자열입니다. (Put)
put_DocumentName IFaxDocument::get_DocumentName 속성은 팩스 문서에 표시할 사용자 친화적인 이름을 포함하는 null로 끝나는 문자열입니다. 값은 표시 용도로만 사용됩니다. (Put)
put_Email IFaxSender::get_Email 속성은 보낸 사람과 연결된 전자 메일 주소를 포함하는 null로 끝나는 문자열입니다. (Put)
put_EmailAddress FaxDoc 개체의 EmailAddress 속성을 설정하거나 검색합니다. EmailAddress 속성은 팩스 전송 보낸 사람의 이메일 주소를 포함하는 null로 끝나는 문자열입니다. (Put)

put_Enable IFaxRoutingMethod::get_Enable 속성은 특정 팩스 포트에서 팩스 라우팅 방법을 사용할 수 있는지 여부를 나타내는 부울 값입니다. (Put)
put_FaxNumber IFaxRecipient::get_FaxNumber 속성은 받는 사람과 연결된 팩스 번호를 포함하는 null로 끝나는 문자열입니다. (Put)
put_FaxNumber IFaxSender::get_FaxNumber 속성은 보낸 사람과 연결된 팩스 번호를 포함하는 null로 끝나는 문자열입니다. (Put)
put_FaxNumber FaxDoc 개체의 FaxNumber 속성을 설정하거나 검색합니다. FaxNumber 속성은 팩스 서버에서 팩스 전송을 보낼 팩스 번호를 포함하는 null로 끝나는 문자열입니다. (Put)
put_FileName FaxDoc 개체의 FileName 속성을 설정하거나 검색합니다. FileName 속성은 개체와 연결된 문서 파일의 이름을 포함하는 null로 끝나는 문자열입니다. (Put)
put_GeneralEventsLevel IFaxEventLogging::get_GeneralEventsLevel 속성은 팩스 서비스가 애플리케이션 로그에서 일반 이벤트를 기록하는 세부 정보 수준을 나타냅니다. (Put)
put_GroupBroadcastReceipts IFaxDocument::get_GroupBroadcastReceipts 속성은 브로드캐스트의 각 수신자에 대해 개별 배달 영수증을 보낼지 또는 모든 받는 사람에 대한 요약 영수증을 보낼지 여부를 나타내는 부울 값입니다. (Put)
put_GroupName IFaxOutboundRoutingRule::get_GroupName 속성은 아웃바운드 라우팅 규칙이 팩스 디바이스 그룹을 가리키는 경우 그룹 이름을 지정합니다. (Put)
put_HasCoverPage 팩스에 표지가 있는지 여부를 나타내는 플래그입니다. (Put)
put_HighQuotaWaterMark 워터마크의 할당된 최대 크기를 나타내는 값을 설정하거나 검색합니다. (Put)

put_HighQuotaWaterMark HighQuotaWaterMark 속성은 인바운드 팩스 메시지 보관 크기의 상위 경고 임계값을 메가바이트 단위로 지정하는 값입니다. (Put)
put_HighQuotaWaterMark IFaxOutgoingArchive::get_HighQuotaWaterMark 속성은 인바운드 팩스 메시지 보관 크기의 상한 임계값(MB)을 지정하는 값입니다. (Put)
put_HomePhone IFaxSender::get_HomePhone 속성은 보낸 사람과 연결된 집 전화 번호를 포함하는 null로 끝나는 문자열입니다. (Put)
put_Image FaxTiff 개체의 Image 속성을 설정하거나 검색합니다. (Put)
put_InboundEventsLevel IFaxEventLogging::get_InboundEventsLevel 속성은 팩스 서비스가 애플리케이션 로그의 인바운드 팩스 전송에 대한 이벤트를 기록하는 세부 수준을 나타냅니다. (Put)
put_IncomingFaxesArePublic 들어오는 팩스가 모든 사람이 볼 수 있는지 또는 비공개로 볼 수 있는지 여부를 나타냅니다. (Put)
put_IncomingQueueBlocked 들어오는 팩스의 팩스 서버 큐가 차단되었는지 여부를 나타내는 값을 설정하거나 검색합니다. (Put)
put_InformationType IFaxSecurity::InformationType 속성은 보안 정보 유형을 나타냅니다. (Put)
put_InformationType 보안 정보 유형을 검색합니다. (Put)
put_InitEventsLevel IFaxEventLogging::get_InitEventsLevel 속성은 팩스 서비스가 애플리케이션 로그에서 초기화(서버 시작) 및 종료(서버 종료) 이벤트를 기록하는 세부 정보 수준을 나타냅니다. (Put)

put_LogIncoming IFaxActivityLogging::get_LogIncoming 속성은 팩스 서비스가 활동 로그 데이터베이스에서 들어오는 팩스에 대한 항목을 기록하는지 여부를 나타내는 부울 값입니다. (Put)
put_LogOutgoing IFaxActivityLogging::get_LogOutgoing 속성은 팩스 서비스가 활동 로그 데이터베이스에서 보내는 팩스에 대한 항목을 기록하는지 여부를 나타내는 부울 값입니다. (Put)
put_LowQuotaWaterMark 워터마크의 최소 크기를 나타내는 값을 설정하거나 검색합니다. (Put)
put_LowQuotaWaterMark LowQuotaWaterMark 속성은 인바운드 팩스 메시지 보관에 대한 낮은 경고 임계값을 메가바이트 단위로 지정하는 값입니다. (Put)
put_LowQuotaWaterMark IFaxOutgoingArchive::get_LowQuotaWaterMark 속성은 아웃바운드 팩스 메시지 보관에 대한 낮은 임계값(MB)을 지정하는 값입니다. (Put)
put_Name IFaxRecipient::get_Name 속성은 받는 사람의 이름을 포함하는 null로 끝나는 문자열입니다. (Put)
put_Name IFaxSender::get_Name 속성은 보낸 사람의 이름을 포함하는 null로 끝나는 문자열입니다. (Put)
put_Note IFaxDocument::get_Note 속성은 팩스의 표지에 있는 메모 필드의 내용을 포함하는 null로 끝나는 문자열입니다. (Put)
put_OfficeLocation IFaxSender::get_OfficeLocation 속성은 보낸 사람의 사무실 위치를 포함하는 null로 끝나는 문자열입니다. (Put)

put_OfficePhone IFaxSender::get_OfficePhone 속성은 보낸 사람과 연결된 사무실 전화 번호를 포함하는 null로 끝나는 문자열입니다. (Put)
put_OutboundEventsLevel IFaxEventLogging::get_OutboundEventsLevel 속성은 팩스 서비스가 애플리케이션 로그의 아웃바운드 팩스 전송에 대한 이벤트를 기록하는 세부 수준을 나타냅니다. (Put)
put_OutgoingQueueAgeLimit 배달할 수 없는 팩스 메시지가 삭제되기 전에 팩스 서버에 보관되는 시간을 나타내는 값을 설정하거나 검색합니다. (Put)
put_OutgoingQueueBlocked 보내는 팩스의 팩스 서버 큐가 차단되었는지 여부를 나타내는 값을 설정하거나 검색합니다. (Put)
put_OutgoingQueuePaused 나가는 큐가 일시 중지되었는지 여부를 나타내는 값을 설정하거나 검색합니다. (Put)
put_Paused IFaxOutgoingQueue::get_Paused 속성은 보내는 팩스의 작업 큐가 일시 중지되었는지 여부를 나타내는 부울 값입니다. (Put)
put_PauseServerQueue FaxServer 개체에 대한 PauseServerQueue 속성을 설정하거나 검색합니다. PauseServerQueue 속성은 팩스 서버가 팩스 작업 큐를 일시 중지했는지 여부를 나타내는 부울 값입니다. (Put)
put_PrefetchSize PrefetchSize 속성은 프리페치(미리 읽기) 버퍼의 크기를 나타냅니다. (Put)
put_PrefetchSize IFaxOutgoingMessageIterator::get_PrefetchSize 속성은 프리페치(미리 읽기) 버퍼의 크기를 나타냅니다. 그러면 개체가 콘텐츠를 새로 고쳐야 할 때 반복기 개체가 팩스 서버에서 검색하는 팩스 메시지의 수를 결정합니다. (Put)

put_Priority IFaxDocument::get_Priority 속성은 팩스를 보낼 때 사용할 우선 순위를 지정합니다. 예를 들어 보통, 낮음 또는 높은 우선 순위입니다. (Put)
put_Priority Priority 속성은 서비스가 팩스 작업을 받을 때 팩스 서비스가 라우팅 메서드를 호출하는 순서와 연결된 값입니다. (Put)
put_Priority IFaxPort::get_Priority 속성은 지정된 팩스 포트에 지정된 전송 우선 순위를 나타내는 숫자입니다. 우선 순위는 사용 가능한 팩스 디바이스가 발신 전송을 보내는 상대 순서를 결정합니다. (Put)
put_Read 팩스를 읽었는지를 나타내는 플래그입니다. (Put)
put_Read 팩스를 읽었는지를 나타냅니다. (Put)
put_ReceiptAddress IFaxDocument::get_ReceiptAddress 속성은 팩스 작업이 최종 상태에 도달할 때 팩스 서비스가 배달 확인을 보내야 하는 전자 메일 주소를 나타내는 null로 종료된 문자열입니다. (Put)
put_ReceiptType IFaxDocument::get_ReceiptType 속성은 팩스 작업이 최종 상태에 도달할 때 배달할 배달 확인 유형을 지정합니다. (Put)
put_Receive IFaxPort::get_Receive 속성은 지정된 팩스 포트에서 팩스를 받을 수 있는지 여부를 나타내는 부울 값입니다. (Put)
put_ReceiveMode ReceiveMode 속성은 디바이스가 들어오는 호출에 응답하는 방법을 정의하는 FAX_DEVICE_RECEIVE_MODE_ENUM 열거형의 값입니다. (Put)

put_RecipientAddress FaxDoc 개체의 RecipientAddress 속성을 설정하거나 검색합니다. RecipientAddress 속성은 팩스 전송 수신자의 주소가 포함된 null로 끝나는 문자열입니다. (Put)
put_RecipientCity FaxDoc 개체의 RecipientCity 속성을 설정하거나 검색합니다. RecipientCity 속성은 팩스 전송 수신자의 도시 이름을 포함하는 null로 끝나는 문자열입니다. (Put)
put_RecipientCompany FaxDoc 개체의 RecipientCompany 속성을 설정하거나 검색합니다. RecipientCompany 속성은 팩스 전송 수신자의 회사 이름을 포함하는 null로 끝나는 문자열입니다. (Put)
put_RecipientCountry FaxDoc 개체의 RecipientCountry 속성을 설정하거나 검색합니다. RecipientCountry 속성은 팩스 전송 수신자의 국가/지역을 포함하는 null로 끝나는 문자열입니다. (Put)
put_RecipientDepartment FaxDoc 개체의 RecipientDepartment 속성을 설정하거나 검색합니다. RecipientDepartment 속성은 팩스 전송 받는 사람의 부서를 포함하는 null로 끝나는 문자열입니다. (Put)
put_RecipientHomePhone FaxDoc 개체의 RecipientHomePhone 속성을 설정하거나 검색합니다. RecipientHomePhone 속성은 팩스 전송 수신자의 집 전화 번호를 포함하는 null로 끝나는 문자열입니다. (Put)
put_RecipientName FaxDoc 개체의 RecipientName 속성을 설정하거나 검색합니다. RecipientName 속성은 팩스 전송의 받는 사람의 이름을 포함하는 null로 끝나는 문자열입니다. (Put)
put_RecipientOffice FaxDoc 개체의 RecipientOffice 속성을 설정하거나 검색합니다. RecipientOffice 속성은 팩스 전송 수신자의 사무실이 포함된 null로 끝나는 문자열입니다. (Put)

put_RecipientOfficePhone FaxDoc 개체의 RecipientOfficePhone 속성을 설정하거나 검색합니다. RecipientOfficePhone 속성은 팩스 전송 수신자의 사무실 전화 번호를 포함하는 null로 끝나는 문자열입니다. (Put)
put_Recipients 인바운드 팩스 메시지와 연결된 받는 사람을 포함합니다. 이 속성은 null로 끝나는 문자열입니다. (Put)
put_RecipientState FaxDoc 개체의 RecipientState 속성을 설정하거나 검색합니다. RecipientState 속성은 팩스 전송 수신자의 상태를 포함하는 null로 끝나는 문자열입니다. (Put)
put_RecipientTitle FaxDoc 개체의 RecipientTitle 속성을 설정하거나 검색합니다. RecipientTitle 속성은 팩스 전송 수신자의 제목을 포함하는 null로 끝나는 문자열입니다. (Put)
put_RecipientZip FaxDoc 개체의 RecipientZip 속성을 설정하거나 검색합니다. RecipientZip 속성은 팩스 전송 수신자의 우편 번호를 포함하는 null로 끝나는 문자열입니다. (Put)
put_Retries 지정된 팩스 작업에 대한 다시 시도 횟수를 나타내는 값을 설정하거나 검색합니다. (Put)
put_Retries IFaxOutgoingQueue::get_Retries 속성은 초기 전송이 실패할 때 팩스 서비스가 송신 팩스를 재전송하려고 시도하는 횟수를 나타내는 값입니다. (Put)
put_Retries FaxServer 개체에 대한 Retries 속성을 설정하거나 검색합니다. Retries 속성은 초기 전송이 실패할 때 팩스 서버에서 송신 팩스를 다시 전송하려고 시도하는 횟수를 나타내는 값입니다. (Put)
put_RetryDelay 팩스 서비스가 실패한 팩스 전송을 다시 시도하기 전에 대기해야 하는 시간을 나타내는 값을 설정하거나 검색합니다. (Put)

put_RetryDelay IFaxOutgoingQueue::get_RetryDelay 속성은 팩스 서비스가 아웃바운드 팩스 작업을 다시 전송하기 전에 대기하는 시간 간격(분)을 나타내는 값입니다. (Put)
put_RetryDelay FaxServer 개체에 대한 RetryDelay 속성을 설정하거나 검색합니다. RetryDelay 속성은 시간 간격(분)을 나타내는 값으로, 팩스 서버는 아웃바운드 팩스 작업을 다시 전송하기 전에 대기합니다. (Put)
put_Rings IFaxPort::get_Rings 속성은 팩스 포트가 통화에 응답하기 전에 수신 팩스 호출이 대기해야 하는 링 수를 나타냅니다. (Put)
put_RingsBeforeAnswer IFaxDevice::get_RingsBeforeAnswer 속성은 팩스 장치가 들어오는 팩스 호출에 응답하기 전에 발생하는 링 수를 지정하는 숫자입니다. (Put)
put_ScheduleTime IFaxDocument::get_ScheduleTime 속성은 팩스 서비스에 처리하기 위해 팩스를 제출하는 시간을 나타냅니다. (Put)
put_ScheduleType IFaxDocument::get_ScheduleType 속성은 팩스 작업을 예약할 시기를 나타냅니다. 예를 들어 팩스 서비스가 지정된 시간에 또는 미리 정의된 할인 기간 동안 즉시 팩스를 보내도록 지정할 수 있습니다. (Put)
put_Send IFaxPort::get_Send 속성은 팩스 포트에서 팩스를 보낼 수 있는지 여부를 나타내는 부울 값입니다. (Put)
put_SendCoverpage FaxDoc 개체의 SendCoverpage 속성을 설정하거나 검색합니다. SendCoverpage 속성은 지정된 표지 파일이 팩스 서버에 저장되는지 여부를 나타내는 부울 값입니다. (Put)
put_SendEnabled IFaxDevice::get_SendEnabled 속성은 팩스 디바이스에서 팩스를 보낼 수 있는지 여부를 나타내는 부울 값입니다. (Put)

put_SenderAddress FaxDoc 개체의 SenderAddress 속성을 설정하거나 검색합니다. SenderAddress 속성은 팩스 전송 발신자의 주소가 포함된 null로 끝나는 문자열입니다. (Put)
put_SenderCompany FaxDoc 개체의 SenderCompany 속성을 설정하거나 검색합니다. SenderCompany 속성은 팩스 전송 보낸 사람의 회사 이름을 포함하는 null로 끝나는 문자열입니다. (Put)
put_SenderDepartment FaxDoc 개체의 SenderDepartment 속성을 설정하거나 검색합니다. SenderDepartment 속성은 팩스 전송 보낸 사람의 부서를 포함하는 null로 끝나는 문자열입니다. (Put)
put_SenderFax FaxDoc 개체의 SenderFax 속성을 설정하거나 검색합니다. SenderFax 속성은 아웃바운드 팩스 전송 보낸 사람의 팩스 번호를 포함하는 null로 끝나는 문자열입니다. (Put)
put_SenderFaxNumber 인바운드 팩스 메시지와 연결된 보낸 사람의 팩스 번호를 포함합니다. 이 속성은 null로 끝나는 문자열입니다. (Put)
put_SenderHomePhone FaxDoc 개체의 SenderHomePhone 속성을 설정하거나 검색합니다. SenderHomePhone 속성은 팩스 전송 발신자의 집 전화 번호를 포함하는 null로 끝나는 문자열입니다. (Put)
put_SenderName 인바운드 팩스 메시지와 연결된 보낸 사람의 이름을 포함합니다. 이 속성은 null로 끝나는 문자열입니다. (Put)
put_SenderName FaxDoc 개체의 SenderName 속성을 설정하거나 검색합니다. SenderName 속성은 팩스 전송 보낸 사람의 이름을 포함하는 null로 끝나는 문자열입니다. (Put)
put_SenderOffice FaxDoc 개체의 SenderOffice 속성을 설정하거나 검색합니다. SenderOffice 속성은 팩스 전송 발신자의 사무실이 포함된 null로 종료된 문자열입니다. (Put)

put_SenderOfficePhone FaxDoc 개체의 SenderOfficePhone 속성을 설정하거나 검색합니다. SenderOfficePhone 속성은 팩스 전송 발신자의 사무실 전화 번호를 포함하는 null로 끝나는 문자열입니다. (Put)
put_SenderTitle FaxDoc 개체의 SenderTitle 속성을 설정하거나 검색합니다. SenderTitle 속성은 팩스 전송 발신자의 제목을 포함하는 null로 끝나는 문자열입니다. (Put)
put_ServerCoverpage FaxDoc 개체의 ServerCoverpage 속성을 설정하거나 검색합니다. ServerCoverpage 속성은 지정된 표지 파일이 팩스 서버에 저장되는지 여부를 나타내는 부울 값입니다. (Put)
put_ServerCoverpage FaxServer 개체의 ServerCoverpage 속성을 설정하거나 검색합니다. ServerCoverpage 속성은 팩스 서버에서 일반 표지만 사용할 수 있는지 여부를 나타내는 부울 값입니다. (Put)
put_ServerMapiProfile FaxServer 개체의 ServerMapiProfile 속성을 설정하거나 검색합니다. ServerMapiProfile 속성은 팩스 서버가 들어오는 팩스 전송을 라우팅하는 데 사용하는 MAPI 사용자 프로필을 포함하는 null로 종료된 문자열입니다. (Put)
put_SizeQuotaWarning 크기 할당량 경고가 켜져 있는지 여부를 나타내는 값을 설정하거나 검색합니다. (Put)
put_SizeQuotaWarning SizeQuotaWarning 속성은 인바운드 보관 파일의 크기가 HighQuotaWaterMark 속성에 정의된 제한을 초과할 때 팩스 서비스가 이벤트 로그에 경고를 발생시키는지 여부를 나타내는 부울 값입니다. (Put)
put_SizeQuotaWarning IFaxOutgoingArchive::get_SizeQuotaWarning 속성은 아웃바운드 보관 파일의 크기가 IFaxOutgoingArchive::get_HighQuotaWaterMark 속성에 정의된 제한을 초과할 때 팩스 서비스가 이벤트 로그에 경고를 발생시키는지 여부를 나타내는 부울 값입니다. (Put)

put_SMTPPassword IFaxReceiptOptions::get_SMTPPassword 속성은 인증된 연결에 사용되는 SMTP(Simple Mail Transport Protocol) 암호를 포함하는 null로 끝나는 문자열입니다. (Put)
put_SMTPPort IFaxReceiptOptions::get_SMTPPort 속성은 SMTP(Simple Mail Transport Protocol) 포트 번호를 지정하는 값입니다. (Put)
put_SMTPSender IFaxReceiptOptions::get_SMTPSender 속성은 메일 메시지 확인 보낸 사람의 SMTP(Simple Mail Transport Protocol) 전자 메일 주소를 포함하는 null로 끝나는 문자열입니다. (Put)
put_SMTPServer IFaxReceiptOptions::get_SMTPServer 속성은 SMTP(Simple Mail Transport Protocol) 서버의 이름을 포함하는 null로 끝나는 문자열입니다. (Put)
put_SMTPUser IFaxReceiptOptions::get_SMTPUser 속성은 인증된 연결에 사용되는 SMTP(Simple Mail Transport Protocol) 사용자 이름을 포함하는 null로 끝나는 문자열입니다. (Put)
put_StreetAddress IFaxSender::get_StreetAddress 속성은 보낸 사람과 연결된 거리 주소를 포함하는 null로 끝나는 문자열입니다. (Put)
put_Subject IFaxDocument::get_Subject 속성은 팩스의 표지에 있는 제목 필드의 내용을 포함하는 null로 끝나는 문자열입니다. (Put)
put_Subject Subject 속성에는 인바운드 팩스 메시지와 연결된 제목이 포함됩니다. 이 속성은 null로 끝나는 문자열입니다. (Put)
put_Title IFaxSender::get_Title 속성은 보낸 사람과 연결된 제목을 포함하는 null로 끝나는 문자열입니다. (Put)

put_Tsid FaxDoc 개체의 Tsid 속성을 설정하거나 검색합니다. Tsid 속성은 사용자 정의 TSID(전송 스테이션 식별자)를 포함하는 null로 끝나는 문자열입니다. (Put)
put_Tsid IFaxPort::get_Tsid 속성은 팩스 포트와 연결된 TSID(전송 스테이션 식별자)를 포함하는 null로 끝나는 문자열입니다. (Put)
put_TSID IFaxDevice::get_TSID 속성은 디바이스에 대한 TSID(전송 스테이션 식별자)를 포함하는 null로 끝나는 문자열입니다. (Put)
put_TSID IFaxSender::get_TSID 속성은 보낸 사람의 디바이스에 대한 TSID(전송 스테이션 식별자)를 포함하는 null로 끝나는 문자열입니다. (Put)
put_UseArchive 팩스를 보관해야 하는지 여부를 나타내는 값을 설정하거나 검색합니다. (Put)
put_UseArchive UseArchive 속성은 팩스 서비스가 인바운드 팩스 메시지를 보관하는지 여부를 나타내는 부울 값입니다. (Put)
put_UseArchive IFaxOutgoingArchive::get_UseArchive 속성은 팩스 서비스가 아웃바운드 팩스 메시지를 보관하는지 여부를 나타내는 부울 값입니다. (Put)
put_UseDevice IFaxOutboundRoutingRule::get_UseDevice 속성은 아웃바운드 라우팅 규칙이 단일 팩스 디바이스를 가리키는지 여부를 나타내는 부울 값입니다. (Put)
put_UseDeviceTsid FaxServer 개체의 UseDeviceTsid 속성을 설정하거나 검색합니다. UseDeviceTsid 속성은 팩스 서버가 사용자 지정 TSID 대신 디바이스의 TSID(전송 스테이션 식별자)를 사용하는지 여부를 나타내는 부울 값입니다. (Put)
put_UseDeviceTSID TSID(전송 스테이션 식별자)가 사용되는지 여부를 나타내는 값을 설정하거나 검색합니다. (Put)

put_UseDeviceTSID IFaxOutgoingQueue::get_UseDeviceTSID 속성은 팩스 서비스에서 보낸 사람 TSID 대신 TSID(디바이스 전송 스테이션 식별자)를 사용하는지 여부를 나타내는 부울 값입니다. (Put)
put_UseForInboundRouting IFaxReceiptOptions::get_UseForInboundRouting 속성은 들어오는 팩스를 전자 메일 주소로 라우팅할 수 있도록 하는 Microsoft 라우팅 확장에 대해 FaxReceiptOptions 설정을 사용할지 여부를 설정하거나 검색합니다. (Put)
재할당 하나 이상의 받는 사람에게 팩스를 다시 할당합니다. 또한 IFaxIncomingMessage2::Subject, IFaxIncomingMessage2::SenderName, IFaxIncomingMessage2::SenderFaxNumber 및 IFaxIncomingMessage2::HasCoverPage 속성에 대한 변경 내용을 커밋합니다.
새로 고침 팩스 서버에서 특정 팩스 계정에 대한 FaxAccountIncomingArchive 개체 정보를 새로 고칩니다.
새로 고침 팩스 서버에서 특정 팩스 계정에 대한 FaxAccountOutgoingArchive 개체 정보를 새로 고칩니다.
새로 고침 IFaxActivity::Refresh 메서드는 팩스 서버에서 FaxActivity 정보를 새로 고칩니다.
새로 고침 IFaxActivityLogging::Refresh 메서드는 팩스 서버에서 FaxActivityLogging 개체 정보를 새로 고칩니다.
새로 고침 개체를 새로 고칩니다.
새로 고침 IFaxDevice::Refresh 메서드는 팩스 서버에서 FaxDevice 개체 정보를 새로 고칩니다. IFaxDevice::Refresh 메서드가 호출되면 마지막 IFaxDevice::Save 메서드 호출 후에 수행된 모든 구성 변경 내용이 손실됩니다.

새로 고침 IFaxEventLogging::Refresh 메서드는 팩스 서버에서 IFaxEventLogging 인터페이스 정보를 새로 고칩니다.
새로 고침 IFaxInboundRoutingMethod::Refresh 메서드는 팩스 서버에서 IFaxInboundRoutingMethod 인터페이스 정보를 새로 고칩니다.
새로 고침 Refresh 메서드는 팩스 서버에서 FaxIncomingArchive 개체 정보를 새로 고칩니다. Refresh 메서드가 호출되면 마지막 Save 메서드 호출 후 변경된 구성이 모두 손실됩니다.
새로 고침 Refresh 메서드는 팩스 서버에서 FaxIncomingJob 개체 정보를 새로 고칩니다.
새로 고침 팩스 서버에서 FaxIncomingMessage 개체 정보를 새로 고칩니다.
새로 고침 Refresh 메서드는 팩스 서버에서 FaxIncomingQueue 개체 정보를 새로 고칩니다. Refresh 메서드가 호출되면 마지막 Save 메서드 호출 후 변경된 구성이 모두 손실됩니다.
새로 고침 IFaxOutboundRoutingRule::Refresh 메서드는 팩스 서버에서 FaxOutboundRoutingRule 개체 정보를 새로 고칩니다.
새로 고침 IFaxOutgoingArchive::Refresh 메서드는 팩스 서버에서 FaxOutgoingArchive 개체 정보를 새로 고칩니다.
새로 고침 Refresh 메서드는 팩스 서버에서 FaxOutgoingJob 개체 정보를 새로 고칩니다.
새로 고침 팩스 서버에서 FaxOutgoingMessage 개체 정보를 새로 고칩니다. Refresh 메서드가 호출되면 마지막 Save 메서드 호출 후 변경된 구성이 모두 손실됩니다.

새로 고침 IFaxOutgoingQueue::Refresh 메서드는 팩스 서버에서 FaxOutgoingQueue 개체 정보를 새로 고칩니다. IFaxOutgoingQueue::Refresh 메서드가 호출되면 마지막 IFaxOutgoingQueue::Save 메서드 호출 후에 수행된 모든 구성 변경 내용이 손실됩니다.
새로 고침 IFaxReceiptOptions::Refresh 메서드는 팩스 서버에서 FaxReceiptOptions 개체 정보를 새로 고칩니다. IFaxReceiptOptions::Refresh 메서드가 호출되면 마지막 IFaxReceiptOptions::Save 메서드 호출 후에 수행된 모든 구성 변경 내용이 손실됩니다.
새로 고침 IFaxSecurity::Refresh 메서드는 팩스 서버에서 FaxSecurity 개체 정보를 새로 고칩니다.
새로 고침 팩스 서버에서 FaxSecurity2 개체 정보를 새로 고칩니다.
새로 고침 IFaxJob::Refresh 메서드는 연결된 팩스 작업에 대한 FaxJob 개체 정보를 업데이트합니다.
새로 고침 Refresh 메서드는 연결된 부모 FaxPort 개체에 대한 FaxStatus 개체 정보를 업데이트합니다.
RegisterDeviceProvider IFaxServer::RegisterDeviceProvider 메서드는 팩스 서비스에 FSP(팩스 서비스 공급자)를 등록합니다. 팩스 서비스가 다시 시작되면 등록이 이루어집니다.
RegisterInboundRoutingExtension IFaxServer::RegisterInboundRoutingExtension 메서드는 팩스 서비스에 팩스 인바운드 라우팅 확장을 등록합니다. 팩스 서비스가 다시 시작되면 등록이 이루어집니다.
제거 IFaxDeviceIds::Remove 메서드는 FaxDeviceIds 컬렉션에서 항목을 제거합니다.

제거 Remove 메서드는 FaxOutboundRoutingGroups 컬렉션에서 항목을 제거합니다.
제거 IFaxOutboundRoutingRules::Remove 메서드는 라우팅 규칙의 인덱스를 사용하여 FaxOutboundRoutingRules 컬렉션에서 아웃바운드 라우팅 규칙 (FaxOutboundRoutingRule 개체)을 제거합니다.
제거 IFaxRecipients::Remove 메서드는 FaxRecipients 컬렉션에서 항목을 제거합니다.
RemoveAccount 팩스 서버에서 팩스 계정을 제거합니다.
RemoveByCountryAndArea IFaxOutboundRoutingRules::RemoveByCountryAndArea 메서드는 라우팅 규칙의 국가/지역 코드 및 지역 코드를 사용하여 컬렉션에서 아웃바운드 라우팅 규칙 (FaxOutboundRoutingRule 개체)을 제거합니다.
다시 시작 IFaxOutgoingJob::Restart 메서드는 실패한 아웃바운드 팩스 작업을 다시 시작합니다. 예를 들어 팩스 작업이 재시도 횟수를 초과하면 IFaxOutgoingJob::Restart가 팩스 작업을 다시 시작합니다.
재개 IFaxOutgoingJob::Resume 메서드는 일시 중지된 아웃바운드 팩스 작업을 다시 시작합니다.
저장 IFaxActivityLogging::Save 메서드는 FaxActivityLogging 개체의 데이터를 저장합니다.
저장 개체를 저장합니다.
저장 IFaxDevice::Save 메서드는 FaxDevice 개체의 데이터를 저장합니다.

저장 IFaxEventLogging::Save 메서드는 IFaxEventLogging 인터페이스의 데이터를 저장합니다.
저장 IFaxInboundRoutingMethod::Save 메서드는 IFaxInboundRoutingMethod 인터페이스의 데이터를 저장합니다.
저장 Save 메서드는 FaxIncomingArchive 개체의 데이터를 저장합니다.
저장 FaxIncomingMessage 개체의 데이터를 저장합니다.
저장 Save 메서드는 FaxIncomingQueue 개체의 데이터를 저장합니다.
저장 IFaxOutboundRoutingRule::Save 메서드는 FaxOutboundRoutingRule 개체 데이터를 저장합니다.
저장 IFaxOutgoingArchive::Save 메서드는 FaxOutgoingArchive 개체 데이터를 저장합니다.
저장 FaxOutgoingMessage 개체의 데이터를 저장합니다.
저장 IFaxOutgoingQueue::Save 메서드는 FaxOutgoingQueue 개체 데이터를 저장합니다.
저장 IFaxReceiptOptions::Save 메서드는 FaxReceiptOptions 개체 데이터를 저장합니다.
저장 IFaxSecurity::Save 메서드는 FaxSecurity 개체 데이터를 저장합니다.

저장 FaxSecurity 개체 데이터를 저장합니다.
SaveDefaultSender IFaxSender::SaveDefaultSender 메서드는 FaxSender 개체의 기본 보낸 사람 정보를 저장합니다.
'보내기' Send 메서드는 FaxDoc 개체의 FileName 속성으로 지정된 문서를 전송합니다. 이 메서드는 팩스를 FaxNumber 속성에 지정된 팩스 번호로 보낼 수 있습니다.
SendToFaxRecipient 파일을 팩스로 보내도록 애플리케이션에서 호출합니다.
SetExtensionProperty IFaxDevice::SetExtensionProperty 메서드는 디바이스 수준에서 확장 구성 속성을 저장합니다.
SetExtensionProperty IFaxServer::SetExtensionProperty 메서드는 서버 수준에서 확장 구성 속성을 저장합니다.
SetOrder IFaxDeviceIds::SetOrder 메서드는 정렬된 FaxDeviceIds 컬렉션에서 디바이스의 순서를 변경합니다.
SetStatus IFaxJob::SetStatus 메서드를 호출하여 지정된 팩스 작업을 일시 중지, 다시 시작, 취소 또는 다시 시작합니다.
제출 IFaxDocument::Submit 메서드는 처리를 위해 팩스 서비스에 단일 팩스 문서를 제출합니다.
Submit2 처리를 위해 팩스 서비스에 하나 이상의 문서를 제출합니다. 참고 이 메서드는 Windows Vista 이상에서만 지원됩니다.

UnregisterDeviceProvider IFaxServer::UnregisterDeviceProvider 메서드는 기존 디바이스 공급자의 등록을 등록 취소(제거)합니다. 등록 취소는 팩스 서버를 다시 시작한 후에만 수행됩니다.
UnregisterInboundRoutingExtension IFaxServer::UnregisterInboundRoutingExtension 메서드는 기존 인바운드 라우팅 확장의 등록을 취소합니다. 등록 취소는 팩스 서버를 다시 시작한 후에만 수행됩니다.
UseRoutingMethod IFaxDevice::UseRoutingMethod 메서드는 인바운드 팩스 라우팅 메서드를 팩스 디바이스와 연결된 라우팅 방법 목록에서 팩스 라우팅 메서드(FaxInboundRoutingMethod)에 추가하거나 제거합니다.

인터페이스

_IFaxAccountNotify 특정 팩스 계정에 대한 이벤트 알림을 보내기 위해 팩스 서비스에서 호출됩니다. 이 속성은 이벤트 알림을 보냅니다. 이벤트에는 들어오는 작업 큐 및 나가는 작업 큐에 대한 변경 내용, 들어오는 보관 및 나가는 보관 파일의 변경 내용이 포함됩니다.
_IFaxServerNotify2 IFaxServerNotify2 인터페이스는 팩스 알림에 사용됩니다.
IFaxAccount 팩스 서버의 팩스 계정을 나타냅니다.
IFaxAccountFolders 수신 및 송신 팩스 큐 및 팩스 보관에 대한 액세스를 제공합니다.
IFaxAccountIncomingArchive 팩스 클라이언트 애플리케이션에서 성공적으로 수신된 인바운드 팩스 메시지의 특정 팩스 계정 보관에 액세스하는 데 사용됩니다. 이 인터페이스를 사용하여 메시지를 검색하고 보관 파일의 크기를 가져옵니다.

IFaxAccountIncomingQueue 팩스 클라이언트 애플리케이션에서 특정 팩스 계정에 대한 작업 큐에서 인바운드 팩스 작업(FaxIncomingJobs 개체)을 검색하는 데 사용됩니다.
IFaxAccountNotify 특정 팩스 계정에 대한 이벤트 알림을 보내기 위해 팩스 서비스에서 호출됩니다. 이 속성은 이벤트 알림을 보냅니다. 이벤트에는 들어오는 작업 큐 및 나가는 작업 큐에 대한 변경 내용, 들어오는 보관 및 나가는 보관 파일의 변경 내용이 포함됩니다. (IIFaxAccountNotify)
IFaxAccountOutgoingArchive 팩스 클라이언트 애플리케이션에서 성공적으로 보낸 아웃바운드 팩스 메시지의 지정된 팩스 계정 보관에 액세스하는 데 사용됩니다. 이 인터페이스를 사용하여 메시지를 검색하고 보관 파일의 크기를 가져옵니다.
IFaxAccountOutgoingQueue 팩스 클라이언트 애플리케이션에서 특정 팩스 계정에 대한 작업 큐에서 아웃바운드 팩스 작업(FaxOutgoingJobs 개체)을 검색하는 데 사용됩니다.
IFaxAccounts 팩스 서버의 팩스 계정 컬렉션을 나타냅니다. 계정을 열거하고, 특정 계정을 검색하고, 총 계정 수를 보고하기 위한 메서드와 속성을 제공합니다.
IFaxAccountSet 팩스 계정 추가, 제거 및 검색을 포함하여 팩스 계정 관리를 위한 메서드를 제공합니다.
IFaxActivity IFaxActivity 인터페이스는 읽기 전용 구성 개체를 정의합니다.
IFaxActivityLogging IFaxActivityLogging 인터페이스는 팩스 클라이언트 애플리케이션에서 활동 로깅 옵션을 검색하고 설정하는 데 사용하는 구성 개체를 정의합니다.
IFaxConfiguration 팩스 서비스에 대한 구성 옵션을 제공하는 다양한 메서드를 정의합니다.
IFaxDevice IFaxDevice 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 디바이스 정보를 검색 및 설정하고 팩스 디바이스와 연결된 팩스 라우팅 메서드를 추가 및 제거하기 위해 사용하는 구성 개체를 정의합니다.

IFaxDeviceIds IFaxDeviceIds 인터페이스는 팩스 클라이언트 애플리케이션에서 FaxOutboundRoutingGroup 개체와 연결된 정렬된 팩스 디바이스 ID를 열거하는 데 사용하는 구성 컬렉션을 정의합니다.
IFaxDeviceProvider IFaxDeviceProvider 인터페이스는 팩스 서비스에 등록된 FSP(팩스 서비스 공급자)에 대한 정보를 검색하기 위해 팩스 클라이언트 애플리케이션에서 사용하는 구성 개체를 정의합니다.
IFaxDeviceProviders IFaxDeviceProviders 인터페이스는 연결된 팩스 서버의 팩스 디바이스 공급자를 포함하는 구성 컬렉션을 정의합니다.
IFaxDevices IFaxDevices 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 디바이스를 관리하는 데 사용하는 컬렉션을 정의합니다. 여기서 각 디바이스는 FaxDevice 개체로 표시됩니다.
IFaxDoc IFaxDoc 이중 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 문서 및 표지를 전송하는 데 사용됩니다.
IFaxDocument IFaxDocument 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 문서를 작성하고 처리를 위해 팩스 서비스에 제출하는 데 사용하는 메시징 개체를 정의합니다.
IFaxDocument2 팩스 클라이언트 애플리케이션에서 팩스 문서를 작성하고 처리를 위해 팩스 서비스에 제출하는 데 사용하는 메시징 개체를 정의합니다.
IFaxEventLogging IFaxEventLogging 인터페이스는 팩스 서비스에서 사용하는 이벤트 로깅 범주를 구성하기 위해 팩스 클라이언트 애플리케이션에서 사용하는 구성 개체를 정의합니다.
IFaxFolders IFaxFolders 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 서버의 폴더, 대기 중인 작업 및 보관된 메시지에 액세스하는 데 사용하는 구성 개체를 정의합니다.

IFaxInboundRouting IFaxInboundRouting 인터페이스는 팩스 서비스에 등록된 인바운드 라우팅 확장에 액세스하기 위해 팩스 클라이언트 애플리케이션에서 사용하는 구성 개체를 정의하며, FaxInboundRoutingExtensions 개체로 표현되며 FaxInboundRoutingMethods 개체로 표현되는 확장이 노출하는 라우팅 메서드를 정의합니다.
IFaxInboundRoutingExtension IFaxInboundRoutingExtension 인터페이스는 팩스 서비스에 등록된 팩스 라우팅 확장 프로그램에 대한 정보를 검색하기 위해 팩스 클라이언트 애플리케이션에서 사용하는 구성 개체를 정의합니다.
IFaxInboundRoutingExtensions IFaxInboundRoutingExtensions 인터페이스는 팩스 서비스에 등록된 인바운드 팩스 라우팅 확장을 관리하기 위해 팩스 클라이언트 애플리케이션에서 사용하는 구성 컬렉션을 정의합니다.
IFaxInboundRoutingMethod IFaxInboundRoutingMethod 인터페이스는 팩스 클라이언트 애플리케이션에서 연결된 팩스 서버의 개별 팩스 인바운드 라우팅 방법에 대한 정보를 검색하는 데 사용하는 구성 개체를 정의합니다.
IFaxInboundRoutingMethods IFaxInboundRoutingMethods 인터페이스는 팩스 클라이언트 애플리케이션에서 정렬된 인바운드 팩스 라우팅 메서드를 관리하는 데 사용하는 구성 컬렉션을 정의합니다.
IFaxIncomingArchive IFaxIncomingArchive 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 서비스에서 수신한 인바운드 팩스 메시지의 보관에 액세스하고 구성하는 데 사용됩니다.
IFaxIncomingJob IFaxIncomingJob 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 서버 큐에서 들어오는 팩스 작업에 대한 정보를 검색하는 데 사용됩니다.
IFaxIncomingJobs IFaxIncomingJobs 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 서버의 작업 큐에서 인바운드 팩스 작업을 관리하는 데 사용됩니다. 들어오는 각 작업은 FaxIncomingJob 개체로 표시됩니다.
IFaxIncomingMessage 팩스 클라이언트 애플리케이션에서 인바운드 팩스 보관 파일에서 받은 팩스 메시지에 대한 정보를 검색하는 데 사용됩니다. (IFaxIncomingMessage)

IFaxIncomingMessage2 팩스 클라이언트 애플리케이션에서 인바운드 팩스 보관 파일에서 받은 팩스 메시지에 대한 정보를 검색하는 데 사용됩니다. (IFaxIncomingMessage2)
IFaxIncomingMessageIterator IFaxIncomingMessageIterator 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 서비스가 성공적으로 수신한 인바운드 팩스 메시지의 보관을 통해 이동하는 데 사용됩니다.
IFaxIncomingQueue IFaxIncomingQueue 인터페이스는 팩스 클라이언트 애플리케이션에서 작업 큐의 인바운드 팩스 작업(FaxIncomingJobs 개체)을 관리하는 데 사용됩니다. 또한 이 개체에는 팩스 작업 큐에서 인바운드 팩스를 차단하는 메서드도 포함되어 있습니다.
IFaxJob IFaxJob 이중 인터페이스는 팩스 클라이언트 애플리케이션에서 연결된 팩스 서버의 팩스 작업에 대한 정보에 액세스하는 데 사용됩니다. FaxJobs 개체는 FaxJob 개체의 컬렉션입니다.
IFaxJobs IFaxJobs 이중 인터페이스는 팩스 클라이언트 애플리케이션에서 FaxServer 개체에서 파생된 FaxJob 개체에 액세스하는 데 사용됩니다. 인터페이스는 연결된 팩스 서버와 연결된 팩스 작업을 열거합니다.
IFaxJobStatus IFaxJobStatus 인터페이스는 알림 및 작업의 동적 정보를 유지하는 데 사용됩니다.
IFaxLoggingOptions IFaxLoggingOptions 인터페이스는 팩스 클라이언트 애플리케이션에서 이벤트 로깅 범주 및 팩스 서비스에서 사용하는 활동 로깅 옵션에 액세스하고 구성하는 데 사용됩니다.
IFaxOutboundRouting IFaxOutboundRouting 인터페이스는 팩스 클라이언트 애플리케이션에서 아웃바운드 라우팅 그룹(IFaxOutboundRoutingGroups 인터페이스) 및 아웃바운드 라우팅 규칙(IFaxOutboundRoutingRules 인터페이스)을 구성하는 데 사용되는 구성 개체를 정의합니다.
IFaxOutboundRoutingGroup IFaxOutboundRoutingGroup 인터페이스는 팩스 클라이언트 애플리케이션에서 개별 팩스 아웃바운드 라우팅 그룹에 대한 정보를 검색하는 데 사용하는 구성 개체를 설명합니다.

IFaxOutboundRoutingGroups IFaxOutboundRoutingGroups 인터페이스는 팩스 클라이언트 애플리케이션에서 IFaxOutboundRoutingGroup 인터페이스로 표시되는 팩스 아웃바운드 라우팅 그룹을 관리하는 데 사용하는 구성 컬렉션을 설명합니다.
IFaxOutboundRoutingRule IFaxOutboundRoutingRule 인터페이스는 팩스 클라이언트 애플리케이션에서 개별 팩스 아웃바운드 라우팅 규칙에 대한 정보를 설정하고 검색하는 데 사용하는 구성 개체를 설명합니다.
IFaxOutboundRoutingRules IFaxOutboundRoutingRules 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 아웃바운드 라우팅 규칙을 관리하는 데 사용되는 구성 컬렉션을 설명합니다.
IFaxOutgoingArchive IFaxOutgoingArchive 인터페이스는 팩스 서비스에서 성공적으로 전송된 아웃바운드 팩스 메시지의 보관 파일에 액세스하고 구성하기 위해 팩스 클라이언트 애플리케이션에서 사용하는 구성 개체에 대해 설명합니다.
IFaxOutgoingJob IFaxOutgoingJob 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 서버 큐에서 보내는 팩스 작업에 대한 정보를 검색하는 데 사용되는 개체를 설명합니다.
IFaxOutgoingJob2 팩스 클라이언트 애플리케이션에서 팩스 서버 큐에서 보내는 팩스 작업에 대한 정보를 검색하는 데 사용되는 개체에 대해 설명합니다.
IFaxOutgoingJobs IFaxOutgoingJobs 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 서버의 작업 큐에서 아웃바운드 팩스 작업을 관리하는 데 사용되는 메시징 컬렉션을 설명합니다. 각 아웃바운드 작업은 IFaxOutgoingJob 인터페이스로 표시됩니다.
IFaxOutgoingMessage IFaxOutgoingMessage 인터페이스는 팩스 클라이언트 애플리케이션에서 아웃바운드 팩스 보관 파일의 팩스 메시지에 대한 정보를 검색하는 데 사용되는 개체를 설명합니다.
IFaxOutgoingMessage2 팩스 클라이언트 애플리케이션에서 아웃바운드 팩스 보관 파일에서 보낸 팩스 메시지에 대한 정보를 검색하는 데 사용됩니다.

IFaxOutgoingMessageIterator IFaxOutgoingMessageIterator 인터페이스는 팩스 서비스가 성공적으로 전송한 팩스 메시지의 보관을 이동하기 위해 팩스 클라이언트 애플리케이션에서 사용하는 개체를 설명하며, FaxOutgoingMessage 개체로 표시됩니다.
IFaxOutgoingQueue IFaxOutgoingQueue 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 서버의 아웃바운드 팩스 큐에서 구성 매개 변수를 설정하고 검색하는 데 사용하는 FaxOutgoingQueue 구성 개체를 정의합니다.
IFaxPort IFaxPort 이중 인터페이스는 팩스 클라이언트 애플리케이션에서 연결된 팩스 서버의 팩스 포트에 대한 구성 정보에 액세스하는 데 사용됩니다.
IFaxPorts IFaxPorts 이중 인터페이스는 팩스 클라이언트 애플리케이션에서 FaxServer 개체에서 파생된 FaxPort 개체에 액세스하는 데 사용됩니다. 인터페이스는 활성 팩스 서버에 대한 연결에 대한 포트 구성 정보를 열거합니다.
IFaxReceiptOptions IFaxReceiptOptions 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 전송에 대한 배달 영수증을 보내는 데 사용하는 영수증 구성을 설정하고 검색하는 데 사용하는 FaxReceiptOptions 구성 개체를 정의합니다.
IFaxRecipient IFaxRecipient 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 수신자에 대한 개인 정보를 검색하고 설정하는 데 사용되는 FaxRecipient 메시징 개체를 정의합니다.
IFaxRecipients IFaxRecipients 인터페이스는 팩스 클라이언트 애플리케이션에서 단일 팩스 문서의 받는 사람을 나타내는 팩스 수신자 개체(FaxRecipient)를 관리하는 데 사용되는 FaxRecipients 메시징 컬렉션을 정의합니다.
IFaxRoutingMethod IFaxRoutingMethod 이중 인터페이스는 팩스 클라이언트 애플리케이션에서 연결된 팩스 서버의 팩스 포트에 대한 팩스 라우팅 구성 정보를 검색하는 데 사용됩니다.

IFaxRoutingMethods IFaxRoutingMethods 이중 인터페이스는 팩스 클라이언트 애플리케이션에서 FaxPort 개체에서 파생된 FaxRoutingMethod 개체에 액세스하는 데 사용됩니다.
IFaxSecurity IFaxSecurity 구성 개체는 팩스 클라이언트 애플리케이션에서 팩스 서버의 보안을 구성하는 데 사용되며 호출 애플리케이션이 팩스 서버에 대한 보안 설명자를 설정하고 검색하도록 허용합니다.
IFaxSecurity2 팩스 클라이언트 애플리케이션에서 팩스 서버의 보안을 구성하는 데 사용됩니다. 또한 호출 애플리케이션이 팩스 서버에 대한 보안 설명자를 설정하고 검색할 수 있도록 허용합니다.
IFaxSender IFaxSender 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 보낸 사람 정보를 검색하고 설정하는 데 사용하는 메시징 개체를 정의합니다. 또한 이 개체에는 보낸 사람 데이터를 저장하고 로컬 레지스트리에서 보낸 사람 데이터를 검색하는 메서드도 포함됩니다.
IFaxServer IFaxServer 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 서비스에 대한 연결을 관리하는 데 사용되는 메시징 컬렉션을 설명합니다.
IFaxServer IFaxServer 이중 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 서비스에 대한 연결을 관리하는 데 사용됩니다.
IFaxServer2 IFaxServer 인터페이스의 모든 기능을 상속하고 서버 구성, 계정 관리, 보안 개체 및 현재 계정에 대한 읽기 전용 속성을 추가합니다.
IFaxServerNotify2 IFaxServerNotify2 인터페이스는 팩스 알림에 사용됩니다. (IIFaxServerNotify2)
IFaxStatus FaxStatus 이중 인터페이스는 팩스 클라이언트 애플리케이션에서 연결된 팩스 서버의 특정 포트에 대한 상태 정보를 검색하는 데 사용됩니다.
IFaxTiff IFaxTiff 이중 인터페이스는 팩스 클라이언트 애플리케이션에서 FaxTiff 개체에 대한 정보를 검색하는 데 사용됩니다.

구조체

FAX_CONFIGURATIONA FAX_CONFIGURATION 구조에는 팩스 서버의 전역 구성 설정에 대한 정보가 포함되어 있습니다. (ANSI)
FAX_CONFIGURATIONW FAX_CONFIGURATION 구조에는 팩스 서버의 전역 구성 설정에 대한 정보가 포함되어 있습니다. (유니코드)
FAX_CONTEXT_INFOA FAX_CONTEXT_INFO 구조에는 팩스 프린터 디바이스 컨텍스트에 대한 정보가 포함되어 있습니다. SizeOfStruct 멤버가 필요합니다. 다른 멤버에 대한 정보는 FaxStartPrintJob 함수에 대한 호출을 통해 제공됩니다. (ANSI)
FAX_CONTEXT_INFOW FAX_CONTEXT_INFO 구조에는 팩스 프린터 디바이스 컨텍스트에 대한 정보가 포함되어 있습니다. SizeOfStruct 멤버가 필요합니다. 다른 멤버에 대한 정보는 FaxStartPrintJob 함수에 대한 호출을 통해 제공됩니다. (유니코드)
FAX_COVERPAGE_INFOA FAX_COVERPAGE_INFO 구조에는 팩스 전송의 표지에 표시할 데이터가 포함되어 있습니다. SizeOfStruct 및 CoverPageName 멤버가 필요합니다. 다른 멤버는 선택 사항입니다. (ANSI)
FAX_COVERPAGE_INFOW FAX_COVERPAGE_INFO 구조에는 팩스 전송의 표지에 표시할 데이터가 포함되어 있습니다. SizeOfStruct 및 CoverPageName 멤버가 필요합니다. 다른 멤버는 선택 사항입니다. (유니코드)
FAX_DEV_STATUS FAX_DEV_STATUS 구조에는 개별 활성 팩스 작업에 대한 상태 및 식별 정보가 포함됩니다.
FAX_DEVICE_STATUSA FAX_DEVICE_STATUS 구조에는 팩스 장치의 현재 상태에 대한 정보가 포함됩니다. (ANSI)
FAX_DEVICE_STATUSW FAX_DEVICE_STATUS 구조에는 팩스 장치의 현재 상태에 대한 정보가 포함됩니다. (유니코드)

FAX_EVENTA FAX_EVENT 구조체는 I/O 완료 패킷의 내용을 나타냅니다. 팩스 서버는 완료 패킷을 전송하여 팩스 클라이언트 애플리케이션에 비동기 팩스 서버 이벤트를 알립니다. (ANSI)
FAX_EVENTW FAX_EVENT 구조체는 I/O 완료 패킷의 내용을 나타냅니다. 팩스 서버는 완료 패킷을 전송하여 팩스 클라이언트 애플리케이션에 비동기 팩스 서버 이벤트를 알립니다. (유니코드)
FAX_GLOBAL_ROUTING_INFOA FAX_GLOBAL_ROUTING_INFO 구조에는 팩스 서비스와 전 세계적으로 관련된 하나의 팩스 라우팅 방법에 대한 정보가 포함됩니다. (ANSI)
FAX_GLOBAL_ROUTING_INFOW FAX_GLOBAL_ROUTING_INFO 구조에는 팩스 서비스와 전 세계적으로 관련된 하나의 팩스 라우팅 방법에 대한 정보가 포함됩니다. (유니코드)
FAX_JOB_ENTRYA FAX_JOB_ENTRY 구조는 하나의 팩스 작업을 설명합니다. (ANSI)
FAX_JOB_ENTRYW FAX_JOB_ENTRY 구조는 하나의 팩스 작업을 설명합니다. (유니코드)
FAX_JOB_PARAMA FAX_JOB_PARAM 구조에는 팩스 서버에서 개별 팩스 전송을 보내는 데 필요한 정보가 포함되어 있습니다. (ANSI)
FAX_JOB_PARAMW FAX_JOB_PARAM 구조에는 팩스 서버에서 개별 팩스 전송을 보내는 데 필요한 정보가 포함되어 있습니다. (유니코드)
FAX_LOG_CATEGORYA FAX_LOG_CATEGORY 구조는 하나의 로깅 범주를 설명합니다. (ANSI)
FAX_LOG_CATEGORYW FAX_LOG_CATEGORY 구조는 하나의 로깅 범주를 설명합니다. (유니코드)

FAX_PORT_INFOA FAX_PORT_INFO 구조는 하나의 팩스 포트를 설명합니다. 데이터에는 디바이스 식별자, 포트 이름 및 현재 상태, 스테이션 식별자가 포함됩니다. (ANSI)
FAX_PORT_INFOW FAX_PORT_INFO 구조는 하나의 팩스 포트를 설명합니다. 데이터에는 디바이스 식별자, 포트 이름 및 현재 상태, 스테이션 식별자가 포함됩니다. (유니코드)
FAX_PRINT_INFOA FAX_PRINT_INFO 구조에는 팩스 서버에서 팩스 전송을 인쇄하는 데 필요한 정보가 포함되어 있습니다. 이 구조에는 보낸 사람 및 받는 사람 데이터, 선택적 청구 코드 및 배달 보고서 정보가 포함됩니다. (ANSI)
FAX_PRINT_INFOW FAX_PRINT_INFO 구조에는 팩스 서버에서 팩스 전송을 인쇄하는 데 필요한 정보가 포함되어 있습니다. 이 구조에는 보낸 사람 및 받는 사람 데이터, 선택적 청구 코드 및 배달 보고서 정보가 포함됩니다. (유니코드)
FAX_RECEIVE FAX_RECEIVE 구조에는 인바운드 팩스 문서에 대한 정보가 포함됩니다. 이 정보에는 팩스 데이터 스트림을 받을 파일의 이름과 수신 디바이스의 이름 및 전화 번호가 포함됩니다.
FAX_ROUTE FAX_ROUTE 구조에는 받은 팩스 문서에 대한 정보가 포함됩니다. 팩스 서비스는 FaxRouteMethod 함수에 대한 호출에서 팩스 라우팅 메서드에 구조를 전달합니다.
FAX_ROUTE_CALLBACKROUTINES FAX_ROUTE_CALLBACKROUTINES 구조체에는 팩스 서비스에서 제공하는 콜백 함수에 대한 포인터가 포함되어 있습니다.
FAX_ROUTING_METHODA FAX_ROUTING_METHOD 구조에는 하나의 팩스 디바이스와 관련된 하나의 팩스 라우팅 방법에 대한 정보가 포함됩니다. (ANSI)
FAX_ROUTING_METHODW FAX_ROUTING_METHOD 구조에는 하나의 팩스 디바이스와 관련된 하나의 팩스 라우팅 방법에 대한 정보가 포함됩니다. (유니코드)

FAX_SEND FAX_SEND 구조에는 아웃바운드 팩스 문서에 대한 정보가 포함됩니다.
FAX_TIME FAX_TIME 구조체는 현재 시간 및 분 동안 개별 멤버를 사용하여 시간을 나타냅니다. 시간은 UTC(협정 세계시)로 표현됩니다.

File Server Resource Manager

아티클2023. 03. 09.

파일 서버 Resource Manager 기술 개요입니다.

파일 서버 Resource Manager 개발하려면 다음 헤더가 필요합니다.

- fsrm.h
- fsrmenums.h
- fsrmpipeline.h
- fsrmquota.h
- fsrmreports.h
- fsrmscreen.h
- fsrmtlb.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 파일 서버 리소스 관리자

열거형

AdrClientDisplayFlags 가능한 ADR(액세스 거부 수정) 클라이언트 표시 플래그 유형에 대해 설명합니다.
AdrClientErrorType 가능한 액세스 거부 수정(ADR) 클라이언트 오류 유형에 대해 설명합니다.
AdrClientFlags ADR(액세스 거부 수정) 클라이언트 작업을 수행할 수 없는 이유를 나타내는 플래그를 열거합니다.
AdrEmailFlags ADR(액세스 거부 수정) 전자 메일에 대한 옵션을 설명합니다.
FsrmAccountType 명령 작업(FsrmActionType 참조)을 실행할 수 있는 컴퓨터 계정 유형을 정의합니다.

FsrmActionType 할당량 또는 파일 화면 이벤트에 대한 응답으로 트리거할 수 있는 작업을 정의합니다(예: 할당량이 초과되거나 파일이 파일 화면을 위반하는 경우).
FsrmClassificationLoggingFlags 분류를 실행하는 동안 정보를 로깅하는 다양한 옵션을 정의합니다.
FsrmCollectionState 컬렉션 개체의 가능한 상태를 정의합니다.
FsrmCommitOptions 개체 컬렉션을 커밋하는 옵션을 정의합니다.
FsrmEnumOptions 개체 컬렉션을 열거하는 옵션을 정의합니다.
FsrmEventType 이벤트 로깅 작업(FsrmActionType 참조)이 기록할 수 있는 이벤트 유형을 정의합니다.
FsrmExecutionOption 파일에 규칙을 적용하는 방법에 대한 옵션을 정의합니다.
FsrmFileManagementLoggingFlags 파일 관리 작업을 실행할 때 로깅 옵션을 정의합니다.
FsrmFileManagementType 파일 관리 작업 유형을 정의합니다.
FsrmFileScreenFlags 파일 화면을 위반하는 실패한 IO 작업에 대한 옵션을 정의합니다.
FsrmFileStreamingInterfaceType 가능한 스트리밍 인터페이스 형식을 정의합니다.
FsrmFileStreamingMode 파일 스트림에 사용할 스트리밍 모드를 정의합니다.

FsrmFileSystemPropertyId 파일 시스템 속성 ID의 가능한 형식을 정의합니다.
FsrmGetFilePropertyOptions 파일과 연결된 분류 속성을 검색하는 방법을 정의하는 플래그입니다.
FsrmPipelineModuleType 정의할 수 있는 모듈 유형을 정의합니다.
FsrmPropertyBagField 속성 모음의 유형을 설명합니다.
FsrmPropertyBagFlags 속성 모음에 대한 추가 정보를 제공하는 플래그 값을 정의합니다.
FsrmPropertyConditionType 파일의 속성 값이 특정 조건을 충족하는지 여부를 확인하는 데 사용할 수 있는 가능한 비교 작업을 정의합니다.
FsrmPropertyDefinitionAppliesTo FSRM 분류 속성을 적용할 수 있는 것을 나타내는 플래그입니다.
FsrmPropertyDefinitionFlags 분류 속성의 형식을 설명하는 플래그입니다.
FsrmPropertyDefinitionType 정의할 수 있는 파일 분류 속성의 형식을 정의합니다.
FsrmPropertyFlags 분류 속성에 대한 추가 정보를 제공하는 플래그 값을 정의합니다.
FsrmPropertyValueType 속성 조건에서 FSRM 속성에 할당되는 값의 형식을 열거합니다.
FsrmQuotaFlags 할당량을 위반하고, 할당량 추적을 사용하거나 사용하지 않도록 설정하고, 할당량 검색 작업의 상태를 제공하는 IO 작업 실패에 대한 옵션을 정의합니다.

FsrmReportFilter 보고서에 포함된 파일을 제한하는 데 사용할 수 있는 필터를 정의합니다.
FsrmReportFormat 보고서를 생성할 때 사용할 수 있는 파일 형식을 정의합니다.
FsrmReportGenerationContext 보고서가 시작되는 컨텍스트를 정의합니다.
FsrmReportLimit 보고서에 포함된 파일을 제한하는 데 사용되는 제한을 정의합니다.
FsrmReportRunningStatus 보고서 작업에 대한 실행 상태를 정의합니다.
FsrmReportType 생성할 수 있는 보고서 유형을 정의합니다.
FsrmRuleFlags 규칙의 가능한 상태를 정의합니다.
FsrmRuleType 정의할 수 있는 규칙 유형을 정의합니다.
FsrmStorageModuleCaps 스토리지 모듈의 기능을 정의하는 플래그입니다.
FsrmStorageModuleType 가능한 스토리지 모듈 유형을 정의합니다.
FsrmTemplateApplyOptions 파생 개체에 템플릿 변경 내용을 적용하는 옵션을 정의합니다.

Functions

추가 컬렉션에 개체를 추가합니다. (IFsrmMutableCollection.Add)
AddMessage 모음에 오류 메시지를 추가합니다.
AddNotification 파일 관리 작업의 알림 목록에 새 알림 값(기간)을 추가합니다.
AddThreshold 할당량 개체에 임계값을 추가합니다.
Applytemplate 지정된 할당량 템플릿의 속성 값을 이 할당량 개체에 적용합니다.
Applytemplate 지정된 파일 화면 템플릿의 속성 값을 이 파일 화면 개체에 적용합니다.
바인딩 파이프라인 모듈 구현을 FSRM 통신 채널에 바인딩합니다.
취소 개체가 비동기적으로 수집될 때 개체 컬렉션을 취소합니다.
취소 작업이 실행 중인 경우 작업을 취소합니다.
취소 이 보고서 작업에 대한 실행 중인 보고서를 취소합니다.
CancelClassification 실행 중인 경우 분류를 취소합니다.
ClassifyFiles 이 메서드는 파일 속성의 대량 열거, 설정 및 지우기를 수행하는 데 사용됩니다.

ClearFileProperty 파일 또는 폴더에서 지정된 속성을 제거하려고 시도합니다.
복제 중복된 IFsrmMutableCollection 컬렉션을 만듭니다.
커밋 컬렉션의 모든 개체를 커밋하고 각 개체에 대한 커밋 결과를 반환합니다.
커밋 개체를 서버의 개체 목록에 저장합니다.
CommitAndUpdateDerived 할당량을 저장한 다음 파생 할당량에 변경 내용을 적용합니다.
CommitAndUpdateDerived 할당량 템플릿을 저장한 다음 파생 할당량 개체에 변경 내용을 적용합니다.
CommitAndUpdateDerived 파일 화면 템플릿을 저장한 다음 파생 파일 화면 개체에 변경 내용을 적용합니다.
CopyTemplate 지정된 템플릿의 속성 값을 이 템플릿에 복사합니다. (IFsrmQuotaTemplate.CopyTemplate)
CopyTemplate 지정된 템플릿의 속성 값을 이 템플릿에 복사합니다. (IFsrmFileScreenTemplate.CopyTemplate)
CreateAction 이 파일 화면 개체에 대한 작업을 만듭니다.
CreateAutoApplyQuota 지정된 디렉터리에 대한 자동 할당량을 만듭니다.
CreateCustomAction 사용자 지정 작업 개체를 만듭니다.

CreateFileGroup 파일 그룹 개체를 만듭니다.
CreateFileManagementJob 파일 관리 작업을 만듭니다.
CreateFileScreen 파일 화면 개체를 만듭니다.
CreateFileScreenCollection 파일 화면을 추가할 수 있는 빈 컬렉션을 만듭니다.
CreateFileScreenException 파일 화면 예외 개체를 만듭니다.
CreateModuleDefinition 지정된 형식의 모듈 정의를 만듭니다.
CreateNotificationAction 알림 작업을 만들고 알림 값과 연결합니다.
CreatePropertyCondition 새 속성 조건을 만들고 속성 조건 컬렉션에 추가합니다.
CreatePropertyDefinition 속성 정의를 만듭니다.
CreateQuota 지정된 디렉터리에 대한 할당량을 만듭니다.
CreateQuotaCollection 할당량을 추가할 수 있는 빈 컬렉션을 만듭니다.
CreateReport 지정된 형식의 새 보고서 개체를 만듭니다.

CreateReportJob 보고서 작업을 만듭니다.
CreateRule 지정된 형식의 규칙을 만듭니다.
CreateScheduleTask 보고서 작업을 트리거하는 데 사용되는 예약된 작업을 만듭니다.
CreateTemplate 할당량 템플릿 개체를 만듭니다.
CreateTemplate 파일 화면 템플릿 개체를 만듭니다.
CreateThresholdAction 작업을 만들고 지정된 임계값에 연결합니다.
삭제 할당량 또는 파일 화면의 작업 목록에서 작업을 제거합니다.
삭제 서버의 개체 목록에서 개체를 제거합니다.
삭제 파일 관리 작업에 대해 지정된 속성 조건 컬렉션에서 이 속성 조건을 제거합니다.
삭제 보고서 작업 개체에서 이 보고서 개체를 제거합니다.
DeleteNotification 파일 관리 작업의 알림 목록에서 알림 값을 삭제합니다.
DeleteScheduleTask 보고서 작업을 트리거하는 데 사용되는 작업을 삭제합니다.

DeleteThreshold 할당량 개체에서 임계값을 삭제합니다.
DoesPropertyValueApply 분류자를 쿼리하여 지정된 속성 값이 IFsrmClassifierModuleImplementation::OnBeginFile 메서드에 의해 가장 최근에 지정된 파일에 적용되는지 여부를 확인합니다.
EmailTest 지정된 전자 메일 주소로 전자 메일 메시지를 보냅니다.
EnumActions 파일 화면 개체에 대한 모든 작업을 열거합니다.
EnumAutoApplyQuotas 지정된 디렉터리에 연결된 자동 할당량을 열거합니다.
EnumEffectiveQuotas 지정된 경로에 영향을 주는 모든 할당량을 열거합니다.
EnumFileGroups FSRM에서 파일 그룹을 열거합니다.
EnumFileManagementJobs 기존 파일 관리 작업 목록을 열거합니다.
EnumFileProperties 지정된 파일의 속성을 열거합니다.
EnumFileScreenExceptions 지정된 디렉터리와 해당 하위 디렉터리에 대한 파일 화면 예외를 열거합니다.
EnumFileScreens 지정된 디렉터리와 해당 하위 디렉터리에 대한 파일 화면을 열거합니다.
EnumModuleDefinitions 지정된 형식의 모듈 정의를 열거합니다.

EnumNotificationActions 알림 값과 연결된 작업을 열거합니다.
EnumPropertyDefinitions 속성 정의를 열거합니다.
EnumQuotas 지정된 디렉터리에 대한 할당량과 해당 하위 디렉터리와 연결된 모든 할당량을 열거합니다(재귀적으로).
EnumReportJobs 보고서 작업을 열거합니다.
EnumReports 이 보고서 작업에 대해 구성된 모든 보고서를 열거합니다.
EnumRules 지정된 형식의 규칙을 열거합니다.
EnumTemplates 서버에서 할당량 템플릿을 열거합니다.
EnumTemplates 서버에서 파일 화면 템플릿을 열거합니다.
EnumThresholdActions 지정된 임계값에 대한 모든 작업을 열거합니다.
ExportFileGroups 하나 이상의 파일 그룹을 지정된 파일로 내보냅니다.
ExportFileGroups 지정된 파일 그룹을 XML 문자열로 내보냅니다.
ExportFileScreenTemplates 하나 이상의 파일 화면 템플릿을 지정된 파일로 내보냅니다.

ExportQuotaTemplates 하나 이상의 할당량 템플릿을 지정된 파일로 내보냅니다.
ExportTemplates 할당량 템플릿을 XML 문자열로 내보냅니다.
ExportTemplates 템플릿을 XML 문자열로 내보냅니다.
get__NewEnum 컬렉션의 항목에 대한 새 IEnumVARIANT 열거형의 IUnknown 포인터를 검색합니다.
get_Account ExecutablePath 속성에 지정된 실행 프로그램을 실행하는 데 사용되는 시스템 계정을 검색하거나 설정합니다. (Get)
get_Account 모듈을 실행할 때 사용할 계정입니다. (Get)
get_ActionType 작업의 형식을 검색합니다.
get_ActionVariableDescriptions IFsrmQuotaManager::ActionVariables 속성에 포함된 매크로에 대한 설명을 검색합니다.
get_ActionVariableDescriptions IFsrmFileManagementJobManager::ActionVariables 속성에 포함된 매크로에 대한 설명을 검색합니다.
get_ActionVariableDescriptions IFsrmFileScreenManager::ActionVariables 속성에 포함된 매크로에 대한 설명을 검색합니다.
get_ActionVariables 작업 속성 값에서 지정할 수 있는 매크로 목록을 검색합니다. (IFsrmQuotaManagerEx.get_ActionVariables)

get_ActionVariables 작업 속성 값에서 지정할 수 있는 매크로 목록을 검색합니다. (IFsrmFileManagementJobManager.get_ActionVariables)
get_ActionVariables 작업 속성 값에서 지정할 수 있는 매크로 목록을 검색합니다. (IFsrmFileScreenManager.get_ActionVariables)
get_AdminEmail 관리자의 전자 메일 주소를 검색하거나 설정합니다. (Get)
get_AllowedFileGroups 디렉터리에 허용되는 파일의 파일 이름 패턴을 포함하는 파일 그룹의 이름을 검색하거나 설정합니다. (Get)
get_AppliesTo 이 속성에는 FSRM 속성 정의를 적용할 수 있는 항목을 나타내는 FsrmPropertyDefinitionAppliesTo 열거형의 값이 있는 플래그가 포함됩니다.
get_Arguments ExecutablePath 속성에 지정된 실행 프로그램에 전달할 인수를 검색하거나 설정합니다. (Get)
get_AttachmentFileListSize 목록에 포함할 최대 파일 수입니다. (Get)
get_Attributes 파일의 특성입니다.
get_BlockedFileGroups 이 화면에서 차단된 파일을 지정하는 데 사용되는 파일 이름 패턴을 포함하는 파일 그룹의 이름을 검색하거나 설정합니다. (Get)
get_Capabilities 스토리지 모듈의 기능을 지정하는 플래그입니다. (Get)
get_ClassificationLastError 분류가 마지막으로 실행된 시점의 오류 메시지입니다.

get_ClassificationLastReportPathWithoutExtension 분류가 마지막으로 실행되었을 때 보고서가 저장된 로컬 디렉터리 경로입니다.
get_ClassificationReportEnabled 분류 보고를 사용할 수 있는지 여부를 결정합니다. (Get)
get_ClassificationReportFormats 분류 보고서를 생성할 형식 목록입니다. (Get)
get_ClassificationReportMailTo 분류 보고서를 보낼 이메일 주소(있는 경우)입니다. (Get)
get_ClassificationRunningStatus 분류의 실행 상태입니다.
get_Company 분류 또는 스토리지 모듈을 만든 회사의 이름입니다. (Get)
get_Count 컬렉션의 항목 수를 검색합니다. (IFsrmCollection.get_Count)
get_CreationTime 파일을 만든 날짜와 시간입니다.
get_CustomAction 모든 조건이 충족될 때 실행할 작업입니다.
get_DaysSinceFileCreated 파일을 만든 후 경과된 일 수입니다. (Get)
get_DaysSinceFileLastAccessed 파일에 마지막으로 액세스한 후 경과된 일 수입니다. (Get)
get_DaysSinceFileLastModified 파일이 마지막으로 수정된 후 경과된 일 수입니다. (Get)

get_DerivedObjects 업데이트된 파생 개체의 컬렉션을 검색합니다.
get_Description 개체에 대한 설명을 검색하거나 설정합니다. (Get)
get_Description 분류 속성 정의 값에 대한 설명입니다.
get_Description 보고서에 대한 설명을 검색하거나 설정합니다. (Get)
get_DisableCommandLine FSRM이 명령줄 동작의 실행을 차단하는지 여부를 결정하는 값을 검색하거나 설정합니다. (Get)
get_DisplayName 속성 정의의 표시 이름입니다. (Get)
get_DisplayName 분류 속성 정의 값의 표시 이름입니다.
get_Enabled 모듈을 사용할 수 있는지 여부를 결정합니다. (Get)
get_Enabled 작업을 사용할 수 있는지 여부를 나타냅니다(실행할 수 있습니다). (Get)
get_EnableScreeningAudit FSRM이 파일 화면 위반에 대한 감사 레코드를 유지하는지 여부를 결정하는 값을 검색하거나 설정합니다. (Get)
get_EventType 실행 시 동작이 기록하는 이벤트 유형을 검색하거나 설정합니다. (Get)
get_ExcludeFolders 자동 할당량에서 제외할 즉시 하위 디렉터리의 배열을 검색하거나 설정합니다. (Get)

get_ExecutablePath 실행할 실행 프로그램 또는 스크립트의 전체 경로를 검색하거나 설정합니다. (Get)
get_ExecutionOption 파일에 규칙을 적용하는 방법을 결정합니다. (Get)
get_ExpirationDirectory 만료된 파일을 포함할 루트 디렉터리입니다. (Get)
get_FileId 파일의 NTFS 파일 식별자입니다.
get_FileNamePattern 이름에 대한 조건 속성:_wildcard 필터입니다. (Get)
get_FilePropertyNames 모음에 포함된 속성의 이름 목록입니다.
get_FileScreenFlags 파일 화면 플래그를 검색하거나 설정합니다. (Get)
get_Formats 작업이 실행되면 생성할 보고서의 형식입니다. (Get)
get_Formats 보고서의 콘텐츠 형식을 결정하는 형식 배열을 검색하거나 설정합니다. (Get)
get_FromDate 파일 관리 작업이 만료되는 파일(만료된 파일 디렉터리로 파일 이동)을 시작할 날짜입니다. 이 속성은 파일 관리 작업에 대한 사용자 지정 명령에도 적용됩니다. (Get)
get_HostingProcessPid 모듈의 프로세스 식별자입니다.
get_HostingUserAccount 모듈이 실행되는 사용자 계정입니다.

get_Id 작업의 식별자를 검색합니다.
get_Id 개체의 식별자를 검색합니다.
get_Item 컬렉션에서 요청된 항목을 검색합니다. (IFsrmCollection.get_Item)
get_KillTimeOut ExecutablePath 속성에 지정된 실행 프로그램을 실행하는 프로세스를 종료하기 전에 서버에서 대기하는 시간(분)을 검색하거나 설정합니다. (Get)
get_LastAccessTime 파일에 마지막으로 액세스한 날짜와 시간입니다.
get_LastError 작업이 마지막으로 실행된 시점의 오류 메시지입니다.
get_LastError 보고서를 마지막으로 실행한 시간에서 오류 메시지를 검색합니다.
get_LastGeneratedFileNamePrefix 보고서가 마지막으로 실행되었을 때 보고서의 생성된 파일 이름을 검색합니다.
get_LastGeneratedInDirectory 보고서가 마지막으로 실행되었을 때 보고서가 저장된 로컬 디렉터리 경로를 검색합니다.
get_LastModificationTime 파일이 마지막으로 수정된 날짜 및 시간입니다.
get_LastModified 분류자의 내부 규칙이 64비트 FILETIME 값으로 마지막으로 수정된 시간입니다.
get_LastModified 규칙이 마지막으로 수정된 날짜입니다.

get_LastReportPathWithoutExtension 작업이 마지막으로 실행되었을 때 보고서가 저장된 로컬 디렉터리 경로입니다.
get_LastRun 파일 관리 작업이 마지막으로 실행된 시간입니다.
get_LastRun 보고서가 마지막으로 실행된 시기에 대한 타임스탬프를 검색합니다.
get_Logging 분류 규칙을 실행할 때 수행할 로깅 유형입니다. (Get)
get_Logging 수행할 로깅 유형입니다. (Get)
get_LogResult FSRM이 실행 프로그램의 반환 코드를 포함하는 애플리케이션 이벤트를 기록할지 여부를 결정하는 값을 검색하거나 설정합니다. (Get)
get_MailBcc 이 작업에서 전자 메일을 생성할 때 전자 메일이 전송되는 숨은 참조(숨은 참조) 전자 메일 주소를 검색하거나 설정합니다. (Get)
get_MailCc 이 작업에서 전자 메일을 생성할 때 전자 메일이 전송되는 참조(Carbon Copy) 전자 메일 주소를 검색하거나 설정합니다. (Get)
get_MailFrom 작업에서 전자 메일 메시지를 생성할 때 전자 메일의 보낸 사람으로 사용할 전자 메일 주소를 검색하거나 설정합니다. (Get)
get_MailFrom 전자 메일 메시지가 전송되는 기본 전자 메일 주소를 검색하거나 설정합니다. (Get)
get_MailReplyTo 전자 메일 메시지의 받는 사람이 회신할 때 회신 주소로 사용할 전자 메일 주소를 검색하거나 설정합니다. (Get)

get_MailSubject 이 작업에서 전자 메일을 생성할 때 전송되는 전자 메일의 제목을 검색하거나 설정합니다. (Get)
get_MailTo 이 작업에서 전자 메일을 생성할 때 전자 메일이 전송되는 전자 메일 주소를 검색하거나 설정합니다. (Get)
get_MailTo 보고서를 보낼 전자 메일 주소를 검색하거나 설정합니다. (Get)
get_MailTo 보고서를 보낼 전자 메일 주소(있는 경우)입니다. (Get)
get_MailTo 이메일을 통해 보고서를 받을 전자 메일 주소를 검색하거나 설정합니다. (Get)
get_MatchesSourceTemplate 이 할당량 개체의 속성 값이 파생된 템플릿의 속성 값과 일치하는지 여부를 결정하는 값을 검색합니다.
get_MatchesSourceTemplate 이 파일 화면 개체의 속성 값이 개체가 파생된 템플릿의 값과 일치하는지 여부를 결정하는 값을 검색합니다.
get_Members 파일 그룹에 포함된 파일을 결정하는 파일 이름 패턴을 검색하거나 설정합니다. (Get)
get_Messages 모음에 추가된 오류 메시지 목록입니다.
get_MessageText 이 작업에서 전자 메일을 생성할 때 전송되는 전자 메일의 메시지 텍스트를 검색하거나 설정합니다. (Get)
get_MessageText 작업이 실행될 때 기록되는 이벤트 텍스트를 검색하거나 설정합니다. (IFsrmActionEventLog.get_MessageText)

get_ModuleClsid 이 모듈 정의에 정의된 모듈을 구현하는 클래스에 대한 COM 클래스 식별자를 지정하는 문자열 표현입니다. (Get)
get_ModuleDefinitionName 이 규칙을 실행하려는 모듈 정의의 이름입니다. (Get)
get_ModuleImplementation 파이프라인 모듈을 구현하는 인터페이스입니다.
get_ModuleName 모듈의 이름입니다.
get_ModuleType 모듈의 형식입니다.
get_MonitorCommand FSRM이 ExecutablePath 속성에 지정된 실행 프로그램을 모니터링할지 여부를 결정하는 값을 검색하거나 설정합니다. (Get)
get_Name 할당량 템플릿의 이름을 검색하고 설정합니다. (Get)
get_Name 모듈 정의의 이름입니다. (Get)
get_Name 속성의 이름입니다. (IFsrmProperty.get_Name)
get_Name 모음의 속성이 들어 있는 파일의 이름입니다.
get_Name 속성의 이름입니다. (IFsrmPropertyDefinition.get_Name)
get_Name 분류 속성 정의 값의 이름입니다.

get_Name 규칙의 이름입니다. (Get)
get_Name 파일 관리 작업의 이름입니다. (Get)
get_Name 속성 조건의 값과 비교할 값을 가진 분류 속성의 이름입니다. (Get)
get_Name 보고서의 이름을 검색하거나 설정합니다. (Get)
get_Name 파일 그룹의 이름을 검색하거나 설정합니다. (Get)
get_Name 파일 화면 템플릿의 이름을 검색하고 설정합니다. (Get)
get_NamespaceRoots 분류를 실행할 때 규칙이 적용되는 디렉터리 경로의 배열입니다. (Get)
get_NamespaceRoots 파일 관리 작업이 실행될 때 검색될 로컬 디렉터리 경로의 배열입니다. (Get)
get_NamespaceRoots 보고서 작업이 실행될 때 검색할 로컬 디렉터리 경로의 배열을 검색하거나 설정합니다. (Get)
get_NeedsExplicitValue 분류자를 사용하는 규칙이 분류 속성의 값을 제공해야 하는지 여부를 결정합니다. (Get)
get_NeedsFileContent 모듈이 파일의 내용을 읽어야 하는지 여부를 결정합니다. (Get)
get_NonMembers 파일 그룹에서 제외된 파일을 결정하는 파일 이름 패턴을 검색하거나 설정합니다. (Get)

get_Notifications 작업에 대해 정의된 알림 기간 목록입니다.
get_OperationType 파일 관리 작업의 형식입니다. 형식은 모든 조건이 충족될 때 파일에서 수행할 작업을 결정합니다. (Get)
get_Operator FsrmPropertyConditionType 열거형으로 열거된 비교 연산자를 지정합니다. (Get)
get_OverwriteOnCommit 템플릿을 가져올 때 할당량 템플릿이 있는지 여부를 결정하는 값을 검색하거나 설정합니다. (Get)
get_OverwriteOnCommit 파일 그룹이 있는지 여부를 결정하는 값을 검색하거나 설정합니다. (Get)
get_OverwriteOnCommit 템플릿을 가져올 때 파일 화면 서식 파일이 있는지 여부를 결정하는 값을 검색하거나 설정합니다. (Get)
get_OwnerSid 파일 소유자의 SID입니다.
get_Parameters 모듈에 전달할 선택적 매개 변수입니다. (Get)
get_Parameters 속성 정의에 대한 매개 변수입니다. (Get)
get_Parameters 분류기에 전달되는 매개 변수입니다. (Get)
get_Parameters 파일 관리 작업에 대한 매개 변수입니다. (Get)
get_ParentDirectoryId 파일의 부모 디렉터리의 NTFS 식별자입니다.

get_Path 할당량이 적용되는 경로를 검색합니다.
get_Path 파일 화면 개체와 연결된 디렉터리 경로를 검색합니다.
get_Path 이 파일 화면 예외와 연결된 경로를 검색합니다.
get_PossibleValues 속성을 설정할 수 있는 가능한 값입니다. (Get)
get_PropertiesAffected 분류자가 영향을 줄 수 있는 속성 이름 목록입니다. (Get)
get_PropertiesUsed 분류자가 검사하는 속성 이름 목록입니다. (Get)
get_PropertyAffected 이 규칙에 영향을 주는 속성의 이름입니다. (Get)
get_PropertyBagFlags 속성 모음에 대한 추가 정보를 제공하는 플래그 집합입니다.
get_PropertyConditions 작업에 대해 지정된 속성 조건 목록입니다.
get_PropertyDefinitionFlags 속성 정의에 대한 플래그를 포함합니다.
get_PropertyFlags 속성에 대한 추가 정보를 제공하는 플래그 값입니다.
get_PropertyId FsrmFileSystemPropertyId 열거형으로 열거된 미리 정의된 파일 속성을 지정합니다. (Get)

get_PropertyName 파일 조건 속성의 이름을 지정합니다. (Get)
get_QuotaFlags 개체의 할당량 플래그를 검색하거나 설정합니다. (Get)
get_QuotaLimit 개체의 할당량 제한을 검색하거나 설정합니다. (Get)
get_QuotaPeakUsage 이 할당량에 부과되는 가장 많은 양의 디스크 공간 사용량을 검색합니다.
get_QuotaPeakUsageTime IFsrmQuota::QuotaPeakUsage 속성이 설정된 날짜 및 시간을 검색합니다.
get_QuotaUsed 이 할당량에 부과되는 현재 디스크 공간 사용량을 검색합니다.
get_RelativeNamespaceRoot 파일이 평가되는 네임스페이스 루트의 상대 경로입니다.
get_RelativePath 파일에 대한 상대 경로입니다.
get_ReportEnabled 작업이 실행될 때 보고서를 생성할지 여부를 나타냅니다. (Get)
get_ReportTypes 생성할 보고서의 형식을 검색하거나 설정합니다. (Get)
get_Results 각 파생 개체에 대한 업데이트의 성공 또는 실패를 나타내는 HRESULT 값을 검색합니다.
get_RuleFlags 규칙의 상태를 정의하는 플래그입니다. (Get)

get_RuleType 규칙의 형식입니다.
get_RunLimitInterval 작업이 다시 실행되기 전에 만료되어야 하는 간격을 검색하거나 설정합니다. (Get)
get_RunningStatus 작업의 실행 상태입니다.
get_RunningStatus 보고서 작업의 실행 상태를 검색합니다.
get_Size 파일의 크기입니다.
get_SizeAllocated 파일의 할당 크기입니다.
get_SmtpServer FSRM에서 전자 메일을 보내는 데 사용하는 SMTP 서버를 검색하거나 설정합니다. (Get)
get_Sources 값을 설정한 모듈 및 규칙입니다.
get_SourceTemplateName 이 할당량이 파생된 템플릿의 이름을 검색합니다.
get_SourceTemplateName 이 파일 화면 개체가 파생된 템플릿의 이름을 검색합니다.
get_State 컬렉션의 상태를 검색합니다.
get_StorageType 스토리지 모듈에서 사용하는 스토리지 유형입니다. (Get)

get_SupportedExtensions 이 모듈에서 지원하는 파일 확장자 목록입니다. (Get)
get_Task 작업과 연결할 예약된 작업의 이름입니다. (Get)
get_Task 보고서 작업의 이름을 검색하거나 설정합니다. (Get)
get_Thresholds 할당량 개체의 임계값을 검색합니다.
get_Type 속성에 포함된 데이터의 형식입니다. (Get)
get_Type 속성 조건이 충족되는지 여부를 확인하는 데 사용되는 비교 연산자입니다. (Get)
get_Type 생성할 보고서의 형식을 검색합니다.
get_UniqueID 분류 속성 정의 값의 고유 ID입니다.
get_UpdatesFileContent 모듈이 파일의 내용을 업데이트하는지 여부를 결정합니다. (Get)
get_UserAccount 개체와 연결된 사용자 계정의 문자열 형식을 검색합니다.
get_UserAccount 파일을 검사할 사용자의 계정 이름입니다.
get_UserSid 개체와 연결된 사용자 SID(보안 식별자)의 문자열 형식을 검색합니다.

get_UserSid 파일을 선별할 사용자의 SID입니다.
get_Value 이 규칙이 속성을 설정할 값입니다. (Get)
get_Value 속성 값입니다.
get_Value 파일 조건 속성 값을 지정합니다. (Get)
get_Value 속성 조건의 값입니다. (Get)
get_ValueDefinitions 이 속성에는 속성 정의의 가능한 값 정의가 포함됩니다.
get_ValueDescriptions PossibleValues 속성에 지정된 각 가능한 값에 대한 설명입니다. (Get)
get_ValueType FsrmPropertyValueType 열거형으로 열거된 파일 조건 속성 값의 형식을 지정합니다. (Get)
get_Version 모듈 정의의 버전입니다. (Get)
get_VolumeIndex 스캐너가 파일이 있는 볼륨을 참조하는 데 사용하는 인덱스입니다.
get_VolumeName 파일이 있는 볼륨의 이름입니다.
get_WorkingDirectory 실행 프로그램이 실행될 작업 디렉터리를 검색하거나 설정합니다. (Get)

GetActionRunLimitInterval 전역 실행 제한 간격을 사용하는 작업이 다시 실행되기 전에 기다려야 하는 시간을 가져옵니다.
GetAutoApplyQuota 지정된 디렉터리에 대한 자동 할당량을 검색합니다.
GetById 컬렉션에서 지정된 개체를 검색합니다.
GetDefaultFilter 지정된 보고서 형식에 사용되는 기본 보고서 필터 값을 검색합니다.
GetFieldValue 속성 모음에서 지정된 필드의 값을 가져옵니다.
GetFileGroup FSRM에서 지정된 파일 그룹을 검색합니다.
GetFileManagementJob 지정된 파일 관리 작업을 가져옵니다.
GetFileProperty 파일 또는 폴더에서 지정된 속성을 검색합니다.
GetFileProperty 속성 모음에서 지정된 속성을 검색합니다.
GetFileScreen 지정된 파일 화면을 검색합니다.
GetFileScreenException 지정된 파일 화면 예외를 검색합니다.
GetFileStreamInterface 파일 내용에 액세스하는 데 사용할 수 있는 파일 스트림 인터페이스를 검색합니다.

GetFilter 지정된 보고서 필터의 값을 검색합니다.
GetModuleDefinition 지정된 모듈 정의를 검색합니다.
GetOutputDirectory 지정된 컨텍스트가 있는 보고서가 저장되는 로컬 디렉터리 경로를 검색합니다.
GetPropertyDefinition 지정된 속성 정의를 검색합니다.
GetPropertyValueToApply IFsrmClassifierModuleImplementation::OnBeginFile 메서드에 의해 가장 최근에 지정된 파일의 지정된 속성에 적용해야 하는 분류자에서 값을 검색합니다.
GetQuota 지정된 디렉터리에 대한 할당량을 검색합니다.
GetReportJob 지정된 보고서 작업을 검색합니다.
GetReportSizeLimit 지정된 보고서 크기 제한의 현재 값을 검색합니다.
GetRestrictiveQuota 지정된 경로에 대한 가장 제한적인 할당량을 검색합니다.
GetRule 지정된 규칙을 검색합니다.
GetSharePathsForLocalPath 지정된 로컬 경로를 가리키는 네트워크 공유 목록을 검색합니다.
GetTemplate 지정된 할당량 템플릿을 검색합니다.

GetTemplate 지정된 파일 화면 템플릿을 검색합니다.
ImportFileGroups 지정된 파일에서 하나 이상의 파일 그룹을 가져옵니다.
ImportFileGroups XML 문자열에서 지정된 파일 그룹을 가져옵니다.
ImportFileScreenTemplates 지정된 파일에서 하나 이상의 파일 화면 템플릿을 가져옵니다.
ImportQuotaTemplates 지정된 파일에서 하나 이상의 할당량 템플릿을 가져옵니다.
ImportTemplates XML 문자열에서 지정된 할당량 템플릿을 가져옵니다.
ImportTemplates XML 문자열에서 지정된 파일 화면 템플릿을 가져옵니다.
IsAffectedByQuota 지정된 경로에 할당량이 적용되는지 여부를 결정하는 값을 검색합니다.
IsFilterValidForReportType 지정된 보고서 형식에 대해 지정된 보고서 필터를 구성할 수 있는지 여부를 결정하는 값을 검색합니다.
LoadProperties 스토리지 모듈에 모든 속성을 로드하도록 지시합니다.
ModifyNotification 파일 관리 작업의 알림 목록에서 알림 값을 변경합니다.
ModifyScheduleTask 보고서 작업을 트리거하는 데 사용되는 작업을 수정합니다.

ModifyThreshold 임계값을 변경합니다.
OnBeginFile 지정된 속성 모음을 사용하여 파일 처리를 준비하도록 분류자를 지시합니다.
OnEndFile 파일을 처리한 후 정리를 수행하도록 분류자를 지시합니다.
Onload 파이프라인 모듈을 초기화합니다.
OnUnload 정리 작업을 수행하도록 모듈에 알합니다.
put_Account ExecutablePath 속성에 지정된 실행 프로그램을 실행하는 데 사용되는 시스템 계정을 검색하거나 설정합니다. (Put)
put_Account 모듈을 실행할 때 사용할 계정입니다. (Put)
put_AdminEmail 관리자의 전자 메일 주소를 검색하거나 설정합니다. (Put)
put_AllowedFileGroups 디렉터리에 허용되는 파일의 파일 이름 패턴을 포함하는 파일 그룹의 이름을 검색하거나 설정합니다. (Put)
put_Arguments ExecutablePath 속성에 지정된 실행 프로그램에 전달할 인수를 검색하거나 설정합니다. (Put)
put_AttachmentFileListSize 목록에 포함할 최대 파일 수입니다. (Put)

put_BlockedFileGroups 이 화면에서 차단된 파일을 지정하는 데 사용되는 파일 이름 패턴을 포함하는 파일 그룹의 이름을 검색하거나 설정합니다. (Put)
put_Capabilities 스토리지 모듈의 기능을 지정하는 플래그입니다. (Put)
put_ClassificationReportEnabled 분류 보고를 사용할 수 있는지 여부를 결정합니다. (Put)
put_ClassificationReportFormats 분류 보고서를 생성할 형식 목록입니다. (Put)
put_ClassificationReportMailTo 분류 보고서를 보낼 이메일 주소(있는 경우)입니다. (Put)
put_Company 분류 또는 스토리지 모듈을 만든 회사의 이름입니다. (Put)
put_DaysSinceFileCreated 파일을 만든 후 경과된 일 수입니다. (Put)
put_DaysSinceFileLastAccessed 파일에 마지막으로 액세스한 후 경과된 일 수입니다. (Put)
put_DaysSinceFileLastModified 파일이 마지막으로 수정된 후 경과된 일 수입니다. (Put)
put_Description 개체에 대한 설명을 검색하거나 설정합니다. (Put)
put_Description 보고서에 대한 설명을 검색하거나 설정합니다. (Put)
put_DisableCommandLine FSRM이 명령줄 동작의 실행을 차단하는지 여부를 결정하는 값을 검색하거나 설정합니다. (Put)

put_DisplayName 속성 정의의 표시 이름입니다. (Put)
put_Enabled 모듈을 사용할 수 있는지 여부를 결정합니다. (Put)
put_Enabled 작업을 사용할 수 있는지 여부를 나타냅니다(실행할 수 있습니다). (Put)
put_EnableScreeningAudit FSRM이 파일 화면 위반에 대한 감사 레코드를 유지하는지 여부를 결정하는 값을 검색하거나 설정합니다. (Put)
put_EventType 실행 시 동작이 기록하는 이벤트 유형을 검색하거나 설정합니다. (Put)
put_ExcludeFolders 자동 할당량에서 제외할 즉시 하위 디렉터리의 배열을 검색하거나 설정합니다. (Put)
put_ExecutablePath 실행할 실행 프로그램 또는 스크립트의 전체 경로를 검색하거나 설정합니다. (Put)
put_ExecutionOption 파일에 규칙을 적용하는 방법을 결정합니다. (Put)
put_ExpirationDirectory 만료된 파일을 포함할 루트 디렉터리입니다. (Put)
put_FileNamePattern 이름에 대한 조건 속성:_wildcard 필터입니다. (Put)
put_FileScreenFlags 파일 화면 플래그를 검색하거나 설정합니다. (Put)
put_Formats 작업이 실행되면 생성할 보고서의 형식입니다. (Put)

put_Formats 보고서의 콘텐츠 형식을 결정하는 형식 배열을 검색하거나 설정합니다. (Put)
put_FromDate 파일 관리 작업이 만료되는 파일(만료된 파일 디렉터리로 파일 이동)을 시작할 날짜입니다. 이 속성은 파일 관리 작업에 대한 사용자 지정 명령에도 적용됩니다. (Put)
put_KillTimeOut ExecutablePath 속성에 지정된 실행 프로그램을 실행하는 프로세스를 종료하기 전에 서버에서 대기하는 시간(분)을 검색하거나 설정합니다. (Put)
put_Logging 분류 규칙을 실행할 때 수행할 로깅 유형입니다. (Put)
put_Logging 수행할 로깅 유형입니다. (Put)
put_LogResult FSRM이 실행 프로그램의 반환 코드를 포함하는 애플리케이션 이벤트를 기록할지 여부를 결정하는 값을 검색하거나 설정합니다. (Put)
put_MailBcc 이 작업에서 전자 메일을 생성할 때 전자 메일이 전송되는 숨은 참조(숨은 참조) 전자 메일 주소를 검색하거나 설정합니다. (Put)
put_MailCc 이 작업에서 전자 메일을 생성할 때 전자 메일이 전송되는 참조(참조) 전자 메일 주소를 검색하거나 설정합니다. (Put)
put_MailFrom 작업이 전자 메일 메시지를 생성할 때 전자 메일의 보낸 사람으로 사용할 전자 메일 주소를 검색하거나 설정합니다. (Put)
put_MailFrom 전자 메일 메시지가 전송되는 기본 전자 메일 주소를 검색하거나 설정합니다. (Put)

put_MailReplyTo 전자 메일 메시지 수신자가 회신할 때 회신 주소로 사용할 전자 메일 주소를 검색하거나 설정합니다. (Put)
put_MailSubject 이 작업에서 전자 메일을 생성할 때 전송되는 전자 메일의 제목을 검색하거나 설정합니다. (Put)
put_MailTo 이 작업에서 전자 메일을 생성할 때 전자 메일이 전송되는 전자 메일 주소를 검색하거나 설정합니다. (Put)
put_MailTo 보고서를 보낼 전자 메일 주소를 검색하거나 설정합니다. (Put)
put_MailTo 보고서를 보낼 전자 메일 주소(있는 경우)입니다. (Put)
put_MailTo 전자 메일을 통해 보고서를 받을 전자 메일 주소를 검색하거나 설정합니다. (Put)
put_Members 파일 그룹에 포함된 파일을 결정하는 파일 이름 패턴을 검색하거나 설정합니다. (Put)
put_MessageText 이 작업에서 전자 메일을 생성할 때 전송되는 전자 메일의 메시지 텍스트를 검색하거나 설정합니다. (Put)
put_MessageText 작업이 실행될 때 기록되는 이벤트 텍스트를 검색하거나 설정합니다. (IFsrmActionEventLog.put_MessageText)
put_ModuleClsid 이 모듈 정의에 정의된 모듈을 구현하는 클래스에 대한 COM 클래스 식별자를 지정하는 문자열 표현입니다. (Put)
put_ModuleDefinitionName 이 규칙을 실행하려는 모듈 정의의 이름입니다. (Put)

put_MonitorCommand FSRM이 ExecutablePath 속성에 지정된 실행 프로그램을 모니터링할지 여부를 결정하는 값을 검색하거나 설정합니다. (Put)
put_Name 할당량 템플릿의 이름을 검색하고 설정합니다. (Put)
put_Name 모듈 정의의 이름입니다. (Put)
put_Name 속성의 이름입니다. (IFsrmPropertyDefinition.put_Name)
put_Name 규칙의 이름입니다. (Put)
put_Name 파일 관리 작업의 이름입니다. (Put)
put_Name 속성 조건의 값과 비교할 값을 가진 분류 속성의 이름입니다. (Put)
put_Name 보고서의 이름을 검색하거나 설정합니다. (Put)
put_Name 파일 그룹의 이름을 검색하거나 설정합니다. (Put)
put_Name 파일 화면 템플릿의 이름을 검색하고 설정합니다. (Put)
put_NamespaceRoots 분류를 실행할 때 규칙이 적용되는 디렉터리 경로의 배열입니다. (Put)
put_NamespaceRoots 파일 관리 작업이 실행될 때 검색될 로컬 디렉터리 경로의 배열입니다. (Put)

put_NamespaceRoots 보고서 작업이 실행될 때 검색할 로컬 디렉터리 경로의 배열을 검색하거나 설정합니다. (Put)
put_NeedsExplicitValue 분류자를 사용하는 규칙이 분류 속성의 값을 제공해야 하는지 여부를 결정합니다. (Put)
put_NeedsFileContent 모듈이 파일의 내용을 읽어야 하는지 여부를 결정합니다. (Put)
put_NonMembers 파일 그룹에서 제외된 파일을 결정하는 파일 이름 패턴을 검색하거나 설정합니다. (Put)
put_OperationType 파일 관리 작업의 형식입니다. 형식은 모든 조건이 충족될 때 파일에서 수행할 작업을 결정합니다. (Put)
put_Operator FsrmPropertyConditionType 열거형으로 열거된 비교 연산자를 지정합니다. (Put)
put_OverwriteOnCommit 템플릿을 가져올 때 할당량 템플릿이 있는지 여부를 결정하는 값을 검색하거나 설정합니다. (Put)
put_OverwriteOnCommit 파일 그룹이 있는지 여부를 결정하는 값을 검색하거나 설정합니다. (Put)
put_OverwriteOnCommit 템플릿을 가져올 때 파일 화면 서식 파일이 있는지 여부를 결정하는 값을 검색하거나 설정합니다. (Put)
put_Parameters 모듈에 전달할 선택적 매개 변수입니다. (Put)
put_Parameters 속성 정의에 대한 매개 변수입니다. (Put)
put_Parameters 분류기에 전달되는 매개 변수입니다. (Put)

put_Parameters 파일 관리 작업에 대한 매개 변수입니다. (Put)
put_PossibleValues 속성을 설정할 수 있는 가능한 값입니다. (Put)
put_PropertiesAffected 분류자가 영향을 줄 수 있는 속성 이름 목록입니다. (Put)
put_PropertiesUsed 분류자가 검사하는 속성 이름 목록입니다. (Put)
put_PropertyAffected 이 규칙에 영향을 주는 속성의 이름입니다. (Put)
put_PropertyId FsrmFileSystemPropertyId 열거형으로 열거된 미리 정의된 파일 속성을 지정합니다. (Put)
put_PropertyName 파일 조건 속성의 이름을 지정합니다. (Put)
put_QuotaFlags 개체의 할당량 플래그를 검색하거나 설정합니다. (Put)
put_QuotaLimit 개체의 할당량 제한을 검색하거나 설정합니다. (Put)
put_ReportEnabled 작업이 실행될 때 보고서를 생성할지 여부를 나타냅니다. (Put)
put_ReportTypes 생성할 보고서의 형식을 검색하거나 설정합니다. (Put)
put_RuleFlags 규칙의 상태를 정의하는 플래그입니다. (Put)

put_RunLimitInterval 작업이 다시 실행되기 전에 만료되어야 하는 간격을 검색하거나 설정합니다. (Put)
put_SmtpServer FSRM이 전자 메일을 보내는 데 사용하는 SMTP 서버를 검색하거나 설정합니다. (Put)
put_StorageType 스토리지 모듈에서 사용하는 스토리지 유형입니다. (Put)
put_SupportedExtensions 이 모듈에서 지원하는 파일 확장자 목록입니다. (Put)
put_Task 작업과 연결할 예약된 작업의 이름입니다. (Put)
put_Task 보고서 작업의 이름을 검색하거나 설정합니다. (Put)
put_Type 속성에 포함된 데이터 형식입니다. (Put)
put_Type 속성 조건이 충족되는지 여부를 확인하는 데 사용되는 비교 연산자입니다. (Put)
put_UpdatesFileContent 모듈이 파일 내용을 업데이트하는지 여부를 결정합니다. (Put)
put_Value 이 규칙이 속성을 설정할 값입니다. (Put)
put_Value 파일 조건 속성 값을 지정합니다. (Put)
put_Value 속성 조건의 값입니다. (Put)

put_ValueDescriptions PossibleValues 속성에 지정된 각 가능한 값에 대한 설명입니다. (Put)
put_ValueType FsrmPropertyValueType 열거형으로 열거된 파일 조건 속성 값의 형식을 지정합니다. (Put)
put_Version 모듈 정의의 버전입니다. (Put)
put_WorkingDirectory 실행 프로그램이 실행될 작업 디렉터리를 검색하거나 설정합니다. (Put)
RefreshUsageProperties FSRM의 현재 정보에서 이 개체의 할당량 사용 정보를 새로 고칩니다.
제거 인덱스 값을 사용하여 컬렉션에서 지정된 개체를 제거합니다.
RemoveById 개체 식별자를 사용하여 컬렉션에서 지정된 개체를 제거합니다.
ResetPeakUsage 이 할당량의 최대 사용량을 현재 사용량으로 다시 설정합니다.
실행 작업을 실행합니다.
실행 작업의 모든 보고서를 실행합니다.
RunClassification 분류 규칙을 실행하고 분류 보고서를 생성합니다.
SaveProperties 스토리지 모듈에 파일과 연결된 속성을 저장하도록 지시합니다.

검사 지정된 경로에서 할당량 검사를 시작합니다.
SetActionRunLimitInterval 전역 실행 제한 간격을 사용하는 작업이 다시 실행되기 전에 기다려야 하는 시간을 설정합니다.
SetDefaultFilter 지정된 보고서 형식에 사용할 기본 보고서 필터 값을 설정합니다.
SetFileProperty 파일 또는 폴더에서 지정된 속성의 값을 설정합니다.
SetFileProperty 속성 모음에서 지정된 속성을 설정합니다.
SetFilter 지정된 보고서 필터의 현재 값을 설정합니다.
SetOutputDirectory 보고서가 저장되는 로컬 디렉터리 경로를 설정합니다.
SetReportSizeLimit 지정된 보고서 크기 제한의 현재 값을 설정합니다.
표시 ADR(액세스 거부 수정) 클라이언트 대화 상자를 표시합니다.
UseDefinitions FSRM이 인식하는 속성 정의를 지정합니다.
UseRulesAndDefinitions 분류자가 처리해야 하는 규칙 및 관련 속성 정의의 컬렉션을 지정합니다.
VerifyNamespaces 보고서의 원본으로 사용되는 지정된 로컬 디렉터리 경로가 유효한지 확인합니다.

WaitForClassificationCompletion 지정된 기간 또는 분류 실행이 완료될 때까지 기다립니다.
WaitForCompletion 비동기 컬렉션이 개체를 수집하는 데 걸리는 시간을 제한합니다.
WaitForCompletion 지정된 기간 또는 작업 실행이 완료될 때까지 기다립니다.
WaitForCompletion 작업의 보고서가 완료되기를 기다립니다.

인터페이스

DIFsrmClassificationEvents ClassifyFiles 호출을 처리하는 동안 수신되는 이벤트를 처리합니다.
IFsrmAccessDeniedRemediationClient ADR(액세스 거부 수정) 클라이언트 사용자 인터페이스를 표시하는 데 사용됩니다.
IFsrmAction 모든 FSRM 작업 인터페이스에 대한 기본 클래스입니다.
IFsrmActionCommand 할당량, 파일 화면 또는 파일 관리 작업 이벤트에 대한 응답으로 명령 또는 스크립트를 실행하는 데 사용됩니다.
IFsrmActionEmail 할당량 또는 파일 화면 이벤트에 대한 응답으로 전자 메일 메시지를 보내는 데 사용됩니다.
IFsrmActionEmail2 전자 메일 알림에 나열된 만료된 파일의 수를 제한하는 데 사용됩니다.

IFsrmActionEventLog 할당량, 파일 화면 또는 파일 관리 작업 이벤트에 대한 응답으로 Windows 애플리케이션 이벤트 로그에 이벤트를 기록하는 데 사용됩니다.
IFsrmActionReport 할당량 또는 파일 화면 이벤트에 대한 응답으로 보고서를 생성하는 데 사용됩니다.
IFsrmAutoApplyQuota 자동 할당량이 적용되는 디렉터리의 새 하위 디렉터리와 기존 하위 디렉터리에 할당량을 자동으로 추가하는 데 사용됩니다.
IFsrmClassificationManager 파일 분류를 관리합니다. 이 인터페이스를 사용하여 분류에 사용할 속성을 정의하고, 파일을 분류하기 위한 분류 규칙을 추가하고, 분류 및 스토리지 모듈을 정의하고, 분류 보고를 사용하도록 설정합니다. (IFsrmClassificationManager)
IFsrmClassificationManager2 파일 분류를 관리합니다. 이 인터페이스를 사용하여 분류에 사용할 속성을 정의하고, 파일을 분류하기 위한 분류 규칙을 추가하고, 분류 및 스토리지 모듈을 정의하고, 분류 보고를 사용하도록 설정합니다. (IFsrmClassificationManager2)
IFsrmClassificationRule 분류 규칙을 정의합니다.
IFsrmClassifierModuleDefinition 분류자 모듈을 정의합니다.
IFsrmClassifierModuleImplementation 분류자 모듈은 이 인터페이스를 구현합니다. FSRM은 분류를 실행할 때 모듈의 구현을 호출합니다.
IFsrmCollection FSRM 개체의 컬렉션을 정의합니다.
IFsrmCommittableCollection 컬렉션에 추가하거나 컬렉션에서 제거할 수 있는 동일한 형식의 개체를 가질 수 있는 FSRM 개체의 컬렉션을 정의합니다. 컬렉션의 모든 개체는 단일 일괄 처리 작업에서 커밋할 수도 있습니다.

IFsrmDerivedObjectsResult 원본 템플릿이 CommitAndUpdateDerived 메서드를 호출할 때 결과에 액세스하는 데 사용됩니다.
IFsrmExportImport FSRM 개체를 내보내고 가져오는 데 사용됩니다.
IFsrmFileConditionProperty 파일 조건 속성을 정의합니다.
IFsrmFileGroup 하나 이상의 파일 이름 패턴을 기반으로 파일 그룹을 정의하는 데 사용됩니다.
IFsrmFileGroupImported 가져온 파일 그룹 개체를 구성하는 데 사용됩니다.
IFsrmFileGroupManager 파일 그룹 개체를 관리하는 데 사용됩니다.
IFsrmFileManagementJob 파일 관리 작업을 정의합니다.
IFsrmFileManagementJobManager 파일 관리 작업을 관리하는 데 사용됩니다.
IFsrmFileScreen 파일 그룹이 지정된 디렉터리에 저장되지 않도록 차단하는 파일 화면을 구성하는 데 사용됩니다.
IFsrmFileScreenBase 모든 파일 화면 인터페이스에 대한 기본 클래스입니다.
IFsrmFileScreenException 파일 차단 프로세스에서 지정된 파일을 제외하는 예외를 구성하는 데 사용됩니다.
IFsrmFileScreenManager 파일 화면 개체를 관리하는 데 사용됩니다.

IFsrmFileScreenTemplate 새 파일 화면을 파생할 수 있는 템플릿을 구성하는 데 사용됩니다.
IFsrmFileScreenTemplateImported 가져온 파일 화면 템플릿을 저장하는 데 사용됩니다.
IFsrmFileScreenTemplateManager 파일 화면 템플릿을 관리하는 데 사용됩니다.
IFsrmMutableCollection 컬렉션에 개체를 추가하거나 컬렉션에서 제거할 수 있는 FSRM 개체의 컬렉션을 관리하는 데 사용됩니다.
IFsrmObject 모든 FSRM 개체에 대한 기본 클래스입니다.
IFsrmPathMapper 로컬 경로에 매핑된 네트워크 공유 경로를 검색하는 데 사용됩니다.
IFsrmPipelineModuleConnector FSRM과 파이프라인 모듈 구현 간의 통신 채널을 만듭니다.
IFsrmPipelineModuleDefinition 파일을 분류하거나 파일에서 속성을 저장하고 검색하는 데 사용되는 모듈을 정의합니다.
IFsrmPipelineModuleImplementation IFsrmClassifierModuleImplementation 및 IFsrmStorageModuleImplementation에 대한 추상 인터페이스입니다.
IFsrmProperty 속성의 인스턴스를 정의합니다.
IFsrmPropertyBag 파일에 대한 분류 속성을 포함합니다.
IFsrmPropertyBag2 IFsrmPropertyBag2 인터페이스

IFsrmPropertyCondition 파일 관리 작업에서 파일이 만료되었는지 확인하는 데 사용하는 속성 조건을 정의합니다.
IFsrmPropertyDefinition 파일을 분류하는 데 사용할 속성을 정의합니다. (IFsrmPropertyDefinition)
IFsrmPropertyDefinition2 파일을 분류하는 데 사용할 속성을 정의합니다. (IFsrmPropertyDefinition2)
IFsrmPropertyDefinitionValue 분류 속성 정의 값을 설명하는 속성을 포함합니다.
IFsrmQuota 지정된 디렉터리에 대한 할당량을 정의하고 사용 통계를 검색하는 데 사용됩니다.
IFsrmQuotaBase 모든 할당량 인터페이스에 대한 기본 인터페이스입니다.
IFsrmQuotaManager 할당량을 관리하는 데 사용됩니다.
IFsrmQuotaManagerEx 할당량, 확장 버전을 관리하는 데 사용됩니다.
IFsrmQuotaObject 할당량 및 자동 할당량 인터페이스의 기본 클래스입니다.
IFsrmQuotaTemplate 새 할당량 개체를 파생할 수 있는 템플릿을 구성하는 데 사용됩니다.
IFsrmQuotaTemplateImported 가져온 할당량 템플릿을 수정하고 저장하는 데 사용됩니다.
IFsrmQuotaTemplateManager 할당량 템플릿을 관리하는 데 사용됩니다.

IFsrmReport 단일 보고서에 대한 설명 및 필터를 구성하는 데 사용됩니다.
IFsrmReportJob 보고서 작업을 구성하는 데 사용됩니다.
IFsrmReportManager 보고서 작업을 관리하는 데 사용됩니다.
IFsrmReportScheduler 보고서 작업 및 파일 관리 작업에 대한 예약된 작업을 관리하는 데 사용됩니다.
IFsrmRule 규칙을 정의합니다.
IFsrmSetting FSRM을 구성하는 데 사용됩니다.
IFsrmStorageModuleDefinition 속성 값을 읽고 쓰는 데 사용되는 로컬 스토리지 모듈을 정의합니다.
IFsrmStorageModuleImplementation 스토리지 모듈은 이 인터페이스를 구현합니다.

게임 모드

아티클2023. 03. 09.

게임 모드 기술 개요입니다.

게임 모드를 개발하려면 다음 헤더가 필요합니다.

- expandedresources.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 게임 모드

Functions

GetExpandedResourceExclusiveCpuCount 게임 모드에서 앱에 사용할 수 있는 전용 CPU 집합의 예상 수를 가져옵니다.
HasExpandedResources 현재 리소스 상태(즉, 앱이 게임 모드 또는 공유 모드에서 실행 중인지 여부)를 가져옵니다.
ReleaseExclusiveCpuSets CPU 독점성을 옵트아웃하여 앱이 모든 코어에 액세스할 수 있도록 하지만 다른 프로세스와 공유해야 하는 비용이 듭니다.

게임 디바이스 정보

아티클2023. 03. 09.

게임 장치 정보 기술 개요입니다.

게임 장치 정보를 개발하려면 다음 헤더가 필요합니다.

- gamingdeviceinformation.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 게임 디바이스 정보

열거형

GAMING_DEVICE_DEVICE_ID 게임이 실행 중인 디바이스의 유형을 나타냅니다.
GAMING_DEVICE_VENDOR_ID 게임이 실행 중인 콘솔의 공급업체를 나타냅니다.

함수

GetGamingDeviceModelInformation 게임이 실행 중인 디바이스에 대한 정보를 가져옵니다.

구조체

GAMING_DEVICE_MODEL_INFORMATION 게임이 실행 중인 디바이스에 대한 정보를 포함합니다.

GDI+

아티클2023. 03. 11.

GDI+ 기술 개요입니다.

GDI+를 개발하려면 다음 헤더가 필요합니다.

- gdiplusbase.h
- gdiplusbrush.h
- gdipluscolor.h
- gdipluscolormatrix.h
- gdipluseffects.h
- gdiplusenums.h
- gdiplusgraphics.h
- gdiplusheaders.h
- gdiplusimageattributes.h
- gdiplusimagecodec.h
- gdiplusimaging.h
- gdiplusinit.h
- gdipluslinecaps.h
- gdiplusmatrix.h
- gdiplusmetaheader.h
- gdipluspath.h
- gdipluspen.h
- gdipluspixelformats.h
- gdiplusstringformat.h
- gdiplustypes.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- GDI+

클래스

AdjustableArrowCap AdjustableArrowCap 클래스는 CustomLineCap의 하위 클래스입니다. 이 클래스는 화살표처럼 보이는 선 뚜껑을 작성합니다.

Bitmap Bitmap 클래스는 Image 클래스에서 상속됩니다.
Bitmapdata BitmapData 클래스는 Bitmap 클래스의 Bitmap::LockBits 및 Bitmap::UnlockBits 메서드에서 사용됩니다. BitmapData 개체는 비트맵의 특성을 저장합니다.
흐리게 Blur 클래스를 사용하면 가우스 흐림 효과를 비트맵에 적용하고 흐림의 특성을 지정할 수 있습니다.
BrightnessContrast BrightnessContrast 클래스를 사용하면 비트맵의 밝기와 대비를 변경할 수 있습니다.
Brush Brush 클래스는 Brush 개체를 정의하는 추상 기본 클래스입니다. Brush 개체는 사각형, 타원, 원형, 다각형 및 경로와 같은 그래픽 셰이프의 내부를 그리는 데 사용됩니다.
CachedBitmap CachedBitmap 개체는 특정 디바이스에 표시하기 위해 최적화된 형식으로 비트맵을 저장합니다. 캐시된 비트맵을 표시하려면 Graphics::D rawCachedBitmap 메서드를 호출합니다.
CharacterRange CharacterRange 개체는 문자열 내의 문자 위치 범위를 지정합니다.
색상 Color 개체는 색을 나타내는 32비트 값을 저장합니다.
ColorBalance ColorBalance 클래스를 사용하면 비트맵의 색 균형(상대적인 빨간색, 녹색 및 파랑 양)을 변경할 수 있습니다.
ColorCurve ColorCurve 클래스는 _exposure, 밀도, 대비, 강조 표시, 그림자, 중간, 흰색 채도 및 검은색 채도의 8가지 개별 조정을 포함합니다.
ColorLUT ColorLUTParams 구조체에는 각각 특정 색 채널에 대한 조회 테이블인 _alpha, 빨간색, 녹색 또는 파란색의 4개의 멤버가 있습니다.

ColorMatrixEffect ColorMatrixEffect 클래스를 사용하면 비트맵에 아핀 변환을 적용할 수 있습니다.
CustomLineCap CustomLineCap 클래스는 사용자 지정 줄 한도를 캡슐화합니다.
효과 Effect 클래스는 비트맵에 효과 및 조정을 적용하는 데 사용할 수 있는 11개의 클래스에 대한 기본 클래스 역할을 합니다. 다음 클래스는 효과에서 내림차순입니다.
EncoderParameter EncoderParameter 개체는 이미지 인코더에 전달할 수 있는 매개 변수를 보유합니다. EncoderParameter 개체를 사용하여 특정 이미지 인코더의 특정 매개 변수에서 지원하는 가능한 값 목록을 받을 수도 있습니다.
EncoderParameters EncoderParameters 개체는 배열의 EncoderParameter 개체 수를 지정하는 데이터 멤버와 함께 EncoderParameter 개체의 배열입니다.
글꼴 Font 클래스는 특정 글꼴의 패밀리, 높이, 크기 및 스타일(또는 스타일 조합)과 같은 특성을 캡슐화합니다. Font 개체는 문자열을 그릴 때 사용됩니다.
FontCollection FontCollection 클래스는 추상 기본 클래스입니다. 글꼴 컬렉션에서 글꼴 패밀리를 열거하는 메서드가 포함되어 있습니다. 이 클래스에서 빌드된 개체에는 InstalledFontCollection 및 PrivateFontCollection 클래스가 포함됩니다.
FontFamily 이 FontFamily 클래스는 글꼴 패밀리를 구성하는 글꼴 집합을 캡슐화합니다. 글꼴 패밀리는 서체가 같지만 스타일이 다른 글꼴 그룹입니다.
GdiplusBase GdiplusBase 클래스는 GDI+ 개체에 대한 메모리 할당 및 할당 취소를 제공합니다. GdiplusBase는 다른 모든 GDI+ 클래스의 기본 클래스 역할을 하므로 GdiplusBase 인스턴스를 만들 필요가 없습니다.

그래픽 Graphics 클래스는 선, 곡선, 그림, 이미지 및 텍스트를 그리는 메서드를 제공합니다. Graphics 개체는 표시 디바이스의 특성과 그릴 항목의 특성을 저장합니다.
Graphicspath GraphicsPath 개체는 선, 곡선 및 셰이프 시퀀스를 저장합니다.
GraphicsPathIterator 이 GraphicsPathIterator 클래스는 GraphicsPath 개체에 저장된 경로의 선택한 하위 집합을 격리하는 메서드를 제공합니다.
HatchBrush 이 HatchBrush 클래스는 해치 스타일, 전경색 및 배경색을 사용하여 사각형 브러시를 정의합니다.
HueSaturationLightness HueSaturationLightness 클래스를 사용하면 비트맵의 색조, 채도 및 가벼움을 변경할 수 있습니다.
이미지 Image 클래스는 래스터 이미지(비트맵) 및 벡터 이미지(메타파일)를 로드하고 저장하는 메서드를 제공합니다.
ImageAttributes ImageAttributes 개체에는 렌더링 중에 비트맵 및 메타파일 색을 조작하는 방법에 대한 정보가 포함되어 있습니다.
ImageCodecInfo ImageCodecInfo 개체는 이미지 코덱(인코더/디코더)에 대한 정보를 저장합니다.
ImageItemData ImageItemData 클래스는 사용자 지정 이미지 메타데이터를 저장하고 검색하는 데 사용됩니다. Windows GDI+는 JPEG, PNG 및 GIF 이미지 파일에 대한 사용자 지정 메타데이터를 지원합니다.
InstalledFontCollection InstalledFontCollection 클래스는 시스템에 설치된 글꼴을 나타내는 클래스를 정의합니다.
수준 Levels 클래스에는 _highlight, 중간 및 그림자의 세 가지 비트맵 조정이 포함됩니다.

LinearGradientBrush LinearGradientBrush 클래스는 선형 그라데이션 브러시의 시작 경계선에서 선형 그라데이션 브러시의 끝 경계선까지 색이 균등하게 변하는 색 그라데이션을 그리는 브러시를 정의합니다.
행렬 Matrix 개체는 아핀 변환을 나타내는 3 3 행렬을 나타냅니다.
Metafile Metafile 클래스는 그래픽 메타파일을 정의합니다. 메타파일은 그래픽 API 호출 시퀀스를 설명하는 레코드를 포함합니다. 메타파일을 기록(생성)하고 재생(표시)할 수 있습니다.
MetafileHeader MetafileHeader 개체는 연결된 메타파일의 속성을 저장합니다.
PathData PathData 클래스는 GraphicsPath 및 GraphicsPathIterator 클래스에 대한 도우미 클래스입니다.
PathGradientBrush PathGradientBrush 개체는 경로의 내부를 점차 변화하는 색으로 채우는 데 사용할 수 있는 색 그라데이션의 특성을 저장합니다.
펜 펜 개체는 선과 곡선을 그리는 데 사용되는 Windows GDI+ 개체입니다.
Point Point 클래스는 2차원 좌표계의 점을 캡슐화합니다.
Pointf PointF 클래스는 2차원 좌표계의 점을 캡슐화합니다.
PrivateFontCollection PrivateFontCollection은 글꼴에 대한 컬렉션 클래스입니다. 이 클래스는 애플리케이션에 대한 글꼴 컬렉션을 유지합니다. 컬렉션의 글꼴에는 설치된 글꼴과 시스템에 설치되지 않은 글꼴이 포함될 수 있습니다.
PropertyItem PropertyItem 클래스는 Image 및 Bitmap 클래스의 도우미 클래스입니다. PropertyItem 개체는 이미지 메타데이터의 한 부분을 보유합니다.

Rect Rect 개체는 사각형의 왼쪽 위 모서리, 너비 및 높이를 저장합니다.
RectF RectF 개체는 사각형의 왼쪽 위 모서리, 너비 및 높이를 저장합니다.
RedEyeCorrection RedEyeCorrection 클래스를 사용하면 때때로 플래시 사진에서 발생하는 붉은 눈을 수정할 수 있습니다.
지역 Region 클래스는 디스플레이 화면의 영역을 설명합니다.
선명하게 Sharpen 클래스를 사용하면 비트맵의 선명도를 조정할 수 있습니다.
크기 Size 클래스는 2차원 좌표계의 Width 및 Height 차원을 캡슐화합니다.
Sizef SizeF 클래스는 2차원 좌표계의 Width 및 Height 차원을 캡슐화합니다. SizeF 클래스는 부동 소수점 좌표를 사용합니다.
SolidBrush SolidBrush 클래스는 단색 Brush 개체를 정의합니다. Brush 개체는 페인트 브러시가 도형 내부를 칠할 수 있는 방식과 유사한 도형을 채우는 데 사용됩니다. 이 클래스는 Brush 추상 기본 클래스에서 상속됩니다.
StringFormat StringFormat 클래스는 텍스트 레이아웃 정보(예: 맞춤, 방향, 탭 정지 및 클리핑)와 표시 조작(예: 트리밍, 요청된 글꼴에서 지원되지 않는 문자의 글꼴 대체 및 서유럽 숫자를 사용하지 않는 언어에 대한 숫자 대체)을 캡슐화합니다. StringFormat 개체를 DrawString Methods 메서드에 전달하여 문자열 형식을 지정할 수 있습니다.
TextureBrush TextureBrush 클래스는 채우기에 사용되는 Image 개체를 포함하는 Brush 개체를 정의합니다. Fill 이미지는 Brush 개체에 포함된 로컬 Matrix 개체를 사용하여 변환할 수 있습니다.

색조 Tint 클래스를 사용하면 비트맵에 색조를 적용할 수 있습니다.

열거형

BrushType BrushType 열거형은 브러시 유형을 나타냅니다. 브러시에는 5가지 유형이 있습니다.
ColorAdjustType ColorAdjustType 열거형은 색 조정 정보를 사용하는 GDI+ 개체를 지정합니다.
ColorChannelFlags ColorChannelFlags 열거형은 CMYK(cyan, magenta, yellow, black) 색 공간에서 개별 채널을 지정합니다. 이 열거형은 ImageAttributes::SetOutputChannel 메서드에서 사용됩니다.
ColorMatrixFlags ColorMatrixFlags 열거형은 ImageAttributes 개체의 색 및 회색조 조정 설정의 영향을 받는 이미지 및 색 유형을 지정합니다.
CombineMode CombineMode 열거형은 새 지역이 기존 지역과 결합되는 방법을 지정합니다.
CompositingMode CompositingMode 열거형은 렌더링된 색을 배경색과 결합하는 방법을 지정합니다. 이 열거형은 Graphics 클래스의 Graphics::GetCompositingMode 및 Graphics::SetCompositingMode 메서드에서 사용됩니다.
CompositingQuality CompositingQuality 열거형은 색이 배경색과 혼합될 때 감마 보정이 적용되는지 여부를 지정합니다.
CoordinateSpace CoordinateSpace 열거형은 좌표 공간을 지정합니다.

CurveAdjustments ColorCurve 클래스는 CurveAdjustments 열거형에 나열된 8개의 비트맵 조정을 포함합니다.
CurveChannel CurveChannel 열거형은 ColorCurve 비트맵 조정의 영향을 받는 색 채널을 지정합니다.
DashCap DashCap 열거형은 파선의 각 대시 양쪽 끝에 사용할 그래픽 셰이프의 유형을 지정합니다.
DashStyle DashStyle 열거형은 Windows GDI+ 펜으로 그린 선의 선 스타일을 지정합니다. 선은 미리 정의된 여러 스타일 또는 사용자 지정 스타일 중 하나를 사용하여 그릴 수 있습니다.
DebugEventLevel 디버그 이벤트 수준을 지정하는 상수를 정의합니다.
DitherType DitherType 열거형은 비트맵을 변환할 때 디더링에 사용할 수 있는 알고리즘을 식별합니다.
DriverStringOptions DriverStringOptions 열거형은 드라이버 문자열에 대한 렌더링의 간격, 방향 및 품질을 지정합니다.
EmfPlusRecordType EmfPlusRecordType 열거형은 WMF(Windows 메타파일 형식), EMF(Enhanced Metafile) 및 EMF+ 파일에 사용되는 메타파일 레코드 형식을 식별합니다. EmfPlusRecordType 열거형의 요소는 세 개의 그룹으로 제공됩니다.
EmfToWmfBitsFlags EMF(Enhanced Metafile) 메타파일을 WMF(Windows 메타파일 형식) 메타파일로 변환하는 Metafile::EmfToWmfBits 메서드에 대한 옵션을 지정합니다.
EmfType EmfType 열거형은 EMF(Enhanced Metafile) 파일에 배치되는 레코드의 특성을 지정합니다. 이 열거형은 Metafile 클래스의 여러 생성자에서 사용됩니다.
EncoderParameterValueType EncoderParameterValueType 열거형은 이미지 코덱(인코더/디코더) 매개 변수에 대한 데이터 형식을 지정합니다. 이 열거형의 요소는 EncoderParameter 개체의 Type 데이터 멤버에 할당됩니다.

EncoderValue EncoderValue 열거형은 이미지 인코더에 인수로 전달할 수 있는 값을 지정합니다. 이미지 인코더에 대한 자세한 내용은 이미지 인코더 및 디코더 사용을 참조하세요.
FillMode FillMode 열거형은 경로 또는 곡선이 자체적으로 교차할 때 형성되는 영역을 채우는 방법을 지정합니다.
FlushIntention FlushIntention 열거형은 그래픽 작업의 큐를 플러시할 시기를 지정합니다.
FontStyle FontStyle 열거형은 글꼴 서체의 스타일을 지정합니다. 스타일을 결합할 수 있습니다.
GdiplusStartupParams 시작에 대한 매개 변수를 지정하는 상수를 정의합니다.
HatchStyle HatchStyle 열거형은 HatchBrush 형식의 브러시에서 사용하는 해치 패턴을 지정합니다. 빗살 무늬 배경색 및 배경 위에 그리는 선의 구성 됩니다.
히스토그램Format HistogramFormat 열거형은 비트맵의 색 채널을 나타내는 히스토그램의 수와 형식을 지정합니다. 이 열거형은 Bitmap::GetHistogram 메서드와 함께 사용됩니다.
HotkeyPrefix HotkeyPrefix 열거형은 핫 키를 표시하는 방법을 지정합니다. 세 가지 옵션이 있습니다. 아무것도 _do 핫 키 밑줄을 표시하고 핫 키 밑줄을 숨깁니다.
ImageCodecFlags ImageCodecFlags 열거형은 이미지 코덱의 특성을 나타냅니다.
ImageFlags ImageFlags 열거형은 Image 개체에 포함된 픽셀 데이터의 특성을 지정합니다. Image::GetFlags 메서드는 이 열거형의 요소를 반환합니다.

ImageLockMode ImageLockMode 열거형은 Bitmap::LockBits 메서드의 flags 매개 변수에 전달되는 플래그를 지정합니다. Bitmap::LockBits 메서드는 픽셀 데이터를 읽거나 쓸 수 있도록 이미지의 일부를 잠급니다.
ImageType ImageType 열거형은 이미지가 비트맵인지 메타파일인지를 나타냅니다. Image::GetType 메서드는 이 열거형의 요소를 반환합니다.
보간Mode InterpolationMode 열거형은 이미지 크기 조정 또는 회전 시 사용되는 알고리즘을 지정합니다. 이 열거형은 Graphics 클래스의 Graphics::GetInterpolationMode 및 Graphics::SetInterpolationMode 메서드에서 사용됩니다.
ItemDataPosition ItemDataPosition 열거형은 이미지 파일에서 사용자 지정 메타데이터의 위치를 지정하는 데 사용됩니다.
LinearGradientMode LinearGradientMode 열거형은 선형 그라데이션 브러시의 색 변경이 발생하는 방향을 지정합니다.
LineCap LineCap 열거형은 Windows GDI+ 펜으로 그린 선의 끝에 사용할 그래픽 셰이프의 유형을 지정합니다.
LineJoin LineJoin 열거형은 동일한 펜으로 그려지고 끝이 만나는 두 줄을 조인하는 방법을 지정합니다. 두 줄 끝의 교집합에서 선 조인은 조인을 더 연속적으로 보이게 합니다.
MatrixOrder MatrixOrder 열거형은 새 행렬에 기존 행렬을 곱할 때 곱하기 순서를 지정합니다.
MetafileFrameUnit MetafileFrameUnit 열거형은 메타파일 프레임 사각형에 대한 측정 단위를 지정합니다.
MetafileType MetafileType 열거형은 메타파일의 형식을 지정합니다. MetafileHeader::GetType 메서드는 이 열거형의 요소를 반환합니다.

ObjectType ObjectType 열거형은 EMF+ 레코드의 개체 형식 값을 나타냅니다.
PaletteFlags PaletteFlags 열거형은 색상표에 있는 색 데이터의 특성을 나타냅니다.
PaletteType PaletteType 열거형은 Bitmap 클래스의 Bitmap::InitializePalette 및 Bitmap::ConvertFormat 메서드에서 사용됩니다. 열거형의 멤버는 여러 표준 색상표 형식을 식별합니다.
PathPointType PathPointType 열거형은 경로의 데이터 요소에 대한 점 형식 및 플래그를 나타냅니다.
PenAlignment PenAlignment 열거형은 그리는 스트로크를 기준으로 펜의 맞춤을 지정합니다.
PenType PenType 열거형은 펜이 그리는 패턴, 질감 또는 그라데이션의 형식을 나타냅니다.
PixelOffsetMode PixelOffsetMode 열거형은 Graphics 개체의 픽셀 오프셋 모드를 지정합니다. 이 열거형은 Graphics 클래스의 Graphics::GetPixelOffsetMode 및 Graphics::SetPixelOffsetMode 메서드에서 사용됩니다.
RotateFlipType RotateFlipType 열거형은 이미지 회전 방향과 이미지를 대칭 이동하는 데 사용되는 축을 지정합니다.
SmoothingMode SmoothingMode 열거형은 선과 곡선에 적용되는 스무딩(앤티앨리어싱)의 유형을 지정합니다. 이 열거형은 Graphics 클래스의 Graphics::GetSmoothingMode 및 Graphics::SetSmoothingMode 메서드에서 사용됩니다.
Status 상태 열거형은 Windows GDI+ 메서드 호출의 결과를 나타냅니다.
StringAlignment StringAlignment 열거형은 경계 사각형을 참조하여 문자열을 정렬하는 방법을 지정합니다. 경계 사각형은 텍스트가 표시되는 영역을 정의하는 데 사용됩니다.

StringDigitSubstitute StringDigitSubstitute 열거형은 사용자의 로캘 또는 언어에 따라 문자열의 숫자를 대체하는 방법을 지정합니다.
StringFormatFlags StringFormatFlags 열거형은 텍스트 레이아웃 정보(예: 방향 및 클리핑) 및 표시 조작(예: 줄임표 삽입, 숫자 대체 및 글꼴에서 지원되지 않는 문자 표현)을 지정합니다.
StringTrimming StringTrimming 열거형은 문자열이 레이아웃 사각형에 맞도록 문자열에서 문자를 자르는 방법을 지정합니다. 레이아웃 사각형은 표시 문자열의 위치를 지정하고 크기를 조정하는 데 사용됩니다.
TextRenderingHint TextRenderingHint 열거형은 텍스트를 렌더링하는 데 사용되는 프로세스를 지정합니다. 이 프로세스는 텍스트의 품질에 영향을 줍니다.
단위 단위 열거형은 지정된 데이터 형식에 대한 측정 단위를 지정합니다.
WarpMode WarpMode 열거형은 이미지를 변환하는 데 사용할 수 있는 워프 모드를 지정합니다.
Wrapmode WrapMode 열거형은 이미지의 반복 복사본을 사용하여 영역을 타일로 지정하는 방법을 지정합니다.

Functions

~PathData **PathData** 클래스에 대한 소멸자입니다.
중단 *Abort*는 시간이 많이 걸리는 렌더링 작업 중에 Windows GDI+에서 주기적으로 호출되는 애플리케이션 정의 메서드입니다. **GdiplusAbort** 구조를 참조하세요.

AddArc GraphicsPath::AddArc 메서드는 이 경로의 현재 그림에 타원형 호를 추가합니다. (오버로드 1/4)
AddArc GraphicsPath::AddArc 메서드는 이 경로의 현재 그림에 타원형 호를 추가합니다. (오버로드 4/4)
AddArc GraphicsPath::AddArc 메서드는 이 경로의 현재 그림에 타원형 호를 추가합니다. (오버로드 3/4)
AddArc GraphicsPath::AddArc 메서드는 이 경로의 현재 그림에 타원형 호를 추가합니다. (오버로드 2/4)
AddBezier GraphicsPath::AddBezier 메서드는 이 경로의 현재 그림에 B zier 스플라인을 추가합니다. (오버로드 2/3)
AddBezier GraphicsPath::AddBezier 메서드는 이 경로의 현재 그림에 Bezier 스플라인을 추가합니다.
AddBezier GraphicsPath::AddBezier 메서드는 이 경로의 현재 그림에 B zier 스플라인을 추가합니다. (오버로드 3/3)
AddBezier GraphicsPath::AddBezier 메서드는 이 경로의 현재 그림에 B zier 스플라인을 추가합니다. (오버로드 1/3)
AddBeziers GraphicsPath::AddBeziers 메서드는 이 경로의 현재 그림에 연결된 B zier 스플라인 시퀀스를 추가합니다.
AddBeziers GraphicsPath::AddBeziers 메서드는 연결된 Bezier 스플라인 시퀀스를 이 경로의 현재 그림에 추가합니다.
AddClosedCurve GraphicsPath::AddClosedCurve 메서드는 닫힌 카디널 스플라인을 이 경로에 추가합니다. (오버로드 2/4)

AddClosedCurve GraphicsPath::AddClosedCurve 메서드는 닫힌 카디널 스플라인을 이 경로에 추가합니다. (오버로드 4/4)
AddClosedCurve GraphicsPath::AddClosedCurve 메서드는 닫힌 카디널 스플라인을 이 경로에 추가합니다. (오버로드 1/4)
AddClosedCurve GraphicsPath::AddClosedCurve 메서드는 닫힌 카디널 스플라인을 이 경로에 추가합니다. (오버로드 3/4)
AddCurve GraphicsPath::AddCurve 메서드는 이 경로의 현재 그림에 카디널 스플라인을 추가합니다. (오버로드 5/6)
AddCurve GraphicsPath::AddCurve 메서드는 이 경로의 현재 그림에 카디널 스플라인을 추가합니다. (오버로드 6/6)
AddCurve GraphicsPath::AddCurve 메서드는 이 경로의 현재 그림에 카디널 스플라인을 추가합니다. (오버로드 3/6)
AddCurve GraphicsPath::AddCurve 메서드는 이 경로의 현재 그림에 카디널 스플라인을 추가합니다. (오버로드 2/6)
AddCurve GraphicsPath::AddCurve 메서드는 이 경로의 현재 그림에 카디널 스플라인을 추가합니다. (오버로드 1/6)
AddCurve GraphicsPath::AddCurve 메서드는 이 경로의 현재 그림에 카디널 스플라인을 추가합니다. (오버로드 4/6)
AddEllipse GraphicsPath::AddEllipse 메서드는 이 경로에 타원을 추가합니다. (오버로드 4/4)

AddEllipse GraphicsPath::AddEllipse 메서드는 이 경로에 타원을 추가합니다. (오버로드 2/4)
AddEllipse GraphicsPath::AddEllipse 메서드는 이 경로에 타원을 추가합니다. (오버로드 3/4)
AddEllipse GraphicsPath::AddEllipse 메서드는 이 경로에 타원을 추가합니다. (오버로드 1/4)
AddFontFile PrivateFontCollection::AddFontFile 메서드는 이 개인 글꼴 컬렉션에 글꼴 파일을 추가합니다.
AddLine GraphicsPath::AddLine 메서드는 이 경로의 현재 그림에 선을 추가합니다. (오버로드 2/4)
AddLine GraphicsPath::AddLine 메서드는 이 경로의 현재 그림에 선을 추가합니다. (오버로드 4/4)
AddLine GraphicsPath::AddLine 메서드는 이 경로의 현재 그림에 선을 추가합니다. (오버로드 3/4)
AddLine GraphicsPath::AddLine 메서드는 이 경로의 현재 그림에 선을 추가합니다. (오버로드 1/4)
AddLines GraphicsPath::AddLines 메서드는 이 경로의 현재 그림에 연결된 선 시퀀스를 추가합니다. (오버로드 2/2)
AddLines GraphicsPath::AddLines 메서드는 이 경로의 현재 그림에 연결된 선 시퀀스를 추가합니다. (오버로드 1/2)
AddMemoryFont PrivateFontCollection::AddMemoryFont 메서드는 시스템 메모리에 포함된 글꼴을 Windows GDI+ 글꼴 컬렉션에 추가합니다.

AddMetafileComment Graphics::AddMetafileComment 메서드는 기존 메타파일에 텍스트 주석을 추가합니다.
AddPath GraphicsPath::AddPath 메서드는 이 경로에 경로를 추가합니다.
AddPie GraphicsPath::AddPie 메서드는 이 경로에 원형을 추가합니다. (오버로드 1/4)
AddPie GraphicsPath::AddPie 메서드는 이 경로에 원형을 추가합니다. (오버로드 4/4)
AddPie GraphicsPath::AddPie 메서드는 이 경로에 원형을 추가합니다. (오버로드 3/4)
AddPie GraphicsPath::AddPie 메서드는 이 경로에 원형을 추가합니다. (오버로드 2/4)
AddPolygon GraphicsPath::AddPolygon 메서드는 이 경로에 다각형을 추가합니다. (오버로드 2/2)
AddPolygon GraphicsPath::AddPolygon 메서드는 이 경로에 다각형을 추가합니다. (오버로드 1/2)
AddRectangle GraphicsPath::AddRectangle 메서드는 이 경로에 사각형을 추가합니다. (오버로드 1/2)
AddRectangle GraphicsPath::AddRectangle 메서드는 이 경로에 사각형을 추가합니다. (오버로드 2/2)
AddRectangles GraphicsPath::AddRectangles 메서드는 이 경로에 직사각형 시퀀스를 추가합니다.
AddRectangles GraphicsPath::AddRectangles 메서드는 이 경로에 직사각형 시퀀스를 추가합니다.

AddString GraphicsPath::AddString 메서드는 문자열의 윤곽선을 이 경로에 추가합니다.
AddString GraphicsPath::AddString 메서드는 문자열의 윤곽선을 이 경로에 추가합니다. (오버로드 1/3)
AddString GraphicsPath::AddString 메서드는 문자열의 윤곽선을 이 경로에 추가합니다. (오버로드 3/3)
AddString GraphicsPath::AddString 메서드는 문자열의 윤곽선을 이 경로에 추가합니다. (오버로드 2/3)
AdjustableArrowCap 지정된 높이와 너비를 사용하여 조정 가능한 화살표 선 뚜껑을 만듭니다. 화살표 선 뚜껑은 채워지거나 채워지지 않을 수 있습니다. 중간 inset의 기본값은 0입니다.
ApplyEffect Bitmap::ApplyEffect 메서드는 기존 Bitmap 개체에 지정된 효과를 적용하여 새 Bitmap 개체를 만듭니다.
ApplyEffect Bitmap::ApplyEffect 메서드는 지정된 효과를 적용하여 이 Bitmap 개체를 변경합니다.
BeginContainer Graphics::BeginContainer 메서드는 새 그래픽 컨테이너를 시작합니다. (오버로드 1/2)
BeginContainer Graphics::BeginContainer 메서드는 새 그래픽 컨테이너를 시작합니다. (오버로드 2/2)
Bitmap BITMAPINFO 구조체 및 픽셀 데이터 배열을 기반으로 Bitmap::Bitmap 개체를 만듭니다.
Bitmap 이미지 파일을 기반으로 Bitmap::Bitmap 개체를 만듭니다.

Bitmap Windows GDI(Windows Graphics Device Interface) 비트맵에 대한 핸들과 GDI 팔레트에 대한 핸들을 기반으로 Bitmap::Bitmap 개체를 만듭니다.
Bitmap 아이콘을 기반으로 Bitmap 개체를 만듭니다.
Bitmap 애플리케이션 또는 DLL 인스턴스 핸들과 비트맵 리소스의 이름을 기반으로 Bitmap::Bitmap 개체를 만듭니다.
Bitmap DirectDraw 표면을 기반으로 Bitmap::Bitmap 개체를 만듭니다. Bitmap::Bitmap 개체는 Bitmap::Bitmap 개체가 삭제되거나 범위를 벗어날 때까지 DirectDraw 표면에 대한 참조를 유지합니다.
Bitmap Graphics 개체, 너비 및 높이를 기반으로 Bitmap::Bitmap 개체를 만듭니다.
Bitmap 크기 및 서식 정보와 함께 바이트 배열을 기반으로 Bitmap::Bitmap 개체를 만듭니다.
Bitmap 지정된 크기 및 픽셀 형식의 Bitmap::Bitmap 개체를 만듭니다. Bitmap::Bitmap 개체가 생성된 후에 픽셀 데이터를 제공해야 합니다.
Bitmap IStream COM 인터페이스를 기반으로 Bitmap::Bitmap 개체를 만듭니다.
흐리게 Blur 개체를 만듭니다.
BrightnessContrast 새 BrightnessContrast 개체를 만듭니다.
CachedBitmap Bitmap 개체 및 Graphics 개체를 기반으로 CachedBitmap::CachedBitmap 개체를 만듭니다.

CachedBitmap **CachedBitmap**에 대한 복사 생성자입니다.
CharacterRange CharacterRange::CharacterRange 개체를 만들고 데이터 멤버를 지정된 값으로 초기화합니다.
CharacterRange 데이터 멤버가 0으로 설정된 CharacterRange::CharacterRange 개체를 만듭니다.
지우기 Graphics::Clear 메서드는 Graphicsobject를 지정된 색으로 지웁니다.
ClearBrushRemapTable ImageAttributes::ClearBrushRemapTable 메서드는 이 ImageAttributes 개체의 브러시 색 다시 매핑 테이블을 지웁니다.
ClearColorKey ImageAttributes::ClearColorKey 메서드는 지정된 범주에 대한 색 키(투명도 범위)를 지웁니다.
ClearColorMatrices ImageAttributes::ClearColorMatrices 메서드는 지정된 범주에 대한 색 조정 행렬과 회색조 조정 행렬을 지웁니다.
ClearColorMatrix ImageAttributes::ClearColorMatrix 메서드는 지정된 범주에 대한 색 조정 행렬을 지웁니다.
ClearGamma ImageAttributes::ClearGamma 메서드는 지정된 범주에 대해 감마 수정을 사용하지 않도록 설정합니다.
ClearMarkers GraphicsPath::ClearMarkers 메서드는 이 경로에서 표식을 지웁니다.
ClearNoOp ImageAttributes::ClearNoOp 메서드는 지정된 범주에 대한 NoOp 설정을 지웁니다.

ClearOutputChannel ImageAttributes::ClearOutputChannel 메서드는 지정된 범주에 대한 CMYK(cyan-magenta-yellow-black) 출력 채널 설정을 지웁니다.
ClearOutputChannelColorProfile ImageAttributes::ClearOutputChannelColorProfile 메서드는 지정된 범주에 대한 출력 채널 색 프로필 설정을 지웁니다.
ClearRemapTable ImageAttributes::ClearRemapTable 메서드는 지정된 범주에 대한 색 다시 매핑 테이블을 지웁니다.
ClearThreshold ImageAttributes::ClearThreshold 메서드는 지정된 범주의 임계값을 지웁니다.
복제 Bitmap::Clone 메서드는 이 비트맵의 일부를 복사하여 새 Bitmap 개체를 만듭니다. (오버로드 1/2)
복제 Bitmap::Clone 메서드는 이 비트맵의 일부를 복사하여 새 Bitmap 개체를 만듭니다. (오버로드 2/2)
복제 Bitmap::Clone 메서드는 이 비트맵의 일부를 복사하여 새 Bitmapobject를 만듭니다. (오버로드 1/2)
복제 Bitmap::Clone 메서드는 이 비트맵의 일부를 복사하여 새 Bitmapobject를 만듭니다. (오버로드 2/2)
복제 CustomLineCap::Clone 메서드는 기존 개체의 내용을 새 CustomLineCap 개체에 복사합니다.
복제 Font::Clone 메서드는 이 Font 개체를 기반으로 새 Font 개체를 만듭니다.
복제 FontFamily::Clone 메서드는 이 FontFamily 개체를 기반으로 새 FontFamily 개체를 만듭니다.
복제 Image::Clone 메서드는 새 Image 개체를 만들고 이 Image 개체의 내용으로 초기화합니다.

복제 Region::Clone 메서드는 이 Regionobject의 복사본을 만들고 새 Regionobject의 주소를 반환합니다.
복제 Brush::Clone 메서드는 이 브러시를 기반으로 새 Brush 개체를 만듭니다.
복제 Rect::Clone 메서드는 새 Rect 개체를 만들고 이 Rect 개체의 내용으로 초기화합니다.
복제 RectF::Clone 메서드는 새 RectF 개체를 만들고 이 RectF 개체의 내용으로 초기화합니다.
복제 GraphicsPath::Clone 메서드는 새 GraphicsPath 개체를 만들고 이 GraphicsPath 개체의 내용으로 초기화합니다.
복제 ImageAttributes::Clone 메서드는 이 ImageAttributes 개체의 복사본을 만듭니다.
복제 Matrix::Clone 메서드는 이 Matrix 개체의 복사본인 새 Matrix 개체를 만듭니다.
복제 Pen::Clone 메서드는 펜 개체를 복사합니다.
복제 StringFormat::Clone 메서드는 새 StringFormat 개체를 만들고 이 StringFormat 개체의 내용으로 초기화합니다.
CloseAllFigures GraphicsPath::CloseAllFigures 메서드는 이 경로의 열려 있는 모든 그림을 닫습니다.
CloseFigure GraphicsPath::CloseFigure 메서드는 이 경로의 현재 그림을 닫습니다.

색상 Color::Color 개체를 만들고 불투명 검정으로 초기화합니다. 이것이 기본 생성자입니다.
색상 ARGB 값을 사용하여 Color::Color 개체를 만듭니다.
색상 빨강, 녹색 및 파랑 구성 요소에 대해 지정된 값을 사용하여 Color::Color 개체를 만듭니다. 이 생성자는 알파 구성 요소를 255(불투명)로 설정합니다.
색상 알파, 빨강, 녹색 및 파랑 구성 요소에 대해 지정된 값을 사용하여 Color::Color 개체를 만듭니다.
ColorBalance 새 ColorBalance 개체를 만듭니다.
ColorCurve ColorCurve 개체를 만듭니다.
ColorLUT 새 ColorLUT 개체를 만듭니다.
ColorMatrixEffect ColorMatrixEffect 개체를 만듭니다.
보완 Region::Complement 메서드는 이 지역을 이 지역과 교차하지 않는 지정된 경로의 내부 부분으로 업데이트합니다.
보완 Region::Complement 메서드는 이 지역과 교차하지 않는 지역을 업데이트합니다.
보완 Region::Complement 메서드는 이 지역을 이 지역과 교차하지 않는 지정된 사각형 내부 부분으로 업데이트합니다.

보완 Region::Complement 메서드는 이 지역을 이 지역과 교차하지 않는 다른 지역의 부분으로 업데이트합니다.
포함 Rect::Contains 메서드는 점이 이 사각형 내에 있는지 여부를 결정합니다.
포함 이 항목에서는 Rect 클래스의 Contains 메서드를 나열합니다. Rect 클래스에 대한 메서드의 전체 목록은 Rect 메서드를 참조하세요. (오버로드 1/2)
포함 이 항목에서는 Rect 클래스의 Contains 메서드를 나열합니다. Rect 클래스에 대한 메서드의 전체 목록은 Rect 메서드를 참조하세요. (오버로드 2/2)
포함 RectF::Contains 메서드는 점이 이 사각형 내에 있는지 여부를 결정합니다.
포함 RectF::Contains 메서드는 다른 사각형이 이 사각형 내에 있는지 여부를 결정합니다.
포함 이 항목에서는 RectF 클래스의 Contains 메서드를 나열합니다. RectF 클래스에 대한 메서드의 전체 목록은 RectF 메서드를 참조하세요.
ConvertFormat Bitmap::ConvertFormat 메서드는 비트맵을 지정된 픽셀 형식으로 변환합니다. 비트맵의 원래 픽셀 데이터는 새 픽셀 데이터로 대체됩니다.
ConvertToEmfPlus Metafile::ConvertToEmfPlus 메서드는 이 Metafile 개체를 EMF+ 형식으로 변환합니다. (오버로드 3/3)
ConvertToEmfPlus Metafile::ConvertToEmfPlus 메서드는 이 Metafile 개체를 EMF+ 형식으로 변환합니다. (오버로드 1/3)

ConvertToEmfPlus Metafile::ConvertToEmfPlus 메서드는 이 Metafile 개체를 EMF+ 형식으로 변환합니다. (오버로드 2/3)
CopyData GraphicsPathIterator::CopyData 메서드는 경로 데이터 요소의 하위 집합을 PointF 배열에 복사하고 경로의 점 형식 하위 집합을 BYTE 배열에 복사합니다.
CustomLineCap **CustomLineCap**에 대한 생성자를 복사합니다.
CustomLineCap CustomLineCap::CustomLineCap 개체를 만듭니다. (오버로드 1/2)
CustomLineCap CustomLineCap::CustomLineCap 개체를 만듭니다. (오버로드 2/2)
DebugEventProc *DebugEventProc*는 애플리케이션에서 구현하고 **GdiplusStartupInput**의 생성자에 전달하는 콜백 함수의 서명입니다.
DrawArc Graphics::D rawArc 메서드는 호를 그립니다. 호는 타원의 일부입니다. (오버로드 3/3)
DrawArc Graphics::D rawArc 메서드는 호를 그립니다. 호는 타원의 일부입니다. (오버로드 1/3)
DrawArc Graphics::D rawArc 메서드는 호를 그립니다. 호는 타원의 일부입니다. (오버로드 2/3)
DrawArc Graphics::D rawArc 메서드는 호를 그립니다.
DrawBezier Graphics::D rawBezier 메서드는 B zier 스플라인을 그립니다. (오버로드 1/3)

DrawBezier Graphics::D rawBezier 메서드는 B zier 스플라인을 그립니다. (오버로드 3/3)
DrawBezier Graphics::D rawBezier 메서드는 B zier 스플라인을 그립니다. (오버로드 2/3)
DrawBezier Graphics::D rawBezier 메서드는 Bezier 스플라인을 그립니다.
DrawBeziers Graphics::D rawBeziers 메서드는 연결된 B zier 스플라인 시퀀스를 그립니다.
DrawBeziers Graphics::D rawBeziers 메서드는 연결된 Bezier 스플라인 시퀀스를 그립니다.
DrawCachedBitmap Graphics::D rawCachedBitmap 메서드는 CachedBitmap 개체에 저장된 이미지를 그립니다.
DrawClosedCurve Graphics::D rawClosedCurve 메서드는 닫힌 카디널 스플라인을 그립니다. (오버로드 2/4)
DrawClosedCurve Graphics::D rawClosedCurve 메서드는 닫힌 카디널 스플라인을 그립니다. (오버로드 4/4)
DrawClosedCurve Graphics::D rawClosedCurve 메서드는 닫힌 카디널 스플라인을 그립니다. (오버로드 3/4)
DrawClosedCurve Graphics::D rawClosedCurve 메서드는 닫힌 카디널 스플라인을 그립니다. (오버로드 1/4)
DrawCurve Graphics::D rawCurve 메서드는 카디널 스플라인을 그립니다. (오버로드 1/6)
DrawCurve Graphics::D rawCurve 메서드는 카디널 스플라인을 그립니다. (오버로드 3/6)

DrawCurve Graphics::D rawCurve 메서드는 카디널 스플라인을 그립니다. (오버로드 5/6)
DrawCurve Graphics::D rawCurve 메서드는 카디널 스플라인을 그립니다. (오버로드 2/6)
DrawCurve Graphics::D rawCurve 메서드는 카디널 스플라인을 그립니다. (오버로드 6/6)
DrawCurve Graphics::D rawCurve 메서드는 카디널 스플라인을 그립니다. (오버로드 4/6)
DrawDriverString Graphics::D rawDriverString 메서드는 지정된 위치에 문자를 그립니다. 메서드는 클라이언트가 텍스트의 모양을 완벽하게 제어할 수 있게 해줍니다. 메서드는 클라이언트가 적용할 형식 및 레이아웃을 이미 설정했음을 가정합니다.
DrawEllipse Graphics::D rawEllipse 메서드는 타원을 그립니다. (오버로드 2/4)
DrawEllipse Graphics::D rawEllipse 메서드는 타원을 그립니다. (오버로드 1/4)
DrawEllipse Graphics::D rawEllipse 메서드는 타원을 그립니다. (오버로드 3/4)
DrawEllipse Graphics::D rawEllipse 메서드는 타원을 그립니다. (오버로드 4/4)
Drawimage Graphics::D rawImage 메서드는 이미지를 그립니다. (오버로드 1/14)
Drawimage Graphics::D rawImage 메서드는 이미지를 그립니다. (오버로드 7/14)
Drawimage Graphics::D rawImage 메서드는 이미지를 그립니다. (오버로드 6/14)

DrawImage Graphics::DrawImage 메서드는 이미지를 그립니다. (오버로드 3/14)
DrawImage Graphics::DrawImage 메서드는 이미지를 그립니다. (오버로드 5/14)
DrawImage Graphics::DrawImage 메서드는 이미지를 그립니다. (오버로드 2/14)
DrawImage Graphics::DrawImage 메서드는 이미지를 그립니다. (오버로드 11/14)
DrawImage Graphics::DrawImage 메서드는 이미지를 그립니다. (오버로드 8/14)
DrawImage Graphics::DrawImage 메서드는 이미지를 그립니다. (오버로드 9/14)
DrawImage Graphics::DrawImage 메서드는 지정된 위치에 이미지의 지정된 부분을 그립니다.
DrawImage Graphics::DrawImage 메서드는 이미지를 그립니다. (오버로드 12/14)
DrawImage Graphics::DrawImage 메서드는 지정된 위치에 이미지를 그립니다. (오버로드 2/2)
DrawImage Graphics::DrawImage 메서드는 이미지를 그립니다. (오버로드 14/14)
DrawImage Graphics::DrawImage 메서드는 이미지를 그립니다. (오버로드 10/14)
DrawImage Graphics::DrawImage 메서드는 지정된 위치에 이미지를 그립니다. (오버로드 1/2)

DrawImage Graphics::DrawImage 메서드는 이미지를 그립니다. (오버로드 13/14)
DrawImage Graphics::DrawImage 메서드는 이미지를 그립니다. (오버로드 4/14)
DrawImage 메서드는 지정된 효과를 적용한 후 이미지의 일부를 그립니다.
DrawLine Graphics::DrawLine 메서드는 두 점을 연결하는 선을 그립니다. (오버로드 4/4)
DrawLine Graphics::DrawLine 메서드는 두 점을 연결하는 선을 그립니다. (오버로드 3/4)
DrawLine Graphics::DrawLine 메서드는 두 점을 연결하는 선을 그립니다. (오버로드 2/4)
DrawLine Graphics::DrawLine 메서드는 두 점을 연결하는 선을 그립니다. (오버로드 1/4)
DrawLines Graphics::DrawLines 메서드는 연결된 선 시퀀스를 그립니다. (오버로드 2/2)
DrawLines Graphics::DrawLines 메서드는 연결된 선 시퀀스를 그립니다. (오버로드 1/2)
DrawPath Graphics::DrawPath 메서드는 GraphicsPath 개체로 정의된 선과 곡선 시퀀스를 그립니다.
DrawPie Graphics::DrawPie 메서드는 원형을 그립니다. (오버로드 2/4)
DrawPie Graphics::DrawPie 메서드는 원형을 그립니다. (오버로드 1/4)

DrawPie Graphics::D rawPie 메서드는 원형을 그립니다. (오버로드 3/4)
DrawPie Graphics::D rawPie 메서드는 원형을 그립니다. (오버로드 4/4)
DrawPolygon Graphics::D rawPolygon 메서드는 다각형을 그립니다. (오버로드 2/2)
DrawPolygon Graphics::D rawPolygon 메서드는 다각형을 그립니다. (오버로드 1/2)
DrawRectangle Graphics::D rawRectangle 메서드는 사각형을 그립니다. (오버로드 2/4)
DrawRectangle Graphics::D rawRectangle 메서드는 사각형을 그립니다. (오버로드 4/4)
DrawRectangle Graphics::D rawRectangle 메서드는 사각형을 그립니다. (오버로드 3/4)
DrawRectangle Graphics::D rawRectangle 메서드는 사각형을 그립니다. (오버로드 1/4)
DrawRectangles Graphics::D rawRectangles 메서드는 직사각형 시퀀스를 그립니다. (오버로드 1/2)
DrawRectangles Graphics::D rawRectangles 메서드는 직사각형 시퀀스를 그립니다. (오버로드 2/2)
Drawstring Graphics::D rawString 메서드는 글꼴 및 문자열의 원본을 기반으로 문자열을 그립니다.
Drawstring Graphics::D rawString 메서드는 글꼴, 문자열 원본 및 형식을 기반으로 문자열을 그립니다.

Drawstring Graphics::D rawString 메서드는 글꼴, 레이아웃 사각형 및 형식을 기반으로 문자열을 그립니다.
효과 Effect 개체를 만듭니다.
EmfToWmfBits 고급 형식 메타파일을 WMF(Windows 메타파일 형식) 메타파일로 변환하고 변환된 레코드를 지정된 버퍼에 저장합니다.
비어 있음 Size::Empty 메서드는 Size 개체가 비어 있는지 여부를 결정합니다.
비어 있음 SizeF::Empty 메서드는 SizeF 개체가 비어 있는지 여부를 결정합니다.
EndContainer Graphics::EndContainer 메서드는 Graphics::BeginContainer 메서드에서 이전에 연 그래픽 컨테이너를 닫습니다.
열거 GraphicsPathIterator::Enumerate 메서드는 경로의 데이터 요소를 PointF 배열에 복사하고 경로의 점 형식을 BYTE 배열에 복사합니다.
EnumerateMetafile Graphics::EnumerateMetafile 메서드는 지정된 메타파일의 각 레코드에 대해 애플리케이션 정의 콜백 함수를 호출합니다. 이 메서드를 사용하여 콜백 함수에서 PlayRecord를 호출하여 메타파일을 표시할 수 있습니다. (오버로드 6/12)
EnumerateMetafile Graphics::EnumerateMetafile 메서드는 지정된 메타파일의 각 레코드에 대해 애플리케이션 정의 콜백 함수를 호출합니다. 이 메서드를 사용하여 콜백 함수에서 PlayRecord를 호출하여 메타파일을 표시할 수 있습니다. (오버로드 7/12)
EnumerateMetafile Graphics::EnumerateMetafile 메서드는 지정된 메타파일의 각 레코드에 대해 애플리케이션 정의 콜백 함수를 호출합니다. 이 메서드를 사용하여 콜백 함수에서 PlayRecord를 호출하여 메타파일을 표시할 수 있습니다. (오버로드 10/12)

EnumerateMetafile Graphics::EnumerateMetafile 메서드는 지정된 메타파일의 각 레코드에 대해 애플리케이션 정의 콜백 함수를 호출합니다. 이 메서드를 사용하여 콜백 함수에서 PlayRecord를 호출하여 메타파일을 표시할 수 있습니다. (오버로드 11/12)
EnumerateMetafile Graphics::EnumerateMetafile 메서드는 지정된 메타파일의 각 레코드에 대해 애플리케이션 정의 콜백 함수를 호출합니다. 이 메서드를 사용하여 콜백 함수에서 PlayRecord를 호출하여 메타파일을 표시할 수 있습니다. (오버로드 1/12)
EnumerateMetafile Graphics::EnumerateMetafile 메서드는 지정된 메타파일의 각 레코드에 대해 애플리케이션 정의 콜백 함수를 호출합니다. 이 메서드를 사용하여 콜백 함수에서 PlayRecord를 호출하여 메타파일을 표시할 수 있습니다. (오버로드 3/12)
EnumerateMetafile Graphics::EnumerateMetafile 메서드는 지정된 메타파일의 각 레코드에 대해 애플리케이션 정의 콜백 함수를 호출합니다. 이 메서드를 사용하여 콜백 함수에서 PlayRecord를 호출하여 메타파일을 표시할 수 있습니다. (오버로드 12/12)
EnumerateMetafile Graphics::EnumerateMetafile 메서드는 지정된 메타파일의 각 레코드에 대해 애플리케이션 정의 콜백 함수를 호출합니다. 이 메서드를 사용하여 콜백 함수에서 PlayRecord를 호출하여 메타파일을 표시할 수 있습니다. (오버로드 9/12)
EnumerateMetafile Graphics::EnumerateMetafile 메서드는 지정된 메타파일의 각 레코드에 대해 애플리케이션 정의 콜백 함수를 호출합니다. 이 메서드를 사용하여 콜백 함수에서 PlayRecord를 호출하여 메타파일을 표시할 수 있습니다. (오버로드 5/12)
EnumerateMetafile Graphics::EnumerateMetafile 메서드는 지정된 메타파일의 각 레코드에 대해 애플리케이션 정의 콜백 함수를 호출합니다. 이 메서드를 사용하여 콜백 함수에서 PlayRecord를 호출하여 메타파일을 표시할 수 있습니다. (오버로드 8/12)
EnumerateMetafile Graphics::EnumerateMetafile 메서드는 지정된 메타파일의 각 레코드에 대해 애플리케이션 정의 콜백 함수를 호출합니다. 이 메서드를 사용하여 콜백 함수에서 PlayRecord를 호출하여 메타파일을 표시할 수 있습니다. (오버로드 4/12)

EnumerateMetafile Graphics::EnumerateMetafile 메서드는 지정된 메타파일의 각 레코드에 대해 애플리케이션 정의 콜백 함수를 호출합니다. 이 메서드를 사용하여 콜백 함수에서 PlayRecord를 호출하여 메타파일을 표시할 수 있습니다. (오버로드 2/12)
EnumerateMetafileProc *EnumerateMetafileProc*는 **Graphics::EnumerateMetafile** 메서드(및 오버로드)에 대해 애플리케이션에서 구현하는 콜백 함수의 서명입니다.
같음 Region::Equals 메서드는 이 지역이 지정된 지역과 같은지 여부를 결정합니다.
같음 Point::Equals 메서드는 두 Point 개체가 같은지 여부를 결정합니다. 두 지점은 동일한 X 및 Y 데이터 멤버가 있는 경우 같은 것으로 간주됩니다.
같음 PointF::Equals 메서드는 두 PointF 개체가 같은지 여부를 결정합니다. 두 지점은 동일한 X 및 Y 데이터 멤버가 있는 경우 같은 것으로 간주됩니다.
같음 Rect::Equals 메서드는 두 사각형이 같은지 여부를 결정합니다.
같음 RectF::Equals 메서드는 두 사각형이 같은지 여부를 결정합니다.
같음 Size::Equals 메서드는 두 Size 개체가 같은지 여부를 결정합니다.
같음 SizeF::Equals 메서드는 두 SizeF 개체가 같은지 여부를 결정합니다.
같음 Matrix::Equals 메서드는 이 행렬의 요소가 다른 행렬의 요소와 같은지 여부를 결정합니다.
제외 Region::Exclude 메서드는 지정된 경로의 내부와 교차하지 않는 자체 부분으로 이 지역을 업데이트합니다.

제외 Region::Exclude 메서드는 지정된 사각형의 내부와 교차하지 않는 지역을 업데이트합니다.
제외 Region::Exclude 메서드는 지정된 사각형의 내부와 교차하지 않는 자체 부분으로 이 지역을 업데이트합니다.
제외 Region::Exclude 메서드는 이 지역을 다른 지역과 교차하지 않는 자체 부분으로 업데이트합니다.
ExcludeClip Graphics::ExcludeClip 메서드는 지정된 사각형과 교차하지 않는 자체 부분으로 클리핑 영역을 업데이트합니다. (오버로드 2/2)
ExcludeClip Graphics::ExcludeClip 메서드는 지정된 사각형과 교차하지 않는 자체 부분으로 클리핑 영역을 업데이트합니다. (오버로드 1/2)
ExcludeClip Graphics::ExcludeClip 메서드는 클리핑 영역을 지정된 영역과 겹치지 않는 부분으로 업데이트합니다.
FillClosedCurve Graphics::FillClosedCurve 메서드는 점 배열에서 닫힌 카디널 스플라인을 만들고 브러시를 사용하여 스플라인의 내부를 채웁니다. (오버로드 2/2)
FillClosedCurve Graphics::FillClosedCurve 메서드는 점 배열에서 닫힌 카디널 스플라인을 만들고 지정된 모드인 스플라인의 내부에 따라 브러시를 사용하여 채웁니다. (오버로드 2/2)
FillClosedCurve Graphics::FillClosedCurve 메서드는 점 배열에서 닫힌 카디널 스플라인을 만들고 브러시를 사용하여 스플라인의 내부를 채웁니다. (오버로드 1/2)
FillClosedCurve Graphics::FillClosedCurve 메서드는 점 배열에서 닫힌 카디널 스플라인을 만들고 지정된 모드인 스플라인의 내부에 따라 브러시를 사용하여 채웁니다. (오버로드 1/2)

FillEllipse Graphics::FillEllipse 메서드는 브러시를 사용하여 사각형으로 지정된 타원의 내부를 채웁니다.
FillEllipse **Graphics::FillEllipse** 메서드는 브러시를 사용하여 사각형으로 지정된 타원의 내부를 채웁니다.
FillEllipse Graphics::FillEllipse 메서드는 브러시를 사용하여 좌표 및 차원으로 지정된 타원의 내부를 채웁니다. (오버로드 1/2)
FillEllipse Graphics::FillEllipse 메서드는 브러시를 사용하여 좌표 및 차원으로 지정된 타원의 내부를 채웁니다. (오버로드 2/2)
FillPath Graphics::FillPath 메서드는 브러시를 사용하여 경로의 내부를 채웁니다. 경로의 그림이 닫혀 있지 않으면 이 메서드는 닫혀 있지 않은 그림을 그림의 시작점과 끝점을 연결하는 직선으로 닫힌 것처럼 처리합니다.
FillPie Graphics::FillPie 메서드는 브러시를 사용하여 파이의 내부를 채웁니다. (오버로드 3/4)
FillPie Graphics::FillPie 메서드는 브러시를 사용하여 파이의 내부를 채웁니다. (오버로드 4/4)
FillPie Graphics::FillPie 메서드는 브러시를 사용하여 파이의 내부를 채웁니다. (오버로드 2/4)
FillPie Graphics::FillPie 메서드는 브러시를 사용하여 파이의 내부를 채웁니다. (오버로드 1/4)
FillPolygon Graphics::FillPolygon 메서드는 브러시를 사용하여 다각형의 내부를 채웁니다. (오버로드 4/4)
FillPolygon Graphics::FillPolygon 메서드는 브러시를 사용하여 다각형의 내부를 채웁니다. (오버로드 2/4)

FillPolygon Graphics::FillPolygon 메서드는 브러시를 사용하여 다각형의 내부를 채웁니다. (오버로드 3/4)
FillPolygon Graphics::FillPolygon 메서드는 브러시를 사용하여 다각형의 내부를 채웁니다. (오버로드 1/4)
FillRectangle Graphics::FillRectangle 메서드는 브러시를 사용하여 사각형의 내부를 채웁니다. (오버로드 2/4)
FillRectangle Graphics::FillRectangle 메서드는 브러시를 사용하여 사각형의 내부를 채웁니다. (오버로드 1/4)
FillRectangle Graphics::FillRectangle 메서드는 브러시를 사용하여 사각형의 내부를 채웁니다. (오버로드 3/4)
FillRectangle Graphics::FillRectangle 메서드는 브러시를 사용하여 사각형의 내부를 채웁니다. (오버로드 4/4)
FillRectangles Graphics::FillRectangles 메서드는 브러시를 사용하여 직사각형 시퀀스의 내부를 채웁니다. (오버로드 2/2)
FillRectangles Graphics::FillRectangles 메서드는 브러시를 사용하여 직사각형 시퀀스의 내부를 채웁니다. (오버로드 1/2)
FillRegion Graphics::FillRegion 메서드는 브러시를 사용하여 지정된 영역을 채웁니다.
FindFirstItem Image::FindFirstItem 메서드는 이 Image 개체의 첫 번째 메타데이터 항목에 대한 설명과 데이터 크기를 검색합니다.
FindNextItem Image::FindNextItem 메서드는 Image::FindFirstItem 메서드와 함께 이 Image 개체에 저장된 메타데이터 항목을 열거하는 데 사용됩니다.

평면화(flatten) GraphicsPath::Flatten 메서드는 이 경로에 변환을 적용하고 경로의 각 곡선을 연결된 선 시퀀스로 변환합니다.
플러시 Graphics::Flush 메서드는 보류 중인 모든 그래픽 작업을 플러시합니다.
글꼴 이 항목에서는 Font 클래스의 생성자를 나열합니다. 전체 클래스 목록은 Font 클래스를 참조하세요. (오버로드 1/2)
글꼴 FontFamily 개체, 크기, 글꼴 스타일 및 측정 단위를 기반으로 Font::Font 개체를 만듭니다.
글꼴 글꼴 패밀리, 크기, 글꼴 스타일, 측정 단위 및 FontCollection 개체를 기반으로 Font::Font 개체를 만듭니다.
글꼴 이 항목에서는 Font 클래스의 생성자를 나열합니다. 전체 클래스 목록은 Font 클래스를 참조하세요. (오버로드 2/2)
글꼴 현재 지정된 디바이스 컨텍스트로 선택된 Windows GDI(그래픽 디바이스 인터페이스) 글꼴 개체를 기반으로 Font::Font 개체를 만듭니다. 이 생성자는 GDI와의 호환성을 위해 제공됩니다.
글꼴 GDILOGFONT 구조체에 대한 핸들을 사용하여 GDI(Windows Graphics Device Interface) 논리 글꼴에서 간접적으로 Font::Font 개체를 만듭니다.
글꼴 GDI(Windows Graphics Device Interface) 논리 글꼴에서 직접 Font::Font 개체를 만듭니다. (오버로드 2/2)
글꼴 GDI(Windows Graphics Device Interface) 논리 글꼴에서 직접 Font::Font 개체를 만듭니다. (오버로드 1/2)

FontCollection 빈 FontCollection::FontCollection 개체를 만듭니다. (FontCollection.FontCollection)
FontCollection 빈 FontCollection::FontCollection 개체를 만듭니다. (오버로드 1/2)
FontFamily 빈 FontFamily::FontFamily 개체를 만듭니다.
FontFamily 이 항목에서는 FontFamily 클래스의 생성자를 나열합니다. 전체 클래스 목록은 FontFamilyClass를 참조하세요. (오버로드 2/2)
FontFamily 지정된 글꼴 패밀리를 기반으로 FontFamily::FontFamily 개체를 만듭니다.
FontFamily 이 항목에서는 FontFamily 클래스의 생성자를 나열합니다. 전체 클래스 목록은 FontFamilyClass를 참조하세요. (오버로드 1/2)
FromBITMAPINFO Bitmap::FromBITMAPINFO 메서드는 BITMAPINFO 구조체 및 픽셀 데이터 배열을 기반으로 Bitmap 개체를 만듭니다.
FromDirectDrawSurface7 Bitmap::FromDirectDrawSurface7 메서드는 DirectDraw 표면을 기반으로 비트맵 개체를 만듭니다. Bitmap 개체는 Bitmap 개체가 삭제될 때까지 DirectDraw 표면에 대한 참조를 유지합니다.
FromFile Bitmap::FromFile 메서드는 이미지 파일을 기반으로 Bitmap 개체를 만듭니다.
FromFile Image::FromFile 메서드는 파일을 기반으로 Image 개체를 만듭니다.
FromHBITMAP Bitmap::FromHBITMAP 메서드는 GDI(Windows Graphics Device Interface) 비트맵에 대한 핸들과 GDI 팔레트에 대한 핸들을 기반으로 Bitmap 개체를 만듭니다.

FromHDC Graphics::FromHDC 메서드는 지정된 디바이스 컨텍스트와 연결된 Graphics 개체를 만듭니다.
FromHDC Graphics::FromHDC 메서드는 지정된 디바이스 컨텍스트 및 지정된 디바이스와 연결된 Graphics 개체를 만듭니다.
FromHICON Bitmap::FromHICON 메서드는 아이콘에 대한 핸들을 기반으로 Bitmap 개체를 만듭니다.
FromHRGN Region::FromHRGN 메서드는 Windows GDI(그래픽 디바이스 인터페이스) 영역에서 Windows GDI+Region 개체를 만듭니다.
FromHWND Graphics::FromHWND 메서드는 지정된 창과 연결된 Graphicsobject를 만듭니다.
FromImage Graphics::FromImage 메서드는 지정된 Image 개체와 연결된 Graphicsobject를 만듭니다.
FromResource Bitmap::FromResource 메서드는 애플리케이션 또는 DLL 인스턴스 핸들과 비트맵 리소스의 이름을 기반으로 Bitmap 개체를 만듭니다.
FromStream Bitmap::FromStream 메서드는 스트림을 기반으로 Bitmap 개체를 만듭니다.
FromStream Image::FromStream 메서드는 스트림을 기반으로 새 Image 개체를 만듭니다.
GdiplusShutdown GdiplusShutdown 함수는 Windows GDI+에서 사용하는 리소스를 정리합니다. GdiplusStartup에 대한 각 호출은 GdiplusShutdown에 대한 호출과 쌍을 이루어야 합니다.
GdiplusStartup GdiplusStartup 함수는 Windows GDI+를 초기화합니다. 다른 GDI+ 호출을 하기 전에 GdiplusStartup을 호출하고 GDI+를 사용한 후 GdiplusShutdown을 호출합니다.

GdiplusStartupInput **GdiplusStartupInput** 구조체에 대한 생성자입니다.
GdiplusStartupInputEx **GdiplusStartupInputEx** 구조체에 대한 생성자입니다.
GenericDefault StringFormat::GenericDefault 메서드는 제네릭 기본 StringFormat 개체를 만듭니다.
GenericMonospace FontFamily::GenericMonospace 메서드는 제네릭 모노스페이스 서체를 지정하는 FontFamily 개체를 가져옵니다.
GenericSansSerif FontFamily::GenericSansSerif 메서드는 제네릭 sans serif 서체를 지정하는 FontFamily 개체를 가져옵니다.
GenericSerif FontFamily::GenericSerif 메서드는 제네릭 세리프 서체를 나타내는 FontFamily 개체를 가져옵니다.
GenericTypographic StringFormat::GenericTypographic 메서드는 제네릭 형식의 StringFormat 개체를 만듭니다.
GetA Color::GetA 메서드는 이 Color 개체의 알파 구성 요소를 가져옵니다.
GetAdjustedPalette ImageAttributes::GetAdjustedPalette 메서드는 지정된 범주의 조정 설정에 따라 색상표의 색을 조정합니다.
GetAlignment Pen::GetAlignment 메서드는 이 Pen 개체에 대해 현재 설정된 맞춤을 가져옵니다.
GetAlignment StringFormat::GetAlignment 메서드는 레이아웃 사각형의 원점과 관련하여 이 StringFormat 개체의 문자 맞춤을 나타내는 StringAlignment 열거형의 요소를 가져옵니다.

GetAllPropertyItems Image::GetAllPropertyItems 메서드는 이 Image 개체에 저장된 모든 속성 항목(메타데이터)을 가져옵니다.
GetAlpha Color::GetAlpha 메서드는 이 Color 개체의 알파 구성 요소를 가져옵니다.
GetAuxData Effect::GetAuxData는 Bitmap::ApplyEffect 메서드에 대한 이전 호출에서 만든 조회 테이블 집합에 대한 포인터를 가져옵니다.
GetAuxDataSize Effect::GetAuxDataSize 메서드는 Bitmap::ApplyEffect 메서드에 대한 이전 호출에서 만든 보조 데이터의 크기를 바이트 단위로 가져옵니다.
GetB Color::GetB 메서드는 이 Color 개체의 파란색 구성 요소를 가져옵니다.
GetBackgroundColor HatchBrush::GetBackgroundColor 메서드는 이 해치 브러시의 배경색을 가져옵니다.
GetBaseCap CustomLineCap::GetBaseCap 메서드는 기본 캡의 스타일을 가져옵니다. 기본 상한은 이 CustomLineCap 개체와 함께 선 끝의 캡으로 사용되는 LineCap 개체입니다.
GetBaseInset CustomLineCap::GetBaseInset 메서드는 기본 캡과 줄의 시작 사이의 거리를 가져옵니다.
GetBlend LinearGradientBrush::GetBlend 메서드는 LinearGradientBrush 개체에서 혼합 요소와 해당 혼합 위치를 가져옵니다.
GetBlend PathGradientBrush::GetBlend 메서드는 혼합 요소와 이 경로 그라데이션 브러시에 대해 현재 설정된 해당 혼합 위치를 가져옵니다.

GetBlendCount LinearGradientBrush::GetBlendCount 메서드는 이 LinearGradientBrush 개체에 대해 현재 설정된 혼합 요소 수를 가져옵니다.
GetBlendCount PathGradientBrush::GetBlendCount 메서드는 이 경로 그라데이션 브러시에 대해 현재 설정된 혼합 요소 수를 가져옵니다.
GetBlue Color::GetBlue 메서드는 이 Color 개체의 파란색 구성 요소를 가져옵니다.
GetBottom Rect::GetBottom 메서드는 사각형 아래쪽 가장자리의 y 좌표를 가져옵니다.
GetBottom RectF::GetBottom 메서드는 사각형 아래쪽 가장자리의 y 좌표를 가져옵니다.
GetBounds Image::GetBounds 메서드는 이 이미지에 대한 경계 사각형을 가져옵니다.
GetBounds Region::GetBounds 메서드는 이 지역을 둘러싸는 사각형을 가져옵니다. (오버로드 2/2)
GetBounds Region::GetBounds 메서드는 이 지역을 둘러싸는 사각형을 가져옵니다. (오버로드 1/2)
GetBounds Rect::GetBounds 메서드는 이 사각형의 복사본을 만듭니다.
GetBounds RectF::GetBounds 메서드는 이 사각형의 복사본을 만듭니다.
GetBounds GraphicsPath::GetBounds 메서드는 이 경로에 대한 경계 사각형을 가져옵니다. (오버로드 1/2)
GetBounds GraphicsPath::GetBounds 메서드는 이 경로에 대한 경계 사각형을 가져옵니다. (오버로드 2/2)

GetBounds MetafileHeader::GetBounds 메서드는 연결된 메타파일의 경계 사각형을 가져옵니다.
GetBrush Pen::GetBrush 메서드는 이 Pen 개체에 대해 현재 설정된 Brush 개체를 가져옵니다.
GetCellAscent FontFamily::GetCellAscent 메서드는 지정한 스타일 또는 스타일 조합에 대해 이 글꼴 패밀리의 셀 상승을 디자인 단위로 가져옵니다.
GetCellDescent FontFamily::GetCellDescent 메서드는 지정된 스타일 또는 스타일 조합에 대해 이 글꼴 패밀리의 셀 하강을 디자인 단위로 가져옵니다.
GetCenterColor PathGradientBrush::GetCenterColor 메서드는 이 경로 그라데이션 브러시의 중심점 색을 가져옵니다.
GetCenterPoint PathGradientBrush::GetCenterPoint 메서드는 이 경로 그라데이션 브러시의 중심점을 가져옵니다. (오버로드 1/2)
GetCenterPoint PathGradientBrush::GetCenterPoint 메서드는 이 경로 그라데이션 브러시의 중심점을 가져옵니다. (오버로드 2/2)
GetClip Graphics::GetClip 메서드는 이 Graphics 개체의 클리핑 영역을 가져옵니다.
GetClipBounds Graphics::GetClipBounds 메서드는 이 Graphics 개체의 클리핑 영역을 묶는 사각형을 가져옵니다. (오버로드 1/2)
GetClipBounds Graphics::GetClipBounds 메서드는 이 Graphics 개체의 클리핑 영역을 묶는 사각형을 가져옵니다. (오버로드 2/2)

GetColor SolidBrush::GetColor 메서드는 이 단색 브러시의 색을 가져옵니다.
GetColor Pen::GetColor 메서드는 이 Pen 개체에 대해 현재 설정된 색을 가져옵니다.
GetCompositingMode Graphics::GetCompositingMode 메서드는 이 Graphics 개체에 대해 현재 설정된 작성 모드를 가져옵니다.
GetCompositingQuality Graphics::GetCompositingQuality 메서드는 이 Graphics 개체에 대해 현재 설정된 작성 품질을 가져옵니다.
GetCompoundArray Pen::GetCompoundArray 메서드는 이 Pen 개체에 대해 현재 설정된 복합 배열을 가져옵니다.
GetCompoundArrayCount Pen::GetCompoundArrayCount 메서드는 복합 배열의 요소 수를 가져옵니다.
GetCount GraphicsPathIterator::GetCount 메서드는 경로의 데이터 요소 수를 반환합니다.
GetCustomEndCap Pen::GetCustomEndCap 메서드는 이 Pen 개체에 대해 현재 설정된 사용자 지정 끝 캡을 가져옵니다.
GetCustomStartCap Pen::GetCustomStartCap 메서드는 이 Pen 개체에 대해 현재 설정된 사용자 지정 시작 캡을 가져옵니다.
GetDashCap Pen::GetDashCap 메서드는 이 Pen 개체에 대해 현재 설정된 대시 캡 스타일을 가져옵니다.
GetDashOffset Pen::GetDashOffset 메서드는 줄의 시작부터 파선의 첫 번째 공백 시작까지의 거리를 가져옵니다.

GetDashPattern Pen::GetDashPattern 메서드는 이 Pen 개체에 대해 현재 설정된 사용자 지정 대시 및 공백 배열을 가져옵니다.
GetDashPatternCount Pen::GetDashPatternCount 메서드는 대시 패턴 배열의 요소 수를 가져옵니다.
GetDashStyle Pen::GetDashStyle 메서드는 이 Pen 개체에 대해 현재 설정된 대시 스타일을 가져옵니다.
GetData Region::GetData 메서드는 이 지역을 설명하는 데이터를 가져옵니다.
GetDataSize Region::GetDataSize 메서드는 이 지역을 설명하는 데이터 바이트 수를 가져옵니다.
GetDigitSubstitutionLanguage StringFormat::GetDigitSubstitutionLanguage 메서드는 서유럽 숫자로 대체될 숫자와 일치하는 언어를 가져옵니다.
GetDigitSubstitutionMethod StringFormat::GetDigitSubstitutionMethod 메서드는 이 StringFormat 개체에서 사용되는 숫자 대체 메서드를 나타내는 StringDigitSubstitute 열거형의 요소를 가져옵니다.
GetDownLevelRasterizationLimit 이 메타파일에 대해 현재 설정된 래스터화 제한을 가져옵니다.
GetDpiX Graphics::GetDpiX 메서드는 이 Graphics 개체와 연결된 디스플레이 디바이스의 가로 해상도를 인치당 점으로 가져옵니다.
GetDpiX MetafileHeader::GetDpiX 메서드는 연결된 메타파일의 인치당 가로 점을 가져옵니다.
GetDpiY Graphics::GetDpiY 메서드는 이 Graphics 개체와 연결된 디스플레이 디바이스의 세로 해상도를 인치당 점으로 가져옵니다.

GetDpiY MetafileHeader::GetDpiY 메서드는 연결된 메타파일의 인치당 세로 점을 가져옵니다.
GetElements Matrix::GetElements 메서드는 이 행렬의 요소를 가져옵니다. 요소는 m11, m12, m21, m22, m31, m32 순서로 배열에 배치됩니다. 여기서 mij는 행 i, 열 j의 요소를 나타냅니다.
GetEmfHeader MetafileHeader::GetEmfHeader 메서드는 연결된 메타파일의 속성을 포함하는 ENHMETAHEADER3 구조를 가져옵니다.
GetEmfPlusFlags MetafileHeader::GetEmfPlusFlags 메서드는 연결된 메타파일이 비디오 디스플레이 디바이스 컨텍스트에 대해 기록되었는지 여부를 나타내는 플래그를 가져옵니다.
GetEmHeight FontFamily::GetEmHeight 메서드는 이 글꼴 패밀리의 크기(일반적으로 em 크기 또는 em 높이라고 함)를 디자인 단위로 가져옵니다.
GetEncoderParameterList Image::GetEncoderParameterList 메서드는 지정된 이미지 인코더에서 지원하는 매개 변수 목록을 가져옵니다.
GetEncoderParameterListSize Image::GetEncoderParameterListSize 메서드는 지정된 이미지 인코더에 대한 매개 변수 목록의 크기를 바이트 단위로 가져옵니다.
GetEndCap Pen::GetEndCap 메서드는 이 Pen 개체에 대해 현재 설정된 끝 캡을 가져옵니다.
GetFamilies FontCollection::GetFamilies 메서드는 이 글꼴 컬렉션에 포함된 글꼴 패밀리를 가져옵니다.
GetFamily Font::GetFamily 메서드는 이 글꼴의 기반이 되는 글꼴 패밀리를 가져옵니다.
GetFamilyCount FontCollection::GetFamilyCount 메서드는 이 글꼴 컬렉션에 포함된 글꼴 패밀리 수를 가져옵니다.

GetFamilyName FontFamily::GetFamilyName 메서드는 이 글꼴 패밀리의 이름을 가져옵니다.
GetFillMode GraphicsPath::GetFillMode 메서드는 이 경로의 채우기 모드를 가져옵니다.
GetFlags Image::GetFlags 메서드는 이 Image 개체의 특정 특성을 나타내는 플래그 집합을 가져옵니다.
GetFocusScales PathGradientBrush::GetFocusScales 메서드는 이 경로 그라데이션 브러시의 포커스 배율을 가져옵니다.
GetForegroundColor HatchBrush::GetForegroundColor 메서드는 이 해치 브러시의 전경색을 가져옵니다.
GetFormatFlags StringFormat::GetFormatFlags 메서드는 이 StringFormat 개체에 대한 문자열 형식 플래그를 가져옵니다.
GetFrameCount Image::GetFrameCount 메서드는 이 Image 개체의 지정된 차원에 있는 프레임 수를 가져옵니다.
GetFrameDimensionsCount Image::GetFrameDimensionsCount 메서드는 이 Image 개체의 프레임 차원 수를 가져옵니다.
GetFrameDimensionsList Image::GetFrameDimensionsList 메서드는 이 Image 개체의 프레임 크기에 대한 식별자를 가져옵니다.
GetG(GetG) Color::GetG 메서드는 이 Color 개체의 녹색 구성 요소를 가져옵니다.
GetGammaCorrection LinearGradientBrush::GetGammaCorrection 메서드는 이 LinearGradientBrush 개체에 감마 수정을 사용할 수 있는지 여부를 결정합니다.

GetGammaCorrection PathGradientBrush::GetGammaCorrection 메서드는 이 경로 그라데이션 브러시에 감마 수정을 사용할 수 있는지 여부를 결정합니다.
GetGraphicsPath PathGradientBrush::GetGraphicsPath 메서드는 Windows GDI+ 버전 1.0에서 구현되지 않습니다.
GetGreen Color::GetGreen 메서드는 이 Color 개체의 녹색 구성 요소를 가져옵니다.
GetHalftonePalette Graphics::GetHalftonePalette 메서드는 Windows 하프톤 팔레트를 가져옵니다.
GetHatchStyle HatchBrush::GetHatchStyle 메서드는 이 해치 브러시의 해치 스타일을 가져옵니다.
GetHBITMAP Bitmap::GetHBITMAP 메서드는 이 Bitmap 개체에서 Windows GDI(그래픽 디바이스 인터페이스) 비트맵을 만듭니다.
GetHDC Graphics::GetHDC 메서드는 이 Graphics 개체와 연결된 디바이스 컨텍스트에 대한 핸들을 가져옵니다.
GetHeight AdjustableArrowCap::GetHeight 메서드는 화살표 캡의 높이를 가져옵니다. 높이는 화살표의 기준에서 꼭짓점까지의 거리입니다.
GetHeight Font::GetHeight 메서드는 지정된 Graphics 개체의 현재 단위에서 이 글꼴의 줄 간격을 가져옵니다.
GetHeight Font::GetHeight 메서드는 이 글꼴의 줄 간격을 픽셀 단위로 가져옵니다.
GetHeight Image::GetHeight 메서드는 이 이미지의 이미지 높이를 픽셀 단위로 가져옵니다.

GetHENHMETAFILE Metafile::GetHENHMETAFILE 메서드는 EMF(Enhanced Metafile) 파일에 대한 Windows 핸들을 가져옵니다.
GetHICON Bitmap::GetHICON 메서드는 이 Bitmap 개체에서 아이콘을 만듭니다.
GetHistogram Bitmap::GetHistogram 메서드는 이 Bitmap 개체의 지정된 색 채널에 대해 하나 이상의 히스토그램을 반환합니다.
GetHistogramSize Bitmap::GetHistogramSize는 Bitmap 개체의 Bitmap::GetHistogram 메서드를 호출하기 전에 할당해야 하는 요소 수(UINT 배열)를 반환합니다.
GetHorizontalResolution Image::GetHorizontalResolution 메서드는 이 이미지의 가로 해상도를 인치당 점으로 가져옵니다.
GetHotkeyPrefix StringFormat::GetHotkeyPrefix 메서드는 핫 키 접두사인 앰퍼샌드(&)가 발생할 때 문자열에서 수행되는 처리 유형을 나타내는 HotkeyPrefix 열거형의 요소를 가져옵니다.
GetHRGN Region::GetHRGN 메서드는 이 지역에서 Windows GDI(그래픽 디바이스 인터페이스) 영역을 만듭니다.
GetImage TextureBrush::GetImage 메서드는 이 텍스처 브러시로 정의된 Image 개체에 대한 포인터를 가져옵니다.
GetImageDecoders GetImageDecoders 함수는 사용 가능한 이미지 디코더에 대한 정보를 포함하는 ImageCodecInfo 개체의 배열을 가져옵니다.
GetImageDecodersSize GetImageDecodersSize 함수는 사용 가능한 이미지 디코더 수와 GetImageDecoders 함수에서 반환되는 ImageCodecInfo 개체 배열의 총 크기를 가져옵니다.

GetImageEncoders GetImageEncoders 함수는 사용 가능한 이미지 인코더에 대한 정보를 포함하는 ImageCodecInfo 개체의 배열을 가져옵니다.
GetImageEncodersSize GetImageEncodersSize 함수는 사용 가능한 이미지 인코더 수와 GetImageEncoders 함수에서 반환되는 ImageCodecInfo 개체 배열의 총 크기를 가져옵니다.
GetInterpolationColorCount LinearGradientBrush::GetInterpolationColorCount 메서드는 현재 이 선형 그라데이션 브러시에 대해 보간되도록 설정된 색 수를 가져옵니다.
GetInterpolationColorCount PathGradientBrush::GetInterpolationColorCount 메서드는 이 경로 그라데이션 브러시에 대해 현재 지정된 미리 설정된 색 수를 가져옵니다.
GetInterpolationColors LinearGradientBrush::GetInterpolationColors 메서드는 현재 이 선형 그라데이션 브러시 및 해당 혼합 위치에 대해 보간되도록 설정된 색을 가져옵니다.
GetInterpolationColors PathGradientBrush::GetInterpolationColors 메서드는 이 경로 그라데이션 브러시에 대해 현재 지정된 미리 설정된 색 및 혼합 위치를 가져옵니다.
GetInterpolationMode Graphics::GetInterpolationMode 메서드는 이 Graphics 개체에 대해 현재 설정된 보간 모드를 가져옵니다. 보간 모드는 이미지의 크기를 조정하거나 회전할 때 사용되는 알고리즘을 결정합니다.
GetItemData Image::GetItemData 메서드는 이 Image 개체에서 메타데이터 한 조각을 가져옵니다.
GetLastPoint GraphicsPath::GetLastPoint 메서드는 이 경로에 있는 마지막 그림의 끝점을 가져옵니다.
GetLastStatus CachedBitmap::GetLastStatus 메서드는 이 CachedBitmap 개체가 성공적으로 생성되었는지 여부를 나타내는 값을 반환합니다.

GetLastStatus CustomLineCap::GetLastStatus 메서드는 이 CustomLineCap 개체의 가장 최근 메서드 실패의 특성을 나타내는 값을 반환합니다.
GetLastStatus Font::GetLastStatus 메서드는 이 Font 개체의 가장 최근 메서드 실패의 특성을 나타내는 값을 반환합니다.
GetLastStatus FontCollection::GetLastStatus 메서드는 이 FontCollection 개체의 이전 메서드 호출 결과를 나타내는 값을 반환합니다.
GetLastStatus FontFamily::GetLastStatus 메서드는 이 FontFamily 개체의 가장 최근 메서드 실패의 특성을 나타내는 값을 반환합니다.
GetLastStatus Image::GetLastStatus 메서드는 이 Image 개체의 가장 최근 메서드 실패의 특성을 나타내는 값을 반환합니다.
GetLastStatus Region::GetLastStatus 메서드는 이 Regionobject의 가장 최근 메서드 실패의 특성을 나타내는 값을 반환합니다.
GetLastStatus Brush::GetLastStatus 메서드는 이 Brush 개체의 가장 최근 메서드 실패의 특성을 나타내는 값을 반환합니다.
GetLastStatus Graphics::GetLastStatus 메서드는 이 Graphics 개체의 가장 최근 메서드 오류의 특성을 나타내는 값을 반환합니다.
GetLastStatus GraphicsPath::GetLastStatus 메서드는 이 GraphicsPath 개체의 가장 최근 메서드 오류의 특성을 나타내는 값을 반환합니다.
GetLastStatus GraphicsPathIterator::GetLastStatus 메서드는 이 GraphicsPathIterator 개체의 가장 최근 메서드 오류의 특성을 나타내는 값을 반환합니다.

GetLastStatus ImageAttributes::GetLastStatus 메서드는 이 ImageAttributes 개체의 가장 최근 메서드 실패의 특성을 나타내는 값을 반환합니다.
GetLastStatus Matrix::GetLastStatus 메서드는 이 Matrix 개체의 가장 최근 메서드 실패의 특성을 나타내는 값을 반환합니다.
GetLastStatus Pen::GetLastStatus 메서드는 이 Pen 개체의 가장 최근 메서드 실패의 특성을 나타내는 값을 반환합니다.
GetLastStatus StringFormat::GetLastStatus 메서드는 이 StringFormat 개체의 가장 최근 메서드 실패의 특성을 나타내는 값을 반환합니다.
GetLeft Rect::GetLeft 메서드는 사각형 왼쪽 가장자리의 x 좌표를 가져옵니다.
GetLeft RectF::GetLeft 메서드는 사각형 왼쪽 가장자리의 x 좌표를 가져옵니다.
GetLineAlignment StringFormat::GetLineAlignment 메서드는 레이아웃 사각형의 원점과 관련하여 이 StringFormat 개체의 선 맞춤을 나타내는 StringAlignment 열거형의 요소를 가져옵니다.
GetLinearColors LinearGradientBrush::GetLinearColors 메서드는 이 선형 그라데이션 브러시의 시작 색과 끝 색을 가져옵니다.
GetLineJoin Pen::GetLineJoin 메서드는 이 Pen 개체에 대해 현재 설정된 선 조인 스타일을 가져옵니다.
GetLineSpacing FontFamily::GetLineSpacing 메서드는 지정한 스타일 또는 스타일 조합에 대해 이 글꼴 패밀리의 줄 간격을 디자인 단위로 가져옵니다. 줄 간격은 텍스트에 있는 연속된 두 줄 사이의 세로 거리입니다.

GetLocation Rect::GetLocation 메서드는 사각형의 왼쪽 위 모서리 좌표를 가져옵니다.
GetLocation RectF::GetLocation 메서드는 이 사각형의 왼쪽 위 모서리 좌표를 가져옵니다.
GetLogFontA Font::GetLogFontA 메서드는 LOGFONTA 구조를 사용하여 이 Font 개체의 특성을 가져옵니다.
GetLogFontW Font::GetLogFontW 메서드는 LOGFONTW 구조를 사용하여 이 Font 개체의 특성을 가져옵니다.
GetMeasurableCharacterRangeCount StringFormat::GetMeasurableCharacterRangeCount 메서드는 현재 설정된 측정 가능한 문자 범위 수를 가져옵니다. 설정된 문자 범위는 MeasureCharacterRanges 메서드를 사용하여 문자열에서 측정할 수 있습니다.
GetMetafileHeader Metafile::GetMetafileHeader 메서드는 헤더를 가져옵니다. (오버로드 2/4)
GetMetafileHeader Metafile::GetMetafileHeader 메서드는 헤더를 가져옵니다. (오버로드 3/4)
GetMetafileHeader Metafile::GetMetafileHeader 메서드는 이 메타파일의 메타파일 헤더를 가져옵니다.
GetMetafileHeader Metafile::GetMetafileHeader 메서드는 헤더를 가져옵니다. (오버로드 1/4)
GetMetafileHeader Metafile::GetMetafileHeader 메서드는 헤더를 가져옵니다. (오버로드 4/4)
GetMetafileSize MetafileHeader::GetMetafileSize 메서드는 메타파일의 크기를 바이트 단위로 가져옵니다.

GetMiddleInset AdjustableArrowCap::GetMiddleInset 메서드는 inset의 값을 가져옵니다. 중간 삽입은 기본의 중간 점이 꼭짓점을 향해 이동하는 단위 수입니다.
GetMiterLimit Pen::GetMiterLimit 메서드는 이 Pen 개체에 대해 현재 설정된 마이터 길이를 가져옵니다.
GetNearestColor Graphics::GetNearestColor 메서드는 전달된 색에 가장 가까운 색을 가져옵니다. 이 메서드는 8비트 색상표가 있는 픽셀당 8비트 또는 더 낮은 디스플레이 디바이스에서 작동합니다.
GetPageScale Graphics::GetPageScale 메서드는 이 Graphics 개체의 페이지 변환에 대해 현재 설정된 배율 인수를 가져옵니다. 페이지 변환은 페이지 좌표를 디바이스 좌표로 변환합니다.
GetPageUnit Graphics::GetPageUnit 메서드는 이 Graphics 개체에 대해 현재 설정된 측정 단위를 가져옵니다.
GetPalette Image::GetPalette 메서드는 이 Image 개체의 ColorPalette를 가져옵니다.
GetPaletteSize Image::GetPaletteSize 메서드는 이 Image 개체의 색상표 크기를 바이트 단위로 가져옵니다.
GetParameters Blur::GetParameters 메서드는 이 Blur 개체의 매개 변수의 현재 값을 가져옵니다.
GetParameters BrightnessContrast::GetParameters 메서드는 이 BrightnessContrast 개체의 매개 변수의 현재 값을 가져옵니다.
GetParameters ColorBalance::GetParameters 메서드는 이 ColorBalance 개체의 매개 변수의 현재 값을 가져옵니다.
GetParameters ColorCurve::GetParameters는 이 ColorCurve 개체의 매개 변수의 현재 값을 가져옵니다.

GetParameters ColorLUT::GetParameters 메서드는 이 ColorLUT 개체의 매개 변수의 현재 값을 가져옵니다.
GetParameters ColorMatrixEffect::GetParameters 메서드는 이 ColorMatrixEffect 개체의 현재 5x5 색 행렬 요소를 가져옵니다.
GetParameters HueSaturationLightness::GetParameters 메서드는 이 HueSaturationLightness 개체의 매개 변수의 현재 값을 가져옵니다.
GetParameters Levels::GetParameters 메서드는 이 Levels 개체의 매개 변수의 현재 값을 가져옵니다.
GetParameters RedEyeCorrection::GetParameters 메서드는 이 RedEyeCorrection 개체의 매개 변수의 현재 값을 가져옵니다.
GetParameters Sharpen::GetParameters 메서드는 이 Sharpen 개체의 매개 변수의 현재 값을 가져옵니다.
GetParameters Tint::GetParameters 메서드는 이 Tint 개체의 매개 변수의 현재 값을 가져옵니다.
GetParameterSize Effect::GetParameterSize 메서드는 이 효과에 대해 현재 설정된 매개 변수의 총 크기(바이트)를 가져옵니다. Effect::GetParameterSize 메서드는 일반적으로 Effect 클래스의 하위 항목 인스턴스인 개체에서 호출됩니다.
GetPathData GraphicsPath::GetPathData 메서드는 이 경로에서 점 배열과 점 형식의 배열을 가져옵니다. 이 두 배열은 함께 이 경로의 선, 곡선, 그림 및 표식을 정의합니다.
GetPathPoints GraphicsPath::GetPathPoints 메서드는 이 경로의 점 배열을 가져옵니다. 배열에는 경로를 그리는 데 사용되는 선 및 B zier 스플라인의 엔드포인트와 제어점이 포함됩니다.

GetPathPoints GraphicsPath::GetPathPoints 메서드는 이 경로의 점 배열을 가져옵니다.
GetPathTypes GraphicsPath::GetPathTypes 메서드는 이 경로의 점 형식 배열을 가져옵니다.
GetPenType Pen::GetPenType 메서드는 이 Pen 개체에 대해 현재 설정된 형식을 가져옵니다.
GetPhysicalDimension Image::GetPhysicalDimension 메서드는 이 이미지의 너비와 높이를 가져옵니다.
GetPixel Bitmap::GetPixel 메서드는 이 비트맵에서 지정된 픽셀의 색을 가져옵니다.
GetPixelFormat Image::GetPixelFormat 메서드는 이 Image 개체의 픽셀 형식을 가져옵니다.
GetPixelFormatSize GetPixelFormatSize 메서드는 지정된 픽셀 형식에서 사용하는 픽셀당 비트 수를 반환합니다.
GetPixelOffsetMode Graphics::GetPixelOffsetMode 메서드는 이 Graphics 개체에 대해 현재 설정된 픽셀 오프셋 모드를 가져옵니다.
GetPointCount GraphicsPath::GetPointCount 메서드는 이 경로의 데이터 요소 배열에 있는 점 수를 가져옵니다. 이는 경로의 점 형식 배열에 있는 형식 수와 동일합니다.
GetPointCount PathGradientBrush::GetPointCount 메서드는 이 브러시의 경계 경로를 정의하는 점 배열의 점 수를 가져옵니다.
GetPropertyCount Image::GetPropertyCount 메서드는 이 Image 개체에 저장된 속성(메타데이터 조각)의 수를 가져옵니다.

GetPropertyIdList Image::GetPropertyIdList 메서드는 이 Image 개체의 메타데이터에 사용되는 속성 식별자 목록을 가져옵니다.
GetPropertyItem Image::GetPropertyItem 메서드는 이 Image 개체에서 지정된 속성 항목(메타데이터 조각)을 가져옵니다.
GetPropertyItemSize Image::GetPropertyItemSize 메서드는 이 Image 개체의 지정된 속성 항목의 크기(바이트)를 가져옵니다.
GetPropertySize Image::GetPropertySize 메서드는 이 Image 개체에 저장된 모든 속성 항목의 총 크기(바이트)를 가져옵니다. Image::GetPropertySize 메서드는 이 Image 개체에 저장된 속성 항목의 수도 가져옵니다.
GetR Color::GetR 메서드는 이 Color 개체의 빨간색 구성 요소를 가져옵니다.
GetRawFormat Image::GetRawFormat 메서드는 이 Image 개체의 형식을 식별하는 전역적으로 고유한 식별자(GUID)를 가져옵니다. 다양한 파일 형식을 식별하는 GUID는 Gdiplusimaging.h로 정의됩니다.
GetRectangle LinearGradientBrush::GetRectangle 메서드는 그라데이션의 경계를 정의하는 사각형을 가져옵니다.
GetRectangle 이 항목에서는 LinearGradientBrush 클래스의 GetRectangle 메서드를 나열합니다. LinearGradientBrush 클래스에 대한 메서드의 전체 목록은 LinearGradientBrush 메서드를 참조하세요.
GetRectangle PathGradientBrush::GetRectangle 메서드는 이 경로 그라데이션 브러시의 경계 경로를 묶는 가장 작은 사각형을 가져옵니다. (오버로드 1/2)
GetRectangle PathGradientBrush::GetRectangle 메서드는 이 경로 그라데이션 브러시의 경계 경로를 묶는 가장 작은 사각형을 가져옵니다. (오버로드 2/2)

GetRed Color::GetRed 메서드는 이 Color 개체의 빨간색 구성 요소를 가져옵니다.
GetRegionScans Region::GetRegionScans 메서드는 이 지역과 근사치를 나타내는 사각형 배열을 가져옵니다. 영역은 사각형이 계산되기 전에 지정된 행렬에 의해 변환됩니다.
GetRegionScans Region::GetRegionScans 메서드는 이 지역과 근사치를 나타내는 사각형 배열을 가져옵니다.
GetRegionScansCount Region::GetRegionScansCount 메서드는 이 지역과 근사치를 나타내는 사각형 수를 가져옵니다. 영역은 사각형이 계산되기 전에 지정된 행렬에 의해 변환됩니다.
GetRenderingOrigin Graphics::GetRenderingOrigin 메서드는 이 Graphics 개체에 대해 현재 설정된 렌더링 원본을 가져옵니다.
Getright Rect::GetRight 메서드는 사각형 오른쪽 가장자리의 x 좌표를 가져옵니다.
Getright RectF::GetRight 메서드는 사각형 오른쪽 가장자리의 x 좌표를 가져옵니다.
GetSize Font::GetSize 메서드는 이 Font 개체의 글꼴 크기(일반적으로 em 크기라고 함)를 반환합니다. 크기는 이 Font 개체의 단위에 있습니다.
GetSize Rect::GetSize 메서드는 사각형의 너비와 높이를 가져옵니다.
GetSize RectF::GetSize 메서드는 이 사각형의 너비와 높이를 가져옵니다.
GetSmoothingMode Graphics::GetSmoothingMode 메서드는 평활화(앤티앨리어싱)가 Graphics 개체에 적용되는지 여부를 결정합니다.

GetStartCap Pen::GetStartCap 메서드는 이 Pen 개체에 대해 현재 설정된 시작 캡을 가져옵니다.
GetStrokeCaps CustomLineCap::GetStrokeCaps 메서드는 시작 선 캡과 끝 선 캡 모두에 대한 끝 캡 스타일을 가져옵니다. 선 대문자는 경로 내의 개별 선을 종료하는 LineCap 개체입니다.
GetStrokeJoin CustomLineCap::GetStrokeJoin 메서드는 동일한 GraphicsPath 개체에서 여러 줄을 조인하는 데 사용되는 LineJoin 스타일을 반환합니다.
GetStyle Font::GetStyle 메서드는 이 글꼴 서체의 스타일을 가져옵니다.
GetSubpathCount GraphicsPathIterator::GetSubpathCount 메서드는 경로의 하위 경로(그림이라고도 함)의 수를 반환합니다.
GetSurroundColorCount PathGradientBrush::GetSurroundColorCount 메서드는 이 경로 그라데이션 브러시의 경계 경로에 지정된 색 수를 가져옵니다.
GetSurroundColors PathGradientBrush::GetSurroundColors 메서드는 이 경로 그라데이션 브러시에 대해 현재 지정된 서라운드 색을 가져옵니다.
GetTabStopCount StringFormat::GetTabStopCount 메서드는 이 StringFormat 개체의 탭 중지 오프셋 수를 가져옵니다.
GetTabStops StringFormat::GetTabStops 메서드는 이 StringFormat 개체에서 탭 정지의 오프셋을 가져옵니다.
GetTextContrast Graphics::GetTextContrast 메서드는 이 Graphics 개체에 대해 현재 설정된 대비 값을 가져옵니다. 대비 값은 앤티앨리어싱 텍스트에 사용됩니다.

GetTextRenderingHint Graphics::GetTextRenderingHint 메서드는 이 Graphics 개체에 대해 현재 설정된 텍스트 렌더링 모드를 반환합니다.
GetThumbnailImage Image::GetThumbnailImage 메서드는 이 Image 개체에서 미리 보기 이미지를 가져옵니다.
GetTop Rect::GetTop 메서드는 사각형 위쪽 가장자리의 y 좌표를 가져옵니다.
GetTop RectF::GetTop 메서드는 사각형의 위쪽 가장자리의 y 좌표를 가져옵니다.
GetTransform LinearGradientBrush::GetTransform 메서드는 이 선형 그라데이션 브러시의 변환 매트릭스를 가져옵니다.
GetTransform TextureBrush::GetTransform 메서드는 이 텍스처 브러시의 변환 매트릭스를 가져옵니다.
GetTransform Graphics::GetTransform 메서드는 이 Graphics 개체의 월드 변환 매트릭스를 가져옵니다.
GetTransform PathGradientBrush::GetTransform 메서드는 이 경로 그라데이션 브러시의 변환 행렬을 가져옵니다.
GetTransform Pen::GetTransform 메서드는 이 Pen 개체에 대해 현재 설정된 월드 변환 매트릭스를 가져옵니다.
GetTrimming StringFormat::GetTrimming 메서드는 이 StringFormat 개체의 트리밍 스타일을 나타내는 StringTrimming 열거형의 요소를 가져옵니다.
GetType Image::GetType 메서드는 이 Image 개체의 형식(비트맵 또는 메타파일)을 가져옵니다.

GetType Brush::GetType 메서드는 이 브러시의 형식을 가져옵니다.
GetType MetafileHeader::GetType 메서드는 연결된 메타파일의 형식을 가져옵니다.
GetUnit Font::GetUnit 메서드는 이 Font 개체의 측정 단위를 반환합니다.
GetValue Color::GetValue 메서드는 이 Color 개체의 ARGB 값을 가져옵니다.
GetVersion MetafileHeader::GetVersion 메서드는 메타파일의 버전을 가져옵니다.
GetVerticalResolution Image::GetVerticalResolution 메서드는 이 이미지의 세로 해상도(인치당 점)를 가져옵니다.
GetVisibleClipBounds Graphics::GetVisibleClipBounds 메서드는 이 Graphics 개체의 표시되는 클리핑 영역을 묶는 사각형을 가져옵니다. (오버로드 1/2)
GetVisibleClipBounds Graphics::GetVisibleClipBounds 메서드는 이 Graphics 개체의 표시되는 클리핑 영역을 묶는 사각형을 가져옵니다. (오버로드 2/2)
GetWidth AdjustableArrowCap::GetWidth 메서드는 화살표 캡의 너비를 가져옵니다. 너비는 화살표 밑의 엔드포인트 사이의 거리입니다.
GetWidth Image::GetWidth 메서드는 이 이미지의 너비를 픽셀 단위로 가져옵니다.
GetWidth Pen::GetWidth 메서드는 이 Pen 개체에 대해 현재 설정된 너비를 가져옵니다.

GetWidthScale CustomLineCap::GetWidthScale 메서드는 배율 너비의 값을 가져옵니다. 이는 선을 그리는 데 사용되는 Pen 개체의 너비를 기준으로 사용자 지정 줄 한도의 크기를 조정하는 크기입니다. 기본값 1.0은 줄 한도의 크기를 조정하지 않습니다.
GetWmfHeader MetafileHeader::GetWmfHeader 메서드는 연결된 메타파일의 속성을 포함하는 METAHEADER 구조를 가져옵니다.
GetWrapMode LinearGradientBrush::GetWrapMode 메서드는 이 브러시의 래핑 모드를 가져옵니다. 래핑 모드는 브러시로 칠할 때 영역을 바꿈하는 방법을 결정합니다.
GetWrapMode TextureBrush::GetWrapMode 메서드는 이 텍스처 브러시에 대해 현재 설정된 래핑 모드를 가져옵니다.
GetWrapMode PathGradientBrush::GetWrapMode 메서드는 이 경로 그라데이션 브러시에 대해 현재 설정된 래핑 모드를 가져옵니다.
그래픽 이 항목에서는 Graphics 클래스의 생성자를 나열합니다. 전체 클래스 목록은 그래픽 클래스를 참조하세요. (오버로드 1/2)
그래픽 이 항목에서는 Graphics 클래스의 생성자를 나열합니다. 전체 클래스 목록은 그래픽 클래스를 참조하세요. (오버로드 2/2)
그래픽 지정된 디바이스 컨텍스트와 연결된 Graphics::Graphics 개체를 만듭니다.
그래픽 지정된 디바이스 컨텍스트 및 지정된 디바이스와 연결된 Graphics::Graphics 개체를 만듭니다.
그래픽 지정된 창과 연결된 Graphics::Graphics 개체를 만듭니다.

그래픽 Image 개체와 연결된 Graphics::Graphics 개체를 만듭니다.
Graphicspath 이 항목에서는 GraphicsPath 클래스의 생성자를 나열합니다. 전체 클래스 목록은 GraphicsPath 클래스를 참조하세요. (오버로드 2/2)
Graphicspath 점 배열, 형식 배열 및 채우기 모드를 기반으로 GraphicsPath::GraphicsPath 개체를 만듭니다. (오버로드 2/2)
Graphicspath 점 배열, 형식 배열 및 채우기 모드를 기반으로 GraphicsPath::GraphicsPath 개체를 만듭니다. (오버로드 1/2)
Graphicspath GraphicsPath::GraphicsPath 개체를 만들고 채우기 모드를 초기화합니다. 이것이 기본 생성자입니다.
Graphicspath 이 항목에서는 GraphicsPath 클래스의 생성자를 나열합니다. 전체 클래스 목록은 GraphicsPath 클래스를 참조하세요. (오버로드 1/2)
GraphicsPathIterator 새 GraphicsPathIterator::GraphicsPathIterator 개체를 만들고 GraphicsPath 개체와 연결합니다.
GraphicsPathIterator **GraphicsPathIterator**에 대한 복사 생성자입니다.
HasCurve GraphicsPathIterator::HasCurve 메서드는 경로에 곡선이 있는지 여부를 결정합니다.
HatchBrush **HatchBrush**에 대한 복사 생성자입니다.
HatchBrush 해치 스타일, 전경색 및 배경색을 기반으로 HatchBrush::HatchBrush 개체를 만듭니다.

HueSaturationLightness HueSaturationLightness 개체를 만듭니다.
이미지 이 항목에서는 Image 클래스의 생성자를 나열합니다. 전체 클래스 목록은 Image Class를 참조하세요. (오버로드 2/2)
이미지 파일을 기반으로 Image::Image 개체를 만듭니다.
이미지 이 항목에서는 Image 클래스의 생성자를 나열합니다. 전체 클래스 목록은 Image Class를 참조하세요. (오버로드 1/2)
이미지 스트림을 기반으로 Image::Image 개체를 만듭니다.
이미지 이 항목에서는 Image 클래스의 생성자를 나열합니다. 전체 클래스 목록은 Image Class를 참조하세요.
ImageAbort **ImageAbort*는 애플리케이션에서 구현하는 콜백 함수의 서명입니다. 썸네일 이미지를 만들거나 검색하거나 이미지를 그리는 과정에서 GDI+는 이 함수를 호출하여 프로세스를 중단할 수 있는 기회를 제공합니다.
ImageAttributes ImageAttributes::ImageAttributes 개체를 만듭니다. 이것이 기본 생성자입니다. (ImageAttributes.ImageAttributes)
ImageAttributes ImageAttributes::ImageAttributes 개체를 만듭니다. 이것이 기본 생성자입니다. (오버로드 2/3)
ImageAttributes ImageAttributes::ImageAttributes 개체를 만듭니다. 이것이 기본 생성자입니다. (오버로드 1/3)

팽창 Rect::Inflate 메서드는 사각형을 점 값으로 확장합니다. 왼쪽 및 오른쪽 가장자리의 X와 점 값입니다. 위쪽 및 아래쪽 가장자리에 있는 Y입니다.
팽창 이 항목에서는 Rect 클래스의 Inflate 메서드를 나열합니다. Rect 클래스에 대한 메서드의 전체 목록은 Rect 메서드를 참조하세요.
팽창 RectF::Inflate 메서드는 사각형을 점 값으로 확장합니다. 왼쪽 및 오른쪽 가장자리의 X와 점 값입니다. 위쪽 및 아래쪽 가장자리에 있는 Y입니다.
팽창 이 항목에서는 RectF 클래스의 Inflate 메서드를 나열합니다. RectF 클래스에 대한 메서드의 전체 목록은 RectF 메서드를 참조하세요.
InitializePalette Bitmap::InitializePalette 메서드는 표준, 최적 또는 사용자 지정 색상표를 초기화합니다.
InstalledFontCollection InstalledFontCollection::InstalledFontCollection 개체를 만듭니다. (InstalledFontCollection.InstalledFontCollection)
InstalledFontCollection InstalledFontCollection::InstalledFontCollection 개체를 만듭니다. (오버로드 1/2)
Intersect Region::Intersect 메서드는 지정된 경로의 내부와 교차하는 자체 부분으로 이 지역을 업데이트합니다.
Intersect Region::Intersect 메서드는 지정된 사각형의 내부와 교차하는 지역을 업데이트합니다.
Intersect Region::Intersect 메서드는 지정된 사각형의 내부와 교차하는 자체 부분으로 이 지역을 업데이트합니다.

Intersect Region::Intersect 메서드는 이 지역을 다른 지역과 교차하는 자체 부분으로 업데이트합니다.
Intersect 이 항목에서는 Rect 클래스의 Intersect 메서드를 나열합니다. Rect 클래스에 대한 메서드의 전체 목록은 Rect 메서드를 참조하세요.
Intersect Rect::Intersect 메서드는 두 사각형의 교차점을 확인하고 결과를 Rect 개체에 저장합니다.
Intersect 이 항목에서는 RectF 클래스의 Intersect 메서드를 나열합니다. RectF 클래스에 대한 메서드의 전체 목록은 RectF 메서드를 참조하세요.
Intersect RectF::Intersect 메서드는 두 사각형의 교차점을 결정하고 결과를 RectF 개체에 저장합니다.
IntersectClip Graphics::IntersectClip 메서드는 이 Graphics 개체의 클리핑 영역을 이 Graphics 개체의 현재 클리핑 영역과 교차하는 지정된 사각형 부분으로 업데이트합니다.
IntersectClip Graphics::IntersectClip 메서드는 이 Graphics 개체의 클리핑 영역을 업데이트합니다.
IntersectClip Graphics::IntersectClip 메서드는 이 Graphics 개체의 클리핑 영역을 이 Graphics 개체의 현재 클리핑 영역과 교차하는 지정된 영역의 부분으로 업데이트합니다.
IntersectsWith Rect::IntersectsWith 메서드는 이 사각형이 다른 사각형과 교차하는지 여부를 결정합니다.
IntersectsWith RectF::IntersectsWith 메서드는 이 사각형이 다른 사각형과 교차하는지 여부를 결정합니다.
반전 이 행렬을 반전할 수 없는 경우 Matrix::Invert 메서드는 이 행렬의 요소를 역의 요소로 바꿉니다.

IsAlphaPixelFormat IsAlphaPixelFormat 메서드는 지정된 픽셀 형식에 알파 구성 요소가 있는지 여부를 결정합니다.
IsAvailable Font::IsAvailable 메서드는 이 Font 개체가 성공적으로 만들어졌는지 여부를 결정합니다.
IsAvailable FontFamily::IsAvailable 메서드는 이 FontFamily 개체가 성공적으로 만들어졌는지 여부를 결정합니다.
IsCanonicalPixelFormat IsCanonicalPixelFormat 메서드는 지정된 픽셀 형식이 정식 형식(_PixelFormat32bppARGB 또는 PixelFormat64bppARGB 중 하나인지 여부를 결정합니다.
IsClipEmpty Graphics::IsClipEmpty 메서드는 이 Graphics 개체의 클리핑 영역이 비어 있는지 여부를 결정합니다.
IsDisplay MetafileHeader::IsDisplay 메서드는 연결된 메타파일이 비디오 표시 디바이스 컨텍스트에 대해 기록되었는지 여부를 결정합니다.
IsEmf MetafileHeader::IsEmf 메서드는 연결된 메타파일이 EMF 형식인지 여부를 결정합니다.
IsEmfOrEmfPlus MetafileHeader::IsEmfOrEmfPlus 메서드는 연결된 메타파일이 EMF 또는 EMF+ 형식인지 여부를 결정합니다.
IsEmfPlus MetafileHeader::IsEmfPlus 메서드는 연결된 메타파일이 EMF+ 형식인지 여부를 결정합니다.
IsEmfPlusDual MetafileHeader::IsEmfPlusDual 메서드는 연결된 메타파일이 EMF+ 이중 형식인지 여부를 결정합니다.
IsEmfPlusOnly MetafileHeader::IsEmfPlusOnly 메서드는 연결된 메타파일이 EMF+ Only 형식인지 여부를 결정합니다.

IsEmpty Region::IsEmpty 메서드는 이 지역이 비어 있는지 여부를 결정합니다.
IsEmptyArea Rect::IsEmptyArea 메서드는 이 사각형이 비어 있는지 여부를 결정합니다.
IsEmptyArea RectF::IsEmptyArea 메서드는 이 사각형이 비어 있는지 여부를 결정합니다.
IsExtendedPixelFormat IsExtendedPixelFormat 메서드는 지정된 픽셀 형식이 색 채널당 16비트를 사용하는지 여부를 결정합니다.
IsFilled AdjustableArrowCap::IsFilled 메서드는 화살표 캡이 채워지는지 여부를 결정합니다.
IsIdentity Matrix::IsIdentity 메서드는 이 행렬이 ID 행렬인지 여부를 결정합니다.
IsIndexedPixelFormat IsIndexedPixelFormat 메서드는 지정된 픽셀 형식이 인덱싱된 형식인지 여부를 결정합니다.
IsInfinite Region::IsInfinite 메서드는 이 지역이 무한인지 여부를 결정합니다.
IsInvertible Matrix::IsInvertible 메서드는 이 행렬을 반전할 수 없는지 여부를 결정합니다.
IsOutlineVisible GraphicsPath::IsOutlineVisible 메서드는 지정된 Graphicsobject 및 지정된 펜으로 경로를 그릴 때 지정된 점이 이 경로의 윤곽선에 닿는지 여부를 결정합니다.
IsOutlineVisible GraphicsPath::IsOutlineVisible 메서드는 지정된 점이 경로의 윤곽선에 닿는지 여부를 결정합니다.

IsOutlineVisible GraphicsPath::IsOutlineVisible 메서드는 지정된 Graphics 개체와 지정된 펜으로 경로를 그릴 때 지정된 점이 이 경로의 윤곽선에 닿는지 여부를 결정합니다. (오버로드 1/2)
IsOutlineVisible GraphicsPath::IsOutlineVisible 메서드는 지정된 Graphics 개체와 지정된 펜으로 경로를 그릴 때 지정된 점이 이 경로의 윤곽선에 닿는지 여부를 결정합니다. (오버로드 2/2)
IsStyleAvailable FontFamily::IsStyleAvailable 메서드는 지정된 스타일을 이 글꼴 패밀리에 사용할 수 있는지 여부를 결정합니다.
IsVisible Region::IsVisible 메서드는 지점이 이 지역 내에 있는지 여부를 결정합니다. (오버로드 1/4)
IsVisible Region::IsVisible 메서드는 지점이 이 지역 내에 있는지 여부를 결정합니다. (오버로드 4/4)
IsVisible Region::IsVisible 메서드는 사각형이 이 지역과 교차하는지 여부를 결정합니다. (오버로드 2/4)
IsVisible Region::IsVisible 메서드는 사각형이 이 지역과 교차하는지 여부를 결정합니다. (오버로드 4/4)
IsVisible Region::IsVisible 메서드는 지점이 이 지역 내에 있는지 여부를 결정합니다. (오버로드 2/4)
IsVisible Region::IsVisible 메서드는 사각형이 이 지역과 교차하는지 여부를 결정합니다. (오버로드 3/4)
IsVisible Region::IsVisible 메서드는 지점이 이 지역 내에 있는지 여부를 결정합니다. (오버로드 3/4)
IsVisible Region::IsVisible 메서드는 사각형이 이 지역과 교차하는지 여부를 결정합니다. (오버로드 1/4)

IsVisible Graphics::IsVisible 메서드는 지정한 점이 이 Graphics 개체의 표시되는 클리핑 영역 내에 있는지 여부를 결정합니다. (오버로드 2/4)
IsVisible Graphics::IsVisible 메서드는 지정한 점이 이 Graphics 개체의 표시되는 클리핑 영역 내에 있는지 여부를 결정합니다. (오버로드 1/4)
IsVisible Graphics::IsVisible 메서드는 지정된 사각형이 이 Graphics 개체의 표시되는 클리핑 영역과 교차하는지 여부를 결정합니다. (오버로드 3/4)
IsVisible Graphics::IsVisible 메서드는 지정된 사각형이 이 Graphics 개체의 표시되는 클리핑 영역과 교차하는지 여부를 결정합니다. (오버로드 2/4)
IsVisible Graphics::IsVisible 메서드는 지정한 점이 이 Graphics 개체의 표시되는 클리핑 영역 내에 있는지 여부를 결정합니다. (오버로드 3/4)
IsVisible Graphics::IsVisible 메서드는 지정된 사각형이 이 Graphics 개체의 표시되는 클리핑 영역과 교차하는지 여부를 결정합니다. (오버로드 4/4)
IsVisible Graphics::IsVisible 메서드는 지정한 점이 이 Graphics 개체의 표시되는 클리핑 영역 내에 있는지 여부를 결정합니다. (오버로드 4/4)
IsVisible Graphics::IsVisible 메서드는 지정된 사각형이 이 Graphics 개체의 표시되는 클리핑 영역과 교차하는지 여부를 결정합니다. (오버로드 1/4)
IsVisible GraphicsPath::IsVisible 메서드는 지정된 그래픽 개체로 이 경로를 채울 때 지정된 점이 채워진 영역에 있는지 여부를 결정합니다. (오버로드 1/2)
IsVisible GraphicsPath::IsVisible 메서드는 지정된 점이 영역에 있는지 여부를 결정합니다.

IsVisible GraphicsPath::IsVisible 메서드는 이 경로가 지정된 Graphicsobject로 채워질 때 지정된 점이 채워진 영역에 있는지 여부를 결정합니다.
IsVisible GraphicsPath::IsVisible 메서드는 이 경로가 지정된 Graphics 개체로 채워지면 지정된 점이 채워진 영역에 있는지 여부를 결정합니다. (오버로드 2/2)
IsVisibleClipEmpty Graphics::IsVisibleClipEmpty 메서드는 이 Graphics 개체의 표시되는 클리핑 영역이 비어 있는지 여부를 결정합니다. 표시되는 클리핑 영역은 이 Graphics 개체의 클리핑 영역과 창의 클리핑 영역의 교차점입니다.
IsWmf MetafileHeader::IsWmf 메서드는 연결된 메타파일이 WMF 형식인지 여부를 결정합니다.
IsWmfPlaceable MetafileHeader::IsWmfPlaceable 메서드는 연결된 메타파일이 배치 가능한 메타파일인지 여부를 결정합니다.
수준 Levels 개체를 만듭니다.
LinearGradientBrush 이 항목에서는 LinearGradientBrush 클래스의 생성자를 나열합니다. 전체 클래스 목록은 LinearGradientBrush 클래스를 참조하세요. (오버로드 2/2)
LinearGradientBrush 이 항목에서는 LinearGradientBrush 클래스의 생성자를 나열합니다. 전체 클래스 목록은 LinearGradientBrush 클래스를 참조하세요. (오버로드 1/2)
LinearGradientBrush 경계 지점 및 경계 색 집합에서 LinearGradientBrush::LinearGradientBrush 개체를 만듭니다.
LinearGradientBrush 직사각형 및 방향 모드를 기반으로 LinearGradientBrush::LinearGradientBrush 개체를 만듭니다. (오버로드 1/2)

LinearGradientBrush 직사각형 및 방향 각도에서 LinearGradientBrush::LinearGradientBrush 개체를 만듭니다. (오버로드 2/2)
LinearGradientBrush 직사각형 및 방향 모드를 기반으로 LinearGradientBrush::LinearGradientBrush 개체를 만듭니다. (오버로드 2/2)
LinearGradientBrush 직사각형 및 방향 각도에서 LinearGradientBrush::LinearGradientBrush 개체를 만듭니다. (오버로드 1/2)
LinearGradientBrush 이 항목에서는 LinearGradientBrush 클래스의 생성자를 나열합니다. 전체 클래스 목록은 LinearGradientBrush 클래스를 참조하세요.
Lockbits Bitmap::LockBits 메서드는 이 비트맵의 사각형 부분을 잠그고 지정된 형식으로 픽셀 데이터를 읽거나 쓰는 데 사용할 수 있는 임시 버퍼를 제공합니다.
MakeARGB Color::MakeARGB 메서드는 지정된 알파, 빨간색, 녹색 및 파란색 구성 요소를 통합하는 32비트 값을 만듭니다.
MakeEmpty Region::MakeEmpty 메서드는 이 지역을 빈 지역으로 업데이트합니다. 즉, 지역은 디스플레이 디바이스의 공간을 차지하지 않습니다.
MakeInfinite Region::MakeInfinite 메서드는 이 지역을 무한 지역으로 업데이트합니다.
행렬 ID 행렬을 나타내는 Matrix::Matrix 개체를 만들고 초기화합니다. (Matrix.Matrix)
행렬 ID 행렬을 나타내는 Matrix::Matrix 개체를 만들고 초기화합니다. (오버로드 2/3)

행렬 사각형과 점을 기반으로 Matrix::Matrix 개체를 만듭니다. (오버로드 2/2)
행렬 사각형과 점을 기반으로 Matrix::Matrix 개체를 만듭니다. (오버로드 1/2)
행렬 ID 행렬을 나타내는 Matrix::Matrix 개체를 만들고 초기화합니다. (오버로드 1/3)
행렬 아핀 변환을 정의하는 6개의 숫자를 기반으로 Matrix::Matrix 개체를 만들고 초기화합니다.
MeasureCharacterRanges Graphics::MeasureCharacterRanges 메서드는 문자열 내의 문자 위치 범위를 경계로 하는 각 영역 집합을 가져옵니다.
MeasureDriverString Graphics::MeasureDriverString 메서드는 지정된 문자와 해당 위치에 대한 경계 상자를 측정합니다.
MeasureString Graphics::MeasureString 메서드는 지정된 글꼴, 서식 및 레이아웃 사각형에서 문자열의 범위를 측정합니다. (오버로드 2/3)
MeasureString Graphics::MeasureString 메서드는 지정된 글꼴 및 레이아웃 사각형의 문자열 범위를 측정합니다. (오버로드 1/2)
MeasureString Graphics::MeasureString 메서드는 지정된 글꼴, 서식 및 레이아웃 사각형에서 문자열의 범위를 측정합니다. (오버로드 3/3)
MeasureString Graphics::MeasureString 메서드는 지정된 글꼴 및 레이아웃 사각형의 문자열 범위를 측정합니다. (오버로드 2/2)
MeasureString Graphics::MeasureString 메서드는 지정된 글꼴, 서식 및 레이아웃 사각형에서 문자열의 범위를 측정합니다. (오버로드 1/3)

Metafile 이 항목에서는 Metafile 클래스의 생성자를 나열합니다. 전체 클래스 목록은 메타파일 클래스를 참조하세요. (오버로드 2/2)
Metafile 재생을 위해 Metafile::Metafile 개체를 만듭니다.
Metafile 이 항목에서는 Metafile 클래스의 생성자를 나열합니다. 전체 클래스 목록은 메타파일 클래스를 참조하세요. (오버로드 1/2)
Metafile 기록할 Metafile::Metafile 개체를 만듭니다. (오버로드 2/6)
Metafile 기록할 Metafile::Metafile 개체를 만듭니다. (오버로드 1/6)
Metafile 기록할 Metafile::Metafile 개체를 만듭니다. (오버로드 5/6)
Metafile 기록할 Metafile::Metafile 개체를 만듭니다. (오버로드 4/6)
Metafile 기록할 Metafile::Metafile 개체를 만듭니다. (오버로드 3/6)
Metafile 기록할 Metafile::Metafile 개체를 만듭니다. (오버로드 6/6)
Metafile Windows GDI(GDI) EMF(Enhanced Metafile) 파일을 기반으로 재생을 위한 Windows GDI+ 메타파일 개체를 만듭니다.
Metafile 기록할 Windows GDI+Metafile::Metafile 개체를 만듭니다. 형식은 메타파일을 배치할 수 있습니다.

Metafile 재생을 위해 IStream 인터페이스에서 Metafile::Metafile 개체를 만듭니다.
Metafile IStream 인터페이스에 기록하기 위한 Metafile::Metafile 개체를 만듭니다. (오버로드 1/3)
Metafile IStream 인터페이스에 기록하기 위한 Metafile::Metafile 개체를 만듭니다. (오버로드 3/3)
Metafile IStream 인터페이스에 기록하기 위한 Metafile::Metafile 개체를 만듭니다. (오버로드 2/3)
Metafile 이 항목에서는 Metafile 클래스의 생성자를 나열합니다. 전체 클래스 목록은 메타파일 클래스를 참조하세요.
곱하기 Matrix::Multiply 메서드는 이 행렬을 자체 및 다른 행렬의 곱으로 업데이트합니다.
MultiplyTransform LinearGradientBrush::MultiplyTransform 메서드는 이 브러시의 변환 매트릭스를 자체 및 다른 행렬의 곱으로 업데이트합니다.
MultiplyTransform TextureBrush::MultiplyTransform 메서드는 이 브러시의 변환 매트릭스를 자체 및 다른 행렬의 곱으로 업데이트합니다.
MultiplyTransform Graphics::MultiplyTransform 메서드는 이 Graphics 개체의 월드 변환 매트릭스를 자체 및 다른 행렬의 곱으로 업데이트합니다.
MultiplyTransform PathGradientBrush::MultiplyTransform 메서드는 브러시의 변환 매트릭스를 자체 및 다른 행렬의 곱으로 업데이트합니다.
MultiplyTransform Pen::MultiplyTransform 메서드는 이 Pen 개체의 월드 변환 매트릭스를 자체 및 다른 행렬의 곱으로 업데이트합니다.

NextMarker GraphicsPathIterator::NextMarker 메서드는 이 반복기의 연결된 경로에서 표식으로 구분된 다음 섹션을 가져옵니다.
NextMarker GraphicsPathIterator::NextMarker 메서드는 섹션의 시작 인덱스와 끝 인덱스를 가져옵니다.
NextPathType GraphicsPathIterator::NextPathType 메서드는 모두 형식이 동일한 다음 데이터 요소 그룹의 시작 인덱스와 끝 인덱스를 가져옵니다.
NextSubpath GraphicsPathIterator::NextSubpath 메서드는 이 반복기의 연결된 경로에서 다음 그림(하위 경로)을 가져옵니다.
NextSubpath GraphicsPathIterator::NextSubpath 메서드는 다음 하위 경로의 시작 인덱스와 끝 인덱스를 가져옵니다.
NotificationHookProc **NotificationHookProc*는 **GdiplusStartup이 GdiplusStartupOutput** 구조에서 반환하는 알림 후크 함수의 서명입니다.
NotificationUnhookProc **NotificationUnhookProc*는 **GdiplusStartup이 GdiplusStartupOutput** 구조에서 반환하는 알림 unhook 함수의 서명입니다.
ObjectTypeIsValid ObjectTypeIsValid 함수는 ObjectType 열거형의 요소가 유효한 개체 형식을 나타내는지 여부를 결정합니다.
Offset 이 항목에서는 Rect 클래스의 Offset 메서드를 나열합니다. Rect 클래스에 대한 메서드의 전체 목록은 Rect 메서드를 참조하세요.
Offset Rect::Offset 메서드는 사각형을 가로로 dx씩, 세로로 dy씩 이동합니다.

Offset 이 항목에서는 RectF 클래스의 Offset 메서드를 나열합니다. RectF 클래스에 대한 메서드의 전체 목록은 Rect 메서드를 참조하세요.
Offset RectF::Offset 메서드는 사각형을 가로로, dx를 세로로 이동합니다.
Offsetx Matrix::OffsetX 메서드는 행 3, 열 1의 요소인 이 행렬의 가로 변환 값을 가져옵니다.
OffsetY Matrix::OffsetY 메서드는 행 3, 열 2의 요소인 이 행렬의 세로 변환 값을 가져옵니다.
연산자- Point::operator- 메서드는 두 Point 개체의 X 및 Y 데이터 멤버를 뺍니다.
연산자- PointF::operator- 메서드는 두 PointF 개체의 X 및 Y 데이터 멤버를 뺍니다.
연산자- Size::operator- 메서드는 두 Size 개체의 Width 및 Height 데이터 멤버를 뺍니다.
연산자- **SizeF** 클래스의 빼기 연산자입니다.
operator delete GdiplusBase::operator delete 메서드는 하나의 Windows GDI+ 개체에 대한 메모리 할당을 취소합니다.
operator delete[] xref rid="*gdiplus_CLASS_GdiplusBase_operator_delete_bracket_in_pVoid*" qualify="0"/> 메서드는 Windows GDI+ 개체 배열에 대한 메모리 할당을 취소합니다.
연산자 new GdiplusBase::operator new 메서드는 하나의 Windows GDI+ 개체에 대한 메모리를 할당합니다.

operator new[]

GdiplusBase::operator new[] 메서드는 Windows GDI+ 개체 배열에 대한 메모리를 할당합니다.

operator+

Point::operator+ 메서드는 두 Point 개체의 X 및 Y 데이터 멤버를 추가합니다.

operator+

PointF::operator+ 메서드는 두 PointF 개체의 X 및 Y 데이터 멤버를 추가합니다.

operator+

Size::operator+ 메서드는 두 Size 개체의 Width 및 Height 데이터 멤버를 추가합니다.

operator+

SizeF::operator+ 메서드는 두 SizeF 개체의 Width 및 Height 데이터 멤버를 추가합니다.

operator=

CharacterRange::operator= 메서드는 이 CharacterRange 개체를 지정된 CharacterRange 개체와 동일하게 설정합니다.

개요

GraphicsPath::Outline 메서드는 이 경로를 변환하고 평면화한 다음 경로의 윤곽선만 나타내도록 이 경로의 데이터 요소를 변환합니다.

PathData

PathData::P athData 개체를 만듭니다. Count 데이터 멤버는 0으로 초기화됩니다. Points 및 Types 데이터 멤버는 NULL로 초기화됩니다.

PathGradientBrush

GraphicsPath 개체를 기반으로 PathGradientBrush::P athGradientBrush 개체를 만듭니다.

PathGradientBrush

PathGradientBrush::P athGradientBrush(const PathGradientBrush &) 생성자(gdipluspath.h)에는 PathGradientBrush 클래스의 생성자가 나열됩니다.

PathGradientBrush

점 배열을 기반으로 PathGradientBrush::P athGradientBrush 개체를 만듭니다. 경로 그라데이션 브러시의 래핑 모드를 초기화합니다.

PathGradientBrush 점 배열을 기반으로 PathGradientBrush 개체를 만듭니다. 경로 그라데이션 브러시의 래핑 모드를 초기화합니다.
PathGradientBrush PathGradientBrush::P athGradientBrush~r5 생성자(gdipluspath.h)에는 PathGradientBrush 클래스의 생성자가 나열됩니다.
펜 브러시의 특성과 실제 숫자를 사용하여 이 Pen 개체의 너비를 설정하는 Pen 개체를 만듭니다.
펜 지정된 색과 너비를 사용하는 Pen 개체를 만듭니다.
펜 이 항목에서는 펜 클래스의 생성자를 나열합니다. 전체 클래스 목록은 펜 클래스를 참조하세요. (오버로드 1/2)
펜 이 항목에서는 펜 클래스의 생성자를 나열합니다. 전체 클래스 목록은 펜 클래스를 참조하세요. (오버로드 2/2)
PlayRecord Metafile::P layRecord 메서드는 메타파일 레코드를 재생합니다.
Point Point 개체를 만들고 X 및 Y 데이터 멤버를 0으로 초기화합니다. 이것이 기본 생성자입니다.
Point 새 Point 개체를 만들고 다른 Point 개체에서 데이터 멤버를 복사합니다.
Point Size 개체를 사용하여 Point 개체를 만들어 X 및 Y 데이터 멤버를 초기화합니다.
Point 두 정수로 X 및 Y 데이터 멤버를 초기화하는 Point 개체를 만듭니다.

Pointf PointF 개체를 만들고 X 및 Y 데이터 멤버를 0으로 초기화합니다. 이것이 기본 생성자입니다.
Pointf 새 PointF 개체를 만들고 다른 PointF 개체에서 데이터를 복사합니다.
Pointf SizeF 개체를 사용하여 PointF 개체를 만들어 X 및 Y 데이터 멤버를 지정합니다.
Pointf 두 개의 실제 숫자를 사용하여 X 및 Y 데이터 멤버를 지정하는 PointF 개체를 만듭니다.
PrivateFontCollection 빈 PrivateFontCollection 개체를 만듭니다. (PrivateFontCollection.PrivateFontCollection)
PrivateFontCollection 빈 PrivateFontCollection 개체를 만듭니다. (오버로드 1/2)
Rect x 좌표, y 좌표, 너비 및 높이가 모두 0인 Rect 개체를 만듭니다. 이것이 기본 생성자입니다.
Rect Point 개체를 사용하여 X 및 Y 데이터 멤버를 초기화하고 Size 개체를 사용하여 Width 및 Height 데이터 멤버를 초기화하여 Rect 개체를 만듭니다.
Rect 4개의 정수로 X, Y, Width 및 Height 데이터 멤버를 초기화하여 Rect 개체를 만듭니다.
RectF RectF 개체를 만들고 X, Y, Width 및 Height 데이터 멤버를 0으로 초기화합니다. 이것이 기본 생성자입니다. (RectF.RectF)
RectF RectF 개체를 만들고 X, Y, Width 및 Height 데이터 멤버를 0으로 초기화합니다. 이것이 기본 생성자입니다. (오버로드 1/2)

RectF 4개의 정수로 X, Y, Width 및 Height 데이터 멤버를 초기화하여 RectF 개체를 만듭니다.
RedEyeCorrection RedEyeCorrection 개체를 만듭니다.
지역 무한인 지역을 만듭니다. 이것이 기본 생성자입니다. (Region.Region)
지역 다른 지역에서 가져온 데이터로 정의된 지역을 만듭니다.
지역 경로(GraphicsPath 개체)로 정의되고 GraphicsPath 개체에 포함된 채우기 모드가 있는 영역을 만듭니다.
지역 사각형으로 정의된 영역을 만듭니다. (오버로드 1/2)
지역 사각형으로 정의된 영역을 만듭니다. (오버로드 2/2)
지역 무한인 지역을 만듭니다. 이것이 기본 생성자입니다. (오버로드 2/3)
지역 무한인 지역을 만듭니다. 이것이 기본 생성자입니다. (오버로드 1/3)
지역 Windows GDI(그래픽 디바이스 인터페이스) 영역에 대한 핸들로 지정된 지역과 동일한 지역을 만듭니다.
ReleaseHDC Graphics::ReleaseHDC 메서드는 이 Graphics 개체의 Graphics::GetHDC 메서드에 대한 이전 호출에서 얻은 디바이스 컨텍스트 핸들을 해제합니다.

RemovePropertyItem Image::RemovePropertyItem 메서드는 이 Image 개체에서 속성 항목(메타데이터 조각)을 제거합니다.
재설정 GraphicsPath::Reset 메서드는 경로를 비우고 채우기 모드를 FillModeAlternate로 설정합니다.
재설정 ImageAttributes::Reset 메서드는 지정된 범주에 대한 모든 색 및 회색조 조정 설정을 지웁니다.
재설정 Matrix::Reset 메서드는 ID 행렬의 요소로 이 행렬을 업데이트합니다.
ResetClip Graphics::ResetClip 메서드는 이 Graphics 개체의 클리핑 영역을 무한 영역으로 설정합니다.
ResetTransform LinearGradientBrush::ResetTransform 메서드는 이 선형 그라데이션 브러시의 변환 매트릭스를 ID 행렬로 다시 설정합니다. 즉, 변환이 수행되지 않습니다.
ResetTransform TextureBrush::ResetTransform 메서드는 이 텍스처 브러시의 변환 매트릭스를 ID 행렬로 다시 설정합니다. 즉, 변환이 수행되지 않습니다.
ResetTransform Graphics::ResetTransform 메서드는 이 Graphics 개체의 월드 변환 매트릭스를 ID 행렬로 설정합니다.
ResetTransform PathGradientBrush::ResetTransform 메서드는 이 경로 그라데이션 브러시의 변환 매트릭스를 ID 행렬로 다시 설정합니다. 즉, 변환이 수행되지 않습니다.
ResetTransform Pen::ResetTransform 메서드는 이 Pen 개체의 월드 변환 매트릭스를 ID 행렬로 설정합니다.
복원 Graphics::Restore 메서드는 이 Graphics 개체의 Graphics::Save 메서드에 대한 이전 호출에서 저장한 상태로 이 Graphics 개체의 상태를 설정합니다.

Reverse GraphicsPath::Reverse 메서드는 이 경로의 선과 곡선을 정의하는 점의 순서를 반대로 바꿉니다.
되감기 GraphicsPathIterator::Rewind 메서드는 이 반복기를 연결된 경로의 시작 부분으로 되감습니다.
회전 Matrix::Rotate 메서드는 이 행렬을 자체의 곱과 회전 행렬로 업데이트합니다.
RotateAt Matrix::RotateAt 메서드는 이 행렬을 자체의 곱과 지정된 지점에 대한 회전을 나타내는 행렬로 업데이트합니다.
RotateFlip Image::RotateFlip 메서드는 이 이미지를 회전하고 대칭 이동합니다.
RotateTransform LinearGradientBrush::RotateTransform 메서드는 이 브러시의 현재 변환 매트릭스를 자체의 곱과 회전 행렬로 업데이트합니다.
RotateTransform TextureBrush::RotateTransform 메서드는 이 텍스처 브러시의 현재 변환 매트릭스를 자체의 제품 및 회전 행렬로 업데이트합니다.
RotateTransform Graphics::RotateTransform 메서드는 이 Graphics 개체의 월드 변환 매트릭스를 자체의 곱과 회전 행렬로 업데이트합니다.
RotateTransform PathGradientBrush::RotateTransform 메서드는 이 브러시의 현재 변환 매트릭스를 자체의 곱과 회전 행렬로 업데이트합니다.
RotateTransform Pen::RotateTransform 메서드는 이 Pen 개체의 월드 변환 매트릭스를 자체의 곱과 회전 행렬로 업데이트합니다.
저장 Image::Save 메서드는 이 이미지를 파일에 저장합니다.

저장 Image::Save 메서드는 이 이미지를 스트림에 저장합니다.
저장 Graphics::Save 메서드는 이 Graphics 개체의 현재 상태(변환, 클리핑 영역 및 품질 설정)를 저장합니다. 나중에 Graphics::Restore 메서드를 호출하여 상태를 복원할 수 있습니다.
SaveAdd Image::SaveAdd 메서드는 Save 메서드에 대한 이전 호출에서 지정된 파일 또는 스트림에 프레임을 추가합니다. (오버로드 1/2)
SaveAdd Image::SaveAdd 메서드는 Save 메서드에 대한 이전 호출에서 지정된 파일 또는 스트림에 프레임을 추가합니다. (오버로드 2/2)
규모 Matrix::Scale 메서드는 이 행렬을 자체 제품 및 크기 조정 매트릭스로 업데이트합니다.
ScaleTransform LinearGradientBrush::ScaleTransform 메서드는 이 브러시의 현재 변환 매트릭스를 자체의 제품 및 크기 조정 매트릭스로 업데이트합니다.
ScaleTransform TextureBrush::ScaleTransform 메서드는 이 텍스처 브러시의 현재 변환 매트릭스를 자체 제품 및 크기 조정 매트릭스로 업데이트합니다.
ScaleTransform Graphics::ScaleTransform 메서드는 이 Graphics 개체의 월드 변환 매트릭스를 자체 제품 및 크기 조정 매트릭스로 업데이트합니다.
ScaleTransform PathGradientBrush::ScaleTransform 메서드는 이 브러시의 현재 변환 매트릭스를 자체의 제품 및 크기 조정 매트릭스로 업데이트합니다.
ScaleTransform Pen::ScaleTransform 메서드는 펜 개체의 월드 변환 매트릭스를 자체의 곱 및 크기 조정 매트릭스와 동일하게 설정합니다.

SelectActiveFrame Image::SelectActiveFrame 메서드는 차원 및 인덱스로 지정된 이 Image 개체의 프레임을 선택합니다.
SetAbort Image::SetAbort 메서드는 시간이 많이 걸리는 렌더링 작업 중에 Abort 메서드가 주기적으로 호출되는 개체를 설정합니다.
SetAbort Windows GDI+ 버전 1.0 및 1.1에서는 사용되지 않습니다.
SetAlignment Pen::SetAlignment 메서드는 줄을 기준으로 이 펜 개체의 맞춤을 설정합니다.
SetAlignment StringFormat::SetAlignment 메서드는 레이아웃 사각형의 원점과 관련하여 이 StringFormat 개체의 문자 맞춤을 설정합니다. 레이아웃 사각형은 표시된 문자열의 위치를 지정하는 데 사용됩니다.
SetBaseCap CustomLineCap::SetBaseCap 메서드는 줄 끝에 있는 이 CustomLineCap의 일부로 표시되는 LineCap을 설정합니다.
SetBaseInset CustomLineCap::SetBaseInset 메서드는 이 사용자 지정 줄 한도의 기본 inset 값을 설정합니다. 줄의 끝과 기본 캡 사이의 거리입니다.
SetBlend LinearGradientBrush::SetBlend 메서드는 이 선형 그라데이션 브러시의 혼합 요소와 혼합 위치를 설정하여 사용자 지정 혼합을 만듭니다.
SetBlend PathGradientBrush::SetBlend 메서드는 이 경로 그라데이션 브러시의 혼합 요소와 혼합 위치를 설정합니다.
SetBlendBellShape LinearGradientBrush::SetBlendBellShape 메서드는 이 선형 그라데이션 브러시의 혼합 모양을 설정하여 종 모양의 곡선을 기반으로 사용자 지정 혼합을 만듭니다.

SetBlendBellShape PathGradientBrush::SetBlendBellShape 메서드는 이 경로 그라데이션 브러시의 혼합 모양을 설정합니다.
SetBlendTriangularShape LinearGradientBrush::SetBlendTriangularShape 메서드는 이 선형 그라데이션 브러시의 혼합 모양을 설정하여 삼각형 도형을 기반으로 사용자 지정 혼합을 만듭니다.
SetBlendTriangularShape PathGradientBrush::SetBlendTriangularShape 메서드는 이 경로 그라데이션 브러시의 혼합 모양을 설정합니다.
SetBrush Pen::SetBrush 메서드는 펜이 선을 채우는 데 사용하는 Brush 개체를 설정합니다.
SetBrushRemapTable ImageAttributes::SetBrushRemapTable 메서드는 브러시 범주에 대한 색 다시 매핑 테이블을 설정합니다.
SetCenterColor PathGradientBrush::SetCenterColor 메서드는 이 경로 그라데이션 브러시의 가운데 색을 설정합니다. 가운데 색은 브러시의 중심점에 표시되는 색입니다.
SetCenterPoint PathGradientBrush::SetCenterPoint 메서드는 이 경로 그라데이션 브러시의 중심점을 설정합니다. 기본적으로 중심점은 브러시 경계 경로의 중심이지만 가운데 지점을 경로 내부 또는 외부의 모든 위치로 설정할 수 있습니다.
SetCenterPoint PathGradientBrush::SetCenterPoint 메서드는 이 경로 그라데이션 브러시의 중심점을 설정합니다.
SetClip Graphics::SetClip 메서드는 이 Graphics 개체의 클리핑 영역을 업데이트합니다.
SetClip Graphics::SetClip 메서드는 이 Graphics 개체의 클리핑 영역을 그래픽 경로에 지정된 영역과 자체의 조합인 영역으로 업데이트합니다.

SetClip Graphics::SetClip 메서드는 이 Graphics 개체의 클리핑 영역을 자체 및 사각형의 조합인 영역으로 업데이트합니다. (오버로드 2/2)
SetClip Graphics::SetClip 메서드는 이 Graphics 개체의 클리핑 영역을 자체 및 사각형의 조합인 영역으로 업데이트합니다. (오버로드 1/2)
SetClip Graphics::SetClip 메서드는 이 Graphics 개체의 클리핑 영역을 자체 및 Region 개체에 지정된 영역의 조합인 영역으로 업데이트합니다.
SetClip Graphics::SetClip 메서드는 이 Graphics 개체의 클리핑 영역을 자체 및 Windows GDI(그래픽 디바이스 인터페이스) 영역의 조합인 영역으로 업데이트합니다.
SetColor SolidBrush::SetColor 메서드는 이 단색 브러시의 색을 설정합니다.
SetColor Pen::SetColor 메서드는 이 Pen 개체의 색을 설정합니다.
SetColorKey ImageAttributes::SetColorKey 메서드는 지정된 범주에 대한 색 키(투명도 범위)를 설정합니다.
SetColorMatrices ImageAttributes::SetColorMatrices 메서드는 지정된 범주에 대한 색 조정 행렬과 회색조 조정 행렬을 설정합니다.
SetColorMatrix ImageAttributes::SetColorMatrix 메서드는 지정된 범주에 대한 색 조정 매트릭스를 설정합니다.
SetCompositingMode Graphics::SetCompositingMode 메서드는 이 Graphics 개체의 작성 모드를 설정합니다.
SetCompositingQuality Graphics::SetCompositingQuality 메서드는 이 Graphics 개체의 구성 품질을 설정합니다.

SetCompoundArray Pen::SetCompoundArray 메서드는 이 Pen 개체의 복합 배열을 설정합니다.
SetCustomEndCap Pen::SetCustomEndCap 메서드는 이 Pen 개체에 대한 사용자 지정 끝 캡을 설정합니다.
SetCustomStartCap Pen::SetCustomStartCap 메서드는 이 펜 개체에 대한 사용자 지정 시작 캡을 설정합니다.
SetDashCap Pen::SetDashCap 메서드는 이 Pen 개체의 대시 캡 스타일을 설정합니다.
SetDashOffset Pen::SetDashOffset 메서드는 선의 시작부터 파선의 첫 번째 대시 시작까지의 거리를 설정합니다.
SetDashPattern Pen::SetDashPattern 메서드는 이 Pen 개체에 대한 사용자 지정 대시 및 공백 배열을 설정합니다.
SetDashStyle Pen::SetDashStyle 메서드는 이 Pen 개체의 대시 스타일을 설정합니다.
SetDigitSubstitution StringFormat::SetDigitSubstitution 메서드는 숫자 대체 메서드와 숫자 대체에 해당하는 언어를 설정합니다.
SetDownLevelRasterizationLimit 이 메타파일에 저장된 특정 브러시 비트맵의 해상도를 설정합니다.
SetElements Matrix::SetElements 메서드는 이 행렬의 요소를 설정합니다.
SetEndCap Pen::SetEndCap 메서드는 이 펜 개체의 끝 캡을 설정합니다.
SetFillMode GraphicsPath::SetFillMode 메서드는 이 경로의 채우기 모드를 설정합니다.

SetFillState AdjustableArrowCap::SetFillState 메서드는 화살표 캡의 채우기 상태를 설정합니다. 화살표 뚜껑이 채워지지 않으면 윤곽선만 그려집니다.
SetFocusScales PathGradientBrush::SetFocusScales 메서드는 이 경로 그라데이션 브러시의 포커스 배율을 설정합니다.
SetFormatFlags StringFormat::SetFormatFlags 메서드는 이 StringFormat 개체에 대한 형식 플래그를 설정합니다. 형식 플래그는 StringFormat 개체의 대부분의 특성을 결정합니다.
SetFromCOLORREF Color::SetFromCOLORREF 메서드는 GDI(Windows Graphics Device Interface)COLORREF 값을 사용하여 이 Color 개체의 ARGB 값을 설정합니다.
SetGamma ImageAttributes::SetGamma 메서드는 지정된 범주의 감마 값을 설정합니다.
SetGammaCorrection LinearGradientBrush::SetGammaCorrection 메서드는 이 선형 그라데이션 브러시에 감마 보정을 사용할지 여부를 지정합니다.
SetGammaCorrection PathGradientBrush::SetGammaCorrection 메서드는 이 경로 그라데이션 브러시에 감마 수정을 사용할 수 있는지 여부를 지정합니다.
SetGraphicsPath PathGradientBrush::SetGraphicsPath 메서드는 Windows GDI+ 버전 1.0에서 구현되지 않습니다.
SetHeight AdjustableArrowCap::SetHeight 메서드는 화살표 캡의 높이를 설정합니다. 화살표의 기준에서 꼭짓점까지의 거리입니다.
SetHotkeyPrefix StringFormat::SetHotkeyPrefix 메서드는 핫 키 접두사인 앰퍼샌드(&)가 발견될 때 문자열에서 수행되는 처리 유형을 설정합니다.

SetInterpolationColors LinearGradientBrush::SetInterpolationColors 메서드는 이 선형 그라데이션 브러시 및 해당 혼합 위치에 대해 보간할 색을 설정합니다.
SetInterpolationColors PathGradientBrush::SetInterpolationColors 메서드는 이 경로 그라데이션 브러시의 미리 설정된 색과 혼합 위치를 설정합니다.
SetInterpolationMode Graphics::SetInterpolationMode 메서드는 이 Graphics 개체의 보간 모드를 설정합니다. 보간 모드는 이미지의 크기를 조정하거나 회전할 때 사용되는 알고리즘을 결정합니다.
SetLineAlignment StringFormat::SetLineAlignment 메서드는 레이아웃 사각형의 원점과 관련하여 이 StringFormat 개체의 선 맞춤을 설정합니다.
SetLinearColors LinearGradientBrush::SetLinearColors 메서드는 이 선형 그라데이션 브러시의 시작 색과 끝 색을 설정합니다.
SetLineCap Pen::SetLineCap 메서드는 이 펜으로 그린 선에서 시작, 끝 및 대시의 캡 스타일을 설정합니다.
SetLineJoin Pen::SetLineJoin 메서드는 이 Pen 개체의 줄 조인을 설정합니다.
SetMarker GraphicsPath::SetMarker 메서드는 이 경로의 마지막 점을 표식점으로 지정합니다.
SetMeasurableCharacterRanges StringFormat::SetMeasurableCharacterRanges 메서드는 문자열에 있을 때 MeasureCharacterRanges 메서드로 측정할 수 있는 이 StringFormat 개체에 대한 일련의 문자 범위를 설정합니다.
SetMiddleInset AdjustableArrowCap::SetMiddleInset 메서드는 기본의 중간점이 꼭짓점을 향해 이동하는 단위 수를 설정합니다.

SetMiterLimit Pen::SetMiterLimit 메서드는 이 Pen 개체의 miter 제한을 설정합니다.
SetNoOp ImageAttributes::SetNoOp 메서드는 지정된 범주에 대한 색 조정을 해제합니다. ImageAttributes::ClearNoOp를 호출하여 ImageAttributes::SetNoOp 호출 전에 있던 색 조정 설정을 복원할 수 있습니다.
SetOutputChannel ImageAttributes::SetOutputChannel 메서드는 지정된 범주에 대한 CMYK 출력 채널을 설정합니다.
SetOutputChannelColorProfile ImageAttributes::SetOutputChannelColorProfile 메서드는 지정된 범주에 대한 출력 채널 색 프로필 파일을 설정합니다.
SetPageScale Graphics::SetPageScale 메서드는 이 Graphics 개체의 페이지 변환에 대한 배율 인수를 설정합니다. 페이지 변환은 페이지 좌표를 디바이스 좌표로 변환합니다.
SetPageUnit Graphics::SetPageUnit 메서드는 이 Graphics 개체에 대한 측정 단위를 설정합니다. 페이지 단위는 페이지 좌표를 디바이스 좌표로 변환하는 페이지 변환에 속합니다.
SetPalette Image::SetPalette 메서드는 이 Image 개체의 색상표를 설정합니다.
SetParameters Blur::SetParameters 메서드는 이 Blur 개체의 매개 변수를 설정합니다.
SetParameters BrightnessContrast::SetParameters 메서드는 이 BrightnessContrast 개체의 매개 변수를 설정합니다.
SetParameters ColorBalance::SetParameters 메서드는 이 ColorBalance 개체의 매개 변수를 설정합니다.
SetParameters ColorCurve::SetParameters 메서드는 이 ColorCurve 개체의 매개 변수를 설정합니다.

SetParameters ColorLUT::SetParameters 메서드는 이 ColorLUT 개체의 매개 변수를 설정합니다.
SetParameters ColorMatrixEffect::SetParameters 메서드는 이 ColorMatrixEffect 개체의 5x5 색 행렬을 설정합니다.
SetParameters HueSaturationLightness::SetParameters 메서드는 이 HueSaturationLightness 개체의 매개 변수를 설정합니다.
SetParameters Levels::SetParameters 메서드는 이 Levels 개체의 매개 변수를 설정합니다.
SetParameters RedEyeCorrection::SetParameters 메서드는 이 RedEyeCorrection 개체의 매개 변수를 설정합니다.
SetParameters Sharpen::SetParameters 메서드는 이 Sharpen 개체의 매개 변수를 설정합니다.
SetParameters Tint::SetParameters 메서드는 이 Tint 개체의 매개 변수를 설정합니다.
Setpixel Bitmap::SetPixel 메서드는 이 비트맵에서 지정된 픽셀의 색을 설정합니다.
SetPixelOffsetMode Graphics::SetPixelOffsetMode 메서드는 이 Graphics 개체의 픽셀 오프셋 모드를 설정합니다.
SetPropertyItem Image::SetPropertyItem 메서드는 이 Image 개체에 대한 속성 항목(메타데이터 조각)을 설정합니다. 항목이 이미 있는 경우 해당 내용이 업데이트됩니다. 그렇지 않으면 새 항목이 추가됩니다.
SetRemapTable ImageAttributes::SetRemapTable 메서드는 지정된 범주에 대한 색 다시 매핑 테이블을 설정합니다.

SetRenderingOrigin Graphics::SetRenderingOrigin 메서드는 이 Graphics 개체의 렌더링 원본을 설정합니다. 렌더링 원점은 픽셀당 8비트 및 픽셀당 16비트 디더링에 대한 디더 원점 설정에 사용되며 해치 브러시의 원점도 설정하는 데 사용됩니다.
SetResolution Bitmap::SetResolution 메서드는 이 Bitmap 개체의 해상도를 설정합니다.
SetSmoothingMode Graphics::SetSmoothingMode 메서드는 Graphics 개체의 렌더링 품질을 설정합니다.
SetStartCap Pen::SetStartCap 메서드는 이 펜 개체의 시작 캡을 설정합니다.
SetStrokeCap CustomLineCap::SetStrokeCap 메서드는 이 CustomLineCap 개체를 정의하는 GraphicsPath 개체 내에서 줄을 시작하고 종료하는 데 사용되는 LineCap 개체를 설정합니다.
SetStrokeCaps CustomLineCap::SetStrokeCaps 메서드는 이 CustomLineCap 개체를 정의하는 GraphicsPath 개체 내에서 줄을 시작하고 종료하는 데 사용되는 LineCap 개체를 설정합니다.
SetStrokeJoin CustomLineCap::SetStrokeJoin 메서드는 스트로크의 선 조인 스타일을 설정합니다. 선 조인은 사용자 지정 줄 한도를 구성하는 GraphicsPath 개체 내에서 교차하는 두 줄이 조인되는 방법을 지정합니다.
SetSurroundColors PathGradientBrush::SetSurroundColors 메서드는 이 경로 그라데이션 브러시의 서라운드 색을 설정합니다. 서라운드 색은 브러시의 경계 경로에 있는 불연속 지점에 지정된 색입니다.
SetTabStops StringFormat::SetTabStops 메서드는 이 StringFormat 개체의 탭 중지에 대한 오프셋을 설정합니다.
SetTextContrast Graphics::SetTextContrast 메서드는 이 Graphics 개체의 대비 값을 설정합니다. 대비 값은 앤티앨리어싱 텍스트에 사용됩니다.

SetTextRenderingHint Graphics::SetTextRenderingHint 메서드는 이 Graphics 개체의 텍스트 렌더링 모드를 설정합니다.
SetThreshold ImageAttributes::SetThreshold 메서드는 지정된 범주에 대한 임계값(투명도 범위)을 설정합니다.
SetToIdentity ImageAttributes::SetToIdentity 메서드는 지정된 범주의 색 조정 행렬을 ID 행렬로 설정합니다.
SetTransform LinearGradientBrush::SetTransform 메서드는 이 선형 그라데이션 브러시의 변환 행렬을 설정합니다.
SetTransform TextureBrush::SetTransform 메서드는 이 텍스처 브러시의 변환 매트릭스를 설정합니다.
SetTransform Graphics::SetTransform 메서드는 이 Graphics 개체의 월드 변환을 설정합니다.
SetTransform PathGradientBrush::SetTransform 메서드는 이 경로 그라데이션 브러시의 변환 매트릭스를 설정합니다.
SetTransform Pen::SetTransform 메서드는 이 Pen 개체의 월드 변환을 설정합니다.
SetTrimming StringFormat::SetTrimming 메서드는 이 StringFormat 개체의 트리밍 스타일을 설정합니다. 트리밍 스타일은 레이아웃 사각형에 맞도록 문자열을 자르는 방법을 결정합니다.
SetValue Color::SetValue 메서드는 이 Color 개체의 색을 설정합니다.
SetWidth AdjustableArrowCap::SetWidth 메서드는 화살표 캡의 너비를 설정합니다. 너비는 화살표 밑의 엔드포인트 사이의 거리입니다.

SetWidth Pen::SetWidth 메서드는 이 Pen 개체의 너비를 설정합니다.
SetWidthScale CustomLineCap::SetWidthScale 메서드는 배율 너비의 값을 설정합니다. 이는 선을 그리는 데 사용되는 펜의 너비를 기준으로 사용자 지정 줄 한도의 크기를 조정하는 양입니다. 기본값 1.0은 줄 한도의 크기를 조정하지 않습니다.
SetWrapMode LinearGradientBrush::SetWrapMode 메서드는 이 선형 그라데이션 브러시의 래핑 모드를 설정합니다.
SetWrapMode TextureBrush::SetWrapMode 메서드는 이 텍스처 브러시의 래핑 모드를 설정합니다.
SetWrapMode PathGradientBrush::SetWrapMode 메서드는 이 경로 그라데이션 브러시의 래핑 모드를 설정합니다.
SetWrapMode ImageAttributes::SetWrapMode 메서드는 이 ImageAttributes 개체의 래핑 모드를 설정합니다.
선명하게 Sharpen 개체를 만듭니다.
전단 Matrix::Shear 메서드는 이 행렬을 자체의 곱과 전단 행렬로 업데이트합니다.
크기 새 Size 개체를 만들고 멤버를 0으로 초기화합니다. 이것이 기본 생성자입니다.
크기 Size 개체를 만들고 다른 Size 개체의 멤버를 복사하여 멤버를 초기화합니다.
크기 Size 개체를 만들고 Width 및 Height 데이터 멤버를 초기화합니다.

Sizef SizeF 개체를 만들고 멤버를 0으로 초기화합니다. 이것이 기본 생성자입니다.
Sizef SizeF 개체를 만들고 다른 SizeF 개체의 멤버를 복사하여 멤버를 초기화합니다.
Sizef SizeF 개체를 만들고 Width 및 Height 데이터 멤버를 초기화합니다.
SolidBrush SolidBrush::SolidBrush(IN const Color &) 생성자(gdiplusbrush.h)는 색을 기반으로 SolidBrush 개체를 만듭니다.
SolidBrush **SolidBrush**에 대한 복사 생성자입니다.
SolidBrush SolidBrush::SolidBrush~r3 생성자(gdiplusbrush.h)는 색을 기반으로 SolidBrush 개체를 만듭니다.
StartFigure GraphicsPath::StartFigure 메서드는 현재 그림을 닫지 않고 새 그림을 시작합니다. 이 경로에 추가된 후속 지점이 새 그림에 추가됩니다.
StringFormat 다른 StringFormat 개체에서 StringFormat 개체를 만듭니다.
StringFormat 이 항목에서는 StringFormat 클래스의 생성자를 나열합니다. 전체 클래스 목록은 StringFormat 클래스를 참조하세요. (오버로드 2/2)
StringFormat 이 항목에서는 StringFormat 클래스의 생성자를 나열합니다. 전체 클래스 목록은 StringFormat 클래스를 참조하세요. (오버로드 1/2)
StringFormat 문자열 형식 플래그 및 언어를 기반으로 StringFormat 개체를 만듭니다.

TextureBrush TextureBrush::TextureBrush(const TextureBrush &) 생성자(gdiplusbrush.h)는 TextureBrush 클래스의 생성자를 나열합니다.
TextureBrush 이미지, 정의 사각형 및 이미지 속성 집합을 기반으로 TextureBrush 개체를 만듭니다. (오버로드 1/2)
TextureBrush 이미지, 정의 사각형 및 이미지 속성 집합을 기반으로 TextureBrush 개체를 만듭니다. (오버로드 2/2)
TextureBrush 이미지 및 래핑 모드를 기반으로 TextureBrush 개체를 만듭니다. 브러시 크기는 기본적으로 이미지 크기로 설정되므로 전체 이미지가 브러시에서 사용됩니다.
TextureBrush 이미지, 래핑 모드 및 정의 사각형을 기반으로 TextureBrush 개체를 만듭니다. (오버로드 1/2)
TextureBrush 이미지, 래핑 모드 및 정의 사각형을 기반으로 TextureBrush 개체를 만듭니다. (오버로드 2/2)
TextureBrush 이미지, 래핑 모드 및 정의 좌표 집합을 기반으로 TextureBrush 개체를 만듭니다. (오버로드 1/2)
TextureBrush 이미지, 래핑 모드 및 정의 좌표 집합을 기반으로 TextureBrush 개체를 만듭니다. (오버로드 2/2)
TextureBrush TextureBrush::TextureBrush~r9 생성자(gdiplusbrush.h)에는 TextureBrush 클래스의 생성자가 나열됩니다.
색조 Tint 개체를 만듭니다.
ToCOLORREF Color::ToCOLORREF 메서드는 이 Color 개체의 ARGB 값을 GDI(Windows Graphics Device Interface)COLORREF 값으로 변환합니다.

변환 Region::Transform 메서드는 각 데이터 요소를 지정된 행렬에 곱하여 이 지역을 변환합니다.
변환 GraphicsPath::Transform 메서드는 이 경로의 각 데이터 요소를 지정된 행렬에 곱합니다.
TransformPoints Graphics::TransformPoints 메서드는 점 배열을 한 좌표 공간에서 다른 좌표 공간으로 변환합니다. 변환은 이 Graphics 개체의 현재 세계 및 페이지 변환을 기반으로 합니다. (오버로드 1/2)
TransformPoints Graphics::TransformPoints 메서드는 점 배열을 한 좌표 공간에서 다른 좌표 공간으로 변환합니다. 변환은 이 Graphics 개체의 현재 세계 및 페이지 변환을 기반으로 합니다. (오버로드 2/2)
TransformPoints Matrix::TransformPoints 메서드는 배열의 각 점에 이 행렬을 곱합니다. 각 지점은 행 행렬로 처리됩니다. 곱하기는 왼쪽의 행 행렬과 오른쪽에 있는 이 행렬을 사용하여 수행됩니다.
TransformPoints 이 항목에서는 Matrix 클래스의 TransformPoints 메서드를 나열합니다. Matrix 클래스에 대한 메서드의 전체 목록은 행렬 메서드를 참조하세요.
TransformVectors Matrix::TransformVectors 메서드는 배열의 각 벡터에 이 행렬을 곱합니다.
TransformVectors 이 항목에서는 Matrix 클래스의 TransformVectors 메서드를 나열합니다. Matrix 클래스에 대한 메서드의 전체 목록은 행렬 메서드를 참조하세요.
번역 Region::Translate 메서드는 가로 및 세로 방향으로 지정된 양만큼 이 지역을 오프셋합니다. (오버로드 1/2)
번역 Region::Translate 메서드는 가로 및 세로 방향으로 지정된 양만큼 이 지역을 오프셋합니다. (오버로드 2/2)

번역 Matrix::Translate 메서드는 이 행렬을 자체의 곱과 번역 행렬로 업데이트합니다.
TranslateClip Graphics::TranslateClip 메서드는 이 Graphics 개체의 클리핑 영역을 변환합니다. (오버로드 2/2)
TranslateClip Graphics::TranslateClip 메서드는 이 Graphics 개체의 클리핑 영역을 변환합니다. (오버로드 1/2)
TranslateTransform LinearGradientBrush::TranslateTransform 메서드는 이 브러시의 현재 변환 매트릭스를 자체의 곱과 번역 행렬로 업데이트합니다.
TranslateTransform TextureBrush::TranslateTransform 메서드는 이 브러시의 현재 변환 매트릭스를 자체의 제품 및 번역 행렬로 업데이트합니다.
TranslateTransform Graphics::TranslateTransform 메서드는 이 Graphics 개체의 월드 변환 매트릭스를 자체의 제품 및 번역 매트릭스로 업데이트합니다.
TranslateTransform PathGradientBrush::TranslateTransform 메서드는 이 브러시의 현재 변환 매트릭스를 자체의 곱과 번역 행렬로 업데이트합니다.
TranslateTransform Pen::TranslateTransform 메서드는 Pen 개체의 월드 변환 매트릭스를 자체의 곱 및 크기 조정 행렬과 동일하게 설정합니다.
Union Region::Union 메서드는 이 지역을 자체의 모든 부분(교차 및 비연결)과 지정된 경로 내부의 모든 부분으로 업데이트합니다.
Union Region::Union 메서드는 이 지역을 업데이트합니다.

Union Region::Union 메서드는 이 지역을 자체의 모든 부분(교차 및 비연결)과 지정된 사각형 내부의 모든 부분으로 업데이트합니다.
Union Region::Union 메서드는 이 지역을 자체의 모든 부분(교차 및 비연결)과 다른 지역의 모든 부분으로 업데이트합니다.
Union Rect::Union 메서드는 두 사각형의 결합을 결정하고 결과를 Rect 개체에 저장합니다.
Union RectF::Union 메서드는 두 사각형의 결합을 결정하고 결과를 RectF 개체에 저장합니다.
UnlockBits Bitmap::UnlockBits 메서드는 이전에 Bitmap::LockBits 호출로 잠근 이 비트맵의 일부를 잠금 해제합니다.
UseAuxData Effect::UseAuxData 메서드는 Bitmap::ApplyEffect 메서드가 만든 보조 데이터에 대한 포인터를 반환해야 하는지 여부를 지정하는 플래그를 설정하거나 지웁니다.
워프 GraphicsPath::Warp 메서드는 이 경로에 워프 변환을 적용합니다. GraphicsPath::Warp 메서드는 경로를 평면화(직선 시퀀스로 변환)합니다.
넓힐 GraphicsPath::Widen 메서드는 이 경로를 지정된 펜으로 그릴 때 채워진 영역을 묶는 곡선으로 바꿉니다. GraphicsPath::Widen 메서드도 경로를 평면화합니다.
Xor Region::Xor 메서드는 이 지역을 자체의 비인터넥션 부분과 지정된 경로의 내부로 업데이트합니다.
Xor **Region::Xor** 메서드는 영역을 직사각형의 내부를 사용하여 연결되지 않는 부분으로 업데이트합니다.

Xor Region::Xor 메서드는 이 지역을 자체의 비연속 부분과 지정된 사각형의 내부로 업데이트합니다.
Xor Region::Xor 메서드는 이 지역을 자체의 비연속 부분 및 다른 지역으로 업데이트합니다.

구조체

BlurParams BlurParams 구조체에는 가우스 흐림의 특성을 지정하는 멤버가 포함됩니다.
BrightnessContrastParams BrightnessContrastParams 구조체에는 밝기 또는 대비 조정의 특성을 지정하는 멤버가 포함됩니다.
ColorBalanceParams ColorBalanceParams 구조체에는 색 균형 조정의 특성을 지정하는 멤버가 포함됩니다.
ColorCurveParams ColorCurveParams 구조체에는 비트맵의 색에 대한 조정을 지정하는 멤버가 포함됩니다.
ColorLUTParams ColorLUTParams 구조체에는 비트맵에 대한 색 조정을 지정하는 멤버(색 조회 테이블)가 포함됩니다.
Colormap ColorMap 구조체에는 두 개의 Color 개체가 포함됩니다. ImageAttributes 클래스의 여러 메서드는 ColorMap 구조의 배열인 색 다시 매핑 테이블을 사용하여 이미지 색을 조정합니다.
ColorMatrix ColorMatrix 구조체에는 5개의 실제 숫자 행렬이 포함됩니다. ImageAttributes 클래스의 여러 메서드는 색 행렬을 사용하여 이미지 색을 조정합니다.
ColorPalette ColorPalette 구조체는 색상표를 구성하는 색 배열을 정의합니다. 해당 색들은 32비트 ARGB 색입니다.

ENHMETAHEADER3 ENHMETAHEADER3 구조체에는 메타파일 이미지의 차원, 메타파일의 레코드 수 및 메타파일을 만든 디바이스의 해상도를 포함하여 향상된 메타파일 데이터가 포함됩니다.
GdiplusAbort GdiplusAbort 구조체는 Windows GDI+가 시간이 많이 걸리는 렌더링 작업 중에 애플리케이션 정의 Abort 메서드를 주기적으로 호출할 수 있는 메커니즘을 제공합니다.
GdiplusStartupInput **GdiplusStartupInput** 구조체는 GdiplusStartup 함수에 필요한 인수 블록을 보유합니다.
GdiplusStartupInputEx **GdiplusStartupInputEx** 구조체에는 GdiplusStartup 함수에 필요한 인수 블록이 있습니다.
GdiplusStartupOutput GdiplusStartup 함수는 GdiplusStartupOutput 구조를 사용하여 후크 함수에 대한 포인터와 unhook 함수에 대한 포인터를 반환합니다(출력 매개 변수).
HueSaturationLightnessParams HueSaturationLightnessParams 구조체에는 비트맵에 대한 색조, 채도 및 가벼움 조정을 지정하는 멤버가 포함되어 있습니다.
LevelParams LevelsParams 구조체에는 비트맵의 밝게, 중간 또는 어두운 영역에 대한 조정을 지정하는 멤버가 포함되어 있습니다.
PWMFRect16 PWMFRect16 구조체는 배치 가능한 메타파일을 경계로 하는 사각형을 정의합니다. 사각형은 메타파일의 크기와 위치를 정의합니다.
RedEyeCorrectionParams RedEyeCorrectionParams 구조체에는 적목 보정이 적용되는 비트맵 영역을 지정하는 멤버가 포함됩니다.
SharpenParams SharpenParams 구조체에는 비트맵에 대한 선명도 조정의 특성을 지정하는 멤버가 포함됩니다.

TintParams TintParams 구조체에는 비트맵에 대한 색조 조정의 특성을 지정하는 멤버가 포함됩니다.
WmfPlaceableFileHeader WmfPlaceableFileHeader 구조체는 배치 가능한 메타파일 헤더의 필드를 정의합니다. 표시 장치에서 메타파일을 매핑하고 크기를 조정하는 방법을 지정하는 방법으로 배치 가능한 메타파일을 만들었습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

그룹 정책 API

아티클2023. 03. 09.

그룹 정책 API 기술 개요입니다.

그룹 정책 API를 개발하려면 다음 헤더가 필요합니다.

- gpedit.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 그룹 정책 API

열거형

INSTALLSPECTYPE INSTALLSPECTYPE 열거형 값은 그룹 정책 애플리케이션을 InstallApplication 함수에 지정할 수 있는 방법을 정의합니다. 값은 INSTALLDATA의 Type 멤버에 사용됩니다.

Functions

BrowseForGPO BrowseForGPO 함수는 사용자가 GPO를 열거나 만들 수 있는 GPO 브라우저 대화 상자를 만듭니다.
CreateGPOLink CreateGPOLink 함수는 지정된 GPO와 지정된 사이트, 도메인 또는 조직 구성 단위 간에 링크를 만듭니다.
삭제 Delete 메서드는 GPO를 삭제합니다.
DeleteAllGPOLinks DeleteAllGPOLinks 함수는 지정된 사이트, 도메인 또는 조직 구성 단위에 대한 모든 GPO 링크를 삭제합니다.

DeleteGPOLink DeleteGPOLink 함수는 지정된 GPO와 지정된 사이트, 도메인 또는 조직 구성 단위 간의 링크를 삭제합니다.
EnterCriticalPolicySection EnterCriticalPolicySection 함수는 애플리케이션이 정책 설정을 안전하게 읽을 수 있도록 정책 애플리케이션을 일시 중지합니다.
ExportRSoPData ExportRSoPData 함수는 RSoP 정보가 포함된 WMI 네임스페이스를 데이터 파일로 내보냅니다. 이 함수는 ImportRSoPData 함수를 호출하여 WMI 네임스페이스로 가져올 수 있는 데이터 파일에 정보를 씁니다.
FreeGPOListA FreeGPOList 함수는 지정된 GPO 목록을 해제합니다. (ANSI)
FreeGPOListW FreeGPOList 함수는 지정된 GPO 목록을 해제합니다. (유니코드)
GetAppliedGPOListA GetAppliedGPOList 함수는 지정된 사용자 또는 컴퓨터에 적용된 GPO 목록을 검색합니다. (ANSI)
GetAppliedGPOListW GetAppliedGPOList 함수는 지정된 사용자 또는 컴퓨터에 적용된 GPO 목록을 검색합니다. (유니코드)
GetDisplayName GetDisplayName 메서드는 GPO의 표시 이름을 검색합니다. (IGPEInformation.GetDisplayName)
GetDisplayName GetDisplayName 메서드는 GPO의 표시 이름을 검색합니다. (IGroupPolicyObject.GetDisplayName)
GetDSPath GetDSPath 메서드는 GPO의 지정된 섹션에 대한 Active Directory 경로를 검색합니다.
GetDSPath GetDSPath 메서드는 지정된 GPO 섹션의 루트에 대한 Active Directory 경로를 검색합니다.

GetEventLogEntryText GetEventLogEntryText 메서드는 이벤트 로그의 특정 항목에 대한 텍스트를 반환합니다.
GetFileSysPath GetFileSysPath 메서드는 GPO의 지정된 섹션에 대한 파일 시스템 경로를 반환합니다. 경로가 UNC 형식입니다.
GetFileSysPath GetFileSysPath 메서드는 지정된 GPO 섹션의 루트에 대한 파일 시스템 경로를 검색합니다. 경로가 UNC 형식입니다.
GetFlags GetFlags 메서드는 RSoP 사용자 인터페이스 세션에 대한 정보를 검색합니다.
GetGPOListA GetGPOList 함수는 지정된 사용자 또는 컴퓨터에 대한 GPO 목록을 검색합니다. (ANSI)
GetGPOListW GetGPOList 함수는 지정된 사용자 또는 컴퓨터에 대한 GPO 목록을 검색합니다. (유니코드)
GetHint GetHint 메서드는 이 GPO를 연결할 수 있는 Active Directory 개체의 형식을 검색합니다.
GetLocalManagedApplications 대상 컴퓨터에서 GetLocalManagedApplications 함수를 실행하여 해당 컴퓨터에서 관리되는 애플리케이션 목록을 가져올 수 있습니다.
GetMachineName GetMachineName 메서드는 원격 GPO의 컴퓨터 이름을 검색합니다. OpenRemoteMachineGPO 메서드에서 지정한 이름입니다.
GetManagedApplicationCategories GetManagedApplicationCategories 함수는 도메인에 대한 애플리케이션 범주 목록을 가져옵니다. 목록은 도메인의 모든 사용자에 대해 동일합니다.
GetManagedApplications GetManagedApplications 함수는 지정된 사용자 컨텍스트에 대한 ARP(프로그램 추가/제거)의 추가 창에 표시되는 애플리케이션 목록을 가져옵니다.

GetName GetName 메서드는 GPO의 고유한 이름을 검색합니다. 이 값은 일반적으로 GUID입니다.
GetName GetName 메서드는 고유한 GPO 이름을 검색합니다.
GetNamespace GetNameSpace 메서드는 RSoP 데이터가 표시되는 네임스페이스를 검색합니다.
GetOptions GetOptions 메서드는 사용자가 그룹 정책 개체 편집기에서 선택한 옵션을 검색합니다.
GetOptions GetOptions 메서드는 GPO에 대한 옵션을 검색합니다.
GetPath GetPath 메서드는 GPO에 대한 경로를 검색합니다.
GetPropertySheetPages GetPropertySheetPages 메서드는 GPO와 연결된 속성 시트 페이지를 검색합니다.
GetRegistryKey GetRegistryKey 메서드는 GPO의 지정된 섹션에 대한 레지스트리 키의 루트에 대한 핸들을 검색합니다.
GetRegistryKey GetRegistryKey 메서드는 지정된 GPO 섹션에 대한 레지스트리 키의 루트에 대한 핸들을 검색합니다.
GetType GetType 메서드는 편집 중인 GPO에 대한 형식 정보를 검색합니다. (IGPEInformation.GetType)
GetType GetType 메서드는 편집 중인 GPO에 대한 형식 정보를 검색합니다. (IGroupPolicyObject.GetType)

ImportRSoPData ImportRSoPData 함수는 RSoP 데이터가 포함된 데이터 파일을 WMI 네임스페이스로 가져옵니다. 파일은 ExportRSoPData 함수를 호출하여 생성되어야 합니다.
InstallApplication InstallApplication 함수는 도메인에 속한 대상 사용자에게 배포된 애플리케이션을 설치할 수 있습니다.
LeaveCriticalPolicySection LeaveCriticalPolicySection 함수는 정책의 백그라운드 적용을 다시 시작합니다. 이 함수는 정책 섹션에 대한 핸들을 닫습니다.
새로 만들기 New 메서드는 지정된 표시 이름을 사용하여 Active Directory에 새 GPO를 만듭니다. 이 메서드는 OpenDSGPO 메서드를 사용하여 GPO를 엽니다.
OpenDSGPO OpenDSGPO 메서드는 지정된 GPO를 열고 필요에 따라 레지스트리 정보를 로드합니다.
OpenLocalMachineGPO OpenLocalMachineGPO 메서드는 컴퓨터에 대한 기본 GPO를 열고 필요에 따라 레지스트리 정보를 로드합니다.
OpenRemoteMachineGPO OpenRemoteMachineGPO 메서드는 지정된 원격 컴퓨터에 대한 기본 GPO를 열고 필요에 따라 레지스트리 정보를 로드합니다.
PFNGENERATEGROUPPOLICY GenerateGroupPolicy 콜백 함수는 계획 모드에서 RSoP 데이터를 생성할 때 각 정책 확장이 내보내야 하는 애플리케이션 정의 콜백 함수입니다.
PFNPROCESSGROUPPOLICY ProcessGroupPolicy 함수는 정책을 적용할 때 사용되는 애플리케이션 정의 콜백 함수입니다.
PFNPROCESSGROUPPOLICYEX ProcessGroupPolicyEx 함수는 정책을 적용할 때 사용되는 애플리케이션 정의 콜백 함수입니다.

PFNSTATUSMESSAGECALLBACK StatusMessageCallback 함수는 정책을 적용할 때 상태 메시지를 표시하는 데 사용되는 애플리케이션 정의 콜백 함수입니다.
PolicyChanged PolicyChanged 메서드는 그룹 정책 개체 편집기에서 정책 설정이 변경되었다는 사실을 알 수 있습니다.
ProcessGroupPolicyCompleted ProcessGroupPolicyCompleted 함수는 지정된 확장이 정책 적용을 완료했음을 시스템에 알릴 수 있습니다.
ProcessGroupPolicyCompletedEx ProcessGroupPolicyCompletedEx 함수는 지정된 정책 확장이 정책 적용을 완료했음을 시스템에 알릴 수 있습니다. 또한 이 함수는 RSoP(결과 정책 집합) 로깅 상태를 보고합니다.
RefreshPolicy RefreshPolicy 함수를 사용하면 정책이 클라이언트 컴퓨터에 즉시 적용됩니다.
RefreshPolicyEx RefreshPolicyEx 함수를 사용하면 정책이 컴퓨터에 즉시 적용됩니다. 확장 함수를 사용하면 적용할 정책 새로 고침 유형을 지정할 수 있습니다.
RegisterGPNotification RegisterGPNotification 함수를 사용하면 정책이 변경될 때 애플리케이션에서 알림을 받을 수 있습니다. 정책 변경이 발생하면 지정된 이벤트 개체가 신호 상태로 설정됩니다.
RsopAccessCheckByType RSoPAccessCheckByType 함수는 보안 설명자가 RSOPTOKEN으로 식별된 클라이언트에 지정된 액세스 권한 집합을 부여하는지 여부를 결정합니다.
RsopFileAccessCheck RSoPFileAccessCheck 함수는 파일의 보안 설명자가 RSOPTOKEN으로 식별된 클라이언트에 지정된 파일 액세스 권한 집합을 부여하는지 여부를 결정합니다.
RsopResetPolicySettingStatus RSoPResetPolicySettingStatus 함수는 RSOP_PolicySetting 인스턴스에서 RSOP_PolicySettingStatus 인스턴스의 연결을 해제합니다.

RsopSetPolicySettingStatus RSoPSetPolicySettingStatus 함수는 RSOP_PolicySettingStatus 인스턴스와 RSOP_PolicySettingLink 인스턴스를 만듭니다. 함수는 RSOP_PolicySetting 인스턴스에 RSOP_PolicySettingStatus 연결(연결)합니다.
저장 Save 메서드는 지정된 레지스트리 정책 설정을 디스크에 저장하고 GPO의 수정 번호를 업데이트합니다.
SetDisplayName SetDisplayName 메서드는 GPO의 표시 이름을 설정합니다.
SetOptions SetOptions 메서드는 GPO에 대한 옵션을 설정합니다.
UninstallApplication UninstallApplication 함수는 Windows Installer .msi 파일을 사용하여 설치 및 설치를 처리하는 그룹 정책 애플리케이션을 제거합니다.
UnregisterGPNotification UnregisterGPNotification 함수는 정책 변경 알림을 받을 때 지정된 정책 알림 핸들의 등록을 취소합니다.

인터페이스

IGPEInformation IGPEInformation 인터페이스는 MMC(Microsoft Management Console) 확장 스냅인이 그룹 정책 개체 편집기와 통신할 수 있는 메서드를 제공합니다. MMC에 대한 자세한 내용은 Microsoft 관리 콘솔을 참조하세요.
IGroupPolicyObject IGroupPolicyObject 인터페이스는 그룹 정책 개체 편집기를 사용하지 않고 GPO를 직접 만들고 수정하는 메서드를 제공합니다.

IRSOPInformation IRSOPInformation 인터페이스는 MMC(Microsoft Management Console) 확장 스냅인이 주 RSoP(Resultant Set of Policy) 스냅인과 통신하는 메서드를 제공합니다. MMC에 대한 자세한 내용은 Microsoft 관리 콘솔을 참조하세요.

구조체

GPOBROWSEINFO GPOBROWSEINFO 구조체에는 BrowseForGPO 함수가 GPO 브라우저 대화 상자를 초기화하는 데 사용하는 정보가 포함되어 있습니다. 사용자가 대화 상자를 닫은 후 시스템은 이 구조에서 사용자의 작업에 대한 정보를 반환합니다.
GROUP_POLICY_OBJECTA GROUP_POLICY_OBJECT 구조체는 GPO 목록의 GPO에 대한 정보를 제공합니다. (ANSI)
GROUP_POLICY_OBJECTW GROUP_POLICY_OBJECT 구조체는 GPO 목록의 GPO에 대한 정보를 제공합니다. (유니코드)
INSTALLDATA INSTALLDATA 구조는 InstallApplication에서 설치할 그룹 정책 애플리케이션을 지정합니다.
INSTALLSPEC INSTALLSPEC 구조는 사용자 친화적인 이름 및 그룹 정책 GUID 또는 파일 이름 확장명을 사용하여 그룹 정책 애플리케이션을 지정합니다. INSTALLDATA 구조의 Spec 멤버는 InstallApplication 함수에 이 정보를 제공합니다.
LOCALMANAGEDAPPLICATION LOCALMANAGEDAPPLICATION 구조는 사용자 또는 컴퓨터에 대해 설치된 관리되는 애플리케이션을 설명합니다. GetLocalManagedApplications 함수에서 반환됩니다.
MANAGEDAPPLICATION MANAGEDAPPLICATION 구조에는 애플리케이션에 대한 정보가 포함됩니다. GetManagedApplications 함수는 MANAGEDAPPLICATION 구조체의 배열을 반환합니다.

POLICYSETTINGSTATUSINFO POLICYSETTINGSTATUSINFO 구조체는 정책 설정 이벤트에 대한 정보를 제공합니다.
RSOP_TARGET RSOP_TARGET 구조에는 GenerateGroupPolicy 함수에 필요한 컴퓨터 및 사용자 정보가 포함됩니다.

그룹 정책 관리 콘솔

아티클2023. 08. 22.

그룹 정책 관리 콘솔 기술 개요입니다.

그룹 정책 관리 콘솔을 개발하려면 다음 헤더가 필요합니다.

- gpmgmt.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 그룹 정책 관리 콘솔

열거형

GPMBackupType 만든 백업 유형입니다.
GPMPermissionType 범주, 범주에 포함된 권한 및 적용할 수 있는 개체입니다.
GPMReportingOptions 그룹 정책 관리 콘솔 보고서에 대한 옵션입니다.
GPMSearchProperty 검색 조건의 속성입니다.
GPMStarterGPOType Starter 그룹 정책 개체는 시스템 Starter 그룹 정책 개체 또는 사용자 지정 Starter 그룹 정책 개체입니다.

Functions

추가

IGPMGPO 및 IGPMBackup 인터페이스의 항목을 추가합니다. 메서드는 마이그레이션 테이블에 이미 있는 항목을 업데이트합니다.
추가 검색 작업에 대한 조건을 추가합니다.
추가 GPMPermission 개체에 지정된 권한을 GPMSecurityInfo 컬렉션에 추가합니다. 기존 사용 권한 수준보다 높은 권한을 추가할 수 있습니다. 적용되는 제한 사항에 대한 자세한 내용은 다음 설명 섹션을 참조하세요.
AddEntry 마이그레이션 테이블에 항목을 만듭니다. 메서드는 기존 항목을 업데이트합니다.
Backup GPO(그룹 정책 개체)를 지정된 디렉터리에 백업합니다.
Backup 현재 Starter GPO의 백업을 만듭니다.
취소 클라이언트는 이 메서드를 호출하여 GPMC(비동기 그룹 정책 관리 콘솔) 작업을 취소합니다. 백업, 복원, 가져오기, 복사 및 보고서 생성과 같은 GPMC 작업은 비동기적으로 실행할 수 있습니다.
CopyTo 현재 GPO(그룹 정책 개체)를 지정된 도메인에 복사한 다음 GPO의 복사본에 대한 포인터를 반환합니다.
CopyTo CopyTo 메서드는 현재 Starter GPO를 복사하고 시작 GPO의 복사본에 대한 포인터를 반환합니다.
CreateGPO 기본 표시 이름을 사용하여 GPMGPO 개체를 만들고 검색합니다. 일반적으로 호출자는 이 메서드를 호출한 직후 표시 이름을 설정합니다.
CreateGPOFromStarterGPO GPMStarterGPO 개체에서 GPMGPO 개체를 만들고 검색합니다.
CreateGPOLink

지정된 GPO를 특정 SOM에 연결된 GPO 목록의 지정된 위치에 연결합니다.
CreateMigrationTable 빈 마이그레이션 테이블을 만듭니다.
CreatePermission 트러스티(예: 사용자, 컴퓨터 또는 보안 그룹)와 단일 개체에 적용되는 권한을 나타내는 인터페이스 또는 개체를 만들고 반환합니다. 예를 들어 GPO, SOM 또는 WMI 필터에 대해 설명합니다.
CreateQueryResults RSoP(결과 정책 집합) 쿼리를 실행합니다.
CreateSearchCriteria GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 검색 작업을 수행하는 데 사용할 조건을 나타내는 GPMSearchCriteria를 만들고 반환합니다.
CreateStarterGPO GPMStarterGPO 개체를 만들고 검색합니다.
CreateTrustee 트러스티에 대한 정보를 검색할 수 있는 GPMTrustee를 만들고 반환합니다.
삭제 백업 디렉터리와 파일 시스템에서 GPO(그룹 정책 개체) 백업을 제거합니다.
삭제 디렉터리 서비스 및 시스템 볼륨 폴더(SYSVOL)에서 GPO(그룹 정책 개체)를 삭제합니다.
삭제 SOM(관리 scope)에서 GPO 링크를 제거합니다. 메서드는 GPO를 삭제하지 않습니다.
삭제 현재 도메인의 시스템 볼륨 폴더(SysVol)에서 GPO를 삭제합니다.
삭제 백업 디렉터리와 파일 시스템에서 Starter GPO 백업을 제거합니다.
DeleteEntry

마이그레이션 테이블에서 항목을 삭제합니다.
ErrorCode GPMC 작업 중에 발생한 오류를 반환합니다.
GenerateReport GPO(백업 그룹 정책 개체)에 대한 보고서를 가져옵니다.
GenerateReport GPO에 대한 보고서를 가져옵니다.
GenerateReport GenerateReport 메서드는 RSoP 데이터에 대한 보고서를 생성합니다.
GenerateReport Starter GPO에 대한 보고서를 가져옵니다.
GenerateReport GenerateReport 메서드는 백업 GPO에 대한 보고서를 가져옵니다.
GenerateReportToFile GenerateReportToFile 메서드는 GPO(백업 그룹 정책 개체)에 대한 보고서를 가져오고 지정된 경로의 파일에 보고서를 저장합니다.
GenerateReportToFile GPO에 대한 보고서를 가져오고 지정된 경로의 파일에 보고서를 저장합니다.
GenerateReportToFile GenerateReportToFile 메서드는 RSoP 데이터에 대한 보고서를 생성하고 지정된 경로의 파일에 저장합니다.
GenerateReportToFile GenerateReportToFile 메서드는 GPO에 대한 보고서를 가져오고 지정된 경로의 파일에 저장합니다.
GenerateReportToFile GenerateReportToFile은 백업 시작 GPO에 대한 보고서를 가져오고 지정된 경로의 파일에 저장합니다.

get__NewEnum 컬렉션의 열거자를 검색합니다. (IGPMBackupCollection.get__NewEnum)
get__NewEnum 컬렉션의 열거자를 검색합니다. (IGPMCSECollection.get__NewEnum)
get__NewEnum 컬렉션의 열거자를 검색합니다. (IGPMGPOCollection.get__NewEnum)
get__NewEnum get_NewEnum 메서드는 컬렉션에 대한 열거자를 검색합니다.
get__NewEnum 컬렉션의 열거자를 검색합니다. (IGPMMapEntryCollection.get__NewEnum)
get__NewEnum 컬렉션의 열거자를 검색합니다. (IGPMSecurityInfo.get__NewEnum)
get__NewEnum 컬렉션의 열거자를 검색합니다. (IGPMSOMCollection.get__NewEnum)
get__NewEnum 컬렉션의 열거자를 검색합니다. (IGPMstarterGPOBackupCollection.get__NewEnum)
get__NewEnum 컬렉션의 열거자를 검색합니다. (IGPMStarterGPOCollection.get__NewEnum)
get__NewEnum 컬렉션의 열거자를 검색합니다. (IGPMStatusMsgCollection.get__NewEnum)
get__NewEnum 컬렉션의 열거자를 검색합니다. (IGPMWMIFilterCollection.get__NewEnum)
get_Count 컬렉션의 GPMBackup 개체 수를 반환합니다.
get_Count

컬렉션의 CSE(클라이언트 쪽 확장) 수를 반환합니다.
get_Count 컬렉션의 GPO 수를 반환합니다. (IGPMGPOCollection.get_Count)
get_Count 컬렉션의 GPO 링크 수를 반환합니다.
get_Count 컬렉션의 맵 항목 수를 반환합니다.
get_Count 컬렉션의 GPMPermission 개체 수를 반환합니다.
get_Count 컬렉션의 SOM 수를 반환합니다.
get_Count 컬렉션의 GPMStarterGPOBackup 개체 수를 반환합니다.
get_Count 컬렉션의 GPO 수를 반환합니다. (IGPMStarterGPOCollection.get_Count)
get_Count 컬렉션의 메시지 수를 반환합니다.
get_Count 컬렉션의 WMI 필터 수를 반환합니다.
get_Item 인덱스가 지정된 경우 컬렉션에서 GPMBackup 개체를 반환합니다.
get_Item 인덱스가 지정된 경우 컬렉션에서 클라이언트 쪽 확장을 반환합니다.
get_Item 인덱스가 지정되면 컬렉션에서 GPMGPO 개체에 대한 포인터를 반환합니다.

get_Item 인덱스가 지정되면 컬렉션에서 GPO 링크를 반환합니다.
get_Item 인덱스가 지정되면 컬렉션에서 맵 항목을 반환합니다.
get_Item 인덱스가 지정되면 컬렉션에서 GPMPermission 개체를 반환합니다.
get_Item 인덱스가 지정되면 컬렉션에서 SOM을 반환합니다.
get_Item 인덱스가 지정되면 컬렉션에서 GPMStarterGPOBackup 개체를 반환합니다.
get_Item 인덱스가 지정되면 컬렉션에서 IGPMGPO 인터페이스에 대한 포인터를 반환합니다.
get_Item 인덱스가 지정되면 컬렉션에서 메시지를 반환합니다.
get_Item 인덱스가 지정된 경우 컬렉션에서 WMI 필터를 반환합니다.
get_SecurityFlags GPO에 대해 검색하거나 설정할 보안 설명자의 부분을 나타내는 SecurityFlags 속성의 값을 검색합니다.
GetBackup GUID(백업 ID)가 지정된 GPMBackup 개체를 검색합니다. 백업 ID는 GPO의 ID가 아니라 백업된 GPO의 ID입니다.
GetBackup 지정된 백업 ID를 사용하여 GPMBackup 또는 GPMStarterGPOBackup 개체를 검색합니다. 백업 ID는 GUID입니다. 백업 ID는 GPO의 ID가 아닌 백업된 GPO(그룹 정책 개체)의 ID입니다.
GetBackupDir

GPMBackup 및 GPMBackupCollection 개체에 액세스하는 데 사용할 수 있는 GPMBackupDir 개체를 만들고 반환합니다.
GetBackupDirEx GPO(그룹 정책 개체)의 경우 GetBackupDirEx 메서드는 GPMBackup 또는 GPMBackupCollection 개체에 액세스하는 데 사용할 수 있는 GPMBackupDirEx 개체를 만들고 반환합니다.
GetClientSideExtensions 로컬 컴퓨터에 등록된 그룹 정책 CSE(클라이언트 쪽 확장)를 열거할 수 있는 GPMCSECollection 개체를 만들고 반환합니다.
GetConstants GPMC(여러 그룹 정책 관리 콘솔) 상수의 값을 검색할 수 있는 GPMConstants 개체를 만들고 반환합니다.
GetDomain 지정된 도메인에 해당하는 GPMDomain 개체를 만들고 반환합니다.
GetEntries IGPMMapEntryCollection 인터페이스를 반환합니다.
GetEntry GetEntry 메서드는 지정된 원본 필드에 대한 마이그레이션 테이블의 항목을 가져옵니다.
GetGPO 지정된 GPO(그룹 정책 개체) ID를 사용하여 GPMGPO 개체를 검색합니다. 그룹 정책 개체 ID는 GUID로 표시됩니다.
GetGPOLinks SOM(관리 scope)에 대한 GPO 링크가 포함된 GPMGPOLinksCollection 개체를 반환합니다. 컬렉션은 SOM 링크 순서로 정렬되며 활성화된 링크와 사용하지 않도록 설정된 링크를 모두 포함합니다. SOM 링크 순서의 정의는 IGPMGPOLink를 참조하세요.
GetInheritedGPOLinks 부모 컨테이너(OU 및 도메인)에서 상속된 링크를 포함하여 SOM(관리 scope 적용된 GPO 링크)이 포함된 GPOLinksCollection 개체를 반환합니다.
GetMigrationTable 지정된 경로에 마이그레이션 테이블을 로드합니다.

GetQueryList WMI 필터에 저장된 쿼리 목록을 검색합니다.
GetRSOP GPMRSOP를 만들고 반환합니다. RSoP(결과 정책 집합) 모드와 WMI(Windows Management Instrumentation) 네임스페이스를 지정할 수 있습니다.
GetSecurityDescriptor GPO(그룹 정책 개체)의 보안 설명자를 검색할 수 있는 IDispatch 인터페이스에 대한 포인터를 검색합니다. 스크립트 프로그래머의 경우 이 메서드는 IADsSecurityDescriptor 개체에 대한 참조를 반환합니다.
GetSecurityInfo GPO에 대한 사용 권한 집합(예: 편집 권한이 부여된 사용자)을 검색합니다.
GetSecurityInfo SOM(관리 scope)에 대한 GPMPermission 개체의 컬렉션을 나타내는 개체를 반환합니다.
GetSecurityInfo 시작 GPO에 대한 사용 권한 집합(예: 편집 권한이 부여된 사용자)을 검색합니다.
GetSecurityInfo 현재 WMI 필터에 대한 사용 권한 목록을 나타내는 인터페이스 또는 개체를 반환합니다.
GetSite 사이트에 해당하는 SOM(관리) 개체의 scope 반환합니다.
GetSitesContainer 사이트를 열고 쿼리할 수 있는 GPMSitesContainer 개체를 만들고 반환합니다.
GetSOM 지정된 경로에서 도메인 또는 OU(조직 구성 단위)를 나타내는 IGPMSOM 인터페이스를 검색합니다.
GetStarterGPO 지정된 그룹 정책 개체 ID가 있는 GPMStarterGPO 개체를 검색합니다.

GetWMIFilter 지정된 경로에 대한 GPMWMIFilter 개체를 검색합니다.
GetWMIFilter GPO(그룹 정책 개체)에 연결된 GPMWMIFilter 개체를 검색합니다.
가져오기 지정된 GPMBackup 개체에서 정책 설정을 가져옵니다.
InitializeReporting InitializeReporting 메서드는 .adm 파일을 검색할 위치를 설정합니다. 이 메서드는 비동기 방식으로 보고를 초기화합니다.
InitializeReportingEx .adm 파일을 검색할 위치와 보고서에 주석을 포함할지 여부를 결정하는 보고 옵션을 설정합니다. 이 메서드는 비동기 방식으로 보고를 초기화합니다.
IsACLConsistent 디렉터리 서비스와 시스템 볼륨 폴더(SysVol) 간의 ACL 일관성을 확인합니다.
IsComputerEnabled 컴퓨터 정책을 처리하는 동안 클라이언트 쪽 확장을 호출할 수 있는지 여부를 확인합니다.
IsComputerEnabled GPO의 컴퓨터 정책을 사용할 수 있는지 여부를 확인합니다.
IsUserEnabled 사용자 정책을 처리하는 동안 클라이언트 쪽 확장을 호출할 수 있는지 여부를 확인합니다.
IsUserEnabled GPO의 사용자 정책을 사용할 수 있는지 여부를 확인합니다.
LoadStarterGPO CAB(Starter 그룹 정책 개체) 캐비닛 파일을 열고 도메인으로 가져옵니다.
LoggingEnumerateUsers 특정 컴퓨터에 로깅 모드 데이터가 있는 모든 사용자를 열거합니다.

MakeACLConsistent GPO의 디렉터리 서비스 및 시스템 볼륨 폴더(SysVol)에서 ACL을 일관되게 만듭니다. IsACLConsistent를 사용하여 디렉터리 서비스와 시스템 볼륨 폴더(SysVol) 간의 ACL 일관성을 위해 검사 수 있습니다.
OperationCode GPMC 작업과 관련된 코드를 반환합니다.
OverallStatus 복사, 복원, 백업 또는 가져오기와 같은 GPMC 작업의 전체 상태 반환합니다. 작업 중에 오류가 발생하지 않으면 메서드는 성공 코드를 반환합니다. 그렇지 않으면 메서드는 실패 코드를 반환합니다.
ReleaseQueryResults IGPMRSOP::CreateQueryResults 메서드를 호출하고 IGPM::GetRSOP 메서드를 호출하여 할당된 WMI 네임스페이스를 해제합니다.
제거 지정된 GPMPermission 개체에 지정된 권한을 GPMSecurityInfo 컬렉션에서 제거합니다.
RemoveTrustee 지정된 트러스티에 대한 모든 정책 관련 권한을 제거합니다. 트러스티는 GPO, SOM 또는 WMI 필터에 대한 권한을 부여할 수 있는 사용자, 컴퓨터 또는 보안 그룹입니다.
RestoreGPO GPMBackup 개체에서 GPO(그룹 정책 개체)를 복원합니다.
RestoreStarterGPO GPMStarterGPOBackup 개체에서 GPO(Starter 그룹 정책 개체)를 복원합니다.
저장 현재 메모리에 있는 마이그레이션 테이블을 지정된 위치에 저장합니다.
저장 모든 시작 GPO 설정을 단일 CAB 파일에 저장합니다.
SearchBackups 지정된 조건에 따라 GPMBackup 개체에 대한 검색을 실행하고 GPMBackupCollection 개체를 반환합니다.

SearchBackups 지정된 조건에 따라 GPMBackup 개체 또는 IGPMStarterGPOBackup 인터페이스에 대한 검색을 실행하고 GPMBackupCollection 또는 GPMStarterGPOBackupCollection 개체를 반환합니다.
SearchGTO 도메인에서 GPMGPO 개체에 대한 검색을 실행한 다음 GPMGPOCollection 개체를 반환합니다.
SearchSites 지정된 검색 조건에 따라 SOM(관리) 개체의 scope 컬렉션을 검색합니다. 이 메서드는 사이트 개체만 반환합니다.
SearchSOM 도메인에서 GPMSOM 개체(도메인 및 조직 구성 단위)에 대한 검색을 실행한 다음 GPMSOMCollection 개체를 반환합니다.
SearchStarterGTO GPMStarterGPO 개체에 대한 검색을 실행합니다.
SearchWMIFilters 도메인에서 GPMWMIFilter 개체에 대한 검색을 실행한 다음 GPMWMIFilterCollection 개체를 반환합니다.
SetComputerEnabled GPO에서 컴퓨터 설정을 사용하거나 사용하지 않도록 설정합니다.
SetSecurityDescriptor GPO에 대한 보안 설명자를 설정합니다. 메서드는 기존 보안 설명자를 대체합니다.
SetSecurityInfo GPO(그룹 정책 개체)에 대한 사용 권한 목록을 설정합니다(예: 편집할 수 있는 권한이 부여된 사용자). 메서드는 기존 사용 권한 목록을 대체합니다.
SetSecurityInfo SOM(관리 scope)에 대한 사용 권한 목록을 지정된 개체의 사용 권한 목록으로 설정합니다.
SetSecurityInfo 그룹 정책 개체(GPO)에 대한 사용 권한 목록을 설정합니다.

SetSecurityInfo 현재 WMI 필터에 대한 사용 권한 목록을 개체에서 지정한 로 설정합니다.
SetUserEnabled GPO에서 사용자 설정을 사용하거나 사용하지 않도록 설정합니다.
SetWMIFilter GPMWMIFilter 개체를 현재 GPO(그룹 정책 개체)에 연결합니다. 이 메서드를 사용하여 GPO에서 기존 WMI 필터의 연결을 해제할 수도 있습니다.
상태 서버는 이 메서드를 호출하여 클라이언트에 GPMC(그룹 정책 Management Console) 작업의 상태 알립니다.
UpdateDestination 마이그레이션 테이블에 있는 항목의 대상 필드를 업데이트. 대상 옵션과 대상을 지정할 수 있습니다.
유효성 검사 마이그레이션 테이블의 유효성을 검사합니다.

인터페이스

IGPM IGPM 인터페이스는 GPMC(그룹 정책 관리 콘솔)의 다른 인터페이스에 액세스하는 메서드와 다양한 검색 작업을 수행할 수 있는 다른 개체를 만드는 메서드를 제공합니다.
IGPM2 IGPM2 인터페이스는 GPMC(그룹 정책 관리 콘솔)의 IGPM 인터페이스의 GPMBackupDir 및 InitializeReporting 메서드를 확장합니다.
IGPMAsyncCancel IGPMAsyncCancel 인터페이스에 대한 포인터는 클라이언트가 비동기적으로 호출하는 GPMC(그룹 정책 Management Console) 메서드에 의해 클라이언트에 반환됩니다.

IGPMAsyncProgress IGPMAsyncProgress 인터페이스는 클라이언트에서 구현하고 비동기적으로 실행할 수 있는 GPMC(그룹 정책 Management Console) 메서드에 입력 매개 변수로 전달할 수 있습니다.
IGPMBackup IGPMBackup 인터페이스는 GPMBackup 개체를 삭제하고 GPMBackup 개체의 다양한 속성을 검색할 수 있는 메서드를 지원합니다.
IGPMBackupCollection IGPMBackupCollection 인터페이스에는 GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 애플리케이션이 GPMBackup 개체 컬렉션에 액세스할 수 있도록 하는 메서드가 포함되어 있습니다.
IGPMBackupDir IGPMBackupDir 인터페이스는 GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 GPMBackup 및 GPMBackupCollection 개체를 쿼리할 수 있는 메서드를 지원합니다.
IGPMBackupDirEx IGPMBackupDirEx 인터페이스는 GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 GPMBackup, GPMBackupCollection, GPMStarterGPOBackup 및 GPMStarterGPOBackupCollection 개체를 쿼리할 수 있는 메서드를 지원합니다.
IGPMClientSideExtension IGPMClientSideExtension 인터페이스는 GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 클라이언트 쪽 확장 속성을 쿼리할 수 있는 메서드를 지원합니다.
IGPMConstants IGPMConstants 인터페이스는 여러 GPMC(그룹 정책 Management Console) 상수의 값을 검색하는 메서드를 지원합니다. GPMConstants 개체를 만들려면 IGPM::GetConstants 메서드를 호출합니다.
IGPMConstants2 IGPMConstants2 인터페이스는 여러 GPMC(그룹 정책 관리 콘솔) 상수의 값을 검색하는 메서드를 지원합니다.
IGPMCSECollection IGPMCSECollection 인터페이스에는 GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 애플리케이션이 CSE(클라이언트 쪽 확장) 컬렉션을 쿼리할 수 있도록 하는 메서드가 포함되어 있습니다.
IGPMDomain 지정된 도메인을 나타내며, GPMC(그룹 정책 Management Console) 인터페이스를 사용할 때

SOM(관리) 개체의 scope 쿼리하고, GPO를 만들고, 복원하고, 쿼리하고, WMI 필터를 만들고 쿼리할 수 있는 메서드를 지원합니다.
IGPMDomain2 지정된 도메인을 나타내며, 그룹 정책 관리 콘솔(GPMC) 인터페이스를 사용할 때 SOM(관리) 개체의 scope 쿼리하고, Starter GPO를 만들고, 복원 및 쿼리하고, WMI 필터를 만들고 쿼리할 수 있는 메서드를 지원합니다.
IGPMGPO IGPMGPO 인터페이스는 디렉터리 서비스에서 GPO(그룹 정책 Objects)를 관리할 수 있는 메서드를 지원합니다.
IGPMGPO2 IGPMGPO2 인터페이스는 디렉터리 서비스에서 GPO(그룹 정책 개체) 및 Starter 그룹 정책 개체를 관리할 수 있는 메서드를 지원합니다.
IGPMGPOCollection IGPMGPOCollection 인터페이스에는 GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 애플리케이션이 GPO(그룹 정책 Objects) 컬렉션에 액세스할 수 있는 메서드가 포함되어 있습니다.
IGPMGPOLink IGPMGPOLink 인터페이스는 SOM(관리 scope)에서 GPO 링크를 제거하고 링크 사용 및 적용을 포함하여 GPO 링크의 다양한 속성을 설정하고 검색할 수 있는 메서드를 지원합니다.
IGPMGPOLinksCollection IGPMGPOLinksCollection 인터페이스에는 애플리케이션이 GPMC(그룹 정책 Management) 인터페이스를 사용할 때 GPO 링크 컬렉션에 액세스할 수 있도록 하는 메서드가 포함되어 있습니다.
IGPMMapEntry IGPMMapEntry 인터페이스는 맵 항목에 대한 액세스를 제공합니다.
IGPMMapEntryCollection IGPMMapEntryCollection 인터페이스를 사용하면 애플리케이션이 맵 항목 개체에 액세스할 수 있습니다.
IGPMMigrationTable IGPMMigrationTable 인터페이스는 마이그레이션 테이블에 대한 인터페이스를 제공합니다.
IGPMPermission

IGPMPermission 인터페이스에는 GPMC를 사용할 때 권한 관련 속성을 검색하는 메서드가 포함되어 있습니다.
IGPMResult IGPMResult 인터페이스에는 복원, 가져오기, 복사 및 백업과 같은 다양한 유형의 GPO 처리 작업을 수행하는 동안 상태 메시지 정보를 검색하는 메서드가 포함되어 있습니다.
IGPMRSOP IGPMRSOP 인터페이스는 로깅 및 계획 모드 모두에서 RSoP(결과 정책 집합) 쿼리 만들기를 지원하는 메서드를 제공합니다.
IGPMSearchCriteria IGPMSearchCriteria 인터페이스를 사용하면 GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 검색 작업에 사용할 조건을 정의할 수 있습니다. GPMSearchCriteria 개체를 만들려면 IGPM::CreateSearchCriteria 메서드를 호출합니다.
IGPMSecurityInfo IGPMSecurityInfo 인터페이스는 GPMSecurityInfo 컬렉션의 메서드를 정의합니다. 이 컬렉션은 SOM(관리 scope), GPO 또는 WMI 필터와 같은 특정 개체에 설정할 수 있는 정책 관련 권한 집합을 나타냅니다.
IGPMSitesContainer IGPMSitesContainer 인터페이스는 포리스트의 사이트를 나타내는 SOM(관리) 개체의 scope 액세스하는 데 필요한 메서드를 제공합니다.
IGPMSOM IGPMSOM 인터페이스에는 SOM(scope 관리)에 대한 GPO 링크를 만들고 검색하고 SOM에 대한 보안 특성 및 다양한 속성을 설정하고 검색할 수 있는 메서드가 포함되어 있습니다. SOM은 사이트, 도메인 또는 OU일 수 있습니다.
IGPMSOMCollection IGPMSOMCollection 인터페이스는 GPMSOM 개체의 컬렉션을 나타냅니다.
IGPMStarterGPO IGPMStarterGPO 인터페이스는 디렉터리 서비스에서 스타터 그룹 정책 개체(GPO)를 관리할 수 있는 메서드를 지원합니다.
IGPMStarterGPOBackup IGPMStarterGPOBackup 인터페이스는 GPMStarterGPOBackup 개체를 삭제하고 GPMStarterGPOBackup 개체의 다양한 속성을 검색할 수 있는 메서드를 지원합니다.

IGPMStarterGPOBackupCollection IGPMStarterGPOBackupCollection 인터페이스에는 GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 애플리케이션이 GPMStarterGPOBackup 개체 컬렉션에 액세스할 수 있도록 하는 메서드가 포함되어 있습니다.
IGPMStarterGPOCollection IGPMStarterGPOCollection 인터페이스에는 GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 애플리케이션이 GPO(그룹 정책 Objects) 컬렉션에 액세스할 수 있도록 하는 메서드가 포함되어 있습니다.
IGPMStatusMessage IGPMStatusMessage 인터페이스에는 GPO 작업과 관련된 상태 메시지의 다양한 속성을 검색하는 속성 메서드가 포함되어 있습니다.
IGPMStatusMsgCollection IGPMStatusMsgCollection 인터페이스에는 애플리케이션이 GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 상태 메시지 컬렉션에 액세스할 수 있는 메서드가 포함되어 있습니다.
IGPMTrustee IGPMTrustee 인터페이스에는 GPMC(그룹 정책 관리 콘솔)를 사용할 때 지정된 트러스티에 대한 정보를 검색하는 메서드가 포함되어 있습니다.
IGPMWMIFilter IGPMWMIFilter 인터페이스에는 WMI 필터에 대한 보안 특성 및 다양한 속성을 설정하고 검색할 수 있는 메서드가 포함되어 있습니다. WMI 필터 쿼리는 WQL(WMI 쿼리 언어)을 사용하여 지정됩니다.
IGPMWMIFilterCollection IGPMWMIFilterCollection 인터페이스에는 GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 애플리케이션이 WMI 필터 컬렉션에 액세스할 수 있도록 하는 메서드가 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

하드웨어 카운터 프로파일링

아티클2023. 03. 09.

하드웨어 카운터 프로파일링 기술 개요입니다.

하드웨어 카운터 프로파일링 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 하드웨어 카운터 프로파일링

열거형

HARDWARE_COUNTER_TYPE 프로파일되는 하드웨어 카운터의 유형을 정의합니다.

Functions

DisableThreadProfiling 스레드 프로파일링을 사용하지 않도록 설정합니다.
EnableThreadProfiling 지정된 스레드에서 스레드 프로파일링을 사용하도록 설정합니다.
QueryThreadProfiling 지정된 스레드에 대해 스레드 프로파일링을 사용할 수 있는지 여부를 결정합니다.
ReadThreadProfilingData 스레드와 연결된 지정된 프로파일링 데이터를 읽습니다.

구조체

HARDWARE_COUNTER_DATA 하드웨어 카운터 값을 포함합니다.
PERFORMANCE_DATA 요청한 스레드 프로파일링 및 하드웨어 카운터 데이터를 포함합니다.

하드웨어 요구 사항 계산기(HWREQCHK)

아티클2023. 06. 02.

HWREQCHK(하드웨어 요구 사항 평가기)는 개발자가 하드웨어 디바이스에 대한 정보를 얻고 컴퓨터가 특정 버전의 Windows 11 이상을 실행할 수 있는지 확인할 수 있는 API 집합입니다.

HWREQCHK용으로 개발하려면 다음 헤더가 필요합니다.

- hwreqchkapi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- HWREQCHK API 개요
- HWREQCHK 예제

일반적인 Win32 프로그래밍 지침은 다음을 참조하세요.

- 오류 처리
- 디버그 도움말 라이브러리
- 동적 링크 라이브러리
- 이벤트 로깅
- 프로세스 및 스레드

열거형

HWREQCHK_CPU_VENDOR CPU 공급업체를 나타냅니다.
HWREQCHK_PRODUCT_TYPE 제품 유형을 나타냅니다.
HWREQCHK_TARGET_RELEASE 대상 Windows 릴리스를 나타냅니다.

Functions

EvaluateHardwareRequirement 특정 요구 사항을 평가하고 디바이스가 하드웨어 요구 사항을 충족하는지 여부를 호출자에게 알리는 통과 또는 실패 결과를 반환합니다.
GetHardwareRequirements 모든 제품 유형에 대해 정의된 하드웨어 요구 사항 컬렉션을 반환합니다.
GetHardwareRequirementSystemInfo 지정된 하드웨어 요구 사항을 평가하고 비교하는 데 사용되는 실제 하드웨어 디바이스 시스템 정보를 제공합니다.
GetLatestHardwareRequirement 지정된 제품 유형에 대해 정의된 최신 요구 사항을 반환합니다.

구조체

HWREQCHK_DEVICE_HARDWARE_EVALUATION 단일 하드웨어 요구 사항 검사 결과를 포함합니다.
HWREQCHK_DEVICE_HARDWARE_REQUIREMENT 단일 하드웨어 요구 사항 검사 결과를 포함합니다.
HWREQCHK_DEVICE_HARDWARE_SYSINFO 디바이스 하드웨어에 대한 정보를 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

높은 DPI

아티클2023. 04. 27.

높은 DPI 기술에 대한 개요입니다.

높은 DPI를 개발하려면 다음 헤더가 필요합니다.

- shellscalingapi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 높은 DPI

열거형

DIALOG_CONTROL_DPI_CHANGE_BEHAVIORS 대화 상자 내의 자식 창에 대한 모니터별 DPI 크기 조정 동작 재정의에 대해 설명합니다. 이 열거형의 값은 비트 필드이며 결합할 수 있습니다.
DIALOG_DPI_CHANGE_BEHAVIORS 모니터별 v2 컨텍스트에서 대화 상자는 자체 크기를 조정하고 자식 창의 위치를 다시 계산하여 DPI 변경에 자동으로 응답합니다(여기서는 다시 레이아웃이라고 함).
DPI_AWARENESS 스레드, 프로세스 또는 창에 대한 dpi(인치당 점) 설정을 식별합니다.
DPI_HOSTING_BEHAVIOR 창에 대한 DPI 호스팅 동작을 식별합니다. 이 동작을 사용하면 스레드에서 만든 창이 다른 DPI_AWARENESS_CONTEXT 있는 자식 창을 호스트할 수 있습니다.
MONITOR_DPI_TYPE 모니터의 dpi(인치당 점) 설정을 식별합니다.
PROCESS_DPI_AWARENESS dpi(인치당 점) 인식 값을 식별합니다. DPI 인식은 애플리케이션이 DPI에 대해 수행하는 크기 조정 작업과 시스템에서 수행하는 작업을 나타냅니다.

Functions

AdjustWindowRectExForDpi 클라이언트 사각형의 원하는 크기 및 제공된 DPI에 따라 창 사각형의 필요한 크기를 계산합니다.
AreDpiAwarenessContextsEqual 두 DPI_AWARENESS_CONTEXT 값이 동일한지 여부를 결정합니다.
EnableNonClientDpiScaling 높은 DPI 디스플레이에서 는 지정된 최상위 창의 비 클라이언트 영역 부분의 자동 디스플레이 크기 조정을 사용하도록 설정합니다. 해당 창을 초기화하는 동안 호출해야 합니다.
GetAwarenessFromDpiAwarenessContext DPI_AWARENESS_CONTEXT DPI_AWARENESS 값을 검색합니다.
GetDialogControlDpiChangeBehavior 대화 상자에서 자식 창의 DPI 크기 조정 동작 재정의를 검색하고 모니터링합니다.
GetDialogDpiChangeBehavior SetDialogDpiChangeBehavior에 대한 이전 호출로 지정된 대화 상자에서 설정되었을 수 있는 플래그를 반환합니다.
GetDpiAwarenessContextForProcess 지정된 프로세스에 대한 DPI_AWARENESS_CONTEXT 핸들을 가져옵니다.
GetDpiForMonitor 디스플레이의 dpi(인치당 점)를 쿼리합니다.
GetDpiForSystem 시스템 DPI를 반환합니다.
GetDpiForWindow 지정된 창의 dpi(인치당 점) 값을 반환합니다.
GetDpiFromDpiAwarenessContext 지정된 DPI_AWARENESS_CONTEXT 핸들에서 DPI를 검색합니다. 이렇게 하면 해당 스레드 내에서 만든 창을 검사할 필요 없이 스레드의 DPI를 확인할 수 있습니다.

GetProcessDpiAwareness 지정된 프로세스의 dpi(인치당 점) 인식을 검색합니다.
GetSystemDpiForProcess 지정된 프로세스와 연결된 시스템 DPI를 검색합니다. 이는 여러 시스템 인식 프로세스 간에 서로 다른 시스템 DPI 값을 사용하여 DPI에 민감한 정보를 공유하여 발생하는 호환성 문제를 방지하는 데 유용합니다.
GetSystemMetricsForDpi 제공된 DPI를 고려하여 지정된 시스템 메트릭 또는 시스템 구성 설정을 검색합니다.
GetThreadDpiAwarenessContext 현재 스레드에 대한 DPI_AWARENESS_CONTEXT 가져옵니다.
GetThreadDpiHostingBehavior 현재 스레드에서 DPI_HOSTING_BEHAVIOR 검색합니다.
GetWindowDpiAwarenessContext 창과 연결된 DPI_AWARENESS_CONTEXT 반환합니다.
GetWindowDpiHostingBehavior 지정된 창의 DPI_HOSTING_BEHAVIOR 반환합니다.
InheritWindowMonitor 지정된 창이 다른 창의 모니터를 상속하도록 합니다.
IsValidDpiAwarenessContext 지정된 DPI_AWARENESS_CONTEXT 유효하고 현재 시스템에서 지원하는지 확인합니다.
LogicalToPhysicalPointForPerMonitorDPI 호출자의 dpi(인치당 점) 인식에 관계없이 창의 점을 논리적 좌표에서 물리적 좌표로 변환합니다.
OpenThemeDataForDpi 특정 DPI와 연결된 테마 핸들을 여는 OpenThemeData의 변형입니다.
PhysicalToLogicalPointForPerMonitorDPI 호출자의 dpi(인치당 점 수) 인식에 관계없이 창의 점을 실제 좌표에서 논리적 좌표로 변환합니다.

SetDialogControlDpiChangeBehavior 대화 상자에서 자식 창의 기본 모니터별 DPI 크기 조정 동작을 재정의합니다.
SetDialogDpiChangeBehavior Per-Monitor v2 컨텍스트의 대화 상자는 자동으로 DPI 크기가 조정됩니다. 이 메서드를 사용하면 DPI 변경 동작을 사용자 지정할 수 있습니다.
SetProcessDpiAwareness 프로세스 기본 DPI 인식 수준을 설정합니다. 이는 해당 DPI_AWARENESS_CONTEXT 값으로 SetProcessDpiAwarenessContext를 호출하는 것과 같습니다.
SetProcessDpiAwarenessContext 현재 프로세스를 지정된 dpi(인치당 점) 인식 컨텍스트로 설정합니다. DPI 인식 컨텍스트는 DPI_AWARENESS_CONTEXT 값에서 가져옵니다.
SetThreadCursorCreationScaling 이 스레드에서 만들어지는 커서가 의도된 DPI 배율을 설정합니다. 이 값은 표시되는 특정 모니터의 커서 크기를 조정하는 경우 고려됩니다.
SetThreadDpiAwarenessContext 현재 스레드에 대한 DPI 인식을 제공된 값으로 설정합니다.
SetThreadDpiHostingBehavior 스레드의 DPI_HOSTING_BEHAVIOR 설정합니다. 이 동작을 사용하면 스레드에서 만든 창이 다른 DPI_AWARENESS_CONTEXT 있는 자식 창을 호스트할 수 있습니다.
SystemParametersInfoForDpi 제공된 DPI 값을 고려하여 시스템 전체 매개 변수 중 하나의 값을 검색합니다.

HLSL

아티클2023. 04. 26.

HLSL 기술 개요입니다.

HLSL을 개발하려면 다음 헤더가 필요합니다.

- d3dcompiler.h
- dxcapi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- HLSL

열거형

D3D_BLOB_PART 임의 길이 데이터 버퍼의 콘텐츠 부분을 식별하는 값입니다.
D3D_BLOB_PART 임의 길이 데이터 버퍼의 콘텐츠 부분을 식별하는 값입니다.
D3DCOMPILER_STRIP_FLAGS 스트립 플래그 옵션입니다.
D3DCOMPILER_STRIP_FLAGS 스트립 플래그 옵션입니다.
DXC_OUT_KIND IDxcResult에서 검색할 출력의 종류를 지정합니다.

Functions

AddArguments 이 컴파일러 인수 목록에 인수를 더 추가합니다.

AddArgumentsUTF8 이 컴파일러 인수 목록에 UTF-8로 인코딩된 인수를 추가합니다.
AddDefines 이 컴파일러 인수 목록에 추가 정의를 추가합니다.
AddPart 컨테이너에 파트를 추가합니다.
AssembleToContainer LL 또는 LLVM 비트 코드에서 DXIL을 DXIL 컨테이너에 어셈블합니다.
BuildArguments Compile 메서드에 전달할 수 있는 인수를 빌드합니다.
Compile 대상 셰이더 모델에 대한 단일 진입점을 컴파일합니다. **IDxcCompiler::Compile** 는 더 이상 사용되지 않습니다. 대신 IDxcCompiler3::Compile를 사용합니다.
Compile 대상 셰이더 모델에 대한 단일 진입점을 컴파일하거나 라이브러리 대상에 라이브러리를 컴파일하거나 루트 서명을 컴파일하거나 HLSL 원본을 전처리합니다.
CompileForFullPDB TBD
CompileWithDebug 디버그 정보를 사용하여 대상 셰이더 모델에 대한 단일 진입점을 컴파일합니다. **IDxcCompiler::CompileWithDebug** 는 더 이상 사용되지 않습니다. 대신 IDxcCompiler3::Compile를 사용합니다.
CreateBlob 새 Blob이 소유한 메모리에 Blob 콘텐츠를 복사합니다. 새 Blob 및 복사된 콘텐츠는 현재 할당자와 함께 할당됩니다.
CreateBlobFromBlob IDxcUtils는 IDxcLibrary를 대체합니다. 대신 IDxcUtils를 사용합니다.

CreateBlobFromBlob 외부 Blob에 대한 참조를 보유하고 해당 메모리를 가리키는 하위 Blob을 만듭니다.
CreateBlobFromFile IDxcUtils는 IDxcLibrary를 대체합니다. 대신 IDxcUtils를 사용합니다.
CreateBlobFromPinned 복사본 없이 기존 메모리를 참조하는 Blob을 만듭니다. 메모리 수명을 별도로 관리해야 합니다.
CreateBlobWithEncodingFromPinned IDxcUtils는 IDxcLibrary를 대체합니다. 대신 IDxcUtils를 사용합니다.
CreateBlobWithEncodingOnHeapCopy IDxcUtils는 IDxcLibrary를 대체합니다. 대신 IDxcUtils를 사용합니다.
CreateBlobWithEncodingOnMalloc IDxcUtils는 IDxcLibrary를 대체합니다. 대신 IDxcUtils를 사용합니다.
CreateDefaultIncludeHandler 기본 파일 기반 포함 처리기를 만듭니다.
CreateIncludeHandler IDxcUtils는 IDxcLibrary를 대체합니다. 대신 IDxcUtils를 사용합니다.
CreateReadOnlyStreamFromBlob Blob에서 데이터를 읽는 스트림을 만듭니다.
CreateReflection 직렬화된 DXIL 컨테이너 또는 **DXC_PART_REFLECTION_DATA** Blob 콘텐츠에서 리플렉션 인터페이스를 만듭니다.
CreateStreamFromBlobReadOnly IDxcUtils는 IDxcLibrary를 대체합니다. 대신 IDxcUtils를 사용합니다.
D3DCompile HLSL 코드 또는 효과 파일을 지정된 대상에 대한 바이트코드로 컴파일합니다.

D3DCompile HLSL 코드 또는 효과 파일을 지정된 대상에 대한 바이트코드로 컴파일합니다.
D3DCompile2 Microsoft HLSL(High Level Shader Language) 코드를 지정된 대상에 대한 바이트코드로 컴파일합니다. (D3DCompile2)
D3DCompile2 Microsoft HLSL(High Level Shader Language) 코드를 지정된 대상에 대한 바이트코드로 컴파일합니다. (D3DCompile2)
D3DCompileFromFile Microsoft HLSL(High Level Shader Language) 코드를 지정된 대상에 대한 바이트코드로 컴파일합니다. (D3DCompileFromFile)
D3DCompileFromFile Microsoft HLSL(High Level Shader Language) 코드를 지정된 대상에 대한 바이트코드로 컴파일합니다. (D3DCompileFromFile)
D3DCompressShaders 셰이더 집합을 보다 압축된 형태로 압축합니다.
D3DCompressShaders 셰이더 집합을 보다 압축된 형태로 압축합니다.
D3DCreateBlob 버퍼를 만듭니다.
D3DCreateBlob 버퍼를 만듭니다.
D3DCreateFunctionLinkingGraph 함수 연결 그래프 인터페이스를 만듭니다.
D3DCreateFunctionLinkingGraph 함수 연결 그래프 인터페이스를 만듭니다.

D3DCreateLinker 링커 인터페이스를 만듭니다. 참고 이 함수는 모든 Direct3D 11 플랫폼에서 미리 컴파일된 HLSL 함수를 만들고 라이브러리에 패키지하고 런타임에 전체 셰이더에 연결하는 데 사용할 수 있는 HLSL 셰이더 연결 기술의 일부입니다. .
D3DCreateLinker 링커 인터페이스를 만듭니다. 참고 이 함수는 모든 Direct3D 11 플랫폼에서 미리 컴파일된 HLSL 함수를 만들고 라이브러리에 패키지하고 런타임에 전체 셰이더에 연결하는 데 사용할 수 있는 HLSL 셰이더 연결 기술의 일부입니다. .
D3DDecompressShaders 압축 집합에서 하나 이상의 셰이더를 압축 해제합니다.
D3DDecompressShaders 압축 집합에서 하나 이상의 셰이더를 압축 해제합니다.
D3DDisassemble 컴파일된 HLSL 코드를 분해합니다.
D3DDisassemble 컴파일된 HLSL 코드를 분해합니다.
D3DDisassemble10Effect Direct3D10 효과에서 컴파일된 HLSL 코드를 분해합니다.
D3DDisassemble10Effect Direct3D10 효과에서 컴파일된 HLSL 코드를 분해합니다.
D3DDisassembleRegion 컴파일된 Microsoft HLSL(High Level Shader Language) 코드의 특정 지역을 디스어셈블합니다.
D3DDisassembleRegion 컴파일된 Microsoft HLSL(High Level Shader Language) 코드의 특정 지역을 디스어셈블합니다.
D3DGetBlobPart 컴파일 결과에서 특정 부분을 검색합니다.

D3DGetBlobPart 컴파일 결과에서 특정 부분을 검색합니다.
D3DGetDebugInfo 참고 이 API를 사용하여 Windows 스토어 앱을 개발할 수 있지만 Windows 스토어에 제출하는 앱에서는 사용할 수 없습니다. 셰이더 디버그 정보를 가져옵니다.
D3DGetDebugInfo 참고 이 API를 사용하여 Windows 스토어 앱을 개발할 수 있지만 Windows 스토어에 제출하는 앱에서는 사용할 수 없습니다. 셰이더 디버그 정보를 가져옵니다.
D3DGetInputAndOutputSignatureBlob 참고 D3DGetInputAndOutputSignatureBlob은 Windows 8.1 후 릴리스에서 변경되거나 사용할 수 없습니다. 대신 D3D_BLOB_INPUT_AND_OUTPUT_SIGNATURE_BLOB 값과 함께 D3DGetBlobPart를 사용합니다. 컴파일 결과에서 입출력 서명을 가져옵니다.
D3DGetInputAndOutputSignatureBlob 참고 D3DGetInputAndOutputSignatureBlob은 Windows 8.1 후 릴리스에서 변경되거나 사용할 수 없습니다. 대신 D3D_BLOB_INPUT_AND_OUTPUT_SIGNATURE_BLOB 값과 함께 D3DGetBlobPart를 사용합니다. 컴파일 결과에서 입출력 서명을 가져옵니다.
D3DGetInputSignatureBlob 참고 D3DGetInputSignatureBlob은 Windows 8.1 후 릴리스에서 변경되거나 사용할 수 없습니다. 대신 D3D_BLOB_INPUT_SIGNATURE_BLOB 값과 함께 D3DGetBlobPart를 사용합니다. 컴파일 결과에서 입력 서명을 가져옵니다.
D3DGetInputSignatureBlob 참고 D3DGetInputSignatureBlob은 Windows 8.1 후 릴리스에서 변경되거나 사용할 수 없습니다. 대신 D3D_BLOB_INPUT_SIGNATURE_BLOB 값과 함께 D3DGetBlobPart를 사용합니다. 컴파일 결과에서 입력 서명을 가져옵니다.
D3DGetOutputSignatureBlob 참고 D3DGetOutputSignatureBlob은 Windows 8.1 후 릴리스에서 변경되거나 사용할 수 없습니다. 대신 D3D_BLOB_OUTPUT_SIGNATURE_BLOB 값과 함께 D3DGetBlobPart를 사용합니다. 컴파일 결과에서 출력 서명을 가져옵니다.

D3DGetOutputSignatureBlob 참고 D3DGetOutputSignatureBlob은 Windows 8.1 후 릴리스에서 변경되거나 사용할 수 없습니다. 대신 D3D_BLOB_OUTPUT_SIGNATURE_BLOB 값과 함께 D3DGetBlobPart를 사용합니다. 컴파일 결과에서 출력 서명을 가져옵니다.
D3DGetTraceInstructionOffsets 셰이더 코드 섹션 내 명령에 대한 바이트 오프셋을 검색합니다.
D3DGetTraceInstructionOffsets 셰이더 코드 섹션 내 명령에 대한 바이트 오프셋을 검색합니다.
D3DLoadModule 셰이더 모듈의 원본 데이터에서 셰이더 모듈 인터페이스를 만듭니다.
D3DLoadModule 셰이더 모듈의 원본 데이터에서 셰이더 모듈 인터페이스를 만듭니다.
D3DPreprocess 컴파일되지 않은 HLSL 코드를 전처리합니다.
D3DPreprocess 컴파일되지 않은 HLSL 코드를 전처리합니다.
D3DReadFileToBlob 디스크에 있는 파일을 메모리로 읽습니다.
D3DReadFileToBlob 디스크에 있는 파일을 메모리로 읽습니다.
D3DReflection 리플렉션 인터페이스에 대한 포인터를 가져옵니다.
D3DReflection 리플렉션 인터페이스에 대한 포인터를 가져옵니다.
D3DReflectLibrary 함수의 HLSL 라이브러리를 포함하는 원본 데이터에서 라이브러리 반사 인터페이스를 만듭니다.

D3DReflectLibrary 함수의 HLSL 라이브러리를 포함하는 원본 데이터에서 라이브러리 반사 인터페이스를 만듭니다.
D3DSetBlobPart 컴파일 결과에 정보를 설정합니다.
D3DSetBlobPart 컴파일 결과에 정보를 설정합니다.
D3DStripShader 컴파일 결과에서 원하지 않는 Blob을 제거합니다.
D3DStripShader 컴파일 결과에서 원하지 않는 Blob을 제거합니다.
D3DWriteBlobToFile 메모리 Blob을 디스크의 파일에 씁니다.
D3DWriteBlobToFile 메모리 Blob을 디스크의 파일에 씁니다.
Disassemble 프로그램을 디스어셈블합니다. **IDxcCompiler::D isassemble** 은 더 이상 사용되지 않습니다. 대신 IDxcCompiler3::D isassemble 을 사용합니다.
Disassemble 프로그램을 디스어셈블합니다.
DxcCreateInstance 지정된 CLSID와 연결된 클래스의 초기화되지 않은 단일 개체를 만듭니다.
DxcCreateInstance2 지정된 CLSID와 연결된 클래스의 초기화되지 않은 단일 개체를 만듭니다(사용자 지정 메모리 할당자를 사용하여 컴파일러의 instance 만드는 데 사용할 수 있음).

DxcCreateInstance2Proc **DxcCreateInstance2Proc** typedef는 **GetProcAddress**를 사용하여 DxcCreateInstance2를 호출하는 것이 더 편리하도록 제공됩니다.
DxcCreateInstanceProc **DxcCreateInstanceProc** typedef는 **GetProcAddress**를 사용하여 DxcCreateInstance를 호출하는 것이 더 편리하도록 제공됩니다.
FindFirstPartKind 지정된 종류가 있는 첫 번째 부분의 인덱스를 검색합니다.
GetArg TBD
GetArgCount TBD
GetArgPair TBD
GetArgPairCount TBD
GetArguments 인수 배열을 검색합니다.
GetAvailablePass TBD
GetAvailablePassCount TBD
GetBlobAsUtf16 IDxcUtils는 IDxcLibrary를 대체합니다. 대신 IDxcUtils를 사용합니다.
GetBlobAsUtf16 일치하는 인코딩된 텍스트 Blob을 UTF-16으로 변환하거나 반환합니다.

GetBlobAsUtf8 IDxcUtils는 IDxcLibrary를 대체합니다. 대신 IDxcUtils를 사용합니다.
GetBlobAsUtf8 일치하는 인코딩된 텍스트 Blob을 UTF-8로 변환하거나 반환합니다.
GetBufferPointer Blob의 데이터에 대한 포인터를 검색합니다.
GetBufferSize Blob 데이터의 크기(바이트 및 null 종결자 포함)를 검색합니다.
GetCommitInfo TBD
GetCount 인수 수를 검색합니다.
GetCustomVersionString TBD
GetDefine TBD
GetDefineCount TBD
GetDescription TBD
GetDxilContainerPart TBD
GetEncoding 이 Blob에 대한 인코딩을 검색합니다.

GetEntryPoint TBD
GetErrorBuffer 작업에서 오류 버퍼를 검색합니다(있는 경우). 이는 DXC_OUT_ERRORS사용하여 IDxcResult::GetOutput을 호출하는 데 해당합니다.
GetFlag TBD
GetFlagCount TBD
GetFlags TBD
GetFullPDB TBD
GetHash TBD
GetMainFileName TBD
GetName TBD
GetNumOutputs 이 결과에서 사용할 수 있는 출력 수를 검색합니다.
GetOptionArgCount TBD
GetOptionArgDescription TBD

GetOptionArgName TBD
GetOptionName TBD
GetOutput 지정된 출력을 검색합니다.
GetOutput 지정된 출력을 검색합니다.
GetOutputByIndex 지정된 인덱스에서 출력 종류를 검색합니다.
GetOutputCount 사용 가능한 출력 수를 검색합니다.
GetPartContent 지정된 부분의 콘텐츠를 검색합니다.
GetPartCount 컨테이너의 파트 수를 검색합니다.
GetPartKind 지정된 부분의 종류를 검색합니다.
GetPartReflection 지정된 부분에 대한 리플렉션 인터페이스를 검색합니다.
GetPDBContents 셰이더 PDB의 해시 및 콘텐츠를 검색합니다.
GetResult 작업의 기본 출력을 검색합니다.

GetSource TBD
GetSourceCount TBD
GetSourceName TBD
GetStatus 작업의 전체 상태 검색합니다.
GetStringLength null 종결자를 제외한 이 Blob에 저장된 문자열의 길이를 문자로 검색합니다.
GetStringPointer 이 Blob에 저장된 문자열에 대한 포인터를 검색합니다.
GetTargetProfile TBD
GetVersion TBD
GetVersionInfo TBD
HasOutput 이 결과에 지정된 출력이 있는지 여부를 확인합니다.
IsFullPDB TBD
링크 셰이더를 연결하고 Direct3D 런타임에서 사용할 수 있는 셰이더 Blob을 생성합니다.

로드 DxilContainer를 작성기로 로드합니다.
로드 리플렉션을 수행할 컨테이너를 선택합니다.
로드 TBD
Loadfile 파일에서 로드된 데이터를 사용하여 Blob을 만듭니다.
LoadSource TBD
OverrideArgs TBD
OverrideRootSignature TBD
전처리 원본 텍스트를 전처리합니다.
PrimaryOutput TBD
RegisterLibrary 나중에 해당 이름으로 참조할 수 있도록 이름으로 라이브러리를 등록합니다.
RemovePart 컨테이너에서 파트를 제거합니다.
RunOptimizer TBD

SerializeContainer 컨테이너를 빌드합니다.
SetCompiler TBD
SetMalloc IDxcUtils는 IDxcLibrary를 대체합니다. 대신 IDxcUtils를 사용합니다.
유효성 검사 셰이더의 유효성을 검사합니다.
ValidateWithDebug 선택적 디버그 비트코드를 사용하여 셰이더의 유효성을 검사합니다.

구조체

D3D_SHADER_DATA 셰이더 데이터를 설명합니다. (D3D_SHADER_DATA)
D3D_SHADER_DATA 셰이더 데이터를 설명합니다. (D3D_SHADER_DATA)
DxcArgPair TBD
DxcDefine Dxc API에 정의를 제공하기 위한 구조체입니다.
DxcShaderHash ShaderHash에 대한 해시 다이제스트 형식입니다.
DxcText Dxc API에 바이트 또는 텍스트 입력을 제공하는 구조입니다.

IDxcAssembler DxcAssembler를 나타내는 인터페이스입니다.
IDxcBlob DXC API로 전달 및 외부로 전달할 수 있는 크기의 버퍼입니다. **IDxcBlob** 은 **ID3D10Blob** 및 **ID3DBlob**의 별칭입니다.
IDxcBlobEncoding 알려진 인코딩이 있을 수 있는 Blob입니다.
IDxcBlobUtf8 TBD
IDxcCompiler IDxcCompiler는 더 이상 사용되지 않습니다. 대신 IDxcCompiler3 을 사용합니다.
IDxcCompiler2 IDxcCompiler2는 더 이상 사용되지 않습니다. 대신 IDxcCompiler3 을 사용합니다.
IDxcCompiler3 DirectX 셰이더 컴파일러를 나타내는 인터페이스입니다.
IDxcCompilerArgs DXC에 전달된 인수를 관리하기 위한 인터페이스입니다.
IDxcContainerBuilder DXC 컨테이너 작성기를 나타내는 인터페이스입니다.
IDxcContainerReflection DxcContainerReflection을 나타내는 인터페이스입니다.
IDxcExtraOutputs DXC 작업의 추가 출력을 나타냅니다.
IDxcIncludeHandler 포함 지시문을 처리하기 위한 인터페이스입니다. include 지시문의 처리를 사용자 지정하려면 이 인터페이스의 구현을 제공할 수 있습니다.

IDxcLibrary IDxcUtils는 IDxcLibrary를 대체합니다. 대신 IDxcUtils를 사용합니다.
IDxcLinker DXC 링커 인터페이스입니다.
IDxcOperationResult DXC 작업의 결과입니다. IDxcResult는 IDxcOperationResult를 대체하며 가능한 경우 사용해야 합니다.
IDxcOptimizer DxcOptimizer를 나타내는 인터페이스입니다.
IDxcOptimizerPass 최적화 프로그램 패스를 나타내는 인터페이스입니다.
IDxcPdbUtils PDB 버전 정보를 나타냅니다.
IDxcResult DXC 작업의 결과를 나타냅니다. DXC 작업에는 셰이더 개체 및 오류와 같은 여러 출력이 있을 수 있습니다. 이 인터페이스는 출력에 대한 액세스를 제공합니다.
IDxcUtils DXC에 대한 다양한 유틸리티 함수입니다.
IDxcValidator DXC 셰이더 유효성 검사기를 나타내는 인터페이스입니다.
IDxcValidator2 DXC 셰이더 유효성 검사기를 나타내는 인터페이스입니다.
IDxcVersionInfo PDB 버전 정보를 나타냅니다.
IDxcVersionInfo2 PDB 버전 정보를 나타냅니다.

IDxcVersionInfo3 PDB 버전 정보를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

HBA(호스트 버스 어댑터)

아티클2023. 03. 09.

HBA(호스트 버스 어댑터) 기술 개요입니다.

HBA(호스트 버스 어댑터)를 개발하려면 다음 헤더가 필요합니다.

- hbaapi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- HBA(호스트 버스 어댑터)

열거형

HBA_FCPBINDINGTYPE
HBA_WWNTYPE

함수

HBA_CloseAdapter
HBA_FreeLibrary
HBA_GetAdapterAttributes
HBA_GetAdapterName
HBA_GetAdapterPortAttributes
HBA_GetBindingCapability

HBA_GetBindingSupport
HBA_GetDiscoveredPortAttributes
HBA_GetEventBuffer
HBA_GetFC4Statistics
HBA_GetFcpPersistentBinding
HBA_GetFCPStatistics
HBA_GetFcpTargetMapping
HBA_GetFcpTargetMappingV2
HBA_GetNumberOfAdapters
HBA_GetPersistentBindingV2
HBA_GetPortAttributesByWWN
HBA_GetPortStatistics
HBA_GetRNIDMgmtInfo
HBA_GetSBStatistics
HBA_GetSBTargetMapping
HBA_GetVendorLibraryAttributes
HBA_GetVersion

HBA_GetWrapperLibraryAttributes
HBA_LoadLibrary
HBA_OpenAdapter HBA(호스트 버스 어댑터)를 열고 핸들을 반환합니다.
HBA_OpenAdapterByWWN
HBA_RefreshAdapterConfiguration
HBA_RefreshInformation
HBA_RegisterForAdapterAddEvents 새 어댑터가 시스템에 추가될 때 호출하도록 표시된 사용자 콜백 루틴을 등록합니다.
HBA_RegisterForAdapterEvents 어댑터 이벤트에 대해 알 수 있는 사용자 루틴을 선언합니다.
HBA_RegisterForAdapterPortEvents 포트 이벤트가 발생할 때 호출하도록 표시된 사용자 콜백 루틴을 등록합니다.
HBA_RegisterForAdapterPortStatEvents 포트 통계 이벤트가 발생할 때 호출하도록 표시된 사용자 콜백 루틴을 등록합니다.
HBA_RegisterForLinkEvents
HBA_RegisterForTargetEvents 지정된 대상 또는 어댑터와 연결된 모든 대상을 사용하여 대상 이벤트를 등록합니다.
HBA_RegisterLibrary
HBA_RegisterLibraryV2

HBA_RemoveAllPersistentBindings
HBA_RemoveCallback 콜백 루틴을 등록 해제합니다.
HBA_RemovePersistentBinding
HBA_ResetStatistics
HBA_SBDskGetCapacity
HBA_ScsiInquiryV2
HBA_ScsiReadCapacityV2
HBA_ScsiReportLUNsV2
HBA_SendCTPassThru
HBA_SendCTPassThruV2
HBA_SendLIRR
HBA_SendReadCapacity
HBA_SendReportLUNs
HBA_SendRLS
HBA_SendRNID
HBA_SendRNIDV2

HBA_SendRPL
HBA_SendRPS
HBA_SendScsiInquiry
HBA_SendSRL
HBA_SetBindingSupport
HBA_SetPersistentBindingV2
HBA_SetRNIDMgmtInfo
SMHBA_GetAdapterAttributes
SMHBA_GetAdapterPortAttributes
SMHBA_GetBindingCapability
SMHBA_GetBindingSupport
SMHBA_GetDiscoveredPortAttributes
SMHBA_GetFCPhyAttributes
SMHBA_GetLUNStatistics
SMHBA_GetNumberOfPorts
SMHBA_GetPersistentBinding
SMHBA_GetPhyStatistics

SMHBA_GetPortAttributesByWWN
SMHBA_GetPortType
SMHBA_GetProtocolStatistics
SMHBA_GetSASPhyAttributes
SMHBA_GetTargetMapping
SMHBA_GetVendorLibraryAttributes
SMHBA_GetVersion
SMHBA_GetWrapperLibraryAttributes
SMHBA_RegisterForAdapterAddEvents
SMHBA_RegisterForAdapterEvents
SMHBA_RegisterForAdapterPhyStatEvents
SMHBA_RegisterForAdapterPortEvents
SMHBA_RegisterForAdapterPortStatEvents
SMHBA_RegisterForTargetEvents
SMHBA_RegisterLibrary
SMHBA_RemoveAllPersistentBindings
SMHBA_RemovePersistentBinding

SMHBA_ScsiInquiry
SMHBA_ScsiReadCapacity
SMHBA_ScsiReportLuns
SMHBA_SendECHO
SMHBA_SendSMPPassThru
SMHBA_SendTEST
SMHBA_SetBindingSupport
SMHBA_SetPersistentBinding

구조체

HBA_ADAPTERATTRIBUTES
HBA_DEVICESELFDESC
HBA_EVENTINFO
HBA_FC4STATISTICS
HBA_FC4TYPES
HBA_FCPBINDING
HBA_FCPBINDING2

HBA_FCPBINDINGENTRY
HBA_FCPBINDINGENTRY2
HBA_FCPID
HBA_FCPSCSIENTRY
HBA_FCPSCSIENTRYV2
HBA_FCPTARGETMAPPING
HBA_FCPTARGETMAPPINGV2
HBA_IPADDRESS
HBA_LIBRARYATTRIBUTES
HBA_LINK_EVENTINFO
HBA_LUID
HBA_MGMTINFO
HBA_NED
HBA_PORTATTRIBUTES
HBA_PORTSTATISTICS
HBA_PTY_EVENTINFO
HBA_RSCN_EVENTINFO

HBA_SBDEVID
HBA_SBDSKCAPACITY
HBA_SCSIID
HBA_WWN
SMHBA_ADAPTERATTRIBUTES
SMHBA_BINDING
SMHBA_BINDINGENTRY
SMHBA_FC_PHY
SMHBA_FC_PORT
SMHBA_LIBRARYATTRIBUTES
SMHBA_LUID
SMHBA_PHYSTATISTICS
SMHBA_PORT
SMHBA_PORTATTRIBUTES
SMHBA_PORTLUN
SMHBA_PROTOCOLSTATISTICS
SMHBA_SAS_PHY

SMHBA_SAS_PORT
SMHBA_SASPHYSTATISTICS
SMHBA_SCSIENTRY
SMHBA_SCSIID
SMHBA_SCSILUN
SMHBA_TARGETMAPPING

HTTP Server API(HTTP 서버 API)

아티클2023. 03. 16.

HTTP Server API 기술 개요입니다.

HTTP 서버 API를 개발하려면 다음 헤더가 필요합니다.

- http.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- HTTP Server API(HTTP 서버 API)

열거형

HTTP_503_RESPONSE_VERBOSITY HTTP_503_RESPONSE_VERBOSITY 열거형은 503, 서비스를 사용할 수 없음, 오류 응답에 대한 세부 정보 표시 수준을 정의합니다. 이 구조는 요청 큐에서 HttpServer503ResponseProperty를 설정하거나 쿼리할 때 사용해야 합니다.
HTTP_AUTH_STATUS 요청의 인증 상태를 정의합니다.
HTTP_AUTHENTICATION_HARDENING_LEVELS 서버 강화 수준입니다.
HTTP_CACHE_POLICY_TYPE HTTP_CACHE_POLICY_TYPE 열거형 유형은 사용 가능한 캐시 정책을 정의합니다.
HTTP_DATA_CHUNK_TYPE 데이터 청크의 데이터 원본을 정의합니다.
HTTP_DELEGATE_REQUEST_PROPERTY_ID 대리자 요청에 대한 속성 정보를 설명합니다.
HTTP_ENABLED_STATE 요청 큐, 서버 세션 또는 URL 그룹의 상태를 정의합니다.

HTTP_FEATURE_ID HTTP 기능에 대한 식별자를 지정하는 상수를 정의합니다.
HTTP_HEADER_ID HTTP_HEADER_ID 열거형 형식은 HTTP 요청 및 응답에 대해 알려진 헤더를 나열하고 배열 인덱스를 각 헤더와 연결합니다.
HTTP_LOG_DATA_TYPE 로그 데이터의 형식을 식별합니다.
HTTP_LOGGING_ROLLOVER_TYPE 로그 파일 롤오버 형식을 정의합니다.
HTTP_LOGGING_TYPE 수행되는 로깅 유형을 정의합니다.
HTTP_QOS_SETTING_TYPE HTTP_QOS_SETTING_INFO 구조에 포함된 QOS 설정의 형식을 식별합니다.
HTTP_REQUEST_AUTH_TYPE HTTP_REQUEST_AUTH_TYPE 열거형은 HTTP 서버 API에서 지원하는 인증 유형을 정의합니다. 이 열거형은 HTTP_REQUEST_AUTH_INFO 구조체에서 사용됩니다.
HTTP_REQUEST_INFO_TYPE HTTP_REQUEST_INFO_TYPE 열거형은 HTTP_REQUEST_INFO 구조에 포함된 정보의 형식을 정의합니다. 이 열거형은 HTTP_REQUEST_INFO 구조체에서 사용됩니다.
HTTP_REQUEST_PROPERTY 요청에 대해 HTTP 서버 API에 의해 구성된 속성을 정의합니다.
HTTP_RESPONSE_INFO_TYPE HTTP_RESPONSE_INFO_TYPE 열거형은 HTTP_RESPONSE_INFO 구조에 포함된 정보의 형식을 정의합니다. 이 열거형은 HTTP_RESPONSE_INFO 구조체에서 사용됩니다.
HTTP_SERVER_PROPERTY URL 그룹, 서버 세션 또는 요청 큐에서 HTTP 서버 API에 의해 구성된 속성을 정의합니다.

HTTP_SERVICE_BINDING_TYPE HTTP_SERVICE_BINDING_TYPE 열거형 형식은 서비스 이름에 대한 문자열 형식을 지정합니다.
HTTP_SERVICE_CONFIG_CACHE_KEY HttpSetServiceConfiguration 및 HttpQueryServiceConfiguration 함수에 사용됩니다.
HTTP_SERVICE_CONFIG_ID 서비스 구성 옵션을 정의합니다.
HTTP_SERVICE_CONFIG_QUERY_TYPE HTTP_SERVICE_CONFIG_QUERY_TYPE 열거형 형식은 수행할 다양한 유형의 쿼리를 정의합니다. HTTP_SERVICE_CONFIG_SSL_QUERY, HTTP_SERVICE_CONFIG_SSL_CCS_QUERY 및 HTTP_SERVICE_CONFIG_URLACL_QUERY 구조에 사용됩니다.
HTTP_SERVICE_CONFIG_TIMEOUT_KEY HTTP_SERVICE_CONFIG_TIMEOUT_KEY 열거형은 HTTP_SERVICE_CONFIG_TIMEOUT_SET 구조를 통해 쿼리되거나 구성된 타이머 유형을 정의합니다.
HTTP_VERB HTTP_VERB 열거형 형식은 HTTP_REQUEST 구조체에서 알려진 표준 HTTP 동사를 지정하는 데 사용되는 값을 정의합니다. 이러한 알려진 동사의 대부분은 아래 표시된 대로 RFC 2616 및 RFC 2518에 설명되어 있습니다.

Functions

HTTP_EQUAL_VERSION HTTP_VERSION 구조체가 지정된 주/부 버전 조합과 같거나 그렇지 않으면 0이 아닌 값을 반환합니다.
HTTP_GREATER_EQUAL_VERSION HTTP_GREATER_EQUAL_VERSION 매크로는 HTTP_VERSION 구조체가 지정된 주/부 버전 조합보다 크거나 같거나 0이 아니면 0이 아닌 값을 반환합니다.

HTTP_GREATER_VERSION HTTP_VERSION 구조체가 지정된 주/부 버전 조합보다 크거나 그렇지 않으면 0이 아닌 값을 반환합니다.
HTTP_IS_NULL_ID HTTP_OPAQUE_ID NULL인지 확인합니다.
HTTP_LESS_EQUAL_VERSION HTTP_VERSION 구조체가 지정된 주/부 버전 조합보다 작거나 같거나 0이 아니면 0이 아닌 값을 반환합니다.
HTTP_LESS_VERSION HTTP_VERSION 구조체가 지정된 주/부 버전 조합보다 작거나 그렇지 않으면 0이 아닌 값을 반환합니다.
HTTP_NOT_EQUAL_VERSION HTTP_VERSION 구조체가 지정된 주/부 버전 조합보다 크거나 작으면 0이 아닌 값을 반환하고, 0이 같으면 0이 아닌 값을 반환합니다.
HTTP_SET_NULL_ID HTTP_SET_NULL_ID 매크로는 HTTP_OPAQUE_ID NULL로 설정합니다.
HTTP_SET_VERSION 지정된 HTTP_VERSION 구조를 지정된 주/부 버전 조합으로 설정합니다.
HttpAddFragmentToCache HttpAddFragmentToCache 함수는 검색할 수 있는 지정된 이름의 데이터 조각을 캐시하거나 지정된 이름으로 캐시된 데이터를 업데이트합니다.
HttpAddUrl 지정된 URL과 일치하는 요청이 지정된 HTTP Server API 요청 큐로 라우팅되도록 지정된 URL을 등록합니다.
HttpAddUrlToUrlGroup URL 그룹 ID로 식별된 URL 그룹에 지정된 URL을 추가합니다.

HTTPAPI_EQUAL_VERSION HTTPAPI_VERSION 구조체가 지정된 주/부 버전 조합과 정확히 같거나 그렇지 않으면 0이 아닌 값을 반환합니다.
HTTPAPI_GREATER_VERSION HTTPAPI_VERSION 구조체가 지정된 주/부 버전 조합보다 크거나 그렇지 않으면 0이 아닌 값을 반환합니다.
HTTPAPI_LESS_VERSION HTTPAPI_VERSION 구조체가 지정된 주/부 버전 조합보다 작거나 그렇지 않으면 0이 아닌 값을 반환합니다.
HTTPAPI_VERSION_GREATER_OR_EQUAL HTTPAPI_VERSION_GREATER_OR_EQUAL HTTPAPI_VERSION 구조체가 지정된 주/부 버전 조합보다 크거나 같거나 0이 아니면 0이 아닌 값을 반환합니다.
HttpCancelHttpRequest HttpCancelHttpRequest 함수는 지정된 reqest를 취소합니다.
HttpCloseRequestQueue HttpCreateRequestQueue에서 만든 지정된 요청 큐에 대한 핸들을 닫습니다.
HttpCloseServerSession 서버 세션 ID로 식별되는 서버 세션을 삭제합니다.
HttpCloseUrlGroup URL 그룹 ID로 식별된 URL 그룹을 닫습니다.
HttpCreateHttpHandle 호출 애플리케이션에 대한 HTTP 요청 큐를 만들고 해당 애플리케이션에 대한 핸들을 반환합니다.
HttpCreateRequestQueue 새 요청 큐를 만들거나 기존 요청 큐를 엽니다.
HttpCreateServerSession 지정된 버전에 대한 서버 세션을 만듭니다.

HttpCreateUrlGroup 지정된 서버 세션 아래에 URL 그룹을 만듭니다.
HttpDeclarePush HTTP 서버 푸시에 사용할 리소스-하위 리소스 관계를 선언합니다. 그런 다음 HTTP.sys 기본 프로토콜, 연결, 클라이언트 및 정책에서 푸시 작업을 허용하는 경우 지정된 리소스에 대해 HTTP 2.0 서버 푸시를 수행합니다.
HttpDelegateRequestEx 원본 요청 큐에서 대상 요청 큐로 요청을 위임합니다.
HttpDeleteServiceConfiguration HTTP 서버 API 구성 저장소에서 IP 주소 또는 SSL 인증서와 같은 지정된 데이터를 한 번에 하나의 레코드로 삭제합니다.
HttpFindUrlGroupId URL 및 요청 큐에 대한 URL 그룹 ID를 검색합니다.
HttpFlushResponseCache 지정된 요청 큐와 연결된 HTTP Server API 캐시에서 사이트 부분이 지정된 UrlPrefix와 일치하는 이름을 가진 모든 응답 조각을 제거합니다.
HttpInitialize HttpInitialize 함수는 HTTP 서버 API 드라이버를 초기화하고, 아직 시작하지 않은 경우 시작하고, 응답 큐 만들기 및 기타 작업을 지원하기 위해 호출 애플리케이션에 대한 데이터 구조를 할당합니다.
HttpIsFeatureSupported 특정 기능이 지원되는지 여부를 확인합니다.
HttpPrepareUrl 다른 HTTP 함수에서 안전하고 사용할 수 있도록 정규화되지 않은 유니코드 또는 punycode URL을 구문 분석, 분석 및 정규화합니다.
HttpQueryRequestQueueProperty 지정된 핸들로 식별된 요청 큐의 속성을 쿼리합니다.
HttpQueryServerSessionProperty 지정된 서버 세션에서 서버 속성을 쿼리합니다.

HttpQueryServiceConfiguration 하나 이상의 HTTP 서버 API 구성 레코드를 검색합니다.
HttpQueryUrlGroupProperty 지정된 URL 그룹의 속성을 쿼리합니다.
HttpReadFragmentFromCache HttpReadFragmentFromCache 함수는 HTTP 서버 API 캐시에서 지정된 이름을 가진 응답 조각을 검색합니다.
HttpReceiveClientCertificate HttpReceiveClientCertificate 함수는 서버 애플리케이션에서 클라이언트 SSL 인증서 또는 CBT(채널 바인딩 토큰)를 검색하는 데 사용됩니다.
HttpReceiveHttpRequest 지정된 요청 큐에서 동기적으로 또는 비동기적으로 사용 가능한 다음 HTTP 요청을 검색합니다.
HttpReceiveRequestEntityBody 지정된 HTTP 요청에 대한 추가 엔터티 본문 데이터를 받습니다.
HttpRemoveUrl 시스템에서 지정된 UrlPrefix 문자열과 일치하는 요청을 지정된 요청 큐에 라우팅하지 않도록 합니다.
HttpRemoveUrlFromUrlGroup URL 그룹 ID로 식별된 그룹에서 지정된 URL을 제거합니다.
HttpSendHttpResponse 지정된 HTTP 요청에 HTTP 응답을 보냅니다.
HttpSendResponseEntityBody HTTP 응답과 연결된 엔터티 본문 데이터를 보냅니다.
HttpSetRequestProperty 새 속성을 설정하거나 지정된 요청에서 기존 속성을 수정합니다.

HttpSetRequestQueueProperty 새 속성을 설정하거나 지정된 핸들로 식별된 요청 큐에서 기존 속성을 수정합니다.
HttpSetServerSessionProperty 새 서버 세션 속성을 설정하거나 지정된 서버 세션에서 기존 속성을 수정합니다.
HttpSetServiceConfiguration HTTP Server API 구성 저장소에 대한 구성 레코드를 만들고 설정합니다.
HttpSetUrlGroupProperty 새 속성을 설정하거나 지정된 URL 그룹의 기존 속성을 수정합니다.
HttpShutdownRequestQueue 지정된 요청 큐 프로세스에 대한 큐 요청을 중지합니다.
HttpTerminate HTTP Server API에서 애플리케이션의 호출을 처리하는 데 사용하는 리소스를 정리합니다.
HttpUpdateServiceConfiguration 업데이트 HTTP Server API 구성 저장소 내의 구성 레코드에서 TLS(전송 계층 보안) 인증서를 지정하는 서비스 구성 매개 변수입니다.
HttpWaitForDemandStart 새 요청 큐 프로세스에서 처리할 수 있는 새 요청이 도착할 때까지 기다립니다.
HttpWaitForDisconnect 어떤 이유로든 HTTP 클라이언트에 대한 연결이 끊어진 경우 애플리케이션에 알합니다.
HttpWaitForDisconnectEx 이 함수는 HttpWaitForDisconnect에 대한 확장입니다.

구조체

HTTP_BANDWIDTH_LIMIT_INFO HTTP_BANDWIDTH_LIMIT_INFO 구조체는 대역폭 제한 제한을 설정하거나 쿼리하는 데 사용됩니다. 이 구조는 URL 그룹 또는 서버 세션에서 HttpServerBandwidthProperty를 설정하거나 쿼리할 때 사용해야 합니다.
HTTP_BINDING_INFO URL 그룹을 요청 큐와 연결하는 데 사용됩니다.
HTTP_BYTE_RANGE HTTP_BYTE_RANGE 구조는 캐시된 응답 조각, 파일 또는 기타 데이터 블록 내에서 바이트 범위를 지정하는 데 사용됩니다.
HTTP_CACHE_POLICY 캐시된 응답 조각과 연결된 캐시 정책을 정의하는 데 사용됩니다.
HTTP_CHANNEL_BIND_INFO HTTP_CHANNEL_BIND_INFO.
HTTP_CONNECTION_LIMIT_INFO URL 그룹에 대한 최대 미해결 연결 수 제한을 설정하거나 쿼리하는 데 사용됩니다.
HTTP_COOKED_URL 유효성이 검사된 정식 UTF-16 유니코드로 인코딩된 URL 요청 문자열과 포인터 및 요소 길이를 포함합니다.
HTTP_DATA_CHUNK 메모리, 파일 또는 HTTP Server API 응답 조각 캐시의 개별 데이터 블록을 나타냅니다.
HTTP_DELEGATE_REQUEST_PROPERTY_INFO 요청을 위임할 때 추가 속성 정보를 설명합니다.
HTTP_FLOWRATE_INFO 응답의 전송 속도입니다.
HTTP_KNOWN_HEADER HTTP 요청 또는 HTTP 응답의 알려진 헤더에 대한 헤더 값을 포함합니다.

HTTP_LISTEN_ENDPOINT_INFO IP 기반 URL이 특정 IP 주소 또는 와일드카드에서 수신 대기해야 하는지 여부를 제어합니다.
HTTP_LOG_DATA 로그 데이터의 형식을 지정하는 값을 포함합니다.
HTTP_LOG_FIELDS_DATA WC3 로깅을 사용할 때 HTTP 응답에 대해 기록되는 필드를 전달하는 데 사용됩니다.
HTTP_LOGGING_INFO URL 그룹 또는 서버 세션에서 서버 쪽 로깅을 사용하도록 설정하는 데 사용됩니다.
HTTP_MULTIPLE_KNOWN_HEADERS 둘 이상의 헤더가 필요한 경우 HTTP 응답에 포함된 헤더를 지정합니다.
HTTP_PROPERTY_FLAGS 속성 구성을 설정할 때 구성 개체에서 속성을 사용하거나 사용하지 않도록 설정하기 위해 속성 구성 구조에서 사용됩니다.
HTTP_QOS_SETTING_INFO QOS 설정에 대한 정보를 포함합니다.
HTTP_REQUEST_AUTH_INFO 수신 프로세스가 인증된 클라이언트를 가장하는 데 사용할 수 있는 클라이언트 토큰에 대한 핸들이 있는 요청의 인증 상태를 포함합니다.
HTTP_REQUEST_CHANNEL_BIND_STATUS HTTP_REQUEST_CHANNEL_BIND_STATUS.
HTTP_REQUEST_HEADERS HTTP 요청과 함께 전송된 헤더를 포함합니다.
HTTP_REQUEST_INFO 요청에 대한 추가 정보를 사용하여 HTTP_REQUEST 구조를 확장합니다.

HTTP_REQUEST_PROPERTY_STREAM_ERROR HTTP_REQUEST_PROPERTY_STREAM_ERROR 구조체는 HTTP/2 또는 HTTP/3 스트림 오류 코드를 나타냅니다.
HTTP_REQUEST_V1 HTTP_REQUEST 구조를 사용하여 특정 요청과 연결된 데이터를 반환합니다.
HTTP_REQUEST_V2 요청에 대한 자세한 정보를 사용하여 HTTP_REQUEST_V1 요청 구조를 확장합니다.
HTTP_RESPONSE_HEADERS HTTP 응답과 함께 전송된 헤더를 포함합니다.
HTTP_RESPONSE_INFO 응답에 대한 추가 정보를 사용하여 HTTP_RESPONSE 구조를 확장합니다.
HTTP_RESPONSE_V1 HTTP 응답과 연결된 데이터를 포함합니다.
HTTP_RESPONSE_V2 응답에 대한 자세한 정보를 사용하여 HTTP 버전 1.0 응답 구조를 확장합니다.
HTTP_SERVER_AUTHENTICATION_BASIC_PARAMS HTTP_SERVER_AUTHENTICATION_BASIC_PARAMS 구조에는 URL 그룹의 기본 인증에 대한 정보가 포함되어 있습니다. 이 구조체는 HTTP_SERVER_AUTHENTICATION_INFO 구조에 포함되어 있습니다.
HTTP_SERVER_AUTHENTICATION_DIGEST_PARAMS URL 그룹의 다이제스트 인증에 대한 정보를 포함합니다.
HTTP_SERVER_AUTHENTICATION_INFO URL 그룹 또는 서버 세션에서 서버 쪽 인증을 사용하도록 설정하는 데 사용됩니다.
HTTP_SERVICE_BINDING_A HTTP_SERVICE_BINDING_A.

HTTP_SERVICE_BINDING_BASE HTTP_SERVICE_BINDING_BASE.
HTTP_SERVICE_BINDING_W HTTP_SERVICE_BINDING_W.
HTTP_SERVICE_CONFIG_CACHE_SET HttpSetServiceConfiguration 함수의 pConfigInformation 매개 변수에 사용됩니다.
HTTP_SERVICE_CONFIG_IP_LISTEN_PARAM HTTP 서비스가 바인딩하는 IP 주소 목록에서 추가하거나 삭제할 IP 주소를 지정하는 데 사용됩니다.
HTTP_SERVICE_CONFIG_IP_LISTEN_QUERY HttpQueryServiceConfiguration에서 HTTP 서비스가 바인딩하는 IP(인터넷 프로토콜) 주소 목록을 반환하는 데 사용됩니다.
HTTP_SERVICE_CONFIG_SSL_CCS_KEY 포트가 TLS(전송 계층 보안) 핸드셰이크를 수신하는 경우 인증서를 찾기 위해 CCS(중앙 인증서 저장소) 저장소를 참조해야 Http.sys 지정하는 SSL 인증서 레코드를 식별하는 키 역할을 합니다.
HTTP_SERVICE_CONFIG_SSL_CCS_QUERY HttpQueryServiceConfiguration 함수를 호출할 때 포트에서 SSL CCS(중앙 집중식 인증서 저장소) 레코드를 쿼리할 SSL(Secure Sockets Layer) 구성을 지정합니다.
HTTP_SERVICE_CONFIG_SSL_CCS_SET 포트가 TLS(전송 계층 보안) 핸드셰이크를 수신하는 경우 인증서를 찾기 위해 CCS(중앙 인증서 저장소) 저장소를 참조하도록 Http.sys 지정하는 SSL 인증서 레코드를 나타냅니다.
HTTP_SERVICE_CONFIG_SSL_KEY 지정된 SSL(Secure Sockets Layer) 인증서 레코드가 식별되는 키 역할을 합니다.
HTTP_SERVICE_CONFIG_SSL_PARAM SSL 구성 저장소에서 레코드를 정의합니다.
HTTP_SERVICE_CONFIG_SSL_QUERY SSL 구성 저장소에서 쿼리할 특정 레코드를 지정하는 데 사용됩니다.

HTTP_SERVICE_CONFIG_SSL_SET SSL 저장소에 새 레코드를 추가하거나 기존 레코드를 검색하는 데 사용됩니다.
HTTP_SERVICE_CONFIG_SSL_SNI_KEY 지정된 SSL(Secure Sockets Layer) SNI(서버 이름 표시) 인증서 레코드가 SSL SNI 저장소에서 식별되는 키 역할을 합니다.
HTTP_SERVICE_CONFIG_SSL_SNI_QUERY SSL SNI 저장소에서 쿼리할 특정 SSL(Secure Sockets Layer) SNI(서버 이름 표시) 인증서 레코드를 지정하는 데 사용됩니다.
HTTP_SERVICE_CONFIG_SSL_SNI_SET 새 SSL(Secure Sockets Layer) SNI(서버 이름 표시) 인증서 레코드를 SSL SNI 저장소에 추가하거나 기존 레코드를 검색하는 데 사용됩니다.
HTTP_SERVICE_CONFIG_TIMEOUT_SET HTTP Server API 와이드 시간 제한 값을 설정하는 데 사용됩니다.
HTTP_SERVICE_CONFIG_URLACL_KEY URL 네임스페이스 예약 저장소에서 특정 예약 레코드를 지정하는 데 사용됩니다.
HTTP_SERVICE_CONFIG_URLACL_PARAM URL 네임스페이스 예약 저장소의 특정 레코드와 연결된 사용 권한을 지정하는 데 사용됩니다.
HTTP_SERVICE_CONFIG_URLACL_QUERY URL 네임스페이스 예약 저장소에서 쿼리할 특정 예약 레코드를 지정하는 데 사용됩니다.
HTTP_SERVICE_CONFIG_URLACL_SET URL 예약 저장소에 새 레코드를 추가하거나 기존 레코드를 검색하는 데 사용됩니다.
HTTP_SSL_CLIENT_CERT_INFO 인증서가 유효한지 여부를 확인하는 데 사용할 수 있는 SSL(Secure Sockets Layer) 클라이언트 인증서에 대한 데이터를 포함합니다.
HTTP_SSL_INFO SSL 핸드셰이크를 통해 얻은 SSL(Secure Sockets Layer)을 사용하는 연결에 대한 데이터를 포함합니다.

HTTP_STATE_INFO 서버 세션 또는 URL 그룹을 사용하거나 사용하지 않도록 설정하는 데 사용됩니다.
HTTP_TIMEOUT_LIMIT_INFO 애플리케이션별 연결 제한 시간을 정의합니다.
HTTP_TRANSPORT_ADDRESS 특정 HTTP 연결에 사용되는 주소(로컬 및 원격)를 지정합니다.
HTTP_UNKNOWN_HEADER 열거형에 이름이 표시되지 않는 HTTP 요청 또는 응답의 헤더에 대한 이름과 값을 포함합니다.
HTTP_VERSION 요청이 필요하거나 응답이 제공하는 HTTP 프로토콜의 버전을 정의합니다.
HTTPAPI_VERSION HTTP 서버 API의 버전을 정의합니다.

휴먼 인터페이스 디바이스 참조

아티클2023. 03. 09.

HID(휴먼 인터페이스 디바이스) 기술 참조 개요입니다.

휴먼 인터페이스 디바이스를 개발하려면 다음 헤더가 필요합니다.

- dinput.h
- dinputd.h
- ntddkbd.h
- ntddmou.h

HID 드라이버 프로그래밍 지침은 다음을 참조하세요.

- HID(휴먼 인터페이스 디바이스) 소개
- WINDOWS 드라이버 HID(휴먼 인터페이스 디바이스) API

IOCTLs

IOCTL_KEYBOARD_QUERY_ATTRIBUTES IOCTL_KEYBOARD_QUERY_ATTRIBUTES 요청은 키보드 특성에 대한 정보를 반환합니다.
IOCTL_KEYBOARD_QUERY_EXTENDED_ATTRIBUTES IOCTL_KEYBOARD_QUERY_EXTENDED_ATTRIBUTES 요청은 확장된 키보드 특성에 대한 정보를 반환합니다.
IOCTL_KEYBOARD_QUERY_INDICATOR_TRANSLATION IOCTL_KEYBOARD_QUERY_INDICATOR_TRANSLATION 요청은 스캔 코드와 키보드 표시기 간의 매핑에 대한 정보를 반환합니다.
IOCTL_KEYBOARD_QUERY_INDICATORS IOCTL_KEYBOARD_QUERY_INDICATORS 요청은 키보드 표시기 정보를 반환합니다.
IOCTL_KEYBOARD_QUERY_TYPEMATIC IOCTL_KEYBOARD_QUERY_TYPEMATIC 요청은 키보드 형식 설정을 반환합니다.
IOCTL_KEYBOARD_SET_INDICATORS IOCTL_KEYBOARD_SET_INDICATORS 요청은 키보드 표시기를 설정합니다.

IOCTL_KEYBOARD_SET_TYPEMATIC IOCTL_KEYBOARD_SET_TYPEMATIC 요청은 키보드 형식 설정을 설정합니다.
IOCTL_MOUSE_QUERY_ATTRIBUTES IOCTL_MOUSE_QUERY_ATTRIBUTES 요청은 마우스 특성에 대한 정보를 반환합니다.

Functions

획득 IDirectInputJoyConfig8::Acquire 메서드는 "조이스틱 구성 모드"를 획득합니다. 한 번에 하나의 애플리케이션만 조이스틱 구성 모드에 있을 수 있습니다. 이 모드를 획득하려는 다른 애플리케이션의 후속 시도는 오류 DIERR_OTHERAPPHASPRIO 수신해야 합니다. 구성 모드를 입력한 후 애플리케이션은 전역 조이스틱 구성 설정을 변경할 수 있습니다. 다른 애플리케이션이 중간에 설정을 변경한 경우 새 설정을 설치하기 전에 애플리케이션에서 기존 설정을 검사 합니다.
AddNewHardware IDirectInputJoyConfig8::AddNewHardware 메서드는 새 입력 디바이스 설치를 통해 사용자를 안내하는 새 하드웨어 추가 대화 상자를 표시합니다.
AddRef IDirectInputEffectDriver::AddRef 메서드는 DirectInputEffectDriver 개체의 참조 수를 1씩 늘입니다. 이 메서드는 DirectInputEffectDriver에서 상속된 IUnknown 인터페이스의 일부입니다.
AddRef IDirectInputJoyConfig8::AddRef 메서드는 DirectInputJoyConfig 개체의 참조 수를 1로 늘림합니다. 이 메서드는 DirectInputJoyConfig에서 상속하는 IUnknown 인터페이스의 일부입니다.
DeleteConfig IDirectInputJoyConfig8::D eleteConfig 메서드는 조이스틱에 대한 구성 정보를 삭제합니다.
DeleteType IDirectInputJoyConfig8::D eleteType 메서드는 조이스틱 형식에 대한 정보를 제거합니다. 이 메서드는 주의해서 사용합니다. 조이스틱이 삭제된 형식을 참조하지 않도록 하는 것은 호출자의 책임입니다.

DestroyEffect IDirectInputEffectDriver::D estroyEffect 메서드는 디바이스에서 효과를 제거합니다. 효과가 재생되면 드라이버가 언로드하기 전에 중지해야 합니다.
DeviceID IDirectInputEffectDriver::D eviceID 메서드는 드라이버에 디바이스의 ID를 보냅니다.
DownloadEffect IDirectInputEffectDriver::D ownloadEffect 메서드는 디바이스에 효과를 보냅니다.
EnumTypes IDirectInputJoyConfig8::EnumTypes 메서드는 DirectInput에서 현재 지원되는 조이스틱 형식을 열거합니다.
Esc IDirectInputEffectDriver::Escape 메서드는 드라이버로 이스케이프됩니다. 이 메서드는 IDirectInputEffect::Escape 또는 IDirectInputDevice::Escape 메서드를 호출하는 애플리케이션에 대한 응답으로 호출됩니다.
GetConfig IDirectInputJoyConfig8::GetConfig 메서드는 조이스틱의 구성에 대한 정보를 가져옵니다.
GetEffectStatus IDirectInputEffectDriver::GetEffectStatus 메서드는 효과의 상태 대한 정보를 가져옵니다.
GetForceFeedbackState IDirectInputEffectDriver::GetForceFeedbackState 메서드는 디바이스에 대한 힘 피드백 상태를 검색합니다.
GetTypeInfo IDirectInputJoyConfig8::GetTypeInfo 메서드는 조이스틱 형식에 대한 정보를 가져옵니다.
GetUserValues IDirectInputJoyConfig8::GetUserValues 메서드는 조이스틱에 대한 사용자 설정에 대한 정보를 가져옵니다.

GetVersions IDirectInputEffectDriver::GetVersions 메서드는 힘 피드백 하드웨어 및 드라이버에 대한 버전 정보를 가져옵니다.
OpenAppStatusKey IDirectInputJoyConfig8::OpenAppStatusKey 메서드는 애플리케이션 상태 레지스트리 키의 루트 키를 열고 키에 대한 핸들을 반환 매개 변수로 가져옵니다.
OpenTypeKey IDirectInputJoyConfig8::OpenTypeKey 메서드는 조이스틱 형식과 연결된 레지스트리 키를 엽니다.
QueryInterface IDirectInputEffectDriver::QueryInterface 메서드는 DirectInputEffectDriver 개체가 특정 COM 인터페이스를 지원하는지 여부를 결정합니다.
QueryInterface IDirectInputJoyConfig8::QueryInterface 메서드는 DirectInputJoyConfig 개체가 특정 COM 인터페이스를 지원하는지 여부를 결정합니다.
릴리스 IDirectInputEffectDriver::Release 메서드는 DirectInputEffectDriver 개체의 참조 수를 1씩 줄입니다. 이 메서드는 DirectInputEffectDriver에서 상속된 IUnknown 인터페이스의 일부입니다.
릴리스 IDirectInputJoyConfig8::Release 메서드는 DirectInputJoyConfig 개체의 참조 수를 1로 줄입니다. 이 메서드는 DirectInputJoyConfig에서 상속하는 IUnknown 인터페이스의 일부입니다.
SendForceFeedbackCommand IDirectInputEffectDriver::SendForceFeedbackCommand 메서드는 디바이스의 힘 피드백 상태를 변경합니다.
SendNotify IDirectInputJoyConfig8::SendNotify 메서드는 디바이스 드라이버 및 애플리케이션에 디바이스 구성이 변경되었음을 알 수 있습니다.
SetConfig IDirectInputJoyConfig8::SetConfig 메서드는 조이스틱에 대한 구성 정보를 만들거나 재정의합니다.

SetCooperativeLevel IDirectInputJoyConfig8::SetCooperativeLevel 메서드는 디바이스의 instance 대한 협력 수준을 설정합니다. IDirectInputJoyConfig8 인터페이스에 지원되는 유일한 협력 수준은 DISCL_EXCLUSIVE 및 DISCL_BACKGROUND.
SetGain IDirectInputEffectDriver::SetGain 메서드는 전체 디바이스 게인을 설정합니다.
SetTypeInfo IDirectInputJoyConfig8::SetTypeInfo 메서드는 새 조이스틱 형식을 만들거나 기존 조이스틱 형식에 대한 정보를 다시 정의합니다.
SetUserValues IDirectInputJoyConfig8::SetUserValues 메서드는 조이스틱에 대한 사용자 설정을 설정합니다.
StartEffect IDirectInputEffectDriver::StartEffect 메서드는 효과 재생을 시작합니다. 효과가 이미 재생 중인 경우 처음부터 다시 시작됩니다.
StopEffect IDirectInputEffectDriver::StopEffect 메서드는 효과 재생을 중지합니다.
필수 조건 취소 IDirectInputJoyConfig8::Unacquire 메서드는 "조이스틱 구성 모드"를 선택 취소합니다.

인터페이스

IDirectInputEffectDriver 이러한 세 가지 메서드를 사용하면 원래 인터페이스의 기능에 영향을 주지 않고 DirectInputEffectDriver 개체에 추가 인터페이스를 추가할 수 있습니다.
IDirectInputJoyConfig8 IDirectInputJoyConfig8 인터페이스에는 속성 시트를 작성하는 하드웨어 개발자가 레지스트리에 정보를 쓰고 읽을 수 있는 메서드가 포함되어 있습니다.

구조체

DIDEVICESTATE DIDEVICESTATE 구조체는 힘 피드백 디바이스의 상태에 대한 정보를 반환합니다.
디드리버버전 DIDRIVERVERSIONS 구조체는 DirectInput 효과 드라이버에서 버전 정보를 DirectInput에 다시 보고하는 데 사용됩니다.
DIEFFECTATTRIBUTES DIEFFECTATTRIBUTES 구조체는 힘 피드백 디바이스에서 지원하는 각 효과에 대한 레지스트리 키의 "특성" 값에 포함된 정보를 설명합니다.
DIEFFESCAPE DIEFFESCAPE 구조는 하드웨어 관련 데이터를 디바이스 드라이버에 직접 전달합니다.
DIFFDEVICEATTRIBUTES DIFFDEVICEATTRIBUTES 구조체는 OEMForceFeedback 레지스트리 키의 "특성" 값에 포함된 정보를 설명합니다.
DIFFOBJECTATTRIBUTES DIFFOBJECTATTRIBUTES 구조체는 힘 피드백 디바이스의 각 "개체"에 대한 레지스트리 키의 "FFAttributes" 값에 포함된 정보를 설명합니다.
디히드피니틴포 DIHIDFFINITINFO 구조체는 DirectInput에서 제어하도록 요청하는 디바이스에 대한 HID 힘 피드백 드라이버에 정보를 제공하는 데 사용됩니다.
DIJOYCONFIG DIJOYCONFIG 구조체에는 조이스틱의 구성에 대한 정보가 포함되어 있습니다.
DIJOYTYPEINFO DIJOYTYPEINFO 구조체에는 조이스틱 형식에 대한 정보가 포함되어 있습니다.
DIJOYUSERVALUES DIJOYUSERVALUES 구조체에는 사용자의 조이스틱 설정에 대한 정보가 포함되어 있습니다.

DIOBJECTATTRIBUTES DIOBJECTATTRIBUTES 구조체는 디바이스의 각 "개체"에 대한 레지스트리 키의 "특성" 값에 포함된 정보를 설명합니다. "특성" 값이 없으면 기본 특성이 사용됩니다.
DIOBJECTCALIBRATION DIOBJECTCALIBRATION 구조체는 디바이스의 각 축에 대한 레지스트리 키의 "보정" 값에 포함된 정보를 설명합니다.
JOYREGHWVALUES JOYREGHWVALUES(mmddk.h) 구조체에는 하드웨어에서 반환하는 값 범위(보정으로 채워짐)가 포함됩니다.
JOYREGHWVALUES JOYREGHWVALUES(dinputd.h) 구조체에는 하드웨어에서 반환하는 값 범위(보정으로 채워짐)가 포함됩니다.
KEYBOARD_ATTRIBUTES KEYBOARD_ATTRIBUTES 키보드의 특성을 지정합니다.
KEYBOARD_EXTENDED_ATTRIBUTES KEYBOARD_EXTENDED_ATTRIBUTES 키보드의 확장 특성을 지정합니다.
KEYBOARD_INDICATOR_PARAMETERS KEYBOARD_INDICATOR_PARAMETERS 키보드 표시기 LED의 상태를 지정합니다.
KEYBOARD_INDICATOR_TRANSLATION KEYBOARD_INDICATOR_TRANSLATION 키보드 스캔 코드와 LED 표시기 간의 디바이스별 가변 길이 매핑 배열을 지정합니다.
KEYBOARD_INPUT_DATA KEYBOARD_INPUT_DATA 하나의 키보드 입력 데이터 패킷을 포함합니다.
KEYBOARD_TYPEMATIC_PARAMETERS KEYBOARD_TYPEMATIC_PARAMETERS 키보드의 형식 설정을 지정합니다.
KEYBOARD_UNIT_ID_PARAMETER KEYBOARD_UNIT_ID_PARAMETER Kbdclass가 키보드에 할당하는 단위 ID를 지정합니다.

MOUSE_ATTRIBUTES MOUSE_ATTRIBUTES 마우스 디바이스의 특성을 지정합니다.
MOUSE_INPUT_DATA MOUSE_INPUT_DATA 마우스 입력 데이터의 패킷을 포함합니다.
MOUSE_UNIT_ID_PARAMETER MOUSE_UNIT_ID_PARAMETER Mouclass가 마우스에 할당하는 단위 ID를 지정합니다.

Hyper-V 관리

아티클2023. 08. 22.

Hyper-V 관리 기술 개요입니다.

Hyper-V 관리를 개발하려면 다음 헤더가 필요합니다.

- vmgenerationcounter.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Hyper-V 관리

IOCTLs

IOCTL_VMGENCOUNTER_READ IOCTL_VMGENCOUNTER_READ 제어 코드는 가상 머신 생성 식별자를 검색합니다.

구조체

VM_GENCOUNTER 가상 머신 생성 식별자에 대해 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Hyper-V WMI 공급자

아티클2023. 03. 09.

Hyper-V WMI 공급자 기술 개요입니다.

Hyper-V WMI 공급자를 개발하려면 다음 헤더가 필요합니다.

- mmc.h
- mpconfig.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Hyper-V WMI 공급자

이미지 마스터링 API

아티클2023. 03. 09.

이미지 마스터링 API 기술에 대한 개요입니다.

이미지 마스터링 API를 개발하려면 다음 헤더가 필요합니다.

- imapi.h
- imapi2.h
- imapi2fs.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 이미지 마스터링 API

열거형

EmulationType 부팅 이미지가 에뮬레이트할 미디어 형식에 대한 값을 정의합니다.
FsiFileSystems 인식된 파일 시스템에 대한 값을 정의합니다.
FsiItemType IFileSystemImage::Exists 메서드를 사용하여 발견된 파일 시스템 항목의 값을 정의합니다.
IMAPI_BURN_VERIFICATION_LEVEL IBurnVerification 인터페이스에서 구현된 굽기 확인 값을 정의합니다.
IMAPI_CD_SECTOR_TYPE CD 미디어에 쓸 수 있는 섹터 유형을 정의합니다.
IMAPI_CD_TRACK_DIGITAL_COPY_SETTING 지정된 트랙에 사용할 수 있는 디지털 복사 설정 값을 정의합니다.
IMAPI_FEATURE_PAGE_TYPE 논리 단위(CD 및 DVD 장치)에서 지원하는 기능에 대한 값을 정의합니다.

IMAPI_FORMAT2_DATA_MEDIA_STATE 가능한 미디어 상태에 대한 값을 정의합니다.
IMAPI_FORMAT2_DATA_WRITE_ACTION IDiscFormat2DataEventArgs 인터페이스를 사용할 때 쓰기 작업의 현재 상태를 나타내는 값을 정의합니다.
IMAPI_FORMAT2_RAW_CD_DATA_SECTOR_TYPE 하위 채널 데이터의 형식을 나타내는 값을 정의합니다.
IMAPI_FORMAT2_RAW_CD_WRITE_ACTION IDiscFormat2RawCDEventArgs 인터페이스를 사용할 때 쓰기 작업의 현재 상태를 나타내는 값을 정의합니다.
IMAPI_FORMAT2_TAO_WRITE_ACTION IDiscFormat2TrackAtOnceEventArgs 인터페이스를 사용할 때 쓰기 작업의 현재 상태를 나타내는 값을 정의합니다.
IMAPI_MEDIA_PHYSICAL_TYPE IMAPI에서 지원하는 현재 알려진 미디어 형식에 대한 값을 정의합니다.
IMAPI_MEDIA_WRITE_PROTECT_STATE 미디어 쓰기 보호 상태를 나타내는 값을 정의합니다. 지정된 드라이브에서 하나 이상의 쓰기 보호 값을 설정할 수 있습니다.
IMAPI_MODE_PAGE_REQUEST_TYPE MODE_SENSE10 MMC 명령을 사용하여 디바이스로 전송된 요청을 나타내는 값을 정의합니다.
IMAPI_MODE_PAGE_TYPE CD 및 DVD 디바이스에서 지원하는 모드 페이지에 대한 값을 정의합니다.
IMAPI_PROFILE_TYPE CD 및 DVD 디바이스의 가능한 프로필에 대한 값을 정의합니다. 프로필은 디바이스에서 지원하는 미디어 및 기능의 유형을 정의합니다.
IMAPI_READ_TRACK_ADDRESS_TYPE 임의 쓰기가 가능한 하드웨어 결함 관리 미디어 유형의 현재 디스크 프로필에 대한 추적 주소를 해석하는 방법을 나타내는 값을 정의합니다.

PlatformId 부팅 이미지가 지원하는 운영 체제 아키텍처의 값을 정의합니다.

함수

AcquireExclusiveAccess 디바이스에 대한 단독 액세스를 획득합니다.
추가 IFsiItem 개체에서 설명하는 파일 또는 디렉터리를 파일 시스템 이미지에 추가합니다.
AddAudioTrack 데이터 스트림을 현재 미디어에 새 트랙으로 씁니다.
AddAudioTrackBlocks 오디오 데이터 블록을 현재 열려 있는 트랙에 추가합니다. 이 메서드는 사용 가능한 공간이 없거나 트랙이 가득 찼을 때까지 반복적으로 호출할 수 있습니다.
AddData 스테이징된 이미지 파일에 루트 스토리지의 콘텐츠를 추가합니다. 이 스토리지는 스테이지 이미지 파일의 루트 파일 시스템에 모든 하위 저장소 및 스트림을 배치하도록 열거됩니다. 하위 저장소는 폴더가 되고 스트림은 파일이 됩니다.
AddDirectory 파일 시스템 이미지에 디렉터리를 추가합니다.
AddFile 파일 시스템 이미지에 파일을 추가합니다.
AddSpecialPregap 제공된 IStream 개체를 수락하고 트랙 1의 사전 간격에 대한 데이터로 사용할 연결된 포인터를 저장합니다.
AddStream 명명된 스트림을 파일 시스템 이미지의 특정 파일과 연결합니다.

AddSubcodeRWGenerator IStream에서 제공하는 사용자 지정 R-W 하위 코드를 추가하도록 허용합니다. 제공된 개체의 크기는 최종 이미지를 만들 때 원시 디스크 이미지의 섹터 수 * 96바이트와 같아야 합니다.
AddTrack 제공된 IStream 개체를 수락하고 인터페이스 포인터를 이미지의 다음 트랙으로 저장합니다.
AddTree 디렉터리 트리의 내용을 파일 시스템 이미지에 추가합니다.
AddTreeWithNamedStreams 모든 파일과 연결된 명명된 스트림과 함께 디렉터리 트리의 내용을 파일 시스템 이미지에 추가합니다.
Append 이 스트림에 스트림을 추가합니다.
Append2 이 스트림에 스트림 배열을 추가합니다.
AssignBootImage 부팅 이미지가 포함된 데이터 스트림을 설정합니다.
CalculateDiscIdentifier 디스크 및 디스크에 기록된 세션을 식별하는 문자열을 검색합니다.
CancelAddTrack 현재 쓰기 작업을 취소합니다. (IDiscFormat2TrackAtOnce.CancelAddTrack)
CancelWrite 현재 쓰기 작업을 취소합니다. (IDiscFormat2Data.CancelWrite)
CancelWrite 현재 쓰기 작업을 취소합니다. (IDiscFormat2RawCD.CancelWrite)
CancelWrite 진행 중인 쓰기 작업을 취소합니다.

ChooseImageDefaults 현재 미디어를 기반으로 기본 파일 시스템 형식 및 이미지 크기를 설정합니다.
ChooseImageDefaultsForMediaType 지정된 미디어 형식에 따라 기본 파일 시스템 형식 및 이미지 크기를 설정합니다.
ClearFormatContent 화상을 입지 않은 이미지(현재 스태시 파일)의 내용을 지웁니다.
복제 현재 열거형과 열거형 상태가 같은 다른 열거형을 만듭니다. (IEnumFsiItems.Clone)
복제 현재 열거형과 열거형 상태가 같은 다른 열거형을 만듭니다. (IEnumProgressItems.Clone)
닫기 다른 애플리케이션에서 사용할 수 있도록 인터페이스를 닫습니다.
닫기 디스크 레코더에 대한 단독 액세스를 해제합니다. 이렇게 하면 드라이브에 대한 파일 시스템 액세스가 복원됩니다.
CloseAudioTrack 현재 열려 있는 오디오 트랙을 닫습니다. IDiscMaster::RecordDisc 메서드를 호출하려면 먼저 모든 오디오 트랙을 닫아야 합니다.
CloseTray 미디어 트레이를 닫습니다.
CreateAudioTrack 새 오디오 트랙 스테이징을 시작합니다. 이미지에 열려 있는 오디오 트랙이 없는 경우에만 호출할 수 있습니다.
CreateDirectoryItem 지정된 이름을 사용하여 디렉터리 항목을 만듭니다.

CreateFileItem 지정된 이름을 사용하여 파일 항목을 만듭니다.
CreateResultImage 현재 설정에 따라 최종 IStream 개체를 만듭니다.
CreateResultImage 파일 시스템 및 파일 데이터를 포함하는 결과 개체를 만듭니다.
DisableMcn 디바이스에 대한 MCN(미디어 변경 알림)을 사용하지 않도록 설정합니다.
꺼내기 가능하면 디스크 레코더의 트레이를 잠금 해제하고 내보냅니다.
EjectMedia 디바이스에서 미디어를 배출합니다.
EnableMcn 디바이스에 대한 MCN(미디어 변경 알림)을 사용하도록 설정합니다.
EnumDiscMasterFormats 이 디스크 마스터 개체에서 지원하는 모든 디스크 마스터 형식에 대한 열거자를 검색합니다. 디스크 마스터 형식은 스테이징된 이미지 파일(데이터/오디오)의 콘텐츠 구조와 스테이징된 이미지를 관리하는 인터페이스를 지정합니다.
EnumDiscRecorders 활성 디스크 마스터 형식에서 지원하는 모든 디스크 레코더에 대한 열거자를 검색합니다.
지우기 CD-RW 디스크 레코더인 경우 CD-RW 미디어를 지우려고 시도합니다. 전체 지우기와 빠른 지우기는 모두 지원됩니다.
EraseMedia 활성 디스크 레코더에서 미디어를 지웁니다.

Exists 지정된 파일 또는 디렉터리의 존재 여부를 확인합니다.
FileSystemName 지정된 파일 시스템을 준수하도록 수정된 항목의 이름을 검색합니다.
FileSystemPath 지정된 파일 시스템을 준수하도록 수정된 항목의 전체 경로를 검색합니다.
get__NewEnum 컴퓨터에 설치된 CD 및 DVD 디바이스 목록을 검색합니다.
get__NewEnum 파일 시스템 이미지의 디렉터리에 포함된 자식 항목 목록을 검색합니다. (IFsiDirectoryItem.get__NewEnum)
get__NewEnum 파일 시스템 이미지의 파일과 연결된 명명된 스트림의 IEnumVARIANT 목록을 검색합니다.
get__NewEnum 컬렉션에서 진행률 항목 목록을 검색합니다. (IProgressItems.get__NewEnum)
get_ActiveDiscRecorder 디스크 디바이스를 초기화하는 데 사용되는 고유 식별자를 검색합니다.
get_AudioHasPreemphasis 오디오 트랙에 오디오 데이터에 추가된 추가 사전 강조가 있는지를 지정하는 값을 검색합니다.
get_BlockCount 진행률 항목의 블록 수를 검색합니다.
get_BlockRanges VT_Dispatch 형식의 안전한 변형 배열 형식으로 섹터 범위 목록을 반환합니다.
get_BlockSize 데이터 블록의 크기(바이트)를 검색합니다.

get_BootImage 부팅 이미지 데이터 스트림에 대한 포인터를 검색합니다.
get_BootImageOptions 파일 시스템 이미지에 추가하려는 부팅 이미지를 검색합니다.
get_BootImageOptionsArray 파일 시스템 이미지를 생성하는 데 사용할 부팅 옵션 배열을 검색합니다.
get_BufferUnderrunFreeDisabled CDR, CD-RW 및 DVD-R 미디어에 버퍼 언더런 무료 녹화를 사용할 수 있는지 여부를 확인합니다.
get_BufferUnderrunFreeDisabled 버퍼 언더런 무료 기록을 사용할 수 있는지 여부를 확인합니다. (IDiscFormat2RawCD.get_BufferUnderrunFreeDisabled)
get_BufferUnderrunFreeDisabled 버퍼 언더런 무료 기록을 사용할 수 있는지 여부를 확인합니다. (IDiscFormat2TrackAtOnce.get_BufferUnderrunFreeDisabled)
get_BurnVerificationLevel 현재 번 확인 수준을 검색합니다.
get_BytesPerSector 쓰기 중에 각 섹터에 사용할 바이트 수를 검색합니다. 반환된 값은 이전에 IWriteEngine2::put_BytesPerSector로 설정한 값을 나타내며 미디어의 섹터 값당 현재 바이트를 반환하지 않습니다.
get_ChangePoint 변경 지점 식별자를 검색합니다.
get_ClientName 클라이언트의 식별 이름을 검색합니다. (IDiscFormat2Data.get_ClientName)
get_ClientName 클라이언트의 이름을 검색합니다. (IDiscFormat2Erase.get_ClientName)

get_ClientName 클라이언트의 이름을 검색합니다. (IDiscFormat2RawCD.get_ClientName)
get_ClientName 클라이언트의 이름을 검색합니다. (IDiscFormat2TrackAtOnce.get_ClientName)
get_Count 컴퓨터에 설치된 CD 및 DVD 디스크 디바이스의 수를 검색합니다.
get_Count 열거형의 자식 항목 수입니다.
get_Count 파일 시스템 이미지의 파일과 연결된 명명된 스트림의 수를 반환합니다.
get_Count 컬렉션의 진행률 항목 수를 검색합니다.
get_CreateRedundantUdfMetadataFiles UDF 메타데이터가 파일 시스템 이미지에서 중복되는지를 지정하는 속성 값을 검색합니다.
get_CreationTime 디렉터리 또는 파일 항목이 만들어지고 파일 시스템 이미지에 추가된 날짜 및 시간을 검색합니다.
get_CurrentAction 수행 중인 현재 쓰기 작업을 검색합니다. (IDiscFormat2DataEventArgs.get_CurrentAction)
get_CurrentAction 수행 중인 현재 쓰기 작업을 검색합니다. (IDiscFormat2RawCDEventArgs.get_CurrentAction)
get_CurrentAction 수행 중인 현재 쓰기 작업을 검색합니다. (IDiscFormat2TrackAtOnceEventArgs.get_CurrentAction)
get_CurrentFeaturePages 현재로 표시된 디바이스의 기능 페이지 목록을 검색합니다.

get_CurrentMediaStatus 디바이스에서 미디어의 현재 상태를 검색합니다.
get_CurrentPhysicalMediaType 디스크 디바이스의 미디어 유형을 검색합니다. (IDiscFormat2Data.get_CurrentPhysicalMediaType)
get_CurrentPhysicalMediaType 디스크 디바이스의 미디어 유형을 검색합니다. (IDiscFormat2Erase.get_CurrentPhysicalMediaType)
get_CurrentPhysicalMediaType 디스크 디바이스의 미디어 유형을 검색합니다. (IDiscFormat2RawCD.get_CurrentPhysicalMediaType)
get_CurrentPhysicalMediaType 디스크 디바이스의 미디어 유형을 검색합니다. (IDiscFormat2TrackAtOnce.get_CurrentPhysicalMediaType)
get_CurrentProfiles 현재로 표시된 디바이스의 모든 MMC 프로필을 검색합니다.
get_CurrentRotationTypeIsPureCAV 레코더에서 사용하는 현재 회전 속도 컨트롤을 검색합니다. (IDiscFormat2Data.get_CurrentRotationTypeIsPureCAV)
get_CurrentRotationTypeIsPureCAV 레코더에서 사용하는 현재 회전 속도 컨트롤을 검색합니다. (IDiscFormat2RawCD.get_CurrentRotationTypeIsPureCAV)
get_CurrentRotationTypeIsPureCAV 레코더에서 사용하는 현재 회전 속도 컨트롤을 검색합니다. (IDiscFormat2TrackAtOnce.get_CurrentRotationTypeIsPureCAV)
get_CurrentTrackNumber 미디어에 기록되는 현재 트랙 번호를 검색합니다.
get_CurrentWriteSpeed 드라이브의 현재 쓰기 속도를 검색합니다. (IDiscFormat2Data.get_CurrentWriteSpeed)

get_CurrentWriteSpeed 드라이브의 현재 쓰기 속도를 검색합니다. (IDiscFormat2RawCD.get_CurrentWriteSpeed)
get_CurrentWriteSpeed 드라이브의 현재 쓰기 속도를 검색합니다. (IDiscFormat2TrackAtOnce.get_CurrentWriteSpeed)
get_Data 파일 콘텐츠의 데이터 스트림을 검색합니다.
get_DataSize 파일의 바이트 수를 검색합니다.
get_DataSize32BitHigh IFsiFileItem::get_DataSize 속성의 가장 중요한 32비트를 검색합니다.
get_DataSize32BitLow IFsiFileItem::get_DataSize 속성의 가장 중요한 32비트를 검색합니다.
get_Description 진행률 항목에서 설명을 검색합니다.
get_DeviceCanLoadMedia 디바이스가 미디어를 꺼내서 다시 로드할 수 있는지 여부를 결정합니다.
get_DigitalAudioCopySetting 결과 미디어의 현재 디지털 오디오 복사 설정을 나타내는 비트의 값을 검색합니다. 가능한 값은 IMAPI_CD_TRACK_DIGITAL_COPY_SETTING 열거형을 참조하세요.
get_DirectoryCount 파일 시스템 이미지의 디렉터리 수를 검색합니다.
get_DisableConsumerDvdCompatibilityMode DVD 녹음 세션에 장치가 DVD를 재생할 가능성을 높일 수 있는 작업이 포함되어 있는지 여부를 결정합니다. (Get)

get_DisableGaplessAudio "Gapless Audio" 녹음을 사용할 수 없는지 지정하는 현재 값을 검색합니다. 이 속성은 기본적으로 VARIANT_FALSE 값으로 설정되며 연속 오디오 트랙 간의 "간격 없는" 녹음을 사용하지 않도록 설정합니다.
get_DiscId 이 파일 시스템 이미지의 디스크 볼륨 이름을 검색합니다.
get_DoNotFinalizeMedia 오디오 트랙을 작성한 후 미디어가 쓰기 위해 열려 있는지 여부를 확인합니다. (Get)
get_ElapsedTime 쓰기 작업의 총 경과 시간을 검색합니다. (IDiscFormat2DataEventArgs.get_ElapsedTime)
get_ElapsedTime 쓰기 작업의 총 경과 시간을 검색합니다. (IDiscFormat2RawCDEventArgs.get_ElapsedTime)
get_ElapsedTime 쓰기 작업의 총 경과 시간을 검색합니다. (IDiscFormat2TrackAtOnceEventArgs.get_ElapsedTime)
get_Emulation 부팅 이미지가 에뮬레이트할 미디어 유형을 검색합니다.
get_EndingSectorsPerSecond 기록 디바이스가 쓰기 프로세스가 끝날 때 미디어에 쓸 수 있는 초당 예상 섹터 수를 검색합니다.
get_EndLba IBlockRange 인터페이스에서 지정한 범위의 끝 섹터를 검색합니다.
get_EnumFsiItems 파일 시스템 이미지의 디렉터리에 포함된 자식 항목 목록을 검색합니다. (IFsiDirectoryItem.get_EnumFsiItems)
get_EnumNamedStreams 파일 시스템 이미지의 파일과 연결된 명명된 스트림 컬렉션에 대한 변형이 아닌 열거자를 만듭니다.

get_EnumProgressItems 컬렉션에서 진행률 항목 목록을 검색합니다. (IProgressItems.get_EnumProgressItems)
get_ExclusiveAccessOwner 디바이스에 대한 단독 액세스 권한이 있는 클라이언트 애플리케이션의 이름을 검색합니다.
get_ExpectedTableOfContents 트랙 작성 세션 내에서 미디어에 배치된 오디오 트랙의 목차를 검색합니다.
get_ExpectedTableOfContents 결과 디스크에 대한 SCSI 형식 목차를 가져옵니다.
get_ExtendedSeed 난수 생성기에서 사용하는 시드 값 배열을 검색합니다.
get_FileCount 파일 시스템 이미지의 파일 수를 검색합니다.
get_FileSystemsSupported 클라이언트가 파일 시스템 이미지를 빌드하는 데 사용할 수 있는 파일 시스템 형식 목록을 검색합니다.
get_FileSystemsToCreate 결과 스트림을 생성할 때 만들 파일 시스템의 형식을 검색합니다.
get_FirstBlock 결과 이미지의 이 세그먼트에서 첫 번째 블록 번호를 검색합니다.
get_ForceMediaToBeClosed 파일 시스템에 대한 추가가 방지되는지 여부를 확인합니다. (Get)
get_ForceOverwrite 데이터 작성기가 덮어쓸 수 있는 미디어 형식에서 디스크를 덮어써야 하는지 여부를 결정합니다. (Get)
get_FreeMediaBlocks 이미지에 사용할 수 있는 최대 블록 수를 검색합니다.

get_FreeSectorsOnMedia 기존 데이터를 덮어쓰지 않고 증분 기록을 위해 디스크의 사용 가능한 섹터 수를 검색합니다.
get_FreeSectorsOnMedia 미디어에 새 트랙을 추가하는 데 사용할 수 있는 섹터 수를 검색합니다.
get_FreeSectorsOnMedia 미디어에서 사용할 수 있는 무료 섹터의 수를 검색합니다.
get_FreeSystemBuffer 디스크에 쓰는 데 사용되는 내부 데이터 버퍼에서 사용되지 않는 바이트 수를 검색합니다.
get_FsiNamedStreams 파일 시스템 이미지의 파일과 연결된 명명된 스트림의 컬렉션을 검색합니다.
get_FullErase 디스크 지우기의 품질을 결정합니다. (Get)
get_FullPath 파일 시스템 이미지에서 파일 또는 디렉터리 항목의 전체 경로를 검색합니다.
get_ImageSize 부팅 이미지의 크기를 검색합니다.
get_ImageStream 화상 이미지 스트림을 검색합니다.
get_ImportedVolumeName 가져온 파일 시스템에서 제공된 볼륨 이름을 검색합니다.
get_ImportRecorder 하나 이상의 이전 세션을 가져오는 데 사용할 디스크 레코더를 검색합니다.
get_InUse 이 다중 세션 인터페이스가 현재 미디어에서 사용해야 하는 인터페이스인지 확인합니다. (Get)

get_IsFirstDataSession 이 세션이 미디어의 첫 번째 데이터 세션인지 확인합니다.
get_IsHidden 파일 시스템 이미지에서 항목의 숨겨진 특성이 설정되어 있는지 여부를 확인합니다. (Get)
get_IsNamedStream 항목이 명명된 스트림인지 여부를 확인합니다.
get_ISO9660InterchangeLevel 결과 이미지를 만들 때 사용할 ISO9660 호환성 수준을 검색합니다.
get_ISO9660InterchangeLevelsSupported 지원되는 ISO9660 호환성 수준을 검색합니다.
get_ISRC 현재 트랙과 연결된 ISRC(International Standard Recording Code)를 검색합니다. 이 속성 값은 기본적으로 NULL(또는 길이가 0인 문자열)이며 오디오 데이터를 포함하는 트랙에 대해서만 설정할 수 있습니다.
get_IsRealTime 파일 시스템 이미지의 파일 항목이 '실시간' 또는 표준 파일인지 여부를 지정하는 속성 값을 검색합니다.
get_IsSupportedEnvironment 환경에 하나 이상의 광학 디바이스가 포함되어 있고 실행 컨텍스트에 디바이스에 액세스할 수 있는 권한이 있는지 확인하는 값을 검색합니다.
get_IsSupportedOnCurrentMediaState 다중 세션 유형이 현재 광학 미디어에 쓸 수 있는지 여부를 결정합니다.
get_Item 지정된 디스크 디바이스의 고유 식별자를 검색합니다.
get_Item 파일 시스템 이미지에서 지정된 디렉터리 또는 파일 항목을 검색합니다.

get_Item 파일 시스템 이미지의 파일과 연결된 단일 명명된 스트림을 검색합니다.
get_Item 컬렉션에서 지정된 진행률 항목을 검색합니다.
get_LastAccessedTime 파일 시스템 이미지에서 디렉터리 또는 파일 항목에 마지막으로 액세스한 날짜와 시간을 검색합니다.
get_LastBlock 결과 이미지의 이 세그먼트에서 마지막 블록을 검색합니다.
get_LastModifiedTime 파일 시스템 이미지에서 디렉터리 또는 파일 항목이 마지막으로 수정된 날짜와 시간을 검색합니다.
get_LastPossibleStartOfLeadout 리드아웃 영역에 대해 가능한 마지막 시작 위치를 검색합니다.
get_LastReadLba 번 이미지에서 가장 최근에 읽은 섹터의 주소를 검색합니다.
get_LastUsedUserSectorInImage 기존 트랙 간의 오버헤드를 포함하여 현재 미디어에서 사용된 총 섹터 수를 검색합니다.
get_LastWrittenAddress 미디어에서 마지막으로 쓴 주소를 검색합니다.
get_LastWrittenAddressOfPreviousSession 이전 쓰기 세션의 마지막 섹터를 검색합니다.
get_LastWrittenAddressOfPreviousSession 미디어의 이전 세션에서 작성된 마지막 섹터를 검색합니다.
get_LastWrittenLba 디바이스에 가장 최근에 작성된 섹터의 주소를 검색합니다.

get_LegacyDeviceNumber CD 또는 DVD 디바이스의 레거시 디바이스 번호를 검색합니다.
get_Manufacturer CD 제조업체의 식별자를 검색합니다.
get_MediaCatalogNumber 전체 오디오 디스크에 대한 MCN(미디어 카탈로그 번호)을 설정합니다.
get_MediaHeuristicallyBlank 추론을 사용하여 미디어가 비어 있는지 확인하려고 시도합니다(주로 DVD+RW 및 DVD-RAM 미디어용).
get_MediaPhysicallyBlank 현재 미디어가 드라이브에 의해 물리적으로 비어 있는 것으로 보고되는지 여부를 확인합니다.
get_MediaType 현재 드라이브의 미디어 유형을 검색합니다.
get_ModifiedBlocks 결과 이미지에서 수정된 블록 목록을 검색합니다.
get_MultisessionInterfaces 사용 가능한 다중 세션 인터페이스 목록을 검색합니다.
get_MultisessionInterfaces 광학 미디어에 대한 다중 세션 인터페이스 목록을 검색합니다.
get_Name 파일 시스템 이미지에서 디렉터리 또는 파일 항목의 이름을 검색합니다.
get_NextWritableAddress 다음 쓰기 작업의 위치를 검색합니다.
get_NextWritableAddress 사용된 섹터를 포함하여 미디어에서 다음 쓰기 가능한 주소를 검색합니다.

get_NumberOfExistingTracks 미디어의 기존 오디오 트랙 수를 검색합니다. (IDiscFormat2TrackAtOnce.get_NumberOfExistingTracks)
get_NumberOfExistingTracks 미디어의 기존 오디오 트랙 수를 검색합니다. (IRawCDImageCreator.get_NumberOfExistingTracks)
get_Path .iso 이미지의 논리적 경로를 검색합니다.
get_PlatformId 부팅 이미지가 지원하는 운영 체제 아키텍처를 식별하는 플랫폼 식별자를 검색합니다.
get_PostgapAlreadyInImage 데이터 스트림에 사후 쓰기 간격이 포함되어 있는지 확인합니다. (Get)
get_ProductId 디바이스의 제품 ID를 검색합니다.
get_ProductRevision 디바이스의 제품 수정 코드를 검색합니다.
get_ProgressItems 진행률 항목 블록 매핑 컬렉션을 검색합니다.
get_Recorder 쓰기 작업에 사용할 기록 디바이스를 검색합니다. (IDiscFormat2Data.get_Recorder)
get_Recorder 지우기 작업에 사용할 기록 디바이스를 검색합니다.
get_Recorder 쓰기 작업에 사용할 기록 디바이스를 검색합니다. (IDiscFormat2RawCD.get_Recorder)
get_Recorder 쓰기 작업에 사용할 기록 디바이스를 검색합니다. (IDiscFormat2TrackAtOnce.get_Recorder)

get_Recorder 쓰기 작업에 사용할 기록 디바이스를 검색합니다.
get_RemainingTime 쓰기 작업의 예상 남은 시간을 검색합니다. (IDiscFormat2DataEventArgs.get_RemainingTime)
get_RemainingTime 쓰기 작업의 예상 남은 시간을 검색합니다. (IDiscFormat2RawCDEventArgs.get_RemainingTime)
get_RemainingTime 쓰기 작업의 예상 남은 시간을 검색합니다. (IDiscFormat2TrackAtOnceEventArgs.get_RemainingTime)
get_RequestedRotationTypeIsPureCAV 요청된 회전 속도 제어 유형을 검색합니다. (IDiscFormat2Data.get_RequestedRotationTypeIsPureCAV)
get_RequestedRotationTypeIsPureCAV 요청된 회전 속도 제어 유형을 검색합니다. (IDiscFormat2RawCD.get_RequestedRotationTypeIsPureCAV)
get_RequestedRotationTypeIsPureCAV 요청된 회전 속도 제어 유형을 검색합니다. (IDiscFormat2TrackAtOnce.get_RequestedRotationTypeIsPureCAV)
get_RequestedSectorType 스트림을 쓰는 동안 사용할 요청된 데이터 섹터를 검색합니다.
get_RequestedWriteSpeed 요청된 쓰기 속도를 검색합니다. (IDiscFormat2Data.get_RequestedWriteSpeed)
get_RequestedWriteSpeed 요청된 쓰기 속도를 검색합니다. (IDiscFormat2RawCD.get_RequestedWriteSpeed)
get_RequestedWriteSpeed 요청된 쓰기 속도를 검색합니다. (IDiscFormat2TrackAtOnce.get_RequestedWriteSpeed)

get_ResultingImageType 생성될 이미지 파일의 형식을 지정하는 값을 검색합니다.
get_Root 루트 디렉터리 항목을 검색합니다.
get_RotationTypeIsPureCAV 현재 미디어에 대한 레코더에서 사용하는 지원되는 회전 속도 컨트롤을 검색합니다.
get_SectorCount 이 트랙의 사용자 섹터 수를 검색합니다.
get_SectorCount 현재 쓰기 작업에서 디바이스에 쓸 섹터 수를 검색합니다.
get_SectorType 이 트랙의 섹터에 제공된 데이터 형식을 검색합니다. 가능한 섹터 유형에 대한 자세한 내용은 IMAPI_CD_SECTOR_TYPE 참조하세요.
get_Seed 난수 생성기에서 사용하는 시드 값을 검색합니다.
get_SessionStartBlock 녹음/녹화 세션의 시작 블록 주소를 검색합니다.
get_StageFiles 파일 시스템 이미지에 추가되는 파일을 굽기 전에 준비해야 하는지를 나타냅니다.
get_StartAddressOfPreviousSession 이전 쓰기 세션의 첫 번째 섹터를 검색합니다.
get_StartAddressOfPreviousSession 미디어의 이전 세션에서 작성된 첫 번째 섹터를 검색합니다.
get_StartingLba 이 트랙에서 첫 번째 사용자 섹터의 LBA를 검색합니다.

get_StartingSectorsPerSecond 기록 장치가 쓰기 프로세스를 시작할 때 미디어에 쓸 수 있는 초당 예상 섹터 수를 검색합니다.
get_StartingTrackNumber 시작 트랙 번호를 검색합니다.
get_StartLba IBlockRange에서 설명하는 범위의 시작 섹터를 검색합니다.
get_StartLba 현재 쓰기 작업의 LBA(시작 논리 블록 주소)를 검색합니다.
get_StartOfLeadout 리드아웃의 시작 부분에 대한 LBA를 정의하는 값을 검색합니다. 이 메서드를 사용하여 미디어의 LastPossibleStartOfLeadout과 비교하여 이미지를 미디어에 쓸 수 있는지 여부를 확인할 수 있습니다.
get_StartOfLeadoutLimit 현재 StartOfLeadoutLimit 속성 값을 검색합니다. 이 값은 결과 이미지가 LBA보다 크거나 같은 StartOfLeadout이 있는 미디어 조각에 맞춰야 하는지 여부를 지정합니다.
get_StartOfNextSession 다음 세션의 첫 번째 섹터를 검색합니다.
get_Stream .iso 이미지와 연결된 IStream 개체를 검색합니다.
get_StrictFileSystemCompliance 파일 시스템 이미지를 만들고 개발하기 위한 준수 수준을 결정합니다. (Get)
get_SupportedFeaturePages 디바이스에서 지원하는 기능 목록을 검색합니다.
get_SupportedMediaTypes IDiscFormat2 인터페이스의 현재 구현에서 지원하는 미디어 형식을 검색합니다.

get_SupportedModePages 디바이스에서 지원하는 MMC 모드 페이지 목록을 검색합니다.
get_SupportedProfiles 디바이스에서 지원하는 MMC 프로필 목록을 검색합니다.
get_SupportedSectorTypes 현재 레코더에 대해 지원되는 데이터 섹터 형식을 검색합니다.
get_SupportedWriteSpeedDescriptors 디스크 레코더 및 현재 미디어에서 지원하는 자세한 쓰기 구성 목록을 검색합니다. (IDiscFormat2Data.get_SupportedWriteSpeedDescriptors)
get_SupportedWriteSpeedDescriptors 디스크 레코더 및 현재 미디어에서 지원하는 자세한 쓰기 구성 목록을 검색합니다. (IDiscFormat2RawCD.get_SupportedWriteSpeedDescriptors)
get_SupportedWriteSpeedDescriptors 디스크 레코더 및 현재 미디어에서 지원하는 자세한 쓰기 구성 목록을 검색합니다. (IDiscFormat2TrackAtOnce.get_SupportedWriteSpeedDescriptors)
get_SupportedWriteSpeeds 디스크 레코더 및 현재 미디어에서 지원하는 쓰기 속도 목록을 검색합니다. (IDiscFormat2Data.get_SupportedWriteSpeeds)
get_SupportedWriteSpeeds 디스크 레코더 및 현재 미디어에서 지원하는 쓰기 속도 목록을 검색합니다. (IDiscFormat2RawCD.get_SupportedWriteSpeeds)
get_SupportedWriteSpeeds 디스크 레코더 및 현재 미디어에서 지원하는 쓰기 속도 목록을 검색합니다. (IDiscFormat2TrackAtOnce.get_SupportedWriteSpeeds)
get_TotalBlocks 결과 이미지의 블록 수를 검색합니다.
get_TotalSectorsOnMedia 디바이스의 미디어에서 섹터 수를 검색합니다.

get_TotalSectorsOnMedia 하나의 연속 오디오 트랙을 작성하는 경우 미디어에서 사용할 수 있는 총 섹터를 검색합니다.
get_TotalSectorsOnMedia 미디어의 총 섹터 수를 검색합니다.
get_TotalSystemBuffer 디스크에 쓰는 데 사용되는 내부 데이터 버퍼의 크기를 검색합니다.
get_TotalTime 쓰기 작업의 예상 총 시간을 검색합니다.
get_TrackIndexes 디스크에 있는 트랙의 1 기반 인덱스를 검색합니다.
get_TrackInfo 사용자가 쿼리하는 추적을 결정하는 인덱스로 범위가 1~99인 LONG 값을 사용하는 인덱싱된 속성을 검색합니다. 그런 다음 반환된 개체는 관심 있는 특정 트랙별 속성에 대해 쿼리/설정됩니다.
get_TrackNumber 이 트랙의 트랙 번호를 검색합니다.
get_UDFRevision 가져온 파일 시스템 이미지의 UDF 수정 수준을 검색합니다.
get_UDFRevisionsSupported 지원되는 UDF 수정 수준 목록을 검색합니다.
get_UsedBlocks 사용 중인 블록 수를 검색합니다.
get_UsedSectorsOnMedia 미디어에서 사용된 총 섹터 수를 검색합니다.
get_UsedSystemBuffer 디스크에 쓰는 데 사용되는 내부 데이터 버퍼의 사용된 바이트 수를 검색합니다.

get_UseRestrictedCharacterSet 파일 및 디렉터리 이름이 제한된 문자를 사용하는지 여부를 확인합니다.
get_UseStreamingWrite12 쓰기 작업에서 WRITE12 또는 WRITE10 명령을 사용하는지 여부를 나타내는 값을 검색합니다.
get_VendorId 디바이스의 공급업체 ID를 검색합니다.
get_VolumeName 디바이스와 연결된 고유한 볼륨 이름을 검색합니다.
get_VolumeName 이 파일 시스템 이미지의 볼륨 이름을 검색합니다.
get_VolumeNameISO9660 ISO9660 시스템 이미지의 볼륨 이름을 검색합니다.
get_VolumeNameJoliet Joliet 시스템 이미지의 볼륨 이름을 검색합니다.
get_VolumeNameUDF UDF 시스템 이미지의 볼륨 이름을 검색합니다.
get_VolumePathNames 디바이스의 드라이브 문자 및 NTFS 탑재 지점 목록을 검색합니다.
get_WorkingDirectory 스태시 파일이 빌드되는 임시 디렉터리를 검색합니다.
get_WriteInProgress 레코더가 현재 디스크에 데이터를 쓰고 있는지 여부를 나타내는 값을 검색합니다.
get_WriteProtectStatus 디바이스에서 미디어의 현재 쓰기 보호 상태를 검색합니다.

get_WriteSpeed 미디어에 쓰기에 지원되는 쓰기 속도를 검색합니다.
get_WriteUnitSize 미디어에서 쓰기 가능한 단위의 크기를 검색합니다. (IMultisessionRandomWrite.get_WriteUnitSize)
get_WriteUnitSize 미디어에서 쓰기 가능한 단위의 크기를 검색합니다. (IMultisessionSequential2.get_WriteUnitSize)
GetActiveDiscMasterFormat 활성 디스크 레코더 형식을 검색합니다. 활성 형식은 스테이징된 이미지 파일 콘텐츠(오디오/데이터)의 구조와 해당 스테이징된 이미지를 조작하는 데 사용해야 하는 COM 인터페이스를 모두 지정합니다.
GetActiveDiscRecorder 활성 디스크 레코더에 대한 인터페이스 포인터를 검색합니다. 활성 디스크 레코더는 RecordDisc가 호출될 때 굽기가 발생하는 레코더입니다.
GetAdapterDescriptor 디바이스의 어댑터 설명자를 검색합니다.
GetAudioBlockSize 오디오 블록의 크기(바이트)를 검색합니다.
GetAvailableAudioTrackBlocks 추가 추가로 인해 공간이 부족하기 전에 트랙에 추가할 수 있는 현재 블록 수를 검색합니다.
GetBasePnPID 생성 및 모델을 통해 특정 디바이스 클래스를 일관되게 식별하는 데 사용할 수 있는 기본 PnP 문자열을 검색합니다. 이 문자열은 애플리케이션에서 특정 레코더 유형에 따라 동작을 사용자 지정하는 데 사용할 수 있습니다.
GetByteAlignmentMask 디바이스의 바이트 맞춤 마스크를 검색합니다.
GetDataBlockSize 데이터 블록의 크기를 검색합니다.

GetDefaultFileSystemForImport 기본적으로 가져올 파일 시스템을 검색합니다.
GetDeviceDescriptor 디바이스에 대한 디바이스 설명자를 검색합니다.
GetDiscInformation 미디어에서 디스크 정보를 검색합니다.
GetDisplayNames 표시할 수 있는 레코더의 서식이 지정된 이름을 검색합니다. 이름은 디바이스의 제조업체 및 제품 식별자로 구성됩니다.
GetFeaturePage 디바이스에서 지정된 기능 페이지를 검색합니다.
GetJolietProperties Joliet 속성을 포함하는 IPropertyStorage 인터페이스에 대한 포인터를 검색합니다.
GetMaximumNonPageAlignedTransferSize 디바이스에 대한 페이지 정렬되지 않은 최대 전송 크기를 검색합니다.
GetMaximumPageAlignedTransferSize 디바이스의 최대 페이지 정렬 전송 크기를 검색합니다.
GetModePage 디바이스에서 지정된 모드 페이지를 검색합니다.
GetPath 운영 체제 내에서 디바이스의 경로를 검색합니다. 이 경로는 사용 가능한 디스크 레코더를 완전히 식별하기 위해 표시 이름과 함께 사용해야 합니다.
GetRecorderGUID 현재 레코더 개체와 연결된 실제 디스크 레코더의 GUID를 검색합니다.
GetRecorderProperties IPropertyStorage 인터페이스에 대한 포인터를 검색합니다.

GetRecorderState 디스크 레코더 상태를 검색합니다.
GetRecorderType 디스크 레코더가 CD-R 또는 CD-RW 유형 디바이스인지 여부를 확인합니다. 이는 현재 디바이스에 삽입된 미디어 유형을 나타내지 않습니다.
GetSupportedFeaturePages 지원되는 기능 페이지 또는 디바이스의 현재 기능 페이지 목록을 검색합니다.
GetSupportedModePages 디바이스에 대해 지원되는 모드 페이지를 검색합니다.
GetSupportedProfiles 지원되는 프로필 또는 디바이스의 현재 프로필을 검색합니다.
GetTotalAudioBlocks 스테이징 오디오 트랙에 사용할 수 있는 총 블록 수를 검색합니다. 총계에는 트랙 간격에 할당해야 할 수 있는 블록을 포함하여 모든 블록 형식이 포함됩니다.
GetTotalAudioTracks 스테이징되었거나 준비 중인 총 트랙 수를 검색합니다.
GetTotalDataBlocks Joliet 데이터 디스크를 준비하는 데 사용할 수 있는 총 블록 수를 검색합니다.
GetTrackInformation 미디어에서 트랙 정보를 검색합니다.
GetUsedAudioBlocks 사용 중인 총 오디오 블록 수를 검색합니다.
GetUsedDataBlocks 사용 중인 총 데이터 블록 수를 검색합니다.
IdentifyFileSystemsOnDisc 광학 미디어에서 다양한 유형의 파일 시스템 목록을 검색합니다.

ImportFileSystem 현재 디스크의 기본 파일 시스템을 가져옵니다.
ImportSpecificFileSystem 디스크에서 특정 파일 시스템을 가져옵니다.
초기화 두 개의 입력 스트림에서 이 스트림을 초기화합니다.
초기화 입력 스트림 및 인터리브 크기의 배열에서 이 인터리브 스트림을 초기화합니다.
Initialize2 입력 스트림 배열에서 이 스트림을 초기화합니다.
InitializeDiscRecorder 개체를 지정된 디스크 디바이스와 연결합니다.
IsCurrentMediaSupported 지원되는 레코더의 현재 미디어가 지정된 형식을 지원하는지 여부를 확인합니다.
IsRecorderSupported 레코더가 지정된 형식을 지원하는지 여부를 결정합니다.
LockInChangePoint 현재 변경 지점 수준에서 파일 시스템 정보를 잠급 수 있습니다.
다음 열거형 시퀀스에서 지정된 수의 항목을 검색합니다. (IEnumFsiItems.Next)
다음 열거형 시퀀스에서 지정된 수의 항목을 검색합니다. (IEnumProgressItems.Next)
NotifyAddProgress IRedbookDiscMaster::AddAudioTrackBlocks 또는 IJolietDiscMaster::AddData 호출에 대한 응답으로 애플리케이션에 진행 상황을 알릴 수 있습니다. 알림은 첫 번째 단계와 마지막 단계와 그 사이의 지점에서 전송됩니다.

NotifyBlockProgress 활성 레코더에서 디스크를 레코딩하는 진행 상황을 애플리케이션에 알합니다. 알림은 첫 번째 및 마지막 블록과 그 사이의 지점에서 전송됩니다.
NotifyBurnComplete IDiscMaster::RecordDisc에 대한 호출이 완료되었음을 애플리케이션에 알합니다.
NotifyClosingDisc 디스크를 닫기 시작했음을 애플리케이션에 알합니다. 화상이 완료될 때까지 추가 알림이 전송되지 않습니다.
NotifyDeviceAdded 광학 미디어 디바이스가 컴퓨터에 추가되면 알림을 받습니다.
NotifyDeviceRemoved 컴퓨터에서 광학 미디어 디바이스가 제거되면 알림을 받습니다.
NotifyEraseComplete IDiscRecorder::Erase에 대한 호출이 완료되었음을 애플리케이션에 알합니다.
NotifyPnPActivity 유효한 디스크 레코더 목록이 변경되었다는 것을 애플리케이션에 알합니다. (예를 들어 USB CD-R 드라이버는 시스템에서 제거됩니다.).
NotifyPreparingBurn 디스크를 구울 준비를 하고 있음을 애플리케이션에 알립니다. 화상이 시작될 때까지 추가 알림이 전송되지 않습니다.
NotifyTrackProgress 오디오 디스크를 태우는 동안 트랙이 시작되거나 완료되었음을 애플리케이션에 알립니다.
열기 클라이언트 애플리케이션에서 액세스할 수 있도록 상위 수준 IMAPI 개체를 엽니다.
OpenExclusive 단독 액세스를 위해 디스크 레코더를 엽니다.

PrepareMedia 전용 액세스를 위해 현재 미디어를 잠급 수 있습니다. (IDiscFormat2RawCD.PrepareMedia)
PrepareMedia 전용 액세스를 위해 현재 미디어를 잠급 수 있습니다. (IDiscFormat2TrackAtOnce.PrepareMedia)
ProbeSpecificFileSystem 현재 미디어의 특정 파일 시스템이 IMAPI를 통해 추가할 수 있는지 여부를 확인합니다.
ProgressAdvise 진행률 알림을 위해 애플리케이션을 등록합니다.
ProgressItemFromBlock 지정된 블록 번호에 따라 진행률 항목을 검색합니다.
ProgressItemFromDescription 지정된 파일 이름을 기반으로 진행률 항목을 검색합니다.
ProgressUnadvise 애플리케이션에 대한 진행률 알림을 취소합니다.
put_AudioHasPreemphasis 오디오 트랙에 CD에 기록되기 전에 오디오 데이터에 추가된 추가 사전 강조가 있는지를 지정하는 값을 설정합니다.
put_BootImageOptions 파일 시스템 이미지에 추가하려는 부팅 이미지를 설정합니다. 이 메서드는 제공된 IBootOptions 인터페이스에서 스트림을 복사하여 전달된 부팅 옵션의 전체 복사본을 만듭니다.
put_BootImageOptionsArray 파일 시스템 이미지를 생성하는 데 사용할 부팅 옵션 배열을 설정합니다. IFileSystemImage::put_BootImageOptions와 달리 이 메서드는 각 부팅 옵션 배열 요소의 전체 복사본을 만들지 않고 대신 각 요소에 대한 참조를 사용합니다.
put_BufferUnderrunFreeDisabled 버퍼 언더런 무료 기록을 사용할 수 있는지 여부를 확인합니다. (IDiscFormat2Data.put_BufferUnderrunFreeDisabled)

put_BufferUnderrunFreeDisabled 버퍼 언더런 무료 기록을 사용할 수 있는지 여부를 확인합니다. (IDiscFormat2RawCD.put_BufferUnderrunFreeDisabled)
put_BufferUnderrunFreeDisabled 버퍼 언더런 무료 기록을 사용할 수 있는지 여부를 확인합니다.
put_BurnVerificationLevel 번 확인 수준을 설정합니다.
put_BytesPerSector 쓰기 중에 각 섹터에 사용할 바이트 수를 설정합니다.
put_ClientName 클라이언트의 식별 이름을 설정합니다. (IDiscFormat2Data.put_ClientName)
put_ClientName 클라이언트의 식별 이름을 설정합니다. (IDiscFormat2Erase.put_ClientName)
put_ClientName 클라이언트의 식별 이름을 설정합니다. (IDiscFormat2RawCD.put_ClientName)
put_ClientName 클라이언트의 식별 이름을 설정합니다. (IDiscFormat2TrackAtOnce.put_ClientName)
put_CreateRedundantUdfMetadataFiles UDF 메타데이터가 파일 시스템 이미지에서 중복되는지를 지정하는 속성을 설정합니다.
put_CreationTime 디렉터리 또는 파일 항목을 만들고 파일 시스템 이미지에 추가한 날짜 및 시간을 설정합니다.
put_Data 파일 콘텐츠의 데이터 스트림을 설정합니다.
put_DigitalAudioCopySetting 디지털 오디오 복사본 "허용됨" 비트를 결과 미디어의 세 값 중 하나로 설정합니다. 가능한 각 값에 대한 추가 정보는 IMAPI_CD_TRACK_DIGITAL_COPY_SETTING 열거형을 참조하세요.

put_DisableConsumerDvdCompatibilityMode DVD 녹음 세션에 장치가 DVD를 재생할 수 있는 가능성을 높일 수 있는 작업이 포함되어 있는지 확인합니다. (Put)
put_DisableGaplessAudio "갭리스 오디오" 녹음을 사용할 수 없는지 지정하는 값을 설정합니다. 이 속성은 기본적으로 VARIANT_FALSE 값으로 설정되며 연속 오디오 트랙 간에 "간격 없는" 녹음을 사용할 수 없습니다.
put_DoNotFinalizeMedia 오디오 트랙을 작성한 후 미디어가 쓰기 위해 열려 있는지 여부를 결정합니다. (Put)
put_Emulation 부팅 이미지가 에뮬레이트할 미디어 유형을 설정합니다.
put_EndingSectorsPerSecond 기록 디바이스가 쓰기 프로세스가 끝날 때 미디어에 쓸 수 있는 초당 예상 섹터 수를 설정합니다.
put_ExtendedSeed 난수 생성기의 시드 값 목록을 설정하고 스트림의 시작을 찾습니다.
put_FileSystemsToCreate 결과 스트림을 생성할 때 만들 파일 시스템을 설정합니다.
put_ForceMediaToBeClosed 파일 시스템에 대한 추가가 방지되는지 여부를 확인합니다. (Put)
put_ForceOverwrite 데이터 작성기가 덮어쓸 수 있는 미디어 형식에서 디스크를 덮어써야 하는지 여부를 결정합니다. (Put)
put_FreeMediaBlocks 이미지에 사용할 수 있는 최대 블록 수를 설정합니다.
put_FullErase 디스크 지우기의 품질을 결정합니다. (Put)

put_InUse 이 다중 세션 인터페이스가 현재 미디어에서 사용해야 하는 인터페이스인지 확인합니다. (Put)
put_IsHidden 파일 시스템 이미지에서 항목의 숨겨진 특성이 설정되어 있는지 여부를 확인합니다. (Put)
put_ISO9660InterchangeLevel 파일 시스템 이미지의 ISO9660 호환성 수준을 설정합니다.
put_ISRC 현재 트랙과 연결된 ISRC(International Standard Recording Code)를 설정합니다. 이 속성 값은 기본적으로 NULL(또는 길이가 0인 문자열)이며 오디오 데이터를 포함하는 트랙에 대해서만 설정할 수 있습니다.
put_IsRealTime 파일 시스템에서 파일의 '실시간' 특성을 설정합니다. 이 특성은 콘텐츠에 오디오 및 비디오 데이터와 같은 쓰기 또는 읽기 시 최소 데이터 전송 속도가 필요한지 여부를 지정합니다.
put_LastAccessedTime 파일 시스템 이미지에서 디렉터리 또는 파일 항목에 마지막으로 액세스한 날짜 및 시간을 설정합니다.
put_LastModifiedTime 파일 시스템 이미지에서 항목이 마지막으로 수정된 날짜 및 시간을 설정합니다.
put_Manufacturer CD의 제조업체 또는 개발자를 식별하는 식별자를 설정합니다.
put_MediaCatalogNumber 전체 오디오 디스크에 대한 MCN(미디어 카탈로그 번호)을 검색합니다.
put_MultisessionInterfaces 광학 미디어의 다중 세션 인터페이스 목록을 설정합니다.
put_PlatformId 부팅 이미지가 지원하는 운영 체제 아키텍처를 식별하는 플랫폼 식별자를 설정합니다.

put_PostgapAlreadyInImage 데이터 스트림에 사후 쓰기 간격이 포함되어 있는지 확인합니다. (Put)
put_Recorder 쓰기 작업에 사용할 기록 디바이스를 설정합니다. (IDiscFormat2Data.put_Recorder)
put_Recorder 지우기 작업에 사용할 기록 디바이스를 설정합니다.
put_Recorder 쓰기 작업에 사용할 기록 디바이스를 설정합니다. (IDiscFormat2RawCD.put_Recorder)
put_Recorder 쓰기 작업에 사용할 기록 디바이스를 설정합니다. (IDiscFormat2TrackAtOnce.put_Recorder)
put_Recorder 쓰기 작업에 대한 기록 디바이스를 설정합니다.
put_RequestedSectorType 스트림을 작성하는 데 사용할 요청된 데이터 섹터를 설정합니다.
put_ResultingImageType 생성될 이미지 파일의 형식을 정의하는 값을 설정합니다.
put_Seed 난수 생성기에서 사용하는 시드 값을 설정하고 스트림의 시작을 찾습니다.
put_SessionStartBlock 기록 세션의 시작 블록 주소를 설정합니다.
put_StageFiles 파일 시스템 이미지에 추가되는 파일을 화상 전에 준비해야 하는지 여부를 결정합니다.
put_StartingSectorsPerSecond 기록 디바이스가 쓰기 프로세스를 시작할 때 미디어에 쓸 수 있는 초당 예상 섹터 수를 설정합니다.

put_StartingTrackNumber 시작 트랙 번호를 설정합니다.
put_StartOfLeadoutLimit StartOfLeadoutLimit 속성 값을 설정합니다.
put_StrictFileSystemCompliance 파일 시스템 이미지를 만들고 개발하기 위한 준수 수준을 결정합니다. (Put)
put_UDFRevision 파일 시스템 이미지의 UDF 수정 수준을 설정합니다.
put_UseRestrictedCharacterSet 파일 및 디렉터리 이름을 CP_ANSI 문자만 사용하도록 제한해야 하는지 여부를 결정합니다.
put_UseStreamingWrite12 쓰기 작업에서 WRITE12 또는 WRITE10 명령을 사용하는지 여부를 나타내는 값을 설정합니다.
put_VolumeName 이 파일 시스템 이미지의 볼륨 이름을 설정합니다.
put_WorkingDirectory 스태시 파일이 빌드되는 임시 디렉터리를 설정합니다.
QueryCancel AddData, AddAudioTrackBlocks 또는 RecordDisc 작업을 취소해야 하는지 여부를 확인합니다.
QueryMediaInfo 미디어에 사용되는 총 블록 수와 같이 현재 탑재된 미디어에 대한 정보를 검색합니다.
QueryMediaType 현재 레코더에 삽입된 미디어 유형(있는 경우)을 검색합니다.
ReadDvdStructure 미디어에서 DVD 구조를 읽습니다.

RecordDisc 활성 디스크 레코더의 미디어에 스테이징된 이미지를 연화합니다.
ReleaseExclusiveAccess 디바이스에 대한 단독 액세스를 해제합니다.
ReleaseMedia 원시 이미지의 DAO(Disc-At-Once) 쓰기 세션을 닫고 잠금을 해제합니다.
ReleaseMedia 트랙 쓰기 세션을 닫고 잠금을 해제합니다.
제거 파일 시스템 이미지에서 지정된 항목을 제거합니다.
RemoveStream 파일과 명명된 스트림 연결을 제거합니다.
RemoveTree 파일 시스템 이미지에서 지정된 디렉터리 트리를 제거합니다.
재설정 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumFsiItems.Reset)
재설정 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumProgressItems.Reset)
RollbackToChangePoint 이미지를 지정된 변경 지점으로 되돌려 줍니다.
SendCommandGetDataFromDevice 디바이스에서 데이터를 요청하는 기록 디바이스에 MMC 명령을 보냅니다.
SendCommandNoData 기록 디바이스에 MMC 명령을 보냅니다. 데이터 버퍼가 디바이스로 전송되거나 디바이스에서 수신되지 않는 경우 이 함수를 사용합니다.

SendCommandSendDataToDevice MMC 명령 및 관련 데이터 버퍼를 기록 디바이스로 보냅니다.
SendDvdStructure DVD 구조를 미디어에 보냅니다.
SetActiveDiscMasterFormat 현재 활성 디스크 레코더 형식을 설정합니다. 활성 형식은 스테이징된 이미지 파일 콘텐츠(오디오/데이터)의 구조와 해당 스테이징된 이미지를 조작하는 데 사용해야 하는 COM 인터페이스를 모두 지정합니다.
SetActiveDiscRecorder 활성 디스크 레코더를 선택합니다. 활성 디스크 레코더는 RecordDisc가 호출될 때 굽기가 발생하는 레코더입니다.
SetJolietProperties Joliet 속성을 설정합니다.
SetMaxMediaBlocksFromDevice 레코더의 기능에 따라 사용할 수 있는 최대 블록 수를 설정합니다.
SetModePage 디바이스의 모드 페이지 데이터를 설정합니다.
SetPath .iso 이미지에 대한 논리적 경로를 사용하여 Path 속성 값을 설정합니다.
SetRecorderProperties 애플리케이션이 변경하려는 모든 속성을 가진 개체에 대한 IPropertyStorage 포인터를 허용합니다. 스파스 설정이 지원됩니다.
SetStream .iso 이미지와 연결된 IStream 개체를 사용하여 Stream 속성을 설정합니다.
SetWriteSpeed 디스크 레코더의 쓰기 속도를 설정합니다. (IDiscFormat2Data.SetWriteSpeed)

SetWriteSpeed 디스크 레코더의 쓰기 속도를 설정합니다. (IDiscFormat2RawCD.SetWriteSpeed)
SetWriteSpeed 디스크 레코더의 쓰기 속도를 설정합니다. (IDiscFormat2TrackAtOnce.SetWriteSpeed)
skip 열거형 시퀀스에서 지정된 수의 항목을 건너뜁니다. (IEnumFsiItems.Skip)
skip 열거형 시퀀스에서 지정된 수의 항목을 건너뜁니다. (IEnumProgressItems.Skip)
업데이트 현재 쓰기 작업의 진행률 알림을 받으려면 이 메서드를 구현합니다. (DDiscFormat2DataEvents.Update)
업데이트 현재 지우기 작업의 진행률 알림을 받으려면 이 메서드를 구현합니다.
업데이트 현재 원시 이미지 쓰기 작업의 진행률 알림을 받으려면 이 메서드를 구현합니다.
업데이트 현재 트랙 쓰기 작업의 진행률 알림을 받으려면 이 메서드를 구현합니다.
업데이트 현재 쓰기 작업의 진행률 알림을 받으려면 이 메서드를 구현합니다. (DWriteEngine2Events.Update)
업데이트 현재 쓰기 작업의 진행률 알림을 받으려면 이 메서드를 구현합니다. 파일의 콘텐츠를 복사하거나 파일 시스템 이미지에 디렉터리 또는 파일을 추가하는 동안 알림이 전송됩니다.
UpdateImport 광학 매체에서 가져온 모든 파일 및 디렉터리 항목에 대한 가져오기 알림을 받습니다.
유효성 검사 제공된 .iso 이미지가 유효한지 확인합니다.

쓰기 디바이스에 데이터 스트림을 씁니다.
WriteMedia 시작 주소로 MSF 95:00:00을 사용하여 빈 미디어에 DAO-96 원시 이미지를 씁니다.
WriteMedia2 지정된 시작 주소를 사용하여 빈 미디어에 DAO-96 원시 이미지를 씁니다.
WriteSection 현재 레코더에 데이터 스트림을 씁니다.

인터페이스

DDiscFormat2DataEvents 현재 쓰기 작업의 알림을 받도록 이 인터페이스를 구현합니다. (DDiscFormat2DataEvents)
DDiscFormat2EraseEvents 현재 지우기 작업에 대한 알림을 받도록 이 인터페이스를 구현합니다.
DDiscFormat2RawCDEvents 이 인터페이스를 구현하여 현재 원시 이미지 쓰기 작업의 알림을 받습니다.
DDiscFormat2TrackAtOnceEvents 이 인터페이스를 구현하여 현재 추적 쓰기 작업에 대한 알림을 받습니다.
DDiscMaster2Events CD 또는 DVD 디바이스가 컴퓨터에 추가되거나 컴퓨터에서 제거될 때 알림을 받도록 이 인터페이스를 구현합니다.
DFileSystemImageEvents 현재 쓰기 작업의 알림을 받도록 이 인터페이스를 구현합니다. (DFileSystemImageEvents)

DFileSystemImageImportEvents 이 인터페이스를 사용하여 현재 파일 시스템 가져오기 작업에 대한 알림을 받습니다.
DWriteEngine2Events 현재 쓰기 작업의 알림을 받도록 이 인터페이스를 구현합니다. (DWriteEngine2Events)
IBlockRange 이 인터페이스를 사용하여 미디어에서 단일 연속 섹터 범위에 대한 정보를 검색합니다. 이 인터페이스는 일반적으로 IBlockRangeList 인터페이스와 함께 섹터 범위 컬렉션을 설명하는 데 사용됩니다.
IBlockRangeList 이 인터페이스를 사용하여 미디어에서 연속 섹터 범위 목록을 검색합니다. 이 인터페이스는 새 논리 세션이 기록될 때 다시 작성 가능한 디스크에서 업데이트해야 하는 섹터를 설명하는 데 사용됩니다.
IBootOptions 이 인터페이스를 사용하여 광학 디스크에 추가할 부팅 이미지를 지정합니다. 부팅 이미지에는 컴퓨터를 시작하는 데 사용되는 하나 이상의 코드 섹터가 포함되어 있습니다.
IBurnVerification IDiscFormat2Data 또는 IDiscFormat2TrackAtOnce와 함께 이 인터페이스를 사용하여 쓰기 작업 후 불타는 미디어가 무결성을 확인하는 방법을 나타내는 Burn 확인 수준 속성을 얻거나 설정합니다.
IDiscFormat2 기본 인터페이스입니다. 이 인터페이스 IDiscFormat2Data, IDiscFormat2Erase, IDiscFormat2TrackAtOnce, IDiscFormat2RawCD를 상속하는 다음 인터페이스를 사용합니다.
IDiscFormat2Data 이 인터페이스를 사용하여 디스크에 데이터 스트림을 작성합니다.
IDiscFormat2DataEventArgs 이 인터페이스를 사용하여 현재 쓰기 작업에 대한 정보를 검색합니다. (IDiscFormat2DataEventArgs)
IDiscFormat2Erase 디스크에서 데이터를 지우려면 이 인터페이스를 사용합니다.

IDiscFormat2RawCD 이 인터페이스를 사용하여 DAO(디스크 한 번에) 모드(중단 없는 기록이라고도 함)를 사용하여 디스크 디바이스에 원시 이미지를 작성합니다.
IDiscFormat2RawCDEventArgs 이 인터페이스를 사용하여 현재 쓰기 작업에 대한 정보를 검색합니다. (IDiscFormat2RawCDEventArgs)
IDiscFormat2TrackAtOnce 이 인터페이스를 사용하여 한 번에 추적 모드에서 빈 CD-R 또는 CD-RW 미디어에 오디오를 작성합니다.
IDiscFormat2TrackAtOnceEventArgs 이 인터페이스를 사용하여 현재 쓰기 작업에 대한 정보를 검색합니다. (IDiscFormat2TrackAtOnceEventArgs)
IDiscMaster IDiscMaster 인터페이스를 사용하면 애플리케이션에서 이미지 마스터 API를 예약하고, 이미지 마스터 개체에서 지원하는 디스크 마스터 형식 및 디스크 레코더를 열거하고, 디스크의 시뮬레이트되거나 실제 화상을 시작할 수 있습니다.
IDiscMaster2 이 인터페이스를 사용하여 컴퓨터에 설치된 CD 및 DVD 디바이스를 열거합니다.
IDiscMasterProgressEvents IDiscMasterProgressEvents 인터페이스는 IMAPI에서 애플리케이션으로 만들 수 있는 모든 콜백에 대한 단일 인터페이스를 제공합니다.
IDiscRecorder IDiscRecorder 인터페이스를 사용하면 활성 디스크 레코더로 레이블이 지정된 단일 디스크 레코더 디바이스에 액세스할 수 있습니다. MSDiscMasterObj와 같은 IMAPI 개체는 활성 디스크 레코더를 유지 관리합니다.
IDiscRecorder2 이 인터페이스는 물리적 디바이스를 나타냅니다. 이 인터페이스를 사용하여 컴퓨터에 설치된 CD 및 DVD 장치에 대한 정보를 검색하고 트레이를 닫거나 미디어를 꺼내는 등의 작업을 수행합니다.
IDiscRecorder2Ex 이 인터페이스는 물리적 디바이스를 나타냅니다.

IEnumFsiItems FsiDirectoryItem 개체에 대한 자식 디렉터리 및 파일 항목을 열거하려면 이 인터페이스를 사용합니다.
IEnumProgressItems 이 인터페이스를 사용하여 진행률 항목 컬렉션을 열거합니다.
IFileSystemImage 이 인터페이스를 사용하여 파일 시스템 이미지를 빌드하고, 세션 매개 변수를 설정하고, 이미지를 가져오거나 내보냅니다.
IFileSystemImage2 이 인터페이스를 사용하여 EFI/UEFI 지원에 필요한 여러 부팅 항목 또는 부팅 이미지를 작성합니다. 예를 들어 Windows XP 및 Windows Vista용 부팅 스트랩이 있는 부팅 미디어입니다.
IFileSystemImage3 이 인터페이스를 사용하여 UDF 파일 시스템(rev 2.50 이상)에서 메타데이터 및 메타데이터 미러 파일을 설정하거나 확인하여 중복성을 확인합니다.
IFileSystemImageResult 이 인터페이스를 사용하여 화상 이미지, 이미지 데이터 스트림 및 진행률 정보에 대한 정보를 가져옵니다.
IFileSystemImageResult2 IFileSystemImageResult2 인터페이스를 사용하면 데이터 레코더 개체가 다시 쓰기 가능한 디스크에 대해 만든 이미지에서 수정된 블록에 대한 정보를 검색할 수 있습니다.
IFsiDirectoryItem 이 인터페이스를 사용하여 파일 시스템 이미지에서 항목을 추가하거나 항목을 제거합니다.
IFsiDirectoryItem2 이 인터페이스를 사용하여 모든 하위 디렉터리, 파일 및 연결된 명명된 스트림을 포함하는 디렉터리 트리를 파일 시스템 이미지에 추가합니다.
IFsiFileItem 이 인터페이스를 사용하여 파일 콘텐츠의 파일 크기 및 데이터 스트림을 식별합니다.

IFsiFileItem2 이 인터페이스를 사용하여 파일과 연결된 명명된 스트림을 추가, 제거 및 열거합니다. 또한 이 인터페이스는 파일의 '실시간' 특성에 대한 액세스 권한을 제공합니다.
IFsiItem 파일 및 디렉터리 항목 모두에 공통적인 속성을 포함하는 기본 인터페이스입니다.
IFsiNamedStreams 이 인터페이스를 사용하여 파일 시스템 이미지의 파일과 연결된 명명된 스트림을 열거합니다.
IIsoImageManager 이 인터페이스를 사용하여 기존 .iso 파일에 굽기 위한 유효한 파일 시스템이 포함되어 있는지 확인합니다.
IJolietDiscMaster IJolietDiscMaster 인터페이스를 사용하면 CD 데이터 디스크를 스테이징할 수 있습니다.
IMultisession 파생 다중 분할 인터페이스에 공통적인 속성을 포함하는 기본 인터페이스입니다.
IMultisessionRandomWrite 이 인터페이스를 사용하여 임의 쓰기를 허용하고 실제 세션의 개념을 지원하지 않는 미디어의 현재 상태에 대한 정보를 검색합니다.
IMultisessionSequential 미디어에 이전 세션이 포함된 경우 이 인터페이스를 사용하여 순차적으로 녹화된 미디어에서 이전 가져오기 세션에 대한 정보를 검색합니다.
IMultisessionSequential2 이 인터페이스를 사용하여 순차적으로 녹화된 미디어에서 쓰기 가능한 단위의 크기에 대한 정보를 검색합니다.
IProgressItem 이 인터페이스를 사용하여 결과 파일 이미지의 한 세그먼트에 대한 블록 정보를 검색합니다.
IProgressItems 이 인터페이스를 사용하여 결과 이미지의 진행률 항목을 열거합니다.

IRawCDImageCreator 이 인터페이스를 사용하여 DAO(Disc-at-Once) 모드에서 CD 미디어에 서면으로 사용할 RAW CD 이미지를 만듭니다. 이 인터페이스를 사용하여 만든 이미지는 IDiscFormat2RawCD 인터페이스를 사용하여 CD 미디어에 쓸 수 있습니다.
IRawCDImageTrackInfo 이 인터페이스를 사용하여 CD 미디어에 적용되는 트랙별 속성을 추적합니다.
IRedbookDiscMaster IRedbookDiscMaster 인터페이스를 사용하면 오디오 CD 이미지를 스테이징할 수 있습니다. MSDiscMasterObj에서 지원하는 형식 중 하나를 나타내며, 한 번에 트랙 모드(고정 크기 오디오 간격)에서 멀티 트랙 오디오 디스크를 만들 수 있습니다.
IStreamConcatenate 이 인터페이스를 사용하여 여러 데이터 스트림을 단일 스트림으로 결합합니다.
IStreamInterleave 이 인터페이스를 사용하여 각 데이터 스트림의 일부를 번갈아 산재하여 여러 데이터 스트림을 단일 스트림으로 결합합니다.
IStreamPseudoRandomBased 이 인터페이스를 사용하여 의사 임의 데이터로 데이터가 초기화되는 읽기 전용 데이터 스트림을 생성합니다(암호화적으로 안전하지 않음). SetSize 메서드를 호출하여 스트림의 요청된 크기를 설정해야 합니다.
IWriteEngine2 이 인터페이스를 사용하여 디바이스에 데이터 스트림을 작성합니다.
IWriteEngine2EventArgs 이 인터페이스를 사용하여 현재 쓰기 작업에 대한 정보를 검색합니다. 이 인터페이스는 구현하는 DWriteEngine2Events::Update 메서드에 전달됩니다.
IWriteSpeedDescriptor 이 인터페이스는 디스크 레코더 및 현재 미디어(예: 미디어 유형, 쓰기 속도, 회전 속도 제어 유형)에서 지원하는 자세한 쓰기 구성을 검색합니다.

인덱싱 서비스

아티클2023. 08. 22.

인덱싱 서비스 기술 개요입니다.

인덱싱 서비스를 개발하려면 다음 헤더가 필요합니다.

- cmdtree.h
- filter.h
- indexsrv.h
- ntquery.h
- oledbguid.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 인덱싱 서비스

열거형

CHUNK_BREAKTYPE 현재 청크를 이전 청크와 구분하는 중단 유형을 설명합니다.
CHUNKSTATE 현재 청크가 텍스트 형식 속성인지 아니면 값 형식 속성인지를 지정합니다.
DBCOMMANDOPENUM DBCOMMANDOPENUM 열거형 형식에는 명령 트리의 노드에 대해 가능한 명령 연산자 목록이 포함되어 있습니다.
DBCOMMANDREUSEENUM DBCOMMANDREUSEENUM 열거형 형식은 이전 명령의 상태가 유지되는지 여부를 지정합니다.
DBKINDENUM DBKINDENUM 열거형 형식은 데이터베이스 개체를 식별하는 데 사용할 GUID, 속성 번호 또는 속성 이름의 조합을 지정합니다.
DBVALUEKINDENUM

DBVALUEKINDENUM 열거형 형식은 DBCOMMANDTREE 구조체 내의 공용 구조체 멤버 형식을 나타내는 데 사용됩니다.
IFILTER_FLAGS 호출자가 IPropertySetStorage 및 IPropertyStorage 인터페이스를 사용하여 추가 속성을 찾아야 하는지 여부를 나타냅니다.
IFILTER_INIT 필터링 프로세스를 제어하는 플래그입니다.
WORDREP_BREAK_TYPE 현재 단어를 이전 단어와 구분하는 중단 유형을 설명합니다.

Functions

BindIFilterFromStorage 지정된 스토리지 개체에 대한 IFilter 인터페이스 포인터를 검색합니다. 이는 문서의 내용을 필터링하고 IStorage 인터페이스를 통해 액세스할 수 있는 포함된 OLE 개체를 처리할 때 특히 유용합니다.
BindIFilterFromStream 지정된 스토리지 개체에 대한 IFilter 인터페이스 포인터를 검색합니다. 이는 문서의 내용을 필터링하고 IStream 인터페이스를 통해 액세스할 수 있는 포함된 OLE 개체를 처리할 때 특히 유용합니다.
BindRegion 개체의 지정된 부분을 나타내는 인터페이스를 검색합니다. 현재 향후 사용을 위해 예약되어 있습니다.
FindErrorNodes ICommandTree::FindErrorNodes 메서드는 명령 트리를 트래버스하고 오류가 있는 노드 배열을 반환합니다.
FreeCommandTree ICommandTree::FreeCommandTree 메서드는 명령 트리를 트래버스하고 해당 구조체의 모든 변형뿐만 아니라 모든 DBCOMMANDTREE 노드 구조체의 할당을 취소합니다. 그런 다음, 루트 포인터를 NULL 포인터로 설정합니다.

GetChunk GetChunk 메서드에 대한 첫 번째 호출인 경우 필터를 다음 청크의 시작 부분이나 첫 번째 청크에 배치하고 현재 청크에 대한 설명을 반환합니다.
GetCommandTree ICommandTree::GetCommandTree 메서드는 추가된 모든 사후 처리 작업을 포함하여 현재 명령을 트리로 에코합니다.
GetText CHUNKSTATE 열거형 값이 CHUNK_TEXT 현재 청크에서 텍스트(텍스트 형식 속성)를 검색합니다.
GetValue 청크에서 값(내부 값 형식 속성)을 검색합니다. 청크에는 CHUNK_VALUE CHUNKSTATE 열거형 값이 있어야 합니다.
Init 필터링 세션을 초기화합니다.
LoadIFilter 개체의 경로 이름에서 IFilter를 검색합니다.
SetCatalogState 백업 또는 기타 용도로 카탈로그 상태를 설정합니다.
SetCommandTree ICommandTree::SetCommandTree 메서드는 명령 개체의 명령 트리를 설정하여 기존 명령 트리를 바꾸거나 지정된 텍스트 명령을 ICommandText 인터페이스로 대체합니다.

인터페이스

ICommandTree ICommandTree 인터페이스는 명령을 지원하는 공급자에 대한 선택 사항입니다. 쿼리 트리를 조작하는 메서드가 포함되어 있습니다. 명령 트리를 지원하는 공급자는 ICommandText 인터페이스를 통해 동일한 기능을 지정하는 것도 지원해야 합니다.

IFilter 문서에서 텍스트 및 속성(특성이라고도 함)을 검색합니다.
IPhraseSink 쿼리 시간 동안 단어 분리기가 쿼리 텍스트에서 구문 분석하는 구를 처리합니다.
IQuery IQuery 인터페이스는 나중에 사용하도록 예약되어 있습니다.

구조체

CI_STATE 인덱싱 서비스 카탈로그의 현재 상태를 나타냅니다.
CIPROPERTYDEF 식별 이름, 형식 및 속성 식별자(ID) 정보를 나타냅니다.
DBBYGUID DBBYGUID 구조체는 노드에 대한 추가 정보를 제공합니다.
DBCOMMANDTREE DBCOMMANDTREE 구조는 이 참조의 데이터 조작 연산자 및 데이터 정의 연산자 섹션에 설명된 대로 OLE DB 명령 트리의 노드를 나타내는 데 사용되는 기본 데이터 구조입니다.
DBCONTENT DBCONTENT 구조체는 DBOP_content 연산자에 필요한 특정 정보를 나타냅니다.
DBCONTENTPROXIMITY DBCONTENTPROXIMITY 구조체는 DBOP_content_proximity 연산자에 필요한 특정 정보를 나타냅니다.
DBCONTENTSCOPE DBCONTENTSCOPE 구조체는 명령 트리에서 scope 인수를 전달하는 데 사용됩니다.

DBCONTENTTABLE DBCONTENTTABLE 구조체는 명령 트리의 컴퓨터 및 카탈로그 이름을 나타냅니다.
DBCONTENTVECTOR DBCONTENTVECTOR 구조체는 DBOP_content_vector_or 연산자에 필요한 특정 정보를 나타냅니다. (DBCONTENTVECTOR)
DBGROUPINFO DBCONTENTVECTOR 구조체는 DBOP_content_vector_or 연산자에 필요한 특정 정보를 나타냅니다. (DBGROUPINFO)
Dbid DBID 구조체는 데이터베이스 개체를 식별하는 다양한 방법을 캡슐화합니다.
DBLIKE DBLIKE 구조체는 DBOP_like 연산자에 필요한 특정 정보를 나타냅니다.
DBPARAMETER DBPARAMETER 구조체는 스칼라 매개 변수에 대한 값을 정의하는 데 사용됩니다.
DBPROBABILISTIC DBPROBABILISTIC 구조체는 나중에 사용하도록 예약되어 있습니다.
DBRELEVANTDOCUMENT DBRELEVANTDOCUMENT 구조체는 나중에 사용하도록 예약되어 있습니다.
DBSETFUNC DBSETFUNC 구조체는 선택 작업에 사용할 집계 함수를 지정합니다.
DBSORTINFO DBSORTINFO 구조체는 열이 정렬되는 순서(즉, 오름차순 또는 내림차순)를 저장합니다. 이 정보는 DBOP_sort_list_element 노드 내에 저장됩니다.
DBTEXT DBTEXT 구조체는 DBOP_text_command 노드에서 사용됩니다.
FILTERREGION

개체 내에서 지정된 텍스트 부분의 위치와 익스텐트를 설명합니다.
FULLPROPSPEC 속성 집합 및 속성 집합 내의 속성을 지정합니다.
STAT_CHUNK 청크의 특성을 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

잉크 입력

아티클2023. 11. 16.

잉크 입력 기술 개요입니다.

잉크 입력을 개발하려면 다음 헤더가 필요합니다.

- inkpresenterdesktop.h
- inkrenderer.h
- Shellhandwriting.h 헤더

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 잉크 입력

열거형

INK_HIGH_CONTRAST_ADJUSTMENT 시스템이 대비 테마 모드에 있을 때 IInkD2DRenderer2 개체가 잉크(표준 및 수정)를 그리는 방법을 지정합니다.
TfHandwritingFocusTargetResponse FocusHandwritingTarget 함수를 호출할 때 IHandwritingInputRoutingCallback 인터페이스를 구현하는 클라이언트가 응답하는 방법을 지정합니다.
TfHandwritingState 시스템에서 필기를 처리하는 방법을 지정합니다.
TfInputEvaluation 펜 입력을 인식하는 방법을 지정합니다.
TfProximateHandwritingTargetResponse 필기 대상 개체의 근접 위치에 따라 지원되는 필기 동작을 지정합니다.

Functions

CreateAndInitializeInkPresenter 애플리케이션 스레드에 IInkPresenterDesktop 개체를 만들고, 앱의 DirectComposition 시각적 트리에 연결하고, 개체의 크기를 설정합니다.
CreateInkPresenter 애플리케이션 스레드에 IInkPresenterDesktop 개체를 만듭니다.
DetermineProximateHandwritingTarget
그리기 잉크 스트로크를 앱의 지정된 Direct2D 디바이스 컨텍스트로 렌더링합니다.
그리기 잉크 스트로크를 앱의 지정된 Direct2D 디바이스 컨텍스트로 렌더링합니다.
FocusHandwritingTarget
GetHandwritingDistanceThreshold *targetScreenPoint*에서 필기 기능을 사용하도록 설정하는 유효한 편집 컨트롤에서 거리(세로 및 가로)를 픽셀 단위로 검색합니다.
GetHandwritingState TSF(Text Services Framework) 스레드 관리자의 현재 필기 상태를 검색합니다.
GetHandwritingStrokeIdForPointer 지정된 포인터 ID와 연결된 잉크 스트로크의 고유 ID를 검색합니다.
GetPointerTargetInfo 포인터 입력의 근접 대상에 대한 세부 정보를 검색합니다.
GetPointerTargetInfo 포인터 입력의 최상위 대상 창에 대한 세부 정보를 검색합니다.
GetSize InkPresenter 개체의 크기를 가져옵니다.
GetThreadIdForInput

입력 대상 UI에 대한 메시지 처리 스레드의 ID를 검색합니다.
Invoke IInkDesktopHost 개체 스레드에서 잉크 작업을 실행합니다.
OnCommitRequested 앱이 보류 중인 모든 Microsoft DirectComposition 명령을 앱의 DirectComposition 시각적 트리에 커밋하도록 요청합니다.
OnHighContrastChanged 고대비 변경 처리기를 지정합니다. 이 처리기는 고대비 시스템 설정의 변경 내용에 대해 알림을 받습니다.
QueueWorkItem IInkDesktopHost 스레드에서 실행하기 위해 작업 큐에 잉크 작업을 추가합니다.
RegisterHandwritingInputRoutingCallback
RequestHandwritingForPointer 지정한 포인터 및 잉크 스트로크를 사용하여 필기 환경을 제공하도록 요청합니다.
SetCommitRequestHandler 앱(IInkPresenterDesktop 개체 대신)이 보류 중인 모든 Microsoft DirectComposition 명령을 앱의 DirectComposition 시각적 트리에 커밋할 수 있도록 하는 IInkCommitRequestHandler 개체를 설정합니다.
SetHandwritingState TSF(Text Services Framework) 스레드 관리자의 현재 필기 상태를 설정합니다.
SetInputEvaluation 펜 입력을 인식하는 방법을 설정합니다.
SetResponse 근접 대상에 대한 콜백 구현의 응답을 지정합니다.
SetResponse 대상에 대한 콜백 구현의 응답을 지정합니다.

SetRootVisual 앱의 DirectComposition 시각적 트리에 대한 연결을 설정합니다.
SetSize InkPresenter 개체의 크기를 설정합니다.

인터페이스

IHandwritingInputRoutingCallback 펜 입력 처리 시스템 스레드를 입력 대상의 UI 스레드에서 분리하기 위한 입력 라우팅 지원을 제공합니다.
IInkCommitRequestHandler IInkCommitRequestHandler 개체를 사용하면 앱(IInkPresenterDesktop 개체 대신)이 보류 중인 모든 Microsoft DirectComposition 명령을 앱의 DirectComposition 시각적 트리에 커밋할 수 있습니다.
IInkD2DRenderer IInkD2DRenderer 개체를 사용하면 기본 InkCanvas 컨트롤 대신 유니버설 Windows 앱의 지정된 Direct2D 디바이스 컨텍스트에 잉크 스트로크를 렌더링할 수 있습니다.
IInkD2DRenderer2 IInkD2DRenderer2 개체를 사용하면 기본 InkCanvas 컨트롤 대신 유니버설 Windows 앱의 지정된 Direct2D 디바이스 컨텍스트에 잉크 스트로크를 렌더링할 수 있습니다.
IInkDesktopHost IInkDesktopHost 개체를 사용하면 앱 스레드 만들기를 통해 잉크 입력, 처리 및 렌더링을 통해 IInkPresenterDesktop 개체를 호스트하고 앱의 DirectComposition 시각적 트리에 삽입할 수 있습니다.
IInkHostWorkItem IInkHostWorkItem 개체는 IInkDesktopHost 개체 스레드에서 실행할 잉크 작업을 나타냅니다.
IInkPresenterDesktop

IInkPresenterDesktop 개체는 클래식 Windows 앱의 DirectComposition 시각적 트리에 구성하고 삽입할 수 있는 InkPresenter를 나타냅니다.
ITfDetermineProximateHandwritingTargetArgs 필기 응답 동작 및 근접 대상 정보를 제공합니다.
ITfFocusHandwritingTargetArgs 필기 응답 동작 및 대상 정보를 제공합니다.
ITfHandwriting 포커스를 편집 컨트롤로 설정하지 않고도 근접 편집 필드에서 펜으로 잉크 입력을 지원합니다. 잉크 스트로크가 렌더링되고, 필기 인식이 수행되고(텍스트 또는 편집 제스처로) 텍스트가 편집 필드에 입력됩니다.
ITfHandwritingRequest 애플리케이션은 이 인터페이스를 사용하여 필기 요청 후에 발생한 펜 입력을 평가했음을 시스템에 알려야 합니다.
ITfHandwritingSink 포커스가 편집 필드로 설정되기 전에 사용자 지정 필기 동작을 지정하는 기능을 지원합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

입력 피드백 구성

아티클2023. 08. 22.

입력 피드백 구성 기술 개요입니다.

입력 피드백 구성 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 입력 피드백 구성

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

입력 원본 식별

아티클2023. 08. 22.

입력 원본 식별 기술 개요입니다.

입력 소스 식별 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 입력 원본 식별

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

설치 가능한 파일 시스템 DDI 참조

아티클2023. 08. 22.

설치 가능한 파일 시스템 DDI 참조 기술 개요입니다.

설치 가능한 파일 시스템 DDI 참조를 개발하려면 다음 헤더가 필요합니다.

- fltuser.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 설치 가능한 파일 시스템 DDI 참조

Functions

FilterAttach FilterAttach 함수는 지정된 볼륨에 새 미니필터 instance 연결합니다.
FilterAttachAtAltitude FilterAttachAtAltitude 함수는 지정된 고도의 볼륨에 새 미니필터 instance 연결하여 미니필터의 INF(설정 정보) 파일의 모든 설정을 재정의하는 디버깅 지원 함수입니다.
FilterClose FilterClose 함수는 열린 미니필터 핸들을 닫습니다.
FilterConnectCommunicationPort FilterConnectCommunicationPort는 파일 시스템 미니필터에서 만든 통신 서버 포트에 대한 새 연결을 엽니다.
필터만들기 FilterCreate 함수는 지정된 미니필터에 대한 핸들을 만듭니다.
FilterDetach FilterDetach 함수는 지정된 미니 필터 instance 지정된 볼륨에서 분리합니다.
FilterFindClose FilterFindClose 함수는 지정된 미니필터 검색 핸들을 닫습니다. FilterFindFirst 및 FilterFindNext 함수는 이 검색 핸들을 사용하여 미니필터를 찾습니다.

FilterFindFirst FilterFindFirst 함수는 필터 드라이버(미니필터 드라이버 instance 또는 레거시 필터 드라이버)에 대한 정보를 반환하고 등록된 필터의 전역 목록에서 필터 검사를 시작하는 데 사용됩니다.
FilterFindNext FilterFindNext 함수는 FilterFindFirst 호출로 시작된 필터 검색을 계속합니다.
FilterGetDosName FilterGetDosName 함수는 지정된 볼륨 이름에 해당하는 MS-DOS 디바이스 이름을 반환합니다.
FilterGetInformation FilterGetInformation 함수는 미니필터에 대한 다양한 종류의 정보를 반환합니다.
FilterGetMessage FilterGetMessage 함수는 커널 모드 미니필터에서 메시지를 가져옵니다.
FilterInstanceClose FilterInstanceClose 함수는 FilterInstanceCreate에서 연 미니필터 instance 핸들을 닫습니다.
FilterInstanceCreate FilterInstanceCreate 함수는 지정된 미니필터 instance 통신하는 데 사용할 수 있는 핸들을 만듭니다.
FilterInstanceFindClose FilterInstanceFindClose 함수는 지정된 미니필터 instance 검색 핸들을 닫습니다. FilterInstanceFindFirst 및 FilterInstanceFindNext 함수는 이 검색 핸들을 사용하여 미니필터의 인스턴스를 찾습니다.
FilterInstanceFindFirst FilterInstanceFindFirst 함수는 instance 미니필터 드라이버에 대한 정보를 반환하며 미니필터 인스턴스를 검사하기 위한 시작점으로 사용됩니다.
FilterInstanceFindNext FilterInstanceFindNext 함수는 FilterInstanceFindFirst 호출로 시작된 검색 instance 미니필터 드라이버를 계속합니다.
FilterInstanceGetInformation FilterInstanceGetInformation 함수는 미니필터 instance 대한 다양한 종류의 정보를 반환합니다.

FilterLoad FilterLoad 함수는 미니필터 드라이버를 시스템에 동적으로 로드합니다.
FilterReplyMessage FilterReplyMessage 함수는 커널 모드 미니필터의 메시지에 회신합니다.
FilterSendMessage FilterSendMessage 함수는 커널 모드 미니필터에 메시지를 보냅니다.
FilterUnload FilterLoad를 호출하여 지원 미니필터를 로드한 애플리케이션은 FilterUnload 함수를 호출하여 미니필터를 언로드할 수 있습니다.
FilterVolumeFindClose FilterVolumeFindClose 함수는 지정된 볼륨 검색 핸들을 닫습니다. FilterVolumeFindFirst 및 FilterVolumeFindNext는 이 검색 핸들을 사용하여 볼륨을 찾습니다.
FilterVolumeFindFirst FilterVolumeFindFirst 함수는 볼륨에 대한 정보를 반환합니다.
FilterVolumeFindNext FilterVolumeFindNext 함수는 FilterVolumeFindFirst 호출로 시작된 볼륨 검색을 계속합니다.
FilterVolumeInstanceFindClose FilterVolumeInstanceFindClose 함수는 지정된 볼륨 instance 검색 핸들을 닫습니다. FilterVolumeInstanceFindFirst 및 FilterVolumeInstanceFindNext는 이 검색 핸들을 사용하여 볼륨에서 인스턴스를 찾습니다.
FilterVolumeInstanceFindFirst FilterVolumeInstanceFindFirst 함수는 미니필터 드라이버 instance 또는 레거시 필터 드라이버에 대한 정보를 반환하며 볼륨에 연결된 필터 드라이버 검사를 시작하는 데 사용됩니다.
FilterVolumeInstanceFindNext FilterVolumeInstanceFindNext 함수는 FilterVolumeInstanceFindFirst 호출로 시작된 미니필터 드라이버 instance 또는 레거시 필터 드라이버 검색을 계속합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

상호 작용 컨텍스트

상호 작용 컨텍스트 기술의 개요입니다.

상호 작용 컨텍스트를 개발하려면 다음 헤더가 필요합니다.

- interactioncontext.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 상호 작용 컨텍스트

열거 목록

테이블 확장

CROSS_SLIDE_FLAGS 슬라이드 간 상호 작용의 상태를 지정합니다.
CROSS_SLIDE_THRESHOLD 슬라이드 간 동작 임계값을 지정합니다.
HOLD_PARAMETER 누름 제스처와 관련된 다양한 값을 지정합니다.
INERTIA_PARAMETER 조작에 대한 관성 값(변환, 회전, 크기 조정)을 지정합니다.
INTERACTION_CONFIGURATION_FLAGS Interaction Context 개체를 구성할 때 사용할 상호 작용을 지정합니다.
INTERACTION_CONTEXT_PROPERTY Interaction Context 개체의 속성을 지정합니다.
INTERACTION_FLAGS 상호 작용의 상태를 지정합니다.
INTERACTION_ID

Interaction Context 개체를 구성하는 데 사용되는 상호 작용 상태를 지정합니다.
INTERACTION_STATE Interaction Context 개체의 상태를 지정합니다.
MANIPULATION_RAILS_STATE 상호 작용에 대한 레일 상태를 지정합니다.
MOUSE_WHEEL_PARAMETER 마우스 휠 회전에 매핑할 수 있는 조작을 지정합니다.
TAP_PARAMETER 탭 제스처와 관련된 다양한 값을 지정합니다.
TRANSLATION_PARAMETER 끌기 조작과 관련된 다양한 값을 지정합니다.

Functions

테이블 확장

AddPointerInteractionContext Interaction Context 개체에서 처리하는 포인터 집합에 지정된 포인터를 포함합니다.
BufferPointerPacketsInteractionContext 상호 작용 컨텍스트 개체의 버퍼에 단일 입력 포인터의 기록을 추가합니다.
CreateInteractionContext Interaction Context 개체를 만들고 초기화합니다.
DestroyInteractionContext 지정된 상호 작용 컨텍스트 개체를 삭제합니다.
GetCrossSlideParameterInteractionContext 슬라이드 간 상호 작용 동작을 가져옵니다.

GetHoldParameterInteractionContext 보류 상호 작용 동작을 가져옵니다.
GetInertiaParameterInteractionContext 조작의 관성 동작(변환, 회전, 크기 조정)을 가져옵니다.
GetInteractionConfigurationInteractionContext Interaction Context 개체에 대한 상호 작용 구성 상태를 가져옵니다.
GetMouseWheelParameterInteractionContext Interaction Context 개체의 마우스 휠 상태를 가져옵니다.
GetPropertyInteractionContext 상호 작용 컨텍스트 개체 속성을 가져옵니다.
GetStateInteractionContext 현재 상호 작용 컨텍스트 상태 및 컨텍스트가 유휴 상태로 돌아가는 시간을 가져옵니다.
GetTapParameterInteractionContext 탭 조작 동작을 가져옵니다.
GetTranslationParameterInteractionContext 끌기 상호 작용 동작을 가져옵니다.
INTERACTION_CONTEXT_OUTPUT_CALLBACK Interaction Context 개체에서 이벤트를 수신하는 콜백입니다.
INTERACTION_CONTEXT_OUTPUT_CALLBACK2 Interaction Context 개체에서 이벤트를 수신하는 콜백입니다.
ProcessBufferedPacketsInteractionContext 포인터 입력 프레임의 끝에서 버퍼링된 패킷을 처리합니다.
ProcessInertiaInteractionContext 관성 처리를 위해 Interaction Context 개체에 타이머 입력을 보냅니다.
ProcessPointerFramesInteractionContext

포인터 입력 프레임 집합을 처리합니다.
RegisterOutputCallbackInteractionContext 상호 작용 컨텍스트 개체에서 상호 작용 이벤트를 수신하도록 콜백을 등록합니다.
RegisterOutputCallbackInteractionContext2 상호 작용 컨텍스트 개체에서 상호 작용 이벤트를 수신하도록 콜백을 등록합니다.
RemovePointerInteractionContext Interaction Context 개체에서 처리되는 포인터 집합에서 지정된 포인터를 제거합니다.
ResetInteractionContext 상호 작용 상태, 상호 작용 구성 설정 및 모든 매개 변수를 초기 상태로 다시 설정합니다. 현재 상호 작용은 알림 없이 취소됩니다. 다음에 사용하기 전에 상호 작용 컨텍스트를 다시 구성해야 합니다.
SetCrossSlideParametersInteractionContext 슬라이드 간 상호 작용을 구성합니다.
SetHoldParameterInteractionContext 보류 상호 작용 동작을 설정합니다.
SetInertiaParameterInteractionContext 접촉이 해제된 후 조작(변환, 회전, 크기 조정)의 관성 동작을 구성합니다.
SetInteractionConfigurationInteractionContext 지정된 조작을 처리하도록 Interaction Context 개체를 구성합니다.
SetMouseWheelParameterInteractionContext 마우스 휠 입력에 대한 매개 변수 값을 설정합니다.
SetPivotInteractionContext 단일 입력 포인터를 사용하여 회전 조작을 위해 중심점과 중심점의 피벗 반경을 설정합니다.
SetPropertyInteractionContext 상호 작용 컨텍스트 개체 속성을 설정합니다.
SetTapParameterInteractionContext 탭 조작 동작을 설정합니다.

SetTranslationParameterInteractionContext 끌기 상호 작용 동작을 설정합니다.
StopInteractionContext 상호 작용 상태를 INTERACTION_STATE_IDLE 설정하고 모든 상호 작용 구성 설정 및 매개 변수를 그대로 둡니다.

구조

테이블 확장

CROSS_SLIDE_PARAMETER 교차 슬라이드 임계값 및 해당 거리 임계값을 정의합니다.
INTERACTION_ARGUMENTS_CROSS_SLIDE 슬라이드 간 상호 작용의 상태를 정의합니다.
INTERACTION_ARGUMENTS_MANIPULATION 조작 상태를 정의합니다.
INTERACTION_ARGUMENTS_TAP 탭 상호 작용의 상태를 정의합니다.
INTERACTION_CONTEXT_CONFIGURATION 상호 작용 동작을 사용하거나 사용하지 않도록 설정하거나 수정하는 Interaction Context 개체의 구성을 정의합니다.
INTERACTION_CONTEXT_OUTPUT Interaction Context 개체의 출력을 정의합니다.
INTERACTION_CONTEXT_OUTPUT2 Interaction Context 개체의 출력을 정의합니다.
MANIPULATION_TRANSFORM

조작에 대한 변환 데이터를 정의합니다.
MANIPULATION_VELOCITY 조작의 속도 데이터를 정의합니다.

Last updated on 2025. 11. 26.

Windows 애플리케이션에 대한 국제화

아티클2023. 03. 11.

Windows 애플리케이션 기술 국제화 개요입니다.

Windows 애플리케이션에 대한 국제화를 개발하려면 다음 헤더가 필요합니다.

- datetimeapi.h
- elscore.h
- gb18030.h
- imepad.h
- imm.h
- immdev.h
- msime.h
- msimeapi.h
- muiload.h
- spellcheck.h
- spellcheckprovider.h
- stringapiset.h
- usp10.h
- winnls.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 애플리케이션에 대한 국제화

열거형

CORRECTIVE_ACTION 맞춤법 오류에 대해 수행할 수정 작업의 유형을 식별합니다.
IMEUCT IMEWRD 구조의 사용자 주석 유형입니다.
NORM_FORM 지원되는 정규화 양식을 지정합니다.

SCRIPT_JUSTIFY 애플리케이션에서 근거를 구현하는 데 필요한 문자 모양 특성 정보를 정의합니다.
SYSGEOCLASS 지리적 위치 클래스를 지정합니다.
SYSGEOTYPE GetGeoInfo 또는 GetGeoInfoEx 함수에서 요청된 지리적 위치 정보의 형식을 정의합니다.
SYSNLS_FUNCTION NLS 함수 기능을 지정합니다.
WORDLIST_TYPE 맞춤법 검사기에서 사용하는 단어 목록 유형 중 하나를 식별합니다.

함수

추가 제공된 단어를 원래 사전의 일부인 것처럼 처리합니다.
add_SpellCheckerChanged SpellCheckerChanged 이벤트에 대한 이벤트 처리기(ISpellCheckerChangedEventHandler)를 추가합니다.
자동 고침 한 단어의 발생을 다른 단어로 바꿉니다.
확인 제공된 텍스트의 맞춤법을 검사하고 맞춤법 오류 컬렉션을 반환합니다. (ISpellCheckProvider.Check)
확인 제공된 텍스트의 맞춤법을 검사하고 맞춤법 오류 컬렉션을 반환합니다. (ISpellChecker.Check)

닫기 사전 파일을 닫습니다.
닫기 IFELanguage 개체를 종료합니다.
CompareString CompareString 함수(winnls.h)는 식별자로 지정된 로캘에 대해 두 개의 문자열을 비교합니다.
CompareStringA 식별자로 지정된 로캘에 대해 두 개의 문자열을 비교합니다. CompareString을 잘못 사용하면 애플리케이션의 보안이 손상됩니다. (CompareStringA)
CompareStringEx 이름으로 지정된 로캘에 대해 두 개의 유니코드(와이드 문자) 문자열을 비교합니다. CompareStringEx를 잘못 사용하면 애플리케이션의 보안이 손상됩니다.
CompareStringOrdinal 두 유니코드 문자열을 비교하여 이진 동등성을 테스트합니다.
CompareStringW CompareStringW(유니코드) 함수(stringapiset.h)는 식별자로 지정된 로캘에 대해 두 개의 문자열을 비교합니다.
ComprehensiveCheck ISpellCheckProvider::Check보다 더 철저한 방식으로 공급자 텍스트를 맞춤법 검사합니다.
ComprehensiveCheck 제공된 텍스트의 맞춤법을 ISpellChecker::Check보다 더 철저하게 검사하고 맞춤법 오류 컬렉션을 반환합니다.
ConvertDefaultLocale 기본 로캘 값을 실제 로캘 식별자로 변환합니다.
만들기 새 사전 파일을 만듭니다.

CreateIFECommonInstance IFECommon 인터페이스에 대한 포인터를 반환합니다.
CreateIFEDictionaryInstance IFEDictionary 인터페이스에 대한 포인터를 반환합니다.
CreateIFELanguageInstance IFELanguage 인터페이스에 대한 포인터를 반환합니다.
CreateSpellChecker 지정된 언어를 지원하는 맞춤법 검사기를 만듭니다.
CreateSpellCheckProvider 지정된 언어를 지원하는 맞춤법 검사 공급자가 구현하는 맞춤법 검사기를 만듭니다.
CreateUI 애플릿의 창 핸들, 스타일 및 크기를 가져오기 위해 IImePad에서 호출됩니다.
DeleteDictionary IME의 플러그 인 사전 목록에서 사전을 삭제합니다.
DisplayProperty 이 메서드는 Windows 8 시작 하 고 더 이상 지원 되지 않습니다.
EnumCalendarInfoA 지정된 로캘의 일정 정보를 열거합니다. 참고 일정 정보 외에도 일정 식별자를 받으려면 애플리케이션에서 EnumCalendarInfoEx 함수를 사용해야 합니다. (ANSI)
EnumCalendarInfoExA 식별자가 지정한 로캘의 일정 정보를 열거합니다. 참고 Windows Vista 이상에서만 실행되는 모든 애플리케이션은 이 함수에 대한 기본 설정으로 EnumCalendarInfoExEx를 사용해야 합니다. (ANSI)
EnumCalendarInfoExEx 이름으로 지정된 로캘의 일정 정보를 열거합니다. Windows Vista 이상에서만 실행되도록 설계된 경우 애플리케이션은 EnumCalendarInfo 또는 EnumCalendarInfoEx를 기본 설정으로 이 함수를 호출해야 합니다. 참고: 이 함수는 예를 들어 사용자 지정 로캘로 인해 릴리스 간에 변경되는 데이터를 열거할 수 있습니다. 애플리케이션이 데이터를 유지하거나 전송해야 하는 경우 영구 로캘 데이터 사용을 참조하세요.

EnumCalendarInfoExW 식별자가 지정한 로캘의 일정 정보를 열거합니다. 참고 Windows Vista 이상에서만 실행되는 모든 애플리케이션은 이 함수에 대한 기본 설정으로 EnumCalendarInfoExEx를 사용해야 합니다. (유니코드)
EnumCalendarInfoW 지정된 로캘의 일정 정보를 열거합니다. 참고 일정 정보 외에도 일정 식별자를 받으려면 애플리케이션에서 EnumCalendarInfoEx 함수를 사용해야 합니다. (유니코드)
EnumDateFormatsA 지정된 로캘에 사용할 수 있는 긴 날짜, 짧은 날짜 또는 연도/월 형식을 열거합니다. (ANSI)
EnumDateFormatsExA 지정된 로캘에 사용할 수 있는 긴 날짜, 짧은 날짜 또는 연도/월 형식을 열거합니다. 참고 Windows Vista 이상에서만 실행되는 모든 애플리케이션은 이 함수에 대한 기본 설정으로 EnumDateFormatsExEx를 사용해야 합니다. (ANSI)
EnumDateFormatsExEx 이름으로 지정된 로캘에 사용할 수 있는 긴 날짜, 짧은 날짜 또는 연도/월 형식을 열거합니다. Windows Vista 이상에서만 실행되도록 설계된 경우 애플리케이션은 EnumDateFormats 또는 EnumDateFormatsEx를 기본 설정으로 이 함수를 호출해야 합니다. 참고: 이 함수는 예를 들어 사용자 지정 로캘로 인해 릴리스 간에 변경되는 데이터를 열거할 수 있습니다. 애플리케이션이 데이터를 유지하거나 전송해야 하는 경우 영구 로캘 데이터 사용을 참조하세요.
EnumDateFormatsExW 지정된 로캘에 사용할 수 있는 긴 날짜, 짧은 날짜 또는 연도/월 형식을 열거합니다. 참고 Windows Vista 이상에서만 실행되는 모든 애플리케이션은 이 함수에 대한 기본 설정으로 EnumDateFormatsExEx를 사용해야 합니다. (유니코드)
EnumDateFormatsW 지정된 로캘에 사용할 수 있는 긴 날짜, 짧은 날짜 또는 연도/월 형식을 열거합니다. (유니코드)
EnumLanguageGroupLocalesA 지정된 언어 그룹의 로캘을 열거합니다. (ANSI)
EnumLanguageGroupLocalesW 지정된 언어 그룹의 로캘을 열거합니다. (유니코드)

EnumSystemCodePagesA 운영 체제에 설치되거나 지원되는 코드 페이지를 열거합니다. (ANSI)
EnumSystemCodePagesW 운영 체제에 설치되거나 지원되는 코드 페이지를 열거합니다. (유니코드)
EnumSystemGeoID 운영 체제에서 사용할 수 있는 지리적 위치 식별자(GEOID 형식)를 열거합니다.
EnumSystemGeoNames 운영 체제에서 사용할 수 있는 지리적 위치에 대한 두 글자 ISO(국제 표준화 기구) 3166-1 코드 또는 숫자 UN(UN) 시리즈 M, 번호 49(M.49) 코드를 열거합니다.
EnumSystemLanguageGroupsA 운영 체제에 설치되거나 지원되는 언어 그룹을 열거합니다. (ANSI)
EnumSystemLanguageGroupsW 운영 체제에 설치되거나 지원되는 언어 그룹을 열거합니다. (유니코드)
EnumSystemLocalesA 운영 체제에 설치되거나 지원되는 로캘을 열거합니다. 참고 상호 운용성을 위해 Microsoft는 새 로캘에 대한 로캘 식별자 대신 로캘 이름을 사용하도록 마이그레이션하기 때문에 애플리케이션에서 EnumSystemLocalesEx 함수를 EnumSystemLocales에 선호해야 합니다. Windows Vista 이상에서만 실행되는 모든 애플리케이션은 EnumSystemLocalesEx를 사용해야 합니다. (ANSI)
EnumSystemLocalesEx 운영 체제에 설치되거나 지원되는 로캘을 열거합니다. 참고: 애플리케이션은 Windows Vista 이상에서만 실행되도록 설계된 경우 EnumSystemLocales를 기본 설정으로 이 함수를 호출해야 합니다.
EnumSystemLocalesW 운영 체제에 설치되거나 지원되는 로캘을 열거합니다. 참고 상호 운용성을 위해 Microsoft는 새 로캘에 대한 로캘 식별자 대신 로캘 이름을 사용하도록 마이그레이션하기 때문에 애플리케이션에서 EnumSystemLocalesEx 함수를 EnumSystemLocales에 선호해야 합니다. Windows Vista 이상에서만 실행되는 모든 애플리케이션은 EnumSystemLocalesEx를 사용해야 합니다. (유니코드)

EnumTimeFormatsA 식별자가 지정한 로캘에 사용할 수 있는 시간 형식을 열거합니다. 참고 상호 운용성을 위해 Microsoft는 새 로캘에 대한 로캘 식별자 대신 로캘 이름을 사용하도록 마이그레이션하기 때문에 애플리케이션에서 EnumTimeFormatsEx 함수를 EnumTimeFormats에 선호해야 합니다. Windows Vista 이상에서만 실행되는 모든 애플리케이션은 EnumTimeFormatsEx를 사용해야 합니다. (ANSI)
EnumTimeFormatsEx 이름으로 지정된 로캘에 사용할 수 있는 시간 형식을 열거합니다. Windows Vista 이상에서만 실행되도록 설계된 경우 애플리케이션은 EnumTimeFormats를 기본 설정으로 이 함수를 호출해야 합니다. 참고: 이 함수는 예를 들어 사용자 지정 로캘로 인해 릴리스 간에 변경되는 데이터를 열거할 수 있습니다. 애플리케이션이 데이터를 유지하거나 전송해야 하는 경우 영구 로캘 데이터 사용을 참조하세요.
EnumTimeFormatsW 식별자가 지정한 로캘에 사용할 수 있는 시간 형식을 열거합니다. 참고 상호 운용성을 위해 Microsoft는 새 로캘에 대한 로캘 식별자 대신 로캘 이름을 사용하도록 마이그레이션하기 때문에 애플리케이션에서 EnumTimeFormatsEx 함수를 EnumTimeFormats에 선호해야 합니다. Windows Vista 이상에서만 실행되는 모든 애플리케이션은 EnumTimeFormatsEx를 사용해야 합니다. (유니코드)
EnumUILanguagesA 운영 체제에서 사용할 수 있는 사용자 인터페이스 언어를 열거하고 목록의 모든 언어로 콜백 함수를 호출합니다. (ANSI)
EnumUILanguagesW 운영 체제에서 사용할 수 있는 사용자 인터페이스 언어를 열거하고 목록의 모든 언어로 콜백 함수를 호출합니다. (유니코드)
ExistWord IFEDictionary에 지정된 단어가 있는지 여부를 확인합니다.
FILEMUIINFO_GET_CULTURE FILEMUIINFO 구조체의 dwLanguageNameOffset 멤버에서 언어 이름 오프셋 정보와 연결된 로캘 이름을 가져옵니다.
FILEMUIINFO_GET_MAIN_TYPEID FILEMUIINFO 구조체의 형식 식별자 크기 및 오프셋과 연결된 주 모듈 형식 배열 요소를 가져옵니다. 이 정보는 구조체의 dwTypeIDMainSize 및 dwTypeIDMainOffset 멤버에 제공됩니다.

FILEMUIINFO_GET_MAIN_TYPEIDS FILEMUIINFO 구조체의 dwTypeIDMainOffset 멤버에서 형식 식별자 오프셋 정보와 연결된 주 모듈 형식 배열을 가져옵니다.
FILEMUIINFO_GET_MAIN_TYPENAMES FILEMUIINFO 구조체의 dwTypeNameMainOffset 멤버에 있는 형식 이름 오프셋 정보와 연결된 주 모듈 이름 다중 문자열 배열을 가져옵니다.
FILEMUIINFO_GET_MUI_TYPEID FILEMUIINFO 구조체의 형식 식별자 크기 및 오프셋 정보와 연결된 MUI 모듈 형식 배열 요소를 가져옵니다. 이 정보는 구조체의 dwTypeIDMUISize 및 dwTypeIDMUIOffset 멤버에 제공됩니다.
FILEMUIINFO_GET_MUI_TYPEIDS FILEMUIINFO 구조체의 dwTypeIDMUIOffset 멤버에서 형식 식별자 오프셋 정보와 연결된 MUI 모듈 형식 배열을 가져옵니다.
FILEMUIINFO_GET_MUI_TYPENAMES FILEMUIINFO 구조체의 dwTypeNameMUIOffset 멤버에 있는 형식 이름 오프셋 정보와 연결된 MUI 모듈 이름 다중 문자열 배열을 가져옵니다.
FindNLSString 식별자로 지정된 로캘의 다른 유니코드 문자열에서 유니코드 문자열(와이드 문자) 또는 해당 문자열을 찾습니다. 이진 표현이 매우 다른 문자열은 동일하게 비교할 수 있으므로 이 함수는 특정 보안 문제를 제기할 수 있습니다. 자세한 내용은 보안 고려 사항:_International 기능의 비교 함수에 대한 설명을 참조하세요. 참고 상호 운용성을 위해 Microsoft는 새 로캘에 대한 로캘 식별자 대신 로캘 이름을 사용하도록 마이그레이션하기 때문에 애플리케이션에서 FindNLSStringEx 함수를 선호해야 합니다. FindNLSString은 사용자 지정 로캘을 지원하지만 대부분의 애플리케이션은 이러한 유형의 지원을 위해 FindNLSStringEx를 사용해야 합니다.
FindNLSStringEx 이름으로 지정된 로캘의 다른 유니코드 문자열에서 유니코드 문자열(와이드 문자) 또는 해당 문자열을 찾습니다. 이진 표현이 매우 다른 문자열은 동일하게 비교할 수 있으므로 이 함수는 특정 보안 문제를 제기할 수 있습니다. 자세한 내용은 보안 고려 사항:_International 기능의 비교 함수에 대한 설명을 참조하세요.
FindStringOrdinal 비언어적 비교를 위해 다른 유니코드 문자열에서 유니코드 문자열(와이드 문자)을 찾습니다.

FoldStringA 한 유니코드 문자열을 다른 유니코드 문자열에 매핑하여 지정된 변환을 수행합니다. (FoldStringA)
FoldStringW 한 유니코드 문자열을 다른 유니코드 문자열에 매핑하여 지정된 변환을 수행합니다. (FoldStringW)
FreeMUILibrary LoadMUILibrary에서 로드한 리소스 모듈에 대한 핸들을 해제합니다.
GEO_ENUMNAMEPROC EnumSystemGeoNames 함수에서 제공하는 열거된 지리적 위치 정보를 처리하는 애플리케이션 정의 콜백 함수입니다.
get_CorrectiveAction 맞춤법 오류에 대해 수행해야 하는 수정 작업을 나타냅니다.
get_Description 맞춤법 검사기 옵션에 대한 설명을 가져옵니다.
get_Heading 맞춤법 검사기 옵션의 제목을 가져옵니다.
get_Id 이 맞춤법 검사기 엔진의 식별자를 가져옵니다.
get_Id 맞춤법 검사기 옵션의 식별자를 가져옵니다.
get_Id 이 맞춤법 검사기 식별자를 가져옵니다.
get_Labels 맞춤법 검사기 옵션의 레이블 열거자를 가져옵니다.
get_LanguageTag 이 맞춤법 검사기 인스턴스에서 지원하는 BCP47 언어 태그를 가져옵니다. (ISpellCheckProvider.get_LanguageTag)

get_LanguageTag 이 맞춤법 검사기 인스턴스에서 지원하는 BCP47 언어 태그를 가져옵니다. (ISpellChecker.get_LanguageTag)
get_Length 잘못된 텍스트의 길이를 가져옵니다.
get_LocalizedName 이 맞춤법 검사기를 설명하는 사용자에게 표시하기에 적합한 텍스트를 가져옵니다. (ISpellCheckProvider.get_LocalizedName)
get_LocalizedName 이 맞춤법 검사기를 설명하는 사용자에게 표시하기에 적합한 텍스트를 가져옵니다. (ISpellChecker.get_LocalizedName)
get_OptionIds 맞춤법 검사기에서 인식되는 선언된 옵션 식별자를 모두 가져옵니다.
get_OptionIds 선언된 옵션 식별자를 모두 가져옵니다.
get_Replacement 수정 작업을 바꿀 때 대체 텍스트로 사용할 텍스트를 가져옵니다.
get_StartIndex 오류가 시작되는 선택된 텍스트의 위치를 가져옵니다.
get_SupportedLanguages 맞춤법 검사기에서 지원하는 언어/방언 집합을 가져옵니다.
get_SupportedLanguages 등록된 맞춤법 검사기에서 지원하는 언어/방언 집합을 가져옵니다.
GetACP 운영 체제에 대한 현재 Windows ANSI 코드 페이지 식별자를 검색합니다. ANSI API 함수(예: TEXTOut의 ANSI 버전)는 암시적으로 GetACP를 사용하여 유니코드로 또는 유니코드에서 텍스트를 번역합니다.

GetAppletIIDList 구현된 IImePadApplet 인터페이스를 열거하기 위해 IImePad 인터페이스에서 호출됩니다.
GetCalendarInfoA 식별자가 지정한 로캘의 달력에 대한 정보를 검색합니다. (ANSI)
GetCalendarInfoEx 이름으로 지정된 로캘의 일정 정보를 검색합니다. 참고: Windows Vista 이상에서만 실행되도록 설계된 경우 애플리케이션은 GetCalendarInfo를 기본 설정으로 이 함수를 호출해야 합니다. 참고: 이 함수는 예를 들어 사용자 지정 로캘로 인해 릴리스 간에 변경되는 데이터를 검색할 수 있습니다. 애플리케이션이 데이터를 유지하거나 전송해야 하는 경우 영구 로캘 데이터 사용을 참조하세요.
GetCalendarInfoW 식별자가 지정한 로캘의 달력에 대한 정보를 검색합니다. (유니코드)
GetConversion 입력 문자열(일반적으로 히라가나 문자 포함)을 변환된 문자열로 변환합니다.
GetConversionModeCaps IFELanguage 개체의 변환 모드 기능을 가져옵니다.
GetCPInfo 유효한 설치 또는 사용 가능한 코드 페이지에 대한 정보를 검색합니다. (GetCPInfo)
GetCPInfoExA 유효한 설치 또는 사용 가능한 코드 페이지에 대한 정보를 검색합니다. (GetCPInfoExA)
GetCPInfoExW 유효한 설치 또는 사용 가능한 코드 페이지에 대한 정보를 검색합니다. (GetCPInfoExW)
GetCurrencyFormatA 숫자 문자열의 형식을 식별자로 지정된 로캘의 통화 문자열로 지정합니다. (ANSI)

GetCurrencyFormatEx 숫자 문자열의 형식을 이름으로 지정된 로캘의 통화 문자열로 지정합니다. 참고: Windows Vista 이상에서만 실행되도록 설계된 경우 애플리케이션은 GetCurrencyFormat을 기본 설정으로 이 함수를 호출해야 합니다. 참고: 이 함수는 예를 들어 사용자 지정 로캘로 인해 릴리스 간에 변경되는 데이터의 서식을 지정할 수 있습니다. 애플리케이션이 데이터를 유지하거나 전송해야 하는 경우 영구 로캘 데이터 사용을 참조하세요.
GetCurrencyFormatW 숫자 문자열의 형식을 식별자로 지정된 로캘의 통화 문자열로 지정합니다. (유니코드)
GetDateFormatA 로캘 식별자가 지정한 로캘에 대한 날짜 문자열로 날짜 형식을 지정합니다. (ANSI)
GetDateFormatEx 날짜의 형식을 이름으로 지정된 로캘의 날짜 문자열로 지정합니다.
GetDateFormatW 로캘 식별자가 지정한 로캘에 대한 날짜 문자열로 날짜 형식을 지정합니다. (유니코드)
GetDictionariesInUse IME에서 사용 중인 IME 플러그 인 사전의 GUID(사전 ID) 목록을 생성 날짜 및 암호화 플래그와 함께 가져옵니다.
GetDurationFormat 시간 기간을 식별자로 지정된 로캘의 시간 문자열로 서식을 지정합니다.
GetDurationFormatEx 시간 기간을 이름으로 지정된 로캘의 시간 문자열로 서식을 지정합니다.
GetFileMUIInfo 파일에 대한 리소스 관련 정보를 검색합니다.
GetFileMUIPath 제공된 LN 파일과 연결된 모든 언어별 리소스 파일의 경로를 검색합니다. 애플리케이션은 각 리소스 파일에 대한 경로를 가져오기 위해 이 함수를 반복적으로 호출해야 합니다.
GetGeoInfoA 지정된 지리적 위치에 대한 정보를 검색합니다. (ANSI)

GetGeoInfoEx ISO(International Organization for Standardization) 3166-1 코드 또는 숫자 UN(UN) Series M, Number 49(M.49) 코드를 사용하여 지정한 지리적 위치에 대한 정보를 검색합니다.
GetGeoInfoW 지정된 지리적 위치에 대한 정보를 검색합니다. (유니코드)
GetHeader 사전을 열지 않고 사전 파일에서 사전 헤더를 가져옵니다.
GetJMorphResult 형태 분석 결과를 가져옵니다.
GetLocaleInfoA 식별자가 지정한 로캘에 대한 정보를 검색합니다. (ANSI)
GetLocaleInfoEx 이름으로 지정된 로캘에 대한 정보를 검색합니다. Windows Vista 이상에서만 실행되도록 설계된 경우 애플리케이션은 GetLocaleInfo에 대한 기본 설정으로 이 함수를 호출해야 합니다. 참고: 이 함수는 예를 들어 사용자 지정 로캘로 인해 릴리스 간에 변경되는 데이터를 검색할 수 있습니다. 애플리케이션이 데이터를 유지하거나 전송해야 하는 경우 영구 로캘 데이터 사용을 참조하세요.
GetLocaleInfoW 식별자가 지정한 로캘에 대한 정보를 검색합니다. (유니코드)
GetNLSVersion 식별자가 지정한 로캘에 대해 지정된 NLS 기능의 현재 버전에 대한 정보를 검색합니다. 참고 상호 운용성을 위해 Microsoft는 새 로캘에 대한 로캘 식별자 대신 로캘 이름을 사용하도록 마이그레이션하기 때문에 애플리케이션에서 GetNLSVersionEx 함수를 GetNLSVersion에 선호해야 합니다. 이 권장 사항은 특히 GetNLSVersionEx가 정렬 동작이 변경되었는지 확인하기에 충분한 정보를 검색하는 사용자 지정 로캘에 적용됩니다. Windows Vista 이상에서만 실행되는 모든 애플리케이션은 GetNLSVersionEx를 사용하거나 GetNLSVersion을 호출하여 추가 정렬 버전 관리 데이터를 가져올 때 적어도 NLSVERSIONINFOEX 구조를 전달해야 합니다.
GetNLSVersionEx 이름으로 지정된 로캘에 대해 지정된 NLS 기능의 현재 버전에 대한 정보를 검색합니다. 참고: Windows Vista 이상에서만 실행되도록 설계된 경우 애플리케이션은 GetNLSVersion을 기본 설정으로 이 함수를 호출해야 합니다.

GetNumberFormatA 숫자 문자열의 서식을 식별자로 지정된 로캘에 대해 사용자 지정된 숫자 문자열로 지정합니다. (ANSI)
GetNumberFormatEx 숫자 문자열의 서식을 이름으로 지정된 로캘에 대해 사용자 지정된 숫자 문자열로 지정합니다. Windows Vista 이상에서만 실행되도록 설계된 경우 애플리케이션은 GetNumberFormat을 기본 설정으로 이 함수를 호출해야 합니다. 참고: 이 함수는 예를 들어 사용자 지정 로캘로 인해 릴리스 간에 변경되는 데이터의 서식을 지정할 수 있습니다. 애플리케이션이 데이터를 유지하거나 전송해야 하는 경우 영구 로캘 데이터 사용을 참조하세요.
GetNumberFormatW 숫자 문자열의 서식을 식별자로 지정된 로캘에 대해 사용자 지정된 숫자 문자열로 지정합니다. (유니코드)
GetOEMCP 운영 체제의 현재 OEM(원본 장비 제조업체) 코드 페이지 식별자를 반환합니다.
GetOptionDescription 특정 옵션의 정보(ID, 설명, 제목 및 레이블)를 검색합니다. (ISpellCheckProvider.GetOptionDescription)
GetOptionDescription 특정 옵션의 정보(ID, 설명, 제목 및 레이블)를 검색합니다. (ISpellChecker.GetOptionDescription)
GetOptionValue 지정된 옵션과 연결된 값을 검색합니다. (ISpellCheckProvider.GetOptionValue)
GetOptionValue 지정된 옵션과 연결된 값을 검색합니다. (ISpellChecker.GetOptionValue)
GetPosTable 공용 POS(음성의 일부) 테이블을 가져옵니다.
GetProcessPreferredUILanguages 프로세스 기본 설정 UI 언어를 검색합니다. 자세한 내용은 사용자 인터페이스 언어 관리를 참조하세요.

GetStringScripts 지정된 유니코드 문자열에 사용되는 스크립트 목록을 제공합니다.
GetStringTypeA 더 이상 사용되지 않습니다. (GetStringTypeA)
GetStringTypeExW 지정된 소스 문자열의 문자에 대한 문자 형식 정보를 검색합니다.
GetStringTypeW 지정된 유니코드 소스 문자열의 문자에 대한 문자 형식 정보를 검색합니다.
GetSystemDefaultLangID 시스템 로캘의 언어 식별자를 반환합니다.
GetSystemDefaultLCID 시스템 로캘의 로캘 식별자를 반환합니다. 참고 Windows Vista 이상에서만 실행되는 모든 애플리케이션은 이 함수에 대한 기본 설정으로 GetSystemDefaultLocaleName을 사용해야 합니다.
GetSystemDefaultLocaleName 시스템 기본 로캘 이름을 검색합니다. 참고: 애플리케이션은 이 함수보다 기본 설정으로 GetUserDefaultLocaleName을 호출하는 것이 좋습니다.
GetSystemDefaultUILanguage Windows Vista 이상에서 "설치 언어"라고도 하는 운영 체제의 시스템 기본 UI 언어에 대한 언어 식별자를 검색합니다. 자세한 내용은 사용자 인터페이스 언어 관리를 참조하세요.
GetSystemPreferredUILanguages 시스템 기본 설정 UI 언어를 검색합니다. 자세한 내용은 사용자 인터페이스 언어 관리를 참조하세요.
GetTextCharset 현재 지정된 디바이스 컨텍스트로 선택된 글꼴의 문자 집합 식별자를 검색합니다.
GetTextCharsetInfo 현재 지정된 디바이스 컨텍스트로 선택된 글꼴의 문자 집합에 대한 정보를 검색합니다.

GetThreadLocale 호출 스레드에 대한 현재 로캘의 로캘 식별자를 반환합니다. 참고: 이 함수는 예를 들어 사용자 지정 로캘로 인해 릴리스 간에 변경되는 데이터를 검색할 수 있습니다.
GetThreadPreferredUILanguages 현재 스레드에 대한 스레드 기본 설정 UI 언어를 검색합니다. 자세한 내용은 사용자 인터페이스 언어 관리를 참조하세요.
GetThreadUILanguage 현재 스레드에 대한 첫 번째 사용자 인터페이스 언어의 언어 식별자를 반환합니다.
GetTimeFormatA 시간을 식별자로 지정된 로캘의 시간 문자열로 서식을 지정합니다. 함수는 지정된 시간 또는 로컬 시스템 시간의 형식을 지정합니다. (ANSI)
GetTimeFormatEx 시간을 이름으로 지정된 로캘의 시간 문자열로 서식을 지정합니다.
GetTimeFormatW 시간을 식별자로 지정된 로캘의 시간 문자열로 서식을 지정합니다. 함수는 지정된 시간 또는 로컬 시스템 시간의 형식을 지정합니다. (유니코드)
GetUILanguageFallbackList 언어 이름으로 표시되는 UI 언어의 대체 목록을 가져옵니다.
GetUILanguageInfo 설치된 UI 언어에 대한 다양한 정보를 검색합니다.
GetUserDefaultGeoName 사용자의 기본 지리적 위치에 대한 두 글자로 된 ISO(International Organization for Standardization) 3166-1 코드 또는 숫자 UN(UN) Series M, Number 49(M.49) 코드를 검색합니다.
GetUserDefaultLangID 현재 사용자에 대한 지역 형식 설정의 언어 식별자를 반환합니다.
GetUserDefaultLCID 사용자 기본 로캘의 로캘 식별자를 반환합니다. 주의 사용자 기본 로캘이 사용자 지정 로캘인 경우 애플리케이션은 데이터를 값으로 정확하게 태그 지정하거나 교환할 수 없습니다.

GetUserDefaultLocaleName 사용자 기본 로캘 이름을 검색합니다. Windows Vista 이상에서만 실행되도록 설계된 경우 애플리케이션은 GetUserDefaultLCID를 기본 설정으로 이 함수를 호출해야 합니다.
GetUserDefaultUILanguage 현재 사용자의 사용자 UI 언어에 대한 언어 식별자를 반환합니다.
GetUserGeoID 사용자의 지리적 위치에 대한 정보를 검색합니다. 자세한 내용은 지리적 위치 표를 참조하세요.
GetUserPreferredUILanguages 사용자 기본 설정 UI 언어에 대한 정보를 검색합니다. 자세한 내용은 사용자 인터페이스 언어 관리를 참조하세요.
GetWords 사전에서 단어 항목을 가져옵니다.
IdnToAscii IDN(국제화된 도메인 이름) 또는 다른 국제화된 레이블을 Punycode 전송 인코딩 구문의 이름을 나타내는 ASCII 문자열의 유니코드(와이드 문자) 표현으로 변환합니다.
IdnToNameprepUnicode IDN(국제화된 도메인 이름) 또는 다른 국제화된 레이블을 네트워크 작업 그룹 RFC 3491에서 지정한 NamePrep 양식으로 변환하지만 Punycode로의 추가 변환은 수행하지 않습니다.
IdnToUnicode IDN(국제화된 도메인 이름) 또는 다른 국제화된 레이블의 Punycode 형식을 일반 유니코드 UTF-16 인코딩 구문으로 변환합니다.
무시 이 세션의 나머지 부분에 대해 제공된 단어를 무시합니다.
IMCENUMPROC IMCENUMPROC(imm.h)는 ImmEnumInputContext 함수에서 제공하는 입력 컨텍스트를 처리하는 애플리케이션 정의 콜백 함수입니다.

IMCENUMPROC IMCENUMPROC(immdev.h)는 ImmEnumInputContext 함수에서 제공하는 입력 컨텍스트를 처리하는 애플리케이션 정의 콜백 함수입니다.
ImmAssociateContext ImmAssociateContext 함수(imm.h)는 지정된 입력 컨텍스트를 지정된 창에 연결합니다.
ImmAssociateContext ImmAssociateContext 함수(immdev.h)는 지정된 입력 컨텍스트를 지정된 창과 연결합니다.
ImmAssociateContextEx ImmAssociateContextEx 함수(imm.h)는 입력 메서드 컨텍스트와 지정된 창 또는 해당 자식 간의 연결을 변경합니다.
ImmAssociateContextEx ImmAssociateContextEx 함수(immdev.h)는 입력 메서드 컨텍스트와 지정된 창 또는 자식 간의 연결을 변경합니다.
ImmConfigureIMEA ImmConfigureIMEA(ANSI) 함수(imm.h)는 지정된 입력 로캘 식별자의 IME에 대한 구성 대화 상자를 표시합니다.
ImmConfigureIMEA ImmConfigureIMEA(ANSI) 함수(immdev.h)는 지정된 입력 로캘 식별자의 IME에 대한 구성 대화 상자를 표시합니다.
ImmConfigureIMEW ImmConfigureIMEW(유니코드) 함수(imm.h)는 지정된 입력 로캘 식별자의 IME에 대한 구성 대화 상자를 표시합니다.
ImmConfigureIMEW ImmConfigureIMEW(유니코드) 함수(immdev.h)는 지정된 입력 로캘 식별자의 IME에 대한 구성 대화 상자를 표시합니다.
ImmCreateContext ImmCreateContext 함수(imm.h)는 컨텍스트에 대한 메모리를 할당하고 초기화하는 새 입력 컨텍스트를 만듭니다.

ImmCreateContext ImmCreateContext 함수(immdev.h)는 컨텍스트에 대한 메모리를 할당하고 초기화하는 새 입력 컨텍스트를 만듭니다.
ImmDestroyContext ImmDestroyContext 함수(imm.h)는 입력 컨텍스트를 해제하고 연결된 메모리를 해제합니다.
ImmDestroyContext ImmDestroyContext 함수(immdev.h)는 입력 컨텍스트를 해제하고 연결된 메모리를 해제합니다.
ImmDisableIME ImmDisableIME 함수(imm.h)는 스레드 또는 프로세스의 모든 스레드에 대해 IME를 사용하지 않도록 설정합니다.
ImmDisableIME ImmDisableIME 함수(immdev.h)는 스레드 또는 프로세스의 모든 스레드에 대해 IME를 사용하지 않도록 설정합니다.
ImmDisableLegacyIME ImmDisableLegacyIME 함수(imm.h)는 이 스레드가 Windows 스토어 앱 UI 스레드임을 나타냅니다.
ImmDisableLegacyIME ImmDisableLegacyIME 함수(immdev.h)는 이 스레드가 Windows 스토어 앱 UI 스레드임을 나타냅니다.
ImmDisableTextFrameService Windows Vista에서는 ImmDisableTextFrameService 함수(imm.h)를 더 이상 사용할 수 없습니다.
ImmDisableTextFrameService Windows Vista에서는 ImmDisableTextFrameService 함수(immdev.h)를 더 이상 사용할 수 없습니다.
ImmEnumInputContext ImmEnumInputContext 함수(imm.h)는 지정된 스레드에 대한 입력 컨텍스트를 검색합니다.
ImmEnumInputContext ImmEnumInputContext 함수(immdev.h)는 지정된 스레드에 대한 입력 컨텍스트를 검색합니다.

ImmEnumRegisterWordA ANSI(ImmEnumRegisterWordA) 함수(imm.h)는 지정된 읽기 문자열, 스타일 및 레지스터 문자열을 갖는 레지스터 문자열을 열거합니다.
ImmEnumRegisterWordA ImmEnumRegisterWordA(ANSI) 함수(immdev.h)는 지정된 읽기 문자열, 스타일 및 레지스터 문자열을 갖는 레지스터 문자열을 열거합니다.
ImmEnumRegisterWordW ImmEnumRegisterWordW(유니코드) 함수(imm.h)는 지정된 읽기 문자열, 스타일 및 레지스터 문자열을 갖는 레지스터 문자열을 열거합니다.
ImmEnumRegisterWordW ImmEnumRegisterWordW(유니코드) 함수(immdev.h)는 지정된 읽기 문자열, 스타일 및 레지스터 문자열을 갖는 레지스터 문자열을 열거합니다.
ImmEscapeA ImmEscapeA(ANSI) 함수(imm.h)는 다른 IME API 함수를 통해 사용할 수 없는 특정 IME의 기능에 액세스합니다.
ImmEscapeA ImmEscapeA(ANSI) 함수(immdev.h)는 다른 IME API 함수를 통해 사용할 수 없는 특정 IME의 기능에 액세스합니다.
ImmEscapeW ImmEscapeW(유니코드) 함수(imm.h)는 다른 IME API 함수를 통해 사용할 수 없는 특정 IME의 기능에 액세스합니다.
ImmEscapeW ImmEscapeW(유니코드) 함수(immdev.h)는 다른 IME API 함수를 통해 사용할 수 없는 특정 IME의 기능에 액세스합니다.
ImmGetCandidateListA ImmGetCandidateListA(ANSI) 함수(imm.h)는 후보 목록을 검색합니다.
ImmGetCandidateListA ImmGetCandidateListA(ANSI) 함수(immdev.h)는 후보 목록을 검색합니다.

ImmGetCandidateListCountA ImmGetCandidateListCountA(ANSI) 함수(imm.h)는 후보 목록의 크기를 검색합니다.
ImmGetCandidateListCountA ImmGetCandidateListCountA(ANSI) 함수(immdev.h)는 후보 목록의 크기를 검색합니다.
ImmGetCandidateListCountW ImmGetCandidateListCountW(유니코드) 함수(imm.h)는 후보 목록의 크기를 검색합니다.
ImmGetCandidateListCountW ImmGetCandidateListCountW(유니코드) 함수(immdev.h)는 후보 목록의 크기를 검색합니다.
ImmGetCandidateListW ImmGetCandidateListW(유니코드) 함수(imm.h)는 후보 목록을 검색합니다.
ImmGetCandidateListW ImmGetCandidateListW(유니코드) 함수(immdev.h)는 후보 목록을 검색합니다.
ImmGetCandidateWindow ImmGetCandidateWindow 함수(imm.h)는 후보 창에 대한 정보를 검색합니다.
ImmGetCandidateWindow ImmGetCandidateWindow 함수(immdev.h)는 후보 창에 대한 정보를 검색합니다.
ImmGetCompositionFontA ImmGetCompositionFontA(ANSI) 함수(imm.h)는 컴퍼지션 창에 문자를 표시하는 데 사용되는 논리적 글꼴에 대한 정보를 검색합니다.
ImmGetCompositionFontA ANSI(ImmGetCompositionFontA) 함수(immdev.h)는 컴퍼지션 창에 문자를 표시하는 데 현재 사용되는 논리 글꼴에 대한 정보를 검색합니다.
ImmGetCompositionFontW ImmGetCompositionFontW(유니코드) 함수(imm.h)는 컴퍼지션 창에 문자를 표시하는 데 사용되는 논리 글꼴에 대한 정보를 검색합니다.

ImmGetCompositionFontW ImmGetCompositionFontW(유니코드) 함수(immdev.h)는 컴퍼지션 창에 문자를 표시하는 데 현재 사용되는 논리 글꼴에 대한 정보를 검색합니다.
ImmGetCompositionStringA ImmGetCompositionStringA(ANSI) 함수(imm.h)는 컴퍼지션 문자열에 대한 정보를 검색합니다.
ImmGetCompositionStringA ImmGetCompositionStringA(ANSI) 함수(immdev.h)는 컴퍼지션 문자열에 대한 정보를 검색합니다.
ImmGetCompositionStringW ImmGetCompositionStringW(유니코드) 함수(imm.h)는 컴퍼지션 문자열에 대한 정보를 검색합니다.
ImmGetCompositionStringW ImmGetCompositionStringW(유니코드) 함수(immdev.h)는 컴퍼지션 문자열에 대한 정보를 검색합니다.
ImmGetCompositionWindow ImmGetCompositionWindow 함수(imm.h)는 컴퍼지션 창에 대한 정보를 검색합니다.
ImmGetCompositionWindow ImmGetCompositionWindow 함수(immdev.h)는 컴퍼지션 창에 대한 정보를 검색합니다.
ImmGetContext ImmGetContext 함수(imm.h)는 지정된 창과 연결된 입력 컨텍스트를 반환합니다.
ImmGetContext ImmGetContext 함수(immdev.h)는 지정된 창과 연결된 입력 컨텍스트를 반환합니다.
ImmGetConversionListA ImmGetConversionListA(ANSI) 함수(imm.h)는 IME 관련 메시지를 생성하지 않고 문자 또는 단어의 변환 결과 목록을 검색합니다.
ImmGetConversionListA ImmGetConversionListA(ANSI) 함수(immdev.h)는 IME 관련 메시지를 생성하지 않고 문자 또는 단어의 변환 결과 목록을 검색합니다.

ImmGetConversionListW ImmGetConversionListW(유니코드) 함수(imm.h)는 IME 관련 메시지를 생성하지 않고 문자 또는 단어의 변환 결과 목록을 검색합니다.
ImmGetConversionListW ImmGetConversionListW(유니코드) 함수(immdev.h)는 IME 관련 메시지를 생성하지 않고 문자 또는 단어의 변환 결과 목록을 검색합니다.
ImmGetConversionStatus ImmGetConversionStatus 함수(imm.h)는 현재 변환 상태를 검색합니다.
ImmGetConversionStatus ImmGetConversionStatus 함수(immdev.h)는 현재 변환 상태를 검색합니다.
ImmGetDefaultIMEWnd ImmGetDefaultIMEWnd 함수(imm.h)는 IME 클래스에 대한 기본 창 핸들을 검색합니다.
ImmGetDefaultIMEWnd ImmGetDefaultIMEWnd 함수(immdev.h)는 IME 클래스에 대한 기본 창 핸들을 검색합니다.
ImmGetDescriptionA ImmGetDescriptionA(ANSI) 함수(imm.h)는 IME에 대한 설명을 지정된 버퍼에 복사합니다.
ImmGetDescriptionA ImmGetDescriptionA(ANSI) 함수(immdev.h)는 IME에 대한 설명을 지정된 버퍼에 복사합니다.
ImmGetDescriptionW ImmGetDescriptionW(유니코드) 함수(imm.h)는 IME에 대한 설명을 지정된 버퍼에 복사합니다.
ImmGetDescriptionW ImmGetDescriptionW(유니코드) 함수(immdev.h)는 IME에 대한 설명을 지정된 버퍼에 복사합니다.
ImmGetGuideLineA ImmGetGuideLineA(ANSI) 함수(imm.h)는 오류에 대한 정보를 검색합니다. 애플리케이션은 사용자 알림에 대한 정보를 사용합니다.

ImmGetGuideLineA ImmGetGuideLineA(ANSI) 함수(immdev.h)는 오류에 대한 정보를 검색합니다. 애플리케이션은 사용자 알림에 대한 정보를 사용합니다.
ImmGetGuideLineW ImmGetGuideLineW(유니코드) 함수(imm.h)는 오류에 대한 정보를 검색합니다. 애플리케이션은 사용자 알림에 대한 정보를 사용합니다.
ImmGetGuideLineW ImmGetGuideLineW(유니코드) 함수(immdev.h)는 오류에 대한 정보를 검색합니다. 애플리케이션은 사용자 알림에 대한 정보를 사용합니다.
ImmGetIMEFileNameA ImmGetIMEFileNameA(ANSI) 함수(imm.h)는 지정된 입력 로캘과 연결된 IME의 파일 이름을 검색합니다.
ImmGetIMEFileNameA ImmGetIMEFileNameA(ANSI) 함수(immdev.h)는 지정된 입력 로캘과 연결된 IME의 파일 이름을 검색합니다.
ImmGetIMEFileNameW ImmGetIMEFileNameW(유니코드) 함수(imm.h)는 지정된 입력 로캘과 연결된 IME의 파일 이름을 검색합니다.
ImmGetIMEFileNameW ImmGetIMEFileNameW(유니코드) 함수(immdev.h)는 지정된 입력 로캘과 연결된 IME의 파일 이름을 검색합니다.
ImmGetImeMenuItemsA ImmGetImeMenuItemsA(ANSI) 함수(imm.h)는 지정된 입력 컨텍스트의 IME 메뉴에 등록된 메뉴 항목을 검색합니다.
ImmGetImeMenuItemsA ImmGetImeMenuItemsA(ANSI) 함수(immdev.h)는 지정된 입력 컨텍스트의 IME 메뉴에 등록된 메뉴 항목을 검색합니다.
ImmGetImeMenuItemsW ImmGetImeMenuItemsW(유니코드) 함수(imm.h)는 지정된 입력 컨텍스트의 IME 메뉴에 등록된 메뉴 항목을 검색합니다.

ImmGetImeMenuItemsW ImmGetImeMenuItemsW(유니코드) 함수(immdev.h)는 지정된 입력 컨텍스트의 IME 메뉴에 등록된 메뉴 항목을 검색합니다.
ImmGetOpenStatus ImmGetOpenStatus 함수(imm.h)는 IME가 열려 있는지 닫혔는지 여부를 결정합니다.
ImmGetOpenStatus ImmGetOpenStatus 함수(immdev.h)는 IME가 열려 있는지 아니면 닫혀 있는지를 결정합니다.
ImmGetProperty ImmGetProperty 함수(imm.h)는 지정된 입력 로캘과 연결된 IME의 속성 및 기능을 검색합니다.
ImmGetProperty ImmGetProperty 함수(immdev.h)는 지정된 입력 로캘과 연결된 IME의 속성 및 기능을 검색합니다.
ImmGetRegisterWordStyleA ImmGetRegisterWordStyleA(ANSI) 함수(imm.h)는 지정된 입력 로캘과 연결된 IME에서 지원하는 스타일 목록을 검색합니다.
ImmGetRegisterWordStyleA ImmGetRegisterWordStyleA(ANSI) 함수(immdev.h)는 지정된 입력 로캘과 연결된 IME에서 지원하는 스타일 목록을 검색합니다.
ImmGetRegisterWordStyleW ImmGetRegisterWordStyleW(유니코드) 함수(imm.h)는 지정된 입력 로캘과 연결된 IME에서 지원하는 스타일 목록을 검색합니다.
ImmGetRegisterWordStyleW ImmGetRegisterWordStyleW(유니코드) 함수(immdev.h)는 지정된 입력 로캘과 연결된 IME에서 지원하는 스타일 목록을 검색합니다.
ImmGetStatusWindowPos ImmGetStatusWindowPos 함수(imm.h)는 상태 창의 위치를 검색합니다.
ImmGetStatusWindowPos ImmGetStatusWindowPos 함수(immdev.h)는 상태 창의 위치를 검색합니다.

ImmGetVirtualKey ImmGetVirtualKey 함수(imm.h)는 IME가 이미 처리한 키 입력 메시지와 연결된 원래 가상 키 값을 검색합니다.
ImmGetVirtualKey ImmGetVirtualKey 함수(immdev.h)는 IME가 이미 처리한 키 입력 메시지와 연결된 원래 가상 키 값을 검색합니다.
ImmInstallIMEA ImmInstallIMEA(ANSI) 함수(imm.h)는 IME를 설치합니다.
ImmInstallIMEA ImmInstallIMEA(ANSI) 함수(immdev.h)는 IME를 설치합니다.
ImmInstallIMEW ImmInstallIMEW(유니코드) 함수(imm.h)는 IME를 설치합니다.
ImmInstallIMEW ImmInstallIMEW(유니코드) 함수(immdev.h)는 IME를 설치합니다.
ImmIsIME ImmIsIME 함수(imm.h)는 지정된 입력 로캘에 IME가 있는지 여부를 결정합니다.
ImmIsIME ImmIsIME 함수(immdev.h)는 지정된 입력 로캘에 IME가 있는지 여부를 결정합니다.
ImmIsUIMessageA Imm.h(ImmIsUIMessageA) 함수(imm.h)는 IME 창에 대한 메시지를 확인하고 해당 메시지를 창으로 보냅니다.
ImmIsUIMessageA ImmIsUIMessageA(ANSI) 함수(immdev.h)는 IME 창에 대한 메시지를 확인하고 해당 메시지를 창으로 보냅니다.
ImmIsUIMessageW ImmIsUIMessageW(유니코드) 함수(imm.h)는 IME 창에 대한 메시지를 확인하고 해당 메시지를 창으로 보냅니다.

ImmIsUIMessageW ImmIsUIMessageW(유니코드) 함수(immdev.h)는 IME 창용 메시지를 확인하고 해당 메시지를 창으로 보냅니다.
ImmNotifyIME ImmNotifyIME 함수(imm.h)는 입력 컨텍스트의 상태 변경에 대해 IME에 알합니다.
ImmNotifyIME ImmNotifyIME 함수(immdev.h)는 입력 컨텍스트의 상태 변경에 대해 IME에 알합니다.
ImmRegisterWordA ImmRegisterWordA(ANSI) 함수(imm.h)는 지정된 입력 로캘과 연결된 IME 사전을 사용하여 문자열을 등록합니다.
ImmRegisterWordA ImmRegisterWordA(ANSI) 함수(immdev.h)는 지정된 입력 로캘과 연결된 IME 사전을 사용하여 문자열을 등록합니다.
ImmRegisterWordW ImmRegisterWordW(유니코드) 함수(imm.h)는 지정된 입력 로캘과 연결된 IME 사전을 사용하여 문자열을 등록합니다.
ImmRegisterWordW ImmRegisterWordW(유니코드) 함수(immdev.h)는 지정된 입력 로캘과 연결된 IME 사전에 문자열을 등록합니다.
ImmReleaseContext ImmReleaseContext 함수(imm.h)는 입력 컨텍스트를 해제하고 입력 컨텍스트에 연결된 메모리의 잠금을 해제합니다. 애플리케이션은 ImmGetContext 함수에 대한 각 호출에 대해 이 함수를 호출해야 합니다.
ImmReleaseContext ImmReleaseContext 함수(immdev.h)는 입력 컨텍스트를 해제하고 입력 컨텍스트에 연결된 메모리의 잠금을 해제합니다.
ImmRequestMessageA WM_IME_REQUEST 메시지를 생성합니다. (ANSI)

ImmRequestMessageW WM_IME_REQUEST 메시지를 생성합니다. (유니코드)
ImmSetCandidateWindow ImmSetCandidateWindow 함수(imm.h)는 후보 창에 대한 정보를 설정합니다.
ImmSetCandidateWindow ImmSetCandidateWindow 함수(immdev.h)는 후보 창에 대한 정보를 설정합니다.
ImmSetCompositionFontA ImmSetCompositionFontA(ANSI) 함수(imm.h)는 컴퍼지션 창에 문자를 표시하는 데 사용할 논리 글꼴을 설정합니다.
ImmSetCompositionFontA ImmSetCompositionFontA(ANSI) 함수(immdev.h)는 컴퍼지션 창에 문자를 표시하는 데 사용할 논리 글꼴을 설정합니다.
ImmSetCompositionFontW ImmSetCompositionFontW(유니코드) 함수(imm.h)는 컴퍼지션 창에 문자를 표시하는 데 사용할 논리 글꼴을 설정합니다.
ImmSetCompositionFontW ImmSetCompositionFontW(유니코드) 함수(immdev.h)는 컴퍼지션 창에 문자를 표시하는 데 사용할 논리 글꼴을 설정합니다.
ImmSetCompositionStringA ImmSetCompositionStringA(ANSI) 함수(imm.h)는 컴퍼지션 및 읽기 문자열의 문자, 특성 및 절을 설정합니다.
ImmSetCompositionStringA ImmSetCompositionStringA(ANSI) 함수(immdev.h)는 컴퍼지션 및 읽기 문자열의 문자, 특성 및 절을 설정합니다.
ImmSetCompositionStringW ImmSetCompositionStringW(유니코드) 함수(imm.h)는 컴퍼지션 및 읽기 문자열의 문자, 특성 및 절을 설정합니다.

ImmSetCompositionStringW ImmSetCompositionStringW(유니코드) 함수(immdev.h)는 컴퍼지션 및 읽기 문자열의 문자, 특성 및 절을 설정합니다.
ImmSetCompositionWindow ImmSetCompositionWindow 함수(imm.h)는 컴퍼지션 창의 위치를 설정합니다.
ImmSetCompositionWindow ImmSetCompositionWindow 함수(immdev.h)는 컴퍼지션 창의 위치를 설정합니다.
ImmSetConversionStatus ImmSetConversionStatus 함수(imm.h)는 현재 변환 상태를 설정합니다.
ImmSetConversionStatus ImmSetConversionStatus 함수(immdev.h)는 현재 변환 상태를 설정합니다.
ImmSetOpenStatus ImmSetOpenStatus 함수(imm.h)가 열리거나 IME를 닫습니다.
ImmSetOpenStatus ImmSetOpenStatus 함수(immdev.h)가 IME를 열거나 닫습니다.
ImmSetStatusWindowPos ImmSetStatusWindowPos 함수(imm.h)는 상태 창의 위치를 설정합니다.
ImmSetStatusWindowPos ImmSetStatusWindowPos 함수(immdev.h)는 상태 창의 위치를 설정합니다.
ImmSimulateHotKey ImmSimulateHotKey 함수(imm.h)는 지정된 IME 핫 키를 시뮬레이트하여 사용자가 지정된 창에서 핫 키를 누를 때와 동일한 응답을 생성합니다.
ImmSimulateHotKey ImmSimulateHotKey 함수(immdev.h)는 지정된 IME 핫 키를 시뮬레이트하여 사용자가 지정된 창에서 핫 키를 누를 때와 동일한 응답을 발생시킵니다.

ImmUnregisterWordA ImmUnregisterWordA(ANSI) 함수(imm.h)는 지정된 입력 로캘과 연결된 IME 사전에서 레지스터 문자열을 제거합니다.
ImmUnregisterWordA ImmUnregisterWordA(ANSI) 함수(immdev.h)는 지정된 입력 로캘과 연결된 IME 사전에서 레지스터 문자열을 제거합니다.
ImmUnregisterWordW ImmUnregisterWordW(유니코드) 함수(imm.h)는 지정된 입력 로캘과 연결된 IME 사전에서 레지스터 문자열을 제거합니다.
ImmUnregisterWordW ImmUnregisterWordW(유니코드) 함수(immdev.h)는 지정된 입력 로캘과 연결된 IME 사전에서 레지스터 문자열을 제거합니다.
초기화 IImePadApplet을 초기화하기 위해 IImePad 인터페이스에서 호출됩니다.
InitializeWordlist 지정된 단어만 포함하도록 지정한 단어 목록을 초기화합니다.
Invoke SpellCheckerChanged 이벤트를 받습니다.
InvokeDictToolDialog 앱에서 Microsoft IME의 사전 도구를 호출합니다.
InvokeWordRegDialog 앱에서 Microsoft IME Word 등록 대화 상자 창을 호출합니다.
IS_HIGH_SURROGATE 문자가 utF-16 높은 서로게이트 코드 지점인지 여부를 확인합니다(0xd800 0xdbff 포함).
IS_LOW_SURROGATE 문자가 utF-16 하위 서로게이트 코드 지점인지 여부를 확인합니다(0xdc00부터 0xdfff까지 포함).

IS_SURROGATE_PAIR 지정된 코드 단위가 UTF-16 서로게이트 쌍을 형성하는지 여부를 확인합니다.
IsDBCSLeadByte 지정된 문자가 시스템 기본 Windows ANSI 코드 페이지(CP_ACP)의 리드 바이트인지 여부를 확인합니다. 리드 바이트는 코드 페이지에 대한 DBCS(더블 바이트 문자 집합)에서 2 바이트 문자의 첫 번째 바이트입니다.
IsDBCSLeadByteEx 지정된 문자가 잠재적으로 리드 바이트인지 여부를 확인합니다. 리드 바이트는 코드 페이지에 대한 DBCS(더블 바이트 문자 집합)에서 2 바이트 문자의 첫 번째 바이트입니다.
IsDefaultIME 클래스 ID로 지정된 IME가 로컬 컴퓨터의 기본 IME인지 여부를 확인합니다.
IsNLSDefinedString 문자열의 각 문자에 지정된 NLS 기능에 대해 정의된 결과가 있는지 확인합니다.
IsNormalizedString 유니코드 4.0 TR#15에 따라 문자열이 정규화되었는지 확인합니다. 자세한 내용은 유니코드 정규화를 사용하여 문자열을 나타냅니다.
Issupported 지정한 언어가 이 맞춤법 검사기에서 지원되는지 여부를 확인합니다.
Issupported 지정된 언어가 등록된 맞춤법 검사기에서 지원되는지 여부를 확인합니다.
IsTextUnicode 버퍼에 유니코드 텍스트 형식이 포함될 가능성이 있는지 확인합니다.
IsValidCodePage 지정된 코드 페이지가 유효한지 확인합니다.
IsValidLanguageGroup 언어 그룹이 운영 체제에 설치되거나 지원되는지 여부를 확인합니다. 자세한 내용은 NLS 용어를 참조하세요.

IsValidLocale 지정된 로캘이 운영 체제에 설치되거나 지원되는지 여부를 확인합니다. 자세한 내용은 로캘 및 언어를 참조하세요.
IsValidLocaleName 운영 체제에 설치되거나 지원되는 로캘에 대해 지정된 로캘 이름이 유효한지 확인합니다. 참고 Windows Vista 이상에서만 실행되는 애플리케이션은 기본 설정에서 IsValidLocale로 이 함수를 호출하여 추가 로캘의 유효성을 확인해야 합니다.
IsValidNLSVersion 지정된 NLS 함수에 대해 NLS 버전이 유효한지 확인합니다.
LANGGROUPLOCALE_ENUMPROCA EnumLanguageGroupLocales 함수에서 제공하는 열거형 언어 그룹 로캘 정보를 처리하는 애플리케이션 정의 콜백 함수입니다. (ANSI)
LANGGROUPLOCALE_ENUMPROCW EnumLanguageGroupLocales 함수에서 제공하는 열거형 언어 그룹 로캘 정보를 처리하는 애플리케이션 정의 콜백 함수입니다. (유니코드)
LANGIDFROMLCID 로캘 식별자에서 언어 식별자를 검색합니다.
LCIDToLocaleName 로캘 식별자를 로캘 이름으로 변환합니다.
LCMapStringA 식별자로 지정된 로캘의 경우 지정된 변환을 사용하여 한 입력 문자 문자열을 다른 입력 문자열에 매핑하거나 입력 문자열에 대한 정렬 키를 생성합니다. (ANSI)
LCMapStringEx 이름으로 지정된 로캘의 경우 지정된 변환을 사용하여 입력 문자 문자열을 다른 문자열에 매핑하거나 입력 문자열에 대한 정렬 키를 생성합니다. 참고 Windows Vista 이상에서만 실행되도록 설계된 경우 애플리케이션은 LCMapString을 기본 설정으로 이 함수를 호출해야 합니다.
LCMapStringW 식별자로 지정된 로캘의 경우 지정된 변환을 사용하여 한 입력 문자 문자열을 다른 입력 문자열에 매핑하거나 입력 문자열에 대한 정렬 키를 생성합니다. (유니코드)

LoadMUILibraryA 특정 LN(언어 중립적) 파일과 연결된 언어별 리소스에 대한 핸들을 반환합니다. (ANSI)
LoadMUILibraryW 특정 LN(언어 중립적) 파일과 연결된 언어별 리소스에 대한 핸들을 반환합니다. (유니코드)
LoadStringByReference 지원 안 됨 LoadStringByReference를 변경하거나 사용할 수 없습니다. 대신 SHLoadIndirectString을 사용합니다.
LOCALE_ENUMPROCEX EnumSystemLocalesEx 함수에서 제공하는 열거형 로캘 정보를 처리하는 애플리케이션 정의 콜백 함수입니다.
LocaleNameToLCID 로캘 이름을 로캘 식별자로 변환합니다.
MAKELANGID 기본 언어 식별자 및 하위 언어 식별자에서 언어 식별자를 만듭니다.
MAKELCID 언어 식별자 및 정렬 순서 식별자에서 로캘 식별자를 만듭니다.
MAKESORTLCID 언어 식별자, 정렬 순서 식별자 및 정렬 버전에서 LCID(로캘 식별자)를 생성합니다.
MappingDoAction 텍스트 인식이 발생한 후 ELS 서비스가 작업을 수행하도록 합니다. 예를 들어 전화 걸기 서비스는 먼저 전화 번호를 인식한 다음 전화 걸기의 "동작"을 수행할 수 있습니다.
MappingFreePropertyBag ELS 텍스트 인식 작업 중에 할당된 메모리 및 리소스를 해제합니다.
MappingFreeServices 애플리케이션이 하나 이상의 ELS 서비스와 상호 작용하도록 할당된 메모리 및 리소스를 해제합니다. 메모리 및 리소스는 MappingGetServices에 대한 애플리케이션 호출에서 할당됩니다.

MappingGetServices 애플리케이션에서 지정한 기준에 따라 연결된 정보와 함께 사용 가능한 ELS 플랫폼 지원 서비스 목록을 검색합니다.
MappingRecognizeText ELS 서비스를 호출하여 텍스트를 인식합니다. 예를 들어 Microsoft 언어 감지 서비스는 입력 텍스트가 작성된 언어를 인식하려고 시도합니다.
MultiByteToWideChar 문자열을 UTF-16(와이드 문자) 문자열에 매핑합니다.
다음 다음 맞춤법 오류를 가져옵니다.
NextWords 사전에서 다음 단어 항목을 가져옵니다.
NlsDllCodePageTranslation 플래그 설정에 따라 코드 페이지 정보를 얻거나 변환을 수행하는 데 사용됩니다.
NormalizeString 유니코드 4.0 TR#15에 따라 텍스트 문자열의 문자를 정규화합니다. 자세한 내용은 유니코드 정규화를 사용하여 문자열을 나타내는 방법을 참조하세요.
알림 알림 코드와 함께 정보를 전달하기 위해 IImePad에서 호출됩니다.
NotifyUILanguageChange 지원 안 됨 NotifyUILanguageChange를 변경하거나 사용할 수 없습니다.
열기 사전 파일을 엽니다.
열기 IFELanguage 개체를 초기화합니다.

PFN_MAPPINGCALLBACKPROC MappingRecognizeText 함수에서 생성된 데이터를 비동기적으로 처리하는 애플리케이션 정의 콜백 함수입니다.
PRIMARYLANGID 언어 식별자에서 기본 언어 식별자를 추출합니다.
RegisterUserDictionary 등록 취소될 때까지 현재 사용자의 사용자 사전으로 사용할 파일을 등록합니다.
RegisterWord IFEDictionary에서 새 단어를 등록하거나 기존 단어를 삭제합니다.
REGISTERWORDENUMPROCA ANSI(REGISTERWORDENUMPROCA)(imm.h)는 ImmEnumRegisterWord 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
REGISTERWORDENUMPROCA ANSI(REGISTERWORDENUMPROCA)는 ImmEnumRegisterWord 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
REGISTERWORDENUMPROCW REGISTERWORDENUMPROCW(유니코드)(imm.h)는 ImmEnumRegisterWord 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
REGISTERWORDENUMPROCW REGISTERWORDENUMPROCW(유니코드)는 ImmEnumRegisterWord 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (registerwordenumprocw)
제거 이전에 ISpellChecker.Add에서 추가되었거나 ISpellChecker.Ignore에서 설정한 단어를 제거하여 무시합니다.
remove_SpellCheckerChanged SpellCheckerChanged 이벤트에 대해 추가된 이벤트 처리기(ISpellCheckerChangedEventHandler)를 제거합니다.

요청 앱에 텍스트를 삽입하기 위해 IImePadApplet에서 호출됩니다.
ResolveLocaleName 제공된 이름에 대해 가능한 로캘 이름 일치를 찾습니다.
ScriptApplyDigitSubstitution 지정된 숫자 대체 설정을 지정된 스크립트 컨트롤 및 스크립트 상태 구조에 적용합니다.
ScriptApplyLogicalWidth 실행에 대한 선진 너비 배열을 사용하고 조정된 선진 문자 모양 너비의 배열을 생성합니다.
ScriptBreak 줄 바꿈을 결정하기 위한 정보를 검색합니다.
ScriptCacheGetHeight 현재 캐시된 글꼴의 높이를 검색합니다.
ScriptCPtoX 실행의 왼쪽 끝 또는 선행 가장자리에서 논리 문자 클러스터의 선행 또는 후행 가장자리로 x 오프셋을 생성합니다.
ScriptFreeCache 스크립트 캐시를 해제합니다.
ScriptGetCMap TrueType cmap 테이블 또는 이전 스타일 글꼴에 대해 구현된 표준 cmap 테이블에 따라 문자열에서 유니코드 문자의 문자 모양 인덱스를 검색합니다.
ScriptGetFontAlternateGlyphs 지정된 OpenType 기능을 통해 액세스할 수 있는 지정된 문자의 대체 문자 모양 목록을 검색합니다.
ScriptGetFontFeatureTags OpenType 처리를 위해 정의된 쓰기 시스템에 대한 입력 기능 목록을 검색합니다. 목록을 구성하는 입력 기능 태그는 제공된 디바이스 컨텍스트 또는 캐시의 글꼴에서 검색됩니다.

ScriptGetFontLanguageTags 지정한 항목에 사용할 수 있고 OpenType 처리를 위해 지정된 스크립트 태그에서 지원되는 언어 태그 목록을 검색합니다. 목록을 구성하는 태그는 지정된 디바이스 컨텍스트 또는 캐시의 글꼴에서 검색됩니다.
ScriptGetFontProperties 글꼴에서 사용하는 특수 문자 모양의 글꼴 캐시에서 정보를 검색합니다.
ScriptGetFontScriptTags OpenType 처리를 위해 글꼴에서 사용할 수 있는 스크립트 목록을 검색합니다. 목록을 구성하는 스크립트는 제공된 디바이스 컨텍스트에 있는 글꼴 또는 현재 실행의 글꼴을 처리하는 스크립트 셰이핑 엔진에서 검색됩니다.
ScriptGetGlyphABCWidth 지정된 문자 모양의 ABC 너비를 검색합니다.
ScriptGetLogicalWidths 특정 글꼴의 문자 모양 앞으로 너비를 논리적 너비로 변환합니다.
ScriptGetProperties 현재 스크립트에 대한 정보를 검색합니다.
ScriptIsComplex 유니코드 문자열에 복잡한 스크립트 처리가 필요한지 여부를 결정합니다.
ScriptItemize 유니코드 문자열을 개별적으로 셰이프 가능한 항목으로 나눕니다.
ScriptItemizeOpenType 유니코드 문자열을 개별적으로 셰이프 가능한 항목으로 나누고 OpenType 처리를 위해 각 셰이프 가능한 항목에 대한 기능 태그 배열을 제공합니다.
ScriptJustify ScriptTextOut 함수에 전달할 때 텍스트 근거를 허용하는 고급 너비 테이블을 만듭니다.
ScriptLayout 실행 포함 수준 배열을 시각적-논리적 위치 및/또는 논리적-시각적 위치의 맵으로 변환합니다.

ScriptPlace ScriptShape의 출력에서 문자 모양 선진 너비 및 2차원 오프셋 정보를 생성합니다.
ScriptPlaceOpenType ScriptShapeOpenType의 출력에서 OpenType 정보를 사용하여 유니코드 실행에 대한 문자 모양 및 시각적 특성을 생성합니다.
ScriptPositionSingleGlyph OpenType 처리를 위해 글꼴에 제공된 지정된 기능을 사용하여 단일 조정으로 단일 문자 모양을 배치합니다. 대부분의 경우 애플리케이션은 이 함수를 사용하여 선의 시작 또는 끝에 문자 모양을 광학적으로 정렬합니다.
ScriptRecordDigitSubstitution NLS(국가 언어 지원) 네이티브 숫자 및 숫자 대체 설정을 읽고 SCRIPT_DIGITSUBSTITUTE 구조에 기록합니다. 자세한 내용은 숫자 셰이프를 참조하세요.
ScriptShape 유니코드 실행에 대한 문자 모양 및 시각적 특성을 생성합니다.
ScriptShapeOpenType OpenType 정보를 사용하여 유니코드 실행에 대한 문자 모양 및 시각적 특성을 생성합니다. 각 실행은 이 함수에 대한 하나의 호출로 구성됩니다.
ScriptString_pcOutChars 클리핑 후 문자열의 길이에 대한 포인터를 반환합니다.
ScriptString_pLogAttr 분석된 문자열의 논리 특성 버퍼에 대한 포인터를 반환합니다.
ScriptString_pSize 분석된 문자열의 SIZE 구조에 대한 포인터를 반환합니다.
ScriptStringAnalyse 일반 텍스트 문자열을 분석합니다.
ScriptStringCPtoX 문자 위치의 선행 또는 후행 가장자리에 대한 x 좌표를 검색합니다.

ScriptStringFree SCRIPT_STRING_ANALYSIS 구조를 해제합니다.
ScriptStringGetLogicalWidths 시각적 개체 너비를 논리적 너비로 변환합니다.
ScriptStringGetOrder 원래 문자 위치를 문자 모양 위치에 매핑하는 배열을 만듭니다.
ScriptStringOut ScriptStringAnalyse에 대한 이전 호출로 생성된 문자열을 표시하고 필요에 따라 강조 표시를 추가합니다.
ScriptStringValidate SCRIPT_STRING_ANALYSIS 구조에서 잘못된 시퀀스를 확인합니다.
ScriptStringXtoCP x 좌표를 문자 위치로 변환합니다.
ScriptSubstituteSingleGlyph OpenType 처리에 대해 동일한 문자 모양의 대체 형식으로 단일 문자 모양을 대체합니다.
ScriptTextOut 지정한 스크립트 셰이프의 텍스트를 표시하고 정보를 배치합니다.
ScriptXtoCP 실행의 x 오프셋에서 논리 문자 클러스터의 선행 또는 후행 가장자리를 생성합니다.
SetCalendarInfoA 일정에 대한 로캘 정보의 항목을 설정합니다. 자세한 내용은 날짜 및 일정을 참조하세요. (ANSI)
SetCalendarInfoW 일정에 대한 로캘 정보의 항목을 설정합니다. 자세한 내용은 날짜 및 일정을 참조하세요. (유니코드)
SetDefaultIME Microsoft IME가 키보드 레이아웃의 기본 IME가 되도록 허용합니다.

SetHeader 사전 파일에서 사전 헤더를 설정합니다.
SetLocaleInfoA 현재 로캘의 사용자 재정의 부분에 있는 정보 항목을 설정합니다. 이 함수는 시스템 기본값을 설정하지 않습니다. (ANSI)
SetLocaleInfoW 현재 로캘의 사용자 재정의 부분에 있는 정보 항목을 설정합니다. 이 함수는 시스템 기본값을 설정하지 않습니다. (유니코드)
SetOptionValue 지정된 옵션과 연결된 값을 설정합니다.
SetProcessPreferredUILanguages 애플리케이션 프로세스에 대한 프로세스 기본 설정 UI 언어를 설정합니다. 자세한 내용은 사용자 인터페이스 언어 관리를 참조하세요.
SetThreadLocale 호출 스레드의 현재 로캘을 설정합니다.
SetThreadPreferredUILanguages 현재 스레드에 대한 스레드 기본 설정 UI 언어를 설정합니다. 자세한 내용은 사용자 인터페이스 언어 관리를 참조하세요.
SetThreadUILanguage 현재 스레드에 대한 사용자 인터페이스 언어를 설정합니다.
SetUserGeoID 사용자의 지리적 위치 식별자를 설정합니다. 이 식별자에는 지리적 위치 테이블에 설명된 값 중 하나가 있어야 합니다.
SetUserGeoName 현재 사용자의 지리적 위치를 지정된 ISO(International Organization for Standardization) 3166-1 코드 또는 숫자 UN(UN) 시리즈 M, 번호 49(M.49) 코드로 설정합니다.
SORTIDFROMLCID 로캘 식별자에서 정렬 순서 식별자를 검색합니다.

SORTVERSIONFROMLCID 로캘 식별자에서 정렬 버전을 검색합니다.
SUBLANGID 언어 식별자에서 하위 언어 식별자를 추출합니다.
제안 제공된 텍스트에 대한 맞춤법 제안을 검색합니다. (ISpellCheckProvider.Suggest)
제안 제공된 텍스트에 대한 맞춤법 제안을 검색합니다. (ISpellChecker.Suggest)
Terminate IMEPad 인스턴스가 종료되면 IImePadApplet을 종료하기 위해 IImePad에서 호출됩니다.
텍스트 컴파일하는 동안 유니코드가 전처리기 지시문에 의해 정의되는 경우 문자열을 유니코드로 식별합니다. 그렇지 않으면 매크로는 문자열을 ANSI 문자열로 식별합니다.
TranslateCharsetInfo 문자 집합 정보를 변환하고 대상 구조의 모든 멤버를 적절한 값으로 설정합니다.
UILANGUAGE_ENUMPROCA EnumUILanguages 함수에서 제공하는 열거형 사용자 인터페이스 언어 정보를 처리하는 애플리케이션 정의 콜백 함수입니다. (ANSI)
UILANGUAGE_ENUMPROCW EnumUILanguages 함수에서 제공하는 열거형 사용자 인터페이스 언어 정보를 처리하는 애플리케이션 정의 콜백 함수입니다. (유니코드)
UnregisterUserDictionary 이전에 등록된 사용자 사전의 등록을 취소합니다.
VerifyScripts 열거된 두 스크립트 목록을 비교합니다.

WideCharToMultiByte UTF-16(와이드 문자) 문자열을 새 문자열에 매핑합니다.

인터페이스

IComprehensiveSpellCheckProvider 공급자가 필요에 따라 보다 포괄적인 맞춤법 검사 기능을 지원할 수 있습니다.
IEnumSpellingError 맞춤법 오류의 열거형입니다.
IFECommon IFECommon 인터페이스는 다양한 언어에 공통적인 IME 관련 서비스를 제공합니다.
IFEDictionary IFEDictionary 인터페이스를 사용하면 클라이언트가 Microsoft IME 사용자 사전에 액세스할 수 있습니다.
IFELanguage IFELanguage 인터페이스는 Microsoft IME를 사용하여 언어 처리 서비스를 제공합니다.
IImePad IImePad 인터페이스는 IImePadApplet 인터페이스를 구현하는 IMEPadApplet의 앱에 텍스트를 삽입합니다.
IImePadApplet IImePadApplet 인터페이스는 IImePad 인터페이스를 통해 문자열을 앱에 입력합니다.
IImePlugInDictDictionaryList IME 플러그 인 사전 목록에 대한 액세스를 제공합니다.
IImeSpecifyApplets IImeSpecifyApplets 인터페이스는 IImePadApplet 인터페이스를 에뮬레이트하기 위해 IImePad 인터페이스 개체에서 호출된 메서드를 지정합니다.

IOptionDescription 맞춤법 검사기 옵션의 설명을 나타냅니다.
ISpellChecker 특정 언어에 대한 특정 맞춤법 검사기를 나타냅니다.
ISpellChecker2 추가된 단어 사전 또는 무시 목록에서 단어를 제거하는 기능이 추가된 특정 언어의 특정 맞춤법 검사기를 나타냅니다.
ISpellCheckerChangedEventHandler 호출자가 맞춤법 검사기의 상태가 변경된 알림에 대한 처리기를 만들 수 있습니다.
ISpellCheckerFactory 맞춤법 검사기(ISpellChecker)를 인스턴스화하고 지원되는 언어를 결정하는 기능을 제공하는 팩터리입니다.
ISpellCheckProvider 맞춤법 검사 인프라에서 사용할 특정 언어에 대한 특정 맞춤법 검사기 공급자를 나타냅니다.
ISpellCheckProviderFactory 맞춤법 검사기(ISpellCheckProvider)를 인스턴스화하고 지원되는 언어를 결정하는 기능을 제공하는 팩터리입니다.
ISpellingError 맞춤법 오류에 대한 정보를 제공합니다.
IUserDictionariesRegistrar 사용자 사전의 등록을 관리합니다.

구조체

APPLETIDLIST IImePadApplet IID 목록을 지정합니다.

CANDIDATEFORM CANDIDATEFORM(imm.h) 구조체에는 후보 창에 대한 위치 정보가 포함됩니다.
CANDIDATEFORM CANDIDATEFORM 구조체(immdev.h)에는 후보 창에 대한 위치 정보가 포함됩니다.
후보 목록 CANDIDATELIST(imm.h) 구조체에는 후보 목록에 대한 정보가 포함됩니다.
후보 목록 CANDIDATELIST 구조(immdev.h)에는 후보 목록에 대한 정보가 포함됩니다.
CHARSETINFO 문자 집합에 대한 정보를 포함합니다.
COMPOSITIONFORM COMPOSITIONFORM(imm.h) 구조체에는 컴퍼지션 창에 대한 스타일 및 위치 정보가 포함됩니다.
COMPOSITIONFORM COMPOSITIONFORM 구조체(immdev.h)에는 컴퍼지션 창에 대한 스타일 및 위치 정보가 포함됩니다.
CPINFO 코드 페이지에 대한 정보를 포함합니다. 이 구조체는 GetCPInfo 함수에서 사용됩니다.
CPINFOEXA 코드 페이지에 대한 정보를 포함합니다. 이 구조체는 GetCPInfoEx 함수에서 사용됩니다. (ANSI)
CPINFOEXW 코드 페이지에 대한 정보를 포함합니다. 이 구조체는 GetCPInfoEx 함수에서 사용됩니다. (유니코드)
CURRENCYFMTA 통화 문자열의 형식을 정의하는 정보를 포함합니다. GetCurrencyFormat 함수는 이 정보를 사용하여 지정된 로캘에 대한 통화 문자열을 사용자 지정합니다. (ANSI)

CURRENCYFMTW 통화 문자열의 형식을 정의하는 정보를 포함합니다. GetCurrencyFormat 함수는 이 정보를 사용하여 지정된 로캘에 대한 통화 문자열을 사용자 지정합니다. (유니코드)
FILEMUIINFO MUI 사용과 관련된 파일에 대한 정보를 포함합니다.
FONTSIGNATURE 지정된 글꼴에서 문자 모양을 제공하는 코드 페이지 및 유니코드 하위 범위를 식별하는 정보를 포함합니다.
GOFFSET 결합 문자 모양의 x 및 y 오프셋을 포함합니다.
IMEAPPLETCFG IImePad에서 애플릿 구성을 지정하고 설정하는 데 사용됩니다.
IMEAPPLETUI IImePadApplet::CreateUI에서 애플릿 창 스타일을 지정하는 데 사용됩니다.
IMECHARPOSITION IMECHARPOSITION(imm.h) 구조체에는 컴퍼지션 창의 문자 위치에 대한 정보가 포함됩니다.
IMECHARPOSITION IMECHARPOSITION 구조체(immdev.h)는 컴퍼지션 창의 문자 위치에 대한 정보를 포함합니다.
IMECOMPOSITIONSTRINGINFO 앱에서 IME의 컴퍼지션 문자열에 대한 정보를 포함합니다.
IMEDLG 앱에서 Microsoft IME의 사전 도구 또는 Word 등록 대화 상자 창을 호출할 때 사용됩니다.
IMEMENUITEMINFOA IMEMENUITEMINFOA(ANSI) 구조체(imm.h)에는 IME 메뉴 항목에 대한 정보가 포함되어 있습니다.

IMEMENUITEMINFOA IMEMENUITEMINFOA(ANSI) 구조체(immdev.h)에는 IME 메뉴 항목에 대한 정보가 포함되어 있습니다.
IMEMENUITEMINFOW IMEMENUITEMINFOW(유니코드) 구조체(imm.h)에는 IME 메뉴 항목에 대한 정보가 포함되어 있습니다.
IMEMENUITEMINFOW IMEMENUITEMINFOW(유니코드) 구조체(immdev.h)에는 IME 메뉴 항목에 대한 정보가 포함되어 있습니다.
IMESHF 열린 사용자 사전 파일의 헤더입니다. 버전, 제목, 설명 및 저작권과 같은 사용자 사전의 속성을 가져오는 데 사용됩니다.
IMEWRD Microsoft IME 사전의 Word 데이터에 있는 단어에 대한 데이터를 포함합니다.
LOCALESIGNATURE 기본 및 지원되는 문자 집합 및 코드 페이지를 정의하는 두 개의 CPB(코드 페이지 비트 필드)를 포함하여 확장된 글꼴 서명 정보를 포함합니다. 이 구조는 일반적으로 글꼴 검사와 로캘 간의 관계를 나타내는 데 사용됩니다.
MAPPING_DATA_RANGE 인식된 텍스트 하위 범위의 텍스트 인식 결과를 포함합니다. 이 형식의 구조 배열은 MAPPING_PROPERTY_BAG 구조체의 ELS(확장 언어 서비스) 서비스에 의해 검색됩니다.
MAPPING_ENUM_OPTIONS MAppingGetServices 함수에서 ELS 서비스를 열거하는 데 사용하는 옵션을 포함합니다.
MAPPING_OPTIONS 텍스트 인식 옵션을 포함합니다. 이 구조에 저장된 값은 MappingRecognizeText의 동작 및 결과에 영향을 미칩니다.
MAPPING_PROPERTY_BAG MappingRecognizeText에서 검색한 텍스트 인식 데이터 속성을 포함합니다.

MAPPING_SERVICE_INFO ELS 서비스에 대한 정보를 포함합니다.
NLSVERSIONINFOEX NLS 기능에 대한 버전 정보를 포함합니다.
NUMBERFMTA 숫자 문자열의 형식을 정의하는 정보를 포함합니다. GetNumberFormat 함수는 이 정보를 사용하여 지정된 로캘에 대한 숫자 문자열을 사용자 지정합니다. (ANSI)
NUMBERFMTW 숫자 문자열의 형식을 정의하는 정보를 포함합니다. GetNumberFormat 함수는 이 정보를 사용하여 지정된 로캘에 대한 숫자 문자열을 사용자 지정합니다. (유니코드)
OPENTYPE_FEATURE_RECORD 실행에 적용할 단일 OpenType 기능에 대한 정보를 포함합니다.
POSTBL 공용 POS(음성의 일부) 테이블의 항목입니다.
RECONVERTSTRING RECONVERTSTRING(imm.h) 구조체는 IME 재연결에 대한 문자열을 정의합니다. 다시 변환할 문자열을 포함하는 메모리 블록의 첫 번째 항목입니다.
RECONVERTSTRING RECONVERTSTRING 구조체(immdev.h)는 IME 재구성에 대한 문자열을 정의합니다.
REGISTERWORDA ANSI(REGISTERWORDA) 구조체(imm.h)에는 등록할 읽기 정보 또는 단어가 포함됩니다.
REGISTERWORDA REGISTERWORDA(ANSI) 구조체(immdev.h)에는 등록할 읽기 정보 또는 단어가 포함됩니다.
REGISTERWORDW REGISTERWORDW(유니코드) 구조체(imm.h)에는 등록할 읽기 정보 또는 단어가 포함됩니다.

REGISTERWORDW REGISTERWORDW(유니코드) 구조체(immdev.h)에는 등록할 읽기 정보 또는 단어가 포함됩니다.
SCRIPT_ANALYSIS 유니코드 문자열의 일부, 즉 "item"을 포함합니다.
SCRIPT_CHARPROP 실행의 단일 문자(입력 문자열)에 대한 정보를 포함합니다. 이 정보는 문자 문자 모양이 실행의 주변 문자에 의해 영향을 받는지 나타냅니다.
SCRIPT_CONTROL 여러 Uniscribe 함수(예: ScriptItemize)에 대한 스크립트 컨트롤 플래그를 포함합니다.
SCRIPT_DIGITSUBSTITUTE 네이티브 숫자 및 숫자 대체 설정을 포함합니다.
SCRIPT_FONTPROPERTIES 현재 글꼴의 속성에 대한 정보를 포함합니다.
SCRIPT_GLYPHPROP 출력 문자 모양 배열의 일부인 문자 모양에 대한 정보를 포함합니다.
SCRIPT_ITEM 항목의 첫 번째 문자에 대한 문자열 오프셋이 있는 SCRIPT_ANALYSIS 구조체를 포함하는 스크립트 항목을 포함합니다.
SCRIPT_LOGATTR 텍스트를 편집하고 서식을 지정할 때 유용한 논리 문자의 특성을 포함합니다.
SCRIPT_PROPERTIES 각 스크립트에 대한 특수 처리에 대한 정보를 포함합니다.
SCRIPT_STATE 스크립트 상태 정보를 포함합니다.
SCRIPT_TABDEF ScriptStringAnalyse에 대한 탭 위치의 정의를 포함합니다.

SCRIPT_VISATTR ScriptShape에서 생성된 클러스터 및 근거 지점을 식별하는 시각적 개체(문자 모양) 특성을 포함합니다.
STYLEBUFA ANSI(STYLEBUFA) 구조체(imm.h)에는 스타일의 식별자와 이름이 포함됩니다.
STYLEBUFA STYLEBUFA(ANSI) 구조체(immdev.h)에는 스타일의 식별자와 이름이 포함됩니다.
STYLEBUFW STYLEBUFW(유니코드) 구조체(imm.h)에는 스타일의 식별자와 이름이 포함됩니다.
STYLEBUFW STYLEBUFW(유니코드) 구조체(immdev.h)에는 스타일의 식별자와 이름이 포함됩니다.
TEXTRANGE_PROPERTIES 실행에 적용할 OpenType 기능 그룹을 포함합니다.

IP 도우미

아티클2023. 03. 09.

IP 도우미 기술 개요입니다.

IP 도우미를 개발하려면 다음 헤더가 필요합니다.

- icmpapi.h
- ifdef.h
- inaddr.h
- ip2string.h
- ipexport.h
- iphlpapi.h
- iprtrmib.h
- iptypes.h
- netioapi.h
- nldef.h
- tcpestats.h
- ws2def.h
- ws2ipdef.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- IP 도우미

열거형

DNS_SERVER_PROPERTY_TYPE DNS_SERVER_PROPERTY::P roperty 멤버에 있는 속성의 유효성을 지정하는 상수를 정의합니다.
IF_OPER_STATUS IF_OPER_STATUS 열거형은 인터페이스의 작동 상태 지정합니다.
NET_ADDRESS_FORMAT NET_ADDRESS_FORMAT 열거형은 ParseNetworkString 함수에서 반환하는 네트워크 주소의 형식을 지정합니다.

NL_DAD_STATE IP_DAD_STATE 열거형은 IPv4 또는 IPv6 주소에 대한 DAD(중복 주소 검색) 상태에 대한 정보를 지정합니다.
NL_NETWORK_CONNECTIVITY_COST_HINT 네트워크 연결에 대한 사용 요금에 대한 힌트를 지정하는 상수를 정의합니다.
NL_NETWORK_CONNECTIVITY_LEVEL_HINT 네트워크 연결 수준에 대한 힌트를 지정하는 상수를 정의합니다.
NL_PREFIX_ORIGIN IP_PREFIX_ORIGIN 열거형은 IPv4 또는 IPv6 주소 접두사 원본을 지정하고 IP_ADAPTER_UNICAST_ADDRESS 구조에 사용됩니다.
NL_SUFFIX_ORIGIN IP_SUFFIX_ORIGIN 열거형은 IPv4 또는 IPv6 주소 접미사의 원본을 지정하고 IP_ADAPTER_UNICAST_ADDRESS 구조에 사용됩니다.
SCOPE_LEVEL SCOPE_LEVEL 열거형은 IP_ADAPTER_ADDRESSES 구조와 함께 사용하여 IPv6 주소에 대한 scope 수준을 식별합니다.
TCP_BOOLEAN_OPTIONAL TCP 연결에 대한 읽기/쓰기 정보의 멤버를 업데이트할 때 호출자가 지정할 수 있는 상태를 정의합니다.
TCP_ESTATS_TYPE 요청되거나 설정되는 TCP 연결에 대한 확장 통계 유형을 정의합니다.
TCP_SOFT_ERROR TCP 연결에 기록된 치명적이거나 일시적이지 않은 오류의 이유를 정의합니다.
TCP_TABLE_CLASS GetExtendedTcpTable 호출에서 반환되는 테이블 유형을 나타내는 데 사용되는 값 집합을 정의합니다.
TCPIP_OWNER_MODULE_INFO_CLASS GetOwnerModuleFromXXXEntry 제품군의 호출에 전달되는 모듈 정보 구조의 형식을 정의합니다.

UDP_TABLE_CLASS GetExtendedUdpTable 호출에서 반환되는 테이블 유형을 나타내는 데 사용되는 값 집합을 정의합니다.

Functions

AddIPAddress AddIPAddress 함수는 지정된 IPv4 주소를 지정된 어댑터에 추가합니다.
AllocateAndGetTcpExTableFromStack TCP 연결 테이블을 검색하고 로컬 힙에서 메모리를 할당하여 테이블을 저장합니다.
AllocateAndGetUdpExTableFromStack UDP 연결 테이블을 검색하고 로컬 힙에서 메모리를 할당하여 테이블을 저장합니다.
CancelIfTimestampConfigChange 이 함수는 시스템 사용을 위해 예약되어 있으며 코드에서 호출하면 안 됩니다. (CancelIfTimestampConfigChange)
CancelIPChangeNotify NotifyAddrChange 또는 NotifyRouteChange 함수를 성공적으로 호출하여 이전에 요청된 IPv4 주소 및 경로 변경 내용에 대한 알림을 취소합니다.
CancelMibChangeNotify2 IP 인터페이스 변경, IP 주소 변경, IP 경로 변경, Teredo 포트 변경 및 유니캐스트 IP 주소 테이블이 안정적이며 검색할 수 있는 경우 변경 알림에 대한 등록을 취소합니다.
CaptureInterfaceHardwareCrossTimestamp 네트워크 어댑터에 대한 크로스 타임스탬프 정보를 검색합니다.
ConvertInterfaceAliasToLuid 네트워크 인터페이스의 인터페이스 별칭 이름을 인터페이스의 LUID(로컬 고유 식별자)로 변환합니다.

ConvertInterfaceGuidToLuid 네트워크 인터페이스의 GUID(Globally Unique Identifier)를 인터페이스의 LUID(로컬 고유 식별자)로 변환합니다.
ConvertInterfaceIndexToLuid 네트워크 인터페이스의 로컬 인덱스를 인터페이스의 LUID(로컬 고유 식별자)로 변환합니다.
ConvertInterfaceLuidToAlias 네트워크 인터페이스에 대한 LUID(로컬 고유 식별자)를 인터페이스 별칭으로 변환합니다.
ConvertInterfaceLuidToGuid 네트워크 인터페이스에 대한 LUID(로컬 고유 식별자)를 인터페이스의 GUID(Globally Unique Identifier)로 변환합니다.
ConvertInterfaceLuidToIndex 네트워크 인터페이스에 대한 LUID(로컬 고유 식별자)를 인터페이스의 로컬 인덱스로 변환합니다.
ConvertInterfaceLuidToNameA 네트워크 인터페이스의 LUID(로컬 고유 식별자)를 ANSI 인터페이스 이름으로 변환합니다.
ConvertInterfaceLuidToNameW 네트워크 인터페이스의 LUID(로컬 고유 식별자)를 유니코드 인터페이스 이름으로 변환합니다.
ConvertInterfaceNameToLuidA ANSI 네트워크 인터페이스 이름을 인터페이스의 LUID(로컬 고유 식별자)로 변환합니다.
ConvertInterfaceNameToLuidW 유니코드 네트워크 인터페이스 이름을 인터페이스의 LUID(로컬 고유 식별자)로 변환합니다.
ConvertIpv4MaskToLength IPv4 서브넷 마스크를 IPv4 접두사 길이로 변환합니다.
ConvertLengthToIpv4Mask IPv4 접두사 길이를 IPv4 서브넷 마스크로 변환합니다.
CreateAnycastIpAddressEntry 로컬 컴퓨터에 새 애니캐스트 IP 주소 항목을 추가합니다.

CreateIpForwardEntry CreateIpForwardEntry 함수는 로컬 컴퓨터의 IPv4 라우팅 테이블에 경로를 만듭니다.
CreateIpForwardEntry2 로컬 컴퓨터에 새 IP 경로 항목을 만듭니다.
CreateIpNetEntry CreateIpNetEntry 함수는 로컬 컴퓨터의 ARP 테이블에 ARP(주소 확인 프로토콜) 항목을 만듭니다.
CreateIpNetEntry2 로컬 컴퓨터에 새 인접 IP 주소 항목을 만듭니다.
CreatePersistentTcpPortReservation 로컬 컴퓨터에서 TCP 포트의 연속 블록에 대한 영구 TCP 포트 예약을 만듭니다.
CreatePersistentUdpPortReservation 로컬 컴퓨터에서 연속된 UDP 포트 블록에 대한 영구 UDP 포트 예약을 만듭니다.
CreateProxyArpEntry CreateProxyArpEnry 함수는 지정된 IPv4 주소에 대한 로컬 컴퓨터에 PARP(프록시 주소 확인 프로토콜) 항목을 만듭니다.
CreateSortedAddressPairs 제공된 잠재적 IP 대상 주소 목록을 사용하고, 대상 주소를 호스트 컴퓨터의 로컬 IP 주소와 페어링하고, 두 피어 간의 통신에 가장 적합한 주소 쌍에 따라 쌍을 정렬합니다.
CreateUnicastIpAddressEntry 로컬 컴퓨터에 새 유니캐스트 IP 주소 항목을 추가합니다.
DeleteAnycastIpAddressEntry 로컬 컴퓨터에서 기존 애니캐스트 IP 주소 항목을 삭제합니다.
DeleteIPAddress DeleteIPAddress 함수는 AddIPAddress를 사용하여 이전에 추가한 IP 주소를 삭제합니다.
DeleteIpForwardEntry 로컬 컴퓨터의 IPv4 라우팅 테이블에서 기존 경로를 삭제합니다.

DeleteIpForwardEntry2 로컬 컴퓨터에서 IP 경로 항목을 삭제합니다.
DeleteIpNetEntry DeleteIpNetEntry 함수는 로컬 컴퓨터의 ARP 테이블에서 ARP 항목을 삭제합니다.
DeleteIpNetEntry2 로컬 컴퓨터에서 인접 IP 주소 항목을 삭제합니다.
DeletePersistentTcpPortReservation 로컬 컴퓨터에서 연속된 TCP 포트 블록에 대한 영구 TCP 포트 예약을 삭제합니다. (DeletePersistentTcpPortReservation)
DeletePersistentUdpPortReservation 로컬 컴퓨터에서 연속된 TCP 포트 블록에 대한 영구 TCP 포트 예약을 삭제합니다. (DeletePersistentUdpPortReservation)
DeleteProxyArpEntry DeleteProxyArpEntry 함수는 dwAddress 및 dwIfIndex 매개 변수로 지정된 로컬 컴퓨터에서 PARP 항목을 삭제합니다.
DeleteUnicastIpAddressEntry 로컬 컴퓨터에서 기존 유니캐스트 IP 주소 항목을 삭제합니다.
DisableMediaSense DisableMediaSense 함수는 로컬 컴퓨터에서 TCP/IP 스택의 미디어 감지 기능을 사용하지 않도록 설정합니다.
EnableRouter EnableRouter 함수는 로컬 컴퓨터에서 IPv4 전달을 켭니다. 또한 EnableRouter는 IPv4 전달을 사용하도록 설정하는 요청 수를 추적하는 참조 횟수를 증가합니다.
FlushIpNetTable FlushIpNetTable 함수는 로컬 컴퓨터의 ARP 테이블에서 지정된 인터페이스에 대한 모든 ARP 항목을 삭제합니다.
FlushIpNetTable2 FlushIpNetTable2 함수는 로컬 컴퓨터의 IP 인접 테이블을 플러시합니다.

FlushIpPathTable FlushIpPathTable 함수는 로컬 컴퓨터의 IP 경로 테이블을 플러시합니다.
FreeInterfaceDnsSettings GetInterfaceDnsSettings에서 반환된 설정 개체를 해제합니다.
FreeMibTable 네트워크 인터페이스, 주소 및 경로의 테이블을 반환하는 함수에서 할당한 버퍼를 해제합니다(예: GetIfTable2 및 GetAnycastIpAddressTable).
GetAdapterIndex GetAdapterIndex 함수는 이름이 지정된 어댑터의 인덱스를 가져옵니다.
GetAdapterOrderMap GetAdapterOrderMap 함수는 로컬 컴퓨터의 인터페이스에 대한 우선 순위를 나타내는 어댑터 순서 맵을 가져옵니다.
GetAdaptersAddresses 로컬 컴퓨터의 어댑터와 연결된 주소를 검색합니다.
GetAdaptersInfo GetAdaptersInfo 함수는 로컬 컴퓨터에 대한 어댑터 정보를 검색합니다.
GetAnycastIpAddressEntry 로컬 컴퓨터에서 기존 애니캐스트 IP 주소 항목에 대한 정보를 검색합니다.
GetAnycastIpAddressTable 로컬 컴퓨터에서 애니캐스트 IP 주소 테이블을 검색합니다.
GetBestInterface GetBestInterface 함수는 지정된 IPv4 주소에 가장 적합한 경로가 있는 인터페이스의 인덱스를 검색합니다.
GetBestInterfaceEx GetBestInterfaceEx 함수는 지정된 IPv4 또는 IPv6 주소에 가장 적합한 경로가 있는 인터페이스의 인덱스를 검색합니다.

GetBestRoute GetBestRoute 함수는 지정된 대상 IP 주소에 대한 최상의 경로를 검색합니다.
GetBestRoute2 지정된 대상 IP 주소에 대한 최상의 경로를 위해 로컬 컴퓨터에서 IP 경로 항목을 검색합니다.
GetDefaultCompartmentId GetDefaultCompartmentId 함수는 로컬 컴퓨터의 기본 네트워크 라우팅 구획 식별자를 검색합니다.
GetExtendedTcpTable 애플리케이션에서 사용할 수 있는 TCP 엔드포인트 목록이 포함된 테이블을 검색합니다.
GetExtendedUdpTable 애플리케이션에서 사용할 수 있는 UDP 엔드포인트 목록이 포함된 테이블을 검색합니다.
GetFriendlyIfIndex 인터페이스 인덱스를 사용하고 이전 버전과 호환되는 인터페이스 인덱스, 즉 하위 24비트만 사용하는 인덱스를 반환합니다.
GetIcmpStatistics GetIcmpStatistics 함수는 로컬 컴퓨터에 대한 IPv4 통계에 대한 ICMP(Internet Control Message Protocol)를 검색합니다.
GetIcmpStatisticsEx GetIcmpStatisticsEx 함수는 로컬 컴퓨터에 대한 ICMP(Internet Control Message Protocol) 통계를 검색합니다. GetIcmpStatisticsEx 함수는 IPv6 ICMP 통계를 검색할 수 있습니다.
GetIfEntry GetIfEntry 함수는 로컬 컴퓨터에서 지정된 인터페이스에 대한 정보를 검색합니다.
GetIfEntry2 로컬 컴퓨터에서 지정된 인터페이스에 대한 정보를 검색합니다.
GetIfEntry2Ex 로컬 컴퓨터에서 지정된 인터페이스에 대해 지정된 수준의 정보를 검색합니다.

GetIfStackTable 인터페이스 스택에서 네트워크 인터페이스의 관계를 지정하는 네트워크 인터페이스 스택 행 항목의 테이블을 검색합니다.
GetIfTable GetIfTable 함수는 MIB-II 인터페이스 테이블을 검색합니다.
GetIfTable2 MIB-II 인터페이스 테이블을 검색합니다. (GetIfTable2)
GetIfTable2Ex MIB-II 인터페이스 테이블을 검색합니다. (GetIfTable2Ex)
GetInterfaceActiveTimestampCapabilities 네트워크 어댑터의 현재 활성화된 타임스탬프 기능을 검색합니다.
GetInterfaceCurrentTimestampCapabilities 이 함수는 시스템 사용을 위해 예약되어 있으므로 코드에서 호출하면 안 됩니다. (GetInterfaceCurrentTimestampCapabilities)
GetInterfaceDnsSettings *Interface* 매개 변수에 지정된 인터페이스에서 DNS 설정을 검색합니다.
GetInterfaceHardwareTimestampCapabilities 이 함수는 시스템 사용을 위해 예약되어 있으므로 코드에서 호출하면 안 됩니다. (GetInterfaceHardwareTimestampCapabilities)
GetInterfaceInfo GetInterfaceInfo 함수는 로컬 시스템에서 IPv4를 사용하도록 설정된 네트워크 인터페이스 어댑터의 목록을 가져옵니다.
GetInterfaceSupportedTimestampCapabilities 네트워크 어댑터의 지원되는 타임스탬프 기능을 검색합니다.
GetInvertedIfStackTable 인터페이스 스택에서 네트워크 인터페이스의 관계를 지정하는 반전된 네트워크 인터페이스 스택 행 항목의 테이블을 검색합니다.

GetIpAddrTable GetIpAddrTable 함수는 IPv4 주소 매핑 테이블에 대한 인터페이스를 검색합니다.
GetIpErrorString GetIpErrorString 함수는 IP 도우미 오류 문자열을 검색합니다.
GetIpForwardEntry2 로컬 컴퓨터에서 IP 경로 항목에 대한 정보를 검색합니다.
GetIpForwardTable GetIpForwardTable 함수는 IPv4 라우팅 테이블을 검색합니다.
GetIpForwardTable2 GetIpForwardTable2 함수는 로컬 컴퓨터에서 IP 경로 항목을 검색합니다.
GetIpInterfaceEntry 로컬 컴퓨터에서 지정된 인터페이스에 대한 IP 정보를 검색합니다.
GetIpInterfaceTable 로컬 컴퓨터에서 IP 인터페이스 항목을 검색합니다.
GetIpNetEntry2 로컬 컴퓨터에서 인접 IP 주소 항목에 대한 정보를 검색합니다.
GetIpNetTable GetIpNetTable 함수는 IPv4에서 실제 주소 매핑 테이블을 검색합니다.
GetIpNetTable2 GetIpNetTable2 함수는 로컬 컴퓨터에서 IP 인접 테이블을 검색합니다.
GetIpNetworkConnectionBandwidthEstimates 지정된 인터페이스의 네트워크 연결에 대한 기록 대역폭 예상값을 검색합니다.
GetIpPathEntry 로컬 컴퓨터에서 IP 경로 항목에 대한 정보를 검색합니다.

GetIpPathTable GetIpPathTable 함수는 로컬 컴퓨터에서 IP 경로 테이블을 검색합니다.
GetIpStatistics GetIpStatistics 함수는 현재 컴퓨터의 IP 통계를 검색합니다.
GetIpStatisticsEx GetIpStatisticsEx 함수는 현재 컴퓨터에 대한 IP(인터넷 프로토콜) 통계를 검색합니다.
GetMulticastIpAddressEntry 로컬 컴퓨터에서 기존 멀티캐스트 IP 주소 항목에 대한 정보를 검색합니다.
GetMulticastIpAddressTable 로컬 컴퓨터에서 멀티캐스트 IP 주소 테이블을 검색합니다.
GetNetworkConnectivityHint 애플리케이션 또는 서비스에서 발생할 수 있는 네트워크 연결의 집계 수준 및 비용을 검색합니다.
GetNetworkConnectivityHintForInterface 지정된 인터페이스에 대한 네트워크 연결 수준 및 비용을 검색합니다.
GetNetworkParams GetNetworkParams 함수는 로컬 컴퓨터에 대한 네트워크 매개 변수를 검색합니다.
GetNumberOfInterfaces GetNumberOfInterfaces 함수는 로컬 컴퓨터의 인터페이스 수를 검색합니다.
GetOwnerModuleFromTcp6Entry MIB 테이블 행에서 특정 IPv6 TCP 엔드포인트에 대한 컨텍스트 바인딩을 실행한 모듈에 대한 데이터를 검색합니다.
GetOwnerModuleFromTcpEntry MIB 테이블 행에서 특정 IPv4 TCP 엔드포인트에 대한 컨텍스트 바인딩을 실행한 모듈에 대한 데이터를 검색합니다.

GetOwnerModuleFromUdp6Entry MIB 테이블 행에서 특정 IPv6 UDP 엔드포인트에 대한 컨텍스트 바인딩을 실행한 모듈에 대한 데이터를 검색합니다.
GetOwnerModuleFromUdpEntry MIB 테이블 행에서 특정 IPv4 UDP 엔드포인트에 대한 컨텍스트 바인딩을 실행한 모듈에 대한 데이터를 검색합니다.
GetPerAdapterInfo GetPerAdapterInfo 함수는 지정된 인터페이스에 해당하는 어댑터에 대한 정보를 검색합니다.
GetPerTcp6ConnectionEStats IPv6 TCP 연결에 대한 확장 통계를 검색합니다.
GetPerTcpConnectionEStats IPv4 TCP 연결에 대한 확장 통계를 검색합니다.
GetRTTAndHopCount GetRTTAndHopCount 함수는 RTT(왕복 시간) 및 지정된 대상에 대한 홉 수를 결정합니다.
GetTcp6Table IPv6에 대한 TCP 연결 테이블을 검색합니다. (GetTcp6Table)
GetTcp6Table2 IPv6에 대한 TCP 연결 테이블을 검색합니다. (GetTcp6Table2)
GetTcpStatistics GetTcpStatistics 함수는 로컬 컴퓨터에 대한 TCP 통계를 검색합니다.
GetTcpStatisticsEx GetTcpStatisticsEx 함수는 현재 컴퓨터에 대한 TCP(Transmission Control Protocol) 통계를 검색합니다.
GetTcpStatisticsEx2 GetTcpStatisticsEx2 함수는 현재 컴퓨터에 대한 TCP(Transmission Control Protocol) 통계를 검색합니다.

GetTcpTable IPv4 TCP 연결 테이블을 검색합니다. (GetTcpTable)
GetTcpTable2 IPv4 TCP 연결 테이블을 검색합니다. (GetTcpTable2)
GetTeredoPort 로컬 컴퓨터에서 Teredo 클라이언트에서 사용하는 동적 UDP 포트 번호를 검색합니다.
GetUdp6Table IPv6 UDP(사용자 데이터그램 프로토콜) 수신기 테이블을 검색합니다.
GetUdpStatistics GetUdpStatistics 함수는 로컬 컴퓨터에 대한 UDP(사용자 데이터그램 프로토콜) 통계를 검색합니다.
GetUdpStatisticsEx GetUdpStatisticsEx 함수는 현재 컴퓨터에 대한 UDP(사용자 데이터그램 프로토콜) 통계를 검색합니다.
GetUdpStatisticsEx2 GetUdpStatisticsEx2 함수는 현재 컴퓨터에 대한 UDP(사용자 데이터그램 프로토콜) 통계를 검색합니다.
GetUdpTable IPv4 UDP(사용자 데이터그램 프로토콜) 수신기 테이블을 검색합니다.
GetUnicastIpAddressEntry 로컬 컴퓨터에서 기존 유니캐스트 IP 주소 항목에 대한 정보를 검색합니다.
GetUnicastIpAddressTable 로컬 컴퓨터에서 유니캐스트 IP 주소 테이블을 검색합니다.
GetUniDirectionalAdapterInfo GetUniDirectionalAdapterInfo 함수는 로컬 컴퓨터에 설치된 단방향 어댑터에 대한 정보를 검색합니다. 단방향 어댑터는 데이터그램을 수신할 수 있지만 전송할 수 없는 어댑터입니다.

Icmp6CreateFile Icmp6CreateFile 함수는 IPv6 ICMP 에코 요청을 실행할 수 있는 핸들을 엽니다.
Icmp6ParseReplies Icmp6ParseReplies 함수는 제공된 회신 버퍼를 구문 분석하고 IPv6 ICMPv6 에코 응답 응답을 반환합니다(있는 경우).
Icmp6SendEcho2 Icmp6SendEcho2 함수는 IPv6 ICMPv6 에코 요청을 보내고 즉시(Event 또는 ApcRoutine이 NULL이 아닌 경우) 반환하거나 지정된 시간 제한 후에 를 반환합니다. ReplyBuffer에는 IPv6 ICMPv6 에코 응답(있는 경우)이 포함됩니다.
IcmpCloseHandle IcmpCloseHandle 함수는 IcmpCreateFile 또는 Icmp6CreateFile 함수를 호출하여 연 핸들을 닫습니다.
IcmpCreateFile IcmpCreateFile 함수는 IPv4 ICMP 에코 요청을 실행할 수 있는 핸들을 엽니다.
IcmpParseReplies 제공된 회신 버퍼를 구문 분석하고 찾은 ICMP 에코 요청 응답 수를 반환합니다.
IcmpSendEcho IcmpSendEcho 함수는 IPv4 ICMP 에코 요청을 보내고 모든 에코 응답 응답을 반환합니다. 제한 시간이 만료되었거나 회신 버퍼가 채워지면 호출이 반환됩니다.
IcmpSendEcho2 **IcmpSendEcho2** 함수는 IPv4 ICMP 에코 요청을 보내고 즉시(*Event* 또는 *ApcRoutine*이 **NULL**이 아닌 경우) 반환하거나 지정된 시간 제한 후에 를 반환합니다. *ReplyBuffer*에는 ICMP 에코 응답(있는 경우)이 포함됩니다.
IcmpSendEcho2Ex IPv4 ICMP 에코 요청을 보내고 즉시(Event 또는 ApcRoutine이 NULL이 아닌 경우) 반환하거나 지정된 제한 시간 후에 를 반환합니다. ReplyBuffer에는 ICMP 응답(있는 경우)이 포함됩니다.
if_indextoname 네트워크 인터페이스의 로컬 인덱스를 ANSI 인터페이스 이름으로 변환합니다.

if_nametoindex 네트워크 인터페이스의 ANSI 인터페이스 이름을 인터페이스의 로컬 인덱스로 변환합니다.
InitializeIpForwardEntry 로컬 컴퓨터의 IP 경로 항목에 대한 기본값을 사용하여 MIB_IPFORWARD_ROW2 구조를 초기화합니다.
InitializeIpInterfaceEntry 기본값을 사용하여 MIB_IPINTERFACE_ROW 항목의 멤버를 초기화합니다.
InitializeUnicastIpAddressEntry 로컬 컴퓨터의 유니캐스트 IP 주소 항목에 대한 기본값을 사용하여 MIB_UNICASTIPADDRESS_ROW 구조를 초기화합니다.
INTERFACE_TIMESTAMP_CONFIG_CHANGE_CALLBACK 네트워크 어댑터의 타임스탬프 기능에 대한 변경 내용을 알리기 위해 앱에서 구현하는 콜백 함수입니다.
IpReleaseAddress IpReleaseAddress 함수는 DHCP(동적 호스트 구성 프로토콜)를 통해 이전에 가져온 IPv4 주소를 해제합니다.
IpRenewAddress IpRenewAddressfunction은 이전에 DHCP(동적 호스트 구성 프로토콜)를 통해 가져온 IPv4 주소에 대한 임대를 갱신합니다.
LookupPersistentTcpPortReservation 로컬 컴퓨터에서 연속된 TCP 포트 블록에 대한 영구 TCP 포트 예약에 대한 토큰을 조회합니다.
LookupPersistentUdpPortReservation 로컬 컴퓨터에서 연속된 TCP 포트 블록에 대한 영구 UDP 포트 예약에 대한 토큰을 조회합니다.
NhpAllocateAndGetInterfaceInfoFromStack NhpAllocateAndGetInterfaceInfoFromStack 함수는 로컬 컴퓨터에 대한 어댑터 정보를 가져옵니다.
NotifyAddrChange NotifyAddrChange 함수는 IPv4 주소를 인터페이스에 매핑하는 테이블에서 변경이 발생할 때마다 호출자에게 알림을 보냅니다.

NotifyIfTimestampConfigChange 이 함수는 시스템 사용을 위해 예약되어 있으며 코드에서 호출하면 안 됩니다. (NotifyIfTimestampConfigChange)
NotifyIpInterfaceChange 로컬 컴퓨터의 모든 IP 인터페이스, IPv4 인터페이스 또는 IPv6 인터페이스 변경에 대한 알림을 받도록 등록합니다.
NotifyNetworkConnectivityHintChange 집계 네트워크 연결 수준 및 비용 힌트가 변경될 때 호출할 애플리케이션 정의 콜백 함수를 등록합니다.
NotifyRouteChange NotifyRouteChange 함수는 IPv4 라우팅 테이블에서 변경이 발생할 때마다 호출자에게 알림을 보냅니다.
NotifyRouteChange2 로컬 컴퓨터의 IP 경로 항목 변경에 대한 알림을 받도록 등록합니다.
NotifyStableUnicastIpAddressTable 로컬 컴퓨터에서 안정적인 유니캐스트 IP 주소 테이블을 검색합니다.
NotifyTeredoPortChange 로컬 컴퓨터의 Teredo 서비스 포트에 대해 Teredo 클라이언트에서 사용하는 UDP 포트 번호 변경에 대한 알림을 받도록 등록합니다.
NotifyUnicastIpAddressChange 로컬 컴퓨터의 모든 유니캐스트 IP 인터페이스, 유니캐스트 IPv4 주소 또는 유니캐스트 IPv6 주소에 대한 변경 내용에 대한 알림을 받도록 등록합니다.
ParseNetworkString 입력 네트워크 문자열을 구문 분석하고 지정된 IP 네트워크 문자열 형식의 법적 표현인지 확인합니다. 문자열이 형식 및 해당 사양과 일치하는 경우 함수는 필요에 따라 구문 분석된 결과를 반환할 수 있습니다.
PNETWORK_CONNECTIVITY_HINT_CHANGE_CALLBACK 네트워크 집계 연결 수준 및 비용 힌트가 변경되면 애플리케이션 정의 함수가 호출됩니다.

RegisterInterfaceTimestampConfigChange 시스템에서 타임스탬프 기능 변경을 알리기 위해 호출하는 사용자 구현 콜백 함수를 등록합니다.
ResolveIpNetEntry2 로컬 컴퓨터의 인접 IP 주소 항목에 대한 실제 주소를 확인합니다. (ResolveIpNetEntry2)
ResolveNeighbor 로컬 컴퓨터의 인접 IP 주소 항목에 대한 실제 주소를 확인합니다. (ResolveNeighbor)
RestoreMediaSense RestoreMediaSense 함수는 DisableMediaSense 함수가 이전에 호출된 로컬 컴퓨터에서 TCP/IP 스택의 미디어 감지 기능을 복원합니다.
RtlEthernetAddressToStringA 이진 이더넷 주소를 이더넷 MAC 주소의 문자열 표현으로 변환합니다. (ANSI)
RtlEthernetAddressToStringW 이진 이더넷 주소를 이더넷 MAC 주소의 문자열 표현으로 변환합니다. (유니코드)
RtlEthernetStringToAddressA 이더넷 MAC 주소의 문자열 표현을 이더넷 주소의 이진 형식으로 변환합니다. (ANSI)
RtlEthernetStringToAddressW 이더넷 MAC 주소의 문자열 표현을 이더넷 주소의 이진 형식으로 변환합니다. (유니코드)
RtlIpv4AddressToStringA IPv4 주소를 인터넷 표준 점선 소수점 형식의 문자열로 변환합니다. (ANSI)
RtlIpv4AddressToStringExA IPv4 주소 및 포트 번호를 인터넷 표준 형식의 문자열로 변환합니다. (ANSI)
RtlIpv4AddressToStringExW IPv4 주소 및 포트 번호를 인터넷 표준 형식의 문자열로 변환합니다. (유니코드)
RtlIpv4AddressToStringW IPv4 주소를 인터넷 표준 점선 소수점 형식의 문자열로 변환합니다. (유니코드)

RtlIpv4StringToAddressA IPv4 주소의 문자열 표현을 이진 IPv4 주소로 변환합니다. (ANSI)
RtlIpv4StringToAddressExA IPv4 주소 및 포트 번호의 문자열 표현을 이진 IPv4 주소 및 포트로 변환합니다. (ANSI)
RtlIpv4StringToAddressExW IPv4 주소 및 포트 번호의 문자열 표현을 이진 IPv4 주소 및 포트로 변환합니다. (유니코드)
RtlIpv4StringToAddressW IPv4 주소의 문자열 표현을 이진 IPv4 주소로 변환합니다. (유니코드)
RtlIpv6AddressToStringA IPv6 주소를 인터넷 표준 형식의 문자열로 변환합니다. (ANSI)
RtlIpv6AddressToStringExA IPv6 주소, scope ID 및 포트 번호를 문자열로 변환합니다. (ANSI)
RtlIpv6AddressToStringExW IPv6 주소, scope ID 및 포트 번호를 문자열로 변환합니다. (유니코드)
RtlIpv6AddressToStringW IPv6 주소를 인터넷 표준 형식의 문자열로 변환합니다. (유니코드)
RtlIpv6StringToAddressA IPv6 주소의 문자열 표현을 이진 IPv6 주소로 변환합니다. (ANSI)
RtlIpv6StringToAddressExA IPv6 주소, scope ID 및 포트 번호의 문자열 표현을 이진 IPv6 주소, scope ID 및 포트로 변환합니다. (ANSI)
RtlIpv6StringToAddressExW IPv6 주소, scope ID 및 포트 번호의 문자열 표현을 이진 IPv6 주소, scope ID 및 포트로 변환합니다. (유니코드)
RtlIpv6StringToAddressW IPv6 주소의 문자열 표현을 이진 IPv6 주소로 변환합니다. (유니코드)

SendARP SendARP 함수는 지정된 대상 IPv4 주소에 해당하는 실제 주소를 얻기 위해 ARP(주소 확인 프로토콜) 요청을 보냅니다.
SetIfEntry SetIfEntry 함수는 인터페이스의 관리 상태 설정합니다.
SetInterfaceDnsSettings *Settings* 매개 변수에 지정된 인터페이스별 DNS 설정을 설정합니다.
SetIpForwardEntry SetIpForwardEntry 함수는 로컬 컴퓨터의 IPv4 라우팅 테이블에서 기존 경로를 수정합니다.
SetIpForwardEntry2 로컬 컴퓨터에서 IP 경로 항목의 속성을 설정합니다.
SetIpInterfaceEntry 로컬 컴퓨터에서 IP 인터페이스의 속성을 설정합니다.
SetIpNetEntry SetIpNetEntry 함수는 로컬 컴퓨터의 ARP 테이블에 있는 기존 ARP 항목을 수정합니다.
SetIpNetEntry2 로컬 컴퓨터에서 기존 인접 IP 주소 항목의 실제 주소를 설정합니다.
SetIpStatistics SetIpStatistics 함수는 IP 전달을 설정하거나 해제하고 로컬 컴퓨터의 기본 TTL(Time-to-Live) 값을 설정합니다.
SetIpStatisticsEx IP 전달을 설정하거나 해제하고 로컬 컴퓨터의 기본 TTL(TTL) 값을 설정합니다.
SetIpTTL SetIpTTL 함수는 로컬 컴퓨터의 기본 TTL(Time-to-Live) 값을 설정합니다.

SetPerTcp6ConnectionEStats IPv6 TCP 연결에 대한 읽기/쓰기 정보의 값을 설정합니다. 이 함수는 IPv6 TCP 연결에 대한 확장 통계를 사용하거나 사용하지 않도록 설정하는 데 사용됩니다.
SetPerTcpConnectionEStats IPv4 TCP 연결에 대한 읽기/쓰기 정보의 값을 설정합니다. 이 함수는 IPv4 TCP 연결에 대한 확장 통계를 사용하거나 사용하지 않도록 설정하는 데 사용됩니다.
SetTcpEntry SetTcpEntry 함수는 TCP 연결의 상태를 설정합니다.
SetUnicastIpAddressEntry 로컬 컴퓨터에서 기존 유니캐스트 IP 주소 항목의 속성을 설정합니다.
UnenableRouter UnenableRouter 함수는 IPv4 전달을 사용하도록 설정하는 요청 수를 추적하는 참조 수를 감소합니다. 이 참조 수가 0에 도달하면 UnenableRouter는 로컬 컴퓨터에서 IPv4 전달을 해제합니다.
UnregisterInterfaceTimestampConfigChange **RegisterInterfaceTimestampConfigChange** 호출에서 등록한 콜백 함수를 등록 취소하여 타임스탬프 기능 변경에 대한 알림을 취소합니다.

구조체

ARP_SEND_REPLY ARP_SEND_REPLY 구조는 ARP(주소 확인 프로토콜) 회신 메시지에 대한 정보를 저장합니다.
DNS_DOH_SERVER_SETTINGS DNS over-HTTPS 서버에 대해 설명합니다.
DNS_INTERFACE_SETTINGS **SetInterfaceDnsSettings 함수를 호출하거나 GetInterfaceDnsSettings 함수를 호출하여** 지정된 인터페이스에 대해 검색하여 지정된 인터페이스에서 구성할 수 있는 DNS 설정을 나타냅니다. (DNS_INTERFACE_SETTINGS)

DNS_INTERFACE_SETTINGS_EX **SetInterfaceDnsSettings 함수를 호출하거나 GetInterfaceDnsSettings 함수를 호출하여** 지정된 인터페이스에 대해 검색하여 지정된 인터페이스에서 구성할 수 있는 DNS 설정을 나타냅니다. (DNS_INTERFACE_SETTINGS_EX)
DNS_INTERFACE_SETTINGS3 **SetInterfaceDnsSettings 함수를 호출하거나 GetInterfaceDnsSettings 함수를 호출하여** 지정된 인터페이스에 대해 검색하여 지정된 인터페이스에서 구성할 수 있는 DNS 설정을 나타냅니다. (DNS_INTERFACE_SETTINGS3)
DNS_SERVER_PROPERTY **DNS_INTERFACE_SETTINGS3** 구조에서 설정되고 **SetInterfaceDnsSettings** 함수를 통해 구성된 DNS 서버 속성에 대해 설명합니다.
DNS_SERVER_PROPERTY_TYPES DNS 서버 속성에 대한 포인터를 포함합니다. 속성의 형식은 DNS_SERVER_PROPERTY::Type의 값에 따라 달라집니다.
FIXED_INFO_W2KSP1 FIXED_INFO 구조에는 컴퓨터의 모든 인터페이스에서 동일한 정보가 포함됩니다.
ICMP_ECHO_REPLY IPv4 에코 요청에 대한 응답으로 반환된 데이터를 설명합니다.
ICMP_ECHO_REPLY32 64비트 플랫폼의 IPv4 에코 요청에 대한 응답으로 반환된 데이터를 설명합니다.
ICMPV6_ECHO_REPLY_LH IPv6 에코 요청에 대한 응답으로 반환된 데이터를 설명합니다.
IN_ADDR in_addr 구조체는 IPv4 주소를 나타냅니다.
INTERFACE_HARDWARE_CROSSTIMESTAMP 네트워크 어댑터에서 검색된 크로스 타임스탬프를 설명합니다.
INTERFACE_HARDWARE_TIMESTAMP_CAPABILITIES 네트워크 인터페이스 카드(NIC) 하드웨어의 타임스탬프 기능을 설명합니다.

INTERFACE_SOFTWARE_TIMESTAMP_CAPABILITIES NIC 미니포트 드라이버의 소프트웨어 타임스탬프 기능을 설명합니다.
INTERFACE_TIMESTAMP_CAPABILITIES 네트워크 어댑터가 지원하는 정확한 타임스탬프 기능을 설명합니다.
INTERFACE_TIMESTAMP_CAPABILITY_FLAGS 이 구조체는 시스템 사용을 위해 예약되어 있으므로 코드에서 사용하지 않아야 합니다.
IP_ADAPTER_ADDRESSES_LH IP_ADAPTER_ADDRESSES_LH 구조체(iptypes.h)는 특정 어댑터의 연결된 주소 목록에 대한 헤더 노드입니다.
IP_ADAPTER_ADDRESSES_XP IP_ADAPTER_ADDRESSES_XP 구조체(iptypes.h)는 특정 어댑터의 연결된 주소 목록에 대한 헤더 노드입니다.
IP_ADAPTER_ANYCAST_ADDRESS_XP 단일 애니캐스트 IP 주소를 특정 어댑터의 연결된 주소 목록에 저장합니다.
IP_ADAPTER_DNS_SERVER_ADDRESS_XP IP_ADAPTER_DNS_SERVER_ADDRESS 구조는 특정 어댑터에 대한 DNS 서버 주소의 연결된 목록에 단일 DNS 서버 주소를 저장합니다.
IP_ADAPTER_DNS_SUFFIX IP_ADAPTER_DNS_SUFFIX 구조체는 DNS 접미사를 특정 어댑터의 연결된 DNS 접미사 목록에 저장합니다.
IP_ADAPTER_GATEWAY_ADDRESS_LH 특정 어댑터의 연결된 게이트웨이 주소 목록에 단일 게이트웨이 주소를 저장합니다.
IP_ADAPTER_INDEX_MAP IP_ADAPTER_INDEX_MAP 구조체는 네트워크 어댑터의 이름과 함께 IPv4를 사용하도록 설정된 네트워크 어댑터와 연결된 인터페이스 인덱스가 저장됩니다.
IP_ADAPTER_INFO 로컬 컴퓨터의 특정 네트워크 어댑터에 대한 정보를 포함합니다.

IP_ADAPTER_MULTICAST_ADDRESS_XP IP_ADAPTER_MULTICAST_ADDRESS 구조체는 특정 어댑터의 연결된 주소 목록에 단일 멀티캐스트 주소를 저장합니다.
IP_ADAPTER_ORDER_MAP IP_ADAPTER_ORDER_MAP 구조체는 어댑터에 대한 정보 배열과 해당 상대 우선 순위를 로컬 컴퓨터에 저장합니다.
IP_ADAPTER_PREFIX_XP IP 주소 접두사를 저장합니다. 나
IP_ADAPTER_UNICAST_ADDRESS_LH IP_ADAPTER_UNICAST_ADDRESS_LH 구조체(iptypes.h)는 단일 유니캐스트 IP 주소를 특정 어댑터의 연결된 IP 주소 목록에 저장합니다.
IP_ADAPTER_UNICAST_ADDRESS_XP IP_ADAPTER_UNICAST_ADDRESS_XP 구조체(iptypes.h)는 단일 유니캐스트 IP 주소를 특정 어댑터의 연결된 IP 주소 목록에 저장합니다.
IP_ADAPTER_WINS_SERVER_ADDRESS_LH 특정 어댑터에 대한 WINS 서버 주소의 연결된 목록에 단일 WINS(Windows Internet Name Service) 서버 주소를 저장합니다.
IP_ADDR_STRING IPv4 주소의 연결된 목록에 있는 노드를 나타냅니다.
IP_ADDRESS_PREFIX IP 주소 접두사를 저장합니다. (IP_ADDRESS_PREFIX)
IP_ADDRESS_STRING IPv4 주소를 점선 10진수 표기법으로 저장합니다.
IP_INTERFACE_INFO IP_INTERFACE_INFO 구조에는 로컬 시스템에서 IPv4를 사용하도록 설정된 네트워크 인터페이스 어댑터 목록이 포함되어 있습니다.
IP_INTERFACE_NAME_INFO_W2KSP1 로컬 컴퓨터의 IPv4 인터페이스에 대한 정보를 포함합니다.

IP_MCAST_COUNTER_INFO IP_MCAST_COUNTER_INFO 구조는 멀티캐스트 트래픽에 대한 통계 정보를 저장합니다.
IP_OPTION_INFORMATION IP 패킷의 헤더에 포함할 옵션에 대해 설명합니다.
IP_OPTION_INFORMATION32 64비트 플랫폼의 IP 패킷 헤더에 포함할 옵션에 대해 설명합니다.
IP_PER_ADAPTER_INFO_W2KSP1 IP_PER_ADAPTER_INFO 구조체에는 특정 어댑터와 관련된 정보가 포함됩니다.
IP_UNIDIRECTIONAL_ADAPTER_ADDRESS IP_UNIDIRECTIONAL_ADAPTER_ADDRESS 구조체는 단방향 어댑터와 연결된 IPv4 주소를 저장합니다.
IPV6_ADDRESS_EX IPv6 주소를 저장합니다.
NET_ADDRESS_INFO ParseNetworkString 함수에서 반환된 IP 주소 정보를 포함합니다.
NET_LUID_LH 네트워크 인터페이스에 대한 LUID(로컬 고유 식별자)입니다.
NL_BANDWIDTH_INFORMATION TCP/IP 스택에 의해 결정되는 사용 가능한 대역폭 예상 및 관련 분산에 대한 읽기 전용 정보를 포함합니다.
NL_NETWORK_CONNECTIVITY_HINT 네트워크 연결 수준, 네트워크 연결에 대한 사용 요금 및 비용 요소를 반영하는 다른 멤버에 대해 설명합니다.
SOCKADDR_IN6_PAIR 원본 및 대상 주소 쌍을 나타내는 IP 주소 쌍에 대한 포인터를 포함합니다.

SOCKADDR_INET IPv4, IPv6 주소 또는 주소 패밀리를 포함합니다.
TCP_ESTATS_BANDWIDTH_ROD_v0 TCP 연결에 대한 대역폭 추정에 대한 확장된 TCP 통계에 대한 읽기 전용 동적 정보를 포함합니다.
TCP_ESTATS_BANDWIDTH_RW_v0 TCP 연결에 대한 대역폭 추정에 대한 확장된 TCP 통계에 대한 읽기/쓰기 구성 정보를 포함합니다.
TCP_ESTATS_DATA_ROD_v0 TCP 연결에 대한 데이터 전송에 대한 확장된 TCP 통계에 대한 읽기 전용 동적 정보를 포함합니다.
TCP_ESTATS_DATA_RW_v0 TCP 연결에 대한 데이터 전송에 대한 확장된 TCP 통계에 대한 읽기/쓰기 구성 정보를 포함합니다.
TCP_ESTATS_FINE_RTT_ROD_v0 TCP 연결에 대한 세분화된 RTT(왕복 시간) 추정에 대한 확장된 TCP 통계에 대한 읽기 전용 동적 정보를 포함합니다.
TCP_ESTATS_FINE_RTT_RW_v0 TCP 연결에 대한 세분화된 RTT(왕복 시간) 추정 통계에 대한 확장된 TCP 통계에 대한 읽기/쓰기 구성 정보를 포함합니다.
TCP_ESTATS_OBS_REC_ROD_v0 TCP 연결에 대한 원격 수신기에서 관찰된 확장된 TCP 통계에 대한 읽기 전용 동적 정보를 포함합니다.
TCP_ESTATS_OBS_REC_RW_v0 TCP 연결에 대한 원격 수신기에서 관찰되는 확장된 TCP 통계에 대한 읽기/쓰기 구성 정보를 포함합니다.
TCP_ESTATS_PATH_ROD_v0 TCP 연결에 대한 네트워크 경로 측정에 대한 확장된 TCP 통계에 대한 읽기 전용 동적 정보를 포함합니다.
TCP_ESTATS_PATH_RW_v0 TCP 연결의 경로 측정에 대한 확장된 TCP 통계에 대한 읽기/쓰기 구성 정보를 포함합니다.

TCP_ESTATS_REC_ROD_v0 TCP 연결에 대한 로컬 수신기의 확장된 TCP 통계에 대한 읽기 전용 동적 정보를 포함합니다.
TCP_ESTATS_REC_RW_v0 TCP 연결에 대한 로컬 수신기의 확장된 TCP 통계에 대한 읽기/쓰기 구성 정보를 포함합니다.
TCP_ESTATS_SEND_BUFF_ROD_v0 TCP 연결의 출력 큐에 대한 확장된 TCP 통계에 대한 읽기 전용 동적 정보를 포함합니다.
TCP_ESTATS_SEND_BUFF_RW_v0 TCP 연결의 출력 큐에 대한 확장된 TCP 통계에 대한 읽기/쓰기 구성 정보를 포함합니다.
TCP_ESTATS_SND_CONG_ROD_v0 TCP 연결에 대한 보낸 사람 정체 관련 데이터에 대한 확장된 TCP 통계에 대한 읽기 전용 동적 정보를 포함합니다.
TCP_ESTATS_SND_CONG_ROS_v0 TCP 연결의 최대 정체 창에서 확장된 TCP 통계에 대한 읽기 전용 정적 정보를 포함합니다.
TCP_ESTATS_SND_CONG_RW_v0 TCP 연결의 보낸 사람 정체에 대한 확장된 TCP 통계에 대한 읽기/쓰기 구성 정보를 포함합니다.
TCP_ESTATS_SYN_OPTS_ROS_v0 TCP 연결에 대한 SYN 교환의 확장된 TCP 통계에 대한 읽기 전용 정적 정보를 포함합니다.
TCP_RESERVE_PORT_RANGE TCP_RESERVE_PORT_RANGE 구조체는 예약할 TCP 포트 범위를 지정합니다.
TCPIP_OWNER_MODULE_BASIC_INFO TCP 연결과 연결된 모듈 이름 및 모듈 경로 값에 대한 포인터를 포함합니다. TCPIP_OWNER_MODULE_BASIC_INFO 구조체는 GetOwnerModuleFromTcpEntry 및 GetOwnerModuleFromTcp6Entry 함수에 의해 반환됩니다.

iSCSI 검색 라이브러리 API

아티클2023. 03. 09.

iSCSI 검색 라이브러리 API 기술에 대한 개요입니다.

iSCSI 검색 라이브러리 API를 개발하려면 다음 헤더가 필요합니다.

- iscsidsc.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- iSCSI 검색 라이브러리 API

열거형

IKE_AUTHENTICATION_METHOD IKE_AUTHENTICATION_METHOD 열거형은 IKE(인터넷 키 교환) 인증 방법의 유형을 나타냅니다.
ISCSI_AUTH_TYPES ISCSI_AUTH_TYPES 열거형은 사용된 인증 방법의 유형을 나타냅니다.
ISCSI_DIGEST_TYPES ISCSI_DIGEST_TYPES 열거형은 다이제스트 유형을 나타냅니다.
TARGET_INFORMATION_CLASS TARGET_INFORMATION_CLASS 열거형은 GetIScsiTargetInformation 함수가 검색하는 지정된 대상 디바이스에 대한 정보를 지정합니다.
TARGETPROTOCOLTYPE TARGETPROTOCOLTYPE 열거형은 초기자가 대상과 통신하는 데 사용해야 하는 프로토콜 유형을 나타냅니다.

함수

AddIScsiConnectionA AddIscsiConnection 함수는 기존 세션에 새 iSCSI 연결을 추가합니다. (ANSI)
AddIScsiConnectionW AddIscsiConnection 함수는 기존 세션에 새 iSCSI 연결을 추가합니다. (유니코드)
AddIScsiSendTargetPortalA AddIscsiSendTargetPortal 함수는 iSCSI 초기자 서비스가 SendTargets 요청을 전송하는 대상 포털 목록에 정적 대상 포털을 추가합니다. (ANSI)
AddIScsiSendTargetPortalW AddIscsiSendTargetPortal 함수는 iSCSI 초기자 서비스가 SendTargets 요청을 전송하는 대상 포털 목록에 정적 대상 포털을 추가합니다. (유니코드)
AddIScsiStaticTargetA AddIscsiStaticTarget 함수는 iSCSI 초기자에서 사용할 수 있는 정적 대상 목록에 대상을 추가합니다. (ANSI)
AddIScsiStaticTargetW AddIscsiStaticTarget 함수는 iSCSI 초기자에서 사용할 수 있는 정적 대상 목록에 대상을 추가합니다. (유니코드)
AddISNSServerA AddIsnsServer 함수는 iSCSI 초기자 서비스가 대상을 검색하는 데 사용하는 iSNS(Internet Storage Name Service) 서버 목록에 새 서버를 추가합니다. (ANSI)
AddISNSServerW AddIsnsServer 함수는 iSCSI 초기자 서비스가 대상을 검색하는 데 사용하는 iSNS(Internet Storage Name Service) 서버 목록에 새 서버를 추가합니다. (유니코드)
AddPersistentIScsiDeviceA AddPersistentIscsiDevice 함수는 볼륨 디바이스 이름, 드라이브 문자 또는 탑재 지점 기호 링크를 iSCSI 영구 바인딩 볼륨 및 디바이스 목록에 추가합니다. (ANSI)
AddPersistentIScsiDeviceW AddPersistentIscsiDevice 함수는 볼륨 디바이스 이름, 드라이브 문자 또는 탑재 지점 기호 링크를 iSCSI 영구 바인딩 볼륨 및 디바이스 목록에 추가합니다. (유니코드)

AddRadiusServerA AddRadiusServer. (ANSI)
AddRadiusServerW AddRadiusServer. (유니코드)
ClearPersistentIScsiDevices ClearPersistentIscsiDevices 함수는 영구적으로 바인딩된 iSCSI 볼륨 목록에서 모든 볼륨 및 디바이스를 제거합니다.
GetDevicesForIScsiSessionA GetDevicesForIscsiSession 함수는 현재 세션과 연결된 디바이스에 대한 정보를 검색합니다. (ANSI)
GetDevicesForIScsiSessionW GetDevicesForIscsiSession 함수는 현재 세션과 연결된 디바이스에 대한 정보를 검색합니다. (유니코드)
GetIScsiIKEInfoA GetIscsiIKEInfo 함수는 IPsec 정책 및 초기자 Host-Bus 어댑터(HBA)와 연결된 설정된 미리 공유된 키 값을 검색합니다. (ANSI)
GetIScsiIKEInfoW GetIscsiIKEInfo 함수는 IPsec 정책 및 초기자 Host-Bus 어댑터(HBA)와 연결된 설정된 미리 공유된 키 값을 검색합니다. (유니코드)
GetIScsiInitiatorNodeNameA GetIscsiInitiatorNodeName 함수는 로컬 컴퓨터에서 세션을 설정할 때 사용되는 공통 초기자 노드 이름을 검색합니다. (ANSI)
GetIScsiInitiatorNodeNameW GetIscsiInitiatorNodeName 함수는 로컬 컴퓨터에서 세션을 설정할 때 사용되는 공통 초기자 노드 이름을 검색합니다. (유니코드)
GetIScsiSessionListA GetIscsiSessionList 함수는 활성 iSCSI 세션 목록을 검색합니다. (ANSI)
GetIScsiSessionListW GetIscsiSessionList 함수는 활성 iSCSI 세션 목록을 검색합니다. (유니코드)

GetIScsiTargetInformationA GetIscsiTargetInformation 함수는 지정된 대상에 대한 정보를 검색합니다. (ANSI)
GetIScsiTargetInformationW GetIscsiTargetInformation 함수는 지정된 대상에 대한 정보를 검색합니다. (유니코드)
GetIScsiVersionInformation GetIscsiVersionInformation 함수는 초기자 버전에 대한 정보를 검색합니다.
LoginIScsiTargetA LoginIscsiTarget 함수는 표시된 대상을 사용하여 전체 주요 로그인 세션을 설정합니다. (ANSI)
LoginIScsiTargetW LoginIscsiTarget 함수는 표시된 대상을 사용하여 전체 주요 로그인 세션을 설정합니다. (유니코드)
LogoutIScsiTarget LogoutIscsiTarget 루틴은 지정된 로그인 세션을 닫습니다.
RefreshIScsiSendTargetPortalA RefreshIscsiSendTargetPortal 함수는 iSCSI 초기자 서비스에 지정된 대상 포털을 사용하여 검색 세션을 설정하고 SendTargets 요청을 전송하여 iSCSI 초기자 서비스에 대해 검색된 대상 목록을 새로 고치도록 지시합니다. (ANSI)
RefreshIScsiSendTargetPortalW RefreshIscsiSendTargetPortal 함수는 iSCSI 초기자 서비스에 지정된 대상 포털을 사용하여 검색 세션을 설정하고 SendTargets 요청을 전송하여 iSCSI 초기자 서비스에 대해 검색된 대상 목록을 새로 고치도록 지시합니다. (유니코드)
RefreshISNSServerA RefreshIsnsServer 함수는 iSCSI 초기자 서비스에 대해 검색된 대상 목록을 새로 고치도록 표시된 iSNS(Internet Storage Name Service) 서버를 쿼리하도록 iSCSI 초기자 서비스에 지시합니다. (ANSI)
RefreshISNSServerW RefreshIsnsServer 함수는 iSCSI 초기자 서비스에 대해 검색된 대상 목록을 새로 고치도록 표시된 iSNS(Internet Storage Name Service) 서버를 쿼리하도록 iSCSI 초기자 서비스에 지시합니다. (유니코드)

RemoveIScsiConnection RemoveIscsiConnection 함수는 활성 세션에서 연결을 제거합니다.
RemoveIScsiPersistentTargetA RemoveIscsiPersistentTarget 함수는 지정된 하드웨어 초기자 HBA(호스트 버스 어댑터), 초기자 포트 및 대상 포털에 대한 영구 로그인을 제거합니다. (ANSI)
RemoveIScsiPersistentTargetW RemoveIscsiPersistentTarget 함수는 지정된 하드웨어 초기자 HBA(호스트 버스 어댑터), 초기자 포트 및 대상 포털에 대한 영구 로그인을 제거합니다. (유니코드)
RemoveIScsiSendTargetPortalA RemoveIscsiSendTargetPortal 함수는 iSCSI 초기자 서비스가 대상 검색을 위해 SendTargets 요청을 보내는 포털 목록에서 포털을 제거합니다. (ANSI)
RemoveIScsiSendTargetPortalW RemoveIscsiSendTargetPortal 함수는 iSCSI 초기자 서비스가 대상 검색을 위해 SendTargets 요청을 보내는 포털 목록에서 포털을 제거합니다. (유니코드)
RemoveIScsiStaticTargetA RemoveIscsiStaticTarget 함수는 컴퓨터에서 사용할 수 있는 정적 대상 목록에서 대상을 제거합니다. (ANSI)
RemoveIScsiStaticTargetW RemoveIscsiStaticTarget 함수는 컴퓨터에서 사용할 수 있는 정적 대상 목록에서 대상을 제거합니다. (유니코드)
RemoveISNSServerA RemoveIsnsServer 함수는 iSCSI 초기자 서비스가 대상을 검색하는 데 사용하는 iSNS(Internet Storage Name Service) 서버 목록에서 서버를 제거합니다. (ANSI)
RemoveISNSServerW RemoveIsnsServer 함수는 iSCSI 초기자 서비스가 대상을 검색하는 데 사용하는 iSNS(Internet Storage Name Service) 서버 목록에서 서버를 제거합니다. (유니코드)
RemovePersistentIScsiDeviceA RemovePersistentIscsiDevice 함수는 영구적으로 바인딩된 iSCSI 볼륨 목록에서 디바이스 또는 볼륨을 제거합니다. (ANSI)

RemovePersistentIScsiDeviceW RemovePersistentIscsiDevice 함수는 영구적으로 바인딩된 iSCSI 볼륨 목록에서 디바이스 또는 볼륨을 제거합니다. (유니코드)
RemoveRadiusServerA RemoveRadiusServer 함수는 iSCSI 초기자가 구성된 RADIUS 서버 목록에서 RADIUS(원격 인증 전화 접속 사용자 서비스) 서버 항목을 제거합니다. (ANSI)
RemoveRadiusServerW RemoveRadiusServer 함수는 iSCSI 초기자가 구성된 RADIUS 서버 목록에서 RADIUS(원격 인증 전화 접속 사용자 서비스) 서버 항목을 제거합니다. (유니코드)
ReportActiveIScsiTargetMappingsA ReportActiveIscsiTargetMappings 함수는 컴퓨터의 모든 초기자에 대해 현재 활성화된 대상 매핑을 검색합니다. (ANSI)
ReportActiveIScsiTargetMappingsW ReportActiveIscsiTargetMappings 함수는 컴퓨터의 모든 초기자에 대해 현재 활성화된 대상 매핑을 검색합니다. (유니코드)
ReportIScsiInitiatorListA ReportIscsiInitiatorList 함수는 컴퓨터에서 실행 중인 초기자 호스트 버스 어댑터 목록을 검색합니다. (ANSI)
ReportIScsiInitiatorListW ReportIscsiInitiatorList 함수는 컴퓨터에서 실행 중인 초기자 호스트 버스 어댑터 목록을 검색합니다. (유니코드)
ReportIScsiPersistentLoginsA ReportIscsiPersistentLogins 함수는 영구 로그인 대상 목록을 검색합니다. (ANSI)
ReportIScsiPersistentLoginsW ReportIscsiPersistentLogins 함수는 영구 로그인 대상 목록을 검색합니다. (유니코드)
ReportIScsiSendTargetPortalsA ReportIscsiSendTargetPortals 함수는 iSCSI 초기자 서비스가 SendTarget 요청을 사용하여 자동 검색을 수행하는 데 사용하는 대상 포털 목록을 검색합니다. (ANSI)

ReportIScsiSendTargetPortalsExA ReportIscsiSendTargetPortalsEx 함수는 iSCSI 초기자 서비스가 SendTarget 요청을 사용하여 자동 검색을 수행하는 데 사용하는 정적 대상 포털 목록을 검색합니다. (ANSI)
ReportIScsiSendTargetPortalsExW ReportIscsiSendTargetPortalsEx 함수는 iSCSI 초기자 서비스가 SendTarget 요청을 사용하여 자동 검색을 수행하는 데 사용하는 정적 대상 포털 목록을 검색합니다. (유니코드)
ReportIScsiSendTargetPortalsW ReportIscsiSendTargetPortals 함수는 iSCSI 초기자 서비스가 SendTarget 요청을 사용하여 자동 검색을 수행하는 데 사용하는 대상 포털 목록을 검색합니다. (유니코드)
ReportIScsiTargetPortalsA ReportIscsiTargetPortals 함수는 iSCSI 초기자 서비스에서 검색한 대상 포털 정보를 검색합니다. (ANSI)
ReportIScsiTargetPortalsW ReportIscsiTargetPortals 함수는 iSCSI 초기자 서비스에서 검색한 대상 포털 정보를 검색합니다. (유니코드)
ReportIScsiTargetsA ReportIscsiTargets 함수는 iSCSI 초기자 서비스에서 검색한 대상 목록을 검색하고 iSCSI 초기자 서비스에 목록을 새로 고치도록 지시할 수도 있습니다. (ANSI)
ReportIScsiTargetsW ReportIscsiTargets 함수는 iSCSI 초기자 서비스에서 검색한 대상 목록을 검색하고 iSCSI 초기자 서비스에 목록을 새로 고치도록 지시할 수도 있습니다. (유니코드)
ReportISNSServerListA ReportIsnsServerList 함수는 iSCSI 초기자 서비스가 검색된 대상에 대해 쿼리하는 iSNS(Internet Storage Name Service) 서버 목록을 검색합니다. (ANSI)
ReportISNSServerListW ReportIsnsServerList 함수는 iSCSI 초기자 서비스가 검색된 대상에 대해 쿼리하는 iSNS(Internet Storage Name Service) 서버 목록을 검색합니다. (유니코드)
ReportPersistentIScsiDevicesA ReportPersistentIscsiDevices 함수는 영구적으로 바인딩된 볼륨 및 디바이스 목록을 검색합니다. (ANSI)

ReportPersistentIScsiDevicesW ReportPersistentIscsiDevices 함수는 영구적으로 바인딩된 볼륨 및 디바이스 목록을 검색합니다. (유니코드)
ReportRadiusServerListA ReportRadiusServerList 함수는 인증 중에 iSCSI 초기자 서비스에서 사용하는 RADIUS(원격 인증 전화 접속 서비스) 서버 목록을 검색합니다. (ANSI)
ReportRadiusServerListW ReportRadiusServerList 함수는 인증 중에 iSCSI 초기자 서비스에서 사용하는 RADIUS(원격 인증 전화 접속 서비스) 서버 목록을 검색합니다. (유니코드)
SendScsiInquiry SendScsiInquiry 함수는 SCSI INQUIRY 명령을 지정된 대상에 보냅니다.
SendScsiReadCapacity SendScsiReadCapacity 함수는 SCSI READ CAPACITY 명령을 표시된 대상으로 보냅니다.
SendScsiReportLuns SendScsiReportLuns 함수는 SCSI REPORT LUNS 명령을 지정된 대상으로 보냅니다.
SetIScsiGroupPresharedKey SetIscsiGroupPresharedKey 함수는 컴퓨터의 모든 초기자에 대한 기본 그룹 사전 공유 키를 설정합니다.
SetIScsiIKEInfoA SetIscsiIKEInfo 함수는 iSCSI 연결을 수행할 때 사용할 지정된 초기자에 대한 IPsec 정책 및 미리 공유된 키를 설정합니다. (ANSI)
SetIScsiIKEInfoW SetIscsiIKEInfo 함수는 iSCSI 연결을 수행할 때 사용할 지정된 초기자에 대한 IPsec 정책 및 미리 공유된 키를 설정합니다. (유니코드)
SetIScsiInitiatorCHAPSharedSecret SetIscsiInitiatorCHAPSharedSecret 함수는 컴퓨터의 모든 초기자에 대한 기본 CHAP(Challenge Handshake Authentication Protocol) 공유 비밀을 설정합니다.

SetIScsiInitiatorNodeNameA SetIscsiInitiatorNodeName 함수는 컴퓨터의 초기자 노드 이름을 설정합니다. 이 이름은 다른 노드와 통신하는 컴퓨터의 모든 초기자 노드에서 활용됩니다. (ANSI)
SetIScsiInitiatorNodeNameW SetIscsiInitiatorNodeName 함수는 컴퓨터의 초기자 노드 이름을 설정합니다. 이 이름은 다른 노드와 통신하는 컴퓨터의 모든 초기자 노드에서 활용됩니다. (유니코드)
SetIScsiInitiatorRADIUSSharedSecret SetIscsiInitiatorRADIUSSharedSecret 함수는 RADIUS(원격 인증 전화 접속 사용자 서비스) 공유 비밀을 설정합니다.
SetIScsiTunnelModeOuterAddressA SetIscsiTunnelModeOuterAddress 함수는 지정된 포트를 통해 IPsec 터널 모드로 통신할 때 표시된 초기자 HBA(호스트 버스 어댑터)가 사용하는 터널 모드 외부 주소를 설정합니다. (ANSI)
SetIScsiTunnelModeOuterAddressW SetIscsiTunnelModeOuterAddress 함수는 지정된 포트를 통해 IPsec 터널 모드로 통신할 때 표시된 초기자 HBA(호스트 버스 어댑터)가 사용하는 터널 모드 외부 주소를 설정합니다. (유니코드)
SetupPersistentIScsiDevices SetupPersistentIscsiDevices 함수는 컴퓨터에 연결된 iSCSI 대상에 할당된 디바이스 및 볼륨 목록을 빌드하고 iSCSI 초기자 서비스의 비휘발성 캐시에 이 목록을 저장합니다.

구조체

IKE_AUTHENTICATION_INFORMATION IKE_AUTHENTICATION_INFORMATION 구조에는 두 키 관리 디먼 간에 보안 채널을 설정하는 데 사용되는 IKE(인터넷 키 교환) 인증 정보가 포함되어 있습니다.
IKE_AUTHENTICATION_PRESHARED_KEY IKE_AUTHENTICATION_PRESHARED_KEY 구조에는 IKE(인터넷 키 교환) 프로토콜에 사용되는 미리 공유된 키에 대한 정보가 포함되어 있습니다.

ISCSI_CONNECTION_INFOA ISCSI_CONNECTION_INFO 구조에는 연결에 대한 정보가 포함됩니다. (ANSI)
ISCSI_CONNECTION_INFOW ISCSI_CONNECTION_INFO 구조에는 연결에 대한 정보가 포함됩니다. (유니코드)
ISCSI_DEVICE_ON_SESSIONA ISCSI_DEVICE_ON_SESSION 구조는 iSCSI 로그인 세션과 연결된 디바이스를 식별하기 위한 여러 메서드를 지정합니다. (ANSI)
ISCSI_DEVICE_ON_SESSIONW ISCSI_DEVICE_ON_SESSION 구조는 iSCSI 로그인 세션과 연결된 디바이스를 식별하기 위한 여러 메서드를 지정합니다. (유니코드)
ISCSI_LOGIN_OPTIONS ISCSI_LOGIN_OPTIONS 구조는 초기자가 로그인 세션의 특성을 지정하는 데 사용됩니다.
ISCSI_SESSION_INFOA ISCSI_SESSION_INFO. (ANSI)
ISCSI_SESSION_INFOW ISCSI_SESSION_INFO. (유니코드)
ISCSI_TARGET_MAPPINGA ISCSI_TARGET_MAPPING. (ANSI)
ISCSI_TARGET_MAPPINGW ISCSI_TARGET_MAPPING. (유니코드)
ISCSI_TARGET_PORTAL_GROUPA ISCSI_TARGET_PORTAL_GROUP. (ANSI)
ISCSI_TARGET_PORTAL_GROUPW ISCSI_TARGET_PORTAL_GROUP. (유니코드)

ISCSI_TARGET_PORTAL_INFO_EXA ISCSI_TARGET_PORTAL_INFO_EX 구조에는 대상 포털에 대한 로그인 자격 증명에 대한 정보가 포함됩니다. (ANSI)
ISCSI_TARGET_PORTAL_INFO_EXW ISCSI_TARGET_PORTAL_INFO_EX 구조에는 대상 포털에 대한 로그인 자격 증명에 대한 정보가 포함됩니다. (유니코드)
ISCSI_TARGET_PORTAL_INFOA ISCSI_TARGET_PORTAL_INFO 구조에는 대상 포털에 대한 정보가 포함됩니다. (ANSI)
ISCSI_TARGET_PORTAL_INFOW ISCSI_TARGET_PORTAL_INFO 구조에는 대상 포털에 대한 정보가 포함됩니다. (유니코드)
ISCSI_TARGET_PORTALA ISCSI_TARGET_PORTAL 구조에는 포털에 대한 정보가 포함됩니다. (ANSI)
ISCSI_TARGET_PORTALW ISCSI_TARGET_PORTAL 구조에는 포털에 대한 정보가 포함됩니다. (유니코드)
ISCSI_UNIQUE_SESSION_ID ISCSI_UNIQUE_SESSION_ID 구조체는 세션을 고유하게 식별하는 데이터를 포함하는 불투명 엔터티입니다.
ISCSI_VERSION_INFO ISCSI_VERSION_INFO 구조에는 iSCSI 소프트웨어 초기자 및 초기자 서비스의 버전 및 빌드 번호가 포함됩니다.
PERSISTENT_ISCSI_LOGIN_INFOA PERSISTENT_ISCSI_LOGIN_INFO 구조에는 컴퓨터가 부팅된 후 Microsoft iSCSI 초기자 서비스에서 설정한 로그인 세션을 설명하는 정보가 포함되어 있습니다. (ANSI)
PERSISTENT_ISCSI_LOGIN_INFOW PERSISTENT_ISCSI_LOGIN_INFO 구조에는 컴퓨터가 부팅된 후 Microsoft iSCSI 초기자 서비스에서 설정한 로그인 세션을 설명하는 정보가 포함되어 있습니다. (유니코드)

SCSI_LUN_LIST SCSI_LUN_LIST 구조체는 대상 디바이스와 연결된 LUN(논리 단위 번호) 목록을 생성하는 데 사용됩니다.

iSCSI 대상 서버 공급자

아티클2023. 03. 09.

iSCSI 대상 서버 공급자 기술 개요입니다.

iSCSI 대상 서버 공급자 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- iSCSI 대상 서버 공급자

Kernel-Mode 드라이버 참조

아티클2023. 08. 26.

Kernel-Mode 드라이버 참조 기술 개요입니다.

Kernel-Mode 드라이버 참조를 개발하려면 다음 헤더가 필요합니다.

- ntdef.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 커널 모드 드라이버 참조

열거형

COMPARTMENT_ID COMPARTMENT_ID 열거형은 네트워크 라우팅 구획 식별자를 나타냅니다.

Functions

FIELD_OFFSET FIELD_OFFSET 매크로는 알려진 구조체 형식의 명명된 필드의 바이트 오프셋을 반환합니다. (FIELD_OFFSET 매크로)
FIELD_OFFSET FIELD_OFFSET 매크로는 알려진 구조체 형식의 명명된 필드의 바이트 오프셋을 반환합니다. (FIELD_OFFSET 매크로)
InitializeObjectAttributes InitializeObjectAttributes 매크로는 핸들을 여는 루틴에 대한 개체 핸들의 속성을 지정하는 불투명 OBJECT_ATTRIBUTES 구조를 초기화합니다.
NtCurrentTeb NtCurrentTeb 루틴은 현재 스레드의 TEB(스레드 환경 블록)에 대한 포인터를 반환합니다.

RtlIsEcCode 제공된 포인터가 가리키는 코드가 ARM 에뮬레이션 호환(ARM64EC)인지 여부를 나타내는 값을 반환합니다.

구조체

LIST_ENTRY LIST_ENTRY 구조체는 이중으로 연결된 목록의 항목을 설명하거나 이러한 목록의 헤더 역할을 합니다.
Luid LUID 구조체는 로컬 컴퓨터에서 고유하도록 보장되는 식별자를 지정하는 불투명 구조체입니다. 자세한 내용은 Microsoft Windows SDK 설명서의 LUID에 대한 참조 페이지를 참조하세요.
OBJECT_ATTRIBUTES OBJECT_ATTRIBUTES 구조체는 개체 및/또는 반환 핸들을 만드는 루틴에 의해 개체 또는 개체 핸들에 적용할 수 있는 특성을 지정합니다.
SINGLE_LIST_ENTRY SINGLE_LIST_ENTRY 구조체는 적절하게 연결된 목록의 항목을 설명하거나 이러한 목록의 헤더 역할을 합니다.
STRING ANSI_STRING 구조체는 ANSI 문자열에 사용되는 계산된 문자열을 정의합니다.
UNICODE_STRING UNICODE_STRING 구조체는 유니코드 문자열을 정의하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

키보드 및 마우스 입력

키보드 및 마우스 입력 기술 개요입니다.

키보드 및 마우스 입력 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 키보드 및 마우스 입력

열거 목록

테이블 확장

TOOLTIP_DISMISS_FLAGS TOOLTIP_DISMISS_FLAGS 열거형은 도구 설명 해제 알림을 수신하기 위해 창이 등록되었는지 또는 등록 취소되었는지를 나타내는 상수를 정의합니다.

Functions

테이블 확장

_TrackMouseEvent 마우스 포인터가 창을 벗어나거나 지정된 시간 동안 창 위로 마우스를 가져가면 메시지를 게시합니다. 이 함수는 TrackMouseEvent가 있는 경우 호출하고, 그렇지 않으면 에뮬레이트합니다.
ActivateKeyboardLayout 호출 스레드 또는 현재 프로세스에 대한 입력 로캘 식별자(이전의 키보드 레이아웃 핸들)를 설정합니다. 입력 로캘 식별자는 키보드의 실제 레이아웃뿐만 아니라 로캘을 지정합니다.
BlockInput 키보드 및 마우스 입력 이벤트가 애플리케이션에 도달하지 못하도록 차단합니다.
DefRawInputProc RAWINPUTHEADER 구조체의 크기가 올바른지 확인합니다.

DragDetect 마우스를 캡처하고 사용자가 왼쪽 단추를 놓거나, ESC 키를 누르거나, 마우스를 지정된 지점 주위의 끌기 사각형 밖으로 이동할 때까지 해당 움직임을 추적합니다.
EnableWindow 지정한 창이나 컨트롤에 대한 마우스 및 키보드 입력을 사용하거나 사용하지 않도록 설정합니다. 입력을 사용하지 않도록 설정하면 창에 마우스 클릭 및 키 누름과 같은 입력이 수신되지 않습니다. 입력을 사용하도록 설정하면 창에서 모든 입력을 받습니다.
GET_APPCOMMAND_LPARAM 지정된 LPARAM 값에서 애플리케이션 명령을 검색합니다.
GET_DEVICE_LPARAM 지정된 LPARAM 값에서 입력 디바이스 유형을 검색합니다.
GET_FLAGS_LPARAM 지정된 LPARAM 값에서 특정 가상 키의 상태를 검색합니다. (GET_FLAGS_LPARAM)
GET_KEYSTATE_LPARAM 지정된 LPARAM 값에서 특정 가상 키의 상태를 검색합니다. (GET_KEYSTATE_LPARAM)
GET_KEYSTATE_WPARAM 지정된 WPARAM 값에서 특정 가상 키의 상태를 검색합니다.
GET_NCHITTEST_WPARAM 지정된 WPARAM 값에서 적중 테스트 값을 검색합니다.
GET_RAWINPUT_CODE_WPARAM WM_INPUT wParam에서 입력 코드를 검색합니다.
GET_WHEEL_DELTA_WPARAM 지정된 WPARAM 값에서 휠 델타 값을 검색합니다.
GET_XBUTTON_WPARAM 지정된 WPARAM 값에서 특정 단추의 상태를 검색합니다.
GetActiveWindow

호출 스레드의 메시지 큐에 연결된 활성 창에 대한 창 핸들을 검색합니다.
GetAsyncKeyState 함수가 호출될 때 키가 위쪽 또는 아래쪽인지 여부와 GetAsyncKeyState에 대한 이전 호출 후에 키를 눌렀는지 여부를 결정합니다.
GetCapture 마우스를 캡처한 창(있는 경우)에 대한 핸들을 검색합니다. 한 번에 하나의 창만 마우스를 캡처할 수 있습니다. 이 창은 커서가 테두리 내에 있는지 여부에 관계없이 마우스 입력을 받습니다.
GetDoubleClickTime 마우스의 현재 두 번 클릭 시간을 검색합니다.
GetFocus 창이 호출 스레드의 메시지 큐에 연결된 경우 키보드 포커스가 있는 창에 대한 핸들을 검색합니다.
GetKBCodePage 현재 코드 페이지를 검색합니다.
GetKeyboardLayout 활성 입력 로캘 식별자(이전의 키보드 레이아웃)를 검색합니다.
GetKeyboardLayoutList 시스템의 현재 입력 로캘 집합에 해당하는 입력 로캘 식별자(이전의 키보드 레이아웃 핸들)를 검색합니다. 이 함수는 식별자를 지정된 버퍼에 복사합니다.
GetKeyboardLayoutNameA 호출 스레드에 대한 활성 입력 로캘 식별자(이전의 키보드 레이아웃)의 이름을 검색합니다. (ANSI)
GetKeyboardLayoutNameW 호출 스레드에 대한 활성 입력 로캘 식별자(이전의 키보드 레이아웃)의 이름을 검색합니다. (유니코드)
GetKeyboardState 256개 가상 키의 상태를 지정된 버퍼에 복사합니다.
GetKeyboardType 현재 키보드에 대한 정보를 검색합니다.

GetKeyNameTextA 키의 이름을 나타내는 문자열을 검색합니다. (ANSI)
GetKeyNameTextW 키의 이름을 나타내는 문자열을 검색합니다. (유니코드)
GetKeyState 지정된 가상 키의 상태를 검색합니다. 상태는 키가 위쪽, 아래쪽 또는 토글(키를 누를 때마다 번갈아 켜기, 해제) 여부를 지정합니다.
GetLastInputInfo 마지막 입력 이벤트의 시간을 검색합니다.
GetMouseMovePointsEx 마우스 또는 펜의 최대 64개의 이전 좌표 기록을 검색합니다.
GetRawInputBuffer 원시 입력 데이터의 버퍼링된 읽기를 수행합니다.
GetRawInputData 지정된 디바이스에서 원시 입력을 검색합니다.
GetRawInputDeviceInfoA 원시 입력 디바이스에 대한 정보를 검색합니다. (ANSI)
GetRawInputDeviceInfoW 원시 입력 디바이스에 대한 정보를 검색합니다. (유니코드)
GetRawInputDeviceList 시스템에 연결된 원시 입력 디바이스를 열거합니다.
GetRegisteredRawInputDevices 현재 애플리케이션의 원시 입력 디바이스에 대한 정보를 검색합니다.
IsWindowEnabled 마우스 및 키보드 입력에 대해 지정된 창을 사용할 수 있는지 여부를 결정합니다.

keybd_event 키 입력을 합성합니다.
LoadKeyboardLayoutA 새 입력 로캘 식별자(이전의 키보드 레이아웃)를 시스템에 로드합니다. (ANSI)
LoadKeyboardLayoutW 새 입력 로캘 식별자(이전의 키보드 레이아웃)를 시스템에 로드합니다. (유니코드)
MapVirtualKeyA 가상 키 코드를 검색 코드 또는 문자 값으로 변환(매핑)하거나 스캔 코드를 가상 키 코드로 변환합니다. (ANSI)
MapVirtualKeyExA 가상 키 코드를 검색 코드 또는 문자 값으로 변환(매핑)하거나 스캔 코드를 가상 키 코드로 변환합니다. 이 함수는 입력 언어 및 입력 로캘 식별자를 사용하여 코드를 번역합니다. (ANSI)
MapVirtualKeyExW 가상 키 코드를 검색 코드 또는 문자 값으로 변환(매핑)하거나 스캔 코드를 가상 키 코드로 변환합니다. 이 함수는 입력 언어 및 입력 로캘 식별자를 사용하여 코드를 번역합니다. (유니코드)
MapVirtualKeyW 가상 키 코드를 검색 코드 또는 문자 값으로 변환(매핑)하거나 스캔 코드를 가상 키 코드로 변환합니다. (유니코드)
mouse_event mouse_event 함수는 마우스 동작과 단추 클릭을 합성합니다.
NEXTRAWINPUTBLOCK RAWINPUT 구조체 배열에서 다음 구조체의 위치를 검색합니다.
OemKeyScan OEMASCII 코드 0부터 0x0FF OEM 검사 코드 및 시프트 상태에 매핑합니다. 이 함수는 프로그램이 키보드 입력을 시뮬레이션하여 OEM 텍스트를 다른 프로그램으로 보낼 수 있도록 하는 정보를 제공합니다.
RegisterForTooltipDismissNotification 앱 또는 UI 프레임워크에서 도구 설명 창을 해제하라는 알림을 받을 수 있도록 창을 등록 및 등록 취소할 수 있습니다.

RegisterHotKey 시스템 전체 핫 키를 정의합니다.
RegisterRawInputDevices 원시 입력 데이터를 제공하는 디바이스를 등록합니다.
ReleaseCapture 현재 스레드의 창에서 마우스 캡처를 해제하고 일반적인 마우스 입력 처리를 복원합니다.
SendInput 키 입력, 마우스 동작 및 단추 클릭을 합성합니다.
SetActiveWindow 창을 활성화합니다. 창은 호출 스레드의 메시지 큐에 연결되어야 합니다.
SetCapture 마우스 캡처를 현재 스레드에 속한 지정된 창으로 설정합니다.
SetDoubleClickTime 마우스의 두 번 클릭 시간을 설정합니다.
SetFocus 키보드 포커스를 지정된 창으로 설정합니다. 창은 호출 스레드의 메시지 큐에 연결되어야 합니다.
SetKeyboardState 키보드 키 상태 배열을 호출 스레드의 키보드 입력 상태 테이블에 복사합니다. GetKeyboardState 및 GetKeyState 함수에서 액세스하는 것과 동일한 테이블입니다. 이 테이블을 변경해도 다른 스레드에 대한 키보드 입력에는 영향을 주지 않습니다.
SwapMouseButton 마우스 왼쪽 단추와 오른쪽 단추의 의미를 반대로 되돌리거나 복원합니다.
ToAscii 지정된 가상 키 코드 및 키보드 상태를 해당 문자 또는 문자로 변환합니다.
ToAsciiEx

지정된 가상 키 코드 및 키보드 상태를 해당 문자 또는 문자로 변환합니다. 이 함수는 입력 언어 및 입력 로캘 식별자에 의해 식별된 실제 키보드 레이아웃을 사용하여 코드를 변환합니다.
ToUnicode 지정된 가상 키 코드 및 키보드 상태를 해당 유니코드 문자 또는 문자로 변환합니다. (ToUnicode)
ToUnicodeEx 지정된 가상 키 코드 및 키보드 상태를 해당 유니코드 문자 또는 문자로 변환합니다. (ToUnicodeEx)
TrackMouseEvent 마우스 포인터가 창을 벗어나거나 지정된 시간 동안 창 위로 마우스를 가져가면 메시지를 게시합니다.
UnloadKeyboardLayout 입력 로캘 식별자(이전의 키보드 레이아웃이라고 함)를 언로드합니다.
UnregisterHotKey 호출 스레드에서 이전에 등록한 핫 키를 해제합니다.
VkKeyScanA 문자를 현재 키보드의 해당 가상 키 코드 및 시프트 상태로 변환합니다. (ANSI)
VkKeyScanExA 문자를 해당 가상 키 코드 및 시프트 상태로 변환합니다. 이 함수는 입력 언어 및 입력 로캘 식별자로 식별된 실제 키보드 레이아웃을 사용하여 문자를 변환합니다. (ANSI)
VkKeyScanExW 문자를 해당 가상 키 코드 및 시프트 상태로 변환합니다. 이 함수는 입력 언어 및 입력 로캘 식별자로 식별된 실제 키보드 레이아웃을 사용하여 문자를 변환합니다. (유니코드)
VkKeyScanW 문자를 현재 키보드의 해당 가상 키 코드 및 시프트 상태로 변환합니다. (유니코드)

구조

테이블 확장

HARDWAREINPUT 키보드 또는 마우스 이외의 입력 디바이스에서 생성된 시뮬레이션된 메시지에 대한 정보를 포함합니다.
입력 SendInput에서 키 입력, 마우스 이동 및 마우스 클릭과 같은 입력 이벤트를 합성하기 위한 정보를 저장하는 데 사용됩니다.
KEYBDINPUT 시뮬레이션된 키보드 이벤트에 대한 정보를 포함합니다.
LASTINPUTINFO 마지막 입력 시간을 포함합니다.
MOUSEINPUT 시뮬레이션된 마우스 이벤트에 대한 정보를 포함합니다.
MOUSEMOVEPOINT 화면 좌표에서 마우스의 위치에 대한 정보를 포함합니다.
RAWHID HID(휴먼 인터페이스 디바이스)의 원시 입력 형식을 설명합니다.
RAWINPUT 디바이스의 원시 입력을 포함합니다.
RAWINPUTDEVICE 원시 입력 디바이스에 대한 정보를 정의합니다.
RAWINPUTDEVICELIST 원시 입력 디바이스에 대한 정보를 포함합니다.
RAWINPUTHEADER 원시 입력 데이터의 일부인 헤더 정보를 포함합니다.
RAWKEYBOARD 키보드의 상태에 대한 정보를 포함합니다.

RAWMOUSE 마우스 상태에 대한 정보를 포함합니다.
RID_DEVICE_INFO 모든 디바이스에서 들어오는 원시 입력 데이터를 정의합니다.
RID_DEVICE_INFO_HID 지정된 HID(휴먼 인터페이스 디바이스)에서 들어오는 원시 입력 데이터를 정의합니다.
RID_DEVICE_INFO_KEYBOARD 지정된 키보드에서 들어오는 원시 입력 데이터를 정의합니다.
RID_DEVICE_INFO_MOUSE 지정된 마우스에서 들어오는 원시 입력 데이터를 정의합니다.
TRACKMOUSEEVENT TrackMouseEvent 함수에서 마우스 포인터가 창을 벗어나거나 지정된 시간 동안 창 위로 마우스를 가져가는 시기를 추적하는 데 사용됩니다.

Last updated on 2025. 10. 08.

레거시 Windows 환경 기능

아티클2023. 08. 22.

레거시 Windows 환경 기능 기술 개요입니다.

레거시 Windows 환경 기능을 개발하려면 다음 헤더가 필요합니다.

- emptyvc.h
- mmcobj.h
- provider.h
- reconcil.h
- searchapi.h
- shldisp.h
- shlobj_core.h
- windowsdefender.h
- wmiutils.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 레거시 Windows 환경 기능

열거형

_ColumnSortOrder IResultsViewer::SortOrderProperty에서 쿼리를 정렬하는 방법을 나타내거나 설정하는 데 사용됩니다.

Functions

AddDesktopItem 데스크톱 항목을 추가합니다.
AddDesktopItemWithUI 데스크톱 항목의 추가를 확인하는 사용자 인터페이스를 표시하고, 보안 영역 권한을 확인하고, 사용자가 구독을 만들 것인지 묻는 후 활성 데스크톱에 데스크톱 항목을 추가합니다.

AddUrl 지정된 URL과 연결된 데스크톱 항목을 추가합니다.
ApplyChanges 활성 데스크톱에 변경 내용을 적용하고 레지스트리에 저장합니다.
비활성화 디스크 정리 관리자가 종료되고 있음을 처리기에 알려 줍니다.
GenerateDesktopItemHtml 지정된 데스크톱 항목이 포함된 일반 HTML 페이지를 생성합니다.
GetDesktopItem 지정된 데스크톱 항목을 가져옵니다.
GetDesktopItemByID 지정된 ID와 일치하는 데스크톱 항목을 가져옵니다.
GetDesktopItemBySource 원본 URL을 사용하여 데스크톱 항목을 가져옵니다.
GetDesktopItemCount 데스크톱 항목의 수를 가져옵니다.
GetDesktopItemOptions 데스크톱 항목에 대한 옵션을 가져옵니다.
GetPattern 현재 패턴을 가져옵니다.
GetProgressFeedbackMaxEstimate 조정을 완료하는 데 필요한 예상 작업량 측정값을 검색합니다.
GetSpaceUsed 디스크 정리 처리기가 해제할 수 있는 디스크 공간의 양을 요청합니다.
GetWallpaper

현재 배경 화면을 가져옵니다.
GetWallpaperOptions 배경 화면 옵션을 가져옵니다.
초기화 지정된 레지스트리 키 아래에 저장된 정보에 따라 디스크 정리 처리기를 초기화합니다.
InitializeEx 디스크 정리 처리기를 초기화합니다. 초기화보다 지역화에 대한 더 나은 지원을 제공합니다.
ModifyDesktopItem 데스크톱 항목을 수정합니다.
제거 처리기에 불필요한 파일 삭제를 시작하도록 알 수 있습니다.
PurgeProgress 삭제 가능한 파일 제거 진행률에 따라 디스크 정리 관리자를 업데이트하기 위해 디스크 정리 처리기에 의해 주기적으로 호출됩니다.
Reconcile 개체의 상태를 하나 이상의 다른 개체와 조정합니다. 조정은 모든 개체의 상태를 병합하여 결합된 상태를 형성하여 개체의 내부 상태를 업데이트합니다.
RemoveDesktopItem 바탕 화면에서 지정된 데스크톱 항목을 제거합니다.
ScanProgress 삭제 가능한 파일 검색 진행률에 대한 디스크 정리 관리자를 업데이트하기 위해 디스크 정리 처리기에서 호출됩니다.
SetAbortCallback 초기자가 조정을 비동기적으로 종료할 수 있는 개체를 설정합니다. 서류 가방 조정자는 일반적으로 긴 조정 또는 사용자 상호 작용을 포함 하는 조정에 대 한이 개체를 설정 합니다.
SetDesktopItemOptions 항목의 옵션을 설정합니다.

SetPattern 활성 데스크톱 패턴을 설정합니다.
SetProgressFeedback 서류 가방 조정자가 조정을 완료하기 위해 수행한 진행률을 나타냅니다.
SetSafeMode Microsoft Active Desktop을 안전 모드로 설정하거나 업데이트합니다.
SetWallpaper 활성 데스크톱의 배경 화면을 설정합니다.
SetWallpaperOptions 배경 화면 옵션을 설정합니다.
ShowProperties 처리기에 UI를 표시하도록 알 수 있습니다.
UpdateAllDesktopSubscriptions UpdateAllDesktopSubscriptions 함수를 호출하여 데스크톱 구독을 업데이트합니다.
WDEnable Windows Defender 상태 변경 내용을 켜거나 끕니다.
WDStatus Windows Defender 현재 상태 반환합니다.

인터페이스

IActiveDesktop 클라이언트 프로그램에서 로컬 컴퓨터의 데스크톱 항목과 배경 화면을 관리할 수 있습니다.
IActiveDesktopP Windows 데스크톱을 관리하는 메서드를 노출합니다.

IADesktopP2 Windows 데스크톱을 관리하는 메서드를 제공합니다.
IEmptyVolumeCache 디스크 정리 관리자가 디스크 정리 처리기와 통신하는 데 사용됩니다. 관리자가 처리기에서 정보를 요청하고 검사 시작 또는 제거와 같은 이벤트를 알릴 수 있는 메서드를 노출합니다.
IEmptyVolumeCache2 IEmptyVolumeCache를 확장합니다. 이 인터페이스는 IEmptyVolumeCache::Initialize보다 더 나은 지역화 지원을 제공하는 하나의 추가 메서드 InitializeEx를 정의합니다.
IEmptyVolumeCacheCallBack 디스크 정리 처리기에서 디스크 정리 관리자와 통신하는 데 사용되는 메서드를 노출합니다.
IReconcilableObject 지정된 문서를 조정하는 메서드를 노출합니다. 서류 가방 조정자는 이 인터페이스 구현을 담당합니다.
IReconcileInitiator 초기자에게 진행 상황을 알리고, 종료 개체를 설정하고, 지정된 버전의 문서를 요청하는 수단을 서류 가방 조정자에게 제공하는 메서드를 노출합니다. 초기자는 이 인터페이스를 구현해야 합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Lightweight Directory Access Protocol

아티클2023. 03. 09.

경량 디렉터리 액세스 프로토콜 기술 개요입니다.

경량 디렉터리 액세스 프로토콜을 개발하려면 다음 헤더가 필요합니다.

- winber.h
- winldap.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Lightweight Directory Access Protocol

Functions

ber_alloc_t 새 BerElement 구조를 할당하고 생성합니다.
ber_bvdup ber_bvdup 함수는 제공된 berval 구조의 복사본을 만듭니다.
ber_bvecfree ber_bvecfree 함수는 버벌 구조의 배열을 해제합니다.
ber_bvfree ber_bvfree 함수(winldap.h)는 BER(기본 인코딩 규칙)에 따라 인코딩되는 임의의 이진 데이터를 나타내는 berval 구조를 해제합니다.
ber_bvfree ber_bvfree 함수(winber.h)는 버벌 구조를 해제합니다.
ber_first_element ber_first_element 함수는 제공된 BerElement 구조에 저장된 SET, SET OF, SEQUENCE 또는 SEQUENCE OF 데이터 값의 통과를 시작하는 데 사용됩니다. 첫 번째 요소의 태그와 길이를 반환합니다.

ber_flatten ber_flatten 함수는 제공된 BerElement 구조에서 가져온 데이터를 포함하는 새 berval 구조를 할당합니다.
ber_free ber_free 함수는 이전에 ber_alloc_t, ber_init 또는 ldap_first_attribute/ldap_next_attribute 검색 함수를 사용하여 할당된 BerElement 구조를 해제합니다.
ber_init ber_init 함수는 제공된 berval 구조에서 가져온 데이터를 포함하는 새 BerElement 구조를 할당합니다.
ber_next_element ber_next_element 함수는 제공된 BerElement 구조에 저장된 SET, SET OF, SEQUENCE 또는 SEQUENCE OF 데이터 값을 트래버스하기 위해 ber_first_element 함께 사용됩니다. 생성된 형식에서 다음 요소의 태그 및 길이를 반환합니다.
ber_peek_tag 제공된 BerElement 구조체에서 구문 분석할 다음 요소의 태그를 반환합니다.
ber_printf BER 요소를 인코딩하는 데 사용되며 sprintf_s 비슷합니다.
ber_scanf ber_scanf 함수는 SSCANF_S 비슷한 방식으로 BER 요소를 디코딩합니다.
ber_skip_tag ber_skip_tag 함수는 현재 태그를 건너뛰고 제공된 BerElement 구조에서 다음 요소의 태그를 반환합니다.
cldap_open cldap_open 함수(winldap.h)는 연결이 없는 UDP(사용자 데이터그램 프로토콜) 서비스를 통해 LDAP 서버와 세션을 설정합니다.
cldap_openA 연결 없는 UDP(사용자 데이터그램 프로토콜) 서비스를 통해 LDAP 서버로 세션을 설정합니다. (cldap_openA)

cldap_openW cldap_openW(유니코드) 함수(winldap.h)는 연결되지 않은 UDP(사용자 데이터그램 프로토콜) 서비스를 통해 LDAP 서버와 세션을 설정합니다.
ldap_abandon 클라이언트는 ldap_abandon 호출하여 in-process 비동기 LDAP 호출을 취소합니다.
ldap_add ldap_add 함수(winldap.h)는 디렉터리 트리에 비동기 추가 작업을 시작합니다.
ldap_add_ext ldap_add_ext(winldap.h) 함수는 트리에 비동기 추가 작업을 시작합니다.
ldap_add_ext_s ldap_add_ext_s 함수(winldap.h)는 트리에 동기 추가 작업을 시작합니다.
ldap_add_ext_sA ldap_add_ext_s 함수는 트리에 동기 추가 작업을 시작합니다. 추가 작업이 성공하려면 추가된 항목의 부모가 있어야 하거나 부모가 비어 있어야 합니다(루트의 고유 이름과 같음). (ldap_add_ext_sA)
ldap_add_ext_sW ldap_add_ext_sW(유니코드) 함수(winldap.h)는 트리에 동기 추가 작업을 시작합니다.
ldap_add_extA ldap_add_ext 함수는 트리에 비동기 추가 작업을 시작합니다. 추가된 항목의 부모가 있어야 합니다. 또는 추가 작업이 성공하려면 부모가 비어 있어야 합니다(루트의 고유 이름과 같음). (ldap_add_extA)
ldap_add_extW ldap_add_extW(유니코드) 함수(winldap.h)는 트리에 비동기 추가 작업을 시작합니다.
ldap_add_s ldap_add_s 함수(winldap.h)는 트리에 항목을 추가하는 동기 추가 작업을 시작합니다.
ldap_add_sA ldap_add_s 함수는 트리에 항목을 추가하는 동기 추가 작업을 시작합니다. 추가되는 항목의 부모가 이미 있거나 추가 작업이 성공하려면 부모가 비어 있어야 합니다(루트 고유 이름과 같음). (ldap_add_sA)

ldap_add_sW ldap_add_sW(유니코드) 함수(winldap.h)는 트리에 항목을 추가하는 동기 추가 작업을 시작합니다.
ldap_addA 디렉터리 트리에 비동기 추가 작업을 시작합니다. (ldap_addA)
ldap_addW ldap_addW(유니코드) 함수(winldap.h)는 디렉터리 트리에 비동기 추가 작업을 시작합니다.
ldap_bind ldap_bind 함수(winldap.h)는 LDAP 서버를 사용하여 클라이언트를 비동기적으로 인증합니다.
ldap_bind_s ldap_bind_s 함수(winldap.h)는 LDAP 서버에 대한 클라이언트를 동기적으로 인증합니다.
ldap_bind_sA 클라이언트를 LDAP 서버에 동기적으로 인증합니다. (ldap_bind_sA)
ldap_bind_sW ldap_bind_sW(유니코드) 함수(winldap.h)는 LDAP 서버에 대한 클라이언트를 동기적으로 인증합니다.
ldap_bindA LDAP 서버를 사용하여 클라이언트를 비동기적으로 인증합니다. (ldap_bindA)
ldap_bindW ldap_bindW(유니코드) 함수(winldap.h)는 LDAP 서버를 사용하여 클라이언트를 비동기적으로 인증합니다.
ldap_check_filterA ldap_check_filter 함수는 필터 구문을 확인하는 데 사용됩니다. (ANSI)
ldap_check_filterW ldap_check_filter 함수는 필터 구문을 확인하는 데 사용됩니다. (유니코드)

ldap_cleanup 경고 ldap_cleanup 함수는 DLL 언로드 시 예기치 않은 동작을 일으킬 수 있으므로 wldap32.dll 동적으로 로드하고 언로드할 때 리소스를 안전하게 정리할 수 없습니다. 이 때문에 라이브러리의 언로드 시 리소스 누수가 발생할 수 있습니다. 따라서 ldap_cleanup 사용하는 것은 권장되지 않으며 사용자 고유의 위험에 처해 있습니다. .
ldap_close_extended_op ldap_close_extended_op 함수는 ldap_extended_operation 호출하여 만들어진 요청을 종료합니다.
ldap_compare ldap_compare 함수(winldap.h)는 지정된 항목의 특성에 알려진 값이 있는지 여부를 결정합니다.
ldap_compare_ext ldap_compare_ext 함수(winldap.h)는 지정된 항목에 대한 특성이 알려진 값을 보유하는지 여부를 결정합니다.
ldap_compare_ext_s ldap_compare_ext_s 함수(winldap.h)는 지정된 항목에 대한 특성이 알려진 값을 보유하는지 여부를 결정합니다.
ldap_compare_ext_sA ldap_compare_ext_s 함수를 사용하여 지정된 항목의 특성에 알려진 값이 있는지 확인합니다. (ldap_compare_ext_sA)
ldap_compare_ext_sW ldap_compare_ext_sW(유니코드) 함수(winldap.h)는 지정된 항목에 대한 특성이 알려진 값을 보유하는지 여부를 결정합니다.
ldap_compare_extA ldap_compare_ext 함수를 사용하여 지정된 항목의 특성에 알려진 값이 있는지 확인합니다. (ldap_compare_extA)
ldap_compare_extW ldap_compare_extW(유니코드) 함수(winldap.h)는 지정된 항목에 대한 특성이 알려진 값을 보유하는지 여부를 결정합니다.
ldap_compare_s ldap_compare_s 함수(winldap.h)는 지정된 항목에 대한 특성이 알려진 값을 보유하는지 여부를 결정합니다.

ldap_compare_sA ldap_compare_s 함수를 사용하여 지정된 항목의 특성에 알려진 값이 있는지 여부를 확인합니다. (ldap_compare_sA)
ldap_compare_sW ldap_compare_sW(유니코드) 함수(winldap.h)는 지정된 항목의 특성에 알려진 값이 있는지 여부를 결정합니다.
ldap_compareA ldap_compare 함수를 사용하여 지정된 항목의 특성에 알려진 값이 있는지 여부를 확인합니다. (ldap_compareA)
ldap_compareW ldap_compareW(유니코드) 함수(winldap.h)는 지정된 항목에 대한 특성이 알려진 값을 보유하는지 여부를 결정합니다.
ldap_conn_from_msg 특정 메시지에 대한 LDAP 세션 핸들(연결 포인터)을 반환합니다.
ldap_connect ldap_connect 함수는 서버와의 연결을 설정합니다.
ldap_control_free ldap_control_free 함수(winldap.h)는 LDAPControl 구조를 해제합니다.
ldap_control_freeA ldap_control_free 함수는 LDAPControl 구조를 해제합니다. (ldap_control_freeA)
ldap_control_freeW ldap_control_freeW(유니코드) 함수(winldap.h)는 LDAPControl 구조를 해제합니다.
ldap_controls_free ldap_controls_free 함수(winldap.h)는 LDAPControl 구조의 배열을 해제합니다.
ldap_controls_freeA LDAPControl 구조의 배열을 해제합니다. (ldap_controls_freeA)

ldap_controls_freeW ldap_controls_freeW(유니코드) 함수(winldap.h)는 LDAPControl 구조의 배열을 해제합니다.
ldap_count_entries ldap_count_entries 함수는 서버가 반환한 검색 항목 수를 계산합니다.
ldap_count_references ldap_count_references 함수는 검색 요청에 대한 응답으로 서버에서 반환한 하위 참조 수를 계산합니다.
ldap_count_values ldap_count_values 함수(winldap.h)는 목록의 값 수를 계산합니다.
ldap_count_values_len 목록의 값 수를 계산합니다.
ldap_count_valuesA ldap_count_values 함수는 목록의 값 수를 계산합니다. (ldap_count_valuesA)
ldap_count_valuesW ldap_count_valuesW(유니코드) 함수(winldap.h)는 목록의 값 수를 계산합니다.
ldap_create_page_control ldap_create_page_control 함수(winldap.h)는 페이징 결과에 대한 기본 컨트롤을 만듭니다.
ldap_create_page_controlA ldap_create_page_control 함수를 사용하여 페이징 결과에 대한 기본 컨트롤을 만듭니다. 컨트롤에 대한 지원은 LDAP 3에서 유효하지만 페이지 컨트롤이 지원되는지 여부는 특정 서버에 종속됩니다. (ldap_create_page_controlA)
ldap_create_page_controlW ldap_create_page_controlW(유니코드) 함수(winldap.h)는 페이징 결과에 대한 기본 컨트롤을 만듭니다.
ldap_create_sort_control ldap_create_sort_control 함수(winldap.h)는 정렬 키 목록을 검색 컨트롤로 서식을 지정하는 데 사용됩니다.

ldap_create_sort_controlA ldap_create_sort_control 함수는 정렬 키 목록을 검색 컨트롤에 서식을 지정하는 데 사용됩니다. 컨트롤에 대한 지원은 LDAP 3에서 유효하지만 정렬 컨트롤이 지원되는지 여부는 특정 서버에 종속됩니다. (ldap_create_sort_controlA)
ldap_create_sort_controlW ldap_create_sort_controlW(유니코드) 함수(winldap.h)는 정렬 키 목록을 검색 컨트롤에 서식을 지정하는 데 사용됩니다.
ldap_create_vlv_controlA ldap_create_vlv_control 함수는 서버에서 요청 컨트롤(LDAP_CONTROL_VLVREQUEST)을 만드는 데 사용됩니다. (ANSI)
ldap_create_vlv_controlW ldap_create_vlv_control 함수는 서버에서 요청 컨트롤(LDAP_CONTROL_VLVREQUEST)을 만드는 데 사용됩니다. (유니코드)
ldap_delete ldap_delete 함수(winldap.h)는 디렉터리 트리에서 항목을 삭제합니다.
ldap_delete_ext ldap_delete_ext 함수(winldap.h)는 디렉터리 트리에서 리프 항목을 제거하는 확장 루틴입니다.
ldap_delete_ext_s ldap_delete_ext_s 함수(winldap.h)는 디렉터리 트리에서 리프 항목을 제거하는 동기 작업을 수행하는 확장 루틴입니다.
ldap_delete_ext_sA ldap_delete_ext_s 함수는 디렉터리 트리에서 리프 항목을 제거하는 동기 작업을 수행하는 확장 루틴입니다. (ldap_delete_ext_sA)
ldap_delete_ext_sW ldap_delete_ext_sW(유니코드) 함수(winldap.h)는 디렉터리 트리에서 리프 항목을 제거하는 동기 작업을 수행하는 확장 루틴입니다.
ldap_delete_extA ldap_delete_ext 함수는 디렉터리 트리에서 리프 항목을 제거하는 확장 루틴입니다. (ldap_delete_extA)

ldap_delete_extW ldap_delete_extW(유니코드) 함수(winldap.h)는 디렉터리 트리에서 리프 항목을 제거하는 확장 루틴입니다.
ldap_delete_s ldap_delete_s 함수(winldap.h)는 디렉터리 트리에서 리프 항목을 제거하는 동기 작업입니다.
ldap_delete_sA ldap_delete_s 함수는 디렉터리 트리에서 리프 항목을 제거하는 동기 연산입니다. (ldap_delete_sA)
ldap_delete_sW ldap_delete_sW(유니코드) 함수(winldap.h)는 디렉터리 트리에서 리프 항목을 제거하는 동기 작업입니다.
ldap_deleteA ldap_delete 함수는 디렉터리 트리에서 항목을 삭제합니다. (ldap_deleteA)
ldap_deleteW ldap_deleteW(유니코드) 함수(winldap.h)는 디렉터리 트리에서 항목을 삭제합니다.
ldap_dn2ufn ldap_dn2ufn 함수(winldap.h)는 고유 이름을 사용자에게 친숙한 형식으로 변환합니다.
ldap_dn2ufnA 고유 이름을 사용자에게 친숙한 형식으로 변환합니다. (ldap_dn2ufnA)
ldap_dn2ufnW ldap_dn2ufnW(유니코드) 함수(winldap.h)는 고유 이름을 사용자에게 친숙한 형식으로 변환합니다.
ldap_encode_sort_control ldap_encode_sort_control 함수는 정렬 키 목록의 형식을 검색 컨트롤로 지정합니다. 이 함수는 사용되지 않으며 대신 ldap_create_sort_control 함수를 사용합니다.
ldap_encode_sort_controlA ldap_encode_sort_control 함수는 정렬 키 목록의 형식을 검색 컨트롤로 지정합니다. 이 함수는 사용되지 않습니다. 대신 ldap_create_sort_control 사용합니다. (ldap_encode_sort_controlA)

ldap_encode_sort_controlW ldap_encode_sort_controlW(유니코드) 함수는 정렬 키 목록의 형식을 검색 컨트롤로 지정합니다. 이 함수는 사용되지 않으며 대신 ldap_create_sort_controlW(유니코드) 함수를 사용합니다.
ldap_err2string ldap_err2string 함수(winldap.h)는 숫자 LDAP 오류 코드를 오류를 설명하는 null로 끝나는 문자열로 변환합니다.
ldap_err2stringA 숫자 LDAP 오류 코드를 오류를 설명하는 null로 끝나는 문자열로 변환합니다. (ldap_err2stringA)
ldap_err2stringW ldap_err2stringW(유니코드) 함수(winldap.h)는 숫자 LDAP 오류 코드를 오류를 설명하는 null로 끝나는 문자열로 변환합니다.
ldap_escape_filter_element ldap_escape_filter_element 함수(winldap.h)는 필터 요소를 검색 필터에서 안전하게 전달할 수 있는 null로 끝나는 문자열로 변환합니다.
ldap_escape_filter_elementA ldap_escape_filter_element 함수는 필터 요소를 검색 필터에서 안전하게 전달할 수 있는 null로 끝나는 문자열로 변환합니다. (ldap_escape_filter_elementA)
ldap_escape_filter_elementW ldap_escape_filter_elementW(유니코드) 함수(winldap.h)는 필터 요소를 검색 필터에서 안전하게 전달할 수 있는 null로 끝나는 문자열로 변환합니다.
ldap_explode_dn ldap_explode_dn 함수(winldap.h)는 항목 이름을 구성 요소 부분으로 나눕니다.
ldap_explode_dnA ldap_explode_dn 함수는 항목 이름을 해당 구성 요소 부분으로 구분합니다. (ldap_explode_dnA)
ldap_explode_dnW ldap_explode_dnW(유니코드) 함수(winldap.h)는 항목 이름을 구성 요소 부분으로 나눕니다.

ldap_extended_operation ldap_extended_operation 함수(winldap.h)를 사용하면 확장된 LDAP 작업을 서버에 전달할 수 있습니다.
ldap_extended_operation_sA 확장된 LDAP 작업을 서버에 전달하는 데 사용됩니다. (ANSI)
ldap_extended_operation_sW 확장된 LDAP 작업을 서버에 전달하는 데 사용됩니다. (유니코드)
ldap_extended_operationA ldap_extended_operation 함수를 사용하면 확장된 LDAP 작업을 서버에 전달할 수 있습니다. (ldap_extended_operationA)
ldap_extended_operationW ldap_extended_operationW(유니코드) 함수(winldap.h)를 사용하면 확장된 LDAP 작업을 서버에 전달할 수 있습니다.
ldap_first_attribute ldap_first_attribute 함수(winldap.h)는 첫 번째 특성을 반환합니다.
ldap_first_attributeA 첫 번째 특성을 반환합니다. (ldap_first_attributeA)
ldap_first_attributeW ldap_first_attributeW(유니코드) 함수(winldap.h)는 첫 번째 특성을 반환합니다.
ldap_first_entry ldap_first_entry 함수는 메시지의 첫 번째 항목을 반환합니다.
ldap_first_reference 메시지에서 첫 번째 참조를 반환합니다.
ldap_free_controls ldap_free_controls 함수(winldap.h)는 LDAPControl 구조체의 배열을 해제하는 사용되지 않는 함수입니다.

ldap_free_controlsA LDAPControl 구조체의 배열을 해제하는 사용되지 않는 함수입니다. (ldap_free_controlsA)
ldap_free_controlsW ldap_free_controlsW(유니코드) 함수(winldap.h)는 LDAPControl 구조체의 배열을 해제하는 사용되지 않는 함수입니다.
ldap_get_dn ldap_get_dn 함수(winldap.h)는 지정된 항목의 고유 이름을 검색합니다.
ldap_get_dnA ldap_get_dn 함수는 지정된 항목의 고유 이름을 검색합니다. (ldap_get_dnA)
ldap_get_dnW ldap_get_dnW(유니코드) 함수(winldap.h)는 지정된 항목의 고유 이름을 검색합니다.
ldap_get_next_page 비동기 페이징 검색 결과 시퀀스의 다음 페이지를 반환합니다.
ldap_get_next_page_s 동기 페이징된 검색 결과 시퀀스의 다음 페이지를 반환합니다.
ldap_get_option ldap_get_option 함수(winldap.h)는 세션 전체 매개 변수의 현재 값을 검색합니다.
ldap_get_optionW ldap_get_optionW(유니코드) 함수(winldap.h)는 세션 전체 매개 변수의 현재 값을 검색합니다.
ldap_get_paged_count 서버가 검색을 위해 반환한 페이징된 결과 수를 기록합니다.
ldap_get_values ldap_get_values 함수(winldap.h)는 지정된 특성의 값 목록을 검색합니다.
ldap_get_values_len ldap_get_values_len 함수(winldap.h)는 지정된 특성의 값 목록을 검색합니다.

ldap_get_values_lenA ldap_get_values_len 함수는 지정된 특성의 값 목록을 검색합니다. (ldap_get_values_lenA)
ldap_get_values_lenW ldap_get_values_lenW(유니코드) 함수(winldap.h)는 지정된 특성의 값 목록을 검색합니다.
ldap_get_valuesA ldap_get_values 함수는 지정된 특성의 값 목록을 검색합니다. (ldap_get_valuesA)
ldap_get_valuesW ldap_get_valuesW(유니코드) 함수(winldap.h)는 지정된 특성의 값 목록을 검색합니다.
ldap_init ldap_init 함수(winldap.h)는 LDAP 서버를 사용하여 세션을 초기화합니다.
ldap_initA LDAP 서버를 사용하여 세션을 초기화합니다. (ldap_initA)
ldap_initW ldap_initW(유니코드) 함수(winldap.h)는 LDAP 서버를 사용하여 세션을 초기화합니다.
ldap_memfree ldap_memfree 함수(winldap.h)는 LDAP 힙에서 할당된 메모리를 해제합니다.
ldap_memfreeA LDAP 힙에서 할당된 메모리를 해제합니다. (ldap_memfreeA)
ldap_memfreeW ldap_memfreeW(유니코드) 함수(winldap.h)는 LDAP 힙에서 할당된 메모리를 해제합니다.
ldap_modify ldap_modify 함수(winldap.h)는 비동기 작업을 시작하여 기존 항목을 수정합니다.
ldap_modify_ext ldap_modify_ext 함수는 기존 항목을 변경합니다. (ldap_modify_extW)

ldap_modify_ext_s ldap_modify_ext_s 함수(winldap.h)는 비동기 작업을 시작하여 기존 항목을 수정합니다.
ldap_modify_ext_sA ldap_modify_ext_s 함수는 기존 항목을 변경합니다. (ldap_modify_ext_sA)
ldap_modify_ext_sW ldap_modify_ext_sW(유니코드) 함수(winldap.h)는 기존 항목을 수정하는 비동기 작업을 시작합니다.
ldap_modify_extA ldap_modify_ext 함수는 기존 항목을 변경합니다. (ldap_modify_extA)
ldap_modify_extW ldap_modify_extW(유니코드) 함수(winldap.h)는 기존 항목을 수정하는 비동기 작업을 시작합니다.
ldap_modify_s ldap_modify_s 함수(winldap.h)는 비동기 작업을 시작하여 기존 항목을 수정합니다.
ldap_modify_sA ldap_modify_s 함수는 기존 항목을 변경합니다. (ldap_modify_sA)
ldap_modify_sW ldap_modify_sW(유니코드) 함수(winldap.h)는 비동기 작업을 시작하여 기존 항목을 수정합니다.
ldap_modifyA ldap_modify 함수는 기존 항목을 변경합니다. (ldap_modifyA)
ldap_modifyW ldap_modifyW(유니코드) 함수(winldap.h)는 비동기 작업을 시작하여 기존 항목을 수정합니다.
ldap_modrdn ldap_modrdn 함수(winldap.h)는 LDAP 항목의 상대적인 고유 이름을 변경합니다.
ldap_modrdn_s ldap_modrdn_s 함수(winldap.h)는 LDAP 항목의 상대적인 고유 이름을 변경합니다.

ldap_modrdn_sA LDAP 항목의 상대 고유 이름을 변경합니다. (ldap_modrdn_sA)
ldap_modrdn_sW ldap_modrdn_sW(유니코드) 함수(winldap.h)는 LDAP 항목의 상대적인 고유 이름을 변경합니다.
ldap_modrdn2 ldap_modrdn2 함수(winldap.h)는 LDAP 항목의 상대적인 고유 이름을 변경합니다.
ldap_modrdn2_s ldap_modrdn2_s 함수(winldap.h)는 LDAP 항목의 상대 고유 이름을 변경합니다.
ldap_modrdn2_sA ldap_modrdn2_s 함수는 LDAP 항목의 상대적인 고유 이름을 변경합니다. (ldap_modrdn2_sA)
ldap_modrdn2_sW ldap_modrdn2_sW(유니코드) 함수(winldap.h)는 LDAP 항목의 상대적인 고유 이름을 변경합니다.
ldap_modrdn2A ldap_modrdn2 함수는 LDAP 항목의 상대적인 고유 이름을 변경합니다. (ldap_modrdn2A)
ldap_modrdn2W ldap_modrdn2W(유니코드) 함수(winldap.h)는 LDAP 항목의 상대적인 고유 이름을 변경합니다.
ldap_modrdnA ldap_modrdn 함수는 LDAP 항목의 상대적인 고유 이름을 변경합니다. (ldap_modrdnA)
ldap_modrdnW ldap_modrdnW(유니코드) 함수(winldap.h)는 LDAP 항목의 상대적인 고유 이름을 변경합니다.
ldap_msgfree ldap_msgfree 함수는 이전 호출에서 얻은 결과를 ldap_result 또는 동기 검색 루틴 중 하나로 해제합니다.
ldap_next_attribute ldap_next_attribute 함수(winldap.h)는 다음 특성을 반환합니다.

ldap_next_attributeA 다음 특성을 반환합니다. (ldap_next_attributeA)
ldap_next_attributeW ldap_next_attributeW(유니코드) 함수(winldap.h)는 다음 특성을 반환합니다.
ldap_next_entry ldap_next_entry 함수는 검색 결과 체인에서 항목을 검색합니다.
ldap_next_reference 검색 결과 체인에서 참조를 검색합니다.
ldap_open ldap_open 함수는 연결 블록을 만들고 초기화한 다음 LDAP 서버에 대한 연결을 엽니다. 대신 ldap_init 함수를 사용하지 않는 것이 좋습니다.
ldap_openA ldap_open 사용하지 않는 것이 좋습니다. (ldap_openA)
ldap_openW ldap_openW(유니코드) 함수는 연결 블록을 만들고 초기화한 다음 LDAP 서버에 대한 연결을 엽니다. 대신 ldap_initW(유니코드) 함수를 사용하지 않는 것이 좋습니다.
ldap_parse_extended_resultA ldap_parse_extended_result LDAP 확장 작업의 결과를 구문 분석합니다. (ANSI)
ldap_parse_extended_resultW ldap_parse_extended_result LDAP 확장 작업의 결과를 구문 분석합니다. (유니코드)
ldap_parse_page_control ldap_parse_page_control 함수(winldap.h)는 검색 결과를 페이지로 구문 분석합니다.
ldap_parse_page_controlA ldap_parse_page_control 검색 결과를 페이지로 구문 분석합니다. (ldap_parse_page_controlA)
ldap_parse_page_controlW ldap_parse_page_controlW(유니코드) 함수(winldap.h)는 검색 결과를 페이지로 구문 분석합니다.

ldap_parse_reference ldap_parse_reference 함수(winldap.h)는 검색 응답 메시지에서 하위 조회 목록을 반환합니다.
ldap_parse_referenceA ldap_parse_reference 함수는 검색 응답 메시지에서 하위 조회 목록을 반환합니다. (ldap_parse_referenceA)
ldap_parse_referenceW ldap_parse_referenceW(유니코드) 함수(winldap.h)는 검색 응답 메시지에서 하위 조회 목록을 반환합니다.
ldap_parse_result ldap_parse_result 함수(winldap.h)는 서버의 응답을 구문 분석하고 적절한 필드를 반환합니다.
ldap_parse_resultA ldap_parse_result 함수는 서버의 응답을 구문 분석하고 적절한 필드를 반환합니다. (ldap_parse_resultA)
ldap_parse_resultW ldap_parse_resultW(유니코드) 함수(winldap.h)는 서버의 응답을 구문 분석하고 적절한 필드를 반환합니다.
ldap_parse_sort_control ldap_parse_sort_control 함수(winldap.h)는 서버에서 반환된 정렬 컨트롤을 구문 분석합니다.
ldap_parse_sort_controlA ldap_parse_sort_control 함수는 서버에서 반환된 정렬 컨트롤을 구문 분석합니다. (ldap_parse_sort_controlA)
ldap_parse_sort_controlW ldap_parse_sort_controlW(유니코드) 함수(winldap.h)는 서버에서 반환된 정렬 컨트롤을 구문 분석합니다.
ldap_parse_vlv_controlA VLV 검색 결과를 찾아 구문 분석하는 데 사용됩니다. (ANSI)
ldap_parse_vlv_controlW VLV 검색 결과를 찾아 구문 분석하는 데 사용됩니다. (유니코드)

ldap_perror 사용되지 않는 함수입니다. 호환성을 위해서만 존재합니다.
ldap_rename_ext ldap_rename_ext 함수(winldap.h)는 디렉터리에 있는 항목의 고유 이름을 변경하는 비동기 작업을 시작합니다.
ldap_rename_ext_s ldap_rename_ext_s 함수(winldap.h)는 디렉터리에 있는 항목의 고유 이름을 변경하는 동기 연산입니다.
ldap_rename_ext_sA ldap_rename_ext_s 함수는 디렉터리에 있는 항목의 고유 이름을 변경하는 동기 연산입니다. 이 함수는 LDAP 3에서 사용할 수 있습니다. (ldap_rename_ext_sA)
ldap_rename_ext_sW ldap_rename_ext_sW(유니코드) 함수(winldap.h)는 디렉터리에 있는 항목의 고유 이름을 변경하는 동기 연산입니다.
ldap_rename_extA ldap_rename_ext 함수는 디렉터리에 있는 항목의 고유 이름을 변경하는 비동기 작업을 시작합니다. 이 함수는 LDAP 3에서 사용할 수 있습니다. (ldap_rename_extA)
ldap_rename_extW ldap_rename_extW(유니코드) 함수(winldap.h)는 디렉터리에 있는 항목의 고유 이름을 변경하는 비동기 작업을 시작합니다.
ldap_result 비동기 작업의 결과를 가져옵니다.
ldap_result2error ldap_result2error 함수는 메시지를 구문 분석하고 오류 코드를 반환합니다.
ldap_sasl_bind_sA ldap_sasl_bind_s 함수는 SASL을 사용하여 LDAP 서버에 클라이언트를 인증하는 동기 함수입니다. (ANSI)

ldap_sasl_bind_sW ldap_sasl_bind_s 함수는 SASL을 사용하여 LDAP 서버에 클라이언트를 인증하는 동기 함수입니다. (유니코드)
ldap_sasl_bindA ldap_sasl_bind SASL을 사용하여 클라이언트를 LDAP 서버에 인증하는 비동기 함수입니다. (ANSI)
ldap_sasl_bindW ldap_sasl_bind SASL을 사용하여 클라이언트를 LDAP 서버에 인증하는 비동기 함수입니다. (유니코드)
ldap_search ldap_search 함수(winldap.h)는 LDAP 디렉터리를 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다.
ldap_search_abandon_page ldap_search_abandon_page 함수는 페이징된 결과 검색을 종료합니다.
ldap_search_ext ldap_search_ext 함수(winldap.h)는 LDAP 디렉터리를 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다.
ldap_search_ext_s ldap_search_ext_s 함수(winldap.h)는 LDAP 디렉터리를 동기적으로 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다.
ldap_search_ext_sA 동기적으로 LDAP 디렉터리를 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다. (ldap_search_ext_sA)
ldap_search_ext_sW ldap_search_ext_sW(유니코드) 함수(winldap.h)는 LDAP 디렉터리를 동기적으로 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다.
ldap_search_extA LDAP 디렉터리를 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다. (ldap_search_extA)

ldap_search_extW ldap_search_extW(유니코드) 함수(winldap.h)는 LDAP 디렉터리를 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다.
ldap_search_init_page ldap_search_init_page 함수(winldap.h)는 간단한 페이징 결과 검색에 대한 검색 블록을 초기화합니다.
ldap_search_init_pageA 간단한 페이징 결과 검색에 대한 검색 블록을 초기화합니다. (ldap_search_init_pageA)
ldap_search_init_pageW ldap_search_init_pageW(유니코드) 함수(winldap.h)는 간단한 페이징 결과 검색에 대한 검색 블록을 초기화합니다.
ldap_search_s ldap_search_s 함수(winldap.h)는 LDAP 디렉터리를 동기적으로 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다.
ldap_search_sA ldap_search_s 함수는 LDAP 디렉터리를 동기적으로 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다. (ldap_search_sA)
ldap_search_st ldap_search_st 함수(winldap.h)는 LDAP 디렉터리를 동기적으로 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다.
ldap_search_stA ldap_search_st 함수는 LDAP 디렉터리를 동기적으로 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다. 추가 매개 변수는 검색에 대한 로컬 시간 초과를 지정합니다. (ldap_search_stA)
ldap_search_stW ldap_search_stW(유니코드) 함수(winldap.h)는 LDAP 디렉터리를 동기적으로 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다.
ldap_search_sW ldap_search_sW(유니코드) 함수(winldap.h)는 LDAP 디렉터리를 동기적으로 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다.

ldap_searchA LDAP 디렉터리를 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다. (ldap_searchA)
ldap_searchW ldap_searchW(유니코드) 함수(winldap.h)는 LDAP 디렉터리를 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다.
ldap_set_option ldap_set_option 함수(winldap.h)는 연결 블록에 대한 옵션을 설정합니다.
ldap_set_optionW ldap_set_optionW(유니코드) 함수(winldap.h)는 연결 블록에 대한 옵션을 설정합니다.
ldap_simple_bind ldap_simple_bind 함수(winldap.h)는 일반 텍스트 암호를 사용하여 서버에 대한 클라이언트를 비동기적으로 인증합니다.
ldap_simple_bind_s ldap_simple_bind_s 함수(winldap.h)는 일반 텍스트 암호를 사용하여 서버에 클라이언트를 동기적으로 인증합니다.
ldap_simple_bind_sA ldap_simple_bind_s 함수는 일반 텍스트 암호를 사용하여 서버에 클라이언트를 동기적으로 인증합니다. (ldap_simple_bind_sA)
ldap_simple_bind_sW ldap_simple_bind_sW(유니코드) 함수(winldap.h)는 일반 텍스트 암호를 사용하여 서버에 클라이언트를 동기적으로 인증합니다.
ldap_simple_bindA 일반 텍스트 암호를 사용하여 서버에 대한 클라이언트를 비동기적으로 인증합니다. (ldap_simple_bindA)
ldap_simple_bindW ldap_simple_bindW(유니코드) 함수(winldap.h)는 일반 텍스트 암호를 사용하여 서버에 대한 클라이언트를 비동기적으로 인증합니다.

ldap_sslinit ldap_sslinit 함수(winldap.h)는 LDAP 서버를 사용하여 SSL(Secure Sockets Layer) 세션을 초기화합니다.
ldap_sslinitA LDAP 서버를 사용하여 SSL(Secure Sockets Layer) 세션을 초기화합니다. (ldap_sslinitA)
ldap_sslinitW ldap_sslinitW(유니코드) 함수(winldap.h)는 LDAP 서버를 사용하여 SSL(Secure Sockets Layer) 세션을 초기화합니다.
ldap_start_tls_sA 활성 LDAP 세션에서 TLS 암호화 사용을 시작하는 데 사용됩니다. (ANSI)
ldap_start_tls_sW 활성 LDAP 세션에서 TLS 암호화 사용을 시작하는 데 사용됩니다. (유니코드)
ldap_stop_tls_s ldap_start_tls_s 호출로 시작된 암호화 작업을 중지합니다.
ldap_ufn2dn ldap_ufn2dn 함수(winldap.h)는 사용자에게 친숙한 이름을 고유 이름으로 변환합니다.
ldap_ufn2dnA 사용자에게 친숙한 이름을 고유 이름으로 변환합니다. (ldap_ufn2dnA)
ldap_ufn2dnW ldap_ufn2dnW(유니코드) 함수(winldap.h)는 사용자에게 친숙한 이름을 고유 이름으로 변환합니다.
ldap_unbind ldap_unbind 함수는 LDAP 세션과 연결된 리소스를 해제합니다.
ldap_unbind_s ldap_unbind_s 함수는 LDAP 세션과 연결된 리소스를 동기적으로 해제합니다.
ldap_value_free ldap_value_free 함수(winldap.h)는 ldap_get_values 반환된 구조를 해제합니다.

ldap_value_free_len 이 ldap_value_free_len ldap_get_values_len 반환된 버벌 구조를 해제합니다.
ldap_value_freeA ldap_get_values 반환된 구조를 해제합니다. (ldap_value_freeA)
ldap_value_freeW ldap_value_freeW(유니코드) 함수(winldap.h)는 ldap_get_values 반환된 구조를 해제합니다.
LdapGetLastError LdapGetLastError 함수는 LDAP 호출에서 반환된 마지막 오류 코드를 검색합니다.
LdapMapErrorToWin32 LdapMapErrorToWin32 함수는 LdapError 값을 가장 가까운 Win32 오류 코드로 변환합니다.
LdapUnicodeToUTF8 유니코드 문자열을 UTF-8로 변환합니다.
LdapUTF8ToUnicode UTF-8 코드 페이지가 없는 모듈의 문자열을 번역하는 데 사용됩니다.
QUERYCLIENTCERT SSL(Secure Sockets Layer) 연결을 설정할 때 서버가 클라이언트에서 인증서를 요청할 수 있도록 합니다.
VERIFYSERVERCERT 클라이언트가 연결된 서버의 인증서 체인을 평가할 수 있습니다.

구조체

BerElement BER(기본 인코딩 규칙) 인코딩을 수행하는 C++ 클래스 개체입니다.

LDAP LDAP 세션을 나타냅니다.
LDAP_BERVAL berval 구조는 BER(기본 인코딩 규칙)에 따라 인코딩되는 임의의 이진 데이터를 나타냅니다. 버발을 사용하여 null로 끝나는 문자열로 나타낼 수 없는 특성을 나타냅니다.
LDAP_REFERRAL_CALLBACK 연결의 외부 캐싱을 구현하는 데 사용됩니다.
LDAP_TIMEVAL 시간 간격을 나타내는 데 사용됩니다.
LDAPAPIFeatureInfoA 지원되는 모든 LDAP API 확장에 대한 데이터를 검색합니다. (ANSI)
LDAPAPIFeatureInfoW 지원되는 모든 LDAP API 확장에 대한 데이터를 검색합니다. (유니코드)
LDAPAPIInfoA 사용된 API 및 구현에 대한 데이터를 검색합니다. (ANSI)
LDAPAPIInfoW 사용된 API 및 구현에 대한 데이터를 검색합니다. (유니코드)
LDAPControlA 클라이언트 쪽 컨트롤과 서버 컨트롤을 모두 나타냅니다. (ANSI)
LDAPControlW 클라이언트 쪽 컨트롤과 서버 컨트롤을 모두 나타냅니다. (유니코드)
LDAPMessage LDAP 함수에서 결과 및 오류 데이터를 반환하는 데 사용됩니다.
LDAPModA 수정 작업을 수행하는 데 필요한 데이터를 보유합니다. (ANSI)

LDAPModW 수정 작업을 수행하는 데 필요한 데이터를 보유합니다. (유니코드)
LDAPSortKeyA LDAPSortKey 구조체는 정렬 컨트롤에서 사용할 정렬 조건을 저장합니다. (ANSI)
LDAPSortKeyW LDAPSortKey 구조체는 정렬 컨트롤에서 사용할 정렬 조건을 저장합니다. (유니코드)
LDAPVLVInfo VLV(가상 목록 보기) 요청 컨트롤(LDAP_CONTROL_VLVREQUEST)에 대한 검색 매개 변수를 설정하는 데 사용됩니다.

위치 API

아티클2023. 08. 22.

위치 API 기술 개요입니다.

위치 API를 개발하려면 다음 헤더가 필요합니다.

- locationapi.h
- sensorsapi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 위치 API

열거형

LOCATION_REPORT_STATUS 특정 보고서 유형의 새 보고서에 대해 가능한 상태 정의합니다.

Functions

연결 Windows 8의 Windows 스토어 앱 브라우저에서 앱이 일시 중단(연결 끊기) 및 복원(연결)되었음을 위치 플랫폼에 알리는 데 사용됩니다. (ILocationPower.Connect)
연결 끊기 Windows 8의 Windows 스토어 앱 브라우저에서 앱이 일시 중단(연결 끊기) 및 복원(연결)되었음을 위치 플랫폼에 알리는 데 사용됩니다. (ILocationPower.Disconnect)
GetAddressLine1 주소의 첫 번째 줄을 검색합니다.
GetAddressLine2 주소의 두 번째 줄을 검색합니다.

GetAltitude 고도를 미터 단위로 검색합니다. 고도는 참조 타원을 기준으로 합니다.
GetAltitudeError 고도 오류를 미터 단위로 검색합니다.
GetCity 도시 이름을 검색합니다.
GetCountryRegion 두 글자로 된 국가 또는 지역 코드를 검색합니다.
GetDesiredAccuracy 현재 요청된 정확도 설정을 검색합니다.
GetDetailLevel 예약되어 있습니다. (ICivicAddressReport.GetDetailLevel)
GetErrorRadius 보고된 위치(미터)에서 거리를 검색합니다. 원본으로 보고된 위치와 결합된 이 반지름은 실제 위치가 있는 원을 설명합니다.
GetLatitude 위도(도)를 검색합니다.
GetLongitude 경도를 도로 검색합니다.
GetPostalCode 우편 번호를 검색합니다.
GetReport 기본 위치 공급자에서 지정된 보고서 형식을 검색합니다.
GetReport 위치 보고서를 검색합니다.

GetReportInterval 보고서 이벤트 간에 요청된 시간(밀리초)을 검색합니다.
GetReportStatus 지정된 보고서 형식의 상태 검색합니다.
GetSensorID 위치 보고서를 생성한 센서의 ID를 검색합니다.
GetStateProvince 주 또는 지방 이름을 검색합니다.
GetTimestamp 보고서가 생성된 날짜와 시간을 검색합니다.
GetValue 위치 보고서에서 속성 값을 검색합니다.
OnLocationChanged 새 위치 보고서를 사용할 수 있을 때 호출됩니다.
OnStatusChanged 보고서가 변경되면 호출 상태.
RegisterForReport 위치 보고서 이벤트를 요청합니다.
RequestPermissions 위치 디바이스를 사용하도록 설정하는 사용자 권한을 요청하는 시스템 대화 상자를 엽니다.
SetDesiredAccuracy 사용할 정확도를 지정합니다.
SetReport 기본 위치를 설정합니다.
SetReportInterval

보고서 이벤트 간에 요청된 최소 시간(밀리초)을 지정합니다.
UnregisterForReport 지정된 보고서 유형에 대한 이벤트 알림을 중지합니다.

인터페이스

ICivicAddressReport ICivicAddressReport는 주소 형식의 정보를 포함하는 위치 보고서를 나타냅니다.
IDefaultLocation IDefaultLocation은 기본 위치를 지정하거나 검색하는 데 사용되는 메서드를 제공합니다.
ILatLongReport ILatLongReport는 위도 및 경도 형식의 정보를 포함하는 위치 보고서를 나타냅니다.
ILocation 위치 보고서, 이벤트 등록 및 센서 권한을 관리하는 데 사용되는 메서드를 제공합니다.
ILocationEvents ILocationEvents는 이벤트 알림을 수신하려는 경우 구현해야 하는 콜백 메서드를 제공합니다.
ILocationPower Windows 8의 Windows 스토어 앱 브라우저에서 앱이 일시 중단(연결 끊기) 및 복원(연결)되었음을 위치 플랫폼에 알리는 데 사용됩니다. (ILocationPower)
ILocationReport 위치 보고서의 부모 인터페이스입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

배율 API

아티클2023. 08. 22.

배율 API 기술에 대한 개요입니다.

배율 API를 개발하려면 다음 헤더가 필요합니다.

- magnification.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 배율 API

Functions

MagGetColorEffect 돋보기 컨트롤의 색 변환 행렬을 가져옵니다.
MagGetFullscreenColorEffect 전체 화면 돋보기와 연결된 색 변환 행렬을 검색합니다.
MagGetFullscreenTransform 전체 화면 돋보기에 대한 배율 설정을 검색합니다.
MagGetImageScalingCallback 이미지 크기 조정을 위한 사용자 지정 변환을 구현하는 등록된 콜백 함수를 검색합니다.
MagGetInputTransform 원본 사각형 및 대상 사각형으로 표시되는 펜 및 터치 입력에 대한 현재 입력 변환을 검색합니다.
MagGetWindowFilterList 확대 또는 배율에서 제외된 창 목록을 검색합니다.
MagGetWindowSource 확대되는 영역의 사각형을 가져옵니다.
MagGetWindowTransform

돋보기 컨트롤과 연결된 변환 행렬을 검색합니다.
MagImageScalingCallback 이미지 크기 조정을 위한 사용자 지정 변환을 구현하는 콜백 함수의 프로토타입입니다.
MagInitialize 돋보기 런타임 개체를 만들고 초기화합니다.
MagSetColorEffect 돋보기 컨트롤의 색 변환 매트릭스를 설정합니다.
MagSetFullscreenColorEffect 전체 화면 돋보기와 연결된 색 변환 매트릭스를 변경합니다.
MagSetFullscreenTransform 전체 화면 돋보기에 대한 돋보기 설정을 변경합니다.
MagSetImageScalingCallback 외부 이미지 필터링 및 크기 조정을 위한 콜백 함수를 설정합니다.
MagSetInputTransform 원본 사각형 및 대상 사각형으로 표시되는 펜 및 터치 입력에 대한 현재 활성 입력 변환을 설정합니다.
MagSetWindowFilterList 확대할 창 목록 또는 확대에서 제외할 창 목록을 설정합니다.
MagSetWindowSource 배율 창의 원본 사각형을 설정합니다.
MagSetWindowTransform 돋보기 컨트롤의 변환 매트릭스를 설정합니다.
MagShowSystemCursor 시스템 커서를 표시하거나 숨깁니다.
MagUninitialize 돋보기 런타임 개체를 제거합니다.

구조체

MAGCOLOREFFECT 돋보기 컨트롤이 돋보기 화면 콘텐츠에 색 효과를 적용하는 데 사용하는 색 변환 매트릭스에 대해 설명합니다.
MAGIMAGEHEADER 이미지 형식을 설명합니다.
MAGTRANSFORM 돋보기 컨트롤이 화면 콘텐츠를 확대하는 데 사용하는 변환 매트릭스에 대해 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

관리 정보 기반

아티클2023. 03. 09.

관리 정보 기반 기술에 대한 개요입니다.

관리 정보 베이스를 개발하려면 다음 헤더가 필요합니다.

- ifmib.h
- ipmib.h
- tcpmib.h
- udpmib.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 관리 정보 기반

열거형

ICMP4_TYPE IP 버전 4.0(IPv4) 메시지 유형에 대한 ICMP(Internet Control Message Protocol) 집합을 정의합니다.
ICMP6_TYPE IP 버전 6.0(ICMPv6) 메시지 유형에 대한 인터넷 제어 메시지 프로토콜 집합을 정의합니다.
MIB_NOTIFICATION_TYPE 알림이 발생할 때 콜백 함수에 전달되는 알림 유형을 정의합니다.
TCP_CONNECTION_OFFLOAD_STATE TCP 연결에 사용할 수 있는 TCP 오프로드 상태를 정의합니다.

구조체

MIB_ANYCASTIPADDRESS_ROW 임의의 IP 주소에 대한 정보를 저장합니다.

MIB_ANYCASTIPADDRESS_TABLE 임의의 IP 주소 항목의 테이블을 포함합니다.
MIB_BEST_IF 특정 대상 IPv4 주소에 가장 적합한 경로가 있는 인터페이스의 인덱스가 저장됩니다.
MIB_BOUNDARYROW 멀티캐스트 경계에 대한 IPv4 그룹 주소 값과 마스크를 포함합니다.
MIB_ICMP 특정 컴퓨터에 대한 ICMP(Internet Control Message Protocol) 통계를 포함합니다.
MIB_ICMP_EX_XPSP1 특정 컴퓨터에 대한 확장된 ICMP(Internet Control Message Protocol) 통계를 포함합니다.
MIB_IF_ROW2 특정 인터페이스에 대한 정보를 저장합니다. (MIB_IF_ROW2)
MIB_IF_TABLE2 논리 및 물리적 인터페이스 항목의 테이블을 포함합니다.
MIB_IFNUMBER 특정 컴퓨터의 인터페이스 수를 저장합니다.
MIB_IFROW 특정 인터페이스에 대한 정보를 저장합니다. (MIB_IFROW)
MIB_IFSTACK_ROW 두 네트워크 인터페이스 간의 관계를 나타냅니다. (MIB_IFSTACK_ROW)
MIB_IFSTACK_TABLE 네트워크 인터페이스 스택 행 항목의 테이블을 포함합니다. 인터페이스 스택에서 네트워크 인터페이스의 관계를 지정합니다.
MIB_IFSTATUS 특정 인터페이스에 대한 상태 정보를 저장합니다.

MIB_IFTABLE 인터페이스 항목의 테이블을 포함합니다.
MIB_INVERTEDIFSTACK_ROW 두 네트워크 인터페이스 간의 관계를 나타냅니다. (MIB_INVERTEDIFSTACK_ROW)
MIB_INVERTEDIFSTACK_TABLE 반전된 네트워크 인터페이스 스택 행 항목의 테이블을 포함합니다. 인터페이스 스택의 네트워크 인터페이스 관계를 역순으로 지정합니다.
MIB_IP_NETWORK_CONNECTION_BANDWIDTH_ESTIMATES 네트워크 연결에 대한 TCP/IP 스택에서 계산한 대역폭 예상치에 대한 읽기 전용 정보를 포함합니다.
MIB_IPADDRROW_W2K MIB_IPADDRROW_W2K(ipmib.h)는 MIB_IPADDRTABLE 구조의 특정 IPv4 주소에 대한 정보를 지정합니다.
MIB_IPADDRROW_XP MIB_IPADDRROW_XP(ipmib.h)는 MIB_IPADDRTABLE 구조의 특정 IPv4 주소에 대한 정보를 지정합니다.
MIB_IPADDRTABLE IPv4 주소 항목의 테이블을 포함합니다.
MIB_IPFORWARD_ROW2 IP 경로 항목에 대한 정보를 저장합니다.
MIB_IPFORWARD_TABLE2 IP 경로 항목의 테이블을 포함합니다.
MIB_IPFORWARDNUMBER 특정 IP 라우팅 테이블에 경로 수를 저장합니다.
MIB_IPFORWARDROW IPv4 네트워크 경로를 설명하는 정보를 포함합니다.

MIB_IPFORWARDTABLE IPv4 경로 항목의 테이블을 포함합니다.
MIB_IPINTERFACE_ROW 네트워크 인터페이스에 특정 IP 주소 패밀리에 대한 인터페이스 관리 정보를 저장합니다.
MIB_IPINTERFACE_TABLE IP 인터페이스 항목의 테이블을 포함합니다.
MIB_IPMCAST_BOUNDARY 라우터의 범위가 지정된 IPv4 멀티캐스트 주소 경계를 나열하는 MIB_IPMCAST_BOUNDARY_TABLE 구조의 행을 포함합니다.
MIB_IPMCAST_BOUNDARY_TABLE 라우터의 범위가 지정된 IPv4 멀티캐스트 주소 경계 목록을 포함합니다.
MIB_IPMCAST_GLOBAL 특정 컴퓨터에 IP 멀티캐스트에 대한 전역 정보를 저장합니다.
MIB_IPMCAST_IF_ENTRY IP 멀티캐스트 인터페이스에 대한 정보를 저장합니다.
MIB_IPMCAST_IF_TABLE IP 멀티캐스트 인터페이스 항목의 테이블을 포함합니다.
MIB_IPMCAST_MFE IP(인터넷 프로토콜) MFE(멀티캐스트 전달 항목)에 대한 정보를 저장합니다.
MIB_IPMCAST_MFE_STATS MFE(멀티캐스트 전달 항목)와 연결된 통계를 저장합니다.
MIB_IPMCAST_MFE_STATS_EX_XP MFE(멀티캐스트 전달 항목)와 연결된 확장된 통계를 저장합니다.
MIB_IPMCAST_OIF_STATS_LH MIB_IPMCAST_OIF_STATS_LH(ipmib.h)는 나가는 멀티캐스트 인터페이스와 연결된 통계를 저장합니다.

MIB_IPMCAST_OIF_STATS_W2K MIB_IPMCAST_OIF_STATS_W2K(ipmib.h)는 나가는 멀티캐스트 인터페이스와 연결된 통계를 저장합니다.
MIB_IPMCAST_OIF_W2K MIB_IPMCAST_OIF_W2K(ipmib.h)는 나가는 IP 멀티캐스트 패킷을 보내는 데 필요한 정보를 저장합니다.
MIB_IPMCAST_OIF_XP MIB_IPMCAST_OIF_XP(ipmib.h)는 나가는 IP 멀티캐스트 패킷을 보내는 데 필요한 정보를 저장합니다.
MIB_IPMCAST_SCOPE 멀티캐스트 범위 이름과 범위를 정의하는 연결된 IPv4 멀티캐스트 그룹 주소 및 마스크를 포함합니다.
MIB_IPNET_ROW2 인접 IP 주소에 대한 정보를 저장합니다.
MIB_IPNET_TABLE2 인접 IP 주소 항목의 테이블을 포함합니다.
MIB_IPNETROW_LH MIB_IPNETROW_LH(ipmib.h)에는 IPv4 주소에 대한 ARP(주소 확인 프로토콜) 테이블 항목에 대한 정보가 포함되어 있습니다.
MIB_IPNETROW_W2K MIB_IPNETROW_W2K(ipmib.h)에는 IPv4 주소에 대한 ARP(주소 확인 프로토콜) 테이블 항목에 대한 정보가 포함되어 있습니다.
MIB_IPNETTABLE IPv4 주소에 대한 ARP(주소 확인 프로토콜) 항목의 테이블을 포함합니다.
MIB_IPPATH_ROW IP 경로 항목에 대한 정보를 저장합니다.
MIB_IPPATH_TABLE IP 경로 항목의 테이블을 포함합니다.

MIB_IPSTATS_LH MIB_IPSTATS_LH(ipmib.h)는 특정 컴퓨터에서 실행되는 IP 프로토콜에 대한 정보를 저장합니다.
MIB_IPSTATS_W2K MIB_IPSTATS_W2K(ipmib.h)는 특정 컴퓨터에서 실행되는 IP 프로토콜에 대한 정보를 저장합니다.
MIB_MCAST_LIMIT_ROW MIB_MCAST_LIMIT_ROW 구조체에는 해당 MIB_IPMCAST_IF_ENTRY 구조체의 구성 가능한 제한 정보가 포함됩니다.
MIB_MFE_STATS_TABLE MFE(멀티캐스트 전달 항목) 그룹에 대한 통계를 저장합니다.
MIB_MFE_STATS_TABLE_EX_XP MFE(멀티캐스트 전달 항목)에 대한 확장 통계 테이블을 포함합니다.
MIB_MFE_TABLE MFE(멀티캐스트 전달 항목)의 테이블을 포함합니다.
MIB_MULTICASTIPADDRESS_ROW 멀티캐스트 IP 주소에 대한 정보를 저장합니다.
MIB_MULTICASTIPADDRESS_TABLE 멀티캐스트 IP 주소 항목의 테이블을 포함합니다.
MIB_OPAQUE_INFO MIB 불투명 쿼리에서 반환된 정보를 포함합니다.
MIB_OPAQUE_QUERY MIB 불투명 쿼리에 대한 정보를 포함합니다.
MIB_PROXYARP PARP(프록시 주소 확인 프로토콜) 항목에 대한 정보를 저장합니다.
MIB_TCP6ROW IPv6 TCP 연결을 설명하는 정보를 포함합니다. (MIB_TCP6ROW)

MIB_TCP6ROW_OWNER_MODULE 소유권 데이터가 있는 특정 프로세스 ID(PID)에 바인딩된 IPv6 TCP 연결을 설명하는 정보를 포함합니다.
MIB_TCP6ROW_OWNER_PID 특정 프로세스 ID(PID)와 연결된 IPv6 TCP 연결을 설명하는 정보를 포함합니다.
MIB_TCP6ROW2 IPv6 TCP 연결을 설명하는 정보를 포함합니다. (MIB_TCP6ROW2)
MIB_TCP6TABLE 로컬 컴퓨터의 IPv6에 대한 TCP 연결 테이블을 포함합니다.
MIB_TCP6TABLE_OWNER_MODULE 사용 가능한 모든 소유권 데이터와 함께 이러한 PID에 바인딩된 프로세스 ID(PID) 및 IPv6 TCP 링크 컨텍스트의 테이블을 포함합니다.
MIB_TCP6TABLE_OWNER_PID 이러한 PID에 바인딩된 컨텍스트인 프로세스 ID(ID) 및 IPv6 TCP 링크의 테이블이 포함되어 있습니다.
MIB_TCP6TABLE2 로컬 컴퓨터의 IPv6 TCP 연결 테이블을 포함합니다.
MIB_TCPROW_LH MIB_TCPROW_LH(tcpmib.h)에는 IPv4 TCP 연결을 설명하는 정보가 포함되어 있습니다.
MIB_TCPROW_OWNER_MODULE 소유권 데이터, IPv4 주소, TCP 연결에서 사용되는 포트 및 연결과 연결된 특정 프로세스 ID(PID)를 사용하여 IPv4 TCP 연결을 설명하는 정보를 포함합니다.
MIB_TCPROW_OWNER_PID IPv4 주소와 IPv4 TCP 연결, TCP 연결에서 사용되는 포트 및 연결과 연결된 특정 프로세스 ID(PID)를 설명하는 정보를 포함합니다.
MIB_TCPROW_W2K MIB_TCPROW_W2K(tcpmib.h)에는 IPv4 TCP 연결을 설명하는 정보가 포함되어 있습니다.

MIB_TCPROW2 IPv4 TCP 연결을 설명하는 정보를 포함합니다.
MIB_TCPSTATS_LH MIB_TCPSTATS_LH(tcpmib.h)에는 로컬 컴퓨터에서 실행되는 TCP 프로토콜에 대한 통계가 포함되어 있습니다.
MIB_TCPSTATS_W2K MIB_TCPSTATS_W2K(tcpmib.h)에는 로컬 컴퓨터에서 실행되는 TCP 프로토콜에 대한 통계가 포함되어 있습니다.
MIB_TCPSTATS2 로컬 컴퓨터에서 실행되는 TCP 프로토콜에 대한 통계를 포함합니다. (MIB_TCPSTATS2)
MIB_TCPTABLE 로컬 컴퓨터의 IPv4에 대한 TCP 연결 테이블을 포함합니다.
MIB_TCPTABLE_OWNER_MODULE PID(프로세스 ID) 테이블과 PID에 바인딩된 IPv4 TCP 링크 컨텍스트 및 사용 가능한 모든 소유권 데이터를 포함합니다.
MIB_TCPTABLE_OWNER_PID 이러한 PID에 바인딩된 컨텍스트인 프로세스 ID(PID) 및 IPv4 TCP 링크의 테이블이 포함되어 있습니다.
MIB_TCPTABLE2 로컬 컴퓨터의 IPv4 TCP 연결 테이블을 포함합니다.
MIB_UDP6ROW 로컬 컴퓨터의 IPv6에 대한 UDP(사용자 데이터그램 프로토콜) 수신기 테이블의 항목을 포함합니다.
MIB_UDP6ROW_OWNER_MODULE 로컬 컴퓨터의 IPv6에 대한 UDP(사용자 데이터그램 프로토콜) 수신기 테이블의 항목을 포함합니다. 또한 이 항목에는 사용 가능한 모든 소유권 데이터와 UDP 엔드포인트에 대한 바인딩 함수 호출을 실행한 PID(프로세스 ID)도 포함됩니다.

MIB_UDP6ROW_OWNER_PID 로컬 컴퓨터의 IPv6에 대한 UDP(사용자 데이터그램 프로토콜) 수신기 테이블의 항목을 포함합니다. 항목에는 UDP 엔드포인트에 대한 바인딩 함수에 대한 호출을 실행한 PID(프로세스 ID)도 포함됩니다.
MIB_UDP6TABLE 로컬 컴퓨터의 IPv6에 대한 UDP(사용자 데이터그램 프로토콜) 수신기 테이블을 포함합니다.
MIB_UDP6TABLE_OWNER_MODULE 로컬 컴퓨터의 IPv6에 대한 UDP(사용자 데이터그램 프로토콜) 수신기 테이블을 포함합니다. 또한 테이블에는 사용 가능한 모든 소유권 데이터와 각 UDP 엔드포인트에 대한 바인딩 함수 호출을 실행한 PID(프로세스 ID)가 포함됩니다.
MIB_UDP6TABLE_OWNER_PID 로컬 컴퓨터의 IPv6에 대한 UDP(사용자 데이터그램 프로토콜) 수신기 테이블을 포함합니다. 테이블에는 각 UDP 엔드포인트에 대한 바인딩 함수 호출을 실행한 PID(프로세스 ID)도 포함되어 있습니다.
MIB_UDPROW 로컬 컴퓨터의 IPv4용 UDP(사용자 데이터그램 프로토콜) 수신기 테이블의 항목을 포함합니다.
MIB_UDPROW_OWNER_MODULE 로컬 컴퓨터의 IPv4 UDP(사용자 데이터그램 프로토콜) 수신기 테이블의 항목을 포함합니다. 또한 이 항목에는 사용 가능한 모든 소유권 데이터와 UDP 엔드포인트에 대한 바인딩 함수 호출을 실행한 PID(프로세스 ID)도 포함됩니다.
MIB_UDPROW_OWNER_PID 로컬 컴퓨터의 IPv4용 UDP(사용자 데이터그램 프로토콜) 수신기 테이블의 항목을 포함합니다. 항목에는 UDP 엔드포인트에 대한 바인딩 함수 호출을 실행한 PID(프로세스 ID)도 포함됩니다.
MIB_UDPSTATS 로컬 컴퓨터에서 실행되는 UDP(사용자 데이터그램 프로토콜)에 대한 통계를 포함합니다. (MIB_UDPSTATS)
MIB_UDPSTATS2 로컬 컴퓨터에서 실행되는 UDP(사용자 데이터그램 프로토콜)에 대한 통계를 포함합니다. (MIB_UDPSTATS2)

MIB_UDPTABLE 로컬 컴퓨터의 IPv4용 UDP(사용자 데이터그램 프로토콜) 수신기 테이블을 포함합니다.
MIB_UDPTABLE_OWNER_MODULE 로컬 컴퓨터의 IPv4용 UDP(사용자 데이터그램 프로토콜) 수신기 테이블을 포함합니다. 또한 테이블에는 사용 가능한 모든 소유권 데이터와 각 UDP 엔드포인트에 대한 바인딩 함수 호출을 실행한 PID(프로세스 ID)가 포함되어 있습니다.
MIB_UDPTABLE_OWNER_PID 로컬 컴퓨터의 IPv4용 UDP(사용자 데이터그램 프로토콜) 수신기 테이블을 포함합니다. 테이블에는 각 UDP 엔드포인트에 대한 바인딩 함수 호출을 실행한 PID(프로세스 ID)도 포함되어 있습니다.
MIB_UNICASTIPADDRESS_ROW 유니캐스트 IP 주소에 대한 정보를 저장합니다.
MIB_UNICASTIPADDRESS_TABLE 유니캐스트 IP 주소 항목의 테이블을 포함합니다.
MIBICMPINFO 특정 컴퓨터에 대한 ICMP(인터넷 제어 메시지 프로토콜) 통계를 포함합니다.
MIBICMPSTATS 특정 컴퓨터에서 들어오거나 나가는 ICMP(인터넷 제어 메시지 프로토콜) 메시지에 대한 통계를 포함합니다.
MIBICMPSTATS_EX_XPSP1 특정 컴퓨터에서 들어오거나 나가는 ICMP(Internet Control Message Protocol) 메시지에 대한 확장된 통계를 포함합니다.
NL_INTERFACE_OFFLOAD_ROD IP 인터페이스에 대한 오프로드 기능을 나타내는 플래그 집합을 지정합니다.

메뉴 및 기타 리소스

아티클2023. 03. 16.

메뉴 및 기타 리소스 기술 개요입니다.

메뉴 및 기타 리소스를 개발하려면 다음 헤더가 필요합니다.

- resourceindexer.h
- strsafe.h
- verrsrc.h
- winver.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 메뉴 및 기타 리소스

Functions

AppendMenuA 지정된 메뉴 모음, 드롭다운 메뉴, 하위 메뉴 또는 바로 가기 메뉴의 끝에 새 항목을 추가합니다. 이 함수를 사용하여 메뉴 항목의 콘텐츠, 모양 및 동작을 지정할 수 있습니다. (ANSI)
AppendMenuW 지정된 메뉴 모음, 드롭다운 메뉴, 하위 메뉴 또는 바로 가기 메뉴의 끝에 새 항목을 추가합니다. 이 함수를 사용하여 메뉴 항목의 콘텐츠, 모양 및 동작을 지정할 수 있습니다. (유니코드)
BeginUpdateResourceA UpdateResource 함수에서 이진 모듈의 리소스를 추가, 삭제 또는 바꾸는 데 사용할 수 있는 핸들을 검색합니다. (ANSI)
BeginUpdateResourceW UpdateResource 함수에서 이진 모듈의 리소스를 추가, 삭제 또는 바꾸는 데 사용할 수 있는 핸들을 검색합니다. (유니코드)
CharLowerA 문자열 또는 단일 문자를 소문자로 변환합니다. 피연산자가 문자열인 경우 함수는 현재 위치에 있는 문자를 변환합니다. (ANSI)

CharLowerBuffA 버퍼의 대문자를 소문자로 변환합니다. 함수는 현재 위치에 있는 문자를 변환합니다. (ANSI)
CharLowerBuffW 버퍼의 대문자를 소문자로 변환합니다. 함수는 현재 위치에 있는 문자를 변환합니다. (유니코드)
CharLowerW 문자열 또는 단일 문자를 소문자로 변환합니다. 피연산자가 문자열인 경우 함수는 현재 위치에 있는 문자를 변환합니다. (유니코드)
CharNextA 문자열의 다음 문자에 대한 포인터를 검색합니다. 이 함수는 단일 또는 다중 바이트 문자로 구성된 문자열을 처리할 수 있습니다. (ANSI)
CharNextExA 문자열의 다음 문자에 대한 포인터를 검색합니다. 이 함수는 단일 또는 다중 바이트 문자로 구성된 문자열을 처리할 수 있습니다.
CharNextW 문자열의 다음 문자에 대한 포인터를 검색합니다. 이 함수는 단일 또는 다중 바이트 문자로 구성된 문자열을 처리할 수 있습니다. (유니코드)
CharPrevA 문자열에서 이전 문자에 대한 포인터를 검색합니다. 이 함수는 단일 또는 다중 바이트 문자로 구성된 문자열을 처리할 수 있습니다. (ANSI)
CharPrevExA 문자열에서 이전 문자에 대한 포인터를 검색합니다. 이 함수는 단일 또는 다중 바이트 문자로 구성된 문자열을 처리할 수 있습니다.
CharPrevW 문자열에서 이전 문자에 대한 포인터를 검색합니다. 이 함수는 단일 또는 다중 바이트 문자로 구성된 문자열을 처리할 수 있습니다. (유니코드)
CharToOemA 문자열을 OEM 정의 문자 집합으로 변환합니다. 경고 사용하지 마세요. (ANSI)

CharToOemBuffA 문자열에서 지정된 수의 문자를 OEM 정의 문자 집합으로 변환합니다. (ANSI)
CharToOemBuffW 문자열에서 지정된 수의 문자를 OEM 정의 문자 집합으로 변환합니다. (유니코드)
CharToOemW 문자열을 OEM 정의 문자 집합으로 변환합니다. 경고 사용하지 마세요. (유니코드)
CharUpperA 문자열 또는 단일 문자를 대문자로 변환합니다. 피연산자가 문자열인 경우 함수는 현재 위치에 있는 문자를 변환합니다. (ANSI)
CharUpperBuffA 버퍼의 소문자를 대문자로 변환합니다. 함수는 현재 위치에 있는 문자를 변환합니다. (ANSI)
CharUpperBuffW 버퍼의 소문자를 대문자로 변환합니다. 함수는 현재 위치에 있는 문자를 변환합니다. (유니코드)
CharUpperW 문자열 또는 단일 문자를 대문자로 변환합니다. 피연산자가 문자열인 경우 함수는 현재 위치에 있는 문자를 변환합니다. (유니코드)
CheckMenuItem 지정된 메뉴 항목의 검사 표시 특성의 상태를 선택 또는 지우기로 설정합니다.
CheckMenuRadioItem 지정된 메뉴 항목을 확인하고 라디오 항목으로 만듭니다. 동시에 함수는 연결된 그룹의 다른 모든 메뉴 항목을 지우고 해당 항목에 대한 라디오 항목 형식 플래그를 지웁니다.
ClipCursor 커서를 화면의 사각형 영역으로 제한합니다.
CopyAcceleratorTableA 지정된 가속기 테이블을 복사합니다. 이 함수는 accelerator-table 핸들에 해당하는 accelerator-table 데이터를 가져오거나 accelerator-table 데이터의 크기를 결정하는 데 사용됩니다. (ANSI)

CopyAcceleratorTableW 지정된 가속기 테이블을 복사합니다. 이 함수는 accelerator-table 핸들에 해당하는 accelerator-table 데이터를 가져오거나 accelerator-table 데이터의 크기를 결정하는 데 사용됩니다. (유니코드)
CopyCursor 지정된 커서를 복사합니다.
CopyIcon 지정된 아이콘을 다른 모듈에서 현재 모듈로 복사합니다.
CopyImage 새 이미지(아이콘, 커서 또는 비트맵)를 만들고 지정된 이미지의 특성을 새 이미지에 복사합니다. 필요한 경우 함수는 새 이미지의 원하는 크기에 맞게 비트를 늘립니다.
CreateAcceleratorTableA 가속기 테이블을 만듭니다. (ANSI)
CreateAcceleratorTableW 가속기 테이블을 만듭니다. (유니코드)
CreateCaret 시스템 caret에 대한 새 셰이프를 만들고 지정된 창에 caret의 소유권을 할당합니다. 캐럿 셰이프는 선, 블록 또는 비트맵일 수 있습니다.
CreateCursor 지정된 크기, 비트 패턴 및 핫 스폿이 있는 커서를 만듭니다.
CreateIcon 지정된 크기, 색 및 비트 패턴이 있는 아이콘을 만듭니다.
CreateIconFromResource 아이콘을 설명하는 리소스 비트에서 아이콘 또는 커서를 만듭니다. (CreateIconFromResource)
CreateIconFromResourceEx 아이콘을 설명하는 리소스 비트에서 아이콘 또는 커서를 만듭니다. (CreateIconFromResourceEx)

CreateIconIndirect ICONINFO 구조체에서 아이콘 또는 커서를 만듭니다.
CreateMenu 메뉴를 만듭니다. 메뉴는 처음에는 비어 있지만 InsertMenuItem, AppendMenu 및 InsertMenu 함수를 사용하여 메뉴 항목으로 채울 수 있습니다.
CreatePopupMenu 드롭다운 메뉴, 하위 메뉴 또는 바로 가기 메뉴를 만듭니다.
CreateResourceIndexer 프로젝트 파일의 루트 및 확장 DLL의 지정된 경로에 대한 새 리소스 인덱서를 만듭니다.
DeleteMenu 지정된 메뉴에서 항목을 삭제합니다. 메뉴 항목이 메뉴 또는 하위 메뉴를 열면 이 함수는 메뉴 또는 하위 메뉴에 대한 핸들을 삭제하고 메뉴 또는 하위 메뉴에서 사용되는 메모리를 해제합니다.
DestroyAcceleratorTable 가속기 테이블을 삭제합니다.
DestroyCaret caret의 현재 모양을 삭제하고, 창에서 캐리트를 해제하고, 화면에서 캐리트를 제거합니다.
DestroyCursor 커서를 삭제하고 커서가 점유한 메모리를 해제합니다. 이 함수를 사용하여 공유 커서를 삭제하지 마세요.
DestroyIcon 아이콘을 삭제하고 아이콘이 차지하는 메모리를 해제합니다.
DestroyIndexedResults IndexFilePath 메서드가 반환한 매개 변수를 해제합니다.
DestroyMenu 지정된 메뉴를 삭제하고 메뉴가 차지하는 모든 메모리를 해제합니다.

DestroyResourceIndexer 지정된 리소스 인덱서와 연결된 계산 리소스를 해제합니다.
DrawIcon 지정된 디바이스 컨텍스트에 아이콘 또는 커서를 그립니다.
DrawIconEx 지정된 디바이스 컨텍스트에 아이콘 또는 커서를 그리고 지정된 래스터 작업을 수행하고 지정된 대로 아이콘 또는 커서를 확장하거나 압축합니다.
DrawMenuBar 지정된 창의 메뉴 모음을 다시 그립니다. 시스템에서 창을 만든 후 메뉴 모음이 변경되면 변경된 메뉴 모음을 그리려면 이 함수를 호출해야 합니다.
EnableMenuItem 지정된 메뉴 항목을 사용하거나 사용하지 않도록 설정하거나 회색으로 표시합니다.
EndMenu 호출 스레드의 활성 메뉴를 종료합니다.
EndUpdateResourceA UpdateResource를 호출하기 전에 변경한 내용을 커밋하거나 삭제합니다. (ANSI)
EndUpdateResourceW UpdateResource를 호출하기 전에 변경한 내용을 커밋하거나 삭제합니다. (유니코드)
ENUMRESLANGPROCA EnumResourceLanguages 및 EnumResourceLanguagesEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (ANSI)
ENUMRESLANGPROCW EnumResourceLanguages 및 EnumResourceLanguagesEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (ANSI)
ENUMRESNAMEPROCA EnumResourceNames 및 EnumResourceNamesEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (ANSI)

ENUMRESNAMEPROCW EnumResourceNames 및 EnumResourceNamesEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (유니코드)
EnumResourceLanguagesA 이진 모듈과 연결된 지정된 형식 및 이름의 언어별 리소스를 열거합니다. (ANSI)
EnumResourceLanguagesExA 지정된 이진 모듈과 연결된 지정된 형식 및 이름의 언어별 리소스를 열거합니다. 열거형에 대한 제어를 더 많이 허용하여 EnumResourceLanguages를 확장합니다. (ANSI)
EnumResourceLanguagesExW 지정된 이진 모듈과 연결된 지정된 형식 및 이름의 언어별 리소스를 열거합니다. 열거형에 대한 제어를 더 많이 허용하여 EnumResourceLanguages를 확장합니다. (유니코드)
EnumResourceLanguagesW 이진 모듈과 연결된 지정된 형식 및 이름의 언어별 리소스를 열거합니다. (유니코드)
EnumResourceNamesA 이진 모듈 내에서 지정된 형식의 리소스를 열거합니다. (ANSI)
EnumResourceNamesExA 지정된 이진 모듈과 연결된 지정된 형식의 리소스를 열거합니다. 검색에는 LN 파일과 연결된 .mui 파일이 모두 포함되거나 여러 가지 방법으로 제한될 수 있습니다. (ANSI)
EnumResourceNamesExW 지정된 이진 모듈과 연결된 지정된 형식의 리소스를 열거합니다. 검색에는 LN 파일과 연결된 .mui 파일이 모두 포함되거나 여러 가지 방법으로 제한될 수 있습니다. (유니코드)
EnumResourceNamesW 이진 모듈 내에서 지정된 형식의 리소스를 열거합니다. (유니코드)
EnumResourceTypesA 이진 모듈 내에서 리소스 종류를 열거합니다. (ANSI)
EnumResourceTypesExA 지정된 이진 모듈과 연결된 리소스 종류를 열거합니다. (ANSI)

EnumResourceTypesExW 지정된 이진 모듈과 연결된 리소스 종류를 열거합니다. (유니코드)
EnumResourceTypesW 이진 모듈 내에서 리소스 종류를 열거합니다. (유니코드)
ENUMRESTYPEPROCA EnumResourceTypes 및 EnumResourceTypesEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (ANSI)
ENUMRESTYPEPROCW EnumResourceTypes 및 EnumResourceTypesEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (유니코드)
FindResourceA 지정된 모듈에서 지정된 형식과 이름을 가진 리소스의 위치를 결정합니다. (FindResourceA)
FindResourceExA 지정된 모듈에서 지정된 형식, 이름 및 언어를 사용하여 리소스의 위치를 결정합니다. (FindResourceExA)
FindResourceExW 지정된 모듈에서 지정된 형식, 이름 및 언어를 사용하여 리소스의 위치를 결정합니다. (FindResourceExW)
FindResourceW 지정된 모듈에서 지정된 형식과 이름을 가진 리소스의 위치를 결정합니다. (FindResourceW)
FreeResource 로드된 리소스의 참조 수를 1씩 줄입니다. 참조 수가 0에 도달하면 리소스가 차지하는 메모리가 해제됩니다.
GetCaretBlinkTime caret의 픽셀을 반전하는 데 필요한 시간을 검색합니다. 사용자는 이 값을 설정할 수 있습니다.
GetCaretPos 지정된 POINT 구조체에 caret의 위치를 복사합니다.

GetClipCursor 커서가 제한된 사각형 영역의 화면 좌표를 검색합니다.
GetCursor 현재 커서에 대한 핸들을 검색합니다.
GetCursorInfo 전역 커서에 대한 정보를 검색합니다.
GetCursorPos 화면 좌표에서 마우스 커서의 위치를 검색합니다.
GetDistanceOfClosestLanguageInList 지정된 언어 코드와 언어 목록에서 가장 가까운 일치 항목 사이의 거리를 결정합니다.
GetFileVersionInfoA 지정된 파일에 대한 버전 정보를 검색합니다. (GetFileVersionInfoA)
GetFileVersionInfoExA 지정된 파일에 대한 버전 정보를 검색합니다. (GetFileVersionInfoExA)
GetFileVersionInfoExW 지정된 파일에 대한 버전 정보를 검색합니다. (GetFileVersionInfoExW)
GetFileVersionInfoSizeA 운영 체제에서 지정된 파일에 대한 버전 정보를 검색할 수 있는지 여부를 결정합니다. 버전 정보를 사용할 수 있는 경우 GetFileVersionInfoSize는 해당 정보의 크기를 바이트 단위로 반환합니다. (ANSI)
GetFileVersionInfoSizeExA 운영 체제에서 지정된 파일에 대한 버전 정보를 검색할 수 있는지 여부를 결정합니다. 버전 정보를 사용할 수 있는 경우 GetFileVersionInfoSizeEx는 해당 정보의 크기를 바이트 단위로 반환합니다. (ANSI)
GetFileVersionInfoSizeExW 운영 체제에서 지정된 파일에 대한 버전 정보를 검색할 수 있는지 여부를 결정합니다. 버전 정보를 사용할 수 있는 경우 GetFileVersionInfoSizeEx는 해당 정보의 크기를 바이트 단위로 반환합니다. (유니코드)

GetFileVersionInfoSizeW 운영 체제에서 지정된 파일에 대한 버전 정보를 검색할 수 있는지 여부를 결정합니다. 버전 정보를 사용할 수 있는 경우 GetFileVersionInfoSize는 해당 정보의 크기를 바이트 단위로 반환합니다. (유니코드)
GetFileVersionInfoW 지정된 파일에 대한 버전 정보를 검색합니다. (GetFileVersionInfoW)
GetIconInfo 지정된 아이콘 또는 커서에 대한 정보를 검색합니다.
GetIconInfoExA 지정된 아이콘 또는 커서에 대한 정보를 검색합니다. GetIconInfoEx는 최신 ICONINFOEX 구조를 사용하여 GetIconInfo를 확장합니다. (ANSI)
GetIconInfoExW 지정된 아이콘 또는 커서에 대한 정보를 검색합니다. GetIconInfoEx는 최신 ICONINFOEX 구조를 사용하여 GetIconInfo를 확장합니다. (유니코드)
GetMenu 지정된 창에 할당된 메뉴에 대한 핸들을 검색합니다.
GetMenuBarInfo 지정된 메뉴 모음에 대한 정보를 검색합니다.
GetMenuCheckMarkDimensions 기본 검사 표시 비트맵의 차원을 검색합니다.
GetMenuDefaultItem 지정된 메뉴에서 기본 메뉴 항목을 결정합니다.
GetMenuInfo 지정된 메뉴에 대한 정보를 검색합니다.
GetMenuItemCount 지정된 메뉴의 항목 수를 결정합니다.

GetMenuItemID 메뉴의 지정된 위치에 있는 메뉴 항목의 메뉴 항목 식별자를 검색합니다.
GetMenuItemInfoA 메뉴 항목에 대한 정보를 검색합니다. (ANSI)
GetMenuItemInfoW 메뉴 항목에 대한 정보를 검색합니다. (유니코드)
GetMenuItemRect 지정된 메뉴 항목에 대한 경계 사각형을 검색합니다.
GetMenuState 지정된 메뉴 항목과 연결된 메뉴 플래그를 검색합니다.
GetMenuStringA 지정된 메뉴 항목의 텍스트 문자열을 지정된 버퍼에 복사합니다. (ANSI)
GetMenuStringW 지정된 메뉴 항목의 텍스트 문자열을 지정된 버퍼에 복사합니다. (유니코드)
GetPhysicalCursorPos 실제 좌표에서 커서의 위치를 검색합니다.
GetSubMenu 지정된 메뉴 항목에 의해 활성화된 드롭다운 메뉴 또는 하위 메뉴에 대한 핸들을 검색합니다.
GetSystemMenu 애플리케이션이 복사 및 수정을 위해 창 메뉴(시스템 메뉴 또는 컨트롤 메뉴라고도 함)에 액세스할 수 있도록 합니다.
HideCaret 화면에서 케어를 제거합니다. caret를 숨기면 현재 셰이프가 삭제되거나 삽입 지점이 무효화되지 않습니다.
HiliteMenuItem 메뉴 모음의 항목에서 강조 표시를 추가하거나 제거합니다.

IndexFilePath 파일 및 폴더 명명 규칙에 대한 파일 경로를 인덱싱합니다.
InsertMenuA 메뉴에 새 메뉴 항목을 삽입하고 다른 항목을 메뉴 아래로 이동합니다. (ANSI)
InsertMenuItemA 메뉴의 지정된 위치에 새 메뉴 항목을 삽입합니다. (ANSI)
InsertMenuItemW 메뉴의 지정된 위치에 새 메뉴 항목을 삽입합니다. (유니코드)
InsertMenuW 메뉴에 새 메뉴 항목을 삽입하고 다른 항목을 메뉴 아래로 이동합니다. (유니코드)
IS_INTRESOURCE 값이 리소스의 정수 식별자인지 여부를 확인합니다.
IsCharAlphaA 문자가 사전순 문자인지 여부를 결정합니다. 이 결정은 설치 중 또는 제어판 통해 사용자가 선택한 언어의 의미 체계를 기반으로 합니다. (ANSI)
IsCharAlphaNumericA 문자가 사전순인지 숫자 문자인지 여부를 결정합니다. 이 결정은 설치 중 또는 제어판 통해 사용자가 선택한 언어의 의미 체계를 기반으로 합니다. (ANSI)
IsCharAlphaNumericW 문자가 사전순인지 숫자 문자인지 여부를 결정합니다. 이 결정은 설치 중 또는 제어판 통해 사용자가 선택한 언어의 의미 체계를 기반으로 합니다. (유니코드)
IsCharAlphaW 문자가 사전순 문자인지 여부를 결정합니다. 이 결정은 설치 중 또는 제어판 통해 사용자가 선택한 언어의 의미 체계를 기반으로 합니다. (유니코드)
IsCharLowerA 문자가 소문자인지 여부를 확인합니다. 이 결정은 설치 중 또는 제어판 통해 사용자가 선택한 언어의 의미 체계를 기반으로 합니다.

IsCharUpperA 문자가 대문자인지 여부를 결정합니다. 이 결정은 설치 중 또는 제어판 통해 사용자가 선택한 언어의 의미 체계를 기반으로 합니다. (ANSI)
IsCharUpperW 문자가 대문자인지 여부를 결정합니다. 이 결정은 설치 중 또는 제어판 통해 사용자가 선택한 언어의 의미 체계를 기반으로 합니다. (유니코드)
IsMenu 핸들이 메뉴 핸들인지 여부를 확인합니다.
IsWellFormedTag BCP-47 언어 태그가 올바른 형식인지 여부를 결정합니다.
LoadAcceleratorsA 지정된 가속기 테이블을 로드합니다. (ANSI)
LoadAcceleratorsW 지정된 가속기 테이블을 로드합니다. (유니코드)
LoadCursorA 애플리케이션 instance 연결된 실행 파일(.EXE) 파일에서 지정된 커서 리소스를 로드합니다. (ANSI)
LoadCursorFromFileA 파일에 포함된 데이터를 기반으로 커서를 만듭니다. (ANSI)
LoadCursorFromFileW 파일에 포함된 데이터를 기반으로 커서를 만듭니다. (유니코드)
LoadCursorW 애플리케이션 instance 연결된 실행 파일(.EXE) 파일에서 지정된 커서 리소스를 로드합니다. (유니코드)
LoadIconA 애플리케이션 instance 연결된 실행 파일(.exe) 파일에서 지정된 아이콘 리소스를 로드합니다. (ANSI)

LoadIconW 애플리케이션 instance 연결된 실행 파일(.exe) 파일에서 지정된 아이콘 리소스를 로드합니다. (유니코드)
LoadImageA 아이콘, 커서, 애니메이션 커서 또는 비트맵을 로드합니다. (ANSI)
LoadImageW 아이콘, 커서, 애니메이션 커서 또는 비트맵을 로드합니다. (유니코드)
LoadMenuA 애플리케이션 instance 연결된 실행 파일(.exe) 파일에서 지정된 메뉴 리소스를 로드합니다. (ANSI)
LoadMenuIndirectA 지정된 메뉴 템플릿을 메모리에 로드합니다. (ANSI)
LoadMenuIndirectW 지정된 메뉴 템플릿을 메모리에 로드합니다. (유니코드)
LoadMenuW 애플리케이션 instance 연결된 실행 파일(.exe) 파일에서 지정된 메뉴 리소스를 로드합니다. (유니코드)
LoadResource 메모리에서 지정된 리소스의 첫 번째 바이트에 대한 포인터를 가져오는 데 사용할 수 있는 핸들을 검색합니다.
LoadStringA 지정된 모듈과 연결된 실행 파일에서 문자열 리소스를 로드하고, 문자열을 버퍼에 복사하고, 종료 null 문자를 추가합니다. (ANSI)
LoadStringW 지정된 모듈과 연결된 실행 파일에서 문자열 리소스를 로드하고, 문자열을 버퍼에 복사하고, 종료 null 문자를 추가합니다. (유니코드)
LockResource 메모리에서 지정된 리소스에 대한 포인터를 검색합니다.

LookupIconIdFromDirectory 아이콘 또는 커서 데이터를 검색하여 현재 디스플레이 디바이스에 가장 적합한 아이콘 또는 커서를 검색합니다. (LookupIconIdFromDirectory)
LookupIconIdFromDirectoryEx 아이콘 또는 커서 데이터를 검색하여 현재 디스플레이 디바이스에 가장 적합한 아이콘 또는 커서를 검색합니다. (LookupIconIdFromDirectoryEx)
lstrcatA 한 문자열을 다른 문자열에 추가합니다. 경고 사용하지 마세요. (ANSI)
lstrcatW 한 문자열을 다른 문자열에 추가합니다. 경고 사용하지 마세요. (유니코드)
lstrcmpA 두 문자 문자열을 비교합니다. 비교는 대/소문자를 구분합니다. (ANSI)
lstrcmpiA 두 문자 문자열을 비교합니다. 대/소문자를 구분하지 않고 비교합니다. (ANSI)
lstrcmpiW 두 문자 문자열을 비교합니다. 대/소문자를 구분하지 않고 비교합니다. (유니코드)
lstrcmpW 두 문자 문자열을 비교합니다. 비교는 대/소문자를 구분합니다. (유니코드)
lstrcpyA 버퍼에 문자열을 복사합니다. (ANSI)
lstrcpynA 원본 문자열에서 지정된 수의 문자를 버퍼에 복사합니다. 경고 사용하지 마세요. (ANSI)
lstrcpynW 원본 문자열에서 지정된 수의 문자를 버퍼에 복사합니다. 경고 사용하지 마세요. (유니코드)
lstrcpyW 버퍼에 문자열을 복사합니다. (유니코드)

lstrlenA 지정된 문자열의 길이를 결정합니다(종료 null 문자를 포함하지 않음). (ANSI)
lstrlenW 지정된 문자열의 길이를 결정합니다(종료 null 문자를 포함하지 않음). (유니코드)
MAKEINTRESOURCEA 정수 값을 리소스 관리 함수와 호환되는 리소스 형식으로 변환합니다. 이 매크로는 리소스 이름이 포함된 문자열 대신 사용됩니다. (ANSI)
MAKEINTRESOURCEW 정수 값을 리소스 관리 함수와 호환되는 리소스 형식으로 변환합니다. 이 매크로는 리소스 이름이 포함된 문자열 대신 사용됩니다. (유니코드)
MenuItemFromPoint 지정된 위치에 있는 메뉴 항목(있는 경우)을 결정합니다.
ModifyMenuA 기존 메뉴 항목을 변경합니다. (ANSI)
ModifyMenuW 기존 메뉴 항목을 변경합니다. (유니코드)
OemToChara OEM 정의 문자 집합의 문자열을 ANSI 또는 와이드 문자열로 변환합니다. 경고 사용하지 마세요. (ANSI)
OemToCharBuffA OEM 정의 문자 집합에서 문자열의 지정된 수의 문자를 ANSI 또는 와이드 문자열로 변환합니다. (ANSI)
OemToCharBuffW OEM 정의 문자 집합에서 문자열의 지정된 수의 문자를 ANSI 또는 와이드 문자열로 변환합니다. (유니코드)
OemToCharW OEM 정의 문자 집합의 문자열을 ANSI 또는 와이드 문자열로 변환합니다. 경고 사용하지 마세요. (유니코드)

PrivateExtractIconsA 지정된 파일에서 추출된 아이콘에 대한 핸들 배열을 만듭니다. (ANSI)
PrivateExtractIconsW 지정된 파일에서 추출된 아이콘에 대한 핸들 배열을 만듭니다. (유니코드)
RemoveMenu 메뉴 항목을 삭제하거나 지정된 메뉴에서 하위 메뉴를 분리합니다.
SetCaretBlinkTime 캐리트 깜박임 시간을 지정된 시간(밀리초)으로 설정합니다. 깜박임 시간은 캐리트의 픽셀을 반전하는 데 필요한 경과 시간(밀리초)입니다.
SetCaretPos 지정된 좌표로 캐리트를 이동합니다. caret를 소유하는 창이 CS_OWNDC 클래스 스타일로 만들어진 경우 지정된 좌표에는 해당 창과 연결된 디바이스 컨텍스트의 매핑 모드가 적용됩니다.
SetCursor 커서 모양을 설정합니다.
SetCursorPos 커서를 지정된 화면 좌표로 이동합니다.
SetMenu 지정된 창에 새 메뉴를 할당합니다.
SetMenuDefaultItem 지정된 메뉴의 기본 메뉴 항목을 설정합니다.
SetMenuInfo 지정된 메뉴에 대한 정보를 설정합니다.
SetMenuItemBitmaps 지정된 비트맵을 메뉴 항목과 연결합니다. 메뉴 항목을 선택하든 선택 취소하든 관계없이 시스템은 메뉴 항목 옆에 적절한 비트맵을 표시합니다.

SetMenuItemInfoA 메뉴 항목에 대한 정보를 변경합니다. (ANSI)
SetMenuItemInfoW 메뉴 항목에 대한 정보를 변경합니다. (유니코드)
SetPhysicalCursorPos 실제 좌표에서 커서의 위치를 설정합니다.
SetSystemCursor 애플리케이션이 시스템 커서를 사용자 지정할 수 있도록 합니다. id 매개 변수로 지정된 시스템 커서의 내용을 hcur 매개 변수로 지정된 커서의 내용으로 바꾼 다음 hcur를 삭제합니다.
ShowCaret caret의 현재 위치에 있는 화면에 캐리트를 표시합니다. 캐리트가 표시되면 자동으로 깜박이기 시작합니다.
ShowCursor 커서를 표시하거나 숨깁니다. (ShowCursor)
SizeofResource 지정된 리소스의 크기(바이트)를 검색합니다.
StringCbCatA 한 문자열을 다른 문자열에 연결합니다. (StringCbCatA)
StringCbCatExA 한 문자열을 다른 문자열에 연결합니다. (StringCbCatExA)
StringCbCatExW 한 문자열을 다른 문자열에 연결합니다. (StringCbCatExW)
StringCbCatNA 한 문자열에서 다른 문자열로 지정된 바이트 수를 연결합니다. (StringCbCatNA)
StringCbCatNExA 한 문자열에서 다른 문자열로 지정된 바이트 수를 연결합니다. (StringCbCatNExA)

StringCbCatNExW 한 문자열에서 다른 문자열로 지정된 바이트 수를 연결합니다. (StringCbCatNExW)
StringCbCatNW 한 문자열에서 다른 문자열로 지정된 바이트 수를 연결합니다. (StringCbCatNW)
StringCbCatW 한 문자열을 다른 문자열에 연결합니다. (StringCbCatW)
StringCbCopyA 한 문자열을 다른 문자열에 복사합니다. (StringCbCopyA)
StringCbCopyExA 한 문자열을 다른 문자열에 복사합니다. (StringCbCopyExA)
StringCbCopyExW 한 문자열을 다른 문자열에 복사합니다. (StringCbCopyExW)
StringCbCopyNA 한 문자열에서 다른 문자열로 지정된 바이트 수를 복사합니다. (StringCbCopyNA)
StringCbCopyNExA 지정된 바이트 수를 한 문자열에서 다른 문자열로 복사합니다. (StringCbCopyNExA)
StringCbCopyNExW 지정된 바이트 수를 한 문자열에서 다른 문자열로 복사합니다. (StringCbCopyNExW)
StringCbCopyNW 지정된 바이트 수를 한 문자열에서 다른 문자열로 복사합니다. (StringCbCopyNW)
StringCbCopyW 한 문자열을 다른 문자열에 복사합니다. (StringCbCopyW)
StringCbGetsA 줄 바꿈 문자('\n')를 포함하여 stdin에서 한 줄의 텍스트를 가져옵니다. (StringCbGetsA)

StringCbGetsExA 줄 바꿈 문자('\n')를 포함하여 stdin에서 한 줄의 텍스트를 가져옵니다. (StringCbGetsExA)
StringCbGetsExW 줄 바꿈 문자('\n')를 포함하여 stdin에서 한 줄의 텍스트를 가져옵니다. (StringCbGetsExW)
StringCbGetsW 줄 바꿈 문자('\n')를 포함하여 stdin에서 한 줄의 텍스트를 가져옵니다. (StringCbGetsW)
StringCbLengthA 문자열이 지정된 길이(바이트)를 초과하는지 여부를 결정합니다. (ANSI)
StringCbLengthW 문자열이 지정된 길이(바이트)를 초과하는지 여부를 결정합니다. (유니코드)
StringCbPrintf_lA 서식이 지정된 데이터를 지정된 문자열에 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCbPrintf_lA)
StringCbPrintf_lExA 서식이 지정된 데이터를 지정된 문자열에 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCbPrintf_lExA)
StringCbPrintf_lExW 서식이 지정된 데이터를 지정된 문자열에 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCbPrintf_lExW)
StringCbPrintf_lW 서식이 지정된 데이터를 지정된 문자열에 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCbPrintf_lW)
StringCbPrintfA 서식이 지정된 데이터를 지정된 문자열에 씁니다. (StringCbPrintfA)
StringCbPrintfExA 서식이 지정된 데이터를 지정된 문자열에 씁니다. (StringCbPrintfExA)

StringCbPrintfExW 서식이 지정된 데이터를 지정된 문자열에 씁니다. (StringCbPrintfExW)
StringCbPrintfW 서식이 지정된 데이터를 지정된 문자열에 씁니다. (StringCbPrintfW)
StringCbVPrintf_lA 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCbVPrintf_lA)
StringCbVPrintf_lExA 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCbVPrintf_lExA)
StringCbVPrintf_lExW 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCbVPrintf_lExW)
StringCbVPrintf_lW 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCbVPrintf_lW)
StringCbVPrintfA 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. (StringCbVPrintfA)
StringCbVPrintfExA 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. (StringCbVPrintfExA)
StringCbVPrintfExW 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. (StringCbVPrintfExW)
StringCbVPrintfW 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. (StringCbVPrintfW)

StringCchCatA 한 문자열을 다른 문자열에 연결합니다. (StringCchCatA)
StringCchCatExA 한 문자열을 다른 문자열에 연결합니다. (StringCchCatExA)
StringCchCatExW 한 문자열을 다른 문자열에 연결합니다. (StringCchCatExW)
StringCchCatNA 한 문자열에서 다른 문자열로 지정된 수의 문자를 연결합니다. (StringCchCatNA)
StringCchCatNExA 한 문자열에서 다른 문자열로 지정된 수의 문자를 연결합니다. (StringCchCatNExA)
StringCchCatNExW 한 문자열에서 다른 문자열로 지정된 수의 문자를 연결합니다. (StringCchCatNExW)
StringCchCatNW 한 문자열에서 다른 문자열로 지정된 수의 문자를 연결합니다. (StringCchCatNW)
StringCchCatW 한 문자열을 다른 문자열에 연결합니다. (StringCchCatW)
StringCchCopyA 한 문자열을 다른 문자열에 복사합니다. (StringCchCopyA)
StringCchCopyExA 한 문자열을 다른 문자열에 복사합니다. (StringCchCopyExA)
StringCchCopyExW 한 문자열을 다른 문자열에 복사합니다. (StringCchCopyExW)
StringCchCopyNA 지정된 수의 문자를 한 문자열에서 다른 문자열로 복사합니다. (StringCchCopyNA)

StringCchCopyNExA 지정된 수의 문자를 한 문자열에서 다른 문자열로 복사합니다. (StringCchCopyNExA)
StringCchCopyNExW 지정된 수의 문자를 한 문자열에서 다른 문자열로 복사합니다. (StringCchCopyNExW)
StringCchCopyNW 지정된 수의 문자를 한 문자열에서 다른 문자열로 복사합니다. (StringCchCopyNW)
StringCchCopyW 한 문자열을 다른 문자열에 복사합니다. (StringCchCopyW)
StringCchGetsA 줄 바꿈 문자('\n')를 포함하여 stdin에서 한 줄의 텍스트를 가져옵니다. (StringCchGetsA)
StringCchGetsExA 줄 바꿈 문자('\n')를 포함하여 stdin에서 한 줄의 텍스트를 가져옵니다. (StringCchGetsExA)
StringCchGetsExW 줄 바꿈 문자('\n')를 포함하여 stdin에서 한 줄의 텍스트를 가져옵니다. (StringCchGetsExW)
StringCchGetsW 줄 바꿈 문자('\n')를 포함하여 stdin에서 한 줄의 텍스트를 가져옵니다. (StringCchGetsW)
StringCchLengthA 문자열이 지정된 길이(문자)를 초과하는지 여부를 결정합니다. (ANSI)
StringCchLengthW 문자열이 지정된 길이(문자)를 초과하는지 여부를 결정합니다. (유니코드)
StringCchPrintf_lA 서식이 지정된 데이터를 지정된 문자열에 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCchPrintf_lA)
StringCchPrintf_lExA 서식이 지정된 데이터를 지정된 문자열에 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCchPrintf_lExA)

StringCchPrintf_lExW 서식이 지정된 데이터를 지정된 문자열에 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCchPrintf_lExW)
StringCchPrintf_lW 서식이 지정된 데이터를 지정된 문자열에 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCchPrintf_lW)
StringCchPrintfA 서식이 지정된 데이터를 지정된 문자열에 씁니다. (StringCchPrintfA)
StringCchPrintfExA 서식이 지정된 데이터를 지정된 문자열에 씁니다. (StringCchPrintfExA)
StringCchPrintfExW 서식이 지정된 데이터를 지정된 문자열에 씁니다. (StringCchPrintfExW)
StringCchPrintfW 서식이 지정된 데이터를 지정된 문자열에 씁니다. (StringCchPrintfW)
StringCchVPrintf_lA 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCchVPrintf_lA)
StringCchVPrintf_lExA 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCchVPrintf_lExA)
StringCchVPrintf_lExW 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCchVPrintf_lExW)
StringCchVPrintf_lW 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCchVPrintf_lW)

StringCchVPrintfA 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. (StringCchVPrintfA)
StringCchVPrintfExA 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. (StringCchVPrintfExA)
StringCchVPrintfExW 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. (StringCchVPrintfExW)
StringCchVPrintfW 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. (StringCchVPrintfW)
TrackPopupMenu 지정된 위치에 바로 가기 메뉴를 표시하고 메뉴의 항목 선택을 추적합니다. 바로 가기 메뉴는 화면의 아무 곳에나 나타날 수 있습니다.
TrackPopupMenuEx 지정된 위치에 바로 가기 메뉴를 표시하고 바로 가기 메뉴에서 항목 선택을 추적합니다. 바로 가기 메뉴는 화면의 아무 곳에나 나타날 수 있습니다.
TranslateAcceleratorA 메뉴 명령에 대한 가속기 키를 처리합니다. (ANSI)
TranslateAcceleratorW 메뉴 명령에 대한 가속기 키를 처리합니다. (유니코드)
UpdateResourceA PE(이식 가능한 실행 파일) 파일의 리소스를 추가, 삭제 또는 대체합니다. (ANSI)
UpdateResourceW PE(이식 가능한 실행 파일) 파일의 리소스를 추가, 삭제 또는 대체합니다. (유니코드)

VerFindFileA 시스템에서 파일의 다른 버전을 찾을지 여부에 따라 파일을 설치할 위치를 결정합니다. 지정된 버퍼에서 VerFindFile이 반환하는 값은 VerInstallFile 함수에 대한 후속 호출에서 사용됩니다. (ANSI)
VerFindFileW 시스템에서 파일의 다른 버전을 찾을지 여부에 따라 파일을 설치할 위치를 결정합니다. 지정된 버퍼에서 VerFindFile이 반환하는 값은 VerInstallFile 함수에 대한 후속 호출에서 사용됩니다. (유니코드)
VerInstallFileA VerFindFile 함수에서 반환된 정보에 따라 지정된 파일을 설치합니다. VerInstallFile은 필요한 경우 파일의 압축을 풀고 고유한 파일 이름을 할당하고 오래된 파일과 같은 오류를 확인합니다. (ANSI)
VerInstallFileW VerFindFile 함수에서 반환된 정보에 따라 지정된 파일을 설치합니다. VerInstallFile은 필요한 경우 파일의 압축을 풀고 고유한 파일 이름을 할당하고 오래된 파일과 같은 오류를 확인합니다. (유니코드)
VerLanguageNameA 지정된 이진 Microsoft 언어 식별자와 연결된 언어에 대한 설명 문자열을 검색합니다. (ANSI)
VerLanguageNameW 지정된 이진 Microsoft 언어 식별자와 연결된 언어에 대한 설명 문자열을 검색합니다. (유니코드)
VerQueryValueA 지정된 버전 정보 리소스에서 지정된 버전 정보를 검색합니다. (ANSI)
VerQueryValueW 지정된 버전 정보 리소스에서 지정된 버전 정보를 검색합니다. (유니코드)
wsprintfA 지정된 버퍼에 서식이 지정된 데이터를 씁니다. (ANSI)
wsprintfW 지정된 버퍼에 서식이 지정된 데이터를 씁니다. (유니코드)
wvsprintfA 인수 목록에 대한 포인터를 사용하여 지정된 버퍼에 서식이 지정된 데이터를 씁니다. (ANSI)

wvsprintfW 인수 목록에 대한 포인터를 사용하여 지정된 버퍼에 서식이 지정된 데이터를 씁니다. (유니코드)

구조체

가속 액셀러레이터 테이블에 사용되는 가속기 키를 정의합니다.
CURSORINFO 전역 커서 정보를 포함합니다.
커서 셰이프 커서에 대한 정보를 포함합니다.
ICONINFO 아이콘 또는 커서에 대한 정보를 포함합니다.
ICONINFOEXA 아이콘 또는 커서에 대한 정보를 포함합니다. ICONINFO를 확장합니다. GetIconInfoEx에서 사용됩니다. (ANSI)
ICONINFOEXW 아이콘 또는 커서에 대한 정보를 포함합니다. ICONINFO를 확장합니다. GetIconInfoEx에서 사용됩니다. (유니코드)
ICONMETRICSA 아이콘과 연결된 확장 가능한 메트릭을 포함합니다. 이 구조체는 SPI_GETICONMETRICS 또는 SPI_SETICONMETRICS 작업이 지정된 경우 SystemParametersInfo 함수와 함께 사용됩니다. (ANSI)
ICONMETRICSW 아이콘과 연결된 확장 가능한 메트릭을 포함합니다. 이 구조체는 SPI_GETICONMETRICS 또는 SPI_SETICONMETRICS 작업이 지정된 경우 SystemParametersInfo 함수와 함께 사용됩니다. (유니코드)

IndexedResourceQualifier 리소스가 적절한 컨텍스트를 나타냅니다.
MDINEXTMENU 활성화할 메뉴에 대한 정보를 포함합니다.
MENUBARINFO 메뉴 모음 정보를 포함합니다.
MENUGETOBJECTINFO 마우스 커서가 있는 메뉴에 대한 정보를 포함합니다.
MENUINFO 메뉴에 대한 정보를 포함합니다.
MENUITEMINFOA 메뉴 항목에 대한 정보를 포함합니다. (MENUITEMINFOA)
MENUITEMINFOW 메뉴 항목에 대한 정보를 포함합니다. (MENUITEMINFOW)
MENUITEMTEMPLATE 메뉴 템플릿에서 메뉴 항목을 정의합니다.
MENUITEMTEMPLATEHEADER 메뉴 템플릿의 헤더를 정의합니다. 전체 메뉴 템플릿은 헤더와 하나 이상의 메뉴 항목 목록으로 구성됩니다.
MESSAGE_RESOURCE_BLOCK LowId 및 HighId 멤버가 나타내는 범위의 식별자가 있는 메시지 문자열에 대한 정보를 포함합니다.
MESSAGE_RESOURCE_DATA 오류 메시지로 표시하거나 메시지 테이블 리소스의 메시지 상자에 표시할 서식이 지정된 텍스트에 대한 정보를 포함합니다.
MESSAGE_RESOURCE_ENTRY 메시지 테이블 리소스에 대한 오류 메시지 또는 메시지 상자 표시 텍스트를 포함합니다.

TPMPARAMS TrackPopupMenuEx 함수에 대한 확장 매개 변수를 포함합니다.
VS_FIXEDFILEINFO 파일에 대한 버전 정보를 포함합니다. 이 정보는 언어 및 코드 페이지 독립적입니다.

Microsoft HTML 도움말 1.4

아티클2023. 08. 22.

Microsoft HTML 도움말 1.4 기술에 대한 개요입니다.

Microsoft HTML 도움말 1.4를 개발하려면 다음 헤더가 필요합니다.

- htmlhelp.h
- infotech.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Microsoft HTML 도움말 1.4

Functions

추가 속성 ID 쌍과 속성 형식이 포함된 헤더가 지정된 경우 결과 집합에 열을 추가합니다.
추가 결과 집합에 열을 추가합니다. (오버로드 2/3)
추가 결과 집합에 열을 추가합니다. (오버로드 1/3)
추가 결과 집합에 열을 추가합니다. (오버로드 3/3)
지우기 속성 목록과 연결된 메모리를 지우고 목록을 다시 초기화합니다.
닫기 데이터베이스를 닫습니다.
닫기 단어 휠을 닫습니다.

Count 단어 휠의 항목 수를 반환합니다.
CreateObject *pdwObjInstance 매개 변수를 통해 나중에 참조할 수 있는 명명되지 않은 개체를 만듭니다.
가져오기 지정된 속성 ID와 연결된 속성 개체를 반환합니다.
가져오기 지정한 행과 열의 속성을 가져오고 지정된 속성 개체를 채웁니다.
GetDataSize 속성 데이터를 저장하는 데 필요한 바이트 수를 반환합니다.
GetFirst 속성 목록의 첫 번째 속성 개체를 반환합니다.
GetHeaderSize 헤더를 저장하는 데 필요한 바이트 수를 반환합니다.
GetObject dwObjInstance 매개 변수로 식별된 개체에서 지정된 IUnknown 기반 인터페이스를 검색합니다.
GetRowCount 결과 집합의 행 수를 가져옵니다.
HtmlHelpA 도움말 창을 표시합니다. (ANSI)
HtmlHelpW 도움말 창을 표시합니다. (유니코드)
조회 항목을 조회하고 버퍼의 내용을 반환합니다. (오버로드 2/3)
조회

항목을 조회하고 버퍼의 내용을 반환합니다. (오버로드 1/3)
조회 항목을 조회하고 버퍼의 내용을 반환합니다. (오버로드 3/3)
열기 데이터베이스를 엽니다.
열기 단어 휠을 엽니다.
SaveData 속성 목록의 데이터 크기와 데이터를 버퍼에 저장합니다.
SaveHeader 속성 목록에서 버퍼로 속성 ID 및 데이터 형식을 저장합니다. TRUE의 지속성 상태로 표시된 속성만 저장합니다.
설정 속성을 지정된 값으로 설정하거나 목록에서 속성을 삭제합니다. (오버로드 1/3)
설정 속성을 지정된 값으로 설정하거나 목록에서 속성을 삭제합니다. (오버로드 3/3)
설정 속성을 지정된 값으로 설정하거나 목록에서 속성을 삭제합니다. (오버로드 2/3)
SetLocaleInfo 형태소 분석기 에 대한 로캘 정보를 설정합니다.
SetPersist 모든 속성의 지속성 상태를 설정하거나 해제합니다.
SetPersist 지정된 속성의 지속성 상태를 설정하거나 해제합니다.

인터페이스

IITDatabase 데이터베이스 개체를 열고 닫고 데이터베이스에 저장된 개체를 인스턴스화하는 데 이 인터페이스를 사용합니다.
IITPropList 이 인터페이스를 사용하여 단어 휠 및 인덱스와 같은 빌드 개체에 대한 속성을 설정합니다. 문서 빌드 프로세스에서 이러한 메서드를 호출하여 모든 빌드 개체에 대한 속성을 정의합니다.
IITResultSet 런타임 애플리케이션에서 이 인터페이스를 사용하여 결과 집합에 대한 정보를 초기화, 빌드 및 가져옵니다.
IITWordWheel 이 인터페이스를 사용하여 단어 휠 키워드(keyword) 조회를 수행합니다. Lookup 메서드는 검색을 수행하는 여러 가지 방법을 제공합니다._it 지정된 접두사를 기반으로 정확한 일치 또는 가장 가까운 근사값을 반환합니다.
IStemmerConfig 이 인터페이스를 사용하여 형태소 분석을 제어하는 구성 정보를 제공합니다.

구조체

HH_AKLINK 검색하려는 하나 이상의 ALink 이름 또는 KLink 키워드를 지정하려면 이 구조를 사용합니다.
HH_FTS_QUERY 전체 텍스트 검색에는 이 구조를 사용합니다.
HH_POPUP 이 구조를 사용하여 팝업 창의 특성을 지정하거나 수정합니다.
HH_WINTYPE 창 형식의 특성을 지정하거나 수정하려면 이 구조를 사용합니다.
HHN_NOTIFY

이 구조를 사용하여 탐색된 토픽의 파일 이름을 반환하거나 생성된 도움말 창의 창 형식 이름을 반환합니다.
HHNTRACK 이 구조체는 현재 항목의 파일 이름과 도구 모음에서 숨기기 단추를 클릭하여 탐색 창을 숨기는 것과 같이 발생할 사용자 작업을 지정하는 상수를 반환합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Microsoft 관리 콘솔 2.0

아티클2023. 03. 09.

Microsoft Management Console 2.0 기술에 대한 개요입니다.

Microsoft Management Console 2.0 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Microsoft 관리 콘솔 2.0

열거형

_DocumentMode DocumentMode 열거형은 Document.Mode 속성에서 사용되며 문서를 여는 방법을 지정합니다. 이 열거형은 MMC 2.0 Automation 개체 모델에 적용됩니다.
_ExportListOptions ExportListOptions 열거형은 View.ExportList 메서드에서 사용되며 목록 보기 콘텐츠를 파일에 쓸 때 옵션을 지정합니다.
_ListViewMode ListViewMode 열거형은 View.ListViewMode 속성에서 목록 보기를 정의하는 데 사용됩니다.
_ViewOptions ViewOptions 열거형은 Views.Add 메서드에서 사용되며 뷰, 범위 트리 및 도구 모음의 표시 여부와 보기의 지속성 상태를 지정합니다.
DATA_OBJECT_TYPES DATA_OBJECT_TYPES 열거형은 IComponentData::QueryDataObject 및 IComponent::QueryDataObject의 형식 매개 변수에 의해 지정된 쿠키에 대한 컨텍스트 정보를 가져오는 데 사용됩니다.
IconIdentifier IconIdentifier 열거형은 MMC 1.2에서 도입되었습니다.
MMC_ACTION_TYPE MMC_ACTION_TYPE 열거형은 MMC 1.1에 도입되었습니다.

MMC_BUTTON_STATE MMC_BUTTON_STATE 열거형은 MMC에서 사용할 수 있는 단추의 가능한 상태를 정의합니다. 이러한 값은 IConsoleVerb::GetVerbState, IConsoleVerb::SetVerbState, IToolbar::GetButtonState 및 IToolbar::SetButtonState의 nState 매개 변수에 사용됩니다.
MMC_CONSOLE_VERB MMC_CONSOLE_VERB 열거형은 MMC 동사에 사용할 수 있는 명령 식별자를 정의합니다. 이러한 값은 IConsoleVerb::GetVerbState, IConsoleVerb::SetVerbState 및 IConsoleVerb::SetDefaultVerb의 m_eCmdID 매개 변수에 사용됩니다.
MMC_CONTROL_TYPE MMC_CONTROL_TYPE 열거형은 컨트롤 표시줄에 표시할 수 있는 컨트롤의 종류를 정의합니다. 값은 IControlbar::Attach 및 IControlbar::Create 메서드의 nType 매개 변수에서 사용할 수 있습니다.
MMC_FILTER_CHANGE_CODE MMC_FILTER_CHANGE_CODE 열거형은 MMC 1.2에 도입되었습니다.
MMC_FILTER_TYPE MMC_FILTER_TYPE 열거형은 MMC 1.2에 도입되었습니다.
MMC_MENU_COMMAND_IDS MMC_MENU_COMMAND_IDS 열거형은 MMC에서 예약된 명령 식별자를 정의합니다.
MMC_NOTIFY_TYPE MMC_NOTIFY_TYPE 열거형은 IComponentData::Notify, IComponent::Notify 또는 IExtendControlbar::ControlbarNotify를 호출할 때 콘솔의 노드 관리자가 스냅인으로 보낼 수 있는 사용자 작업의 알림을 정의합니다.
MMC_PROPERTY_ACTION MMC_PROPERTY_ACTION 열거형은 MMC_SNAPIN_PROPERTY 구조체에 포함된 속성에 발생할 수 있는 작업을 지정합니다.
MMC_RESULT_VIEW_STYLE MMC_RESULT_VIEW_STYLE 열거형은 MMC 결과 보기 창에서 보기 스타일을 설정하는 데 사용할 수 있는 Win32 목록 보기 스타일(LVS_*) 플래그를 정의합니다. IResultData::ModifyViewStyle 메서드의 매개 변수 추가 및 제거에 사용할 수 있습니다.
MMC_SCOPE_ITEM_STATE SCOPEDATAITEM 구조체의 nState 멤버를 지정하는 데 사용됩니다.

MMC_TASK_DISPLAY_TYPE MMC_TASK_DISPLAY_TYPE 열거형은 MMC 1.1에 도입되었습니다.
MMC_VIEW_TYPE MMC_VIEW_TYPE 열거형은 결과 뷰 형식을 지정하고 RESULT_VIEW_TYPE_INFO 구조체에 사용됩니다.

Functions

AddBitmap 스냅인을 사용하여 도구 모음에 이미지를 추가할 수 있습니다.
AddButton IMenuButton::AddButton 메서드를 사용하면 사용자가 특정 보기의 MMC 메뉴 모음에 단추를 추가할 수 있습니다.
AddButtons 스냅인을 사용하여 도구 모음에 단추 배열을 추가할 수 있습니다.
AddExtension IConsoleNameSpace2::AddExtension 메서드를 사용하면 스냅인에서 선택한 항목의 네임스페이스를 동적으로 확장하는 확장 스냅인을 추가할 수 있습니다.
AddExtensionPages IPropertySheetProvider::AddExtensionPages 메서드는 확장 스냅인에서 페이지를 수집합니다.
Additem IContextMenuCallback::AddItem 메서드는 상황에 맞는 메뉴에 단일 항목을 추가합니다.
Additem IContextMenuCallback2::AddItem 메서드는 상황에 맞는 메뉴에 단일 항목을 추가합니다.
AddMenuItems IExtendContextMenu::AddMenuItems 메서드를 사용하면 스냅인에서 상황에 맞는 메뉴에 항목을 추가할 수 있습니다.

Addpage IPropertySheetCallback::AddPage 메서드를 사용하면 스냅인에서 속성 시트에 페이지를 추가할 수 있습니다.
AddPrimaryExtensionItems IContextMenuProvider::AddPrimaryExtensionItems 메서드를 사용하면 특정 확장이 이 상황에 맞는 메뉴에 정의된 삽입 지점에 항목을 추가할 수 있습니다.
AddPrimaryPages IPropertySheetProvider::AddPrimaryPages 메서드는 기본 스냅인에서 페이지를 수집합니다.
AddPropertyName AddPropertyName 메서드는 스냅인에서 사용할 속성을 이름으로 추가합니다.
AddString 스냅인을 사용하여 스냅인의 문자열 테이블에 문자열을 추가할 수 있습니다.
AddThirdPartyExtensionItems IContextMenuProvider::AddThirdPartyExtensionItems 메서드를 사용하면 타사 확장에서 이 상황에 맞는 메뉴의 지정된 삽입 지점에 있는 항목을 추가할 수 있습니다.
AddView 결과 창에 보기를 추가합니다.
연결 IControlbar::Attach 메서드를 사용하면 스냅인에서 컨트롤을 컨트롤 막대와 연결할 수 있습니다.
CacheHint 가상 목록이 항목 범위에 대한 표시 정보를 요청하려고 할 때 호출되므로 스냅인에서 최적화를 수행할 수 있는 경우 정보를 미리 수집할 수 있습니다.
지우기 IMessageView::Clear 메서드를 사용하면 스냅인에서 MMC 메시지 OCX 컨트롤을 사용하여 표시되는 결과 창 메시지의 제목, 텍스트 및 아이콘을 지울 수 있습니다.
복제 IEnumTASK::Clone 메서드는 이 IEnumTASK 개체와 상태가 같은 새 IEnumTASK 개체를 만듭니다. MMC는 이 메서드를 사용하지 않습니다. 완전성을 위해 포함해야 합니다.

명령 IExtendContextMenu::Command 메서드는 IExtendContextMenu::AddMenuItems를 사용하여 상황에 맞는 메뉴에 추가한 항목 중 하나를 선택한 경우 호출됩니다.
비교 기본 스냅인에서 결과 창에 삽입하는 결과 항목을 정렬하기 위해 쿠키를 비교하는 방법을 제공합니다.
비교 기본 스냅인에서 결과 창에 삽입하는 범위 및 결과 항목을 정렬하기 위한 항목을 비교하는 방법을 제공합니다.
CompareObjects IComponent::CompareObjects 메서드를 사용하면 스냅인에서 IComponent::QueryDataObject를 통해 획득한 두 데이터 개체를 비교할 수 있습니다. 데이터 개체는 IComponent의 서로 다른 두 인스턴스에서 가져올 수 있습니다.
CompareObjects IComponentData::CompareObjects 메서드를 사용하면 스냅인에서 QueryDataObject를 통해 획득한 두 데이터 개체를 비교할 수 있습니다. 데이터 개체는 IComponentData의 서로 다른 두 인스턴스에서 가져올 수 있습니다.
ControlbarNotify IExtendControlbar::ControlbarNotify 메서드는 사용자 작업의 결과로 콘솔에서 스냅인으로 전송되는 알림을 지정합니다.
만들기 IControlbar::Create 메서드는 스냅인에서 요청한 컨트롤을 만들고 반환합니다.
CreateComponent IComponentData::CreateComponent 메서드는 이 IComponentData 인터페이스와 연결될 IComponent의 인스턴스를 만듭니다.
CreatePropertyPages 속성 시트에 페이지를 추가합니다.
CreatePropertySheet 속성 시트 프레임을 만듭니다.

DeleteAllRsltItems IResultData::D eleteAllRsltItems 메서드를 사용하면 스냅인에서 결과 보기 창에서 사용자가 삽입한 모든 항목 및 하위 항목을 제거할 수 있습니다.
DeleteAllStrings IStringTable::D eleteAllStrings 메서드를 사용하면 스냅인에서 스냅인의 문자열 테이블에서 모든 문자열을 삭제할 수 있습니다.
DeleteButton 스냅인을 사용하여 지정된 도구 모음 단추를 제거할 수 있습니다.
DeleteColumn 결과 뷰의 머리글에서 열을 제거합니다.
DeleteItem IConsoleNameSpace2::D eleteItem 메서드 IConsoleNameSpaceenables는 스냅인을 통해 범위 창에서 단일 항목을 삭제합니다.
DeleteItem 스냅인을 사용하여 결과 보기 창에서 단일 항목을 삭제할 수 있습니다.
DeleteString 스냅인이 스냅인 문자열 테이블에서 지정된 문자열을 삭제할 수 있도록 합니다.
파괴 IComponent::D estroy 메서드는 이 구성 요소에서 보유한 콘솔에 대한 모든 참조를 해제합니다.
파괴 IComponentData::D estroy 메서드는 콘솔에 대한 모든 참조를 해제합니다.
분리 IControlbar::D etach 메서드는 컨트롤과 컨트롤 막대 간의 연결을 끊습니다. 이 명령은 컨트롤이 표시되는 컨트롤 표시줄에서 컨트롤을 제거하거나 분리합니다.
EmptyMenuList IContextMenuProvider::EmptyMenuList 메서드는 상황에 맞는 메뉴를 지웁니다.

EnableAllExtensions IRequiredExtensions::EnableAllExtensions 메서드를 사용하면 스냅인에 등록된 모든 확장 스냅인이 필요하도록 스냅인을 지정할 수 있습니다.
열거 스냅인의 문자열 테이블에서 문자열을 반환할 수 있는 열거자의 IEnumString 인터페이스에 대한 포인터를 제공합니다.
EnumTasks IExtendTaskPad::EnumTasks 메서드를 사용하면 MMC가 스냅인 작업을 포함하는 개체의 IEnumTASK 인터페이스에 대한 포인터를 가져올 수 있습니다.
확장 IConsole2::Expand 메서드를 사용하면 스냅인에서 범위 창에서 항목을 확장하거나 축소할 수 있습니다.
확장 IConsoleNameSpace2::Expand 메서드를 사용하면 스냅인에서 범위 창에서 항목을 눈에 띄게 확장하지 않고도 네임스페이스의 항목을 확장할 수 있습니다.
Finditem 지정된 문자열과 일치하는 가상 목록에서 다음 항목을 찾습니다.
FindItemByLParam IResultData::FindItemByLParam 메서드를 사용하면 스냅인에서 사용자가 삽입한 lParam 값에 따라 항목 또는 하위 항목을 찾을 수 있습니다.
FindPropertySheet 특정 속성 시트가 있는지 여부를 확인합니다.
FindString 스냅인 문자열 테이블에서 문자열을 검색할 수 있도록 합니다.
GetBackground IExtendTaskPad::GetBackground 메서드를 사용하면 MMC가 MMC 작업 패드 템플릿을 사용하는 작업 패드에 표시할 작업 패드의 배경 이미지를 가져올 수 있습니다.

GetButtonState 스냅인을 사용하여 단추의 특성을 가져올 수 있습니다.
GetChildItem IConsoleNameSpace2::GetChildItem 메서드를 사용하면 스냅인에서 범위 창의 자식 항목에 대한 핸들을 가져올 수 있습니다.
GetColumnConfigData IColumnData::GetColumnConfigData 메서드를 사용하면 스냅인에서 MMC에서 메모리에 저장된 열 집합에 있는 각 열의 현재 너비, 순서 및 숨겨진 상태를 검색할 수 있습니다.
GetColumnFilter IHeaderCtrl2::GetColumnFilter 메서드는 지정된 열에 설정된 필터 값을 검색합니다.
GetColumnSortData IColumnData::GetColumnSortData 메서드를 사용하면 스냅인을 사용하여 메모리에서 정렬된 열과 열 집합의 열 정렬 방향을 검색할 수 있습니다.
GetColumnText 지정된 열에서 텍스트를 검색합니다.
GetColumnWidth 열의 너비를 픽셀 단위로 검색합니다.
GetDefaultVerb GetDefaultVerb 메서드는 스냅인의 기본 동사를 가져옵니다.
GetDescriptiveText IExtendTaskPad::GetDescriptiveText 메서드를 사용하면 MMC가 MMC 작업 패드 템플릿을 사용하는 작업 패드에 표시할 작업 패드의 설명 텍스트를 가져올 수 있습니다.
GetDisplayInfo IComponent::GetDisplayInfo 메서드는 결과 창의 항목에 대한 표시 정보를 검색합니다.
GetDisplayInfo IComponentData::GetDisplayInfo 메서드는 범위 항목에 대한 표시 정보를 검색합니다.

GetFirstExtension 스냅인에서 필요한 확장 스냅인 목록의 첫 번째 확장 스냅인을 지정할 수 있습니다.
GetHelpTopic 스냅인을 사용하여 컴파일된 HTML 도움말 파일을 MMC 도움말 컬렉션 파일에 추가할 수 있습니다.
GetItem IConsoleNameSpace2::GetItem 메서드를 사용하면 스냅인에서 단일 범위 항목의 특성 중 일부 또는 전부를 검색할 수 있습니다.
GetItem 사용자가 단일 항목의 매개 변수를 검색할 수 있도록 합니다.
GetLinkedTopics 스냅인을 사용하여 스냅인의 도움말 파일(GetHelpTopic 메서드에 지정됨)에 연결된 HTML 도움말 파일의 이름과 위치를 지정할 수 있습니다.
GetListPadInfo IExtendTaskPad::GetListPadInfo 메서드는 목록 보기 작업 패드에만 사용됩니다.
GetMainWindow 주 프레임 창에 대한 핸들을 검색합니다.
GetMMCVersion GetMMCVersion 메서드는 MMC 애플리케이션에 대한 버전 정보를 검색합니다.
GetNextExtension 스냅인을 사용하여 필요한 확장 스냅인 목록에서 다음 확장 스냅인을 지정할 수 있습니다.
GetNextItem IConsoleNameSpace2::GetNextItem 메서드를 사용하면 스냅인에서 범위 보기의 다음 항목에 대한 핸들을 검색할 수 있습니다.
GetNextItem IResultData::GetNextItem 메서드는 지정된 상태 플래그가 설정된 결과 보기의 다음 항목을 가져옵니다.

GetParentItem IConsoleNameSpace2::GetParentItem 메서드를 사용하면 스냅인에서 범위 보기의 부모 항목에 대한 핸들을 검색할 수 있습니다.
Getproperty GetProperty 메서드는 노드에 대한 텍스트 전용 속성 값을 검색합니다. MMC 2.0 Automation 개체 모델을 기반으로 하는 애플리케이션이 Node.Property 속성을 검색할 때 INodeProperties::GetProperty 메서드의 구현이 호출됩니다.
GetProvider ISnapinAbout::GetProvider 메서드를 사용하면 콘솔에서 스냅인 공급자 이름을 가져올 수 있습니다.
GetResultViewType IComponent::GetResultViewType 메서드는 결과 창 보기를 결정합니다.
GetResultViewType2 GetResultViewType2 메서드는 결과 뷰 형식을 검색합니다. 이 메서드는 IComponent::GetResultViewType 메서드를 대체합니다.
GetSnapinDescription 콘솔이 스냅인의 설명 상자에 대한 텍스트를 가져올 수 있도록 합니다.
GetSnapinImage 콘솔이 정보 상자에서 사용할 스냅인의 기본 아이콘을 가져올 수 있도록 합니다.
GetSnapinVersion 콘솔에서 스냅인의 버전 번호를 가져올 수 있도록 합니다.
GetStaticFolderImage ISnapinAbout::GetStaticFolderImage 메서드를 사용하면 콘솔에서 범위 및 결과 창에 대한 정적 폴더 이미지를 가져올 수 있습니다.
GetString 스냅인이 스냅인의 문자열 테이블에서 문자열을 검색할 수 있도록 합니다.
GetStringLength 스냅인을 사용하여 스냅인 문자열 테이블에서 문자열의 길이를 확인할 수 있습니다.

GetTitle IExtendTaskPad::GetTitle 메서드를 사용하면 MMC에서 MMC 작업 패드 템플릿을 사용하는 작업 패드에 표시할 작업 패드 제목 텍스트를 가져올 수 있습니다.
GetVerbState GetVerbState 메서드를 사용하면 스냅인에서 지정된 동사의 현재 상태를 가져올 수 있습니다.
GetViewMode 스냅인을 사용하여 결과 보기 창에 대한 보기 모드를 검색할 수 있습니다.
GetViews GetViews 메서드는 확장 보기에 대한 정보를 검색하고 결과 창에 확장 보기를 추가합니다.
GetWatermarks IExtendPropertySheet2::GetWatermarks 메서드는 마법사 97 스타일 마법사로 구현된 마법사 시트에 대한 워터마크 비트맵 및 헤더 비트맵을 가져옵니다.
ImageListSetIcon IImageList::ImageListSetIcon 메서드를 사용하면 사용자가 이미지 목록에 아이콘을 설정하거나 아이콘이 없는 경우 아이콘을 만들 수 있습니다.
ImageListSetStrip IImageList::ImageListSetStrip 메서드를 사용하면 사용자가 nStartLoc로 식별된 위치에서 시작하여 비트맵 쌍(크고 작은 아이콘)을 사용하여 이미지 목록에 아이콘 스트립을 추가할 수 있습니다.
초기화 Initialize 메서드는 스냅인을 초기화합니다.
초기화 IComponent::Initialize 메서드는 콘솔에 진입점을 제공합니다.
초기화 IComponentData::Initialize 메서드는 콘솔에 진입점을 제공합니다.
InsertButton 스냅인을 사용하여 도구 모음에 단일 단추를 추가할 수 있습니다.

InsertColumn 기본 결과 창에 열을 추가합니다.
Insertitem IConsoleNameSpace2::InsertItem 메서드를 사용하면 스냅인에서 단일 항목을 범위 보기에 삽입할 수 있습니다.
Insertitem IResultData::InsertItem 메서드를 사용하면 스냅인에서 결과 창 보기에 단일 새 항목을 추가할 수 있습니다.
IS_SPECIAL_COOKIE IS_SPECIAL_COOKIE 매크로는 스냅인의 IComponent::QueryDataObject 메서드 호출에서 MMC가 전달한 MMC_COOKIE 값이 특별한 쿠키 형식인지 여부를 결정합니다.
IS_SPECIAL_DATAOBJECT 스냅인의 Notify 메서드 호출에서 MMC가 전달한 LPDATAOBJECT가 실제 IDataObject 개체에 대한 포인터 대신 특수 형식의 데이터 개체인지 여부를 확인합니다.
IsTaskpadViewP 참조 IConsole2::IsTaskpadViewPreferred 메서드는 사용되지 않습니다. 항상 S_OK를 반환합니다.
MessageBox 메시지 상자를 표시합니다.
MMCFreeNotifyHandle 속성 시트에서 만든 MMCPropertyChangeNotify 호출에 대한 응답으로 MMC가 스냅인으로 보낸 MMCN_PROPERTY_CHANGE 알림 메시지에 대한 핸들을 해제하기 위해 스냅인에 의해 호출됩니다.
MMCPropertyChangeNotify 스냅인 속성 시트가 항목의 속성이 변경되었음을 IComponent 또는 IComponentData 인터페이스에 알릴 수 있도록 합니다.
MMCPropertyHelp 속성 페이지의 병합된 MMC HTML 도움말 파일에 지정된 HTML 도움말 항목을 표시합니다.

MMCPropPageCallback MMCPropPageCallback 함수는 MFC(Microsoft Foundation Classs) 기반 스냅인에서만 필요합니다. 이 함수는 페이지를 만드는 동안 올바른 모듈 상태를 설정합니다.
ModifyItemState 스냅인을 사용하여 항목의 상태를 수정할 수 있습니다.
ModifyViewStyle IResultData::ModifyViewStyle 메서드를 사용하면 스냅인에서 결과 창의 뷰 스타일을 설정할 수 있습니다.
NewWindow 지정된 범위 항목에 루팅된 새 MDI(다중 문서 인터페이스) 자식 창을 만듭니다.
다음 IEnumTASK::Next 메서드를 사용하면 MMC가 스냅인의 작업 목록에서 다음 작업을 검색할 수 있습니다.
알림 IComponent::Notify 메서드는 사용자가 수행한 작업의 스냅인을 알립니다.
알림 IComponentData::Notify 메서드는 사용자가 수행한 작업의 스냅인을 알립니다.
OnPowerBroadcast OnPowerBroadcast 메서드는 컴퓨터의 전원 관리와 관련된 WM_POWERBROADCAST 알림 메시지를 처리합니다.
PropertiesChanged 속성을 추가, 변경 또는 삭제할 때 호출됩니다.
QueryConsoleVerb IConsoleVerb 인터페이스에 대한 쿼리입니다.
QueryDataObject IComponent::QueryDataObject 메서드는 지정된 쿠키에 대한 컨텍스트 정보를 검색하는 데 사용할 수 있는 데이터 개체를 반환합니다.

QueryDataObject IComponentData::QueryDataObject 메서드는 지정된 쿠키에 대한 컨텍스트 정보를 검색하는 데 사용할 수 있는 데이터 개체를 반환합니다.
QueryDispatch QueryDispatch 메서드는 지정된 항목에 대한 스냅인 IDispatch 인터페이스를 반환합니다.
QueryDispatch QueryDispatch 메서드는 지정된 항목에 대한 스냅인의 IDispatch 인터페이스를 반환합니다.
QueryPagesFor 개체에 페이지가 필요한지 여부를 확인합니다.
QueryPropertyNames QueryPropertyNames 메서드는 스냅인 구성에 사용되는 속성의 이름을 반환합니다.
QueryResultImageList 콘솔에서 제공하는 결과 보기 이미지 목록을 검색합니다. 이 이미지 목록은 스냅인에서 기본 목록 보기를 사용하는 경우에만 사용해야 합니다.
QueryResultView 결과 뷰 개체 IUnknown 인터페이스 포인터에 대해 IConsole을 쿼리합니다.
QueryScopeImageList 콘솔에서 제공하는 범위 창 이미지 목록을 쿼리합니다.
RemovePage IPropertySheetCallback::RemovePage 메서드를 사용하면 스냅인에서 속성 시트에서 페이지를 제거할 수 있습니다.
RenameResultItem RenameResultItem 메서드는 프로그래밍 방식으로 지정된 결과 항목을 이름 바꾸기 모드로 전환한 다음 사용자가 새 이름을 수동으로 입력할 수 있습니다.
RenameScopeItem RenameScopeItem 메서드는 프로그래밍 방식으로 지정된 범위 항목을 이름 바꾸기 모드로 전환합니다. 그런 다음 사용자가 새 이름을 수동으로 입력할 수 있습니다.

재설정 IEnumTASK::Reset 메서드를 사용하면 MMC가 스냅인 작업 목록의 시작 부분으로 열거형을 다시 설정할 수 있습니다.
ResetIdleTimer ResetIdleTimer 메서드는 지정된 전원 관리 유휴 타이머를 다시 설정합니다.
RestoreResultView RestoreResultView 메서드는 결과 뷰를 복원합니다. 이 메서드를 사용하면 스냅인에서 결과 보기의 스냅인별 세부 정보를 복원할 수 있습니다. 자세한 내용은 결과 뷰 복원을 참조하세요.
SelectScopeItem 지정된 범위 항목을 선택합니다.
SetBodyText IMessageView::SetBodyText 메서드를 사용하면 스냅인에서 MMC 메시지 OCX 컨트롤을 사용하여 표시되는 결과 창 메시지의 본문 텍스트를 설정할 수 있습니다.
SetButton IMenuButton::SetButton 메서드를 사용하면 사용자가 메뉴 모음에서 변경된 단추의 텍스트 특성을 설정할 수 있습니다.
SetButtonState IMenuButton::SetButtonState 메서드를 사용하면 사용자가 메뉴 단추의 상태를 변경할 수 있습니다.
SetButtonState 스냅인을 사용하여 단추의 특성을 설정할 수 있습니다.
SetChangeTimeOut IHeaderCtrl2::SetChangeTimeOut은 필터 특성에서 변경이 발생하는 시간과 스냅인의 IComponent::Notify 메서드로 전송되는 MMCN_FILTER_CHANGE 필터 변경 알림 게시 사이의 시간 제한 간격을 설정합니다.
SetColumnConfigData IColumnData::SetColumnConfigData 메서드를 사용하면 스냅인에서 열 집합에 있는 열의 지속형 너비, 순서 및 숨겨진 상태를 설정할 수 있습니다.

SetColumnFilter IHeaderCtrl2::SetColumnFilter는 필터링된 목록의 지정된 열에 대한 필터 값 및 최대 문자 길이를 설정합니다.
SetColumnSortData IColumnData::SetColumnSortData 메서드를 사용하면 스냅인에서 열 집합의 열에 대해 정렬된 열 및 정렬 방향을 설정할 수 있습니다.
SetColumnText 특정 열에 있는 제목의 텍스트를 설정합니다.
SetColumnWidth 특정 열의 너비를 픽셀 단위로 설정합니다.
SetControlbar IExtendControlbar::SetControlbar 메서드는 컨트롤 막대를 연결하거나 분리합니다.
SetDefaultVerb SetDefaultVerb 메서드는 개체에 대한 기본 작업을 설정합니다.
SetDescBarText 결과 보기 창의 설명 표시줄 텍스트를 설정합니다.
SetExecutionState SetExecutionState 메서드는 현재 스레드의 실행 상태를 설정합니다.
SetHeader 이 IComponent 인스턴스에 사용할 헤더 인터페이스를 설정합니다. IComponent 인스턴스에서만 사용됩니다.
SetIcon IMessageView::SetIcon 메서드를 사용하면 스냅인에서 MMC 메시지 OCX 컨트롤을 사용하여 표시되는 결과 창 메시지의 아이콘을 설정할 수 있습니다.
Setitem IConsoleNameSpace2::SetItem 메서드를 사용하면 스냅인에서 단일 범위 보기 항목의 특성을 설정할 수 있습니다.

Setitem IResultData::SetItem 메서드를 사용하면 스냅인에서 결과 창에서 단일 항목을 설정할 수 있습니다.
SetItemCount 가상 목록의 항목 수를 설정합니다.
SetStatusText IConsole2::SetStatusText 메서드를 사용하면 스냅인에서 상태 표시줄의 텍스트를 변경할 수 있습니다. IComponent 인스턴스에서만 사용됩니다.
SetTitleText IMessageView::SetTitleText 메서드를 사용하면 스냅인에서 MMC 메시지 OCX 컨트롤을 사용하여 표시되는 결과 창 메시지의 제목 텍스트를 설정할 수 있습니다.
SetToolbar 이 IComponent 인스턴스에 사용할 도구 모음 인터페이스를 설정합니다. IComponent 인스턴스에서만 사용됩니다.
SetVerbState SetVerbState 메서드를 사용하면 스냅인에서 지정된 동사의 단추 상태를 설정할 수 있습니다.
SetViewMode 스냅인을 사용하여 결과 보기 창에 해당 항목이 표시되는 보기 모드를 설정할 수 있습니다.
표시 IPropertySheetProvider::CreatePropertySheet에 설정된 형식이 속성 시트인 경우 IPropertySheetProvider::Show는 숨겨진 창에 부모로 설정된 속성 시트 프레임을 표시합니다.
ShowContextMenu IContextMenuProvider::ShowContextMenu 메서드는 상황에 맞는 메뉴를 표시합니다.
ShowTopic IDisplayHelp::ShowTopic 메서드는 병합된 MMC HTML 도움말 파일에 지정된 HTML 도움말 항목을 표시합니다.
skip IEnumTASK::Skip 메서드는 스냅인의 작업 목록에 있는 지정된 작업 수를 건너뜁니다. MMC는 이 메서드를 사용하지 않습니다. 완전성을 위해 포함해야 합니다.

Sort 결과 창의 모든 항목을 정렬합니다.
SortItems 가상 결과 목록의 항목을 정렬합니다.
TaskNotify IExtendTaskPad::TaskNotify 메서드를 사용하면 MMC가 작업이 확장될 때 스냅인을 알릴 수 있습니다. 작업 패드가 목록 보기 작업 패드인 경우 MMC는 목록 보기 단추가 확장될 때 IExtendTaskPad::TaskNotify를 호출합니다.
UpdateAllViews 결과 창에 콘텐츠가 변경되면 스냅인에 의해 호출됩니다. 이 메서드는 IComponent 또는 IComponentData에서 호출할 수 있습니다.
Updateitem 결과 창에서 지정된 항목이 다시 그려지도록 합니다.

인터페이스

IColumnData IColumnData 인터페이스는 MMC 1.2에 도입되었습니다.
IComponent IComponent 인터페이스를 사용하면 MMC가 스냅인과 통신할 수 있습니다. IComponentData 인터페이스와 마찬가지로 IComponent는 일반적으로 보기 수준에서 구현되며 결과 창에 표시되는 항목과 밀접하게 연결됩니다.
IComponent2 스냅인으로 구현된 IComponent2 인터페이스는 MMC 2.0에 도입되어 IComponent 인터페이스를 대체합니다.

IComponentData IComponentData 인터페이스를 사용하면 MMC가 스냅인과 통신할 수 있습니다. IComponent 인터페이스와 마찬가지로 IComponentData는 일반적으로 문서 수준에서 구현되며 범위 창에 표시되는 항목(폴더)과 밀접하게 연결됩니다.
IComponentData2 IComponentData2 인터페이스는 IComponentData 인터페이스를 대체합니다.
IConsole 콘솔과의 통신을 사용하도록 설정합니다.
IConsole2 IConsole2 인터페이스는 MMC 1.1에서 도입되었습니다.
IConsole3 IConsole3 인터페이스는 IConsole2 인터페이스를 대체합니다. IConsole3 인터페이스에는 범위 노드를 프로그래밍 방식으로 이름 바꾸기 모드로 배치할 수 있는 IConsole3::RenameScopeItem 메서드가 포함되어 있습니다.
IConsoleNameSpace 스냅인을 사용하여 범위 창에서 동적 하위 컨테이너를 열거할 수 있습니다. 특정 스냅인은 하위 컨테이너로 자격이 있는 항목을 결정합니다.
IConsoleNameSpace2 IConsoleNameSpace2 인터페이스는 MMC 1.1에 도입되었습니다.
IConsolePower IConsolePower 인터페이스는 전원 관리를 지원하는 운영 체제에서 실행 상태 및 유휴 타이머를 제어합니다.
IConsolePowerSink IConsolePowerSink 인터페이스는 전원 관리 메시지를 모니터링하고 응답합니다.
IConsoleVerb IConsoleVerb 인터페이스를 사용하면 스냅인을 통해 잘라내기, 복사, 붙여넣기, 삭제, 속성, 이름 바꾸기, 새로 고침 및 인쇄를 비롯한 표준 동사를 사용할 수 있습니다. 항목을 선택하면 스냅인에서 이러한 동사의 상태를 업데이트할 수 있습니다.

IContextMenuCallback IContextMenuCallback 인터페이스는 상황에 맞는 메뉴에 메뉴 항목을 추가하는 데 사용됩니다.
IContextMenuCallback2 IContextMenuCallback2 인터페이스는 상황에 맞는 메뉴에 메뉴 항목을 추가하는 데 사용됩니다. 이 인터페이스는 IContextMenuCallback을 대체합니다.
IContextMenuProvider IContextMenuProvider 인터페이스는 해당 메뉴에 항목을 추가하고 확장에서 해당 메뉴를 확장하고 결과 상황에 맞는 메뉴를 표시하기 위해 새 상황에 맞는 메뉴를 만드는 메서드를 구현합니다.
IControlbar IControlbar 인터페이스는 도구 모음 및 기타 컨트롤을 만드는 방법을 제공합니다.
IDisplayHelp IDisplayHelp 인터페이스는 MMC 버전 1.1에 도입되었습니다.
IEnumTASK IEnumTASK 인터페이스는 MMC 1.1에서 도입되었습니다.
IExtendContextMenu IExtendContextMenu 인터페이스를 사용하면 스냅인에서 기존 상황에 맞는 메뉴에 항목을 추가할 수 있습니다.
IExtendControlbar IExtendControlbar 인터페이스를 사용하면 확장에서 콘솔에 컨트롤 막대를 추가할 수 있습니다. 이렇게 하면 도구 모음 또는 기타 사용자 인터페이스 개선 사항을 추가하여 스냅인의 기능과 모양을 개선할 수 있습니다.
IExtendPropertySheet 스냅인 구성 요소가 항목의 속성 시트에 페이지를 추가할 수 있도록 합니다.
IExtendPropertySheet2 IExtendPropertySheet2 인터페이스는 MMC 1.1에 도입되었습니다.
IExtendTaskPad IExtendTaskPad 인터페이스는 MMC 1.1에 도입되었습니다.

IExtendView IExtendView 인터페이스는 확장 보기에 대한 정보를 제공합니다.
IHeaderCtrl 열을 조작할 수 있도록 설정하고 콘솔의 결과 보기 창에 표시할 정보의 종류를 나타냅니다.
IHeaderCtrl2 IHeaderCtrl2 인터페이스는 MMC 1.2에 도입되었습니다.
IImageList IImageList 인터페이스를 사용하면 사용자가 콘솔의 결과 또는 범위 창에서 항목의 아이콘으로 사용할 이미지를 삽입할 수 있습니다.
IMenuButton IMenuButton 인터페이스를 사용하면 사용자가 스냅인에 대한 메뉴 단추를 추가하고 관리할 수 있습니다.
IMessageView IMessageView 인터페이스는 MMC 1.2에 도입되었습니다.
IMMCVersionInfo IMMCVersionInfo 인터페이스는 설치된 MMC 애플리케이션에 대한 버전 정보를 제공합니다.
INodeProperties INodeProperties 인터페이스는 노드에 대한 텍스트 전용 속성을 검색합니다.
IPropertySheetCallback IPropertySheetCallback 인터페이스는 속성 시트에 속성 페이지를 추가하기 위해 스냅인에서 사용하는 COM 기반 인터페이스입니다.
IPropertySheetProvider IPropertySheetProvider 인터페이스는 Windows 속성 시트를 COM 개체로 구현합니다.
IRequiredExtensions IRequiredExtensions 인터페이스는 MMC 1.1에 도입되었습니다.

IResultData IResultData 인터페이스를 사용하면 사용자가 결과 보기 창과 연결된 항목을 추가, 제거, 찾기 및 수정할 수 있습니다. 또한 결과 보기 창의 보기 스타일을 조작할 수 있습니다.
IResultData2 IResultData2 인터페이스는 IResultData 인터페이스를 대체합니다. IResultData2 인터페이스에는 결과 노드를 프로그래밍 방식으로 이름 바꾸기 모드로 전환할 수 있는 IResultData2::RenameResultItem 메서드가 포함되어 있습니다.
IResultDataCompare 기본 스냅인에서 결과 창에 정렬된 순서로 표시되는 결과 항목을 비교할 수 있습니다.
IResultDataCompareEx 기본 스냅인에서 결과 창에 정렬된 순서로 표시되는 범위와 결과 항목을 모두 비교할 수 있습니다.
IResultOwnerData IResultOwnerData 인터페이스는 LVS_OWNERDATA 스타일이 설정된 목록 보기 컨트롤인 가상 목록의 사용을 지원합니다.
ISnapinAbout ISnapinAbout 인터페이스를 사용하면 콘솔이 스냅인에서 저작권 및 버전 정보를 가져올 수 있습니다. 또한 콘솔은 이 인터페이스를 사용하여 스냅인에서 정적 폴더에 대한 이미지를 가져옵니다.
ISnapinHelp 스냅인에서 HTML 도움말 지원을 추가할 수 있습니다. (ISnapinHelp)
ISnapinHelp2 스냅인에서 HTML 도움말 지원을 추가할 수 있습니다. (ISnapinHelp2)
ISnapinProperties ISnapinProperties 인터페이스를 사용하면 스냅인에서 스냅인의 속성을 초기화하고 속성이 추가, 변경 또는 삭제될 때 알림을 받을 수 있습니다.
ISnapinPropertiesCallback ISnapinPropertiesCallback 인터페이스는 스냅인에 대한 속성 이름을 추가합니다. 이 인터페이스는 스냅인을 위해 MMC에 의해 구현됩니다.

IStringTable IStringTable 인터페이스는 MMC 1.1에 도입되었습니다.
IToolbar IToolbar 인터페이스는 새 도구 모음을 만들고, 항목을 추가하고, 도구 모음을 확장하고, 결과 새 도구 모음을 표시하는 데 사용됩니다. 각 도구 모음은 컨트롤 바 내의 자체 밴드에 만들어집니다.
IViewExtensionCallback IViewExtensionCallback 인터페이스는 결과 창에 보기를 추가하는 데 사용됩니다.

구조체

CONTEXTMENUITEM CONTEXTMENUITEM 구조체는 IContextMenuCallback::AddItem 메서드 또는 IContextMenuProvider::AddItem 메서드(IContextMenuCallback에서 상속됨)에 전달되어 새 메뉴 항목, 하위 메뉴 또는 삽입 지점을 정의합니다.
CONTEXTMENUITEM2 CONTEXTMENUITEM2 구조체는 MMC 2.0에서 도입되었습니다.
MENUBUTTONDATA MENUBUTTONDATA 구조에는 도구 모음에서 단추를 만드는 데 사용되는 값이 포함되어 있습니다.
MMC_COLUMN_DATA MMC_COLUMN_DATA 구조는 MMC 1.2에 도입되었습니다.
MMC_COLUMN_SET_DATA MMC_COLUMN_SET_DATA 구조는 MMC 1.2에 도입되었습니다.
MMC_EXPANDSYNC_STRUCT MMC_EXPANDSYNC_STRUCT 구조는 MMC 1.1에 도입되었습니다.
MMC_EXT_VIEW_DATA MMC_EXT_VIEW_DATA 구조는 MMC 2.0에 도입되었습니다.

MMC_FILTERDATA MMC_FILTERDATA 구조는 MMC 1.2에 도입되었습니다.
MMC_LISTPAD_INFO MMC_LISTPAD_INFO 구조는 MMC 1.1에 도입되었습니다.
MMC_RESTORE_VIEW MMC_RESTORE_VIEW 구조는 MMC 1.1에 도입되었습니다.
MMC_SNAPIN_PROPERTY MMC_SNAPIN_PROPERTY 구조는 MMC 2.0에서 도입되었습니다.
MMC_SORT_DATA 열 집합에 있는 단일 열의 열 정렬 데이터를 포함합니다.
MMC_SORT_SET_DATA 정렬 데이터가 영구적으로 저장되는 목록 보기 열 집합을 설정 및 검색하는 데 사용됩니다.
MMC_TASK MMC_TASK 구조는 MMC 1.1에 도입되었습니다.
MMC_TASK_DISPLAY_BITMAP MMC_TASK_DISPLAY_BITMAP 구조는 MMC 1.1에 도입되었습니다.
MMC_TASK_DISPLAY_OBJECT 작업 패드에 작업 또는 배경을 표시하기 위해 해당 이미지를 사용하는 데 필요한 이미지 유형과 모든 데이터를 지정합니다.
MMC_TASK_DISPLAY_SYMBOL MMC_TASK_DISPLAY_SYMBOL 구조는 MMC 1.1에 도입되었습니다.
MMC_VISIBLE_COLUMNS MMC에서 MMCN_COLUMNS_CHANGED 알림과 함께 열 집합의 열이 표시되는 스냅인을 알리는 데 사용됩니다.

MMCBUTTON MMCBUTTON 구조에는 도구 모음에서 단추를 만드는 데 사용되는 값이 포함되어 있습니다. 이 구조는 일반적인 컨트롤과 관련된 SDK(Platform Software Development Kit) 항목에서 설명하는 TBBUTTON 구조와 유사합니다.
RDCOMPARE RDCOMPARE 구조는 MMC 1.2에 도입되었습니다.
RDITEMHDR RDITEMHDR 구조체는 MMC 1.2에 도입되었습니다.
RESULT_VIEW_TYPE_INFO RESULT_VIEW_TYPE_INFO 구조는 MMC 2.0에 도입되었습니다.
RESULTDATAITEM RESULTDATAITEM 구조체는 스냅인의 결과 창에서 결과 항목의 특성을 지정하거나 받습니다.
RESULTFINDINFO IResultOwnerData::FindItem 메서드에서 결과 창의 가상 목록에서 키보드 탐색을 지원하는 데 사용됩니다.
SColumnSetID SColumnSetID 구조체는 MMC 1.2에 도입되었습니다.
SCOPEDATAITEM SCOPEDATAITEM 구조는 범위 창에 삽입할 항목을 지정합니다.
SMMCDataObjects SMMCDataObjects 구조체는 CCF_MULTI_SELECT_SNAPINS 클립보드 형식의 데이터 형식을 정의합니다.
SMMCObjectTypes SMMCDynamicExtensions 구조는 MMC 1.1에 도입되었습니다.
SNodeID SNodeID 구조체는 MMC 1.1에서 도입되었으며 MMC 1.2의 SNodeID2 구조체로 대체됩니다.

SNodeID2 SNodeID2 구조체는 MMC 1.2에 도입되어 SNodeID 구조를 대체합니다.

Microsoft Media Foundation

아티클2024. 07. 19.

Microsoft Media Foundation 기술 개요입니다.

Microsoft Media Foundation을 개발하려면 다음 헤더가 필요합니다.

- codecapi.h
- dxva2api.h
- dxva9typ.h
- dxvahd.h
- icodecapi.h
- ksopmapi.h
- mfapi.h
- mfcaptureengine.h
- mfcontentdecryptionmodule.h
- mfd3d12.h
- mfidl.h
- mfmediacapture.h
- mfmediaengine.h
- mfmp2dlna.h
- mfobjects.h
- mfreadwrite.h
- mfsharingengine.h
- mfspatialaudio.h
- mftransform.h
- mfvirtualcamera.h
- opmapi.h
- wmcodecdsp.h
- wmcontainer.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Microsoft Media Foundation

열거형

테이블 확장

_MFT_DRAIN_TYPE 지원되지 않습니다. (_MFT_DRAIN_TYPE)
_MFT_ENUM_FLAG MFT(Media Foundation 변환)를 등록하고 열거하기 위한 플래그를 포함합니다.

_MFT_INPUT_DATA_BUFFER_FLAGS IMFTransform::P rocessInput 메서드에 대한 플래그를 정의합니다. 현재 플래그가 정의되어 있지 않습니다.
_MFT_INPUT_STATUS_FLAGS MFT(Media Foundation 변환)에서 입력 스트림의 상태를 나타냅니다.
_MFT_INPUT_STREAM_INFO_FLAGS MFT(Media Foundation 변환)의 입력 스트림에 대해 설명합니다.
_MFT_OUTPUT_DATA_BUFFER_FLAGS IMFTransform::P rocessOutput 메서드에 대한 플래그를 정의합니다.
_MFT_OUTPUT_STATUS_FLAGS MFT(Media Foundation 변환)에서 출력 데이터를 생성할 수 있는지 여부를 나타냅니다.
_MFT_OUTPUT_STREAM_INFO_FLAGS MFT(Media Foundation 변환)의 출력 스트림에 대해 설명합니다.
_MFT_PROCESS_OUTPUT_FLAGS MFT(Media Foundation 변환)에서 출력 샘플을 처리하기 위한 플래그를 정의합니다.
_MFT_PROCESS_OUTPUT_STATUS IMFTransform::P rocessOutput에 대한 호출의 상태를 나타냅니다.
_MFT_SET_TYPE_FLAGS MFT(Media Foundation 변환)에서 미디어 형식을 설정하거나 테스트하는 플래그를 정의합니다.
AEC_SYSTEM_MODE 음성 캡처 DSP의 처리 모드를 지정합니다. 이 열거형은 MFPKEY_WMAAECMA_SYSTEM_MODE 속성과 함께 사용됩니다.
AEC_VAD_MODE 음성 캡처 DSP에 대한 VAD(음성 활동 감지) 유형을 지정합니다. 이 열거형은 MFPKEY_WMAAECMA_FEATR_VAD 속성과 함께 사용됩니다.
ASF_SELECTION_STATUS ASF 스트림에 대한 선택 옵션을 정의합니다.
ASF_STATUSFLAGS IMFASFSplitter::GetNextSample 메서드의 상태 조건을 정의합니다.
D3D11_AUTHENTICATED_CHANNEL_TYPE

Microsoft Direct3D 인증 채널의 유형을 지정합니다.
D3D11_AUTHENTICATED_PROCESS_IDENTIFIER_TYPE D3D11_AUTHENTICATED_QUERY_RESTRICTED_SHARED_RESOURCE_PROCESS_OUTPUT 구조체에서 식별되는 프로세스의 형식을 지정합니다.
D3D11_BUS_TYPE 그래픽 어댑터에서 사용하는 I/O 버스의 유형을 지정합니다.
D3D11_CONTENT_PROTECTION_CAPS 콘텐츠 보호 기능을 설명하는 플래그를 포함합니다.
D3D11_CRYPTO_SESSION_STATUS ID3D11CryptoSession 인터페이스의 상태를 나타냅니다.
D3D11_FEATURE_VIDEO 쿼리할 Direct3D 11 비디오 기능 또는 기능 집합을 지정합니다.
D3D11_VDOV_DIMENSION 비디오 디코딩 출력 뷰에서 사용되는 리소스에 액세스하는 방법을 지정합니다.
D3D11_VIDEO_DECODER_BUFFER_TYPE 디코딩을 위해 압축된 버퍼의 형식을 지정합니다.
D3D11_VIDEO_DECODER_CAPS 비디오 디코더의 기능을 지정합니다.
D3D11_VIDEO_FRAME_FORMAT 비디오 스트림이 인터레이스되는 방법을 설명합니다. (D3D11_VIDEO_FRAME_FORMAT)
D3D11_VIDEO_PROCESSOR_ALPHA_FILL_MODE 비디오 처리를 위한 알파 채우기 모드를 지정합니다. (D3D11_VIDEO_PROCESSOR_ALPHA_FILL_MODE)
D3D11_VIDEO_PROCESSOR_AUTO_STREAM_CAPS 비디오 프로세서의 자동 이미지 처리 기능을 지정합니다.
D3D11_VIDEO_PROCESSOR_BEHAVIOR_HINTS 비디오 처리 작업을 수행하기 위한 가장 효율적인 방법을 나타내는 플래그를 지정합니다.
D3D11_VIDEO_PROCESSOR_DEVICE_CAPS Microsoft Direct3D 11 비디오 프로세서에 대한 비디오 처리 기능을 정의합니다.
D3D11_VIDEO_PROCESSOR_FEATURE_CAPS

Microsoft Direct3D 11 비디오 프로세서가 지원할 수 있는 기능을 정의합니다.
D3D11_VIDEO_PROCESSOR_FILTER 비디오 프로세서 필터를 식별합니다.
D3D11_VIDEO_PROCESSOR_FILTER_CAPS Microsoft Direct3D 11 비디오 프로세서에 대한 이미지 필터 기능을 정의합니다.
D3D11_VIDEO_PROCESSOR_FORMAT_CAPS Microsoft Direct3D 11 비디오 프로세서의 입력 형식과 관련된 기능을 정의합니다.
D3D11_VIDEO_PROCESSOR_FORMAT_SUPPORT 비디오 처리에 비디오 형식을 사용할 수 있는 방법을 지정합니다.
D3D11_VIDEO_PROCESSOR_ITELECINE_CAPS 비디오 프로세서의 IVTC(역 텔레신) 기능을 지정합니다.
D3D11_VIDEO_PROCESSOR_NOMINAL_RANGE YUV 데이터의 광도 범위에 대한 값을 지정합니다.
D3D11_VIDEO_PROCESSOR_OUTPUT_RATE 비디오 프로세서가 입력 스트림에서 출력 프레임을 생성하는 속도를 지정합니다.
D3D11_VIDEO_PROCESSOR_PROCESSOR_CAPS 디인터레이싱, IVTC(역 텔레신) 및 프레임 속도 변환과 관련된 비디오 처리 기능을 지정합니다.
D3D11_VIDEO_PROCESSOR_ROTATION 비디오 회전 상태를 지정합니다.
D3D11_VIDEO_PROCESSOR_STEREO_CAPS Microsoft Direct3D 11 비디오 프로세서의 스테레오 3D 기능을 정의합니다.
D3D11_VIDEO_PROCESSOR_STEREO_FLIP_MODE 스테레오 3D 비디오의 경우 프레임 0 또는 프레임 1의 데이터가 가로 또는 세로로 대칭 이동되는지 여부를 지정합니다.
D3D11_VIDEO_PROCESSOR_STEREO_FORMAT 스테레오 3D 비디오 프레임의 메모리 레이아웃을 지정합니다.
D3D11_VIDEO_USAGE 비디오 프로세서의 용도를 지정합니다.
D3D11_VPIV_DIMENSION

비디오 프로세서 입력 보기에서 사용되는 리소스에 액세스하는 방법을 지정합니다.
D3D11_VPOV_DIMENSION 비디오 프로세서 출력 뷰에서 사용되는 리소스에 액세스하는 방법을 지정합니다.
D3D12_BITSTREAM_ENCRYPTION_TYPE 비트 스트림 암호화 유형을 지정합니다.
D3D12_FEATURE_VIDEO 쿼리할 Direct3D 12 비디오 기능 또는 기능 집합을 지정합니다.
D3D12_VIDEO_DECODE_ARGUMENT_TYPE D3D12_VIDEO_DECODE_FRAME_ARGUMENT 인수 형식을 지정합니다.
D3D12_VIDEO_DECODE_CONFIGURATION_FLAGS 비디오 디코딩에 대한 구성을 지정합니다.
D3D12_VIDEO_DECODE_CONVERSION_SUPPORT_FLAGS 비디오 디코딩 변환 작업이 지원되는지 여부를 지정합니다.
D3D12_VIDEO_DECODE_HISTOGRAM_COMPONENT 구성 요소별 히스토그램 정보의 배열에 대한 인덱스를 지정합니다. (D3D12_VIDEO_DECODE_HISTOGRAM_COMPONENT)
D3D12_VIDEO_DECODE_HISTOGRAM_COMPONENT_FLAGS 비디오 디코딩 히스토그램과 함께 사용되는 구성 요소의 하위 집합을 나타내는 플래그입니다. (D3D12_VIDEO_DECODE_HISTOGRAM_COMPONENT_FLAGS)
D3D12_VIDEO_DECODE_STATUS 비디오 디코딩 작업의 상태를 지정합니다.
D3D12_VIDEO_DECODE_SUPPORT_FLAGS 비디오 디코딩 작업이 지원되는지 여부를 지정합니다.
D3D12_VIDEO_DECODE_TIER 애플리케이션 정의 텍스처 및 버퍼의 필수 형식을 결정하는 하드웨어 비디오 디코더의 디코딩 계층을 지정합니다.
D3D12_VIDEO_ENCODER_CODEC Direct3D 12 비디오 인코딩에 대한 코덱을 지정합니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_H264_DIRECT_MODES H.264 비디오 인코딩에 대한 직접 모드를 지정합니다.

D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_H264_FLAGS H.264 비디오 인코딩에 대한 구성 플래그를 지정합니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_H264_SLICES_DEBLOCKING_MODE_FLAGS D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_H264_SLICES_DEBLOCKING_MODES 값의 비트 OR 조합을 허용하는 플래그 열거형입니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_H264_SLICES_DEBLOCKING_MODES H.264 사양의 *disable_deblocking_filter_idc* 구문에 정의된 조각 디버킹 모드를 지정합니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_HEVC_CUSIZE HEVC에 대한 루마 코딩 블록 크기에 사용할 수 있는 값을 지정합니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_HEVC_FLAGS HEVC 비디오 인코딩에 대한 구성 플래그를 지정합니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_HEVC_TUSIZE HEVC에 대한 루마 변환 블록 크기에 사용할 수 있는 값을 지정합니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_SUPPORT_H264_FLAGS H.264 비디오 인코딩에 대한 구성 지원 플래그를 지정합니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_SUPPORT_HEVC_FLAGS HEVC 비디오 인코딩에 대한 구성 지원 플래그를 지정합니다.
D3D12_VIDEO_ENCODER_ENCODE_ERROR_FLAGS ID3D12VideoEncodeCommandList2::EncodeFrame 작업 중에 발생한 오류를 지정합니다.
D3D12_VIDEO_ENCODER_FLAGS 비디오 인코더 만들기에 대한 플래그를 지정합니다.
D3D12_VIDEO_ENCODER_FRAME_SUBREGION_LAYOUT_MODE 비디오 인코더 프레임 하위 구성 레이아웃 모드를 지정합니다.
D3D12_VIDEO_ENCODER_FRAME_TYPE_H264 H.264 비디오 프레임의 형식을 지정합니다.
D3D12_VIDEO_ENCODER_FRAME_TYPE_HEVC HEVC 비디오 프레임의 형식을 지정합니다.
D3D12_VIDEO_ENCODER_HEAP_FLAGS 비디오 인코딩에 대한 힙 옵션을 지정합니다.

D3D12_VIDEO_ENCODER_INTRA_REFRESH_MODE 비디오 인코더 인트라 새로 고침 모드를 지정합니다.
D3D12_VIDEO_ENCODER_LEVELS_H264 H.264 인코딩에 대한 인코더 수준을 지정합니다.
D3D12_VIDEO_ENCODER_LEVELS_HEVC HEVC(고효율 비디오 코딩) 인코딩에 대한 인코더 수준을 지정합니다.
D3D12_VIDEO_ENCODER_MOTION_ESTIMATION_PRECISION_MODE 비디오 인코딩에 대한 동작 추정 정밀도 모드를 지정합니다.
D3D12_VIDEO_ENCODER_PICTURE_CONTROL_CODEC_DATA_H264_FLAGS H.264 관련 그림 컨트롤 속성에 대한 플래그를 지정합니다.
D3D12_VIDEO_ENCODER_PICTURE_CONTROL_CODEC_DATA_HEVC_FLAGS HEVC 관련 그림 컨트롤 속성에 대한 플래그를 지정합니다.
D3D12_VIDEO_ENCODER_PICTURE_CONTROL_FLAGS 비디오 인코더 그림 컨트롤 플래그를 지정합니다.
D3D12_VIDEO_ENCODER_PROFILE_H264 H.264 인코딩에 대한 인코더 프로필을 지정합니다.
D3D12_VIDEO_ENCODER_PROFILE_HEVC HEVC(고효율 비디오 코딩) 인코딩에 대한 인코더 프로필을 지정합니다.
D3D12_VIDEO_ENCODER_RATE_CONTROL_FLAGS 3D12_VIDEO_ENCODER_RATE_CONTROL 구조체의 플래그를 지정합니다.
D3D12_VIDEO_ENCODER_RATE_CONTROL_MODE 비디오 인코더 속도 제어 모드를 지정합니다.
D3D12_VIDEO_ENCODER_SEQUENCE_CONTROL_FLAGS 비디오 인코더 시퀀스 컨트롤 속성에 대한 플래그를 지정합니다.
D3D12_VIDEO_ENCODER_SUPPORT_FLAGS 비디오 인코더 기능에 대한 플래그를 지정합니다.
D3D12_VIDEO_ENCODER_TIER_HEVC HEVC(고효율 비디오 코딩) 인코딩에 대한 인코더 계층을 지정합니다.

D3D12_VIDEO_ENCODER_VALIDATION_FLAGS 지원되지 않는 인코더 기능을 지정하는 플래그입니다.
D3D12_VIDEO_EXTENSION_COMMAND_PARAMETER_FLAGS 연결된 비디오 확장 명령 매개 변수의 사용을 지정합니다.
D3D12_VIDEO_EXTENSION_COMMAND_PARAMETER_STAGE 비디오 확장 명령에 대한 매개 변수 단계를 지정합니다.
D3D12_VIDEO_EXTENSION_COMMAND_PARAMETER_TYPE 비디오 확장 명령에 대한 매개 변수 유형을 지정합니다.
D3D12_VIDEO_FIELD_TYPE 비디오 프레임이 인터레이스되는 방법을 지정합니다.
D3D12_VIDEO_FRAME_CODED_INTERLACE_TYPE 코딩된 비디오 프레임의 인터레이스 형식을 지정합니다.
D3D12_VIDEO_FRAME_STEREO_FORMAT 스테레오 3D 비디오 프레임의 메모리 레이아웃을 정의합니다.
D3D12_VIDEO_MOTION_ESTIMATOR_SEARCH_BLOCK_SIZE 비디오 동작 추정에 지원되는 검색 블록 크기를 정의합니다.
D3D12_VIDEO_MOTION_ESTIMATOR_SEARCH_BLOCK_SIZE_FLAGS 비디오 인코더가 지원할 수 있는 동작 예측 검색 블록 크기를 지정합니다.
D3D12_VIDEO_MOTION_ESTIMATOR_VECTOR_PRECISION 비디오 동작 추정에 대한 벡터 정밀도 값을 정의합니다.
D3D12_VIDEO_MOTION_ESTIMATOR_VECTOR_PRECISION_FLAGS 비디오 인코더가 지원하는 동작 예측 벡터 정밀도를 지정합니다.
D3D12_VIDEO_PROCESS_ALPHA_FILL_MODE 비디오 처리를 위한 알파 채우기 모드를 지정합니다. (D3D12_VIDEO_PROCESS_ALPHA_FILL_MODE)
D3D12_VIDEO_PROCESS_AUTO_PROCESSING_FLAGS 비디오 프로세서에서 지원할 수 있는 자동 처리 기능을 지정합니다.
D3D12_VIDEO_PROCESS_DEINTERLACE_FLAGS 디인터레이싱 비디오 프로세서 기능을 지정합니다.

D3D12_VIDEO_PROCESS_FEATURE_FLAGS 비디오 프로세서에서 지원할 수 있는 기능을 지정합니다.
D3D12_VIDEO_PROCESS_FILTER_FLAGS D3D12_VIDEO_PROCESS_FILTER 열거형에서 정의한 이미지 필터에 대한 지원을 지정합니다.
D3D12_VIDEO_PROCESS_INPUT_STREAM_FLAGS 비디오 처리 입력 스트림에 대한 플래그를 지정합니다.
D3D12_VIDEO_PROCESS_ORIENTATION 비디오 프로세서에서 수행할 방향 작업을 지정합니다.
D3D12_VIDEO_PROCESS_SUPPORT_FLAGS 비디오 형식 및 색 영역 변환 작업이 지원되는지 여부를 지정합니다.
D3D12_VIDEO_PROTECTED_RESOURCE_SUPPORT_FLAGS 비디오 작업에서 보호된 리소스에 대한 지원을 지정합니다.
D3D12_VIDEO_SCALE_SUPPORT_FLAGS 비디오 스케일러의 크기 조정 기능을 지정합니다.
DXVA2_NominalRange 색 데이터를 정규화된 [0...1] 범위에 매핑하는 방법을 설명합니다.
DXVA2_SampleFormat 비디오 샘플의 내용을 설명합니다. 이러한 플래그는 DXVA2_ExtendedFormat 구조체에서 사용됩니다.
DXVA2_SurfaceType DXVA(DirectX Video Acceleration)의 DirectX 표면 형식에 대해 설명합니다.
DXVA2_VideoChromaSubSampling YUV 비디오 프레임에서 루마 샘플을 기준으로 크로마 값을 배치하는 방법을 설명합니다.
DXVA2_VideoLighting 비디오 콘텐츠를 보기 위한 의도된 조명 조건에 대해 설명합니다.
DXVA2_VideoPrimaries 비디오 원본의 색 주체를 지정합니다. (DXVA2_VideoPrimaries)
DXVA2_VideoTransferFunction 선형 RGB에서 비선형 RGB(R'G'B')로 변환 함수를 지정합니다. (DXVA2_VideoTransferFunction)

DXVA2_VideoTransferMatrix Y'PbPr(구성 요소 비디오)과 스튜디오 R'G'B' 간의 변환 행렬에 대해 설명합니다. (DXVA2_VideoTransferMatrix)
DXVAHD_ALPHA_FILL_MODE Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 블릿 작업에 대해 출력 알파 값을 계산하는 방법을 지정합니다.
DXVAHD_BLT_STATE Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용하는 경우 blit 작업에 대한 상태 매개 변수를 지정합니다.
DXVAHD_DEVICE_CAPS Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스에 대한 비디오 처리 기능을 정의합니다.
DXVAHD_DEVICE_TYPE Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스의 유형을 지정합니다.
DXVAHD_DEVICE_USAGE Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스에 사용할 용도를 지정합니다.
DXVAHD_FEATURE_CAPS Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스에서 지원할 수 있는 기능을 정의합니다.
DXVAHD_FILTER Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 처리에 대한 이미지 필터를 지정합니다.
DXVAHD_FILTER_CAPS Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스에 대한 이미지 조정 및 필터링과 관련된 기능을 정의합니다.
DXVAHD_FRAME_FORMAT 비디오 스트림이 인터레이스되는 방법을 설명합니다. (DXVAHD_FRAME_FORMAT)
DXVAHD_INPUT_FORMAT_CAPS Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스의 입력 형식과 관련된 기능을 정의합니다.
DXVAHD_ITELECINE_CAPS Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서의 IVTC(역 텔레신) 기능을 지정합니다.
DXVAHD_OUTPUT_RATE Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용하는 경우 입력 스트림의 출력 프레임 속도를 지정합니다.
DXVAHD_PROCESSOR_CAPS Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서의 처리 기능을 지정합니다.

DXVAHD_STREAM_STATE Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서에 대한 입력 스트림의 상태 매개 변수를 지정합니다.
DXVAHD_SURFACE_TYPE Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스에서 만든 비디오 화면 유형을 지정합니다.
eAVDecDDStereoDownMixMode Dolby Digital 오디오 디코더의 스테레오 다운믹스 모드를 지정합니다.
eAVEncAdaptiveMode 비디오 인코더에 대한 적응형 인코딩 모드를 지정합니다.
eAVEncH264PictureType 비디오 인코더에서 출력하는 그림의 형식을 지정합니다.
eAVEncH264VLevel H.264 인코딩 수준을 지정합니다.
eAVEncH264VProfile H.264 비디오 프로필을 지정합니다.
eAVEncVideoContentType 비디오 콘텐츠의 형식을 지정합니다.
MF_ATTRIBUTE_SERIALIZE_OPTIONS 특성 저장소를 직렬화 및 역직렬화하기 위한 플래그를 정의합니다.
MF_ATTRIBUTE_TYPE 키/값 쌍의 데이터 형식을 정의합니다.
MF_ATTRIBUTES_MATCH_TYPE 두 개체의 특성을 비교하는 방법을 지정합니다.
MF_CAMERA_CONTROL_CONFIGURATION_TYPE 카메라 컨트롤의 구성 유형을 지정합니다.
MF_CAPTURE_ENGINE_DEVICE_TYPE 캡처 디바이스의 유형을 지정합니다.
MF_CAPTURE_ENGINE_SINK_TYPE 캡처 싱크의 형식을 지정합니다.

MF_CAPTURE_ENGINE_STREAM_CATEGORY 원본 스트림 범주의 값을 정의합니다.
MF_CONNECT_METHOD 토폴로지 로더가 토폴로지 노드를 연결하는 방법을 지정합니다.
MF_CROSS_ORIGIN_POLICY HTML5 미디어 요소에서 사용하는 W3C CORS(원본 간 설정) 특성에 매핑됩니다.
MF_FILE_ACCESSMODE 파일을 열기 위해 요청된 액세스 모드를 지정합니다.
MF_FILE_FLAGS 파일을 열 때의 동작을 지정합니다.
MF_FILE_OPENMODE 파일을 열거나 만드는 방법을 지정합니다.
MF_MEDIA_ENGINE_CANPLAY 미디어 엔진이 지정된 형식의 미디어 리소스를 재생할 수 있는 가능성을 지정합니다.
MF_MEDIA_ENGINE_CREATEFLAGS IMFMediaEngineClassFactory::CreateInstance 메서드에 대한 플래그를 포함합니다.
MF_MEDIA_ENGINE_ERR 미디어 엔진에 대한 오류 상태 코드를 정의합니다.
MF_MEDIA_ENGINE_EVENT 미디어 엔진에 대한 이벤트 코드를 정의합니다.
MF_MEDIA_ENGINE_EXTENSION_TYPE 미디어 엔진 확장 유형을 지정합니다.
MF_MEDIA_ENGINE_FRAME_PROTECTION_FLAGS 비디오 프레임에 대한 콘텐츠 보호 요구 사항을 지정합니다.
MF_MEDIA_ENGINE_KEYERR 미디어 엔진에 대한 미디어 키 오류 코드를 정의합니다.
MF_MEDIA_ENGINE_NETWORK 미디어 엔진에 대한 네트워크 상태 코드를 정의합니다.

MF_MEDIA_ENGINE_OPM_STATUS OPM(출력 보호 관리자)의 상태를 정의합니다.
MF_MEDIA_ENGINE_PRELOAD 미디어 엔진에 대한 미리 로드 힌트를 정의합니다.
MF_MEDIA_ENGINE_PROTECTION_FLAGS 미디어 엔진이 보호된 콘텐츠를 재생할지 여부와 미디어 엔진이 PMP(보호된 미디어 경로)를 사용할지 여부를 지정하는 플래그를 포함합니다.
MF_MEDIA_ENGINE_READY 미디어 엔진에 대한 준비 상태 값을 정의합니다.
MF_MEDIA_ENGINE_S3D_PACKING_MODE 압축된 3D 비디오 프레임의 레이아웃을 지정합니다.
MF_MEDIA_ENGINE_SEEK_MODE 미디어 엔진 검색 모드에 대한 값을 정의합니다.
MF_MEDIA_ENGINE_STATISTIC 미디어 엔진이 재생 중에 추적하는 통계를 식별합니다.
MF_MEDIA_ENGINE_STREAMTYPE_FAILED 실패한 미디어 스트림의 종류를 식별합니다.
MF_MEDIAKEY_STATUS CDM(콘텐츠 암호 해독 모듈) 세션 키의 상태를 지정합니다.
MF_MEDIAKEYSESSION_MESSAGETYPE CDM(콘텐츠 암호 해독 모듈) 메시지의 형식을 지정합니다.
MF_MEDIAKEYSESSION_TYPE IMFContentDecryptionModuleSession 개체로 표시되는 CDM(Content Decryption Module) 세션의 형식을 지정합니다.
MF_MSE_ERROR 미디어 원본 확장의 다양한 오류 상태를 정의합니다.
MF_MSE_READY 미디어 원본 확장의 다양한 준비 상태를 정의합니다.
MF_MT_D3D_RESOURCE_VERSION_ENUM

미디어 형식과 연결된 스트림에 사용되는 리소스의 Direct3D 버전을 지정합니다.
MF_OBJECT_TYPE 원본 확인자에서 만든 개체 형식을 정의합니다.
MF_OPM_ACP_PROTECTION_LEVEL MFPROTECTION_ACP 대한 보호 수준을 정의합니다.
MF_OPM_CGMSA_PROTECTION_LEVEL MFPROTECTION_CGMSA 대한 보호 수준을 정의합니다.
MF_PLUGIN_CONTROL_POLICY IMFPluginControl2::SetPolicy 메서드에 대한 정책 설정을 정의합니다.
MF_Plugin_Type IMFPluginControl 인터페이스의 개체 형식을 지정합니다.
MF_QUALITY_ADVISE_FLAGS IMFQualityAdvise2::NotifyQualityEvent 메서드에 대한 플래그를 포함합니다.
MF_QUALITY_DROP_MODE 파이프라인 구성 요소가 샘플을 삭제하는 방법을 적극적으로 지정합니다.
MF_QUALITY_LEVEL 파이프라인 구성 요소의 품질 수준을 지정합니다.
MF_SHARING_ENGINE_EVENT 공유 엔진에서 보내는 이벤트를 정의합니다.
MF_SOURCE_READER_CONTROL_FLAG IMFSourceReader::ReadSample 메서드에 대한 플래그를 포함합니다.
MF_SOURCE_READER_FLAG IMFSourceReader::ReadSample 메서드의 상태를 나타내는 플래그를 포함합니다.
MF_TIMED_TEXT_ALIGNMENT 부모 블록 요소에서 텍스트가 정렬되는 방식을 지정합니다.
MF_TIMED_TEXT_CUE_EVENT 시간 제한 텍스트 큐 이벤트의 형식을 지정합니다.
MF_TIMED_TEXT_DECORATION 텍스트가 데코레이팅되는 방식(밑줄 등)을 지정합니다.

MF_TIMED_TEXT_DISPLAY_ALIGNMENT 텍스트가 디스플레이에 맞춰지는 방식을 지정합니다.
MF_TIMED_TEXT_ERROR_CODE 시간 제한 텍스트 트랙에서 발생한 종류 오류를 지정합니다.
MF_TIMED_TEXT_FONT_STYLE 시간 제한 텍스트의 글꼴 스타일을 지정합니다.
MF_TIMED_TEXT_SCROLL_MODE 부모 요소를 스크롤할 때 텍스트가 표시되는 방식을 지정합니다.
MF_TIMED_TEXT_TRACK_KIND 시간 제한 텍스트 트랙의 종류를 지정합니다.
MF_TIMED_TEXT_UNIT_TYPE 시간 제한 텍스트를 측정할 단위를 지정합니다.
MF_TIMED_TEXT_WRITING_MODE 부모 요소에 텍스트를 쓰는 시퀀스를 지정합니다.
MF_TOPOLOGY_RESOLUTION_STATUS_FLAGS MF_TOPOLOGY_RESOLUTION_STATUS 특성의 상태 플래그를 정의합니다.
MF_TOPOLOGY_TYPE 토폴로지 노드의 형식을 정의합니다.
MF_TOPONODE_DRAIN_MODE 토폴로지의 변환이 드레이닝되는 시간을 정의합니다.
MF_TOPONODE_FLUSH_MODE 토폴로지의 변환이 플러시되는 시기를 정의합니다.
MF_TOPOSTATUS 재생 중에 토폴로지의 상태를 지정합니다.
MF_TRANSCODE_ADJUST_PROFILE_FLAGS MF_TRANSCODE_ADJUST_PROFILE 특성에 설정된 프로필 플래그를 정의합니다.
MF_TRANSCODE_TOPOLOGYMODE_FLAGS MF_TRANSCODE_TOPOLOGYMODE 특성에 대한 플래그를 정의합니다.

MF_URL_TRUST_STATUS URL이 신뢰할 수 있는 원본에서 온 것인지 여부를 나타냅니다.
MF_VIDEO_PROCESSOR_ALGORITHM_TYPE MF_VIDEO_PROCESSOR_ALGORITHM 사용하는 비디오 프로세서에 대한 알고리즘을 정의합니다.
MF_VIDEO_PROCESSOR_MIRROR 비디오 이미지를 대칭 이동 하는 방법을 지정 합니다.
MF_VIDEO_PROCESSOR_ROTATION 비디오 이미지를 회전하는 방법을 지정합니다.
MF2DBuffer_LockFlags IMF2DBuffer2::Lock2DSize 메서드에 대한 플래그를 포함합니다.
MF3DVideoOutputType 3D 스테레오스코픽 비디오 스트림을 출력하는 방법을 지정합니다.
MFASF_INDEXER_FLAGS ASF 인덱서 옵션을 정의합니다.
MFASF_MULTIPLEXERFLAGS ASF 멀티플렉서 옵션을 정의합니다.
MFASF_SPLITTERFLAGS ASF 분할자 옵션을 정의합니다.
MFASF_STREAMSELECTOR_FLAGS ASF 스트림 선택기 옵션을 정의합니다.
MFASYNC_WORKQUEUE_TYPE 만들 MFAllocateWorkQueueEx 함수의 작업 큐 유형을 지정합니다.
MFAudioConstriction 오디오 수축 값을 지정합니다.
MFBYTESTREAM_SEEK_ORIGIN 검색 요청의 원본을 지정합니다.
MFCameraOcclusionState 카메라의 폐색 상태를 지정합니다.

MFCLOCK_CHARACTERISTICS_FLAGS 시계의 특성을 설명하는 플래그를 포함합니다.
MFCLOCK_RELATIONAL_FLAGS 클록의 속성을 정의합니다.
MFCLOCK_STATE 클록의 상태를 정의합니다.
MFDepthMeasurement 비디오 프레임의 깊이 값에 대한 측정 시스템을 지정합니다.
MFFrameSourceTypes 프레임 원본에서 제공하는 데이터 형식을 설명합니다.
MFMEDIASOURCE_CHARACTERISTICS 미디어 원본의 특성을 정의합니다.
MFNET_PROXYSETTINGS 기본 프록시 로케이터가 프록시 서버에 대한 연결 설정을 지정하는 방법을 지정합니다.
MFNetAuthenticationFlags 사용자의 자격 증명을 사용하는 방법을 지정합니다.
MFNetCredentialOptions 캐싱 네트워크 자격 증명에 대한 옵션을 설명합니다.
MFNetCredentialRequirements 자격 증명 관리자가 사용자 자격 증명을 가져오는 방법을 지정합니다.
MFNETSOURCE_CACHE_STATE 미디어 파일 또는 항목의 캐시 상태를 정의합니다.
MFNETSOURCE_PROTOCOL_TYPE 스트리밍 또는 다운로드에 사용되는 제어 프로토콜의 유형을 나타냅니다.
MFNETSOURCE_STATISTICS_IDS 네트워크 원본에서 수집한 통계를 정의합니다.
MFNETSOURCE_TRANSPORT_TYPE 스트리밍 또는 다운로드 데이터(TCP 또는 UDP)에 사용되는 전송 유형을 설명합니다.

MFNominalRange 색 데이터에 헤드룸과 토룸이 포함되는지 여부를 지정합니다.
MFPMPSESSION_CREATION_FLAGS MFCreatePMPMediaSession 함수의 동작을 정의하는 플래그를 포함합니다.
MFPOLICYMANAGER_ACTION 스트림에서 수행할 수 있는 작업을 정의합니다.
MFRATE_DIRECTION 재생 방향(정방향 또는 역방향)을 지정합니다.
MFSampleAllocatorUsage MFSampleAllocatorUsage 열거형은 IMFSampleAllocatorControl 인터페이스를 구현하는 구성 요소의 할당자 사용을 지정합니다.
MFSampleEncryptionProtectionScheme 암호화된 샘플에 대해 지원되는 보호 체계를 지정합니다.
MFSensorDeviceMode IMFSensorDevice의 공유 모드를 지정합니다.
MFSensorDeviceType 센서 디바이스의 유형을 지정합니다. 이 열거형의 값은 IMFSensorDevice::GetDeviceType에서 반환됩니다.
MFSensorStreamType IMFSensorDevice와 연결된 스트림이 입력 또는 출력 스트림인지 여부를 지정합니다.
MFSequencerTopologyFlags 시퀀서 원본에 토폴로지를 추가하거나 큐에 이미 있는 토폴로지를 업데이트하기 위한 플래그를 포함합니다.
MFSESSION_GETFULLTOPOLOGY_FLAGS IMFMediaSession::GetFullTopology 메서드에 대한 플래그를 정의합니다.
MFSESSION_SETTOPOLOGY_FLAGS IMFMediaSession::SetTopology 메서드의 동작을 정의합니다.
MFSHUTDOWN_STATUS IMFShutdown::Shutdown 메서드에 대한 호출의 현재 상태를 설명합니다.
MFSINK_WMDRMACTION ASF 파일 싱크에서 Windows Media DRM을 적용하는 방법을 지정합니다.

MFStandardVideoFormat 일반적인 비디오 형식을 지정하는 값을 포함합니다.
MFSTREAMSINK_MARKER_TYPE IMFStreamSink::P laceMarker 메서드에 대한 스트림 마커 정보를 정의합니다.
MFT_MESSAGE_TYPE MFT(Media Foundation 변환)에 대한 메시지를 정의합니다.
MFTIMER_FLAGS IMFTimer::SetTimer 메서드에 대한 플래그를 포함합니다.
MFTOPOLOGY_DXVA_MODE 토폴로지 로더에서 토폴로지에서 Microsoft DXVA(DirectX Video Acceleration)를 사용할 수 있는지 여부를 지정합니다.
MFTOPOLOGY_HARDWARE_MODE 토폴로지 로더가 하드웨어 기반 MFT(Media Foundation 변환)를 토폴로지에 삽입할지 여부를 지정합니다.
MFVideo3DFormat 3D 비디오 프레임이 메모리에 저장되는 방법을 지정합니다.
MFVideo3DSampleFormat 3D 비디오 프레임이 미디어 샘플에 저장되는 방법을 지정합니다.
MFVideoChromaSubsampling Y'Cb'Cr' 데이터에 대한 크로마 인코딩 체계를 정의하는 플래그를 포함합니다.
MFVideoDRMFlags 비디오 스트림에 필요한 복사 보호 유형을 지정합니다.
MFVideoDSPMode 비디오 안정화 MFT의 처리 모드를 지정합니다.
MFVideoFlags 비디오 스트림을 설명하는 플래그를 포함합니다.
MFVideoInterlaceMode 비디오 스트림이 인터레이스되는 방법을 지정합니다.
MFVideoLighting 특정 비디오 콘텐츠 집합을 보기 위한 최적의 조명에 대해 설명합니다.

MFVideoPadFlags 비디오 이미지가 지정된 가로 세로 비율에 맞게 채워질지 여부를 지정합니다.
MFVideoPrimaries 비디오 원본의 색 주체를 지정합니다. (MFVideoPrimaries)
MFVideoRotationFormat 시계 반대 방향으로 비디오 이미지의 회전을 설명합니다.
MFVideoSrcContentHintFlags 비디오 스트림의 의도된 가로 세로 비율을 설명합니다.
MFVideoTransferFunction 선형 RGB에서 비선형 RGB(R'G'B')로 변환 함수를 지정합니다. (MFVideoTransferFunction)
MFVideoTransferMatrix Y'PbPr(구성 요소 비디오)과 스튜디오 R'G'B' 간의 변환 행렬에 대해 설명합니다. (MFVideoTransferMatrix)
MFVirtualCameraAccess 가상 카메라에 대한 액세스 제한을 지정합니다.
MFVirtualCameraLifetime 가상 카메라의 수명을 지정합니다.
MFVirtualCameraType 가상 카메라의 유형을 지정합니다.
MFWaveFormatExConvertFlags 오디오 미디어 형식을 변환하는 방법을 지정하는 플래그를 포함합니다.
MIC_ARRAY_MODE 음성 캡처 DSP가 마이크 배열 처리를 수행하는 방법을 지정합니다. 이 열거형은 MFPKEY_WMAAECMA_FEATR_MICARR_MODE 속성과 함께 사용됩니다.
OPM_ACP_PROTECTION_LEVEL ACP(아날로그 복사 보호)의 보호 수준을 지정합니다.
OPM_DPCP_PROTECTION_LEVEL DPCP(DisplayPort Content Protection)에 대한 보호 수준을 지정합니다.
OPM_HDCP_PROTECTION_LEVEL High-Bandwidth HDCP(Digital Content Protection)에 대한 보호 수준을 지정합니다.

OPM_IMAGE_ASPECT_RATIO_EN300294 ETSI EN 300 294의 가로 세로 비율을 지정합니다.
OPM_VIDEO_OUTPUT_SEMANTICS IOPMVideoOutput 인터페이스에서 OPM(Output Protection Manager) 의미 체계 또는 COPP(Certified Output Protection Protocol) 의미 체계를 사용할지 여부를 지정합니다.
PLAYTO_SOURCE_CREATEFLAGS IPlayToSourceClassFactory::CreateInstance 메서드에 대한 플래그를 포함합니다.
SAMPLE_PROTECTION_VERSION 샘플 보호를 위한 버전 번호를 정의합니다.
TOC_POS_TYPE TOC_POS_TYPE 열거형에는 목차의 위치 형식을 지정하는 멤버가 포함됩니다.
WMT_PROP_DATATYPE IWMCodecProps 인터페이스의 메서드를 사용하여 액세스하는 코덱 및 DSP 속성에 사용되는 데이터 형식을 정의합니다.

함수

테이블 확장

중단 현재 미디어 세그먼트의 처리를 중단합니다.
ActivateClassById PMP(보호된 미디어 경로) 프로세스에서 Windows 런타임 개체를 만듭니다.
ActivateObject 이 활성화 개체와 연결된 개체를 만듭니다.
AddBlockedControl 지정된 차단된 컨트롤을 추가합니다.
AddBuffer 샘플의 버퍼 목록 끝에 버퍼를 추가합니다.
AddClockStateSink

클록이 시작, 중지 또는 일시 중지되거나 속도가 변경될 때마다 알림을 받을 개체를 등록합니다.
AddControlSubscription IMFCameraControlNotify::OnChange 알림이 발생할 컨트롤 목록에 카메라 컨트롤을 추가합니다.
AddDataSource 시간 제한 텍스트 데이터 원본을 추가합니다.
AddDataSourceFromUrl 지정된 URL에서 시간 제한 텍스트 데이터 원본을 추가합니다.
AddDeviceSourceInfo 지정된 디바이스 기호 이름으로 지정된 물리적 카메라에 대한 단독 제어가 필요한 가상 카메라를 파이프라인에 알릴 수 있습니다.
AddEffect 캡처 스트림에 효과를 추가합니다.
AddElement 컬렉션에 개체를 추가합니다. (IMFCollection.AddElement)
AddElement 목록의 끝에 원본 요소를 추가합니다.
AddElementEx 향상된 버전의 IMFMediaEngineSrcElements::AddElement를 제공하여 요소에 콘텐츠와 함께 사용할 키 시스템을 추가합니다.
AddEntry AddEntry 메서드는 컬렉션에 개별 목차를 추가하고 추가된 목차에 인덱스를 할당합니다.
AddEntry AddEntry 메서드는 목록에 개별 항목을 추가하고 항목에 인덱스를 할당합니다.
AddEntryList AddEntryList 메서드는 목차에 항목 목록을 추가하고 항목 목록에 인덱스를 할당합니다.
AddHeader 단일 HTTP 헤더를 HTTP 요청에 추가하기 위해 Microsoft Media Foundation에서 호출합니다. Microsoft Media Foundation은 BEGINSendRequest 메서드를 호출하기 전에 HTTP 요청에 포함해야 하는 각 헤더에 대해 이 메서드를 한 번 호출합니다.
AddInputStreams 이 MFT(Media Foundation 변환)에 하나 이상의 새 입력 스트림을 추가합니다.

AddMutualExclusion 구성된 ASF(Advanced Systems Format) 상호 제외 개체를 프로필에 추가합니다.
AddNode 토폴로지에 노드를 추가합니다.
AddPayloadExtension 스트림에 대한 페이로드 확장을 구성합니다.
AddProfile 컬렉션에 지정된 프로필을 추가합니다.
AddProfileFilter 지정된 미디어 스트림에 프로필 필터를 추가합니다.
AddProperty 가상 카메라에 사용자 지정 디바이스 인터페이스 속성을 추가합니다.
AddRange 시간 범위 목록에 새 범위를 추가합니다.
AddRecord 상호 제외 개체에 레코드를 추가합니다. 레코드는 다른 모든 레코드의 스트림과 함께 사용할 수 없는 스트림을 지정합니다.
AddRegistryEntry 디바이스 인터페이스 레지스트리 키에 사용자 지정 레지스트리 항목을 추가합니다.
AddSourceBuffer IMFMediaSourceExtension과 연결된 버퍼 컬렉션에 IMFSourceBuffer를 추가합니다.
AddSpatialAudioObject IMFSpatialAudioObjectBuffer 개체로 표현되는 새 공간 오디오 개체를 샘플에 추가합니다.
AddStream 이 인터페이스는 이 버전의 Media Foundation에서 구현되지 않습니다. 스트림 우선 순위 목록에 스트림을 추가합니다.
AddStream 캡처 원본의 스트림을 이 캡처 싱크에 연결합니다.
AddStream

싱크 작성기에 스트림을 추가합니다.
AddStreamConfiguration 멀티플렉싱된 출력을 포함할 수 있는 하위 스트림 집합을 정의하는 스트림 구성을 등록합니다.
AddStreamForRecord Advanced Systems Format 상호 제외 개체의 레코드에 스트림 번호를 추가합니다.
AddStreamPrioritization IMFASFProfile::AddStreamPrioritization 메서드
AddStreamSink 미디어 싱크에 새 스트림 싱크를 추가합니다.
addToc AddToc 메서드는 목차를 TOC 파서 개체에 추가하고 추가된 목차에 인덱스를 할당합니다.
AddTransformForStream 오디오 또는 비디오 효과와 같은 변환을 스트림에 추가합니다.
AllocateSample 할당자에서 비디오 샘플을 가져옵니다.
추가 지정된 미디어 세그먼트를 IMFSourceBuffer에 추가합니다.
AppendByteStream 지정된 바이트 스트림의 미디어 세그먼트를 IMFSourceBuffer에 추가합니다.
AppendTopology 큐의 끝에 토폴로지 추가
ApplyStreamSelections SetStreamSelection에 이전 호출의 스트림 선택을 적용합니다.
AttachMediaSource 미디어 원본을 연결합니다.
automaticEnable 사용자 상호 작용 없이 콘텐츠를 사용하도록 설정하는 작업을 수행합니다.
장벽 비디오 디코딩 명령 목록 기록에 장벽 컬렉션을 추가합니다.

장벽 비디오 인코딩 명령 목록 기록에 장벽 컬렉션을 추가합니다.
장벽 비디오 프로세스 명령 목록 기록에 장벽 컬렉션을 추가합니다.
시작 일시 중단 프로세스가 시작되고 리소스가 일관된 상태로 전환되어야 했음을 나타냅니다.
beginConvertHNSToTimecode 시간을 100나노초 단위로 SMPTE(영화 및 텔레비전 엔지니어 협회) 시간 코드로 변환하는 비동기 호출을 시작합니다.
BeginConvertTimecodeToHNS SMPTE(영화 및 텔레비전 엔지니어 협회) 시간 코드를 100나노초 단위로 변환하기 위한 비동기 호출을 시작합니다.
beginCreateObject 바이트 스트림에서 미디어 원본을 만들기 위한 비동기 요청을 시작합니다. (IMFByteStreamHandler.BeginCreateObject)
beginCreateObject URL에서 개체를 만들기 위한 비동기 요청을 시작합니다. 원본 확인자가 URL에서 미디어 원본을 만들면 요청이 스키마 처리기에 전달됩니다.
beginCreateObject 바이트 스트림 또는 미디어 원본을 만드는 비동기 요청을 시작합니다.
BeginCreateObjectFromByteStream 바이트 스트림에서 미디어 원본을 만들기 위한 비동기 요청을 시작합니다. (IMFSourceResolver.BeginCreateObjectFromByteStream)
BeginCreateObjectFromURL URL에서 미디어 원본 또는 바이트 스트림을 만드는 비동기 요청을 시작합니다.
BeginEnableContent 콘텐츠를 사용하도록 설정하는 작업을 수행하기 위한 비동기 요청을 시작합니다.
BeginEvent 내부 전용입니다. 직접 호출할 수 없습니다. (ID3D12VideoDecodeCommandList::BeginEvent)
BeginEvent 내부 전용입니다. 직접 호출할 수 없습니다. (ID3D12VideoEncodeCommandList::BeginEvent)

BeginEvent 내부 전용입니다. 직접 호출할 수 없습니다. (ID3D12VideoProcessCommandList::BeginEvent)
BeginFinalize 미디어 싱크에 작업을 완료하는 데 필요한 모든 단계를 비동기적으로 수행하도록 알 수 있습니다.
BeginFrame 디코딩 작업을 시작합니다.
beginGetClientCertificate 클라이언트 SSL 인증서를 가져오기 위한 비동기 호출을 시작합니다.
BeginGetCredentials 사용자의 자격 증명을 검색하기 위한 비동기 요청을 시작합니다.
beginGetEvent 큐에서 다음 이벤트에 대한 비동기 요청을 시작합니다.
beginGetEvent 큐에서 다음 이벤트에 대한 비동기 요청을 시작합니다. IMFMediaEventGenerator::BeginGetEvent 구현 내에서 이 메서드를 호출합니다. 해당 메서드의 매개 변수를 이 메서드로 직접 전달합니다.
beginGetReport 상대 패널 위치를 나타내는 IMFRelativePanelReport 인터페이스를 가져오기 위한 비동기 요청을 시작합니다.
BeginQuery 쿼리 실행을 시작합니다. (ID3D12VideoDecodeCommandList::BeginQuery)
BeginQuery 쿼리 실행을 시작합니다. (ID3D12VideoEncodeCommandList::BeginQuery)
BeginQuery 쿼리 실행을 시작합니다. (ID3D12VideoProcessCommandList::BeginQuery)
beginRead 스트림에서 비동기 읽기 작업을 시작합니다.
beginReadPayload 이전에 보낸 HTTP 또는 HTTPS 요청에 대한 응답의 메시지 본문을 수신하기 위해 Microsoft Media Foundation에서 호출합니다.
BeginReceiveResponse 이전에 전송된 HTTP 또는 HTTPS 요청에 대한 응답으로 서버에서 제공하는 응답을 받기 위해 Microsoft Media

Foundation에서 호출합니다. Media Foundation은 EndSendRequest 메서드를 성공적으로 호출한 후에만 이 메서드를 호출합니다.
BeginRegisterPlatformWorkQueueWithMMCSS 플랫폼 작업 큐를 MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업과 연결합니다.
BeginRegisterPlatformWorkQueueWithMMCSSEx 지정된 클래스 및 작업 ID를 사용하여 플랫폼 작업 큐를 MMCSS(멀티미디어 클래스 스케줄러 서비스)에 등록합니다.
BeginRegisterTopologyWorkQueuesWithMMCSS 토폴로지 작업 큐를 MMCSS(멀티미디어 클래스 스케줄러 서비스)에 등록합니다.
BeginSave Windows Media 파일을 애플리케이션의 바이트 스트림에 저장하기 시작합니다.
beginSendRequest HTTP 또는 HTTPS 요청을 보내기 위해 Microsoft Media Foundation에서 호출합니다.
BeginUnregisterPlatformWorkQueueWithMMCSS MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업에서 플랫폼 작업 큐의 등록을 취소합니다.
BeginUnregisterTopologyWorkQueuesWithMMCSS MMCSS(멀티미디어 클래스 스케줄러 서비스)에서 토폴로지 작업 큐의 등록을 취소합니다.
BeginWrite 스트림에 대한 비동기 쓰기 작업을 시작합니다.
beginWriteSample 스트림에 미디어 샘플을 쓰는 비동기 요청을 시작합니다.
BeginWriting 쓰기를 위해 싱크 작성기를 초기화합니다.
BindAccess 요청된 작업이 수행될 예정임을 ITA(입력 신뢰 기관)에 알깁니다.
BitrateToStepNumber 지정된 비트 전송률에 적합한 대역폭 단계의 인덱스를 검색합니다. 이 메서드는 MBR(다중 비트 전송률) 콘텐츠에 사용됩니다.
통화 콘텐츠 보호 시스템이 보호된 환경에 액세스할 수 있도록 허용합니다.

취소 보류 중인 콘텐츠 사용 작업을 취소합니다.
cancelObjectCreation 미디어 원본을 만들기 위한 현재 요청을 취소합니다.
cancelObjectCreation URL에서 개체를 만들기 위한 현재 요청을 취소합니다.
cancelObjectCreation 개체를 만들기 위한 비동기 요청을 취소합니다.
cancelObjectCreation 개체를 만들기 위한 현재 요청을 취소합니다.
CancelSave IMFSaveJob::BeginSave에서 시작한 작업을 취소합니다.
CancelTimelineMarkerTimer 보류 중인 다음 타임라인 표식을 취소합니다.
CancelTimer IMFTimer::SetTimer 메서드를 사용하여 설정된 타이머를 취소합니다.
CanPlayType 미디어 엔진이 지정된 형식의 미디어 리소스를 재생할 수 있는지 쿼리합니다.
CanPlayType 개체가 지정된 형식의 미디어 리소스를 로드할 수 있는지 여부를 쿼리합니다.
CheckCryptoKeyExchange 드라이버에서 지원하는 암호화 키 교환 메커니즘을 가져옵니다.
checkCryptoSessionStatus 암호화 세션의 상태를 확인합니다.
checkDeviceOverlayType Direct3D 디바이스의 오버레이 하드웨어 기능을 쿼리합니다. (IDirect3D9ExOverlayExtension.CheckDeviceOverlayType)
CheckFeatureSupport 현재 비디오 드라이버에서 지원하는 기능에 대한 정보를 가져옵니다. (ID3D12VideoDevice::CheckFeatureSupport)

CheckVideoDecoderDownsampling 비디오 디코더가 지정된 입력 형식으로 다운샘플링을 지원하는지 여부와 실시간 다운샘플링이 지원되는지 여부를 나타냅니다.
CheckVideoDecoderFormat aprofile이 지정된 경우 드라이버가 지정된 출력 형식을 지원하는지 여부를 확인합니다.
CheckVideoProcessorFormat 비디오 프로세서가 지정된 비디오 형식을 지원하는지 여부를 쿼리합니다.
CheckVideoProcessorFormatConversion 드라이버가 지정된 형식 및 색 영역 변환 조합을 지원하는지 여부를 나타냅니다.
지우기 토폴로지에서 모든 노드를 제거합니다.
지우기 시간 범위 목록을 지웁니다.
ClearState 직접 명령 목록의 상태를 명령 목록을 만들 때의 상태로 다시 설정합니다. (ID3D12VideoDecodeCommandList::ClearState)
ClearState 직접 명령 목록의 상태를 명령 목록을 만들 때의 상태로 다시 설정합니다. (ID3D12VideoEncodeCommandList::ClearState)
ClearState 직접 명령 목록의 상태를 명령 목록을 만들 때의 상태로 다시 설정합니다. (ID3D12VideoProcessCommandList::ClearState)
ClearTopologies 미디어 세션에서 재생을 위해 대기 중인 모든 프레젠테이션을 지웁니다.
복제 기본 프록시 로케이터의 새 인스턴스를 만듭니다.
복제 이 프레젠테이션 설명자의 복사본을 만듭니다.
복제 Advanced Systems Format 상호 제외 개체의 복사본을 만듭니다.

복제 Advanced Systems Format 프로필 개체의 복사본을 만듭니다.
복제 ASF(Advanced Systems Format) 스트림 구성 개체의 복사본을 만듭니다.
복제 이 인터페이스는 이 버전의 Media Foundation에서 구현되지 않습니다. ASF 스트림 우선 순위 지정 개체의 복사본을 만듭니다.
CloneFrom 이 토폴로지의 복사본을 다른 토폴로지로 변환합니다.
CloneFrom 다른 토폴로지 노드의 데이터를 이 노드로 복사합니다.
CloneSensorStream IMFSensorStream을 복제합니다.
닫기 명령 목록에 대한 기록이 완료되었음을 나타냅니다. (ID3D12VideoDecodeCommandList::Close)
닫기 명령 목록에 대한 기록이 완료되었음을 나타냅니다. (ID3D12VideoEncodeCommandList::Close)
닫기 명령 목록에 대한 기록이 완료되었음을 나타냅니다. (ID3D12VideoProcessCommandList::Close)
닫기 MICROSOFT Media Foundation에서 호출하여 IMFHttpDownloadRequest가 내부 리소스를 해제할 수 있도록 합니다. 현재 요청이 아직 진행 중인 경우에도 취소됩니다.
닫기 더 이상 HTTP 요청이 생성되지 않음을 지정하기 위해 Microsoft Media Foundation에서 호출하고 IMFHttpDownloadSession에서 내부 리소스를 해제할 수 있도록 합니다.
닫기 미디어 세션을 닫고 사용 중인 모든 리소스를 해제합니다.
닫기 스트림을 닫고 소켓 또는 파일 핸들과 같은 스트림과 연결된 모든 리소스를 해제합니다. 또한 이 메서드는 보류 중인 모든 비동기 I/O 요청을 취소합니다.

닫기 미디어 키 세션을 닫고 키 세션을 해제하기 전에 호출해야 합니다.
닫기 애플리케이션에 세션이 더 이상 필요하지 않으며 CDM(콘텐츠 암호 해독 모듈)이 세션과 연결된 모든 리소스를 해제하고 닫아야 했음을 나타냅니다.
closeDeviceHandle Microsoft Direct3D 디바이스 핸들을 닫습니다.
closeDeviceHandle Direct3D 디바이스 핸들을 닫습니다.
커밋 Commit 메서드는 TOC 파서 개체의 현재 상태를 연결된 미디어 파일에 저장합니다.
CommitSettings 구성된 컨트롤 설정을 카메라 드라이버에 커밋합니다.
비교 이 개체의 특성을 다른 개체의 특성과 비교합니다.
CompareItem 저장된 특성 값이 지정된 PROPVARIANT와 같은지 여부를 쿼리합니다.
구성 인증된 채널에 구성 명령을 보냅니다.
구성 비디오 출력을 구성합니다.
ConfigureAuthenticatedChannel 인증된 채널에 구성 명령을 보냅니다.
연결 미디어 요소를 미디어 공유 엔진에 연결합니다.
ConnectOutput 이 노드의 출력 스트림을 다른 노드의 입력 스트림에 연결합니다.
ConnectWebAudio 지정된 샘플 속도를 사용하여 웹 오디오를 미디어 엔진에 연결합니다.

ContainsTime 지정된 시간이 시간 범위 내에 속하는지 여부를 쿼리합니다.
ContiguousCopyFrom 연속 형식의 버퍼에서 이 버퍼로 데이터를 복사합니다.
ContiguousCopyTo 이 버퍼를 호출자의 버퍼에 복사하여 데이터를 연속 형식으로 변환합니다.
ConvertToContiguousBuffer 여러 버퍼가 있는 샘플을 단일 버퍼가 있는 샘플로 변환합니다.
COPPCompatibleGetInformation 표시 드라이버에 OPM(출력 보호 관리자) 상태 요청을 보냅니다. OPM이 COPP(Certified Output Protection Manager)를 에뮬레이트하는 경우 이 메서드를 사용합니다.
Copy2DTo 버퍼를 다른 2D 버퍼 개체에 복사합니다.
copyAllItems 이 개체의 모든 특성을 다른 특성 저장소에 복사합니다.
copyToBuffer 샘플 데이터를 버퍼에 복사합니다. 이 메서드는 샘플의 모든 버퍼에서 유효한 데이터를 순서대로 연결합니다.
CreateAuthenticatedChannel Microsoft Direct3D 디바이스 또는 그래픽 드라이버와 통신할 채널을 만듭니다.
CreateAuthenticatedChannel Direct3D 디바이스 또는 그래픽 드라이버와 통신할 채널을 만듭니다.
CreateByteStreamProxy 바이트 스트림에 대한 프록시를 만듭니다. (IMFByteStreamProxyClassFactory.CreateByteStreamProxy)
CreateContentDecryptionModule IMFContentDecryptionModuleAccess::CreateContentDecryptionModule 함수는 DRM 키 시스템에 대한 CDM(Content Decryption Module)을 나타내는 IMFContentDecryptionModule을 만듭니다.
createContentDecryptionModuleAccess IMFContentDecryptionModuleAccess 인터페이스의 인스턴스를 만듭니다.
CreateContentDecryptionModuleFactory

지정된 키 시스템에 대한 CDM(Content Decryption Module) 개체의 클래스 팩터리인 IMFContentDecryptionModuleFactory의 인스턴스를 만듭니다.
CreateCryptoSession 그래픽 드라이버로 전송되는 비디오 콘텐츠를 암호화하는 암호화 세션을 만듭니다.
CreateCryptoSession 디스플레이 드라이버로 전송되는 비디오 콘텐츠를 암호화하는 암호화 세션을 만듭니다.
createError 미디어 오류 개체를 만듭니다.
CreateHttpDownloadSession Microsoft Media Foundation에서 기본 구현을 사용하는 대신 HTTP 또는 HTTPS URL을 열도록 호출합니다.
CreateInstance 캡처 엔진의 인스턴스를 만듭니다. (IMFCaptureEngineClassFactory.CreateInstance)
CreateInstance 미디어 엔진의 새 인스턴스를 만듭니다.
CreateInstance IMFMediaSharingEngine의 인스턴스를 만듭니다. (IMFMediaSharingEngineClassFactory.CreateInstance)
CreateInstance 미디어 공유 엔진의 인스턴스를 만듭니다. (IMFSharingEngineClassFactory.CreateInstance)
CreateInstance PlayToController 개체의 인스턴스를 만듭니다.
createInstanceFromObject IUnknown 포인터가 지정된 경우 싱크 기록기 또는 원본 판독기의 인스턴스를 만듭니다.
CreateInstanceFromUDN 제공된 고유한 디바이스 이름에서 IMFImageSharingEngine의 인스턴스를 만듭니다.
CreateInstanceFromURL URL이 지정된 싱크 작성기 또는 원본 판독기의 인스턴스를 만듭니다.
CreateMediaKeys 지정된 키 시스템을 기반으로 미디어 키 개체를 만듭니다. (IMFMediaEngineClassFactoryEx.CreateMediaKeys)
CreateMediaKeys2

지정된 키 시스템을 기반으로 미디어 키 개체를 만듭니다. (IMFMediaEngineClassFactory2.CreateMediaKeys2)
CreateMediaSource 센서 그룹을 가상화하는 IMFMediaSource를 만듭니다.
CreateMediaSourceExtension IMFMediaSourceExtension의 인스턴스를 만듭니다.
CreateMutualExclusion 새 ASF(Advanced Systems Format) 상호 제외 개체를 만듭니다. AddMutualExclusion 메서드를 호출하여 프로필에 상호 제외 개체를 추가할 수 있습니다.
CreateNamedPropertyStore 이름/값 쌍을 저장할 빈 속성 저장소를 만듭니다.
CreateObjectByCLSID CLSID에서 PMP(미디어 경로 보호) 프로세스에 개체를 만듭니다.
CreateObjectByCLSID PMP(보호된 미디어 경로) 프로세스에서 개체를 만듭니다.
CreateObjectFromByteStream 바이트 스트림에서 미디어 원본을 만듭니다. 이 메서드는 동기적입니다.
CreateObjectFromURL URL에서 미디어 원본 또는 바이트 스트림을 만듭니다. 이 메서드는 동기적입니다.
CreatePresentationDescriptor 미디어 원본의 프레젠테이션 설명자의 복사본을 검색합니다. 애플리케이션은 프레젠테이션 설명자를 사용하여 스트림을 선택하고 원본 콘텐츠에 대한 정보를 가져옵니다.
CreatePropertyStore 빈 속성 저장소 개체를 만듭니다.
createProxyLocator 프로토콜 이름을 기반으로 IMFNetProxyLocator 인터페이스 프록시 로케이터 개체를 만듭니다.
createRequest 단일 HTTP 또는 HTTPS 요청을 보내는 데 사용되는 IMFHttpDownloadRequest 인터페이스를 구현하는 개체를 만들기 위해 Microsoft Media Foundation에서 호출합니다.
createSession 지정된 초기화 데이터 및 사용자 지정 데이터를 사용하여 미디어 키 세션 개체를 만듭니다. .

createSession IMFContentDecryptionModule::CreateSession 함수는 CDM(Content Decryption Module) 세션을 나타내는 IMFContentDecryptionModuleSession 개체를 만듭니다.
CreateStream ASF(Advanced Systems Format) 스트림 구성 개체를 만듭니다.
CreateStreamPrioritization IMFASFProfile::CreateStreamPrioritization 메서드
CreateSurface DXVA(DirectX Video Acceleration) 비디오 프로세서 또는 DXVA 디코더 렌더링 대상을 만듭니다.
CreateSyncEvent 이벤트를 사용하거나 사용하지 않도록 설정하는 IKsControl::KsEvent 메서드 주위의 래퍼입니다.
CreateSyncSemaphore 세마포를 사용하거나 사용하지 않도록 설정하는 IKsControl::KsEvent 메서드 주위의 래퍼입니다.
CreateTimeRange 시간 범위 개체를 만듭니다.
CreateTransform 변환을 만들기 위해 미디어 파이프라인에서 호출됩니다.
CreateTrustedInput 콘텐츠의 암호 해독을 구현하는 IMFTrustedInput 개체를 만듭니다.
CreateVideoDecoder Microsoft Direct3D 11용 비디오 디코더 디바이스를 만듭니다.
CreateVideoDecoder 비디오 디코더 인스턴스를 만듭니다.
CreateVideoDecoder 비디오 디코더 디바이스를 만듭니다.
CreateVideoDecoder1 보호된 리소스를 지원하는 해상도 독립적 드라이버 리소스 및 상태를 포함하는 비디오 디코더 인스턴스를 만듭니다.
CreateVideoDecoderHeap

비디오 디코더 힙을 할당합니다.
CreateVideoDecoderHeap1 보호된 리소스를 지원하는 해상도 종속 드라이버 리소스 및 상태를 포함하는 비디오 디코더 힙을 할당합니다.
CreateVideoDecoderOutputView 디코딩 작업에 대한 출력 샘플을 설명하는 비디오 디코더에 대한 리소스 뷰를 만듭니다.
CreateVideoEncoder ID3D12VideoEncoder의 새 인스턴스를 만듭니다.
CreateVideoEncoderHeap ID3D12VideoEncoderHeap의 새 인스턴스를 만듭니다.
CreateVideoExtensionCommand 비디오 확장 명령을 만듭니다.
CreateVideoMotionEstimator 비디오 동작 추정 작업에 대한 컨텍스트를 유지하는 ID3D12VideoMotionEstimator를 만듭니다.
CreateVideoMotionVectorHeap 비디오 동작 추정을 위한 동작 벡터가 포함된 힙을 할당합니다.
CreateVideoProcessor Microsoft Direct3D 11용 비디오 프로세서 디바이스를 만듭니다.
CreateVideoProcessor 비디오 프로세서 인스턴스를 만듭니다.
CreateVideoProcessor Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서를 만듭니다.
CreateVideoProcessor 비디오 프로세서 디바이스를 만듭니다.
createVideoProcessor1 보호된 리소스를 지원하는 비디오 프로세서 인스턴스를 만듭니다.
createVideoProcessorEnumerator 드라이버의 비디오 프로세서 기능을 열거합니다.
createVideoProcessorInputView 비디오 처리 작업에 대한 입력 샘플을 설명하는 비디오 프로세서에 대한 리소스 보기를 만듭니다.

CreateVideoProcessorOutputView 비디오 처리 작업에 대한 출력 샘플을 설명하는 비디오 프로세서에 대한 리소스 보기를 만듭니다.
CreateVideoSurface 하나 이상의 Microsoft Direct3D 비디오 화면을 만듭니다.
큐 텍스트 트랙에서 큐 이벤트가 발생할 때 호출됩니다.
DecodeFrame 디코딩 프레임 작업을 명령 목록에 기록합니다.
DecodeFrame1 디코딩 프레임 작업을 명령 목록에 기록합니다. 디코딩에 대한 입력, 출력 및 매개 변수는 이 메서드의 인수로 지정됩니다.
DecoderBeginFrame 비디오 프레임을 디코딩하는 디코딩 작업을 시작합니다.
DecoderEnableDownsampling 디코더 다운샘플링이 사용되고 드라이버가 적절한 참조 프레임을 할당해야 임을 나타냅니다.
DecoderEndFrame 디코딩 작업의 종료를 알릴 수 있습니다.
DecoderExtension 디코딩을 위해 확장 함수를 수행합니다.
DecoderUpdateDownsampling 디코더 다운샘플링 매개 변수를 업데이트합니다.
DecryptionBlt 암호화된 데이터를 보호된 표면에 씁니다. (ID3D11VideoContext.DecryptionBlt)
DecryptionBlt 암호화된 데이터를 보호된 표면에 씁니다. (IDirect3DCryptoSession9.DecryptionBlt)
DEFINE_MEDIATYPE_GUID FOURCC 코드, D3DFORMAT 값 또는 오디오 형식 형식에서 미디어 하위 형식 GUID를 정의합니다.
DeleteAllItems 개체의 특성 목록에서 모든 키/값 쌍을 제거합니다.

DeleteInputStream 이 MFT(Media Foundation 변환)에서 입력 스트림을 제거합니다.
DeleteItem 개체의 특성 목록에서 키/값 쌍을 제거합니다.
DeleteProperty 메타데이터 속성을 삭제합니다.
DeleteTopology 큐에서 토폴로지 삭제합니다.
DeselectStream 프레젠테이션에서 스트림의 선택을 취소합니다.
DetachMediaSource 미디어 원본을 분리합니다.
DetachObject 생성된 개체를 활성화 개체에서 분리합니다.
DiscardResource 리소스의 현재 콘텐츠를 삭제할 수 있음을 나타냅니다. (ID3D12VideoDecodeCommandList::D iscardResource)
DiscardResource 리소스의 현재 콘텐츠를 삭제할 수 있음을 나타냅니다. (ID3D12VideoEncodeCommandList::D iscardResource)
DiscardResource 리소스의 현재 콘텐츠를 삭제할 수 있음을 나타냅니다.
연결 끊기 미디어 공유 엔진에서 미디어 요소의 연결을 끊습니다.
DisconnectOutput 이 노드에서 출력 스트림의 연결을 끊습니다.
DisconnectWebAudio 미디어 엔진에서 웹 오디오 연결을 끊습니다.
dropTime 지정된 시간 간격 동안 샘플을 삭제합니다.

DXVA2_Fixed32OpaqueAlpha 불투명 알파 값을 포함하는 DXVA2_Fixed32 구조체를 반환합니다.
DXVA2_Fixed32TransparentAlpha 투명한 알파 값을 포함하는 DXVA2_Fixed32 구조를 반환합니다.
DXVA2CreateDirect3DDeviceManager9 Direct3D 디바이스 관리자의 인스턴스를 만듭니다.
DXVA2CreateVideoService DXVA(DirectX Video Acceleration) 서비스 개체를 만듭니다.
DXVA2FixedToFloat DXVA2_Fixed32 값을 부동 소수점 숫자로 변환합니다.
DXVA2FloatToFixed 부동 소수점 숫자를 DXVA2_Fixed32 값으로 변환합니다.
DXVAHD_CreateDevice Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스를 만듭니다.
EnableBuffering 버퍼링을 사용하거나 사용하지 않도록 설정합니다.
EnableHardwareEffects IDirectXVideoProcessor::VideoProcessorBlt로 구현된 효과를 사용하도록 설정합니다.
EnableHorizontalMirrorMode 비디오 미러링을 사용하거나 사용하지 않도록 설정합니다.
EnableTimeUpdateTimer 시간 업데이트 타이머를 사용하거나 사용하지 않도록 설정합니다.
EnableWindowlessSwapchainMode 창 없는 스왑 체인 모드를 사용하거나 사용하지 않도록 설정합니다.
EncodeFrame 비트 스트림을 인코딩합니다.
EncryptionBlt 보호된 화면에서 암호화된 데이터를 읽습니다. (ID3D11VideoContext.EncryptionBlt)

EncryptionBlt 보호된 화면에서 암호화된 데이터를 읽습니다. (IDirect3DCryptoSession9.EncryptionBlt)
끝 멀티플렉서에서 데이터를 수집하고 ASF Header 개체에 해당 정보를 포함하도록 ASF ContentInfo 개체를 업데이트합니다.
끝 실제 일시 중단이 발생하려고 하며 CDM(콘텐츠 암호 해독 모듈)에 대한 호출이 더 이상 수행되지 않습니다.
endConvertHNSToTimecode 시간을 100나노초 단위로 SMPTE(영화 및 텔레비전 엔지니어 협회) 시간 코드로 변환하는 비동기 요청을 완료합니다.
EndConvertTimecodeToHNS SMPTE(영화 및 텔레비전 엔지니어 협회) 시간 코드를 100나노초 단위로 변환하는 비동기 요청을 완료합니다.
EndCreateObject 미디어 원본을 만들기 위한 비동기 요청을 완료합니다.
EndCreateObject URL에서 개체를 만드는 비동기 요청을 완료합니다. (IMFSchemeHandler.EndCreateObject)
EndCreateObject 바이트 스트림 또는 미디어 원본을 만드는 비동기 요청을 완료합니다.
EndCreateObjectFromByteStream 바이트 스트림에서 미디어 원본을 만드는 비동기 요청을 완료합니다.
EndCreateObjectFromURL URL에서 개체를 만드는 비동기 요청을 완료합니다. (IMFSourceResolver.EndCreateObjectFromURL)
endEnableContent 콘텐츠를 사용하도록 설정하는 작업을 수행하기 위한 비동기 요청을 종료합니다. 이 메서드는 PMP(보호된 미디어 경로)에 의해 호출되어 IMFContentProtectionManager::BeginEnableContent에 대한 비동기 호출을 완료합니다.
EndEvent 내부 전용입니다. 직접 호출할 수 없습니다. (ID3D12VideoDecodeCommandList::EndEvent)
EndEvent 내부 전용입니다. 직접 호출할 수 없습니다. (ID3D12VideoEncodeCommandList::EndEvent)
EndEvent

내부 전용입니다. 직접 호출할 수 없습니다. (ID3D12VideoProcessCommandList::EndEvent)
EndFinalize 비동기 종료 작업을 완료합니다.
엔드프레임 디코딩 작업의 종료를 알릴 수 있습니다.
EndGetClientCertificate 클라이언트 SSL 인증서를 가져오기 위한 비동기 요청을 완료합니다.
EndGetCredentials 사용자의 자격 증명을 검색하는 비동기 요청을 완료합니다.
EndGetEvent 큐에서 다음 이벤트에 대한 비동기 요청을 완료합니다.
EndGetEvent 큐에서 다음 이벤트에 대한 비동기 요청을 완료합니다. IMFMediaEventGenerator::EndGetEvent 구현 내에서 이 메서드를 호출합니다. 해당 메서드의 매개 변수를 이 메서드로 직접 전달합니다.
EndGetReport 상대 패널 위치를 나타내는 IMFRelativePanelReport 인터페이스를 가져오기 위한 비동기 요청을 종료합니다.
EndQuery 쿼리를 종료합니다. (ID3D12VideoDecodeCommandList::EndQuery)
EndQuery 쿼리를 종료합니다. (ID3D12VideoEncodeCommandList::EndQuery)
EndQuery 쿼리를 종료합니다. (ID3D12VideoProcessCommandList::EndQuery)
EndRead 비동기 읽기 작업을 완료합니다.
EndReadPayload BeginReadPayload에서 시작된 비동기 작업을 완료하기 위해 Microsoft Media Foundation에서 호출합니다.
EndReceiveResponse BeginReceiveResponse에서 시작한 비동기 작업을 완료하기 위해 Microsoft Media Foundation에서 호출합니다.
EndRegisterPlatformWorkQueueWithMMCSS

플랫폼 작업 큐를 MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업과 연결하는 비동기 요청을 완료합니다.
EndRegisterTopologyWorkQueuesWithMMCSS 토폴로지 작업 큐를 MMCSS(멀티미디어 클래스 스케줄러 서비스)에 등록하기 위한 비동기 요청을 완료합니다.
EndSave IMFSaveJob::BeginSave에서 시작한 작업을 완료합니다.
EndSendRequest BeginSendRequest에서 시작한 비동기 작업을 완료하기 위해 Microsoft Media Foundation에서 호출합니다.
EndUnregisterPlatformWorkQueueWithMMCSS MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업에서 플랫폼 작업 큐의 등록을 취소하는 비동기 요청을 완료합니다.
EndUnregisterTopologyWorkQueuesWithMMCSS MMCSS(멀티미디어 클래스 스케줄러 서비스)에서 토폴로지 작업 큐의 등록을 취소하는 비동기 요청을 완료합니다.
EndWrite 비동기 쓰기 작업을 완료합니다.
endWriteSample 스트림에 미디어 샘플을 쓰는 비동기 요청을 완료합니다.
EnqueueResourceReady 연결된 D3D12 리소스를 사용할 준비가 되면 다운스트림 소비자에게 신호를 보낼 지정된 생산자 명령 큐에 펜스를 큐에 대기합니다.
EnqueueResourceReadyWait 지정된 소비자 명령 큐에서 대기 명령을 큐에 대기하고 생산자 명령 큐에서 리소스 준비 신호를 대기하기 시작합니다.
EnqueueResourceRelease GPU가 소비자 명령 처리를 완료할 때 동기화 개체에 신호를 보낼 지정된 명령 큐에 펜스를 큐에 넣습니다.
오류 텍스트 트랙에서 오류가 발생할 때 호출됩니다.
EstimateMotion 동작 추정 작업을 수행합니다.
EventNotify 재생 이벤트가 발생할 때 애플리케이션에 알깁니다.

실행 현재 프레임에서 디코딩 작업을 실행합니다.
executeExtensionCommand 비디오 확장 명령을 디코딩 명령 목록에 실행하는 명령을 기록합니다.
executeExtensionCommand 비디오 확장 명령을 실행합니다.
executeExtensionCommand 비디오 확장 명령을 인코딩 명령 목록에 실행하는 명령을 기록합니다.
executeExtensionCommand 비디오 확장 명령을 비디오 프로세스 명령 목록에 실행하는 명령을 기록합니다.
완료 싱크 작성기에 대한 모든 쓰기 작업을 완료합니다.
FindFirstProxy 프록시 로케이터 개체를 초기화합니다.
FindNextProxy 사용할 다음 프록시를 결정합니다.
FindProfile 지정된 프로필 ID를 기반으로 프로필을 찾습니다.
FinishInitialization OPM(출력 보호 관리자) 세션에 대한 초기화 시퀀스를 완료합니다.
FinishSessionKeyRefresh 새 세션 키로 전환합니다. (ID3D11VideoContext.FinishSessionKeyRefresh)
FinishSessionKeyRefresh 새 세션 키로 전환합니다. (IDirect3DCryptoSession9.FinishSessionKeyRefresh)
플러시 스트림 싱크가 수신되어 아직 렌더링되지 않은 샘플을 삭제하도록 합니다.
플러시 멀티플렉서에 큐에 대기 중인 모든 출력 미디어 샘플을 처리하도록 신호를 전송합니다. 마지막 샘플을 멀티플렉서에 전달한 후 이 메서드를 호출합니다.

플러시 ASF(Advanced Systems Format) 분할자를 다시 설정하여 보류 중인 모든 샘플을 해제합니다.
플러시 스트림에서 사용하는 내부 버퍼를 지웁니다. 스트림에 쓰는 경우 버퍼링된 데이터는 기본 파일 또는 디바이스에 기록됩니다.
플러시 하나 이상의 스트림을 플러시합니다. (IMFSinkWriter.Flush)
플러시 하나 이상의 스트림을 플러시합니다. (IMFSourceReader.Flush)
ForceEndOfPresentation 재생이 세그먼트의 끝에 도달하면 소스에 알합니다. 타임라인의 경우 이는 마크아웃 지점에 도달하는 것에 해당합니다.
FrameStep 한 프레임 앞으로 또는 뒤로 이동합니다.
freeRepresentation IMFMediaType::GetRepresentation 메서드에 의해 할당된 메모리를 해제합니다.
generateHeader MFASFContentInfo 개체의 데이터를 이진 ASF(Advanced Systems Format) 헤더로 인코딩합니다.
GenerateIndexEntries 파일에 대한 ASF 패킷을 수락하고 인덱스 항목을 만듭니다.
GeneratePresentationDescriptor ASF 콘텐츠에 대한 프레젠테이션 설명자를 만듭니다.
generateRequest 제공된 데이터를 기반으로 라이선스 요청을 생성합니다.
GenerateRequiredSchemas 각 보호 시스템에 대한 구성 데이터와 함께 OTA(출력 신뢰 기관)에서 적용해야 하는 출력 보호 시스템의 목록을 검색합니다.
get_Keys 미디어 엔진과 연결된 미디어 키 개체를 가져오거나 미디어 키 개체가 없는 경우 null을 가져옵니다.
get_KeySystem

IMFMediaKeys 개체를 만든 키 시스템 문자열을 가져옵니다.
get_KeySystem 미디어 키 개체를 만든 키 시스템 이름의 이름을 가져옵니다.
get_SessionId 이 세션에 대해 만든 고유한 세션 ID를 가져옵니다.
GetAction 이 OTA(출력 신뢰 기관)에서 수행하는 작업을 검색합니다.
GetActiveSourceBuffers 미디어 원본에 미디어 데이터를 적극적으로 제공하는 원본 버퍼를 가져옵니다.
GetActiveTracks 시간 제한 텍스트 구성 요소의 활성 시간 제한 텍스트 트랙 목록을 가져옵니다.
GetActivityReport 지정된 인덱스를 기반으로 IMFSensorActivityReport를 검색합니다.
GetActivityReportByDeviceName 지정된 디바이스 이름을 기반으로 IMFSensorActivityReport를 검색합니다.
getAdvancedMediaCaptureSettings 고급 미디어 캡처 설정을 가져옵니다.
GetAllLanguages 메타데이터를 사용할 수 있는 언어 목록을 가져옵니다.
GetAllocatedBlob 키와 연결된 바이트 배열을 검색합니다. 이 메서드는 배열에 대한 메모리를 할당합니다.
getAllocatedString 키와 연결된 와이드 문자열을 가져옵니다. 이 메서드는 문자열에 대한 메모리를 할당합니다.
GetAllocatorUsage 지정된 출력 스트림에 대한 샘플 할당자 사용량을 검색합니다.
GetAllPropertyNames 이 개체의 모든 메타데이터 속성 이름 목록을 가져옵니다.
GetAllSettings GetAllSettings 메서드는 코덱의 현재 속성을 가져오고 스트림에 씁니다. (ICodecAPI::GetAllSettings)

GetAllVolumes 오디오 스트림의 모든 채널에 대한 볼륨 수준을 검색합니다.
GetAppendWindowEnd 추가 창의 끝에 대한 타임스탬프를 가져옵니다.
GetAppendWindowStart 추가 창의 시작에 대한 타임스탬프를 가져옵니다.
GetAtEndOfPayload Microsoft Media Foundation에서 호출하여 BeginReadPayload를 호출하여 응답의 메시지 본문에서 데이터를 읽어야 하는지 확인합니다.
GetAttributes 지정된 인덱스가 있는 하위 스트림의 IMFAttributes를 가져옵니다.
GetAttributes 이 MFT(Media Foundation 변환)에 대한 전역 특성 저장소를 가져옵니다.
GetAudioAttributes 현재 코드 변환 프로필에 설정된 오디오 스트림 설정을 가져옵니다.
GetAudioEndpointRole SetSource 또는 Load에 대한 다음 호출에 사용되는 오디오 디바이스 엔드포인트 역할을 가져옵니다.
GetAudioFormat Windows 7을 기준으로 GetAudioFormat을 더 이상 사용할 수 없습니다.
GetAudioStreamCategory SetSource 또는 Load에 대한 다음 호출에 사용되는 오디오 스트림 범주를 가져옵니다.
GetAutoPlay 미디어 엔진이 자동으로 재생을 시작하는지 여부를 쿼리합니다.
GetAvailableDeviceMediaType 캡처 스트림 중 하나에서 지원하는 형식을 가져옵니다.
GetBackgroundColor 영역의 배경색을 가져옵니다.
GetBackgroundColor 시간 제한 텍스트 스타일의 배경색을 가져옵니다.

GetBalance 오디오 균형을 가져옵니다.
GetBandwidthStep 대역폭 단계에 적용되는 스트림 번호를 검색합니다. 이 메서드는 MBR(다중 비트 전송률) 콘텐츠에 사용됩니다.
GetBandwidthStepCount 콘텐츠에 대해 존재하는 대역폭 단계 수를 검색합니다. 이 메서드는 MBR(다중 비트 전송률) 콘텐츠에 사용됩니다.
GetBlob 키와 연결된 바이트 배열을 검색합니다. 이 메서드는 배열을 호출자가 할당한 버퍼에 복사합니다.
GetBlobSize 키와 연결된 바이트 배열의 길이를 검색합니다.
GetBold 시간 제한 텍스트의 스타일이 굵게 표시되어 있는지 여부를 결정합니다.
GetBuffer DXVA(DirectX Video Acceleration) 디코더 버퍼에 대한 포인터를 검색합니다.
GetBufferCount 샘플의 버퍼 수를 검색합니다.
getBuffered 미디어 엔진이 버퍼링한 리소스 데이터의 양을 쿼리합니다.
getBuffered 버퍼링된 시간 범위를 가져옵니다.
GetBufferFullnessBits 이 릴리스에서는 구현되지 않았습니다. (IWMCodecLeakyBucket.GetBufferFullnessBits)
GetBufferSizeBits 버퍼의 현재 크기를 비트 단위로 검색합니다.
GetByteRanges 현재 캐시에 저장된 바이트 범위를 가져옵니다.
GetCapabilities 캡처 디바이스에서 지원하는 속성 기능에 대한 쿼리입니다.

GetCapabilities 바이트 스트림의 특성을 검색합니다.
GetCapabilities 콘텐츠에 대한 기능 정보를 가져옵니다.
GetCaptureDeviceActivate 현재 캡처 디바이스의 IMFActivate 개체 포인터를 가져옵니다.
GetCaptureDeviceSource 현재 캡처 디바이스의 IMFMediaSource 개체 포인터를 가져옵니다.
GetCertificate 드라이버의 인증서 체인을 가져옵니다. (ID3D11AuthenticatedChannel.GetCertificate)
GetCertificate 드라이버의 인증서 체인을 가져옵니다. (ID3D11CryptoSession.GetCertificate)
GetCertificate 드라이버의 인증서 체인을 가져옵니다. (IDirect3DAuthenticatedChannel9.GetCertificate)
GetCertificate 드라이버의 인증서 체인을 가져옵니다. (IDirect3DCryptoSession9.GetCertificate)
GetCertificate 클라이언트의 인증서를 검색합니다.
GetCertificatePolicy 서버 SSL 인증서를 호출자, Media Foundation 또는 IMFSSLCertificateManager 구현 클래스에서 확인해야 하는지 여부를 나타냅니다.
GetCertificateSize 드라이버의 인증서 체인 크기를 가져옵니다. (ID3D11AuthenticatedChannel.GetCertificateSize)
GetCertificateSize 드라이버의 인증서 체인 크기를 가져옵니다. (ID3D11CryptoSession.GetCertificateSize)
GetCertificateSize 드라이버의 인증서 체인 크기를 가져옵니다. (IDirect3DAuthenticatedChannel9.GetCertificateSize)
GetCertificateSize 드라이버의 인증서 체인 크기를 가져옵니다. (IDirect3DCryptoSession9.GetCertificateSize)

GetChainedLicenseResponse 이 릴리스에서는 구현되지 않았습니다. (IMFDRMNetHelper.GetChainedLicenseResponse)
GetChannelCount 오디오 스트림의 채널 수를 검색합니다.
GetChannelHandle 인증된 채널에 대한 핸들을 가져옵니다.
GetChannelVolume 오디오 스트림에서 지정된 채널의 볼륨 수준을 검색합니다.
GetCharacteristics 미디어 싱크의 특성을 가져옵니다.
GetCharacteristics 미디어 원본의 특성을 검색합니다.
GetClientCertificate 클라이언트 SSL 인증서를 동기적으로 가져옵니다.
GetClipOverflow 텍스트 클립이 영역으로 오버플로되었는지 여부를 확인합니다.
GetClock 미디어 세션의 프레젠테이션 시계를 검색합니다.
GetClockCharacteristics 클록의 특성을 검색합니다.
GetCodec 비디오 인코더와 연결된 코덱을 가져옵니다.
GetCodec 비디오 인코더 힙과 연결된 코덱을 가져옵니다.
GetCodecConfiguration 비디오 인코더와 연결된 코덱 구성 매개 변수를 가져옵니다.
GetCodecLevel 비디오 인코더 힙과 연결된 코덱 수준을 가져옵니다.

GetCodecProfile 비디오 인코더와 연결된 코덱 프로필을 가져옵니다.
GetCodecProfile 비디오 인코더 힙과 연결된 코덱 프로필을 가져옵니다.
GetCodecProp 출력 형식과 관련된 코덱 속성을 검색합니다.
GetColor 시간 제한 텍스트 스타일의 색을 가져옵니다.
GetConfiguration 지원되는 구성 옵션 조합을 반환합니다.
GetConfigurationData 출력 보호 시스템에 대한 구성 데이터를 반환합니다. 구성 데이터는 보호 시스템을 사용하거나 사용하지 않도록 설정하고 보호 수준을 설정하는 데 사용됩니다.
getContainerAttributes 현재 트랜스코드 프로필에 설정된 컨테이너 설정을 가져옵니다.
GetContentDesc 비디오 프로세서를 만드는 데 사용된 콘텐츠 설명을 가져옵니다.
GetContentProtectionCaps 드라이버에서 콘텐츠 보호 기능을 쿼리합니다.
GetContentProtectionCaps 디스플레이 드라이버의 콘텐츠 보호 기능을 쿼리합니다.
GetContext GetContext 메서드는 SetContext를 호출하여 이전에 목차와 연결된 바이트 블록을 검색합니다.
GetContiguousLength 버퍼의 콘텐츠를 연속 형식으로 저장하는 데 필요한 바이트 수를 검색합니다.
GetContinuityKey 클록의 연속성 키를 검색합니다. (지원되지 않음).
GetControl 지정된 인덱스(0부터 시작)를 사용하여 컬렉션에서 컨트롤을 가져옵니다.

GetControlCount 현재 컨트롤의 기본 컬렉션에 있는 컨트롤 수를 가져옵니다.
GetCorrelatedTime 시스템 시간과 상관 관계가 있는 마지막 시계 시간을 검색합니다.
GetCount 검색할 수 있는 IMFSensorActivityReport 개체의 수를 가져옵니다.
GetCount 이 개체에 설정된 특성 수를 검색합니다.
GetCreationParameters 디코더를 만드는 데 사용된 매개 변수를 가져옵니다.
GetCreationParameters 이 디바이스를 만드는 데 사용된 매개 변수를 검색합니다. (IDirectXVideoDecoder.GetCreationParameters)
GetCreationParameters 이 디바이스를 만드는 데 사용된 매개 변수를 검색합니다. (IDirectXVideoProcessor.GetCreationParameters)
GetCredential 지정된 URL에 대한 자격 증명 개체를 검색합니다.
GetCrossOriginPolicy 다운로드 세션에 적용할 클라이언트의 현재 원본 간 정책을 반환합니다.
getCryptoSessionHandle 암호화 세션에 대한 핸들을 가져옵니다.
GetCryptoSessionPrivateDataSize 프라이빗 드라이버 데이터의 선택적 크기를 검색합니다.
GetCryptoType 이 세션에서 지원하는 암호화 유형을 가져옵니다.
GetCueKind 시간 제한 텍스트 큐의 종류를 가져옵니다.
GetCueTimeOffset 큐 시간에 대한 오프셋을 가져옵니다.

GetCurrentDeviceMediaType 캡처 스트림의 현재 미디어 형식을 가져옵니다.
GetCurrentLength 버퍼에서 유효한 데이터의 길이를 검색합니다.
GetCurrentMediaType 개체의 현재 미디어 형식을 검색합니다.
GetCurrentMediaType 스트림의 현재 미디어 형식을 가져옵니다.
GetCurrentPosition 스트림에서 현재 읽기 또는 쓰기 위치를 검색합니다.
GetCurrentProxy 호스트 이름 및 포트를 포함하여 현재 프록시 정보를 검색합니다.
GetCurrentSource 현재 미디어 리소스의 URL을 가져오거나 미디어 리소스가 없는 경우 빈 문자열을 가져옵니다.
GetCurrentTime 현재 재생 위치를 가져옵니다. (IMFMediaEngine.GetCurrentTime)
GetData 시스템 ID 데이터를 검색합니다.
GetData 시간 제한 텍스트 개체의 데이터 콘텐츠를 가져옵니다.
GetData 시간 제한 텍스트 큐의 데이터 콘텐츠를 가져옵니다.
GetDataFormat 트랙의 기본 데이터 형식을 식별하는 GUID를 가져옵니다.
GetDataForNewHardwareKey 드라이버는 새 하드웨어 키를 초기화할 때 사용되는 IHV 특정 정보를 반환할 수 있습니다.
GetDecoderBuffer 디코더 버퍼에 대한 포인터를 가져옵니다.

GetDecoderConfigurations 디코더 디바이스에 사용할 수 있는 구성을 가져옵니다.
GetDecoderDeviceGuids 그래픽 하드웨어에서 지원하는 디코더 디바이스를 식별하는 GUID 배열을 검색합니다.
GetDecoderProfile 세션의 디코딩 프로필을 가져옵니다.
GetDecoderRenderTargets 지정된 디코더 디바이스에 대해 지원되는 렌더링 대상을 검색합니다.
GetDecrypter 암호 해독기 변환을 검색합니다.
GetDefaultPlaybackRate 기본 재생 속도를 가져옵니다.
GetDefaultValue GetDefaultValue 메서드는 코덱 속성의 기본값을 가져옵니다. (ICodecAPI::GetDefaultValue)
GetDesc 비디오 디코더 출력 뷰의 속성을 가져옵니다.
GetDesc 비디오 프로세서 입력 보기의 속성을 가져옵니다.
GetDesc 비디오 프로세서 출력 뷰의 속성을 가져옵니다.
GetDesc ID3D12VideoDecoder를 만들 때 ID3D12VideoDevice::CreateVideoDecoder에 전달된 D3D12_VIDEO_DECODER_DESC 구조를 가져옵니다.
GetDesc ID3D12VideoDecoderHeap을 만들 때 ID3D12VideoDevice::CreateVideoDecoderHeap에 전달된 D3D12_VIDEO_DECODER_HEAP_DESC 구조를 가져옵니다.
GetDesc 인터페이스를 만들 때 제공된 D3D12_VIDEO_EXTENSION_COMMAND_DESC 가져옵니다.
GetDesc

ID3D12VideoMotionEstimator를 만들 때 ID3D12VideoDevice1::CreateVideoMotionEstimator)에 전달된 D3D12_VIDEO_MOTION_ESTIMATOR_DESC 구조를 가져옵니다.
GetDesc ID3D12VideoMotionEstimatorHeap을 만들 때 ID3D12VideoDevice1::CreateVideoMotionEstimatorHeap에 전달된 D3D12_VIDEO_MOTION_VECTOR_HEAP_DESC 구조를 가져옵니다.
GetDescription GetDescription 메서드는 목차의 SetDescription에 대한 이전 호출에 의해 설정된 설명을 검색합니다.
GetDescription 출력 형식에 대한 설명을 검색합니다.
GetDescriptionData GetDescriptionData 메서드는 SetDescriptionData를 호출하여 이전에 항목과 연결된 설명 데이터 블록을 가져옵니다.
GetDescriptor GetDescriptor 메서드는 목차의 설명자(이전에 SetDescriptor에서 설정)를 검색합니다.
GetDescriptor GetDescriptor 메서드는 이전에 SetDescriptor를 호출하여 설정한 설명자를 검색합니다.
GetDevice 이미지 공유 디바이스에 대한 정보를 가져옵니다.
GetDevice 미디어 공유 디바이스에 대한 정보를 가져옵니다.
GetDeviceAttributes 센서 그룹에 대한 IMFAttributes를 가져옵니다.
GetDeviceId 디바이스의 고유 식별자를 가져옵니다. 이 값은 현재 사용되지 않습니다.
GetDeviceStreamCategory 지정된 원본 스트림 인덱스 스트림 범주를 가져옵니다.
GetDeviceStreamCount 디바이스 스트림 수를 가져옵니다.
GetDeviceType 개체가 나타내는 센서 디바이스의 유형을 지정하는 값을 가져옵니다.

GetDirectxDeviceManager DirectX 디바이스 관리자를 가져옵니다.
GetDisplayAlignment 영역의 표시 맞춤을 가져옵니다.
GetDisplayName 오디오 세션의 표시 이름을 검색합니다. Windows 볼륨 컨트롤에 이 이름이 표시됩니다.
GetDouble 키와 연결된 이중 값을 검색합니다.
GetDriverHandle 드라이버에 대한 핸들을 가져옵니다.
GetDropMode 현재 드롭 모드를 검색합니다.
getDuration 미디어 리소스의 기간을 가져옵니다.
getDuration 미디어 원본의 기간을 100나노초 단위로 가져옵니다.
getDuration 트랙에서 큐의 기간 시간을 가져옵니다.
GetElement 컬렉션에서 개체를 검색합니다.
GetElementCount 컬렉션의 개체 수를 검색합니다. (IMFCollection.GetElementCount)
GetEnableData 수동 콘텐츠 사용 작업에 대한 데이터를 검색합니다.
GetEnableType 이 콘텐츠 인에이블러가 수행하는 작업의 유형을 검색합니다.
GetEnableURL 수동 콘텐츠 사용 작업을 수행하기 위한 URL을 검색합니다.

GetEncoderFlags 비디오 인코더가 초기화된 인코더 플래그를 가져옵니다.
GetEncoderHeapFlags 비디오 인코더 힙이 초기화된 인코더 힙 플래그를 가져옵니다.
GetEncodingConfigurationPropertyStore 인코딩 속성을 설정하는 데 사용할 수 있는 속성 저장소를 검색합니다.
GetEncryptionBltKey ID3D11VideoContext::EncryptionBlt 메서드에서 반환된 데이터를 해독하는 암호화 키를 가져옵니다.
GetEncryptionBltKey IDirect3DCryptoSession9::EncryptionBlt 메서드에서 반환된 데이터를 해독하는 데 사용되는 암호화 키를 가져옵니다.
GetEnd 지정된 시간 범위의 종료 시간을 가져옵니다.
GetEntryCount GetEntryCount 메서드는 컬렉션의 목차 수를 검색합니다.
GetEntryCount GetEntryCount 메서드는 목록의 항목 수를 검색합니다.
GetEntryListCount GetEntryListCount 메서드는 목차의 항목 목록 수를 검색합니다.
GetError 가장 최근의 오류 상태를 가져옵니다.
GetError 미디어 키 세션과 연결된 오류 상태를 가져옵니다.
GetErrorCode 오류 코드를 가져옵니다.
GetErrorCode 트랙과 연결된 최신 오류의 오류 유형을 나타내는 값을 가져옵니다.
GetEvent 큐에서 다음 이벤트를 검색합니다. 이 메서드는 동기적입니다.

GetEvent 큐에서 다음 이벤트를 검색합니다. 이 메서드는 동기적입니다. IMFMediaEventGenerator::GetEvent 구현 내에서 이 메서드를 호출합니다. 해당 메서드의 매개 변수를 이 메서드로 직접 전달합니다.
GetExpiration CDM 세션의 모든 키에 대한 만료 시간을 가져옵니다.
GetExtendedCameraControl 앱이 현재 캡처 디바이스의 확장 속성 컨트롤을 가져올 수 있도록 하는 IMFExtendedCameraControl의 인스턴스를 가져옵니다.
GetExtendedErrorCode 확장 오류 코드를 가져옵니다.
GetExtendedErrorCode 트랙과 연결된 최신 오류에 대한 확장 오류 코드를 가져옵니다.
GetExtendedType 이벤트의 확장 형식을 검색합니다.
GetExtent 영역의 범위를 가져옵니다.
GetFastestRate 개체에서 지원하는 가장 빠른 재생 속도를 가져옵니다.
GetFilterPropertyRange 이 디바이스에서 지원하는 이미지 필터의 값 범위를 검색합니다.
GetFilterPropertyRange 비디오 프로세서 디바이스에서 지원하는 이미지 필터의 값 범위를 검색합니다.
GetFlags 캡처 디바이스의 속성 플래그에 대한 쿼리입니다.
GetFlags 센서 디바이스에 대해 설정된 플래그를 가져옵니다. 이 메서드는 나중에 사용하도록 예약되어 있습니다.
GetFlags 센서 그룹에 대해 설정된 플래그를 가져옵니다.
GetFlags

선택한 인덱서 옵션을 나타내는 플래그를 검색합니다.
GetFlags 구성된 멀티플렉서 옵션을 나타내는 플래그를 검색합니다.
GetFlags ASF 분할자에서 설정된 옵션 플래그를 검색합니다.
GetFontFamily 시간 제한 텍스트 스타일의 글꼴 패밀리를 가져옵니다.
GetFontSize 시간 제한 텍스트 스타일의 글꼴 크기를 가져옵니다.
GetFontStyle 시간 제한 텍스트 스타일의 글꼴 스타일을 가져옵니다.
GetFormatProp 출력 미디어 형식의 형식 속성을 검색합니다. 이 메서드를 사용하여 열거된 오디오 형식에 대한 정보를 가져옵니다.
GetFreeSampleCount 현재 사용할 수 있는 비디오 샘플 수를 가져옵니다.
GetFriendlyName 보고서와 연결된 센서의 이름을 가져옵니다.
GetFullCropRegion 원본 및 대상 사각형을 검색합니다.
GetFullTopology 미디어 세션에서 토폴로지 가져옵니다.
GetGroupingParam 이 오디오 세션이 속한 세션 그룹을 검색합니다.
GetGUID 키와 연결된 GUID 값을 검색합니다.
GetHeaderSize ASF(Advanced Systems Format) 파일의 헤더 섹션 크기를 검색합니다.
GetHttpStatus

Microsoft Media Foundation에서 호출하여 서버가 응답에 지정한 HTTP 상태 코드를 검색합니다. Media Foundation은 EndReceiveResponse를 성공적으로 호출한 후 이 메서드를 호출합니다.
GetHurryup 비디오 디코더의 현재 속도 모드를 검색합니다.
GetIconPath 오디오 세션의 아이콘 리소스를 검색합니다. Windows 볼륨 컨트롤에 이 아이콘이 표시됩니다.
GetId 시간 제한 텍스트 큐의 식별자를 가져옵니다.
GetId 시간 제한 텍스트 트랙의 식별자를 가져옵니다.
GetID 버퍼가 나타내는 공간 오디오 개체의 서명되지 않은 고유한 32비트 ID를 반환합니다.
GetIdentifier 이 스트림 싱크에 대한 스트림 식별자를 검색합니다.
GetInBandMetadataTrackDispatchType 트랙의 대역 내 메타데이터를 가져옵니다.
GetIndexByteStreamCount 인덱서 개체에서 사용 중인 바이트 스트림 수를 검색합니다.
GetIndexPosition 콘텐츠의 시작 부분부터 인덱스 개체의 오프셋을 검색합니다.
GetIndexStatus 지정된 스트림 및 인덱스 형식에 대한 인덱스 설정을 검색합니다.
GetIndexWriteSpace 완료된 인덱스를 저장하는 데 필요한 버퍼의 크기(바이트)를 검색합니다.
GetInformation 표시 드라이버에 OPM(출력 보호 관리자) 상태 요청을 보냅니다.
GetInput 이 노드에서 지정된 입력 스트림에 연결된 노드를 검색합니다.
GetInputAvailableType

이 MFT(Media Foundation 변환)에서 입력 스트림에 사용할 수 있는 미디어 형식을 가져옵니다.
GetInputCount 이 노드에 현재 존재하는 입력 스트림의 수를 검색합니다.
GetInputCurrentType 이 MFT(Media Foundation 변환)에서 입력 스트림의 현재 미디어 형식을 가져옵니다.
GetInputFormat 비디오 인코더의 입력 형식을 가져옵니다.
GetInputPrefType 이 노드의 입력 스트림에 대한 기본 미디어 형식을 검색합니다.
GetInputProtectionVersion 구성 요소가 입력 시 구현하는 샘플 보호 버전을 검색합니다.
GetInputStatus 이 MFT(Media Foundation 변환)의 입력 스트림이 더 많은 데이터를 허용할 수 있는지 여부를 쿼리합니다.
GetInputStreamAttributes 이 MFT(Media Foundation 변환)에서 입력 스트림의 특성 저장소를 가져옵니다.
GetInputStreamDescs ID3D12VideoDevice::CreateVideoProcessor를 호출하여 비디오 프로세서를 만들 때 제공된 입력 스트림 설명을 가져옵니다.
GetInputStreamInfo 이 MFT(Media Foundation 변환)의 입력 스트림에 대한 버퍼 요구 사항 및 기타 정보를 가져옵니다.
GetInputTrustAuthority 지정된 스트림에 대한 ITA(입력 신뢰 기관)를 검색합니다.
GetItem 키와 연결된 값을 검색합니다.
GetItemType 키와 연결된 값의 데이터 형식을 검색합니다.
GetKeyStatuses CDM(Content Decryption Module) 세션에 알려진 키 ID와 연결된 키의 현재 상태를 나타내는 구조체 배열에 대한 참조를 가져옵니다.

GetKeySystem 지정된 소스 요소 인덱스 키 시스템을 가져옵니다.
GetKeySystem CDM(콘텐츠 암호 해독 모듈)에서 사용되는 키 시스템을 식별하는 문자열을 가져옵니다.
GetLabel 트랙의 레이블을 가져옵니다.
GetLanguage 현재 언어 설정을 가져옵니다.
GetLanguage 트랙의 언어를 가져옵니다.
GetLastSendTime 받은 마지막 샘플의 전송 시간을 검색합니다.
GetLength 스트림의 길이를 검색합니다.
GetLength 목록의 원본 요소 수를 가져옵니다.
GetLength 개체에 포함된 시간 범위의 수를 가져옵니다.
GetLength 목록의 IMFSourceBuffer 개체 수를 가져옵니다.
GetLength 시간 제한 텍스트 트랙 목록의 길이(트랙)를 가져옵니다.
GetLine 줄의 인덱스에서 큐의 텍스트 줄을 가져옵니다.
GetLineCount 시간 제한 텍스트 큐의 텍스트 줄 수를 가져옵니다.
GetLineHeight 영역에 있는 각 텍스트 줄의 높이를 가져옵니다.

GetLoop 미디어 엔진이 재생을 반복할지 여부를 쿼리합니다.
GetMajorType 개체의 주 미디어 형식을 가져옵니다.
GetMajorType 형식의 주 형식을 가져옵니다.
GetManager Microsoft Media Foundation 비디오 렌더링 싱크에서 IMFDXGIDeviceManager를 가져옵니다.
GetMasterVolume 마스터 볼륨 수준을 검색합니다.
GetMaximumDropMode 최대 드롭 모드를 가져옵니다.
GetMaxLength 버퍼의 할당된 크기를 검색합니다.
getMaxMotionEstimationPrecision 비디오 인코더의 최대 동작 예측 정밀도를 가져옵니다.
getMaxNumberOfBytesRequiredForResolution 미디어 원본을 만드는 데 필요한 최대 바이트 수를 검색하거나 바이트 스트림 처리기가 이 스트림을 구문 분석할 수 없는지 확인합니다.
GetMedia 목록에 있는 요소의 의도한 미디어 형식을 가져옵니다.
GetMediaSink 이 스트림 싱크를 소유하는 미디어 싱크를 검색합니다.
GetMediaSource 이 미디어 스트림을 만든 미디어 원본에 대한 포인터를 검색합니다.
GetMediaSource 가상 카메라에서 미디어 데이터를 제공하는 IMFMediaSource를 가져옵니다.
GetMediaSourceTopology 내부 토폴로지 빌드 미디어 소스에 대한 토폴로지 반환

GetMediaType 센서 스트림에 대해 지원되는 미디어 형식을 나타내는 IMFMediaType을 검색합니다.
GetMediaType 스트림의 미디어 형식을 검색합니다.
GetMediaType 지정된 인덱스가 있는 하위 스트림의 IMFMediaType을 가져옵니다.
GetMediaTypeCount 개체의 지원되는 미디어 형식 목록에서 미디어 형식의 수를 검색합니다.
GetMediaTypeCount 센서 스트림에서 지원하는 미디어 형식의 수를 가져옵니다.
GetMediaTypeHandler 스트림에 대한 미디어 형식 처리기를 검색합니다. 미디어 형식 처리기를 사용하여 스트림에 대해 지원되는 미디어 형식을 열거하고, 현재 미디어 형식을 가져와서 미디어 형식을 설정할 수 있습니다.
GetMediaTypeHandler 스트림 싱크에 대한 미디어 형식 처리기를 검색합니다. 미디어 형식 처리기를 사용하여 스트림이 지원하는 형식을 찾고 스트림에서 미디어 형식을 설정할 수 있습니다.
GetMetadataItems 공간 오디오 메타데이터를 포함할 수 있는 버퍼에 대한 포인터를 검색합니다.
GetMetadataTracks 시간 제한 텍스트 구성 요소의 시간 제한 메타데이터 트랙 목록을 가져옵니다.
GetMFMetadata 전체 프레젠테이션 또는 프레젠테이션의 한 스트림에 대한 메타데이터 컬렉션을 가져옵니다.
GetMinimumGRLVersion 이 정책에 대해 보호된 환경에서 적용해야 하는 GRL(전역 해지 목록)의 최소 버전을 검색합니다.
GetMinimumQualityLevel 구성 요소에서 지원하는 최소 품질 수준을 가져옵니다.
GetMirrorState 비디오 미리 보기 스트림의 현재 미러링 상태를 가져옵니다. (IMFCapturePreviewSink.GetMirrorState)
GetMirrorState

비디오 미리 보기 스트림의 현재 미러링 상태를 가져옵니다. (IMFCaptureSource.GetMirrorState)
GetMute 오디오가 음소거되었는지 여부를 쿼리합니다. (IMFSimpleAudioVolume.GetMute)
getMuted 오디오가 음소거되었는지 여부를 쿼리합니다. (IMFMediaEngine.GetMuted)
GetMutualExclusion 프로필에서 ASF(Advanced Systems Format) 상호 제외 개체를 검색합니다.
GetMutualExclusionCount 프로필과 연결된 ASF(Advanced Systems Format) 상호 제외 개체의 수를 검색합니다.
GetName 코덱의 이름을 검색합니다.
GetName 지역의 이름을 가져옵니다.
GetName 시간 제한 텍스트 스타일의 이름을 가져옵니다.
GetNativeMediaType 미디어 원본에서 기본적으로 지원되는 형식을 가져옵니다.
GetNativeVideoSize 가로 세로 비율에 맞게 조정된 비디오 프레임의 크기를 가져옵니다.
GetNearestKeyFrames 특정 검색 위치의 경우 가장 가까운 두 키 프레임을 가져옵니다. (IMFSeekInfo.GetNearestKeyFrames)
GetNetworkState 미디어 엔진의 현재 네트워크 상태를 가져옵니다.
GetNextOutputTime 예정된 출력 샘플의 타임스탬프를 디코더에 쿼리합니다. IMediaObject::P rocessOutput 또는 IMFTransform::P rocessOutput을 호출하기 전에 샘플의 시간을 알아야 하는 경우 이 메서드를 사용합니다.
GetNextPacket 멀티플렉서에서 다음 출력 ASF 패킷을 검색합니다.
GetNextSample

데이터를 구문 분석한 후 ASF(Advanced Systems Format) 분할자에서 샘플을 검색합니다.
GetNode 인덱스로 지정된 토폴로지의 노드를 가져옵니다.
GetNodeByID 노드 식별자에 지정된 토폴로지의 노드를 가져옵니다.
GetNodeCount 토폴로지의 노드 수를 가져옵니다.
GetNodeMask 비디오 인코더의 노드 마스크를 가져옵니다.
GetNodeMask 비디오 인코더 힙의 노드 마스크를 가져옵니다.
GetNodeMask ID3D12VideoDevice::CreateVideoProcessor를 호출하여 비디오 프로세서를 만들 때 제공된 노드 마스크를 가져옵니다.
GetNodeType 노드 형식을 검색합니다.
GetNumberOfStreams 미디어 리소스의 스트림 수를 가져옵니다.
GetNumberOfSupportedProtocols 네트워크 구성표 플러그 인에서 지원하는 프로토콜 수를 검색합니다.
GetNumInputStreamDescs ID3D12VideoDevice::CreateVideoProcessor를 호출하여 비디오 프로세서를 만들 때 제공된 입력 스트림 설명의 수를 가져옵니다.
GetObject 이 노드와 연결된 개체를 가져옵니다.
GetObject 비동기 작업과 연결된 개체를 반환합니다. 개체의 형식(있는 경우)은 호출된 비동기 메서드에 따라 달라집니다. (IMFAsyncResult.GetObject)
GetObjectCount 샘플에서 IMFSpatialAudioObjectBuffer 개체로 표시되는 공간 오디오 개체의 수를 가져옵니다.

GetObjectPointer 비동기 콜백이 연결된 부모 개체에 대한 포인터를 가져옵니다.
GetObjectTag 비동기 콜백이 연결된 개체인 부모 개체의 태그를 가져옵니다.
GetOPMInfo OPM(출력 보호 관리자)에 대한 상태 정보를 가져옵니다.
GetOrAddControl 카메라 컨트롤 컬렉션에 새 사용자 지정 카메라 컨트롤을 추가합니다.
GetOrAddExtendedControl 카메라 컨트롤 컬렉션에 새로운 확장 카메라 컨트롤을 추가합니다.
GetOriginalId 사용 가능한 경우 텍스트 트랙 데이터 형식으로 제공되는 큐 식별자를 가져옵니다.
GetOriginatorID 이 출력 정책 개체를 만든 ITA(입력 신뢰 기관)를 식별하는 GUID를 검색합니다.
GetOriginatorID 이 출력 스키마 개체를 생성한 ITA(입력 신뢰 기관)를 식별하는 GUID를 검색합니다.
GetOutput 이 노드에서 지정된 출력 스트림에 연결된 노드를 검색합니다.
GetOutputAvailableType 이 MFT(Media Foundation 변환)에서 출력 스트림에 사용할 수 있는 미디어 형식을 가져옵니다.
GetOutputCount 이 노드에 현재 존재하는 출력 스트림의 수를 검색합니다.
GetOutputCount ASF(Advanced Systems Format) 콘텐츠의 출력 수를 검색합니다.
GetOutputCurrentType 이 MFT(Media Foundation 변환)에서 출력 스트림의 현재 미디어 형식을 가져옵니다.
GetOutputFromStream 스트림과 연결된 출력 번호를 검색합니다.

GetOutputMediaType 이 캡처 싱크에서 스트림의 출력 형식을 가져옵니다.
GetOutputMutex 출력에 대한 상호 제외 개체를 검색합니다.
GetOutputMutexCount 출력과 연결된 상호 제외 개체의 수를 검색합니다.
GetOutputNodeCollection 토폴로지의 출력 노드를 가져옵니다.
GetOutputOverride 스트림에 대해 설정된 수동 출력 재정의 선택을 검색합니다.
GetOutputPrefType 이 노드의 출력 스트림에 대한 기본 미디어 형식을 검색합니다.
GetOutputProtectionVersion 구성 요소가 출력에 구현하는 샘플 보호 버전을 검색합니다.
GetOutputStatus MFT(Media Foundation 변환)가 출력 데이터를 생성할 준비가 되었는지 여부를 쿼리합니다.
GetOutputStreamAttributes 이 MFT(Media Foundation 변환)의 출력 스트림에 대한 특성 저장소를 가져옵니다.
GetOutputStreamCount 출력과 연결된 스트림 수를 검색합니다.
GetOutputStreamDesc ID3D12VideoDevice::CreateVideoProcessor를 호출하여 비디오 프로세서를 만들 때 제공된 출력 스트림 설명을 가져옵니다.
GetOutputStreamInfo 이 MFT(Media Foundation 변환)의 출력 스트림에 대한 버퍼 요구 사항 및 기타 정보를 가져옵니다.
GetOutputStreamNumbers 출력과 연결된 모든 스트림의 스트림 번호를 검색합니다.
GetOutputTrustAuthorityCount 이 신뢰할 수 있는 출력에서 제공하는 OTA(출력 신뢰 기관)의 수를 가져옵니다. 각 OTA는 단일 작업을 보고합니다.

GetPadding 영역을 둘러싸는 안쪽 여백을 가져옵니다.
GetParameterRange GetParameterRange 메서드는 코덱 속성의 값 범위를 가져옵니다. (ICodecAPI::GetParameterRange)
GetParameters 콜백을 위해 디스패치 스레드에 구성 정보를 제공합니다. (IMFAsyncCallback.GetParameters)
GetParameterValues GetParameterValues 메서드는 코덱 속성에 사용할 수 있는 값 목록을 가져옵니다. (ICodecAPI::GetParameterValues)
GetPassword 암호를 검색합니다.
GetPayloadExtension 기존 페이로드 확장에 대한 정보를 검색합니다.
GetPayloadExtensionCount 스트림에 대해 구성된 페이로드 확장 수를 검색합니다.
getPlaftormWorkQueueMMCSSClass 지정된 플랫폼 작업 큐에 대한 MMCSS(멀티미디어 클래스 Scheduler Service) 클래스를 검색합니다.
getPlatformWorkQueueMMCSSPriority 지정된 플랫폼 작업 큐와 연결된 MMCSS(멀티미디어 클래스 스케줄러 서비스) 우선 순위의 우선 순위를 가져옵니다.
GetPlatformWorkQueueMMCSSTaskId 지정된 플랫폼 작업 큐에 대한 MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업 식별자를 검색합니다.
GetPlaybackRate 현재 재생 속도를 가져옵니다. (IMFMediaEngine.GetPlaybackRate)
GetPlayed 렌더링된 시간 범위를 가져옵니다.
GetPolicy 이 스트림에 허용되는 출력 보호 시스템과 각 보호 시스템의 구성 데이터를 정의하는 정책을 검색합니다.
GetPosition 영역의 위치를 가져옵니다.

GetPreferredClsid 기본 설정 목록에서 지정된 키 이름과 일치하는 CLSID(클래스 식별자)를 검색합니다.
GetPreload 미리 로드 플래그를 가져옵니다.
GetPresentationAttribute 미디어 리소스에서 프레젠테이션 특성을 가져옵니다.
GetPresentationAttribute 기본 미디어 원본에서 특성을 가져옵니다.
GetPresentationClock 미디어 파이프라인에서 IMFPresentationClock 인스턴스를 가져오기 위해 호출됩니다.
GetPresentationClock 미디어 싱크에 설정된 프레젠테이션 시계를 가져옵니다.
GetPresentationContext 프레젠테이션 설명자를 연결된 sequencer 요소 식별자 및 프레젠테이션이 나타내는 토폴로지에 매핑합니다.
GetPrivateData IWMCodecPrivateData::SetPartialOutputType 메서드를 사용하여 전달된 출력 형식을 기반으로 비디오 콘텐츠의 코덱 데이터를 검색합니다.
GetPrivateDataByteCount InvokeFunction 메서드를 통해 보안 프로세서를 호출할 때 입력 및 출력 버퍼 앞에 추가해야 하는 필요한 바이트 수를 가져옵니다.
GetProcAmpRange 이 비디오 프로세서 디바이스에서 비디오 프로세서(ProcAmp) 설정의 값 범위를 검색합니다.
GetProcAmpRange 비디오 프로세서(ProcAmp) 설정의 값 범위를 가져옵니다.
GetProcedureAddress 서명된 라이브러리에서 지정된 함수의 프로시저 주소를 가져옵니다.
GetProcessActivity 센서의 현재 프로세스 작업을 나타내는 IMFSensorProcessActivity 개체를 가져옵니다.
GetProcessCount

검색할 수 있는 센서와 연결된 프로세스의 현재 활동을 나타내는 IMFSensorProcessActivity 개체의 수를 가져옵니다.
GetProcessId 작업이 연결된 프로세스의 ID를 가져옵니다.
GetProfile 지정된 프로필을 검색합니다.
GetProfile ASF 콘텐츠를 설명하는 ASF(고급 시스템 형식) 프로필을 검색합니다.
GetProfileId 센서 프로필 ID를 검색합니다.
GetProgress 제공된 바이트 스트림에 저장된 콘텐츠의 백분율을 검색합니다.
GetProperties 클록의 속성을 검색합니다.
GetProperty 메타데이터 속성의 값을 가져옵니다.
GetProtectedResourceSession ID3D12VideoDecoder1을 만들 때 ID3D12VideoDevice2::CreateVideoDecoder1에 전달된 ID3D12ProtectedResourceSession을 가져옵니다.
GetProtectedResourceSession ID3D12VideoDecoderHeap1을 만들 때 ID3D12VideoDevice2::CreateVideoDecoderHeap1에 전달된 ID3D12ProtectedResourceSession을 가져옵니다.
GetProtectedResourceSession ID3D12VideoExtensionCommand를 만들 때 ID3D12VideoDevice2::CreateVideoExtensionCommand에 전달된 ID3D12ProtectedResourceSession을 가져옵니다.
GetProtectedResourceSession ID3D12VideoMotionEstimator를 만들 때 ID3D12VideoDevice1::CreateVideoMotionEstimator에 전달된 ID3D12ProtectedResourceSession을 가져옵니다.
GetProtectedResourceSession ID3D12VideoMotionEstimatorHeap을 만들 때 ID3D12VideoDevice1::CreateVideoMotionEstimatorHeap에 전달된 ID3D12ProtectedResourceSession을 가져옵니다.

GetProtectedResourceSession ID3D12VideoProcessor1을 만들 때 ID3D12VideoDevice2::CreateVideoProcessor1에 전달된 ID3D12ProtectedResourceSession을 가져옵니다.
GetProtectionCertificate 샘플 보호 인증서를 검색합니다.
GetProtectionSystemIds IMFContentDecryptionModule::GetProtectionSystemIds는 IMFContentDecryptionModule이 지원하는 SystemID 목록을 가져옵니다.
GetQualityLevel 현재 품질 수준을 검색합니다.
GetRangeEndOffset 현재 응답에서 마지막 바이트의 오프셋을 검색하기 위해 Microsoft Media Foundation에서 호출하며 리소스 시작부터 계산됩니다. 이는 요청이 HTTP "Range" 헤더를 사용하여 리소스의 일부만 다운로드하는 경우에 유용합니다.
GetRangeInfo 카메라 컨트롤에 허용되는 범위, 단계 값 및 기본값에 대한 정보를 가져옵니다.
GetRate 현재 재생 속도를 가져옵니다. (IMFRateControl.GetRate)
GetRateConversionCaps 비디오 프로세서의 속도 변환 기능을 가져옵니다.
GetReadyState 현재 미디어 리소스를 렌더링할 수 있는지 여부를 나타내는 준비 상태를 가져옵니다.
GetReadyState 미디어 원본의 준비 상태를 가져옵니다.
GetRealTimeMode SetSource 또는 Load에 대한 다음 호출에 사용되는 실시간 모드를 가져옵니다.
GetReconstructedVideoFrame 참고: 이 메서드는 사용되지 않으므로 사용하면 안 됩니다. 현재 재구성된 비디오 프레임을 검색합니다.
getReconstructedVideoFrameSize 참고: 이 메서드는 사용되지 않으므로 사용하면 안 됩니다. 현재 재구성된 비디오 프레임의 크기를 검색합니다.
GetRecordCount

Advanced Systems Format 상호 제외 개체의 레코드 수를 검색합니다.
getRegion 시간 제한 텍스트 큐의 표시 영역에 대한 정보를 가져옵니다.
GetRelativePanel 캡처 디바이스의 위치를 나타내는 ACPI_PLD_PANEL 열거형에서 값을 가져옵니다.
GetRemoteHost 이 프록시를 호스팅하는 개체에 대한 포인터를 검색합니다.
GetRemoteObject 이 개체가 프록시인 원격 개체에 대한 포인터를 검색합니다.
GetReport 상대 패널 위치를 나타내는 IMFRelativePanelReport 인터페이스를 가져옵니다.
GetReportTime 센서 활동 보고서와 연결된 시간을 가져옵니다.
GetRepresentation 미디어 형식의 대체 표현을 검색합니다. 현재 DirectShow AM_MEDIA_TYPE 구조만 지원됩니다.
GetRequiredProtections 프레임 서버 모드에서 적용해야 하는 콘텐츠 보호를 가져옵니다.
GetResolutionList 비디오 인코더 힙과 연결된 확인 목록을 가져옵니다.
GetResolutionListCount ID3D12VideoEncoderHeap::GetResolutionListCount 메서드(d3d12video.h)는 비디오 인코더 힙과 연결된 확인 목록 수를 가져옵니다.
GetResource 인터페이스에 대한 Microsoft DXGI(DirectX Graphics Infrastructure) 화면을 쿼리합니다.
GetResourceCharacteristics 미디어 리소스를 설명하는 다양한 플래그를 가져옵니다.
GetRightToLeft 시간 제한 텍스트 스타일의 오른쪽에서 왼쪽 쓰기 모드를 사용할 수 있는지 여부를 결정합니다.
GetRotation

비디오 미리 보기 스트림의 회전을 가져옵니다.
GetRotation 현재 녹화된 비디오 스트림에 적용되는 회전을 가져옵니다.
getSample 지정된 인덱스를 사용하여 하위 스트림과 연결된 IMFSample을 가져옵니다.
getSampleDuration 샘플의 기간을 검색합니다.
GetSampleFlags 샘플과 연결된 플래그를 검색합니다. 현재 플래그가 정의되어 있지 않습니다.
GetSampleTime 샘플의 프레젠테이션 시간을 검색합니다.
getScanline0AndPitch 버퍼 메모리 및 표면 보폭에 대한 포인터를 검색합니다.
GetSchemaType 이 개체가 나타내는 출력 보호 시스템을 검색합니다. 출력 보호 시스템은 GUID 값으로 식별됩니다.
GetScrollMode 영역의 스크롤 모드를 가져옵니다.
GetSeekable 미디어 엔진이 현재 검색할 수 있는 시간 범위를 가져옵니다.
GetSeekPositionForValue 원하는 검색 시간이 지정된 경우 클라이언트가 데이터 읽기를 시작해야 하는 오프셋을 가져옵니다.
GetSelectedStreams 현재 선택한 스트림의 목록을 가져옵니다.
getSelectedStyle SAMI 미디어 원본에서 현재 스타일을 가져옵니다.
GetSensorDevice 센서 그룹의 디바이스에 해당하는 IMFSensorDevice를 가져옵니다.
GetSensorDeviceCount 센서 그룹에 의해 가상화된 디바이스 수를 가져옵니다.

GetSensorDeviceMode 컨트롤러 또는 공유 센서 디바이스의 현재 공유 모드를 지정하는 값을 가져옵니다.
GetSensorGroupAttributes 센서 그룹에 대한 IMFAttributes를 가져옵니다. 반환된 개체는 내부 특성 저장소에 대한 라이브 참조입니다.
GetService 서비스 인터페이스를 검색합니다.
GetService 인터페이스에 대한 기본 Sink Writer 개체를 쿼리합니다.
GetService 기본 원본 판독기 개체에 대한 포인터를 가져옵니다.
GetServiceForStream 인터페이스에 대한 기본 미디어 싱크 또는 인코더를 쿼리합니다.
GetServiceForStream 인터페이스에 대한 기본 미디어 원본 또는 디코더를 쿼리합니다.
GetSessionCapabilities 현재 프레젠테이션에 따라 미디어 세션의 기능을 검색합니다.
GetSessionId 애플리케이션에서 세션 개체를 식별하는 데 사용할 수 있는 CDM(Content Decryption Module)에서 생성된 고유 문자열 식별자를 가져옵니다.
GetShowBackgroundAlways 시간 제한 텍스트의 스타일이 항상 배경을 표시하는지 여부를 결정합니다.
GetShutdownStatus IMFShutdown::Shutdown 메서드에 대한 이전 호출의 상태를 쿼리합니다.
getSink 캡처 싱크 개체 중 하나에 대한 포인터를 가져옵니다.
GetSinkInfo 코드 변환 프로필에 지정된 오디오 및 비디오 스트림의 미디어 형식을 가져옵니다.
GetSlowestRate 개체에서 지원하는 가장 느린 재생 속도를 검색합니다.

GetSource 캡처 원본 개체에 대한 포인터를 가져옵니다.
GetSourceAttributes 미디어 원본에 대한 특성 저장소를 가져옵니다.
GetSourceBuffer 버퍼 컬렉션의 지정된 인덱스에 있는 IMFSourceBuffer를 가져옵니다.
GetSourceBuffer 목록의 지정된 인덱스에 있는 IMFSourceBuffer를 가져옵니다.
GetSourceBuffers 이 미디어 원본과 연결된 원본 버퍼의 컬렉션을 가져옵니다.
GetSourceNodeCollection 토폴로지의 원본 노드를 가져옵니다.
GetSourceOrigin HTML5 미디어 요소의 W3C 원본을 반환합니다.
GetStart 지정된 시간 범위의 시작 시간을 가져옵니다.
GetStartTime 초기 재생 위치를 가져옵니다.
GetStartTime 트랙에서 큐의 시작 시간을 가져옵니다.
GetState 클록의 현재 상태를 검색합니다.
GetState 비동기 Begin 메서드에서 호출자가 지정한 상태 개체를 반환합니다. (IMFAsyncResult.GetState)
GetStateNoAddRef 개체의 참조 수를 증가하지 않고 비동기 Begin 메서드에서 호출자가 지정한 상태 개체를 반환합니다. (IMFAsyncResult.GetStateNoAddRef)
GetStatistics 멀티플렉서 통계를 검색합니다.

GetStatistics 미디어 엔진에서 재생 통계를 가져옵니다.
GetStatistics 싱크 기록기의 성능에 대한 통계를 가져옵니다.
GetStatus 비동기 작업의 상태를 반환합니다. (IMFAsyncResult.GetStatus)
GetStatus 이벤트 상태를 지정하는 HRESULT를 검색합니다.
GetStereo3DFramePackingMode 입체 3D 비디오의 경우 비디오 프레임 내에서 두 보기의 레이아웃을 가져옵니다.
GetStereo3DRenderMode 입체 3D 비디오의 경우 미디어 엔진이 3D 비디오 콘텐츠를 렌더링하는 방법을 쿼리합니다.
GetStream 스트림 인덱스별로 프로필에서 스트림을 검색하고 스트림 인덱스의 스트림 번호를 검색합니다.
GetStream 이 인터페이스는 이 버전의 Media Foundation에서 구현되지 않습니다. 스트림 우선 순위 목록에서 스트림의 스트림 번호를 검색합니다.
GetStreamAttribute 미디어 리소스에서 스트림 수준 특성을 가져옵니다.
GetStreamAttributes 미디어 원본의 스트림에 대한 특성 저장소를 가져옵니다.
GetStreamAttributes 지정된 인덱스가 있는 스트림 특성 저장소를 가져옵니다.
GetStreamAttributesCount 센서 디바이스에 대한 스트림 특성 저장소 수를 가져옵니다. 이 숫자는 유효한 모든 스트림에 최소한 스트림 ID 및 스트림 범주가 포함된 특성 저장소가 있어야 하므로 디바이스에 사용할 수 있는 총 스트림 수를 나타냅니다.
GetStreamByNumber 프로필의 스트림에 대한 ASF(Advanced Systems Format) 스트림 구성 개체를 검색합니다. 이 메서드는 스트림 인덱스 대신 스트림 번호를 기준으로 스트림을 참조합니다.
GetStreamConfiguration

멀티플렉싱된 출력을 포함할 수 있는 하위 스트림 집합을 정의하는 지정된 인덱스를 사용하여 스트림 구성을 가져옵니다.
GetStreamConfiguration 멀티플렉싱된 출력이 포함된 하위 스트림 집합을 정의하는 미디어 원본에 대한 활성 스트림 구성을 가져옵니다.
GetStreamConfigurationCount 멀티플렉싱된 출력을 포함할 수 있는 하위 스트림 집합을 정의하는 등록된 스트림 구성의 수를 가져옵니다.
GetStreamCount 프로필의 스트림 수를 검색합니다.
GetStreamCount 이 인터페이스는 이 버전의 Media Foundation에서 구현되지 않습니다. 스트림 우선 순위 목록의 항목 수를 검색합니다.
GetStreamCount ASF(Advanced Systems Format) 콘텐츠에 있는 스트림 수를 검색합니다.
GetStreamCount 멀티플렉싱된 미디어 소스에서 관리하는 하위 스트림의 수를 가져옵니다. (IMFMuxStreamAttributesManager.GetStreamCount)
GetStreamCount 멀티플렉싱된 미디어 소스에서 관리하는 하위 스트림의 수를 가져옵니다. (IMFMuxStreamMediaTypeManager.GetStreamCount)
GetStreamCount 멀티플렉싱된 미디어 소스에서 관리하는 하위 스트림의 수를 가져옵니다. (IMFMuxStreamSampleManager.GetStreamCount)
GetStreamCount 이 MFT(Media Foundation 변환)의 현재 입력 및 출력 스트림 수를 가져옵니다.
GetStreamDescriptor 이 미디어 스트림에 대한 스트림 설명자를 검색합니다.
GetStreamDescriptorCount 프레젠테이션의 스트림 설명자 수를 검색합니다. 각 스트림 설명자에는 미디어 원본의 한 스트림에 대한 정보가 포함됩니다. 스트림 설명자를 검색하려면 IMFPresentationDescriptor::GetStreamDescriptorByIndex 메서드를 호출합니다.
GetStreamIdentifier 스트림의 식별자를 검색합니다.

GetStreamIDs 이 MFT(Media Foundation 변환)의 입력 및 출력 스트림에 대한 스트림 식별자를 가져옵니다.
GetStreamIndexFromFriendlyName 친숙한 스트림 이름에서 변환된 실제 디바이스 스트림 인덱스입니다.
GetStreamingMode 센서 프로세스의 스트리밍 모드를 가져옵니다.
GetStreamingState 센서가 현재 스트리밍 중인지 여부를 나타내는 값을 가져옵니다.
GetStreamLimits 이 MFT(Media Foundation 변환)에 대한 최소 및 최대 입력 및 출력 스트림 수를 가져옵니다.
GetStreamNumber 스트림의 스트림 번호를 검색합니다.
GetStreamPrioritization IMFASFProfile::GetStreamPrioritization 메서드
GetStreamSelection 스트림이 재생되도록 선택되었는지 여부를 쿼리합니다. (IMFMediaEngineEx.GetStreamSelection)
GetStreamSelection 스트림이 선택되었는지 여부를 쿼리합니다.
GetStreamsForRecord Advanced Systems Format 상호 제외 개체의 레코드에 포함된 스트림 번호를 검색합니다.
GetStreamSinkById 스트림 식별자에 의해 지정된 스트림 싱크를 가져옵니다.
GetStreamSinkCount 이 미디어 싱크의 스트림 싱크 수를 가져옵니다.
GetStreamType 스트림의 주요 미디어 형식을 가져옵니다.
GetString 키와 연결된 와이드 문자열을 검색합니다.

GetStringLength 키와 연결된 문자열 값의 길이를 검색합니다.
GetStyle 시간 제한 텍스트 큐의 스타일에 대한 정보를 가져옵니다.
GetStyleCount SAMI 파일에 정의된 스타일 수를 가져옵니다.
getStyles SAMI 파일에 정의된 스타일 이름 목록을 가져옵니다.
GetSubEntries GetSubEntries 메서드는 SetSubEntries에 대한 이전 호출에 의해 설정된 하위 엔터티 인덱스의 배열을 가져옵니다.
getSubformatting 형식이 지정된 시간 제한 텍스트 개체의 하위 폼을 가져옵니다.
GetSubformattingCount 형식이 지정된 시간 제한 텍스트 개체의 하위 폼 수를 가져옵니다.
GetSupportedHardwareEffects 현재 구성된 비디오 프로세서에서 지원되는 효과 목록을 반환합니다.
GetSupportedProtocolType 인덱스별로 지원되는 프로토콜을 검색합니다.
GetSupportedStates 현재 디바이스에서 지원하는 페색 상태를 가져옵니다.
getSurfacePitch 보호된 표면의 보폭을 가져옵니다.
getSuspendNotify CDM(Content Decryption Module)의 일시 중단 알림 인터페이스를 가져옵니다.
getSuspendNotify IMFContentDecryptionModuleSession 일시 중단 이벤트에 대한 개체를 검색합니다.
getSymbolicLink 보고서와 연결된 센서의 기호 링크를 가져옵니다.

getSymbolicLink 센서 디바이스의 기호 링크 이름을 가져옵니다.
getSymbolicLink 센서 그룹의 기호 링크 이름을 가져옵니다.
GetText 형식이 지정된 시간 제한 텍스트 개체의 텍스트를 가져옵니다.
GetTextAlignment 시간 제한 텍스트 스타일의 텍스트 맞춤을 가져옵니다.
GetTextDecoration 시간 제한 텍스트 스타일에 대해 텍스트를 데코레이트하는 방법을 가져옵니다.
GetTextOutline 시간 제한 텍스트 스타일의 텍스트 윤곽선을 가져옵니다.
GetTextTracks 시간 제한 텍스트 구성 요소의 모든 시간 제한 텍스트 트랙 목록을 가져옵니다.
GetTime 최신 클록 시간을 검색합니다.
GetTimelineMarkerTimer 다음 타임라인 표식의 시간(있는 경우)을 가져옵니다.
GetTimeSeekResult 시간 기반 검색의 결과를 가져옵니다.
GetTimeSeekResult Microsoft Media Foundation에서 호출하여 서버가 응답에 지정한 TimeSeekRange.DLNA.ORG HTTP 헤더(있는 경우)의 값을 검색합니다.
GetTimeSource 클록의 프레젠테이션 시간 원본을 검색합니다.
GetTimeStampOffset IMFSourceBuffer에 추가된 미디어 세그먼트의 타임스탬프 오프셋을 가져옵니다.
GetTitle GetTitle 메서드는 항목의 SetTitle에 대한 이전 호출에 의해 설정된 제목을 검색합니다.

GetTocByType GetTocByType은 TOC 파서 개체에서 지정된 형식의 모든 목차를 검색합니다.
GetTocCount GetTocCount 메서드는 TOC 파서 개체에서 지정된 위치 형식의 목차 수를 검색합니다.
GetTopologyID 토폴로지의 식별자를 가져옵니다.
GetTopologyWorkQueueMMCSSClass 현재 토폴로지의 지정된 분기에 대한 MMCSS(멀티미디어 클래스 Scheduler Service) 클래스를 검색합니다.
GetTopologyWorkQueueMMCSSPriority 지정된 토폴로지 작업 큐와 연결된 MMCSS(멀티미디어 클래스 스케줄러 서비스) 문자열을 검색합니다.
GetTopologyWorkQueueMMCSSTaskId 현재 토폴로지의 지정된 분기에 대한 MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업 식별자를 검색합니다.
GetTopoNodeID 노드의 식별자를 검색합니다.
GetTotalLength 알려진 경우 다운로드 중인 리소스의 총 길이를 검색하기 위해 Microsoft Media Foundation에서 호출합니다.
GetTotalLength 샘플의 모든 버퍼에서 유효한 데이터의 총 길이를 검색합니다. 길이는 IMFMediaBuffer::GetCurrentLength 메서드에서 검색한 값의 합계로 계산됩니다.
GetTrack 트랙의 인덱스에서 목록의 텍스트 트랙을 가져옵니다.
GetTrackById 트랙의 식별자에서 목록의 텍스트 트랙을 가져옵니다.
GetTrackId 시간 제한 텍스트 큐의 식별자를 가져옵니다.
GetTrackKind 시간 제한 텍스트 트랙의 종류를 가져옵니다.
GetTracks IMFTimedText에 등록된 모든 시간 제한 텍스트 트랙 목록을 검색합니다.

GetTransformCount 센서 변환에서 제공하는 변환 수를 가져오기 위해 미디어 파이프라인에서 호출됩니다.
GetTransformForStream 지정된 스트림에 대한 MFT(Media Foundation 변환)에 대한 포인터를 가져옵니다. (IMFSinkWriterEx.GetTransformForStream)
GetTransformForStream 지정된 스트림에 대한 MFT(Media Foundation 변환)에 대한 포인터를 가져옵니다. (IMFSourceReaderEx.GetTransformForStream)
GetTransformInformation 센서 변환에서 제공하는 변환에 대한 정보를 가져오기 위해 미디어 파이프라인에서 호출됩니다.
GetType 스트리밍이 시작되기 전에 또는 스트리밍이 시작된 후 컨트롤 값을 설정해야 하는지 여부를 나타내는 컨트롤의 구성 형식을 가져옵니다.
GetType ASF(Advanced Systems Format) 상호 제외 개체가 나타내는 상호 제외 유형을 검색합니다.
GetType 이벤트 유형을 검색합니다. 이벤트 유형은 이벤트를 트리거하기 위해 발생한 작업을 나타냅니다. 또한 이벤트 값의 의미를 정의합니다.
GetType 목록에 있는 요소의 MIME 형식을 가져옵니다.
GetType 버퍼가 나타내는 공간 오디오 개체의 형식을 가져옵니다. SetType이 이전에 호출되지 않은 경우 이 메서드는 AudioObjectType_None 기본값을 반환합니다.
GetUINT32 키와 연결된 UINT32 값을 검색합니다.
GetUINT64 키와 연결된 UINT64 값을 검색합니다.
getUnderlyingClock 프레젠테이션 시간 원본에서 해당 시계 시간을 생성하는 데 사용하는 기본 클록을 검색합니다.
GetUnknown 키와 연결된 인터페이스 포인터를 검색합니다.

GetUnknown 이전에 미디어 버퍼 개체에 저장된 IUnknown 포인터를 가져옵니다.
getUpdating Append, AppendByteStream 또는 Remove가 처리 중인지 여부를 나타내는 값을 가져옵니다.
GetURL 요청을 보내는 데 사용되는 URL을 반환합니다.
GetURL 목록에 있는 요소의 URL을 가져옵니다.
GetUser 사용자 이름을 검색합니다.
GetValue ICodecAPI::GetValue 메서드는 코덱 속성의 현재 값을 가져옵니다.
GetValue 이벤트와 연결된 값(있는 경우)을 검색합니다. 값은 PROPVARIANT 구조체로 검색됩니다. 실제 데이터 형식과 값의 의미는 이벤트에 따라 달라집니다.
GetVideoAspectRatio 비디오 스트림의 그림 가로 세로 비율을 가져옵니다.
GetVideoAttributes 현재 코드 변환 프로필에 설정된 비디오 스트림 설정을 가져옵니다.
GetVideoDecoderCaps 비디오 디코더의 기능 및 제한 사항을 검색합니다.
GetVideoDecoderConfig 드라이버에서 지원하는 디코더 구성을 가져옵니다.
GetVideoDecoderConfigCount 드라이버가 지정된 비디오 설명에 대해 지원하는 디코더 구성 수를 가져옵니다.
GetVideoDecoderProfile 드라이버에서 지원하는 프로필을 가져옵니다.
GetVideoDecoderProfileCount 드라이버에서 지원하는 프로필 수를 가져옵니다.

GetVideoDecoderService 이 디코더 디바이스를 만든 DXVA(DirectX Video Acceleration) 디코더 서비스를 검색합니다.
GetVideoFormat GetVideoFormat은 Windows 7을 기준으로 더 이상 사용할 수 없습니다.
GetVideoProcessBltState Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스에서 수행하는 blit 작업에 대한 상태 매개 변수 값을 가져옵니다.
GetVideoProcessorCaps 비디오 프로세서의 기능을 가져옵니다.
GetVideoProcessorCaps 하나 이상의 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서의 기능을 가져옵니다.
GetVideoProcessorCaps 비디오 프로세서 디바이스의 기능을 검색합니다.
GetVideoProcessorCaps 지정된 비디오 프로세서 디바이스의 기능을 가져옵니다.
GetVideoProcessorContentDesc 이 열거자를 만드는 데 사용된 콘텐츠 설명을 가져옵니다.
GetVideoProcessorCustomRate 비디오 프로세서에서 지원하는 사용자 지정 프레임 속도 목록을 가져옵니다.
GetVideoProcessorCustomRates Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서에서 지원하는 사용자 지정 속도 목록을 가져옵니다. 사용자 지정 속도는 프레임 속도 변환 및 IVTC(역 텔레신)에 사용됩니다.
GetVideoProcessorDeviceCaps Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스의 기능을 가져옵니다.
GetVideoProcessorDeviceGuids 그래픽 하드웨어에서 지원하는 비디오 프로세서를 식별하는 GUID 배열을 가져옵니다.
GetVideoProcessorFilterRange 이미지 필터의 값 범위를 가져옵니다.
GetVideoProcessorFilterRange

Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스에서 지원하는 이미지 필터의 값 범위를 가져옵니다.
GetVideoProcessorInputFormats Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스에서 지원하는 입력 형식의 목록을 가져옵니다.
GetVideoProcessorOutputFormats Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스에서 지원하는 출력 형식의 목록을 가져옵니다.
GetVideoProcessorRateConversionCaps 디인터레이싱 및 역 텔레신을 포함하여 프레임 속도 변환과 연결된 비디오 프로세서 기능 그룹을 반환합니다.
GetVideoProcessorRenderTargets 비디오 프로세서 디바이스에서 지원하는 렌더링 대상 형식을 가져옵니다. 목록에는 RGB 및 YUV 형식이 포함될 수 있습니다.
GetVideoProcessorService 이 비디오 프로세서 디바이스를 만든 DXVA(DirectX Video Acceleration) 비디오 프로세서 서비스를 검색합니다.
GetVideoProcessorSubStreamFormats 지정된 비디오 프로세서 디바이스에서 지원하는 하위 스트림 형식의 목록을 가져옵니다.
GetVideoProcessStreamState Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스의 입력 스트림에 대한 상태 매개 변수 값을 가져옵니다.
GetVideoRepresentation 미디어 형식의 대체 표현을 검색합니다.
GetVideoService Microsoft Direct3D 디바이스에서 인터페이스를 쿼리합니다.
GetVideoService DXVA(DirectX Video Acceleration) 서비스 인터페이스를 가져옵니다.
getVideoSwapchainHandle 창 없는 스왑 체인에 대한 핸들을 가져옵니다.
GetVolume 오디오 볼륨 수준을 가져옵니다.
GetWrap 지역에서 단어 줄 바꿈 기능을 사용할 수 있는지 여부를 결정합니다.
GetWritingMode

지역의 쓰기 모드를 가져옵니다.
HasAudio 현재 미디어 리소스에 오디오 스트림이 포함되어 있는지 여부를 쿼리합니다.
HasNullSourceOrigin HTTP 또는 HTTPS 요청이 다른 "원본"의 다른 서버로 리디렉션된 경우를 감지하기 위해 Microsoft Media Foundation에서 호출합니다.
HasVideo 현재 미디어 리소스에 비디오 스트림이 포함되어 있는지 여부를 쿼리합니다.
Init Init 메서드는 TOC 파서 개체를 초기화하고 미디어 파일과 연결합니다.
초기화 인덱서 개체를 초기화합니다.
초기화 ASF ContentInfo 개체의 데이터를 사용하여 멀티플렉서를 초기화합니다.
초기화 ASF(Advanced Systems Format) 분할자를 다시 설정하여 ASF 데이터 섹션의 데이터를 구문 분석하도록 구성합니다.
초기화 캡처 엔진을 초기화합니다.
초기화 DLNA(Digital Living Network Alliance) 미디어 싱크를 초기화합니다. (IMFDLNASinkInit.Initialize)
InitializeCaptureSampleAllocator 비디오 캡처 시나리오와 관련된 매개 변수를 사용하여 샘플 할당자를 초기화합니다.
InitializeExtensionCommand 비디오 확장 명령을 초기화하거나 다시 초기화하는 명령을 비디오 디코딩 명령 목록으로 기록합니다.
InitializeExtensionCommand 비디오 확장 명령을 초기화하거나 비디오 프로세서 명령 목록으로 다시 초기화하는 명령을 기록합니다.
InitializeFactory 센서 변환을 초기화하기 위해 미디어 파이프라인에서 호출됩니다.
InitializeHardwareKey

표시 드라이버가 새 하드웨어 키를 초기화할 때 사용되는 IHV 관련 정보를 반환할 수 있도록 허용합니다.
InitializeSampleAllocator 할당할 샘플 수와 샘플의 미디어 형식을 지정합니다.
InitializeSampleAllocatorEx 비디오 샘플 할당자 개체를 초기화합니다.
InitInputProtection 다운스트림 구성 요소에서 샘플 보호를 초기화합니다.
initOutputProtection 업스트림 구성 요소에서 샘플 보호를 위한 초기화 정보를 검색합니다.
InsertAudioEffect 오디오 효과를 삽입합니다.
InsertElementAt 컬렉션의 지정된 인덱스에서 개체를 추가합니다.
InsertVideoEffect 비디오 효과를 삽입합니다.
호출 비동기 작업이 완료될 때 호출됩니다. (IMFAsyncCallback.Invoke)
InvokeFunction 보안 프로세서에서 보호 시스템의 구현을 호출합니다.
IsActive 시간 제한 텍스트 트랙이 활성 상태인지 여부를 확인합니다.
IsAutomaticSupported 콘텐츠 인에이블러가 모든 작업을 자동으로 수행할 수 있는지 여부를 쿼리합니다.
IsBackgroundTransferActive 백그라운드 전송이 활성 상태인지 여부를 쿼리합니다.
IsCompressedFormat 미디어 형식이 임시로 압축된 형식인지 여부를 쿼리합니다.
isContiguousFormat 버퍼가 네이티브 형식으로 연속되는지 여부를 쿼리합니다.

IsDisabled CLSID(클래스 식별자)가 차단된 목록에 표시되는지 여부를 쿼리합니다. (IMFPluginControl.IsDisabled)
IsEnded 재생이 종료되었는지 여부를 쿼리합니다.
isEndOfStream 현재 위치가 스트림의 끝에 도달했는지 여부를 쿼리합니다.
IsEqual 두 미디어 형식을 비교하고 동일한지 여부를 결정합니다. 동일하지 않은 경우 메서드는 두 형식의 차이점을 나타냅니다.
IsExternal 시간 제한 텍스트 스타일이 외부 스타일인지 여부를 확인합니다.
IsFinal 이 출력이 정책 싱크인지 여부를 쿼리합니다. 즉, ITA(입력 신뢰 기관)에 필요한 권한 및 제한을 처리합니다.
IsInBand 시간 제한 텍스트 트랙이 인밴드되어 있는지 여부를 확인합니다.
isInBandEnabled 인밴드 모드를 사용할지 여부를 결정합니다.
IsMediaTypeSupported 개체가 지정된 미디어 형식을 지원하는지 여부를 쿼리합니다.
IsMediaTypeSupported 미디어 스트림이 지정된 미디어 형식을 지원하는지 여부를 확인합니다.
isModifiable IsModifiable 메서드는 코덱의 현재 구성에 따라 코덱 속성을 변경할 수 있는지 여부를 쿼리합니다. (ICodecAPI::IsModifiable)
IsPaused 재생이 현재 일시 중지되었는지 여부를 쿼리합니다.
IsPlaybackRateSupported 미디어 엔진이 지정된 재생 속도로 재생할 수 있는지 여부를 쿼리합니다.
isProtected

미디어 리소스에 보호된 콘텐츠가 포함되어 있는지 여부를 쿼리합니다.
IsRateSupported 개체가 지정된 재생 속도를 지원하는지 여부를 쿼리합니다.
IsSameOrigin 지정된 URL의 원본이 HTML5 미디어 요소와 같으면 true를 반환합니다.
IsSeeking 미디어 엔진이 현재 새 재생 위치를 찾고 있는지 여부를 쿼리합니다.
IsStereo3D 미디어 리소스에 스테레오스코픽 3D 비디오가 포함되어 있는지 여부를 쿼리합니다.
IsSupported IsSupported 메서드는 코덱이 지정된 속성을 지원하는지 여부를 쿼리합니다. (ICodecAPI::IsSupported)
IsTimeSeekSupported 바이트 스트림이 시간 기반 검색을 지원하는지 여부를 쿼리합니다.
isTypeSupported 지정된 키 시스템에서 지정된 미디어 형식을 지원하는지 나타내는 값을 가져옵니다.
isTypeSupported 지정된 MIME 형식이 미디어 원본에서 지원되는지를 나타내는 값을 가져옵니다.
isTypeSupported 지정된 키 시스템 및 선택적으로 콘텐츠 형식이 지원되는지 여부를 쿼리합니다.
isTypeSupportedEx 지정된 콘텐츠 형식이 지정된 키 시스템에 대해 지원되는 쿼리입니다.
KeyAdded 키가 추가되었음을 애플리케이션에 알깁니다.
KeyError 키를 처리하는 동안 오류가 발생했음을 애플리케이션에 알깁니다.
KeyMessage 키 획득을 시작할 수 있도록 애플리케이션에 정보를 전달합니다.
KeyMessage CDM(Content Decryption Module)이 세션에 대한 메시지를 생성했을 때 호출됩니다.

KeyStatusChanged CDM(콘텐츠 암호 해독 모듈) 세션 또는 해당 상태에 키가 변경된 경우 IMFContentDecryptionModuleSessionCallbacks::KeyStatusChanged 함수가 호출됩니다.
로드 입력 부분 토폴로지에서 완전히 로드된 토폴로지 만들기
로드 현재 미디어 원본을 로드합니다.
로드 IMFContentDecryptionModuleSession::Load 함수는 지정된 세션의 데이터를 IMFContentDecryptionModuleSession 개체에 로드합니다.
LoadDefaults 지정된 캡처 원본에 대한 카메라 컨트롤 기본값을 로드합니다.
loadReference 스트림에 저장된 데이터에서 인터페이스를 마샬링합니다.
잠금 호출자에게 읽기 또는 쓰기를 위해 버퍼의 메모리에 대한 액세스 권한을 부여합니다.
Lock2D 호출자에게 버퍼의 메모리에 대한 액세스 권한을 부여합니다. (IMF2DBuffer.Lock2D)
Lock2DSize 호출자에게 버퍼의 메모리에 대한 액세스 권한을 부여합니다. (IMF2DBuffer2.Lock2DSize)
LockControlData IMFCameraControlDefaults 인스턴스와 연결된 컨트롤에 대한 데이터 페이로드를 검색하여 클라이언트가 컨트롤 값을 직접 수정할 수 있도록 합니다.
LockDevice 호출자에게 Microsoft Direct3D 디바이스에 대한 단독 액세스 권한을 부여합니다.
LockDevice 호출자에게 Direct3D 디바이스에 대한 단독 액세스 권한을 부여합니다.
LockPayload 캡처 디바이스 컨트롤에 포함된 내부 페이로드 버퍼를 잠가 페이로드를 쿼리하거나 변경할 수 있도록 합니다.
LockProcess

PMP(보호된 미디어 경로) 프로세스가 종료되지 않도록 차단합니다. (IMFPMPHost.LockProcess)
LockProcess PMP(보호된 미디어 경로) 프로세스가 종료되지 않도록 차단합니다. (IMFPMPHostApp.LockProcess)
LockProcess PMP(보호된 미디어 경로) 프로세스가 종료되지 않도록 차단합니다. (IMFPMPServer.LockProcess)
LockStore 다른 스레드가 액세스할 수 없도록 특성 저장소를 잠글 수 있습니다.
LoggedOnUser 로그온한 자격 증명을 사용해야 하는지 여부를 쿼리합니다.
MFAddPeriodicCallback 고정 간격으로 호출할 콜백 함수를 설정합니다. (MFAddPeriodicCallback)
MFAllocateSerialWorkQueue 작업 항목을 직렬화하도록 보장되는 작업 큐를 만듭니다.
MFAllocateWorkQueue 새 작업 큐를 만듭니다. (MFAllocateWorkQueue)
MFAllocateWorkQueueEx 새 작업 큐를 만듭니다. (MFAllocateWorkQueueEx)
MFAverageTimePerFrameToFrameRate 비디오 프레임의 평균 기간에서 프레임 속도를 초당 프레임 단위로 계산합니다.
MFBeginCreateFile 파일에서 바이트 스트림을 만드는 비동기 요청을 시작합니다.
MFBeginRegisterWorkQueueWithMMCSS 작업 큐를 MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업에 연결합니다. (MFBeginRegisterWorkQueueWithMMCSS)
MFBeginRegisterWorkQueueWithMMCSSEx 작업 큐를 MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업에 연결합니다. (MFBeginRegisterWorkQueueWithMMCSSEx)
MFBeginUnregisterWorkQueueWithMMCSS MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업에서 작업 큐의 등록을 취소합니다. (MFBeginUnregisterWorkQueueWithMMCSS)

MFCalculateBitmapImageSize 비디오 형식의 이미지 크기를 검색합니다.
MFCalculateImageSize 압축되지 않은 비디오 형식의 이미지 크기(바이트)를 검색합니다. (MFCalculateImageSize)
MFCancelCreateFile 파일에서 바이트 스트림을 만드는 비동기 요청을 취소합니다.
MFCancelWorkItem MFScheduleWorkItem 또는 MFScheduleWorkItemEx로 예약된 비동기 작업을 취소하려고 시도합니다.
MFCombineSamples 결합된 기간이 지정된 기간을 초과하지 않는 경우 미디어 샘플을 다른 샘플에 연결합니다.
MFCompareFullToPartialMediaType 전체 미디어 형식과 부분 미디어 형식을 비교합니다.
MFConvertColorInfoFromDXVA MFVIDEOFORMAT 구조체의 확장 색 정보를 설정합니다.
MFConvertColorInfoToDXVA 확장 색 정보를 MFVIDEOFORMAT에서 해당하는 DXVA(DirectX Video Acceleration) 색 정보로 변환합니다.
MFConvertFromFP16Array 16비트 부동 소수점 숫자의 배열을 32비트 부동 소수점 숫자의 배열로 변환합니다.
MFConvertToFP16Array 32비트 부동 소수점 숫자의 배열을 16비트 부동 소수점 숫자의 배열로 변환합니다.
MFCopyImage 한 버퍼에서 다른 버퍼로 이미지 또는 이미지 평면을 복사합니다.
MFCreate2DMediaBuffer 2D 이미지 데이터를 저장할 시스템 메모리 버퍼 개체를 만듭니다.
MFCreate3GPMediaSink 3GP 파일을 작성하기 위한 미디어 싱크를 만듭니다.
MFCreateAC3MediaSink AC-3 미디어 싱크의 인스턴스를 만듭니다.

MFCreateADTSMediaSink ADTS(오디오 데이터 전송 스트림) 미디어 싱크의 인스턴스를 만듭니다.
MFCreateAggregateSource 미디어 원본 컬렉션을 집계하는 미디어 원본을 만듭니다.
MFCreateAlignedMemoryBuffer 지정된 바이트 맞춤을 사용하여 시스템 메모리를 할당하고 메모리를 관리하는 미디어 버퍼를 만듭니다.
MFCreateAMMediaTypeFromMFMediaType Media Foundation 미디어 형식에서 DirectShow AM_MEDIA_TYPE 구조를 만듭니다.
MFCreateASFContentInfo ASF Header 개체 개체를 만듭니다.
MFCreateASFIndexer ASF 인덱서 개체를 만듭니다.
MFCreateASFIndexerByteStream ASF 스트림의 인덱스 액세스에 대한 바이트 스트림을 만듭니다.
MFCreateASFMediaSink ASF 미디어 싱크를 만듭니다.
MFCreateASFMediaSinkActivate ASF 미디어 싱크를 만드는 데 사용할 수 있는 활성화 개체를 만듭니다.
MFCreateASFMultiplexer ASF 멀티플렉서를 만듭니다.
MFCreateASFProfile ASF 프로필 개체를 만듭니다.
MFCreateASFProfileFromPresentationDescriptor 프레젠테이션 설명자에서 ASF 프로필 개체를 만듭니다.
MFCreateASFSplitter ASF 분할자를 만듭니다.
MFCreateASFStreamingMediaSink ASF 스트리밍 싱크에 대한 활성화 개체를 만듭니다. (MFCreateASFStreamingMediaSink)

MFCreateASFStreamingMediaSinkActivate ASF 스트리밍 싱크에 대한 활성화 개체를 만듭니다. (MFCreateASFStreamingMediaSinkActivate)
MFCreateASFStreamSelector ASF 스트림 선택기를 만듭니다.
MFCreateAsyncResult 비동기 결과 개체를 만듭니다. 비동기 메서드를 구현하는 경우 이 함수를 사용합니다. (MFCreateAsyncResult)
MFCreateAttributes 빈 특성 저장소를 만듭니다.
MFCreateAudioMediaType WAVEFORMATEX 구조체에서 오디오 미디어 형식을 만듭니다.
MFCreateAudioRenderer 스트리밍 오디오 렌더러를 만듭니다.
MFCreateAudioRendererActivate 스트리밍 오디오 렌더러에 대한 활성화 개체를 만듭니다.
MFCreateAVIMediaSink Audio-Video AVI(인터리브) 싱크를 만듭니다.
MFCreateCameraControlMonitor IMFCameraControlMonitor의 인스턴스를 만듭니다.
MFCreateCameraOcclusionStateMonitor 카메라 폐색 상태가 변경될 때 애플리케이션이 알림을 받을 수 있도록 하는 IMFCameraOcclusionStateMonitor의 새 인스턴스를 만듭니다.
MFCreateCollection 빈 컬렉션 개체를 만듭니다.
MFCreateContentDecryptorContext 지정된 미디어 보호 시스템에 대한 IMFContentDecryptorContext 인터페이스를 만듭니다.
MFCreateContentProtectionDevice 지정된 미디어 보호 시스템에 대한 IMFContentProtectionDevice 인터페이스를 만듭니다.
MFCreateCredentialCache

자격 증명 캐시 개체를 만듭니다. 애플리케이션은 이 개체를 사용하여 사용자 지정 자격 증명 관리자를 구현할 수 있습니다.
MFCreateD3D12SynchronizationObject Media Foundation 개체에 저장된 D3D12 리소스에 대한 액세스를 동기화하는 데 사용되는 Media Foundation D3D12 동기화 기본 형식을 인스턴스화합니다.
MFCreateDeviceSource 하드웨어 캡처 디바이스에 대한 미디어 원본을 만듭니다.
MFCreateDeviceSourceActivate 하드웨어 캡처 디바이스를 나타내는 활성화 개체를 만듭니다.
MFCreateDXGIDeviceManager Microsoft DXGI(DirectX Graphics Infrastructure) 디바이스 관리자의 인스턴스를 만듭니다.
MFCreateDXGISurfaceBuffer Microsoft DXGI(DirectX Graphics Infrastructure) 화면을 관리하는 미디어 버퍼를 만듭니다.
MFCreateDXSurfaceBuffer Direct3D 9 화면을 관리하는 미디어 버퍼 개체를 만듭니다.
MFCreateEncryptedMediaExtensionsStoreActivate MFCreateEncryptedMediaExtensionsStoreActivate 함수는 IMFActivate를 구현하는 개체를 만듭니다. 이 개체의 ActivateObject 구현은 지정된 IMFPMPHostApp 및 클래스 ID를 기반으로 합니다.
MFCreateEventQueue 이벤트 큐를 만듭니다.
MFCreateFile 파일에서 바이트 스트림을 만듭니다.
MFCreateFMPEG4MediaSink 조각화된 MP4 파일을 작성하기 위한 미디어 싱크를 만듭니다.
MFCreateLegacyMediaBufferOnMFMediaBuffer Media Foundation 미디어 버퍼를 DMO(DirectX Media Objects)와 호환되는 버퍼로 변환합니다.
MFCreateMediaBufferFromMediaType 지정된 미디어 형식에 가장 적합한 시스템 메모리 버퍼를 할당합니다.
MFCreateMediaBufferWrapper 기존 미디어 버퍼를 래핑하는 미디어 버퍼를 만듭니다.

MFCreateMediaEvent 미디어 이벤트 개체를 만듭니다.
MFCreateMediaExtensionActivate Windows 런타임 클래스에 대한 활성화 개체를 만듭니다.
MFCreateMediaSession 애플리케이션 프로세스에서 미디어 세션을 만듭니다.
MFCreateMediaType 빈 미디어 형식을 만듭니다.
MFCreateMediaTypeFromProperties 속성에서 IMFMediaType을 만듭니다.
MFCreateMediaTypeFromRepresentation 다른 형식 표현에서 Media Foundation 미디어 형식을 만듭니다.
MFCreateMemoryBuffer 시스템 메모리를 할당하고 이를 관리하는 미디어 버퍼를 만듭니다.
MFCreateMFByteStreamOnStream IStream 포인터를 래핑하는 Microsoft Media Foundation 바이트 스트림을 만듭니다.
MFCreateMFByteStreamOnStreamEx IRandomAccessStream 개체를 래핑하는 Microsoft Media Foundation 바이트 스트림을 만듭니다.
MFCreateMFByteStreamWrapper 바이트 스트림에 대한 래퍼를 만듭니다.
MFCreateMFVideoFormatFromMFMediaType 비디오 미디어 형식에서 MFVIDEOFORMAT 구조를 만듭니다.
MFCreateMP3MediaSink MP3 미디어 싱크를 만듭니다.
MFCreateMPEG4MediaSink MP4 파일을 작성하기 위한 미디어 싱크를 만듭니다.
MFCreateMuxSink 멀티플렉서 MFT(Microsoft Media Foundation 변환)를 래핑하는 일반 미디어 싱크를 만듭니다.

MFCreateMuxStreamAttributes 멀티플렉싱된 하위 스트림의 콘텐츠를 설명하는 IMFAttributes를 만듭니다.
MFCreateMuxStreamMediaType 멀티플렉싱된 하위 스트림의 미디어 형식을 설명하는 IMFMediaType을 만듭니다.
MFCreateMuxStreamSample 멀티플렉싱된 하위 스트림의 샘플을 포함하는 IMFSample을 만듭니다.
MFCreateNetSchemePlugin 네트워크 원본에 대한 스키마 처리기를 만듭니다.
MFCreatePMPMediaSession PMP(보호된 미디어 경로) 프로세스 내에 미디어 세션의 인스턴스를 만듭니다.
MFCreatePMPServer PMP(보호된 미디어 경로) 서버 개체를 만듭니다.
MFCreatePresentationClock 프레젠테이션 시계를 만듭니다.
MFCreatePresentationDescriptor 프레젠테이션 설명자를 만듭니다.
MFCreatePresentationDescriptorFromASFProfile ASF 프로필 개체에서 프레젠테이션 설명자를 만듭니다.
MFCreatePropertiesFromMediaType IMFMediaType에서 속성을 만듭니다.
MFCreateProtectedEnvironmentAccess 콘텐츠 보호 시스템이 보호된 환경과 핸드셰이크를 수행할 수 있도록 하는 IMFProtectedEnvironmentAccess 개체를 만듭니다.
MFCreateProxyLocator 기본 프록시 로케이터를 만듭니다.
MFCreateRelativePanelWatcher **IMFRelativePanelWatcher** 인터페이스의 새 인스턴스를 만듭니다.
MFCreateRemoteDesktopPlugin

원격 데스크톱 플러그 인 개체를 만듭니다. 애플리케이션이 터미널 서비스 클라이언트 세션에서 실행 중인 경우 이 개체를 사용합니다.
MFCreateSample 빈 미디어 샘플을 만듭니다.
MFCreateSampleCopierMFT 샘플 복사기 변환의 인스턴스를 만듭니다.
MFCreateSampleGrabberSinkActivate 샘플 그래버 미디어 싱크에 대한 활성화 개체를 만듭니다.
MFCreateSensorActivityMonitor IMFSensorActivityMonitor 인터페이스의 새 인스턴스를 초기화합니다.
MFCreateSensorGroup 제공된 기호 링크 이름을 기반으로 IMFSensorGroup 인터페이스의 인스턴스를 만듭니다.
MFCreateSensorProfile 지정된 형식, 인덱스 및 선택적 제약 조건에 따라 센서 프로필을 만듭니다.
MFCreateSensorProfileCollection 센서 프로필 컬렉션을 만듭니다.
MFCreateSensorStream IMFSensorStream 인터페이스의 인스턴스를 만듭니다.
MFCreateSequencerSegmentOffset 시퀀서 소스 프레젠테이션 내에서 검색하는 데 사용할 수 있는 PROPVARIANT를 만듭니다.
MFCreateSequencerSource 시퀀서 원본을 만듭니다.
MFCreateSimpleTypeHandler 한 번에 단일 미디어 형식을 지원하는 미디어 형식 처리기를 만듭니다.
MFCreateSinkWriterFromMediaSink 미디어 싱크에서 싱크 작성기를 만듭니다.
MFCreateSinkWriterFromURL URL 또는 바이트 스트림에서 싱크 작성기를 만듭니다.
MFCreateSourceReaderFromByteStream

바이트 스트림에서 원본 판독기를 만듭니다.
MFCreateSourceReaderFromMediaSource 미디어 원본에서 원본 판독기를 만듭니다.
MFCreateSourceReaderFromURL URL에서 원본 판독기를 만듭니다.
MFCreateSourceResolver URL 또는 바이트 스트림에서 미디어 원본을 만드는 데 사용되는 원본 확인자를 만듭니다.
MFCreateStandardQualityManager 품질 관리자의 기본 구현을 만듭니다.
MFCreateStreamDescriptor 스트림 설명자를 만듭니다.
MFCreateStreamOnMFByteStream Microsoft Media Foundation 바이트 스트림을 래핑하는 IStream 포인터를 반환합니다.
MFCreateStreamOnMFByteStreamEx Microsoft Media Foundation 바이트 스트림을 래핑하는 IRandomAccessStream 개체를 만듭니다.
MFCreateSystemTimeSource 시스템 시간을 기반으로 하는 프레젠테이션 시간 원본을 만듭니다.
MFCreateTempFile 임시 로컬 파일에서 지원되는 바이트 스트림을 만듭니다.
MFCreateTopoLoader 토폴로지 로더의 새 인스턴스를 만듭니다.
MFCreateTopology 토폴로지 개체를 만듭니다.
MFCreateTopologyNode 토폴로지 노드를 만듭니다.
MFCreateTrackedSample 비디오 미디어 샘플에서 참조 횟수를 추적하는 IMFTrackedSample 개체를 만듭니다.
MFCreateTranscodeProfile 빈 코드 변환 프로필 개체를 만듭니다.

MFCreateTranscodeSinkActivate 트랜스코드 싱크 활성화 개체를 만듭니다.
MFCreateTranscodeTopology 부분 트랜스코드 토폴로지 만들기
MFCreateTranscodeTopologyFromByteStream 바이트 스트림으로 코드 변환하기 위한 토폴로지 만들기
MFCreateTransformActivate MFT(Media Foundation 변환)에 대한 제네릭 활성화 개체를 만듭니다.
MFCreateVideoMediaType MFVIDEOFORMAT 구조체에서 비디오 미디어 형식을 만듭니다.
MFCreateVideoMediaTypeFromBitMapInfoHeader 이 함수는 구현되지 않습니다. (MFCreateVideoMediaTypeFromBitMapInfoHeader)
MFCreateVideoMediaTypeFromBitMapInfoHeaderEx BITMAPINFOHEADER 구조체에서 비디오 미디어 형식을 만듭니다.
MFCreateVideoMediaTypeFromSubtype 지정된 하위 형식을 사용하여 부분 비디오 미디어 형식을 만듭니다.
MFCreateVideoMediaTypeFromVideoInfoHeader KS_VIDEOINFOHEADER 구조체에서 미디어 형식을 만듭니다.
MFCreateVideoMediaTypeFromVideoInfoHeader2 KS_VIDEOINFOHEADER2 구조체에서 미디어 형식을 만듭니다.
MFCreateVideoRendererActivate 향상된 EVR(비디오 렌더러) 미디어 싱크에 대한 활성화 개체를 만듭니다.
MFCreateVideoSampleAllocatorEx Microsoft DXGI(DirectX Graphics Infrastructure)와 호환되는 비디오 샘플을 할당하는 개체를 만듭니다.
MFCreateVirtualCamera 호출자가 시스템에서 가상 카메라를 등록, 등록 취소 또는 제거하는 데 사용할 수 있는 가상 카메라 개체를 만듭니다.
MFCreateWaveFormatExFromMFMediaType Media Foundation 오디오 미디어 형식을 WAVEFORMATEX 구조체로 변환합니다.

MFCreateWAVEMediaSink WAVE 보관 싱크를 만듭니다. WAVE 보관 싱크는 오디오를 가져와서 .wav 파일에 씁니다.
MFCreateWICBitmapBuffer WIC(Windows 이미징 구성 요소)를 관리하는 미디어 버퍼 개체를 만듭니다.
MFCreateWMAEncoderActivate WMA(Windows Media Audio) 인코더를 만드는 데 사용할 수 있는 활성화 개체를 만듭니다.
MFCreateWMVEncoderActivate WMV(Windows Media Video) 인코더를 만드는 데 사용할 수 있는 활성화 개체를 만듭니다.
MFDeserializeAttributesFromStream 스트림에서 특성 저장소로 특성을 로드합니다.
MFDeserializePresentationDescriptor 바이트 배열에서 프레젠테이션 설명자를 역직렬화합니다.
MFEndCreateFile 파일에서 바이트 스트림을 만드는 비동기 요청을 완료합니다.
MFEndRegisterWorkQueueWithMMCSS 작업 큐를 MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업과 연결하는 비동기 요청을 완료합니다. (MFEndRegisterWorkQueueWithMMCSS)
MFEndUnregisterWorkQueueWithMMCSS MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업에서 작업 큐의 등록을 취소하는 비동기 요청을 완료합니다.
MFEnumDeviceSources 오디오 또는 비디오 캡처 디바이스 목록을 열거합니다.
MFFrameRateToAverageTimePerFrame 비디오 프레임 속도를 프레임 기간으로 변환합니다.
MFGetAttribute2UINT32asUINT64 값이 UINT64로 압축된 두 UINT32 값인 특성을 가져옵니다.
MFGetAttributeDouble 특성 저장소에서 double 값을 반환하거나 특성이 없는 경우 기본값을 반환합니다.
MFGetAttributeRatio 값이 비율인 특성을 검색합니다.

MFGetAttributesAsBlob 특성 저장소의 내용을 바이트 배열로 변환합니다.
MFGetAttributesAsBlobSize MFGetAttributesAsBlob 함수에 필요한 버퍼의 크기를 검색합니다.
MFGetAttributeSize 값이 너비 및 높이로 표현된 크기인 특성을 검색합니다.
MFGetAttributeString 특성 저장소에서 문자열 값을 가져옵니다.
MFGetAttributeUINT32 특성 저장소에서 UINT32 값을 반환하거나 특성이 없는 경우 기본값을 반환합니다.
MFGetAttributeUINT64 특성 저장소에서 UINT64 값을 반환하거나 특성이 없는 경우 기본값을 반환합니다.
MFGetContentProtectionSystemCLSID 콘텐츠 보호 시스템의 클래스 식별자를 가져옵니다.
MFGetLocalId 로컬 시스템 ID를 가져옵니다.
MFGetMFTMerit 하드웨어 코덱의 장점 값을 가져옵니다.
MFGetPlaneSize 압축되지 않은 비디오 형식의 이미지 크기(바이트)를 검색합니다. (MFGetPlaneSize)
MFGetPluginControl Microsoft Media Foundation 플러그 인 관리자에 대한 포인터를 가져옵니다.
MFGetService 지정된 서비스 인터페이스에 대한 개체를 쿼리합니다. (MFGetService)
MFGetStrideForBitmapInfoHeader 비디오 형식의 최소 표면 보폭을 계산합니다.
MFGetSupportedMimeTypes 원본 확인자용으로 등록된 MIME 형식을 검색합니다.

MFGetSupportedSchemes 원본 확인자용으로 등록된 URL 스키마를 검색합니다.
MFGetSystemId 시스템 ID 데이터를 검색하기 위한 IMFSystemId 개체를 반환합니다.
MFGetSystemTime 시스템 시간을 반환합니다.
MFGetTimerPeriodicity MFAddPeriodicCallback 함수에 대한 타이머 간격을 검색합니다.
MFGetTopoNodeCurrentType 토폴로지 노드와 연결된 스트림의 미디어 형식을 가져옵니다.
MFGetUncompressedVideoFormat 압축되지 않은 비디오 형식의 FOURCC 또는 D3DFORMAT 값을 반환합니다.
MFGetWorkQueueMMCSSClass 현재 이 작업 큐와 연결된 MMCSS(멀티미디어 클래스 Scheduler Service) 클래스를 검색합니다. (MFGetWorkQueueMMCSSClass)
MFGetWorkQueueMMCSSPriority 작업 큐의 상대 스레드 우선 순위를 가져옵니다. (MFGetWorkQueueMMCSSPriority)
MFGetWorkQueueMMCSSTaskId 현재 이 작업 큐와 연결된 MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업 식별자를 검색합니다. (MFGetWorkQueueMMCSSTaskId)
MFHeapAlloc 메모리 블록을 할당합니다. (MFHeapAlloc)
MFHeapFree MFHeapAlloc 함수를 호출하여 할당된 메모리 블록을 해제합니다.
MFInitAMMediaTypeFromMFMediaType Media Foundation 미디어 형식에서 DirectShow AM_MEDIA_TYPE 구조를 초기화합니다.
MFInitAttributesFromBlob 바이트 배열에서 특성 저장소의 콘텐츠를 초기화합니다.
MFInitMediaTypeFromAMMediaType

DirectShow AM_MEDIA_TYPE 구조체에서 미디어 형식을 초기화합니다.
MFInitMediaTypeFromMFVideoFormat MFVIDEOFORMAT 구조체에서 미디어 형식을 초기화합니다.
MFInitMediaTypeFromMPEG1VideoInfo DirectShow MPEG1VIDEOINFO 구조체에서 미디어 형식을 초기화합니다.
MFInitMediaTypeFromMPEG2VideoInfo DirectShow MPEG2VIDEOINFO 구조체에서 미디어 형식을 초기화합니다.
MFInitMediaTypeFromVideoInfoHeader DirectShow VIDEOINFOHEADER 구조체에서 미디어 형식을 초기화합니다.
MFInitMediaTypeFromVideoInfoHeader2 DirectShow VIDEOINFOHEADER2 구조체에서 미디어 형식을 초기화합니다.
MFInitMediaTypeFromWaveFormatEx WAVEFORMATEX 구조체에서 미디어 형식을 초기화합니다.
MFInitVideoFormat DVD, 아날로그 텔레비전 또는 ATSC 디지털 텔레비전과 같은 표준 비디오 형식의 MFVIDEOFORMAT 구조를 초기화합니다.
MFInitVideoFormat_RGB 압축되지 않은 RGB 비디오 형식에 대한 MFVIDEOFORMAT 구조를 초기화합니다.
MFInvokeCallback 콜백 메서드를 호출하여 비동기 작업을 완료합니다. (MFInvokeCallback)
MFIsContentProtectionDeviceSupported 지정된 미디어 보호 시스템에 하드웨어 보안 프로세서가 지원되는지 여부를 확인합니다.
MFIsFormatYUV FOURCC 코드 또는 D3DFORMAT 값이 YUV 형식인지 여부를 쿼리합니다.
MFIsVirtualCameraTypeSupported 특정 가상 카메라가 플랫폼에서 지원되는지 여부를 나타내는 값을 반환합니다.
MFllMulDiv 각 용어가 64비트 부호 있는 값인 (a * b) + d) / c를 계산합니다.
MFLoadSignedLibrary

보호된 환경에 대해 서명된 동적 링크 라이브러리를 로드합니다.
MFLockDXGIDeviceManager 공유 Microsoft DXGI(DirectX Graphics Infrastructure) 디바이스 관리자를 잠깁니다.
MFLockPlatform MFShutdown 함수를 차단합니다.
MFLockSharedWorkQueue 공유 작업 큐를 가져오고 잠깁니다. (MFLockSharedWorkQueue)
MFLockWorkQueue 작업 큐를 잠깁니다. (MFLockWorkQueue)
MFMapDX9FormatToDXGIFormat Microsoft Direct3D 9 형식 식별자를 Microsoft DXGI(DirectX Graphics Infrastructure) 형식 식별자로 변환합니다.
MFMapDXGIFormatToDX9Format Microsoft DXGI(DirectX Graphics Infrastructure) 형식 식별자를 Microsoft Direct3D 9 형식 식별자로 변환합니다.
MFPERIODICCALLBACK MFAddPeriodicCallback 함수에 대한 콜백 함수입니다.
MFPutWaitingWorkItem 이벤트가 신호를 받을 때까지 기다리는 작업 항목을 큐에 대기합니다. (MFPutWaitingWorkItem)
MFPutWorkItem 작업 큐에 비동기 작업을 배치합니다. (MFPutWorkItem)
MFPutWorkItem2 지정된 우선 순위를 사용하여 작업 큐에 비동기 작업을 배치합니다. (MFPutWorkItem2)
MFPutWorkItemEx 작업 큐에 비동기 작업을 배치합니다. (MFPutWorkItemEx)
MFPutWorkItemEx2 지정된 우선 순위를 사용하여 작업 큐에 비동기 작업을 배치합니다. (MFPutWorkItemEx2)
MFRegisterLocalByteStreamHandler 호출자의 프로세스에 바이트 스트림 처리기를 등록합니다.
MFRegisterLocalSchemeHandler 호출자의 프로세스에 스키마 처리기를 등록합니다.

MFRegisterPlatformWithMMCSS 표준 Microsoft Media Foundation 플랫폼 작업 큐를 MMCSS(멀티미디어 클래스 스케줄러 서비스)에 등록합니다.
MFRemovePeriodicCallback MFAddPeriodicCallback 함수에 의해 설정된 콜백 함수를 취소합니다.
MFRequireProtectedEnvironment 미디어 프레젠테이션에 PMP(보호된 미디어 경로)가 필요한지 여부를 쿼리합니다.
MFScheduleWorkItem 지정된 간격 후에 완료될 비동기 작업을 예약합니다. (MFScheduleWorkItem)
MFScheduleWorkItemEx 지정된 간격 후에 완료될 비동기 작업을 예약합니다. (MFScheduleWorkItemEx)
MFSerializeAttributesToStream 특성 저장소의 내용을 스트림에 씁니다.
MFSerializePresentationDescriptor 프레젠테이션 설명자를 바이트 배열로 직렬화합니다.
MFSetAttribute2UINT32asUINT64 두 UINT32 값을 UINT64 특성 값으로 압축합니다.
MFSetAttributeRatio 비율을 64비트 특성 값으로 설정합니다.
MFSetAttributeSize 너비와 높이를 단일 64비트 특성 값으로 설정합니다.
MFShutdown Microsoft Media Foundation 플랫폼을 종료합니다.
MFShutdownObject Media Foundation 개체를 종료하고 개체와 연결된 모든 리소스를 해제합니다. (MFShutdownObject)
MFSplitSample 결합된 미디어 샘플을 개별 샘플로 다시 분할합니다.
MFStartup Microsoft Media Foundation을 초기화합니다.

MFTEnum 레지스트리에서 MFT(Media Foundation 변환)를 열거합니다.
MFTEnum2 지정된 검색 조건과 일치하는 MFT(Microsoft Media Foundation 변환) 목록을 가져옵니다.
MFTEnumEx 지정된 검색 조건과 일치하는 MFT(Microsoft Media Foundation 변환) 목록을 가져옵니다. 이 함수는 MFTEnum 함수를 확장합니다.
MFTGetInfo 레지스트리에서 MFT(Media Foundation 변환)에 대한 정보를 가져옵니다.
MFTranscodeGetAudioOutputAvailableTypes 오디오 인코더에서 출력 형식 목록을 가져옵니다.
MFTRegister MFT(Media Foundation 변환)에 대한 정보를 레지스트리에 추가합니다.
MFTRegisterLocal 호출자 프로세스에 MFT(Media Foundation 변환)를 등록합니다. (MFTRegisterLocal)
MFTRegisterLocalByCLSID 호출자 프로세스에 MFT(Media Foundation 변환)를 등록합니다. (MFTRegisterLocalByCLSID)
MFTUnregister MFT(Media Foundation 변환)를 등록 취소합니다.
MFTUnregisterLocal 호출자의 프로세스에서 하나 이상의 MFT(Media Foundation 변환)를 등록 취소합니다.
MFTUnregisterLocalByCLSID 호출자의 프로세스에서 MFT(Media Foundation 변환)를 등록 취소합니다.
MFUnlockDXGIDeviceManager 공유 Microsoft DXGI(DirectX Graphics Infrastructure) 디바이스 관리자의 잠금을 해제합니다.
MFUnlockPlatform MFLockPlatform 함수를 호출하여 잠긴 후 Media Foundation 플랫폼의 잠금을 해제합니다.
MFUnlockWorkQueue 작업 큐의 잠금을 해제합니다. (MFUnlockWorkQueue)

MFUnregisterPlatformFromMMCSS MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업에서 Microsoft Media Foundation 플랫폼 작업 큐의 등록을 취소합니다.
MFUnwrapMediaType MFWrapMediaType 함수에 의해 다른 미디어 형식으로 래핑된 미디어 형식을 검색합니다.
MFValidateMediaTypeSize 비디오 형식 블록에 대한 버퍼의 크기를 확인합니다.
MFWrapMediaType 다른 미디어 형식을 래핑하는 미디어 형식을 만듭니다.
monitorEnable 사용 작업이 완료되면 알림을 요청합니다.
NeedKey 초기화 데이터와 함께 키 또는 키가 필요하다는 것을 애플리케이션에 알깁니다.
NegotiateAuthenticatedChannelKeyExchange 인증된 채널에 대한 세션 키를 설정합니다.
negotiateCryptoSessionKeyExchange 암호화 세션에 대한 세션 키를 설정합니다.
negotiateKeyExchange 인증된 채널에 대한 세션 키를 설정합니다.
negotiateKeyExchange 암호화 세션에 대한 세션 키를 설정합니다.
NotifyEndOfSegment 스트림이 세그먼트의 끝에 도달했음을 미디어 싱크에 알깁니다.
NotifyPreroll 프레젠테이션 시계가 시작되도록 미디어 싱크에 알깁니다.
NotifyPresentationClock 미디어 세션에서 프레젠테이션 시계를 선택할 때 호출됩니다.
NotifyProcessInput 미디어 프로세서가 파이프라인 구성 요소에 입력 샘플을 배달하려고 할 때 호출됩니다.

NotifyProcessOutput 미디어 프로세서가 파이프라인 구성 요소에서 출력 샘플을 가져오면 호출됩니다.
NotifyPrune 할당자 샘플이 할당자에 의해 정리를 위해 해제되거나 할당자가 제거될 때 호출됩니다.
NotifyQualityEvent 미디어 싱크에서 MEQualityNotify 이벤트를 전달합니다.
NotifyQualityEvent 파이프라인 구성 요소가 MEQualityNotify 이벤트를 보낼 때 호출됩니다.
notifyRelease 비디오 샘플이 할당자로 반환되면 호출됩니다.
notifyTopology 미디어 세션이 새 토폴로지 재생을 시작할 때 호출됩니다.
OnAbort 원본 버퍼가 중단되었음을 나타내는 데 사용됩니다.
onActivitiesReport 새 IMFSensorActivitiesReport를 사용할 수 있을 때 미디어 파이프라인에서 발생합니다.
onAddSourceBuffer IMFSourceBuffer가 추가되었음을 나타냅니다.
OnChange 카메라 컨트롤 값이 변경될 때 발생합니다.
OnClockPause 프레젠테이션 시계가 일시 중지되면 호출됩니다.
OnClockRestart 일시 중지된 상태에서 프레젠테이션 클록이 동일한 위치에서 다시 시작될 때 호출됩니다.
onClockSetRate 프레젠테이션 시계에서 속도가 변경되면 호출됩니다.
OnClockStart 프레젠테이션 시계가 시작될 때 호출됩니다.

OnClockStop 프레젠테이션 시계가 중지되면 호출됩니다.
onError 관련 IMFCameraControlMonitor 내에서 복구할 수 없는 오류가 발생할 때 발생합니다.
onError 원본 버퍼에서 오류가 발생했음을 나타내는 데 사용됩니다.
OnEvent 캡처 엔진에서 캡처 이벤트를 애플리케이션에 알리기 위해 호출됩니다.
OnEvent 원본 판독기에서 미디어 원본에서 특정 이벤트를 수신할 때 호출됩니다.
OnFinalize IMFSinkWriter::Finalize 메서드가 완료되면 호출됩니다.
OnFlush IMFSourceReader::Flush 메서드가 완료되면 호출됩니다.
OnMarker IMFSinkWriter::P laceMarker 메서드가 완료되면 호출됩니다.
onOcclusionStateReport 카메라 폐색 상태가 변경되면 시스템에서 호출됩니다.
OnProcessSample 샘플 그래버 싱크가 새 미디어 샘플을 받을 때 호출됩니다. (IMFSampleGrabberSinkCallback.OnProcessSample)
OnProcessSampleEx 샘플 그래버 싱크가 새 미디어 샘플을 받을 때 호출됩니다. (IMFSampleGrabberSinkCallback2.OnProcessSampleEx)
OnReadSample IMFSourceReader::ReadSample 메서드가 완료되면 호출됩니다.
onRedirect 바이트 스트림이 URL로 리디렉션되는 경우 호출됩니다.
onRemoveSourceBuffer IMFSourceBuffer가 제거되었음을 나타냅니다.

OnSample 캡처 싱크가 샘플을 받을 때 호출됩니다.
OnSendingRequest 바이트 스트림이 URL을 요청할 때 호출됩니다.
OnServerCertificate 서버 SSL 인증서가 수신되었을 때 Media Foundation에서 호출됩니다. 는 서버 인증서가 허용되는지 여부를 나타냅니다.
OnSetPresentationClock 샘플 그래버 싱크에서 프레젠테이션 시계가 설정되면 호출됩니다.
OnShutdown 샘플 그래버 싱크가 종료되면 호출됩니다.
OnSourceClose 미디어 원본이 닫혔음을 나타내는 데 사용됩니다.
OnSourceEnded 미디어 원본이 종료되었음을 나타내는 데 사용됩니다.
onSourceEvent 열린 작업이 시작되거나 종료되면 네트워크 원본에서 호출됩니다.
onSourceOpen 미디어 원본이 열렸음을 나타내는 데 사용됩니다.
OnStreamError IMFSinkWriter에서 비동기 오류가 발생할 때 호출됩니다.
OnStreamError IMFSourceReader에서 비동기 오류가 발생할 때 호출됩니다.
OnSynchronizedEvent 샘플의 형식이 변경될 때 캡처 싱크에서 호출됩니다.
OnTransformChange IMFSourceReader의 변환 체인을 빌드하거나 수정할 때 호출됩니다. (IMFSinkWriterCallback2.OnTransformChange)
OnTransformChange

IMFSourceReader의 변환 체인을 빌드하거나 수정할 때 호출됩니다. (IMFSourceReaderCallback2.OnTransformChange)
OnUpdate 원본 버퍼가 업데이트 중임을 나타내는 데 사용됩니다.
OnUpdateEnd 원본 버퍼의 업데이트가 완료되었음을 나타내는 데 사용됩니다.
OnUpdateStart 원본 버퍼가 업데이트를 시작했음을 나타내는 데 사용됩니다.
onVideoStreamTick 미디어 엔진을 쿼리하여 새 비디오 프레임이 준비되었는지 확인합니다.
openDeviceHandle Microsoft Direct3D 디바이스에 대한 핸들을 가져옵니다.
openDeviceHandle Direct3D 디바이스에 대한 핸들을 가져옵니다.
OPMGetVideoOutputForTarget 지정된 어댑터의 VidPN 대상에 대한 비디오 출력 개체를 반환합니다.
OPMGetVideoOutputsFromHMONITOR 특정 HMONITOR 핸들과 연결된 각 물리적 모니터에 대한 OPM(Output Protection Manager) 개체를 만듭니다.
OPMGetVideoOutputsFromIDirect3DDevice9Object 특정 Direct3D 디바이스와 연결된 각 물리적 모니터에 대한 OPM(Output Protection Manager) 개체를 만듭니다.
Pack2UINT32AsUINT64 두 UINT32 값을 UINT64 값으로 압축합니다.
PackRatio 비율을 나타내는 두 개의 UINT32 값을 UINT64 값으로 압축합니다.
PackSize UINT32 너비 값과 UINT32 높이 값을 크기를 나타내는 UINT64 값으로 압축합니다.
ParseData 처리를 위해 ASF(패킷화된 고급 시스템 형식) 데이터를 ASF 분할자로 보냅니다.
ParseHeader

ASF 헤더의 정보를 구문 분석하고 해당 정보를 사용하여 ContentInfo 개체의 값을 설정합니다. 전체 헤더를 단일 버퍼로 전달하거나 여러 조각으로 보낼 수 있습니다.
일시 중지 미디어 세션을 일시 중지합니다.
일시 중지 미디어 원본의 모든 활성 스트림을 일시 중지합니다.
일시 중지 프레젠테이션 시계를 일시 중지합니다. 시계가 일시 중지되는 동안 클록 시간은 진행되지 않으며 시계의 IMFPresentationClock::GetTime은 시계가 일시 중지된 시간을 반환합니다.
일시 중지 재생을 일시 중지합니다. (IMFMediaEngine.Pause)
PDXVAHDSW_CreateDevice 소프트웨어 플러그 인 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스의 인스턴스를 만듭니다.
PDXVAHDSW_CreateVideoProcessor Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서 플러그 인 소프트웨어를 만듭니다.
PDXVAHDSW_DestroyDevice 소프트웨어 플러그 인 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스의 인스턴스를 제거합니다.
PDXVAHDSW_DestroyVideoProcessor 소프트웨어 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서를 제거합니다.
PDXVAHDSW_GetVideoProcessBltStatePrivate 소프트웨어 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서에서 프라이빗 블릿 상태를 가져옵니다.
PDXVAHDSW_GetVideoProcessorCaps 하나 이상의 소프트웨어 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서의 기능을 가져옵니다.
PDXVAHDSW_GetVideoProcessorCustomRates 소프트웨어 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서에서 지원하는 사용자 지정 속도를 가져옵니다.
PDXVAHDSW_GetVideoProcessorDeviceCaps 소프트웨어 플러그 인 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스의 기능을 가져옵니다.
PDXVAHDSW_GetVideoProcessorFilterRange

소프트웨어 플러그 인 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스에서 지원되는 이미지 필터 값 범위를 가져옵니다.
PDXVAHDSW_GetVideoProcessorInputFormats 소프트웨어 플러그 인 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스에서 지원하는 입력 형식을 가져옵니다.
PDXVAHDSW_GetVideoProcessorOutputFormats 소프트웨어 플러그 인 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스에서 지원하는 출력 형식을 가져옵니다.
PDXVAHDSW_GetVideoProcessStreamStatePrivate 소프트웨어 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서에서 프라이빗 스트림 상태를 가져옵니다.
PDXVAHDSW_Plugin Microsoft DirectX 비디오 가속 고화질(DXVA-HD)용 소프트웨어 플러그 인 디바이스를 초기화하는 함수에 대한 포인터입니다.
PDXVAHDSW_ProposeVideoPrivateFormat 소프트웨어 플러그 인 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스에서 프라이빗 표면 형식을 가져옵니다.
PDXVAHDSW_SetVideoProcessBltState 소프트웨어 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서에서 blit 작업에 대한 상태 매개 변수를 설정합니다.
PDXVAHDSW_SetVideoProcessStreamState 소프트웨어 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서의 입력 스트림에 대한 상태 매개 변수를 설정합니다.
PDXVAHDSW_VideoProcessBltHD 비디오 처리 블릿을 수행합니다.
PlaceEncodingParameters 새 인코더 설정 컬렉션을 사용하여 인코더 구성을 동적으로 업데이트합니다.
PlaceMarker 스트림에 마커를 배치합니다.
PlaceMarker 지정된 스트림에 표식을 배치합니다.
Play 재생을 시작합니다. (IMFMediaEngine.Play)

준비 인코더, 비디오 프로세서 및 미디어 싱크와 같은 필수 파이프라인 구성 요소를 로드하여 캡처 싱크를 준비합니다.
ProcessEvent 이 MFT(Media Foundation 변환)의 입력 스트림에 이벤트를 보냅니다.
ProcessFrames 하나 이상의 입력 샘플에서 작동하고 결과를 출력 화면에 쓰는 비디오 처리 작업을 명령 목록에 기록합니다. (ID3D12VideoProcessCommandList::P rocessFrames)
processFrames1 하나 이상의 입력 샘플에서 작동하고 결과를 출력 화면에 쓰는 비디오 처리 작업을 명령 목록에 기록합니다. (ID3D12VideoProcessCommandList1::P rocessFrames1)
processInput 이 MFT(Media Foundation 변환)의 입력 스트림에 데이터를 전달합니다.
processLicenseRequest 지정된 요청에 대한 라이선스 응답을 가져옵니다.
processMessage MFT(Media Foundation 변환)에 메시지를 보냅니다.
ProcessOutput 현재 입력 데이터에서 출력을 생성합니다.
ProcessSample 스트림에 샘플을 제공합니다. 미디어 싱크는 샘플을 처리합니다.
ProcessSample 입력 샘플을 멀티플렉서에 전달합니다.
쿼리 인증된 채널에 쿼리를 보냅니다.
QueryAuthenticatedChannel 인증된 채널에 쿼리를 보냅니다.
QueryHeader 이전에 보낸 HTTP 또는 HTTPS 요청에 대한 응답에서 지정된 HTTP 헤더의 값을 검색하기 위해 Microsoft Media Foundation에서 호출합니다.
QueueEvent

개체의 큐에 새 이벤트를 넣습니다.
QueueEvent 큐에 이벤트를 넣습니다.
QueueEventParamUnk 이벤트를 만들고, IUnknown 포인터를 이벤트 데이터로 설정하고, 이벤트를 큐에 넣습니다.
QueueEventParamVar 이벤트를 만들고 PROPVARIANT를 이벤트 데이터로 설정하고 이벤트를 큐에 넣습니다. IMFMediaEventGenerator::QueueEvent 구현 내에서 이 메서드를 호출합니다.
읽기 스트림에서 데이터를 읽습니다.
ReadGRL GLR(전역 해지 목록)을 가져옵니다.
ReadSample 미디어 원본에서 다음 샘플을 읽습니다.
RecommendVideoDecoderDownsampleParameters 드라이버는 입력 매개 변수에서 최적의 출력 다운샘플 매개 변수를 추천할 수 있습니다.
RegisterForEvent RegisterForEvent 메서드는 코덱에서 이벤트를 수신하도록 애플리케이션을 등록합니다. (ICodecAPI::RegisterForEvent)
RegisterMFTs 호출자 프로세스에서 하나 이상의 MFT(Media Foundation 변환)를 등록합니다.
RegisterNotifications 시간 제한 텍스트 알림 개체를 등록합니다.
RegisterProxyResult 현재 프록시 사용의 성공 또는 실패에 대한 레코드를 유지합니다.
RegisterThreads 개체에 해당 작업자 스레드를 MMCSS(멀티미디어 클래스 스케줄러 서비스)에 등록하도록 알깁니다. (IMFRealTimeClient.RegisterThreads)
registerThreadsEx 개체에 해당 작업자 스레드를 MMCSS(멀티미디어 클래스 스케줄러 서비스)에 등록하도록 알깁니다. (IMFRealTimeClientEx.RegisterThreadsEx)

RegisterVideoProcessorSoftwareDevice 소프트웨어 비디오 처리 디바이스를 등록합니다.
ReleaseBuffer IDirectXVideoDecoder::GetBuffer를 호출하여 얻은 버퍼를 해제합니다.
ReleaseDecoderBuffer ID3D11VideoContext::GetDecoderBuffer 메서드를 호출하여 가져온 버퍼를 해제합니다.
제거 IMFSourceBuffer에서 지정된 시간 범위로 정의된 미디어 세그먼트를 제거합니다.
제거 세션과 연결된 모든 라이선스 및 키를 제거합니다.
제거 시스템에서 가상 카메라를 사용하지 않도록 설정하고 등록을 취소합니다.
RemoveAllBuffers 샘플에서 모든 버퍼를 제거합니다.
RemoveAllControls 카메라 컨트롤 컬렉션에서 모든 컨트롤을 제거합니다.
RemoveAllEffects 캡처 스트림에서 모든 효과를 제거합니다.
RemoveAllEffects 모든 오디오 및 비디오 효과를 제거합니다.
RemoveAllElements 컬렉션에서 모든 항목을 제거합니다.
RemoveAllElements 목록에서 모든 원본 요소를 제거합니다.
RemoveAllPayloadExtensions 스트림에 대해 구성된 모든 페이로드 확장을 제거합니다.
RemoveAllStreams 캡처 싱크에서 모든 스트림을 제거합니다.

RemoveAllTransformsForStream 디코더를 제외하고 지정된 스트림에 대한 모든 MFT(Media Foundation 변환)를 제거합니다.
RemoveClockStateSink 시계에서 상태 변경 알림을 수신하는 개체의 등록을 취소합니다.
RemoveControl 카메라 컨트롤 컬렉션에서 지정된 컨트롤 집합 및 ID가 있는 컨트롤을 제거합니다.
RemoveControlSubscription IMFCameraControlNotify::OnChange 알림이 발생할 컨트롤 목록에서 카메라 컨트롤을 제거합니다.
RemoveEffect 캡처 스트림에서 효과를 제거합니다.
RemoveElement 컬렉션에서 개체를 제거합니다.
RemoveMutualExclusion 프로필에서 ASF(Advanced Systems Format) 상호 제외 개체를 제거합니다.
RemoveNode 토폴로지에서 노드를 제거합니다.
RemoveProfile 지정된 프로필 ID에 따라 지정된 프로필을 제거합니다.
RemoveRecord ASF(Advanced Systems Format) 상호 제외 개체에서 레코드를 제거합니다.
RemoveSourceBuffer IMFMediaSourceExtension 개체에서 관리하는 원본 버퍼 컬렉션에서 지정된 원본 버퍼를 제거합니다.
RemoveStream ASF(Advanced Systems Format) 프로필 개체에서 스트림을 제거합니다.
RemoveStream 이 인터페이스는 이 버전의 Media Foundation에서 구현되지 않습니다. 스트림 우선 순위 목록에서 스트림을 제거합니다.
RemoveStreamConfiguration 멀티플렉싱된 출력을 포함할 수 있는 하위 스트림 집합을 정의하는 스트림 구성의 등록을 취소합니다.

RemoveStreamFromRecord Advanced Systems Format 상호 제외 개체의 레코드에서 스트림 번호를 제거합니다.
RemoveStreamPrioritization IMFASFProfile::RemoveStreamPrioritization 메서드
RemoveStreamSink 미디어 싱크에서 스트림 싱크를 제거합니다.
RemoveTocByType RemoveTocByType 메서드는 TOC 파서 개체에서 지정된 형식의 모든 목차를 제거합니다.
RemoveTrack 지정된 식별자를 사용하여 시간 제한 텍스트 트랙을 제거합니다.
RequestAccess 스트림에서 지정된 작업을 수행할 수 있는 권한을 요청합니다.
requestSample 미디어 원본에서 샘플을 요청합니다.
다시 설정 새 명령 목록을 방금 만든 것처럼 명령 목록을 초기 상태로 다시 설정합니다. (ID3D12VideoDecodeCommandList::Reset)
다시 설정 새 명령 목록을 방금 만든 것처럼 명령 목록을 초기 상태로 다시 설정합니다. (ID3D12VideoEncodeCommandList::Reset)
다시 설정 새 명령 목록을 방금 만든 것처럼 명령 목록을 초기 상태로 다시 설정합니다. (ID3D12VideoProcessCommandList::Reset)
다시 설정 ITA(입력 신뢰 기관)를 초기 상태로 다시 설정합니다.
다시 설정 시간 제한 텍스트 알림 개체를 다시 설정합니다.
다시 설정 IMFD3D12SynchronizationObject::Reset 함수는 동기화 개체 상태를 다시 설정하여 할당자가 동기화 개체와 연결된 D3D12 리소스를 다시 사용할 수 있도록 합니다.

ResetDevice Microsoft Direct3D 디바이스를 설정하거나 장치 관리자에게 Direct3D 디바이스가 다시 설정되었음을 알 수 있습니다.
ResetDevice Direct3D 디바이스를 설정하거나 장치 관리자에게 Direct3D 디바이스가 다시 설정되었음을 알 수 있습니다.
ResetProtocolRolloverSettings 이 릴리스에서는 구현되지 않았습니다. (IMFNetSchemeHandlerConfig.ResetProtocolRolloverSettings)
ResolveEncoderOutputMetadata ID3D12VideoEncodeCommandList2::EncodeFrame에 대한 호출에서 출력 메타데이터를 읽을 수 있는 형식으로 확인합니다.
ResolveMotionVectorHeap EstimateMotion 메서드의 동작 벡터 출력을 하드웨어 종속 형식에서 비디오 동작 추정 API로 정의된 일관된 형식으로 변환합니다.
ResolveQueryData 쿼리에서 데이터를 추출합니다. (ID3D12VideoDecodeCommandList::ResolveQueryData)
ResolveQueryData 쿼리에서 데이터를 추출합니다. (ID3D12VideoEncodeCommandList::ResolveQueryData)
ResolveQueryData 쿼리에서 데이터를 추출합니다. (ID3D12VideoProcessCommandList::ResolveQueryData)
ResourceBarrier 여러 액세스를 리소스에 동기화해야 한다는 것을 드라이버에 알깁니다. (ID3D12VideoDecodeCommandList::ResourceBarrier)
ResourceBarrier 여러 액세스를 리소스에 동기화해야 한다는 것을 드라이버에 알깁니다. (ID3D12VideoEncodeCommandList::ResourceBarrier)
ResourceBarrier 여러 액세스를 리소스에 동기화해야 한다는 것을 드라이버에 알깁니다. (ID3D12VideoProcessCommandList::ResourceBarrier)
SaveDefaults 제공된 카메라 컨트롤 기본값 컬렉션을 저장합니다.
SaveReference

프로세스 경계를 넘어 인터페이스를 마샬링하는 데 필요한 데이터를 저장합니다.
검색 스트림의 현재 위치를 지정된 오프셋으로 이동합니다.
SelectStream 프레젠테이션에서 스트림을 선택합니다.
SelectStreams ASF(Advanced Systems Format) 분할자로 구문 분석할 스트림을 설정합니다.
SelectTrack 시간 제한 텍스트 구성 요소에서 텍스트 트랙을 선택하거나 선택 취소합니다.
SendCameraProperty 내부 IKsControl::KsProperty 메서드 주위의 래퍼입니다.
SendStreamTick 입력 스트림의 간격을 나타냅니다.
SetAllDefaults SetAllDefaults 메서드는 모든 코덱 속성을 기본값으로 다시 설정합니다. (ICodecAPI::SetAllDefaults)
SetAllDefaultsWithNotify SetAllDefaultsWithNotify 메서드는 모든 코덱 속성을 기본값으로 다시 설정하고 변경된 속성 목록을 반환합니다. (ICodecAPI::SetAllDefaultsWithNotify)
SetAllocator 샘플의 소유자를 설정합니다.
SetAllSettings SetAllSettings 메서드는 스트림에서 코덱 속성을 읽고 코덱에 설정합니다. (ICodecAPI::SetAllSettings)
SetAllSettingsWithNotify SetAllSettingsWithNotify 메서드는 스트림에서 코덱 속성을 읽고, 코덱에 설정하고, 변경된 속성 목록을 반환합니다. (ICodecAPI::SetAllSettingsWithNotify)
SetAllVolumes 오디오 스트림의 모든 채널에 대한 개별 볼륨 수준을 설정합니다.
SetAppendWindowEnd 추가 창의 끝에 대한 타임스탬프를 설정합니다.

SetAppendWindowStart 추가 창의 시작에 대한 타임스탬프를 설정합니다.
SetApplicationCertificate 애플리케이션의 인증서를 설정합니다.
SetAudioAttributes 코드 변환 프로필에서 오디오 스트림 구성 설정을 설정합니다.
SetAudioEndpointRole SetSource 또는 Load에 대한 다음 호출에 사용되는 오디오 디바이스 엔드포인트를 설정합니다.
SetAudioStreamCategory SetSource 또는 Load에 대한 다음 호출에 대한 오디오 스트림 범주를 설정합니다.
SetAutoPlay 미디어 엔진이 자동으로 재생을 시작할지 여부를 지정합니다.
SetBalance 오디오 균형을 설정합니다. (IMFMediaEngineEx.SetBalance)
SetBlob 바이트 배열을 키와 연결합니다.
setBorderColor 테두리 색을 설정합니다.
SetBufferFullnessBits 이 릴리스에서는 구현되지 않았습니다. (IWMCodecLeakyBucket.SetBufferFullnessBits)
SetBufferingParams 버퍼링 매개 변수를 설정합니다.
SetBufferSizeBits 버퍼 크기를 비트 단위로 설정합니다.
SetCacheLimit 캐시 크기를 제한합니다.
SetCallback 비디오 샘플이 할당자에 반환될 때마다 알림을 받는 콜백 개체를 설정합니다.

SetChannelVolume 오디오 스트림에서 지정된 채널의 볼륨 수준을 설정합니다.
SetClipRegion 원본 사각형을 설정합니다. (IWMResizerProps.SetClipRegion)
SetConstrictionSize 출력에서 수행할 다운샘플링의 양을 지정합니다.
setContainerAttributes 코드 변환 프로필에서 컨테이너 구성 설정을 설정합니다.
SetContentEnabler 호출자가 CDM(콘텐츠 암호 해독 모듈)에서 사용할 IMFContentEnabler 인터페이스를 지정할 수 있도록 허용합니다.
SetContentProtectionManager CPM(콘텐츠 보호 관리자)을 설정합니다.
SetContext SetContext 메서드는 호출자가 제공한 컨텍스트 블록을 목차와 연결합니다.
SetCueTimeOffset 오프셋을 큐 시간으로 설정합니다.
SetCurrentDeviceMediaType 캡처 스트림의 출력 형식을 설정합니다.
SetCurrentLength 버퍼에 있는 유효한 데이터의 길이를 설정합니다.
SetCurrentMediaType 개체의 미디어 형식을 설정합니다.
SetCurrentMediaType 스트림의 미디어 형식을 설정합니다.
SetCurrentPosition 현재 읽기 또는 쓰기 위치를 설정합니다.
SetCurrentPosition 미디어 소스에서 새 위치를 찾습니다.

SetCurrentTime 새 재생 위치를 찾습니다.
SetCurrentTimeEx 지정된 MF_MEDIA_ENGINE_SEEK_MODE 사용하여 새 재생 위치를 찾습니다.
setCustomSink 미리 보기용 사용자 지정 미디어 싱크를 설정합니다.
setCustomSink 녹음/녹화를 위한 사용자 지정 미디어 싱크를 설정합니다.
SetD3DManager 미디어 원본에서 Microsoft DXGI(DirectX Graphics Infrastructure) 디바이스 관리자에 대한 포인터를 설정합니다.
SetDefaultAllocator 지정된 출력 스트림에 사용할 기본 샘플 할당자를 설정합니다.
SetDefaultPlaybackRate 기본 재생 속도를 설정합니다.
SetDescription SetDescription 메서드는 설명을 목차와 연결합니다.
SetDescriptionData SetDescriptionData 메서드는 호출자가 제공한 데이터 블록을 항목과 연결합니다.
SetDescriptor SetDescriptor 메서드는 설명자를 목차와 연결합니다.
SetDescriptor SetDescriptor 메서드는 설명자를 항목과 연결합니다.
SetDestinationRectangle 대상 사각형을 설정합니다.
SetDirectxDeviceManager DirectX 디바이스 관리자를 설정합니다.
SetDirectXManager 사용할 비디오 미디어 싱크의 Direct3D 디바이스 관리자를 지정합니다.

SetDisabled 차단된 목록에 CLSID(클래스 식별자)를 추가하거나 목록에서 CLSID를 제거합니다. (IMFPluginControl.SetDisabled)
SetDisplayName 오디오 세션의 표시 이름을 설정합니다. Windows 볼륨 컨트롤에 이 이름이 표시됩니다.
setDouble double 값을 키와 연결합니다.
SetDropMode 드롭 모드를 설정합니다. 드롭 모드에서 구성 요소는 드롭 모드의 수준에 따라 다소 공격적으로 샘플을 삭제합니다.
setDuration 미디어 원본의 기간을 100나노초 단위로 설정합니다.
SetEndOfStream 미디어 스트림의 끝에 도달했음을 나타냅니다.
setErrorCode 현재 오류 코드를 설정합니다.
setErrorCode 오류 코드를 설정합니다.
SetExtendedErrorCode 확장 오류 코드를 설정합니다.
SetFlags 캡처 디바이스에서 속성의 플래그를 설정합니다.
SetFlags 인덱서 옵션을 설정합니다.
SetFlags 멀티플렉서 옵션을 설정합니다.
SetFlags ASF(Advanced Systems Format) 분할자에서 옵션 플래그를 설정합니다.
SetFullCroppingParam 원본 및 대상 사각형을 설정합니다. (IWMColorConvProps.SetFullCroppingParam)

SetFullCropRegion 원본 및 대상 사각형을 설정합니다. (IWMResizerProps.SetFullCropRegion)
SetGood 제공된 자격 증명 개체가 인증 챌린지를 성공적으로 통과했는지 여부를 보고합니다.
SetGood 인증 챌린지에서 사용자의 자격 증명이 성공했는지 여부를 지정합니다. 네트워크 원본은 이 메서드를 호출하여 사용자의 자격 증명이 인증되었는지 여부를 애플리케이션에 알릴 수 있습니다.
SetGroupingParam 오디오 세션을 세션 그룹에 할당합니다.
SetGUID GUID 값을 키와 연결합니다.
SetHalfFilterLength 출력의 품질을 지정합니다.
setHurryup 비디오 디코더의 속도 모드를 설정합니다.
SetIconPath 오디오 세션의 아이콘 리소스를 설정합니다. Windows 볼륨 컨트롤에 이 아이콘이 표시됩니다.
SetID 버퍼가 나타내는 공간 오디오 개체의 ID를 설정합니다.
setInBandEnabled 인밴드 모드를 사용하거나 사용하지 않도록 설정합니다.
SetIndexByteStreams 인덱싱할 바이트 스트림을 추가합니다.
SetIndexStatus 스트림에 대한 인덱스를 구성합니다.
SetInputMediaType 싱크 기록기에서 스트림의 입력 형식을 설정합니다.
SetInputPrefType 이 노드의 입력 스트림에 대한 기본 설정 미디어 형식을 설정합니다.

SetInputType 이 MFT(Media Foundation 변환)에서 입력 스트림의 미디어 형식을 설정, 테스트 또는 지웁니다.
SetInterlaceMode SetInterlaceMode 메서드는 입력 비디오 스트림이 인터레이스되는지 여부를 지정합니다.
setItem 지정된 키를 사용하여 특성 값을 추가합니다.
SetKeyFrame 현재 프레임이 키 프레임으로 인코딩되도록 지정합니다.
SetLabel 시간 제한 텍스트 트랙의 레이블을 설정합니다.
setLanguage 메타데이터를 설정하고 검색하기 위한 언어를 설정합니다.
setLength 스트림의 길이를 설정합니다.
SetLoop 미디어 엔진이 재생을 반복하는지 여부를 지정합니다.
SetMarker 내부 전용입니다. 직접 호출할 수 없습니다. (ID3D12VideoDecodeCommandList::SetMarker)
SetMarker 내부 전용입니다. 직접 호출할 수 없습니다. (ID3D12VideoEncodeCommandList::SetMarker)
SetMarker 내부 전용입니다. 직접 호출할 수 없습니다.
setMasterVolume 마스터 볼륨 수준을 설정합니다.
SetMediaKeys 미디어 엔진에 사용할 미디어 키 개체를 설정합니다.
SetMediaType ASF(Advanced Systems Format) 스트림 구성 개체의 미디어 형식을 설정합니다.

SetMirror 비디오 이미지를 대칭 이동할지 여부를 지정합니다.
SetMirrorState 비디오 미리 보기 스트림의 미러링을 사용하거나 사용하지 않도록 설정합니다. (IMFCapturePreviewSink.SetMirrorState)
SetMirrorState 비디오 미리 보기 스트림의 미러링을 사용하거나 사용하지 않도록 설정합니다. (IMFCaptureSource.SetMirrorState)
SetMode 입력 비디오 스트림이 인터레이스되는지 여부를 지정합니다.
setMute 오디오를 음소거하거나 음소거 해제합니다. (IMFSimpleAudioVolume.SetMute)
SetMuted 오디오를 음소거하거나 음소거 해제합니다. (IMFMediaEngine.SetMuted)
SetNativeMediaType 미디어 원본의 스트림에 대한 네이티브 미디어 형식을 설정합니다.
SetObject 이 노드와 연결된 개체를 설정합니다.
SetOPMWindow 출력 링크 보호를 받아야 하는 창을 지정합니다.
SetOutputBounds 클라이언트가 출력에 필요한 타임스탬프를 설정합니다.
SetOutputByteStream 코드 변환 미디어 싱크에 대한 출력 바이트 스트림을 설정합니다.
SetOutputByteStream 스틸 이미지 데이터를 받을 바이트 스트림을 지정합니다.
SetOutputByteStream 기록에 대한 데이터를 받을 바이트 스트림을 지정합니다.
SetOutputFile 인코딩된 출력 파일의 이름을 설정합니다.

SetOutputFileName 스틸 이미지의 출력 파일 이름을 지정합니다.
SetOutputFileName 녹음/녹화에 대한 출력 파일의 이름을 지정합니다.
SetOutputMediaType 레코드 싱크 또는 미리 보기 싱크의 출력 미디어 형식을 동적으로 설정합니다.
SetOutputMutexSelection 출력과 연결된 상호 제외 개체에 사용할 상호 제외 레코드를 선택합니다.
SetOutputOverride 출력의 선택 상태를 설정하고 다른 선택 조건을 재정의합니다.
SetOutputPrefType 이 노드의 출력 스트림에 대한 기본 설정 미디어 형식을 설정합니다.
SetOutputType 이 MFT(Media Foundation 변환)에서 출력 스트림의 미디어 형식을 설정, 테스트 또는 지웁니다.
SetPartialOutputType 코덱에 코덱 데이터 없이 출력 미디어 형식을 제공합니다. 이렇게 하면 코덱이 프라이빗 데이터를 생성할 수 있습니다.
SetPassword 암호를 설정합니다.
SetPlaybackRate 현재 재생 속도를 설정합니다.
SetPMPHost IMFPMPHost 인터페이스에 대한 포인터를 제공합니다.
SetPMPHost 미디어 원본이 PMP 프로세스에서 개체를 만들 수 있도록 하는 IMFPMPHostApp 인터페이스에 대한 포인터를 설정합니다.
SetPMPHostApp 호출자가 보호된 프로세스를 나타내는 IMFPMPHostApp 인터페이스를 지정할 수 있습니다.
SetPolicy

출력 신뢰 기관(OTA)에서 하나 이상의 정책 개체를 설정합니다.
SetPolicy 미디어 원본 및 변환이 열거되는 정책을 설정합니다.
setPredication 조건자 값이 지정된 작업을 통과하는 경우 후속 명령을 수행하지 않도록 지정합니다. (ID3D12VideoDecodeCommandList::SetPredication)
setPredication 조건자 값이 지정된 작업을 통과하는 경우 후속 명령을 수행하지 않도록 지정합니다. (ID3D12VideoEncodeCommandList::SetPredication)
setPredication 조건자 값이 지정된 작업을 통과하는 경우 후속 명령을 수행하지 않도록 지정합니다. (ID3D12VideoProcessCommandList::SetPredication)
SetPreferredClsid 기본 설정 목록에 CLSID(클래스 식별자)를 추가하거나 목록에서 CLSID를 제거합니다. (IMFPluginControl.SetPreferredClsid)
SetPreload 미리 로드 플래그를 설정합니다.
SetPresentationClock IMFPresentationClock 인스턴스를 앱에 제공하기 위해 미디어 파이프라인에서 호출됩니다.
SetPresentationClock 미디어 싱크의 프레젠테이션 시계를 설정합니다.
SetPrivateData 비디오 디바이스에서 개인 데이터를 설정하고 해당 데이터를 GUID와 연결합니다.
SetPrivateDataInterface 비디오 디바이스에서 프라이빗 IUnknown 포인터를 설정하고 해당 포인터를 GUID와 연결합니다.
SetProfile 코드 변환 싱크 활성화 개체의 코드 변환 프로필을 설정합니다.
SetProfile 프로필 개체의 프로필 데이터를 사용하여 ContentInfo 개체의 설정을 구성합니다.
SetProperty 메타데이터 속성의 값을 설정합니다.

SetProtectedResourceSession 비디오 디코딩 명령 목록의 후속 명령으로 보호된 리소스에 액세스할 수 있는지 여부를 지정합니다.
SetProtectedResourceSession 비디오 인코딩 명령 목록의 후속 명령으로 보호된 리소스에 액세스할 수 있는지 여부를 지정합니다.
SetProtectedResourceSession 비디오 프로세스 명령 목록의 후속 명령으로 보호된 리소스에 액세스할 수 있는지 여부를 지정합니다.
SetQualityLevel 품질 수준을 설정합니다. 품질 수준은 구성 요소가 샘플을 사용하거나 생성하는 방법을 결정합니다.
setRate 재생 속도를 설정합니다. (IMFRateControl.SetRate)
SetRealTimeMode SetSource 또는 Load에 대한 다음 호출에 사용되는 실시간 모드를 설정합니다.
SetReconstructedVideoFrame 참고: 이 메서드는 사용되지 않으므로 사용하면 안 됩니다. 현재 재구성된 비디오 프레임을 복원합니다.
setRenderHandle 미리 보기의 창을 지정합니다.
setRenderSurface 미리 보기용 Microsoft DirectComposition 시각적 개체를 지정합니다.
SetResizerQuality 고품질 비디오를 생성하는 알고리즘을 사용할지 아니면 더 빠른 알고리즘을 사용할지 지정합니다.
SetRotation 비디오를 올바른 방향으로 회전할지 여부를 지정합니다.
SetRotation 비디오 미리 보기 스트림을 회전합니다.
SetRotation 녹화된 비디오 스트림을 회전합니다.
SetRotationOverride 비디오 프로세서에서 수행되는 회전 작업을 재정의합니다.

SetSample 미디어 스트림 원본에 대한 샘플을 설정합니다.
SetSampleCallback 스틸 이미지 데이터를 수신하도록 콜백을 설정합니다.
SetSampleCallback 하나의 스트림에 대한 미리 보기 데이터를 수신하도록 콜백을 설정합니다.
SetSampleCallback 하나의 스트림에 대한 기록 데이터를 수신하도록 콜백을 설정합니다.
SetSampleDuration 샘플의 기간을 설정합니다.
SetSampleFlags 샘플과 연결된 플래그를 설정합니다. 현재 플래그가 정의되어 있지 않습니다.
SetSampleTime 샘플의 프레젠테이션 시간을 설정합니다.
SetSelectedStyle SAMI 미디어 원본에서 현재 스타일을 설정합니다.
SetSensorDeviceMode 센서 디바이스의 공유 모드를 컨트롤러 또는 공유로 지정하는 값을 설정합니다.
SetServer 이 IMFHttpDownloadSession 인스턴스에서 만든 모든 요청에 공통적인 매개 변수를 지정하기 위해 Microsoft Media Foundation에서 호출합니다.
SetServerCertificate 라이선스 서버에 메시지를 암호화하는 데 사용할 서버 인증서를 제공합니다.
SetSource 미디어 리소스의 URL을 설정합니다.
SetSource 원본 스트림을 설정합니다.
SetSourceElements 미디어 원본 목록을 설정합니다.

SetSourceFromByteStream 바이트 스트림에서 미디어 리소스를 엽니다.
setSourceRectangle 원본 사각형을 설정합니다. (IMFVideoProcessorControl.SetSourceRectangle)
setStatus 비동기 작업의 상태를 설정합니다. (IMFAsyncResult.SetStatus)
SetStereo3DFramePackingMode 스테레오스코픽 3D 비디오의 경우 비디오 프레임 내에서 두 보기의 레이아웃을 설정합니다.
setStereo3DRenderMode 입체 3D 비디오의 경우 미디어 엔진이 3D 비디오 콘텐츠를 렌더링하는 방법을 지정합니다.
SetStream 프로필에 스트림을 추가하거나 기존 스트림을 다시 구성합니다.
SetStreamNumber 스트림에 스트림 번호를 할당합니다.
SetStreamSelection 재생을 위해 스트림을 선택하거나 선택 취소합니다.
SetStreamSelection 하나 이상의 스트림을 선택하거나 선택 취소합니다.
SetStreamSelectorFlags 스트림 선택기 옵션을 설정합니다.
setString 와이드 문자열을 키와 연결합니다.
SetSubEntries SetSubEntries는 항목 집합을 이 항목의 하위 엔터티로 식별합니다.
SetSurfaceType 디코더가 DVXA(DirectX Video Acceleration) 2.0에 사용할 비디오 화면 유형을 설정합니다.
SetSyncTolerance 다양한 스트림의 샘플이 동기화되지 않을 수 있는 최대 시간을 설정합니다.

SetTargetMediaType 싱크 기록기가 인코딩하는 대상 미디어 형식을 동적으로 변경합니다.
SetTimelineMarkerTimer 미디어 엔진이 마커 이벤트를 보낼 프레젠테이션 시간을 지정합니다.
SetTimer 지정된 시간에 콜백을 호출하는 타이머를 설정합니다.
SetTimeSource 프레젠테이션 시계의 시간 원본을 설정합니다. 시간 원본은 현재 시간을 제공하여 시계를 구동하는 개체입니다.
SetTimeStampOffset IMFSourceBuffer에 추가된 미디어 세그먼트의 타임스탬프 오프셋을 설정합니다.
SetTitle SetTitle 메서드는 항목의 제목을 설정합니다.
setTopology 미디어 세션에서 토폴로지 설정
SetTopoNodeID 노드의 식별자를 설정합니다.
SetType ASF(Advanced Systems Format) 상호 제외 개체로 표현되는 상호 제외 유형을 설정합니다.
SetType 버퍼가 나타내는 공간 오디오 개체의 형식을 설정합니다.
setUINT32 UINT32 값을 키와 연결합니다.
SetUINT64 UINT64 값을 키와 연결합니다.
SetUnknown IUnknown 포인터를 키와 연결합니다.
SetUnknown 미디어 버퍼 개체에 임의의 IUnknown 포인터를 저장합니다.

설치 IMFSystemId를 설정합니다.
SetupSession 암호화된 세션 키를 클라이언트에 전달합니다.
SetUser 사용자 이름을 설정합니다.
SetUserChannelMtx 채널 행렬을 지정합니다.
SetUserOptions 사용자 자격 증명을 저장하는 방법을 지정합니다.
SetUseSampleExtensions 코덱이 샘플 확장을 지원하는지 여부를 구성합니다.
SetValue SetValue 메서드는 코덱 속성의 값을 설정합니다. (ICodecAPI::SetValue)
SetValueWithNotify SetValueWithNotify 메서드는 코덱에 속성을 설정하고 결과적으로 변경된 다른 속성의 목록을 반환합니다. (ICodecAPI::SetValueWithNotify)
SetVideoAttributes 코드 변환 프로필에서 비디오 스트림 구성 설정을 설정합니다.
SetVideoProcessBltState Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스에서 blit 작업에 대한 상태 매개 변수를 설정합니다.
SetVideoProcessStreamState Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스의 입력 스트림에 대한 상태 매개 변수를 설정합니다.
SetVolume 오디오 볼륨 수준을 설정합니다.
SetWorkQueue 이 개체를 포함하는 토폴로지 분기의 작업 큐를 지정합니다.
setWorkQueueEx 이 개체가 비동기 작업 항목에 사용해야 하는 작업 큐를 지정합니다.

shareResources 미디어 엔진이 프레임 서버 모드에서 보호된 콘텐츠에 액세스할 수 있도록 합니다.
ShouldDelayTheLoadEvent 웹 오디오에 연결하면 페이지의 로드 이벤트가 지연되는지 여부를 나타내는 값을 가져옵니다.
ShouldTransferSource 원본을 전송해야 하는지 여부를 지정합니다.
종료 IMFCameraControlMonitor::Shutdown 함수는 카메라 컨트롤 모니터를 종료하고 관련 리소스를 정리합니다.
종료 미디어 세션을 종료하고 미디어 세션에서 사용하는 모든 리소스를 해제합니다.
종료 미디어 싱크를 종료하고 사용 중인 리소스를 해제합니다.
종료 미디어 원본을 종료하고 사용 중인 리소스를 해제합니다.
종료 미디어 세션이 종료되면 호출됩니다.
종료 Media Foundation 개체를 종료하고 개체와 연결된 모든 리소스를 해제합니다. (IMFShutdown.Shutdown)
종료 이벤트 큐를 종료합니다.
종료 미디어 엔진을 종료하고 사용 중인 리소스를 해제합니다.
종료 IMFMediaKeys::Shutdown(mfmediaengine.h) 메서드는 연결된 CDM(콘텐츠 암호 해독 모듈)을 종료합니다.
종료 이미지 공유 엔진을 종료합니다.
종료 동기화 개체를 종료합니다.

종료 가상 카메라의 모든 내부 리소스를 해제합니다.
ShutdownObject 만든 개체를 종료합니다.
SignalEventOnFinalResourceRelease 연결된 D3D12 리소스가 무료이고 재활용, 재사용 또는 소멸될 때 설정되는 이벤트 핸들을 저장합니다.
SignalEventOnResourceReady D3D12 리소스가 준비되면 설정될 이벤트 핸들을 저장합니다.
시작 IMFCameraControlMonitor::AddControlSubscription에 등록된 컨트롤의 변경 내용에 대한 IMFCameraControlNotify::OnChange 이벤트를 발생시켜 카메라 컨트롤 모니터를 시작합니다.
시작 카메라 폐색 상태 모니터를 시작합니다.
시작 미디어 세션을 시작합니다.
시작 재생을 시작할 위치를 지정하여 미디어 원본을 시작, 검색 또는 다시 시작합니다.
시작 프레젠테이션 시계를 시작합니다.
시작 센서 활동 모니터를 시작합니다.
시작 가상 카메라를 만들고 등록합니다.
startInitialization OPM(출력 보호 관리자) 세션의 초기화 시퀀스를 시작합니다.
StartPreview 미리 보기를 시작합니다.
startRecord 파일에 오디오 및/또는 비디오 녹음을 시작합니다.

StartSessionKeyRefresh 세션 키를 새로 고치는 데 사용할 수 있는 난수를 가져옵니다. (ID3D11VideoContext.StartSessionKeyRefresh)
StartSessionKeyRefresh 세션 키를 새로 고치는 데 사용할 수 있는 난수를 가져옵니다. (IDirect3DCryptoSession9.StartSessionKeyRefresh)
StartStreaming 미디어 세션이 스트리밍을 시작하기 전에 스트리밍 미디어 클라이언트에서 호출되어 바이트 오프셋 또는 시간 오프셋을 지정합니다.
중지 카메라 컨트롤 모니터를 중지합니다.
중지 카메라 폐색 상태 모니터를 중지합니다.
중지 미디어 세션을 중지합니다.
중지 미디어 원본의 모든 활성 스트림을 중지합니다.
중지 프레젠테이션 시계를 중지합니다. 클록이 중지되는 동안 클록 시간은 진행되지 않으며 클록의 IMFPresentationClock::GetTime 메서드는 0을 반환합니다.
중지 센서 활동 모니터를 중지합니다.
중지 등록된 가상 카메라를 사용하지 않도록 설정하여 앱이 가상 카메라를 열거하거나 활성화하지 못하도록 차단합니다.
StopBackgroundTransfer 로컬 캐시에 대한 데이터의 백그라운드 전송을 중지합니다.
StopBuffering 진행 중인 버퍼링을 중지합니다.
StopPreview 미리 보기를 중지합니다.
StopRecord

기록을 중지합니다.
SubmitDecoderBuffers 디코딩을 위해 하나 이상의 버퍼를 제출합니다. (ID3D11VideoContext.SubmitDecoderBuffers)
SubmitDecoderBuffers1 디코딩을 위해 하나 이상의 버퍼를 제출합니다. (ID3D11VideoContext1.SubmitDecoderBuffers1)
takePhoto 비디오 스트림에서 스틸 이미지를 캡처합니다.
testDevice Microsoft Direct3D 디바이스 핸들이 유효한지 테스트합니다.
testDevice Direct3D 디바이스 핸들이 유효한지 테스트합니다.
TimeSeek 바이트 스트림에서 새 위치를 찾습니다.
TrackAdded 텍스트 트랙이 추가되면 호출됩니다.
TrackRemoved 텍스트 트랙이 제거될 때 호출됩니다.
TrackSelected 트랙을 선택하거나 선택 취소할 때 호출됩니다.
TransferVideoFrame 현재 비디오 프레임을 DXGI 표면 또는 WIC 비트맵에 복사합니다.
TransferVideoFrame 보호된 비디오 프레임을 DXGI 화면에 복사합니다.
uninitializeSampleAllocator 할당된 모든 비디오 샘플을 릴리스합니다.
잠금 해제 애플리케이션에서 사용할 수 있도록 MFT(Media Foundation 변환)의 잠금을 해제합니다.
잠금 해제

이전에 잠긴 버퍼의 잠금을 해제합니다. IMFMediaBuffer::Lock에 대한 모든 호출에 대해 이 메서드를 한 번 호출합니다.
Unlock2D 이전에 잠긴 버퍼의 잠금을 해제합니다. IMF2DBuffer::Lock2D에 대한 각 호출에 대해 이 메서드를 한 번 호출합니다.
UnlockControlData LockControlData를 호출하여 잠금 해제된 컨트롤 데이터 버퍼의 잠금을 해제합니다.
unlockDevice Microsoft Direct3D 디바이스의 잠금을 해제합니다.
unlockDevice Direct3D 디바이스의 잠금을 해제합니다.
UnlockPayload 캡처 디바이스 컨트롤에 포함된 원시 페이로드의 잠금을 해제합니다.
UnlockProcess PMP(보호된 미디어 경로) 프로세스의 잠금 수를 줄입니다. IMFPMPHost::LockProcess에 대한 각 호출에 대해 이 메서드를 한 번 호출합니다.
UnlockProcess PMP(보호된 미디어 경로) 프로세스의 잠금 수를 줄입니다. IMFPMPHostApp::LockProcess에 대한 각 호출에 대해 이 메서드를 한 번 호출합니다.
UnlockProcess PMP(보호된 미디어 경로) 프로세스의 잠금 수를 줄입니다. IMFPMPServer::LockProcess에 대한 각 호출에 대해 이 메서드를 한 번 호출합니다.
UnlockStore IMFAttributes::LockStore 메서드를 호출한 후 특성 저장소의 잠금을 해제합니다. 개체가 잠금 해제되는 동안 여러 스레드가 개체의 특성에 액세스할 수 있습니다.
Unpack2UINT32AsUINT64 UINT64 값에서 하위 순서 및 상위 UINT32 값을 가져옵니다.
UnpackRatio 비율을 나타내는 UINT64 값에서 낮은 순서 및 상위 UINT32 값을 가져옵니다.
unpackSize 크기를 나타내는 UINT64 값에서 하위 순서 및 상위 UINT32 값을 가져옵니다.
UnregisterForEvent

UnregisterForEvent 메서드는 지정된 인코더 이벤트에 대한 애플리케이션의 등록을 취소합니다. (ICodecAPI::UnregisterForEvent)
UnregisterThreads MMCSS(멀티미디어 클래스 스케줄러 서비스)에서 해당 작업자 스레드의 등록을 취소하도록 개체에 알깁니다. (IMFRealTimeClient.UnregisterThreads)
UnregisterThreads MMCSS(멀티미디어 클래스 스케줄러 서비스)에서 해당 작업자 스레드의 등록을 취소하도록 개체에 알깁니다. (IMFRealTimeClientEx.UnregisterThreads)
업데이트 지정된 키 시스템에 대한 콘텐츠 암호 해독 모듈에 필요한 모든 관련 데이터를 사용하여 키 값을 전달합니다.
업데이트 CDM(콘텐츠 암호 해독 모듈) 세션에 라이선스를 포함한 메시지를 제공합니다.
UpdateAccess 지정된 작업을 수행할 OTA(출력 신뢰 기관)의 수가 변경되면 ITA(입력 신뢰 기관)에 알릴 수 있습니다.
UpdateNodeAttributes 현재 토폴로지에서 하나 이상의 노드 특성을 업데이트합니다.
UpdateTopology 터미널 서비스 환경에서 사용할 토폴로지 수정 (IMFRemoteDesktopPlugin.UpdateTopology)
UpdateTopology 큐의 토폴로지 업데이트
UpdateTopologyFlags 큐의 토폴로지 플래그를 업데이트합니다.
UpdateVideo 비디오 프레임을 업데이트합니다. (IMFCapturePreviewSink.UpdateVideo)
UpdateVideoStream 비디오의 원본 사각형, 대상 사각형 및 테두리 색을 업데이트합니다.
VideoProcessBlt 하나 이상의 입력 샘플에서 비디오 프로세스 작업을 수행하고 결과를 Direct3D9 화면에 씁니다.
VideoProcessBltHD 하나 이상의 입력 샘플에서 비디오 처리 블릿을 수행하고 결과를 Microsoft Direct3D 화면에 씁니다.

VideoProcessorBlt 하나 이상의 입력 샘플에서 비디오 처리 작업을 수행하고 결과를 Direct3D 화면에 씁니다.
videoProcessorGetBehaviorHints ID3D11VideoContext::VideoProcessorBlt 메서드가 아닌 다중 평면 오버레이 하드웨어를 사용하여 가장 잘 수행되는 비디오 프로세서 작업을 나타내는 드라이버 힌트를 반환합니다.
videoProcessorGetOutputAlphaFillMode 비디오 프로세서의 현재 알파 채우기 모드를 가져옵니다.
VideoProcessorGetOutputBackgroundColor 비디오 프로세서의 현재 배경색을 가져옵니다.
VideoProcessorGetOutputColorSpace 비디오 프로세서의 현재 출력 색 공간을 가져옵니다.
VideoProcessorGetOutputColorSpace1 비디오 프로세서 출력 화면의 색 공간 정보를 가져옵니다.
VideoProcessorGetOutputConstriction 비디오 프로세서에서 수행하는 다운샘플링의 현재 수준을 가져옵니다.
VideoProcessorGetOutputExtension 비디오 프로세서에서 프라이빗 상태 데이터를 가져옵니다.
VideoProcessorGetOutputHDRMetaData 콘텐츠가 표시될 디스플레이를 설명하는 HDR 메타데이터를 가져옵니다.
VideoProcessorGetOutputShaderUsage Direct3D 셰이더에서 ID3D11VideoContext::VideoProcessorBlt 호출의 출력 화면을 읽을 수 있는지 여부를 나타내는 값을 가져옵니다.
VideoProcessorGetOutputStereoMode 비디오 프로세서가 스테레오 비디오 프레임을 생성하는지 여부를 쿼리합니다.
VideoProcessorGetOutputTargetRect 비디오 프로세서의 현재 대상 사각형을 가져옵니다.
VideoProcessorGetStreamAlpha 비디오 프로세서의 입력 스트림에 대한 평면 알파를 가져옵니다.
VideoProcessorGetStreamAutoProcessingMode

비디오 프로세서의 자동 처리 기능을 사용할 수 있는지 여부를 쿼리합니다.
VideoProcessorGetStreamColorSpace 비디오 프로세서의 입력 스트림에 대한 색 공간을 가져옵니다.
VideoProcessorGetStreamColorSpace1 비디오 프로세서 입력 스트림의 색 공간 정보를 가져옵니다.
VideoProcessorGetStreamDestRect 비디오 프로세서의 입력 스트림에 대한 대상 사각형을 가져옵니다.
VideoProcessorGetStreamExtension 비디오 처리 스트림에 대한 드라이버별 상태를 가져옵니다.
VideoProcessorGetStreamFilter 비디오 프로세서의 입력 스트림에 대한 이미지 필터 설정을 가져옵니다.
VideoProcessorGetStreamFrameFormat 비디오 프로세서에서 입력 스트림의 형식을 가져옵니다.
VideoProcessorGetStreamHDRMetaData 비디오 스트림과 연결된 HDR 메타데이터를 가져옵니다.
VideoProcessorGetStreamLumaKey 비디오 프로세서의 입력 스트림에 대한 루마 키를 가져옵니다.
VideoProcessorGetStreamMirror 비디오 프로세서 입력 스트림이 세로 또는 가로로 대칭 이동 중인지 여부를 나타내는 값을 가져옵니다.
VideoProcessorGetStreamOutputRate 비디오 프로세서가 입력 스트림에 대한 출력 프레임을 생성하는 속도를 가져옵니다.
VideoProcessorGetStreamPalette 비디오 프로세서의 입력 스트림에 대한 색상표 항목을 가져옵니다.
videoProcessorGetStreamPixelAspectRatio 비디오 프로세서의 입력 스트림에 대한 픽셀 가로 세로 비율을 가져옵니다.
VideoProcessorGetStreamRotation 비디오 프로세서의 입력 스트림에 대한 스트림 회전을 가져옵니다.
VideoProcessorGetStreamSourceRect 비디오 프로세서의 입력 스트림에 대한 원본 사각형을 가져옵니다.

VideoProcessorGetStreamStereoFormat 비디오 프로세서의 입력 스트림에 대한 스테레오 3D 형식을 가져옵니다.
VideoProcessorSetOutputAlphaFillMode 비디오 프로세서가 렌더링 대상에 쓰는 데이터의 알파 채우기 모드를 설정합니다.
VideoProcessorSetOutputBackgroundColor 비디오 프로세서의 배경색을 설정합니다.
VideoProcessorSetOutputColorSpace 비디오 프로세서의 출력 색 공간을 설정합니다.
videoProcessorSetOutputColorSpace1 비디오 프로세서 출력 화면의 색 공간 정보를 설정합니다.
VideoProcessorSetOutputConstriction 출력에서 수행할 다운샘플링의 양을 설정합니다.
VideoProcessorSetOutputExtension 드라이버별 비디오 처리 상태를 설정합니다.
videoProcessorSetOutputHDRMetaData 콘텐츠가 표시될 디스플레이를 설명하는 HDR 메타데이터를 설정합니다.
VideoProcessorSetOutputShaderUsage ID3D11VideoContext::VideoProcessorBlt 호출의 출력 화면을 Direct3D 셰이더에서 읽을지 여부를 나타내는 값을 설정합니다.
videoProcessorSetOutputStereoMode 비디오 프로세서에서 스테레오 비디오 프레임을 생성하는지 여부를 지정합니다.
videoProcessorSetOutputTargetRect 비디오 프로세서의 대상 사각형을 설정합니다.
VideoProcessorSetStreamAlpha 비디오 프로세서의 입력 스트림에 대한 평면 알파를 설정합니다.
VideoProcessorSetStreamAutoProcessingMode 비디오 프로세서에서 자동 처리 기능을 사용하거나 사용하지 않도록 설정합니다.
VideoProcessorSetStreamColorSpace 비디오 프로세서에서 입력 스트림의 색 공간을 설정합니다.

VideoProcessorSetStreamColorSpace1 비디오 프로세서 입력 스트림의 색 공간 정보를 설정합니다.
VideoProcessorSetStreamDestRect 비디오 프로세서의 입력 스트림에 대한 대상 사각형을 설정합니다.
VideoProcessorSetStreamExtension 비디오 처리 스트림에서 드라이버별 상태를 설정합니다.
VideoProcessorSetStreamFilter 비디오 프로세서에서 입력 스트림에 대한 이미지 필터를 사용하거나 사용하지 않도록 설정합니다.
VideoProcessorSetStreamFrameFormat 비디오 프로세서의 입력 스트림에 인터레이스 또는 프로그레시브 프레임이 포함되어 있는지 여부를 지정합니다.
VideoProcessorSetStreamHDRMetaData 비디오 스트림과 연결된 HDR 메타데이터를 설정합니다.
VideoProcessorSetStreamLumaKey 비디오 프로세서의 입력 스트림에 대한 루마 키를 설정합니다.
VideoProcessorSetStreamMirror 비디오 프로세서 입력 스트림을 세로 또는 가로로 대칭 이동해야 하는지 여부를 지정합니다.
VideoProcessorSetStreamOutputRate 비디오 프로세서가 입력 스트림에 대한 출력 프레임을 생성하는 속도를 설정합니다.
VideoProcessorSetStreamPalette 비디오 프로세서의 입력 스트림에 대한 색상표 항목을 설정합니다.
VideoProcessorSetStreamPixelAspectRatio 비디오 프로세서의 입력 스트림에 대한 픽셀 가로 세로 비율을 설정합니다.
VideoProcessorSetStreamRotation 비디오 프로세서의 입력 스트림에 대한 스트림 회전을 설정합니다.
VideoProcessorSetStreamSourceRect 비디오 프로세서의 입력 스트림에 대한 원본 사각형을 설정합니다.
VideoProcessorSetStreamStereoFormat 비디오 프로세서의 입력 스트림에 스테레오 3D 비디오를 사용하거나 사용하지 않도록 설정합니다.

WaitOnSignal 시간 제한 지정 시간 제한 간격이 경과하거나 동기화 개체가 신호를 보낼 때까지 차단합니다.
쓰기 스트림에 데이터를 씁니다.
WriteBufferImmediate 명령 스트림에서 직접 지정된 버퍼 위치에 32비트 직접 값을 씁니다. (ID3D12VideoDecodeCommandList::WriteBufferImmediate)
WriteBufferImmediate 명령 스트림에서 직접 지정된 버퍼 위치에 32비트 직접 값을 씁니다. (ID3D12VideoEncodeCommandList::WriteBufferImmediate)
WriteBufferImmediate 명령 스트림에서 직접 지정된 버퍼 위치에 32비트 직접 값을 씁니다. (ID3D12VideoProcessCommandList::WriteBufferImmediate)
writeSample 싱크 작성기에 샘플을 제공합니다.

인터페이스

테이블 확장

IAdvancedMediaCapture 고급 미디어 캡처를 사용하도록 설정합니다.
IAdvancedMediaCaptureInitializationSettings 고급 미디어 캡처에 대한 초기화 설정을 제공합니다.
IAdvancedMediaCaptureSettings 고급 미디어 캡처에 대한 설정을 제공합니다.
ICodecAPI ICodecAPI 인터페이스는 인코더 또는 디코더 필터에서 설정을 설정하고 검색하며 코덱에서 속성을 설정하기 위한 제네릭 메커니즘을 정의합니다.
ID3D11AuthenticatedChannel 그래픽 드라이버 또는 Microsoft Direct3D 런타임과 통신 채널을 제공합니다.

ID3D11CryptoSession 암호화 세션을 나타냅니다.
ID3D11VideoContext Microsoft Direct3D 11 디바이스의 비디오 기능을 제공합니다. (ID3D11VideoContext)
ID3D11VideoContext1 Microsoft Direct3D 11 디바이스의 비디오 기능을 제공합니다. (ID3D11VideoContext1)
ID3D11VideoContext2 Microsoft Direct3D 11 디바이스의 비디오 기능을 제공합니다. (ID3D11VideoContext2)
ID3D11VideoDecoder Microsoft Direct3D 11용 하드웨어 가속 비디오 디코더를 나타냅니다.
ID3D11VideoDecoderOutputView 비디오 디코딩 중에 액세스할 수 있는 출력 화면을 식별합니다.
ID3D11VideoDevice Microsoft Direct3D 11 디바이스의 비디오 디코딩 및 비디오 처리 기능을 제공합니다. (ID3D11VideoDevice)
ID3D11VideoDevice1 Microsoft Direct3D 11 디바이스의 비디오 디코딩 및 비디오 처리 기능을 제공합니다. (ID3D11VideoDevice1)
ID3D11VideoProcessor Microsoft Direct3D 11의 비디오 프로세서를 나타냅니다.
ID3D11VideoProcessorEnumerator Microsoft Direct3D 11 디바이스의 비디오 프로세서 기능을 열거합니다. (ID3D11VideoProcessorEnumerator)
ID3D11VideoProcessorEnumerator1 Microsoft Direct3D 11 디바이스의 비디오 프로세서 기능을 열거합니다. (ID3D11VideoProcessorEnumerator1)
ID3D11VideoProcessorInputView 비디오 처리 중에 액세스할 수 있는 입력 화면을 식별합니다.
ID3D11VideoProcessorOutputView 비디오 처리 중에 액세스할 수 있는 출력 화면을 식별합니다.
ID3D12VideoDecodeCommandList 비디오 디코딩을 위한 그래픽 명령 목록을 캡슐화합니다.

ID3D12VideoDecodeCommandList1 비디오 디코딩을 위한 그래픽 명령 목록을 캡슐화합니다. 이 인터페이스는 ID3D12VideoDecodeCommandList에서 상속되며 비디오 디코딩 히스토그램에 대한 지원을 추가합니다.
ID3D12VideoDecodeCommandList2 비디오 확장 명령에 대한 지원을 추가합니다. (ID3D12VideoDecodeCommandList2)
ID3D12VideoDecodeCommandList3 비디오 디코딩을 위한 그래픽 명령 목록을 캡슐화합니다.
id3D12VideoDecoder Direct3D 12 비디오 디코더를 나타냅니다.
ID3D12VideoDecoder1 디코딩 작업을 수행하기 위한 해상도 독립적 리소스 및 상태를 포함하는 Direct3D 12 비디오 디코더를 나타냅니다. 보호된 리소스에 대한 지원을 추가합니다.
ID3D12VideoDecoderHeap Direct3D 12 비디오 디코더 힙을 나타냅니다.
ID3D12VideoDevice 비디오 기능을 쿼리하고 비디오 디코더 및 프로세서를 인스턴스화하는 기능을 포함하여 Microsoft Direct3D 12 디바이스의 비디오 디코딩 및 처리 기능을 제공합니다.
ID3D12VideoDevice1 동작 예측에 대한 지원을 추가합니다.
ID3D12VideoDevice2 보호된 리소스에 대한 지원을 추가합니다.
ID3D12VideoDevice3 ID3D12VideoDevice 인터페이스를 확장하여 지원 비디오 인코딩 기능을 추가합니다.
ID3D12VideoEncodeCommandList 동작 추정을 포함하여 비디오 인코딩을 위한 그래픽 명령 목록을 캡슐화합니다.
ID3D12VideoEncodeCommandList1 이 인터페이스는 ID3D12VideoEncodeCommandList에서 상속되며 비디오 확장 명령에 대한 지원을 추가합니다.
ID3D12VideoEncodeCommandList2 이 인터페이스는 ID3D12VideoEncodeCommandList1에서 상속되며 비디오를 인코딩하고 인코딩 작업 메타데이터를 확인하는 메서드를 추가합니다.

ID3D12VideoEncodeCommandList3 비디오 인코딩을 위한 그래픽 명령 목록을 캡슐화합니다.
id3D12VideoEncoder Direct3D 12 비디오 인코더를 나타냅니다.
ID3D12VideoEncoderHeap Direct3D 12 비디오 인코더 힙을 나타냅니다.
ID3D12VideoExtensionCommand 비디오 확장 명령을 나타내는 참조 개수 개체입니다.
ID3D12VideoMotionEstimator 이 인터페이스는 비디오 동작 추정 작업에 대한 컨텍스트를 유지 관리합니다.
ID3D12VideoMotionVectorHeap 예상 동작 벡터가 저장되는 힙을 나타냅니다.
ID3D12VideoProcessCommandList 비디오 처리를 위한 그래픽 명령 목록을 캡슐화합니다. (ID3D12VideoProcessCommandList)
ID3D12VideoProcessCommandList1 비디오 처리를 위한 그래픽 명령 목록을 캡슐화합니다. (ID3D12VideoProcessCommandList1)
ID3D12VideoProcessCommandList2 비디오 확장 명령에 대한 지원을 추가합니다. (ID3D12VideoProcessCommandList2)
ID3D12VideoProcessCommandList3 비디오 처리를 위한 그래픽 명령 목록을 캡슐화합니다.
id3D12VideoProcessor 비디오 프로세서를 만든 ID3D12VideoDevice::CreateVideoProcessor 호출에 대한 매개 변수에 대한 정보를 가져오는 메서드를 제공합니다.
ID3D12VideoProcessor1 ID3D12VideoProcessor에서 상속되며 보호된 리소스에 대한 지원을 제공합니다.
IDirect3D9ExOverlayExtension Direct3D 디바이스의 오버레이 하드웨어 기능을 쿼리합니다. (IDirect3D9ExOverlayExtension)
IDirect3DAuthenticatedChannel9

그래픽 드라이버 또는 Direct3D 런타임과 통신 채널을 제공합니다. 이 인터페이스에 대한 포인터를 가져오려면 IDirect3DDevice9Video::CreateAuthenticatedChannel을 호출합니다.
IDirect3DCryptoSession9 암호화 세션을 나타냅니다. 이 인터페이스에 대한 포인터를 얻으려면 IDirect3DDevice9Video::CreateCryptoSession을 호출합니다.
IDirect3DDevice9Video 애플리케이션이 그래픽 드라이버에서 구현한 콘텐츠 보호 및 암호화 서비스를 사용할 수 있도록 합니다. 이 인터페이스에 대한 포인터를 얻으려면 D3D9Ex 디바이스에서 QueryInterface를 호출합니다.
IDirect3DDeviceManager9 두 스레드가 동일한 Direct3D 9 디바이스를 공유할 수 있도록 하고 디바이스의 DXVA(DirectX Video Acceleration) 기능에 대한 액세스를 제공합니다.
IDirectXVideoAccelerationService Direct3D 디바이스에서 DXVA(DirectX Video Acceleration) 서비스를 제공합니다.
IDirectXVideoDecoder DXVA(DirectX Video Acceleration) 비디오 디코더 디바이스를 나타냅니다.
IDirectXVideoDecoderService DXVA(DirectX Video Acceleration) 디코더 서비스에 대한 액세스를 제공합니다.
IDirectXVideoMemoryConfiguration 압축되지 않은 비디오 화면에 대한 비디오 메모리 유형을 설정합니다.
IDirectXVideoProcessor DXVA(DirectX Video Acceleration) 비디오 프로세서 디바이스를 나타냅니다.
IDirectXVideoProcessorService DXVA(DirectX Video Acceleration) 비디오 처리 서비스에 대한 액세스를 제공합니다.
IDXVAHD_Device Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스를 나타냅니다.
IDXVAHD_VideoProcessor Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서를 나타냅니다.
IMF2DBuffer 비디오 프레임과 같은 2차원 표면을 포함하는 버퍼를 나타냅니다. (IMF2DBuffer)
IMF2DBuffer2 비디오 프레임과 같은 2차원 표면을 포함하는 버퍼를 나타냅니다. (IMF2DBuffer2)

IMFActivate 애플리케이션에서 개체 만들기를 연기할 수 있도록 합니다.
IMFASFContentInfo ASF(Advanced Systems Format) 사양을 준수하는 파일의 헤더 섹션을 사용할 메서드를 제공합니다.
IMFASFIndexer ASF(시스템 형식) 파일에서 인덱스를 사용할 수 있는 메서드를 제공합니다.
IMFASFMultiplexer ASF(Advanced Systems Format) 데이터 패킷을 만드는 메서드를 제공합니다.
IMFASFMutualExclusion ASF 프로필에서 상호 배타적인 스트림 그룹에 대한 정보를 관리하는 ASF(Advanced Systems Format) 상호 제외 개체를 구성합니다.
IMFASFProfile ASF(고급 시스템 형식) 프로필을 관리합니다.
IMFASFSplitter ASF(Advanced Systems Format) 파일에서 데이터를 읽는 메서드를 제공합니다.
IMFASFStreamConfig ASF 파일에서 스트림의 설정을 구성합니다.
IMFASFStreamPrioritization 구현되지 않았습니다. (IMFASFStreamPrioritization)
IMFASFStreamSelector ASF 헤더의 상호 제외 정보에 따라 ASF(Advanced Systems Format) 파일에서 스트림을 선택합니다.
IMFAsyncCallback 비동기 메서드가 완료되면 애플리케이션에 알리는 콜백 인터페이스입니다. (IMFAsyncCallback)
IMFAsyncCallbackLogging 비동기 콜백이 연결된 부모 개체에 대한 로깅 정보를 제공합니다.
IMFAsyncResult 비동기 작업의 결과에 대한 정보를 제공합니다. (IMFAsyncResult)
IMFAttributes 개체에 키/값 쌍을 저장하는 일반적인 방법을 제공합니다.

IMFAudioMediaType IMFAudioMediaType은 Windows 7을 기준으로 더 이상 사용할 수 없습니다.
IMFAudioPolicy SAR(스트리밍 오디오 렌더러)과 연결된 오디오 세션을 구성합니다.
IMFAudioStreamVolume 개별 오디오 채널의 볼륨 수준을 제어합니다.
IMFBufferListNotify IMFSourceBufferList 개체가 클라이언트에 중요한 상태 변경을 알릴 수 있도록 합니다.
IMFByteStream 로컬 파일, 네트워크 파일 또는 다른 원본일 수 있는 일부 데이터 원본의 바이트 스트림을 나타냅니다.
IMFByteStreamBuffering 바이트 스트림이 네트워크에서 데이터를 버퍼링하는 방법을 제어합니다.
IMFByteStreamCacheControl 네트워크 바이트 스트림이 로컬 캐시로 데이터를 전송하는 방법을 제어합니다. (IMFByteStreamCacheControl)
IMFByteStreamCacheControl2 네트워크 바이트 스트림이 로컬 캐시로 데이터를 전송하는 방법을 제어합니다. (IMFByteStreamCacheControl2)
IMFByteStreamHandler 바이트 스트림에서 미디어 원본을 만듭니다.
IMFByteStreamProxyClassFactory 바이트 스트림에 대한 프록시를 만듭니다. (IMFByteStreamProxyClassFactory)
IMFByteStreamTimeSeek 시간별 바이트 스트림을 찾습니다.
IMFCameraConfigurationManager IMFCameraConfigurationManager 인터페이스는 COM 함수 CoCreateInstance를 호출하고 CLSID_CAMERACONFIGURATIONMANAGER CLSID 매개 변수로 전달하여 만들 수 있습니다.
IMFCameraControlDefaults 카메라 컨트롤 기본값을 지정하는 메서드를 제공합니다.
IMFCameraControlDefaultsCollection 카메라 컨트롤의 컬렉션을 나타냅니다.

IMFCameraControlMonitor 카메라 컨트롤의 상태가 변경되면 알림을 구독 및 구독 취소하는 데 사용되는 카메라 컨트롤 모니터를 나타냅니다.
IMFCameraControlNotify 카메라 컨트롤 변경에 대한 알림 콜백을 나타냅니다.
IMFCameraOcclusionStateMonitor 카메라 디바이스의 폐색 상태를 모니터링합니다.
IMFCameraOcclusionStateReport IMFCameraOcclusionStateMonitor에서 감지한 상태 변경과 관련된 카메라 폐색 상태를 제공합니다.
IMFCameraOcclusionStateReportCallback 카메라 폐색 상태 변경 알림을 수신하기 위해 애플리케이션에서 구현합니다.
IMFCameraSyncObject 가상 카메라를 만들고 관리하는 앱과 가상 카메라 원본 간의 동기화 메커니즘을 제공합니다.
IMFCaptureEngine 하나 이상의 캡처 디바이스를 제어합니다.
IMFCaptureEngineClassFactory 캡처 엔진의 인스턴스를 만듭니다. (IMFCaptureEngineClassFactory)
IMFCaptureEngineOnEventCallback 캡처 엔진에서 이벤트를 수신하기 위한 콜백 인터페이스입니다.
IMFCaptureEngineOnSampleCallback 캡처 엔진에서 데이터를 수신하는 콜백 인터페이스입니다.
IMFCaptureEngineOnSampleCallback2 캡처 엔진에서 데이터를 수신하는 데 사용되는 IMFCaptureEngineOnSampleCallback 콜백 인터페이스에 대한 확장입니다.
IMFCapturePhotoSink 사진 싱크를 제어합니다.
IMFCapturePreviewSink 미리 보기 싱크를 제어합니다.
IMFCaptureRecordSink 기록 싱크를 제어합니다.

IMFCaptureSink 캡처 디바이스에서 하나 이상의 스트림을 수신하는 개체인 캡처 싱크를 제어합니다.
IMFCaptureSink2 IMFCaptureSink 인터페이스를 확장하여 레코드 싱크 또는 미리 보기 싱크의 출력 미디어 형식을 동적으로 설정하는 기능을 제공합니다.
IMFCaptureSource 캡처 원본 개체를 제어합니다. 캡처 원본은 오디오 및 비디오 캡처 디바이스를 관리합니다.
IMFCdmSuspendNotify 일시 중단 전에 전역 리소스를 일관된 상태로 전환해야 하는 경우 클라이언트가 CDM(콘텐츠 암호 해독 모듈)에 알릴 수 있도록 하는 데 사용됩니다.
IMFClock Microsoft Media Foundation의 시계에서 타이밍 정보를 제공합니다.
IMFClockConsumer IMFPresentationClock에 액세스하기 위해 앱에서 구현합니다.
IMFClockStateSink 프레젠테이션 시계에서 상태 변경 알림을 받습니다.
IMFCollection IUnknown 포인터의 제네릭 컬렉션을 나타냅니다.
IMFContentDecryptionModule DRM 키 시스템의 CDM(콘텐츠 암호 해독 모듈)을 나타냅니다.
IMFContentDecryptionModuleAccess 미디어 키 시스템에 대한 액세스를 제공합니다.
IMFContentDecryptionModuleFactory IMFContentDecryptionModuleAccess 개체를 만들기 위한 팩터리 인터페이스입니다.
IMFContentDecryptionModuleSession 암호화된 미디어 확장 지원을 위해 CDM(콘텐츠 암호 해독 모듈)에 대한 액세스를 제공합니다.
IMFContentDecryptionModuleSessionCallbacks IMFContentDecryptionModuleSession에서 키 상태 변경 업데이트를 수신하기 위한 콜백 메커니즘을 제공합니다.
IMFContentDecryptorContext

암호 해독기가 하드웨어 키를 관리하고 하드웨어 샘플의 암호를 해독할 수 있습니다.
IMFContentEnabler 사용자가 미디어 콘텐츠에 액세스하기 위해 수행해야 하는 한 단계를 구현합니다.
IMFContentProtectionDevice 암호 해독기가 보호 시스템에 대한 하드웨어 암호 해독을 구현하는 보안 프로세서와 통신할 수 있도록 허용합니다.
IMFContentProtectionManager 애플리케이션에 콘텐츠 사용 프로그램 개체에 대한 포인터를 제공하여 보호된 콘텐츠를 재생할 수 있습니다.
IMFD3D12SynchronizationObject D3D12 리소스 할당자가 D3D12 리소스의 수명을 관리할 수 있도록 허용합니다.
IMFD3D12SynchronizationObjectCommands 리소스를 사용할 준비가 되거나 해제될 수 있을 때 D3D12 리소스의 생산자 또는 소비자가 다운스트림 구성 요소에 신호를 보낼 수 있도록 하는 명령을 제공합니다.
IMFDLNASinkInit DLNA(Digital Living Network Alliance) 미디어 싱크를 초기화합니다. (IMFDLNASinkInit)
IMFDRMNetHelper 네트워크 싱크에서 네트워크 디바이스에 대한 Windows Media DRM(디지털 권한 관리)을 구성합니다.
IMFDXGIBuffer Microsoft DXGI(DirectX Graphics Infrastructure)가 포함된 버퍼를 나타냅니다.
IMFDXGIDeviceManager 두 스레드가 동일한 Microsoft Direct3D 11 디바이스를 공유할 수 있도록 합니다.
IMFDXGIDeviceManagerSource Microsoft Media Foundation 비디오 렌더링 싱크에서 IMFDXGIDeviceManager를 가져오는 기능을 제공합니다.
IMFExtendedCameraControl 이 인터페이스는 캡처 디바이스의 확장 속성을 구성하는 데 사용됩니다.
IMFExtendedCameraController IMFExtendedCameraController 인터페이스를 사용하면 앱이 캡처 디바이스의 확장 속성을 구성하는 데 사용되는 IMFExtendedCameraControl 인스턴스를 검색할 수 있습니다.
IMFExtendedDRMTypeSupport 지정된 키 시스템에 대해 지정된 콘텐츠 형식이 지원되는지 여부를 쿼리하는 메서드를 제공합니다.

IMFFieldOfUseMFTUnlock 애플리케이션에서 사용 제한이 있는 MFT(Media Foundation 변환)를 사용할 수 있도록 합니다.
IMFFinalizableMediaSink 필요에 따라 종료하기 전에 필요한 작업을 수행하기 위해 미디어 싱크에서 지원됩니다.
IMFGetService 지정된 서비스 인터페이스에 대한 개체를 쿼리합니다. (IMFGetService)
IMFHttpDownloadRequest 애플리케이션은 Microsoft Media Foundation에서 사용하는 HTTP 및 HTTPS 프로토콜의 기본 구현을 재정의하기 위해 이 인터페이스를 구현합니다. (IMFHttpDownloadRequest)
IMFHttpDownloadSession 애플리케이션은 Microsoft Media Foundation에서 사용하는 HTTP 및 HTTPS 프로토콜의 기본 구현을 재정의하기 위해 이 인터페이스를 구현합니다. (IMFHttpDownloadSession)
IMFHttpDownloadSessionProvider 애플리케이션은 사용자 지정 HTTP 또는 HTTPS 다운로드 구현을 사용자 지정하기 위해 이 인터페이스를 구현합니다.
IMFImageSharingEngine 이미지 공유를 사용하도록 설정합니다.
IMFImageSharingEngineClassFactory IMFImageSharingEngine의 인스턴스를 만듭니다.
IMFInputTrustAuthority PMP(보호된 미디어 경로)의 다른 구성 요소가 ITA(입력 신뢰 기관)에서 제공하는 입력 보호 시스템을 사용할 수 있도록 합니다.
IMFLocalMFTRegistration 호출자의 프로세스에서 MFT(Media Foundation 변환)를 등록합니다.
IMFMediaBuffer 미디어 데이터가 포함된 메모리 블록을 나타냅니다.
IMFMediaEngine 애플리케이션에서 오디오 또는 비디오 파일을 재생할 수 있도록 합니다.
IMFMediaEngineClassFactory 미디어 엔진의 인스턴스를 만듭니다.

IMFMediaEngineClassFactory2 IMFMediaKeys 개체의 인스턴스를 만듭니다.
IMFMediaEngineClassFactory4 지정된 키 시스템에 대한 CDM(Content Decryption Module) 개체의 클래스 팩터리인 IMFContentDecryptionModuleFactory 인스턴스를 만들기 위한 CreateContentDecryptionModuleFactory 메서드를 제공합니다.
IMFMediaEngineClassFactoryEx IMFMediaEngineClassFactory 인터페이스에 대한 확장입니다.
IMFMediaEngineEME 암호화된 미디어 확장 메서드를 추가하기 위해 미디어 엔진에 의해 구현됩니다.
IMFMediaEngineEx IMFMediaEngine 인터페이스를 확장합니다.
IMFMediaEngineExtension 애플리케이션이 미디어 엔진에서 미디어 리소스를 로드할 수 있도록 합니다.
IMFMediaEngineNeedKeyNotify 키 요청 데이터를 알리기 위해 미디어 엔진에 대한 콜백을 나타냅니다.
IMFMediaEngineNotify IMFMediaEngine 인터페이스에 대한 콜백 인터페이스입니다.
IMFMediaEngineOPMInfo OPM(출력 보호 관리자)에 대한 정보를 가져오는 메서드를 제공합니다.
IMFMediaEngineProtectedContent 미디어 엔진이 보호된 비디오 콘텐츠를 재생할 수 있도록 합니다.
IMFMediaEngineSrcElements 미디어 엔진에 미디어 리소스 목록을 제공합니다.
IMFMediaEngineSrcElementsEx IMFMediaEngineSrcElements 인터페이스를 확장하여 추가 기능을 제공합니다.
IMFMediaEngineSupportsSourceTransfer 미디어 엔진과 Play To의 공유 엔진 간에 미디어 원본을 전송할 수 있습니다.
IMFMediaEngineWebSupport

웹 오디오 재생을 사용하도록 설정합니다.
IMFMediaError 미디어 엔진에 대한 현재 오류 상태를 제공합니다.
IMFMediaEvent Media Foundation 개체에 의해 생성된 이벤트를 나타냅니다. 이 인터페이스를 사용하여 이벤트에 대한 정보를 가져옵니다.
IMFMediaEventGenerator 이벤트를 생성하는 Media Foundation 개체에서 이벤트를 검색합니다.
IMFMediaEventQueue IMFMediaEventGenerator 인터페이스를 구현해야 하는 애플리케이션에 대한 이벤트 큐를 제공합니다.
IMFMediaKeys DRM(디지털 권한 관리) 키 시스템을 사용하여 미디어 데이터의 암호를 해독하는 데 사용되는 미디어 키를 나타냅니다.
IMFMediaKeySession DRM(디지털 권한 관리) 키 시스템을 사용하는 세션을 나타냅니다.
IMFMediaKeySessionNotify 미디어 키 세션에 대한 정보를 앱에 알리는 메커니즘을 제공합니다.
IMFMediaSession 보호된 콘텐츠와 보호되지 않은 콘텐츠에 대한 재생 컨트롤을 제공합니다.
IMFMediaSharingEngine 미디어 공유를 사용하도록 설정합니다.
IMFMediaSharingEngineClassFactory IMFMediaSharingEngine의 인스턴스를 만듭니다. (IMFMediaSharingEngineClassFactory)
IMFMediaSink 미디어 싱크 개체에 의해 구현됩니다.
IMFMediaSinkPreroll 프레젠테이션 클록이 시작되기 전에 미디어 싱크에서 샘플을 받을 수 있도록 합니다.
IMFMediaSource 미디어 원본 개체에 의해 구현됩니다.

IMFMediaSourceEx IMFMediaSource 인터페이스를 확장하여 미디어 원본에 대한 추가 기능을 제공합니다.
IMFMediaSourceExtension MSE(미디어 원본 확장)에 대한 기능을 제공합니다.
IMFMediaSourceExtensionNotify IMFMediaSourceExtension과 관련된 이벤트를 발생시키는 기능을 제공합니다.
IMFMediaSourcePresentationProvider 시퀀서 원본에 알림을 제공합니다.
IMFMediaSourceTopologyProvider 애플리케이션이 시퀀서 소스에서 토폴로지를 가져올 수 있도록 합니다.
IMFMediaStream 미디어 원본에서 하나의 스트림을 나타냅니다.
IMFMediaStreamSourceSampleRequest MediaStreamSource의 샘플 요청을 나타냅니다.
IMFMediaTimeRange 각 범위가 시작 및 종료 시간으로 정의되는 시간 범위 목록을 나타냅니다.
IMFMediaType 미디어 형식에 대한 설명을 나타냅니다.
IMFMediaTypeHandler 미디어 원본 또는 미디어 싱크와 같은 개체의 미디어 형식을 가져오고 설정합니다.
IMFMetadata 개체에 대한 메타데이터를 관리합니다.
IMFMetadataProvider 미디어 원본 또는 다른 개체에서 메타데이터를 가져옵니다.
IMFMuxStreamAttributesManager 멀티플렉싱된 미디어 소스의 하위 스트림에 대한 IMFAttributes에 대한 액세스를 제공합니다.
IMFMuxStreamMediaTypeManager 멀티플렉싱된 미디어 원본에 대한 스트림 구성 관리를 사용하도록 설정합니다. 스트림 구성은 멀티플렉싱된 출력을 포함할 수 있는 하위 스트림 집합을 정의합니다.

IMFMuxStreamSampleManager 멀티플렉싱된 미디어 소스의 출력 내에서 개별 하위 스트림에 대한 IMFSample 개체를 검색하는 기능을 제공합니다.
IMFNetCredential 인증을 위해 사용자 이름 및 암호 정보를 설정하고 검색합니다.
IMFNetCredentialCache 자격 증명 캐시에서 자격 증명을 가져옵니다.
IMFNetCredentialManager 네트워크 원본에 대한 사용자 자격 증명을 제공하기 위해 애플리케이션에서 구현합니다.
IMFNetCrossOriginSupport HTML5 미디어 다운로드에 대한 원본 간 정책을 적용하려는 클라이언트에 의해 구현됩니다.
IMFNetProxyLocator 서버에 연결할 때 사용할 프록시를 결정합니다.
IMFNetProxyLocatorFactory 사용할 프록시를 결정하는 프록시 로케이터 개체를 만듭니다.
IMFNetResourceFilter 바이트 스트림이 URL을 요청할 때 애플리케이션에 알리고 애플리케이션이 URL 리디렉션을 차단할 수 있도록 합니다.
IMFNetSchemeHandlerConfig 네트워크 구성표 플러그 인을 구성합니다.
IMFObjectReferenceStream 스트림에 대한 인터페이스 포인터를 마샬링합니다. IStream을 지원하는 스트림 개체는 인터페이스 포인터에 대한 사용자 지정 마샬링을 제공하기 위해 이 인터페이스를 노출할 수 있습니다.
IMFOutputPolicy ITA(입력 신뢰 기관)에서 사용 정책을 캡슐화합니다.
IMFOutputSchema 출력 보호 시스템 및 해당 구성 데이터에 대한 정보를 캡슐화합니다.
IMFOutputTrustAuthority 신뢰할 수 있는 출력이 지원하는 하나 이상의 출력 보호 시스템의 기능을 캡슐화합니다.
IMFPluginControl

Microsoft Media Foundation에서 미디어 원본 및 변환을 열거하는 방법을 제어합니다. (IMFPluginControl)
IMFPluginControl2 Microsoft Media Foundation에서 미디어 원본 및 변환을 열거하는 방법을 제어합니다. (IMFPluginControl2)
IMFPMPClient 미디어 원본이 IMFPMPHost 인터페이스에 대한 포인터를 받을 수 있도록 합니다.
IMFPMPClientApp 미디어 원본이 Windows 스토어 앱에서 콘텐츠 보호 기능을 구현하는 메커니즘을 제공합니다.
IMFPMPHost 애플리케이션 프로세스의 미디어 원본이 PMP(보호된 미디어 경로) 프로세스에서 개체를 만들 수 있도록 합니다.
IMFPMPHostApp 미디어 원본이 PMP(보호된 미디어 경로) 프로세스에서 Windows 런타임 개체를 만들 수 있도록 허용합니다.
IMFPMPServer 미디어 세션의 두 인스턴스가 동일한 PMP(보호된 미디어 경로) 프로세스를 공유할 수 있도록 합니다.
IMFPresentationClock 샘플이 렌더링되는 시기를 예약하고 여러 스트림을 동기화하는 데 사용되는 프레젠테이션 시계를 나타냅니다.
IMFPresentationDescriptor 프레젠테이션의 세부 정보를 설명합니다. 프레젠테이션은 일반적인 프레젠테이션 시간을 공유하는 관련 미디어 스트림 집합입니다.
IMFPresentationTimeSource 프레젠테이션 시계의 클록 시간을 제공합니다.
IMFProtectedEnvironmentAccess 콘텐츠 보호 시스템이 보호된 환경과 핸드셰이크를 수행할 수 있도록 하는 메서드를 제공합니다. CreateFile 및 DeviceIoControl API는 Windows 스토어 앱에서 사용할 수 없기 때문에 필요합니다.
IMFQualityAdvise 품질 관리자가 파이프라인에서 구성 요소의 오디오 또는 비디오 품질을 조정할 수 있도록 합니다.
IMFQualityAdvise2 파이프라인 개체가 품질 메시지에 대한 응답으로 자체 오디오 또는 비디오 품질을 조정할 수 있도록 합니다.
IMFQualityAdviseLimits 개체에서 지원하는 품질 모드 수를 쿼리합니다.

IMFQualityManager 재생 품질을 조정합니다. 이 인터페이스는 품질 관리자에 의해 노출됩니다.
IMFRateControl 재생 속도를 가져오거나 설정합니다.
IMFRateSupport 역방향 재생을 포함하여 지원되는 재생 속도의 범위를 쿼리합니다.
IMFReadWriteClassFactory 싱크 작성기 또는 원본 판독기의 인스턴스를 만듭니다.
IMFRealTimeClient MMCSS(멀티미디어 클래스 스케줄러 서비스)에 등록하도록 파이프라인 개체에 알깁니다. (IMFRealTimeClient)
IMFRealTimeClientEx MMCSS(멀티미디어 클래스 스케줄러 서비스)에 등록하도록 파이프라인 개체에 알깁니다. (IMFRealTimeClientEx)
IMFRelativePanelReport 이 인터페이스는 표시 영역에 대한 현재 상대 패널 위치를 검색하는 데 사용됩니다.
IMFRelativePanelWatcher 디스플레이와 연결된 패널을 모니터링하여 패널의 상대 위치가 변경될 때 앱이 알림을 받도록 합니다.
IMFRemoteAsyncCallback Microsoft Media Foundation 프록시/스텁 DLL에서 프로세스 경계를 넘어 특정 비동기 메서드 호출을 마샬링하는 데 사용됩니다. 애플리케이션은 이 인터페이스를 사용하거나 구현하지 않습니다.
IMFRemoteDesktopPlugin 터미널 서비스 환경에서 사용할 토폴로지 수정 (IMFRemoteDesktopPlugin)
IMFRemoteProxy 원격 개체의 프록시 역할을 하는 개체에 의해 노출됩니다.
IMFSAMIStyle SAMI 미디어 원본에서 동기화된 SAMI(액세스 가능 미디어 교환) 스타일을 설정하고 검색합니다.
IMFSample 미디어 데이터의 컨테이너 개체인 미디어 샘플을 나타냅니다.
IMFSampleAllocatorControl

비디오 캡처 원본 및 변환에 의해 구현됩니다. 시스템에서 컨테이너 내에서 액세스할 수 있는 메모리를 사용하여 샘플을 할당하는 샘플 할당자를 구성 요소에 제공할 수 있습니다.
IMFSampleGrabberSinkCallback 샘플 그래버 싱크에서 미디어 데이터를 가져오는 콜백 인터페이스입니다.
IMFSampleGrabberSinkCallback2 IMFSampleGrabberSinkCallback 인터페이스를 확장합니다.
IMFSampleOutputStream 미디어 샘플을 바이트 스트림에 씁니다.
IMFSampleProtection PMP(보호된 미디어 경로) 내의 미디어 데이터에 대한 암호화를 제공합니다.
IMFSaveJob 원본 바이트 스트림에서 애플리케이션 제공 바이트 스트림으로 미디어 데이터를 유지합니다.
IMFSchemeHandler URL에서 미디어 원본 또는 바이트 스트림을 만듭니다.
IMFSecureChannel 두 개체 간에 단방향 보안 채널을 설정합니다.
IMFSeekInfo 특정 검색 위치의 경우 가장 가까운 두 키 프레임을 가져옵니다. (IMFSeekInfo)
IMFSensorActivitiesReport 센서의 현재 활동을 설명하는 IMFSensorActivityReport 개체에 대한 액세스를 제공합니다.
IMFSensorActivitiesReportCallback 센서 활동 보고서를 사용할 수 있을 때 콜백을 수신하기 위해 클라이언트가 구현한 인터페이스입니다.
IMFSensorActivityMonitor 센서 활동 모니터를 제어하는 방법을 제공합니다.
IMFSensorActivityReport 센서의 활동 보고서를 나타냅니다.
IMFSensorDevice IMFSensorGroup 인터페이스로 표시되는 센서 그룹에 속할 수 있는 센서 디바이스를 나타냅니다. 이 컨텍스트의 "디바이스"라는 용어는 물리적 디바이스, 사용자 지정 미디어 원본 또는 프레임 공급자를 참조할 수 있습니다.

IMFSensorGroup IMFMediaSource를 만들 수 있는 센서 디바이스 그룹을 나타냅니다.
IMFSensorProcessActivity 센서와 연결된 프로세스의 활동을 나타냅니다.
IMFSensorProfile 미디어 파운데이션 센서 프로필을 설명합니다.
IMFSensorProfileCollection 미디어 파운데이션 센서 프로필 개체의 컬렉션을 포함합니다.
IMFSensorStream . (IMFSensorStream)
IMFSensorTransformFactory 센서 변환에 의해 구현된 인터페이스는 미디어 파이프라인이 센서 변환의 요구 사항을 쿼리하고 센서 변환의 런타임 인스턴스를 만들 수 있도록 합니다.
IMFSequencerSource Sequencer 원본에 의해 구현됩니다.
IMFSharingEngineClassFactory 미디어 공유 엔진의 인스턴스를 만듭니다. (IMFSharingEngineClassFactory)
IMFShutdown 명시적으로 종료해야 하는 일부 Media Foundation 개체에 의해 노출됩니다.
IMFSignedLibrary 콘텐츠 보호 시스템이 서명된 라이브러리에서 함수의 프로시저 주소를 가져올 수 있도록 하는 메서드를 제공합니다. 이 메서드는 Windows 스토어 앱에서 사용할 수 없는 GetProcAddress와 동일한 기능을 제공합니다.
IMFSimpleAudioVolume SAR(스트리밍 오디오 렌더러) 및 오디오 캡처 원본과 연결된 오디오 세션의 마스터 볼륨 수준을 제어합니다.
IMFSinkWriter Microsoft Media Foundation 싱크 기록기 개체에 의해 구현됩니다.
IMFSinkWriterCallback Microsoft Media Foundation 싱크 작성기에 대한 콜백 인터페이스입니다.
IMFSinkWriterCallback2

IMFSinkWriterCallback 인터페이스를 확장합니다.
IMFSinkWriterEncoderConfig 미디어 유형 및 인코더 구성을 동적으로 변경하기 위해 싱크 기록기에 추가 기능을 제공합니다.
IMFSinkWriterEx IMFSinkWriter 인터페이스를 확장합니다.
IMFSourceBuffer IMFMediaSourceExtension에 대한 미디어 데이터가 포함된 버퍼를 나타냅니다.
IMFSourceBufferList IMFSourceBuffer 개체의 컬렉션을 나타냅니다.
IMFSourceBufferNotify IMFSourceBuffer와 관련된 이벤트를 발생시키는 기능을 제공합니다.
IMFSourceOpenMonitor 비동기 열기 작업의 진행률에 대한 네트워크 원본으로부터 알림을 수신하는 콜백 인터페이스입니다.
IMFSourceReader Microsoft Media Foundation 원본 판독기 개체에 의해 구현됩니다.
IMFSourceReaderCallback Microsoft Media Foundation 원본 판독기용 콜백 인터페이스입니다.
IMFSourceReaderCallback2 IMFSourceReaderCallback 인터페이스를 확장합니다.
IMFSourceReaderEx IMFSourceReader 인터페이스를 확장합니다.
IMFSourceResolver URL 또는 바이트 스트림에서 미디어 원본을 만듭니다.
IMFSpatialAudioObjectBuffer 연결된 위치 및 렌더링 메타데이터가 있는 오디오 데이터의 섹션을 나타냅니다. 공간 오디오 개체는 IMFSpatialAudioSample 인스턴스에 저장되며 Media Foundation 구성 요소 간에 공간 오디오 정보를 전달할 수 있습니다.
IMFSpatialAudioSample 공간 사운드 정보가 있는 멀티미디어 샘플을 나타냅니다. 모든 IMFSpatialAudioSample에는 하나 이상의 IMFSpatialAudioObjectBuffer 개체가 포함됩니다.

IMFSSLCertificateManager 클라이언트에서 구현하고 Microsoft Media Foundation에서 호출하여 서버에서 요청한 클라이언트 SSL(Secure Sockets Layer) 인증서를 가져옵니다.
IMFStreamDescriptor 미디어 원본의 한 스트림에 대한 정보를 가져옵니다.
IMFStreamingSinkConfig 콘텐츠를 스트리밍하는 데 사용되는 미디어 싱크에 구성 정보를 전달합니다.
IMFStreamSink 미디어 싱크 개체의 스트림을 나타냅니다.
IMFSystemId 시스템 ID 데이터를 검색하는 메서드를 제공합니다.
IMFTimecodeTranslate 영화 협회와 텔레비전 엔지니어(SMPTE) 시간 코드와 100나노초 시간 단위를 변환합니다.
IMFTimedText 시간 제한 텍스트 개체는 시간 제한 텍스트의 구성 요소를 나타냅니다.
IMFTimedTextBinary 시간 제한 텍스트 개체의 데이터 콘텐츠를 나타냅니다.
IMFTimedTextCue 시간 제한 텍스트 큐 개체를 나타냅니다.
IMFTimedTextFormattedText 형식이 지정된 시간 제한 텍스트 블록을 나타냅니다.
IMFTimedTextNotify Microsoft Media Foundation 시간 제한 텍스트 알림에 대한 콜백을 정의하는 인터페이스입니다.
IMFTimedTextRegion 시간 제한 텍스트 개체의 표시 영역을 나타냅니다.
IMFTimedTextStyle 시간 제한 텍스트의 스타일을 나타냅니다.
IMFTimedTextTrack 시간 제한 텍스트의 트랙을 나타냅니다.

IMFTimedTextTrackList 시간 제한 텍스트 트랙 목록을 나타냅니다.
IMFTimer 지정된 시간에 콜백을 호출하는 타이머를 제공합니다.
IMFTopoLoader 부분 토폴로지에서 전체 토폴로지로 변환합니다.
IMFTopology 토폴로지 나타냅니다. 토폴로지는 특정 순서로 연결된 미디어 원본, 싱크 및 변환의 컬렉션을 설명합니다.
IMFTopologyNode 토폴로지의 노드를 나타냅니다.
IMFTopologyNodeAttributeEditor 미디어 세션의 현재 토폴로지에서 하나 이상의 노드 특성을 업데이트합니다.
IMFTrackedSample 비디오 미디어 샘플의 참조 횟수를 추적합니다.
IMFTranscodeProfile 코드 변환 프로필 개체에 의해 구현됩니다.
IMFTranscodeSinkInfoProvider 코드 변환 싱크 활성화 개체에 의해 구현됩니다.
IMFTransform 모든 MFT(Media Foundation Transforms)에서 구현됩니다.
IMFTrustedInput ITA(입력 신뢰 기관)를 제공하는 구성 요소에 의해 구현됩니다. 이 인터페이스는 각 구성 요소의 스트림에 대한 ITA를 가져오는 데 사용됩니다.
IMFTrustedOutput OTA(출력 신뢰 기관)를 제공하는 구성 요소에 의해 구현됩니다.
IMFVideoCaptureSampleAllocator 비디오 캡처 디바이스에 대한 특수 기능을 사용하여 비디오 미디어 싱크에 대한 비디오 샘플을 할당합니다.
IMFVideoMediaType 비디오 형식에 대한 설명을 나타냅니다.

IMFVideoProcessorControl 비디오 프로세서 MFT를 구성합니다. (IMFVideoProcessorControl)
IMFVideoProcessorControl2 비디오 프로세서 MFT를 구성합니다. (IMFVideoProcessorControl2)
IMFVideoSampleAllocator 비디오 미디어 싱크에 대한 비디오 샘플을 할당합니다.
IMFVideoSampleAllocatorCallback 애플리케이션이 EVR(고급 비디오 렌더러)에서 할당한 비디오 샘플을 추적할 수 있도록 합니다.
IMFVideoSampleAllocatorEx Microsoft Direct3D 11 텍스처 표면을 포함하는 비디오 샘플을 할당합니다.
IMFVideoSampleAllocatorNotify IMFVideoSampleAllocatorCallback 인터페이스에 대한 콜백입니다. (IMFVideoSampleAllocatorNotify)
IMFVideoSampleAllocatorNotifyEx IMFVideoSampleAllocatorCallback 인터페이스에 대한 콜백입니다. (IMFVideoSampleAllocatorNotifyEx)
IMFVirtualCamera Media Foundation 프레임 서버 파이프라인에 연결할 수 있는 가상 카메라를 나타냅니다.
IMFWorkQueueServices 미디어 세션에서 만든 작업 큐를 제어합니다.
IMFWorkQueueServicesEx IMFWorkQueueServices 인터페이스를 확장합니다.
IOPMVideoOutput OPM(출력 보호 관리자) 세션에 대한 비디오 출력을 나타냅니다.
IPlayToControl PlayToConnection 개체가 미디어 요소에 연결할 수 있도록 합니다.
IPlayToControlWithCapabilities IPlayToSource에 대한 기능을 제공하여 콘텐츠의 기능을 확인합니다.
IPlayToSourceClassFactory PlayToSource 개체의 인스턴스를 만듭니다.

IToc IToc 인터페이스는 개별 목차를 나타냅니다. 목차에 항목을 추가하고 항목을 제거하는 메서드를 제공합니다.
ITocCollection ITocCollection은 목차 컬렉션을 나타냅니다. 컬렉션에서 목차를 추가, 검색 및 제거하는 메서드를 제공합니다.
ITocEntry ITocEntry 인터페이스는 목차의 개별 항목을 나타냅니다. 항목에 대한 설명 정보를 설정하고 검색하는 메서드를 제공합니다.
ITocEntryList ITocEntryList 인터페이스는 목차의 항목 목록을 나타냅니다. 목록에 항목을 추가하고 목록에서 항목을 제거하는 메서드를 제공합니다.
ITocParser ITocParser 인터페이스는 TOC 파서 개체를 나타냅니다. 비디오 파일에 목차를 저장하고 비디오 파일에서 목차를 검색하는 메서드를 제공합니다.
IWMCodecLeakyBucket 비디오 인코더에서 "새는 버킷" 매개 변수를 구성합니다.
IWMCodecOutputTimestamp 디코딩할 다음 비디오 프레임의 타임스탬프를 가져옵니다.
IWMCodecPrivateData 출력 미디어 형식에 추가해야 하는 프라이빗 코덱 데이터를 가져옵니다. 이 코덱 데이터는 Windows Media Video 콘텐츠를 제대로 디코딩하는 데 필요합니다.
IWMCodecProps 형식별 코덱 속성을 검색하는 메서드를 제공합니다.
IWMCodecStrings 코덱 및 형식에 대한 이름 및 설명 문자열을 검색합니다.
IWMColorConvProps 색 변환기 DSP의 속성을 설정합니다.
IWMResamplerProps 오디오 리샘플러 DSP의 속성을 설정합니다.
IWMResizerProps 비디오 리사이저 DSP의 속성을 설정합니다.

IWMSampleExtensionSupport 샘플 확장에 대한 코덱 지원을 구성합니다.
IWMVideoDecoderHurryup 비디오 디코더의 속도를 제어합니다.
IWMVideoDecoderReconBuffer 참고: 이 인터페이스는 사용되지 않으며 사용해서는 안 됩니다. 재구성된 비디오 프레임을 관리합니다.
IWMVideoForceKeyFrame 인코더가 현재 프레임을 키 프레임으로 인코딩하도록 합니다.

구조

테이블 확장

AecQualityMetrics_Struct AEC(음향 에코 취소)에 대한 품질 메트릭을 포함합니다. 이 구조체는 MFPKEY_WMAAECMA_RETRIEVE_TS_STATS 속성과 함께 사용됩니다.
ASF_FLAT_PICTURE 미디어 원본에 대한 메타데이터로 저장된 이미지를 포함합니다. 이 구조는 WM/Picture 메타데이터 특성의 데이터 항목으로 사용됩니다.
ASF_FLAT_SYNCHRONISED_LYRICS 미디어 원본에 대한 메타데이터로 저장된 동기화된 가사를 포함합니다. 이 구조는 WM/Lyrics_Synchronised 메타데이터 특성의 데이터 항목으로 사용됩니다.
ASF_INDEX_DESCRIPTOR 스트림 및 인덱스 유형에 대한 인덱싱 구성에 대해 설명합니다.
ASF_INDEX_IDENTIFIER ASF 인덱서 개체의 인덱스를 지정합니다.
ASF_MUX_STATISTICS ASF 멀티플렉서의 진행률에 대한 통계를 포함합니다.
codecAPIEventData CodecAPIEventData 구조체는 EC_CODECAPI_EVENT 이벤트에 대한 이벤트 데이터를 포함하며 ICodecAPI 인터페이스를 지원하는 코덱에 의해 전송됩니다.

D3D11_AES_CTR_IV 128비트 Advanced Encryption Standard CTR 모드(AES-CTR) 블록 암호화에 대한 IV(초기화 벡터)를 포함합니다. (D3D11_AES_CTR_IV)
D3D11_AUTHENTICATED_CONFIGURE_ACCESSIBLE_ENCRYPTION_INPUT D3D11_AUTHENTICATED_CONFIGURE_ENCRYPTION_WHEN_ACCESSIBLE 명령에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_CONFIGURE_CRYPTO_SESSION_INPUT D3D11_AUTHENTICATED_CONFIGURE_CRYPTO_SESSION 명령에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_CONFIGURE_INITIALIZE_INPUT D3D11_AUTHENTICATED_CONFIGURE_INITIALIZE 명령에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_CONFIGURE_INPUT ID3D11VideoContext::ConfigureAuthenticatedChannel 메서드에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_CONFIGURE_OUTPUT ID3D11VideoContext::ConfigureAuthenticatedChannel 메서드의 응답을 포함합니다.
D3D11_AUTHENTICATED_CONFIGURE_PROTECTION_INPUT D3D11_AUTHENTICATED_CONFIGURE_PROTECTION 명령에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_CONFIGURE_SHARED_RESOURCE_INPUT D3D11_AUTHENTICATED_CONFIGURE_SHARED_RESOURCE 명령에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_QUERY_ACCESSIBILITY_ENCRYPTION_GUID_COUNT_OUTPUT D3D11_AUTHENTICATED_QUERY_ENCRYPTION_WHEN_ACCESSIBLE_GUID_COUNT 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_ACCESSIBILITY_ENCRYPTION_GUID_INPUT D3D11_AUTHENTICATED_QUERY_ENCRYPTION_WHEN_ACCESSIBLE_GUID 쿼리에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_QUERY_ACCESSIBILITY_ENCRYPTION_GUID_OUTPUT D3D11_AUTHENTICATED_QUERY_ENCRYPTION_WHEN_ACCESSIBLE_GUID 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_ACCESSIBILITY_OUTPUT D3D11_AUTHENTICATED_QUERY_ACCESSIBILITY_ATTRIBUTES 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_CHANNEL_TYPE_OUTPUT D3D11_AUTHENTICATED_QUERY_CHANNEL_TYPE 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_CRYPTO_SESSION_INPUT D3D11_AUTHENTICATED_QUERY_CRYPTO_SESSION 쿼리에 대한 입력 데이터를 포함합니다.

D3D11_AUTHENTICATED_QUERY_CRYPTO_SESSION_OUTPUT D3D11_AUTHENTICATED_QUERY_CRYPTO_SESSION 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_CURRENT_ACCESSIBILITY_ENCRYPTION_OUTPUT D3D11_AUTHENTICATED_QUERY_CURRENT_ENCRYPTION_WHEN_ACCESSIBLE 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_DEVICE_HANDLE_OUTPUT D3D11_AUTHENTICATED_QUERY_DEVICE_HANDLE 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_INPUT ID3D11VideoContext::QueryAuthenticatedChannel 메서드에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_QUERY_OUTPUT ID3D11VideoContext::QueryAuthenticatedChannel 메서드의 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_OUTPUT_ID_COUNT_INPUT D3D11_AUTHENTICATED_QUERY_OUTPUT_ID_COUNT 쿼리에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_QUERY_OUTPUT_ID_COUNT_OUTPUT D3D11_AUTHENTICATED_QUERY_OUTPUT_ID_COUNT 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_OUTPUT_ID_INPUT D3D11_AUTHENTICATED_QUERY_OUTPUT_ID 쿼리에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_QUERY_OUTPUT_ID_OUTPUT D3D11_AUTHENTICATED_QUERY_OUTPUT_ID 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_PROTECTION_OUTPUT D3D11_AUTHENTICATED_QUERY_PROTECTION 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_RESTRICTED_SHARED_RESOURCE_PROCESS_COUNT_OUTPUT D3D11_AUTHENTICATED_QUERY_RESTRICTED_SHARED_RESOURCE_PROCESS_COUNT 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_RESTRICTED_SHARED_RESOURCE_PROCESS_INPUT D3D11_AUTHENTICATED_QUERY_RESTRICTED_SHARED_RESOURCE_PROCESS 쿼리에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_QUERY_RESTRICTED_SHARED_RESOURCE_PROCESS_OUTPUT D3D11_AUTHENTICATED_QUERY_RESTRICTED_SHARED_RESOURCE_PROCESS 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_UNRESTRICTED_PROTECTED_SHARED_RESOURCE_COUNT_OUTPUT D3D11_AUTHENTICATED_QUERY_UNRESTRICTED_PROTECTED_SHARED_RESOURCE_COUNT 쿼리에 대한 응답을 포함합니다.

D3D11_ENCRYPTED_BLOCK_INFO 비디오 화면에서 암호화되는 바이트를 지정합니다.
D3D11_KEY_EXCHANGE_HW_PROTECTION_DATA 하드웨어 콘텐츠 보호를 위한 키 교환 데이터를 나타냅니다.
D3D11_KEY_EXCHANGE_HW_PROTECTION_INPUT_DATA 하드웨어 콘텐츠 보호를 위한 키 교환 입력 데이터를 나타냅니다.
D3D11_KEY_EXCHANGE_HW_PROTECTION_OUTPUT_DATA 하드웨어 콘텐츠 보호를 위한 키 교환 출력 데이터를 나타냅니다.
D3D11_OMAC MAC(메시지 인증 코드)를 포함합니다.
D3D11_TEX2D_ARRAY_VPOV 비디오 프로세서 출력 보기에 대한 질감 리소스를 식별합니다. (D3D11_TEX2D_ARRAY_VPOV)
D3D11_TEX2D_VDOV 비디오 디코더 출력 뷰의 질감 리소스를 식별합니다.
D3D11_TEX2D_VPIV 비디오 프로세서 입력 보기의 텍스처 리소스를 식별합니다.
D3D11_TEX2D_VPOV 비디오 프로세서 출력 보기에 대한 질감 리소스를 식별합니다. (D3D11_TEX2D_VPOV)
D3D11_VIDEO_COLOR Microsoft Direct3D 11 비디오의 색 값을 정의합니다.
D3D11_VIDEO_COLOR_RGBA RGB 색 값을 지정합니다. (D3D11_VIDEO_COLOR_RGBA)
D3D11_VIDEO_COLOR_YCbCrA YCbCr 색 값을 지정합니다. (D3D11_VIDEO_COLOR_YCbCrA)
D3D11_VIDEO_CONTENT_PROTECTION_CAPS 그래픽 드라이버의 콘텐츠 보호 기능을 설명합니다.
D3D11_VIDEO_DECODER_BEGIN_FRAME_CRYPTO_SESSION ID3D11VideoContext::D ecoderBeginFrame 메서드에 데이터를 제공합니다.

D3D11_VIDEO_DECODER_BUFFER_DESC 디코딩을 위해 압축된 버퍼에 대해 설명합니다. (D3D11_VIDEO_DECODER_BUFFER_DESC)
D3D11_VIDEO_DECODER_BUFFER_DESC1 디코딩을 위해 압축된 버퍼에 대해 설명합니다. (D3D11_VIDEO_DECODER_BUFFER_DESC1)
D3D11_VIDEO_DECODER_CONFIG DXVA(DirectX Video Acceleration)용 Microsoft Direct3D 11 디코더 디바이스의 구성에 대해 설명합니다.
D3D11_VIDEO_DECODER_DESC Microsoft Direct3D 11 비디오 디코더 또는 비디오 프로세서의 비디오 스트림에 대해 설명합니다.
D3D11_VIDEO_DECODER_EXTENSION ID3D11VideoContext::D ecoderExtension 메서드에 대한 드라이버별 데이터를 포함합니다.
D3D11_VIDEO_DECODER_OUTPUT_VIEW_DESC 비디오 디코더 출력 보기를 설명합니다.
D3D11_VIDEO_DECODER_SUB_SAMPLE_MAPPING_BLOCK 하위 샘플 매핑 블록에 대해 설명합니다.
D3D11_VIDEO_PROCESSOR_CAPS Microsoft Direct3D 11 비디오 프로세서의 기능에 대해 설명합니다.
D3D11_VIDEO_PROCESSOR_COLOR_SPACE 비디오 처리를 위한 색 공간을 지정합니다.
D3D11_VIDEO_PROCESSOR_CONTENT_DESC 비디오 프로세서에 대한 비디오 스트림을 설명합니다.
D3D11_VIDEO_PROCESSOR_CUSTOM_RATE 프레임 속도 변환 또는 IVTC(역 텔레신)에 대한 사용자 지정 속도를 지정합니다. (D3D11_VIDEO_PROCESSOR_CUSTOM_RATE)
D3D11_VIDEO_PROCESSOR_FILTER_RANGE 이미지 필터에 대해 지원되는 값의 범위를 정의합니다. (D3D11_VIDEO_PROCESSOR_FILTER_RANGE)
D3D11_VIDEO_PROCESSOR_INPUT_VIEW_DESC 비디오 프로세서 입력 보기를 설명합니다.
D3D11_VIDEO_PROCESSOR_OUTPUT_VIEW_DESC 비디오 프로세서 출력 보기를 설명합니다.

D3D11_VIDEO_PROCESSOR_RATE_CONVERSION_CAPS 디인터레이싱 및 역 텔레신을 포함하여 프레임 속도 변환과 연결된 비디오 프로세서 기능 그룹을 정의합니다.
D3D11_VIDEO_PROCESSOR_STREAM ID3D11VideoContext::VideoProcessorBlt 메서드에 대한 스트림 수준 데이터를 포함합니다.
D3D11_VIDEO_PROCESSOR_STREAM_BEHAVIOR_HINT ID3DVideoContext1::VideoProcessorGetBehaviorHints 메서드에 전달되는 입력 스트림에 대한 정보를 제공합니다.
D3D11_VIDEO_SAMPLE_DESC 비디오 샘플을 설명합니다.
D3D12_FEATURE_DATA_VIDEO_DECODE_CONVERSION_SUPPORT 지원되는 프로필 목록을 검색합니다. (D3D12_FEATURE_DATA_VIDEO_DECODE_CONVERSION_SUPPORT)
D3D12_FEATURE_DATA_VIDEO_DECODE_FORMATS 지원되는 형식 목록을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_DECODE_HISTOGRAM 지정된 기능이 D3D12_FEATURE_VIDEO_DECODE_HISTOGRAM 경우 ID3D12VideoDevice::CheckFeatureSupport 호출에 대한 데이터를 제공합니다.
D3D12_FEATURE_DATA_VIDEO_DECODE_PROFILES 지원되는 프로필 목록을 검색합니다. (D3D12_FEATURE_DATA_VIDEO_DECODE_PROFILES)
D3D12_FEATURE_DATA_VIDEO_DECODE_SUPPORT 비디오 디코딩에 대한 지원 정보를 검색합니다.
D3D12_FEATURE_DATA_VIDEO_DECODER_HEAP_SIZE 비디오 디코더 힙의 할당 크기를 설명합니다. (D3D12_FEATURE_DATA_VIDEO_DECODER_HEAP_SIZE)
D3D12_FEATURE_DATA_VIDEO_ENCODER_CODEC 지정된 코덱이 비디오 인코딩에 지원되는지 여부를 나타내는 값을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_CODEC_CONFIGURATION_SUPPORT 지정된 코덱 구성 지원 매개 변수가 제공된 HEVC 인코딩 구성에 대해 지원되는지 또는 H.264 인코딩에 대해 지원되는 구성을 검색하는지를 나타내는 값을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_CODEC_PICTURE_CONTROL_SUPPORT 지정된 코덱 및 프로필에 대한 그림 컨트롤 지원을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_FRAME_SUBREGION_LAYOUT_MODE

지정된 코드, 프로필 및 수준에 대해 지정된 프레임 하위 변경 레이아웃 모드가 지원되는지 여부를 나타내는 값을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_HEAP_SIZE 힙 개체의 L0 및 L1 크기뿐만 아니라 비디오 인코딩에 대해 지정된 코덱이 지원되는지 여부를 나타내는 값을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_INPUT_FORMAT 지정된 코덱, 프로필 및 형식이 비디오 인코딩에 지원되는지 여부를 나타내는 값을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_INTRA_REFRESH_MODE 지정된 코덱, 프로필 및 수준에 대해 지정된 인트라 새로 고침 모드가 지원되는지 여부를 나타내는 값을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_OUTPUT_RESOLUTION 지정된 코덱에 대해 지원되는 해결 방법 목록을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_OUTPUT_RESOLUTION_RATIOS_COUNT D3D12_FEATURE_DATA_VIDEO_ENCODER_OUTPUT_RESOLUTION_RATIOS_COUNT 구조체(d3d12video.h)는 지정된 코덱에 대해 지원되는 해상도 비율 수를 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_PROFILE_LEVEL 지정된 프로필이 비디오 인코딩에 지원되는지 여부를 나타내는 값을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_RATE_CONTROL_MODE 지정된 코덱을 사용하여 비디오 인코딩에 대해 지정된 속도 제어 모드가 지원되는지 여부를 나타내는 값을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_RESOLUTION_SUPPORT_LIMITS D3D12_FEATURE_DATA_VIDEO_ENCODER_SUPPORT 구조체에 대한 비디오 인코더 해상도 지원 제한을 나타냅니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_RESOURCE_REQUIREMENTS 지정된 인코딩 구성을 사용하여 비디오 인코딩에 대한 리소스 요구 사항을 나타내는 값을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_SUPPORT 지정된 비디오 인코딩 기능 및 구성 값에 대한 지원을 나타내는 값을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_EXTENSION_COMMAND_COUNT 비디오 확장 명령 수를 검색합니다.
D3D12_FEATURE_DATA_VIDEO_EXTENSION_COMMAND_PARAMETER_COUNT 지정된 매개 변수 단계에 대해 지원되는 매개 변수 수를 검색합니다.
D3D12_FEATURE_DATA_VIDEO_EXTENSION_COMMAND_PARAMETERS 지정된 매개 변수 단계에 대한 비디오 확장 명령 매개 변수 목록을 검색합니다.

D3D12_FEATURE_DATA_VIDEO_EXTENSION_COMMAND_SIZE 비디오 확장 명령의 할당 크기를 확인합니다.
D3D12_FEATURE_DATA_VIDEO_EXTENSION_COMMAND_SUPPORT 명령 정의 입력 및 출력 구조를 사용하여 비디오 확장 명령 지원을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_EXTENSION_COMMANDS 드라이버에서 비디오 확장 명령 목록을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_MOTION_ESTIMATOR 지정된 기능이 D3D12_FEATURE_VIDEO_MOTION_ESTIMATOR 경우 ID3D12VideoDevice::CheckFeatureSupport 호출에 대한 데이터를 제공합니다. 비디오 인코더의 동작 예측 기능을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_MOTION_ESTIMATOR_PROTECTED_RESOURCES 지정된 기능이 D3D12_FEATURE_VIDEO_MOTION_ESTIMATOR_PROTECTED_RESOURCES 경우 ID3D12VideoDevice::CheckFeatureSupport 호출에 대한 데이터를 제공합니다. 비디오 동작 추정에 대한 보호된 리소스 지원을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_MOTION_ESTIMATOR_SIZE 비디오 동작 추정기 힙의 할당 크기를 설명합니다.
D3D12_FEATURE_DATA_VIDEO_PROCESS_MAX_INPUT_STREAMS 비디오 프로세서에서 지원하는 사용 가능한 입력 스트림의 최대 수를 검색합니다.
D3D12_FEATURE_DATA_VIDEO_PROCESS_REFERENCE_INFO 지정된 디인터레이스 모드, 필터, 속도 변환 또는 자동 처리 기능에 필요한 과거 및 이후 참조 프레임 수를 검색합니다.
D3D12_FEATURE_DATA_VIDEO_PROCESS_SUPPORT 지정된 기능이 D3D12_FEATURE_VIDEO_PROCESS_SUPPORT 경우 ID3D12VideoDevice::CheckFeatureSupport 호출에 대한 데이터를 제공합니다.
D3D12_FEATURE_DATA_VIDEO_PROCESSOR_SIZE 비디오 디코더 힙의 할당 크기를 설명합니다. (D3D12_FEATURE_DATA_VIDEO_PROCESSOR_SIZE)
D3D12_QUERY_DATA_VIDEO_DECODE_STATISTICS ID3D12VideoDecodeCommandList::EndQuery를 호출하여 호출된 비디오 디코딩 통계 쿼리의 데이터를 나타냅니다.
D3D12_RESOLVE_VIDEO_MOTION_VECTOR_HEAP_INPUT ID3D12VideoEncodeCommandList::ResolveMotionVectorHeap 호출에 대한 입력 데이터를 제공합니다.
D3D12_RESOLVE_VIDEO_MOTION_VECTOR_HEAP_OUTPUT ID3D12VideoEncodeCommandList::ResolveMotionVectorHeap에 대한 호출에서 출력 데이터를 수신합니다.

D3D12_RESOURCE_COORDINATE 리소스의 좌표를 설명합니다.
D3D12_VIDEO_DECODE_COMPRESSED_BITSTREAM 비디오가 디코딩되는 압축 비트스트림을 나타냅니다.
D3D12_VIDEO_DECODE_CONFIGURATION 비디오 디코더의 구성에 대해 설명합니다.
D3D12_VIDEO_DECODE_CONVERSION_ARGUMENTS 출력 변환 디코딩을 위한 매개 변수를 지정합니다. (D3D12_VIDEO_DECODE_CONVERSION_ARGUMENTS)
D3D12_VIDEO_DECODE_CONVERSION_ARGUMENTS1 출력 변환 디코딩을 위한 매개 변수를 지정합니다. (D3D12_VIDEO_DECODE_CONVERSION_ARGUMENTS1)
D3D12_VIDEO_DECODE_FRAME_ARGUMENT 프레임의 디코딩 매개 변수를 나타냅니다.
D3D12_VIDEO_DECODE_INPUT_STREAM_ARGUMENTS 비디오 디코딩 작업의 입력 스트림에 대한 매개 변수를 지정합니다.
D3D12_VIDEO_DECODE_OUTPUT_HISTOGRAM 단일 구성 요소에 대한 히스토그램 출력 버퍼를 나타냅니다.
D3D12_VIDEO_DECODE_OUTPUT_STREAM_ARGUMENTS 비디오 디코딩 작업의 출력 스트림에 대한 매개 변수를 지정합니다. (D3D12_VIDEO_DECODE_OUTPUT_STREAM_ARGUMENTS)
D3D12_VIDEO_DECODE_OUTPUT_STREAM_ARGUMENTS1 비디오 디코딩 작업의 출력 스트림에 대한 매개 변수를 지정합니다. (D3D12_VIDEO_DECODE_OUTPUT_STREAM_ARGUMENTS1)
D3D12_VIDEO_DECODE_REFERENCE_FRAMES 현재 디코딩 작업에 대한 참조 프레임 목록을 포함합니다.
D3D12_VIDEO_DECODER_DESC ID3D12VideoDecoder에 대해 설명합니다.
D3D12_VIDEO_DECODER_HEAP_DESC ID3D12VideoDecoderHeap에 대해 설명합니다.
D3D12_VIDEO_ENCODE_REFERENCE_FRAMES

인코딩 작업에 대한 재구성된 참조 이미지를 나타냅니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION 비디오 인코딩에 대한 코덱 구성 구조를 나타냅니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_H264 H.264 인코딩에 대한 코덱 구성을 나타냅니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_HEVC HEVC 인코딩에 대한 코덱 구성을 나타냅니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_SUPPORT 비디오 인코딩에 대한 코덱 구성 지원 구조를 나타냅니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_SUPPORT_H264 H.264 인코딩에 대한 인코더 코덱 구성 지원을 나타냅니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_SUPPORT_HEVC HEVC 인코딩에 대한 인코더 코덱 구성 지원을 나타냅니다.
D3D12_VIDEO_ENCODER_CODEC_PICTURE_CONTROL_SUPPORT 여러 코덱에 대한 그림 컨트롤 지원 구조를 나타냅니다.
D3D12_VIDEO_ENCODER_CODEC_PICTURE_CONTROL_SUPPORT_H264 H.264 비디오 인코딩에 대한 그림 컨트롤 지원 설정을 나타냅니다.
D3D12_VIDEO_ENCODER_CODEC_PICTURE_CONTROL_SUPPORT_HEVC HEVC 비디오 인코딩에 대한 그림 컨트롤 지원 설정을 나타냅니다.
D3D12_VIDEO_ENCODER_COMPRESSED_BITSTREAM 인코딩 작업에 대한 압축 비트스트림 출력을 캡슐화합니다.
D3D12_VIDEO_ENCODER_DESC ID3D12VideoEncoder에 대해 설명합니다.
D3D12_VIDEO_ENCODER_ENCODE_OPERATION_METADATA_BUFFER ID3D12VideoEncodeCommandList2::EncodeFrame 작업에 대한 메타데이터를 포함하는 버퍼를 나타냅니다.
D3D12_VIDEO_ENCODER_ENCODEFRAME_INPUT_ARGUMENTS ID3D12VideoEncodeCommandList2::EncodeFrame에 대한 입력 인수를 나타냅니다.
D3D12_VIDEO_ENCODER_ENCODEFRAME_OUTPUT_ARGUMENTS ID3D12VideoEncodeCommandList2::EncodeFrame에 대한 출력 인수를 나타냅니다.

D3D12_VIDEO_ENCODER_FRAME_SUBREGION_METADATA 비디오 인코더 프레임 하위 리소스 메타데이터를 나타냅니다.
D3D12_VIDEO_ENCODER_HEAP_DESC ID3D12VideoEncoderHeap에 대해 설명합니다.
D3D12_VIDEO_ENCODER_INTRA_REFRESH 비디오 인코딩에 대한 인트라 새로 고침 설정을 나타냅니다.
D3D12_VIDEO_ENCODER_LEVEL_SETTING 비디오 인코더 수준 설정을 나타냅니다.
D3D12_VIDEO_ENCODER_LEVEL_TIER_CONSTRAINTS_HEVC HEVC(고효율 비디오 코딩) 수준 설정 구성에 대한 수준 및 계층을 연결합니다.
D3D12_VIDEO_ENCODER_OUTPUT_METADATA ID3D12VideoEncodeCommandList2::EncodeFrame 작업에 대한 메타데이터를 나타냅니다.
D3D12_VIDEO_ENCODER_OUTPUT_METADATA_STATISTICS ID3D12VideoEncodeCommandList2::EncodeFrame 작업에 대한 인코딩 통계를 나타냅니다.
D3D12_VIDEO_ENCODER_PICTURE_CONTROL_CODEC_DATA 여러 코덱에 대한 연결된 EncodeFrame 명령에 대한 그림 수준 컨트롤 요소를 나타냅니다.
D3D12_VIDEO_ENCODER_PICTURE_CONTROL_CODEC_DATA_H264 H.264 인코딩에 연결된 EncodeFrame 명령에 대한 그림 수준 컨트롤 요소를 나타냅니다.
D3D12_VIDEO_ENCODER_PICTURE_CONTROL_CODEC_DATA_H264_REFERENCE_PICTURE_LIST_MODIFICATION_OPERATION H264 비디오 인코딩에 대한 그림 목록 수정 작업을 나타냅니다.
D3D12_VIDEO_ENCODER_PICTURE_CONTROL_CODEC_DATA_H264_REFERENCE_PICTURE_MARKING_OPERATION 참조 그림의 변경 내용을 작업 식별자 및 작업에 필요한 관련 매개 변수의 튜플로 메모리 작업으로 설명합니다.
D3D12_VIDEO_ENCODER_PICTURE_CONTROL_CODEC_DATA_HEVC HEVC 인코딩에 연결된 EncodeFrame 명령에 대한 그림 수준 컨트롤 요소를 나타냅니다.
D3D12_VIDEO_ENCODER_PICTURE_CONTROL_DESC 06/30/2021
D3D12_VIDEO_ENCODER_PICTURE_CONTROL_SUBREGIONS_LAYOUT_DATA 그림 컨트롤 하위 영역을 여러 코덱에 대한 조각으로 정의합니다.

D3D12_VIDEO_ENCODER_PICTURE_CONTROL_SUBREGIONS_LAYOUT_DATA_SLICES 이 분할 모드를 지원하는 코덱에 대한 조각으로 하위 리소스를 정의합니다.
D3D12_VIDEO_ENCODER_PICTURE_RESOLUTION_DESC 비디오 인코더 그림 해상도를 정의합니다.
D3D12_VIDEO_ENCODER_PICTURE_RESOLUTION_RATIO_DESC 해상도 비율을 돌이킬 수 없는 분수로 정의합니다.
D3D12_VIDEO_ENCODER_PROFILE_DESC 인코더 프로필을 설명합니다.
D3D12_VIDEO_ENCODER_RATE_CONTROL 비디오 인코더 속도 제어 구성을 나타냅니다.
D3D12_VIDEO_ENCODER_RATE_CONTROL_CBR 상수 비트 전송률 모드에 대한 속도 제어 구조 정의를 나타냅니다.
D3D12_VIDEO_ENCODER_RATE_CONTROL_CONFIGURATION_PARAMS D3D12_VIDEO_ENCODER_RATE_CONTROL 구조체에 대한 비디오 인코더 속도 제어 구조 정의를 나타냅니다.
D3D12_VIDEO_ENCODER_RATE_CONTROL_CQP 상수 양자화 매개 변수 모드에 대한 속도 제어 구조 정의를 나타냅니다.
D3D12_VIDEO_ENCODER_RATE_CONTROL_QVBR 비트 전송률이 제한된 상수 품질 대상에 대한 속도 제어 구조 정의를 나타냅니다.
D3D12_VIDEO_ENCODER_RATE_CONTROL_VBR 가변 비트 전송률 모드에 대한 속도 제어 구조 정의를 나타냅니다.
D3D12_VIDEO_ENCODER_RECONSTRUCTED_PICTURE 인코딩 작업에 전달된 입력 프레임에서 생성된 재구성된 그림을 나타냅니다.
D3D12_VIDEO_ENCODER_REFERENCE_PICTURE_DESCRIPTOR_H264 H.264 비디오 인코딩에 대한 참조 그림 설명자를 나타냅니다.
D3D12_VIDEO_ENCODER_REFERENCE_PICTURE_DESCRIPTOR_HEVC HEVC 비디오 인코딩에 대한 참조 그림 설명자를 나타냅니다.
D3D12_VIDEO_ENCODER_RESOLVE_METADATA_INPUT_ARGUMENTS ID3D12VideoEncodeCommandList2::ResolveEncoderOutputMetadata 호출에 대한 입력 인수를 나타냅니다.

D3D12_VIDEO_ENCODER_RESOLVE_METADATA_OUTPUT_ARGUMENTS ID3D12VideoEncodeCommandList2::ResolveEncoderOutputMetadata 호출에 대한 출력 인수를 나타냅니다.
D3D12_VIDEO_ENCODER_SEQUENCE_CONTROL_DESC D3D12_VIDEO_ENCODER_SEQUENCE_CONTROL_DESC 시퀀스 수준에서 비디오 인코딩 작업의 입력에 대한 매개 변수를 나타냅니다.
D3D12_VIDEO_ENCODER_SEQUENCE_GOP_STRUCTURE 여러 비디오 코덱의 GOP 구조를 나타냅니다.
D3D12_VIDEO_ENCODER_SEQUENCE_GOP_STRUCTURE_H264 H.264 비디오 인코딩에 대한 GOP 구조를 나타냅니다.
D3D12_VIDEO_ENCODER_SEQUENCE_GOP_STRUCTURE_HEVC HEVC 비디오 인코딩에 대한 GOP 구조를 나타냅니다.
D3D12_VIDEO_EXTENSION_COMMAND_DESC 비디오 확장 명령을 설명합니다. (D3D12_VIDEO_EXTENSION_COMMAND_DESC)
D3D12_VIDEO_EXTENSION_COMMAND_INFO 비디오 확장 명령을 설명합니다. (D3D12_VIDEO_EXTENSION_COMMAND_INFO)
D3D12_VIDEO_EXTENSION_COMMAND_PARAMETER_INFO 비디오 확장 명령 매개 변수에 대해 설명합니다.
D3D12_VIDEO_FORMAT 리소스 콘텐츠 설명에 대한 픽셀 형식과 색 공간의 조합을 정의합니다.
D3D12_VIDEO_MOTION_ESTIMATOR_DESC ID3D12VideoMotionEstimator에 대해 설명합니다. ID3D12VideoDevice1::CreateVideoMotionEstimator에 이 구조를 전달하여 ID3D12VideoMotionEstimator의 인스턴스를 만듭니다.
D3D12_VIDEO_MOTION_ESTIMATOR_INPUT ID3D12VideoEncodeCommandList::EstimateMotion 호출에 대한 입력 매개 변수를 지정합니다.
D3D12_VIDEO_MOTION_ESTIMATOR_OUTPUT ID3D12VideoEncodeCommandList::EstimateMotion 호출에 대한 출력 매개 변수를 지정합니다.
D3D12_VIDEO_MOTION_VECTOR_HEAP_DESC ID3D12VideoMotionEstimatorHeap에 대해 설명합니다. ID3D12VideoDevice1::CreateVideoMotionEstimatorHeap에 이 구조를 전달하여 ID3D12VideoMotionEstimatorHeap의 인스턴스를 만듭니다.
D3D12_VIDEO_PROCESS_ALPHA_BLENDING

비디오 처리를 위한 알파 혼합 매개 변수를 지정합니다.
D3D12_VIDEO_PROCESS_FILTER_RANGE 이미지 필터에 대해 지원되는 값의 범위를 정의합니다. (D3D12_VIDEO_PROCESS_FILTER_RANGE)
D3D12_VIDEO_PROCESS_INPUT_STREAM 비디오 프로세서 혼합 기능에 대한 입력 정보를 포함합니다.
D3D12_VIDEO_PROCESS_INPUT_STREAM_ARGUMENTS ID3D12VideoCommandList::P rocessFrames에 전달된 입력 스트림에 대한 입력 스트림 인수를 지정합니다.
D3D12_VIDEO_PROCESS_INPUT_STREAM_ARGUMENTS1 각 호출에 대한 필드 형식 변경을 지원하는 ID3D12VideoProcessCommandList1::P rocessFrames1에 전달된 입력 스트림에 대한 입력 스트림 인수를 지정합니다.
D3D12_VIDEO_PROCESS_INPUT_STREAM_DESC 비디오 프로세스 작업의 입력 스트림에 대한 매개 변수를 지정합니다.
D3D12_VIDEO_PROCESS_INPUT_STREAM_RATE 스트림 속도에 대한 정보를 제공합니다.
D3D12_VIDEO_PROCESS_LUMA_KEY 루마 키 지정에 사용되는 설정을 지정합니다.
D3D12_VIDEO_PROCESS_OUTPUT_STREAM 비디오 처리 명령에 대한 출력 스트림을 나타냅니다.
D3D12_VIDEO_PROCESS_OUTPUT_STREAM_ARGUMENTS ID3D12VideoCommandList::P rocessFrames에 전달되는 출력에 대한 출력 스트림 인수를 지정합니다.
D3D12_VIDEO_PROCESS_OUTPUT_STREAM_DESC ID3D12VideoProcessCommandList::P rocessFrames에 전달되는 출력에 대한 출력 스트림 인수를 지정합니다.
D3D12_VIDEO_PROCESS_REFERENCE_SET 비디오 처리를 수행하는 데 필요한 참조 프레임을 포함합니다.
D3D12_VIDEO_PROCESS_TRANSFORM 비디오 처리를 위한 변환 매개 변수를 지정합니다.
D3D12_VIDEO_SAMPLE 그림 버퍼의 너비, 높이, 서식 및 색 공간에 대해 설명합니다.
D3D12_VIDEO_SCALE_SUPPORT

비디오 스케일러에 대해 지원되는 출력 크기의 크기 조정 범위에 대해 설명합니다.
D3D12_VIDEO_SIZE_RANGE 비디오 스케일러에 대해 지원되는 크기 범위에 대해 설명합니다.
D3DCONTENTPROTECTIONCAPS 디스플레이 드라이버의 콘텐츠 보호 기능에 대해 설명합니다.
D3DOVERLAYCAPS Direct3D 디바이스에 대한 하드웨어 오버레이 기능을 지정합니다.
DEVICE_INFO 미디어 공유 디바이스에 대한 정보를 포함합니다.
DigitalWindowSetting 비디오 캡처를 위한 디지털 창의 경계 설정을 나타냅니다.
DXVA2_AES_CTR_IV 128비트 Advanced Encryption Standard CTR 모드(AES-CTR) 블록 암호화에 대한 IV(초기화 벡터)를 포함합니다. (DXVA2_AES_CTR_IV)
DXVA2_AYUVSample16 16비트 AYUV 픽셀 값을 정의합니다.
DXVA2_AYUVSample8 8비트 AYUV 픽셀 값을 정의합니다.
DXVA2_ConfigPictureDecode DXVA 디코더 디바이스의 구성에 대해 설명합니다.
DXVA2_DecodeBufferDesc 디코더에서 DXVA(DirectX Video Acceleration) 디바이스로 전송되는 버퍼에 대해 설명합니다.
DXVA2_DecodeExecuteParams IDirectXVideoDecoder::Execute 메서드에 대한 매개 변수를 포함합니다.
DXVA2_DecodeExtensionData IDirectXVideoDecoder::Execute 메서드에 대한 프라이빗 데이터를 포함합니다.
DXVA2_ExtendedFormat 비디오 스트림의 형식을 설명합니다.
DXVA2_FilterValues

DXVA(DirectX Video Acceleration) 이미지 필터에 대한 매개 변수를 포함합니다.
DXVA2_Fixed32 32비트 고정 소수점 숫자를 정의합니다.
DXVA2_Frequency 비디오 빈도를 정의합니다.
DXVA2_ProcAmpValues DXVA(DirectX Video Acceleration) 비디오 처리 작업에 대한 값을 포함합니다.
DXVA2_ValueRange DXVA(DirectX Video Acceleration) 작업에 대해 지원되는 값의 범위를 정의합니다.
DXVA2_VideoDesc DXVA 디코더 디바이스 또는 비디오 프로세서 디바이스에 대한 비디오 스트림을 설명합니다.
DXVA2_VideoProcessBltParams IDirectXVideoProcessor::VideoProcessBlt 메서드에 대한 매개 변수를 포함합니다.
DXVA2_VideoProcessorCaps DVXA(DirectX Video Acceleration) 비디오 프로세서 모드의 기능에 대해 설명합니다.
DXVA2_VideoSample IDirectXVideoProcessor::VideoProcessBlt 메서드에 대한 입력 샘플을 지정합니다.
DXVABufferInfo IDirect3DDXVADevice9::Execute 메서드에 대한 버퍼를 지정합니다.
DXVACompBufferInfo DXVA(DirectX Video Acceleration)의 압축된 표면에 대한 요구 사항을 지정합니다.
DXVAHD_BLT_STATE_ALPHA_FILL_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용하는 경우 출력 알파 값이 블릿 작업에 대해 계산되는 방법을 지정합니다.
DXVAHD_BLT_STATE_BACKGROUND_COLOR_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용하는 경우 블릿 작업의 배경색을 지정합니다.
DXVAHD_BLT_STATE_CONSTRICTION_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용할 때 출력이 블릿 작업에서 다운샘플링되는지 여부를 지정합니다.
DXVAHD_BLT_STATE_OUTPUT_COLOR_SPACE_DATA

Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용하는 경우 블릿 작업의 출력 색 공간을 지정합니다.
DXVAHD_BLT_STATE_PRIVATE_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)의 프라이빗 블릿 상태에 대한 데이터를 포함합니다.
DXVAHD_BLT_STATE_TARGET_RECT_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용하는 경우 블리팅을 위한 대상 사각형을 지정합니다.
DXVAHD_COLOR DXVA-HD의 색 값을 정의합니다.
DXVAHD_COLOR_RGBA RGB 색 값을 지정합니다. (DXVAHD_COLOR_RGBA)
DXVAHD_COLOR_YCbCrA YCbCr 색 값을 지정합니다. (DXVAHD_COLOR_YCbCrA)
DXVAHD_CONTENT_DESC Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서의 비디오 스트림에 대해 설명합니다.
DXVAHD_CUSTOM_RATE_DATA 프레임 속도 변환 또는 IVTC(역 텔레신)에 대한 사용자 지정 속도를 지정합니다. (DXVAHD_CUSTOM_RATE_DATA)
DXVAHD_FILTER_RANGE_DATA 이미지 필터에 대해 지원되는 값의 범위를 정의합니다. (DXVAHD_FILTER_RANGE_DATA)
DXVAHD_RATIONAL 합리적인 숫자(비율)를 포함합니다.
DXVAHD_STREAM_DATA IDXVAHD_VideoProcessor::VideoProcessBltHD 메서드에 대한 스트림별 데이터를 포함합니다.
DXVAHD_STREAM_STATE_ALPHA_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용하는 경우 입력 스트림의 평면 알파 값을 지정합니다.
DXVAHD_STREAM_STATE_ASPECT_RATIO_DATA 원본 및 대상 사각형의 PAR(픽셀 가로 세로 비율)을 지정합니다.
DXVAHD_STREAM_STATE_D3DFORMAT_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용하는 경우 입력 스트림의 형식을 지정합니다.
DXVAHD_STREAM_STATE_DESTINATION_RECT_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용하는 경우 입력 스트림의 대상 사각형을 지정합니다.

DXVAHD_STREAM_STATE_FILTER_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 입력 스트림에 대한 필터링 작업의 수준을 지정합니다.
DXVAHD_STREAM_STATE_FRAME_FORMAT_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 입력 스트림이 인터레이스되는 방법을 지정합니다.
DXVAHD_STREAM_STATE_INPUT_COLOR_SPACE_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 입력 스트림의 색 공간을 지정합니다.
DXVAHD_STREAM_STATE_LUMA_KEY_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용하는 경우 입력 스트림의 루마 키를 지정합니다.
DXVAHD_STREAM_STATE_OUTPUT_RATE_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용할 때 입력 스트림의 출력 프레임 속도를 지정합니다.
DXVAHD_STREAM_STATE_PALETTE_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용하는 경우 입력 스트림에 대한 색상표 항목을 포함합니다.
DXVAHD_STREAM_STATE_PRIVATE_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 입력 스트림의 프라이빗 스트림 상태에 대한 데이터를 포함합니다.
DXVAHD_STREAM_STATE_PRIVATE_IVTC_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스의 역 텔레신(IVTC) 통계를 포함합니다.
DXVAHD_STREAM_STATE_SOURCE_RECT_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용할 때 입력 스트림의 원본 사각형을 지정합니다.
DXVAHD_VPCAPS Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서의 기능을 지정합니다.
DXVAHD_VPDEVCAPS Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스의 기능을 지정합니다.
DXVAHDSW_CALLBACKS Microsoft DirectX 비디오 가속 고화질(DXVA-HD)용 소프트웨어 플러그 인에서 구현하는 함수에 대한 포인터를 포함합니다.
DXVAUncompDataInfo DXVA(DirectX Video Acceleration) 비디오 디코딩을 위해 압축되지 않은 표면의 크기 및 픽셀 형식을 지정합니다.
MACROBLOCK_DATA 비디오 디코딩 중에 macroblock에 대한 데이터를 제공합니다.

MF_BYTE_STREAM_CACHE_RANGE 바이트 범위를 지정합니다.
MF_CAMERA_CONTROL_RANGE_INFO MF_CAMERA_CONTROL_RANGE_INFO 구조체는 카메라 컨트롤에 허용되는 범위, 단계 값 및 기본값을 나타냅니다.
MF_FLOAT2 두 구성 요소가 있는 벡터입니다.
MF_FLOAT3 세 가지 구성 요소가 있는 벡터입니다.
MF_LEAKY_BUCKET_PAIR 파일의 버퍼링 요구 사항을 지정합니다.
MF_QUATERNION 회전을 나타내는 데 사용되는 4차원 벡터입니다.
MF_SINK_WRITER_STATISTICS 싱크 기록기의 성능에 대한 통계를 포함합니다.
MF_TRANSCODE_SINK_INFO 트랜스코드 싱크 활성화 개체의 오디오 및 비디오 스트림에 대한 정보를 포함합니다.
MFARGB ARGB 색 값을 설명합니다.
MFASYNCRESULT IMFAsyncResult 인터페이스를 구현하는 데 필요한 데이터를 포함합니다.
MFAYUVSample 4:4:4:4 Y'Cb'Cr' 샘플을 설명합니다.
MFBYTESTREAM_BUFFERING_PARAMS 네트워크 바이트 스트림에 대한 버퍼링 매개 변수를 지정합니다.
MFCameraExtrinsic_CalibratedTransform 다른 카메라 또는 설정된 외부 참조를 기준으로 카메라의 위치를 설명하는 변환입니다.
MFCameraExtrinsics 다른 카메라 또는 설정된 외부 참조를 기준으로 카메라의 위치를 설명합니다.

MFCameraIntrinsic_DistortionModel 다항식 렌즈 왜곡 모델을 나타냅니다.
MFCameraIntrinsic_PinholeCameraModel 핀홀 카메라 모델을 나타냅니다.
MFCLOCK_PROPERTIES 클록의 속성을 정의합니다.
MFCONTENTPROTECTIONDEVICE_INPUT_DATA 보호 시스템 함수에 대한 입력으로 제공하려는 데이터에 대한 정보를 포함합니다.
MFCONTENTPROTECTIONDEVICE_OUTPUT_DATA 보호 시스템 함수에서 출력으로 받은 데이터에 대한 정보를 포함합니다.
MFCONTENTPROTECTIONDEVICE_REALTIMECLIENT_DATA 실시간 작업을 예상 우선 순위로 예약할 수 있도록 MMCSS(멀티미디어 클래스 스케줄러 서비스) 매개 변수의 보안 프로세서를 권장합니다.
MFFOLDDOWN_MATRIX 다중 채널 오디오를 더 적은 수의 오디오 채널로 변환하는 데 사용되는 계수를 포함합니다. 이 프로세스를 접는 프로세스라고 합니다.
MFINPUTTRUSTAUTHORITY_ACCESS_ACTION OTA(출력 신뢰 기관)에서 요청한 작업에 대해 설명합니다. 요청은 ITA(입력 신뢰 기관)로 전송됩니다.
MFINPUTTRUSTAUTHORITY_ACCESS_PARAMS IMFInputTrustAuthority::BindAccess 또는 IMFInputTrustAuthority::UpdateAccess 메서드에 대한 매개 변수를 포함합니다.
MFMediaKeyStatus CDM(콘텐츠 암호 해독 모듈) 세션 키의 상태를 나타냅니다.
MFMPEG2DLNASINKSTATS DLNA(Digital Living Network Alliance) 미디어 싱크의 인코딩 통계를 포함합니다.
MFNetCredentialManagerGetParam 자격 증명 관리자에 대한 인증 정보를 포함합니다.
MFOffset 오프셋을 고정 소수점 실수로 지정합니다.
MFPaletteEntry

색상표에 하나의 색상표 항목을 포함합니다.
MFPinholeCameraIntrinsic_IntrinsicModel 지정된 해상도에 대한 핀홀 카메라 내장 모델을 나타냅니다.
MFPinholeCameraIntrinsics 실제 세계의 3D 지점을 카메라의 2D 이미지 프레임에 투사하는 방법을 설명하는 0개 또는 1개의 핀홀 카메라 내장 모델이 포함되어 있습니다.
MFRatio 비율을 나타냅니다.
MFRR_COMPONENT_HASH_INFO 해지된 구성 요소에 대한 정보를 포함합니다.
MFRR_COMPONENTS 하나 이상의 해지된 구성 요소에 대한 정보를 포함합니다.
MFT_INPUT_STREAM_INFO MFT(Media Foundation 변환)의 입력 스트림에 대한 정보를 포함합니다. 이러한 값을 얻으려면 IMFTransform::GetInputStreamInfo를 호출합니다.
MFT_OUTPUT_DATA_BUFFER Media Foundation 변환에 대한 출력 버퍼에 대한 정보를 포함합니다. 이 구조체는 IMFTransform::P rocessOutput 메서드에 사용됩니다.
MFT_OUTPUT_STREAM_INFO MFT(Media Foundation 변환)의 출력 스트림에 대한 정보를 포함합니다. 이러한 값을 얻으려면 IMFTransform::GetOutputStreamInfo를 호출합니다.
MFT_REGISTER_TYPE_INFO MFT(Media Foundation 변환)를 등록하기 위한 미디어 형식 정보를 포함합니다.
MFT_REGISTRATION_INFO IMFLocalMFTRegistration::RegisterMFTs 메서드에 대한 매개 변수를 포함합니다.
MFTOPONODE_ATTRIBUTE_UPDATE 토폴로지 노드에 대한 새 특성 값을 지정합니다.
MFVideoArea 비디오 프레임 내의 사각형 영역을 지정합니다.
MFVideoCompressedInfo 비디오 압축 형식에 대한 정보를 포함합니다. 이 구조체는 MFVIDEOFORMAT 구조체에 사용됩니다.

MFVIDEOFORMAT 비디오 형식을 설명합니다.
MFVideoInfo 압축된 형식과 압축되지 않은 형식 모두에 적용되는 비디오 형식 정보를 포함합니다. 이 구조체는 MFVIDEOFORMAT 구조체에 사용됩니다.
MFVideoNormalizedRect MFVideoNormalizedRect(mfcaptureengine.h) 구조체는 비디오 사각형에서 하위 사각형을 지정하는 데 사용되는 정규화된 사각형을 정의합니다.
MFVideoNormalizedRect MFVideoNormalizedRect(mfmediaengine.h) 구조체는 비디오 사각형에서 하위 사각형을 지정하는 데 사용되는 정규화된 사각형을 정의합니다.
MFVideoSurfaceInfo 압축되지 않은 비디오 형식에 대한 정보를 포함합니다. 이 구조체는 MFVIDEOFORMAT 구조체에 사용됩니다.
MT_ARBITRARY_HEADER ASF(고급 스트리밍 형식) 파일의 이진 스트림에 대한 형식 데이터를 포함합니다.
MT_CUSTOM_VIDEO_PRIMARIES 비디오 원본에 대한 사용자 지정 색 주체를 정의합니다. 색 주체는 RGB 색 공간에서 CIE XYZ 색 공간으로 색을 변환하는 방법을 정의합니다.
OPM_ACP_AND_CGMSA_SIGNALING OPM_GET_ACP_AND_CGMSA_SIGNALING 쿼리의 결과를 포함합니다.
OPM_ACTUAL_OUTPUT_FORMAT OPM(Output Protection Manager)의 OPM_GET_ACTUAL_OUTPUT_FORMAT 쿼리 결과를 포함합니다.
OPM_CONFIGURE_PARAMETERS OPM(Output Protection Manager) 또는 COPP(Certified Output Protection Manager) 명령을 포함합니다.
OPM_CONNECTED_HDCP_DEVICE_INFORMATION OPM_GET_CONNECTED_HDCP_DEVICE_INFORMATION 쿼리의 결과를 포함합니다.
OPM_COPP_COMPATIBLE_GET_INFO_PARAMETERS IOPMVideoOutput::COPPCompatibleGetInformation 메서드에 대한 매개 변수를 포함합니다.
OPM_ENCRYPTED_INITIALIZATION_PARAMETERS OPM(출력 보호 관리자) 세션에 대한 암호화된 초기화 매개 변수를 포함합니다. (OPM_ENCRYPTED_INITIALIZATION_PARAMETERS).

OPM_ENCRYPTED_INITIALIZATION_PARAMETERS OPM_ENCRYPTED_INITIALIZATION_PARAMETERS(ksopmapi.h) 구조체에는 OPM(출력 보호 관리자) 세션에 대한 초기화 매개 변수가 포함되어 있습니다.
OPM_GET_CODEC_INFO_INFORMATION OPM_GET_CODEC_INFO_INFORMATION(opmapi.h)에는 OPM_GET_CODEC_INFO 쿼리의 결과가 포함됩니다.
OPM_GET_CODEC_INFO_INFORMATION OPM_GET_CODEC_INFO_INFORMATION(ksopmapi.h) 구조체에는 OPM_GET_CODEC_INFO 쿼리의 결과가 포함됩니다.
OPM_GET_CODEC_INFO_PARAMETERS OPM_GET_CODEC_INFO_PARAMETERS(opmapi.h)에는 OPM_GET_CODEC_INFO 명령에 대한 정보가 포함되어 있습니다.
OPM_GET_CODEC_INFO_PARAMETERS OPM_GET_CODEC_INFO_PARAMETERS(ksopmapi.h) 구조체에는 OPM_GET_CODEC_INFO 명령에 대한 정보가 포함됩니다.
OPM_GET_INFO_PARAMETERS OPM_GET_INFO_PARAMETERS(opmapi.h)에는 IOPMVideoOutput::GetInformation 메서드에 대한 매개 변수가 포함되어 있습니다.
OPM_GET_INFO_PARAMETERS OPM_GET_INFO_PARAMETERS(ksopmapi.h) 구조체에는 IOPMVideoOutput::GetInformation 메서드에 대한 매개 변수가 포함됩니다.
OPM_HDCP_KEY_SELECTION_VECTOR High-Bandwidth HDCP(Digital Content Protection) 수신기에 대한 KSV(키 선택 벡터)를 포함합니다.
OPM_OMAC OPM_OMAC(opmapi.h)에는 OPM(출력 보호 관리자) 메시지에 대한 MAC(메시지 인증 코드)가 포함되어 있습니다.
OPM_OMAC OPM_OMAC(ksopmapi.h) 구조에는 OPM(출력 보호 관리자) 메시지에 대한 MAC(메시지 인증 코드)가 포함됩니다.
OPM_OUTPUT_ID_DATA OPM_GET_OUTPUT_ID 상태 요청의 결과를 포함합니다.
OPM_RANDOM_NUMBER OPM_RANDOM_NUMBER(opmapi.h)에는 OPM(출력 보호 관리자)에 사용할 128비트 난수가 포함되어 있습니다.
OPM_RANDOM_NUMBER OPM_RANDOM_NUMBER(ksopmapi.h) 구조체에는 OPM(출력 보호 관리자)에 사용할 128비트 난수가 포함됩니다.
OPM_REQUESTED_INFORMATION

OPM_REQUESTED_INFORMATION(opmapi.h)에는 OPM(출력 보호 관리자) 상태 요청의 결과가 포함됩니다.
OPM_REQUESTED_INFORMATION OPM_REQUESTED_INFORMATION(ksopmapi.h) 구조에는 OPM(출력 보호 관리자) 상태 요청의 결과가 포함됩니다.
OPM_SET_ACP_AND_CGMSA_SIGNALING_PARAMETERS OPM(출력 보호 관리자)의 OPM_SET_ACP_AND_CGMSA_SIGNALING 명령에 대한 정보를 포함합니다.
OPM_SET_HDCP_SRM_PARAMETERS OPM_SET_HDCP_SRM 명령에 대한 매개 변수를 포함합니다.
OPM_SET_PROTECTION_LEVEL_PARAMETERS OPM(출력 보호 관리자)의 OPM_SET_PROTECTION_LEVEL 명령에 대한 데이터를 포함합니다.
OPM_STANDARD_INFORMATION OPM_STANDARD_INFORMATION(opmapi.h)에는 OPM(출력 보호 관리자) 상태 요청의 결과가 포함됩니다.
OPM_STANDARD_INFORMATION OPM_STANDARD_INFORMATION(ksopmapi.h) 구조체에는 OPM(출력 보호 관리자) 상태 요청의 결과가 포함됩니다.
ROI_AREA 관심 있는 지역을 정의합니다.
SENSORPROFILEID 센서 프로필 ID를 설명합니다.
STREAM_MEDIUM 애플리케이션용이 아닙니다.
TOC_DESCRIPTOR TOC_DESCRIPTOR 구조체는 목차에 대한 설명 정보를 보유합니다.
TOC_ENTRY_DESCRIPTOR TOC_ENTRY_DESCRIPTOR 구조체는 목차의 항목에 대한 설명 정보를 보유합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

모바일 광대역

아티클2023. 08. 22.

모바일 광대역 기술 개요입니다.

모바일 광대역을 개발하려면 다음 헤더가 필요합니다.

- mbnapi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 모바일 광대역

열거형

MBN_ACTIVATION_STATE MBN_ACTIVATION_STATE 열거형 형식은 현재 데이터 연결 상태를 나타냅니다.
MBN_AUTH_PROTOCOL MBN_AUTH_PROTOCOL 열거형 형식은 PDP(패킷 데이터 프로토콜) 활성화에 사용되는 인증 프로토콜을 지정합니다.
MBN_BAND_CLASS MBN_BAND_CLASS 열거형 형식은 빈도 대역 클래스를 정의합니다.
MBN_CELLULAR_CLASS MBN_CELLULAR_CLASS 열거형 형식은 셀룰러 디바이스의 유형을 정의합니다.
MBN_COMPRESSION MBN_COMPRESSION 열거형 형식은 헤더 및 데이터에 대한 데이터 링크에서 압축을 사용할지 여부를 지정합니다.
MBN_CONNECTION_MODE MBN_CONNECTION_MODE 열거형 형식은 요청된 연결 모드를 지정합니다.
MBN_CONTEXT_CONSTANTS MBN_CONTEXT_CONSTANTS 열거형 형식은 MBN_CONTEXT 구조체의 멤버가 지원하는 최대 문자열 길이를 지정합니다.

MBN_CONTEXT_TYPE MBN_CONTEXT_TYPE 열거형 형식은 표시된 컨텍스트 형식을 지정합니다.
MBN_CTRL_CAPS MBN_CTRL_CAPS 열거형 형식은 모든 모바일 광대역 디바이스 제어 기능을 비트 필드로 나타냅니다.
MBN_DATA_CLASS MBN_DATA_CLASS 열거형 형식은 공급자가 지원하는 데이터 클래스를 지정합니다.
MBN_DEVICE_SERVICES_INTERFACE_STATE . (MBN_DEVICE_SERVICES_INTERFACE_STATE)
MBN_INTERFACE_CAPS_CONSTANTS MBN_INTERFACE_CAPS_CONSTANTS 열거형 형식은 MBN_INTERFACE_CAPS 구조체의 다양한 요소에서 사용하는 문자열 값의 최대 길이를 정의합니다.
MBN_MSG_STATUS MBN_MSG_STATUS 열거형 형식은 처리되는 메시지 유형을 정의합니다.
MBN_PIN_CONSTANTS MBN_PIN_CONSTANTS 열거형 형식은 MBN_PIN_INFO 구조체에서 사용하는 상수 값을 정의합니다.
MBN_PIN_FORMAT MBN_PIN_FORMAT 열거형 형식은 PIN이 숫자인지 영숫자인지 여부를 나타냅니다.
MBN_PIN_MODE MBN_PIN_MODE 열거형 형식은 PIN 형식을 사용할 수 있는지를 나타냅니다.
MBN_PIN_STATE MBN_PIN_STATE 열거형 형식은 모바일 광대역 디바이스의 현재 PIN 상태를 나타냅니다.
MBN_PIN_TYPE MBN_PIN_TYPE 열거형 형식은 인터페이스에 저장된 정보의 잠금을 해제하는 데 필요한 암호 유형을 나타냅니다.

MBN_PROVIDER_CONSTANTS MBN_PROVIDER_CONSTANTS 열거형 형식에는 MBN_PROVIDER 멤버의 버퍼 길이를 정의하는 값이 포함되어 있습니다.
MBN_PROVIDER_STATE MBN_PROVIDER_STATE 열거형 형식은 공급자 항목에 태그를 지정할 수 있는 다양한 상태를 지정합니다.
MBN_RADIO MBN_RADIO 열거형 형식은 디바이스 라디오가 켜지거나 꺼져 있는지 여부를 나타냅니다.
MBN_READY_STATE MBN_READY_STATE 열거형 형식에는 모바일 광대역 디바이스가 셀룰러 네트워크 트래픽 작업에 참여할 준비가 되었음을 나타내는 값이 포함되어 있습니다.
MBN_REGISTER_MODE MBN_REGISTER_MODE 열거형 형식은 디바이스의 네트워크 선택 모드를 나타냅니다.
MBN_REGISTER_STATE MBN_REGISTER_STATE 열거형 형식은 모바일 광대역 디바이스의 네트워크 등록 상태를 나타냅니다.
MBN_REGISTRATION_CONSTANTS MBN_REGISTRATION_CONSTANTS 열거형 형식에는 IMbnRegistration 인터페이스 작업에서 사용하는 특정 값이 포함됩니다.
MBN_SIGNAL_CONSTANTS MBN_SIGNAL_CONSTANTS 열거형 형식에는 IMbnSignal 인터페이스 작업에서 사용하는 특정 값이 포함됩니다.
MBN_SMS_CAPS MBN_SMS_CAPS 열거형 형식에는 SMS 기능을 지정하는 비트 필드 값이 포함되어 있습니다.
MBN_SMS_CDMA_ENCODING MBN_SMS_CDMA_ENCODING 열거형 형식은 CDMA에 대한 문자 인코딩 형식을 지정합니다.
MBN_SMS_CDMA_LANG

MBN_SMS_CDMA_LANG 열거형 형식은 CDMA 메시지에 사용할 수 있는 다양한 언어를 나타냅니다.
MBN_SMS_FLAG MBN_SMS_FLAG 열거형 형식은 SMS 메시지 클래스를 지정합니다.
MBN_SMS_FORMAT SMS 메시지의 형식입니다.
MBN_SMS_STATUS_FLAG MBN_SMS_STATUS_FLAG 열거형 형식은 디바이스의 SMS 메시지 저장소 상태 나타냅니다.
MBN_VOICE_CALL_STATE MBN_VOICE_CALL_STATE 열거형 형식은 디바이스의 현재 음성 통화 상태를 지정합니다.
MBN_VOICE_CLASS MBN_VOICE_CLASS 열거형 형식은 디바이스의 음성 기능과 데이터 서비스와 상호 작용하는 방법을 지정합니다.
WWAEXT_SMS_CONSTANTS MBN_SMS_CONSTANTS 열거형 형식에는 SMS 상수 값이 포함됩니다.

Functions

활성화 네트워크로 서비스 활성화 요청을 보냅니다.
변경 PIN을 변경합니다.
CloseCommandSession 모바일 광대역 디바이스의 디바이스 서비스에 대한 명령 세션을 닫습니다.
CloseDataSession 모바일 광대역 디바이스의 디바이스 서비스에 대한 데이터 세션을 닫습니다.

연결 데이터 연결을 설정합니다.
CreateConnectionProfile 디바이스에 대한 새 연결 프로필을 만듭니다.
삭제 시스템에서 프로필을 삭제합니다.
사용 안 함 PIN을 사용하지 않도록 설정합니다.
연결 끊기 데이터 연결의 연결을 끊습니다.
사용 PIN을 사용하도록 설정합니다.
Enter PIN을 입력합니다.
EnumerateDeviceServices 모바일 광대역 디바이스에서 지원되는 디바이스 서비스 목록을 가져옵니다.
get_Address 메시지와 연결된 휴대폰 번호입니다.
get_CdmaShortMsgSize 최대 CDMA 짧은 메시지 문자 길이입니다.
get_ConnectionID 연결에 대한 고유 식별자를 가져옵니다.
get_DeviceServiceID 이 개체가 연결된 디바이스 서비스의 ID입니다.
get_EncodingID

메시지에 사용되는 데이터 인코딩입니다.
get_HardwareRadioState 모바일 광대역 디바이스의 하드웨어 라디오 상태입니다.
get_Index 디바이스 SMS 저장소에 있는 메시지의 인덱스입니다.
get_Index 디바이스 SMS 메모리에 있는 메시지의 인덱스입니다.
get_InterfaceID 인터페이스 식별자를 가져옵니다.
get_InterfaceID 이 개체가 연결된 모바일 광대역 디바이스의 인터페이스 ID입니다.
get_InterfaceID 인터페이스 ID입니다.
get_IsCommandSessionOpen 디바이스 서비스 명령 세션이 열려 있는지 보고합니다.
get_IsDataSessionOpen 디바이스 서비스 데이터 세션이 열려 있는지 보고합니다.
get_LanguageID 메시지에 사용되는 언어입니다.
get_MaxCommandSize 디바이스 서비스 SET 또는 QUERY 명령과 연결할 수 있는 데이터의 최대 길이(바이트)입니다.
get_MaxDataSize 디바이스 서비스 세션에서 쓰거나 읽을 수 있는 데이터의 최대 길이(바이트)입니다.
get_MaxMessageIndex SMS 메시지 메모리 용량.

get_Message CDMA 디바이스에서 사용하는 WMT 형식의 메시지입니다.
get_Message 메시지의 내용입니다.
get_PduData GSM 디바이스에서 사용하는 16진수 형식의 PDU 메시지입니다.
get_PinFormat PIN 형식입니다.
get_PinLengthMax PIN의 최대 길이입니다.
get_PinLengthMin PIN의 최소 길이입니다.
get_PinMode PIN 모드입니다.
get_PinType PIN 유형입니다.
get_ServiceCenterAddress SMS 기본 서비스 센터 주소입니다. (Get)
get_SimIccID 디바이스에 대한 SIM International 회로 카드 번호(SimICCID)입니다.
get_SizeInCharacters 메시지의 문자 크기입니다.
get_SmsFormat 디바이스에서 새로 받은 SMS를 보고해야 하는 형식입니다. (Get)
get_SoftwareRadioState

모바일 광대역 디바이스의 소프트웨어 라디오 상태입니다.
get_Status 메시지의 형식입니다. (IMbnSmsReadMsgPdu.get_Status)
get_Status 메시지의 형식입니다. (IMbnSmsReadMsgTextCdma.get_Status)
get_SubscriberID 디바이스의 구독자 ID입니다.
get_TelephoneNumbers 디바이스와 연결된 전화 번호입니다.
get_Timestamp 메시지의 타임스탬프입니다.
GetActivationNetworkError PDP(패킷 데이터 프로토콜) 컨텍스트 활성화 실패에서 반환된 네트워크 오류를 가져옵니다.
GetAvailableDataClasses 현재 네트워크에서 사용 가능한 데이터 클래스를 가져옵니다.
GetConnection 연결을 가져옵니다.
GetConnection IMbnConnection 개체를 가져옵니다.
GetConnectionProfile 지정된 모바일 광대역 디바이스와 연결된 특정 연결 프로필을 가져옵니다.
GetConnectionProfiles 디바이스와 연결된 연결 프로필 목록을 가져옵니다.
GetConnections 사용 가능한 연결 목록을 가져옵니다.

GetConnectionState 디바이스의 현재 연결 상태를 가져옵니다.
GetCurrentCellularClass 다중 통신 사업자 디바이스에 대한 현재 셀룰러 클래스를 가져옵니다.
GetCurrentDataClass 현재 네트워크의 현재 데이터 클래스를 가져옵니다.
GetDeviceService 모바일 광대역 디바이스에서 디바이스 서비스와 통신하는 데 사용할 수 있는 IMbnDeviceService 개체를 가져옵니다.
GetDeviceServicesContext 특정 모바일 광대역 디바이스에 대한 IMbnDeviceServicesContext 인터페이스를 가져옵니다.
GetHomeProvider 홈 공급자를 가져옵니다.
GetInterface 특정 인터페이스를 가져옵니다.
GetInterfaceCapability 디바이스의 기능을 가져옵니다.
GetInterfaces 사용 가능한 모든 IMbnInterface 개체의 목록을 가져옵니다.
GetPacketAttachNetworkError 패킷 연결 작업에서 네트워크 오류를 가져옵니다.
GetPin 특정 유형의 PIN을 가져옵니다.
GetPinList 디바이스에서 지원하는 다양한 PIN 유형의 목록을 가져옵니다.

GetPinManager IMbnPinManager를 가져옵니다.
GetPinState 디바이스의 현재 PIN 상태를 가져옵니다.
GetPreferredProviders 기본 설정 공급자 목록을 가져옵니다.
GetPreferredProviders 다중 통신 사업자 디바이스의 현재 영역에 표시되는 구독된 공급자 목록을 가져오고 현재 등록된 공급자를 뺀 값입니다.
GetProfileXmlData 현재 프로필의 XML 데이터를 가져옵니다.
GetProviderID 현재 등록된 네트워크의 공급자 ID를 가져옵니다.
GetProviderName 현재 등록된 네트워크의 공급자 이름을 가져옵니다.
GetProvisionedContexts 연결 컨텍스트 목록을 가져옵니다.
GetReadyState 준비 상태를 가져옵니다.
GetRegisterMode 모바일 광대역 디바이스의 네트워크 등록 모드를 가져옵니다.
GetRegisterState 등록 상태를 가져옵니다.
GetRegistrationNetworkError 등록 작업에서 네트워크 오류를 가져옵니다.

GetRoamingText 로밍 공급자를 설명하는 로밍 텍스트를 가져옵니다.
GetSignalError 수신된 신호 오류율을 가져옵니다.
GetSignalStrength 디바이스에서 받은 신호 강도를 가져옵니다.
GetSmsConfiguration 디바이스의 SMS 구성을 가져옵니다.
GetSmsStatus 디바이스에 대한 SMS 상태 가져옵니다.
GetSubscriberInformation 구독자 정보를 가져옵니다.
GetSupportedCellularClasses 다중 통신 사업자 디바이스에 대해 지원되는 셀룰러 클래스 목록을 가져옵니다.
GetVisibleProviders 표시되는 공급자 목록을 가져옵니다.
GetVisibleProviders 다중 통신 사업자 디바이스의 현재 영역에 표시되는 공급자 목록을 가져오고 기본 설정 및 등록된 공급자를 뺀 값입니다.
GetVoiceCallState 디바이스의 음성 통화 상태를 가져옵니다.
InEmergencyMode 디바이스가 비상 모드인지 여부를 확인합니다.
OnActivationComplete 서비스 활성화 요청이 완료되었음을 나타내기 위해 모바일 광대역 서비스에서 호출한 알림 메서드입니다.

OnChangeComplete PIN 변경 작업이 완료되었음을 나타내기 위해 모바일 광대역 서비스에서 호출한 알림 메서드입니다.
OnCloseCommandSessionComplete 디바이스 서비스 CloseCommandSession 요청이 완료되었음을 나타내는 알림 메서드입니다.
OnCloseDataSessionComplete 디바이스 서비스 세션 CloseDataSession 요청이 완료되었음을 나타내는 알림 메서드입니다.
OnConnectComplete 연결 작업의 완료를 알리는 알림 메서드입니다.
OnConnectionArrival 시스템에 새 연결이 추가되었음을 나타내는 알림 메서드입니다.
OnConnectionProfileArrival 새 연결 프로필이 시스템에 추가되었음을 나타내는 알림 메서드입니다.
OnConnectionProfileRemoval 연결 프로필이 시스템에서 제거되었음을 나타내는 알림 메서드입니다.
OnConnectionRemoval 시스템에서 연결이 제거되었음을 나타내는 알림 메서드입니다.
OnConnectStateChange 디바이스의 연결 상태가 변경되었는지 여부를 나타내는 알림 메서드입니다.
OnCurrentCellularClassChange 이 알림 메서드는 GetCurrentCellularClass 작업의 완료를 나타내기 위해 모바일 광대역 서비스에서 호출됩니다.
OnDisableComplete PIN 사용 안 함 작업이 완료되었음을 나타내기 위해 모바일 광대역 서비스에서 호출하는 알림 메서드입니다.
OnDisconnectComplete

연결 끊기 작업이 수행되었음을 나타내는 알림 메서드입니다.
OnEmergencyModeChange 이 알림 방법은 모바일 광대역 서비스에서 긴급 모드가 변경되었음을 나타내기 위해 호출됩니다.
OnEnableComplete PIN 사용 작업이 완료되었음을 나타내기 위해 모바일 광대역 서비스에서 호출한 알림 방법입니다.
OnEnterComplete PIN 입력 작업이 완료되었음을 나타내기 위해 모바일 광대역 서비스에서 호출하는 알림 메서드입니다.
OnEventNotification 모바일 광대역 디바이스에서 디바이스 서비스 상태 변경 이벤트를 알리는 알림 방법입니다.
OnEventNotification 기본 모바일 광대역 디바이스 미니포트 드라이버에서 변경 이벤트를 알리는 알림 방법입니다.
OnGetPinStateComplete IMbnPinManager의 GetPinState 메서드 호출에 의해 트리거되는 비동기 작업의 완료를 나타내기 위해 모바일 광대역 서비스에서 호출한 알림 메서드입니다.
OnHomeProviderAvailable 이 알림 방법은 모바일 광대역 서비스에서 호출되어 디바이스에 대한 홈 공급자 정보를 사용할 수 있음을 나타냅니다.
OnInterfaceArrival 디바이스가 시스템에 추가되었음을 알리는 알림 방법입니다.
OnInterfaceCapabilityAvailable 이 알림 방법은 모바일 광대역 서비스에서 인터페이스 기능 정보를 사용할 수 있음을 나타내기 위해 호출됩니다.
OnInterfaceCapabilityChange 이 알림 메서드는 인터페이스 기능을 업데이트하는 SetHomeProvider 작업의 완료를 나타내기 위해 모바일 광대역 서비스에서 호출됩니다.
OnInterfaceRemoval

디바이스가 시스템에서 제거되었음을 알리는 알림 방법입니다.
OnInterfaceStateChange 시스템의 디바이스 서비스 상태 변경을 알리는 알림 방법입니다.
OnOpenCommandSessionComplete 디바이스 서비스 CommandSessionOpen 요청이 완료되었음을 나타내는 알림 메서드입니다.
OnOpenDataSessionComplete 디바이스 서비스 OpenDataSession 요청이 완료되었음을 나타내는 알림 메서드입니다.
OnPacketServiceStateChange 모바일 광대역 서비스에서 디바이스 패킷 서비스 상태의 변경을 나타내는 알림 메서드를 호출합니다.
OnPinListAvailable 모바일 광대역 서비스에서 디바이스 PIN 목록을 사용할 수 있음을 나타내는 알림 메서드입니다.
OnPreferredProvidersChange 이 알림 방법은 모바일 광대역 서비스에서 호출되어 디바이스의 기본 설정 공급자 목록의 변경 사항을 나타냅니다.
OnPreferredProvidersChange 이 알림 방법은 모바일 광대역 서비스에서 호출하여 GetPreferredProviders 작업의 완료 및 디바이스의 기본 설정 공급자 목록 변경을 나타냅니다.
OnProfileUpdate 프로필 업데이트 작업이 완료되었음을 나타내는 알림 메서드입니다.
OnProvisionedContextListChange 모바일 광대역 서비스에서 디바이스에 저장된 프로비전된 컨텍스트를 사용하거나 업데이트할 수 있음을 나타내는 알림 메서드입니다.
OnQueryCommandComplete 디바이스 서비스 QUERY 요청이 완료되었음을 나타내는 알림 메서드입니다.
OnQuerySupportedCommandsComplete 디바이스 서비스에서 지원되는 메시지에 대한 쿼리가 완료되었음을 나타내는 알림 메서드입니다.

OnRadioStateChange 디바이스의 라디오 상태가 변경되었음을 알리는 알림입니다.
OnReadData 디바이스 서비스 데이터 세션에서 읽는 데이터에 대한 알림입니다.
OnReadyStateChange 이 알림 메서드는 모바일 광대역 서비스에서 호출되어 인터페이스의 준비 상태 변경을 나타냅니다.
OnRegisterModeAvailable 등록 모드 정보를 사용할 수 있음을 나타내기 위해 모바일 광대역 서비스에서 호출하는 알림 방법입니다.
OnRegisterStateChange 모바일 광대역 서비스에서 디바이스의 등록 상태 변경을 나타내는 알림 메서드를 호출합니다.
OnScanNetworkComplete 이 알림 방법은 모바일 광대역 서비스에서 네트워크 검사 완료를 나타내기 위해 호출됩니다.
OnScanNetworkComplete 이 알림 메서드는 Mobile 광대역 서비스에서 호출되어 ScanNetwork 작업의 완료를 나타냅니다.
OnSetCommandComplete 디바이스 서비스 SET 요청이 완료되었음을 나타내는 알림 방법입니다.
OnSetHomeProviderComplete 이 알림 메서드는 Mobile Broadband 서비스에서 SetHomeProvider 작업의 완료를 나타내기 위해 호출됩니다.
OnSetPreferredProvidersComplete 이 알림 메서드는 모바일 광대역 서비스에서 SetPreferredProviders 작업의 완료를 나타내기 위해 호출됩니다.
OnSetProvisionedContextComplete 모바일 광대역 서비스에서 디바이스의 프로비전된 컨텍스트가 설정되었음을 나타내는 알림 메서드입니다.

OnSetRegisterModeComplete 모바일 광대역 서비스에서 설정된 등록 작업을 완료했음을 나타내기 위해 호출된 알림 방법입니다.
OnSetSmsConfigurationComplete 설정된 SMS 구성 작업이 완료되었거나 SMS 하위 시스템이 초기화되어 작업할 준비가 되었음을 알리는 알림 방법입니다.
OnSetSoftwareRadioStateComplete 설정된 소프트웨어 라디오 상태 작업이 완료되었음을 알립니다.
OnSetVendorSpecificComplete 공급업체별 작업이 완료되었음을 나타내는 알림 방법입니다.
OnSignalStateChange 이 알림 방법은 모바일 광대역 서비스에서 신호 품질 업데이트를 사용할 수 있음을 나타내기 위해 호출됩니다.
OnSmsConfigurationChange SMS 구성이 변경되었거나 사용할 수 있음을 나타내는 알림 메서드입니다.
OnSmsDeleteComplete SMS 삭제 작업의 완료를 알리는 알림 메서드입니다.
OnSmsNewClass0Message 새 클래스 0/플래시 메시지의 도착을 알리는 알림 방법입니다.
OnSmsReadComplete 메시지 읽기 작업의 완료를 나타내는 알림 메서드입니다.
OnSmsSendComplete 메시지 보내기 작업의 완료를 나타내는 알림 메서드입니다.
OnSmsStatusChange 메시지 저장소의 상태 변경을 나타내는 알림 메서드입니다.
OnSubscriberInformationChange

이 알림 방법은 모바일 광대역 서비스에서 호출되어 디바이스의 구독자 정보가 변경되었음을 나타냅니다.
OnUnblockComplete PIN 차단 해제 작업이 완료되었음을 나타내기 위해 모바일 광대역 서비스에서 호출한 알림 방법입니다.
OnVoiceCallStateChange 디바이스의 음성 통화 상태 변경을 나타내는 알림 방법입니다.
OnWriteDataComplete 디바이스 서비스 세션 쓰기 요청이 완료되었음을 나타내는 알림 방법입니다.
OpenCommandSession 모바일 광대역 디바이스의 디바이스 서비스에 대한 명령 세션을 엽니다.
OpenDataSession 모바일 광대역 디바이스에서 디바이스 서비스에 대한 데이터 세션을 엽니다.
put_ServiceCenterAddress SMS 기본 서비스 센터 주소입니다. (Put)
put_SmsFormat 디바이스에서 새로 받은 SMS를 보고해야 하는 형식입니다. (Put)
QueryCommand QUERY 컨트롤 명령을 모바일 광대역 디바이스의 디바이스 서비스로 보냅니다.
QuerySupportedCommands 모바일 광대역 디바이스 서비스에서 지원하는 명령 ID 목록을 가져옵니다.
ScanNetwork 네트워크를 비동기적으로 검사하여 표시되는 공급자 목록을 가져옵니다.
ScanNetwork 네트워크를 검사하여 다중 통신 사업자 디바이스에 대해 표시되는 공급자 목록을 가져옵니다.
SetCommand

SET 컨트롤 명령을 모바일 광대역 디바이스의 디바이스 서비스로 보냅니다.
SetHomeProvider 다중 통신 사업자 디바이스에 대한 홈 공급자를 업데이트.
SetPreferredProviders 디바이스에 대한 기본 공급자 목록을 업데이트.
SetProvisionedContext 프로비전된 컨텍스트를 추가하거나 업데이트합니다.
SetRegisterMode 디바이스의 등록 모드를 설정합니다.
SetSmsConfiguration 디바이스에 대한 SMS 구성을 업데이트.
SetSoftwareRadioState 모바일 광대역 디바이스의 소프트웨어 라디오 상태를 설정합니다.
SetVendorSpecific 기본 모바일 광대역 디바이스 미니포트 드라이버에 요청을 보냅니다.
SmsDelete 디바이스에서 SMS 메시지 집합을 삭제합니다.
SmsRead 디바이스에서 SMS 메시지 집합을 읽습니다.
SmsSendCdma CDMA 형식으로 메시지를 보냅니다.
SmsSendCdmaPdu CDMA 이진 형식으로 메시지를 보냅니다.
SmsSendPdu PDU 형식으로 메시지를 보냅니다.

차단을 해제 차단된 PIN의 차단을 해제합니다.
UpdateProfile 프로필 내용을 업데이트.
WriteData 디바이스 서비스 데이터 세션에 데이터를 씁니다.

인터페이스

IMbnConnection 디바이스의 네트워크 연결을 나타냅니다.
IMbnConnectionContext 연결 컨텍스트를 관리합니다.
IMbnConnectionContextEvents 이 알림 인터페이스는 비동기 프로비전된 컨텍스트 이벤트를 처리하는 데 사용됩니다.
IMbnConnectionEvents 이 알림 인터페이스는 비동기 연결 요청의 변경 및 완료 상태 대해 애플리케이션에 신호를 보냅니다.
IMbnConnectionManager IMbnConnection 개체 및 연결 알림에 대한 액세스를 제공합니다.
IMbnConnectionManagerEvents 이 알림 인터페이스는 시스템에서 IMbnConnection 인터페이스의 도착 및 제거에 대해 애플리케이션에 신호를 보냅니다.
IMbnConnectionProfile 이 인터페이스는 모바일 광대역 프로필에 저장된 연결 매개 변수 및 기본 설정에 액세스합니다.

IMbnConnectionProfileEvents 이 알림 인터페이스는 IMbnConnectionProfile 메서드가 모바일 광대역 프로필 상태를 변경할 때 애플리케이션에 신호를 보냅니다.
IMbnConnectionProfileManager 연결 프로필 및 연결 알림에 대한 액세스를 제공합니다.
IMbnConnectionProfileManagerEvents 이 알림 인터페이스는 시스템에서 IMbnConnectionProfile 인터페이스의 도착 및 제거에 대해 애플리케이션에 신호를 보냅니다.
IMbnDeviceService 특정 모바일 광대역 디바이스에서 디바이스 서비스와 통신할 수 있습니다.
IMbnDeviceServicesContext 시스템에서 모바일 광대역 디바이스 개체를 열거하고 검색할 수 있습니다.
IMbnDeviceServicesEvents 시스템의 모바일 광대역 디바이스 서비스와 관련된 알림 이벤트에 대해 애플리케이션에 신호를 보냅니다.
IMbnDeviceServicesManager IMbnDeviceServicesContext 개체 및 모바일 광대역 디바이스 서비스 알림에 대한 액세스를 제공합니다.
IMbnInterface 모바일 광대역 디바이스를 나타냅니다.
IMbnInterfaceEvents 이 인터페이스는 비동기 IMbnInterface 메서드 호출 및 디바이스 상태 변경을 처리하는 데 사용되는 알림 인터페이스입니다.
IMbnInterfaceManager IMbnInterface 개체 및 알림에 대한 액세스를 제공합니다.
IMbnInterfaceManagerEvents 이 알림 인터페이스는 시스템에서 디바이스의 도착 및 제거에 대해 애플리케이션에 신호를 보냅니다.

IMbnMultiCarrier 이 인터페이스는 지원되는 모바일 광대역 디바이스의 다중 통신 사업자 기능을 노출합니다.
IMbnMultiCarrierEvents 이 인터페이스는 비동기 IMbnMultiCarrier 메서드 호출을 처리하는 데 사용되는 알림 인터페이스입니다.
IMbnPin 디바이스 PIN을 나타냅니다.
IMbnPinEvents 이 인터페이스는 비동기 PIN 요청이 완료된 시기를 나타내는 데 사용되는 알림 인터페이스입니다.
IMbnPinManager 디바이스 PIN에 대한 중요한 세부 정보를 제공합니다.
IMbnPinManagerEvents PIN 관리자 이벤트가 발생한 시기를 나타내는 데 사용되는 알림 인터페이스입니다.
IMbnRadio IMbnRadio 인터페이스는 모바일 광대역 디바이스의 라디오 상태를 쿼리하고 업데이트하는 데 사용됩니다.
IMbnRadioEvents 라디오 상태의 변경과 상태의 프로그래밍 방식 변경 완료를 나타내는 데 사용되는 알림 인터페이스입니다.
IMbnRegistration 네트워크 등록 데이터에 대한 액세스를 제공합니다.
IMbnRegistrationEvents 등록 이벤트가 발생한 시기를 나타내는 데 사용되는 알림 인터페이스입니다.
IMbnServiceActivation 셀룰러 서비스 활성화를 위한 통과 메커니즘입니다.
IMbnServiceActivationEvents

이 알림 인터페이스는 서비스 활성화 요청의 완료에 대해 애플리케이션에 신호를 보냅니다.
IMbnSignal 모바일 광대역 연결의 무선 신호 품질을 가져옵니다.
IMbnSignalEvents 신호 이벤트가 발생했음을 나타내는 데 사용되는 알림 인터페이스입니다.
IMbnSms 메시지 전송 및 수신 및 메시징 구성 제어를 위한 SMS 인터페이스입니다.
IMbnSmsConfiguration 디바이스의 SMS 구성에 대한 액세스를 제공합니다.
IMbnSmsEvents 이 알림 인터페이스는 SMS 작업 및 디바이스 SMS 상태 변경 내용의 완료 상태 애플리케이션에 신호를 보냅니다.
IMbnSmsReadMsgPdu 디바이스 메모리에서 읽은 SMS 메시지를 나타내는 속성 컬렉션입니다.
IMbnSmsReadMsgTextCdma 디바이스 메모리에서 읽은 CDMA 형식 SMS 메시지를 나타내는 속성 컬렉션입니다.
IMbnSubscriberInformation 구독자 정보에 대한 액세스를 제공합니다.
IMbnVendorSpecificEvents 이 알림 인터페이스는 공급업체별 작업 및 기타 공급업체별 디바이스 상태 변경 상태 완료를 애플리케이션에 알릴 수 있습니다.
IMbnVendorSpecificOperation 애플리케이션의 요청을 기본 모바일 광대역 미니포트 드라이버로 전달하는 인터페이스입니다.

구조체

MBN_CONTEXT MBN_CONTEXT 구조체는 연결 컨텍스트에 대한 정보를 저장합니다.
MBN_DEVICE_SERVICE MBN_DEVICE_SERVICE 구조는 모바일 광대역 디바이스 서비스에 대한 정보를 제공합니다.
MBN_INTERFACE_CAPS MBN_INTERFACE_CAPS 구조체는 인터페이스 기능을 나타냅니다.
MBN_PIN_INFO MBN_PIN_INFO 구조체는 디바이스의 현재 PIN 상태를 나타냅니다.
MBN_PROVIDER MBN_PROVIDER 구조는 네트워크 서비스 공급자를 나타냅니다.
MBN_PROVIDER2 MBN_PROVIDER2 구조는 네트워크 서비스 공급자를 나타냅니다. IMbnMultiCarrier 인터페이스의 여러 공급자별 메서드에서 사용되며 다중 통신 사업자를 지원하기 위해 MBN_PROVIDER 확장을 제공합니다.
MBN_SMS_FILTER MBN_SMS_FILTER 구조에는 SMS 메시지 집합을 설명하는 값이 포함되어 있습니다.
MBN_SMS_STATUS_INFO MBN_SMS_STATUS_INFO 구조에는 디바이스의 SMS 메시지 저장소 상태 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

모바일 장치 관리 등록

아티클2023. 08. 22.

모바일 장치 관리 등록 기술 개요입니다.

모바일 장치 관리 등록을 개발하려면 다음 헤더가 필요합니다.

- mdmregistration.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 모바일 장치 관리 등록

열거형

REGISTRATION_INFORMATION_CLASS 디바이스 등록에 대한 정보를 포함합니다.

Functions

DiscoverManagementService MDM 서비스를 검색합니다.
DiscoverManagementServiceEx 후보 서버를 사용하여 MDM 서비스를 검색합니다.
GetDeviceManagementConfigInfo 공급자 ID와 연결된 구성 정보를 가져옵니다.
GetDeviceRegistrationInfo 디바이스 등록 정보를 검색합니다.
GetManagementAppHyperlink MDM 서비스와 연결된 관리 앱 하이퍼링크를 검색합니다.

IsDeviceRegisteredWithManagement 디바이스가 MDM 서비스에 등록되어 있는지 확인합니다.
IsManagementRegistrationAllowed 로컬 정책에서 MDM 등록이 허용되는지 여부를 확인합니다.
RegisterDeviceWithManagement [MS-MDE]:_Mobile 디바이스 등록 프로토콜을 사용하여 MDM 서비스에 디바이스를 등록합니다.
RegisterDeviceWithManagementUsingAADCredentials AAD(Azure Active Directory) 자격 증명을 사용하여 MDM 서비스에 디바이스를 등록합니다.
RegisterDeviceWithManagementUsingAADDeviceCredentials AAD(Azure Active Directory) 디바이스 자격 증명을 사용하여 MDM 서비스에 디바이스를 등록합니다.
SetDeviceManagementConfigInfo 공급자 ID와 연결된 구성 정보를 설정합니다.
SetManagedExternally 디바이스가 외부에서 관리되고 MDM 서비스에 등록되지 않음을 MDM 에이전트에 나타냅니다.
UnregisterDeviceWithManagement MDM 서비스를 사용하여 디바이스 등록을 취소합니다.

구조체

MANAGEMENT_SERVICE_INFO 관리 서비스에 대한 엔드포인트 및 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

모바일 장치 관리 설정 공급자

아티클2023. 03. 09.

모바일 장치 관리 설정 공급자 기술 개요입니다.

모바일 장치 관리 설정 공급자 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 모바일 장치 관리 설정 공급자

모니터 구성

아티클2023. 03. 16.

모니터 구성 기술 개요입니다.

모니터 구성을 개발하려면 다음 헤더가 필요합니다.

- highlevelmonitorconfigurationapi.h
- lowlevelmonitorconfigurationapi.h
- physicalmonitorenumerationapi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 구성 모니터링

열거형

MC_COLOR_TEMPERATURE 모니터의 색 온도를 설명합니다.
MC_DISPLAY_TECHNOLOGY_TYPE 모니터 디스플레이 기술을 식별합니다.
MC_DRIVE_TYPE 모니터의 빨간색, 녹색 또는 파란색 드라이브를 설정할지 또는 가져올지를 지정합니다.
MC_GAIN_TYPE 모니터의 빨간색, 녹색 또는 파랑 게인을 가져올지 또는 설정할지를 지정합니다.
MC_POSITION_TYPE 모니터 표시 영역의 세로 또는 가로 위치를 가져올지 또는 설정할지를 지정합니다.
MC_SIZE_TYPE 모니터 표시 영역의 너비 또는 높이를 가져올지 또는 설정할지를 지정합니다.
MC_VCP_CODE_TYPE VCP(가상 제어판) 코드 형식에 대해 설명합니다.

함수

CapabilitiesRequestAndCapabilitiesReply 모니터의 기능을 설명하는 문자열을 검색합니다.
DegaussMonitor 모니터를 분해합니다.
DestroyPhysicalMonitor 실제 모니터에 대한 핸들을 닫습니다.
DestroyPhysicalMonitors 물리적 모니터 핸들의 배열을 닫습니다.
GetCapabilitiesStringLength 모니터의 기능 문자열 길이를 검색합니다.
GetMonitorBrightness 모니터의 최소, 최대 및 현재 밝기 설정을 검색합니다.
GetMonitorCapabilities 모니터의 구성 기능을 검색합니다. 이 함수를 호출하여 모니터에서 지원하는 상위 수준 모니터 구성 함수를 확인합니다.
GetMonitorColorTemperature 모니터의 현재 색 온도를 검색합니다.
GetMonitorContrast 모니터의 최소, 최대 및 현재 대비 설정을 검색합니다.
GetMonitorDisplayAreaPosition 모니터의 최소, 최대값 및 현재 가로 또는 세로 위치를 검색합니다.
GetMonitorDisplayAreaSize 모니터의 최소, 최대 및 현재 너비 또는 높이를 검색합니다.

GetMonitorRedGreenOrBlueDrive 모니터의 빨간색, 녹색 또는 파란색 드라이브 값을 검색합니다.
GetMonitorRedGreenOrBlueGain 모니터의 빨강, 녹색 또는 파랑 게인 값을 검색합니다.
GetMonitorTechnologyType 모니터에서 사용하는 기술 유형을 검색합니다.
GetNumberOfPhysicalMonitorsFromHMONITOR HMONITOR 모니터 핸들과 연결된 실제 모니터 수를 검색합니다.
GetNumberOfPhysicalMonitorsFromIDirect3DDevice9 Direct3D 디바이스와 연결된 실제 모니터 수를 검색합니다.
GetPhysicalMonitorsFromHMONITOR HMONITOR 모니터 핸들과 연결된 물리적 모니터를 검색합니다.
GetPhysicalMonitorsFromIDirect3DDevice9 Direct3D 디바이스와 연결된 물리적 모니터를 검색합니다.
GetTimingReport 모니터의 가로 및 세로 동기화 빈도를 검색합니다.
GetVCPFeatureAndVCPFeatureReply 모니터에 대한 VCP(가상 제어판) 코드의 현재 값, 최대값 및 코드 형식을 검색합니다.
RestoreMonitorFactoryColorDefaults 모니터의 색 설정을 공장 기본값으로 복원합니다.
RestoreMonitorFactoryDefaults 모니터의 설정을 공장 기본값으로 복원합니다.
SaveCurrentMonitorSettings 현재 모니터 설정을 디스플레이의 비휘발성 스토리지에 저장합니다. (SaveCurrentMonitorSettings)

SaveCurrentSettings 현재 모니터 설정을 디스플레이의 비휘발성 스토리지에 저장합니다. (SaveCurrentSettings)
SetMonitorBrightness 모니터의 밝기 값을 설정합니다.
SetMonitorColorTemperature 모니터의 색 온도를 설정합니다.
SetMonitorContrast 모니터의 대비 값을 설정합니다.
SetMonitorDisplayAreaPosition 모니터 표시 영역의 가로 또는 세로 위치를 설정합니다.
SetMonitorDisplayAreaSize 모니터 표시 영역의 너비 또는 높이를 설정합니다.
SetMonitorRedGreenOrBlueDrive 모니터의 빨간색, 녹색 또는 파란색 드라이브 값을 설정합니다.
SetMonitorRedGreenOrBlueGain 모니터의 빨간색, 녹색 또는 파란색 게인 값을 설정합니다.
SetVCPFeature 모니터에 대한 VCP(가상 제어판) 코드의 값을 설정합니다.

구조체

MC_TIMING_REPORT 모니터의 타이밍 보고서의 정보를 포함합니다.
PHYSICAL_MONITOR 실제 모니터에 해당하는 핸들 및 텍스트 설명을 포함합니다.

MS-CHAP 암호 관리 API

아티클2023. 08. 22.

MS-CHAP 암호 관리 API 기술 개요입니다.

MS-CHAP 암호 관리 API를 개발하려면 다음 헤더가 필요합니다.

- mschapp.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- MS-CHAP 암호 관리 API

Functions

MSChapSrvChangePassword 사용자 계정의 암호를 변경합니다.
MSChapSrvChangePassword2 MSChapSrvChangePassword2 함수는 상호 암호화를 지원하면서 사용자 계정의 암호를 변경합니다.

구조체

CYPHER_BLOCK CYPHER_BLOCK OWF(단방향 함수) 암호 해시에 대한 스토리지의 기본 단위입니다.
ENCRYPTED_LM_OWF_PASSWORD ENCRYPTED_LM_OWF_PASSWORD 사용자의 암호화된 LM(Lan Manager) OWF(단방향 함수) 암호 해시를 저장합니다.
LM_OWF_PASSWORD LM_OWF_PASSWORD 사용자 암호의 LM(Lan Manage) OWF(단방향 함수)를 저장합니다.
SAMPR_ENCRYPTED_USER_PASSWORD

SAMPR_ENCRYPTED_USER_PASSWORD 사용자의 암호화된 암호를 저장합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

멀티캐스트 주소 MADCAP(동적 클라이언트 할당 프로토콜)

아티클2023. 08. 22.

멀티캐스트 주소 MADCAP(동적 클라이언트 할당 프로토콜) 기술의 개요입니다.

멀티캐스트 MADCAP(Address Dynamic Client Allocation Protocol)를 개발하려면 다음 헤더가 필요합니다.

- madcapcl.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 멀티캐스트 주소 MADCAP(동적 클라이언트 할당 프로토콜)

Functions

McastApiCleanup McastApiCleanup 함수는 McastApiStartup으로 할당된 리소스의 할당을 취소합니다. McastApiCleanup 함수는 McastApiStartup을 성공적으로 호출한 후에만 호출해야 합니다.
McastApiStartup McastApiStartup 함수는 요청 클라이언트와 시스템에 구현된 MADCAP 버전 간의 MADCAP 버전 협상을 용이하게 합니다.
McastEnumerateScopes McastEnumerateScopes 함수는 네트워크에서 사용할 수 있는 멀티캐스트 범위를 열거합니다.
McastGenUID McastGenUID 함수는 고유 식별자를 생성하며, 이후에 클라이언트가 주소를 요청하고 갱신하는 데 사용합니다.
McastReleaseAddress McastReleaseAddress 함수는 MCAST 서버에서 임대된 멀티캐스트 주소를 해제합니다.
McastRenewAddress McastRenewAddress 함수는 MADCAP 서버에서 하나 이상의 멀티캐스트 주소를 갱신합니다.

McastRequestAddress McastRequestAddress 함수는 MADCAP 서버에서 하나 이상의 멀티캐스트 주소를 요청합니다.

구조체

IPNG_ADDRESS IPNG_ADDRESS 공용 구조체는 IPv4(인터넷 프로토콜 버전 4) 및 IPv6(인터넷 프로토콜 버전 6) 주소를 제공합니다.
MCAST_CLIENT_UID MCAST_CLIENT_UID 구조는 각 멀티캐스트 요청에 대한 고유 클라이언트 식별자를 설명합니다.
MCAST_LEASE_REQUEST MCAST_LEASE_REQUEST 구조는 지정된 멀티캐스트 scope 대한 요청, 갱신 또는 릴리스 매개 변수를 정의합니다. MCAST_API_VERSION_1 구현에서는 한 번에 하나의 IP 주소만 할당할 수 있습니다.
MCAST_LEASE_RESPONSE MCAST_LEASE_RESPONSE 구조는 멀티캐스트 임대 요청에 응답하는 데 사용됩니다.
MCAST_SCOPE_CTX MCAST_SCOPE_CTX 구조는 멀티캐스트 주소와의 프로그래밍 방식 상호 작용을 위한 scope 컨텍스트를 정의합니다. MCAST_SCOPE_CTX 구조체는 MADCAP 주소를 할당, 갱신 또는 해제하기 위한 핸들로 다양한 MADCAP 함수에서 사용됩니다.
MCAST_SCOPE_ENTRY MCAST_SCOPE_ENTRY 구조체는 지정된 멀티캐스트 scope 대한 전체 정보 집합을 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

네이티브 와이파이

아티클2023. 08. 22.

네이티브 Wifi 기술 개요입니다.

네이티브 Wifi를 개발하려면 다음 헤더가 필요합니다.

- adhoc.h
- dot1x.h
- wlanapi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 네이티브 와이파이

열거형

DOT11_ADHOC_AUTH_ALGORITHM 임시 네트워크의 사용자 또는 컴퓨터 인증에 대한 인증 알고리즘을 지정합니다.
DOT11_ADHOC_CIPHER_ALGORITHM 임시 네트워크에서 정보를 암호화하고 암호를 해독하는 데 사용되는 암호 알고리즘을 지정합니다.
DOT11_ADHOC_CONNECT_FAIL_REASON 연결 시도가 실패한 이유를 지정합니다.
DOT11_ADHOC_NETWORK_CONNECTION_STATUS 임시 네트워크의 연결 상태를 지정합니다.
ONEX_AUTH_IDENTITY 802.1X 인증 상태 사용되는 ID의 가능한 값을 지정합니다.
ONEX_AUTH_RESTART_REASON 802.1X 인증이 다시 시작되는 가능한 이유를 지정합니다.
ONEX_AUTH_STATUS 802.1X 인증 상태 가능한 값을 지정합니다.

ONEX_EAP_METHOD_BACKEND_SUPPORT 인증 서버에서 802.1X 인증을 위해 지원되는 에 구성된 EAP 메서드가 지원되는지 여부에 대한 가능한 값을 지정합니다.
ONEX_NOTIFICATION_TYPE 802.1X 모듈 알림에 대한 WLAN_NOTIFICATION_DATA 구조체의 NotificationCode 멤버의 가능한 값을 지정합니다.
ONEX_REASON_CODE 802.1X 인증이 실패한 이유를 나타내는 가능한 값을 지정합니다.
WL_DISPLAY_PAGES 무선 프로필 사용자 인터페이스 대화 상자가 나타날 때 활성 탭을 지정합니다.
WLAN_CONNECTION_MODE 연결 모드를 정의합니다.
WLAN_FILTER_LIST_TYPE 필터 목록의 유형을 나타냅니다.
WLAN_HOSTED_NETWORK_NOTIFICATION_CODE 무선 Hosted Network에서 수신된 알림에 대한 NotificationCode 매개 변수의 가능한 값을 지정합니다.
WLAN_HOSTED_NETWORK_OPCODE 무선 Hosted Network에서 쿼리하거나 설정할 속성에 대한 작업 코드의 가능한 값을 지정합니다.
WLAN_HOSTED_NETWORK_PEER_AUTH_STATE 무선 Hosted Network에서 피어의 인증 상태에 사용할 수 있는 값을 지정합니다.
WLAN_HOSTED_NETWORK_REASON 무선 Hosted Network 함수 호출 결과에 사용할 수 있는 값을 지정합니다.
WLAN_HOSTED_NETWORK_STATE 무선 Hosted Network의 네트워크 상태에 사용할 수 있는 값을 지정합니다.
WLAN_INTERFACE_TYPE

무선 인터페이스 유형을 지정합니다.
WLAN_SECURABLE_OBJECT 네이티브 Wifi Functions에서 사용하는 보안 개체를 정의합니다.

Functions

복제 새 열거형 인터페이스를 만듭니다. (IEnumDot11AdHocInterfaces.Clone)
복제 새 열거형 인터페이스를 만듭니다. (IEnumDot11AdHocNetworks.Clone)
복제 새 열거형 인터페이스를 만듭니다. (IEnumDot11AdHocSecuritySettings.Clone)
CommitCreatedNetwork 생성된 네트워크를 초기화하고 필요에 따라 네트워크의 프로필을 프로필 저장소에 커밋합니다.
연결 이전에 만든 무선 임시 네트워크에 연결합니다.
CreateNetwork 무선 임시 네트워크를 만듭니다.
DeleteProfile 네트워크와 연결된 프로필을 삭제합니다.
연결 끊기 임시 네트워크에서 연결을 끊습니다.
GetActiveNetwork 인터페이스에서 현재 활성 상태인 네트워크를 가져옵니다.
GetContextGuid

네트워크와 연결된 컨텍스트 식별자를 가져옵니다.
GetDeviceSignature NIC의 서명을 가져옵니다.
GetDot11AuthAlgorithm 보안 설정과 연결된 인증 알고리즘을 가져옵니다.
GetDot11CipherAlgorithm 보안 설정과 연결된 암호 알고리즘을 가져옵니다.
GetFriendlyName NIC의 이름을 가져옵니다.
GetIEnumDot11AdHocInterfaces 컴퓨터에서 사용할 수 있는 무선 NIC(네트워크 인터페이스 카드) 집합을 반환합니다.
GetIEnumDot11AdHocNetworks 이 NIC와 연결된 네트워크의 컬렉션을 가져옵니다.
GetIEnumDot11AdHocNetworks 연결 범위 내에서 사용 가능한 임시 네트워크 대상 목록을 반환합니다.
GetIEnumSecuritySettings 이 NIC와 연결된 보안 설정의 컬렉션을 가져옵니다.
GetInterface 네트워크와 연결된 인터페이스를 가져옵니다.
GetNetwork 서명과 연결된 네트워크를 반환합니다.
GetProfileName 네트워크와 연결된 프로필 이름을 가져옵니다.
GetSecuritySetting 네트워크에 대한 보안 설정을 가져옵니다.

GetSignalQuality 네트워크의 라디오와 연결된 신호 품질 값을 가져옵니다.
GetSignature 임시 네트워크와 연결된 고유 서명을 가져옵니다.
GetSSID 네트워크의 SSID를 가져옵니다.
GetStatus 이 NIC와 연결된 활성 네트워크의 연결 상태 가져옵니다.
GetStatus 네트워크의 연결 상태 가져옵니다.
HasProfile 네트워크와 연결된 저장된 프로필이 있는지 여부를 지정하는 부울 값을 반환합니다.
IsAdHocCapable NIC가 임시 네트워크 만들기 또는 사용을 지원하는지 여부를 지정합니다.
IsDot11d NIC가 802.11d 규격인지 여부를 지정합니다.
IsRadioOn 라디오가 켜지는지 여부를 지정합니다.
다음 시퀀스에서 지정된 요소 수를 가져오고 검색된 항목 수만큼 현재 위치를 이동합니다. (IEnumDot11AdHocInterfaces.Next)
다음 시퀀스에서 지정된 요소 수를 가져오고 검색된 항목 수만큼 현재 위치를 이동합니다. (IEnumDot11AdHocNetworks.Next)
다음

시퀀스에서 지정된 요소 수를 가져오고 검색된 항목 수만큼 현재 위치를 이동합니다. (IEnumDot11AdHocSecuritySettings.Next)
OnConnectFail 클라이언트에 연결 시도가 실패했음을 알 수 있습니다.
OnConnectionStatusChange 클라이언트에 NIC와 연결된 네트워크의 연결 상태 변경되었음을 알 수 있습니다.
OnInterfaceAdd 클라이언트에 새 NIC(네트워크 인터페이스 카드)가 활성화되어 있음을 알 수 있습니다.
OnInterfaceRemove 클라이언트에 NIC(네트워크 인터페이스 카드)가 비활성 상태가 되었음을 알 수 있습니다.
OnNetworkAdd 새 무선 임시 네트워크 대상이 범위에 있고 연결에 사용할 수 있음을 클라이언트에 알 수 있습니다.
OnNetworkRemove 무선 임시 네트워크 대상을 더 이상 연결할 수 없음을 클라이언트에 알깁니다.
OnStatusChange 클라이언트에 네트워크의 연결 상태 변경되었음을 알 수 있습니다.
Reset 열거형 시퀀스의 시작 부분으로 다시 설정됩니다. (IEnumDot11AdHocInterfaces.Reset)
Reset 열거형 시퀀스의 시작 부분으로 다시 설정됩니다. (IEnumDot11AdHocNetworks.Reset)
Reset 열거형 시퀀스의 시작 부분으로 다시 설정됩니다. (IEnumDot11AdHocSecuritySettings.Reset)
Skip 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다. (IEnumDot11AdHocInterfaces.Skip)
Skip 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다. (IEnumDot11AdHocNetworks.Skip)

Skip 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다. (IEnumDot11AdHocSecuritySettings.Skip)
WFD_OPEN_SESSION_COMPLETE_CALLBACK WFDStartOpenSession 작업이 완료된 경우 WFDStartOpenSession 함수에서 호출하는 콜백 함수를 정의합니다.
WFDCancelOpenSession 애플리케이션이 완료되지 않은 보류 중인 WFDStartOpenSession 함수를 취소하려고 했음을 나타냅니다.
WFDCloseHandle Wi-Fi Direct 서비스에 대한 핸들을 닫습니다.
WFDCloseSession 이전에 WFDStartOpenSession 함수를 성공적으로 호출한 후 세션을 닫습니다.
WFDOpenHandle Wi-Fi Direct 서비스에 대한 핸들을 열고 사용할 Wi-FI Direct API 버전을 협상합니다.
WFDOpenLegacySession Wi-Fi Direct 레거시 디바이스에 대해 저장된 프로필을 검색하고 적용합니다.
WFDStartOpenSession 이전에 Windows 페어링 환경을 통해 페어링된 특정 Wi-Fi Direct 디바이스에 대한 주문형 연결을 시작합니다.
WFDUpdateDeviceVisibility 지정된 Wi-Fi 직접 디바이스 노드에 대한 Wi-Fi Direct 디바이스 주소에 대한 디바이스 가시성을 업데이트.
WLAN_NOTIFICATION_CALLBACK 알림 콜백 함수의 유형을 정의합니다.
WlanAllocateMemory 메모리를 할당합니다.

WlanCloseHandle 서버에 대한 연결을 닫습니다.
WlanConnect 특정 네트워크에 연결하려고 시도합니다.
WlanDeleteProfile 로컬 컴퓨터에서 무선 인터페이스에 대한 무선 프로필을 삭제합니다.
WlanDeviceServiceCommand OEM 또는 IHV 구성 요소가 특정 무선 LAN 인터페이스에서 디바이스 서비스와 통신할 수 있도록 허용합니다.
WlanDisconnect 현재 네트워크에서 인터페이스의 연결을 끊습니다.
WlanEnumInterfaces 로컬 컴퓨터에서 현재 사용하도록 설정된 모든 무선 LAN 인터페이스를 열거합니다.
WlanExtractPsdIEDataList 비콘에 포함된 원시 IE 데이터에서 PSD(근접 서비스 검색) IE(정보 요소) 데이터 목록을 추출합니다.
WlanFreeMemory 메모리를 해제합니다.
WlanGetAvailableNetworkList 무선 LAN 인터페이스에서 사용 가능한 네트워크 목록을 검색합니다.
WlanGetFilterList 그룹 정책 또는 사용자 권한 목록을 검색합니다.
WlanGetInterfaceCapability 인터페이스의 기능을 검색합니다.
WlanGetNetworkBssList 지정된 무선 LAN 인터페이스에서 무선 네트워크 또는 네트워크의 기본 서비스 집합(BSS) 항목 목록을 검색합니다.

WlanGetProfile 지정된 무선 프로필에 대한 모든 정보를 검색합니다.
WlanGetProfileCustomUserData 무선 프로필과 연결된 사용자 지정 사용자 데이터를 가져옵니다.
WlanGetProfileList 프로필 목록을 검색합니다.
WlanGetSecuritySettings 구성 가능한 개체와 연결된 보안 설정을 가져옵니다.
WlanGetSupportedDeviceServices 지정된 무선 LAN 인터페이스에서 지원되는 디바이스 서비스 목록을 검색합니다.
WlanHostedNetworkForceStart 요청을 애플리케이션의 호출 핸들과 연결하지 않고 무선 호스트 네트워크를 wlan_hosted_network_active 상태로 전환합니다.
WlanHostedNetworkForceStop 요청을 애플리케이션의 호출 핸들과 연결하지 않고 무선 호스트 네트워크를 wlan_hosted_network_idle 전환합니다.
WlanHostedNetworkInitSettings 이러한 설정이 아직 구성되지 않은 경우 무선 호스트 네트워크에서 네트워크 연결 설정(예: SSID 및 최대 피어 수)을 구성하고 저장합니다.
WlanHostedNetworkQueryProperty 무선 Hosted Network의 현재 정적 속성을 쿼리합니다.
WlanHostedNetworkQuerySecondaryKey 무선 호스트 네트워크에서 사용하도록 구성된 보조 보안 키를 쿼리합니다.
WlanHostedNetworkQueryStatus 무선 호스트 네트워크의 현재 상태 쿼리합니다.
WlanHostedNetworkRefreshSecuritySettings

무선 호스트 네트워크 보안 설정의 구성 가능하고 자동으로 생성된 부분을 새로 고칩니다.
WlanHostedNetworkSetProperty 무선 호스트 네트워크의 정적 속성을 설정합니다.
WlanHostedNetworkSetSecondaryKey 무선 호스트 네트워크에서 사용할 보조 보안 키를 구성합니다.
WlanHostedNetworkStartUsing 무선 호스트 네트워크를 시작합니다.
WlanHostedNetworkStopUsing 무선 호스트 네트워크를 중지합니다.
WlanIhvControl WLAN 드라이버 또는 서비스의 IHV(독립 하드웨어 공급업체) 제어를 위한 메커니즘을 제공합니다.
WlanOpenHandle 서버에 대한 연결을 엽니다.
WlanQueryAutoConfigParameter 자동 구성 서비스의 매개 변수에 대한 쿼리입니다.
WlanQueryInterface WlanQueryInterface 함수는 지정된 인터페이스의 다양한 매개 변수를 쿼리합니다.
WlanReasonCodeToString 지정된 이유 코드를 설명하는 문자열을 검색합니다.
WlanRegisterDeviceServiceNotification 관리자 권한이 있는 사용자 모드 클라이언트 또는 User-Mode 드라이버 프레임워크(UMDF) 드라이버가 관심 있는 디바이스 서비스에 해당하는 원치 않는 알림을 등록할 수 있습니다.
WlanRegisterNotification 모든 무선 인터페이스에서 알림을 등록 및 등록 취소하는 데 사용됩니다.
WlanRegisterVirtualStationNotification 가상 스테이션에서 알림을 등록 및 등록 취소하는 데 사용됩니다.

WlanRenameProfile 지정된 프로필의 이름을 바꿉니다.
WlanSaveTemporaryProfile 프로필 저장소에 임시 프로필을 저장합니다.
WlanScan 표시된 인터페이스에서 사용 가능한 네트워크에 대한 검사를 요청합니다.
WlanSetAutoConfigParameter 자동 구성 서비스에 대한 매개 변수를 설정합니다.
WlanSetFilterList 허용/거부 목록을 설정합니다.
WlanSetInterface 사용자 구성 가능한 매개 변수를 설정합니다.
WlanSetProfile 특정 프로필의 콘텐츠를 설정합니다.
WlanSetProfileCustomUserData 프로필과 연결된 사용자 지정 사용자 데이터를 설정합니다.
WlanSetProfileEapUserData 원시 EAP 데이터에 지정된 대로 EAP(Extensible Authentication Protocol) 사용자 자격 증명을 설정합니다.
WlanSetProfileEapXmlUserData XML 문자열에 지정된 대로 EAP(Extensible Authentication Protocol) 사용자 자격 증명을 설정합니다.
WlanSetProfileList 프로필의 기본 설정 순서를 설정합니다.
WlanSetProfilePosition 기본 설정 목록에서 지정된 단일 프로필의 위치를 설정합니다.

WlanSetPsdIEDataList PSD(근접 서비스 검색) IE(정보 요소) 데이터 목록을 설정합니다.
WlanSetSecuritySettings 구성 가능한 개체의 보안 설정을 지정합니다.
WlanUIEditProfile 무선 프로필 UI(사용자 인터페이스)를 표시합니다.

인터페이스

IDot11AdHocInterface NIC(무선 네트워크 인터페이스 카드)를 나타냅니다.
IDot11AdHocInterfaceNotificationSink IDot11AdHocInterface에서 지원하는 알림을 정의합니다.
IDot11AdHocManager 802.11 임시 네트워크를 만들고 관리합니다.
IDot11AdHocManagerNotificationSink IDot11AdHocManager 인터페이스에서 지원하는 알림을 정의합니다.
IDot11AdHocNetwork 연결 범위 내에서 사용 가능한 임시 네트워크 대상을 나타냅니다.
IDot11AdHocNetworkNotificationSink IDot11AdHocNetwork 인터페이스에서 지원하는 알림을 정의합니다.
IDot11AdHocSecuritySettings 무선 임시 네트워크에 대한 보안 설정을 지정합니다.
IEnumDot11AdHocInterfaces 현재 표시되는 802.11 임시 네트워크 인터페이스의 컬렉션을 나타냅니다.

IEnumDot11AdHocNetworks 현재 표시되는 802.11 임시 네트워크의 컬렉션을 나타냅니다.
IEnumDot11AdHocSecuritySettings 표시되는 각 무선 임시 네트워크와 연결된 보안 설정의 컬렉션을 나타냅니다.

구조체

DOT11_NETWORK 사용 가능한 무선 네트워크에 대한 정보를 포함합니다. (DOT11_NETWORK)
DOT11_NETWORK_LIST 802.11 무선 네트워크 목록을 포함합니다.
ONEX_AUTH_PARAMS 802.1X 인증에 사용되는 802.1X 인증 매개 변수를 포함합니다.
ONEX_EAP_ERROR 802.1X 인증으로 오류가 발생할 경우 802.1X EAP 오류가 포함됩니다.
ONEX_RESULT_UPDATE_DATA 802.1X 인증에 대한 상태 변경 내용에 대한 정보를 포함합니다.
ONEX_STATUS 현재 802.1X 인증 상태 포함합니다.
ONEX_VARIABLE_BLOB 가변 크기 멤버를 포함하기 위해 다른 802.1X 인증 구조의 멤버로 사용됩니다.
WLAN_ASSOCIATION_ATTRIBUTES 연결에 대한 연결 특성을 포함합니다.
WLAN_AUTH_CIPHER_PAIR_LIST 인증 및 암호 알고리즘 쌍 목록을 포함합니다.

WLAN_AVAILABLE_NETWORK 사용 가능한 무선 네트워크에 대한 정보를 포함합니다. (WLAN_AVAILABLE_NETWORK)
WLAN_AVAILABLE_NETWORK_LIST 사용 가능한 네트워크에 대한 정보 배열을 포함합니다.
WLAN_BSS_ENTRY BSS(기본 서비스 집합)에 대한 정보를 포함합니다.
WLAN_BSS_LIST BSS(기본 서비스 집합) 항목 목록을 포함합니다.
WLAN_CONNECTION_ATTRIBUTES 무선 연결의 특성을 정의합니다.
WLAN_CONNECTION_NOTIFICATION_DATA 연결 관련 알림에 대한 정보를 포함합니다.
WLAN_CONNECTION_PARAMETERS WlanConnect 함수를 사용할 때 사용되는 매개 변수를 지정합니다.
WLAN_COUNTRY_OR_REGION_STRING_LIST 지원되는 국가 또는 지역 문자열 목록을 포함합니다.
WLAN_DEVICE_SERVICE_GUID_LIST 디바이스 서비스 GUID의 배열을 포함합니다.
WLAN_DEVICE_SERVICE_NOTIFICATION_DATA 디바이스 서비스 알림을 나타내는 구조체입니다.
WLAN_HOSTED_NETWORK_CONNECTION_SETTINGS 무선 호스트 네트워크의 연결 설정에 대한 정보를 포함합니다.
WLAN_HOSTED_NETWORK_DATA_PEER_STATE_CHANGE 무선 호스티드 네트워크의 데이터 피어에 대한 네트워크 상태 변경에 대한 정보를 포함합니다.
WLAN_HOSTED_NETWORK_PEER_STATE

무선 호스트 네트워크의 피어에 대한 피어 상태에 대한 정보를 포함합니다.
WLAN_HOSTED_NETWORK_RADIO_STATE 무선 호스트 네트워크의 라디오 상태에 대한 정보를 포함합니다.
WLAN_HOSTED_NETWORK_SECURITY_SETTINGS 무선 호스트 네트워크의 보안 설정에 대한 정보를 포함합니다.
WLAN_HOSTED_NETWORK_STATE_CHANGE 무선 호스트 네트워크의 네트워크 상태 변경에 대한 정보를 포함합니다.
WLAN_HOSTED_NETWORK_STATUS 무선 호스트 네트워크의 상태 대한 정보를 포함합니다.
WLAN_INTERFACE_CAPABILITY 인터페이스의 기능에 대한 정보를 포함합니다.
WLAN_INTERFACE_INFO 무선 LAN 인터페이스에 대한 정보를 포함합니다.
WLAN_INTERFACE_INFO_LIST NIC 인터페이스 정보의 배열입니다.
WLAN_MAC_FRAME_STATISTICS 전송 및 수신된 MAC 프레임에 대한 정보를 포함합니다.
WLAN_MSM_NOTIFICATION_DATA MSM(미디어 관련 모듈) 연결 관련 알림에 대한 정보를 포함합니다.
WLAN_PHY_FRAME_STATISTICS 전송 및 수신된 PHY 프레임에 대한 정보를 포함합니다.
WLAN_PHY_RADIO_STATE 라디오 상태를 지정합니다.
WLAN_PROFILE_INFO 프로필에 대한 기본 정보입니다.

WLAN_PROFILE_INFO_LIST 무선 프로필 정보 목록을 포함합니다.
WLAN_RADIO_STATE PHY(물리적 계층) 형식 목록에서 라디오 상태를 지정합니다.
WLAN_RATE_SET 지원되는 데이터 속도 집합입니다.
WLAN_RAW_DATA 일부 네이티브 Wifi 함수에서 사용되는 Blob 형식의 원시 데이터를 포함합니다.
WLAN_RAW_DATA_LIST 일부 네이티브 Wifi 함수에서 사용되는 데이터 Blob 배열 형식의 원시 데이터를 포함합니다.
WLAN_SECURITY_ATTRIBUTES 무선 연결에 대한 보안 특성을 정의합니다.
WLAN_STATISTICS 인터페이스에 대한 다양한 통계입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

NetBIOS

아티클2023. 08. 22.

NetBIOS 기술 개요입니다.

NetBIOS를 개발하려면 다음 헤더가 필요합니다.

- nb30.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Netbios

Functions

Netbios Netbios 함수는 지정된 NCB(네트워크 제어 블록)를 해석하고 실행합니다.

구조체

ACTION_HEADER ACTION_HEADER 구조에는 작업에 대한 정보가 포함됩니다. 이 작업은 표준 전송 인터페이스의 확장입니다.
ADAPTER_STATUS ADAPTER_STATUS 구조에는 네트워크 어댑터에 대한 정보가 포함됩니다.
FIND_NAME_BUFFER FIND_NAME_BUFFER 구조에는 로컬 네트워크 세션에 대한 정보가 포함됩니다.
FIND_NAME_HEADER FIND_NAME_HEADER 구조에는 네트워크 이름에 대한 정보가 포함됩니다. 이 구조체 뒤에는 이름을 설명하는 데 필요한 만큼의 FIND_NAME_BUFFER 구조가 잇습니다.
LANA_ENUM

LANA_ENUM 구조체에는 현재 LAN 어댑터의 숫자가 포함됩니다.
NAME_BUFFER NAME_BUFFER 구조에는 로컬 네트워크 이름에 대한 정보가 포함됩니다. 하나 이상의 NAME_BUFFER 구조체는 애플리케이션이 NCB 구조체의 ncb_command 멤버에서 NCBASTAT 명령을 지정하는 경우 ADAPTER_STATUS 구조체를 따릅니다.
Ncb NCB 구조체는 네트워크 제어 블록을 나타냅니다.
SESSION_BUFFER SESSION_BUFFER 구조에는 로컬 네트워크 세션에 대한 정보가 포함됩니다. 하나 이상의 SESSION_BUFFER 구조체는 애플리케이션이 NCB 구조체의 ncb_command 멤버에서 NCBSSTAT 명령을 지정하는 경우 SESSION_HEADER 구조체를 따릅니다.
SESSION_HEADER SESSION_HEADER 구조에는 네트워크 세션에 대한 정보가 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Netshell

아티클2023. 08. 22.

NetSh 기술 개요입니다.

NetSh를 개발하려면 다음 헤더가 필요합니다.

- netsh.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Netsh

열거형

NS_CMD_FLAGS NetShell에서 사용할 수 있는 명령 플래그를 지정합니다.
NS_REQS 이벤트 수를 지정합니다.

Functions

FN_HANDLE_CMD 도우미에 대한 명령 함수입니다.
MatchEnumTag 법적 값 테이블을 검색하여 특정 토큰과 일치하는 값을 찾습니다.
MatchToken 사용자가 입력한 문자열이 특정 문자열과 일치하는지 여부를 결정합니다.
NS_CONTEXT_COMMIT_FN 도우미에 대한 커밋 함수입니다.

NS_CONTEXT_CONNECT_FN 도우미를 위한 연결 함수입니다.
NS_CONTEXT_DUMP_FN 도우미에 대한 덤프 함수입니다.
NS_DLL_INIT_FN 도우미의 초기 로드를 수행하기 위해 NetShell에서 호출됩니다.
NS_DLL_STOP_FN 도우미 DLL에 대한 DLL 중지 함수입니다.
NS_HELPER_START_FN 도우미의 시작 함수입니다.
NS_HELPER_STOP_FN 도우미에 대한 중지 함수입니다.
NS_OSVERSIONCHECK 도우미를 위한 운영 체제 검사 함수입니다.
PreprocessCommand 인수 문자열을 구문 분석하고 필요한 모든 태그가 있는지 확인합니다.
PrintError NetShell 콘솔에 시스템 또는 애플리케이션 오류 메시지를 표시합니다.
PrintMessage NetShell 콘솔에 출력을 표시합니다.
PrintMessageFromModule NetShell 콘솔에 지역화된 출력을 표시합니다.
RegisterContext NetShell에 도우미 컨텍스트를 등록합니다.
RegisterHelper

도우미의 노출된 InitHelperDll 함수 내에서 호출되고 NetShell 컨텍스트에 도우미를 등록합니다.

구조체

CMD_ENTRY 도우미 명령을 정의합니다.
CMD_GROUP_ENTRY 도우미 명령 그룹을 정의합니다.
NS_CONTEXT_ATTRIBUTES 컨텍스트의 특성을 정의합니다.
NS_HELPER_ATTRIBUTES 도우미의 특성을 제공합니다.
TAG_TYPE PreprocessCommand 함수에 사용되는 태그를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

네트워크 액세스 보호

아티클2023. 03. 09.

네트워크 액세스 보호 기술 개요입니다.

네트워크 액세스 보호를 개발하려면 다음 헤더가 필요합니다.

- naptypes.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 네트워크 액세스 보호
- 네트워크 액세스 보호 WMI 공급자

열거형

ExtendedIsolationState 연결의 확장된 격리 상태를 설명합니다.
FailureCategory 오류의 원인을 나타냅니다.
FixupState SHA(시스템 상태 에이전트)의 수정 상태를 정의합니다.
IsolationState 연결의 격리 상태를 설명합니다.
NapNotifyType NapAgent 서비스에서 보낸 서비스 알림 유형을 열거합니다.
RemoteConfigurationType 구성 요소에 대해 가능한 원격 구성의 유형을 설명합니다.

구조체

CorrelationId SoHResponses와 SoHRequests를 페어링하는 데 사용되며 SoH 교환을 고유하게 설명합니다.
CountedString 정의된 길이로 null로 끝나는 문자열을 정의합니다.
FailureCategoryMapping SHV(시스템 상태 유효성 검사기)에서 반환되는 선택적 준수 상태 정보를 포함합니다.
FixupInfo Sysytem Health Agent(SHA)에 대한 수정 정보를 포함합니다.
Ipv4Address IPv4 주소를 정의합니다.
Ipv6Address IPv6 주소를 정의합니다.
IsolationInfo 컴퓨터 또는 연결의 격리 상태를 정의합니다.
IsolationInfoEx 컴퓨터 또는 연결의 확장된 격리 상태를 정의합니다.
NapComponentRegistrationInfo SHA, SHV 또는 적용 클라이언트와 같은 등록된 NAP 구성 요소를 정의합니다.
NetworkSoH 유선 SoH 프로토콜을 정의합니다.
PrivateData 불투명 데이터 Blob을 저장하고 검색하는 데 사용됩니다.
ResultCodes 결과 코드 목록을 정의합니다.

Soh SoH(Health 문) 데이터를 포함합니다.
SoHAttribute SHA/SHV와 NAP 시스템 간의 SoH 프로토콜을 정의합니다.
SystemHealthAgentState SHA의 동적 상태를 저장합니다.

네트워크 진단 프레임워크

아티클2023. 08. 22.

네트워크 진단 프레임워크 기술 개요입니다.

네트워크 진단 프레임워크를 개발하려면 다음 헤더가 필요합니다.

- ndattrib.h
- ndfapi.h
- ndhelper.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 네트워크 진단 프레임워크

열거형

ATTRIBUTE_TYPE ATTRIBUTE_TYPE 열거형은 도우미 특성에 대해 가능한 값을 정의합니다.
DIAGNOSIS_STATUS DIAGNOSIS_STATUS 열거형은 구성 요소의 상태가 결정된 도우미 클래스에 제출된 가설의 결과를 설명합니다.
PROBLEM_TYPE PROBLEM_TYPE 열거형은 도우미 클래스가 나타내는 문제의 유형을 설명합니다.
REPAIR_RISK REPAIR_RISK 열거형은 복구 변경 내용이 영구적인지 여부와 취소할 수 있는지 여부를 지정합니다.
REPAIR_SCOPE REPAIR_SCOPE 열거형은 지정된 복구에 대한 수정 scope 설명합니다.
REPAIR_STATUS REPAIR_STATUS 열거형은 도우미 클래스가 복구 옵션을 시도한 결과를 설명합니다.
UI_INFO_TYPE

UI_INFO_TYPE 열거형은 사용자 인터페이스 작업을 수행하는 복구를 식별합니다.

Functions

취소 진행 중인 진단 또는 복구를 취소합니다.
정리 도우미 클래스 확장에서 진단 또는 복구 작업 후 리소스를 클린 수 있습니다.
CreateUtilityInstance NDF(네트워크 진단 프레임워크)에서 사용됩니다. 이 메서드는 시스템 사용을 위해 예약되어 있습니다. (INetDiagHelperUtilFactory.CreateUtilityInstance)
GetAttributeInfo GetAttributeInfo 메서드는 도우미 클래스 확장에 필요한 주요 매개 변수 목록을 검색합니다.
GetAttributes 도우미 클래스 확장에서 진단한 문제에 대한 추가 정보를 검색합니다.
GetCacheTime 진단 및 복구 작업의 캐시된 결과가 만료된 시간을 지정합니다.
GetDiagnosticsInfo 도우미 클래스 확장 instance 예상값을 제공할 수 있습니다.
GetDownStreamHypotheses 도우미 클래스 확장에 가설을 생성하도록 요청합니다.
GetHigherHypotheses 높은 사용률의 가능한 원인에 대한 가설을 생성합니다.
GetKeyAttributes 도우미 클래스 확장의 주요 특성을 검색합니다.

GetLifeTime 도우미 클래스 확장 instance 수명을 검색합니다.
GetLowerHypotheses 로컬 구성 요소에서 낮은 상태의 가능한 원인에 대한 가설을 생성합니다.
GetRepairInfo 지정된 문제 유형에 대해 도우미 클래스 확장에 있는 복구 정보를 검색합니다.
GetUpStreamHypotheses 업스트림 네트워크 구성 요소에서 높은 사용률의 가능한 원인에 대한 가설을 생성합니다.
HighUtilization 해당 구성 요소가 높은 사용률인지 확인합니다.
초기화 Initialize 메서드는 가설에서 도우미 클래스 확장에 특성을 전달합니다. 도우미 클래스는 기본 진단 함수에 사용할 이러한 매개 변수를 저장해야 합니다. 이 메서드는 진단 함수 앞에 호출해야 합니다.
LowHealth 진단 중인 구성 요소가 정상인지 확인합니다.
NdfCancelIncident 기존 인시던트에서 이전에 호출된 불필요한 함수를 취소하는 데 사용됩니다.
NdfCloseIncident 해결 후 NDF(네트워크 진단 프레임워크) 인시던트를 닫는 데 사용됩니다.
NdfCreateConnectivityIncident 일반적인 인터넷 연결 문제를 진단합니다.
NdfCreateDNSIncident 특정 호스트 이름 확인에서 이름 확인 문제를 진단합니다.
NdfCreateGroupingIncident 피어 투 피어 그룹화 기능 문제를 진단하는 세션을 만듭니다.

NdfCreateInboundIncident 특정 애플리케이션 또는 서비스에 대한 인바운드 연결을 진단하는 세션을 만듭니다.
NdfCreateIncident 애플리케이션에 통합된 NDF 기능을 테스트합니다.
NdfCreateNetConnectionIncident NetConnection 도우미 클래스를 사용하여 연결 문제를 진단합니다.
NdfCreatePnrpIncident PNRP(피어 이름 확인 프로토콜) 서비스 문제를 진단하는 세션을 만듭니다.
NdfCreateSharingIncident 특정 네트워크 공유 액세스의 네트워크 문제를 진단합니다.
NdfCreateWebIncident 웹 연결 문제를 진단합니다. (NdfCreateWebIncident)
NdfCreateWebIncidentEx 웹 연결 문제를 진단합니다. (NdfCreateWebIncidentEx)
NdfCreateWinSockIncident Microsoft에서 제공하는 Winsock 도우미 클래스에 대한 액세스를 제공합니다.
NdfDiagnoseIncident 사용자 인터페이스를 표시하지 않고 인시던트 근본 원인을 진단합니다.
NdfExecuteDiagnosis NdfExecuteDiagnosis 함수는 발생한 인시던트 근본 원인을 진단하는 데 사용됩니다.
NdfGetTraceFile 진단 세션에서 ETW(Windows용 이벤트 추적) 이벤트가 포함된 ETL(이벤트 추적 로그) 파일이 포함된 경로를 검색하는 데 사용됩니다.
NdfRepairIncident 사용자 인터페이스를 표시하지 않고 인시던트를 복구합니다.

ReconfirmLowHealth 가설이 진단된 후 복구가 검색되기 전에 두 번째 Low Health 패스를 추가하는 데 사용됩니다.
복구 입력 매개 변수로 지정된 복구를 수행합니다.
ReproduceFailure NDF(네트워크 진단 프레임워크)에서 사용됩니다. 이 메서드는 시스템 사용을 위해 예약되어 있습니다. (INetDiagHelperEx.ReproduceFailure)
SetLifeTime 도우미 클래스 확장은 진단을 해당 기간 내의 이벤트로 제한할 수 있습니다.
SetUtilities NDF(네트워크 진단 프레임워크)에서 사용됩니다. 이 메서드는 시스템 사용을 위해 예약되어 있습니다. (INetDiagHelperEx.SetUtilities)
유효성 검사 복구가 성공적으로 완료된 후 NDF에서 호출됩니다.

인터페이스

INetDiagHelper INetDiagHelper 인터페이스는 네트워크 관련 문제의 진단 및 해결과 관련된 정보를 캡처하고 제공하는 메서드를 제공합니다.
INetDiagHelperEx INetDiagHelper 인터페이스에서 확장하여 네트워크 관련 문제 진단 및 해결과 관련된 정보를 캡처하고 제공하는 메서드를 제공합니다.
INetDiagHelperInfo INetDiagHelperInfo 인터페이스는 도우미 클래스에 필요한 정보가 있고 올바른 도우미 클래스를 선택했는지 확인해야 하는 경우 NDF(네트워크 진단 프레임워크)에서 호출되는 메서드를 제공합니다.

INetDiagHelperUtilFactory NDF(네트워크 진단 프레임워크)에서 사용하는 예약된 메서드를 제공합니다.

구조체

DIAG_SOCKADDR Windows 소켓 통신에 참여하는 컴퓨터의 IP(인터넷 프로토콜) 주소를 저장합니다.
DiagnosticsInfo DiagnosticsInfo 구조에는 진단 시간의 추정치와 호출 플래그가 포함됩니다.
HELPER_ATTRIBUTE HELPER_ATTRIBUTE 구조에는 지원되는 모든 NDF 데이터 형식이 포함됩니다.
HelperAttributeInfo HelperAttributeInfo 구조체에는 도우미 특성의 이름과 해당 형식이 포함됩니다.
가설 HYPOTHESIS 구조에는 다른 도우미 클래스에 대한 가설을 NDF에 제출하는 데 사용되는 데이터가 포함되어 있습니다.
HypothesisResult 도우미 클래스에서 반환된 가설에 대한 정보를 포함합니다.
LIFE_TIME LIFE_TIME 구조에는 시작 시간과 종료 시간이 포함됩니다.
OCTET_STRING OCTET_STRING 구조체에는 바이트 데이터 문자열에 대한 포인터가 포함되어 있습니다.
RepairInfo RepairInfo 구조에는 특정 복구 옵션에 필요한 데이터가 포함되어 있습니다.
RepairInfoEx

인시던트 근본 원인을 resolve 데 사용할 수 있는 자세한 복구 정보가 포함되어 있습니다.
RootCauseInfo 인시던트 근본 원인에 대한 자세한 정보를 포함합니다.
ShellCommandInfo ShellCommandInfo 구조에는 수동 복구 옵션을 위해 추가 애플리케이션을 시작하는 데 필요한 데이터가 포함되어 있습니다.
UiInfo UiInfo 구조체는 복구 메시지를 사용자에게 표시하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Windows Vista 이전의 네트워크 드라이버

아티클2023. 08. 22.

Windows Vista 이전의 네트워크 드라이버 기술 개요입니다.

Windows Vista 이전의 네트워크 드라이버 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows Vista 이전의 네트워크 드라이버

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

네트워크 드라이버, Windows Vista 이상

아티클2023. 03. 09.

네트워크 드라이버, Windows Vista 및 이후 기술 개요입니다.

네트워크 드라이버, Windows Vista 이상을 개발하려면 다음 헤더가 필요합니다.

- fwpstypes.h
- l2cmn.h
- ndkinfo.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 네트워크 드라이버, Windows Vista 이상

열거형

FWPS_DISCARD_MODULE0 FWPS_DISCARD_MODULE0 열거형 형식은 데이터를 삭제한 모듈의 형식을 지정합니다. FWPS_DISCARD_MODULE0 특정 버전의 FWPS_DISCARD_MODULE.
MIB_IF_TABLE_LEVEL MIB_IF_TABLE_LEVEL 열거형 형식은 검색할 인터페이스 정보의 수준을 정의합니다.
NET_IF_ACCESS_TYPE NET_IF_ACCESS_TYPE 열거형 형식은 NDIS 네트워크 인터페이스 액세스 형식을 지정합니다.
NET_IF_ADMIN_STATUS NET_IF_ADMIN_STATUS 열거형 유형은 RFC 2863에 설명된 대로 NDIS 네트워크 인터페이스 관리 상태를 지정합니다.
NET_IF_CONNECTION_TYPE NET_IF_CONNECTION_TYPE 열거형 형식은 NDIS 네트워크 인터페이스 연결 형식을 지정합니다.
NET_IF_DIRECTION_TYPE NET_IF_ACCESS_TYPE 열거형 형식은 NDIS 네트워크 인터페이스 방향 형식을 지정합니다.

NET_IF_MEDIA_CONNECT_STATE NET_IF_MEDIA_CONNECT_STATE 열거형 형식은 NDIS 네트워크 인터페이스 연결 상태를 지정합니다.
NET_IF_MEDIA_DUPLEX_STATE NET_IF_MEDIA_DUPLEX_STATE 열거형 형식은 NDIS 네트워크 인터페이스 이중 상태를 지정합니다.
NET_IF_OPER_STATUS NET_IF_OPER_STATUS 열거형 형식은 현재 NDIS 네트워크 인터페이스 작동 상태를 정의합니다.
NL_ADDRESS_TYPE NL_ADDRESS_TYPE 열거형 유형은 네트워크 계층의 IP 주소 유형을 지정합니다.
NL_LINK_LOCAL_ADDRESS_BEHAVIOR NL_LINK_LOCAL_ADDRESS_BEHAVIOR 열거형 형식은 링크 로컬 주소 동작을 정의합니다.
NL_NEIGHBOR_STATE NL_NEIGHBOR_STATE 열거형 형식은 RFC 2461, 섹션 7.3.2에 설명된 대로 네트워크 계층 인접 IP 주소의 상태를 정의합니다.
NL_ROUTE_ORIGIN NL_ROUTE_ORIGIN 열거형 형식은 IP 경로의 원본을 정의합니다.
NL_ROUTE_PROTOCOL NL_ROUTE_PROTOCOL 열거형 형식은 RFC 4292에 설명된 대로 IP 경로가 추가된 라우팅 메커니즘을 정의합니다.
NL_ROUTER_DISCOVERY_BEHAVIOR NL_ROUTER_DISCOVERY_BEHAVIOR 열거형 형식은 RFC 2461에 설명된 대로 라우터 검색 동작을 정의합니다.
TUNNEL_TYPE TUNNEL_TYPE 열거형 형식은 IANA(Internet Assigned Names Authority)에 설명된 대로 터널에서 사용하는 캡슐화 메서드를 정의합니다.

Functions

GetCurrentThreadCompartmentId 다음에 사용하도록 예약됩니다. 이 함수를 사용하지 마세요. (GetCurrentThreadCompartmentId)
GetNetworkInformation 다음에 사용하도록 예약됩니다. 이 함수를 사용하지 마세요. (GetNetworkInformation)
GetSessionCompartmentId 다음에 사용하도록 예약됩니다. 이 함수를 사용하지 마세요. (GetSessionCompartmentId)
SetCurrentThreadCompartmentId 다음에 사용하도록 예약됩니다. 이 함수를 사용하지 마세요. (SetCurrentThreadCompartmentId)
SetNetworkInformation 다음에 사용하도록 예약됩니다. 이 함수를 사용하지 마세요. (SetNetworkInformation)
SetSessionCompartmentId 다음에 사용하도록 예약됩니다. 이 함수를 사용하지 마세요. (SetSessionCompartmentId)

구조체

FWPS_ACTION0 FWPS_ACTION0 구조는 필터의 모든 필터링 조건이 true인 경우 필터 엔진이 수행하는 런타임 작업을 지정합니다. FWPS_ACTION0 특정 버전의 FWPS_ACTION.
FWPS_ALE_ENDPOINT_ENUM_TEMPLATE0 FWPS_ALE_ENDPOINT_ENUM_TEMPLATE0 구조는 ALE(애플리케이션 계층 적용) 엔드포인트를 열거할 템플릿을 지정합니다. FWPS_ALE_ENDPOINT_ENUM_TEMPLATE0 특정 버전의 FWPS_ALE_ENDPOINT_ENUM_TEMPLATE.
FWPS_ALE_ENDPOINT_PROPERTIES0 FWPS_ALE_ENDPOINT_PROPERTIES0 구조는 ALE(애플리케이션 계층 적용) 엔드포인트의 속성을 지정합니다. FWPS_ALE_ENDPOINT_PROPERTIES0 특정 버전의 FWPS_ALE_ENDPOINT_PROPERTIES.

FWPS_CLASSIFY_OUT0 FWPS_CLASSIFY_OUT0 구조체는 설명선의 classifyFn 설명선 함수의 호출자에게 반환되는 데이터를 정의합니다. FWPS_CLASSIFY_OUT0 특정 버전의 FWPS_CLASSIFY_OUT.
FWPS_DISCARD_METADATA0 FWPS_DISCARD_METADATA0 구조는 필터 엔진, 네트워크 계층 또는 전송 계층에서 삭제된 데이터를 설명합니다. FWPS_DISCARD_METADATA0 특정 버전의 FWPS_DISCARD_METADATA.
FWPS_FILTER_CONDITION0 FWPS_FILTER_CONDITION0 구조체는 필터에 대한 런타임 필터링 조건을 정의합니다. FWPS_FILTER_CONDITION0 특정 버전의 FWPS_FILTER_CONDITION.
FWPS_FILTER0 FWPS_FILTER0 구조체는 필터 엔진에서 런타임 필터를 정의합니다. 참고 FWPS_FILTER0 Windows Vista 이상에서 사용되는 특정 버전의 FWPS_FILTER.
FWPS_FILTER1 FWPS_FILTER1 구조체는 필터 엔진에서 런타임 필터를 정의합니다. 참고 FWPS_FILTER1 Windows 7 이상에서 사용되는 특정 버전의 FWPS_FILTER.
FWPS_FILTER2 FWPS_FILTER2 구조체는 필터 엔진에서 런타임 필터를 정의합니다. FWPS_FILTER2 Windows 8 이상에서 사용되는 특정 버전의 FWPS_FILTER.
FWPS_INBOUND_FRAGMENT_METADATA0 FWPS_INBOUND_FRAGMENT_METADATA0 구조는 수신된 패킷 조각에 대한 조각 데이터를 설명합니다. FWPS_INBOUND_FRAGMENT_METADATA0 특정 버전의 FWPS_INBOUND_FRAGMENT_METADATA.
FWPS_INCOMING_VALUE0 FWPS_INCOMING_VALUE0 구조는 개별 데이터 값을 정의합니다. FWPS_INCOMING_VALUE0 특정 버전의 FWPS_INCOMING_VALUE.
FWPS_INCOMING_VALUES0 FWPS_INCOMING_VALUES0 구조체는 필터 엔진이 설명선의 classifyFn 설명선 함수에 전달하는 데이터 값을 정의합니다. FWPS_INCOMING_VALUES0 특정 버전의 FWPS_INCOMING_VALUES.
IF_COUNTED_STRING_LH IF_COUNTED_STRING 구조체는 NDIS 인터페이스에 대해 계산된 문자열을 지정합니다.

IN6_ADDR IN6_ADDR 구조체는 IPv6 전송 주소를 지정합니다.
L2_NOTIFICATION_DATA 중요 네이티브 802.11 무선 LAN 인터페이스는 Windows 10 이상에서 사용되지 않습니다.
NDIS_INTERFACE_INFORMATION NDIS_INTERFACE_INFORMATION 구조는 OID_GEN_INTERFACE_INFO OID에 대한 네트워크 인터페이스에 대한 정보를 제공합니다.
NDK_ADAPTER_INFO NDK_ADAPTER_INFO 구조는 NDK 어댑터의 제한 및 기능에 대한 정보를 지정합니다.
NDK_VERSION NDK_VERSION 구조체는 NDK 인터페이스에서 주 버전과 부 버전을 지정합니다.
NET_PHYSICAL_LOCATION_LH NET_PHYSICAL_LOCATION 구조는 등록된 네트워크 인터페이스의 물리적 위치에 대한 정보를 NDIS에 제공합니다.
SOCKADDR SOCKADDR 구조체는 전송 주소를 지정하는 제네릭 구조체입니다.
SOCKADDR_IN SOCKADDR_IN 구조는 AF_INET 주소 패밀리의 전송 주소와 포트를 지정합니다.
SOCKADDR_IN6_LH SOCKADDR_IN6_LH(ws2ipdef.h) 구조체는 AF_INET6 주소 패밀리의 전송 주소와 포트를 지정합니다.
SOCKADDR_IN6_W2KSP1 SOCKADDR_IN6_W2KSP1(ws2ipdef.h) 구조체는 AF_INET6 주소 패밀리의 전송 주소와 포트를 지정합니다.
SOCKADDR_STORAGE_LH SOCKADDR_STORAGE 구조체는 전송 주소를 지정하는 제네릭 구조체입니다. (SOCKADDR_STORAGE_LH)

SOCKADDR_STORAGE_XP SOCKADDR_STORAGE 구조체는 전송 주소를 지정하는 제네릭 구조체입니다. S
SOCKET_ADDRESS_LIST SOCKET_ADDRESS_LIST 구조체는 가변 크기의 전송 주소 목록을 정의합니다.
WSACMSGHDR CMSGHDR 구조체는 데이터그램과 연결된 컨트롤 데이터 개체의 헤더를 정의합니다.

네트워크 목록 관리자

아티클2023. 03. 16.

네트워크 목록 관리자 기술 개요입니다.

네트워크 목록 관리자를 개발하려면 다음 헤더가 필요합니다.

- netlistmgr.h
- ondemandconnroutehelper.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 네트워크 목록 관리자

열거형

NLM_CONNECTION_COST NLM_CONNECTION_COST 열거형은 Windows 8 Cost API에서 지원되는 비용 수준 및 비용 플래그 집합을 지정합니다.
NLM_CONNECTION_PROPERTY_CHANGE NLM_CONNECTION PROPERTY_CHANGE 열거형은 네트워크 연결의 속성에 대한 변경 내용을 정의하는 플래그 집합입니다.
NLM_CONNECTIVITY NLM_Connectivity 열거형은 연결 관련 매개 변수가 변경 될 때마다 알림을 제공하는 플래그 집합입니다.
NLM_DOMAIN_AUTHENTICATION_KIND 도메인 인증 방법을 지정하는 상수를 정의합니다.
NLM_DOMAIN_TYPE NLM_DOMAIN_TYPE 열거형은 네트워크의 도메인 유형을 지정하는 플래그 집합입니다.
NLM_ENUM_NETWORK NLM_ENUM_NETWORK 열거형에는 열거되는 네트워크 유형을 지정하는 플래그 집합이 포함되어 있습니다.

NLM_INTERNET_CONNECTIVITY NLM_INTERNET_CONNECTIVITY 열거형은 IPv4 또는 IPv6 네트워크 연결에 대한 추가 데이터를 제공하는 플래그 집합을 정의합니다.
NLM_NETWORK_CATEGORY NLM_NETWORK_CATEGORY 열거형은 네트워크의 범주 유형을 지정하는 플래그 집합입니다.
NLM_NETWORK_CLASS NLM_NETWORK_CLASS 열거형은 네트워크가 식별되었는지를 지정하는 플래그 집합을 정의합니다.
NLM_NETWORK_PROPERTY_CHANGE NLM_NETWORK_PROPERTY_CHANGE 열거형은 네트워크의 속성에 대한 변경 내용을 정의하는 플래그 집합입니다.

Functions

ClearSimulatedProfileInfo SetSimulatedProfileInfo에서 이전에 인터넷 연결 프로필에 적용한 연결 프로필 값을 지웁니다. GetInternetConnectionProfile을 통해 다음 인터넷 연결 쿼리는 시스템 정보를 사용합니다.
복제 Clone 메서드는 현재 사용 중인 열거자와 동일한 열거형 상태를 포함하는 열거자를 만듭니다. (IEnumNetworkConnections.Clone)
복제 Clone 메서드는 현재 사용 중인 열거자와 동일한 열거형 상태를 포함하는 열거자를 만듭니다. (IEnumNetworks.Clone)
ConnectionCostChanged ConnectionCostChanged 메서드는 연결에 대한 네트워크 비용 변경의 애플리케이션을 알 수 있습니다.

ConnectionDataPlanStatusChanged ConnectionDataPlanStatusChanged 메서드는 애플리케이션에 연결 시 데이터 계획 상태 변경을 알 수 있습니다.
ConnectivityChanged NetworkConnectivityChanged 메서드는 네트워크 연결 관련 변경이 발생할 때 호출됩니다. (INetworkListManagerEvents.ConnectivityChanged)
CostChanged CostChanged 메서드는 컴퓨터 전체 인터넷 연결에 대한 비용 변경 또는 연결에서 특정 대상으로 라우팅하는 첫 번째 홉을 나타내기 위해 호출됩니다.
DataPlanStatusChanged DataPlanStatusChanged 메서드는 컴퓨터 전체 인터넷 연결에 사용되는 연결 또는 연결의 특정 대상으로 라우팅하는 첫 번째 홉과 연결된 데이터 계획의 상태 변경을 나타내기 위해 호출됩니다.
FreeInterfaceContextTable 이 함수는 GetInterfaceContextTableForHostName 함수를 사용하여 검색된 인터페이스 컨텍스트 테이블을 해제합니다.
get__NewEnum get_NewEnum 속성은 IEnumNetworkConnections 컬렉션을 반복하는 데 사용할 수 있는 자동화 열거자 개체를 반환합니다.
get__NewEnum get_NewEnum 속성은 IEnumNetworks 컬렉션을 반복하는 데 사용할 수 있는 자동화 열거자 개체를 반환합니다.
get_IsConnected get_IsConnected 속성은 네트워크에 네트워크 연결이 있는지를 지정합니다.
get_IsConnected get_IsConnected 속성은 연결된 네트워크 연결에 네트워크 연결이 있는지를 지정합니다.
get_IsConnected get_IsConnected 속성은 로컬 컴퓨터에 네트워크 연결이 있는지를 지정합니다.

get_IsConnectedToInternet get_IsConnectedToInternet 속성은 네트워크에 인터넷 연결이 있는지를 지정합니다.
get_IsConnectedToInternet get_IsConnectedToInternet 속성은 연결된 네트워크 연결에 인터넷 연결이 있는지를 지정합니다.
get_IsConnectedToInternet get_IsConnectedToInternet 속성은 로컬 컴퓨터에 인터넷 연결이 있는지를 지정합니다.
GetAdapterId GetAdapterID 메서드는 이 연결에 사용되는 네트워크 어댑터의 ID를 반환합니다.
GetCategory GetCategory 메서드는 네트워크의 범주를 반환합니다.
GetConnectionId GetConnectionID 메서드는 이 네트워크 연결과 연결된 연결 ID를 반환합니다.
GetConnectivity GetConnectivity 메서드는 네트워크의 연결 상태를 반환합니다.
GetConnectivity GetConnectivity 메서드는 네트워크 연결의 연결 상태를 반환합니다.
GetConnectivity GetConnectivity 메서드는 컴퓨터의 전체 연결 상태를 반환합니다.
GetCost GetCost 메서드는 연결과 관련된 네트워크 비용을 검색합니다.
GetCost GetCost 메서드는 컴퓨터 전체 인터넷 연결의 현재 비용 또는 연결에서 특정 대상으로 라우팅하는 첫 번째 홉을 검색합니다.
GetDataPlanStatus GetDataPlanStatus 메서드는 연결과 연결된 데이터 계획의 상태를 검색합니다.

GetDataPlanStatus GetDataPlanStatus는 컴퓨터 전체 인터넷 연결 또는 연결에서 특정 대상으로 라우팅하는 첫 번째 홉에 대한 데이터 계획 상태를 검색합니다.
GetDescription GetDescription 메서드는 네트워크에 대한 설명 문자열을 반환합니다.
GetDomainType GetDomainType 메서드는 네트워크의 도메인 형식을 반환합니다.
GetDomainType GetDomainType 메서드는 네트워크 연결의 도메인 형식을 반환합니다.
GetInterfaceContextTableForHostName 이 함수는 지정된 호스트 이름 및 연결 프로필 필터에 대한 인터페이스 컨텍스트 테이블을 검색합니다.
GetName GetName 메서드는 네트워크의 이름을 반환합니다.
GetNetwork GetNetwork 메서드는 연결과 연결된 네트워크를 반환합니다.
GetNetwork GetNetwork 메서드는 제공된 네트워크 ID를 기반으로 네트워크를 검색합니다.
GetNetworkConnection GetNetworkConnection 메서드는 제공된 네트워크 연결 ID를 기반으로 네트워크를 검색합니다.
GetNetworkConnections GetNetworkConnections 메서드는 네트워크에 대한 모든 네트워크 연결의 열거형을 반환합니다. 네트워크는 서로 다른 인터페이스 또는 동일한 인터페이스의 다른 링크에서 여러 연결을 가질 수 있습니다.
GetNetworkConnections GetNetworkConnections 메서드는 만들어진 네트워크 연결의 전체 목록을 열거합니다.

GetNetworkId GetNetworkId 메서드는 네트워크의 고유 식별자를 반환합니다.
GetNetworks GetNetworks 메서드는 로컬 컴퓨터에서 사용할 수 있는 네트워크 목록을 검색합니다.
GetTimeCreatedAndConnected GetTimeCreatedAndConnected 메서드는 네트워크를 만들고 연결한 로컬 날짜와 시간을 반환합니다.
IsDomainAuthenticatedBy 지정된 도메인 인증 방법이 이 네트워크에 성공했는지 여부를 쿼리합니다.
IsDomainAuthenticatedBy 이 네트워크 연결에 대해 지정된 도메인 인증 방법이 성공했는지 여부를 쿼리합니다.
NetworkAdded NetworkAdded 메서드는 새 네트워크가 추가될 때 호출됩니다. 새 네트워크의 GUID가 제공됩니다.
NetworkConnectionConnectivityChanged NetworkConnectionConnectivityChanged 메서드는 네트워크 연결 수준에서 연결 변경 이벤트가 발생할 때 클라이언트에 알합니다.
NetworkConnectionPropertyChanged NetworkConnectionPropertyChanged 메서드는 특정 네트워크 연결과 관련된 속성 변경 이벤트가 발생할 때 클라이언트에 알합니다.
NetworkConnectivityChanged NetworkConnectivityChanged 메서드는 네트워크 연결 관련 변경이 발생할 때 호출됩니다. (INetworkEvents.NetworkConnectivityChanged)
NetworkDeleted NetworkDeleted 메서드는 네트워크를 삭제할 때 호출됩니다.
NetworkPropertyChanged NetworkPropertyChanged 메서드는 네트워크 속성 변경이 감지될 때 호출됩니다.

다음 Next 메서드는 열거형 시퀀스에서 지정한 다음 개수의 요소를 가져옵니다. (IEnumNetworkConnections.Next)
다음 Next 메서드는 열거형 시퀀스에서 지정한 다음 개수의 요소를 가져옵니다. (IEnumNetworks.Next)
OnDemandGetRoutingHint OnDemandGetRoutingHint 함수는 경로 요청 캐시에서 대상을 조회하고 일치하는 항목이 발견되면 해당 인터페이스 ID를 반환합니다.
OnDemandRegisterNotification OnDemandRegisterNotification 함수를 사용하면 경로 요청 캐시가 수정될 때 애플리케이션에 알림을 등록할 수 있습니다.
OnDemandUnRegisterNotification OnDemandUnregisterNotification 함수를 사용하면 애플리케이션이 알림을 등록 취소하고 리소스를 정리할 수 있습니다.
재설정 Reset 메서드는 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumNetworkConnections.Reset)
재설정 Reset 메서드는 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumNetworks.Reset)
SetCategory SetCategory 메서드는 네트워크의 범주를 설정합니다. 변경 내용은 즉시 적용됩니다. 이 API의 호출자는 Administrators 그룹의 구성원이어야 합니다.
SetDescription SetDescription 메서드는 네트워크에 대한 설명을 설정하거나 바꿉니다.
SetDestinationAddresses SetDestinationAddresses 메서드는 지정된 대상 IPv4/IPv6 주소를 등록하여 비용 또는 데이터 계획 상태 변경 알림을 받습니다.

SetName SetName 메서드는 네트워크를 설정하거나 이름을 바꿉니다.
SetSimulatedProfileInfo SetSimulatedProfileInfo 메서드는 특정 계량된 인터넷 연결 조건 시뮬레이션을 지원하기 위해 인터넷 연결 프로필에 특정 연결 프로필 값 집합을 적용합니다.
skip Skip 메서드는 열거형 시퀀스에서 지정한 다음 개수의 요소를 건너뜁니다. (IEnumNetworkConnections.Skip)
skip Skip 메서드는 열거형 시퀀스에서 지정한 다음 개수의 요소를 건너뜁니다. (IEnumNetworks.Skip)

인터페이스

IEnumNetworkConnections IEnumNetworkConnections 인터페이스는 네트워크 연결에 대한 표준 열거자를 제공합니다. 네트워크 내의 활성, 연결 끊김 또는 모든 네트워크 연결을 열거합니다. 이 인터페이스는 INetwork 인터페이스에서 가져올 수 있습니다.
IEnumNetworks IEnumNetworks 인터페이스는 네트워크의 표준 열거자입니다. 로컬 컴퓨터에서 사용할 수 있는 모든 네트워크를 열거합니다. 이 인터페이스는 INetworkListManager 인터페이스에서 가져올 수 있습니다.
INetwork INetwork 인터페이스는 로컬 컴퓨터의 네트워크를 나타냅니다. 또한 비슷한 네트워크 서명이 있는 네트워크 연결 컬렉션을 나타낼 수도 있습니다.
INetwork2 **INetwork2** 인터페이스는 로컬 컴퓨터의 네트워크를 나타냅니다. 또한 비슷한 네트워크 서명이 있는 네트워크 연결 컬렉션을 나타낼 수도 있습니다.

INetworkConnection INetworkConnection 인터페이스는 단일 네트워크 연결을 나타냅니다.
INetworkConnection2 **INetworkConnection2** 인터페이스는 단일 네트워크 연결을 나타냅니다.
INetworkConnectionCost 연결과 연결된 현재 네트워크 비용 및 데이터 계획 상태를 쿼리합니다.
INetworkConnectionCostEvents 이 인터페이스는 애플리케이션에 연결에 대한 비용 및 데이터 계획 상태 변경 이벤트를 알립니다.
INetworkConnectionEvents INetworkConnectionEvents 인터페이스는 클라이언트가 네트워크 연결 관련 이벤트를 가져오기 위해 구현하는 메시지 싱크 인터페이스입니다. 하위 수준 이벤트(예: 인증 변경)에 관심이 있는 애플리케이션은 이 인터페이스를 구현해야 합니다.
INetworkCostManager 이 인터페이스를 사용하여 컴퓨터 전체 인터넷 연결에 사용되는 연결 또는 연결의 특정 대상에 대한 라우팅의 첫 번째 홉과 관련된 컴퓨터 차원의 비용 및 데이터 계획 상태 정보를 쿼리합니다.
INetworkCostManagerEvents 이 인터페이스는 머신 전체 비용 및 데이터 계획 관련 이벤트를 애플리케이션에 알립니다.
INetworkEvents INetworkEvents는 클라이언트가 네트워크 관련 이벤트를 가져오기 위해 구현하는 알림 싱크 인터페이스입니다. 이러한 API는 각 이벤트가 발생할 때 자동으로 호출되는 모든 콜백 함수입니다.
INetworkListManager INetworkListManager 인터페이스는 네트워크 목록 관리 기능을 수행하는 메서드 집합을 제공합니다.
INetworkListManagerEvents INetworkListManagerEvents는 클라이언트가 전체 컴퓨터 상태 관련 이벤트를 가져오기 위해 구현하는 메시지 싱크 인터페이스입니다. 인터넷 연결과 같은 상위 수준 이벤트에 관심이 있는 애플리케이션은 이 인터페이스를 구현합니다.

구조체

NET_INTERFACE_CONTEXT NET_INTERFACE_CONTEXT_TABLE 구조체의 일부인 인터페이스 컨텍스트입니다.
NET_INTERFACE_CONTEXT_TABLE NET_INTERFACE_CONTEXT 구조체의 테이블입니다.
NLM_DATAPLAN_STATUS NLM_DATAPLAN_STATUS 구조체는 운송업체에서 제공하는 현재 데이터 계획 상태 정보를 저장합니다.
NLM_SIMULATED_PROFILE_INFO SetSimulatedProfileInfo에서 RDP 자식 세션에서 현재 인터넷 연결 프로필 값을 재정의하여 특정 데이터 통신 인터넷 연결 조건의 시뮬레이션을 지원하는 데 사용되는 값을 지정하는 데 사용됩니다.
NLM_SOCKADDR NLM_SOCKADDR 구조에는 IPv4/IPv6 대상 주소가 포함됩니다.
NLM_USAGE_DATA NLM_USAGE_DATA 구조체는 계획의 데이터 사용량을 나타내는 정보를 저장합니다.

네트워크 관리

아티클2023. 03. 11.

네트워크 관리 기술 개요입니다.

네트워크 관리를 개발하려면 다음 헤더가 필요합니다.

- atacct.h
- lmalert.h
- lmapibuf.h
- lmat.h
- lmaudit.h
- lmconfig.h
- lmerrlog.h
- lmjoin.h
- lmmsg.h
- lmremutl.h
- lmserver.h
- lmsvc.h
- lmuse.h
- lmwksta.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 네트워크 관리

열거형

DSREG_JOIN_TYPE 디바이스를 Microsoft Azure Active Directory 조인할 수 있는 가능한 방법을 지정합니다.

함수

ALERT_OTHER_INFO ALERT_OTHER_INFO 매크로는 경고 메시지의 경고 관련 데이터에 대한 포인터를 반환합니다. 데이터는 STD_ALERT 구조를 따르며 ADMIN_OTHER_INFO, PRINT_OTHER_INFO 또는 USER_OTHER_INFO 구조체일 수 있습니다.
ALERT_VAR_DATA ALERT_VAR_DATA 매크로는 경고 메시지의 가변 길이 부분에 대한 포인터를 반환합니다. 가변 길이 데이터는 ADMIN_OTHER_INFO, PRINT_OTHER_INFO 또는 USER_OTHER_INFO 구조를 따를 수 있습니다.
GetNetScheduleAccountInformation GetNetScheduleAccountInformation 함수는 AT 서비스 계정 이름을 검색합니다.
NetAccessAdd 지원 안 됨 (NetAccessAdd)
NetAccessDel 지원 안 됨 (NetAccessDel)
NetAccessEnum 지원 안 됨 (NetAccessEnum)
NetAccessGetInfo 지원 안 됨 (NetAccessGetInfo)
NetAccessGetUserPerms 지원 안 됨 (NetAccessGetUserPerms)
NetAccessSetInfo 지원 안 됨 (NetAccessSetInfo)
NetAddAlternateComputerName 지정된 컴퓨터의 대체 이름을 추가합니다.
NetAlertRaise NetAlertRaise 함수는 특정 이벤트가 발생할 때 등록된 모든 클라이언트에 알합니다.

NetAlertRaiseEx NetAlertRaiseEx 함수는 특정 이벤트가 발생할 때 등록된 모든 클라이언트에 알합니다. NetAlertRaiseEx에서 STD_ALERT 구조를 지정할 필요가 없으므로 이 확장 함수를 호출하여 경고 메시지 전송을 간소화할 수 있습니다.
NetApiBufferAllocate NetApiBufferAllocate 함수는 힙에서 메모리를 할당합니다. NetApiBufferFree 함수와의 호환성이 필요한 경우에만 이 함수를 사용합니다. 그렇지 않으면 메모리 관리 함수를 사용합니다.
NetApiBufferFree NetApiBufferFree 함수는 NetApiBufferAllocate 함수가 할당하는 메모리를 해제합니다. 또한 애플리케이션은 NetApiBufferFree를 호출하여 다른 네트워크 관리 함수가 내부적으로 정보를 반환하는 데 사용하는 메모리를 해제해야 합니다.
NetApiBufferReallocate NetApiBufferReallocate 함수는 NetApiBufferAllocate 함수에 대한 이전 호출에 의해 할당된 버퍼의 크기를 변경합니다.
NetApiBufferSize NetApiBufferSize 함수는 NetApiBufferAllocate 함수 호출에 의해 할당된 버퍼의 크기를 바이트 단위로 반환합니다.
NetAuditClear NetAuditClear 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함됩니다. 다른 애플리케이션은 이벤트 로깅을 사용해야 합니다.
NetAuditRead NetAuditRead 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함됩니다. 다른 애플리케이션은 이벤트 로깅을 사용해야 합니다.
NetAuditWrite NetAuditWrite 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함됩니다. 다른 애플리케이션은 이벤트 로깅을 사용해야 합니다.
NetConfigGet NetConfigGet 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함됩니다. 다른 애플리케이션은 레지스트리를 사용해야 합니다.

NetConfigGetAll NetConfigGetAll 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함됩니다. 다른 애플리케이션은 레지스트리를 사용해야 합니다.
NetConfigSet NetConfigSet 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함됩니다. 다른 애플리케이션은 레지스트리를 사용해야 합니다.
NetCreateProvisioningPackage 오프라인 도메인 조인 작업에서 나중에 사용할 수 있도록 컴퓨터 계정을 프로비전하는 프로비저닝 패키지를 만듭니다. 또한 패키지에는 프로비전 중에 컴퓨터에 추가할 인증서 및 정책에 대한 정보가 포함될 수 있습니다.
NetEnumerateComputerNames 지정된 컴퓨터의 이름을 열거합니다.
NetErrorLogClear NetErrorLogClear 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함되어 있습니다. 다른 애플리케이션은 이벤트 로깅을 사용해야 합니다.
NetErrorLogRead NetErrorLogRead 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함되어 있습니다. 다른 애플리케이션은 이벤트 로깅을 사용해야 합니다.
NetErrorLogWrite NetErrorLogWrite 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함되어 있습니다. 다른 애플리케이션은 이벤트 로깅을 사용해야 합니다.
NetFreeAadJoinInformation 테넌트에 대한 조인 정보와 NetGetAadJoinInformation 함수를 호출하여 검색한 지정된 DSREG_JOIN_INFO 구조에 할당된 메모리를 해제합니다.
NetGetAadJoinInformation 지정된 테넌트에 대한 조인 정보를 검색합니다. 이 함수는 현재 사용자가 추가한 Microsoft Azure Active Directory 및 회사 계정에 대한 조인 정보를 검사합니다.
NetGetAnyDCName NetGetAnyDCName 함수는 지정된 서버에서 직접 신뢰하는 도메인에 대한 DC(도메인 컨트롤러)의 이름을 반환합니다.

NetGetDCName NetGetDCName 함수는 PDC(주 도메인 컨트롤러)의 이름을 반환합니다. 지정된 도메인에 대한 백업 도메인 컨트롤러(BDC)의 이름을 반환하지 않습니다. 또한 이 함수는 PDC가 아닌 서버로 원격으로 연결할 수 없습니다.
NetGetDisplayInformationIndex NetGetDisplayInformationIndex 함수는 이름이 지정된 문자열로 시작하거나 이름이 사전순으로 문자열을 따르는 첫 번째 표시 정보 항목의 인덱스를 반환합니다.
NetGetJoinableOU NetGetJoinableOUs 함수는 컴퓨터 계정을 만들 수 있는 OU(조직 구성 단위) 목록을 검색합니다.
NetGetJoinInformation NetGetJoinInformation 함수는 지정된 컴퓨터에 대한 조인 상태 정보를 검색합니다.
NetGroupAdd NetGroupAdd 함수는 SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory인 보안 데이터베이스에 전역 그룹을 만듭니다.
NetGroupAddUser NetGroupAddUser 함수는 보안 데이터베이스의 기존 전역 그룹(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory의 기존 사용자 계정 멤버 자격을 제공합니다.
NetGroupDel NetGroupDel 함수는 SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory인 보안 데이터베이스에서 전역 그룹을 삭제합니다.
NetGroupDelUser NetGroupDelUser 함수는 SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory인 보안 데이터베이스의 특정 글로벌 그룹에서 사용자를 제거합니다.
NetGroupEnum NetGroupEnum 함수는 SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory인 보안 데이터베이스의 각 글로벌 그룹에 대한 정보를 검색합니다.
NetGroupGetInfo NetGroupGetInfo 함수는 SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory인 보안 데이터베이스의 특정 글로벌 그룹에 대한 정보를 검색합니다.

NetGroupGetUsers NetGroupGetUsers 함수는 보안 데이터베이스의 특정 전역 그룹(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory의 멤버 목록을 검색합니다.
NetGroupSetInfo NetGroupSetInfo 함수는 SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory인 보안 데이터베이스에 있는 전역 그룹의 매개 변수를 설정합니다.
NetGroupSetUsers NetGroupSetUsers 함수는 지정된 전역 그룹에 대한 멤버 자격을 설정합니다.
NetJoinDomain NetJoinDomain 함수는 컴퓨터를 작업 그룹 또는 도메인에 조인합니다.
NetLocalGroupAdd NetLocalGroupAdd 함수는 SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory인 보안 데이터베이스에 로컬 그룹을 만듭니다.
NetLocalGroupAddMember NetLocalGroupAddMember 함수는 사용되지 않습니다. 대신 NetLocalGroupAddMembers 함수를 사용해야 합니다.
NetLocalGroupAddMembers NetLocalGroupAddMembers 함수는 하나 이상의 기존 사용자 계정 또는 전역 그룹 계정의 멤버 자격을 기존 로컬 그룹에 추가합니다.
NetLocalGroupDel NetLocalGroupDel 함수는 SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory인 보안 데이터베이스에서 로컬 그룹 계정 및 모든 해당 멤버를 삭제합니다.
NetLocalGroupDelMember NetLocalGroupDelMember 함수는 사용되지 않습니다. 대신 NetLocalGroupDelMembers 함수를 사용해야 합니다.
NetLocalGroupDelMembers NetLocalGroupDelMembers 함수는 기존 로컬 그룹에서 하나 이상의 멤버를 제거합니다. 로컬 그룹 구성원은 사용자 또는 전역 그룹일 수 있습니다.

NetLocalGroupEnum NetLocalGroupEnum 함수는 지정된 서버의 각 로컬 그룹 계정에 대한 정보를 반환합니다.
NetLocalGroupGetInfo NetLocalGroupGetInfo 함수는 서버의 특정 로컬 그룹 계정에 대한 정보를 검색합니다.
NetLocalGroupGetMembers NetLocalGroupGetMembers 함수는 보안 데이터베이스에서 특정 로컬 그룹의 멤버 목록(SAM(보안 계정 관리자) 데이터베이스) 또는 도메인 컨트롤러의 경우 Active Directory를 검색합니다.
NetLocalGroupSetInfo NetLocalGroupSetInfo 함수는 기존 로컬 그룹의 이름을 변경합니다. 또한 함수는 로컬 그룹과 주석을 연결합니다.
NetLocalGroupSetMembers NetLocalGroupSetMembers 함수는 지정된 로컬 그룹에 대한 멤버 자격을 설정합니다.
NetMessageBufferSend NetMessageBufferSend 함수는 정보 버퍼를 등록된 메시지 별칭으로 보냅니다.
NetMessageNameAdd NetMessageNameAdd 함수는 메시지 이름 테이블에 메시지 별칭을 등록합니다. 이 기능을 사용하려면 메신저 서비스를 시작해야 합니다.
NetMessageNameDel NetMessageNameDel 함수는 메시지 이름 테이블에서 메시지 별칭을 삭제합니다. 이 기능을 사용하려면 메신저 서비스를 시작해야 합니다.
NetMessageNameEnum NetMessageNameEnum 함수는 지정된 컴퓨터에서 메시지를 받는 메시지 별칭을 나열합니다. 이 기능을 사용하려면 메신저 서비스를 시작해야 합니다.
NetMessageNameGetInfo NetMessageNameGetInfo 함수는 메시지 이름 테이블의 특정 메시지 별칭에 대한 정보를 검색합니다. 이 기능을 사용하려면 메신저 서비스를 시작해야 합니다.
NetProvisionComputerAccount 오프라인 도메인 가입 작업에서 나중에 사용할 컴퓨터 계정을 프로비전합니다.

NetQueryDisplayInformation NetQueryDisplayInformation 함수는 사용자 계정, 컴퓨터 또는 그룹 계정 정보를 반환합니다. 이 함수를 호출하여 사용자 인터페이스에 표시할 계정 정보를 빠르게 열거합니다.
NetRemoteComputerSupports NetRemoteComputerSupports 함수는 리트리버를 쿼리하여 원격 시스템에서 지원하는 선택적 기능을 검색합니다.
NetRemoteTOD NetRemoteTOD 함수는 지정된 서버에서 하루 중 시간 정보를 반환합니다.
NetRemoveAlternateComputerName 지정된 컴퓨터의 대체 이름을 제거합니다.
NetRenameMachineInDomain NetRenameMachineInDomain 함수는 도메인에 있는 컴퓨터의 이름을 변경합니다.
NetRequestOfflineDomainJoin 컴퓨터에서 로컬로 실행하여 볼륨에 탑재된 Windows 운영 체제 이미지를 수정합니다. (NetRequestOfflineDomainJoin)
NetRequestProvisioningPackageInstall 컴퓨터에서 로컬로 실행하여 볼륨에 탑재된 Windows 운영 체제 이미지를 수정합니다. (NetRequestProvisioningPackageInstall)
NetScheduleJobAdd NetScheduleJobAdd 함수는 지정된 미래 시간 및 날짜에 실행할 작업을 제출합니다. 이 기능을 사용하려면 작업이 제출된 컴퓨터에서 일정 서비스를 시작해야 합니다.
NetScheduleJobDel NetScheduleJobDel 함수는 컴퓨터에서 실행하기 위해 대기하는 작업 범위를 삭제합니다. 이 함수를 사용하려면 작업 삭제 요청이 전송되는 컴퓨터에서 일정 서비스를 시작해야 합니다.
NetScheduleJobEnum NetScheduleJobEnum 함수는 지정된 컴퓨터에서 큐에 대기 중인 작업을 나열합니다. 이 함수를 사용하려면 일정 서비스를 시작해야 합니다.

NetScheduleJobGetInfo NetScheduleJobGetInfo 함수는 지정된 컴퓨터에서 대기 중인 특정 작업에 대한 정보를 검색합니다. 이 함수를 사용하려면 일정 서비스를 시작해야 합니다.
NetServerComputerNameAdd NetServerComputerNameAdd 함수는 지정된 서버가 활성 상태인 전송을 열거하고 에뮬레이트된 서버 이름을 각 전송에 바인딩합니다.
NetServerComputerNameDel NetServerComputerNameDel 함수를 사용하면 지정된 서버가 NetServerComputerNameAdd 함수에 대한 이전 호출로 설정된 에뮬레이트된 서버 이름 지원을 중단합니다. 이 함수는 에뮬레이트된 이름에서 네트워크 전송을 바인딩 해제하여 이 작업을 수행합니다.
NetServerDiskEnum NetServerDiskEnum 함수는 서버의 디스크 드라이브 목록을 검색합니다. 이 함수는 세 문자 문자열(드라이브 문자, 콜론 및 종료 null 문자)의 배열을 반환합니다.
NetServerEnum NetServerEnum 함수는 도메인에 표시되는 지정된 형식의 모든 서버를 나열합니다.
NetServerGetInfo NetServerGetInfo 함수는 지정된 서버에 대한 현재 구성 정보를 검색합니다.
NetServerSetInfo NetServerSetInfo 함수는 서버의 운영 매개 변수를 설정합니다. 개별적으로 또는 집합적으로 설정할 수 있습니다. 이 정보는 시스템을 다시 초기화한 후에도 계속 적용할 수 있는 방식으로 저장됩니다.
NetServerTransportAdd NetServerTransportAdd 함수는 서버를 전송 프로토콜에 바인딩합니다.
NetServerTransportAddEx NetServerTransportAddEx 함수는 지정된 서버를 전송 프로토콜에 바인딩합니다.
NetServerTransportDel NetServerTransportDel 함수는 서버에서 전송 프로토콜을 바인딩 해제(또는 연결 해제)합니다. 실제로 서버는 지정된 전송 프로토콜(예: TCP 또는 XNS)을 사용하여 클라이언트와 더 이상 통신할 수 없습니다.

NetServerTransportEnum NetServerTransportEnum 함수는 서버에서 관리하는 전송 프로토콜에 대한 정보를 제공합니다.
NetServiceControl NetServiceControl 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함되어 있습니다. 다른 애플리케이션은 서비스 함수를 사용해야 합니다.
NetServiceEnum NetServiceEnum 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함되어 있습니다. 다른 애플리케이션은 서비스 함수를 사용해야 합니다.
NetServiceGetInfo NetServiceGetInfo 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함되어 있습니다. 다른 애플리케이션은 서비스 함수를 사용해야 합니다.
NetServiceInstall NetServiceInstall 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함되어 있습니다. 다른 애플리케이션은 서비스 함수를 사용해야 합니다.
NetSetPrimaryComputerName 지정된 컴퓨터의 기본 컴퓨터 이름을 설정합니다.
NetUnjoinDomain NetUnjoinDomain 함수는 작업 그룹 또는 도메인에서 컴퓨터를 연결 해제합니다.
NetUseAdd NetUseAdd 함수는 로컬 컴퓨터와 원격 서버 간의 연결을 설정합니다.
NetUseDel NetUseDel 함수는 공유 리소스에 대한 연결을 종료합니다.
NetUseEnum NetUseEnum 함수는 로컬 컴퓨터와 원격 서버의 리소스 간의 모든 현재 연결을 나열합니다.
NetUseGetInfo NetUseGetInfo 함수는 공유 리소스에 대한 연결에 대한 정보를 검색합니다.

NetUserAdd NetUserAdd 함수는 사용자 계정을 추가하고 암호 및 권한 수준을 할당합니다.
NetUserChangePassword NetUserChangePassword 함수는 지정된 네트워크 서버 또는 도메인에 대한 사용자의 암호를 변경합니다.
NetUserDel NetUserDel 함수는 서버에서 사용자 계정을 삭제합니다.
NetUserEnum NetUserEnum 함수는 서버의 모든 사용자 계정에 대한 정보를 검색합니다.
NetUserGetGroups NetUserGetGroups 함수는 지정된 사용자가 속한 전역 그룹 목록을 검색합니다.
NetUserGetInfo NetUserGetInfo 함수는 서버의 특정 사용자 계정에 대한 정보를 검색합니다.
NetUserGetLocalGroups NetUserGetLocalGroups 함수는 지정된 사용자가 속한 로컬 그룹 목록을 검색합니다.
NetUserModalsGet NetUserModalsGet 함수는 SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory인 보안 데이터베이스의 모든 사용자 및 전역 그룹에 대한 전역 정보를 검색합니다.
NetUserModalsSet NetUserModalsSet 함수는 SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory인 보안 데이터베이스의 모든 사용자 및 전역 그룹에 대한 전역 정보를 설정합니다.
NetUserSetGroups NetUserSetGroups 함수는 지정된 사용자 계정에 대한 전역 그룹 멤버 자격을 설정합니다.
NetUserSetInfo NetUserSetInfo 함수는 사용자 계정의 매개 변수를 설정합니다.

NetValidateName NetValidateName 함수는 이름이 지정된 이름 유형(컴퓨터 이름, 작업 그룹 이름, 도메인 이름 또는 DNS 컴퓨터 이름)에 대해 유효한지 확인합니다.
NetValidatePasswordPolicy NetValidatePasswordPolicy 함수를 사용하면 애플리케이션이 애플리케이션에서 제공하는 계정 데이터베이스에 대한 암호 준수를 확인하고 암호가 암호 정책의 복잡성, 노후화, 최소 길이 및 기록 재사용 요구 사항을 충족하는지 확인할 수 있습니다.
NetValidatePasswordPolicyFree NetValidatePasswordPolicyFree 함수는 NET_VALIDATE_OUTPUT_ARG 구조인 OutputArg 매개 변수에 대해 NetValidatePasswordPolicy 함수가 할당하는 메모리를 해제합니다.
NetWkstaGetInfo NetWkstaGetInfo 함수는 워크스테이션의 구성에 대한 정보를 반환합니다.
NetWkstaSetInfo NetWkstaSetInfo 함수는 시스템을 다시 초기화한 후에도 계속 적용되는 정보로 워크스테이션을 구성합니다.
NetWkstaTransportAdd 지원 안 됨 (NetWkstaTransportAdd)
NetWkstaTransportDel 지원 안 됨 (NetWkstaTransportDel)
NetWkstaTransportEnum NetWkstaTransportEnum 함수는 서버 컴퓨터에 대한 파일 요청을 생성하는 클라이언트 컴퓨터의 소프트웨어인 리렉터에서 관리하는 전송 프로토콜에 대한 정보를 제공합니다.
NetWkstaUserEnum NetWkstaUserEnum 함수는 현재 워크스테이션에 로그온한 모든 사용자에 대한 정보를 나열합니다. 이 목록에는 대화형, 서비스 및 일괄 로그온이 포함됩니다.
NetWkstaUserGetInfo NetWkstaUserGetInfo 함수는 현재 로그온한 사용자에 대한 정보를 반환합니다. 이 함수는 로그온한 사용자의 컨텍스트에서 호출되어야 합니다.

NetWkstaUserSetInfo NetWkstaUserSetInfo 함수는 워크스테이션의 구성 요소에 대한 사용자별 정보를 설정합니다.
SetNetScheduleAccountInformation SetNetScheduleAccountInformation 함수는 AT 서비스 계정 이름 및 암호를 설정합니다. AT 서비스 계정 이름 및 암호는 NetScheduleJobAdd를 사용하여 만든 예약된 작업의 자격 증명으로 사용됩니다.

구조체

ADMIN_OTHER_INFO ADMIN_OTHER_INFO 구조에는 오류 메시지 정보가 포함되어 있습니다. NetAlertRaise 및 NetAlertRaiseEx 함수는 ADMIN_OTHER_INFO 구조를 사용하여 관리자의 중단 메시지를 발생시키는 경우 정보를 지정합니다.
AT_ENUM AT_ENUM 구조에는 제출된 작업에 대한 정보가 포함됩니다. NetScheduleJobEnum 함수는 이 구조를 사용하여 제출된 작업의 전체 큐에 대한 정보를 열거하고 반환합니다.
AT_INFO AT_INFO 구조에는 작업에 대한 정보가 포함됩니다.
DSREG_JOIN_INFO 디바이스가 Microsoft Azure Active Directory 조인되는 방법에 대한 정보를 포함합니다.
DSREG_USER_INFO 디바이스를 Microsoft Azure Active Directory 조인하는 데 사용되는 사용자 계정에 대한 정보를 포함합니다.
ERRLOG_OTHER_INFO ERRLOG_OTHER_INFO 구조에는 오류 로그 정보가 포함됩니다. NetAlertRaise 및 NetAlertRaiseEx 함수는 ERRLOG_OTHER_INFO 구조를 사용하여 오류 로그에 새 항목을 추가할 때 정보를 지정합니다.

GROUP_INFO_0 GROUP_INFO_0 구조에는 보안 데이터베이스의 전역 그룹 이름(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory의 이름이 포함됩니다.
GROUP_INFO_1 GROUP_INFO_1 구조에는 전역 그룹 이름과 그룹과 연결할 주석이 포함됩니다.
GROUP_INFO_1002 GROUP_INFO_1002 구조에는 전역 그룹과 연결할 주석이 포함되어 있습니다.
GROUP_INFO_1005 GROUP_INFO_1005 구조에는 전역 그룹과 연결된 리소스 특성이 포함됩니다.
GROUP_INFO_2 GROUP_INFO_2 구조에는 이름, 식별자 및 리소스 특성을 포함하여 전역 그룹에 대한 정보가 포함됩니다.
GROUP_INFO_3 GROUP_INFO_3 구조에는 이름, SID(보안 식별자) 및 리소스 특성을 포함하여 전역 그룹에 대한 정보가 포함됩니다.
GROUP_USERS_INFO_0 GROUP_USERS_INFO_0 구조에는 전역 그룹 멤버 정보가 포함됩니다.
GROUP_USERS_INFO_1 GROUP_USERS_INFO_1 구조에는 전역 그룹 멤버 정보가 포함됩니다.
LOCALGROUP_INFO_0 LOCALGROUP_INFO_0 구조체에는 로컬 그룹 이름이 포함됩니다.
LOCALGROUP_INFO_1 LOCALGROUP_INFO_1 구조에는 로컬 그룹 이름과 로컬 그룹을 설명하는 주석이 포함되어 있습니다.
LOCALGROUP_INFO_1002 LOCALGROUP_INFO_1002 구조에는 로컬 그룹을 설명하는 주석이 포함되어 있습니다.

LOCALGROUP_MEMBERS_INFO_0 LOCALGROUP_MEMBERS_INFO_0 구조에는 로컬 그룹 멤버와 연결된 SID(보안 식별자)가 포함됩니다. 멤버는 사용자 계정 또는 전역 그룹 계정일 수 있습니다.
LOCALGROUP_MEMBERS_INFO_1 LOCALGROUP_MEMBERS_INFO_1 구조에는 로컬 그룹의 구성원과 연결된 SID(보안 식별자) 및 계정 정보가 포함됩니다.
LOCALGROUP_MEMBERS_INFO_2 LOCALGROUP_MEMBERS_INFO_2 구조에는 로컬 그룹 구성원과 연결된 SID(보안 식별자) 및 계정 정보가 포함됩니다.
LOCALGROUP_MEMBERS_INFO_3 LOCALGROUP_MEMBERS_INFO_3 구조에는 로컬 그룹 구성원과 연결된 계정 이름 및 도메인 이름이 포함됩니다.
LOCALGROUP_USERS_INFO_0 LOCALGROUP_USERS_INFO_0 구조에는 로컬 그룹 멤버 정보가 포함됩니다.
MSG_INFO_0 MSG_INFO_0 구조체는 메시지 별칭을 지정합니다.
MSG_INFO_1 MSG_INFO_1 구조체는 메시지 별칭을 지정합니다. 이 구조는 호환성을 위해서만 존재합니다. 메시지 전달은 지원되지 않습니다.
NET_DISPLAY_GROUP NET_DISPLAY_GROUP 구조에는 계정 관리자가 그룹 계정에 대한 정보를 확인하기 위해 액세스할 수 있는 정보가 포함되어 있습니다.
NET_DISPLAY_MACHINE NET_DISPLAY_MACHINE 구조에는 계정 관리자가 컴퓨터 및 해당 특성에 대한 정보를 확인하기 위해 액세스할 수 있는 정보가 포함되어 있습니다.
NET_DISPLAY_USER NET_DISPLAY_USER 구조에는 계정 관리자가 사용자 계정에 대한 정보를 확인하기 위해 액세스할 수 있는 정보가 포함되어 있습니다.

NET_VALIDATE_AUTHENTICATION_INPUT_ARG 클라이언트 애플리케이션은 애플리케이션이 인증 유효성 검사를 요청할 때 NET_VALIDATE_AUTHENTICATION_INPUT_ARG 구조를 NetValidatePasswordPolicy 함수에 전달합니다.
NET_VALIDATE_OUTPUT_ARG NET_VALIDATE_OUTPUT_ARG 구조에는 사용자의 마지막 로그온 이후 변경된 영구 암호 관련 데이터와 함수의 암호 유효성 검사 결과에 대한 정보가 포함됩니다.
NET_VALIDATE_PASSWORD_CHANGE_INPUT_ARG 클라이언트 애플리케이션은 애플리케이션이 암호 변경 유효성 검사를 요청할 때 NET_VALIDATE_PASSWORD_CHANGE_INPUT_ARG 구조를 NetValidatePasswordPolicy 함수에 전달합니다.
NET_VALIDATE_PASSWORD_HASH NET_VALIDATE_PASSWORD_HASH 구조에는 암호 해시가 포함되어 있습니다.
NET_VALIDATE_PASSWORD_RESET_INPUT_ARG 클라이언트 애플리케이션은 애플리케이션이 암호 재설정 유효성 검사를 요청할 때 NET_VALIDATE_PASSWORD_RESET_INPUT_ARG 구조를 NetValidatePasswordPolicy 함수에 전달합니다.
NET_VALIDATE_PERSISTED_FIELDS NET_VALIDATE_PERSISTED_FIELDS 구조에는 사용자의 암호 속성에 대한 정보가 포함됩니다.
NETSETUP_PROVISIONING_PARAMS NETSETUP_PROVISIONING_PARAMS 구조에는 NetCreateProvisionPackage 함수를 사용하여 프로비저닝 패키지를 만들 때 사용되는 정보가 포함되어 있습니다.
PRINT_OTHER_INFO 인쇄 작업에 대한 정보를 포함합니다.
SERVER_INFO_100 SERVER_INFO_100 구조에는 이름 및 플랫폼을 포함하여 지정된 서버에 대한 정보가 포함됩니다.
SERVER_INFO_1005 SERVER_INFO_1005 구조에는 지정된 서버를 설명하는 주석이 포함되어 있습니다.

SERVER_INFO_101 SERVER_INFO_101 구조에는 이름, 플랫폼, 서버 유형 및 관련 소프트웨어를 포함하여 지정된 서버에 대한 정보가 포함됩니다.
SERVER_INFO_1010 SERVER_INFO_1010 구조에는 지정된 서버와 연결된 자동 연결 끊기 시간이 포함됩니다.
SERVER_INFO_1016 SERVER_INFO_1016 구조에는 서버가 동일한 네트워크 도메인의 다른 컴퓨터에 표시되는지 여부에 대한 정보가 포함됩니다.
SERVER_INFO_1017 SERVER_INFO_1017 구조에는 지정된 서버와 연결된 네트워크 공지 속도가 포함됩니다.
SERVER_INFO_1018 SERVER_INFO_1018 구조에는 지정된 서버에 대해 공지 속도가 얼마나 달라질 수 있는지에 대한 정보가 포함되어 있습니다.
SERVER_INFO_102 이름, 플랫폼, 서버 유형, 특성 및 관련 소프트웨어를 포함하여 지정된 서버에 대한 정보를 포함합니다.
SERVER_INFO_1107 SERVER_INFO_1107 구조는 지정된 서버에 동시에 로그온할 수 있는 사용자 수를 지정합니다.
SERVER_INFO_1501 SERVER_INFO_1501 구조체는 지정된 서버의 한 세션에서 열 수 있는 파일 수를 지정합니다.
SERVER_INFO_1502 SERVER_INFO_1502 구조는 지정된 서버에 대한 클라이언트당 최대 가상 회로 수를 지정합니다.
SERVER_INFO_1503 SERVER_INFO_1503 구조는 동시에 수행할 수 있는 검색 작업의 수를 지정합니다.
SERVER_INFO_1506 SERVER_INFO_1506 구조에는 지정된 서버에서 할당할 수 있는 최대 작업 항목 수에 대한 정보가 포함됩니다.

SERVER_INFO_1509 SERVER_INFO_1509 구조체는 최대 원시 모드 버퍼 크기를 지정합니다.
SERVER_INFO_1510 SERVER_INFO_1510 구조는 단일 가상 회로를 사용하여 지정된 서버에 로그온할 수 있는 최대 사용자 수를 지정합니다.
SERVER_INFO_1511 SERVER_INFO_1511 구조는 사용자가 단일 가상 회로로 만들 수 있는 최대 트리 연결 수를 지정합니다.
SERVER_INFO_1512 SERVER_INFO_1512 구조에는 지정된 서버가 특정 시간에 할당할 수 있는 페이지가 지정되지 않은 메모리의 최대 크기가 포함됩니다.
SERVER_INFO_1513 SERVER_INFO_1513 구조에는 지정된 서버가 특정 시간에 할당할 수 있는 최대 페이지 가능 메모리 크기가 포함됩니다.
SERVER_INFO_1515 SERVER_INFO_1515 구조는 클라이언트의 로그온 시간이 만료되면 서버에서 클라이언트의 연결을 강제로 끊어야 하는지 여부를 지정합니다.
SERVER_INFO_1516 SERVER_INFO_1516 구조는 서버가 신뢰할 수 있는 시간 원본인지 여부를 지정합니다.
SERVER_INFO_1518 SERVER_INFO_1518 구조는 서버가 LAN Manager 2.x 클라이언트에 표시되는지 여부를 지정합니다.
SERVER_INFO_1523 SERVER_INFO_1523 구조는 서버가 불완전한 검색 작업에 대한 정보를 유지하는 기간을 지정합니다.
SERVER_INFO_1528 SERVER_INFO_1528 구조는 서비스 요청을 깨우기 전에 청소기가 유휴 상태로 유지되는 기간을 지정합니다.

SERVER_INFO_1529 SERVER_INFO_1529 구조는 서버가 더 많은 항목을 할당하기 전에 필요한 최소 수신 작업 항목 수를 지정합니다.
SERVER_INFO_1530 SERVER_INFO_1530 구조는 서버에서 서버 메시지 블록 처리를 시작하는 데 필요한 사용 가능한 수신 작업 항목의 최소 수를 지정합니다.
SERVER_INFO_1533 SERVER_INFO_1533 구조는 클라이언트가 서버에 보낼 수 있는 최대 미해결 요청 수를 지정합니다.
SERVER_INFO_1536 SERVER_INFO_1536 구조는 서버에서 클라이언트가 파일에서 기회 잠금(oplock)을 사용할 수 있도록 허용하는지 여부를 지정합니다.
SERVER_INFO_1538 SERVER_INFO_1538 구조는 여러 MS-DOS FCB(파일 제어 블록)가 단일 위치에 배치되는지 여부를 지정합니다.
SERVER_INFO_1539 SERVER_INFO_1539 구조는 서버가 원시 SMB(서버 메시지 블록)를 처리하는지 여부를 지정합니다.
SERVER_INFO_1540 SERVER_INFO_1540 구조는 서버에서 리디렉션된 서버 드라이브를 공유할 수 있는지 여부를 지정합니다.
SERVER_INFO_1541 SERVER_INFO_1541 구조는 서버에 연결하기 위한 클라이언트의 요청 버스트를 처리하기 위해 서버가 따로 설정한 최소 사용 가능한 연결 블록 수를 지정합니다.
SERVER_INFO_1542 SERVER_INFO_1542 구조는 클라이언트가 서버에 연결하기 위한 요청의 버스트를 처리하기 위해 서버가 따로 설정한 최대 사용 가능한 연결 블록 수를 지정합니다.
SERVER_INFO_1544 SERVER_INFO_1544 구조는 연결 테이블에 할당할 초기 트리 연결 수를 지정합니다.

SERVER_INFO_1550 SERVER_INFO_1550 구조는 경고 메시지를 보내기 전에 남은 사용 가능한 디스크 공간의 백분율을 지정합니다.
SERVER_INFO_1552 SERVER_INFO_1552 구조체는 링크 지연에 허용되는 최대 시간을 지정합니다.
SERVER_INFO_402 지정된 서버에 대한 정보를 포함합니다.
SERVER_INFO_403 SERVER_INFO_403 구조에는 지정된 서버에 대한 정보가 포함됩니다.
SERVER_INFO_502 SERVER_INFO_502 구조체는 사용되지 않습니다. 구조체에는 지정된 서버에 대한 정보가 포함됩니다.
SERVER_INFO_503 SERVER_INFO_503 구조체는 사용되지 않습니다. 구조체에는 지정된 서버에 대한 정보가 포함됩니다.
SERVER_TRANSPORT_INFO_0 SERVER_TRANSPORT_INFO_0 구조에는 네트워크의 이름, 주소 및 위치를 포함하여 지정된 전송 프로토콜에 대한 정보가 포함됩니다.
SERVER_TRANSPORT_INFO_1 SERVER_TRANSPORT_INFO_1 구조에는 이름 및 주소를 포함하여 지정된 전송 프로토콜에 대한 정보가 포함됩니다. 이 정보 수준은 NetServerTransportAddEx 함수에만 유효합니다.
SERVER_TRANSPORT_INFO_2 SERVER_TRANSPORT_INFO_2 구조에는 전송 이름 및 주소를 포함하여 지정된 전송 프로토콜에 대한 정보가 포함됩니다. 이 정보 수준은 NetServerTransportAddEx 함수에만 유효합니다.
SERVER_TRANSPORT_INFO_3 SERVER_TRANSPORT_INFO_3 구조에는 이름, 주소 및 암호(자격 증명)를 포함하여 지정된 전송 프로토콜에 대한 정보가 포함됩니다. 이 정보 수준은 NetServerTransportAddEx 함수에만 유효합니다.

STD_ALERT STD_ALERT 구조에는 중요한 이벤트가 발생한 시간과 날짜가 포함됩니다.
TIME_OF_DAY_INFO TIME_OF_DAY_INFO 구조에는 원격 서버의 하루 중 시간에 대한 정보가 포함됩니다.
USE_INFO_0 USE_INFO_0 구조에는 공유 리소스의 이름과 해당 리소스로 리디렉션된 로컬 디바이스가 포함됩니다.
USE_INFO_1 로컬 디바이스와 공유 리소스 간의 연결에 대한 정보를 포함합니다.
USE_INFO_2 USE_INFO_2 구조에는 연결 유형, 연결 상태, 사용자 이름 및 도메인 이름을 포함하여 로컬 컴퓨터와 공유 리소스 간의 연결에 대한 정보가 포함됩니다.
USE_INFO_3 USE_INFO_3 구조에는 연결 유형, 연결 상태, 사용자 이름, 도메인 이름 및 연결 동작을 설명하는 특정 플래그를 포함하여 로컬 컴퓨터와 공유 리소스 간의 연결에 대한 정보가 포함됩니다.
USER_INFO_0 USER_INFO_0 구조에는 사용자 계정 이름이 포함됩니다.
USER_INFO_1 USER_INFO_1 구조에는 계정 이름, 암호 데이터, 권한 수준 및 사용자의 홈 디렉터리에 대한 경로를 포함하여 사용자 계정에 대한 정보가 포함됩니다.
USER_INFO_10 USER_INFO_10 구조에는 계정 이름, 계정과 연결된 주석 및 사용자의 전체 이름을 포함하여 사용자 계정에 대한 정보가 포함됩니다.
USER_INFO_1003 USER_INFO_1003 구조에는 사용자 암호가 포함되어 있습니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.

USER_INFO_1005 USER_INFO_1005 구조에는 사용자 네트워크 계정에 할당할 권한 수준이 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1006 USER_INFO_1006 구조에는 사용자의 홈 디렉터리 경로가 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1007 USER_INFO_1007 구조에는 사용자 네트워크 계정과 연결된 주석이 포함되어 있습니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1008 USER_INFO_1008 구조에는 여러 사용자 네트워크 계정 매개 변수를 정의하는 비트 플래그 집합이 포함되어 있습니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1009 USER_INFO_1009 구조에는 사용자의 로그온 스크립트 파일에 대한 경로가 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1010 USER_INFO_1010 구조에는 사용자 네트워크 계정에 할당된 연산자 권한을 정의하는 비트 플래그 집합이 포함되어 있습니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1011 USER_INFO_1011 구조에는 네트워크 사용자의 전체 이름이 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1012 USER_INFO_1012 구조에는 사용자 주석이 포함되어 있습니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1013 USER_INFO_1013 구조에는 네트워크 계정에 대한 예약된 정보가 포함되어 있습니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1014 USER_INFO_1014 구조에는 사용자가 로그온할 수 있는 워크스테이션의 이름이 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.

USER_INFO_1017 USER_INFO_1017 구조에는 네트워크 사용자 계정에 대한 만료 정보가 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1018 USER_INFO_1018 구조에는 네트워크 사용자 계정에 사용할 수 있는 최대 디스크 공간이 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1020 USER_INFO_1020 구조에는 사용자가 네트워크에 로그온할 수 있는 시간이 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1023 USER_INFO_1023 구조에는 네트워크 로그온 요청을 보내야 하는 서버의 이름이 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1024 USER_INFO_1024 구조에는 네트워크 사용자가 선택한 언어에 대한 국가/지역 코드가 포함되어 있습니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1025 USER_INFO_1025 구조에는 네트워크 사용자가 선택한 언어에 대한 코드 페이지가 포함되어 있습니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1051 USER_INFO_1051 구조에는 사용자 계정과 연결된 RID(상대 ID)가 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1052 USER_INFO_1052 구조에는 네트워크 사용자 프로필에 대한 경로가 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1053 USER_INFO_1053 구조에는 네트워크 계정에 대한 사용자 정보가 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_11 USER_INFO_11 구조에는 계정 이름, 권한 수준, 사용자의 홈 디렉터리 경로 및 기타 사용자 관련 네트워크 통계를 포함하여 사용자 계정에 대한 정보가 포함됩니다.

USER_INFO_2 USER_INFO_2 구조에는 계정 이름, 암호 데이터, 권한 수준, 사용자의 홈 디렉터리에 대한 경로 및 기타 사용자 관련 네트워크 통계를 포함하여 사용자 계정에 대한 정보가 포함됩니다.
USER_INFO_20 계정 이름, 사용자의 전체 이름, 계정과 연결된 주석 및 RID(사용자의 상대 ID)를 포함하여 사용자 계정에 대한 정보를 포함합니다.
USER_INFO_21 USER_INFO_21 구조에는 단방향 암호화된 LAN Manager 2.x 호환 암호가 포함되어 있습니다.
USER_INFO_22 USER_INFO_22 구조에는 계정 이름, 권한 수준, 사용자의 홈 디렉터리에 대한 경로, 단방향 암호화된 LAN Manager 2.x 호환 암호 및 기타 사용자 관련 네트워크 통계를 포함하여 사용자 계정에 대한 정보가 포함됩니다.
USER_INFO_23 계정 이름, 사용자의 전체 이름, 계정과 연결된 주석 및 사용자의 SID(보안 식별자)를 포함하여 사용자 계정에 대한 정보를 포함합니다.
USER_INFO_24 인터넷 ID에 연결된 계정에 대한 사용자 계정 정보를 포함합니다. 이 정보에는 사용자의 인터넷 공급자 이름, 사용자의 인터넷 이름 및 사용자의 SID(보안 식별자)가 포함됩니다.
USER_INFO_3 USER_INFO_3 구조에는 계정 이름, 암호 데이터, 권한 수준, 사용자의 홈 디렉터리 경로, RID(상대 식별자) 및 기타 사용자 관련 네트워크 통계를 포함하여 사용자 계정에 대한 정보가 포함됩니다.
USER_INFO_4 USER_INFO_4 구조에는 계정 이름, 암호 데이터, 권한 수준, 사용자의 홈 디렉터리 경로, SID(보안 식별자) 및 기타 사용자 관련 네트워크 통계를 포함하여 사용자 계정에 대한 정보가 포함됩니다.
USER_MODALS_INFO_0 USER_MODALS_INFO_0 구조에는 보안 데이터베이스의 사용자 및 전역 그룹에 대한 전역 암호 정보(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory가 포함됩니다.
USER_MODALS_INFO_1 USER_MODALS_INFO_1 구조에는 로그온 서버 및 도메인 컨트롤러 정보가 포함됩니다.

USER_MODALS_INFO_1001 USER_MODALS_INFO_1001 구조에는 보안 데이터베이스의 암호에 대한 최소 길이(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory가 포함됩니다.
USER_MODALS_INFO_1002 USER_MODALS_INFO_1002 구조에는 보안 데이터베이스의 암호에 대한 최대 기간(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory가 포함됩니다.
USER_MODALS_INFO_1003 USER_MODALS_INFO_1003 구조에는 보안 데이터베이스의 암호에 대한 최소 기간(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory가 포함됩니다.
USER_MODALS_INFO_1004 USER_MODALS_INFO_1004 구조에는 보안 데이터베이스의 사용자 및 전역 그룹(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory에 대한 강제 로그오프 정보가 포함됩니다.
USER_MODALS_INFO_1005 USER_MODALS_INFO_1005 구조에는 보안 데이터베이스의 사용자 및 전역 그룹에 대한 암호 기록 정보(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory가 포함됩니다.
USER_MODALS_INFO_1006 USER_MODALS_INFO_1006 구조에는 로그온 서버 정보가 포함됩니다.
USER_MODALS_INFO_1007 USER_MODALS_INFO_1007 구조에는 도메인 컨트롤러 정보가 포함됩니다.
USER_MODALS_INFO_2 USER_MODALS_INFO_2 구조에는 SAM(보안 계정 관리자) 도메인 이름 및 식별자가 포함됩니다.
USER_MODALS_INFO_3 USER_MODALS_INFO_3 구조에는 보안 데이터베이스의 사용자 및 전역 그룹(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory에 대한 잠금 정보가 포함됩니다.
USER_OTHER_INFO USER_OTHER_INFO 구조에는 사용자 오류 코드 정보가 포함됩니다. NetAlertRaise 및 NetAlertRaiseEx 함수는 USER_OTHER_INFO 구조를 사용하여 사용자에게 이벤트 또는 관심 조건에 대한 정보를 지정합니다.

WKSTA_INFO_100 플랫폼별 정보, 도메인 및 로컬 컴퓨터의 이름, 운영 체제 관련 정보를 포함하여 워크스테이션 환경에 대한 정보를 포함합니다.
WKSTA_INFO_101 플랫폼별 정보, 도메인 및 로컬 컴퓨터의 이름, 운영 체제 관련 정보를 포함하여 워크스테이션 환경에 대한 정보를 포함합니다. (WKSTA_INFO_101)
WKSTA_INFO_102 플랫폼별 정보, 도메인 및 로컬 컴퓨터의 이름, 운영 체제 관련 정보를 포함하여 워크스테이션 환경에 대한 정보를 포함합니다. (WKSTA_INFO_102)
WKSTA_INFO_502 WKSTA_INFO_502 구조체는 사용되지 않습니다. 이 구조에는 워크스테이션 환경에 대한 정보가 포함됩니다.
WKSTA_TRANSPORT_INFO_0 WKSTA_TRANSPORT_INFO_0 구조에는 WAN(광역 네트워크) 또는 NetBIOS와 같은 워크스테이션 전송 프로토콜에 대한 정보가 포함되어 있습니다.
WKSTA_USER_INFO_0 WKSTA_USER_INFO_0 구조체에는 지정된 워크스테이션에 있는 사용자의 이름이 포함됩니다.
WKSTA_USER_INFO_1 WKSTA_USER_INFO_1 구조에는 특정 워크스테이션과 관련된 사용자 정보가 포함됩니다. 정보에는 현재 사용자의 이름과 워크스테이션에서 액세스하는 도메인이 포함됩니다.
WKSTA_USER_INFO_1101 WKSTA_USER_INFO_1101 구조에는 워크스테이션에서 액세스하는 도메인에 대한 정보가 포함됩니다.

네트워크 정책 서버

아티클2023. 03. 09.

네트워크 정책 서버 기술 개요입니다.

네트워크 정책 서버를 개발하려면 다음 헤더가 필요합니다.

- authif.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 네트워크 정책 서버

열거형

ACCOUNTINGPROPERTIES ACCOUNTINGPROPERTIES 형식의 값은 기록되는 패킷 유형 및 로그 파일의 특성을 제어하는 속성을 열거합니다.
ATTRIBUTEFILTER NPS 사전 특성의 사용 사례입니다.
ATTRIBUTEID ATTRIBUTEID 열거형 형식은 SDO API에서 지원하는 RADIUS 특성을 열거합니다.
ATTRIBUTEINFO ATTRIBUTEINFO 형식의 값은 지정된 특성의 특성을 열거합니다.
ATTRIBUTEPROPERTIES ATTRIBUTEPROPERTIES 형식의 값은 RADIUS 사전 특성에 대한 속성을 열거합니다.
ATTRIBUTERESTRICTIONS ATTRIBUTERESTRICTIONS 열거형 형식의 값은 특정 특성을 사용하는 방법에 대한 제한을 지정합니다.
ATTRIBUTESYNTAX ATTRIBUTESYNTAX 열거형 형식의 각 값은 가능한 특성 구문을 지정합니다.

AUTHENTICATION_TYPE AUTHENTICATION_TYPE 열거 형식의 값은 인증 방법을 지정하는 데 사용됩니다.
CLIENTPROPERTIES CLIENTPROPERTIES 형식의 값은 RADIUS 클라이언트의 속성을 열거합니다. SDO 컴퓨터는 RADIUS 서버입니다.
CONDITIONPROPERTIES CONDITIONPROPERTIES 열거형 형식의 값은 NAP(네트워크 액세스 정책) 조건의 속성을 지정합니다.
DICTIONARYPROPERTIES DICTIONARYPROPERTIES 속성의 값은 특성 사전과 연결된 속성을 열거합니다.
IASCOMMONPROPERTIES IASCOMMONPROPERTIES 열거형 형식의 값은 모든 SDO 개체에 있는 속성을 열거합니다.
IASCOMPONENTPROPERTIES IASCOMPONENTPROPERTIES 열거형 형식의 값은 SDO 개체에 대한 식별자를 열거합니다.
IASDATASTORE IASDATASTORE 열거형의 값은 SDO 데이터에 대한 가능한 스토리지 위치를 나타냅니다.
IASDOMAINTYPE IASDOMAINTYPE 열거형 형식의 값은 SDO 컴퓨터가 도메인의 일부인지 여부와 도메인 유형(있는 경우)을 지정합니다.
IASOSTYPE IASOSTYPE 열거형 형식의 값은 클라이언트에서 인증을 요청하는 운영 체제 유형(SDO 컴퓨터)을 지정합니다.
IASPROPERTIES IASPROPERTIES 열거형 형식의 값은 NPS와 관련된 속성을 열거합니다.
IDENTITY_TYPE MS_ATTRIBUTE_IDENTITY_TYPE 사용할 수 있는 다양한 값입니다.

NAMESPROPERTIES NAMESPROPERTIES 열거형 형식의 값은 네트워크 액세스를 요청하는 사용자의 이름과 관련된 속성을 열거합니다.
NAPPROPERTIES NAPPROPERTIES 열거형 형식의 값은 NAP(네트워크 액세스 정책)에 대한 속성을 지정합니다.
NEW_LOG_FILE_FREQUENCY NEW_LOG_FILE_FREQUENCY 열거형 형식의 값은 새 로그 파일을 만드는 빈도를 지정합니다.
NTEVENTLOGPROPERTIES NTEVENTLOGPROPERTIES 열거형 형식의 값은 NT 이벤트 로그에 기록해야 하는 이벤트 형식을 열거합니다.
NTSAMPROPERTIES NTSAMPROPERTIES 열거형 형식의 값은 NT SAM(보안 계정 관리자)과 관련된 속성을 지정합니다.
POLICYPROPERTIES POLICYPROPERTIES 열거형 형식의 값은 NAP(네트워크 액세스 정책)의 속성을 열거합니다.
PROFILEPROPERTIES PROFILEPROPERTIES 열거형 형식은 프로필과 연결된 속성을 열거합니다.
PROTOCOLPROPERTIES PROTOCOLPROPERTIES 열거형 형식의 값은 인증 프로토콜의 속성을 열거합니다.
RADIUS_ACTION RADIUS_ACTION 형식은 액세스 요청에 대한 응답으로 NPS 확장 DLL이 생성할 수 있는 응답을 열거합니다.
RADIUS_ATTRIBUTE_TYPE RADIUS_ATTRIBUTE_TYPE 형식은 RADIUS 특성에 사용할 수 있는 형식을 열거합니다.
RADIUS_AUTHENTICATION_PROVIDER RADIUS_AUTHENTICATION_PROVIDER 형식은 NPS에서 사용할 수 있는 가능한 인증 공급자를 열거합니다.

RADIUS_CODE RADIUS_CODE 열거형 형식은 가능한 RADIUS 패킷 코드를 열거합니다.
RADIUS_DATA_TYPE RADIUS_DATA_TYPE 형식은 RADIUS 특성 또는 확장 특성에 대해 가능한 데이터 형식을 열거합니다.
RADIUS_EXTENSION_POINT RADIUS_EXTENSION_POINT 열거형 형식은 RadiusExtensionProcess2 함수를 호출할 수 있는 경우 RADIUS 요청 프로세스에서 가능한 지점을 열거합니다.
RADIUS_REJECT_REASON_CODE RADIUS_REJECT_REASON_CODE 열거형은 가능한 RADIUS 패킷 거부 코드를 정의합니다.
RADIUSPROPERTIES RADIUSPROPERTIES 열거형 형식의 값은 Microsoft RADIUS 프로토콜 SDO의 속성을 열거합니다.
RADIUSPROXYPROPERTIES RADIUSPROXYPROPERTIES 열거형 형식의 값은 RADIUS 프록시 서비스와 관련된 속성을 열거합니다.
RADIUSSERVERGROUPPROPERTIES RADIUSSERVERGROUPPROPERTIES 열거형 형식의 값은 RADIUS 서버 그룹의 속성을 열거합니다.
RADIUSSERVERPROPERTIES RADIUSSERVERPROPERTIES 열거형 형식의 값은 SDO 컴퓨터인 RADIUS 서버의 속성을 열거합니다.
REMEDIATIONSERVERGROUPPROPERTIES REMEDIATIONSERVERGROUPPROPERTIES 열거형 형식의 값은 수정 서버 그룹의 속성을 열거합니다.
REMEDIATIONSERVERPROPERTIES REMEDIATIONSERVERPROPERTIES 열거형 형식의 값은 수정 서버의 속성을 열거합니다.
REMEDIATIONSERVERSPROPERTIES REMEDIATIONSERVERSPROPERTIES 열거형 형식의 값은 수정 서버 그룹 집합의 속성을 열거합니다.

SERVICE_TYPE SERVICE_TYPE 열거형 형식의 값은 SDO API에서 관리되는 서비스 유형을 지정합니다.
SHV_COMBINATION_TYPE SHV_COMBINATION_TYPE 열거형 형식은 SHV(시스템 상태 유효성 검사기) 조합의 형식을 지정합니다.
SHVTEMPLATEPROPERTIES SHVTEMPLATEPROPERTIES 열거형 형식의 값은 SHV(시스템 상태 유효성 검사기) 템플릿의 속성을 열거합니다.
USERPROPERTIES USERPROPERTIES 열거형 형식의 값은 SDO API에서 지원하는 사용자 속성을 열거합니다.
VENDORPROPERTIES VENDORPROPERTIES 열거형 형식의 값은 공급업체 컬렉션에 있는 개체의 속성을 지정합니다.

함수

추가 Add 메서드는 SDO(서버 데이터 개체) 컬렉션에 항목을 추가합니다.
적용 Apply 메서드는 ISdo::P utProperty 메서드를 호출하여 변경한 내용을 영구 스토리지에 씁니다.
연결 Attach 메서드는 SDO 컴퓨터에 연결됩니다. SDO 컴퓨터에 연결하는 첫 번째 단계는 SDO API를 사용하여 해당 컴퓨터를 관리하는 것입니다.
CreateAttribute CreateAttribute 메서드는 새 특성 개체를 만들고 IDispatch 인터페이스를 반환합니다.
EnumAttributes EnumAttributes 메서드는 지정된 특성의 값을 검색합니다.

EnumAttributeValues EnumAttributeValues 메서드는 열거 가능한 특성의 값을 검색합니다.
get__NewEnum get__NewEnum 메서드는 SDO(서버 데이터 개체) 속성에 대한 IEnumVARIANT 인터페이스를 검색합니다.
get__NewEnum get__NewEnum 메서드는 SDO(서버 데이터 개체) 컬렉션에 대한 IEnumVARIANT 인터페이스를 검색합니다.
get_Count get_Count 메서드는 컬렉션의 항목 수를 반환합니다.
GetAttachedComputer GetAttachedComputer 메서드는 현재 SDO 컴퓨터로 연결된 컴퓨터의 이름을 검색합니다.
GetAttributeID GetAttributeID 메서드는 지정된 특성의 ID를 검색합니다.
GetAttributeInfo GetAttributeInfo는 지정된 특성에 대한 정보를 검색합니다.
GetDictionarySDO GetDictionarySDO 메서드는 특성 사전 SDO에 대한 인터페이스를 검색합니다.
GetDomainType GetDomainType은 SDO 컴퓨터가 있는 도메인 유형을 검색합니다.
GetOSType GetOSType 메서드는 SDO 컴퓨터에서 실행되는 운영 체제 유형을 검색합니다.
Getproperty GetProperty 메서드는 지정된 속성의 값을 검색합니다.

GetPropertyInfo GetPropertyInfo 메서드는 지정된 속성에 대한 ISdoPropertyInfo 인터페이스에 대한 포인터를 검색합니다.
GetServiceSDO GetServiceSDO 메서드는 지정된 서비스에 대한 SDO(서버 데이터 개체)를 검색합니다.
GetServiceStatus GetServiceStatus 메서드는 SDO를 통해 관리되는 서비스의 상태를 검색합니다.
GetUserSDO GetUserSDO 메서드는 지정된 사용자에 대한 SDO(서버 데이터 개체)에 대한 인터페이스를 검색합니다.
IsDirectoryAvailable IsDirectoryAvailable 메서드는 SDO 컴퓨터에서 Active Directory 서비스를 사용할 수 있는지 여부를 테스트합니다.
IsNameUnique IsNameUnique 메서드는 지정된 이름이 컬렉션에서 고유한지 여부를 테스트합니다.
Item Item 메서드는 컬렉션에서 지정된 항목을 검색합니다.
PRADIUS_EXTENSION_FREE_ATTRIBUTES RadiusExtensionFreeAttributes 함수는 애플리케이션 정의 함수이며 RadiusExtensionProcessEx에서 반환된 특성이 차지하는 메모리를 해제하기 위해 NPS에서 호출됩니다.
PRADIUS_EXTENSION_INIT RadiusExtensionInit 함수는 애플리케이션 정의 함수이며 서비스가 시작되는 동안 NPS에서 호출됩니다. RadiusExtensionInit를 사용하여 확장 DLL에 대한 초기화 작업을 수행합니다.
PRADIUS_EXTENSION_PROCESS RadiusExtensionProcess 함수는 애플리케이션 정의 함수이며 NPS가 NAS(네트워크 액세스 서버)에서 수신하는 각 인증 또는 회계 패킷에 대해 NPS에서 호출됩니다.
PRADIUS_EXTENSION_PROCESS_2 애플리케이션 정의 함수이며 NPS가 수신하는 각 인증 또는 회계 패킷에 대해 NPS에서 호출됩니다.

PRADIUS_EXTENSION_PROCESS_EX RadiusExtensionProcessEx 함수는 애플리케이션 정의 함수이며 NPS가 NAS(네트워크 액세스 서버)에서 수신하는 각 인증 또는 회계 패킷에 대해 NPS에 의해 호출됩니다.
PRADIUS_EXTENSION_TERM RadiusExtensionTerm 함수는 애플리케이션 정의 함수이며 확장 DLL을 언로드하기 전에 NPS에서 호출됩니다. RadiusExtensionTerm을 사용하여 확장 DLL에 대한 정리 작업을 수행합니다.
PutProperty PutProperty 메서드는 지정된 속성의 값을 설정합니다.
다시 로드 Reload 메서드는 기본 데이터 저장소에서 컬렉션의 모든 개체를 다시 로드합니다.
제거 Remove 메서드는 컬렉션에서 지정된 항목을 제거합니다.
Removeall RemoveAll 메서드는 컬렉션에서 모든 항목을 제거합니다.
ResetProperty ResetProperty 메서드는 지정된 속성을 기본값으로 다시 설정합니다.
ResetService ResetService 메서드는 SDO API에서 관리하는 서비스를 다시 설정합니다. 서비스를 다시 설정하면 서비스가 해당 데이터를 새로 고칩니다.
복원 Restore 메서드는 영구 스토리지에서 SDO(서버 데이터 개체) 속성의 값을 다시 로드합니다.
StartService StartService 메서드는 SDO를 통해 관리되는 서비스를 시작합니다.
StopService StopService 메서드는 SDO를 통해 관리되는 서비스를 종료합니다.

인터페이스

ISdo ISdo 인터페이스를 사용하여 SDO(서버 데이터 개체) 정보를 저장, 검색 및 업데이트합니다.
ISdoCollection ISdoCollection 인터페이스를 사용하여 SDO 개체 컬렉션을 조작합니다.
ISdoDictionaryOld ISdoDictionaryOld 인터페이스를 사용하여 RADIUS(원격 액세스 전화 접속 사용자 서비스) 특성의 사전을 조작합니다.
ISdoMachine ISdoMachine 인터페이스를 사용하여 SDO 컴퓨터에 연결하고, SDO 컴퓨터에 대한 정보를 가져오고, 다른 SDO 개체에 대한 인터페이스를 가져옵니다.
ISdoServiceControl ISdoServiceControl 인터페이스를 사용하여 SDO 컴퓨터에서 관리되는 서비스를 제어합니다.

구조체

RADIUS_ATTRIBUTE RADIUS_ATTRIBUTE 구조체는 RADIUS 특성 또는 확장 특성을 나타냅니다.
RADIUS_ATTRIBUTE_ARRAY RADIUS_ATTRIBUTE_ARRAY 구조체는 특성 배열을 나타냅니다.
RADIUS_EXTENSION_CONTROL_BLOCK RADIUS_EXTENSION_CONTROL_BLOCK 구조는 현재 RADIUS 요청에 대한 정보를 제공합니다. 또한 요청과 연결된 특성을 가져오고 요청의 처리 설정을 위한 함수를 제공합니다.
RADIUS_VSA_FORMAT RADIUS_VSA_FORMAT 구조체는 RADIUS 공급업체별 특성의 문자열 부분 형식을 나타냅니다.

오프라인 파일

아티클2023. 03. 09.

오프라인 파일 기술 개요입니다.

오프라인 파일을 개발하려면 다음 헤더가 필요합니다.

- cscapi.h
- cscobj.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 오프라인 파일

열거형

OFFLINEFILES_CACHING_MODE IOfflineFilesCache::IsPathCacheable 및 IOfflineFilesShareInfo::GetShareCachingMode와 같은 메서드에 사용되는 캐싱 모드에 대해 설명합니다.
OFFLINEFILES_COMPARE IOfflineFilesItemFilter::GetTimeFilter 메서드에서 수행할 비교 유형을 지정합니다.
OFFLINEFILES_CONNECT_STATE 오프라인 파일 캐시에 있는 항목의 연결 상태를 설명합니다.
OFFLINEFILES_EVENTS 이벤트 싱크에서 수신하거나 제외할 이벤트를 설명하는 이벤트 식별자 코드입니다.
OFFLINEFILES_ITEM_COPY 항목의 로컬, 원격 또는 원래 복사본을 쿼리할지 여부를 지정합니다.
OFFLINEFILES_ITEM_TIME 사용할 캐시 항목과 연결된 시간 값을 지정합니다.
OFFLINEFILES_ITEM_TYPE 오프라인 파일 캐시에 있는 항목의 형식을 식별합니다.

OFFLINEFILES_OFFLINE_REASON 항목이 오프라인인 이유를 나타냅니다.
OFFLINEFILES_OP_RESPONSE 항목 처리를 계속, 다시 시도 또는 중지할지 여부를 지정합니다.
OFFLINEFILES_PATHFILTER_MATCH 이벤트가 필터와 얼마나 가깝게 일치해야 하는지 지정합니다.
OFFLINEFILES_SETTING_VALUE_TYPE IOfflineFilesSetting::GetValueType 메서드에서 반환된 데이터 형식을 식별합니다.
OFFLINEFILES_SYNC_CONFLICT_RESOLVE IOfflineFilesSyncConflictHandler::ResolveConflict 메서드에서 반환된 충돌 해결 코드를 식별합니다.
OFFLINEFILES_SYNC_OPERATION 동기화 오류가 발생했을 때 수행된 동기화 작업의 유형을 나타냅니다.
OFFLINEFILES_SYNC_STATE 오프라인 파일 항목의 동기화 상태를 설명합니다.

Functions

BackgroundSyncBegin 오프라인 파일 서비스가 백그라운드 동기화 패스를 수행하기 시작했다고 보고합니다.
BackgroundSyncEnd 오프라인 파일 서비스가 백그라운드 동기화 패스를 완료했다고 보고합니다.
시작 작업이 시작되었다고 보고합니다.
CacheEvictBegin 이 메서드는 나중에 사용하도록 예약됩니다. (IOfflineFilesEvents2.CacheEvictBegin)

CacheEvictEnd 이 메서드는 나중에 사용하도록 예약됩니다. (IOfflineFilesEvents2.CacheEvictEnd)
CacheIsCorrupted 이 메서드는 나중에 사용하도록 예약됩니다. (IOfflineFilesEvents.CacheIsCorrupted)
CacheIsFull 이 메서드는 나중에 사용하도록 예약됩니다. (IOfflineFilesEvents.CacheIsFull)
CacheMoved 이 메서드는 나중에 사용하도록 예약됩니다. (IOfflineFilesEvents.CacheMoved)
복제 현재 열거형 상태와 동일한 열거자의 새 인스턴스를 만듭니다. (IEnumOfflineFilesItems.Clone)
복제 현재 열거형 상태와 동일한 열거자의 새 인스턴스를 만듭니다. (IEnumOfflineFilesSettings.Clone)
DataLost 이 이벤트 싱크를 대상으로 하는 하나 이상의 이벤트가 손실되었으며 전달되지 않을 것이라고 보고합니다.
DeleteItems 로컬 캐시에서 파일 및 디렉터리를 삭제합니다.
DeleteItemsForUser 로컬 캐시에서 사용자의 파일 및 디렉터리를 삭제합니다.
DeletePreference 기본 설정이 제거됩니다.
Enabled 이 메서드는 나중에 사용하도록 예약됩니다. (IOfflineFilesEvents.Enabled)
Encrypt 호출 사용자에 대해 캐시된 오프라인 파일 캐시의 콘텐츠를 암호화하거나 암호화 해제합니다.

EncryptionChanged 이 메서드는 나중에 사용하도록 예약됩니다. (IOfflineFilesEvents.EncryptionChanged)
End 작업이 종료되었다고 보고합니다.
EnumItems 이 메서드를 구현하는 캐시 항목에 대한 자식 항목의 열거자를 반환합니다. (IOfflineFilesItemContainer.EnumItems)
EnumItemsEx 이 메서드를 구현하는 캐시 항목에 대한 자식 항목의 열거자를 반환합니다. (IOfflineFilesItemContainer.EnumItemsEx)
EnumSettingObjects IOfflineFilesSetting 인스턴스의 열거자를 만듭니다.
Finditem 캐시에서 특정 파일 또는 디렉터리 항목을 찾습니다. (IOfflineFilesCache.FindItem)
FindItemEx 캐시에서 특정 파일 또는 디렉터리 항목을 찾습니다. (IOfflineFilesCache.FindItemEx)
GetAttributes 항목에 대한 Win32 특성을 검색합니다.
GetConnectState 항목이 온라인인지 아니면 오프라인인지, 오프라인인 경우 그 이유를 결정합니다.
GetDescription 오류를 설명하는 텍스트 문자열을 검색합니다.
GetDiskSpaceInformation 오프라인 파일 캐시에서 사용하는 디스크 공간의 양과 캐시 사용에 적용되는 공간 제한을 검색합니다.

GetEncryptionStatus 오프라인 파일 캐시의 현재 암호화 상태(암호화 또는 암호화되지 않음)를 검색합니다.
GetExcludedEvents 이벤트 싱크에서 수신해서는 안 되는 이벤트를 설명하는 OFFLINEFILES_EVENTS 열거형 값의 배열을 검색합니다.
GetFileAttributes 항목에 대한 Win32 파일 특성을 검색합니다.
GetFileSize 항목의 크기를 검색합니다.
GetFileSize 항목의 크기를 바이트 단위로 검색합니다.
GetFileTimes 항목의 마지막 쓰기 및 변경 시간을 검색합니다.
GetFilterFlags 항목의 플래그 기반 필터링을 제어하는 플래그를 제공합니다.
GetIncludedEvents 이벤트 싱크에서 수신해야 하는 이벤트를 설명하는 OFFLINEFILES_EVENTS 열거형 값의 배열을 검색합니다.
GetItemChangeFlags 동기화 오류와 관련된 동기화 작업 중에 발생한 변경 내용을 설명하는 플래그 집합이 포함된 값을 검색합니다.
GetItemType 항목의 형식을 식별하는 형식 코드를 반환합니다:_server, 공유, 디렉터리 또는 파일.
GetLocalInfo 동기화에 관련된 항목의 로컬 복사본의 파일 시간, 크기 및 특성을 포함하는 IOfflineFilesSyncErrorItemInfo 인터페이스의 인스턴스를 검색합니다.

GetLocation 오프라인 파일 캐시의 현재 정규화된 디렉터리 경로를 검색합니다.
GetName 특정 오프라인 파일 설정과 연결된 이름을 검색합니다.
GetOriginalInfo 동기화에 관련된 항목의 원래 복사본의 파일 시간, 크기 및 특성을 포함하는 IOfflineFilesSyncErrorItemInfo 인터페이스의 인스턴스를 검색합니다.
GetParentItem 항목의 부모에 대한 IOfflineFilesItem 인터페이스를 검색합니다.
GetPath 오프라인 파일 캐시의 항목에 대한 정규화된 UNC 경로 문자열을 검색합니다.
GetPathFilter 이 이벤트 싱크에 전달되어야 하는 경로 기반 이벤트를 설명하는 UNC 경로 문자열 및 범위 표시기를 검색합니다.
GetPatternFilter 항목 이름 패턴에 따라 열거된 항목을 제한하는 필터 패턴 문자열을 제공합니다.
GetPolicy 특정 오프라인 파일 설정과 연결된 정책을 검색합니다.
GetPolicyScope 이 설정과 연결된 정책의 범위를 검색합니다.
GetPreference 특정 오프라인 파일 설정과 연결된 컴퓨터별 또는 사용자별 기본 설정을 검색합니다.
GetPreferenceScope 이 설정과 연결된 기본 설정의 범위를 나타냅니다.
GetRawData 오류와 연결된 내부 데이터가 포함된 바이트 블록을 검색합니다.

GetRemoteInfo 동기화에 관련된 항목의 원격 복사본의 파일 시간, 크기 및 특성을 포함하는 IOfflineFilesSyncErrorItemInfo 인터페이스의 인스턴스를 검색합니다.
GetSettingObject 특정 오프라인 파일 설정을 나타내는 개체를 만듭니다.
GetShareCachingMode 항목에 가장 가까운 상위 공유의 캐싱 모드 구성을 검색합니다.
GetShareItem 항목에 가장 가까운 상위 항목 공유를 나타내는 캐시 항목을 찾습니다.
GetSyncOperation 오류가 발생했을 때 수행된 동기화 작업의 유형을 나타내는 값을 검색합니다.
GetTimeFilter 시간에 따라 항목 필터링을 제어하는 시간-값 비교 의미 체계를 제공합니다.
GetTimes 항목과 연결된 시간 값을 검색합니다.
GetValue 특정 오프라인 파일 설정의 값을 검색합니다.
GetValueType 특정 오프라인 파일 설정의 데이터 형식을 검색합니다.
InfoAvailable 동기화 중에 항목의 로컬, 원격 또는 원본 복사본에 대한 정보를 얻었는지 여부를 나타냅니다.
InfoEnumerated 동기화 중에 항목의 로컬, 원격 또는 원래 복사본에 대한 정보를 쿼리했는지 여부를 나타냅니다.
IsCreatedOffline 오프라인으로 작업하는 동안 오프라인 파일 캐시에 항목이 만들어졌는지 여부를 확인합니다.

IsDeletedOffline 오프라인으로 작업하는 동안 오프라인 파일 캐시에서 항목이 삭제되었는지 여부를 확인합니다.
IsDirty 오프라인 파일 캐시의 항목이 수정되었는지 여부를 확인합니다.
IsEncrypted 오프라인 파일 캐시의 항목이 암호화되는지 여부를 확인합니다.
IsGhosted 항목이 고스트인지 여부를 확인합니다.
IsLocallyModifiedAttributes 오프라인으로 작업하는 동안 항목의 특성 중 하나 이상이 수정되었는지 여부를 확인합니다.
IsLocallyModifiedData 오프라인으로 작업하는 동안 항목의 데이터가 수정되었는지 여부를 확인합니다.
IsLocallyModifiedTime 오프라인으로 작업하는 동안 항목의 시간 값 중 하나 이상이 수정되었는지 여부를 확인합니다.
IsMarkedForDeletion 오프라인 파일 캐시에서 항목이 삭제되었는지 여부를 확인합니다.
IsPartlyPinned 항목이 부분적으로 고정되었는지 여부를 결정합니다.
IsPathCacheable 지정한 UNC 경로가 오프라인 파일 캐시에 있는지 여부를 확인합니다.
IsPinned 항목이 고정되어 있는지 여부를 확인합니다.
IsPinnedForComputer 그룹 정책 컴퓨터의 모든 사용자에 대해 항목이 고정되었는지 여부를 확인합니다.

IsPinnedForFolderRedirection 항목이 폴더 리디렉션에 의해 고정되었는지 여부를 확인합니다.
IsPinnedForUser 사용자가 항목을 고정했는지 여부를 확인합니다.
IsPinnedForUserByPolicy 그룹 정책 항목이 사용자에 대해 고정되었는지 여부를 확인합니다.
IsShareDfsJunction 공유 항목이 DFS 접합인지 아니면 서버의 공유 폴더인지를 확인합니다.
IsSparse 오프라인 파일 캐시의 항목이 드물게 캐시되는지 여부를 확인합니다.
IsSuspended 항목이 일시 중단되었는지 여부를 확인합니다.
IsTransparentlyCached 항목이 투명하게 캐시되는지 여부를 확인합니다.
ItemAddedToCache 항목이 오프라인 파일 캐시에 추가되었음을 보고합니다.
ItemAvailableOffline 원격 복사본을 사용할 수 없게 되면 오프라인 파일 캐시의 항목을 오프라인에서 사용할 수 있다고 보고합니다.
ItemBegin 항목에 대한 작업이 시작되었다고 보고합니다.
ItemDeletedFromCache 오프라인 파일 캐시에서 항목이 제거되었다고 보고합니다.
ItemDisconnected 오프라인 파일 캐시의 항목이 온라인에서 오프라인으로 전환되었다고 보고합니다.

ItemModified 오프라인 파일 캐시의 항목이 수정되었음을 보고합니다.
ItemNotAvailableOffline 원격 복사본을 사용할 수 없게 되면 오프라인 파일 캐시의 항목을 더 이상 오프라인으로 사용할 수 없다고 보고합니다.
ItemNotPinned 오프라인 파일 캐시의 항목이 더 이상 고정되지 않는다고 보고합니다.
ItemPinned 이제 오프라인 파일 캐시의 항목이 고정되어 원격 복사본을 사용할 수 없게 되면 오프라인으로 사용할 수 있다고 보고합니다.
ItemReconnectBegin 오프라인 파일 서비스가 모든 오프라인 범위를 다시 연결하려고 시도하기 시작했다고 보고합니다.
ItemReconnected 오프라인 파일 캐시의 항목이 오프라인에서 온라인으로 전환되었다고 보고합니다.
ItemReconnectEnd 오프라인 파일 서비스가 모든 오프라인 범위를 다시 연결하려고 시도했다고 보고합니다.
ItemRenamed 오프라인 파일 캐시의 항목 경로 이름이 변경되었음을 보고합니다.
ItemResult 작업 중에 항목이 처리되었다고 보고합니다.
LocalDirtyByteCount 로컬 오프라인 파일 캐시의 연결된 파일에 대한 동기화되지 않은("더티") 데이터의 양을 검색합니다.
NetTransportArrived 오프라인 파일 기능이 네트워크 전송의 도착을 감지했다고 보고합니다.

다음 열거형에서 다음 항목을 검색하고 열거자를 이동합니다. (IEnumOfflineFilesItems.Next)
다음 열거형에서 다음 항목을 검색하고 열거자를 이동합니다. (IEnumOfflineFilesSettings.Next)
NoNetTransports 오프라인 파일 기능이 네트워크 전송을 사용할 수 없음을 감지했다고 보고합니다.
OfflineFilesEnable 오프라인 파일 기능을 사용하거나 사용하지 않도록 설정합니다.
OfflineFilesQueryStatus 오프라인 파일 기능을 사용할 수 있는지 여부와 활성화되어 있는지 여부를 결정합니다.
OfflineFilesQueryStatusEx 오프라인 파일 기능을 사용할 수 있는지 여부와 활성화되어 있고 사용 가능한지 여부를 결정합니다. 이 함수는 추가 출력 매개 변수가 있다는 점을 제외하고 OfflineFilesQueryStatus 함수와 동일합니다.
OfflineFilesStart 오프라인 파일 서비스를 시작합니다.
고정 파일, 디렉터리 및 네트워크 공유 폴더를 고정합니다.
Ping 이 이벤트는 등록된 모든 이벤트 구독자에게 주기적으로 전달됩니다.
PolicyChangeDetected 오프라인 파일 서비스에서 그룹 정책 제어되는 설정 값 중 하나 이상의 변경 내용을 감지했다고 보고합니다.
PreferenceChangeDetected 오프라인 파일 서비스에서 그룹 정책 제어되지 않는 설정 값 중 하나 이상의 변경 내용을 감지했다고 보고합니다.

PrefetchFileBegin 파일 프리페치 작업이 시작되었다고 보고합니다.
PrefetchFileEnd 파일 프리페치 작업이 종료되었다고 보고합니다.
ProcessAdminPinPolicy 오프라인 파일이 "관리적으로 할당된 오프라인 파일" 그룹 정책을 처리하도록 합니다.
QueryAbort 작업이 취소되어야 하는지 여부를 확인하기 위해 긴 작업 중에 호출될 수 있습니다.
새로 고침 오프라인 파일 캐시에서 다시 읽어 개체에 캐시된 모든 데이터를 새로 고칩니다.
RemoteDirtyByteCount 이 메서드는 나중에 사용하도록 예약됩니다. (IOfflineFilesDirtyInfo.RemoteDirtyByteCount)
RenameItem 캐시의 항목 이름을 바꿉니다.
RenameItemEx 캐시의 항목 이름을 바꿉니다. 이 메서드는 이름 바꾸기 작업을 즉시 수행하려고 한다는 점을 제외하고 IOfflineFilesCache::RenameItem 메서드와 동일합니다.
재설정 열거형을 시작 부분으로 다시 설정합니다. (IEnumOfflineFilesItems.Reset)
재설정 열거형을 시작 부분으로 다시 설정합니다. (IEnumOfflineFilesSettings.Reset)
ResolveConflict 동기화 충돌에 대한 해결 결정을 제공합니다.
SetConnectState 항목의 연결 상태를 설정합니다.

SetDiskSpaceLimits 오프라인 파일 캐시의 디스크 공간 사용 제한을 설정합니다.
SetPreference 오프라인 파일 설정과 연결된 컴퓨터별 또는 사용자별 기본 설정을 지정합니다.
SettingsChangesApplied 오프라인 파일 서비스에서 그룹 정책 또는 기본 설정 값에서 검색된 변경 내용을 적용했다고 보고합니다.
skip 열거형에서 지정한 다음 개수의 요소를 건너뜁니다. (IEnumOfflineFilesItems.Skip)
skip 열거형에서 지정한 다음 개수의 요소를 건너뜁니다. (IEnumOfflineFilesSettings.Skip)
SuspendRoot 공유 루트 또는 디렉터리 트리를 일시 중단하거나 해제합니다.
SyncBegin 오프라인 파일 캐시가 동기화 작업을 시작했다고 보고합니다.
SyncConflictRecAdded 동기화 충돌이 감지되어 동기화 충돌 로그에 기록되었음을 보고합니다.
SyncConflictRecRemoved 동기화 충돌이 더 이상 존재하지 않으며 해당 레코드가 동기화 충돌 로그에서 제거되었음을 보고합니다.
SyncConflictRecUpdated 동기화 충돌이 감지되었고 충돌 레코드가 동기화 충돌 로그에 이미 있음을 보고합니다.
SyncEnd 오프라인 파일 캐시가 동기화 작업을 종료했다고 보고합니다.
SyncFileResult 특정 파일을 동기화한 결과를 보고합니다.

동기화 오프라인 파일 캐시의 파일 및 디렉터리를 해당 네트워크 공유 폴더의 해당 복사본과 동기화합니다.
SyncItemBegin 항목에 대한 동기화 작업이 시작되었다고 보고합니다.
SyncItemResult 동기화 작업 중에 항목이 처리되었음을 보고합니다.
TransitionOffline 가능하면 항목을 오프라인으로 전환합니다.
TransitionOnline 가능하면 온라인으로 항목을 전환합니다.
TransparentCacheItemNotify 투명하게 캐시된 항목에 대해 작업이 수행되었다고 보고합니다.
고정 해제 오프라인 파일 캐시에서 파일, 디렉터리 및 네트워크 공유 폴더의 고정을 해제합니다.

인터페이스

IEnumOfflineFilesItems IOfflineFilesItem 인터페이스 포인터의 컬렉션을 나타냅니다.
IEnumOfflineFilesSettings 오프라인 파일 서비스와 연결된 설정 개체를 열거합니다.
IOfflineFilesCache 오프라인 파일 캐시를 관리하는 데 사용됩니다.

IOfflineFilesCache2 RenameItemEx 메서드를 구현합니다.
IOfflineFilesChangeInfo 오프라인으로 작업하는 동안 항목에 대한 로컬 변경 내용과 관련된 정보를 나타냅니다.
IOfflineFilesConnectionInfo 오프라인 파일의 온라인 오프라인 전환 동작과 관련된 쿼리 및 작업 기능을 제공합니다.
IOfflineFilesDirectoryItem 오프라인 파일 캐시의 디렉터리 항목을 나타냅니다.
IOfflineFilesDirtyInfo 오프라인 파일 캐시의 동기화되지 않은("더티") 파일에 대한 정보를 나타냅니다.
IOfflineFilesErrorInfo 오류와 연결된 텍스트 설명 및 원시 데이터 블록을 제공합니다.
IOfflineFilesEvents 오프라인 파일과 관련된 중요한 이벤트를 보고하는 데 사용됩니다.
IOfflineFilesEvents2 오프라인 파일과 연결된 추가 이벤트를 보고하는 데 사용됩니다.
IOfflineFilesEvents3 투명하게 캐시된 항목과 연결된 이벤트를 보고하는 데 사용됩니다.
IOfflineFilesEventsFilter 게시된 이벤트의 수신자가 수신하는 이벤트 인스턴스 수를 제한하는 메커니즘을 제공합니다.
IOfflineFilesFileItem 오프라인 파일 캐시의 파일 항목을 나타냅니다.
IOfflineFilesFileSysInfo 오프라인 파일 캐시의 파일 시스템 항목과 연결된 표준 정보를 나타냅니다.

IOfflineFilesGhostInfo 오프라인 파일 캐시에 있는 항목의 고스팅 상태를 나타냅니다.
IOfflineFilesItem 오프라인 파일 캐시의 단일 항목을 나타냅니다.
IOfflineFilesItemContainer 오프라인 파일 캐시의 항목 열거 기능에 액세스하는 데 사용됩니다.
IOfflineFilesItemFilter 열거형에 적용할 필터의 인스턴스를 나타냅니다.
IOfflineFilesPinInfo 오프라인 파일 캐시에 있는 항목의 고정 상태를 나타냅니다.
IOfflineFilesPinInfo2 오프라인 파일 캐시의 항목이 부분적으로 고정되어 있는지 여부를 확인하는 메서드를 정의합니다.
IOfflineFilesProgress 긴 오프라인 파일 작업의 호출자에게 진행률을 다시 보고하는 데 사용됩니다. (IOfflineFilesProgress)
IOfflineFilesServerItem 오프라인 파일 캐시의 서버 항목을 나타냅니다.
IOfflineFilesSetting 오프라인 파일 서비스의 동작을 제어하는 설정을 나타냅니다.
IOfflineFilesShareInfo 캐시된 항목에 대한 공유 관련 정보를 제공합니다.
IOfflineFilesShareItem 오프라인 파일 캐시의 공유 항목을 나타냅니다.
IOfflineFilesSimpleProgress 긴 오프라인 파일 작업의 호출자에게 진행률을 다시 보고하는 데 사용됩니다. (IOfflineFilesSimpleProgress)

IOfflineFilesSuspend 오프라인 파일 캐시에서 공유 루트 또는 디렉터리 트리를 일시 중단하거나 해제합니다.
IOfflineFilesSuspendInfo 항목이 일시 중단되었는지 여부와 일시 중단된 루트인 경우 항목을 결정합니다.
IOfflineFilesSyncConflictHandler IOfflineFilesCache::Synchronize 메서드를 호출하는 클라이언트에서 동기화 충돌이 감지될 때 충돌 해결 전략을 규정하는 데 사용됩니다.
IOfflineFilesSyncErrorInfo 동기화 오류가 발생한 항목에 대한 세부 정보를 전달하기 위해 IOfflineFilesSyncProgress::SyncItemResult 메서드와 함께 제공됩니다.
IOfflineFilesSyncErrorItemInfo 동기화 오류와 관련된 항목에 대한 파일 특성, 시간 정보 및 파일 크기를 제공합니다.
IOfflineFilesSyncProgress 동기화 및 동기화 관련 작업 중에 호출자에게 진행률을 다시 보고하는 데 사용됩니다.
IOfflineFilesTransparentCacheInfo 투명하게 캐시된 항목과 관련된 정보를 나타냅니다.

설정 상태 및 알림에 대한 OOBE API

아티클2023. 08. 22.

설정 상태 및 알림 기술에 대한 OOBE API 개요입니다.

설정 상태 및 알림을 위한 OOBE API를 개발하려면 다음 헤더가 필요합니다.

- oobenotification.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 설정 상태 및 알림에 대한 OOBE API

Functions

OOBE_COMPLETED_CALLBACK RegisterWaitUntilOOBECompleted 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
OOBEComplete OOBE(Windows 시작)가 완료되었는지 여부를 확인합니다.
RegisterWaitUntilOOBECompleted OOBE(Windows 시작)가 완료되면 호출할 콜백을 등록합니다.
UnregisterWaitUntilOOBECompleted RegisterWaitUntilOOBECompleted를 통해 이전에 등록된 콜백의 등록을 취소합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Opengl

아티클2023. 08. 22.

OpenGL 기술 개요입니다.

OpenGL 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Opengl

Functions

PixelFormat 선택 ChoosePixelFormat 함수는 디바이스 컨텍스트에서 지원하는 적절한 픽셀 형식을 지정된 픽셀 형식 사양과 일치시키려고 시도합니다.
DescribePixelFormat DescribePixelFormat 함수는 hdc와 연결된 디바이스의 iPixelFormat으로 식별되는 픽셀 형식에 대한 정보를 가져옵니다. 함수는 해당 픽셀 형식 데이터를 사용하여 ppfd가 가리키는 PIXELFORMATDESCRIPTOR 구조체의 멤버를 설정합니다.
GetEnhMetaFilePixelFormat GetEnhMetaFilePixelFormat 함수는 향상된 메타파일의 픽셀 형식 정보를 검색합니다.
GetPixelFormat GetPixelFormat 함수는 지정된 디바이스 컨텍스트의 현재 선택된 픽셀 형식의 인덱스를 가져옵니다.
SetPixelFormat SetPixelFormat 함수는 지정된 디바이스 컨텍스트의 픽셀 형식을 iPixelFormat 인덱스에 지정된 형식으로 설정합니다.
SwapBuffers 지정한 디바이스 컨텍스트에서 참조하는 창의 현재 픽셀 형식에 백 버퍼가 포함된 경우 SwapBuffers 함수는 전면 및 후면 버퍼를 교환합니다.
wglCopyContext

wglCopyContext 함수는 선택한 렌더링 상태 그룹을 한 OpenGL 렌더링 컨텍스트에서 다른 OpenGL 렌더링 컨텍스트로 복사합니다.
wglCreateContext wglCreateContext 함수는 hdc에서 참조하는 디바이스에서 그리기에 적합한 새 OpenGL 렌더링 컨텍스트를 만듭니다. 렌더링 컨텍스트는 디바이스 컨텍스트와 동일한 픽셀 형식을 입니다.
wglCreateLayerContext wglCreateLayerContext 함수는 디바이스 컨텍스트에서 지정된 레이어 평면에 그리기 위한 새 OpenGL 렌더링 컨텍스트를 만듭니다.
wglDeleteContext wglDeleteContext 함수는 지정된 OpenGL 렌더링 컨텍스트를 삭제합니다.
wglDescribeLayerPlane wglDescribeLayerPlane 함수는 지정된 픽셀 형식의 레이어 평면에 대한 정보를 가져옵니다.
wglGetCurrentContext wglGetCurrentContext 함수는 호출 스레드의 현재 OpenGL 렌더링 컨텍스트에 대한 핸들을 가져옵니다.
wglGetCurrentDC wglGetCurrentDC 함수는 호출 스레드의 현재 OpenGL 렌더링 컨텍스트와 연결된 디바이스 컨텍스트에 대한 핸들을 가져옵니다.
wglGetLayerPaletteEntries 지정된 디바이스 컨텍스트에 대해 지정된 색 인덱스 계층 평면에서 색상표 항목을 검색합니다.
wglGetProcAddress wglGetProcAddress 함수는 현재 OpenGL 렌더링 컨텍스트에 사용할 OpenGL 확장 함수의 주소를 반환합니다.
wglMakeCurrent wglMakeCurrent 함수는 지정된 OpenGL 렌더링 컨텍스트를 호출 스레드의 현재 렌더링 컨텍스트로 만듭니다.
wglRealizeLayerPalette wglRealizeLayerPalette 함수는 지정된 색인 계층 평면의 색상표 항목을 실제 색상표로 매핑하거나 RGBA 레이어 평면의 팔레트를 초기화합니다.

wglSetLayerPaletteEntries 지정된 디바이스 컨텍스트에 대해 지정된 색 인덱스 계층 평면의 색상표 항목을 설정합니다.
wglShareLists wglShareLists 함수를 사용하면 여러 OpenGL 렌더링 컨텍스트에서 단일 표시 목록 공간을 공유할 수 있습니다.
wglSwapLayerBuffers wglSwapLayerBuffers 함수는 지정된 디바이스 컨텍스트에서 참조하는 창의 오버레이, 언더레이 및 기본 평면에서 전면 및 후방 버퍼를 교환합니다.
wglUseFontBitmapsA wglUseFontBitmaps 함수는 현재 OpenGL 렌더링 컨텍스트에서 사용할 비트맵 표시 목록 집합을 만듭니다. (ANSI)
wglUseFontBitmapsW wglUseFontBitmaps 함수는 현재 OpenGL 렌더링 컨텍스트에서 사용할 비트맵 표시 목록 집합을 만듭니다. (유니코드)
wglUseFontOutlinesA wglUseFontOutlines 함수는 현재 렌더링 컨텍스트에 사용할 디바이스 컨텍스트의 현재 선택된 개요 글꼴의 각 문자 모양에 대해 하나씩 표시 목록 집합을 만듭니다. (ANSI)
wglUseFontOutlinesW wglUseFontOutlines 함수는 현재 렌더링 컨텍스트에 사용할 디바이스 컨텍스트의 현재 선택된 개요 글꼴의 각 문자 모양에 대해 하나씩 표시 목록 집합을 만듭니다. (유니코드)

구조체

GLYPHMETRICSFLOAT GLYPHMETRICSFLOAT 구조체에는 문자 셀의 문자 모양 배치 및 방향에 대한 정보가 포함되어 있습니다.
LAYERPLANEDESCRIPTOR LAYERPLANEDESCRIPTOR 구조체는 그리기 표면의 픽셀 형식을 설명합니다.

PIXELFORMATDESCRIPTOR PIXELFORMATDESCRIPTOR 구조체는 그리기 표면의 픽셀 형식을 설명합니다.
POINTFLOAT POINTFLOAT 구조체에는 점의 x 및 y 좌표가 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

작업 레코더

아티클2023. 08. 22.

운영 레코더 기술 개요입니다.

작업 레코더 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 작업 레코더

Functions

OperationEnd 애플리케이션이 작업을 종료하려고 했음을 시스템에 알 수 있습니다.
OperationStart 애플리케이션이 작업을 시작하려고 했음을 시스템에 알 수 있습니다.

구조체

OPERATION_END_PARAMETERS 이 구조체는 OperationEnd 함수에서 사용됩니다.
OPERATION_START_PARAMETERS 이 구조체는 OperationStart 함수에서 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

기타 디바이스 기술

아티클2023. 08. 22.

기타 디바이스 기술 기술 개요

다른 디바이스 기술을 개발하려면 다음 헤더가 필요합니다.

- functiondiscoveryapi.h
- functiondiscoveryconstraints.h
- functiondiscoveryprovider.h
- pnpxassoc.h
- wsdattachment.h
- wsdbase.h
- wsdclient.h
- wsddisco.h
- wsdhost.h
- wsdtypes.h
- wsdutil.h
- wsdxml.h
- wsdxmldom.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 함수 검색
- 디바이스의 웹 서비스

열거형

PropertyConstraint 함수 인스턴스를 검색하는 데 사용되는 필터 조건을 한정합니다.
QueryUpdateAction 지정된 함수 instance 함수 검색이 수행하는 작업 유형을 나타냅니다. 이 정보는 클라이언트 프로그램의 변경 알림 처리기에서 사용됩니다.
SystemVisibilityFlags 함수 instance 데이터의 표시 여부를 결정합니다.

WSD_CONFIG_PARAM_TYPE WSD_CONFIG_PARAM 구조에 저장된 데이터의 종류를 지정합니다.
WSD_PROTOCOL_TYPE 포트에서 지원하는 프로토콜 유형을 식별합니다.
WSDEventType 세션 계층에서 생성되는 이벤트의 유형을 식별합니다.
WSDUdpMessageType UDP 메시지의 유형을 식별합니다.

Functions

중단 비동기 작업을 중단합니다.
중단 첨부 파일 MIME 데이터 스트림의 데이터 전송을 중단합니다.
AbortAsyncOperation 보류 중인 비동기 작업을 중단합니다.
추가 컬렉션에 함수 instance 추가합니다.
추가 새로 발표된 검색 호스트(Hello 메시지에서) 또는 사용자가 시작한 쿼리와 일치하는 항목에 대한 정보를 제공합니다.
AddDynamicService 들어오는 요청에 대한 서비스 개체를 등록하지만 디바이스 호스트 메타데이터에 서비스를 추가하지는 않습니다. 일시적(동적) 서비스에 사용됩니다.

AddInstance 함수 instance 만들거나 수정합니다.
AddNamespace XML 컨텍스트에서 네임스페이스를 나타내는 개체를 만듭니다.
AddNameToNamespace XML 컨텍스트에서 네임스페이스의 이름을 나타내는 개체를 만듭니다.
AddPropertyConstraint 쿼리에 속성 제약 조건을 추가합니다.
AddQueryConstraint 쿼리 제약 조건을 쿼리에 추가합니다.
연결 연결 데이터베이스 항목을 연결된 것으로 표시합니다.
연결 연결 데이터베이스 항목을 연결된 것으로 표시하고 적절한 알림을 보냅니다.
AsyncOperationComplete 비동기 작업이 완료되었음을 나타냅니다.
연결 검색 공급자에 콜백 인터페이스를 연결합니다.
BeginGetMetadata 메타데이터에 대한 비동기 요청을 보냅니다.
BeginGetMetadata 원격 서비스를 사용하여 비동기 메타데이터 교환 요청을 시작합니다.
BeginGetStatusForMultipleOperations 현재 상태 검색하는 비동기 작업을 시작합니다.
BeginRenewMultipleOperations

새 기간을 제출하여 기존 알림 구독의 컬렉션을 갱신하는 비동기 작업을 초기화합니다.
BeginSubscribeToMultipleOperations 알림 또는 요청/응답 이벤트 컬렉션을 구독하는 비동기 작업을 초기화합니다.
BeginUnsubscribeToMultipleOperations 알림 또는 요청/응답 이벤트 컬렉션에 대한 구독에 대한 비동기 취소 요청을 초기화합니다.
지우기 SOAP over-HTTP 전송에 사용되는 HTTP 헤더를 지웁니다.
닫기 현재 첨부 파일 MIME 데이터 스트림을 닫습니다. (IWSDInboundAttachment.Close)
닫기 현재 첨부 파일 MIME 데이터 스트림을 닫습니다. (IWSDOutboundAttachment.Close)
CreateFunctionInstanceCollection 함수 instance 컬렉션을 만듭니다.
CreateInstance 함수 instance 만듭니다.
CreateInstance instance 새 함수를 만듭니다.
CreateInstanceCollectionQuery 특정 함수 인스턴스의 컬렉션에 대한 쿼리를 만듭니다.
CreateInstanceQuery 특정 함수 instance 대한 쿼리를 만듭니다.
CreatePropertyStore 공급자가 메모리 내 속성 저장소 구현을 다시 사용할 수 있도록 합니다.
삭제 컬렉션에서 지정된 함수 instance 제거합니다.

삭제 연결 데이터베이스에서 항목을 제거합니다.
삭제 연결 데이터베이스에서 항목을 제거하고 적절한 알림을 보냅니다.
DeleteAll 컬렉션에서 모든 함수 인스턴스를 제거합니다.
역직렬화 주소를 구문 분석하고 구성 요소 부분의 유효성을 검사한 다음 개체에 저장합니다.
분리 검색 공급자에서 콜백 인터페이스를 분리합니다.
EndGetMetadata 메타데이터에 대한 비동기 요청을 종료합니다.
EndGetMetadata 비동기 메타데이터 교환 요청을 완료하고 응답에서 서비스 메타데이터를 검색합니다.
EndGetStatusForMultipleOperations 현재 상태 검색하는 비동기 작업을 완료합니다.
EndQuery 공급자가 실행 중인 쿼리를 종료합니다.
EndRenewMultipleOperations 새 기간을 제출하여 기존 알림 구독의 컬렉션을 갱신하는 비동기 작업을 완료합니다.
EndSubscribeToMultipleOperations 알림 또는 요청/응답 이벤트 컬렉션을 구독하는 비동기 작업을 완료합니다.
EndUnsubscribeToMultipleOperations 알림 또는 요청/응답 이벤트 컬렉션에 대한 구독에 대한 비동기 취소 요청을 완료합니다.
실행

IFunctionDiscovery::CreateInstanceCollectionQuery로 정의된 쿼리를 수행합니다.
실행 IFunctionDiscovery::CreateInstanceQuery에서 정의한 쿼리를 수행합니다.
FaultRequest 지정된 요청 컨텍스트와 일치하는 오류를 보냅니다.
가져오기 컬렉션에서 지정된 함수 instance 및 해당 인덱스 를 가져옵니다.
가져오기 속성 키로 지정된 속성 제약 조건의 이름과 값을 가져옵니다.
가져오기 지정된 쿼리 제약 조건의 값을 이름으로 가져옵니다.
GetAlias 검색 주소의 별칭을 가져옵니다.
GetAllMetadata 이 디바이스에 대한 모든 메타데이터를 검색합니다.
GetAsyncState 비동기 작업의 상태를 가져옵니다.
GetAt 지정된 인덱스에서 속성 키를 가져옵니다.
GetAuthType GetAuthType 메서드는 클라이언트 인증 중에 사용되는 HTTP 인증 체계를 검색합니다.
GetCategory 함수 instance 대한 범주 및 하위 범주 문자열을 가져옵니다.
GetClientAccessToken GetClientAccessToken 메서드는 클라이언트를 인증하거나 가장하는 데 사용할 수 있는 클라이언트 액세스 토큰을 검색합니다.

GetClientCertificate 클라이언트 인증서를 가져옵니다.
GetContext 현재 트랜잭션에 대한 프라이빗 전송 컨텍스트를 검색합니다.
GetCount 컬렉션의 함수 인스턴스 수를 가져옵니다.
GetCount 속성 저장소의 속성 수를 가져옵니다.
GetCount 컬렉션의 항목 수를 가져옵니다. (IProviderPropertyConstraintCollection.GetCount)
GetCount 컬렉션의 항목 수를 가져옵니다. (IProviderQueryConstraintCollection.GetCount)
GetEndpointProxy 비동기 작업에 대한 엔드포인트 프록시를 검색합니다.
GetEndpointProxy 디바이스에 대한 엔드포인트 프록시를 검색합니다.
GetEndpointProxy 디바이스에 대한 엔드포인트 프록시를 가져옵니다.
GetEndpointReference 원격 디바이스의 엔드포인트를 참조하는 WS-Addressing 주소를 검색합니다.
GetErrorInfo 마지막 오류에 대한 정보를 검색합니다.
GetEvent 이벤트의 결과를 포함하는 WSD_EVENT 구조를 검색합니다.
GetExclusive

소켓이 배타적 모드인지 여부를 확인합니다.
GetExtendedDiscoXML SOAP 메시지의 헤더 또는 본문에 제공된 사용자 지정 또는 확장 가능한 데이터를 검색합니다.
GetFaultInfo 마지막으로 받은 오류에 대한 정보를 검색합니다.
GetHostMetadata 디바이스의 기능과 호스트하는 서비스를 설명하는 디바이스에 대한 클래스별 메타데이터를 검색합니다.
GetID 함수 instance 대한 식별자 문자열을 가져옵니다.
GetID 현재 트랜잭션의 전송 ID를 검색합니다.
GetInboundHttpHeaders 인바운드 SOAP over-HTTP 전송에 사용되는 현재 HTTP 헤더를 검색합니다.
Getinstance 식별자에 따라 지정된 함수 instance 가져옵니다.
GetInstanceCollection 범주 및 하위 범주에 따라 함수 인스턴스의 지정된 컬렉션을 가져옵니다.
GetInstanceId 이 메시지의 instance 식별자를 검색합니다.
GetLocalAddress 메시지를 받은 로컬 주소를 나타내는 제네릭 주소 개체를 검색합니다.
GetLocalInterfaceGUID 메시지가 수신된 로컬 네트워크 인터페이스의 GUID를 검색합니다.
GetLocalTransportAddress 로컬 전송(IP) 주소의 문자열 표현을 검색합니다.

GetLowerParameters 프로토콜 스택의 이 계층 아래 계층에서 메시지 매개 변수를 검색합니다.
GetMappedAccessToken 매핑된 액세스 토큰을 가져옵니다.
GetMessageType 이 UDP 주소 구성에 대한 메시지 유형을 가져옵니다.
GetMetadata 개체에 대한 메타데이터를 검색합니다.
GetMetadataVersion 이 메시지의 메타데이터 버전을 검색합니다.
GetOutboundHttpHeaders 아웃바운드 SOAP over-HTTP 전송에 사용되는 현재 HTTP 헤더를 검색합니다.
GetPath 이 주소의 URI 경로를 가져옵니다.
GetPort 이 전송 주소와 연결된 IP 포트 번호를 가져옵니다.
GetProbeResolveTag 검색된 이 서비스 개체에 해당하는 검색 태그를 검색합니다.
GetPropertyConstraints 현재 속성 제약 조건을 검색합니다.
GetProviderInstanceID 공급자 instance 대한 식별자 문자열을 가져옵니다.
GetQueryConstraints 현재 쿼리 제약 조건을 검색합니다.
GetRemoteAddress

메시지를 보낸 원격 주소를 나타내는 제네릭 주소 개체를 검색합니다.
GetRemoteTransportAddress IP(원격 전송) 주소의 문자열 표현을 검색합니다.
GetRetransmitParams WSD에서 메시지 전송을 반복하는 빈도를 결정하는 데 사용하는 값을 검색합니다.
GetScopeRule 구현된 scope 일치 규칙을 정의하는 URI를 반환하기 위해 호출됩니다.
GetScopes WS-Discovery 범위 목록을 검색합니다.
GetSecure 이 주소에 대해 TLS 보안 세션을 사용할 수 있는지 여부에 대한 상태 검색합니다.
GetService 현재 등록되지 않은 서비스 개체를 검색합니다.
GetServiceMetadata IWSDServiceProxy 개체에 대한 메타데이터를 검색합니다.
GetServiceProxyById 서비스 ID별로 일반 IWSDServiceProxy 서비스 프록시를 검색합니다.
GetServiceProxyByType 포트 유형 이름으로 디바이스에서 노출하는 서비스에 대한 일반 IWSDServiceProxy 프록시를 검색합니다.
GetSignature 메시지의 서명을 가져옵니다.
GetSignedInfoHash 메시지 서명의 해시를 가져옵니다.
GetSockaddr 소켓 주소 정보를 가져옵니다.

GetStatusForMultipleOperations 현재 상태 검색합니다.
GetThisDeviceMetadata 이 디바이스에 대한 디바이스별 메타데이터를 검색합니다.
GetThisModelMetadata 디바이스에 대한 모델별 메타데이터를 검색합니다.
GetTransportAddress 주소 개체의 문자열 표현에 대한 포인터를 가져옵니다. (IWSDTransportAddress.GetTransportAddress)
GetTransportAddressEx 주소 개체의 문자열 표현에 대한 포인터를 가져옵니다. (IWSDTransportAddress.GetTransportAddressEx)
GetTTL 이 주소를 사용하여 모든 아웃바운드 패킷에 대한 TTL(Time to Live)을 가져옵니다.
GetTypes WS-Discovery 형식 목록을 검색합니다.
GetValue 지정된 속성 키의 값을 가져옵니다.
GetXAddrs WS-Discovery XAddrs 목록을 검색합니다.
GetXMLContext 이 공급자와 연결된 XML 컨텍스트를 가져옵니다.
GetXMLContext 디바이스와 연결된 XML 컨텍스트를 가져옵니다.
HasCompleted 작업이 완료되었는지 여부를 나타냅니다.

Init IWSDDeviceHost 개체의 instance 초기화합니다.
Init 디바이스 프록시를 초기화하고 필요에 따라 이전에 초기화된 스폰서 디바이스 프록시와 세션을 공유합니다.
초기화 함수 검색 공급자 개체를 초기화합니다.
초기화 함수 instance 나타내는 리소스에 바인딩된 특정 인터페이스를 제공하는 개체를 초기화합니다.
InstancePropertyStoreFlush 공급자가 속성을 유지하는 메커니즘을 제공합니다.
InstancePropertyStoreOpen 공급자의 속성 저장소를 엽니다.
InstancePropertyStoreValidateAccess 공급자가 요청된 액세스를 지원하는지 확인합니다.
InstanceQueryService 함수 instance 대한 공급자별 COM 개체를 만듭니다.
InstanceReleased 지정된 함수 instance 해제하고 이전에 할당된 메모리를 해제합니다.
IsInstanceQuery 쿼리가 단일 함수 instance 또는 여러 함수 인스턴스에 대한 쿼리인지 여부를 결정합니다.
IsMessageSignatureTrusted 메시지 서명을 신뢰할 수 있는지를 지정합니다.
IsMessageSigned 메시지가 서명되었는지를 지정합니다.

IsSubcategoryQuery 쿼리가 특정 하위 범주의 함수 인스턴스에 대한 쿼리인지 여부를 결정합니다.
Item 인덱스별로 지정된 함수 instance 가져옵니다.
Item 인덱스별로 지정된 속성 제약 조건의 이름과 값을 가져옵니다.
Item 인덱스별로 지정된 쿼리 제약 조건의 이름과 값을 가져옵니다.
MatchProbe 프로브 메시지가 지정된 호스트와 일치하는지 여부를 확인하고 일치하는 경우 WS-Discovery ProbeMatches 메시지를 보냅니다.
MatchProbeEx 프로브 메시지가 지정된 호스트와 일치하는지 여부를 확인하고 일치하는 경우 확장 정보가 포함된 WS-Discovery ProbeMatches 메시지를 보냅니다.
MatchResolve Resolve 메시지가 지정된 호스트와 일치하는지 여부를 확인하고 일치하는 경우 WS-Discovery ResolveMatches 메시지를 보냅니다.
MatchResolveEx Resolve 메시지가 지정된 호스트와 일치하는지 여부를 확인하고 일치하는 경우 확장 정보가 포함된 WS-Discovery ResolveMatches 메시지를 보냅니다.
MatchScopes 두 범위를 비교하여 일치하는지 확인하기 위해 호출됩니다.
다음 컬렉션에서 다음 속성 제약 조건의 이름과 값을 가져옵니다.
다음 컬렉션에서 다음 쿼리 제약 조건의 이름과 값을 가져옵니다.

OnError 비동기 쿼리를 처리하는 동안 발생하는 오류를 수신합니다.
Onevent 알림 중에 추가, 제거 또는 업데이트 이벤트를 수신합니다.
OnUpdate 함수 instance 추가, 제거 또는 변경되었음을 나타냅니다.
OpenPropertyStore 함수 instance 대한 속성 저장소를 엽니다.
ProbeHandler 검색 게시자가 프로브를 수신할 때 호출됩니다.
ProcessFault GetFaultInfo에서 검색한 SOAP 오류를 처리합니다.
게시 Hello 메시지를 보내 네트워크 호스트의 존재를 알릴 수 있습니다.
PublishEx 확장된 정보가 포함된 Hello 메시지를 보내 네트워크 호스트의 존재를 알릴 수 있습니다.
PWSD_SOAP_MESSAGE_HANDLER 들어오는 메시지에 대한 SOAP 메시지 처리기를 참조합니다.
쿼리 지정된 제약 조건을 충족하는 함수 인스턴스의 컬렉션을 검색합니다.
읽기 원격 호스트에서 보낸 메시지에서 첨부 파일 데이터를 검색합니다.
RegisterNotificationSink 검색 게시자에 콜백 알림 싱크를 연결합니다.
RegisterPortType

들어오는 메시지에 대한 포트 형식을 등록합니다.
RegisterScopeMatchingRule 사용자 지정 scope 일치 규칙에 대한 지원을 추가합니다.
RegisterService 들어오는 요청에 대한 서비스 개체를 등록하고 디바이스 호스트 메타데이터에 서비스를 추가합니다.
제거 지정된 함수 instance 삭제하고 제거되는 instance 함수에 대한 포인터를 반환합니다.
제거 최근에 출발한 검색 호스트에 대한 정보를 제공합니다(Bye 메시지에서).
RemoveDynamicService AddDynamicService를 사용하여 등록된 서비스 개체의 등록을 취소합니다.
RemoveInstance 범주 및 하위 범주에 따라 지정된 함수 instance 제거합니다.
RemoveInstance 기존 함수 instance 삭제합니다.
RenewMultipleOperations 새 기간을 제출하여 기존 알림 구독의 컬렉션을 갱신합니다.
Reset 현재 인덱스를 컬렉션의 시작 부분까지 다시 설정합니다. (IProviderPropertyConstraintCollection.Reset)
Reset 현재 인덱스를 컬렉션의 시작 부분까지 다시 설정합니다. (IProviderQueryConstraintCollection.Reset)
ResolveHandler 검색 게시자가 Resolve를 수신할 때 호출됩니다.

RetireService RegisterService를 사용하여 등록된 서비스 개체의 등록을 취소하고 디바이스 호스트 메타데이터에서 서비스를 제거합니다.
SearchByAddress 디바이스 주소별로 WS-Discovery 호스트에 대한 검색을 초기화합니다.
SearchById 디바이스 식별자를 통해 WS-Discovery 호스트에 대한 검색을 초기화합니다.
SearchByType 디바이스 유형별로 WS-Discovery 호스트에 대한 검색을 초기화합니다.
SearchComplete 사용자가 시작한 검색이 성공적으로 완료되었으며 검색에 대한 일치 항목이 더 이상 허용되지 않음을 나타내기 위해 호출됩니다.
SearchFailed 사용자가 시작한 검색이 실패했음을 나타내기 위해 호출됩니다.
SendOneWayRequest 단방향 요청 메시지를 보냅니다.
SendResponse 지정된 요청 컨텍스트와 일치하는 응답 메시지를 보냅니다.
SendTwoWayRequest 동기 호출 패턴을 사용하여 양방향 요청 메시지를 보냅니다.
SendTwoWayRequestAsync 비동기 호출 패턴을 사용하여 양방향 요청 메시지를 보냅니다.
직렬화 주소의 구성 요소 부분을 문자열로 어셈블합니다.
SetAddressFamily WSD 디바이스를 검색할 때 검색할 IP 주소 패밀리(IPv4, IPv6 또는 둘 다)를 지정합니다.

SetAddressFamily 호스트가 게시될 IP 주소 패밀리(IPv4, IPv6 또는 둘 다)를 지정합니다.
SetAlias 검색 주소의 별칭을 설정합니다.
SetCallback 비동기 작업이 완료된 경우 호출할 콜백 인터페이스를 지정합니다.
Setcontext 현재 트랜잭션에 대한 프라이빗 전송 컨텍스트를 설정합니다.
SetEventingStatusCallback 이벤트 상태 콜백을 설정하거나 지웁니다.
SetExclusive 소켓이 배타적 모드인지 여부를 제어합니다.
SetID 현재 트랜잭션의 전송 ID를 설정합니다.
SetInboundHttpHeaders 인바운드 SOAP over-HTTP 전송에 사용되는 HTTP 헤더를 설정합니다.
SetLocalAddress 메시지를 보내야 하는 원본 주소를 나타내는 제네릭 주소 개체를 설정합니다.
SetMessageType 이 UDP 주소 구성에 대한 메시지 유형을 설정합니다.
SetMetadata 사용자 정의 서비스 메타데이터를 제외한 디바이스의 메타데이터를 설정합니다.
SetNamespaces 사용자 지정 네임스페이스를 XML 컨텍스트 개체와 연결합니다.
SetOutboundHttpHeaders

아웃바운드 SOAP over-HTTP 전송에 사용되는 HTTP 헤더를 설정합니다.
SetPath 이 주소의 URI 경로를 설정합니다.
SetPort 이 전송 주소의 IP 포트 번호만 설정합니다.
SetRemoteAddress 원격 주소를 나타내는 제네릭 주소 개체를 메시지가 전송되는 위치로 설정합니다.
SetRetransmitParams WSD에서 메시지 전송을 반복하는 빈도를 결정하는 데 사용하는 값을 설정합니다.
SetSecure 이 주소에 대해 TLS 보안 세션을 사용하거나 사용하지 않도록 설정합니다.
SetServiceDiscoverable WS-Discovery를 사용하여 서비스를 보급할지 여부를 제어합니다.
SetSockaddr 소켓 주소 정보를 설정합니다.
SetTransportAddress 전송 주소의 문자열 표현을 설정합니다.
SetTTL 이 주소를 사용하여 모든 아웃바운드 패킷에 대한 TTL(Time to Live)을 설정합니다.
SetTypes 사용자 지정 메시지 형식을 XML 컨텍스트 개체와 연결합니다.
SetValue 지정된 속성 키의 값을 설정합니다.
SetWaitHandle 작업이 완료되면 설정할 대기 핸들을 지정합니다.

SignalEvent 모든 구독된 클라이언트에 이벤트가 발생했음을 알 수 있습니다.
Skip 컬렉션의 다음 항목을 건너뜁니다. (IProviderPropertyConstraintCollection.Skip)
Skip 컬렉션의 다음 항목을 건너뜁니다. (IProviderQueryConstraintCollection.Skip)
시작 디바이스 호스트를 시작하고 WS-Discovery Hello 메시지를 사용하여 디바이스 호스트를 게시합니다.
중지 WS-Discovery Bye 메시지를 보내고 호스트를 중지합니다.
SubscribeToMultipleOperations 알림 또는 요청/응답 이벤트 컬렉션을 구독합니다.
SubscribeToOperation 알림 또는 요청/응답 이벤트를 구독합니다.
SubscriptionEnded 디바이스가 구독을 종료할 때 호출됩니다.
SubscriptionRenewalFailed 지정된 이벤트 작업에 대한 구독을 갱신할 수 없을 때 호출됩니다.
SubscriptionRenewed 지정된 이벤트 작업에 대한 구독이 성공적으로 갱신되었을 때 호출됩니다.
Terminate 호스트를 종료하고 연결된 서비스를 해제합니다.
연결 해제 연결 데이터베이스 항목을 연결되지 않은 것으로 표시합니다.

연결 해제 연결 데이터베이스 항목을 연결되지 않은 것으로 표시하고 적절한 알림을 보냅니다.
취소 Bye 메시지를 보내 네트워크 호스트의 출발을 알릴 수 있습니다.
UnRegisterNotificationSink 검색 게시자에서 콜백 알림 싱크를 분리합니다.
UnRegisterScopeMatchingRule 사용자 지정 scope 일치 규칙에 대한 지원을 제거합니다.
UnsubscribeToMultipleOperations 알림 또는 요청/응답 이벤트에 대한 구독 컬렉션을 취소합니다.
UnsubscribeToOperation 알림 또는 요청/응답 이벤트에 대한 구독을 취소합니다.
쓰기 MIME 컨테이너를 사용하여 원격 호스트에 첨부 파일 데이터를 보냅니다.
WSD_STUB_FUNCTION 들어오는 메시지를 처리하는 데 사용되는 스텁 함수에 대해 설명합니다.
WSDAllocateLinkedMemory 연결된 메모리 블록을 할당합니다.
WSDAttachLinkedMemory 자식 메모리 블록을 부모 메모리 블록에 연결합니다.
WSDCreateDeviceHost 디바이스 호스트를 만들고 IWSDDeviceHost 인터페이스에 대한 포인터를 반환합니다. (WSDCreateDeviceHost)
WSDCreateDeviceHost2 서명된 메시지를 지원할 수 있는 디바이스 호스트를 만들고 IWSDDeviceHost 인터페이스에 대한 포인터를 반환합니다.

WSDCreateDeviceHostAdvanced 디바이스 호스트를 만들고 IWSDDeviceHost 인터페이스에 대한 포인터를 반환합니다. (WSDCreateDeviceHostAdvanced)
WSDCreateDeviceProxy 디바이스 프록시를 만들고 IWSDDeviceProxy 인터페이스에 대한 포인터를 반환합니다. (WSDCreateDeviceProxy)
WSDCreateDeviceProxy2 서명된 메시지를 지원할 수 있는 디바이스 프록시를 만들고 IWSDDeviceProxy 인터페이스에 대한 포인터를 반환합니다.
WSDCreateDeviceProxyAdvanced 디바이스 프록시를 만들고 IWSDDeviceProxy 인터페이스에 대한 포인터를 반환합니다. (WSDCreateDeviceProxyAdvanced)
WSDCreateDiscoveryProvider IWSDiscoveryProvider 개체를 만듭니다.
WSDCreateDiscoveryProvider2 서명된 메시지를 지원하는 IWSDiscoveryProvider 개체를 만듭니다.
WSDCreateDiscoveryPublisher IWSDiscoveryPublisher 개체를 만듭니다.
WSDCreateDiscoveryPublisher2 서명된 메시지를 지원하는 IWSDiscoveryPublisher 개체를 만듭니다.
WSDCreateHttpAddress IWSDHttpAddress 개체를 만듭니다.
WSDCreateHttpMessageParameters IWSDHttpMessageParameters 개체를 만듭니다.
WSDCreateOutboundAttachment IWSDOutboundAttachment 개체를 만듭니다.

WSDCreateUdpAddress IWSDUdpAddress 개체를 만듭니다.
WSDCreateUdpMessageParameters IWSDUdpMessageParameters 인터페이스에 대한 포인터를 검색합니다.
WSDDetachLinkedMemory 부모 메모리 블록에서 자식 메모리 블록을 분리합니다.
WSDFreeLinkedMemory 이전에 WSDAllocateLinkedMemory로 할당된 메모리 블록을 해제합니다.
WSDGenerateFault SOAP 오류를 생성합니다. (WSDGenerateFault)
WSDGenerateFaultEx SOAP 오류를 생성합니다. (WSDGenerateFaultEx)
WSDGetConfigurationOption WSDAPI 구성 옵션을 가져옵니다.
WSDSetConfigurationOption WSDAPI 구성 옵션을 설정합니다.
WSDUriDecode RFC2396의 규칙에 따라 URI를 디코딩합니다.
WSDUriEncode RFC2396의 URI 인코딩 규칙에 따라 URI를 인코딩합니다.
WSDXMLAddChild 자식 요소를 추가합니다.
WSDXMLAddSibling 형제 요소를 추가합니다.
WSDXMLBuildAnyForSingleElement

지정된 이름과 값을 사용하여 XML 요소를 만듭니다.
WSDXMLCleanupElement XML 요소와 연결된 메모리를 해제합니다.
WSDXMLCreateContext 새 IWSDXMLContext 개체를 만듭니다.
WSDXMLGetNameFromBuiltinNamespace 기본 제공 네임스페이스에서 지정된 이름을 가져옵니다.
WSDXMLGetValueFromAny XML 요소의 지정된 자식 요소에서 텍스트 값을 검색합니다.

인터페이스

IFunctionDiscovery 이 인터페이스는 클라이언트 프로그램에서 함수 인스턴스를 검색하고, 범주에 대한 기본 함수 instance 가져와서 함수 검색 기본값을 등록할 수 있는 고급 함수 검색 쿼리 개체를 만드는 데 사용됩니다.
IFunctionDiscoveryNotification 이 인터페이스는 비동기 쿼리를 지원하기 위해 클라이언트 프로그램에 의해 구현되며, 쿼리 매개 변수를 충족하는 함수 instance 추가되거나 제거된 경우 함수 검색에서 클라이언트 프로그램에 알리기 위해 호출됩니다.
IFunctionDiscoveryProvider 검색 공급자가 구현하는 기본 인터페이스입니다. 함수 검색 인프라가 공급자 및 해당 리소스와 통신하는 데 사용하는 기본 인터페이스입니다.
IFunctionDiscoveryProviderFactory 함수 검색 개체를 만드는 팩터리 메서드를 제공합니다.
IFunctionDiscoveryProviderQuery 이 인터페이스는 모든 IFunctionDiscoveryProvider::Query 메서드 호출에 전달되며 쿼리 정의 정보

를 포함합니다. 공급자는 이를 사용하여 수신하는 각 쿼리 요청에 대한 제약 조건을 결정해야 합니다.
IFunctionDiscoveryServiceProvider 이 인터페이스는 개체를 만들고 초기화하여 함수 instance 나타내는 리소스에 지정된 액세스 인터페이스를 제공하기 위해 구현됩니다. 개체를 만든 후 개체를 초기화하기 위해 Initialize 메서드가 호출됩니다.
IFunctionInstance instance 함수는 IFunctionDiscovery 메서드 중 하나를 호출한 결과로 만들어집니다. 클라이언트 프로그램은 이러한 개체를 직접 만들지 않습니다.
IFunctionInstanceCollection 쿼리 또는 가져오기 요청의 결과로 반환된 IFunctionInstance 개체의 그룹을 instance.
IFunctionInstanceCollectionQuery 범주 및 하위 범주를 기반으로 하는 함수 인스턴스 컬렉션에 대한 비동기 쿼리를 구현합니다.
IFunctionInstanceQuery 범주 및 하위 범주를 기반으로 instance 함수에 대한 비동기 쿼리를 구현합니다.
IPNPXAssociation PnP-X 디바이스에 대한 연결 데이터베이스 항목을 관리하는 메서드를 정의합니다.
IPNPXDeviceAssociation PnP-X 디바이스에 대한 연결 데이터베이스 항목을 관리하는 메서드를 정의합니다. 이러한 메서드는 해당 PnP devnode가 변경되면 알림을 보냅니다.
IProviderProperties 필요에 따라 검색 공급자가 자체 속성 저장소를 직접 만들고 관리하기 위해 구현됩니다.
IProviderPropertyConstraintCollection 이 인터페이스는 IFunctionDiscoveryProviderQuery::GetPropertyConstraints를 통해 공급자가 액세스할 수 있습니다.
IProviderPublishing 클라이언트 프로그램이 함수 인스턴스를 추가하고 제거할 수 있도록 검색 공급자에 의해 구현됩니다.

IProviderQueryConstraintCollection 이 인터페이스는 IFunctionDiscoveryProviderQuery::GetQueryConstraints 메서드를 통해 공급자가 액세스할 수 있습니다.
IWSDAddress 전송 주소의 개별 구성 요소에 대한 액세스를 제공합니다.
IWSDAsyncCallback 비동기 작업 완료를 위한 콜백을 처리합니다.
IWSDAsyncResult 비동기 작업을 나타냅니다.
IWSDAttachment 다른 모든 첨부 파일 형식의 기본 인터페이스입니다.
IWSDDeviceHost DPWS 규격 디바이스를 나타냅니다.
IWSDDeviceHostNotify IWSDDeviceHost 개체의 instance 디바이스 관련 알림을 제공합니다.
IWSDDeviceProxy 클라이언트 애플리케이션 및 미들웨어에 대한 DPWS(Web Services) 디바이스의 원격 디바이스 프로필을 나타냅니다.
IWSDEndpointProxy 디바이스 서비스 메시징 프록시를 구현합니다.
IWSDEventingStatus 이벤트 구독에서 상태 변경이 발생할 때 알림을 받도록 이 인터페이스를 구현합니다.
IWSDHttpAddress HTTP 주소의 개별 구성 요소에 대한 액세스를 제공합니다.
IWSDHttpAuthParameters

이 인터페이스를 사용하여 클라이언트 인증 중에 사용되는 액세스 토큰 또는 권한 부여 체계를 검색합니다.
IWSDHttpMessageParameters SOAP over-HTTP를 통해 메시지를 전송할 때 사용되는 HTTP 헤더에 대한 액세스를 제공합니다.
IWSDInboundAttachment 애플리케이션이 들어오는 메시지에서 MIME로 인코딩된 첨부 파일 데이터를 읽을 수 있습니다.
IWSDiscoveredService 이 인터페이스는 원격으로 검색된 호스트를 나타냅니다.
IWSDiscoveryProvider WS-Discovery에서 보급한 네트워크에서 서비스를 검색하는 데 사용됩니다.
IWSDiscoveryProviderNotify IWSDiscoveryProvider에서 콜백 알림을 수신하기 위해 클라이언트 프로그램에 의해 구현됩니다.
IWSDiscoveryPublisher 호스트를 발표하고 호스트에 들어오는 쿼리를 관리하는 메서드를 제공합니다.
IWSDiscoveryPublisherNotify IWSDiscoveryPublisher에서 콜백 알림을 수신하기 위해 클라이언트 프로그램에 의해 구현됩니다.
IWSDMessageParameters 이 인터페이스를 사용하여 프로토콜 스택 위아래로 메시지 관련 정보를 전달합니다.
IWSDMetadataExchange 메타데이터에 액세스하는 다른 개체의 기본 클래스입니다.
IWSDOutboundAttachment 애플리케이션이 MIME 컨테이너를 사용하여 메시지의 첨부 파일 데이터를 보낼 수 있도록 합니다.
IWSDScopeMatchingRule WS-Discovery에 정의된 표준 scope 일치 규칙을 확장하는 데 사용할 수 있는 사용자 지정 scope 일치 규칙을 제공하기 위해 클라이언트 프로그램에서 구현됩니다.
IWSDServiceMessaging

생성된 스텁 코드에서 들어오는 메시지에 오류 또는 응답을 보내는 데 사용됩니다.
IWSDServiceProxy 클라이언트 애플리케이션 및 미들웨어에 대한 원격 WSD 서비스를 나타냅니다.
IWSDServiceProxyEventing 클라이언트 애플리케이션 및 미들웨어에 대한 원격 WSD 서비스를 나타냅니다. 이 인터페이스를 사용하면 여러 비동기 작업을 구현할 수 있습니다.
IWSDSignatureProperty 서명된 메시지의 속성을 제공합니다.
IWSDSSLClientCertificate 클라이언트 SSL 인증서를 검색합니다.
IWSDTransportAddress IP 기반 전송 주소를 나타냅니다.
IWSDUdpAddress UDP 주소의 개별 구성 요소에 대한 액세스를 제공합니다.
IWSDUdpMessageParameters 이 인터페이스를 사용하여 WSD가 메시지 전송을 반복하는 빈도를 지정합니다.
IWSDXMLContext WSDAPI 스택에서 사용되는 네임스페이스 및 형식의 컬렉션입니다.

구조체

REQUESTBODY_GetStatus WS-Eventing GetStatus 요청 메시지를 나타냅니다.
REQUESTBODY_Renew WS-Eventing 갱신 요청 메시지를 나타냅니다.

REQUESTBODY_Subscribe WS-Eventing 구독 요청 메시지를 나타냅니다.
REQUESTBODY_Unsubscribe WS-Eventing 구독 취소 요청 메시지를 나타냅니다.
RESPONSEBODY_GetMetadata WS-MetadataExchange GetMetadata 응답 메시지를 나타냅니다.
RESPONSEBODY_GetStatus WS-Eventing GetStatus 응답 메시지를 나타냅니다.
RESPONSEBODY_Renew WS-Eventing 응답 갱신 메시지를 나타냅니다.
RESPONSEBODY_Subscribe WS-Eventing 구독 응답 메시지를 나타냅니다.
RESPONSEBODY_SubscriptionEnd WS-Eventing SubscriptionEnd 응답 메시지를 나타냅니다.
WSD_APP_SEQUENCE WS-Discovery 메시지와 관련된 애플리케이션 시퀀스 정보를 나타냅니다.
WSD_BYE Bye 메시지를 나타냅니다.
WSD_CONFIG_ADDRESSES 호스트가 수신 대기해야 하는 특정 주소에 대한 정보입니다.
WSD_CONFIG_PARAM WSDAPI 개체를 만들기 위한 구성 매개 변수를 나타냅니다.
WSD_DATETIME 타임스탬프를 나타냅니다.
WSD_DURATION

시간 길이를 나타냅니다.
WSD_ENDPOINT_REFERENCE WS-Addressing 엔드포인트 참조를 나타냅니다.
WSD_ENDPOINT_REFERENCE_LIST WSD_ENDPOINT_REFERENCE 구조체의 단일 연결 목록에 있는 노드를 나타냅니다.
WSD_EVENT SOAP 메시지의 내부 표현을 제공합니다.
WSD_EVENTING_DELIVERY_MODE WS-Eventing 구독 메시지에 사용되는 배달 모드를 나타냅니다.
WSD_EVENTING_DELIVERY_MODE_PUSH WS-Eventing 구독 메시지에서 이벤트를 푸시 배달하는 데 사용되는 엔드포인트 참조를 나타냅니다.
WSD_EVENTING_EXPIRES WS-Eventing 메시지의 만료 시간을 나타냅니다.
WSD_EVENTING_FILTER WS-Eventing 구독 메시지에 사용되는 이벤트 필터를 나타냅니다.
WSD_EVENTING_FILTER_ACTION 이벤트 필터링에 사용되는 부울 식을 나타냅니다.
WSD_HANDLER_CONTEXT 들어오는 메시지를 처리하기 위한 컨텍스트를 지정합니다.
WSD_HEADER_RELATESTO WS-Addressing 사양에 지정된 RelatesTo SOAP 봉투 헤더 블록을 나타냅니다.
WSD_HELLO Hello 메시지를 나타냅니다.
WSD_HOST_METADATA 디바이스에서 호스트하는 모든 서비스에 대한 메타데이터를 제공합니다.

WSD_LOCALIZED_STRING 지역화된 단일 문자열을 나타냅니다.
WSD_LOCALIZED_STRING_LIST 지역화된 문자열의 단일 연결 목록에 있는 노드를 나타냅니다.
WSD_METADATA_SECTION 제네릭 형식의 메타데이터 섹션을 나타냅니다.
WSD_METADATA_SECTION_LIST 메타데이터 섹션의 단일 연결 목록에 있는 노드를 나타냅니다.
WSD_NAME_LIST 단일 연결된 XML 이름 구조 목록의 노드를 나타냅니다.
WSD_OPERATION WSDL에서 정의한 작업을 하나 또는 두 개의 메시지로 설명합니다.
WSD_PORT_TYPE 포트 형식에 대한 데이터를 제공합니다.
WSD_PROBE 프로브 메시지를 나타냅니다.
WSD_PROBE_MATCH ProbeMatch 메시지를 나타냅니다.
WSD_PROBE_MATCH_LIST ProbeMatch 메시지 구조의 단일 연결 목록에 있는 노드를 나타냅니다.
WSD_PROBE_MATCHES ProbeMatches 메시지를 나타냅니다.
WSD_REFERENCE_PARAMETERS 엔드포인트에서 사용되는 불투명 데이터를 지정합니다.
WSD_REFERENCE_PROPERTIES

엔드포인트를 고유하게 식별하는 데 사용되는 추가 데이터를 지정합니다.
WSD_RELATIONSHIP_METADATA 두 개 이상의 서비스 간의 관계에 대한 메타데이터를 제공합니다.
WSD_RESOLVE 확인 메시지를 나타냅니다.
WSD_RESOLVE_MATCH ResolveMatch 메시지를 나타냅니다.
WSD_RESOLVE_MATCHES ResolveMatches 메시지를 나타냅니다.
WSD_SCOPES WS-Discovery 메시징에 사용되는 범위의 컬렉션입니다.
WSD_SECURITY_CERT_VALIDATION HTTPS 서버와 클라이언트 인증서를 일치시키는 조건을 나타냅니다. (WSD_SECURITY_CERT_VALIDATION)
WSD_SECURITY_CERT_VALIDATION_V1 HTTPS 서버와 클라이언트 인증서를 일치시키는 조건을 나타냅니다. W
WSD_SECURITY_SIGNATURE_VALIDATION 메시지와 클라이언트 압축 서명을 일치시키는 조건을 나타냅니다.
WSD_SERVICE_METADATA 디바이스에서 호스트하는 서비스에 대한 메타데이터를 제공합니다.
WSD_SERVICE_METADATA_LIST 서비스 메타데이터 구조의 단일 연결 목록에 있는 노드를 나타냅니다.
WSD_SOAP_FAULT 생성된 SOAP 오류를 나타냅니다.
WSD_SOAP_FAULT_CODE 생성된 SOAP 오류 코드를 나타냅니다.

WSD_SOAP_FAULT_REASON WSD_SOAP_FAULT 연결된 이유 코드의 컬렉션입니다.
WSD_SOAP_FAULT_SUBCODE 생성된 SOAP 오류 하위 코드를 나타냅니다.
WSD_SOAP_HEADER WSD_SOAP_MESSAGE 구조체에 대한 SOAP 헤더 데이터를 제공합니다.
WSD_SOAP_MESSAGE WSD SOAP 메시지의 내용입니다.
WSD_SYNCHRONOUS_RESPONSE_CONTEXT 양방향 요청에 대한 응답을 처리하기 위한 컨텍스트를 제공합니다.
WSD_THIS_DEVICE_METADATA 특정 디바이스에 고유한 메타데이터를 지정합니다.
WSD_THIS_MODEL_METADATA 디바이스와 관련된 모델별 정보를 제공합니다.
WSD_UNKNOWN_LOOKUP 구문 분석할 수 없는 XML 요소를 나타냅니다.
WSD_URI_LIST 연결된 URI 목록의 노드를 나타냅니다.
WSDUdpRetransmitParams 메시지 전송을 반복하기 위한 매개 변수를 정의합니다.
WSDXML_ATTRIBUTE XML 특성에 대해 설명합니다.
WSDXML_ELEMENT XML 요소를 설명합니다.
WSDXML_ELEMENT_LIST

XML 요소의 연결된 목록에 있는 노드를 나타냅니다.
WSDXML_NAME XML 정규화된 이름을 지정합니다.
WSDXML_NAMESPACE XML 네임스페이스를 지정합니다.
WSDXML_NODE XML 노드에 대해 설명합니다.
WSDXML_PREFIX_MAPPING XML 네임스페이스 접두사를 설명합니다.
WSDXML_TEXT XML 노드의 텍스트를 설명합니다.
WSDXML_TYPE XSD 형식에 대해 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Windows 스토어 앱의 패키징, 배포 및 쿼리

아티클2025. 01. 28.

Windows 스토어 앱 기술의 패키징, 배포 및 쿼리 개요입니다.

Windows 스토어 앱의 패키징, 배포 및 쿼리를 개발하려면 다음 헤더가 필요합니다.

- appmodel.h
- appxpackaging.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 스토어 앱 패키징, 배포 및 쿼리

열거형

테이블 확장

AddPackageDependencyOptions 패키지 종속성을 추가할 때 적용할 수 있는 옵션을 정의합니다.
AppPolicyClrCompat AppPolicyClrCompat 열거형은 프라이빗 리플렉션을 사용하도록 설정할지 또는 관리되는 개체를 민첩하게 설정할지 결정할 수 있도록 프로세스의 애플리케이션 형식을 나타냅니다.
AppPolicyCreateFileAccess AppPolicyCreateFileAccess 열거형은 프로세스에 IO 디바이스(파일, 파일 스트림, 디렉터리, 실제 디스크, 볼륨, 콘솔 버퍼, 테이프 드라이브, 통신 리소스, mailslot 및 파이프)에 대한 전체 또는 제한된 액세스 권한이 있는지 여부를 나타냅니다.
AppPolicyLifecycleManagement AppPolicyLifecycleManagement 열거형은 프로세스가 수명 주기 관리인지 여부를 나타냅니다.
appPolicyMediaFoundationCodecLoading AppPolicyMediaFoundationCodecLoading 열거형은 프로세스의 정책에서 비 Windows(타사) 플러그 인을 로드할 수 있는지 여부를 나타냅니다.

AppPolicyProcessTerminationMethod AppPolicyProcessTerminationMethod 열거형은 프로세스를 종료하는 데 사용되는 메서드를 나타냅니다.
AppPolicyShowDeveloperDiagnostic AppPolicyShowDeveloperDiagnostic 열거형은 어설션과 같은 개발자 정보를 사용자에게 표시하는 프로세스에 사용되는 메서드를 나타냅니다.
AppPolicyThreadInitializationType AppPolicyThreadInitializationType 열거형은 beginthread[ex]가 스레드를 만들 때 프로세스에 대해 자동으로 수행되어야 하는 초기화의 종류를 나타냅니다.
AppPolicyWindowingModel AppPolicyWindowingModel 열거형은 프로세스에서 CoreWindow 기반 또는 HWND 기반 창 모델을 사용하는지 여부를 나타냅니다.
APPX_BUNDLE_FOOTPRINT_FILE_TYPE 번들에 있는 공간 파일의 형식을 지정합니다.
APPX_BUNDLE_PAYLOAD_PACKAGE_TYPE IAppxBundleManifestPackageInfo 개체에 대한 패키지 유형을 지정합니다.
APPX_CAPABILITIES 패키지에서 요청한 기능 또는 권한을 지정합니다.
APPX_COMPRESSION_OPTION 패키지에 파일을 저장하는 데 사용되는 압축 수준을 지정합니다.
APPX_ENCRYPTED_PACKAGE_OPTIONS 암호화된 앱 패키지 옵션입니다.
APPX_FOOTPRINT_FILE_TYPE 패키지의 공간 파일 형식을 지정합니다.
APPX_PACKAGE_ARCHITECTURE 패키지에서 지원하는 프로세서 아키텍처를 지정합니다. (APPX_PACKAGE_ARCHITECTURE)

APPX_PACKAGE_ARCHITECTURE2 패키지에서 지원하는 프로세서 아키텍처를 지정합니다. (APPX_PACKAGE_ARCHITECTURE2)
APPX_PACKAGE_EDITOR_UPDATE_PACKAGE_MANIFEST_OPTIONS 매니페스트를 업데이트할 때 앱 매니페스트 유효성 검사 옵션입니다.
APPX_PACKAGE_EDITOR_UPDATE_PACKAGE_OPTION 앱 패키지를 업데이트할 때 사용할 옵션입니다.
CreatePackageDependencyOptions TryCreatePackageDependency 함수를 사용하여 패키지 종속성을 만들 때 적용할 수 있는 옵션을 정의합니다.
packageDependencyLifetimeKind 패키지 종속성의 수명을 정의하는 데 사용되는 아티팩트 형식을 나타내는 값을 지정합니다.
packageDependencyProcessorArchitectures TryCreatePackageDependency 함수를 사용하여 만드는 프레임워크 패키지 종속성에 대한 프로세서 아키텍처를 정의합니다.
PackageOrigin 패키지의 원본을 지정합니다.
PackagePathType 검색할 패키지 폴더의 유형을 나타냅니다.

함수

테이블 확장

AddAutomaticFile 콘텐츠 그룹 맵의 자동 콘텐츠 그룹에 파일을 추가합니다.
AddAutomaticGroup

콘텐츠 그룹 맵에 자동 콘텐츠 그룹을 추가합니다.
AddExternalPackageReference 패키지 번들에 외부 패키지에 대한 참조를 추가합니다.
AddExternalPackageReference 패키지 번들 내의 참조를 외부 앱 패키지에 추가합니다.
AddExternalPackageReference 암호화된 패키지 번들 내의 참조를 외부 앱 패키지에 추가합니다. (IAppxEncryptedBundleWriter2.AddExternalPackageReference)
AddExternalPackageReference 암호화된 패키지 번들 내의 참조를 외부 앱 패키지에 추가합니다. (IAppxEncryptedBundleWriter3.AddExternalPackageReference)
AddPackageDependency 이전에 정의된 PackageDependency를 특정 패키지에 확인하고 호출 프로세스의 패키지 그래프에 추가합니다. 종속성이 추가된 후 다른 코드 로드 메서드(예: LoadLibrary 및 CoCreateInstance)는 확인된 패키지에서 이진 파일을 찾을 수 있습니다.
addPackageReference 앱 번들 내의 선택적 앱 패키지 또는 페이로드 파일에 대한 참조를 추가합니다. (IAppxBundleWriter3.AddPackageReference)
addPackageReference 앱 번들 내의 선택적 앱 패키지 또는 페이로드 파일에 대한 참조를 추가합니다. (IAppxBundleWriter4.AddPackageReference)
AddPayloadFile 앱 패키지에 새 페이로드 파일을 추가합니다.
AddPayloadFileEncrypted appx 패키지에 암호화된 새 페이로드 파일을 추가합니다.
AddPayloadFiles 앱 패키지에 하나 이상의 페이로드 파일을 추가합니다.
AddPayloadFilesEncrypted

암호화된 앱 패키지에 하나 이상의 페이로드 파일을 추가합니다.
AddPayloadPackage 번들에 새 앱 패키지를 추가합니다. (IAppxBundleWriter.AddPayloadPackage)
AddPayloadPackage 번들에 새 앱 패키지를 추가합니다. (IAppxBundleWriter4.AddPayloadPackage)
addPayloadPackageEncrypted 번들에 대한 새 페이로드 패키지를 암호화합니다. (IAppxEncryptedBundleWriter.AddPayloadPackageEncrypted)
addPayloadPackageEncrypted 번들에 대한 새 페이로드 패키지를 암호화합니다. (IAppxEncryptedBundleWriter3.AddPayloadPackageEncrypted)
AppPolicyGetClrCompat 프라이빗 리플렉션을 사용하도록 설정하고 관리되는 개체를 민첩하게 만들지 여부를 결정할 수 있도록 프로세스의 애플리케이션 유형을 나타내는 값을 검색합니다.
appPolicyGetCreateFileAccess 프로세스에 IO 디바이스(파일, 파일 스트림, 디렉터리, 실제 디스크, 볼륨, 콘솔 버퍼, 테이프 드라이브, 통신 리소스, mailslot 및 파이프)에 대한 전체 또는 제한된 액세스 권한이 있는지 여부를 나타내는 값을 검색합니다.
AppPolicyGetLifecycleManagement PLM(프로세스 수명 주기 관리자)에서 프로세스를 일시 중단/재개할 수 있는지 여부를 나타내는 값을 검색합니다.
AppPolicyGetMediaFoundationCodecLoading 프로세스의 정책에서 비 Windows(타사) 플러그 인을 로드할 수 있는지 여부를 나타내는 값을 검색합니다. 이 값을 사용하여 비 Windows(타사) 플러그 인을 허용할지 여부를 결정할 수 있습니다.
AppPolicyGetProcessTerminationMethod 프로세스를 종료하는 데 사용되는 메서드를 검색합니다.
AppPolicyGetShowDeveloperDiagnostic 어설션과 같은 개발자 정보를 사용자에게 노출하는 프로세스에 사용되는 메서드를 검색합니다.

appPolicyGetThreadInitializationType beginthread[ex]가 스레드를 만들 때 프로세스에 대해 자동으로 수행되어야 하는 초기화 종류를 검색합니다.
appPolicyGetWindowingModel 프로세스에서 CoreWindow 기반 또는 HWND 기반 창 모델을 사용하는지 여부를 나타내는 값을 검색합니다. 값을 사용하여 창 상태 변경 알림(크기 변경, 표시 유형 변경 등)에 등록하는 방법을 결정할 수 있습니다.
닫기 패키지의 끝에 발자국 파일을 작성하여 번들 패키지를 마무리하고 기록기의 출력 스트림을 닫습니다. (IAppxBundleWriter.Close)
닫기 패키지의 끝에 발자국 파일을 작성하여 번들 패키지를 마무리하고 기록기의 출력 스트림을 닫습니다. (IAppxBundleWriter3.Close)
닫기 번들 매니페스트 및 블록맵 공간 파일을 번들에 씁니다.
닫기 작성된 패키지 스트림을 닫고 종료합니다.
닫기 앱 패키지의 끝에 공간 파일을 쓰고 패키지 작성기 개체의 출력 스트림을 닫습니다.
닫기 패키지 작성기 개체의 출력 스트림을 닫습니다.
ClosePackageInfo 지정된 패키지 정보에 대한 참조를 닫습니다.
ComparePublisher 매니페스트에 정의된 게시자를 사용하여 지정된 게시자를 비교합니다.
CreateAppInstallerReader 앱 설치 관리자 파일에 대한 예상 다이제스트를 지정하기 위한 선택적 매개 변수를 사용하여 IAppInstallerReader의 인스턴스를 만듭니다.

CreateBlockMapReader IStream에서 제공하는 콘텐츠에서 읽기 전용 블록 맵 개체 모델을 만듭니다.
CreateBundleManifestReader 독립 실행형 스트림에서 AppxBundleManifest.xml읽기 전용 번들 매니페스트 개체를 만듭니다.
CreateBundleReader IStream 개체에서 콘텐츠를 읽는 읽기 전용 번들 개체를 만듭니다.
CreateBundleReader2 IStream 개체에서 콘텐츠를 읽는 읽기 전용 번들 개체를 만들고, 번들에 대해 예상되는 다이제스트를 지정하기 위한 선택적 매개 변수를 사용합니다.
CreateBundleWriter 앱 패키지를 추가할 수 있는 쓰기 전용 번들 개체를 만듭니다.
CreateContentGroupMapReader IAppxContentGroupMapReader를 만듭니다.
CreateContentGroupMapWriter IAppxContentGroupMapWriter를 만듭니다.
CreateDeltaPackage 업데이트된 패키지와 기준 패키지의 차이점에서 델타 패키지를 만듭니다.
CreateDeltaPackageUsingBaselineBlockMap 업데이트된 패키지와 기준 블록 맵의 차이점에서 델타 패키지를 만듭니다.
createEncryptedBundleReader 암호화된 Windows 앱 패키지를 추가할 수 있는 읽기 전용 번들 개체를 만듭니다.
CreateEncryptedBundleReader2 암호화된 번들에 필요한 다이제스트를 지정하기 위한 선택적 매개 변수를 사용하여 암호화된 Windows 앱 패키지를 추가할 수 있는 읽기 전용 번들 개체를 만듭니다.
CreateEncryptedBundleWriter

암호화된 Windows 앱 패키지를 추가할 수 있는 쓰기 전용 번들 개체를 만듭니다. (IAppxEncryptionFactory.CreateEncryptedBundleWriter)
CreateEncryptedBundleWriter 암호화된 Windows 앱 패키지를 추가할 수 있는 쓰기 전용 번들 개체를 만듭니다. (IAppxEncryptionFactory3.CreateEncryptedBundleWriter)
CreateEncryptedPackageReader 암호화된 패키지를 읽기 위한 IAppxPackageReader의 새 인스턴스를 만듭니다.
createEncryptedPackageReader2 패키지에 대한 예상 다이제스트를 지정하기 위한 선택적 매개 변수를 사용하여 암호화된 패키지를 읽기 위한 IAppxPackageReader의 새 인스턴스를 만듭니다.
CreateEncryptedPackageWriter IAppxEncryptedPackageWriter의 새 인스턴스를 만듭니다. (IAppxEncryptionFactory.CreateEncryptedPackageWriter)
CreateEncryptedPackageWriter IAppxEncryptedPackageWriter의 새 인스턴스를 만듭니다. (IAppxEncryptionFactory2.CreateEncryptedPackageWriter)
CreateEncryptedPackageWriter IAppxEncryptedPackageWriter의 새 인스턴스를 만듭니다. (IAppxEncryptionFactory3.CreateEncryptedPackageWriter)
CreateManifestReader IStream에서 제공하는 콘텐츠에서 읽기 전용 매니페스트 개체 모델을 만듭니다.
createManifestReader2 IStream에서 제공하는 콘텐츠에서 읽기 전용 매니페스트 개체 모델을 만들고 매니페스트에 필요한 다이제스트를 지정하기 위한 선택적 매개 변수를 사용합니다.
CreatePackageReader IStream에서 제공하는 콘텐츠에서 읽기 전용 패키지 판독기를 만듭니다. 이 메서드는 디지털 서명의 유효성을 검사하지 않습니다.
CreatePackageReader2

패키지에 필요한 다이제스트를 지정하기 위한 선택적 매개 변수를 사용하여 IStream에서 제공하는 콘텐츠에서 읽기 전용 패키지 판독기를 만듭니다.
CreatePackageWriter 파일을 추가할 수 있는 쓰기 전용 패키지 개체를 만듭니다.
CreateSourceContentGroupMapReader IAppxSourceContentGroupMapReader를 만듭니다.
CreateValidatedBlockMapReader IStream 및 디지털 서명에서 제공하는 콘텐츠에서 읽기 전용 블록 맵 개체 모델을 만듭니다.
DecryptBundle 암호화된 Windows 앱 번들에서 암호화되지 않은 Windows 앱 번들을 만듭니다.
DecryptPackage 암호화된 Windows 앱 패키지에서 암호화되지 않은 Windows 앱 패키지를 만듭니다.
DeletePackageDependency TryCreatePackageDependency 메서드를 사용하여 이전에 만든 프레임워크 패키지 종속성에 대한 설치 시간 참조를 삭제합니다. 이 메서드는 다른 앱에 종속성이 없는 경우 프레임워크 패키지를 제거하는 것이 안전하다는 것을 OS에 알릴 수 있습니다.
EncryptBundle 암호화되지 않은 Windows 앱 번들에서 암호화된 Windows 앱 번들을 만듭니다. (IAppxEncryptionFactory.EncryptBundle)
EncryptBundle 암호화되지 않은 Windows 앱 번들에서 암호화된 Windows 앱 번들을 만듭니다. (IAppxEncryptionFactory3.EncryptBundle)
EncryptPackage 암호화되지 않은 Windows 앱 패키지에서 암호화된 Windows 앱 패키지를 만듭니다. (IAppxEncryptionFactory.EncryptPackage)
EncryptPackage 암호화되지 않은 Windows 앱 패키지에서 암호화된 Windows 앱 패키지를 만듭니다. (IAppxEncryptionFactory3.EncryptPackage)

EncryptPackage 암호화되지 않은 Windows 앱 패키지에서 암호화된 Windows 앱 패키지를 만듭니다. (IAppxEncryptionFactory4.EncryptPackage)
FindPackagesByPackageFamily 현재 사용자의 지정된 패밀리 이름을 가진 패키지를 찾습니다.
FormatApplicationUserModelId 패키지 패밀리 이름 및 PRAID(패키지 상대 애플리케이션 ID)에서 애플리케이션 사용자 모델 ID를 생성합니다.
GetApplications 매니페스트에 정의된 애플리케이션을 반복하는 열거자를 가져옵니다.
GetApplicationUserModelId 지정된 프로세스에 대한 애플리케이션 사용자 모델 ID를 가져옵니다.
GetApplicationUserModelIdFromToken 지정된 토큰의 애플리케이션 사용자 모델 ID를 가져옵니다.
GetAppUserModelId 애플리케이션 사용자 모델 식별자를 가져옵니다.
GetArchitecture 매니페스트에 정의된 대로 프로세서 아키텍처를 가져옵니다. (IAppxManifestPackageId.GetArchitecture)
getArchitecture2 매니페스트에 정의된 대로 프로세서 아키텍처를 가져옵니다. (IAppxManifestPackageId2.GetArchitecture2)
GetAutomaticGroups 콘텐츠 그룹 맵에서 자동 콘텐츠 그룹을 가져옵니다.
GetAutomaticGroups 원본 콘텐츠 그룹 맵에서 자동 콘텐츠 그룹을 가져옵니다.

GetBlockMap 번들에서 읽기 전용 블록 맵 개체를 검색합니다.
GetBlockMap 패키지의 블록 맵 개체 모델을 검색합니다.
GetBlocks 블록 맵에 나열된 파일의 블록을 트래버스하기 위한 열거자를 검색합니다.
GetBoolValue 속성 섹션에서 지정된 부울 요소의 값을 가져옵니다.
GetCapabilities 패키지에서 요청한 기능 목록을 가져옵니다.
GetCompressedSize 블록의 압축된 크기를 검색합니다.
GetCompressionOption 패키지에 파일을 저장하는 데 사용되는 압축 옵션을 검색합니다.
GetContentType 파일의 콘텐츠 형식을 검색합니다.
GetCurrent 열거자의 현재 위치에 있는 블록을 가져옵니다.
GetCurrent 열거자의 현재 위치에 있는 파일을 가져옵니다.
GetCurrent 열거자의 현재 위치에 있는 선택적 번들 정보를 가져옵니다.
GetCurrent 열거자의 현재 위치에 있는 <Package> 요소를 가져옵니다.
GetCurrent

열거자의 현재 위치에 있는 콘텐츠 그룹에서 파일을 가져옵니다.
GetCurrent 열거자의 현재 위치에 있는 콘텐츠 그룹을 가져옵니다.
GetCurrent 열거자의 현재 위치에 있는 페이로드 파일을 가져옵니다.
GetCurrent 열거자의 현재 위치에 있는 애플리케이션을 가져옵니다.
GetCurrent 열거자의 현재 위치에 있는 디바이스 기능을 가져옵니다.
GetCurrent 열거자의 현재 위치에 있는 <MainPackageDependency> 요소를 가져옵니다.
GetCurrent 열거자의 현재 위치에 있는 종속성 패키지를 가져옵니다.
GetCurrent 열거자의 현재 위치에 있는 리소스를 가져옵니다.
GetCurrentApplicationUserModelId 현재 프로세스에 대한 애플리케이션 사용자 모델 ID를 가져옵니다.
GetCurrentPackageFamilyName 호출 프로세스의 패키지 패밀리 이름을 가져옵니다.
GetCurrentPackageFullName 호출 프로세스의 패키지 전체 이름을 가져옵니다.
GetCurrentPackageId 호출 프로세스의 패키지 식별자(ID)를 가져옵니다.
GetCurrentPackageInfo 호출 프로세스에 대한 패키지 정보를 가져옵니다. (GetCurrentPackageInfo)

GetCurrentPackageInfo2 호출 프로세스에 대한 패키지 정보를 가져옵니다. (GetCurrentPackageInfo2)
GetCurrentPackagePath 호출 프로세스의 패키지 경로를 가져옵니다. (GetCurrentPackagePath)
GetCurrentPackagePath2 호출 프로세스의 패키지 경로를 가져옵니다. (GetCurrentPackagePath2)
GetDeviceCapabilities 매니페스트에 정의된 디바이스 기능을 반복하는 열거자를 가져옵니다.
GetDigest 연결된 인터페이스에서 관리하는 앱 패키징 개체 개체의 다이제스트 표현이 포함된 LPWSTR에 대한 포인터를 받습니다.
GetFile 지정된 파일 이름을 사용하여 블록 맵의 파일에 해당하는 데이터를 검색합니다.
GetFileName <OptionalBundle>파일 이름 특성을 검색합니다.
GetFileName 패키지의 파일 이름 특성을 검색합니다.
GetFiles 블록 맵에 나열된 파일을 트래버스하기 위한 열거자를 검색합니다.
GetFiles 콘텐츠 그룹에서 파일을 가져옵니다.
GetFootprintFile 번들에서 지정된 유형의 발자국 파일을 검색합니다.
GetFootprintFile 패키지에서 공간 파일을 검색합니다.

GetHasCurrent 열거자의 현재 위치에 블록이 있는지 여부를 확인합니다.
GetHasCurrent 열거자의 현재 위치에 파일이 있는지 여부를 확인합니다. (IAppxBlockMapFilesEnumerator.GetHasCurrent)
GetHasCurrent 열거자의 현재 위치에 선택적 번들 정보가 있는지 여부를 확인합니다.
GetHasCurrent 열거자에 요소가 더 있는지 여부를 결정합니다.
GetHasCurrent 열거자의 현재 위치에 파일이 있는지 여부를 확인합니다. (IAppxContentGroupFilesEnumerator.GetHasCurrent)
GetHasCurrent 열거자의 현재 위치에 콘텐츠 그룹이 있는지 여부를 확인합니다.
GetHasCurrent 열거자의 현재 위치에 페이로드 파일이 있는지 여부를 확인합니다.
GetHasCurrent 열거자의 현재 위치에 애플리케이션이 있는지 여부를 확인합니다.
GetHasCurrent 열거자의 현재 위치에 디바이스 기능이 있는지 여부를 확인합니다.
GetHasCurrent 열거자의 현재 위치에 <MainPackageDependency> 요소가 있는지 여부를 확인합니다.
GetHasCurrent 열거자의 현재 위치에 패키지 종속성이 있는지 여부를 결정합니다.
GetHasCurrent 열거자의 현재 위치에 리소스가 있는지 여부를 확인합니다.

GetHash 블록의 해시 값을 검색합니다.
GetHashMethod 블록 맵에서 블록 해시를 만드는 데 사용되는 해시 알고리즘의 URI를 검색합니다.
GetIdForPackageDependencyContext 지정된 컨텍스트 핸들에 대한 패키지 종속성을 반환합니다.
GetIsDefaultApplicablePackage 앱 패키지가 기본 적용 가능한 패키지인지 여부를 확인합니다.
GetIsNonQualifiedResourcePackage 앱 패키지가 정규화되지 않은 리소스 패키지인지 여부를 확인합니다.
GetIsNonQualifiedResourcePackage 앱 패키지가 정규화되지 않은 리소스 패키지인지 여부를 쿼리합니다.
GetIsOptionalPackage 패키지가 선택 사항인지 여부를 확인합니다.
GetIsPackageReference 패키지가 앱 번들 내에 저장되는지 또는 패키지에 대한 참조인지 여부를 결정합니다.
GetLocalFileHeaderSize 연결된 zip 파일 항목의 zip 로컬 파일 헤더 크기를 검색합니다.
GetMainPackageDependencies 기본 패키지 종속성 열거자를 가져옵니다.
GetMainPackageName 선택적 패키지에서 기본 패키지 이름을 가져옵니다.
GetManifest 번들에서 읽기 전용 매니페스트 개체를 검색합니다.
GetManifest

패키지 앱 매니페스트의 개체 모델을 검색합니다.
getMaxMajorVersionTested 현재 패키지와 호환되도록 테스트된 패키지의 최대 주 버전 번호를 반환합니다.
getMaxVersionTested AppxManifest.xml테스트된 최대 버전을 가져옵니다.
GetMinVersion 현재 패키지에 종속성이 있는 패키지의 최소 버전을 가져옵니다.
GetMinVersion AppxManifest.xml대상 디바이스 패밀리의 최소 버전을 가져옵니다.
GetName 연결된 zip 파일 항목의 이름을 검색합니다.
GetName 콘텐츠 그룹의 이름을 가져옵니다.
GetName 패키지 루트 디렉터리를 기준으로 하는 경로를 포함하여 파일의 이름을 검색합니다.
GetName AppxManifest.xml기본 패키지 종속성의 이름을 가져옵니다.
GetName 현재 패키지에 종속성이 있는 패키지의 이름을 가져옵니다.
GetName 매니페스트에 정의된 패키지의 이름을 가져옵니다.
GetName AppxManifest.xml대상 디바이스 패밀리의 이름을 가져옵니다.
GetOffset 번들의 시작 부분을 기준으로 패키지의 오프셋을 검색합니다.

GetOptionalBundles 루트 <Bundle> 요소 아래의 <OptionalBundles> 요소를 나타내는 개체를 검색합니다.
GetPackageApplicationIds 지정된 패키지에 있는 앱의 ID를 가져옵니다.
GetPackageDependencies 매니페스트에 정의된 종속성을 반복하는 열거자를 가져옵니다.
GetPackageFamilyName AppxManifest.xml기본 패키지 종속성의 패키지 패밀리 이름을 가져옵니다.
GetPackageFamilyName 패키지 패밀리 이름을 가져옵니다.
GetPackageFamilyName 지정된 프로세스의 패키지 패밀리 이름을 가져옵니다.
GetPackageFamilyNameFromToken 지정된 토큰의 패키지 패밀리 이름을 가져옵니다.
GetPackageFullName 패키지 전체 이름을 가져옵니다.
GetPackageFullName 지정된 프로세스의 패키지 전체 이름을 가져옵니다.
GetPackageFullNameFromToken 지정된 토큰의 패키지 전체 이름을 가져옵니다.
GetPackageId <OptionalBundle>ID를 나타내는 개체를 검색합니다.
GetPackageId 앱 패키지의 ID를 나타내는 개체를 검색합니다.
GetPackageId

루트 <Bundle> 요소 아래의 <Identity> 요소를 나타내는 개체를 검색합니다.
GetPackageId 매니페스트에 정의된 패키지 식별자를 가져옵니다.
GetPackageId 지정된 프로세스의 패키지 식별자(ID)를 가져옵니다.
GetPackageInfo 지정된 패키지에 대한 패키지 정보를 가져옵니다. (GetPackageInfo)
GetPackageInfo2 지정된 패키지에 대한 패키지 정보를 가져옵니다. (GetPackageInfo2)
GetPackageInfoItems 번들에서 선택적 패키지를 검색합니다.
GetPackageInfoItems <Packages> 요소 아래의 모든 <Package> 요소에 대한 열거자를 검색합니다.
GetPackagePath 지정된 패키지의 경로를 가져옵니다.
GetPackagePathByFullName 지정된 패키지의 경로를 가져옵니다. (GetPackagePathByFullName)
GetPackagePathByFullName2 지정된 패키지의 경로를 가져옵니다. (GetPackagePathByFullName2)
GetPackagesByPackageFamily 현재 사용자의 지정된 패밀리 이름을 가진 패키지를 가져옵니다.
GetPackageType 패키지 정보로 표시되는 패키지 유형을 검색합니다.
GetPayloadFile 패키지에서 페이로드 파일을 검색합니다.

GetPayloadFiles 패키지의 페이로드 파일을 반복하는 열거자를 검색합니다.
GetPayloadPackage 지정된 파일 이름을 사용하여 페이로드 패키지에 대한 appx 파일 개체를 검색합니다.
GetPayloadPackages 번들의 모든 페이로드 패키지 목록을 반복하는 열거자를 검색합니다.
GetPrerequisite 패키지 매니페스트에 정의된 대로 지정된 필수 구성 요소를 가져옵니다.
GetProperties 매니페스트에 정의된 대로 패키지의 속성을 가져옵니다.
GetPublisher AppxManifest.xml기본 패키지 종속성의 게시자를 가져옵니다.
GetPublisher 현재 패키지가 의존하는 패키지를 생성한 게시자의 이름을 가져옵니다.
GetPublisher 매니페스트에 정의된 패키지 게시자의 이름을 가져옵니다.
GetQualifiedResources 매니페스트에 정의된 정규화된 리소스를 반복하는 열거자를 가져옵니다.
getRequiredGroup 콘텐츠 그룹 맵에서 필요한 콘텐츠 그룹을 가져옵니다.
getRequiredGroup 원본 콘텐츠 그룹 맵에서 필요한 콘텐츠 그룹을 가져옵니다.
getResolvedPackageFullNameForPackageDependency 패키지 종속성을 확인할 경우 사용할 패키지 전체 이름을 반환합니다. 이 함수는 프로세스 그래프에 패키지를 추가하지 않습니다.

GetResourceId 매니페스트에 정의된 패키지 리소스 식별자를 가져옵니다.
GetResources 앱 패키지의 매니페스트에 정의된 모든 <Resource> 요소를 반복하는 열거자를 검색합니다.
GetResources 매니페스트에 정의된 리소스를 반복하는 열거자를 가져옵니다.
GetSize 패키지의 크기를 바이트 단위로 검색합니다.
GetSize 압축되지 않은 파일 크기를 검색합니다.
GetStagedPackageOrigin 지정된 패키지의 원본을 가져옵니다.
GetStagedPackagePathByFullName 지정된 스테이징된 패키지의 경로를 가져옵니다. (GetStagedPackagePathByFullName)
GetStagedPackagePathByFullName2 지정된 스테이징된 패키지의 경로를 가져옵니다. (GetStagedPackagePathByFullName2)
GetStream 블록 맵의 XML 콘텐츠를 나타내는 읽기 전용 스트림을 검색합니다.
GetStream 전처리 없이 원시 XML 문서를 가져옵니다.
GetStream 파일의 압축되지 않은 콘텐츠가 포함된 읽기 전용 스트림을 가져옵니다.
GetStream 매니페스트 판독기에서 구문 분석하고 읽은 원시 XML을 가져옵니다.
GetStringValue

매니페스트의 애플리케이션 메타데이터 섹션에 있는 문자열 요소의 값을 가져옵니다.
GetStringValue 속성 섹션에서 지정된 문자열 요소의 값을 가져옵니다.
getUncompressedSize 연결된 zip 파일 항목의 압축되지 않은 크기를 검색합니다.
GetVersion 매니페스트에 정의된 패키지의 버전을 가져옵니다.
GetXmlDom 앱 설치 관리자 파일의 XML DOM을 가져옵니다.
moveNext 열거자의 위치를 다음 블록으로 진행합니다.
moveNext 열거자의 위치를 다음 파일로 진행합니다. (IAppxBlockMapFilesEnumerator.MoveNext)
moveNext 열거자의 위치를 다음 선택적 번들 정보 집합으로 승급합니다.
moveNext 열거자의 위치를 다음 <Package> 요소로 승급합니다.
moveNext 열거자의 위치를 다음 파일로 진행합니다. (IAppxContentGroupFilesEnumerator.MoveNext)
moveNext 열거자의 위치를 다음 콘텐츠 그룹으로 진행합니다.
moveNext 열거자의 위치를 다음 페이로드 파일로 진행합니다.
moveNext 열거자의 위치를 다음 애플리케이션으로 진행합니다.

moveNext 열거자의 위치를 다음 디바이스 기능으로 확장합니다.
moveNext 열거자의 위치를 다음 <MainPackageDependency> 요소로 이동합니다.
moveNext 열거자의 위치를 다음 패키지 종속성으로 진행합니다.
moveNext 열거자의 위치를 다음 리소스로 진행합니다.
OpenPackageInfoByFullName 지정된 패키지의 패키지 정보를 엽니다.
packageFamilyNameFromFullName 지정된 패키지 전체 이름의 패키지 패밀리 이름을 가져옵니다.
packageFamilyNameFromId 지정된 패키지 식별자의 패키지 패밀리 이름을 가져옵니다.
PackageFullNameFromId 지정된 패키지 식별자(ID)의 패키지 전체 이름을 가져옵니다.
PackageIdFromFullName 지정된 패키지 전체 이름의 패키지 식별자(ID)를 가져옵니다.
PackageNameAndPublisherIdFromFamilyName 지정된 패키지 패밀리 이름의 패키지 이름 및 게시자 ID(ID)를 가져옵니다.
ParseApplicationUserModelId 애플리케이션 사용자 모델 ID를 패키지 패밀리 이름 및 PRAID(패키지 상대 애플리케이션 ID)로 분해합니다.
RemovePackageDependency 현재 프로세스의 패키지 그래프(즉, AddPackageDependency 함수를 사용하여 추가된 프레임워크 패키지 종속성에 대한 런타임 참조)에서 확인된 패키지 종속성을 제거합니다.

TryCreatePackageDependency 지정된 패키지 패밀리 이름, 최소 버전 및 추가 조건을 사용하여 현재 앱에 대한 프레임워크 패키지 종속성에 대한 설치 시간 참조를 만듭니다.
UpdateEncryptedPackage 암호화된 앱 패키지를 업데이트합니다.
UpdatePackage 앱 패키지를 업데이트합니다.
UpdatePackageManifest 앱 패키지 매니페스트를 업데이트합니다.
ValidateFileHash 이 블록 맵 파일의 블록 요소에 저장된 해시에 대해 파일 내용의 유효성을 검사합니다.
VerifyApplicationUserModelId 애플리케이션 사용자 모델 ID가 구문적으로 유효한지 여부를 확인합니다.
VerifyPackageFamilyName 패키지 패밀리 이름이 구문적으로 유효한지 여부를 확인합니다.
VerifyPackageFullName 패키지 전체 이름이 구문적으로 유효한지 여부를 확인합니다.
VerifyPackageId 패키지 ID가 구문적으로 유효한지 여부를 확인합니다.
verifyPackageRelativeApplicationId 패키지 상대 애플리케이션 ID가 구문적으로 유효한지 여부를 확인합니다.

인터페이스

테이블 확장

IAppxAppInstallerReader 앱 설치 관리자 파일에 대한 XML DOM에 대한 액세스를 제공합니다.
IAppxBlockMapBlock IAppxBlockMapBlock 인터페이스는 앱 패키지에 대한 블록 맵 파일(AppxBlockMap.xml)에 포함된 파일 내의 개별 블록을 나타내는 읽기 전용 개체를 제공합니다.
IAppxBlockMapBlocksEnumerator 블록 맵의 블록을 단일 파일로 열거합니다.
IAppxBlockMapFile 블록 맵의 파일을 나타냅니다.
IAppxBlockMapFilesEnumerator 블록 맵에서 파일을 열거합니다.
IAppxBlockMapReader 파일 특성 및 블록 해시에 대한 액세스를 제공하는 블록 맵의 읽기 전용 개체 모델을 나타냅니다.
IAppxBundleFactory 번들 패키지를 읽고 쓰기 위한 개체를 만듭니다.
IAppxBundleFactory2 번들 파일의 예상 다이제스트를 지정하는 기능을 사용하여 번들 패키지를 읽고 쓰는 개체를 만듭니다.
IAppxBundleManifestOptionalBundleInfo 번들 패키지 매니페스트의 <OptionalBundle> 요소에 대한 읽기 전용 개체 모델을 제공합니다.
IAppxBundleManifestOptionalBundleInfoEnumerator 번들의 선택적 번들 정보를 열거합니다.
IAppxBundleManifestPackageInfo 번들 패키지 매니페스트의 <Package> 요소에 대한 읽기 전용 개체 모델을 제공합니다. (IAppxBundleManifestPackageInfo)
IAppxBundleManifestPackageInfo2

번들 패키지 매니페스트의 <Package> 요소에 대한 읽기 전용 개체 모델을 제공합니다. (IAppxBundleManifestPackageInfo2)
IAppxBundleManifestPackageInfoEnumerator 번들 패키지 매니페스트에 설명된 페이로드 패키지 목록에 대한 읽기 전용 개체 모델을 제공합니다.
IAppxBundleManifestReader 번들 패키지의 매니페스트에 대한 읽기 전용 개체 모델을 제공합니다. (IAppxBundleManifestReader)
IAppxBundleManifestReader2 번들 패키지의 매니페스트에 대한 읽기 전용 개체 모델을 제공합니다. (IAppxBundleManifestReader2)
IAppxBundleReader 번들 패키지에 대한 읽기 전용 개체 모델을 제공합니다.
IAppxBundleWriter 번들 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxBundleWriter)
IAppxBundleWriter2 번들 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxBundleWriter2)
IAppxBundleWriter3 번들 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxBundleWriter3)
IAppxBundleWriter4 번들 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxBundleWriter4)
IAppxContentGroup 콘텐츠 그룹에 대한 정보를 검색합니다.
IAppxContentGroupFilesEnumerator 콘텐츠 그룹 맵에서 콘텐츠 그룹의 파일을 열거합니다.
IAppxContentGroupMapReader 콘텐츠 그룹 맵에 대한 정보를 가져옵니다.

IAppxContentGroupMapWriter 콘텐츠 그룹 맵에 대한 쓰기 전용 개체 모델을 제공합니다.
IAppxContentGroupsEnumerator 콘텐츠 그룹 맵에서 콘텐츠 그룹을 열거합니다.
IAppxDigestProvider 앱 패키징 개체의 다이제스트 문자열 표현을 검색하기 위한 API를 제공합니다.
IAppxEncryptedBundleWriter 암호화된 번들 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxEncryptedBundleWriter)
IAppxEncryptedBundleWriter2 암호화된 번들 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxEncryptedBundleWriter2)
IAppxEncryptedBundleWriter3 암호화된 번들 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxEncryptedBundleWriter3)
IAppxEncryptedPackageWriter 암호화된 앱 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxEncryptedPackageWriter)
IAppxEncryptedPackageWriter2 암호화된 앱 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxEncryptedPackageWriter2)
IAppxEncryptionFactory 패키지와 번들을 암호화, 암호 해독, 읽기 및 쓰기 위한 개체를 만듭니다.
IAppxEncryptionFactory2 Windows 앱 패키지 및 번들을 암호화, 암호 해독, 읽기 및 쓰기 위한 개체를 만듭니다. (IAppxEncryptionFactory2)
IAppxEncryptionFactory3 Windows 앱 패키지 및 번들을 암호화, 암호 해독, 읽기 및 쓰기 위한 개체를 만듭니다. (IAppxEncryptionFactory3)
IAppxEncryptionFactory4 Windows 앱 패키지 및 번들을 암호화하기 위한 개체를 만듭니다.

IAppxEncryptionFactory5 암호화된 Windows 앱 패키지 및 번들을 읽기 위한 개체를 만듭니다.
IAppxFactory 앱 패키지를 읽고 쓰기 위한 개체를 만듭니다. (IAppxFactory)
IAppxFactory2 앱 패키지를 읽고 쓰기 위한 개체를 만듭니다. (IAppxFactory2)
IAppxFactory3 앱 패키지, 매니페스트 및 앱 설치 관리자 파일을 읽고 쓰기 위한 개체를 만듭니다.
IAppxFile 패키지의 페이로드 또는 공간 파일에 대한 정보를 검색합니다.
IAppxFilesEnumerator 패키지의 페이로드 파일을 열거합니다.
IAppxManifestApplication 애플리케이션의 특성 값에 대한 액세스를 제공합니다.
IAppxManifestApplicationsEnumerator 패키지 매니페스트에 정의된 애플리케이션을 열거합니다.
IAppxManifestDeviceCapabilitiesEnumerator 패키지 매니페스트에 정의된 디바이스 기능을 열거합니다.
IAppxManifestMainPackageDependenciesEnumerator 앱 매니페스트의 <MainPackageDependency> 요소를 열거합니다.
IAppxManifestMainPackageDependency 기본 패키지 종속성의 특성 값에 대한 액세스를 제공합니다.
IAppxManifestOptionalPackageInfo 선택적 패키지 정보의 특성 값에 대한 액세스를 제공합니다.
IAppxManifestPackageDependenciesEnumerator

패키지 매니페스트에 정의된 패키지 종속성을 열거합니다.
IAppxManifestPackageDependency 다른 패키지에서 한 패키지의 종속성을 설명합니다. (IAppxManifestPackageDependency)
IAppxManifestPackageDependency2 다른 패키지에서 한 패키지의 종속성을 설명합니다. (IAppxManifestPackageDependency2)
IAppxManifestPackageId 패키지 ID에 대한 액세스를 제공합니다.
IAppxManifestPackageId2 앱 패키지 ID에 대한 액세스를 제공합니다.
IAppxManifestProperties 패키지 매니페스트의 속성 섹션에 대한 읽기 전용 액세스를 제공합니다.
IAppxManifestReader 매니페스트 요소 및 특성에 액세스하는 메서드를 제공하는 패키지 매니페스트의 개체 모델을 나타냅니다. (IAppxManifestReader)
IAppxManifestReader2 매니페스트 요소 및 특성에 액세스하는 메서드를 제공하는 패키지 매니페스트의 개체 모델을 나타냅니다. (IAppxManifestReader2)
IAppxManifestReader5 매니페스트 요소 및 특성에 액세스하는 메서드를 제공하는 패키지 매니페스트의 개체 모델을 나타냅니다. (IAppxManifestReader5)
IAppxManifestReader6 매니페스트 요소 및 특성에 액세스하는 메서드를 제공하는 패키지 매니페스트의 개체 모델을 나타냅니다. (IAppxManifestReader6)
IAppxManifestResourcesEnumerator 패키지 매니페스트에 정의된 리소스를 열거합니다.
IAppxManifestTargetDeviceFamily AppxManifest.xml대상 디바이스 패밀리에 대한 정보를 검색합니다.

IAppxPackageEditor 앱 패키지를 편집하는 기능을 제공합니다.
IAppxPackageReader 앱 패키지에 대한 읽기 전용 개체 모델을 제공합니다.
IAppxPackageWriter 앱 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxPackageWriter)
IAppxPackageWriter2 앱 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxPackageWriter2)
IAppxPackageWriter3 앱 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxPackageWriter3)
IAppxSourceContentGroupMapReader 원본 콘텐츠 그룹 맵에 대한 정보를 가져옵니다.

구조

테이블 확장

APPX_ENCRYPTED_EXEMPTIONS Windows 앱 패키지 암호화에서 제외된 파일입니다.
APPX_ENCRYPTED_PACKAGE_SETTINGS 암호화된 Windows 앱 패키지에 대한 설정입니다.
APPX_ENCRYPTED_PACKAGE_SETTINGS2 암호화된 Windows 앱 패키지 설정입니다.
APPX_KEY_INFO Windows 앱 패키지 키 정보입니다.

APPX_PACKAGE_SETTINGS 패키지를 만드는 데 사용되는 패키지 설정을 나타냅니다.
APPX_PACKAGE_WRITER_PAYLOAD_STREAM 앱 패키지에 쓸 파일의 데이터 및 메타데이터를 포함합니다.
PACKAGE_ID 이름, 버전 및 게시자와 같은 패키지 식별 정보를 나타냅니다.
PACKAGE_INFO 패키지 식별자, 전체 이름 및 설치 위치를 포함하는 패키지 식별 정보를 나타냅니다.
PACKAGE_VERSION 패키지 버전 정보를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

패키징

아티클2023. 03. 09.

패키징 기술 개요입니다.

패키징을 개발하려면 다음 헤더가 필요합니다.

- msopc.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 포장

열거형

OPC_CANONICALIZATION_METHOD XML 태그에 적용할 정식화 메서드에 대해 설명합니다.
OPC_CERTIFICATE_EMBEDDING_OPTION 서명에 사용되는 인증서의 스토리지 위치를 설명합니다.
OPC_COMPRESSION_OPTIONS 파트 콘텐츠를 압축하는 방법을 설명합니다.
OPC_READ_FLAGS 패키지 구성 요소를 캐싱하고 ECMA-376 OpenXML, 1 버전, 2부:_Open OPC(패키징 규칙) 규칙 요구 사항에 대해 유효성을 검사하기 위한 읽기 설정을 설명합니다.
OPC_RELATIONSHIP_SELECTOR IOpcRelationshipSelector::GetSelectionCriterion 메서드의 selectionCriterion 매개 변수를 해석하는 방법을 설명합니다.
OPC_RELATIONSHIPS_SIGNING_OPTION IOpcSignatureRelationshipReference 인터페이스가 나타내는 참조가 관계 집합 개체에서 관계 개체로 표시되는 관계의 전체 또는 하위 집합을 참조하는지 여부를 설명합니다.

OPC_SIGNATURE_TIME_FORMAT 서명이 만들어진 시점의 레코드인 signingTime 매개 변수를 IOpcDigitalSignature::GetSigningTime 메서드의 해석하는 방법을 설명합니다.
OPC_SIGNATURE_VALIDATION_RESULT 서명의 상태를 나타냅니다.
OPC_STREAM_IO_MODE 스트림의 읽기/쓰기 상태를 설명합니다.
OPC_URI_TARGET_MODE 관계의 대상 모드를 나타냅니다.
OPC_WRITE_FLAGS serialization 개체가 패키지를 생성하는 데 사용하는 인코딩 메서드에 대해 설명합니다.

Functions

추가 집합에 인증서를 추가합니다.
복제 현재 IOpcCertificateEnumerator 인터페이스 포인터 및 모든 하위 항목의 복사본을 만듭니다.
복제 현재 IOpcDigitalSignatureEnumerator 인터페이스 포인터 및 모든 하위 항목의 복사본을 만듭니다.
복제 현재 열거자 및 모든 하위 항목의 복사본을 만듭니다. (IOpcPartEnumerator.Clone)
복제 현재 열거자 및 모든 하위 항목의 복사본을 만듭니다. (IOpcRelationshipEnumerator.Clone)

복제 현재 IOpcRelationshipSelectorEnumeratorinterface 포인터 및 모든 하위 항목의 복사본을 만듭니다.
복제 현재 IOpcSignatureCustomObjectEnumerator 인터페이스 포인터 및 모든 하위 항목의 복사본을 만듭니다.
복제 현재 IOpcSignaturePartReferenceEnumerator 인터페이스 포인터 및 모든 하위 항목의 복사본을 만듭니다.
복제 현재 IOpcSignatureReferenceEnumerator 인터페이스 포인터 및 모든 하위 항목의 복사본을 만듭니다.
복제 현재 IOpcSignatureRelationshipReferenceEnumerator 인터페이스 포인터 및 모든 하위 항목의 복사본을 만듭니다.
CombinePartUri 지정된 상대 URI에서 참조하는 파트의 파트 이름을 형성합니다.
ComparePartUri 현재 파트 URI 개체와 지정된 파트 URI 개체가 나타내는 URI가 동일한지 여부를 나타내는 정수 값을 반환합니다.
만들기 서명할 관계의 하위 집합을 선택하는 방법을 나타내는 IOpcRelationshipSelector 인터페이스 포인터를 만들고 집합에 새 포인터를 추가합니다.
만들기 서명에서 애플리케이션별 Object 요소를 나타내는 IOpcSignatureCustomObject 인터페이스 포인터를 만들고 집합에 새 인터페이스를 추가합니다.
만들기 서명할 파트에 대한 참조를 나타내는 IOpcSignaturePartReference 인터페이스 포인터를 만들고 집합에 새 인터페이스를 추가합니다.

만들기 서명할 XML 요소에 대한 참조를 나타내는 IOpcSignatureReference 인터페이스 포인터를 만듭니다.
만들기 관계 부분에 대한 참조를 나타내는 IOpcSignatureRelationshipReference 인터페이스 포인터를 만들고 집합에 새 인터페이스 포인터를 추가합니다.
CreateDigitalSignatureManager 패키지 개체에 대한 디지털 서명 관리자 개체를 만듭니다.
CreatePackage 빈 패키지를 나타내는 패키지 개체를 만듭니다.
CreatePackageRootUri 패키지의 루트를 나타내는 OPC URI 개체를 만듭니다.
CreatePart 파트를 나타내는 파트 개체를 만들고 개체의 IOpcPart 인터페이스에 대한 포인터를 집합에 추가합니다.
CreatePartUri 파트 이름을 나타내는 파트 URI 개체를 만듭니다.
CreateRelationship 지정된 관계를 나타내는 관계 개체를 만든 다음 개체의 IOpcRelationship 인터페이스에 대한 포인터를 설정에 추가합니다.
CreateRelationshipSelectorSet Create 메서드의 selectorSet 매개 변수 값으로 사용되는 IOpcRelationshipSelectorSet 인터페이스 포인터를 만듭니다.
CreateSigningOptions IOpcSigningOptions 인터페이스 포인터를 만듭니다.
CreateStreamOnFile 파일을 통해 스트림을 만듭니다.

삭제 집합에서 지정된 IOpcRelationshipSelector 인터페이스 포인터를 삭제합니다.
삭제 집합에서 지정된 IOpcSignatureCustomObject 인터페이스 포인터를 삭제합니다.
삭제 집합에서 지정된 IOpcSignaturePartReference 인터페이스 포인터를 삭제합니다.
삭제 집합에서 지정된 IOpcSignatureReference 인터페이스 포인터를 삭제합니다.
삭제 집합에서 지정된 IOpcSignatureRelationshipReference 인터페이스 포인터를 삭제합니다.
DeletePart 집합에서 지정된 파트 개체의 IOpcPart 인터페이스 포인터를 삭제합니다.
DeleteRelationship 집합에서 지정된 IOpcRelationship 인터페이스 포인터를 삭제합니다.
GetCanonicalizationMethod serialize된 서명의 SignedInfo 요소에 적용된 정식화 메서드를 가져옵니다.
GetCertificateEmbeddingOption 서명에 사용할 인증서 패키지의 스토리지 위치를 지정하는 값을 가져옵니다.
GetCertificateEnumerator 서명에 사용되는 인증서의 열거자를 가져옵니다.
GetCertificateSet IOpcCertificateSet 인터페이스 포인터를 가져옵니다.
GetCompressionOptions 파트 콘텐츠가 압축되는 방식을 설명하는 값을 가져옵니다.

GetContentStream 파트 콘텐츠에 대한 읽기/쓰기 액세스를 제공하는 스트림을 가져옵니다.
GetContentType 파트 콘텐츠의 미디어 형식을 가져옵니다.
GetContentType 참조된 파트의 콘텐츠 형식을 가져옵니다.
GetCurrent 열거자의 현재 위치에 있는 CERT_CONTEXT 구조를 가져옵니다.
GetCurrent 열거자의 현재 위치에 있는 IOpcDigitalSignature 인터페이스 포인터를 가져옵니다.
GetCurrent 열거자의 현재 위치에 있는 IOpcPart 인터페이스 포인터를 가져옵니다.
GetCurrent 열거자의 현재 위치에 있는 IOpcRelationship 인터페이스 포인터를 가져옵니다.
GetCurrent 열거자의 현재 위치에 있는 IOpcRelationshipSelector 인터페이스 포인터를 가져옵니다.
GetCurrent 열거자의 현재 위치에 있는 IOpcSignatureCustomObject 인터페이스를 가져옵니다.
GetCurrent 열거자의 현재 위치에 있는 IOpcSignaturePartReference 인터페이스 포인터를 가져옵니다.
GetCurrent 열거자의 현재 위치에 있는 IOpcSignatureReference 인터페이스 포인터를 가져옵니다.
GetCurrent 열거자의 현재 위치에 있는 IOpcSignatureRelationshipReference 인터페이스 포인터를 가져옵니다.

GetCustomObjectEnumerator 서명 태그에서 애플리케이션별 개체 요소를 나타내는 IOpcSignatureCustomObject 인터페이스 포인터의 열거자를 가져옵니다.
GetCustomObjectSet IOpcSignatureCustomObjectSet 인터페이스를 가져옵니다.
GetCustomReferenceEnumerator 서명된 애플리케이션별 XML 요소에 대한 참조를 나타내는 IOpcSignatureReference 인터페이스 포인터의 열거자를 가져옵니다.
GetCustomReferenceSet IOpcSignatureReferenceSet 인터페이스 포인터를 가져옵니다.
GetDefaultDigestMethod 서명할 개체의 다이제스트 값을 계산하는 데 사용할 기본 다이제스트 메서드를 가져옵니다.
GetDigestMethod 파트가 서명된 경우 참조된 파트의 일부 콘텐츠에 사용할 다이제스트 메서드를 가져옵니다.
GetDigestMethod 요소가 서명된 경우 참조된 XML 요소에서 사용할 다이제스트 메서드를 가져옵니다.
GetDigestMethod 선택한 관계의 관계 태그에 사용할 다이제스트 메서드를 가져옵니다.
GetDigestValue 파트가 서명된 경우 참조된 파트의 파트 콘텐츠에 대해 계산되는 다이제스트 값을 가져옵니다.
GetDigestValue 요소가 서명된 경우 참조된 XML 요소에 대해 계산되는 다이제스트 값을 가져옵니다.
GetDigestValue 선택한 관계가 서명될 때 계산된 다이제스트 값을 가져옵니다.
GetEnumerator 집합의 인증서 열거자를 가져옵니다.

GetEnumerator 집합에서 IOpcPart 인터페이스 포인터의 열거자를 가져옵니다.
GetEnumerator 집합에서 IOpcRelationshipSelector 인터페이스 포인터의 열거자를 가져옵니다.
GetEnumerator 집합에서 IOpcRelationship 인터페이스 포인터의 열거자를 가져옵니다.
GetEnumerator 집합에서 IOpcSignatureCustomObject 인터페이스 포인터의 열거자를 가져옵니다.
GetEnumerator 집합에서 IOpcSignaturePartReference 인터페이스 포인터의 열거자를 가져옵니다.
GetEnumerator 집합에서 IOpcSignatureReference 인터페이스 포인터의 열거자를 가져옵니다.
GetEnumerator 집합에서 IOpcSignatureRelationshipReference 인터페이스 포인터의 열거자를 가져옵니다.
GetEnumeratorForType 지정된 관계 형식이 있는 관계의 표현을 가리키는 집합의 IOpcRelationship 인터페이스 포인터 열거자를 가져옵니다.
GetId 관계의 고유 식별자를 가져옵니다.
GetId 참조의 식별자를 가져옵니다.
GetName 부품 이름을 나타내는 파트 URI 개체를 가져옵니다.
GetNamespaces 서명 태그의 Signature 요소에 대한 접두사 및 네임스페이스 매핑을 가져옵니다.

GetPackageObjectReference 서명된 패키지별 Object 요소에 대한 참조를 나타내는 IOpcSignatureReference 인터페이스 포인터를 가져옵니다.
GetPart 집합에서 지정된 부분을 나타내는 파트 개체를 가져옵니다.
GetPartName 참조된 파트의 부품 이름을 가져옵니다.
GetPartSet IOpcPart 인터페이스 포인터가 포함된 파트 집합 개체를 가져옵니다.
GetRelationship 지정된 관계를 나타내는 집합에서 관계 개체를 가져옵니다.
GetRelationshipsContentStream 집합이 나타내는 관계 파트의 파트 콘텐츠를 포함하는 읽기 전용 스트림을 가져옵니다.
GetRelationshipSelectorEnumerator 참조된 관계 부분에서 관계의 하위 집합을 선택하는 데 사용되는 기술을 나타내는 IOpcRelationshipSelector 인터페이스 포인터의 열거자를 가져옵니다.
GetRelationshipSet 패키지 관계를 저장하는 관계 부분을 나타내는 관계 집합 개체를 가져옵니다.
GetRelationshipSet 파트가 있는 관계를 원본으로 저장하는 관계 부분을 나타내는 관계 집합 개체를 가져옵니다.
GetRelationshipSigningOption 참조된 관계 부분에 저장된 관계의 전체 또는 하위 집합이 선택되어 있는지 여부를 설명하는 값을 가져옵니다.
GetRelationshipsPartUri 현재 OPC URI 개체가 나타내는 원본 URI가 있는 관계를 저장하는 관계 파트의 파트 이름을 가져옵니다.

GetRelationshipType 관계 유형을 가져옵니다.
GetRelativeUri 현재 OPC URI 개체가 나타내는 URI를 기준으로 지정된 부분에 대한 상대 URI를 형성합니다.
GetSelectionCriterion 서명을 위해 참조할 관계를 선택하는 데 사용되는 문자열을 가져옵니다.
GetSelectorType 서명을 위해 관계를 참조하도록 선택하는 방법을 설명하는 값을 가져옵니다.
GetSignatureEnumerator 패키지 서명을 나타내는 IOpcDigitalSignature 인터페이스 포인터의 열거자를 가져옵니다.
GetSignatureId 서명 태그의 Signature 요소에서 ID 특성의 값을 가져옵니다.
GetSignatureId Signature 요소에서 Id 특성의 값을 가져옵니다.
GetSignatureMethod 서명 태그의 SignatureValue 요소에서 값을 계산하는 데 사용되는 서명 메서드를 가져옵니다.
GetSignatureMethod 서명의 SignatureValue 요소로 serialize되는 SignedInfo 요소의 해시 값을 계산하고 암호화하는 데 사용할 서명 메서드를 가져옵니다.
GetSignatureOriginPartName 디지털 서명 원본 파트의 파트 이름을 나타내는 IOpcPartUri 인터페이스 포인터를 가져옵니다.
GetSignaturePartName 서명 태그가 포함된 파트의 파트 이름을 가져옵니다.
GetSignaturePartName 서명 태그가 저장될 서명 파트의 파트 이름을 가져옵니다.

GetSignaturePartReferenceEnumerator 서명된 파트에 대한 참조를 나타내는 IOpcSignaturePartReference 인터페이스 포인터의 열거자를 가져옵니다.
GetSignaturePartReferenceSet IOpcSignaturePartReferenceSet 인터페이스를 가져옵니다.
GetSignatureRelationshipReferenceEnumerator 서명된 관계에 대한 참조를 나타내는 IOpcSignatureRelationshipReference 인터페이스 포인터의 열거자를 가져옵니다.
GetSignatureRelationshipReferenceSet IOpcSignatureRelationshipReferenceSet 인터페이스 포인터를 가져옵니다.
GetSignatureValue 서명 태그의 SignatureValue 요소에서 디코딩된 값을 가져옵니다.
GetSignatureXml 서명 태그를 가져옵니다.
GetSigningTime 서명이 생성된 시간을 나타내는 문자열을 가져옵니다.
GetSourceUri 관계 부분에 저장된 관계의 원본 URI를 가져옵니다. 현재 파트 URI 개체는 관계 파트의 파트 이름을 나타냅니다.
GetSourceUri 관계 원본의 URI를 가져옵니다.
GetSourceUri 참조된 관계 부분에 저장된 관계의 원본 URI를 가져옵니다.
GetTargetMode 관계의 대상이 패키지 내부인지 아니면 외부인지를 설명하는 값을 가져옵니다.

GetTargetUri 관계 대상의 URI를 가져옵니다.
GetTimeFormat GetSigningTime 메서드에서 반환된 문자열의 형식을 가져옵니다.
GetTimeFormat IOpcDigitalSignature::GetSigningTime 메서드에서 검색한 문자열의 형식을 가져옵니다.
GetTransformMethod 파트가 서명된 경우 참조된 파트의 일부 콘텐츠에 사용할 정식화 메서드를 가져옵니다.
GetTransformMethod 요소가 서명된 경우 참조된 XML 요소에서 사용할 정식화 메서드를 가져옵니다.
GetTransformMethod 선택한 관계가 서명될 때 관계 태그에 사용할 정식화 메서드를 가져옵니다.
GetType 참조된 XML 요소의 형식을 나타내는 문자열을 가져옵니다.
GetUri 참조된 XML 요소의 URI를 가져옵니다.
Getxml 애플리케이션별 Object 요소의 XML 태그를 가져옵니다.
IsRelationshipsPartUri 현재 파트 URI 개체가 관계 파트의 파트 이름을 나타내는지 여부를 나타내는 값을 반환합니다.
MoveNext 열거자의 현재 위치를 다음 CERT_CONTEXT 구조체로 이동합니다.
MoveNext 열거자의 현재 위치를 다음 IOpcDigitalSignature 인터페이스 포인터로 이동합니다.

MoveNext 열거자의 현재 위치를 다음 IOpcPart 인터페이스 포인터로 이동합니다.
MoveNext 열거자의 현재 위치를 다음 IOpcRelationship 인터페이스 포인터로 이동합니다.
MoveNext 열거자의 현재 위치를 다음 IOpcRelationshipSelectorinterface 포인터로 이동합니다.
MoveNext 열거자의 현재 위치를 다음 IOpcSignatureCustomObject 인터페이스 포인터로 이동합니다.
MoveNext 열거자의 현재 위치를 다음 IOpcSignaturePartReference 인터페이스 포인터로 이동합니다.
MoveNext 열거자의 현재 위치를 다음 IOpcSignatureReference 인터페이스 포인터로 이동합니다.
MoveNext 열거자의 현재 위치를 다음 IOpcSignatureRelationshipReference 인터페이스 포인터로 이동합니다.
MovePrevious 열거자의 현재 위치를 이전 CERT_CONTEXT 구조체로 이동합니다.
MovePrevious 열거자의 현재 위치를 이전 IOpcDigitalSignature 인터페이스 포인터로 이동합니다.
MovePrevious 열거자의 현재 위치를 이전 IOpcPart 인터페이스 포인터로 이동합니다.
MovePrevious 열거자의 현재 위치를 이전 IOpcRelationship 인터페이스 포인터로 이동합니다.
MovePrevious 열거자의 현재 위치를 이전 IOpcRelationshipSelectorinterface 포인터로 이동합니다.

MovePrevious 열거자의 현재 위치를 이전 IOpcSignatureCustomObjectinterface 포인터로 이동합니다.
MovePrevious 열거자의 현재 위치를 이전 IOpcSignaturePartReference 인터페이스 포인터로 이동합니다.
MovePrevious 열거자의 현재 위치를 이전 IOpcSignatureReferenceinterface 포인터로 이동합니다.
MovePrevious 열거자의 현재 위치를 이전 IOpcSignatureRelationshipReference 인터페이스 포인터로 이동합니다.
PartExists 지정된 부분이 집합의 파트 개체로 표시되는지 여부를 나타내는 값을 가져옵니다.
ReadPackageFromStream 스트림에서 패키지 데이터를 역직렬화하고 읽는 패키지를 나타내는 패키지 개체를 만듭니다.
RelationshipExists 지정된 관계가 집합에서 관계 개체로 표시되는지 여부를 나타내는 값을 가져옵니다.
제거 집합에서 지정된 인증서를 제거합니다.
RemoveSignature 패키지에서 서명 태그를 저장하는 지정된 서명 부분을 제거합니다.
ReplaceSignatureXml 지정된 서명 부분에 저장된 기존 서명 태그를 바꿉니다.
SetCertificateEmbeddingOption 서명에 사용할 인증서의 스토리지 위치를 설정합니다.
SetDefaultDigestMethod 서명할 개체의 다이제스트 값을 계산하는 데 사용할 기본 다이제스트 메서드를 설정합니다.

SetSignatureId Signature 요소의 ID 특성 값을 설정합니다. (IOpcSigningOptions.SetSignatureId)
SetSignatureMethod 서명의 SignatureValue 요소에 포함될 SignedInfo 요소의 해시 값을 계산하고 암호화하는 데 사용할 서명 메서드를 설정합니다.
SetSignatureOriginPartName 디지털 서명 원본 부분의 파트 이름을 지정된 IOpcPartUri 인터페이스 포인터가 나타내는 이름으로 설정합니다.
SetSignaturePartName 서명 태그가 저장될 서명 파트의 파트 이름을 설정합니다.
SetTimeFormat IOpcDigitalSignature::GetSigningTime 메서드로 검색된 문자열의 형식을 설정합니다.
서명 지정된 인증서 및 IOpcSigningOptions 인터페이스 포인터를 사용하여 서명을 생성하여 패키지에 서명합니다.
유효성 검사 지정된 인증서를 사용하여 지정된 패키지 서명의 유효성을 검사합니다.
WritePackageToStream 패키지 개체가 나타내는 패키지를 직렬화합니다.

인터페이스

IOpcCertificateEnumerator CERT_CONTEXT 구조체에 대한 포인터의 읽기 전용 열거자입니다.
IOpcCertificateSet 서명과 함께 사용할 순서가 지정되지 않은 인증서 집합입니다.

IOpcDigitalSignature 패키지 디지털 서명을 나타냅니다.
IOpcDigitalSignatureEnumerator IOpcDigitalSignature 인터페이스 포인터의 읽기 전용 열거자입니다.
IOpcDigitalSignatureManager 패키징 API 개체로 표현되는 패키지에 대한 패키징 디지털 서명 인터페이스에 대한 액세스를 제공합니다.
IOpcFactory 패키징 API 개체를 만들고 패키지 저장 및 로드를 지원합니다.
IOpcPackage 패키지를 나타내며 패키지의 파트 및 관계에 액세스하는 메서드를 제공합니다.
IOpcPart 데이터가 들어 있고 관계 부분이 아닌 부분을 나타냅니다.
IOpcPartEnumerator IOpcPart 인터페이스 포인터의 읽기 전용 열거자입니다.
IOpcPartSet 관계 부분이 아닌 패키지의 파트를 나타내는 파트 개체에 대한 순서가 지정되지 않은 IOpcPart 인터페이스 포인터 집합입니다.
IOpcPartUri 파트의 파트 이름을 나타냅니다.
IOpcRelationship 요소 또는 패키지인 원본과 대상 간의 링크인 관계를 나타냅니다.
IOpcRelationshipEnumerator IOpcRelationship 인터페이스 포인터의 읽기 전용 열거자입니다.
IOpcRelationshipSelector 관계 부분에서 서명을 위해 참조할 관계를 선택하는 방법을 나타냅니다.

IOpcRelationshipSelectorEnumerator IOpcRelationshipSelector 인터페이스 포인터의 읽기 전용 열거자입니다.
IOpcRelationshipSelectorSet 서명에 대한 관계를 식별하는 데 사용되는 선택 조건을 나타내는 순서가 지정되지 않은 IOpcRelationshipSelector 인터페이스 포인터 집합입니다.
IOpcRelationshipSet 관계 부분을 관계 개체에 대한 IOpcRelationship 인터페이스 포인터의 순서가 지정되지 않은 집합으로 나타냅니다.
IOpcSignatureCustomObject 서명되었거나 서명될 애플리케이션별 Object 요소를 나타냅니다.
IOpcSignatureCustomObjectEnumerator IOpcSignatureCustomObject 인터페이스 포인터의 읽기 전용 열거자입니다.
IOpcSignatureCustomObjectSet 애플리케이션별 개체 요소의 XML 태그를 포함하는 순서가 지정되지 않은 IOpcSignatureCustomObject 인터페이스 포인터 집합입니다.
IOpcSignaturePartReference 서명되었거나 서명될 파트에 대한 참조를 나타냅니다.
IOpcSignaturePartReferenceEnumerator IOpcSignaturePartReference 인터페이스 포인터의 읽기 전용 열거자입니다.
IOpcSignaturePartReferenceSet 서명할 파트에 대한 참조를 나타내는 순서가 지정되지 않은 IOpcSignaturePartReference 인터페이스 포인터 집합입니다.
IOpcSignatureReference 서명되었거나 서명될 XML 태그에 대한 참조를 나타냅니다.
IOpcSignatureReferenceEnumerator IOpcSignatureReference 인터페이스 포인터의 읽기 전용 열거자입니다.

IOpcSignatureReferenceSet 서명할 XML 요소에 대한 참조를 나타내는 순서가 지정되지 않은 IOpcSignatureReference 인터페이스 포인터 집합입니다.
IOpcSignatureRelationshipReference 서명되었거나 서명될 관계가 포함된 관계 부분에 대한 참조를 나타냅니다.
IOpcSignatureRelationshipReferenceEnumerator IOpcSignatureRelationshipReference 인터페이스 포인터의 읽기 전용 열거자입니다.
IOpcSignatureRelationshipReferenceSet 서명할 관계가 포함된 관계 부분에 대한 참조를 나타내는 순서가 지정되지 않은 IOpcSignatureRelationshipReference 인터페이스 포인터 집합입니다.
IOpcSigningOptions 서명을 생성하는 데 필요한 정보를 설정하고 액세스하는 메서드를 제공합니다.
IOpcUri 패키지 루트 또는 패키지 루트를 기준으로 하는 파트의 URI를 나타냅니다.

자녀

아티클2023. 08. 22.

자녀 보호 기술 개요입니다.

자녀 보호 기능을 개발하려면 다음 헤더가 필요합니다.

- wpcapi.h
- wpcevent.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 자녀

열거형

WPC_ARGS_CONVERSATIONINITEVENT 대화 시작에 대한 정보를 나타냅니다.
WPC_ARGS_CONVERSATIONJOINEVENT 기존 대화에 참가하는 방법에 대한 정보를 나타냅니다.
WPC_ARGS_CONVERSATIONLEAVEEVENT 대화에서 나가는 방법에 대한 정보를 나타냅니다.
WPC_ARGS_CUSTOMEVENT 일반 이벤트에서 다루지 않는 사용자 정의 이벤트에 대한 정보를 나타냅니다.
WPC_ARGS_EMAILCONTACTEVENT 전자 메일을 사용하여 다른 사람에게 연락하는 방법에 대한 정보를 나타냅니다.
WPC_ARGS_EMAILRECEIEVEDEVENT 받은 전자 메일 메시지에 대한 정보를 나타냅니다.
WPC_ARGS_EMAILSENTEVENT 전송된 전자 메일 메시지에 대한 정보를 나타냅니다.

WPC_ARGS_FILEDOWNLOADEVENT 다운로드된 파일에 대한 정보를 나타냅니다.
WPC_ARGS_GAMESTARTEVENT 컴퓨터 게임의 시작에 대한 정보를 나타냅니다.
WPC_ARGS_IMCONTACTEVENT 인스턴트 메시징 애플리케이션을 사용하여 다른 사람에게 연락하는 방법에 대한 정보를 나타냅니다.
WPC_ARGS_IMFEATUREEVENT 인스턴트 메시징 상호 작용의 기능에 대한 정보를 나타냅니다.
WPC_ARGS_MEDIADOWNLOADEVENT 미디어 파일 다운로드에 대한 정보를 나타냅니다.
WPC_ARGS_MEDIAPLAYBACKEVENT 미디어 파일의 재생에 대한 정보를 나타냅니다.
WPC_ARGS_SAFERAPPBLOCKED 차단되는 안전한 애플리케이션에 대한 정보를 나타냅니다.
WPC_ARGS_SETTINGSCHANGEEVENT 사용자가 변경한 설정에 대한 정보를 나타냅니다.
WPC_ARGS_URLVISITEVENT 본 웹 사이트의 주소 URL에 대한 정보를 나타냅니다.
WPC_MEDIA_EXPLICIT 미디어 파일의 명시적 등급에 대한 정보를 나타냅니다.
WPC_MEDIA_TYPE 액세스하는 미디어 파일의 형식에 대한 정보를 나타냅니다.
WPCFLAG_IM_FEATURE 인스턴트 메시징 상호 작용 중에 액세스하는 기능에 대한 정보를 나타냅니다.

WPCFLAG_IM_LEAVE 참가자가 인스턴트 메시징 상호 작용을 떠나는 시기에 대한 정보를 나타냅니다.
WPCFLAG_ISBLOCKED 사용이 차단된 이벤트 및 현재 위치에 있는 컨트롤에 대한 정보를 나타냅니다.
WPCFLAG_LOGOFF_TYPE 사용된 로그오프 메서드의 유형에 대한 정보를 나타냅니다.

Functions

구성 자녀 보호 제어판 사용자 타일을 클릭하면 현재 공급자가 호출됩니다. 이 메서드를 사용하면 구성을 변경할 수 있습니다.
사용 안 함 타사 애플리케이션에 현재 공급자가 아님을 알 수 있습니다.
사용 타사 애플리케이션이 새 현재 공급자로 선택되었음을 알 수 있습니다.
GetCurrent 현재 공급자의 GUID를 검색합니다.
GetGamesSettings 지정된 사용자의 게임 제한 설정에 대한 인터페이스에 대한 포인터를 검색합니다.
GetLastSettingsChangeTime 구성 설정이 마지막으로 업데이트된 시간을 검색합니다.
GetRestrictions 웹 제한, 시간 제한 또는 게임 제한이 적용되는지 여부를 결정합니다.
설정가져오기

웹 제한 설정을 검색합니다.
GetUserSettings 지정된 사용자에 대한 일반 설정에 대한 인터페이스에 대한 포인터를 검색합니다.
GetUserSummary 자녀 보호 제어판 사용하여 각 사용자의 정보를 검색합니다.
GetVisibility 자녀 보호 사용자 인터페이스의 표시 유형을 나타냅니다.
GetWebFilterInfo 현재 활성 웹 콘텐츠 필터의 이름과 식별자를 검색합니다.
GetWebSettings 지정된 사용자의 웹 제한 설정에 대한 인터페이스에 대한 포인터를 검색합니다.
IsBlocked 지정된 게임이 실행에서 차단되는지 여부를 확인합니다.
IsLoggingRequired IWPCSettings 인터페이스를 가져올 때 활동 로깅을 수행해야 하는지 여부를 결정합니다.
RequestOverride 현재 공급자가 구성 재정의를 사용하도록 설정하도록 호출됩니다.
RequestURLOverride 자녀 보호 웹 제한 하위 시스템이 지정된 기본 및 하위 URL을 허용된 상태로 설정하도록 요청합니다.

인터페이스

IWindowsParentalControls 최소 준수 API의 모든 설정 인터페이스에 액세스할 수 있습니다.

IWindowsParentalControlsCore . (IWindowsParentalControlsCore)
IWPCGamesSettings 사용자의 게임 설정에 액세스합니다.
IWPCProviderConfig 타사에서 구현하는 구성 메서드를 노출합니다.
IWPCProviderState 타사에서 구현하는 공급자 상태 메서드를 노출합니다.
IWPCProviderSupport 타사 공급자가 현재 사용 가능한 공급자를 쿼리할 수 있도록 하는 메서드를 노출합니다.
IWPCSettings 사용자의 일반 설정에 액세스합니다.
IWPCWebSettings 사용자의 웹 제한 설정에 액세스합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

피어 투 피어

아티클2023. 03. 16.

피어 투 피어 기술 개요입니다.

피어 투 피어를 개발하려면 다음 헤더가 필요합니다.

- drt.h
- p2p.h
- peerdist.h
- pnrpdef.h
- pnrpns.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 피어 투 피어

열거형

DRT_ADDRESS_FLAGS DRT_ADDRESS_FLAGS 열거형입니다.
DRT_EVENT_TYPE DRT_EVENT_TYPE 열거형은 분산 라우팅 테이블에서 발생할 수 있는 이벤트 집합을 정의합니다.
DRT_LEAFSET_KEY_CHANGE_TYPE DRT_LEAFSET_KEY_CHANGE_TYPE 열거형은 로컬로 등록된 키의 리프 집합에 있는 노드에서 발생할 수 있는 변경 내용 집합을 정의합니다.
DRT_MATCH_TYPE DRT_MATCH_TYPE 열거형은 DrtStartSearch API를 사용하여 검색을 시작한 후 DrtGetSearchResult에서 반환된 검색 결과의 정확도를 정의합니다.
DRT_REGISTRATION_STATE DRT_REGISTRATION_STATE 열거형은 등록된 키에 대한 법적 상태 집합을 정의합니다.

DRT_SCOPE DRT_SCOPE 열거형은 DrtCreateIpv6UdpTransport에서 만든 IPv6 UDP 전송을 사용하는 동안 DRT가 작동하는 IPv6 범위 집합을 정의합니다.
DRT_SECURITY_MODE DRT_SECURITY_MODE 열거형은 DRT에 대해 가능한 보안 모드를 정의합니다. 보안 모드는 DRT_SETTINGS 구조의 필드에 의해 지정됩니다.
DRT_STATUS DRT_STATUS 열거형은 로컬 DRT 인스턴스의 상태를 정의합니다.
PEER_APPLICATION_REGISTRATION_TYPE PEER_APPLICATION_REGISTRATION_TYPE 열거형은 피어 애플리케이션 등록 플래그 집합을 정의합니다.
PEER_CHANGE_TYPE PEER_CHANGE_TYPE 열거형은 피어 이벤트의 피어 개체, 엔드포인트 또는 애플리케이션에서 수행된 변경 내용 집합을 정의합니다. 변경 유형과 연결된 피어 이벤트를 한정하는 데 사용됩니다.
PEER_COLLAB_EVENT_TYPE PEER_COLLAB_EVENT_TYPE 열거형은 피어 협업 네트워크 이벤트 인프라에서 피어에서 발생할 수 있는 이벤트 집합을 정의합니다.
PEER_CONNECTION_FLAGS PEER_CONNECTION_FLAGS 열거형은 피어가 가질 수 있는 연결 유형을 지정합니다.
PEER_CONNECTION_STATUS PEER_CONNECTION_STATUS 열거형은 피어 직접 또는 인접 연결의 상태를 지정합니다.
PEER_GRAPH_EVENT_TYPE PEER_GRAPH_EVENT_TYPE 열거형은 애플리케이션에 알림을 받을 피어 이벤트 유형을 지정합니다.
PEER_GRAPH_PROPERTY_FLAGS PEER_GRAPH_PROPERTY_FLAGS 열거형은 피어 그래프의 속성을 지정합니다.
PEER_GRAPH_SCOPE PEER_GRAPH_SCOPE 열거형은 피어 그래프의 네트워크 범위를 지정합니다.

PEER_GRAPH_STATUS_FLAGS PEER_GRAPH_STATUS_FLAGS 열거형은 피어 그래프 내 노드의 현재 상태를 표시하는 플래그 집합입니다.
PEER_GROUP_AUTHENTICATION_SCHEME 피어 그룹에 조인하는 피어를 인증하는 데 사용할 수 있는 가능한 인증 체계 집합을 정의합니다.
PEER_GROUP_EVENT_TYPE PEER_GROUP_EVENT_TYPE 열거형에는 피어 그룹 내에서 발생할 수 있는 특정 피어 이벤트 유형이 포함됩니다.
PEER_GROUP_ISSUE_CREDENTIAL_FLAGS . (PEER_GROUP_ISSUE_CREDENTIAL_FLAGS)
PEER_GROUP_PROPERTY_FLAGS PEER_GROUP_PROPERTY_FLAGS 플래그는 다양한 피어 그룹 멤버 자격 설정을 지정하는 데 사용됩니다.
PEER_GROUP_STATUS PEER_GROUP_STATUS 플래그는 피어 그룹에 연결이 있는지 여부를 나타냅니다.
PEER_INVITATION_RESPONSE_TYPE 피어 협업 작업을 시작하기 위해 초대에 수신된 응답 유형을 정의합니다.
PEER_MEMBER_CHANGE_TYPE PEER_MEMBER_CHANGE_TYPE 열거형은 피어에 대해 가능한 피어 그룹 멤버 자격 및 현재 상태 집합을 정의합니다.
PEER_MEMBER_FLAGS PEER_MEMBER_FLAGS 플래그를 사용하면 애플리케이션에서 PeerGroupEnumMembers 함수가 호출될 때 모든 멤버 또는 현재 멤버만 열거할지 여부를 지정하거나 피어 그룹 내에 멤버가 있는지 여부를 나타낼 수 있습니다.
PEER_NODE_CHANGE_TYPE PEER_NODE_CHANGE_TYPE 열거형은 피어 노드 그래프 상태 변경 유형을 지정합니다.

PEER_PRESENCE_STATUS PEER_PRESENCE_STATUS 열거형은 피어 협업 네트워크에 참여하는 피어에서 사용할 수 있는 가능한 현재 상태 설정 집합을 정의합니다.
PEER_PUBLICATION_SCOPE 피어 개체 또는 데이터의 게시에 대한 범위 집합을 정의합니다.
PEER_RECORD_CHANGE_TYPE PEER_RECORD_CHANGE_TYPE 열거형은 레코드에 발생할 수 있는 변경 내용을 지정합니다.
PEER_RECORD_FLAGS PEER_RECORD_FLAGS 열거형은 피어 레코드 동작에 대한 플래그 집합을 지정합니다.
PEER_SIGNIN_FLAGS PEER_SIGNIN_FLAGS 열거형은 피어가 피어 협업 네트워크에 로그인할 때 사용할 수 있는 피어 프레즌스 게시 동작 집합을 정의합니다.
PEER_WATCH_PERMISSION 피어 연락처가 연락처에서 현재 상태 업데이트를 받을 수 있는지 여부를 정의합니다.
PEERDIST_CLIENT_INFO_BY_HANDLE_CLASS PEERDIST_CLIENT_INFO_BY_HANDLE_CLASS 열거형은 가능한 클라이언트 정보 값을 정의합니다.
PEERDIST_STATUS PEERDIST_STATUS 열거형은 피어 배포 서비스의 가능한 상태 값을 정의합니다.
PNRP_CLOUD_FLAGS PNRP_CLOUD_FLAGS 열거형은 클라우드 이름의 유효성을 지정합니다.
PNRP_CLOUD_STATE PNRP_CLOUD_STATE 열거형은 PNRP 클라우드가 있을 수 있는 다양한 상태를 지정합니다.
PNRP_RESOLVE_CRITERIA PNRP_RESOLVE_CRITERIA 열거형은 PNRP가 검색을 확인하는 데 사용하는 조건을 지정합니다.

Functions

DrtClose DrtClose 함수는 DRT의 로컬 인스턴스를 닫습니다.
DrtContinueSearch DrtContinueSearch 함수는 키에 대한 반복 검색을 계속합니다.
DrtCreateDerivedKey DrtCreateDerivedKey 함수는 DRT가 파생 키 보안 공급자를 사용할 때 DrtRegisterKey에서 사용할 수 있는 키를 만듭니다.
DrtCreateDerivedKeySecurityProvider DrtCreateDerivedKeySecurityProvider 함수는 분산 라우팅 테이블에 대한 파생 키 보안 공급자를 만듭니다.
DrtCreateDnsBootstrapResolver DrtCreateDnsBootstrapResolver 함수는 GetAddrInfo 시스템 함수를 사용하여 DRT 메시에 이미 있는 알려진 노드의 호스트 이름을 확인하는 부트스트랩 확인자를 만듭니다.
DrtCreateIpv6UdpTransport DrtCreateIpv6UdpTransport 함수는 IPv6 UDP 프로토콜을 기반으로 전송을 만듭니다.
DrtCreateNullSecurityProvider DrtCreateNullSecurityProvider 함수는 null 보안 공급자를 만듭니다. 이 보안 공급자는 키를 인증하기 위해 노드가 필요하지 않습니다.
DrtCreatePnrpBootstrapResolver DrtCreatePnrpBootstrapResolver.
DrtDeleteDerivedKeySecurityProvider DrtDeleteDerivedKeySecurityProvider 함수는 분산 라우팅 테이블에 대한 파생 키 보안 공급자를 삭제합니다.
DrtDeleteDnsBootstrapResolver DrtDeleteDnsBootstrapResolver 함수는 DNS 부트스트랩 공급자 인스턴스를 삭제합니다.
DrtDeleteIpv6UdpTransport DrtDeleteIpv6UdpTransport 함수는 IPv6 UDP 프로토콜을 기반으로 전송을 삭제합니다.

DrtDeleteNullSecurityProvider DrtDeleteNullSecurityProvider 함수는 분산 라우팅 테이블에 대한 null 보안 공급자를 삭제합니다.
DrtDeletePnrpBootstrapResolver DrtDeletePnrpBootstrapResolver 함수는 PNRP(피어 이름 확인 프로토콜)에 따라 부트스트랩 확인자를 삭제합니다.
DrtEndSearch DrtEndSearch 함수는 DRT에서 키 검색을 취소합니다.
DrtGetEventData DrtGetEventData 함수는 신호 이벤트와 연결된 이벤트 데이터를 검색합니다.
DrtGetEventDataSize DrtGetEventDataSize 함수는 신호 이벤트와 연결된 DRT_EVENT_DATA 구조체의 크기를 반환합니다.
DrtGetInstanceName DrtGetInstanceName 함수는 지정된 DRT 핸들에 해당하는 분산 라우팅 테이블 인스턴스의 전체 이름을 검색합니다.
DrtGetInstanceNameSize DrtGetInstanceNameSize 함수는 분산 라우팅 테이블 인스턴스 이름의 크기를 반환합니다.
DrtGetSearchPath DrtGetSearchPath 함수는 검색 작업 중에 연락한 노드 목록을 반환합니다.
DrtGetSearchPathSize DrtGetSearchPathSize 함수는 검색 작업에 사용된 노드 수를 나타내는 검색 경로의 크기를 반환합니다.
DrtGetSearchResult DrtGetSearchResult 함수입니다.
DrtGetSearchResultSize DrtGetSearchResultSize 함수는 사용 가능한 다음 검색 결과의 크기를 반환합니다.

DrtOpen DrtOpen 함수는 DRT_SETTINGS 구조에 지정된 조건에 대해 로컬 분산 라우팅 테이블 인스턴스를 만듭니다.
DrtRegisterKey DrtRegisterKey 함수는 DRT에 키를 등록합니다.
DrtStartSearch DrtStartSearch 함수는 DRT_SEARCH_INFO 구조에 지정된 조건을 사용하여 DRT에서 키를 검색합니다.
DrtUnregisterKey DrtUnregisterKey 함수는 DRT에서 키를 등록 취소합니다.
DrtUpdateKey DrtUpdateKey 함수는 등록된 키와 연결된 애플리케이션 데이터를 업데이트합니다.
PeerCollabAddContact 피어의 연락처 목록에 연락처를 추가합니다.
PeerCollabAsyncInviteContact 보안 연결을 통해 보낸 사람의 피어 협업 활동에 참가하기 위해 신뢰할 수 있는 피어 연락처에 초대를 보냅니다. 초대 응답의 가용성은 비동기 이벤트를 통해 업데이트됩니다.
PeerCollabAsyncInviteEndpoint 보낸 사람의 피어 공동 작업 활동에 참가하기 위해 지정된 피어 엔드포인트에 초대를 보냅니다. 초대에 대한 응답의 가용성은 비동기 이벤트를 통해 업데이트됩니다.
PeerCollabCancelInvitation 이전에 발신자가 연락처에 보낸 초대를 취소합니다.
PeerCollabCloseHandle 피어 협업 활동 초대에 대한 핸들을 닫습니다.
PeerCollabDeleteContact 호출자와 연결된 로컬 연락처 저장소에서 연락처를 삭제합니다.

PeerCollabDeleteEndpointData 제공된 엔드포인트 데이터와 일치하는 호출 피어 노드에서 피어 엔드포인트 데이터를 삭제합니다.
PeerCollabDeleteObject 호출 엔드포인트에서 피어 개체를 삭제합니다.
PeerCollabEnumApplicationRegistrationInfo 피어 애플리케이션 정보를 검색하는 데 사용되는 열거형 핸들을 가져옵니다.
PeerCollabEnumApplications 특정 피어의 엔드포인트에 등록된 애플리케이션을 포함하는 열거형에 핸들을 반환합니다.
PeerCollabEnumContacts 호출 피어에서 현재 사용할 수 있는 모든 피어 협업 네트워크 연락처를 포함하는 열거형 집합에 대한 핸들을 반환합니다.
PeerCollabEnumEndpoints 특정 피어 연락처와 연결된 엔드포인트를 포함하는 열거형에 대한 핸들을 반환합니다.
PeerCollabEnumObjects 특정 피어의 엔드포인트와 연결된 피어 개체를 포함하는 열거형에 핸들을 반환합니다.
PeerCollabEnumPeopleNearMe 호출 피어의 서브넷에서 현재 사용할 수 있는 모든 피어 협업 네트워크 "내 근처 사람" 엔드포인트가 포함된 열거형 집합에 대한 핸들을 반환합니다.
PeerCollabExportContact 피어 이름과 연결된 연락처 데이터를 문자열 버퍼로 내보냅니다. 버퍼에는 XML 형식의 연락처 데이터가 포함됩니다.
PeerCollabGetAppLaunchInfo 연락처 이름, 피어 엔드포인트 및 초대 요청을 포함하여 피어 애플리케이션 시작 정보를 가져옵니다.
PeerCollabGetApplicationRegistrationInfo 애플리케이션별 등록 정보를 가져옵니다.

PeerCollabGetContact 연락처의 피어 이름이 지정된 경우 피어 연락처에 대한 정보를 가져옵니다.
PeerCollabGetEndpointName 이전에 PeerCollabSetEndpointName을 호출하여 설정한 대로 호출 피어의 현재 엔드포인트 이름을 검색합니다.
PeerCollabGetEventData 피어에서 발생한 피어 협업 이벤트와 연결된 데이터를 가져옵니다.
PeerCollabGetInvitationResponse 피어 공동 작업 활동에 참가하도록 이전에 초대된 피어에서 응답을 가져옵니다.
PeerCollabGetPresenceInfo 특정 연락처와 연결된 엔드포인트에 대한 현재 상태 정보를 검색합니다.
PeerCollabGetSigninOptions 피어의 현재 로그인된 피어 협업 네트워크 현재 상태 옵션을 가져옵니다.
PeerCollabInviteContact 피어 협업 활동에 참가하기 위한 초대를 신뢰할 수 있는 연락처에 보냅니다. 이 호출은 동기적이며, 성공하면 연락처로부터 응답을 받습니다.
PeerCollabInviteEndpoint 보낸 사람의 피어 공동 작업 활동에 참가하기 위해 지정된 피어 엔드포인트에 초대를 보냅니다. 이 호출은 동기적이며, 성공하면 피어 엔드포인트에서 응답을 가져옵니다.
PeerCollabParseContact 연락처 XML 데이터가 포함된 유니코드 문자열 버퍼를 PEER_CONTACT 데이터 구조로 구문 분석합니다.
PeerCollabQueryContactData 제공된 피어 엔드포인트에 대한 연락처 정보를 검색합니다.
PeerCollabRefreshEndpointData 호출 피어 노드를 새 엔드포인트 데이터로 업데이트.

PeerCollabRegisterApplication 피어 협업 작업에서 시작할 수 있도록 로컬 컴퓨터에 애플리케이션을 등록합니다.
PeerCollabRegisterEvent 특정 피어 협업 이벤트에 대한 콜백을 수신하도록 피어 협업 인프라에 애플리케이션을 등록합니다.
PeerCollabSetEndpointName 피어 애플리케이션에서 사용하는 현재 엔드포인트의 이름을 설정합니다.
PeerCollabSetObject 피어 협업 네트워크에 사용되는 피어 데이터 개체를 만들거나 업데이트합니다.
PeerCollabSetPresenceInfo 발신자의 현재 상태 정보를 보고 있는 모든 연락처에 업데이트.
PeerCollabShutdown 피어 협업 인프라를 종료하고 연결된 모든 리소스를 해제합니다.
PeerCollabSignin 피어를 호스트된 인터넷(서버리스 현재 상태) 또는 서브넷("가까운 사람") 피어 협업 네트워크 프레즌스 공급자에 서명합니다.
PeerCollabSignout 특정 유형의 피어 협업 네트워크 현재 상태 공급자에서 피어를 서명합니다.
PeerCollabStartup 피어 협업 인프라를 초기화합니다.
PeerCollabSubscribeEndpointData 사용 가능한 엔드포인트에 대한 구독을 만듭니다.
PeerCollabUnregisterApplication 로컬 컴퓨터에서 피어의 특정 애플리케이션을 등록 취소합니다.
PeerCollabUnregisterEvent 특정 피어 협업 이벤트에 대한 알림에서 애플리케이션을 등록 취소합니다.

PeerCollabUnsubscribeEndpointData PeerCollabSubscribeEndpointData로 만든 엔드포인트에 대한 구독을 제거합니다.
PeerCollabUpdateContact 호출자의 로컬 연락처 저장소에 지정된 피어 연락처와 연결된 정보를 업데이트.
PeerCreatePeerName PeerCreatePeerName 함수는 지정된 피어 ID 및 분류자의 기존 이름을 기반으로 새 이름을 만듭니다. 그러나 PeerCreatePeerName을 호출하면 새 ID가 만들어지지 않습니다.
PeerDistClientAddContentInformation PeerDistClientAddContentInformation 함수는 PeerDistClientOpenContent에서 연 콘텐츠 핸들과 연결된 콘텐츠 정보를 추가합니다.
PeerDistClientAddData PeerDistClientAddData 함수는 로컬 캐시에 콘텐츠를 제공하는 데 사용됩니다.
PeerDistClientBlockRead PeerDistClientBlockRead 함수는 콘텐츠 데이터 블록을 읽습니다.
PeerDistClientCancelAsyncOperation PeerDistClientCancelAsyncOperation 함수는 OVERLAPPED 구조와 연결된 비동기 작업 및 PeerDistClientOpenContent에서 반환하는 콘텐츠 핸들을 취소합니다.
PeerDistClientCloseContent PeerDistClientCloseContent 함수는 PeerDistClientOpenContent에서 연 콘텐츠 핸들을 닫습니다.
PeerDistClientCompleteContentInformation PeerDistClientCompleteContentInformation 함수는 콘텐츠 정보를 추가하는 프로세스를 완료합니다.
PeerDistClientFlushContent PeerDistClientFlushContent 함수를 사용하면 클라이언트가 연결된 PEERDIST_CONTENT_TAG 사용하여 PeerDistClientAddData 함수를 사용하여 로컬 캐시에 추가된 콘텐츠를 제거할 수 있습니다.
PeerDistClientGetInformationByHandle PeerDistClientGetInformationByHandle 함수는 특정 콘텐츠 핸들에 대한 피어 배포 서비스에서 추가 정보를 검색합니다.

PeerDistClientOpenContent PeerDistClientOpenContent 함수가 열리고 PEERDIST_CONTENT_HANDLE 반환합니다. 클라이언트는 콘텐츠 핸들을 사용하여 피어 배포 서비스에서 데이터를 검색합니다.
PeerDistClientStreamRead PeerDistClientStreamRead는 콘텐츠 스트림에서 바이트 시퀀스를 읽습니다.
PeerDistGetOverlappedResult PeerDistGetOverlappedResult 함수는 비동기 작업의 결과를 검색합니다.
PeerDistGetStatus PeerDistGetStatus 함수는 피어 배포 서비스의 현재 상태를 반환합니다.
PeerDistGetStatusEx PeerDistGetStatusEx 함수는 피어 배포 서비스의 현재 상태 및 기능을 반환합니다.
PeerDistRegisterForStatusChangeNotification PeerDistRegisterForStatusChangeNotification 함수는 피어 배포 서비스 상태 변경 알림을 요청합니다.
PeerDistRegisterForStatusChangeNotificationEx PeerDistRegisterForStatusChangeNotificationEx 함수는 피어 배포 서비스 상태 변경 알림을 요청합니다.
PeerDistServerCancelAsyncOperation PeerDistServerCancelAsyncOperation 함수는 콘텐츠 식별자 및 OVERLAPPED 구조와 연결된 비동기 작업을 취소합니다.
PeerDistServerCloseContentInformation PeerDistServerCloseContentInformation 함수는 PeerDistServerOpenContentInformation에서 연 핸들을 닫습니다.
PeerDistServerCloseStreamHandle PeerDistServerCloseStreamHandle 함수는 PeerDistServerPublishStream에서 반환된 핸들을 닫습니다.

PeerDistServerOpenContentInformation PeerDistServerOpenContentInformation 함수는 PEERDIST_CONTENTINFO_HANDLE 엽니다. 클라이언트는 핸들을 사용하여 콘텐츠 정보를 검색합니다.
PeerDistServerOpenContentInformationEx PeerDistServerOpenContentInformationEx 함수는 PEERDIST_CONTENTINFO_HANDLE 엽니다. 클라이언트는 핸들을 사용하여 콘텐츠 정보를 검색합니다.
PeerDistServerPublishAddToStream PeerDistServerPublishAddToStream 함수는 게시 스트림에 데이터를 추가합니다.
PeerDistServerPublishCompleteStream PeerDistServerPublishCompleteStream 함수는 스트림에 데이터를 추가하는 프로세스를 완료합니다.
PeerDistServerPublishStream PeerDistServerPublishStream 함수는 피어 배포 서비스에 게시할 새 스트림을 초기화합니다.
PeerDistServerRetrieveContentInformation PeerDistServerRetrieveContentInformation 함수는 PeerDistServerOpenContentInformation에서 반환된 핸들과 연결된 인코딩된 콘텐츠 정보를 검색합니다.
PeerDistServerUnpublish PeerDistServerUnpublish 함수는 PeerDistServerPublishStream을 통해 만든 게시를 제거합니다.
PeerDistShutdown PeerDistShutdown 함수는 PeerDistStartup 호출에 의해 할당된 리소스를 해제합니다. PeerDistStartup 호출에서 반환되는 각 핸들은 PeerDistShutdown에 대한 일치 호출로 닫혀야 합니다.
PeerDistStartup PeerDistStartup 함수는 다른 모든 피어 배포 API에 전달되어야 하는 새 피어 배포 인스턴스 핸들을 만듭니다.
PeerDistUnregisterForStatusChangeNotification PeerDistUnregisterForStatusChangeNotification 함수는 지정된 핸들과 연결된 세션에 대한 상태 변경 알림을 등록 취소합니다.

PeerEndEnumeration PeerEndEnumeration 함수는 레코드 또는 멤버 열거와 같은 열거형을 해제하고 열거형과 연결된 모든 리소스의 할당을 취소합니다.
PeerEnumGroups PeerEnumGroups 함수는 특정 피어 ID와 연결된 모든 피어 그룹을 열거하는 데 사용되는 피어 열거형 핸들을 만들고 반환합니다.
PeerEnumIdentities PeerEnumIdentities 함수는 특정 사용자에 속하는 모든 피어 ID를 열거하는 데 사용되는 피어 열거형 핸들을 만들고 반환합니다.
PeerFreeData PeerFreeData 함수는 데이터 블록을 할당 취소하고 메모리 풀로 반환합니다. PeerFreeData 함수를 사용하여 피어 ID 관리자, 피어 그룹화 및 피어 협업 API가 반환하는 데이터를 해제합니다.
PeerGetItemCount PeerGetItemCount 함수는 피어 열거형의 항목 수를 반환합니다.
PeerGetNextItem PeerGetNextItem 함수는 피어 열거형에서 특정 개수의 항목을 반환합니다.
PeerGraphAddRecord PeerGraphAddRecord 함수는 피어 그래프에 새 레코드를 추가합니다. 이 함수와 함께 추가된 레코드는 피어 그래프의 각 노드로 전송됩니다.
PeerGraphClose PeerGraphClose 함수는 PeerGraphCreate 또는 PeerGraphOpen 호출로 반환된 피어 그래프 핸들을 무효화하고 지정된 피어 그래프에 대한 모든 네트워크 연결을 닫습니다.
PeerGraphCloseDirectConnection PeerGraphCloseDirectConnection 함수는 지정된 직접 연결을 닫습니다.
PeerGraphConnect PeerGraphConnect 함수는 피어 그래프에서 지정된 노드에 연결을 시도합니다.

PeerGraphCreate PeerGraphCreate 함수는 새 피어 그래프를 만듭니다. 애플리케이션은 피어 그래프에 대한 정보와 피어 그래프에서 사용하는 보안 유형을 지정할 수 있습니다. 피어 그래프에 대한 핸들이 반환되지만 네트워크 연결은 설정되지 않습니다.
PeerGraphDelete PeerGraphDelete 함수는 지정된 피어 그래프와 연결된 데이터를 삭제합니다.
PeerGraphDeleteRecord PeerGraphDeleteRecord 함수는 레코드를 피어 그래프 내에서 삭제된 것으로 표시합니다. 레코드는 로컬 노드에서 함수 호출(예: PeerGraphGetRecord 및 PeerGraphEnumRecords 호출)에서 사용할 수 없습니다.
PeerGraphEndEnumeration PeerGraphEndEnumeration 함수는 열거형 핸들을 해제하고 열거형과 연결된 리소스를 해제합니다.
PeerGraphEnumConnections PeerGraphEnumConnections 함수는 로컬 노드의 연결을 열거하는 데 사용되는 열거형 핸들을 만들고 반환합니다.
PeerGraphEnumNodes PeerGraphEnumNodes 함수는 피어 그래프에서 노드를 열거하는 데 사용되는 열거형 핸들을 만들고 반환합니다.
PeerGraphEnumRecords PeerGraphEnumRecords 함수는 특정 유형의 레코드, 사용자 또는 둘 다의 레코드를 열거하는 데 사용되는 열거형 핸들을 만들고 반환합니다. 열거형은 열거형이 수행될 때 레코드의 스냅샷을 제공합니다.
PeerGraphExportDatabase PeerGraphExportDatabase 함수는 피어 그래프 데이터베이스를 다른 컴퓨터로 이동할 수 있는 파일로 내보냅니다. PeerGraphImportDatabase를 사용하여 피어 그래프 데이터베이스를 다른 컴퓨터로 가져올 수 있습니다.
PeerGraphFreeData PeerGraphFreeData 함수는 여러 피어 그래프 API 함수가 반환하는 리소스를 해제합니다.

PeerGraphGetEventData PeerGraphGetEventData 함수는 피어 이벤트를 검색합니다. 애플리케이션은 반환 값 PEER_S_NO_EVENT_DATA 반환될 때까지 이 함수를 호출합니다. 이는 호출이 성공하지만 검색할 피어 이벤트가 더 이상 없음을 나타냅니다.
PeerGraphGetItemCount PeerGraphGetItemCount 함수는 열거형의 항목 수를 검색합니다.
PeerGraphGetNextItem 다음 함수를 호출하여 만든 열거형의 다음 항목 또는 항목을 가져옵니다.
PeerGraphGetNodeInfo PeerGraphGetNodeInfo 함수는 특정 노드에 대한 정보를 검색합니다.
PeerGraphGetProperties PeerGraphGetProperties 함수는 현재 피어 그래프 속성을 검색합니다.
PeerGraphGetRecord PeerGraphGetRecord 함수는 지정된 레코드 ID에 따라 특정 레코드를 검색합니다. 반환된 레코드는 PeerGraphFreeData를 호출하여 해제해야 합니다.
PeerGraphGetStatus PeerGraphGetStatus 함수는 피어 그래프의 현재 상태를 반환합니다.
PeerGraphImportDatabase PeerGraphImportDatabase 함수는 피어 그래프 데이터베이스의 정보를 포함하는 파일을 가져옵니다. 이 함수는 애플리케이션이 아직 PeerGraphListen 또는 PeerGraphConnect 함수를 호출하지 않은 경우에만 호출할 수 있습니다.
PeerGraphListen PeerGraphListen 함수는 피어 그래프가 들어오는 연결을 수신 대기하기 시작해야 임을 나타냅니다.
PeerGraphOpen PeerGraphOpen 함수는 로컬 노드 또는 원격 노드에서 이전에 만든 피어 그래프를 엽니다. 피어 그래프에 대한 핸들이 반환되지만 네트워크 연결이 설정되지 않았습니다.

PeerGraphOpenDirectConnection PeerGraphOpenDirectConnection 함수를 사용하면 애플리케이션이 피어 그래프의 노드와 직접 연결을 설정할 수 있습니다.
PeerGraphPeerTimeToUniversalTime PeerGraphPeerTimeToUniversalTime 함수는 피어 그래프 유지 관리 참조 시간 값을 피어의 컴퓨터에 표시하기에 적합한 지역화된 시간 값으로 변환합니다.
PeerGraphRegisterEvent PeerGraphRegisterEvent 함수는 피어 그래프 및 이벤트 유형과 관련된 변경 내용에 대한 알림을 받을 피어의 요청을 등록합니다.
PeerGraphSearchRecords PeerGraphSearchRecords 함수는 피어 그래프에서 특정 레코드를 검색합니다.
PeerGraphSendData PeerGraphSendData 함수는 인접 노드 또는 직접 연결된 노드로 데이터를 보냅니다.
PeerGraphSetNodeAttributes PeerGraphSetNodeAttributes 함수는 로컬 노드에 대한 PEER_NODE_INFO 구조의 특성을 설정합니다.
PeerGraphSetPresence PeerGraphSetPresence 함수는 특정 노드에 대한 현재 상태 레코드 게시를 명시적으로 설정하거나 해제합니다.
PeerGraphSetProperties PeerGraphSetProperties 함수는 피어 그래프 속성을 설정합니다.
PeerGraphShutdown PeerGraphShutdown 함수는 PeerGraphStartup 호출에 의해 할당된 모든 리소스를 정리합니다. PeerGraphStartup에 대한 각 호출에 대해 PeerGraphShutdown에 대한 호출이 있어야 합니다.
PeerGraphStartup PeerGraphStartup 함수는 피어 그래프 인프라에 호출 애플리케이션에 필요한 피어 프로토콜 버전을 나타냅니다.

PeerGraphUniversalTimeToPeerTime PeerGraphUniversalTimeToPeerTime 함수는 피어의 컴퓨터에서 공용 피어 그래프 시간 값으로 유니버설 시간 값을 변환합니다.
PeerGraphUnregisterEvent PeerGraphUnregisterEvent 함수는 피어 그래프 및 레코드 형식과 관련된 변경 내용을 애플리케이션에 더 이상 알리지 않도록 요청합니다.
PeerGraphUpdateRecord PeerGraphUpdateRecord 함수는 피어 그래프의 레코드를 업데이트한 다음, 피어 그래프의 각 노드에 레코드를 플러시합니다.
PeerGraphValidateDeferredRecords PeerGraphValidateDeferredRecords 함수는 피어 그래프 인프라에 보안 모듈의 유효성을 검사하기 위해 지연된 레코드를 다시 제출할 때임을 나타냅니다.
PeerGroupAddRecord PeerGroupAddRecord 함수는 피어 그룹에 새 레코드를 추가합니다. 이 레코드는 참여하는 모든 피어에 전파됩니다.
PeerGroupClose PeerGroupClose 함수는 PeerGroupCreate, PeerGroupJoin 또는 PeerGroupOpen 함수에 대한 이전 호출에서 얻은 피어 그룹 핸들을 무효화합니다.
PeerGroupCloseDirectConnection PeerGroupCloseDirectConnection 함수는 두 피어 간의 특정 직접 연결을 닫습니다.
PeerGroupConnect PeerGroupConnect 함수는 피어 그룹에 대한 PNRP 검색을 시작하고 연결을 시도합니다. 이 함수가 성공적으로 호출되면 피어는 피어 그룹의 다른 멤버와 통신할 수 있습니다.
PeerGroupConnectByAddress 알려진 IPv6 주소를 가진 다른 피어가 참여하는 피어 그룹에 연결하려고 시도합니다.
PeerGroupCreate PeerGroupCreate 함수는 새 피어 그룹을 만듭니다.

PeerGroupCreateInvitation PeerGroupCreateInvitation 함수는 지정된 피어에서 그룹에 조인하는 데 사용할 수 있는 XML 문자열을 반환합니다.
PeerGroupCreatePasswordInvitation 지정된 피어에서 일치하는 암호를 사용하여 그룹에 조인하는 데 사용할 수 있는 XML 문자열을 반환합니다.
PeerGroupDelete PeerGroupDelete 함수는 피어 그룹과 연결된 로컬 데이터 및 인증서를 삭제합니다.
PeerGroupDeleteRecord PeerGroupDeleteRecord 함수는 피어 그룹에서 레코드를 삭제합니다. 작성자와 관리 역할의 다른 멤버는 특정 레코드를 삭제할 수 있습니다.
PeerGroupEnumConnections PeerGroupEnumConnections 함수는 피어에서 현재 활성화된 연결의 열거형을 만듭니다.
PeerGroupEnumMembers PeerGroupEnumMembers 함수는 사용 가능한 피어 그룹 멤버 및 연결된 멤버 자격 정보의 열거형을 만듭니다.
PeerGroupEnumRecords PeerGroupEnumRecords 함수는 피어 그룹 레코드의 열거형을 만듭니다.
PeerGroupExportConfig PeerGroupExportConfig 함수는 피어에 대한 그룹 구성을 ID, 그룹 이름 및 ID에 대한 GMC를 포함하는 XML 문자열로 내보냅니다.
PeerGroupExportDatabase PeerGroupExportDatabase 함수는 피어 그룹 데이터베이스를 다른 컴퓨터로 전송하고 PeerGroupImportDatabase 함수를 사용하여 가져올 수 있는 특정 파일로 내보냅니다.
PeerGroupGetEventData PeerGroupGetEventData 함수를 사용하면 애플리케이션이 그룹화 이벤트에서 반환된 데이터를 검색할 수 있습니다.

PeerGroupGetProperties PeerGroupGetProperties 함수는 지정된 그룹의 속성에 대한 정보를 검색합니다.
PeerGroupGetRecord PeerGroupGetRecord 함수는 특정 그룹 레코드를 검색합니다.
PeerGroupGetStatus PeerGroupGetStatus 함수는 그룹의 현재 상태를 검색합니다.
PeerGroupImportConfig PeerGroupImportConfig 함수는 제공된 XML 구성 문자열의 특정 설정에 따라 ID에 대한 피어 그룹 구성을 가져옵니다.
PeerGroupImportDatabase PeerGroupImportDatabase 함수는 로컬 파일에서 피어 그룹 데이터베이스를 가져옵니다.
PeerGroupIssueCredentials PeerGroupIssueCredentials 함수는 GMC를 포함한 자격 증명을 특정 ID에 발급하고, 필요에 따라 초대된 피어가 피어 그룹에 조인하는 데 사용할 수 있는 초대 XML 문자열을 반환합니다.
PeerGroupJoin PeerGroupJoin 함수는 PeerGroupConnect 또는 PeerGroupConnectByAddress를 호출하기 전에 기존 피어 그룹에 가입하라는 초대가 있는 피어를 준비합니다.
PeerGroupOpen PeerGroupOpen 함수는 피어가 만들거나 조인한 피어 그룹을 엽니다. 피어 그룹이 열리면 피어가 이벤트 알림을 등록할 수 있습니다.
PeerGroupOpenDirectConnection PeerGroupOpenDirectConnection 함수는 피어 그룹의 다른 피어와 직접 연결을 설정합니다.
PeerGroupParseInvitation PeerGroupParseInvitation 함수는 특정 초대의 세부 정보가 포함된 PEER_INVITATION_INFO 구조를 반환합니다.
PeerGroupPasswordJoin PeerGroupConnect 또는 PeerGroupConnectByAddress를 호출하기 전에 암호로 보호된 피어 그룹에 조인할 초대 및 올바른 암호를 사용하여 피어를 준비합니다.

PeerGroupPeerTimeToUniversalTime PeerGroupPeerTimeToUniversalTime 함수는 피어 그룹 유지 관리 참조 시간 값을 피어 컴퓨터에 표시하기에 적합한 지역화된 시간 값으로 변환합니다.
PeerGroupRegisterEvent PeerGroupRegisterEvent 함수는 특정 피어 그룹 이벤트에 대한 피어를 등록합니다.
PeerGroupSearchRecords PeerGroupSearchRecords 함수는 로컬 피어 그룹 데이터베이스에서 제공된 조건과 일치하는 레코드를 검색합니다.
PeerGroupSendData PeerGroupSendData 함수는 인접 또는 직접 연결을 통해 멤버에게 데이터를 보냅니다.
PeerGroupSetProperties PeerGroupSetProperties 함수는 현재 피어 그룹 속성을 설정합니다. 이 API 버전 1.0에서는 피어 그룹의 작성자만 이 작업을 수행할 수 있습니다.
PeerGroupShutdown PeerGroupShutdown 함수는 PeerGroupStartup으로 만든 피어 그룹을 닫고 할당된 리소스를 삭제합니다.
PeerGroupStartup PeerGroupStartup 함수는 요청된 버전의 피어 인프라를 사용하여 피어 그룹을 시작합니다.
PeerGroupUniversalTimeToPeerTime PeerGroupUniversalTimeToPeerTime 함수는 피어의 컴퓨터에서 일반 피어 그룹 시간 값으로 로컬 시간 값을 변환합니다.
PeerGroupUnregisterEvent PeerGroupUnregisterEvent 함수는 제공된 이벤트 핸들과 연결된 피어 이벤트의 알림에서 피어를 등록 취소합니다.
PeerGroupUpdateRecord PeerGroupUpdateRecord 함수는 특정 피어 그룹 내의 레코드를 업데이트합니다.

PeerHostNameToPeerName PeerNameToPeerHostName에서 반환된 호스트 이름을 나타내는 피어 이름 문자열로 디코딩합니다.
PeerIdentityCreate PeerIdentityCreate 함수는 새 피어 ID를 만들고 해당 이름을 반환합니다.
PeerIdentityDelete PeerIdentityDelete 함수는 피어 ID를 영구적으로 삭제합니다. 여기에는 지정된 피어 ID와 연결된 모든 인증서, 개인 키 및 모든 그룹 정보를 제거하는 것이 포함됩니다.
PeerIdentityExport PeerIdentityExport 함수를 사용하면 사용자가 하나의 피어 ID를 내보낼 수 있습니다. 그러면 사용자는 피어 ID를 다른 컴퓨터로 전송할 수 있습니다.
PeerIdentityGetCryptKey PeerIdentityGetCryptKey 함수는 CSP(암호화 서비스 공급자)에 대한 핸들을 검색합니다.
PeerIdentityGetDefault PeerIdentityGetDefault 함수는 현재 사용자의 기본 피어 이름 집합을 검색합니다.
PeerIdentityGetFriendlyName PeerIdentityGetFriendlyName 함수는 피어 ID의 이름을 반환합니다.
PeerIdentityGetXML PeerIdentityGetXML 함수는 피어 ID에 대한 설명을 반환합니다. 그러면 타사에 전달되고 피어 ID를 피어 그룹에 초대하는 데 사용할 수 있습니다. 이 정보는 XML 조각으로 반환됩니다.
PeerIdentityImport PeerIdentityImport 함수는 하나의 피어 ID를 가져옵니다. 피어 ID가 컴퓨터에 있으면 PEER_E_ALREADY_EXISTS 반환됩니다.
PeerIdentitySetFriendlyName PeerIdentitySetFriendlyName 함수는 지정된 피어 ID의 이름을 수정합니다. 친숙한 이름은 사람이 읽을 수 있는 이름입니다.

PeerNameToPeerHostName 제공된 피어 이름을 getaddrinfo Windows Sockets 함수에 대한 후속 호출과 함께 사용할 수 있는 형식으로 인코딩합니다.
PeerPnrpEndResolve PeerPnrpStartResolve에 대한 이전 호출로 시작된 비동기 PNRP 확인 작업에 대한 핸들을 닫습니다.
PeerPnrpGetCloudInfo 호출 피어가 참여하는 PNRP(피어 이름 확인 프로토콜) 클라우드에 대한 정보를 검색합니다.
PeerPnrpGetEndpoint 비동기 피어 이름 확인 작업 중에 확인된 피어 엔드포인트 주소를 검색합니다.
PeerPnrpRegister PNRP 클라우드에 피어를 등록하고 등록 업데이트에 사용할 수 있는 핸들을 반환합니다.
PeerPnrpResolve 특정 피어 이름에 대해 등록된 엔드포인트 주소를 가져옵니다.
PeerPnrpShutdown PNRP(피어 이름 확인 프로토콜) 서비스의 실행 중인 인스턴스를 종료하고 연결된 모든 리소스를 해제합니다.
PeerPnrpStartResolve 비동기 피어 이름 확인 작업을 시작합니다.
PeerPnrpStartup 호출 피어에 대한 PNRP(피어 이름 확인 프로토콜) 서비스를 시작합니다.
PeerPnrpUnregister PNRP 클라우드에서 피어를 등록 취소합니다.
PeerPnrpUpdateRegistration 이름에 대한 PNRP 등록 정보를 업데이트.

PFNPEER_FREE_SECURITY_DATA PFNPEER_FREE_SECURITY_DATA 콜백은 피어 그래프 인프라가 PFNPEER_SECURE_RECORD 및 PFNPEER_VALIDATE_RECORD 콜백에서 반환된 데이터를 해제하기 위해 호출하는 함수를 지정합니다.
PFNPEER_SECURE_RECORD PFNPEER_SECURE_RECORD 콜백은 피어 그래프 인프라가 레코드를 보호하기 위해 호출하는 함수를 지정합니다.
PFNPEER_VALIDATE_RECORD PFNPEER_VALIDATE_RECORD 콜백은 피어 그래프 인프라가 레코드의 유효성을 검사하기 위해 호출하는 함수를 지정합니다.

구조체

DRT_ADDRESS DRT_ADDRESS 구조에는 검색에 참여한 DRT 노드에 대한 엔드포인트 정보가 포함됩니다. 이 정보는 연결 문제를 디버깅하는 데 사용하기 위한 것입니다.
DRT_ADDRESS_LIST DRT_ADDRESS_LIST 구조체에는 키를 검색하는 동안 연결한 노드를 나타내는 DRT_ADDRESS 구조 집합이 포함되어 있습니다.
DRT_BOOTSTRAP_PROVIDER DRT_BOOTSTRAP_PROVIDER 구조체는 부트스트랩 공급자가 구현해야 하는 DRT 인터페이스를 정의합니다.
DRT_DATA DRT_DATA 구조에는 데이터 Blob이 포함됩니다. 이 구조는 여러 DRT 함수에서 사용됩니다.
DRT_EVENT_DATA DRT_EVENT_DATA.
DRT_REGISTRATION DRT_REGISTRATION 구조에는 등록을 구성하는 키 및 애플리케이션 데이터가 포함됩니다.

DRT_SEARCH_INFO DRT_SEARCH_INFO 구조체는 DrtStartSearch에서 발급된 검색 쿼리를 나타냅니다.
DRT_SEARCH_RESULT DRT_SEARCH_RESULT.
DRT_SECURITY_PROVIDER DRT_SECURITY_PROVIDER 구조는 보안 공급자가 구현해야 하는 DRT 인터페이스를 정의합니다.
DRT_SETTINGS DRT_SETTINGS 구조에는 로컬 분산 라우팅 테이블에서 사용하는 설정이 포함됩니다.
PEER_ADDRESS PEER_ADDRESS 구조체는 IP 주소에 대한 정보를 지정합니다.
PEER_APP_LAUNCH_INFO PEER_APP_LAUNCH_INFO 구조에는 이전 피어 초대 요청에서 연락처가 제공한 피어 애플리케이션 시작 정보가 포함됩니다.
PEER_APPLICATION PEER_APPLICATION 구조에는 피어 협업 네트워크 내에서 신뢰할 수 있는 연락처와 등록 및 공유할 수 있는 로컬로 설치된 소프트웨어 애플리케이션 또는 구성 요소를 설명하는 데이터가 포함되어 있습니다.
PEER_APPLICATION_REGISTRATION_INFO PEER_APPLICATION_REGISTRATION_INFO 구조에는 로컬 컴퓨터에 등록하기 위한 피어 애플리케이션 정보가 포함됩니다.
PEER_COLLAB_EVENT_REGISTRATION PEER_COLLAB_EVENT_REGISTRATION 구조에는 피어가 특정 피어 협업 네트워크 이벤트에 등록하는 데 사용하는 데이터가 포함됩니다.
PEER_CONNECTION_INFO PEER_CONNECTION_INFO 구조에는 연결에 대한 정보가 포함됩니다. 이 구조는 피어 그래프를 열거하거나 연결을 그룹화할 때 반환됩니다.
PEER_CONTACT PEER_CONTACT 구조에는 특정 연락처에 대한 정보가 포함됩니다.

PEER_CREDENTIAL_INFO PEER_CREDENTIAL_INFO 구조는 피어의 보안 자격 증명을 가져오고 발급하는 데 사용되는 정보를 정의합니다.
PEER_DATA PEER_DATA 구조에는 이진 데이터가 포함됩니다.
PEER_ENDPOINT PEER_ENDPOINT 구조체에는 피어 엔드포인트의 주소 및 식별 이름이 포함됩니다.
PEER_EVENT_APPLICATION_CHANGED_DATA PEER_EVENT_APPLICATION_CHANGED_DATA 구조에는 피어 협업 네트워크에 참여하는 피어에서 PEER_EVENT_ENDPOINT_APPLICATION_CHANGED 또는 PEER_EVENT_MY_APPLICATION_CHANGED 이벤트가 발생할 때 반환되는 정보가 포함됩니다.
PEER_EVENT_CONNECTION_CHANGE_DATA 다음 피어 이벤트 중 하나가 트리거되는 경우 PEER_EVENT_CONNECTION_CHANGE_DATA 구조를 가리킵니다.
PEER_EVENT_ENDPOINT_CHANGED_DATA PEER_EVENT_ENDPOINT_CHANGED_DATA 구조에는 피어 협업 네트워크에 참여하는 피어에서 PEER_EVENT_ENDPOINT_CHANGED 또는 PEER_EVENT_MY_ENDPOINT_CHANGED 이벤트가 발생할 때 반환되는 정보가 포함됩니다.
PEER_EVENT_INCOMING_DATA 다음 피어 이벤트 중 하나가 트리거되는 경우 PEER_EVENT_INCOMING_DATA 구조를 가리킵니다.
PEER_EVENT_MEMBER_CHANGE_DATA PEER_EVENT_MEMBER_CHANGE_DATA 구조에는 피어 그룹 멤버의 상태 변경을 설명하는 데이터가 포함되어 있습니다.
PEER_EVENT_NODE_CHANGE_DATA PEER_GRAPH_EVENT_NODE_CHANGE 이벤트가 트리거되는 경우 PEER_EVENT_NODE_CHANGE_DATA 구조에는 데이터에 대한 포인터가 포함됩니다.
PEER_EVENT_OBJECT_CHANGED_DATA PEER_EVENT_OBJECT_CHANGED_DATA 구조에는 피어 협업 네트워크에 참여하는 피어에서 PEER_EVENT_ENDPOINT_OBJECT_CHANGED 또는 PEER_EVENT_MY_OBJECT_CHANGED 이벤트가 발생할 때 반환되는 정보가 포함됩니다.

PEER_EVENT_PEOPLE_NEAR_ME_CHANGED_DATA PEER_EVENT_PEOPLE_NEAR_ME_CHANGED_DATA 구조에는 서브넷별 피어 협업 네트워크에 참여하는 피어에서 PEER_EVENT_PEOPLE_NEAR_ME_CHANGED 이벤트가 발생할 때 반환되는 정보가 포함됩니다.
PEER_EVENT_PRESENCE_CHANGED_DATA PEER_EVENT_PRESENCE_CHANGED_DATA 구조에는 피어 협업 네트워크에 참여하는 피어에서 PEER_EVENT_ENDPOINT_PRESENCE_CHANGED 또는 PEER_EVENT_MY_PRESENCE_CHANGED 이벤트가 발생할 때 반환되는 정보가 포함됩니다.
PEER_EVENT_RECORD_CHANGE_DATA 다음 피어 이벤트 중 하나가 트리거되는 경우 PEER_EVENT_RECORD_CHANGE_DATA 구조를 가리킵니다.
PEER_EVENT_REQUEST_STATUS_CHANGED_DATA PEER_EVENT_REQUEST_STATUS_CHANGED_DATA 구조에는 피어 협업 네트워크에 참여하는 피어에서 PEER_EVENT_REQUEST_STATUS_CHANGED 이벤트가 발생할 때 반환되는 정보가 포함됩니다.
PEER_EVENT_SYNCHRONIZED_DATA PEER_EVENT_SYNCHRONIZED_DATA PEER_GRAPH_EVENT_RECORD_CHANGE 또는 PEER_GROUP_EVENT_RECORD_CHANGE 이벤트가 트리거되는 경우 PEER_GRAPH_EVENT_DATA 구조체의 공용 구조체에 의해 지적됩니다.
PEER_EVENT_WATCHLIST_CHANGED_DATA PEER_EVENT_WATCHLIST_CHANGED_DATA 구조에는 피어 협업 네트워크에 참여하는 피어에서 PEER_EVENT_WATCHLIST_CHANGED 이벤트가 발생할 때 반환되는 정보가 포함됩니다.
PEER_GRAPH_EVENT_DATA PEER_GRAPH_EVENT_DATA 구조에는 피어 이벤트와 연결된 데이터가 포함됩니다.
PEER_GRAPH_EVENT_REGISTRATION PEER_GRAPH_EVENT_REGISTRATION 구조는 피어 이벤트 알림을 등록하는 동안 사용됩니다. 등록하는 동안 애플리케이션에 알림이 필요한 피어 이벤트를 지정합니다.
PEER_GRAPH_PROPERTIES PEER_GRAPH_PROPERTIES 구조에는 피어 그래프, ID, 범위 및 기타 정보의 정책에 대한 데이터가 포함됩니다.

PEER_GROUP_EVENT_REGISTRATION PEER_GROUP_EVENT_REGISTRATION 구조체는 멤버가 등록할 수 있는 특정 피어 그룹 이벤트를 정의합니다.
PEER_GROUP_PROPERTIES PEER_GROUP_PROPERTIES 구조에는 피어 그룹의 멤버 자격 정책에 대한 데이터가 포함됩니다.
PEER_INVITATION PEER_INVITATION 구조에는 피어 협업 활동을 시작하거나 조인하는 요청이 포함됩니다.
PEER_INVITATION_INFO PEER_INVITATION_INFO 구조는 피어 그룹에 가입하기 위한 초대에 대한 정보를 정의합니다.
PEER_INVITATION_RESPONSE PEER_INVITATION_RESPONSE 구조에는 피어 협업 활동에 참가하기 위한 초대에 대한 응답이 포함됩니다.
PEER_MEMBER PEER_MEMBER 구조에는 피어 그룹의 멤버를 설명하는 정보가 포함되어 있습니다.
PEER_NAME_PAIR PEER_NAME_PAIR 구조체에는 PeerGetNextItem에 대한 호출 결과가 포함됩니다.
PEER_NODE_INFO PEER_NODE_INFO 구조에는 피어 그래프의 특정 노드와 관련된 정보가 포함됩니다.
PEER_OBJECT PEER_OBJECT 구조에는 피어 협업 네트워크 내에서 신뢰할 수 있는 연락처와 공유할 수 있는 애플리케이션별 런타임 정보가 포함됩니다.
PEER_PEOPLE_NEAR_ME 동일한 논리 또는 가상 서브넷의 피어에 대한 정보를 포함합니다.
PEER_PNRP_CLOUD_INFO PNRP(피어 이름 확인 프로토콜) 클라우드에 대한 정보를 포함합니다.

PEER_PNRP_ENDPOINT_INFO 피어 엔드포인트와 연결된 IP 주소 및 데이터를 포함합니다.
PEER_PNRP_REGISTRATION_INFO PNRP 클라우드에 등록할 때 피어 ID에서 제공하는 정보를 포함합니다.
PEER_PRESENCE_INFO PEER_PRESENCE_INFO 구조에는 특정 피어 프레즌스 정보가 포함됩니다.
PEER_RECORD PEER_RECORD 구조체에는 애플리케이션에서 사용하는 레코드 개체가 포함됩니다.
PEER_SECURITY_INTERFACE PEER_SECURITY_INTERFACE 구조는 피어 그래프 API를 호출하여 레코드의 유효성을 검사하고, 보호하고, 사용 가능한 레코드를 호출하는 보안 인터페이스를 지정합니다.
PEER_VERSION_DATA PEER_VERSION_DATA 구조에는 피어 그래프 및 그룹화 API에 대한 버전 정보가 포함됩니다.
PEERDIST_CLIENT_BASIC_INFO PEERDIST_CLIENT_BASIC_INFO 구조는 동일한 콘텐츠를 동시에 다운로드하는 클라이언트가 많은지 여부를 나타냅니다.
PEERDIST_CONTENT_TAG PEERDIST_CONTENT_TAG 구조체는 PeerDistClientOpenContent API에 대한 입력으로 클라이언트 제공 콘텐츠 태그를 포함합니다.
PEERDIST_PUBLICATION_OPTIONS PEERDIST_PUBLICATION_OPTIONS 구조에는 API 버전 정보 및 가능한 옵션 플래그를 포함한 게시 옵션이 포함됩니다.
PEERDIST_RETRIEVAL_OPTIONS PEER_RETRIEVAL_OPTIONS 구조에는 검색할 콘텐츠 정보의 버전이 포함되어 있습니다.
PEERDIST_STATUS_INFO PEERDIST_STATUS_INFO 구조에는 로컬 컴퓨터의 BranchCache 서비스의 현재 상태 및 기능에 대한 정보가 포함되어 있습니다.

PNRP_CLOUD_ID PNRP_CLOUD_ID 구조에는 네트워크 클라우드를 정의하는 값이 포함됩니다.
PNRPCLOUDINFO PNRPCLOUDINFO 구조체는 WSAQUERYSET 구조체의 lpBlob 멤버를 가리킨다.
PNRPINFO_V1 PNRPINFO_V1 구조체는 WSAQUERYSET 구조체의 lpBlob 멤버를 가리켰습니다. (PNRPINFO_V1)
PNRPINFO_V2 PNRPINFO_V1 구조체는 WSAQUERYSET 구조체의 lpBlob 멤버를 가리켰습니다. P

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

성능 카운터

아티클2023. 03. 11.

성능 카운터 기술 개요입니다.

성능 카운터를 개발하려면 다음 헤더가 필요합니다.

- pdh.h
- perflib.h
- winperf.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 성능 카운터

열거형

PerfCounterDataType PerfQueryCounterData 함수가 함수가 출력으로 생성하는 PERF_DATA_HEADER 블록의 일부로 포함하는 PERF_COUNTER_HEADER 블록의 콘텐츠 형식을 나타냅니다.
PerfRegInfoType PerfQueryCounterSetRegistrationInfo 함수를 호출하여 성능 카운터 집합에 대해 요청할 수 있는 정보 유형을 나타냅니다.

함수

CounterPathCallBack 애플리케이션은 CounterPathCallBack 함수를 구현하여 찾아보기 대화 상자에서 반환된 카운터 경로 문자열을 처리합니다.
LoadPerfCounterTextStringsA 지정된 초기화 파일에 정의된 성능 개체 및 카운터를 컴퓨터에 로드합니다. (ANSI)

LoadPerfCounterTextStringsW 지정된 초기화 파일에 정의된 성능 개체 및 카운터를 컴퓨터에 로드합니다. (유니코드)
PdhAddCounterA 지정된 카운터를 쿼리에 추가합니다. (ANSI)
PdhAddCounterW 지정된 카운터를 쿼리에 추가합니다. (유니코드)
PdhAddEnglishCounterA 지정된 언어 중립적 카운터를 쿼리에 추가합니다. (ANSI)
PdhAddEnglishCounterW 지정된 언어 중립적 카운터를 쿼리에 추가합니다. (유니코드)
PdhBindInputDataSourceA 로그 데이터를 읽기 위해 하나 이상의 이진 로그 파일을 함께 바인딩합니다. (ANSI)
PdhBindInputDataSourceW 로그 데이터를 읽기 위해 하나 이상의 이진 로그 파일을 함께 바인딩합니다. (유니코드)
PdhBrowseCountersA 사용자가 쿼리에 추가할 카운터를 하나 이상 선택하는 데 사용할 수 있는 카운터 찾아보기 대화 상자를 표시합니다. 데이터 원본에 핸들을 사용하려면 PdhBrowseCountersH 함수를 사용합니다. (ANSI)
PdhBrowseCountersHA 사용자가 쿼리에 추가할 카운터를 하나 이상 선택하는 데 사용할 수 있는 카운터 찾아보기 대화 상자를 표시합니다. 이 함수는 데이터 원본에 대한 핸들 사용을 지원한다는 점을 제외하고 PdhBrowseCounters 함수와 동일합니다. (ANSI)
PdhBrowseCountersHW 사용자가 쿼리에 추가할 카운터를 하나 이상 선택하는 데 사용할 수 있는 카운터 찾아보기 대화 상자를 표시합니다. 이 함수는 데이터 원본에 대한 핸들 사용을 지원한다는 점을 제외하고 PdhBrowseCounters 함수와 동일합니다. (유니코드)

PdhBrowseCountersW 사용자가 쿼리에 추가할 카운터를 하나 이상 선택하는 데 사용할 수 있는 카운터 찾아보기 대화 상자를 표시합니다. 데이터 원본에 핸들을 사용하려면 PdhBrowseCountersH 함수를 사용합니다. (유니코드)
PdhCalculateCounterFromRawValue 두 원시 카운터 값의 표시 가능한 값을 계산합니다.
PdhCloseLog 지정된 로그 파일을 닫습니다.
PdhCloseQuery 지정된 쿼리에 포함된 모든 카운터를 닫고, 쿼리와 관련된 모든 핸들을 닫고, 쿼리와 연결된 모든 메모리를 해제합니다.
PdhCollectQueryData 지정된 쿼리의 모든 카운터에 대한 현재 원시 데이터 값을 수집하고 각 카운터의 상태 코드를 업데이트합니다. (PdhCollectQueryData)
PdhCollectQueryDataEx 별도 스레드를 사용하여 지정된 쿼리의 모든 카운터에 대한 현재 원시 데이터 값을 수집합니다. 그런 다음, 함수는 애플리케이션 정의 이벤트에 신호를 전송하고 반환하기 전에 지정된 시간 간격을 기다립니다.
PdhCollectQueryDataWithTime 지정된 쿼리의 모든 카운터에 대한 현재 원시 데이터 값을 수집하고 각 카운터의 상태 코드를 업데이트합니다. (PdhCollectQueryDataWithTime)
PdhComputeCounterStatistics 원시 값 배열의 카운터에 대한 통계를 계산합니다.
PdhConnectMachineA 지정된 컴퓨터에 연결합니다. (ANSI)
PdhConnectMachineW 지정된 컴퓨터에 연결합니다. (유니코드)

PdhEnumLogSetNamesA DSN 내에서 로그 집합의 이름을 열거합니다. (ANSI)
PdhEnumLogSetNamesW DSN 내에서 로그 집합의 이름을 열거합니다. (유니코드)
PdhEnumMachinesA 로그 파일의 카운터와 연결된 컴퓨터 이름 목록을 반환합니다. (PdhEnumMachinesA)
PdhEnumMachinesHA 로그 파일의 카운터와 연결된 컴퓨터 이름 목록을 반환합니다. (PdhEnumMachinesHA)
PdhEnumMachinesHW 로그 파일의 카운터와 연결된 컴퓨터 이름 목록을 반환합니다. (PdhEnumMachinesHW)
PdhEnumMachinesW 로그 파일의 카운터와 연결된 컴퓨터 이름 목록을 반환합니다. (PdhEnumMachinesW)
PdhEnumObjectItemsA 지정된 컴퓨터 또는 지정된 로그 파일에 있는 지정된 개체의 카운터 및 인스턴스 이름을 반환합니다. 데이터 원본에 대한 핸들을 사용하려면 PdhEnumObjectItemsH 함수를 사용합니다. (ANSI)
PdhEnumObjectItemsHA 지정된 컴퓨터 또는 지정된 로그 파일에 있는 지정된 개체의 카운터 및 인스턴스 이름을 반환합니다. 이 함수는 데이터 원본에 대한 핸들 사용을 지원한다는 점을 제외하고 PdhEnumObjectItems 함수와 동일합니다. (ANSI)
PdhEnumObjectItemsHW 지정된 컴퓨터 또는 지정된 로그 파일에 있는 지정된 개체의 카운터 및 인스턴스 이름을 반환합니다. 이 함수는 데이터 원본에 대한 핸들 사용을 지원한다는 점을 제외하고 PdhEnumObjectItems 함수와 동일합니다. (유니코드)
PdhEnumObjectItemsW 지정된 컴퓨터 또는 지정된 로그 파일에 있는 지정된 개체의 카운터 및 인스턴스 이름을 반환합니다. 데이터 원본에 대한 핸들을 사용하려면 PdhEnumObjectItemsH 함수를 사용합니다. (유니코드)

PdhEnumObjectsA 지정된 컴퓨터 또는 지정된 로그 파일에서 사용할 수 있는 개체 목록을 반환합니다. 데이터 원본에 대한 핸들을 사용하려면 PdhEnumObjectsH 함수를 사용합니다. (ANSI)
PdhEnumObjectsHA 지정된 컴퓨터 또는 지정된 로그 파일에서 사용할 수 있는 개체 목록을 반환합니다. 이 함수는 데이터 원본에 대한 핸들 사용을 지원한다는 점을 제외하고 PdhEnumObjects와 동일합니다. (ANSI)
PdhEnumObjectsHW 지정된 컴퓨터 또는 지정된 로그 파일에서 사용할 수 있는 개체 목록을 반환합니다. 이 함수는 데이터 원본에 대한 핸들 사용을 지원한다는 점을 제외하고 PdhEnumObjects와 동일합니다. (유니코드)
PdhEnumObjectsW 지정된 컴퓨터 또는 지정된 로그 파일에서 사용할 수 있는 개체 목록을 반환합니다. 데이터 원본에 대한 핸들을 사용하려면 PdhEnumObjectsH 함수를 사용합니다. (유니코드)
PdhExpandCounterPathA 카운터 경로의 와일드카드 문자열과 일치하는 카운터 및 카운터 인스턴스에 대해 지정된 컴퓨터(또는 지정된 컴퓨터가 없는 경우 로컬 컴퓨터)를 검사합니다. (ANSI)
PdhExpandCounterPathW 카운터 경로의 와일드카드 문자열과 일치하는 카운터 및 카운터 인스턴스에 대해 지정된 컴퓨터(또는 지정된 컴퓨터가 없는 경우 로컬 컴퓨터)를 검사합니다. (유니코드)
PdhExpandWildCardPathA 지정된 컴퓨터 또는 로그 파일을 검사하고 와일드카드 문자를 포함하는 지정된 카운터 경로와 일치하는 카운터 경로를 반환합니다. 데이터 원본에 대한 핸들을 사용하려면 PdhExpandWildCardPathH 함수를 사용합니다. (ANSI)
PdhExpandWildCardPathHA 지정된 컴퓨터 또는 로그 파일을 검사하고 와일드카드 문자를 포함하는 지정된 카운터 경로와 일치하는 카운터 경로를 반환합니다. 이 함수는 데이터 원본에 대한 핸들 사용을 지원한다는 점을 제외하고 PdhExpandWildCardPath 함수와 동일합니다. (ANSI)
PdhExpandWildCardPathHW 지정된 컴퓨터 또는 로그 파일을 검사하고 와일드카드 문자를 포함하는 지정된 카운터 경로와 일치하는 카운터 경로를 반환합니다. 이 함수는 데이터 원본에 대한 핸들 사용을 지원한다는 점을 제외하고 PdhExpandWildCardPath 함수와 동일합니다. (유니코드)

PdhExpandWildCardPathW 지정된 컴퓨터 또는 로그 파일을 검사하고 와일드카드 문자를 포함하는 지정된 카운터 경로와 일치하는 카운터 경로를 반환합니다. 데이터 원본에 대한 핸들을 사용하려면 PdhExpandWildCardPathH 함수를 사용합니다. (유니코드)
PdhFormatFromRawValue 지정된 원시 카운터 값에 대해 표시 가능한 값을 계산합니다.
PdhGetCounterInfoA 데이터 크기, 카운터 형식, 경로 및 사용자가 제공한 데이터 값과 같은 카운터에 대한 정보를 검색합니다. (ANSI)
PdhGetCounterInfoW 데이터 크기, 카운터 형식, 경로 및 사용자가 제공한 데이터 값과 같은 카운터에 대한 정보를 검색합니다. (유니코드)
PdhGetCounterTimeBase 지정된 카운터의 시간 기반을 반환합니다.
PdhGetDataSourceTimeRangeA 시간 범위, 항목 수 및 해당하는 경우 지정된 입력 원본의 성능 데이터를 포함하는 버퍼의 크기를 결정합니다. 데이터 원본에 대한 핸들을 사용하려면 PdhGetDataSourceTimeRangeH 함수를 사용합니다. (ANSI)
PdhGetDataSourceTimeRangeH 시간 범위, 항목 수 및 해당하는 경우 지정된 입력 원본의 성능 데이터를 포함하는 버퍼의 크기를 결정합니다. 이 함수는 데이터 원본에 대한 핸들 사용을 지원한다는 점을 제외하고 PdhGetDataSourceTimeRange 함수와 동일합니다.
PdhGetDataSourceTimeRangeW 시간 범위, 항목 수 및 해당하는 경우 지정된 입력 원본의 성능 데이터를 포함하는 버퍼의 크기를 결정합니다. 데이터 원본에 대한 핸들을 사용하려면 PdhGetDataSourceTimeRangeH 함수를 사용합니다. (유니코드)
PdhGetDefaultPerfCounterA 지정된 개체에 대한 기본 카운터의 이름을 검색합니다. 이 이름은 카운터 찾아보기 대화 상자에서 초기 카운터 선택을 설정하는 데 사용할 수 있습니다. 데이터 원본에 대한 핸들을 사용하려면 PdhGetDefaultPerfCounterH 함수를 사용합니다. (ANSI)

PdhGetDefaultPerfCounterHA 지정된 개체에 대한 기본 카운터의 이름을 검색합니다. (ANSI)
PdhGetDefaultPerfCounterHW 지정된 개체에 대한 기본 카운터의 이름을 검색합니다. (유니코드)
PdhGetDefaultPerfCounterW 지정된 개체에 대한 기본 카운터의 이름을 검색합니다. 이 이름은 카운터 찾아보기 대화 상자에서 초기 카운터 선택을 설정하는 데 사용할 수 있습니다. 데이터 원본에 대한 핸들을 사용하려면 PdhGetDefaultPerfCounterH 함수를 사용합니다. (유니코드)
PdhGetDefaultPerfObjectA 기본 개체의 이름을 검색합니다. 이 이름은 카운터 찾아보기 대화 상자에서 초기 개체 선택을 설정하는 데 사용할 수 있습니다. 데이터 원본에 대한 핸들을 사용하려면 PdhGetDefaultPerfObjectH 함수를 사용합니다. (ANSI)
PdhGetDefaultPerfObjectHA 기본 개체의 이름을 검색합니다. (ANSI)
PdhGetDefaultPerfObjectHW 기본 개체의 이름을 검색합니다. (유니코드)
PdhGetDefaultPerfObjectW 기본 개체의 이름을 검색합니다. 이 이름은 카운터 찾아보기 대화 상자에서 초기 개체 선택을 설정하는 데 사용할 수 있습니다. 데이터 원본에 대한 핸들을 사용하려면 PdhGetDefaultPerfObjectH 함수를 사용합니다. (유니코드)
PdhGetDllVersion 현재 설치된 Pdh.dll 파일의 버전을 반환합니다.
PdhGetFormattedCounterArrayA 서식이 지정된 카운터 값의 배열을 반환합니다. 인스턴스 이름에 대한 와일드카드 문자가 포함된 카운터의 카운터 값에 서식을 지정하려면 이 함수를 사용합니다. (ANSI)
PdhGetFormattedCounterArrayW 서식이 지정된 카운터 값의 배열을 반환합니다. 인스턴스 이름에 대한 와일드카드 문자가 포함된 카운터의 카운터 값에 서식을 지정하려면 이 함수를 사용합니다. (유니코드)

PdhGetFormattedCounterValue 지정된 카운터에 대해 표시 가능한 값을 계산합니다.
PdhGetLogFileSize 지정된 로그 파일의 크기를 반환합니다.
PdhGetRawCounterArrayA 지정된 카운터에서 원시 값 배열을 반환합니다. 인스턴스 이름에 대한 와일드카드 문자가 포함된 카운터의 원시 카운터 값을 검색하려는 경우 이 함수를 사용합니다. (ANSI)
PdhGetRawCounterArrayW 지정된 카운터에서 원시 값 배열을 반환합니다. 인스턴스 이름에 대한 와일드카드 문자가 포함된 카운터의 원시 카운터 값을 검색하려는 경우 이 함수를 사용합니다. (유니코드)
PdhGetRawCounterValue 카운터의 현재 원시 값을 반환합니다.
PdhIsRealTimeQuery 지정된 쿼리가 실시간 쿼리인지 확인합니다.
PdhLookupPerfIndexByNameA 지정된 카운터 이름에 해당하는 카운터 인덱스 값을 반환합니다. (ANSI)
PdhLookupPerfIndexByNameW 지정된 카운터 이름에 해당하는 카운터 인덱스 값을 반환합니다. (유니코드)
PdhLookupPerfNameByIndexA 지정된 인덱에 해당하는 성능 개체 이름 또는 카운터 이름을 반환합니다. (ANSI)
PdhLookupPerfNameByIndexW 지정된 인덱에 해당하는 성능 개체 이름 또는 카운터 이름을 반환합니다. (유니코드)
PdhMakeCounterPathA PDH_COUNTER_PATH_ELEMENTS 구조체에 지정된 멤버를 사용하여 전체 카운터 경로를 만듭니다. (ANSI)

PdhMakeCounterPathW PDH_COUNTER_PATH_ELEMENTS 구조체에 지정된 멤버를 사용하여 전체 카운터 경로를 만듭니다. (유니코드)
PdhOpenLogA 읽기 또는 쓰기를 위해 지정된 로그 파일을 엽니다. (ANSI)
PdhOpenLogW 읽기 또는 쓰기를 위해 지정된 로그 파일을 엽니다. (유니코드)
PdhOpenQueryA 성능 데이터 컬렉션을 관리하는 데 사용되는 새 쿼리를 만듭니다. 데이터 원본에 대한 핸들을 사용하려면 PdhOpenQueryH 함수를 사용합니다. (ANSI)
PdhOpenQueryH 성능 데이터 컬렉션을 관리하는 데 사용되는 새 쿼리를 만듭니다. 이 함수는 데이터 원본에 대한 핸들 사용을 지원한다는 점을 제외하고 PdhOpenQuery 함수와 동일합니다.
PdhOpenQueryW 성능 데이터 컬렉션을 관리하는 데 사용되는 새 쿼리를 만듭니다. 데이터 원본에 대한 핸들을 사용하려면 PdhOpenQueryH 함수를 사용합니다. (유니코드)
PdhParseCounterPathA 카운터 경로의 요소를 구문 분석하고 결과를 PDH_COUNTER_PATH_ELEMENTS 구조에 저장합니다. (ANSI)
PdhParseCounterPathW 카운터 경로의 요소를 구문 분석하고 결과를 PDH_COUNTER_PATH_ELEMENTS 구조에 저장합니다. (유니코드)
PdhParseInstanceNameA 인스턴스 문자열의 요소를 구문 분석합니다. (ANSI)
PdhParseInstanceNameW 인스턴스 문자열의 요소를 구문 분석합니다. (유니코드)
PdhReadRawLogRecord 지정된 이진 추적 로그 파일의 정보를 읽습니다.

PdhRemoveCounter 쿼리에서 카운터를 제거합니다.
PdhSelectDataSourceA 성능 데이터의 원본을 지정하라는 메시지를 표시하는 대화 상자 창이 표시됩니다. (ANSI)
PdhSelectDataSourceW 성능 데이터의 원본을 지정하라는 메시지를 표시하는 대화 상자 창이 표시됩니다. (유니코드)
PdhSetCounterScaleFactor 서식이 지정된 카운터 값을 요청할 때 지정된 카운터의 계산 값에 적용되는 배율 인수를 설정합니다. PDH_FMT_NOSCALE 플래그가 설정된 경우 이 배율 인수는 무시됩니다.
PdhSetDefaultRealTimeDataSource 실시간 데이터의 원본을 지정합니다.
PdhSetQueryTimeRange 로그 파일에서 읽을 수 있는 샘플을 지정된 시간 범위 내의 샘플로 제한합니다.
PdhUpdateLogA 현재 쿼리에 대한 카운터 데이터를 수집하고 로그 파일에 데이터를 씁니다. (ANSI)
PdhUpdateLogFileCatalog 로그 파일 카탈로그의 정보를 로그 파일의 성능 데이터와 동기화합니다.
PdhUpdateLogW 현재 쿼리에 대한 카운터 데이터를 수집하고 로그 파일에 데이터를 씁니다. (유니코드)
PdhValidatePathA 카운터 경로에 지정된 컴퓨터에 카운터가 있는지 확인합니다. (ANSI)
PdhValidatePathExA 지정된 카운터가 컴퓨터 또는 로그 파일에 있는지 확인합니다. (ANSI)
PdhValidatePathExW 지정된 카운터가 컴퓨터 또는 로그 파일에 있는지 확인합니다. (유니코드)

PdhValidatePathW 카운터 경로에 지정된 컴퓨터에 카운터가 있는지 확인합니다. (유니코드)
PERF_MEM_ALLOC 공급자는 PERFLIB에 대한 사용자 지정 메모리 관리를 제공하기 위해 이 함수를 구현합니다. A
PERF_MEM_FREE 공급자는 PERFLIB에 대한 사용자 지정 메모리 관리를 제공하기 위해 이 함수를 구현합니다. F
PerfAddCounters 지정된 쿼리에 성능 카운터 사양을 추가합니다.
PerfCloseQueryHandle PerfOpenQueryHandle을 호출하여 연 쿼리 핸들을 닫습니다.
PerfCreateInstance 지정된 카운터 집합의 인스턴스를 만듭니다.
PerfDecrementULongCounterValue 값이 부호 없는 4 바이트 정수인 카운터의 값을 감소합니다. 공급자는 이 함수를 사용합니다.
PerfDecrementULongLongCounterValue 값이 부호 없는 8 바이트 정수인 카운터의 값을 감소합니다. 공급자는 이 함수를 사용합니다.
PerfDeleteCounters 지정된 쿼리에서 지정된 성능 카운터 사양을 제거합니다.
PerfDeleteInstance PerfCreateInstance 함수에서 만든 카운터 집합의 인스턴스를 삭제합니다.
PerfEnumerateCounterSet 지정된 시스템에 등록된 카운터 집합의 카운터 집합 식별자를 가져옵니다. 카운터 집합 식별자는 GUID(Globally Unique Identifier)입니다.
PerfEnumerateCounterSetInstances 지정된 시스템에 설정된 카운터의 활성 인스턴스 이름 및 식별자를 가져옵니다.

PerfIncrementULongCounterValue 값이 부호 없는 4 바이트 정수인 카운터의 값을 증분합니다. 공급자는 이 함수를 사용합니다.
PerfIncrementULongLongCounterValue 값이 부호 없는 8 바이트 정수인 카운터의 값을 증분합니다. 공급자는 이 함수를 사용합니다.
PERFLIBREQUEST 공급자는 소비자가 쿼리에서 카운터 추가 또는 제거와 같은 특정 작업을 수행할 때 알림을 받도록 이 함수를 구현할 수 있습니다.
PerfOpenQueryHandle 지정된 시스템에서 쿼리를 참조하는 핸들을 만듭니다. 쿼리는 카운터 사양의 목록입니다.
PerfQueryCounterData 지정된 쿼리의 카운터 사양과 일치하는 성능 카운터의 값을 가져옵니다.
PerfQueryCounterInfo 지정된 쿼리의 카운터 사양을 가져옵니다.
PerfQueryCounterSetRegistrationInfo 지정된 시스템의 카운터 집합에 대한 정보를 가져옵니다.
PerfQueryInstance 지정된 카운터 집합 인스턴스에 대한 포인터를 검색합니다. 공급자는 이 함수를 사용합니다.
PerfSetCounterRefValue 값이 실제 데이터에 대한 포인터인 카운터의 값을 업데이트. 공급자는 이 함수를 사용합니다.
PerfSetCounterSetInfo 특정 카운터 집합의 레이아웃을 지정합니다.
PerfSetULongCounterValue 값이 4 바이트 부호 없는 정수인 카운터의 값을 업데이트. 공급자는 이 함수를 사용합니다.
PerfSetULongLongCounterValue 값이 부호 없는 8 바이트 정수인 카운터의 값을 업데이트. 공급자는 이 함수를 사용합니다.

PerfStartProvider 공급자를 등록합니다. (PerfStartProvider)
PerfStartProviderEx 공급자를 등록합니다. (PerfStartProviderEx)
PerfStopProvider 등록된 공급자 목록에서 공급자의 등록을 제거하고 공급자와 연결된 모든 리소스를 해제합니다.
PM_CLOSE_PROC 성능 DLL에 필요한 정리를 수행합니다.
PM_COLLECT_PROC 성능 데이터를 수집하고 소비자에게 반환합니다.
UnloadPerfCounterTextStringsA 지정된 애플리케이션에 대해 컴퓨터에서 성능 개체 및 카운터를 언로드합니다. (ANSI)
UnloadPerfCounterTextStringsW 지정된 애플리케이션에 대해 컴퓨터에서 성능 개체 및 카운터를 언로드합니다. (유니코드)

구조체

PDH_BROWSE_DLG_CONFIG_A PDH_BROWSE_DLG_CONFIG 구조는 PdhBrowseCounters 함수에서 성능 카운터 찾아보기 대화 상자를 구성하는 데 사용됩니다. (ANSI)
PDH_BROWSE_DLG_CONFIG_HA PDH_BROWSE_DLG_CONFIG_H 구조는 PdhBrowseCountersH 함수에서 성능 카운터 찾아보기 대화 상자를 구성하는 데 사용됩니다. (ANSI)
PDH_BROWSE_DLG_CONFIG_HW PDH_BROWSE_DLG_CONFIG_H 구조는 PdhBrowseCountersH 함수에서 성능 카운터 찾아보기 대화 상자를 구성하는 데 사용됩니다. (유니코드)

PDH_BROWSE_DLG_CONFIG_W PDH_BROWSE_DLG_CONFIG 구조는 PdhBrowseCounters 함수에서 성능 카운터 찾아보기 대화 상자를 구성하는 데 사용됩니다. (유니코드)
PDH_COUNTER_INFO_A PDH_COUNTER_INFO 구조에는 카운터의 속성을 설명하는 정보가 포함되어 있습니다. 이 정보에는 카운터 경로도 포함됩니다. (ANSI)
PDH_COUNTER_INFO_W PDH_COUNTER_INFO 구조에는 카운터의 속성을 설명하는 정보가 포함되어 있습니다. 이 정보에는 카운터 경로도 포함됩니다. (유니코드)
PDH_COUNTER_PATH_ELEMENTS_A PDH_COUNTER_PATH_ELEMENTS 구조에는 카운터 경로의 구성 요소가 포함됩니다. (ANSI)
PDH_COUNTER_PATH_ELEMENTS_W PDH_COUNTER_PATH_ELEMENTS 구조에는 카운터 경로의 구성 요소가 포함됩니다. (유니코드)
PDH_DATA_ITEM_PATH_ELEMENTS_A PDH_DATA_ITEM_PATH_ELEMENTS 구조에는 특정 데이터 항목의 경로 요소가 포함됩니다. (ANSI)
PDH_DATA_ITEM_PATH_ELEMENTS_W PDH_DATA_ITEM_PATH_ELEMENTS 구조에는 특정 데이터 항목의 경로 요소가 포함됩니다. (유니코드)
PDH_FMT_COUNTERVALUE PDH_FMT_COUNTERVALUE 구조에는 카운터의 계산 값과 해당 상태가 포함됩니다.
PDH_FMT_COUNTERVALUE_ITEM_A PDH_FMT_COUNTERVALUE_ITEM 구조체에는 카운터의 인스턴스 이름과 서식이 지정된 값이 포함됩니다. (ANSI)
PDH_FMT_COUNTERVALUE_ITEM_W PDH_FMT_COUNTERVALUE_ITEM 구조체에는 카운터의 인스턴스 이름과 서식이 지정된 값이 포함됩니다. (유니코드)

PDH_RAW_COUNTER PDH_RAW_COUNTER 구조체는 카운터 공급자로부터 수집된 데이터를 반환합니다. 데이터에 대해 번역, 서식 또는 기타 해석이 수행되지 않습니다.
PDH_RAW_COUNTER_ITEM_A PDH_RAW_COUNTER_ITEM 구조에는 카운터의 인스턴스 이름과 원시 값이 포함됩니다. (ANSI)
PDH_RAW_COUNTER_ITEM_W PDH_RAW_COUNTER_ITEM 구조에는 카운터의 인스턴스 이름과 원시 값이 포함됩니다. (유니코드)
PDH_RAW_LOG_RECORD PDH_RAW_LOG_RECORD 구조에는 이진 추적 로그 파일 레코드에 대한 정보가 포함됩니다.
PDH_STATISTICS PDH_STATISTICS 구조체에는 원시 카운터 값 배열에 대한 최소값, 최대값 및 평균 값이 포함됩니다.
PDH_TIME_INFO PDH_TIME_INFO 구조에는 성능 데이터 샘플링에 적용되는 시간 간격에 대한 정보가 포함됩니다.
PERF_COUNTER_BLOCK 개체의 카운터에 대한 원시 성능 카운터 데이터를 포함하는 메모리 블록을 설명합니다.
PERF_COUNTER_DATA 구조를 포함하는 PERF_COUNTER_DATA 블록에 대한 정보를 포함합니다.
PERF_COUNTER_DEFINITION 성능 카운터에 대해 설명합니다.
PERF_COUNTER_HEADER 구조를 포함하는 PERF_COUNTER_HEADER 블록에 대한 정보를 포함합니다.
PERF_COUNTER_IDENTIFIER 구조를 포함하는 PERF_COUNTER_IDENTIFIER 블록에 대한 정보를 포함합니다.
PERF_COUNTER_IDENTITY 소비자가 쿼리에서 카운터를 추가하거나 제거할 때 공급자의 콜백으로 전송되는 카운터를 정의합니다.

PERF_COUNTER_INFO 공급자가 사용하는 카운터에 대한 정보를 정의합니다. CTRPP 도구는 지정한 스키마에 따라 이 구조를 자동으로 생성합니다.
PERF_COUNTER_REG_INFO 성능 카운터에 대한 등록 정보를 제공합니다.
PERF_COUNTERSET_INFO 공급자가 사용하는 카운터 집합에 대한 정보를 정의합니다. CTRPP 도구는 지정한 스키마에 따라 이 구조를 자동으로 생성합니다.
PERF_COUNTERSET_INSTANCE 카운터 집합의 인스턴스를 정의합니다.
PERF_COUNTERSET_REG_INFO 구조를 포함하는 PERF_COUNTERSET_REG_INFO 블록에 대한 정보를 포함합니다.
PERF_DATA_BLOCK 쿼리한 성능 데이터 블록(예: 공급자가 반환한 성능 개체 수 및 성능 값을 계산할 때 사용하는 시간 기반 값)에 대해 설명합니다.
PERF_DATA_HEADER 구조를 포함하는 PERF_DATA_HEADER 블록에 대한 정보를 제공합니다.
PERF_INSTANCE_DEFINITION 성능 개체의 인스턴스를 설명합니다.
PERF_INSTANCE_HEADER 구조를 포함하는 PERF_INSTANCE_HEADER 블록에 대한 정보를 제공합니다.
PERF_MULTI_COUNTERS 구조를 포함하는 PERF_MULTI_COUNTERS 블록에 대한 정보를 제공합니다.
PERF_MULTI_INSTANCES 구조를 포함하는 PERF_MULTI_INSTANCES 블록에 대한 정보를 제공합니다.

PERF_OBJECT_TYPE 개체별 성능 정보(예: 개체 인스턴스 수 및 개체가 정의하는 카운터 수)를 설명합니다.
PERF_PROVIDER_CONTEXT 공급자 컨텍스트 정보를 정의합니다.
PERF_STRING_BUFFER_HEADER 구조를 포함하는 PERF_STRING_BUFFER_HEADER 블록에 대한 정보를 제공합니다.
PERF_STRING_COUNTER_HEADER PERF_STRING_BUFFER_HEADER 블록에서 표시된 성능 카운터의 이름 또는 도움말 문자열이 포함된 문자열이 시작되는 위치를 나타냅니다.

성능 로그 및 경고

아티클2023. 03. 09.

성능 로그 및 경고 기술 개요입니다.

성능 로그 및 경고를 개발하려면 다음 헤더가 필요합니다.

- pla.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 성능 로그 및 경고

열거형

AutoPathFormat 파일 이름 또는 하위 디렉터리 이름을 데코레이트하는 방법을 정의합니다.
ClockType 이벤트를 추적할 때 사용할 클록 해상도를 정의합니다.
CommitMode 데이터 수집기 집합에 대한 변경 내용을 커밋할 때 수행할 작업을 정의합니다.
DataCollectorSetStatus 데이터 수집기 집합의 실행 상태를 정의합니다.
DataCollectorType 데이터 수집기 형식을 정의합니다.
DataManagerSteps 데이터 관리자가 실행할 때 수행하는 작업을 정의합니다.
Fileformat 로그 파일의 데이터 형식을 정의합니다.
FolderActionSteps 연령 및 크기 제한이 모두 충족되는 경우 데이터 관리자가 수행하는 작업을 정의합니다.
ResourcePolicy 디스크 리소스 제한 중 하나를 초과할 때 폴더를 삭제하는 방법을 정의합니다.
StreamMode 추적 이벤트가 전달되는 위치를 정의합니다.
ValueMapType 값의 형식을 정의합니다.
평일 데이터 수집기 집합을 실행할 요일을 정의합니다.

Functions

추가 컬렉션에 데이터 수집기를 추가합니다.
추가 컬렉션에 데이터 수집기 집합을 추가합니다.
추가 컬렉션에 폴더 작업을 추가합니다.
추가 컬렉션에 일정을 추가합니다.
추가 컬렉션에 추적 공급자를 추가합니다.
추가 컬렉션에 항목을 추가합니다. (IValueMap.Add)
Addrange 컬렉션에 하나 이상의 데이터 수집기를 추가합니다.
Addrange 컬렉션에 하나 이상의 데이터 수집기 집합을 추가합니다.
Addrange 컬렉션에 하나 이상의 폴더 작업을 추가합니다.
Addrange 컬렉션에 하나 이상의 일정을 추가합니다.
Addrange 컬렉션에 하나 이상의 추적 공급자를 추가합니다.
Addrange 컬렉션에 하나 이상의 항목을 추가합니다.
지우기 컬렉션에서 모든 데이터 수집기를 제거합니다.
지우기 컬렉션에서 모든 데이터 수집기 집합을 제거합니다.
지우기 컬렉션에서 모든 폴더 작업을 제거합니다.
지우기 컬렉션에서 모든 일정을 제거합니다.
지우기 컬렉션에서 모든 추적 공급자를 제거합니다.
지우기 컬렉션에서 모든 항목을 제거합니다.
커밋 데이터 수집기 집합을 저장, 업데이트 또는 유효성 검사합니다. 이 메서드를 사용하여 추적 세션을 플러시할 수도 있습니다.

CreateDataCollector 지정된 형식의 데이터 수집기를 만듭니다.
CreateDataCollectorFromXml 지정된 XML을 사용하여 데이터 수집기를 만듭니다.
CreateFolderAction 폴더 작업 개체를 만듭니다.
CreateSchedule 일정 개체를 만듭니다.
CreateTraceDataProvider 추적 데이터 공급자 개체를 만듭니다.
CreateValueMapItem 값 맵 항목을 만듭니다.
Delete 집합이 실행되고 있지 않으면 데이터 수집기 집합의 지속형 복사본을 삭제합니다.
추출 지정된 CAB 파일을 추출합니다.
get__NewEnum 열거형에 대한 인터페이스를 검색합니다. (IDataCollectorCollection.get__NewEnum)
get__NewEnum 열거형에 대한 인터페이스를 검색합니다. (IDataCollectorSetCollection.get__NewEnum)
get__NewEnum 열거형에 대한 인터페이스를 검색합니다. (IFolderActionCollection.get__NewEnum)
get__NewEnum 열거형에 대한 인터페이스를 검색합니다. (IScheduleCollection.get__NewEnum)
get__NewEnum 열거형에 대한 인터페이스를 검색합니다. (ITraceDataProviderCollection.get__NewEnum)
get__NewEnum 열거형에 대한 인터페이스를 검색합니다. (IValueMap.get__NewEnum)
get_Actions 두 조건(연령 및 크기)이 모두 충족되는 경우 데이터 관리자가 수행할 작업을 검색하거나 설정합니다. (Get)
get_Age 작업을 적용하는 간격을 검색하거나 설정합니다. (Get)
get_AlertThresholds 모니터링할 성능 카운터 및 임계값 목록을 검색하거나 설정합니다. (Get)
get_BufferSize 이벤트 추적 세션에서 각 버퍼에 대해 제안된 버퍼 크기를 검색하거나 설정합니다. (Get)
get_BuffersLost 로그 파일에 기록되지 않은 버퍼 수를 검색합니다.

get_BuffersWritten 로그 파일에 기록된 버퍼 수를 검색합니다.
get_CheckBeforeRunning 데이터를 수집하기 전에 데이터 관리자가 사용 가능한 최소 사용 가능한 디스크 공간과 같은 적용된 제한을 확인해야 하는지 여부를 나타내는 값을 검색하거나 설정합니다. (Get)
get_ClockType 각 이벤트에 대한 타임스탬프를 로깅할 때 사용할 클록 해상도를 검색하거나 설정합니다. (Get)
get_Count 컬렉션의 데이터 수집기 수를 검색합니다.
get_Count 컬렉션의 데이터 수집기 집합 수를 검색합니다.
get_Count 컬렉션의 폴더 작업 수를 검색합니다.
get_Count 컬렉션의 일정 수를 검색합니다.
get_Count 컬렉션의 추적 공급자 수를 검색합니다.
get_Count 컬렉션의 항목 수를 검색합니다. (IValueMap.get_Count)
get_DataCollectors 이 집합의 데이터 수집기 목록을 검색합니다.
get_DataCollectorSet 이 데이터 수집기가 속한 데이터 수집기 집합을 검색합니다.
get_DataCollectorType 이 데이터 수집기의 형식(예: 성능 데이터 수집기)을 검색합니다.
get_DataManager 이 데이터 수집기 집합과 연결된 데이터 관리자를 검색합니다.
get_DataSourceName 로그 파일이 SQL 로그 파일인 경우 데이터 원본 이름을 검색하거나 설정합니다. (Get)
get_Days 데이터 수집기 집합이 실행되는 날짜를 검색하거나 설정합니다. (Get)
get_Description 데이터 수집기 집합에 대한 설명을 검색하거나 설정합니다. 설명은 모든 출력 파일에 메타데이터로 추가되고 성능 데이터 도우미 로그에 주석으로 삽입됩니다. (Get)
get_Description 컬렉션에 대한 설명을 검색하거나 설정합니다. (Get)
get_Description 항목에 대한 설명을 검색하거나 설정합니다. (Get)

get_DescriptionUnresolved 원래 형식으로 설정된 데이터 수집기 설명을 검색합니다.
get_DisplayName 데이터 수집기 집합의 표시 이름을 검색하거나 설정합니다. (Get)
get_DisplayName 공급자의 표시 이름을 검색하거나 설정합니다. (Get)
get_DisplayNameUnresolved 원래 형식으로 설정된 데이터 수집기의 표시 이름을 검색합니다.
get_Duration 데이터 수집기 집합이 실행되는 기간을 검색하고 설정합니다. (Get)
get_Enabled 데이터 관리자를 실행할 수 있는지 여부를 나타내는 값을 검색하거나 설정합니다. (Get)
get_Enabled 항목을 사용할 수 있는지 여부를 나타내는 값을 검색하거나 설정합니다. (Get)
get_EndDate 일정이 유효한 마지막 날짜를 검색하거나 설정합니다. (Get)
get_EventLog 카운터 값이 임계값을 초과할 때마다 PLA가 이벤트를 기록해야 하는지 여부를 나타내는 값을 검색하거나 설정합니다. (Get)
get_EventsFileName 이벤트 파일의 이름을 검색하거나 설정합니다. (Get)
get_EventsLost 버퍼에 기록되지 않은 이벤트 수를 검색합니다.
get_ExcludeApis 추적에서 제외할 함수 목록을 검색하거나 설정합니다. (Get)
get_ExePath 추적하려는 API 호출의 실행 파일 경로를 검색하거나 설정합니다. (Get)
get_ExtendedModes 확장 로그 파일 모드를 검색하거나 설정합니다. (Get)
get_FileMaxCount 수집할 최대 파일 수를 검색하거나 설정합니다. (Get)
get_FileMaxRecursiveDepth 파일을 재귀적으로 수집할 최대 하위 폴더 수를 검색하거나 설정합니다. (Get)
get_FileMaxTotalSize 수집할 수 있는 모든 파일의 최대 총 파일 크기를 검색하거나 설정합니다. (Get)
get_FileName 데이터 수집기 데이터를 포함할 파일의 기본 이름을 검색하거나 설정합니다. (Get)
get_FileNameFormat 파일 이름을 데코레이트하는 방법을 설명하는 플래그를 검색하거나 설정합니다. (Get)

get_FileNameFormatPattern 파일 이름을 데코레이팅할 때 사용할 형식 패턴을 검색하거나 설정합니다. (Get)
get_Files 수집할 파일을 검색하거나 설정합니다. (Get)
get_FilterData 필터링을 위해 추적 데이터 공급자로 전송되는 임의의 데이터를 검색하거나 설정합니다. (Get)
get_FilterEnabled 필터 데이터가 공급자를 사용하도록 설정하는 데 사용되는지 여부를 결정하는 값을 검색하거나 설정합니다. (Get)
get_FilterType 공급자 정의 필터 형식을 검색하거나 설정합니다. (Get)
get_FlushTimer 버퍼를 플러시하기 전에 대기할 시간을 검색하거나 설정합니다. (Get)
get_FolderActions 사용 기간 및 크기 조건이 충족될 때 데이터 수집기 집합의 각 폴더에 대해 수행할 작업을 관리하는 데 사용하는 컬렉션을 검색합니다.
get_FreeBuffers 이벤트 추적 세션의 버퍼 풀에서 할당되었지만 사용되지 않는 버퍼 수를 검색합니다.
get_Guid 세션 GUID를 검색하거나 설정합니다. (Get)
get_Guid 공급자의 GUID를 검색하거나 설정합니다. (Get)
get_IncludeApis 추적에 포함할 함수 목록을 검색하거나 설정합니다. (Get)
get_IncludeModules 추적에 포함할 모듈 목록을 검색하거나 설정합니다. (Get)
get_Index 데이터 수집기의 인덱스 값을 검색합니다. 인덱스 값은 데이터 수집기 집합 내의 데이터 수집기를 식별합니다.
get_IsKernelTrace 추적에 커널 공급자가 포함되어 있는지 여부를 나타내는 값을 검색합니다.
get_Item 컬렉션에서 요청된 데이터 수집기를 검색합니다.
get_Item 컬렉션에서 요청된 데이터 수집기 집합을 검색합니다.
get_Item 컬렉션에서 요청된 폴더 작업을 검색합니다.
get_Item 컬렉션에서 요청된 일정을 검색합니다.
get_Item 컬렉션에서 요청된 추적 공급자를 검색합니다.

get_Item 컬렉션에서 요청된 항목을 검색합니다. (IValueMap.get_Item)
get_Key 항목의 이름을 검색하거나 설정합니다. (Get)
get_Keywords 데이터 수집기 집합을 설명하는 키워드를 검색하거나 설정합니다. 키워드 목록이 출력 파일에 메타데이터로 추가됩니다. (Get)
get_KeywordsAll 공급자가 작성하려는 이벤트의 범주를 제한하는 키워드 목록을 검색합니다.
get_KeywordsAny 공급자가 작성하려는 이벤트의 범주를 결정하는 키워드 목록을 검색합니다.
get_LatestOutputLocation PLA가 마지막으로 파일을 만들 때 사용한 완전히 데코레이트된 파일 이름을 검색하거나 설정합니다. (IDataCollector.get_LatestOutputLocation)
get_LatestOutputLocation 로그가 마지막으로 기록되었을 때 PLA에서 사용한 완전히 데코레이트된 폴더 이름을 검색하거나 설정합니다. (IDataCollectorSet.get_LatestOutputLocation)
get_Level 공급자를 사용하도록 설정하는 데 사용되는 정보 수준을 검색합니다.
get_LogApiNamesOnly PLA가 함수 이름만 기록하는지 여부를 나타내는 값을 검색하거나 설정합니다. (Get)
get_LogApisRecursively API 추적이 애플리케이션에서 직접 가져온 호출을 기록하는지 여부를 나타내는 값을 검색하거나 설정합니다. (Get)
get_LogAppend PLA가 수집된 데이터를 현재 파일에 추가해야 하는지 여부를 나타내는 값을 검색하거나 설정합니다. (Get)
get_LogCircular PLA가 순환 파일을 만들어야 하는지 여부를 나타내는 값을 검색하거나 설정합니다. (Get)
get_LogFileFormat 로그 파일의 형식을 검색하거나 설정합니다. (Get)
get_LogFilePath API 추적 데이터가 포함된 로그 파일의 이름을 검색하거나 설정합니다. (Get)
get_LogOverwrite PLA가 현재 파일을 덮어쓸지 여부를 나타내는 값을 검색하거나 설정합니다. (Get)
get_ManagementQueries 실행할 WMI(Windows Management Instrumentation) 쿼리를 검색하거나 설정합니다. (Get)
get_MaxFolderCount 집합의 모든 데이터 수집기에서 사용할 최대 폴더 수를 검색하거나 설정합니다. (Get)
get_MaximumBuffers 이벤트 추적 세션의 버퍼 풀에 할당된 최대 버퍼 수를 검색하거나 설정합니다. (Get)
get_MaxSize 집합의 모든 데이터 수집기에서 사용할 최대 디스크 공간을 검색하거나 설정합니다. (Get)

get_MinFreeDisk 데이터 수집이 시작되기 전에 존재해야 하는 최소 사용 가능한 디스크 공간을 검색하거나 설정합니다. (Get)
get_MinimumBuffers 이벤트 추적 세션의 버퍼 풀에 할당된 최소 버퍼 수를 검색하거나 설정합니다. (Get)
get_Name 데이터 수집기의 이름을 검색하거나 설정합니다. (Get)
get_Name 데이터 수집기 집합을 식별하는 데 사용되는 고유한 이름을 검색합니다.
get_NumberOfBuffers 로깅에 사용할 제안된 버퍼 수를 검색하거나 설정합니다. (Get)
get_OutputLocation PLA가 지금 만드는 경우 데코레이트된 파일 이름을 검색합니다.
get_OutputLocation PLA가 지금 만드는 경우 데코레이트된 폴더 이름을 검색합니다.
get_PerformanceCounters 쿼리할 성능 카운터를 검색하거나 설정합니다. (Get)
get_PreallocateFile 로깅 전에 PLA가 전체 로그 파일 크기를 할당해야 하는지 여부를 나타내는 값을 검색하거나 설정합니다. (Get)
get_ProcessMode 세션이 프라이빗 in-process 세션인지 여부를 나타내는 값을 검색하거나 설정합니다. (Get)
get_Properties ETW(Windows용 이벤트 추적)에서 이벤트와 함께 포함하는 확장 데이터 항목 목록을 검색합니다.
get_QueryNetworkAdapters 설치된 네트워크 어댑터가 오프로드 기능 및 기타 구성 정보에 대해 쿼리되는지 여부를 나타내는 값을 검색하거나 설정합니다. (Get)
get_RealTimeBuffersLost 소비자에게 실시간으로 전달되지 않은 버퍼 수를 검색합니다.
get_RegistryKeys 수집할 레지스트리 키 목록을 검색하거나 설정합니다.
get_RegistryMaxRecursiveDepth 레지스트리 값을 재귀적으로 수집할 최대 하위 키 수를 검색하거나 설정합니다. (Get)
get_ReportFileName IDataManager::RuleTargetFileName 속성의 파일을 XML에서 HTML로 변환한 결과 HTML 파일의 이름을 검색하거나 설정합니다. (Get)
get_ReportSchema TraceRpt.exe 애플리케이션에서 생성하는 보고서를 사용자 지정하는 데 사용되는 스키마를 검색하거나 설정합니다. (Get)
get_ResourcePolicy 디스크 리소스 제한 중 하나를 초과할 때 수행할 작업을 검색하거나 설정합니다. (Get)
get_RootPath 하위 디렉터리가 만들어지는 기본 경로를 검색하거나 설정합니다. (Get)

get_Rules 보고서에 적용할 규칙을 검색하거나 설정합니다. (Get)
get_RuleTargetFileName TraceRpt.exe 애플리케이션이 만드는 보고서 파일의 이름을 검색하거나 설정합니다. (IDataManager.get_RuleTargetFileName)
get_SampleInterval 샘플링 카운터 데이터 간에 대기할 시간 간격을 검색하거나 설정합니다. (Get)
get_SampleInterval 샘플링 카운터 데이터 간에 대기할 간격을 검색하거나 설정합니다. (Get)
get_Schedules 데이터 수집기 집합이 실행되는 시기를 결정하는 일정 목록을 검색합니다.
get_SchedulesEnabled 일정을 사용할 수 있는지 여부를 나타내는 값을 검색하거나 설정합니다. (Get)
get_Security 이 데이터 수집기 집합에 액세스할 수 있는 사용자를 결정하는 액세스 제어 정보를 검색하거나 설정합니다. (Get)
get_Segment 데이터 수집기 집합이 중지되기 전에 최대 크기 또는 세그먼트 기간에 도달하는 경우 PLA가 새 로그를 만드는지 여부를 나타내는 값을 검색하거나 설정합니다. (Get)
get_SegmentMaxDuration 데이터 수집기 집합이 새 로그 파일에 쓰기 시작하기 전에 실행할 수 있는 기간을 검색하거나 설정합니다. (Get)
get_SegmentMaxRecords 기록할 최대 샘플 수를 검색하거나 설정합니다. (Get)
get_SegmentMaxSize 데이터 수집기 집합에서 로그 파일의 최대 크기를 검색하거나 설정합니다. (Get)
get_SendCabTo 작업이 캐비닛 파일을 보내는 경우 캐비닛 파일의 대상을 검색하거나 설정합니다. (Get)
get_SerialNumber 세그먼트를 포함하여 이 데이터 수집기 집합이 시작된 횟수를 검색하거나 설정합니다. (Get)
get_Server 데이터 수집기 집합이 실행되는 서버의 이름을 검색합니다.
get_SessionId 세션 식별자를 검색합니다.
get_SessionName 세션의 이름을 검색하거나 설정합니다. (Get)
get_SessionThreadId 스레드가 실행 중인 경우 로그 세션의 현재 스레드를 검색합니다.
get_Size 초과하면 작업을 시작하는 최소 폴더 크기를 검색하거나 설정합니다. (Get)

get_StartDate 일정이 유효한 날짜를 검색하거나 설정합니다. (Get)
get_StartTime 데이터 수집기 집합이 실행되는 시간을 검색하거나 설정합니다. (Get)
get_Status 데이터 수집기 집합의 상태를 검색합니다.
get_StopOnCompletion 집합의 모든 데이터 수집기가 완료된 상태일 때 데이터 수집기 집합이 중지되는지 여부를 결정하는 값을 검색하거나 설정합니다. (Get)
get_StreamMode 추적 세션의 로깅 모드를 검색하거나 설정합니다. (Get)
get_Subdirectory 데이터 수집기 집합의 다음 인스턴스가 로그를 작성하는 루트 경로의 기본 하위 디렉터리를 검색하거나 설정합니다. (Get)
get_SubdirectoryFormat 하위 디렉터리 이름을 데코레이트하는 방법을 설명하는 플래그를 검색하거나 설정합니다. (Get)
get_SubdirectoryFormatPattern 폴더 이름을 데코레이팅할 때 사용할 형식 패턴을 검색하거나 설정합니다. (Get)
get_SystemStateFile 저장된 시스템 상태가 포함된 파일의 이름을 검색하거나 설정합니다. (Get)
get_Task 카운터 값이 임계값을 초과할 때마다 시작할 작업 스케줄러 작업의 이름을 검색하거나 설정합니다. (Get)
get_Task 세그먼트 간을 포함하여 데이터 수집기 집합이 중지될 때마다 시작할 작업 스케줄러 작업의 이름을 검색하거나 설정합니다. (Get)
get_TaskArguments IAlertDataCollector::Task 속성에 지정된 작업 스케줄러 작업에 전달할 명령줄 인수를 검색하거나 설정합니다. (IAlertDataCollector.get_TaskArguments)
get_TaskArguments IDataCollectorSet::Task 속성에 지정된 작업 스케줄러 작업에 전달할 명령줄 인수를 검색하거나 설정합니다. (Get)
get_TaskRunAsSelf 태스크가 데이터 수집기 집합 사용자로 실행되는지 또는 태스크에 지정된 사용자로 실행되는지 여부를 결정하는 값을 검색하거나 설정합니다. (IAlertDataCollector.get_TaskRunAsSelf)
get_TaskRunAsSelf 태스크가 데이터 수집기 집합 사용자로 실행되는지 또는 태스크에 지정된 사용자로 실행되는지 여부를 결정하는 값을 검색하거나 설정합니다. (IDataCollectorSet.get_TaskRunAsSelf)
get_TaskUserTextArguments IAlertDataCollector::Task 속성에 지정된 작업 스케줄러 작업에 전달할 명령줄 인수를 검색하거나 설정합니다. (IAlertDataCollector.get_TaskUserTextArguments)
get_TaskUserTextArguments IDataCollectorSet::TaskArguments 속성에서 {usertext} 대체 변수로 대체되는 명령줄 인수를 검색하거나 설정합니다. (Get)
get_TraceDataProviders 이 추적 세션에 대해 사용하도록 설정된 공급자 목록을 검색합니다.

get_TriggerDataCollectorSet 카운터 값이 임계값을 초과할 때마다 시작할 데이터 수집기 집합의 이름을 검색하거나 설정합니다. (Get)
get_UserAccount 데이터 수집기 집합이 실행될 사용자 계정을 검색합니다.
get_Value 컬렉션의 값을 검색하거나 설정합니다. (Get)
get_Value 항목의 값을 검색하거나 설정합니다. (Get)
get_ValueMapType 컬렉션의 항목 유형을 검색하거나 설정합니다. (Get)
get_ValueMapType 항목의 형식을 검색하거나 설정합니다. (Get)
get_Xml 데이터 수집기 속성의 값을 설명하는 XML 문자열을 검색합니다.
get_Xml 집합에 포함된 데이터 수집기 속성을 포함하여 데이터 수집기 집합 속성의 값을 설명하는 XML 문자열을 검색합니다.
GetDataCollectorSets 데이터 수집기 집합 컬렉션을 채웁니다.
GetRegisteredProcesses ETW(Windows용 이벤트 추적) 공급자로 등록된 프로세스 목록을 검색합니다.
GetSecurity 추적 데이터 공급자에 대한 보안 정보를 검색합니다.
GetTraceDataProviders 등록된 추적 공급자로 컬렉션을 채웁니다.
GetTraceDataProvidersByProcess 지정된 프로세스에 의해 등록된 공급자 목록으로 컬렉션을 채웁니다.
GetValue 사용자 정의 값을 검색합니다.
put_Actions 두 조건(연령 및 크기)이 모두 충족되는 경우 데이터 관리자가 수행할 작업을 검색하거나 설정합니다. (Put)
put_Age 작업 적용 사이에 대기할 간격을 검색하거나 설정합니다. (Put)
put_AlertThresholds 모니터링할 성능 카운터 및 임계값 목록을 검색하거나 설정합니다. (Put)
put_BufferSize 이벤트 추적 세션에서 각 버퍼에 대해 제안된 버퍼 크기를 검색하거나 설정합니다. (Put)

put_CheckBeforeRunning 데이터를 수집하기 전에 데이터 관리자가 사용 가능한 최소 디스크 공간과 같은 부과된 제한을 확인해야 하는지 여부를 나타내는 값을 검색하거나 설정합니다. (Put)
put_ClockType 각 이벤트에 대한 타임스탬프를 로깅할 때 사용할 클록 해상도를 검색하거나 설정합니다. (Put)
put_DataSourceName 로그 파일이 SQL 로그 파일인 경우 데이터 원본 이름을 검색하거나 설정합니다. (Put)
put_Days 데이터 수집기 집합이 실행되는 날짜를 검색하거나 설정합니다. (Put)
put_Description 데이터 수집기 집합에 대한 설명을 검색하거나 설정합니다. 설명은 모든 출력 파일에 메타데이터로 추가되고 성능 데이터 도우미 로그에 주석으로 삽입됩니다. (Put)
put_Description 컬렉션에 대한 설명을 검색하거나 설정합니다. (Put)
put_Description 항목에 대한 설명을 검색하거나 설정합니다. (Put)
put_DisplayName 데이터 수집기 집합의 표시 이름을 검색하거나 설정합니다. (Put)
put_DisplayName 공급자의 표시 이름을 검색하거나 설정합니다. (Put)
put_Duration 데이터 수집기 집합이 실행되는 기간을 검색하고 설정합니다. (Put)
put_Enabled 데이터 관리자를 실행할 수 있는지 여부를 나타내는 값을 검색하거나 설정합니다. (Put)
put_Enabled 항목을 사용할 수 있는지 여부를 나타내는 값을 검색하거나 설정합니다. (Put)
put_EndDate 일정이 유효한 마지막 날짜를 검색하거나 설정합니다. (Put)
put_EventLog 카운터 값이 임계값을 초과할 때마다 PLA가 이벤트를 기록해야 하는지 여부를 나타내는 값을 검색하거나 설정합니다. (Put)
put_EventsFileName 이벤트 파일의 이름을 검색하거나 설정합니다. (Put)
put_ExcludeApis 추적에서 제외할 함수 목록을 검색하거나 설정합니다. (Put)
put_ExePath 추적하려는 API 호출의 실행 파일 경로를 검색하거나 설정합니다. (Put)
put_ExtendedModes 확장 로그 파일 모드를 검색하거나 설정합니다. (Put)

put_FileMaxCount 수집할 최대 파일 수를 검색하거나 설정합니다. (Put)
put_FileMaxRecursiveDepth 파일을 재귀적으로 수집할 최대 하위 폴더 수를 검색하거나 설정합니다. (Put)
put_FileMaxTotalSize 수집할 수 있는 모든 파일의 최대 총 파일 크기를 검색하거나 설정합니다. (Put)
put_FileName 데이터 수집기 데이터를 포함할 파일의 기본 이름을 검색하거나 설정합니다. (Put)
put_FileNameFormat 파일 이름을 데코레이트하는 방법을 설명하는 플래그를 검색하거나 설정합니다. (Put)
put_FileNameFormatPattern 파일 이름을 데코레이팅할 때 사용할 형식 패턴을 검색하거나 설정합니다. (Put)
put_Files 수집할 파일을 검색하거나 설정합니다. (Put)
put_FilterData 필터링을 위해 추적 데이터 공급자로 전송되는 임의의 데이터를 검색하거나 설정합니다. (Put)
put_FilterEnabled 필터 데이터가 공급자를 사용하도록 설정하는 데 사용되는지 여부를 결정하는 값을 검색하거나 설정합니다. (Put)
put_FilterType 공급자 정의 필터 형식을 검색하거나 설정합니다. (Put)
put_FlushTimer 버퍼를 플러시하기 전에 대기할 시간을 검색하거나 설정합니다. (Put)
put_Guid 세션 GUID를 검색하거나 설정합니다. (Put)
put_Guid 공급자의 GUID를 검색하거나 설정합니다. (Put)
put_IncludeApis 추적에 포함할 함수 목록을 검색하거나 설정합니다. (Put)
put_IncludeModules 추적에 포함할 모듈 목록을 검색하거나 설정합니다. (Put)
put_Key 항목의 이름을 검색하거나 설정합니다. (Put)
put_Keywords 데이터 수집기 집합을 설명하는 키워드를 검색하거나 설정합니다. 키워드 목록이 출력 파일에 메타데이터로 추가됩니다. (Put)
put_LatestOutputLocation PLA가 마지막으로 파일을 만들 때 사용한 완전히 데코레이트된 파일 이름을 검색하거나 설정합니다. (IDataCollector.put_LatestOutputLocation)
put_LatestOutputLocation 로그가 마지막으로 기록되었을 때 PLA에서 사용한 완전히 데코레이트된 폴더 이름을 검색하거나 설정합니다. (IDataCollectorSet.put_LatestOutputLocation)

put_LogApiNamesOnly PLA가 함수 이름만 기록하는지 여부를 나타내는 값을 검색하거나 설정합니다. (Put)
put_LogApisRecursively API 추적이 애플리케이션에서 직접 가져온 호출을 기록하는지 여부를 나타내는 값을 검색하거나 설정합니다. (Put)
put_LogAppend PLA가 수집된 데이터를 현재 파일에 추가해야 하는지 여부를 나타내는 값을 검색하거나 설정합니다. (Put)
put_LogCircular PLA가 순환 파일을 만들어야 하는지 여부를 나타내는 값을 검색하거나 설정합니다. (Put)
put_LogFileFormat 로그 파일의 형식을 검색하거나 설정합니다. (Put)
put_LogFilePath API 추적 데이터가 포함된 로그 파일의 이름을 검색하거나 설정합니다. (Put)
put_LogOverwrite PLA가 현재 파일을 덮어쓸지 여부를 나타내는 값을 검색하거나 설정합니다. (Put)
put_ManagementQueries 실행할 WMI(Windows Management Instrumentation) 쿼리를 검색하거나 설정합니다. (Put)
put_MaxFolderCount 집합의 모든 데이터 수집기에서 사용할 최대 폴더 수를 검색하거나 설정합니다. (Put)
put_MaximumBuffers 이벤트 추적 세션의 버퍼 풀에 할당된 최대 버퍼 수를 검색하거나 설정합니다. (Put)
put_MaxSize 집합의 모든 데이터 수집기에서 사용할 최대 디스크 공간을 검색하거나 설정합니다. (Put)
put_MinFreeDisk 데이터 수집이 시작되기 전에 존재해야 하는 최소 사용 가능한 디스크 공간을 검색하거나 설정합니다. (Put)
put_MinimumBuffers 이벤트 추적 세션의 버퍼 풀에 할당된 최소 버퍼 수를 검색하거나 설정합니다. (Put)
put_Name 데이터 수집기의 이름을 검색하거나 설정합니다. (Put)
put_NumberOfBuffers 로깅에 사용할 제안된 버퍼 수를 검색하거나 설정합니다. (Put)
put_PerformanceCounters 쿼리할 성능 카운터를 검색하거나 설정합니다. (Put)
put_PreallocateFile 로깅 전에 PLA가 전체 로그 파일 크기를 할당해야 하는지 여부를 나타내는 값을 검색하거나 설정합니다. (Put)
put_ProcessMode 세션이 프라이빗 in-process 세션인지 여부를 나타내는 값을 검색하거나 설정합니다. (Put)
put_QueryNetworkAdapters 설치된 네트워크 어댑터가 오프로드 기능 및 기타 구성 정보에 대해 쿼리되는지 여부를 나타내는 값을 검색하거나 설정합니다. (Put)

put_RegistryKeys 수집할 레지스트리 키 목록을 검색하거나 설정합니다. (Put)
put_RegistryMaxRecursiveDepth 레지스트리 값을 재귀적으로 수집할 최대 하위 키 수를 검색하거나 설정합니다. (Put)
put_ReportFileName IDataManager::RuleTargetFileName 속성의 파일을 XML에서 HTML로 변환한 결과 HTML 파일의 이름을 검색하거나 설정합니다. (Put)
put_ReportSchema TraceRpt.exe 애플리케이션에서 생성하는 보고서를 사용자 지정하는 데 사용되는 스키마를 검색하거나 설정합니다. (Put)
put_ResourcePolicy 디스크 리소스 제한 중 하나를 초과할 때 수행할 작업을 검색하거나 설정합니다. (Put)
put_RootPath 하위 디렉터리가 만들어지는 기본 경로를 검색하거나 설정합니다. (Put)
put_Rules 보고서에 적용할 규칙을 검색하거나 설정합니다. (Put)
put_RuleTargetFileName TraceRpt.exe 애플리케이션이 만드는 보고서 파일의 이름을 검색하거나 설정합니다. (IDataManager.put_RuleTargetFileName)
put_SampleInterval 샘플링 카운터 데이터 간에 대기할 시간 간격을 검색하거나 설정합니다. (Put)
put_SampleInterval 샘플링 카운터 데이터 간에 대기할 간격을 검색하거나 설정합니다. (Put)
put_SchedulesEnabled 일정을 사용할 수 있는지 여부를 나타내는 값을 검색하거나 설정합니다. (Put)
put_Security 이 데이터 수집기 집합에 액세스할 수 있는 사용자를 결정하는 액세스 제어 정보를 검색하거나 설정합니다. (Put)
put_Segment 데이터 수집기 집합이 중지되기 전에 최대 크기 또는 세그먼트 기간에 도달하는 경우 PLA가 새 로그를 만드는지 여부를 나타내는 값을 검색하거나 설정합니다. (Put)
put_SegmentMaxDuration 데이터 수집기 집합이 새 로그 파일에 쓰기 시작하기 전에 실행할 수 있는 기간을 검색하거나 설정합니다. (Put)
put_SegmentMaxRecords 기록할 최대 샘플 수를 검색하거나 설정합니다. (Put)
put_SegmentMaxSize 데이터 수집기 집합에서 로그 파일의 최대 크기를 검색하거나 설정합니다. (Put)
put_SendCabTo 작업이 캐비닛 파일을 보내는 경우 캐비닛 파일의 대상을 검색하거나 설정합니다. (Put)
put_SerialNumber 세그먼트를 포함하여 이 데이터 수집기 집합이 시작된 횟수를 검색하거나 설정합니다. (Put)

put_SessionName 세션의 이름을 검색하거나 설정합니다. (Put)
put_Size 초과하면 작업을 시작하는 최소 폴더 크기를 검색하거나 설정합니다. (Put)
put_StartDate 일정이 유효한 날짜를 검색하거나 설정합니다. (Put)
put_StartTime 데이터 수집기 집합이 실행되는 시간을 검색하거나 설정합니다. (Put)
put_StopOnCompletion 집합의 모든 데이터 수집기가 완료된 상태일 때 데이터 수집기 집합이 중지되는지 여부를 결정하는 값을 검색하거나 설정합니다. (Put)
put_StreamMode 추적 세션의 로깅 모드를 검색하거나 설정합니다. (Put)
put_Subdirectory 데이터 수집기 집합의 다음 인스턴스가 로그를 작성하는 루트 경로의 기본 하위 디렉터리를 검색하거나 설정합니다. (Put)
put_SubdirectoryFormat 하위 디렉터리 이름을 데코레이트하는 방법을 설명하는 플래그를 검색하거나 설정합니다. (Put)
put_SubdirectoryFormatPattern 폴더 이름을 데코레이팅할 때 사용할 형식 패턴을 검색하거나 설정합니다. (Put)
put_SystemStateFile 저장된 시스템 상태가 포함된 파일의 이름을 검색하거나 설정합니다. (Put)
put_Task 카운터 값이 임계값을 초과할 때마다 시작할 작업 스케줄러 작업의 이름을 검색하거나 설정합니다. (Put)
put_Task 세그먼트 간을 포함하여 데이터 수집기 집합이 중지될 때마다 시작할 작업 스케줄러 작업의 이름을 검색하거나 설정합니다. (Put)
put_TaskArguments IAlertDataCollector::Task 속성에 지정된 작업 스케줄러 작업에 전달할 명령줄 인수를 검색하거나 설정합니다. (IAlertDataCollector.put_TaskArguments)
put_TaskArguments IDataCollectorSet::Task 속성에 지정된 작업 스케줄러 작업에 전달할 명령줄 인수를 검색하거나 설정합니다. (Put)
put_TaskRunAsSelf 태스크가 데이터 수집기 집합 사용자로 실행되는지 또는 태스크에 지정된 사용자로 실행되는지 여부를 결정하는 값을 검색하거나 설정합니다. (IAlertDataCollector.put_TaskRunAsSelf)
put_TaskRunAsSelf 태스크가 데이터 수집기 집합 사용자로 실행되는지 또는 태스크에 지정된 사용자로 실행되는지 여부를 결정하는 값을 검색하거나 설정합니다. (IDataCollectorSet.put_TaskRunAsSelf)
put_TaskUserTextArguments IAlertDataCollector::Task 속성에 지정된 작업 스케줄러 작업에 전달할 명령줄 인수를 검색하거나 설정합니다. (IAlertDataCollector.put_TaskUserTextArguments)
put_TaskUserTextArguments IDataCollectorSet::TaskArguments 속성에서 {usertext} 대체 변수로 대체되는 명령줄 인수를 검색하거나 설정합니다. (Put)

put_TriggerDataCollectorSet 카운터 값이 임계값을 초과할 때마다 시작할 데이터 수집기 집합의 이름을 검색하거나 설정합니다. (Put)
put_Value 컬렉션의 값을 검색하거나 설정합니다. (Put)
put_Value 항목의 값을 검색하거나 설정합니다. (Put)
put_ValueMapType 컬렉션의 항목 유형을 검색하거나 설정합니다. (Put)
put_ValueMapType 항목의 형식을 검색하거나 설정합니다. (Put)
쿼리 지정된 데이터 수집기 집합을 검색합니다.
쿼리 등록된 공급자에 대한 세부 정보를 검색합니다.
제거 컬렉션에서 데이터 수집기를 제거합니다.
제거 컬렉션에서 데이터 수집기 집합을 제거합니다.
제거 지정된 인덱스에 따라 컬렉션에서 폴더 작업을 제거합니다.
제거 컬렉션에서 일정을 제거합니다.
제거 컬렉션에서 추적 공급자를 제거합니다.
제거 컬렉션에서 항목을 제거합니다.
해결 공급자에 대한 세부 정보를 이 인스턴스와 병합합니다.
실행 데이터 관리자를 수동으로 실행합니다.
SetCredentials 데이터 수집기 집합이 실행되는 사용자 계정을 지정합니다.
SetSecurity 추적 데이터 공급자에 대한 보안 정보를 설정합니다.
SetValue 사용자 정의 값을 설정합니다.
SetXml XML에 포함된 속성의 속성 값을 설정합니다. (IDataCollector.SetXml)

SetXml XML에 포함된 속성의 속성 값을 설정합니다. (IDataCollectorSet.SetXml)
시작 데이터 수집기 집합을 수동으로 시작합니다.
중지 데이터 수집기 집합을 수동으로 중지합니다.

인터페이스

IAlertDataCollector 카운터 값이 지정된 임계값을 초과할 때마다 성능 카운터를 모니터링하고 작업을 수행합니다. 경고 데이터 수집기를 만들려면 IDataCollectorCollection::CreateDataColl IDataCollectorCollection::CreateDataCollectorFromXml 메서드를 호출합니다. CreateDataCollectorFromXml에 전달하는 XML에 대한 자세한 내용은 비고를 참조하세요.
IApiTracingDataCollector Win32 호출을 Kernel32.dll, Advapi32.dll, Gdi32.dll 및 User32.dll 기록합니다.
IConfigurationDataCollector 컬렉션 시 컴퓨터 설정을 수집합니다.
IDataCollector XML을 사용하여 수집기 속성을 설정하고 검색하고, 로그 파일 이름을 지정하고, 로그 파일의 위치를 검색합니다. 이 인터페이스는 다음 데이터 수집기가 파생되는 추상 다.IAlertDataCollectorIApiTracingDataCollectorIConfigurationDataCollectorIPerformanceCounterDataCollectorITraceDataCollector
IDataCollectorCollection DataCollector 개체의 컬렉션을 관리합니다. 이 인터페이스를 얻으려면 IDataCollectorSet::D ataCollectors 속성에 액세스합니다.
IDataCollectorSet 집합의 모든 데이터 수집기 개체에 공통적인 구성 정보를 관리합니다. 는 집합에서 데이터 수집기를 추가하고 제거합니다. 및 는 데이터 수집을 시작합니다. 사용하는 니다.
IDataCollectorSetCollection DataCollectorSet 개체의 컬렉션을 관리합니다. 이 인터페이스를 얻으려면 coCreateInstance 함수를 호출하여 __uuidof(DataCollectorSetCollection)을 클래스 식별자로 __uuidof(IDataCollectorSetCollection)을 인터페이스 식별자로 전달합니다. 그런 다음 컬렉션을 채웁니다. IDataCollectorSetCollection::GetDataCollectorSets 메서드를 호
IDataManager 보고서 생성, 데이터 보존 정책 및 데이터 전송을 포함하여 데이터 수집기에서 생성된 데이터를 관리합니다. 이 인터페이스를 얻으려면 IDataCollectorSet::D ataManage 다.
IFolderAction 두 조건(연령 및 크기)이 모두 충족되는 경우 데이터 관리자가 데이터 수집기 집합의 루트 경로 아래에 있는 각 폴더에 대해 수행할 작업을 지정합니다. 이 인터페이스를 IFolderActionCollection::CreateFolderAction 메서드를 호출합니다.
IFolderActionCollection FolderAction 개체의 컬렉션을 관리합니다. 이 인터페이스를 얻으려면 IDataManager::FolderActions 속성에 액세스합니다.
IPerformanceCounterDataCollector 쿼리할 성능 카운터와 카운터 데이터가 기록되는 로그 파일을 지정합니다. 이 데이터 수집기를 만들려면 IDataCollectorCollection::CreateDataCollector 또는 IDataCollectorCollection::CreateDataCollectorFromXml 메서드를 호출합니다.
ISchedule 데이터 수집기 집합이 실행되는 시기를 지정합니다. 이 인터페이스를 얻으려면 IScheduleCollection::CreateSchedule 메서드를 호출합니다.

IScheduleCollection Schedule 개체의 컬렉션을 관리합니다. 이 인터페이스를 얻으려면 IDataCollectorSet::Schedules 속성에 액세스합니다.
ITraceDataCollector 등록된 공급자에서 추적 이벤트를 수집합니다. 이 인터페이스는 추적 세션을 정의합니다.
ITraceDataProvider 추적 세션에서 사용하도록 설정할 추적 공급자를 지정합니다.
ITraceDataProviderCollection TraceDataProvider 개체의 컬렉션을 관리합니다. 이 인터페이스를 얻으려면 ITraceDataCollector::TraceDataProviders 속성에 액세스합니다. CoCreateInstance 함수를 호 TraceDataProviderCollection 개체의 새 인스턴스를 만들 수도 있습니다.
IValueMap 이름/값 쌍의 컬렉션을 관리합니다. 이 인터페이스를 얻으려면 다음 속성 또는 메서드 중 하나에 액세스합니 다.IDataCollector::SetXmlIDataCollectorSet::CommitIDataCollectorSet::SetXmlITraceDataProvider::KeywordsAllITraceDataProvider::KeywordsAnyITraceDataProvider::Leve roperties
IValueMapItem 이름/값 쌍을 정의합니다. 이 인터페이스를 얻으려면 IValueMap::Item 속성을 호출합니다. 이 인터페이스를 만들려면 IValueMap::CreateValueMapItem 메서드를 호출합

그림 구입

아티클2023. 03. 09.

그림 획득 기술에 대한 개요입니다.

그림 구입을 개발하려면 다음 헤더가 필요합니다.

- photoacquire.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 그림 구입

열거형

DEVICE_SELECTION_DEVICE_TYPE DEVICE_SELECTION_DEVICE_TYPE 열거형 유형은 선택한 디바이스의 유형을 나타냅니다.
ERROR_ADVISE_MESSAGE_TYPE ERROR_ADVISE_MESSAGE_TYPE 열거형 형식은 IPhotoAcquireProgressCB::ErrorAdvise의 nMessageType 매개 변수에 전달할 수 있는 오류 값의 형식을 나타냅니다.
ERROR_ADVISE_RESULT ERROR_ADVISE_RESULT 열거형 형식은 IPhotoAcquireProgressCB::ErrorAdvise의 pnErrorAdviseResult 매개 변수에 할당할 수 있는 오류 값의 형식을 나타냅니다.
PROGRESS_DIALOG_CHECKBOX_ID PROGRESS_DIALOG_CHECKBOX_ID 열거형 형식은 IPhotoProgressDialog 개체의 확인란을 나타냅니다.
PROGRESS_DIALOG_IMAGE_TYPE PROGRESS_DIALOG_IMAGE_TYPE 열거형 형식은 IPhotoProgressDialog::SetImage에 설정된 이미지 형식을 나타냅니다.
USER_INPUT_STRING_TYPE USER_INPUT_STRING_TYPE 열거형 형식은 IPhotoAcquireProgressCB::GetUserInput의 사용자로부터 가져올 문자열의 형식을 나타냅니다.

Functions

획득 Acquire 메서드는 디바이스에서 사진을 가져옵니다.
취소 Cancelled 메서드는 취득 세션 중에 취소가 발생할 때 확장된 기능을 제공합니다. 애플리케이션은 Cancelled 메서드의 구현을 제공합니다.
CanDelete CanDelete 메서드는 항목을 삭제할 수 있는지 여부를 나타냅니다.
만들기 Create 메서드는 부모 창 내에서 호스트되는 사진 옵션 대화 상자의 모델리스 인스턴스를 만들고 표시합니다.
만들기 Create 메서드는 이미지 열거 및 획득 중에 표시할 수 있는 진행률 대화 상자를 만들고 표시합니다.
CreatePhotoSource CreatePhotoSource 메서드는 IPhotoAcquire::Acquire에 전달할 IPhotoAcquireSource 개체를 초기화합니다.
삭제 Delete 메서드는 항목을 삭제합니다.
파괴 Destroy 메서드는 Create 메서드를 사용하여 만든 모델리스 대화 상자를 닫고 삭제합니다.
파괴 Destroy 메서드는 이미지 열거 및 획득 중에 표시되는 진행률 대화 상자를 닫고 삭제합니다.
DirectoryCreated DirectoryCreated 메서드는 획득 프로세스 중에 대상 디렉터리를 만들 때 확장된 기능을 제공합니다. 애플리케이션은 DirectoryCreated 메서드의 구현을 제공합니다.

DisplayConfigureDialog DisplayConfigureDialog 메서드는 구성 대화 상자가 표시될 때 확장된 기능을 제공합니다. 애플리케이션은 DisplayConfigureDialog 메서드의 구현을 제공합니다.
Domodal DoModal 메서드는 디바이스 선택 대화 상자를 표시합니다. 사용자가 모달 대화 상자를 사용하여 디바이스를 선택하면 함수가 반환됩니다.
Domodal DoModal 메서드는 옵션 대화 상자를 모달 대화 상자로 만들고 표시합니다.
EndDelete EndDelete 메서드는 이미지 원본에서 파일을 삭제할 때 확장된 기능을 제공합니다. 애플리케이션은 EndDelete 메서드의 구현을 제공합니다.
EndEnumeration EndEnumeration 메서드는 이미지 원본의 파일 열거가 완료되면 확장된 기능을 제공합니다. 애플리케이션은 EndEnumeration 메서드의 구현을 제공합니다.
EndItemDelete EndItemDelete 메서드는 이미지 원본에서 파일을 삭제할 때마다 확장된 기능을 제공합니다. 애플리케이션은 EndItemDelete 메서드의 구현을 제공합니다.
EndItemTransfer EndItemTransfer 메서드는 이미지 원본에서 파일을 전송할 때마다 확장된 기능을 제공합니다. 애플리케이션은 EndItemTransfer 메서드의 구현을 제공합니다.
EndSession EndSession 메서드는 취득 세션이 완료되면 확장된 기능을 제공합니다. 애플리케이션은 EndSession 메서드의 구현을 제공합니다.
EndTransfer EndTransfer 메서드는 모든 파일의 전송이 완료되면 확장된 기능을 제공합니다. 애플리케이션은 EndTransfer 메서드의 구현을 제공합니다.
EnumResults EnumResults 메서드는 가장 최근에 Acquire를 호출하는 동안 성공적으로 전송된 모든 파일의 경로를 포함하는 열거형을 검색합니다.

ErrorAdvise ErrorAdvise 메서드는 취득 중에 발생하는 오류에 대한 사용자 지정 오류 처리를 제공합니다. 애플리케이션은 ErrorAdvise 메서드의 구현을 제공합니다.
FoundItem FoundItem 메서드는 디바이스에서 항목을 열거하는 동안 항목을 찾을 때마다 확장된 기능을 제공합니다.
GetAcquisitionTime GetAcquisitionTime 메서드는 현재 세션의 획득 시간을 검색합니다.
GetDefault GetDefault 메서드는 편집 컨트롤(또는 해당)을 초기화하는 데 사용되는 기본 문자열을 검색합니다.
GetDeleteAfterAcquire GetDeleteAfterAcquire 메서드는 구입 후 사진을 삭제해야 하는지 여부를 나타내는 값을 반환합니다.
GetDeviceIcons GetDeviceIcons 메서드는 디바이스를 나타내는 데 사용되는 아이콘을 검색합니다.
GetDeviceId GetDeviceId 메서드는 디바이스의 식별자(ID)를 검색합니다.
GetFlags GetFlags 메서드는 사진 획득 플래그를 검색합니다.
GetFriendlyName GetFriendlyName 메서드는 표시 형식이 지정된 디바이스의 이름을 검색합니다.
GetGroupTag GetGroupTag 메서드는 디바이스에서 다운로드되는 파일 그룹에 대한 태그 문자열을 검색합니다.
GetImage GetImage 메서드는 편집 컨트롤을 초기화하는 데 사용되는 기본 이미지를 검색합니다.

GetItemAt GetItemAt 메서드는 항목 목록의 지정된 인덱스에서 IPhotoAcquireItem 개체를 검색합니다.
GetItemCount GetItemCount 메서드는 InitializeItemList 메서드에서 찾은 항목 수를 검색합니다.
GetItemName GetItemName 메서드는 항목의 파일 이름을 검색합니다.
GetMaxLength GetMaxLength 메서드는 UI(사용자 인터페이스)에서 허용해야 하는 최대 문자열 길이를 검색합니다.
GetMruCount GetMruCount 메서드는 가장 최근에 사용한 항목 목록에서 항목 수를 검색합니다.
GetMruEntryAt GetMruEntryAt 메서드는 가장 최근에 사용한 목록의 지정된 인덱스에서 항목을 검색합니다.
GetOutputFilenameTemplate GetOutputFilenameTemplate 메서드는 파일 이름의 형식을 지정하는 서식 문자열(템플릿)을 검색합니다.
GetPhotoAcquireSettings GetPhotoAcquireSettings 메서드는 취득 설정 작업을 위한 IPhotoAcquireSettings 개체를 가져옵니다.
GetPrompt 메시지가 모달 대화 상자인 경우 GetPrompt 메서드는 프롬프트의 제목을 검색합니다.
Getproperty GetProperty 메서드는 항목의 속성 값을 검색합니다.
GetSequencePaddingWidth GetSequencePaddingWidth 메서드는 파일 이름의 순차 필드 너비를 나타내는 값을 검색합니다.

GetSequenceZeroPadding GetSequenceZeroPadding 메서드는 0 또는 공백을 사용하여 순차적 파일 이름을 채울지 여부를 나타내는 값을 검색합니다.
GetStream GetStream 메서드는 항목의 내용을 포함하는 읽기 전용 스트림을 검색합니다.
GetStringId GetStringId 메서드는 요청된 문자열의 할당되지 않은 정식 이름을 검색합니다. 예를 들어 태그 이름을 요청할 때 정식 이름은 "TagName"일 수 있습니다.
GetStringType GetStringType 메서드는 사용자로부터 가져올 문자열의 형식을 나타내는 값을 검색합니다.
GetSubItemAt GetSubItemAt 메서드는 하위 항목의 인덱스가 지정된 경우 항목의 하위 항목을 검색합니다.
GetSubItemCount GetSubItemCount 메서드는 항목에 포함된 하위 항목의 수를 검색합니다.
GetSubmitButtonText GetSubmitButtonText 메서드는 제출 단추에 대한 텍스트를 검색합니다.
GetThumbnail GetThumbnail 메서드는 항목에 제공된 썸네일을 검색합니다.
GetTooltipText GetTooltipText 메서드는 컨트롤에 대해 표시되는 도구 설명 텍스트를 검색합니다.
GetUserInput GetUserInput 메서드는 획득 중에 사용자에게 문자열 입력을 요청하는 메시지를 표시하는 기본 기능을 재정의합니다. 애플리케이션은 GetUserInput 메서드의 구현을 제공합니다.
GetUserInput 저장할 이미지의 태그 이름과 같이 사용자가 입력한 설명 정보를 검색합니다.

GetWindow GetWindow 메서드는 진행률 대화 상자에 대한 핸들을 검색합니다.
초기화 옵션 대화 상자를 초기화하고 레지스트리에서 저장된 옵션을 읽습니다.
초기화 Initialize 메서드는 플러그 인이 초기화되면 확장된 기능을 제공합니다. 애플리케이션은 Initialize 메서드의 구현을 제공합니다.
InitializeFromRegistry InitializeFromRegistry 메서드는 설정을 초기화할 레지스트리 키를 지정합니다.
InitializeItemList InitializeItemList 메서드는 디바이스에서 전송 가능한 항목을 열거하고 제공된 경우 각 항목을 선택적 진행률 콜백에 전달합니다.
IsCancelled IsCancelled 메서드는 진행률 대화 상자를 통해 작업이 취소되었는지 여부를 나타냅니다.
IsCheckboxChecked IsCheckboxChecked 메서드는 진행률 대화 상자의 확인란(일반적으로 전송 후 파일을 삭제할지 여부를 나타내는)이 선택되었는지 여부를 나타냅니다.
ProcessItem ProcessItem 메서드는 항목이 처리될 때마다 추가 기능을 제공합니다. 애플리케이션은 ProcessItem 메서드의 구현을 제공합니다.
SaveData SaveData 메서드는 대화 상자의 후속 인스턴스를 동일한 설정으로 초기화할 수 있도록 옵션 대화 상자의 취득 설정을 레지스트리에 저장합니다.
SetAcquisitionTime SetAcquisitionTime 메서드는 획득 시간을 명시적으로 설정합니다.
SetCaption 진행률 대화 상자의 캡션을 설정합니다.

SetCheckboxText SetCheckboxText 메서드는 전송 후 이미지를 삭제할지 여부를 나타내는 진행률 대화 상자의 확인란에 대한 텍스트를 설정합니다.
SetCheckboxTooltip SetCheckboxTooltip 메서드는 진행률 대화 상자의 확인란에 대한 도구 설명 텍스트를 설정합니다.
SetFlags SetFlags 메서드는 사진 획득 플래그를 설정합니다.
SetGroupTag SetGroupTag 메서드는 획득 세션에 대한 그룹 태그를 설정합니다.
SetImage 진행률 대화 상자에 표시되는 썸네일 이미지, 진행률 대화 상자의 제목 표시줄에 있는 아이콘 또는 ALT+TAB 키 조합 창의 아이콘을 설정합니다.
SetOutputFilenameTemplate SetOutputFilenameTemplate 메서드는 파일 이름의 형식을 지정하는 서식 문자열(템플릿)을 지정합니다.
SetPercentComplete SetPercentComplete 메서드는 현재 작업의 완료된 부분을 나타내는 값을 설정합니다.
SetProgressText SetProgressText 메서드는 진행률 대화 상자의 진행률 표시줄에 대한 텍스트를 설정합니다.
SetProperty SetProperty 메서드는 항목의 속성을 설정합니다.
SetSequencePaddingWidth SetSequencePaddingWidth 메서드는 파일 이름의 순차 필드 너비를 나타내는 값을 설정합니다.
SetSequenceZeroPadding SetSequenceZeroPadding 메서드는 0 또는 공백을 사용하여 순차적 파일 이름을 채우는지 여부를 나타내는 값을 설정합니다.

SetSubmitButtonText SetPrompt 메서드는 사용자에게 디바이스를 선택하라는 메시지를 표시하는 대화 상자에 표시되는 텍스트를 설정합니다.
SetTitle SetTitle 메서드는 디바이스 선택 대화 상자의 제목을 설정합니다.
SetTitle SetTitle 메서드는 진행률 대화 상자의 제목을 설정합니다.
ShowCheckbox ShowCheckbox 메서드는 전송 후 이미지를 삭제할지 여부를 나타내는 진행률 대화 상자에 확인란을 표시할지 여부를 나타냅니다.
StartDelete StartDelete 메서드는 디바이스에서 항목을 삭제하기 시작할 때 확장된 기능을 제공합니다.
StartEnumeration StartEnumeration 메서드는 가져올 항목의 열거가 시작될 때 확장된 기능을 제공합니다.
StartItemDelete StartItemDelete 메서드는 디바이스에서 개별 항목을 삭제할 때마다 확장된 기능을 제공합니다. 애플리케이션은 StartItemDelete 메서드의 구현을 제공합니다.
StartItemTransfer StartItemTransfer 메서드는 항목 전송이 시작될 때마다 확장된 기능을 제공합니다. 애플리케이션은 StartItemTransfer 메서드의 구현을 제공합니다.
StartTransfer StartTransfer 메서드는 디바이스에서 항목 전송이 시작될 때 추가 처리를 제공합니다. 애플리케이션은 StartTransfer 메서드의 구현을 제공합니다.
TransferComplete 전송 세션이 완료되면 확장된 기능을 제공합니다. 애플리케이션은 TransferComplete 메서드의 구현을 제공합니다.

UpdateDeletePercent UpdateDeletePercent 메서드는 삭제된 항목의 백분율이 변경되면 확장된 기능을 제공합니다. 애플리케이션은 UpdateDeletePercent 메서드의 구현을 제공합니다.
UpdateTransferPercent UpdateTransferPercent 메서드는 전송된 항목의 백분율이 변경될 때 확장된 기능을 제공합니다. 애플리케이션은 UpdateTransferPercent 메서드의 구현을 제공합니다.

인터페이스

IPhotoAcquire IPhotoAcquire 인터페이스는 디바이스에서 사진을 가져오는 방법을 제공합니다.
IPhotoAcquireDeviceSelectionDialog 이미지를 가져올 디바이스를 선택하는 대화 상자를 제공합니다.
IPhotoAcquireItem IPhotoAcquireItem 인터페이스는 디바이스에서 획득할 때 항목 작업을 위한 메서드를 제공합니다.
IPhotoAcquireOptionsDialog IPhotoAcquireOptionsDialog 인터페이스는 사용자가 파일 이름 형식과 같은 사진 취득 설정과 이미지를 회전할지, 태그 이름을 묻는 메시지를 표시할지, 가져온 후 카메라에서 사진을 지울지 여부를 선택할 수 있는 옵션 대화 상자를 표시하는 데 사용됩니다.
IPhotoAcquirePlugin 이미지 획득을 위해 Windows Vista UI(사용자 인터페이스)와 함께 실행할 플러그 인을 만들려는 경우 IPhotoAcquirePlugin 인터페이스를 구현합니다. 플러그 인을 사용하도록 설정하려면 레지스트리 설정이 필요합니다.
IPhotoAcquireProgressCB 획득 프로세스의 다양한 단계에서 추가 처리를 수행하려는 경우 IPhotoAcquireProgressCB 인터페이스를 구현할 수 있습니다.

IPhotoAcquireSettings IPhotoAcquireSettings 인터페이스는 파일 이름 형식과 같은 이미지 획득 설정을 사용하는 데 사용됩니다.
IPhotoAcquireSource IPhotoAcquireSource 인터페이스는 디바이스에서 항목을 획득하는 데 사용됩니다.
IPhotoProgressDialog 이미지를 열거하거나 가져올 때 표시될 수 있는 진행률 대화 상자를 제공합니다. 대화 상자는 모달이며 자체 스레드에서 실행됩니다.
IUserInputString IUserInputString 인터페이스는 태그 이름을 가져올 때와 같이 사용자에게 문자열을 요청할 때 생성되는 개체를 나타냅니다.

포인터 디바이스 입력 스택

아티클2023. 03. 09.

포인터 디바이스 입력 스택 기술의 개요입니다.

포인터 디바이스 입력 스택 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 포인터 디바이스 입력 스택

포인터 입력 메시지 및 알림

Windows 포인터 입력 메시지 및 알림 기능의 개요입니다.

포인터 입력 메시지 및 알림 기능은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 포인터 입력 메시지 및 알림

열거 목록

테이블 확장

LEGACY_TOUCHPAD_FEATURES 레거시 터치 패드에서 지원을 표시한 설정을 식별합니다.
POINTER_BUTTON_CHANGE_TYPE 포인터와 연결된 단추의 상태 변경을 식별합니다.
tagPOINTER_INPUT_TYPE 포인터 입력 형식을 식별합니다.
TOUCHPAD_SENSITIVITY_LEVEL 터치 패드 민감도 설정의 값을 식별합니다.

Functions

테이블 확장

EnableMouseInPointer 마우스가 포인터 입력 디바이스 역할을 하고 WM_POINTER 메시지를 보낼 수 있도록 합니다.
GET_POINTERID_WPARAM 지정된 값을 사용하여 포인터 ID를 검색합니다.

GetPointerCursorId 지정된 포인터와 연결된 커서 식별자를 검색합니다.
GetPointerFrameInfo 현재 메시지와 연결된 지정된 포인터에 대한 전체 정보 프레임을 가져옵니다.
GetPointerFrameInfoHistory 현재 메시지와 연결된 지정된 포인터에 대한 전체 정보 프레임(병합된 입력 프레임 포함)을 가져옵니다.
GetPointerFramePenInfo 현재 메시지와 연결된 지정된 포인터(PT_PEN 형식)에 대한 펜 기반 정보의 전체 프레임을 가져옵니다.
GetPointerFramePenInfoHistory 현재 메시지와 연결된 지정된 포인터(PT_PEN 형식)에 대한 펜 기반 정보(병합된 입력 프레임 포함)의 전체 프레임을 가져옵니다.
GetPointerFrameTouchInfo 현재 메시지와 연결된 지정된 포인터(PT_TOUCH 형식)에 대한 터치 기반 정보의 전체 프레임을 가져옵니다.
GetPointerFrameTouchInfoHistory 현재 메시지와 연결된 지정된 포인터(PT_TOUCH 형식)에 대한 터치 기반 정보(병합된 입력 프레임 포함)의 전체 프레임을 가져옵니다.
GetPointerInfo 현재 메시지와 연결된 지정된 포인터에 대한 정보를 가져옵니다.
GetPointerInfoHistory 지정된 포인터에 대한 현재 메시지로 병합된 개별 입력(있는 경우)과 연결된 정보를 가져옵니다.
GetPointerInputTransform 현재 메시지와 연결된 포인터 정보 좌표에 대한 하나 이상의 변환을 가져옵니다.
GetPointerPenInfo 현재 메시지와 연결된 지정된 포인터(PT_PEN 형식)에 대한 펜 기반 정보를 가져옵니다.
GetPointerPenInfoHistory

지정된 포인터(PT_PEN 형식)에 대한 현재 메시지로 병합된 개별 입력(있는 경우)과 연결된 펜 기반 정보를 가져옵니다.
GetPointerTouchInfo 현재 메시지와 연결된 지정된 포인터(PT_TOUCH 형식)에 대한 터치 기반 정보를 가져옵니다.
GetPointerTouchInfoHistory 지정된 포인터(PT_TOUCH 형식)의 현재 메시지에 병합된 개별 입력(있는 경우)과 연결된 터치 기반 정보를 가져옵니다.
GetPointerType 지정된 포인터의 포인터 형식을 검색합니다.
GetUnpredictedMessagePos 터치 예측 처리를 거치기 전에 포인터 데이터를 가져옵니다.
HAS_POINTER_CONFIDENCE_WPARAM 지정한 포인터 메시지가 실수로 인한 것이 아니라 의도적인 것으로 간주되는지 여부를 확인합니다.
IS_POINTER_CANCELED_WPARAM 지정한 포인터 입력이 갑자기 종료되었는지, 아니면 잘못된 것인지 확인하여 상호 작용이 완료되지 않았음을 나타냅니다.
IS_POINTER_FIFTHBUTTON_WPARAM 지정된 포인터가 다섯 번째 작업을 수행했는지 여부를 확인합니다.
IS_POINTER_FIRSTBUTTON_WPARAM 지정한 포인터가 첫 번째 작업을 수행했는지 여부를 확인합니다.
IS_POINTER_FLAG_SET_WPARAM 포인터 매크로가 지정된 플래그를 설정하는지 여부를 확인합니다.
IS_POINTER_FOURTHBUTTON_WPARAM 지정한 포인터가 네 번째 작업을 수행했는지 여부를 확인합니다.
IS_POINTER_INCONTACT_WPARAM 지정한 포인터가 연결되었는지 여부를 확인합니다.

IS_POINTER_INRANGE_WPARAM 지정한 포인터가 범위에 있는지 여부를 확인합니다.
IS_POINTER_NEW_WPARAM 지정한 포인터가 새 포인터인지 여부를 확인합니다.
IS_POINTER_PRIMARY_WPARAM 지정된 포인터가 기본 작업을 수행했는지 여부를 확인합니다.
IS_POINTER_SECONDBUTTON_WPARAM 지정된 포인터가 두 번째 작업을 수행했는지 여부를 확인합니다.
IS_POINTER_THIRDBUTTON_WPARAM 지정한 포인터가 세 번째 작업을 수행했는지 여부를 확인합니다.
IsMouseInPointerEnabled 마우스가 포인터 입력 장치 역할을 하고 WM_POINTER 메시지를 보낼 수 있도록 EnableMouseInPointer가 설정되어 있는지 여부를 나타냅니다.
SkipPointerFrameMessages 지정된 포인터에 대해 가장 최근에 검색된 메시지를 생성한 포인터 입력 프레임을 결정하고 동일한 포인터 입력 프레임에서 생성된 큐에 대기(시도되지 않은) 포인터 입력 메시지를 삭제합니다.

구조

테이블 확장

INPUT_TRANSFORM 메시지 소비자의 변환을 나타내는 행렬을 정의합니다.
POINTER_INFO 모든 포인터 형식에 공통적인 기본 포인터 정보를 포함합니다. 애플리케이션은 GetPointerInfo, GetPointerFrameInfo, GetPointerInfoHistory 및 GetPointerFrameInfoHistory 함수를 사용하여 이 정보를 검색할 수 있습니다.

POINTER_PEN_INFO 모든 포인터 형식에 공통적인 기본 펜 정보를 정의합니다.
POINTER_TOUCH_INFO 모든 포인터 형식에 공통적인 기본 터치 정보를 정의합니다.
TOUCHPAD_PARAMETERS_V1 검색된 모든 터치 패드와 관련된 사용자 터치 패드 설정 및 시스템 정보를 포함합니다.
TOUCHPREDICTIONPARAMETERS 터치 대상을 예측하고 거리 및 속도 데이터를 포함하는 터치 및 제스처 입력을 처리할 때 하드웨어 대기 시간을 보정하는 데 사용할 수 있는 하드웨어 입력 세부 정보를 포함합니다.

Last updated on 2026. 01. 27.

전력 계량 및 예산 참조

아티클2023. 03. 09.

파워 계량 및 예산 참조 기술 개요입니다.

Power Metering 및 예산 참조를 개발하려면 다음 헤더가 필요합니다.

- emi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 전력 계량 및 예산 참조

IOCTLs

IOCTL_EMI_GET_MEASUREMENT IOCTL_EMI_GET_MEASUREMENT 제어 코드는 현재 에너지 측정 및 측정을 수행한 시간을 검색합니다.
IOCTL_EMI_GET_METADATA IOCTL_EMI_GET_METADATA 제어 코드는 디바이스에서 EMI 메타데이터를 검색합니다.
IOCTL_EMI_GET_METADATA_SIZE IOCTL_EMI_GET_METADATA_SIZE 제어 코드는 IOCTL_EMI_GET_METADATA 요청을 실행하여 디바이스에서 가져올 수 있는 EMI 메타데이터 개체의 크기를 검색합니다.
IOCTL_EMI_GET_VERSION IOCTL_EMI_GET_VERSION 제어 코드는 디바이스에서 지원하는 현재 버전의 EMI 인터페이스를 검색합니다.

열거형

EMI_MEASUREMENT_UNIT EMI_MEASUREMENT_UNIT 열거형은 IOCTL_EMI_GET_MEASUREMENT 사용하여 디바이스에서 검색할 수 있는 사용 가능한 에너지 측정 단위를 나타냅니다.

구조체

EMI_CHANNEL_MEASUREMENT_DATA EMI_MEASUREMENT_DATA 구조체는 현재 에너지 측정 및 측정이 수행된 시간에 대한 데이터를 제공합니다.
EMI_CHANNEL_V2 EMI_CHANNEL_V2 구조는 EMI V2 채널에 대한 데이터를 제공합니다.
EMI_MEASUREMENT_DATA_V2 EMI_MEASUREMENT_DATA_V2 구조는 EMI_VERSION_V2 디바이스의 모든 채널에 대한 현재 에너지 측정 데이터에 대한 데이터를 제공합니다.
EMI_METADATA_SIZE EMI_METADATA_SIZE 구조는 IOCTL_EMI_GET_METADATA 요청을 실행하여 디바이스에서 가져올 수 있는 EMI(에너지 계량 인터페이스) 메타데이터 개체의 크기를 지정합니다.
EMI_METADATA_V1 EMI_METADATA 구조는 하드웨어 모델 및 하드웨어 수정 버전과 같은 EMI(에너지 계량 인터페이스) 인터페이스를 지원하는 디바이스에 대한 메타데이터를 제공합니다.
EMI_METADATA_V2 EMI_METADATA_V2 구조는 EMI_VERSION_V2 인터페이스를 지원하는 디바이스에 대한 메타데이터를 제공합니다. 이 메타데이터에는 하드웨어 디바이스 및 이 디바이스에서 노출하는 EMI 채널에 대한 정보가 포함됩니다.
EMI_VERSION EMI_VERSION 구조는 디바이스에서 지원하는 EMI(에너지 계량 인터페이스)의 버전을 설명합니다.

DDI 참조 인쇄

아티클2023. 08. 22.

인쇄 DDI 참조 기술 개요입니다.

인쇄 DDI 참조 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- DDI 참조 인쇄

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

프로세스 스냅샷

아티클2023. 08. 22.

프로세스 스냅샷 기술의 개요입니다.

프로세스 스냅샷을 개발하려면 다음 헤더가 필요합니다.

- processsnapshot.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 프로세스 스냅샷

열거형

PSS_CAPTURE_FLAGS PssCaptureSnapshot 캡처를 지정하는 플래그입니다.
PSS_DUPLICATE_FLAGS PssDuplicateSnapshot에서 사용할 중복 플래그입니다.
PSS_HANDLE_FLAGS PSS_HANDLE_ENTRY 구조체의 유효한 부분을 지정하는 플래그입니다.
PSS_OBJECT_TYPE PSS_HANDLE_ENTRY 구조체의 개체 형식을 지정합니다.
PSS_PROCESS_FLAGS 프로세스를 설명하는 플래그입니다.
PSS_QUERY_INFORMATION_CLASS PssQuerySnapshot 함수가 반환하는 정보를 지정합니다.
PSS_THREAD_FLAGS 스레드를 설명하는 플래그입니다.
PSS_WALK_INFORMATION_CLASS

PssWalkSnapshot 함수가 반환하는 정보를 지정합니다.

Functions

PssCaptureSnapshot 대상 프로세스의 스냅샷 캡처합니다.
PssDuplicateSnapshot 스냅샷 핸들을 한 프로세스에서 다른 프로세스로 복제합니다.
PssFreeSnapshot 스냅샷 해제합니다.
PssQuerySnapshot 스냅샷 쿼리합니다.
PssWalkMarker만들기 워크 마커를 만듭니다.
PssWalkMarkerFree PssWalkMarkerCreate에서 만든 워크 마커를 해제합니다.
PssWalkMarkerGetPosition 워크 마커의 현재 위치를 반환합니다.
PssWalkMarkerSeekToBeginning 걷기 마커를 다시 시작 부분으로 되돌아 줍니다.
PssWalkMarkerSetPosition 워크 마커의 위치를 설정합니다.
PssWalkSnapshot 현재 워크 위치에서 정보를 반환하고 다음 위치로 이동 마커를 진행합니다.

구조체

PSS_ALLOCATOR 프로세스 스냅샷 함수가 내부 워크 마커 구조를 할당하고 해제하는 데 사용하는 사용자 지정 함수를 지정합니다.
PSS_AUXILIARY_PAGE_ENTRY PssWalkSnapshot에서 반환된 보조 페이지 항목 정보를 보유합니다.
PSS_AUXILIARY_PAGES_INFORMATION PssQuerySnapshot에서 반환된 보조 페이지 정보를 보유합니다.
PSS_HANDLE_ENTRY PssWalkSnapshot에서 반환된 핸들에 대한 정보를 보유합니다.
PSS_HANDLE_INFORMATION PssQuerySnapshot에서 반환된 핸들 정보를 보유합니다.
PSS_HANDLE_TRACE_INFORMATION PssQuerySnapshot에서 반환된 핸들 추적 정보를 보유합니다.
PSS_PERFORMANCE_COUNTERS PssQuerySnapshot에서 반환된 성능 카운터를 보유합니다.
PSS_PROCESS_INFORMATION PssQuerySnapshot에서 반환된 프로세스 정보를 보유합니다.
PSS_THREAD_ENTRY PssWalkSnapshotPssWalkSnapshot에서 반환된 스레드 정보를 보유합니다.
PSS_THREAD_INFORMATION PssQuerySnapshot에서 반환된 스레드 정보를 보유합니다.
PSS_VA_CLONE_INFORMATION PssQuerySnapshot에서 반환된 VA(가상 주소) 복제 정보를 보유합니다.
PSS_VA_SPACE_ENTRY

PssWalkSnapshot이 VA(가상 주소) 지역에 대해 반환한 MEMORY_BASIC_INFORMATION 보유합니다.
PSS_VA_SPACE_INFORMATION PssQuerySnapshot에서 반환된 VA(가상 주소) 공간 정보를 보유합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

프로세스 상태 API(PSAPI)

아티클2024. 01. 11.

PSAPI(프로세스 상태 API) 기술 개요입니다.

PSAPI(프로세스 상태 API)를 개발하려면 다음 헤더가 필요합니다.

- psapi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 프로세스 상태 API(PSAPI)

Functions

테이블 확장

EmptyWorkingSet 지정된 프로세스의 작업 집합에서 가능한 한 많은 페이지를 제거합니다.
EnumDeviceDrivers 시스템의 각 디바이스 드라이버에 대한 부하 주소를 검색합니다.
EnumPageFilesA 시스템에 설치된 각 페이지 파일에 대한 콜백 루틴을 호출합니다. (ANSI)
EnumPageFilesW 시스템에 설치된 각 페이지 파일에 대한 콜백 루틴을 호출합니다. (유니코드)
EnumProcesses 시스템의 각 프로세스 개체에 대한 프로세스 식별자를 검색합니다.
EnumProcessModules 지정된 프로세스의 각 모듈에 대한 핸들을 검색합니다.
EnumProcessModulesEx 지정된 필터 조건을 충족하는 지정된 프로세스의 각 모듈에 대한 핸들을 검색합니다.

GetDeviceDriverBaseNameA 지정된 디바이스 드라이버의 기본 이름을 검색합니다. (ANSI)
GetDeviceDriverBaseNameW 지정된 디바이스 드라이버의 기본 이름을 검색합니다. (유니코드)
GetDeviceDriverFileNameA 지정된 디바이스 드라이버에 사용할 수 있는 경로를 검색합니다. (ANSI)
GetDeviceDriverFileNameW 지정된 디바이스 드라이버에 사용할 수 있는 경로를 검색합니다. (유니코드)
GetMappedFileNameA 지정된 주소가 지정된 프로세스의 주소 공간에 있는 메모리 매핑된 파일 내에 있는지 확인합니다. 이 경우 함수는 메모리 매핑 파일의 이름을 반환합니다. (ANSI)
GetMappedFileNameW 지정된 주소가 지정된 프로세스의 주소 공간에 있는 메모리 매핑된 파일 내에 있는지 확인합니다. 이 경우 함수는 메모리 매핑 파일의 이름을 반환합니다. (유니코드)
GetModuleBaseNameA 지정된 모듈의 기본 이름을 검색합니다. (ANSI)
GetModuleBaseNameW 지정된 모듈의 기본 이름을 검색합니다. (유니코드)
GetModuleFileNameExA 지정된 모듈을 포함하는 파일의 정규화된 경로를 검색합니다. (ANSI)
GetModuleFileNameExW 지정된 모듈을 포함하는 파일의 정규화된 경로를 검색합니다. (유니코드)
GetModuleInformation MODULEINFO 구조에서 지정된 모듈에 대한 정보를 검색합니다.
GetPerformanceInfo PERFORMANCE_INFORMATION 구조에 포함된 성능 값을 검색합니다.

GetProcessImageFileNameA 지정된 프로세스에 대한 실행 파일의 이름을 검색합니다. (ANSI)
GetProcessImageFileNameW 지정된 프로세스에 대한 실행 파일의 이름을 검색합니다. (유니코드)
GetProcessMemoryInfo 지정된 프로세스의 메모리 사용량에 대한 정보를 검색합니다.
GetWsChanges 이 함수 또는 InitializeProcessForWsWatch 함수가 마지막으로 호출된 이후 지정된 프로세스의 작업 집합에 추가된 페이지에 대한 정보를 검색합니다.
GetWsChangesEx 이 함수 또는 InitializeProcessForWsWatch 함수가 마지막으로 호출된 이후 지정된 프로세스의 작업 집합에 추가된 페이지에 대한 확장 정보를 검색합니다.
InitializeProcessForWsWatch 지정된 프로세스의 작업 집합에 대한 모니터링을 시작합니다.
PENUM_PAGE_FILE_CALLBACKA EnumPageFiles 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (ANSI)
PENUM_PAGE_FILE_CALLBACKW EnumPageFiles 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (유니코드)
QueryWorkingSet 지정된 프로세스의 작업 집합에 현재 추가된 페이지에 대한 정보를 검색합니다.
QueryWorkingSetEx 지정된 프로세스의 주소 공간에 있는 특정 가상 주소의 페이지에 대한 확장 정보를 검색합니다.

구조체

테이블 확장

ENUM_PAGE_FILE_INFORMATION 페이지 파일에 대한 정보를 포함합니다.
MODULEINFO 모듈 로드 주소, 크기 및 진입점을 포함합니다.
PERFORMANCE_INFORMATION 성능 정보를 포함합니다. (PERFORMANCE_INFORMATION)
PROCESS_MEMORY_COUNTERS 프로세스에 대한 메모리 통계를 포함합니다.
PROCESS_MEMORY_COUNTERS_EX 프로세스에 대한 확장 메모리 통계를 포함합니다.
PROCESS_MEMORY_COUNTERS_EX2 프로세스에 대한 확장 메모리 통계를 포함합니다. PROCESS_MEMORY_COUNTERS_EX 확장합니다.
PSAPI_WORKING_SET_BLOCK 페이지에 대한 작업 집합 정보를 포함합니다.
PSAPI_WORKING_SET_EX_BLOCK 페이지에 대한 확장 작업 집합 정보를 포함합니다.
PSAPI_WORKING_SET_EX_INFORMATION 프로세스에 대한 확장 작업 집합 정보를 포함합니다.
PSAPI_WORKING_SET_INFORMATION 프로세스에 대한 작업 집합 정보를 포함합니다.
PSAPI_WS_WATCH_INFORMATION 프로세스 작업 집합에 추가된 페이지에 대한 정보를 포함합니다.
PSAPI_WS_WATCH_INFORMATION_EX 프로세스 작업 집합에 추가된 페이지에 대한 확장된 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

프로세스 및 스레드

아티클2024. 08. 08.

새로 만든 프로세스 및 기본 스레드에 대한 정보를 제공합니다.

다음 헤더가 필요합니다.

- processthreadsapi.h

열거형

테이블 확장

MACHINE_ATTRIBUTES 호스트 운영 체제에서 코드 아키텍처를 실행할 수 있는 방법을 지정합니다. 둘 이상의 비트를 설정할 수 있습니다.
PROCESS_INFORMATION_CLASS 특정 프로세스 정보 클래스를 나타냅니다.
PROCESS_MEMORY_EXHAUSTION_TYPE 다양한 메모리 소모 유형을 나타냅니다.
QUEUE_USER_APC_FLAGS QUEUE_USER_APC_FLAGS 열거형(processthreadsapi.h)은 APC(사용자 모드 비동기 프로시저 호출) 개체에 대한 한정자 플래그를 지정합니다.
THREAD_INFORMATION_CLASS THREAD_INFORMATION_CLASS 열거형(processthreadsapi.h)은 지원되는 스레드 형식의 컬렉션을 지정합니다.

함수

테이블 확장

CreateProcessA 새 프로세스와 해당 기본 스레드를 만듭니다. 새 프로세스는 호출 프로세스의 보안 컨텍스트에서 실행됩니다. (ANSI)
CreateProcessAsUserA 새 프로세스와 해당 기본 스레드를 만듭니다. 새 프로세스는 지정된 토큰으로 표시되는 사용자의 보안 컨텍스트에서 실행됩니다. (ANSI)
CreateProcessAsUserW 새 프로세스와 해당 기본 스레드를 만듭니다. 새 프로세스는 지정된 토큰으로 표시되는 사용자의 보안 컨텍스트에서 실행됩니다. (유니코드)
CreateProcessW 새 프로세스와 해당 기본 스레드를 만듭니다. 새 프로세스는 호출 프로세스의 보안 컨텍스트에서 실행됩니다. (유니코드)
CreateRemoteThread 다른 프로세스의 가상 주소 공간에서 실행되는 스레드를 만듭니다.
CreateRemoteThreadEx 다른 프로세스의 가상 주소 공간에서 실행되는 스레드를 만들고 필요에 따라 프로세서 그룹 선호도와 같은 확장 특성을 지정합니다.
CreateThread 호출 프로세스의 가상 주소 공간 내에서 실행할 스레드를 만듭니다.
DeleteProcThreadAttributeList 프로세스 및 스레드 만들기에 대해 지정된 특성 목록을 삭제합니다.
ExitProcess 호출 프로세스와 모든 스레드를 종료합니다.
ExitThread 호출 스레드를 종료합니다.
FlushInstructionCache 지정된 프로세스에 대한 명령 캐시를 플러시합니다.

FlushProcessWriteBuffers 현재 프로세스의 스레드를 실행하는 각 프로세서의 쓰기 큐를 플러시합니다.
GetCurrentProcess 현재 프로세스에 대한 의사 핸들을 검색합니다.
GetCurrentProcessId 호출 프로세스의 프로세스 식별자를 검색합니다.
GetCurrentProcessorNumber 이 함수를 호출하는 동안 현재 스레드가 실행되고 있던 프로세서의 수를 검색합니다.
GetCurrentProcessorNumberEx 호출 스레드가 실행 중인 논리 프로세서의 프로세서 그룹 및 수를 검색합니다.
GetCurrentProcessToken 프로세스와 연결된 액세스 토큰을 참조하는 약식 방법으로 사용할 수 있는 의사 핸들을 검색합니다.
GetCurrentThread 호출 스레드에 대한 의사 핸들을 검색합니다.
GetCurrentThreadEffectiveToken 스레드에 현재 적용되는 토큰을 참조하는 약식 방법으로 사용할 수 있는 의사 핸들을 검색합니다. 이 토큰은 스레드 토큰이 있는 경우 스레드 토큰이고, 그렇지 않으면 프로세스 토큰입니다.
GetCurrentThreadId 호출 스레드의 스레드 식별자를 검색합니다.
GetCurrentThreadStackLimits 현재 스레드에 대해 시스템에서 할당한 스택의 경계를 검색합니다.
GetCurrentThreadToken 현재 스레드에 할당된 가장 토큰을 참조하는 약식 방법으로 사용할 수 있는 의사 핸들을 검색합니다.
GetExitCodeProcess

지정된 프로세스의 종료 상태를 검색합니다.
GetExitCodeThread 지정된 스레드의 종료 상태를 검색합니다.
getMachineTypeAttributes 지정된 아키텍처가 기본적으로 또는 호환성 또는 에뮬레이션 계층의 형태로 현재 시스템에서 지원되는지 쿼리합니다.
GetPriorityClass 지정된 프로세스의 우선 순위 클래스를 검색합니다. 이 값은 프로세스의 각 스레드의 우선 순위 값과 함께 각 스레드의 기본 우선 순위 수준을 결정합니다.
GetProcessDefaultCpuSetMasks SetProcessDefaultCpuSetMasks 또는 SetProcessDefaultCpuSets에서 설정한 프로세스 기본 집합의 CPU 집합 목록을 검색합니다.
GetProcessDefaultCpuSets SetProcessDefaultCpuSets에서 설정한 프로세스 기본 집합의 CPU 집합 목록을 검색합니다.
GetProcessHandleCount 지정된 프로세스에 속하는 열린 핸들의 수를 검색합니다.
GetProcessId 지정된 프로세스의 프로세스 식별자를 검색합니다.
GetProcessIdOfThread 지정된 스레드와 연결된 프로세스의 프로세스 식별자를 검색합니다.
GetProcessInformation 지정된 프로세스에 대한 정보를 검색합니다. (GetProcessInformation)
getProcessMitigationPolicy 호출 프로세스에 대한 완화 정책 설정을 검색합니다.
getProcessPriorityBoost 지정된 프로세스의 우선 순위 상승 제어 상태를 검색합니다.

GetProcessShutdownParameters 현재 호출하는 프로세스에 대한 종료 매개 변수를 검색합니다.
GetProcessTimes 지정된 프로세스에 대한 타이밍 정보를 검색합니다.
GetProcessVersion 지정된 프로세스가 실행될 것으로 예상되는 시스템의 주 버전 및 부 버전 번호를 검색합니다.
GetStartupInfoW 호출 프로세스를 만들 때 지정된 STARTUPINFO 구조체의 내용을 검색합니다.
getSystemCpuSetInformation 애플리케이션이 시스템에서 사용 가능한 CPU 집합과 해당 현재 상태를 쿼리할 수 있도록 허용합니다.
GetSystemTimes 시스템 타이밍 정보를 검색합니다. 다중 프로세서 시스템에서 반환되는 값은 모든 프로세서에서 지정된 시간의 합계입니다.
GetThreadContext 지정된 스레드의 컨텍스트를 검색합니다.
GetThreadDescription SetThreadDescription을 호출하여 스레드에 할당된 설명을 검색합니다.
GetThreadId 지정된 스레드의 스레드 식별자를 검색합니다.
GetThreadIdealProcessorEx 지정된 스레드에 적합한 프로세서의 프로세서 번호를 검색합니다.
GetThreadInformation 지정된 스레드에 대한 정보를 검색합니다. (GetThreadInformation)
getThreadIOPendingFlag 지정된 스레드에 보류 중인 I/O 요청이 있는지 여부를 확인합니다.

GetThreadPriority 지정된 스레드의 우선 순위 값을 검색합니다. 이 값은 스레드 프로세스의 우선 순위 클래스와 함께 스레드의 기본 우선 순위 수준을 결정합니다.
getThreadPriorityBoost 지정된 스레드의 우선 순위 상승 제어 상태를 검색합니다.
GetThreadSelectedCpuSetMasks SetThreadSelectedCpuSetMasks 또는 SetThreadSelectedCpuSets를 사용하여 할당이 설정된 경우 지정된 스레드의 명시적 CPU 집합 할당을 반환합니다.
GetThreadSelectedCpuSets SetThreadSelectedCpuSets API를 사용하여 할당이 설정된 경우 지정된 스레드의 명시적 CPU 집합 할당을 반환합니다.
GetThreadTimes 지정된 스레드에 대한 타이밍 정보를 검색합니다.
InitializeProcThreadAttributeList 프로세스 및 스레드 생성을 위해 지정된 특성 목록을 초기화합니다.
isProcessCritical 지정된 프로세스가 중요한 프로세스로 간주되는지 여부를 결정합니다.
IsProcessorFeaturePresent 지정된 프로세서 기능이 현재 컴퓨터에서 지원되는지 여부를 확인합니다.
OpenProcess 기존 로컬 프로세스 개체를 엽니다.
OpenProcessToken 프로세스와 연결된 액세스 토큰을 엽니다.
OpenThread 기존 스레드 개체를 엽니다.
OpenThreadToken

스레드와 연결된 액세스 토큰을 엽니다.
ProcessIdToSessionId 지정된 프로세스와 연결된 원격 데스크톱 서비스 세션을 검색합니다.
QueryProcessAffinityUpdateMode 지정된 프로세스의 선호도 업데이트 모드를 검색합니다.
QueryProtectedPolicy 보호된 정책과 연결된 값을 쿼리합니다.
queueUserAPC 지정된 스레드의 APC 큐에 사용자 모드 APC(비동기 프로시저 호출) 개체를 추가합니다. (QueueUserAPC)
queueUserAPC2 지정된 스레드의 APC 큐에 사용자 모드 APC(비동기 프로시저 호출) 개체를 추가합니다. (QueueUserAPC2)
ResumeThread 스레드의 일시 중단 횟수를 줄입니다. 일시 중단 횟수가 0으로 감소하면 스레드 실행이 다시 시작됩니다.
SetPriorityClass 지정된 프로세스의 우선 순위 클래스를 설정합니다. 이 값은 프로세스의 각 스레드의 우선 순위 값과 함께 각 스레드의 기본 우선 순위 수준을 결정합니다.
SetProcessAffinityUpdateMode 지정된 프로세스의 선호도 업데이트 모드를 설정합니다.
SetProcessDefaultCpuSetMasks SetProcessDefaultCpuSetMasks 함수(processthreadsapi.h)는 지정된 프로세스의 스레드에 대한 기본 CPU 집합 할당을 설정합니다.
SetProcessDefaultCpuSets SetProcessDefaultCpuSets 함수(processthreadsapi.h)는 지정된 프로세스의 스레드에 대한 기본 CPU 집합 할당을 설정합니다.
setProcessDynamicEnforcedCetCompatibleRanges

지정된 프로세스에 대해 동적으로 적용된 CETCOMPAT 범위를 설정합니다.
SetProcessInformation 지정된 프로세스에 대한 정보를 설정합니다.
SetProcessMitigationPolicy 호출 프로세스에 대한 완화 정책을 설정합니다. 완화 정책을 사용하면 프로세스가 다양한 유형의 공격에 대해 자신을 강화할 수 있습니다.
setProcessPriorityBoost 지정된 프로세스의 스레드 우선 순위를 일시적으로 높일 수 있는 시스템 기능을 사용하지 않도록 설정하거나 활성화합니다.
SetProcessShutdownParameters 현재 호출 프로세스에 대한 종료 매개 변수를 설정합니다. 이 함수는 시스템의 다른 프로세스에 상대적인 프로세스에 대한 종료 순서를 설정합니다.
SetProtectedPolicy 보호된 정책을 설정합니다.
SetThreadContext 지정된 스레드의 컨텍스트를 설정합니다.
SetThreadDescription 스레드에 설명을 할당합니다.
setThreadIdealProcessor 스레드에 대한 기본 프로세서를 설정합니다. 시스템은 가능하면 원하는 프로세서에서 스레드를 예약합니다.
SetThreadIdealProcessorEx 지정된 스레드에 이상적인 프로세서를 설정하고 선택적으로 이전의 이상적인 프로세서를 검색합니다.
SetThreadInformation 지정된 스레드에 대한 정보를 설정합니다.
setThreadPriority

지정된 스레드의 우선 순위 값을 설정합니다. 이 값은 스레드 프로세스의 우선 순위 클래스와 함께 스레드의 기본 우선 순위 수준을 결정합니다.
SetThreadPriorityBoost 스레드의 우선 순위를 일시적으로 높일 수 있는 시스템 기능을 사용하지 않도록 설정하거나 활성화합니다.
setThreadSelectedCpuSetMasks 지정된 스레드에 대해 선택한 CPU 집합 할당을 설정합니다. 이 할당은 설정된 경우 프로세스 기본 할당을 재정의합니다. (SetThreadSelectedCpuSetMasks)
setThreadSelectedCpuSets 지정된 스레드에 대해 선택한 CPU 집합 할당을 설정합니다. 이 할당은 설정된 경우 프로세스 기본 할당을 재정의합니다. (SetThreadSelectedCpuSets)
SetThreadStackGuarantee 스택 오버플로 예외 중에 사용할 수 있는 호출 스레드 또는 파이버와 연결된 스택의 최소 크기를 설정합니다.
SuspendThread 지정된 스레드를 일시 중단합니다.
SwitchToThread 호출 스레드가 현재 프로세서에서 실행할 준비가 된 다른 스레드로 실행을 생성하도록 합니다. 운영 체제는 실행할 다음 스레드를 선택합니다.
TerminateProcess 지정된 프로세스와 모든 해당 스레드를 종료합니다.
TerminateThread 스레드를 종료합니다.
TlsAlloc 스레드 TLS(로컬 스토리지) 인덱스를 할당합니다. 각 스레드가 인덱스에 대한 자체 슬롯을 받기 때문에 프로세스의 모든 스레드는 이후에 이 인덱스를 사용하여 스레드에 로컬인 값을 저장하고 검색할 수 있습니다.
TlsFree 스레드 TLS(로컬 스토리지) 인덱스를 해제하여 다시 사용할 수 있도록 합니다.

TlsGetValue 지정된 TLS 인덱스에 대한 호출 스레드의 TLS(스레드 로컬 스토리지) 슬롯에서 값을 검색합니다. 프로세스의 각 스레드에는 각 TLS 인덱스의 자체 슬롯이 있습니다.
TlsGetValue2 지정된 TLS 인덱스에 대한 호출 스레드의 TLS(스레드 로컬 스토리지) 슬롯에서 값을 검색합니다. 프로세스의 각 스레드에는 각 TLS 인덱스의 자체 슬롯이 있습니다.
TlsSetValue 지정된 TLS 인덱스의 호출 스레드 TLS(로컬 스토리지) 슬롯에 값을 저장합니다. 프로세스의 각 스레드에는 각 TLS 인덱스의 자체 슬롯이 있습니다.
UpdateProcThreadAttribute 프로세스 및 스레드 만들기에 대한 특성 목록에서 지정된 특성을 업데이트합니다.

구조

테이블 확장

APP_MEMORY_INFORMATION 단일 시점의 앱 메모리 사용량을 나타냅니다. 이 구조체는 PROCESS_INFORMATION_CLASS 클래스에서 사용됩니다.
MEMORY_PRIORITY_INFORMATION 스레드 또는 프로세스의 메모리 우선 순위를 지정합니다.
OVERRIDE_PREFETCH_PARAMETER
PROCESS_INFORMATION 새로 만든 프로세스 및 기본 스레드에 대한 정보를 포함합니다. CreateProcess, CreateProcessAsUser, CreateProcessWithLogonW 또는 CreateProcessWithTokenW 함수와 함께 사용됩니다.
PROCESS_LEAP_SECOND_INFO 시스템에서 양수 윤초를 처리하는 방법을 지정합니다.

PROCESS_MEMORY_EXHAUSTION_INFO 할당이 메모리를 커밋하지 못하는 경우 애플리케이션에서 프로세스를 종료하도록 구성할 수 있습니다. 이 구조체는 PROCESS_INFORMATION_CLASS 클래스에서 사용됩니다.
PROCESS_POWER_THROTTLING_STATE 제한 정책과 해당 프로세스에 전원 관리가 적용될 때 대상 프로세스에 적용하는 방법을 지정합니다.
PROCESS_PROTECTION_LEVEL_INFORMATION PPL(Protected Process Light)을 사용할지 여부를 지정합니다.
STARTUPINFOA 생성 시 프로세스에 대한 주 창의 창 스테이션, 데스크톱, 표준 핸들 및 모양을 지정합니다. (ANSI)
STARTUPINFOW 생성 시 프로세스에 대한 주 창의 창 스테이션, 데스크톱, 표준 핸들 및 모양을 지정합니다. (유니코드)
THREAD_POWER_THROTTLING_STATE 해당 스레드에 전원 관리가 적용될 때 제한 정책 및 대상 스레드에 적용하는 방법을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

프로젝션된 파일 시스템

아티클2023. 08. 22.

프로젝션된 파일 시스템 기술 개요입니다.

프로젝션된 파일 시스템을 개발하려면 다음 헤더가 필요합니다.

- projectedfslib.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 프로젝션된 파일 시스템

열거형

PRJ_CALLBACK_DATA_FLAGS 열거형에서 반환되는 내용을 제어하는 플래그입니다.
PRJ_COMPLETE_COMMAND_TYPE 명령 형식을 지정합니다.
PRJ_EXT_INFO_TYPE 확장 정보 유형을 지정합니다.
PRJ_FILE_STATE 항목의 상태입니다.
PRJ_NOTIFICATION 콜백에서 알림을 보낼 때 지정된 알림 값입니다.
PRJ_NOTIFY_TYPES 파일 또는 폴더의 변경 사항을 설명하는 알림 유형입니다.
PRJ_PLACEHOLDER_ID 자리 표시자 식별자의 길이를 정의합니다.
PRJ_STARTVIRTUALIZING_FLAGS

가상화 instance 시작할 때 제공할 플래그입니다.
PRJ_UPDATE_FAILURE_CAUSES 업데이트가 실패한 이유에 대한 설명입니다.
PRJ_UPDATE_TYPES 디스크의 파일 또는 디렉터리 상태를 고려하여 업데이트를 허용할지 여부를 지정하는 플래그입니다.

Functions

PRJ_CANCEL_COMMAND_CB 이전 콜백 호출에 의한 작업을 취소해야 한다는 것을 공급자에게 알릴 수 있습니다.
PRJ_END_DIRECTORY_ENUMERATION_CB 디렉터리 열거형이 끝났다는 것을 공급자에게 알릴 수 있습니다.
PRJ_GET_DIRECTORY_ENUMERATION_CB 공급자로부터 디렉터리 열거 정보를 요청합니다.
PRJ_GET_FILE_DATA_CB 파일의 기본 데이터 스트림 콘텐츠를 요청합니다.
PRJ_GET_PLACEHOLDER_INFO_CB 공급자로부터 파일 또는 디렉터리에 대한 정보를 요청합니다.
PRJ_NOTIFICATION_CB 파일 시스템 작업에 대한 알림을 공급자에게 전달합니다.
PRJ_QUERY_FILE_NAME_CB 지정된 파일 경로가 공급자의 지원 저장소에 있는지 여부를 확인합니다.
PRJ_START_DIRECTORY_ENUMERATION_CB 디렉터리 열거가 시작 중임을 공급자에게 알릴 수 있습니다.

PrjAllocateAlignedBuffer 가상화 instance 스토리지 디바이스의 메모리 맞춤 요구 사항을 충족하는 버퍼를 할당합니다.
PrjClearNegativePathCache 활성 상태인 경우 가상화 instance 음수 경로 캐시를 제거합니다.
PrjCompleteCommand 공급자가 이전에 HRESULT_FROM_WIN32(ERROR_IO_PENDING)을 반환한 콜백 처리를 완료했음을 나타냅니다.
PrjDeleteFile 공급자가 로컬 파일 시스템에 캐시된 항목을 삭제할 수 있도록 합니다.
PrjDoesNameContainWildCards 이름에 와일드카드 문자가 포함되어 있는지 여부를 확인합니다.
PrjFileNameCompare 두 파일 이름을 비교하고 상대 데이터 정렬 순서를 나타내는 값을 반환합니다.
PrjFileNameMatch 파일 이름이 검색 패턴과 일치하는지 여부를 결정합니다.
PrjFillDirEntryBuffer 열거형에 하나의 파일 또는 디렉터리에 대한 정보를 제공합니다.
PrjFillDirEntryBuffer2 하나의 파일 또는 디렉터리에 대한 정보를 열거형에 제공하고 호출자가 확장된 정보를 지정할 수 있도록 합니다.
PrjFreeAlignedBuffer 할당된 버퍼를 해제합니다.
PrjGetOnDiskFileState 파일 또는 디렉터리에 대한 디스크 내 파일 상태를 가져옵니다.
PrjGetVirtualizationInstanceInfo 가상화 instance 대한 정보를 검색합니다.

PrjMarkDirectoryAsPlaceholder 기존 디렉터리를 디렉터리 자리 표시자로 변환합니다.
PrjStartVirtualizing ProjFS 가상화 instance 구성하고 시작하여 서비스 I/O에 사용할 수 있도록 하고 공급자에서 콜백을 호출합니다.
PrjStopVirtualizing 실행 중인 ProjFS 가상화 instance 중지하여 I/O를 서비스하거나 공급자에서 콜백을 포함할 수 없게 합니다.
PrjUpdateFileIfNeeded 공급자가 로컬 파일 시스템에 캐시된 항목을 업데이트할 수 있도록 합니다.
PrjWriteFileData PrjWriteFileData 함수는 PRJ_GET_FILE_DATA_CB 콜백 호출에서 요청된 데이터를 제공합니다. (PrjWriteFileData)
PrjWritePlaceholderInfo 파일 또는 디렉터리 메타데이터를 ProjFS로 보냅니다.
PrjWritePlaceholderInfo2 파일 또는 디렉터리 메타데이터를 ProjFS로 보내고 호출자가 확장 정보를 지정할 수 있도록 합니다.

구조체

PRJ_CALLBACK_DATA 모든 작업 콜백에 대해 공급자에게 전달되는 표준 정보를 정의합니다.
PRJ_CALLBACKS 공급자가 콜백의 구현을 저장하는 콜백 루틴 집합입니다.
PRJ_COMPLETE_COMMAND_EXTENDED_PARAMETERS 특정 콜백을 완료하는 데 필요한 매개 변수를 지정합니다.

PRJ_EXTENDED_INFO 디렉터리 열거형 및 자리 표시자 정보에 대한 선택적 확장 정보를 지정합니다.
PRJ_FILE_BASIC_INFO 항목에 대한 기본 정보입니다.
PRJ_NOTIFICATION_MAPPING 디렉터리("알림 루트"라고 함)와 비트 마스크로 표현된 알림 집합 간의 페어링인 알림 매핑에 대해 설명합니다.
PRJ_NOTIFICATION_PARAMETERS 알림에 대한 추가 매개 변수입니다.
PRJ_PLACEHOLDER_INFO 자리 표시자 파일 또는 디렉터리에 대한 메타데이터 버퍼입니다.
PRJ_PLACEHOLDER_VERSION_INFO 자리 표시자 파일의 내용을 고유하게 식별하는 정보입니다.
PRJ_STARTVIRTUALIZING_OPTIONS 가상화 instance 시작할 때 제공하는 옵션입니다.
PRJ_VIRTUALIZATION_INSTANCE_INFO 가상화 instance 대한 정보입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

서비스 품질(QOS)

아티클2023. 03. 09.

QOS(서비스 품질) 기술에 대한 개요입니다.

QOS(서비스 품질)를 개발하려면 다음 헤더가 필요합니다.

- infocard.h
- lpmapi.h
- ntddpsch.h
- qos.h
- qos2.h
- qosobjs.h
- qospol.h
- qossp.h
- traffic.h
- winsock2.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 서비스 품질(QOS)

열거형

FilterType FilterType 열거형은 RSVP FILTERSPEC에 사용되는 필터의 형식을 지정합니다.
QOS_FLOWRATE_REASON QOS_FLOWRATE_REASON 열거형은 흐름의 대역폭이 변경된 이유를 나타냅니다.
QOS_NOTIFY_FLOW QOS_NOTIFY_FLOW 열거형은 QOSNotifyFlow 함수가 알림을 보내기 위해 존재해야 하는 상황을 지정합니다.
QOS_QUERY_FLOW QOS_QUERY_FLOW 열거형은 QOSQueryFlow 함수가 요청할 정보의 형식을 나타냅니다.

QOS_SET_FLOW QOS_SET_FLOW 열거형은 흐름에 대해 변경되는 내용을 나타냅니다.
QOS_SHAPING QOS_SHAPING 열거형은 흐름의 셰이핑 동작을 정의합니다.
QOS_TRAFFIC_TYPE QOS_TRAFFIC_TYPE 열거형은 다양한 트래픽 유형을 정의합니다. 각 흐름에는 단일 트래픽 유형이 있습니다. 이렇게 하면 QOS 하위 시스템이 각 형식에 사용자 지정 정책을 적용할 수 있습니다.

Functions

CBADMITRESULT cbAdmitResult 함수는 LPM에서 LPM_AdmitRsvpMsg 요청에 대한 결과를 반환하는 데 사용됩니다.
CBGETRSVPOBJECTS cbGetRsvpObjects 함수는 LM이 LPM_GetRsvpObjects 요청에 대한 결과를 비동기적으로 반환하는 콜백 함수입니다.
LPM_AdmitRsvpMsg LPM_AdmitRsvpMsg 함수는 정책 기반 허용 제어 결정을 위해 RSVP 메시지를 LPM에 전달하기 위해 PCM에서 호출됩니다.
LPM_CommitResv LPM_CommitResv 함수는 LPM에서 예약 약정 결정을 얻기 위해 PCM에 의해 호출됩니다.
LPM_Deinitialize LPM_Deinitialize 함수를 사용하면 시스템 종료 또는 DSBM(지정된 서브넷 대역폭 관리자) 상태 변경으로 인해 PCM이 LPM의 초기화를 해제하도록 지시할 수 있습니다.
LPM_DeleteState LPM_DeleteState 함수는 PCM에서 LPM의 RSVP 상태 정보를 삭제하기 위해 호출됩니다.
LPM_GetRsvpObjects LPM_GetRsvpObjects 함수를 사용하면 PCM에서 정책 데이터에 대한 LPM을 쿼리할 수 있습니다.

LPM_Initialize LPM_Initialize 함수는 LPM(로컬 정책 모듈)을 초기화합니다.
LPM_IpAddressTable LPM_IpAddressTable 함수는 PCM에서 LPM이 초기화되는 Windows 2000 Server에 할당된 IP 주소 목록을 전달하는 데 사용됩니다.
PALLOCMEM PALLOCMEM 함수는 PCM에서 제공하는 메모리 할당 함수로, 정책 정보를 PCM에 반환할 때 메모리를 할당하는 데 사용됩니다.
PFREEMEM PFREEMEM 함수는 PCM에서 제공하는 메모리 해제 함수입니다.
QOSAddSocketToFlow 트래픽에 대한 새 흐름을 추가합니다.
QOSCancel QOSSetFlow와 같이 보류 중인 겹치는 작업을 취소합니다.
QOSCloseHandle QOSCloseHandle 함수는 QOSCreateHandle 함수에서 반환된 핸들을 닫습니다.
QOSCreateHandle 이 함수는 QOS 하위 시스템 및 QOSHandle 매개 변수를 초기화합니다. QOSHandle 매개 변수는 다른 QOS 함수를 호출할 때 사용됩니다. 다른 함수 앞에 QOSCreateHandle을 호출해야 합니다.
QOSEnumerateFlows 모든 기존 흐름을 열거합니다.
QOSNotifyFlow 알림을 받을 호출 애플리케이션을 등록합니다.
QOSQueryFlow 특정 흐름에 대한 정보를 요청합니다.

QOSRemoveSocketFromFlow 이전에 추가된 흐름이 종료되었음을 QOS 하위 시스템에 알합니다.
QOSSetFlow 애플리케이션에서 애플리케이션의 패킷의 우선 순위를 지정하고 흐름 트래픽을 변경하도록 QOS 하위 시스템을 요청하기 위해 호출됩니다.
QOSStartTrackingClient QOSStartTrackingClient 함수는 QOS 하위 시스템에 새 클라이언트가 있는지 알 수 있습니다.
QOSStopTrackingClient QOSStopTrackingClient 함수는 이전에 QOSStartTrackingClient 함수를 사용한 클라이언트 추적을 중지하도록 QoS 하위 시스템에 알린 것입니다. 흐름이 현재 진행 중인 경우 이 함수는 흐름에 영향을 미치지 않습니다.
TcAddFilter TcAddFilter 함수는 필터와 일치하는 패킷을 연결된 흐름으로 보낼 수 있도록 기존 흐름과 새 필터를 연결합니다.
TcAddFlow TcAddFlow 함수는 지정된 인터페이스에 새 흐름을 추가합니다.
TcCloseInterface TcCloseInterface 함수는 TcOpenInterface 호출을 사용하여 이전에 연 인터페이스를 닫습니다. TcCloseInterface 호출을 사용하여 인터페이스를 닫기 전에 특정 인터페이스의 모든 흐름과 필터를 닫아야 합니다.
TcDeleteFilter TcDeleteFilter 함수는 이전에 TcAddFilter 함수와 함께 추가된 필터를 삭제합니다.
TcDeleteFlow TcDeleteFlow 함수는 TcAddFlow 함수와 함께 추가된 흐름을 삭제합니다. 클라이언트는 흐름을 삭제하기 전에 흐름과 연결된 모든 필터를 삭제해야 합니다. 그렇지 않으면 오류가 반환되고 함수는 흐름을 삭제하지 않습니다.
TcDeregisterClient TcDeregisterClient 함수는 TCI(트래픽 제어 인터페이스)를 사용하여 클라이언트를 등록 취소합니다.

TcEnumerateFlows TcEnumerateFlows 함수는 설치된 흐름과 연결된 필터를 인터페이스에 열거합니다.
TcEnumerateInterfaces TcEnumerateInterfaces 함수는 모든 트래픽 제어 사용 네트워크 인터페이스를 열거합니다. 클라이언트는 ClNotifyHandler 함수를 통해 인터페이스 변경에 대한 알림을 받습니다.
TcGetFlowNameA TcGetFlowName 함수는 호출 클라이언트에서 만든 흐름의 이름을 제공합니다. (ANSI)
TcGetFlowNameW TcGetFlowName 함수는 호출 클라이언트에서 만든 흐름의 이름을 제공합니다. (유니코드)
TCI_ADD_FLOW_COMPLETE_HANDLER ClAddFlowComplete 함수는 트래픽 제어에서 TcAddFlow 함수에 대한 이전 호출의 완료를 클라이언트에 알리는 데 사용됩니다.
TCI_DEL_FLOW_COMPLETE_HANDLER ClDeleteFlowComplete 함수는 트래픽 제어에서 TcDeleteFlow 함수에 대한 이전 호출의 완료를 클라이언트에 알리는 데 사용됩니다.
TCI_MOD_FLOW_COMPLETE_HANDLER ClModifyFlowComplete 함수는 트래픽 제어에서 TcModifyFlow 함수에 대한 이전 호출이 완료되었음을 클라이언트에 알리는 데 사용됩니다.
TCI_NOTIFY_HANDLER ClNotifyHandler 함수는 트래픽 제어에서 흐름 삭제, 필터 매개 변수 변경 또는 인터페이스 닫기를 비롯한 다양한 트래픽 제어 특정 이벤트를 클라이언트에 알리는 데 사용됩니다.
TcModifyFlow TcModifyFlow 함수는 기존 흐름을 수정합니다. TcModifyFlow를 호출할 때 새 Flowspec 매개 변수 및 모든 트래픽 제어 개체를 채워야 합니다.
TcOpenInterfaceA TcOpenInterface 함수는 인터페이스를 엽니다. (ANSI)
TcOpenInterfaceW TcOpenInterface 함수는 인터페이스를 엽니다. (유니코드)

TcQueryFlowA TcQueryFlow 함수는 흐름 이름에 따라 특정 흐름 매개 변수의 값에 대한 트래픽 제어를 쿼리합니다. 흐름의 이름은 TcEnumerateFlows 함수 또는 TcGetFlowName 함수에서 검색할 수 있습니다. (ANSI)
TcQueryFlowW TcQueryFlow 함수는 흐름 이름에 따라 특정 흐름 매개 변수의 값에 대한 트래픽 제어를 쿼리합니다. 흐름의 이름은 TcEnumerateFlows 함수 또는 TcGetFlowName 함수에서 검색할 수 있습니다. (유니코드)
TcQueryInterface TcQueryInterface 함수는 관련 인터페이스별 매개 변수에 대한 트래픽 제어를 쿼리합니다.
TcRegisterClient TcRegisterClient 함수는 TCI(트래픽 제어 인터페이스)에 클라이언트를 등록하는 데 사용됩니다. TcRegisterClient 함수는 클라이언트가 TCI에 대해 호출하는 첫 번째 함수 호출이어야 합니다.
TcSetFlowA TcSetFlow 함수는 지정된 흐름에 대한 개별 매개 변수를 설정합니다. (ANSI)
TcSetFlowW TcSetFlow 함수는 지정된 흐름에 대한 개별 매개 변수를 설정합니다. (유니코드)
TcSetInterface TcSetInterface 함수는 지정된 인터페이스에 대한 개별 매개 변수를 설정합니다.

구조체

AD_GENERAL_PARAMS AD_GENERAL_PARAMS 구조체에는 RSVP Adspec 개체에 포함된 일반 문자화 매개 변수가 포함되어 있습니다.
ADDRESS_LIST_DESCRIPTOR ADDRESS_LIST_DESCRIPTOR 구조는 지정된 인터페이스에 대한 네트워크 주소 설명자 정보를 제공합니다.

ADSPEC ADSPEC 구조에는 RSVP에 대한 Adspec 메시지 정보가 포함됩니다.
CONTROL_SERVICE CONTROL_SERVICE 구조에는 지원되는 RSVP 서비스 유형이 포함되어 있습니다.
CtrlLoadFlowspec CtrlLoadFlowspec 구조체에는 제어된 Load FLOWSPEC가 포함됩니다.
ENUMERATION_BUFFER ENUMERATION_BUFFER 구조에는 흐름 이름, 흐름과 연결된 필터 수 및 흐름과 연결된 필터 배열을 포함하여 지정된 흐름과 관련된 정보가 포함됩니다.
ERROR_SPEC ERROR_SPEC 구조에는 RSVP 오류 메시지가 포함됩니다.
Error_Spec_IPv4 Error_Spec_IPv4 구조는 RSVP 전송에 대한 오류 코드 정보를 저장합니다.
FILTER_SPEC FILTER_SPEC 구조체는 RSVP FILTERSPEC에 대한 정보를 저장합니다.
Filter_Spec_IPv4 Filter_Spec_IPv4 구조에는 IPv4 FILTERSPEC에 대한 정보가 포함되어 있습니다.
Filter_Spec_IPv4GPI Filter_Spec_IPv4GPI 구조에는 IPv4 FILTERSPEC에 대한 일반화된 포트 ID 정보가 포함되어 있습니다.
FLOW_DESC FLOW_DESC 구조에는 RSVP에 대한 흐름 설명자 정보가 포함되어 있습니다.
FLOWDESCRIPTOR FLOWDESCRIPTOR 구조체는 지정된 FLOWSPEC에 대해 하나 이상의 필터를 지정합니다.
FLOWSPEC FLOWSPEC 구조는 RSVP SP에 서비스 품질 매개 변수를 제공합니다.

Gads_parms_t Gads_parms_t 구조는 보장된 서비스 Adspec 매개 변수를 저장합니다.
GenAdspecParams GenAdspecParams 구조체에는 일반 경로 특성화 매개 변수가 포함됩니다.
GenTspec GenTspec 구조체는 제네릭 Tspec 정보를 저장합니다.
GenTspecParms GenTspecParms 구조체는 제네릭 Tspec 매개 변수를 저장합니다.
GuarFlowSpec GuarFlowSpec 구조에는 보장된 flowspec 정보가 포함되어 있습니다.
GuarRspec GuarRspec 구조에는 보장된 Rspec 정보가 포함됩니다.
ID_ERROR_OBJECT ID_ERROR_OBJECT 구조에는 RSVP에 대한 ID 정책 요소에 대한 오류 메시지 정보가 포함되어 있습니다.
IDPE_ATTR IDPE_ATTR 구조에는 ID 정책 요소 특성 정보가 포함됩니다.
IN_ADDR_IPV4 IN_ADDR_IPV4 공용 구조체는 RSVP FILTERSPEC와 함께 사용할 IPv4 주소를 저장합니다.
IN_ADDR_IPV6 IN_ADDR_IPV6 구조체는 RSVP FILTERSPEC와 함께 사용할 IPv6 주소를 저장합니다.
IntServFlowSpec IntServFlowSpec 구조에는 통합 서비스 흐름 지정에 대한 정보가 포함되어 있습니다.
IntServMainHdr IntServMainHdr 구조체는 Integrated Services RSVP 개체의 헤더입니다.

IntServParmHdr IntServParmHdr 구조체는 통합 서비스 매개 변수의 헤더입니다.
IntServiceHdr IntServiceHdr 구조체는 Integrated Services 서비스 개체의 헤더입니다.
IntServTspecBody IntServTspecBody 구조체에는 RSVP Tspec에 대한 정보가 포함되어 있습니다.
IP_PATTERN IP_PATTERN 구조는 IP 프로토콜에 특정 패턴 또는 해당 마스크를 적용합니다. IP_PATTERN 구조 지정은 패킷 필터 애플리케이션의 트래픽 제어 인터페이스에서 사용됩니다.
IPX_PATTERN IPX_PATTERN 구조는 IPX 프로토콜에 대한 특정 패턴 또는 해당 마스크를 적용합니다. IPX_PATTERN 구조 지정은 패킷 필터 애플리케이션의 트래픽 제어 인터페이스에서 사용됩니다.
IS_ADSPEC_BODY IS_ADSPEC_BODY 구조에는 Integrated Services Adspec 정보가 포함됩니다.
IS_FLOWSPEC IS_FLOWSPEC 구조체는 Integrated Services FLOWSPEC 개체를 저장합니다.
LPM_INIT_INFO LPM_INIT_INFO 구조에는 로컬 정책 모듈 초기화 정보가 포함됩니다.
LPMIPTABLE LPMIPTABLE 구조에는 각 인터페이스에 대한 SNMP 인덱스, IP 주소 및 서브넷 마스크를 포함한 IP 정보가 포함됩니다. LPMIPTABLE 구조체는 Lpm_IpAddressTable 함수에 대한 인수로 제공됩니다.
PARAM_BUFFER PARAM_BUFFER 구조체는 CONTROL_SERVICE 구조에 포함될 수 있는 매개 변수 버퍼의 형식을 설명합니다.
POLICY_DATA POLICY_DATA 구조에는 RSVP 메시지에 대한 정책 데이터가 포함됩니다.

POLICY_DECISION POLICY_DECISION 구조에는 RSVP 정책 결정 정보가 포함됩니다.
POLICY_ELEMENT POLICY_ELEMENT(lpmapi.h) 구조체에는 RSVP 정책 요소가 포함됩니다.
POLICY_ELEMENT POLICY_ELEMENT(infocard.h) 구조체에는 RSVP 정책 요소가 포함되어 있습니다. (POLICY_ELEMENT)
PS_ADAPTER_STATS PS_ADAPTER_STATS 구조는 지정된 어댑터에 대한 통계 패킷 쉐이퍼 정보를 제공합니다. PS_ADAPTER_STATS 구조체는 PS_COMPONENT_STATS 구조체와 함께 사용됩니다.
PS_COMPONENT_STATS PS_COMPONENT_STATS 구조를 사용하면 애플리케이션이 TC 사용 흐름에 대한 통계 정보를 가져올 수 있습니다.
PS_CONFORMER_STATS PS_CONFORMER_STATS 구조는 특정 흐름에 대한 통계 패킷 셰이퍼 정보를 제공합니다. PS_CONFORMER_STATS 구조체는 PS_COMPONENT_STATS 구조체와 함께 사용됩니다.
PS_DRRSEQ_STATS PS_DRRSEQ_STATS 구조는 NIC(네트워크 인터페이스 카드) 및 패킷 시퀀서 패킷 셰이퍼 통계를 제공합니다. PS_DRRSEQ_STATS 구조체는 PS_COMPONENT_STATS 구조체와 함께 사용됩니다.
PS_FLOW_STATS PS_FLOW_STATS 구조는 특정 흐름에 대한 통계 패킷 셰이퍼 정보를 제공합니다. PS_FLOW_STATS 구조체는 PS_COMPONENT_STATS 구조체와 함께 사용됩니다.
PS_SHAPER_STATS PS_SHAPER_STATS 구조는 컴퓨터의 패킷 쉐이퍼 구성 요소에 대한 통계 패킷 쉐이퍼 정보를 제공합니다. PS_SHAPER_STATS 구조체는 PS_COMPONENT_STATS 구조체와 함께 사용됩니다.
Qos QOS 구조는 QOS 지원 애플리케이션이 특정 흐름에서 전송 및 수신된 트래픽에 대한 서비스 매개 변수의 품질을 지정할 수 있는 방법을 제공합니다.

QOS_DESTADDR QOS_DESTADDR QOS 개체는 송신 소켓에 대한 연결 함수 호출을 방지하기 위해 WSAIoctl(SIO_SET_QOS) 함수를 호출하는 동안 사용됩니다.
QOS_DIFFSERV QOS_DIFFSERV 트래픽 제어 개체는 차별화된 서비스 모드에서 작동할 때 패킷 스케줄러에 대한 필터를 지정하는 데 사용됩니다.
QOS_DIFFSERV_RULE QOS_DIFFSERV_RULE 구조는 지정된 흐름에 대한 Diffserv 규칙을 제공하기 위해 트래픽 제어 개체 QOS_DIFFSERV 함께 사용됩니다.
QOS_DS_CLASS QOS_DS_CLASS 트래픽 제어 개체를 사용하면 애플리케이션 개발자가 지정된 흐름과 연결된 IP 패킷에 대한 기본 DSCP(Diffserv 코드 포인트) 값을 재정의할 수 있습니다. 기본적으로 DSCP 값은 흐름의 ServiceType에서 파생됩니다.
QOS_FLOW_FUNDAMENTALS QOS_FLOW_FUNDAMENTALS 구조에는 흐름에 대한 기본 정보가 포함되어 있습니다.
QOS_FLOWRATE_OUTGOING QOS_FLOWRATE_OUTGOING 구조는 QOSSetFlow 함수에서 흐름 속도 정보를 설정하는 데 사용됩니다.
QOS_FRIENDLY_NAME QOS_FRIENDLY_NAME 트래픽 제어 개체는 친숙한 이름을 흐름과 연결합니다.
QOS_OBJECT_HDR QOS_OBJECT_HDR QOS 개체는 각 QOS 개체에 연결됩니다. 개체 형식과 길이를 지정합니다.
QOS_PACKET_PRIORITY 흐름 트래픽의 우선 순위를 나타내는 QOS_PACKET_PRIORITY 구조체입니다.
QOS_SD_MODE QOS 개체 QOS_SD_MODE 트래픽 제어 패킷 셰이퍼 구성 요소의 동작을 정의합니다.

QOS_SHAPING_RATE QOS 개체 QOS_SHAPING_RATE 지정된 흐름에 균일한 트래픽 셰이핑 속도를 적용하도록 지정합니다.
QOS_TCP_TRAFFIC QOS_TCP_TRAFFIC 구조체는 지정된 흐름에 대한 IP 우선 순위 및 UserPriority 매핑을 TCP 트래픽의 시스템 기본값으로 설정해야 함을 나타내는 데 사용됩니다.
QOS_TRAFFIC_CLASS QOS_TRAFFIC_CLASS 트래픽 제어 개체는 지정된 흐름의 트래픽을 분류하는 패킷에 지정된 기본 UserPriority 값을 재정의하는 데 사용됩니다.
QOS_VERSION QOS_VERSION 구조는 QOS 프로토콜의 버전을 나타냅니다.
QualAppFlowSpec QualAppFlowSpec 구조에는 정성적 애플리케이션에 대한 FLOWSPEC 정보가 포함되어 있습니다.
QualTspec QualTspec 구조에는 정성적 Tspec 정보가 포함됩니다.
QualTspecParms QualTspecParms 구조에는 정성적 Tspec 매개 변수가 포함됩니다.
RESV_STYLE RESV_STYLE 구조체에는 RSVP RESV 스타일에 대한 정보가 포함되어 있습니다.
RSVP_ADSPEC RSVP_ADSPEC QOS 개체는 RSVP 기능 및 사용 가능한 서비스와 관련된 발신자와 수신자 간의 데이터 경로를 따라 네트워크 디바이스를 설명하는 정보를 제공하거나 검색하는 방법을 제공합니다.
RSVP_FILTERSPEC RSVP_FILTERSPEC 구조는 RSVP FILTERSPEC 정보를 제공합니다.
RSVP_FILTERSPEC_V4 RSVP_FILTERSPEC_V4 구조체는 IPv4 주소에 FILTERSPEC에 대한 정보를 저장합니다.

RSVP_FILTERSPEC_V4_GPI RSVP_FILTERSPEC_V4_GPI 구조는 지정된 FILTERSPEC에 대한 일반 포트 식별자 정보를 제공합니다.
RSVP_FILTERSPEC_V6 RSVP_FILTERSPEC_V6 구조체는 IPv6 주소에 FILTERSPEC에 대한 정보를 저장합니다.
RSVP_FILTERSPEC_V6_FLOW RSVP_FILTERSPEC_V6_FLOW 구조체는 IPv6 FILTERSPEC에 대한 흐름 레이블 정보를 제공합니다.
RSVP_FILTERSPEC_V6_GPI RSVP_FILTERSPEC_V6_GPI 구조체는 IPv6 주소에서 지정된 FILTERSPEC에 대한 일반 포트 식별자 정보를 제공합니다.
RSVP_HOP RSVP_HOP 구조에는 RSVP 사용 홉에 대한 정보가 포함되어 있습니다.
Rsvp_Hop_IPv4 Rsvp_Hop_IPv4 구조체는 RSVP 사용 IPv4 홉에 대한 정보를 저장합니다.
RSVP_MSG_OBJS RSVP_MSG_OBJS 구조체에는 RSVP 메시지 개체가 포함됩니다.
RSVP_POLICY RSVP_POLICY 구조는 하나 이상의 정의되지 않은 정책 요소를 저장합니다.
RSVP_POLICY_INFO RSVP_POLICY_INFO 구조는 RSVP에서 검색된 정의되지 않은 정책 요소를 저장합니다.
RSVP_RESERVE_INFO ProviderSpecific 버퍼를 통해 RSVP_RESERVE_INFO QOS 개체를 사용하면 지정된 흐름에 대한 RSVP 동작을 세분화된 수준에서 지정하거나 수정할 수 있으며 흐름에 대한 기본 RSVP 스타일 설정을 재정의할 수 있습니다.
RSVP_SCOPE RSVP_SCOPE 구조는 RSVP 범위 정보를 제공합니다.

RSVP_SESSION RSVP_SESSION 구조체는 RSVP SESSION 메시지에 대한 정보를 저장합니다.
RSVP_STATUS_INFO QOS 개체 RSVP_STATUS_INFO 모니터링 FD_QOS 이벤트와 관련된 이벤트 알림과 오류 정보를 포함하여 지정된 흐름의 RSVP 상태에 대한 정보를 제공합니다.
RsvpObjHdr RsvpObjHdr 구조체는 RSVP 메시지에 대한 개체 헤더를 제공합니다.
Scope_list_ipv4 Scope_list_ipv4 구조체에는 RSVP SCOPE 개체 정보가 포함됩니다.
SENDER_TSPEC SENDER_TSPEC 구조에는 RSVP 보낸 사람 Tspec에 대한 정보가 포함되어 있습니다.
Session_IPv4 Session_IPv4 구조체는 IPv4 RSVP SESSION 개체에 대한 정보를 저장합니다.
TC_GEN_FILTER TC_GEN_FILTER 구조는 특정 패킷 특성 또는 조건 집합과 일치하는 필터를 만듭니다. 이 필터는 이후에 특성 조건을 충족하는 패킷을 특정 흐름과 연결하는 데 사용할 수 있습니다.
TC_GEN_FLOW TC_GEN_FLOW 구조는 트래픽 제어 인터페이스와 함께 사용할 제네릭 흐름을 만듭니다. 흐름은 이 구조체의 멤버를 통해 사용자 지정됩니다.
TC_IFC_DESCRIPTOR TC_IFC_DESCRIPTOR 구조체는 인터페이스를 열거하는 데 사용되는 인터페이스 식별자입니다.
TCI_CLIENT_FUNC_LIST TCI_CLIENT_FUNC_LIST 구조는 트래픽 제어 인터페이스에서 클라이언트 콜백 함수를 등록하고 액세스하는 데 사용됩니다. TCI_CLIENT_FUNC_LIST 각 멤버는 클라이언트 제공 콜백 함수에 대한 포인터입니다.

방사형 컨트롤러 입력

아티클2023. 10. 05.

방사형 컨트롤러 입력 기술에 대한 개요입니다.

방사형 컨트롤러 입력을 개발하려면 다음 헤더가 필요합니다.

- radialcontrollerinterop.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 방사형 컨트롤러 입력

피드백

이 페이지가 도움이 되었나요? Yes No

원격 지원

아티클2023. 08. 22.

원격 지원 기술 개요입니다.

원격 지원을 개발하려면 다음 헤더가 필요합니다.

- rendezvoussession.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 원격 지원

열거형

RENDEZVOUS_SESSION_FLAGS 세션 초대에 사용할 수 있는 플래그 목록을 제공합니다.
RENDEZVOUS_SESSION_STATE 세션 초대의 가능한 상태 코드 목록을 제공합니다.

Functions

get_Flags 세션 정보를 나타내는 값을 검색합니다. 예를 들어 세션 플래그는 사용자가 초대자인지 초대 대상인지 여부를 나타낼 수 있습니다.
get_RemoteUser Windows Messenger 연락처 이름이 포함된 BSTR에 대한 포인터를 검색합니다.
get_State 세션 상태를 나타내는 값을 검색합니다.
SendContextData 컨텍스트 데이터를 원격 RendezvousApplication으로 보냅니다.

SetRendezvousSession IRendezvousSession을 Windows 원격 지원 애플리케이션에 전달합니다. 이 메서드는 인스턴트 메시징 애플리케이션에서 사용됩니다.
Terminate 원격 RendezvousApplication을 종료합니다.

인터페이스

DRendezvousSessionEvents IRendezvousSession 개체에서 생성되거나 수신되는 이벤트를 처리합니다.
IRendezvousApplication 메신저 대화(메신저) 애플리케이션에서 원격 지원 세션을 만드는 데 사용하는 메서드를 노출합니다.
IRendezvousSession 세션에 대한 데이터를 보내고 종료할 수 있는 메서드를 노출합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

원격 데스크톱 서비스

아티클2023. 03. 09.

원격 데스크톱 서비스 기술 개요입니다.

원격 데스크톱 서비스를 개발하려면 다음 헤더가 필요합니다.

- audioapotypes.h
- cchannel.h
- pchannel.h
- rdpappcontainerclient.h
- sbtsv.h
- sessdirpublictypes.h
- tsgauthenticationengine.h
- tsgpolicyengine.h
- tspubplugin2com.h
- tspubplugincom.h
- tssbx.h
- tsuserex.h
- tsvirtualchannels.h
- workspaceax.h
- workspaceruntime.h
- workspaceruntimeclientext.h
- wrdsgraphicschannels.h
- wtsapi32.h
- wtsdefs.h
- wtshintapi.h
- wtsprotocol.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 원격 데스크톱 서비스

열거형

AAAccountingDataType ITSGAccountingEngine::D oAccounting 메서드가 알림을 받는 이벤트의 형식을 지정합니다.

AAAuthSchemes RD 게이트웨이(원격 데스크톱 게이트웨이)에 연결하는 데 사용되는 인증 유형을 지정합니다.
AE_POSITION_FLAGS AE_CURRENT_POSITION 구조체에 대한 상수 정의 이러한 상수는 현재 위치의 유효성 정도를 설명합니다.
PolicyAttributeType 연결과 연결된 리디렉션 설정을 지정합니다.
RDV_TASK_STATUS IRDVTaskPluginNotifySink::OnTaskStateChange 메서드와 함께 작업 상태를 나타내는 데 사용됩니다.
RemoteActionType 원격 세션으로 보낼 작업입니다.
SnapshotEncodingType RDP(원격 데스크톱 프로토콜) 앱 컨테이너 클라이언트 스냅샷에 사용되는 인코딩 유형입니다.
SnapshotFormatType RDP(원격 데스크톱 프로토콜) 앱 컨테이너 클라이언트 스냅샷에 사용되는 데이터 형식입니다.
TARGET_CHANGE_TYPE 대상에서 발생한 변경 유형을 지정합니다.
TARGET_STATE 대상의 상태를 나타냅니다.
TARGET_TYPE 대상이 풀 또는 팜에 속하는지 여부를 나타냅니다.
TS_SB_SORT_BY 정렬 순서를 지정합니다. EnumerateTargets 메서드에서 매개 변수로 사용됩니다.
TSPUB_PLUGIN_PD_ASSIGNMENT_TYPE 개인 데스크톱 해상도에 대한 할당 유형을 지정합니다.

TSPUB_PLUGIN_PD_RESOLUTION_TYPE 요청되는 개인 데스크톱 해상도의 유형을 지정합니다.
TSSD_AddrV46Type IP 주소 유형을 나타냅니다.
TSSESSION_STATE 세션의 상태를 나타냅니다.
WRDS_CONNECTION_SETTING_LEVEL WRDS_CONNECTION_SETTINGS 구조체의 WRdsConnectionSetting 멤버에 포함된 구조체의 형식을 지정합니다.
WRDS_LISTENER_SETTING_LEVEL WRDS_LISTENER_SETTINGS 구조체의 WRdsListenerSetting 멤버에 포함된 구조체의 형식을 지정하는 데 사용됩니다.
WRDS_SETTING_LEVEL WRDS_SETTINGS 구조체의 WRdsSetting 멤버에 포함된 구조체의 형식을 지정합니다.
WRDS_SETTING_STATUS WRDS_SETTINGS_1 구조체의 다양한 멤버에 대한 정책 설정의 상태를 지정합니다.
WRDS_SETTING_TYPE WRDS_SETTINGS 구조에 저장되는 설정의 범주를 지정합니다.
WRdsGraphicsChannelType IWRdsGraphicsChannelManager::CreateChannel 메서드에서 만들 그래픽 가상 채널의 형식을 지정하는 데 사용됩니다.
WTS_CERT_TYPE 라이선스를 획득하는 데 사용되는 인증서 유형을 지정하는 값을 포함합니다.
WTS_CONFIG_CLASS WTSQueryUserConfig 및 WTSSetUserConfig 함수에 대한 호출에서 설정하거나 검색할 사용자 구성 정보의 형식을 나타내는 값을 포함합니다.

WTS_CONFIG_SOURCE WTSQueryUserConfig 함수에서 반환되는 구성 정보의 원본을 지정합니다.
WTS_CONNECTSTATE_CLASS 원격 데스크톱 서비스 세션의 연결 상태를 지정합니다.
WTS_INFO_CLASS WTSQuerySessionInformation 함수 호출에서 검색할 세션 정보의 유형을 나타내는 값을 포함합니다.
WTS_LOGON_ERROR_REDIRECTOR_RESPONSE 로그온 오류에 대한 프로토콜의 기본 응답을 지정하는 값을 포함합니다.
WTS_RCM_DRAIN_STATE RD 세션 호스트(원격 데스크톱 세션 호스트) 서버의 드레이닝 상태에 대한 정보를 포함합니다.
WTS_RCM_SERVICE_STATE 원격 데스크톱 서비스 서비스의 상태에 대한 정보를 포함합니다.
WTS_TYPE_CLASS 원격 데스크톱 서비스 함수가 버퍼에서 반환한 구조체의 형식을 지정합니다.
WTS_VIRTUAL_CLASS 검색할 가상 채널 정보의 유형을 나타내는 값을 포함합니다.
WTSSBX_ADDRESS_FAMILY 리디렉션에 사용되는 네트워크 주소의 주소 패밀리를 나타내는 값을 포함합니다.
WTSSBX_MACHINE_DRAIN RD 세션 호스트(원격 데스크톱 세션 호스트) 서버의 드레이닝 상태를 나타내는 값을 포함합니다.
WTSSBX_MACHINE_SESSION_MODE RD 세션 호스트(원격 데스크톱 세션 호스트) 서버의 세션 모드를 나타내는 값을 포함합니다.
WTSSBX_MACHINE_STATE 서버의 현재 상태를 나타내는 값을 포함합니다.

WTSSBX_NOTIFICATION_TYPE RD 세션 호스트(원격 데스크톱 세션 호스트) 서버 또는 사용자 세션에서 발생한 상태 변경 유형을 나타내는 값을 포함합니다.
WTSSBX_SESSION_STATE 사용자 세션의 연결 상태를 나타내는 값을 포함합니다.

함수

AcceptConnection 연결 요청을 계속하도록 프로토콜을 지시합니다.
AcceptConnection IWTSProtocolConnection::AcceptConnection을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnection::AcceptConnection을 사용합니다.
AcquireTargetLock 대상을 잠가.
AddEnvironmentToStore 리소스 플러그 인 저장소에 환경을 추가합니다.
Addresource RemoteApp 및 데스크톱 연결의 연결에 리소스를 추가합니다. (IWorkspaceRegistration2.AddResource)
AddResourceEx RemoteApp 및 데스크톱 연결의 연결에 리소스를 추가합니다. (IWorkspaceRegistration2::AddResourceEx)
AddResourceType RemoteApp 및 데스크톱 연결 런타임에 타사 파일 이름 확장명을 등록합니다.
AddSessionToStore 리소스 플러그 인 저장소에 새 세션을 추가합니다.

AddTargetToStore 리소스 플러그 인 저장소에 대상을 추가합니다.
ApplySettings 지정된 내용을 RDP 파일에 저장합니다.
attachEvent 이벤트에 이벤트 처리기를 연결합니다.
AuthenticateClientToSession 연결을 다시 연결해야 하는 세션을 지정합니다.
AuthenticateClientToSession IWTSProtocolConnection::AuthenticateClientToSession을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnection::AuthenticateClientToSession을 사용합니다.
AuthenticateUser 사용자를 인증합니다.
AuthorizeConnection 지정된 연결이 RD 게이트웨이(원격 데스크톱 게이트웨이)에 연결할 수 있는 권한이 있는지 여부를 확인합니다.
AuthorizeResource 지정된 연결에 연결할 권한이 있는 리소스를 결정합니다.
BrokenConnection 클라이언트 연결이 끊어졌는지 원격 데스크톱 서비스 서비스에 알릴 수 있습니다.
BrokenConnection IWTSProtocolConnectionCallback::BrokenConnection을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnectionCallback::BrokenConnection을 사용합니다.
CancelAuthentication 기존 인증 요청을 취소합니다.

CancelJob 프로비전 작업을 취소합니다.
CHANNEL_INIT_EVENT_FN 원격 데스크톱 서비스에서 클라이언트 DLL에 가상 채널 이벤트를 알리기 위해 호출하는 애플리케이션 정의 콜백 함수입니다.
CHANNEL_OPEN_EVENT_FN 원격 데스크톱 서비스에서 특정 가상 채널에 대한 이벤트를 클라이언트 DLL에 알리기 위해 호출하는 애플리케이션 정의 콜백 함수입니다.
ClearWorkspaceCredential 지정된 연결 ID와 연결된 사용자 자격 증명을 삭제합니다.
닫기 채널을 닫습니다. (IWTSVirtualChannel.Close)
닫기 채널을 닫기 위해 호출됩니다.
닫기 세션 연결이 끊긴 후 연결을 닫습니다.
닫기 IWTSProtocolConnection::Close를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnection::Close를 사용합니다.
연결 RDP(원격 데스크톱 프로토콜) 앱 컨테이너 클라이언트 컨트롤에 현재 설정된 속성을 사용하여 연결을 시작합니다.
연결됨 RDC(원격 데스크톱 연결) 클라이언트가 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에 성공적으로 연결되었음을 플러그 인에 알합니다.
ConnectNotify 세션이 초기화되었음을 프로토콜에 알리는 신호입니다.

ConnectNotify IWTSProtocolConnection::ConnectNotify를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnection::ConnectNotify를 사용합니다.
CreateChannel 그래픽 가상 채널을 만드는 데 사용됩니다.
CreateEnvironmentObject ITsSbEnvironment 환경 개체를 만듭니다.
CreateEnvironmentPropertySetObject ITsSbEnvironmentPropertySet 환경 속성 집합 개체를 만듭니다.
CreateListener 특정 엔드포인트에서 수신 대기하는 수신기 개체의 인스턴스를 반환합니다.
CreateListener 들어오는 클라이언트 연결 요청을 수신 대기하는 IWRdsProtocolListener 개체의 생성을 요청합니다.
CreateListener IWTSProtocolManager::CreateListener를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolManager::CreateListener를 사용합니다.
CreateLoadBalanceResultObject ITsSbLoadBalanceResult 부하 분산 결과 개체를 만듭니다.
CreatePluginPropertySet ITsSbPluginPropertySet 플러그 인 속성 집합 개체를 만듭니다.
CreateSessionObject 플러그 인은 CreateSessionObject 메서드를 사용하여 ITsSbSession 세션 개체를 만들 수 있습니다.
CreateTargetObject ITsSbTarget 대상 개체를 만듭니다.

CreateTargetPropertySetObject ITsSbTargetPropertySet 대상 속성 집합 개체를 만듭니다.
CreateVirtualChannel 프로토콜이 가상 채널을 만들 것을 요청합니다.
CreateVirtualChannel IWTSProtocolConnection::CreateVirtualChannel을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnection::CreateVirtualChannel을 사용합니다.
CreateVirtualMachines 가상 머신을 비동기적으로 만듭니다.
DeletePersonalDesktopAssignment 개인 가상 데스크톱 컬렉션에서 지정된 사용자와 가상 머신 간의 매핑을 삭제하기 위해 호출됩니다.
DeleteProperties 속성을 삭제합니다.
DeleteResourceType RemoteApp 및 데스크톱 연결 런타임을 사용하여 타사 파일 이름 확장명을 등록 취소합니다.
DeleteSavedCredentials 지정된 원격 컴퓨터에 대해 저장된 자격 증명을 삭제합니다.
DeleteTarget 대상을 삭제합니다.
DeleteVirtualMachines 가상 머신을 비동기적으로 삭제합니다.
detachEvent 이벤트에서 이벤트 처리기를 분리합니다.
연결 끊기 활성 연결을 끊습니다.

연결 끊김 RDC(원격 데스크톱 연결) 클라이언트가 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 연결이 끊어졌습니다.
DisconnectNotify 세션의 연결이 끊어졌는지 프로토콜에 알합니다.
DisconnectNotify IWTSProtocolConnection::D isconnectNotify를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnection::D isconnectNotify를 사용합니다.
DisconnectUser 원격 데스크톱 게이트웨이(RD 게이트웨이)에 클라이언트의 연결을 끊어야 한다는 것을 알 수 있습니다.
DisconnectWorkspace 지정된 연결 ID와 연결된 모든 기존 연결의 연결을 끊습니다.
DisconnectWorkspaceByFriendlyName 지정된 이름을 가진 연결과 연결된 모든 기존 연결의 연결을 끊습니다.
DisplayIOCtl IWTSProtocolConnectionCallback::D isplayIOCtl을 더 이상 사용할 수 없습니다.
DoAccounting 연결에 대한 세션을 만들거나 닫는 방법에 대한 정보를 제공합니다.
DoTarget 프로토콜이 섀도 연결의 대상 쪽을 시작하게 요청합니다.
DoTarget IWTSProtocolShadowConnection::D oTarget을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolShadowConnection::D oTarget을 사용합니다.
EnableWddmIdd Termsrv는 이 메서드를 사용하여 작동 중인 모드를 프로토콜 스택에 알릴 수 있습니다.

EnumerateEnvironments 리소스 플러그 인 저장소에 있는 환경을 포함하는 배열을 반환합니다.
EnumerateEnvironmentsByProvider 지정된 공급자에 있는 환경을 포함하는 배열을 반환합니다.
EnumerateFarms 지정된 리소스 플러그 인에 의해 추가된 모든 팜을 열거합니다.
EnumerateFarms 리소스 플러그 인 저장소에 추가된 모든 팜을 열거합니다.
EnumerateProperties 속성 집합을 열거합니다.
EnumerateSessions 지정된 공급자의 세션이 포함된 배열을 반환합니다.
EnumerateSessions 지정된 세션 집합을 열거합니다.
EnumerateTargets 전역 저장소에 있는 지정된 대상이 포함된 배열을 반환합니다.
EnumerateTargets 리소스 플러그 인 저장소에 있는 지정된 대상이 포함된 배열을 반환합니다.
ExecuteRemoteAction 원격 세션에서 작업을 수행하도록 합니다.
get_Actions RDP(원격 데스크톱 프로토콜) 앱 컨테이너 클라이언트에 대한 작업 개체를 검색합니다.
get_AllowLogon 사용자가 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에 로그온할 수 있는지 여부를 지정하는 값입니다. (Get)

get_BrokenConnectionAction 원격 데스크톱 서비스 세션 제한에 도달할 때 수행할 작업을 지정하는 값입니다. (Get)
get_ClientConnectionPropertySet 클라이언트 연결과 연결된 속성을 포함하는 개체를 검색합니다.
get_ClientDisplay 클라이언트 컴퓨터의 표시 디바이스에 대한 정보를 검색하거나 지정합니다. (Get)
get_ConnectClientDrivesAtLogon 원격 데스크톱 서비스 세션이 시작될 때 매핑된 클라이언트 드라이브를 다시 연결해야 하는지를 지정하는 값입니다. (Get)
get_ConnectClientPrintersAtLogon 로그온 시 매핑된 클라이언트 프린터에 다시 연결할지 여부를 지정하는 값입니다. (Get)
get_ConnectionError 클라이언트 연결이 처리되는 동안 발생한 오류를 나타내는 값을 검색합니다.
get_Context 작업과 연결된 컨텍스트 바이트를 검색합니다.
get_CreateTime 세션을 만든 시간을 검색하거나 지정합니다. (Get)
get_Deadline 작업을 시작해야 하는 시간을 검색합니다. 패치의 우선 순위를 지정하는 데 사용됩니다. 가장 빠른 최종 기한이 있는 패치가 먼저 시작됩니다.
get_DefaultToMainPrinter 클라이언트의 기본 프린터에 자동으로 인쇄할지 여부를 지정하는 값입니다. (Get)
get_DisconnectTime 세션 연결이 끊긴 시간을 검색하거나 지정합니다. (Get)
get_Domain RDC(원격 데스크톱 연결) 클라이언트의 도메인 이름을 나타내는 값을 검색합니다.

get_Domain 사용자의 도메인 이름을 검색합니다.
get_Enabled RDP 앱 컨테이너 클라이언트 컨트롤에서 터치 포인터 기능을 사용할 수 있는지 여부입니다. (Get)
get_EnableRemoteControl 사용자의 원격 데스크톱 서비스 세션에 대한 원격 관찰 또는 원격 제어를 허용할지 여부를 지정하는 값입니다. (Get)
get_EndTime 작업 에이전트가 작업을 시작할 수 있는 최신 시간을 검색합니다.
get_Environment 대상 컴퓨터를 호스트하는 환경에 대한 정보가 포함된 개체를 검색합니다.
get_EnvironmentName 대상과 연결된 환경의 이름을 검색하거나 지정합니다. (ITsSbTargetEx.get_EnvironmentName)
get_EnvironmentPropertySet 대상 컴퓨터를 호스트하는 환경에 대한 속성 집합을 검색하거나 지정합니다. (Get)
get_EventsEnabled RDP 앱 컨테이너 클라이언트 컨트롤에 대해 터치 포인터 이벤트 알림을 사용할 수 있는지 여부입니다. (Get)
get_FarmName 팜 이름입니다.
get_FarmName 이 대상이 조인되는 팜의 이름을 검색하거나 지정합니다. (Get)
get_Identifier 작업 에이전트에서 고유 식별자로 사용되는 GUID를 검색합니다.
get_InitialProgram 사용자가 대상 컴퓨터에 로그온할 때 시작되는 프로그램을 나타내는 값을 검색합니다.

get_InitialProgram 이 세션의 초기 프로그램을 검색하거나 지정합니다. (Get)
get_IpAddresses 대상의 외부 IP 주소를 검색하거나 지정합니다. (ITsSbTargetEx.get_IpAddresses)
get_IsFirstAssignment 이것이 첫 번째 할당인지 여부입니다.
get_Label 작업의 용도를 설명하는 레이블을 검색합니다.
get_LoadBalanceResult 부하 분산을 통해 반환된 대상 컴퓨터의 이름을 나타내는 값을 검색합니다.
get_MaxConnectionTime 원격 데스크톱 서비스 세션의 최대 기간입니다. 지정된 시간(분)이 경과한 후 세션의 연결이 끊어지거나 종료될 수 있습니다. (Get)
get_MaxDisconnectionTime RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 연결이 끊긴 원격 데스크톱 서비스 세션이 활성 상태로 유지되는 최대 시간입니다. 지정된 시간(분)이 경과하면 세션이 종료됩니다. (Get)
get_MaxIdleTime 원격 데스크톱 서비스 세션이 유휴 상태로 유지될 수 있는 최대 시간입니다. 지정된 시간(분)이 경과한 후 세션의 연결이 끊어지거나 종료될 수 있습니다. (Get)
get_Name 대상 컴퓨터를 호스트하는 환경의 이름을 나타내는 값을 검색합니다.
get_NumPendingConnections 대상에 대한 보류 중인 사용자 연결 수를 검색합니다.
get_NumSessions broker가 대상에 대해 유지 관리하는 세션 수를 검색합니다.

get_Plugin 작업 에이전트의 표시 이름을 검색합니다.
get_pluginName 플러그 인의 이름을 검색합니다.
get_pluginVersion 플러그 인의 버전을 검색합니다.
get_PointerSpeed 터치 포인터 커서가 클라이언트의 제스처 속도를 기준으로 가상 데스크톱에서 이동하는 속도입니다.
get_ProtocolType 세션의 프로토콜 형식을 검색하거나 지정합니다. (Get)
get_RdFarmType Rd 팜 유형입니다.
get_ReconnectionAction 연결이 끊긴 원격 데스크톱 서비스 세션에 다시 연결할 수 있는지를 지정하는 값입니다. (Get)
get_SamUserAccount 연결을 시작한 사용자의 도메인 이름 및 사용자 이름을 나타내는 값을 검색합니다.
get_ServerWeight 대상 컴퓨터를 호스트하는 환경의 서버 가중치를 나타내는 값을 검색합니다.
get_SessionId 세션 ID를 검색합니다.
get_Settings RDP(원격 데스크톱 프로토콜) 앱 컨테이너 클라이언트에 대한 설정 개체를 검색합니다.
get_StartTime 작업 에이전트가 작업을 시작할 수 있는 가장 빠른 시간을 검색합니다.

get_State 세션 상태를 검색하거나 지정합니다. (Get)
get_Status 작업의 상태를 나타내는 RDV_TASK_STATUS 열거형 값을 검색합니다.
get_TargetFQDN 대상의 정규화된 도메인 이름을 검색하거나 지정합니다. (ITsSbTargetEx.get_TargetFQDN)
get_TargetId 대상 식별자를 검색합니다.
get_TargetLoad 대상의 상대 부하를 검색합니다.
get_TargetName 부하 분산 알고리즘에서 반환된 대상 이름을 검색합니다.
get_TargetName 이 세션을 만든 대상의 이름을 검색합니다.
get_TargetName 대상의 이름을 지정하거나 검색합니다. (Get)
get_TargetNetbios 대상의 NetBIOS 이름을 검색하거나 지정합니다. (Get)
get_TargetPropertySet 대상의 속성 집합을 검색하거나 지정합니다. (ITsSbTargetEx.get_TargetPropertySet)
get_TargetState 대상 상태를 검색하거나 지정합니다. (ITsSbTargetEx.get_TargetState)
get_TerminalServicesHomeDirectory 사용자의 루트 디렉터리입니다. RD 세션 호스트(원격 데스크톱 세션 호스트) 서버의 각 사용자에게는 고유한 루트 디렉터리가 있습니다. 이렇게 하면 애플리케이션 정보가 다중 사용자 환경의 각 사용자에 대해 별도로 저장됩니다. (Get)

get_TerminalServicesHomeDrive 사용자의 루트 드라이브입니다. 네트워크 환경에서 이 속성은 루트 디렉터리로 지정된 UNC 경로가 매핑되는 드라이브 사양(드라이브 문자 뒤에 콜론)이 포함된 문자열입니다. (Get)
get_TerminalServicesInitialProgram 사용자가 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에 로그온할 때 사용자가 자동으로 시작하려는 애플리케이션의 경로 및 파일 이름입니다. (Get)
get_TerminalServicesProfilePath 사용자가 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에 로그온할 때 사용할 로밍 또는 필수 프로필 경로입니다. (Get)
get_TerminalServicesWorkDirectory 사용자의 작업 디렉터리 경로입니다. (Get)
get_TouchPointer RDP(원격 데스크톱 프로토콜) 앱 컨테이너 클라이언트에 대한 RemoteDesktopClientTouchPointer 개체를 포함합니다.
get_Username 이 세션의 사용자 이름을 검색합니다.
get_UserName 연결을 시작한 사용자의 이름을 나타내는 값을 검색합니다.
get_UserSidString 사용자 SID를 문자열로 지정합니다.
GetCacheLastUpdateTime 캐시가 마지막으로 업데이트된 시간을 반환합니다.
GetClaimsToken2 클레임 토큰을 검색합니다.
GetClientData 프로토콜에서 클라이언트 설정을 요청합니다.

GetClientData IWTSProtocolConnection::GetClientData를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnection::GetClientData를 사용합니다.
GetClientMonitorData 클라이언트의 모니터 수와 기본 모니터 번호를 검색합니다.
GetConfiguration 수신기 관련 구성을 검색합니다.
GetConnectionId 연결 식별자를 검색합니다.
GetContext PutContext 메서드를 사용하여 플러그 인에 의해 저장된 컨텍스트 정보를 검색합니다.
GetCurrentPadding 엔드포인트에서 대기 중인 데이터의 양을 100나노초 단위로 가져옵니다.
GetDisconnectedSession 연결이 끊어진 세션을 가져옵니다.
GetEventDrivenCapable 디바이스 엔드포인트가 이벤트 구동인지 여부를 나타냅니다. 디바이스 엔드포인트는 버퍼 가용성을 알리는 이벤트를 설정하여 오디오 엔진의 기간을 제어합니다.
GetFarmProperty 팜의 속성을 검색합니다. (ITsSbGlobalStore.GetFarmProperty)
GetFarmProperty 팜의 속성을 검색합니다. (ITsSbResourcePluginStoreEx.GetFarmProperty)
GetFilterPluginStore 필터 플러그 인 저장소의 FilterPluginStore 인스턴스를 검색합니다.
GetFrameFormat 오디오 엔드포인트의 형식을 검색합니다.

GetFramesPerPacket 엔드포인트의 기간 및 샘플 속도에 따라 오디오 엔드포인트가 지원할 수 있는 패킷당 최대 프레임 수를 가져옵니다.
GetHardwareId 프로토콜 스택은 이 메서드를 사용하여 WDDM ID 드라이버의 하드웨어 ID를 반환합니다.
GetInputDataPointer 오디오 엔진에서 데이터를 읽을 버퍼에 대한 포인터를 가져옵니다.
GetInputHandles 프로토콜에 대한 입력/출력 디바이스에 대한 핸들을 가져옵니다.
GetInstanceOfGlobalStore 전역 저장소 개체의 ITsSbGlobalStore 인스턴스를 검색합니다.
GetLastInputTime 프로토콜이 사용자 입력을 마지막으로 받은 시간을 검색합니다.
GetLastInputTime IWTSProtocolConnection::GetLastInputTime을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnection::GetLastInputTime을 사용합니다.
GetLatency 오디오 엔드포인트의 대기 시간을 가져옵니다.
GetLicenseConnection 클라이언트 라이선스 프로세스를 시작하는 데 사용되는 IWRdsProtocolLicenseConnection 개체를 검색합니다.
GetLicenseConnection IWTSProtocolConnection::GetLicenseConnection을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnection::GetLicenseConnection을 사용합니다.
GetLogonErrorRedirector 프로토콜이 클라이언트 로그온 오류를 처리하는 방법을 지정하는 IWRdsProtocolLogonErrorRedirector 인터페이스를 검색합니다.

GetLogonErrorRedirector IWTSProtocolConnection::GetLogonErrorRedirector를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnection::GetLogonErrorRedirector를 사용합니다.
GetMappedRenderer 서버에서 미디어를 렌더링하는 데 사용되는 비트맵 렌더링 개체를 가져옵니다.
GetMostSuitableTarget 들어오는 클라이언트 연결을 지시할 가장 적합한 대상을 결정합니다.
GetOutputDataPointer 오디오 엔진에서 데이터를 쓸 출력 버퍼에 대한 포인터를 반환합니다.
GetProcessId RemoteApp 및 데스크톱 연결에서 현재 연결의 프로세스 ID를 검색합니다.
GetProtocolHandles IWTSProtocolConnection::GetProtocolHandles를 더 이상 사용할 수 없습니다.
GetProtocolStatus 프로토콜 상태에 대한 정보를 검색합니다.
GetProtocolStatus IWTSProtocolConnection::GetProtocolStatus를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnection::GetProtocolStatus를 사용합니다.
GetRdpProperty 명명된 단일 RDP 속성 값을 검색합니다.
GetRegisteredFileExtensions RemoteApp 및 데스크톱 연결 런타임에 등록된 타사 파일 이름 확장명을 검색합니다.
GetRendererStatistics RemoteFX 미디어 리디렉션 비트맵 렌더러에 대한 통계를 검색합니다.
GetResource 이 메서드는 예약되어 있으며 항상 E_NOTIMPL 반환해야 합니다. (ItsPubPlugin.GetResource)

GetResource2 이 메서드는 예약되어 있으며 항상 E_NOTIMPL 반환해야 합니다. (ItsPubPlugin2.GetResource2)
GetResource2List 지정된 사용자에게 할당된 리소스 목록을 검색합니다. (ItsPubPlugin2.GetResource2List)
GetResourceDisplayName RemoteApp 및 데스크톱 연결에서 사용자 지정 클라이언트의 표시 이름을 반환합니다.
GetResourceId RemoteApp 및 데스크톱 연결에서 사용자 지정 클라이언트의 ID를 반환합니다.
GetResourceList 지정된 사용자에게 할당된 리소스 목록을 검색합니다. (ItsPubPlugin.GetResourceList)
GetResourcePluginStore 리소스 플러그 인 저장소의 ITsSbResourcePluginStore 인스턴스를 검색합니다.
GetResourceTypeInfo RemoteApp 및 데스크톱 연결 런타임에 등록된 타사 파일 이름 확장명 정보를 검색합니다.
GetRTCaps 오디오 디바이스가 실시간(RT) 지원인지 여부를 쿼리합니다. 이 메서드는 IAudioDeviceEndpoint의 원격 데스크톱 서비스 구현에서 사용되지 않습니다.
GetServerState 지정된 서버의 상태를 검색합니다.
GetService 지정된 서비스를 가져옵니다.
GetSessionForEnhancedFastReconnect 프로토콜 스택은 이 메서드를 사용하여 향상된 빠른 다시 연결에 사용할 세션 ID를 반환합니다.
설정가져오기 클라이언트 연결 요청에 대한 수신기 설정 정보를 가져옵니다.

설정가져오기 특정 정책에 대한 설정을 검색합니다.
GetShadowConnection 프로토콜에서 섀도 연결 개체에 대한 참조를 검색합니다.
GetShadowConnection IWTSProtocolConnection::GetShadowConnection을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnection::GetShadowConnection을 사용합니다.
GetSnapshot RDP(원격 데스크톱 프로토콜) 앱 컨테이너 클라이언트의 세션 내 데스크톱 스냅샷을 생성합니다.
GetUserCredentials 사용자 자격 증명을 반환합니다.
GetUserCredentials IWTSProtocolConnection::GetUserCredentials를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnection::GetUserCredentials를 사용합니다.
GetUserData IWTSProtocolConnection::GetUserData를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolSettings::MergeSettings를 사용합니다.
GetVideoHandle 프로토콜에 대한 비디오 디바이스의 핸들을 가져옵니다.
GetWaitTimeout 대기 시간 제한을 검색합니다.
GetWorkspaceNames 현재 프로세스에서 연결의 이름을 검색합니다.
초기화 플러그 인을 초기화하고 플러그 인의 리디렉션 기능을 나타내는 값을 반환합니다.

초기화 플러그 인을 초기화합니다.
초기화 클라이언트에서 플러그 인으로 이루어진 첫 번째 호출에 사용됩니다.
초기화 프로토콜 관리자를 초기화합니다.
InitializeTaskPlugin 원격 데스크톱 연결 브로커 플러그 인의 큐에 있는 작업을 초기화합니다.
InvokeTargetShadow 원격 데스크톱 서비스 서비스에 그림자의 대상 쪽을 시작하도록 지시하고 클라이언트와 대상 간에 교환해야 하는 모든 정보를 전달합니다.
InvokeTargetShadow IWTSProtocolShadowCallback::InvokeTargetShadow를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolShadowCallback::InvokeTargetShadow를 사용합니다.
IsErrorMessageRegistered 지정된 오류 메시지가 지정된 작업 영역에 등록되어 있는지 여부를 확인합니다.
IsQuarantineEnabled 권한 부여 플러그 인에 사용자의 컴퓨터에서 SoH(상태 문)가 필요한지 여부를 나타냅니다.
IssueDisconnect RemoteApp 및 데스크톱 연결에서 사용자 지정 클라이언트의 연결을 끊습니다.
IsUserAllowedToLogon 프로토콜에서 사용자가 세션에 로그온할 수 있는지 여부를 결정합니다.
IsUserAllowedToLogon IWTSProtocolConnection::IsUserAllowedToLogon을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnection::IsUserAllowedToLogon을 사용합니다.

IsWorkspaceCredentialSpecified 지정된 연결 ID에 대한 사용자 자격 증명이 있는지 여부를 확인합니다.
IsWorkspaceSSOEnabled RemoteApp 및 데스크톱 연결에 대해 SSO(Single Sign-On)를 사용할 수 있는지 여부를 확인합니다.
LockVirtualMachine 원격 데스크톱 연결 브로커(RD 연결 브로커)에 가상 머신이 잠겨 있음을 알 수 있습니다.
LogonNotify 사용자가 세션에 로그온했을 때 호출됩니다.
LogonNotify IWTSProtocolConnection::LogonNotify를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnection::LogonNotify를 사용합니다.
MergeSettings 지정한 정책 관련 설정을 더 큰 연결 설정 그룹에 추가(병합)합니다.
ModifyResourceType RemoteApp 및 데스크톱 연결 런타임에 등록된 타사 파일 이름 확장명을 수정합니다.
NotifyClientConnectionStateChange 등록된 플러그 인에 클라이언트 연결의 상태 변경에 대해 알 수 있습니다. (ITsSbResourceNotification.NotifyClientConnectionStateChange)
NotifyClientConnectionStateChangeEx 등록된 플러그 인에 클라이언트 연결의 상태 변경에 대해 알 수 있습니다. (ITsSbResourceNotificationEx.NotifyClientConnectionStateChangeEx)
NotifyCommandProcessCreated 프로토콜에 Winlogon.exe 프로세스가 만들어지고 초기화되었음을 알 수 있습니다.
NotifyServiceFailure 등록된 플러그 인에 RD 연결 브로커(원격 데스크톱 연결 브로커) 서비스가 중지되었음을 알 수 있습니다.

NotifyServiceStateChange 프로토콜 공급자에게 원격 데스크톱 서비스 서비스의 상태가 변경되고 있음을 알 수 있습니다.
NotifyServiceStateChange IWTSProtocolManager::NotifyServiceStateChange를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolManager::NotifyServiceStateChange를 사용합니다.
NotifyServiceSuccess 등록된 플러그 인에 RD 연결 브로커(원격 데스크톱 연결 브로커) 서비스가 시작되었음을 알 수 있습니다.
NotifySessionChange 등록된 플러그 인에 세션 개체의 상태 변경에 대해 알 수 있습니다. (ITsSbResourceNotification.NotifySessionChange)
NotifySessionChangeEx 등록된 플러그 인에 세션 개체의 상태 변경에 대해 알 수 있습니다. (ITsSbResourceNotificationEx.NotifySessionChangeEx)
NotifySessionId 새 세션의 식별자를 프로토콜로 보냅니다.
NotifySessionId IWTSProtocolConnection::NotifySessionId를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnection::NotifySessionId를 사용합니다.
NotifySessionOfServiceStart 지정된 세션에 대해 원격 데스크톱 서비스 서비스가 시작되었음을 프로토콜 공급자에게 알합니다.
NotifySessionOfServiceStart IWTSProtocolManager::NotifySessionOfServiceStart를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolManager::NotifySessionOfServiceStart를 사용합니다.
NotifySessionOfServiceStop 지정된 세션에 대해 원격 데스크톱 서비스 서비스가 중지되었음을 프로토콜 공급자에게 알합니다.

NotifySessionOfServiceStop IWTSProtocolManager::NotifySessionOfServiceStop을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolManager::NotifySessionOfServiceStop을 사용합니다.
NotifySessionStateChange 프로토콜 공급자에게 세션 상태의 변경 내용을 알 수 있습니다.
NotifySessionStateChange IWTSProtocolManager::NotifySessionStateChange를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolManager::NotifySessionStateChange를 사용합니다.
NotifySettingsChange 원격 데스크톱 서비스 서비스 내의 설정 변경 내용을 프로토콜 공급자에게 알합니다.
NotifyTargetChange 등록된 플러그 인에 대상 개체의 상태 변경에 대해 알 수 있습니다. (ITsSbResourceNotification.NotifyTargetChange)
NotifyTargetChangeEx 등록된 플러그 인에 대상 개체의 상태 변경에 대해 알 수 있습니다. (ITsSbResourceNotificationEx.NotifyTargetChangeEx)
OnAuthenticated 연결 ID에 대한 사용자 자격 증명의 인증을 표시한 후 작업 표시줄 알림 영역에 연결 알림을 표시합니다.
OnBeginPainting 로그온 사용자 인터페이스가 그리기를 시작할 준비가 되었다는 것을 프로토콜에 알렸습니다.
OnBeginPainting IWTSProtocolLogonErrorRedirector::OnBeginPainting을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolLogonErrorRedirector::OnBeginPainting을 사용합니다.
OnChannelAuthorized 리소스에 권한을 부여하려는 시도의 결과에 대해 RD 게이트웨이(원격 데스크톱 게이트웨이)에 알합니다.

OnChannelOpened 채널이 열리고 사용할 준비가 되었을 때 또는 채널이 열릴 때 오류가 발생할 때 호출됩니다.
OnClose 채널이 닫혔는지 사용자에게 알합니다.
OnClose 채널이 닫혔을 때 호출됩니다.
Oncompleted 원격 데스크톱 연결 브로커(RD 연결 브로커)에 완료를 보고합니다.
OnConnected 원격 데스크톱 서비스 서비스에 클라이언트 연결 요청이 수신되었음을 알 수 있습니다.
OnConnected IWTSProtocolListenerCallback::OnConnected를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolListenerCallback::OnConnected를 사용합니다.
OnConnectionAuthorized RD 게이트웨이(원격 데스크톱 게이트웨이)에 연결 권한을 부여하려는 시도의 결과를 알 수 있습니다.
OnDataReceived 수신되는 데이터에 대해 사용자에게 알합니다.
OnDataReceived 서버에서 전체 메시지를 받을 때 호출됩니다.
OnDataSent IWRdsGraphicsChannel::Write 메서드가 호출되고 데이터가 전송되었을 때 호출됩니다.
OnDeleteTaskTime RD 연결 브로커(원격 데스크톱 연결 브로커)에 작업이 큐에서 제거되었음을 알 수 있습니다.

OnDriverLoad Termsrv는 이 메서드를 사용하여 로드된 WDDM ID 드라이버의 핸들을 프로토콜 스택에 반환합니다. 이 시점부터 스택은 핸들을 소유하며 드라이버와의 통신이 더 이상 필요하지 않은 후 CloseHandle()을 호출해야 합니다.
OnDriverUnload Termsrv는 이 메서드를 사용하여 PnP가 WDDM ID 드라이버를 언로드했음을 프로토콜 스택에 알릴 수 있습니다.
OnError 원격 데스크톱 연결 브로커(RD 연결 브로커)에 오류 조건을 보고합니다.
OnGetMostSuitableTarget RD 연결 브로커(원격 데스크톱 연결 브로커)에 부하 분산 결과를 반환합니다.
OnInitialized RD 연결 브로커(원격 데스크톱 연결 브로커)에 플러그 인이 Initialize 호출을 완료했음을 알 수 있습니다.
OnJobCancelled RD 연결 브로커(원격 데스크톱 연결 브로커)에 작업이 취소되었음을 알 수 있습니다.
OnJobCompleted 작업이 완료되었다는 것을 RD 연결 브로커(원격 데스크톱 연결 브로커)에 알합니다.
OnJobCreated RD 연결 브로커(원격 데스크톱 연결 브로커)에 프로비저닝 작업이 생성되었다는 것을 알 수 있습니다.
OnMetricsUpdate 네트워크 조건이 변경되었음을 RemoteFX 그래픽 서비스에 알리기 위해 호출됩니다.
OnNewChannelConnection RDC(원격 데스크톱 연결) 클라이언트 플러그 인에서 들어오는 연결에 대한 연결 요청을 수락하거나 거부할 수 있습니다.

OnQueryEnvironmentCompleted 원격 데스크톱 연결 브로커(RD 연결 브로커)에 ITsSbClientConnection 개체에 지정된 환경이 이미 올바른 대상을 호스팅하고 있음을 알 수 있습니다.
OnReady 원격 데스크톱 서비스 서비스가 해당 클라이언트에 대한 연결 프로세스를 계속할 것을 요청합니다.
OnReady IWTSProtocolConnectionCallback::OnReady를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnectionCallback::OnReady를 사용합니다.
OnReadyToConnect 대상이 연결에 대해 성공적으로 준비되면 ITsSbTarget 개체를 RD 연결 브로커(원격 데스크톱 연결 브로커)에 반환합니다.
OnReportStatus 클라이언트 연결 처리와 관련하여 RDC(원격 데스크톱 연결) 클라이언트에 상태 메시지를 보냅니다.
OnReportTasks RD 연결 브로커(원격 데스크톱 연결 브로커)에 새 작업 보고서를 알 수 있습니다.
OnSetTaskTime RD 연결 브로커(원격 데스크톱 연결 브로커)에 작업이 예약되었음을 알 수 있습니다.
OnTargetSizeChanged 렌더링 대상의 크기가 변경되면 호출됩니다.
OnTerminated RD 연결 브로커(원격 데스크톱 연결 브로커)에 플러그 인이 Terminate 호출을 완료했음을 알 수 있습니다.
OnUpdateTaskStatus RD 연결 브로커(원격 데스크톱 연결 브로커)에 작업의 상태가 변경되었음을 알 수 있습니다.
OnUserAuthenticated RD 게이트웨이(원격 데스크톱 게이트웨이)에 인증 플러그 인이 사용자를 성공적으로 인증했음을 알 수 있습니다.

OnUserAuthenticationFailed RD 게이트웨이(원격 데스크톱 게이트웨이)에 인증 플러그 인이 사용자를 인증하지 못했음을 알 수 있습니다.
OnVirtualMachineHostStatusChanged 원격 데스크톱 연결 브로커(RD 연결 브로커)에 가상 머신 호스트의 상태가 변경됨을 알 수 있습니다.
OnVirtualMachineStatusChanged 원격 데스크톱 연결 브로커(RD 연결 브로커)에 가상 머신의 상태가 변경됨을 알 수 있습니다.
열기 채널을 열기 위해 호출됩니다.
PatchVirtualMachines 가상 머신을 비동기적으로 패치합니다.
PreDisconnect 세션의 연결이 끊어질 예정임을 프로토콜에 알합니다.
PrepareTargetForConnect 클라이언트 연결에 대한 대상을 준비합니다.
ProcessIdToSessionId 지정된 프로세스와 연결된 원격 데스크톱 서비스 세션을 검색합니다.
ProcessingComplete 처리 패스가 완료되었음을 엔드포인트에 알합니다.
ProtocolComplete 라이선스 프로세스가 성공적으로 완료되었는지 여부를 프로토콜에 알릴 수 있습니다.
ProtocolComplete IWTSProtocolLicenseConnection::P rotocolComplete를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolLicenseConnection::P rotocolComplete를 사용합니다.

PulseEndpoint 예약되어 있습니다. (IAudioInputEndpointRT.PulseEndpoint)
PulseEndpoint 예약되어 있습니다. (IAudioOutputEndpointRT.PulseEndpoint)
put_AllowLogon 사용자가 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에 로그온할 수 있는지 여부를 지정하는 값입니다. (Put)
put_BrokenConnectionAction 원격 데스크톱 서비스 세션 제한에 도달할 때 수행할 작업을 지정하는 값입니다. (Put)
put_ClientDisplay 클라이언트 컴퓨터의 표시 디바이스에 대한 정보를 검색하거나 지정합니다. (Put)
put_ConnectClientDrivesAtLogon 원격 데스크톱 서비스 세션이 시작될 때 매핑된 클라이언트 드라이브를 다시 연결해야 하는지를 지정하는 값입니다. (Put)
put_ConnectClientPrintersAtLogon 로그온 시 매핑된 클라이언트 프린터에 다시 연결할지 여부를 지정하는 값입니다. (Put)
put_CreateTime 세션을 만든 시간을 검색하거나 지정합니다. (Put)
put_DefaultToMainPrinter 클라이언트의 기본 프린터에 자동으로 인쇄할지 여부를 지정하는 값입니다. (Put)
put_DisconnectTime 세션의 연결이 끊긴 시간을 검색하거나 지정합니다. (Put)
put_Enabled RDP 앱 컨테이너 클라이언트 컨트롤에서 터치 포인터 기능을 사용할 수 있는지 여부입니다. (Put)

put_EnableRemoteControl 사용자의 원격 데스크톱 서비스 세션에 대한 원격 관찰 또는 원격 제어를 허용할지 여부를 지정하는 값입니다. (Put)
put_EnvironmentName 대상과 연결된 환경의 이름을 검색하거나 지정합니다. (ITsSbTargetEx.put_EnvironmentName)
put_EnvironmentPropertySet 대상 컴퓨터를 호스트하는 환경에 대한 속성 집합을 검색하거나 지정합니다. (Put)
put_EventsEnabled RDP 앱 컨테이너 클라이언트 컨트롤에 대해 터치 포인터 이벤트 알림을 사용할 수 있는지 여부입니다. (Put)
put_FarmName 이 대상이 조인된 팜의 이름을 검색하거나 지정합니다. (Put)
put_InitialProgram 이 세션의 초기 프로그램을 검색하거나 지정합니다. (Put)
put_IpAddresses 대상의 외부 IP 주소를 검색하거나 지정합니다. (ITsSbTargetEx.put_IpAddresses)
put_MaxConnectionTime 원격 데스크톱 서비스 세션의 최대 기간입니다. 지정된 시간(분)이 경과하면 세션의 연결이 끊어지거나 종료될 수 있습니다. (Put)
put_MaxDisconnectionTime 연결이 끊긴 원격 데스크톱 서비스 세션이 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 활성 상태로 유지되는 최대 시간입니다. 지정된 시간(분)이 경과하면 세션이 종료됩니다. (Put)
put_MaxIdleTime 원격 데스크톱 서비스 세션이 유휴 상태로 유지될 수 있는 최대 시간입니다. 지정된 시간(분)이 경과하면 세션의 연결이 끊어지거나 종료될 수 있습니다. (Put)
put_ProtocolType 세션의 프로토콜 형식을 검색하거나 지정합니다. (Put)

put_ReconnectionAction 연결이 끊어진 원격 데스크톱 서비스 세션에 다시 연결할 수 있는지를 지정하는 값입니다. (Put)
put_State 세션 상태를 검색하거나 지정합니다. (Put)
put_TargetFQDN 대상의 정규화된 도메인 이름을 검색하거나 지정합니다. (ITsSbTargetEx.put_TargetFQDN)
put_TargetName 대상의 이름을 지정하거나 검색합니다. (Put)
put_TargetNetbios 대상의 NetBIOS 이름을 검색하거나 지정합니다. (Put)
put_TargetPropertySet 대상의 속성 집합을 검색하거나 지정합니다. (ITsSbTargetEx.put_TargetPropertySet)
put_TargetState 대상 상태를 검색하거나 지정합니다. (ITsSbTargetEx.put_TargetState)
put_TerminalServicesHomeDirectory 사용자의 루트 디렉터리입니다. RD 세션 호스트(원격 데스크톱 세션 호스트) 서버의 각 사용자에게는 고유한 루트 디렉터리가 있습니다. 이렇게 하면 애플리케이션 정보가 다중 사용자 환경의 각 사용자에 대해 별도로 저장됩니다. (Put)
put_TerminalServicesHomeDrive 사용자의 루트 드라이브입니다. 네트워크 환경에서 이 속성은 루트 디렉터리로 지정된 UNC 경로가 매핑되는 드라이브 사양(드라이브 문자 뒤에 콜론)이 포함된 문자열입니다. (Put)
put_TerminalServicesInitialProgram 사용자가 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에 로그온할 때 사용자가 자동으로 시작하려는 애플리케이션의 경로 및 파일 이름입니다. (Put)
put_TerminalServicesProfilePath 사용자가 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에 로그온할 때 사용할 로밍 또는 필수 프로필 경로입니다. (Put)

put_TerminalServicesWorkDirectory 사용자의 작업 디렉터리 경로입니다. (Put)
PutContext 플러그 인에서 연결과 관련된 컨텍스트 정보를 저장하는 데 사용할 수 있습니다.
QueryEnvironment 지정된 환경 개체를 반환합니다.
QueryEnvironmentForTarget 지정된 환경이 부하 분산에 의해 반환된 대상을 호스트할 준비가 되었는지 여부를 결정합니다.
QueryProperty 프로토콜에서 속성 값을 검색합니다.
QueryProperty IWTSProtocolConnection::QueryProperty를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnection::QueryProperty를 사용합니다.
QuerySessionBySessionId 지정된 세션 ID와 연결된 ITsSbSession 개체를 검색합니다.
QuerySessionBySessionId 지정된 세션 ID가 있는 세션 개체를 반환합니다.
QueryTarget 지정된 매개 변수에 대한 ITsSbTarget 개체를 검색합니다.
QueryTarget 지정된 대상 이름과 팜 이름을 포함하는 대상을 반환합니다.
ReauthenticateUser 원격 데스크톱 게이트웨이(RD 게이트웨이)에 사용자를 자동으로 다시 인증하고 다시 인증해야 한다는 알림이 표시됩니다.

다시 연결 세션을 새 너비 및 높이에 맞게 RDP(원격 데스크톱 프로토콜) 앱 컨테이너 클라이언트 컨트롤의 자동 다시 연결을 시작합니다.
RedirectLogonError 로그온 오류에 대한 응답으로 수행할 작업에 대한 프로토콜을 쿼리합니다.
RedirectLogonError IWTSProtocolLogonErrorRedirector::RedirectLogonError를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolLogonErrorRedirector::RedirectLogonError를 사용합니다.
RedirectMessage 로그온 메시지를 리디렉션하는 방법에 대해 프로토콜을 쿼리합니다.
RedirectMessage IWTSProtocolLogonErrorRedirector::RedirectMessage를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolLogonErrorRedirector::RedirectMessage를 사용합니다.
RedirectStatus 클라이언트 로그온 상태 업데이트를 리디렉션하는 방법과 관련하여 프로토콜을 쿼리합니다.
RedirectStatus IWTSProtocolLogonErrorRedirector::RedirectStatus를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolLogonErrorRedirector::RedirectStatus를 사용합니다.
RedrawWindow 원격 데스크톱 서비스 서비스가 클라이언트 창을 다시 그릴 것을 요청합니다.
RedrawWindow IWTSProtocolConnectionCallback::RedrawWindow를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnectionCallback::RedrawWindow를 사용합니다.
새로 고침 이 메서드는 예약되어 있습니다.
RegisterErrorEvent UI에서 사용할 지정된 오류 이벤트 메시지를 등록합니다.

RegisterErrorLogMessage UI에서 사용할 지정된 오류 메시지를 등록합니다.
RegisterForNotification RD 연결 브로커(원격 데스크톱 연결 브로커)가 지정된 이벤트에 대한 알림을 보내도록 요청합니다.
ReleaseInputDataPointer 획득한 데이터 포인터를 해제합니다.
ReleaseOutputDataPointer 출력 버퍼에 대한 포인터를 해제합니다.
ReleaseTargetLock 대상에 대한 잠금을 해제합니다.
RemoveEnvironmentFromStore 리소스 플러그 인 저장소에서 지정된 환경을 제거합니다.
RemoveMapping 동적 가상 채널 플러그 인에 의해 호출되어 렌더링 매핑을 제거합니다.
RemoveResource RemoteApp 및 데스크톱 연결 런타임에 클라이언트가 연결 연결을 끊고 있음을 알릴 수 있습니다. (IWorkspaceRegistration2.RemoveResource)
RemoveResourceEx RemoteApp 및 데스크톱 연결 런타임에 클라이언트가 연결 연결을 끊고 있음을 알릴 수 있습니다. (IWorkspaceRegistration2::RemoveResourceEx)
렌더링 비트맵을 렌더링하기 위해 동적 가상 채널 플러그 인에서 호출됩니다.
RequestClientLicense 클라이언트에서 라이선스를 요청합니다.

RequestClientLicense IWTSProtocolLicenseConnection::RequestClientLicense를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolLicenseConnection::RequestClientLicense를 사용합니다.
RequestLicensingCapabilities 클라이언트에서 라이선스 기능을 요청합니다.
RequestLicensingCapabilities IWTSProtocolLicenseConnection::RequestLicensingCapabilities를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolLicenseConnection::RequestLicensingCapabilities를 사용합니다.
재설정 엔드포인트 스트림을 다시 설정합니다.
ResolvePersonalDesktop 개인 가상 데스크톱 컬렉션에서 지정된 사용자와 가상 머신 간의 매핑을 해결하기 위해 호출됩니다.
ResolveResource 사용자의 할당된 개인 가상 데스크톱에 연결하는 방법에 대한 정보를 제공합니다.
ResourceDismissed 리소스가 비활성화되었거나 해제되었음을 사용자에게 알릴 수 있습니다.
ResumeScreenUpdates 클라이언트로 전송되는 화면 업데이트를 다시 시작합니다.
RetrieveSettings 전체 RDP 파일을 문자열로 검색합니다.
SaveEnvironment 환경을 저장합니다.
SaveProperties 속성 집합을 저장합니다.

SaveSession 세션을 저장합니다.
SaveTarget 대상을 저장합니다.
SendBeep IWTSProtocolConnection::SendBeep을 더 이상 사용할 수 없습니다.
SendClientLicense 클라이언트에 라이선스를 보냅니다.
SendClientLicense IWTSProtocolLicenseConnection::SendClientLicense를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolLicenseConnection::SendClientLicense를 사용합니다.
SendPolicyData IWTSProtocolConnection::SendPolicyData를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolManager::NotifySettingsChange를 사용합니다.
SessionArbitrationEnumeration 프로토콜이 세션을 다시 연결하도록 지정할 수 있도록 중재 후에 호출됩니다.
SessionArbitrationEnumeration IWTSProtocolConnection::SessionArbitrationEnumeration을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnection::SessionArbitrationEnumeration을 사용합니다.
SetBuffer 엔드포인트를 초기화하고 오디오 데이터가 스트리밍되는 엔드포인트의 형식에 따라 버퍼를 만듭니다.
SetClaimsToken 클레임 토큰을 설정합니다.
SetEnvironmentProperty 환경의 속성을 설정합니다. (ITsSbResourcePluginStoreEx.SetEnvironmentProperty)

SetEnvironmentPropertyWithVersionCheck 환경의 속성을 설정합니다. (ITsSbResourcePluginStore.SetEnvironmentPropertyWithVersionCheck)
SetErrorInfo 프로토콜에서 오류 정보를 설정합니다.
SetErrorInfo IWTSProtocolConnection::SetErrorInfo를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnection::SetErrorInfo를 사용합니다.
SetEventHandle 엔드포인트가 버퍼 처리를 완료했음을 알리는 데 사용하는 이벤트에 대한 핸들을 설정합니다.
SetPinActive 기본 스트리밍 리소스의 상태를 활성 상태로 변경해야 한다는 사실을 엔드포인트에 알합니다.
SetPinInactive 기본 스트림 리소스의 상태를 비활성 상태로 변경해야 한다는 사실을 엔드포인트에 알합니다.
SetRdpProperty 명명된 단일 RDP 속성의 값을 설정합니다.
SetServerDrainMode 지정된 서버의 드레이닝 모드를 설정합니다.
SetServerWaitingToStart 서버가 시작 대기 중임을 세션 호스트에 나타냅니다.
SetSessionState 세션 상태를 설정합니다.
SetStreamFlags 오디오 엔드포인트에서 스트림 구성 플래그를 설정합니다.
SetTargetProperty 대상의 속성 값을 설정합니다. (ITsSbResourcePluginStoreEx.SetTargetProperty)

SetTargetPropertyWithVersionCheck 대상의 속성 값을 설정합니다. (ITsSbResourcePluginStore.SetTargetPropertyWithVersionCheck)
SetTargetState 대상 개체의 상태를 설정합니다.
SetTaskQueue 원격 데스크톱 연결 브로커 플러그 인의 큐에 작업을 업데이트.
시작 엔드포인트 스트림을 시작합니다.
시작 섀도링이 시작되었음을 프로토콜에 알합니다.
시작 IWTSProtocolShadowConnection::Start는 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolShadowConnection::Start를 사용합니다.
StartListen 클라이언트 연결 요청 수신 대기를 시작하도록 프로토콜에 알합니다.
StartListen IWTSProtocolListener::StartListen을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolListener::StartListen을 사용합니다.
StartRemoteApplication RemoteApp 프로그램을 시작합니다.
StartRemoteApplicationEx 지원 안 됨 (IWorkspace2.StartRemoteApplicationEx)
StartWorkspace 사용자 자격 증명 및 인증서를 연결 ID와 연결합니다.

StartWorkspaceEx 사용자 자격 증명 및 인증서를 연결 ID와 연결합니다. 에는 추가 보안 및 UI 요소도 포함되어 있습니다.
StartWorkspaceEx2 구현되지 않았습니다. (IWorkspaceScriptable3::StartWorkspaceEx2)
중지 엔드포인트 스트림을 중지합니다.
중지 섀도링이 중지되었음을 프로토콜에 알합니다.
중지 IWTSProtocolShadowConnection::Stop을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolShadowConnection::Stop을 사용합니다.
StopListen 클라이언트 연결 요청 수신 대기를 중지하도록 프로토콜에 알합니다.
StopListen IWTSProtocolListener::StopListen을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolListener::StopListen을 사용합니다.
StopScreenUpdates 원격 데스크톱 서비스 서비스가 클라이언트 화면 업데이트를 중지할 것을 요청합니다.
StopScreenUpdates IWTSProtocolConnectionCallback::StopScreenUpdates를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnectionCallback::StopScreenUpdates를 사용합니다.
StopShadow 대상 그림자를 중지하도록 원격 데스크톱 서비스 서비스에 지시합니다.
StopShadow IWTSProtocolShadowCallback::StopShadow를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolShadowCallback::StopShadow를 사용합니다.

SuspendScreenUpdates 클라이언트로 전송되는 화면 업데이트를 일시 중단합니다.
Terminate 정리를 수행하고 플러그 인을 언로드합니다.
종료됨 터미널 서비스 세션 브로커(TS 세션 브로커)에 의해 제거될 예정임을 플러그 인에 알합니다.
종료됨 RDC(원격 데스크톱 연결) 클라이언트가 종료되었음을 플러그 인에 알합니다.
TestAndSetServerState 서버에서 조건부로 새 상태를 설정합니다.
Uninitialize 프로토콜 관리자를 초기화하지 않습니다.
UnRegisterForNotification 원격 데스크톱 연결 브로커(RD 연결 브로커)가 지정된 이벤트에 대한 알림을 보내지 않도록 요청합니다.
UpdateSessionDisplaySettings RDP(원격 데스크톱 프로토콜) 앱 컨테이너 클라이언트 컨트롤의 너비 및 높이 설정을 업데이트.
VIRTUALCHANNELCLOSE 가상 채널의 클라이언트 끝을 닫습니다.
VIRTUALCHANNELENTRY 원격 데스크톱 서비스 가상 채널을 사용하는 애플리케이션의 클라이언트 쪽 DLL에 대한 애플리케이션 정의 진입점입니다.
VIRTUALCHANNELINIT 원격 데스크톱 서비스 가상 채널에 대한 클라이언트 DLL의 액세스를 초기화합니다.
VIRTUALCHANNELOPEN 가상 채널의 클라이언트 끝을 엽니다.

VIRTUALCHANNELWRITE 가상 채널의 클라이언트 끝에서 서버 끝의 파트너 애플리케이션으로 데이터를 보냅니다.
쓰기 채널에서 쓰기 요청을 시작합니다.
쓰기 가상 채널에 데이터를 보내기 위해 호출되었습니다.
WriteExclusiveModeParametersToSharedMemory 전용 모드 매개 변수를 만들고 공유 메모리에 씁니다.
WTSCloseServer RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에 대한 열린 핸들을 닫습니다.
WTSConnectSessionA 원격 데스크톱 서비스 세션을 로컬 컴퓨터의 기존 세션에 연결합니다. (ANSI)
WTSConnectSessionW 원격 데스크톱 서비스 세션을 로컬 컴퓨터의 기존 세션에 연결합니다. (유니코드)
WTSCreateListenerA 새 원격 데스크톱 서비스 수신기를 만들거나 기존 수신기를 구성합니다. (ANSI)
WTSCreateListenerW 새 원격 데스크톱 서비스 수신기를 만들거나 기존 수신기를 구성합니다. (유니코드)
WTSDisconnectSession 세션을 닫지 않고 지정된 원격 데스크톱 서비스 세션에서 로그온한 사용자의 연결을 끊습니다.
WTSEnableChildSessions 자식 세션을 사용하거나 사용하지 않도록 설정합니다.
WTSEnumerateListenersA RD 세션 호스트(원격 데스크톱 세션 호스트) 서버의 모든 원격 데스크톱 서비스 수신기를 열거합니다. (ANSI)

WTSEnumerateListenersW RD 세션 호스트(원격 데스크톱 세션 호스트) 서버의 모든 원격 데스크톱 서비스 수신기를 열거합니다. (유니코드)
WTSEnumerateProcessesA 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 활성 프로세스에 대한 정보를 검색합니다. (ANSI)
WTSEnumerateProcessesExA 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버 또는 RD 가상화 호스트(원격 데스크톱 가상화 호스트) 서버의 활성 프로세스에 대한 정보를 검색합니다. (ANSI)
WTSEnumerateProcessesExW 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버 또는 RD 가상화 호스트(원격 데스크톱 가상화 호스트) 서버의 활성 프로세스에 대한 정보를 검색합니다. (유니코드)
WTSEnumerateProcessesW 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 활성 프로세스에 대한 정보를 검색합니다. (유니코드)
WTSEnumerateServersA 지정된 도메인 내의 모든 원격 데스크톱 세션 호스트(RD 세션 호스트) 서버 목록을 반환합니다. (ANSI)
WTSEnumerateServersW 지정된 도메인 내의 모든 원격 데스크톱 세션 호스트(RD 세션 호스트) 서버 목록을 반환합니다. (유니코드)
WTSEnumerateSessionsA RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 세션 목록을 검색합니다. (ANSI)
WTSEnumerateSessionsExA 지정된 RD 세션 호스트(RD 세션 호스트) 서버 또는 RD 가상화 호스트(원격 데스크톱 가상화 호스트) 서버에서 세션 목록을 검색합니다. (ANSI)
WTSEnumerateSessionsExW 지정된 RD 세션 호스트(RD 세션 호스트) 서버 또는 RD 가상화 호스트(원격 데스크톱 가상화 호스트) 서버에서 세션 목록을 검색합니다. (유니코드)

WTSEnumerateSessionsW RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 세션 목록을 검색합니다. (유니코드)
WTSFreeMemory 원격 데스크톱 서비스 함수에 의해 할당된 메모리를 해제합니다.
WTSFreeMemoryExA 원격 데스크톱 서비스 함수에서 할당한 WTS_PROCESS_INFO_EX 또는 WTS_SESSION_INFO_1 구조를 포함하는 메모리를 해제합니다. (ANSI)
WTSFreeMemoryExW 원격 데스크톱 서비스 함수에서 할당한 WTS_PROCESS_INFO_EX 또는 WTS_SESSION_INFO_1 구조를 포함하는 메모리를 해제합니다. (유니코드)
WTSGetActiveConsoleSessionId 콘솔 세션의 세션 식별자를 검색합니다.
WTSGetChildSessionId 자식 세션 식별자(있는 경우)를 검색합니다.
WTSGetListenerSecurityA 원격 데스크톱 서비스 수신기의 보안 설명자를 검색합니다. (ANSI)
WTSGetListenerSecurityW 원격 데스크톱 서비스 수신기의 보안 설명자를 검색합니다. (유니코드)
WTSIsChildSessionsEnabled 자식 세션을 사용할 수 있는지 여부를 결정합니다.
WTSLogoffSession 지정된 원격 데스크톱 서비스 세션을 로그오프합니다.
WTSOpenServerA 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에 대한 핸들을 엽니다. (ANSI)

WTSOpenServerExA 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버 또는 RD 가상화 호스트(원격 데스크톱 가상화 호스트) 서버에 대한 핸들을 엽니다. (ANSI)
WTSOpenServerExW 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버 또는 RD 가상화 호스트(원격 데스크톱 가상화 호스트) 서버에 대한 핸들을 엽니다. (유니코드)
WTSOpenServerW 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에 대한 핸들을 엽니다. (유니코드)
WTSQueryListenerConfigA 원격 데스크톱 서비스 수신기에 대한 구성 정보를 검색합니다. (ANSI)
WTSQueryListenerConfigW 원격 데스크톱 서비스 수신기에 대한 구성 정보를 검색합니다. (유니코드)
WTSQuerySessionInformationA 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 지정된 세션에 대한 세션 정보를 검색합니다. (ANSI)
WTSQuerySessionInformationW 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 지정된 세션에 대한 세션 정보를 검색합니다. (유니코드)
WTSQueryUserConfigA 지정된 도메인 컨트롤러 또는 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 지정된 사용자에 대한 구성 정보를 검색합니다. (ANSI)
WTSQueryUserConfigW 지정된 도메인 컨트롤러 또는 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 지정된 사용자에 대한 구성 정보를 검색합니다. (유니코드)
WTSQueryUserToken 세션 ID로 지정된 로그온한 사용자의 기본 액세스 토큰을 가져옵니다.
WTSRegisterSessionNotification 지정된 창을 등록하여 세션 변경 알림을 받습니다. (WTSRegisterSessionNotification)

WTSRegisterSessionNotificationEx 지정된 창을 등록하여 세션 변경 알림을 받습니다. (WTSRegisterSessionNotificationEx)
WTSSBX_GetMostSuitableServer 터미널 서비스 세션 브로커(TS 세션 브로커)가 들어오는 연결을 지시할 서버의 ID를 반환합니다.
WTSSBX_GetUserExternalSession 리소스에 대한 정보를 포함하는 WTSSBX_MACHINE_CONNECT_INFO 구조를 제공하여 들어오는 연결을 가상 머신, 블레이드 서버 또는 사용자 고유의 회사 데스크톱과 같은 컴퓨팅 리소스로 리디렉션합니다.
WTSSBX_MachineChangeNotification 플러그 인에 서버 환경에서 변경이 발생했음을 알 수 있습니다.
WTSSBX_SessionChangeNotification 플러그 인에 로그온, 로그오프, 연결 끊기 또는 다시 연결과 같은 변경 내용이 세션에서 발생했음을 알 수 있습니다.
WTSSendMessageA 지정된 원격 데스크톱 서비스 세션의 클라이언트 데스크톱에 메시지 상자를 표시합니다. (ANSI)
WTSSendMessageW 지정된 원격 데스크톱 서비스 세션의 클라이언트 데스크톱에 메시지 상자를 표시합니다. (유니코드)
WTSSetListenerSecurityA 원격 데스크톱 서비스 수신기의 보안 설명자를 구성합니다. (ANSI)
WTSSetListenerSecurityW 원격 데스크톱 서비스 수신기의 보안 설명자를 구성합니다. (유니코드)
WTSSetRenderHint 실제 콘텐츠인 창의 영역을 식별하기 위해 원격 세션에 표시하도록 최적화할 수 있는 콘텐츠를 표시하는 애플리케이션에서 사용됩니다.
WTSSetUserConfigA 지정된 도메인 컨트롤러 또는 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 지정된 사용자에 대한 구성 정보를 수정합니다. (ANSI)

WTSSetUserConfigW 지정된 도메인 컨트롤러 또는 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 지정된 사용자에 대한 구성 정보를 수정합니다. (유니코드)
WTSShutdownSystem 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버를 종료하고 필요에 따라 다시 시작합니다.
WTSStartRemoteControlSessionA 다른 원격 데스크톱 서비스 세션의 원격 제어를 시작합니다. 원격 세션에서 이 함수를 호출해야 합니다. (ANSI)
WTSStartRemoteControlSessionW 다른 원격 데스크톱 서비스 세션의 원격 제어를 시작합니다. 원격 세션에서 이 함수를 호출해야 합니다. (유니코드)
WTSStopRemoteControlSession 원격 제어 세션을 중지합니다.
WTSTerminateProcess 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 지정된 프로세스를 종료합니다.
WTSUnRegisterSessionNotification 더 이상 세션 변경 알림을 받지 않도록 지정된 창을 등록 취소합니다. (WTSUnRegisterSessionNotification)
WTSUnRegisterSessionNotificationEx 더 이상 세션 변경 알림을 받지 않도록 지정된 창을 등록 취소합니다. (WTSUnRegisterSessionNotificationEx)
WTSVirtualChannelClose 열린 가상 채널 핸들을 닫습니다.
WTSVirtualChannelOpen 지정된 가상 채널의 서버 끝에 대한 핸들을 엽니다.
WTSVirtualChannelOpenEx WTSVirtualChannelOpen과 유사한 방식으로 가상 채널을 만듭니다.

WTSVirtualChannelPurgeInput 클라이언트에서 지정된 가상 채널의 서버로 전송된 큐에 대기 중인 모든 입력 데이터를 삭제합니다.
WTSVirtualChannelPurgeOutput 지정된 가상 채널에서 서버에서 클라이언트로 전송된 큐에 대기 중인 모든 출력 데이터를 삭제합니다.
WTSVirtualChannelQuery 지정된 가상 채널에 대한 정보를 반환합니다.
WTSVirtualChannelRead 가상 채널의 서버 끝에서 데이터를 읽습니다.
WTSVirtualChannelWrite 가상 채널의 서버 끝에 데이터를 씁니다.
WTSWaitSystemEvent 호출자에게 돌아가기 전에 원격 데스크톱 서비스 이벤트를 기다립니다.

인터페이스

IADsTSUserEx 원격 데스크톱 서비스 사용자 속성을 검사하고 구성하는 데 사용됩니다.
IAudioDeviceEndpoint 디바이스 엔드포인트 개체를 초기화하고 해당 개체가 나타내는 디바이스의 기능을 가져옵니다.
IAudioEndpoint 오디오 엔드포인트에 대한 정보를 오디오 엔진에 제공합니다. 이 인터페이스는 오디오 엔드포인트에 의해 구현됩니다.
IAudioEndpointControl 엔드포인트의 스트림 상태를 제어합니다.

IAudioEndpointRT 엔드포인트 버퍼에서 현재 읽기 및 쓰기 위치 간의 차이를 가져옵니다.
IAudioInputEndpointRT 각 처리 패스에 대한 입력 버퍼를 가져옵니다.
IAudioOutputEndpointRT 각 처리 패스에 대한 출력 버퍼를 가져옵니다.
IRemoteDesktopClient RDP(원격 데스크톱 프로토콜) 앱 컨테이너 클라이언트 컨트롤을 구성하고 사용하는 데 사용되는 메서드 및 속성을 제공합니다.
IRemoteDesktopClientActions RDP(원격 데스크톱 프로토콜) 앱 컨테이너 클라이언트 컨트롤과 상호 작용하는 데 사용되는 메서드를 제공합니다.
IRemoteDesktopClientSettings RDP(원격 데스크톱 프로토콜) 앱 컨테이너 클라이언트 컨트롤에 대한 연결 설정을 구성하는 데 필요한 메서드를 제공합니다.
IRemoteDesktopClientTouchPointer RDP(원격 데스크톱 프로토콜) 앱 컨테이너 클라이언트 컨트롤의 터치 포인터 기능을 제어하는 데 필요한 속성을 제공합니다.
ITSGAccountingEngine 연결에 대한 세션 생성 또는 종료에 대한 정보를 제공하는 메서드를 노출합니다.
ITSGAuthenticateUserSink 인증 이벤트에 대해 RD 게이트웨이(원격 데스크톱 게이트웨이)에 알리는 메서드를 노출합니다.
ITSGAuthenticationEngine RD 게이트웨이(원격 데스크톱 게이트웨이)에 대한 사용자를 인증하는 메서드를 노출합니다.
ITSGAuthorizeConnectionSink 연결 권한을 부여하려는 시도의 결과에 대해 RD 게이트웨이(원격 데스크톱 게이트웨이)에 알리는 메서드를 노출합니다.

ITSGAuthorizeResourceSink 리소스에 권한을 부여하려는 시도의 결과에 대해 RD 게이트웨이(원격 데스크톱 게이트웨이)에 알리는 메서드를 노출합니다.
ITSGPolicyEngine 연결 및 리소스에 권한을 부여하는 메서드를 노출합니다.
ItsPubPlugin RemoteApp 및 데스크톱 연결 사용자가 사용할 수 있는 리소스에 대한 정보를 제공하는 속성 및 메서드를 노출합니다.
ItsPubPlugin2 RemoteApp 및 데스크톱 연결 사용자가 사용할 수 있는 리소스에 대한 정보를 제공하는 메서드를 지정합니다.
ITsSbBaseNotifySink 상태 및 오류 메시지를 보고하는 메서드를 RD 연결 브로커(원격 데스크톱 연결 브로커)에 노출합니다.
ITsSbClientConnection RDC(원격 데스크톱 연결) 클라이언트에서 들어오는 연결 요청에 대한 상태 정보를 저장하는 메서드 및 속성을 노출합니다.
ITsSbClientConnectionPropertySet 클라이언트 연결의 사용자 지정 속성을 적절하게 정의하는 데 사용할 수 있습니다.
ITsSbEnvironment 대상 컴퓨터를 호스트하는 환경에 대한 정보가 포함된 메서드 및 속성을 노출합니다. 이 인터페이스는 가상 머신을 호스트하는 물리적 서버에 대한 정보를 저장하는 데 사용할 수 있습니다.
ITsSbEnvironmentPropertySet 대상 컴퓨터를 적절하게 호스트하는 환경의 사용자 지정 속성을 정의하는 데 사용할 수 있습니다.
ITsSbFilterPluginStore 플러그 인 저장소 필터링

ITsSbGenericNotifySink 완료를 보고하고 RD 연결 브로커(원격 데스크톱 연결 브로커)에서 대기 시간을 가져오는 메서드를 노출합니다.
ITsSbGlobalStore RD 연결 브로커(원격 데스크톱 연결 브로커) 저장소에 추가된 대상 컴퓨터, 세션, 환경 및 팜을 쿼리하는 메서드를 노출합니다.
ITsSbLoadBalanceResult 부하 분산 알고리즘에서 반환된 대상 이름을 저장하는 메서드 및 속성을 노출합니다.
ITsSbLoadBalancing 사용자 지정 부하 분산 알고리즘을 제공하는 데 사용할 수 있는 메서드를 노출합니다.
ITsSbLoadBalancingNotifySink 부하 분산 알고리즘의 결과를 반환하는 메서드를 RD 연결 브로커(원격 데스크톱 연결 브로커)에 노출합니다.
ITsSbOrchestration RD 연결 브로커(원격 데스크톱 연결 브로커)가 클라이언트를 리디렉션하기 전에 대상이 준비되었는지 확인하는 데 사용하는 메서드를 노출합니다.
ITsSbOrchestrationNotifySink 대상이 연결에 대해 준비되면 ITsSbTarget 개체를 RD 연결 브로커(원격 데스크톱 연결 브로커)에 반환하는 메서드를 노출합니다.
ITsSbPlacement 환경을 준비하는 메서드(가상 머신을 호스트하는 컴퓨터)를 노출합니다.
ITsSbPlacementNotifySink 환경에 대한 정보를 반환하는 메서드를 RD 연결 브로커(원격 데스크톱 연결 브로커)에 노출합니다.
ITsSbPlugin 플러그 인을 초기화하고 종료하는 메서드를 노출합니다.
ITsSbPluginNotifySink RD 연결 브로커(원격 데스크톱 연결 브로커)에 플러그 인의 초기화 또는 종료를 알리는 메서드를 노출합니다.

ITsSbPluginPropertySet 사용자 지정 플러그 인 속성을 적절하게 정의하는 데 사용할 수 있습니다.
ITsSbPropertySet 사용자 지정 속성을 적절하게 정의하는 데 사용할 수 있습니다.
ITsSbProvider 원격 데스크톱 가상화에 사용되는 개체의 기본 구현을 만드는 메서드를 노출합니다.
ITsSbProvisioning 가상 머신을 만들고 유지 관리하는 메서드를 노출합니다.
ITsSbProvisioningPluginNotifySink RD 연결 브로커(원격 데스크톱 연결 브로커)에 가상 머신 프로비저닝에 대해 알리는 메서드를 노출합니다.
ITsSbResourceNotification RD 연결 브로커(원격 데스크톱 연결 브로커)가 세션, 대상 및 클라이언트 연결 개체에서 발생하는 상태 변경 내용을 플러그 인에 알리는 데 사용하는 메서드를 노출합니다. (ITsSbResourceNotification)
ITsSbResourceNotificationEx RD 연결 브로커(원격 데스크톱 연결 브로커)가 세션, 대상 및 클라이언트 연결 개체에서 발생하는 상태 변경 내용을 플러그 인에 알리는 데 사용하는 메서드를 노출합니다. (ITsSbResourceNotificationEx)
ITsSbResourcePlugin RD 연결 브로커(원격 데스크톱 연결 브로커)의 기능을 확장하는 메서드를 노출합니다.
ITsSbResourcePluginStore 리소스 플러그 인이 세션 및 대상과 같은 개체를 저장할 수 있도록 하는 메서드를 노출합니다.
ITsSbServiceNotification RD 연결 브로커(원격 데스크톱 연결 브로커)가 RD 연결 브로커 자체에서 발생하는 상태 변경 내용을 플러그 인에 알리는 데 사용하는 메서드를 노출합니다.
ITsSbSession 사용자 세션에 대한 정보를 저장하는 속성을 노출합니다.

ITsSbTarget 대상에 대한 구성 및 상태 정보를 저장하는 속성을 노출합니다.
ITsSbTargetPropertySet 이 인터페이스에서 파생하여 사용자 지정 대상 속성 집합을 정의합니다.
ITsSbTaskInfo 원격 데스크톱 연결 브로커가 플러그 인의 큐를 설정하는 데 사용하는 속성을 노출합니다.
ITsSbTaskPlugin 원격 데스크톱 연결 브로커 플러그 인에 대한 작업 큐를 업데이트하는 메서드를 노출합니다.
ITsSbTaskPluginNotifySink 작업에 대한 상태 및 오류 메시지를 보고하는 메서드를 RD 연결 브로커(원격 데스크톱 연결 브로커)에 노출합니다.
IWorkspace RemoteApp 및 데스크톱 연결의 연결에 대한 정보를 제공하는 메서드를 노출합니다.
IWorkspace2 RemoteApp 및 데스크톱 연결의 연결에 대한 정보를 제공하는 추가 메서드를 노출합니다.
IWorkspace3 RemoteApp 및 데스크톱 연결에서 연결에 대한 정보를 제공하는 메서드를 노출하고 클레임 토큰을 검색하거나 설정하는 기능을 추가합니다.
IWorkspaceClientExt 런타임이 RemoteApp 및 데스크톱 연결에서 사용자 지정 클라이언트의 연결을 끊을 수 있도록 하는 메서드를 노출합니다.
IWorkspaceRegistration RemoteApp 및 데스크톱 연결에서 사용자 지정 클라이언트에 대한 참조를 추가하고 제거하는 메서드를 노출합니다. (IWorkspaceRegistration)
IWorkspaceRegistration2 RemoteApp 및 데스크톱 연결에서 사용자 지정 클라이언트에 대한 참조를 추가하고 제거하는 메서드를 노출합니다. (IWorkspaceRegistration2)

IWorkspaceReportMessage 원격 작업 영역에 대한 오류 메시지 처리를 지원하는 메서드를 노출합니다.
IWorkspaceResTypeRegistry 플러그 인이 RemoteApp 및 데스크톱 연결 런타임에서 타사 파일 이름 확장명을 관리할 수 있도록 하는 메서드를 노출합니다.
IWorkspaceScriptable RemoteApp 및 데스크톱 연결 자격 증명 및 연결을 관리하는 메서드를 노출합니다. (IWorkspaceScriptable)
IWorkspaceScriptable2 RemoteApp 및 데스크톱 연결 자격 증명 및 연결을 관리하는 메서드를 노출합니다. (IWorkspaceScriptable2)
IWorkspaceScriptable3 RemoteApp 및 데스크톱 연결 자격 증명 및 연결을 관리하는 메서드를 노출합니다. (IWorkspaceScriptable3)
IWRdsEnhancedFastReconnectArbitrator 원격 데스크톱 서비스 서비스에서 호출한 메서드를 노출하여 향상된 빠른 다시 연결 시퀀스에서 다시 연결할 세션 ID를 가져옵니다.
IWRdsGraphicsChannel 이 인터페이스는 RemoteFX 그래픽 서비스에서 가상 그래픽 채널로 데이터를 보내고 받는 데 사용됩니다.
IWRdsGraphicsChannelEvents 이 인터페이스는 그래픽 가상 채널과 관련된 알림을 받습니다.
IWRdsGraphicsChannelManager 이 인터페이스는 RemoteFX 그래픽 서비스 API에서 그래픽 데이터 원격에 필요한 그래픽 가상 채널을 만드는 데 사용됩니다.
IWRdsProtocolConnection 클라이언트 연결을 구성하기 위해 원격 데스크톱 서비스 서비스에서 호출한 메서드를 노출합니다.

IWRdsProtocolConnectionCallback 클라이언트 연결의 상태에 대한 정보를 제공하고 클라이언트에 대한 작업을 수행하는 메서드를 노출합니다. 이 인터페이스는 원격 데스크톱 서비스 서비스에 의해 구현되고 프로토콜에 의해 호출됩니다.
IWRdsProtocolLicenseConnection 원격 데스크톱 서비스 서비스에서 연결 시퀀스 중에 라이선스 핸드셰이크를 수행하는 데 사용하는 메서드를 노출합니다.
IWRdsProtocolListener 프로토콜이 클라이언트 연결 요청 수신 대기를 시작하고 중지할 것을 요청하는 메서드를 노출합니다.
IWRdsProtocolListenerCallback 클라이언트가 연결되었음을 원격 데스크톱 서비스 서비스에 알리는 메서드를 노출합니다.
IWRdsProtocolLogonErrorRedirector 원격 데스크톱 서비스 서비스에서 로그온 상태를 업데이트하고 로그온 오류 메시지를 보내는 방법을 결정하는 메서드를 노출합니다.
IWRdsProtocolManager 원격 데스크톱 서비스 서비스에서 프로토콜 공급자와 통신하는 데 사용하는 메서드를 노출합니다.
IWRdsProtocolSettings 정책 관련 설정을 검색하고 추가하는 메서드를 노출합니다.
IWRdsProtocolShadowCallback 원격 데스크톱 서비스 서비스에 그림자의 대상 쪽을 시작하거나 중지하도록 알리기 위해 프로토콜에서 호출한 메서드를 노출합니다.
IWRdsProtocolShadowConnection 프로토콜 공급자에게 세션 섀도링 상태를 알리는 메서드를 노출합니다.
IWRdsWddmIddProps 이 인터페이스를 사용하면 사용자 지정 IDD 드라이버를 원격 세션에 로드할 수 있습니다.
IWTSBitmapRenderer 동적 가상 채널 플러그 인에서 비트맵을 렌더링하는 데 사용됩니다.

IWTSBitmapRendererCallback 동적 가상 채널 플러그 인은 렌더링 영역의 크기가 변경될 때 알림을 받을 이 인터페이스를 구현합니다.
IWTSBitmapRenderService 이 서비스는 서버의 매핑된 창에 해당하는 클라이언트에서 시각적 매핑을 만드는 데 사용됩니다.
IWTSListener DVC(동적 가상 채널) 연결에 대한 각 수신기에 대한 구성 설정을 관리합니다.
IWTSListenerCallback 특정 수신기에서 들어오는 요청에 대해 RDC(원격 데스크톱 연결) 클라이언트 플러그 인에 알리는 데 사용됩니다.
IWTSPlugin RDC(원격 데스크톱 연결) 클라이언트 플러그 인을 RDC(원격 데스크톱 연결) 클라이언트에서 로드할 수 있습니다.
IWTSPluginServiceProvider 동적 가상 채널 플러그 인이 다양한 원격 데스크톱 클라이언트 서비스를 쿼리하는 방법을 제공합니다.
IWTSProtocolConnection IWTSProtocolConnection을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnection을 사용합니다.
IWTSProtocolConnectionCallback IWTSProtocolConnectionCallback을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnectionCallback을 사용합니다.
IWTSProtocolLicenseConnection IWTSProtocolLicenseConnection을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolLicenseConnection을 사용합니다.
IWTSProtocolListener IWTSProtocolListener를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolListener를 사용합니다.

IWTSProtocolListenerCallback IWTSProtocolListenerCallback을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolListenerCallback을 사용합니다.
IWTSProtocolLogonErrorRedirector IWTSProtocolLogonErrorRedirector를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolLogonErrorRedirector를 사용합니다.
IWTSProtocolManager IWTSProtocolManager를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolManager를 사용합니다.
IWTSProtocolShadowCallback IWTSProtocolShadowCallback을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolShadowCallback을 사용합니다.
IWTSProtocolShadowConnection IWTSProtocolShadowConnection을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolShadowConnection을 사용합니다.
IWTSSBPlugin 터미널 서비스 세션 브로커(TS 세션 브로커)의 기능을 확장하는 데 사용됩니다. TS Session Broker의 리디렉션 논리를 재정의하는 플러그 인을 제공하려는 경우 이 인터페이스를 구현합니다.
IWTSVirtualChannel 채널 상태를 제어하고 채널에 쓰는 데 사용됩니다.
IWTSVirtualChannelCallback 채널 상태 변경 또는 수신된 데이터에 대한 알림을 받습니다.
IWTSVirtualChannelManager 모든 RDC(원격 데스크톱 연결) 클라이언트 플러그 인 및 동적 DVC(가상 채널) 수신기를 관리합니다.

구조체

__WTS_PROPERTY_VALUE 프로토콜에서 검색할 속성 값에 대한 정보를 포함합니다.
_WTS_CACHE_STATS 프로토콜 캐시 통계를 포함합니다.
_WTS_CACHE_STATS_UN 캐시 통계를 포함합니다.
_WTS_CLIENT_DATA 클라이언트 연결에 대한 정보를 포함합니다.
_WTS_DISPLAY_IOCTL 클라이언트 디스플레이에 대한 정보를 포함합니다.
_WTS_LICENSE_CAPABILITIES 클라이언트의 라이선스 기능에 대한 정보를 포함합니다.
_WTS_PROTOCOL_CACHE 캐시 읽기 및 캐시 적중 횟수를 포함합니다.
_WTS_PROTOCOL_COUNTERS 프로토콜 성능 카운터를 포함합니다.
_WTS_PROTOCOL_STATUS 프로토콜의 상태에 대한 정보를 포함합니다.
_WTS_SERVICE_STATE 원격 데스크톱 서비스 서비스의 상태 변경에 대한 정보를 포함합니다.
_WTS_SESSION_ID 세션을 고유하게 식별하는 GUID를 포함합니다.
_WTS_SMALL_RECT 클라이언트 창 좌표를 포함합니다.

_WTS_SOCKADDR 소켓 주소를 포함합니다.
_WTS_SYSTEMTIME 표준 시간과 일광 절약 시간 사이의 전환 날짜 및 시간 정보를 지정합니다.
_WTS_TIME_ZONE_INFORMATION 클라이언트 표준 시간대 정보를 포함합니다.
_WTS_USER_CREDENTIAL 사용자에 대한 자격 증명 정보를 포함합니다.
AAAccountingData 이 구조에는 연결 이벤트에 대한 정보가 포함됩니다.
AE_CURRENT_POSITION 디바이스에서 클라이언트로 현재 프레임 위치를 보고합니다.
BITMAP_RENDERER_STATISTICS RemoteFX 미디어 리디렉션 비트맵 렌더러에 대한 통계를 포함합니다.
CHANNEL_ENTRY_POINTS 가상 채널에 액세스하기 위해 클라이언트 쪽 DLL에서 호출하는 함수에 대한 포인터를 포함합니다.
CHANNEL_PDU_HEADER 가상 채널의 서버 끝에서 수신되는 데이터 블록에 대한 정보를 포함합니다.
CLIENT_DISPLAY RDC(원격 데스크톱 연결) 클라이언트의 표시에 대한 정보를 포함합니다. (CLIENT_DISPLAY)
pluginResource RemoteApp 및 데스크톱 연결에서 사용자에게 할당할 수 있는 리소스에 대한 정보를 포함합니다.
pluginResource2 RemoteApp 및 데스크톱 연결에서 사용자에게 할당할 수 있는 리소스에 대한 추가 정보를 포함합니다.

pluginResource2FileAssociation RemoteApp 및 데스크톱 연결의 파일 연결에 대한 정보를 포함합니다.
TSSD_ConnectionPoint 대상의 IP 주소를 정의합니다.
WRDS_CONNECTION_SETTING 원격 데스크톱 연결에 대한 다양한 수준의 설정을 포함합니다.
WRDS_CONNECTION_SETTINGS 원격 세션에 대한 연결 설정 정보를 포함합니다. (WRDS_CONNECTION_SETTINGS)
WRDS_CONNECTION_SETTINGS_1 원격 세션에 대한 연결 설정 정보를 포함합니다. (WRDS_CONNECTION_SETTINGS_1)
WRDS_DYNAMIC_TIME_ZONE_INFORMATION 동적 표준 시간대 정보를 포함합니다.
WRDS_LISTENER_SETTING 원격 데스크톱 연결에 대한 다양한 수준의 수신기 설정을 포함합니다.
WRDS_LISTENER_SETTINGS 원격 세션에 대한 수신기 설정 정보를 포함합니다.
WRDS_LISTENER_SETTINGS_1 원격 세션에 대한 수신기 설정을 포함합니다.
WRDS_SETTING 원격 데스크톱 연결에 대한 다양한 수준의 정책 관련 설정을 포함합니다.
WRDS_SETTINGS 원격 세션에 대한 정책 관련 설정 정보를 포함합니다.
WRDS_SETTINGS_1 원격 세션에 대한 정책 관련 설정을 포함합니다.

WTS_CLIENT_ADDRESS 원격 데스크톱 서비스 세션의 클라이언트 네트워크 주소를 포함합니다.
WTS_CLIENT_DISPLAY RDC(원격 데스크톱 연결) 클라이언트의 표시에 대한 정보를 포함합니다. (WTS_CLIENT_DISPLAY)
WTS_POLICY_DATA 원격 데스크톱 서비스 서비스에서 프로토콜에 전달하는 정책 정보를 포함합니다.
WTS_PROCESS_INFO_EXA RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 실행되는 프로세스에 대한 확장된 정보를 포함합니다. (ANSI)
WTS_PROCESS_INFO_EXW RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 실행되는 프로세스에 대한 확장된 정보를 포함합니다. (유니코드)
WTS_PROCESS_INFOA RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 실행되는 프로세스에 대한 정보를 포함합니다. (ANSI)
WTS_PROCESS_INFOW RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 실행되는 프로세스에 대한 정보를 포함합니다. (유니코드)
WTS_SERVER_INFOA 특정 원격 데스크톱 서비스 서버에 대한 정보를 포함합니다. (ANSI)
WTS_SERVER_INFOW 특정 원격 데스크톱 서비스 서버에 대한 정보를 포함합니다. (유니코드)
WTS_SESSION_ADDRESS 세션에 할당된 가상 IP 주소를 포함합니다.
WTS_SESSION_INFO_1A RD 세션 호스트(원격 데스크톱 세션 호스트) 서버 또는 RD 가상화 호스트(원격 데스크톱 가상화 호스트) 서버의 클라이언트 세션에 대한 확장된 정보를 포함합니다. (ANSI)

WTS_SESSION_INFO_1W RD 세션 호스트(원격 데스크톱 세션 호스트) 서버 또는 RD 가상화 호스트(원격 데스크톱 가상화 호스트) 서버의 클라이언트 세션에 대한 확장된 정보를 포함합니다. (유니코드)
WTS_SESSION_INFOA RD 세션 호스트(원격 데스크톱 세션 호스트) 서버의 클라이언트 세션에 대한 정보를 포함합니다. (ANSI)
WTS_SESSION_INFOW RD 세션 호스트(원격 데스크톱 세션 호스트) 서버의 클라이언트 세션에 대한 정보를 포함합니다. (유니코드)
WTS_USER_DATA 선택 클라이언트 속성 값을 포함합니다.
WTSCLIENTA RDC(원격 데스크톱 연결) 클라이언트에 대한 정보를 포함합니다. (ANSI)
WTSCLIENTW RDC(원격 데스크톱 연결) 클라이언트에 대한 정보를 포함합니다. (유니코드)
WTSCONFIGINFOA 원격 데스크톱 서비스 세션에 대한 정보를 포함합니다. (WTSCONFIGINFOA)
WTSCONFIGINFOW 원격 데스크톱 서비스 세션에 대한 정보를 포함합니다. (WTSCONFIGINFOW)
WTSINFOA 원격 데스크톱 서비스 세션에 대한 정보를 포함합니다. (WTSINFOA)
WTSINFOEX_LEVEL_A 원격 데스크톱 서비스 세션에 대한 확장된 정보를 포함하는 WTSINFOEX_LEVEL1 구조를 포함합니다. (ANSI)
WTSINFOEX_LEVEL_W 원격 데스크톱 서비스 세션에 대한 확장된 정보를 포함하는 WTSINFOEX_LEVEL1 구조를 포함합니다. (유니코드)

WTSINFOEX_LEVEL1_A 원격 데스크톱 서비스 세션에 대한 확장된 정보를 포함합니다. (ANSI)
WTSINFOEX_LEVEL1_W 원격 데스크톱 서비스 세션에 대한 확장된 정보를 포함합니다. (유니코드)
WTSINFOEXA 원격 데스크톱 서비스 세션에 대한 확장된 정보를 포함하는 WTSINFOEX_LEVEL 공용 구조체를 포함합니다. (ANSI)
WTSINFOEXW 원격 데스크톱 서비스 세션에 대한 확장된 정보를 포함하는 WTSINFOEX_LEVEL 공용 구조체를 포함합니다. (유니코드)
WTSINFOW 원격 데스크톱 서비스 세션에 대한 정보를 포함합니다. (WTSINFOW)
WTSLISTENERCONFIGA 원격 데스크톱 서비스 수신기에 대한 정보를 포함합니다. (ANSI)
WTSLISTENERCONFIGW 원격 데스크톱 서비스 수신기에 대한 정보를 포함합니다. (유니코드)
WTSSBX_IP_ADDRESS 네트워크 리소스의 IP 주소에 대한 정보를 포함합니다.
WTSSBX_MACHINE_CONNECT_INFO 원격 연결을 허용하는 컴퓨터에 대한 정보를 포함합니다.
WTSSBX_MACHINE_INFO 컴퓨터 및 현재 상태에 대한 정보를 포함합니다.
WTSSBX_SESSION_INFO 원격 데스크톱 연결 브로커(RD 연결 브로커)에서 사용할 수 있는 세션에 대한 정보를 포함합니다.

WTSSESSION_NOTIFICATION 세션 변경 알림에 대한 정보를 제공합니다. 서비스는 세션 변경 이벤트에 대한 응답으로 HandlerEx 함수에서 이 구조를 받습니다.
WTSUSERCONFIGA 도메인 컨트롤러 또는 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버의 사용자에 대한 구성 정보를 포함합니다. (ANSI)
WTSUSERCONFIGW 도메인 컨트롤러 또는 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버의 사용자에 대한 구성 정보를 포함합니다. (유니코드)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

원격 차등 압축

아티클2023. 03. 09.

원격 차등 압축 기술에 대한 개요입니다.

원격 차등 압축을 개발하려면 다음 헤더가 필요합니다.

- msrdc.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 원격 차등 압축

열거형

GeneratorParametersType 지원되는 생성기 매개 변수 집합을 정의합니다.
RDC_ErrorCode RDC 관련 오류 코드 집합을 정의합니다.
RdcCreatedTables 유사성 특성 테이블, 유사성 파일 ID 테이블 또는 둘 다의 상태를 설명하는 값을 정의합니다.
RdcMappingAccessMode RDC 파일 매핑 개체에 대한 액세스 모드 값을 정의합니다.
RdcNeedType 원격 복사본을 생성하는 데 사용되는 데이터 청크 집합을 정의합니다.

Functions

Append 유사성 파일의 테이블에 파일 ID 및 유사성 데이터 정보를 추가합니다.

Append 파일 ID를 유사성 파일 ID 테이블에 추가합니다.
Append SimilarityData 구조체를 유사성 특성 테이블에 추가합니다.
BeginDump 유사성 특성 테이블에서 유사성 데이터를 검색합니다.
CloseMapping 유사성 특성 테이블 파일에 대한 파일 매핑 개체를 닫습니다.
CloseTable 유사성 파일에서 테이블을 닫습니다.
CloseTable 유사성 파일 ID 테이블을 닫습니다.
CloseTable 유사성 특성 테이블을 닫습니다.
ComputeDefaultRecursionDepth 지정된 파일 크기에 대한 최대 재귀 수준을 계산합니다.
CopyAndSwap 기존 유사성 특성 테이블 및 기존 유사성 파일 ID 테이블의 복사본을 만들고, 내부 포인터를 교환하고, 기존 테이블을 삭제합니다.
CreateComparator 서명 비교자를 만듭니다.
CreateGenerator 지정된 수준의 서명을 생성하는 서명 생성기를 만듭니다.
CreateGeneratorParameters 서명 생성기에 필요한 매개 변수를 사용하여 초기화된 IRdcGeneratorParameters 인터페이스 포인터를 반환합니다.

CreateSignatureReader 애플리케이션에서 서명 파일의 내용을 디코딩할 수 있도록 서명 판독기를 만듭니다.
CreateTable 유사성 특성 테이블과 유사성 파일 ID 테이블을 만들거나 엽니다.
CreateTable 유사성 파일 ID 테이블을 만들거나 엽니다.
CreateTable 유사성 특성 테이블을 만들거나 엽니다.
CreateTableIndirect ISimilarityTraitsMapping 및 IRdcFileWriter 인터페이스의 RDC 애플리케이션 구현을 사용하여 유사성 특성 테이블 및 유사성 파일 ID 테이블을 만들거나 엽니다.
CreateTableIndirect RDC 애플리케이션의 IRdcFileWriter 인터페이스 구현을 사용하여 유사성 파일 ID 테이블을 만들거나 엽니다.
CreateTableIndirect RDC 애플리케이션의 ISimilarityTraitsMapping 인터페이스 구현을 사용하여 유사성 특성 테이블을 만들거나 엽니다.
CreateView 유사성 특성 테이블 파일에 대한 파일 매핑 보기를 매핑합니다.
DeleteOnClose 닫을 때 삭제(또는 잘리기)할 파일을 설정합니다.
EnableSimilarity 서명 생성기가 유사성 데이터를 생성할 수 있도록 합니다.
FindSimilarFileId 지정된 파일과 유사한 파일 목록을 반환합니다.

FindSimilarFileIndex 지정된 파일과 유사한 파일 목록을 반환합니다. 목록의 결과는 가장 유사한 파일부터 시작하여 유사성 순서로 정렬됩니다.
플러시 유사성 특성 테이블 파일의 매핑된 보기 내에서 더티 페이지를 디스크에 씁니다.
가져오기 유사성 특성 테이블 파일의 매핑된 뷰에 대한 정보를 반환합니다.
GetFilePosition 현재 파일 위치를 반환합니다.
GetFileSize 파일의 크기를 반환합니다.
GetFileSize 유사성 특성 테이블 파일의 크기를 반환합니다.
GetGeneratorParameters 생성기를 만드는 데 사용되는 매개 변수의 복사본을 반환합니다.
GetGeneratorParametersType 매개 변수의 특정 형식을 반환합니다.
GetHashWindowSize 로컬 최대 계산에 사용되는 해시를 계산하기 위해 FilterMax 생성기에서 사용하는 슬라이딩 윈도우 크기의 해시 창 크기를 반환합니다.
GetHorizonSize FilterMax 생성기가 로컬 최대값을 찾는 길이를 반환합니다.
GetLastIndex 유사성 특성 테이블에 저장된 마지막 항목의 인덱스를 검색합니다.

GetNextData ISimilarityTraitsTable::BeginDump 메서드에서 반환된 유사성 특성 목록에서 하나 이상의 SimilarityDumpData 구조를 검색합니다.
GetNextFileId ISimilarity::FindSimilarFileId 메서드에서 반환된 파일 목록에서 다음 유효한 유사성 파일 ID를 검색합니다.
GetPageSize 유사성 특성 테이블 파일의 페이지 크기(디스크 블록 크기)를 반환합니다.
GetParametersVersion 매개 변수를 serialize하는 데 사용되는 RDC 버전에 대한 정보를 반환합니다.
GetRDCVersion 설치된 RDC 런타임의 버전과 설치된 런타임에서 지원하는 가장 오래된 버전의 RDC 인터페이스를 검색합니다.
GetRecordCount 유사성 파일 ID 테이블에 저장된 레코드 수를 유사성 파일에서 검색합니다.
GetRecordCount 유사성 파일 ID 테이블에 저장된 레코드 수를 검색합니다.
GetSerializeSize serialize된 매개 변수 데이터의 크기(바이트)를 반환합니다.
GetSize ISimilarity::FindSimilarFileId 메서드에서 반환된 파일 목록의 항목 수를 검색합니다.
Getview 유사성 특성 테이블 파일의 매핑된 뷰에 대한 시작 및 끝 주소를 반환합니다.
Invalidate 파일 ID를 유사성 파일 ID 테이블에서 유효하지 않은 것으로 표시합니다.

조회 유사성 파일 ID 테이블에서 지정된 파일 인덱스에 해당하는 파일 ID를 검색합니다.
OpenGeneratorParameters 기존 직렬화된 매개 변수 블록을 열고 데이터로 초기화된 IRdcGeneratorParameters 인터페이스 포인터를 반환합니다.
OpenMapping 유사성 특성 테이블 파일에 대한 파일 매핑 개체를 엽니다.
처리 두 개의 서명 스트림(시드 및 원본)을 비교하고 대상 파일을 만드는 데 필요한 파일 데이터의 청크를 설명하는 필요 목록을 생성합니다.
처리 입력 데이터를 처리하고 0개 이상의 출력 바이트를 생성합니다.
읽기 지정된 위치에서 시작하는 지정된 양의 데이터를 읽습니다.
ReadHeader 서명 헤더를 읽고 서명을 생성하는 데 사용되는 매개 변수의 복사본을 반환합니다.
ReadSignatures 현재 위치에서 서명 블록을 읽습니다.
ReportProgress 진행 중인 유사성 작업의 현재 완료율을 보고합니다.
ResizeMapping 유사성 특성 테이블 파일에 대한 파일 매핑 개체의 크기를 조정합니다.
결과 서명 생성기에서 파일에 대해 생성된 유사성 데이터를 검색합니다.
직렬화 매개 변수 데이터를 메모리 블록으로 직렬화합니다.

SetFileSize 유사성 특성 테이블 파일의 크기를 설정합니다.
SetHashWindowSize 로컬 최대 계산에 사용되는 해시를 계산하기 위해 FilterMax 생성기에서 사용하는 슬라이딩 윈도우 크기를 해시 창 크기로 설정합니다.
SetHorizonSize FilterMax 생성기가 로컬 최대값을 찾는 길이를 수평선 크기를 설정합니다.
Truncate 파일을 길이가 0으로 자립니다.
Unmap 유사성 특성 테이블 파일의 매핑된 뷰 매핑을 해제합니다.
쓰기 지정된 오프셋에서 시작하는 파일에 바이트를 씁니다.

인터페이스

IFindSimilarResults ISimilarity::FindSimilarFileId 메서드에서 반환된 파일 목록에서 정보를 검색하는 메서드를 제공합니다.
IRdcComparator 두 개의 서명 스트림(시드 및 원본)을 비교하고 대상 파일을 만드는 데 필요한 원본 및 시드 파일 데이터 청크 목록을 생성하는 데 사용됩니다.
IRdcFileReader 동기화되는 데이터가 디스크에 파일로 존재하지 않을 수 있으므로 IRdcFileReader 인터페이스는 파일 핸들과 동등한 값을 제공하는 데 사용됩니다.
IRdcFileWriter 파일에서 읽고 파일에 쓸 추상 인터페이스입니다.

IRdcGenerator 입력 데이터를 처리하고 생성기에서 사용하는 매개 변수를 읽는 데 사용됩니다.
IRdcGeneratorFilterMaxParameters FilterMax 생성기에서 사용하는 매개 변수를 설정하고 검색합니다.
IRdcGeneratorParameters 모든 유형의 생성기 매개 변수에 대한 제네릭 인터페이스입니다. 모든 생성기 매개 변수 개체는 이 인터페이스를 지원해야 합니다.
IRdcLibrary RDC를 사용하기 위한 기본 인터페이스입니다.
IRdcSignatureReader 서명을 생성하는 데 사용되는 서명 및 매개 변수를 읽습니다.
IRdcSimilarityGenerator 서명 생성기가 유사성 데이터를 생성할 수 있도록 설정하고 생성된 후 유사성 데이터를 검색하는 메서드를 정의합니다.
ISimilarity 파일별 유사성 데이터 및 파일 ID를 유사성 파일에 저장하고 검색하는 메서드를 정의합니다.
ISimilarityFileIdTable 유사성 파일 ID 정보를 저장하고 검색하는 메서드를 정의합니다.
ISimilarityReportProgress 유사성 작업의 현재 완료율을 보고하는 RDC에 대한 메서드를 정의합니다.
ISimilarityTableDumpState ISimilarityTraitsTable::BeginDump 메서드에서 반환된 유사성 특성 목록에서 정보를 검색하는 메서드를 제공합니다.
ISimilarityTraitsMappedView RDC 애플리케이션이 유사성 특성 테이블 파일의 매핑된 뷰를 조작하기 위해 구현할 수 있는 메서드를 제공합니다.

ISimilarityTraitsMapping RDC 애플리케이션이 유사성 특성 테이블 파일에 대한 파일 매핑 개체를 만들고 조작하기 위해 구현할 수 있는 메서드를 제공합니다.
ISimilarityTraitsTable 파일별 유사성 데이터를 저장하고 유사성 조회를 수행하는 메서드를 정의합니다.

구조체

FindSimilarFileIndexResults ISimilarityTraitsTable::FindSimilarFileIndex 메서드가 일치하는 파일에 대해 반환한 파일 인덱스 정보를 포함합니다.
RdcBufferPointer 버퍼에 대해 설명합니다.
RdcNeed 두 데이터 집합을 동기화하는 데 필요한 청크에 대한 정보를 포함합니다.
RdcNeedPointer RdcNeed 구조체의 배열에 대해 설명합니다.
RdcSignature 단일 서명과 이를 생성하는 데 사용되는 청크의 길이를 포함합니다.
RdcSignaturePointer RdcSignature 구조의 배열에 대해 설명합니다.
SimilarityData 파일의 유사성 데이터를 포함합니다.
SimilarityDumpData ISimilarityTableDumpState::GetNextData 메서드에 의해 파일에 대해 반환된 유사성 정보를 포함합니다.

SimilarityFileId 파일의 유사성 파일 ID를 포함합니다.
SimilarityMappedViewInfo 유사성 매핑 보기에 대한 정보를 포함합니다.

RPC(원격 프로시저 호출)

아티클2023. 03. 16.

RPC(원격 프로시저 호출) 기술 개요입니다.

RPC(원격 프로시저 호출)를 개발하려면 다음 헤더가 필요합니다.

- midles.h
- rpc.h
- rpcasync.h
- rpcdcep.h
- rpcnsi.h
- rpcproxy.h
- rpcssl.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- RPC(원격 프로시저 호출)

열거형

RPC_ASYNC_EVENT RPC_ASYNC_EVENT 열거형 형식은 RPC 애플리케이션이 받을 수 있는 비동기 알림 이벤트를 설명합니다.
RPC_NOTIFICATION_TYPES RPC_NOTIFICATION_TYPES 열거형 형식에는 클라이언트 프로그램에서 사용할 비동기 알림 메서드를 지정하는 값이 포함되어 있습니다.
RPC_NOTIFICATIONS RPC_NOTIFICATIONS 열거형은 서버가 RPC에서 받을 수 있는 알림을 지정합니다.
RpcCallClientLocality 가능한 RPC 클라이언트 지역 집합을 지정합니다.
RpcCallType RPC 호출 형식 집합을 지정합니다.

RpcLocalAddressFormat RPC에서 지원하는 가능한 로컬 IP 주소 형식을 지정합니다.

Functions

CStdAsyncStubBuffer_AddRef IRpcStubBuffer::AddRef 메서드를 구현합니다.
CStdAsyncStubBuffer_Connect IRpcStubBuffer::Connect 메서드를 구현하고 서버 개체를 스텁에 연결합니다. (CStdAsyncStubBuffer_Connect)
CStdAsyncStubBuffer_Disconnect IRpcStubBuffer::D isconnect 메서드를 구현하고 스텁에서 서버 개체의 연결을 끊습니다. (CStdAsyncStubBuffer_Disconnect)
CStdAsyncStubBuffer_Invoke IRpcStubBuffer::Invoke 메서드를 구현합니다.
CStdAsyncStubBuffer_QueryInterface IRpcStubBuffer::QueryInterface 메서드를 구현합니다. (CStdAsyncStubBuffer_QueryInterface)
CStdAsyncStubBuffer_Release IRpcStubBuffer::Release 메서드를 구현합니다. (CStdAsyncStubBuffer_Release)
CStdAsyncStubBuffer2_Connect IRpcStubBuffer::Connect 메서드를 구현하고 서버 개체를 스텁에 연결합니다. (CStdAsyncStubBuffer2_Connect)
CStdAsyncStubBuffer2_Disconnect IRpcStubBuffer::D isconnect 메서드를 구현하고 스텁에서 서버 개체의 연결을 끊습니다. (CStdAsyncStubBuffer2_Disconnect)
CStdAsyncStubBuffer2_Release IRpcStubBuffer::Release 메서드를 구현합니다. (CStdAsyncStubBuffer2_Release)

CStdStubBuffer_AddRef CStdStubBuffer_AddRef 함수는 IRpcStubBuffer::AddRef 메서드를 구현합니다.
CStdStubBuffer_Connect CStdStubBuffer_Connect 함수는 IRpcStubBuffer::Connect 메서드를 구현하고 서버 개체를 스텁에 연결합니다.
CStdStubBuffer_CountRefs CStdStubBuffer_CountRefs 함수는 IRpcStubBuffer::CountRefs 메서드를 구현합니다.
CStdStubBuffer_DebugServerQueryInterface CStdStubBuffer_DebugServerQueryInterface 함수는 IRpcStubBuffer::D ebugServerQueryInterface 메서드를 구현합니다.
CStdStubBuffer_DebugServerRelease CStdStubBuffer_DebugServerRelease 함수는 IRpcStubBuffer::D ebugServerRelease 메서드를 구현합니다.
CStdStubBuffer_Disconnect CStdStubBuffer_Disconnect 함수는 IRpcStubBuffer::D isconnect 메서드를 구현하고 스텁에서 서버 개체의 연결을 끊습니다.
CStdStubBuffer_Invoke CStdStubBuffer_Invoke 함수는 IRpcStubBuffer::Invoke 메서드를 구현합니다.
CStdStubBuffer_IsIIDSupported CStdStubBuffer_IsIIDSupported 함수는 IRpcStubBuffer::IsIIDSupported 메서드를 구현합니다.
CStdStubBuffer_QueryInterface CStdStubBuffer_QueryInterface 함수는 IRpcStubBuffer::QueryInterface 메서드를 구현합니다.
CStdStubBuffer2_Connect IRpcStubBuffer::Connect 메서드를 구현하고 서버 개체를 스텁에 연결합니다. (CStdStubBuffer2_Connect)
CStdStubBuffer2_CountRefs IRpcStubBuffer::CountRefs 메서드를 구현합니다.

CStdStubBuffer2_Disconnect IRpcStubBuffer::D isconnect 메서드를 구현하고 스텁에서 서버 개체의 연결을 끊습니다. (CStdStubBuffer2_Disconnect)
CStdStubBuffer2_QueryInterface IRpcStubBuffer::QueryInterface 메서드를 구현합니다. (CStdStubBuffer2_QueryInterface)
DceErrorInqText DceErrorInqText 함수(rpcdce.h)는 상태 코드에 대한 메시지 텍스트를 반환합니다.
DceErrorInqTextA DceErrorInqText 함수는 상태 코드에 대한 메시지 텍스트를 반환합니다. (DceErrorInqTextA)
DceErrorInqTextW DceErrorInqTextW(유니코드) 함수(rpcdce.h)는 상태 코드에 대한 메시지 텍스트를 반환합니다.
I_RpcBindingInqLocalClientPID 클라이언트 프로세스 ID를 가져옵니다.
I_RpcExceptionFilter 예외가 치명적인지 아니면 치명적이 아닌지 확인합니다.
IUnknown_AddRef_Proxy IUnknown_AddRef_Proxy 함수는 모든 인터페이스 프록시에 대해 AddRef 메서드를 구현합니다.
IUnknown_QueryInterface_Proxy IUnknown_QueryInterface_Proxy 함수는 모든 인터페이스 프록시에 대해 QueryInterface 메서드를 구현합니다.
IUnknown_Release_Proxy IUnknown_Release_Proxy 함수는 모든 인터페이스 프록시에 대해 Release 메서드를 구현합니다.
MesBufferHandleReset MesBufferHandleReset 함수는 버퍼 serialization에 대한 핸들을 다시 초기화합니다.

MesDecodeBufferHandleCreate MesDecodeBufferHandleCreate 함수는 디코딩 핸들을 만들고 serialization의 (고정) 버퍼 스타일에 맞게 초기화합니다.
MesDecodeIncrementalHandleCreate MesDecodeIncrementalHandleCreate 함수는 serialization의 증분 스타일에 대한 디코딩 핸들을 만듭니다.
MesEncodeDynBufferHandleCreate MesEncodeDynBufferHandleCreate 함수는 인코딩 핸들을 만든 다음 동적 버퍼 스타일의 serialization을 위해 초기화합니다.
MesEncodeFixedBufferHandleCreate MesEncodeFixedBufferHandleCreate 함수는 인코딩 핸들을 만든 다음 직렬화의 고정 버퍼 스타일을 위해 초기화합니다.
MesEncodeIncrementalHandleCreate MesEncodeIncrementalHandleCreate 함수는 인코딩을 만든 다음 serialization의 증분 스타일에 맞게 초기화합니다.
MesHandleFree MesHandleFree 함수는 serialization 핸들에 의해 할당된 메모리를 해제합니다.
MesIncrementalHandleReset MesIncrementalHandleReset 함수는 증분 serialization에 대한 핸들을 다시 초기화합니다.
MesInqProcEncodingId MesInqProcEncodingId 함수는 인코딩의 ID를 제공합니다.
Ndr64AsyncClientCall Ndr64AsyncClientCall은 애플리케이션에서 직접 호출할 수 없습니다.
Ndr64AsyncServerCallAll Ndr64AsyncServerCallAll은 애플리케이션에서 직접 호출할 수 없습니다.
NdrAsyncClientCall NdrAsyncClientCall 함수는 /Oi 및 /Oic 모드 스텁의 비동기 클라이언트 쪽 진입점입니다.

NdrAsyncClientCall2 NdrAsyncClientCall2를 변경하거나 사용할 수 없습니다.
NdrAsyncServerCall NdrAsyncServerCall은 애플리케이션에서 직접 호출할 수 없습니다.
NdrClearOutParameters NdrClearOutParameters 함수는 out 매개 변수의 리소스를 해제하고 서버에 대한 RPC 호출이 실패하면 메모리를 지웁니다.
NdrClientCall NdrClientCall 함수는 /Oicf 모드 스텁의 클라이언트 쪽 진입점입니다.
NdrClientCall2 NdrClientCall2 함수는 /Oicf 모드 스텁의 클라이언트 쪽 진입점입니다.
NdrClientCall3 NdrClientCall3을 변경하거나 사용할 수 없습니다.
NdrClientCall4 NdrClientCall4를 변경하거나 사용할 수 없습니다.
NdrConformantArrayUnmarshall NdrConformantArrayUnmarshall 함수는 규격 배열을 unmarshalshals합니다.
NdrConformantStringBufferSize NdrConformantStringBufferSize 함수는 규칙 문자열을 마샬링하는 데 필요한 버퍼의 크기를 바이트 단위로 계산합니다.
NdrConformantStringMarshall NdrConformantStringMarshall 함수는 규칙 문자열을 서버로 보낼 네트워크 버퍼로 마샬링합니다.
NdrConformantStringUnmarshall NdrConformantStringUnmarshall 함수는 네트워크 버퍼에서 메모리로 규칙 문자열을 unmarshalshals합니다.

NdrContextHandleInitialize 새 RPC 컨텍스트 핸들을 초기화합니다.
NdrContextHandleMemorySize 로컬 메모리에 표시된 대로 제공된 RPC 컨텍스트 핸들의 크기를 반환합니다.
NdrContextHandleSize NdrContextHandleSize 함수는 제공된 RPC 컨텍스트 핸들의 크기를 반환합니다.
NdrConvert NdrConvert 함수는 네트워크 버퍼를 보낸 사람의 데이터 표현에서 받는 사람의 데이터 표현(다른 경우)으로 변환합니다.
NdrCStdStubBuffer_Release NdrCStdStubBuffer_Release 함수는 IRpcStubBuffer::Release 메서드를 구현합니다.
NdrCStdStubBuffer2_Release NdrCStdStubBuffer2_Release 함수는 IRpcStubBuffer::Release 메서드를 구현합니다.
NdrDcomAsyncClientCall NdrDcomAsyncClientCall을 변경하거나 사용할 수 없습니다.
NdrDcomAsyncClientCall2 NdrDcomAsyncClientCall2를 변경하거나 사용할 수 없습니다.
NdrDllCanUnloadNow NdrDllCanUnloadNow 함수는 프록시 및 스텁을 구현하는 DLL이 여전히 사용 중인지 여부를 결정합니다. 그렇지 않은 경우 호출자는 메모리에서 DLL을 안전하게 언로드할 수 있습니다.
NdrDllGetClassObject NdrDllGetClassObject 함수는 프록시 또는 스텁의 클래스 개체를 검색합니다.
NdrDllRegisterProxy NdrDllRegisterProxy 함수는 프록시 DLL에 포함된 인터페이스에 대한 레지스트리 항목을 만듭니다.

NdrDllUnregisterProxy NdrDllUnregisterProxy 함수는 프록시 DLL에 포함된 인터페이스에 대한 레지스트리 항목을 제거합니다.
NdrGetUserMarshalInfo NdrGetUserMarshalInfo 함수는 도우미 함수를 wire_marshal user_marshal 추가 정보를 제공합니다.
NdrInterfacePointerBufferSize NdrInterfacePointerBufferSize 함수는 인터페이스 포인터를 마샬링하는 데 필요한 버퍼의 크기를 바이트 단위로 계산합니다.
NdrInterfacePointerFree NdrInterfacePointerFree 함수는 인터페이스 포인터를 해제합니다.
NdrInterfacePointerMarshall NdrInterfacePointerMarshall 함수는 인터페이스 포인터를 서버로 보낼 네트워크 버퍼로 마샬링합니다.
NdrInterfacePointerUnmarshall NdrInterfacePointerUnmarshall 함수는 인터페이스 포인터에서 참조하는 데이터를 네트워크 버퍼에서 메모리로 분리합니다.
NdrMesProcEncodeDecode2 NdrMesProcEncodeDecode2를 변경하거나 사용할 수 없습니다.
NdrMesProcEncodeDecode4 NdrMesProcEncodeDecode4를 변경하거나 사용할 수 없습니다.
NdrOleAllocate NdrOleAllocate 함수는 RPC에서 개체 인터페이스에 대한 메모리를 할당하는 데 사용됩니다. 이 함수는 CoTaskMemAlloc 함수에 대한 래퍼입니다.
NdrOleFree NdrOleFree 함수는 CoTaskMemFree 함수에 대한 래퍼입니다.

NdrPointerBufferSize NdrPointerBufferSize 함수는 모든 항목에 대한 최상위 포인터에 필요한 버퍼 크기를 바이트 단위로 계산합니다.
NdrPointerFree NdrPointerFree 함수는 메모리를 해제합니다.
NdrPointerMarshall NdrPointerMarshall 함수는 모든 항목에 대한 최상위 포인터를 마샬링합니다. 구조체, 배열 또는 공용 구조체에 포함된 포인터는 NdrPointerMarshall을 직접 호출합니다.
NdrPointerUnmarshall NdrPointerUnmarshall 함수는 모든 항목에 대한 최상위 포인터를 unmarshallss합니다. 구조체, 배열 또는 공용 구조체에 포함된 포인터는 NdrPointerUnmarshall을 직접 호출합니다.
NdrProxyErrorHandler NdrProxyErrorHandler 함수는 RPC 기능 코드를 사용하여 예외를 HRESULT에 매핑합니다.
NdrProxyForwardingFunction10 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction10)
NdrProxyForwardingFunction11 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction11)
NdrProxyForwardingFunction12 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction12)
NdrProxyForwardingFunction13 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction13)
NdrProxyForwardingFunction14 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction14)
NdrProxyForwardingFunction15 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction15)

NdrProxyForwardingFunction16 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction16)
NdrProxyForwardingFunction17 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction17)
NdrProxyForwardingFunction18 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction18)
NdrProxyForwardingFunction19 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction19)
NdrProxyForwardingFunction20 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction20)
NdrProxyForwardingFunction21 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction21)
NdrProxyForwardingFunction22 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction22)
NdrProxyForwardingFunction23 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction23)
NdrProxyForwardingFunction24 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction24)
NdrProxyForwardingFunction25 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction25)
NdrProxyForwardingFunction26 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction26)
NdrProxyForwardingFunction27 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction27)

NdrProxyForwardingFunction28 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction28)
NdrProxyForwardingFunction29 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction29)
NdrProxyForwardingFunction3 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction3)
NdrProxyForwardingFunction30 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction30)
NdrProxyForwardingFunction31 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction31)
NdrProxyForwardingFunction32 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction32)
NdrProxyForwardingFunction4 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction4)
NdrProxyForwardingFunction5 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction5)
NdrProxyForwardingFunction6 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction6)
NdrProxyForwardingFunction7 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction7)
NdrProxyForwardingFunction8 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction8)
NdrProxyForwardingFunction9 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction9)

NdrProxyFreeBuffer NdrProxyFreeBuffer 함수는 RPC 버퍼를 해제합니다.
NdrProxyGetBuffer NdrProxyGetBuffer 함수는 IRpcChannelBuffer::GetBuffer 메서드 호출을 사용하여 COM에서 네트워크 버퍼를 가져옵니다.
NdrProxyInitialize NdrProxyInitialize 함수는 개체 메서드에 대한 프록시를 초기화합니다.
NdrProxySendReceive NdrProxySendReceive 함수는 서버에 메시지를 보낸 다음, 회신을 기다립니다.
NdrServerCall2 NdrServerCall2는 애플리케이션에서 직접 호출할 수 없습니다.
NdrServerCallAll NdrServerCallAll은 애플리케이션에서 직접 호출할 수 없습니다.
NdrSimpleTypeMarshall NdrSimpleTypeMarshall 함수는 단순 형식을 마샬링합니다.
NdrSimpleTypeUnmarshall NdrSimpleTypeUnmarshall 함수는 단순 형식을 unmarshalls합니다.
NdrStubCall2 NdrStubCall2 함수는 /Oicf 모드 스텁의 서버 쪽 진입점입니다.
NdrStubCall3 NdrStubCall3은 애플리케이션에서 직접 호출할 수 없습니다.
NdrStubForwardingFunction NdrStubForwardingFunction 함수는 기본 인터페이스에 정의된 서버 쪽 개체 메서드의 진입점입니다.
NdrStubGetBuffer NdrStubGetBuffer 함수는 RPC 채널에서 버퍼를 검색합니다.

NdrStubInitialize NdrStubInitialize 함수는 경계를 풀기 전에 서버 스텁에서 호출됩니다. 일부 스텁 메시지 필드를 설정합니다.
NdrUserMarshalBufferSize NdrUserMarshalBufferSize 함수는 사용자 마샬링 개체를 마샬링하는 데 필요한 버퍼의 크기를 바이트 단위로 계산합니다.
NdrUserMarshalFree NdrUserMarshalFree 함수는 사용자 마샬링 개체를 해제합니다.
NdrUserMarshalMarshall NdrUserMarshalMarshall 함수는 제공된 데이터 버퍼를 마샬링합니다.
ObjectStublessClient10 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient10)
ObjectStublessClient11 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient11)
ObjectStublessClient12 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient12)
ObjectStublessClient13 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient13)
ObjectStublessClient14 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient14)
ObjectStublessClient15 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient15)
ObjectStublessClient16 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient16)
ObjectStublessClient17 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient17)

ObjectStublessClient18 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient18)
ObjectStublessClient19 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient19)
ObjectStublessClient20 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient20)
ObjectStublessClient21 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient21)
ObjectStublessClient22 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient22)
ObjectStublessClient23 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient23)
ObjectStublessClient24 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient24)
ObjectStublessClient25 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient25)
ObjectStublessClient26 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient26)
ObjectStublessClient27 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient27)
ObjectStublessClient28 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient28)
ObjectStublessClient29 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient29)

ObjectStublessClient3 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient3)
ObjectStublessClient30 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient30)
ObjectStublessClient31 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient31)
ObjectStublessClient32 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient32)
ObjectStublessClient4 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient4)
ObjectStublessClient5 ObjectStublessClient5 함수는 COM 프록시에 대한 스텁 함수입니다.
ObjectStublessClient6 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient6)
ObjectStublessClient7 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient7)
ObjectStublessClient8 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient8)
ObjectStublessClient9 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient9)
RPC_AUTH_KEY_RETRIEVAL_FN RPC_AUTH_KEY_RETRIEVAL_FN 함수는 암호화 키를 반환하는 서버 애플리케이션 제공 루틴의 주소를 지정하는 함수의 프로토타입입니다.
RPC_IF_CALLBACK_FN RPC_IF_CALLBACK_FN 애플리케이션에서 제공하는 보안 콜백 함수의 프로토타입입니다. 프로그램에서 정의하는 각 인터페이스에 대한 콜백 함수를 제공할 수 있습니다.

RPC_INTERFACE_GROUP_IDLE_CALLBACK_FN RPC_INTERFACE_GROUP_IDLE_CALLBACK_FN 정의된 각 인터페이스 그룹에 대해 구현할 수 있는 사용자 정의 콜백입니다. 이 콜백은 인터페이스 그룹의 유휴 상태가 변경되었음을 감지할 때 RPC 런타임에 의해 호출됩니다.
RPC_MGMT_AUTHORIZATION_FN 이 RPC_MGMT_AUTHORIZATION_FN 통해 서버 프로그램에서 사용자 지정 RPC 권한 부여 기술을 구현할 수 있습니다.
RPC_OBJECT_INQ_FN RPC_OBJECT_INQ_FN 함수는 기본 개체 UUID를 UUID 매핑 형식으로 쉽게 대체할 수 있는 함수의 프로토타입입니다.
RpcAsyncAbortCall 서버는 RpcAsyncAbortCall을 호출하여 비동기 호출을 중단합니다.
RpcAsyncCancelCall 클라이언트는 RpcAsyncCancelCall 함수를 호출하여 비동기 호출을 취소합니다.
RpcAsyncCompleteCall 클라이언트와 서버는 RpcAsyncCompleteCall 함수를 호출하여 비동기 원격 프로시저 호출을 완료합니다.
RpcAsyncGetCallHandle RpcAsyncGetCallHandle 매크로는 비동기 원격 프로시저 호출에서 바인딩 핸들을 반환합니다.
RpcAsyncGetCallStatus 클라이언트는 RpcAsyncGetCallStatus 함수를 호출하여 비동기 원격 호출의 현재 상태를 확인합니다.
RpcAsyncInitializeHandle 클라이언트는 RpcAsyncInitializeHandle 함수를 호출하여 비동기 호출에 사용할 RPC_ASYNC_STATE 구조를 초기화합니다.
RpcAsyncRegisterInfo RpcAsyncRegisterInfo 함수는 사용되지 않습니다.

RpcBindingBind RpcBindingBind 함수는 RPC 서버에 연결하고 바인딩합니다.
RpcBindingCopy RpcBindingCopy 함수는 바인딩 정보를 복사하고 새 바인딩 핸들을 만듭니다.
RpcBindingCreateA RpcBindingCreate 함수는 제공된 템플릿을 기반으로 새로운 빠른 RPC 바인딩 핸들을 만듭니다. (ANSI)
RpcBindingCreateW RpcBindingCreate 함수는 제공된 템플릿을 기반으로 새로운 빠른 RPC 바인딩 핸들을 만듭니다. (유니코드)
RpcBindingFree RpcBindingFree 함수는 바인딩 핸들 리소스를 해제합니다.
RpcBindingFromStringBinding RpcBindingFromStringBinding 함수(rpcdce.h)는 바인딩 핸들의 문자열 표현에서 바인딩 핸들을 반환합니다.
RpcBindingFromStringBindingA 바인딩 핸들의 문자열 표현에서 바인딩 핸들을 반환합니다. (RpcBindingFromStringBindingA)
RpcBindingFromStringBindingW RpcBindingFromStringBindingW(유니코드) 함수(rpcdce.h)는 바인딩 핸들의 문자열 표현에서 바인딩 핸들을 반환합니다.
RpcBindingInqAuthClient RpcBindingInqAuthClient 함수(rpcdce.h)는 원격 프로시저 호출을 수행한 인증된 클라이언트의 주체 이름 또는 권한 특성을 가져옵니다.
RpcBindingInqAuthClientA 서버 애플리케이션은 RpcBindingInqAuthClient 함수를 호출하여 원격 프로시저 호출을 수행한 인증된 클라이언트의 보안 주체 이름 또는 권한 특성을 가져옵니다. (RpcBindingInqAuthClientA)

RpcBindingInqAuthClientEx RpcBindingInqAuthClientEx 함수(rpcdce.h)는 원격 프로시저 호출을 수행한 클라이언트 프로그램에 대한 확장된 정보를 가져옵니다.
RpcBindingInqAuthClientExA 서버 애플리케이션은 RpcBindingInqAuthClientEx 함수를 호출하여 원격 프로시저를 호출한 클라이언트 프로그램에 대한 확장된 정보를 가져옵니다. (RpcBindingInqAuthClientExA)
RpcBindingInqAuthClientExW RpcBindingInqAuthClientExW(유니코드) 함수(rpcdce.h)는 원격 프로시저 호출을 수행한 클라이언트 프로그램에 대한 확장된 정보를 가져옵니다.
RpcBindingInqAuthClientW RpcBindingInqAuthClientW(유니코드) 함수(rpcdce.h)는 원격 프로시저 호출을 수행한 클라이언트의 보안 주체 이름 또는 권한 특성을 가져옵니다.
RpcBindingInqAuthInfo RpcBindingInqAuthInfo 함수(rpcdce.h)는 바인딩 핸들에서 인증 및 권한 부여 정보를 반환합니다.
RpcBindingInqAuthInfoA RpcBindingInqAuthInfo 함수는 바인딩 핸들에서 인증 및 권한 부여 정보를 반환합니다. (RpcBindingInqAuthInfoA)
RpcBindingInqAuthInfoExA RpcBindingInqAuthInfoEx 함수는 바인딩 핸들에서 인증, 권한 부여 및 보안 서비스 품질 정보를 반환합니다. (ANSI)
RpcBindingInqAuthInfoExW RpcBindingInqAuthInfoEx 함수는 바인딩 핸들에서 인증, 권한 부여 및 보안 서비스 품질 정보를 반환합니다. (유니코드)
RpcBindingInqAuthInfoW RpcBindingInqAuthInfoW(유니코드) 함수(rpcdce.h)는 바인딩 핸들에서 인증 및 권한 부여 정보를 반환합니다.
RpcBindingInqObject RpcBindingInqObject 함수는 바인딩 핸들에서 개체 UUID를 반환합니다.

RpcBindingInqOption RPC 클라이언트 프로세스는 RpcBindingInqOption을 사용하여 지정된 바인딩 핸들에 대한 바인딩 옵션의 현재 값을 결정합니다.
RpcBindingReset RpcBindingReset 함수는 호스트가 지정되었지만 해당 호스트의 서버가 지정되지 않도록 바인딩 핸들을 다시 설정합니다.
RpcBindingServerFromClient 애플리케이션은 RpcBindingServerFromClient를 호출하여 클라이언트 바인딩 핸들을 부분적으로 바인딩된 서버 바인딩 핸들로 변환합니다.
RpcBindingSetAuthInfo RpcBindingSetAuthInfo 함수(rpcdce.h)는 바인딩 핸들의 인증 및 권한 부여 정보를 설정합니다.
RpcBindingSetAuthInfoA RpcBindingSetAuthInfo 함수는 바인딩 핸들의 인증 및 권한 부여 정보를 설정합니다. (RpcBindingSetAuthInfoA)
RpcBindingSetAuthInfoExA RpcBindingSetAuthInfoEx 함수는 바인딩 핸들의 인증, 권한 부여 및 보안 서비스 품질 정보를 설정합니다. (ANSI)
RpcBindingSetAuthInfoExW RpcBindingSetAuthInfoEx 함수는 바인딩 핸들의 인증, 권한 부여 및 보안 서비스 품질 정보를 설정합니다. (유니코드)
RpcBindingSetAuthInfoW RpcBindingSetAuthInfoW(유니코드) 함수(rpcdce.h)는 바인딩 핸들의 인증 및 권한 부여 정보를 설정합니다.
RpcBindingSetObject RpcBindingSetObject 함수는 바인딩 핸들에서 개체 UUID 값을 설정합니다.
RpcBindingSetOption RpcBindingSetOption 함수를 사용하면 클라이언트 애플리케이션이 바인딩 핸들에 메시지 큐 옵션을 지정할 수 있습니다.

RpcBindingToStringBinding RpcBindingToStringBinding 함수(rpcdce.h)는 바인딩 핸들의 문자열 표현을 반환합니다.
RpcBindingToStringBindingA RpcBindingToStringBinding 함수는 바인딩 핸들의 문자열 표현을 반환합니다. (RpcBindingToStringBindingA)
RpcBindingToStringBindingW RpcBindingToStringBindingW(유니코드) 함수(rpcdce.h)는 바인딩 핸들의 문자열 표현을 반환합니다.
RpcBindingUnbind 이전에 RpcBindingBind에 의해 바인딩된 바인딩 핸들의 바인딩을 해제합니다.
RpcBindingVectorFree RpcBindingVectorFree 함수는 벡터와 벡터 자체에 포함된 바인딩 핸들을 해제합니다.
RpcCancelThread RpcCancelThread 함수는 스레드를 취소합니다. RpcCancelThread 함수는 비동기 RPC 호출을 취소하는 데 사용하면 안 됩니다. 대신 RpcAsyncCancelCall 함수를 사용하여 비동기 RPC 호출을 취소합니다.
RpcCancelThreadEx RpcCancelThreadEx 함수는 스레드 실행을 중지합니다.
RpcCertGeneratePrincipalName RpcCertGeneratePrincipalName 함수(rpcssl.h)는 서버 프로그램에서 보안 인증서의 보안 이름 생성에 사용됩니다.
RpcCertGeneratePrincipalNameA 서버 프로그램은 RpcCertGeneratePrincipalName 함수를 사용하여 보안 인증서에 대한 보안 주체 이름을 생성합니다. (RpcCertGeneratePrincipalNameA)
RpcCertGeneratePrincipalNameW RpcCertGeneratePrincipalNameW(유니코드) 함수(rpcssl.h)는 서버 프로그램에서 보안 인증서의 보안 주체 이름을 생성하는 데 사용됩니다.

RpcEpRegister RpcEpRegister 함수(rpcdce.h)는 로컬 엔드포인트 맵 데이터베이스의 서버 주소 정보를 추가하거나 대체합니다.
RpcEpRegisterA RpcEpRegister 함수는 로컬 엔드포인트 맵 데이터베이스의 서버 주소 정보를 추가하거나 대체합니다. (RpcEpRegisterA)
RpcEpRegisterNoReplace RpcEpRegisterNoReplace 함수(rpcdce.h)는 로컬 엔드포인트 맵 데이터베이스에 서버 주소 정보를 추가합니다.
RpcEpRegisterNoReplaceA RpcEpRegisterNoReplace 함수는 로컬 엔드포인트 맵 데이터베이스에 서버 주소 정보를 추가합니다. (RpcEpRegisterNoReplaceA)
RpcEpRegisterNoReplaceW RpcEpRegisterNoReplaceW(유니코드) 함수(rpcdce.h)는 로컬 엔드포인트 맵 데이터베이스에 서버 주소 정보를 추가합니다.
RpcEpRegisterW RpcEpRegisterW(유니코드) 함수(rpcdce.h)는 로컬 엔드포인트 맵 데이터베이스의 서버 주소 정보를 추가하거나 대체합니다.
RpcEpResolveBinding RpcEpResolveBinding 함수는 부분적으로 바인딩된 서버 바인딩 핸들을 완전히 바인딩된 서버 바인딩 핸들로 확인합니다.
RpcEpUnregister RpcEpUnregister 함수는 로컬 엔드포인트 맵 데이터베이스에서 서버 주소 정보를 제거합니다.
RpcErrorAddRecord RpcErrorAddRecord 함수는 확장 오류 정보 레코드 체인에 확장 오류 정보를 추가합니다.
RpcErrorClearInformation RpcErrorClearInformation 함수는 현재 스레드에 대한 모든 확장된 오류 정보를 지웁니다.

RpcErrorEndEnumeration RpcErrorEndEnumeration 함수는 확장 오류 정보의 열거를 종료하고 열거를 위해 RPC에서 할당한 모든 리소스를 해제합니다.
RpcErrorGetNextRecord RpcErrorGetNextRecord 함수는 열거형 핸들에 대한 다음 확장 오류 정보 레코드를 검색합니다.
RpcErrorGetNumberOfRecords RpcErrorGetNumberOfRecords 함수는 확장 오류 정보의 레코드 수를 반환합니다.
RpcErrorLoadErrorInfo RpcErrorLoadErrorInfo 함수는 RpcErrorSaveErrorInfo 호출에서 얻은 BLOB을 확장된 오류 정보로 변환합니다.
RpcErrorResetEnumeration RpcErrorResetEnumeration 함수는 모든 In-process 열거형에 대한 열거형 커서를 다시 설정하여 RpcErrorGetNextRecord에 대한 후속 호출이 첫 번째 확장 오류 정보 레코드를 검색하도록 프로세스를 다시 설정합니다.
RpcErrorSaveErrorInfo RpcErrorSaveErrorInfo 함수는 열거형 핸들에 대한 모든 오류 정보를 BLOB으로 반환합니다.
RpcErrorStartEnumeration RpcErrorStartEnumeration 함수는 확장 오류 정보의 열거를 시작합니다.
RpcExcept RpcExcept 문은 RPC 애플리케이션에 대한 구조적 예외 처리를 제공합니다.
RpcExceptionFilter 예외가 치명적인지 아니면 치명적이 아닌지를 결정하는 기본 예외 필터입니다.
RpcFreeAuthorizationContext RpcFreeAuthorizationContext 함수는 RpcGetAuthorizationContextForClient 함수에 대한 이전 호출에서 얻은 Authz 컨텍스트를 해제합니다.

RpcGetAuthorizationContextForClient RpcGetAuthorizationContextForClient 함수는 고성능 인증을 위해 Authz 함수와 함께 사용할 수 있는 RPC 클라이언트에 대한 Authz 컨텍스트를 반환합니다. ncalrpc 및 ncacn_* 프로토콜 시퀀스에 대해서만 지원됩니다.
RpcIfIdVectorFree RpcIfIdVectorFree 함수는 벡터와 벡터에 포함된 인터페이스 식별 구조를 해제합니다.
RpcIfInqId RpcIfInqId 함수는 인터페이스 사양의 인터페이스 식별 부분을 반환합니다.
RpcImpersonateClient 클라이언트 원격 프로시저 호출을 처리하는 서버 스레드는 RpcImpersonateClient 함수를 호출하여 활성 클라이언트를 가장할 수 있습니다.
RpcImpersonateClientContainer RpcImpersonateClientContainer를 변경하거나 사용할 수 없습니다.
RpcMgmtEnableIdleCleanup RpcMgmtEnableIdleCleanup 함수를 사용하면 RPC가 클라이언트에서 네트워크 연결과 같은 유휴 리소스를 닫을 수 있습니다.
RpcMgmtEpEltInqBegin RpcMgmtEpEltInqBegin 함수는 엔드포인트 맵의 요소를 보기 위한 조회 컨텍스트를 만듭니다.
RpcMgmtEpEltInqDone RpcMgmtEpEltInqDone 함수는 엔드포인트 맵에서 요소를 보기 위한 조회 컨텍스트를 삭제합니다.
RpcMgmtEpEltInqNext RpcMgmtEpEltInqNext 함수(rpcdce.h)는 엔드포인트 맵에서 하나의 요소를 반환합니다.
RpcMgmtEpEltInqNextA RpcMgmtEpEltInqNext 함수는 엔드포인트 맵에서 하나의 요소를 반환합니다. (RpcMgmtEpEltInqNextA)
RpcMgmtEpEltInqNextW RpcMgmtEpEltInqNextW(유니코드) 함수(rpcdce.h)는 엔드포인트 맵에서 하나의 요소를 반환합니다.

RpcMgmtEpUnregister RpcMgmtEpUnregister 함수는 엔드포인트 맵에서 서버 주소 정보를 제거합니다.
RpcMgmtInqComTimeout RpcMgmtInqComTimeout 함수는 바인딩 핸들에서 바인딩 통신 제한 시간 값을 반환합니다.
RpcMgmtInqDefaultProtectLevel RpcMgmtInqDefaultProtectLevel 함수는 인증 서비스에 대한 기본 인증 수준을 반환합니다.
RpcMgmtInqIfIds RpcMgmtInqIfIds 함수는 서버에서 제공하는 인터페이스의 식별자를 포함하는 벡터를 반환합니다.
RpcMgmtInqServerPrincName RpcMgmtInqServerPrincName 함수(rpcdce.h)는 서버의 주체 이름을 반환합니다.
RpcMgmtInqServerPrincNameA RpcMgmtInqServerPrincName 함수는 서버의 주체 이름을 반환합니다. (RpcMgmtInqServerPrincNameA)
RpcMgmtInqServerPrincNameW RpcMgmtInqServerPrincNameW(유니코드) 함수(rpcdce.h)는 서버의 주체 이름을 반환합니다.
RpcMgmtInqStats RpcMgmtInqStats 함수는 RPC 런타임 통계를 반환합니다.
RpcMgmtIsServerListening RpcMgmtIsServerListening 함수는 서버가 원격 프로시저 호출을 수신 대기하는지 여부를 알려줍니다.
RpcMgmtSetAuthorizationFn RpcMgmtSetAuthorizationFn 함수는 서버의 관리 함수에 대한 원격 호출을 처리하기 위한 권한 부여 함수를 설정합니다.
RpcMgmtSetCancelTimeout RpcMgmtSetCancelTimeout 함수는 취소를 전달한 후 시간이 초과되기 전에 대기할 시간에 하한을 설정합니다.

RpcMgmtSetComTimeout RpcMgmtSetComTimeout 함수는 바인딩 핸들에서 바인딩 통신 제한 시간 값을 설정합니다.
RpcMgmtSetServerStackSize RpcMgmtSetServerStackSize 함수는 RPC 런타임에서 만든 서버 스레드의 스택 크기를 지정합니다.
RpcMgmtStatsVectorFree RpcMgmtStatsVectorFree 함수는 통계 벡터를 해제합니다.
RpcMgmtStopServerListening RpcMgmtStopServerListening 함수는 서버에 원격 프로시저 호출 수신 대기를 중지하도록 지시합니다. 이 함수는 자동 수신 대기 인터페이스에 영향을 미치지 않습니다. 자세한 내용은 RpcServerRegisterIfEx를 참조하세요.
RpcMgmtWaitServerListen RpcMgmtWaitServerListen 함수는 일반적으로 RpcServerListen과 연결된 대기 작업을 수행합니다.
RpcNetworkInqProtseqs RpcNetworkInqProtseqs 함수(rpcdce.h)는 RPC 런타임 라이브러리와 운영 체제에서 지원하는 모든 프로토콜 시퀀스를 반환합니다.
RpcNetworkInqProtseqsA RpcNetworkInqProtseqs 함수는 RPC 런타임 라이브러리와 운영 체제에서 지원하는 모든 프로토콜 시퀀스를 반환합니다. (RpcNetworkInqProtseqsA)
RpcNetworkInqProtseqsW RpcNetworkInqProtseqsW(유니코드) 함수(rpcdce.h)는 RPC 런타임 라이브러리와 운영 체제 모두에서 지원되는 모든 프로토콜 시퀀스를 반환합니다.
RpcNetworkIsProtseqValid RpcNetworkIsProtseqValid 함수(rpcdce.h)는 RPC 런타임 라이브러리와 운영 체제가 지정된 프로토콜 시퀀스를 지원하는지 알려줍니다.
RpcNetworkIsProtseqValidA RpcNetworkIsProtseqValid 함수는 지정된 프로토콜 시퀀스가 RPC 런타임 라이브러리와 운영 체제 모두에서 지원되는지 여부를 알려줍니다. 서버 애플리케이션은 종종 RpcNetworkInqProtseqs를 사용합니다. (RpcNetworkIsProtseqValidA)

RpcNetworkIsProtseqValidW RpcNetworkIsProtseqValidW(유니코드) 함수(rpcdce.h)는 RPC 런타임 라이브러리와 운영 체제가 지정된 프로토콜 시퀀스를 지원하는지 알려줍니다.
RPCNOTIFICATION_ROUTINE RPCNOTIFICATION_ROUTINE 함수는 비동기 이벤트에 대한 응답을 사용자 지정하는 기능을 사용하여 비동기 RPC를 활용하는 프로그램을 제공합니다.
RpcNsBindingExportA RpcNsBindingExport 함수는 여러 바인딩 핸들과 서버에 대한 여러 개체가 있는 이름 서비스 데이터베이스 항목을 설정합니다. (ANSI)
RpcNsBindingExportPnPA RpcNsBindingExportPnP 함수는 플러그 앤 플레이 지원하는 서버에 대해 여러 바인딩 핸들 및 여러 개체가 있는 이름 서비스 데이터베이스 항목을 설정합니다. (ANSI)
RpcNsBindingExportPnPW RpcNsBindingExportPnP 함수는 플러그 앤 플레이 지원하는 서버에 대해 여러 바인딩 핸들 및 여러 개체가 있는 이름 서비스 데이터베이스 항목을 설정합니다. (유니코드)
RpcNsBindingExportW RpcNsBindingExport 함수는 여러 바인딩 핸들과 서버에 대한 여러 개체가 있는 이름 서비스 데이터베이스 항목을 설정합니다. (유니코드)
RpcNsBindingImportBeginA RpcNsBindingImportBegin 함수는 지정된 인터페이스 및 개체를 제공하는 서버에 대해 클라이언트 호환 바인딩 핸들을 가져오기 위한 가져오기 컨텍스트를 만듭니다. (ANSI)
RpcNsBindingImportBeginW RpcNsBindingImportBegin 함수는 지정된 인터페이스 및 개체를 제공하는 서버에 대해 클라이언트 호환 바인딩 핸들을 가져오기 위한 가져오기 컨텍스트를 만듭니다. (유니코드)
RpcNsBindingImportDone RpcNsBindingImportDone 함수는 클라이언트가 호환되는 서버 검색을 완료하고 가져오기 컨텍스트를 삭제했음을 알 수 있습니다.
RpcNsBindingImportNext RpcNsBindingImportNext 함수는 인터페이스(및 선택적으로 이름 서비스 데이터베이스의 개체)를 조회하고 호환되는 서버의 바인딩 핸들을 반환합니다(있는 경우).

RpcNsBindingInqEntryName RpcNsBindingInqEntryName 함수(rpcdce.h)는 바인딩 핸들이 제공된 항목 이름을 반환합니다.
RpcNsBindingInqEntryNameA RpcNsBindingInqEntryName 함수는 바인딩 핸들이 제공된 항목 이름을 반환합니다. (RpcNsBindingInqEntryNameA)
RpcNsBindingInqEntryNameW RpcNsBindingInqEntryNameW(유니코드) 함수(rpcdce.h)는 바인딩 핸들이 제공된 항목 이름을 반환합니다.
RpcNsBindingLookupBeginA RpcNsBindingLookupBegin 함수는 인터페이스 및 개체에 대한 조회 컨텍스트를 만듭니다. (ANSI)
RpcNsBindingLookupBeginW RpcNsBindingLookupBegin 함수는 인터페이스 및 개체에 대한 조회 컨텍스트를 만듭니다. (유니코드)
RpcNsBindingLookupDone RpcNsBindingLookupDone 함수는 클라이언트가 호환되는 서버 찾기를 완료하고 조회 컨텍스트를 삭제했음을 나타냅니다.
RpcNsBindingLookupNext RpcNsBindingLookupNext 함수는 지정된 인터페이스 및 선택적으로 개체에 대해 호환되는 바인딩 핸들 목록을 반환합니다.
RpcNsBindingSelect RpcNsBindingSelect 함수는 호환되는 바인딩 핸들 목록에서 바인딩 핸들을 반환합니다.
RpcNsBindingUnexportA RpcNsBindingUnexport 함수는 이름 서비스 데이터베이스의 항목에서 인터페이스 및 개체에 대한 바인딩 핸들을 제거합니다. (ANSI)
RpcNsBindingUnexportPnPA RpcNsBindingUnexportPnP 함수는 이름 서비스 데이터베이스의 항목에서 플러그 앤 플레이 인터페이스 및 개체에 대한 바인딩 핸들을 제거합니다. (ANSI)

RpcNsBindingUnexportPnPW RpcNsBindingUnexportPnP 함수는 이름 서비스 데이터베이스의 항목에서 플러그 앤 플레이 인터페이스 및 개체에 대한 바인딩 핸들을 제거합니다. (유니코드)
RpcNsBindingUnexportW RpcNsBindingUnexport 함수는 이름 서비스 데이터베이스의 항목에서 인터페이스 및 개체에 대한 바인딩 핸들을 제거합니다. (유니코드)
RpcNsEntryExpandNameA RpcNsEntryExpandName 함수는 이름 서비스 항목 이름을 확장합니다. 이 함수는 Active Directory에서 지원됩니다. (ANSI)
RpcNsEntryExpandNameW RpcNsEntryExpandName 함수는 이름 서비스 항목 이름을 확장합니다. 이 함수는 Active Directory에서 지원됩니다. (유니코드)
RpcNsEntryObjectInqBeginA RpcNsEntryObjectInqBegin 함수는 이름 서비스 데이터베이스 항목의 개체에 대한 조회 컨텍스트를 만듭니다. (ANSI)
RpcNsEntryObjectInqBeginW RpcNsEntryObjectInqBegin 함수는 이름 서비스 데이터베이스 항목의 개체에 대한 조회 컨텍스트를 만듭니다. (유니코드)
RpcNsEntryObjectInqDone RpcNsEntryObjectInqDone 함수는 이름 서비스 데이터베이스 항목의 개체에 대한 조회 컨텍스트를 삭제합니다.
RpcNsEntryObjectInqNext RpcNsEntryObjectInqNext 함수는 이름 서비스 데이터베이스 항목에서 한 번에 하나의 개체를 반환합니다.
RpcNsGroupDeleteA RpcNsGroupDelete 함수는 그룹 특성을 삭제합니다. (ANSI)
RpcNsGroupDeleteW RpcNsGroupDelete 함수는 그룹 특성을 삭제합니다. (유니코드)

RpcNsGroupMbrAddA RpcNsGroupMbrAdd 함수는 그룹에 항목 이름을 추가합니다. 필요한 경우 항목을 만듭니다. (ANSI)
RpcNsGroupMbrAddW RpcNsGroupMbrAdd 함수는 그룹에 항목 이름을 추가합니다. 필요한 경우 항목을 만듭니다. (유니코드)
RpcNsGroupMbrInqBeginA RpcNsGroupMbrInqBegin 함수는 그룹 멤버를 보기 위한 조회 컨텍스트를 만듭니다. (ANSI)
RpcNsGroupMbrInqBeginW RpcNsGroupMbrInqBegin 함수는 그룹 멤버를 보기 위한 조회 컨텍스트를 만듭니다. (유니코드)
RpcNsGroupMbrInqDone RpcNsGroupMbrInqDone 함수는 그룹에 대한 조회 컨텍스트를 삭제합니다.
RpcNsGroupMbrInqNextA RpcNsGroupMbrInqNext 함수는 한 번에 그룹에서 하나의 항목 이름을 반환합니다. (ANSI)
RpcNsGroupMbrInqNextW RpcNsGroupMbrInqNext 함수는 한 번에 그룹에서 하나의 항목 이름을 반환합니다. (유니코드)
RpcNsGroupMbrRemoveA RpcNsGroupMbrRemove 함수는 그룹에서 항목 이름을 제거합니다. (ANSI)
RpcNsGroupMbrRemoveW RpcNsGroupMbrRemove 함수는 그룹에서 항목 이름을 제거합니다. (유니코드)
RpcNsMgmtBindingUnexportA RpcNsMgmtBindingUnexport 함수는 이름 서비스 데이터베이스의 항목에서 여러 바인딩 핸들 및 개체를 제거합니다. (ANSI)
RpcNsMgmtBindingUnexportW RpcNsMgmtBindingUnexport 함수는 이름 서비스 데이터베이스의 항목에서 여러 바인딩 핸들 및 개체를 제거합니다. (유니코드)

RpcNsMgmtEntryCreateA RpcNsMgmtEntryCreate 함수는 이름 서비스 데이터베이스 항목을 만듭니다. (ANSI)
RpcNsMgmtEntryCreateW RpcNsMgmtEntryCreate 함수는 이름 서비스 데이터베이스 항목을 만듭니다. (유니코드)
RpcNsMgmtEntryDeleteA RpcNsMgmtEntryDelete 함수는 이름 서비스 데이터베이스 항목을 삭제합니다. (ANSI)
RpcNsMgmtEntryDeleteW RpcNsMgmtEntryDelete 함수는 이름 서비스 데이터베이스 항목을 삭제합니다. (유니코드)
RpcNsMgmtEntryInqIfIdsA RpcNsMgmtEntryInqIfIds 함수는 이름 서비스 데이터베이스 항목으로 내보낸 인터페이스 목록을 반환합니다. (ANSI)
RpcNsMgmtEntryInqIfIdsW RpcNsMgmtEntryInqIfIds 함수는 이름 서비스 데이터베이스 항목으로 내보낸 인터페이스 목록을 반환합니다. (유니코드)
RpcNsMgmtHandleSetExpAge RpcNsMgmtHandleSetExpAge 함수는 이름 서비스 데이터의 로컬 복사본에 대한 이름 서비스 핸들의 만료 기간을 설정합니다.
RpcNsMgmtInqExpAge RpcNsMgmtInqExpAge 함수는 이름 서비스 데이터의 로컬 복사본에 대한 전역 만료 기간을 반환합니다.
RpcNsMgmtSetExpAge RpcNsMgmtSetExpAge 함수는 이름 서비스 데이터의 로컬 복사본에 대한 애플리케이션의 전역 만료 기간을 수정합니다.
RpcNsProfileDeleteA RpcNsProfileDelete 함수는 프로필 특성을 삭제합니다. (ANSI)
RpcNsProfileDeleteW RpcNsProfileDelete 함수는 프로필 특성을 삭제합니다. (유니코드)

RpcNsProfileEltAddA RpcNsProfileEltAdd 함수는 프로필에 요소를 추가합니다. 필요한 경우 항목을 만듭니다. (ANSI)
RpcNsProfileEltAddW RpcNsProfileEltAdd 함수는 프로필에 요소를 추가합니다. 필요한 경우 항목을 만듭니다. (유니코드)
RpcNsProfileEltInqBeginA RpcNsProfileEltInqBegin 함수는 프로필의 요소를 보기 위한 조회 컨텍스트를 만듭니다. (ANSI)
RpcNsProfileEltInqBeginW RpcNsProfileEltInqBegin 함수는 프로필의 요소를 보기 위한 조회 컨텍스트를 만듭니다. (유니코드)
RpcNsProfileEltInqDone RpcNsProfileEltInqDone 함수는 프로필의 요소를 보기 위한 조회 컨텍스트를 삭제합니다.
RpcNsProfileEltInqNextA RpcNsProfileEltInqNext 함수는 프로필에서 한 번에 하나의 요소를 반환합니다. (ANSI)
RpcNsProfileEltInqNextW RpcNsProfileEltInqNext 함수는 프로필에서 한 번에 하나의 요소를 반환합니다. (유니코드)
RpcNsProfileEltRemoveA RpcNsProfileEltRemove 함수는 프로필에서 요소를 제거합니다. (ANSI)
RpcNsProfileEltRemoveW RpcNsProfileEltRemove 함수는 프로필에서 요소를 제거합니다. (유니코드)
RpcObjectInqType RpcObjectInqType 함수는 개체의 형식을 반환합니다.
RpcObjectSetInqFn RpcObjectSetInqFn 함수는 개체 조회 함수를 등록합니다. null 값은 이전에 등록된 개체 조회 함수를 해제합니다.
RpcObjectSetType RpcObjectSetType 함수는 개체의 형식을 할당합니다.

RpcProtseqVectorFree RpcProtseqVectorFree 함수(rpcdce.h)는 벡터와 벡터 자체에 포함된 프로토콜 시퀀스를 해제합니다.
RpcProtseqVectorFreeA RpcProtseqVectorFree 함수는 벡터와 벡터 자체에 포함된 프로토콜 시퀀스를 해제합니다. (RpcProtseqVectorFreeA)
RpcProtseqVectorFreeW RpcProtseqVectorFreeW(유니코드) 함수(rpcdce.h)는 벡터와 벡터 자체에 포함된 프로토콜 시퀀스를 해제합니다.
RpcRaiseException RpcRaiseException 함수를 사용하여 예외를 발생합니다. 함수가 호출자에게 반환되지 않습니다.
RpcRevertContainerImpersonation RpcRevertContainerImpersonation을 변경하거나 사용할 수 없습니다.
RpcRevertToSelf RpcImpersonateClient를 호출하고 클라이언트 가장이 필요한 작업을 완료한 후 서버는 RpcRevertToSelf를 호출하여 가장을 종료하고 자체 보안 ID를 다시 설정합니다.
RpcRevertToSelfEx RpcRevertToSelfEx 함수를 사용하면 서버가 클라이언트를 가장한 다음, 클라이언트를 가장하는 호출이 원래 RPC에서 디스패치된 스레드가 아닌 스레드에서 올 수 있는 다중 스레드 작업에서 되돌릴 수 있습니다.
RpcServerCompleteSecurityCallback RpcServerCompleteSecurityCallback 함수는 비동기 보안 콜백을 완료합니다.
RpcServerInqBindingHandle RpcServerInqBindingHandle이 호출되는 스레드에서 서비스되는 RPC 호출에 대한 바인딩 핸들을 가져옵니다.
RpcServerInqBindings RpcServerInqBindings 함수는 원격 프로시저 호출을 받을 수 있는 바인딩 핸들을 반환합니다.

RpcServerInqCallAttributesA RpcServerInqCallAttributes 함수는 클라이언트 보안 컨텍스트 특성을 가져오는 RPC 서버 호출입니다. (ANSI)
RpcServerInqCallAttributesW RpcServerInqCallAttributes 함수는 클라이언트 보안 컨텍스트 특성을 가져오는 RPC 서버 호출입니다. (유니코드)
RpcServerInqDefaultPrincName RpcServerInqDefaultPrincName 함수(rpcdce.h)는 지정된 인증 서비스의 기본 보안 주체 이름을 가져옵니다.
RpcServerInqDefaultPrincNameA RpcServerInqDefaultPrincName 함수는 지정된 인증 서비스에 대한 기본 보안 주체 이름을 가져옵니다. (RpcServerInqDefaultPrincNameA)
RpcServerInqDefaultPrincNameW RpcServerInqDefaultPrincNameW(유니코드) 함수(rpcdce.h)는 지정된 인증 서비스에 대한 기본 보안 주체 이름을 가져옵니다.
RpcServerInqIf RpcServerInqIf 함수는 인터페이스에 등록된 EPV(관리자 진입점 벡터)를 반환합니다.
RpcServerInterfaceGroupActivate RpcServerInterfaceGroupActivate 함수는 RPC 서버 런타임에 인터페이스 그룹의 인터페이스 및 엔드포인트를 등록하고 호출 수신 대기를 시작하도록 지시합니다.
RpcServerInterfaceGroupClose RpcServerInterfaceGroupClose 함수는 인터페이스 그룹을 해제하는 데 사용됩니다.
RpcServerInterfaceGroupCreate RpcServerInterfaceGroupCreate 함수(rpcdce.h)는 서버 애플리케이션에 대한 RPC 서버 인터페이스 그룹을 만듭니다.
RpcServerInterfaceGroupCreateA RpcServerInterfaceGroupCreate 함수는 서버 애플리케이션에 대한 RPC 서버 인터페이스 그룹을 만듭니다. (RpcServerInterfaceGroupCreateA)

RpcServerInterfaceGroupCreateW RpcServerInterfaceGroupCreateW(유니코드) 함수(rpcdce.h)는 서버 애플리케이션에 대한 RPC 서버 인터페이스 그룹을 만듭니다.
RpcServerInterfaceGroupDeactivate RpcServerInterfaceGroupDeactivate 함수는 RPC 런타임에 지정된 인터페이스 그룹을 닫으려고 시도하도록 지시하고, 선택적으로 처리되지 않은 클라이언트 작업이 있는 경우 작업을 중단합니다.
RpcServerInterfaceGroupInqBindings RpcServerInterfaceGroupInqBindings 함수는 지정된 인터페이스 그룹에 대해 원격 프로시저 호출을 받을 수 있는 바인딩 핸들을 반환합니다.
RpcServerListen RpcServerListen 함수는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신 대기하도록 신호를 전송합니다. 이 함수는 자동 수신 대기 인터페이스에 영향을 미치지 않습니다. 해당 기능이 필요한 경우 RpcServerRegisterIfEx를 사용합니다.
RpcServerRegisterAuthInfo RpcServerRegisterAuthInfo 함수(rpcdce.h)는 RPC 런타임 라이브러리에 인증 정보를 등록합니다.
RpcServerRegisterAuthInfoA RpcServerRegisterAuthInfo 함수는 RPC 런타임 라이브러리에 인증 정보를 등록합니다. (RpcServerRegisterAuthInfoA)
RpcServerRegisterAuthInfoW RpcServerRegisterAuthInfoW(유니코드) 함수(rpcdce.h)는 RPC 런타임 라이브러리에 인증 정보를 등록합니다.
RpcServerRegisterIf RpcServerRegisterIf 함수는 RPC 런타임 라이브러리에 인터페이스를 등록합니다.
RpcServerRegisterIf2 RpcServerRegisterIf2 함수는 RPC 런타임 라이브러리에 인터페이스를 등록합니다.
RpcServerRegisterIf3 RpcServerRegisterIf3 함수는 RPC 런타임 라이브러리에 인터페이스를 등록합니다.

RpcServerRegisterIfEx RpcServerRegisterIfEx 함수는 RPC 런타임 라이브러리에 인터페이스를 등록합니다.
RpcServerSubscribeForNotification RPC 알림을 위해 서버를 구독합니다.
RpcServerTestCancel 서버는 RpcServerTestCancel을 호출하여 클라이언트 취소 요청을 테스트합니다.
RpcServerUnregisterIf RpcServerUnregisterIf 함수는 RPC 런타임 라이브러리 레지스트리에서 인터페이스를 제거합니다.
RpcServerUnregisterIfEx RpcServerUnregisterIfEx 함수는 RPC 런타임 라이브러리 레지스트리에서 인터페이스를 제거합니다. 이 함수는 RpcServerUnregisterIf 함수의 기능을 확장합니다.
RpcServerUnsubscribeForNotification RPC 알림에서 서버 구독을 취소합니다.
RpcServerUseAllProtseqs RpcServerUseAllProtseqs 함수는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위해 지원되는 모든 프로토콜 시퀀스를 사용하도록 지시합니다.
RpcServerUseAllProtseqsEx RpcServerUseAllProtseqsEx 함수는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위해 지원되는 모든 프로토콜 시퀀스를 사용하도록 지시합니다.
RpcServerUseAllProtseqsIf RpcServerUseAllProtseqsIf 함수는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위해 인터페이스 사양에서 지정된 모든 프로토콜 시퀀스 및 엔드포인트를 사용하도록 지시합니다.
RpcServerUseAllProtseqsIfEx RpcServerUseAllProtseqsIfEx 함수는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위해 인터페이스 사양에서 지정된 모든 프로토콜 시퀀스 및 엔드포인트를 사용하도록 지시합니다.
RpcServerUseProtseq RpcServerUseProtseq 함수(rpcdce.h)는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위해 지정된 프로토콜 시퀀스를 사용하도록 지시합니다.

RpcServerUseProtseqA RpcServerUseProtseq 함수는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위해 지정된 프로토콜 시퀀스를 사용하도록 지시합니다. (RpcServerUseProtseqA)
RpcServerUseProtseqEp RpcServerUseProtseqEp 함수(rpcdce.h)는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위해 지정된 프로토콜 시퀀스 및 엔드포인트를 사용하도록 지시합니다.
RpcServerUseProtseqEpA RpcServerUseProtseqEp 함수는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위해 지정된 엔드포인트와 결합된 지정된 프로토콜 시퀀스를 사용하도록 지시합니다. (RpcServerUseProtseqEpA)
RpcServerUseProtseqEpEx RpcServerUseProtseqEpEx 함수(rpcdce.h)는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위해 지정된 프로토콜 시퀀스 및 엔드포인트를 사용하도록 지시합니다.
RpcServerUseProtseqEpExA RpcServerUseProtseqEpEx 함수는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위해 지정된 엔드포인트와 결합된 지정된 프로토콜 시퀀스를 사용하도록 지시합니다. (RpcServerUseProtseqEpExA)
RpcServerUseProtseqEpExW RpcServerUseProtseqEpExW(유니코드) 함수(rpcdce.h)는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위해 지정된 프로토콜 시퀀스 및 엔드포인트를 사용하도록 지시합니다.
RpcServerUseProtseqEpW RpcServerUseProtseqEpW(유니코드) 함수(rpcdce.h)는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위해 지정된 프로토콜 시퀀스 및 엔드포인트를 사용하도록 지시합니다.
RpcServerUseProtseqEx RpcServerUseProtseqEx 함수(rpcdce.h)는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위해 지정된 프로토콜 시퀀스를 사용하도록 지시합니다.
RpcServerUseProtseqExA RpcServerUseProtseqEx 함수는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위해 지정된 프로토콜 시퀀스를 사용하도록 지시합니다. (RpcServerUseProtseqExA)

RpcServerUseProtseqExW RpcServerUseProtseqExW(유니코드) 함수(rpcdce.h)는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위해 지정된 프로토콜 시퀀스를 사용하도록 지시합니다.
RpcServerUseProtseqIf RpcServerUseProtseqIf 함수(rpcdce.h)는 RPC 런타임 라이브러리에 원격 프로시저 호출 수신을 위해 인터페이스 사양에서 지정된 프로토콜 시퀀스 및 엔드포인트를 사용하도록 지시합니다.
RpcServerUseProtseqIfA RpcServerUseProtseqIf 함수는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위한 인터페이스 사양의 엔드포인트와 결합된 지정된 프로토콜 시퀀스를 사용하도록 지시합니다. (RpcServerUseProtseqIfA)
RpcServerUseProtseqIfEx RpcServerUseProtseqIfEx 함수(rpcdce.h)는 RPC 런타임 라이브러리에 원격 프로시저 호출 수신을 위한 인터페이스 사양에서 지정된 프로토콜 시퀀스 및 엔드포인트를 사용하도록 지시합니다.
RpcServerUseProtseqIfExA RpcServerUseProtseqIfEx 함수는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위한 인터페이스 사양의 엔드포인트와 결합된 지정된 프로토콜 시퀀스를 사용하도록 지시합니다. (RpcServerUseProtseqIfExA)
RpcServerUseProtseqIfExW RpcServerUseProtseqIfExW(유니코드) 함수(rpcdce.h)는 RPC 런타임 라이브러리에 원격 프로시저 호출 수신을 위해 인터페이스 사양에서 지정된 프로토콜 시퀀스 및 엔드포인트를 사용하도록 지시합니다.
RpcServerUseProtseqIfW RpcServerUseProtseqIfW(유니코드) 함수(rpcdce.h)는 RPC 런타임 라이브러리에 원격 프로시저 호출 수신을 위한 인터페이스 사양에서 지정된 프로토콜 시퀀스 및 엔드포인트를 사용하도록 지시합니다.
RpcServerUseProtseqW RpcServerUseProtseqW(유니코드) 함수(rpcdce.h)는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위해 지정된 프로토콜 시퀀스를 사용하도록 지시합니다.
RpcSmAllocate RpcSmAllocate 함수는 RPC 스텁 메모리 관리 함수 내에서 메모리를 할당하고 할당된 메모리 또는 NULL에 대한 포인터를 반환합니다.

RpcSmClientFree RpcSmClientFree 함수는 클라이언트 스텁에서 반환된 메모리를 해제합니다.
RpcSmDestroyClientContext RpcSmDestroyClientContext 함수는 컨텍스트 핸들에 대한 클라이언트 메모리 리소스를 회수하고 컨텍스트 핸들 NULL을 만듭니다.
RpcSmDisableAllocate RpcSmDisableAllocate 함수는 스텁 메모리 관리 환경 내에서 리소스와 메모리를 해제합니다.
RpcSmEnableAllocate RpcSmEnableAllocate 함수는 스텁 메모리 관리 환경을 설정합니다.
RpcSmFree RpcSmFree 함수는 RpcSmAllocate에 의해 할당된 메모리를 해제합니다.
RpcSmGetThreadHandle RpcSmGetThreadHandle 함수는 스텁 메모리 관리 환경에 대한 스레드 핸들 또는 NULL을 반환합니다.
RpcSmSetClientAllocFree RpcSmSetClientAllocFree 함수는 클라이언트 스텁에서 사용하는 메모리 할당 및 해제 메커니즘을 사용하도록 설정합니다.
RpcSmSetThreadHandle RpcSmSetThreadHandle 함수는 스텁 메모리 관리 환경에 대한 스레드 핸들을 설정합니다.
RpcSmSwapClientAllocFree RpcSmSwapClientAllocFree 함수는 클라이언트 스텁의 메모리 할당 및 메모리 해제 메커니즘을 클라이언트에서 제공하는 메커니즘과 교환합니다.
RpcSsAllocate RpcSsAllocate 함수는 RPC 스텁 메모리 관리 함수 내에서 메모리를 할당하고 할당된 메모리 또는 NULL에 대한 포인터를 반환합니다.
RpcSsContextLockExclusive RpcSsContextLockExclusive 함수를 사용하면 애플리케이션이 단독 모드에서 컨텍스트 핸들 사용을 시작할 수 있습니다.

RpcSsContextLockShared RpcSsContextLockShared 함수를 사용하면 애플리케이션이 공유 모드에서 컨텍스트 핸들 사용을 시작할 수 있습니다.
RpcSsDestroyClientContext RpcSsDestroyClientContext 함수는 서버에 연결하지 않고 클라이언트에서 더 이상 필요하지 않은 컨텍스트 핸들을 삭제합니다.
RpcSsDisableAllocate RpcSsDisableAllocate 함수는 스텁 메모리 관리 환경 내에서 리소스 및 메모리를 해제합니다.
RpcSsDontSerializeContext RpcSsDontSerializeContext 함수는 동일한 컨텍스트 핸들에서 서버 관리자 루틴으로 디스패치된 여러 호출의 런타임 serialization을 사용하지 않도록 설정합니다.
RpcSsEnableAllocate RpcSsEnableAllocate 함수는 스텁 메모리 관리 환경을 설정합니다.
RpcSsFree RpcSsFree 함수는 RpcSsAllocate에 의해 할당된 메모리를 해제합니다.
RpcSsGetThreadHandle RpcSsGetThreadHandle 함수는 스텁 메모리 관리 환경에 대한 스레드 핸들을 반환합니다.
RpcSsSetClientAllocFree RpcSsSetClientAllocFree 함수는 클라이언트 스텁에서 사용하는 메모리 할당 및 해제 메커니즘을 사용하도록 설정합니다.
RpcSsSetThreadHandle RpcSsSetThreadHandle 함수는 스텁 메모리 관리 환경에 대한 스레드 핸들을 설정합니다.
RpcSsSwapClientAllocFree RpcSsSwapClientAllocFree 함수는 클라이언트 스텁에서 사용하는 메모리 할당 및 릴리스 메커니즘을 클라이언트에서 제공하는 메커니즘과 교환합니다.
RpcStringBindingCompose RpcStringBindingCompose 함수(rpcdce.h)는 문자열 바인딩 핸들을 만듭니다.

RpcStringBindingComposeA RpcStringBindingCompose 함수는 문자열 바인딩 핸들을 만듭니다. (RpcStringBindingComposeA)
RpcStringBindingComposeW RpcStringBindingComposeW(유니코드) 함수(rpcdce.h)는 문자열 바인딩 핸들을 만듭니다.
RpcStringBindingParse RpcStringBindingParse 함수(rpcdce.h)는 개체 UUID 부분과 문자열 바인딩의 주소 부분을 별도의 문자열로 반환합니다.
RpcStringBindingParseA RpcStringBindingParse 함수는 개체 UUID 부분과 문자열 바인딩의 주소 부분을 별도의 문자열로 반환합니다. (RpcStringBindingParseA)
RpcStringBindingParseW RpcStringBindingParseW(유니코드) 함수(rpcdce.h)는 개체 UUID 부분과 문자열 바인딩의 주소 부분을 별도의 문자열로 반환합니다.
RpcStringFree RpcStringFree 함수(rpcdce.h)는 RPC 런타임 라이브러리에서 할당한 문자열을 해제합니다.
RpcStringFreeA RpcStringFree 함수는 RPC 런타임 라이브러리에서 할당한 문자열을 해제합니다. (RpcStringFreeA)
RpcStringFreeW RpcStringFreeW(유니코드) 함수(rpcdce.h)는 RPC 런타임 라이브러리에서 할당한 문자 문자열을 해제합니다.
RpcTestCancel RpcTestCancel 함수는 취소 표시를 확인합니다.
RpcUserFree RpcUserFree를 변경하거나 사용할 수 없습니다.
UuidCompare 애플리케이션은 UuidCompare 함수를 호출하여 두 UUID를 비교하고 순서를 결정합니다. 반환된 값은 순서를 지정합니다.

UuidCreate UuidCreate 함수는 새 UUID를 만듭니다.
UuidCreateNil UuidCreateNil 함수는 nil 반환 UUID를 만듭니다.
UuidCreateSequential UuidCreateSequential 함수는 새 UUID를 만듭니다.
UuidEqual 애플리케이션은 UuidEqual 함수를 호출하여 두 UUID를 비교하고 같은지 여부를 확인합니다.
UuidFromString UuidFromString 함수(rpcdce.h)는 문자열을 UUID로 변환합니다.
UuidFromStringA UuidFromString 함수는 문자열을 UUID로 변환합니다. (UuidFromStringA)
UuidFromStringW UuidFromStringW(유니코드) 함수(rpcdce.h)는 문자열을 UUID로 변환합니다.
UuidHash 애플리케이션은 UuidHash 함수를 호출하여 지정된 UUID에 대한 해시 값을 생성합니다.
UuidIsNil 애플리케이션은 UuidIsNil 함수를 호출하여 지정된 UUID가 Nil 반환 UUID인지 여부를 확인합니다.
UuidToString UuidToString 함수(rpcdce.h)는 UUID를 문자열로 변환합니다.
UuidToStringA UuidToString 함수는 UUID를 문자열로 변환합니다. (UuidToStringA)
UuidToStringW UuidToStringW(유니코드) 함수(rpcdce.h)는 UUID를 문자열로 변환합니다.

구조체

MIDL_STUB_DESC MIDL_STUB_DESC 구조체는 클라이언트와 서버 간의 RPC 호출과 관련된 인터페이스 스텁에 대한 정보를 포함하는 MIDL 생성 구조체입니다.
MIDL_STUB_MESSAGE MIDL에서 생성되며 RPC 스텁의 현재 상태를 포함합니다. 애플리케이션은 MIDL_STUB_MESSAGE 구조를 직접 수정하지 않습니다.
NDR_USER_MARSHAL_INFO NDR_USER_MARSHAL_INFO 구조체는 wire_marshal 및 user_marshal 도우미 함수에 전달할 수 있는 RPC 호출의 상태에 대한 정보를 보유합니다.
NDR_USER_MARSHAL_INFO_LEVEL1 NDR_USER_MARSHAL_INFO_LEVEL1 구조는 wire_marshal 및 user_marshal 도우미 함수에 전달할 수 있는 RPC 호출의 상태에 대한 정보를 보유합니다.
ProxyFileInfo ProxyFileInfo 구조체에는 프록시 DLL의 인터페이스 프록시에 대한 정보가 포함됩니다.
RPC_ASYNC_NOTIFICATION_INFO 비동기 원격 프로시저 호출에 대한 알림 정보를 포함합니다. 이 알림 정보는 IOC(I/O 완료 포트), Windows APC(비동기 프로시저 호출), Windows 메시징 및 Windows 이벤트 알림에 대해 구성할 수 있습니다.
RPC_ASYNC_STATE RPC_ASYNC_STATE 구조체는 비동기 원격 프로시저 호출의 상태를 보유합니다. RPC_ASYNC_STATE 비동기 호출을 대기, 쿼리, 회신 또는 취소하는 데 사용되는 이 구조체에 대한 핸들입니다.
RPC_BINDING_HANDLE_OPTIONS_V1 RPC 바인딩 핸들을 만드는 추가 옵션을 포함합니다.
RPC_BINDING_HANDLE_SECURITY_V1_A RPC 바인딩 핸들을 만드는 기본 보안 옵션을 포함합니다. (ANSI)
RPC_BINDING_HANDLE_SECURITY_V1_W RPC 바인딩 핸들을 만드는 기본 보안 옵션을 포함합니다. (유니코드)

RPC_BINDING_HANDLE_TEMPLATE_V1_A RPC 바인딩 핸들을 만드는 기본 옵션을 포함합니다. (ANSI)
RPC_BINDING_HANDLE_TEMPLATE_V1_W RPC 바인딩 핸들을 만드는 기본 옵션을 포함합니다. (유니코드)
RPC_BINDING_VECTOR RPC_BINDING_VECTOR 구조에는 서버 애플리케이션이 원격 프로시저 호출을 받을 수 있는 바인딩 핸들 목록이 포함되어 있습니다.
RPC_C_OPT_COOKIE_AUTH_DESCRIPTOR RPC_C_OPT_COOKIE_AUTH_DESCRIPTOR 구조에는 RPC/HTTP 트래픽의 헤더에 삽입되는 쿠키가 포함되어 있습니다.
RPC_CALL_ATTRIBUTES_V1_A RPC_CALL_ATTRIBUTES_V1 구조체는 RpcServerInqCallAttributes 함수에 매개 변수를 제공합니다. Windows XP 및 Windows Server 2003 운영 체제용 ANSI 및 UNICODE 버전에서 구현됩니다. (ANSI)
RPC_CALL_ATTRIBUTES_V1_W RPC_CALL_ATTRIBUTES_V1 구조체는 RpcServerInqCallAttributes 함수에 매개 변수를 제공합니다. Windows XP 및 Windows Server 2003 운영 체제용 ANSI 및 UNICODE 버전에서 구현됩니다. (유니코드)
RPC_CALL_ATTRIBUTES_V2_A RPC_CALL_ATTRIBUTES_V2 구조체는 RpcServerInqCallAttributes 함수에 매개 변수를 제공합니다. 버전 2는 로컬 주소 및 클라이언트 프로세스 ID에 대한 지원을 지정합니다. (ANSI)
RPC_CALL_ATTRIBUTES_V2_W RPC_CALL_ATTRIBUTES_V2 구조체는 RpcServerInqCallAttributes 함수에 매개 변수를 제공합니다. 버전 2는 로컬 주소 및 클라이언트 프로세스 ID에 대한 지원을 지정합니다. (유니코드)
RPC_CALL_LOCAL_ADDRESS_V1 호출이 이루어진 로컬 주소에 대한 정보를 포함합니다.
RPC_CLIENT_INTERFACE RPC_CLIENT_INTERFACE 구조는 런타임 라이브러리와 스텁 간의 프라이빗 인터페이스의 일부입니다. Microsoft RPC를 사용하는 대부분의 분산 애플리케이션에는 이 구조가 필요하지 않습니다.

RPC_DISPATCH_TABLE RPC_DISPATCH_TABLE 구조는 런타임 라이브러리와 스텁 간의 프라이빗 인터페이스의 일부입니다. Microsoft RPC를 사용하는 대부분의 분산 애플리케이션에는 이 구조가 필요하지 않습니다.
RPC_EE_INFO_PARAM RPC_EE_INFO_PARAM 구조체는 확장 오류 정보를 저장하는 데 사용됩니다.
RPC_ENDPOINT_TEMPLATE RPC_ENDPOINT_TEMPLATE 구조체(rpcdce.h)는 프로토콜 시퀀스 및 이름을 포함하여 RPC 인터페이스 그룹 서버 엔드포인트의 속성을 지정합니다.
RPC_ENDPOINT_TEMPLATEA 프로토콜 시퀀스 및 이름을 포함하여 RPC 인터페이스 그룹 서버 엔드포인트의 속성을 지정합니다. (RPC_ENDPOINT_TEMPLATEA)
RPC_ENDPOINT_TEMPLATEW RPC_ENDPOINT_TEMPLATEW(유니코드) 구조체(rpcdce.h)는 프로토콜 시퀀스 및 이름을 포함하여 RPC 인터페이스 그룹 서버 엔드포인트의 속성을 지정합니다.
RPC_ERROR_ENUM_HANDLE RPC_ERROR_ENUM_HANDLE 구조체는 확장 오류 정보를 처리하기 위해 RpcError* 함수에서 사용하는 열거형 핸들을 제공합니다.
RPC_EXTENDED_ERROR_INFO RPC_EXTENDED_ERROR_INFO 구조는 확장 오류 정보를 저장하는 데 사용됩니다.
RPC_HTTP_TRANSPORT_CREDENTIALS_A RPC_HTTP_TRANSPORT_CREDENTIALS 구조는 RPC/HTTP를 사용할 때 RPC 프록시 서버에 인증하기 위한 추가 자격 증명을 정의합니다. (ANSI)
RPC_HTTP_TRANSPORT_CREDENTIALS_V2_A RPC_HTTP_TRANSPORT_CREDENTIALS_V2 구조는 RPC/HTTP를 사용할 때 RPC 프록시 서버 또는 HTTP 프록시 서버에 인증하기 위한 추가 자격 증명을 정의합니다. RPC_HTTP_TRANSPORT_CREDENTIALS_V2 HTTP 프록시 서버에 대한 인증을 허용하여 RPC_HTTP_TRANSPORT_CREDENTIALS 확장합니다. (ANSI)

RPC_HTTP_TRANSPORT_CREDENTIALS_V2_W RPC_HTTP_TRANSPORT_CREDENTIALS_V2 구조는 RPC/HTTP를 사용할 때 RPC 프록시 서버 또는 HTTP 프록시 서버에 인증하기 위한 추가 자격 증명을 정의합니다. RPC_HTTP_TRANSPORT_CREDENTIALS_V2 HTTP 프록시 서버에 대한 인증을 허용하여 RPC_HTTP_TRANSPORT_CREDENTIALS 확장합니다. (유니코드)
RPC_HTTP_TRANSPORT_CREDENTIALS_V3_A RPC_HTTP_TRANSPORT_CREDENTIALS_V3 구조는 RPC/HTTP를 사용할 때 RPC 프록시 서버 또는 HTTP 프록시 서버에 인증하기 위한 추가 자격 증명을 정의합니다. RPC_HTTP_TRANSPORT_CREDENTIALS_V3 임의의 자격 증명 양식을 사용할 수 있도록 하여 RPC_HTTP_TRANSPORT_CREDENTIALS_V2 확장합니다. (ANSI)
RPC_HTTP_TRANSPORT_CREDENTIALS_V3_W RPC_HTTP_TRANSPORT_CREDENTIALS_V3 구조는 RPC/HTTP를 사용할 때 RPC 프록시 서버 또는 HTTP 프록시 서버에 인증하기 위한 추가 자격 증명을 정의합니다. RPC_HTTP_TRANSPORT_CREDENTIALS_V3 임의의 자격 증명 양식을 사용할 수 있도록 하여 RPC_HTTP_TRANSPORT_CREDENTIALS_V2 확장합니다. (유니코드)
RPC_HTTP_TRANSPORT_CREDENTIALS_W RPC_HTTP_TRANSPORT_CREDENTIALS 구조는 RPC/HTTP를 사용할 때 RPC 프록시 서버에 인증하기 위한 추가 자격 증명을 정의합니다. (유니코드)
RPC_IF_ID RPC_IF_ID 구조체에는 인터페이스 UUID 및 인터페이스의 주 버전 및 부 버전 번호가 포함됩니다.
RPC_IF_ID_VECTOR RPC_IF_ID_VECTOR 구조에는 서버에서 제공하는 인터페이스 목록이 포함되어 있습니다.
RPC_INTERFACE_TEMPLATE RPC_INTERFACE_TEMPLATE(rpcdce.h) 구조체는 RPC 인터페이스 그룹 서버 인터페이스를 정의합니다.
RPC_INTERFACE_TEMPLATEA RPC 인터페이스 그룹 서버 인터페이스를 정의합니다. (RPC_INTERFACE_TEMPLATEA)
RPC_INTERFACE_TEMPLATEW RPC_INTERFACE_TEMPLATEW(유니코드) 구조체(rpcdce.h)는 RPC 인터페이스 그룹 서버 인터페이스를 정의합니다.

RPC_MESSAGE RPC_MESSAGE 구조에는 NDR과 RPC 또는 OLE 런타임의 나머지 부분 간에 공유되는 정보가 포함됩니다.
RPC_POLICY RPC_POLICY 구조에는 다중 호스팅 컴퓨터에서 바인딩을 결정하는 플래그와 ncacn_ip_tcp 및 ncadg_ip_udp 프로토콜을 사용할 때 포트 할당이 포함됩니다.
RPC_PROTSEQ_VECTOR RPC_PROTSEQ_VECTOR 구조에는 RPC 런타임 라이브러리가 원격 프로시저 호출을 보내고 받는 데 사용하는 프로토콜 시퀀스 목록이 포함되어 있습니다.
RPC_SECURITY_QOS RPC_SECURITY_QOS 구조는 바인딩 핸들에서 보안 서비스 품질 설정을 정의합니다. Windows 버전의 버전 가용성에 대한 주의 사항을 참조하세요.
RPC_SECURITY_QOS_V2_A RPC_SECURITY_QOS_V2 구조는 바인딩 핸들에서 버전 2 보안 서비스 품질 설정을 정의합니다. Windows 버전의 버전 가용성에 대한 주의 사항을 참조하세요. (ANSI)
RPC_SECURITY_QOS_V2_W RPC_SECURITY_QOS_V2 구조는 바인딩 핸들에서 버전 2 보안 서비스 품질 설정을 정의합니다. Windows 버전의 버전 가용성에 대한 주의 사항을 참조하세요. (유니코드)
RPC_SECURITY_QOS_V3_A RPC_SECURITY_QOS_V3 구조는 바인딩 핸들에서 버전 3 보안 서비스 품질 설정을 정의합니다. Windows 버전의 버전 가용성에 대한 주의 사항을 참조하세요. (ANSI)
RPC_SECURITY_QOS_V3_W RPC_SECURITY_QOS_V3 구조는 바인딩 핸들에서 버전 3 보안 서비스 품질 설정을 정의합니다. Windows 버전의 버전 가용성에 대한 주의 사항을 참조하세요. (유니코드)
RPC_SECURITY_QOS_V4_A RPC_SECURITY_QOS_V4 구조는 바인딩 핸들에서 버전 4 보안 서비스 품질 설정을 정의합니다. Windows 버전의 버전 가용성에 대한 주의 사항을 참조하세요. (ANSI)
RPC_SECURITY_QOS_V4_W RPC_SECURITY_QOS_V4 구조는 바인딩 핸들에서 버전 4 보안 서비스 품질 설정을 정의합니다. Windows 버전의 버전 가용성에 대한 주의 사항을 참조하세요. (유니코드)

RPC_SECURITY_QOS_V5_A RPC_SECURITY_QOS_V5 구조는 바인딩 핸들에서 버전 5 보안 서비스 품질 설정을 정의합니다. Windows 버전의 버전 가용성에 대한 주의 사항을 참조하세요. (ANSI)
RPC_SECURITY_QOS_V5_W RPC_SECURITY_QOS_V5 구조는 바인딩 핸들에서 버전 5 보안 서비스 품질 설정을 정의합니다. Windows 버전의 버전 가용성에 대한 주의 사항을 참조하세요. (유니코드)
RPC_STATS_VECTOR RPC_STATS_VECTOR 구조에는 서버별로 RPC 런타임 라이브러리의 통계가 포함됩니다.
SEC_WINNT_AUTH_IDENTITY_A SEC_WINNT_AUTH_IDENTITY 구조를 사용하면 인증을 위해 특정 사용자 이름 및 암호를 런타임 라이브러리에 전달할 수 있습니다. 이 구조는 Windows 및 Macintosh에 유효합니다. (ANSI)
SEC_WINNT_AUTH_IDENTITY_W SEC_WINNT_AUTH_IDENTITY 구조를 사용하면 인증을 위해 특정 사용자 이름 및 암호를 런타임 라이브러리에 전달할 수 있습니다. 이 구조는 Windows 및 Macintosh에 유효합니다. (유니코드)
UUID_VECTOR UUID_VECTOR 구조체에는 UUID 목록이 포함되어 있습니다.

관리자 다시 시작

아티클2023. 08. 22.

Restart Manager 기술 개요입니다.

Restart Manager를 개발하려면 다음 헤더가 필요합니다.

- restartmanager.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 관리자 다시 시작

열거형

RM_APP_STATUS Restart Manager에서 수행하는 애플리케이션의 현재 상태 설명합니다.
RM_APP_TYPE RM_PROCESS_INFO 구조에서 설명하는 애플리케이션 유형을 지정합니다.
RM_FILTER_ACTION 다시 시작 또는 종료 작업에 적용되는 수정 유형을 지정합니다.
RM_FILTER_TRIGGER 애플리케이션 또는 서비스에 대한 다시 시작 또는 종료 작업에 대해 설명합니다.
RM_REBOOT_REASON 시스템을 다시 시작해야 하는 이유를 설명합니다.
RM_SHUTDOWN_TYPE 애플리케이션의 종료를 구성합니다.

Functions

RM_WRITE_STATUS_CALLBACK RM_WRITE_STATUS_CALLBACK 함수는 Restart Manager를 제어하는 사용자 인터페이스에서 구현할 수 있습니다.
RmAddFilter 애플리케이션 또는 서비스에 적용되는 종료 또는 다시 시작 작업을 수정합니다.
RmCancelCurrentTask 현재 RmShutdown 또는 RmRestart 작업을 취소합니다. 이 함수는 RmStartSession 함수를 호출하여 세션을 시작한 애플리케이션에서 호출해야 합니다.
RmEndSession 다시 시작 관리자 세션을 종료합니다.
RmGetFilterList RmAddFilter 함수에서 이미 적용한 종료 및 다시 시작 작업에 대한 수정 사항을 나열합니다.
RmGetList 현재 다시 시작 관리자 세션에 등록된 리소스를 사용하는 모든 애플리케이션 및 서비스의 목록을 가져옵니다.
RmJoinSession 보조 설치 관리자를 기존 다시 시작 관리자 세션에 조인합니다.
RmRegisterResources 리소스를 다시 시작 관리자 세션에 등록합니다.
RmRemoveFilter RmAddFilter 함수를 사용하여 적용된 작업을 종료하거나 다시 시작하도록 수정한 내용을 제거합니다.
RmRestart RmShutdown 함수에 의해 종료되고 RegisterApplicationRestart 함수를 사용하여 다시 시작되도록 등록된 애플리케이션 및 서비스를 다시 시작합니다.
RmShutdown 애플리케이션의 종료를 시작합니다.

RmStartSession 새 다시 시작 관리자 세션을 시작합니다.

구조체

RM_FILTER_INFO 다시 시작 또는 종료 작업을 수정하는 방법에 대한 정보를 포함합니다.
RM_PROCESS_INFO Restart Manager에 등록할 애플리케이션에 대해 설명합니다.
RM_UNIQUE_PROCESS PID 및 프로세스가 시작된 시간을 통해 프로세스를 고유하게 식별합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

라우팅 및 원격 액세스 서비스

아티클2023. 03. 16.

라우팅 및 원격 액세스 서비스 기술에 대한 개요입니다.

라우팅 및 원격 액세스 서비스를 개발하려면 다음 헤더가 필요합니다.

- ipxrtdef.h
- mgm.h
- mprapi.h
- ras.h
- rasdlg.h
- rasshost.h
- routprot.h
- rtmv2.h
- stm.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 라우팅 및 원격 액세스 서비스

열거형

MGM_ENUM_TYPES MGM_ENUM_TYPES 열거형에는 멀티캐스트 그룹 관리자가 사용하는 그룹 열거형 유형이 나열됩니다. 이 구조체는 MgmGroupEnumerationStart 함수에서 사용됩니다.
MPRAPI_OBJECT_TYPE MPRAPI_OBJECT_HEADER 구조체의 구조체 형식을 지정합니다.
RAS_HARDWARE_CONDITION RAS_HARDWARE_CONDITION 열거형 유형은 지정된 RAS 포트에 대한 하드웨어 상태 정보를 지정합니다.
RAS_PORT_CONDITION RAS_PORT_CONDITION 열거형 형식은 지정된 RAS 포트의 연결 조건에 대한 정보를 지정합니다.

RAS_QUARANTINE_STATE RAS_QUARANTINE_STATE 열거형 형식은 클라이언트 연결의 격리 상태를 나타냅니다.
RASPROJECTION_INFO_TYPE RAS_PROJECTION_INFO 구조체에서 RASPPP_PROJECTION_INFO 또는 RASIKEV2_PROJECTION_INFO 구조를 나타내는 데 사용됩니다.
ROUTER_CONNECTION_STATE 라우터에서 인터페이스의 가능한 상태를 열거합니다.
ROUTER_INTERFACE_TYPE ROUTER_INTERFACE_TYPE 형식은 라우터에서 다양한 종류의 인터페이스를 열거합니다.
RTM_EVENT_TYPE RTM_EVENT_TYPE 열거형은 라우팅 테이블 관리자가 RTM_EVENT_CALLBACK 콜백 사용에 대해 클라이언트에 알릴 수 있는 이벤트를 열거합니다.

Functions

_ENTITY_METHOD RTM_ENTITY_EXPORT_METHOD 콜백은 클라이언트에서 내보낸 모든 메서드의 프로토타입입니다.
_EVENT_CALLBACK RTM_EVENT_CALLBACK 콜백은 라우팅 테이블 관리자가 클라이언트에 지정된 이벤트가 발생했음을 알리는 데 사용됩니다.
MgmAddGroupMembershipEntry MgmAddGroupMembershipEntry 함수는 지정된 인터페이스에 지정된 그룹에 대한 새 수신기가 있음을 멀티캐스트 그룹 관리자에게 알릴 수 있습니다.
MgmDeleteGroupMembershipEntry MgmDeleteGroupMembershipEntry 함수는 멀티캐스트 그룹 관리자에게 지정된 인터페이스에 지정된 그룹에 대한 수신기가 더 이상 없음을 알릴 수 있습니다.

MgmDeRegisterMProtocol MgmDeRegisterMProtocol 함수는 MgmRegisterMProtocol 호출에서 얻은 클라이언트 핸들의 등록을 취소합니다.
MgmGetFirstMfe MgmGetFirstMfe 함수는 MFE 목록의 시작 부분에서 MFE를 검색합니다.
MgmGetFirstMfeStats MgmGetFirstMfeStats 함수는 MFE 목록의 시작 부분에서 시작하는 MFE 통계를 검색합니다.
MgmGetMfe MgmGetMfe 함수는 특정 MFE를 검색합니다.
MgmGetMfeStats MgmGetMfeStats 함수는 특정 MFE에 대한 통계를 검색합니다. 반환되는 통계에는 수신된 패킷, 수신된 바이트 및 나가는 각 인터페이스에서 전달된 패킷이 포함됩니다.
MgmGetNextMfe MgmGetNextMfe 함수는 하나 이상의 MFE를 검색합니다.
MgmGetNextMfeStats MgmGetNextMfeStats 함수는 하나 이상의 MFE 통계 집합을 검색합니다.
MgmGetProtocolOnInterface MgmGetProtocolOnInterface 함수는 지정된 인터페이스를 소유하는 멀티캐스트 라우팅 프로토콜의 프로토콜 ID를 검색합니다.
MgmGroupEnumerationEnd MgmGroupEnumerationEnd 함수는 MgmGroupEnumerationStart에 대한 이전 호출에서 가져온 지정된 열거형 핸들을 해제합니다.
MgmGroupEnumerationGetNext MgmGroupEnumerationGetNext 함수는 다음 그룹 항목 집합을 검색합니다. 이 함수에서 반환되는 정보는 조인된 그룹 목록과 요청된 원본(있는 경우)입니다.

MgmGroupEnumerationStart MgmGroupEnumerationStart 함수는 나중에 조인된 그룹 목록을 가져오는 데 사용되는 열거형 핸들을 가져옵니다. 클라이언트가 핸들을 가져온 후 MgmGroupEnumerationGetNext 함수를 사용하여 그룹을 열거해야 합니다.
MgmRegisterMProtocol MgmRegisterMProtocol 함수는 클라이언트에서 멀티캐스트 그룹 관리자에 등록하는 데 사용됩니다.
MgmReleaseInterfaceOwnership MgmReleaseInterfaceOwnership 함수는 클라이언트가 인터페이스의 소유권을 포기하는 데 사용됩니다. 이 함수가 호출되면 클라이언트를 대신하여 멀티캐스트 그룹 관리자가 유지 관리하는 모든 MFE와 지정된 인터페이스에 대해 유지 관리되는 모든 MFE가 삭제됩니다.
MgmSetMfe MgmSetMfe 함수는 MFE에 대한 업스트림 인접 항목을 변경합니다. MFE에는 수신하는 인터페이스와 멀티캐스트 데이터를 전달하는 인터페이스에 대한 정보가 포함되어 있습니다.
MgmTakeInterfaceOwnership MgmTakeInterfaceOwnership 함수는 인터페이스에서 사용하도록 설정된 경우 클라이언트(예: 라우팅 프로토콜)에 의해 호출됩니다.
MprAdminAcceptNewConnection 원격 액세스 서비스는 새 사용자가 전화를 걸어 RAS 인증을 성공적으로 완료할 때마다 MprAdminAcceptNewConnection 함수를 호출합니다. MprAdminAcceptNewConnection은 사용자가 연결할 수 있는지 여부를 결정합니다.
MprAdminAcceptNewConnection2 원격 액세스 서비스는 새 사용자가 전화를 걸어 RAS 인증을 성공적으로 완료할 때마다 MprAdminAcceptNewConnection2 함수를 호출합니다. MprAdminAcceptNewConnection2는 사용자가 연결할 수 있는지 여부를 결정합니다.
MprAdminAcceptNewConnection3 원격 액세스 서비스는 새 사용자가 전화를 걸어 RAS 인증을 성공적으로 완료할 때마다 MprAdminAcceptNewConnection3 함수를 호출합니다. MprAdminAcceptNewConnection3은 사용자가 연결할 수 있는지 여부를 결정합니다.

MprAdminAcceptNewConnectionEx RAS(원격 액세스 서비스)는 새 사용자가 전화를 걸어 RAS 인증을 성공적으로 완료할 때마다 MprAdminAcceptNewConnectionEx 함수를 호출합니다. MprAdminAcceptNewConnectionEx는 사용자가 연결할 수 있는지 여부를 결정합니다.
MprAdminAcceptNewLink RAS(원격 액세스 서비스)는 특정 연결에 대한 링크를 만들 때마다 MprAdminAcceptNewLink 함수를 호출합니다.
MprAdminAcceptReauthentication 원격 액세스 서비스는 클라이언트의 격리 상태가 변경 될 때마다 MprAdminAcceptReauthentication 함수를 호출합니다.
MprAdminAcceptReauthenticationEx RAS(원격 액세스 서비스)는 클라이언트의 격리 상태가 변경 될 때마다 MprAdminAcceptReauthenticationEx 함수를 호출합니다.
MprAdminBufferFree MprAdminBufferFree 함수는 _MprAdminDeviceEnum, MprAdminInterfaceGetInfo, MprAdminInterfaceDeviceGetInfo, MprAdminInterfaceEnum, MprAdminServerGetInfo, MprAdminInterfaceTransportGetInfo 및 MprAdminTransportGetInfo에서 반환된 메모리 버퍼를 해제합니다.
MprAdminConnectionClearStats MprAdminConnectionClearStats 함수는 지정된 연결에 대한 통계 카운터를 다시 설정합니다.
MprAdminConnectionEnum MprAdminConnectionEnum 함수는 모든 활성 연결을 열거합니다.
MprAdminConnectionEnumEx MprAdminConnectionEnumEx 함수는 지정된 RRAS 서버에 대한 활성 연결을 열거합니다.
MprAdminConnectionGetInfo 특정 연결에 대한 데이터를 검색합니다.
MprAdminConnectionGetInfoEx 지정된 RRAS 서버의 특정 연결에 대한 연결 정보를 검색합니다.

MprAdminConnectionHangupNotification 원격 액세스 서비스는 지정된 연결에 대한 마지막 링크가 해체된 후 MprAdminConnectionHangupNotification 함수를 호출합니다.
MprAdminConnectionHangupNotification2 원격 액세스 서비스는 지정된 연결에 대한 마지막 링크가 해체된 후 MprAdminConnectionHangupNotification2 함수를 호출합니다.
MprAdminConnectionHangupNotification3 원격 액세스 서비스는 지정된 연결에 대한 마지막 링크가 해체된 후 MprAdminConnectionHangupNotification3 함수를 호출합니다.
MprAdminConnectionHangupNotificationEx RAS(원격 액세스 서비스)는 지정된 연결에 대한 마지막 링크가 해체된 후 MprAdminConnectionHangupNotificationEx 함수를 호출합니다.
MprAdminConnectionRemoveQuarantine 필터가 IAS(인터넷 인증 서비스) 정책의 결과로 적용된 경우 MprAdminConnectionRemoveQuarantine 함수는 전화 접속된 RAS 클라이언트에서 격리 필터를 제거합니다.
MprAdminDeregisterConnectionNotification MprAdminDeregisterConnectionNotification 함수는 MprAdminRegisterConnectionNotification을 사용하여 이전에 등록된 이벤트 개체의 등록을 취소합니다. 등록을 취소하면 인터페이스가 연결되거나 연결이 끊어지면 이 이벤트가 더 이상 신호를 받지 않습니다.
MprAdminDeviceEnum 함수는 이름과 형식을 반환할 수 있는 컴퓨터에 설치된 RAS 지원 디바이스를 열거하기 위해 호출됩니다.
MprAdminEstablishDomainRasServer MprAdminEstablishDomainRasServer 함수는 지정된 컴퓨터를 도메인의 원격 액세스 서버로 설정합니다. 이 함수는 도메인에 가입된 컴퓨터에서만 실행되어야 합니다.
MprAdminGetErrorString MprAdminGetErrorString 함수는 Mprerror.h의 라우터 오류와 연결된 문자열을 반환합니다.

MprAdminGetIpAddressForUser RAS는 IP 주소가 필요한 각 사용자에 대해 MprAdminGetIpAddressForUser 함수를 한 번 호출합니다.
MprAdminGetIpv6AddressForUser RAS는 IPv6 주소가 필요한 각 사용자에 대해 MprAdminGetIpv6AddressForUser 함수를 한 번 호출합니다.
MprAdminGetPDCServer MprAdminGetPDCServer 함수는 도메인 이름 또는 서버 이름에서 마스터 UAS(사용자 계정 하위 시스템)가 있는 서버의 이름을 검색합니다. 도메인 이름 매개 변수 또는 서버 이름 매개 변수는 NULL일 수 있지만 둘 다일 수는 없습니다.
MprAdminInitializeDll RRAS(라우팅 및 원격 액세스 서비스)가 시작되면 관리 DLL에서 내보낸 MprAdminInitializeDll 함수를 호출합니다. 이 함수를 사용하여 DLL에 필요한 초기화를 수행합니다.
MprAdminInitializeDllEx RRAS(라우팅 및 원격 액세스 서비스)가 시작되면 관리 DLL에서 내보낸 MprAdminInitializeDll 함수를 호출합니다.
MprAdminInterfaceConnect MprAdminInterfaceConnect 함수는 지정된 WAN 인터페이스에 대한 연결을 만듭니다.
MprAdminInterfaceCreate MprAdminInterfaceCreate 함수는 지정된 서버에 인터페이스를 만듭니다.
MprAdminInterfaceDelete MprAdminInterfaceDelete 함수는 지정된 서버에서 인터페이스를 삭제합니다.
MprAdminInterfaceDeviceGetInfo MprAdminInterfaceDeviceGetInfo는 라우터 수요 다이얼 인터페이스에 사용되는 디바이스에 대한 정보를 검색합니다.
MprAdminInterfaceDeviceSetInfo MprAdminInterfaceDeviceSetInfo는 라우터 수요 다이얼 인터페이스에 사용되는 디바이스를 만들거나 수정합니다.

MprAdminInterfaceDisconnect MprAdminInterfaceDisconnect 함수는 연결된 WAN 인터페이스의 연결을 끊습니다.
MprAdminInterfaceEnum MprAdminInterfaceEnum 함수는 지정된 서버의 모든 인터페이스를 열거합니다.
MprAdminInterfaceGetCredentials MprAdminInterfaceGetCredentials 함수를 사용하여 지정된 요청 다이얼 인터페이스에서 전화를 걸기 위한 도메인, 사용자 이름 및 암호를 검색합니다.
MprAdminInterfaceGetCredentialsEx MprAdminInterfaceGetCredentialsEx 함수를 사용하여 지정된 인터페이스에 대한 확장 자격 증명 정보를 검색합니다. 이 함수를 사용하여 EAP(Extensible Authentication Protocols)에 사용되는 자격 증명 정보를 검색합니다.
MprAdminInterfaceGetCustomInfoEx 지정된 서버에서 지정된 수요 다이얼 인터페이스에 대한 터널별 구성을 검색합니다.
MprAdminInterfaceGetHandle MprAdminInterfaceGetHandle 함수는 지정된 인터페이스에 대한 핸들을 검색합니다.
MprAdminInterfaceGetInfo MprAdminInterfaceGetInfo 함수는 지정된 서버에서 지정된 인터페이스에 대한 정보를 검색합니다.
MprAdminInterfaceQueryUpdateResult MprAdminInterfaceQueryUpdateResult 함수는 지정된 라우터 관리자에게 마지막 요청의 결과를 반환하여 인터페이스에 대한 경로를 업데이트합니다. 자세한 내용은 MprAdminInterfaceUpdateRoutes를 참조하세요.
MprAdminInterfaceSetCredentials MprAdminInterfaceSetCredentials 함수를 사용하여 지정된 수요 다이얼 인터페이스에서 전화 걸기에 사용할 도메인, 사용자 이름 및 암호를 설정합니다.
MprAdminInterfaceSetCredentialsEx MprAdminInterfaceSetCredentialsEx 함수를 사용하여 인터페이스에 대한 확장 자격 증명 정보를 설정합니다. 이 함수를 사용하여 EAP(Extensible Authentication Protocols)에 사용되는 자격 증명 정보를 설정합니다.

MprAdminInterfaceSetCustomInfoEx 지정된 서버의 지정된 수요 다이얼 인터페이스에 대한 터널별 사용자 지정 구성을 설정합니다.
MprAdminInterfaceSetInfo MprAdminInterfaceSetInfo 함수는 지정된 서버의 지정된 인터페이스에 대한 정보를 설정합니다.
MprAdminInterfaceTransportAdd MprAdminInterfaceTransportAdd 함수는 지정된 인터페이스에 전송(예: IP 또는 IPX)을 추가합니다.
MprAdminInterfaceTransportGetInfo MprAdminInterfaceTransportGetInfo 함수는 지정된 인터페이스에서 실행되는 전송에 대한 정보를 검색합니다.
MprAdminInterfaceTransportRemove MprAdminInterfaceTransportRemove 함수는 지정된 인터페이스에서 전송(예: IP 또는 IPX)을 제거합니다.
MprAdminInterfaceTransportSetInfo MprAdminInterfaceTransportSetInfo 함수는 지정된 인터페이스에서 실행되는 전송에 대한 정보를 설정합니다.
MprAdminInterfaceUpdatePhonebookInfo MprAdminInterfaceUpdatePhonebookInfo 함수는 라우터가 지정된 수요 다이얼 인터페이스에서 변경한 내용을 선택하도록 강제합니다. 요청 전화 접속 인터페이스에 대한 전화 번호부 항목이 변경된 후 이 함수를 호출합니다.
MprAdminInterfaceUpdateRoutes MprAdminInterfaceUpdateRoutes 함수는 지정된 라우터 관리자에게 지정된 인터페이스에 대한 라우팅 정보를 업데이트하도록 요청합니다.
MprAdminIsDomainRasServer MprAdminIsDomainRasServer 함수는 지정된 컴퓨터가 도메인의 원격 액세스 서버로 등록되었는지 여부에 대한 정보를 반환합니다.
MprAdminIsServiceInitialized MprAdminIsServiceInitialized 함수는 호출 프로세스에 액세스 권한이 있는 경우 RRAS 서비스가 지정된 서버에서 실행 중인지 여부를 확인합니다.

MprAdminIsServiceRunning MprAdminIsServiceRunning 함수는 호출 프로세스에 액세스 권한이 있는 경우 RRAS 서비스가 지정된 서버에서 실행 중인지 여부를 확인합니다.
MprAdminLinkHangupNotification RAS는 특정 연결에 대한 링크가 해체될 때마다 MprAdminLinkHangupNotification 함수를 호출합니다.
MprAdminMIBBufferFree MprAdminMIBBufferFree 함수는 MprAdminMIBEntryGet, MprAdminMIBEntryGetFirst, MprAdminMIBEntryGetNext 함수에서 반환된 버퍼를 해제합니다.
MprAdminMIBEntryCreate MprAdminMIBEntryCreate 함수는 라우팅 프로토콜 또는 라우터 관리자가 내보낸 변수 중 하나에 대한 항목을 만듭니다.
MprAdminMIBEntryDelete MprAdminMIBEntryDelete 함수는 라우팅 프로토콜 또는 라우터 관리자가 내보낸 변수 중 하나에 대한 항목을 삭제합니다.
MprAdminMIBEntryGet MprAdminMIBEntryGet 함수는 라우팅 프로토콜 또는 라우터 관리자가 내보낸 변수 중 하나의 값을 검색합니다.
MprAdminMIBEntryGetFirst MprAdminMIBEntryGetFirst 함수는 프로토콜 또는 라우터 관리자가 내보낸 일부 변수 집합의 첫 번째 변수를 검색합니다. 호출을 서비스하는 모듈이 먼저 정의합니다.
MprAdminMIBEntryGetNext MprAdminMIBEntryGetNext 함수는 프로토콜 또는 라우터 관리자가 내보낸 일부 변수 집합의 다음 변수를 검색합니다. 호출을 서비스하는 모듈은 다음에 정의합니다.
MprAdminMIBEntrySet MprAdminMIBEntrySet 함수는 라우팅 프로토콜 또는 라우터 관리자가 내보낸 변수 중 하나의 값을 설정합니다.
MprAdminMIBGetTrapInfo MprAdminMIBGetTrapInfo 함수는 트랩에 대한 자세한 정보를 위해 트랩 이벤트를 설정하는 모듈을 쿼리합니다.

MprAdminMIBServerConnect MprAdminMIBServerConnect 함수는 관리되는 라우터에 대한 연결을 설정합니다. 이 호출은 서버에 대한 다른 호출 전에 수행해야 합니다. 이 함수에서 반환된 핸들은 후속 MIB 호출에 사용됩니다.
MprAdminMIBServerDisconnect MprAdminMIBServerDisconnect 함수는 MprAdminMIBServerConnect에 대한 이전 호출로 인한 연결의 연결을 끊습니다.
MprAdminMIBSetTrapInfo MprAdminMIBSetTrapInfo 함수는 TRAP을 발급해야 할 때마다 신호를 수신하는 이벤트에 대한 핸들을 지정합니다.
MprAdminPortClearStats MprAdminPortClearStats 함수는 지정된 포트에 대한 통계를 다시 설정합니다.
MprAdminPortDisconnect MprAdminPortDisconnect 함수는 특정 포트에서 연결을 끊습니다.
MprAdminPortEnum 특정 연결의 모든 활성 포트 또는 RAS에서 사용하거나 현재 사용할 수 있는 모든 포트를 열거합니다.
MprAdminPortGetInfo MprAdminPortGetInfo 함수는 특정 포트에 대한 정보를 가져옵니다.
MprAdminPortReset MprAdminPortReset 함수는 지정된 포트에 연결된 통신 디바이스를 다시 설정합니다.
MprAdminRegisterConnectionNotification MprAdminRegisterConnectionNotification 함수는 DDM(Demand Dial Manager)에 이벤트 개체를 등록하므로 인터페이스가 연결되거나 연결이 끊어지면 이벤트가 신호를 보냅니다.
MprAdminReleaseIpAddress MprAdminReleaseIpAddress 함수는 사용자가 연결을 끊고 사용자의 IP 주소가 해제될 때 호출됩니다.

MprAdminReleaseIpv6AddressForUser MprAdminReleaseIpv6AddressForUser 함수는 IPv6 주소를 해제해야 하는 각 사용자에 대해 한 번 호출됩니다.
MprAdminSendUserMessage MprAdminSendUserMessage 함수는 지정된 연결에 연결된 사용자에게 메시지를 보냅니다.
MprAdminServerConnect MprAdminServerConnect 함수는 해당 라우터를 관리하기 위해 라우터에 대한 연결을 설정합니다.
MprAdminServerDisconnect MprAdminServerDisconnect 함수는 MprAdminServerConnect에 대한 이전 호출로 인한 연결의 연결을 끊습니다.
MprAdminServerGetCredentials MprAdminServerGetCredentials 함수는 지정된 서버에 대한 미리 공유된 키를 검색합니다.
MprAdminServerGetInfo MprAdminServerGetInfo 함수는 지정된 RRAS 서버에 대한 정보를 검색합니다.
MprAdminServerGetInfoEx MprAdminServerGetInfoEx 함수는 지정된 RRAS 서버에 대한 포트 정보를 검색합니다.
MprAdminServerSetCredentials MprAdminServerSetCredentials 함수는 지정된 서버에 대한 미리 공유된 키를 설정합니다.
MprAdminServerSetInfo MprAdminServerSetInfo 함수는 RRAS 서비스가 실행 중일 때 L2TP, PPTP 및 SSTP 디바이스에 대한 포트 수를 설정하는 데 사용됩니다.
MprAdminServerSetInfoEx MprAdminServerSetInfoEx 함수는 지정된 RRAS 서버에서 포트 정보를 설정합니다.
MprAdminTerminateDll RAS가 종료되면 관리 DLL에서 내보낸 MprAdminTerminateDll 함수를 호출합니다. 이 함수를 사용하여 DLL에 필요한 정리를 수행합니다.

MprAdminTransportCreate MprAdminTransportCreate 함수는 새 전송을 로드하고 전송에 대한 라우터 관리자를 시작합니다.
MprAdminTransportGetInfo MprAdminTransportGetInfo 함수는 지정된 전송에 대한 전역 정보, 기본 클라이언트 인터페이스 정보 또는 둘 다를 검색합니다.
MprAdminTransportSetInfo MprAdminTransportSetInfo 함수는 지정된 전송에 대해 전역 정보 또는 기본 클라이언트 인터페이스 정보 또는 둘 다를 설정합니다.
MprAdminUserGetInfo MprAdminUserGetInfo 함수는 특정 사용자에 대한 모든 RAS 정보를 검색합니다.
MprAdminUserSetInfo MprAdminUserSetInfo 함수는 지정된 사용자에 대한 RAS 정보를 설정합니다.
MprConfigBufferFree MprConfigBufferFree 함수는 버퍼를 해제합니다. MprConfigXEnum, MprConfigXGetInfo
MprConfigFilterGetInfo 지정된 전송 프로토콜 형식에 대한 정적 필터링 정보를 반환합니다.
MprConfigFilterSetInfo 지정된 전송 프로토콜 형식에 대한 정적 필터링 정보를 설정합니다.
MprConfigGetFriendlyName MprConfigGetFriendlyName 함수는 지정된 GUID 이름에 해당하는 인터페이스의 이름을 반환합니다.
MprConfigGetGuidName MprConfigGetGuidName 함수는 지정된 이름에 해당하는 인터페이스의 GUID 이름을 반환합니다.
MprConfigInterfaceCreate MprConfigInterfaceCreate 함수는 지정된 라우터 구성에서 라우터 인터페이스를 만듭니다.

MprConfigInterfaceDelete MprConfigInterfaceDelete 함수는 라우터 구성에서 라우터 인터페이스를 제거합니다. 이 인터페이스와 연결된 모든 전송 정보도 제거됩니다.
MprConfigInterfaceEnum MprConfigInterfaceEnum 함수는 라우터에 대해 구성된 인터페이스를 열거합니다.
MprConfigInterfaceGetCustomInfoEx 지정된 인터페이스에 대한 사용자 지정 IKEv2 정책 구성을 검색합니다.
MprConfigInterfaceGetHandle MprConfigInterfaceGetHandle 함수는 지정된 라우터 구성에서 지정된 인터페이스의 구성에 대한 핸들을 검색합니다.
MprConfigInterfaceGetInfo MprConfigInterfaceGetInfo 함수는 라우터에서 지정된 인터페이스에 대한 구성을 검색합니다.
MprConfigInterfaceSetCustomInfoEx 지정된 인터페이스에 대한 사용자 지정 IKEv2 정책 구성을 설정합니다.
MprConfigInterfaceSetInfo MprConfigInterfaceSetInfo 함수는 지정된 인터페이스에 대한 구성을 설정합니다.
MprConfigInterfaceTransportAdd MprConfigInterfaceTransportAdd 함수는 라우터의 인터페이스 구성에 전송 프로토콜을 추가합니다.
MprConfigInterfaceTransportEnum MprConfigInterfaceTransportEnum 함수는 지정된 인터페이스에 구성된 전송을 열거합니다.
MprConfigInterfaceTransportGetHandle MprConfigInterfaceTransportGetHandle 함수는 지정된 라우터 구성에서 인터페이스의 전송 구성에 대한 핸들을 검색합니다.
MprConfigInterfaceTransportGetInfo MprConfigInterfaceTransportGetInfo 함수는 지정된 인터페이스에서 지정된 클라이언트에 대한 구성 정보를 검색합니다.

MprConfigInterfaceTransportRemove MprConfigInterfaceTransportRemove 함수는 라우터의 지정된 인터페이스 구성에서 지정된 전송을 제거합니다.
MprConfigInterfaceTransportSetInfo MprConfigInterfaceTransportSetInfo 함수는 지정된 인터페이스 및 전송 프로토콜에서 클라이언트에 대한 구성 정보를 업데이트합니다.
MprConfigServerBackup MprConfigServerBackup 함수는 라우터에 대한 라우터 관리자, 인터페이스 및 전화 번호부 구성의 백업을 만듭니다.
MprConfigServerConnect MprConfigServerConnect 함수는 구성할 라우터에 연결합니다.
MprConfigServerDisconnect MprConfigServerDisconnect 함수는 MprConfigServerConnect에 대한 이전 호출로 인한 연결의 연결을 끊습니다.
MprConfigServerGetInfo MprConfigServerGetInfo 함수는 지정된 라우터에 대한 서버 수준 구성 정보를 검색합니다.
MprConfigServerGetInfoEx MprConfigServerGetInfoEx 함수는 지정된 서버에 대한 포트 정보를 검색합니다.
MprConfigServerInstall MprConfigServerInstall 함수는 라우팅 및 원격 액세스 서비스를 기본 구성으로 구성합니다.
MprConfigServerRestore MprConfigServerRestore 함수는 MprConfigServerBackup에 대한 이전 호출에서 만든 백업에서 라우터 관리자, 인터페이스 및 전화 번호부 구성을 복원합니다.
MprConfigServerSetInfo MprConfigServerSetInfo 함수는 L2TP, PPTP 및 SSTP 포트에 대한 포트 수를 설정하고 RRAS 서비스가 실행되지 않을 때 레지스트리에서 RRAS를 사용하거나 사용하지 않도록 설정하여 다음에 시스템을 다시 시작할 때 선택할 수 있도록 하는 데 사용됩니다.

MprConfigServerSetInfoEx MprConfigServerSetInfoEx 함수는 지정된 RRAS 서버에서 포트 정보를 설정합니다.
MprConfigTransportCreate MprConfigTransportCreate 함수는 지정된 전송을 지정된 라우터 구성에 있는 전송 프로토콜 목록에 추가합니다.
MprConfigTransportDelete MprConfigTransportDelete 함수는 지정된 라우터 구성에 있는 전송 목록에서 지정된 전송을 제거합니다.
MprConfigTransportEnum MprConfigTransportEnum 함수는 라우터에 구성된 전송을 열거합니다.
MprConfigTransportGetHandle MprConfigTransportGetHandle 함수는 지정된 라우터 구성에서 지정된 전송 프로토콜의 구성에 대한 핸들을 검색합니다.
MprConfigTransportGetInfo MprConfigTransportGetInfo 함수는 라우터에서 지정된 전송 프로토콜에 대한 구성을 검색합니다.
MprConfigTransportSetInfo MprConfigTransportSetInfo 함수는 지정된 라우터 구성에서 지정된 전송 프로토콜에 대한 구성을 변경합니다.
MprInfoBlockAdd MprInfoBlockAdd 함수는 새 블록을 추가하여 기존 헤더와 동일한 새 헤더를 만듭니다.
MprInfoBlockFind MprInfoBlockFind 함수는 정보 헤더에서 지정된 블록을 찾고 블록에 대한 정보를 검색합니다.
MprInfoBlockQuerySize MprInfoBlockQuerySize 함수는 정보 헤더의 크기를 반환합니다.
MprInfoBlockRemove MprInfoBlockRemove 함수는 지정된 블록이 제거된 기존 헤더와 동일한 새 헤더를 만듭니다.

MprInfoBlockSet MprInfoBlockSet은 지정된 블록이 수정된 기존 헤더와 동일한 새 헤더를 만듭니다.
MprInfoCreate MprInfoCreate 함수는 새 정보 헤더를 만듭니다.
MprInfoDelete MprInfoDelete 함수는 MprInfoCreate를 사용하여 만들거나 MprInfoBlockAdd, MprInfoBlockRemove 또는 MprInfoBlockSet에서 검색한 정보 헤더를 삭제합니다.
MprInfoDuplicate MprInfoDuplicate 함수는 기존 정보 헤더를 복제합니다.
MprInfoRemoveAll MprInfoRemoveAll 함수는 지정된 헤더에서 모든 정보 블록을 제거합니다.
ORASADFUNC ORASADFunc 함수는 자동 완성을 위해 사용자 지정된 사용자 인터페이스를 제공하는 데 사용되는 애플리케이션 정의 콜백 함수입니다.
PADD_INTERFACE AddInterface 함수는 라우팅 프로토콜에서 관리할 인터페이스를 추가합니다.
PBLOCK_CONVERT_SERVICES_TO_STATIC BlockConvertServicesToStatic 함수는 지정된 인터페이스에서 받은 모든 서비스를 정적으로 변환합니다.
PBLOCK_DELETE_STATIC_SERVICES BlockDeleteStaticServices 함수는 지정된 인터페이스와 연결된 모든 정적 서비스를 삭제합니다.
PCLOSE_SERVICE_ENUMERATION_HANDLE CloseServiceEnumerationHandle 함수는 열거형을 종료하고 연결된 리소스를 해제합니다.
PCONNECT_CLIENT 라우터 관리자는 클라이언트가 라우팅 프로토콜이 실행 중인 인터페이스에 연결할 때 ConnectClient 함수를 호출합니다.

PCREATE_SERVICE_ENUMERATION_HANDLE CreateServiceEnumerationHandle 함수는 빠르고 변경 허용 열거형 함수를 사용할 수 있는 핸들을 반환합니다.
PCREATE_STATIC_SERVICE CreateStaticService 함수는 테이블에 정적 서비스를 추가합니다.
PDELETE_INTERFACE DeleteInterface 함수는 라우팅 프로토콜에 의해 관리되는 집합에서 인터페이스를 제거합니다.
PDELETE_STATIC_SERVICE DeleteStaticService 함수는 테이블에서 정적 서비스를 삭제합니다.
PDISCONNECT_CLIENT 라우터 관리자는 클라이언트가 라우팅 프로토콜이 실행 중인 인터페이스에서 연결을 끊을 때 DisconnectClient 함수를 호출합니다.
PDO_UPDATE_ROUTES DoUpdateRoutes 함수는 지정된 인터페이스를 통해 라우팅 정보 업데이트를 수행하여 정적 경로 정보를 얻기 위해 라우팅 프로토콜을 요청합니다.
PENUMERATE_GET_NEXT_SERVICE EnumerateGetNextService 함수는 CreateServiceEnumerationHandle에서 시작된 열거형에서 다음 서비스 항목을 반환합니다.
PFNRASFREEBUFFER 사용자 지정 스크립팅 DLL은 RasFreeBuffer를 호출하여 RasGetBuffer에 대한 이전 호출에 의해 할당된 메모리 버퍼를 해제합니다.
PFNRASGETBUFFER 사용자 지정 스크립팅 DLL은 RasGetBuffer를 호출하여 서버에 연결된 포트를 통해 데이터를 보내거나 받기 위한 메모리를 할당합니다.
PFNRASRECEIVEBUFFER 사용자 지정 스크립팅 DLL은 RasReceiveBuffer 함수를 호출하여 지정된 포트를 통해 서버에서 데이터를 받을 준비가 되었음을 RAS에 알릴 수 있습니다.

PFNRASRETRIEVEBUFFER 사용자 지정 스크립팅 DLL은 RasRetrieveBuffer 함수를 호출하여 지정된 포트를 통해 RAS 서버에서 받은 데이터를 가져옵니다.
PFNRASSENDBUFFER 사용자 지정 스크립팅 DLL은 RasSendBuffer 함수를 호출하여 지정된 포트를 통해 서버에 데이터를 보냅니다.
PFNRASSETCOMMSETTINGS 사용자 지정 스크립팅 DLL에서 RasSetCommSettings를 호출하여 연결에 대한 포트의 설정을 변경합니다.
PGET_EVENT_MESSAGE GetEventMessage 함수는 라우팅 프로토콜의 메시지 큐에서 항목을 가져옵니다. 라우팅 프로토콜은 큐를 사용하여 라우터 관리자에게 비동기 이벤트를 알릴 수 있습니다.
PGET_FIRST_ORDERED_SERVICE GetFirstOrderedService 함수는 테이블의 지정된 서비스 하위 집합에서 지정된 순서로 첫 번째 서비스를 반환합니다.
PGET_GLOBAL_INFO GetGlobalInfo 함수는 라우팅 프로토콜에 의해 유지되는 전역(인터페이스별 구성 정보와 반대)을 검색합니다.
PGET_INTERFACE_INFO GetInterfaceInfo 함수는 특정 인터페이스에 대한 라우팅 프로토콜에 의해 유지되는 구성 정보를 가져옵니다.
PGET_MFE_STATUS 라우터 관리자는 GetMfeStatus 함수를 호출하여 지정된 인터페이스, 그룹 주소 및 원본 주소에 대한 MFE(멀티캐스트 전달 항목)의 상태를 가져옵니다.
PGET_NEIGHBORS 라우터 관리자는 GetNeighbors 함수를 호출하여 지정된 인터페이스를 통해 연결된 네트워크에 대한 쿼리를 가져옵니다.
PGET_NEXT_ORDERED_SERVICE GetNextOrderedService 함수는 테이블의 서비스 하위 집합에서 다음 서비스를 반환합니다. 반환된 서비스는 지정된 순서 지정 방법을 사용하여 지정된 입력 서비스 이후의 다음 서비스입니다.

PGET_SERVICE_COUNT GetServiceCount 함수는 테이블의 서비스 수를 반환합니다.
PINTERFACE_STATUS 라우터 관리자는 InterfaceStatus 함수를 호출하여 인터페이스의 상태를 변경합니다.
PIS_SERVICE IsService 함수는 지정된 형식 및 이름의 서비스가 서비스 테이블에 있는지 확인하고 필요에 따라 서비스의 매개 변수를 반환합니다.
PMGM_CREATION_ALERT_CALLBACK PMGM_CREATION_ALERT_CALLBACK 콜백은 라우팅 프로토콜에 대한 호출입니다. 이 호출은 새 원본의 멀티캐스트 패킷을 전달해야 하는 라우팅 프로토콜이 소유한 인터페이스의 하위 집합을 결정합니다.
PMGM_DISABLE_IGMP_CALLBACK PMGM_DISABLE_IGMP_CALLBACK 콜백은 라우팅 프로토콜이 IGMP를 사용하도록 설정된 인터페이스의 소유권을 취하거나 해제하고 있음을 알리기 위해 IGMP에 대한 호출입니다.
PMGM_ENABLE_IGMP_CALLBACK PMGM_ENABLE_IGMP_CALLBACK 콜백은 IGMP를 호출하여 라우팅 프로토콜이 인터페이스 소유권 가져오기 또는 해제를 완료했음을 알립니다.
PMGM_JOIN_ALERT_CALLBACK PMGM_JOIN_ALERT_CALLBACK 콜백은 다른 라우팅 프로토콜이 소유한 인터페이스에서 하나 이상의 그룹에 대해 새 수신기가 있다는 것을 프로토콜에 알리기 위해 라우팅 프로토콜을 호출하는 것입니다.
PMGM_LOCAL_JOIN_CALLBACK PMGM_LOCAL_JOIN_CALLBACK 콜백은 라우팅 프로토콜을 호출하여 IGMP가 현재 라우팅 프로토콜이 소유하고 있는 인터페이스에서 그룹에 대한 새 수신기를 감지했음을 프로토콜에 알립니다.
PMGM_LOCAL_LEAVE_CALLBACK PMGM_LOCAL_LEAVE_CALLBACK 콜백은 라우팅 프로토콜을 호출하여 IGMP가 라우팅 프로토콜이 현재 소유하고 있는 인터페이스의 그룹에 대한 수신기가 더 이상 없음을 감지했음을 라우팅 프로토콜에 알립니다.

PMGM_PRUNE_ALERT_CALLBACK PMGM_PRUNE_ALERT_CALLBACK 콜백은 다른 라우팅 프로토콜이 소유한 인터페이스에 수신기가 더 이상 존재하지 않는다는 것을 프로토콜에 알리기 위해 라우팅 프로토콜을 호출하는 것입니다.
PMGM_RPF_CALLBACK PMGM_RPF_CALLBACK 콜백은 지정된 패킷이 올바른 인터페이스에서 수신되었는지 확인하기 위해 라우팅 프로토콜을 호출하는 것입니다.
PMGM_WRONG_IF_CALLBACK PMGM_WRONG_IF_CALLBACK 지정된 소스와 잘못된 인터페이스의 지정된 그룹에 대해 패킷이 수신되었음을 프로토콜에 알리기 위해 라우팅 프로토콜을 호출하는 것입니다.
PMIB_CREATE MibCreate 함수는 SNMP MIB 스타일 만들기 요청을 라우팅 프로토콜에 전달합니다.
PMIB_DELETE MibDelete 함수는 SNMP MIB 스타일 삭제 요청을 라우팅 프로토콜에 전달합니다.
PMIB_GET MibGet 함수는 SNMP MIB 스타일 Get Request를 라우팅 프로토콜 DLL에 전달합니다.
PMIB_GET_FIRST MibGetFirst 함수는 SNMP MIB 스타일 Get First Request를 라우팅 프로토콜에 전달합니다.
PMIB_GET_NEXT MibGetNext 함수는 SNMP MIB 스타일 Get Next Request를 라우팅 프로토콜에 전달합니다.
PMIB_GET_TRAP_INFO MibGetTrapInfo 함수는 트랩에 대한 자세한 정보를 위해 트랩 이벤트를 설정하는 모듈을 쿼리합니다.
PMIB_SET MibSet 함수는 SNMP MIB 스타일 Set Request를 라우팅 프로토콜에 전달합니다.
PMIB_SET_TRAP_INFO MibSetTrapInfo 함수는 TRAP을 실행해야 할 때마다 신호를 수신하는 이벤트에 핸들을 전달합니다.

PQUERY_POWER QueryPower 함수는 나중에 사용하도록 예약되어 있습니다.
PREGISTER_PROTOCOL RegisterProtocol 함수는 라우터 관리자에 라우팅 프로토콜을 등록합니다. 또한 라우팅 프로토콜이 지원하는 기능을 라우터 관리자에게 알릴 수 있습니다.
PSET_GLOBAL_INFO SetGlobalInfo 함수는 라우팅 프로토콜에 의해 유지되는 전역(인터페이스별 구성 정보와 반대)을 설정합니다. 이 정보의 형식은 라우팅 프로토콜에 따라 다릅니다.
PSET_INTERFACE_INFO SetInterfaceInfo 함수는 라우팅 프로토콜에서 관리하는 특정 인터페이스의 구성을 설정합니다.
PSET_POWER SetPower 함수는 나중에 사용할 수 있습니다.
PSTART_COMPLETE 라우터 관리자는 StartComplete 함수를 호출하여 초기화가 완료되고 모든 인터페이스가 추가되었음을 라우팅 프로토콜에 알릴 수 있습니다. 라우팅 프로토콜은 프로토콜별 동작을 시작하기 전에 이 호출을 기다려야 합니다.
PSTART_PROTOCOL StartProtocol 함수는 라우팅 프로토콜의 기능을 초기화합니다.
PSTOP_PROTOCOL StopProtocol 함수를 사용하면 라우팅 프로토콜이 순서대로 종료됩니다.
RASADFUNCA RASADFunc 함수는 자동 진단용 사용자 지정 사용자 인터페이스를 제공하는 데 사용되는 애플리케이션 정의 콜백 함수입니다. (ANSI)
RASADFUNCW RASADFunc 함수는 자동 진단용 사용자 지정 사용자 인터페이스를 제공하는 데 사용되는 애플리케이션 정의 콜백 함수입니다. (유니코드)
RasClearConnectionStatistics RasClearConnectionStatistics 함수는 지정된 RAS 연결에 대해 누적된 모든 통계를 지웁니다.

RasClearLinkStatistics RasClearLinkStatistics 함수는 RAS 멀티링크 연결에서 지정된 링크에 대해 누적된 모든 통계를 지웁니다.
RasConnectionNotificationA RasConnectionNotification 함수는 RAS 연결을 만들거나 종료할 때 시스템에서 신호 상태로 설정하는 이벤트 개체를 지정합니다. (ANSI)
RasConnectionNotificationW RasConnectionNotification 함수는 RAS 연결을 만들거나 종료할 때 시스템에서 신호 상태로 설정하는 이벤트 개체를 지정합니다. (유니코드)
RasCreatePhonebookEntryA RasCreatePhonebookEntry 함수는 새 전화 번호부 항목을 만듭니다. 이 함수는 사용자가 전화 번호부 항목에 대한 정보를 입력하는 대화 상자를 표시합니다. (ANSI)
RasCreatePhonebookEntryW RasCreatePhonebookEntry 함수는 새 전화 번호부 항목을 만듭니다. 이 함수는 사용자가 전화 번호부 항목에 대한 정보를 입력하는 대화 상자를 표시합니다. (유니코드)
RasCustomDeleteEntryNotifyFn RasCustomDeleteEntryNotify 함수는 타사 사용자 지정 전화 걸기 DLL에서 내보내는 애플리케이션 정의 함수입니다. 이 기능을 사용하면 타사 공급업체가 전화 번호부 항목을 관리하기 위한 사용자 지정 대화 상자를 구현할 수 있습니다.
RasCustomDialDlgFn RasCustomDialDlg 함수는 타사 사용자 지정 전화 걸기 DLL에서 내보내는 애플리케이션 정의 함수입니다. 이 함수를 사용하면 타사 공급업체에서 사용자 지정 RAS 연결 대화 상자를 구현할 수 있습니다.
RasCustomDialFn RasCustomDial 함수는 타사 사용자 지정 전화 걸기 DLL에서 내보내는 애플리케이션 정의 함수입니다. 이 기능을 사용하면 타사 공급업체에서 사용자 지정 원격 액세스 전화 걸기 루틴을 구현할 수 있습니다.
RasCustomEntryDlgFn RasCustomEntryDlg 함수는 타사 사용자 지정 전화 걸기 DLL에서 내보내는 애플리케이션 정의 함수입니다. 이 기능을 사용하면 타사 공급업체가 전화 번호부 항목을 관리하기 위한 사용자 지정 대화 상자를 구현할 수 있습니다.

RasCustomHangUpFn RasCustomHangUp 함수는 타사 사용자 지정 전화 걸기 DLL에서 내보내는 애플리케이션 정의 함수입니다. 이 기능을 사용하면 타사 공급업체에서 사용자 지정 연결 중단 루틴을 구현할 수 있습니다.
RasCustomScriptExecuteFn RAS는 RASEO_CustomScript 옵션이 설정된 전화 번호부 항목에 대한 연결을 설정할 때 RasCustomScriptExecute 함수를 호출합니다.
RasDeleteEntryA RasDeleteEntry 함수는 전화 번호부에서 항목을 삭제합니다. (ANSI)
RasDeleteEntryW RasDeleteEntry 함수는 전화 번호부에서 항목을 삭제합니다. (유니코드)
RasDeleteSubEntryA RasDeleteSubEntry 함수는 지정된 전화 번호부 항목에서 지정된 하위 엔트리를 삭제합니다. (ANSI)
RasDeleteSubEntryW RasDeleteSubEntry 함수는 지정된 전화 번호부 항목에서 지정된 하위 엔트리를 삭제합니다. (유니코드)
RasDialA RasDial 함수는 RAS 클라이언트와 RAS 서버 간에 RAS 연결을 설정합니다. 연결 데이터에는 콜백 및 사용자 인증 정보가 포함됩니다. (ANSI)
RasDialDlgA RasDialDlg 함수는 지정된 전화 번호부 항목 및 로그온한 사용자의 자격 증명을 사용하여 RAS 연결을 설정합니다. 이 함수는 연결 작업의 상태를 나타내는 대화 상자의 스트림을 표시합니다. (ANSI)
RasDialDlgW RasDialDlg 함수는 지정된 전화 번호부 항목 및 로그온한 사용자의 자격 증명을 사용하여 RAS 연결을 설정합니다. 이 함수는 연결 작업의 상태를 나타내는 대화 상자의 스트림을 표시합니다. (유니코드)
RASDIALFUNC RasDialFunc 콜백 함수는 RAS 연결 프로세스 중에 상태 변경이 발생할 때 RasDial 함수에 의해 호출됩니다.

RASDIALFUNC1 RasDialFunc1 함수는 원격 액세스 연결 프로세스 중에 상태 변경이 발생할 때 RasDial 함수에 의해 호출됩니다.
RASDIALFUNC2 RasDialFunc2 콜백 함수는 원격 액세스 연결 프로세스 중에 상태 변경이 발생할 때 RasDial 함수 호출에 의해 호출됩니다.
RasDialW RasDial 함수는 RAS 클라이언트와 RAS 서버 간에 RAS 연결을 설정합니다. 연결 데이터에는 콜백 및 사용자 인증 정보가 포함됩니다. (유니코드)
RasEditPhonebookEntryA RasEditPhonebookEntry 함수는 기존 전화 번호부 항목을 편집합니다. 함수는 사용자가 기존 정보를 수정할 수 있는 대화 상자를 표시합니다. (ANSI)
RasEditPhonebookEntryW RasEditPhonebookEntry 함수는 기존 전화 번호부 항목을 편집합니다. 함수는 사용자가 기존 정보를 수정할 수 있는 대화 상자를 표시합니다. (유니코드)
RasEntryDlgA RasEntryDlg 함수는 사용자가 전화 번호부 항목을 조작할 수 있는 모달 속성 시트를 표시합니다. (ANSI)
RasEntryDlgW RasEntryDlg 함수는 사용자가 전화 번호부 항목을 조작할 수 있는 모달 속성 시트를 표시합니다. (유니코드)
RasEnumAutodialAddressesA RasEnumAutodialAddresses 함수는 AutoDial 매핑 데이터베이스의 모든 주소 목록을 반환합니다. (ANSI)
RasEnumAutodialAddressesW RasEnumAutodialAddresses 함수는 AutoDial 매핑 데이터베이스의 모든 주소 목록을 반환합니다. (유니코드)
RasEnumConnectionsA RasEnumConnections 함수는 모든 활성 RAS 연결을 나열합니다. 각 연결의 핸들과 전화 번호부 항목 이름을 반환합니다. (ANSI)

RasEnumConnectionsW RasEnumConnections 함수는 모든 활성 RAS 연결을 나열합니다. 각 연결의 핸들과 전화 번호부 항목 이름을 반환합니다. (유니코드)
RasEnumDevicesA RasEnumDevices 함수는 사용 가능한 모든 RAS 지원 디바이스의 이름과 유형을 반환합니다. (ANSI)
RasEnumDevicesW RasEnumDevices 함수는 사용 가능한 모든 RAS 지원 디바이스의 이름과 유형을 반환합니다. (유니코드)
RasEnumEntriesA RasEnumEntries 함수는 원격 액세스 전화 번호부의 모든 항목 이름을 나열합니다. (ANSI)
RasEnumEntriesW RasEnumEntries 함수는 원격 액세스 전화 번호부의 모든 항목 이름을 나열합니다. (유니코드)
RasFreeEapUserIdentityA RasFreeEapUserIdentity 함수를 사용하여 RasGetEapUserIdentity에서 반환된 메모리 버퍼를 해제합니다. (ANSI)
RasFreeEapUserIdentityW RasFreeEapUserIdentity 함수를 사용하여 RasGetEapUserIdentity에서 반환된 메모리 버퍼를 해제합니다. (유니코드)
RasGetAutodialAddressA RasGetAutodialAddress 함수는 AutoDial 매핑 데이터베이스의 네트워크 주소와 연결된 모든 AutoDial 항목에 대한 정보를 검색합니다. (ANSI)
RasGetAutodialAddressW RasGetAutodialAddress 함수는 AutoDial 매핑 데이터베이스의 네트워크 주소와 연결된 모든 AutoDial 항목에 대한 정보를 검색합니다. (유니코드)
RasGetAutodialEnableA RasGetAutodialEnable 함수는 지정된 TAPI 전화 걸기 위치에 대해 AutoDial 기능을 사용할 수 있는지 여부를 나타냅니다. (ANSI)

RasGetAutodialEnableW RasGetAutodialEnable 함수는 지정된 TAPI 전화 걸기 위치에 대해 AutoDial 기능을 사용할 수 있는지 여부를 나타냅니다. (유니코드)
RasGetAutodialParamA RasGetAutodialParam 함수는 AutoDial 매개 변수의 값을 검색합니다. (ANSI)
RasGetAutodialParamW RasGetAutodialParam 함수는 AutoDial 매개 변수의 값을 검색합니다. (유니코드)
RasGetConnectionStatistics RasGetConnectionStatistics 함수는 지정된 연결에 대한 누적 연결 통계를 검색합니다.
RasGetConnectStatusA RasGetConnectStatus 함수는 지정된 원격 액세스 연결의 현재 상태에 대한 정보를 검색합니다. 애플리케이션은 이 호출을 사용하여 비동기 RasDial 호출이 완료되는 시기를 결정할 수 있습니다. (ANSI)
RasGetConnectStatusW RasGetConnectStatus 함수는 지정된 원격 액세스 연결의 현재 상태에 대한 정보를 검색합니다. 애플리케이션은 이 호출을 사용하여 비동기 RasDial 호출이 완료되는 시기를 결정할 수 있습니다. (유니코드)
RasGetCountryInfoA RasGetCountryInfo 함수는 국가/지역의 Windows 전화 통신 목록에서 국가/지역별 전화 걸기 정보를 검색합니다. (ANSI)
RasGetCountryInfoW RasGetCountryInfo 함수는 국가/지역의 Windows 전화 통신 목록에서 국가/지역별 전화 걸기 정보를 검색합니다. (유니코드)
RasGetCredentialsA RasGetCredentials 함수는 지정된 RAS 전화 번호부 항목과 연결된 사용자 자격 증명을 검색합니다. (ANSI)
RasGetCredentialsW RasGetCredentials 함수는 지정된 RAS 전화 번호부 항목과 연결된 사용자 자격 증명을 검색합니다. (유니코드)

RasGetCustomAuthDataA RasGetCustomAuthData 함수를 사용하여 연결 관련 인증 정보를 검색합니다. 이 정보는 특정 사용자에 한정되지 않습니다. (ANSI)
RasGetCustomAuthDataW RasGetCustomAuthData 함수를 사용하여 연결 관련 인증 정보를 검색합니다. 이 정보는 특정 사용자에 한정되지 않습니다. (유니코드)
RasGetEapUserDataA RasGetEapUserData 함수를 사용하여 지정된 전화 번호부 항목에 대한 사용자별 EAP(확장 가능 인증 프로토콜) 정보를 검색합니다. (ANSI)
RasGetEapUserDataW RasGetEapUserData 함수를 사용하여 지정된 전화 번호부 항목에 대한 사용자별 EAP(확장 가능 인증 프로토콜) 정보를 검색합니다. (유니코드)
RasGetEapUserIdentityA RasGetEapUserIdentity 함수는 현재 사용자의 ID 정보를 검색합니다. 이 정보를 사용하여 EAP(Extensible Authentication Protocol)가 필요한 전화 번호부 항목으로 RasDial을 호출합니다. (ANSI)
RasGetEapUserIdentityW RasGetEapUserIdentity 함수는 현재 사용자의 ID 정보를 검색합니다. 이 정보를 사용하여 EAP(Extensible Authentication Protocol)가 필요한 전화 번호부 항목으로 RasDial을 호출합니다. (유니코드)
RasGetEntryDialParamsA RasGetEntryDialParams 함수는 지정된 전화 번호부 항목에 대해 RasDial 또는 RasSetEntryDialParams 함수에 마지막으로 성공한 호출로 저장된 연결 정보를 검색합니다. (ANSI)
RasGetEntryDialParamsW RasGetEntryDialParams 함수는 지정된 전화 번호부 항목에 대해 RasDial 또는 RasSetEntryDialParams 함수에 마지막으로 성공한 호출로 저장된 연결 정보를 검색합니다. (유니코드)
RasGetEntryPropertiesA RasGetEntryProperties 함수는 전화 번호부 항목의 속성을 검색합니다. (ANSI)

RasGetEntryPropertiesW RasGetEntryProperties 함수는 전화 번호부 항목의 속성을 검색합니다. (유니코드)
RasGetErrorStringA RasGetErrorString 함수는 지정된 RAS 오류 값에 대한 오류 메시지 문자열을 가져옵니다. (ANSI)
RasGetErrorStringW RasGetErrorString 함수는 지정된 RAS 오류 값에 대한 오류 메시지 문자열을 가져옵니다. (유니코드)
RasGetLinkStatistics RasGetLinkStatistics 함수는 RAS 다중 링크 연결에서 지정된 링크에 대한 누적 통계를 검색합니다.
RasGetNapStatus 지정된 원격 액세스 연결에 대한 NAP(네트워크 액세스 보호) 연결 상태 변수를 검색합니다.
RasGetProjectionInfoEx 로컬 클라이언트의 모든 RAS 연결에 대한 PPP(지점 간 프로토콜) 또는 IKEv2(인터넷 키 교환 버전 2) 원격 액세스 프로젝션 작업에 대한 정보를 가져옵니다.
RasGetSubEntryHandleA RasGetSubEntryHandle 함수는 멀티링크 연결의 지정된 하위 엔트리에 대한 연결 핸들을 검색합니다. (ANSI)
RasGetSubEntryHandleW RasGetSubEntryHandle 함수는 멀티링크 연결의 지정된 하위 엔트리에 대한 연결 핸들을 검색합니다. (유니코드)
RasGetSubEntryPropertiesA RasGetSubEntryProperties 함수는 지정된 전화 번호부 항목의 하위 항목에 대한 정보를 검색합니다. (ANSI)
RasGetSubEntryPropertiesW RasGetSubEntryProperties 함수는 지정된 전화 번호부 항목의 하위 항목에 대한 정보를 검색합니다. (유니코드)

RasHangUpA RasHangUp 함수는 원격 액세스 연결을 종료합니다. 연결은 RAS 연결 핸들로 지정됩니다. 함수는 핸들과 연결된 모든 RASAPI32.DLL 리소스를 해제합니다. (ANSI)
RasHangUpW RasHangUp 함수는 원격 액세스 연결을 종료합니다. 연결은 RAS 연결 핸들로 지정됩니다. 함수는 핸들과 연결된 모든 RASAPI32.DLL 리소스를 해제합니다. (유니코드)
RasInvokeEapUI RasInvokeEapUI 함수는 사용자로부터 EAP(Extensible Authentication Protocol) 정보를 가져오는 사용자 지정 사용자 인터페이스를 표시합니다.
RASPBDLGFUNCA RasPBDlgFunc 함수는 RasPhonebookDlg 대화 상자가 열려 있는 동안 사용자 활동에 대한 알림을 받는 애플리케이션 정의 콜백 함수입니다. (ANSI)
RASPBDLGFUNCW RasPBDlgFunc 함수는 RasPhonebookDlg 대화 상자가 열려 있는 동안 사용자 활동에 대한 알림을 받는 애플리케이션 정의 콜백 함수입니다. (유니코드)
RasPhonebookDlgA RasPhonebookDlg 함수는 기본 전화 접속 네트워킹 대화 상자를 표시합니다. (ANSI)
RasPhonebookDlgW RasPhonebookDlg 함수는 기본 전화 접속 네트워킹 대화 상자를 표시합니다. (유니코드)
RasRenameEntryA RasRenameEntry 함수는 전화 번호부에 있는 항목의 이름을 변경합니다. (ANSI)
RasRenameEntryW RasRenameEntry 함수는 전화 번호부에 있는 항목의 이름을 변경합니다. (유니코드)
RasSecurityDialogBegin RasSecurityDialogBegin 함수는 원격 사용자가 연결을 시도할 때 RAS 서버가 호출하는 타사 RAS 보안 DLL 진입점입니다. 이렇게 하면 보안 DLL이 원격 사용자 인증을 시작할 수 있습니다.

RasSecurityDialogComplete RasSecurityDialogComplete 함수는 타사 보안 인증 트랜잭션의 결과를 RAS 서버에 알릴 수 있습니다.
RasSecurityDialogEnd RasSecurityDialogEnd 함수는 RAS 서버가 인증 트랜잭션을 종료하기 위해 호출하는 타사 RAS 보안 DLL 진입점입니다.
RasSecurityDialogGetInfo RasSecurityDialogGetInfo 함수는 RAS 보안 DLL에 의해 호출되어 RAS 서버에서 포트에 대한 정보를 가져옵니다.
RasSecurityDialogReceive RasSecurityDialogReceive 함수는 보안 문제에 대한 원격 사용자의 응답을 받는 비동기 작업을 시작합니다.
RasSecurityDialogSend RasSecurityDialogSend 함수는 원격 컴퓨터의 터미널 창에 표시할 메시지를 보냅니다. 타사 RAS 보안 DLL은 원격 사용자 인증의 일부로 이 메시지를 보냅니다.
RasSetAutodialAddressA RasSetAutodialAddress 함수는 AutoDial 매핑 데이터베이스에 주소를 추가할 수 있습니다. 또는 함수는 데이터베이스의 기존 주소와 연결된 데이터를 삭제하거나 수정할 수 있습니다. (ANSI)
RasSetAutodialAddressW RasSetAutodialAddress 함수는 AutoDial 매핑 데이터베이스에 주소를 추가할 수 있습니다. 또는 함수는 데이터베이스의 기존 주소와 연결된 데이터를 삭제하거나 수정할 수 있습니다. (유니코드)
RasSetAutodialEnableA RasSetAutodialEnable 함수는 지정된 TAPI 전화 걸기 위치에 대해 AutoDial 기능을 사용하거나 사용하지 않도록 설정합니다. (ANSI)
RasSetAutodialEnableW RasSetAutodialEnable 함수는 지정된 TAPI 전화 걸기 위치에 대해 AutoDial 기능을 사용하거나 사용하지 않도록 설정합니다. (유니코드)
RasSetAutodialParamA RasSetAutodialParam 함수는 AutoDial 매개 변수의 값을 설정합니다. (ANSI)

RasSetAutodialParamW RasSetAutodialParam 함수는 AutoDial 매개 변수의 값을 설정합니다. (유니코드)
RasSetCredentialsA 지정된 RAS 전화 번호부 항목과 연결된 사용자 자격 증명을 설정합니다. (ANSI)
RasSetCredentialsW 지정된 RAS 전화 번호부 항목과 연결된 사용자 자격 증명을 설정합니다. (유니코드)
RasSetCustomAuthDataA RasSetCustomAuthData 함수를 사용하여 연결 관련 인증 정보를 설정합니다. 이 정보는 특정 사용자에 한정되어서는 안 됩니다. (ANSI)
RasSetCustomAuthDataW RasSetCustomAuthData 함수를 사용하여 연결 관련 인증 정보를 설정합니다. 이 정보는 특정 사용자에 한정되어서는 안 됩니다. (유니코드)
RasSetEapUserDataA RasSetEapUserData 함수를 사용하여 레지스트리에 지정된 전화 번호부 항목에 대한 사용자별 EAP(Extensible Authentication Protocol) 정보를 저장합니다. (ANSI)
RasSetEapUserDataW RasSetEapUserData 함수를 사용하여 레지스트리에 지정된 전화 번호부 항목에 대한 사용자별 EAP(Extensible Authentication Protocol) 정보를 저장합니다. (유니코드)
RasSetEntryDialParamsA RasSetEntryDialParams 함수는 지정된 전화 번호부 항목에 대해 RasDial 또는 RasSetEntryDialParams 함수에 마지막으로 성공한 호출로 저장된 연결 정보를 변경합니다. (ANSI)
RasSetEntryDialParamsW RasSetEntryDialParams 함수는 지정된 전화 번호부 항목에 대해 RasDial 또는 RasSetEntryDialParams 함수에 마지막으로 성공한 호출로 저장된 연결 정보를 변경합니다. (유니코드)
RasSetEntryPropertiesA RasSetEntryProperties 함수는 전화 번호부의 항목에 대한 연결 정보를 변경하거나 새 전화 번호부 항목을 만듭니다. (ANSI)

RasSetEntryPropertiesW RasSetEntryProperties 함수는 전화 번호부의 항목에 대한 연결 정보를 변경하거나 새 전화 번호부 항목을 만듭니다. (유니코드)
RasSetSubEntryPropertiesA RasSetSubEntryProperties 함수는 새 하위 엔트리를 만들거나 지정된 전화 번호부 항목의 기존 하위 엔트리를 수정합니다. (ANSI)
RasSetSubEntryPropertiesW RasSetSubEntryProperties 함수는 새 하위 엔트리를 만들거나 지정된 전화 번호부 항목의 기존 하위 엔트리를 수정합니다. (유니코드)
RasUpdateConnection RasUpdateConnection 함수는 IKEv2(인터넷 키 교환 버전 2) 연결의 터널 엔드포인트를 업데이트합니다.
RasValidateEntryNameA RasValidateEntryName 함수는 연결 항목 이름의 형식의 유효성을 검사합니다. 이름에는 공백이 아닌 영숫자가 하나 이상 포함되어야 합니다. (ANSI)
RasValidateEntryNameW RasValidateEntryName 함수는 연결 항목 이름의 형식의 유효성을 검사합니다. 이름에는 공백이 아닌 영숫자가 하나 이상 포함되어야 합니다. (유니코드)
RTM_IPV4_GET_ADDR_AND_LEN RTM_IPV4_GET_ADDR_AND_LEN 매크로는 제네릭 net 주소와 길이를 IPv4 RTM_NET_ADDRESS 구조체 및 길이로 변환합니다.
RTM_IPV4_GET_ADDR_AND_MASK RTM_IPV4_GET_ADDR_AND_MASK 매크로는 제네릭 RTM_NET_ADDRESS 구조와 길이를 IPv4 주소 및 마스크로 변환합니다.
RTM_IPV4_LEN_FROM_MASK RTM_IPV4_LEN_FROM_MASK 매크로는 IPv4 마스크를 제네릭 경로 길이로 변환합니다.
RTM_IPV4_MAKE_NET_ADDRESS RTM_IPV4_MAKE_NET_ADDRESS 매크로는 IPv4 주소와 길이를 제네릭 RTM_NET_ADDRESS 구조체로 변환합니다.

RTM_IPV4_MASK_FROM_LEN RTM_IPV4_MASK_FROM_LEN 매크로는 제네릭 경로 길이를 IPv4 마스크로 변환합니다.
RTM_IPV4_SET_ADDR_AND_LEN RTM_IPV4_SET_ADDR_AND_LEN 매크로는 IPv4 주소와 길이를 제네릭 RTM_NET_ADDRESS 구조체로 변환합니다.
RTM_IPV4_SET_ADDR_AND_MASK RTM_IPV4_SET_ADDR_AND_MASK 매크로는 IPv4 주소와 마스크를 제네릭 RTM_NET_ADDRESS 구조체로 변환합니다.
RTM_SIZE_OF_DEST_INFO RTM_SIZE_OF_DEST_INFO 매크로는 대상 정보 구조(RTM_DEST_INFO)의 크기를 반환합니다.
RTM_SIZE_OF_ROUTE_INFO RTM_SIZE_OF_ROUTE_INFO 매크로는 RTM_ROUTE_INFO 경로 정보 구조의 크기를 반환합니다.
RtmAddNextHop RtmAddNextHop 함수는 새 다음 홉 항목을 추가하거나 기존 다음 홉 항목을 클라이언트의 다음 홉 목록에 업데이트합니다.
RtmAddRouteToDest RtmAddRouteToDest 함수는 라우팅 테이블에 새 경로를 추가하거나 라우팅 테이블의 기존 경로를 업데이트합니다. 최상의 경로가 변경되면 변경 알림이 생성됩니다.
RtmBlockMethods RtmBlockMethods 함수는 지정된 대상, 경로 또는 다음 홉 또는 모든 대상, 경로 및 다음 홉에 대한 메서드 실행을 차단하거나 차단 해제합니다.
RtmCreateDestEnum RtmCreateDestEnum 함수는 라우팅 테이블에서 대상의 열거를 시작합니다. 클라이언트는 하나 이상의 뷰 또는 모든 보기에 대한 대상을 열거할 수 있습니다.
RtmCreateNextHopEnum RtmCreateNextHopEnum은 다음 홉 목록에서 다음 홉을 열거합니다.

RtmCreateRouteEnum RtmCreateRouteEnum 함수는 라우팅 테이블의 특정 대상 또는 대상 범위에 대한 경로의 열거형을 만듭니다. 클라이언트는 하나 이상의 뷰 또는 모든 보기에 대한 경로를 열거할 수 있습니다.
RtmCreateRouteList RtmCreateRouteList 함수는 호출자가 소유한 경로의 복사본을 유지할 수 있는 목록을 만듭니다.
RtmCreateRouteListEnum RtmCreateRouteListEnum 함수는 지정된 경로 목록에 경로의 열거형을 만듭니다.
RtmDeleteEnumHandle RtmDeleteEnumHandle 함수는 지정된 열거형 핸들을 삭제하고 열거형에 할당된 모든 리소스를 해제합니다.
RtmDeleteNextHop RtmDeleteNextHop 함수는 다음 홉 목록에서 다음 홉을 삭제합니다.
RtmDeleteRouteList RtmDeleteRouteList 함수는 클라이언트별 경로 목록에서 모든 경로를 제거한 다음 목록에 할당된 모든 리소스를 해제합니다.
RtmDeleteRouteToDest RtmDeleteRouteToDest 함수는 라우팅 테이블에서 경로를 삭제하고 최상의 경로가 변경된 경우 해당 대상에 대한 최상의 경로 정보를 업데이트합니다. 최상의 경로가 변경되면 변경 알림이 생성됩니다.
RtmDeregisterEntity RtmDeregisterEntity 함수는 라우팅 테이블 관리자 인스턴스 및 주소 패밀리에서 클라이언트의 등록을 취소합니다.
RtmDeregisterFromChangeNotification RtmDeregisterFromChangeNotification 함수는 변경 알림에서 클라이언트의 등록을 취소하고 알림에 할당된 모든 리소스를 해제합니다.
RtmFindNextHop RtmFindNextHop 함수는 클라이언트의 다음 홉 목록에서 특정 다음 홉을 찾습니다.

RtmGetChangedDests RtmGetChangedDests 함수는 변경된 정보가 있는 대상 집합을 반환합니다.
RtmGetChangeStatus RtmGetChangeStatus 함수는 RtmGetChangedDests를 사용하여 검색되지 않은 보류 중인 변경 내용이 있는지 확인합니다.
RtmGetDestInfo RtmGetDestInfo 함수는 대상에 대한 정보를 반환합니다.
RtmGetEntityInfo RtmGetEntityInfo 함수는 이전에 등록된 클라이언트에 대한 정보를 반환합니다.
RtmGetEntityMethods RtmGetEntityMethods 함수는 지정된 클라이언트를 쿼리하여 다른 클라이언트가 호출할 수 있는 메서드를 결정합니다.
RtmGetEnumDests RtmGetEnumDests 함수는 지정된 열거형의 다음 대상 집합을 검색합니다.
RtmGetEnumNextHops RtmGetEnumNextHops 함수는 지정된 열거형에서 다음 홉 집합을 검색합니다.
RtmGetEnumRoutes RtmGetEnumRoutes 함수는 지정된 열거형의 다음 경로 집합을 검색합니다.
RtmGetExactMatchDestination RtmGetExactMatchDestination 함수는 라우팅 테이블에서 지정된 네트워크 주소 및 서브넷 마스크와 정확히 일치하는 대상을 검색합니다. 정확히 일치하는 항목이 발견되면 해당 대상에 대한 정보가 반환됩니다.
RtmGetExactMatchRoute RtmGetExactMatchRoute 함수는 라우팅 테이블에서 지정된 경로와 정확히 일치하는 경로를 검색합니다.
RtmGetLessSpecificDestination RtmGetLessSpecificDestination 함수는 대상 접두사로 지정된 다음 최적 일치(가장 긴) 접두사로 대상에 대한 라우팅 테이블을 검색합니다. 요청된 대상 정보가 반환됩니다.

RtmGetListEnumRoutes RtmGetListEnumRoutes 함수는 지정된 경로 목록의 경로 집합을 열거합니다.
RtmGetMostSpecificDestination RtmGetMostSpecificDestination 함수는 라우팅 테이블에서 지정된 네트워크 주소 및 서브넷 마스크와 정확히 일치하는 대상을 검색합니다. 정확한 일치 항목을 찾을 수 없으면 최상의 접두사는 일치합니다.
RtmGetNextHopInfo RtmGetNextHopInfo 함수는 지정된 다음 홉에 대한 정보를 반환합니다.
RtmGetNextHopPointer RtmGetNextHopPointer 함수는 지정된 다음 홉에 대한 직접 포인터를 가져옵니다. 포인터를 사용하면 다음 홉 소유자가 라우팅 테이블 관리자의 RTM_NEXTHOP_INFO 구조에 직접 읽기 권한을 부여할 수 있습니다.
RtmGetOpaqueInformationPointer RtmGetOpaqueInformationPointer 함수는 이 클라이언트용으로 예약된 대상의 불투명 정보 필드에 대한 포인터를 반환합니다.
RtmGetRegisteredEntities RtmGetRegisteredEntities 함수는 라우팅 테이블 관리자의 지정된 인스턴스 및 지정된 주소 패밀리에 등록된 모든 클라이언트에 대한 정보를 반환합니다.
RtmGetRouteInfo RtmGetRouteInfo 함수는 지정된 경로에 대한 정보를 반환합니다.
RtmGetRoutePointer RtmGetRoutePointer 함수는 경로 소유자가 읽기 액세스를 허용하는 경로에 대한 직접 포인터를 가져옵니다.
RtmHoldDestination RtmHoldDestination 함수는 일정 시간 동안 보류 상태로 배치할 대상을 표시합니다. 보류는 모든 보기에서 대상의 마지막 경로가 삭제된 경우에만 발생합니다.
RtmIgnoreChangedDests RtmIgnoreChangedDests 함수는 이미 발생한 경우 각 대상에 대한 다음 변경 사항을 건너뜁니다.

RtmInsertInRouteList RtmInsertInRouteList 함수는 지정된 경로 집합을 클라이언트의 경로 목록에 삽입합니다. 경로가 이미 다른 목록에 있는 경우 경로가 이전 목록에서 제거되고 새 목록에 삽입됩니다.
RtmInvokeMethod RtmInvokeMethod 함수는 다른 클라이언트에서 내보낸 메서드를 호출합니다.
RtmIsBestRoute RtmIsBestRoute 함수는 지정된 경로가 대상에 가장 적합한 경로인 뷰 집합을 반환합니다.
RtmIsMarkedForChangeNotification RtmIsMarkedForChangeNotification 함수는 라우팅 테이블 관리자를 쿼리하여 대상이 이전에 RtmMarkDestForChangeNotification 호출로 표시되었는지 확인합니다.
RtmLockDestination RtmLockDestination 함수는 라우팅 테이블에서 대상을 잠그거나 잠금 해제합니다. 불투명 포인터를 변경하는 동안 대상을 보호하려면 이 함수를 사용합니다.
RtmLockNextHop RtmLockNextHop 함수는 다음 홉을 잠그거나 잠금 해제합니다. 다음 홉을 변경하기 전에 다음 홉의 소유자가 다음 홉을 잠그도록 이 함수를 호출해야 합니다. 다음 홉에 대한 포인터가 반환됩니다.
RtmLockRoute RtmLockRoute 함수는 라우팅 테이블에서 경로를 잠그거나 잠금을 해제합니다. 이렇게 하면 클라이언트가 클라이언트가 소유한 불투명 경로 포인터에 필요한 변경을 수행하는 동안 경로가 보호됩니다.
RtmMarkDestForChangeNotification RtmMarkDestForChangeNotification 함수는 클라이언트의 대상을 표시합니다.
RtmReferenceHandles RtmReferenceHandles 함수는 라우팅 관리자가 해당 개체에 액세스하는 데 사용한 하나 이상의 핸들이 가리키는 개체에 대한 참조 횟수를 증가합니다.
RtmRegisterEntity RtmRegisterEntity 함수는 특정 주소 패밀리에 대한 라우팅 테이블 관리자 인스턴스에 클라이언트를 등록합니다.

RtmRegisterForChangeNotification RtmRegisterForChangeNotification 함수는 라우팅 테이블 관리자에게 클라이언트가 지정된 유형의 변경 내용에 대한 변경 알림을 받아야 한다고 알려줍니다.
RtmReleaseChangedDests RtmReleaseChangedDests 함수는 변경된 대상 핸들을 해제합니다.
RtmReleaseDestInfo RtmReleaseDestInfo 함수는 대상 구조를 해제합니다.
RtmReleaseDests RtmReleaseDests 함수는 대상 핸들을 해제합니다.
RtmReleaseEntities RtmReleaseEntities 함수는 RtmGetRegisteredEntities에서 반환된 클라이언트 핸들을 해제합니다.
RtmReleaseEntityInfo RtmReleaseEntityInfo 함수는 클라이언트 구조를 해제합니다.
RtmReleaseNextHopInfo RtmReleaseNextHopInfo 함수는 다음 홉 구조를 해제합니다.
RtmReleaseNextHops RtmReleaseNextHops 함수는 다음 홉 핸들을 해제합니다.
RtmReleaseRouteInfo RtmReleaseRouteInfo 함수는 경로 구조를 해제합니다.
RtmReleaseRoutes RtmReleaseRoutes 함수는 경로 핸들을 해제합니다.
RtmUpdateAndUnlockRoute RtmUpdateAndUnlockRoute 함수는 대상에 대한 경로 집합에서 경로의 위치를 업데이트하고 대상에 대한 최상의 경로 정보를 조정합니다.

구조체

AUTH_VALIDATION_EX SSTP(Secure Socket Tunneling Protocol) 연결 설정 중에 클라이언트가 PPP(지점 간) 인증을 바이패스하도록 설정하는 데 사용됩니다.
IKEV2_CONFIG_PARAMS IKEv2(Internet Key Exchange 버전 2) 디바이스(RFC 4306)에 대한 매개 변수를 얻거나 설정하는 데 사용됩니다.
IKEV2_PROJECTION_INFO IKE(인터넷 키 교환) 협상 중에 얻은 정보를 포함합니다. (IKEV2_PROJECTION_INFO)
IKEV2_PROJECTION_INFO2 IKE(인터넷 키 교환) 협상 중에 얻은 정보를 포함합니다. (IKEV2_PROJECTION_INFO2)
IKEV2_TUNNEL_CONFIG_PARAMS1 IKEv2(Internet Key Exchange 버전 2) 디바이스에 대한 터널 매개 변수를 얻거나 설정하는 데 사용됩니다.
IKEV2_TUNNEL_CONFIG_PARAMS2 IKEv2(Internet Key Exchange 버전 2) 디바이스에 대한 터널 매개 변수를 가져오거나 설정합니다.
IP_ADAPTER_BINDING_INFO IP_ADAPTER_BINDING_INFO 구조에는 특정 네트워크 어댑터에 대한 IP 관련 정보가 포함됩니다.
IP_LOCAL_BINDING IP_LOCAL_BINDING 구조에는 어댑터에 대한 IP 주소 정보가 포함됩니다.
IPV6_ADAPTER_BINDING_INFO IPV6_ADAPTER_BINDING_INFO 구조에는 특정 네트워크 어댑터에 대한 IPv6 관련 정보가 포함되어 있습니다.
IPV6_LOCAL_BINDING IPV6_LOCAL_BINDING 구조에는 어댑터에 대한 IPv6 주소 정보가 포함되어 있습니다.
IPX_IF_INFO IPX_IF_INFO 구조체는 IPX 인터페이스에 대한 정보를 저장합니다.

IPX_SERVER_ENTRY IPX_SERVER_ENTRY 구조는 특정 IPX 서비스를 설명합니다.
IPX_SERVICE IPX_SERVICE 구조는 IPX 서비스에 대한 정보를 포함하고 이 정보를 얻은 인터페이스 및 프로토콜을 식별합니다.
IPXWAN_IF_INFO IPXWAN_IF_INFO 구조체는 IPX WAN 인터페이스에 대한 관리 상태를 저장합니다.
L2TP_CONFIG_PARAMS0 RAS 서버에서 L2TP(계층 2 터널링 프로토콜)에 대한 디바이스 구성을 가져와서 설정하는 데 사용됩니다. (L2TP_CONFIG_PARAMS0)
L2TP_CONFIG_PARAMS1 RAS 서버에서 L2TP(계층 2 터널링 프로토콜)에 대한 디바이스 구성을 가져와서 설정하는 데 사용됩니다. (L2TP_CONFIG_PARAMS1)
메시지 MESSAGE 공용 구조체에는 라우팅 프로토콜의 메시지 큐를 통해 라우터 관리자에게 보고된 이벤트에 대한 정보가 포함됩니다.
MGM_IF_ENTRY MGM_IF_ENTRY 구조는 라우터 인터페이스를 설명합니다.
MPR_CREDENTIALSEX_0 MPR_CREDENTIALSEX_0 구조에는 EAP(Extensible Authentication Protocols)에서 사용하는 정보와 같은 확장 자격 증명 정보가 포함됩니다.
MPR_CREDENTIALSEX_1 MPR_CREDENTIALSEX_1 구조에는 수요 다이얼 인터페이스에서 사용하는 미리 공유된 키가 포함되어 있습니다.
MPR_DEVICE_0 MPR_DEVICE_0 구조는 다중 링크 수요 다이얼 인터페이스의 링크에 사용되는 디바이스에 대한 정보를 저장합니다.

MPR_DEVICE_1 MPR_DEVICE_1 구조는 링크에 사용되는 디바이스에 대한 정보를 다중 링크된 수요 다이얼 인터페이스에 저장합니다. MPR_DEVICE_0 정보 외에도 MPR_DEVICE_1 전화 번호 정보를 포함합니다.
MPR_FILTER_0 정적 필터 구성 정보를 포함합니다.
MPR_IF_CUSTOMINFOEX0 수요 다이얼 인터페이스에 대한 터널별 사용자 지정 구성을 가져오거나 설정합니다.
MPR_IFTRANSPORT_0 MPR_IFTRANSPORT_0 구조에는 특정 인터페이스 전송에 대한 정보가 포함됩니다.
MPR_INTERFACE_0 MPR_INTERFACE_0 구조에는 특정 라우터 인터페이스에 대한 정보가 포함됩니다.
MPR_INTERFACE_1 MPR_INTERFACE_1 구조에는 특정 라우터 인터페이스에 대한 구성 및 상태 정보가 포함됩니다.
MPR_INTERFACE_2 라우터 수요 다이얼 인터페이스에 대한 데이터를 포함합니다. (MPR_INTERFACE_2)
MPR_INTERFACE_3 라우터 수요 다이얼 인터페이스에 대한 데이터를 포함합니다. (MPR_INTERFACE_3)
MPR_SERVER_0 MPR_SERVER_0 구조는 디바이스에 대한 정보를 검색하는 데 사용됩니다.
MPR_SERVER_1 디바이스에서 PPTP(지점 간 터널링 프로토콜) 및 계층 2 터널링 프로토콜(L2TP)에 사용할 수 있는 포트 수를 검색하고 설정하는 데 사용됩니다.
MPR_SERVER_2 디바이스에서 PPTP(지점 간 터널링 프로토콜), L2TP(계층 2 터널링 프로토콜) 및 SSTP(Secure Socket Tunneling Protocol)에 사용할 수 있는 포트 수를 검색하고 설정하는 데 사용됩니다.

MPR_SERVER_EX0 RAS 서버의 구성을 가져오기 또는 설정하는 데 사용됩니다. (MPR_SERVER_EX0)
MPR_SERVER_EX1 RAS 서버의 구성을 가져오기 또는 설정하는 데 사용됩니다. (MPR_SERVER_EX1)
MPR_SERVER_SET_CONFIG_EX0 RAS 서버의 터널 구성 정보를 얻거나 설정하는 데 사용됩니다. (MPR_SERVER_SET_CONFIG_EX0)
MPR_SERVER_SET_CONFIG_EX1 RAS 서버의 터널 구성 정보를 얻거나 설정하는 데 사용됩니다. (MPR_SERVER_SET_CONFIG_EX1)
MPR_TRANSPORT_0 MPR_TRANSPORT_0 구조에는 특정 전송에 대한 정보가 포함됩니다.
MPR40_SERVICE_CHARACTERISTICS MPR40_SERVICE_CHARACTERISTICS(stm.h) 구조체에는 라우터 관리자에 라우팅 프로토콜을 등록하는 데 사용되는 정보가 포함되어 있습니다.
MPR50_ROUTING_CHARACTERISTICS MPR_ROUTING_CHARACTERISTICS 구조에는 라우터 관리자에 라우팅 프로토콜을 등록하는 데 사용되는 정보가 포함되어 있습니다. (MPR50_ROUTING_CHARACTERISTICS)
MPR50_SERVICE_CHARACTERISTICS MPR50_SERVICE_CHARACTERISTICS(stm.h) 구조체에는 라우터 관리자에 라우팅 프로토콜을 등록하는 데 사용되는 정보가 포함되어 있습니다.
MPR60_ROUTING_CHARACTERISTICS MPR_ROUTING_CHARACTERISTICS 구조에는 라우터 관리자에 라우팅 프로토콜을 등록하는 데 사용되는 정보가 포함되어 있습니다. M
MPRAPI_ADMIN_DLL_CALLBACKS MPRAPI_ADMIN_DLL_CALLBACKS 구조는 MprAdminInitializeDllEx 함수에서 콜백 함수 포인터를 등록하는 데 사용됩니다.

MPRAPI_OBJECT_HEADER mprAdminConnectionEnumEx 메서드에서 사용하는 RAS_CONNECTION_EX, MPR_SERVER_EX, MPR_SERVER_SET_CONFIG_EX, RAS_UPDATE_CONNECTION, AUTH_VALIDATION_EX 구조체 및 구조체 버전에 대한 구조체 버전을 정의합니다.
MPRAPI_TUNNEL_CONFIG_PARAMS0 RAS 서버에서 터널 매개 변수의 구성을 얻거나 설정하는 데 사용됩니다. (MPRAPI_TUNNEL_CONFIG_PARAMS0)
MPRAPI_TUNNEL_CONFIG_PARAMS1 RAS 서버에서 터널 매개 변수의 구성을 얻거나 설정하는 데 사용됩니다. (MPRAPI_TUNNEL_CONFIG_PARAMS1)
PPP_ATCP_INFO PPP_ATCP_INFO 구조에는 PPP AppleTalk 프로젝션 작업의 결과가 포함됩니다.
PPP_CCP_INFO PPP_CCP_INFO 구조에는 CCP(압축 제어 프로토콜) 협상 결과를 설명하는 정보가 포함되어 있습니다.
PPP_INFO PPP_INFO 구조체는 연결에 대한 다양한 PPP(Point to Point) 프로젝션 작업의 결과를 보고하는 데 사용됩니다.
PPP_INFO_2 PPP_INFO_2 구조체는 연결에 대한 다양한 PPP(Point to Point) 프로젝션 작업의 결과를 보고하는 데 사용됩니다.
PPP_INFO_3 PPP_INFO_3 구조체는 연결에 대한 다양한 PPP(Point to Point) 프로젝션 작업의 결과를 보고하는 데 사용됩니다.
PPP_IPCP_INFO PPP_IPCP_INFO 구조에는 PPP IP(인터넷 프로토콜) 협상 결과가 포함됩니다.
PPP_IPCP_INFO2 PPP_IPCP_INFO2 구조에는 PPP IP(인터넷 프로토콜) 협상 결과가 포함됩니다.

PPP_IPV6_CP_INFO IPv6 제어 프로토콜 협상 결과를 포함합니다.
PPP_IPXCP_INFO PPP_IPXCP_INFO 구조에는 PPP IPX(Internetwork Packet Exchange) 프로젝션 작업의 결과가 포함됩니다.
PPP_LCP_INFO PPP_LCP_INFO 구조에는 PPP LCP(링크 제어 프로토콜) 협상 결과를 설명하는 정보가 포함되어 있습니다.
PPP_NBFCP_INFO PPP_NBFCP_INFO 구조체에는 PPP NBF(NetBEUI Framer) 프로젝션 작업의 결과가 포함됩니다.
PPP_PROJECTION_INFO SSTP(Secure Socket Tunneling Protocol), PPTP(Point to Point Tunneling Protocol) 및 L2TP(Layer 2 Tunneling Protocol)에 대한 PPP(지점 간) 협상 중에 얻은 정보를 포함합니다. (PPP_PROJECTION_INFO)
PPP_PROJECTION_INFO2 SSTP(Secure Socket Tunneling Protocol), PPTP(Point to Point Tunneling Protocol) 및 L2TP(Layer 2 Tunneling Protocol)에 대한 PPP(지점 간) 협상 중에 얻은 정보를 포함합니다. (PPP_PROJECTION_INFO2)
PPTP_CONFIG_PARAMS RAS 서버에서 PPTP(지점 간 터널링 프로토콜)에 대한 디바이스 구성을 가져와서 설정하는 데 사용됩니다.
PROJECTION_INFO RAS_CONNECTION_EX 구조체에서 PPP_PROJECTION_INFO 및 IKEV2_PROJECTION_INFO 구조체의 자리 표시자로 사용됩니다.
PROJECTION_INFO2 RAS_CONNECTION_4 구조체에서 PPP_PROJECTION_INFO2 및 IKEV2_PROJECTION_INFO2 구조체의 자리 표시자로 사용됩니다.

RAS_CONNECTION_0 RAS_CONNECTION_0 구조에는 사용자 이름 또는 도메인과 같은 특정 연결에 대한 일반적인 정보가 포함됩니다. 전송되거나 수신된 바이트와 같은 특정 연결에 대한 자세한 내용은 RAS_CONNECTION_1 참조하세요.
RAS_CONNECTION_1 RAS_CONNECTION_1 구조에는 수신된 오류 수 및 바이트와 같은 특정 연결에 대한 자세한 정보가 포함됩니다. 사용자 이름 또는 도메인과 같은 특정 연결에 대한 자세한 내용은 RAS_CONNECTION_0 참조하세요.
RAS_CONNECTION_2 RAS_CONNECTION_2 구조에는 연결을 식별하는 GUID(Globally Unique Identifier)를 포함하여 연결에 대한 정보가 포함됩니다.
RAS_CONNECTION_3 RAS_CONNECTION_3 구조에는 연결 및 연결의 격리 상태를 식별하는 GUID(Globally Unique Identifier)를 포함하여 연결에 대한 정보가 포함됩니다.
RAS_CONNECTION_4 연결에 연결된 :_the 사용자 이름, 도메인, GUID(Globally Unique Identifier), NAP(네트워크 액세스 보호) 격리 상태, 패킷 통계, PPP(Point-to-Point) 및 IKEv2(Internet Key Exchange 버전 2) 관련 정보를 포함하는 연결에 대한 특정 정보를 포함합니다.
RAS_CONNECTION_EX 연결에 연결된 :_the 사용자 이름, 도메인 및 GUID(Globally Unique Identifier), NAP(네트워크 액세스 보호) 격리 상태, 패킷 통계, PPP(Point-to-Point) 및 IKEv2(Internet Key Exchange 버전 2) 관련 정보가 포함된 연결에 대한 특정 정보를 포함합니다.
RAS_PORT_0 RAS_PORT_0 구조에는 포트 조건 및 포트 이름과 같은 특정 RAS 포트에 대한 일반적인 정보가 포함됩니다. 라인 속도 또는 오류와 같은 특정 포트에 대한 자세한 내용은 RAS_PORT_1 참조하세요.
RAS_PORT_1 RAS_PORT_1 구조에는 라인 속도 또는 오류와 같은 특정 RAS 포트에 대한 자세한 정보가 포함되어 있습니다. 포트 조건 또는 포트 이름과 같은 포트에 대한 자세한 내용은 RAS_PORT_0 참조하세요.
RAS_PROJECTION_INFO RAS 연결에 대한 PPP(Point to Point) 또는 IKEv2(Internet Key Exchange 버전 2) 프로젝션 정보를 포함합니다.

RAS_SECURITY_INFO RAS_SECURITY_INFO 구조는 RasSecurityDialogGetInfo 함수와 함께 RAS 보안 DLL 인증 트랜잭션과 연결된 RAS 포트에 대한 정보를 반환하는 데 사용됩니다.
RAS_STATS RAS_STATS 구조는 단일 링크 RAS 연결 또는 다중 링크 RAS 연결의 링크 중 하나에 대한 통계를 저장합니다.
RAS_UPDATE_CONNECTION 활성 RAS 연결을 업데이트하는 데 사용됩니다.
RAS_USER_0 RAS_USER_0 구조에는 특정 원격 액세스 서비스 사용자에 대한 정보가 포함됩니다.
RAS_USER_1 RAS_USER_1 구조에는 특정 원격 액세스 서비스 사용자에 대한 정보가 포함됩니다. RAS_USER_1 구조는 RAS_USER_1 추가 멤버인 bfPrivilege2를 지원한다는 점을 제외하고 RAS_USER_0 구조와 비슷합니다.
RASDEVSPECIFICINFO 서버 유효성 검사를 위해 쿠키를 보내고 PPP(지점 및 지점간) 인증을 바이패스하는 데 사용됩니다.
RASIKEV2_PROJECTION_INFO IKE(인터넷 키 교환) 협상 중에 얻은 프로젝션 정보를 포함합니다.
RASNAPSTATE 원격 액세스 연결에 대한 NAP(네트워크 액세스 보호) 변수입니다.
RASPPP_PROJECTION_INFO 인터넷 프로토콜 버전 4(IPv4) 및 IPv6 프로젝션 작업 및 PPP LCP(링크 제어 프로토콜)/멀티링크 및 CCP(압축 제어 프로토콜) 협상 중 얻은 정보를 포함합니다.
ROUTER_CUSTOM_IKEv2_POLICY0 IKEv2 기본 모드 및 빠른 모드 정책 구성을 포함합니다.
ROUTER_IKEv2_IF_CUSTOM_CONFIG0 IKEv2 터널 기반 수요 다이얼 인터페이스에 대한 IKEv2 터널 구성 매개 변수를 가져오거나 설정합니다.

ROUTING_PROTOCOL_CONFIG ROUTING_PROTOCOL_CONFIG 구조는 프로토콜이 멀티캐스트 그룹 관리자에 등록될 때 멀티캐스트 그룹 관리자에게 전달되는 라우팅 프로토콜 구성 정보를 설명합니다.
RTM_DEST_INFO RTM_DEST_INFO 구조는 라우팅 테이블 관리자에 등록된 클라이언트와 대상 정보를 교환하는 데 사용됩니다.
RTM_ENTITY_EXPORT_METHODS RTM_ENTITY_EXPORT_METHODS 구조에는 클라이언트에서 내보낸 메서드 집합이 포함됩니다.
RTM_ENTITY_ID RTM_ENTITY_ID 구조는 라우팅 테이블 관리자에 대한 클라이언트를 고유하게 식별하는 데 사용됩니다. 프로토콜 식별자와 인스턴스 식별자는 클라이언트를 고유하게 식별하는 데 사용되는 값입니다.
RTM_ENTITY_INFO RTM_ENTITY_INFO 구조는 라우팅 테이블 관리자와 클라이언트 정보를 교환하는 데 사용됩니다.
RTM_ENTITY_METHOD_INPUT RTM_ENTITY_METHOD_INPUT 구조체는 메서드를 호출할 때 클라이언트에 정보를 전달하는 데 사용됩니다.
RTM_ENTITY_METHOD_OUTPUT RTM_ENTITY_METHOD_OUTPUT 구조는 라우팅 테이블 관리자가 메서드를 호출할 때 호출 클라이언트에 정보를 전달하는 데 사용됩니다.
RTM_NET_ADDRESS RTM_NET_ADDRESS 구조는 주소 패밀리의 라우팅 테이블 관리자에게 주소 정보를 전달하는 데 사용됩니다. 주소 패밀리는 8바이트 미만의 연속 주소 마스크에서만 사용해야 합니다.
RTM_NEXTHOP_INFO RTM_NEXTHOP_INFO 구조는 라우팅 테이블 관리자와 다음 홉 정보를 교환하는 데 사용됩니다.
RTM_NEXTHOP_LIST RTM_NEXTHOP_LIST 구조에는 경로에서 동일한 비용 경로를 결정하는 데 사용되는 다음 홉 목록이 포함되어 있습니다.

RTM_PREF_INFO RTM_PREF_INFO 구조에는 두 경로를 비교할 때 사용되는 정보가 포함됩니다. 기본 설정 멤버의 값은 메트릭 멤버의 값보다 가중치가 더 높습니다.
RTM_REGN_PROFILE RTM_REGN_PROFILE 구조에는 등록 프로세스 중에 반환되는 정보가 포함됩니다. 이 정보는 이후 함수 호출(예: RtmGetEnumRoutes 호출에 의해 반환될 수 있는 최대 경로 수)에 사용됩니다.
RTM_ROUTE_INFO RTM_ROUTE_INFO 구조는 라우팅 테이블 관리자와 경로 정보를 교환하는 데 사용됩니다. 읽기 전용 정보를 변경하지 마세요.
SECURITY_MESSAGE SECURITY_MESSAGE 구조는 RasSecurityDialogComplete 함수와 함께 RAS 보안 DLL 인증 트랜잭션의 결과를 나타내는 데 사용됩니다.
SOURCE_GROUP_ENTRY SOURCE_GROUP_ENTRY 구조체는 그룹 열거형 함수 MgmGroupEnumerationGetNext에서 반환되는 항목을 설명합니다.
SSTP_CERT_INFO SSTP(Secure Socket Tunneling Protocol) 기반 인증서에 대한 정보를 포함합니다.
SSTP_CONFIG_PARAMS RAS 서버에서 SSTP(Secure Socket Tunneling Protocol)에 대한 디바이스 구성을 가져와서 설정하는 데 사용됩니다.
SUPPORT_FUNCTIONS_50 SUPPORT_FUNCTIONS_50 구조체(routprot.h)는 라우터 관리자가 제공하는 함수에 대한 포인터 집합을 라우팅 프로토콜에 전달합니다.
SUPPORT_FUNCTIONS_60 SUPPORT_FUNCTIONS_60 구조체(routprot.h)는 라우터 관리자가 제공하는 함수에 대한 포인터 집합을 라우팅 프로토콜에 전달합니다.
UPDATE_COMPLETE_MESSAGE UPDATE_COMPLETE_MESSAGE 구조에는 업데이트 작업의 완료 상태를 설명하는 정보가 포함되어 있습니다.

보안 및 ID

보안 및 ID 기술 개요입니다.

보안 및 ID를 개발하려면 다음 헤더가 필요합니다.

- aclapi.h
- aclui.h
- adtgen.h
- authz.h
- azroles.h
- bcrypt.h
- casetup.h
- ccplugins.h
- celib.h
- certadm.h
- certbcli.h
- certcli.h
- certenc.h
- certenroll.h
- certexit.h
- certif.h
- certmod.h
- certpol.h
- certpoleng.h
- certsrv.h
- certview.h
- credssp.h
- cryptdlg.h
- cryptuiapi.h
- cryptxml.h
- diagnosticdataquery.h
- diagnosticdataquerytypes.h
- dpapi.h
- dssec.h
- iads.h
- identitycommon.h
- identityprovider.h
- identitystore.h
- isolatedwindowsenvironmentutils.h
- keycredmgr.h

- lsalookup.h
- mscat.h
- mssip.h
- ncrypt.h
- ncryptprotect.h
- npapi.h
- ntlsa.h
- ntsecapi.h
- sas.h
- scesvc.h
- schannel.h
- sddl.h
- securityappcontainer.h
- slpublic.h
- sspi.h
- subauth.h
- tokenbinding.h
- tpmvscmgr.h
- wincred.h
- wincrypt.h
- winnetwk.h
- winsafer.h
- winscard.h
- winsvc.h
- wintrust.h
- winwlx.h
- xenroll.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 인증
- 승인
- 보안 API 대한 모범 사례
- 인증서 등록 API
- 암호화 API: 차세대
- 암호화
- 보안 용어집
- 보안 관리
- 보안 WMI 공급자
- 소프트웨어 라이선스 API

열거 목록

테이블 확장

ACCESS_MODE EXPLICIT_ACCESS 구조의 액세스 권한이 트러스티에 적용되는 방식을 나타내는 값을 포함합니다.
ACL_INFORMATION_CLASS ACL(액세스 제어 목록)에 할당되거나 검색되는 정보의 유형을 지정하는 값을 포함합니다.
AlgorithmFlags 암호화 알고리즘에 대한 검색을 구체화하는 데 사용할 수 있는 플래그를 포함합니다.
AlgorithmOperationFlags 알고리즘이 수행할 수 있는 작업을 지정합니다.
AlgorithmType 암호화 공급자가 지원하는 암호화 알고리즘의 용도를 지정합니다.
AlternativeNameType IAlternativeName 개체를 초기화할 때 지정할 수 있는 대체 이름 형식을 지정합니다.
AUDIT_EVENT_TYPE 감사되는 개체의 형식을 나타내는 값을 정의합니다. AccessCheckByTypeAndAuditAlarm 및 AccessCheckByTypeResultListAndAuditAlarm 함수는 이러한 값을 사용합니다.
AUDIT_PARAM_TYPE 사용할 수 있는 감사 매개 변수의 유형을 정의합니다.
AUTHZ_CONTEXT_INFORMATION_CLASS 기존 AuthzClientContext에서 검색할 정보의 형식을 지정합니다. 이 열거형은 AuthzGetInformationFromContext 함수에서 사용됩니다.
AUTHZ_SECURITY_ATTRIBUTE_OPERATION AuthzModifySecurityAttributes 함수를 호출하여 보안 특성에 대해 수정할 형식을 나타냅니다.
AUTHZ_SID_OPERATION

AuthzModifySids 함수를 호출하여 수행할 수 있는 SID 작업의 형식을 나타냅니다.
AZ_PROP_CONSTANTS 권한 부여 관리자에서 사용하는 상수 정의
BCRYPT_HASH_OPERATION_TYPE BCRYPT_HASH_OPERATION_TYPE 열거형은 해시 작업 유형을 지정합니다.
BCRYPT_MULTI_OPERATION_TYPE BCRYPT_MULTI_OPERATION_TYPE 열거형은 BCryptProcessMultiOperations 함수에 전달되는 다중 작업의 형식을 지정합니다.
CASetupProperty ICertSrvSetup 인터페이스를 사용할 때 CA(인증 기관) 역할의 설정 및 구성을 위한 속성 형식을 지정합니다.
CEPSetupProperty ICertificateEnrollmentPolicyServerSetup 인터페이스의 GetProperty 및 SetProperty 메서드에서 검색하거나 설정할 속성 정보의 형식을 지정하는 데 사용됩니다.
CERTENROLL_OBJECTID 인증서 등록 API에서 지원하는 미리 정의된 OID(개체 식별자)를 포함합니다.
CERTENROLL_PROPERTYID 인증서 저장소의 인증서와 연결할 수 있는 외부 속성에 대해 미리 정의된 개체 식별자를 포함합니다.
CESSetupProperty ICertificateEnrollmentServerSetup 인터페이스의 GetProperty 및 SetProperty 메서드에서 검색하거나 설정할 속성 정보의 형식을 지정하는 데 사용됩니다.
CommitTemplateFlags 템플릿 저장 및 삭제 옵션을 지정합니다.
CRED_MARSHAL_TYPE CredMarshalCredential에 의해 마샬링되거나 CredUnmarshalCredential에 의해 unmarshaledial로 마샬링될 자격 증명 유형을 지정합니다.
CRED_PROTECTION_TYPE CredProtect 함수를 사용할 때 자격 증명이 암호화되는 보안 컨텍스트를 지정합니다.

CREDSPP_SUBMIT_TYPE CREDSSP_CRED 구조에 지정된 자격 증명의 유형을 지정합니다.
CRYPT_XML_CHARSET XML에 사용되는 문자 집합을 지정하는 데 사용됩니다.
CRYPT_XML_KEYINFO_SPEC CryptXmlSign 함수에서 dwKeyInfoSpec 매개 변수의 값을 지정합니다.
CRYPT_XML_PROPERTY_ID XML 속성의 형식 및 사용을 지정합니다.
DdqAccessLevel 이 리소스는 진단 데이터 쿼리 세션에 대한 권한 수준을 나타냅니다.
DSAFIPSVERSION_ENUM FIPS 버전 정보를 포함합니다.
EncodingType 표시를 위해 바이트 배열에 적용되는 인코딩 유형을 지정합니다.
EnrollmentCAProperty 인증 기관 속성 값을 지정합니다.
EnrollmentDisplayStatus 사용자 인터페이스에 등록 상태 정보를 표시할지 여부를 지정합니다.
EnrollmentEnrollStatus 인증서 요청의 등록 상태를 지정합니다.
EnrollmentPolicyFlags 그룹 정책 플래그를 지정합니다.
EnrollmentPolicyServerPropertyFlags 기본 정책 서버를 지정합니다.
EnrollmentSelectionStatus

등록 프로세스 중에 개체의 등록 상태를 모니터링할지 여부를 지정합니다.
EnrollmentTemplateProperty 지정된 템플릿에 대한 속성 값을 포함합니다.
ENUM_CATYPES CA(인증 기관) 유형을 지정합니다.
ENUM_PERIOD 시간 범위의 단위를 지정합니다.
eTlsAlgorithmUsage 암호화 설정을 사용하지 않도록 설정하는 데 사용되는 알고리즘을 지정합니다.
HASHALGORITHM_ENUM 서명 및 해시 알고리즘을 지정합니다.
IDENTITY_TYPE 열거할 ID의 형식을 지정합니다.
ImportPFXFlags PFX 인증서를 가져올 때 사용할 플래그입니다.
InnerRequestLevel PKCS 내에서 인증서 요청의 포함 수준을 지정합니다.
InstallResponseRestrictionFlags 인증서 체인의 로컬 설치에 적용되는 제한을 식별하는 플래그를 포함합니다.
KERB_CERTIFICATE_INFO_TYPE 제공되는 인증서 정보의 유형을 지정합니다.
KERB_LOGON_SUBMIT_TYPE 요청되는 로그온 유형을 식별합니다.
KERB_PROFILE_BUFFER_TYPE 반환된 로그온 프로필의 유형을 나열합니다.

KERB_PROTOCOL_MESSAGE_TYPE LsaCallAuthenticationPackage 함수를 호출하여 Kerberos 인증 패키지로 보낼 수 있는 메시지 유형을 나열합니다.
KeyCredentialManagerOperationErrorStates KeyCredentialManagerGetOperationErrorStates 함수에서 플래그로 반환된 오류 상태의 열거형입니다.
KeyCredentialManagerOperationType KeyCredentialManagerShowUIOperation에 전달되는 작업 열거형 값입니다.
KeyIdentifierHashAlgorithm 인증서 요청에서 공개 키를 해시하는 데 사용되는 알고리즘을 지정합니다.
LSA_FOREST_TRUST_COLLISION_RECORD_TYPE 로컬 보안 기관 포리스트 트러스트 레코드 간에 발생할 수 있는 충돌 유형을 정의합니다.
LSA_FOREST_TRUST_RECORD_TYPE 로컬 보안 기관 포리스트 트러스트 레코드의 형식을 정의합니다.
LSA_TOKEN_INFORMATION_TYPE 로그온 토큰에 포함할 수 있는 정보 수준을 지정합니다.
MANDATORY_LEVEL 가능한 보안 수준을 나열합니다.
MSA_INFO_LEVEL 관리되는 서비스 계정의 수준을 나타냅니다.
MSA_INFO_STATE 관리되는 서비스 계정의 상태를 나타냅니다.
MSCEPSetupProperty IMSCEPSetup을 사용하여 MICROSOFT SCEP(단순 인증서 등록 프로토콜) 역할의 설정 및 구성을 위한 속성 형식을 지정합니다.
MSV1_0_LOGON_SUBMIT_TYPE 요청되는 로그온의 종류를 나타냅니다.

MSV1_0_PROFILE_BUFFER_TYPE 반환된 로그온 프로필의 종류를 나열합니다.
MSV1_0_PROTOCOL_MESSAGE_TYPE LsaCallAuthenticationPackage 함수를 호출하여 MSV1_0 인증 패키지로 보낼 수 있는 메시지 유형을 나열합니다.
MULTIPLE_TRUSTEE_OPERATION TRUSTEE 구조가 가장 트러스티인지 여부를 나타내는 값을 포함합니다.
ObjectIdGroupId OID(개체 식별자)가 속한 범주 또는 그룹을 지정합니다.
ObjectIdPublicKeyFlags 공개 키 알고리즘을 서명 또는 암호화에 사용할지 여부를 지정합니다.
PFXExportOptions PFX(개인 정보 교환) 메시지를 만들 때 포함되는 인증서 체인의 양을 지정합니다.
Pkcs10AllowedSignatureTypes 인증서 요청에 서명할 때 허용되는 서명 유형을 지정합니다.
PKU2U_LOGON_SUBMIT_TYPE PKU2U_CERTIFICATE_S4U_LOGON 구조에 전달된 로그온 메시지의 유형을 나타냅니다.
POLICY_AUDIT_EVENT_TYPE POLICY_AUDIT_EVENT_TYPE 열거형은 시스템에서 감사할 수 있는 이벤트 유형을 나타내는 값을 정의합니다.
POLICY_DOMAIN_INFORMATION_CLASS 정책 도메인 정보의 유형을 정의합니다.
POLICY_INFORMATION_CLASS Policy 개체에서 설정하거나 쿼리할 정보 유형을 나타내는 값을 정의합니다.
POLICY_LSA_SERVER_ROLE LSA 서버의 역할을 나타내는 값을 정의합니다.

POLICY_NOTIFICATION_INFORMATION_CLASS POLICY_NOTIFICATION_INFORMATION_CLASS 열거형은 애플리케이션에서 변경 알림을 요청할 수 있는 정책 정보 및 정책 도메인 정보의 유형을 정의합니다.
POLICY_SERVER_ENABLE_STATE POLICY_SERVER_ENABLE_STATE 열거형은 LSA 서버의 상태를 나타냅니다. 즉, LSA 서버가 활성화되어 있는지 또는 사용하지 않도록 설정되었는지 여부입니다. 일부 작업은 사용하도록 설정된 LSA 서버에서만 수행할 수 있습니다.
PolicyQualifierType 인증서 정책에 적용되는 한정자 유형을 지정합니다.
PolicyServerUrlFlags CEP(인증서 등록 정책) 서버 플래그를 포함합니다.
PolicyServerUrlPropertyID GetStringProperty 메서드에서 반환하거나 IX509PolicyServerUrl 인터페이스의 SetStringProperty 메서드에서 설정할 속성 값의 형식을 지정하는 값을 포함합니다.
PROG_INVOKE_SETTING TreeSetNamedSecurityInfo 또는 TreeResetNamedSecurityInfo 함수에 대한 호출의 진행률을 추적하는 데 사용되는 함수의 초기 설정을 나타냅니다.
RequestClientInfoClientId 인증서 요청을 만든 애플리케이션의 유형을 지정합니다.
SAFER_IDENTIFICATION_TYPES SAFER_IDENTIFICATION_HEADER 구조체로 식별할 수 있는 식별 규칙 구조의 가능한 형식을 정의합니다.
SAFER_OBJECT_INFO_CLASS Safer 개체에 대해 요청된 정보의 형식을 정의합니다.
SAFER_POLICY_INFO_CLASS 정책을 쿼리할 수 있는 방법을 정의합니다.
SCESVC_INFO_TYPE SCESVC_INFO_TYPE 열거형은 PFSCE_QUERY_INFO PFSCE_SET_INFO 보안 데이터베이스에서 요청되거나 보안 데이터베이스에 전달된 정보의 유형을 나타내는 데 사용됩니다. 다음 값 중 하나일 수 있습니다.

SE_OBJECT_TYPE 보안을 지원하는 Windows 개체 유형에 해당하는 값을 포함합니다.
SEC_APPLICATION_PROTOCOL_NEGOTIATION_EXT 협상된 애플리케이션 프로토콜 확장을 나타냅니다.
SEC_APPLICATION_PROTOCOL_NEGOTIATION_STATUS SEC 애플리케이션 프로토콜 협상의 상태를 설명합니다.
SEC_TRAFFIC_SECRET_TYPE 사용된 트래픽 비밀 유형을 나타냅니다.
SecDelegationType 사용되는 위임의 형식입니다.
SECPKG_ATTR_LCT_STATUS InitializeSecurityContext 함수에 대한 최신 호출의 토큰이 클라이언트의 마지막 토큰인지 여부를 나타냅니다.
SECPKG_CRED_CLASS 클라이언트 컨텍스트에서 사용되는 자격 증명의 형식을 나타냅니다. SECPKG_CRED_CLASS 열거형은 SecPkgContext_CredInfo 구조체에서 사용됩니다.
SECPKG_EXTENDED_INFORMATION_CLASS SECPKG_EXTENDED_INFORMATION_CLASS 열거형은 보안 패키지에 대해 설정하거나 가져올 정보 유형을 설명합니다. 이 열거형은 SpGetExtendedInformation 및 SpSetExtendedInformation 함수에서 사용됩니다.
SECPKG_NAME_TYPE SECPKG_NAME_TYPE 열거형은 계정에 지정된 이름의 유형을 설명하는 데 사용됩니다. SECPKG_NAME_TYPE 열거형은 GetAuthDataForUser 및 OpenSamUser 함수에서 사용됩니다.
SECPKG_SESSIONINFO_TYPE 세션 정보의 형식을 지정합니다.
SECURITY_IMPERSONATION_LEVEL 보안 가장 수준을 지정하는 값을 포함합니다. 보안 가장 수준은 서버 프로세스가 클라이언트 프로세스를 대신하여 작동할 수 있는 정도를 제어합니다.

SECURITY_LOGON_TYPE 로그온 프로세스에서 요청한 로그온 유형을 나타냅니다.
SERVICE_DIRECTORY_TYPE 서비스별 디렉터리 경로의 형식을 지정합니다.
SERVICE_REGISTRY_STATE_TYPE 서비스 레지스트리 키의 상태 유형을 지정합니다. (SERVICE_REGISTRY_STATE_TYPE)
SERVICE_SHARED_DIRECTORY_TYPE 서비스별 공유 디렉터리 경로의 형식을 지정합니다.
SERVICE_SHARED_REGISTRY_STATE_TYPE 서비스 레지스트리 키의 상태 유형을 지정합니다. (SERVICE_SHARED_REGISTRY_STATE_TYPE)
SI_PAGE_TYPE 액세스 제어 편집기 속성 시트의 속성 페이지 유형을 나타내는 값을 포함합니다.
SID_NAME_USE SID(보안 식별자) 형식을 지정하는 값을 포함합니다.
SL_ACTIVATION_TYPE 라이선스에 대한 오프라인 정품 인증 유형을 나타냅니다.
SL_GENUINE_STATE 애플리케이션 설치의 상태를 지정합니다.
SLDATATYPE SLGetWindowsInformation 함수에서 반환된 버퍼의 데이터 형식을 지정합니다.
슬라이딩타입 소프트웨어 라이선스 ID의 유형을 나타냅니다.
SLLICENSINGSTATUS 라이선스 상태를 나타냅니다. (SLLICENSINGSTATUS)
SLREFERRALTYPE

SLGetReferralInformation 함수를 사용하여 쿼리할 수 있는 정보 유형을 나타냅니다.
TOKEN_ELEVATION_TYPE GetTokenInformation 함수에서 쿼리하거나 SetTokenInformation 함수에 의해 설정되는 토큰의 권한 상승 유형을 나타냅니다.
TOKEN_INFORMATION_CLASS 액세스 토큰에 할당되거나 액세스 토큰에서 검색되는 정보의 유형을 지정하는 값을 포함합니다.
TOKEN_TYPE 기본 토큰과 가장 토큰을 구분하는 값을 포함합니다.
TOKENBINDING_EXTENSION_FORMAT 확장 데이터를 해석하는 데 사용할 수 있는 형식을 지정합니다.
TOKENBINDING_TYPE 토큰 바인딩에 사용할 수 있는 형식을 지정합니다.
TPMVSCMGR_ERROR TPM 가상 스마트 카드 관리자의 오류 컨텍스트를 나타내는 미리 정의된 오류 코드를 제공합니다.
TPMVSCMGR_STATUS TPM 가상 스마트 카드 관리자의 진행률을 나타내는 미리 정의된 상태 코드를 제공합니다.
TRUSTED_INFORMATION_CLASS TRUSTED_INFORMATION_CLASS 열거형 형식은 신뢰할 수 있는 도메인에 대해 설정하거나 쿼리할 정보 형식을 나타내는 값을 정의합니다.
TRUSTEE_FORM TRUSTEE 구조의 ptstrName 멤버가 가리키는 데이터 형식을 나타내는 값입니다.
TRUSTEE_TYPE TRUSTEE 구조로 식별되는 트러스티의 형식을 나타내는 값입니다.
WebEnrollmentFlags 웹 등록 동작을 지정합니다.
WebSecurityLevel

웹 사용 메서드 또는 속성이 스크립팅에 안전한지 여부를 지정합니다.
WELL_KNOWN_SID_TYPE 일반적으로 사용되는 SID(보안 식별자) 목록입니다. 프로그램은 이러한 값을 CreateWellKnownSid 함수에 전달하여 이 목록에서 SID를 만들 수 있습니다.
X500NameFlags 고유 이름 또는 RDN(상대 고유 이름)의 표시 및 인코딩 특성을 지정합니다.
X509CertificateEnrollmentContext 인증서가 의도된 최종 엔터티의 특성을 지정합니다.
X509CertificateTemplateEnrollmentFlag 등록하는 동안 서버 및 클라이언트 작업을 지정하는 값을 포함합니다.
X509CertificateTemplateGeneralFlag 템플릿 및 관련 인증서에 대한 사용 및 수정 정보를 포함합니다.
X509CertificateTemplatePrivateKeyFlag 프라이빗 키에 대한 클라이언트 작업을 지정하는 값을 포함합니다.
X509CertificateTemplateSubjectNameFlag 주체 이름과 관련된 서버 및 클라이언트 작업을 지정하는 값을 포함합니다.
X509EnrollmentAuthFlags 인증 유형을 지정합니다.
X509EnrollmentPolicyExportFlags IX509EnrollmentPolicyServer 인터페이스의 Export 메서드에서 정책 서버에서 내보낼 항목을 지정하는 데 사용됩니다.
X509EnrollmentPolicyLoadOption IX509EnrollmentPolicyServer 인터페이스의 LoadPolicy 메서드에서 정책 서버에서 정책을 검색하는 방법을 지정하는 데 사용됩니다.
X509KeySpec 레거시 CSP(암호화 서비스 공급자)에 대한 키의 의도된 사용을 지정합니다.

X509KeyUsageFlags 인증서에 포함된 키의 용도를 지정합니다.
X509PrivateKeyExportFlags 프라이빗 키에 대한 내보내기 정책을 지정합니다.
X509PrivateKeyProtection 암호화 공급자가 지원하는 프라이빗 키 보호 수준을 지정합니다.
X509PrivateKeyUsageFlags 프라이빗 키의 허용된 사용을 지정합니다.
X509PrivateKeyVerify 프라이빗 키 확인 중에 사용자 인터페이스가 표시되는지 여부와 암호화 공급자가 스마트 카드 공급자인 경우 확인을 진행할 수 있는지 여부를 지정합니다.
X509ProviderType 암호화 공급자의 유형을 지정합니다.
X509RequestInheritOptions 기존 인증서에서 새 요청을 만들 때 키, 확장 값 및 외부 속성을 상속하는 방법을 지정합니다.
X509RequestType 인증서 요청 유형을 지정합니다.
X509SCEPDisposition 응답 메시지를 처리하기 위한 요청의 결과 처리에 대해 설명합니다.
X509SCEPFailInfo SCEP 인증서 등록 실패의 특성에 대해 설명합니다.

기능

테이블 확장

acceptFilePKCS7 PKCS를 포함하는 파일을 수락하고 처리합니다.
acceptFilePKCS7WStr PKCS 수락 및 처리(IEnroll.acceptFilePKCS7WStr)
acceptFileResponse createFileRequest에 대한 이전 호출에 대한 응답으로 발급된 자격 증명 배달을 수락하고 자격 증명을 적절한 저장소에 배치합니다.
acceptFileResponseWStr createFileRequestWStr에 대한 이전 호출에 대한 응답으로 발급된 자격 증명 배달을 수락하고 자격 증명을 적절한 저장소에 배치합니다.
acceptPKCS7 PKCS 수락 및 처리(ICEnroll.acceptPKCS7)
acceptPKCS7Blob PKCS 수락 및 처리(IEnroll.acceptPKCS7Blob)
acceptResponse createRequest에 대한 이전 호출에 대한 응답으로 발급된 자격 증명 배달을 수락하고 자격 증명을 적절한 저장소에 배치합니다.
acceptResponseBlob createRequestWStr에 대한 이전 호출에 대한 응답으로 발급된 자격 증명 배달을 수락하고 자격 증명을 적절한 저장소에 배치합니다.
보안 컨텍스트 수락 전송 애플리케이션의 서버 구성 요소가 서버와 원격 클라이언트 간에 보안 컨텍스트를 설정할 수 있도록 합니다.
AccessCheck 보안 설명자가 액세스 토큰으로 식별된 클라이언트에 지정된 액세스 권한 집합을 부여할지 여부를 결정합니다. (AccessCheck)
AccessCheck 현재 클라이언트 컨텍스트에서 지정된 작업을 수행할 수 있는지 여부를 결정합니다.

AccessCheck2 현재 클라이언트 컨텍스트에서 나타내는 보안 주체가 지정된 작업을 수행할 수 있는지 여부를 지정하는 값을 반환합니다.
AccessCheckAndAuditAlarmA 보안 설명자가 호출 스레드에서 가장하는 클라이언트에 대해 지정된 액세스 권한 집합을 부여할지 여부를 결정합니다. (AccessCheckAndAuditAlarmA)
AccessCheckAndAuditAlarmW 보안 설명자가 호출 스레드에서 가장하는 클라이언트에 대해 지정된 액세스 권한 집합을 부여할지 여부를 결정합니다.
AccessCheckByType 보안 설명자가 액세스 토큰으로 식별된 클라이언트에 지정된 액세스 권한 집합을 부여할지 여부를 결정합니다. (AccessCheckByType)
AccessCheckByTypeAndAuditAlarmA 보안 설명자가 호출 스레드에서 가장하는 클라이언트에 대해 지정된 액세스 권한 집합을 부여할지 여부를 결정합니다. (AccessCheckByTypeAndAuditAlarmA)
AccessCheckByTypeAndAuditAlarmW 보안 설명자가 호출 스레드에서 가장하는 클라이언트에 대해 지정된 액세스 권한 집합을 부여할지 여부를 결정합니다. (AccessCheckByTypeAndAuditAlarmW)
AccessCheckByTypeResultList 보안 설명자가 액세스 토큰으로 식별된 클라이언트에 지정된 액세스 권한 집합을 부여할지 여부를 결정합니다. (AccessCheckByTypeResultList)
AccessCheckByTypeResultListAndAuditAlarmA 보안 설명자가 호출 스레드에서 가장하는 클라이언트에 대해 지정된 액세스 권한 집합을 부여할지 여부를 결정합니다. (AccessCheckByTypeResultListAndAuditAlarmA)
AccessCheckByTypeResultListAndAuditAlarmByHandleA AccessCheckByTypeResultListAndAuditAlarmByHandleA(ANSI) 함수(winbase.h)는 보안 설명자가 호출 스레드가 가장하는 클라이언트에 지정된 액세스 권한 집합을 부여할지 여부를 결정합니다.
AccessCheckByTypeResultListAndAuditAlarmByHandleW AccessCheckByTypeResultListAndAuditAlarmByHandleW(유니코드) 함수(securitybaseapi.h)는 보안 설명자가 호출 스레드가 가장하는 클라이언트에 대한 액세스 권한을 부여할지 여부를 결정합니다.

AccessCheckByTypeResultListAndAuditAlarmW 보안 설명자가 호출 스레드에서 가장하는 클라이언트에 대해 지정된 액세스 권한 집합을 부여할지 여부를 결정합니다. (AccessCheckByTypeResultListAndAuditAlarmW)
AcquireCredentialsHandleA AcquireCredentialsHandle(CredSSP) 함수는 보안 주체의 기존 자격 증명에 대한 핸들을 획득합니다. (ANSI)
AcquireCredentialsHandleW AcquireCredentialsHandle(CredSSP) 함수는 보안 주체의 기존 자격 증명에 대한 핸들을 획득합니다. (유니코드)
추가 컬렉션에 개체를 추가합니다. (IAlternativeNames.Add)
추가 컬렉션에 개체를 추가합니다. (ICertificatePolicies.Add)
추가 컬렉션에 ICertificationAuthority 개체를 추가합니다.
추가 컬렉션에 속성을 추가합니다.
추가 컬렉션에 ICryptAttribute 개체를 추가합니다.
추가 컬렉션에 ICspAlgorithm 개체를 추가합니다.
추가 컬렉션에 ICspInformation 개체를 추가합니다.
추가 컬렉션에 ICspStatus 개체를 추가합니다.
추가 컬렉션에 IObjectId 개체를 추가합니다.

추가 컬렉션에 개체를 추가합니다. (IPolicyQualifiers.Add)
추가 컬렉션에 ISignerCertificate 개체를 추가합니다.
추가 컬렉션에 ISmimeCapability 개체를 추가합니다.
추가 컬렉션에 IX509Attribute 개체를 추가합니다.
추가 컬렉션에 IX509CertificateTemplate 개체를 추가합니다.
추가 컬렉션에 IX509Extension 개체를 추가합니다.
추가 컬렉션에 IX509NameValuePair 개체를 추가합니다.
추가 컬렉션에 IX509PolicyServerUrl 개체를 추가합니다.
추가 컬렉션에 ICertSrvSetupKeyInformation 개체를 추가합니다.
AddAccessAllowedAce 액세스 허용 ACE(액세스 제어 항목)를 ACL(액세스 제어 목록)에 추가합니다. 지정된 SID(보안 식별자)에 대한 액세스 권한이 부여됩니다.
AddAccessAllowedAceEx DACL(임의 액세스 제어 목록)의 끝에 ACE(액세스 허용 액세스 제어 항목)를 추가합니다. (AddAccessAllowedAceEx)
AddAccessAllowedObjectAce

DACL(임의 액세스 제어 목록)의 끝에 ACE(액세스 허용 액세스 제어 항목)를 추가합니다. (AddAccessAllowedObjectAce)
AddAccessDeniedAce 액세스 거부 ACE(액세스 거부 액세스 제어 항목)를 ACL(액세스 제어 목록)에 추가합니다. 지정된 SID(보안 식별자)에 대한 액세스가 거부됩니다.
AddAccessDeniedAceEx DACL(임의 액세스 제어 목록)의 끝에 ACE(액세스 거부 액세스 제어 항목)를 추가합니다.
AddAccessDeniedObjectAce DACL(임의 액세스 제어 목록)의 끝에 ACE(액세스 거부 액세스 제어 항목)를 추가합니다. 새 ACE는 개체 또는 개체의 속성 집합 또는 속성에 대한 액세스를 거부할 수 있습니다.
AddAce 지정된 ACL(액세스 제어 목록)에 하나 이상의 ACE(액세스 제어 항목)를 추가합니다.
AddApplicationGroups 기존 IAzApplicationGroup 개체의 지정된 배열을 클라이언트 컨텍스트 개체에 추가합니다.
AddAppMember 지정된 IAzApplicationGroup 개체를 이 애플리케이션 그룹에 속한 애플리케이션 그룹 목록에 추가합니다.
AddAppMember 지정된 IAzApplicationGroup 개체를 이 역할에 속한 애플리케이션 그룹 목록에 추가합니다.
AddAppNonMember 지정된 IAzApplicationGroup 개체를 이 애플리케이션 그룹의 멤버 자격이 거부된 애플리케이션 그룹 목록에 추가합니다.
addAttributeToRequest 인증서 요청에 특성을 추가합니다. 이 메서드는 ICEnroll4 인터페이스에서 처음 정의되었습니다.
addAttributeToRequestWStr 인증서 요청에 특성을 추가합니다.
AddAuditAccessAce

시스템 감사 ACE(액세스 제어 항목)를 ACL(시스템 액세스 제어 목록)에 추가합니다. 지정된 SID(보안 식별자)의 액세스가 감사됩니다.
AddAuditAccessAceEx 시스템 감사 ACE(액세스 제어 항목)를 SACL(시스템 액세스 제어 목록) 끝에 추가합니다. (AddAuditAccessAceEx)
AddAuditAccessObjectAce 시스템 감사 ACE(액세스 제어 항목)를 SACL(시스템 액세스 제어 목록) 끝에 추가합니다. (AddAuditAccessObjectAce)
AddAuthenticatedAttributesToPKCS7Request AddAuthenticatedAttributesToPKCS7Request 메서드는 PKCS에 인증된 특성을 추가합니다.
AddAvailableCsps 컴퓨터에 설치된 공급자를 컬렉션에 추가합니다.
AddAvailableSmimeCapabilities 기본 RSA 암호화 공급자가 지원하는 암호화 알고리즘을 식별하여 컬렉션에 ISmimeCapability 개체를 추가합니다.
addBlobPropertyToCertificate 인증서에 BLOB 속성을 추가합니다.
addBlobPropertyToCertificateWStr IEnroll4::addBlobPropertyToCertificateWStr 메서드는 인증서에 BLOB 속성을 추가합니다.
AddCertificate 인증 키를 지원하는 KSP(키 스토리지 공급자)에 인증 키 인증서를 추가합니다.
addCertTypeToRequest 요청에 인증서 템플릿을 추가합니다(CA(엔터프라이즈 인증 기관)를 지원하는 데 사용됨). 이 메서드는 ICEnroll2 인터페이스에 의해 처음 정의되었습니다.
addCertTypeToRequestEx 요청에 인증서 템플릿(또는 "인증서 유형")을 추가합니다.
AddCertTypeToRequestWStr

요청에 인증서 템플릿을 추가합니다(CA(엔터프라이즈 인증 기관)를 지원하는 데 사용됨).
AddCertTypeToRequestWStrEx 인증서 템플릿(인증서 유형이라고도 함)을 요청에 추가합니다.
AddConditionalAce 지정된 ACL(액세스 제어 목록)에 ACE(조건부 액세스 제어 항목)를 추가합니다.
AddConnectNotify 각 연결 추가 작업(WNetAddConnection, WNetAddConnection2 및 WNetAddConnection3)이 MPR(다중 공급자 라우터)에 의해 시도되기 전과 후에 호출됩니다.
AddCredentialsA ANSI(AddCredentialsA)는 자격 증명 목록에 자격 증명을 추가합니다.
AddCredentialsW AddCredentialsW(유니코드)는 자격 증명 목록에 자격 증명을 추가합니다.
AddDelegatedPolicyUser 텍스트 형식의 지정된 SID(보안 식별자)를 위임된 정책 사용자 역할을 하는 보안 주체 목록에 추가합니다. (IAzApplication.AddDelegatedPolicyUser)
AddDelegatedPolicyUser 텍스트 형식의 지정된 SID(보안 식별자)를 위임된 정책 사용자 역할을 하는 보안 주체 목록에 추가합니다. (IAzAuthorizationStore.AddDelegatedPolicyUser)
AddDelegatedPolicyUserName 위임된 정책 사용자 역할을 하는 보안 주체 목록에 지정된 계정 이름을 추가합니다. (IAzApplication.AddDelegatedPolicyUserName)
AddDelegatedPolicyUserName 위임된 정책 사용자 역할을 하는 보안 주체 목록에 지정된 계정 이름을 추가합니다. (IAzAuthorizationStore.AddDelegatedPolicyUserName)
AddEnrollmentServer 인증서 등록 서버(CES) 액세스 자격 증명을 자격 증명 캐시에 저장합니다.
AddExtensionsToRequest

AddExtensionsToRequest 메서드는 인증서 요청에 확장을 추가합니다. 이 메서드는 IEnroll 인터페이스에서 처음 정의되었습니다.
addExtensionToRequest ICEnroll4::addExtensionToRequest 메서드는 요청에 확장을 추가합니다.
addExtensionToRequestWStr 요청에 확장을 추가합니다.
AddFromCsp 특정 암호화 공급자가 지원하는 암호화 알고리즘을 식별하여 컬렉션에 개체를 추가합니다.
AddInterface BizRule(비즈니스 규칙) 스크립트에서 호출할 수 있는 IDispatch 인터페이스 목록에 지정된 인터페이스를 추가합니다.
AddInterfaces 비즈니스 규칙(BizRule) 스크립트에서 호출할 수 있는 IDispatch 인터페이스 목록에 지정된 인터페이스를 추가합니다.
AddMandatoryAce 지정된 SACL(시스템 액세스 제어 목록)에 SYSTEM_MANDATORY_LABEL_ACE ACE(액세스 제어 항목)를 추가합니다.
AddMember 텍스트 형식의 지정된 SID(보안 식별자)를 애플리케이션 그룹에 속한 계정 목록에 추가합니다.
AddMember 텍스트 형식의 지정된 SID(보안 식별자)를 역할에 속한 Windows 계정 목록에 추가합니다.
AddMemberName 애플리케이션 그룹에 속한 계정 목록에 지정된 계정 이름을 추가합니다.
AddMemberName 역할에 속한 계정 목록에 지정된 계정 이름을 추가합니다.
addNameValuePairToRequest

인증되지 않은 이름-값 문자열 쌍을 요청에 추가합니다. 이 메서드는 ICEnroll4 인터페이스에서 처음 정의되었습니다.
addNameValuePairToRequestWStr 인증되지 않은 이름-값 문자열 쌍을 요청에 추가합니다.
addNameValuePairToSignature 특성의 인증된 이름-값 쌍을 요청에 추가합니다. 이름-값 쌍의 의미를 해석하는 것은 CA(인증 기관)의 책임입니다.
AddNameValuePairToSignatureWStr 특성의 인증된 이름-값 쌍을 요청에 추가합니다. CA(인증 기관)는 이름-값 쌍의 의미를 해석합니다.
AddNonMember 텍스트 형식의 지정된 SID(보안 식별자)를 애플리케이션 그룹의 멤버 자격이 거부된 계정 목록에 추가합니다.
AddNonMemberName 애플리케이션 그룹의 멤버 자격이 거부된 계정 목록에 지정된 계정 이름을 추가합니다.
AddOperation 지정된 이름의 IAzOperation 개체를 역할에 추가합니다.
AddOperation 지정된 이름의 IAzOperation 개체를 작업에 추가합니다.
AddParameter 비즈니스 규칙(BizRule) 스크립트에 사용할 수 있는 매개 변수 목록에 매개 변수를 추가합니다.
AddParameters 비즈니스 규칙(BizRule) 스크립트에 사용할 수 있는 매개 변수 목록에 매개 변수를 추가합니다.
AddPolicyAdministrator 텍스트 형식의 지정된 SID(보안 식별자)를 정책 관리자 역할을 하는 보안 주체 목록에 추가합니다. (IAzApplication.AddPolicyAdministrator)
AddPolicyAdministrator

텍스트 형식의 지정된 SID(보안 식별자)를 정책 관리자 역할을 하는 보안 주체 목록에 추가합니다. (IAzAuthorizationStore.AddPolicyAdministrator)
AddPolicyAdministrator IAzScope의 AddPolicyAdministrator 메서드는 텍스트 형식의 지정된 보안 식별자를 정책 관리자 역할을 하는 보안 주체 목록에 추가합니다.
AddPolicyAdministratorName 정책 관리자 역할을 하는 보안 주체 목록에 지정된 계정 이름을 추가합니다. (IAzApplication.AddPolicyAdministratorName)
AddPolicyAdministratorName 정책 관리자 역할을 하는 보안 주체 목록에 지정된 계정 이름을 추가합니다. (IAzAuthorizationStore.AddPolicyAdministratorName)
AddPolicyAdministratorName IAzScope의 AddPolicyAdministratorName 메서드는 정책 관리자 역할을 하는 보안 주체 목록에 지정된 계정 이름을 추가합니다.
AddPolicyReader 텍스트 형식의 지정된 SID(보안 식별자)를 정책 판독기 역할을 하는 보안 주체 목록에 추가합니다. (IAzApplication.AddPolicyReader)
AddPolicyReader 텍스트 형식의 지정된 SID(보안 식별자)를 정책 판독기 역할을 하는 보안 주체 목록에 추가합니다. (IAzAuthorizationStore.AddPolicyReader)
AddPolicyReader IAzScope의 AddPolicyReader 메서드는 텍스트 형식의 지정된 보안 식별자를 정책 판독기 역할을 하는 보안 주체 목록에 추가합니다.
AddPolicyReaderName 정책 판독기 역할을 하는 보안 주체 목록에 지정된 계정 이름을 추가합니다. (IAzApplication.AddPolicyReaderName)
AddPolicyReaderName 정책 판독기 역할을 하는 보안 주체 목록에 지정된 계정 이름을 추가합니다. (IAzAuthorizationStore.AddPolicyReaderName)
AddPolicyReaderName

IAzScope의 AddPolicyReaderName 메서드는 정책 판독기 역할을 하는 보안 주체 목록에 지정된 계정 이름을 추가합니다.
AddPolicyServer CEP(인증서 등록 정책) 서버를 등록하고 CEP 액세스 자격 증명을 자격 증명 캐시에 저장합니다.
AddPropertyItem 지정된 보안 주체 목록에 지정된 보안 주체를 추가합니다. (IAzApplication.AddPropertyItem)
AddPropertyItem 지정된 엔터티를 지정된 목록에 추가합니다. (IAzApplicationGroup.AddPropertyItem)
AddPropertyItem 지정된 보안 주체 목록에 지정된 보안 주체를 추가합니다. (IAzAuthorizationStore.AddPropertyItem)
AddPropertyItem 지정된 엔터티를 지정된 목록에 추가합니다. (IAzRole.AddPropertyItem)
AddPropertyItem 지정된 보안 주체 목록에 지정된 보안 주체를 추가합니다. (IAzScope.AddPropertyItem)
AddPropertyItem 지정된 엔터티를 지정된 목록에 추가합니다. (IAzTask.AddPropertyItem)
AddRange 컬렉션에 ICryptAttribute 개체의 범위를 추가합니다. 특성은 다른 ICryptAttributes 컬렉션에 포함됩니다.
AddRange 컬렉션에 IObjectId 개체의 범위를 추가합니다.
AddRange 컬렉션에 IX509Extension 개체 범위를 추가합니다.
AddResourceAttributeAce SACL(시스템 액세스 제어 목록)의 끝에 ACE(SYSTEM_RESOURCE_ATTRIBUTE_ACEaccess 컨트롤 항목)를 추가합니다.
AddRoleDefinition

지정된 IAzRoleDefinition 개체를 이 IAzRoleAssignment 개체에 추가합니다.
AddRoleDefinition 지정된 IAzRoleDefinition 개체를 이 IAzRoleDefinition 개체에 추가합니다.
AddRoles 기존 IAzRole 개체의 지정된 배열을 클라이언트 컨텍스트에 추가합니다.
AddScopedPolicyIDAce SACL(시스템 액세스 제어 목록) 끝에 ACE(SYSTEM_SCOPED_POLICY_ID_ACEaccess 컨트롤 항목)를 추가합니다.
AddSecurityPackageA Microsoft Negotiate에서 지원하는 공급자 목록에 보안 지원 공급자를 추가합니다. (ANSI)
AddSecurityPackageW Microsoft Negotiate에서 지원하는 공급자 목록에 보안 지원 공급자를 추가합니다. (유니코드)
AddStringSids 클라이언트 컨텍스트에 SID(보안 식별자)의 문자열 표현 배열을 추가합니다.
AddTask 지정된 이름의 IAzTask 개체를 역할에 추가합니다.
AddTask 지정된 이름의 IAzTask 개체를 작업에 추가합니다.
AddToCache 레지스트리에서 지정된 ID를 캐시합니다.
AdjustTokenGroups 지정된 액세스 토큰에 이미 있는 그룹을 사용하거나 사용하지 않도록 설정합니다. 액세스 토큰에서 그룹을 사용하거나 사용하지 않도록 설정하려면 TOKEN_ADJUST_GROUPS 액세스가 필요합니다.
AdjustTokenPrivileges 지정된 액세스 토큰에서 권한을 사용하거나 사용하지 않도록 설정합니다. 액세스 토큰에서 권한을 사용하거나 사용하지 않도록 설정하려면 TOKEN_ADJUST_PRIVILEGES 액세스가 필요합니다.

조언하다 호출 애플리케이션에서 애플리케이션에 알림을 받을 ID 이벤트 목록을 지정할 수 있습니다.
AllocateAndInitializeSid 최대 8개의 하위 인증을 사용하여 SID(보안 식별자)를 할당하고 초기화합니다.
AllocateLocallyUniqueId LUID(로컬 고유 식별자)를 할당합니다.
AppendText Text 속성에 포함된 상태 정보에 문자열을 추가합니다.
ApplyControlToken 보안 컨텍스트에 컨트롤 토큰을 적용하는 방법을 제공합니다.
AreAllAccessesGranted 요청된 액세스 권한 집합이 부여되었는지 여부를 확인합니다. 액세스 권한은 액세스 마스크에서 비트 플래그로 표시됩니다.
AreAnyAccessesGranted 요청된 액세스 권한 집합이 부여되었는지 여부를 테스트합니다. 액세스 권한은 액세스 마스크에서 비트 플래그로 표시됩니다.
AssociateIdentity ID를 로컬 사용자 계정과 연결합니다.
AuditComputeEffectivePolicyBySid 지정된 보안 주체에 대한 하나 이상의 하위 범주에 대한 유효 감사 정책을 계산합니다. 이 함수는 시스템 감사 정책을 사용자별 정책과 결합하여 효과적인 감사 정책을 계산합니다.
AuditComputeEffectivePolicyByToken 지정된 토큰과 연결된 보안 주체에 대한 하나 이상의 하위 범주에 대한 유효 감사 정책을 계산합니다. 이 함수는 시스템 감사 정책을 사용자별 정책과 결합하여 효과적인 감사 정책을 계산합니다.
AuditEnumerateCategories 사용 가능한 감사 정책 범주를 열거합니다.

AuditEnumeratePerUserPolicy 사용자별 감사 정책이 지정된 사용자를 열거합니다.
AuditEnumerateSubCategories 사용 가능한 감사 정책 하위 범주를 열거합니다.
AuditFree 지정된 버퍼에 대한 감사 함수에 의해 할당된 메모리를 해제합니다.
AuditLookupCategoryGuidFromCategoryId 지정된 감사 정책 범주를 나타내는 GUID 구조를 검색합니다.
AuditLookupCategoryIdFromCategoryGuid 지정된 감사 정책 범주를 나타내는 POLICY_AUDIT_EVENT_TYPE 열거형의 요소를 검색합니다.
AuditLookupCategoryNameA 지정된 감사 정책 범주의 표시 이름을 검색합니다. (ANSI)
AuditLookupCategoryNameW 지정된 감사 정책 범주의 표시 이름을 검색합니다. (유니코드)
AuditLookupSubCategoryNameA 지정된 감사 정책 하위 범주의 표시 이름을 검색합니다. (ANSI)
AuditLookupSubCategoryNameW 지정된 감사 정책 하위 범주의 표시 이름을 검색합니다. (유니코드)
AuditQueryGlobalSaclA 감사 메시지에 대한 액세스를 위임하는 SACL(전역 시스템 액세스 제어 목록)을 검색합니다. (ANSI)
AuditQueryGlobalSaclW 감사 메시지에 대한 액세스를 위임하는 SACL(전역 시스템 액세스 제어 목록)을 검색합니다. (유니코드)
AuditQueryPerUserPolicy 지정된 보안 주체에 대한 하나 이상의 감사 정책 하위 범주에서 사용자별 감사 정책을 검색합니다.
AuditQuerySecurity

감사 정책에 대한 액세스를 위임하는 보안 설명자를 검색합니다.
AuditQuerySystemPolicy 하나 이상의 감사 정책 하위 범주에 대한 시스템 감사 정책을 검색합니다.
AuditSetGlobalSaclA 감사 메시지에 대한 액세스를 위임하는 SACL(전역 시스템 액세스 제어 목록)을 설정합니다. (ANSI)
AuditSetGlobalSaclW 감사 메시지에 대한 액세스를 위임하는 SACL(전역 시스템 액세스 제어 목록)을 설정합니다. (유니코드)
AuditSetPerUserPolicy 지정된 보안 주체에 대한 하나 이상의 감사 하위 범주에서 사용자별 감사 정책을 설정합니다.
AuditSetSecurity 감사 정책에 대한 액세스를 위임하는 보안 설명자를 설정합니다.
AuditSetSystemPolicy 하나 이상의 감사 정책 하위 범주에 대한 시스템 감사 정책을 설정합니다.
AuthzAccessCheck 지정된 보안 설명자 집합에 대해 클라이언트에 부여할 수 있는 액세스 비트를 결정합니다.
AuthzAddSidsToContext 기존 컨텍스트의 복사본을 만들고 지정된 SID(보안 식별자) 및 제한된 SID 집합을 추가합니다.
AuthzCachedAccessCheck 이전 AuthzAccessCheck 호출에서 정적으로 부여된 비트를 포함하는 캐시된 핸들을 기반으로 빠른 액세스 검사를 수행합니다.
AuthzEnumerateSecurityEventSources 기본적으로 설치되지 않은 등록된 보안 이벤트 원본을 검색합니다.
AuthzFreeAuditEvent AuthzInitializeObjectAccessAuditEvent 함수에 의해 할당된 구조를 해제합니다.
AuthzFreeCentralAccessPolicyCache CAP 캐시의 할당을 취소할 수 있도록 CAP 캐시 참조 수를 1씩 줄입니다.

AuthzFreeContext 클라이언트 컨텍스트와 연결된 모든 구조 및 메모리를 해제합니다. 클라이언트에 대한 핸들 목록은 이 호출에서 해제됩니다.
AuthzFreeHandle 핸들 목록에서 핸들을 찾아 삭제합니다.
AuthzFreeResourceManager Resource Manager 개체를 해제합니다.
AuthzGetInformationFromContext Authz 컨텍스트에 대한 정보를 반환합니다.
AuthzInitializeCompoundContext 지정된 사용자 및 디바이스 보안 컨텍스트에서 사용자 모드 컨텍스트를 만듭니다.
AuthzInitializeContextFromAuthzContext 기존 클라이언트 컨텍스트를 기반으로 새 클라이언트 컨텍스트를 만듭니다.
AuthzInitializeContextFromSid SID(사용자 보안 식별자)에서 사용자 모드 클라이언트 컨텍스트를 만듭니다.
AuthzInitializeContextFromToken 커널 토큰에서 클라이언트 권한 부여 컨텍스트를 초기화합니다. 커널 토큰은 TOKEN_QUERY 위해 열렸어야 합니다.
AuthzInitializeObjectAccessAuditEvent 개체에 대한 감사를 초기화합니다.
AuthzInitializeObjectAccessAuditEvent2 AuthzAccessCheck 함수에 사용할 AUTHZ_AUDIT_EVENT_HANDLE 핸들을 할당하고 초기화합니다.
AuthzInitializeRemoteResourceManager 원격 리소스 관리자를 할당하고 초기화합니다. 호출자는 결과 핸들을 사용하여 서버에 구성된 리소스 관리자의 원격 인스턴스에 대한 RPC 호출을 수행할 수 있습니다.
AuthzInitializeResourceManager

Authz를 사용하여 클라이언트가 다양한 리소스에 액세스할 수 있는지 확인합니다.
AuthzInitializeResourceManagerEx 리소스 관리자 구조를 할당하고 초기화합니다.
AuthzInstallSecurityEventSource 지정된 소스를 보안 이벤트 원본으로 설치합니다.
AuthzModifyClaims Authz 클라이언트 컨텍스트에서 사용자 및 디바이스 클레임을 추가, 삭제 또는 수정합니다.
AuthzModifySecurityAttributes 지정된 클라이언트 컨텍스트에서 보안 특성 정보를 수정합니다.
AuthzModifySids Authz 클라이언트 컨텍스트에서 사용자 및 디바이스 그룹을 추가, 삭제 또는 수정합니다.
AuthzOpenObjectAudit 지정된 보안 설명자의 SACL(시스템 액세스 제어 목록)을 읽고 해당 SACL에서 지정한 적절한 감사를 생성합니다.
AuthzRegisterCapChangeNotification CAP 업데이트 알림 콜백을 등록합니다.
AuthzRegisterSecurityEventSource LSA(로컬 보안 기관)에 보안 이벤트 원본을 등록합니다.
AuthzReportSecurityEvent 등록된 보안 이벤트 원본에 대한 보안 감사를 생성합니다.
AuthzReportSecurityEventFromParams 지정된 감사 매개 변수 배열을 사용하여 등록된 보안 이벤트 원본에 대한 보안 감사를 생성합니다.
AuthzSetAppContainerInformation 현재 Authz 컨텍스트에서 앱 컨테이너 및 기능 정보를 설정합니다.
AuthzUninstallSecurityEventSource 유효한 보안 이벤트 원본 목록에서 지정된 소스를 제거합니다.

AuthzUnregisterCapChangeNotification 이전에 등록된 CAP 업데이트 알림 콜백을 제거합니다.
AuthzUnregisterSecurityEventSource LSA(로컬 보안 기관)를 사용하여 보안 이벤트 원본을 등록 취소합니다.
BCRYPT_INIT_AUTH_MODE_INFO BCryptEncrypt 및 BCryptDecrypt 함수 호출에 사용할 BCRYPT_AUTHENTICATED_CIPHER_MODE_INFO 구조를 초기화합니다.
BCryptAddContextFunction 기존 CNG 컨텍스트에서 지원하는 함수 목록에 암호화 함수를 추가합니다.
BCryptCloseAlgorithmProvider 알고리즘 공급자를 닫습니다.
BCryptConfigureContext 기존 CNG 컨텍스트에 대한 구성 정보를 설정합니다.
BCryptConfigureContextFunction 기존 CNG 컨텍스트의 암호화 함수에 대한 구성 정보를 설정합니다.
BCryptCreateContext 새 CNG 구성 컨텍스트를 만듭니다.
BCryptCreateHash 해시 또는 MAC(메시지 인증 코드) 개체를 만들기 위해 호출됩니다.
BCryptCreateMultiHash BCryptCreateMultiHash 함수는 여러 해시 작업의 병렬 계산을 허용하는 다중 해시 상태를 만듭니다.
BCryptDecrypt 데이터 블록을 해독합니다.
BCryptDeleteContext 기존 CNG 구성 컨텍스트를 삭제합니다.

BCryptDeriveKey 비밀 핸들에서 키를 파생합니다. (BCryptDeriveKey)
BCryptDeriveKeyCapi 해시 값에서 키를 파생합니다.
BCryptDeriveKeyPBKDF2 RFC 2898에 정의된 대로 PBKDF2 키 파생 알고리즘을 사용하여 해시 값에서 키를 파생합니다.
BCryptDestroyHash 해시 또는 MAC(메시지 인증 코드) 개체를 제거합니다.
BCryptDestroyKey 키를 삭제합니다.
BCryptDestroySecret BCryptSecretAgreement 함수를 사용하여 만든 비밀 계약 핸들을 삭제합니다.
BCryptDuplicateHash 기존 해시 또는 MAC(메시지 인증 코드) 개체를 복제합니다.
BCryptDuplicateKey 대칭 키의 중복을 만듭니다.
BCryptEncrypt 데이터 블록을 암호화합니다. (BCryptEncrypt)
BCryptEnumAlgorithms 등록된 알고리즘 식별자 목록을 가져옵니다.
BCryptEnumContextFunctionProviders 지정된 구성 테이블의 컨텍스트에 대한 암호화 함수에 대한 공급자를 가져옵니다.
BCryptEnumContextFunctions 지정된 구성 테이블의 컨텍스트에 대한 암호화 함수를 가져옵니다.
BCryptEnumContexts

지정된 구성 테이블에 있는 컨텍스트의 식별자를 가져옵니다.
BCryptEnumProviders 지정된 알고리즘을 지원하는 모든 CNG 공급자를 가져옵니다.
BCryptEnumRegisteredProviders 등록된 공급자에 대한 정보를 검색합니다.
BCryptExportKey 나중에 사용할 수 있도록 유지할 수 있는 메모리 BLOB에 키를 내보냅니다.
BCryptFinalizeKeyPair 퍼블릭/프라이빗 키 쌍을 완료합니다.
BCryptFinishHash BCryptHashData에 대한 이전 호출에서 누적된 데이터에 대한 해시 또는 MAC(메시지 인증 코드) 값을 검색합니다.
BCryptFreeBuffer CNG 함수 중 하나에서 할당한 메모리를 해제하는 데 사용됩니다.
BCryptGenerateKeyPair 빈 공개/프라이빗 키 쌍을 만듭니다.
BCryptGenerateSymmetricKey 제공된 키에서 대칭 키 암호화 알고리즘과 함께 사용할 키 개체를 만듭니다.
BCryptGenRandom 난수를 생성합니다.
BCryptGetFipsAlgorithmmode FIPS(Federal Information Processing Standard) 규정 준수를 사용할 수 있는지 여부를 결정합니다.
BCryptGetProperty CNG 개체의 명명된 속성 값을 검색합니다.
BCryptHash

단일 해시 또는 MAC 계산을 수행합니다. BCryptCreateHash, BCryptHashData, BCryptFinishHash 및 BCryptDestroyHash에 대한 호출을 래핑하는 편리한 함수입니다.
BCryptHashData 데이터 버퍼에서 단방향 해시 또는 MAC(메시지 인증 코드)를 수행합니다.
BCryptImportKey 키 BLOB에서 대칭 키를 가져옵니다.
BCryptImportKeyPair 키 BLOB에서 퍼블릭/프라이빗 키 쌍을 가져옵니다.
BCryptKeyDerivation 비밀 계약을 요구하지 않고 키를 파생합니다.
BCryptOpenAlgorithmProvider CNG 공급자를 로드하고 초기화합니다.
BCryptProcessMultiOperations BCryptProcessMultiOperations 함수는 다중 개체 상태에서 연산 시퀀스를 처리합니다.
BCryptQueryContextConfiguration 지정된 CNG 컨텍스트에 대한 현재 구성을 검색합니다.
BCryptQueryContextFunctionConfiguration 기존 CNG 컨텍스트에 대한 암호화 함수 구성 정보를 가져옵니다.
BCryptQueryContextFunctionProperty 기존 CNG 컨텍스트에서 암호화 함수의 명명된 속성 값을 가져옵니다.
BCryptQueryProviderRegistration CNG 공급자에 대한 정보를 검색합니다.
BCryptRegisterConfigChangeNotify 사용자 모드 CNG 구성 변경 이벤트 처리기를 만듭니다.
BCryptRemoveContextFunction 기존 CNG 컨텍스트에서 지원되는 함수 목록에서 암호화 함수를 제거합니다.

BCryptResolveProviders 지정된 조건을 충족하는 모든 공급자의 컬렉션을 가져옵니다.
BCryptSecretAgreement 프라이빗 및 공개 키에서 비밀 계약 값을 만듭니다. (BCryptSecretAgreement)
BCryptSetContextFunctionProperty 기존 CNG 컨텍스트에서 암호화 함수의 명명된 속성 값을 설정합니다.
BCryptSetProperty CNG 개체의 명명된 속성 값을 설정합니다.
BCryptSignHash 해시 값의 서명을 만듭니다. (BCryptSignHash)
BCryptUnregisterConfigChangeNotify BCryptRegisterConfigChangeNotify(HANDLE*) 함수를 사용하여 만든 사용자 모드 CNG 구성 변경 이벤트 처리기를 제거합니다.
BCryptVerifySignature 지정된 서명이 지정된 해시와 일치하는지 확인합니다. (BCryptVerifySignature)
binaryBlobToString 이진 데이터 BLOB을 문자열로 변환합니다. 이 메서드는 CryptBinaryToString 함수를 사용하여 변환을 수행합니다. 이 메서드는 IEnroll4 인터페이스에서 처음 정의되었습니다.
binaryToString 이진 데이터 BLOB을 문자열로 변환합니다. 이 메서드는 ICEnroll4 인터페이스에서 처음 정의되었습니다.
BizruleGroupSupported 이 IAzAuthorizationStore3 개체가 BizRule(비즈니스 규칙) 스크립트를 사용하는 애플리케이션 그룹을 지원하는지 여부를 지정하는 부울 값을 반환합니다.
BuildExplicitAccessWithNameA 호출자가 지정한 데이터를 사용하여 EXPLICIT_ACCESS 구조를 초기화합니다. 트러스티는 이름 문자열로 식별됩니다. (ANSI)

BuildExplicitAccessWithNameW 호출자가 지정한 데이터를 사용하여 EXPLICIT_ACCESS 구조를 초기화합니다. 트러스티는 이름 문자열로 식별됩니다. (유니코드)
BuildSecurityDescriptorA 새 보안 설명자를 할당하고 초기화합니다. (ANSI)
BuildSecurityDescriptorW 새 보안 설명자를 할당하고 초기화합니다. (유니코드)
BuildTrusteeWithNameA TRUSTEE 구조를 초기화합니다. 호출자는 트러스티 이름을 지정합니다. 이 함수는 구조체의 다른 멤버를 기본값으로 설정합니다. (ANSI)
BuildTrusteeWithNameW TRUSTEE 구조를 초기화합니다. 호출자는 트러스티 이름을 지정합니다. 이 함수는 구조체의 다른 멤버를 기본값으로 설정합니다. (유니코드)
BuildTrusteeWithObjectsAndNameA ACE(개체별 액세스 제어 항목) 정보를 사용하여 TRUSTEE 구조를 초기화하고 구조체의 나머지 멤버를 기본값으로 초기화합니다. 호출자는 트러스티의 이름도 지정합니다. (ANSI)
BuildTrusteeWithObjectsAndNameW ACE(개체별 액세스 제어 항목) 정보를 사용하여 TRUSTEE 구조를 초기화하고 구조체의 나머지 멤버를 기본값으로 초기화합니다. 호출자는 트러스티의 이름도 지정합니다. (유니코드)
BuildTrusteeWithObjectsAndSidA ACE(개체별 액세스 제어 항목) 정보를 사용하여 TRUSTEE 구조를 초기화하고 구조체의 나머지 멤버를 기본값으로 초기화합니다. (ANSI)
BuildTrusteeWithObjectsAndSidW ACE(개체별 액세스 제어 항목) 정보를 사용하여 TRUSTEE 구조를 초기화하고 구조체의 나머지 멤버를 기본값으로 초기화합니다. (유니코드)
BuildTrusteeWithSidA TRUSTEE 구조를 초기화합니다. 호출자는 트러스티의 SID(보안 식별자)를 지정합니다. 이 함수는 구조체의 다른 멤버를 기본값으로 설정하고 SID와 연결된 이름을 조회하지 않습니다. (ANSI)

BuildTrusteeWithSidW TRUSTEE 구조를 초기화합니다. 호출자는 트러스티의 SID(보안 식별자)를 지정합니다. 이 함수는 구조체의 다른 멤버를 기본값으로 설정하고 SID와 연결된 이름을 조회하지 않습니다. (유니코드)
CAImportPFX CA(인증 기관) 인증서와 연결된 프라이빗 키를 로컬 컴퓨터 저장소로 가져옵니다.
CancelConnectNotify 각 연결 취소 작업 전후에 CancelConnectNotify를 호출합니다(WNetCancelConnection 및 WNetCancelConnection2).
CertAddCertificateContextToStore 인증서 저장소에 인증서 컨텍스트를 추가합니다.
CertAddCertificateLinkToStore 인증서 저장소의 링크를 다른 저장소의 인증서 컨텍스트에 추가합니다.
CertAddCRLContextToStore 지정된 인증서 저장소에 CRL(인증서 해지 목록) 컨텍스트를 추가합니다.
CertAddCRLLinkToStore 저장소의 링크를 다른 저장소의 CRL(인증서 해지 목록) 컨텍스트에 추가합니다.
CertAddCTLContextToStore 인증서 저장소에 CTL(인증서 신뢰 목록) 컨텍스트를 추가합니다.
CertAddCTLLinkToStore CertAddCTLLinkToStore 함수는 저장소의 링크를 다른 저장소의 CTL(인증서 신뢰 목록) 컨텍스트에 추가합니다. 이 함수는 CTL 컨텍스트의 중복을 만들고 추가하는 대신 원래 CTL 컨텍스트에 대한 링크를 추가합니다.
CertAddEncodedCertificateToStore 인코딩된 인증서에서 인증서 컨텍스트를 만들고 인증서 저장소에 추가합니다.
CertAddEncodedCertificateToSystemStoreA 지정된 시스템 저장소를 열고 인코딩된 인증서를 추가합니다. (ANSI)

CertAddEncodedCertificateToSystemStoreW 지정된 시스템 저장소를 열고 인코딩된 인증서를 추가합니다. (유니코드)
CertAddEncodedCRLToStore 인코딩된 CRL에서 CRL(인증서 해지 목록) 컨텍스트를 만들고 인증서 저장소에 추가합니다.
CertAddEncodedCTLToStore 인코딩된 CTL에서 CTL(인증서 신뢰 목록) 컨텍스트를 만들고 인증서 저장소에 추가합니다.
CertAddEnhancedKeyUsageIdentifier CertAddEnhancedKeyUsageIdentifier 함수는 인증서의 EKU(고급 키 사용) 확장 속성에 OID(사용 식별자 개체 식별자)를 추가합니다.
CertAddRefServerOcspResponse HCERT_SERVER_OCSP_RESPONSE 핸들에 대한 참조 수를 증분합니다.
CertAddRefServerOcspResponseContext CERT_SERVER_OCSP_RESPONSE_CONTEXT 구조체에 대한 참조 수를 증분합니다.
CertAddSerializedElementToStore 직렬화된 인증서, CRL(인증서 해지 목록) 또는 CTL(인증서 신뢰 목록) 요소를 저장소에 추가합니다.
CertAddStoreToCollection CertAddStoreToCollection 함수는 컬렉션 인증서 저장소에 형제 인증서 저장소를 추가합니다.
CertAlgIdToOID cryptoAPI 알고리즘 식별자(ALG_ID)를 ASN.1(Abstract Syntax Notation One) OID(개체 식별자) 문자열로 변환합니다.
CertCloseServerOcspResponse OCSP(온라인 인증서 상태 프로토콜) 서버 응답 핸들을 닫습니다.
CertCloseStore 인증서 저장소 핸들을 닫고 저장소의 참조 수를 줄입니다.
CertCompareCertificate 인증서의 발급자 이름과 일련 번호를 비교하여 두 인증서가 동일한지 여부를 결정합니다.

CertCompareCertificateName CertCompareCertificateName 함수는 두 인증서 CERT_NAME_BLOB 구조를 비교하여 동일한지 여부를 확인합니다. CERT_NAME_BLOB 구조는 인증서의 주체 및 발급자용으로 사용됩니다.
CertCompareIntegerBlob CertCompareIntegerBlob 함수는 두 개의 정수 BLOB을 비교하여 같은 숫자 값을 나타내는지 여부를 확인합니다.
CertComparePublicKeyInfo CertComparePublicKeyInfo 함수는 두 개의 인코딩된 공용 키를 비교하여 동일한지 여부를 확인합니다.
CertControlStore 사용 중인 캐시된 저장소의 콘텐츠와 스토리지에 유지되는 해당 저장소의 콘텐츠 간에 차이가 있는 경우 애플리케이션에 알림을 받을 수 있습니다.
CertCreateCertificateChainEngine CertCreateCertificateChainEngine 함수는 애플리케이션에 대한 기본이 아닌 새로운 체인 엔진을 만듭니다.
CertCreateCertificateContext 인코딩된 인증서에서 인증서 컨텍스트를 만듭니다. 만든 컨텍스트는 인증서 저장소에 유지되지 않습니다. 함수는 생성된 컨텍스트 내에서 인코딩된 인증서의 복사본을 만듭니다.
CertCreateContext 인코딩된 바이트에서 지정된 컨텍스트를 만듭니다. 만든 컨텍스트에는 확장 속성이 포함되지 않습니다.
CertCreateCRLContext CertCreateCRLContext 함수는 인코딩된 CRL에서 CRL(인증서 해지 목록) 컨텍스트를 만듭니다. 만든 컨텍스트는 인증서 저장소에 유지되지 않습니다. 생성된 컨텍스트 내에서 인코딩된 CRL의 복사본을 만듭니다.
CertCreateCTLContext CertCreateCTLContext 함수는 인코딩된 CTL에서 CTL(인증서 신뢰 목록) 컨텍스트를 만듭니다. 만든 컨텍스트는 인증서 저장소에 유지되지 않습니다. 함수는 생성된 컨텍스트 내에서 인코딩된 CTL의 복사본을 만듭니다.
CertCreateCTLEntryFromCertificateContextProperties CertCreateCTLEntryFromCertificateContextProperties 함수는 해당 특성이 인증서 컨텍스트의 속성인 CTL(인증서 신뢰 목록) 항목을 만듭니다. CTL 항목의 SubjectIdentifier는 인증서의 SHA1 해시입니다.

CertCreateSelfSignCertificate 자체 서명된 인증서를 빌드하고 인증서를 나타내는 CERT_CONTEXT 구조체에 대한 포인터를 반환합니다.
CertDeleteCertificateFromStore CertDeleteCertificateFromStore 함수는 인증서 저장소에서 지정된 인증서 컨텍스트를 삭제합니다.
CertDeleteCRLFromStore CertDeleteCRLFromStore 함수는 인증서 저장소에서 지정된 CRL(인증서 해지 목록) 컨텍스트를 삭제합니다.
CertDeleteCTLFromStore CertDeleteCTLFromStore 함수는 인증서 저장소에서 지정된 CTL(인증서 신뢰 목록) 컨텍스트를 삭제합니다.
CertDuplicateCertificateChain CertDuplicateCertificateChain 함수는 체인의 참조 수를 증가시켜 인증서 체인에 대한 포인터를 복제합니다.
CertDuplicateCertificateContext 참조 횟수를 증가시켜 인증서 컨텍스트를 복제합니다.
CertDuplicateCRLContext CertDuplicateCRLContext 함수는 참조 횟수를 증가시켜 CRL(인증서 해지 목록) 컨텍스트를 복제합니다.
CertDuplicateCTLContext CertDuplicateCTLContext 함수는 참조 수를 증가시켜 CTL(인증서 신뢰 목록) 컨텍스트를 복제합니다.
CertDuplicateStore 저장소의 참조 수를 증가시켜 저장소 핸들을 복제합니다.
CertEnumCertificateContextProperties CertEnumCertificateContextProperties 함수는 인증서 컨텍스트와 연결된 첫 번째 또는 다음 확장 속성을 검색합니다.
CertEnumCertificatesInStore 인증서 저장소에서 첫 번째 또는 다음 인증서를 검색합니다. 루프에서 사용되는 이 함수는 인증서 저장소의 모든 인증서를 순서대로 검색할 수 있습니다.

CertEnumCRLContextProperties CertEnumCRLContextProperties 함수는 CRL(인증서 해지 목록) 컨텍스트와 연결된 첫 번째 또는 다음 확장 속성을 검색합니다.
CertEnumCRLsInStore CertEnumCRLsInStore 함수는 인증서 저장소의 첫 번째 또는 다음 CRL(인증서 해지 목록) 컨텍스트를 검색합니다. 루프에서 사용되는 이 함수는 인증서 저장소의 모든 CRL 컨텍스트를 순서대로 검색할 수 있습니다.
CertEnumCTLContextProperties CertEnumCTLContextProperties 함수는 CTL(인증서 신뢰 목록) 컨텍스트와 연결된 첫 번째 또는 다음 확장 속성을 검색합니다. 루프에서 사용되는 이 함수는 CTL 컨텍스트와 연결된 모든 확장 속성을 순서대로 검색할 수 있습니다.
CertEnumCTLsInStore CertEnumCTLsInStore 함수는 인증서 저장소의 첫 번째 또는 다음 CTL(인증서 신뢰 목록) 컨텍스트를 검색합니다. 루프에서 사용되는 이 함수는 인증서 저장소의 모든 CTL 컨텍스트를 순서대로 검색할 수 있습니다.
CertEnumPhysicalStore CertEnumPhysicalStore 함수는 컴퓨터의 실제 저장소를 검색합니다. 함수는 찾은 각 실제 저장소에 대해 제공된 콜백 함수를 호출합니다.
CertEnumSubjectInSortedCTL 정렬된 CTL(인증서 신뢰 목록)에서 첫 번째 또는 다음 TrustedSubject를 검색합니다.
CertEnumSystemStore CertEnumSystemStore 함수는 사용 가능한 시스템 저장소를 검색합니다. 함수는 찾은 각 시스템 저장소에 대해 제공된 콜백 함수를 호출합니다.
CertEnumSystemStoreLocation CertEnumSystemStoreLocation 함수는 모든 시스템 저장소 위치를 검색합니다. 함수는 찾은 각 시스템 저장소 위치에 대해 제공된 콜백 함수를 호출합니다.
CertFindAttribute CertFindAttribute 함수는 OID(개체 식별자)로 식별되는 CRYPT_ATTRIBUTE 배열에서 첫 번째 특성을 찾습니다.
CertFindCertificateInCRL CertFindCertificateInCRL 함수는 CRL(인증서 해지 목록)에서 지정된 인증서를 검색합니다.

CertFindCertificateInStore dwFindType 및 연결된 pvFindPara에서 설정한 검색 조건과 일치하는 인증서 저장소에서 첫 번째 또는 다음 인증서 컨텍스트를 찾습니다.
CertFindChainInStore 지정된 조건을 충족하는 저장소에서 첫 번째 또는 다음 인증서를 찾습니다.
CertFindCRLInStore dwFindType 매개 변수 및 연결된 pvFindPara 매개 변수에 의해 설정된 검색 조건과 일치하는 인증서 저장소에서 첫 번째 또는 다음 CRL(인증서 해지 목록) 컨텍스트를 찾습니다.
CertFindCTLInStore dwFindType 및 연결된 pvFindPara에서 설정한 검색 조건과 일치하는 첫 번째 또는 다음 CTL(인증서 신뢰 목록) 컨텍스트를 찾습니다.
CertFindExtension CertFindExtension 함수는 OID(개체 식별자)로 식별되는 CERT_EXTENSION 배열에서 첫 번째 확장을 찾습니다.
CertFindRDNAttr CertFindRDNAttr 함수는 RDN(Relative Distinguished Names) 목록에서 OID(개체 식별자)로 식별된 첫 번째 RDN 특성을 찾습니다.
CertFindSubjectInCTL CertFindSubjectInCTL 함수는 CTL(인증서 신뢰 목록)에서 지정된 주체를 찾으려고 시도합니다.
CertFindSubjectInSortedCTL CertFindSubjectInSortedCTL 함수는 정렬된 CTL(인증서 신뢰 목록)에서 지정된 주체를 찾으려고 시도합니다.
CertFreeCertificateChain CertFreeCertificateChain 함수는 참조 횟수를 줄여 인증서 체인을 해제합니다. 참조 수가 0이 되면 체인에 할당된 메모리가 해제됩니다.
CertFreeCertificateChainEngine CertFreeCertificateChainEngine 함수는 인증서 신뢰 엔진을 해제합니다.
CertFreeCertificateChainList

포인터 배열을 체인 컨텍스트로 해제합니다.
CertFreeCertificateContext 참조 횟수를 줄여 인증서 컨텍스트를 해제합니다. 참조 수가 0으로 이동하면 CertFreeCertificateContext는 인증서 컨텍스트에서 사용하는 메모리를 해제합니다.
CertFreeCRLContext 참조 수를 줄여 CRL(인증서 해지 목록) 컨텍스트를 해제합니다.
CertFreeCTLContext 참조 수를 줄여 CTL(인증서 신뢰 목록) 컨텍스트를 해제합니다.
CertFreeServerOcspResponseContext CERT_SERVER_OCSP_RESPONSE_CONTEXT 구조체에 대한 참조 수를 줄입니다.
CertGetCertificateChain 최종 인증서에서 시작하여 가능한 경우 신뢰할 수 있는 루트 인증서로 돌아가는 인증서 체인 컨텍스트를 빌드합니다.
CertGetCertificateContextProperty 인증서 컨텍스트의 확장 속성에 포함된 정보를 검색합니다.
CertGetCRLContextProperty 지정된 CRL(인증서 해지 목록) 컨텍스트에 대한 확장 속성을 가져옵니다.
CertGetCRLFromStore 지정된 발급자의 인증서 저장소에서 첫 번째 또는 다음 CRL(인증서 해지 목록) 컨텍스트를 가져옵니다.
CertGetCTLContextProperty CTL(인증서 신뢰 목록) 컨텍스트의 확장 속성을 검색합니다.
CertGetEnhancedKeyUsage EKU(고급 키 사용) 확장 또는 인증서의 EKU 확장 속성에서 정보를 반환합니다.
CertGetIntendedKeyUsage 인증서에서 의도한 키 사용 바이트를 획득합니다.
CertGetIssuerCertificateFromStore

지정된 주체 인증서의 첫 번째 또는 다음 발급자의 인증서 저장소에서 인증서 컨텍스트를 검색합니다. 이 함수를 사용하는 대신 새 인증서 체인 확인 함수를 사용하는 것이 좋습니다.
CertGetNameStringA 인증서 CERT_CONTEXT 구조체에서 주체 또는 발급자 이름을 가져와 null로 끝나는 문자열로 변환합니다. (ANSI)
CertGetNameStringW 인증서 CERT_CONTEXT 구조체에서 주체 또는 발급자 이름을 가져와 null로 끝나는 문자열로 변환합니다. (유니코드)
CertGetPublicKeyLength CertGetPublicKeyLength 함수는 공개 키 BLOB에서 퍼블릭/프라이빗 키의 비트 길이를 획득합니다.
CertGetServerOcspResponseContext 지정된 핸들에 대한 비차단, 시간 유효한 OCSP(온라인 인증서 상태 프로토콜) 응답 컨텍스트를 검색합니다.
CertGetStoreProperty 저장소 속성을 검색합니다.
CertGetSubjectCertificateFromStore 인증서 저장소에서 발급자 및 일련 번호로 고유하게 식별되는 주체 인증서 컨텍스트를 반환합니다.
CertGetValidUsages 인증서 배열의 모든 인증서에 대해 유효한 사용법의 교집합으로 구성된 사용량 배열을 반환합니다.
CertIsRDNAttrsInCertificateName CertIsRDNAttrsInCertificateName 함수는 인증서 이름의 특성을 지정된 CERT_RDN 비교하여 모든 특성이 포함되는지 여부를 확인합니다.
CertIsStrongHashToSign 서명 인증서의 지정된 해시 알고리즘과 공개 키를 사용하여 강력한 서명을 수행할 수 있는지 여부를 결정합니다.
CertIsValidCRLForCertificate CertIsValidCRLForCertificate 함수는 CRL을 검사하여 인증서가 해지된 경우 특정 인증서를 포함하는 CRL인지 확인합니다.

CertModifyCertificatesToTrust 지정된 목적을 위해 CTL(인증서 신뢰 목록)의 인증서 집합을 수정합니다.
CertNameToStrA CERT_NAME_BLOB 구조체의 인코딩된 이름을 null로 끝나는 문자열로 변환합니다. (ANSI)
CertNameToStrW CERT_NAME_BLOB 구조체의 인코딩된 이름을 null로 끝나는 문자열로 변환합니다. (유니코드)
CertOIDToAlgId CNG에서 ALG_ID 식별자가 더 이상 지원되지 않으므로 이 함수 대신 CryptFindOIDInfo 함수를 사용합니다.
CertOpenServerOcspResponse 서버 인증서 체인과 연결된 OCSP(온라인 인증서 상태 프로토콜) 응답에 대한 핸들을 엽니다.
CertOpenStore 지정된 저장소 공급자 유형을 사용하여 인증서 저장소를 엽니다.
CertOpenSystemStoreA 가장 일반적인 시스템 인증서 저장소를 엽니다. 파일 기반 또는 메모리 기반 저장소와 같이 더 복잡한 요구 사항이 있는 인증서 저장소를 열려면 CertOpenStore를 사용합니다. (ANSI)
CertOpenSystemStoreW 가장 일반적인 시스템 인증서 저장소를 엽니다. 파일 기반 또는 메모리 기반 저장소와 같이 더 복잡한 요구 사항이 있는 인증서 저장소를 열려면 CertOpenStore를 사용합니다. (유니코드)
CertRDNValueToStrA CertRDNValueToStr 함수는 CERT_RDN_VALUE_BLOB 이름을 null로 끝나는 문자열로 변환합니다. (ANSI)
CertRDNValueToStrW CertRDNValueToStr 함수는 CERT_RDN_VALUE_BLOB 이름을 null로 끝나는 문자열로 변환합니다. (유니코드)
CertRegisterPhysicalStore 레지스트리 시스템 저장소 컬렉션에 물리적 저장소를 추가합니다.
CertRegisterSystemStore

시스템 저장소를 등록합니다.
CertRemoveEnhancedKeyUsageIdentifier CertRemoveEnhancedKeyUsageIdentifier 함수는 인증서의 EKU(고급 키 사용) 확장 속성에서 OID(사용 식별자 개체 식별자)를 제거합니다.
CertRemoveStoreFromCollection 컬렉션 저장소에서 형제 인증서 저장소를 제거합니다.
CertResyncCertificateChainEngine 인증서 체인 엔진을 다시 동기화하여 저장소의 엔진을 다시 동기화하고 엔진 캐시를 업데이트합니다.
CertRetrieveLogoOrBiometricInfo szOID_LOGOTYPE_EXT 또는 szOID_BIOMETRIC_EXT 인증서 확장에 지정된 로고 또는 생체 인식 정보의 URL 검색을 수행합니다.
CertSaveStore 인증서 저장소를 파일 또는 메모리 BLOB에 저장합니다.
CertSelectCertificateA 사용자가 지정된 조건과 일치하는 인증서 집합에서 인증서를 선택할 수 있는 대화 상자를 표시합니다. (ANSI)
CertSelectCertificateChains 지정된 선택 조건에 따라 인증서 체인을 검색합니다.
CertSelectCertificateW 사용자가 지정된 조건과 일치하는 인증서 집합에서 인증서를 선택할 수 있는 대화 상자를 표시합니다. (유니코드)
CertSelectionGetSerializedBlob CERT_SELECTUI_INPUT 구조에서 직렬화된 인증서 BLOB을 검색하는 데 사용되는 도우미 함수입니다.
CertSerializeCertificateStoreElement CertSerializeCertificateStoreElement 함수는 인증서 컨텍스트의 인코딩된 인증서 및 인코딩된 속성을 직렬화합니다. 나중에 인증서 및 속성을 검색할 수 있도록 결과를 스토리지에 유지할 수 있습니다.
CertSerializeCRLStoreElement

CertSerializeCRLStoreElement 함수는 인코딩된 CRL(인증서 해지 목록) 컨텍스트와 해당 속성의 인코딩된 표현을 직렬화합니다.
CertSerializeCTLStoreElement CertSerializeCTLStoreElement 함수는 인코딩된 CTL(인증서 신뢰 목록) 컨텍스트와 해당 속성의 인코딩된 표현을 직렬화합니다. 나중에 CTL 및 속성을 검색할 수 있도록 결과를 스토리지에 유지할 수 있습니다.
CertSetCertificateContextPropertiesFromCTLEntry 지정된 CTL(인증서 신뢰 목록) 항목의 특성을 사용하여 인증서 컨텍스트의 속성을 설정합니다.
CertSetCertificateContextProperty 지정된 인증서 컨텍스트에 대한 확장 속성을 설정합니다.
CertSetCRLContextProperty 지정된 CRL(인증서 해지 목록) 컨텍스트에 대한 확장 속성을 설정합니다.
CertSetCTLContextProperty 지정된 CTL(인증서 신뢰 목록) 컨텍스트에 대한 확장 속성을 설정합니다.
CertSetEnhancedKeyUsage CertSetEnhancedKeyUsage 함수는 인증서에 대한 EKU(고급 키 사용) 속성을 설정합니다.
CertSetStoreProperty CertSetStoreProperty 함수는 저장소 속성을 설정합니다.
CertSrvBackupClose CertSrvBackupOpenFile 함수에서 연 파일을 닫습니다.
CertSrvBackupEnd 인증서 서비스 백업 세션을 종료합니다.
CertSrvBackupFree 특정 인증서 서비스 백업 API에서 할당된 메모리를 해제하는 데 사용됩니다.
CertSrvBackupGetBackupLogsW 지정된 백업 컨텍스트에 대해 백업해야 하는 인증서 서비스 로그 파일 이름 목록을 검색합니다.
CertSrvBackupGetDatabaseNamesW

지정된 백업 컨텍스트에 대해 백업해야 하는 Certificate Services 데이터베이스 파일 이름 목록을 검색합니다.
CertSrvBackupGetDynamicFileListW 지정된 백업 컨텍스트에 대해 백업해야 하는 Certificate Services 동적 파일 이름 목록을 검색합니다.
CertSrvBackupOpenFileW 백업할 파일을 엽니다.
CertSrvBackupPrepareW 백업 작업을 위해 Certificate Services 서버를 준비하는 데 사용됩니다.
CertSrvBackupRead 인증서 서비스 파일에서 바이트를 읽습니다.
CertSrvBackupTruncateLogs 중복 레코드를 제거하고 로그 파일에 사용되는 디스크 스토리지 공간을 줄입니다.
CertSrvIsServerOnlineW 인증서 서비스 서버가 온라인 상태인지 확인합니다. 인증서 서비스 서버가 온라인 상태가 아니면 백업 작업이 성공하지 않습니다.
CertSrvRestoreEnd 인증서 서비스 복원 세션을 종료합니다.
CertSrvRestoreGetDatabaseLocationsW 백업 및 복원 시나리오에서 모두 사용되며 백업 또는 복원되는 모든 파일에 대한 Certificate Services 데이터베이스 위치 이름 목록을 검색합니다.
CertSrvRestorePrepareW 복원 작업을 위해 Certificate Services 인스턴스를 준비합니다.
CertSrvRestoreRegisterComplete 등록된 인증서 서비스 복원 작업을 완료합니다.
CertSrvRestoreRegisterThroughFile 인증서 서비스 복원을 등록합니다. (CertSrvRestoreRegisterThroughFile)

CertSrvRestoreRegisterW 인증서 서비스 복원을 등록합니다. (CertSrvRestoreRegisterW)
CertSrvServerControlW 프로그래밍 방식으로 인증서 서비스를 중지하는 서비스 제어 명령을 실행합니다.
CertStrToNameA null로 종료된 X.500 문자열을 인코딩된 인증서 이름으로 변환합니다. (ANSI)
CertStrToNameW null로 종료된 X.500 문자열을 인코딩된 인증서 이름으로 변환합니다. (유니코드)
CertUnregisterPhysicalStore CertUnregisterPhysicalStore 함수는 지정된 시스템 저장소 컬렉션에서 물리적 저장소를 제거합니다. CertUnregisterPhysicalStore를 사용하여 실제 저장소를 삭제할 수도 있습니다.
CertUnregisterSystemStore CertUnregisterSystemStore 함수는 지정된 시스템 저장소의 등록을 취소합니다.
CertVerifyCertificateChainPolicy 인증서 체인을 검사하여 지정된 유효성 정책 조건 준수를 포함하여 인증서 체인의 유효성을 확인합니다.
CertVerifyCRLRevocation CRL(인증서 해지 목록)을 확인하여 주체의 인증서가 해지되었는지 여부를 확인합니다.
CertVerifyCRLTimeValidity CertVerifyCRLTimeValidity 함수는 CRL의 시간 유효성을 확인합니다.
CertVerifyCTLUsage 주체가 포함된 사용 식별자를 사용하여 서명되고 시간이 유효한 CTL(인증서 신뢰 목록)을 찾아 주체가 지정된 사용에 대해 신뢰할 수 있는지 확인합니다.
CertVerifyRevocation rgpvContext 배열에 포함된 인증서의 해지 상태를 확인합니다. 목록에 있는 인증서가 해지된 것으로 확인되면 추가 검사가 수행되지 않습니다.
CertVerifySubjectCertificateContext

CertVerifySubjectCertificateContext 함수는 인증서 발급자의 유효성을 확인하여 인증서에 대해 활성화된 확인 검사를 수행합니다. 이 함수 대신 새 인증서 체인 확인 함수를 사용하는 것이 좋습니다.
CertVerifyTimeValidity CertVerifyTimeValidity 함수는 인증서의 시간 유효성을 확인합니다.
CertVerifyValidityNesting CertVerifyValidityNesting 함수는 주체 인증서의 시간 유효성이 발급자의 시간 유효성 내에 올바르게 중첩되었는지 확인합니다.
CertViewPropertiesA CertViewProperties 함수는 UI(사용자 인터페이스) 대화 상자에 인증서의 속성을 표시합니다. 이 함수에는 연결된 가져오기 라이브러리가 없습니다. LoadLibrary 및 GetProcAddress 함수를 사용하여 CryptDlg.dll동적으로 연결해야 합니다. (ANSI)
CertViewPropertiesW CertViewProperties 함수는 UI(사용자 인터페이스) 대화 상자에 인증서의 속성을 표시합니다. 이 함수에는 연결된 가져오기 라이브러리가 없습니다. LoadLibrary 및 GetProcAddress 함수를 사용하여 CryptDlg.dll동적으로 연결해야 합니다. (유니코드)
ChangeAccountPasswordA 지정된 보안 지원 공급자를 사용하여 Windows 도메인 계정의 암호를 변경합니다. (ANSI)
ChangeAccountPasswordW 지정된 보안 지원 공급자를 사용하여 Windows 도메인 계정의 암호를 변경합니다. (유니코드)
ChangeCredential 지정된 ID와 연결된 자격 증명을 변경합니다.
ChangeServiceConfig2A 서비스의 선택적 구성 매개 변수를 변경합니다. (ANSI)
ChangeServiceConfig2W 서비스의 선택적 구성 매개 변수를 변경합니다. (유니코드)
CheckCertificateSignature 지정된 서명자의 서명을 확인합니다.
CheckCertificateSignature

인증서 서명을 확인합니다.
CheckPublicKeySignature 서명 인증서의 공개 키를 사용하여 인증서 서명을 확인합니다.
CheckSignature 인증서 요청이 서명되었고 서명이 유효한지 확인합니다. (IX509CertificateRequestCmc2.CheckSignature)
CheckSignature 인증서 요청이 서명되었고 서명이 유효한지 확인합니다. (IX509CertificateRequestPkcs10.CheckSignature)
CheckTokenCapability 지정된 토큰의 기능을 확인합니다.
CheckTokenMembership 액세스 토큰에서 지정된 SID(보안 식별자)를 사용할 수 있는지 여부를 확인합니다.
CheckTokenMembershipEx 지정된 토큰에서 지정된 SID를 사용할 수 있는지 여부를 확인합니다.
지우기 컬렉션에서 모든 개체를 제거합니다. (IAlternativeNames.Clear)
지우기 컬렉션에서 모든 개체를 제거합니다. (ICertificatePolicies.Clear)
지우기 컬렉션에서 모든 ICertificationAuthority 개체를 제거합니다.
지우기 컬렉션에서 모든 속성을 제거합니다.
지우기 컬렉션에서 모든 ICryptAttribute 개체를 제거합니다.
지우기 컬렉션에서 모든 ICspAlgorithm 개체를 제거합니다.

지우기 컬렉션에서 모든 ICspInformation 개체를 제거합니다.
지우기 컬렉션에서 모든 ICspStatus 개체를 제거합니다.
지우기 컬렉션에서 모든 IObjectId 개체를 제거합니다.
지우기 컬렉션에서 모든 개체를 제거합니다. (IPolicyQualifiers.Clear)
지우기 컬렉션에서 모든 ISignerCertificate 개체를 제거합니다.
지우기 컬렉션에서 모든 개체를 제거합니다. (ISmimeCapabilities.Clear)
지우기 컬렉션에서 모든 IX509Attribute 개체를 제거합니다.
지우기 컬렉션에서 모든 IX509CertificateTemplate 개체를 제거합니다.
지우기 컬렉션에서 모든 IX509Extension 개체를 제거합니다.
지우기 컬렉션에서 모든 IX509NameValuePair 개체를 제거합니다.
지우기 컬렉션에서 모든 IX509PolicyServerUrl 개체를 제거합니다.
클론 특성 열거형 시퀀스 개체의 복사본을 현재 상태로 만듭니다.
클론

열거형 시퀀스의 복사본을 만듭니다.
클론 확장 열거형 시퀀스의 복사본을 만듭니다.
닫기 인증 키를 닫습니다. Open 메서드가 성공적으로 호출된 후에만 Close 메서드를 호출할 수 있습니다.
닫기 CSP(암호화 서비스 공급자) 또는 CNG(암호화 API:_Next Generation) KSP(키 스토리지 공급자)의 핸들을 해제합니다.
CloseApplication 캐시에서 지정된 IAzApplication 개체를 언로드합니다.
closeHandle CloseHandle 메서드는 ISceSvcAttachmentData::Initialize에 대한 이전 호출 중에 열린 핸들을 닫습니다.
CloseServiceHandle 서비스 제어 관리자 또는 서비스 개체에 대한 핸들을 닫습니다.
커밋 템플릿을 삭제하거나 Active Directory에 저장합니다.
completeAuth토큰 인증 토큰을 완료합니다. (CompleteAuthToken)
ComputeEffectivePermissionWithSecondarySecurity 개체에 대한 보조 보안을 사용하여 유효 권한을 계산합니다.
ComputeKeyIdentifier 공개 키의 160비트 SHA-1 해시에서 식별자를 만듭니다.
ComputeSiteCosts 현재 사용되지 않습니다.
설정 모듈 사용자 인터페이스를 표시합니다.

ConnectIdentity ID를 도메인 사용자에 연결합니다.
ControlService 컨트롤 코드를 서비스로 보냅니다. (ControlService)
ControlServiceExA 컨트롤 코드를 서비스로 보냅니다. (ControlServiceExA)
ControlServiceExW 컨트롤 코드를 서비스로 보냅니다. (ControlServiceExW)
ConvertSecurityDescriptorToStringSecurityDescriptorA 보안 설명자를 문자열 형식으로 변환합니다. 문자열 형식을 사용하여 보안 설명자를 저장하거나 전송할 수 있습니다. (ANSI)
ConvertSecurityDescriptorToStringSecurityDescriptorW 보안 설명자를 문자열 형식으로 변환합니다. 문자열 형식을 사용하여 보안 설명자를 저장하거나 전송할 수 있습니다. (유니코드)
ConvertSidToStringSidA SID(보안 식별자)를 표시, 스토리지 또는 전송에 적합한 문자열 형식으로 변환합니다. (ANSI)
ConvertSidToStringSidW SID(보안 식별자)를 표시, 스토리지 또는 전송에 적합한 문자열 형식으로 변환합니다. (유니코드)
ConvertStringSecurityDescriptorToSecurityDescriptorA 문자열 형식 보안 설명자를 유효한 기능 보안 설명자로 변환합니다. (ANSI)
ConvertStringSecurityDescriptorToSecurityDescriptorW 문자열 형식 보안 설명자를 유효한 기능 보안 설명자로 변환합니다. (유니코드)
ConvertStringSidToSidA 문자열 형식 SID(보안 식별자)를 유효한 기능 SID로 변환합니다. 이 함수를 사용하여 ConvertSidToStringSid 함수가 문자열 형식으로 변환된 SID를 검색할 수 있습니다. (ANSI)
ConvertStringSidToSidW

문자열 형식 SID(보안 식별자)를 유효한 기능 SID로 변환합니다. 이 함수를 사용하여 ConvertSidToStringSid 함수가 문자열 형식으로 변환된 SID를 검색할 수 있습니다. (유니코드)
ConvertToAutoInheritPrivateObjectSecurity 보안 설명자와 ACL(액세스 제어 목록)을 상속 가능한 ACE(액세스 제어 항목)의 자동 전파를 지원하는 형식으로 변환합니다.
ConvertToSid 지정된 ID 및 ID 공급자와 연결된 SID(보안 식별자)를 검색합니다.
CopySid SID(보안 식별자)를 버퍼에 복사합니다.
창조하다 지정된 사용자 이름과 연결된 새 ID를 만듭니다.
창조하다 비대칭 프라이빗 키를 만듭니다.
CreateApplication 지정된 이름을 사용하여 IAzApplication 개체를 만듭니다.
CreateApplication2 지정된 이름을 사용하여 IAzApplication2 개체를 만듭니다.
CreateApplicationGroup 지정된 이름을 사용하여 IAzApplicationGroup 개체를 만듭니다. (IAzApplication.CreateApplicationGroup)
CreateApplicationGroup 지정된 이름을 사용하여 IAzApplicationGroup 개체를 만듭니다. (IAzAuthorizationStore.CreateApplicationGroup)
CreateApplicationGroup 지정된 이름을 사용하여 IAzApplicationGroup 개체를 만듭니다. (IAzScope.CreateApplicationGroup)
CreateCAConfiguration 새 CA(인증 기관) 구성을 만들고 구성 집합에 추가합니다.

createFilePFX 허용된 인증서 체인 및 프라이빗 키를 PFX(개인 정보 교환) 형식의 파일에 저장합니다. 이 메서드는 ICEnroll4 인터페이스에서 처음 정의되었습니다.
createFilePFXWStr 허용된 인증서 체인 및 프라이빗 키를 PFX(개인 정보 교환) 형식의 파일에 저장합니다.
createFilePKCS10 base64로 인코딩된 PKCS(ICEnroll.createFilePKCS10)를 만듭니다.
createFilePKCS10WStr base64로 인코딩된 PKCS(IEnroll.createFilePKCS10WStr)를 만듭니다.
createFileRequest PKCS(ICEnroll4.createFileRequest)를 만듭니다.
createFileRequestWStr PKCS(IEnroll4.createFileRequestWStr)를 만듭니다.
CreateObject 웹 페이지의 사용자 컨텍스트에서 개체를 만드는 데 사용할 수 있습니다.
CreateObject 웹 페이지에 IX509EnrollmentHelper 개체를 만듭니다.
CreateOperation 지정된 이름을 사용하여 IAzOperation 개체를 만듭니다.
createPFX 허용된 인증서 체인 및 프라이빗 키를 PFX(개인 정보 교환) 형식 문자열에 저장합니다. PFX 형식을 PKCS(ICEnroll4.createPFX)라고도 합니다.
CREATEPFX PFX(개인 정보 교환) 메시지를 만듭니다.
createPFXWStr

허용된 인증서 체인 및 프라이빗 키를 PFX(개인 정보 교환) 형식 문자열에 저장합니다. PFX 형식을 PKCS(IEnroll4.createPFXWStr)라고도 합니다.
createPKCS10 base64로 인코딩된 PKCS(ICEnroll.createPKCS10)를 만듭니다.
createPKCS10WStr base64로 인코딩된 PKCS(IEnroll.createPKCS10WStr)를 만듭니다.
createPKCS7RequestFromRequest CreatePKCS7RequestFromRequest 메서드는 PKCS를 만듭니다.
CreatePrivateObjectSecurity 새 프라이빗 개체에 대한 자체 상대 보안 설명자를 할당하고 초기화합니다. 보호된 서버는 새 프라이빗 개체를 만들 때 이 함수를 호출합니다.
CreatePrivateObjectSecurityEx 이 함수를 호출하는 리소스 관리자가 만든 새 프라이빗 개체에 대한 자체 상대 보안 설명자를 할당하고 초기화합니다. (CreatePrivateObjectSecurityEx)
CreatePrivateObjectSecurityWithMultipleInheritance 이 함수를 호출하는 리소스 관리자가 만든 새 프라이빗 개체에 대한 자체 상대 보안 설명자를 할당하고 초기화합니다. (CreatePrivateObjectSecurityWithMultipleInheritance)
CreateProperty 새 속성을 만들어 속성 집합에 추가합니다.
createRequest PKCS(ICEnroll4.createRequest)를 만듭니다.
createRequest 인코딩된 인증서 요청을 검색합니다.
CreateRequestMessage 챌린지 암호를 사용하여 PKCS10 요청 메시지를 만듭니다. 요청 메시지는 SCEP 서버 암호화 인증서로 암호화되고 서버 서명 인증서로 서명된 봉투형 PKCS7에 있습니다.
createRequestWStr

PKCS(IEnroll4.createRequestWStr)를 만듭니다.
CreateRestrictedToken 기존 액세스 토큰의 제한된 버전인 새 액세스 토큰을 만듭니다. 제한된 토큰에는 SID(보안 식별자), 삭제된 권한 및 제한 SID 목록이 있을 수 있습니다.
CreateRetrieveCertificateMessage 이전에 발급한 인증서를 검색합니다.
CreateRetrievePendingMessage 인증서 폴링(수동 등록)에 대한 메시지를 만듭니다.
CreateRole 지정된 이름을 사용하여 IAzRole 개체를 만듭니다. (IAzApplication.CreateRole)
CreateRole 지정된 이름을 사용하여 IAzRole 개체를 만듭니다. (IAzScope.CreateRole)
CreateRoleAssignment 지정된 이름을 사용하여 새 IAzRoleAssignment 개체를 만듭니다.
CreateRoleAssignment 이 범위에서 지정된 이름을 사용하여 새 IAzRoleAssignment 개체를 만듭니다.
CreateRoleDefinition 지정된 이름을 사용하여 새 IAzRoleDefinition 개체를 만듭니다.
CreateRoleDefinition 이 범위에서 지정된 이름을 사용하여 새 IAzRoleDefinition 개체를 만듭니다.
CreateScope 지정된 이름을 사용하여 IAzScope 개체를 만듭니다.
CreateScope2 지정된 이름을 사용하여 새 IAzScope2 개체를 만듭니다.
CreateSecurityPage

사용자가 개체의 DACL(임의 액세스 제어 목록)에서 ACE(액세스 제어 항목)에서 허용되거나 거부된 액세스 권한을 보고 편집할 수 있도록 하는 기본 보안 속성 페이지를 만듭니다.
CreateTask 지정된 이름을 사용하여 IAzTask 개체를 만듭니다. (IAzApplication.CreateTask)
CreateTask 지정된 이름을 사용하여 IAzTask 개체를 만듭니다. (IAzScope.CreateTask)
CreateVirtualSmartCard 지정된 매개 변수를 사용하여 TPM 가상 스마트 카드를 만듭니다.
CreateWellKnownSid 미리 정의된 별칭에 대한 SID를 만듭니다.
CredDeleteA 사용자의 자격 증명 집합에서 자격 증명을 삭제합니다. (ANSI)
CredDeleteW 사용자의 자격 증명 집합에서 자격 증명을 삭제합니다. (유니코드)
CredEnumerateA 사용자의 자격 증명 집합에서 자격 증명을 열거합니다. (ANSI)
CredEnumerateW 사용자의 자격 증명 집합에서 자격 증명을 열거합니다. (유니코드)
CredFindBestCredentialA CredMan(자격 증명 관리) 데이터베이스에서 현재 로그온 세션과 연결되고 지정된 대상 리소스와 가장 일치하는 일반 자격 증명 집합을 검색합니다. (ANSI)
CredFindBestCredentialW CredMan(자격 증명 관리) 데이터베이스에서 현재 로그온 세션과 연결되고 지정된 대상 리소스와 가장 일치하는 일반 자격 증명 집합을 검색합니다. (유니코드)
CredFree CredFree 함수는 자격 증명 관리 함수에서 반환된 버퍼를 해제합니다.

CredFreeCredentialsFn 보안 패키지에서 사용하는 자격 증명을 저장하는 데 사용되는 메모리를 해제합니다.
CredGetSessionTypes CredGetSessionTypes 함수는 현재 로그온 세션에서 지원하는 최대 지속성을 반환합니다. 각 자격 증명 형식에 대해 별도의 최대 지속성이 반환됩니다.
CredGetTargetInfoA CredGetTargetInfo 함수는 명명된 대상 컴퓨터에 대해 알려진 모든 대상 이름 정보를 검색합니다. (ANSI)
CredGetTargetInfoW CredGetTargetInfo 함수는 명명된 대상 컴퓨터에 대해 알려진 모든 대상 이름 정보를 검색합니다. (유니코드)
CredIsMarshaledCredentialA 지정된 사용자 이름 문자열이 이전에 CredMarshalCredential에 의해 마샬링된 마샬링된 자격 증명인지 여부를 확인합니다. (ANSI)
CredIsMarshaledCredentialW 지정된 사용자 이름 문자열이 이전에 CredMarshalCredential에 의해 마샬링된 마샬링된 자격 증명인지 여부를 확인합니다. (유니코드)
CredIsProtectedA CredProtect 함수에 대한 이전 호출을 통해 지정된 자격 증명을 암호화할지 여부를 지정합니다. (ANSI)
CredIsProtectedW CredProtect 함수에 대한 이전 호출을 통해 지정된 자격 증명을 암호화할지 여부를 지정합니다. (유니코드)
CrediUnmarshalandDecodeStringFn 마샬링된 문자열을 원래 형식으로 다시 변환하고, 숨을 들이지 않은 문자열의 암호를 해독합니다.
CredMarshalCredentialA CredMarshalCredential 함수는 자격 증명을 텍스트 문자열로 변환합니다. (ANSI)
CredMarshalCredentialW CredMarshalCredential 함수는 자격 증명을 텍스트 문자열로 변환합니다. (유니코드)

CredMarshalTargetInfo 지정된 대상을 바이트 값 배열로 직렬화합니다.
CredPackAuthenticationBufferA 문자열 사용자 이름 및 암호를 인증 버퍼로 변환합니다. (ANSI)
CredPackAuthenticationBufferW 문자열 사용자 이름 및 암호를 인증 버퍼로 변환합니다. (유니코드)
CredProtectA 현재 보안 컨텍스트만 암호를 해독할 수 있도록 지정된 자격 증명을 암호화합니다. (ANSI)
CredProtectW 현재 보안 컨텍스트만 암호를 해독할 수 있도록 지정된 자격 증명을 암호화합니다. (유니코드)
CredReadA 사용자의 자격 증명 집합에서 자격 증명을 읽습니다. (ANSI)
CredReadDomainCredentialsA 사용자의 자격 증명 집합에서 도메인 자격 증명을 읽습니다. (ANSI)
CredReadDomainCredentialsFn 자격 증명 관리자에서 도메인 자격 증명을 읽습니다.
CredReadDomainCredentialsW 사용자의 자격 증명 집합에서 도메인 자격 증명을 읽습니다. (유니코드)
CredReadFn 자격 증명 관리자에서 자격 증명을 읽습니다.
CredReadW 사용자의 자격 증명 집합에서 자격 증명을 읽습니다. (유니코드)
CredRenameA CredRename은 더 이상 지원되지 않습니다. (ANSI)
CredRenameW

CredRename은 더 이상 지원되지 않습니다. (유니코드)
CredUICmdLinePromptForCredentialsA 명령줄(콘솔) 애플리케이션에서 작업하는 사용자의 자격 증명 정보를 묻는 메시지를 표시하고 수락합니다. 사용자가 입력한 이름과 암호는 확인을 위해 호출 애플리케이션에 다시 전달됩니다. (ANSI)
CredUICmdLinePromptForCredentialsW 명령줄(콘솔) 애플리케이션에서 작업하는 사용자의 자격 증명 정보를 묻는 메시지를 표시하고 수락합니다. 사용자가 입력한 이름과 암호는 확인을 위해 호출 애플리케이션에 다시 전달됩니다. (유니코드)
CredUIConfirmCredentialsA 수집된 자격 증명의 유효성을 확인하기 위해 CredUIPromptForCredentials 또는 CredUICmdLinePromptForCredentials 후에 호출됩니다. (ANSI)
CredUIConfirmCredentialsW 수집된 자격 증명의 유효성을 확인하기 위해 CredUIPromptForCredentials 또는 CredUICmdLinePromptForCredentials 후에 호출됩니다. (유니코드)
CredUIParseUserNameA CredUIParseUserName 함수는 정규화된 사용자 이름에서 도메인 및 사용자 계정 이름을 추출합니다. (ANSI)
CredUIParseUserNameW CredUIParseUserName 함수는 정규화된 사용자 이름에서 도메인 및 사용자 계정 이름을 추출합니다. (유니코드)
CredUIPromptForCredentialsA 사용자의 자격 증명 정보를 허용하는 구성 가능한 대화 상자를 만들고 표시합니다. (ANSI)
CredUIPromptForWindowsCredentialsA 로컬 컴퓨터에 설치된 자격 증명 공급자를 사용하여 사용자가 자격 증명 정보를 제공할 수 있는 구성 가능한 대화 상자를 만들고 표시합니다. (ANSI)
CredUIPromptForWindowsCredentialsW 로컬 컴퓨터에 설치된 자격 증명 공급자를 사용하여 사용자가 자격 증명 정보를 제공할 수 있는 구성 가능한 대화 상자를 만들고 표시합니다. (유니코드)
CredUIReadSSOCredW CredUIReadSSOCredW 함수는 단일 로그온 자격 증명에 대한 사용자 이름을 검색합니다.

CredUIStoreSSOCredW CredUIStoreSSOCredW 함수는 단일 로그온 자격 증명을 저장합니다.
CredUnmarshalCredentialA CredUnmarshalCredential 함수는 마샬링된 자격 증명을 원래 형식으로 다시 변환합니다. (ANSI)
CredUnmarshalCredentialW CredUnmarshalCredential 함수는 마샬링된 자격 증명을 원래 형식으로 다시 변환합니다. (유니코드)
CredUnPackAuthenticationBufferA CredUIPromptForWindowsCredentials 함수를 호출하여 반환된 인증 버퍼를 문자열 사용자 이름 및 암호로 변환합니다. (ANSI)
CredUnPackAuthenticationBufferW CredUIPromptForWindowsCredentials 함수를 호출하여 반환된 인증 버퍼를 문자열 사용자 이름 및 암호로 변환합니다. (유니코드)
CredUnprotectA CredProtect 함수를 사용하여 이전에 암호화된 자격 증명의 암호를 해독합니다. (ANSI)
CredUnprotectW CredProtect 함수를 사용하여 이전에 암호화된 자격 증명의 암호를 해독합니다. (유니코드)
CredWriteA 새 자격 증명을 만들거나 사용자의 자격 증명 집합에서 기존 자격 증명을 수정합니다. (ANSI)
CredWriteDomainCredentialsA 사용자의 자격 증명 집합에 도메인 자격 증명을 씁니다. (ANSI)
CredWriteDomainCredentialsW 사용자의 자격 증명 집합에 도메인 자격 증명을 씁니다. (유니코드)
CredWriteFn 지정된 자격 증명을 자격 증명 관리자에 씁니다.
CredWriteW 새 자격 증명을 만들거나 사용자의 자격 증명 집합에서 기존 자격 증명을 수정합니다. (유니코드)

CryptAcquireCertificatePrivateKey 인증서의 프라이빗 키를 가져옵니다.
CryptAcquireContextA 특정 CSP(암호화 서비스 공급자) 내의 특정 키 컨테이너에 대한 핸들을 획득하는 데 사용됩니다. 이 반환된 핸들은 선택한 CSP를 사용하는 CryptoAPI 함수 호출에 사용됩니다. (ANSI)
CryptAcquireContextW 특정 CSP(암호화 서비스 공급자) 내의 특정 키 컨테이너에 대한 핸들을 획득하는 데 사용됩니다. 이 반환된 핸들은 선택한 CSP를 사용하는 CryptoAPI 함수 호출에 사용됩니다. (유니코드)
CryptBinaryToStringA 바이트 배열을 형식이 지정된 문자열로 변환합니다. (ANSI)
CryptBinaryToStringW 바이트 배열을 형식이 지정된 문자열로 변환합니다. (유니코드)
CryptCATAdminAcquireContext 카탈로그 관리자 컨텍스트에 대한 핸들을 가져옵니다.
CryptCATAdminAcquireContext2 지정된 해시 알고리즘 및 해시 정책에 대한 카탈로그 관리자 컨텍스트에 대한 핸들을 가져옵니다.
CryptCATAdminAddCatalog 카탈로그 데이터베이스에 카탈로그를 추가합니다.
CryptCATAdminCalcHashFromFileHandle 파일의 해시를 계산합니다.
CryptCATAdminCalcHashFromFileHandle2 지정된 알고리즘을 사용하여 파일의 해시를 계산합니다.
CryptCATAdminEnumCatalogFromHash 지정된 해시를 포함하는 카탈로그를 열거합니다.
CryptCATAdminReleaseCatalogContext 이전에 CryptCATAdminAddCatalog 함수에서 반환한 카탈로그 컨텍스트에 대한 핸들을 해제합니다.

CryptCATAdminReleaseContext 이전에 CryptCATAdminAcquireContext 함수에 의해 할당된 핸들을 해제합니다.
CryptCATAdminRemoveCatalog 카탈로그 파일을 삭제하고 Windows 카탈로그 데이터베이스에서 해당 카탈로그의 항목을 제거합니다.
CryptCATAdminResolveCatalogPath 지정된 카탈로그의 정규화된 경로를 검색합니다.
CryptCATCatalogInfoFromContext 지정된 카탈로그 컨텍스트에서 카탈로그 정보를 검색합니다.
CryptCATCDFClose CDF(카탈로그 정의 파일)를 닫고 해당 CRYPTCATCDF 구조에 대한 메모리를 해제합니다.
CryptCATCDFEnumCatAttributes CDF(카탈로그 정의 파일)의 CatalogHeader 섹션 내에서 카탈로그 수준 특성을 열거합니다.
CryptCATCDFOpen 기존 CDF(카탈로그 정의 파일)를 열어 CRYPTCATCDF 구조를 읽고 초기화합니다.
CryptCATClose CryptCATOpen 함수에서 이전에 연 카탈로그 핸들을 닫습니다.
CryptCATEnumerateAttr 카탈로그의 멤버와 연결된 특성을 열거합니다. 이 함수에는 연결된 가져오기 라이브러리가 없습니다.
CryptCATEnumerateCatAttr 카탈로그와 연결된 특성을 열거합니다. 이 함수에는 연결된 가져오기 라이브러리가 없습니다.
CryptCATEnumerateMember 카탈로그의 멤버를 열거합니다.
CryptCATGetAttrInfo 카탈로그 멤버의 특성에 대한 정보를 검색합니다.
CryptCATGetMemberInfo

카탈로그의 PKCS에서 멤버 정보를 검색합니다.
CryptCATHandleFromStore 메모리에서 카탈로그 핸들을 검색합니다.
CryptCATOpen 카탈로그를 열고 열려 있는 카탈로그에 대한 컨텍스트 핸들을 반환합니다.
CryptCATPersistStore 지정된 카탈로그 저장소의 정보를 서명되지 않은 카탈로그 파일에 저장합니다.
CryptCATPutAttrInfo 특성에 대한 메모리를 할당하고 카탈로그 멤버에 추가합니다.
CryptCATPutCatAttrInfo 카탈로그 파일 특성에 대한 메모리를 할당하고 카탈로그에 추가합니다.
CryptCATPutMemberInfo 카탈로그 멤버에 대한 메모리를 할당하고 카탈로그에 추가합니다.
CryptCATStoreFromHandle 카탈로그 핸들에서 CRYPTCATSTORE 구조를 검색합니다.
CryptContextAddRef HCRYPTPROV CSP(암호화 서비스 공급자) 핸들의 참조 수에 1개를 추가합니다.
CryptCreateHash 데이터 스트림의 해시를 시작합니다. 호출 애플리케이션에 CSP(암호화 서비스 공급자) 해시 개체에 대한 핸들을 만들고 반환합니다.
CryptCreateKeyIdentifierFromCSP 중요 이 API는 더 이상 사용되지 않습니다. (CryptCreateKeyIdentifierFromCSP)
CryptDecodeMessage 암호화 메시지를 디코딩, 암호 해독 및 확인합니다.
CryptDecodeObject

CryptDecodeObject 함수는 lpszStructType 매개 변수로 표시된 형식의 구조를 디코딩합니다. CryptDecodeObjectEx는 성능이 크게 향상되어 동일한 기능을 수행하는 API로 사용하는 것이 좋습니다.
CryptDecodeObjectEx lpszStructType 매개 변수로 표시된 형식의 구조를 디코딩합니다.
CryptDecrypt CryptEncrypt 함수를 사용하여 이전에 암호화된 데이터의 암호를 해독합니다.
CryptDecryptAndVerifyMessageSignature CryptDecryptAndVerifyMessageSignature 함수는 메시지의 암호를 해독하고 해당 서명을 확인합니다.
CryptDecryptMessage CryptDecryptMessage 함수는 메시지를 디코딩하고 암호를 해독합니다.
CryptDeriveKey 기본 데이터 값에서 파생된 암호화 세션 키를 생성합니다.
CryptDestroyHash hHash 매개 변수에서 참조하는 해시 개체를 제거합니다.
CryptDestroyKey hKey 매개 변수에서 참조하는 핸들을 해제합니다.
CryptDuplicateHash 복제가 완료된 시점까지 해시의 정확한 복사본을 만듭니다.
CryptDuplicateKey 키의 정확한 복사본과 키의 상태를 만듭니다.
CryptEncodeObject CryptEncodeObject 함수는 lpszStructType 매개 변수의 값으로 표시된 형식의 구조를 인코딩합니다. CryptEncodeObjectEx는 성능이 크게 향상되어 동일한 함수를 수행하는 API로 사용하는 것이 좋습니다.
CryptEncodeObjectEx lpszStructType 매개 변수의 값으로 표시된 형식의 구조를 인코딩합니다.
CryptEncrypt

데이터를 암호화합니다. 데이터를 암호화하는 데 사용되는 알고리즘은 CSP 모듈에서 보유한 키에 의해 지정되며 hKey 매개 변수에 의해 참조됩니다.
CryptEncryptMessage CryptEncryptMessage 함수는 메시지를 암호화하고 인코딩합니다.
CryptEnumKeyIdentifierProperties CryptEnumKeyIdentifierProperties 함수는 키 식별자와 해당 속성을 열거합니다.
CryptEnumOIDFunction CryptEnumOIDFunction 함수는 등록된 OID(개체 식별자) 함수를 열거합니다.
CryptEnumOIDInfo 미리 정의되고 등록된 OID(개체 식별자) CRYPT_OID_INFO 구조를 열거합니다. 이 함수는 미리 정의된 구조체와 등록된 구조체를 모두 열거하거나 선택한 OID 그룹으로 식별된 구조체만 열거합니다.
CryptEnumProvidersA 중요 이 API는 더 이상 사용되지 않습니다. (CryptEnumProvidersA)
CryptEnumProvidersW 중요 이 API는 더 이상 사용되지 않습니다. (CryptEnumProvidersW)
CryptEnumProviderTypesA 컴퓨터에서 지원되는 CSP(암호화 서비스 공급자)의 첫 번째 또는 다음 유형을 검색합니다. (ANSI)
CryptEnumProviderTypesW 컴퓨터에서 지원되는 CSP(암호화 서비스 공급자)의 첫 번째 또는 다음 유형을 검색합니다. (유니코드)
CryptExportKey CSP(암호화 서비스 공급자)에서 암호화 키 또는 키 쌍을 안전한 방식으로 내보냅니다.
CryptExportPKCS8 PKCS에서 프라이빗 키 내보내기(CryptExportPKCS8)
CryptExportPKCS8Ex PKCS(CryptExportPKCS8Ex)에서 프라이빗 키를 내보냅니다.
CryptExportPublicKeyInfo

CryptExportPublicKeyInfo 함수는 공급자의 해당 프라이빗 키와 연결된 공개 키 정보를 내보냅니다. 이 함수의 업데이트된 버전은 CryptExportPublicKeyInfoEx를 참조하세요.
CryptExportPublicKeyInfoEx 공급자의 해당 프라이빗 키와 연결된 공개 키 정보를 내보냅니다.
CryptExportPublicKeyInfoFromBCryptKeyHandle 공급자의 해당 프라이빗 키와 연결된 공개 키 정보를 내보냅니다.
CryptFindCertificateKeyProvInfo 암호화 공급자 및 해당 컨테이너를 열거하여 인증서의 공개 키에 해당하는 프라이빗 키를 찾습니다.
CryptFindLocalizedName 지정된 이름의 지역화된 이름(예: "Root" 시스템 저장소의 지역화 이름)을 찾습니다.
CryptFindOIDInfo 지정된 키 형식 및 키와 일치하는 첫 번째 미리 정의되거나 등록된 CRYPT_OID_INFO 구조를 검색합니다. 검색은 지정된 OID 그룹 내의 OID(개체 식별자)로 제한될 수 있습니다.
CryptFormatObject CryptFormatObject 함수는 인코딩된 데이터의 형식을 지정하고 인증서 인코딩 형식에 따라 할당된 버퍼에 유니코드 문자열을 반환합니다.
CryptFreeOIDFunctionAddress CryptFreeOIDFunctionAddress 함수는 함수 핸들의 참조 수를 줄여 CryptGetOIDFunctionAddress 또는 CryptGetDefaultOIDFunctionAddress에서 반환된 핸들을 해제합니다.
CryptGenKey 임의의 암호화 세션 키 또는 퍼블릭/프라이빗 키 쌍을 생성합니다. 키 또는 키 쌍에 대한 핸들이 phKey로 반환됩니다. 그런 다음 키 핸들이 필요한 CryptoAPI 함수에서 필요에 따라 이 핸들을 사용할 수 있습니다.
CryptGenRandom 암호화 임의 바이트로 버퍼를 채웁니다.
CryptGetDefaultOIDDllList CryptGetDefaultOIDDllList 함수는 지정된 함수 집합 및 인코딩 형식에 대해 등록된 OID(기본 개체 식별자) 함수를 포함하는 DLL 파일의 이름 목록을 가져옵니다.
CryptGetDefaultOIDFunctionAddress

CryptGetDefaultOIDFunctionAddress 함수는 기본 함수 주소를 포함하는 DLL을 로드합니다.
CryptGetDefaultProviderA 로컬 컴퓨터 또는 현재 사용자에 대해 지정된 공급자 유형의 기본 CSP(암호화 서비스 공급자)를 찾습니다. (ANSI)
CryptGetDefaultProviderW 로컬 컴퓨터 또는 현재 사용자에 대해 지정된 공급자 유형의 기본 CSP(암호화 서비스 공급자)를 찾습니다. (유니코드)
CryptGetHashParam 해시 개체의 작업을 제어하는 데이터를 검색합니다.
CryptGetKeyIdentifierProperty CryptGetKeyIdentifierProperty는 지정된 키 식별자에서 특정 속성을 획득합니다.
CryptGetKeyParam 키의 작업을 제어하는 데이터를 검색합니다.
CryptGetMessageCertificates CryptGetMessageCertificates 함수는 메시지의 인증서 및 CRL을 포함하는 열린 인증서 저장소의 핸들을 반환합니다. 이 함수는 공급자 형식 CERT_STORE_PROV_PKCS7 사용하여 CertOpenStore를 lpszStoreProvider 매개 변수로 호출합니다.
CryptGetMessageSignerCount CryptGetMessageSignerCount 함수는 서명된 메시지의 서명자 수를 반환합니다.
CryptGetObjectUrl 인증서, CTL(인증서 신뢰 목록) 또는 CRL(인증서 해지 목록)에서 원격 개체의 URL을 가져옵니다.
CryptGetOIDFunctionAddress 등록된 함수 및 설치된 함수 목록에서 인코딩 형식 및 OID(개체 식별자) 일치 항목을 검색합니다.
CryptGetOIDFunctionValue CryptGetOIDFunctionValue 함수는 OID와 연결된 값을 쿼리합니다.
CryptGetProvParam CSP(암호화 서비스 공급자)의 작업을 제어하는 매개 변수를 검색합니다.

CryptGetTimeValidObject 지정된 컨텍스트 및 시간 내에서 유효한 CRL, OCSP 응답 또는 CTL 개체를 검색합니다.
CryptGetUserKey 사용자의 두 공개/프라이빗 키 쌍 중 하나의 핸들을 검색합니다.
CryptHashCertificate CryptHashCertificate 함수는 서명을 포함하여 인증서의 인코딩된 전체 콘텐츠를 해시합니다.
CryptHashCertificate2 CNG 해시 공급자를 사용하여 데이터 블록을 해시합니다.
CryptHashData 지정된 해시 개체에 데이터를 추가합니다.
CryptHashMessage 메시지의 해시를 만듭니다.
CryptHashPublicKeyInfo CERT_PUBLIC_KEY_INFO 구조에서 공개 키 정보를 인코딩하고 인코딩된 바이트의 해시를 계산합니다.
CryptHashSessionKey 세션 키 개체의 암호화 해시를 계산합니다.
CryptHashToBeSigned 중요 이 API는 더 이상 사용되지 않습니다. (CryptHashToBeSigned)
CryptImportKey 키 BLOB에서 CSP(암호화 서비스 공급자)로 암호화 키를 전송합니다.
CryptImportPKCS8 PKCS에서 프라이빗 키 가져오기
CryptImportPublicKeyInfo 공개 키 정보를 변환하고 공급자로 가져오고 공개 키의 핸들을 반환합니다.
CryptImportPublicKeyInfoEx

중요 이 API는 더 이상 사용되지 않습니다. (CryptImportPublicKeyInfoEx)
CryptImportPublicKeyInfoEx2 공개 키를 OID(공개 키 개체 식별자)에 해당하는 CNG 비대칭 공급자로 가져오고 키에 대한 CNG 핸들을 반환합니다.
CryptInitOIDFunctionSet CryptInitOIDFunctionSet은 제공된 함수 집합 이름으로 식별된 OID 함수 집합의 핸들을 초기화하고 반환합니다.
CryptInstallDefaultContext 지정된 알고리즘의 기본 컨텍스트 공급자가 되도록 특정 공급자를 설치합니다.
CryptInstallOIDFunctionAddress CryptInstallOIDFunctionAddress 함수는 OID(호출 가능한 개체 식별자) 함수 주소 집합을 설치합니다.
CryptMemAlloc CryptMemAlloc 함수는 버퍼에 대한 메모리를 할당합니다. 할당된 버퍼를 반환하는 모든 Crypt32.lib 함수에서 사용됩니다.
CryptMemFree CryptMemFree 함수는 CryptMemAlloc 또는 CryptMemRealloc에서 할당한 메모리를 해제합니다.
CryptMemRealloc CryptMemRealloc 함수는 현재 버퍼에 할당된 메모리를 해제하고 새 버퍼에 대한 메모리를 할당합니다.
CryptMsgCalculateEncodedLength 인코딩할 메시지 유형, 인코딩 매개 변수 및 총 데이터 길이가 지정된 경우 인코딩된 암호화 메시지에 필요한 최대 바이트 수를 계산합니다.
CryptMsgClose CryptMsgClose 함수는 암호화 메시지 핸들을 닫습니다. 이 함수를 호출할 때마다 메시지의 참조 수가 1씩 줄어듭니다. 참조 수가 0에 도달하면 메시지가 완전히 해제됩니다.
CryptMsgControl CryptMsgUpdate 함수에 대한 최종 호출에 의해 메시지가 디코딩된 후 제어 작업을 수행합니다.
CryptMsgCountersign

메시지의 기존 서명을 카운터 서명합니다.
CryptMsgCountersignEncoded 기존 PKCS를 카운터 서명합니다.
CryptMsgDuplicate CryptMsgDuplicate 함수는 참조 수를 증가시켜 암호화 메시지 핸들을 복제합니다.
CryptMsgEncodeAndSignCTL CryptMsgEncodeAndSignCTL 함수는 CTL을 인코딩하고 인코딩된 CTL을 포함하는 서명된 메시지를 만듭니다. 이 함수는 먼저 pCtlInfo가 가리키는 CTL을 인코딩한 다음, CryptMsgSignCTL을 호출하여 인코딩된 메시지에 서명합니다.
CryptMsgGetAndVerifySigner CryptMsgGetAndVerifySigner 함수는 암호화 메시지의 서명을 확인합니다.
CryptMsgGetParam 암호화 메시지가 인코딩 또는 디코딩된 후 메시지 매개 변수를 가져옵니다.
CryptMsgOpenToDecode 디코딩을 위해 암호화 메시지를 열고 열린 메시지의 핸들을 반환합니다.
CryptMsgOpenToEncode 인코딩을 위해 암호화 메시지를 열고 열린 메시지의 핸들을 반환합니다.
CryptMsgSignCTL CryptMsgSignCTL 함수는 인코딩된 CTL을 포함하는 서명된 메시지를 만듭니다.
CryptMsgUpdate 암호화 메시지에 콘텐츠를 추가합니다.
CryptMsgVerifyCountersignatureEncoded SignerInfo 구조체(PKCS에서 정의한 대로)를 기준으로 카운터 서명 확인
CryptMsgVerifyCountersignatureEncodedEx pbSignerInfoCounterSignature 매개 변수가 pbSignerInfo 매개 변수 구조의 encryptedDigest 필드의 암호화된 해시를 포함하는지 확인합니다.

CryptProtectData DATA_BLOB 구조의 데이터에 대한 암호화를 수행합니다.
CryptProtectMemory 는 다른 사용자가 프로세스에서 중요한 정보를 볼 수 없도록 메모리를 암호화합니다.
CryptQueryObject 인증서, 인증서 해지 목록 또는 인증서 신뢰 목록과 같은 암호화 API 개체의 내용에 대한 정보를 검색합니다.
CryptRegisterDefaultOIDFunction CryptRegisterDefaultOIDFunction은 지정된 인코딩 형식 및 함수 이름에 대해 호출할 기본 함수를 포함하는 DLL을 등록합니다. CryptRegisterOIDFunction과 달리 DLL에서 내보낼 함수 이름은 재정의할 수 없습니다.
CryptRegisterOIDFunction 지정된 인코딩 형식, 함수 이름 및 OID(개체 식별자)에 대해 호출할 함수가 포함된 DLL을 등록합니다.
CryptRegisterOIDInfo CryptRegisterOIDInfo 함수는 CRYPT_OID_INFO 구조에 지정된 OID 정보를 등록하여 레지스트리에 유지합니다.
CryptReleaseContext CSP(암호화 서비스 공급자) 및 키 컨테이너의 핸들을 해제합니다.
CryptRetrieveObjectByUrlA URL로 지정된 위치에서 PKI(공개 키 인프라) 개체를 검색합니다. (ANSI)
CryptRetrieveObjectByUrlW URL로 지정된 위치에서 PKI(공개 키 인프라) 개체를 검색합니다. (유니코드)
CryptRetrieveTimeStamp 타임스탬프를 요청하고 URL로 지정된 위치에서 TSA(타임스탬핑 기관)로 타임스탬프를 검색합니다.
CryptSetHashParam 초기 해시 콘텐츠 설정 및 특정 해시 알고리즘 선택 등 해시 개체의 작업을 사용자 지정합니다.
CryptSetKeyIdentifierProperty

CryptSetKeyIdentifierProperty 함수는 지정된 키 식별자의 속성을 설정합니다. 이 함수는 pwszComputerName에서 식별된 컴퓨터에서 속성을 설정할 수 있습니다.
CryptSetKeyParam 세션 키 작업의 다양한 측면을 사용자 지정합니다.
CryptSetOIDFunctionValue CryptSetOIDFunctionValue 함수는 지정된 인코딩 형식, 함수 이름, OID 및 값 이름에 대한 값을 설정합니다.
CryptSetProviderA 현재 사용자의 기본 CSP(암호화 서비스 공급자)를 지정합니다. (ANSI)
CryptSetProviderExA 로컬 컴퓨터 또는 현재 사용자에 대해 지정된 공급자 유형의 기본 CSP(암호화 서비스 공급자)를 지정합니다. (ANSI)
CryptSetProviderExW 로컬 컴퓨터 또는 현재 사용자에 대해 지정된 공급자 유형의 기본 CSP(암호화 서비스 공급자)를 지정합니다. (유니코드)
CryptSetProviderW 현재 사용자의 기본 CSP(암호화 서비스 공급자)를 지정합니다. (유니코드)
CryptSetProvParam CSP(암호화 서비스 공급자)의 작업을 사용자 지정합니다. 이 함수는 일반적으로 CSP와 연결된 키 컨테이너에 보안 설명자를 설정하여 해당 키 컨테이너의 프라이빗 키에 대한 액세스를 제어하는 데 사용됩니다.
CryptSignAndEncodeCertificate 인증서, CRL(인증서 해지 목록), CTL(인증서 신뢰 목록) 또는 인증서 요청을 인코딩하고 서명합니다.
CryptSignAndEncryptMessage CryptSignAndEncryptMessage 함수는 지정된 콘텐츠의 해시를 만들고, 해시에 서명하고, 콘텐츠를 암호화하고, 암호화된 콘텐츠와 서명된 해시를 해시한 다음, 암호화된 콘텐츠와 서명된 해시를 모두 인코딩합니다.
CryptSignCertificate CryptSignCertificate 함수는 인코딩된 서명된 콘텐츠의 "서명할" 정보에 서명합니다.
CryptSignHashA

데이터에 서명합니다. (CryptSignHashA)
CryptSignHashW 데이터에 서명합니다. (CryptSignHashW)
CryptSignMessage CryptSignMessage 함수는 지정된 콘텐츠의 해시를 만들고 해시에 서명한 다음 원본 메시지 콘텐츠와 서명된 해시를 모두 인코딩합니다.
CryptSignMessageWithKey 매개 변수에 지정된 CSP의 프라이빗 키를 사용하여 메시지에 서명합니다.
CryptSIPAddProvider CryptSIPAddProvider 함수는 SIP(주체 인터페이스 패키지)를 구현하는 지정된 DLL 파일에서 내보낸 함수를 등록합니다.
CryptSIPCreateIndirectData 제공된 SIP_SUBJECTINFO 구조체의 해시, 다이제스트 알고리즘 및 인코딩 특성이 포함된 SIP_INDIRECT_DATA 구조체를 반환합니다. 해시는 데이터에 대한 간접 참조로 사용할 수 있습니다.
CryptSIPGetCaps SIP(주체 인터페이스 패키지)의 기능을 검색합니다.
CryptSIPGetSignedDataMsg 파일에서 Authenticode 서명을 검색합니다.
CryptSIPLoad SIP(주체 인터페이스 패키지)를 구현하는 DLL(동적 연결 라이브러리)을 로드하고 적절한 라이브러리 내보내기 함수를 SIP_DISPATCH_INFO 구조에 할당합니다.
CryptSIPPutSignedDataMsg 대상 파일에 Authenticode 서명을 저장합니다.
CryptSIPRemoveProvider CryptSIPAddProvider 함수에 대한 이전 호출에서 추가한 SIP(주체 인터페이스 패키지) DLL 파일의 레지스트리 세부 정보를 제거합니다.
CryptSIPRemoveSignedDataMsg

지정된 Authenticode 서명을 제거합니다.
CryptSIPRetrieveSubjectGuid 지정된 파일의 헤더 정보를 기반으로 GUID를 검색합니다.
CryptSIPRetrieveSubjectGuidForCatalogFile 지정된 파일과 연결된 주체 GUID를 검색합니다.
CryptSIPVerifyIndirectData 제공된 주체에 대해 간접 해시된 데이터의 유효성을 검사합니다.
CryptStringToBinaryA 서식이 지정된 문자열을 바이트 배열로 변환합니다. (ANSI)
CryptStringToBinaryW 서식이 지정된 문자열을 바이트 배열로 변환합니다. (유니코드)
CryptUIDlgCertMgr 사용자가 인증서를 관리할 수 있는 대화 상자를 표시합니다.
CryptUIDlgSelectCertificateFromStore 지정된 저장소에서 인증서를 선택할 수 있는 대화 상자를 표시합니다.
CryptUIDlgViewCertificateA 지정된 인증서를 표시하는 대화 상자를 표시합니다. (ANSI)
CryptUIDlgViewCertificateW 지정된 인증서를 표시하는 대화 상자를 표시합니다. (유니코드)
CryptUIDlgViewContext 인증서, CTL 또는 CRL 컨텍스트를 표시합니다.
CryptUIWizDigitalSign 문서 또는 BLOB에 디지털 서명합니다.
CryptUIWizExport 인증서, CTL(인증서 신뢰 목록), CRL(인증서 해지 목록) 또는 인증서 저장소를 파일로 내보냅니다.

CryptUIWizFreeDigitalSignContext CryptUIWizDigitalSign 함수에 의해 할당된 CRYPTUI_WIZ_DIGITAL_SIGN_CONTEXT 구조를 해제합니다.
CryptUIWizImport 인증서, CTL(인증서 신뢰 목록), CRL(인증서 해지 목록) 또는 인증서 저장소를 인증서 저장소로 가져옵니다.
CryptUninstallDefaultContext 중요 이 API는 더 이상 사용되지 않습니다. (CryptUninstallDefaultContext)
CryptUnprotectData DATA_BLOB 구조에서 데이터의 무결성 검사를 해독하고 수행합니다.
CryptUnprotectMemory CryptProtectMemory 함수를 사용하여 암호화된 메모리의 암호를 해독합니다.
CryptUnregisterDefaultOIDFunction CryptUnregisterDefaultOIDFunction은 지정된 인코딩 형식 및 함수 이름에 대해 호출할 기본 함수를 포함하는 DLL의 등록을 제거합니다.
CryptUnregisterOIDFunction 지정된 인코딩 형식, 함수 이름 및 OID에 대해 호출할 함수가 포함된 DLL 등록을 제거합니다.
CryptUnregisterOIDInfo CryptUnregisterOIDInfo 함수는 지정된 CRYPT_OID_INFO OID 정보 구조의 등록을 제거합니다. 등록 취소할 구조체는 구조체의 pszOID 및 dwGroupId 멤버로 식별됩니다.
CryptUpdateProtectedState 사용자의 SID(보안 식별자)가 변경된 후 현재 사용자의 마스터 키를 마이그레이션합니다.
CryptVerifyCertificateSignature CERT_PUBLIC_KEY_INFO 구조에서 공개 키를 사용하여 인증서, CRL(인증서 해지 목록) 또는 인증서 요청의 서명을 확인합니다.
CryptVerifyCertificateSignatureEx 발급자의 공개 키를 사용하여 주체 인증서, 인증서 해지 목록, 인증서 요청 또는 키 생성 요청의 서명을 확인합니다.

CryptVerifyDetachedMessageHash CryptVerifyDetachedMessageHash 함수는 분리된 해시를 확인합니다.
CryptVerifyDetachedMessageSignature CryptVerifyDetachedMessageSignature 함수는 분리된 서명 또는 서명이 포함된 서명된 메시지를 확인합니다.
CryptVerifyMessageHash CryptVerifyMessageHash 함수는 지정된 콘텐츠의 해시를 확인합니다.
CryptVerifyMessageSignature 서명된 메시지의 서명을 확인합니다.
CryptVerifyMessageSignatureWithKey 지정된 공개 키 정보를 사용하여 서명된 메시지의 서명을 확인합니다.
CryptVerifySignatureA 해시 개체의 서명을 확인합니다. (ANSI)
CryptVerifySignatureW 해시 개체의 서명을 확인합니다. (유니코드)
CryptVerifyTimeStampSignature 지정된 바이트 배열의 타임스탬프를 확인합니다.
CryptXmlAddObject 인코딩을 위해 열린 문서 컨텍스트의 서명에 Object 요소를 추가합니다.
CryptXmlClose 암호화 XML 개체 핸들을 닫습니다.
CryptXmlCreateReference XML 서명에 대한 참조를 만듭니다.
CryptXmlDigestReference 애플리케이션에서 확인된 참조를 다이제스트하는 데 사용됩니다. 이 함수는 다이제스트를 업데이트하기 전에 변환을 적용합니다.

CryptXmlDllCloseDigest CryptXmlDllCreateDigest 함수에 의해 할당된 CRYPT_XML_DIGEST 해제합니다.
CryptXmlDllCreateDigest 지정된 메서드에 대한 다이제스트 개체를 만듭니다.
CryptXmlDllCreateKey KeyValue 요소를 구문 분석하고 CNG(Cryptography API:_Next Generation) BCrypt 키 핸들을 만들어 서명을 확인합니다.
CryptXmlDllDigestData 다이제스트에 데이터를 넣습니다.
CryptXmlDllEncodeAlgorithm 기본 매개 변수를 사용하여 Agile 알고리즘에 대한 SignatureMethod 또는 DigestMethod 요소를 인코딩합니다.
CryptXmlDllEncodeKeyValue KeyValue 요소를 인코딩합니다.
CryptXmlDllFinalizeDigest 다이제스트 값을 검색합니다.
CryptXmlDllGetAlgorithmInfo XML 알고리즘을 디코딩하고 알고리즘에 대한 정보를 반환합니다.
CryptXmlDllGetInterface 지정된 알고리즘에 대한 암호화 확장 함수에 대한 포인터를 검색합니다.
CryptXmlDllSignData 데이터에 서명합니다. (CryptXmlDllSignData)
CryptXmlDllVerifySignature 서명을 확인합니다.
CryptXmlEncode 제공된 XML 기록기 콜백 함수를 사용하여 서명 데이터를 인코딩합니다.

CryptXmlGetAlgorithmInfo CRYPT_XML_ALGORITHM 구조를 디코딩하고 알고리즘에 대한 정보를 반환합니다.
CryptXmlGetDocContext 제공된 핸들에서 지정한 문서 컨텍스트를 반환합니다.
CryptXmlGetReference 제공된 핸들에서 지정한 Reference 요소를 반환합니다.
CryptXmlGetSignature XML Signature 요소를 반환합니다.
CryptXmlGetStatus 제공된 핸들에서 지정한 개체에 대한 상태 정보를 포함하는 CRYPT_XML_STATUS 구조를 반환합니다.
CryptXmlGetTransforms 기본 변환 체인 엔진에 대한 정보를 반환합니다.
CryptXmlImportPublicKey 제공된 핸들에서 지정한 공개 키를 가져옵니다.
CryptXmlOpenToDecode XML 디지털 서명을 열어 디코딩하고 CRYPT_XML_SIGNATURE 구조를 캡슐화하는 문서 컨텍스트의 핸들을 반환합니다. 문서 컨텍스트에는 하나 이상의 서명 요소가 포함될 수 있습니다.
CryptXmlOpenToEncode XML 디지털 서명을 열어 인코딩하고 열린 Signature 요소의 핸들을 반환합니다. 핸들은 단일 CRYPT_XML_SIGNATURE 구조로 문서 컨텍스트를 캡슐화하고 CryptXmlClose 함수가 호출될 때까지 열린 상태로 유지됩니다.
CryptXmlSetHMACSecret CryptXmlSign 또는 CryptXmlVerify 함수를 호출하기 전에 핸들에서 HMAC 비밀을 설정합니다.
CryptXmlSign SignedInfo 요소의 암호화 서명을 만듭니다.
CryptXmlVerifySignature

SignedInfo 요소의 암호화 서명 유효성 검사를 수행합니다.
DdqCancelDiagnosticRecordOperation 이 세션에 대한 모든 미해결 진단 데이터 쿼리 API 내부 쿼리 작업을 취소합니다. 이는 장기 실행 쿼리 API를 중단하기 위해 다른 스레드에서 호출할 수 있습니다.
DdqCloseSession 진단 데이터 쿼리 세션 핸들을 닫습니다.
DdqCreateSession 진단 데이터 쿼리 세션을 고유하게 식별하는 데 사용할 진단 데이터 쿼리 API 세션 핸들을 만듭니다.
DdqExtractDiagnosticReport Windows 오류 보고 보고서를 검색하는 데 사용되는 이 API는 지정된 대상 경로로 cab을 추출합니다. 오류 보고서에 cabs가 없으면 작업이 수행되지 않습니다.
DdqFreeDiagnosticRecordLocaleTags HDIAGNOSTIC_EVENT_TAG_DESCRIPTION 핸들에서 참조하는 태그 정보에 할당된 메모리를 해제합니다.
DdqFreeDiagnosticRecordPage HDIAGNOSTIC_RECORD 핸들에서 참조하는 진단 레코드 페이지에 할당된 메모리를 해제합니다.
DdqFreeDiagnosticRecordProducerCategories 범주 집합 및 HDIAGNOSTIC_EVENT_CATEGORY_DESCRIPTION 핸들에서 참조하는 범주의 텍스트 표현에 할당된 메모리를 해제합니다.
DdqFreeDiagnosticRecordProducers HDIAGNOSTIC_EVENT_PRODUCER_DESCRIPTION 핸들에서 참조하는 생산자 집합에 할당된 메모리를 해제합니다.
DdqFreeDiagnosticReport HDIAGNOSTIC_REPORT_DATA 핸들에서 참조하는 오류 보고서에 할당된 메모리를 해제합니다.
DdqGetDiagnosticDataAccessLevelAllowed API 호출자에 대해 사용 가능한 가장 높은 데이터 액세스 수준을 반환합니다. NoData, CurrentUserData 또는 AllUserData일 수 있습니다.
DdqGetDiagnosticRecordAtIndex

HDIAGNOSTIC_DATA_RECORD 핸들이 가리키는 리소스의 지정된 인덱스에서 진단 데이터 레코드 정보를 가져옵니다.
DdqGetDiagnosticRecordBinaryDistribution 총 예상 업로드 크기에 따라 상위 N개의 시끄러운 이진 파일에 대한 진단 데이터 이벤트 볼륨의 이진 이름 및 관련 예상 총 업로드를 바이트 단위로 가져옵니다. 여기서 N은 topNBinaries에 전달된 값입니다.
DdqGetDiagnosticRecordCategoryAtIndex HDIAGNOSTIC_DATA_EVENT_CATEGORY_DESCRIPTION 핸들이 가리키는 리소스의 지정된 인덱스에서 진단 레코드 범주를 가져옵니다.
DdqGetDiagnosticRecordCategoryCount HDIAGNOSTIC_EVENT_CATEGORY_DESCRIPTION 핸들이 가리키는 리소스에서 진단 레코드 범주의 수(크기)를 가져옵니다.
DdqGetDiagnosticRecordCount HDIAGNOSTIC_DATA_RECORD 핸들이 가리키는 리소스의 요소 수(크기)를 가져옵니다.
DdqGetDiagnosticRecordLocaleTagAtIndex HDIAGNOSTIC_DATA_EVENT_TAG_DESCRIPTION 핸들이 가리키는 리소스의 지정된 인덱스에서 태그 설명을 가져옵니다.
DdqGetDiagnosticRecordLocaleTagCount HDIAGNOSTIC_DATA_EVENT_TAG_DESCRIPTION 핸들이 가리키는 리소스의 태그 수(크기)를 가져옵니다.
DdqGetDiagnosticRecordLocaleTags 지정된 로캘에서 알려진 모든 태그에 대한 정보를 가져오고 데이터에 대한 핸들(HDIAGNOSTIC_EVENT_TAG_DESCRIPTION)을 제공합니다. 예를 들어 로캘은 "en-US"입니다. 반환 값의 예는 태그: 11, 이름: "디바이스 연결 및 구성" 및 설명과 같은 데이터를 포함하는 DIAGNOSTIC_EVENT_TAG_DESCRIPTION 리소스입니다. "디바이스 식별자(예: IP 주소) 구성, 설정 및 성능을 포함하여 서비스와 네트워크에 연결된 디바이스의 연결 및 구성을 설명하는 데이터".
DdqGetDiagnosticRecordPage 필터링된 레코드의 페이지(일괄 처리)를 가져옵니다. 반환된 레코드에 대한 필터링은 searchCriteria, pageRecordCount, offset 및 baseRowId와 DIAGNOSTIC_DATA_SEARCH_CRITERIA 입력 매개 변수를 사용하여 내부적으로 수행됩니다.
DdqGetDiagnosticRecordPayload rowId로 지정된 이벤트 레코드의 페이로드 텍스트를 가져옵니다.

DdqGetDiagnosticRecordProducerAtIndex HDIAGNOSTIC_EVENT_PRODUCER_DESCRIPTION 핸들이 가리키는 리소스의 지정된 인덱스에서 생산자에 대한 설명을 가져옵니다.
DdqGetDiagnosticRecordProducerCategories 생산자와 범주에는 계층적 관계가 있습니다. 즉, 범주는 생산자에 속합니다. 이 함수는 지정된 진단 생산자 이름에 사용할 수 있는 범주 ID 및 범주의 텍스트 표현을 가져옵니다.
DdqGetDiagnosticRecordProducerCount HDIAGNOSTIC_EVENT_PRODUCER_DESCRIPTION 가리키는 리소스의 생산자 수(크기)를 가져옵니다.
DdqGetDiagnosticRecordProducers 진단 데이터 쿼리 세션에 사용할 수 있는 진단 데이터 생산자를 가져옵니다.
DdqGetDiagnosticRecordStats 필터링된 이벤트 기록 진단 데이터 레코드 통계를 가져옵니다. 반환된 통계에 대한 필터링은 입력 매개 변수 DIAGNOSTIC_DATA_SEARCH_CRITERIA 필터를 사용하여 수행됩니다. 레코드 상태는 사용 가능한 검색 조건과 일치하는 레코드 수를 설명하고 데이터의 추가 쿼리에 사용되는 매개 변수를 반환합니다. 이 API의 사용 중 하나는 데이터를 마지막으로 쿼리한 이후 변경 내용이 있는지 확인하는 것입니다. 출력 매개 변수를 변경하면 이벤트 기록 레코드 상태의 상태가 변경되었음을 나타냅니다.
DdqGetDiagnosticRecordSummary 생산자가 필터링할 수 있는 진단 데이터 레코드에 대한 일반 통계를 가져옵니다.
DdqGetDiagnosticRecordTagDistribution 지정된 생산자 이름에 따라 개인 정보 태그 이벤트 배포 통계당 진단 데이터 이벤트를 가져옵니다.
DdqGetDiagnosticReport HDIAGNOSTIC_REPORT_DATA 핸들을 통해 이 PC에서 업로드하거나 큐에 추가한 오류 보고서를 가져옵니다.
DdqGetDiagnosticReportAtIndex HDIAGNOSTIC_REPORT_DATA 핸들이 가리키는 리소스의 지정된 인덱스에서 오류 보고서 및 해당 정보를 가져옵니다.
DdqGetDiagnosticReportCount HDIAGNOSTIC_REPORT_DATA 핸들에서 가리키는 리소스의 오류 보고서 수(크기)를 가져옵니다.

DdqGetDiagnosticReportStoreReportCount 요청된 저장소에 저장된 보고서의 수(크기)를 가져옵니다.
DdqGetSessionAccessLevel 현재 진단 데이터 쿼리 세션의 데이터 액세스 수준을 반환합니다.
DdqGetTranscriptConfiguration 최대 스토리지 크기 및 데이터 기록 시간과 같은 이벤트 기록 구성을 가져옵니다.
DdqIsDiagnosticRecordSampledIn 이벤트에 대한 디바이스의 샘플링된 상태를 가져옵니다.
DdqSetTranscriptConfiguration 최대 스토리지 크기 및 데이터 기록 시간과 같은 이벤트 기록 구성을 설정합니다. 사용자가 상승되지 않으면 구성 설정이 실패합니다.
디코딩 유니코드로 인코딩된 고유 이름에서 개체를 초기화합니다.
디코딩 ASN.1(Abstract Syntax Notation One)으로 인코딩된 대체 이름 확장을 디코딩하고 결과 문자열 배열을 CertEncodeAltName 개체에 저장합니다.
디코딩 ASN.1(Abstract Syntax Notation One)으로 인코딩된 비트 문자열을 디코딩하고 결과 비트 문자열을 이 개체에 저장합니다.
디코딩 추상 구문 표기법 1(ASN.1)로 인코딩된 CRL(인증서 해지 목록) 배포 정보 확장을 디코딩하고 결과 배열을 COM 개체에 저장합니다.
디코딩 ASN.1(Abstract Syntax Notation One)으로 인코딩된 날짜 배열을 디코딩하고 결과 날짜 값 배열을 CertEncodeDateArray 개체에 저장합니다.
디코딩 ASN.1(Abstract Syntax Notation One)으로 인코딩된 Long 배열을 디코딩하고 결과 Long 값 배열을 CertEncodeLongArray 개체에 저장합니다.

디코딩 ASN.1(Abstract Syntax Notation One)으로 인코딩된 문자열 배열을 디코딩하고 결과 문자열 배열을 CertEncodeStringArray 개체에 저장합니다.
DecryptChallenge CMS(CMS) 응답을 통해 인증서 관리에서 챌린지의 암호를 해독하고 CA에 보낼 다시 암호화된 응답을 만듭니다.
DecryptMessage 다이제스트를 사용하여 메시지의 암호를 해독합니다.
DelegateSecurityContext 지정된 서버에 보안 컨텍스트를 위임합니다.
삭제하다 ID 저장소에서 지정된 ID 또는 ID에서 지정된 속성을 제거합니다.
삭제하다 AzAuthorizationStore 개체에서 현재 사용 중인 정책 저장소를 삭제합니다.
삭제하다 CSP(암호화 서비스 공급자) 또는 CNG(Cryptography API:_Next Generation) KSP(키 스토리지 공급자)의 핸들을 해제하고 디스크 또는 스마트 카드에서 키를 삭제합니다.
DeleteAce ACL(액세스 제어 목록)에서 ACE(액세스 제어 항목)를 삭제합니다.
DeleteApplication AzAuthorizationStore 개체에서 지정된 이름의 IAzApplication 개체를 제거합니다.
DeleteApplicationGroup IAzApplication 개체에서 지정된 이름의 IAzApplicationGroup 개체를 제거합니다.
DeleteApplicationGroup AzAuthorizationStore 개체에서 지정된 이름의 IAzApplicationGroup 개체를 제거합니다.
DeleteApplicationGroup

IAzScope 개체에서 지정된 이름의 IAzApplicationGroup 개체를 제거합니다.
DeleteAppMember 이 애플리케이션 그룹에 속한 애플리케이션 그룹 목록에서 지정된 IAzApplicationGroup 개체를 제거합니다.
DeleteAppMember 역할에 속한 애플리케이션 그룹 목록에서 지정된 IAzApplicationGroup 개체를 제거합니다.
DeleteAppNonMember 이 애플리케이션 그룹의 멤버 자격이 거부된 애플리케이션 그룹 목록에서 지정된 IAzApplicationGroup 개체를 제거합니다.
DeleteCAConfiguration 구성 집합에서 명명된 CA(인증 기관) 구성을 제거합니다.
DeleteDelegatedPolicyUser IAzApplication::D eleteDelegatedPolicyUser 메서드는 위임된 정책 사용자 역할을 하는 보안 주체 목록에서 텍스트 형식의 지정된 보안 식별자를 제거합니다.
DeleteDelegatedPolicyUser 위임된 정책 사용자 역할을 하는 보안 주체 목록에서 텍스트 형식의 지정된 SID(보안 식별자)를 제거합니다.
DeleteDelegatedPolicyUserName 위임된 정책 사용자 역할을 하는 보안 주체 목록에서 지정된 계정 이름을 제거합니다. (IAzApplication.DeleteDelegatedPolicyUserName)
DeleteDelegatedPolicyUserName 위임된 정책 사용자 역할을 하는 보안 주체 목록에서 지정된 계정 이름을 제거합니다. (IAzAuthorizationStore.DeleteDelegatedPolicyUserName)
DeleteMember 애플리케이션 그룹에 속한 계정 목록에서 텍스트 형식의 지정된 SID(보안 식별자)를 제거합니다.
DeleteMember 역할에 속한 Windows 계정 목록에서 텍스트 형식의 지정된 SID(보안 식별자)를 제거합니다.
DeleteMemberName

애플리케이션 그룹에 속한 계정 목록에서 지정된 계정 이름을 제거합니다.
DeleteMemberName 역할에 속한 계정 목록에서 지정된 계정 이름을 제거합니다.
DeleteNonMember 애플리케이션 그룹의 멤버 자격이 거부된 계정 목록에서 텍스트 형식의 지정된 SID(보안 식별자)를 제거합니다.
DeleteNonMemberName 애플리케이션 그룹의 멤버 자격이 거부된 계정 목록에서 지정된 계정 이름을 제거합니다.
DeleteOperation IAzApplication 개체에서 지정된 이름의 IAzOperation 개체를 제거합니다.
DeleteOperation 역할에서 지정된 이름의 IAzOperation 개체를 제거합니다.
DeleteOperation 작업에서 지정된 이름의 IAzOperation 개체를 제거합니다.
DeletePolicyAdministrator IAzApplication의 DeletePolicyAdministrator 메서드는 정책 관리자 역할을 하는 보안 주체 목록에서 텍스트 형식의 지정된 보안 식별자를 제거합니다.
DeletePolicyAdministrator 정책 관리자 역할을 하는 보안 주체 목록에서 텍스트 형식의 지정된 SID(보안 식별자)를 제거합니다.
DeletePolicyAdministrator IAzScope의 DeletePolicyAdministrator 메서드는 정책 관리자 역할을 하는 보안 주체 목록에서 텍스트 형식의 지정된 보안 식별자를 제거합니다.
DeletePolicyAdministratorName 정책 관리자 역할을 하는 보안 주체 목록에서 지정된 계정 이름을 제거합니다. (IAzApplication.DeletePolicyAdministratorName)
DeletePolicyAdministratorName

정책 관리자 역할을 하는 보안 주체 목록에서 지정된 계정 이름을 제거합니다. (IAzAuthorizationStore.DeletePolicyAdministratorName)
DeletePolicyAdministratorName IAzScope의 DeletePolicyAdministratorName 메서드는 정책 관리자 역할을 하는 보안 주체 목록에서 지정된 계정 이름을 제거합니다.
DeletePolicyReader IAzApplication의 DeletePolicyReader 메서드는 정책 판독기 역할을 하는 보안 주체 목록에서 텍스트 형식의 지정된 보안 식별자를 제거합니다.
DeletePolicyReader 정책 판독기 역할을 하는 보안 주체 목록에서 텍스트 형식의 지정된 SID(보안 식별자)를 제거합니다.
DeletePolicyReader IAzScope의 DeletePolicyReader 메서드는 정책 판독기 역할을 하는 보안 주체 목록에서 텍스트 형식의 지정된 보안 식별자를 제거합니다.
DeletePolicyReaderName 정책 판독기 역할을 하는 보안 주체 목록에서 지정된 계정 이름을 제거합니다. (IAzApplication.DeletePolicyReaderName)
DeletePolicyReaderName 정책 판독기 역할을 하는 보안 주체 목록에서 지정된 계정 이름을 제거합니다. (IAzAuthorizationStore.DeletePolicyReaderName)
DeletePolicyReaderName IAzScope의 DeletePolicyReaderName 메서드는 정책 판독기 역할을 하는 보안 주체 목록에서 지정된 계정 이름을 제거합니다.
DeleteProperty 속성 집합에서 명명된 속성을 제거합니다.
DeletePropertyItem 지정된 보안 주체 목록에서 지정된 보안 주체를 제거합니다. (IAzApplication.DeletePropertyItem)
DeletePropertyItem 지정된 목록에서 지정된 엔터티를 제거합니다. (IAzApplicationGroup.DeletePropertyItem)

DeletePropertyItem 지정된 보안 주체 목록에서 지정된 보안 주체를 제거합니다. (IAzAuthorizationStore.DeletePropertyItem)
DeletePropertyItem 지정된 목록에서 지정된 엔터티를 제거합니다. (IAzRole.DeletePropertyItem)
DeletePropertyItem 지정된 보안 주체 목록에서 지정된 보안 주체를 제거합니다. (IAzScope.DeletePropertyItem)
DeletePropertyItem 지정된 목록에서 지정된 엔터티를 제거합니다. (IAzTask.DeletePropertyItem)
DeleteRequest 요청에 대해 만든 인증서 또는 키를 삭제합니다.
DeleteRole IAzApplication 개체에서 지정된 이름의 IAzRole 개체를 제거합니다.
DeleteRole IAzScope 개체에서 지정된 이름의 IAzRole 개체를 제거합니다.
DeleteRoleAssignment IAzApplication3 개체에서 지정된 IAzRoleAssignment 개체를 제거합니다.
DeleteRoleAssignment 이 범위에서 지정된 IAzRoleAssignment 개체를 제거합니다.
DeleteRoleDefinition IAzApplication3 개체에서 지정된 IAzRoleDefinition 개체를 제거합니다.
DeleteRoleDefinition 이 IAzRoleAssignment 개체에서 지정된 이름의 IAzRoleDefinition 개체를 제거합니다.
DeleteRoleDefinition 이 IAzRoleDefinition 개체에서 지정된 이름의 IAzRoleDefinition 개체를 제거합니다.
DeleteRoleDefinition

이 범위에서 지정된 IAzRoleDefinition 개체를 제거합니다.
DeleteRow DeleteRow 메서드는 데이터베이스 테이블에서 행 또는 행 집합을 삭제합니다. 호출자는 데이터베이스 테이블과 행 ID 또는 종료 날짜를 지정합니다.
DeleteScope IAzApplication 개체에서 지정된 이름의 IAzScope 개체를 제거합니다.
DeleteScope2 IAzApplication3 개체에서 지정된 IAzScope2 개체를 제거합니다.
DeleteSecurityContext InitializeSecurityContext(일반) 함수 또는 AcceptSecurityContext(일반) 함수에 대한 이전 호출에서 시작한 지정된 보안 컨텍스트와 연결된 로컬 데이터 구조를 삭제합니다.
DeleteSecurityPackageA Microsoft Negotiate에서 지원하는 공급자 목록에서 보안 지원 공급자를 삭제합니다. (ANSI)
DeleteSecurityPackageW Microsoft Negotiate에서 지원하는 공급자 목록에서 보안 지원 공급자를 삭제합니다. (유니코드)
DeleteService 서비스 제어 관리자 데이터베이스에서 삭제할 지정된 서비스를 표시합니다.
DeleteTask IAzApplication 개체에서 지정된 이름의 IAzTask 개체를 제거합니다.
DeleteTask 역할에서 지정된 이름의 IAzTask 개체를 제거합니다.
DeleteTask IAzScope 개체에서 지정된 이름의 IAzTask 개체를 제거합니다.
DeleteTask 작업에서 지정된 이름의 IAzTask 개체를 제거합니다.
DenyRequest

보류 중인 지정된 인증서 요청을 거부합니다.
DeriveCapabilitySidsFromName 이 함수는 기능 이름에서 두 개의 SID 배열을 생성합니다. 하나는 NT 기관을 사용하는 배열 그룹 SID이고, 다른 하나는 AppAuthority를 사용하는 기능 SID의 배열입니다.
DestroyPrivateObjectSecurity 개인 개체의 보안 설명자를 삭제합니다.
DestroyVirtualSmartCard 지정된 인스턴스 ID가 있는 TPM 가상 스마트 카드를 삭제합니다.
DisassociateIdentity 로컬 사용자 계정에서 지정된 ID를 연결 해제합니다.
DisconnectIdentity 현재 도메인 사용자로부터 온라인 ID 연결을 끊습니다.
DSCreateISecurityInfoObject 지정된 DS(디렉터리 서비스) 개체와 연결된 ISecurityInformation 인터페이스의 인스턴스를 만듭니다.
DSCreateISecurityInfoObjectEx 지정된 서버에서 지정된 DS(디렉터리 서비스) 개체와 연결된 ISecurityInformation 인터페이스의 인스턴스를 만듭니다.
DSCreateSecurityPage Active Directory 개체에 대한 보안 속성 페이지를 만듭니다.
DSEditSecurity DS(Directory Services) 개체의 보안을 편집하기 위한 모달 대화 상자를 표시합니다.
DuplicateToken 이미 존재하는 토큰을 복제하는 새 액세스 토큰을 만듭니다.
DuplicateTokenEx 기존 토큰을 복제하는 새 액세스 토큰을 만듭니다. 이 함수는 기본 토큰 또는 가장 토큰을 만들 수 있습니다.

EditSecurity 기본 보안 속성 페이지가 포함된 속성 시트를 표시합니다. 이 속성 페이지를 사용하면 사용자가 개체의 DACL에서 ACE가 허용하거나 거부하는 액세스 권한을 보고 편집할 수 있습니다.
EditSecurityAdvanced 기본 보안 속성 페이지가 포함된 속성 시트를 표시할 때 보안 페이지 형식을 포함하도록 EditSecurity 함수를 확장합니다.
인코딩하 고유 이름을 포함하는 문자열에서 개체를 초기화합니다.
인코딩하 인증서 요청에 서명하고 인코딩하고 키 쌍이 없으면 만듭니다.
인코딩하 이 개체에 저장된 대체 이름 배열의 ASN.1로 인코딩된 문자열을 반환합니다. 개체의 이름은 인코딩되지 않습니다.
인코딩하 지정된 비트 문자열에서 ASN.1(추상 구문 표기법 1) 인코딩을 수행합니다.
인코딩하 COM 개체에 저장된 CRL(인증서 해지 목록) 배포 정보 배열에서 ASN.1(추상 구문 표기법 1) 인코딩을 수행하고 ASN.1로 인코딩된 확장을 반환합니다.
인코딩하 이 개체에 저장된 날짜 배열의 ASN.1(Abstract Syntax Notation One) 인코딩 문자열을 반환합니다.
인코딩하 이 개체에 저장된 LONG 배열의 ASN.1로 인코딩된 문자열을 반환합니다.
인코딩하 이 개체에 저장된 문자열 배열의 ASN.1로 인코딩된 문자열을 반환합니다.
메시지암호화 다이제스트를 사용하여 개인 정보를 제공하도록 메시지를 암호화합니다.

등록 요청을 인코딩하고, 적절한 CA(인증 기관)에 제출하고, 응답을 설치합니다.
등록 인증서 요청을 등록하고 발급된 인증서를 검색합니다.
EnumAlgs ICEnroll4::EnumAlgs 메서드는 현재 CSP(암호화 서비스 공급자)에서 지원하는 지정된 알고리즘 클래스에서 암호화 알고리즘의 ID를 검색합니다.
EnumAlgs 현재 CSP(암호화 서비스 공급자)에서 지원하는 지정된 알고리즘 클래스에서 암호화 알고리즘의 ID를 검색합니다.
EnumCertViewAttribute 행 열거형 시퀀스의 현재 행에 대한 특성 열거형 시퀀스의 인스턴스를 가져옵니다.
EnumCertViewColumn 데이터베이스 스키마에 대한 열거형 시퀀스의 인스턴스를 가져옵니다.
EnumCertViewColumn 행 열거형 시퀀스의 현재 행에 대한 열거형 시퀀스의 인스턴스를 가져옵니다.
EnumCertViewExtension 행 열거형 시퀀스의 현재 행에 대한 확장 열거형 시퀀스의 인스턴스를 가져옵니다.
enumContainers ProviderName 속성으로 지정된 CSP(암호화 서비스 공급자)의 컨테이너 이름을 검색합니다. 이 메서드는 ICEnroll 인터페이스에서 처음 정의되었습니다.
enumContainersWStr ProviderNameWStr 속성으로 지정된 CSP(암호화 서비스 공급자)의 컨테이너 이름을 검색합니다.
EnumDependentServicesA 지정된 서비스에 따라 달라지는 각 서비스의 이름과 상태를 검색합니다. (ANSI)
EnumDependentServicesW

지정된 서비스에 따라 달라지는 각 서비스의 이름과 상태를 검색합니다. (유니코드)
EnumerateAttributes 현재 컨텍스트 내에서 다음 요청 특성의 이름을 반환한 다음 내부 포인터를 다음 특성으로 증분합니다.
EnumerateAttributes 현재 특성의 이름을 검색하고 내부 열거형 포인터를 다음 특성으로 이동합니다.
EnumerateAttributesClose 특성 열거형으로 연결된 모든 리소스를 해제합니다.
EnumerateAttributesClose 특성 열거형으로 연결된 리소스를 해제합니다.
EnumerateAttributesSetup 현재 컨텍스트와 연결된 첫 번째 요청 특성에 대한 내부 열거형 포인터를 초기화합니다. (ICertServerExit.EnumerateAttributesSetup)
EnumerateAttributesSetup 현재 컨텍스트와 연결된 첫 번째 요청 특성에 대한 내부 열거형 포인터를 초기화합니다. (ICertServerPolicy.EnumerateAttributesSetup)
EnumerateExtensions 열거할 다음 인증서 확장의 OID(개체 식별자) 문자열(확장명이라고도 함)을 반환한 다음 내부 포인터를 다음 확장으로 증분합니다.
EnumerateExtensions 현재 확장의 OID(개체 식별자)를 검색하고 내부 열거형 포인터를 다음 확장으로 이동합니다.
EnumerateExtensionsClose 확장 열거형으로 연결된 모든 리소스를 해제합니다.
EnumerateExtensionsClose 확장 열거형으로 연결된 리소스를 해제합니다.
EnumerateExtensionsSetup 현재 컨텍스트와 연결된 첫 번째 인증서 확장에 대한 내부 열거형 포인터를 초기화합니다. (ICertServerExit.EnumerateExtensionsSetup)

EnumerateExtensionsSetup 현재 컨텍스트와 연결된 첫 번째 인증서 확장에 대한 내부 열거형 포인터를 초기화합니다. (ICertServerPolicy.EnumerateExtensionsSetup)
EnumerateIdentities ID 공급자 간에 ID를 열거하는 데 사용할 수 있는 IEnumUnknown 인터페이스 포인터에 대한 포인터를 가져옵니다.
EnumerateSecurityPackagesA 클라이언트에서 사용할 수 있는 보안 패키지에 대한 정보를 제공하는 SecPkgInfo 구조체의 배열을 반환합니다. (ANSI)
EnumerateSecurityPackagesW 클라이언트에서 사용할 수 있는 보안 패키지에 대한 정보를 제공하는 SecPkgInfo 구조체의 배열을 반환합니다. (유니코드)
enumPendingRequest 보류 중인 인증서 요청을 열거하고 각 요청에서 지정된 속성을 검색합니다. 이 메서드는 ICEnroll4 인터페이스에서 처음 정의되었습니다.
enumPendingRequestWStr 보류 중인 인증서 요청을 열거하고 각 요청에서 지정된 속성을 검색합니다.
enumProviders ProviderType 속성에 지정된 사용 가능한 CSP(암호화 서비스 공급자)의 이름을 검색합니다. 이 메서드는 ICEnroll 인터페이스에서 처음 정의되었습니다.
enumProvidersWStr IEnroll4::enumProvidersWStr 메서드는 ProviderType 속성에 지정된 사용 가능한 CSP(암호화 서비스 공급자)의 이름을 검색합니다.
EnumServicesStatusA 지정된 서비스 제어 관리자 데이터베이스에서 서비스를 열거합니다. 각 서비스의 이름과 상태가 제공됩니다. (ANSI)
EnumServicesStatusExA 지정된 서비스 제어 관리자 데이터베이스에서 서비스를 열거합니다. 지정된 정보 수준에 따라 추가 데이터와 함께 각 서비스의 이름과 상태가 제공됩니다. (ANSI)

EnumServicesStatusExW 지정된 서비스 제어 관리자 데이터베이스에서 서비스를 열거합니다. 지정된 정보 수준에 따라 추가 데이터와 함께 각 서비스의 이름과 상태가 제공됩니다. (유니코드)
EnumServicesStatusW 지정된 서비스 제어 관리자 데이터베이스에서 서비스를 열거합니다. 각 서비스의 이름과 상태가 제공됩니다. (유니코드)
EqualDomainSid 두 SID가 동일한 도메인에서 온 것인지 여부를 확인합니다.
EqualPrefixSid 같음의 두 SID(보안 식별자) 접두사 값을 테스트합니다. SID 접두사는 마지막 하위 인증 값을 제외한 전체 SID입니다.
EqualSid 두 개의 SID(보안 식별자) 값이 같은지 테스트합니다. 2개의 SID는 동일하게 간주되려면 정확히 일치해야 합니다.
내보내기 CEP(인증서 등록 정책) 서버와 연결된 템플릿 및 개체 식별자를 버퍼로 내보냅니다.
내보내기 프라이빗 키를 바이트 배열에 복사합니다.
ExportPublicKey 인증 공개 키를 내보냅니다.
ExportPublicKey 비대칭 키 쌍의 공개 키 부분을 내보냅니다.
ExportSecurityContext ExportSecurityContext 함수는 나중에 ImportSecurityContext를 호출하여 다른 프로세스로 가져올 수 있는 보안 컨텍스트의 직렬화된 표현을 만듭니다.
찾기 ISignerCertificate 개체의 인덱스 번호를 검색합니다.

FindByUniqueID 지정된 ID와 연결된 IPropertyStore 인터페이스 인스턴스에 대한 포인터를 검색합니다.
FindFirstFreeAce ACL(액세스 제어 목록)에서 첫 번째 무료 바이트에 대한 포인터를 검색합니다.
freeBuffer FreeBuffer 메서드는 보안 구성 스냅인에 의해 할당된 메모리를 해제합니다.
freeBuffer FreeBuffer 메서드는 첨부 파일 스냅인 확장에 의해 할당된 메모리를 해제합니다.
프리컨텍스트버퍼 보안 패키지 함수의 호출자가 보안 패키지에 의해 할당된 메모리 버퍼를 해제할 수 있도록 합니다.
FreeCredentialsHandle 자격 증명이 더 이상 필요하지 않다는 것을 보안 시스템에 알깁니다.
freeInheritedFromArray GetInheritanceSource 함수에 의해 할당된 메모리를 해제합니다.
freeRequestInfo 세션 식별자가 더 이상 필요하지 않은 경우 해제합니다.
freeRequestInfoBlob freeRequestInfoBlob 메서드는 인증서 컨텍스트를 삭제합니다. 이 메서드는 IEnroll 인터페이스에서 처음 정의되었습니다.
FreeSid AllocateAndInitializeSid 함수를 사용하여 이전에 할당된 SID(보안 식별자)를 해제합니다.
GenerateChallenge SCEP 클라이언트에 챌린지 암호를 발급할지 여부를 정책 결정합니다.
get__NewEnum 컬렉션을 열거하는 데 사용할 수 있는 개체에서 IEnumVARIANT 인터페이스를 검색합니다. 이 속성은

Visual Basic 및 Visual Basic Scripting Edition(VBScript) 내에서 숨겨집니다. (IAzApplicationGroups.get__NewEnum)
get__NewEnum 컬렉션을 열거하는 데 사용할 수 있는 개체에서 IEnumVARIANT 인터페이스를 검색합니다. 이 속성은 Visual Basic 및 Visual Basic Scripting Edition(VBScript) 내에서 숨겨집니다. (IAzApplications.get__NewEnum)
get__NewEnum IAzOperations의 _NewEnum 속성은 컬렉션을 열거하는 데 사용할 수 있는 개체에서 IEnumVARIANT 인터페이스를 검색합니다. 이 속성은 Visual Basic 및 Visual Basic Scripting Edition(VBScript) 내에서 숨겨집니다.
get__NewEnum IAzRoleAssignments 컬렉션을 열거하는 데 사용할 수 있는 개체에서 IEnumVARIANT 인터페이스를 검색합니다. 이 속성은 Visual Basic 및 Visual Basic Scripting Edition(VBScript) 내에서 숨겨집니다.
get__NewEnum IAzRoleDefinitions 컬렉션을 열거하는 데 사용할 수 있는 개체에서 IEnumVARIANT 인터페이스를 검색합니다. 이 속성은 Visual Basic 및 Visual Basic Scripting Edition(VBScript) 내에서 숨겨집니다.
get__NewEnum IAzRoles의 _NewEnum 속성은 컬렉션을 열거하는 데 사용할 수 있는 개체에서 IEnumVARIANT 인터페이스를 검색합니다. 이 속성은 Visual Basic 및 Visual Basic Scripting Edition(VBScript) 내에서 숨겨집니다.
get__NewEnum IAzScopes의 _NewEnum 속성은 컬렉션을 열거하는 데 사용할 수 있는 개체에서 IEnumVARIANT 인터페이스를 검색합니다. 이 속성은 Visual Basic 및 Visual Basic Scripting Edition(VBScript) 내에서 숨겨집니다.
get__NewEnum IAzTasks의 _NewEnum 속성은 컬렉션을 열거하는 데 사용할 수 있는 개체에서 IEnumVARIANT 인터페이스를 검색합니다. 이 속성은 Visual Basic 및 Visual Basic Scripting Edition(VBScript) 내에서 숨겨집니다.
get__NewEnum 컬렉션의 열거자를 검색합니다. (IAlternativeNames.get__NewEnum)
get__NewEnum 컬렉션의 열거자를 검색합니다. (ICertificatePolicies.get__NewEnum)
get__NewEnum

컬렉션의 열거자를 검색합니다. (ICertificationAuthorities.get__NewEnum)
get__NewEnum 컬렉션의 열거자를 검색합니다. (ICertProperties.get__NewEnum)
get__NewEnum 컬렉션의 열거자를 검색합니다. (ICryptAttributes.get__NewEnum)
get__NewEnum 컬렉션의 열거자를 검색합니다. (ICspAlgorithms.get__NewEnum)
get__NewEnum 컬렉션의 열거자를 검색합니다. (ICspInformations.get__NewEnum)
get__NewEnum 컬렉션의 열거자를 검색합니다. (ICspStatuses.get__NewEnum)
get__NewEnum 컬렉션의 열거자를 검색합니다. (IObjectIds.get__NewEnum)
get__NewEnum 컬렉션의 열거자를 검색합니다. (IPolicyQualifiers.get__NewEnum)
get__NewEnum 컬렉션의 열거자를 검색합니다. (ISignerCertificates.get__NewEnum)
get__NewEnum 컬렉션의 열거자를 검색합니다. (ISmimeCapabilities.get__NewEnum)
get__NewEnum 컬렉션의 열거자를 검색합니다. (IX509Attributes.get__NewEnum)
get__NewEnum 컬렉션의 열거자를 검색합니다. (IX509CertificateTemplates.get__NewEnum)
get__NewEnum 컬렉션의 열거자를 검색합니다. (IX509Extensions.get__NewEnum)

get__NewEnum 컬렉션의 열거자를 검색합니다. (IX509NameValuePairs.get__NewEnum)
get__NewEnum 컬렉션의 열거자를 검색합니다. (IX509PolicyServerListManager.get__NewEnum)
get__NewEnum 정보 집합의 열거자를 가져옵니다.
get__NewEnum 구성 집합의 열거자를 가져옵니다.
get__NewEnum 속성 집합의 열거자를 가져옵니다.
get_Algorithm 공개 키 알고리즘에 대한 OID(개체 식별자)를 지정하거나 검색합니다. (가져오기)
get_Algorithm 공개 키 알고리즘의 OID(개체 식별자)를 검색합니다.
get_AlternateSignatureAlgorithm PKCS(Get)의 OID(서명 알고리즘 개체 식별자) 여부를 나타내는 부울 값을 지정하고 검색합니다.
get_AlternateSignatureAlgorithm GetSignatureAlgorithm 메서드가 PKCS(Get)에 대한 OID(불연속 또는 결합 알고리즘 개체 식별자)를 검색해야 하는지 여부를 지정하는 부울 값을 지정하고 검색합니다.
get_AlternateSignatureAlgorithmSet AlternateSignatureAlgorithm 속성이 호출자에 의해 명시적으로 설정되었는지 여부를 지정하는 부울 값을 검색합니다.
get_AlternativeNames 주체 대체 이름의 컬렉션을 검색합니다.
get_ApplicationData

애플리케이션에서 정보를 저장하는 데 사용할 수 있는 불투명 필드를 설정하거나 검색합니다. (IAzApplication.get_ApplicationData)

get_ApplicationData

애플리케이션에서 정보를 저장하는 데 사용할 수 있는 불투명 필드를 설정하거나 검색합니다. (IAzAuthorizationStore.get_ApplicationData)

get_ApplicationData

IAzOperation의 ApplicationData 속성은 애플리케이션에서 정보를 저장하는 데 사용할 수 있는 불투명 필드를 설정하거나 검색합니다. (가져오기)

get_ApplicationData

IAzRole의 ApplicationData 속성은 애플리케이션에서 정보를 저장하는 데 사용할 수 있는 불투명 필드를 설정하거나 검색합니다. (가져오기)

get_ApplicationData

IAzScope의 ApplicationData 속성은 애플리케이션에서 정보를 저장하는 데 사용할 수 있는 불투명 필드를 설정하거나 검색합니다. (가져오기)

get_ApplicationData

IAzTask의 ApplicationData 속성은 애플리케이션에서 정보를 저장하는 데 사용할 수 있는 불투명 필드를 설정하거나 검색합니다. (가져오기)

get_ApplicationGroups

정책 데이터에서 IAzApplicationGroup 개체를 열거하는 데 사용되는 IAzApplicationGroups 개체를 검색합니다. (IAzApplication.get_ApplicationGroups)

get_ApplicationGroups

정책 데이터에서 IAzApplicationGroup 개체를 열거하는 데 사용되는 IAzApplicationGroups 개체를 검색합니다. (IAzAuthorizationStore.get_ApplicationGroups)

get_ApplicationGroups

정책 데이터에서 IAzApplicationGroup 개체를 열거하는 데 사용되는 IAzApplicationGroups 개체를 검색합니다. (IAzScope.get_ApplicationGroups)

get_Applications

정책 저장소에서 IAzApplication 개체를 열거하는 데 사용되는 IAzApplications 개체를 검색합니다.

get_ApplyStoreSacl

권한 부여 저장소를 수정할 때 정책 감사를 생성해야 하는지 여부를 나타내는 값을 설정하거나 검색합니다. (IAzApplication.get_ApplyStoreSacl)
get_ApplyStoreSacl 권한 부여 저장소를 수정할 때 정책 감사를 생성해야 하는지 여부를 나타내는 값을 설정하거나 검색합니다. (IAzAuthorizationStore.get_ApplyStoreSacl)
get_AppMembers 이 애플리케이션 그룹에 속하는 애플리케이션 그룹을 검색합니다.
get_AppMembers 역할에 속하는 애플리케이션 그룹을 검색합니다.
get_AppNonMembers 이 애플리케이션 그룹의 멤버 자격이 거부된 애플리케이션 그룹을 검색합니다.
get_Archived 인증서가 보관되었는지 여부를 지정하는 부울 값을 검색합니다.
get_ArchivedKeyHash 프라이빗 키의 SHA-1 해시를 검색합니다.
get_ArchivePrivateKey CA(인증 기관)에서 프라이빗 키를 보관할지 여부를 나타내는 부울 값을 지정하거나 검색합니다. (가져오기)
get_AttestationEncryptionCertificate 클라이언트에서 EKPUB 및 EKCERT 값을 암호화하는 데 사용되는 인증서입니다. 이 속성은 신뢰할 수 있는 컴퓨터 루트에 연결된 유효한 인증서로 설정해야 합니다. (가져오기)
get_AttestPrivateKey 만든 프라이빗 키를 증명해야 하는 경우 True입니다. 그렇지 않으면 false입니다. true이면 AttestationEncryptionCertificate 속성이 설정된 것으로 예상됩니다. (가져오기)
get_AuthFlags 클라이언트가 CEP(인증서 등록 정책) 서버에 인증하는 데 사용하는 인증 유형을 나타내는 값을 지정하고 검색합니다. (가져오기)
get_AuthorityKeyIdentifier

확장 값을 포함하는 바이트 배열을 검색합니다. (IX509ExtensionAuthorityKeyIdentifier.get_AuthorityKeyIdentifier)
get_AuthzInterfaceClsid UI(사용자 인터페이스)가 애플리케이션별 작업을 수행하는 데 사용하는 인터페이스의 CLSID(클래스 식별자)를 설정하거나 검색합니다. (가져오기)
get_BackedUpTime 인증서가 백업된 날짜 및 시간을 검색합니다.
get_BackedUpValue 인증서가 백업되었는지 여부를 식별하는 부울 값을 검색합니다.
get_BitCount 암호화 키의 길이(비트)를 검색합니다.
get_BizRule 이 애플리케이션 그룹의 멤버 자격을 결정하는 스크립트를 가져오거나 설정합니다. (가져오기)
get_BizRule 비즈니스 규칙(BizRule)을 구현하는 스크립트의 텍스트를 설정하거나 검색합니다. (가져오기)
get_BizRuleImportedPath 이 애플리케이션 그룹과 연결된 비즈니스 규칙 스크립트를 포함하는 파일의 경로를 가져오거나 설정합니다. (가져오기)
get_BizRuleImportedPath 비즈니스 규칙(BizRule)을 가져오는 파일의 경로를 설정하거나 검색합니다. (가져오기)
get_BizRuleInterfaces 이 클라이언트 컨텍스트와 연결된 BizRule(비즈니스 규칙) 스크립트에서 호출할 수 있는 IDispatch 인터페이스의 컬렉션을 가져옵니다.
get_BizRuleLanguage 이 애플리케이션 그룹과 연결된 비즈니스 규칙 스크립트의 프로그래밍 언어를 가져오거나 설정합니다. (가져오기)
get_BizRuleLanguage

비즈니스 규칙(BizRule)이 구현되는 스크립팅 언어를 설정하거나 검색합니다. (가져오기)
get_BizRuleParameters 이 클라이언트 컨텍스트와 연결된 BizRule(비즈니스 규칙) 스크립트에서 전달할 수 있는 매개 변수 컬렉션을 가져옵니다.
get_BizRulesEnabled 이 애플리케이션에 대해 비즈니스 규칙을 사용할 수 있는지 여부를 나타내는 값을 가져오거나 설정합니다. (가져오기)
get_BizrulesWritable 위임되지 않은 범위가 쓰기 가능한지 여부를 나타내는 값을 검색합니다.
get_BusinessRuleString 비즈니스 규칙(BizRule)에 대한 애플리케이션별 문자열을 설정하거나 검색합니다. (가져오기)
get_CACertificate DER(고유 인코딩 규칙)를 사용하여 인코딩되었으며 CA(인증 기관)용인 X.509 인증서를 가져옵니다.
get_CAConfig 서명 인증서에 서명해야 하는 CA(인증 기관) 이름을 가져오거나 설정합니다. (가져오기)
get_CAConfigString 인증서 요청이 제출된 CA(인증 기관)를 식별하는 구성 문자열을 검색합니다.
get_CADnsName CA(인증 기관)의 DNS(Domain Naming System) 이름을 검색합니다.
get_CAErrorId 실패한 CA(인증 기관) 사양과 관련된 추가 오류 정보의 ID를 가져옵니다.
get_CAErrorString 실패한 CA(인증 기관) 사양과 관련된 추가 오류 정보에 대한 문자열 데이터를 가져옵니다.
get_CAName CA(인증 기관)의 CN(일반 이름)을 검색합니다.
get_CanBeDelegated

범위를 위임할 수 있는지 여부를 나타내는 값을 검색합니다.
get_CAStoreFlags 저장소를 열 때 CA(인증 기관) 저장소를 제어하는 플래그를 설정하거나 검색합니다. (가져오기)
get_CAStoreFlags IEnroll4의 CAStoreFlags 속성은 저장소가 열릴 때 CA(인증 기관) 저장소를 제어하는 플래그를 설정하거나 검색합니다. (가져오기)
get_CAStoreName 모든 비"ROOT" 및 비"MY" 인증서가 유지되는 저장소의 이름을 설정하거나 검색합니다. (가져오기)
get_CAStoreNameWStr IEnroll4의 CAStoreNameWStr 속성은 모든 비"ROOT" 및 비"MY" 인증서가 유지되는 저장소의 이름을 설정하거나 검색합니다. (가져오기)
get_CAStoreType CAStoreName 속성에 지정된 저장소에 사용할 저장소 유형을 설정하거나 검색합니다. (가져오기)
get_CAStoreTypeWStr CAStoreNameWStr 속성에 지정된 저장소에 사용할 저장소 유형을 설정하거나 검색합니다. (가져오기)
get_Certificate 인증서를 포함하는 DER(고유 인코딩 규칙)로 인코딩된 바이트 배열을 검색합니다.
get_Certificate 설치된 인증서를 검색합니다.
get_Certificate 프라이빗 키와 연결된 인증서를 포함하는 바이트 배열을 지정하거나 검색합니다. (가져오기)
get_Certificate 요청에 대한 인증서를 가져옵니다.
get_CertificateDescription 인증서에 대한 설명이 포함된 문자열을 지정하거나 검색합니다. (가져오기)
get_CertificateFriendlyName

인증서의 표시 이름을 지정하거나 검색합니다. (가져오기)
get_CertificateFriendlyName 인증서의 이름을 가져오거나 설정합니다. (가져오기)
get_ChallengePassword 챌린지가 있는 요청을 만들 때 사용할 암호입니다. 챌린지 없이 요청을 만들려면 ChallengePassword 속성을 설정하지 마세요. (가져오기)
get_ClientId 요청을 생성한 클라이언트 애플리케이션의 형식을 검색합니다.
get_ClientId 요청을 만든 실행 파일을 식별하는 값을 지정하고 검색합니다. (가져오기)
get_ClientId 클라이언트 ID 요청 특성을 설정하거나 검색합니다. 클라이언트 ID 요청 특성은 인증서 요청의 원본을 나타냅니다. 이 속성은 ICEnroll4 인터페이스에서 처음 정의되었습니다. (가져오기)
get_ClientId ClientId 속성은 클라이언트 ID 요청 특성을 설정하거나 검색합니다. 클라이언트 ID 요청 특성은 인증서 요청의 원본을 나타냅니다. 이 속성은 IEnroll4 인터페이스에서 처음 정의되었습니다. (가져오기)
get_ContainerName 키 컨테이너의 이름을 지정하거나 검색합니다. (가져오기)
get_ContainerName CSP(암호화 서비스 공급자)가 키를 생성, 저장 또는 액세스하는 데 사용하는 이름을 가져오거나 설정합니다. (가져오기)
get_ContainerName ICEnroll4의 ContainerName 속성은 사용할 키 컨테이너의 이름을 설정하거나 검색합니다. (가져오기)
get_ContainerNamePrefix 키 컨테이너의 이름에 추가된 접두사를 지정하거나 검색합니다. (가져오기)
get_ContainerNameWStr 사용할 키 컨테이너의 이름을 설정하거나 검색합니다. (가져오기)

get_Cost 인증서 등록 정책 서버에 연결하기 위한 임의의 비용을 지정하고 검색합니다. (IX509EnrollmentPolicyServer.get_Cost)
get_Cost 인증서 등록 정책 서버에 연결하기 위한 임의의 비용을 지정하고 검색합니다. (IX509PolicyServerUrl.get_Cost)
get_Count 컬렉션의 IAzApplicationGroup 개체 수를 검색합니다.
get_Count 컬렉션의 IAzApplication 개체 수를 검색합니다.
get_Count 비즈니스 규칙(BizRule) 스크립트에서 호출할 수 있는 인터페이스 수를 지정합니다.
get_Count BizRule(비즈니스 규칙) 스크립트에 사용할 수 있는 매개 변수 수를 가져옵니다.
get_Count 컬렉션의 IAzOperation 개체 수를 검색합니다.
get_Count 컬렉션의 IAzRoleAssignments 개체 수를 검색합니다.
get_Count 컬렉션의 IAzRoleDefinitions 개체 수를 검색합니다.
get_Count 컬렉션의 IAzRole 개체 수를 검색합니다.
get_Count 컬렉션의 IAzScope 개체 수를 검색합니다.
get_Count 컬렉션의 IAzTask 개체 수를 검색합니다.

get_Count 컬렉션의 개체 수를 검색합니다. (IAlternativeNames.get_Count)
get_Count 컬렉션의 개체 수를 검색합니다. (ICertificatePolicies.get_Count)
get_Count 컬렉션의 ICertificationAuthority 개체 수를 검색합니다.
get_Count 컬렉션의 속성 수를 검색합니다.
get_Count 컬렉션의 ICryptAttribute 개체 수를 검색합니다.
get_Count 컬렉션의 ICspAlgorithm 개체 수를 검색합니다.
get_Count 컬렉션의 ICspInformation 개체 수를 검색합니다.
get_Count 컬렉션의 ICspStatus 개체 수를 검색합니다.
get_Count 컬렉션의 개체 수를 검색합니다. (IObjectIds.get_Count)
get_Count 컬렉션의 개체 수를 검색합니다. (IPolicyQualifiers.get_Count)
get_Count 컬렉션의 ISignerCertificate 개체 수를 검색합니다.
get_Count 컬렉션의 개체 수를 검색합니다. (ISmimeCapabilities.get_Count)
get_Count

컬렉션의 IX509Attribute 개체 수를 검색합니다.
get_Count 컬렉션의 IX509CertificateTemplate 개체 수를 검색합니다.
get_Count 컬렉션의 IX509Extension 개체 수를 검색합니다.
get_Count 컬렉션의 IX509NameValuePair 개체 수를 검색합니다.
get_Count 컬렉션의 IX509PolicyServerUrl 개체 수를 검색합니다.
get_Count 컬렉션의 ICertSrvSetupKeyInformation 개체 수를 가져옵니다.
get_Count 구성 집합의 CA(인증 기관) 구성 수를 가져옵니다.
get_Count 속성 집합의 속성 수를 가져옵니다.
get_Critical 인증서 확장이 중요한지 여부를 식별하는 부울 값을 지정하고 검색합니다. (가져오기)
get_CriticalExtensions 중요로 표시된 버전 3 인증서 확장을 식별하는 IObjectIds 컬렉션을 검색합니다. (IX509CertificateRequestCmc.get_CriticalExtensions)
get_CriticalExtensions 중요로 표시된 버전 3 인증서 확장을 식별하는 IObjectIds 컬렉션을 검색합니다. (IX509CertificateRequestPkcs10.get_CriticalExtensions)
get_CryptAttributes 선택적 인증서 특성의 ICryptAttributes 컬렉션을 검색합니다. (IX509CertificateRequestCmc.get_CryptAttributes)

get_CryptAttributes 선택적 인증서 특성의 ICryptAttributes 컬렉션을 검색합니다. (IX509CertificateRequestPkcs10.get_CryptAttributes)
get_CspAlgorithm 공급자가 지원하는 알고리즘에 대한 정보를 포함하는 ICspAlgorithm 개체를 검색합니다.
get_CspAlgorithms 공급자가 지원하는 알고리즘에 대한 정보를 포함하는 ICspAlgorithm 인터페이스의 컬렉션을 검색합니다.
get_CspInformation 공급자에 대한 일반 정보를 포함하는 ICspInformation 개체를 검색합니다.
get_CspInformations 요청 개체에서 사용할 수 있는 암호화 공급자의 컬렉션을 지정하고 검색합니다. (가져오기)
get_CspInformations 프라이빗 키와 연결된 공개 키 알고리즘을 지원하는 사용 가능한 암호화 공급자에 대한 정보를 포함하는 ICspInformation 개체의 컬렉션을 지정하거나 검색합니다. (가져오기)
get_CSPName CSP(암호화 서비스 공급자) 또는 KSP(키 스토리지 공급자) 이름을 가져옵니다.
get_CspStatus 프라이빗 키와 연결된 암호화 공급자 및 알고리즘 쌍에 대한 정보를 포함하는 ICspStatus 개체를 지정하거나 검색합니다. (가져오기)
get_CspStatuses 인증서 요청과 연결된 프라이빗 키의 의도된 사용과 일치하는 ICspStatus 개체의 컬렉션을 검색합니다.
get_Default 기본 CEP(인증서 등록 정책) 서버인지 여부를 나타내는 부울 값을 지정하고 검색합니다. (가져오기)
get_DefaultContainer 프라이빗 키가 기본 키 컨테이너를 나타내는지 여부를 지정하는 부울 값을 검색합니다.
get_DefaultLength

키의 기본 길이를 검색합니다.
get_DelegatedPolicyUsers 위임된 정책 사용자 역할을 하는 보안 주체의 SID(보안 식별자)를 텍스트 형식으로 검색합니다.
get_DelegatedPolicyUsers 텍스트 양식에서 위임된 정책 사용자 역할을 하는 보안 주체의 SID(보안 식별자)를 검색합니다.
get_DelegatedPolicyUsersName IAzApplication의 DelegatedPolicyUsersName 속성은 위임된 정책 사용자 역할을 하는 보안 주체의 계정 이름을 검색합니다.
get_DelegatedPolicyUsersName 위임된 정책 사용자 역할을 하는 보안 주체의 계정 이름을 검색합니다.
get_DeleteRequestCert 요청 저장소의 더미 인증서가 삭제되는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (가져오기)
get_DeleteRequestCert IEnroll4의 DeleteRequestCert 속성은 요청 저장소의 더미 인증서가 삭제되는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (가져오기)
get_Description 애플리케이션을 설명하는 주석을 설정하거나 검색합니다. (가져오기)
get_Description 애플리케이션 그룹을 설명하는 주석을 설정하거나 검색합니다. (가져오기)
get_Description 작업을 설명하는 주석을 설정하거나 검색합니다. (가져오기)
get_Description IAzOperation의 Description 속성은 작업을 설명하는 주석을 설정하거나 검색합니다. (가져오기)
get_Description 역할을 설명하는 주석을 설정하거나 검색합니다. (가져오기)
get_Description

범위를 설명하는 주석을 설정하거나 검색합니다. (가져오기)
get_Description 작업을 설명하는 주석을 설정하거나 검색합니다. (가져오기)
get_Description 인증서에 대한 설명을 검색합니다.
get_Description 프라이빗 키에 대한 설명이 포함된 문자열을 지정하거나 검색합니다. (가져오기)
get_Display 사용자 인터페이스에 상태 정보를 표시할지 여부를 나타내는 값을 지정하거나 검색합니다. (가져오기)
get_DisplayName 공급자의 이름, 알고리즘 이름 및 알고리즘에서 수행할 수 있는 작업이 포함된 문자열을 검색합니다.
get_DomainTimeout 도메인에 연결할 수 없는 것으로 결정되는 시간(밀리초)을 설정하거나 검색합니다. (가져오기)
get_EnableSMIMECapabilities ICEnroll4::EnableSMIMECapabilities 속성은 PKCS(Get) 여부를 제어합니다.
get_EnableSMIMECapabilities PKCS(가져오기) 여부를 제어합니다.
get_EnableT61DNEncoding ICEnroll4의 EnableT61DNEncoding 속성은 요청의 고유 이름이 유니코드 문자열이 아닌 T61 문자열로 인코딩되는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (가져오기)
get_EnableT61DNEncoding 요청의 고유 이름이 유니코드 문자열이 아닌 T61 문자열로 인코딩되는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (가져오기)
get_EncodedKey 공개 키가 포함된 바이트 배열을 검색합니다.
get_EncodedName

유니코드로 인코딩된 고유 이름을 검색합니다.
get_EncodedParameters 공개 키 알고리즘과 연결된 매개 변수를 포함하는 바이트 배열을 검색합니다.
get_EncryptedKeyBlob 암호화된 키를 포함하는 바이트 배열을 검색합니다.
get_EncryptedKeyHash 보관할 프라이빗 키의 해시를 검색합니다.
get_EncryptedKeyHashBlob 암호화된 프라이빗 키의 해시를 포함하는 문자열을 검색합니다.
get_EncryptionAlgorithm 프라이빗 키를 암호화하는 데 사용되는 대칭 암호화 알고리즘의 OID(개체 식별자)를 검색합니다.
get_EncryptionAlgorithm 보관할 프라이빗 키를 암호화하는 데 사용되는 알고리즘의 OID(개체 식별자)를 지정하거나 검색합니다. (가져오기)
get_EncryptionAlgorithm 클라이언트에서 EKPUB 및 EKCERT 값을 암호화하는 데 사용되는 암호화 알고리즘입니다. (가져오기)
get_EncryptionStrength 키를 암호화하는 데 사용되는 대칭 알고리즘의 암호화 강도가 포함된 정수를 검색합니다.
get_EncryptionStrength 보관할 프라이빗 키에 적용되는 상대 암호화 수준을 지정하거나 검색합니다. (가져오기)
get_EncryptionStrength 암호화에 사용할 EncryptionAlgorithm의 비트 길이를 식별합니다. EncryptionAlgorithm이 1비트 길이만 지원하는 경우 EncryptionStrength 속성에 대한 값을 지정할 필요가 없습니다. (가져오기)
get_EnhancedKeyUsage OID(키 사용 개체 식별자)의 컬렉션을 검색합니다.
get_EnrollmentContext

인증서가 컴퓨터 또는 사용자용인지 여부를 지정하는 값을 검색합니다.
get_EnrollmentContext 인증서가 컴퓨터용인지 최종 사용자용인지를 식별하는 등록 컨텍스트를 검색합니다.
get_EnrollmentStatus 인증서 등록에 대한 정보가 포함된 IX509EnrollmentStatus 개체를 검색합니다.
get_Error 인증서 등록 프로세스의 오류 상태를 식별하는 값을 지정하고 검색합니다. (가져오기)
get_ErrorCode CA 구성에서 오류 조건을 식별하는 코드를 가져옵니다.
get_ErrorString CEP(인증서 등록 정책) 웹 서비스 설치 실패에 대한 추가 정보가 포함된 문자열을 검색합니다.
get_ErrorString CES(인증서 등록 웹 서비스) 설치 실패에 대한 추가 정보가 포함된 문자열을 검색합니다.
get_ErrorText Error 속성에서 반환된 오류 결과 코드와 연결된 메시지가 포함된 문자열을 검색합니다.
get_Existing 프라이빗 키를 만들거나 가져왔는지 여부를 나타내는 부울 값을 지정하거나 검색합니다. (가져오기)
get_Existing 프라이빗 키가 이미 있는지 여부를 나타내는 값을 가져오거나 설정합니다. (가져오기)
get_ExistingCACertificate DER(고유 인코딩 규칙)을 사용하여 인코딩되었으며 기존 키에 해당하는 CA(인증 기관) 인증서의 이진 값인 이진 값을 가져오거나 설정합니다. (가져오기)
get_ExportPolicy 프라이빗 키에 대한 내보내기 제약 조건을 지정하거나 검색합니다. (가져오기)
get_FailInfo ProcessResponseMessage 메서드가 실패한 환경을 검색할 때 정보를 가져옵니다.

get_Flags CEP(인증서 등록 정책) 서버 정책 정보를 그룹 정책, 레지스트리 또는 둘 다에서 로드할 수 있는지 여부를 나타내는 값을 지정하거나 검색합니다. (가져오기)
get_FriendlyName 인증서의 표시 이름을 검색합니다. (ICertPropertyEnrollment.get_FriendlyName)
get_FriendlyName 인증서의 표시 이름을 검색합니다. (ICertPropertyFriendlyName.get_FriendlyName)
get_FriendlyName 개체 식별자의 표시 이름을 지정하고 검색합니다. (가져오기)
get_FriendlyName 프라이빗 키의 표시 이름을 지정하거나 검색합니다. (가져오기)
get_GenerateAudits IAzApplication의 GenerateAudits 속성은 런타임 감사를 생성해야 하는지 여부를 나타내는 값을 설정하거나 검색합니다. (가져오기)
get_GenerateAudits 런타임 감사를 생성해야 하는지 여부를 나타내는 값을 설정하거나 검색합니다. (가져오기)
get_GenKeyFlags 인증서 요청이 생성될 때 CryptGenKey 함수에 전달된 값을 설정하거나 검색합니다. (가져오기)
get_GenKeyFlags 인증서 요청이 생성될 때 CryptGenKey에 전달된 값을 설정하거나 검색합니다. (가져오기)
get_HashAlgID PKCS(Get)에 서명할 때 사용되는 해시 알고리즘을 설정하거나 검색합니다.
get_HashAlgID IEnroll4의 HashAlgID 속성은 PKCS(Get)에 서명할 때 사용되는 해시 알고리즘을 설정하거나 검색합니다.
get_HashAlgorithm

인증서 요청에 서명하는 데 사용되는 해시 알고리즘의 OID(개체 식별자)를 지정하고 검색합니다. (가져오기)
get_HashAlgorithm GetSignatureAlgorithm 메서드에 사용되는 해시 알고리즘에 대한 OID(개체 식별자)를 지정하고 검색합니다. (가져오기)
get_HashAlgorithm 키에 대한 CA(인증 기관) 인증서에 서명하거나 확인하는 데 사용되는 해시 알고리즘의 이름을 가져오거나 설정합니다. (가져오기)
get_HashAlgorithm PKCS(ICEnroll.get_HashAlgorithm)에 서명하는 데 사용되는 서명 해시 알고리즘만 설정하거나 검색합니다.
get_HashAlgorithm 인증서에 서명하는 데 사용되는 해시 알고리즘의 식별자를 가져오거나 설정합니다. (가져오기)
get_HashAlgorithmWStr PKCS에 서명하는 데 사용되는 서명 해시 알고리즘만 설정하거나 검색합니다 (IEnroll.get_HashAlgorithmWStr).
get_HasHardwareRandomNumberGenerator 공급자가 암호화 작업에 대한 임의 바이트를 만드는 데 사용할 수 있는 하드웨어 난수 생성기를 지원하는지 여부를 지정하는 부울 값을 검색합니다.
get_Identifier CA(인증 기관) 구성의 이름을 가져옵니다.
get_IncludeSubjectKeyID 주체 키 ID 확장이 생성된 인증서 요청에 추가되는지 여부를 결정합니다. (가져오기)
get_IncludeSubjectKeyID IEnroll4의 IncludeSubjectKeyID 속성은 주체 키 ID 확장이 생성된 인증서 요청에 추가되는지 여부를 결정합니다. (가져오기)
get_IncrementLength 여러 키 크기를 지원하는 알고리즘의 유효한 증분 키 길이를 확인하는 데 사용할 수 있는 값을 비트 단위로 검색합니다.

get_IndexByObjectId OID(개체 식별자)를 사용하여 특성의 인덱스를 검색합니다.
get_IndexByObjectId OID(개체 식별자)를 사용하여 ICspAlgorithm 개체의 인덱스를 검색합니다.
get_IndexByObjectId OID(개체 식별자)를 사용하여 컬렉션에서 확장의 인덱스를 검색합니다.
get_IsCA 인증서의 주체가 CA(인증 기관)인지 여부를 식별하는 부울 값을 검색합니다.
get_IsHardwareDevice 공급자가 하드웨어 디바이스에서 구현되는지 여부를 결정하는 부울 값을 검색합니다.
get_IsRemovable 키를 포함하는 토큰을 제거할 수 있는지 여부를 지정하는 부울 값을 검색합니다.
get_IsRoleDefinition 태스크가 역할 정의인지 여부를 나타내는 값을 설정하거나 검색합니다. (가져오기)
get_IsSmartCard 공급자가 스마트 카드 공급자인지 여부를 지정하는 부울 값을 검색합니다.
get_IsSoftwareDevice 공급자가 소프트웨어에서 구현되는지 여부를 지정하는 부울 값을 검색합니다.
get_Issuer 인증서 발급자의 이름을 지정하거나 검색합니다. (가져오기)
get_Item 지정된 인덱스의 IAzApplicationGroup 개체를 IAzApplicationGroups 컬렉션으로 검색합니다.
get_Item 지정된 인덱스의 IAzApplication 개체를 IAzApplications 컬렉션으로 검색합니다.
get_Item

지정된 인덱스의 IAzOperation 개체를 IAzOperations 컬렉션으로 검색합니다.
get_Item IAzRoleAssignments 컬렉션의 지정된 인덱스에서 IAzRoleAssignment 개체를 검색합니다.
get_Item IAzRoleDefinitions 컬렉션의 지정된 인덱스에서 IAzRoleDefinition 개체를 검색합니다.
get_Item 지정된 인덱스의 IAzRole 개체를 IAzRoles 컬렉션으로 검색합니다.
get_Item 지정된 인덱스의 IAzScope 개체를 IAzScopes 컬렉션으로 검색합니다.
get_Item 지정된 인덱스의 IAzTask 개체를 IAzTasks 컬렉션으로 검색합니다.
get_Item 컬렉션의 인덱스로 식별되는 ICertSrvSetupKeyInformation 개체를 가져옵니다.
get_Item 구성 집합의 인덱스로 식별되는 CA(인증 기관) 구성을 가져옵니다.
get_Item 속성 집합의 인덱스로 식별되는 속성을 가져옵니다.
get_ItemByName 인증 기관 이름으로 컬렉션에서 ICertificationAuthority 개체를 검색합니다.
get_ItemByName 이름으로 컬렉션에서 ICspAlgorithm 개체를 검색합니다.
get_ItemByName 이름으로 컬렉션에서 ICspInformation 개체를 검색합니다.
get_ItemByName 공급자 및 알고리즘 이름을 사용하여 컬렉션에서 ICspStatus 개체를 검색합니다.

get_ItemByName 이름으로 컬렉션에서 IX509CertificateTemplate 개체를 검색합니다.
get_ItemByName 구성 집합에서 이름으로 식별되는 CA(인증 기관) 구성을 가져옵니다.
get_ItemByName 속성 집합의 이름으로 식별되는 속성을 가져옵니다.
get_ItemByOid 개체 식별자를 사용하여 컬렉션에서 IX509CertificateTemplate 개체를 검색합니다.
get_ItemByOperations 입력 및 동일한 알고리즘에 지정된 공급자와 이름이 같지만 다른 암호화 작업을 식별하는 ICspStatus 개체를 검색합니다.
get_ItemByOrdinal 컬렉션에서 서수로 ICspStatus 개체를 검색합니다.
get_ItemByProvider 입력에 지정된 공급자와 이름이 같지만 다른 의도한 키 사용을 지원하는 알고리즘을 식별하는 ICspStatus 개체를 검색합니다.
get_KeyArchivalCertificate CA(인증 기관) 암호화 인증서를 지정하거나 검색합니다. (가져오기)
get_KeyContainerNamePrefix 새 프라이빗 키의 컨테이너 이름을 만드는 데 사용되는 접두사를 지정하거나 검색합니다. (가져오기)
get_KeyProtection 사용하기 전에 프라이빗 키를 보호하는 방법을 나타내는 값을 지정하거나 검색합니다. (가져오기)
get_KeySpec 공급자가 지원하는 알고리즘의 의도된 사용을 지정하는 값을 검색합니다.
get_KeySpec

공급자 또는 키 컨테이너에 저장된 키 쌍이 암호화에 사용되는지 또는 콘텐츠 서명에 사용되는지 여부를 식별하는 값을 검색합니다.
get_KeySpec 프라이빗 키를 서명, 암호화 또는 둘 다에 사용할 수 있는지 여부를 식별하는 값을 지정하거나 검색합니다. (가져오기)
get_KeySpec ICEnroll4의 KeySpec 속성은 생성된 키 유형을 설정하거나 검색합니다. (가져오기)
get_KeySpec 생성된 키의 형식을 설정하거나 검색합니다. (가져오기)
get_KeySpec 구성에 바인딩된 키가 암호화 또는 서명 콘텐츠에 사용되는지 여부를 나타내는 값을 가져옵니다.
get_KeyUsage 공개 키에 대한 제한을 검색합니다.
get_KeyUsage 프라이빗 키를 사용할 수 있는 특정 용도를 식별하는 값을 지정하거나 검색합니다. (가져오기)
get_LdapQuery LDAP 쿼리 애플리케이션 그룹에 대한 멤버 자격을 정의하는 데 사용되는 LDAP(Lightweight Directory Access Protocol) 쿼리를 설정하거나 검색합니다. (가져오기)
get_LDAPQueryDN LDAP 쿼리 그룹을 평가하는 동안 사용할 디렉터리 개체의 도메인 이름을 검색하거나 설정합니다. (가져오기)
get_LegacyCsp 공급자가 CNG(Cryptography API:_Next Generation) 공급자인지 또는 CryptoAPI(레거시) CSP인지 여부를 지정하는 부울 값을 검색합니다.
get_LegacyCsp 공급자가 CryptoAPI(레거시) CSP(암호화 서비스 공급자)인지 여부를 나타내는 부울 값을 지정하거나 검색합니다. (가져오기)
get_Length

인증 키의 비트 길이입니다. Open 메서드가 호출된 후에만 이 속성에 액세스할 수 있습니다.
get_Length 프라이빗 키의 길이를 비트 단위로 지정하거나 검색합니다. (가져오기)
get_Length 공개 키의 길이를 검색합니다.
get_Length CSP(암호화 서비스 공급자)에서 지원하는 값 중 하나에 대한 키의 강도를 가져오거나 설정합니다. (가져오기)
get_LimitExchangeKeyToEncipherment AT_KEYEXCHANGE 요청에 디지털 서명 및 비공제 키 사용이 포함되어 있는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (가져오기)
get_LimitExchangeKeyToEncipherment IEnroll4의 LimitExchangeKeyToEncipherment 속성은 AT_KEYEXCHANGE 요청에 디지털 서명 및 비공급 키 사용이 포함되어 있는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (가져오기)
get_LocalRevocationInformation 로컬 컴퓨터의 CRL(인증서 해지 목록)을 가져오거나 설정합니다. (가져오기)
get_LongName 알고리즘의 전체 이름을 검색합니다.
get_MachineContext 로컬 인증서 저장소 컨텍스트를 식별하는 부울 값을 지정하거나 검색합니다. (가져오기)
get_MachineDnsName 요청을 생성한 컴퓨터의 DNS(도메인 이름 시스템) 이름을 검색합니다.
get_MajorVersion 인증서 템플릿의 최소 주 버전 번호를 검색합니다.
get_MaxKeyContainerNameLength 공급자와 연결된 프라이빗 키 컨테이너의 이름에 대해 지원되는 최대 길이를 검색합니다.

get_MaxLength 키에 허용되는 최대 길이를 검색합니다.
get_MaxScriptEngines 캐시할 BizRule(비즈니스 규칙) 스크립트 엔진의 최대 수를 설정하거나 검색합니다. (가져오기)
get_Members 애플리케이션 그룹에 속한 계정의 SID(보안 식별자)를 텍스트 형식으로 검색합니다.
get_Members 역할에 속한 Windows 계정의 SID(보안 식별자)를 텍스트 형식으로 검색합니다.
get_MembersName 애플리케이션 그룹에 속한 계정의 계정 이름을 검색합니다.
get_MembersName 역할에 속한 계정의 계정 이름을 검색합니다.
get_MinLength 키에 허용되는 최소 길이를 검색합니다.
get_MinorVersion 인증서 템플릿의 최소 부 버전 번호를 검색합니다.
get_Modified OCSPCAConfiguration 개체가 만들어진 이후 수정되었는지 여부를 나타내는 값을 가져옵니다.
get_Modified OCSPProperty 개체가 인스턴스화된 이후 수정되었는지 여부를 나타내는 값을 가져옵니다.
get_MSCEPErrorId 실패한 NDES(네트워크 디바이스 등록 서비스) 사양과 관련된 추가 오류 정보의 ID를 가져옵니다. 부모 개체에 대한 모든 메서드 호출은 이 속성을 다시 설정합니다.
get_MSCEPErrorString 실패한 NDES(네트워크 디바이스 등록 서비스) 사양과 관련된 추가 오류 정보에 대한 문자열 데이터를 포함합니다. 부모 개체에 대한 모든 메서드 호출은 이 속성을 다시 설정합니다.

get_MyStoreFlags 내 저장소에 사용되는 레지스트리 위치를 설정하거나 검색합니다. (가져오기)
get_MyStoreFlags 내 저장소에 사용되는 레지스트리 위치를 설정하거나 검색합니다. (가져오기)
get_MyStoreName 연결된 프라이빗 키가 있는 인증서가 보관되는 저장소의 이름을 설정하거나 검색합니다. (가져오기)
get_MyStoreNameWStr IEnroll4의 MyStoreNameWStr 속성은 연결된 프라이빗 키가 있는 인증서가 보관되는 저장소의 이름을 설정하거나 검색합니다. (가져오기)
get_MyStoreType MyStoreName 속성에 지정된 저장소 유형을 설정하거나 검색합니다. (가져오기)
get_MyStoreTypeWStr MyStoreTypeWStr 속성에 지정된 저장소 유형을 설정하거나 검색합니다. (가져오기)
get_Name 애플리케이션의 이름을 설정하거나 검색합니다. (가져오기)
get_Name 애플리케이션 그룹의 이름을 설정하거나 검색합니다. (가져오기)
get_Name IAzObjectPicker 개체의 이름을 가져옵니다.
get_Name 작업의 이름을 설정하거나 검색합니다. (가져오기)
get_Name 역할의 이름을 설정하거나 검색합니다. (가져오기)
get_Name 범위의 이름을 설정하거나 검색합니다. (가져오기)

get_Name 작업의 이름을 설정하거나 검색합니다. (가져오기)
get_Name 축약된 알고리즘 이름을 검색합니다.
get_Name 이름을 검색합니다.
get_Name 개체 식별자를 포함하는 CERTENROLL_OBJECTID 값을 검색합니다.
get_Name 고유 이름을 검색합니다.
get_Name 이름-값 쌍의 이름 부분을 검색합니다.
get_Name OCSPProperty 개체가 나타내는 이름-값 쌍의 식별자 부분을 가져옵니다.
get_NameResolver 이 IAzPrincipalLocator 개체와 연결된 IAzNameResolver 인터페이스에 대한 포인터를 가져옵니다.
get_NameValuePairs 인증서 요청과 연결된 IX509NameValuePairs 컬렉션을 검색합니다.
get_NameValuePairs 추가 인증서 속성 값의 이름/값 쌍 컬렉션입니다.
get_NameValuePairs 등록 개체와 연결된 이름-값 쌍의 컬렉션을 검색합니다.
get_NonMembers 애플리케이션 그룹의 멤버 자격이 거부된 계정의 ID(보안 식별자)를 텍스트 형식으로 검색합니다.
get_NonMembersName

애플리케이션 그룹의 멤버 자격이 거부된 계정의 계정 이름을 검색합니다.
get_NotAfter 인증서가 더 이상 유효하지 않은 날짜와 시간을 지정하거나 검색합니다. (가져오기)
get_NotBefore 인증서가 유효하지 않은 날짜와 시간을 지정하거나 검색합니다. (가져오기)
get_NullSigned 인증서 요청의 기본 서명이 null로 서명되었는지 여부를 지정하는 부울 값을 검색합니다.
get_NullSigned 인증서 요청이 null로 서명되었는지 여부를 나타내는 부울 값을 검색합니다.
get_NullSigned 인증서 요청이 null로 서명되었는지 여부를 나타내는 부울 값을 지정하고 검색합니다. (가져오기)
get_ObjectId 이름과 연결된 OID(개체 식별자)가 있는 경우 검색합니다.
get_ObjectId 정책 개체의 OID(개체 식별자)를 검색합니다.
get_ObjectId 특성의 OID(개체 식별자)를 검색합니다. (ICryptAttribute.get_ObjectId)
get_ObjectId 한정자의 OID(개체 식별자)를 검색합니다.
get_ObjectId 대칭 암호화 알고리즘의 OID(개체 식별자)를 검색합니다.
get_ObjectId 특성의 OID(개체 식별자)를 검색합니다. (IX509Attribute.get_ObjectId)
get_ObjectId 확장에 대한 OID(개체 식별자)를 검색합니다.

get_ObjectPicker 이 IAzPrincipalLocator 개체와 연결된 IAzObjectPicker 인터페이스에 대한 포인터를 가져옵니다.
get_OCSPCAConfigurationCollection OCSPCAConfigurationCollection 개체의 인스턴스를 가져옵니다. 이 개체는 OCSP(온라인 인증서 상태 프로토콜) 응답자 서비스가 상태 요청을 처리할 수 있는 CA(인증 기관) 인증서 집합을 나타냅니다.
get_OCSPServiceProperties OCSPPropertyCollection 개체의 인스턴스를 가져옵니다. 이 개체는 OCSP(온라인 인증서 상태 프로토콜) 응답자 서비스의 특성을 나타냅니다.
get_OldCertificate InitializeFromCertificate 메서드에 전달된 인증서를 검색합니다.
get_OldCertificate 요청을 바꾸려는 이전 인증서를 가져오거나 설정합니다. (가져오기)
get_Opened Open 메서드가 성공적으로 호출되었는지 여부를 나타냅니다.
get_Opened 프라이빗 키가 열려 있는지 여부를 지정하는 부울 값을 검색합니다.
get_OperationID 애플리케이션 내에서 작업을 고유하게 식별하는 애플리케이션별 값을 설정하거나 검색합니다. (가져오기)
get_Operations 정책 데이터에서 IAzOperation 개체를 열거하는 데 사용되는 IAzOperations 개체를 검색합니다.
get_Operations 역할과 연결된 작업을 검색합니다.
get_Operations 작업과 연결된 작업을 검색합니다.
get_Operations 알고리즘에서 수행할 수 있는 작업을 검색합니다.

get_Ordinal ICspStatuses 컬렉션에서 ICspStatus 개체의 위치를 지정하거나 검색합니다. (가져오기)
get_OSVersion 클라이언트 운영 체제 버전 정보를 검색합니다.
get_Parameters 서명 알고리즘과 연결된 매개 변수를 포함하는 바이트 배열을 검색합니다. (가져오기)
get_ParentWindow 서명 인증서 정보를 표시하는 데 사용되는 창의 ID를 지정하거나 검색합니다. (가져오기)
get_ParentWindow 키 관련 사용자 인터페이스 대화 상자에서 사용하는 창의 ID를 지정하고 검색합니다. (가져오기)
get_ParentWindow 등록 정보를 표시하는 데 사용되는 창의 ID를 지정하거나 검색합니다. (가져오기)
get_ParentWindow 키 정보를 표시하는 데 사용되는 창의 ID를 지정하거나 검색합니다. (가져오기)
get_PathLenConstraint 하위 인증 기관 체인의 깊이를 검색합니다.
get_Policies 인증서 정책 컬렉션을 검색합니다.
get_Policies 애플리케이션 정책의 컬렉션을 검색합니다.
get_PolicyAdministrators 정책 관리자 역할을 하는 보안 주체의 SID(보안 식별자)를 텍스트 형식으로 검색합니다.
get_PolicyAdministrators 텍스트 형식의 정책 관리자 역할을 하는 보안 주체의 SID(보안 식별자)를 검색합니다.
get_PolicyAdministrators

IAzScope의 PolicyAdministrators 속성은 정책 관리자 역할을 하는 보안 보안 식별자(SID)를 텍스트 형식으로 검색합니다.
get_PolicyAdministratorsName IAzApplication::P olicyAdministratorsName 속성은 정책 관리자 역할을 하는 보안 주체의 계정 이름을 검색합니다.
get_PolicyAdministratorsName 정책 관리자 역할을 하는 보안 주체의 계정 이름을 검색합니다. (IAzAuthorizationStore.get_PolicyAdministratorsName)
get_PolicyAdministratorsName 정책 관리자 역할을 하는 보안 주체의 계정 이름을 검색합니다. (IAzScope.get_PolicyAdministratorsName)
get_PolicyQualifiers 인증서 정책에 적용할 수 있는 선택적 정책 한정자 컬렉션을 검색합니다.
get_PolicyReaders 정책 판독기 역할을 하는 보안 ID(보안 식별자)를 텍스트 형식으로 검색합니다.
get_PolicyReaders 텍스트 형식의 정책 판독기 역할을 하는 보안 주체의 SID(보안 식별자)를 검색합니다.
get_PolicyReaders IAzScope의 PolicyReaders 속성은 정책 판독기 역할을 하는 보안 주체의 SID(보안 식별자)를 텍스트 형식으로 검색합니다.
get_PolicyReadersName IAzApplication::P olicyReadersName 속성은 정책 판독기 역할을 하는 보안 주체의 계정 이름을 검색합니다.
get_PolicyReadersName 정책 판독기 역할을 하는 보안 주체의 계정 이름을 검색합니다. (IAzAuthorizationStore.get_PolicyReadersName)
get_PolicyReadersName 정책 판독기 역할을 하는 보안 주체의 계정 이름을 검색합니다. (IAzScope.get_PolicyReadersName)
get_PolicyServer

초기화 중에 사용된 템플릿을 포함하는 CEP(인증서 등록 정책) 서버를 검색합니다. (IX509CertificateRequestCertificate2.get_PolicyServer)
get_PolicyServer 초기화 중에 사용된 템플릿을 포함하는 CEP(인증서 등록 정책) 서버를 검색합니다. (IX509CertificateRequestCmc2.get_PolicyServer)
get_PolicyServer 초기화 중에 사용된 템플릿을 포함하는 CEP(인증서 등록 정책) 서버를 검색합니다. (IX509CertificateRequestPkcs10V2.get_PolicyServer)
get_PolicyServer 초기화 중에 사용된 템플릿을 포함하는 CEP(인증서 등록 정책) 서버를 검색합니다. (IX509CertificateRequestPkcs7V2.get_PolicyServer)
get_PolicyServer 초기화 중에 사용된 템플릿을 포함하는 CEP(인증서 등록 정책) 서버를 검색합니다. (IX509Enrollment2.get_PolicyServer)
get_PrivateKey 인증서와 연결된 프라이빗 키를 검색합니다.
get_PrivateKey ISignerCertificate 개체와 연결된 프라이빗 키를 검색합니다.
get_PrivateKey 인증서 요청에 서명하는 데 사용되는 프라이빗 키가 포함된 IX509PrivateKey 개체를 검색합니다.
get_PrivateKeyArchiveCertificate PKCS를 사용하여 프라이빗 키를 보관하는 데 사용되는 인증서를 설정하거나 검색합니다(가져오기).
get_ProcessName 요청을 생성한 애플리케이션의 이름을 검색합니다.
get_Property 인증 기관 속성 값을 검색합니다.
get_Property

템플릿 속성 값을 검색합니다.
get_Property IX509CertificateTemplateWritable 개체의 속성 값을 지정하거나 검색합니다. (가져오기)
get_PropertyId 외부 인증서 속성을 식별하는 CERTENROLL_PROPERTYID 열거형의 값을 지정하거나 검색합니다. (가져오기)
get_ProviderCLSID CA 구성에서 사용하는 해지 정보 공급자의 CLSID를 가져오거나 설정합니다. (가져오기)
get_ProviderFlags 공급자 유형을 설정하거나 검색합니다. (가져오기)
get_ProviderFlags IEnroll4의 ProviderFlags 속성은 공급자 유형을 설정하거나 검색합니다. (가져오기)
get_ProviderName 공급자 이름을 검색합니다.
get_ProviderName 암호화 공급자의 이름입니다. 기본값은 Microsoft 플랫폼 암호화 공급자입니다. Open 메서드를 호출하기 전에 ProviderName 속성을 설정해야 합니다. Open 메서드를 호출한 후에는 ProviderName 속성을 변경할 수 없습니다. (가져오기)
get_ProviderName 암호화 공급자의 이름을 지정하거나 검색합니다. (가져오기)
get_ProviderName 프라이빗 키를 생성하거나 저장하는 데 사용되는 CSP(암호화 서비스 공급자) 또는 KSP(키 스토리지 공급자)의 이름을 가져오거나 설정합니다. (가져오기)
get_ProviderName ICEnroll4의 ProviderName 속성은 사용할 CSP(암호화 서비스 공급자)의 이름을 설정하거나 검색합니다. (가져오기)
get_ProviderNameWStr

사용할 CSP(암호화 서비스 공급자)의 이름을 설정하거나 검색합니다. (가져오기)
get_ProviderProperties 인증서 상태 응답을 제공하는 정보를 가져오거나 설정합니다. (가져오기)
get_ProviderType 프라이빗 키와 연결된 암호화 공급자의 유형을 지정하거나 검색합니다. (가져오기)
get_ProviderType ICEnroll4의 ProviderType 속성은 공급자 유형을 설정하거나 검색합니다. (가져오기)
get_ProviderType 공급자 유형을 설정하거나 검색합니다. (가져오기)
get_PublicKey 인증서 요청에 포함된 공개 키가 포함된 IX509PublicKey 개체를 검색합니다.
get_PublicKeyAlgorithm GetSignatureAlgorithm 메서드에 사용되는 공개 키 알고리즘에 대한 OID(개체 식별자)를 지정하고 검색합니다. (가져오기)
get_PVKFileName ICEnroll4의 PVKFileName 속성은 내보낸 키를 포함할 파일의 이름을 설정하거나 검색합니다. (가져오기)
get_PVKFileNameWStr 내보낸 키를 포함할 파일의 이름을 설정하거나 검색합니다. (가져오기)
get_Qualifier 개체를 초기화하는 데 사용되는 한정자가 포함된 문자열을 검색합니다.
get_RawData 이름이 포함된 DER(고유 인코딩 규칙)로 인코딩된 바이트 배열을 검색합니다.
get_RawData 인증서 속성의 값을 검색합니다.
get_RawData DER(고유 인코딩 규칙) 인코딩 한정자 개체를 검색합니다.

get_RawData 특성 값을 검색합니다.
get_RawData 서명된 DER(고유 인코딩 규칙) 인코딩된 인증서 요청을 포함하는 바이트 배열을 검색합니다.
get_RawData 확장 값을 포함하는 바이트 배열을 검색합니다. (IX509Extension.get_RawData)
get_RawDataToBeSigned 인코딩 메서드에서 만든 서명되지 않은 인증서 요청을 검색합니다.
get_ReaderName 스마트 카드 판독기의 이름을 지정하거나 검색합니다. (가져오기)
get_ReminderDuration 경고 이벤트가 기록된 후 서명 인증서 수명 백분율을 가져오거나 설정합니다. (가져오기)
get_Renewal 새 인증서의 SHA-1 해시를 검색합니다.
get_RenewalCertificate 갱신할 인증서를 검색합니다.
get_RenewalCertificate 갱신 중인 DER(고유 인코딩 규칙) 인코딩 인증서를 포함하는 바이트 배열을 지정하거나 검색합니다. (가져오기)
get_RenewalCertificate 갱신 인증서의 인증서 컨텍스트를 지정합니다. (가져오기)
get_Request 등록 개체와 연결된 인증서 요청을 검색합니다.
get_Request 내부 PKCS10 요청을 가져옵니다.

get_RequesterName 인증서를 요청하는 최종 엔터티의 SAM(보안 계정 관리자) 이름을 포함하는 문자열을 지정하거나 검색합니다. (가져오기)
get_RequestId 고유한 인증서 요청 식별자를 검색합니다.
get_RequestId 등록 메서드를 통해 인증 기관에 전송된 인증서 요청에 대한 고유 식별자를 검색합니다.
get_RequestID CMS(CMS) 응답을 통해 인증서 관리에서 요청 ID를 가져옵니다.
get_RequestIdString CES(인증 등록 서버)로 전송된 인증서 요청에 대한 고유 식별자가 포함된 문자열을 검색합니다.
get_RequestOriginator 원래 컴퓨터의 DNS 이름을 포함하는 문자열을 검색합니다.
get_RequestStoreFlags 요청 저장소에 사용되는 레지스트리 위치를 설정하거나 검색합니다. (가져오기)
get_RequestStoreFlags IEnroll4의 RequestStoreFlags 속성은 요청 저장소에 사용되는 레지스트리 위치를 설정하거나 검색합니다. (가져오기)
get_RequestStoreName 더미 인증서를 포함하는 저장소의 이름을 설정하거나 검색합니다. (가져오기)
get_RequestStoreNameWStr IEnroll4의 RequestStoreNameWStr 속성은 더미 인증서가 포함된 저장소의 이름을 설정하거나 검색합니다. (가져오기)
get_RequestStoreType RequestStoreName 속성에 지정된 저장소에 사용할 저장소 유형을 설정하거나 검색합니다. 이 저장소 형식은 CertOpenStore 함수에 직접 전달됩니다. (가져오기)

get_RequestStoreTypeWStr RequestStoreNameWStr 속성에 지정된 저장소에 사용할 저장소 유형을 설정하거나 검색합니다. 이 저장소 형식은 CertOpenStore 함수에 직접 전달됩니다. (가져오기)
get_Response 인증 기관에서 반환된 인증서 응답을 검색합니다.
get_ReuseHardwareKeyIfUnableToGenNew 새 키를 생성할 때 오류가 발생하는 경우 인증서 등록 제어 개체에서 수행한 작업을 결정하는 부울 값을 설정하거나 검색합니다. (가져오기)
get_ReuseHardwareKeyIfUnableToGenNew IEnroll4의 ReuseHardwareKeyIfUnableToGenNew 속성은 새 키를 생성할 때 오류가 발생하는 경우 인증서 등록 제어 개체에서 수행한 작업을 결정하는 부울 값을 설정하거나 검색합니다. (가져오기)
get_ReuseKey 기존 프라이빗 키가 요청에 서명하는 데 사용되었는지 여부를 나타내는 부울 값을 검색합니다.
get_RoleAssignments 현재 IAzApplication3 개체와 연결된 IAzRoleAssignment 개체의 컬렉션을 나타내는 IAzRoleAssignments 개체를 가져옵니다.
get_RoleAssignments 이 범위와 연결된 IAzRoleAssignment 개체의 컬렉션을 나타내는 IAzRoleAssignments 개체를 검색합니다.
get_RoleDefinitions 현재 IAzApplication3 개체와 연결된 IAzRoleDefinition 개체의 컬렉션을 나타내는 IAzRoleDefinitions 개체를 가져옵니다.
get_RoleDefinitions 이 IAzRoleAssignment 개체와 연결된 IAzRoleDefinition 개체의 컬렉션을 검색합니다.
get_RoleDefinitions 이 IAzRoleDefinition 개체와 연결된 IAzRoleDefinition 개체의 컬렉션을 검색합니다.
get_RoleDefinitions 이 범위와 연결된 IAzRoleDefinition 개체의 컬렉션을 나타내는 IAzRoleDefinitions 개체를 검색합니다.

get_RoleForAccessCheck 액세스 검사를 수행하는 데 사용되는 역할을 설정하거나 검색합니다. (가져오기)
get_Roles IAzApplication의 Roles 속성은 정책 데이터에서 IAzRole 개체를 열거하는 데 사용되는 IAzRoles 개체를 검색합니다.
get_Roles 정책 데이터에서 IAzRole 개체를 열거하는 데 사용되는 IAzRoles 개체를 검색합니다.
get_RootStoreFlags ICEnroll4의 RootStoreFlags 속성은 루트 저장소에 사용되는 레지스트리 위치를 설정하거나 검색합니다. (가져오기)
get_RootStoreFlags 루트 저장소에 사용되는 레지스트리 위치를 설정하거나 검색합니다. (가져오기)
get_RootStoreName 기본적으로 신뢰할 수 있는 자체 서명된 루트 인증서가 모두 유지되는 루트 저장소의 이름을 설정하거나 검색합니다. (가져오기)
get_RootStoreNameWStr IEnroll4의 RootStoreNameWStr 속성은 기본적으로 신뢰할 수 있는 자체 서명된 루트 인증서가 모두 유지되는 루트 저장소의 이름을 설정하거나 검색합니다. (가져오기)
get_RootStoreType RootStoreName 속성에 지정된 저장소에 사용할 저장소 유형을 설정하거나 검색합니다. (가져오기)
get_RootStoreTypeWStr RootStoreNameWStr 속성에 지정된 저장소에 사용할 저장소 유형을 설정하거나 검색합니다. (가져오기)
get_Scope 이 IAzRoleAssignment 개체가 정의된 범위를 나타내는 IAzScope 개체를 검색합니다.
get_Scopes 정책 데이터에서 IAzScope 개체를 열거하는 데 사용되는 IAzScopes 개체를 검색합니다.

get_ScriptEngineTimeout IAzClientContext::AccessCheck 메서드가 비즈니스 규칙(BizRule)이 실행을 완료할 때까지 기다린 후 취소할 시간을 밀리초 단위로 설정하거나 검색합니다. (가져오기)
get_SecurityDescriptor 프라이빗 키에 대한 보안 설명자를 지정하거나 검색합니다. (가져오기)
get_Selected 등록 프로세스 중에 항목을 사용할 수 있는지 여부를 나타내는 값을 지정하거나 검색합니다. (가져오기)
get_SenderNonce nonce를 포함하는 바이트 배열을 지정하거나 검색합니다. (가져오기)
get_SerialNumber 인증서 일련 번호를 지정하고 검색합니다. (가져오기)
get_SHA1Hash 인증서의 SHA-1 해시를 검색합니다.
get_Sids 이 클라이언트 컨텍스트와 연결된 SID(보안 식별자)의 배열을 가져옵니다.
get_Signature 공급자에서 디지털 서명을 검색합니다.
get_Signature 인코딩 메서드에서 만든 요청 서명을 검색합니다.
get_SignatureInformation 인증서 서명에 대한 정보가 포함된 IX509SignatureInformation 개체를 검색합니다.
get_SignatureInformation 인증서 요청에 서명하는 데 사용되는 기본 서명에 대한 정보가 포함된 IX509SignatureInformation 개체를 검색합니다.
get_SignatureInformation 인증서 요청 서명에 대한 정보를 포함하는 IX509SignatureInformation 개체를 검색합니다.

get_SignerCertificate 인증서 서명에 사용되는 ISignerCertificate 개체를 지정하거나 검색합니다. (가져오기)
get_SignerCertificate 인증서 요청에 서명하는 데 사용되는 인증서를 지정하거나 검색합니다. (가져오기)
get_SignerCertificate 요청에 대한 서명자 인증서를 가져오거나 설정합니다. (가져오기)
get_SignerCertificates 요청에 서명하는 데 사용되는 인증서 컬렉션을 검색합니다.
get_SigningCertificate DER(고유 인코딩 규칙)를 사용하여 인코딩된 서명 인증서를 가져오거나 설정합니다. OCSP(온라인 인증서 상태 프로토콜) 응답자 서비스는 이 인증서를 사용하여 인증서 상태 요청에 대한 응답에 서명합니다. (가져오기)
get_SigningCertificateTemplate 서명 인증서의 템플릿 이름을 가져오거나 설정합니다. (가져오기)
get_SigningFlags 플래그 값의 조합을 가져오거나 설정합니다. 이러한 값은 CA(인증 기관) 구성에 속하는 서명 인증서의 관리를 지정합니다. (가져오기)
get_Silent 프라이빗 키를 사용하여 인증서 요청에 서명할 때 사용자에게 알림을 받을지 여부를 나타내는 부울 값을 지정하거나 검색합니다. (가져오기)
get_Silent 인증서 등록 프로세스 중에 키 관련 모달 대화 상자가 표시되는지 여부를 나타내는 부울 값을 지정하거나 검색합니다. (가져오기)
get_Silent 인증서 등록 프로세스 중에 사용자 인터페이스가 표시되는지 여부를 나타내는 부울 값을 지정하거나 검색합니다. (가져오기)
get_Silent

프라이빗 키에 액세스할 때 인증서 등록 컨트롤에서 대화 상자를 표시할 수 있는지 여부를 나타내는 부울 값을 지정하거나 검색합니다. (가져오기)
get_SmimeCapabilities 컴퓨터에서 지원하는 암호화 기능을 식별하는 IX509ExtensionSmimeCapabilities 컬렉션을 만들지 여부를 인코딩 메서드에 알려주는 부울 값을 지정하거나 검색합니다. (가져오기)
get_SmimeCapabilities ISmimeCapability 개체의 컬렉션을 검색합니다.
get_SPCFileName base64로 인코딩된 PKCS(Get)를 쓸 파일의 이름을 설정하거나 검색합니다.
get_SPCFileNameWStr IEnroll4의 SPCFileNameWStr 속성은 base64로 인코딩된 PKCS(Get)를 쓸 파일의 이름을 설정하거나 검색합니다.
get_Status 등록 프로세스의 상태를 모니터링하고 오류 정보를 검색하는 데 사용할 수 있는 IX509EnrollmentStatus 개체를 검색합니다.
get_Status 등록 프로세스의 상태를 나타내는 값을 지정하거나 검색합니다. (가져오기)
get_Status 요청의 상태를 가져옵니다.
get_StrValue 전자 메일 주소, DNS(도메인 이름 시스템) 이름, URL, 등록된 개체 식별자(OID) 또는 UPN(사용자 계정 이름)이 포함된 문자열을 검색합니다.
get_Subject 인증서를 요청하는 엔터티의 X.500 고유 이름을 지정하거나 검색합니다. (가져오기)
get_SubjectKeyIdentifier 키 식별자를 포함하는 바이트 배열을 검색합니다.
get_SuppressDefaults

기본 확장과 특성이 요청에 포함되는지 여부를 나타내는 부울 값을 지정하거나 검색합니다. (가져오기)
get_SuppressOids 인코딩 프로세스 중에 인증서에서 표시하지 않을 확장 또는 OID(특성 개체 식별자) 컬렉션을 검색합니다.
get_SuppressOids 요청이 인코딩될 때 요청에 추가되지 않은 기본 확장 및 특성 개체 식별자(OID)의 컬렉션을 검색합니다.
get_TargetMachine 계정 확인이 수행되어야 하는 컴퓨터의 이름을 검색합니다.
get_Tasks IAzApplication의 Tasks 속성은 정책 데이터에서 IAzTask 개체를 열거하는 데 사용되는 IAzTasks 개체를 검색합니다.
get_Tasks 역할과 연결된 작업을 검색합니다.
get_Tasks 정책 데이터에서 IAzTask 개체를 열거하는 데 사용되는 IAzTasks 개체를 검색합니다.
get_Tasks 작업과 연결된 작업을 검색합니다.
get_Template 초기화 중에 사용되는 인증서 요청 템플릿을 검색합니다. (IX509CertificateRequestCertificate2.get_Template)
get_Template 초기화 중에 사용되는 인증서 요청 템플릿을 검색합니다. (IX509CertificateRequestCmc2.get_Template)
get_Template 초기화 중에 사용되는 인증서 요청 템플릿을 검색합니다. (IX509CertificateRequestPkcs10V2.get_Template)
get_Template 초기화 중에 사용되는 인증서 요청 템플릿을 검색합니다. (IX509CertificateRequestPkcs7V2.get_Template)
get_Template

이 IX509CertificateTemplateWritable 인스턴스를 초기화하는 데 사용된 IX509CertificateTemplate 개체의 복사본을 검색합니다.
get_Template 초기화 중에 사용되는 인증서 요청 템플릿을 검색합니다. (IX509Enrollment2.get_Template)
get_TemplateName 인증서가 자동 등록에 사용할 수 있는 템플릿의 이름을 포함하는 문자열을 검색합니다.
get_TemplateName 템플릿의 이름을 검색합니다.
get_TemplateObjectId 인증서 요청을 만드는 데 사용되는 템플릿의 OID(개체 식별자)를 검색합니다. (IX509CertificateRequestCmc.get_TemplateObjectId)
get_TemplateObjectId 인증서 요청을 만드는 데 사용되는 템플릿의 OID(개체 식별자)를 검색합니다. (IX509CertificateRequestPkcs10.get_TemplateObjectId)
get_TemplateOid 템플릿 개체 식별자(OID)를 검색합니다.
get_Text 등록 프로세스의 상태와 연결된 메시지가 포함된 문자열을 지정하거나 검색합니다. (가져오기)
get_ThumbPrint 인증서 데이터의 해시를 설정하거나 검색합니다. (가져오기)
get_ThumbPrintWStr 인증서 데이터의 해시를 설정하거나 검색합니다. 지문은 보류 중인 인증서를 가리키는 데 사용됩니다. (가져오기)
get_TransactionId 인증서 요청 또는 응답을 추적하는 데 사용할 수 있는 트랜잭션 식별자를 지정하거나 검색합니다. (가져오기)
get_TransactionId

요청에 대한 트랜잭션 ID를 가져오거나 설정합니다. (가져오기)
get_Type 애플리케이션 그룹의 그룹 유형을 설정하거나 검색합니다. (가져오기)
get_Type 대체 이름 형식을 검색합니다.
get_Type 알고리즘 유형을 검색합니다.
get_Type 공급자의 형식을 검색합니다.
get_Type 한정자 형식을 검색합니다.
get_Type 요청 개체의 형식을 지정하는 값을 검색합니다.
get_UIContextMessage 서명 인증서와 연결된 사용자 인터페이스 텍스트가 포함된 문자열을 지정하거나 검색합니다. (가져오기)
get_UIContextMessage 사용자 인터페이스에 표시할 컨텍스트 문자열을 지정하거나 검색합니다. (가져오기)
get_UIContextMessage 프라이빗 키와 연결된 사용자 인터페이스 텍스트가 포함된 문자열을 지정하거나 검색합니다. (가져오기)
get_UniqueContainerName 키 컨테이너의 고유한 이름을 검색합니다.
get_Url CEP(인증서 등록 정책) 서버의 URL을 지정하거나 검색합니다. (가져오기)
get_UseExistingKeySet 기존 키를 사용해야 하는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (가져오기)

get_UseExistingKeySet IEnroll4의 UseExistingKeySet 속성은 기존 키를 사용해야 하는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (가져오기)
get_UserCanonical 현재 클라이언트의 이름을 정식 형식으로 검색합니다.
get_UserDisplay 현재 클라이언트의 이름을 사용자 표시 이름 형식으로 검색합니다.
get_UserDn 현재 클라이언트의 이름을 DN(고유 이름) 형식으로 검색합니다.
get_UserDnsSamCompat 현재 클라이언트의 이름을 SAM(Windows Security Account Manager)과 호환되는 DNS 형식으로 검색합니다.
get_UserGuid GUID 형식으로 현재 클라이언트의 이름을 검색합니다.
get_UserSamCompat 현재 클라이언트의 이름을 SAM(Windows Security Account Manager)과 호환되는 형식으로 검색합니다.
get_UserSamName 사용자의 SAM(보안 계정 관리자) 이름을 검색합니다.
get_UserUpn 현재 클라이언트의 이름을 UPN(사용자 계정 이름) 형식으로 검색합니다.
get_Valid 알고리즘 개체가 유효한지 여부를 지정하는 부울 값을 검색합니다.
get_Valid 공급자가 클라이언트 컴퓨터에 설치되어 있는지 여부를 지정하는 부울 값을 검색합니다.
get_Value 점선 10진수 개체 식별자(OID)가 포함된 문자열을 검색합니다.

get_Value 이름-값 쌍의 값 부분을 검색합니다.
get_Value OCSPProperty 개체가 나타내는 이름-값 쌍의 데이터 부분을 가져오거나 설정합니다. (가져오기)
get_Values 특성 컬렉션을 포함하는 IX509Attributes 개체를 검색합니다.
get_Version 애플리케이션의 버전을 설정하거나 검색합니다. (가져오기)
get_Version 공급자의 버전 번호를 검색합니다.
get_Writable 개체를 초기화한 사용자 컨텍스트에서 개체를 수정할 수 있는지 여부를 나타내는 값을 검색합니다.
get_Writable 애플리케이션 그룹을 초기화한 사용자 컨텍스트에서 수정할 수 있는지 여부를 나타내는 값을 검색합니다.
get_Writable Initialize 메서드를 호출한 사용자 컨텍스트에서 개체를 수정할 수 있는지 여부를 나타내는 값을 검색합니다.
get_Writable 작업을 초기화한 사용자 컨텍스트에서 수정할 수 있는지 여부를 나타내는 값을 검색합니다.
get_Writable 역할을 초기화한 사용자 컨텍스트에서 역할을 수정할 수 있는지 여부를 나타내는 값을 검색합니다.
get_Writable 범위를 초기화한 사용자 컨텍스트에서 범위를 수정할 수 있는지 여부를 나타내는 값을 검색합니다.
get_Writable 작업을 초기화한 사용자 컨텍스트에서 수정할 수 있는지 여부를 나타내는 값을 검색합니다.

get_WriteCertToCSP ICEnroll4의 WriteCertToCSP 속성은 인증서를 CSP(암호화 서비스 공급자)에 기록해야 하는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (가져오기)
get_WriteCertToCSP 인증서를 CSP(암호화 서비스 공급자)에 기록해야 하는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (가져오기)
get_WriteCertToUserDS 인증서가 사용자의 Active Directory 저장소에 기록되는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (가져오기)
get_WriteCertToUserDS IEnroll4의 WriteCertToUserDS 속성은 인증서가 사용자의 Active Directory 저장소에 기록되는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (가져오기)
get_X509Extensions 인증서 확장을 검색합니다.
get_X509Extensions 인증서 요청에 포함된 확장의 컬렉션을 검색합니다. (IX509CertificateRequestCmc.get_X509Extensions)
get_X509Extensions 인증서 요청에 포함된 확장의 컬렉션을 검색합니다. (IX509CertificateRequestPkcs10.get_X509Extensions)
GetAccessRights GetAccessRights 메서드는 보안 개체에 대해 제어할 수 있는 액세스 권한에 대한 정보를 요청합니다.
GetAce ACL(액세스 제어 목록)에서 ACE(액세스 제어 항목)에 대한 포인터를 가져옵니다.
GetAclInformation ACL(액세스 제어 목록)에 대한 정보를 검색합니다.
GetAlgName ID가 지정된 암호화 알고리즘의 이름을 검색합니다. 이 메서드에서 검색하는 값은 현재 CSP(암호화 서비스 공급자)에 따라 달라집니다. 이 메서드는 ICEnroll3 인터페이스에서 처음 정의되었습니다.

GetAlgNameWStr ID가 지정된 암호화 알고리즘의 이름을 검색합니다. 이 메서드에서 검색하는 값은 현재 CSP(암호화 서비스 공급자)에 따라 달라집니다.
GetAlgorithmName OID(알고리즘 개체 식별자)와 연결된 표시 이름을 검색합니다.
GetAlgorithmOid OID(알고리즘 개체 식별자)를 검색합니다. 이 메서드는 웹을 사용할 수 있습니다.
GetAllowUnTrustedCA 신뢰할 수 없는 인증 기관 인증서를 허용할지 여부를 지정하는 값을 검색합니다.
GetAllProperties 속성 집합의 모든 속성을 가져옵니다.
GetAppContainerAce 패키지 또는 기능 SID가 있는지 여부를 나타내는 값을 검색합니다.
GetAppContainerNamedObjectPath 앱 컨테이너의 명명된 개체 경로를 검색합니다.
GetArchivedKey 보관된 키 복구 BLOB을 검색합니다.
GetAssignedScopesPage 현재 IAzClientContext2 개체가 나타내는 클라이언트가 하나 이상의 역할에 할당된 범위 목록을 검색합니다.
GetAt 지정된 ID 공급자에 대한 IIdentityProvider 인터페이스 포인터를 검색합니다.
GetAuditedPermissionsFromAclA 지정된 트러스티에 대한 감사된 액세스 권한을 검색합니다. (ANSI)
GetAuditedPermissionsFromAclW 지정된 트러스티에 대한 감사된 액세스 권한을 검색합니다. (유니코드)

GetAuthentication GetAuthentication 메서드는 클라이언트를 인증하기 위해 CEP(인증서 등록 정책) 서버에서 사용하는 인증 유형을 지정하는 값을 검색합니다. 이 값은 Initialize 메서드에 의해 설정됩니다.
GetAuthFlags 클라이언트가 CEP(인증서 등록 정책) 서버에 인증하는 데 사용하는 인증 유형을 지정하는 값을 검색합니다.
GetBitCount CertEncodeBitString 개체에 속하고 ICertEncodeBitString::D ecode에 대한 이전 호출로 초기화된 비트 문자열의 비트 수를 반환합니다.
GetBitString 개체의 비트 문자열에 있는 비트 문자열을 반환합니다.
GetBusinessRuleString 비즈니스 규칙(BizRule)에 대한 애플리케이션별 문자열을 반환합니다.
GetCACertificate 인증서 서비스 서버에 대한 CA(인증 기관) 인증서를 반환합니다.
GetCacheDir 정책 캐시 파일이 포함된 CEP(인증서 등록 정책) 서버에서 디렉터리의 이름을 검색합니다.
GetCachedSigningLevel 캐시된 서명 수준을 검색합니다.
GetCachePath CEP(인증서 등록 정책) 서버에서 정책 캐시 파일의 경로를 검색합니다.
GetCAProperty CA(인증 기관)의 속성 값을 검색합니다. (ICertRequest2.GetCAProperty)
GetCAProperty CA(인증 기관)의 속성 값을 검색합니다. (ICertAdmin2.GetCAProperty)
GetCAPropertyDisplayName

CA(인증 기관) 속성의 속성 표시 이름을 검색합니다.
GetCAPropertyDisplayName ICertAdmin2::GetCAPropertyDisplayName 메서드는 CA(인증 기관) 속성의 속성 표시 이름을 검색합니다.
GetCAPropertyFlags CA(인증 기관) 속성에 대한 속성 플래그를 검색합니다.
GetCAPropertyFlags ICertAdmin2::GetCAPropertyFlags 메서드는 CA(인증 기관) 속성에 대한 속성 플래그를 검색합니다.
GetCA 정책에 포함된 인증 등록 서버의 컬렉션을 검색합니다.
GetCASetupProperty CA(인증 기관) 구성에 대한 속성 값을 가져옵니다.
GetCAsForTemplate 지정된 템플릿을 지원하는 인증서 등록 서버의 컬렉션을 검색합니다.
getCAStore getCAStore 메서드가 구현되지 않았습니다.
getCertContextFromFileResponseWStr 인증 기관의 응답을 포함하는 파일에서 인증서를 검색합니다.
getCertContextFromPKCS7 PKCS를 기반으로 인증서 컨텍스트를 검색합니다.
getCertContextFromResponseBlob 인증 기관의 응답에서 인증서를 검색합니다.
getCertFromFileResponse 인증 기관의 응답을 포함하는 파일에서 인증서를 검색합니다. 이 메서드는 ICEnroll4 인터페이스에서 처음 정의되었습니다.
getCertFromPKCS7 PKCS에 포함된 인증서를 검색합니다.

getCertFromResponse 인증 기관의 응답에서 인증서를 검색합니다. 이 메서드는 ICEnroll4 인터페이스에 의해 처음 정의되었습니다.
GetCertificate 요청에 대해 X.509 인증서로 발급되거나 PKCS(공개 키 암호화 표준)에 선택적으로 패키지된 인증서를 반환합니다.
GetCertificateCount 키 스토리지 공급자의 인증 인증서 수를 가져옵니다.
GetCertificateExtension 지정된 인증서 확장을 가져옵니다.
GetCertificateExtension 특정 인증서 확장을 검색합니다.
GetCertificateExtensionFlags ICertServerExit::GetCertificateExtension에 대한 최신 호출에서 획득한 확장의 플래그를 가져옵니다.
GetCertificateExtensionFlags GetCertificateExtension에 대한 최신 호출에서 획득한 확장과 연결된 플래그를 검색합니다.
GetCertificateProperty 인증서에서 명명된 속성을 반환합니다. (ICertServerExit.GetCertificateProperty)
GetCertificateProperty 인증서에서 명명된 속성을 반환합니다. (ICertServerPolicy.GetCertificateProperty)
GetColumnCount Certificate Services 데이터베이스 보기의 열 수를 검색합니다.
GetConfig Certificate Services 서버에 대한 구성 문자열을 검색합니다. 이 메서드는 ICertConfig 인터페이스에서 처음 정의되었습니다.
GetConfigEntry

CA(인증 기관)에 대한 구성 정보를 검색합니다.
GetConfiguration OCSP(온라인 인증서 상태 프로토콜) 응답기 서버에 연결하고 서버의 구성 정보를 사용하여 OCSPAdmin 개체를 초기화합니다.
GetCount 시스템에 등록된 ID 공급자 수를 가져옵니다.
GetCount 개체의 DATE 배열에 있는 DATE 값의 수를 반환합니다.
GetCount 개체의 Long 배열에 있는 Long 값의 수를 반환합니다.
GetCount 문자열 배열의 문자열 값 수를 반환합니다.
GetCRL CA(인증서 서비스 인증 기관)에 대한 현재 CRL(인증서 해지 목록)을 검색합니다.
GetCspStatuses 호출자가 지정한 프라이빗 키의 의도된 사용과 일치하는 모든 공급자/알고리즘 쌍을 포함하는 ICspStatuses 컬렉션을 검색합니다.
GetCspStatusesFromOperations 지원되는 키 작업 및 선택적 공급자 정보를 통해 ICspStatuses 컬렉션을 검색합니다.
GetCspStatusFromOperations 지정된 서명, 암호화, 해시 또는 암호 작업과 일치하는 지원되는 첫 번째 알고리즘에 대한 ICspStatus 개체를 만듭니다.
GetCspStatusFromProviderName 공급자 이름 및 지원되는 키 작업별로 레거시 공급자에 대한 ICspStatus 개체를 검색합니다.
GetCustomOids 구현되지 않았습니다.

GetData GetData 메서드는 보안 구성 스냅인에서 구성 정보를 검색합니다.
GetDefaultSecurityDescriptor 기본 프라이빗 키 보안 설명자를 검색합니다.
GetDescription 정책 모듈 및 해당 함수에 대해 사람이 읽을 수 있는 설명을 반환합니다.
GetDescription 종료 모듈 및 해당 함수에 대한 사람이 읽을 수 있는 설명을 반환합니다.
GetDisplayName 열거형 시퀀스에서 현재 열의 지역화된 이름을 검색합니다.
GetDispositionMessage 인증서 요청의 현재 처리를 제공하는 사람이 읽을 수 있는 메시지를 가져옵니다.
GetDistPointCount CRL 배포 정보 배열의 CRL(인증서 해지 목록) 배포 지점 수를 반환합니다.
GetEffectivePermission 개체 형식에 대한 유효 사용 권한을 반환합니다.
GetEffectiveRightsFromAclA ACL 구조가 지정된 트러스티에 부여하는 유효 액세스 권한을 검색합니다. 트러스티의 유효한 액세스 권한은 ACL이 트러스티 또는 트러스티가 멤버인 그룹에 부여하는 액세스 권한입니다. (ANSI)
GetEffectiveRightsFromAclW ACL 구조가 지정된 트러스티에 부여하는 유효 액세스 권한을 검색합니다. 트러스티의 유효한 액세스 권한은 ACL이 트러스티 또는 트러스티가 멤버인 그룹에 부여하는 액세스 권한입니다. (유니코드)
GetEncryptionCspAlgorithms 공급자가 지원하는 암호화 알고리즘의 컬렉션을 검색합니다.
GetEncSChannel 이 함수를 사용할 수 없습니다.

GetEnrollmentServerAuthentication GetEnrollmentServerAuthentication 메서드는 클라이언트를 인증하기 위해 CES(인증서 등록 서버)에서 사용하는 인증 유형을 지정하는 값을 검색합니다. 이 값은 Initialize 메서드에 의해 설정됩니다.
GetEnrollmentServerUrl 인증서 등록 서버의 URL을 포함하는 문자열을 검색합니다.
GetErrorMessageText HRESULT 오류 코드에 대한 오류 메시지 텍스트를 검색합니다.
GetExistingCACertificates 컴퓨터에 현재 설치된 유효한 CA(인증 기관) 인증서를 나타내는 CertSrvSetupKeyInformation 개체의 컬렉션을 가져옵니다.
GetExplicitEntriesFromAclA ACL(액세스 제어 목록)에서 ACE(액세스 제어 항목)를 설명하는 구조의 배열을 검색합니다. (ANSI)
GetExplicitEntriesFromAclW ACL(액세스 제어 목록)에서 ACE(액세스 제어 항목)를 설명하는 구조의 배열을 검색합니다. (유니코드)
GetField 구성 데이터베이스의 현재 레코드에서 특정 필드를 가져옵니다. 이 메서드는 ICertConfig 인터페이스에서 처음 정의되었습니다.
GetFileSecurityA 파일 또는 디렉터리의 보안에 대한 지정된 정보를 가져옵니다. 가져온 정보는 호출자의 액세스 권한 및 권한에 의해 제한됩니다. (GetFileSecurityA)
GetFileSecurityW 파일 또는 디렉터리의 보안에 대한 지정된 정보를 가져옵니다. 가져온 정보는 호출자의 액세스 권한 및 권한에 의해 제한됩니다. (GetFileSecurityW)
GetFlags 확장 열거형 시퀀스에서 현재 확장의 정책 및 원본 플래그를 검색합니다.
GetFriendlyName CEP(인증서 등록 정책) 서버의 표시 이름을 검색합니다.

GetFriendlyNameOfCertA 인증서의 표시 이름을 검색합니다. (ANSI)
GetFriendlyNameOfCertW 인증서의 표시 이름을 검색합니다. (유니코드)
GetFullResourceName OpenElevatedEditor 메서드를 호출하여 표시되는 액세스 제어 편집기와 연결된 개체의 전체 경로 및 파일 이름을 검색합니다.
GetFullResponseProperty 서버에서 반환된 캐시된 응답 데이터를 검색합니다.
GetGroups 이 클라이언트 컨텍스트와 연결된 애플리케이션 그룹의 배열을 반환합니다.
GetHashAlgorithmList 비대칭 서명 키 알고리즘에 대해 지정된 CSP(암호화 서비스 공급자)에서 지원하는 해시 알고리즘 목록을 가져옵니다.
GetHashAlgorithms 공급자가 지원하는 해시 알고리즘의 컬렉션을 검색합니다.
GetHashAlgorithms 해시 알고리즘 이름 목록을 가져옵니다. OCSP(온라인 인증서 상태 프로토콜) 응답자 서버는 이러한 이름을 사용하여 지정된 CA(인증 기관) 구성에 대한 OCSP 응답에 서명합니다.
GetIdentityEnum ID를 열거하는 데 사용할 수 있는 IEnumUnknown 인터페이스 포인터를 검색합니다.
GetInheritanceSourceA ACL(액세스 제어 목록)에서 상속된 ACE(액세스 제어 항목)의 원본에 대한 정보를 반환합니다. (ANSI)
GetInheritanceSourceW ACL(액세스 제어 목록)에서 상속된 ACE(액세스 제어 항목)의 원본에 대한 정보를 반환합니다. (유니코드)
GetInheritSource

ISecurityObjectTypeInfo::GetInheritSource 메서드는 임의 액세스 제어 목록 및 시스템 액세스 제어 목록에서 상속된 액세스 제어 항목의 원본을 결정하는 방법을 제공합니다.
GetInheritTypes GetInheritTypes 메서드는 자식 개체에서 ACE를 상속하는 방법에 대한 정보를 요청합니다. 자세한 내용은 ACE 상속을 참조하세요.
GetInnerRequest 중첩된 요청 개체를 검색합니다.
GetInterfaceValue 지정된 인터페이스 이름에 해당하는 인터페이스의 ID 및 플래그를 가져옵니다.
GetIsDefaultCEP 이것이 기본 CEP(인증서 등록 정책) 서버인지 여부를 지정하는 부울 값을 검색합니다.
GetIssuedCertificate 요청 ID 또는 인증서 일련 번호를 지정하여 인증서의 처리를 검색합니다.
GetIssuedCertificate2 요청 ID 문자열 또는 인증서 일련 번호를 지정하여 인증서의 처리를 검색합니다.
GetKernelObjectSecurity 커널 개체를 보호하는 보안 설명자의 복사본을 검색합니다.
GetKeyLen 서명 및 교환 키의 최소 및 최대 키 길이를 검색합니다.
GetKeyLen IEnroll4::GetKeyLen 메서드는 서명 및 교환 키에 대한 최소 및 최대 키 길이를 검색합니다.
GetKeyLenEx 서명 및 교환 키에 대한 크기 정보를 검색합니다. 이 메서드는 ICEnroll4 인터페이스에서 처음 정의되었습니다.
GetKeyLenEx 서명 및 교환 키에 대한 크기 정보를 검색합니다.

GetKeyLengthList 지정된 CSP(암호화 서비스 공급자)에서 지원하는 키 길이 목록을 가져옵니다. (ICertSrvSetup.GetKeyLengthList)
GetKeyLengthList 지정된 CSP(암호화 서비스 공급자)에서 지원하는 키 길이 목록을 가져옵니다. (IMSCEPSetup.GetKeyLengthList)
GetLastStatus 이 요청에 대한 마지막 반환 코드를 가져옵니다. 그러면 요청 처리 대신 오류 코드 정보가 반환됩니다.
GetLastUpdateTime 정책이 마지막으로 다운로드된 날짜와 시간을 검색합니다.
GetLengthSid 유효한 SID(보안 식별자)의 길이(바이트)를 반환합니다.
GetManageModule GetManageModule을 호출하고 ICertManageModule에 대한 포인터의 주소를 전달하여 ICertPolicy2 인터페이스와 연결된 ICertManageModule 인터페이스를 검색합니다.
GetManageModule GetManageModule을 호출하고 ICertManageModule에 대한 포인터의 주소를 전달하여 ICertExit2 인터페이스와 연결된 ICertManageModule 인터페이스를 검색합니다.
GetMaxLength 열 데이터에 대해 허용되는 최대 길이(바이트)를 검색합니다.
GetMSCEPSetupProperty NDES(네트워크 디바이스 등록 서비스) 구성에 대한 속성 값을 가져옵니다.
GetMyRoles 호출자의 CA(인증 기관) 역할을 검색합니다.
GetMyRoles 지정된 OCSP(온라인 인증서 상태 프로토콜) 응답기 서버에서 사용자에 대한 권한 역할의 액세스 마스크를 가져옵니다.

getMyStore getMyStore 메서드가 구현되지 않았습니다.
GetName 대체 이름 배열에서 지정된 이름을 반환합니다.
GetName CRL(인증서 해지 목록) 배포 정보 지점의 지정된 인덱스에서 이름을 반환합니다.
GetName 특성 열거형 시퀀스에서 현재 특성의 이름을 검색합니다.
GetName 열 열거형 시퀀스에서 현재 열의 비로컬화된 이름을 검색합니다.
GetName 확장 열거형 시퀀스에서 현재 확장의 이름을 검색합니다.
GetNameChoice 대체 이름 배열의 지정된 인덱스에서 이름 선택을 반환합니다.
GetNameChoice CRL(인증서 해지 목록) 배포 정보 지점의 지정된 인덱스에서 선택한 이름을 반환합니다.
GetNameCount 대체 이름 배열의 이름 수를 반환합니다.
GetNameCount CRL(인증서 해지 목록) 배포 지점의 이름 수를 반환합니다.
GetNamedSecurityInfoA 이름으로 지정된 개체에 대한 보안 설명자의 복사본을 검색합니다. (ANSI)
GetNamedSecurityInfoW 이름으로 지정된 개체에 대한 보안 설명자의 복사본을 검색합니다. (유니코드)
GetNextUpdateTime

정책이 만료되고 새로 고쳐야 하는 날짜와 시간을 검색합니다.
GetObjectInformation GetObjectInformation 메서드는 액세스 제어 편집기에서 해당 페이지를 초기화하고 사용자가 사용할 수 있는 편집 옵션을 결정하는 데 사용하는 정보를 요청합니다.
GetOpenCardNameA GetOpenCardName 함수는 스마트 카드 "카드 선택" 대화 상자를 표시합니다. (ANSI)
GetOpenCardNameW GetOpenCardName 함수는 스마트 카드 "카드 선택" 대화 상자를 표시합니다. (유니코드)
GetOperations 현재 클라이언트 컨텍스트에서 나타내는 보안 주체가 수행할 권한이 있는 작업 컬렉션을 지정된 범위 내에서 반환합니다.
GetParameter IAzClientContext::AccessCheck 메서드의 varParameterValues 매개 변수에서 지정된 값을 가져옵니다.
GetParameterValue 지정된 이름을 가진 비즈니스 규칙(BizRule) 매개 변수의 값 형식을 가져옵니다.
GetPasswordCredentials Active Directory를 사용하여 도메인에 가입되지 않은 컨테이너를 인증하는 자격 증명을 반환합니다.
GetPolicyServerId CEP(인증서 등록 정책) 서버를 고유하게 식별하는 문자열을 검색합니다.
GetPolicyServerId CEP(인증서 등록 정책) 서버를 고유하게 식별하는 문자열 값을 검색합니다.
GetPolicyServerUrl CEP(인증서 등록 정책) 서버의 URL을 포함하는 문자열을 검색합니다.
GetPolicyServerUrl CEP(인증서 등록 정책) 서버의 URL을 포함하는 문자열 값을 검색합니다.
GetPrincipals

사용자가 하나 이상의 보안 주체를 선택할 수 있는 대화 상자를 표시한 다음 선택한 보안 주체 목록과 해당 SID(보안 식별자)를 반환합니다.
GetPrivateKeyArchiveCertificate GetPrivateKeyArchiveCertificate 메서드는 프라이빗 키를 보관하는 데 사용되는 인증서를 검색합니다. 이 메서드는 IEnroll4 인터페이스에서 처음 정의되었습니다.
GetPrivateKeyContainerList 비대칭 서명 키 알고리즘에 대해 지정된 CSP(암호화 서비스 공급자)가 저장한 키 컨테이너 이름 목록을 가져옵니다.
GetPrivateObjectSecurity 개인 개체의 보안 설명자에서 정보를 검색합니다.
GetProperty 지정된 속성 ID를 사용하여 IAzApplication 개체 속성을 반환합니다.
GetProperty 지정된 속성 ID를 사용하여 IAzApplicationGroup 개체 속성을 반환합니다.
GetProperty 지정된 속성 ID를 사용하여 AzAuthorizationStore 개체 속성을 반환합니다.
GetProperty 지정된 속성 ID를 사용하여 IAzClientContext 개체 속성을 반환합니다.
GetProperty 지정된 속성 ID를 사용하여 IAzOperation 개체 속성을 반환합니다.
GetProperty 지정된 속성 ID를 사용하여 IAzRole 개체 속성을 반환합니다.
GetProperty 지정된 속성 ID를 사용하여 IAzScope 개체 속성을 반환합니다.
GetProperty 지정된 속성 ID를 사용하여 IAzTask 개체 속성을 반환합니다.

GetProperty CEP(인증서 등록 정책) 웹 서비스 구성에 대한 CEPSetupProperty 열거형 값을 검색합니다.
GetProperty CES(인증서 등록 웹 서비스) 구성에 대한 CESSetupProperty 열거형 값을 검색합니다.
GetProperty 모듈의 속성 값을 검색합니다.
GetPropertyFlags 기본 정책 서버 URL을 지정하는 값을 검색합니다.
GetProviderNameList 컴퓨터에서 비대칭 키 서명 알고리즘을 제공하는 CSP(암호화 서비스 공급자) 목록을 가져옵니다.
GetProviderNameList 컴퓨터에서 비대칭 키 서명 및 교환 알고리즘을 제공하는 CSP(암호화 서비스 공급자) 목록을 가져옵니다.
GetProviderPropertyStore ID 공급자와 연결된 IPropertyStore 인터페이스에 대한 포인터를 검색합니다.
getProviderType 지정된 CSP(암호화 서비스 공급자)의 형식을 검색합니다. 이 메서드는 ICEnroll4 인터페이스에서 처음 정의되었습니다.
getProviderTypeWStr 지정된 CSP(암호화 서비스 공급자)의 형식을 검색합니다.
GetRefreshPolicy 클라이언트의 캐시된 인증서 등록 정책이 만료되어 새로 고쳐야 하는지 여부를 나타내는 값을 반환합니다.
GetRequestAttribute 요청에서 명명된 특성 값을 반환합니다.
GetRequestAttribute 요청에서 명명된 특성을 반환합니다.

GetRequestId 요청 및 후속 인증서에 대한 현재 내부 요청 번호를 가져옵니다.
GetRequestIdString 등록하는 동안 인증 기관에 전송된 인증서 요청에 대한 고유 문자열 식별자를 검색합니다.
GetRequestIdString 요청 및 후속 인증서에 대한 문자열 형식의 현재 내부 요청 번호를 가져옵니다.
GetRequestProperty 요청에서 명명된 속성을 반환합니다.
GetRequestProperty 요청에서 특정 속성을 검색합니다.
GetRevocationReason 인증서가 해지된 이유를 반환합니다. 이 메서드는 ICertAdmin 인터페이스에서 처음 정의되었습니다.
GetRoles 클라이언트 컨텍스트에 대한 역할을 반환합니다.
getROOTHStore getROOTHStore 메서드가 구현되지 않았습니다.
GetSchemaVersion 이 권한 부여 저장소의 버전 번호를 가져옵니다.
GetSecondarySecurity 리소스에 대한 액세스에 영향을 미칠 수 있는 추가 보안 컨텍스트를 반환합니다.
GetSecurity GetSecurity 메서드는 보안 설명자를 편집 중인 보안 개체에 대한 보안 설명자를 요청합니다. 액세스 제어 편집기는 이 메서드를 호출하여 개체의 현재 또는 기본 보안 설명자를 검색합니다.
GetSecurity OCSP(온라인 인증서 상태 프로토콜) 응답자 서버에 대한 보안 설명자 정보를 가져옵니다.

GetSecurityDescriptorControl 보안 설명자 컨트롤 및 수정 정보를 검색합니다.
GetSecurityDescriptorDacl 지정된 보안 설명자의 DACL(임의 액세스 제어 목록)에 대한 포인터를 검색합니다.
GetSecurityDescriptorGroup 보안 설명자에서 기본 그룹 정보를 검색합니다.
GetSecurityDescriptorLength 구조적으로 유효한 보안 설명자의 길이(바이트)를 반환합니다. 길이에는 연결된 모든 구조체의 길이가 포함됩니다.
GetSecurityDescriptorOwner 보안 설명자에서 소유자 정보를 검색합니다.
GetSecurityDescriptorRMControl 리소스 관리자 컨트롤 비트를 검색합니다.
GetSecurityDescriptorSacl 지정된 보안 설명자에서 SACL(시스템 액세스 제어 목록)에 대한 포인터를 검색합니다.
GetSecurityInfo 핸들에서 지정한 개체에 대한 보안 설명자의 복사본을 검색합니다.
GetServiceDirectory 서비스에서 상태를 읽고/또는 쓸 서비스별 파일 시스템 위치에 대한 경로를 반환합니다.
GetServiceDisplayNameA 지정된 서비스의 표시 이름을 검색합니다. (ANSI)
GetServiceDisplayNameW 지정된 서비스의 표시 이름을 검색합니다. (유니코드)
GetServiceKeyNameA 지정된 서비스의 서비스 이름을 검색합니다. (ANSI)

GetServiceKeyNameW 지정된 서비스의 서비스 이름을 검색합니다. (유니코드)
GetServiceRegistryStateKey 서비스에서 상태를 읽고/또는 쓸 레지스트리 키에 대한 핸들을 반환합니다.
GetSharedServiceDirectory 서비스 및 관련 프로그램에서 상태를 읽고/또는 쓸 서비스별 파일 시스템 위치에 대한 경로를 반환합니다.
GetSharedServiceRegistryStateKey 상태를 읽고/또는 쓸 서비스 및 관련 프로그램에 대한 레지스트리 키에 대한 핸들을 반환합니다.
GetSidIdentifierAuthority 지정된 SID(보안 식별자)의 SID_IDENTIFIER_AUTHORITY 구조체에 대한 포인터를 반환합니다.
GetSidLengthRequired 지정된 수의 하위 인증을 사용하여 SID를 저장하는 데 필요한 버퍼의 길이(바이트)를 반환합니다.
GetSidSubAuthority SID(보안 식별자)에서 지정된 하위 인증에 대한 포인터를 반환합니다. 하위 인증 값은 RID(상대 식별자)입니다.
GetSidSubAuthorityCount 하위 인증 수를 포함하는 SID(보안 식별자) 구조체의 멤버에 대한 포인터를 반환합니다.
GetSignatureAlgorithm 서명 알고리즘 개체 식별자(OID)를 검색합니다.
GetSigningCertificates 지정된 CA(인증 기관) 인증서에 대해 응답자 서버에서 사용할 수 있는 서명 인증서를 가져옵니다.
GetStringProperty CEP(인증서 등록 정책) 서버 ID 또는 CEP 서버의 표시 이름을 검색합니다.
GetStringType 문자열 배열에 포함된 문자열 값의 형식을 반환합니다.

GetSupportedCATypes 호출자 컨텍스트에서 컴퓨터에 설치할 수 있는 CA(인증 기관) 유형을 가져옵니다.
GetSupportedKeySpec 서명 및/또는 교환 작업에 대한 현재 CSP(암호화 서비스 공급자) 지원에 대한 정보를 검색합니다. 이 메서드는 ICEnroll3 인터페이스에서 처음 정의되었습니다.
GetSupportedKeySpec 서명 및/또는 교환 작업에 대한 현재 CSP(암호화 서비스 공급자) 지원에 대한 정보를 검색합니다.
GetTasks 현재 클라이언트 컨텍스트에서 나타내는 보안 주체가 수행할 수 있는 권한이 있는 지정된 범위 내의 태스크 컬렉션을 반환합니다.
GetTemplates CEP(인증서 등록 정책) 서버에서 지원하는 템플릿의 컬렉션을 검색합니다.
GetTokenInformation 액세스 토큰에 대한 지정된 유형의 정보를 검색합니다. 호출 프로세스에는 정보를 얻기 위한 적절한 액세스 권한이 있어야 합니다.
GetTrusteeFormA 지정된 TRUSTEE 구조에서 트러스티 이름을 검색합니다. 이 값은 구조에서 이름 문자열 또는 SID(보안 식별자)를 사용하여 트러스티를 식별하는지 여부를 나타냅니다. (ANSI)
GetTrusteeFormW 지정된 TRUSTEE 구조에서 트러스티 이름을 검색합니다. 이 값은 구조에서 이름 문자열 또는 SID(보안 식별자)를 사용하여 트러스티를 식별하는지 여부를 나타냅니다. (유니코드)
GetTrusteeNameA 지정된 TRUSTEE 구조에서 트러스티 이름을 검색합니다. (ANSI)
GetTrusteeNameW 지정된 TRUSTEE 구조에서 트러스티 이름을 검색합니다. (유니코드)
GetTrusteeTypeA 지정된 TRUSTEE 구조에서 트러스티 형식을 검색합니다. 이 값은 트러스티가 사용자, 그룹 또는 트러스티 유형을 알 수 없는지 여부를 나타냅니다. (ANSI)

GetTrusteeTypeW 지정된 TRUSTEE 구조에서 트러스티 형식을 검색합니다. 이 값은 트러스티가 사용자, 그룹 또는 트러스티 유형을 알 수 없는지 여부를 나타냅니다. (유니코드)
GetType 열거형 시퀀스에서 현재 열의 데이터 형식을 검색합니다.
GetUrl 지정된 마법사 또는 웹 페이지의 URL 문자열을 반환합니다.
GetUrlFlags CEP(인증서 등록 정책) 서버에 대한 기타 정책 정보를 포함하는 플래그 집합을 검색합니다.
GetUseClientId ClientId 특성이 CEP(인증서 등록 정책) 서버의 정책 서버 플래그에 설정되어 있는지 여부를 지정하는 값을 검색합니다.
GetUserObjectSecurity 지정된 사용자 개체에 대한 보안 정보를 검색합니다.
GetValue DATE 배열에서 지정된 날짜를 반환합니다.
GetValue Long 배열에서 지정된 Long 값을 반환합니다.
GetValue 문자열 배열에서 지정된 문자열을 반환합니다.
GetValue 특성 열거형 시퀀스에서 현재 특성의 값을 검색합니다.
GetValue 열 열거형 시퀀스의 현재 열에 포함된 데이터 값을 검색합니다.
GetValue 확장 열거형 시퀀스에서 현재 확장의 값을 검색합니다.

GetWindowsAccountDomainSid SID(보안 식별자)를 받고 해당 SID의 도메인을 나타내는 SID를 반환합니다.
IdentityUpdated ID 이벤트가 발생했음을 호출 애플리케이션에 알리기 위해 ID 공급자가 호출합니다.
ImpersonateAnonymousToken 지정된 스레드가 시스템의 익명 로그온 토큰을 가장할 수 있도록 합니다.
ImpersonateLoggedOnUser 호출 스레드가 로그온한 사용자의 보안 컨텍스트를 가장할 수 있도록 합니다. 사용자는 토큰 핸들로 표시됩니다.
ImpersonateNamedPipeClient 명명된 파이프 클라이언트 애플리케이션을 가장합니다.
ImpersonateSecurityContext 서버가 AcceptSecurityContext(일반) 또는 QuerySecurityContextToken 호출로 이전에 가져온 토큰을 사용하여 클라이언트를 가장할 수 있습니다.
자체 가장 호출 프로세스의 보안 컨텍스트를 가장하는 액세스 토큰을 가져옵니다. 토큰은 호출 스레드에 할당됩니다.
가져오기 ID를 시스템으로 가져옵니다.
가져오기 기존 프라이빗 키를 암호화 공급자 내의 키 컨테이너로 가져옵니다.
ImportCertificate 이전에 발급한 인증서를 가져와 CA(인증 기관) 데이터베이스로 가져옵니다. 이 메서드는 ICertAdmin 인터페이스에서 처음 정의되었습니다.
ImportKey 인증서 서비스 데이터베이스의 항목에 암호화된 키 집합을 추가합니다. 키 집합은 하나 이상의 KRA(키 복구 에이전트) 인증서로 암호화됩니다.

ImportPFXToProvider PFX 인증서를 가져옵니다.
ImportPFXToProviderFreeData PFX 인증서 컨텍스트를 해제합니다.
ImportSecurityContextA 보안 컨텍스트를 가져옵니다. 보안 컨텍스트는 ExportSecurityContext에 대한 이전 호출에 의해 ImportSecurityContext를 호출하는 프로세스로 내보내졌어야 합니다. (ANSI)
ImportSecurityContextW 보안 컨텍스트를 가져옵니다. 보안 컨텍스트는 ExportSecurityContext에 대한 이전 호출에 의해 ImportSecurityContext를 호출하는 프로세스로 내보내졌어야 합니다. (유니코드)
초기화 권한 부여 관리자를 초기화합니다.
초기화 CA에서 반환된 CMS(CMS) 응답을 통해 전체 인증서 관리를 사용하여 초기화합니다.
초기화 OID(개체 식별자)에서 개체를 초기화합니다.
초기화 인증서가 보관되었는지 여부를 지정하는 부울 값에서 개체를 초기화합니다.
초기화 해시를 포함하는 바이트 배열에서 개체를 초기화합니다.
초기화 자동 등록에 사용할 템플릿의 이름을 지정하여 개체를 초기화합니다.
초기화 부울 값 및 날짜에서 개체를 초기화합니다.
초기화 인증서에 대한 설명 정보가 포함된 문자열에서 개체를 초기화합니다.

초기화 인증서 요청 ID, CA(인증 기관) 구성 문자열 및 선택적 인증서 표시 이름에서 속성을 초기화합니다.
초기화 ICertPropertyEnrollmentPolicyServer 개체를 초기화합니다.
초기화 인증서 표시 이름에서 개체를 초기화합니다.
초기화 프라이빗 키에서 개체를 초기화합니다.
초기화 새 인증서의 SHA-1 해시에서 개체를 초기화합니다.
초기화 원래 컴퓨터의 DNS 이름을 포함하는 문자열에서 개체를 초기화합니다.
초기화 인증서의 SHA-1 해시에서 개체를 초기화합니다.
초기화 암호화 공급자 및 연결된 알고리즘에서 개체를 초기화합니다.
초기화 서명 인증서에서 개체를 초기화합니다.
초기화 OID(대칭 암호화 알고리즘 개체 식별자) 및 선택적 키 길이에서 개체를 초기화합니다.
초기화 OID(개체 식별자) 및 값에서 개체를 초기화합니다.
초기화 사용자 또는 컴퓨터에 대한 요청 개체를 초기화합니다.
초기화

템플릿에서 IX509CertificateTemplateWritable 개체를 초기화합니다.
초기화 등록 개체를 초기화하고 기본 PKCS를 만듭니다.
초기화 IX509EnrollmentHelper 개체를 초기화합니다.
초기화 IX509EnrollmentPolicyServer 개체를 초기화합니다.
초기화 OID(개체 식별자) 및 DER(고유 인코딩 규칙) 인코딩 확장이 포함된 바이트 배열을 사용하여 IX509Extension 개체를 초기화합니다.
초기화 이름 및 연결된 값을 포함하는 문자열에서 개체를 초기화합니다.
초기화 IX509PolicyServerListManager 개체를 초기화합니다.
초기화 컴퓨터 또는 사용자 컨텍스트에 대한 IX509PolicyServerUrl 개체를 초기화합니다.
초기화 공개 키 알고리즘 개체 식별자(OID)와 공개 키 및 연결된 매개 변수가 포함된 바이트 배열(있는 경우)에서 개체를 초기화합니다.
초기화 새 요청에 대비하여 인스턴스를 초기화합니다.
초기화 정책 모듈이 초기화 작업을 수행할 수 있도록 서버 엔진에서 호출됩니다.
초기화 NDES 정책 모듈을 초기화합니다.
초기화

자신을 초기화할 때 서버 엔진에서 호출됩니다.
초기화 Initialize 메서드는 스냅인 확장이 로드되었음을 보안 구성 스냅인에 알리고 통신에 대한 컨텍스트를 설정합니다.
InitializeAcl 새 ACL 구조를 초기화합니다.
InitializeClientContext2 IAzClientContext2 개체 포인터를 검색합니다.
InitializeClientContextFromName 클라이언트 ID에서 IAzClientContext 개체 포인터를 (도메인 이름, 클라이언트 이름) 쌍으로 가져옵니다.
InitializeClientContextFromStringSid 텍스트 형식의 지정된 SID(보안 식별자)에서 IAzClientContext 개체 포인터를 가져옵니다.
InitializeClientContextFromToken 지정된 클라이언트 토큰에서 IAzClientContext 개체 포인터를 가져옵니다.
InitializeClientContextFromToken2 지정된 클라이언트 토큰에서 IAzClientContext2 개체 포인터를 검색합니다.
InitializeDecode 속성 값이 포함된 바이트 배열에서 개체를 초기화합니다.
InitializeDecode 암호화된 프라이빗 키를 포함하는 DER(고유 인코딩 규칙)로 인코딩된 바이트 배열에서 개체를 초기화합니다.
InitializeDecode 암호화된 프라이빗 키의 SHA-1 해시를 포함하는 DER(고유 인코딩 규칙)로 인코딩된 바이트 배열에서 개체를 초기화합니다.
InitializeDecode 특성 값을 포함하는 DER(고유 인코딩 규칙)로 인코딩된 바이트 배열에서 개체를 초기화합니다. (IX509AttributeClientId.InitializeDecode)

InitializeDecode 공급자에 대한 정보를 포함하는 DER(고유 인코딩 규칙)로 인코딩된 바이트 배열에서 개체를 초기화합니다.
InitializeDecode 특성 값을 포함하는 DER(고유 인코딩 규칙)로 인코딩된 바이트 배열에서 개체를 초기화합니다. (IX509AttributeExtensions.InitializeDecode)
InitializeDecode 운영 체제 버전 정보를 포함하는 DER(고유 인코딩 규칙)로 인코딩된 바이트 배열에서 개체를 초기화합니다.
InitializeDecode 갱신할 인증서를 포함하는 DER(고유 인코딩 규칙)로 인코딩된 바이트 배열에서 개체를 초기화합니다.
InitializeDecode 기존 서명되거나 서명되지 않은 PKCS를 디코딩합니다(IX509CertificateRequestPkcs10.InitializeDecode).
InitializeDecode 기존 서명되거나 서명되지 않은 PKCS(IX509CertificateRequestPkcs7.InitializeDecode)를 디코딩합니다.
InitializeDecode 확장 값을 포함하는 DER(고유 인코딩 규칙)로 인코딩된 바이트 배열에서 확장을 초기화합니다. (IX509ExtensionAlternativeNames.InitializeDecode)
InitializeDecode 확장 값을 포함하는 DER(고유 인코딩 규칙)로 인코딩된 바이트 배열에서 확장을 초기화합니다. (IX509ExtensionAuthorityKeyIdentifier.InitializeDecode)
InitializeDecode 확장 값을 포함하는 DER(고유 인코딩 규칙)로 인코딩된 바이트 배열에서 확장을 초기화합니다. (IX509ExtensionBasicConstraints.InitializeDecode)
InitializeDecode 확장 값을 포함하는 DER(고유 인코딩 규칙)로 인코딩된 바이트 배열에서 개체를 초기화합니다.
InitializeDecode 확장 값을 포함하는 DER(고유 인코딩 규칙)로 인코딩된 바이트 배열에서 확장을 초기화합니다. (IX509ExtensionEnhancedKeyUsage.InitializeDecode)

InitializeDecode 확장 값을 포함하는 DER(고유 인코딩 규칙)로 인코딩된 바이트 배열에서 확장을 초기화합니다. (IX509ExtensionKeyUsage.InitializeDecode)
InitializeDecode 확장 값을 포함하는 DER(고유 인코딩 규칙)로 인코딩된 바이트 배열에서 확장을 초기화합니다. (IX509ExtensionMSApplicationPolicies.InitializeDecode)
InitializeDecode 확장 값을 포함하는 DER(고유 인코딩 규칙)로 인코딩된 바이트 배열에서 확장을 초기화합니다. (IX509ExtensionSmimeCapabilities.InitializeDecode)
InitializeDecode 확장 값을 포함하는 DER(고유 인코딩 규칙)로 인코딩된 바이트 배열에서 확장을 초기화합니다. (IX509ExtensionSubjectKeyIdentifier.InitializeDecode)
InitializeDecode 확장 값을 포함하는 DER로 인코딩된 바이트 배열에서 확장을 초기화합니다.
InitializeDecode 확장 값을 포함하는 DER(고유 인코딩 규칙)로 인코딩된 바이트 배열에서 확장을 초기화합니다. (IX509ExtensionTemplateName.InitializeDecode)
InitializeDefaults 인증 기관 역할을 설치할 수 있도록 기본값으로 CCertSrvSetup 개체를 초기화합니다.
InitializeDefaults 기본값을 사용하여 CMSCEPSetup 개체를 초기화하여 NDES(네트워크 디바이스 등록 서비스) 역할을 설치할 수 있도록 합니다.
InitializeEncode 문자열과 한정자 형식을 식별하는 값에서 개체를 초기화합니다.
InitializeEncode IX509PrivateKey 개체, 인증 기관 암호화 인증서 및 대칭 암호화 알고리즘 개체 식별자(OID)에서 특성을 초기화합니다.
InitializeEncode

인증서 요청을 제출한 사용자, 클라이언트 컴퓨터 및 애플리케이션에 대한 정보에서 특성을 초기화합니다.
InitializeEncode 공급자에 대한 정보에서 특성을 초기화합니다.
InitializeEncode IX509Extensions 컬렉션에서 개체를 초기화합니다.
InitializeEncode 운영 체제 버전 정보에서 특성을 초기화합니다.
InitializeEncode 갱신할 인증서를 사용하여 특성을 초기화합니다.
InitializeEncode IAlternativeNames 컬렉션에서 확장을 초기화합니다.
InitializeEncode 바이트 배열에서 확장을 초기화합니다.
InitializeEncode 인증서 주체가 CA(인증 기관) 및 하위 CA 체인의 깊이를 포함하는 정수인지 여부를 나타내는 부울 값에서 확장을 초기화합니다.
InitializeEncode ICertificatePolicies 컬렉션에서 개체를 초기화합니다.
InitializeEncode 공용 키의 의도된 사용을 지정하는 IObjectId 개체 식별자(OID) 컬렉션에서 확장을 초기화합니다.
InitializeEncode X509KeyUsageFlags 열거형을 사용하여 확장을 초기화합니다.
InitializeEncode ICertificatePolicies 컬렉션에서 확장을 초기화합니다.
InitializeEncode ISmimeCapabilities 컬렉션에서 확장을 초기화합니다.

InitializeEncode 키 식별자를 포함하는 바이트 배열에서 확장을 초기화합니다.
InitializeEncode 템플릿 OID(개체 식별자) 및 주 버전 및 부 버전 번호에서 확장을 초기화합니다.
InitializeEncode 템플릿 이름을 포함하는 문자열에서 확장을 초기화합니다.
InitializeEncodeFromEncryptedKeyBlob 암호화된 프라이빗 키에서 특성을 초기화합니다.
InitializeForPending 발급된 인증서를 검색하거나 발급자의 이전 요청에 대한 응답을 설치하는 메시지를 생성하도록 준비하도록 인스턴스를 초기화합니다.
InitializeFromAlgorithmName 알고리즘 이름 또는 개체 식별자에서 개체를 초기화합니다.
InitializeFromCertificate 인증서에 포함된 속성에서 컬렉션을 초기화합니다.
InitializeFromCertificate 기존 인증서와 연결된 속성 값을 사용하여 개체를 초기화합니다.
InitializeFromCertificate 기존 인증서를 사용하여 인증서 요청을 초기화합니다. (IX509CertificateRequestPkcs10.InitializeFromCertificate)
InitializeFromCertificate 기존 인증서를 사용하여 인증서 요청을 초기화합니다. (IX509CertificateRequestPkcs7.InitializeFromCertificate)
InitializeFromCertificateHash 새 인증서에서 개체를 초기화합니다.
InitializeFromCurrentTime

부울 값과 현재 시스템 날짜 및 시간에서 속성을 초기화합니다.
InitializeFromEncodedPublicKeyInfo 공개 키가 포함된 바이트 배열에서 개체를 초기화합니다.
InitializeFromInnerRequest 내부 PKCS에서 인증서 요청을 초기화합니다.
InitializeFromInnerRequestTemplate 내부 요청 개체 및 템플릿에서 인증서 요청을 초기화합니다.
InitializeFromInnerRequestTemplateName InitializeFromInnerRequestTemplateName 메서드는 내부 요청 개체 및 템플릿에서 인증서 요청을 초기화합니다.
InitializeFromLocalRequestOriginator 로컬 컴퓨터의 DNS 이름에서 개체를 초기화합니다.
InitializeFromName 공급자 이름을 포함하는 문자열에서 개체를 초기화합니다.
InitializeFromName CERTENROLL_OBJECTID 열거형 값에서 개체를 초기화합니다.
InitializeFromObjectId 개체 식별자를 사용하여 암호화 특성을 초기화합니다.
InitializeFromOtherName OID(개체 식별자) 및 연결된 원시 데이터(바이트 배열)에서 개체를 초기화합니다.
InitializeFromPrivateKey IX509PrivateKey 개체 및 필요에 따라 템플릿을 사용하여 인증서 요청을 초기화합니다.
InitializeFromPrivateKeyTemplate IX509PrivateKey 개체 및 인증서 템플릿을 사용하여 인증서 요청을 초기화합니다. (IX509CertificateRequestCertificate2.InitializeFromPrivateKeyTemplate)
InitializeFromPrivateKeyTemplate

IX509PrivateKey 개체 및 인증서 템플릿을 사용하여 인증서 요청을 초기화합니다. (IX509CertificateRequestPkcs10V2.InitializeFromPrivateKeyTemplate)
InitializeFromProperties 기존 서버 구성에 포함된 속성에서 속성 집합을 만듭니다.
InitializeFromPublicKey IX509PublicKey 개체 및 필요에 따라 템플릿을 사용하여 null 서명된 인증서 요청을 초기화합니다.
InitializeFromPublicKeyTemplate IX509PublicKey 개체 및 템플릿을 사용하여 null 서명된 인증서 요청을 초기화합니다.
InitializeFromRawData DSA(디지털 서명 알고리즘) GUID, X.500 디렉터리 이름 또는 DER(고유 인코딩 규칙)로 인코딩된 바이트 배열에 포함된 IP(인터넷 프로토콜) 주소에서 개체를 초기화합니다.
InitializeFromRequest 기존 IX509CertificateRequest 개체에서 등록 개체를 초기화합니다.
InitializeFromString 전자 메일 주소, DNS(도메인 이름 시스템) 이름, URL, 등록된 개체 식별자(OID) 또는 UPN(사용자 계정 이름)이 포함된 문자열에서 개체를 초기화합니다.
InitializeFromTemplate 템플릿을 사용하여 인증서 요청을 초기화합니다. (IX509CertificateRequestCertificate2.InitializeFromTemplate)
InitializeFromTemplate 템플릿을 사용하여 인증서 요청을 초기화합니다. (IX509CertificateRequestCmc2.InitializeFromTemplate)
InitializeFromTemplate 템플릿을 사용하여 인증서 요청을 초기화합니다. (IX509CertificateRequestPkcs10V2.InitializeFromTemplate)
InitializeFromTemplate 템플릿을 사용하여 인증서 요청을 초기화합니다. (IX509CertificateRequestPkcs7V2.InitializeFromTemplate)
InitializeFromTemplate

템플릿을 사용하여 등록 개체를 초기화합니다.
InitializeFromTemplateName . (IX509CertificateRequestPkcs10.InitializeFromTemplateName)
InitializeFromTemplateName 템플릿을 사용하여 인증서 요청을 초기화합니다. (IX509CertificateRequestPkcs7.InitializeFromTemplateName)
InitializeFromTemplateName 템플릿 CN(일반 이름)에서 등록 개체를 초기화합니다.
InitializeFromType 기본 암호화 공급자에서 개체를 초기화합니다.
InitializeFromValue 점선 OID(10진수 개체 식별자)가 포함된 문자열에서 개체를 초기화합니다.
InitializeFromValues IX509Attributes 개체를 사용하여 암호화 특성을 초기화합니다.
InitializeImport 템플릿 및 개체 식별자 컬렉션에서 CEP(인증서 등록 정책) 서버를 초기화합니다.
InitializeInstallDefaults 기본 구성을 사용하여 ICertificateEnrollmentPolicyServerSetup 개체를 초기화합니다.
InitializeInstallDefaults 기본 구성을 사용하여 ICertificateEnrollmentServerSetup 개체를 초기화합니다.
InitializeSecurityContextA 자격 증명 핸들에서 클라이언트 쪽 아웃바운드 보안 컨텍스트를 시작합니다. (ANSI)
InitializeSecurityContextW 자격 증명 핸들에서 클라이언트 쪽 아웃바운드 보안 컨텍스트를 시작합니다. (유니코드)
InitializeSecurityDescriptor 새 보안 설명자를 초기화합니다.

InitializeSid SID(보안 식별자)를 초기화합니다.
InitSecurityInterfaceA InitSecurityInterface 함수는 SSPI 디스패치 테이블에 대한 포인터를 반환합니다. 이 함수를 사용하면 클라이언트가 인터페이스 구현에 직접 바인딩하지 않고 SSPI를 사용할 수 있습니다. (ANSI)
InitSecurityInterfaceW InitSecurityInterface 함수는 SSPI 디스패치 테이블에 대한 포인터를 반환합니다. 이 함수를 사용하면 클라이언트가 인터페이스 구현에 직접 바인딩하지 않고 SSPI를 사용할 수 있습니다. (유니코드)
설치 ICertificateEnrollmentPolicyServerSetup 개체로 구성된 CEP(인증서 등록 정책) 웹 서비스를 설치합니다.
설치 ICertificateEnrollmentServerSetup 개체에 의해 구성된 CES(인증서 등록 웹 서비스)를 설치합니다.
설치 CCertSrvSetup 개체에 구성된 역할을 설치합니다.
설치 CMSCEPSetup 개체에 구성된 대로 NDES(네트워크 디바이스 등록 서비스) 역할을 설치합니다.
InstallPKCS7 인증서 또는 인증서 체인을 처리하여 적절한 인증서 저장소에 배치합니다. 이 메서드는 InstallPKCS7이 요청 인증서를 받지 않는다는 점에서 acceptPKCS7 메서드와 다릅니다.
InstallPKCS7Blob 인증서 또는 인증서 체인을 처리하여 적절한 인증서 저장소에 배치합니다. 이 메서드는 InstallPKCS7Blob이 요청 인증서를 받지 않는다는 점에서 acceptPKCS7Blob 메서드와 다릅니다.
InstallPKCS7BlobEx 로컬 저장소에 실제로 설치된 인증서 수를 반환한다는 점을 제외하고 InstallPKCS7Blob과 동일합니다.
InstallPKCS7Ex 인증서 또는 인증서 체인을 처리하여 적절한 인증서 저장소에 배치합니다. InstallPKCS7은 로컬 저장소에 실제로 설치된 인증서 수를 반환한다는 점을 제외하고.

InstallResponse 최종 엔터티 컴퓨터에 인증서 체인을 설치합니다. (IX509Enrollment.InstallResponse)
InstallResponse2 최종 엔터티 컴퓨터에 인증서 체인을 설치합니다. (IX509Enrollment2.InstallResponse2)
IsCatalogFile 지정된 파일이 카탈로그 파일인지 여부를 나타내는 부울 값을 검색합니다.
IsCrossIsolatedEnvironmentClipboardContent IsCrossIsolatedEnvironmentClipboardContent는 붙여넣는 콘텐츠가 MDAG(Microsoft Defender Application Guard) 경계의 다른 쪽에서 제공되었는지 확인하기 위해 붙여넣기 실패를 감지한 후 호출됩니다.
IsDaclCanonical IsDaclCanonical 메서드는 지정된 DACL 구조에 포함된 ACE가 클라이언트에서 구현한 DACL 순서 정의에 따라 정렬되는지 여부를 결정합니다.
IsDirty IsDirty 메서드는 첨부 파일 스냅인의 데이터가 마지막으로 저장된 이후 수정되었는지 여부를 나타내는 값을 반환합니다.
IsFunctionalLevelUpgradeSupported 이 권한 부여 저장소의 버전을 업그레이드할 수 있는지 여부를 나타내는 부울 값을 가져옵니다.
IsIndexed 열의 데이터가 인덱싱되는지 여부를 보고합니다.
IsInRoleAssignment 현재 클라이언트 컨텍스트에서 나타내는 보안 주체가 지정된 범위에서 지정된 역할의 멤버인지 여부를 확인합니다.
IsMSCEPStoreEmpty 항상 VARIANT_TRUE 반환합니다. 사용해서는 안 됩니다.
IsProcessInIsolatedWindowsEnvironment 애플리케이션이 실행 중인 실행 환경을 결정합니다.

IsPropertyEditable 지정된 속성을 편집할 수 있는지 여부를 호출자에게 나타냅니다.
IsSmartCard 요청 개체와 연결된 암호화 공급자가 스마트 카드 공급자인지 여부를 나타내는 부울 값을 검색합니다.
IsTokenRestricted 토큰에 제한된 SID(보안 식별자) 목록이 포함되어 있는지 여부를 나타냅니다.
IsUpdateNeeded 이 권한 부여 저장소의 지속형 버전이 캐시된 버전보다 최신 버전인지 확인합니다.
IsValidAcl ACL(액세스 제어 목록)의 유효성을 검사합니다.
IsValidCertificate CA(인증 기관) 키에 대해 인증서를 확인하고 인증서가 해지되지 않은지 확인합니다. 이 메서드는 ICertAdmin 인터페이스에서 처음 정의되었습니다.
IsValidSecurityDescriptor 보안 설명자의 구성 요소가 유효한지 여부를 확인합니다.
IsValidSid 수정 번호가 알려진 범위 내에 있고 하위 인증의 수가 최대값보다 작은지 확인하여 SID(보안 식별자)의 유효성을 검사합니다.
IsWellKnownSid SID를 잘 알려진 SID와 비교하고 일치하는 경우 TRUE를 반환합니다.
KeyCredentialManagerFreeInformation KeyCredentialManagerGetInformation 호출에서 KeyCredentialManagerInfo 포인터 변수를 해제하는 API입니다.
KeyCredentialManagerGetInformation 사용자 등록의 고유 식별자를 가져오는 API입니다.
KeyCredentialManagerGetOperationErrorStates

작업이 이전에 성공할지 확인하기 위해 호출하기 위한 필수 구성 요소 API입니다.
KeyCredentialManagerShowUIOperation 요청된 WHFB 작업을 수행하는 API입니다.
KspDeleteContextFn 보안 컨텍스트를 삭제합니다.
KspMakeSignatureFn 지정된 메시지 및 보안 컨텍스트에 따라 서명을 생성합니다.
KspVerifySignatureFn 받은 메시지가 서명에 따라 올바른지 확인합니다.
LoadPolicy CEP(인증서 등록 정책) 서버에서 정책 정보를 검색합니다.
LockServiceDatabase SCM(서비스 제어 관리자) 데이터베이스 잠금의 소유권을 요청합니다. 지정된 시간에는 하나의 프로세스만 잠금을 소유할 수 있습니다.
LogonUserA Win32 LogonUser 함수는 사용자를 로컬 컴퓨터에 로그온하려고 시도합니다. LogonUser는 사용자를 가장하는 데 사용할 수 있는 사용자 토큰에 대한 핸들을 반환합니다. (ANSI)
LogonUserExA LogonUserEx 함수는 사용자를 로컬 컴퓨터에 로그온하려고 시도합니다. (ANSI)
LogonUserExW LogonUserEx 함수는 사용자를 로컬 컴퓨터에 로그온하려고 시도합니다. (유니코드)
LogonUserW Win32 LogonUser 함수는 사용자를 로컬 컴퓨터에 로그온하려고 시도합니다. LogonUser는 사용자를 가장하는 데 사용할 수 있는 사용자 토큰에 대한 핸들을 반환합니다. (유니코드)
lookupAccountNameA 시스템 및 계정의 이름을 입력으로 허용합니다. 계정의 SID(보안 식별자)와 계정이 발견된 도메인의 이름을 검색합니다. (ANSI)

LookupAccountNameW 시스템 및 계정의 이름을 입력으로 허용합니다. 계정의 SID(보안 식별자)와 계정이 발견된 도메인의 이름을 검색합니다. (유니코드)
LookupAccountSidA SID(보안 식별자)를 입력으로 허용합니다. 이 SID의 계정 이름과 이 SID가 발견된 첫 번째 도메인의 이름을 검색합니다. (ANSI)
LookupAccountSidLocalA 로컬 컴퓨터에서 지정된 SID에 대한 계정의 이름을 검색합니다. (ANSI)
LookupAccountSidLocalW 로컬 컴퓨터에서 지정된 SID에 대한 계정의 이름을 검색합니다. (유니코드)
LookupAccountSidW SID(보안 식별자)를 입력으로 허용합니다. 이 SID의 계정 이름과 이 SID가 발견된 첫 번째 도메인의 이름을 검색합니다. (유니코드)
LookupPrivilegeDisplayNameA 지정된 권한을 나타내는 표시 이름을 검색합니다. (ANSI)
lookupPrivilegeDisplayNameW 지정된 권한을 나타내는 표시 이름을 검색합니다. (유니코드)
lookupPrivilegeNameA 지정된 LUID(로컬 고유 식별자)에 의해 특정 시스템에 표시되는 권한에 해당하는 이름을 검색합니다. (ANSI)
lookupPrivilegeNameW 지정된 LUID(로컬 고유 식별자)에 의해 특정 시스템에 표시되는 권한에 해당하는 이름을 검색합니다. (유니코드)
lookupPrivilegeValueA 지정된 시스템에서 지정된 권한 이름을 로컬로 나타내는 데 사용되는 LUID(로컬 고유 식별자)를 검색합니다. (ANSI)
LookupPrivilegeValueW

지정된 시스템에서 지정된 권한 이름을 로컬로 나타내는 데 사용되는 LUID(로컬 고유 식별자)를 검색합니다. (유니코드)
LookupSecurityDescriptorPartsA 자체 상대 보안 설명자에서 보안 정보를 검색합니다. (ANSI)
LookupSecurityDescriptorPartsW 자체 상대 보안 설명자에서 보안 정보를 검색합니다. (유니코드)
LookupSids LookupSids 메서드는 지정된 SID 목록에 있는 각 요소에 해당하는 일반 이름을 반환합니다.
LPHANDLER_FUNCTION RegisterServiceCtrlHandler 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. 서비스 프로그램은 특정 서비스의 제어 처리기 함수로 사용할 수 있습니다.
LPHANDLER_FUNCTION_EX RegisterServiceCtrlHandlerEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. 서비스 프로그램은 특정 서비스의 제어 처리기 함수로 사용할 수 있습니다.
LPSERVICE_MAIN_FUNCTIONA 서비스의 진입점입니다. (ANSI)
LPSERVICE_MAIN_FUNCTIONW 서비스의 진입점입니다. (유니코드)
LSA_ADD_CREDENTIAL 로그온 세션에 자격 증명을 추가합니다.
LSA_ALLOCATE_CLIENT_BUFFER 클라이언트의 주소 공간에 버퍼를 할당합니다.
LSA_ALLOCATE_LSA_HEAP 힙에 메모리를 할당합니다. LSA로 다시 전달되는 일부 정보는 이 함수를 사용하여 할당되어야 합니다.
LSA_ALLOCATE_PRIVATE_HEAP 프라이빗 힙에 메모리를 할당합니다.

LSA_ALLOCATE_SHARED_MEMORY AllocateSharedMemory 함수는 이전에 CreateSharedMemory 함수를 호출하여 예약한 메모리 섹션에서 공유 메모리 블록을 할당합니다.
LSA_AP_CALL_PACKAGE LSA에 대한 신뢰할 수 있는 연결을 가진 로그온 애플리케이션이 LsaCallAuthenticationPackage 함수를 호출하고 인증 패키지의 식별자를 지정하는 경우 LSA(로컬 보안 기관)에서 호출됩니다.
LSA_AP_CALL_PACKAGE_PASSTHROUGH LsaCallAuthenticationPackage 함수로 전송된 통과 로그온 요청에 대한 디스패치 함수입니다.
LSA_AP_INITIALIZE_PACKAGE 시스템 초기화 중에 LSA(로컬 보안 기관)에서 한 번 호출하여 인증 패키지를 초기화할 기회를 제공합니다.
LSA_AP_LOGON_TERMINATED 로그온 세션이 종료되면 인증 패키지에 알리는 데 사용됩니다. 로그온 세션을 참조하는 마지막 토큰이 삭제되면 로그온 세션이 종료됩니다.
LSA_AP_LOGON_USER LSA_AP_LOGON_USER(ntsecpkg.h) 콜백 함수는 사용자의 로그온 자격 증명을 인증합니다.
LSA_AP_LOGON_USER_EX LSA_AP_LOGON_USER_EX(ntsecpkg.h) 콜백 함수는 사용자의 로그온 자격 증명을 인증합니다.
LSA_AP_LOGON_USER_EX2 사용자의 초기 로그온에서 사용자 로그온 시도를 인증하는 데 사용됩니다. 사용자에 대한 새 로그온 세션이 설정되고 사용자에 대한 유효성 검사 정보가 반환됩니다.
LSA_AUDIT_ACCOUNT_LOGON AuditAccountLogon 함수는 외부 보안 주체 이름을 Windows 계정에 매핑하는 것을 나타내는 감사 레코드를 생성합니다.
LSA_AUDIT_LOGON AuditLogon 함수는 로그온 시도를 감사하는 데 사용됩니다.
LSA_CALL_PACKAGE CallPackage 함수는 다른 보안 패키지를 호출하여 해당 서비스에 액세스하는 데 사용됩니다.

LSA_CALL_PACKAGE_PASSTHROUGH CallPackagePassthrough 함수는 다른 보안 패키지를 호출하여 해당 서비스에 액세스하는 데 사용됩니다.
LSA_CALL_PACKAGEEX CallPackageEx 함수는 다른 보안 패키지를 호출하여 해당 서비스에 액세스하는 데 사용됩니다.
LSA_CANCEL_NOTIFICATION CancelNotification 함수는 이전에 등록된 알림을 취소합니다.
LSA_CLIENT_CALLBACK LSA(로컬 보안 기관) 모드 보안 패키지가 해당 사용자 모드 패키지로 다시 호출하고 해당 DLL에서 함수를 호출할 수 있도록 허용합니다.
LSA_CLOSE_SAM_USER SAM(보안 계정 관리자) 사용자 계정에 대한 핸들을 닫습니다.
LSA_CONVERT_AUTH_DATA_TO_TOKEN ConvertAuthDataToToken 함수는 GetAuthDataForUser 또는 GetUserAuthData 함수에서 반환된 권한 부여 데이터에서 액세스 토큰을 만듭니다.
LSA_COPY_FROM_CLIENT_BUFFER 클라이언트 프로세스의 주소 공간에서 현재 프로세스의 버퍼로 정보를 복사합니다.
LSA_COPY_TO_CLIENT_BUFFER 현재 프로세스의 버퍼에서 클라이언트 프로세스의 주소 공간으로 정보를 복사합니다.
LSA_CRACK_SINGLE_NAME CrackSingleName 함수는 이름을 한 형식에서 다른 형식으로 변환합니다.
LSA_CREATE_LOGON_SESSION 로그온 세션을 만듭니다.
LSA_CREATE_SHARED_MEMORY CreateSharedMemory 함수는 클라이언트 프로세스 및 보안 패키지에서 공유하는 메모리 섹션을 만듭니다.
LSA_CREATE_THREAD LSA(로컬 보안 기관)에서 사용해야 하는 Windows CreateThread 함수에 대한 래퍼입니다.

LSA_CREATE_TOKEN CreateToken 함수는 SSP/AP에서 SpAcceptLsaModeContext에 대한 호출을 처리하는 동안 토큰을 만드는 데 사용됩니다.
LSA_CREATE_TOKEN_EX SpAcceptLsaModeContext에 대한 호출을 처리하는 동안 토큰을 만듭니다.
LSA_DELETE_CREDENTIAL 기존 자격 증명을 삭제합니다.
LSA_DELETE_LOGON_SESSION 사용자의 인증 정보가 합법적인지 여부를 확인하면서 생성된 로그온 세션을 정리합니다.
LSA_DELETE_SHARED_MEMORY DeleteSharedMemory 함수는 클라이언트와 보안 패키지에서 공유하는 메모리 섹션을 해제합니다.
LSA_DUPLICATE_HANDLE DuplicateHandle 함수는 중복 핸들을 만듭니다. 반환된 중복 항목은 호출자의 프로세스 공간에 있습니다.
LSA_EXPAND_AUTH_DATA_FOR_DOMAIN 지정된 사용자 인증 데이터의 도메인 그룹을 확장합니다.
LSA_FREE_CLIENT_BUFFER AllocateClientBuffer 함수를 사용하여 이전에 할당된 클라이언트 버퍼를 해제합니다.
LSA_FREE_LSA_HEAP FreeReturnBuffer 함수는 LSA(로컬 보안 기관)에서 할당한 버퍼를 해제하고 보안 패키지로 반환하는 데 사용됩니다. 패키지는 반환된 버퍼의 정보가 더 이상 필요하지 않은 경우 이 함수를 호출합니다.
LSA_FREE_LSA_HEAP AllocateLsaHeap에서 이전에 할당한 힙 메모리 할당을 취소합니다.
LSA_FREE_PRIVATE_HEAP AllocatePrivateHeap 함수를 사용하여 할당된 메모리를 해제합니다.
LSA_FREE_SHARED_MEMORY

FreeSharedMemory 함수는 AllocateSharedMemory 함수에서 이전에 할당한 공유 메모리 블록을 해제합니다.
LSA_GET_AUTH_DATA_FOR_USER GetAuthDataForUser 함수는 SAM(Security Accounts Manager) 데이터베이스에서 사용자에 대한 인증 정보를 검색하여 ConvertAuthDataToken 함수에 적합한 형식으로 만듭니다.
LSA_GET_CALL_INFO GetCallInfo 함수는 최신 함수 호출에 대한 정보를 검색합니다.
LSA_GET_CLIENT_INFO GetClientInfo 함수는 스레드 및 프로세스 ID와 같은 클라이언트 프로세스에 대한 정보와 클라이언트의 상태 및 권한을 나타내는 플래그를 가져옵니다.
LSA_GET_CREDENTIALS 로그온 세션과 연결된 자격 증명을 검색합니다.
LSA_GET_USER_AUTH_DATA GetUserAuthData 함수는 단일 버퍼에서 사용자의 권한 부여 데이터를 반환합니다.
LSA_MAP_BUFFER SecBuffer 구조를 SSP/AP(보안 지원 공급자/인증 패키지)의 주소 공간에 매핑합니다.
LSA_OPEN_SAM_USER SAM(보안 계정 관리자) 데이터베이스에서 사용자 계정에 대한 핸들을 검색합니다.
LSA_OPEN_TOKEN_BY_LOGON_ID 지정된 사용자 로그온과 연결된 사용자 액세스 토큰을 엽니다.
LSA_PROTECT_MEMORY 지정된 메모리 버퍼를 암호화합니다.
LSA_REGISTER_NOTIFICATION 보안 패키지에 알림을 제공하는 메커니즘을 제공합니다. 알림은 일정한 간격으로, 이벤트 개체에 신호를 보낼 때 또는 특정 시스템 이벤트 중에 발생할 수 있습니다.
LSA_UPDATE_PRIMARY_CREDENTIALS 한 보안 패키지가 다른 패키지에 로그온 세션의 자격 증명이 변경되었음을 알리는 메커니즘을 제공합니다.

LsaAddAccountRights 계정에 하나 이상의 권한을 할당합니다.
LsaCallAuthenticationPackage 로그온 애플리케이션에서 인증 패키지와 통신하는 데 사용됩니다.
LsaClose LsaClose 함수는 Policy 또는 TrustedDomain 개체에 대한 핸들을 닫습니다.
LsaConnectUntrusted LSA 서버에 대한 신뢰할 수 없는 연결을 설정합니다.
LsaCreateTrustedDomainEx LsaCreateTrustedDomainEx 함수는 새 TrustedDomain 개체를 만들어 신뢰할 수 있는 새 도메인을 설정합니다.
LsaDeleteTrustedDomain LsaDeleteTrustedDomain 함수는 시스템의 신뢰할 수 있는 도메인 목록에서 신뢰할 수 있는 도메인을 제거하고 연결된 TrustedDomain 개체를 삭제합니다.
LsaDeregisterLogonProcess 호출자의 로그온 애플리케이션 컨텍스트를 삭제하고 LSA 서버에 대한 연결을 닫습니다.
LsaEnumerateAccountRights LsaEnumerateAccountRights 함수는 계정에 할당된 권한을 열거합니다.
LsaEnumerateAccountsWithUserRight 지정된 권한을 보유하는 LSA(로컬 보안 기관) 정책 개체의 데이터베이스에 있는 계정을 반환합니다.
LsaEnumerateLogonSessions 기존 로그온 세션 식별자(LUD) 집합 및 세션 수를 검색합니다.
LsaEnumerateTrustedDomains LsaEnumerateTrustedDomains 함수는 로그온 자격 증명을 인증하기 위해 신뢰할 수 있는 도메인의 이름과 SID를 검색합니다.
LsaEnumerateTrustedDomainsEx

로컬 시스템에서 신뢰하는 도메인에 대한 정보를 반환합니다.
LsaFreeMemory LsaFreeMemory 함수는 LSA 함수 호출에 의해 출력 버퍼에 할당된 메모리를 해제합니다.
LsaFreeReturnBuffer 이전에 LSA에서 할당한 버퍼에서 사용하는 메모리를 해제합니다.
LsaGetAppliedCAPIDs 특정 컴퓨터에 적용된 모든 CAP의 CAPID(중앙 액세스 정책) 식별자 배열을 반환합니다.
LsaGetLogonSessionData 지정된 로그온 세션에 대한 정보를 검색합니다.
LsaLogonUser 저장된 자격 증명 정보를 사용하여 보안 주체의 로그온 데이터를 인증합니다.
LsaLookupAuthenticationPackage 인증 패키지의 고유 식별자를 가져옵니다.
LsaLookupNames 사용자, 그룹 또는 로컬 그룹 이름의 배열에 해당하는 SID(보안 식별자)를 검색합니다.
LsaLookupPrivilegeValue 지정된 권한 이름을 나타내기 위해 LSA(로컬 보안 기관)에서 사용하는 LUID(로컬 고유 식별자)를 검색합니다.
LsaLookupSids SID(보안 식별자) 배열에 해당하는 이름을 조회합니다. LsaLookupSids가 SID에 해당하는 이름을 찾을 수 없는 경우 함수는 문자 형식으로 SID를 반환합니다.
LsaLookupSids2 SID(보안 식별자) 배열에 해당하는 이름을 조회하고 인터넷 공급자 ID를 지원합니다. LsaLookupSids2에서 SID에 해당하는 이름을 찾을 수 없는 경우 함수는 문자 형식으로 SID를 반환합니다.
LsaNtStatusToWinError LsaNtStatusToWinError 함수는 LSA 함수에서 반환된 NTSTATUS 코드를 Windows 오류 코드로 변환합니다.

LsaOpenPolicy 로컬 또는 원격 시스템의 Policy 개체에 대한 핸들을 엽니다.
LsaOpenTrustedDomainByName LsaOpenTrustedDomainByName 함수는 원격 신뢰할 수 있는 도메인의 LSA 정책 핸들을 엽니다. 원격 컴퓨터의 LSA 정책을 설정하거나 쿼리하기 위해 이 핸들을 LSA 함수 호출에 전달할 수 있습니다.
LsaQueryCAP 지정된 ID에 대한 중앙 액세스 정책(CAP)을 반환합니다.
LsaQueryDomainInformationPolicy Policyobject에서 도메인 정보를 검색합니다.
LsaQueryForestTrustInformation 지정된 로컬 보안 기관 TrustedDomain 개체에 대한 포리스트 트러스트 정보를 검색합니다.
LsaQueryInformationPolicy Policy 개체에 대한 정보를 검색합니다.
LsaQueryTrustedDomainInfo LsaQueryTrustedDomainInfo 함수는 신뢰할 수 있는 도메인에 대한 정보를 검색합니다.
LsaQueryTrustedDomainInfoByName LsaQueryTrustedDomainInfoByName 함수는 신뢰할 수 있는 도메인에 대한 정보를 반환합니다.
LsaRegisterLogonProcess LSA 서버에 대한 연결을 설정하고 호출자가 로그온 애플리케이션인지 확인합니다.
LsaRegisterPolicyChangeNotification LsaRegisterPolicyChangeNotification 함수는 LSA(로컬 보안 기관)에 이벤트 핸들을 등록합니다. 이 이벤트 핸들은 표시된 LSA 정책이 수정될 때마다 신호를 수신합니다.
LsaRemoveAccountRights 계정에서 하나 이상의 권한을 제거합니다.
LsaRetrievePrivateData

제네릭 데이터 암호화 및 암호 해독에는 LSA 프라이빗 데이터 함수를 사용하지 마세요. 대신 CryptProtectData 및 CryptUnprotectData 함수를 사용합니다. (LsaRetrievePrivateData)
LsaSetDomainInformationPolicy 도메인 정보를 Policyobject로 설정합니다.
LsaSetForestTrustInformation 지정된 로컬 보안 기관 TrustedDomain 개체에 대한 포리스트 트러스트 정보를 설정합니다.
LsaSetInformationPolicy Policy 개체의 정보를 수정합니다.
LsaSetTrustedDomainInfoByName LsaSetTrustedDomainInfoByName 함수는 TrustedDomain 개체의 값을 설정합니다.
LsaSetTrustedDomainInformation LsaSetTrustedDomainInformation 함수는 신뢰할 수 있는 도메인에 대한 정책 개체의 정보를 수정합니다.
LsaStorePrivateData 제네릭 데이터 암호화 및 암호 해독에는 LSA 프라이빗 데이터 함수를 사용하지 마세요. 대신 CryptProtectData 및 CryptUnprotectData 함수를 사용합니다. LSA 비밀을 조작해야 하는 경우에만 LSA 프라이빗 데이터 함수를 사용합니다(LsaStorePrivateData).
LsaUnregisterPolicyChangeNotification LsaUnregisterPolicyChangeNotification 함수는 이전에 등록된 알림 이벤트를 사용하지 않도록 설정합니다.
MakeAbsoluteSD 자체 상대 형식의 보안 설명자를 템플릿으로 사용하여 절대 형식의 보안 설명자를 만듭니다.
MakeSelfRelativeSD 절대 형식의 보안 설명자를 템플릿으로 사용하여 자체 상대 형식의 보안 설명자를 만듭니다.
MakeSignature 메시지의 암호화 체크섬을 생성하고 메시지 손실 또는 삽입을 방지하기 위한 시퀀싱 정보도 포함합니다.
MapGeneric MapGeneric 메서드는 액세스 마스크의 일반 액세스 권한을 해당 표준 및 특정 액세스 권한에 매핑하도록 요청합니다.

MapGenericMask 액세스 마스크의 일반 액세스 권한을 특정 및 표준 액세스 권한에 매핑합니다. 이 함수는 GENERIC_MAPPING 구조에 제공된 매핑을 적용합니다.
Msv1_0SubAuthenticationFilter 도메인 컨트롤러와 관련된 사용자 로그온 인증을 수행합니다.
Msv1_0SubAuthenticationRoutine 클라이언트/서버별 인증을 수행합니다.
Msv1_0SubAuthenticationRoutineEx LogonUser 함수를 호출하여 하위 인증을 요청할 때 원격 액세스 서비스 인증을 수행합니다.
Msv1_0SubAuthenticationRoutineGeneric LsaCallAuthenticationPackage 함수를 호출하여 하위 인증을 요청할 때 원격 액세스 서비스 인증을 수행합니다.
NameFromSid 지정된 SID(보안 식별자)에 해당하는 표시 이름을 가져옵니다.
NamesFromSids 지정된 SID(보안 식별자)에 해당하는 표시 이름을 가져옵니다.
NCryptCloseProtectionDescriptor 보호 설명자 개체를 0으로 해제하고 해당 핸들을 해제합니다.
NCryptCreateClaim 키 증명 클레임을 만듭니다.
NCryptCreatePersistedKey 새 키를 만들고 지정된 키 스토리지 공급자에 저장합니다.
NCryptCreateProtectionDescriptor 보호 설명자 개체에 대한 핸들을 검색합니다.
NCryptDecrypt 암호화된 데이터 블록을 해독합니다.

NCryptDeleteKey 스토리지에서 CNG 키를 삭제합니다.
NCryptDeriveKey 비밀 핸들에서 키를 파생합니다. (NCryptDeriveKey)
NCryptEncrypt 데이터 블록을 암호화합니다. (NCryptEncrypt)
NCryptEnumAlgorithms 지정된 키 스토리지 공급자가 지원하는 알고리즘의 이름을 가져옵니다.
NCryptEnumKeys 공급자가 저장한 키의 이름을 가져옵니다.
NCryptEnumStorageProviders 등록된 키 스토리지 공급자의 이름을 가져옵니다.
NCryptExportKey CNG 키를 메모리 BLOB으로 내보냅니다.
NCryptFinalizeKey CNG 키 스토리지 키를 완료합니다.
NCryptFreeBuffer CNG 키 스토리지 공급자가 할당한 메모리 블록을 해제합니다.
NCryptFreeObject CNG 키 스토리지 개체를 해제합니다.
NCryptGetProperty 키 스토리지 개체의 명명된 속성 값을 검색합니다.
NCryptGetProtectionDescriptorInfo 보호 설명자 규칙 문자열을 검색합니다.
NCryptImportKey

메모리 BLOB에서 암호화 API - CNG(차세대) 키를 가져옵니다.
NCryptIsAlgSupported CNG 키 스토리지 공급자가 특정 암호화 알고리즘을 지원하는지 여부를 결정합니다.
NCryptIsKeyHandle 지정된 핸들이 CNG 키 핸들인지 여부를 확인합니다.
NCryptKeyDerivation 지정된 키 파생 함수를 사용하여 다른 키에서 키를 만듭니다.
NCryptNotifyChangeKey 키 변경 알림을 만들거나 제거합니다.
NCryptOpenKey 지정된 CNG 키 스토리지 공급자에 있는 키를 엽니다.
NCryptOpenStorageProvider CNG 키 스토리지 공급자를 로드하고 초기화합니다.
NCryptProtectSecret 지정된 보호 설명자로 데이터를 암호화합니다.
NCryptQueryProtectionDescriptorName 등록된 설명자 표시 이름과 연결된 보호 설명자 규칙 문자열을 검색합니다.
NCryptRegisterProtectionDescriptorName 보호 설명자에 대한 표시 이름 및 연결된 규칙 문자열을 등록합니다.
NCryptSecretAgreement 프라이빗 및 공개 키에서 비밀 계약 값을 만듭니다. (NCryptSecretAgreement)
NCryptSetProperty CNG 키 스토리지 개체의 명명된 속성 값을 설정합니다.
NCryptSignHash 해시 값의 서명을 만듭니다. (NCryptSignHash)

NCryptStreamClose NCryptStreamOpenToProtect 또는 NCryptStreamOpenToUnprotect 함수를 사용하여 연 데이터 보호 스트림 개체를 닫습니다.
NCryptStreamOpenToProtect 지정된 보호 설명자에 대용량 데이터를 암호화하는 데 사용할 수 있는 스트림 개체를 엽니다.
NCryptStreamOpenToUnprotect 암호화에 사용되는 것과 동일한 보호 설명자로 대량의 데이터를 암호 해독하는 데 사용할 수 있는 스트림 개체를 엽니다. (NCryptStreamOpenToUnprotect)
NCryptStreamOpenToUnprotectEx 암호화에 사용되는 것과 동일한 보호 설명자로 대량의 데이터를 암호 해독하는 데 사용할 수 있는 스트림 개체를 엽니다. (NCryptStreamOpenToUnprotectEx)
NCryptStreamUpdate 데이터 블록을 암호화하고 암호를 해독합니다.
NCryptTranslateHandle CryptoAPI 핸들을 CNG 키 핸들로 변환합니다.
NCryptUnprotectSecret 지정된 보호 설명자로 데이터를 해독합니다.
NCryptVerifyClaim 키 증명 클레임을 확인합니다.
NCryptVerifySignature 지정된 서명이 지정된 해시와 일치하는지 확인합니다. (NCryptVerifySignature)
NetAddServiceAccount sMSA(독립 실행형 관리 서비스 계정)를 만들거나 gMSA(그룹 관리 서비스 계정)에 대한 자격 증명을 검색하고 로컬 컴퓨터에 계정 정보를 저장합니다.
NetEnumerateServiceAccounts 지정된 서버에서 독립 실행형 sMSA(관리 서비스 계정)를 열거합니다.

NetIsServiceAccount 지정된 SMSA(독립 실행형 관리 서비스 계정) 또는 gMSA(그룹 관리 서비스 계정)가 지정된 서버의 Netlogon 저장소에 있는지 테스트합니다.
NetQueryServiceAccount 지정된 관리 서비스 계정에 대한 정보를 가져옵니다.
NetRemoveServiceAccount 계정이 SMSA(독립 실행형 관리 서비스 계정)인 경우 Active Directory 데이터베이스에서 지정된 서비스 계정을 삭제합니다.
다음 구성 지점에서 사용 가능한 다음 인증서 서비스 서버 구성의 인덱스를 검색합니다. 이 메서드는 ICertConfig 인터페이스에서 처음 정의되었습니다.
다음 특성 열거형 시퀀스에서 다음 특성으로 이동합니다.
다음 열 열거형 시퀀스의 다음 열로 이동합니다.
다음 확장 열거형 시퀀스에서 다음 확장으로 이동합니다.
다음 행 열거형 시퀀스의 다음 행으로 이동합니다.
알림 SCEP 인증서 요청의 트랜잭션 상태를 플러그 인에 알립니다.
알림 종료 모듈에 이벤트가 발생했음을 알리기 위해 서버 엔진에서 호출됩니다.
NotifyBootConfigStatus 서비스 제어 관리자에 부팅 상태를 보고합니다. 부팅 확인 프로그램에서 사용됩니다.
NotifyServiceStatusChangeA

지정된 서비스를 만들거나 삭제하거나 상태가 변경될 때 애플리케이션이 알림을 받을 수 있도록 합니다. (ANSI)
NotifyServiceStatusChangeW 지정된 서비스를 만들거나 삭제하거나 상태가 변경될 때 애플리케이션이 알림을 받을 수 있도록 합니다. (유니코드)
NPAddConnection 로컬 디바이스를 네트워크 리소스에 연결합니다. (NPAddConnection)
NPAddConnection3 로컬 디바이스를 네트워크 리소스에 연결합니다. (NPAddConnection3)
NPCancelConnection 네트워크 연결을 끊습니다.
NPCloseEnum 열거형을 닫습니다.
NPDeviceMode 디바이스의 부모 창을 지정합니다. 이 창에는 디바이스에서 시작된 모든 대화 상자가 있습니다.
NPDirectoryNotify 네트워크 공급자에게 특정 디렉터리 작업을 알 수 있습니다.
NPEnumResource NPOpenEnum에서 반환된 핸들을 기반으로 열거형을 수행합니다.
NPFMXEditPerm 네트워크 공급업체가 자체 사용 권한 편집기 대화 상자를 제공할 수 있도록 합니다.
NPFMXGetPermCaps 권한 편집기의 기능을 검색합니다. 반환 값은 파일 관리자에서 사용할 보안 메뉴 항목을 나타내는 비트 마스크입니다.
NPFMXGetPermHelp 파일 관리자의 보안 메뉴에서 메뉴 항목을 선택하고 F1 키를 누르면 사용 권한 편집기 대화 상자의 도움말 파일 및 도움말 컨텍스트를 검색합니다.

NPFormatNetworkName 컨트롤에 표시할 네트워크 이름을 공급자별 형식으로 서식을 지정합니다.
NPGetCaps 네트워크에서 지원되는 서비스에 대한 정보를 반환합니다.
NPGetConnection 연결에 대한 정보를 검색합니다.
NPGetConnection3 현재 연결이 끊긴 경우에도 네트워크 연결에 대한 정보를 검색합니다.
NPGetConnectionPerformance 네트워크 리소스에 액세스하는 데 사용되는 연결의 예상 성능에 대한 정보를 반환합니다. 요청은 현재 연결된 네트워크 리소스에 대해서만 가능합니다.
NPGetDirectoryType 네트워크 디렉터리의 유형을 결정합니다.
NPGetPropertyText 네트워크 리소스의 속성 대화 상자에 추가할 단추의 이름을 검색합니다.
NPGetResourceInformation WNet API를 통해 액세스하는 네트워크 리소스의 부분을 리소스 유형과 관련된 API를 통해 액세스하는 부분과 구분합니다.
NPGetResourceParent 찾아보기 계층에서 지정된 네트워크 리소스의 부모를 검색합니다.
NPGetUniversalName 네트워크 리소스의 범용 이름을 검색합니다. NPGetUniversalName 함수는 UNC 형식 또는 이전 원격 이름 형식으로 이 유니버설 이름을 검색할 수 있습니다.
NPGetUser 네트워크 연결을 설정하는 데 사용되는 현재 기본 사용자 이름 또는 사용자 이름의 값을 검색합니다.
NPLogonNotify

MPR은 이 함수를 호출하여 자격 증명 관리자에게 로그온 이벤트가 발생했음을 알리고 자격 증명 관리자가 로그온 스크립트를 반환할 수 있도록 합니다.
NPOpenEnum 네트워크 리소스 또는 기존 연결의 열거형을 엽니다. 열거형에 유효한 핸들을 가져오려면 NPOpenEnum 함수를 호출해야 합니다.
NPPasswordChangeNotify MPR은 이 함수를 호출하여 자격 증명 관리자에게 암호 변경 이벤트를 알립니다.
NPPropertyDialog 사용자가 NPPropertyDialog 함수를 사용하여 추가된 단추를 클릭하면 호출됩니다. NPPropertyDialog 함수는 파일 및 디렉터리 네트워크 속성에 대해서만 호출됩니다.
NPSearchDialog 네트워크 공급업체는 연결 대화 상자에 표시된 계층적 보기를 넘어 고유한 형태의 검색 및 검색을 제공할 수 있습니다.
ObjectCloseAuditAlarmA 프라이빗 개체에 대한 핸들이 삭제되면 보안 이벤트 로그에 감사 메시지를 생성합니다. (ObjectCloseAuditAlarmA)
ObjectCloseAuditAlarmW 프라이빗 개체에 대한 핸들이 삭제되면 보안 이벤트 로그에 감사 메시지를 생성합니다. (ObjectCloseAuditAlarmW)
ObjectDeleteAuditAlarmA ObjectDeleteAuditAlarmA(ANSI) 함수(winbase.h)는 개체가 삭제될 때 감사 메시지를 생성합니다.
ObjectDeleteAuditAlarmW ObjectDeleteAuditAlarmW(유니코드) 함수(securitybaseapi.h)는 개체가 삭제될 때 감사 메시지를 생성합니다.
ObjectOpenAuditAlarmA 클라이언트 애플리케이션이 개체에 대한 액세스 권한을 얻거나 새 개체를 만들려고 할 때 감사 메시지를 생성합니다. (ObjectOpenAuditAlarmA)
ObjectOpenAuditAlarmW

클라이언트 애플리케이션이 개체에 대한 액세스 권한을 얻거나 새 개체를 만들려고 할 때 감사 메시지를 생성합니다. (ObjectOpenAuditAlarmW)
ObjectPrivilegeAuditAlarmA 보안 이벤트 로그에 감사 메시지를 생성합니다. (ObjectPrivilegeAuditAlarmA)
ObjectPrivilegeAuditAlarmW 보안 이벤트 로그에 감사 메시지를 생성합니다. (ObjectPrivilegeAuditAlarmW)
열기 인증 키를 엽니다. 인증 키에서 정보를 검색하거나, 인증서를 추가 또는 제거하거나, 인증 키를 내보내려면 먼저 인증 키를 열어야 합니다.
열기 기존 프라이빗 키를 엽니다.
OpenApplication 지정된 이름을 가진 IAzApplication 개체를 엽니다.
OpenApplication2 지정된 이름을 가진 IAzApplication2 개체를 엽니다.
OpenApplicationGroup 이름을 지정하여 IAzApplicationGroup 개체를 엽니다. (IAzApplication.OpenApplicationGroup)
OpenApplicationGroup 이름을 지정하여 IAzApplicationGroup 개체를 엽니다. (IAzAuthorizationStore.OpenApplicationGroup)
OpenApplicationGroup 이름을 지정하여 IAzApplicationGroup 개체를 엽니다. (IAzScope.OpenApplicationGroup)
OpenConnection 인증서 서비스 서버와의 연결을 설정합니다.
OpenElevatedEditor 사용자가 해당 편집 단추에 방패 이미지를 표시하는 액세스 제어 편집기 페이지에서 편집 단추를 클릭하면 액세스 제어 편집기를 엽니다.

OpenOperation 지정된 이름의 IAzOperation 개체를 엽니다.
OpenPersonalTrustDBDialog 인증서 대화 상자를 표시합니다. (OpenPersonalTrustDBDialog)
OpenPersonalTrustDBDialogEx 인증서 대화 상자를 표시합니다. (OpenPersonalTrustDBDialogEx)
OpenRole 지정된 이름을 가진 IAzRole 개체를 엽니다. (IAzApplication.OpenRole)
OpenRole 지정된 이름을 가진 IAzRole 개체를 엽니다. (IAzScope.OpenRole)
OpenRoleAssignment 지정된 이름의 IAzRoleAssignment 개체를 엽니다.
OpenRoleAssignment 이 범위에서 지정된 이름을 가진 IAzRoleAssignment 개체를 엽니다.
OpenRoleDefinition 지정된 이름을 가진 IAzRoleDefinition 개체를 엽니다.
OpenRoleDefinition 이 범위에서 지정된 이름을 가진 IAzRoleDefinition 개체를 엽니다.
OpenSCManagerA 지정된 컴퓨터에서 서비스 제어 관리자에 대한 연결을 설정하고 지정된 서비스 제어 관리자 데이터베이스를 엽니다. (ANSI)
OpenSCManagerW 지정된 컴퓨터에서 서비스 제어 관리자에 대한 연결을 설정하고 지정된 서비스 제어 관리자 데이터베이스를 엽니다. (유니코드)
openScope 지정된 이름의 IAzScope 개체를 엽니다.

OpenScope2 지정된 이름의 IAzScope2 개체를 엽니다.
OpenServiceA 기존 서비스를 엽니다. (ANSI)
OpenServiceW 기존 서비스를 엽니다. (유니코드)
OpenTask 지정된 이름을 가진 IAzTask 개체를 엽니다. (IAzApplication.OpenTask)
OpenTask 지정된 이름을 가진 IAzTask 개체를 엽니다. (IAzScope.OpenTask)
OpenView Certificate Services 데이터베이스에 대한 보기를 열고 IEnumCERTVIEWROW 개체의 인스턴스를 인스턴스화합니다.
PCRYPT_DECRYPT_PRIVATE_KEY_FUNC 프라이빗 키의 암호를 해독하고 pbClearTextKey 매개 변수에서 암호 해독된 키를 반환합니다.
PCRYPT_ENCRYPT_PRIVATE_KEY_FUNC 프라이빗 키를 암호화하고 pbEncryptedKey 매개 변수에서 암호화된 콘텐츠를 반환합니다.
PCRYPT_RESOLVE_HCRYPTPROV_FUNC phCryptProv 매개 변수를 사용하여 가져올 키를 수신하여 CSP(암호화 서비스 공급자)에 대한 핸들을 반환합니다.
pCryptSIPGetCaps SIP(주체 인터페이스 패키지)에 의해 구현되어 기능을 보고합니다.
PFN_CDF_PARSE_ERROR_CALLBACK CDF(카탈로그 정의 파일)를 구문 분석하는 동안 카탈로그 정의 함수 오류에 대해 호출됩니다.
PFN_CERT_CHAIN_FIND_BY_ISSUER_CALLBACK

애플리케이션이 인증서 체인에 추가될 수 있는 인증서를 필터링할 수 있도록 하는 애플리케이션 정의 콜백 함수입니다.
PFN_CERT_CREATE_CONTEXT_SORT_FUNC 컨텍스트를 만들 때 정렬된 각 컨텍스트 항목에 대해 호출됩니다.
PFN_CERT_DLL_OPEN_STORE_PROV_FUNC 저장소 공급자가 구현하고 저장소를 여는 데 사용됩니다.
PFN_CERT_ENUM_PHYSICAL_STORE CertEnumPhysicalStoreCallback 콜백 함수는 CertEnumPhysicalStore를 호출하여 찾은 각 물리적 저장소에 대한 정보를 형식화하고 제공합니다.
PFN_CERT_ENUM_SYSTEM_STORE CertEnumSystemStoreCallback 콜백 함수는 CertEnumSystemStore를 호출하여 찾은 각 시스템 저장소에 대한 정보를 형식화하고 제공합니다.
PFN_CERT_ENUM_SYSTEM_STORE_LOCATION CertEnumSystemStoreLocationCallback 콜백 함수는 CertEnumSystemStoreLocation 호출에서 찾은 각 시스템 저장소 위치에 대한 정보를 형식화하고 제공합니다.
PFN_CERT_STORE_PROV_CLOSE 저장소의 참조 수가 0으로 감소할 때 CertCloseStore에서 호출되는 애플리케이션 정의 콜백 함수입니다.
PFN_CERT_STORE_PROV_CONTROL CertStoreProvControl 콜백 함수는 CertControlStore API를 지원합니다. 모든 API의 매개 변수는 콜백으로 바로 전달됩니다. 자세한 내용은 CertControlStore를 참조하세요.
PFN_CERT_STORE_PROV_DELETE_CERT 저장소에서 인증서를 삭제하기 전에 CertDeleteCertificateFromStore에서 호출하는 애플리케이션 정의 콜백 함수입니다.
PFN_CERT_STORE_PROV_DELETE_CRL 저장소에서 CRL을 삭제하기 전에 CertDeleteCRLFromStore에서 호출하는 애플리케이션 정의 콜백 함수입니다.
PFN_CERT_STORE_PROV_READ_CERT 인증서 컨텍스트의 공급자 복사본을 읽는 애플리케이션 정의 콜백 함수입니다.

PFN_CERT_STORE_PROV_READ_CRL CRL 컨텍스트의 공급자 복사본을 읽는 애플리케이션 정의 콜백 함수입니다.
PFN_CERT_STORE_PROV_READ_CTL CertStoreProvReadCTL 콜백 함수는 CTL 컨텍스트의 공급자 복사본을 읽고 있는 경우 새 CTL 컨텍스트를 만들기 위해 호출됩니다.
PFN_CERT_STORE_PROV_SET_CERT_PROPERTY 인증서의 속성을 설정하기 전에 CertSetCertificateContextProperty에서 호출하는 애플리케이션 정의 콜백 함수입니다.
PFN_CERT_STORE_PROV_SET_CRL_PROPERTY CRL의 속성을 설정하기 전에 CertSetCRLContextProperty에서 호출하는 애플리케이션 정의 콜백 함수입니다.
PFN_CERT_STORE_PROV_SET_CTL_PROPERTY CertStoreProvSetCTLProperty 콜백 함수는 CTL에서 속성을 설정할 수 있는지 여부를 결정합니다.
PFN_CERT_STORE_PROV_WRITE_CERT 저장소에 추가하기 전에 CertAddEncodedCertificateToStore, CertAddCertificateContextToStore 및 CertAddSerializedElementToStore에서 호출하는 애플리케이션 정의 콜백 함수입니다.
PFN_CERT_STORE_PROV_WRITE_CRL 저장소에 추가하기 전에 CertAddEncodedCRLToStore, CertAddCRLContextToStore 및 CertAddSerializedElementToStore에서 호출하는 애플리케이션 정의 콜백 함수입니다.
PFN_CERT_STORE_PROV_WRITE_CTL CertStoreProvWriteCTL 콜백 함수는 CTL이 저장소에 추가되기 전에 CertAddEncodedCTLToStore, CertAddCTLContextToStore 또는 CertAddSerializedElementToStore에서 호출할 수 있습니다.
PFN_CMSG_CNG_IMPORT_CONTENT_ENCRYPT_KEY 이미 암호가 해독된 CEK(콘텐츠 암호화 키)를 가져옵니다.
PFN_CMSG_CNG_IMPORT_KEY_AGREE 키 계약 수신자를 위한 CEK(콘텐츠 암호화 키)의 암호를 해독합니다.
PFN_CMSG_CNG_IMPORT_KEY_TRANS 키 전송 수신자를 위한 CEK(콘텐츠 암호화 키)를 가져오고 암호 해독합니다.

PFN_CMSG_EXPORT_KEY_AGREE 봉투형 메시지의 키 계약 수신자에 대한 콘텐츠 암호화 키를 암호화하고 내보냅니다.
PFN_CMSG_EXPORT_KEY_TRANS 봉투형 메시지의 키 전송 받는 사람에 대한 콘텐츠 암호화 키를 암호화하고 내보냅니다.
PFN_CMSG_EXPORT_MAIL_LIST 봉투형 메시지의 메일 그룹 받는 사람에 대한 콘텐츠 암호화 키를 암호화하고 내보냅니다.
PFN_CMSG_GEN_CONTENT_ENCRYPT_KEY 봉투형 메시지의 콘텐츠를 암호화하는 데 사용되는 대칭 키를 생성합니다.
PFN_CMSG_IMPORT_KEY_AGREE 봉투형 메시지의 키 전송 받는 사람에 대한 콘텐츠 암호화 키를 가져옵니다. (PFN_CMSG_IMPORT_KEY_AGREE)
PFN_CMSG_IMPORT_KEY_TRANS 봉투형 메시지의 키 전송 받는 사람에 대한 콘텐츠 암호화 키를 가져옵니다. (PFN_CMSG_IMPORT_KEY_TRANS)
PFN_CMSG_IMPORT_MAIL_LIST 봉투형 메시지의 키 전송 받는 사람에 대한 콘텐츠 암호화 키를 가져옵니다. (PFN_CMSG_IMPORT_MAIL_LIST)
PFN_CRYPT_ENUM_KEYID_PROP CRYPT_ENUM_KEYID_PROP 콜백 함수는 CryptEnumKeyIdentifierProperties 함수와 함께 사용됩니다.
PFN_CRYPT_ENUM_OID_FUNC CRYPT_ENUM_OID_FUNCTION 콜백 함수는 CryptEnumOIDFunction 함수와 함께 사용됩니다.
PFN_CRYPT_ENUM_OID_INFO CRYPT_ENUM_OID_INFO 콜백 함수는 CryptEnumOIDInfo 함수와 함께 사용됩니다.
PFN_CRYPT_EXPORT_PUBLIC_KEY_INFO_EX2_FUNC 공개 키 BLOB을 내보내고 인코딩하기 위해 CryptExportPublicKeyInfoEx에서 호출됩니다.
PFN_CRYPT_EXTRACT_ENCODED_SIGNATURE_PARAMETERS_FUNC

해시 알고리즘 식별자 및 선택적으로 서명 매개 변수를 디코딩하고 반환하기 위해 호출됩니다.
PFN_CRYPT_GET_SIGNER_CERTIFICATE CryptGetSignerCertificateCallback 사용자가 제공한 콜백 함수는 CRYPT_VERIFY_MESSAGE_PARA 구조와 함께 사용하여 메시지 서명자의 인증서를 가져와서 확인합니다.
PFN_CRYPT_OBJECT_LOCATOR_PROVIDER_FLUSH 개체가 변경되었음을 지정합니다.
PFN_CRYPT_OBJECT_LOCATOR_PROVIDER_FREE 공급자가 반환한 개체를 해제합니다.
PFN_CRYPT_OBJECT_LOCATOR_PROVIDER_FREE_IDENTIFIER 개체 식별자에 대한 메모리를 해제합니다.
PFN_CRYPT_OBJECT_LOCATOR_PROVIDER_FREE_PASSWORD PFX(개인 정보 교환) 바이트 배열을 암호화하는 데 사용되는 암호를 해제합니다.
PFN_CRYPT_OBJECT_LOCATOR_PROVIDER_GET 개체를 검색합니다.
PFN_CRYPT_OBJECT_LOCATOR_PROVIDER_INITIALIZE 공급자를 초기화합니다.
PFN_CRYPT_OBJECT_LOCATOR_PROVIDER_RELEASE 공급자를 해제합니다.
PFN_CRYPT_SIGN_AND_ENCODE_HASH_FUNC 계산된 해시에 서명하고 인코딩하기 위해 호출됩니다.
PFN_CRYPT_VERIFY_ENCODED_SIGNATURE_FUNC 인코딩된 서명을 해독하고 계산된 해시와 비교하기 위해 호출됩니다.
PFN_CRYPT_XML_CREATE_TRANSFORM 지정된 데이터 공급자에 대한 변환을 만듭니다.
PFN_CRYPT_XML_DATA_PROVIDER_CLOSE 데이터 공급자를 해제합니다.

PFN_CRYPT_XML_DATA_PROVIDER_READ XML 데이터를 읽습니다.
PFN_CRYPT_XML_ENUM_ALG_INFO 미리 정의되고 등록된 CRYPT_XML_ALGORITHM_INFO 항목을 열거합니다.
PFN_CRYPT_XML_WRITE_CALLBACK XML 데이터를 씁니다.
PFN_IMPORT_PUBLIC_KEY_INFO_EX2_FUNC 공개 키 알고리즘 식별자를 디코딩하고, 알고리즘 공급자를 로드하고, 키 쌍을 가져오기 위해 CryptImportPublicKeyInfoEx2에서 호출됩니다.
PFNCFILTERPROC CryptUIWizDigitalSign 함수에 의해 표시되는 디지털 서명 마법사에 표시되는 인증서를 필터링하는 애플리케이션 정의 콜백 함수입니다.
PFNCMFILTERPROC 각 인증서를 필터링하여 인증서가 CertSelectCertificate 함수에 의해 표시되는 인증서 선택 대화 상자에 표시되는지 여부를 확인합니다.
PFNCMHOOKPROC CertSelectCertificate 함수에 의해 생성된 인증서 선택 대화 상자에서 메시지를 처리하기 전에 호출됩니다.
PFNCryptStreamOutputCallback NCryptStreamOpenToProtect 또는 NCryptStreamOpenToUnprotect 함수를 사용하여 시작한 작업에서 암호화되거나 암호 해독된 데이터를 받습니다.
pfnIsFileSupported 레지스트리에 나열된 SIP(주체 인터페이스 패키지)를 쿼리하여 파일 형식을 처리하는 SIP를 결정합니다. (pfnIsFileSupported)
pfnIsFileSupportedName 레지스트리에 나열된 SIP(주체 인터페이스 패키지)를 쿼리하여 파일 형식을 처리하는 SIP를 결정합니다. (pfnIsFileSupportedName)
PFSCE_FREE_INFO PFSCE_QUERY_INFO 호출할 때 Security Configuration 도구 집합에서 할당한 버퍼의 메모리를 해제합니다.

PFSCE_LOG_INFO 구성 로그 파일 또는 분석 로그 파일에 메시지를 기록합니다.
PFSCE_QUERY_INFO 보안 구성 파일 또는 분석 데이터베이스에서 서비스 관련 정보를 쿼리합니다.
PFSCE_SET_INFO 서비스별 구성 및 분석 정보를 설정하거나 덮어씁니다.
PFXExportCertStore 인증서를 내보내고, 사용 가능한 경우 참조된 인증서 저장소에서 연결된 프라이빗 키를 내보냅니다.
PFXExportCertStoreEx 인증서를 내보내고, 사용 가능한 경우 참조된 인증서 저장소에서 연결된 프라이빗 키를 내보냅니다.
PFXImportCertStore PFX BLOB을 가져오고 인증서 및 연결된 프라이빗 키가 포함된 저장소의 핸들을 반환합니다.
PFXIsPFXBlob PFXIsPFXBlob 함수는 BLOB의 외부 계층을 PFX 패킷으로 디코딩하려고 시도합니다.
PFXVerifyPassword PFXVerifyPassword 함수는 BLOB의 외부 계층을 PFX(개인 정보 교환) 패킷으로 디코딩하고 지정된 암호로 암호를 해독하려고 시도합니다. BLOB에서 데이터를 가져오지 않습니다.
핑 OCSP(온라인 인증서 상태 프로토콜) 응답자 서비스를 사용하여 DCOM 연결을 테스트합니다.
PostUnInstall 구현되지 않으며 나중에 사용하도록 예약되어 있습니다.
PostUnInstall 구현되지 않았습니다. 나중에 사용할 수 있습니다.
PreUnInstall 역할별 상태 정보를 일시적으로 저장한 다음 역할을 제거합니다.

PreUnInstall NDES(네트워크 디바이스 등록 서비스) 역할에 대한 레지스트리 및 IIS 설정을 제거합니다.
PrivilegeCheck 액세스 토큰에서 지정된 권한 집합을 사용할 수 있는지 여부를 결정합니다.
PrivilegedServiceAuditAlarmA 보안 이벤트 로그에 감사 메시지를 생성합니다. (PrivilegedServiceAuditAlarmA)
PrivilegedServiceAuditAlarmW 보안 이벤트 로그에 감사 메시지를 생성합니다. (PrivilegedServiceAuditAlarmW)
ProcessResponseMessage 응답 메시지를 처리하고 메시지 처리를 반환합니다.
PropertySheetPageCallback PropertySheetPageCallback 메서드는 EditSecurity 또는 CreateSecurityPage 호출자에게 액세스 제어 편집기 속성 페이지가 생성되거나 제거되고 있음을 알립니다.
PSAM_INIT_NOTIFICATION_ROUTINE InitializeChangeNotify 함수는 암호 필터 DLL에 의해 구현됩니다. 이 함수는 DLL을 초기화합니다.
PSAM_PASSWORD_FILTER_ROUTINE 암호 필터 DLL에 의해 구현됩니다. 이 함수에서 반환된 값은 시스템에서 새 암호를 허용하는지 여부를 결정합니다.
PSAM_PASSWORD_NOTIFICATION_ROUTINE 암호 필터 DLL에 의해 구현됩니다. 암호가 변경되었음을 DLL에 알릴 수 있습니다.
PstAcquirePrivateKey 호출자의 프라이빗 키를 지정된 인증서와 연결합니다.
PstGetCertificates 지정된 서버에서 사용자를 인증하는 데 사용할 수 있는 인증서를 지정하는 인증서 체인을 검색합니다.
PstGetTrustAnchors 지정된 서버에서 신뢰할 수 있는 CA(인증 기관) 목록을 검색합니다.

PstGetUserNameForCertificate 지정된 인증서와 연결된 사용자 이름을 검색합니다.
PstMapCertificate 지정된 인증서와 연결된 사용자 토큰을 만드는 데 사용할 수 있는 정보를 지정하는 구조를 검색합니다.
PstValidate 지정된 인증서의 유효성을 검사합니다.
PublishCRL 새 CRL(인증서 해지 목록)을 게시하기 위해 CA(인증서 서비스 인증 기관)에 요청을 보냅니다. 이 메서드는 ICertAdmin 인터페이스에서 처음 도입되었습니다.
PublishCRL CA(인증 기관)에 대한 CRL(인증서 해지 목록)을 게시합니다.
put_Algorithm 공개 키 알고리즘에 대한 OID(개체 식별자)를 지정하거나 검색합니다. (Put)
put_AlternateSignatureAlgorithm PKCS(Put)에 대한 OID(서명 알고리즘 개체 식별자) 여부를 나타내는 부울 값을 지정하고 검색합니다.
put_AlternateSignatureAlgorithm GetSignatureAlgorithm 메서드가 PKCS(Put)에 대한 OID(불연속 또는 결합 알고리즘 개체 식별자)를 검색해야 하는지 여부를 지정하는 부울 값을 지정하고 검색합니다.
put_ApplicationData 애플리케이션에서 정보를 저장하는 데 사용할 수 있는 불투명 필드를 설정하거나 검색합니다. (IAzApplication.put_ApplicationData)
put_ApplicationData 애플리케이션에서 정보를 저장하는 데 사용할 수 있는 불투명 필드를 설정하거나 검색합니다. (IAzAuthorizationStore.put_ApplicationData)
put_ApplicationData IAzOperation의 ApplicationData 속성은 애플리케이션에서 정보를 저장하는 데 사용할 수 있는 불투명 필드를 설정하거나 검색합니다. (Put)

put_ApplicationData IAzRole의 ApplicationData 속성은 애플리케이션에서 정보를 저장하는 데 사용할 수 있는 불투명 필드를 설정하거나 검색합니다. (Put)
put_ApplicationData IAzScope의 ApplicationData 속성은 애플리케이션에서 정보를 저장하는 데 사용할 수 있는 불투명 필드를 설정하거나 검색합니다. (Put)
put_ApplicationData IAzTask의 ApplicationData 속성은 애플리케이션에서 정보를 저장하는 데 사용할 수 있는 불투명 필드를 설정하거나 검색합니다. (Put)
put_ApplyStoreSacl 권한 부여 저장소를 수정할 때 정책 감사를 생성해야 하는지 여부를 나타내는 값을 설정하거나 검색합니다. (IAzApplication.put_ApplyStoreSacl)
put_ApplyStoreSacl 권한 부여 저장소를 수정할 때 정책 감사를 생성해야 하는지 여부를 나타내는 값을 설정하거나 검색합니다. (IAzAuthorizationStore.put_ApplyStoreSacl)
put_ArchivePrivateKey CA(인증 기관)에서 프라이빗 키를 보관할지 여부를 나타내는 부울 값을 지정하거나 검색합니다. (Put)
put_AttestationEncryptionCertificate 클라이언트에서 EKPUB 및 EKCERT 값을 암호화하는 데 사용되는 인증서입니다. 이 속성은 신뢰할 수 있는 컴퓨터 루트에 연결된 유효한 인증서로 설정해야 합니다. (Put)
put_AttestPrivateKey 만든 프라이빗 키를 증명해야 하는 경우 True입니다. 그렇지 않으면 false입니다. true이면 AttestationEncryptionCertificate 속성이 설정된 것으로 예상됩니다. (Put)
put_AuthFlags 클라이언트가 CEP(인증서 등록 정책) 서버에 인증하는 데 사용하는 인증 유형을 나타내는 값을 지정하고 검색합니다. (Put)
put_AuthzInterfaceClsid UI(사용자 인터페이스)가 애플리케이션별 작업을 수행하는 데 사용하는 인터페이스의 CLSID(클래스 식별자)를 설정하거나 검색합니다. (Put)

put_BizRule 이 애플리케이션 그룹의 멤버 자격을 결정하는 스크립트를 가져오거나 설정합니다. (Put)
put_BizRule 비즈니스 규칙(BizRule)을 구현하는 스크립트의 텍스트를 설정하거나 검색합니다. (Put)
put_BizRuleImportedPath 이 애플리케이션 그룹과 연결된 비즈니스 규칙 스크립트를 포함하는 파일의 경로를 가져오거나 설정합니다. (Put)
put_BizRuleImportedPath 비즈니스 규칙(BizRule)을 가져오는 파일의 경로를 설정하거나 검색합니다. (Put)
put_BizRuleLanguage 이 애플리케이션 그룹과 연결된 비즈니스 규칙 스크립트의 프로그래밍 언어를 가져오거나 설정합니다. (Put)
put_BizRuleLanguage 비즈니스 규칙(BizRule)이 구현되는 스크립팅 언어를 설정하거나 검색합니다. (Put)
put_BizRulesEnabled 이 애플리케이션에 대해 비즈니스 규칙을 사용할 수 있는지 여부를 나타내는 값을 가져오거나 설정합니다. (Put)
put_BusinessRuleResult 비즈니스 규칙(BizRule)에서 사용자가 요청된 작업을 수행할 수 있는지 여부를 나타내는 값을 설정합니다.
put_BusinessRuleString 비즈니스 규칙(BizRule)에 대한 애플리케이션별 문자열을 설정하거나 검색합니다. (Put)
put_CAConfig 서명 인증서에 서명해야 하는 CA(인증 기관) 이름을 가져오거나 설정합니다. (Put)
put_CAStoreFlags 저장소를 열 때 CA(인증 기관) 저장소를 제어하는 플래그를 설정하거나 검색합니다. (Put)
put_CAStoreFlags

IEnroll4의 CAStoreFlags 속성은 저장소가 열릴 때 CA(인증 기관) 저장소를 제어하는 플래그를 설정하거나 검색합니다. (Put)
put_CAStoreName 모든 비"ROOT" 및 비"MY" 인증서가 유지되는 저장소의 이름을 설정하거나 검색합니다. (Put)
put_CAStoreNameWStr IEnroll4의 CAStoreNameWStr 속성은 모든 비"ROOT" 및 비"MY" 인증서가 유지되는 저장소의 이름을 설정하거나 검색합니다. (Put)
put_CAStoreType CAStoreName 속성에 지정된 저장소에 사용할 저장소 유형을 설정하거나 검색합니다. (Put)
put_CAStoreTypeWStr CAStoreNameWStr 속성에 지정된 저장소에 사용할 저장소 유형을 설정하거나 검색합니다. (Put)
put_Certificate 프라이빗 키와 연결된 인증서를 포함하는 바이트 배열을 지정하거나 검색합니다. (Put)
put_CertificateDescription 인증서에 대한 설명이 포함된 문자열을 지정하거나 검색합니다. (Put)
put_CertificateFriendlyName 인증서의 표시 이름을 지정하거나 검색합니다. (Put)
put_CertificateFriendlyName 인증서의 이름을 가져오거나 설정합니다. (Put)
put_ChallengePassword 챌린지가 있는 요청을 만들 때 사용할 암호입니다. 챌린지 없이 요청을 만들려면 ChallengePassword 속성을 설정하지 마세요. (Put)
put_ClientId 요청을 만든 실행 파일을 식별하는 값을 지정하고 검색합니다. (Put)
put_ClientId 클라이언트 ID 요청 특성을 설정하거나 검색합니다. 클라이언트 ID 요청 특성은 인증서 요청의 원본을 나타냅니다. 이 속성은 ICEnroll4 인터페이스에서 처음 정의되었습니다. (Put)

put_ClientId ClientId 속성은 클라이언트 ID 요청 특성을 설정하거나 검색합니다. 클라이언트 ID 요청 특성은 인증서 요청의 원본을 나타냅니다. 이 속성은 IEnroll4 인터페이스에서 처음 정의되었습니다. (Put)
put_ContainerName 키 컨테이너의 이름을 지정하거나 검색합니다. (Put)
put_ContainerName CSP(암호화 서비스 공급자)가 키를 생성, 저장 또는 액세스하는 데 사용하는 이름을 가져오거나 설정합니다. (Put)
put_ContainerName ICEnroll4의 ContainerName 속성은 사용할 키 컨테이너의 이름을 설정하거나 검색합니다. (Put)
put_ContainerNamePrefix 키 컨테이너의 이름에 추가된 접두사를 지정하거나 검색합니다. (Put)
put_ContainerNameWStr 사용할 키 컨테이너의 이름을 설정하거나 검색합니다. (Put)
put_Cost 인증서 등록 정책 서버에 연결하기 위한 임의의 비용을 지정하고 검색합니다. (IX509EnrollmentPolicyServer.put_Cost)
put_Cost 인증서 등록 정책 서버에 연결하기 위한 임의의 비용을 지정하고 검색합니다. (IX509PolicyServerUrl.put_Cost)
put_Critical 인증서 확장이 중요한지 여부를 식별하는 부울 값을 지정하고 검색합니다. (Put)
put_CspInformations 요청 개체에서 사용할 수 있는 암호화 공급자의 컬렉션을 지정하고 검색합니다. (Put)
put_CspInformations 프라이빗 키와 연결된 공개 키 알고리즘을 지원하는 사용 가능한 암호화 공급자에 대한 정보를 포함하는 ICspInformation 개체의 컬렉션을 지정하거나 검색합니다. (Put)

put_CspStatus 프라이빗 키와 연결된 암호화 공급자 및 알고리즘 쌍에 대한 정보를 포함하는 ICspStatus 개체를 지정하거나 검색합니다. (Put)
put_Default 기본 CEP(인증서 등록 정책) 서버인지 여부를 나타내는 부울 값을 지정하고 검색합니다. (Put)
put_DeleteRequestCert 요청 저장소의 더미 인증서가 삭제되는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (Put)
put_DeleteRequestCert IEnroll4의 DeleteRequestCert 속성은 요청 저장소의 더미 인증서가 삭제되는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (Put)
put_Description 애플리케이션을 설명하는 주석을 설정하거나 검색합니다. (Put)
put_Description 애플리케이션 그룹을 설명하는 주석을 설정하거나 검색합니다. (Put)
put_Description 작업을 설명하는 주석을 설정하거나 검색합니다. (Put)
put_Description IAzOperation의 Description 속성은 작업을 설명하는 주석을 설정하거나 검색합니다. (Put)
put_Description 역할을 설명하는 주석을 설정하거나 검색합니다. (Put)
put_Description 범위를 설명하는 주석을 설정하거나 검색합니다. (Put)
put_Description 작업을 설명하는 주석을 설정하거나 검색합니다. (Put)
put_Description 프라이빗 키에 대한 설명이 포함된 문자열을 지정하거나 검색합니다. (Put)

put_Display 사용자 인터페이스에 상태 정보를 표시할지 여부를 나타내는 값을 지정하거나 검색합니다. (Put)
put_DomainTimeout 도메인에 연결할 수 없는 것으로 결정되는 시간(밀리초)을 설정하거나 검색합니다. (Put)
put_EnableSMIMECapabilities ICEnroll4::EnableSMIMECapabilities 속성은 PKCS(Put) 여부를 제어합니다.
put_EnableSMIMECapabilities PKCS(Put) 여부를 제어합니다.
put_EnableT61DNEncoding ICEnroll4의 EnableT61DNEncoding 속성은 요청의 고유 이름이 유니코드 문자열이 아닌 T61 문자열로 인코딩되는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (Put)
put_EnableT61DNEncoding 요청의 고유 이름이 유니코드 문자열이 아닌 T61 문자열로 인코딩되는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (Put)
put_EncryptionAlgorithm 보관할 프라이빗 키를 암호화하는 데 사용되는 알고리즘의 OID(개체 식별자)를 지정하거나 검색합니다. (Put)
put_EncryptionAlgorithm 클라이언트에서 EKPUB 및 EKCERT 값을 암호화하는 데 사용되는 암호화 알고리즘입니다. (Put)
put_EncryptionStrength 보관할 프라이빗 키에 적용되는 상대 암호화 수준을 지정하거나 검색합니다. (Put)
put_EncryptionStrength 암호화에 사용할 EncryptionAlgorithm의 비트 길이를 식별합니다. EncryptionAlgorithm이 1비트 길이만 지원하는 경우 EncryptionStrength 속성에 대한 값을 지정할 필요가 없습니다. (Put)
put_Error 인증서 등록 프로세스의 오류 상태를 식별하는 값을 지정하고 검색합니다. (Put)

put_Existing 프라이빗 키를 만들거나 가져왔는지 여부를 나타내는 부울 값을 지정하거나 검색합니다. (Put)
put_Existing 프라이빗 키가 이미 있는지 여부를 나타내는 값을 가져오거나 설정합니다. (Put)
put_ExistingCACertificate DER(고유 인코딩 규칙)을 사용하여 인코딩되었으며 기존 키에 해당하는 CA(인증 기관) 인증서의 이진 값인 이진 값을 가져오거나 설정합니다. (Put)
put_ExportPolicy 프라이빗 키에 대한 내보내기 제약 조건을 지정하거나 검색합니다. (Put)
put_Flags CEP(인증서 등록 정책) 서버 정책 정보를 그룹 정책, 레지스트리 또는 둘 다에서 로드할 수 있는지 여부를 나타내는 값을 지정하거나 검색합니다. (Put)
put_FriendlyName 개체 식별자의 표시 이름을 지정하고 검색합니다. (Put)
put_FriendlyName 프라이빗 키의 표시 이름을 지정하거나 검색합니다. (Put)
put_GenerateAudits IAzApplication의 GenerateAudits 속성은 런타임 감사를 생성해야 하는지 여부를 나타내는 값을 설정하거나 검색합니다. (Put)
put_GenerateAudits 런타임 감사를 생성해야 하는지 여부를 나타내는 값을 설정하거나 검색합니다. (Put)
put_GenKeyFlags 인증서 요청이 생성될 때 CryptGenKey 함수에 전달된 값을 설정하거나 검색합니다. (Put)
put_GenKeyFlags 인증서 요청이 생성될 때 CryptGenKey에 전달된 값을 설정하거나 검색합니다. (Put)
put_HashAlgID

PKCS(Put)에 서명할 때 사용되는 해시 알고리즘을 설정하거나 검색합니다.
put_HashAlgID IEnroll4의 HashAlgID 속성은 PKCS(Put)에 서명할 때 사용되는 해시 알고리즘을 설정하거나 검색합니다.
put_HashAlgorithm 인증서 요청에 서명하는 데 사용되는 해시 알고리즘의 OID(개체 식별자)를 지정하고 검색합니다. (Put)
put_HashAlgorithm GetSignatureAlgorithm 메서드에 사용되는 해시 알고리즘에 대한 OID(개체 식별자)를 지정하고 검색합니다. (Put)
put_HashAlgorithm 키에 대한 CA(인증 기관) 인증서에 서명하거나 확인하는 데 사용되는 해시 알고리즘의 이름을 가져오거나 설정합니다. (Put)
put_HashAlgorithm PKCS(ICEnroll.put_HashAlgorithm)에 서명하는 데 사용되는 서명 해시 알고리즘만 설정하거나 검색합니다.
put_HashAlgorithm 인증서에 서명하는 데 사용되는 해시 알고리즘의 식별자를 가져오거나 설정합니다. (Put)
put_HashAlgorithmWStr PKCS(IEnroll.put_HashAlgorithmWStr)에 서명하는 데 사용되는 서명 해시 알고리즘만 설정하거나 검색합니다.
put_IncludeSubjectKeyID 주체 키 ID 확장이 생성된 인증서 요청에 추가되는지 여부를 결정합니다. (Put)
put_IncludeSubjectKeyID IEnroll4의 IncludeSubjectKeyID 속성은 주체 키 ID 확장이 생성된 인증서 요청에 추가되는지 여부를 결정합니다. (Put)
put_IsRoleDefinition 태스크가 역할 정의인지 여부를 나타내는 값을 설정하거나 검색합니다. (Put)
put_Issuer 인증서 발급자의 이름을 지정하거나 검색합니다. (Put)

put_KeyArchivalCertificate CA(인증 기관) 암호화 인증서를 지정하거나 검색합니다. (Put)
put_KeyContainerNamePrefix 새 프라이빗 키의 컨테이너 이름을 만드는 데 사용되는 접두사를 지정하거나 검색합니다. (Put)
put_KeyProtection 사용하기 전에 프라이빗 키를 보호하는 방법을 나타내는 값을 지정하거나 검색합니다. (Put)
put_KeySpec 프라이빗 키를 서명, 암호화 또는 둘 다에 사용할 수 있는지 여부를 식별하는 값을 지정하거나 검색합니다. (Put)
put_KeySpec ICEnroll4의 KeySpec 속성은 생성된 키 유형을 설정하거나 검색합니다. (Put)
put_KeySpec 생성된 키의 형식을 설정하거나 검색합니다. (Put)
put_KeyUsage 프라이빗 키를 사용할 수 있는 특정 용도를 식별하는 값을 지정하거나 검색합니다. (Put)
put_LdapQuery LDAP 쿼리 애플리케이션 그룹에 대한 멤버 자격을 정의하는 데 사용되는 LDAP(Lightweight Directory Access Protocol) 쿼리를 설정하거나 검색합니다. (Put)
put_LDAPQueryDN LDAP 쿼리 그룹을 평가하는 동안 사용할 디렉터리 개체의 도메인 이름을 검색하거나 설정합니다. (Put)
put_LegacyCsp 공급자가 CryptoAPI(레거시) CSP(암호화 서비스 공급자)인지 여부를 나타내는 부울 값을 지정하거나 검색합니다. (Put)
put_Length 프라이빗 키의 길이를 비트 단위로 지정하거나 검색합니다. (Put)
put_Length

CSP(암호화 서비스 공급자)에서 지원하는 값 중 하나에 대한 키의 강도를 가져오거나 설정합니다. (Put)
put_LimitExchangeKeyToEncipherment AT_KEYEXCHANGE 요청에 디지털 서명 및 비공제 키 사용이 포함되어 있는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (Put)
put_LimitExchangeKeyToEncipherment IEnroll4의 LimitExchangeKeyToEncipherment 속성은 AT_KEYEXCHANGE 요청에 디지털 서명 및 비공급 키 사용이 포함되어 있는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (Put)
put_LocalRevocationInformation 로컬 컴퓨터의 CRL(인증서 해지 목록)을 가져오거나 설정합니다. (Put)
put_MachineContext 로컬 인증서 저장소 컨텍스트를 식별하는 부울 값을 지정하거나 검색합니다. (Put)
put_MaxScriptEngines 캐시할 BizRule(비즈니스 규칙) 스크립트 엔진의 최대 수를 설정하거나 검색합니다. (Put)
put_MyStoreFlags 내 저장소에 사용되는 레지스트리 위치를 설정하거나 검색합니다. (Put)
put_MyStoreFlags 내 저장소에 사용되는 레지스트리 위치를 설정하거나 검색합니다. (Put)
put_MyStoreName 연결된 프라이빗 키가 있는 인증서가 보관되는 저장소의 이름을 설정하거나 검색합니다. (Put)
put_MyStoreNameWStr IEnroll4의 MyStoreNameWStr 속성은 연결된 프라이빗 키가 있는 인증서가 보관되는 저장소의 이름을 설정하거나 검색합니다. (Put)
put_MyStoreType MyStoreName 속성에 지정된 저장소 유형을 설정하거나 검색합니다. (Put)
put_MyStoreTypeWStr MyStoreTypeWStr 속성에 지정된 저장소 유형을 설정하거나 검색합니다. (Put)

put_Name 애플리케이션의 이름을 설정하거나 검색합니다. (Put)
put_Name 애플리케이션 그룹의 이름을 설정하거나 검색합니다. (Put)
put_Name 작업의 이름을 설정하거나 검색합니다. (Put)
put_Name 역할의 이름을 설정하거나 검색합니다. (Put)
put_Name 범위의 이름을 설정하거나 검색합니다. (Put)
put_Name 작업의 이름을 설정하거나 검색합니다. (Put)
put_NotAfter 인증서가 더 이상 유효하지 않은 날짜와 시간을 지정하거나 검색합니다. (Put)
put_NotBefore 인증서가 유효하지 않은 날짜와 시간을 지정하거나 검색합니다. (Put)
put_NullSigned 인증서 요청이 null로 서명되었는지 여부를 나타내는 부울 값을 지정하고 검색합니다. (Put)
put_OldCertificate 요청을 바꾸려는 이전 인증서를 가져오거나 설정합니다. (Put)
put_OperationID 애플리케이션 내에서 작업을 고유하게 식별하는 애플리케이션별 값을 설정하거나 검색합니다. (Put)
put_Ordinal ICspStatuses 컬렉션에서 ICspStatus 개체의 위치를 지정하거나 검색합니다. (Put)
put_Parameters

서명 알고리즘과 연결된 매개 변수를 포함하는 바이트 배열을 검색합니다. (Put)
put_ParentWindow 서명 인증서 정보를 표시하는 데 사용되는 창의 ID를 지정하거나 검색합니다. (Put)
put_ParentWindow 키 관련 사용자 인터페이스 대화 상자에서 사용하는 창의 ID를 지정하고 검색합니다. (Put)
put_ParentWindow 등록 정보를 표시하는 데 사용되는 창의 ID를 지정하거나 검색합니다. (Put)
put_ParentWindow 키 정보를 표시하는 데 사용되는 창의 ID를 지정하거나 검색합니다. (Put)
put_Pin 스마트 카드 사용자를 인증하는 데 사용되는 PIN(개인 식별 번호)을 지정합니다.
put_Pin 스마트 카드의 프라이빗 키 컨테이너에 액세스하기 전에 사용자를 인증하는 데 사용되는 PIN(개인 식별 번호)을 지정합니다.
put_PrivateKeyArchiveCertificate PKCS(Put)를 사용하여 프라이빗 키를 보관하는 데 사용되는 인증서를 설정하거나 검색합니다.
put_Property IX509CertificateTemplateWritable 개체의 속성 값을 지정하거나 검색합니다. (Put)
put_PropertyId 외부 인증서 속성을 식별하는 CERTENROLL_PROPERTYID 열거형의 값을 지정하거나 검색합니다. (Put)
put_ProviderCLSID CA 구성에서 사용하는 해지 정보 공급자의 CLSID를 가져오거나 설정합니다. (Put)
put_ProviderFlags 공급자 유형을 설정하거나 검색합니다. (Put)
put_ProviderFlags IEnroll4의 ProviderFlags 속성은 공급자 유형을 설정하거나 검색합니다. (Put)

put_ProviderName 암호화 공급자의 이름입니다. 기본값은 Microsoft 플랫폼 암호화 공급자입니다. Open 메서드를 호출하기 전에 ProviderName 속성을 설정해야 합니다. Open 메서드를 호출한 후에는 ProviderName 속성을 변경할 수 없습니다. (Put)
put_ProviderName 암호화 공급자의 이름을 지정하거나 검색합니다. (Put)
put_ProviderName 프라이빗 키를 생성하거나 저장하는 데 사용되는 CSP(암호화 서비스 공급자) 또는 KSP(키 스토리지 공급자)의 이름을 가져오거나 설정합니다. (Put)
put_ProviderName ICEnroll4의 ProviderName 속성은 사용할 CSP(암호화 서비스 공급자)의 이름을 설정하거나 검색합니다. (Put)
put_ProviderNameWStr 사용할 CSP(암호화 서비스 공급자)의 이름을 설정하거나 검색합니다. (Put)
put_ProviderProperties 인증서 상태 응답을 제공하는 정보를 가져오거나 설정합니다. (Put)
put_ProviderType 프라이빗 키와 연결된 암호화 공급자의 유형을 지정하거나 검색합니다. (Put)
put_ProviderType ICEnroll4의 ProviderType 속성은 공급자 유형을 설정하거나 검색합니다. (Put)
put_ProviderType 공급자 유형을 설정하거나 검색합니다. (Put)
put_PublicKeyAlgorithm GetSignatureAlgorithm 메서드에 사용되는 공개 키 알고리즘에 대한 OID(개체 식별자)를 지정하고 검색합니다. (Put)
put_PVKFileName ICEnroll4의 PVKFileName 속성은 내보낸 키를 포함할 파일의 이름을 설정하거나 검색합니다. (Put)

put_PVKFileNameWStr 내보낸 키를 포함할 파일의 이름을 설정하거나 검색합니다. (Put)
put_ReaderName 스마트 카드 판독기의 이름을 지정하거나 검색합니다. (Put)
put_ReminderDuration 경고 이벤트가 기록된 후 서명 인증서 수명 백분율을 가져오거나 설정합니다. (Put)
put_RenewalCertificate 갱신 중인 DER(고유 인코딩 규칙) 인코딩 인증서를 포함하는 바이트 배열을 지정하거나 검색합니다. (Put)
put_RenewalCertificate 갱신 인증서의 인증서 컨텍스트를 지정합니다. (Put)
put_RequesterName 인증서를 요청하는 최종 엔터티의 SAM(보안 계정 관리자) 이름을 포함하는 문자열을 지정하거나 검색합니다. (Put)
put_RequestStoreFlags 요청 저장소에 사용되는 레지스트리 위치를 설정하거나 검색합니다. (Put)
put_RequestStoreFlags IEnroll4의 RequestStoreFlags 속성은 요청 저장소에 사용되는 레지스트리 위치를 설정하거나 검색합니다. (Put)
put_RequestStoreName 더미 인증서를 포함하는 저장소의 이름을 설정하거나 검색합니다. (Put)
put_RequestStoreNameWStr IEnroll4의 RequestStoreNameWStr 속성은 더미 인증서가 포함된 저장소의 이름을 설정하거나 검색합니다. (Put)
put_RequestStoreType RequestStoreName 속성에 지정된 저장소에 사용할 저장소 유형을 설정하거나 검색합니다. 이 저장소 형식은 CertOpenStore 함수에 직접 전달됩니다. (Put)

put_RequestStoreTypeWStr RequestStoreNameWStr 속성에 지정된 저장소에 사용할 저장소 유형을 설정하거나 검색합니다. 이 저장소 형식은 CertOpenStore 함수에 직접 전달됩니다. (Put)
put_ReuseHardwareKeyIfUnableToGenNew 새 키를 생성할 때 오류가 발생하는 경우 인증서 등록 제어 개체에서 수행한 작업을 결정하는 부울 값을 설정하거나 검색합니다. (Put)
put_ReuseHardwareKeyIfUnableToGenNew IEnroll4의 ReuseHardwareKeyIfUnableToGenNew 속성은 새 키를 생성할 때 오류가 발생하는 경우 인증서 등록 제어 개체에서 수행한 작업을 결정하는 부울 값을 설정하거나 검색합니다. (Put)
put_RoleForAccessCheck 액세스 검사를 수행하는 데 사용되는 역할을 설정하거나 검색합니다. (Put)
put_RootStoreFlags ICEnroll4의 RootStoreFlags 속성은 루트 저장소에 사용되는 레지스트리 위치를 설정하거나 검색합니다. (Put)
put_RootStoreFlags 루트 저장소에 사용되는 레지스트리 위치를 설정하거나 검색합니다. (Put)
put_RootStoreName 기본적으로 신뢰할 수 있는 자체 서명된 루트 인증서가 모두 유지되는 루트 저장소의 이름을 설정하거나 검색합니다. (Put)
put_RootStoreNameWStr IEnroll4의 RootStoreNameWStr 속성은 기본적으로 신뢰할 수 있는 자체 서명된 루트 인증서가 모두 유지되는 루트 저장소의 이름을 설정하거나 검색합니다. (Put)
put_RootStoreType RootStoreName 속성에 지정된 저장소에 사용할 저장소 유형을 설정하거나 검색합니다. (Put)
put_RootStoreTypeWStr RootStoreNameWStr 속성에 지정된 저장소에 사용할 저장소 유형을 설정하거나 검색합니다. (Put)
put_ScriptEngineTimeout

IAzClientContext::AccessCheck 메서드가 비즈니스 규칙(BizRule)이 실행을 완료할 때까지 기다린 후 취소할 시간을 밀리초 단위로 설정하거나 검색합니다. (Put)
put_SecurityDescriptor 프라이빗 키에 대한 보안 설명자를 지정하거나 검색합니다. (Put)
put_Selected 등록 프로세스 중에 항목을 사용할 수 있는지 여부를 나타내는 값을 지정하거나 검색합니다. (Put)
put_SenderNonce nonce를 포함하는 바이트 배열을 지정하거나 검색합니다. (Put)
put_SerialNumber 인증서 일련 번호를 지정하고 검색합니다. (Put)
put_ServerCapabilities 요청에 대한 기본 해시 및 암호화 알고리즘을 설정합니다.
put_SignerCertificate 인증서 서명에 사용되는 ISignerCertificate 개체를 지정하거나 검색합니다. (Put)
put_SignerCertificate 인증서 요청에 서명하는 데 사용되는 인증서를 지정하거나 검색합니다. (Put)
put_SignerCertificate 요청에 대한 서명자 인증서를 가져오거나 설정합니다. (Put)
put_SignerCertificate 서명자의 인증서를 설정합니다.
put_SigningCertificate DER(고유 인코딩 규칙)를 사용하여 인코딩된 서명 인증서를 가져오거나 설정합니다. OCSP(온라인 인증서 상태 프로토콜) 응답자 서비스는 이 인증서를 사용하여 인증서 상태 요청에 대한 응답에 서명합니다. (Put)
put_SigningCertificateTemplate 서명 인증서의 템플릿 이름을 가져오거나 설정합니다. (Put)
put_SigningFlags

플래그 값의 조합을 가져오거나 설정합니다. 이러한 값은 CA(인증 기관) 구성에 속하는 서명 인증서의 관리를 지정합니다. (Put)
put_Silent 프라이빗 키를 사용하여 인증서 요청에 서명할 때 사용자에게 알림을 받을지 여부를 나타내는 부울 값을 지정하거나 검색합니다. (Put)
put_Silent 인증서 등록 프로세스 중에 키 관련 모달 대화 상자가 표시되는지 여부를 나타내는 부울 값을 지정하거나 검색합니다. (Put)
put_Silent 인증서 등록 프로세스 중에 사용자 인터페이스가 표시되는지 여부를 나타내는 부울 값을 지정하거나 검색합니다. (Put)
put_Silent 프라이빗 키에 액세스할 때 인증서 등록 컨트롤에서 대화 상자를 표시할 수 있는지 여부를 나타내는 부울 값을 지정하거나 검색합니다. (Put)
put_Silent 요청 중에 UI를 허용할지 여부를 가져오거나 설정합니다.
put_SmimeCapabilities 컴퓨터에서 지원하는 암호화 기능을 식별하는 IX509ExtensionSmimeCapabilities 컬렉션을 만들지 여부를 인코딩 메서드에 알려주는 부울 값을 지정하거나 검색합니다. (Put)
put_SPCFileName base64로 인코딩된 PKCS(Put)를 쓸 파일의 이름을 설정하거나 검색합니다.
put_SPCFileNameWStr IEnroll4의 SPCFileNameWStr 속성은 base64로 인코딩된 PKCS(Put)를 쓸 파일의 이름을 설정하거나 검색합니다.
put_Status 등록 프로세스의 상태를 나타내는 값을 지정하거나 검색합니다. (Put)
put_Subject 인증서를 요청하는 엔터티의 X.500 고유 이름을 지정하거나 검색합니다. (Put)

put_SuppressDefaults 기본 확장과 특성이 요청에 포함되는지 여부를 나타내는 부울 값을 지정하거나 검색합니다. (Put)
put_Text 등록 프로세스의 상태와 연결된 메시지가 포함된 문자열을 지정하거나 검색합니다. (Put)
put_ThumbPrint 인증서 데이터의 해시를 설정하거나 검색합니다. (Put)
put_ThumbPrintWStr 인증서 데이터의 해시를 설정하거나 검색합니다. 지문은 보류 중인 인증서를 가리키는 데 사용됩니다. (Put)
put_TransactionId 인증서 요청 또는 응답을 추적하는 데 사용할 수 있는 트랜잭션 식별자를 지정하거나 검색합니다. (Put)
put_TransactionId 요청에 대한 트랜잭션 ID를 가져오거나 설정합니다. (Put)
put_Type 애플리케이션 그룹의 그룹 유형을 설정하거나 검색합니다. (Put)
put_UIContextMessage 서명 인증서와 연결된 사용자 인터페이스 텍스트가 포함된 문자열을 지정하거나 검색합니다. (Put)
put_UIContextMessage 사용자 인터페이스에 표시할 컨텍스트 문자열을 지정하거나 검색합니다. (Put)
put_UIContextMessage 프라이빗 키와 연결된 사용자 인터페이스 텍스트가 포함된 문자열을 지정하거나 검색합니다. (Put)
put_Url CEP(인증서 등록 정책) 서버의 URL을 지정하거나 검색합니다. (Put)
put_UseExistingKeySet 기존 키를 사용해야 하는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (Put)

put_UseExistingKeySet IEnroll4의 UseExistingKeySet 속성은 기존 키를 사용해야 하는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (Put)
put_Value OCSPProperty 개체가 나타내는 이름-값 쌍의 데이터 부분을 가져오거나 설정합니다. (Put)
put_Version 애플리케이션의 버전을 설정하거나 검색합니다. (Put)
put_WriteCertToCSP ICEnroll4의 WriteCertToCSP 속성은 인증서를 CSP(암호화 서비스 공급자)에 기록해야 하는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (Put)
put_WriteCertToCSP 인증서를 CSP(암호화 서비스 공급자)에 기록해야 하는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (Put)
put_WriteCertToUserDS 인증서가 사용자의 Active Directory 저장소에 기록되는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (Put)
put_WriteCertToUserDS IEnroll4의 WriteCertToUserDS 속성은 인증서가 사용자의 Active Directory 저장소에 기록되는지 여부를 결정하는 부울 값을 설정하거나 검색합니다. (Put)
PWLX_ASSIGN_SHELL_PROTECTION 새로 로그온한 사용자의 셸 프로그램에 보호를 할당하기 위해 GINA에서 호출합니다.
PWLX_CHANGE_PASSWORD_NOTIFY 암호가 변경되었음을 나타내기 위해 GINA에서 호출합니다.
PWLX_CHANGE_PASSWORD_NOTIFY_EX GINA에서 호출하여 특정 네트워크 공급자(또는 모든 네트워크 공급자)에게 암호가 변경되었음을 알립니다.
PWLX_CLOSE_USER_DESKTOP GINA에서 호출하여 대체 사용자 데스크톱을 닫고 데스크톱을 닫은 후 정리합니다.

PWLX_CREATE_USER_DESKTOP GINA에서 호출하여 사용자에 대한 대체 애플리케이션 데스크톱을 만듭니다.
PWLX_DIALOG_BOX 대화 상자 템플릿에서 모달 대화 상자를 만들기 위해 GINA에서 호출됩니다.
PWLX_DIALOG_BOX_INDIRECT GINA에서 호출하여 메모리의 대화 상자 템플릿에서 모달 대화 상자를 만듭니다.
PWLX_DIALOG_BOX_INDIRECT_PARAM GINA에서 호출하여 대화 상자 컨트롤을 초기화한 다음 메모리의 대화 상자 템플릿에서 모달 대화 상자를 만듭니다.
PWLX_DIALOG_BOX_PARAM GINA에서 호출하여 대화 상자 컨트롤을 초기화한 다음 대화 상자 템플릿 리소스에서 모달 대화 상자를 만듭니다.
PWLX_DISCONNECT 터미널 서비스를 사용하는 경우 대체 GINA DLL에 의해 호출됩니다. GINA는 터미널 서비스 네트워크 세션에서 연결을 끊기 위해 이 함수를 호출합니다.
PWLX_GET_OPTION 옵션의 현재 값을 검색하기 위해 GINA에서 호출됩니다.
PWLX_GET_SOURCE_DESKTOP Winlogon이 Winlogon 데스크톱으로 전환되기 전에 현재 데스크톱의 이름과 핸들을 확인하기 위해 GINA에서 호출되었습니다.
PWLX_MESSAGE_BOX 메시지 상자를 만들고 표시하고 작동하기 위해 GINA에서 호출합니다.
PWLX_QUERY_CLIENT_CREDENTIALS 터미널 서비스를 사용하는 경우 대체 GINA DLL에 의해 호출됩니다. GINA는 이 함수를 호출하여 인터넷 커넥터 라이선스를 사용하지 않는 원격 터미널 서비스 클라이언트의 자격 증명을 검색합니다.
PWLX_QUERY_CONSOLESWITCH_CREDENTIALS GINA가 임시 세션의 Winlogon에서 대상 세션의 Winlogon으로 전송된 자격 증명을 읽기 위해 호출합니다.

PWLX_QUERY_IC_CREDENTIALS 터미널 서비스를 사용하는 경우 대체 GINA DLL에 의해 호출됩니다. GINA는 이 함수를 호출하여 터미널 서버가 인터넷 커넥터 라이선스를 사용하고 있는지 확인하고 자격 증명 정보를 검색합니다.
PWLX_QUERY_TERMINAL_SERVICES_DATA 사용자가 로그온한 후 터미널 서비스 사용자 구성 정보를 검색하기 위해 GINA에서 호출됩니다.
PWLX_QUERY_TS_LOGON_CREDENTIALS 터미널 서비스를 사용하는 경우 대체 GINA DLL에 의해 호출되어 자격 증명 정보를 검색합니다. 그런 다음 GINA DLL은 이 정보를 사용하여 로그온 상자를 자동으로 채우고 사용자를 로그인하려고 시도할 수 있습니다.
PWLX_SAS_NOTIFY GINA가 Winlogon에 SAS(보안 주의 순서) 이벤트를 알리기 위해 호출합니다.
PWLX_SET_CONTEXT_POINTER GINA에서 호출하여 Winlogon에서 GINA 함수에 대한 모든 이후 호출에 대한 첫 번째 매개 변수로 전달되는 컨텍스트 포인터를 지정합니다.
PWLX_SET_OPTION 옵션 값을 설정하기 위해 GINA에서 호출됩니다.
PWLX_SET_RETURN_DESKTOP GINA에서 호출하여 Winlogon이 현재 SAS(보안 주의 시퀀스) 이벤트 처리 함수가 완료될 때 전환할 대체 애플리케이션 데스크톱을 지정합니다.
PWLX_SET_TIMEOUT 대화 상자와 연결된 제한 시간을 변경하기 위해 GINA에서 호출됩니다. 기본 제한 시간은 2분입니다.
PWLX_SWITCH_DESKTOP_TO_USER 애플리케이션 데스크톱으로 전환하기 위해 GINA에서 호출됩니다.
PWLX_SWITCH_DESKTOP_TO_WINLOGON GINA DLL을 Winlogon 데스크톱으로 전환할 수 있습니다.
PWLX_USE_CTRL_ALT_DEL GINA가 표준 Ctrl+Alt+DEL 키 조합을 SAS(보안 주의 시퀀스)로 사용하도록 Winlogon에 지시하기 위해 GINA가 호출합니다.

PWLX_WIN31_MIGRATE 터미널 서비스를 사용하는 경우 대체 GINA DLL에 의해 호출됩니다. GINA는 이 함수를 호출하여 터미널 서비스 클라이언트의 설정을 완료합니다.
QueryChanges Active Directory에서 템플릿 또는 인증 기관 컬렉션이 변경되었는지 여부를 지정하는 값을 검색합니다.
QueryContextAttributesA 전송 애플리케이션이 보안 컨텍스트의 특정 특성에 대해 CredSSP(자격 증명 보안 지원 공급자) 보안 패키지를 쿼리할 수 있습니다. (ANSI)
QueryContextAttributesExA ANSI(QueryContextAttributesExA) 함수(sspi.h)를 사용하면 전송 애플리케이션이 보안 컨텍스트의 특정 특성에 대한 보안 패키지를 쿼리할 수 있습니다.
QueryContextAttributesExW QueryContextAttributesExW(유니코드) 함수(sspi.h)를 사용하면 전송 애플리케이션이 보안 컨텍스트의 특정 특성에 대한 보안 패키지를 쿼리할 수 있습니다.
QueryContextAttributesW 전송 애플리케이션이 보안 컨텍스트의 특정 특성에 대해 CredSSP(자격 증명 보안 지원 공급자) 보안 패키지를 쿼리할 수 있습니다. (유니코드)
QueryCredentialsAttributesA 자격 증명과 연결된 이름과 같은 자격 증명의 특성을 검색합니다. (ANSI)
QueryCredentialsAttributesExA 보안 컨텍스트의 특성을 쿼리합니다.
QueryCredentialsAttributesExW 보안 컨텍스트의 특성을 쿼리합니다.
QueryCredentialsAttributesW 자격 증명과 연결된 이름과 같은 자격 증명의 특성을 검색합니다. (유니코드)
QuerySecurityAccessMask 지정된 개체 보안 정보를 쿼리하는 데 필요한 액세스 권한을 나타내는 액세스 마스크를 만듭니다.

QuerySecurityContextToken 클라이언트 보안 컨텍스트에 대한 액세스 토큰을 가져오고 직접 사용합니다.
QuerySecurityPackageInfoA 지정된 보안 패키지에 대한 정보를 검색합니다. 이 정보에는 인증 정보, 자격 증명 및 컨텍스트의 크기에 대한 범위가 포함됩니다. (ANSI)
QuerySecurityPackageInfoW 지정된 보안 패키지에 대한 정보를 검색합니다. 이 정보에는 인증 정보, 자격 증명 및 컨텍스트의 크기에 대한 범위가 포함됩니다. (유니코드)
QueryServiceConfig2A 지정된 서비스의 선택적 구성 매개 변수를 검색합니다. (ANSI)
QueryServiceConfig2W 지정된 서비스의 선택적 구성 매개 변수를 검색합니다. (유니코드)
QueryServiceConfigA 지정된 서비스의 구성 매개 변수를 검색합니다. (ANSI)
QueryServiceConfigW 지정된 서비스의 구성 매개 변수를 검색합니다. (유니코드)
QueryServiceDynamicInformation 현재 서비스 시작과 관련된 동적 정보를 검색합니다.
QueryServiceLockStatusA 지정된 서비스 제어 관리자 데이터베이스의 잠금 상태를 검색합니다. (ANSI)
QueryServiceLockStatusW 지정된 서비스 제어 관리자 데이터베이스의 잠금 상태를 검색합니다. (유니코드)
QueryServiceObjectSecurity 서비스 개체와 연결된 보안 설명자의 복사본을 검색합니다.
QueryServiceStatus 지정된 서비스의 현재 상태를 검색합니다.

QueryServiceStatusEx 지정된 정보 수준에 따라 지정된 서비스의 현재 상태를 검색합니다.
RegGetKeySecurity 지정된 열린 레지스트리 키를 보호하는 보안 설명자의 복사본을 검색합니다.
RegisterServiceCtrlHandlerA 서비스 제어 요청을 처리하는 함수를 등록합니다. (ANSI)
RegisterServiceCtrlHandlerExA 확장 서비스 제어 요청을 처리하는 함수를 등록합니다. (ANSI)
RegisterServiceCtrlHandlerExW 확장 서비스 제어 요청을 처리하는 함수를 등록합니다. (유니코드)
RegisterServiceCtrlHandlerW 서비스 제어 요청을 처리하는 함수를 등록합니다. (유니코드)
RegSetKeySecurity 열린 레지스트리 키의 보안을 설정합니다.
제거 인터페이스 목록에서 지정된 인터페이스를 제거합니다. BizRule 스크립트에서 호출할 수 있는 인터페이스 목록의 인터페이스 수입니다.
제거 비즈니스 규칙(BizRule) 스크립트에 사용할 수 있는 매개 변수 목록에서 지정된 매개 변수를 제거합니다.
제거 인덱스 번호별로 컬렉션에서 개체를 제거합니다. (IAlternativeNames.Remove)
제거 인덱스 번호별로 컬렉션에서 개체를 제거합니다. (ICertificatePolicies.Remove)
제거 인덱스 번호별로 컬렉션에서 ICertificationAuthority 개체를 제거합니다.

제거 인덱스 값으로 컬렉션에서 속성을 제거합니다.
제거 인덱스 번호별로 컬렉션에서 ICryptAttribute 개체를 제거합니다.
제거 인덱스 번호별로 컬렉션에서 ICspAlgorithm 개체를 제거합니다.
제거 인덱스 번호별로 컬렉션에서 ICspInformation 개체를 제거합니다.
제거 인덱스 번호로 컬렉션에서 ICspStatus 개체를 제거합니다.
제거 인덱스 값으로 컬렉션에서 IObjectId 개체를 제거합니다.
제거 인덱스 값으로 컬렉션에서 개체를 제거합니다. (IPolicyQualifiers.Remove)
제거 인덱스 번호별로 컬렉션에서 ISignerCertificate 개체를 제거합니다.
제거 인덱스 값으로 컬렉션에서 개체를 제거합니다. (ISmimeCapabilities.Remove)
제거 인덱스 번호별로 컬렉션에서 IX509Attribute 개체를 제거합니다.
제거 인덱스 번호별로 컬렉션에서 IX509CertificateTemplate 개체를 제거합니다.
제거 인덱스 번호로 컬렉션에서 IX509Extension 개체를 제거합니다.
제거

인덱스 번호로 컬렉션에서 IX509NameValuePair 개체를 제거합니다.
제거 인덱스 번호별로 컬렉션에서 IX509PolicyServerUrl 개체를 제거합니다.
removeAll 비즈니스 규칙(BizRule) 스크립트에서 호출할 수 있는 인터페이스 목록에서 모든 인터페이스를 제거합니다.
removeAll 비즈니스 규칙(BizRule) 스크립트에 사용할 수 있는 매개 변수 목록에서 모든 매개 변수를 제거합니다.
RemoveCertificate 키 스토리지 공급자에서 인증 키와 관련된 인증 인증서를 제거합니다. Open 메서드가 성공적으로 호출된 후에만 RemoveCertificate 메서드를 호출할 수 있습니다.
RemoveFromCertificate 인증서에서 속성을 연결 해제합니다.
RemoveFromRegistry CEP(인증서 등록 정책) 서버의 등록을 취소합니다.
removePendingRequest 클라이언트의 요청 저장소에서 보류 중인 요청을 제거합니다. 이 메서드는 ICEnroll4 인터페이스에서 처음 정의되었습니다.
removePendingRequestWStr 클라이언트의 요청 저장소에서 보류 중인 요청을 제거합니다.
ReportError 요청된 작업의 오류를 보고합니다.
ReportProgress 현재 작업의 진행률을 보고합니다.
재설정 ID 열거형의 현재 인덱스(0)를 설정합니다.

재설정

지정된 구성 지점에서 인덱싱된 Certificate Services 서버 구성을 가리키도록 구성 쿼리 상태를 다시 설정합니다. 이 메서드는 ICertConfig 인터페이스에서 처음 정의되었습니다.

재설정

인증서 등록 컨트롤 개체를 초기 상태로 반환하여 컨트롤을 다시 사용할 수 있습니다. 이 메서드는 ICEnroll3 인터페이스에서 처음 정의되었습니다.

재설정

인증서 등록 컨트롤 개체를 초기 상태로 반환하여 컨트롤을 다시 사용할 수 있습니다.

재설정

이 개체의 대체 이름 배열 크기를 지정합니다. 배열에 있는 모든 요소의 값은 0으로 설정됩니다.

재설정

CRL(인증서 해지 목록) 배포 정보 배열을 지정된 수의 배포 지점 구조로 다시 설정합니다.

재설정

이 개체의 DATE 배열 크기를 지정합니다.

재설정

이 개체의 배열 크기를 지정합니다.

재설정

문자열 배열의 크기와 배열에 포함될 문자열의 형식을 지정합니다.

재설정

특성 열거형 시퀀스의 시작 부분으로 이동합니다.

재설정

열 열거형 시퀀스의 시작 부분으로 이동합니다.

재설정

확장 열거형 시퀀스의 시작 부분으로 이동합니다.

재설정

행 열거형 시퀀스의 시작 부분으로 이동합니다.

resetAttributes 요청에서 모든 특성을 제거합니다. 이 메서드는 ICEnroll4 인터페이스에서 처음 정의되었습니다.
resetAttributes 요청에서 모든 특성을 제거합니다.
resetBlobProperties BLOB의 속성을 다시 설정합니다.
resetExtensions 요청에서 모든 확장을 제거합니다. 이 메서드는 ICEnroll4 인터페이스에서 처음 정의되었습니다.
resetExtensions 요청에서 모든 확장을 제거합니다.
ResetForEncode 요청 개체의 상태를 Encode 메서드가 호출되기 전에 존재했던 상태로 복원합니다.
ResubmitRequest 지정된 인증 기관의 정책 모듈에 지정된 인증서 요청을 제출합니다. 이 메서드는 ICertAdmin 인터페이스에서 처음 도입되었습니다.
RetrievePending 이전에 CR_DISP_INCOMPLETE 또는 CR_DISP_UNDER_SUBMISSION 반환했을 수 있는 이전 요청에서 인증서의 처리 상태를 검색합니다.
RevertSecurityContext 보안 패키지가 호출자의 가장을 중단하고 자체 보안 컨텍스트를 복원할 수 있도록 허용합니다.
RevertToSelf 클라이언트 애플리케이션의 가장을 종료합니다.
RevokeCertificate 지정된 날짜 또는 즉시 인증서를 해지합니다. 이 메서드는 ICertAdmin 인터페이스에서 처음 정의되었습니다.
RoleAssignments

이 애플리케이션 그룹과 연결된 IAzRoleAssignment 개체의 컬렉션을 가져옵니다.
RoleAssignments 이 작업과 연결된 역할 할당의 컬렉션을 반환합니다.
RoleAssignments 이 IAzRoleDefinition 개체와 연결된 역할 할당을 나타내는 IAzRoleAssignment 개체의 컬렉션을 검색합니다.
RoleAssignments 이 작업과 연결된 역할 할당의 컬렉션을 반환합니다.
RtlConvertSidToUnicodeString SID(보안 식별자)를 유니코드 문자 표현으로 변환합니다.
RtlDecryptMemory RtlEncryptMemory 함수에 의해 이전에 암호화된 메모리 콘텐츠를 해독합니다.
RtlEncryptMemory 메모리 콘텐츠를 암호화합니다.
RtlGenRandom 의사 난수를 생성합니다.
SaferCloseLevel SaferIdentifyLevel 함수 또는 SaferCreateLevel 함수를 사용하여 연 SAFER_LEVEL_HANDLE 닫습니다.
SaferComputeTokenFromLevel SAFER_LEVEL_HANDLE 지정된 제한을 사용하여 토큰을 제한합니다.
SaferCreateLevel SAFER_LEVEL_HANDLE 엽니다.
SaferGetLevelInformation 정책 수준에 대한 정보를 검색합니다.
SaferGetPolicyInformation 정책에 대한 정보를 가져옵니다.

SaferIdentifyLevel 수준에 대한 정보를 검색합니다.
SaferIsExecutableFileType 지정된 파일이 실행 파일인지 여부를 확인합니다.
SaferRecordEventLogEntry 이벤트 로그에 메시지를 저장합니다.
SaferSetLevelInformation 정책 수준에 대한 정보를 설정합니다.
SaferSetPolicyInformation 전역 정책 컨트롤을 설정합니다.
SaslAcceptSecurityContext 보안 지원 공급자 인터페이스 AcceptSecurityContext(일반) 함수에 대한 표준 호출을 래핑하고 SASL 서버 쿠키 만들기를 포함합니다.
SaslEnumerateProfilesA SASL 인터페이스를 제공하는 패키지를 나열합니다. (ANSI)
SaslEnumerateProfilesW SASL 인터페이스를 제공하는 패키지를 나열합니다. (유니코드)
SaslGetContextOption 지정된 SASL 컨텍스트의 지정된 속성을 검색합니다.
SaslGetProfilePackageA 지정된 패키지에 대한 패키지 정보를 반환합니다. (ANSI)
SaslGetProfilePackageW 지정된 패키지에 대한 패키지 정보를 반환합니다. (유니코드)
SaslIdentifyPackageA 지정된 SASL 협상 버퍼와 일치하는 협상 접두사를 반환합니다. (ANSI)

SaslIdentifyPackageW 지정된 SASL 협상 버퍼와 일치하는 협상 접두사를 반환합니다. (유니코드)
SaslInitializeSecurityContextA 보안 지원 공급자 인터페이스 InitializeSecurityContext(일반) 함수에 대한 표준 호출을 래핑하고 서버에서 SASL 서버 쿠키를 처리합니다. (ANSI)
SaslInitializeSecurityContextW 보안 지원 공급자 인터페이스 InitializeSecurityContext(일반) 함수에 대한 표준 호출을 래핑하고 서버에서 SASL 서버 쿠키를 처리합니다. (유니코드)
SaslSetContextOption 지정된 SASL 컨텍스트에 대해 지정된 속성의 값을 설정합니다.
저장 Save 메서드를 사용하면 스냅인 확장 프로그램이 저장해야 하는 데이터에 대한 정보를 반환합니다. 호출자는 데이터를 저장할 책임이 있습니다.
SCardAccessStartedEvent 이벤트가 스마트 카드 리소스 관리자가 시작되었음을 알리는 신호가 표시되면 이벤트 핸들을 반환합니다.
SCardAddReaderToGroupA 판독기 그룹에 판독기를 추가합니다. (ANSI)
SCardAddReaderToGroupW 판독기 그룹에 판독기를 추가합니다. (유니코드)
SCardAudit Windows 애플리케이션 로그 Microsoft-Windows-SmartCard-Audit/Authentication에 이벤트 메시지를 씁니다.
SCardBeginTransaction 트랜잭션을 시작합니다.
SCardCancel 특정 리소스 관리자 컨텍스트 내에서 처리 중인 모든 작업을 종료합니다.

SCardConnectA 호출 애플리케이션과 특정 판독기에서 포함된 스마트 카드 간에 연결(특정 리소스 관리자 컨텍스트 사용)을 설정합니다. 지정된 판독기에서 카드가 없으면 오류가 반환됩니다. (ANSI)
SCardConnectW 호출 애플리케이션과 특정 판독기에서 포함된 스마트 카드 간에 연결(특정 리소스 관리자 컨텍스트 사용)을 설정합니다. 지정된 판독기에서 카드가 없으면 오류가 반환됩니다. (유니코드)
SCardControl 판독기를 직접 제어할 수 있습니다. SCardConnect를 성공적으로 호출한 후 SCardDisconnect를 성공적으로 호출하기 전에 언제든지 호출할 수 있습니다.
SCardDisconnect 대상 판독기에서 호출 애플리케이션과 스마트 카드 간에 이전에 열린 연결을 종료합니다.
SCardEndTransaction 이전에 선언된 트랜잭션을 완료하여 다른 애플리케이션이 카드와의 상호 작용을 다시 시작할 수 있도록 합니다.
SCardEstablishContext 데이터베이스 작업이 수행되는 리소스 관리자 컨텍스트(범위)를 설정합니다.
SCardForgetCardTypeA 스마트 카드 하위 시스템의 도입된 스마트 카드를 제거합니다. (ANSI)
SCardForgetCardTypeW 스마트 카드 하위 시스템의 도입된 스마트 카드를 제거합니다. (유니코드)
SCardForgetReaderA 이전에 도입된 판독기를 스마트 카드 하위 시스템에 의해 제어에서 제거합니다. 추가되었을 수 있는 판독기 그룹을 포함하여 스마트 카드 데이터베이스에서 제거됩니다. (ANSI)
SCardForgetReaderGroupA 이전에 도입된 스마트 카드 판독기 그룹을 스마트 카드 하위 시스템으로부터 제거합니다. 이 함수는 그룹의 모든 판독기를 자동으로 지우지만 데이터베이스에 개별 판독기의 존재 여부에는 영향을 주지 않습니다. (ANSI)
SCardForgetReaderGroupW

이전에 도입된 스마트 카드 판독기 그룹을 스마트 카드 하위 시스템으로부터 제거합니다. 이 함수는 그룹의 모든 판독기를 자동으로 지우지만 데이터베이스에 개별 판독기의 존재 여부에는 영향을 주지 않습니다. (유니코드)
SCardForgetReaderW 이전에 도입된 판독기를 스마트 카드 하위 시스템에 의해 제어에서 제거합니다. 추가되었을 수 있는 판독기 그룹을 포함하여 스마트 카드 데이터베이스에서 제거됩니다. (유니코드)
SCardFreeMemory SCARD_AUTOALLOCATE 길이 지정자를 사용하여 리소스 관리자에서 반환된 메모리를 해제합니다.
SCardGetAttrib 지정된 핸들에 대한 현재 판독기 특성을 검색합니다. 판독기, 드라이버 또는 카드의 상태에는 영향을 주지 않습니다.
SCardGetCardTypeProviderNameA 지정된 카드 이름 및 공급자 유형에 대한 공급자를 포함하는 모듈(동적 링크 라이브러리)의 이름을 반환합니다. (ANSI)
SCardGetCardTypeProviderNameW 지정된 카드 이름 및 공급자 유형에 대한 공급자를 포함하는 모듈(동적 링크 라이브러리)의 이름을 반환합니다. (유니코드)
SCardGetDeviceTypeIdA 지정된 판독기 이름에 대한 카드 판독기의 디바이스 유형 식별자를 가져옵니다. 이 함수는 판독기의 상태에 영향을 주지 않습니다. (ANSI)
SCardGetDeviceTypeIdW 지정된 판독기 이름에 대한 카드 판독기의 디바이스 유형 식별자를 가져옵니다. 이 함수는 판독기의 상태에 영향을 주지 않습니다. (유니코드)
SCardGetProviderIdA 지정된 카드에 대한 기본 서비스 공급자의 식별자(GUID)를 반환합니다. (ANSI)
SCardGetProviderIdW 지정된 카드에 대한 기본 서비스 공급자의 식별자(GUID)를 반환합니다. (유니코드)
SCardGetReaderDeviceInstanceIdA

지정된 판독기 이름에 대한 카드 판독기의 디바이스 인스턴스 식별자를 가져옵니다. 이 함수는 판독기의 상태에 영향을 주지 않습니다. (ANSI)
SCardGetReaderDeviceInstanceIdW 지정된 판독기 이름에 대한 카드 판독기의 디바이스 인스턴스 식별자를 가져옵니다. 이 함수는 판독기의 상태에 영향을 주지 않습니다. (유니코드)
SCardGetReaderIconA 지정된 판독기 이름에 대한 스마트 카드 판독기의 아이콘을 가져옵니다. (ANSI)
SCardGetReaderIconW 지정된 판독기 이름에 대한 스마트 카드 판독기의 아이콘을 가져옵니다. (유니코드)
SCardGetStatusChangeA 특정 판독기 집합에서 카드의 현재 가용성이 변경될 때까지 실행을 차단합니다. (ANSI)
SCardGetStatusChangeW 특정 판독기 집합에서 카드의 현재 가용성이 변경될 때까지 실행을 차단합니다. (유니코드)
SCardGetTransmitCount 지정된 카드 판독기를 삽입한 이후 완료된 전송 작업 수를 검색합니다.
SCardIntroduceCardTypeA 스마트 카드 데이터베이스에 스마트 카드를 추가하여 스마트 카드 하위 시스템(활성 사용자용)을 소개합니다. (ANSI)
SCardIntroduceCardTypeW 스마트 카드 데이터베이스에 스마트 카드를 추가하여 스마트 카드 하위 시스템(활성 사용자용)을 소개합니다. (유니코드)
SCardIntroduceReaderA 기존 스마트 카드 판독기 이름을 소개합니다. (ANSI)
SCardIntroduceReaderGroupA 스마트 카드 하위 시스템에 판독기 그룹을 소개합니다. 그러나 판독기 그룹은 스마트 카드 데이터베이스에 판독기를 추가할 때 그룹이 지정될 때까지 만들어지지 않습니다. (ANSI)
SCardIntroduceReaderGroupW

스마트 카드 하위 시스템에 판독기 그룹을 소개합니다. 그러나 판독기 그룹은 스마트 카드 데이터베이스에 판독기를 추가할 때 그룹이 지정될 때까지 만들어지지 않습니다. (유니코드)
SCardIntroduceReaderW 기존 스마트 카드 판독기 이름을 소개합니다. (유니코드)
SCardIsValidContext 스마트 카드 컨텍스트 핸들이 유효한지 여부를 확인합니다.
SCardListCardsA 스마트 카드 데이터베이스를 검색하고 사용자가 이전에 시스템에 도입한 명명된 카드 목록을 제공합니다. (ANSI)
SCardListCardsW 스마트 카드 데이터베이스를 검색하고 사용자가 이전에 시스템에 도입한 명명된 카드 목록을 제공합니다. (유니코드)
SCardListInterfacesA 지정된 카드에서 제공하는 인터페이스 목록을 제공합니다. (ANSI)
SCardListInterfacesW 지정된 카드에서 제공하는 인터페이스 목록을 제공합니다. (유니코드)
SCardListReaderGroupsA 이전에 시스템에 도입된 판독기 그룹 목록을 제공합니다. (ANSI)
SCardListReaderGroupsW 이전에 시스템에 도입된 판독기 그룹 목록을 제공합니다. (유니코드)
SCardListReadersA 명명된 판독기 그룹 집합 내에 판독기 목록을 제공하여 중복 항목을 제거합니다. (ANSI)
SCardListReadersW 명명된 판독기 그룹 집합 내에 판독기 목록을 제공하여 중복 항목을 제거합니다. (유니코드)
SCardListReadersWithDeviceInstanceIdA 디바이스 인스턴스 식별자를 제공한 판독기 목록을 가져옵니다. 이 함수는 판독기의 상태에 영향을 주지 않습니다. (ANSI)

SCardListReadersWithDeviceInstanceIdW 디바이스 인스턴스 식별자를 제공한 판독기 목록을 가져옵니다. 이 함수는 판독기의 상태에 영향을 주지 않습니다. (유니코드)
SCardLocateCardsA rgReaderStates 매개 변수에 나열된 판독기에서 mszCards에 지정된 카드 이름 중 하나와 일치하는 ATR 문자열이 있는 카드를 검색하여 결과를 즉시 반환합니다. (ANSI)
SCardLocateCardsByATRA rgReaderStates 매개 변수에 나열된 판독기에서 rgAtrMasks 매개 변수로 지정된 SCARD_ATRMASK 구조 중 하나에 포함된 카드 이름 중 하나와 일치하는 카드를 검색합니다. (ANSI)
SCardLocateCardsByATRW rgReaderStates 매개 변수에 나열된 판독기에서 rgAtrMasks 매개 변수로 지정된 SCARD_ATRMASK 구조 중 하나에 포함된 카드 이름 중 하나와 일치하는 카드를 검색합니다. (유니코드)
SCardLocateCardsW rgReaderStates 매개 변수에 나열된 판독기에서 mszCards에 지정된 카드 이름 중 하나와 일치하는 ATR 문자열이 있는 카드를 검색하여 결과를 즉시 반환합니다. (유니코드)
SCardReadCacheA 스마트 카드 리소스 관리자가 유지 관리하는 전역 캐시에서 이름-값 쌍의 값 부분을 검색합니다. (ANSI)
SCardReadCacheW 스마트 카드 리소스 관리자가 유지 관리하는 전역 캐시에서 이름-값 쌍의 값 부분을 검색합니다. (유니코드)
SCardReconnect 호출 애플리케이션과 스마트 카드 간의 기존 연결을 다시 설정합니다.
SCardReleaseContext 설정된 리소스 관리자 컨텍스트를 닫고 SCARDHANDLE 개체 및 SCARD_AUTOALLOCATE 길이 지정자를 사용하여 할당된 메모리를 포함하여 해당 컨텍스트에 할당된 모든 리소스를 해제합니다.
SCardReleaseStartedEvent SCardAccessStartedEvent 함수에 대한 이전 호출에서 얻은 핸들의 참조 수를 줄입니다.
SCardRemoveReaderFromGroupA

기존 판독기 그룹에서 판독기를 제거합니다. 이 함수는 판독기에서 영향을 주지 않습니다. (ANSI)
SCardRemoveReaderFromGroupW 기존 판독기 그룹에서 판독기를 제거합니다. 이 함수는 판독기에서 영향을 주지 않습니다. (유니코드)
SCardSetAttrib 지정된 핸들에 대해 지정된 판독기 특성을 설정합니다.
SCardSetCardTypeProviderNameA 지정된 카드 이름 및 공급자 유형에 대한 공급자를 포함하는 모듈(동적 링크 라이브러리)의 이름을 지정합니다. (ANSI)
SCardSetCardTypeProviderNameW 지정된 카드 이름 및 공급자 유형에 대한 공급자를 포함하는 모듈(동적 링크 라이브러리)의 이름을 지정합니다. (유니코드)
SCardStatusA 판독기에서 스마트 카드의 현재 상태를 제공합니다. (ANSI)
SCardStatusW 판독기에서 스마트 카드의 현재 상태를 제공합니다. (유니코드)
SCardTransmit 스마트 카드에 서비스 요청을 보내고 카드에서 데이터를 다시 받을 것으로 예상합니다.
SCardUIDlgSelectCardA 스마트 카드 카드 선택 대화 상자를 표시합니다. (ANSI)
SCardUIDlgSelectCardW 스마트 카드 카드 선택 대화 상자를 표시합니다. (유니코드)
SCardWriteCacheA 스마트 카드에서 스마트 카드 리소스 관리자가 유지 관리하는 전역 캐시에 이름-값 쌍을 씁니다. (ANSI)
SCardWriteCacheW 스마트 카드에서 스마트 카드 리소스 관리자가 유지 관리하는 전역 캐시에 이름-값 쌍을 씁니다. (유니코드)

ScopeExists 지정된 범위가 이 IAzApplication3 개체에 있는지 여부를 나타냅니다.
SendSAS SAS(보안 주의 시퀀스)를 시뮬레이트합니다.
SetAccountInformation IIS NDES(네트워크 디바이스 등록 서비스) 확장에서 네트워크 디바이스를 대신하여 등록을 수행하는 데 사용하는 사용자 계정 정보를 설정합니다.
SetAclInformation ACL(액세스 제어 목록)에 대한 정보를 설정합니다.
SetApplicationPoolCredentials CES(인증서 등록 웹 서비스)가 실행되는 애플리케이션 풀에 대한 사용자 계정 정보를 지정합니다.
SetCachedSigningLevel 캐시된 서명 수준을 설정합니다.
SetCADistinguishedName CA(인증 기관) 일반 이름과 선택적 고유 이름 접미사를 설정합니다.
SetCAProperty CA(인증 기관)의 속성 값을 설정합니다.
SetCASetupProperty CA(인증 기관) 구성에 대한 속성 값을 설정합니다.
SetCertificateExtension 인증서 요청에 대한 응답으로 발급된 인증서에 새 확장을 추가합니다. 이 메서드는 ICertAdmin 인터페이스에 의해 처음 정의되었습니다.
SetCertificateExtension 인증서에 새 확장을 추가합니다.
SetCertificateProperty 인증서와 연결된 속성을 설정하려면

SetConfigEntry CA(인증 기관)에 대한 구성 정보를 설정합니다.
SetConfiguration 구성 변경 내용으로 응답자 서비스를 업데이트합니다.
SetContext 컨텍스트에서 참조하는 요청에 대해 인터페이스의 현재 인스턴스화가 작동하도록 합니다.
SetContext 인증서 서비스에 대한 후속 호출의 컨텍스트로 사용할 요청을 지정합니다.
SetContextAttributesA 전송 애플리케이션이 보안 패키지에 대한 보안 컨텍스트의 특성을 설정할 수 있도록 합니다. 이 함수는 Schannel 보안 패키지에서만 지원됩니다. (ANSI)
SetContextAttributesW 전송 애플리케이션이 보안 패키지에 대한 보안 컨텍스트의 특성을 설정할 수 있도록 합니다. 이 함수는 Schannel 보안 패키지에서만 지원됩니다. (유니코드)
SetCredential CEP(인증서 등록 정책) 서버에 문의하는 데 사용되는 자격 증명을 설정합니다.
SetCredential 인증서 등록 웹 서비스에 문의하는 데 사용되는 자격 증명을 설정합니다.
SetCredentialsAttributesA 자격 증명과 연결된 이름과 같은 자격 증명의 특성을 설정합니다. (ANSI)
SetCredentialsAttributesW 자격 증명과 연결된 이름과 같은 자격 증명의 특성을 설정합니다. (유니코드)
SetDatabaseInformation CA(인증 기관) 역할에 대한 데이터베이스 관련 정보를 설정합니다.
SetDefaultValues 서명하기 전에 인증서 요청의 다이제스트를 만드는 데 사용되는 기본 해시 알고리즘을 지정합니다.

SetEntriesInAclA 새 액세스 제어 또는 감사 제어 정보를 기존 ACL 구조로 병합하여 새 ACL(액세스 제어 목록)을 만듭니다. (ANSI)
SetEntriesInAclW 새 액세스 제어 또는 감사 제어 정보를 기존 ACL 구조로 병합하여 새 ACL(액세스 제어 목록)을 만듭니다. (유니코드)
SetFileSecurityA ANSI(SetFileSecurityA) 함수(winbase.h)는 파일 또는 디렉터리 개체의 보안을 설정합니다.
SetFileSecurityW SetFileSecurityW(유니코드) 함수(securitybaseapi.h)는 파일 또는 디렉터리 개체의 보안을 설정합니다.
SetHStoreCA SetHStoreCA 메서드는 CA 저장소에 사용할 핸들을 지정합니다. 이 메서드는 IEnroll2 인터페이스에서 처음 정의되었습니다.
SetHStoreMy SetHStoreMy 메서드는 MY 저장소에 사용할 핸들을 지정합니다. 이 메서드는 IEnroll2 인터페이스에서 처음 정의되었습니다.
SetHStoreRequest SetHStoreRequest 메서드는 요청 저장소에 사용할 핸들을 지정합니다. 이 메서드는 IEnroll2 인터페이스에서 처음 정의되었습니다.
SetHStoreROOT SetHStoreROOT 메서드는 루트 저장소에 사용할 핸들을 지정합니다. 이 메서드는 IEnroll2 인터페이스에서 처음 정의되었습니다.
SetKernelObjectSecurity 커널 개체의 보안을 설정합니다.
SetMSCEPSetupProperty NDES(네트워크 디바이스 등록 서비스) 구성에 대한 속성 값을 설정합니다.
SetNameCount CRL(인증서 해지 목록) 배포 정보 배열에서 지정된 배포 지점의 이름 수를 설정합니다.

SetNamedSecurityInfoA 지정된 개체의 보안 설명자에서 지정된 보안 정보를 설정합니다. (ANSI)
SetNamedSecurityInfoW 지정된 개체의 보안 설명자에서 지정된 보안 정보를 설정합니다. (유니코드)
SetNameEntry 대체 이름 배열의 지정된 인덱스에서 이름을 설정합니다.
SetNameEntry CRL(인증서 해지 목록) 배포 정보 배열에 있는 배포 지점의 지정된 인덱스에서 이름을 설정합니다.
SetParentCAInformation 하위 CA 구성에 대한 부모 CA(인증 기관) 정보를 설정합니다.
setPendingRequestInfo 보류 중인 요청에 대한 속성을 설정합니다. 이 메서드는 ICEnroll4 인터페이스에서 처음 정의되었습니다.
setPendingRequestInfoWStr 보류 중인 요청에 대한 속성을 설정합니다.
SetPrivateKeyArchiveCertificate SetPrivateKeyArchiveCertificate 메서드는 프라이빗 키를 보관하는 데 사용되는 인증서를 지정합니다. 이 메서드는 IEnroll4 인터페이스에서 처음 정의되었습니다.
SetPrivateObjectSecurity 개인 개체의 보안 설명자를 수정합니다.
SetPrivateObjectSecurityEx 이 함수를 호출하는 리소스 관리자가 유지 관리하는 프라이빗 개체의 보안 설명자를 수정합니다.
SetProperty 지정된 속성 ID를 사용하여 지정된 값을 IAzApplication 개체 속성으로 설정합니다.
SetProperty 지정된 속성 ID를 사용하여 지정된 값을 IAzApplicationGroup 개체 속성으로 설정합니다.

SetProperty 지정된 속성 ID를 사용하여 지정된 값을 AzAuthorizationStore 개체 속성으로 설정합니다.
SetProperty 지정된 속성 ID를 사용하여 지정된 값을 IAzOperation 개체 속성으로 설정합니다.
SetProperty 지정된 속성 ID를 사용하여 지정된 값을 IAzRole 개체 속성으로 설정합니다.
SetProperty 지정된 속성 ID를 사용하여 지정된 값을 IAzScope 개체 속성으로 설정합니다.
SetProperty 지정된 속성 ID를 사용하여 지정된 값을 IAzTask 개체 속성으로 설정합니다.
SetProperty CEP(인증서 등록 정책) 웹 서비스 구성에 대한 CEPSetupProperty 열거형 값을 지정합니다.
SetProperty CES(인증서 등록 웹 서비스) 구성에 대한 CESSetupProperty 열거형 값을 지정합니다.
SetProperty 모듈에서 속성 값을 설정할 수 있습니다.
SetRequestAttributes 지정된 보류 중인 인증서 요청의 특성을 설정합니다. 이 메서드는 ICertAdmin 인터페이스에서 처음 정의되었습니다.
SetRestriction 열에 대한 정렬 및 한정 제한을 설정합니다.
SetResultColumn 인증서 서비스 데이터베이스의 사용자 지정된 뷰 결과 집합에 대한 열을 지정합니다.
SetResultColumnCount 인증서 서비스 데이터베이스의 사용자 지정된 뷰 결과 집합에 대한 최대 열 수를 지정합니다.

SetSecurity SetSecurity 메서드는 사용자가 보안 개체에 적용하려는 보안 정보를 포함하는 보안 설명자를 제공합니다. 사용자가 확인 또는 적용을 클릭할 때 액세스 제어 편집기에서 이 메서드를 호출합니다.
SetSecurity OCSP(온라인 인증서 상태 프로토콜) 응답자 서버에 대한 보안 설명자 정보를 업데이트합니다.
SetSecurityAccessMask 지정된 개체 보안 정보를 설정하는 데 필요한 액세스 권한을 나타내는 액세스 마스크를 만듭니다.
SetSecurityDescriptorControl 보안 설명자의 컨트롤 비트를 설정합니다. 함수는 ACE의 자동 상속과 관련된 컨트롤 비트만 설정할 수 있습니다.
SetSecurityDescriptorDacl DACL(임의 액세스 제어 목록)의 정보를 설정합니다. DACL이 보안 설명자에 이미 있는 경우 DACL이 대체됩니다.
SetSecurityDescriptorGroup 보안 설명자에 이미 있는 기본 그룹 정보를 대체하여 절대 형식 보안 설명자의 기본 그룹 정보를 설정합니다.
SetSecurityDescriptorOwner 절대 형식 보안 설명자의 소유자 정보를 설정합니다. 보안 설명자에 이미 있는 소유자 정보를 대체합니다.
SetSecurityDescriptorRMControl SECURITY_DESCRIPTOR 구조에서 Resource Manager 컨트롤 비트를 설정합니다.
SetSecurityDescriptorSacl SACL(시스템 액세스 제어 목록)의 정보를 설정합니다. 보안 설명자에 SACL이 이미 있는 경우 대체됩니다.
SetSecurityInfo 지정된 개체의 보안 설명자에서 지정된 보안 정보를 설정합니다. 호출자는 핸들로 개체를 식별합니다.
SetServiceObjectSecurity 서비스 개체의 보안 설명자를 설정합니다.

SetServiceStatus 호출 서비스에 대한 서비스 제어 관리자의 상태 정보를 업데이트합니다.
SetSharedFolder CA(인증 기관) 공유 폴더로 사용할 경로를 지정합니다.
SetSignerCertificate SetSignerCertificate 메서드는 서명자의 인증서를 지정합니다. 이 메서드는 IEnroll4 인터페이스에서 처음 정의되었습니다.
SetStringProperty CEP(인증서 등록 정책) 서버 ID 또는 CEP 서버의 표시 이름을 지정합니다.
설정할 ICertView2 인터페이스의 메서드에 대한 후속 호출에 사용되는 Certificate Services 데이터베이스 테이블을 지정합니다.
SetThreadToken 스레드에 가장 토큰을 할당합니다. 이 함수는 스레드가 가장 토큰 사용을 중지하도록 할 수도 있습니다.
SetTokenInformation 지정된 액세스 토큰에 대한 다양한 유형의 정보를 설정합니다.
SetUserObjectSecurity 사용자 개체의 보안을 설정합니다. 예를 들어 창 또는 DDE 대화일 수 있습니다.
SetValue DATE 배열의 지정된 인덱스에서 DATE 값을 설정합니다.
SetValue Long 배열의 지정된 인덱스에서 Long 값을 설정합니다.
SetValue 문자열 배열의 지정된 인덱스에서 문자열 값을 설정합니다.
SetValueOnCertificate 속성 값을 기존 인증서와 연결합니다.

SetWebCAInformation 인증 기관 웹 등록 역할에 대한 CA(인증 기관) 정보를 설정합니다.
종료 서버가 종료되기 전에 서버 엔진에서 호출됩니다.
건너뛰기 특성 열거형 시퀀스에서 지정된 수의 특성을 건너뜁니다.
건너뛰기 열 열거형 시퀀스에서 지정된 수의 열을 건너뜁니다.
건너뛰기 확장 열거형 시퀀스에서 지정된 수의 확장을 건너뜁니다.
건너뛰기 행 열거형 시퀀스에서 지정된 수의 행을 건너뜁니다.
SLAcquireGenuineTicket SLS(소프트웨어 라이선싱 서버)에서 획득한 XrML 정품 티켓을 가져옵니다.
SLActivateProduct SLS(소프트웨어 라이선스 서버)에서 사용 라이선스를 획득합니다.
SLClose SLC(소프트웨어 라이선스 클라이언트) 컨텍스트 핸들을 닫습니다.
SLConsumeRight 애플리케이션에서 로컬로 저장된 라이선스에 대한 권한을 행사할 수 있도록 합니다.
SLDepositMigrationBlob SLGatherMigrationBlob 함수를 사용하여 이전에 수집하고 수집한 라이선스 정보를 입금합니다.
SLDepositOfflineConfirmationId 오프라인 정품 인증을 위해 IID(설치 ID) 및 CID(확인 ID)를 입금합니다. (SLDepositOfflineConfirmationId)
SLDepositOfflineConfirmationIdEx

오프라인 정품 인증을 위해 IID(설치 ID) 및 CID(확인 ID)를 입금합니다. (SLDepositOfflineConfirmationIdEx)
SLFireEvent 지정된 이벤트를 등록된 수신기에 보냅니다.
SLGatherMigrationBlob 제공된 파일 핸들에 대한 라이선스 정보를 수집합니다. 이 라이선스 정보는 나중에 SLDepositMigrationBlob 함수를 사용하여 적용하거나 입금할 수 있습니다.
SLGenerateOfflineInstallationId IID(설치 ID)를 생성합니다.
SLGenerateOfflineInstallationIdEx IID(설치 ID)를 생성합니다.
SLGetApplicationInformation 지정된 애플리케이션에 대한 정보를 가져옵니다.
SLGetApplicationPolicy SLPersistApplicationPolicies 함수와 함께 저장되고 SLLoadApplicationPolicies 함수를 사용하여 로드된 집합에서 정책을 쿼리합니다.
SLGetAuthenticationResult 인증 결과를 가져옵니다.
SLGetGenuineInformation Windows 컴퓨터의 정품 상태에 대한 정보를 가져옵니다.
SLGetGenuineInformationEx Windows 컴퓨터의 정품 상태에 대한 정보를 지정합니다. (SLGetGenuineInformationEx)
SLGetInstalledProductKeyIds 이 함수는 지정된 제품 SKU ID와 연결된 제품 키 ID 목록을 반환합니다.
SLGetLicense 라이선스 파일 BLOB을 반환합니다.
SLGetLicenseFileId

라이선스 BLOB이 이미 설치되어 있는지 확인합니다.
SLGetLicenseInformation 지정된 라이선스 정보를 가져옵니다.
SLGetLicensingStatusInformation 지정된 애플리케이션 또는 SKU의 라이선스 상태를 가져옵니다.
SLGetPKeyId 제품과 연결된 등록된 제품 키 ID를 가져옵니다.
SLGetPKeyInformation 지정된 제품 키의 정보를 가져옵니다.
SLGetPolicyInformation 오른쪽이 성공적으로 사용된 후 정책 정보를 가져옵니다. (SLGetPolicyInformation)
SLGetPolicyInformationDWORD 오른쪽이 성공적으로 사용된 후 정책 정보를 가져옵니다. (SLGetPolicyInformationDWORD)
SLGetProductSkuInformation 지정된 제품 SKU에 대한 정보를 가져옵니다.
SLGetReferralInformation 지정된 제품에 대한 조회 정보를 가져옵니다.
SLGetServerStatus 지정된 URL 및 RequestType에 따라 서버 상태를 확인합니다.
SLGetServiceInformation 전역 데이터 정보를 가져옵니다.
SLGetSLIDList 입력 쿼리 ID 형식 및 ID 값에 따라 SLID 목록을 가져옵니다.
SLGetWindowsInformation 소프트웨어 구성 요소의 라이선스 정책에서 이름-값 쌍의 값 부분을 검색합니다.

SLGetWindowsInformationDWORD 소프트웨어 구성 요소의 라이선스 정책에서 이름-값 쌍의 DWORD 값 부분을 검색합니다.
SLInstallLicense 지정된 라이선스를 저장하고 라이선스 파일 ID를 반환합니다.
SLInstallProofOfPurchase 제품 키를 SL에 등록합니다.
SLInstallProofOfPurchaseEx 제품 키를 SL에 등록합니다.
SLIsGenuineLocal 지정된 애플리케이션이 정품 Windows 설치인지 여부를 확인합니다.
SLIsGenuineLocalEx 지정된 애플리케이션 설치가 정품인지 여부를 확인합니다.
SLLoadApplicationPolicies SLGetApplicationPolicy 함수에서 사용할 SLPersistApplicationPolicies 함수로 설정된 애플리케이션 정책을 로드합니다.
SLOpen SLC(소프트웨어 라이선스 클라이언트)를 초기화하고 SLC를 SLS(소프트웨어 라이선싱 서비스)에 연결합니다.
SLPersistApplicationPolicies 빠른 정책 액세스를 위해 현재 사용된 정책을 디스크에 저장합니다.
SLPersistRTSPayloadOverride 온라인 및 전화 정품 인증을 위해 지정된 제품과 정보를 연결합니다.
SLQueryLicenseValueFromApp 지정된 구성 요소 정책의 값을 가져옵니다.
SLReArm 이 함수는 애플리케이션 활성화를 다시 정렬합니다.

SLRegisterEvent SL 서비스에 이벤트를 등록합니다.
SLSetAuthenticationData 인증 데이터를 설정합니다.
SLSetCurrentProductKey 현재 제품 키를 이전에 설치된 제품 키로 설정합니다.
SLSetGenuineInformation Windows 컴퓨터의 정품 상태에 대한 정보를 지정합니다. (SLSetGenuineInformation)
SLUninstallLicense 라이선스 파일 ID 및 대상 사용자 옵션으로 지정된 라이선스를 제거합니다.
SLUninstallProofOfPurchase 제품 키 정보를 등록 취소합니다.
SLUnloadApplicationPolicies SLLoadApplicationPolicies 함수에서 반환된 정책 컨텍스트 핸들을 해제합니다.
SLUnregisterEvent SL 서비스에서 등록된 이벤트를 등록 취소합니다.
SpAcceptCredentialsFn 인증된 보안 주체에 대해 저장된 자격 증명을 보안 패키지에 전달하기 위해 LSA(로컬 보안 기관)에서 호출합니다.
SpAcceptLsaModeContextFn 서버 및 클라이언트에서 공유하는 보안 컨텍스트를 만드는 데 사용되는 서버 디스패치 함수입니다.
SpAcquireCredentialsHandleFn 보안 주체의 자격 증명에 대한 핸들을 가져오기 위해 호출됩니다.
SpAddCredentialsFn 보안 주체에 대한 자격 증명을 추가하는 데 사용됩니다.

SpApplyControlTokenFn 보안 컨텍스트에 컨트롤 토큰을 적용합니다. 이 함수는 현재 LSA(로컬 보안 기관)에서 호출되지 않습니다.
SpCompleteAuthTokenFn 인증 토큰을 완료합니다. S
SpDeleteCredentialsFn 보안 패키지의 기본 또는 추가 자격 증명 목록에서 자격 증명을 삭제합니다.
SpExchangeMetaDataFn 보안 지원 공급자에게 메타데이터를 보냅니다.
SpExportSecurityContextFn 보안 컨텍스트를 다른 프로세스로 내보냅니다.
SpFormatCredentialsFn 사용자 개체에 저장할 자격 증명의 형식을 지정합니다.
SpFreeCredentialsHandleFn SpAcquireCredentialsHandle 함수를 호출하여 획득한 자격 증명을 해제합니다.
SpGetContextTokenFn 가장할 토큰을 가져옵니다.
SpGetCredentialsFn 사용자 개체에서 기본 및 추가 자격 증명을 검색합니다.
SpGetCredUIContextFn 자격 증명 공급자에서 컨텍스트 정보를 검색합니다. (SpGetCredUIContextFn)
SpGetExtendedInformationFn 보안 패키지에 대한 확장 정보를 제공합니다.
SpGetInfoFn 보안 패키지에 대한 일반 정보(예: 이름 및 기능)를 제공합니다.
SpGetUserInfoFn

로그온 세션에 대한 정보를 검색합니다.
SpImportSecurityContextFn 다른 프로세스에서 보안 컨텍스트를 가져옵니다.
SpInitializeFn LSA(로컬 보안 기관)에서 일반 보안 정보와 지원 함수의 디스패치 테이블을 제공하는 보안 패키지를 한 번 호출합니다.
SpInitLsaModeContextFn 서버와 클라이언트 간에 보안 컨텍스트를 설정하는 데 사용되는 클라이언트 디스패치 함수입니다.
SpInitUserModeContextFn 압축된 LSA(로컬 보안 기관) 모드 컨텍스트에서 사용자 모드 보안 컨텍스트를 만듭니다.
SpInstanceInitFn SSP/AP에서 사용자 모드 보안 패키지를 초기화합니다.
SpLsaModeInitializeFn SSP/AP DLL의 각 보안 패키지에서 구현하는 함수에 대한 포인터를 LSA에 제공합니다.
SpMarshallSupplementalCredsFn 공용 형식의 추가 자격 증명을 로컬 프로시저 호출에 적합한 형식으로 변환합니다.
SpQueryContextAttributesFn 보안 컨텍스트의 특성을 검색합니다.
SpQueryCredentialsAttributesFn 자격 증명의 특성을 검색합니다.
SpQueryMetaDataFn 보안 컨텍스트를 시작할 때 SSP(보안 지원 공급자)에서 메타데이터를 가져옵니다.
SpSaveCredentialsFn 사용자 개체에 추가 자격 증명을 저장합니다.
SpSealMessageFn 클라이언트와 서버 간에 교환되는 메시지를 암호화합니다.

SpSetExtendedInformationFn 보안 패키지에 대한 확장 정보를 설정합니다.
SpUnsealMessageFn 이전에 SpSealMessage 함수로 암호화된 메시지의 암호를 해독합니다.
SpUpdateCredentialsFn 지정된 컨텍스트와 연결된 자격 증명을 업데이트합니다. (SpUpdateCredentialsFn)
SpUserModeInitializeFn 보안 지원 공급자/인증 패키지(SSP/AP) DLL이 클라이언트/서버 애플리케이션의 프로세스 공간에 로드될 때 호출됩니다. 이 함수는 SSP/AP DLL의 각 보안 패키지에 대한 SECPKG_USER_FUNCTION_TABLE 테이블을 제공합니다.
SpValidateTargetInfoFn 지정된 SECPKG_TARGETINFO 구조체가 유효한 대상을 나타내는지 확인합니다.
SslCrackCertificate 지정된 인증서 BLOB에 포함된 정보를 사용하여 X509Certificate 구조를 반환합니다.
SslEmptyCacheA Schannel 캐시에서 지정된 문자열을 제거합니다. (ANSI)
SslEmptyCacheW Schannel 캐시에서 지정된 문자열을 제거합니다. (유니코드)
SslFreeCertificate SslCrackCertificate 함수에 대한 이전 호출에 의해 할당된 인증서를 해제합니다.
SslGetServerIdentity 서버의 ID를 가져옵니다.
SspiAcceptSecurityContextAsync 전송 애플리케이션의 서버 구성 요소가 서버와 원격 클라이언트 간에 보안 컨텍스트를 비동기적으로 설정할 수 있습니다.
SspiAcquireCredentialsHandleAsyncA

보안 주체의 기존 자격 증명에 대한 핸들을 비동기적으로 획득합니다. (ANSI)
SspiAcquireCredentialsHandleAsyncW 보안 주체의 기존 자격 증명에 대한 핸들을 비동기적으로 획득합니다. (유니코드)
SspiAsyncContextRequiresNotify 지정된 비동기 컨텍스트에 호출 완료 시 알림이 필요한지 여부를 결정합니다.
SspiAsyncNotifyCallback 비동기 SSPI 호출의 완료를 알리는 데 사용되는 콜백입니다.
SspiCompareAuthIdentities 지정된 두 자격 증명을 비교합니다.
SspiCopyAuthIdentity 지정된 불투명 자격 증명 구조의 복사본을 만듭니다.
SspiCreateAsyncContext 비동기 호출을 추적하는 데 사용되는 SspiAsyncContext의 인스턴스를 만듭니다.
SspiDecryptAuthIdentity 지정된 암호화된 자격 증명의 암호를 해독합니다.
SspiDecryptAuthIdentityEx SEC_WINNT_AUTH_IDENTITY_OPAQUE 구조체의 암호를 해독합니다.
SspiDeleteSecurityContextAsync SspiInitializeSecurityContextAsync 함수 또는 SspiAcceptSecurityContextAsync 함수에 대한 이전 호출에서 시작된 지정된 보안 컨텍스트와 연결된 로컬 데이터 구조를 삭제합니다.
SspiEncodeAuthIdentityAsStrings 지정된 인증 ID를 세 개의 문자열로 인코딩합니다.
SspiEncodeStringsAsAuthIdentity 세 개의 자격 증명 문자열 집합을 인증 ID 구조로 인코딩합니다.
SspiEncryptAuthIdentity 지정된 ID 구조를 암호화합니다.

SspiEncryptAuthIdentityEx SEC_WINNT_AUTH_IDENTITY_OPAQUE 구조를 암호화합니다.
SspiExcludePackage 지정된 SSP(보안 지원 공급자)를 제외하도록 수정된 지정된 ID 구조의 복사본인 새 ID 구조를 만듭니다.
SspiFreeAsyncContext SspiCreateAsyncContext 함수 호출에서 만든 컨텍스트를 해제합니다.
SspiFreeAuthIdentity 지정된 ID 구조에 할당된 메모리를 해제합니다.
SspiFreeCredentialsHandleAsync 자격 증명 핸들을 해제합니다.
SspiGetAsyncCallStatus 제공된 컨텍스트와 연결된 비동기 호출의 현재 상태를 가져옵니다.
SspiGetCredUIContext 자격 증명 공급자에서 컨텍스트 정보를 검색합니다. (SspiGetCredUIContext)
SspiGetTargetHostName 지정된 대상과 연결된 호스트 이름을 가져옵니다.
SspiInitializeSecurityContextAsyncA 비동기 보안 컨텍스트를 초기화합니다. (ANSI)
SspiInitializeSecurityContextAsyncW 비동기 보안 컨텍스트를 초기화합니다. (유니코드)
SspiIsAuthIdentityEncrypted 지정된 ID 구조가 암호화되는지 여부를 나타냅니다.
SspiIsPromptingNeeded InitializeSecurityContext 또는 AcceptSecurityContext 함수를 호출한 후 반환된 오류가 SspiPromptForCredentials 함수에 대한 추가 호출이 필요한지 여부를 나타냅니다.

SspiLocalFree 지정된 버퍼와 연결된 메모리를 해제합니다.
SspiMarshalAuthIdentity 지정된 ID 구조를 바이트 배열로 직렬화합니다.
SspiPrepareForCredRead 지정된 ID 구조에서 대상 이름 및 자격 증명 형식을 생성합니다.
SspiPrepareForCredWrite CredWrite 함수 호출에서 매개 변수 값으로 전달될 수 있는 ID 구조에서 값을 생성합니다.
SspiPromptForCredentialsA SSPI(보안 지원 공급자 인터페이스) 애플리케이션에서 사용자에게 자격 증명을 입력하라는 메시지를 표시할 수 있습니다. (ANSI)
SspiPromptForCredentialsW SSPI(보안 지원 공급자 인터페이스) 애플리케이션에서 사용자에게 자격 증명을 입력하라는 메시지를 표시할 수 있습니다. (유니코드)
SspiReinitAsyncContext 다시 사용할 비동기 컨텍스트를 표시합니다.
SspiSetAsyncNotifyCallback 비동기 호출 완료 시 알림을 받은 콜백을 등록합니다.
SspiUnmarshalAuthIdentity 지정된 바이트 값 배열을 ID 구조로 역직렬화합니다.
SspiUnmarshalCredUIContext ICredentialProvider::SetSerialization 메서드에 대한 이전 호출 중에 자격 증명 공급자가 가져온 자격 증명 정보를 역직렬화합니다.
SspiUpdateCredentials 지정된 컨텍스트와 연결된 자격 증명을 업데이트합니다. (SspiUpdateCredentials)
SspiValidateAuthIdentity

지정된 ID 구조가 유효한지 여부를 나타냅니다.
SspiZeroAuthIdentity 지정된 ID 구조와 연결된 메모리 블록을 0으로 채웁니다.
StartServiceA 서비스를 시작합니다. (ANSI)
StartServiceCtrlDispatcherA 서비스 프로세스의 주 스레드를 서비스 제어 관리자에 연결하여 스레드가 호출 프로세스의 서비스 제어 디스패처 스레드가 됩니다. (ANSI)
StartServiceCtrlDispatcherW 서비스 프로세스의 주 스레드를 서비스 제어 관리자에 연결하여 스레드가 호출 프로세스의 서비스 제어 디스패처 스레드가 됩니다. (유니코드)
StartServiceW 서비스를 시작합니다. (유니코드)
stringToBinary 인코딩된 문자열을 이진 데이터 BLOB으로 변환합니다. 이 메서드는 ICEnroll4 인터페이스에서 처음 정의되었습니다.
stringToBinaryBlob 인코딩된 문자열을 이진 데이터 BLOB으로 변환합니다.
StringToString 문자열에 적용된 유니코드 인코딩의 형식을 수정합니다.
StringToVariantByteArray 유니코드로 인코딩된 문자열에서 바이트 배열을 만듭니다.
제출 IAzApplication 개체에 대한 변경 내용을 유지합니다.
제출 IAzApplicationGroup 개체에 대한 변경 내용을 유지합니다.

제출 AzAuthorizationStore 개체에 대한 변경 내용을 유지합니다.
제출 IAzOperation 개체에 대한 변경 내용을 유지합니다.
제출 IAzRole 개체에 대한 변경 내용을 유지합니다.
제출 IAzScope 개체에 대한 변경 내용을 유지합니다.
제출 IAzTask 개체에 대한 변경 내용을 유지합니다.
제출 인증서 서비스 서버에 요청을 제출합니다.
TokenBindingDeleteAllBindings 호출하는 사용자 또는 앱 컨테이너와 연결된 모든 토큰 바인딩 키를 삭제합니다.
TokenBindingDeleteBinding 지정된 대상 문자열과 연결된 토큰 바인딩 키를 삭제합니다.
TokenBindingGenerateBinding 토큰 바인딩에 대해 지정된 키 형식, 토큰 바인딩 키를 만들고 검색하기 위한 대상 식별자 문자열 및 고유 데이터를 사용하여 내보낸 공개 키와 서명을 포함하는 하나의 토큰 바인딩을 생성합니다.
TokenBindingGenerateID 키 형식에서 서명 알고리즘을 추출하고 내보낸 공개 키를 복사하여 토큰 바인딩 식별자를 생성합니다.
TokenBindingGenerateMessage 토큰 바인딩 목록을 어셈블하고 클라이언트 디바이스에 대한 최종 메시지를 서버에 생성합니다.
TokenBindingGetKeyTypesClient 클라이언트 디바이스에서 지원하는 키 형식의 목록을 검색합니다.

TokenBindingGetKeyTypesServer 서버에서 지원하는 키 형식의 목록을 검색합니다.
TokenBindingVerifyMessage 토큰 바인딩 메시지의 유효성을 검사하고 메시지에 포함된 토큰 바인딩을 확인합니다.
TreeResetNamedSecurityInfoA 지정된 개체 트리의 보안 설명자에서 지정된 보안 정보를 다시 설정합니다. (ANSI)
TreeResetNamedSecurityInfoW 지정된 개체 트리의 보안 설명자에서 지정된 보안 정보를 다시 설정합니다. (유니코드)
TreeSetNamedSecurityInfoA 지정된 개체 트리의 보안 설명자에 지정된 보안 정보를 설정합니다. (ANSI)
TreeSetNamedSecurityInfoW 지정된 개체 트리의 보안 설명자에 지정된 보안 정보를 설정합니다. (유니코드)
UnAdvise Advise 메서드를 호출하여 만든 연결을 삭제합니다.
초기화 해제 NDES 정책 모듈을 초기화하지 않습니다.
제거 CEP(인증서 등록 정책) 웹 서비스를 제거합니다.
제거 CES(인증서 등록 웹 서비스)를 제거합니다.
UnlockServiceDatabase 지정된 잠금을 해제하여 서비스 제어 관리자 데이터베이스의 잠금을 해제합니다.
UpdateCache 기본 정책 저장소와 일치하도록 개체 및 개체 특성의 캐시를 업데이트합니다.
UpdateRegistry

CEP(인증서 등록 정책) 서버를 등록합니다.
UpgradeStoresFunctionalLevel 이 권한 부여 저장소를 버전 1에서 버전 2로 업그레이드합니다.
유효성 검사 현재 정책 정보의 유효성을 검사합니다.
VariantByteArrayToString 바이트 배열에서 유니코드로 인코딩된 문자열을 만듭니다.
확인 프라이빗 키가 존재하며 클라이언트에서 사용할 수 있지만 키를 열지 않는지 확인합니다.
VerifyRequest 정책 모듈에 새 요청이 시스템에 들어갔다는 것을 알 수 있습니다.
VerifyRequest CA에 제출하기 위한 NDES 인증서 요청을 확인합니다.
VerifySignature MakeSignature 함수를 사용하여 서명된 메시지가 올바른 순서로 수신되었으며 수정되지 않은지 확인합니다.
WintrustAddActionID 사용자 시스템에 신뢰 공급자 작업을 추가합니다.
WintrustAddDefaultForUsage 공급자에 대한 기본 사용 식별자 및 콜백 정보를 지정합니다.
WintrustGetDefaultForUsage 기본 사용 식별자 및 콜백 정보를 검색합니다.
WintrustGetRegPolicyFlags 정책 공급자에 대한 정책 플래그를 검색합니다.
WintrustLoadFunctionPointers

지정된 작업 GUID에 대한 함수 진입점을 로드합니다. 이 함수에는 연결된 가져오기 라이브러리가 없습니다.
WintrustRemoveActionID WintrustAddActionID 함수에 의해 추가된 작업을 제거합니다. 이 함수에는 연결된 가져오기 라이브러리가 없습니다.
WintrustSetDefaultIncludePEPageHashes PE 파일에 대한 SIP(주체 인터페이스 패키지) 간접 데이터를 만들 때 페이지 해시가 포함되는지 여부를 결정하는 기본 설정을 설정합니다.
WintrustSetRegPolicyFlags 정책 공급자에 대한 정책 플래그를 설정합니다.
WinVerifyTrust 지정된 개체에 대해 신뢰 확인 작업을 수행합니다.
WinVerifyTrustEx 지정된 개체에 대해 신뢰 확인 작업을 수행하고 WINTRUST_DATA 구조체에 대한 포인터를 사용합니다.
WlxActivateUserShell 사용자 셸 프로그램을 활성화합니다.
WlxDisconnectNotify Winlogon은 터미널 서비스 네트워크 세션의 연결이 끊어지면 이 함수를 호출합니다.
WlxDisplayLockedNotice GINA에서 워크스테이션을 잠근 사용자 및 잠금 시기와 같은 잠금에 대한 정보를 표시할 수 있습니다.
WlxDisplaySASNotice Winlogon은 로그온한 사용자가 없을 때 이 함수를 호출합니다.
WlxDisplayStatusMessage Winlogon은 GINA DLL이 메시지를 표시해야 하는 경우 이 함수를 호출합니다.
WlxGetConsoleSwitchCredentials Winlogon은 이 함수를 호출하여 현재 로그온한 사용자의 자격 증명을 읽어 대상 세션으로 투명하게 전송합니다.

WlxGetStatusMessage Winlogon은 이 함수를 호출하여 GINA DLL에 의해 표시되는 상태 메시지를 가져옵니다.
WlxInitialize Winlogon은 컴퓨터에 있는 각 창 스테이션에 대해 이 함수를 한 번 호출합니다. 현재 운영 체제는 워크스테이션당 하나의 창 스테이션을 지원합니다.
WlxIsLockOk Winlogon은 워크스테이션을 잠그기 전에 이 함수를 호출합니다.
WlxIsLogoffOk Winlogon은 사용자가 로그오프 작업을 시작할 때 이 함수를 호출합니다.
WlxLoggedOnSAS Winlogon은 사용자가 로그온되고 워크스테이션이 잠겨 있지 않은 동안 SAS(보안 주의 순서) 이벤트를 받으면 이 함수를 호출합니다.
WlxLoggedOutSAS Winlogon은 사용자가 로그온되지 않은 상태에서 SAS(보안 주의 순서) 이벤트를 수신할 때 이 함수를 호출합니다.
WlxLogoff Winlogon은 이 함수를 호출하여 GINA에 이 워크스테이션에서 로그오프 작업을 알리고 GINA가 필요할 수 있는 모든 로그오프 작업을 수행할 수 있도록 합니다.
WlxNegotiate WlxNegotiate 함수는 대체 GINA DLL에 의해 구현되어야 합니다. Winlogon에서 GINA DLL에 대한 첫 번째 호출입니다. WlxNegotiate를 사용하면 GINA가 설치된 Winlogon 버전을 지원하는지 확인할 수 있습니다.
WlxNetworkProviderLoad Winlogon은 이 함수를 호출하여 유효한 인증 및 식별 정보를 수집합니다.
WlxReconnectNotify Winlogon은 터미널 서비스 네트워크 세션이 다시 연결될 때 이 함수를 호출합니다.
WlxRemoveStatusMessage Winlogon은 이 함수를 호출하여 GINA DLL에 상태 메시지 표시를 중지하도록 지시합니다.

WlxScreenSaverNotify Winlogon은 화면 보호기가 활성화되기 직전에 이 함수를 호출하여 GINA가 화면 보호기 프로그램과 상호 작용할 수 있도록 합니다.
WlxShutdown Winlogon은 종료 직전에 이 함수를 호출하여 GINA가 판독기에서 스마트 카드를 꺼내는 것과 같은 모든 종료 작업을 수행할 수 있도록 합니다.
WlxStartApplication Winlogon은 시스템에서 사용자의 컨텍스트에서 애플리케이션을 시작해야 하는 경우 이 함수를 호출합니다.
WlxWkstaLockedSAS Winlogon은 SAS(보안 주의 순서)를 수신하고 워크스테이션이 잠겨 있을 때 이 함수를 호출합니다.
WNetSetLastErrorA 확장 오류 정보를 설정합니다. 네트워크 공급자는 SetLastError 대신 이 함수를 호출해야 합니다. (ANSI)
WNetSetLastErrorW 확장 오류 정보를 설정합니다. 네트워크 공급자는 SetLastError 대신 이 함수를 호출해야 합니다. (유니코드)
WTHelperCertCheckValidSignature 서명이 유효한지 여부를 확인합니다.
WTHelperCertIsSelfSigned 인증서가 자체 서명되었는지 여부를 확인합니다.
WTHelperGetProvCertFromChain 인증서 체인에서 트러스트 공급자 인증서를 검색합니다.
WTHelperGetProvPrivateDataFromChain 공급자 ID를 사용하여 체인에서 CRYPT_PROVIDER_PRIVDATA 구조를 받습니다.
WTHelperGetProvSignerFromChain 체인에서 인덱스로 서명자 또는 카운터 서명자를 검색합니다.
WTHelperProvDataFromStateData

지정된 핸들에서 신뢰 공급자 정보를 검색합니다.

인터페이스

테이블 확장

IAlternativeName IX509ExtensionAlternativeNames 개체에서 AlternativeNames 확장의 인스턴스를 나타내는 데 사용됩니다.
IAlternativeNames IAlternativeName 개체의 컬렉션을 관리할 수 있는 메서드와 속성을 포함합니다.
IAssociatedIdentityProvider ID 공급자가 로컬 사용자 계정과 ID를 연결할 수 있도록 허용합니다.
IAzApplication 애플리케이션의 설치된 인스턴스를 정의합니다. IAzApplication 개체는 애플리케이션이 설치될 때 만들어집니다.
IAzApplication2 IAzApplication 인터페이스에서 상속되며 IAzClientContext2 개체를 초기화하는 추가 메서드를 구현합니다.
IAzApplication3 IAzRoleAssignment, IAzRoleDefinition 및 IAzScope2 개체를 관리하는 메서드를 제공합니다.
IAzApplicationGroup 보안 주체의 컬렉션을 정의합니다.
IAzApplicationGroup2 BizRule 그룹 유형에 대한 지원을 추가하여 IAzApplicationGroup 인터페이스를 확장합니다.
IAzApplicationGroups IAzApplicationGroup 개체의 컬렉션을 나타냅니다.
IAzApplications

IAzApplication 개체의 컬렉션을 나타냅니다.
IAzAuthorizationStore 권한 부여 정책 저장소의 루트인 컨테이너를 정의합니다.
IAzAuthorizationStore2 AzAuthorizationStore 개체에서 상속되며 IAzApplication2 개체를 만들고 여는 메서드를 구현합니다.
IAzAuthorizationStore3 비즈니스 규칙(BizRule) 지원 및 캐싱을 관리하는 메서드를 사용하여 IAzAuthorizationStore2 인터페이스를 확장합니다.
IAzBizRuleContext BizRule(비즈니스 규칙) 작업에 대한 정보를 포함합니다.
IAzBizRuleInterfaces BizRule(비즈니스 규칙) 스크립트에서 호출할 수 있는 IDispatch 인터페이스 목록을 관리하는 데 사용되는 메서드 및 속성을 제공합니다.
IAzBizRuleParameters 비즈니스 규칙(BizRule) 스크립트에 전달할 수 있는 매개 변수 목록을 관리하는 데 사용되는 메서드 및 속성을 제공합니다.
IAzClientContext 특정 클라이언트를 설명하는 상태를 유지 관리합니다.
IAzClientContext2 IAzClientContext 인터페이스에서 상속되며 클라이언트 컨텍스트를 조작하는 새 메서드를 구현합니다.
IAzClientContext3 IAzClientContext2 인터페이스를 확장합니다.
IAzNameResolver SID(보안 식별자)를 보안 식별자 표시 이름으로 변환합니다.
IAzObjectPicker 사용자가 목록에서 하나 이상의 보안 주체를 선택할 수 있는 대화 상자를 표시합니다.

IAzOperation 애플리케이션에서 지원하는 하위 수준 작업을 정의합니다.
IAzOperation2 작업과 연결된 역할 할당을 반환하는 메서드를 사용하여 IAzOperation을 확장합니다.
IAzOperations IAzOperation 개체의 컬렉션을 나타냅니다.
IAzPrincipalLocator 권한 부여 관리자에서 ADAM 보안 주체를 찾아 선택합니다.
IAzRole 범위 내의 사용자 집합에서 수행할 수 있는 작업 집합을 정의합니다.
IAzRoleAssignment 사용자 및 그룹을 할당할 수 있는 역할을 나타냅니다.
IAzRoleAssignments IAzRoleAssignment 개체의 컬렉션을 나타냅니다.
IAzRoleDefinition 작업 집합을 지정하는 하나 이상의 IAzRoleDefinition, IAzTask 및 IAzOperation 개체를 나타냅니다.
IAzRoleDefinitions IAzRoleDefinition 개체의 컬렉션을 나타냅니다.
IAzRoles IAzRole 개체의 컬렉션을 나타냅니다.
IAzScope 애플리케이션이 액세스를 관리하는 리소스의 논리적 컨테이너를 정의합니다.
IAzScope2 IAzScope 인터페이스를 확장하여 IAzRoleAssignment 및 IAzRoleDefinition 개체를 관리합니다.
IAzScopes

IAzScope 개체의 컬렉션을 나타냅니다.
IAzTask 작업 집합에 대해 설명합니다.
IAzTask2 작업과 연결된 역할 할당을 반환하는 메서드를 사용하여 IAzTask 인터페이스를 확장합니다.
IAzTasks IAzTask 개체의 컬렉션을 나타냅니다.
IBinaryConverter 바이트 배열에서 유니코드로 인코딩된 문자열을 만들고, 유니코드로 인코딩된 문자열에서 바이트 배열을 만들고, 문자열에 적용된 유니코드 인코딩 형식을 수정할 수 있는 일반적인 메서드가 포함되어 있습니다.
ICcgDomainAuthCredentials 개발자가 런타임에 자신의 자격 증명을 동적으로 제공하여 Active Directory를 사용하여 도메인에 가입되지 않은 컨테이너를 인증할 수 있는 클라이언트 구현 인터페이스입니다.
ICEnroll ICEnroll 인터페이스는 인증서 등록 컨트롤을 나타내는 여러 인터페이스 중 하나입니다.
ICEnroll2 ICEnroll2 인터페이스는 인증서 등록 컨트롤을 나타내는 여러 인터페이스 중 하나입니다.
ICEnroll3 인증서 등록 제어를 나타내는 여러 인터페이스 중 하나입니다.
ICEnroll4 ICEnroll4 인터페이스는 인증서 등록 컨트롤을 나타내는 여러 인터페이스 중 하나입니다.
ICertAdmin 제대로 권한이 부여된 클라이언트에 대한 관리 기능을 제공합니다.
ICertAdmin2 제대로 권한이 부여된 클라이언트에 대한 관리 기능을 제공합니다.
ICertConfig

ICertConfig 인터페이스는 인증서 서비스 서버에 대한 공용 구성 데이터(클라이언트 설정 중에 지정됨)를 검색하는 기능을 제공합니다.
ICertConfig2 인증서 서비스 서버에 대한 공용 구성 데이터(클라이언트 설정 중에 지정됨)를 검색하는 기능을 제공합니다.
ICertEncodeAltName 인증서 확장에 사용되는 대체 이름을 처리하는 메서드를 제공합니다.
ICertEncodeBitString 인증서 확장에 사용되는 비트 문자열을 처리하는 메서드를 제공합니다.
ICertEncodeCRLDistInfo 인증서 확장에 사용되는 CRL(인증서 해지 목록) 배포 정보 배열을 처리하는 메서드를 제공합니다.
ICertEncodeDateArray 인증서 확장에 사용되는 날짜 배열을 처리하는 메서드를 제공합니다.
ICertEncodeLongArray 인증서 확장에 사용되는 Long 배열을 처리하는 메서드를 제공합니다.
ICertEncodeStringArray 인증서 확장에 사용되는 문자열 배열을 처리하는 메서드를 제공합니다.
ICertExit 인증서 서비스 서버와 종료 모듈 간의 통신을 제공합니다.
ICertExit2 인증서 서비스 서버와 종료 모듈 간의 통신을 제공합니다.
ICertGetConfig 인증서 서비스 서버에 대한 공용 구성 데이터(클라이언트 설정 중에 지정됨)를 검색하는 기능을 제공합니다.
ICertificateAttestationChallenge 애플리케이션에서 서버에서 받은 주요 증명 챌린지의 암호를 해독할 수 있습니다.

ICertificateEnrollmentPolicyServerSetup ICertificateEnrollmentPolicyServerSetup 인터페이스는 ADCS(Active Directory Certificate Services) 내의 CEP(인증서 등록 정책) 웹 서비스를 나타냅니다.
ICertificateEnrollmentServerSetup ICertificateEnrollmentServerSetup 인터페이스는 ADCS(Active Directory 인증서 서비스) 내의 CES(인증서 등록 웹 서비스)를 나타냅니다.
ICertificatePolicies ICertificatePolicy 개체의 컬렉션을 관리할 수 있는 메서드와 속성을 포함합니다.
ICertificatePolicy 인증서를 사용할 수 있는 목적을 식별하는 인증서 정책을 지정하는 데 사용할 수 있습니다.
ICertificationAuthorities ICertificationAuthorities 인터페이스는 ICertificationAuthority 개체의 컬렉션을 관리하는 다음 메서드 및 속성을 정의합니다.
ICertificationAuthority ICertificationAuthority 인터페이스는 단일 인증 기관을 나타냅니다. 인증 기관의 컬렉션은 ICertificationAuthorities 인터페이스로 표시됩니다.
ICertManageModule 인증서 서비스 정책 또는 종료 모듈에 대한 정보를 검색하기 위해 제공됩니다.
ICertPolicy 인증서 서비스 서버 엔진과 정책 모듈 간의 통신을 제공합니다.
ICertPolicy2 인증서 서비스 서버 엔진과 정책 모듈 간의 통신을 제공합니다.
ICertProperties 인증서 속성의 컬렉션을 관리할 수 있는 메서드와 속성을 포함합니다.
ICertProperty 외부 속성을 인증서와 연결하는 데 사용할 수 있습니다.

ICertPropertyArchived 인증서가 보관되었는지 여부를 식별하는 인증서 속성을 나타냅니다.
ICertPropertyArchivedKeyHash 보관을 위해 인증 기관에 제출된 암호화된 프라이빗 키의 SHA-1 해시를 나타냅니다.
ICertPropertyAutoEnroll 인증서 자동 등록을 사용하도록 구성된 템플릿을 식별하는 인증서 속성을 나타냅니다.
ICertPropertyBackedUp 인증서가 백업되었는지 여부와 인증서가 저장된 날짜 및 시간을 식별하는 외부 인증서 속성을 나타냅니다.
ICertPropertyDescription 인증서에 대한 설명 정보가 포함된 문자열을 지정하고 검색할 수 있습니다.
ICertPropertyEnrollment 클라이언트가 IX509Enrollment 인터페이스에서 등록 메서드를 호출할 때 생성되는 CA(인증 기관) 정보를 포함하는 인증서 속성을 나타냅니다.
ICertPropertyEnrollmentPolicyServer CEP(인증서 등록 정책) 서버 및 CES(인증서 등록 서버)에 대한 정보가 포함된 외부 인증서 속성을 나타냅니다.
ICertPropertyFriendlyName 인증서의 표시 이름을 포함하는 문자열을 지정하고 검색할 수 있습니다.
ICertPropertyKeyProvInfo 프라이빗 키에 대한 정보를 포함하는 인증서 속성을 나타냅니다.
ICertPropertyRenewal 기존 인증서를 갱신할 때 만든 새 인증서의 SHA-1 해시를 포함하는 인증서 속성을 나타냅니다.
ICertPropertyRequestOriginator 요청을 만든 컴퓨터의 DNS(Domain Naming System) 이름을 포함하는 인증서 속성을 나타냅니다.
ICertPropertySHA1Hash 인증서의 SHA-1 해시를 포함하는 인증서 속성을 나타냅니다.

ICertRequest 클라이언트 또는 중간 애플리케이션과 인증서 서비스 간의 통신을 제공합니다.
ICertRequest2 클라이언트 또는 중간 애플리케이션과 Certificate Services 간의 통신을 제공합니다. (ICertRequest2)
ICertRequest3 클라이언트 또는 중간 애플리케이션과 Certificate Services 간의 통신을 제공합니다. (ICertRequest3)
ICertServerExit 서버 엔진에서 내보내고 종료 모듈에서 호출됩니다.
ICertServerPolicy 정책 모듈이 인증서 서비스와 통신할 수 있도록 허용합니다.
ICertSrvSetup 인증서 서비스 컴퓨터에 CA(인증 기관) 및 인증 기관 웹 등록 역할을 설치하고 제거하는 기능을 정의합니다.
ICertSrvSetupKeyInformation CA(인증 기관) 또는 MICROSOFT SCEP(단순 인증서 등록 프로토콜) 역할 설정에 사용되는 프라이빗 키 속성 집합을 정의합니다.
ICertSrvSetupKeyInformationCollection ICertSrvSetupKeyInformation 개체의 컬렉션을 채우고 열거하는 기능을 정의합니다.
ICertView 제대로 권한이 부여된 클라이언트가 인증서 서비스 데이터베이스의 사용자 지정 또는 전체 보기를 만들 수 있습니다.
ICertView2 제대로 권한이 부여된 클라이언트가 인증서 서비스 데이터베이스의 사용자 지정 또는 전체 보기를 만들 수 있도록 허용합니다.
IConnectedIdentityProvider 연결된 ID 공급자와의 상호 작용 메서드를 제공합니다.

ICryptAttribute ICryptAttribute 인터페이스는 인증서 요청의 암호화 특성을 나타냅니다. 이러한 특성의 컬렉션은 PKCS의 CertificateRequestInfo 구조에 포함됩니다.
ICryptAttributes ICryptAttributes 인터페이스에는 ICryptAttribute 개체의 컬렉션을 관리할 수 있는 메서드와 속성이 포함되어 있습니다.
ICspAlgorithm 암호화 공급자가 구현한 알고리즘을 나타냅니다.
ICspAlgorithms ICspAlgorithms 인터페이스는 ICspAlgorithm 개체의 컬렉션을 관리하는 다음 메서드 및 속성을 정의합니다.
iCspInformation 암호화 공급자에 대한 일반 정보에 대한 액세스를 제공합니다.
iCspInformations ICspInformations 인터페이스는 ICspInformation 개체의 컬렉션을 관리하는 다음 메서드와 속성을 정의합니다.
ICspStatus 암호화 공급자/알고리즘 쌍에 대한 정보를 포함합니다. (ICspStatus)
ICspStatuses 암호화 공급자/알고리즘 쌍에 대한 정보를 포함합니다. (ICspStatuses)
IEffectivePermission 개체에 대한 보안 주체에 대한 유효 권한을 결정하는 방법을 제공합니다.
IEffectivePermission2 개체에 대한 보안 주체에 대한 유효 권한을 확인하는 방법을 제공합니다.
IEnroll 인증서 등록 컨트롤을 나타내며 주로 인증서 요청을 생성하는 데 사용됩니다. (IEnroll)

IEnroll2 인증서 등록 컨트롤을 나타내며 주로 인증서 요청을 생성하는 데 사용됩니다. (IEnroll2)
IEnroll4 IEnroll4 인터페이스는 인증서 등록 제어를 나타내며 주로 인증서 요청을 생성하는 데 사용됩니다.
IEnumCERTVIEWATTRIBUTE 행 열거형 시퀀스의 현재 행에 대한 인증서 특성을 포함하는 특성 열거형 시퀀스를 나타냅니다.
IEnumCERTVIEWCOLUMN 열거형 시퀀스의 현재 행에 대한 열 데이터를 포함하는 열거형 시퀀스를 나타냅니다.
IEnumCERTVIEWEXTENSION 행 열거형 시퀀스의 현재 행에 대한 인증서 확장 데이터를 포함하는 확장 열거형 시퀀스를 나타냅니다.
IEnumCERTVIEWROW 인증서 서비스 보기의 행에 있는 데이터를 포함하는 행 열거형 시퀀스를 나타내며, 각 행과 연결된 열, 특성 및 확장에 대한 추가 액세스를 허용합니다.
IIdentityAdvise ID가 업데이트되면 ID 공급자가 호출 애플리케이션에 알릴 수 있습니다.
IIdentityProvider ID 공급자를 나타냅니다.
IIdentityStore ID 및 ID 공급자를 열거하고 관리하는 메서드를 제공합니다.
IMSCEPSetup 인증서 서비스 컴퓨터에 NDES(네트워크 디바이스 등록 서비스) 역할을 설치하고 제거하는 기능을 정의합니다.
INDESPolicy NDES 정책 모듈 인터페이스입니다. 엔터프라이즈 CA에 대해 설치되면 NDES는 사용자가 구성된 NDES 템플릿(사용자 및 컴퓨터 템플릿 모두)에 대한 등록 권한이 있는지 확인한 후 암호를 생성합니다.
IObjectId

OID(개체 식별자)를 나타냅니다.
IObjectIds IObjectIds 인터페이스는 IObjectId 개체의 컬렉션을 관리할 수 있는 메서드와 속성을 정의합니다.
IOCSPAdmin OCSP(온라인 인증서 상태 프로토콜) 응답자 서버를 관리하는 기능을 제공합니다.
IOCSPCAConfiguration OCSP(온라인 인증서 상태 프로토콜) 서비스가 특정 CA(인증 기관)에 대한 인증서 상태 요청에 응답할 수 있도록 하는 정의 집합을 나타냅니다.
IOCSPCAConfigurationCollection OCSP(온라인 인증서 상태 프로토콜) 서비스가 인증서 상태 응답을 제공하도록 구성된 인증서 집합을 나타냅니다.
IOCSPProperty OCSPServiceProperties 또는 ProviderProperties의 이름-값 쌍을 나타냅니다.
IOCSPPropertyCollection OCSP(온라인 인증서 상태 프로토콜) 서비스에 대한 구성 가능한 특성 속성(이름-값 쌍) 집합을 나타냅니다.
IPolicyQualifier 인증서 정책과 연결할 수 있는 한정자를 나타냅니다.
IPolicyQualifiers IPolicyQualifier 개체의 컬렉션을 관리할 수 있는 메서드와 속성을 정의합니다.
ISceSvcAttachmentData ISceSvcAttachmentData 인터페이스는 보안 구성 스냅인에서 지정된 보안 서비스에 대한 구성 및 분석 데이터를 검색합니다.
ISceSvcAttachmentPersistInfo ISceSvcAttachmentPersistInfo 인터페이스는 첨부 파일 스냅인에서 수정된 구성 또는 분석 정보를 검색합니다.
ISecurityInformation 액세스 제어 편집기가 CreateSecurityPage 및 EditSecurity 함수의 호출자와 통신할 수 있도록 합니다.

ISecurityInformation2 액세스 제어 편집기가 ISecurityInformation 인터페이스에서 제공하지 않는 클라이언트에서 정보를 가져올 수 있도록 합니다.
ISecurityInformation3 사용자가 해당 편집 단추에 방패 이미지를 표시하는 액세스 제어 편집기 페이지에서 편집 단추를 클릭할 때 상승된 액세스 제어 편집기를 표시하는 데 필요한 메서드를 제공합니다.
ISecurityInformation4 ACE(액세스 제어 편집기)가 공유의 보안 설명자를 가져와서 공유 페이지를 초기화할 수 있도록 합니다.
ISecurityObjectTypeInfo DCL(임의 액세스 제어 목록) 및 SACL(시스템 액세스 제어 목록)에서 상속된 ACE(액세스 제어 항목)의 원본을 결정하는 방법을 제공합니다.
ISignerCertificate 인증서 요청에 서명할 수 있는 서명 인증서를 나타냅니다.
ISignerCertificates ISignerCertificates 인터페이스는 다음 메서드와 속성을 정의하여 ISignerCertificate 개체의 컬렉션을 관리합니다.
ISmimeCapabilities 다음 메서드와 속성을 정의하여 ISmimeCapability 개체의 컬렉션을 관리합니다.
ISmimeCapability 전자 메일 받는 사람의 암호 해독 기능을 식별하는 SMIMECapabilities 확장을 나타냅니다.
ITpmVirtualSmartCardManager TPM 가상 스마트 카드를 관리합니다.
ITpmVirtualSmartCardManagerStatusCallback 호출자에게 요청된 작업의 진행률 또는 결과 오류를 알 수 있습니다.
IX500DistinguishedName X.500 DN(고유 이름)을 나타냅니다.

IX509Attribute PKCS의 특성을 나타내는 데 사용할 수 있습니다.
IX509AttributeArchiveKey 인증 기관에서 보관할 암호화된 프라이빗 키가 포함된 특성을 나타냅니다.
IX509AttributeArchiveKeyHash 인증 기관에서 보관할 암호화된 프라이빗 키의 SHA-1 해시를 포함하는 특성을 나타냅니다.
IX509AttributeClientId 인증서 요청을 생성한 클라이언트를 식별하는 데 사용할 수 있는 특성을 나타냅니다.
IX509AttributeCspProvider 인증서를 요청하는 엔터티에서 사용하는 암호화 공급자를 식별하는 특성을 나타냅니다.
IX509AttributeExtensions 인증서 요청에서 인증서 확장을 초기화하고 검색하는 메서드 및 속성을 정의합니다.
IX509AttributeOSVersion 인증서 요청이 생성된 클라이언트 운영 체제에 대한 버전 정보가 포함된 특성을 나타냅니다.
IX509AttributeRenewalCertificate 갱신되는 인증서가 포함된 특성을 나타냅니다. 이 특성은 PKCS에 자동으로 배치됩니다.
IX509Attributes IX509Attributes 인터페이스는 IX509Attribute 개체의 컬렉션을 관리할 수 있는 다음 메서드와 속성을 정의합니다.
IX509CertificateRequest IX509CertificateRequest 인터페이스는 인증서 등록 API에 의해 구현된 각 요청 개체에 공통되고 상속되는 메서드 및 속성을 식별하는 추상 기본 인증서 요청을 나타냅니다.
IX509CertificateRequestCertificate IX509CertificateRequestCertificate 인터페이스는 자체 생성된 인증서에 대한 요청 개체를 나타내며, 등록 또는 인증 기관을 거치지 않고도 인증서를 직접 만들 수 있습니다.
IX509CertificateRequestCertificate2

IX509CertificateRequestCertificate2 인터페이스는 자체 생성된 인증서에 대한 요청 개체를 나타내며, 등록 또는 인증 기관을 거치지 않고도 인증서를 직접 만들 수 있습니다.
IX509CertificateRequestCmc CMC(CMS를 통해 인증서 관리 메시지) 인증서 요청을 나타냅니다.
IX509CertificateRequestCmc2 IX509CertificateRequestCmc2 인터페이스는 CMC(CMS를 통해 인증서 관리 메시지) 인증서 요청을 나타냅니다.
IX509CertificateRequestPkcs10 IX509CertificateRequestPkcs10 인터페이스는 PKCS를 나타냅니다.
IX509CertificateRequestPkcs10V2 IX509CertificateRequestPkcs10V2 인터페이스는 PKCS를 나타냅니다.
IX509CertificateRequestPkcs10V3 IX509CertificateRequestPkcs10V3 인터페이스는 PKCS를 나타냅니다.
IX509CertificateRequestPkcs7 IX509CertificateRequestPkcs7 인터페이스는 PKCS를 나타냅니다.
IX509CertificateRequestPkcs7V2 IX509CertificateRequestPkcs7V2 인터페이스는 PKCS를 나타냅니다.
IX509CertificateTemplate IX509CertificateTemplate 인터페이스는 인증서 요청 템플릿을 나타냅니다. IX509CertificateTemplateWritable 인터페이스를 초기화하는 데 사용할 수 있습니다.
IX509CertificateTemplates IX509CertificateTemplates 인터페이스는 IX509CertificateTemplate 개체의 컬렉션을 관리하는 다음 메서드와 속성을 정의합니다.
IX509CertificateTemplateWritable IX509CertificateTemplateWritable 인터페이스를 사용하면 템플릿 저장소에 템플릿을 추가하거나 템플릿 저장소에서 삭제할 수 있습니다. 현재 Active Directory는 사용 가능한 유일한 저장소입니다.
IX509EndorsementKey

X.509 인증 키 인터페이스
IX509Enrollment 최상위 개체를 나타내며 인증서 계층 구조에 등록하고 인증서 응답을 설치할 수 있습니다.
IX509Enrollment2 IX509Enrollment2 인터페이스를 사용하면 인증서 계층 구조에 등록하고 인증서 응답을 설치할 수 있습니다.
IX509EnrollmentHelper IX509EnrollmentHelper 인터페이스는 웹 애플리케이션이 인증서를 등록하고, 자격 증명 캐시에 정책 서버 자격 증명을 저장하고, 정책 서버 및 등록 서버를 등록할 수 있도록 하는 메서드를 정의합니다.
IX509EnrollmentPolicyServer IX509EnrollmentPolicyServer 인터페이스는 CEP(인증서 등록 정책) 서버를 나타냅니다.
IX509EnrollmentStatus IX509EnrollmentStatus 인터페이스를 사용하여 인증서 등록 트랜잭션에 대한 자세한 오류 정보를 지정하거나 검색할 수 있습니다.
IX509EnrollmentWebClassFactory 웹 페이지에서 다음 개체를 만드는 데 사용할 수 있습니다.
IX509Extension 인증서 요청에 대한 확장을 정의하는 데 사용할 수 있습니다.
IX509ExtensionAlternativeNames 인증서의 주체에 대해 하나 이상의 대체 이름 양식을 지정할 수 있습니다. 인증 기관은 인증된 공개 키에 이름을 바인딩하여 확장을 처리합니다.
IX509ExtensionAuthorityKeyIdentifier AuthorityKeyIdentifier 확장을 지정할 수 있습니다.
IX509ExtensionBasicConstraints 인증서 주체가 인증 기관인지 여부를 지정하고, 그렇다면 이 확장 ID가 정의된 인증 기관 아래에 존재할 수 있는 하위 인증 기관 체인의 깊이를 지정할 수 있습니다.
IX509ExtensionCertificatePolicies

각각 OID(개체 식별자) 및 선택적 정책 한정자로 구성된 정책 정보 용어 컬렉션을 지정할 수 있습니다. 단일 정책 용어는 ICertificatePolicy 개체에 의해 정의됩니다.
IX509ExtensionEnhancedKeyUsage 인증서에 포함된 공개 키의 의도된 사용을 식별하는 OID(개체 식별자) 컬렉션을 정의하는 데 사용할 수 있습니다.
IX509ExtensionKeyUsage 인증서에 포함된 공개 키로 수행할 수 있는 작업에 대한 제한을 정의하는 데 사용할 수 있습니다.
IX509ExtensionMSApplicationPolicies 애플리케이션에서 인증서를 사용하는 방법을 나타내는 OID(개체 식별자) 컬렉션을 지정할 수 있습니다.
IX509Extensions IX509Extensions 인터페이스는 IX509Extension 개체의 컬렉션을 관리하는 다음 메서드와 속성을 정의합니다.
IX509ExtensionSmimeCapabilities 보낸 사람이 양쪽에서 지원하는 가장 안전한 알고리즘을 선택할 수 있도록 전자 메일 받는 사람의 암호 해독 기능을 전자 메일 보낸 사람에게 보고하는 데 사용할 수 있습니다.
IX509ExtensionSubjectKeyIdentifier SubjectKeyIdentifier 확장을 지정할 수 있습니다.
IX509ExtensionTemplate CertificateTemplate 확장을 초기화하거나 검색하는 데 사용할 수 있는 메서드와 속성을 정의합니다.
IX509ExtensionTemplateName 템플릿 이름 확장을 초기화하거나 검색하는 데 사용할 수 있는 메서드 및 속성을 정의합니다.
IX509MachineEnrollmentFactory 웹 페이지에서 IX509EnrollmentHelper 개체를 만드는 데 사용할 수 있습니다.
IX509NameValuePair 제네릭 이름-값 쌍을 나타냅니다.
IX509NameValuePairs

IX509NameValuePairs 인터페이스는 IX509NameValuePair 개체의 컬렉션을 관리하는 다음 메서드와 속성을 정의합니다.
IX509PolicyServerListManager IX509PolicyServerListManager 인터페이스는 IX509PolicyServerUrl 개체의 컬렉션을 관리할 수 있는 다음 메서드와 속성을 정의합니다.
IX509PolicyServerUrl IX509PolicyServerUrl 인터페이스를 사용하여 CEP(인증서 등록 정책) 서버와 연결된 속성 값을 설정하거나 검색하고 연결된 레지스트리 값을 업데이트할 수 있습니다.
IX509PrivateKey 암호화, 서명 및 키 계약에 사용할 수 있는 비대칭 프라이빗 키를 나타냅니다.
IX509PublicKey 공개/프라이빗 키 쌍의 공개 키를 나타냅니다.
IX509SCEPEnrollment X.509 간단한 컴퓨터 등록 프로토콜 인터페이스
IX509SignatureInformation 인증서 요청에 서명하는 데 사용되는 정보를 나타냅니다.

구조

테이블 확장

ACCESS_ALLOWED_ACE 개체에 대한 액세스를 제어하는 DACL(임의 액세스 제어 목록)에 대한 ACE(액세스 제어 항목)를 정의합니다. 액세스 허용 ACE를 사용하면 SID(보안 식별자)로 식별된 특정 트러스티에 대한 개체에 액세스할 수 있습니다.
ACCESS_ALLOWED_CALLBACK_ACE ACCESS_ALLOWED_CALLBACK_ACE 구조체는 개체에 대한 액세스를 제어하는 임의 액세스 제어 목록에 대한 액세스 제어 항목을 정의합니다.

ACCESS_ALLOWED_CALLBACK_OBJECT_ACE 개체, 속성 집합 또는 속성에 대한 허용된 액세스를 제어하는 ACE(액세스 제어 항목)를 정의합니다.
ACCESS_ALLOWED_OBJECT_ACE 개체, 속성 집합 또는 속성에 대한 허용된 액세스를 제어하는 ACE(액세스 제어 항목)를 정의합니다.
ACCESS_DENIED_ACE 개체에 대한 액세스를 제어하는 DACL(임의 액세스 제어 목록)에 대한 ACE(액세스 제어 항목)를 정의합니다. 액세스 거부 ACE는 SID(보안 식별자)로 식별된 특정 트러스티에 대한 개체에 대한 액세스를 거부합니다.
ACCESS_DENIED_CALLBACK_ACE ACCESS_DENIED_CALLBACK_ACE 구조체는 개체에 대한 액세스를 제어하는 임의 액세스 제어 목록에 대한 액세스 제어 항목을 정의합니다.
ACCESS_DENIED_CALLBACK_OBJECT_ACE ACCESS_DENIED_CALLBACK_OBJECT_ACE 구조체는 개체, 속성 집합 또는 속성에 대한 거부된 액세스를 제어하는 액세스 제어 항목을 정의합니다.
ACCESS_DENIED_OBJECT_ACE 개체, 속성 집합 또는 속성에 대한 액세스 거부를 제어하는 ACE(액세스 제어 항목)를 정의합니다.
ACE_HEADER ACE(액세스 제어 항목)의 형식과 크기를 정의합니다.
ACL ACL(액세스 제어 목록)의 헤더입니다.
ACL_REVISION_INFORMATION ACL 구조에 대한 수정 정보를 포함합니다.
ACL_SIZE_INFORMATION ACL 구조체의 크기에 대한 정보를 포함합니다.
AUDIT_POLICY_INFORMATION 보안 이벤트 유형 및 해당 형식을 감사할 시기를 지정합니다.

AUTHENTICODE_EXTRA_CERT_CHAIN_POLICY_PARA 파일의 인증서 체인 확인에 사용되는 정책 정보를 보유합니다.
AUTHENTICODE_EXTRA_CERT_CHAIN_POLICY_STATUS AUTHENTICODE_EXTRA_CERT_CHAIN_POLICY_STATUS 구조는 파일의 체인 확인을 위한 추가 Authenticode 정책 정보를 보유합니다.
AUTHENTICODE_TS_EXTRA_CERT_CHAIN_POLICY_PARA AUTHENTICODE_TS_EXTRA_CERT_CHAIN_POLICY_PARA 구조에는 파일의 인증서 체인 확인에 사용할 수 있는 타임스탬프를 정책 정보가 포함됩니다.
AUTHZ_ACCESS_REPLY 액세스 확인 회신을 정의합니다.
AUTHZ_ACCESS_REQUEST 액세스 확인 요청을 정의합니다.
AUTHZ_INIT_INFO 리소스 관리자에 대한 초기화 정보를 정의합니다.
AUTHZ_REGISTRATION_OBJECT_TYPE_NAME_OFFSET 등록 개체 형식 이름의 오프셋을 지정합니다.
AUTHZ_RPC_INIT_INFO_CLIENT 클라이언트에 대한 원격 리소스 관리자를 초기화합니다.
AUTHZ_SECURITY_ATTRIBUTE_FQBN_VALUE 보안 특성과 연결된 정규화된 이진 이름 값을 지정합니다.
AUTHZ_SECURITY_ATTRIBUTE_OCTET_STRING_VALUE 보안 특성에 대한 8진수 문자열 값을 지정합니다.
AUTHZ_SECURITY_ATTRIBUTE_V1 권한 부여 컨텍스트와 연결할 수 있는 보안 특성을 정의합니다.
AUTHZ_SECURITY_ATTRIBUTES_INFORMATION 하나 이상의 보안 특성 및 값을 지정합니다.

AUTHZ_SOURCE_SCHEMA_REGISTRATION 원본 스키마 등록에 대한 정보를 지정합니다.
BCRYPT_ALGORITHM_IDENTIFIER 암호화 알고리즘 식별자를 포함하기 위해 BCryptEnumAlgorithms 함수와 함께 사용됩니다.
BCRYPT_AUTHENTICATED_CIPHER_MODE_INFO 인증된 암호 모드와 관련된 추가 정보를 포함하기 위해 BCryptEncrypt 및 BCryptDecrypt 함수와 함께 사용됩니다.
BCRYPT_DH_KEY_BLOB 메모리에서 Diffie-Hellman 공개 키 또는 프라이빗 키 BLOB의 헤더로 사용됩니다.
BCRYPT_DH_PARAMETER_HEADER Diffie-Hellman 키에 대한 매개 변수 헤더 정보를 포함하는 데 사용됩니다.
BCRYPT_DSA_KEY_BLOB DSA(디지털 서명 알고리즘) 공개 키 또는 메모리의 프라이빗 키 BLOB에 대한 헤더로 사용됩니다. (BCRYPT_DSA_KEY_BLOB)
BCRYPT_DSA_KEY_BLOB_V2 DSA(디지털 서명 알고리즘) 공개 키 또는 메모리의 프라이빗 키 BLOB에 대한 헤더로 사용됩니다. (BCRYPT_DSA_KEY_BLOB_V2)
BCRYPT_DSA_PARAMETER_HEADER DSA(디지털 서명 알고리즘) 키에 대한 매개 변수 헤더 정보를 포함하는 데 사용됩니다.
BCRYPT_DSA_PARAMETER_HEADER_V2 DSA(디지털 서명 알고리즘) 키에 대한 매개 변수 헤더 정보를 포함합니다.
BCRYPT_ECCKEY_BLOB 메모리의 타원 곡선 공개 키 또는 프라이빗 키 BLOB에 대한 헤더로 사용됩니다.
BCRYPT_INTERFACE_VERSION CNG 공급자의 프로그래밍 인터페이스에 대한 버전 정보를 포함합니다.
BCRYPT_KEY_BLOB

모든 CNG 키 BLOB의 기본 구조입니다.
BCRYPT_KEY_DATA_BLOB_HEADER 키 데이터 BLOB에 대한 정보를 포함하는 데 사용됩니다.
BCRYPT_KEY_LENGTHS_STRUCT 공급자가 지원하는 키 크기의 범위를 정의합니다.
BCRYPT_MULTI_HASH_OPERATION BCRYPT_MULTI_HASH_OPERATION 구조체는 다중 해시 작업에서 단일 작업을 정의합니다.
BCRYPT_MULTI_OBJECT_LENGTH_STRUCT BCRYPT_MULTI_OBJECT_LENGTH_STRUCT 구조체에는 BCryptCreateMultiHash 함수에 대한 pbHashObject 버퍼의 크기를 결정하는 정보가 포함되어 있습니다.
BCRYPT_OAEP_PADDING_INFO OAEP(최적 비대칭 암호화 패딩) 체계에 대한 옵션을 제공하는 데 사용됩니다.
BCRYPT_OID DER로 인코딩된 OID(개체 식별자)에 대한 정보를 포함합니다.
BCRYPT_OID_LIST BCRYPT_OID 구조체의 컬렉션을 포함하는 데 사용됩니다. 이 구조체를 BCRYPT_HASH_OID_LIST 속성과 함께 사용하여 DER(고유 인코딩 규칙) 인코딩을 사용하여 인코딩된 해시 개체 식별자(OID) 목록을 검색합니다.
BCRYPT_PKCS1_PADDING_INFO PKCS에 대한 옵션을 제공하는 데 사용됩니다.
BCRYPT_PROVIDER_NAME CNG 공급자의 이름을 포함합니다.
BCRYPT_PSS_PADDING_INFO PSS(Probabilistic Signature Scheme) 패딩 스키마에 대한 옵션을 제공하는 데 사용됩니다.
BCRYPT_RSAKEY_BLOB 메모리의 RSA 공개 키 또는 프라이빗 키 BLOB에 대한 헤더로 사용됩니다.

BCryptBuffer BCryptBuffer 구조가 제네릭 암호화 API: CNG(차세대) 버퍼를 나타내는 방법을 설명합니다.
BCryptBufferDesc BCryptBufferDesc 구조에 제네릭 암호화 API: CNG(차세대) 버퍼 집합이 포함된 방법을 설명합니다.
BLOBHEADER 키의 BLOB 형식 및 키가 사용하는 알고리즘을 나타냅니다.
CATALOG_INFO CATALOG_INFO 구조체에는 카탈로그 파일의 이름이 포함됩니다. 이 구조체는 CryptCATCatalogInfoFromContext 함수에서 사용됩니다.
CENTRAL_ACCESS_POLICY 중앙 액세스 정책 항목 집합을 포함하는 중앙 액세스 정책을 나타냅니다.
CENTRAL_ACCESS_POLICY_ENTRY 보안 설명자 및 스테이징된 보안 설명자 목록을 포함하는 중앙 액세스 정책 항목을 나타냅니다.
CERT_ACCESS_DESCRIPTION CERT_ACCESS_DESCRIPTION 구조체는 CERT_AUTHORITY_INFO_ACCESS 구조체의 멤버입니다.
CERT_ALT_NAME_ENTRY 다양한 이름 형식 중 하나에 대체 이름을 포함합니다.
CERT_ALT_NAME_INFO CERT_ALT_NAME_INFO 구조는 주체 또는 발급자 인증서, CRL(인증서 해지 목록) 및 CTL(인증서 신뢰 목록)에 대한 확장을 인코딩 및 디코딩하는 데 사용됩니다.
CERT_AUTHORITY_INFO_ACCESS 기관 정보 액세스 및 주체 정보 액세스 인증서 확장을 나타내며 주체 또는 인증서 발급자의 추가 정보 및 서비스에 액세스하는 방법을 지정합니다.
CERT_AUTHORITY_KEY_ID_INFO 인증서 또는 CRL(인증서 해지 목록)에 서명하는 데 사용되는 키를 식별합니다.
CERT_AUTHORITY_KEY_ID2_INFO

CERT_AUTHORITY_KEY_ID2_INFO 구조는 인증서 또는 CRL에 서명하는 데 사용되는 키를 식별합니다.
CERT_BASIC_CONSTRAINTS_INFO CERT_BASIC_CONSTRAINTS_INFO 구조에는 인증된 주체가 CA(인증 기관), 최종 엔터티 또는 둘 다로 작동할 수 있는지 여부를 나타내는 정보가 포함됩니다.
CERT_BASIC_CONSTRAINTS2_INFO CERT_BASIC_CONSTRAINTS2_INFO 구조에는 인증된 주체가 CA 또는 최종 엔터티로 작동할 수 있는지 여부를 나타내는 정보가 포함되어 있습니다. 주체가 CA 역할을 할 수 있는 경우 인증 경로 길이 제약 조건을 지정할 수도 있습니다.
CERT_BIOMETRIC_DATA 생체 인식 데이터에 대한 정보를 포함합니다.
CERT_BIOMETRIC_EXT_INFO 생체 인식 정보 집합을 포함합니다.
CERT_CHAIN_CONTEXT 연결된 모든 단순 체인에 대한 요약 유효성 데이터를 나타내는 단순 인증서 체인의 배열과 신뢰 상태 구조를 포함합니다.
CERT_CHAIN_ELEMENT CERT_CHAIN_ELEMENT 구조는 단순 인증서 체인의 단일 요소입니다.
CERT_CHAIN_ENGINE_CONFIG 기본이 아닌 인증서 체인 엔진을 빌드하기 위한 매개 변수를 설정합니다. 사용된 엔진은 인증서 체인을 빌드하는 방법을 결정합니다.
CERT_CHAIN_FIND_ISSUER_PARA 인증서 체인을 빌드하기 위해 CertFindChainInStore 함수에 사용되는 정보를 포함합니다.
CERT_CHAIN_PARA CERT_CHAIN_PARA 구조는 인증서 체인을 빌드하는 데 사용할 검색 및 일치 조건을 설정합니다.
CERT_CHAIN_POLICY_PARA 인증서 체인 확인에 대한 정책 기준을 설정하기 위해 CertVerifyCertificateChainPolicy에 사용되는 정보를 포함합니다.
CERT_CHAIN_POLICY_STATUS

인증서 체인의 유효성을 검사할 때 CertVerifyCertificateChainPolicy 함수에서 반환된 인증서 체인 상태 정보를 보유합니다.
CERT_CONTEXT 인증서의 인코딩된 표현과 디코딩된 표현을 모두 포함합니다.
CERT_CREATE_CONTEXT_PARA CertCreateContext 함수를 호출할 때 사용할 수 있는 추가 값을 정의합니다.
CERT_CREDENTIAL_INFO CERT_CREDENTIAL_INFO 구조에는 인증서에 대한 참조가 포함됩니다.
CERT_CRL_CONTEXT_PAIR CERT_CRL_CONTEXT_PAIR 구조에는 인증서 컨텍스트 및 연결된 CRL 컨텍스트가 포함됩니다.
CERT_DH_PARAMETERS Diffie/Hellman 공개 키 알고리즘과 연결된 매개 변수를 포함합니다.
CERT_DSS_PARAMETERS DSS(디지털 서명 표준) 공개 키 알고리즘과 연결된 매개 변수를 포함합니다.
CERT_ECC_SIGNATURE ECDSA(타원 곡선 디지털 서명 알고리즘) 서명에 대한 r 및 s 값을 포함합니다.
CERT_EXTENSION CERT_EXTENSION 구조에는 인증서, CRL(인증서 해지 목록) 또는 CTL(인증서 신뢰 목록)에 대한 확장 정보가 포함됩니다.
CERT_EXTENSIONS CERT_EXTENSIONS 구조체에는 확장 배열이 포함되어 있습니다.
CERT_GENERAL_SUBTREE CERT_GENERAL_SUBTREE 구조체는 CERT_NAME_CONSTRAINTS_INFO 구조체에 사용됩니다. 이 구조는 포함하거나 제외할 수 있는 인증서의 ID를 제공합니다.
CERT_HASHED_URL 해시된 URL을 포함합니다.

CERT_ID 인증서를 고유하게 식별하는 유연한 수단으로 사용됩니다.
CERT_INFO 인증서의 정보를 포함합니다.
CERT_ISSUER_SERIAL_NUMBER 인증서에 대한 발급자 및 발급자의 일련 번호를 포함하는 인증서의 고유 식별자 역할을 합니다.
CERT_KEY_ATTRIBUTES_INFO CERT_KEY_ATTRIBUTES_INFO 구조에는 인증되는 공개 키에 대한 선택적 추가 정보가 포함됩니다.
CERT_KEY_CONTEXT CERT_KEY_CONTEXT_PROP_ID 속성과 연결된 데이터를 포함합니다.
CERT_KEY_USAGE_RESTRICTION_INFO CERT_KEY_USAGE_RESTRICTION_INFO 구조에는 인증서의 공개 키 사용에 적용되는 제한이 포함되어 있습니다. 여기에는 키를 사용할 수 있는 키 및 정책을 사용하기 위한 목적이 포함됩니다.
CERT_KEYGEN_REQUEST_INFO Netscape 키 생성 요청에 저장된 정보를 포함합니다. 주체 및 주체 공개 키 BLOB이 인코딩됩니다.
CERT_LDAP_STORE_OPENED_PARA CERT_STORE_PROV_LDAP 공급자가 CERT_LDAP_STORE_OPENED_FLAG 플래그를 사용하여 쿼리를 수행하는 데 사용할 기존 LDAP 세션과 LDAP 쿼리 문자열을 모두 지정하여 certOpenStore 함수와 함께 사용됩니다.
CERT_LOGOTYPE_AUDIO 오디오 로고 형식에 대한 정보를 포함합니다.
CERT_LOGOTYPE_AUDIO_INFO 오디오 로고 형식에 대한 자세한 정보를 포함합니다.
CERT_LOGOTYPE_DATA 로고 형식 데이터를 포함합니다.
CERT_LOGOTYPE_DETAILS

로고 형식에 대한 추가 정보를 포함합니다.
CERT_LOGOTYPE_EXT_INFO 로고 형식 정보 집합을 포함합니다.
CERT_LOGOTYPE_IMAGE 이미지 로고 형식에 대한 정보를 포함합니다.
CERT_LOGOTYPE_IMAGE_INFO 이미지 로고 형식에 대한 자세한 정보를 포함합니다.
CERT_LOGOTYPE_INFO 로고 형식 데이터에 대한 정보를 포함합니다.
CERT_LOGOTYPE_REFERENCE 로고 형식 참조 정보를 포함합니다.
CERT_NAME_CONSTRAINTS_INFO CERT_NAME_CONSTRAINTS_INFO 구조에는 특히 허용되거나 신뢰에서 제외되는 인증서에 대한 정보가 포함됩니다.
CERT_NAME_INFO 주체 또는 발급자 이름을 포함합니다.
CERT_NAME_VALUE RDN(상대 고유 이름) 특성 값을 포함합니다.
CERT_OR_CRL_BLOB 인터넷 키 Exchange 메시지와 함께 사용할 인증서를 캡슐화합니다.
CERT_OR_CRL_BUNDLE 인터넷 키 Exchange 메시지와 함께 사용할 인증서 배열을 캡슐화합니다.
CERT_OTHER_LOGOTYPE_INFO 미리 정의되지 않은 로고 유형에 대한 정보를 포함합니다.
CERT_PAIR CERT_PAIR 구조에는 인증서와 해당 쌍 교차 인증서가 포함됩니다.

CERT_PHYSICAL_STORE_INFO 물리적 인증서 저장소에 대한 정보를 포함합니다.
CERT_POLICIES_INFO CERT_POLICIES_INFO 구조체에는 CERT_POLICY_INFO 배열이 포함되어 있습니다.
CERT_POLICY_CONSTRAINTS_INFO CERT_POLICY_CONSTRAINTS_INFO 구조에는 인증서를 신뢰할 수 있는 것으로 수락하기 위한 설정된 정책이 포함되어 있습니다.
CERT_POLICY_ID CERT_POLICY_ID 구조에는 인증서가 명시적으로 지원하는 인증서 정책 목록과 이러한 정책과 관련된 선택적 한정자 정보가 포함되어 있습니다.
CERT_POLICY_INFO CERT_POLICY_INFO 구조에는 정책 및 정책 한정자의 선택적 배열을 지정하는 OID(개체 식별자)가 포함됩니다.
CERT_POLICY_MAPPING 발급자 도메인과 주체 도메인 정책 OID 간의 매핑을 포함합니다.
CERT_POLICY_MAPPINGS_INFO CERT_POLICY_MAPPINGS_INFO 구조는 두 도메인의 정책 OID 간에 매핑을 제공합니다.
CERT_POLICY_QUALIFIER_INFO CERT_POLICY_QUALIFIER_INFO 구조체에는 한정자 및 한정자별 추가 정보를 지정하는 OID(개체 식별자)가 포함됩니다.
CERT_PRIVATE_KEY_VALIDITY CERT_PRIVATE_KEY_VALIDITY 구조는 인증서의 공개 키에 해당하는 프라이빗 키에 대한 유효한 시간 범위를 나타냅니다.
CERT_PUBLIC_KEY_INFO 공개 키 및 해당 알고리즘을 포함합니다.
CERT_QC_STATEMENT QC(정규화된 인증서) 문 확장에 포함할 하나 이상의 문 시퀀스의 단일 문을 나타냅니다.

CERT_QC_STATEMENTS_EXT_INFO QC에 대한 정규화된 인증서(QC) 문 확장을 구성하는 하나 이상의 문 시퀀스를 포함합니다.
CERT_RDN CERT_RDN 구조체에는 CERT_RDN_ATTR 구조의 배열로 구성된 RDN(상대 고유 이름)이 포함됩니다.
CERT_RDN_ATTR RDN(상대 고유 이름)의 단일 특성을 포함합니다. 전체 RDN은 CERT_RDN_ATTR 구조의 배열을 포함하는 CERT_RDN 구조체로 표현됩니다.
CERT_REQUEST_INFO CERT_REQUEST_INFO 구조에는 인증서 요청에 대한 정보가 포함됩니다. 제목, 주체 공개 키 및 특성 BLOB이 인코딩됩니다.
CERT_REVOCATION_CHAIN_PARA 독립적인 OCSP(온라인 인증서 상태 프로토콜) 응답 서명자 인증서에 대한 체인을 빌드하는 데 사용되는 매개 변수를 포함합니다.
CERT_REVOCATION_CRL_INFO CRL(인증서 해지 목록) 해지 유형 처리기에서 업데이트된 정보를 포함합니다.
CERT_REVOCATION_INFO CERT_CHAIN_ELEMENT 인증서의 해지 상태를 나타냅니다.
CERT_REVOCATION_PARA 확인할 컨텍스트의 발급자를 찾는 데 도움이 되도록 CertVerifyRevocation 함수에 대한 호출에서 전달됩니다.
CERT_REVOCATION_STATUS 인증서의 해지 상태에 대한 정보를 포함합니다.
CERT_SELECT_CHAIN_PARA 체인을 빌드하고 선택하는 데 사용되는 매개 변수를 포함합니다.
CERT_SELECT_CRITERIA CertSelectCertificateChains 함수에 전달되는 선택 조건을 지정합니다.

CERT_SELECT_STRUCT_A 인증서 선택 대화 상자에 표시되는 인증서를 선택할 조건을 포함합니다. 이 구조체는 CertSelectCertificate 함수에 사용됩니다. (ANSI)
CERT_SELECT_STRUCT_W 인증서 선택 대화 상자에 표시되는 인증서를 선택할 조건을 포함합니다. 이 구조체는 CertSelectCertificate 함수에 사용됩니다. (유니코드)
CERT_SELECTUI_INPUT CertSelectionGetSerializedBlob 함수에서 저장소 또는 인증서 체인 배열에 포함된 인증서를 직렬화하는 데 사용됩니다. 반환된 직렬화된 BLOB을 CredUIPromptForWindowsCredentials 함수에 전달할 수 있습니다.
CERT_SERVER_OCSP_RESPONSE_CONTEXT 인코딩된 OCSP 응답을 포함합니다.
CERT_SIGNED_CONTENT_INFO CERT_SIGNED_CONTENT_INFO 구조에는 서명할 인코딩된 콘텐츠와 서명을 저장할 BLOB이 포함됩니다. ToBeSigned 멤버는 인코딩된 CERT_INFO, CRL_INFO, CTL_INFO 또는 CERT_REQUEST_INFO.
CERT_SIMPLE_CHAIN CERT_SIMPLE_CHAIN 구조체에는 체인 요소의 배열과 배열이 나타내는 체인에 대한 요약 신뢰 상태가 포함됩니다.
CERT_STORE_PROV_FIND_INFO 많은 저장소 공급자 콜백 함수에서 사용됩니다.
CERT_STORE_PROV_INFO CertOpenStore 함수를 사용하여 저장소를 열 때 설치된 CertDllOpenStoreProv 함수에서 반환하는 정보를 포함합니다.
CERT_STRONG_SIGN_PARA 인증서, CRL(인증서 해지 목록), OCSP(온라인 인증서 상태 프로토콜) 응답 및 PKCS에서 강력한 서명을 확인하는 데 사용되는 매개 변수를 포함합니다.
CERT_STRONG_SIGN_SERIALIZED_INFO 강력한 서명에 사용할 수 있는 서명 알고리즘/해시 알고리즘 및 공개 키 알고리즘/비트 길이 쌍을 포함합니다.

CERT_SYSTEM_STORE_INFO CERT_SYSTEM_STORE_INFO 구조에는 시스템 저장소에서 작동하는 함수에서 사용하는 정보가 포함됩니다. 현재 이 구조에는 필수 정보가 포함되어 있지 않습니다.
CERT_SYSTEM_STORE_RELOCATE_PARA CERT_SYSTEM_STORE_RELOCATE_PARA 구조체에는 해당 함수의 dwFlags 매개 변수가 CERT_SYSTEM_STORE_RELOCATE_FLAG 설정된 경우 CertOpenStore에 전달될 데이터가 포함됩니다.
CERT_TEMPLATE_EXT 인증서 템플릿입니다.
CERT_TRUST_LIST_INFO CTL의 유효한 사용을 나타내는 CERT_TRUST_LIST_INFO 구조체입니다.
CERT_TRUST_STATUS 인증서 체인의 인증서에 대한 신뢰 정보, 간단한 인증서 체인에 대한 요약 신뢰 정보 또는 단순 체인 배열에 대한 요약 정보를 포함합니다.
CERT_USAGE_MATCH 인증서 체인을 빌드하는 데 사용할 발급자 인증서를 식별하기 위한 조건을 제공합니다.
CERT_VIEWPROPERTIES_STRUCT_A CERT_VIEWPROPERTIES_STRUCT 구조는 인증서의 속성을 표시하기 위해 CertViewProperties 함수를 호출할 때 사용되는 정보를 정의합니다. (ANSI)
CERT_VIEWPROPERTIES_STRUCT_W CERT_VIEWPROPERTIES_STRUCT 구조는 인증서의 속성을 표시하기 위해 CertViewProperties 함수를 호출할 때 사용되는 정보를 정의합니다. (유니코드)
CERT_X942_DH_PARAMETERS Diffie-Hellman 공개 키 알고리즘과 연결된 매개 변수를 포함합니다.
CERT_X942_DH_VALIDATION_PARAMS 필요에 따라 CERT_X942_DH_PARAMETERS 구조체의 멤버가 가리키고 추가 시드 정보를 포함합니다.
CLAIM_SECURITY_ATTRIBUTE_FQBN_VALUE 정규화된 이진 이름을 지정합니다.

CLAIM_SECURITY_ATTRIBUTE_OCTET_STRING_VALUE 클레임 보안 특성의 OCTET_STRING 값 형식을 지정합니다.
CLAIM_SECURITY_ATTRIBUTE_RELATIVE_V1 serialize된 보안 설명자 내에서 지속성을 위해 연속 메모리에 정의된 리소스 특성을 정의합니다.
CLAIM_SECURITY_ATTRIBUTE_V1 토큰 또는 권한 부여 컨텍스트와 연결할 수 있는 보안 특성을 정의합니다.
CLAIM_SECURITY_ATTRIBUTES_INFORMATION 클레임의 보안 특성을 정의합니다.
CMC_ADD_ATTRIBUTES_INFO 인증서에 추가할 인증서 특성을 포함합니다.
CMC_ADD_EXTENSIONS_INFO 인증서에 추가할 인증서 확장 제어 특성을 포함합니다.
CMC_DATA_INFO 태그가 지정된 다양한 정보를 전달하는 방법을 제공합니다. (CMC_DATA_INFO)
CMC_PEND_INFO CMC_STATUS_INFO 구조체의 가능한 멤버입니다.
CMC_RESPONSE_INFO 태그가 지정된 다양한 정보를 전달하는 방법을 제공합니다. (CMC_RESPONSE_INFO)
CMC_STATUS_INFO CMS를 통해 인증서 관리 메시지에 대한 상태 정보를 포함합니다.
CMC_TAGGED_ATTRIBUTE CMC_DATA_INFO 및 CMC_RESPONSE_INFO 구조에 사용됩니다. (CMC_TAGGED_ATTRIBUTE)
CMC_TAGGED_CERT_REQUEST CMC_TAGGED_REQUEST 구조체에 사용됩니다.
CMC_TAGGED_CONTENT_INFO

CMC_DATA_INFO 및 CMC_RESPONSE_INFO 구조에 사용됩니다. (CMC_TAGGED_CONTENT_INFO)
CMC_TAGGED_OTHER_MSG CMC_DATA_INFO 및 CMC_RESPONSE_INFO 구조에 사용됩니다. (CMC_TAGGED_OTHER_MSG)
CMC_TAGGED_REQUEST CMC_DATA_INFO 구조에서 인증서를 요청하는 데 사용됩니다.
CMS_DH_KEY_INFO Diffie-Hellman 키 정보를 포함하기 위해 CryptSetKeyParam 함수의 KP_CMS_DH_KEY_INFO 매개 변수와 함께 사용됩니다.
CMS_KEY_INFO 사용되지 않습니다.
CMSG_CMS_RECIPIENT_INFO CryptMsgGetParam 함수와 함께 키 전송, 키 계약 또는 메일 목록 봉투 메시지 받는 사람에 대한 정보를 가져오는 데 사용됩니다.
CMSG_CMS_SIGNER_INFO 서명되거나 서명되거나 봉투에 포함된 메시지에 정의된 SignerInfo의 콘텐츠를 포함합니다.
CMSG_CNG_CONTENT_DECRYPT_INFO CEK(Cryptography API:_Next Generation) CEK(콘텐츠 암호화 키)의 가져오기 및 암호 해독을 위해 CryptMsgControl과 OID(개체 식별자) 설치 가능 함수 간에 전달된 모든 관련 정보를 포함합니다.
CMSG_CONTENT_ENCRYPT_INFO PFN_CMSG_GEN_CONTENT_ENCRYPT_KEY, PFN_CMSG_EXPORT_KEY_TRANS, PFN_CMSG_EXPORT_KEY_AGREE 및 PFN_CMSG_EXPORT_MAIL_LIST 함수 간에 공유되는 정보를 포함합니다.
CMSG_CTRL_ADD_SIGNER_UNAUTH_ATTR_PARA 서명된 메시지의 서명자에 인증되지 않은 특성을 추가하는 데 사용됩니다.
CMSG_CTRL_DECRYPT_PARA 키 전송 받는 사람에 대한 봉투 메시지의 암호를 해독하는 데 사용되는 정보를 포함합니다. dwCtrlType 매개 변수가 CMSG_CTRL_DECRYPT 경우 이 구조체가 CryptMsgControl에 전달됩니다.
CMSG_CTRL_DEL_SIGNER_UNAUTH_ATTR_PARA

서명된 메시지 서명자의 인증되지 않은 특성을 삭제하는 데 사용됩니다.
CMSG_CTRL_KEY_AGREE_DECRYPT_PARA 주요 계약 받는 사람에 대한 정보를 포함합니다.
CMSG_CTRL_KEY_TRANS_DECRYPT_PARA 키 전송 메시지 받는 사람에 대한 정보를 포함합니다.
CMSG_CTRL_MAIL_LIST_DECRYPT_PARA 메일 목록 메시지 받는 사람에 대한 정보를 포함합니다.
CMSG_CTRL_VERIFY_SIGNATURE_EX_PARA 메시지 서명을 확인하는 데 사용되는 정보를 포함합니다. 서명자 인덱스 및 서명자 공개 키가 포함됩니다.
CMSG_ENVELOPED_ENCODE_INFO 봉투 메시지를 인코딩하는 데 필요한 정보를 포함합니다. dwMsgType 매개 변수가 CMSG_ENVELOPED 경우 CryptMsgOpenToEncode에 전달됩니다.
CMSG_HASHED_ENCODE_INFO 해시된 메시지와 함께 사용됩니다. CryptMsgOpenToEncode 함수의 dwMsgType 매개 변수가 CMSG_ENVELOPED 경우 CryptMsgOpenToEncode 함수에 전달됩니다.
CMSG_KEY_AGREE_ENCRYPT_INFO 봉투형 메시지의 모든 주요 계약 수신자에게 적용할 수 있는 암호화 정보를 포함합니다.
CMSG_KEY_AGREE_KEY_ENCRYPT_INFO 봉투형 메시지의 키 계약 수신자에 대한 암호화된 키를 포함합니다.
CMSG_KEY_AGREE_RECIPIENT_ENCODE_INFO 키 계약 키 관리를 사용하는 메시지 받는 사람에 대한 정보를 포함합니다.
CMSG_KEY_AGREE_RECIPIENT_INFO 주요 규약 알고리즘에 사용되는 정보를 포함합니다.
CMSG_KEY_TRANS_ENCRYPT_INFO 봉투형 데이터의 키 전송 받는 사람에 대한 암호화 정보를 포함합니다.
CMSG_KEY_TRANS_RECIPIENT_ENCODE_INFO

메시지 받는 사람에 대한 인코딩된 키 전송 정보를 포함합니다.
CMSG_KEY_TRANS_RECIPIENT_INFO CMSG_KEY_TRANS_RECIPIENT_INFO 구조에는 키 전송 알고리즘에 사용되는 정보가 포함되어 있습니다.
CMSG_MAIL_LIST_ENCRYPT_INFO 봉투형 데이터의 메일 그룹 받는 사람에 대한 암호화 정보를 포함합니다.
CMSG_MAIL_LIST_RECIPIENT_ENCODE_INFO CMSG_MAIL_LIST_RECIPIENT_ENCODE_INFO 구조는 KEK(콘텐츠 키 암호화 키)의 암호를 해독하기 위해 이전에 분산된 대칭 키와 함께 사용됩니다.
CMSG_MAIL_LIST_RECIPIENT_INFO 이전에 분산된 KEK(키 암호화 키)에 사용되는 정보를 포함합니다.
CMSG_RC2_AUX_INFO RC2 암호화 알고리즘에 대한 키의 비트 길이를 포함합니다.
CMSG_RC4_AUX_INFO CMSG_RC4_AUX_INFO 구조에는 RC4 암호화 알고리즘에 대한 키의 비트 길이가 포함됩니다. CMSG_ENVELOPED_ENCODE_INFO pvEncryptionAuxInfo 멤버는 이 구조체의 인스턴스를 가리키도록 설정할 수 있습니다.
CMSG_RECIPIENT_ENCODE_INFO 메시지 받는 사람의 콘텐츠 암호화 키 관리 유형에 대한 정보를 포함합니다.
CMSG_RECIPIENT_ENCRYPTED_KEY_ENCODE_INFO 메시지 내용의 암호를 해독하는 데 필요한 세션 키를 해독하는 데 사용되는 메시지 수신기에 대한 정보를 포함합니다.
CMSG_RECIPIENT_ENCRYPTED_KEY_INFO CMSG_RECIPIENT_ENCRYPTED_KEY_INFO 구조에는 개별 키 계약 수신자에 사용되는 정보가 포함됩니다.
CMSG_SIGNED_ENCODE_INFO dwMsgType이 CMSG_SIGNED 경우 CryptMsgOpenToEncode에 전달할 정보를 포함합니다.
CMSG_SIGNER_ENCODE_INFO 서명자 정보를 포함합니다. dwMsgType 매개 변수가 CMSG_SIGNED 경우 CryptMsgCountersign,

CryptMsgCountersignEncoded에 전달되고, 필요에 따라 CMSG_SIGNED_ENCODE_INFO 구조체의 멤버로 CryptMsgOpenToEncode에 전달됩니다.
CMSG_SIGNER_INFO CMSG_SIGNER_INFO 구조에는 PKCS의 콘텐츠가 포함됩니다.
CMSG_SP3_COMPATIBLE_AUX_INFO SP3 호환 암호화에 필요한 정보를 포함합니다.
CMSG_STREAM_INFO 단일 블록 처리가 아닌 데이터의 스트림 처리를 사용하도록 설정하는 데 사용됩니다.
CREDENTIAL_ATTRIBUTEA CREDENTIAL_ATTRIBUTE 구조에는 자격 증명의 애플리케이션 정의 특성이 포함됩니다. 특성은 키워드-값 쌍입니다. 특성의 의미를 정의하는 것은 애플리케이션에 달려 있습니다. (ANSI)
CREDENTIAL_ATTRIBUTEW CREDENTIAL_ATTRIBUTE 구조에는 자격 증명의 애플리케이션 정의 특성이 포함됩니다. 특성은 키워드-값 쌍입니다. 특성의 의미를 정의하는 것은 애플리케이션에 달려 있습니다. (유니코드)
CREDENTIAL_TARGET_INFORMATIONA CREDENTIAL_TARGET_INFORMATION 구조에는 대상 컴퓨터의 이름, 도메인 및 트리가 포함됩니다. (ANSI)
CREDENTIAL_TARGET_INFORMATIONW CREDENTIAL_TARGET_INFORMATION 구조에는 대상 컴퓨터의 이름, 도메인 및 트리가 포함됩니다. (유니코드)
크레덴티알라 CREDENTIAL 구조에는 개별 자격 증명이 포함됩니다. (ANSI)
CREDENTIALW CREDENTIAL 구조에는 개별 자격 증명이 포함됩니다. (유니코드)
CREDSSP_CRED Schannel 및 Negotiate 보안 패키지에 대한 인증 데이터를 지정합니다.
CREDUI_INFOA

CREDUI_INFO 구조체는 자격 증명 정보를 가져오는 데 사용되는 대화 상자를 만드는 CredUIPromptForCredentials 함수에 정보를 전달하는 데 사용됩니다. (ANSI)
CREDUI_INFOW CREDUI_INFO 구조체는 자격 증명 정보를 가져오는 데 사용되는 대화 상자를 만드는 CredUIPromptForCredentials 함수에 정보를 전달하는 데 사용됩니다. (유니코드)
CREDUIWIN_MARSHALED_CONTEXT ICredentialProvider::SetSerialization 메서드를 사용하여 serialize된 자격 증명 정보를 지정합니다.
CRL_CONTEXT CRL_CONTEXT 구조에는 CRL(인증서 해지 목록)의 인코딩된 표현과 디코딩된 표현이 모두 포함됩니다. CryptoAPI 함수에서 반환된 CRL 컨텍스트는 CertFreeCRLContext 함수를 호출하여 해제해야 합니다.
CRL_DIST_POINT 인증서 사용자가 인증서 해지 여부를 확인하기 위해 참조할 수 있는 단일 CRL(인증서 해지 목록) 배포 지점을 식별합니다.
CRL_DIST_POINT_NAME CRL을 가져올 수 있는 위치를 식별합니다.
CRL_DIST_POINTS_INFO 인증서 사용자가 인증서 해지 여부를 확인하기 위해 참조할 수 있는 CRL(인증서 해지 목록) 배포 지점 목록을 포함합니다.
CRL_ENTRY 해지된 단일 인증서에 대한 정보를 포함합니다. CRL_INFO 구조체의 멤버입니다.
CRL_FIND_ISSUED_FOR_PARA 주체 및 인증서 발급자의 인증서 컨텍스트를 포함합니다.
CRL_INFO CRL(인증서 해지 목록)의 정보를 포함합니다.
CRL_ISSUING_DIST_POINT CRL(인증서 해지 목록)에 나열된 인증서 종류에 대한 정보를 포함합니다.
CROSS_CERT_DIST_POINTS_INFO

동적 교차 인증서를 업데이트하는 데 사용되는 정보를 제공합니다.
CRYPT_AES_128_KEY_STATE AES(Advanced Encryption Standard) 암호화에 대한 128비트 대칭 키 정보를 지정합니다.
CRYPT_AES_256_KEY_STATE AES(Advanced Encryption Standard) 암호에 대한 256비트 대칭 키 정보를 지정합니다.
CRYPT_ALGORITHM_IDENTIFIER 프라이빗 키를 암호화하는 데 사용되는 알고리즘을 지정합니다.
CRYPT_ATTRIBUTE CRYPT_ATTRIBUTE 구조체는 하나 이상의 값을 포함하는 특성을 지정합니다.
CRYPT_ATTRIBUTE_TYPE_VALUE 단일 특성 값을 포함합니다. Value 멤버의 CRYPT_OBJID_BLOB 인코딩됩니다.
CRYPT_ATTRIBUTES 특성 배열을 포함합니다.
CRYPT_BIT_BLOB 바이트 배열로 표시되는 비트 집합을 포함합니다.
CRYPT_BLOB_ARRAY CRYPT_DATA_BLOB 구조체의 배열을 포함합니다.
CRYPT_CONTENT_INFO PKCS에 인코딩된 데이터를 포함합니다.
CRYPT_CONTENT_INFO_SEQUENCE_OF_ANY 인증서의 Netscape 인증서 시퀀스를 나타내는 정보를 포함합니다.
CRYPT_CONTEXT_CONFIG CNG 컨텍스트에 대한 구성 정보를 포함합니다.
CRYPT_CONTEXT_FUNCTION_CONFIG CNG 컨텍스트의 암호화 함수에 대한 구성 정보를 포함합니다.

CRYPT_CONTEXT_FUNCTION_PROVIDERS CNG 구성 컨텍스트에 대한 암호화 함수 공급자 집합을 포함합니다.
CRYPT_CONTEXT_FUNCTIONS CNG 구성 컨텍스트에 대한 암호화 함수 집합을 포함합니다.
CRYPT_CONTEXTS CNG 구성 컨텍스트 식별자 집합을 포함합니다.
CRYPT_CREDENTIALS CryptRetrieveObjectByUrl 또는 CryptGetTimeValidObject와 같은 원격 개체 검색 함수에 선택적 입력으로 전달할 수 있는 자격 증명에 대한 정보를 포함합니다.
CRYPT_DECODE_PARA CryptDecodeObjectEx 함수에서 메모리 할당 및 메모리 해제 콜백 함수에 대한 액세스를 제공하는 데 사용됩니다.
CRYPT_DECRYPT_MESSAGE_PARA CRYPT_DECRYPT_MESSAGE_PARA 구조에는 메시지 암호 해독에 대한 정보가 포함되어 있습니다.
CRYPT_DEFAULT_CONTEXT_MULTI_OID_PARA CryptInstallDefaultContext 함수와 함께 개체 식별자 문자열의 배열을 포함하는 데 사용됩니다.
CRYPT_ECC_CMS_SHARED_INFO CMS(암호화 메시지 구문) EnvelopedData 콘텐츠 형식에서 ECC(타원 곡선 암호화)를 사용할 때의 키 암호화 키 정보를 나타냅니다.
CRYPT_ENCODE_PARA CryptEncodeObjectEx 함수에서 메모리 할당 및 메모리 해제 콜백 함수에 대한 액세스를 제공하는 데 사용됩니다.
CRYPT_ENCRYPT_MESSAGE_PARA 메시지를 암호화하는 데 사용되는 정보를 포함합니다.
CRYPT_ENCRYPTED_PRIVATE_KEY_INFO PKCS의 정보를 포함합니다.

CRYPT_ENROLLMENT_NAME_VALUE_PAIR 사용자를 대신하여 인증서 요청을 만드는 데 사용됩니다.
CRYPT_GET_TIME_VALID_OBJECT_EXTRA_INFO pExtraInfo 매개 변수의 CryptGetTimeValidObject 함수에 전달할 수 있는 선택적 추가 정보를 포함합니다.
CRYPT_HASH_MESSAGE_PARA 해시 메시지에 대한 데이터를 포함합니다.
CRYPT_IMAGE_REF CNG 공급자 모듈에 대한 정보를 포함합니다.
CRYPT_IMAGE_REG CNG 공급자에 대한 이미지 등록 정보를 포함합니다.
CRYPT_INTEGER_BLOB CryptoAPI CRYPT_INTEGER_BLOB 구조체(wincrypt.h)는 임의의 바이트 배열에 사용되며 데이터 형식을 포함할 수 있는 개체에 유연성을 제공합니다.
CRYPT_INTEGER_BLOB CryptoAPI CRYPT_INTEGER_BLOB(dpapi.h) 구조체는 임의의 바이트 배열에 사용됩니다. Wincrypt.h에서 선언되며 다양한 데이터 형식을 포함할 수 있는 개체에 유연성을 제공합니다. (CRYPT_INTEGER_BLOB)
CRYPT_INTERFACE_REG CNG 공급자가 지원하는 인터페이스 형식에 대한 정보를 포함하는 데 사용됩니다.
CRYPT_KEY_PROV_INFO CRYPT_KEY_PROV_INFO 구조에는 CSP(암호화 서비스 공급자) 내의 키 컨테이너에 대한 정보가 포함됩니다.
CRYPT_KEY_PROV_PARAM 키 컨테이너 매개 변수에 대한 정보를 포함합니다.
CRYPT_KEY_SIGN_MESSAGE_PARA CSP(암호화 서비스 공급자) 및 메시지에 서명하는 데 사용되는 알고리즘에 대한 정보를 포함합니다.
CRYPT_KEY_VERIFY_MESSAGE_PARA

서명자에 대한 인증서 없이 서명된 메시지를 확인하는 데 필요한 정보를 포함합니다.
CRYPT_MASK_GEN_ALGORITHM RSA PKCS를 생성하는 데 사용되는 알고리즘을 식별합니다.
CRYPT_OBJECT_LOCATOR_PROVIDER_TABLE 개체 위치 공급자가 구현한 함수에 대한 포인터를 포함합니다.
CRYPT_OID_FUNC_ENTRY OID(개체 식별자) 및 관련 함수에 대한 포인터를 포함합니다.
CRYPT_OID_INFO OID(개체 식별자)에 대한 정보를 포함합니다.
CRYPT_PASSWORD_CREDENTIALSA cryptRetrieveObjectByUrl 또는 CryptGetTimeValidObject와 같은 원격 개체 검색 함수에 대한 선택적 입력으로 CRYPT_CREDENTIALS 구조에서 사용할 사용자 이름 및 암호 자격 증명을 포함합니다. (ANSI)
CRYPT_PASSWORD_CREDENTIALSW cryptRetrieveObjectByUrl 또는 CryptGetTimeValidObject와 같은 원격 개체 검색 함수에 대한 선택적 입력으로 CRYPT_CREDENTIALS 구조에서 사용할 사용자 이름 및 암호 자격 증명을 포함합니다. (유니코드)
CRYPT_PKCS12_PBE_PARAMS PKCS에 대한 암호화 키, IV(초기화 벡터) 또는 MAC(메시지 인증 코드) 키를 만드는 데 사용되는 매개 변수를 포함합니다.
CRYPT_PKCS8_EXPORT_PARAMS 프라이빗 키를 암호화하는 프라이빗 키 및 콜백 함수를 식별합니다. CRYPT_PKCS8_EXPORT_PARAMS PKCS에서 프라이빗 키를 내보내는 CryptExportPKCS8Ex 함수에 대한 매개 변수로 사용됩니다.
CRYPT_PKCS8_IMPORT_PARAMS PKCS 포함
CRYPT_PRIVATE_KEY_INFO PrivateKey 필드(DER 인코딩됨)에 지우기 텍스트 프라이빗 키를 포함합니다. CRYPT_PRIVATE_KEY_INFO PKCS의 정보를 포함합니다.
CRYPT_PROPERTY_REF

CNG 컨텍스트 속성에 대한 정보를 포함합니다.
CRYPT_PROVIDER_CERT 공급자 인증서에 대한 정보를 제공합니다.
CRYPT_PROVIDER_DATA WinVerifyTrust와 트러스트 공급자 간에 데이터를 전달하는 데 사용됩니다.
CRYPT_PROVIDER_DEFUSAGE WintrustGetDefaultForUsage 함수에서 공급자의 기본 사용에 대한 콜백 정보를 검색하는 데 사용됩니다.
CRYPT_PROVIDER_FUNCTIONS WinTrust 작업에 대한 CSP(암호화 서비스 공급자)에서 사용하는 함수를 정의합니다.
CRYPT_PROVIDER_PRIVDATA 공급자가 사용할 프라이빗 데이터를 포함합니다.
CRYPT_PROVIDER_REF 공급자가 지원하는 암호화 인터페이스에 대한 정보를 포함합니다.
CRYPT_PROVIDER_REFS 공급자 참조의 컬렉션을 포함합니다.
CRYPT_PROVIDER_REG CNG 공급자에 대한 등록 정보를 포함하는 데 사용됩니다.
CRYPT_PROVIDER_REGDEFUSAGE WintrustAddDefaultForUsage 함수에서 공급자의 기본 사용량에 대한 콜백 정보를 등록하는 데 사용됩니다.
CRYPT_PROVIDER_SGNR 서명자 또는 카운터 서명자에 대한 정보를 제공합니다.
CRYPT_PROVIDER_SIGSTATE 정책 공급자와 Wintrust 간에 통신하는 데 사용됩니다.
CRYPT_PROVIDERS 등록된 CNG 공급자에 대한 정보를 포함합니다.

CRYPT_PROVUI_DATA 공급자에 대한 UI(사용자 인터페이스) 데이터를 제공합니다. 이 구조체는 CRYPT_PROVUI_FUNCS 구조체에서 사용됩니다.
CRYPT_PROVUI_FUNCS 공급자의 UI(사용자 인터페이스) 함수에 대한 정보를 제공합니다. 이 구조체는 CRYPT_PROVIDER_FUNCTIONS 구조체에서 사용됩니다.
CRYPT_PSOURCE_ALGORITHM 알고리즘과 RSAES-OAEP 키 암호화에 대한 레이블의 값을 (선택적으로) 식별합니다.
CRYPT_RC2_CBC_PARAMETERS szOID_RSA_RC2CBC 암호화에 사용되는 정보를 포함합니다.
CRYPT_REGISTER_ACTIONID 공급자의 함수에 대한 정보를 제공합니다.
CRYPT_RETRIEVE_AUX_INFO CryptRetrieveObjectByUrl 함수에 전달할 선택적 정보를 포함합니다.
CRYPT_RSA_SSA_PSS_PARAMETERS RSA PKCS에 대한 매개 변수를 포함합니다.
CRYPT_RSAES_OAEP_PARAMETERS RSAES-OAEP 키 암호화에 대한 매개 변수를 포함합니다.
CRYPT_SEQUENCE_OF_ANY 인코딩된 BLOB의 임의의 목록을 포함합니다.
CRYPT_SIGN_MESSAGE_PARA CRYPT_SIGN_MESSAGE_PARA 구조에는 지정된 서명 인증서 컨텍스트를 사용하여 메시지에 서명하는 정보가 포함됩니다.
CRYPT_SMART_CARD_ROOT_INFO 인증서 컨텍스트와 연결된 스마트 카드 및 세션 ID를 포함합니다.
CRYPT_SMIME_CAPABILITIES

지원되는 기능의 우선 순위가 지정된 배열을 포함합니다.
CRYPT_SMIME_CAPABILITY CRYPT_SMIME_CAPABILITY 구조체는 단일 기능 및 관련 매개 변수를 지정합니다. 단일 기능은 우선 순위가 지정된 기능 기본 설정 목록을 지정할 수 있는 CRYPT_SMIME_CAPABILITIES 목록으로 그룹화됩니다.
CRYPT_TIME_STAMP_REQUEST_INFO 타임스탬핑에 사용됩니다.
CRYPT_TIMESTAMP_ACCURACY CRYPT_TIMESTAMP_INFO 구조체에서 타임스탬프를 만든 UTC 시간 주위의 시간 편차의 정확도를 나타내기 위해 사용됩니다.
CRYPT_TIMESTAMP_CONTEXT 타임스탬프는 인코딩된 표현과 디코딩된 타임스탬프를 모두 포함합니다.
CRYPT_TIMESTAMP_INFO 서명된 데이터 콘텐츠 형식을 CMS(암호화 메시지 구문) 형식으로 포함합니다.
CRYPT_TIMESTAMP_PARA 타임스탬프를 요청하는 추가 매개 변수를 정의합니다.
CRYPT_TIMESTAMP_REQUEST TimeStampReq 형식의 ASN.1(Abstract Syntax Notation One) 정의에 해당하는 타임스탬프 요청 구조를 정의합니다.
CRYPT_TIMESTAMP_RESPONSE 내부적으로 ASN.1(Abstract Syntax Notation One) DER(고유 인코딩 규칙) 인코딩 응답을 캡슐화하는 데 사용됩니다.
CRYPT_TRUST_REG_ENTRY DLL 이름 및 함수 이름으로 공급자 함수를 식별합니다.
CRYPT_URL_INFO URL 그룹화에 대한 정보를 포함합니다.
CRYPT_VERIFY_CERT_SIGN_STRONG_PROPERTIES_INFO 공개 키의 길이(비트)와 강력한 서명에 사용되는 서명 및 해시 알고리즘의 이름을 포함합니다.

CRYPT_VERIFY_MESSAGE_PARA CRYPT_VERIFY_MESSAGE_PARA 구조에는 서명된 메시지를 확인하는 데 필요한 정보가 포함되어 있습니다.
CRYPT_X942_OTHER_INFO CRYPT_X942_OTHER_INFO 구조에는 추가 키 생성 정보가 포함됩니다.
CRYPT_XML_ALGORITHM 메시지에 서명하거나 변환하는 데 사용되는 알고리즘을 지정합니다.
CRYPT_XML_ALGORITHM_INFO 알고리즘 정보를 포함합니다.
CRYPT_XML_BLOB 임의의 바이트 배열을 포함합니다.
CRYPT_XML_CRYPTOGRAPHIC_INTERFACE 구현된 CryptXML 함수를 노출합니다.
CRYPT_XML_DATA_BLOB XML로 인코딩된 데이터를 포함합니다.
CRYPT_XML_DATA_PROVIDER XML 데이터 공급자에 대한 인터페이스를 지정합니다.
CRYPT_XML_DOC_CTXT 문서 컨텍스트 정보를 정의합니다.
CRYPT_XML_ISSUER_SERIAL X.509에서 발급된 고유 이름(일련 번호 쌍)을 포함합니다.
CRYPT_XML_KEY_DSA_KEY_VALUE DSA(디지털 서명 알고리즘) 키 값을 정의합니다. CRYPT_XML_KEY_DSA_KEY_VALUE 구조체는 CRYPT_XML_KEY_VALUE 구조체에서 키 값 공용 구조체의 요소로 사용됩니다.
CRYPT_XML_KEY_ECDSA_KEY_VALUE ECDSA(타원 곡선 디지털 서명 알고리즘) 키 값을 정의합니다. CRYPT_XML_KEY_ECDSA_KEY_VALUE 구조체는 CRYPT_XML_KEY_VALUE 구조체에서 키 값 공용 구조체의 요소로 사용됩니다.

CRYPT_XML_KEY_INFO 키 정보 데이터를 캡슐화합니다.
CRYPT_XML_KEY_INFO_ITEM KeyInfo 요소에 해당하는 키 정보 데이터를 캡슐화합니다. KeyInfo 요소를 사용하면 받는 사람이 서명의 유효성을 검사하는 데 필요한 키를 가져올 수 있습니다.
CRYPT_XML_KEY_RSA_KEY_VALUE RSA 키 값을 정의합니다. CRYPT_XML_KEY_RSA_KEY_VALUE 구조체는 CRYPT_XML_KEY_VALUE 구조체에서 키 값 공용 구조체의 요소로 사용됩니다.
CRYPT_XML_KEY_VALUE 서명의 유효성을 검사하는 데 유용할 수 있는 단일 공개 키를 포함합니다.
CRYPT_XML_KEYINFO_PARAM CryptXmlSign 함수에서 인코딩할 KeyInfo 요소의 멤버를 지정하는 데 사용됩니다.
CRYPT_XML_OBJECT 서명의 Object 요소에 대해 설명합니다.
CRYPT_XML_PROPERTY CryptXML 속성에 대한 정보를 포함합니다.
CRYPT_XML_REFERENCE Reference 요소를 채우는 데 사용되는 정보를 포함합니다.
CRYPT_XML_REFERENCES CRYPT_XML_REFERENCE 구조체의 배열을 정의합니다.
CRYPT_XML_SIGNATURE Signature 요소를 채우는 데 사용되는 정보를 포함합니다.
CRYPT_XML_SIGNED_INFO XML로 인코딩된 SignedInfo 요소에 대해 설명합니다.
CRYPT_XML_STATUS

서명 유효성 검사 상태, SignedInfo 요소에 대한 요약 상태 정보 또는 참조 요소 배열에 대한 요약 상태 정보에 대한 정보를 반환합니다.
CRYPT_XML_TRANSFORM_CHAIN_CONFIG XML 디지털 서명에서 사용할 수 있는 애플리케이션 정의 변환을 포함합니다.
CRYPT_XML_TRANSFORM_INFO 데이터 변환을 적용할 때 사용되는 정보를 포함합니다.
CRYPT_XML_X509DATA X509Data 요소의 선택 시퀀스를 나타냅니다.
CRYPT_XML_X509DATA_ITEM X509Data 명명된 요소에서 인코딩할 X.509 데이터를 나타냅니다.
CRYPTCATATTRIBUTE CRYPTCATATTRIBUTE 구조체는 카탈로그 특성을 정의합니다. 이 구조체는 CryptCATEnumerateAttr 및 CryptCATEnumerateCatAttr 함수에서 사용됩니다.
CRYPTCATCDF CDF(카탈로그 정의 파일)에서 서명된 카탈로그 파일(.cat)을 만드는 데 사용되는 정보를 포함합니다.
CRYPTCATMEMBER CRYPTCATMEMBER 구조체는 카탈로그 멤버에 대한 정보를 제공합니다. 이 구조체는 CryptCATGetMemberInfo 및 CryptCATEnumerateAttr 함수에서 사용됩니다.
CRYPTCATSTORE 카탈로그 파일을 나타냅니다.
CRYPTNET_URL_CACHE_FLUSH_INFO CUC(Cryptnet URL Cache) 서비스에서 URL 캐시 항목을 유지하기 위해 사용하는 만료 정보를 포함합니다.
CRYPTNET_URL_CACHE_PRE_FETCH_INFO CUC(Cryptnet URL Cache) 서비스에서 URL 캐시 항목을 유지하기 위해 사용하는 업데이트 정보를 포함합니다.
CRYPTNET_URL_CACHE_RESPONSE_INFO CUC(Cryptnet URL Cache) 서비스에서 URL 캐시 항목을 유지하기 위해 사용하는 응답 정보를 포함합니다.

CRYPTO_SETTINGS 비활성화된 암호화 설정을 나타냅니다.
CRYPTPROTECT_PROMPTSTRUCT CryptProtectData 및 CryptUnprotectData 함수를 사용할 때 프롬프트의 텍스트와 해당 프롬프트가 표시되는 시기 및 위치에 대한 정보를 제공합니다.
CRYPTUI_CERT_MGR_STRUCT 인증서 관리자 대화 상자에 대한 정보를 포함합니다.
CRYPTUI_INITDIALOG_STRUCT CRYPTUI_VIEWCERTIFICATE_STRUCT 구조를 지원합니다.
CRYPTUI_VIEWCERTIFICATE_STRUCTA 볼 인증서에 대한 정보를 포함합니다. 이 구조체는 CryptUIDlgViewCertificate 함수에 사용됩니다. (ANSI)
CRYPTUI_VIEWCERTIFICATE_STRUCTW 볼 인증서에 대한 정보를 포함합니다. 이 구조체는 CryptUIDlgViewCertificate 함수에 사용됩니다. (유니코드)
CRYPTUI_WIZ_DIGITAL_SIGN_BLOB_INFO CryptUIWizDigitalSign 함수에서 사용하는 공개 키 BLOB에 대한 정보를 포함합니다.
CRYPTUI_WIZ_DIGITAL_SIGN_CERT_PVK_INFO CryptUIWizDigitalSign 함수에서 사용하는 인증서를 포함하는 PVK 파일에 대한 정보를 포함합니다.
CRYPTUI_WIZ_DIGITAL_SIGN_CONTEXT CryptUIWizDigitalSign 함수와 함께 BLOB에 대한 정보를 포함하는 데 사용됩니다.
CRYPTUI_WIZ_DIGITAL_SIGN_EXTENDED_INFO 서명에 대한 확장된 정보를 포함하기 위해 CRYPTUI_WIZ_DIGITAL_SIGN_INFO 구조체와 함께 사용됩니다.
CRYPTUI_WIZ_DIGITAL_SIGN_INFO 디지털 서명에 대한 정보를 포함합니다.
CRYPTUI_WIZ_DIGITAL_SIGN_PVK_FILE_INFO

디지털 서명 마법사에서 사용하는 PVK 파일에 대한 정보를 포함하기 위해 CRYPTUI_WIZ_DIGITAL_SIGN_INFO 구조체와 함께 사용됩니다.
CRYPTUI_WIZ_DIGITAL_SIGN_STORE_INFO 디지털 서명 마법사에서 사용하는 인증서 저장소에 대한 정보를 포함합니다.
CRYPTUI_WIZ_EXPORT_CERTCONTEXT_INFO 인증서가 내보내는 개체일 때 CryptUIWizExport 함수의 작업을 제어하는 정보를 포함합니다.
CRYPTUI_WIZ_EXPORT_INFO CryptUIWizExport 함수의 작업을 제어하는 정보를 포함합니다.
CRYPTUI_WIZ_IMPORT_SRC_INFO CryptUIWizImport 함수로 가져올 주체를 포함합니다.
CT_EXTRA_CERT_CHAIN_POLICY_STATUS 이 구조체에는 인증서 투명도 체인 정책 검사의 상태에 대한 추가 오류 정보가 포함되어 있습니다.
CTL_ANY_SUBJECT_INFO 인증서 신뢰 목록(CTL)에서 일치시킬 SubjectAlgorithm 및 CertFindSubjectInCTL 호출의 CTL 항목 중 하나에서 일치시킬 SubjectIdentifier를 포함합니다.
CTL_CONTEXT CTL_CONTEXT 구조체에는 CTL의 인코딩된 표현과 디코딩된 표현이 모두 포함됩니다.
CTL_ENTRY CTL(인증서 신뢰 목록)의 요소입니다.
CTL_FIND_SUBJECT_PARA 인증서 신뢰 목록(CTL)을 찾기 위해 CTL_FIND_SUBJECT dwFindType 매개 변수와 함께 CertFindCTLInStore에서 사용하는 데이터를 포함합니다.
CTL_FIND_USAGE_PARA CTL_FIND_SUBJECT_PARA 구조체의 멤버이며 CertFindCTLInStore에서 사용됩니다.
CTL_INFO CTL(인증서 신뢰 목록)에 저장된 정보를 포함합니다.

CTL_MODIFY_REQUEST CTL(인증서 신뢰 목록)을 수정하는 요청을 포함합니다. 이 구조체는 CertModifyCertificatesToTrust 함수에 사용됩니다.
CTL_USAGE CTL(인증서 신뢰 목록) 확장에 대한 OID(개체 식별자) 배열을 포함합니다.
CTL_USAGE_MATCH 인증서 체인을 빌드하는 데 사용되는 CTL(인증서 신뢰 목록)을 찾기 위한 매개 변수를 제공합니다.
CTL_VERIFY_USAGE_PARA CTL_VERIFY_USAGE_PARA 구조에는 CertVerifyCTLUsage에서 CTL 사용의 유효성을 설정하는 데 사용하는 매개 변수가 포함되어 있습니다.
CTL_VERIFY_USAGE_STATUS CertVerifyCTLUsage에서 반환된 CTL(인증서 신뢰 목록)에 대한 정보를 포함합니다.
DHPRIVKEY_VER3 키 BLOB에 포함된 특정 프라이빗 키와 관련된 정보를 포함합니다.
DHPUBKEY 키 BLOB에 포함된 특정 Diffie-Hellman 공개 키와 관련된 정보를 포함합니다.
DHPUBKEY_VER3 키 BLOB에 포함된 특정 공개 키와 관련된 정보를 포함합니다.
DIAGNOSTIC_DATA_EVENT_BINARY_STATS 이 이진 및 전송된 진단 데이터의 양을 설명하는 리소스입니다.
DIAGNOSTIC_DATA_EVENT_CATEGORY_DESCRIPTION 식별자와 이름으로 정의된 범주를 나타내는 리소스입니다. 범주는 지정된 생산자에 대한 레코드를 분류하는 조직 구문입니다. 예를 들어 "데이터 검색"은 생산자 "Microsoft Edge"의 범주일 수 있습니다.
DIAGNOSTIC_DATA_EVENT_PRODUCER_DESCRIPTION 생산자를 나타내는 리소스입니다. 생산자는 이벤트를 내보내는 OS 구성 요소, 애플리케이션 또는 서비스입니다. 예를 들어 "Microsoft Edge"는 Microsoft Edge 브라우저의 생산자 ID입니다.

DIAGNOSTIC_DATA_EVENT_TAG_DESCRIPTION 태그 이름 및 해당 설명으로 정의된 태그를 설명하는 리소스입니다.
DIAGNOSTIC_DATA_EVENT_TAG_STATS 개인 정보 태그 및 이 개인 정보 태그가 있는 이벤트 수를 포함하는 리소스입니다.
DIAGNOSTIC_DATA_EVENT_TRANSCRIPT_CONFIGURATION 최대 스토리지 크기 및 데이터 기록 시간과 같은 이벤트 기록 구성 세부 정보입니다.
DIAGNOSTIC_DATA_GENERAL_STATS 이 리소스에는 진단 데이터 레코드 집합에 대한 일반 통계가 포함되어 있습니다.
DIAGNOSTIC_DATA_RECORD 이 리소스는 개별 진단 데이터 레코드(이벤트)를 설명합니다.
DIAGNOSTIC_DATA_SEARCH_CRITERIA 이 리소스에는 진단 데이터 레코드를 가져올 때 검색 조건의 세부 정보가 포함됩니다.
DIAGNOSTIC_REPORT_DATA 이 리소스에는 진단 보고서에 대한 정보가 포함되어 있습니다.
DIAGNOSTIC_REPORT_PARAMETER 오류 보고서의 매개 변수를 설명하는 리소스입니다.
DIAGNOSTIC_REPORT_SIGNATURE 이 리소스는 진단 보고서의 서명을 설명합니다.
DOMAIN_PASSWORD_INFORMATION 암호의 최소 길이 및 고유 암호가 필요한 방법과 같은 도메인의 암호 정책에 대한 정보를 포함합니다.
DSSSEED DSS 공개 키의 소수를 확인하는 데 사용할 수 있는 시드 및 카운터 값을 보유합니다.
EFFPERM_RESULT_LIST 유효 사용 권한을 나열합니다.
ENCRYPTED_CREDENTIALW

암호화된 자격 증명을 나타냅니다.
ENUM_SERVICE_STATUS_PROCESSA 서비스 제어 관리자 데이터베이스의 서비스 이름과 서비스에 대한 정보를 포함합니다. EnumServicesStatusEx 함수에서 사용됩니다. (ANSI)
ENUM_SERVICE_STATUS_PROCESSW 서비스 제어 관리자 데이터베이스의 서비스 이름과 서비스에 대한 정보를 포함합니다. EnumServicesStatusEx 함수에서 사용됩니다. (유니코드)
ENUM_SERVICE_STATUSA 서비스 제어 관리자 데이터베이스의 서비스 이름과 해당 서비스에 대한 정보를 포함합니다. EnumDependentServices 및 EnumServicesStatus 함수에서 사용됩니다. (ANSI)
ENUM_SERVICE_STATUSW 서비스 제어 관리자 데이터베이스의 서비스 이름과 해당 서비스에 대한 정보를 포함합니다. EnumDependentServices 및 EnumServicesStatus 함수에서 사용됩니다. (유니코드)
EV_EXTRA_CERT_CHAIN_POLICY_PARA EV 정책 유효성 검사를 위해 전달되는 매개 변수를 지정합니다. 애플리케이션은 이 구조를 사용하여 확장 유효성 검사 인증서의 정책 한정자 플래그 중 애플리케이션에 중요한 정책 한정자 플래그를 나타내는 힌트를 API에 전달합니다.
EV_EXTRA_CERT_CHAIN_POLICY_STATUS CertVerifyCertificateChainPolicy 함수 호출에서 반환된 정책 플래그를 포함합니다.
EXPLICIT_ACCESS_A 지정된 트러스티에 대한 액세스 제어 정보를 정의합니다. (ANSI)
EXPLICIT_ACCESS_W 지정된 트러스티에 대한 액세스 제어 정보를 정의합니다. (유니코드)
GENERIC_MAPPING 개체에 대한 특정 및 표준 액세스 권한에 대한 일반 액세스 권한의 매핑을 정의합니다.
HMAC_INFO HMAC_INFO 구조체는 해시 알고리즘과 HMAC 해시를 계산하는 데 사용할 내부 및 외부 문자열을 지정합니다.

HTTPSPolicyCallbackData SSL(Secure Sockets Layer) 클라이언트/서버 인증서 체인의 확인에 사용되는 정책 정보를 보유합니다.
INHERITED_FROMA 개체의 상속된 ACE(액세스 제어 항목)에 대한 정보를 제공합니다. (ANSI)
INHERITED_FROMW 개체의 상속된 ACE(액세스 제어 항목)에 대한 정보를 제공합니다. (유니코드)
KERB_ADD_BINDING_CACHE_ENTRY_EX_REQUEST 사용자가 특정 DC(도메인 컨트롤러)에 바인딩하여 Kerberos 도메인 바인딩 캐시를 재정의할 수 있습니다.
KERB_ADD_BINDING_CACHE_ENTRY_REQUEST 바인딩 캐시 항목을 추가할 메시지를 지정합니다.
KERB_ADD_CREDENTIALS_REQUEST 로그온 세션에 대한 추가 서버 자격 증명을 추가, 제거 또는 바꿀 메시지를 지정합니다.
KERB_ADD_CREDENTIALS_REQUEST_EX 로그온 세션에 대한 추가 서버 자격 증명을 추가, 제거 또는 바꿀 메시지와 해당 자격 증명과 연결할 SPN(서비스 사용자 이름)을 지정합니다.
KERB_BINDING_CACHE_ENTRY_DATA 바인딩 캐시 항목에 대한 데이터를 지정합니다.
KERB_CERTIFICATE_HASHINFO 인증서 해시의 페이로드 정보를 제공합니다.
KERB_CERTIFICATE_INFO 인증서 정보를 포함합니다.
KERB_CERTIFICATE_LOGON 스마트 카드 로그온 세션에 대한 정보를 포함합니다. (KERB_CERTIFICATE_LOGON)
KERB_CERTIFICATE_S4U_LOGON 사용자용 서비스(S4U) 로그온에 대한 인증서에 대한 정보를 포함합니다.

KERB_CERTIFICATE_UNLOCK_LOGON 대화형 스마트 카드 로그온 세션 중에 잠긴 워크스테이션의 잠금을 해제하는 데 사용되는 정보를 포함합니다.
KERB_CHANGEPASSWORD_REQUEST 암호를 변경하는 데 사용되는 정보를 포함합니다.
KERB_CLEANUP_MACHINE_PKINIT_CREDS_REQUEST 컴퓨터에서 PKINIT 디바이스 자격 증명을 정리합니다.
KERB_CRYPTO_KEY Kerberos 암호화 세션 키에 대한 정보를 포함합니다.
KERB_EXTERNAL_NAME 외부 이름에 대한 정보를 포함합니다.
KERB_EXTERNAL_TICKET 외부 티켓에 대한 정보를 포함합니다.
KERB_INTERACTIVE_LOGON 대화형 로그온 세션에 대한 정보를 포함합니다.
KERB_INTERACTIVE_PROFILE KERB_INTERACTIVE_PROFILE 구조에는 대화형 로그온 프로필에 대한 정보가 포함됩니다. 이 구조체는 LsaLogonUser 함수에서 사용됩니다.
KERB_INTERACTIVE_UNLOCK_LOGON 대화형 로그온 세션 중에 잠긴 워크스테이션의 잠금을 해제하는 데 사용되는 정보를 포함합니다.
KERB_PURGE_BINDING_CACHE_REQUEST 바인딩 캐시에 대한 요청을 삭제합니다.
KERB_PURGE_TKT_CACHE_REQUEST 티켓 캐시에서 항목을 삭제하는 데 사용되는 정보를 포함합니다.
KERB_QUERY_BINDING_CACHE_REQUEST 바인딩 캐시를 쿼리하는 데 사용되는 정보를 포함합니다.

KERB_QUERY_BINDING_CACHE_RESPONSE 바인딩 캐시를 쿼리한 결과를 포함합니다.
KERB_QUERY_DOMAIN_EXTENDED_POLICIES_REQUEST 확장 정책에 대한 도메인을 쿼리하는 데 사용되는 정보를 포함합니다.
KERB_QUERY_DOMAIN_EXTENDED_POLICIES_RESPONSE 지정된 도메인의 확장 정책에 대한 쿼리 결과를 포함합니다.
KERB_QUERY_TKT_CACHE_REQUEST 티켓 캐시를 쿼리하는 데 사용되는 정보를 포함합니다.
KERB_QUERY_TKT_CACHE_RESPONSE 티켓 캐시를 쿼리한 결과를 포함합니다.
KERB_RETRIEVE_TKT_REQUEST 티켓을 검색하는 데 사용되는 정보를 포함합니다.
KERB_RETRIEVE_TKT_RESPONSE 티켓 검색의 응답을 포함합니다.
KERB_S4U_LOGON 사용자용 서비스(S4U) 로그온에 대한 정보를 포함합니다.
KERB_SMART_CARD_LOGON 스마트 카드 로그온 세션에 대한 정보를 포함합니다. (KERB_SMART_CARD_LOGON)
KERB_SMART_CARD_UNLOCK_LOGON 스마트 카드 로그온 세션 중에 잠긴 워크스테이션의 잠금을 해제하는 데 사용되는 정보를 포함합니다.
KERB_TICKET_CACHE_INFO 캐시된 Kerberos 티켓에 대한 정보를 포함합니다. Kerberos 티켓은 인터넷 RFC 4120에 정의되어 있습니다. 자세한 내용은 http://www.ietf.org 를 참조하세요.
KERB_TICKET_LOGON 네트워크 로그온에 대한 프로필 정보를 포함합니다.

KERB_TICKET_PROFILE KERB_TICKET_PROFILE 구조에는 대화형 로그온 프로필에 대한 정보가 포함됩니다. 이 구조체는 LsaLogonUser에서 반환됩니다.
KERB_TICKET_UNLOCK_LOGON 워크스테이션의 잠금을 해제하는 정보를 포함합니다.
KeyCredentialManagerInfo KeyCredentialManagerGetInformation에서 반환된 데이터 구조입니다.
LSA_AUTH_INFORMATION LSA_AUTH_INFORMATION 구조에는 신뢰할 수 있는 도메인에 대한 인증 정보가 포함됩니다.
LSA_DISPATCH_TABLE Windows 인증 패키지에서 호출할 수 있는 LSA(로컬 보안 기관) 함수에 대한 포인터를 포함합니다.
LSA_ENUMERATION_INFORMATION LSA_ENUMERATION_INFORMATION 구조체는 LsaEnumerateAccountsWithUserRight 함수와 함께 사용하여 SID에 대한 포인터를 반환합니다.
LSA_FOREST_TRUST_BINARY_DATA 로컬 보안 기관 포리스트 트러스트 작업에 사용되는 이진 데이터를 포함합니다.
LSA_FOREST_TRUST_COLLISION_INFORMATION 로컬 보안 기관 포리스트 트러스트 충돌에 대한 정보를 포함합니다.
LSA_FOREST_TRUST_COLLISION_RECORD 로컬 보안 기관 포리스트 트러스트 충돌에 대한 정보를 포함합니다.
LSA_FOREST_TRUST_DOMAIN_INFO 도메인에 대한 식별 정보를 포함합니다.
LSA_FOREST_TRUST_INFORMATION 로컬 보안 기관 포리스트 트러스트 정보를 포함합니다.
LSA_FOREST_TRUST_RECORD 로컬 보안 기관 포리스트 트러스트 레코드를 나타냅니다.

LSA_LAST_INTER_LOGON_INFO 로그온 세션에 대한 정보를 포함합니다. (LSA_LAST_INTER_LOGON_INFO)
LSA_OBJECT_ATTRIBUTES LSA_OBJECT_ATTRIBUTES 구조체는 LsaOpenPolicy 함수와 함께 사용하여 Policy 개체에 대한 연결의 특성을 지정합니다.
LSA_REFERENCED_DOMAIN_LIST LSA_REFERENCED_DOMAIN_LIST 구조에는 조회 작업에서 참조되는 도메인에 대한 정보가 포함됩니다.
LSA_SECPKG_FUNCTION_TABLE 보안 패키지에서 호출할 수 있는 LSA 함수에 대한 포인터를 포함합니다. LSA(로컬 보안 기관)는 패키지의 SpInitialize 함수를 호출할 때 이 구조를 보안 패키지에 전달합니다.
LSA_STRING LSA(로컬 보안 기관) 함수에서 ANSI 문자열을 지정하는 데 사용됩니다.
LSA_TOKEN_INFORMATION_NULL 인증되지 않은 시스템 액세스가 필요한 경우에 사용됩니다.
LSA_TOKEN_INFORMATION_V1 인증 패키지가 버전 2 Windows 토큰 개체에 배치할 수 있고 LSA_TOKEN_INFORMATION_V1 대체한 정보를 포함합니다.
LSA_TOKEN_INFORMATION_V3 LSA 토큰에 클레임 지원을 추가하고 인증 패키지가 버전 3 Windows 토큰 개체에 배치할 수 있고 LSA_TOKEN_INFORMATION_V1 대체한 정보를 포함합니다.
LSA_TRANSLATED_NAME LsaLookupSids 함수와 함께 SID로 식별된 계정에 대한 정보를 반환하는 데 사용됩니다.
LSA_TRANSLATED_SID LsaLookupNames 함수와 함께 계정을 식별하는 SID에 대한 정보를 반환하는 데 사용됩니다.
LSA_TRANSLATED_SID2 계정 이름에 따라 검색되는 SID를 포함합니다.

LSA_TRUST_INFORMATION 도메인을 식별합니다.
LSA_UNICODE_STRING LSA_UNICODE_STRING 구조체는 다양한 LSA(로컬 보안 기관) 함수에서 유니코드 문자열을 지정하는 데 사용됩니다.
LUID_AND_ATTRIBUTES LUID(로컬 고유 식별자) 및 해당 특성을 나타냅니다.
MS_ADDINFO_BLOB 메모리 내 BLOB 주체 유형에 대한 추가 정보를 제공합니다.
MS_ADDINFO_CATALOGMEMBER 카탈로그 멤버 주체 유형에 대한 추가 정보를 제공합니다.
MS_ADDINFO_FLAT 플랫 또는 엔드 투 엔드 주체 유형에 대한 추가 정보를 제공합니다.
MSA_INFO_0 관리되는 서비스 계정에 대한 정보를 지정합니다.
MSV1_0_INTERACTIVE_LOGON 대화형 로그온에 대한 정보를 포함합니다.
MSV1_0_INTERACTIVE_PROFILE MSV1_0_INTERACTIVE_PROFILE 구조에는 대화형 로그온 프로필에 대한 정보가 포함됩니다. 이 구조체는 LsaLogonUser 함수에서 사용됩니다.
MSV1_0_LM20_LOGON 네트워크 로그온에 사용되는 로그온 정보를 포함합니다.
MSV1_0_LM20_LOGON_PROFILE 네트워크 로그온 세션에 대한 정보를 포함합니다.
MSV1_0_SUBAUTH_LOGON 하위 인증 DLL에서 사용됩니다.

MSV1_0_SUBAUTH_REQUEST 하위 인증 패키지에 전달할 정보를 포함합니다.
MSV1_0_SUBAUTH_RESPONSE 하위 인증 패키지의 응답을 포함합니다.
MSV1_0_SUPPLEMENTAL_CREDENTIAL MSV1_0_SUPPLEMENTAL_CREDENTIAL 구조는 Kerberos 또는 사용자 지정 인증 패키지에서 MSV1_0 자격 증명을 전달하는 데 사용됩니다.
NCRYPT_ALLOC_PARA 데이터를 할당하고 해제하는 데 사용할 수 있는 사용자 지정 함수를 지정할 수 있습니다.
NCRYPT_KEY_BLOB_HEADER 키 BLOB을 포함합니다.
NCRYPT_PROTECT_STREAM_INFO NCryptStreamOpenToProtect 및 NCryptStreamOpenToUnprotect 함수에서 처리된 데이터 블록을 애플리케이션에 전달하는 데 사용됩니다.
NCRYPT_SUPPORTED_LENGTHS 키에 대한 길이 정보를 포함하기 위해 NCRYPT_LENGTHS_PROPERTY 속성과 함께 사용됩니다.
NCRYPT_UI_POLICY 키에 대한 강력한 키 사용자 인터페이스 정보를 포함하기 위해 NCRYPT_UI_POLICY_PROPERTY 속성과 함께 사용됩니다.
NCryptAlgorithmName CNG 알고리즘에 대한 정보를 포함하는 데 사용됩니다.
NCryptKeyName CNG 키에 대한 정보를 포함하는 데 사용됩니다.
NCryptProviderName CNG 키 스토리지 공급자의 이름을 포함하는 데 사용됩니다.
NETCONNECTINFOSTRUCT

NETCONNECTINFOSTRUCT 구조에는 네트워크의 성능에 대한 정보가 포함되어 있습니다. NPGetConnectionPerformance 함수에서 사용됩니다.
NETLOGON_LOGON_IDENTITY_INFO 로그온 하위 인증을 위해 사용자에 대한 정보를 전달하는 데 사용됩니다.
NETRESOURCEA 다음 구조에는 네트워크 리소스에 대한 정보가 포함되어 있습니다. NPOpenEnum 및 NPAddConnection을 비롯한 여러 네트워크 공급자 함수에서 사용됩니다. (ANSI)
NETRESOURCEW 다음 구조에는 네트워크 리소스에 대한 정보가 포함되어 있습니다. NPOpenEnum 및 NPAddConnection을 비롯한 여러 네트워크 공급자 함수에서 사용됩니다. (유니코드)
NOTIFYADD NOTIFYADD 구조에는 네트워크 연결 작업의 세부 정보가 포함됩니다. AddConnectNotify 함수에서 사용됩니다.
NOTIFYCANCEL NOTIFYCANCEL 구조에는 네트워크 연결 끊기 작업의 세부 정보가 포함됩니다. CancelConnectNotify 함수에서 사용됩니다.
NOTIFYINFO NOTIFYINFO 구조에는 네트워크 연결 또는 연결 끊기 작업에 대한 상태 정보가 포함됩니다. AddConnectNotify 및 CancelConnectNotify 함수에서 사용됩니다.
OBJECT_TYPE_LIST 개체 형식의 계층 구조에서 개체 형식 요소를 식별합니다.
OBJECTS_AND_NAME_A 이름별로 트러스티를 식별하는 문자열과 ACE(개체별 액세스 제어 항목)의 개체 형식을 식별하는 추가 문자열을 포함합니다. (ANSI)
OBJECTS_AND_NAME_W 이름별로 트러스티를 식별하는 문자열과 ACE(개체별 액세스 제어 항목)의 개체 형식을 식별하는 추가 문자열을 포함합니다. (유니코드)
OBJECTS_AND_SID

ACE(개체별 액세스 제어 항목)의 개체 유형을 식별하는 트러스티 및 GUID를 식별하는 SID(보안 식별자)를 포함합니다.
OCSP_BASIC_RESPONSE_ENTRY 단일 인증서에 대한 현재 인증서 상태를 포함합니다.
OCSP_BASIC_RESPONSE_INFO RFC 2560에서 지정한 기본 OCSP(온라인 인증서 상태 프로토콜) 응답을 포함합니다.
OCSP_BASIC_REVOKED_INFO 인증서가 해지된 이유를 포함합니다.
OCSP_BASIC_SIGNED_RESPONSE_INFO 서명이 있는 기본 OCSP(온라인 인증서 상태 프로토콜) 응답을 포함합니다.
OCSP_CERT_ID OCSP(온라인 인증서 상태 프로토콜) 요청 또는 응답에서 인증서를 식별하는 정보를 포함합니다.
OCSP_REQUEST_ENTRY OCSP(온라인 인증서 상태 프로토콜) 요청의 단일 인증서에 대한 정보를 포함합니다.
OCSP_REQUEST_INFO RFC 2560에서 지정한 온라인 OCSP(인증서 상태 프로토콜) 요청에 대한 정보를 포함합니다.
OCSP_RESPONSE_INFO 해당 OCSP(온라인 인증서 상태 프로토콜) 요청의 성공 또는 실패를 나타냅니다. 성공적인 요청의 경우 응답 정보의 형식과 값이 포함됩니다.
OCSP_SIGNATURE_INFO OCSP(온라인 인증서 상태 프로토콜) 요청 또는 응답에 대한 서명을 포함합니다.
OCSP_SIGNED_REQUEST_INFO 선택적 서명 정보가 있는 OCSP(온라인 인증서 상태 프로토콜) 요청에 대한 정보를 포함합니다.
OLD_LARGE_INTEGER 부가된 64비트 정수 값을 두 개의 32비트 정수로 나타내는 데 사용됩니다.
OPENCARD_SEARCH_CRITERIAA

OPENCARD_SEARCH_CRITERIA 구조체는 호출자가 설정한 요구 사항을 충족하는 카드를 인식하기 위해 SCardUIDlgSelectCard 함수에서 사용됩니다. 그러나 이 구조를 사용하지 않고 SCardUIDlgSelectCard를 호출할 수 있습니다. (ANSI)
OPENCARD_SEARCH_CRITERIAW OPENCARD_SEARCH_CRITERIA 구조체는 호출자가 설정한 요구 사항을 충족하는 카드를 인식하기 위해 SCardUIDlgSelectCard 함수에서 사용됩니다. 그러나 이 구조를 사용하지 않고 SCardUIDlgSelectCard를 호출할 수 있습니다. (유니코드)
OPENCARDNAME_EXA OPENCARDNAME_EX 구조에는 SCardUIDlgSelectCard 함수가 스마트 카드 선택 대화 상자를 초기화하는 데 사용하는 정보가 포함됩니다. (ANSI)
OPENCARDNAME_EXW OPENCARDNAME_EX 구조에는 SCardUIDlgSelectCard 함수가 스마트 카드 선택 대화 상자를 초기화하는 데 사용하는 정보가 포함됩니다. (유니코드)
OPENCARDNAMEA GetOpenCardName 함수가 스마트 카드 선택 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. (ANSI)
OPENCARDNAMEW GetOpenCardName 함수가 스마트 카드 선택 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. (유니코드)
PKCS12_PBES2_EXPORT_PARAMS dwFlags에 대해 PKCS12_EXPORT_PBES2_PARAMS 플래그를 설정하여 사용할 암호화 알고리즘에 대한 정보를 제공할 때 PFXExportCertStoreEx 함수를 pvPara로 전달합니다.
PKU2U_CERT_BLOB PKU2U 인증서 데이터를 지정합니다.
PKU2U_CERTIFICATE_S4U_LOGON S4U 로그온에 사용되는 인증서를 지정합니다.
PKU2U_CREDUI_CONTEXT PKU2U 클라이언트 컨텍스트를 지정합니다.
POLICY_ACCOUNT_DOMAIN_INFO

시스템 계정 도메인의 이름 및 SID를 설정하고 쿼리하는 데 사용됩니다.
POLICY_AUDIT_EVENTS_INFO POLICY_AUDIT_EVENTS_INFO 구조는 시스템의 감사 규칙을 설정하고 쿼리하는 데 사용됩니다.
POLICY_AUDIT_SID_ARRAY Windows 사용자 또는 그룹을 나타내는 SID 구조의 배열을 지정합니다.
POLICY_DNS_DOMAIN_INFO POLICY_DNS_DOMAIN_INFO 구조는 Policy 개체와 연결된 주 도메인에 대한 DNS(도메인 이름 시스템) 정보를 설정하고 쿼리하는 데 사용됩니다.
POLICY_LSA_SERVER_ROLE_INFO LSA 서버의 역할을 설정하고 쿼리하는 데 사용됩니다.
POLICY_MODIFICATION_INFO POLICY_MODIFICATION_INFO 구조체는 LSA 데이터베이스의 생성 시간 및 마지막 수정에 대한 정보를 쿼리하는 데 사용됩니다.
POLICY_PRIMARY_DOMAIN_INFO PolicyPrimaryDomainInformation 값 및 POLICY_PRIMARY_DOMAIN_INFO 구조체는 사용되지 않습니다. 대신 PolicyDnsDomainInformation 및 POLICY_DNS_DOMAIN_INFO 구조를 사용합니다.
PRIVILEGE_SET 권한 집합을 지정합니다.
PROCESS_MACHINE_INFORMATION 프로세스의 아키텍처를 지정하고 호스트 운영 체제의 사용자 모드, 커널 모드 및/또는 WoW64에서 코드 아키텍처를 실행할 수 있는지 여부를 지정합니다.
PROV_ENUMALGS CSP(암호화 서비스 공급자)에서 지원하는 알고리즘에 대한 정보를 포함하도록 PP_ENUMALGS 매개 변수를 검색할 때 CryptGetProvParam 함수와 함께 사용됩니다.
PROV_ENUMALGS_EX CSP(암호화 서비스 공급자)에서 지원하는 알고리즘에 대한 정보를 포함하도록 PP_ENUMALGS_EX 매개 변수를 검색할 때 CryptGetProvParam 함수와 함께 사용됩니다.
QUERY_SERVICE_CONFIGA

설치된 서비스에 대한 구성 정보를 포함합니다. QueryServiceConfig 함수에서 사용됩니다. (ANSI)
QUERY_SERVICE_CONFIGW 설치된 서비스에 대한 구성 정보를 포함합니다. QueryServiceConfig 함수에서 사용됩니다. (유니코드)
QUERY_SERVICE_LOCK_STATUSA 서비스 제어 관리자 데이터베이스의 잠금 상태에 대한 정보를 포함합니다. QueryServiceLockStatus 함수에서 사용됩니다. (ANSI)
QUERY_SERVICE_LOCK_STATUSW 서비스 제어 관리자 데이터베이스의 잠금 상태에 대한 정보를 포함합니다. QueryServiceLockStatus 함수에서 사용됩니다. (유니코드)
QUOTA_LIMITS 사용자가 사용할 수 있는 시스템 리소스의 양을 설명합니다.
REMOTE_NAME_INFOA REMOTE_NAME_INFO 구조에는 범용 이름의 원격 형식에 대한 정보가 포함됩니다. NPGetUniversalName 함수에서 사용됩니다. (ANSI)
REMOTE_NAME_INFOW REMOTE_NAME_INFO 구조에는 범용 이름의 원격 형식에 대한 정보가 포함됩니다. NPGetUniversalName 함수에서 사용됩니다. (유니코드)
ROOT_INFO_LUID 암호화 스마트 카드 루트 정보에 대한 LUID(로컬 고유 식별자)를 포함합니다.
RSAPUBKEY RSAPUBKEY 구조에는 키 BLOB에 포함된 특정 공개 키와 관련된 정보가 포함되어 있습니다.
SAFER_CODE_PROPERTIES_V1 코드 이미지에서 확인할 코드 이미지 정보 및 조건을 포함합니다. (SAFER_CODE_PROPERTIES_V1)
SAFER_CODE_PROPERTIES_V2 코드 이미지에서 확인할 코드 이미지 정보 및 조건을 포함합니다. S
SAFER_HASH_IDENTIFICATION 해시 식별 규칙을 나타냅니다.

SAFER_IDENTIFICATION_HEADER SAFER_IDENTIFICATION_HEADER 구조체는 SAFER_PATHNAME_IDENTIFICATION, SAFER_HASH_IDENTIFICATION 및 SAFER_URLZONE_IDENTIFICATION 구조체의 헤더로 사용됩니다.
SAFER_PATHNAME_IDENTIFICATION 경로 식별 규칙을 나타냅니다.
SAFER_URLZONE_IDENTIFICATION URL 영역 식별 규칙을 나타냅니다.
SC_ACTION 서비스 제어 관리자가 수행할 수 있는 작업을 나타냅니다.
SCARD_ATRMASK SCardLocateCardsByATR 함수에서 카드를 찾는 데 사용됩니다.
SCARD_READERSTATEA 판독기 내에서 스마트 카드를 추적하기 위한 함수에서 사용됩니다. (ANSI)
SCARD_READERSTATEW 판독기 내에서 스마트 카드를 추적하기 위한 함수에서 사용됩니다. (유니코드)
SCESVC_ANALYSIS_INFO 분석 정보를 포함합니다.
SCESVC_ANALYSIS_LINE SCESVC_ANALYSIS_LINE 구조체에는 SCESVC_ANALYSIS_INFO 구조체에 지정된 특정 줄의 키, 값 및 값 길이가 포함됩니다.
SCESVC_CALLBACK_INFO SCESVC_CALLBACK_INFO 구조에는 쿼리, 설정 및 사용 가능한 정보에 대한 불투명 데이터베이스 핸들 및 콜백 함수 포인터가 포함되어 있습니다.
SCESVC_CONFIGURATION_INFO SCESVC_CONFIGURATION_INFO 구조는 서비스에 대한 구성 정보를 제공합니다. 이 구조는 구성 정보를 지정할 때 PFSCE_QUERY_INFO 및 PFSCE_SET_INFO 함수에서 사용됩니다.

SCESVC_CONFIGURATION_LINE SCESVC_CONFIGURATION_LINE 구조에는 구성 데이터 줄에 대한 정보가 포함됩니다. SCESVC_CONFIGURATION_INFO 구조체에서 사용됩니다.
SCH_CRED_PUBLIC_CERTCHAIN SCH_CRED_PUBLIC_CERTCHAIN 구조에는 단일 인증서가 포함됩니다. 이 인증서에서 인증 체인을 빌드할 수 있습니다.
SCH_CRED_SECRET_PRIVKEY 클라이언트 또는 서버를 인증하는 데 필요한 프라이빗 키 정보를 포함합니다.
SCH_CREDENTIALS Schannel 자격 증명에 대한 데이터를 포함합니다. (SCH_CREDENTIALS)
SCHANNEL_ALERT_TOKEN InitializeSecurityContext(Schannel) 함수 또는 AcceptSecurityContext(Schannel) 함수에 대한 호출 대상으로 보낼 SSL(Secure Sockets Layer Protocol) 또는 TLS(전송 계층 보안 프로토콜) 경고를 생성합니다.
SCHANNEL_ALG SCHANNEL_ALG 구조에는 알고리즘 및 키 크기 정보가 포함됩니다. dwParam이 KP_SCHANNEL_ALG 설정될 때 CryptSetKeyParam에서 pbData로 전달되는 구조체로 사용됩니다.
SCHANNEL_CERT_HASH Schannel에서 사용하는 인증서에 대한 해시 저장소 데이터를 포함합니다.
SCHANNEL_CERT_HASH_STORE Schannel이 커널 모드에서 사용하는 인증서에 대한 해시 저장소 데이터를 포함합니다.
SCHANNEL_CLIENT_SIGNATURE InitializeSecurityContext(Schannel) 함수를 호출할 때 클라이언트 인증서의 프라이빗 키에 액세스할 수 없는 경우 클라이언트 서명을 지정합니다(이 경우 함수는 SEC_I_SIGNATURE_NEEDED 반환).
SCHANNEL_CRED Schannel 자격 증명에 대한 데이터를 포함합니다. (SCHANNEL_CRED)
SCHANNEL_SESSION_TOKEN InitializeSecurityContext(Schannel) 함수 또는 AcceptSecurityContext(Schannel) 함수를 호출하여 만든 인증 세션에 대해 다시 연결을 사용할 수 있는지 여부를 지정합니다.

SEC_APPLICATION_PROTOCOL_LIST 애플리케이션 프로토콜 목록을 저장합니다.
SEC_APPLICATION_PROTOCOLS 애플리케이션 프로토콜 목록의 배열을 저장합니다.
SEC_CERTIFICATE_REQUEST_CONTEXT 인증서 요청 컨텍스트를 저장합니다.
SEC_CHANNEL_BINDINGS 보안 컨텍스트에 대한 채널 바인딩 정보를 지정합니다.
SEC_DTLS_MTU DTLS MTU를 저장합니다.
SEC_FLAGS 보안 플래그를 포함합니다.
SEC_NEGOTIATION_INFO 보안 협상 정보를 저장합니다.
SEC_PRESHAREDKEY 미리 공유된 키 정보를 포함합니다.
SEC_PRESHAREDKEY_IDENTITY 미리 공유된 키의 ID를 포함합니다.
SEC_SRTP_MASTER_KEY_IDENTIFIER SRTP 마스터 키 식별자를 저장합니다.
SEC_SRTP_PROTECTION_PROFILES SRTP 보호 프로필을 저장합니다.
SEC_TOKEN_BINDING 토큰 바인딩 정보를 저장합니다.
SEC_TRAFFIC_SECRETS

연결에 대한 트래픽 비밀을 포함합니다.
SEC_WINNT_AUTH_BYTE_VECTOR 인증 구조에서 데이터의 바이트 오프셋 및 배열 길이를 지정합니다.
SEC_WINNT_AUTH_CERTIFICATE_DATA 직렬화된 인증서 정보를 지정합니다.
SEC_WINNT_AUTH_DATA 인증 데이터를 지정합니다.
SEC_WINNT_AUTH_DATA_PASSWORD 직렬화된 암호를 지정합니다.
SEC_WINNT_AUTH_DATA_TYPE_SMARTCARD_CONTEXTS_DATA 스마트 카드 컨텍스트에 대한 인증 데이터를 포함합니다.
SEC_WINNT_AUTH_FIDO_DATA FIDO 인증에 대한 데이터를 포함합니다.
SEC_WINNT_AUTH_IDENTITY_A 인증을 위해 특정 사용자 이름 및 암호를 런타임 라이브러리에 전달할 수 있습니다. (ANSI)
SEC_WINNT_AUTH_IDENTITY_EX2 인증 ID에 대한 정보를 포함합니다.
SEC_WINNT_AUTH_IDENTITY_EXA ANSI(SEC_WINNT_AUTH_IDENTITY_EXA) 구조체에는 사용자에 대한 정보가 포함됩니다.
SEC_WINNT_AUTH_IDENTITY_EXW SEC_WINNT_AUTH_IDENTITY_EXW(유니코드) 구조체에는 사용자에 대한 정보가 포함됩니다.
SEC_WINNT_AUTH_IDENTITY_INFO 인증에 대한 ID 정보를 포함합니다.
SEC_WINNT_AUTH_IDENTITY_W 인증을 위해 특정 사용자 이름 및 암호를 런타임 라이브러리에 전달할 수 있습니다. (유니코드)

SEC_WINNT_AUTH_NGC_DATA 인증을 위한 NGC 데이터를 포함합니다.
SEC_WINNT_AUTH_PACKED_CREDENTIALS 직렬화된 자격 증명을 지정합니다.
SEC_WINNT_AUTH_PACKED_CREDENTIALS_EX 직렬화된 자격 증명 및 자격 증명을 지원하는 보안 패키지 목록을 지정합니다.
SEC_WINNT_AUTH_SHORT_VECTOR USHORT 값 배열의 오프셋 및 문자 수를 지정합니다.
SEC_WINNT_CREDUI_CONTEXT 비직렬화 자격 증명 정보를 지정합니다.
SEC_WINNT_CREDUI_CONTEXT_VECTOR SEC_WINNT_CREDUI_CONTEXT 구조에서 자격 증명 컨텍스트 데이터의 오프셋 및 크기를 지정합니다.
보안 버퍼 보안 패키지에 전달하기 위해 전송 애플리케이션에서 할당한 버퍼에 대해 설명합니다.
SecBufferDesc (초 버퍼 설명 SecBufferDesc 구조체는 전송 애플리케이션에서 보안 패키지로 전달할 SecBuffer 구조의 배열을 설명합니다.
SecHandle 보안 핸들을 나타냅니다.
SECPKG_BYTE_VECTOR 바이트 벡터 정보를 지정합니다.
SECPKG_CALL_INFO 현재 실행 중인 호출에 대한 정보를 포함합니다.
SECPKG_CLIENT_INFO SECPKG_CLIENT_INFO 구조에는 보안 패키지의 클라이언트에 대한 정보가 포함됩니다. 이 구조체는 GetClientInfo 함수에서 사용됩니다.

SECPKG_CONTEXT_THUNKS SECPKG_CONTEXT_THUNKS 구조에는 LSA 모드에서 실행할 QueryContextAttributes(일반) 호출에 대한 정보가 포함되어 있습니다. 이 구조체는 SpGetExtendedInformation 및 SpSetExtendedInformation 함수에서 사용됩니다.
SECPKG_CREDENTIAL 자격 증명을 지정합니다.
SECPKG_DLL_FUNCTIONS SECPKG_DLL_FUNCTIONS 구조에는 클라이언트/서버 애플리케이션을 사용하여 in-process를 실행하는 동안 보안 패키지에서 호출할 수 있는 LSA 함수에 대한 포인터가 포함됩니다.
SECPKG_EVENT_NOTIFY SECPKG_EVENT_NOTIFY 구조에는 보안 이벤트에 대한 정보가 포함됩니다. 이 구조는 이벤트 알림을 수신하도록 등록된 함수에 전달됩니다. 이벤트 알림 함수는 RegisterNotification 함수를 호출하여 등록됩니다.
SECPKG_EVENT_PACKAGE_CHANGE SECPKG_EVENT_PACKAGE_CHANGE 구조에는 보안 패키지 가용성의 변경 내용에 대한 정보가 포함되어 있습니다.
SECPKG_EXTENDED_INFORMATION SECPKG_EXTENDED_INFORMATION 구조체는 선택적 패키지 기능에 대한 정보를 보관하는 데 사용됩니다. 이 구조체는 SpGetExtendedInformation 및 SpSetExtendedInformation 함수에서 사용됩니다.
SECPKG_EXTRA_OIDS 확장 보안 패키지에 대한 개체 식별자(OID)를 포함합니다.
SECPKG_FUNCTION_TABLE SECPKG_FUNCTION_TABLE 구조체에는 보안 패키지에서 구현해야 하는 LSA 함수에 대한 포인터가 포함되어 있습니다. LSA(로컬 보안 기관)는 SpLsaModeInitialize 함수를 호출할 때 SSP/AP DLL에서 이 구조를 가져옵니다.
SECPKG_GSS_INFO SECPKG_GSS_INFO 구조에는 GSS 호환 협상에 사용되는 정보가 포함되어 있습니다.
SECPKG_MUTUAL_AUTH_LEVEL SECPKG_MUTUAL_AUTH_LEVEL 구조에는 보안 패키지에서 사용하는 인증 수준이 포함됩니다.

SECPKG_NEGO2_INFO NEGO2 협상에 사용되는 확장 패키지 정보를 포함합니다.
SECPKG_PARAMETERS SECPKG_PARAMETERS 구조에는 컴퓨터 시스템에 대한 정보가 포함됩니다. 이 구조체는 SpInitialize 함수에서 사용됩니다.
SECPKG_PRIMARY_CRED SECPKG_PRIMARY_CRED 구조에는 기본 자격 증명이 포함됩니다. 이 구조체는 LsaApLogonUserEx2 및 SpAcceptCredentials 함수에서 사용됩니다.
SECPKG_SERIALIZED_OID 보안 패키지의 OID(개체 식별자)를 포함합니다.
SECPKG_SHORT_VECTOR 짧은 벡터 정보를 지정합니다.
SECPKG_SUPPLEMENTAL_CRED SECPKG_SUPPLEMENTAL_CRED 구조에는 보안 패키지에서 인식하는 추가 자격 증명이 포함되어 있습니다.
SECPKG_SUPPLEMENTAL_CRED_ARRAY SECPKG_SUPPLEMENTAL_CRED_ARRAY 구조에는 추가 자격 증명 정보가 포함됩니다. 이 구조체는 LsaApLogonUserEx2 및 UpdateCredentials 함수에서 사용됩니다.
SECPKG_SUPPLIED_CREDENTIAL 제공된 자격 증명을 지정합니다.
SECPKG_TARGETINFO 인증 요청의 대상을 지정합니다.
SECPKG_USER_FUNCTION_TABLE SECPKG_USER_FUNCTION_TABLE 구조에는 클라이언트/서버 애플리케이션을 사용한 프로세스 실행을 지원하기 위해 보안 패키지가 구현하는 함수에 대한 포인터가 포함되어 있습니다. 이 구조체는 SpUserModeInitialize 함수에서 제공됩니다.
SECPKG_WOW_CLIENT_DLL WOW 인식 32비트 DLL에 대한 경로를 포함합니다.

SecPkgContext_AccessToken 현재 보안 컨텍스트에 대한 액세스 토큰에 대한 핸들을 반환합니다.
SecPkgContext_ApplicationProtocol 보안 컨텍스트의 애플리케이션 프로토콜에 대한 정보를 포함합니다.
SecPkgContext_AuthorityA SecPkgContext_Authority 구조체에는 인증 기관(사용 가능한 경우)의 이름이 포함됩니다. (ANSI)
SecPkgContext_AuthorityW SecPkgContext_Authority 구조체에는 인증 기관(사용 가능한 경우)의 이름이 포함됩니다. (유니코드)
SecPkgContext_AuthzID 보안 컨텍스트의 AuthzID에 대한 정보를 포함합니다.
SecPkgContext_Bindings 보안 컨텍스트에 대한 채널 바인딩 정보를 포함하는 구조를 지정합니다.
SecPkgContext_CipherInfo 암호 정보 구조입니다. 이는 QueryContextAttributes(Schannel) 함수에서 SECPKG_ATTR_CIPHER_INFO ulAttribute에 의해 반환됩니다.
SecPkgContext_ClientCreds QueryContextAttributes(CredSSP) 함수를 호출할 때 클라이언트 자격 증명을 지정합니다.
SecPkgContext_ClientSpecifiedTarget QueryContextAttributes(Digest) 함수를 호출할 때 초기 대상의 SPN(서비스 사용자 이름)을 지정합니다.
SecPkgContext_ConnectionInfo SecPkgContext_ConnectionInfo 구조에는 프로토콜 및 암호 정보가 포함됩니다. 이 구조체는 InitializeSecurityContext(Schannel) 함수에서 사용됩니다. 이 특성은 SSP(Schannel 보안 지원 공급자)에서만 지원됩니다.
SecPkgContext_CredentialNameA 자격 증명 이름 및 형식을 포함합니다.
SecPkgContext_CredentialNameW

보안 컨텍스트의 자격 증명 이름에 대한 정보입니다.
SecPkgContext_CredInfo 클라이언트 컨텍스트를 만드는 데 사용되는 자격 증명 유형을 지정합니다.
SecPkgContext_DceInfo SecPkgContext_DceInfo 구조에는 DCE 서비스에서 사용하는 권한 부여 데이터가 포함됩니다. QueryContextAttributes(일반) 함수는 이 구조를 사용합니다.
SecPkgContext_EapKeyBlock EAP TLS 인증 프로토콜에서 사용하는 키 데이터를 포함합니다.
SecPkgContext_EapPrfInfo Pseudorandom 함수(PRF)를 지정하고 EAP(Extensible Authentication Protocol) TLS(전송 계층 보안 프로토콜) 인증 프로토콜에서 사용하는 키 데이터를 추출합니다.
SecPkgContext_EarlyStart SecPkgContext_EarlyStart 구조에는 보안 컨텍스트에서 False Start 기능을 사용할지 여부에 대한 정보가 포함되어 있습니다.
SecPkgContext_Flags SecPkgContext_Flags 구조에는 현재 보안 컨텍스트의 플래그에 대한 정보가 포함됩니다. 이 구조체는 QueryContextAttributes(일반)에서 반환됩니다.
SecPkgContext_IssuerListInfoEx SecPkgContext_IssuerListInfoEx 구조에는 신뢰할 수 있는 CA(인증 기관) 목록이 있습니다.
SecPkgContext_KeyInfoA SecPkgContext_KeyInfo 구조에는 보안 컨텍스트에서 사용되는 세션 키에 대한 정보가 포함됩니다. (ANSI)
SecPkgContext_KeyInfoW SecPkgContext_KeyInfo 구조에는 보안 컨텍스트에서 사용되는 세션 키에 대한 정보가 포함됩니다. (유니코드)
SecPkgContext_KeyingMaterial SecPkgContext_KeyingMaterial 구조체입니다.
SecPkgContext_KeyingMaterialInfo

SecPkgContext_KeyingMaterialInfo 구조에는 보안 컨텍스트에서 내보낼 수 있는 키 지정 자료에 대한 정보가 포함됩니다.
SecPkgContext_LastClientTokenStatus InitializeSecurityContext 함수에 대한 최신 호출의 토큰이 클라이언트의 마지막 토큰인지 여부를 지정합니다.
SecPkgContext_Lifespan SecPkgContext_Lifespan 구조는 보안 컨텍스트의 수명을 나타냅니다. QueryContextAttributes(일반) 함수는 이 구조를 사용합니다.
SecPkgContext_LogoffTime 보안 컨텍스트의 로그오프 시간입니다.
SecPkgContext_NamesA SecPkgContext_Names 구조는 보안 컨텍스트와 연결된 사용자의 이름을 나타냅니다. QueryContextAttributes(일반) 함수는 이 구조를 사용합니다. (ANSI)
SecPkgContext_NamesW SecPkgContext_Names 구조는 보안 컨텍스트와 연결된 사용자의 이름을 나타냅니다. QueryContextAttributes(일반) 함수는 이 구조를 사용합니다. (유니코드)
SecPkgContext_NativeNamesA 클라이언트 및 서버 보안 주체 이름을 포함합니다.
SecPkgContext_NativeNamesW 보안 컨텍스트에서 클라이언트 및 서버의 네이티브 이름입니다.
SecPkgContext_NegoKeys 협상된 보안 패키지 키를 보유합니다.
SecPkgContext_NegoPackageInfo 협상된 애플리케이션 패키지에 대한 정보를 보유합니다.
SecPkgContext_NegoStatus 클라이언트 컨텍스트를 만들려는 마지막 시도의 오류 상태를 지정합니다.
SecPkgContext_NegotiatedTlsExtensions

SecPkgContext_NegotiatedTlsExtensions 구조에는 현재(D)TLS 연결에 대해 협상된 (D)TLS 확장에 대한 정보가 포함되어 있습니다.
SecPkgContext_NegotiationInfoA SecPkgContext_NegotiationInfo 구조에는 설정되거나 설정된 보안 패키지에 대한 정보가 포함되며, 보안 패키지를 설정하기 위한 협상 상태도 제공됩니다. (ANSI)
SecPkgContext_NegotiationInfoW SecPkgContext_NegotiationInfo 구조에는 설정되거나 설정된 보안 패키지에 대한 정보가 포함되며, 보안 패키지를 설정하기 위한 협상 상태도 제공됩니다. (유니코드)
SecPkgContext_PackageInfoA 애플리케이션 패키지 정보를 보유합니다.
SecPkgContext_PackageInfoW 패키지 정보를 보유합니다.
SecPkgContext_PasswordExpiry SecPkgContext_PasswordExpiry 구조에는 보안 컨텍스트에 사용되는 암호 또는 기타 자격 증명의 만료에 대한 정보가 포함되어 있습니다. 이 구조체는 QueryContextAttributes(일반)에서 반환됩니다.
SecPkgContext_ProtoInfoA SecPkgContext_ProtoInfo 구조에는 사용 중인 프로토콜에 대한 정보가 포함됩니다. (ANSI)
SecPkgContext_ProtoInfoW SecPkgContext_ProtoInfo 구조에는 사용 중인 프로토콜에 대한 정보가 포함됩니다. (유니코드)
SecPkgContext_SessionAppData 세션 컨텍스트에 대한 애플리케이션 데이터를 저장합니다.
SecPkgContext_SessionInfo 세션이 다시 연결인지 여부를 지정하고 세션을 식별하는 값을 검색합니다.
SecPkgContext_SessionKey SecPkgContext_SessionKey 구조에는 보안 컨텍스트에 사용되는 세션 키에 대한 정보가 포함되어 있습니다. 이 구조체는 QueryContextAttributes(일반) 함수에 의해 반환됩니다.
SecPkgContext_Sizes

SecPkgContext_Sizes 구조체는 메시지 지원 함수에 사용되는 중요한 구조체의 크기를 나타냅니다. QueryContextAttributes(일반) 함수는 이 구조를 사용합니다.
SecPkgContext_StreamSizes 메시지 지원 함수와 함께 사용할 스트림의 다양한 부분의 크기를 나타냅니다. QueryContextAttributes(일반) 함수는 이 구조를 사용합니다.
SecPkgContext_SubjectAttributes 보안 특성 정보를 반환합니다.
SecPkgContext_SupportedSignatures Schannel 연결에서 지원하는 서명 알고리즘을 지정합니다.
SecPkgContext_Target 대상 정보를 보유합니다.
SecPkgContext_TargetInformation 보안 컨텍스트에 사용되는 자격 증명에 대한 정보를 반환합니다.
SecPkgContext_UserFlags 사용자 플래그를 보유합니다.
SecPkgCredentials_Cert 인증서 자격 증명을 지정합니다. QueryCredentialsAttributes 함수는 이 구조를 사용합니다.
SecPkgCredentials_KdcProxySettingsW 자격 증명에 대한 Kerberos 프록시 설정을 지정합니다.
SecPkgCredentials_NamesA SecPkgCredentials_Names 구조체는 컨텍스트와 연결된 사용자의 이름을 보유합니다. QueryCredentialsAttributes 함수는 이 구조를 사용합니다. (ANSI)
SecPkgCredentials_NamesW SecPkgCredentials_Names 구조체는 컨텍스트와 연결된 사용자의 이름을 보유합니다. QueryCredentialsAttributes 함수는 이 구조를 사용합니다. (유니코드)
SecPkgCredentials_SSIProviderA

SecPkgCredentials_SSIProvider 구조체는 컨텍스트와 연결된 SSI 공급자 정보를 보유합니다. QueryCredentialsAttributes 함수는 이 구조를 사용합니다. (ANSI)
SecPkgCredentials_SSIProviderW SecPkgCredentials_SSIProvider 구조체는 컨텍스트와 연결된 SSI 공급자 정보를 보유합니다. QueryCredentialsAttributes 함수는 이 구조를 사용합니다. (유니코드)
SecPkgInfoA SecPkgInfo 구조체는 이름 및 기능과 같은 보안 패키지에 대한 일반적인 정보를 제공합니다. (ANSI)
SecPkgInfoW SecPkgInfo 구조체는 이름 및 기능과 같은 보안 패키지에 대한 일반적인 정보를 제공합니다. (유니코드)
SECURITY_CAPABILITIES 앱 컨테이너의 보안 기능을 정의합니다.
SECURITY_DESCRIPTOR 개체와 연결된 보안 정보를 포함합니다.
SECURITY_INTEGER SECURITY_INTEGER 숫자 값을 보유하는 구조체입니다. 다른 형식을 정의하는 데 사용됩니다.
SECURITY_LOGON_SESSION_DATA 로그온 세션에 대한 정보를 포함합니다. (SECURITY_LOGON_SESSION_DATA)
SECURITY_OBJECT 보안 개체 정보를 포함합니다.
SECURITY_PACKAGE_OPTIONS 보안 패키지에 대한 정보를 지정합니다.
SECURITY_QUALITY_OF_SERVICE 클라이언트 가장을 지원하는 데 사용되는 정보를 포함합니다.
SECURITY_STRING 커널 작업의 문자열 인터페이스로 사용되며 UNICODE_STRING 구조체의 복제본입니다.
SECURITY_USER_DATA

SecurityUserData 구조에는 보안 지원 공급자/인증 패키지의 사용자에 대한 정보가 포함되어 있습니다. 이 구조체는 SpGetUserInfo 함수에서 사용됩니다.
SecurityFunctionTableA SecurityFunctionTable 구조체는 SSPI에 정의된 함수에 대한 포인터를 포함하는 디스패치 테이블입니다. (ANSI)
SecurityFunctionTableW SecurityFunctionTable 구조체는 SSPI에 정의된 함수에 대한 포인터를 포함하는 디스패치 테이블입니다. (유니코드)
SERVICE_CONTROL_STATUS_REASON_PARAMSA 서비스 제어 매개 변수를 포함합니다. (ANSI)
SERVICE_CONTROL_STATUS_REASON_PARAMSW 서비스 제어 매개 변수를 포함합니다. (유니코드)
SERVICE_DELAYED_AUTO_START_INFO 자동 시작 서비스의 지연된 자동 시작 설정을 포함합니다.
SERVICE_DESCRIPTIONA 서비스 설명을 포함합니다. (ANSI)
SERVICE_DESCRIPTIONW 서비스 설명을 포함합니다. (유니코드)
SERVICE_FAILURE_ACTIONS_FLAG 서비스의 오류 작업 플래그 설정을 포함합니다. 이 설정은 실패 작업을 실행할 시기를 결정합니다.
SERVICE_FAILURE_ACTIONSA 서비스 컨트롤러가 서비스의 각 실패에 대해 수행해야 하는 작업을 나타냅니다. 서비스가 서비스 컨트롤러에 SERVICE_STOPPED 상태를 보고하지 않고 종료될 때 실패한 것으로 간주됩니다. (ANSI)
SERVICE_FAILURE_ACTIONSW 서비스 컨트롤러가 서비스의 각 실패에 대해 수행해야 하는 작업을 나타냅니다. 서비스가 서비스 컨트롤러에 SERVICE_STOPPED 상태를 보고하지 않고 종료될 때 실패한 것으로 간주됩니다. (유니코드)
SERVICE_LAUNCH_PROTECTED_INFO

서비스 보호 유형을 나타냅니다.
SERVICE_NOTIFY_2A 서비스 상태 알림 정보를 나타냅니다. (ANSI)
SERVICE_NOTIFY_2W 서비스 상태 알림 정보를 나타냅니다. (유니코드)
SERVICE_PREFERRED_NODE_INFO 서비스를 실행할 기본 노드를 나타냅니다.
SERVICE_PRESHUTDOWN_INFO preshutdown 설정을 포함합니다.
SERVICE_REQUIRED_PRIVILEGES_INFOA 서비스에 필요한 권한을 나타냅니다. (ANSI)
SERVICE_REQUIRED_PRIVILEGES_INFOW 서비스에 필요한 권한을 나타냅니다. (유니코드)
SERVICE_SID_INFO SID(서비스 보안 식별자)를 나타냅니다.
SERVICE_STATUS 서비스에 대한 상태 정보를 포함합니다.
SERVICE_STATUS_PROCESS 서비스에 대한 프로세스 상태 정보를 포함합니다. ControlServiceEx, EnumServicesStatusEx, NotifyServiceStatusChange 및 QueryServiceStatusEx 함수는 이 구조를 사용합니다.
SERVICE_TABLE_ENTRYA 호출 프로세스에서 실행할 수 있는 서비스에 대한 ServiceMain 함수를 지정합니다. StartServiceCtrlDispatcher 함수에서 사용됩니다. (ANSI)
SERVICE_TABLE_ENTRYW 호출 프로세스에서 실행할 수 있는 서비스에 대한 ServiceMain 함수를 지정합니다. StartServiceCtrlDispatcher 함수에서 사용됩니다. (유니코드)

SERVICE_TIMECHANGE_INFO 시스템 시간 변경 설정을 포함합니다.
SERVICE_TRIGGER 서비스 트리거 이벤트를 나타냅니다. 이 구조체는 SERVICE_TRIGGER_INFO 구조체에서 사용됩니다.
SERVICE_TRIGGER_INFO 서비스에 대한 트리거 이벤트 정보를 포함합니다. 이 구조체는 ChangeServiceConfig2 및 QueryServiceConfig2 함수에서 사용됩니다.
SERVICE_TRIGGER_SPECIFIC_DATA_ITEM 서비스 트리거 이벤트에 대한 트리거별 데이터를 포함합니다.
SI_ACCESS 보안 개체에 대한 액세스 권한 또는 기본 액세스 마스크에 대한 정보를 포함합니다.
SI_INHERIT_TYPE 자식 개체에서 ACE(액세스 제어 항목)를 상속하는 방법에 대한 정보를 포함합니다.
SI_OBJECT_INFO 액세스 제어 편집기를 초기화하는 데 사용됩니다.
SID 사용자 또는 그룹을 고유하게 식별하는 데 사용됩니다.
SID_AND_ATTRIBUTES SID(보안 식별자) 및 해당 특성을 나타냅니다.
SID_AND_ATTRIBUTES_HASH 지정된 SID(보안 식별자) 배열에 대한 해시 값을 지정합니다.
SID_IDENTIFIER_AUTHORITY SID(보안 식별자)의 최상위 기관을 나타냅니다.
SID_INFO ISecurityInformation2::LookupSids에서 반환된 SID 구조에 해당하는 일반 이름 목록을 포함합니다.

SID_INFO_LIST SID_INFO 구조의 목록을 포함합니다.
SIP_ADD_NEWPROVIDER SIP(주체 인터페이스 패키지)를 정의합니다. 이 구조체는 CryptSIPAddProvider 함수에서 사용됩니다.
SIP_CAP_SET_V2 SIP_CAP_SET_V2 구조는 SIP(주체 인터페이스 패키지)의 기능을 정의합니다. (SIP_CAP_SET_V2 구조)
SIP_CAP_SET_V3 SIP_CAP_SET_V3 구조는 SIP(주체 인터페이스 패키지)의 기능을 정의합니다. (SIP_CAP_SET_V3 구조)
SIP_DISPATCH_INFO 애플리케이션이 SIP(주체 인터페이스 패키지) 작업을 수행하는 데 사용하는 CryptSIPLoad 함수에 의해 할당된 함수 포인터 집합을 포함합니다.
SIP_INDIRECT_DATA 해시된 주체 정보의 다이제스트를 포함합니다.
SIP_SUBJECTINFO 주체 정보 데이터를 SIP(주체 인터페이스 패키지) API로 지정합니다.
SL_ACTIVATION_INFO_HEADER 제품 활성화 정보를 지정합니다.
SL_AD_ACTIVATION_INFO 라이선스의 정품 또는 Active Directory 전화 활성화에 사용되는 정보를 지정합니다.
SL_LICENSING_STATUS 라이선스 상태를 나타냅니다. (SL_LICENSING_STATUS)
SL_NONGENUINE_UI_OPTIONS SLIsGenuineLocal 함수가 설치가 정품이 아님을 나타내는 경우 대화 상자를 표시하는 애플리케이션을 지정합니다.
SPC_INDIRECT_DATA_CONTENT Authenticode 서명에서 서명된 파일의 다이제스트 및 기타 특성을 저장하는 데 사용됩니다.

SR_SECURITY_DESCRIPTOR SR_SECURITY_DESCRIPTOR 구조에는 사용자의 보안 권한에 대한 정보가 포함됩니다.
SSL_F12_EXTRA_CERT_CHAIN_POLICY_STATUS SSL_F12_EXTRA_CERT_CHAIN_POLICY_STATUS 구조는 체인의 인증서에 약한 암호화가 있는지 확인하고 타사 루트 인증서가 Microsoft 루트 프로그램 요구 사항을 준수하는지 확인합니다.
SYSTEM_ALARM_ACE SYSTEM_ALARM_ACE 구조체는 나중에 사용하도록 예약되어 있습니다.
SYSTEM_ALARM_CALLBACK_ACE SYSTEM_ALARM_CALLBACK_ACE 구조체는 나중에 사용하도록 예약되어 있습니다.
SYSTEM_ALARM_CALLBACK_OBJECT_ACE SYSTEM_ALARM_CALLBACK_OBJECT_ACE 구조체는 나중에 사용하도록 예약되어 있습니다.
SYSTEM_ALARM_OBJECT_ACE SYSTEM_ALARM_OBJECT_ACE 구조체는 나중에 사용하도록 예약되어 있습니다.
SYSTEM_AUDIT_ACE 시스템 수준 알림을 유발하는 액세스 유형을 지정하는 SACL(시스템 액세스 제어 목록)에 대한 ACE(액세스 제어 항목)를 정의합니다.
SYSTEM_AUDIT_CALLBACK_ACE SYSTEM_AUDIT_CALLBACK_ACE 구조는 시스템 수준 알림을 유발하는 액세스 유형을 지정하는 시스템 액세스 제어 목록에 대한 액세스 제어 항목을 정의합니다.
SYSTEM_AUDIT_CALLBACK_OBJECT_ACE SYSTEM_AUDIT_CALLBACK_OBJECT_ACE 구조는 시스템 액세스 제어 목록에 대한 액세스 제어 항목을 정의합니다.
SYSTEM_AUDIT_OBJECT_ACE SACL(시스템 액세스 제어 목록)에 대한 ACE(액세스 제어 항목)를 정의합니다.
SYSTEM_MANDATORY_LABEL_ACE 보안 개체에 대한 필수 액세스 수준 및 정책을 지정하는 SACL(시스템 액세스 제어 목록)에 대한 ACE(액세스 제어 항목)를 정의합니다.

SYSTEM_RESOURCE_ATTRIBUTE_ACE 보안 개체의 시스템 리소스 특성을 지정하는 SACL(시스템 액세스 제어 목록)에 대한 ACE(액세스 제어 항목)를 정의합니다.
SYSTEM_SCOPED_POLICY_ID_ACE 보안 개체의 범위가 지정된 정책 식별자를 지정하는 SACL(시스템 액세스 제어 목록)에 대한 ACE(액세스 제어 항목)를 정의합니다.
TLS_PARAMETERS TLS 매개 변수 제한을 나타냅니다.
TOKEN_ACCESS_INFORMATION 액세스 검사를 수행하는 데 필요한 토큰의 모든 정보를 지정합니다.
TOKEN_APPCONTAINER_INFORMATION 앱 컨테이너에 필요한 토큰의 모든 정보를 지정합니다.
TOKEN_AUDIT_POLICY 토큰에 대한 사용자별 감사 정책을 지정합니다.
TOKEN_CONTROL 액세스 토큰을 식별하는 정보를 포함합니다.
TOKEN_DEFAULT_DACL DACL(임의 액세스 제어 목록)을 지정합니다.
TOKEN_DEVICE_CLAIMS 토큰에 대한 디바이스 클레임을 정의합니다.
TOKEN_ELEVATION 토큰에 상승된 권한이 있는지 여부를 나타냅니다.
TOKEN_GROUPS 액세스 토큰의 SID(그룹 보안 식별자)에 대한 정보를 포함합니다.
TOKEN_GROUPS_AND_PRIVILEGES 그룹 SID(보안 식별자) 및 액세스 토큰의 권한에 대한 정보를 포함합니다.

TOKEN_LINKED_TOKEN 토큰에 대한 핸들을 포함합니다. 이 토큰은 GetTokenInformation 함수에서 쿼리하거나 SetTokenInformation 함수에 의해 설정된 토큰에 연결됩니다.
TOKEN_MANDATORY_LABEL 토큰의 필수 무결성 수준을 지정합니다.
TOKEN_MANDATORY_POLICY 토큰에 대한 필수 무결성 정책을 지정합니다.
TOKEN_ORIGIN 로그온 세션의 원본에 대한 정보를 포함합니다.
TOKEN_OWNER 새로 만든 개체에 적용할 기본 SID(소유자 보안 식별자)를 포함합니다.
TOKEN_PRIMARY_GROUP 액세스 토큰에 대한 SID(그룹 보안 식별자)를 지정합니다.
TOKEN_PRIVILEGES 액세스 토큰에 대한 권한 집합에 대한 정보를 포함합니다.
TOKEN_SOURCE 액세스 토큰의 원본을 식별합니다.
TOKEN_STATISTICS 액세스 토큰에 대한 정보를 포함합니다.
TOKEN_USER 액세스 토큰과 연결된 사용자를 식별합니다.
TOKEN_USER_CLAIMS 토큰에 대한 사용자 클레임을 정의합니다.
TOKENBINDING_IDENTIFIER 토큰 바인딩 메시지 교환에서 발생하는 토큰 바인딩 식별자를 나타내는 정보를 포함합니다.

TOKENBINDING_KEY_TYPES 클라이언트 디바이스 또는 서버에서 지원하는 토큰 바인딩 키 형식의 모든 조합을 포함합니다.
TOKENBINDING_RESULT_DATA 토큰 바인딩을 생성하거나 토큰 바인딩 메시지에서 토큰 바인딩 중 하나를 확인한 결과에 대한 데이터를 포함합니다.
TOKENBINDING_RESULT_LIST TokenBindingVerifyMessage가 확인한 각 토큰 바인딩에 대한 결과를 포함합니다.
TRUSTED_DOMAIN_AUTH_INFORMATION TRUSTED_DOMAIN_AUTH_INFORMATION 구조는 신뢰할 수 있는 도메인에 대한 인증 정보를 검색하는 데 사용됩니다. LsaQueryTrustedDomainInfo 함수는 InformationClass 매개 변수가 TrustedDomainAuthInformation으로 설정된 경우 이 구조를 사용합니다.
TRUSTED_DOMAIN_FULL_INFORMATION 신뢰할 수 있는 도메인에 대한 전체 정보를 검색하는 데 사용됩니다.
TRUSTED_DOMAIN_INFORMATION_EX 신뢰할 수 있는 도메인에 대한 확장 정보를 검색하는 데 사용됩니다.
TRUSTED_DOMAIN_NAME_INFO 신뢰할 수 있는 도메인의 이름을 쿼리하거나 설정하는 데 사용됩니다.
TRUSTED_PASSWORD_INFO TRUSTED_PASSWORD_INFO 구조는 신뢰할 수 있는 도메인에 대한 암호를 쿼리하거나 설정하는 데 사용됩니다.
TRUSTED_POSIX_OFFSET_INFO Posix 사용자 및 그룹 식별자를 생성하는 데 사용되는 값을 쿼리하거나 설정하는 데 사용됩니다.
TRUSTEE_A ACE(액세스 제어 항목)가 적용되는 사용자 계정, 그룹 계정 또는 로그온 세션을 식별합니다. (ANSI)
TRUSTEE_W ACE(액세스 제어 항목)가 적용되는 사용자 계정, 그룹 계정 또는 로그온 세션을 식별합니다. (유니코드)

UNICODE_STRING 다양한 LSA(로컬 보안 기관) 함수에서 유니코드 문자열을 지정하는 데 사용됩니다.
UNIVERSAL_NAME_INFOA UNIVERSAL_NAME_INFO 구조체에는 범용 이름의 UNC 형식에 대한 정보가 포함됩니다. NPGetUniversalName 함수에서 사용됩니다. (ANSI)
UNIVERSAL_NAME_INFOW UNIVERSAL_NAME_INFO 구조체에는 범용 이름의 UNC 형식에 대한 정보가 포함됩니다. NPGetUniversalName 함수에서 사용됩니다. (유니코드)
USER_ALL_INFORMATION 세션 사용자에 대한 정보를 포함합니다.
USERNAME_TARGET_CREDENTIAL_INFO USERNAME_TARGET_CREDENTIAL_INFO 구조체에는 자격 증명에 대한 참조가 포함됩니다.
WIN_CERTIFICATE 이 구조는 실행 파일 확인에 사용되는 서명을 캡슐화합니다.
WINTRUST_BLOB_INFO WinVerifyTrust를 호출하여 메모리 BLOB을 확인하는 데 사용됩니다.
WINTRUST_CATALOG_INFO WINTRUST_CATALOG_INFO 구조는 WinVerifyTrust를 호출하여 Microsoft 카탈로그의 멤버를 확인하는 데 사용됩니다.
WINTRUST_CERT_INFO WinVerifyTrust를 호출하여 CERT_CONTEXT 확인할 때 사용됩니다.
WINTRUST_DATA WinVerifyTrust를 호출하여 필요한 정보를 트러스트 공급자에 전달할 때 사용됩니다.
WINTRUST_FILE_INFO WINTRUST_FILE_INFO 구조는 WinVerifyTrust를 호출하여 개별 파일을 확인할 때 사용됩니다.
WINTRUST_SGNR_INFO

WinVerifyTrust를 호출하여 CMSG_SIGNER_INFO 구조를 확인할 때 사용됩니다.
WINTRUST_SIGNATURE_SETTINGS 파일의 서명을 지정하는 데 사용할 수 있습니다.
WLX_CLIENT_CREDENTIALS_INFO_V1_0 WlxQueryClientCredentials 또는 WlxQueryInetConnectorCredentials 호출에서 반환되는 클라이언트 자격 증명을 포함합니다.
WLX_CLIENT_CREDENTIALS_INFO_V2_0 WlxQueryTsLogonCredentials 호출에서 반환된 클라이언트 자격 증명을 포함합니다.
WLX_CONSOLESWITCH_CREDENTIALS_INFO_V1_0 WlxGetConsoleSwitchCredentials 호출에서 반환된 클라이언트 자격 증명을 포함합니다.
WLX_DESKTOP GINA DLL과 Winlogon 간에 데스크톱 정보를 전달하는 데 사용됩니다.
WLX_DISPATCH_VERSION_1_0 WlxInitialize 호출에서 GINA DLL에 전달된 Winlogon 버전 1.0 함수 디스패치 테이블의 형식을 정의합니다.
WLX_DISPATCH_VERSION_1_1 WlxInitialize 호출에서 GINA DLL에 전달된 Winlogon 버전 1.1 함수 디스패치의 형식을 정의합니다.
WLX_DISPATCH_VERSION_1_2 WlxInitialize 호출에서 GINA DLL에 전달된 Winlogon 버전 1.2 함수 디스패치 테이블의 형식을 정의합니다.
WLX_DISPATCH_VERSION_1_3 WlxInitialize 호출에서 GINA DLL에 전달된 Winlogon 버전 1.3 함수 디스패치 테이블의 형식을 정의합니다.
WLX_DISPATCH_VERSION_1_4 WlxInitialize 호출에서 GINA DLL에 전달된 Winlogon 버전 1.4 함수 디스패치 테이블의 형식을 정의합니다.
WLX_MPR_NOTIFY_INFO 네트워크 공급자에게 식별 및 인증 정보를 제공합니다.
WLX_NOTIFICATION_INFO 이 구조체는 Winlogon 이벤트에 대한 정보를 저장합니다.

WLX_PROFILE_V1_0 초기 환경을 설정하는 데 사용되는 정보를 포함합니다.
WLX_PROFILE_V2_0 WLX_PROFILE_V1_0 제공한 정보 외에 프로필 정보를 포함합니다.
WLX_TERMINAL_SERVICES_DATA 터미널 서비스 사용자 구성 정보를 GINA에 제공하는 데 사용됩니다.
X509Certificate X.509 인증서를 나타냅니다.

Last updated on 2025. 07. 22.

센서 API

아티클2023. 08. 22.

센서 API 기술 개요입니다.

센서 API 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 센서 API

열거형

자력계Accuracy 자력계의 정확도를 지정합니다.
SensorConnectionType 센서 디바이스 연결 유형을 정의합니다.
SensorState 센서에 대한 가능한 작동 상태를 정의합니다.

Functions

추가 컬렉션에 센서를 추가합니다.
CheckLocationCapability 지정된 스레드의 Windows 스토어 앱 위치 기능을 가져옵니다.
지우기 센서 컬렉션을 비웁니다.
GetAt

컬렉션의 지정된 인덱스에서 센서를 검색합니다.
GetCategory 센서 범주의 식별자를 검색합니다.
GetCount 컬렉션의 센서 수를 검색합니다.
GetData 가장 최근의 센서 데이터 보고서를 검색합니다.
GetEventInterest 현재 이벤트 관심 설정을 검색합니다.
GetFriendlyName 사용자가 볼 수 있는 센서 이름을 검색합니다.
GetGlobalLocationPermission 사용자가 위치 설정을 변경할 수 있는 시스템 설정의 상태 가져옵니다.
GetID 센서의 고유 식별자를 검색합니다.
GetProperties 여러 센서 속성을 검색합니다.
GetProperty 속성 값을 검색합니다.
GetSensorByID 지정된 센서에 대한 포인터를 검색합니다.
GetSensorsByCategory 지정된 범주와 연결된 모든 센서가 포함된 컬렉션을 검색합니다.
GetSensorsByType 지정된 형식과 연결된 모든 센서가 포함된 컬렉션을 검색합니다.

GetSensorValue 데이터 보고서에서 단일 데이터 필드 값을 검색합니다.
GetSensorValues 데이터 필드 값의 컬렉션을 검색합니다.
GetState 센서의 현재 작동 상태를 검색합니다.
GetSupportedDataFields 센서가 제공할 수 있는 데이터 필드를 나타내는 PROPERTYKEY 집합을 검색합니다.
GetTimestamp 데이터 보고서를 만든 시간을 검색합니다.
GetType 센서 유형 ID를 검색합니다.
OnDataUpdated 센서 이벤트 데이터를 제공합니다.
Onevent 사용자 지정 이벤트 알림을 제공합니다.
OnLeave 센서 디바이스가 더 이상 연결되지 않는다는 알림을 제공합니다.
OnSensorEnter 센서 디바이스가 연결되면 알림을 제공합니다.
OnStateChanged 센서 상태가 변경되었다는 알림을 제공합니다.
제거 컬렉션에서 센서를 제거합니다. 센서는 제거할 ISensor 인터페이스에 대한 포인터로 지정됩니다.
RemoveByID

컬렉션에서 센서를 제거합니다. 제거할 센서는 ID로 지정됩니다.
RequestPermissions 센서 데이터에 액세스할 수 있는 사용자 권한을 요청하는 시스템 대화 상자를 엽니다.
SetEventInterest 받을 센서 이벤트 목록을 지정합니다.
SetEventSink 센서 이벤트 알림을 받을 인터페이스를 지정합니다.
SetEventSink 센서 관리자 이벤트 알림을 받을 인터페이스를 지정합니다.
SetProperties 센서 속성을 지정합니다.
SupportsDataField 센서가 지정된 데이터 필드를 지원하는지 여부를 나타냅니다.
SupportsEvent 센서가 지정된 이벤트를 지원하는지 여부를 나타냅니다.

인터페이스

ILocationPermissions 사용자가 위치 설정을 변경할 수 있는 시스템 설정의 상태 제공합니다.
ISensor 센서를 나타냅니다.
ISensorCollection 컴퓨터에 연결된 모든 센서와 같은 센서 컬렉션을 나타냅니다.
ISensorDataReport

센서 데이터 보고서를 나타냅니다. 센서 데이터 보고서에는 센서에서 생성된 데이터 필드 값과 데이터 보고서가 만들어진 시기를 나타내는 타임스탬프를 포함합니다.
ISensorEvents 센서 이벤트를 수신하려는 경우 구현해야 하는 콜백 인터페이스입니다.
ISensorManager 사용 가능한 센서를 검색 및 검색하는 메서드와 센서 관리자 이벤트를 요청하는 방법을 제공합니다.
ISensorManagerEvents 센서 관리자 이벤트를 수신하기 위한 콜백 인터페이스입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

센서 디바이스 드라이버 참조

> **Important**
>
> 대신 **UWP 센서 API** 를 사용합니다.
>
> COM 기반 센서 API는 더 이상 사용되지 않으며 새 애플리케이션에서 사용해서는 안 됩니다. 추가 기능 또는 향상된 기능은 계획되지 않으며 지원이 제한됩니다.

센서 디바이스 드라이버 참조 기술에 대한 개요입니다.

센서 디바이스 드라이버 참조 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 센서 디바이스 드라이버 참조

열거 목록

테이블 확장

LOCATION_DESIRED_ACCURACY LOCATION_DESIRED_ACCURACY 열거형 형식은 SENSOR_PROPERTY_LOCATION_DESIRED_ACCURACY 속성에 대한 값을 정의합니다.

Last updated on 2025. 10. 06.

직렬 컨트롤러 드라이버 참조

아티클2023. 08. 22.

직렬 컨트롤러 드라이버 참조 기술 개요입니다.

직렬 컨트롤러 드라이버 참조를 개발하려면 다음 헤더가 필요합니다.

- msports.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 직렬 컨트롤러 드라이버 참조

Functions

ComDBClaimNextFreePort ComDBClaimNextFreePort는 아직 사용되지 않는 가장 낮은 COM 포트 번호를 반환합니다.
ComDBClaimPort ComDBClaimPort는 사용되지 않는 COM 포트 번호를 COM 포트 데이터베이스에서 "사용 중"으로 기록합니다.
ComDBClose ComDBClose는 COM 포트 데이터베이스에 대한 핸들을 닫습니다.
ComDBGetCurrentPortUsage ComDBGetCurrentPortUsage는 COM 포트 데이터베이스에서 현재 "사용 중"으로 기록된 COM 포트 번호에 대한 정보를 반환합니다.
ComDBOpen ComDBOpen은 COM 포트 데이터베이스에 대한 핸들을 반환합니다.
ComDBReleasePort ComDBReleasePort는 COM 포트 데이터베이스에서 COM 포트 번호를 해제합니다.
ComDBResizeDatabase ComDBResizeDatabase는 COM 포트 데이터베이스의 크기를 조정합니다.

SerialDisplayAdvancedSettings SerialDisplayAdvancedSettings는 지정된 COM 포트 디바이스에 대해 시스템에서 제공하는 고급 설정 대화 상자를 표시합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

설정 관리 인프라(SMI)

아티클2023. 03. 09.

SMI(설정 관리 인프라) 기술에 대한 개요입니다.

SMI(설정 관리 인프라)를 개발하려면 다음 헤더가 필요합니다.

- wcmconfig.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 설정 관리 인프라(SMI)

열거형

WcmDataType ISettingsItem::GetDataType 메서드에서 반환된 데이터 형식을 열거합니다.
WcmNamespaceAccess ISettingsEngine::GetNamespace 메서드에 전달되어 네임스페이스에 액세스하는 방법을 선택하는 옵션을 설명합니다.
WcmNamespaceEnumerationFlags 열거형 플래그의 형식을 설명합니다.
WcmRestrictionFacets ISettingsItem::GetRestrictionFacets 메서드에서 반환할 수 있는 패싯 값을 열거합니다.
WcmSettingType ISettingsItem::GetSettingType 메서드에서 반환되는 설정 형식을 설명하고 호출 ISettingsItem 인터페이스에 대한 개체 모델 형식을 정의합니다.
WcmTargetMode 다양한 대상 모드를 열거합니다.
WcmUserStatus 사용자의 상태를 설명합니다.

Functions

ApplySettingsContext 설정 컨텍스트를 적용합니다.
특성 특성의 사전을 가져옵니다.
Children 이 항목에 해당하는 자식 항목의 사전을 가져옵니다.
CreateListElement 새 목록 요소를 만듭니다.
CreateSettingByPath 경로에 지정된 설정 개체를 만듭니다.
CreateSettingByPath 경로로 지정된 설정 개체를 만듭니다.
CreateSettingsContext 설정 컨텍스트를 만듭니다.
CreateSettingsIdentity 빈 설정 ID를 만듭니다.
CreateTargetInfo 빈 대상을 만듭니다.
Current 열거자의 현재 위치에서 항목을 검색합니다.
역직렬화 이 컨텍스트에 제공되는 스트림의 데이터를 역직렬화합니다.

ExpandTarget 위치 문자열을 확장하여 오프라인 설치 위치를 나타냅니다. (ITargetInfo.ExpandTarget)
ExpandTargetPath 위치 문자열을 확장하여 오프라인 설치 위치를 나타냅니다. (ITargetInfo.ExpandTargetPath)
GetAttribute 네임스페이스 ID에 대한 ID 특성을 가져옵니다.
GetAttribute 이름을 지정하여 특성의 값을 가져옵니다.
GetAttribute 네임스페이스의 특성 값을 가져옵니다.
Getchild 지정된 이름을 포함하는 자식 항목을 가져옵니다.
GetColumn 오류가 발생한 열 번호를 반환합니다.
GetContextDescription 오류를 둘러싸는 컨텍스트에 대한 설명을 반환합니다.
GetDataType 항목의 형식 정보를 가져옵니다.
GetDescription 오류에 대한 설명을 반환합니다.
GetEnumerator 오프라인 속성 컬렉션에 액세스하는 데 사용되는 열거자를 가져옵니다.
GetErrorCode HRESULT 오류 코드 값을 반환합니다.

GetErrorDescription 반환된 HRESULT 코드에 대한 문자 메시지를 검색합니다.
GetFlags 네임스페이스 ID에 대한 플래그를 반환합니다.
GetIdentity 네임스페이스의 ID를 가져옵니다.
GetKeyValue 이미지에 이미 있는 목록(예: DNS, http 설정 및 사용자 계정 정보)에 대한 키 값을 추출합니다.
GetLine 오류가 발생한 줄 번호를 반환합니다.
GetListKeyInformation 이 항목의 목록 정보를 가져옵니다.
GetName 항목의 이름을 가져옵니다.
GetNamespace ISettingsIdentity 매개 변수에 지정된 대로 기존 네임스페이스를 엽니다.
GetNamespaces 컨텍스트에 있는 네임스페이스를 가져옵니다.
GetNamespaces 설치된 네임스페이스에 대한 열거자를 반환합니다.
GetPath 항목의 경로를 가져옵니다.
Getproperty 오프라인 설치 위치에 대한 속성 값을 가져옵니다.

GetRestriction 지정된 제한에 대한 정보를 가져옵니다.
GetRestrictionFacets 이 항목에 대해 정의된 제한을 가져옵니다.
GetSchemaHiveLocation 스키마 하이브의 위치를 가져옵니다.
GetSchemaHiveMountName 스키마 하이브의 탑재 위치 이름을 가져옵니다.
GetSettingByPath 지정된 경로에 따라 설정을 가져옵니다.
GetSettingByPath 경로에 지정된 설정 개체를 가져옵니다.
GetSettingType 항목의 설정 유형을 가져옵니다.
GetSource 오류가 발생한 파일 또는 경로를 반환합니다.
GetStoredSettings 지정된 네임스페이스에 대한 컨텍스트에서 저장된 설정 변경 내용을 가져옵니다.
GetStoreStatus 스키마 저장소의 상태를 가져옵니다.
GetTargetID 현재 대상과 연결된 고유 식별자를 가져옵니다.
GetTargetInfo 엔진의 현재 오프라인 대상을 가져옵니다.

GetTargetMode 현재 대상 모드를 가져옵니다.
GetTargetProcessorArchitecture 현재 대상과 연결된 프로세서 아키텍처를 가져옵니다.
GetTemporaryStoreLocation 현재 임시 저장소 위치를 가져옵니다.
GetUserData 사용자 정의 데이터를 가져옵니다.
GetValue 항목에서 현재 값을 가져옵니다.
GetValueRaw 현재 항목의 값을 바이트 배열로 가져옵니다.
HasChild 현재 항목에 자식 항목이 있는지 여부를 확인합니다.
LoadModule 오프라인 설치 위치에서 모듈을 로드합니다.
LoadStore 스키마 저장소 하이브를 초기화하고 로드합니다.
MoveNext 사용 가능한 경우 현재 위치를 열거자의 다음 항목으로 이동합니다.
RegisterNamespace 스트림에서 네임스페이스를 등록합니다.
RemoveListElement 제공된 이름의 기존 목록 요소를 제거합니다.

RemoveSettingByPath 경로에 지정된 설정 개체를 제거합니다.
RemoveSettingByPath 경로에 지정된 설정 개체를 제거합니다.
재설정 열거자의 상태를 초기화된 상태로 다시 설정합니다. 열거형의 첫 번째 위치에 현재 포인터를 설정하려면 열거자에서 IItemEnumerator::MoveNext를 호출하여 IItemEnumerator::Reset을 즉시 따라야 합니다.
RevertSetting 네임스페이스의 설정을 되돌려 줍니다.
저장 설정 네임스페이스를 영구 및 표시로 업데이트.
직렬화 이 컨텍스트의 데이터를 제공된 스트림으로 직렬화합니다.
SetAttribute 네임스페이스 ID에 대한 ID 특성을 설정합니다.
SetFlags 네임스페이스 ID의 ID 플래그를 설정합니다.
SetModulePath 오프라인 설치 위치의 모듈 경로를 설정합니다.
SetProperty 오프라인 설치 위치에 대한 속성 값을 설정합니다.
SetSchemaHiveLocation 스키마 하이브의 위치를 설정합니다.
SetSchemaHiveMountName 스키마 하이브의 탑재 위치 이름을 설정합니다.

SetTargetID 현재 대상과 연결된 고유 식별자를 설정합니다.
SetTargetInfo 엔진의 현재 오프라인 대상을 설정합니다.
SetTargetMode 대상 모드를 설정합니다.
SetTargetProcessorArchitecture 현재 대상과 연결된 프로세서 아키텍처를 설정합니다.
SetTemporaryStoreLocation 현재 임시 저장소 위치를 설정합니다.
설정 네임스페이스의 최상위 설정에 대한 열거자를 검색합니다.
SetUserData 사용자 정의 데이터를 설정합니다.
SetValue 항목의 값을 설정합니다.
SetValueRaw 원시 형식으로 데이터를 제공하여 현재 항목의 값을 설정합니다.
SetWow64Context wow64 리디렉션에 대한 불투명 컨텍스트 개체를 설정합니다.
TranslateWow64 wow64 리디렉션에 대한 경로를 변환합니다.
UnloadStore 스키마 저장소 하이브를 언로드하고 리소스를 해제합니다.

UnregisterNamespace 기존 네임스페이스를 등록 취소합니다.

인터페이스

IItemEnumerator 설정 및 특성 컬렉션의 항목을 열거합니다.
ISettingsContext 다른 SMI API를 통해 변경된 설정을 저장하는 데 사용되는 백업 저장소에 대한 인터페이스이며 표현에서 직렬화 및 역직렬화하는 작업을 제공합니다.
ISettingsEngine 네임스페이스를 열고 열 방법을 제어하기 위한 중앙 인터페이스입니다.
ISettingsIdentity 열거나 사용할 네임스페이스를 식별합니다.
ISettingsItem 설정 트리를 탐색하고, 특정 설정에 대한 메타데이터를 검색하고, 해당 값을 검색하거나 수정합니다.
ISettingsNamespace 네임스페이스 인스턴스에 대한 설정을 설정하고, 검색하고, 유효성을 검사하고, 변경 내용을 저장하는 작업을 수행합니다.
ISettingsResult 다양한 작업에서 반환된 오류 및 경고에 대한 코드와 설명을 검색합니다.
ITargetInfo 오프라인 대상 정보, 특히 파일 및 레지스트리 위치와 wow64 정보를 정의합니다.

Simple Network Management Protocol

아티클2023. 03. 16.

간단한 네트워크 관리 프로토콜 기술 개요입니다.

간단한 네트워크 관리 프로토콜을 개발하려면 다음 헤더가 필요합니다.

- mgmtapi.h
- snmp.h
- winsnmp.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Simple Network Management Protocol

Functions

SNMPAPI_CALLBACK Microsoft WinSNMP 구현은 SNMPAPI_CALLBACK 함수를 호출하여 WinSNMP 세션에 SNMP 메시지 또는 비동기 이벤트를 사용할 수 있음을 알립니다.
SnmpCancelMsg WinSNMP 애플리케이션은 SnmpCancelMsg 함수를 호출하여 Microsoft WinSNMP 구현에서 SNMP 요청 메시지에 대한 재전송 시도 및 시간 제한 알림을 취소하도록 요청합니다.
SnmpCleanup SnmpCleanup 함수는 호출하는 WinSNMP 애플리케이션에 더 이상 구현 서비스가 필요하지 않음을 Microsoft WinSNMP 구현에 알릴 수 있습니다.
SnmpCleanupEx SnmpCleanupEx 함수는 WinSNMP(Windows SNMP) 애플리케이션 내에서 SnmpStartup 또는 SnmpStartupEx에 대한 미해결 호출이 없는 경우 정리를 수행합니다.
SnmpClose SnmpClose 함수를 사용하면 Microsoft WinSNMP 구현에서 WinSNMP 세션과 연결된 메모리, 리소스 및 데이터 구조의 할당을 취소할 수 있습니다.

SnmpContextToStr WinSNMP SnmpContextToStr 함수는 관리되는 개체 리소스 집합인 SNMP 컨텍스트를 식별하는 문자열을 반환합니다. 이 함수는 smiOCTETS 구조체의 문자열을 반환합니다.
SnmpCountVbl WinSNMP 애플리케이션은 WinSNMP SnmpCountVbl 함수를 호출하여 지정된 변수 바인딩 목록의 변수 바인딩 항목을 열거합니다.
SnmpCreatePdu WinSNMP SnmpCreatePdu 함수는 SNMP PDU(프로토콜 데이터 단위)를 만들고 초기화합니다.
SnmpCreateSession SnmpCreateSession 함수는 Microsoft WinSNMP 구현에 WinSNMP 애플리케이션에 대한 세션을 열도록 요청합니다.
SnmpCreateVbl WinSNMP SnmpCreateVbl 함수는 호출하는 WinSNMP 애플리케이션에 대한 새 변수 바인딩 목록을 만듭니다.
SnmpDecodeMsg WinSNMP SnmpDecodeMsg 함수는 인코딩된 SNMP 메시지를 해당 구성 요소로 디코딩합니다. 이 함수는 WinSNMP SnmpEncodeMsg 함수의 반대 동작을 수행합니다.
SnmpDeleteVb WinSNMP SnmpDeleteVb 함수는 변수 바인딩 목록에서 변수 바인딩 항목을 제거합니다.
SnmpDuplicatePdu WinSNMP SnmpDuplicatePdu 함수는 PDU 매개 변수가 식별하는 SNMP PDU(프로토콜 데이터 단위)를 복제하여 중복된 PDU에 필요한 메모리를 할당합니다.
SnmpDuplicateVbl WinSNMP SnmpDuplicateVbl 함수는 지정된 WinSNMP 세션에 대한 변수 바인딩 목록을 복사합니다. 이 함수는 복사된 변수 바인딩 목록에 핸들을 반환하고 필요한 메모리를 할당합니다.
SnmpEncodeMsg Microsoft WinSNMP 구현은 WinSNMP SnmpEncodeMsg 함수에 전달된 매개 변수를 사용하여 SNMP 메시지를 인코딩합니다.

SnmpEntityToStr WinSNMP SnmpEntityToStr 함수는 SNMP 관리 엔터티를 식별하는 문자열을 반환합니다.
SnmpExtensionClose Microsoft SNMP 서비스는 SnmpExtensionClose 함수를 호출하여 SNMP 확장 에이전트가 리소스의 할당을 취소하고 작업을 종료하도록 요청합니다. 이 함수는 SNMP 확장 에이전트 API의 요소입니다.
SnmpExtensionInit Microsoft SNMP 서비스는 SnmpExtensionInit 함수를 호출하여 SNMP 확장 에이전트 DLL을 초기화합니다. 이 함수는 SNMP 확장 에이전트 API의 요소입니다.
SnmpExtensionInitEx Microsoft SNMP 서비스는 SnmpExtensionInitEx 함수를 호출하여 SNMP 확장 에이전트가 지원하는 MIB(추가 관리 정보 기반) 하위 트리를 식별합니다. 이 함수는 SNMP 확장 에이전트 API의 요소입니다.
SnmpExtensionMonitor Microsoft SNMP 서비스는 SnmpExtensionMonitor 함수를 호출하여 SNMP 확장 에이전트에 서비스의 내부 카운터 및 매개 변수에 대한 보기를 제공합니다. 이 함수는 SNMP 확장 에이전트 API의 요소입니다.
SnmpExtensionQuery Microsoft SNMP 서비스는 SnmpExtensionQuery 함수를 호출하여 SNMP 확장 에이전트의 등록된 MIB 하위 트리 중 하나 이상 내에 변수가 포함된 SNMP 요청을 확인합니다. 이 함수는 SNMP 확장 에이전트 API의 요소입니다.
SnmpExtensionQueryEx Microsoft SNMP 서비스는 SnmpExtensionQueryEx 함수를 호출하여 SNMP 확장 에이전트에 의해 등록된 하나 이상의 MIB 하위 트리에서 변수를 지정하는 SNMP 요청을 처리합니다. 이 함수는 SNMP 확장 에이전트 API의 요소입니다.
SnmpExtensionTrap Microsoft SNMP 서비스는 SnmpExtensionTrap 함수를 호출하여 서비스가 SNMP 확장 에이전트에 대한 트랩을 생성하는 데 필요한 정보를 검색합니다.
SnmpFreeContext WinSNMP SnmpFreeContext 함수는 관리되는 개체 리소스 집합인 SNMP 컨텍스트와 연결된 리소스를 해제합니다.

SnmpFreeDescriptor WinSNMP 애플리케이션은 SnmpFreeDescriptor 함수를 사용하여 설명자 개체에 더 이상 액세스할 필요가 없음을 Microsoft WinSNMP 구현에 알릴 수 있습니다.
SnmpFreeEntity WinSNMP SnmpFreeEntity 함수는 SNMP 관리 엔터티와 연결된 리소스를 해제합니다.
SnmpFreePdu WinSNMP SnmpFreePdu 함수는 SnmpCreatePdu 또는 SnmpDuplicatePdu 함수에서 만든 SNMP PDU(프로토콜 데이터 단위)와 연결된 리소스를 해제합니다.
SnmpFreeVbl WinSNMP SnmpFreeVbl 함수는 변수 바인딩 목록과 연결된 리소스를 해제합니다. 이는 WinSNMP 애플리케이션에서 SnmpCreateVbl 함수 또는 SnmpDuplicateVbl 함수를 호출하여 이전에 할당한 리소스입니다.
SnmpGetLastError WinSNMP SnmpGetLastError 함수는 호출 애플리케이션의 마지막 오류 코드 값을 반환합니다. 이 값은 WinSNMP 애플리케이션에서 실행된 마지막 함수 호출이 실패한 이유를 나타냅니다.
SnmpGetPduData WinSNMP SnmpGetPduData 함수는 지정된 SNMP 프로토콜 데이터 단위(PDU)에서 선택한 데이터 필드를 반환합니다.
SnmpGetRetransmitMode WinSNMP SnmpGetRetransmitMode 함수는 재전송 모드의 현재 설정을 WinSNMP 애플리케이션에 반환합니다.
SnmpGetRetry WinSNMP SnmpGetRetry 함수는 SNMP 메시지 요청의 재전송에 대한 재시도 횟수 값을 단위로 반환합니다. 재시도 횟수는 WinSNMP 애플리케이션이 지정된 관리 엔터티에 대한 SnmpSendMsg 함수에 대해 만드는 호출에 적용됩니다.
SnmpGetTimeout WinSNMP SnmpGetTimeout 함수는 SNMP 메시지 요청 전송에 대한 시간 제한 값(100초)을 반환합니다.

SnmpGetTranslateMode WinSNMP SnmpGetTranslateMode 함수는 엔터티 및 컨텍스트 변환 모드의 현재 설정을 WinSNMP 애플리케이션에 반환합니다.
SnmpGetVb WinSNMP 애플리케이션은 SnmpGetVb 함수를 호출하여 변수 바인딩 목록에서 정보를 검색합니다. 이 WinSNMP 함수는 인덱스 매개 변수로 지정된 변수 바인딩 항목에서 변수 이름 및 관련 값을 검색합니다.
SnmpGetVendorInfo WinSNMP 애플리케이션은 SnmpGetVendorInfo 함수를 호출하여 Microsoft WinSNMP 구현에 대한 정보를 검색합니다.
SnmpListen WinSNMP SnmpListen 함수는 WinSNMP 애플리케이션을 SNMP 에이전트로 등록합니다.
SnmpMgrClose SnmpMgrClose 함수는 지정된 세션과 연결된 통신 소켓 및 데이터 구조를 닫습니다. 이 함수는 SNMP 관리 API의 요소입니다.
SnmpMgrCtl SnmpMgrCtl 함수는 SNMP 세션과 연결된 운영 매개 변수를 설정합니다. 이 함수는 SNMP 관리 API의 요소입니다.
SnmpMgrGetTrap SnmpMgrGetTrap 함수는 트랩 수신이 활성화된 경우 호출자가 받지 못한 미해결 트랩 데이터를 반환합니다. 이 함수는 SNMP 관리 API의 요소입니다.
SnmpMgrGetTrapEx SnmpMgrGetTrapEx 함수는 트랩 수신이 활성화된 경우 호출자가 받지 못한 미해결 트랩 데이터를 반환합니다.
SnmpMgrOidToStr SnmpMgrOidToStr 함수는 내부 개체 식별자 구조를 문자열 표현으로 변환합니다. 이 함수는 SNMP 관리 API의 요소입니다.
SnmpMgrOpen SnmpMgrOpen 함수는 통신 소켓 및 데이터 구조를 초기화하여 지정된 SNMP 에이전트와의 통신을 허용합니다. 이 함수는 SNMP 관리 API의 요소입니다.

SnmpMgrRequest SnmpMgrRequest 함수는 지정된 에이전트를 사용하여 지정된 작업을 수행할 것을 요청합니다. 이 함수는 SNMP 관리 API의 요소입니다.
SnmpMgrStrToOid SnmpMgrStrToOid 함수는 개체 식별자의 문자열 형식을 내부 개체 식별자 구조로 변환합니다. 이 함수는 SNMP 관리 API의 요소입니다.
SnmpMgrTrapListen SnmpMgrTrapListen 함수는 SNMP 트랩 서비스에서 SNMP 트랩을 수신하는 SNMP 관리자 애플리케이션의 기능을 등록합니다. 이 함수는 SNMP 관리 API의 요소입니다.
SnmpOidCompare WinSNMP SnmpOidCompare 함수는 maxlen 매개 변수에 지정된 길이까지 두 개의 SNMP 개체 식별자를 사전순으로 비교합니다.
SnmpOidCopy WinSNMP SnmpOidCopy 함수는 SNMP 개체 식별자를 복사하여 복사에 필요한 메모리를 할당합니다.
SnmpOidToStr WinSNMP SnmpOidToStr 함수는 SNMP 개체 식별자의 내부 이진 표현을 점선 숫자 문자열 형식(예: "1.2.3.4.5.6")으로 변환합니다.
SnmpOpen SnmpOpen 함수는 Microsoft WinSNMP 구현에 WinSNMP 애플리케이션에 대한 세션을 열도록 요청합니다.
SnmpRecvMsg WinSNMP SnmpRecvMsg 함수는 SNMP 메시지의 형태로 SnmpSendMsg 함수에 대한 호출로 제출된 완료된 비동기 요청의 결과를 검색합니다.
SnmpRegister WinSNMP SnmpRegister 함수는 트랩 및 알림 수신을 위해 WinSNMP 애플리케이션을 등록하거나 등록 취소합니다. 애플리케이션은 트랩 및 알림을 등록 및 수신하거나 트랩 및 알림을 등록 취소하고 사용하지 않도록 설정할 수 있습니다.

SnmpSendMsg WinSNMP 애플리케이션은 SnmpSendMsg 함수를 호출하여 Microsoft WinSNMP 구현이 SNMP 메시지 형식으로 SNMP PDU(프로토콜 데이터 단위)를 전송하도록 요청합니다.
SnmpSetPduData WinSNMP SnmpSetPduData 함수는 지정된 SNMP 프로토콜 데이터 단위(PDU)에서 선택한 데이터 필드를 업데이트합니다.
SnmpSetPort WinSNMP 애플리케이션은 SnmpSetPort 함수를 호출하여 대상 엔터티에 할당된 포트를 변경합니다. SnmpSetPort 함수는 WinSNMP API 버전 2.0의 요소입니다.
SnmpSetRetransmitMode WinSNMP SnmpSetRetransmitMode 함수를 사용하면 WinSNMP 애플리케이션이 재전송 모드를 설정할 수 있습니다.
SnmpSetRetry WinSNMP SnmpSetRetry 함수를 사용하면 WinSNMP 애플리케이션에서 SNMP 메시지 요청 재전송에 대한 재시도 횟수 값을 변경할 수 있습니다.
SnmpSetTimeout WinSNMP SnmpSetTimeout 함수를 사용하면 WinSNMP 애플리케이션에서 SNMP 메시지 요청 전송에 대한 제한 시간 값을 변경할 수 있습니다.
SnmpSetTranslateMode WinSNMP SnmpSetTranslateMode 함수를 사용하면 WinSNMP 애플리케이션이 엔터티 및 컨텍스트 변환 모드를 변경할 수 있습니다. 엔터티 및 컨텍스트 변환 모드는 WinSNMP 입력 및 출력 문자열 매개 변수의 해석 및 반환에 영향을 줍니다.
SnmpSetVb WinSNMP SnmpSetVb 함수는 변수 바인딩 목록의 변수 바인딩 항목을 변경합니다. 또한 이 함수는 기존 변수 바인딩 목록에 새 변수 바인딩 항목을 추가합니다.
SnmpStartup SnmpStartup 함수는 WinSNMP 애플리케이션에 구현의 서비스가 필요하다는 것을 Microsoft WinSNMP 구현에 알린 것입니다.

SnmpStartupEx SnmpStartupEx 함수는 Microsoft WinSNMP 구현에 WinSNMP 애플리케이션에 구현의 서비스가 필요하다는 것을 알 수 있습니다.
SnmpStrToContext WinSNMP SnmpStrToContext 함수는 Microsoft WinSNMP 구현과 관련된 SNMP 컨텍스트 정보에 대한 핸들을 반환합니다.
SnmpStrToEntity WinSNMP SnmpStrToEntity 함수는 Microsoft WinSNMP 구현과 관련된 SNMP 관리 엔터티에 대한 정보에 대한 핸들을 반환합니다.
SnmpStrToOid WinSNMP SnmpStrToOid 함수는 SNMP 개체 식별자의 점선 숫자 문자열 형식(예: "1.2.3.4.5.6")을 내부 이진 표현으로 변환합니다.
SnmpSvcGetUptime SnmpSvcGetUptime 함수는 SNMP 서비스가 실행 중인 센티미터 수를 검색합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpSvcSetLogLevel SnmpSvcSetLogLevel 함수는 SnmpUtilDbgPrint 함수를 사용하여 SNMP 서비스 및 SNMP 확장 에이전트에서 디버그 출력의 세부 정보를 조정합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpSvcSetLogType SnmpSvcSetLogType 함수는 SNMP 서비스 및 SNMP 확장 에이전트에서 SnmpUtilDbgPrint 함수를 사용하여 디버그 출력의 대상을 조정합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilAsnAnyCpy SnmpUtilAsnAnyCpy 함수는 pAnySrc 매개 변수가 가리키는 변수를 pAnyDst 매개 변수에 복사합니다. 이 함수는 대상의 복사본에 필요한 메모리를 할당합니다. SnmpUtilAsnAnyCpy 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilAsnAnyFree SnmpUtilAsnAnyFree 함수는 지정된 AsnAny 구조에 할당된 메모리를 해제합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.

SnmpUtilDbgPrint SnmpUtilDbgPrint 함수를 사용하면 SNMP 서비스에서 출력을 디버깅할 수 있습니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilIdsToA SnmpUtilIdsToA 함수는 OID(개체 식별자)를 null로 끝나는 문자열로 변환합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilMemAlloc SnmpUtilMemAlloc 함수는 프로세스 힙에서 동적 메모리를 할당합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilMemFree SnmpUtilMemFree 함수는 지정된 메모리 개체를 해제합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilMemReAlloc SnmpUtilMemReAlloc 함수는 지정된 메모리 개체의 크기를 변경합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilOctetsCmp SnmpUtilOctetsCmp 함수는 두 개의 8진수 문자열을 비교합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilOctetsCpy SnmpUtilOctetsCpy 함수는 pOctetsSrc 매개 변수가 가리키는 변수를 pOctetsDst 매개 변수가 가리키는 변수에 복사합니다.
SnmpUtilOctetsFree SnmpUtilOctetsFree 함수는 지정된 8진수 문자열에 할당된 메모리를 해제합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilOctetsNCmp SnmpUtilOctetsNCmp 함수는 두 개의 8진수 문자열을 비교합니다. 함수는 nChars 매개 변수로 지정된 하위 ID 수에 도달할 때까지 문자열의 하위 ID를 비교합니다. SnmpUtilOctetsNCmp는 SNMP 유틸리티 API의 요소입니다.

SnmpUtilOidAppend SnmpUtilOidAppend 함수는 원본 개체 식별자를 대상 개체 식별자에 추가합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilOidCmp SnmpUtilOidCmp 함수는 두 개체 식별자를 비교합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilOidCpy SnmpUtilOidCpy 함수는 pOidSrc 매개 변수가 가리키는 변수를 pOidDst 매개 변수에 복사하여 대상의 복사본에 필요한 메모리를 할당합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilOidFree SnmpUtilOidFree 함수는 지정된 개체 식별자에 할당된 메모리를 해제합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilOidNCmp SnmpUtilOidNCmp 함수는 두 개체 식별자를 비교합니다.
SnmpUtilOidToA SnmpUtilOidToA 함수는 OID(개체 식별자)를 null로 끝나는 문자열로 변환합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilPrintAsnAny SnmpUtilPrintAsnAny 함수는 Any 매개 변수의 값을 표준 출력에 출력합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilPrintOid SnmpUtilPrintOid 함수는 지정된 OID(개체 식별자)의 형식을 지정하고 결과를 표준 출력 디바이스에 출력합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilVarBindCpy SnmpUtilVarBindCpy 함수는 지정된 SnmpVarBind 구조를 복사하고 대상 구조에 필요한 모든 메모리를 할당합니다. SnmpUtilVarBindCpy 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilVarBindFree SnmpUtilVarBindFree 함수는 SnmpVarBind 구조에 할당된 메모리를 해제합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.

SnmpUtilVarBindListCpy SnmpUtilVarBindListCpy 함수는 지정된 SnmpVarBindList 구조를 복사하고 대상의 복사본에 필요한 메모리를 할당합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilVarBindListFree SnmpUtilVarBindListFree 함수는 SnmpVarBindList 구조에 할당된 메모리를 해제합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.

구조체

AsnAny AsnAny 구조체에는 SNMP 변수 형식 및 값이 포함됩니다. 이 구조체는 많은 SNMP 함수에서 매개 변수로 사용되는 SnmpVarBind 구조체의 멤버입니다. 이 구조는 WinSNMP API 함수에서 사용되지 않습니다.
AsnObjectIdentifier AsnObjectIdentifier 구조체는 개체 식별자를 나타냅니다. 이 구조는 여러 SNMP 함수에서 사용됩니다. 이 구조는 WinSNMP API 함수에서 사용되지 않습니다.
AsnOctetString AsnOctetString 구조체에는 일반적으로 바이트인 8진수 수량이 포함됩니다. 이 구조는 여러 SNMP 함수에서 사용됩니다. 이 구조는 WinSNMP API 함수에서 사용되지 않습니다.
smiCNTR64 WinSNMP smiCNTR64 구조체에는 부호 없는 64비트 정수 값이 포함됩니다. 구조체는 64비트 카운터를 나타냅니다.
smiOCTETS WinSNMP smiOCTETS 구조체는 컨텍스트 문자열을 여러 WinSNMP 함수에 전달합니다. 또한 구조는 인코딩된 SNMP 메시지를 설명하고 받습니다.
smiOID WinSNMP smiOID 구조체는 개체 식별자를 여러 WinSNMP 함수에 전달합니다. 또한 구조체는 SnmpGetVb 함수에 대한 호출에서 변수 바인딩 항목의 변수 이름을 받습니다.

smiVALUE WinSNMP smiVALUE 구조체는 변수 바인딩 항목의 변수 이름과 연결된 값을 설명합니다.
smiVENDORINFO smiVENDORINFO 구조체에는 Microsoft WinSNMP 구현에 대한 정보가 포함되어 있습니다.
SnmpVarBind SnmpVarBind 구조체는 SNMP 변수 바인딩을 나타냅니다. 이 구조는 여러 SNMP 함수에서 사용됩니다. 이 구조는 WinSNMP API 함수에서 사용되지 않습니다.
SnmpVarBindList SnmpVarBindList 구조체는 SNMP 변수 바인딩 목록을 나타냅니다. 이 구조는 여러 SNMP 함수에서 사용됩니다. 이 구조는 WinSNMP API 함수에서 사용되지 않습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

소프트웨어 디바이스 API

아티클2023. 08. 22.

소프트웨어 디바이스 API 기술 개요입니다.

소프트웨어 디바이스 API를 개발하려면 다음 헤더가 필요합니다.

- swdevice.h
- swdevicedef.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 소프트웨어 디바이스 API

Functions

SW_DEVICE_CREATE_CALLBACK 레지스트리에서 백업이 있는 디바이스를 제공하고 호출자가 hSwDevice 핸들을 사용하여 소프트웨어 디바이스 API 함수를 호출할 수 있도록 합니다.
SwDeviceClose 소프트웨어 디바이스 핸들을 닫습니다. 핸들이 닫히면 SwDevice의 수명이 SWDeviceLifetimeHandle인 경우 PnP는 디바이스를 "플러깅 해제" 프로세스를 시작합니다. 디바이스는 더 이상 부모 디바이스의 자식으로 보고되지 않습니다.
SwDeviceCreate 소프트웨어 디바이스의 열거를 시작합니다.
SwDeviceGetLifetime 소프트웨어 디바이스의 수명을 가져옵니다.
SwDeviceInterfacePropertySet 소프트웨어 디바이스 인터페이스의 속성을 설정합니다.
SwDeviceInterfaceRegister 소프트웨어 디바이스에 대한 디바이스 인터페이스를 등록하고 필요에 따라 해당 인터페이스의 속성을 설정합니다.

SwDeviceInterfaceSetState 소프트웨어 디바이스에 대한 디바이스 인터페이스를 사용하거나 사용하지 않도록 설정합니다.
SwDevicePropertySet 소프트웨어 디바이스에서 속성을 설정합니다.
SwDeviceSetLifetime 소프트웨어 디바이스의 수명을 관리합니다.
SwMemFree 다른 소프트웨어 디바이스 API 함수가 할당한 메모리를 해제합니다.

구조체

SW_DEVICE_CREATE_INFO PnP가 소프트웨어 디바이스를 만드는 데 사용하는 정보를 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

맞춤법 검사기 API

아티클2023. 08. 22.

맞춤법 검사기 API 기술 개요입니다.

Spell Checker API 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 맞춤법 검사기 API

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

스토리지 드라이버 DDI 참조

아티클2023. 08. 22.

스토리지 드라이버 DDI 참조 기술 개요입니다.

스토리지 드라이버 DDI 참조 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 스토리지 드라이버 DDI 참조

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

스토리지 볼륨 공급자

아티클2023. 08. 22.

스토리지 볼륨 공급자 기술 개요입니다.

스토리지 볼륨 공급자 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 스토리지 볼륨 공급자

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

스트리밍 미디어 DDI 참조

아티클2023. 03. 09.

스트리밍 미디어 DDI 참조 기술 개요입니다.

스트리밍 미디어 DDI 참조 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 스트리밍 미디어 DDI 참조

열거형

MF_MEDIASOURCE_STATUS_INFO . (MF_MEDIASOURCE_STATUS_INFO)
MF_TRANSFER_VIDEO_FRAME_FLAGS . (MF_TRANSFER_VIDEO_FRAME_FLAGS)

Functions

FlushInputStream FlushInputStream 메서드는 디바이스 MFT의 입력 스트림을 플러시합니다.
FlushOutputStream FlushOutputStream 메서드는 디바이스 MFT의 출력 스트림을 플러시합니다.
GetInputAvailableType GetInputAvailableType 메서드는 이 MFT(Media Foundation 변환)의 입력 스트림에 사용할 수 있는 미디어 형식을 가져옵니다.
GetInputCurrentType GetInputCurrentType 메서드는 이 MFT(Media Foundation 변환)의 입력 스트림에 대한 현재 미디어 형식을 가져옵니다.

GetInputStreamAttributes GetInputStreamAttributes 메서드는 이 MFT(Media Foundation 변환)의 입력 스트림에 대한 특성 저장소를 가져옵니다.
GetInputStreamPreferredState GetInputStreamPreferredState 메서드는 디바이스 MFT 입력 스트림의 기본 상태 및 미디어 형식을 가져옵니다.
GetInputStreamState GetInputStreamState 메서드는 디바이스 MFT의 입력 스트림 상태를 가져옵니다.
GetOutputAvailableType GetOutputAvailableType 메서드는 이 MFT(Media Foundation 변환)의 출력 스트림에 사용할 수 있는 미디어 형식을 가져옵니다.
GetOutputCurrentType GetOutputCurrentType 메서드는 이 MFT(Media Foundation 변환)의 출력 스트림에 대한 현재 미디어 형식을 가져옵니다.
GetOutputStreamAttributes GetOutputStreamAttributes 메서드는 이 MFT(Media Foundation 변환)의 출력 스트림에 대한 특성 저장소를 가져옵니다.
GetOutputStreamState GetOutputStreamState 메서드는 디바이스 MFT의 출력 스트림 상태를 가져옵니다.
GetStreamCount GetStreamCount 메서드는 이 MFT(Media Foundation 변환)에서 현재 입력 및 출력 스트림 수를 가져옵니다.
GetStreamIDs GetStreamIDs 메서드는 이 MFT(Media Foundation 변환)의 입력 및 출력 스트림에 대한 스트림 식별자를 가져옵니다.
InitializeTransform InitializeTransform은 디바이스 MFT를 초기화하기 위해 호출됩니다.

OnBufferSent 시스템 할당 프레임 버퍼를 디바이스 드라이버로 보낼 때 호출됩니다.
ProcessEvent ProcessEvent 메서드는 이 MFT(Media Foundation 변환)의 입력 스트림에 이벤트를 보냅니다.
ProcessInput ProcessInput 메서드는 이 MFT(Media Foundation 변환)의 입력 스트림에 데이터를 전달합니다.
ProcessMessage ProcessMessage 메서드는 MFT(Device Media Foundation 변환)에 메시지를 보냅니다.
ProcessOutput ProcessOutput 메서드는 디바이스 MFT 출력 스트림에서 처리된 출력을 가져옵니다.
SetInputStreamState SetInputStreamState 메서드는 디바이스 MFT 입력 스트림 상태 및 미디어 형식을 설정합니다.
SetOutputStreamState SetOutputStreamState 메서드는 디바이스 MFT 출력 스트림 상태 및 미디어 형식을 설정합니다.

인터페이스

IMFDeviceTransform 이 섹션에는 IMFDeviceTransform 인터페이스에 대한 참조 정보가 포함되어 있습니다.
IMFDeviceTransformCallback 시스템 할당 프레임 버퍼가 디바이스 드라이버로 전송될 때 알림을 받도록 이 콜백을 구현합니다.

구조체

CapturedMetadataExposureCompensation 이 구조에는 캡처된 사진에 대한 EV 보정 피드백에 대한 Blob 정보가 포함됩니다.
CapturedMetadataISOGains CapturedMetadataISOGains 구조체는 MF_CAPTURE_METADATA_ISO_GAINS 대한 Blob 형식을 설명합니다.
CapturedMetadataWhiteBalanceGains 이 구조에서는 MF_CAPTURE_METADATA_WHITEBALANCE_GAINS 특성에 대한 Blob 형식을 설명합니다.
FaceCharacterization FaceCharacterization 구조는 MF_CAPTURE_METADATA_FACEROICHARACTERIZATIONS 특성에 대한 Blob 형식을 설명합니다.
FaceCharacterizationBlobHeader FaceCharacterizationBlobHeader 구조체는 MF_CAPTURE_METADATA_FACEROICHARACTERIZATIONS 특성에 대한 Blob 형식의 크기 및 개수 정보를 설명합니다.
FaceRectInfo FaceRectInfo 구조체는 MF_CAPTURE_METADATA_FACEROIS 특성에 대한 Blob 형식을 설명합니다.
FaceRectInfoBlobHeader FaceRectInfoBlobHeader 구조체는 MF_CAPTURE_METADATA_FACEROIS 특성에 대한 Blob 형식의 크기 및 개수 정보를 설명합니다.
HistogramBlobHeader HistogramBlobHeader 구조체는 blob 크기 및 MF_CAPTURE_METADATA_HISTOGRAM 특성에 대한 Blob의 히스토그램 수를 설명합니다.
HistogramDataHeader HistogramDataHeader 구조체는 MF_CAPTURE_METADATA_HISTOGRAM 특성에 대한 Blob 형식을 설명합니다.
HistogramGrid HistogramGrid 구조체는 MF_CAPTURE_METADATA_HISTOGRAM 대한 Blob 형식을 설명합니다.

HistogramHeader HistogramHeader 구조는 MF_CAPTURE_METADATA_HISTOGRAM Blob 형식을 설명합니다.
MetadataTimeStamps MetadataTimeStamps 구조체는 MF_CAPTURE_METADATA_FACEROITIMESTAMPS 특성에 대한 Blob 형식을 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

구조적 저장소

아티클2023. 08. 22.

구조적 스토리지 기술 개요입니다.

구조적 스토리지를 개발하려면 다음 헤더가 필요합니다.

- coml2api.h
- propapi.h
- propidlbase.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 구조적 저장소

열거형

LOCKTYPE LOCKTYPE 열거형 값은 지정된 바이트 범위에 대해 요청된 잠금 유형을 나타냅니다. 값은 ILockBytes::LockRegion 및 IStream::LockRegion 메서드에서 사용됩니다.
STATFLAG 메서드가 STATSTG 구조체의 pwcsName 멤버에서 이름을 반환해야 하는지 여부를 나타냅니다.
STGC IStorage::Commit 및 IStream::Commit 메서드에서 커밋 작업을 수행하기 위한 조건을 지정합니다.
STGMOVE 스토리지 요소를 이동하거나 복사할지 여부를 나타냅니다.
STGTY STGTY 열거형 값은 스토리지 요소의 형식을 나타내기 위해 STATSTG 구조체의 형식 멤버에 사용됩니다. 스토리지 요소는 스토리지 개체, 스트림 개체 또는 LOCKBYTES(바이트 배열 개체)입니다.
STREAM_SEEK STREAM_SEEK 열거형 값은 새 검색 포인터 위치를 계산할 원본을 지정합니다.

Functions

BeginMonitor BeginMonitor 메서드는 로드 작업이 시작될 때 모니터링을 시작하는 데 사용됩니다. 작업이 완료되면 애플리케이션에서 ILayoutStorage::EndMonitor를 호출해야 합니다.
복제 현재 STATSTG 구조체 열거자와 동일한 열거형 상태를 포함하는 새 열거자를 만듭니다.
복제 Clone 메서드는 원래 스트림과 동일한 바이트를 참조하는 자체 검색 포인터를 사용하여 새 스트림 개체를 만듭니다.
복제 IEnumSTATPROPSETSTG::Clone 메서드는 현재 STATPROPSETSTG 구조체 열거자와 동일한 열거형 상태를 포함하는 열거자를 만듭니다.
복제 IEnumSTATPROPSTG::Clone 메서드는 현재 STATPROPSTG 구조 열거자와 동일한 열거형 상태를 포함하는 열거자를 만듭니다.
복제 IEnumSTATPROPSETSTG::Clone 메서드는 현재 STATPROPSETSTG 구조체 열거자와 동일한 열거형 상태를 포함하는 열거자를 만듭니다.
복제 IEnumSTATPROPSTG::Clone 메서드는 현재 STATPROPSTG 구조 열거자와 동일한 열거형 상태를 포함하는 열거자를 만듭니다.
커밋 Commit 메서드는 트랜잭션 모드에서 열린 스토리지 개체에 대한 모든 변경 내용이 부모 스토리지에 반영되도록 합니다.
커밋 Commit 메서드를 사용하면 트랜잭션 모드에서 열린 스트림 개체에 대한 모든 변경 내용이 부모 스토리지에 반영됩니다.
커밋

IPropertyStorage::Commit 메서드는 속성 스토리지 개체의 변경 내용을 부모 스토리지 개체에 저장합니다.
커밋 IPropertyStorage::Commit 메서드는 속성 스토리지 개체의 변경 내용을 부모 스토리지 개체에 저장합니다.
CopyTo 열려 있는 스토리지 개체의 전체 내용을 다른 스토리지 개체에 복사합니다.
CopyTo 지정된 바이트 수를 스트림의 현재 검색 포인터에서 다른 스트림의 현재 검색 포인터 위치로 복사합니다.
만들기 속성 집합 스토리지 개체에서 새 속성 집합을 만들고 엽니다.
CreateILockBytesOnHGlobal HGLOBAL 메모리 핸들을 사용하여 복합 파일의 메모리 내 스토리지용 바이트를 저장하는 바이트 배열 개체를 만듭니다.
CreateStorage 지정된 액세스 모드에서 지정된 이름을 사용하여 이 스토리지 개체 내에 중첩된 새 스토리지 개체를 만들고 엽니다.
CreateStream 이 스토리지 개체에 포함된 지정된 이름을 사용하여 스트림 개체를 만들고 엽니다.
CreateStreamOnHGlobal HGLOBAL 메모리 핸들을 사용하여 스트림 콘텐츠를 저장하는 스트림 개체를 만듭니다.
삭제 Delete 메서드는 속성 집합 스토리지 개체에 포함된 속성 집합 중 하나를 삭제합니다.
DeleteMultiple IPropertyStorage::D eleteMultiple 메서드는 이 속성 집합에 있는 만큼 표시된 속성을 삭제합니다.
DeleteMultiple

IPropertyStorage::D eleteMultiple 메서드는 속성 집합에 있는 만큼 표시된 속성을 삭제합니다.
DeletePropertyNames DeletePropertyNames 메서드는 현재 속성 집합에서 지정된 문자열 이름을 삭제합니다. (IPropertyStorage.DeletePropertyNames)
DeletePropertyNames IPropertyStorage::D eletePropertyNames 메서드는 현재 속성 집합에서 지정된 문자열 이름을 삭제합니다.
DestroyElement 이 스토리지 개체에서 지정된 스토리지 또는 스트림을 제거합니다.
EndMonitor EndMonitor 메서드는 복합 파일의 모니터링을 종료합니다. 앞에 ILayoutStorage::BeginMonitor를 호출해야 합니다.
Enum Enum 메서드는 이 속성 집합 스토리지에 저장된 속성 집합에 대한 정보를 포함하는 열거자 개체를 만듭니다. 반환 시 이 메서드는 열거자 개체의 IEnumSTATPROPSETSTG 포인터에 대한 포인터를 제공합니다.
Enum IPropertyStorage::Enum 메서드는 현재 속성 집합에 대한 정보를 포함하는 STATPROPSTG 형식의 데이터를 열거하도록 설계된 열거자 개체를 만듭니다.
Enum IPropertyStorage::Enum 메서드는 현재 속성 집합에 대한 정보를 포함하는 STATPROPSTG 형식의 데이터를 열거하도록 설계된 열거자 개체를 만듭니다.
EnumElements EnumElements 메서드는 이 스토리지 개체에 포함된 스토리지 및 스트림 개체를 열거하는 데 사용할 수 있는 열거자 개체에 대한 포인터를 검색합니다.
FillAppend FillAppend 메서드는 바이트 배열의 끝에 새 바이트 블록을 씁니다.
FillAt FillAt 메서드는 바이트 배열의 지정된 위치에 새 데이터 블록을 씁니다.

플러시 Flush 메서드는 ILockBytes 구현에서 유지 관리하는 모든 내부 버퍼가 기본 물리적 스토리지에 기록되도록 합니다.
FmtIdToPropStgName FMTID(속성 집합 형식 식별자)를 스토리지 또는 스트림 이름으로 변환합니다.
FreePropVariantArray FreePropVariantArray 함수는 rgvars 배열의 각 PROPVARIANT 구조체에서 PropVariantClear를 호출하여 배열의 각 멤버에 대해 값을 0으로 만듭니다.
GetConvertStg GetConvertStg 함수는 지정된 스토리지 개체에 대한 convert 비트의 현재 값을 반환합니다.
GetHGlobalFromILockBytes GetHGlobalFromILockBytes 함수는 CreateILockBytesOnHGlobal 함수를 사용하여 만든 바이트 배열 개체에 대한 전역 메모리 핸들을 검색합니다.
GetHGlobalFromStream GetHGlobalFromStream 함수는 CreateStreamOnHGlobal 함수를 호출하여 만든 스트림에 대한 전역 메모리 핸들을 검색합니다.
HaveWriteAccess HaveWriteAccess 메서드는 쓰기 잠금이 수행되었는지 여부를 나타냅니다.
LayoutScript LayoutScript 메서드는 다운로드 중에 액세스되는 순서와 일치하도록 복합 파일의 스토리지, 스트림 및 컨트롤을 다시 정렬하기 위한 명시적 지침을 제공합니다.
LockRegion LockRegion 메서드는 바이트 배열에서 지정된 바이트 범위에 대한 액세스를 제한합니다.
LockRegion LockRegion 메서드는 스트림에서 지정된 바이트 범위에 대한 액세스를 제한합니다.
MoveElementTo MoveElementTo 메서드는 하위 저장소 또는 스트림을 이 스토리지 개체에서 다른 스토리지 개체로 복사하거나 이동합니다.

다음 열거형 시퀀스에 따라 지정된 수의 STATSTG 구조를 검색합니다.
다음 IEnumSTATPROPSETSTG::Next 메서드는 이후에 열거형 시퀀스에서 이어지는 지정된 수의 STATPROPSETSTG 구조를 검색합니다.
다음 IEnumSTATPROPSTG::Next 메서드는 지정된 수의 STATPROPSTG 구조를 검색합니다. 이 구조체는 이후 열거형 시퀀스에서 수행됩니다.
다음 IEnumSTATPROPSETSTG::Next 메서드는 이후에 열거형 시퀀스에서 이어지는 지정된 수의 STATPROPSETSTG 구조를 검색합니다.
다음 IEnumSTATPROPSTG::Next 메서드는 지정된 수의 STATPROPSTG 구조를 검색합니다. 이 구조체는 이후 열거형 시퀀스에서 수행됩니다.
OleConvertIStorageToOLESTREAM OleConvertIStorageToOLESTREAM 함수는 지정된 스토리지 개체를 OLE 2 구조적 스토리지에서 OLE 1 스토리지 개체 모델로 변환하지만 프레젠테이션 데이터는 포함하지 않습니다. 이는 여러 호환성 함수 중 하나입니다.
OleConvertIStorageToOLESTREAMEx OleConvertIStorageToOLESTREAMEx 함수는 지정된 스토리지 개체를 OLE 2 구조적 스토리지에서 프레젠테이션 데이터를 포함한 OLE 1 스토리지 개체 모델로 변환합니다.
OleConvertOLESTREAMToIStorage 프레젠테이션 데이터를 지정하지 않고 지정된 개체를 OLE 1 스토리지 모델에서 OLE 2 구조적 스토리지 개체로 변환합니다.
OleConvertOLESTREAMToIStorageEx OleConvertOLESTREAMToIStorageEx 함수는 지정된 개체를 OLE 1 스토리지 모델에서 프레젠테이션 데이터를 포함한 OLE 2 구조적 스토리지 개체로 변환합니다. 이는 여러 호환성 함수 중 하나입니다.
열기 속성 집합 스토리지 개체에 포함된 속성 집합을 엽니다.

OpenStorage 지정된 액세스 모드에서 지정된 이름의 기존 스토리지 개체를 엽니다.
OpenStream 지정된 액세스 모드에서 이 스토리지 개체 내의 기존 스트림 개체를 엽니다.
PropStgNameToFmtId 속성 집합 스토리지 또는 스트림 이름을 해당 형식 식별자로 변환합니다.
PropVariantClear 지정된 PROPVARIANT 구조에서 해제할 수 있는 모든 요소를 해제합니다.
PropVariantCopy PropVariantCopy 함수는 한 PROPVARIANT 구조체의 내용을 다른 구조체에 복사합니다.
PropVariantInit PropVariantInit 함수는 PROPVARIANT 구조를 초기화합니다. 참고 이 함수는 제공된 ole2.h 헤더 파일을 포함하여 사용할 수 있는 매크로로 구현됩니다.
읽기 현재 검색 포인터부터 시작하여 스트림 개체에서 메모리로 지정된 바이트 수를 읽습니다.
ReadAt ReadAt 메서드는 바이트 배열 개체의 시작 부분에서 지정된 오프셋에서 시작하는 지정된 바이트 수를 읽습니다.
ReadClassStg ReadClassStg 함수는 WriteClassStg 함수를 사용하여 스토리지 개체에 이전에 작성된 CLSID를 읽습니다.
ReadClassStm WriteClassStm 함수를 사용하여 스트림 개체에 이전에 작성된 CLSID를 읽습니다.
ReadFmtUserTypeStg ReadFmtUserTypeStg 함수는 WriteFmtUserTypeStg 함수로 이전에 저장된 클립보드 형식 및 사용자 유형을 반환합니다.

ReadMultiple IPropertyStorage::ReadMultiple 메서드는 현재 속성 집합에서 지정된 속성을 읽습니다.
ReadMultiple IPropertyStorage::ReadMultiple 메서드는 현재 속성 집합에서 지정된 속성을 읽습니다.
ReadPropertyNames IPropertyStorage::ReadPropertyNames 메서드는 지정된 속성 ID에 대한 기존 문자열 이름을 검색합니다.
ReadPropertyNames IPropertyStorage::ReadPropertyNames 메서드는 지정된 속성 ID에 대한 기존 문자열 이름을 검색합니다.
ReLayoutDocfile ReLayoutDocfile 메서드는 모니터링을 통해 가져오거나 명시적 레이아웃 스크립팅을 통해 제공되는 레이아웃 스크립트를 사용하여 복합 파일을 다시 작성하여 새 복합 파일을 만듭니다.
ReLayoutDocfileOnILockBytes 구현되지 않았습니다. 호출되면 STG_E_UNIMPLEMENTEDFUNCTION 반환합니다.
ReleaseWriteAccess ReleaseWriteAccess 메서드는 이전에 가져온 쓰기 잠금을 해제합니다.
RenameElement RenameElement 메서드는 이 스토리지 개체에서 지정된 하위 저장소 또는 스트림의 이름을 바꿉니다.
Reset 열거형 시퀀스를 STATSTG 구조 배열의 시작 부분으로 다시 설정합니다.
Reset IEnumSTATPROPSETSTG::Reset 메서드는 열거형 시퀀스를 STATPROPSETSTG 구조 배열의 시작 부분으로 다시 설정합니다.
Reset IEnumSTATPROPSTG::Reset 메서드는 열거형 시퀀스를 STATPROPSTG 구조 배열의 시작 부분으로 다시 설정합니다.

Reset IEnumSTATPROPSETSTG::Reset 메서드는 열거형 시퀀스를 STATPROPSETSTG 구조 배열의 시작 부분으로 다시 설정합니다.
Reset IEnumSTATPROPSTG::Reset 메서드는 열거형 시퀀스를 STATPROPSTG 구조 배열의 시작 부분으로 다시 설정합니다.
되돌리기 Revert 메서드는 마지막 커밋 작업 이후 스토리지 개체에 적용된 모든 변경 내용을 삭제합니다.
되돌리기 Revert 메서드는 마지막 IStream::Commit 호출 이후 트랜잭션된 스트림에 적용된 모든 변경 내용을 삭제합니다. 직접 모드로 열리는 스트림과 IStream::Revert의 COM 복합 파일 구현을 사용하는 스트림에서는 이 메서드가 적용되지 않습니다.
되돌리기 IPropertyStorage::Revert 메서드는 마지막으로 연 이후 명명된 속성 집합에 대한 모든 변경 내용을 삭제하거나 속성 집합에 마지막으로 커밋된 변경 내용을 삭제합니다.
되돌리기 IPropertyStorage::Revert 메서드는 마지막으로 연 이후 명명된 속성 집합에 대한 모든 변경 내용을 삭제하거나 속성 집합에 마지막으로 커밋된 변경 내용을 삭제합니다.
Seek 검색 포인터를 새 위치로 변경합니다. 새 위치는 스트림의 시작, 스트림의 끝 또는 현재 검색 포인터를 기준으로 합니다.
SetClass SetClass 메서드는 지정된 CLSID(클래스 식별자)를 이 스토리지 개체에 할당합니다.
SetClass IPropertyStorage::SetClass 메서드는 현재 속성 스토리지 개체에 새 CLSID를 할당하고 CLSID를 개체와 함께 영구적으로 저장합니다.
SetClass IPropertyStorage::SetClass 메서드는 현재 속성 스토리지 개체에 새 CLSID를 할당하고 CLSID를 개체와 함께 영구적으로 저장합니다.

SetConvertStg SetConvertStg 함수는 스토리지 개체의 변환 비트를 설정하여 개체가 열릴 때 새 클래스로 변환될 것임을 나타냅니다. GetConvertStg 함수를 호출하여 설정을 검색할 수 있습니다.
SetElementTimes SetElementTimes 메서드는 기본 파일 시스템이 이 메서드를 지원하는 경우 지정된 스토리지 요소의 수정, 액세스 및 생성 시간을 설정합니다.
SetFillSize SetFillSize 메서드는 바이트 배열의 예상 크기를 설정합니다.
SetSize SetSize 메서드는 바이트 배열의 크기를 변경합니다.
SetSize 스트림 개체의 크기를 변경합니다.
SetStateBits SetStateBits 메서드는 이 스토리지 개체에 최대 32비트의 상태 정보를 저장합니다.
SetTimes 구현에서 지원하는 경우 IPropertyStorage::SetTimes 메서드는 이 속성 집합의 수정, 액세스 및 생성 시간을 설정합니다.
SetTimes 구현에서 지원하는 경우 IPropertyStorage::SetTimes 메서드는 이 속성 집합의 수정, 액세스 및 생성 시간을 설정합니다.
Skip 열거형 시퀀스에서 지정된 수의 STATSTG 구조를 건너뜁니다.
Skip IEnumSTATPROPSETSTG::Skip 메서드는 열거형 시퀀스에서 지정된 수의 STATPROPSETSTG 구조를 건너뜁니다. (IEnumSTATPROPSETSTG.Skip)
Skip IEnumSTATPROPSTG::Skip 메서드는 열거형 시퀀스에서 지정된 수의 STATPROPSTG 구조를 건너뜁니다. (IEnumSTATPROPSTG.Skip)

Skip IEnumSTATPROPSETSTG::Skip 메서드는 열거형 시퀀스에서 지정된 수의 STATPROPSETSTG 구조를 건너뜁니다.
Skip IEnumSTATPROPSTG::Skip 메서드는 열거형 시퀀스에서 지정된 수의 STATPROPSTG 구조를 건너뜁니다.
통계 Stat 메서드는 이 바이트 배열 개체에 대한 정보를 포함하는 STATSTG 구조를 검색합니다.
통계 Stat 메서드는 열려 있는 이 스토리지 개체에 대한 STATSTG 구조를 검색합니다.
통계 Stat 메서드는 이 스트림에 대한 STATSTG 구조를 검색합니다.
통계 IPropertyStorage::Stat 메서드는 현재 열려 있는 속성 집합에 대한 정보를 검색합니다. (IPropertyStorage.Stat)
통계 IPropertyStorage::Stat 메서드는 현재 열려 있는 속성 집합에 대한 정보를 검색합니다. (IPropertyStorage.Stat)
StgConvertPropertyToVariant SERIALIZEDPROPERTYVALUE 데이터 형식을 PROPVARIANT 데이터 형식으로 변환합니다.
StgConvertVariantToProperty PROPVARIANT 데이터 형식을 SERIALIZEDPROPERTYVALUE 데이터 형식으로 변환합니다.
StgCreateDocfile IStorage 인터페이스에 대한 COM 제공 복합 파일 구현을 사용하여 새 복합 파일 스토리지 개체를 만듭니다.
StgCreateDocfileOnILockBytes 호출자가 제공하는 바이트 배열 개체 위에 새 복합 파일 스토리지 개체를 만들고 엽니다.

StgCreatePropSetStg 지정된 스토리지 개체에서 속성 집합 스토리지 개체를 만듭니다.
StgCreatePropStg 지정된 스토리지 또는 스트림 개체에서 속성 집합을 만들고 엽니다.
StgCreateStorageEx IStorage 또는 IPropertySetStorage 인터페이스에 대해 제공된 구현을 사용하여 새 스토리지 개체를 만듭니다.
StgDeserializePropVariant StgDeserializePropVariant 함수는 SERIALIZEDPROPERTYVALUE 데이터 형식을 PROPVARIANT 데이터 형식으로 변환합니다.
StgGetIFillLockBytesOnFile 임시 파일에서 래퍼 개체를 엽니다.
StgGetIFillLockBytesOnILockBytes 호출자가 제공하는 바이트 배열 개체에 새 래퍼 개체를 만듭니다.
StgIsStorageFile StgIsStorageFile 함수는 특정 디스크 파일에 스토리지 개체가 포함되어 있는지 여부를 나타냅니다.
StgIsStorageILockBytes StgIsStorageILockBytes 함수는 지정된 바이트 배열에 스토리지 개체가 포함되어 있는지 여부를 나타냅니다.
StgOpenAsyncDocfileOnIFillLockBytes 호출자가 제공하는 바이트 배열 래퍼 개체에서 기존 루트 비동기 스토리지 개체를 엽니다.
StgOpenLayoutDocfile 섹터 데이터를 모니터링할 수 있는 ILockBytes 구현에서 복합 파일을 엽니다.
StgOpenPropStg 지정된 스토리지 또는 스트림 개체에서 지정된 속성 집합을 엽니다.
StgOpenStorage

파일 시스템에서 기존 루트 스토리지 개체를 엽니다.
StgOpenStorageEx 파일 시스템에서 기존 루트 스토리지 개체를 엽니다. 이 함수를 사용하여 복합 파일 및 일반 파일을 엽니다.
StgOpenStorageOnILockBytes StgOpenStorageOnILockBytes 함수는 디스크 파일에 존재하지 않고 대신 호출자가 제공하는 기본 바이트 배열이 있는 기존 스토리지 개체를 엽니다.
StgPropertyLengthAsVariant StgPropertyLengthAsVariant 함수는 SERIALIZEDPROPERTYVALUE를 검사하고 이 속성이 PROPVARIANT로 차지하는 메모리 양을 반환합니다.
StgSerializePropVariant StgSerializePropVariant 함수는 PROPVARIANT 데이터 형식을 SERIALIZEDPROPERTYVALUE 데이터 형식으로 변환합니다.
StgSetTimes StgSetTimes 함수는 기본 파일 시스템에서 지원하는 경우 표시된 파일의 생성, 액세스 및 수정 시간을 설정합니다.
SwitchToFile SwitchToFile 메서드는 스토리지 개체와 연결된 현재 파일을 새 파일에 복사합니다.
Terminate Terminate 메서드는 다운로드가 성공적으로 또는 실패하여 종료되었음을 바이트 배열에 알릴 수 있습니다.
UnlockRegion UnlockRegion 메서드는 이전에 잠긴 바이트 범위에 대한 액세스 제한을 제거합니다.
UnlockRegion UnlockRegion 메서드는 이전에 IStream::LockRegion으로 제한되었던 바이트 범위에 대한 액세스 제한을 제거합니다.
WaitForWriteAccess WaitForWriteAccess 메서드는 스토리지 개체에 대한 단독 쓰기 액세스 권한을 얻습니다.

쓰기 지정된 바이트 수를 현재 검색 포인터에서 시작하는 스트림 개체에 씁니다.
WriteAt WriteAt 메서드는 바이트 배열의 시작 부분에서 지정된 오프셋에서 시작하는 지정된 바이트 수를 씁니다.
WriteClassStg WriteClassStg 함수는 지정된 CLSID(클래스 식별자)를 스토리지 개체에 저장합니다.
WriteClassStm WriteClassStm 함수는 지정된 CLSID를 스트림에 저장합니다.
WriteFmtUserTypeStg WriteFmtUserTypeStg 함수는 클립보드 형식과 사용자 형식을 스토리지 개체에 씁니다.
WriteMultiple IPropertyStorage::WriteMultiple 메서드는 지정된 속성 그룹을 현재 속성 집합에 씁니다.
WriteMultiple IPropertyStorage::WriteMultiple 메서드는 지정된 속성 그룹을 현재 속성 집합에 씁니다.
WritePropertyNames IPropertyStorage::WritePropertyNames 메서드는 현재 속성 집합의 지정된 속성 ID 배열에 문자열 IPropertyStoragenames를 할당합니다.
WritePropertyNames IPropertyStorage::WritePropertyNames 메서드는 현재 속성 집합의 지정된 속성 ID 배열에 문자열 IPropertyStoragenames를 할당합니다.

인터페이스

IDirectWriterLock IDirectWriterLock 인터페이스를 사용하면 단일 작성기가 직접 모드로 열린 루트 스토리지 개체에 대한 단독 쓰기 액세스 권한을 얻는 동시에 여러 판독기에서 동시에 액세스할 수 있습니다.

IEnumSTATPROPSETSTG IEnumSTATPROPSETSTG 인터페이스는 현재 IPropertySetStorage instance 관리되는 속성 집합에 대한 통계 데이터를 포함하는 STATPROPSETSTG 구조체의 배열을 반복합니다.
IEnumSTATPROPSETSTG IEnumSTATPROPSETSTG 인터페이스는 현재 IPropertySetStorage instance 관리되는 속성 집합에 대한 통계 데이터를 포함하는 STATPROPSETSTG 구조체의 배열을 반복합니다.
IEnumSTATPROPSTG IEnumSTATPROPSTG 인터페이스는 STATPROPSTG 구조체의 배열을 반복합니다. STATPROPSTG 구조체에는 속성 집합의 속성에 대한 통계 데이터가 포함됩니다.
IEnumSTATPROPSTG IEnumSTATPROPSTG 인터페이스는 속성 집합의 속성에 대한 통계 데이터를 포함하는 STATPROPSTG 구조체의 배열을 반복합니다.
IEnumSTATSTG STATSTG 구조체의 배열을 열거합니다.
IFillLockBytes IFillLockBytes 인터페이스를 사용하면 코드를 다운로드하여 구조화된 스토리지 바이트 배열에 비동기적으로 데이터를 쓸 수 있습니다.
ILayoutStorage ILayoutStorage 인터페이스를 사용하면 애플리케이션이 느린 링크를 통해 효율적으로 다운로드할 수 있도록 복합 파일의 레이아웃을 최적화할 수 있습니다.
ILockBytes ILockBytes 인터페이스는 디스크 파일, 전역 메모리 또는 데이터베이스와 같은 일부 물리적 스토리지에서 지원되는 바이트 배열 개체에 구현됩니다.
IPropertySetStorage IPropertySetStorage 인터페이스는 IPropertyStorage 인터페이스의 인스턴스를 지원하는 속성 집합 스토리지를 만들고, 열고, 삭제하고, 열거합니다.
IPropertyStorage IPropertyStorage 인터페이스는 단일 속성 집합의 영구 속성을 관리합니다. (IPropertyStorage 인터페이스)

IPropertyStorage IPropertyStorage 인터페이스는 단일 속성 집합의 영구 속성을 관리합니다. (IPropertyStorage 인터페이스)
IRootStorage IRootStorage 인터페이스에는 스토리지 개체를 다른 기본 파일로 전환하고 스토리지 개체를 해당 파일에 저장하는 단일 메서드가 포함되어 있습니다.
ISequentialStream ISequentialStream 인터페이스는 스트림 개체에 대한 간소화된 순차 액세스를 지원합니다. IStream 인터페이스는 ISequentialStream에서 읽기 및 쓰기 메서드를 상속합니다.
IStorage IStorage 인터페이스는 구조적 스토리지 개체의 생성 및 관리를 지원합니다.
IStream IStream 인터페이스를 사용하면 데이터를 읽고 써서 개체를 스트리밍할 수 있습니다.

구조체

PROPSPEC PROPSPEC 구조체는 IPropertyStorage의 많은 메서드에서 속성 식별자(ID) 또는 연결된 문자열 이름으로 속성을 지정하는 데 사용됩니다.
PROPSPEC PROPSPEC 구조체는 IPropertyStorage의 많은 메서드에서 속성 식별자(ID) 또는 연결된 문자열 이름으로 속성을 지정하는 데 사용됩니다.
PROPVARIANT PROPVARIANT 구조체는 IPropertyStorage의 ReadMultiple 및 WriteMultiple 메서드에서 속성 집합에 있는 속성의 형식 태그 및 값을 정의하는 데 사용됩니다.
PROPVARIANT PROPVARIANT 구조체는 IPropertyStorage의 ReadMultiple 및 WriteMultiple 메서드에서 속성 집합에 있는 속성의 형식 태그 및 값을 정의하는 데 사용됩니다.

RemSNB RemSNB 구조체는 SNB 데이터 형식을 마샬링하는 데 사용됩니다. IStorage 인터페이스(Storag.idl)에 정의됩니다.
STATPROPSETSTG STATPROPSETSTG 구조체에는 속성 집합에 대한 정보가 포함됩니다. (STATPROPSETSTG 구조체)
STATPROPSETSTG STATPROPSETSTG 구조체에는 속성 집합에 대한 정보가 포함됩니다. (STATPROPSETSTG 구조체)
STATPROPSTG STATPROPSTG 구조체에는 속성 집합의 단일 속성에 대한 데이터가 포함됩니다. 이 데이터는 속성 ID 및 형식 태그이며 속성과 연결할 수 있는 선택적 문자열 이름입니다.
STATPROPSTG STATPROPSTG 구조체에는 속성 집합의 단일 속성에 대한 데이터가 포함됩니다. 이 데이터는 속성 ID 및 형식 태그이며 속성과 연결할 수 있는 선택적 문자열 이름입니다.
Statstg 열린 스토리지, 스트림 또는 바이트 배열 개체에 대한 통계 데이터를 포함합니다.
STGOPTIONS StgCreateStorageEx 및 StgOpenStorageEx 함수에서 섹터 크기와 같은 스토리지 개체의 기능을 지정합니다.
StorageLayout 이름, 위치 및 길이를 포함한 단일 데이터 블록에 대해 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

시스템 이벤트 알림 서비스

아티클2023. 08. 22.

시스템 이벤트 알림 서비스 기술 개요입니다.

시스템 이벤트 알림 서비스를 개발하려면 다음 헤더가 필요합니다.

- sensapi.h
- sensevts.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 시스템 이벤트 알림 서비스

Functions

BatteryLow BatteryLow 메서드는 애플리케이션에 배터리 전원이 부족하다는 것을 알 수 있습니다. SENS는 BatteryLow 메서드를 호출하여 컴퓨터가 배터리 전원을 사용하고 있음을 애플리케이션에 알립니다.
ConnectionLost ConnectionLost 메서드는 애플리케이션에 지정된 연결이 삭제되었음을 알 수 있습니다.
ConnectionMade ConnectionMade 메서드는 애플리케이션에 지정된 연결이 설정되었음을 알 수 있습니다.
ConnectionMadeNoQOCInfo ConnectionMadeNoQOCInfo 메서드는 지정된 연결이 사용 가능한 연결 품질 정보 없이 설정되었음을 애플리케이션에 알린 것입니다.
DisplayLock DisplayLock 메서드는 화면 표시가 잠겨 있음을 애플리케이션에 알 수 있습니다.
DisplayUnlock DisplayUnLock 메서드는 화면 표시가 잠금 해제되었음을 애플리케이션에 알 수 있습니다.
IsDestinationReachableA

IsDestinationReachable 함수는 지정된 대상에 연결할 수 있는지 여부를 결정하고 대상에 대한 QOC(연결 품질) 정보를 제공합니다. (ANSI)
IsDestinationReachableW IsDestinationReachable 함수는 지정된 대상에 연결할 수 있는지 여부를 결정하고 대상에 대한 QOC(연결 품질) 정보를 제공합니다. (유니코드)
IsNetworkAlive IsNetworkAlive 함수는 로컬 시스템이 네트워크에 연결되어 있는지 여부를 확인하고 네트워크 연결 유형(예: LAN, WAN 또는 둘 다)을 식별합니다.
Logoff Logoff 메서드는 애플리케이션에 사용자가 로그오프되었음을 알 수 있습니다. (ISensLogon.Logoff)
Logoff Logoff 메서드는 애플리케이션에 사용자가 로그오프되었음을 알 수 있습니다. (ISensLogon2.Logoff)
로그온 Logon 메서드는 애플리케이션에 사용자가 로그온되었음을 알 수 있습니다. (ISensLogon.Logon)
로그온 Logon 메서드는 애플리케이션에 사용자가 로그온되었음을 알 수 있습니다. (ISensLogon2.Logon)
OnACPower SENS는 OnACPower 메서드를 호출하여 컴퓨터가 AC 전원을 사용하고 있음을 애플리케이션에 알립니다.
OnBatteryPower SENS는 OnBatteryPower 메서드를 호출하여 컴퓨터가 배터리 전원을 사용하고 있음을 애플리케이션에 알립니다.
PostShell 사용자가 로그온하고 Windows Explorer 실행 중인 경우 PostShell 메서드를 사용합니다. 셸이 아직 실행되지 않을 때 로그온 후에 로그온이 호출되기 때문에 이 메서드는 Logon 메서드와 다릅니다.
SessionDisconnect SessionDisconnect 메서드는 빠른 사용자 전환 세션 또는 원격 데스크톱 연결에서 연결을 끊는 데

사용됩니다. 이 메서드를 사용하면 세션의 연결이 끊어지므로 세션에서 로그오프하는 경우와 다릅니다.
SessionReconnect 세션이 다시 연결되었습니다. SessionReconnect 메서드는 빠른 사용자 전환 세션 또는 원격 데스크톱 연결에 다시 연결할 때 사용됩니다. 이는 새 세션에 로그온하는 경우와 다릅니다.
StartScreenSaver StartScreenSaver 메서드는 화면 보호기가 시작되었음을 애플리케이션에 알릴 수 있습니다.
StartShell StartShell 메서드는 애플리케이션에 셸이 시작되었음을 알릴 수 있습니다.
StopScreenSaver StopScreenSaver 메서드는 화면 보호기가 중지되었음을 애플리케이션에 알릴 수 있습니다.

인터페이스

ISensLogon ISensLogon 인터페이스는 SENS에서 발생한 로그온 이벤트를 처리합니다.
ISensLogon2 ISensLogon2 인터페이스는 SENS에서 발생한 로그온 이벤트를 처리합니다.
ISensNetwork ISensNetwork 인터페이스는 SENS(시스템 이벤트 알림 서비스)에서 발생한 네트워크 이벤트를 처리합니다.
ISensOnNow ISensOnNow 인터페이스는 SENS(시스템 이벤트 알림 서비스)에서 발생한 AC 및 배터리 전원 이벤트를 처리합니다.

구조체

QOCINFO QOCINFO 구조체는 IsDestinationReachable 함수에 의해 반환되며 호출자에게 연결 품질 정보를 제공합니다.
SENS_QOCINFO SENS_QOCINFO 구조체는 ISensNetwork::ConnectionMade 메서드에서 사용됩니다. 이 구조체에는 SENS를 구독하는 애플리케이션의 싱크 개체에 대한 연결 품질 정보가 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

시스템 모니터

아티클2023. 08. 22.

시스템 모니터 기술 개요입니다.

시스템 모니터를 개발하려면 다음 헤더가 필요합니다.

- isysmon.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- System Monitor(시스템 모니터)

열거형

DataSourceTypeConstants 성능 카운터 데이터의 원본을 결정합니다.
DisplayTypeConstants 예를 들어 성능 카운터 데이터가 꺾은선형 그래프 또는 히스토그램으로 그래프로 표시되는 방식을 결정합니다.
ReportValueTypeConstants 히스토그램 및 보고서 뷰 그래프가 샘플링된 마지막 값인지 아니면 샘플링 기간의 값(예: 평균 또는 최소값)을 사용하여 계산된 값을 그래프로 표시할지 결정합니다.
SysmonBatchReason 시스템 모니터를 잠그는 이유를 정의합니다.
SysmonDataType 그래프의 지정된 데이터 요소에서 반환할 데이터 형식을 결정합니다.
SysmonFileType 카운터 데이터가 파일에 저장되는 형식을 결정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

시스템 복원

아티클2023. 03. 09.

시스템 복원 기술 개요입니다.

시스템 복원을 개발하려면 다음 헤더가 필요합니다.

- srrestoreptapi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 시스템 복원

함수

SRRemoveRestorePoint 지정된 복원 지점을 삭제합니다.
SRSetRestorePointA 시스템 복원에서 복원 지점을 만들 수 있도록 변경 내용 집합의 시작 및 끝을 지정합니다. (ANSI)
SRSetRestorePointW 시스템 복원에서 복원 지점을 만들 수 있도록 변경 내용 집합의 시작 및 끝을 지정합니다. (유니코드)

구조체

RESTOREPOINTINFOA SRSetRestorePoint 함수에서 사용하는 정보를 포함합니다. (ANSI)
RESTOREPOINTINFOW SRSetRestorePoint 함수에서 사용하는 정보를 포함합니다. (유니코드)
STATEMGRSTATUS SRSetRestorePoint 함수에서 사용하는 상태 정보를 포함합니다.

시스템 서비스

아티클2023. 03. 11.

System Services 기술 개요입니다.

System Services를 개발하려면 다음 헤더가 필요합니다.

- atlthunk.h
- avrt.h
- dbghelp.h
- dbt.h
- debugapi.h
- dispatcherqueue.h
- enclaveapi.h
- errhandlingapi.h
- fibersapi.h
- handleapi.h
- heapapi.h
- imagehlp.h
- interlockedapi.h
- ioapiset.h
- jobapi.h
- jobapi2.h
- libloaderapi.h
- libloaderapi2.h
- mcx.h
- memoryapi.h
- minidumpapiset.h
- minwinbase.h
- namedpipeapi.h
- namespaceapi.h
- ntenclv.h
- powerbase.h
- powersetting.h
- powrprof.h
- processenv.h
- processtopologyapi.h
- profileapi.h
- pwm.h
- realtimeapiset.h

- rtworkq.h
- secext.h
- storprop.h
- synchapi.h
- sysinfoapi.h
- systemtopologyapi.h
- threadpoolapiset.h
- threadpoollegacyapiset.h
- timeprov.h
- timezoneapi.h
- utilapiset.h
- vds.h
- vdshwprv.h
- vdslun.h
- versionhelpers.h
- vsadmin.h
- vsbackup.h
- vsmgmt.h
- vsprov.h
- vss.h
- vswriter.h
- waasapi.h
- waasapitypes.h
- wct.h
- windowsceip.h
- winenclaveapi.h
- winioctl.h
- winternl.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 오류 처리
- 디버그 도움말 라이브러리
- 통신 리소스
- 동적 연결 라이브러리
- 이벤트 로깅
- Interprocess Communications
- 메모리 관리
- 전원 관리
- Services
- 시스템 종료

- 동기화
- Windows 시스템 정보
- 가상 디스크 서비스
- VDS 참조
- 볼륨 섀도 복사본 서비스

클래스

CVssWriter CVssWriter 클래스는 작성기가 VSS 및 기타 작성기와 상태를 동기화하는 인터페이스를 정의하는 추상 기본 클래스입니다.
CVssWriterEx CVssWriterEx 클래스는 작성기가 VSS 및 기타 작성기와 상태를 동기화하는 인터페이스를 정의하는 추상 기본 클래스입니다.
CVssWriterEx2 CVssWriterEx2 클래스는 작성기가 VSS 및 기타 작성기와 상태를 동기화하는 인터페이스를 정의하는 추상 기본 클래스입니다.
IVssBackupComponents IVssBackupComponents 인터페이스는 요청자가 파일 상태에 대해 기록기를 폴링하고 백업/복원 작업을 실행하는 데 사용됩니다.
IVssBackupComponentsEx 요청자가 여러 기록기 인스턴스를 사용하여 백업 및 복원 작업을 실행할 수 있는 메서드를 제공합니다.
IVssBackupComponentsEx2 요청자가 백업 및 복원 작업을 실행하는 데 사용할 수 있는 추가 메서드를 정의합니다.
IVssBackupComponentsEx3 요청자가 LUN 다시 동기화를 수행하고 확장 작성기 상태 정보를 반환하는 데 사용할 수 있는 추가 메서드를 정의합니다.
IVssBackupComponentsEx4 요청자의 UNC 파일 공유 경로 처리를 지원하는 추가 메서드를 정의합니다.

IVssComponent IVssComponent 인터페이스는 요청자의 백업 구성 요소 문서에 포함된 구성 요소에 대한 정보를 검사하고 수정하는 방법을 포함하는 C++(COM 아님) 인터페이스입니다.
IVssComponentEx 요청자의 백업 구성 요소 문서에 포함된 구성 요소에 대한 정보를 검사하고 수정하기 위한 추가 메서드를 정의합니다.
IVssComponentEx2 구성 요소 수준 기록기 오류를 보고하고 검색하기 위한 추가 메서드를 정의합니다.
IVssCreateExpressWriterMetadata IVssCreateExpressWriterMetadata 인터페이스는 빠른 작성기에 대한 기록기 메타데이터 문서를 생성하는 메서드를 포함하는 COM 인터페이스입니다.
IVssCreateWriterMetadata IVssCreateWriterMetadata 인터페이스는 Identify 이벤트에 대한 응답으로 기록기 메타데이터 문서를 생성하는 메서드를 포함하는 C++(COM 아님) 인터페이스입니다. CVssWriter::OnIdentify 메서드에서만 사용됩니다.
IVssCreateWriterMetadataEx IVssCreateWriterMetadataEx 인터페이스는 섀도 복사본을 만들 때 명시적으로 제외될 파일 집합을 보고하는 메서드를 정의하는 C++(COM 아님) 인터페이스입니다.
IVssExamineWriterMetadata IVssExamineWriterMetadata 인터페이스는 요청자가 특정 기록기 인스턴스의 메타데이터를 검사할 수 있는 C++(COM 아님) 인터페이스입니다.
IVssExamineWriterMetadataEx 작성기 인스턴스 이름 및 특정 기록기 인스턴스에 대한 기타 기본 정보를 검색하는 메서드를 제공합니다.
IVssExamineWriterMetadataEx2 특정 기록기 인스턴스에 대한 버전 정보 및 기타 기본 정보를 검색하는 메서드를 정의합니다.
IVssExpressWriter VSS Express Writer에 대한 메타데이터를 관리하는 메서드를 정의합니다.

IVssWMComponent IVssWMComponent는 기록기 메타데이터 문서에 저장된 구성 요소 정보에 액세스할 수 있는 C++(COM 아님) 인터페이스입니다.
IVssWMDependency IVssWMDependency는 IVssWMComponent 인터페이스에서 반환되고 다른 작성기에서 관리하는 구성 요소에 대한 명시적 기록기 구성 요소 종속성이 있는 구성 요소를 백업하거나 복원할 때 애플리케이션에서 사용하는 C++(COM 아님) 인터페이스입니다.
IVssWMFiledesc IVssWMFiledesc 인터페이스는 여러 쿼리 메서드에 의해 호출 애플리케이션에 반환된 C++(COM 아님) 인터페이스입니다. 파일 또는 파일 집합(파일 집합)에 대한 자세한 정보를 제공합니다.
IVssWriterComponents 구성 요소 정보를 가져오고 수정하는 데 사용되는 메서드를 포함합니다.
IVssWriterComponentsExt IVssWriterComponentsExt 인터페이스는 요청자가 백업과 관련된 기록기의 구성 요소에 액세스하고 수정하는 데 사용하는 C++(COM 아님) 인터페이스입니다.

IOCTLs

IOCTL_CHANGER_EXCHANGE_MEDIUM 원본 요소에서 하나의 대상으로 미디어 조각을 이동하고 원래 첫 번째 대상에 있는 미디어 부분을 두 번째 대상으로 이동합니다.
IOCTL_CHANGER_GET_ELEMENT_STATUS 모든 요소의 상태 또는 특정 형식의 지정된 수의 요소를 검색합니다.
IOCTL_CHANGER_GET_PARAMETERS 지정된 디바이스의 매개 변수를 검색합니다.
IOCTL_CHANGER_GET_PRODUCT_DATA 지정된 디바이스의 제품 데이터를 검색합니다.

IOCTL_CHANGER_GET_STATUS 지정된 디바이스의 현재 상태를 검색합니다.
IOCTL_CHANGER_INITIALIZE_ELEMENT_STATUS 모든 요소 또는 특정 형식의 지정된 요소의 상태를 초기화합니다.
IOCTL_CHANGER_MOVE_MEDIUM 미디어 조각을 대상으로 이동합니다.
IOCTL_CHANGER_QUERY_VOLUME_TAGS 지정된 요소에 대한 볼륨 태그 정보를 검색합니다.
IOCTL_CHANGER_REINITIALIZE_TRANSPORT 전송 요소를 물리적으로 다시 정정합니다. 재보정에는 전송을 홈 위치로 반환하는 작업이 포함될 수 있습니다.
IOCTL_CHANGER_SET_ACCESS 디바이스의 삽입/배출 포트, 도어 또는 키패드의 상태를 설정합니다.
IOCTL_CHANGER_SET_POSITION 체인저의 로봇 전송 메커니즘을 지정된 요소 주소로 설정합니다. 이렇게 하면 전송을 미리 배치하여 미디어를 이동하거나 교환할 수 있습니다.
IOCTL_PWM_CONTROLLER_GET_ACTUAL_PERIOD PWM(Pulse Width Modulation) 컨트롤러의 유효 출력 신호 기간을 출력 채널에서 측정할 때 검색합니다.
IOCTL_PWM_CONTROLLER_GET_INFO PWM(Pulse Width Modulation) 컨트롤러에 대한 정보를 검색합니다. 컨트롤러가 초기화된 후에는 이 정보가 변경되지 않습니다.
IOCTL_PWM_CONTROLLER_SET_DESIRED_PERIOD PWM(Pulse Width Modulation) 컨트롤러의 출력 신호 기간을 제안된 값으로 설정합니다.
IOCTL_PWM_PIN_GET_ACTIVE_DUTY_CYCLE_PERCENTAGE 핀 또는 채널의 현재 의무 주기 백분율을 검색합니다. 컨트롤 코드는 백분율을 PWM_PIN_GET_ACTIVE_DUTY_CYCLE_PERCENTAGE_OUTPUT 구조체로 반환합니다.

IOCTL_PWM_PIN_GET_POLARITY 핀 또는 채널의 현재 신호 극성을 검색합니다. 컨트롤 코드는 신호 극성을 PWM_PIN_GET_POLARITY_OUTPUT 구조체로 가져옵니다. 신호 극성은 PWM_POLARITY 열거형에 정의된 대로 활성 높음 또는 활성 낮음입니다.
IOCTL_PWM_PIN_IS_STARTED 핀 또는 채널에 대한 신호 생성 상태를 검색합니다. 각 핀의 상태는 PWM_PIN_IS_STARTED_OUTPUT 구조체로 시작되거나 중지됩니다.
IOCTL_PWM_PIN_SET_ACTIVE_DUTY_CYCLE_PERCENTAGE 컨트롤러 핀 또는 채널에 대해 원하는 의무 주기 백분율 값을 설정합니다. 컨트롤 코드는 백분율을 PWM_PIN_SET_ACTIVE_DUTY_CYCLE_PERCENTAGE_INPUT 구조체로 지정합니다.
IOCTL_PWM_PIN_SET_POLARITY 핀 또는 채널의 신호 극성을 설정합니다. 컨트롤 코드는 PWM_PIN_SET_POLARITY_INPUT 구조에 따라 신호 극성을 설정합니다. 신호 극성은 PWM_POLARITY 열거형에 정의된 대로 활성 높음 또는 활성 낮음입니다.
IOCTL_PWM_PIN_START 핀 또는 채널에서 PWM(Pulse Width 변조) 신호 생성을 시작합니다. 핀이 시작되었는지 확인하려면 IOCTL_PWM_PIN_IS_STARTED 사용합니다.
IOCTL_PWM_PIN_STOP 핀 또는 채널에서 PWM(Pulse Width Modulation) 신호 생성을 중지합니다. 핀이 시작되었는지 확인하려면 IOCTL_PWM_PIN_IS_STARTED 사용합니다.
IOCTL_SERIAL_LSRMST_INSERT 애플리케이션이 ReadFile 함수를 통해 획득하는 일반 데이터 스트림에 줄 상태 및 모뎀 상태 값의 배치를 사용하거나 사용하지 않도록 설정합니다.
IOCTL_STORAGE_CHECK_VERIFY 디바이스에 대한 미디어에 액세스할 수 있는지 여부를 결정합니다.
IOCTL_STORAGE_EJECT_MEDIA SCSI 디바이스에서 미디어를 배출합니다.
IOCTL_STORAGE_EJECTION_CONTROL 미디어를 꺼내는 메커니즘을 사용하거나 사용하지 않도록 설정합니다. 메커니즘을 사용하지 않도록 설정하면 드라이브가 잠깁니다.

IOCTL_STORAGE_GET_DEVICE_NUMBER 디바이스 유형, 디바이스 번호 및 분할 가능한 디바이스의 경우 디바이스의 파티션 번호를 검색합니다.
IOCTL_STORAGE_GET_HOTPLUG_INFO 지정된 디바이스의 핫플러그 구성을 검색합니다.
IOCTL_STORAGE_GET_MEDIA_SERIAL_NUMBER USB 디바이스의 일련 번호를 검색합니다.
IOCTL_STORAGE_GET_MEDIA_TYPES 디바이스에 대한 기하 도형 정보를 검색합니다. (IOCTL_STORAGE_GET_MEDIA_TYPES)
IOCTL_STORAGE_GET_MEDIA_TYPES_EX 디바이스에서 지원하는 미디어 유형에 대한 정보를 검색합니다.
IOCTL_STORAGE_LOAD_MEDIA 디바이스에 미디어를 로드합니다.
IOCTL_STORAGE_MANAGE_DATA_SET_ATTRIBUTES IOCTL_STORAGE_MANAGE_DATA_SET_ATTRIBUTES 제어 코드는 볼륨 관리자 및 스토리지 시스템 디바이스에 특성 정보를 전달합니다.
IOCTL_STORAGE_MCN_CONTROL 미디어 변경 알림을 사용하거나 사용하지 않도록 설정합니다. 미디어 변경 알림을 사용하지 않도록 설정하면 GUID_IO_MEDIA_ARRIVAL 및 GUID_IO_MEDIA_REMOVAL 이벤트가 방지됩니다.
IOCTL_STORAGE_MEDIA_REMOVAL 잠금 기능을 소유한 디바이스에 대해 미디어를 꺼내는 메커니즘을 사용하거나 사용하지 않도록 설정합니다.
IOCTL_STORAGE_READ_CAPACITY 디바이스에 대한 기하 도형 정보를 검색합니다. (IOCTL_STORAGE_READ_CAPACITY)
IOCTL_STORAGE_SET_HOTPLUG_INFO 지정된 디바이스의 핫플러그 구성을 설정합니다.

열거형

__VDS_PARTITION_STYLE 이 열거형은 명시적으로 사용되지 않습니다.
EFFECTIVE_POWER_MODE 시스템이 실행 중인 유효 전원 모드를 나타냅니다.
ELEMENT_TYPE 변경자 디바이스의 요소 유형을 지정합니다.
ENCLAVE_SEALING_IDENTITY_POLICY 데이터를 봉인 해제하기 위해 Enclave에 대해 EnclaveSealData를 호출하는 Enclave와 다른 Enclave를 어떻게 관련시켜야 하는지 지정하는 값을 정의합니다.
HEAP_INFORMATION_CLASS 설정하거나 검색할 힙 정보의 클래스를 지정합니다.
MEM_EXTENDED_PARAMETER_TYPE 주소 공간에 파일 매핑에 사용되는 확장 매개 변수의 값을 정의합니다.
POWER_ACTION 시스템 전원 작업 유형을 지정하는 데 사용되는 값을 정의합니다.
POWER_DATA_ACCESSOR PowerEnumerate 및 PowerSettingAccessCheck에서 사용하는 열거형 값입니다.
POWER_PLATFORM_ROLE OEM의 기본 전원 관리 프로필을 나타냅니다.
PWM_POLARITY 의무 주기와 관련하여 PWM(Pulse Width Modulation) 신호 수준 극성을 나타냅니다.
STORAGE_BUS_TYPE 다양한 유형의 스토리지 버스를 지정합니다.

STORAGE_MEDIA_TYPE 다양한 유형의 스토리지 미디어를 지정합니다.
SYSTEM_POWER_CONDITION GUID_ACDC_POWER_SOURCE 전원 이벤트에서 현재 전원을 나타내는 데 사용됩니다.
SYSTEM_POWER_STATE 시스템 전원 상태를 지정하는 데 사용되는 값을 정의합니다.
VDS_ASYNC_OUTPUT_TYPE VDS_ASYNC_OUTPUT_TYPE(vds.h) 열거형은 개체가 처리할 수 있는 작업 집합을 정의합니다.
VDS_ASYNC_OUTPUT_TYPE VDS_ASYNC_OUTPUT_TYPE 열거형(vdshwprv.h)은 개체가 처리할 수 있는 작업 집합을 정의합니다.
VDS_CONTROLLER_STATUS VDS_CONTROLLER_STATUS(vds.h) 열거형은 컨트롤러의 개체 상태 값 집합을 정의합니다.
VDS_CONTROLLER_STATUS VDS_CONTROLLER_STATUS 열거형(vdshwprv.h)은 컨트롤러의 개체 상태 값 집합을 정의합니다.
VDS_DISK_EXTENT_TYPE 디스크 익스텐트 유형 집합을 정의합니다. 형식은 파티션, 볼륨 또는 여유 공간이 될 수 있습니다.
VDS_DISK_FLAG 디스크 개체의 유효한 플래그 집합을 정의합니다. 참고 VDS_DF_READ_ONLY 경우를 제외하고 이러한 플래그는 IVdsDisk::SetFlags 메서드를 사용하여 설정하거나 IVdsDisk::ClearFlags 메서드를 사용하여 지울 수 없습니다.
VDS_DISK_OFFLINE_REASON 디스크가 오프라인 상태가 되는 이유 집합을 정의합니다.
VDS_DISK_STATUS 디스크의 개체 상태 값 집합을 정의합니다.
VDS_DRIVE_FLAG VDS_DRIVE_FLAG(vds.h) 열거형은 드라이브 개체에 대한 유효한 플래그 집합을 정의합니다.

VDS_DRIVE_FLAG VDS_DRIVE_FLAG 열거형(vdshwprv.h)은 드라이브 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_DRIVE_LETTER_FLAG 드라이브 문자에 대한 유효한 플래그 집합을 정의합니다.
VDS_DRIVE_STATUS VDS_DRIVE_STATUS(vds.h) 열거형은 드라이브의 개체 상태 값 집합을 정의합니다.
VDS_DRIVE_STATUS VDS_DRIVE_STATUS 열거형(vdshwprv.h)은 드라이브의 개체 상태 값 집합을 정의합니다.
VDS_FILE_SYSTEM_FLAG 파일 시스템의 유효한 플래그 집합을 정의합니다.
VDS_FILE_SYSTEM_FORMAT_SUPPORT_FLAG 볼륨 서식 지정에 지원되는 파일 시스템의 속성을 정의합니다.
VDS_FILE_SYSTEM_PROP_FLAG 파일 시스템 압축의 세부 정보를 정의합니다.
VDS_FILE_SYSTEM_TYPE VDS_FILE_SYSTEM_TYPE(vds.h) 열거형은 파일 시스템에 유효한 형식 집합을 정의합니다.
VDS_FILE_SYSTEM_TYPE VDS_FILE_SYSTEM_TYPE 열거형(vdshwprv.h)은 파일 시스템에 유효한 형식 집합을 정의합니다.
VDS_FORMAT_OPTION_FLAGS IVdsDiskPartitionMF2::FormatPartitionEx2 메서드에 대한 유효한 서식 옵션 집합을 정의합니다.
VDS_HBAPORT_SPEED_FLAG VDS_HBAPORT_SPEED_FLAG(vds.h) 열거형은 HBA 포트에서 지원하는 속도를 결정하기 위한 유효한 플래그 집합을 정의합니다.
VDS_HBAPORT_SPEED_FLAG VDS_HBAPORT_SPEED_FLAG 열거형(vdshwprv.h)은 HBA 포트에서 지원하는 속도를 결정하기 위한 유효한 플래그 집합을 정의합니다.

VDS_HBAPORT_STATUS VDS_HBAPORT_STATUS(vds.h) 열거형은 HBA 포트에 대한 유효한 상태 집합을 정의합니다.
VDS_HBAPORT_STATUS VDS_HBAPORT_STATUS 열거형(vdshwprv.h)은 HBA 포트에 대한 유효한 상태 집합을 정의합니다.
VDS_HBAPORT_TYPE VDS_HBAPORT_TYPE(vds.h) 열거형은 HBA 포트에 유효한 형식 집합을 정의합니다.
VDS_HBAPORT_TYPE VDS_HBAPORT_TYPE 열거형(vdshwprv.h)은 HBA 포트에 유효한 형식 집합을 정의합니다.
VDS_HEALTH VDS_HEALTH(vds.h) 열거형은 VDS 개체의 상태 값 집합을 정의합니다.
VDS_HEALTH VDS_HEALTH 열거형(vdshwprv.h)은 VDS 개체의 상태 값 집합을 정의합니다.
VDS_HWPROVIDER_TYPE VDS_HWPROVIDER_TYPE(vds.h) 열거형은 하드웨어 공급자에 대한 유효한 형식 집합을 정의합니다.
VDS_HWPROVIDER_TYPE VDS_HWPROVIDER_TYPE 열거형(vdshwprv.h)은 하드웨어 공급자에 대한 유효한 형식 집합을 정의합니다.
VDS_INTERCONNECT_ADDRESS_TYPE 실제 상호 연결의 유효한 주소 형식 집합을 정의합니다.
VDS_INTERCONNECT_FLAG VDS_INTERCONNECT_FLAG(vds.h) 열거형은 하위 시스템이 지원할 수 있는 상호 연결 형식 집합을 정의합니다.
VDS_INTERCONNECT_FLAG VDS_INTERCONNECT_FLAG 열거형(vdshwprv.h)은 하위 시스템이 지원할 수 있는 상호 연결 형식 집합을 정의합니다.

VDS_IPADDRESS_TYPE VDS_IPADDRESS_TYPE(vds.h) 열거형은 IP 주소에 유효한 형식 집합을 정의합니다.
VDS_IPADDRESS_TYPE VDS_IPADDRESS_TYPE 열거형(vdshwprv.h)은 IP 주소에 유효한 형식 집합을 정의합니다.
VDS_ISCSI_AUTH_TYPE VDS_ISCSI_AUTH_TYPE(vds.h) 열거형은 iSCSI 대상에 로그인할 때 인증에 유효한 형식 집합을 정의합니다.
VDS_ISCSI_AUTH_TYPE VDS_ISCSI_AUTH_TYPE 열거형(vdshwprv.h)은 iSCSI 대상에 로그인할 때 인증에 유효한 형식 집합을 정의합니다.
VDS_ISCSI_IPSEC_FLAG VDS_ISCSI_IPSEC_FLAG(vds.h) 열거형은 지원되지 않지만 나중에 사용하도록 예약되어 있습니다.
VDS_ISCSI_IPSEC_FLAG VDS_ISCSI_IPSEC_FLAG 열거형(vdshwprv.h)은 지원되지 않습니다. 이 열거형은 나중에 사용하도록 예약되어 있습니다.
VDS_ISCSI_LOGIN_FLAG VDS_ISCSI_LOGIN_FLAG(vds.h) 열거형은 iSCSI 대상 로그인 옵션을 지정하기 위한 유효한 플래그 집합을 정의합니다.
VDS_ISCSI_LOGIN_FLAG VDS_ISCSI_LOGIN_FLAG 열거형(vdshwprv.h)은 iSCSI 대상 로그인 옵션을 지정하기 위한 유효한 플래그 집합을 정의합니다.
VDS_ISCSI_LOGIN_TYPE VDS_ISCSI_LOGIN_TYPE(vds.h) 열거형은 iSCSI 대상에 로그인하기 위한 유효한 형식 집합을 정의합니다.
VDS_ISCSI_LOGIN_TYPE VDS_ISCSI_LOGIN_TYPE 열거형(vdshwprv.h)은 iSCSI 대상에 로그인하기 위한 유효한 형식 집합을 정의합니다.

VDS_ISCSI_PORTAL_STATUS VDS_ISCSI_PORTAL_STATUS(vds.h) 열거형은 iSCSI 포털의 유효한 상태 값 집합을 정의합니다.
VDS_ISCSI_PORTAL_STATUS VDS_ISCSI_PORTAL_STATUS 열거형(vdshwprv.h)은 iSCSI 포털의 유효한 상태 값 집합을 정의합니다.
VDS_LOADBALANCE_POLICY_ENUM VDS_LOADBALANCE_POLICY_ENUM(vds.h) 열거형은 경로에 대한 유효한 부하 분산 정책 집합을 정의합니다.
VDS_LOADBALANCE_POLICY_ENUM VDS_LOADBALANCE_POLICY_ENUM 열거형(vdshwprv.h)은 경로에 대한 유효한 부하 분산 정책 집합을 정의합니다.
VDS_LUN_FLAG VDS_LUN_FLAG(vds.h) 열거형은 LUN 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_LUN_FLAG VDS_LUN_FLAG 열거형(vdshwprv.h)은 LUN 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_LUN_PLEX_FLAG VDS_LUN_PLEX_FLAG(vds.h) 열거형은 LUN plex 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_LUN_PLEX_FLAG VDS_LUN_PLEX_FLAG 열거형(vdshwprv.h)은 LUN plex 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_LUN_PLEX_STATUS VDS_LUN_PLEX_STATUS(vds.h) 열거형은 LUN 플렉스에 대한 개체 상태 값 집합을 정의합니다.
VDS_LUN_PLEX_STATUS VDS_LUN_PLEX_STATUS 열거형(vdshwprv.h)은 LUN 플렉스에 대한 개체 상태 값 집합을 정의합니다.
VDS_LUN_PLEX_TYPE VDS_LUN_PLEX_TYPE(vds.h) 열거형은 LUN 플렉스에 유효한 형식 집합을 정의합니다.

VDS_LUN_PLEX_TYPE VDS_LUN_PLEX_TYPE 열거형(vdshwprv.h)은 LUN 플렉스에 유효한 형식 집합을 정의합니다.
VDS_LUN_RESERVE_MODE 지원되지 않습니다. 이 열거형은 나중에 사용하도록 예약되어 있습니다. (VDS_LUN_RESERVE_MODE)
VDS_LUN_STATUS VDS_LUN_STATUS(vds.h) 열거형은 LUN에 대한 개체 상태 값 집합을 정의합니다.
VDS_LUN_STATUS VDS_LUN_STATUS 열거형(vdshwprv.h)은 LUN에 대한 개체 상태 값 집합을 정의합니다.
VDS_LUN_TYPE VDS_LUN_TYPE(vds.h) 열거형은 LUN에 유효한 형식 집합을 정의합니다.
VDS_LUN_TYPE VDS_LUN_TYPE 열거형(vdshwprv.h)은 LUN에 유효한 형식 집합을 정의합니다.
VDS_MAINTENANCE_OPERATION VDS_MAINTENANCE_OPERATION(vds.h) 열거형은 유효한 하위 시스템 유지 관리 작업 집합을 정의합니다.
VDS_MAINTENANCE_OPERATION VDS_MAINTENANCE_OPERATION 열거형(vdshwprv.h)은 유효한 하위 시스템 유지 관리 작업 집합을 정의합니다.
VDS_NOTIFICATION_TARGET_TYPE VDS_NOTIFICATION_TARGET_TYPE(vds.h) 열거형은 VDS 알림의 유효한 대상 유형(주체) 집합을 정의합니다.
VDS_NOTIFICATION_TARGET_TYPE VDS_NOTIFICATION_TARGET_TYPE 열거형(vdshwprv.h)은 VDS 알림의 유효한 대상 유형 집합을 정의합니다.
VDS_OBJECT_TYPE VDS_OBJECT_TYPE(vds.h) 열거형은 VDS 개체의 유효한 형식 집합을 정의합니다.

VDS_OBJECT_TYPE VDS_OBJECT_TYPE 열거형(vdshwprv.h)은 VDS 개체의 유효한 형식 집합을 정의합니다.
VDS_PACK_FLAG 팩 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_PACK_STATUS 팩의 개체 상태 값 집합을 정의합니다.
VDS_PARTITION_FLAG 파티션에 대한 유효한 플래그 집합을 정의합니다.
VDS_PARTITION_STYLE 파티션 스타일 값 집합을 정의합니다.
VDS_PATH_STATUS VDS_PATH_STATUS(vds.h) 열거형은 포트에 대한 유효한 상태 값 집합을 정의합니다.
VDS_PATH_STATUS VDS_PATH_STATUS 열거형(vdshwprv.h)은 포트에 대한 유효한 상태 값 집합을 정의합니다.
VDS_PORT_STATUS VDS_PORT_STATUS(vds.h) 열거형은 포트에 대한 개체 상태 값 집합을 정의합니다.
VDS_PORT_STATUS VDS_PORT_STATUS 열거형(vdshwprv.h)은 포트에 대한 개체 상태 값 집합을 정의합니다.
VDS_PROVIDER_FLAG VDS_PROVIDER_FLAG(vds.h) 열거형은 공급자 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_PROVIDER_FLAG VDS_PROVIDER_FLAG 열거형(vdshwprv.h)은 공급자 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_PROVIDER_LBSUPPORT_FLAG VDS_PROVIDER_LBSUPPORT_FLAG(vds.h) 열거형은 하드웨어 공급자가 지원하는 부하 분산 정책을 나타내는 유효한 플래그 집합을 지정합니다.

VDS_PROVIDER_LBSUPPORT_FLAG VDS_PROVIDER_LBSUPPORT_FLAG 열거형(vdshwprv.h)은 하드웨어 공급자가 지원하는 부하 분산 정책을 나타내는 유효한 플래그 집합을 지정합니다.
VDS_PROVIDER_TYPE VDS_PROVIDER_TYPE(vds.h) 열거형은 공급자에 유효한 형식 집합을 정의합니다.
VDS_PROVIDER_TYPE VDS_PROVIDER_TYPE 열거형(vdshwprv.h)은 공급자에 유효한 형식 집합을 정의합니다.
VDS_QUERY_PROVIDER_FLAG 공급자 쿼리 작업에 유효한 플래그 집합을 정의합니다. 호출자는 하드웨어 공급자, 소프트웨어 공급자 또는 둘 다를 쿼리할 수 있습니다.
VDS_RAID_TYPE VDS_RAID_TYPE(vds.h) 열거형은 스토리지 풀의 기본 RAID 형식을 지정하는 데 사용할 수 있는 집합 열거형 값을 정의합니다.
VDS_RAID_TYPE VDS_RAID_TYPE 열거형(vdshwprv.h)은 스토리지 풀의 기본 RAID 형식을 지정하는 데 사용할 수 있는 집합 열거형 값을 정의합니다.
VDS_RECOVER_ACTION VDS_RECOVER_ACTION(vds.h) 열거형은 시스템 사용을 위해 예약되어 있습니다.
VDS_RECOVER_ACTION VDS_RECOVER_ACTION 열거형(vdshwprv.h)은 시스템 사용을 위해 예약되어 있습니다.
VDS_SAN_POLICY 유효한 디스크 SAN 정책 플래그 집합을 정의합니다.
VDS_SERVICE_FLAG 서비스 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_STORAGE_BUS_TYPE 스토리지 디바이스의 유효한 버스 유형 집합을 정의합니다.

VDS_STORAGE_IDENTIFIER_CODE_SET 스토리지 식별자의 유효한 코드 집합(인코딩)을 정의합니다.
VDS_STORAGE_IDENTIFIER_TYPE 스토리지 식별자에 대한 유효한 형식 집합을 정의합니다.
VDS_STORAGE_POOL_STATUS VDS_STORAGE_POOL_STATUS(vds.h) 열거형은 스토리지 풀에 대한 개체 상태 값 집합을 정의합니다.
VDS_STORAGE_POOL_STATUS VDS_STORAGE_POOL_STATUS 열거형(vdshwprv.h)은 스토리지 풀에 대한 개체 상태 값 집합을 정의합니다.
VDS_STORAGE_POOL_TYPE VDS_STORAGE_POOL_TYPE(vds.h) 열거형은 스토리지 풀 유형 집합을 정의합니다.
VDS_STORAGE_POOL_TYPE VDS_STORAGE_POOL_TYPE 열거형(vdshwprv.h)은 스토리지 풀 유형 집합을 정의합니다.
VDS_SUB_SYSTEM_FLAG VDS_SUB_SYSTEM_FLAG(vds.h) 열거형은 하위 시스템 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_SUB_SYSTEM_FLAG VDS_SUB_SYSTEM_FLAG 열거형(vdshwprv.h)은 하위 시스템 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_SUB_SYSTEM_STATUS VDS_SUB_SYSTEM_STATUS(vds.h) 열거형은 하위 시스템에 대한 개체 상태 값 집합을 정의합니다.
VDS_SUB_SYSTEM_STATUS VDS_SUB_SYSTEM_STATUS 열거형(vdshwprv.h)은 하위 시스템에 대한 개체 상태 값 집합을 정의합니다.
VDS_SUB_SYSTEM_SUPPORTED_RAID_TYPE_FLAG VDS_SUB_SYSTEM_SUPPORTED_RAID_TYPE_FLAG(vds.h) 열거형은 하위 시스템에 의해 지원될 수 있는 RAID 수준 집합을 정의합니다.

VDS_SUB_SYSTEM_SUPPORTED_RAID_TYPE_FLAG VDS_SUB_SYSTEM_SUPPORTED_RAID_TYPE_FLAG 열거형(vdshwprv.h)은 하위 시스템이 지원될 수 있는 RAID 수준 집합을 정의합니다.
VDS_TRANSITION_STATE VDS_TRANSITION_STATE(vds.h) 열거형은 VDS 개체에 대한 유효한 전환 상태 값 집합을 정의합니다.
VDS_TRANSITION_STATE VDS_TRANSITION_STATE 열거형(vdshwprv.h)은 VDS 개체에 대한 유효한 전환 상태 값 집합을 정의합니다.
VDS_VDISK_STATE 가상 디스크 개체의 상태 값 집합을 정의합니다.
VDS_VERSION_SUPPORT_FLAG VDS_VERSION_SUPPORT_FLAG(vds.h) 열거형은 지원되는 VDS 인터페이스 버전을 나타냅니다.
VDS_VERSION_SUPPORT_FLAG VDS_VERSION_SUPPORT_FLAG 열거형(vdshwprv.h)은 지원되는 VDS 인터페이스 버전을 나타냅니다.
VDS_VOLUME_FLAG 볼륨 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_VOLUME_PLEX_STATUS 볼륨 플렉스에 대한 개체 상태 값 집합을 정의합니다.
VDS_VOLUME_PLEX_TYPE 볼륨 플렉스에 유효한 형식 집합을 정의합니다.
VDS_VOLUME_STATUS 볼륨의 개체 상태 값 집합을 정의합니다.
VDS_VOLUME_TYPE 볼륨 개체의 유효한 형식 집합을 정의합니다.

VSS_ALTERNATE_WRITER_STATE 지정된 작성기에 연결된 대체 작성기가 있는지 여부를 나타내는 데 사용됩니다.
VSS_APPLICATION_LEVEL 애플리케이션 수준, 기록기에 동결 알림을 받은 섀도 복사본을 만드는 과정에서의 지점을 나타냅니다.
VSS_BACKUP_SCHEMA 작성기에서 참여할 수 있는 백업 작업 유형을 나타내는 데 사용됩니다.
VSS_BACKUP_TYPE 수행할 백업 유형을 나타냅니다.
VSS_COMPONENT_FLAGS 기록기에서 자동 복구에 대한 지원을 나타내는 데 사용됩니다.
VSS_COMPONENT_TYPE 섀도 복사본 백업 작업과 함께 사용되는 구성 요소의 유형을 지정합니다.
VSS_FILE_RESTORE_STATUS 파일 복원 작업의 상태 집합을 정의합니다.
VSS_FILE_SPEC_BACKUP_TYPE 기록기에서 특정 백업 작업에 대한 지원을 나타내는 데 사용됩니다.
VSS_HARDWARE_OPTIONS 섀도 복사본 LUN 플래그를 정의합니다.
VSS_MGMT_OBJECT_TYPE VSS_MGMT_OBJECT_PROP 구조 내에서 VSS_MGMT_OBJECT_UNION 공용 구조체에 대한 비난.
VSS_OBJECT_TYPE 요청자가 개체를 섀도 복사본 집합, 섀도 복사본 또는 공급자로 식별하는 데 사용됩니다.
VSS_PROTECTION_FAULT 섀도 복사본 보호 오류 집합을 정의합니다.

VSS_PROTECTION_LEVEL 볼륨 섀도 복사본 보호 수준 집합을 정의합니다.
VSS_PROVIDER_CAPABILITIES 이 열거형은 나중에 사용할 수 있습니다.
VSS_PROVIDER_TYPE 공급자 유형을 지정합니다.
VSS_RECOVERY_OPTIONS 요청자가 다시 동기화 작업을 수행하는 방법을 지정하는 데 사용됩니다.
VSS_RESTORE_TARGET 복원 시 작성기가 선택한 구성 요소에 포함된 모든 파일과 정의한 구성 요소 집합의 모든 파일을 복원하는 방법을 나타내는 데 사용됩니다.
VSS_RESTORE_TYPE 요청자가 수행하려는 복원 작업의 유형을 나타내는 데 사용됩니다.
VSS_RESTOREMETHOD_ENUM 기록기 메타데이터 문서를 통해 기본 파일 복원 방법을 지정하기 위해 백업 시 작성기에서 사용합니다.
VSS_ROLLFORWARD_TYPE VSS_ROLLFORWARD_TYPE 열거형은 요청자가 수행하려는 롤 포워드 작업의 유형을 나타내는 데 사용됩니다.
VSS_SNAPSHOT_COMPATIBILITY VSS_SNAPSHOT_COMPATIBILITY 열거형은 섀도 복사된 볼륨에 대해 사용하지 않도록 설정된 볼륨 제어 또는 파일 I/O 작업을 나타냅니다.
VSS_SNAPSHOT_CONTEXT 섀도 복사본을 만들거나 쿼리하거나 삭제하는 방법과 기록기 참여 정도를 지정합니다.
VSS_SNAPSHOT_PROPERTY_ID 섀도 복사본에 설정할 속성을 지정합니다.

VSS_SNAPSHOT_STATE 지정된 섀도 복사 작업의 상태를 지정합니다.
VSS_SOURCE_TYPE 기록기가 관리하는 데이터 형식을 지정합니다.
VSS_SUBSCRIBE_MASK 작성자가 받을 이벤트를 나타냅니다.
VSS_USAGE_TYPE 호스트 시스템에서 VSS 작업과 관련된 기록기에서 관리하는 데이터를 사용하는 방법을 지정합니다.
VSS_VOLUME_SNAPSHOT_ATTRIBUTES 섀도 복사본에 대해 추가 특성을 지정할 수 있습니다.
VSS_WRITER_STATE 작성기의 현재 상태를 나타냅니다.
VSS_WRITERRESTORE_ENUM 복원 작업 중에 생성된 이벤트를 처리할 조건을 요청자에게 나타냅니다.

Functions

~CVssWriter ~CVssWriter는 CVssWriter 클래스 개체의 소멸자입니다.
AbortAllSnapshotsInProgress 이 메서드는 시스템 사용을 위해 예약되어 있습니다.
AbortBackup AbortBackup 메서드는 VSS에 백업 작업이 종료되었음을 알 수 있습니다.

AbortSnapshots 이 공급자에서 준비된 섀도 복사본을 중단합니다.
AbortSystemShutdownA 시작된 시스템 종료를 중지합니다. (ANSI)
AbortSystemShutdownW 시작된 시스템 종료를 중지합니다. (유니코드)
AcquireSRWLockExclusive 전용 모드에서 SRW(슬림 판독기/기록기) 잠금을 획득합니다.
AcquireSRWLockShared 공유 모드에서 SRW(슬림 판독기/기록기) 잠금을 획득합니다.
AddAccessPath 액세스 경로를 추가합니다.
AddAlternateLocationMapping AddAlternateLocationMapping 메서드는 파일 집합에 대한 대체 위치 매핑을 만듭니다.
AddAlternativeLocationMapping AddAlternativeLocationMapping 메서드는 요청자가 지정된 구성 요소에 있는 파일 집합의 모든 멤버를 복원하는 데 대체 위치 매핑을 사용했음을 나타내는 데 사용됩니다.
AddComponent 백업할 빠른 작성기의 구성 요소 집합에 파일 그룹을 추가합니다.
AddComponent AddComponent 메서드는 백업할 구성 요소로 데이터베이스 또는 파일 그룹을 추가합니다.
AddComponent 백업 집합에 명시적으로 추가하는 데 사용됩니다.
AddComponentDependency 빠른 작성기가 관리하는 구성 요소에 명시적 기록기-구성 요소 종속성이 있음을 나타낼 수 있습니다. 즉, 다른 구성 요소(다른 작성기에서 관리할 수 있음)를 백업하고 복원해야 합니다.

AddComponentDependency AddComponentDependency 메서드를 사용하면 작성기가 관리하는 구성 요소에 명시적 기록기-구성 요소 종속성이 있음을 나타낼 수 있습니다. 즉, 다른 작성기에서 다른 구성 요소를 백업하고 복원해야 합니다.
AddDatabaseFiles AddDatabaseFiles 메서드는 백업할 데이터베이스 구성 요소를 구성하는 파일 집합(지정된 파일 또는 파일)을 나타냅니다.
AddDatabaseLogFiles AddDatabaseLogFiles 메서드는 백업할 데이터베이스와 연결된 로그 파일과 해당 위치를 나타냅니다.
AddDiffArea 지정된 볼륨에 대한 섀도 복사본 스토리지 영역 연결을 추가합니다.
AddDifferencedFilesByLastModifyLSN 지원 안 됨 (IVssComponent.AddDifferencedFilesByLastModifyLSN)
AddDifferencedFilesByLastModifyTime 기록기에서 전체 파일을 사용하여 타임스탬프를 증분 또는 차등 백업에 포함하기 위해 마지막 수정 타임스탬프를 사용하여 파일 집합(지정된 파일 또는 파일)을 평가해야 함을 나타내는 데 사용됩니다.
AddDirectedTarget AddDirectedTarget 메서드를 사용하면 작성기가 복원 시 파일을 복원할 때 파일(원본 파일)을 다시 매핑해야 함을 나타낼 수 있습니다.
AddDisk 온라인 팩에 디스크를 추가합니다.
AddDllDirectory 프로세스 DLL 검색 경로에 디렉터리를 추가합니다.
AddExcludeFiles 명시적 작성기의 구성 요소가 백업될 때 암시적으로 포함될 수 있는 파일 집합(지정된 파일 또는 파일)을 제외합니다.

AddExcludeFiles AddExcludeFiles 메서드는 현재 작성기의 구성 요소가 백업될 때 암시적으로 포함될 수 있는 파일 집합(지정된 파일 또는 파일)을 명시적으로 제외하는 데 사용됩니다.
AddExcludeFilesFromSnapshot 섀도 복사본을 만들 때 기록기에서 명시적으로 제외할 파일 집합을 보고합니다.
AddFilesToFileGroup 빠른 작성기에 대해 지정된 파일 그룹 구성 요소에 파일 집합(지정된 파일 또는 파일)을 추가합니다.
AddFilesToFileGroup AddFilesToFileGroup 메서드는 지정된 파일 그룹 구성 요소에 파일 집합(지정된 파일 또는 파일)을 추가합니다.
AddIncludeFiles 지원되지 않습니다. (IVssCreateWriterMetadata.AddIncludeFiles)
AddNewTarget AddNewTarget 메서드는 복원 작업 중에 요청자가 백업 애플리케이션이 파일을 새 위치로 복원할 계획임을 나타내는 데 사용됩니다.
AddPartialFile AddPartialFile 메서드는 지정된 파일의 일부만 백업해야 하며 해당 부분만 백업해야 함을 나타냅니다.
AddPlex IVdsLun::AddPlex 메서드(vds.h)는 LUN을 대상 LUN에 새 플렉스로 추가합니다.
AddPlex 볼륨을 현재 볼륨에 플렉스로 추가합니다.
AddPlex IVdsLun::AddPlex(vdshwprv.h) 메서드는 LUN을 대상 LUN에 새 플렉스로 추가합니다.
AddPortal IVdsIscsiPortalGroup::AddPortal 메서드(vds.h)는 포털 그룹에 포털을 추가합니다.

AddPortal IVdsIscsiPortalGroup::AddPortal(vdshwprv.h) 메서드는 포털 그룹에 포털을 추가합니다.
AddRestoreSubcomponent 백업을 위해 선택 불가능으로 표시되었지만 복원을 위해 선택 가능으로 표시된 구성 요소 집합의 하위 구성 요소 멤버가 복원될 것임을 나타냅니다.
AddSecureMemoryCacheCallback 보안 메모리 범위가 해제되거나 보호가 변경될 때 호출할 콜백 함수를 등록합니다.
AddSnapshotToRecoverySet LUN 다시 동기화 작업에 포함할 볼륨을 지정합니다.
AddToSnapshotSet AddToSnapshotSet 메서드는 원래 볼륨 또는 원래 원격 파일 공유를 섀도 복사본 집합에 추가합니다.
AddVDisk 기존 가상 디스크 파일에 대한 가상 디스크 개체를 만듭니다.
조언 호출자가 VDS 서비스에서 알림을 받도록 호출자의 IVdsAdviseSink 인터페이스를 VDS에 등록합니다.
AllocateUserPhysicalPages 지정된 프로세스의 AWE(주소 창 확장) 영역 내에서 매핑 및 매핑 해제할 실제 메모리 페이지를 할당합니다.
AllocateUserPhysicalPagesNuma 지정된 프로세스의 AWE(주소 창 확장) 영역 내에서 매핑 및 매핑 해제할 실제 메모리 페이지를 할당하고 실제 메모리에 대한 NUMA 노드를 지정합니다.
ApplyHints IVdsLun::ApplyHints 메서드(vds.h)는 LUN에 새 힌트 집합을 적용합니다. LUN에 적용되는 힌트는 모든 플렉스에 동시에 적용됩니다.

ApplyHints IVdsLunPlex::ApplyHints 메서드(vds.h)는 LUN 플렉스에 새 힌트 집합을 적용합니다. 플렉스에 적용된 힌트는 LUN이나 다른 플렉스에 영향을 미치지 않습니다.
ApplyHints IVdsLun::ApplyHints(vdshwprv.h) 메서드는 LUN에 새 힌트 집합을 적용합니다. LUN에 적용되는 힌트는 모든 플렉스에 동시에 적용됩니다.
ApplyHints IVdsLunPlex::ApplyHints(vdshwprv.h) 메서드는 LUN 플렉스에 새 힌트 집합을 적용합니다. 플렉스에 적용된 힌트는 LUN이나 다른 플렉스에 영향을 미치지 않습니다.
ApplyHints2 IVdsLun2::ApplyHints2 메서드(vds.h)는 LUN에 새 힌트 집합을 적용합니다. LUN에 적용되는 힌트는 모든 플렉스에 동시에 적용됩니다.
ApplyHints2 IVdsLun2::ApplyHints2(vdshwprv.h) 메서드는 LUN에 새 힌트 집합을 적용합니다. LUN에 적용되는 힌트는 모든 플렉스에 동시에 적용됩니다.
AreComponentsSelected AreComponentsSelected 메서드는 요청자가 구성 요소 모드에서 실행 중인지 여부를 나타내며 전체 볼륨을 백업하거나 백업할 개별 구성 요소 선택을 지원합니다.
AreLunsSupported 하드웨어 공급자가 볼륨에 기여하는 모든 LUN에 대해 섀도 복사본 만들기를 지원하는지 여부를 결정합니다.
AssignDriveLetter 드라이브 문자를 기존 OEM, ESP 또는 알 수 없는 파티션에 할당합니다.
AssociateControllerPorts IVdsLunControllerPorts::AssociateControllerPorts 메서드(vds.h)는 LUN과 관련하여 하위 시스템 컨트롤러 포트를 활성 또는 비활성으로 설정합니다.
AssociateControllerPorts IVdsLunControllerPorts::AssociateControllerPorts(vdshwprv.h) 메서드는 LUN과 관련하여 하위 시스템 컨트롤러 포트를 활성 또는 비활성으로 설정합니다.

AssociateControllers IVdsLun::AssociateControllers 메서드(vds.h)는 LUN과 관련하여 하위 시스템 컨트롤러를 활성 또는 비활성으로 설정합니다.
AssociateControllers IVdsLun::AssociateControllers(vdshwprv.h) 메서드는 LUN과 관련하여 하위 시스템 컨트롤러를 활성 또는 비활성으로 설정합니다.
AssociateTargets IVdsLunIscsi::AssociateTargets 메서드(vds.h)는 LUN을 하위 시스템 iSCSI 대상과 연결합니다.
AssociateTargets IVdsLunIscsi::AssociateTargets(vdshwprv.h) 메서드는 LUN을 하위 시스템 iSCSI 대상과 연결합니다.
AtlThunk_AllocateData ATL 썽크에 대한 메모리 공간을 할당합니다.
AtlThunk_DataToCode AtlThunkData_t 매개 변수에 해당하는 실행 파일 함수를 반환합니다.
AtlThunk_FreeData ATL 썽크와 연결된 메모리를 해제합니다.
AtlThunk_InitData ATL 썽크를 초기화합니다.
연결 가상 디스크를 연결합니다.
BackupComplete BackupComplete 메서드를 사용하면 VSS에서 BackupComplete 이벤트를 생성하여 기록기에 백업 프로세스가 완료되었음을 알릴 수 있습니다.
BackupEventLogA 지정된 이벤트 로그를 백업 파일에 저장합니다. (ANSI)

BackupEventLogW 지정된 이벤트 로그를 백업 파일에 저장합니다. (유니코드)
BeginPrepareSnapshot VSS는 섀도 복사본 집합에 추가되는 각 섀도 복사본에 대해 이 메서드를 호출합니다. (IVssFileShareSnapshotProvider.BeginPrepareSnapshot)
BeginPrepareSnapshot 섀도 복사본 집합에 추가되는 각 섀도 복사본에 대해 호출됩니다.
BeginPrepareSnapshot VSS는 섀도 복사본 집합에 추가되는 각 섀도 복사본에 대해 이 메서드를 호출합니다. (IVssSoftwareSnapshotProvider.BeginPrepareSnapshot)
BreakPlex 현재 볼륨에서 지정된 플렉스를 제거합니다.
BreakSnapshotSet BreakSnapshotSet 메서드를 사용하면 VSS에서 섀도 복사본 집합이 "잊혀지게" 됩니다.
BreakSnapshotSetEx 요청자가 지정한 옵션에 따라 섀도 복사본 집합을 중단합니다.
BuildCommDCBA 지정된 DCB 구조체를 디바이스 제어 문자열에 지정된 값으로 채웁니다. (ANSI)
BuildCommDCBAndTimeoutsA 디바이스 정의 문자열을 적절한 디바이스 제어 블록 코드로 변환하고 디바이스 제어 블록에 배치합니다. (ANSI)
BuildCommDCBAndTimeoutsW 디바이스 정의 문자열을 적절한 디바이스 제어 블록 코드로 변환하고 디바이스 제어 블록에 배치합니다. (유니코드)
BuildCommDCBW 지정된 DCB 구조체를 디바이스 제어 문자열에 지정된 값으로 채웁니다. (유니코드)

CallEnclave enclave 내에서 함수를 호출합니다.
CallNamedPipeA 메시지 형식 파이프에 연결하고 파이프 인스턴스를 사용할 수 없는 경우 대기하고 파이프에 쓰고 파이프에서 읽은 다음 파이프를 닫습니다. (CallNamedPipeA)
CallNamedPipeW 메시지 형식 파이프에 연결하고 파이프 인스턴스를 사용할 수 없는 경우 대기하고 파이프에 쓰고 파이프에서 읽은 다음 파이프를 닫습니다. (CallNamedPipeW)
CallNtPowerInformation 전원 정보를 설정하거나 검색합니다.
취소 IVdsAsync::Cancel(vds.h) 메서드는 비동기 작업을 취소합니다.
취소 IVdsAsync::Cancel 메서드(vdshwprv.h)는 비동기 작업을 취소합니다.
취소 Cancel 메서드는 불완전한 비동기 작업을 취소합니다.
CancelWaitableTimer 지정된 대기 가능 타이머를 비활성 상태로 설정합니다.
CanUserWritePwrScheme 현재 사용자에게 전원 구성표를 작성할 수 있는 충분한 권한이 있는지 여부를 결정합니다.
CdromDisableDigitalPlayback 지정된 CD-ROM 또는 DVD 드라이브에 대한 디지털 재생을 사용하지 않도록 설정합니다.
CdromEnableDigitalPlayback 지정된 CD-ROM 또는 DVD 드라이브에 디지털 재생을 사용하도록 설정합니다.
CdromIsDigitalPlaybackEnabled 지정된 CD-ROM 또는 DVD 드라이브에 대해 디지털 재생을 사용할 수 있는지 여부를 결정합니다.

CdromKnownGoodDigitalPlayback 지정된 CD-ROM 또는 DVD 드라이브에 양호한 것으로 알려진 디지털 재생이 있는지 여부를 확인합니다.
ChangeAttributes 파티션의 특성을 수정합니다.
ChangeDiffAreaMaximumSize 특정 볼륨의 최대 크기인 섀도 복사본 스토리지 영역을 업데이트.
ChangeDiffAreaMaximumSizeEx 특정 볼륨의 최대 크기인 섀도 복사본 스토리지 영역을 업데이트. 이는 즉각적인 영향을 미치지 않을 수 있습니다. bVolatile 매개 변수가 FALSE이면 컴퓨터가 다시 부팅되더라도 변경 내용이 계속됩니다.
ChangePartitionType 지정된 바이트 오프셋에서 디스크의 파티션 유형을 변경합니다.
정리 파티션 정보를 제거하고 기본 또는 동적 디스크를 초기화하지 않습니다. Windows Server 2003: Clean 메서드는 이동식 디바이스에서 지원되지 않습니다.
CleanupObsoleteMountPoints 더 이상 존재하지 않는 볼륨에 대한 사용자 모드 경로 및 탑재된 폴더를 제거합니다.
ClearCommBreak 지정된 통신 디바이스에 대한 문자 전송을 복원하고 전송 라인을 끊지 않는 상태로 만듭니다.
ClearCommError 통신 오류에 대한 정보를 검색하고 통신 디바이스의 현재 상태를 보고합니다.
ClearEventLogA 지정된 이벤트 로그를 지우고 필요에 따라 로그의 현재 복사본을 백업 파일에 저장합니다. (ANSI)
ClearEventLogW 지정된 이벤트 로그를 지우고 필요에 따라 로그의 현재 복사본을 백업 파일에 저장합니다. (유니코드)

ClearFileSystemFlags 파일 시스템 플래그를 지웁니다.
ClearFlags 디스크 개체의 플래그를 지웁니다.
ClearFlags IVdsDrive::ClearFlags 메서드(vds.h)는 드라이브 개체의 플래그를 지웁니다.
ClearFlags 서비스 개체 플래그를 지웁니다.
ClearFlags 볼륨 플래그를 지웁니다.
ClearFlags IVdsDrive::ClearFlags(vdshwprv.h) 메서드는 드라이브 개체의 플래그를 지웁니다.
ClearVolumeProtectFault 지정된 볼륨의 보호 오류 상태를 지웁니다.
복제 IEnumVdsObject::Clone(vds.h) 메서드는 현재 열거형과 동일한 상태로 열거형을 만듭니다.
복제 IEnumVdsObject::Clone 메서드(vdshwprv.h)는 현재 열거형과 동일한 상태로 열거형을 만듭니다.
복제 IVssEnumObject 열거자 개체의 복사본을 만들어 지정된 열거형 요소 목록의 복사본을 만듭니다.
복제 IVssEnumMgmtObject 열거자 개체의 복사본을 만들어 지정된 열거형 요소 목록의 복사본을 만듭니다.
CloseEventLog 지정된 이벤트 로그를 닫습니다. (CloseEventLog)

CommConfigDialogA 드라이버 제공 구성 대화 상자를 표시합니다. (ANSI)
CommConfigDialogW 드라이버 제공 구성 대화 상자를 표시합니다. (유니코드)
CommitSnapshots 이 공급자의 모든 LUN을 신속하게 커밋합니다.
컴팩트 가상 디스크를 압축하여 백업 파일의 물리적 크기를 줄입니다.
ConnectNamedPipe 명명된 파이프 서버 프로세스가 클라이언트 프로세스가 명명된 파이프의 인스턴스에 연결되기를 기다릴 수 있도록 합니다.
ConvertStyle 빈 디스크의 파티션 스타일을 한 스타일에서 다른 스타일로 변환합니다.
CreateEnclave 초기화되지 않은 새 enclave를 만듭니다. enclave는 애플리케이션의 주소 공간 내에서 코드 및 데이터의 격리된 영역입니다. enclave 내에서 실행되는 코드만 동일한 Enclave 내의 데이터에 액세스할 수 있습니다.
CreateEventA 명명되거나 명명되지 않은 이벤트 개체를 만들거나 엽니다. (ANSI)
CreateEventExA 명명되거나 명명되지 않은 이벤트 개체를 만들거나 열고 개체에 대한 핸들을 반환합니다. (ANSI)
CreateEventExW 명명되거나 명명되지 않은 이벤트 개체를 만들거나 열고 개체에 대한 핸들을 반환합니다. (유니코드)
CreateEventW 명명되거나 명명되지 않은 이벤트 개체를 만들거나 엽니다. (유니코드)

CreateFileMapping2 지정된 파일에 대한 명명되거나 명명되지 않은 파일 매핑 개체를 만들거나 엽니다. 물리적 메모리에 대한 기본 NUMA 노드를 확장 매개 변수로 지정할 수 있습니다. *ExtendedParameters* 매개 변수를 참조하세요.
CreateFileMappingA 지정된 파일에 대한 명명되거나 명명되지 않은 파일 매핑 개체를 만들거나 엽니다. (CreateFileMappingA)
CreateFileMappingFromApp Windows 스토어 앱에서 지정된 파일에 대한 명명되거나 명명되지 않은 파일 매핑 개체를 만들거나 엽니다.
CreateFileMappingNumaA 지정된 파일에 대한 명명되거나 명명되지 않은 파일 매핑 개체를 만들거나 열고 실제 메모리에 대한 NUMA 노드를 지정합니다. (CreateFileMappingNumaA)
CreateFileMappingNumaW 지정된 파일에 대한 명명되거나 명명되지 않은 파일 매핑 개체를 만들거나 열고 실제 메모리에 대한 NUMA 노드를 지정합니다. (CreateFileMappingNumaW)
CreateFileMappingW 지정된 파일에 대한 명명되거나 명명되지 않은 파일 매핑 개체를 만들거나 엽니다. (CreateFileMappingW)
CreateLun IVdsSubSystem::CreateLun 메서드(vds.h)는 LUN(논리 단위 번호)을 만듭니다.
CreateLun IVdsSubSystem::CreateLun(vdshwprv.h) 함수는 LUN(논리 단위 번호)을 만듭니다.
CreateLun2 IVdsSubSystem2::CreateLun2 메서드(vds.h)는 LUN을 만듭니다. 자동 매핑 힌트는 VDS_HINTS 구조 대신 VDS_HINTS2 구조를 사용하여 제공됩니다.
CreateLun2 IVdsSubSystem2::CreateLun2(vdshwprv.h) 메서드는 LUN을 만들고 IVdsSubSystem::CreateLun 메서드와 동일합니다.

CreateLunInStoragePool IVdsHwProviderStoragePools::CreateLunInStoragePool 메서드(vds.h)는 스토리지 풀에 LUN을 만듭니다.
CreateLunInStoragePool IVdsHwProviderStoragePools::CreateLunInStoragePool(vdshwprv.h) 메서드는 스토리지 풀에 LUN을 만듭니다.
CreateMailslotA 지정된 이름의 mailslot을 만들고 mailslot 서버가 mailslot에서 작업을 수행하는 데 사용할 수 있는 핸들을 반환합니다. (ANSI)
CreateMailslotW 지정된 이름의 mailslot을 만들고 mailslot 서버가 mailslot에서 작업을 수행하는 데 사용할 수 있는 핸들을 반환합니다. (유니코드)
CreateMemoryResourceNotification 메모리 리소스 알림 개체를 만듭니다.
CreateMetadata 빠른 작성기 메타데이터 개체를 만들고 IVsCreateExpressWriterMetadata 인터페이스 포인터를 반환합니다.
CreateMutexA 명명되거나 명명되지 않은 뮤텍스 개체를 만들거나 엽니다. (ANSI)
CreateMutexExA 명명되거나 명명되지 않은 뮤텍스 개체를 만들거나 열고 개체에 대한 핸들을 반환합니다. (ANSI)
CreateMutexExW 명명되거나 명명되지 않은 뮤텍스 개체를 만들거나 열고 개체에 대한 핸들을 반환합니다. (유니코드)
CreateMutexW 명명되거나 명명되지 않은 뮤텍스 개체를 만들거나 엽니다. (유니코드)
CreatePack pack 개체를 만듭니다.

CreatePartition 기본 디스크에 파티션을 만듭니다. IVdsCreatePartitionEx::CreatePartitionEx 메서드는 이 메서드를 대체합니다.
CreatePartitionEx 기본 디스크에 파티션을 만듭니다. 이 메서드는 IVdsAdvancedDisk::CreatePartition 메서드를 대체합니다.
CreatePipe 익명 파이프를 만들고 파이프의 읽기 및 쓰기 끝에 대한 핸들을 반환합니다.
CreatePortalGroup IVdsIscsiTarget::CreatePortalGroup 메서드(vds.h)는 포털 그룹을 만듭니다.
CreatePortalGroup IVdsIscsiTarget::CreatePortalGroup(vdshwprv.h) 메서드는 포털 그룹을 만들고 새 포털 그룹 개체에 대한 인터페이스 포인터를 검색할 수 있습니다.
CreateSemaphoreExW 명명되거나 명명되지 않은 세마포 개체를 만들거나 열고 개체에 대한 핸들을 반환합니다. (CreateSemaphoreExW)
CreateSemaphoreW 명명되거나 명명되지 않은 세마포 개체를 만들거나 엽니다. (CreateSemaphoreW)
CreateTarget IVdsSubSystemIscsi::CreateTarget 메서드(vds.h)는 iSCSI 대상을 만듭니다.
CreateTarget IVdsSubSystemIscsi::CreateTarget(vdshwprv.h) 메서드는 iSCSI 대상을 만듭니다.
CreateVDisk 가상 디스크를 만듭니다.
CreateVolume 팩 내에 볼륨을 만듭니다.

CreateVolume2 선택적 맞춤 매개 변수를 사용하여 디스크 팩에 볼륨을 만듭니다.
CreateVssBackupComponents CreateVssBackupComponents 함수(vsbackup.h)는 IVssBackupComponents 인터페이스 개체를 만들고 포인터를 반환합니다.
CreateVssBackupComponentsInternal CreateVssBackupComponents 함수(vsbackup.h)는 IVssBackupComponents 인터페이스 개체를 만들고 포인터를 반환합니다.
CreateVssExamineWriterMetadata CreateVssExamineWriterMetadata 함수(vsbackup.h)는 IVssExamineWriterMetadata 개체를 만듭니다.
CreateVssExamineWriterMetadataInternal CreateVssExamineWriterMetadata 함수(vsbackup.h)는 IVssExamineWriterMetadata 개체를 만듭니다.
CreateVssExpressWriter CreateVssExpressWriter 함수(vswriter.h)는 IVssExpressWriter 인터페이스 개체를 만들고 포인터를 반환합니다.
CreateVssExpressWriterInternal CreateVssExpressWriterInternal 함수(vswriter.h)는 IVssExpressWriter 인터페이스 개체를 만들고 포인터를 반환합니다.
CreateWaitableTimerExW 대기 가능한 타이머 개체를 만들거나 열고 개체에 대한 핸들을 반환합니다.
CreateWaitableTimerW 대기 가능한 타이머 개체를 만들거나 엽니다.
CreateWriter 이 함수는 시스템 사용을 위해 예약되어 있습니다. (CreateWriter)
CreateWriterEx 이 함수는 시스템 사용을 위해 예약되어 있습니다. (CreateWriterEx)

CVssWriter CVssWriter는 CVssWriter 클래스 개체의 생성자입니다.
삭제 IVdsIscsiPortalGroup::D elete 메서드(vds.h)는 포털 그룹을 삭제합니다.
삭제 IVdsIscsiTarget::D사용 취소 함수(vds.h)는 대상과 연결된 LUN이 없는 경우 대상 및 모든 포털 그룹을 삭제합니다.
삭제 IVdsLun::D elete 메서드(vds.h)는 LUN 및 모든 해당 플렉스를 삭제합니다. LUN의 모든 데이터가 손실되고 VDS는 LUN에 할당된 범위를 해제합니다.
삭제 볼륨 및 모든 플렉스를 삭제하고 익스텐트 해제합니다.
삭제 IVdsIscsiPortalGroup::D elete(vdshwprv.h) 메서드는 포털 그룹을 삭제합니다.
삭제 IVdsIscsiTarget::D elete(vdshwprv.h) 메서드는 대상과 연결된 LUN이 없는 경우 대상 및 모든 포털 그룹을 삭제합니다.
삭제 IVdsLun::D elete(vdshwprv.h) 메서드는 LUN 및 모든 해당 플렉스를 삭제합니다. LUN의 모든 데이터가 손실됩니다. VDS는 LUN에 할당된 범위를 해제합니다.
DeleteAccessPath 현재 볼륨에서 액세스 경로를 제거합니다.
DeleteCriticalSection 소유하지 않은 중요한 섹션 개체에서 사용하는 모든 리소스를 해제합니다.
DeleteDriveLetter OEM, ESP 또는 알 수 없는 파티션에 할당된 드라이브 문자를 삭제합니다.

DeleteEnclave 지정된 enclave를 삭제합니다.
DeletePartition 기본 디스크에서 파티션을 삭제합니다.
DeletePwrScheme 지정된 전원 구성표를 삭제합니다.
DeleteSnapshots DeleteSnapshots 메서드는 하나 이상의 섀도 복사본 또는 섀도 복사본 집합을 삭제합니다.
DeleteSnapshots 특정 스냅샷 또는 지정된 스냅샷 집합의 모든 스냅샷을 삭제합니다.
DeleteSnapshots 하나 이상의 섀도 복사본 또는 섀도 복사본 집합을 삭제합니다.
DeleteSynchronizationBarrier 동기화 장벽을 삭제합니다.
DeleteUnusedDiffAreas 사용되지 않는 지정된 볼륨에서 모든 섀도 복사본 스토리지 영역(diff 영역이라고도 함)을 삭제합니다.
DeregisterEventSource 지정된 이벤트 로그를 닫습니다. (DeregisterEventSource)
분리 가상 디스크를 분리합니다.
DetachAndDelete 가상 디스크를 분리하고 백업 파일을 삭제합니다.
DEVICE_NOTIFY_CALLBACK_ROUTINE 애플리케이션의 DeviceNotifyCallbackRoutine 콜백 함수는 전원 알림을 수신하는 데 사용됩니다.

DevicePowerClose 디바이스 목록의 모든 노드를 해제하고 디바이스 목록을 삭제합니다.
DevicePowerEnumDevices 지정된 조건을 충족하는 시스템의 디바이스를 열거합니다.
DevicePowerOpen 모든 디바이스를 쿼리하여 디바이스 목록을 초기화합니다.
DevicePowerSetDeviceState 지정된 디바이스에서 지정된 데이터를 수정합니다.
DisableThreadLibraryCalls 지정된 DLL(동적 연결 라이브러리)에 대한 DLL_THREAD_ATTACH 및 DLL_THREAD_DETACH 알림을 사용하지 않도록 설정합니다.
DisableWriterClasses DisableWriterClasses 메서드는 특정 작성기 클래스가 이벤트를 수신하지 못하도록 합니다.
DisableWriterInstances DisableWriterInstances 메서드는 지정된 기록기 인스턴스 또는 인스턴스를 사용하지 않도록 설정합니다.
DiscardVirtualMemory 메모리를 커밋 해제하지 않고 메모리 페이지 범위의 메모리 내용을 삭제합니다. 삭제된 메모리의 내용은 정의되지 않았으며 애플리케이션에서 다시 작성해야 합니다.
DisconnectNamedPipe 클라이언트 프로세스에서 명명된 파이프 인스턴스의 서버 끝 연결을 끊습니다.
분리 탑재된 볼륨을 분리합니다.
DoSnapshotSet 이 집합의 모든 섀도 복사본을 동시에 커밋합니다.

EFFECTIVE_POWER_MODE_CALLBACK 유효 전원 모드 콜백을 위한 함수 클래스입니다.
꺼내기 현재 디바이스에서 미디어를 배출합니다.
EnableWriterClasses EnableWriterClasses 메서드를 사용하면 지정된 작성기가 모든 이벤트를 수신할 수 있습니다.
EnclaveGetAttestationReport 현재 Enclave를 설명하고 enclave 형식을 담당하는 기관에서 서명하는 Enclave 증명 보고서를 가져옵니다.
EnclaveGetEnclaveInformation 현재 실행 중인 enclave에 대한 정보를 가져옵니다.
EnclaveSealData 암호화되지 않은 데이터에서 암호화된 이진 큰 개체(Blob)를 생성합니다.
EnclaveUnsealData 암호화된 이진 큰 개체(Blob)의 암호를 해독합니다.
EnclaveVerifyAttestationReport 현재 시스템에서 생성된 증명 보고서를 확인합니다.
EndPrepareSnapshots 마지막 IVssHardwareSnapshotProvider::BeginPrepareSnapshot 호출 후 전체 섀도 복사본 집합에 대해 한 번 호출됩니다.
EnterCriticalSection 지정된 임계 영역 개체의 소유권을 기다립니다. 함수가 호출 스레드가 소유권을 부여받는 시기를 반환합니다.
EnterSynchronizationBarrier 최대 스레드 수가 장벽에 들어갈 때까지 호출 스레드가 동기화 장벽에서 대기하도록 합니다.

EnumPwrSchemes 모든 전원 구성표를 열거합니다.
EscapeCommFunction 지정된 통신 디바이스에 확장 함수를 수행하도록 지시합니다.
ExitWindows ExitWindowsEx 함수를 호출하여 대화형 사용자를 로그오프합니다.
ExitWindowsEx 대화형 사용자를 로그오프하거나, 시스템을 종료하거나, 시스템을 종료하고 다시 시작합니다.
확장 가상 디스크의 크기를 고정 또는 확장 가능한 디스크에서 사용할 수 있는 최대 크기로 늘입니다.
ExposeSnapshot ExposeSnapshot 메서드는 섀도 복사본을 드라이브 문자, 탑재된 폴더 또는 파일 공유로 노출합니다.
연장 IVdsLun::Extend 메서드(vds.h)는 지정된 바이트 수만큼 LUN을 확장합니다.
연장 각 플렉스의 각 멤버에 디스크 익스텐트 를 추가하여 현재 볼륨의 크기를 확장합니다.
연장 IVdsLun::Extend(vdshwprv.h) 메서드는 지정된 바이트 수만큼 LUN을 확장합니다.
FastRecovery 이 메서드는 나중에 사용하도록 예약됩니다. (IVssBackupComponentsEx2.FastRecovery)
FillInLunInfo 하드웨어 공급자가 해당 디스크 디바이스를 지원하는지 여부를 나타내고 VDS_LUN_INFORMATION 구조의 누락을 수정하라는 메시지를 표시합니다.
FlushCache IVdsController::FlushCache(vds.h) 메서드는 컨트롤러의 캐시를 영구 저장소로 플러시합니다.

FlushCache IVdsController::FlushCache 메서드(vdshwprv.h)는 컨트롤러의 캐시를 영구 저장소로 플러시합니다.
FlushViewOfFile 파일의 매핑된 보기 내에서 바이트 범위를 디스크에 씁니다.
Format 현재 볼륨의 파일 시스템 형식을 지정합니다.
FormatEx 파티션에서 파일 시스템 볼륨의 형식을 지정합니다.
FormatEx2 파티션에서 파일 시스템 볼륨의 형식을 지정합니다. 이 메서드는 형식 지정 옵션이 Options 매개 변수를 사용하여 지정된다는 점을 제외하고 IVdsVolumeMF2::FormatEx 메서드와 동일합니다.
FormatPartition 기존 OEM, ESP 또는 알 수 없는 파티션의 형식을 지정합니다. (IVdsAdvancedDisk.FormatPartition)
FormatPartitionEx 기존 OEM, ESP 또는 알 수 없는 파티션의 형식을 지정합니다. (IVdsDiskPartitionMF.FormatPartitionEx)
FormatPartitionEx2 기존 OEM, ESP 또는 알 수 없는 파티션의 형식을 지정합니다. 이 메서드는 형식 지정 옵션이 Options 매개 변수를 사용하여 지정된다는 점을 제외하고 IVdsDiskPartitionMF::FormatPartition 메서드와 동일합니다.
FreeComponentInfo FreeComponentInfo 메서드는 지정된 구성 요소 정보에 전념하는 시스템 리소스의 할당을 취소합니다.
FreeLibrary 로드된 DLL(동적 연결 라이브러리) 모듈을 해제하고 필요한 경우 참조 수를 줄입니다.
FreeLibraryAndExitThread 로드된 DLL(동적 연결 라이브러리)의 참조 수를 1씩 줄인 다음 ExitThread를 호출하여 호출 스레드를 종료합니다.

FreeUserPhysicalPages AllocateUserPhysicalPages 또는 AllocateUserPhysicalPagesNuma를 사용하여 이전에 할당된 실제 메모리 페이지를 해제합니다.
FreeWriterMetadata FreeWriterMetadata 메서드는 IVssBackupComponents::GatherWriterMetadata가 호출되었을 때 할당된 시스템 리소스를 해제합니다.
FreeWriterStatus FreeWriterStatus 메서드는 IVssBackupComponents::GatherWriterStatus를 호출하는 동안 할당된 시스템 리소스를 해제합니다.
GatherWriterMetadata GatherWriterMetadata 메서드는 각 작성기에게 수집한 메타데이터를 보내라는 메시지를 표시합니다. 메서드는 기록기와 통신하는 Identify 이벤트를 생성합니다.
GatherWriterStatus GatherWriterStatus 메서드는 각 작성기에게 상태 메시지를 보내라는 메시지를 표시합니다.
GetActivePwrScheme 활성 전원 구성표의 인덱스를 검색합니다.
GetAdditionalRestores GetAdditionalRestores 메서드는 증분 또는 차등 복원 작업 중에 기록기에서 지정된 구성 요소를 완전히 검색하기 위해 추가 복원 작업이 필요한지 여부를 결정하는 데 사용됩니다.
GetAlternateLocation GetAlternateLocation 메서드는 파일 집합에 대한 대체 위치를 가져옵니다.
GetAlternateLocationMapping GetAlternateLocationMapping은 파일 복원을 위해 파일 집합의 대체 위치를 반환하는 데 사용됩니다. 이 메서드는 작성기 또는 요청자가 호출할 수 있습니다.
GetAlternateLocationMapping GetAlternateLocationMapping 메서드는 파일 집합의 특정 대체 위치 매핑을 가져옵니다.

GetAlternateLocationMappingCount GetAlternateLocationMappingCount 메서드는 요청자가 데이터를 복원하는 데 사용하는 대체 위치 매핑 수를 반환합니다. 작성자 또는 요청자는 이 메서드를 호출할 수 있습니다.
GetAttributes IVdsStoragePool::GetAttributes 메서드(vds.h)는 스토리지 풀의 특성을 반환합니다.
GetAttributes IVdsStoragePool::GetAttributes(vdshwprv.h) 메서드는 스토리지 풀의 특성을 반환합니다.
GetAuthoritativeRestore 요청자가 복제된 데이터 저장소에 대해 구성 요소의 복원을 신뢰할 수 있는 것으로 표시했는지 여부를 확인합니다.
GetBackupMetadata GetBackupMetadata 메서드는 IVssComponent::SetBackupMetadata를 사용하여 CVssWriter::OnPrepareBackup에 의해 PrepareForBackup 이벤트 중에 설정되었을 수 있는 기록기별 프라이빗 백업 메타데이터를 검색합니다.
GetBackupOptions GetBackupOptions 메서드는 IVssBackupComponents::SetBackupOptions를 사용하여 요청자가 설정한 현재 선택된 구성 요소 또는 구성 요소를 관리하는 작성기에 지정된 백업 옵션을 반환합니다.
GetBackupSchema GetBackupSchema 메서드는 요청자가 기록기 메타데이터 문서에서 지정된 작성자가 참여할 수 있는 백업 작업 유형을 결정하는 데 사용됩니다.
GetBackupStamp GetBackupStamp 메서드는 지정된 구성 요소에 대해 기록기에서 저장한 백업 스탬프 문자열을 반환합니다.
GetBackupSucceeded GetBackupSucceeded 메서드는 선택한 구성 요소 또는 구성 요소 집합의 모든 파일을 VSS_FILE_RESTORE_STATUS 열거형으로 백업하려는 전체 시도의 상태를 반환합니다.
GetBackupType GetBackupType 메서드는 수행할 백업 유형을 나타냅니다.

GetBackupTypeMask GetBackupTypeMask 메서드는 현재 파일 설명자가 지정한 파일에 대한 파일 백업 사양을 VSS_FILE_SPEC_BACKUP_TYPE 값의 비트 마스크(또는 비트 OR)로 반환합니다.
GetCommConfig 통신 디바이스의 현재 구성을 검색합니다.
GetCommMask 지정된 통신 디바이스에 대한 이벤트 마스크 값을 검색합니다.
GetCommModemStatus 모뎀 컨트롤 레지스터 값을 검색합니다.
GetCommPorts 올바른 형식의 COM 포트를 포함하는 배열을 가져옵니다.
GetCommProperties 지정된 통신 디바이스의 통신 속성에 대한 정보를 검색합니다.
GetCommState 지정된 통신 디바이스에 대한 현재 컨트롤 설정을 검색합니다.
GetCommTimeouts 지정된 통신 디바이스에서 모든 읽기 및 쓰기 작업에 대한 제한 시간 매개 변수를 검색합니다.
GetComponent GetComponent 메서드는 백업 구성 요소 문서에 명시적으로 저장된 지정된 작성기의 구성 요소 중 하나에 IVssComponent 인터페이스를 반환합니다.
GetComponent GetComponent 메서드는 지정된 백업 구성 요소에 대한 기록기 메타데이터 문서를 가져옵니다.
GetComponentCount GetComponentCount 메서드는 백업 구성 요소 문서에 명시적으로 저장된 지정된 기록기의 구성 요소 수를 반환합니다.

GetComponentInfo GetComponentInfo 메서드는 지정된 기록기 메타데이터 구성 요소에 대한 기본 정보를 가져옵니다.
GetComponentName GetComponentName 메서드는 이 구성 요소의 논리적 이름을 반환합니다.
GetComponentName GetComponentName 메서드는 명시적 기록기-구성 요소 종속성에서 현재 구성 요소가 의존하는 구성 요소의 이름을 검색합니다.
GetComponentType GetComponentType 메서드는 VSS_COMPONENT_TYPE 열거형 측면에서 이 구성 요소의 형식을 반환합니다.
GetConnectedInitiators IVdsIscsiTarget::GetConnectedInitiators 메서드(vds.h)는 현재 대상에 로그인된 초기자의 iSCSI 이름 목록을 반환합니다.
GetConnectedInitiators IVdsIscsiTarget::GetConnectedInitiators(vdshwprv.h) 메서드는 현재 대상에 로그인된 초기자의 iSCSI 이름 목록을 반환합니다.
GetContext GetContext 정보는 진행 중이거나 가능한 모든 섀도 복사본에 대한 현재 컨텍스트를 반환합니다.
GetController IVdsControllerPort::GetController(vds.h) 메서드는 컨트롤러 포트가 속한 컨트롤러를 반환합니다.
GetController IVdsControllerPort::GetController 메서드(vdshwprv.h)는 컨트롤러 포트가 속한 컨트롤러를 반환합니다.
GetCurrentLevel GetCurrentLevel 메서드는 현재 애플리케이션 수준을 반환합니다.
GetCurrentPowerPolicies 현재 시스템 전원 정책 설정을 검색합니다.

GetCurrentSnapshotSetId GetCurrentSnapshotSetId 메서드는 섀도 복사본 집합의 고유 식별자를 반환합니다.
GetCurrentVolumeArray GetCurrentVolumeArray 메서드는 원래 볼륨의 이름과 섀도 복사본 집합에 속하는 원래 원격 파일 공유의 UNC 경로를 null로 종료된 와이드 문자열의 배열로 반환합니다. Windows 7, Windows Server 2008 R2, Windows Vista, Windows Server 2008, Windows XP 및 Windows Server 2003: 원격 파일 공유는 Windows 8 Windows Server 2012 때까지 지원되지 않습니다.
GetCurrentVolumeCount GetCurrentVolumeCount 메서드는 섀도 복사본 집합의 볼륨 수를 반환합니다.
GetDatabaseFile GetDatabaseFile 메서드는 지정된 데이터베이스 백업 구성 요소 파일에 대한 정보를 포함하는 IVssWMFiledesc 개체를 가져옵니다.
GetDatabaseLogFile GetDatabaseLogFile 메서드는 지정된 데이터베이스 백업 구성 요소와 연결된 로그 파일에 대한 파일 설명자를 가져옵니다.
GetDefaultCommConfigA 지정된 통신 디바이스에 대한 기본 구성을 검색합니다. (ANSI)
GetDefaultCommConfigW 지정된 통신 디바이스에 대한 기본 구성을 검색합니다. (유니코드)
GetDependency GetDependency 메서드는 현재 구성 요소 중 하나의 명시적 기록기 구성 요소 종속성에 대한 정보를 얻기 위한 접근자가 포함된 IVssWMDependency 인터페이스의 인스턴스를 반환합니다.
GetDeviceName 가상 디스크가 있는 볼륨의 디바이스 이름을 반환합니다.
GetDevicePowerState 지정된 디바이스의 현재 전원 상태를 검색합니다.

GetDifferencedFile GetDifferencedFile 메서드는 파일 집합(지정된 파일 또는 파일)에 대한 정보를 반환하여 증분 또는 차등 백업에 참여하거나, 파일과 연결된 백업 및 복원은 부분 파일을 사용하는 것이 아니라 전체 파일이 백업 미디어와 복사되는 것처럼 구현되어야 합니다.
GetDifferencedFilesCount 증분 백업 또는 복원을 지원하는 기록기에서 표시된 이 구성 요소의 파일 사양 수(그리고 정의한 구성 요소 집합의 하위 구성 요소)를 반환합니다.
GetDirectedTarget GetDirectedTarget 메서드는 백업 시 기록기에서 저장한 정보를 백업 구성 요소 문서에 반환하여 파일을 복원할 때 파일(원본 파일)을 다시 매핑해야 함을 나타냅니다.
GetDirectedTargetCount GetDirectedTargetCount 메서드는 현재 구성 요소와 연결된 지시된 대상 사양의 수를 반환합니다. 작성기 또는 요청자는 이 메서드를 호출할 수 있습니다.
GetDiskFromVDisk IVdsVDisk 인터페이스 포인터가 지정된 가상 디스크에 대한 IVdsDisk 인터페이스 포인터를 반환합니다.
GetDiskIdFromLunInfo 지정된 LUN에 해당하는 디스크의 VDS 개체 ID를 검색합니다.
GetDllDirectoryA 애플리케이션에 대한 DLL을 찾는 데 사용되는 검색 경로의 애플리케이션별 부분을 검색합니다. (ANSI)
GetDllDirectoryW 애플리케이션에 대한 DLL을 찾는 데 사용되는 검색 경로의 애플리케이션별 부분을 검색합니다. (유니코드)
GetDocument 지원 안 됨 (IVssCreateWriterMetadata.GetDocument)
GetDocument 지원되지 않습니다. (IVssExamineWriterMetadata.GetDocument)

GetDrive IVdsSubSystem::GetDrive 메서드(vds.h)는 지정된 드라이브를 반환합니다.
GetDrive IVdsSubSystem::GetDrive(vdshwprv.h) 메서드는 지정된 드라이브를 반환합니다.
GetDrive2 IVdsSubSystem2::GetDrive2 메서드(vds.h)는 지정된 드라이브를 반환하고 엔클로저 번호를 매개 변수로 포함합니다.
GetDrive2 IVdsSubSystem2::GetDrive2(vdshwprv.h) 메서드는 지정된 드라이브를 반환합니다.
GetDriveLetter OEM, ESP 또는 알 수 없는 파티션에 할당된 드라이브 문자를 반환합니다.
GetEventLogInformation 지정된 이벤트 로그에 대한 정보를 검색합니다.
GetExcludeFile GetExcludeFile 메서드는 지정된 기록기에 대한 백업에서 명시적으로 제외된 파일에 대한 정보를 가져옵니다.
GetExcludeFromSnapshotCount 지정된 섀도 복사본에서 명시적으로 제외된 파일 집합의 수를 가져옵니다.
GetExcludeFromSnapshotFile 지정된 섀도 복사본에서 명시적으로 제외된 파일 집합에 대한 정보를 가져옵니다.
GetFailure VSS 요청자는 이 메서드를 호출하여 기록기에서 보고한 구성 요소 수준 오류를 검색합니다.
GetFile GetFile 메서드는 파일 그룹과 연결된 파일 설명자를 가져옵니다.
GetFileCounts GetFileCounts 메서드는 제외된 파일과 작성기가 관리하는 구성 요소 수를 가져옵니다.

GetFileRestoreStatus GetFileRestoreStatus 메서드는 선택한 구성 요소 또는 구성 요소 집합의 모든 파일을 VSS_FILE_RESTORE_STATUS 열거형으로 복원하려는 완료된 시도의 상태를 반환합니다.
GetFilespec GetFilespec 메서드는 현재 IVssWMFiledesc 개체가 멤버인 파일 목록을 가져오는 데 사용되는 파일 사양을 반환합니다.
GetFileSystemProperties 현재 볼륨의 파일 시스템에 대한 속성 세부 정보를 반환합니다.
GetFileSystemTypeName 볼륨에서 파일 시스템의 이름을 검색합니다.
GetHostVolume 가상 디스크가 있는 볼륨의 볼륨 개체에 대한 인터페이스 포인터를 반환합니다.
GetIdentificationData 디스크를 고유하게 식별하는 정보를 검색합니다.
GetIdentificationData IVdsLun::GetIdentificationData 메서드(vds.h)는 SCSI 조회 데이터 및 0x80 및 0x83 중요 제품 데이터 페이지의 데이터를 반환합니다.
GetIdentificationData IVdsLun::GetIdentificationData(vdshwprv.h) 메서드는 SCSI 조회 데이터 및 0x80 및 0x83 중요 제품 데이터 페이지의 데이터를 반환합니다.
GetIdentifyInformation 기록기 OnIdentify 또는 OnIdentifyEx 메서드가 이전에 보고한 메타데이터를 가져옵니다.
GetIdentity GetIdentity 메서드는 특정 기록기 인스턴스에 대한 기본 정보를 가져옵니다.
GetIdentityEx GetIdentityEx 메서드는 작성기 인스턴스 이름 및 특정 기록기 인스턴스에 대한 기타 기본 정보를 가져옵니다.

GetImportTarget 이 하위 시스템에 대한 컴퓨터의 VSS(볼륨 섀도 복사본 서비스) 가져오기 대상을 반환합니다.
GetIncludeFile 지원 안 됨 (IVssExamineWriterMetadata.GetIncludeFile)
GetInitiatorAdapter 초기자 포털이 속한 초기자 어댑터를 반환합니다.
GetInitiatorName 로컬 초기자 서비스의 iSCSI 이름을 반환합니다.
GetIpsecSecurity 지원되지 않습니다. 이 메서드는 나중에 사용하도록 예약되어 있습니다. (IVdsIscsiInitiatorPortal.GetIpsecSecurity)
GetIpsecSecurity IVdsIscsiPortal::GetIpsecSecurity 메서드(vds.h)는 지원되지 않으며 나중에 사용하도록 예약되어 있습니다.
GetIpsecSecurity IVdsIscsiPortal::GetIpsecSecurity(vdshwprv.h) 메서드는 지원되지 않지만 나중에 사용하도록 예약되어 있습니다.
GetLargePageMinimum 큰 페이지의 최소 크기를 검색합니다.
GetLoadBalancePolicy IVdsLunMpio::GetLoadBalancePolicy 메서드(vds.h)는 LUN에서 현재 부하 분산 정책을 반환합니다.
GetLoadBalancePolicy IVdsLunMpio::GetLoadBalancePolicy(vdshwprv.h) 메서드는 LUN에서 현재 부하 분산 정책을 반환합니다.
GetLogicalPath GetLogicalPath 메서드는 이 구성 요소의 논리적 경로를 반환합니다.

GetLogicalPath GetLogicalPath 메서드는 명시적 기록기-구성 요소 종속성에서 현재 구성 요소가 의존하는 구성 요소의 논리적 경로를 검색합니다.
GetLun IVdsLunPlex::GetLun 메서드(vds.h)는 플렉스가 속한 LUN을 반환합니다.
GetLun IVdsLunPlex::GetLun(vdshwprv.h) 메서드는 플렉스가 속한 LUN을 반환합니다.
GetLunNumber IVdsLunNumber::GetLunNumber 메서드(vds.h)는 LUN에 대한 LUN 번호를 검색합니다.
GetLunNumber IVdsLunNumber::GetLunNumber(vdshwprv.h) 메서드는 LUN에 대한 LUN 번호를 검색합니다.
GetMailslotInfo 지정된 mailslot에 대한 정보를 검색합니다.
GetMemoryErrorHandlingCapabilities 시스템의 메모리 오류 처리 기능을 가져옵니다.
GetMinDiffAreaSize 섀도 복사본 스토리지 영역의 현재 최소 크기를 반환합니다.
GetModuleFileNameA 지정된 모듈을 포함하는 파일의 정규화된 경로를 검색합니다. 모듈은 현재 프로세스에 의해 로드되어 있어야 합니다. (ANSI)
GetModuleFileNameW 지정된 모듈을 포함하는 파일의 정규화된 경로를 검색합니다. 모듈은 현재 프로세스에 의해 로드되어 있어야 합니다. (유니코드)
GetModuleHandleA 지정된 모듈에 대한 모듈 핸들을 검색합니다. 호출 프로세스에서 모듈을 로드해야 합니다. (ANSI)

GetModuleHandleExA 지정된 모듈에 대한 모듈 핸들을 검색하고 GET_MODULE_HANDLE_EX_FLAG_UNCHANGED_REFCOUNT 지정하지 않는 한 모듈의 참조 횟수를 증분합니다. 호출 프로세스에서 모듈을 로드해야 합니다. (ANSI)
GetModuleHandleExW 지정된 모듈에 대한 모듈 핸들을 검색하고 GET_MODULE_HANDLE_EX_FLAG_UNCHANGED_REFCOUNT 지정하지 않는 한 모듈의 참조 횟수를 증분합니다. 호출 프로세스에서 모듈을 로드해야 합니다. (유니코드)
GetModuleHandleW 지정된 모듈에 대한 모듈 핸들을 검색합니다. 호출 프로세스에서 모듈을 로드해야 합니다. (유니코드)
GetNamedPipeClientComputerNameA GetNamedPipeClientComputerNameA(ANSI) 함수(winbase.h)는 지정된 명명된 파이프에 대한 클라이언트 컴퓨터 이름을 검색합니다.
GetNamedPipeClientComputerNameW GetNamedPipeClientComputerNameW(유니코드) 함수(winbase.h)는 지정된 명명된 파이프에 대한 클라이언트 컴퓨터 이름을 검색합니다.
GetNamedPipeClientProcessId 지정된 명명된 파이프에 대한 클라이언트 프로세스 식별자를 검색합니다.
GetNamedPipeClientSessionId 지정된 명명된 파이프에 대한 클라이언트 세션 식별자를 검색합니다.
GetNamedPipeHandleStateA GETNamedPipeHandleStateA(ANSI) 함수(winbase.h)는 지정된 명명된 파이프에 대한 정보를 검색합니다.
GetNamedPipeHandleStateW GetNamedPipeHandleStateW(유니코드) 함수(winbase.h)는 지정된 명명된 파이프에 대한 정보를 검색합니다.
GetNamedPipeInfo 지정된 명명된 파이프에 대한 정보를 검색합니다.

GetNamedPipeServerProcessId 지정된 명명된 파이프에 대한 서버 프로세스 식별자를 검색합니다.
GetNamedPipeServerSessionId 지정된 명명된 파이프에 대한 서버 세션 식별자를 검색합니다.
GetNewTarget GetNewTarget 메서드는 선택한 구성 요소 또는 구성 요소 집합에 대한 새 파일 복원 위치를 반환합니다.
GetNewTargetCount GetNewTargetCount 메서드는 지정된 구성 요소와 연결된 새 대상 복원 위치의 수를 반환합니다.
GetNumberOfEventLogRecords 지정된 이벤트 로그의 레코드 수를 검색합니다.
GetObject 식별된 개체에 대한 개체 포인터를 반환합니다.
GetObject 지정된 개체를 반환합니다.
GetOldestEventLogRecord 지정된 이벤트 로그에서 가장 오래된 레코드의 절대 레코드 번호를 검색합니다.
GetPack 현재 디스크가 멤버인 디스크 팩을 반환합니다.
GetPack 볼륨이 멤버인 팩을 검색합니다.
GetPartialFile GetPartialFile 메서드는 이 구성 요소와 연결된 부분 파일에 대한 정보를 반환합니다.
GetPartialFileCount GetPartialFileCount 메서드는 구성 요소와 연결된 부분 파일의 수를 반환합니다.

GetPartitionFileSystemProperties 지정된 바이트 오프셋에서 디스크의 파티션에 있는 파일 시스템에 대한 속성 세부 정보를 반환합니다.
GetPartitionFileSystemTypeName 지정된 바이트 오프셋에서 디스크의 파티션에 있는 파일 시스템의 이름을 검색합니다.
GetPartitionProperties 파티션 오프셋으로 식별된 파티션의 속성을 반환합니다.
GetPath GetPath 메서드는 원격 파일 공유의 정규화된 디렉터리 경로 또는 UNC 경로를 가져와 현재 IVssWMFiledesc 개체에 설명된 파일 목록을 가져옵니다.
GetPathInfo IVdsLunMpio::GetPathInfo 메서드(vds.h)는 LUN의 각 경로에 대해 하나씩 VDS_PATH_INFO 구조의 배열을 반환합니다.
GetPathInfo IVdsLunMpio::GetPathInfo(vdshwprv.h) 메서드는 LUN의 각 경로에 대해 하나씩 VDS_PATH_INFO 구조의 배열을 반환합니다.
GetPhysicallyInstalledSystemMemory 컴퓨터에 물리적으로 설치된 RAM의 양을 검색합니다.
GetPortProperties IVdsController::GetPortProperties(vds.h) 메서드는 지정된 컨트롤러 포트의 속성을 반환합니다.
GetPortProperties IVdsController::GetPortProperties 메서드(vdshwprv.h)는 지정된 컨트롤러 포트의 속성을 반환합니다.
GetPostRestoreFailureMsg GetPostRestoreFailureMsg 메서드는 IVssComponent::SetPostRestoreFailureMsg가 1을 설정한 경우 PostRestore 이벤트를 처리하는 동안 기록기에서 생성된 실패 메시지를 반환합니다.
GetPostSnapshotFailureMsg 기록기가 지정된 구성 요소에 대해 설정한 PostSnapshot 실패 메시지 문자열을 반환합니다.

GetPrepareForBackupFailureMsg 작성기가 지정된 구성 요소에 대해 설정한 PrepareForBackup 실패 메시지 문자열을 반환합니다.
GetPreRestoreFailureMsg GetPreRestoreFailureMsg 메서드는 IVssComponent::SetPreRestoreFailureMsg가 1을 설정한 경우 PreRestore 이벤트를 처리하는 동안 기록기에서 생성된 오류 메시지를 검색합니다.
GetPreviousBackupStamp GetPreviousBackupStamp 메서드는 백업 구성 요소 문서의 요청자가 로드한 이전 백업 스탬프를 반환합니다. 이 값은 파일이 차등 백업 또는 증분 백업 작업에 참여해야 하는지 여부를 결정할 때 기록기에서 사용됩니다.
GetProcAddress 지정된 DLL(동적 연결 라이브러리)에서 내보낸 함수 또는 변수의 주소를 검색합니다.
GetProcessDEPPolicy 지정된 32비트 프로세스에 대한 DEP(데이터 실행 방지) 및 DEP-ATL unk 에뮬레이션 설정을 가져옵니다. WINDOWS XP SP3: 현재 프로세스에 대한 DEP 및 DEP-ATL unk 에뮬레이션 설정을 가져옵니다.
GetProcessHeap 호출 프로세스의 기본 힙에 대한 핸들을 검색합니다.
GetProcessHeaps 활성 힙 수를 반환하고 호출 프로세스에 대한 모든 활성 힙에 대한 핸들을 검색합니다.
GetProperties IVdsController::GetProperties(vds.h) 메서드는 컨트롤러의 속성을 반환합니다.
GetProperties IVdsControllerPort::GetProperties(vds.h) 메서드는 컨트롤러 포트의 속성을 검색합니다.
GetProperties 디스크에 대한 속성 정보를 반환합니다.
GetProperties IVdsDrive::GetProperties 메서드(vds.h)는 드라이브 개체의 속성을 반환합니다.

GetProperties HBA 포트의 속성을 검색합니다.
GetProperties 초기자 어댑터의 속성을 반환합니다.
GetProperties 초기자 포털의 속성을 반환합니다.
GetProperties IVdsIscsiPortal::GetProperties 메서드(vds.h)는 포털의 속성을 반환합니다.
GetProperties IVdsIscsiPortalGroup::GetProperties 메서드(vds.h)는 포털 그룹의 속성을 반환합니다.
GetProperties IVdsIscsiTarget::GetProperties 메서드(vds.h)는 iSCSI 대상의 속성을 반환합니다.
GetProperties IVdsLun::GetProperties 메서드(vds.h)는 LUN 개체의 속성을 반환합니다.
GetProperties IVdsLunPlex::GetProperties 메서드(vds.h)는 LUN 플렉스의 속성을 반환합니다.
GetProperties pack 개체의 속성 세부 정보를 반환합니다.
GetProperties IVdsProvider::GetProperties 메서드(vds.h)는 공급자의 속성을 반환합니다.
GetProperties VDS의 속성을 반환합니다.
GetProperties IVdsStoragePool::GetProperties 메서드(vds.h)는 스토리지 풀의 속성을 반환합니다.

GetProperties IVdsSubSystem::GetProperties 메서드(vds.h)는 하위 시스템의 속성을 반환합니다.
GetProperties 가상 디스크가 있는 볼륨에 대한 디스크 속성 정보를 반환합니다.
GetProperties 현재 볼륨의 속성 세부 정보를 반환합니다.
GetProperties 현재 볼륨 플렉스의 속성 세부 정보를 반환합니다.
GetProperties IVdsController::GetProperties 메서드(vdshwprv.h)는 컨트롤러의 속성을 반환합니다.
GetProperties IVdsControllerPort::GetProperties 메서드(vdshwprv.h)는 컨트롤러 포트의 속성을 검색합니다.
GetProperties IVdsDrive::GetProperties(vdshwprv.h) 메서드는 드라이브 개체의 속성을 반환합니다.
GetProperties IVdsIscsiPortal::GetProperties(vdshwprv.h) 메서드는 포털의 속성을 반환합니다.
GetProperties IVdsIscsiPortalGroup::GetProperties(vdshwprv.h) 메서드는 포털 그룹의 속성을 반환합니다.
GetProperties IVdsIscsiTarget::GetProperties(vdshwprv.h) 메서드는 iSCSI 대상의 속성을 반환합니다.
GetProperties IVdsLun::GetProperties(vdshwprv.h) 메서드는 LUN 개체의 속성을 반환합니다.
GetProperties IVdsLunPlex::GetProperties(vdshwprv.h) 메서드는 LUN 플렉스의 속성을 반환합니다.

GetProperties IVdsProvider::GetProperties(vdshwprv.h) 메서드는 공급자의 속성을 반환합니다.
GetProperties IVdsStoragePool::GetProperties(vdshwprv.h) 메서드는 스토리지 풀의 속성을 반환합니다.
GetProperties IVdsSubSystem::GetProperties(vdshwprv.h) 메서드는 하위 시스템의 속성을 반환합니다.
GetProperties2 디스크에 대한 속성 정보를 반환합니다. 이 메서드는 VDS_DISK_PROP 구조체 대신 VDS_DISK_PROP2 구조를 반환한다는 점을 제외하고 IVdsDisk::GetProperties 메서드와 동일합니다.
GetProperties2 IVdsDrive2::GetProperties2 메서드(vds.h)는 드라이브 개체의 속성을 반환합니다.
GetProperties2 IVdsSubSystem2::GetProperties2 메서드(vds.h)는 하위 시스템의 속성을 반환합니다.
GetProperties2 현재 볼륨에 대한 속성 정보를 반환합니다. 이 메서드는 VDS_VOLUME_PROP 구조체 대신 VDS_VOLUME_PROP2 구조를 반환한다는 점을 제외하고 IVdsVolume::GetProperties 메서드와 동일합니다.
GetProperties2 IVdsDrive2::GetProperties2(vdshwprv.h) 메서드는 드라이브 개체의 속성을 반환합니다.
GetProperties2 VdsSubSystem2::GetProperties2(vdshwprv.h) 메서드는 하위 시스템의 속성을 반환합니다.
GetProvider 팩과 연결된 소프트웨어 공급자를 반환합니다.
GetProvider IVdsStoragePool::GetProvider 메서드(vds.h)는 스토리지 풀을 관리하는 하드웨어 공급자를 반환합니다.

GetProvider IVdsSubSystem::GetProvider 메서드(vds.h)는 하위 시스템을 관리하는 공급자를 반환합니다.
GetProvider IVdsStoragePool::GetProvider(vdshwprv.h) 메서드는 스토리지 풀을 관리하는 하드웨어 공급자를 반환합니다.
GetProvider IVdsSubSystem::GetProvider(vdshwprv.h) 메서드는 하위 시스템을 관리하는 공급자를 반환합니다.
GetProviderCapabilities 이 메서드는 나중에 사용하도록 예약됩니다. (IVssHardwareSnapshotProviderEx.GetProviderCapabilities)
GetProviderMgmtInterface 인터페이스를 반환하여 시스템 공급자를 추가로 구성합니다.
GetProviderType IVdsHwProviderType::GetProviderType 메서드(vds.h)는 하드웨어 공급자의 형식을 검색합니다.
GetProviderType IVdsHwProviderType::GetProviderType(vdshwprv.h) 메서드는 하드웨어 공급자의 형식을 검색합니다.
GetProviderType2 IVdsHwProviderType2::GetProviderType2 메서드(vds.h)는 하드웨어 공급자의 형식을 검색합니다.
GetProviderType2 IVdsHwProviderType2::GetProviderType2(vdshwprv.h) 메서드는 하드웨어 공급자의 형식을 검색합니다.
GetPwrCapabilities 시스템 전원 기능에 대한 정보를 검색합니다.
GetPwrDiskSpindownRange 디스크 스핀다운 범위를 검색합니다.

GetRecursive IVssWMFiledesc::GetPath에서 반환된 루트 디렉터리가 있는 IVssWMFiledesc 개체에 설명된 파일 목록에 해당 디렉터리의 파일만 포함되어 있는지 여부를 나타냅니다.
GetRestoreMetadata GetRestoreMetadata 메서드는 IVssComponent::SetRestoreMetadata를 사용하여 CVssWriter::OnPreRestore에 의해 PreRestore 이벤트 중에 설정되었을 수 있는 개인 기록기별 복원 메타데이터를 검색합니다.
GetRestoreMethod GetRestoreMethod 메서드는 기록기에서 데이터를 복원하려는 방법에 대한 정보를 반환합니다.
GetRestoreName 복원 중인 구성 요소에 할당된 논리적 이름을 가져옵니다.
GetRestoreOptions GetRestoreOptions 메서드는 IVssBackupComponents::SetRestoreOptions를 사용하여 요청자가 현재 작성기에 지정한 복원 옵션을 가져옵니다.
GetRestoreSubcomponent GetRestoreSubcomponent 메서드는 지정된 구성 요소와 연결된 지정된 하위 구성 요소를 반환합니다.
GetRestoreSubcomponentCount GetRestoreSubcomponentCount 메서드는 구성 요소와 연결된 하위 구성 요소의 수를 반환합니다.
GetRestoreTarget GetRestoreTarget 메서드는 현재 구성 요소에 대한 복원 대상(VSS_RESTORE_TARGET 열거형 측면에서)을 반환합니다.
GetRestoreType GetRestoreType 메서드는 작성기가 참여하는 복원 유형을 반환합니다.
GetRollForward 구성 요소에 대한 롤포워드 작업 유형을 가져오고 부분 롤포워드 작업의 복원 지점을 가져옵니다.

GetRootAndLogicalPrefixPaths IVsBackupComponents::AddToSnapshotSet 메서드에 전달할 수 있도록 로컬 볼륨 경로 또는 UNC 공유 경로를 정규화합니다.
GetSANPolicy 운영 체제에 대한 디스크 SAN 정책을 가져옵니다.
GetSessionId 기록기 세션 식별자를 반환합니다.
GetSessionId 요청자의 세션 식별자를 반환합니다.
GetSnapshotDeviceName GetSnapshotDeviceName 메서드는 지정된 볼륨 또는 파일 공유의 섀도 복사본을 호스트하는 디바이스의 이름을 반환합니다.
GetSnapshotProperties GetSnapshotProperties 메서드는 지정된 섀도 복사본의 속성을 가져옵니다.
GetSnapshotProperties 파일 공유 스냅샷에 대한 VSS_SNAPSHOT_PROP 구조를 가져옵니다.
GetSnapshotProperties 지정된 섀도 복사본의 속성을 가져옵니다.
GetSubSystem IVdsController::GetSubSystem(vds.h) 메서드는 컨트롤러가 속한 하위 시스템을 반환합니다.
GetSubSystem IVdsDrive::GetSubSystem 메서드(vds.h)는 드라이브가 속한 하위 시스템을 반환합니다.
GetSubSystem IVdsIscsiPortal::GetSubSystem 메서드(vds.h)는 포털이 속한 하위 시스템을 반환합니다.
GetSubSystem IVdsIscsiTarget::GetSubSystem 메서드(vds.h)는 대상이 속한 하위 시스템을 반환합니다.

GetSubSystem IVdsLun::GetSubSystem 메서드(vds.h)는 LUN을 표시하는 하위 시스템을 반환합니다.
GetSubSystem IVdsController::GetSubSystem 메서드(vdshwprv.h)는 컨트롤러가 속한 하위 시스템을 반환합니다.
GetSubSystem IVdsDrive::GetSubSystem(vdshwprv.h) 메서드는 드라이브가 속한 하위 시스템을 반환합니다.
GetSubSystem IVdsIscsiPortal::GetSubSystem(vdshwprv.h) 메서드는 포털이 속한 하위 시스템을 반환합니다.
GetSubSystem IVdsIscsiTarget::GetSubSystem(vdshwprv.h) 메서드는 대상이 속한 하위 시스템을 반환합니다.
GetSubSystem IVdsLun::GetSubSystem(vdshwprv.h) 메서드는 LUN을 표시하는 하위 시스템을 반환합니다.
GetSupportedInterconnects IVdsSubSystemInterconnect::GetSupportedInterconnects 메서드(vds.h)는 하위 시스템이 지원하는 상호 연결 형식을 반환합니다.
GetSupportedInterconnects IVdsSubSystemInterconnect::GetSupportedInterconnects(vdshwprv.h) 메서드는 하위 시스템이 지원하는 상호 연결 형식을 반환합니다.
GetSupportedLbPolicies IVdsLunMpio::GetSupportedLbPolicies 메서드(vds.h)는 하드웨어 공급자가 지원하는 부하 분산 정책을 검색합니다.
GetSupportedLbPolicies IVdsLunMpio::GetSupportedLbPolicies(vdshwprv.h) 메서드는 하드웨어 공급자가 지원하는 부하 분산 정책을 검색합니다.
GetSystemDEPPolicy 시스템에 대한 DEP(데이터 실행 방지) 정책 설정을 가져옵니다.

GetSystemFileCacheSize 시스템 캐시의 작업 집합에 대한 현재 크기 제한을 검색합니다.
GetSystemPowerStatus 시스템의 전원 상태를 검색합니다. 상태는 시스템이 AC 또는 DC 전원에서 실행 중인지 여부, 배터리가 현재 충전 중인지 여부, 배터리 수명이 얼마나 남아 있는지, 배터리 절약 모드가 켜져 있는지 또는 꺼져 있는지 여부를 나타냅니다.
GetTarget IVdsIscsiPortalGroup::GetTarget 메서드(vds.h)는 포털 그룹이 속한 대상을 반환합니다.
GetTarget IVdsIscsiPortalGroup::GetTarget(vdshwprv.h) 메서드는 포털 그룹이 속한 대상을 반환합니다.
GetTargetLuns 새로 만든 섀도 복사본 LUN에 대한 VDS_LUN_INFORMATION 구조를 초기화하라는 메시지를 하드웨어 공급자에게 표시합니다.
GetVDiskFromDisk IVdsDisk 인터페이스 포인터가 지정된 가상 디스크에 대한 IVdsVDisk 인터페이스 포인터를 반환합니다.
GetVersion 기록기 애플리케이션에 대한 버전 정보를 가져옵니다.
GetVersionSupport IVdsProviderSupport::GetVersionSupport 메서드(vds.h)는 지원되는 VDS 인터페이스를 사용하여 VDS_VERSION_SUPPORT_FLAG 열거된 값의 비트 마스크를 반환합니다.
GetVersionSupport IVdsProviderSupport::GetVersionSupport(vdshwprv.h) 메서드는 VDS 인터페이스의 버전을 나타내는 VDS_VERSION_SUPPORT_FLAG 열거된 값의 비트 마스크를 반환합니다.
GetVolume 현재 플렉스가 멤버인 볼륨을 반환합니다.
GetVolumeProtectLevel 지정된 볼륨의 섀도 복사본 보호 수준 및 상태를 가져옵니다.

GetWriterComponents GetWriterComponents 메서드는 요청자의 백업 구성 요소 문서에 저장된 지정된 작성기의 해당 구성 요소에 대한 정보를 반환하는 데 사용됩니다.
GetWriterComponentsCount GetWriterComponentsCount 메서드는 구성 요소가 요청자의 백업 구성 요소 문서에 추가된 기록기 수를 반환합니다.
GetWriterId GetWriterId 메서드는 명시적 기록기-구성 요소 종속성에서 현재 구성 요소가 의존하는 구성 요소를 포함하는 기록기의 클래스 ID를 검색합니다.
GetWriterInfo GetWriterInfo 메서드는 구성 요소를 담당하는 작성기의 인스턴스 및 클래스 식별자를 가져옵니다.
GetWriterMetadata GetWriterMetadata 메서드는 시스템에서 실행되는 특정 기록기에 대한 메타데이터를 반환합니다.
GetWriterMetadataCount GetWriterMetadataCount 메서드는 메타데이터가 있는 기록기 수를 반환합니다.
GetWriterMetadataEx GetWriterMetadataEx 메서드는 시스템에서 실행되는 특정 기록기 인스턴스에 대한 메타데이터를 반환합니다.
GetWriterStatus GetWriterStatus 메서드는 지정된 작성기의 상태를 반환합니다.
GetWriterStatusCount GetWriterStatusCount 메서드는 상태의 기록기 수를 반환합니다.
GetWriterStatusEx 지정된 작성기에 대한 확장 상태 정보를 반환합니다.
GetWriteWatch 가상 메모리 영역에 기록된 페이지의 주소를 검색합니다.

GlobalAlloc 힙에서 지정된 바이트 수를 할당합니다. (GlobalAlloc)
GlobalDiscard 지정된 전역 메모리 블록을 삭제합니다.
GlobalFlags 지정된 전역 메모리 개체에 대한 정보를 검색합니다.
GlobalFree 지정된 전역 메모리 개체를 해제하고 해당 핸들을 무효화합니다.
GlobalHandle 전역 메모리 블록에 대해 지정된 포인터와 연결된 핸들을 검색합니다.
GlobalLock 전역 메모리 개체를 잠그고 개체 메모리 블록의 첫 번째 바이트에 대한 포인터를 반환합니다.
GlobalMemoryStatus 물리적 메모리와 가상 메모리 모두에 대한 시스템의 현재 사용량에 대한 정보를 검색합니다. (GlobalMemoryStatus)
GlobalMemoryStatusEx 물리적 메모리와 가상 메모리 모두에 대한 시스템의 현재 사용량에 대한 정보를 검색합니다. (GlobalMemoryStatusEx)
GlobalReAlloc 지정된 전역 메모리 개체의 크기 또는 특성을 변경합니다. 크기가 증가하거나 감소할 수 있습니다.
GlobalSize 지정된 전역 메모리 개체의 현재 크기(바이트)를 검색합니다.
GlobalUnlock GMEM_MOVEABLE 할당된 메모리 개체와 연결된 잠금 수를 감소합니다.
HeapAlloc 힙에서 메모리 블록을 할당합니다. 할당된 메모리는 움직일 수 없습니다.

HeapCompact 지정된 힙에서 커밋된 가장 큰 자유 블록의 크기를 반환합니다. 사용 가능한 전역 플래그에서 힙 병합 사용 안 함이 설정된 경우 이 함수는 힙의 인접한 사용 가능한 메모리 블록도 병합합니다.
HeapCreate 호출 프로세스에서 사용할 수 있는 프라이빗 힙 개체를 만듭니다. 함수는 프로세스의 가상 주소 공간에 공간을 예약하고 이 블록의 지정된 초기 부분에 대한 물리적 스토리지를 할당합니다.
HeapDestroy 지정된 힙 개체를 제거합니다. 프라이빗 힙 개체의 모든 페이지를 커밋 해제하고 해제하고 힙에 대한 핸들을 무효화합니다.
HeapFree HeapAlloc 또는 HeapReAlloc 함수에 의해 힙에서 할당된 메모리 블록을 해제합니다.
HeapLock 지정된 힙과 연결된 중요한 섹션 개체 또는 잠금을 획득하려고 시도합니다.
HeapQueryInformation 지정된 힙에 대한 정보를 검색합니다.
HeapReAlloc 힙에서 메모리 블록을 다시 할당합니다. 이 함수를 사용하면 메모리 블록의 크기를 조정하고 다른 메모리 블록 속성을 변경할 수 있습니다.
HeapSetInformation 지정된 힙에 대한 기능을 사용하도록 설정합니다.
HeapSize HeapAlloc 또는 HeapReAlloc 함수에 의해 힙에서 할당된 메모리 블록의 크기를 검색합니다.
HeapUnlock 지정된 힙과 연결된 중요한 섹션 개체 또는 잠금의 소유권을 해제합니다.
HeapValidate 지정된 힙의 유효성을 검사합니다. 함수는 힙의 모든 메모리 블록을 검사하고 힙 관리자가 유지 관리하는 힙 제어 구조가 일관된 상태임을 확인합니다.

HeapWalk 지정된 힙의 메모리 블록을 열거합니다.
ImportSnapshots 다른 컴퓨터에서 전송된 섀도 복사본을 가져옵니다.
초기화 CVssWriter 개체를 초기화하고 기록기 애플리케이션이 VSS와 상호 작용할 수 있도록 합니다.
InitializeConditionVariable 조건 변수를 초기화합니다.
InitializeCriticalSection 중요한 섹션 개체를 초기화합니다.
InitializeCriticalSectionAndSpinCount 중요한 섹션 개체를 초기화하고 중요한 섹션의 스핀 수를 설정합니다.
InitializeCriticalSectionEx 스핀 수 및 선택적 플래그를 사용하여 중요한 섹션 개체를 초기화합니다.
InitializeEnclave 데이터를 사용하여 만들고 로드한 enclave를 초기화합니다.
InitializeEx CVssWriterEx 개체를 초기화하고 기록기 애플리케이션이 VSS와 상호 작용할 수 있도록 합니다. Initialize 메서드와 달리 InitializeEx 메서드를 사용하면 호출자가 기록기 버전 정보를 지정할 수 있습니다.
InitializeForBackup InitializeForBackup 메서드는 백업을 준비하기 위해 백업 구성 요소 메타데이터를 초기화합니다.
InitializeForRestore InitializeForRestore 메서드는 복원 작업을 준비하기 위해 IVssBackupComponents 인터페이스를 초기화합니다.

InitializeSRWLock SRW(슬림 판독기/기록기) 잠금을 초기화합니다.
InitializeSynchronizationBarrier 새 동기화 장벽을 초기화합니다.
InitiateShutdownA 지정된 컴퓨터의 종료 및 다시 시작을 시작하고 다시 시작에 등록된 모든 애플리케이션을 다시 시작합니다. (ANSI)
InitiateShutdownW 지정된 컴퓨터의 종료 및 다시 시작을 시작하고 다시 시작에 등록된 모든 애플리케이션을 다시 시작합니다. (유니코드)
InitiateSystemShutdownA 지정된 컴퓨터의 종료 및 선택적 다시 시작을 시작합니다. (ANSI)
InitiateSystemShutdownExA 지정된 컴퓨터의 종료 및 선택적 다시 시작을 시작하고 필요에 따라 종료 이유를 기록합니다. (ANSI)
InitiateSystemShutdownExW 지정된 컴퓨터의 종료 및 선택적 다시 시작을 시작하고 필요에 따라 종료 이유를 기록합니다. (유니코드)
InitiateSystemShutdownW 지정된 컴퓨터의 종료 및 선택적 다시 시작을 시작합니다. (유니코드)
InitOnceBeginInitialize 일회성 초기화를 시작합니다.
InitOnceComplete InitOnceBeginInitialize 함수로 시작된 일회성 초기화를 완료합니다.
InitOnceExecuteOnce 지정된 함수를 한 번 성공적으로 실행합니다. 동일한 일회성 초기화 구조를 지정하는 다른 스레드는 현재 스레드에서 실행되는 동안 지정된 함수를 실행할 수 없습니다.

InitOnceInitialize 일회성 초기화 구조를 초기화합니다.
InstallAlternateWriter 지원되지 않습니다. (CVssWriter.InstallAlternateWriter)
InvalidateCache IVdsController::InvalidateCache(vds.h) 메서드는 컨트롤러의 캐시를 무효화합니다. 캐시의 모든 데이터가 손실됩니다.
InvalidateCache IVdsController::InvalidateCache 메서드(vdshwprv.h)는 컨트롤러의 캐시를 무효화합니다.
IsBadCodePtr 호출 프로세스에 지정된 주소의 메모리에 대한 읽기 권한이 있는지 여부를 확인합니다.
IsBadReadPtr 호출 프로세스에서 지정된 메모리 범위에 대한 읽기 권한이 있는지 확인합니다. (IsBadReadPtr)
IsBadStringPtrA 호출 프로세스에서 지정된 메모리 범위에 대한 읽기 권한이 있는지 확인합니다. (IsBadStringPtrA)
IsBadStringPtrW 호출 프로세스에서 지정된 메모리 범위에 대한 읽기 권한이 있는지 확인합니다. (IsBadStringPtrW)
IsBadWritePtr 호출 프로세스에서 지정된 메모리 범위에 대한 쓰기 권한이 있는지 확인합니다.
IsBootableSystemStateBackedUp IsBootableSystemStateBackedUp 메서드는 부팅 가능한 상태가 백업될지 여부를 나타냅니다.
IsEnclaveTypeSupported 지정된 유형의 enclave가 지원되는지 여부를 검색합니다.
IsPartialFileSupportEnabled IsPartialFileSupportEnabled 메서드는 부분 파일 지원을 사용할지 아니면 사용하지 않도록 설정할지 여부를 결정합니다.

IsPathAffected IsPathAffected 메서드는 지정된 디렉터리 또는 파일이 현재 섀도 복사본 집합에 포함되어 있는지 여부를 결정합니다. 디렉터리 또는 파일의 경로는 로컬 경로 또는 원격 파일 공유의 UNC 경로일 수 있습니다.
IsPathSnapshotted 지정된 UNC(범용 명명 규칙) 경로에 현재 스냅샷이 있는지 여부를 확인합니다.
IsPathSupported 지정된 UNC(범용 명명 규칙) 경로가 이 공급자에서 지원되는지 여부를 결정합니다.
IsPwrHibernateAllowed 컴퓨터가 최대 절전 모드를 지원하는지 여부를 결정합니다.
IsPwrShutdownAllowed 컴퓨터가 소프트 끄기 전원 상태를 지원하는지 여부를 결정합니다.
IsPwrSuspendAllowed 컴퓨터가 절전 모드 상태를 지원하는지 여부를 결정합니다.
IsSelectedForRestore IsSelectedForRestore 메서드는 현재 구성 요소가 복원되도록 선택되었는지 여부를 결정합니다.
IsServiceReady VDS의 초기화 상태를 반환합니다.
IsSystemResumeAutomatic 컴퓨터의 현재 상태를 확인합니다.
IsVolumeSnapshotted IsVolumeSnapshotted 함수(vsbackup.h)는 지정된 볼륨에 대한 섀도 복사본이 있는지 여부를 결정합니다.
IsVolumeSnapshotted 지정된 볼륨에 대한 섀도 복사본이 있는지 여부를 확인합니다.

IsVolumeSnapshottedInternal IsVolumeSnapshottedInternal 함수(vsbackup.h)는 지정된 볼륨에 대한 섀도 복사본이 있는지 여부를 결정합니다.
IsVolumeSupported IsVolumeSupported 메서드는 지정된 공급자가 지정된 볼륨 또는 원격 파일 공유에서 섀도 복사본을 지원하는지 여부를 결정합니다.
IsVolumeSupported 공급자가 지정된 볼륨에서 섀도 복사본을 지원하는지 여부를 결정합니다.
IsWriterShuttingDown 작성기가 종료되는지 여부를 확인합니다.
LeaveCriticalSection 지정된 중요 섹션 개체의 소유권을 해제합니다.
LoadEnclaveData CreateEnclave를 호출하여 만든 초기화되지 않은 enclave에 데이터를 로드합니다.
LoadEnclaveImageA 이미지와 모든 가져오기를 enclave로 로드합니다. (ANSI)
LoadEnclaveImageW 이미지와 모든 가져오기를 enclave로 로드합니다. (유니코드)
LoadFromXML LoadFromXML 메서드는 기록기의 메타데이터 문서가 포함된 XML 문서를 IVssExamineWriterMetadata 인터페이스에 로드합니다.
LoadLibraryA 지정된 모듈을 호출 프로세스의 주소 공간에 로드합니다. (LoadLibraryA)
LoadLibraryExA 지정된 모듈을 호출 프로세스의 주소 공간에 로드합니다. (LoadLibraryExA)

LoadLibraryExW 지정된 모듈을 호출 프로세스의 주소 공간에 로드합니다. (LoadLibraryExW)
LoadLibraryW 지정된 모듈을 호출 프로세스의 주소 공간에 로드합니다. (LoadLibraryW)
LoadMetadata VSS가 기본 작성기 메타데이터 저장소 대신 문자열에서 기록기의 메타데이터를 로드하도록 합니다.
LoadModule 애플리케이션을 로드 및 실행하거나 기존 애플리케이션의 새 인스턴스를 만듭니다.
LoadPackagedLibrary 지정된 패키지된 모듈 및 해당 종속성을 호출 프로세스의 주소 공간에 로드합니다.
LoadService 지정된 컴퓨터에서 VDS를 시작하고 서비스 개체에 대한 포인터를 반환합니다.
LocalAlloc 힙에서 지정된 바이트 수를 할당합니다. (LocalAlloc)
LocalDiscard 지정된 로컬 메모리 개체를 삭제합니다. 메모리 개체의 잠금 수는 0이어야 합니다.
LocalFlags 지정된 로컬 메모리 개체에 대한 정보를 검색합니다.
LocalFree 지정된 로컬 메모리 개체를 해제하고 해당 핸들을 무효화합니다.
LocalHandle 로컬 메모리 개체에 대해 지정된 포인터와 연결된 핸들을 검색합니다.
LocalLock 로컬 메모리 개체를 잠그고 개체 메모리 블록의 첫 번째 바이트에 대한 포인터를 반환합니다.

LocalReAlloc 지정된 로컬 메모리 개체의 크기 또는 특성을 변경합니다. 크기가 증가하거나 감소할 수 있습니다.
LocalSize 지정된 로컬 메모리 개체의 현재 크기(바이트)를 검색합니다.
LocalUnlock LMEM_MOVEABLE 할당된 메모리 개체와 연결된 잠금 수를 감소합니다.
LocateLuns 컴퓨터에 섀도 복사본 LUN을 표시하도록 하드웨어 공급자에게 메시지를 표시합니다.
LockWorkStation 워크스테이션의 디스플레이를 잠근다.
LoginToTarget iSCSI 대상에 대한 수동 로그인을 수행합니다.
LogoutFromTarget 모든 iSCSI 세션의 iSCSI 대상에서 지정된 대상에 대한 수동 로그아웃을 수행합니다.
MapUserPhysicalPages 이전에 할당된 실제 메모리 페이지를 AWE(주소 창 확장) 영역의 지정된 주소에 매핑합니다. (MapUserPhysicalPages)
MapUserPhysicalPagesScatter 이전에 할당된 실제 메모리 페이지를 AWE(주소 창 확장) 영역의 지정된 주소에 매핑합니다. (MapUserPhysicalPagesScatter)
MapViewOfFile 파일 매핑 보기를 호출 프로세스의 주소 공간에 매핑합니다.
MapViewOfFile2 파일 또는 페이지 파일 기반 섹션의 보기를 지정된 프로세스의 주소 공간에 매핑합니다. (MapViewOfFile2)

MapViewOfFile3 파일 또는 페이지 파일 기반 섹션의 보기를 지정된 프로세스의 주소 공간에 매핑합니다. (MapViewOfFile3)
MapViewOfFile3FromApp 호출하는 Windows 스토어 앱의 주소 공간에 파일 매핑 보기를 매핑합니다. (MapViewOfFile3FromApp)
MapViewOfFileEx 파일 매핑 보기를 호출 프로세스의 주소 공간에 매핑합니다. 호출자는 필요에 따라 보기에 대해 제안된 기본 메모리 주소를 지정할 수 있습니다.
MapViewOfFileExNuma 파일 매핑 보기를 호출 프로세스의 주소 공간에 매핑하고 실제 메모리에 대한 NUMA 노드를 지정합니다.
MapViewOfFileFromApp 호출하는 Windows 스토어 앱의 주소 공간에 파일 매핑 보기를 매핑합니다. (MapViewOfFileFromApp)
MapViewOfFileNuma2 파일 또는 페이지 파일 기반 섹션의 보기를 지정된 프로세스의 주소 공간에 매핑합니다. (MapViewOfFileNuma2)
병합 자식 가상 디스크를 차이점 체인의 부모와 병합합니다.
MigrateDiffAreas 이 메서드는 나중에 사용하도록 예약됩니다. (IVssDifferentialSoftwareSnapshotMgmt2.MigrateDiffAreas)
MigrateDisks 디스크 집합을 한 팩에서 다른 팩으로 마이그레이션합니다.
마운트 볼륨을 탑재합니다.

다음 IEnumVdsObject::Next(vds.h) 메서드는 현재 지점에서 시작하여 열거형에 지정된 수의 개체를 반환합니다.
다음 IEnumVdsObject::Next 메서드(vdshwprv.h)는 현재 지점에서 시작하여 열거형에 지정된 수의 개체를 반환합니다.
다음 열거된 개체의 지정된 목록에서 지정한 개수의 개체를 반환합니다. (IVssEnumObject.Next)
다음 열거된 개체의 지정된 목록에서 지정한 개수의 개체를 반환합니다. (IVssEnumMgmtObject.Next)
NotifyChangeEventLog 지정된 이벤트 로그에 이벤트가 기록될 때 애플리케이션이 알림을 받을 수 있도록 합니다.
OfferVirtualMemory 메모리 페이지의 범위에 포함된 데이터가 애플리케이션에서 더 이상 필요하지 않으며 필요한 경우 시스템에서 삭제할 수 있음을 나타냅니다.
오프라인 디스크를 오프라인으로 전환합니다. Windows Vista: 이 방법은 Windows Vista SP1(서비스 팩 1)이 있는 경우 지원되지 않습니다. 대신 IVdsDisk2::SetSANMode를 사용합니다.
OfflineVolume IOCTL_VOLUME_OFFLINE 제어 코드를 사용하여 볼륨을 오프라인으로 전환합니다.
OnAbort OnAbort 메서드는 VSS에서 발급한 Abort 이벤트에 따라 기록기에서 호출되며, 이는 섀도 복사 작업이 조기에 종료되었음을 나타냅니다. 작성기는 이 메서드를 사용하여 해당 작업에 참여하려는 시도에서 정리합니다.
OnBackOffIOOnVolume 지원 안 됨 (CVssWriter.OnBackOffIOOnVolume)

OnBackupComplete OnBackupComplete 메서드는 BackupComplete 이벤트 다음에 기록기에 의해 호출됩니다. 백업 후 필요한 것으로 간주되는 작업을 수행하는 데 사용됩니다. 그러나 이러한 작업은 백업 구성 요소 문서를 수정할 수 없습니다.
OnBackupShutdown OnBackupShutdown 메서드는 BackupShutdown 이벤트 다음에 기록기에 의해 호출됩니다. 백업 애플리케이션이 종료될 때, 특히 백업 애플리케이션이 충돌하는 경우 필요한 것으로 간주되는 작업을 수행하는 데 사용됩니다.
OnContinueIOOnVolume 지원 안 됨 (CVssWriter.OnContinueIOOnVolume)
OnFreeze OnFreeze 메서드는 기록기에서 섀도 복사본 고정 시작 시 Freeze 이벤트를 수신할 때 호출됩니다. 작성기는 이 메서드를 사용하여 동결에 참여하거나 동결을 거부하는 데 필요한 작업을 수행합니다.
OnIdentify OnIdentify 메서드는 Identify 이벤트를 받은 후 기록기에서 호출됩니다.
OnIdentifyEx IVsCreateWriterMetadataEx 개체에 대한 포인터를 반환합니다.
온라인 디스크를 온라인으로 가져옵니다.
온라인 가능한 경우 볼륨을 정상 상태로 반환합니다. 이 메서드는 동적 디스크에 대해서만 지원됩니다.
Onload 공급자에게 자신을 초기화하라는 메시지를 표시하고 공급자가 필요한 인터페이스를 가져오는 데 사용하는 콜백 개체를 전달합니다.
Onload 공급자에게 로드되었음을 알 수 있습니다.

OnLunEmpty VSS에서 섀도 복사본 LUN에 흥미로운 데이터가 포함되어 있지 않은 것으로 판단할 때마다 호출됩니다.
OnLunStateChange VSS 서비스는 이 메서드를 호출하여 하드웨어 공급자에게 LUN 상태 변경을 알립니다.
OnNotify IVdsAdviseSink::OnNotify(vds.h) 메서드는 공급자에서 VDS로, VDS에서 애플리케이션으로 알림을 전달합니다.
OnNotify IVdsAdviseSink::OnNotify 메서드(vdshwprv.h)는 공급자에서 VDS로, VDS에서 애플리케이션으로 알림을 전달합니다.
OnPostRestore OnPostRestore 메서드는 PostRestore 이벤트 다음에 기록기에 의해 호출됩니다. 요청자가 파일을 디스크로 복원한 후 필요한 것으로 간주되는 작업을 수행하는 데 사용됩니다. 그러나 이러한 작업은 백업 구성 요소 문서를 수정할 수 없습니다.
OnPostSnapshot OnPostSnapshot 메서드는 PostSnapshot 이벤트 다음에 기록기에 의해 호출됩니다.
OnPrepareBackup OnPrepareBackup 메서드는 PrepareForBackup 이벤트 다음에 기록기에 의해 호출됩니다. 이 메서드는 백업 작업을 준비하기 위해 기록기 상태 및 해당 구성 요소를 구성하는 데 사용됩니다.
OnPrepareSnapshot 기록기에서 PrepareForSnapshot 이벤트를 처리하기 위해 OnPrepareSnapshot 메서드를 호출합니다. 섀도 복사본에 참여하거나 섀도 복사본을 거부하도록 작성자를 준비하는 데 필요한 작업을 수행하는 데 사용됩니다.
OnPreRestore OnPreRestore 메서드는 PreRestore 이벤트 다음에 기록기에 의해 호출됩니다.
OnReuseLuns 이 메서드는 나중에 사용하도록 예약됩니다. (IVssHardwareSnapshotProviderEx.OnReuseLuns)

OnThaw OnThaw 메서드는 Thaw 이벤트 다음에 기록기에 의해 호출됩니다.
OnUnload 공급자 개체에 자신을 초기화하지 말라는 메시지를 표시합니다.
OnUnload 공급자에게 언로드 준비를 하라는 것을 알 수 있습니다.
OnVSSApplicationStartup 지원 안 됨 (CVssWriter.OnVSSApplicationStartup)
OnVSSShutdown 지원되지 않습니다. (CVssWriter.OnVssShutdown)
열기 지정된 가상 디스크 파일에 대한 핸들을 열고 열린 핸들을 나타내는 개체에 대한 IVdsOpenVDisk 인터페이스 포인터를 반환합니다.
OpenBackupEventLogA BackupEventLog 함수에서 만든 백업 이벤트 로그에 대한 핸들을 엽니다. (ANSI)
OpenBackupEventLogW BackupEventLog 함수에서 만든 백업 이벤트 로그에 대한 핸들을 엽니다. (유니코드)
OpenCommPort 통신 디바이스를 열려고 시도합니다.
OpenEventA 기존 명명된 이벤트 개체를 엽니다. (ANSI)
OpenEventLogA 지정된 이벤트 로그에 대한 핸들을 엽니다. (ANSI)
OpenEventLogW 지정된 이벤트 로그에 대한 핸들을 엽니다. (유니코드)

OpenEventW 기존 명명된 이벤트 개체를 엽니다. (유니코드)
OpenFileMappingA 명명된 파일 매핑 개체를 엽니다. (OpenFileMappingA)
OpenFileMappingFromApp 명명된 파일 매핑 개체를 엽니다. (OpenFileMappingFromApp)
OpenFileMappingW 명명된 파일 매핑 개체를 엽니다. (OpenFileMappingW)
OpenMutexW 기존 명명된 뮤텍스 개체를 엽니다.
OpenSemaphoreW 기존 명명된 세마포 개체를 엽니다.
OpenWaitableTimerW 기존 명명된 대기 가능 타이머 개체를 엽니다.
PeekNamedPipe 명명된 파이프 또는 익명 파이프의 데이터를 파이프에서 제거하지 않고 버퍼에 복사합니다.
PINIT_ONCE_FN 애플리케이션 정의 콜백 함수입니다. InitOnceExecuteOnce 함수를 호출할 때 이 함수에 대한 포인터를 지정합니다.
PostCommitSnapshots 섀도 복사본 집합에 관련된 모든 공급자가 CommitSnapshots에 성공한 후에 호출됩니다.
PostFinalCommitSnapshots 섀도 복사본 자동 복구를 지원합니다.
PostRestore PostRestore 메서드를 사용하면 VSS가 PostRestore 이벤트를 생성하여 기록기에 현재 복원 작업이 완료되었음을 알릴 수 있습니다.

PowerCanRestoreIndividualDefaultPowerScheme 필요한 경우 복원할 수 있도록 현재 사용자가 지정된 전원 구성표의 데이터에 액세스할 수 있는지 여부를 확인합니다.
PowerClearRequest 전원 요청 개체에 대해 지정된 형식의 전원 요청 수를 감소합니다.
PowerCreatePossibleSetting 지정된 전원 설정에 대해 가능한 설정 값을 만듭니다.
PowerCreateRequest 새 전원 요청 개체를 만듭니다.
PowerCreateSetting 지정된 전원 설정에 대한 설정 값을 만듭니다.
PowerDeleteScheme 데이터베이스에서 지정된 전원 구성표를 삭제합니다.
PowerDeterminePlatformRole Windows 7, Windows Server 2008 R2, Windows Vista 또는 Windows Server 2008에 대한 컴퓨터 역할을 결정합니다.
PowerDeterminePlatformRoleEx 지정된 플랫폼에 대한 컴퓨터 역할을 결정합니다.
PowerDuplicateScheme 기존 전원 구성표를 복제합니다.
PowerEnumerate 전원 구성표에서 지정된 요소를 열거합니다.
PowerGetActiveScheme 활성 전원 구성표를 검색하고 스키마를 식별하는 GUID를 반환합니다.
PowerImportPowerScheme 파일에서 전원 구성표를 가져옵니다.

PowerIsSettingRangeDefined 지정된 전원 설정이 가능한 값의 범위를 나타내는지 여부를 쿼리합니다.
PowerReadACValue 지정된 전원 설정에 대한 AC 전원 값을 검색합니다.
PowerReadDCValue 지정된 전원 설정에 대한 DC 전원 값을 검색합니다.
PowerReadDescription 지정된 전원 설정, 하위 그룹 또는 체계에 대한 설명을 검색합니다.
PowerReadFriendlyName 지정된 전원 설정, 하위 그룹 또는 구성표의 이름을 검색합니다.
PowerReadIconResourceSpecifier 지정된 전원 설정, 하위 그룹 또는 체계에 대한 아이콘 리소스를 검색합니다.
PowerReadPossibleDescription 전원 설정 값의 가능한 선택 항목 중 하나에 대한 설명을 검색합니다.
PowerReadPossibleFriendlyName 전원 설정 값의 가능한 선택 항목 중 하나에 대한 식별 이름을 검색합니다.
PowerReadPossibleValue 전원 설정의 가능한 값에 대한 값을 검색합니다.
PowerReadSettingAttributes 지정된 전원 설정의 현재 특성을 반환합니다.
PowerReadValueIncrement 전원 설정 최소값과 최댓값 사이의 유효한 값에 대한 증분을 검색합니다.
PowerReadValueMax 지정된 전원 설정의 최대값을 검색합니다.

PowerReadValueMin 지정된 전원 설정의 최소값을 검색합니다.
PowerReadValueUnitsSpecifier 값 범위를 지원하는 전원 설정의 단위를 설명하는 데 사용되는 문자열을 읽습니다.
PowerRegisterForEffectivePowerModeNotifications 효과적인 전원 모드 변경 알림을 수신하도록 콜백을 등록합니다.
PowerRegisterSuspendResumeNotification 시스템이 일시 중단되거나 다시 시작될 때 알림을 받도록 등록합니다.
PowerRemovePowerSetting 지정된 전원 설정을 삭제합니다.
PowerReplaceDefaultPowerSchemes 기본 전원 구성표를 현재 사용자의 전원 구성표로 바꿉니다.
PowerReportThermalEvent 운영 체제에 열 이벤트를 알 수 있습니다.
PowerRestoreDefaultPowerSchemes 시스템의 전원 구성표를 기본 전원 구성표로 바꿉니다. 모든 현재 전원 구성표 및 설정이 삭제되고 기본 시스템 전원 구성표로 대체됩니다.
PowerRestoreIndividualDefaultPowerScheme 현재 사용자의 특정 전원 구성표를 기본 사용자(HKEY_USERS 저장됨)의 전원 구성표로 바꿉니다. 기본값).
PowerSetActiveScheme 현재 사용자의 활성 전원 구성표를 설정합니다.
PowerSetRequest 전원 요청 개체에 대해 지정된 형식의 전원 요청 수를 증분합니다.
PowerSettingAccessCheck 그룹 정책에 대한 쿼리는 지정된 전원 설정을 재정의합니다.

PowerSettingAccessCheckEx 그룹 정책에 대한 쿼리는 지정된 전원 설정을 재정의하고 설정에 대해 요청된 액세스를 지정합니다.
PowerSettingRegisterNotification 전원 설정이 변경될 때 알림을 받도록 등록합니다.
PowerSettingUnregisterNotification 전원 설정이 변경될 때 알림을 받도록 등록을 취소합니다.
PowerUnregisterFromEffectivePowerModeNotifications 유효 전원 모드 변경 알림에서 등록을 취소합니다. 이 함수는 정리 코드에서 호출될 예정이며 등록을 취소하기 전에 모든 콜백이 완료될 때까지 기다립니다.
PowerUnregisterSuspendResumeNotification 시스템이 일시 중단되거나 다시 시작될 때 알림을 받을 등록을 취소합니다.
PowerWriteDescription 지정된 전원 설정, 하위 그룹 또는 체계에 대한 설명을 설정합니다.
PowerWriteFriendlyName 지정된 전원 설정, 하위 그룹 또는 구성표의 이름을 설정합니다.
PowerWriteIconResourceSpecifier 지정된 전원 설정, 하위 그룹 또는 체계에 대한 아이콘 리소스를 설정합니다.
PowerWritePossibleDescription 전원 설정 값의 가능한 선택 항목 중 하나에 대한 설명을 설정합니다.
PowerWritePossibleFriendlyName 전원 설정의 지정된 가능한 설정에 대한 식별 이름을 설정합니다.
PowerWritePossibleValue 전원 설정의 가능한 값에 대한 값을 설정합니다.
PowerWriteSettingAttributes 전원 키의 전원 특성을 설정합니다.

PowerWriteValueIncrement 전원 설정 최소값과 최댓값 사이의 유효한 값에 대한 증분을 설정합니다.
PowerWriteValueMax 지정된 전원 설정의 최대값을 설정합니다.
PowerWriteValueMin 지정된 전원 설정의 최소값을 설정합니다.
PowerWriteValueUnitsSpecifier 값 범위를 지원하는 전원 설정의 단위를 설명하는 데 사용되는 문자열을 씁니다.
PreCommitSnapshots 공급자가 준비된 LUN을 신속하게 커밋할 준비가 되었는지 확인합니다.
PreFastRecovery 이 메서드는 나중에 사용하도록 예약됩니다. (IVssBackupComponentsEx2.PreFastRecovery)
PrefetchVirtualMemory 프로세스 주소 공간에서 잠재적으로 불연결한 가상 주소 범위를 메모리로 가져오는 효율적인 메커니즘을 제공합니다.
PreFinalCommitSnapshots 공급자가 섀도 복사본 자동 복구를 지원할 수 있습니다.
PrepareForBackup PrepareForBackup 메서드를 사용하면 VSS에서 PrepareForBackup 이벤트를 생성하여 기록기가 예정된 백업 작업을 준비하도록 알릴 수 있습니다. 이렇게 하면 요청자의 백업 구성 요소 문서를 작성자가 사용할 수 있습니다.
PreRestore PreRestore 메서드를 사용하면 VSS에서 PreRestore 이벤트를 생성하여 기록기가 예정된 복원 작업을 준비하도록 알릴 수 있습니다.
PSECURE_MEMORY_CACHE_CALLBACK 보안 메모리 범위가 해제되거나 보호가 변경될 때 호출되는 AddSecureMemoryCacheCallback 함수에 이전에 등록된 애플리케이션 정의 함수입니다.

PTIMERAPCROUTINE 애플리케이션에서 정의한 타이머 완료 루틴입니다. SetWaitableTimer 함수를 호출할 때 이 주소를 지정합니다.
PulseMaintenance IVdsMaintenance::P ulseMaintenance 메서드(vds.h)는 지정된 작업을 지정된 횟수만큼 수행합니다.
PulseMaintenance IVdsMaintenance::P ulseMaintenance(vdshwprv.h) 메서드는 지정된 작업을 지정된 횟수만큼 수행합니다.
PurgeComm 지정된 통신 리소스의 출력 또는 입력 버퍼에서 모든 문자를 삭제합니다. 리소스에 대한 보류 중인 읽기 또는 쓰기 작업을 종료할 수도 있습니다.
쿼리 Query 메서드는 시스템의 공급자 및/또는 현재 컨텍스트에 있는 시스템의 완료된 섀도 복사본을 쿼리합니다. 메서드는 백업 작업 중에만 호출할 수 있습니다.
쿼리 애플리케이션 서버에서 사용할 수 있는 모든 파일 공유 스냅샷에 대한 VSS_SNAPSHOT_PROP 구조체의 열거형을 가져옵니다.
쿼리 공급자가 완료한 섀도 복사본에 대한 정보를 공급자에게 쿼리합니다.
QueryAccessPaths 현재 볼륨에 대한 액세스 경로 목록과 드라이브 문자(있는 경우)를 반환합니다.
QueryActiveControllerPorts IVdsLunControllerPorts::QueryActiveControllerPorts 메서드(vds.h)는 현재 활성 컨트롤러 포트의 열거형을 반환합니다.
QueryActiveControllerPorts IVdsLunControllerPorts::QueryActiveControllerPorts(vdshwprv.h) 메서드는 현재 활성 컨트롤러 포트의 열거형을 반환합니다.

QueryActiveControllers IVdsLun::QueryActiveControllers 메서드(vds.h)는 LUN에 액세스할 수 있는 현재 활성 컨트롤러의 열거형을 반환합니다.
QueryActiveControllers IVdsLun::QueryActiveControllers(vdshwprv.h) 메서드는 현재 활성 컨트롤러의 열거형을 반환합니다.
QueryAllocatedLuns IVdsStoragePool::QueryAllocatedLuns 메서드(vds.h)는 스토리지 풀에 할당된 LUN을 열거하는 개체를 반환합니다.
QueryAllocatedLuns IVdsStoragePool::QueryAllocatedLuns(vdshwprv.h) 메서드는 스토리지 풀에 할당된 LUN을 열거하는 개체를 반환합니다.
QueryAllocatedStoragePools IVdsStoragePool::QueryAllocatedStoragePools 메서드(vds.h)는 공급자가 관리하는 할당된 스토리지 풀을 열거하는 개체를 반환합니다.
QueryAllocatedStoragePools IVdsStoragePool::QueryAllocatedStoragePools 메서드는 공급자가 관리하는 할당된 스토리지 풀을 열거하는 개체를 반환합니다.
QueryAssociatedLuns IVdsController::QueryAssociatedLuns(vds.h) 메서드는 컨트롤러가 연결된 LUN의 열거형을 반환합니다.
QueryAssociatedLuns IVdsControllerPort::QueryAssociatedLuns(vds.h) 메서드는 컨트롤러 포트가 연결된 LUN의 열거형을 반환합니다.
QueryAssociatedLuns IVdsIscsiTarget::QueryAssociatedLuns 메서드(vds.h)는 대상과 연결된 LUN의 열거형을 반환합니다.
QueryAssociatedLuns IVdsController::QueryAssociatedLuns(vdshwprv.h) 메서드는 컨트롤러가 연결된 LUN의 열거형을 반환합니다.

QueryAssociatedLuns IVdsControllerPort::QueryAssociatedLuns(vdshwprv.h) 메서드는 컨트롤러 포트가 연결된 LUN의 열거형을 반환합니다.
QueryAssociatedLuns IVdsIscsiTarget::QueryAssociatedLuns(vdshwprv.h) 메서드는 대상과 연결된 LUN의 열거형을 반환합니다.
QueryAssociatedPortalGroups IVdsIscsiPortal::QueryAssociatedPortalGroups 메서드(vds.h)는 포털이 연결된 포털 그룹의 열거형을 반환합니다.
QueryAssociatedPortalGroups IVdsIscsiPortal::QueryAssociatedPortalGroups(vdshwprv.h) 메서드는 포털이 연결된 포털 그룹의 열거형을 반환합니다.
QueryAssociatedPortals IVdsIscsiPortalGroup::QueryAssociatedPortals 메서드(vds.h)는 포털 그룹이 연결된 포털의 열거형을 반환합니다.
QueryAssociatedPortals IVdsIscsiPortalGroup::QueryAssociatedPortals(vdshwprv.h) 메서드는 포털 그룹이 연결된 포털의 열거형을 반환합니다.
QueryAssociatedTargets IVdsLunIscsi::QueryAssociatedTargets 메서드(vds.h)는 LUN에 대한 액세스 권한을 부여하는 현재 연결된 iSCSI 대상의 열거형을 반환합니다.
QueryAssociatedTargets IVdsLunIscsi::QueryAssociatedTargets(vdshwprv.h) 메서드는 현재 연결된 iSCSI 대상의 열거형을 반환합니다.
QueryControllerPorts IVdsControllerControllerPort::QueryControllerPorts(vds.h) 메서드는 컨트롤러의 포트를 열거하는 IEnumVdsObject 개체를 반환합니다.
QueryControllerPorts IVdsControllerControllerPort::QueryControllerPorts 메서드(vdshwprv.h)는 컨트롤러의 포트를 열거하는 IEnumVdsObject 개체를 반환합니다.

QueryControllers IVdsSubSystem::QueryControllers 메서드(vds.h)는 하위 시스템이 관리하는 온라인 및 오프라인 컨트롤러를 열거하는 개체를 반환합니다.
QueryControllers IVdsSubSystem::QueryControllers(vdshwprv.h) 메서드는 하위 시스템에 의해 관리되는 온라인 및 오프라인 컨트롤러를 열거하는 개체를 반환합니다.
QueryDiffAreasForSnapshot 입력 섀도 복사본과 연결된 원래 볼륨에서 사용 중인 섀도 복사본 스토리지 영역을 쿼리합니다.
QueryDiffAreasForVolume 볼륨에서 사용 중인 섀도 복사본 스토리지 영역을 쿼리합니다.
QueryDiffAreasOnVolume 지정된 볼륨에 물리적으로 상주하는 섀도 복사본 스토리지 영역을 쿼리합니다.
QueryDisks 팩의 디스크를 열거하는 개체를 반환합니다.
QueryDriveExtents IVdsStoragePool::QueryDriveExtents 메서드(vds.h)는 스토리지 풀에서 사용되는 드라이브 익스텐트의 배열을 반환합니다.
QueryDriveExtents IVdsStoragePool::QueryDriveExtents(vdshwprv.h) 메서드는 스토리지 풀에서 사용되는 드라이브 익스텐트의 배열을 반환합니다.
QueryDriveLetters 드라이브 문자 집합에 대한 속성 세부 정보를 반환합니다.
QueryDrives IVdsSubSystem::QueryDrives 메서드(vds.h)는 하위 시스템의 드라이브를 열거하는 개체를 반환합니다.
QueryDrives IVdsSubSystem::QueryDrives(vdshwprv.h) 메서드는 하위 시스템의 드라이브를 열거하는 개체를 반환합니다.

QueryExtents 디스크의 모든 익스텐트 세부 정보를 반환합니다.
QueryExtents IVdsDrive::QueryExtents 메서드(vds.h)는 할당된 익스텐트 및 할당되지 않은 익스텐트를 포함하여 드라이브의 익스텐트 배열을 반환합니다.
QueryExtents IVdsLunPlex::QueryExtents 메서드(vds.h)는 plex에 기여하는 드라이브 익스텐트의 배열을 반환합니다.
QueryExtents 현재 플렉스에 대한 모든 익스텐트 값을 반환합니다.
QueryExtents VdsDrive::QueryExtents(vdshwprv.h) 메서드는 할당된 익스텐트 및 할당되지 않은 익스텐트를 포함하여 드라이브의 익스텐트 배열을 반환합니다.
QueryExtents IVdsLunPlex::QueryExtents(vdshwprv.h) 메서드는 플렉스에 기여하는 드라이브 익스텐트의 배열을 반환합니다.
QueryFileSystemFormatSupport 볼륨 서식 지정에 지원되는 파일 시스템의 속성을 검색합니다.
QueryFileSystemTypes VDS에 알려진 모든 파일 시스템에 대한 속성 세부 정보를 반환합니다.
QueryFreeExtents 디스크의 사용 가능한 익스텐트 값을 반환하고 지정된 맞춤 크기에 맞춥니다.
QueryHbaPorts 로컬 시스템의 VDS에 알려진 HBA 포트 목록을 포함하는 IEnumVdsObject 열거형 개체를 반환합니다.
QueryHints IVdsLun::QueryHints 메서드(vds.h)는 현재 LUN에 적용된 힌트를 반환합니다.

QueryHints IVdsLunPlex::QueryHints 메서드(vds.h)는 현재 LUN 플렉스에 적용되는 힌트를 반환합니다.
QueryHints IVdsLun::QueryHints(vdshwprv.h) 메서드는 현재 LUN에 적용된 힌트를 반환합니다.
QueryHints IVdsLunPlex::QueryHints(vdshwprv.h) 메서드는 현재 LUN 플렉스에 적용되는 힌트를 반환합니다.
QueryHints2 IVdsLun2::QueryHints2 메서드(vds.h)는 현재 LUN에 적용된 힌트를 반환합니다.
QueryHints2 IVdsLun2::QueryHints2(vdshwprv.h) 메서드는 현재 LUN에 적용된 힌트를 반환합니다. 이 메서드는 IVdsLun::QueryHints 메서드와 동일합니다.
QueryIfCreatedLun VDS에서 하드웨어 공급자가 지정된 LUN을 관리하는지 여부를 확인할 수 있습니다.
QueryInitiatorAdapters 초기자의 iSCSI 초기자 어댑터를 열거하는 개체를 반환합니다.
QueryInitiatorPortals 초기자 어댑터의 iSCSI 초기자 포털을 열거하는 개체를 반환합니다.
QueryLuns IVdsSubSystem::QueryLuns 메서드(vds.h)는 하위 시스템에 표시되는 LUN의 열거형을 반환합니다. 이 메서드는 하드웨어 공급자 개체에만 적용됩니다.
QueryLuns IVdsSubSystem::QueryLuns(vdshwprv.h) 메서드는 하위 시스템에 표시되는 LUN의 열거형을 반환하고 하드웨어 공급자 개체에만 적용됩니다.
QueryMaskedDisks 지원되지 않습니다. 이 메서드는 나중에 사용할 수 있습니다. (IVdsService.QueryMaskedDisks)

QueryMaxLunCreateSize IVdsSubSystem::QueryMaxLunCreateSize 메서드(vds.h)는 지정된 LUN 형식 및 힌트를 사용하여 만들 수 있는 최대 LUN의 크기를 반환합니다.
QueryMaxLunCreateSize IVdsSubSystem::QueryMaxLunCreateSize(vdshwprv.h) 메서드는 지정된 LUN 형식 및 힌트를 사용하여 만들 수 있는 최대 LUN의 크기를 반환합니다.
QueryMaxLunCreateSize2 IVdsSubSystem2::QueryMaxLunCreateSize2 메서드(vds.h)는 지정된 LUN 형식 및 힌트를 사용하여 만들 수 있는 최대 LUN의 크기를 반환합니다.
QueryMaxLunCreateSize2 IVdsSubSystem2::QueryMaxLunCreateSize2(vdshwprv.h) 메서드는 지정된 LUN 형식 및 힌트를 사용하여 만들 수 있는 최대 LUN의 크기를 반환합니다.
QueryMaxLunCreateSizeInStoragePool IVdsHwProviderStoragePools::QueryMaxLunCreateSizeInStoragePool 메서드(vds.h)는 스토리지 풀에서 만들 수 있는 LUN의 최대 크기를 반환합니다.
QueryMaxLunCreateSizeInStoragePool IVdsHwProviderStoragePools::QueryMaxLunCreateSizeInStoragePool(vdshwprv.h) 메서드는 지정된 LUN 형식 및 힌트를 기반으로 스토리지 풀에서 만들 수 있는 LUN의 최대 크기를 반환합니다.
QueryMaxLunExtendSize IVdsLun::QueryMaxLunExtendSize 메서드(vds.h)는 LUN을 확장할 수 있는 최대 크기를 반환합니다.
QueryMaxLunExtendSize IVdsLun::QueryMaxLunExtendSize(vdshwprv.h) 메서드는 LUN을 확장할 수 있는 최대 크기를 반환합니다.
QueryMaxReclaimableBytes 현재 볼륨에서 회수할 수 있는 최대 바이트 수를 검색합니다.
QueryMedia 캐시의 디스크 속성을 업데이트. IVdsDisk::GetProperties를 호출하여 현재 미디어에 대한 업데이트된 세부 정보를 가져옵니다.

QueryMemoryResourceNotification 지정된 메모리 리소스 개체의 상태를 검색합니다.
QueryMigrationStatus 이 메서드는 나중에 사용하도록 예약됩니다. (IVssDifferentialSoftwareSnapshotMgmt2.QueryMigrationStatus)
QueryOptionalDelayLoadedAPI 지연 로드된 DLL의 지정된 함수를 시스템에서 사용할 수 있는지 여부를 결정합니다.
QueryPacks 소프트웨어 공급자가 관리하는 모든 팩을 포함하는 열거형 개체를 반환합니다.
QueryPartitionFileSystemFormatSupport 지정된 바이트 오프셋에서 디스크의 파티션 서식을 지정하는 데 지원되는 파일 시스템의 속성을 검색합니다.
QueryPartitions 현재 디스크에 있는 모든 파티션의 세부 정보를 반환합니다.
QueryPlexes IVdsLun::QueryPlexes 메서드(vds.h)는 LUN에서 플렉스의 열거형을 반환합니다.
QueryPlexes 볼륨의 플렉스를 열거하는 개체를 반환합니다.
QueryPlexes IVdsLun::QueryPlexes(vdshwprv.h) 메서드는 LUN에서 플렉스의 열거형을 반환합니다.
QueryPortalGroups IVdsIscsiTarget::QueryPortalGroups 메서드(vds.h)는 대상 내의 iSCSI 포털 그룹의 열거형을 반환합니다.
QueryPortalGroups IVdsIscsiTarget::QueryPortalGroups(vdshwprv.h) 메서드는 대상 내의 iSCSI 포털 그룹의 열거형을 반환합니다.

QueryPortals IVdsSubSystemIscsi::QueryPortals 메서드(vds.h)는 하위 시스템의 iSCSI 포털을 열거하는 개체를 반환합니다.
QueryPortals IVdsSubSystemIscsi::QueryPortals(vdshwprv.h) 메서드는 하위 시스템의 iSCSI 포털을 열거하는 개체를 반환합니다.
QueryProviders VDS에 알려진 하드웨어 및 소프트웨어 공급자 목록을 포함하는 열거형 개체를 반환합니다.
QueryProviders 등록된 모든 공급자를 쿼리합니다.
QueryReparsePoints 현재 볼륨에 대한 모든 재분석 지점을 반환합니다.
QueryRevertStatus 되돌리기 작업의 상태를 확인하는 데 사용할 수 있는 IVssAsync 인터페이스 포인터를 반환합니다. (IVssBackupComponents.QueryRevertStatus)
QueryRevertStatus 되돌리기 작업의 상태를 확인하는 데 사용할 수 있는 IVssAsync 인터페이스 포인터를 반환합니다. (IVssSoftwareSnapshotProvider.QueryRevertStatus)
QuerySnapshotDeltaBitmap 이 메서드는 나중에 사용하도록 예약됩니다. (IVssDifferentialSoftwareSnapshotMgmt3.QuerySnapshotDeltaBitmap)
QuerySnapshotsByVolume 시스템에서 사용하도록 예약되었습니다. (IVssSnapshotMgmt.QuerySnapshotsByVolume)
QueryStatus 비동기 작업이 진행 중이거나 성공적으로 완료되었거나 실패하면 IVdsAsync::QueryStatus(vds.h) 메서드가 반환됩니다.

QueryStatus 비동기 작업이 진행 중이거나, 성공적으로 완료되었거나, 실패하면 IVdsAsync::QueryStatus 메서드(vdshwprv.h)가 반환됩니다.
QueryStatus QueryStatus 메서드는 비동기 작업의 상태를 쿼리합니다.
QueryStoragePools QueryStoragePools 메서드(vds.h)는 하드웨어 공급자가 관리하는 스토리지 풀 목록을 포함하는 IEnumVdsObject 열거형 개체를 반환합니다.
QueryStoragePools IVdsHwProviderStoragePools::QueryStoragePools(vdshwprv.h) 메서드는 하드웨어 공급자가 관리하는 스토리지 풀 목록을 포함하는 IEnumVdsObject 열거형 개체를 반환합니다.
QuerySubSystems IVdsHwProvider::QuerySubSystems 메서드(vds.h)는 공급자가 관리하는 하위 시스템의 열거형을 반환합니다.
QuerySubSystems IVdsHwProvider::QuerySubSystems(vdshwprv.h) 메서드는 공급자가 관리하는 하위 시스템의 열거형을 반환합니다.
QueryTargets IVdsSubSystemIscsi::QueryTargets 메서드(vds.h)는 하위 시스템의 iSCSI 대상을 열거하는 개체를 반환합니다.
QueryTargets IVdsSubSystemIscsi::QueryTargets(vdshwprv.h) 메서드는 하위 시스템의 iSCSI 대상을 열거하는 개체를 반환합니다.
QueryUnallocatedDisks VDS에서 관리하는 할당되지 않은 디스크 목록을 포함하는 열거형 개체를 반환합니다.
QueryVDisks 공급자가 관리하는 모든 가상 디스크 목록을 반환합니다.

QueryVirtualMemoryInformation QueryVirtualMemoryInformation 함수는 지정된 프로세스의 가상 주소 공간 내에 있는 페이지 또는 페이지 집합에 대한 정보를 반환합니다.
QueryVolumeGuidPathnames 현재 볼륨의 볼륨 GUID 경로 목록을 반환합니다.
QueryVolumes 팩의 볼륨을 열거하는 개체를 반환합니다.
QueryVolumesSupportedForDiffAreas 섀도 복사본 스토리지 영역(비활성화된 섀도 복사본 스토리지 영역이 있는 볼륨 포함)을 지원하는 볼륨을 쿼리합니다.
QueryVolumesSupportedForSnapshots 시스템에서 사용하도록 예약되었습니다. (IVssSnapshotMgmt.QueryVolumesSupportedForSnapshots)
ReadEventLogA 지정된 이벤트 로그에서 지정된 수의 항목을 읽습니다. (ANSI)
ReadEventLogW 지정된 이벤트 로그에서 지정된 수의 항목을 읽습니다. (유니코드)
ReadGlobalPwrPolicy 현재 전역 전원 정책 설정을 검색합니다.
ReadProcessorPwrScheme 지정된 전원 구성표에 대한 프로세서 전원 정책 설정을 검색합니다.
ReadPwrScheme 지정된 전원 구성표에 고유한 전원 정책 설정을 검색합니다.
Reboot 공급자를 호스트하는 컴퓨터를 다시 시작합니다.

ReclaimVirtualMemory OfferVirtualMemory를 사용하여 시스템에 제공된 다양한 메모리 페이지를 회수합니다.
복구 IVdsLun::Recovery 메서드(vds.h)는 LUN에서 복구 작업을 시작합니다.
복구 가능하면 실패하거나 실패한 팩을 정상 상태로 반환합니다. 이 메서드는 동적 팩에 대해서만 지원됩니다.
복구 IVdsLun::Recovery(vdshwprv.h) 메서드는 LUN에서 복구 작업을 시작합니다.
RecoverSet LUN 다시 동기화 작업을 시작합니다.
다시 열거 IVdsHwProvider::Reenumerate 메서드(vds.h)는 새로 연결되고 연결이 끊긴 하위 시스템을 검색합니다.
다시 열거 새로 추가되고 새로 제거된 디스크를 검색합니다.
다시 열거 IVdsSubSystem::Reenumerate 메서드(vds.h)는 하위 시스템에 버스를 스캔하여 새로 연결된 드라이브 또는 새로 연결이 끊긴 드라이브를 검색하라는 메시지를 표시합니다.
다시 열거 IVdsHwProvider::Reenumerate(vdshwprv.h) 메서드는 새로 연결되고 연결이 끊긴 하위 시스템을 검색합니다.
다시 열거 IVdsSubSystem::Reenumerate(vdshwprv.h) 메서드는 하위 시스템에 버스를 스캔하여 새로 연결된 드라이브 또는 새로 연결이 끊긴 드라이브를 검색하라는 메시지를 표시합니다.
새로 고침 IVdsHwProvider::Refresh 메서드(vds.h)는 VDS 공급자가 관리하는 기존 하위 시스템에 대해 VDS의 내부적으로 캐시된 데이터를 새로 고칩니다.

새로 고침 디스크 소유권 및 디스크 레이아웃 정보를 새로 고칩니다.
새로 고침 IVdsHwProvider::Refresh(vdshwprv.h) 메서드는 VDS 공급자가 관리하는 기존 하위 시스템에 대한 VDS의 내부적으로 캐시된 데이터를 새로 고칩니다.
등록 VSS가 작성자의 메타데이터를 빠른 기록기 메타데이터 저장소에 저장하도록 합니다.
RegisterBadMemoryNotification 하나 이상의 잘못된 메모리 페이지가 검색될 때 호출되는 잘못된 메모리 알림을 등록합니다.
RegisterDeviceNotificationA 창에서 알림을 받을 디바이스 또는 디바이스 유형을 등록합니다. (ANSI)
RegisterDeviceNotificationW 창에서 알림을 받을 디바이스 또는 디바이스 유형을 등록합니다. (유니코드)
RegisterEventSourceA 지정된 이벤트 로그에 등록된 핸들을 검색합니다. (ANSI)
RegisterEventSourceW 지정된 이벤트 로그에 등록된 핸들을 검색합니다. (유니코드)
RegisterPowerSettingNotification 애플리케이션을 등록하여 특정 전원 설정 이벤트에 대한 전원 설정 알림을 받습니다.
RegisterProvider 지정된 하드웨어 공급자를 VDS에 등록합니다. 하드웨어 공급자는 이 메서드를 호출합니다.
RegisterProvider 새 섀도 복사본 공급자를 등록합니다.

RegisterSuspendResumeNotification 시스템이 일시 중단되거나 다시 시작될 때 알림을 받도록 등록합니다. PowerRegisterSuspendResumeNotification과 유사하지만 사용자 모드에서 작동하며 창 핸들을 사용할 수 있습니다.
ReleaseMutex 지정된 뮤텍스 개체의 소유권을 해제합니다.
ReleaseSemaphore 지정된 세마포 개체의 수를 지정된 양만큼 늘입니다.
ReleaseSRWLockExclusive 전용 모드에서 획득한 SRW(슬림 판독기/기록기) 잠금을 해제합니다.
ReleaseSRWLockShared 공유 모드에서 가져온 SRW(슬림 판독기/기록기) 잠금을 해제합니다.
RememberInitiatorSharedSecret RememberInitiatorSharedSecret 메서드(vds.h)는 초기자가 대상의 상호 CHAP 인증에 사용한 초기자 CHAP 비밀을 전달합니다.
RememberInitiatorSharedSecret IVdsIscsiTarget::RememberInitiatorSharedSecret(vdshwprv.h) 메서드는 상호 CHAP 인증에 사용되는 초기자 CHAP 비밀을 전달합니다.
RememberTargetSharedSecret 대상의 CHAP 공유 비밀을 초기자 서비스에 전달합니다. 이 공유 비밀은 대상이 초기자를 인증할 때 대상 로그인 중에 사용됩니다.
RemoveDllDirectory AddDllDirectory를 사용하여 프로세스 DLL 검색 경로에 추가된 디렉터리를 제거합니다.
RemoveMissingDisk 팩에서 누락된 디스크를 제거합니다. 이 메서드는 소프트웨어 공급자 개체에만 적용됩니다.
RemovePlex IVdsLun::RemovePlex 메서드(vds.h)는 LUN에서 플렉스를 제거합니다.

RemovePlex 현재 볼륨에서 하나 이상의 지정된 플렉스를 제거하고 익스텐트 해제합니다.
RemovePlex IVdsLun::RemovePlex(vdshwprv.h) 메서드는 LUN에서 플렉스를 제거합니다.
RemovePortal IVdsIscsiPortalGroup::RemovePortal 메서드(vds.h)는 포털 그룹에서 포털을 제거합니다.
RemovePortal IVdsIscsiPortalGroup::RemovePortal(vdshwprv.h) 메서드는 포털 그룹에서 포털을 제거합니다.
RemoveSecureMemoryCacheCallback 이전에 AddSecureMemoryCacheCallback 함수에 등록된 콜백 함수의 등록을 취소합니다.
복구 잘못된 멤버를 좋은 디스크로 이동하여 내결함성 볼륨 플렉스를 복구합니다.
ReplaceDisk 지원되지 않습니다. 이 메서드는 나중에 사용할 수 있습니다. (IVdsPack.ReplaceDisk)
ReplaceDrive IVdsSubSystem::ReplaceDrive 메서드(vds.h)는 하위 시스템의 한 드라이브를 다른 드라이브로 바꾸거나 마이그레이션합니다.
ReplaceDrive IVdsSubSystem::ReplaceDrive(vdshwprv.h) 메서드는 하위 시스템의 한 드라이브를 다른 드라이브로 바꾸거나 마이그레이션합니다.
ReportEventA 지정된 이벤트 로그의 끝에 항목을 씁니다. (ANSI)
ReportEventW 지정된 이벤트 로그의 끝에 항목을 씁니다. (유니코드)

RequestWakeupLatency 효과가 없으며 STATUS_NOT_SUPPORTED 반환합니다. 이 함수는 이전 버전의 Windows.Windows Server 2008 및 Windows Vista와의 호환성을 위해서만 제공됩니다. 효과가 없으며 항상 성공을 반환합니다.
Reset IEnumVdsObject::Reset(vds.h) 메서드는 열거형의 시작 부분으로 다시 설정됩니다.
Reset IVdsController::Reset(vds.h) 메서드는 컨트롤러를 다시 초기화하고 캐시를 무효화합니다.
Reset IVdsControllerPort::Reset(vds.h) 메서드는 컨트롤러 포트를 다시 초기화합니다.
Reset IEnumVdsObject::Reset 메서드(vdshwprv.h)는 열거형의 시작 부분으로 다시 설정됩니다.
Reset IVdsController::Reset(vdshwprv.h) 메서드는 컨트롤러를 다시 초기화하고 캐시를 무효화합니다.
Reset IVdsControllerPort::Reset(vdshwprv.h) 메서드는 컨트롤러 포트를 다시 초기화합니다.
Reset IVssEnumObject:Next가 첫 번째 열거형 개체에서 시작되도록 열거자를 다시 설정합니다.
Reset IVssEnumMgmtObject가 첫 번째 열거형 개체에서 시작되도록 열거자를 다시 설정합니다.
ResetEvent 지정된 이벤트 개체를 부호 없는 상태로 설정합니다.
ResetWriteWatch 가상 메모리 영역에 대한 쓰기 추적 상태를 다시 설정합니다. GetWriteWatch 함수에 대한 후속 호출은 다시 설정 작업 이후 작성된 보고서 페이지만 보고합니다.

ResyncLuns VSS 서비스는 이 메서드를 호출하여 하드웨어 공급자에게 LUN 다시 동기화가 필요하다고 알립니다.
RevertToSnapshot 볼륨을 이전 섀도 복사본으로 되돌려 줍니다. (IVssBackupComponents.RevertToSnapshot)
RevertToSnapshot 볼륨을 이전 섀도 복사본으로 되돌려 줍니다. (IVssSoftwareSnapshotProvider.RevertToSnapshot)
SaveAsXML 명시적 기록기 상태 정보가 포함된 기록기 메타데이터 문서를 지정된 문자열에 저장합니다.
SaveAsXML SaveAsXML 메서드는 기록기 상태 정보가 포함된 기록기 메타데이터 문서를 지정된 문자열에 저장합니다.
SaveAsXML SaveAsXML 메서드는 요청자의 상태 정보가 포함된 백업 구성 요소 문서를 지정된 문자열에 저장합니다. 백업 구성 요소 문서가 포함된 이 XML 문서는 항상 백업 작업의 일부로 안전하게 저장되어야 합니다.
SaveAsXML SaveAsXML 메서드는 기록기 상태 정보가 포함된 기록기 메타데이터 문서를 지정된 문자열에 저장합니다. 이 문자열은 백업 작업의 일부로 저장할 수 있습니다.
SetActivePwrScheme 활성 전원 구성표를 설정합니다.
SetAdditionalRestores SetAdditionalRestores 메서드는 증분 또는 차등 복원 작업 중에 요청자가 지정된 구성 요소를 완전히 검색하기 위해 추가 복원 작업이 필요하다는 것을 기록기에 나타내는 데 사용됩니다.
SetAllIpsecSecurity 지원되지 않습니다. (IVdsServiceIscsi.SetAllIpsecSecurity)

SetAllIpsecTunnelAddresses 지원되지 않습니다. 이 메서드는 나중에 사용하도록 예약되어 있습니다. (IVdsServiceIscsi.SetAllIpsecTunnelAddresses)
SetAllPathStatuses HBA 포트에서 시작된 모든 경로의 상태를 지정된 상태로 설정합니다.
SetAllPathStatusesFromHbaPort 특정 HBA 포트에서 시작된 경로의 상태를 지정된 상태로 설정합니다.
SetAuthoritativeRestore 구성 요소의 복원을 복제된 데이터 저장소에 대한 신뢰할 수 있는 것으로 표시합니다.
SetBackupMetadata SetBackupMetadata 메서드는 구성 요소를 사용하여 백업 메타데이터를 설정합니다.
SetBackupOptions SetBackupOptions 메서드는 구성 요소에 대한 프라이빗 또는 기록기 종속 백업 매개 변수의 문자열을 설정합니다.
SetBackupSchema 빠른 작성기에서 작성기 메타데이터 문서에 참여할 수 있는 백업 작업 유형을 나타내는 데 사용됩니다.
SetBackupSchema SetBackupSchema 메서드는 기록기 메타데이터 문서에 참여할 수 있는 백업 작업 유형을 나타내기 위해 작성기에서 사용됩니다.
SetBackupStamp SetBackupStamp 메서드는 백업이 발생한 시기를 나타내는 정보를 포함하는 문자열을 설정합니다.
SetBackupState SetBackupState 메서드는 백업 작업에 대한 전체 구성을 정의합니다.
SetBackupSucceeded SetBackupSucceeded 메서드는 특정 작성기의 지정된 구성 요소 백업이 성공했는지 여부를 나타냅니다.

SetCommBreak 지정한 통신 디바이스에 대한 문자 전송을 일시 중단하고 ClearCommBreak 함수가 호출될 때까지 전송 라인을 중단 상태로 설정합니다.
SetCommConfig 통신 디바이스의 현재 구성을 설정합니다.
SetCommMask 통신 디바이스에 대해 모니터링할 이벤트 집합을 지정합니다.
SetCommState 디바이스 제어 블록(DCB 구조)의 사양에 따라 통신 디바이스를 구성합니다. 함수는 모든 하드웨어 및 컨트롤 설정을 다시 초기화하지만 출력 또는 입력 큐는 비어 있지 않습니다.
SetCommTimeouts 지정된 통신 디바이스의 모든 읽기 및 쓰기 작업에 대한 제한 시간 매개 변수를 설정합니다.
Setcontext SetContext 메서드는 후속 섀도 복사본 관련 작업에 대한 컨텍스트를 설정합니다.
Setcontext 후속 섀도 복사본 관련 작업에 대한 컨텍스트를 설정합니다.
Setcontext 후속 섀도 복사본 관련 작업에 대한 컨텍스트를 설정합니다.
SetControllerStatus IVdsSubSystem::SetControllerStatus 메서드(vds.h)는 하위 시스템에 있는 컨트롤러의 상태(온라인 또는 오프라인)를 설정합니다.
SetControllerStatus IVdsSubSystem::SetControllerStatus(vdshwprv.h) 메서드는 하위 시스템에 있는 컨트롤러의 상태(온라인 또는 오프라인)를 설정합니다.
SetCriticalSectionSpinCount 지정된 중요 섹션의 스핀 수를 설정합니다.

SetDefaultCommConfigA 통신 디바이스의 기본 구성을 설정합니다. (ANSI)
SetDefaultCommConfigW 통신 디바이스의 기본 구성을 설정합니다. (유니코드)
SetDefaultDllDirectories 호출 프로세스가 DLL을 로드할 때 검색할 기본 디렉터리 집합을 지정합니다. 이 검색 경로는 LOAD_LIBRARY_SEARCH 플래그 없이 LoadLibraryEx를 호출할 때 사용됩니다.
SetDllDirectoryA 애플리케이션의 DLL을 찾는 데 사용되는 검색 경로에 디렉터리를 추가합니다. (ANSI)
SetDllDirectoryW 애플리케이션의 DLL을 찾는 데 사용되는 검색 경로에 디렉터리를 추가합니다. (유니코드)
SetEvent 지정된 이벤트 개체를 신호 상태로 설정합니다.
SetFailure VSS 기록기는 이 메서드를 호출하여 구성 요소 수준에서 오류를 보고합니다.
SetFileRestoreStatus SetFileRestoreStatus 메서드는 일부 파일, 전체 또는 파일이 성공적으로 복원되었는지 여부를 나타냅니다.
SetFileSystemFlags 파일 시스템 플래그를 설정합니다.
SetFlags 디스크 개체의 플래그를 설정합니다.
SetFlags IVdsDrive::SetFlags 메서드(vds.h)는 드라이브 개체의 플래그를 설정합니다.
SetFlags 서비스 개체 플래그를 설정합니다.

SetFlags 볼륨 플래그를 설정합니다.
SetFlags IVdsDrive::SetFlags(vdshwprv.h) 메서드는 드라이브 개체의 플래그를 설정합니다.
SetFriendlyName IVdsIscsiTarget::SetFriendlyName 메서드(vds.h)는 대상의 이름을 설정합니다.
SetFriendlyName IVdsLunNaming::SetFriendlyName 메서드(vds.h)는 LUN의 이름을 설정합니다.
SetFriendlyName IVdsSubSystemNaming::SetFriendlyName 메서드(vds.h)는 하위 시스템의 이름을 설정합니다.
SetFriendlyName IVdsIscsiTarget::SetFriendlyName(vdshwprv.h) 메서드는 대상의 이름을 설정합니다.
SetFriendlyName IVdsLunNaming::SetFriendlyName(vdshwprv.h) 메서드는 LUN의 이름을 설정합니다.
SetFriendlyName IVdsSubSystemNaming::SetFriendlyName(vdshwprv.h) 메서드는 하위 시스템의 이름을 설정합니다.
SetImportTarget 이 하위 시스템의 컴퓨터에 대한 VSS(볼륨 섀도 복사본 서비스) 가져오기 대상을 설정합니다.
SetInitiatorSharedSecret 초기자가 대상을 인증할 때 상호 CHAP 인증에 사용되는 초기자 CHAP 공유 비밀을 설정합니다.
SetIpsecGroupPresharedKey 지원되지 않습니다. 이 메서드는 나중에 사용할 수 있습니다. (IVdsServiceIscsi.SetIpsecGroupPresharedKey)

SetIpsecGroupPresharedKey IVdsSubSystemIscsi::SetIpsecGroupPresharedKey 메서드(vds.h)는 지원되지 않으며 나중에 사용하도록 예약되어 있습니다.
SetIpsecGroupPresharedKey IVdsSubSystemIscsi::SetIpsecGroupPresharedKey(vdshwprv.h) 메서드는 지원되지 않으며 나중에 사용하도록 예약되어 있습니다.
SetIpsecSecurity 지원되지 않습니다. 이 메서드는 나중에 사용할 수 있습니다. (IVdsIscsiInitiatorPortal.SetIpsecSecurity)
SetIpsecSecurity IVdsIscsiPortal::SetIpsecSecurity 메서드(vds.h)는 지원되지 않으며 나중에 사용하도록 예약되어 있습니다.
SetIpsecSecurity IVdsIscsiPortal::SetIpsecSecurity(vdshwprv.h) 메서드는 지원되지 않으며 나중에 사용하도록 예약되어 있습니다.
SetIpsecSecurityLocal 지원되지 않습니다. 이 메서드는 나중에 사용할 수 있습니다. (IVdsIscsiPortalLocal.SetIpsecSecurityLocal)
SetIpsecTunnelAddress 지원되지 않습니다. 이 메서드는 나중에 사용할 수 있습니다. (IVdsIscsiInitiatorPortal.SetIpsecTunnelAddress)
SetIpsecTunnelAddress IVdsIscsiPortal::SetIpsecTunnelAddress 메서드(vds.h)는 지원되지 않으며 나중에 사용하도록 예약되어 있습니다.
SetIpsecTunnelAddress IVdsIscsiPortal::SetIpsecTunnelAddress(vdshwprv.h) 메서드는 지원되지 않으며 나중에 사용하도록 예약되어 있습니다.
SetLoadBalancePolicy IVdsLunMpio::SetLoadBalancePolicy 메서드(vds.h)는 LUN에서 부하 분산 정책을 설정합니다.

SetLoadBalancePolicy IVdsLunMpio::SetLoadBalancePolicy(vdshwprv.h) 메서드는 LUN에서 부하 분산 정책을 설정합니다.
SetMailslotInfo 지정된 mailslot에서 읽기 작업에 사용하는 제한 시간 값을 설정합니다.
SetMask IVdsLun::SetMask 메서드(vds.h)는 LUN에 대한 액세스 권한을 부여할 컴퓨터 목록인 마스킹 해제 목록을 지정합니다.
SetMask IVdsLun::SetMask(vdshwprv.h) 메서드는 LUN에 대한 액세스 권한을 부여할 컴퓨터 목록인 마스킹 해제 목록을 지정합니다.
SetNamedPipeHandleState 지정된 명명된 파이프의 읽기 모드 및 차단 모드를 설정합니다. 지정된 핸들이 명명된 파이프의 클라이언트 끝에 있고 명명된 파이프 서버 프로세스가 원격 컴퓨터에 있는 경우 함수를 사용하여 로컬 버퍼링을 제어할 수도 있습니다.
SetPostRestoreFailureMsg SetPostRestoreFailureMsg 메서드는 PostRestore 이벤트 처리 실패를 설명하는 메시지를 만드는 데 사용됩니다.
SetPostSnapshotFailureMsg 구성 요소에 대한 PostSnapshot 실패 메시지 문자열을 설정합니다.
SetPrepareForBackupFailureMsg 구성 요소에 대한 PrepareForBackup 실패 메시지 문자열을 설정합니다.
SetPreRestoreFailureMsg SetPreRestoreFailureMsg 메서드는 PreRestore 이벤트 처리 실패를 설명하는 메시지를 만드는 데 사용됩니다.
SetPreviousBackupStamp SetPreviousBackupStamp 메서드는 차등 또는 증분 백업 작업의 기반이 되는 이전 백업 작업의 백업 스탬프를 설정합니다.

SetProcessDEPPolicy 32비트 프로세스에 대한 DEP(데이터 실행 방지) 및 DEP-ATL unk 에뮬레이션 설정을 변경합니다.
SetProcessValidCallTargets CFG(Control Flow Guard)에 유효한 간접 호출 대상 목록을 제공하고 유효한 것으로 표시할지 여부를 지정합니다.
SetRangesFilePath SetRangesFilePath 메서드는 부분 파일 작업에 범위 파일이 필요하고 해당 파일이 원래 파일이 아닌 다른 위치로 복원된 경우에 사용됩니다.
SetRestoreMetadata SetRestoreMetadata 메서드는 현재 구성 요소에 대한 기록기별 메타데이터를 설정합니다.
SetRestoreMethod 빠른 기록기 데이터를 복원하는 방법을 지정합니다.
SetRestoreMethod SetRestoreMethod 메서드는 기록기 데이터를 복원하는 방법을 나타냅니다.
SetRestoreName 복원 중인 구성 요소에 새 논리 이름을 할당합니다.
SetRestoreOptions SetRestoreOptions 메서드는 기록기 구성 요소에 대한 프라이빗 또는 기록기 종속 복원 매개 변수의 문자열을 설정합니다.
SetRestoreState SetRestoreState 메서드는 복원 작업에 대한 전체 구성을 정의합니다.
SetRestoreTarget SetRestoreTarget 메서드는 현재 구성 요소에 대한 복원 대상(VSS_RESTORE_TARGET 열거형 측면에서)을 설정합니다.
SetRollForward 구성 요소의 롤포워드 작업 유형을 설정하고 부분 롤포워드 작업의 복원 지점을 지정합니다.

SetSANMode 디스크의 SAN 모드를 오프라인 또는 온라인으로 설정합니다.
SetSANPolicy 운영 체제에 대한 디스크 SAN 정책을 설정합니다.
SetSelectedForRestore SetSelectedForRestore 메서드는 지정된 선택 가능한 구성 요소가 복원을 위해 선택되었는지 여부를 나타냅니다.
SetSelectedForRestoreEx SetSelectedForRestoreEx 메서드는 지정된 작성기 인스턴스로 복원하기 위해 지정된 선택 가능한 구성 요소를 선택했는지 여부를 나타냅니다.
SetSharedSecret IVdsIscsiTarget::SetSharedSecret 메서드(vds.h)는 대상이 초기자를 인증할 때 CHAP 인증에 사용되는 대상 CHAP 공유 비밀을 설정합니다.
SetSharedSecret IVdsIscsiTarget::SetSharedSecret 메서드는 대상이 초기자를 인증할 때 CHAP 인증에 사용되는 대상 CHAP 공유 비밀을 설정합니다.
SetSnapshotPriority 이 메서드는 나중에 사용하도록 예약됩니다. (IVssDifferentialSoftwareSnapshotMgmt2.SetSnapshotPriority)
SetSnapshotProperty 공급자에게 지정된 스냅샷에 대한 속성 값을 설정하도록 요청합니다.
SetSnapshotProperty 섀도 복사본의 속성을 설정합니다.
SetStatus IVdsController::SetStatus(vds.h) 메서드는 컨트롤러의 상태를 지정된 값으로 설정합니다.
SetStatus IVdsControllerPort::SetStatus(vds.h) 메서드는 컨트롤러 포트의 상태를 지정된 값으로 설정합니다.

SetStatus IVdsDrive::SetStatus 메서드(vds.h)는 드라이브의 상태를 지정된 값으로 설정합니다.
SetStatus IVdsIscsiPortal::SetStatus 메서드(vds.h)는 포털의 상태를 지정된 값으로 설정합니다.
SetStatus IVdsLun::SetStatus 메서드(vds.h)는 LUN의 상태를 지정된 값으로 설정합니다.
SetStatus IVdsSubSystem::SetStatus 메서드(vds.h)는 하위 시스템의 상태를 지정된 값으로 설정합니다.
SetStatus IVdsController::SetStatus(vdshwprv.h) 메서드는 컨트롤러의 상태를 지정된 값으로 설정합니다.
SetStatus IVdsControllerPort::SetStatus(vdshwprv.h) 메서드는 컨트롤러 포트의 상태를 지정된 값으로 설정합니다.
SetStatus IVdsDrive::SetStatus(vdshwprv.h) 메서드는 드라이브의 상태를 지정된 값으로 설정합니다.
SetStatus IVdsIscsiPortal::SetStatus(vdshwprv.h) 메서드는 포털의 상태를 지정된 값으로 설정합니다.
SetStatus IVdsLun::SetStatus(vdshwprv.h) 메서드는 LUN의 상태를 지정된 값으로 설정합니다.
SetStatus IVdsSubSystem::SetStatus(vdshwprv.h) 메서드는 하위 시스템의 상태를 지정된 값으로 설정합니다.
SetSuspendState 전원을 종료하여 시스템을 일시 중단합니다. 최대 절전 모드 매개 변수에 따라 시스템은 일시 중단(절전 모드) 상태 또는 최대 절전 모드(S4)를 입력합니다.
SetSystemFileCacheSize 파일 시스템 캐시에 대한 작업 집합의 크기를 제한합니다.

SetSystemPowerState 전원을 종료하여 시스템을 일시 중단합니다. ForceFlag 매개 변수에 따라 함수는 작업을 즉시 일시 중단하거나 작업을 수행하기 전에 모든 애플리케이션 및 디바이스 드라이버의 권한을 요청합니다.
SetThreadExecutionState 애플리케이션이 사용 중임을 시스템에 알릴 수 있으므로 애플리케이션이 실행되는 동안 시스템이 절전 모드로 전환되거나 디스플레이가 꺼지는 것을 방지할 수 있습니다.
SetupComm 지정된 통신 디바이스에 대한 통신 매개 변수를 초기화합니다.
SetVolumeProtectLevel 원래 볼륨 또는 섀도 복사본 스토리지 영역 볼륨에 대한 섀도 복사본 보호 수준을 설정합니다.
SetWaitableTimer 지정된 대기 가능 타이머를 활성화합니다. 기한이 되면 타이머가 신호를 받고 타이머를 설정하는 스레드는 선택적 완료 루틴을 호출합니다.
SetWaitableTimerEx 지정된 대기 가능 타이머를 활성화하고 타이머에 대한 컨텍스트 정보를 제공합니다. 기한이 되면 타이머가 신호를 받고 타이머를 설정하는 스레드는 선택적 완료 루틴을 호출합니다.
SetWriterFailure SetWriterFailure 메서드는 이 작성기가 오류 조건을 발견했으며 오류 조건을 설정했음을 나타냅니다.
SetWriterFailureEx 확장 오류 정보를 설정하여 기록기에서 VSS 작업에 참여하는 데 문제가 있음을 나타냅니다.
ShouldBlockRevert 지정된 볼륨에서 되돌리기 작업을 차단해야 하는 기록기에 대한 레지스트리를 확인합니다.
축소 IVdsLun::Shrink 메서드(vds.h)는 지정된 바이트 수만큼 LUN을 축소합니다.
축소 볼륨 및 모든 플렉스의 크기를 줄이고 해제된 익스텐트 공간을 여유 공간으로 반환합니다.

축소 볼륨과 모든 플렉스를 축소하고 해제된 익스텐트 값을 반환합니다.
축소 IVdsLun::Shrink(vdshwprv.h) 메서드는 지정된 바이트 수만큼 LUN을 축소합니다.
ShutdownBlockReasonCreate 시스템을 종료할 수 없음을 나타내고 시스템 종료가 시작된 경우 사용자에게 표시할 이유 문자열을 설정합니다.
ShutdownBlockReasonDestroy 시스템을 종료하고 이유 문자열을 해제할 수 있음을 나타냅니다.
ShutdownBlockReasonQuery ShutdownBlockReasonCreate 함수에서 설정한 이유 문자열을 검색합니다.
SignalObjectAndWait 한 개체에 신호를 전송하고 다른 개체를 단일 작업으로 기다립니다.
Skip IEnumVdsObject::Skip(vds.h) 메서드는 열거형에서 지정된 수의 개체를 건너뜁니다.
Skip IEnumVdsObject::Skip 메서드(vdshwprv.h)는 열거형에서 지정된 수의 개체를 건너뜁니다.
Skip 지정된 수의 개체를 건너뜁니다. (IVssEnumObject.Skip)
Skip 지정된 수의 개체를 건너뜁니다. (IVssEnumMgmtObject.Skip)
Sleep 제한 시간 간격이 경과할 때까지 현재 스레드의 실행을 일시 중단합니다.
SleepConditionVariableCS 지정된 조건 변수에서 절전 모드로 전환하고 지정된 위험 섹션을 원자성 작업으로 해제합니다.

SleepConditionVariableSRW 지정된 조건 변수에서 절전 모드로 전환하고 지정된 잠금을 원자성 작업으로 해제합니다.
SleepEx 지정된 조건이 충족될 때까지 현재 스레드를 일시 중단합니다.
StartMaintenance IVdsMaintenance::StartMaintenance 메서드(vds.h)는 유지 관리 작업을 시작합니다.
StartMaintenance IVdsMaintenance::StartMaintenance(vdshwprv.h) 메서드는 유지 관리 작업을 시작합니다.
StartSnapshotSet StartSnapshotSet 메서드는 빈 새 섀도 복사본 집합을 만듭니다.
StopMaintenance IVdsMaintenance::StopMaintenance 메서드(vds.h)는 유지 관리 작업을 중지합니다.
StopMaintenance IVdsMaintenance::StopMaintenance(vdshwprv.h) 메서드는 유지 관리 작업을 중지합니다.
구독 Subscribe 메서드는 VSS를 사용하여 작성기를 구독합니다.
SubscribeEx 기록기가 VSS 이벤트를 구독하도록 합니다.
TerminateEnclave enclave 내에서 실행되는 스레드의 실행을 종료합니다.
TransactNamedPipe 메시지를 쓰고 지정된 명명된 파이프에서 메시지를 읽는 함수를 단일 작업으로 결합합니다.
TransmitCommChar 지정된 통신 디바이스의 출력 버퍼에서 보류 중인 데이터보다 먼저 지정된 문자를 전송합니다.

TryAcquireSRWLockExclusive 전용 모드에서 SRW(슬림 판독기/기록기) 잠금을 획득하려고 시도합니다. 호출이 성공하면 호출 스레드가 잠금의 소유권을 맡습니다.
TryAcquireSRWLockShared 공유 모드에서 SRW(슬림 판독기/기록기) 잠금을 획득하려고 시도합니다. 호출이 성공하면 호출 스레드가 잠금의 소유권을 맡습니다.
TryEnterCriticalSection 차단 없이 중요한 섹션을 입력하려고 시도합니다. 호출이 성공하면 호출 스레드는 중요한 섹션의 소유권을 맡습니다.
Unadvise 호출자가 더 이상 VDS 서비스에서 알림을 받지 않도록 호출자의 IVdsAdviseSink 인터페이스를 등록 취소합니다.
UnexposeSnapshot 파일 공유를 삭제하거나 드라이브 문자 또는 탑재된 폴더를 제거하여 섀도 복사본의 노출을 제거합니다.
UninstallDisks 디스크 집합을 제거합니다.
UnmapViewOfFile 호출 프로세스의 주소 공간에서 파일의 매핑된 뷰를 매핑 해제합니다.
UnmapViewOfFile2 이전에 매핑된 파일 또는 페이지 파일 기반 섹션의 매핑을 해제합니다.
UnmapViewOfFileEx 추가 flags 매개 변수를 사용하는 UnmapViewOfFile의 확장 버전입니다.
등록 취소 VSS가 기본 작성기 메타데이터 저장소에서 기록기 메타데이터를 삭제하도록 합니다.
UnregisterBadMemoryNotification 지정된 잘못된 메모리 알림 핸들을 닫습니다.

UnregisterDeviceNotification 지정된 디바이스 알림 핸들을 닫습니다.
UnregisterPowerSettingNotification 전원 설정 알림을 등록 취소합니다.
UnregisterProvider VDS 공급자 등록 데이터를 제거합니다. 하드웨어 공급자는 이 메서드를 호출합니다.
UnregisterProvider 기존 공급자의 등록을 취소합니다.
UnregisterSuspendResumeNotification 시스템이 일시 중단되거나 다시 시작될 때 알림을 받을 등록을 취소합니다. PowerUnregisterSuspendResumeNotification과 유사하지만 사용자 모드에서 작동합니다.
구독 취소 Unsubscribe 메서드는 VSS를 사용하여 작성기를 구독 취소합니다.
VirtualAlloc 호출 프로세스의 가상 주소 공간에서 페이지 영역의 상태를 예약, 커밋 또는 변경합니다. (VirtualAlloc)
VirtualAlloc2 지정된 프로세스의 가상 주소 공간 내에서 메모리 영역의 상태를 예약, 커밋 또는 변경합니다. 함수는 할당한 메모리를 0으로 초기화합니다. (VirtualAlloc2)
VirtualAlloc2FromApp 호출 프로세스의 가상 주소 공간에서 페이지 영역의 상태를 예약, 커밋 또는 변경합니다. (VirtualAlloc2FromApp)
VirtualAllocEx 지정된 프로세스의 가상 주소 공간 내에서 메모리 영역의 상태를 예약, 커밋 또는 변경합니다. 함수는 할당한 메모리를 0으로 초기화합니다. (VirtualAllocEx)
VirtualAllocExNuma 지정된 프로세스의 가상 주소 공간 내에서 메모리 영역의 상태를 예약, 커밋 또는 변경하고 실제 메모리에 대한 NUMA 노드를 지정합니다.

VirtualAllocFromApp 호출 프로세스의 가상 주소 공간에서 페이지 영역의 상태를 예약, 커밋 또는 변경합니다. (VirtualAllocFromApp)
VirtualFree 호출 프로세스의 가상 주소 공간 내에서 페이지 영역을 해제, 커밋 해제 또는 해제합니다.
VirtualFreeEx 지정된 프로세스의 가상 주소 공간 내에서 메모리 영역을 해제, 커밋 해제 또는 해제합니다.
VirtualLock 프로세스의 가상 주소 공간의 지정된 영역을 실제 메모리에 잠가서 해당 지역에 대한 후속 액세스에서 페이지 오류가 발생하지 않도록 합니다.
VirtualProtect 호출 프로세스의 가상 주소 공간에서 커밋된 페이지의 영역에 대한 보호를 변경합니다. (VirtualProtect)
VirtualProtectEx 지정된 프로세스의 가상 주소 공간에서 커밋된 페이지의 영역에 대한 보호를 변경합니다.
VirtualProtectFromApp 호출 프로세스의 가상 주소 공간에서 커밋된 페이지의 영역에 대한 보호를 변경합니다. (VirtualProtectFromApp)
Virtualquery 호출 프로세스의 가상 주소 공간에서 페이지 범위에 대한 정보를 검색합니다.
VirtualQueryEx 지정된 프로세스의 가상 주소 공간 내의 페이지 범위에 대한 정보를 검색합니다.
VirtualUnlock 프로세스의 가상 주소 공간에서 지정된 페이지 범위의 잠금을 해제하여 필요한 경우 시스템이 페이지를 페이징 파일로 교환할 수 있도록 합니다.
VssFreeSnapshotProperties VssFreeSnapshotProperties 함수(vsbackup.h)는 수명 주기 관리의 일환으로 VSS_SNAPSHOT_PROP 구조체의 콘텐츠를 해제하는 데 사용됩니다.

VssFreeSnapshotPropertiesInternal VssFreeSnapshotPropertiesInternal 함수(vsbackup.h)는 수명 주기 관리의 일환으로 VSS_SNAPSHOT_PROP 구조체의 내용을 해제하는 데 사용됩니다.
Wait 비동기 작업이 성공적으로 완료되었거나 실패하면 IVdsAsync::Wait(vds.h) 메서드가 반환됩니다.
Wait 비동기 작업이 성공적으로 완료되었거나 실패하면 IVdsAsync::Wait 메서드(vdshwprv.h)가 반환됩니다.
Wait Wait 메서드는 불완전한 비동기 작업이 완료될 때까지 기다립니다.
WaitCommEvent 지정된 통신 디바이스에 대한 이벤트가 발생할 때까지 기다립니다. 이 함수에서 모니터링하는 이벤트 집합은 디바이스 핸들과 연결된 이벤트 마스크에 포함됩니다.
WaitForMultipleObjects 지정된 개체 중 하나 또는 전부가 신호 상태이거나 제한 시간 간격이 경과할 때까지 기다립니다.
WaitForMultipleObjectsEx 지정된 개체 중 하나 또는 전부가 신호 상태에 있거나, I/O 완료 루틴 또는 APC(비동기 프로시저 호출)가 스레드에 큐에 대기하거나 시간 제한 간격이 경과할 때까지 기다립니다.
WaitForServiceReady VDS 초기화가 완료될 때까지 기다렸다가 VDS 초기화 상태를 반환합니다.
WaitForSingleObject 지정된 개체가 신호 상태가 되거나 시간 제한 간격이 경과할 때까지 기다립니다.
WaitForSingleObjectEx 지정된 개체가 신호 상태, I/O 완료 루틴 또는 APC(비동기 프로시저 호출)가 스레드에 큐에 대기하거나 시간 제한 간격이 경과할 때까지 기다립니다.
WaitOnAddress 지정된 주소의 값이 변경될 때까지 기다립니다.

WakeAllConditionVariable 지정된 조건 변수에서 대기 중인 모든 스레드를 절 해제합니다.
WakeByAddressAll 주소 값이 변경되기를 기다리는 모든 스레드를 해제합니다.
WakeByAddressSingle 주소 값이 변경되는 것을 기다리는 스레드 하나를 절 해제합니다.
WakeConditionVariable 지정된 조건 변수에서 대기 중인 단일 스레드를 절 해제합니다.
WriteGlobalPwrPolicy 전역 전원 정책 설정을 씁니다.
WriteProcessorPwrScheme 지정된 전원 구성표에 대한 프로세서 전원 정책 설정을 씁니다.
WritePwrScheme 지정된 전원 구성표에 고유한 정책 설정을 씁니다.

인터페이스

IEnumVdsObject IEnumVdsObject 인터페이스(vds.h)는 지정된 형식의 VDS 개체 집합을 통해 열거합니다.
IEnumVdsObject IEnumVdsObject(vdshwprv.h) 인터페이스는 지정된 형식의 VDS 개체 집합을 통해 열거됩니다.
IVdsAdmin VDS에 공급자를 등록합니다.
IVdsAdvancedDisk 파티션을 만들고 삭제하고 파티션 특성을 수정합니다.

IVdsAdvancedDisk2 파티션 형식을 변경하는 메서드를 제공합니다.
IVdsAdviseSink IVdsAdviseSink 인터페이스(vds.h)는 VDS 알림을 받습니다.
IVdsAdviseSink IVdsAdviseSink(vdshwprv.h) 인터페이스는 VDS 알림을 받습니다.
IVdsAsync IVdsAsync 인터페이스(vds.h)는 비동기 작업을 관리합니다.
IVdsAsync IVdsAsync(vdshwprv.h) 인터페이스는 비동기 작업을 관리합니다. 비동기 작업을 시작하는 메서드는 IVdsAsync 인터페이스에 대한 포인터를 반환합니다.
IVdsController IVdsController 인터페이스(vds.h)는 컨트롤러에서 쿼리 및 구성 작업을 수행하기 위한 메서드를 제공합니다.
IVdsController IVdsController(vdshwprv.h) 인터페이스는 컨트롤러에서 쿼리 및 구성 작업을 수행하기 위한 메서드를 제공합니다.
IVdsControllerControllerPort IVdsControllerControllerPort 인터페이스(vds.h)는 IVdsController 인터페이스를 구현하는 클래스에 대한 컨트롤러 포트를 열거하는 메서드를 제공합니다.
IVdsControllerControllerPort IVdsControllerControllerPort(vdshwprv.h) 인터페이스는 IVdsController 인터페이스를 구현하는 클래스에 대한 컨트롤러 포트를 열거하는 메서드를 제공합니다.
IVdsControllerPort IVdsControllerPort 인터페이스(vds.h)는 컨트롤러 포트에서 쿼리 및 구성 작업을 수행하기 위한 메서드를 제공합니다.

IVdsControllerPort IVdsControllerPort(vdshwprv.h) 인터페이스는 컨트롤러 포트에서 쿼리 및 구성 작업을 수행하기 위한 메서드를 제공합니다.
IVdsCreatePartitionEx 기본 디스크에 파티션을 만듭니다.
IVdsDisk 기본 및 동적 디스크를 쿼리하고 구성하는 메서드를 제공합니다.
IVdsDisk2 디스크의 SAN 모드를 오프라인 또는 온라인으로 설정하는 방법을 제공합니다.
IVdsDisk3 디스크의 위치 경로를 포함하여 디스크에 대한 속성 정보를 검색하는 메서드를 제공합니다.
IVdsDiskOnline 디스크를 온라인 상태로 전환하고 오프라인으로 전환할 수 있는 메서드를 제공합니다. Windows Vista: 이 인터페이스는 Windows Vista SP1(서비스 팩 1)까지 지원되지 않습니다. 대신 IVdsDisk2를 사용합니다.
IVdsDiskPartitionMF 파티션에서 파일 시스템 관리 작업을 수행하는 메서드를 제공합니다.
IVdsDiskPartitionMF2 추가 서식 옵션을 사용하여 파티션의 서식을 지정하는 메서드를 제공합니다.
IVdsDrive IVdsDrive 인터페이스(vds.h)는 드라이브에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsDrive IVdsDrive(vdshwprv.h) 인터페이스는 드라이브에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsDrive2 IVdsDrive2 인터페이스(vds.h)는 드라이브의 속성을 쿼리하는 메서드를 제공합니다.

IVdsDrive2 IVdsDrive2(vdshwprv.h) 인터페이스는 드라이브의 속성을 쿼리하는 메서드를 제공합니다.
IVdsHbaPort 로컬 시스템에서 HBA 포트를 쿼리하고 상호 작용하는 메서드를 제공합니다.
IVdsHwProvider IVdsHwProvider 인터페이스(vds.h)는 하드웨어 공급자에서 쿼리, 다시 열거 및 새로 고침 작업을 수행하는 메서드를 제공합니다.
IVdsHwProvider IVdsHwProvider(vdshwprv.h) 인터페이스는 하드웨어 공급자에서 쿼리, 다시 활성화 및 새로 고침 작업을 수행하는 메서드를 제공합니다.
IVdsHwProviderPrivate VDS가 하드웨어 공급자가 지정된 LUN을 관리하는지 여부를 확인할 수 있도록 하는 메서드를 제공합니다.
IVdsHwProviderPrivateMpio 특정 HBA 포트에서 공급자로 시작되는 경로의 상태를 설정하는 메서드를 제공합니다.
IVdsHwProviderStoragePools IVdsHwProviderStoragePools 인터페이스(vds.h)는 스토리지 풀에서 LUN을 만들고 하드웨어 공급자가 관리하는 스토리지 풀을 열거하는 메서드를 제공합니다.
IVdsHwProviderStoragePools IVdsHwProviderStoragePools 인터페이스(vdshwprv.h)는 스토리지 풀에서 LUN을 만들고 하드웨어 공급자가 관리하는 스토리지 풀을 열거하는 메서드를 제공합니다.
IVdsHwProviderType IVdsHwProviderType 인터페이스(vds.h)는 하드웨어 공급자 유형을 검색하는 메서드를 제공합니다.
IVdsHwProviderType IVdsHwProviderType 인터페이스(vdshwprv.h)는 하드웨어 공급자 유형을 검색하는 메서드를 제공합니다.

IVdsHwProviderType2 IVdsHwProviderType2 인터페이스(vds.h)는 구현되지 않습니다. 대신 IVdsHwProviderType을 사용합니다.
IVdsHwProviderType2 IVdsHwProviderType2 인터페이스(vdshwprv.h)는 구현되지 않습니다. 대신 IVdsHwProviderType 인터페이스(vds.h)를 사용합니다.
IVdsIscsiInitiatorAdapter 로컬 시스템에서 iSCSI 초기자 어댑터를 쿼리하고 상호 작용하는 메서드를 제공합니다.
IVdsIscsiInitiatorPortal 로컬 시스템에서 iSCSI 초기자 포털을 쿼리하고 상호 작용하는 메서드를 제공합니다.
IVdsIscsiPortal IVdsIscsiPortal 인터페이스(vds.h)는 iSCSI 포털에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsIscsiPortal IVdsIscsiPortal 인터페이스(vdshwprv.h)는 iSCSI 포털에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsIscsiPortalGroup IVdsIscsiPortalGroup 인터페이스(vds.h)는 iSCSI 포털 그룹에서 쿼리 및 구성 서비스를 수행하는 메서드를 제공합니다.
IVdsIscsiPortalGroup IVdsIscsiPortalGroup 인터페이스(vdshwprv.h)는 iSCSI 포털 그룹에서 쿼리 및 구성 서비스를 수행하는 메서드를 제공합니다.
IVdsIscsiPortalLocal iSCSI 포털에서 로컬 초기자별 IPSEC 사전 공유 키를 설정하는 메서드를 제공합니다.
IVdsIscsiTarget IVdsIscsiTarget 인터페이스(vds.h)는 iSCSI 대상에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.

IVdsIscsiTarget IVdsIscsiTarget 인터페이스(vdshwprv.h)는 iSCSI 대상에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsLun IVdsLun 인터페이스(vds.h)는 LUN(논리 단위 번호)에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsLun IVdsLun 인터페이스(vdshwprv.h)는 LUN(논리 단위 번호)에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsLun2 IVdsLun2 인터페이스(vds.h)는 LUN(논리 단위 번호) 힌트를 적용하고 쿼리하는 메서드를 제공합니다.
IVdsLun2 IVdsLun2 인터페이스(vdshwprv.h)는 LUN(논리 단위 번호) 힌트를 적용하고 쿼리하는 메서드를 제공합니다.
IVdsLunControllerPorts IVdsLunControllerPorts 인터페이스(vds.h)는 LUN에서 컨트롤러 포트 구성 작업을 수행하기 위한 메서드를 제공합니다.
IVdsLunControllerPorts IVdsLunControllerPorts 인터페이스(vdshwprv.h)는 LUN에서 컨트롤러 포트 구성 작업을 수행하는 메서드를 제공합니다.
IVdsLunIscsi IVdsLunIscsi 인터페이스(vds.h)는 iSCSI LUN에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsLunIscsi IVdsLunIscsi 인터페이스(vdshwprv.h)는 iSCSI LUN에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsLunMpio IVdsLunMpio 인터페이스(vds.h)는 MPIO 확장을 사용하여 LUN에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.

IVdsLunMpio IVdsLunMpio 인터페이스(vdshwprv.h)는 MPIO 확장을 사용하여 LUN에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsLunNaming IVdsLunNaming 인터페이스(vds.h)는 IVdsLun 인터페이스를 구현하는 클래스의 LUN 이름을 지정하는 메서드를 제공합니다.
IVdsLunNaming IVdsLunNaming 인터페이스(vdshwprv.h)는 IVdsLun 인터페이스를 구현하는 클래스의 LUN 이름을 지정하는 메서드를 제공합니다.
IVdsLunNumber IVdsLunNumber 인터페이스(vds.h)는 LUN에 대한 LUN 번호를 쿼리하는 메서드를 제공합니다.
IVdsLunNumber IVdsLunNumber 인터페이스(vdshwprv.h)는 LUN에 대한 LUN 번호를 쿼리하는 메서드를 제공합니다.
IVdsLunPlex IVdsLunPlex 인터페이스(vds.h)는 LUN 플렉스에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsLunPlex IVdsLunPlex 인터페이스(vdshwprv.h)는 LUN 플렉스에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsMaintenance IVdsMaintenance 인터페이스(vds.h)는 하위 시스템, 컨트롤러, LUN 또는 드라이브에서 유지 관리 작업을 수행하는 메서드를 제공합니다.
IVdsMaintenance IVdsMaintenance 인터페이스(vdshwprv.h)는 하위 시스템, 컨트롤러, LUN 또는 드라이브에서 유지 관리 작업을 수행하는 메서드를 제공합니다.
IVdsOpenVDisk 가상 디스크를 관리하기 위한 메서드를 정의합니다. (IVdsOpenVDisk)

IVdsPack 디스크 및 볼륨이 포함된 팩에서 관리 작업을 쿼리하고 수행하는 메서드를 제공합니다.
IVdsPack2 팩에 정렬된 볼륨을 만드는 메서드를 제공합니다.
IVdsProvider IVdsProvider 인터페이스(vds.h)는 하드웨어 또는 소프트웨어 공급자의 속성을 반환합니다.
IVdsProvider IVdsProvider 인터페이스(vdshwprv.h)는 하드웨어 또는 소프트웨어 공급자의 속성을 반환합니다.
IVdsProviderPrivate VDS가 공급자 개체에서 기타 작업을 수행할 수 있도록 하는 메서드를 제공합니다.
IVdsProviderSupport IVdsProviderSupport 인터페이스(vds.h)는 공급자가 지원하는 VDS 인터페이스 버전을 나타내는 메서드를 제공합니다.
IVdsProviderSupport IVdsProviderSupport 인터페이스(vdshwprv.h)는 공급자가 지원하는 VDS 인터페이스 버전을 나타내는 메서드를 제공합니다.
IVdsRemovable CD-ROM과 같은 이동식 디스크를 쿼리하고 꺼내는 메서드를 제공합니다.
IVdsService VDS를 쿼리하고 상호 작용하는 메서드를 제공합니다.
IVdsServiceHba 로컬 시스템에서 HBA 포트를 쿼리하는 메서드를 제공합니다.
IVdsServiceInitialization 이 인터페이스는 시스템 사용을 위해 예약되어 있습니다.

IVdsServiceIscsi CHAP 보안 설정을 설정하고 대상에 로그인하는 기능을 포함하여 로컬 초기자 서비스와 인터페이스하는 메서드를 제공합니다.
IVdsServiceLoader 서비스 로더 개체를 인스턴스화합니다.
IVdsServiceSAN 운영 체제에 대한 온라인 및 오프라인 SAN 정책을 관리하는 방법을 제공합니다.
IVdsServiceUninstallDisk 기본 및 동적 디스크를 제거하는 메서드를 제공합니다.
IVdsStoragePool IVdsStoragePool 인터페이스(vds.h)는 정보를 쿼리하고 스토리지 풀에 대한 관련 개체를 열거하는 메서드를 제공합니다.
IVdsStoragePool IVdsStoragePool 인터페이스(vdshwprv.h)는 정보를 쿼리하고 스토리지 풀에 대한 관련 개체를 열거하는 메서드를 제공합니다.
IVdsSubSystem IVdsSubSystem 인터페이스(vds.h)는 하위 시스템에 대한 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsSubSystem IVdsSubSystem 인터페이스(vdshwprv.h)는 하위 시스템에 대한 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsSubSystem2 IVdsSubSystem2 인터페이스(vds.h)는 VDS_HINTS2 및 VDS_SUB_SYSTEM_PROP2 구조를 사용하여 하위 시스템에 대한 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsSubSystem2 IVdsSubSystem2 인터페이스는 VDS_HINTS2 및 VDS_SUB_SYSTEM_PROP2 구조를 사용하여 하위 시스템에 대한 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.

IVdsSubSystemImportTarget 하위 시스템에 대한 기본 VSS 가져오기 대상을 쿼리하고 구성하는 메서드를 제공합니다.
IVdsSubSystemInterconnect IVdsSubSystemInterconnect 인터페이스(vds.h)는 하위 시스템이 지원하는 상호 연결 형식을 쿼리하는 메서드를 제공합니다.
IVdsSubSystemInterconnect IVdsSubSystemInterconnect 인터페이스(vdshwprv.h)는 하위 시스템이 지원하는 상호 연결 형식을 쿼리하는 메서드를 제공합니다.
IVdsSubSystemIscsi IVdsSubSystemIscsi 인터페이스(vds.h)는 하위 시스템의 iSCSI 대상 및 포털을 쿼리하고 구성하는 메서드를 제공합니다.
IVdsSubSystemIscsi IVdsSubSystemIscsi 인터페이스(vdshwprv.h)는 하위 시스템의 iSCSI 대상 및 포털을 쿼리하고 구성하는 메서드를 제공합니다.
IVdsSubSystemNaming IVdsSubSystemNaming 인터페이스(vds.h)는 IVdsSubSystem 인터페이스를 구현하는 클래스에 대한 하위 시스템의 이름을 지정하는 메서드를 제공합니다.
IVdsSubSystemNaming IVdsSubSystemNaming 인터페이스(vdshwprv.h)는 IVdsSubSystem 인터페이스를 구현하는 클래스에 대한 하위 시스템의 이름을 지정하는 메서드를 제공합니다.
IVdsSwProvider 소프트웨어 공급자와 관련된 작업을 수행하는 메서드를 제공합니다.
IVdsVDisk 가상 디스크를 관리하기 위한 메서드를 정의합니다. (IVdsVDisk)
IVdsVdProvider 가상 디스크를 만들고 관리하기 위한 메서드를 정의합니다.
IVdsVolume 볼륨을 관리하는 메서드를 제공합니다.

IVdsVolume2 볼륨 GUID를 포함하여 볼륨 속성 정보를 반환하는 메서드를 제공합니다.
IVdsVolumeMF 볼륨 개체에서 액세스 경로 및 파일 시스템 작업을 수행하는 메서드를 제공합니다.
IVdsVolumeMF2 볼륨 개체에 대한 추가 파일 시스템 관리 작업을 수행하는 메서드를 제공합니다. (IVdsVolumeMF2)
IVdsVolumeMF3 볼륨 개체에 대한 추가 파일 시스템 관리 작업을 수행하는 메서드를 제공합니다. (IVdsVolumeMF3)
IVdsVolumeOnline 단일 볼륨을 복구하는 메서드를 제공합니다.
IVdsVolumePlex 볼륨 플렉스를 유지 관리하기 위한 메서드를 제공합니다.
IVdsVolumeShrink 볼륨 축소를 지원하는 메서드를 제공합니다.
IVssAdmin IVssAdmin 인터페이스는 VSS에 등록된 공급자를 관리합니다.
IVssAsync IVssAsync 인터페이스는 비동기 작업을 시작하는 메서드를 통해 호출 애플리케이션에 반환되며, 백그라운드에서 실행되며 일반적으로 완료하는 데 시간이 오래 걸립니다.
IVssDifferentialSoftwareSnapshotMgmt 애플리케이션이 시스템 섀도 복사본 공급자에 의해 생성된 섀도 복사본 스토리지 영역을 쿼리하고 관리할 수 있도록 하는 메서드를 포함합니다.
IVssDifferentialSoftwareSnapshotMgmt2 애플리케이션이 시스템 섀도 복사본 공급자에 의해 생성된 섀도 복사본 스토리지 영역을 쿼리하고 관리할 수 있도록 하는 추가 메서드를 정의합니다.

IVssDifferentialSoftwareSnapshotMgmt3 애플리케이션이 VSS의 섀도 복사본 보호 기능을 사용할 수 있도록 하는 메서드를 정의합니다.
IVssEnumMgmtObject 열거된 개체 목록에서 반복하고 다른 작업을 수행하는 메서드를 포함합니다. (IVssEnumMgmtObject)
IVssEnumObject 열거된 개체 목록에서 반복하고 다른 작업을 수행하는 메서드를 포함합니다. (IVssEnumObject)
IVssFileShareSnapshotProvider . (IVssFileShareSnapshotProvider)
IVssHardwareSnapshotProvider VSS에서 볼륨을 LUN에 매핑하고, 섀도 복사 프로세스 중에 생성된 LUN을 검색하고, SAN에서 LUN을 전송하는 데 사용하는 메서드를 포함합니다.
IVssHardwareSnapshotProviderEx VSS에서 하드웨어 공급자에게 LUN 상태 변경을 알리는 데 사용하는 추가 메서드를 제공합니다.
IVssProviderCreateSnapshotSet IVssProviderCreateSnapshotSet 인터페이스에는 섀도 복사본을 만드는 동안 사용되는 메서드가 포함되어 있습니다.
IVssProviderNotifications IVssProviderNotifications 인터페이스는 VSS에 등록된 공급자를 관리합니다.
IVssSnapshotMgmt 섀도 복사본 공급자를 추가로 구성하는 인터페이스를 반환하는 메서드를 제공합니다.
IVssSnapshotMgmt2 섀도 복사본 스토리지 영역의 최소 크기를 검색하는 메서드를 제공합니다.
IVssSoftwareSnapshotProvider VSS에서 섀도 복사본 볼륨을 관리하는 데 사용하는 메서드를 포함합니다. 모든 소프트웨어 공급자는 이 인터페이스를 지원해야 합니다.

구조체

_DEV_BROADCAST_USERDEFINED DBT_USERDEFINED 디바이스 이벤트와 연결된 사용자 정의 이벤트 및 선택적 데이터를 포함합니다.
ADMINISTRATOR_POWER_POLICY 관리자가 전원 정책 설정을 재정의하는 것을 나타냅니다.
BATTERY_REPORTING_SCALE IOCTL_BATTERY_QUERY_STATUS 보고하는 배터리 용량의 세분성을 포함합니다.
CHANGE_ATTRIBUTES_PARAMETERS 파티션 스타일의 파티션 매개 변수를 정의합니다. (CHANGE_ATTRIBUTES_PARAMETERS)
CHANGE_PARTITION_TYPE_PARAMETERS 파티션의 형식을 변경할 때 사용할 매개 변수에 대해 설명합니다.
CHANGER_ELEMENT 변경자 요소를 나타냅니다.
CHANGER_ELEMENT_LIST 일반적으로 여러 요소의 상태를 가져오거나 초기화하는 등의 작업에 대해 단일 형식의 요소 범위를 나타냅니다.
CHANGER_ELEMENT_STATUS 지정된 요소의 상태를 나타냅니다. (CHANGER_ELEMENT_STATUS)
CHANGER_ELEMENT_STATUS_EX 지정된 요소의 상태를 나타냅니다. (CHANGER_ELEMENT_STATUS_EX)
CHANGER_EXCHANGE_MEDIUM IOCTL_CHANGER_EXCHANGE_MEDIUM 제어 코드에서 미디어 조각을 대상으로 이동하는 데 사용하는 정보와 원래 첫 번째 대상에 있는 미디어 조각을 두 번째 대상으로 이동하는 정보가 포함되어 있습니다.

CHANGER_INITIALIZE_ELEMENT_STATUS 모든 미디어 체인저 요소 또는 특정 형식의 지정된 요소의 상태를 나타냅니다.
CHANGER_MOVE_MEDIUM IOCTL_CHANGER_MOVE_MEDIUM 제어 코드가 미디어 조각을 대상으로 이동하는 데 사용하는 정보를 포함합니다.
CHANGER_PRODUCT_DATA 체인저 디바이스의 제품 데이터를 나타냅니다. IOCTL_CHANGER_GET_PRODUCT_DATA 제어 코드에서 사용됩니다.
CHANGER_READ_ELEMENT_STATUS IOCTL_CHANGER_GET_ELEMENT_STATUS 제어 코드가 상태를 검색할 요소를 결정하는 데 필요한 정보를 포함합니다.
CHANGER_SEND_VOLUME_TAG_INFORMATION IOCTL_CHANGER_QUERY_VOLUME_TAGS 제어 코드가 검색할 볼륨 정보를 결정하는 데 사용하는 정보가 포함되어 있습니다.
CHANGER_SET_ACCESS IOCTL_CHANGER_SET_ACCESS 제어 코드가 디바이스의 삽입/꺼내기 포트, 도어 또는 키패드의 상태를 설정하는 데 필요한 정보를 포함합니다.
CHANGER_SET_POSITION 변경자의 로봇 전송 메커니즘을 지정된 요소 주소로 설정하기 위해 IOCTL_CHANGER_SET_POSITION 제어 코드에 필요한 정보를 포함합니다.
CLASS_MEDIA_CHANGE_CONTEXT 미디어 변경 이벤트와 관련된 정보를 포함합니다.
COMMCONFIG 통신 디바이스의 구성 상태에 대한 정보를 포함합니다.
COMMPROP 통신 드라이버에 대한 정보를 포함합니다.
COMMTIMEOUTS 통신 디바이스에 대한 제한 시간 매개 변수를 포함합니다.

COMSTAT 통신 디바이스에 대한 정보를 포함합니다.
CREATE_PARTITION_PARAMETERS 파티션 스타일의 파티션 매개 변수를 정의합니다. (CREATE_PARTITION_PARAMETERS)
DCB 직렬 통신 디바이스에 대한 제어 설정을 정의합니다.
DEV_BROADCAST_DEVICEINTERFACE_A 디바이스 클래스에 대한 정보를 포함합니다. (ANSI)
DEV_BROADCAST_DEVICEINTERFACE_W 디바이스 클래스에 대한 정보를 포함합니다. (유니코드)
DEV_BROADCAST_HANDLE 파일 시스템 핸들에 대한 정보를 포함합니다.
DEV_BROADCAST_HDR WM_DEVICECHANGE 메시지를 통해 보고된 디바이스 이벤트와 관련된 정보에 대한 표준 헤더 역할을 합니다.
DEV_BROADCAST_OEM OEM 정의 디바이스 유형에 대한 정보를 포함합니다.
DEV_BROADCAST_PORT_A 모뎀, 직렬 또는 병렬 포트에 대한 정보를 포함합니다. (ANSI)
DEV_BROADCAST_PORT_W 모뎀, 직렬 또는 병렬 포트에 대한 정보를 포함합니다. (유니코드)
DEV_BROADCAST_VOLUME 논리 볼륨에 대한 정보를 포함합니다.
DEVICE_DATA_SET_LB_PROVISIONING_STATE IOCTL_STORAGE_MANAGE_DATA_SET_ATTRIBUTES 제어 코드의 DeviceDsmAction_Allocation 동작에 대한 출력 구조입니다.

DEVICE_DATA_SET_RANGE IOCTL_STORAGE_MANAGE_DATA_SET_ATTRIBUTES 제어 코드에 사용할 데이터 세트 범위 정보를 제공합니다.
DEVICE_DATA_SET_REPAIR_PARAMETERS 복구 작업에 대한 매개 변수를 지정합니다.
DEVICE_DSM_NOTIFICATION_PARAMETERS IOCTL_STORAGE_MANAGE_DATA_SET_ATTRIBUTES 제어 코드에 대한 DeviceDsmAction_Notification 작업에 대한 매개 변수를 포함합니다.
DEVICE_DSM_OFFLOAD_READ_PARAMETERS IOCTL_STORAGE_MANAGE_DATA_SET_ATTRIBUTES 제어 코드에 대한 DeviceDsmAction_OffloadRead 작업에 대한 매개 변수를 포함합니다.
DEVICE_DSM_OFFLOAD_WRITE_PARAMETERS 오프로드 쓰기 작업에 대한 매개 변수를 지정합니다.
DEVICE_MANAGE_DATA_SET_ATTRIBUTES IOCTL_STORAGE_MANAGE_DATA_SET_ATTRIBUTES 제어 코드에 대한 입력 구조입니다.
DEVICE_MANAGE_DATA_SET_ATTRIBUTES_OUTPUT IOCTL_STORAGE_MANAGE_DATA_SET_ATTRIBUTES 제어 코드의 출력 구조입니다.
DEVICE_MEDIA_INFO 디바이스에서 지원하는 미디어에 대한 정보를 제공합니다.
DEVICE_NOTIFY_SUBSCRIBE_PARAMETERS 전원 알림에 등록할 때 사용되는 매개 변수를 포함합니다.
ENCLAVE_CREATE_INFO_SGX enclave 형식이 ENCLAVE_TYPE_SGX 때 enclave를 만드는 데 사용할 아키텍처 관련 정보를 포함합니다. 이 정보는 SGX(Intel Software Guard Extensions) 아키텍처 확장에 대한 enclave를 지정합니다.
ENCLAVE_CREATE_INFO_VBS enclave 형식이 ENCLAVE_TYPE_VBS 때 enclave를 만드는 데 사용할 아키텍처 관련 정보가 포함되어 있습니다. 이 정보는 VBS(가상화 기반 보안) enclave를 지정합니다.

ENCLAVE_IDENTITY enclave의 기본 모듈의 ID를 설명합니다.
ENCLAVE_INFORMATION 현재 실행 중인 enclave에 대한 정보를 포함합니다.
ENCLAVE_INIT_INFO_SGX Enclave 형식이 ENCLAVE_TYPE_SGX 때 enclave를 초기화하는 데 사용할 아키텍처 관련 정보를 포함합니다. 이 정보는 SGX(Intel Software Guard Extensions) 아키텍처 확장에 대한 enclave를 지정합니다.
ENCLAVE_INIT_INFO_VBS enclave 형식이 ENCLAVE_TYPE_VBS 때 enclave를 초기화하는 데 사용할 아키텍처 관련 정보가 포함되어 있습니다. 이 정보는 VBS(가상화 기반 보안) enclave를 지정합니다.
EVENTLOG_FULL_INFORMATION 이벤트 로그가 가득 찼는지 여부를 나타냅니다.
EVENTLOGRECORD ReadEventLog 함수에서 반환된 이벤트 레코드에 대한 정보를 포함합니다.
GET_CHANGER_PARAMETERS 변경자의 매개 변수를 나타냅니다.
GET_MEDIA_TYPES 디바이스에서 지원하는 미디어 유형에 대한 정보를 포함합니다.
GLOBAL_MACHINE_POWER_POLICY 모든 사용자의 모든 전원 구성표에 적용되는 전역 컴퓨터 전원 정책 설정을 포함합니다.
GLOBAL_POWER_POLICY 모든 전원 구성표에 적용되는 전역 전원 정책 설정을 포함합니다.
GLOBAL_USER_POWER_POLICY 사용자의 모든 전원 구성표에 적용되는 전역 사용자 전원 정책 설정을 포함합니다.

HEAP_OPTIMIZE_RESOURCES_INFORMATION HeapSetInformation으로 시작된 HeapOptimizeResources 작업에 대한 플래그를 지정합니다.
IMAGE_ENCLAVE_CONFIG32 32비트 Windows를 실행하는 시스템의 Enclave 구성 형식을 정의합니다. (32비트)
IMAGE_ENCLAVE_CONFIG64 32비트 Windows를 실행하는 시스템의 Enclave 구성 형식을 정의합니다. (64비트)
IMAGE_ENCLAVE_IMPORT Enclave에서 가져올 수 있는 이미지 배열의 항목을 정의합니다.
MACHINE_POWER_POLICY 컴퓨터의 각 전원 구성표에 고유한 컴퓨터 전원 정책 설정을 포함합니다.
MACHINE_PROCESSOR_POWER_POLICY 시스템이 AC 전원 또는 배터리 전원에서 실행되는 동안 적용되는 프로세서 전원 정책 설정을 포함합니다.
MEM_ADDRESS_REQUIREMENTS 가상 메모리를 관리하는 함수에 대한 확장 매개 변수의 일부로 가장 낮고 가장 높은 기본 주소 및 맞춤을 지정합니다.
MEM_EXTENDED_PARAMETER 가상 메모리를 관리하는 함수에 대한 확장 매개 변수를 나타냅니다.
MEMORY_BASIC_INFORMATION 프로세스의 가상 주소 공간에 있는 페이지 범위에 대한 정보를 포함합니다.
MEMORYSTATUS 실제 메모리와 가상 메모리의 현재 상태에 대한 정보를 포함합니다.
MEMORYSTATUSEX 확장 메모리를 포함하여 물리적 메모리와 가상 메모리의 현재 상태에 대한 정보를 포함합니다.
MODEMDEVCAPS 모뎀의 기능에 대한 정보를 포함합니다.

MODEMSETTINGS 모뎀의 구성에 대한 정보를 포함합니다.
POWER_ACTION_POLICY 시스템 전원 상태를 설정하는 데 사용되는 정보를 포함합니다.
POWER_POLICY 각 전원 구성표에 고유한 전원 정책 설정을 포함합니다.
POWERBROADCAST_SETTING 전원 설정 이벤트와 함께 전송되며 특정 변경 내용에 대한 데이터를 포함합니다.
PREVENT_MEDIA_REMOVAL 이동식 미디어 잠금 데이터를 제공합니다. IOCTL_STORAGE_MEDIA_REMOVAL 제어 코드에서 사용됩니다.
PROCESS_HEAP_ENTRY 힙 요소에 대한 정보를 포함합니다.
PROCESSOR_POWER_POLICY 프로세서 성능 제어 및 C 상태에 대한 정보를 포함합니다.
PROCESSOR_POWER_POLICY_INFO 프로세서 C 상태 정책 설정에 대한 정보를 포함합니다.
PWM_CONTROLLER_INFO PWM(Pulse Width Modulation) 컨트롤러의 특징을 나타내는 정적 정보를 나타냅니다.
PWM_CONTROLLER_SET_DESIRED_PERIOD_INPUT PWM(Pulse Width Modulation) 컨트롤러에 대해 제안된 신호 기간에 대한 입력 값을 포함합니다.
PWM_CONTROLLER_SET_DESIRED_PERIOD_OUTPUT PWM(Pulse Width Modulation) 컨트롤러의 유효 출력 신호 기간을 포함합니다.
PWM_PIN_SET_POLARITY_INPUT 핀 또는 채널의 극성에 대해 원하는 값을 포함합니다.

READ_ELEMENT_ADDRESS_INFO 볼륨 태그 정보를 나타냅니다. IOCTL_CHANGER_QUERY_VOLUME_TAGS 제어 코드에서 사용됩니다.
REASON_CONTEXT 전원 요청에 대한 정보를 포함합니다. 이 구조체는 PowerCreateRequest 및 SetWaitableTimerEx 함수에서 사용됩니다.
STORAGE_DEVICE_NUMBER 디바이스에 대한 정보를 포함합니다. 이 구조체는 IOCTL_STORAGE_GET_DEVICE_NUMBER 제어 코드에서 사용됩니다.
STORAGE_HOTPLUG_INFO 디바이스의 핫플러그 정보에 대한 정보를 제공합니다.
STORAGE_OFFLOAD_READ_OUTPUT IOCTL_STORAGE_MANAGE_DATA_SET_ATTRIBUTES 제어 코드의 DeviceDsmAction_OffloadRead 동작에 대한 출력 구조입니다.
STORAGE_OFFLOAD_TOKEN 읽기 및 쓰기 작업을 오프로드하는 데 사용되는 파일의 일부를 나타내는 데 사용되는 토큰입니다.
STORAGE_OFFLOAD_WRITE_OUTPUT IOCTL_STORAGE_MANAGE_DATA_SET_ATTRIBUTES 제어 코드의 DeviceDsmAction_OffloadWrite 작업에 대한 출력 구조입니다.
SYSTEM_BATTERY_STATE 시스템 배터리의 현재 상태에 대한 정보를 포함합니다.
SYSTEM_POWER_CAPABILITIES 시스템의 전원 기능에 대한 정보를 포함합니다.
SYSTEM_POWER_LEVEL 시스템 배터리 드레이닝 정책 설정에 대한 정보를 포함합니다.
SYSTEM_POWER_POLICY 현재 시스템 전원 정책에 대한 정보를 포함합니다.

SYSTEM_POWER_STATUS 시스템의 전원 상태에 대한 정보를 포함합니다.
THERMAL_EVENT 열 이벤트를 포함합니다.
USER_POWER_POLICY 사용자의 각 전원 구성표에 고유한 전원 정책 설정을 포함합니다.
VBS_ENCLAVE_REPORT EnclaveGetAttestationReport 함수를 호출하여 생성된 보고서에 포함된 서명된 문의 형식을 설명합니다.
VBS_ENCLAVE_REPORT_MODULE enclave에 대해 로드된 모듈에 대해 설명합니다.
VBS_ENCLAVE_REPORT_PKG_HEADER EnclaveGetAttestationReport 함수를 호출하여 생성된 보고서의 내용을 설명합니다.
VBS_ENCLAVE_REPORT_VARDATA_HEADER EnclaveGetAttestationReport 함수가 생성하는 보고서에 포함된 변수 데이터 블록의 형식을 설명합니다.
VDS_ASYNC_OUTPUT VDS_ASYNC_OUTPUT 구조체(vds.h)는 비동기 개체의 출력을 정의합니다. 출력 요소는 작업 유형에 따라 달라집니다.
VDS_ASYNC_OUTPUT VDS_ASYNC_OUTPUT 구조체(vdshwprv.h)는 비동기 개체의 출력을 정의합니다. 출력 요소는 작업 유형에 따라 달라집니다.
VDS_CONTROLLER_NOTIFICATION VDS_CONTROLLER_NOTIFICATION 구조체(vds.h)는 컨트롤러 이벤트의 세부 정보를 정의합니다.
VDS_CONTROLLER_NOTIFICATION VDS_CONTROLLER_NOTIFICATION 구조체(vdshwprv.h)는 컨트롤러 이벤트의 세부 정보를 정의합니다.

VDS_CONTROLLER_PROP VDS_CONTROLLER_PROP 구조체(vds.h)는 컨트롤러 개체의 속성을 정의합니다.
VDS_CONTROLLER_PROP VDS_CONTROLLER_PROP 구조체(vdshwprv.h)는 컨트롤러 개체의 속성을 정의합니다.
VDS_CREATE_VDISK_PARAMETERS 가상 디스크를 만들 때 사용할 매개 변수를 포함합니다.
VDS_DISK_EXTENT 디스크 익스텐트 속성을 정의합니다.
VDS_DISK_FREE_EXTENT 디스크의 사용 가능한 익스텐트 설명
VDS_DISK_NOTIFICATION VDS_DISK_NOTIFICATION 구조체(vds.h)는 디스크 이벤트의 세부 정보를 정의합니다.
VDS_DISK_NOTIFICATION VDS_DISK_NOTIFICATION 구조체(vdshwprv.h)는 디스크 이벤트의 세부 정보를 정의합니다.
VDS_DISK_PROP 디스크 개체의 속성을 정의합니다.
VDS_DISK_PROP2 디스크 개체의 속성을 정의합니다. 이 구조체는 위치 경로도 포함하고 디스크가 오프라인인 경우 오프라인인 이유를 제외하고 VDS_DISK_PROP 구조와 동일합니다.
VDS_DRIVE_EXTENT VDS_DRIVE_EXTENT 구조체(vds.h)는 드라이브 익스텐트 속성을 정의합니다.
VDS_DRIVE_EXTENT VDS_DRIVE_EXTENT 구조체(vdshwprv.h)는 드라이브 익스텐트의 속성을 정의합니다.
VDS_DRIVE_LETTER_NOTIFICATION VDS_DRIVE_LETTER_NOTIFICATION 구조체(vds.h)는 드라이브 문자 이벤트의 세부 정보를 정의합니다.

VDS_DRIVE_LETTER_NOTIFICATION VDS_DRIVE_LETTER_NOTIFICATION 구조체(vdshwprv.h)는 드라이브 문자 이벤트의 세부 정보를 정의합니다.
VDS_DRIVE_LETTER_PROP 드라이브 문자의 속성을 정의합니다.
VDS_DRIVE_NOTIFICATION VDS_DRIVE_NOTIFICATION 구조체(vds.h)는 드라이브 이벤트의 세부 정보를 정의합니다.
VDS_DRIVE_NOTIFICATION VDS_DRIVE_NOTIFICATION 구조체(vdshwprv.h)는 드라이브 이벤트의 세부 정보를 정의합니다.
VDS_DRIVE_PROP VDS_DRIVE_PROP 구조체(vds.h)는 드라이브 개체의 속성을 정의합니다.
VDS_DRIVE_PROP VDS_DRIVE_PROP 구조체(vdshwprv.h)는 드라이브 개체의 속성을 정의합니다.
VDS_DRIVE_PROP2 VDS_DRIVE_PROP2 구조체(vds.h)는 드라이브 개체의 속성을 정의합니다.
VDS_DRIVE_PROP2 VDS_DRIVE_PROP2 구조체(vdshwprv.h)는 드라이브 개체의 속성을 정의합니다.
VDS_FILE_SYSTEM_FORMAT_SUPPORT_PROP 볼륨 서식 지정에 지원되는 파일 시스템에 대한 정보를 제공합니다.
VDS_FILE_SYSTEM_NOTIFICATION VDS_FILE_SYSTEM_NOTIFICATION 구조체(vds.h)는 파일 시스템 이벤트의 세부 정보를 정의합니다.
VDS_FILE_SYSTEM_NOTIFICATION VDS_FILE_SYSTEM_NOTIFICATION 구조체(vdshwprv.h)는 파일 시스템 이벤트의 세부 정보를 정의합니다.
VDS_FILE_SYSTEM_PROP 파일 시스템의 속성을 정의합니다.

VDS_FILE_SYSTEM_TYPE_PROP 파일 시스템 형식의 속성을 정의합니다.
VDS_HBAPORT_PROP VDS_HBAPORT_PROP 구조체(vds.h)는 HBA 포트의 속성을 정의합니다.
VDS_HBAPORT_PROP VDS_HBAPORT_PROP 구조체(vdshwprv.h)는 HBA 포트의 속성을 정의합니다.
VDS_HINTS VDS_HINTS 구조체(vds.h)는 LUN 또는 LUN 플렉스에 대한 자동 매핑 힌트를 정의합니다.
VDS_HINTS VDS_HINTS 구조체(vdshwprv.h)는 LUN 또는 LUN 플렉스에 대한 자동 매핑 힌트를 정의합니다.
VDS_HINTS2 VDS_HINTS2 구조체(vds.h)에는 스토리지 풀의 LUN에 대한 자동 매핑 힌트가 포함되어 있습니다.
VDS_HINTS2 VDS_HINTS2 구조체(vdshwprv.h)에는 스토리지 풀의 LUN에 대한 자동 매핑 힌트가 포함되어 있습니다.
VDS_INPUT_DISK 입력 디스크의 세부 정보를 정의합니다.
VDS_INTERCONNECT 물리적 상호 연결의 주소 데이터를 정의합니다.
VDS_IPADDRESS VDS_IPADDRESS 구조체(vds.h)는 IP 주소와 포트를 정의합니다.
VDS_IPADDRESS VDS_IPADDRESS 구조체(vdshwprv.h)는 IP 주소와 포트를 정의합니다.
VDS_ISCSI_INITIATOR_ADAPTER_PROP VDS_ISCSI_INITIATOR_ADAPTER_PROP 구조체(vds.h)는 iSCSI 초기자 어댑터의 속성을 정의합니다.

VDS_ISCSI_INITIATOR_ADAPTER_PROP VDS_ISCSI_INITIATOR_ADAPTER_PROP 구조체(vdshwprv.h)는 iSCSI 초기자 어댑터의 속성을 정의합니다.
VDS_ISCSI_INITIATOR_PORTAL_PROP VDS_ISCSI_INITIATOR_PORTAL_PROP 구조체(vds.h)는 iSCSI 초기자 포털의 속성을 정의합니다.
VDS_ISCSI_INITIATOR_PORTAL_PROP VDS_ISCSI_INITIATOR_PORTAL_PROP 구조체(vdshwprv.h)는 iSCSI 초기자 포털의 속성을 정의합니다.
VDS_ISCSI_IPSEC_KEY VDS_ISCSI_IPSEC_KEY 구조체(vds.h)는 지원되지 않습니다. 이 구조체는 나중에 사용할 수 있습니다.
VDS_ISCSI_IPSEC_KEY VDS_ISCSI_IPSEC_KEY 구조체(vdshwprv.h)는 지원되지 않으며 나중에 사용하도록 예약되어 있습니다.
VDS_ISCSI_PORTAL_PROP VDS_ISCSI_PORTAL_PROP 구조체(vds.h)는 iSCSI 포털의 속성을 정의합니다. (VDS_ISCSI_PORTAL_PROP)
VDS_ISCSI_PORTAL_PROP VDS_ISCSI_PORTAL_PROP 구조체(vdshwprv.h)는 iSCSI 포털의 속성을 정의합니다.
VDS_ISCSI_PORTALGROUP_PROP VDS_ISCSI_PORTALGROUP_PROP 구조체(vds.h)는 iSCSI 포털 그룹의 속성을 정의합니다.
VDS_ISCSI_PORTALGROUP_PROP VDS_ISCSI_PORTALGROUP_PROP 구조체(vdshwprv.h)는 iSCSI 포털 그룹의 속성을 정의합니다.
VDS_ISCSI_SHARED_SECRET VDS_ISCSI_SHARED_SECRET 구조체(vds.h)는 CHAP 공유 비밀을 정의합니다.
VDS_ISCSI_SHARED_SECRET VDS_ISCSI_SHARED_SECRET 구조체(vdshwprv.h)는 CHAP 공유 비밀을 정의합니다.

VDS_ISCSI_TARGET_PROP VDS_ISCSI_TARGET_PROP 구조체(vds.h)는 iSCSI 대상의 속성을 정의합니다.
VDS_ISCSI_TARGET_PROP VDS_ISCSI_TARGET_PROP 구조체(vdshwprv.h)는 iSCSI 대상의 속성을 정의합니다.
VDS_LUN_INFORMATION LUN 또는 디스크에 대한 정보를 정의합니다. 애플리케이션은 이 구조를 사용하여 항상 LUN을 고유하게 식별할 수 있습니다.
VDS_LUN_NOTIFICATION VDS_LUN_NOTIFICATION 구조체(vds.h)는 LUN 알림의 세부 정보를 정의합니다.
VDS_LUN_NOTIFICATION VDS_LUN_NOTIFICATION 구조체(vdshwprv.h)는 LUN 알림의 세부 정보를 정의합니다.
VDS_LUN_PLEX_PROP VDS_LUN_PLEX_PROP 구조체(vds.h)는 LUN 플렉스 개체의 속성을 정의합니다.
VDS_LUN_PLEX_PROP VDS_LUN_PLEX_PROP 구조체(vdshwprv.h)는 LUN 플렉스 개체의 속성을 정의합니다.
VDS_LUN_PROP VDS_LUN_PROP 구조체(vds.h)는 LUN 개체의 속성을 정의합니다.
VDS_LUN_PROP VDS_LUN_PROP 구조체(vdshwprv.h)는 LUN 개체의 속성을 정의합니다.
VDS_MOUNT_POINT_NOTIFICATION VDS_MOUNT_POINT_NOTIFICATION 구조체(vds.h)는 드라이브 문자 또는 볼륨 GUID 경로가 변경되어 소프트웨어 공급자가 반환한 알림 정보를 나타냅니다.
VDS_MOUNT_POINT_NOTIFICATION VDS_MOUNT_POINT_NOTIFICATION 구조체(vdshwprv.h)는 드라이브 문자 또는 볼륨 GUID 경로가 변경되어 소프트웨어 공급자가 반환한 데이터를 나타냅니다.

VDS_NOTIFICATION VDS_NOTIFICATION 구조(vds.h)는 각 알림 대상 유형과 관련된 VDS 알림 구조를 정의합니다.
VDS_NOTIFICATION VDS_NOTIFICATION 구조체(vdshwprv.h)는 각 알림 대상 유형(제목)과 관련된 VDS 알림 구조를 정의합니다.
VDS_PACK_NOTIFICATION VDS_PACK_NOTIFICATION 구조체(vds.h)는 팩 이벤트의 세부 정보를 정의합니다.
VDS_PACK_NOTIFICATION VDS_PACK_NOTIFICATION 구조체(vdshwprv.h)는 팩 이벤트의 세부 정보를 정의합니다.
VDS_PACK_PROP pack 개체의 속성을 정의합니다.
VDS_PARTITION_INFO_GPT GPT(GUID 파티션 테이블) 파티션의 세부 정보를 정의합니다.
VDS_PARTITION_INFO_MBR MBR(마스터 부팅 레코드) 파티션의 세부 정보를 정의합니다.
VDS_PARTITION_INFORMATION_EX 이 구조체는 시스템 사용을 위해 예약되어 있습니다. (VDS_PARTITION_INFORMATION_EX)
VDS_PARTITION_NOTIFICATION VDS_PARTITION_NOTIFICATION 구조체(vds.h)는 파티션 이벤트의 세부 정보를 정의합니다.
VDS_PARTITION_NOTIFICATION VDS_PARTITION_NOTIFICATION 구조체(vdshwprv.h)는 파티션 이벤트의 세부 정보를 정의합니다.
VDS_PARTITION_PROP 파티션의 속성을 정의합니다.
VDS_PATH_ID VDS_PATH_ID 구조체(vds.h)는 경로에 대한 고유 ID를 정의합니다.

VDS_PATH_ID VDS_PATH_ID 구조체(vdshwprv.h)는 경로에 대한 고유 ID를 정의합니다.
VDS_PATH_INFO VDS_PATH_INFO 구조체(vds.h)는 LUN 경로에 대한 정보를 정의합니다.
VDS_PATH_INFO VDS_PATH_INFO 구조체(vdshwprv.h)는 LUN 경로에 대한 정보를 정의하고 IVdsLunMpio::GetPathInfo 메서드의 ppPaths 매개 변수에 반환됩니다.
VDS_PATH_POLICY VDS_PATH_POLICY 구조체(vds.h)는 특정 경로에 적용되는 부하 분산 정책을 정의합니다.
VDS_PATH_POLICY VDS_PATH_POLICY 구조체(vdshwprv.h)는 특정 경로에 적용되는 부하 분산 정책을 정의합니다.
VDS_POOL_ATTRIBUTES VDS_POOL_ATTRIBUTES 구조체(vds.h)는 스토리지 풀의 특성을 정의합니다.
VDS_POOL_ATTRIBUTES VDS_POOL_ATTRIBUTES 구조체(vdshwprv.h)는 스토리지 풀의 특성을 정의합니다.
VDS_POOL_CUSTOM_ATTRIBUTES VDS_POOL_CUSTOM_ATTRIBUTES 구조체(vds.h)는 스토리지 풀의 사용자 지정 특성을 정의합니다.
VDS_POOL_CUSTOM_ATTRIBUTES VDS_POOL_CUSTOM_ATTRIBUTES 구조체(vdshwprv.h)는 스토리지 풀의 사용자 지정 특성을 정의합니다.
VDS_PORT_NOTIFICATION VDS_PORT_NOTIFICATION 구조체(vds.h)는 컨트롤러 포트 이벤트의 세부 정보를 정의합니다.
VDS_PORT_NOTIFICATION VDS_PORT_NOTIFICATION 구조체(vdshwprv.h)는 컨트롤러 포트 이벤트의 세부 정보를 정의합니다.
VDS_PORT_PROP VDS_PORT_PROP 구조체(vds.h)는 컨트롤러 개체의 포트 속성을 정의합니다.

VDS_PORT_PROP VDS_PORT_PROP 구조체(vdshwprv.h)는 컨트롤러 개체의 포트 속성을 정의합니다.
VDS_PORTAL_GROUP_NOTIFICATION VDS_PORTAL_GROUP_NOTIFICATION 구조체(vds.h)는 iSCSI 포털 이벤트의 세부 정보를 정의합니다.
VDS_PORTAL_GROUP_NOTIFICATION VDS_PORTAL_GROUP_NOTIFICATION 구조체(vdshwprv.h)는 iSCSI 포털 이벤트의 세부 정보를 정의합니다.
VDS_PORTAL_NOTIFICATION VDS_PORTAL_NOTIFICATION 구조체(vds.h)는 iSCSI 포털 이벤트의 세부 정보를 정의합니다.
VDS_PORTAL_NOTIFICATION VDS_PORTAL_NOTIFICATION 구조체(vdshwprv.h)는 iSCSI 포털 이벤트의 세부 정보를 정의합니다.
VDS_PROVIDER_PROP VDS_PROVIDER_PROP 구조체(vds.h)는 공급자 개체의 속성을 정의합니다.
VDS_PROVIDER_PROP VDS_PROVIDER_PROP 구조체(vdshwprv.h)는 공급자 개체의 속성을 정의합니다.
VDS_REPARSE_POINT_PROP 볼륨 개체의 재분석 지점 속성을 정의합니다.
VDS_SERVICE_PROP 서비스 개체의 속성을 정의합니다.
VDS_STORAGE_DEVICE_ID_DESCRIPTOR 스토리지 디바이스에 대해 하나 이상의 스토리지 식별자를 정의합니다(일반적으로 디바이스의 클래스가 아닌 인스턴스).
VDS_STORAGE_IDENTIFIER 특정 코드 집합 및 형식을 사용하여 스토리지 디바이스를 정의합니다.

VDS_STORAGE_POOL_DRIVE_EXTENT VDS_STORAGE_POOL_DRIVE_EXTENT 구조체(vds.h)는 스토리지 풀에서 사용할 수 있는 드라이브 익스텐트 를 정의합니다.
VDS_STORAGE_POOL_DRIVE_EXTENT VDS_STORAGE_POOL_DRIVE_EXTENT 구조체(vdshwprv.h)는 스토리지 풀에서 사용할 수 있는 드라이브 범위를 정의합니다.
VDS_STORAGE_POOL_PROP VDS_STORAGE_POOL_PROP 구조체(vds.h)는 스토리지 풀 개체의 속성을 정의합니다.
VDS_STORAGE_POOL_PROP VDS_STORAGE_POOL_PROP 구조체(vdshwprv.h)는 스토리지 풀 개체의 속성을 정의합니다.
VDS_SUB_SYSTEM_NOTIFICATION VDS_SUB_SYSTEM_NOTIFICATION 구조체(vds.h)는 하위 시스템 이벤트의 세부 정보를 정의합니다.
VDS_SUB_SYSTEM_NOTIFICATION VDS_SUB_SYSTEM_NOTIFICATION 구조체(vdshwprv.h)는 하위 시스템 이벤트의 세부 정보를 정의합니다.
VDS_SUB_SYSTEM_PROP VDS_SUB_SYSTEM_PROP 구조체(vds.h)는 하위 시스템 개체의 속성을 정의합니다.
VDS_SUB_SYSTEM_PROP VDS_SUB_SYSTEM_PROP 구조체(vdshwprv.h)는 하위 시스템 개체의 속성을 정의합니다.
VDS_SUB_SYSTEM_PROP2 VDS_SUB_SYSTEM_PROP2 구조체(vds.h)는 하위 시스템 개체의 속성을 정의합니다.
VDS_SUB_SYSTEM_PROP2 VDS_SUB_SYSTEM_PROP2 구조체(vdshwprv.h)는 하위 시스템 개체의 속성을 정의합니다.
VDS_TARGET_NOTIFICATION VDS_TARGET_NOTIFICATION 구조체(vds.h)는 iSCSI 대상 이벤트의 세부 정보를 정의합니다.

VDS_TARGET_NOTIFICATION VDS_TARGET_NOTIFICATION 구조체(vdshwprv.h)는 iSCSI 대상 이벤트의 세부 정보를 정의합니다.
VDS_VDISK_PROPERTIES 가상 디스크의 속성을 정의합니다.
VDS_VOLUME_NOTIFICATION VDS_VOLUME_NOTIFICATION 구조체(vds.h)는 볼륨 이벤트의 세부 정보를 정의합니다.
VDS_VOLUME_NOTIFICATION VDS_VOLUME_NOTIFICATION 구조체(vdshwprv.h)는 볼륨 이벤트의 세부 정보를 정의합니다.
VDS_VOLUME_PLEX_PROP 볼륨 플렉스 개체의 속성을 정의합니다.
VDS_VOLUME_PROP 볼륨 개체의 속성을 정의합니다.
VDS_VOLUME_PROP2 볼륨 개체의 속성을 정의합니다. 이 구조체는 볼륨 GUID도 포함한다는 점을 제외하고 VDS_VOLUME_PROP 구조체와 동일합니다.
VDS_WWN VDS_WWN 구조체(vds.h)는 WWN(World-Wide Name)을 정의합니다. 이 구조체는 ANSI HBA API에서 정의한 HBA_WWN 구조체에 해당합니다.
VDS_WWN VDS_WWN 구조체(vdshwprv.h)는 WWN(월드 와이드 이름)을 정의합니다. 이 구조체는 ANSI HBA API에서 정의한 HBA_WWN 구조체에 해당합니다.
VSS_COMPONENTINFO 지정된 구성 요소에 대한 정보를 포함합니다.
VSS_DIFF_AREA_PROP 원래 파일 데이터를 포함하는 볼륨과 섀도 복사본 스토리지 영역을 포함하는 볼륨 간의 연결을 설명합니다.

VSS_DIFF_VOLUME_PROP 섀도 복사본 스토리지 영역 볼륨에 대해 설명합니다.
VSS_MGMT_OBJECT_PROP 볼륨, 섀도 복사본 스토리지 볼륨 또는 섀도 복사본 스토리지 영역의 속성을 정의합니다.
VSS_OBJECT_PROP 공급자, 볼륨, 섀도 복사본 또는 섀도 복사본 집합의 속성을 정의합니다.
VSS_PROVIDER_PROP 섀도 복사본 공급자 속성을 지정합니다.
VSS_SNAPSHOT_PROP 섀도 복사본 또는 섀도 복사본 집합의 속성을 포함합니다.
VSS_VOLUME_PROP 섀도 복사본 원본 볼륨의 속성을 포함합니다.
VSS_VOLUME_PROTECTION_INFO 볼륨의 섀도 복사본 보호 수준에 대한 정보를 포함합니다.
WIN32_MEMORY_RANGE_ENTRY 메모리 범위를 지정합니다.
WIN32_MEMORY_REGION_INFORMATION 메모리 영역에 대한 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Tablet PC

아티클2023. 03. 11.

태블릿 PC 기술 개요입니다.

태블릿 PC를 개발하려면 다음 헤더가 필요합니다.

- inked.h
- micaut.h
- msinkaut.h
- msinkaut15.h
- peninputpanel.h
- recapis.h
- rectypes.h
- rtscom.h
- tabflicks.h
- tpcshrd.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 모바일 PC
- Tablet PC

열거형

ALT_BREAKS 최상의 결과 문자열에서 대체를 만드는 방법을 지정합니다.
AppearanceConstants InkEdit 컨트롤이 화면에 표시되는 방식을 지정합니다.
BorderStyleConstants InkEdit 컨트롤의 테두리가 화면에 표시되는 방식을 지정합니다.
CONFIDENCE_LEVEL 인식기에서 인식 결과에 있는 신뢰도 수준을 나타냅니다.

CorrectionMode 태블릿 PC 입력 패널의 수정 모드를 지정합니다.
수정 처리 삽입 후 보정 빗이 표시하는 입력 패널을 기준으로 방향을 지정합니다.
DISPID_InkEdit 다양한 잉크 편집기 상태를 지정합니다.
DISPID_InkEditEvents 다양한 잉크 편집기 이벤트를 지정합니다.
EventMask 주의할 수 있는 ITextInputPanel 인터페이스의 이벤트입니다.
FLICKACTION_COMMANDCODE 펜 플릭에 할당할 수 있는 가능한 영화 동작을 정의합니다.
FLICKDIRECTION 펜 플릭이 발생한 방향을 정의합니다.
FLICKMODE Flick 제스처가 사용되는 위치를 설명합니다.
InkApplicationGesture 애플리케이션별 제스처 집합에 대한 관심을 설정하는 값을 정의합니다. 애플리케이션 제스처는 애플리케이션을 지원하도록 선택할 수 있는 제스처입니다.
InkBoundingBoxMode 잉크의 경계 상자를 계산하는 데 사용되는 그리기 특성과 같은 스트로크의 특성을 지정합니다. 경계 상자는 InkDisp 개체의 모든 점을 포함하는 가장 작은 사각형입니다.
InkClipboardFormats 클립보드에 저장된 잉크의 형식을 지정합니다.
InkClipboardModes 클립보드의 복사 옵션을 지정합니다.

InkCollectionMode 잉크, 제스처 또는 잉크 및 제스처가 사용자가 쓸 때 인식되는지 여부를 결정하는 값을 정의합니다.
InkCollectorEventInterest 잉크 수집기에서 이벤트가 발생했는지 여부와 발생한 이벤트를 지정하는 데 사용되는 값을 정의합니다.
InkCursorButtonState 커서 단추의 상태를 지정하는 값을 정의합니다.
InkDisplayMode 컨트롤에 선택 영역이 표시되는 방식을 지정합니다.
InkDivisionType IInkDivisionResult 개체 내의 구조 형식에 대한 값을 정의합니다.
InkEditStatus InkEdit 컨트롤이 유휴 상태인지, 잉크를 수집하거나, 잉크를 인식하는지를 지정합니다.
InkExtractFlags InkDisp 개체에서 스트로크를 제거하는 방법을 결정합니다.
InkInsertMode InkEdit 컨트롤에 잉크를 삽입하는 방법을 지정합니다.
InkMode 잉크 컬렉션이 비활성화되어 있는지, 잉크가 수집되는지, 잉크 및 제스처가 수집되는지 여부에 관계없이 그리기 잉크에 대한 컬렉션 모드를 지정합니다.
InkMouseButton 누른 마우스 단추를 지정합니다. (InkMouseButton)
InkMousePointer 표시할 마우스 포인터의 유형을 지정합니다.
InkOverlayAttachMode 새 InkOverlay 개체를 현재 레이어 뒤 또는 앞에 연결할 위치를 지정합니다.

InkOverlayEditingMode InkOverlay 개체 및 InkPicture 컨트롤의 동작 모드를 지정합니다.
InkOverlayEraserMode InkOverlay 개체 및 InkPicture 컨트롤에서 잉크를 지우는 방법을 지정합니다. 이 모드는 InkOverlayEditingMode가 Delete로 설정된 경우에 사용됩니다.
InkPenTip 펜 끝 모양을 지정합니다.
InkPersistenceCompressionMode InkDisp 개체를 직렬화된 형식으로 저장하는 데 사용되는 압축 모드의 값을 정의합니다.
InkPersistenceFormat 잉크를 유지하는 방법을 지정합니다.
InkPictureSizeMode 그림이 InkPicture 컨트롤 내에서 동작하는 방식을 지정합니다.
InkRasterOperation 그려진 잉크에서 래스터 작업을 수행하기 위한 값을 정의합니다.
InkRecognitionAlternatesSelection 구현되지 않았습니다. (InkRecognitionAlternatesSelection)
InkRecognitionConfidence 인식기에서 인식 결과에 있는 신뢰도 수준을 나타냅니다.
InkRecognitionModes 인식기가 잉크를 해석하고 결과 문자열을 결정하는 방법을 지정합니다.
InkRecognitionStatus 인식 중에 오류가 발생했는지 여부와 오류가 발생한 경우 를 지정합니다.
InkRecognizerCapabilities 인식기의 특성을 지정합니다. 또한 이 열거형을 사용하여 설치된 인식기를 검색할 때 사용할 특성을 결정합니다.

InkRecognizerCharacterAutoCompletionMode 문자 입력 모드 유형을 지정합니다.
InkSelectionConstants 점 또는 패킷 값 그룹 내의 첫 번째 요소 또는 모든 요소를 사용할지 여부를 지정합니다.
InkShiftKeyModifierFlags 누른 한정자 키를 지정합니다.
InkSystemGesture 운영 체제별 제스처 집합에 대한 관심을 지정합니다.
InPlaceDirection 텍스트 입력 필드를 기준으로 In-Place 입력 패널의 기본 방향을 지정합니다.
InPlaceState 태블릿 PC 입력 패널의 In-Place 상태 값을 지정합니다.
InteractionMode 태블릿 PC 입력 패널에 대해 사용자가 선택할 수 있는 상호 작용 모드를 지정합니다.
KEYMODIFIER 플릭 제스처가 발생했을 때 누른 한정자 키를 결정합니다.
LINE_METRICS 인식 세그먼트에 있는 줄을 나타냅니다.
MICUIELEMENT MIC(수학 입력 컨트롤)의 UI(사용자 인터페이스) 요소를 지정합니다.
MICUIELEMENTSTATE MIC(수학 입력 컨트롤)의 단추 상태를 지정합니다.
MouseButton 누른 마우스 단추를 지정합니다. (MouseButton)

PanelInputArea 태블릿 PC 입력 패널의 기본 입력 영역을 나타내는 값을 지정합니다.
PanelType PenInputPanel 개체에서 현재 사용할 수 있는 입력 유형을 정의합니다.
PROPERTY_UNITS 속성의 측정 단위에 대한 상수 값을 정의합니다.
RealTimeStylusDataInterest 플러그 인이 수신하는 이벤트 알림을 지정하기 위해 플러그 인에서 사용하는 값을 정의합니다.
RealTimeStylusLockType RealTimeStylus Class 개체의 멤버 및 속성을 수정하지 않도록 보호하는 RealTimeStylus Class 개체 내의 잠금을 지정합니다.
ScrollBarsConstants InkEdit 컨트롤의 스크롤 막대가 화면에 표시되는 방식을 지정합니다.
SCROLLDIRECTION 펜 플릭에 대한 스크롤 명령의 방향을 정의합니다.
SelAlignmentConstants InkEdit 컨트롤의 여백을 기준으로 단락의 맞춤을 지정합니다.
SelectionHitResult 적중 테스트 중에 적중된 선택 영역(있는 경우)을 지정합니다.
StylusQueue 스타일러스 데이터가 추가되는 큐를 지정합니다.
TabletHardwareCapabilities 태블릿 PC의 하드웨어 기능을 지정합니다.
TabletPropertyMetricUnit 속성의 측정 단위를 나타냅니다.

Functions

_IInkEditEvents_AddRef IInkEditEvents 인터페이스 개체의 참조 수를 1씩 늘입니다. 이 메서드는 IInkEditEvents에서 상속하는 **IUnknown** 인터페이스의 일부입니다.
_IInkEditEvents_GetIDsOfNames 이름 집합을 해당 DISPID 집합에 매핑합니다.
_IInkEditEvents_GetTypeInfo 개체에 대한 형식 정보를 검색한 다음 인터페이스의 형식 정보를 가져오는 데 사용할 수 있습니다.
_IInkEditEvents_GetTypeInfoCount 개체가 제공하는 형식 정보 인터페이스의 수를 검색합니다.
_IInkEditEvents_Invoke 개체에서 노출하는 속성 및 메서드에 대한 액세스를 제공합니다.
_IInkEditEvents_QueryInterface IInkEditEvents 개체의 지정된 COM 인터페이스에 대한 쿼리입니다.
_IInkEditEvents_Release IInkEditEvents 개체에 대한 참조를 해제합니다.
AcquireLock 지정된 잠금을 검색합니다.
추가 IInkCustomStrokes 컬렉션에 InkStrokes 컬렉션을 추가합니다.
추가 IInkExtendedProperty 개체를 만들고 IInkExtendedProperties 컬렉션에 추가합니다.
추가 기존 InkStrokes 컬렉션에 IInkStrokeDisp 개체 또는 InkStrokes 컬렉션을 추가합니다.

AddCustomStylusDataToQueue RealTimeStylus 클래스 개체의 지정된 큐에 사용자 지정 데이터를 추가합니다.
AddFunctionName 인식기가 허용하는 사용자 지정 수학 함수 목록에 새 함수 이름 정의를 추가합니다.
AddStroke RecognizerContext에 잉크 스트로크를 추가합니다.
AddStrokes 기존 Strokes 컬렉션에 Strokes 컬렉션을 추가합니다.
AddStrokesAtRectangle 지정된 사각형의 이 InkDisp 개체에 지정된 Strokes 컬렉션을 추가합니다.
AddStylusAsyncPlugin 지정된 인덱스의 비동기 플러그 인 컬렉션에 IStylusAsyncPlugin을 추가합니다.
AddStylusSyncPlugin 지정된 인덱스의 동기 플러그 인 컬렉션에 IStylusSyncPlugin을 추가합니다.
AddWord InkWordList 개체에 한 단어를 추가합니다.
AddWords 단일 작업에서 InkWordList에 둘 이상의 단어를 추가합니다.
AddWordsToWordList 단어 목록에 하나 이상의 단어를 추가합니다.
조언 태블릿 PC 입력 패널과 지정된 싱크 개체 간에 권고 연결을 설정합니다.
AdviseInkChange 스트로크가 추가되거나 삭제되었기 때문에 인식기가 잉크 처리를 중지합니다.

AlternatesWithConstantPropertyValues 이 메서드가 호출되는 IInkRecognitionAlternate 개체의 분할인 IInkRecognitionAlternates 컬렉션을 검색합니다.
AppendPackets 디지타이저 입력 패킷 목록의 끝에 패킷을 추가합니다.
BackgroundRecognize IInkRecognizer 개체가 연결된 스트로크 컬렉션을 인식하고 인식이 완료되면 Recognition 이벤트를 발생하도록 합니다.
BackgroundRecognizeWithAlternates IInkRecognizer 개체가 연결된 스트로크 컬렉션을 인식하고 인식이 완료되면 RecognitionWithAlternates 이벤트를 발생하도록 합니다.
BeginStroke RealTimeStylus Class 개체의 패킷 데이터를 사용하여 잉크 개체에 대한 스트로크를 시작합니다.
CanPaste IDataObject를 InkDisp 개체로 변환할 수 있는지 여부를 나타냅니다.
지우기 IInkCustomStrokes 컬렉션에서 모든 InkStrokes 컬렉션을 지웁니다.
지우기 IInkExtendedProperties 컬렉션에서 모든 IInkExtendedProperty 개체를 지웁니다.
지우기 컨트롤에서 모든 잉크를 지웁니다.
ClearStylusQueues 데이터의 RealTimeStylus 클래스 입력 및 출력 큐를 지웁니다.
클립 직사각형 외부에 있는 IInkStrokeDisp 개체 또는 InkStrokes 컬렉션의 일부를 제거합니다. (IInkDisp.Clip)

클립 직사각형 외부에 있는 IInkStrokeDisp 개체 또는 InkStrokes 컬렉션의 일부를 제거합니다. (IInkStrokeDisp.Clip)
클립 직사각형 외부에 있는 IInkStrokeDisp 개체 또는 InkStrokes 컬렉션의 일부를 제거합니다. (IInkStrokes.Clip)
ClipboardCopy InkStrokes 컬렉션을 클립보드에 복사합니다.
ClipboardCopyWithRectangle 알려진 사각형 내에 포함된 IInkStrokeDisp 개체를 클립보드에 복사합니다.
ClipboardPaste 클립보드에서 InkDisp 개체로 IDataObject를 복사합니다.
복제 중복된 InkDisp 개체를 만듭니다.
복제 중복된 InkDrawingAttributes 개체를 만듭니다.
복제 중복된 InkRecognizerContext 개체를 만듭니다.
CloneContext 원본과 동일한 설정을 포함하는 인식기 컨텍스트를 만듭니다. 새 인식기 컨텍스트에는 원본의 잉크 또는 인식 결과가 포함되지 않습니다.
CommitPendingInput 더 이상 사용되지 않습니다. PenInputPanel이 TIP(텍스트 입력 패널)로 대체되었습니다. 수집된 잉크를 인식기로 보내고 인식 결과를 게시합니다.
CommitPendingInput 수집된 잉크를 인식기로 보내고 인식 결과를 게시합니다.

CorrectionModeChanged 태블릿 PC 입력 패널의 수정 빗이 모드를 변경하면 발생합니다.
CorrectionModeChanging 태블릿 PC 입력 패널의 수정 빗이 모드를 변경하려고 할 때 발생합니다.
CreateContext 인식기 컨텍스트를 만듭니다.
CreateRecognizer 인식기를 만듭니다.
CreateRecognizerContext 새 InkRecognizerContext 개체를 만듭니다.
CreateStroke 패킷 데이터 입력 값 배열에서 IInkStrokeDisp 개체를 만듭니다.
CreateStroke RealTimeStylus 클래스 개체에서 온 패킷 데이터를 사용하여 잉크 개체에 스트로크를 만듭니다.
CreateStrokes 기존 IInkStrokeDisp 개체에서 새 InkStrokes 컬렉션을 만듭니다.
CustomStylusDataAdded 구현 플러그 인에 사용자 지정 스타일러스 데이터를 사용할 수 있음을 알릴 수 있습니다.
DataInterest 플러그 인이 알림을 받을 이벤트를 검색합니다.
DeleteStroke InkDisp 개체에서 IInkStrokeDisp 개체를 삭제합니다.
DeleteStrokes InkDisp 개체의 Strokes 컬렉션에서 InkStrokes 컬렉션을 삭제합니다.

DestroyAlternate 이 함수는 사용되지 않으며 사용자 지정 애플리케이션 인식기에서 구현할 필요가 없습니다.
DestroyContext 인식기 컨텍스트를 삭제합니다.
DestroyRecognizer 인식기를 제거합니다.
DestroyWordList 현재 단어 목록을 삭제합니다.
나누기 InkDivider 개체의 스트로크 레이아웃 분석 결과를 포함하는 IInkDivisionResult 개체를 반환합니다.
DoesPropertyExist IInkExtendedProperty 개체가 IInkExtendedProperties 컬렉션 내에 있는지 여부를 나타내는 값을 검색합니다.
그리기 InkOverlay 개체 내에서 잉크를 다시 그릴 사각형을 설정합니다.
그리기 알려진 디바이스 컨텍스트를 사용하여 잉크 스트로크를 그립니다.
그리기 캐시된 데이터를 지정된 디바이스 컨텍스트에 그립니다.
DrawStroke 알려진 디바이스 컨텍스트를 사용하여 IInkStrokeDisp 개체를 그리고 필요에 따라 알려진 InkDrawingAttributes 개체를 사용하여 IInkStrokeDisp 개체를 그립니다.
EnableAutoGrow 입력이 컨트롤의 현재 범위를 벗어나면 컨트롤이 자동으로 증가하는지 여부를 결정합니다.
EnableExtendedButtons 확장된 컨트롤 단추 집합이 표시되는지 여부를 확인합니다.

EnableGestures GestureRecognizer 클래스 개체가 응답하는 애플리케이션 제스처를 나타내는 값을 설정합니다.
EnableTsf 더 이상 사용되지 않습니다. PenInputPanel 개체가 TSF(Text Services Framework)를 통해 연결된 컨트롤에 텍스트를 보내려고 하고 수정 사용자 인터페이스를 사용할 수 있는지 여부를 나타내는 부울 값을 가져오거나 설정합니다.
EndInkInput EndInkInput은 Windows Vista를 기준으로 서양 언어를 인식하는 데 더 이상 사용할 수 없습니다.
EndInkInput 컨텍스트에 더 이상 잉크가 추가되지 않음을 나타냅니다. 이 함수를 호출한 후에는 컨텍스트에 스트로크를 추가할 수 없습니다.
EndStroke 스트로크를 종료하고 스트로크 개체를 반환합니다.
오류 IStylusAsyncPlugin 또는 IStylusSyncPlugin 컬렉션에서 이 플러그 인 또는 이전 플러그 인 중 하나가 예외를 throw했음을 구현 개체에 알립니다.
ExtractStrokes 알려진 스트로크 컬렉션을 사용하여 추출할 스트로크를 결정하여 InkDisp 클래스에서 추출하고 새 InkDisp 클래스로 잘라내거나 복사할 스트로크를 지정합니다.
ExtractWithRectangle 알려진 사각형을 사용하여 추출할 스트로크를 확인하여 기존 InkDisp 개체에서 스트로크를 잘라내거나 복사하여 새 InkDisp 개체에 붙여넣습니다.
FindIntersections 이 IInkStrokeDisp 개체가 알려진 InkStrokes 컬렉션 내의 다른 IInkStrokeDisp 개체와 교차하는 지점을 검색합니다.
get_AntiAliased 스트로크가 앤티앨리어스되는지 여부를 나타내는 값을 가져오거나 설정합니다. (Get)

get_Appearance InkEdit 컨트롤의 모양을 결정하는 값을 가져오거나 설정합니다(평면(시각 효과 없이 그린) 또는 3D(3차원 효과로 그린)인지 여부입니다. (Get)
get_Ascender 한 줄의 텍스트를 나타내는 IInkRecognitionAlternate 개체의 오름차순을 가져옵니다.
get_AttachedEditWindow 더 이상 사용되지 않습니다. PenInputPanel이 TIP(텍스트 입력 패널)로 대체되었습니다. PenInputPanel 개체가 연결된 개체의 창 핸들을 설정하거나 가져옵니다. (Get)
get_AttachedEditWindow ITextInputPanel 개체가 연결된 개체의 창 핸들을 가져오거나 설정합니다. (Get)
get_AttachMode InkOverlay 개체가 알려진 창 뒤 또는 앞에 연결되어 있는지 여부를 지정하는 값을 가져오거나 설정합니다. (Get)
get_AutoRedraw 창이 무효화될 때 잉크 수집기가 잉크를 다시 칠할지 여부를 지정하는 값을 가져오거나 설정합니다. (IInkCollector.get_AutoRedraw)
get_AutoRedraw 창이 무효화될 때 잉크 수집기가 잉크를 다시 칠할지 여부를 지정하는 값을 가져오거나 설정합니다. (IInkOverlay.get_AutoRedraw)
get_AutoRedraw 창이 무효화될 때 잉크 수집기가 잉크를 다시 칠할지 여부를 지정하는 값을 가져오거나 설정합니다. (IInkPicture.get_AutoRedraw)
get_AutoShow 더 이상 사용되지 않습니다. PenInputPanel이 TIP(텍스트 입력 패널)로 대체되었습니다. 펜을 사용하여 연결된 컨트롤에 포커스를 설정할 때 펜 입력 패널이 표시되는지 여부를 나타내는 값을 가져오거나 설정합니다. (IPenInputPanel.get_AutoShow)
get_BackColor InkPicture 컨트롤의 배경색을 가져오거나 설정합니다. (Get)

get_BackColor InkEdit 컨트롤의 배경색을 가져오거나 설정합니다. (Get)
get_Baseline 한 줄의 텍스트를 나타내는 IInkRecognitionAlternate 개체의 기준을 가져옵니다.
get_BezierCusps 스트로크의 Bezier 근사치의 인덱스를 포함하는 배열을 가져옵니다.
get_BezierPoints 스트로크의 베지어 근사치를 나타내는 제어점의 배열을 가져옵니다.
get_BorderStyle InkEdit 컨트롤에 테두리가 있는지 여부를 결정하는 값을 가져오거나 설정합니다. (Get)
get_Bottom InkRectangle 개체의 아래쪽 위치를 가져오거나 설정합니다. (Get)
get_Busy 더 이상 사용되지 않습니다. PenInputPanel이 TIP(텍스트 입력 패널)로 대체되었습니다. PenInputPanel 개체가 현재 잉크를 처리하고 있는지 여부를 나타내는 값을 가져옵니다.
get_Buttons IInkCursor에서 사용할 수 있는 IInkCursorButtons 컬렉션을 가져옵니다.
get_Capabilities IInkRecognizer 개체의 기능을 가져옵니다.
get_CharacterAutoCompletionMode 문자 또는 단어가 인식되는 시기를 결정하는 문자 자동 완성 모드를 가져오거나 설정합니다. (Get)
get_ChildRealTimeStylusPlugin RealTimeStylus 개체를 현재 RealTimeStylus 개체의 비동기 플러그 인으로 가져오거나 설정합니다.
get_ClipRectangle DynamicRenderer Class 개체의 클리핑 사각형을 가져오거나 설정합니다. (Get)

get_ClipRegion DynamicRenderer Class 개체의 클리핑 영역을 가져오거나 설정합니다. (Get)
get_CollectingInk 잉크 수집기(InkCollector, InkOverlay 또는 InkPicture)에서 잉크를 현재 그릴지 여부를 지정하는 값을 가져옵니다. (IInkCollector.get_CollectingInk)
get_CollectingInk 잉크 수집기(InkCollector, InkOverlay 또는 InkPicture)에서 잉크를 현재 그릴지 여부를 지정하는 값을 가져옵니다. (IInkOverlay.get_CollectingInk)
get_CollectingInk 잉크 수집기(InkCollector, InkOverlay 또는 InkPicture)에서 잉크를 현재 그릴지 여부를 지정하는 값을 가져옵니다. (IInkPicture.get_CollectingInk)
get_CollectionMode 잉크, 제스처 또는 둘 다 사용자가 쓰는 것으로 인식되는지 여부를 결정하는 컬렉션 모드를 가져오거나 설정합니다. (IInkCollector.get_CollectionMode)
get_CollectionMode 잉크, 제스처 또는 둘 다 사용자가 쓰는 것으로 인식되는지 여부를 결정하는 컬렉션 모드를 가져오거나 설정합니다. (IInkOverlay.get_CollectionMode)
get_CollectionMode 잉크, 제스처 또는 둘 다 사용자가 쓰는 것으로 인식되는지 여부를 결정하는 컬렉션 모드를 가져오거나 설정합니다. (Get)
get_Color 이 InkDrawingAttributes 개체로 그린 잉크의 색을 가져오거나 설정합니다. (Get)
get_Columns 인식 가이드 상자의 열 수를 가져오거나 설정합니다. (Get)
get_Confidence IInkRecognitionAlternate 개체 또는 제스처를 인식할 때 인식기가 갖는 신뢰도 수준(강력, 중간 또는 불량)을 가져옵니다. (IInkGesture.get_Confidence)

get_Confidence IInkRecognitionAlternate 개체 또는 제스처를 인식할 때 인식기가 갖는 신뢰도 수준(강력, 중간 또는 불량)을 가져옵니다. (IInkRecognitionAlternate.get_Confidence)
get_ConfidenceAlternates 컬렉션의 각 대체 항목이 동일한 속성 값을 가진 세그먼트로 구성된 대체 항목의 컬렉션을 가져옵니다.
get_Count 컬렉션에 포함된 개체 또는 컬렉션의 수를 가져옵니다. (IInkCursorButtons.get_Count)
get_Count 컬렉션에 포함된 개체 또는 컬렉션의 수를 가져옵니다. (IInkCursors.get_Count)
get_Count 컬렉션에 포함된 개체 또는 컬렉션의 수를 가져옵니다. (IInkCustomStrokes.get_Count)
get_Count 컬렉션에 포함된 개체 또는 컬렉션의 수를 가져옵니다. (IInkExtendedProperties.get_Count)
get_Count 컬렉션에 포함된 개체 또는 컬렉션의 수를 가져옵니다. (IInkRecognitionAlternates.get_Count)
get_Count 컬렉션에 포함된 개체 또는 컬렉션의 수를 가져옵니다. (IInkRecognizers.get_Count)
get_Count 컬렉션에 포함된 개체 또는 컬렉션의 수를 가져옵니다. (IInkStrokes.get_Count)
get_Count 컬렉션에 포함된 개체 또는 컬렉션의 수를 가져옵니다. (IInkTablets.get_Count)
get_Count 컬렉션에 포함된 개체 또는 컬렉션의 수를 가져옵니다. (IInkDivisionUnits.get_Count)
get_CurrentCorrectionMode CorrectionMode 열거형에 지정된 현재 수정 빗 모드를 가져옵니다.

get_CurrentInPlaceState InPlaceState 열거형에 지정된 현재 현재 위치 상태를 가져옵니다.
get_CurrentInputArea PanelInputArea 열거형에 지정된 현재 입력 영역을 가져옵니다.
get_CurrentInteractionMode InteractionMode 열거형에 지정된 태블릿 PC 입력 패널의 위치를 가져옵니다.
get_CurrentPanel 더 이상 사용되지 않습니다. PenInputPanel이 TIP(텍스트 입력 패널)로 대체되었습니다. PenInputPanel 개체 내의 입력에 현재 사용되는 패널 형식을 가져오거나 설정합니다. (IPenInputPanel.get_CurrentPanel)
get_Cursors 수동 입력 영역에서 사용할 수 있는 커서 컬렉션을 가져옵니다. 각 커서는 펜 또는 다른 잉크 입력 장치의 끝에 해당합니다. (IInkCollector.get_Cursors)
get_Cursors 수동 입력 영역에서 사용할 수 있는 커서 컬렉션을 가져옵니다. 각 커서는 펜 또는 다른 잉크 입력 장치의 끝에 해당합니다. (IInkOverlay.get_Cursors)
get_Cursors 수동 입력 영역에서 사용할 수 있는 커서 컬렉션을 가져옵니다. 각 커서는 펜 또는 다른 잉크 입력 장치의 끝에 해당합니다. (IInkPicture.get_Cursors)
get_CustomStrokes 잉크와 함께 유지할 사용자 지정 스트로크의 컬렉션을 가져옵니다.
get_Data 확장 속성의 데이터를 가져오거나 설정합니다. (Get)
get_Data 사각형 구조체에 대한 액세스를 가져오거나 설정합니다(C++에만 해당). (Get)
get_Data XFORM 구조체에 대한 액세스를 가져오거나 설정합니다. (Get)

get_DataCacheEnabled DynamicRenderer Class 개체에 대해 데이터 캐싱을 사용할 수 있는지 여부를 나타내는 값을 가져오거나 설정합니다. (Get)
get_DefaultDrawingAttributes 잉크를 그리거나 표시할 때 사용할 기본 그리기 특성을 가져오거나 설정합니다. (IInkCollector.get_DefaultDrawingAttributes)
get_DefaultDrawingAttributes 잉크를 그리거나 표시할 때 사용할 기본 그리기 특성을 가져오거나 설정합니다. (IInkOverlay.get_DefaultDrawingAttributes)
get_DefaultDrawingAttributes 잉크를 그리거나 표시할 때 사용할 기본 그리기 특성을 가져오거나 설정합니다. (IInkPicture.get_DefaultDrawingAttributes)
get_DefaultInPlaceState InPlaceState 열거형에 지정된 기본 현재 위치 상태를 가져오거나 설정합니다. (Get)
get_DefaultInputArea PanelInputArea 열거형에 지정된 기본 입력 영역을 가져오거나 설정합니다. (Get)
get_DefaultPanel 더 이상 사용되지 않습니다. PenInputPanel이 TIP(텍스트 입력 패널)로 대체되었습니다. PenInputPanel 개체 내의 입력에 사용되는 기본 패널 형식을 가져오거나 설정합니다. (Get)
get_DefaultTablet 사용 가능한 태블릿 집합 내에서 기본 태블릿을 가져옵니다.
get_Deleted 알려진 스트로크가 잉크에서 삭제되는지 여부를 지정하는 값을 가져옵니다.
get_Descender 한 줄의 텍스트를 나타내는 IInkRecognitionAlternate 개체의 하위 줄을 가져옵니다.
get_DesiredPacketDescription InkCollector의 원하는 패킷 설명을 가져오거나 설정합니다. (IInkCollector.get_DesiredPacketDescription)

get_DesiredPacketDescription InkCollector의 원하는 패킷 설명을 가져오거나 설정합니다. (IInkOverlay.get_DesiredPacketDescription)
get_DesiredPacketDescription InkCollector의 원하는 패킷 설명을 가져오거나 설정합니다. (IInkPicture.get_DesiredPacketDescription)
get_DeviceKind 사용 중인 태블릿 디바이스의 유형을 가져옵니다.
get_Dirty 잉크를 마지막으로 저장한 이후 InkDisp Class 개체의 스트로크가 수정되었는지 여부를 지정하는 값을 가져오거나 설정합니다. (Get)
get_DisableNoScroll InkEdit 컨트롤의 스크롤 막대를 사용할 수 없는지 여부를 결정하는 값을 가져오거나 설정합니다. (Get)
get_DivisionType 분석 결과 내에서 IInkDivisionUnit 개체가 나타내는 구조 단위의 형식을 가져옵니다.
get_DrawingAttributes 잉크를 그릴 때 잉크에 적용할 그리기 특성을 가져오거나 설정합니다. (IInkCursor.get_DrawingAttributes)
get_DrawingAttributes 잉크를 그릴 때 잉크에 적용할 그리기 특성을 가져오거나 설정합니다. (IInkStrokeDisp.get_DrawingAttributes)
get_DrawingAttributes InkEdit 컨트롤에 아직 그려지지 않은 잉크의 그리기 특성을 가져오거나 설정합니다. (IInkEdit.get_DrawingAttributes)
get_DrawingAttributes DynamicRenderer Class 개체에서 사용하는 DrawingAttributes 개체를 가져오거나 설정합니다.

get_DrawnBox 태블릿의 화면에 물리적으로 그려지고 쓰기가 이루어지는 상자를 가져오거나 설정합니다. (Get)
get_DynamicRendering 잉크가 그려질 때 렌더링되는지 여부를 지정하는 값을 가져오거나 설정합니다. (IInkCollector.get_DynamicRendering)
get_DynamicRendering 잉크가 그려질 때 렌더링되는지 여부를 지정하는 값을 가져오거나 설정합니다. (IInkOverlay.get_DynamicRendering)
get_DynamicRendering 잉크가 그려질 때 렌더링되는지 여부를 지정하는 값을 가져오거나 설정합니다. (IInkPicture.get_DynamicRendering)
get_EditingMode 개체/컨트롤이 잉크 모드, 삭제 모드 또는 선택/편집 모드에 있는지 여부를 지정하는 값을 가져오거나 설정합니다. (Get)
get_EditingMode InkPicture 컨트롤이 잉크 모드, 삭제 모드 또는 선택/편집 모드에 있는지 여부를 지정하는 값을 가져오거나 설정합니다. (Get)
get_eDx InkTransform 개체로 표시되는 아핀 변환 행렬의 첫 번째 열인 세 번째 행의 요소를 가져오거나 설정합니다. (Get)
get_eDy InkTransform 개체로 표시되는 아핀 변환 행렬의 세 번째 행 두 번째 열에 있는 요소를 가져오거나 설정합니다. (Get)
get_eM11 InkTransform 개체로 표시되는 아핀 변환 행렬의 첫 번째 열인 첫 번째 행에 있는 요소를 가져오거나 설정합니다. (Get)
get_eM12 InkTransform 개체가 나타내는 아핀 변환 행렬의 첫 번째 행 두 번째 열에 있는 요소를 가져오거나 설정합니다. (Get)

get_eM21 InkTransform 개체로 표현되는 아핀 변환 행렬의 첫 번째 열인 두 번째 행의 요소를 가져오거나 설정합니다. (Get)
get_eM22 InkTransform 개체로 표시되는 아핀 변환 행렬의 두 번째 열인 두 번째 행에 있는 요소를 가져오거나 설정합니다. (Get)
get_Enabled InkCollector 개체가 펜 입력(대기 패킷, 범위 이벤트의 커서 등)을 수집하는지 여부를 지정하는 값을 가져오거나 설정합니다. (Get)
get_Enabled InkOverlay 개체가 펜 입력(대기 패킷, 범위 이벤트의 커서 등)을 수집하는지 여부를 지정하는 값을 가져오거나 설정합니다. (Get)
get_Enabled InkPicture 컨트롤이 사용자 생성 이벤트에 응답할 수 있는지 여부를 결정하는 값을 가져오거나 설정합니다. (Get)
get_Enabled InkEdit 컨트롤이 사용자 생성 이벤트에 응답할 수 있는지 여부를 결정하는 값을 가져오거나 설정합니다. (Get)
get_Enabled 동적 렌더링을 켜고 끄는 값을 가져오거나 설정합니다. (Get)
get_Enabled 제스처 인식이 사용되는지 여부를 나타내는 값을 가져오거나 설정합니다. (Get)
get_Enabled RealTimeStylus 개체가 태블릿 펜 데이터를 수집하는지 여부를 지정하는 값을 가져오거나 설정합니다. (Get)
get_EnabledUnicodeRanges 인식기 컨텍스트에서 지원할 하나 이상의 유니코드 범위 집합을 가져오거나 설정합니다. (Get)

get_EraserMode 잉크가 스트로크 또는 포인트별로 지워지는지 여부를 지정하는 값을 가져오거나 설정합니다. (Get)
get_EraserMode 잉크가 스트로크 또는 지점별로 지워지는지 여부를 지정하는 값을 가져오거나 설정합니다. (Get)
get_EraserWidth 지우개 펜 팁의 너비를 지정하는 값을 가져오거나 설정합니다. (Get)
get_EraserWidth 지우개 펜 팁의 너비를 지정하는 값을 가져오거나 설정합니다. (Get)
get_ExpandPostInsertionCorrection 태블릿 PC 입력 패널의 수정 빗이 자동으로 확장되는지 여부를 나타내는 값을 가져오거나 설정합니다. (Get)
get_ExtendedProperties 개체에 저장된 애플리케이션 정의 데이터의 컬렉션을 가져옵니다. (IInkDisp.get_ExtendedProperties)
get_ExtendedProperties 개체에 저장된 애플리케이션 정의 데이터의 컬렉션을 가져옵니다. (IInkDrawingAttributes.get_ExtendedProperties)
get_ExtendedProperties 개체에 저장된 애플리케이션 정의 데이터의 컬렉션을 가져옵니다. (IInkStrokeDisp.get_ExtendedProperties)
get_Factoid 인식기가 인식 결과에 대한 검색을 제한하는 데 사용하는 factoid를 가져오거나 설정합니다. (Get)
get_Factoid 더 이상 사용되지 않습니다. PenInputPanel이 TIP(텍스트 입력 패널)으로 대체되었습니다. PenInputPanel 개체에서 사용하는 factoid의 문자열 이름을 가져오거나 설정합니다. (Get)
get_Factoid IInkRecognizer 개체가 인식 결과에 대한 검색을 제한하는 데 사용하는 Factoid 상수를 가져오거나 설정합니다. (Get)

get_FitToCurve Bezier smoothing이 잉크를 렌더링하는 데 사용되는지 여부를 지정하는 값을 가져오거나 설정합니다. (Get)
get_FlicksEnabled RTS에 대해 플릭 제스처를 사용할 수 있는지 여부를 나타내는 값을 반환합니다.
get_Font Font 개체를 가져오거나 설정합니다.
get_Guid IInkExtendedProperty 개체의 GUID(Globally Unique Identifier)를 가져옵니다.
get_Guide 잉크 입력에 사용할 InkRecognizerGuide를 가져오거나 설정합니다. (IInkRecognizerContext.get_Guide)
get_GuideData 인식기에 대한 잉크의 경계를 나타내는 InkRecoGuide 구조를 가져오거나 설정합니다. (Get)
get_HardwareCapabilities 커서가 태블릿과 물리적으로 접촉하여 위치를 보고해야 하는지 여부와 같이 태블릿의 하드웨어 기능을 정의하는 비트 마스크를 가져옵니다.
get_Height InkDrawingAttributes 개체를 사용하여 잉크를 그릴 때 펜의 높이를 가져오거나 설정합니다. (Get)
get_Height 더 이상 사용되지 않습니다. PenInputPanel이 TIP(텍스트 입력 패널)으로 대체되었습니다. 클라이언트 좌표에서 펜 입력 패널의 높이를 가져옵니다.
get_HorizontalOffset 더 이상 사용되지 않습니다. PenInputPanel이 TIP(텍스트 입력 패널)으로 대체되었습니다. 펜 입력 패널의 왼쪽 가장자리와 펜이 연결된 컨트롤의 왼쪽 가장자리 사이의 오프셋을 가져오거나 설정합니다. (IPenInputPanel.get_HorizontalOffset)
get_hWnd 잉크가 그려지는 창의 핸들 값을 가져오거나 설정합니다. (IInkCollector.get_hWnd)

get_hWnd 잉크가 그려지는 창의 핸들 값을 가져오거나 설정합니다. (IInkOverlay.get_hWnd)
get_hWnd 잉크가 그려지는 창의 핸들 값을 가져오거나 설정합니다. (IInkPicture.get_hWnd)
get_Hwnd InkEdit 컨트롤에 대한 핸들을 가져옵니다.
get_HWND DynamicRenderer Class 개체와 연결된 창 핸들 HWND를 가져오거나 설정합니다. (Get)
get_HWND RealTimeStylus 개체에서 사용하는 창과 연결된 핸들 값을 가져오거나 설정합니다. (Get)
get_Id 개체의 식별자를 가져옵니다. (IInkCursor.get_ID)
get_Id 개체의 식별자를 가져옵니다. (IInkCursorButton.get_Id)
get_Id 개체의 식별자를 가져옵니다. (IInkGesture.get_Id)
get_Id InkRecognizer의 ID를 검색합니다.
get_ID 개체의 식별자를 가져옵니다. (IInkStrokeDisp.get_Id)
get_IgnorePressure 태블릿 표면에서 펜 팁의 압력 증가로 그려진 잉크가 자동으로 더 넓어지는지 여부를 지정하는 값을 가져오거나 설정합니다. (Get)
get_Ink InkCollector 개체 또는 InkOverlay 개체와 연결된 InkDisp 개체를 가져오거나 설정합니다. (IInkCollector.get_Ink)

get_Ink InkCollector 개체 또는 InkOverlay 개체와 연결된 InkDisp 개체를 가져오거나 설정합니다. (IInkOverlay.get_Ink)
get_Ink InkPicture 컨트롤과 연결된 InkDisp 개체를 가져오거나 설정합니다.
get_Ink 스트로크의 부모 InkDisp 개체를 가져옵니다.
get_Ink 스트로크 컬렉션을 포함하는 InkDisp 개체를 가져옵니다.
get_Ink IStrokeBuilder 개체와 연결된 잉크 개체를 가져오거나 설정합니다.
get_InkEnabled InkPicture 컨트롤이 펜 입력(대기 패킷, 범위 이벤트의 커서 등)을 수집하는지 여부를 지정하는 값을 가져오거나 설정합니다. (Get)
get_InkInsertMode InkEdit 컨트롤에 잉크를 텍스트 또는 잉크로 삽입하는 방법을 지정하는 값을 가져오거나 설정합니다. (Get)
get_InkMode 잉크 컬렉션을 사용하지 않도록 설정하거나, 잉크를 수집하거나, 잉크 및 제스처를 수집할지 여부를 지정하는 값을 가져오거나 설정합니다. (Get)
get_InPlaceBoundingRectangle 태블릿 PC 입력 패널에 대한 경계 사각형을 가져옵니다.
get_InPlaceVisibleOnFocus 연결된 창이 포커스를 받을 때 태블릿 PC 입력 패널이 자동으로 표시되는지 여부를 나타내는 값을 가져오거나 설정합니다. (Get)
get_Inverted 커서가 펜의 반전된 끝인지 여부를 나타내는 값을 가져옵니다.

get_IsMultiTouch 입력 디바이스가 다중 터치를 지원하는지 여부를 나타내는 값을 가져옵니다.
get_Languages IInkRecognizer 개체가 지원하는 언어에 대한 언어 식별자 배열을 가져옵니다.
get_Left InkRectangle 개체의 왼쪽 위치를 가져오거나 설정합니다. (Get)
get_Left 더 이상 사용되지 않습니다. PenInputPanel이 TIP(텍스트 입력 패널)으로 대체되었습니다. 화면 좌표로 PenInputPanel 개체의 왼쪽 가장자리의 가로 또는 x축 위치를 가져옵니다.
get_LineAlternates 컬렉션의 각 대체 항목이 별도의 줄에 있는 IInkRecognitionAlternates 컬렉션을 가져옵니다.
get_LineHeight HIMETRIC 단위로 예상되는 필기 높이를 가져오거나 설정합니다. (Get)
get_LineNumber 대체를 구성하는 잉크의 줄 번호를 가져옵니다.
get_Locked InkEdit 컨트롤의 내용을 편집할 수 있는지 여부를 나타내는 값을 가져오거나 설정합니다. (Get)
get_MarginX 창 사각형 주위의 x축 여백을 화면 좌표로 가져오거나 설정합니다. 이 여백은 잉크 창의 가장자리 주위에 버퍼를 제공합니다. (IInkCollector.get_MarginX)
get_MarginX 창 사각형 주위의 x축 여백을 화면 좌표로 가져오거나 설정합니다. 이 여백은 잉크 창의 가장자리 주위에 버퍼를 제공합니다. (IInkOverlay.get_MarginX)
get_MarginX 창 사각형 주위의 x축 여백을 화면 좌표로 가져오거나 설정합니다. 이 여백은 잉크 창의 가장자리 주위에 버퍼를 제공합니다. (IInkPicture.get_MarginX)

get_MarginY 창 사각형 주위의 y축 여백을 화면 좌표로 가져오거나 설정합니다. 이 여백은 잉크 창의 가장자리 주위에 버퍼를 제공합니다. (IInkCollector.get_MarginY)
get_MarginY 창 사각형 주위의 y축 여백을 화면 좌표로 가져오거나 설정합니다. 이 여백은 잉크 창의 가장자리 주위에 버퍼를 제공합니다. (IInkOverlay.get_MarginY)
get_MarginY 창 사각형 주위의 y축 여백을 화면 좌표로 가져오거나 설정합니다. 이 여백은 잉크 창의 가장자리 주위에 버퍼를 제공합니다. (IInkPicture.get_MarginY)
get_MaximumCursors 태블릿 디바이스에서 지원하는 최대 커서 수를 가져옵니다.
get_MaximumInputRectangle IInkTablet 개체가 지원하는 태블릿 디바이스 좌표의 최대 입력 사각형을 가져옵니다.
get_MaxLength InkEdit 컨트롤에서 보유할 수 있는 최대 문자 수가 있는지 여부를 나타내는 값을 가져오거나 설정하며, 있는 경우 최대 문자 수를 지정합니다. (Get)
get_MaxStrokeCount 제스처 인식당 허용되는 최대 스트로크 수를 가져오거나 설정합니다. (Get)
get_Midline 텍스트 한 줄을 나타내는 IInkRecognitionAlternate 개체의 중간선을 가져옵니다.
get_Midline 중간선 높이를 가져오거나 설정합니다. 중간선 높이는 기준선에서 중간선까지의 거리이며, 그려진 상자의 거리입니다. (Get)
get_MouseIcon 사용자 지정 마우스 아이콘을 가져오거나 설정합니다. (IInkCollector.get_MouseIcon)
get_MouseIcon 사용자 지정 마우스 아이콘을 가져오거나 설정합니다. (IInkOverlay.get_MouseIcon)

get_MouseIcon 사용자 지정 마우스 아이콘을 가져오거나 설정합니다. (IInkPicture.get_MouseIcon)
get_MouseIcon InkEdit 컨트롤에 대한 사용자 지정 마우스 아이콘을 가져오거나 설정합니다. (Get)
get_MousePointer 표시되는 마우스 포인터의 형식을 나타내는 값을 가져오거나 설정합니다. (IInkCollector.get_MousePointer)
get_MousePointer 표시되는 마우스 포인터의 형식을 나타내는 값을 가져오거나 설정합니다. (IInkOverlay.get_MousePointer)
get_MousePointer 표시되는 마우스 포인터의 형식을 나타내는 값을 가져오거나 설정합니다. (IInkPicture.get_MousePointer)
get_MousePointer 표시할 마우스 포인터의 유형을 나타내는 값을 가져오거나 설정합니다. (Get)
get_MultiLine InkEdit 컨트롤이 여러 줄의 텍스트를 수락하고 표시할 수 있는지 여부를 나타내는 값을 가져오거나 설정합니다. (Get)
get_MultiTouchEnabled IRealTimeStylus3 개체에 다중 터치 입력이 사용하도록 설정되어 있는지 여부를 나타냅니다. (Get)
get_Name 개체의 이름을 가져옵니다. (IInkCursor.get_Name)
get_Name 개체의 이름을 가져옵니다. (IInkCursorButton.get_Name)
get_Name 개체의 이름을 가져옵니다. (IInkRecognizer.get_Name)

get_Name 개체의 이름을 가져옵니다. (IInkTablet.get_Name)
get_PacketCount IInkStrokeDisp 개체에 대해 수신된 패킷 수를 가져옵니다.
get_PacketDescription IInkStrokeDisp 개체에 저장된 패킷 데이터의 형식을 설명하는 GUID(Globally Unique Identifiers)의 배열을 가져옵니다.
get_PacketSize 패킷의 크기(바이트)를 가져옵니다.
get_PenTip 이 InkDrawingAttributes 개체와 연결된 잉크를 그릴 때 사용할 펜 팁을 나타내는 값을 가져오거나 설정합니다. (Get)
get_Picture InkPicture 컨트롤에 표시할 그래픽 파일을 가져옵니다. (Get)
get_PlugAndPlayId IInkTablet 개체의 플러그 앤 플레이 식별자의 문자열 표현을 가져옵니다.
get_PolylineCusps IInkStrokeDisp 개체의 cusps 인덱스를 포함하는 배열을 가져옵니다.
get_PopDownCorrectionHeight 삽입 후 수정 빗이 입력 패널 아래에 배치될 때의 높이를 가져옵니다.
get_PopUpCorrectionHeight 삽입 후 수정 빗이 입력 패널 위에 배치될 때의 높이를 가져옵니다.
get_PreferredInPlaceDirection 텍스트 입력 필드를 기준으로 현재 위치 입력 패널의 기본 방향을 가져오거나 설정합니다. (Get)
get_PreferredPacketDescription 인식기의 기본 패킷 속성을 나타내는 GUID(Globally Unique Identifiers)의 배열을 가져옵니다.

get_PrefixText InkRecognizerContext 개체의 InkStrokes 컬렉션 앞에 오는 문자를 가져오거나 설정합니다. (Get)
get_RasterOperation 펜과 배경의 색이 상호 작용하는 방식을 정의하는 값을 가져오거나 설정합니다. (Get)
get_RecognitionFlags 인식기가 잉크를 해석하고 결과 문자열을 결정하는 방법을 지정하는 플래그를 가져오거나 설정합니다. (Get)
get_RecognitionResult InkStrokes 컬렉션의 IInkRecognitionResult 개체를 가져옵니다.
get_RecognitionTimeout 수집된 마지막 IInkStrokeDisp 개체와 텍스트 인식 시작 사이의 시간 길이(밀리초)를 가져오거나 설정합니다. (Get)
get_Recognizer InkRecognizerContext 개체에서 사용하는 IInkRecognizer 개체를 가져옵니다.
get_Recognizer 인식에 사용할 IInkRecognizer 개체를 가져오거나 설정합니다.
get_RecognizerContext InkDivider 개체가 레이아웃 분석에 사용하는 InkRecognizerContext 개체를 가져오거나 설정합니다. (IInkDivider.get_RecognizerContext)
get_Renderer 잉크를 그리는 데 사용되는 InkRenderer 개체를 가져오거나 설정합니다. (IInkCollector.get_Renderer)
get_Renderer 잉크를 그리는 데 사용되는 InkRenderer 개체를 가져오거나 설정합니다. (IInkOverlay.get_Renderer)
get_Renderer 잉크를 그리는 데 사용되는 InkRenderer 개체를 가져오거나 설정합니다. (IInkPicture.get_Renderer)

get_Right InkRectangle 개체의 올바른 위치를 가져오거나 설정합니다. (Get)
get_RotationTransform IInkDivisionUnit 개체가 스트로크를 가로로 회전하는 데 사용하는 변환 행렬을 가져옵니다.
get_Rows 인식 가이드의 행 수를 가져오거나 설정합니다. (Get)
get_ScrollBars InkEdit 컨트롤에 표시할 스크롤 막대의 형식(있는 경우)을 가져오거나 설정합니다. (Get)
get_SelAlignment InkEdit 컨트롤의 단락 맞춤을 제어하는 값을 가져오거나 설정합니다. (Get)
get_SelBold InkEdit 컨트롤에서 현재 선택한 텍스트의 글꼴 스타일이 굵게 표시되는지 여부를 지정하는 값을 가져오거나 설정합니다. (Get)
get_SelCharOffset InkEdit 컨트롤의 텍스트가 기준선(일반) 위에 위 첨자로 표시되는지 또는 기준선 아래의 아래 첨자로 표시되는지 여부를 결정하는 값을 반환하거나 설정합니다. (Get)
get_SelColor 현재 텍스트 선택 영역 또는 삽입 지점의 텍스트 색을 가져오거나 설정합니다(런타임에만 해당). (Get)
get_Selection InkOverlay 개체 또는 InkPicture 컨트롤 내에서 현재 선택된 InkStrokes 컬렉션을 가져오거나 설정합니다. (Get)
get_Selection InkOverlay 개체 또는 InkPicture 컨트롤 내에서 현재 선택된InkStrokes 컬렉션을 가져오거나 설정합니다. (Get)
get_SelfIntersections 스트로크의 자체 교집합을 가져옵니다.

get_SelFontName InkEdit 컨트롤 내에서 선택한 텍스트의 글꼴 이름을 가져오거나 설정합니다(런타임에만 해당). (Get)
get_SelFontSize InkEdit 컨트롤 내에서 선택한 텍스트의 글꼴 크기를 가져오거나 설정합니다(런타임에만 해당). (Get)
get_SelInks 현재 선택 영역에서 포함된 InkDisp 개체의 배열(잉크로 표시되는 경우)을 가져오거나 설정합니다. (Get)
get_SelInksDisplayMode 잉크와 텍스트 간에 선택 영역의 모양을 전환할 수 있는 값을 가져오거나 설정합니다. (Get)
get_SelItalic InkEdit 컨트롤에서 현재 선택한 텍스트의 글꼴 스타일이 기울임꼴인지 여부를 지정하는 값을 가져오거나 설정합니다(런타임에만 해당). (Get)
get_SelLength InkEdit 컨트롤에서 선택한 문자 수를 가져오거나 설정합니다(런타임에만 해당). (Get)
get_SelRTF InkEdit 컨트롤에서 현재 선택한 RTF(서식 있는 텍스트 서식) 텍스트를 가져오거나 설정합니다(런타임에만 해당). (Get)
get_SelStart InkEdit 컨트롤에서 선택한 텍스트의 시작점을 가져오거나 설정합니다(런타임에만 해당). (Get)
get_SelText InkEdit 컨트롤 내에서 선택한 텍스트를 가져오거나 설정합니다(런타임에만 해당). (Get)
get_SelUnderline InkEdit 컨트롤에서 현재 선택한 텍스트의 글꼴 스타일에 밑줄이 그어지는지 여부를 지정하는 값을 가져오거나 설정합니다(런타임에만 해당). (Get)
get_SizeMode InkPicture 컨트롤이 이미지 배치 및 크기 조정을 처리하는 방법을 가져오거나 설정합니다. (Get)

get_State 단추를 사용할 수 없는지, 위쪽 또는 아래쪽인지와 같은 커서 단추의 상태를 가져옵니다.
get_Status InkEdit 컨트롤이 유휴 상태인지, 잉크를 수집하거나, 잉크를 인식하는지를 지정하는 값을 가져옵니다.
get_String 대체 항목의 위쪽 문자열을 가져옵니다.
get_Strokes 개체에 포함되거나 개체를 만드는 데 사용되는 스트로크 컬렉션을 가져옵니다. (IInkDisp.get_Strokes)
get_Strokes 개체에 포함되거나 개체를 만드는 데 사용되는 스트로크 컬렉션을 가져옵니다. (IInkRecognitionAlternate.get_Strokes)
get_Strokes 개체에 포함되거나 개체를 만드는 데 사용되는 스트로크 컬렉션을 가져옵니다. (IInkRecognitionAlternates.get_Strokes)
get_Strokes 개체에 포함되거나 개체를 만드는 데 사용되는 스트로크 컬렉션을 가져옵니다. (IInkRecognitionResult.get_Strokes)
get_Strokes InkRecognizerContext 개체와 연결된 InkStrokes 컬렉션을 가져오거나 설정합니다. (IInkRecognizerContext.get_Strokes)
get_Strokes InkDivider 개체가 레이아웃 분석을 수행하는 InkStrokes 컬렉션을 가져오거나 설정합니다. (IInkDivider.get_Strokes)
get_Strokes 개체에 포함되거나 개체를 만드는 데 사용되는 스트로크 컬렉션을 가져옵니다. (IInkDivisionResult.get_Strokes)

get_Strokes 개체에 포함되거나 개체를 만드는 데 사용되는 스트로크 컬렉션을 가져옵니다. (IInkDivisionUnit.get_Strokes)
get_SuffixText InkRecognizerContext 개체의 InkStrokes 컬렉션 다음에 오는 문자를 가져오거나 설정합니다. (Get)
get_SupportedProperties IInkRecognizer 개체가 지원하는 속성을 설명하는 GUID(Globally Unique Identifiers)의 배열을 가져옵니다.
get_SupportHighContrastInk 시스템이 고대비 모드일 때 잉크가 하나의 색으로 렌더링되는지 여부를 지정하는 값을 가져오거나 설정합니다. (IInkCollector.get_SupportHighContrastInk)
get_SupportHighContrastInk 시스템이 고대비 모드일 때 잉크가 하나의 색으로 렌더링되는지 여부를 지정하는 값을 가져오거나 설정합니다. (IInkOverlay.get_SupportHighContrastInk)
get_SupportHighContrastInk 시스템이 고대비 모드일 때 잉크가 하나의 색으로 렌더링되는지 여부를 지정하는 값을 가져오거나 설정합니다. (IInkPicture.get_SupportHighContrastInk)
get_SupportHighContrastSelectionUI 시스템이 고대비 모드일 때 모든 선택 UI(사용자 인터페이스) 요소가 고대비로 그려지는지 여부를 지정하는 값을 가져오거나 설정합니다. (Get)
get_SupportHighContrastSelectionUI 시스템이 고대비 모드일 때 모든 선택 사용자 인터페이스(선택 경계 상자 및 선택 핸들)가 고대비로 그려지는지 여부를 지정하는 값을 가져오거나 설정합니다. (Get)
get_Tablet 커서가 속한 IInkTablet 개체 또는 개체 또는 컨트롤이 현재 입력을 수집하는 데 사용하는 IInkTablet 개체를 가져옵니다. (IInkCollector.get_Tablet)
get_Tablet 커서가 속한 IInkTablet 개체 또는 개체 또는 컨트롤이 현재 입력을 수집하는 데 사용하는 IInkTablet 개체를 가져옵니다. (IInkCursor.get_Tablet)

get_Tablet 커서가 속한 IInkTablet 개체 또는 개체 또는 컨트롤이 현재 입력을 수집하는 데 사용하는 IInkTablet 개체를 가져옵니다. (IInkOverlay.get_Tablet)
get_Tablet 커서가 속한 IInkTablet 개체 또는 개체 또는 컨트롤이 현재 입력을 수집하는 데 사용하는 IInkTablet 개체를 가져옵니다. (IInkPicture.get_Tablet)
get_Text InkEdit 컨트롤에서 현재 텍스트를 가져오거나 설정합니다. (Get)
get_TextRTF 모든 RTF 코드를 포함하여 InkEdit 컨트롤의 텍스트를 가져오거나 설정합니다. (Get)
get_Top InkRectangle 개체의 위쪽 위치를 가져오거나 설정합니다. (Get)
get_Top 더 이상 사용되지 않습니다. PenInputPanel이 TIP(텍스트 입력 패널)로 대체되었습니다. PenInputPanel 개체의 위쪽 가장자리 위치를 화면 좌표로 세로 또는 y축으로 가져옵니다.
get_TopAlternate 인식 결과의 상위 대체를 가져옵니다. (IInkRecognitionResult.get_TopAlternate)
get_TopConfidence 인식 결과의 상위 대체를 가져옵니다. (IInkRecognitionResult.get_TopConfidence)
get_TopString TopAlternate 속성의 결과 텍스트를 가져옵니다.
get_Transparency 잉크의 투명도 값을 나타내는 값을 가져오거나 설정합니다. (Get)
get_UnicodeRanges 현재 인식기에서 설정된 유니코드 범위를 검색합니다.

get_UseMouseForInput 마우스를 입력 디바이스로 사용할 수 있는지 여부를 나타내는 값을 가져오거나 설정합니다. (Get)
get_Vendor IInkRecognizer 개체의 공급업체 이름을 가져옵니다.
get_VerticalOffset 더 이상 사용되지 않습니다. PenInputPanel이 TIP(텍스트 입력 패널)로 대체되었습니다. 펜 입력 패널의 가장 가까운 가로 가장자리와 펜이 연결된 컨트롤의 가장 가까운 가로 가장자리 사이의 오프셋을 가져오거나 설정합니다. (IPenInputPanel.get_VerticalOffset)
get_Visible 더 이상 사용되지 않습니다. PenInputPanel이 TIP(텍스트 입력 패널)로 대체되었습니다. PenInputPanel 개체가 표시되는지 여부를 나타내는 값을 가져오거나 설정합니다.
get_Width 잉크를 그릴 때 펜 팁의 y축 차원 또는 너비를 가져오거나 설정합니다. (Get)
get_Width 더 이상 사용되지 않습니다. PenInputPanel이 TIP(텍스트 입력 패널)로 대체되었습니다. 클라이언트 좌표에서 펜 입력 패널의 너비를 가져옵니다.
get_WindowInputRectangle RealTimeStylus Class 개체의 창 입력 사각형을 가져오거나 설정합니다. (Get)
get_WordList 인식 결과를 개선하기 위해 인식 프로세스에 사용되는 단어 목록을 가져오거나 설정합니다. (IInkRecognizerContext.get_WordList)
get_WritingBox 쓰기가 실제로 수행될 수 있는 인식 가이드의 보이지 않는 쓰기 영역을 가져오거나 설정합니다. (Get)
GetAllRecognizers 모든 인식기를 가져옵니다.
GetAllTabletContextIds 현재 활성 태블릿 컨텍스트 식별자를 모두 포함하는 배열을 검색합니다.

GetBestResultString 최상의 결과 문자열을 검색합니다.
GetBoundingBox InkDisp 개체의 모든 스트로크, 개별 스트로크 또는 InkStrokes 컬렉션에 대한 잉크 공간 좌표의 경계 상자를 검색합니다. (IInkDisp.GetBoundingBox)
GetBoundingBox InkDisp 개체의 모든 스트로크, 개별 스트로크 또는 InkStrokes 컬렉션에 대한 잉크 공간 좌표의 경계 상자를 검색합니다. (IInkStrokeDisp.GetBoundingBox)
GetBoundingBox InkDisp 개체의 모든 스트로크, 개별 스트로크 또는 InkStrokes 컬렉션에 대한 잉크 공간 좌표의 경계 상자를 가져옵니다.
GetCandidate 인식 결과 목록에서 하나의 인식 대체를 반환합니다.
GetContextPreferenceFlags 컨텍스트 기본 설정 플래그를 가져옵니다.
GetContextPropertyList 인식기가 지원하는 속성 목록을 검색합니다.
GetContextPropertyValue 인식기 컨텍스트에서 지정된 속성 값을 반환합니다.
GetDefaultRecognizer NLS(국가 언어 지원) LCID(언어 코드 식별자)로 지정된 알려진 언어의 기본 인식기를 검색합니다.
GetDesiredPacketDescription 패킷 스트림에 포함하도록 요청된 속성 목록을 검색합니다.
GetEnabledUnicodeRanges 컨텍스트에서 사용하도록 설정된 유니코드 지점 범위 목록을 검색합니다. SetEnabledUnicodeRanges 함수를 호출하여 사용 범위를 지정하지 않으면 이 함수는 인식기 기본 유니코드 점 범위를 반환합니다.

GetEventInterest InkCollector 클래스, InkOverlay 클래스 또는 InkPicture 클래스에 대한 특정 이벤트에서 개체가 갖는 관심사를 검색합니다. (IInkCollector.GetEventInterest)
GetEventInterest InkCollector 클래스, InkOverlay 클래스 또는 InkPicture 클래스에 대한 특정 이벤트에서 개체가 갖는 관심사를 검색합니다. (IInkOverlay.GetEventInterest)
GetEventInterest InkCollector 클래스, InkOverlay 클래스 또는 InkPicture 클래스에 대한 특정 이벤트에서 개체가 갖는 관심사를 검색합니다. (IInkPicture.GetEventInterest)
GetFlattenedBezierPoints InkDisp 개체의 모든 스트로크, 개별 스트로크 또는 InkStrokes 컬렉션에 대한 잉크 공간 좌표의 경계 상자를 검색합니다.
Getformat 텍스트 잉크 개체(tInk)에 현재 설정된 표시 속성을 반환합니다.
GetGestureStatus InkCollector 또는 InkOverlay 개체가 특정 애플리케이션 제스처에 관심이 있는지 여부를 나타냅니다.
GetGestureStatus InkCollector 또는 InkOverlay 개체가 특정 애플리케이션 제스처에 관심이 있는지 여부를 결정하는 값을 검색합니다.
GetGestureStatus InkPicture 컨트롤이 특정 애플리케이션 제스처에 관심이 있는지 여부를 나타내는 값을 검색합니다.
GetGestureStatus InkEdit 컨트롤이 특정 애플리케이션 제스처에 관심이 있는지 여부를 나타냅니다.
GetGuide 박스형, 줄 지어 또는 자유형 입력에 사용되는 가이드를 검색합니다.

GetHotPoint 잉크 공간 좌표에서 제스처의 핫포인트를 가져옵니다.
GetHoverIcon 가리키기 대상에 사용할 아이콘을 검색하여 수학 입력 컨트롤을 시작합니다.
GetInkExtent 텍스트 잉크 개체(tInk)에서 설정할 표시 속성을 지정하고 HIMETRIC 단위로 텍스트 잉크 개체의 너비를 검색합니다.
GetLatticePtr 현재 결과의 격자에 대한 포인터를 검색합니다.
GetLeftSeparator 인식기 컨텍스트의 왼쪽 구분 기호를 가져옵니다.
GetObjectTransform 잉크를 렌더링하는 데 사용된 개체 변환을 나타내는 InkTransform 개체를 가져옵니다.
GetPacketData IInkStrokeDisp 개체 내의 패킷 범위에 대한 패킷 데이터를 검색합니다.
GetPacketDescriptionData 패킷 속성 및 크기 조정 요소를 검색합니다.
GetPacketDescriptionPropertyMetrics 지정된 패킷 설명 형식에 대한 메트릭을 검색합니다.
GetPacketValuesByProperty 스트로크의 하나 이상의 패킷에서 알려진 패킷 속성에 대한 데이터를 검색합니다.
GetPoints 스트로크를 구성하는 점을 검색합니다.
GetPosition 컨트롤의 위치와 크기를 검색합니다.

GetPreferredPacketDescription 인식기에서 사용하는 패킷 속성이 포함된 패킷 설명을 검색합니다.
GetPreviewHeight 미리 보기 영역의 높이를 픽셀 단위로 검색합니다.
GetPropertyMetrics 지정된 속성에 대한 메트릭 데이터를 검색합니다.
GetPropertyValue 대체의 지정된 속성 값을 검색합니다.
GetRecoAttributes 인식기의 특성을 검색합니다.
GetRectangle 단일 호출에서 InkRectangle 개체의 요소를 가져옵니다.
GetRectangleIntersections IInkStrokeDisp 개체가 지정된 사각형과 교차하는 지점을 찾습니다.
GetResultPropertyList 인식기가 결과 범위에 대해 반환할 수 있는 속성 목록을 검색합니다.
GetRightSeparator 인식기 컨텍스트에 대한 오른쪽 구분 기호를 가져옵니다.
GetStrokesFromStrokeRanges 알려진 입력 InkStrokes 컬렉션을 포함하고 IInkRecognizer 개체가 대체를 제공할 수 있는 가장 작은 InkStrokes 컬렉션을 검색합니다.
GetStrokesFromTextRange 대체 항목 내에서 지정된 문자 범위를 포함하는 가장 작은 인식 세그먼트 집합에 해당하는 컬렉션을 검색합니다.
GetStylusAsyncPlugin 비동기 플러그 인 컬렉션의 지정된 인덱스에서 플러그 인을 검색합니다.

GetStylusAsyncPluginCount 비동기 플러그 인 컬렉션의 플러그 인 수를 검색합니다.
GetStyluses RealTimeStylus Class 개체가 발견된 스타일러스의 컬렉션을 검색합니다.
GetStylusForId 지정된 스타일러스 식별자에 대한 스타일러스를 검색합니다.
GetStylusSyncPlugin 동기 플러그 인 컬렉션의 지정된 인덱스에서 플러그 인을 검색합니다.
GetStylusSyncPluginCount 동기 플러그 인 컬렉션의 플러그 인 수를 검색합니다.
GetTablet 호출자에 대한 IInkTablet Interface 개체를 검색합니다.
GetTabletContextIdFromTablet 지정된 태블릿 디지타이저 개체와 연결된 TabletContextId 속성을 검색합니다.
GetTabletFromTabletContextId 지정된 태블릿 컨텍스트에 대한 IInkTablet 인터페이스를 검색합니다.
GetTextRangeFromStrokes 인식기가 알려진 InkStrokes 컬렉션을 포함하는 대체 항목을 반환할 수 있는 인식된 텍스트의 가장 작은 범위를 검색합니다.
GetTransform InkTransform 멤버 데이터를 가져옵니다.
GetUnicodeRanges 인식기가 지원하는 유니코드 지점의 범위를 반환합니다.
GetViewTransform 잉크를 렌더링하는 데 사용되는 뷰 변환을 나타내는 InkTransform 개체를 가져옵니다.

GetWindowInputRectangle 잉크가 그려지는 창 사각형을 픽셀 단위로 가져옵니다. (IInkCollector.GetWindowInputRectangle)
GetWindowInputRectangle 잉크가 그려지는 창 사각형을 픽셀 단위로 가져옵니다. (IInkOverlay.GetWindowInputRectangle)
GetWindowInputRectangle 잉크가 그려지는 창 사각형을 픽셀 단위로 검색합니다.
숨기기 컨트롤을 숨깁니다.
HitTestCircle 완전히 내부에 있거나 알려진 원에 의해 교차되는 InkStrokes 컬렉션을 검색합니다.
HitTestCircle 스트로크가 완전히 내부에 있는지 또는 지정된 원에 의해 교차되는지 여부를 결정합니다.
HitTestSelection 적중 테스트 중에 선택한 부분이 적중되었는지 확인합니다.
HitTestSelection 선택 영역의 어느 부분(있는 경우)이 적중 테스트 중에 적중되었는지 지정하는 SelectionHitResult 열거형의 멤버를 검색합니다.
HitTestWithLasso 폴리라인 선택 영역 내의 스트로크를 검색합니다.
HitTestWithRectangle 지정된 사각형 내에 포함된 스트로크를 검색합니다.
InAirPackets 플러그 인을 구현하는 개체에 스타일러스가 디지타이저 위로 이동하고 있음을 알릴 수 있습니다.
InkSpaceToPixel 변환을 위해 핸들을 사용하여 잉크 공간 좌표의 위치를 픽셀 공간의 위치로 변환합니다.

InkSpaceToPixelFromPoints 잉크 공간 좌표의 점 배열을 픽셀 공간의 점 배열로 변환합니다.
InPlaceSizeChanged 사용자 크기 조정, 자동 증가 또는 입력 영역 변경으로 인해 현재 위치 입력 패널 크기가 변경된 경우에 발생합니다.
InPlaceSizeChanging 사용자 크기 조정, 자동 증가 또는 입력 영역 변경으로 인해 현재 위치 입력 패널 크기가 변경되려고 할 때 발생합니다.
InPlaceStateChanged In-Place 상태가 변경되면 발생합니다.
InPlaceStateChanging In-Place 상태가 변경하려고 할 때 발생합니다.
InPlaceVisibilityChanged 태블릿 PC 입력 패널이 표시와 보이지 않는 사이를 전환한 경우에 발생합니다.
InPlaceVisibilityChanging 태블릿 PC 입력 패널이 표시와 보이지 않는 사이를 전환하려고 할 때 발생합니다.
InputAreaChanged 태블릿 PC 입력 패널에서 입력 영역이 변경된 경우에 발생합니다.
InputAreaChanging 태블릿 PC 입력 패널에서 입력 영역이 변경되려고 할 때 발생합니다.
InsertInkRecognitionResult 인식 결과를 삽입합니다.
InsertRecognitionResultsArray 인식 결과 배열을 삽입합니다.

IsPacketPropertySupported 태블릿 디바이스의 속성 또는 GUID(Globally Unique Identifier)로 식별된 태블릿 디바이스 컬렉션이 지원되는지 여부를 결정합니다. (IInkTablet.IsPacketPropertySupported)
IsPacketPropertySupported 태블릿 디바이스의 속성 또는 GUID(Globally Unique Identifier)로 식별된 태블릿 디바이스 컬렉션이 지원되는지 여부를 결정합니다. (IInkTablets.IsPacketPropertySupported)
IsStringSupported 시스템 사전, 사용자 사전 또는 단어 목록에 지정된 문자열이 포함되어 있는지 여부를 나타냅니다.
IsStringSupported 전달된 단어, 날짜, 시간, 숫자 또는 기타 텍스트가 사전에 포함되어 있는지 여부를 나타내는 값을 반환합니다. 이 테스트의 결과는 factoid 설정에 따라 달라집니다.
IsTipRunning 메서드가 호출된 시점에 태블릿 PC 입력 패널이 실행 중인지를 나타냅니다.
IsVisible 컨트롤이 표시되는지 여부를 결정합니다.
Item IInkCursorButtons 컬렉션 내의 지정된 인덱스 또는 문자열 식별자에서 IInkCursorButton 개체를 검색합니다.
Item IInkCursors 컬렉션 내의 지정된 인덱스에 있는 IInkCursor 개체를 반환합니다.
Item IInkCustomStrokes 인터페이스 내에 지정된 위치에서 InkStrokes 컬렉션을 검색합니다.
Item IInkExtendedProperties 컬렉션 내의 지정된 인덱스에서 IInkExtendedProperty 개체를 검색합니다.
Item IInkRecognitionAlternates 컬렉션 내의 지정된 인덱스에서 IInkRecognitionAlternate 개체를 검색합니다.

Item InkRecognizers 컬렉션 내의 지정된 인덱스에서 IInkRecognizer 개체를 검색합니다.
Item InkStrokes 컬렉션 내의 지정된 인덱스에서 IInkStrokeDisp 개체를 검색합니다.
Item InkTablets 컬렉션 내의 지정된 인덱스에서 IInkTablet 개체를 검색합니다.
Item IInkDivisionUnits 컬렉션 내의 지정된 인덱스에서 IInkDivisionUnit 개체를 검색합니다.
로드 알려진 이진 데이터로 새 InkDisp 개체를 채웁니다.
LoadCachedAttributes 인식기의 캐시된 특성을 로드합니다.
LoadInk 잉크를 처리하고 인식을 트리거합니다.
MakeWordList 단어 목록을 만듭니다.
측정값 DrawStroke 메서드를 사용하여 스트로크를 InkRenderer 개체로 그린 경우 스트로크 컬렉션을 포함하는 디바이스 컨텍스트의 사각형을 계산합니다.
MeasureStroke DrawStroke 메서드를 사용하여 InkRenderer 개체로 그린 경우 스트로크를 포함하는 디바이스 컨텍스트의 사각형을 계산합니다.
병합 지정된 InkWordList 개체를 이 단어 목록에 병합합니다.
ModifyDrawingAttributes InkStrokes 컬렉션에 있는 모든 스트로크의 그리기 특성을 설정합니다.

ModifyTopAlternate 지정된 대체를 사용하여 인식 결과의 상위 대체를 변경합니다.
이동 잉크 공간 좌표의 뷰 변환에 변환을 적용합니다.
이동 IInkStrokeDisp 개체 또는 InkStrokes 컬렉션의 잉크에 번역을 적용합니다. (IInkStrokeDisp.Move)
이동 IInkStrokeDisp 개체 또는 InkStrokes 컬렉션의 잉크에 번역을 적용합니다. (IInkStrokes.Move)
MoveTo 더 이상 사용되지 않습니다. PenInputPanel이 TIP(텍스트 입력 패널)으로 대체되었습니다. PenInputPanel 개체의 위치를 정적 화면 위치로 설정합니다.
NearestPoint 알려진 지점에 가장 가까운 InkDisp 개체 내에서 IInkStrokeDisp를 검색하여 필요에 따라 가장 가까운 지점의 인덱스와 지정된 지점에서 스트로크까지의 거리를 제공합니다.
NearestPoint 알려진 지점에 가장 가까운 스트로크의 위치를 찾아 해당 지점이 스트로크에서의 거리를 반환합니다. 모든 항목은 잉크 공간 좌표에 있습니다.
패킷 수 플러그 인을 구현하는 개체에 태블릿 펜이 디지타이저에서 이동하고 있음을 알 수 있습니다.
PixelToInkSpace 픽셀 공간 좌표의 위치를 잉크 공간 좌표의 위치로 변환합니다.
PixelToInkSpaceFromPoints 픽셀 공간 좌표의 위치 배열을 잉크 공간 좌표의 위치 배열로 변환합니다.
Process 잉크 인식을 동기적으로 수행합니다.

put_AntiAliased 스트로크가 앤티앨리어스되는지 여부를 나타내는 값을 가져오거나 설정합니다. (Put)
put_Appearance InkEdit 컨트롤의 모양을 결정하는 값을 가져오거나 설정합니다(평면(시각 효과 없이 그린)인지 3D(3차원 효과로 칠해져 있는지 여부). (Put)
put_AttachedEditWindow 더 이상 사용되지 않습니다. PenInputPanel이 TIP(텍스트 입력 패널)으로 대체되었습니다. PenInputPanel 개체가 연결된 개체의 창 핸들을 설정하거나 가져옵니다. (Put)
put_AttachedEditWindow ITextInputPanel 개체가 연결된 개체의 창 핸들을 가져오거나 설정합니다. (Put)
put_AttachMode InkOverlay 개체가 알려진 창 뒤에 또는 앞에 연결되어 있는지 여부를 지정하는 값을 가져오거나 설정합니다. (Put)
put_AutoRedraw 창이 무효화될 때 잉크 수집기가 잉크를 다시 칠할지 여부를 지정하는 값을 가져오거나 설정합니다. (IInkCollector.put_AutoRedraw)
put_AutoRedraw 창이 무효화될 때 잉크 수집기가 잉크를 다시 칠할지 여부를 지정하는 값을 가져오거나 설정합니다. (IInkOverlay.put_AutoRedraw)
put_AutoRedraw 창이 무효화될 때 잉크 수집기가 잉크를 다시 칠할지 여부를 지정하는 값을 가져오거나 설정합니다. (IInkPicture.put_AutoRedraw)
put_AutoShow 더 이상 사용되지 않습니다. PenInputPanel이 TIP(텍스트 입력 패널)으로 대체되었습니다. 펜을 사용하여 연결된 컨트롤에 포커스를 설정할 때 펜 입력 패널이 표시되는지 여부를 나타내는 값을 가져오거나 설정합니다. (IPenInputPanel.put_AutoShow)
put_BackColor InkPicture 컨트롤의 배경색을 가져오거나 설정합니다. (Put)

put_BackColor InkEdit 컨트롤의 배경색을 가져오거나 설정합니다. (Put)
put_BorderStyle InkEdit 컨트롤에 테두리가 있는지 여부를 결정하는 값을 가져오거나 설정합니다. (Put)
put_Bottom InkRectangle 개체의 아래쪽 위치를 가져오거나 설정합니다. (Put)
put_CharacterAutoCompletionMode 문자 또는 단어가 인식되는 시기를 결정하는 문자 자동 완성 모드를 가져오거나 설정합니다. (Put)
put_ClipRectangle DynamicRenderer Class 개체의 클리핑 사각형을 가져오거나 설정합니다. (Put)
put_ClipRegion DynamicRenderer Class 개체의 클리핑 영역을 가져오거나 설정합니다. (Put)
put_CollectionMode 잉크, 제스처 또는 둘 다 사용자가 쓰는 것으로 인식되는지 여부를 결정하는 컬렉션 모드를 가져오거나 설정합니다. (IInkCollector.put_CollectionMode)
put_CollectionMode 잉크, 제스처 또는 둘 다 사용자가 쓰는 것으로 인식되는지 여부를 결정하는 컬렉션 모드를 가져오거나 설정합니다. (IInkOverlay.put_CollectionMode)
put_CollectionMode 잉크, 제스처 또는 둘 다 사용자가 쓰는 것으로 인식되는지 여부를 결정하는 컬렉션 모드를 가져오거나 설정합니다. (Put)
put_Color 이 InkDrawingAttributes 개체로 그린 잉크의 색을 가져오거나 설정합니다. (Put)
put_Columns 인식 안내선 상자의 열 수를 가져오거나 설정합니다. (Put)

put_CurrentPanel 더 이상 사용되지 않습니다. PenInputPanel이 TIP(텍스트 입력 패널)으로 대체되었습니다. PenInputPanel 개체 내의 입력에 현재 사용되는 패널 형식을 가져오거나 설정합니다. (IPenInputPanel.put_CurrentPanel)
put_Data 확장 속성의 데이터를 가져오거나 설정합니다. (Put)
put_Data 사각형 구조체에 대한 액세스를 가져오거나 설정합니다(C++에만 해당). (Put)
put_Data XFORM 구조체에 대한 액세스를 가져오거나 설정합니다. (Put)
put_DataCacheEnabled DynamicRenderer Class 개체에 대해 데이터 캐싱을 사용할 수 있는지 여부를 나타내는 값을 가져오거나 설정합니다. (Put)
put_DefaultInPlaceState InPlaceState 열거형에 지정된 기본 현재 위치 상태를 가져오거나 설정합니다. (Put)
put_DefaultInputArea PanelInputArea 열거형에 지정된 기본 입력 영역을 가져오거나 설정합니다. (Put)
put_DefaultPanel 더 이상 사용되지 않습니다. PenInputPanel이 TIP(텍스트 입력 패널)으로 대체되었습니다. PenInputPanel 개체 내의 입력에 사용되는 기본 패널 형식을 가져오거나 설정합니다. (Put)
put_DesiredPacketDescription InkCollector의 원하는 패킷 설명을 가져오거나 설정합니다. (IInkCollector.put_DesiredPacketDescription)
put_DesiredPacketDescription InkCollector의 원하는 패킷 설명을 가져오거나 설정합니다. (IInkOverlay.put_DesiredPacketDescription)

put_DesiredPacketDescription InkCollector의 원하는 패킷 설명을 가져오거나 설정합니다. (IInkPicture.put_DesiredPacketDescription)
put_Dirty 잉크를 마지막으로 저장한 이후 InkDisp Class 개체의 스트로크가 수정되었는지 여부를 지정하는 값을 가져오거나 설정합니다. (Put)
put_DisableNoScroll InkEdit 컨트롤의 스크롤 막대를 사용할 수 없는지 여부를 결정하는 값을 가져오거나 설정합니다. (Put)
put_DrawnBox 태블릿의 화면에 물리적으로 그려지고 쓰기가 이루어지는 상자를 가져오거나 설정합니다. (Put)
put_DynamicRendering 잉크가 그려질 때 렌더링되는지 여부를 지정하는 값을 가져오거나 설정합니다. (IInkCollector.put_DynamicRendering)
put_DynamicRendering 잉크가 그려질 때 렌더링되는지 여부를 지정하는 값을 가져오거나 설정합니다. (IInkOverlay.put_DynamicRendering)
put_DynamicRendering 잉크가 그려질 때 렌더링되는지 여부를 지정하는 값을 가져오거나 설정합니다. (IInkPicture.put_DynamicRendering)
put_EditingMode 개체/컨트롤이 잉크 모드, 삭제 모드 또는 선택/편집 모드에 있는지 여부를 지정하는 값을 가져오거나 설정합니다. (Put)
put_EditingMode InkPicture 컨트롤이 잉크 모드, 삭제 모드 또는 선택/편집 모드에 있는지 여부를 지정하는 값을 가져오거나 설정합니다. (Put)
put_eDx InkTransform 개체가 나타내는 아핀 변환 행렬의 첫 번째 열인 세 번째 행에 있는 요소를 가져오거나 설정합니다. (Put)

put_eDy InkTransform 개체가 나타내는 아핀 변환 행렬의 세 번째 행 두 번째 열에 있는 요소를 가져오거나 설정합니다. (Put)
put_eM11 InkTransform 개체가 나타내는 아핀 변환 행렬의 첫 번째 열인 첫 번째 행의 요소를 가져오거나 설정합니다. (Put)
put_eM12 InkTransform 개체가 나타내는 아핀 변환 행렬의 첫 번째 행, 두 번째 열에 있는 요소를 가져오거나 설정합니다. (Put)
put_eM21 InkTransform 개체가 나타내는 아핀 변환 행렬의 첫 번째 열인 두 번째 행의 요소를 가져오거나 설정합니다. (Put)
put_eM22 InkTransform 개체가 나타내는 아핀 변환 행렬의 두 번째 열인 두 번째 행에 있는 요소를 가져오거나 설정합니다. (Put)
put_Enabled InkCollector 개체가 펜 입력(대기 패킷, 범위 이벤트의 커서 등)을 수집하는지 여부를 지정하는 값을 가져오거나 설정합니다. (Put)
put_Enabled InkOverlay 개체가 펜 입력(대기 패킷, 범위 이벤트의 커서 등)을 수집하는지 여부를 지정하는 값을 가져오거나 설정합니다. (Put)
put_Enabled InkPicture 컨트롤이 사용자 생성 이벤트에 응답할 수 있는지 여부를 결정하는 값을 가져오거나 설정합니다. (Put)
put_Enabled InkEdit 컨트롤이 사용자 생성 이벤트에 응답할 수 있는지 여부를 결정하는 값을 가져오거나 설정합니다. (Put)
put_Enabled 동적 렌더링을 켜고 끄는 값을 가져오거나 설정합니다. (Put)

put_Enabled 제스처 인식이 사용되는지 여부를 나타내는 값을 가져오거나 설정합니다. (Put)
put_Enabled RealTimeStylus 개체가 태블릿 펜 데이터를 수집하는지 여부를 지정하는 값을 가져오거나 설정합니다. (Put)
put_EnabledUnicodeRanges 인식기 컨텍스트에서 지원할 하나 이상의 유니코드 범위 집합을 가져오거나 설정합니다. (Put)
put_EraserMode 잉크가 스트로크 또는 포인트별로 지워지는지 여부를 지정하는 값을 가져오거나 설정합니다. (Put)
put_EraserMode 잉크가 스트로크 또는 포인트별로 지워지는지 여부를 지정하는 값을 가져오거나 설정합니다. (Put)
put_EraserWidth 지우개 펜 끝의 너비를 지정하는 값을 가져오거나 설정합니다. (Put)
put_EraserWidth 지우개 펜 팁의 너비를 지정하는 값을 가져오거나 설정합니다. (Put)
put_ExpandPostInsertionCorrection 태블릿 PC 입력 패널의 수정 빗이 자동으로 확장되는지 여부를 나타내는 값을 가져오거나 설정합니다. (Put)
put_Factoid 인식기가 인식 결과에 대한 검색을 제한하는 데 사용하는 factoid를 가져오거나 설정합니다. (Put)
put_Factoid 더 이상 사용되지 않습니다. PenInputPanel이 TIP(텍스트 입력 패널)로 대체되었습니다. PenInputPanel 개체에서 사용하는 factoid의 문자열 이름을 가져오거나 설정합니다. (Put)
put_Factoid IInkRecognizer 개체가 인식 결과에 대한 검색을 제한하는 데 사용하는 Factoid 상수를 가져오거나 설정합니다. (Put)

put_FitToCurve Bezier 다듬기를 사용하여 잉크를 렌더링할지 여부를 지정하는 값을 가져오거나 설정합니다. (Put)
put_FlicksEnabled 터치 제스처 인식이 사용되는지 나타냅니다.
put_GuideData 인식기에 대한 잉크의 경계를 나타내는 InkRecoGuide 구조를 가져오거나 설정합니다. (Put)
put_Height InkDrawingAttributes 개체를 사용하여 잉크를 그릴 때 펜의 높이를 가져오거나 설정합니다. (Put)
put_HorizontalOffset 더 이상 사용되지 않습니다. PenInputPanel이 TIP(텍스트 입력 패널)로 대체되었습니다. 펜 입력 패널의 왼쪽 가장자리와 펜이 연결된 컨트롤의 왼쪽 가장자리 사이의 오프셋을 가져오거나 설정합니다. (IPenInputPanel.put_HorizontalOffset)
put_hWnd 잉크가 그려지는 창의 핸들 값을 가져오거나 설정합니다. (IInkCollector.put_hWnd)
put_hWnd 잉크가 그려지는 창의 핸들 값을 가져오거나 설정합니다. (IInkOverlay.put_hWnd)
put_HWND DynamicRenderer Class 개체와 연결된 창 핸들 HWND를 가져오거나 설정합니다. (Put)
put_HWND RealTimeStylus 개체에서 사용하는 창과 연결된 핸들 값을 가져오거나 설정합니다. (Put)
put_IgnorePressure 태블릿 표면에서 펜 팁의 압력 증가로 그려진 잉크가 자동으로 더 넓어지는지 여부를 지정하는 값을 가져오거나 설정합니다. (Put)
put_InkEnabled InkPicture 컨트롤이 펜 입력(대기 패킷, 범위 이벤트의 커서 등)을 수집하는지 여부를 지정하는 값을 가져오거나 설정합니다. (Put)

put_InkInsertMode InkEdit 컨트롤에 잉크를 텍스트 또는 잉크로 삽입하는 방법을 지정하는 값을 가져오거나 설정합니다. (Put)
put_InkMode 잉크 컬렉션을 사용하지 않도록 설정하거나, 잉크를 수집하거나, 잉크 및 제스처를 수집할지 여부를 지정하는 값을 가져오거나 설정합니다. (Put)
put_InPlaceVisibleOnFocus 연결된 창이 포커스를 받을 때 태블릿 PC 입력 패널이 자동으로 표시되는지 여부를 나타내는 값을 가져오거나 설정합니다. (Put)
put_Left InkRectangle 개체의 왼쪽 위치를 가져오거나 설정합니다. (Put)
put_LineHeight HIMETRIC 단위로 예상되는 필기 높이를 가져오거나 설정합니다. (Put)
put_Locked InkEdit 컨트롤의 내용을 편집할 수 있는지 여부를 나타내는 값을 가져오거나 설정합니다. (Put)
put_MarginX 창 사각형 주위의 x축 여백을 화면 좌표로 가져오거나 설정합니다. 이 여백은 잉크 창의 가장자리 주위에 버퍼를 제공합니다. (IInkCollector.put_MarginX)
put_MarginX 창 사각형 주위의 x축 여백을 화면 좌표로 가져오거나 설정합니다. 이 여백은 잉크 창의 가장자리 주위에 버퍼를 제공합니다. (IInkOverlay.put_MarginX)
put_MarginX 창 사각형 주위의 x축 여백을 화면 좌표로 가져오거나 설정합니다. 이 여백은 잉크 창의 가장자리 주위에 버퍼를 제공합니다. (IInkPicture.put_MarginX)
put_MarginY 창 사각형 주위의 y축 여백을 화면 좌표로 가져오거나 설정합니다. 이 여백은 잉크 창의 가장자리 주위에 버퍼를 제공합니다. (IInkCollector.put_MarginY)

put_MarginY 창 사각형 주위의 y축 여백을 화면 좌표로 가져오거나 설정합니다. 이 여백은 잉크 창의 가장자리 주위에 버퍼를 제공합니다. (IInkOverlay.put_MarginY)
put_MarginY 창 사각형 주위의 y축 여백을 화면 좌표로 가져오거나 설정합니다. 이 여백은 잉크 창의 가장자리 주위에 버퍼를 제공합니다. (IInkPicture.put_MarginY)
put_MaxLength InkEdit 컨트롤에서 보유할 수 있는 최대 문자 수가 있는지 여부를 나타내는 값을 가져오거나 설정하며, 있는 경우 최대 문자 수를 지정합니다. (Put)
put_MaxStrokeCount 제스처 인식당 허용되는 최대 스트로크 수를 가져오거나 설정합니다. (Put)
put_Midline 중간선 높이를 가져오거나 설정합니다. 중간선 높이는 기준선에서 중간선까지의 거리이며, 그려진 상자의 거리입니다. (Put)
put_MouseIcon 사용자 지정 마우스 아이콘을 가져오거나 설정합니다. (IInkOverlay.put_MouseIcon)
put_MouseIcon 사용자 지정 마우스 아이콘을 가져오거나 설정합니다. (IInkPicture.put_MouseIcon)
put_MouseIcon InkEdit 컨트롤에 대한 사용자 지정 마우스 아이콘을 가져오거나 설정합니다. (Put)
put_MousePointer 표시되는 마우스 포인터의 형식을 나타내는 값을 가져오거나 설정합니다. (IInkCollector.put_MousePointer)
put_MousePointer 표시되는 마우스 포인터의 형식을 나타내는 값을 가져오거나 설정합니다. (IInkOverlay.put_MousePointer)

put_MousePointer 표시되는 마우스 포인터의 형식을 나타내는 값을 가져오거나 설정합니다. (IInkPicture.put_MousePointer)
put_MousePointer 표시할 마우스 포인터의 유형을 나타내는 값을 가져오거나 설정합니다. (Put)
put_MultiLine InkEdit 컨트롤이 여러 줄의 텍스트를 수락하고 표시할 수 있는지 여부를 나타내는 값을 가져오거나 설정합니다. (Put)
put_MultiTouchEnabled IRealTimeStylus3 개체에 다중 터치 입력을 사용할 수 있는지 여부를 나타냅니다. (Put)
put_PenTip 이 InkDrawingAttributes 개체와 연결된 잉크를 그릴 때 사용할 펜 팁을 나타내는 값을 가져오거나 설정합니다. (Put)
put_Picture InkPicture 컨트롤에 표시할 그래픽 파일을 가져옵니다. (Put)
put_PreferredInPlaceDirection 텍스트 입력 필드를 기준으로 현재 위치 입력 패널의 기본 방향을 가져오거나 설정합니다. (Put)
put_PrefixText InkRecognizerContext 개체의 InkStrokes 컬렉션 앞에 오는 문자를 가져오거나 설정합니다. (Put)
put_RasterOperation 펜과 배경의 색이 상호 작용하는 방식을 정의하는 값을 가져오거나 설정합니다. (Put)
put_RecognitionFlags 인식기가 잉크를 해석하고 결과 문자열을 결정하는 방법을 지정하는 플래그를 가져오거나 설정합니다. (Put)
put_RecognitionTimeout 수집된 마지막 IInkStrokeDisp 개체와 텍스트 인식 시작 사이의 시간 길이(밀리초)를 가져오거나 설정합니다. (Put)

put_Right InkRectangle 개체의 올바른 위치를 가져오거나 설정합니다. (Put)
put_Rows 인식 가이드의 행 수를 가져오거나 설정합니다. (Put)
put_ScrollBars InkEdit 컨트롤에 표시할 스크롤 막대의 형식(있는 경우)을 가져오거나 설정합니다. (Put)
put_SelAlignment InkEdit 컨트롤에서 단락의 맞춤을 제어하는 값을 가져오거나 설정합니다. (Put)
put_SelBold InkEdit 컨트롤에서 현재 선택한 텍스트의 글꼴 스타일이 굵게 표시되는지 여부를 지정하는 값을 가져오거나 설정합니다. (Put)
put_SelCharOffset InkEdit 컨트롤의 텍스트가 기준선(일반)에 표시되는지, 기준선 위의 위 첨자로 표시되는지 또는 기준선 아래의 아래 첨자로 표시되는지 여부를 결정하는 값을 반환하거나 설정합니다. (Put)
put_SelColor 현재 텍스트 선택 영역 또는 삽입 지점의 텍스트 색을 가져오거나 설정합니다(런타임에만 해당). (Put)
put_Selection InkOverlay 개체 또는 InkPicture 컨트롤 내에서 현재 선택된 InkStrokes 컬렉션을 가져오거나 설정합니다. (Put)
put_Selection InkOverlay 개체 또는 InkPicture 컨트롤 내에서 현재 선택된InkStrokes 컬렉션을 가져오거나 설정합니다. (Put)
put_SelFontName InkEdit 컨트롤 내에서 선택한 텍스트의 글꼴 이름을 가져오거나 설정합니다(런타임에만 해당). (Put)

put_SelFontSize InkEdit 컨트롤 내에서 선택한 텍스트의 글꼴 크기를 가져오거나 설정합니다(런타임에만 해당). (Put)
put_SelInks 현재 선택 영역에서 포함된 InkDisp 개체의 배열(잉크로 표시되는 경우)을 가져오거나 설정합니다. (Put)
put_SelInksDisplayMode 잉크와 텍스트 간에 선택 영역의 모양을 전환할 수 있는 값을 가져오거나 설정합니다. (Put)
put_SelItalic InkEdit 컨트롤에서 현재 선택한 텍스트의 글꼴 스타일이 기울임꼴인지 여부를 지정하는 값을 가져오거나 설정합니다(런타임에만 해당). (Put)
put_SelLength InkEdit 컨트롤에서 선택한 문자 수를 가져오거나 설정합니다(런타임에만 해당). (Put)
put_SelRTF InkEdit 컨트롤에서 현재 선택한 RTF(서식 있는 텍스트 서식) 텍스트를 가져오거나 설정합니다(런타임에만 해당). (Put)
put_SelStart InkEdit 컨트롤에서 선택한 텍스트의 시작점을 가져오거나 설정합니다(런타임에만 해당). (Put)
put_SelText InkEdit 컨트롤 내에서 선택한 텍스트를 가져오거나 설정합니다(런타임에만 해당). (Put)
put_SelUnderline InkEdit 컨트롤에서 현재 선택한 텍스트의 글꼴 스타일에 밑줄이 그어지는지 여부를 지정하는 값을 가져오거나 설정합니다(런타임에만 해당). (Put)
put_SizeMode InkPicture 컨트롤이 이미지 배치 및 크기 조정을 처리하는 방법을 가져오거나 설정합니다. (Put)
put_SuffixText InkRecognizerContext 개체의 InkStrokes 컬렉션 다음에 오는 문자를 가져오거나 설정합니다. (Put)

put_SupportHighContrastInk 시스템이 고대비 모드일 때 잉크가 하나의 색으로 렌더링되는지 여부를 지정하는 값을 가져오거나 설정합니다. (IInkCollector.put_SupportHighContrastInk)
put_SupportHighContrastInk 시스템이 고대비 모드일 때 잉크가 하나의 색으로 렌더링되는지 여부를 지정하는 값을 가져오거나 설정합니다. (IInkOverlay.put_SupportHighContrastInk)
put_SupportHighContrastInk 시스템이 고대비 모드일 때 잉크가 하나의 색으로 렌더링되는지 여부를 지정하는 값을 가져오거나 설정합니다. (IInkPicture.put_SupportHighContrastInk)
put_SupportHighContrastSelectionUI 시스템이 고대비 모드일 때 모든 선택 UI(사용자 인터페이스) 요소가 고대비로 그려지는지 여부를 지정하는 값을 가져오거나 설정합니다. (Put)
put_SupportHighContrastSelectionUI 시스템이 고대비 모드일 때 모든 선택 사용자 인터페이스(선택 경계 상자 및 선택 핸들)가 고대비로 그려지는지 여부를 지정하는 값을 가져오거나 설정합니다. (Put)
put_Text InkEdit 컨트롤에서 현재 텍스트를 가져오거나 설정합니다. (Put)
put_TextRTF 모든 RTF 코드를 포함하여 InkEdit 컨트롤의 텍스트를 가져오거나 설정합니다. (Put)
put_Top InkRectangle 개체의 위쪽 위치를 가져오거나 설정합니다. (Put)
put_Transparency 잉크의 투명도 값을 나타내는 값을 가져오거나 설정합니다. (Put)
put_UseMouseForInput 마우스를 입력 디바이스로 사용할 수 있는지 여부를 나타내는 값을 가져오거나 설정합니다. (Put)

put_VerticalOffset 더 이상 사용되지 않습니다. PenInputPanel이 TIP(텍스트 입력 패널)로 대체되었습니다. 펜 입력 패널의 가장 가까운 가로 가장자리와 펜이 연결된 컨트롤의 가장 가까운 가로 가장자리 사이의 오프셋을 가져오거나 설정합니다. (IPenInputPanel.put_VerticalOffset)
put_Width 잉크를 그릴 때 펜 팁의 y축 차원 또는 너비를 가져오거나 설정합니다. (Put)
put_WindowInputRectangle RealTimeStylus Class 개체의 창 입력 사각형을 가져오거나 설정합니다. (Put)
put_WritingBox 쓰기가 실제로 수행될 수 있는 인식 가이드의 보이지 않는 쓰기 영역을 가져오거나 설정합니다. (Put)
putref_DefaultDrawingAttributes 잉크를 그리거나 표시할 때 사용할 기본 그리기 특성을 가져오거나 설정합니다. (IInkCollector.putref_DefaultDrawingAttributes)
putref_DrawingAttributes 잉크가 그려질 때 잉크에 적용할 그리기 특성을 가져오거나 설정합니다.
putref_DrawingAttributes InkEdit 컨트롤에 아직 그려지지 않은 잉크의 그리기 특성을 가져오거나 설정합니다.
putref_Font InkEdit 컨트롤에 아직 그려지지 않은 텍스트의 글꼴을 가져오거나 설정합니다.
putref_Guide 잉크 입력에 사용할 InkRecognizerGuide를 가져오거나 설정합니다.
putref_Ink InkCollector 개체 또는 InkOverlay 개체와 연결된 InkDisp 개체를 가져오거나 설정합니다.
putref_MouseIcon 사용자 지정 마우스 아이콘을 가져오거나 설정합니다.

putref_MouseIcon InkEdit 컨트롤에 표시되는 마우스 아이콘을 가져오거나 설정합니다.
putref_Recognizer InkEdit 컨트롤에 사용되는 인식기를 가져오거나 설정합니다.
putref_RecognizerContext InkDivider 개체가 레이아웃 분석에 사용하는 InkRecognizerContext 개체를 가져오거나 설정합니다.
putref_Renderer 잉크를 그리는 데 사용되는 InkRenderer 개체를 가져오거나 설정합니다.
putref_Strokes InkRecognizerContext 개체와 연결된 InkStrokes 컬렉션을 가져오거나 설정합니다.
putref_Strokes InkDivider 개체가 레이아웃 분석을 수행하는 InkStrokes 컬렉션을 가져오거나 설정합니다.
putref_WordList 인식 결과를 개선하기 위해 인식 프로세스에 사용되는 단어 목록을 가져오거나 설정합니다.
RealTimeStylusDisabled 구현 플러그 인에 RTS(RealTimeStylus Class) 개체가 사용하지 않도록 설정되어 있음을 알릴 수 있습니다.
RealTimeStylusEnabled 구현 플러그 인에 RTS(RealTimeStylus Class) 개체가 사용하도록 설정되어 있음을 알릴 수 있습니다.
Recognize 예약되어 있습니다. (IInkLineInfo.Recognize)
Recognize InkStrokes 컬렉션에서 인식을 수행하고 인식 결과를 반환합니다. (IInkRecognizerContext.Recognize)

Recognize InkStrokes 컬렉션에서 인식을 수행하고 인식 결과를 반환합니다. (IInkEdit.Recognize)
반영 가로 또는 세로 방향으로 변환에 대한 리플렉션을 수행합니다.
새로 고침 Windows XP Tablet PC Edition에서는 새로 고침을 더 이상 사용할 수 없습니다.
새로 고침 InkEdit 컨트롤을 다시 그리도록 합니다.
새로 고침 DynamicRenderer Class 개체가 현재 렌더링 중인 잉크 데이터를 다시 그리도록 합니다.
ReleaseCachedData DynamicRenderer 클래스가 보유한 임시 데이터에서 지정된 스트로크 데이터를 해제합니다.
ReleaseLock 지정된 잠금을 해제합니다.
Remove IInkCustomStrokes 컬렉션에서 InkStrokes 컬렉션을 제거합니다.
Remove IInkExtendedProperties 컬렉션에서 IInkExtendedProperty 개체를 제거합니다.
Remove InkStrokes 컬렉션에서 IInkStrokeDisp 개체를 제거합니다.
RemoveAllStylusAsyncPlugins 비동기 플러그 인 컬렉션에서 모든 플러그 인을 제거합니다.
RemoveAllStylusSyncPlugins 동기 플러그 인 컬렉션에서 모든 플러그 인을 제거합니다.

RemoveFunctionName 인식기가 허용하는 사용자 지정 수학 함수 목록에서 함수 이름 정의를 제거합니다.
RemoveRecognitionResult InkStrokes 컬렉션과 연결된 RecognitionResult를 제거합니다.
RemoveStrokes 컬렉션에서 스트로크를 제거합니다.
RemoveStylusAsyncPlugin 비동기 플러그 인 컬렉션에서 지정된 인덱스를 사용하여 IStylusAsyncPlugin을 제거하고 필요에 따라 반환합니다.
RemoveStylusSyncPlugin 지정된 인덱스의 컬렉션에서 IStylusSyncPlugin을 제거합니다.
RemoveWord InkWordList에서 한 단어를 제거합니다.
Reset 변환을 기본 상태인 ID 변환으로 다시 설정합니다.
Reset GestureRecognizer 클래스 개체에서 과거 스트로크 정보를 삭제합니다.
ResetContext 컨텍스트에서 현재 잉크 및 인식 결과를 삭제합니다. 인식기 컨텍스트의 현재 설정이 유지됩니다.
ResultByType IInkDivisionUnits 컬렉션에 대한 분석 결과의 요청된 구조 단위를 가져옵니다.
회전 InkRenderer의 뷰 변환에 회전을 적용합니다.
회전 회전의 중심점을 중심으로 각도를 사용하여 잉크를 회전합니다. (IInkStrokeDisp.Rotate)

회전 회전의 중심점을 중심으로 각도를 사용하여 잉크를 회전합니다. (IInkStrokes.Rotate)
회전 InkTransform 개체의 회전 계수와 필요에 따라 회전의 중심점을 변경하기 위해 각도로 측정된 양을 변경합니다.
저장 잉크를 지정된 InkPersistenceFormat으로 변환하고, 지정된 InkPersistenceCompressionMode를 사용하여 잉크를 저장하고, 바이트 배열에 이진 데이터를 반환합니다.
ScaleToRectangle 지정된 InkRectangle 개체에 맞게 IInkStrokeDisp 개체 또는 InkStrokes 컬렉션의 크기를 조정합니다. (IInkStrokeDisp.ScaleToRectangle)
ScaleToRectangle 지정된 InkRectangle 개체에 맞게 IInkStrokeDisp 개체 또는 InkStrokes 컬렉션의 크기를 조정합니다. (IInkStrokes.ScaleToRectangle)
ScaleTransform X 및 Y 차원의 뷰 변환 크기를 조정합니다.
ScaleTransform 지정된 가로 및 세로 요소를 변환 또는 잉크에 적용합니다. (IInkStrokeDisp.ScaleTransform)
ScaleTransform 지정된 가로 및 세로 요소를 변환 또는 잉크에 적용합니다. (IInkStrokes.ScaleTransform)
ScaleTransform 지정된 가로 및 세로 요소를 변환 또는 잉크에 적용합니다. (IInkTransform.ScaleTransform)
SetAllTabletsMode 잉크 수집기(InkCollector, InkOverlay 또는 InkPicture)가 태블릿 PC에 연결된 모든 태블릿에서 잉크를 수집할 수 있습니다. (IInkCollector.SetAllTabletsMode)
SetAllTabletsMode 잉크 수집기(InkCollector, InkOverlay 또는 InkPicture)가 태블릿 PC에 연결된 모든 태블릿에서 잉크를 수집할 수 있습니다. (IInkOverlay.SetAllTabletsMode)

SetAllTabletsMode 잉크 수집기(InkCollector, InkOverlay 또는 InkPicture)가 태블릿 PC에 연결된 모든 태블릿에서 잉크를 수집할 수 있습니다. (IInkPicture.SetAllTabletsMode)
SetAllTabletsMode RealTimeStylus 클래스 개체의 모드를 설정하여 모든 디지타이저에서 데이터를 수집합니다.
SetCACMode 문자 또는 단어 인식에 대한 문자 자동 완성 모드를 지정합니다. 문자 자동 완성을 설정한 후에는 해제할 수 없습니다.
SetCandidate 기존 대체 항목을 바꾸거나 목록에 대체 항목을 추가하여 인식 결과 목록에서 하나의 인식 대체를 업데이트.
SetCaptionText 창을 만들 때 컨트롤의 캡션으로 사용할 문자열을 수정합니다.
SetContextPropertyValue 인식기 컨텍스트에 속성을 추가합니다. 속성이 이미 있는 경우 해당 값이 수정됩니다.
SetCustomPaint 단추 또는 배경에 사용자 지정 그리기가 있는지 여부를 결정합니다.
SetDesiredPacketDescription 패킷 스트림에 포함할 속성을 요청합니다.
SetEnabledUnicodeRanges 컨텍스트에서 하나 이상의 유니코드 지점 범위를 사용하도록 설정합니다.
SetEventInterest 개체 또는 컨트롤이 지정된 이벤트에 관심이 있는지 여부를 나타내는 값을 수정합니다. (IInkCollector.SetEventInterest)
SetEventInterest 개체 또는 컨트롤이 지정된 이벤트에 관심이 있는지 여부를 나타내는 값을 설정합니다.

SetEventInterest 개체 또는 컨트롤이 지정된 이벤트에 관심이 있는지 여부를 나타내는 값을 수정합니다. (IInkPicture.SetEventInterest)
SetFactoid 인식기가 결과에 대한 검색을 제한하는 데 사용하는 factoid를 지정합니다. 입력 필드에 날짜가 포함된 경우와 같이 입력 필드가 알려진 형식인 경우 factoid를 지정합니다.
SetFlags 인식기가 잉크를 해석하고 결과 문자열을 결정하는 방법을 지정합니다. 잉크를 처음으로 처리하기 전에 이 함수를 호출합니다. 따라서 Process 함수를 호출하기 전에 SetFlags 함수를 호출합니다.
SetFormat 텍스트 잉크 개체(tInk)에서 설정할 표시 속성을 지정합니다.
SetGestureStatus 알려진 제스처에서 개체 또는 컨트롤의 관심사를 수정합니다. (IInkCollector.SetGestureStatus)
SetGestureStatus 알려진 제스처에서 개체 또는 컨트롤의 관심사를 설정합니다.
SetGestureStatus 알려진 제스처에서 개체 또는 컨트롤의 관심사를 수정합니다. (IInkPicture.SetGestureStatus)
SetGestureStatus 알려진 애플리케이션 제스처에서 InkEdit 컨트롤의 관심을 수정합니다.
SetGuide 박스형 또는 줄 지어 입력에 사용할 인식 가이드를 설정합니다. 컨텍스트에 스트로크를 추가하기 전에 이 함수를 호출해야 합니다.
SetInPlaceHoverTargetPosition 태블릿 PC 입력 패널 가리키기 대상을 화면 좌표로 명시적으로 배치합니다.
SetInPlacePosition 태블릿 PC 입력 패널을 화면 좌표로 명시적으로 배치합니다.

SetInPlaceVisibility 태블릿 PC 입력 패널을 표시하거나 숨깁니다.
SetObjectTransform 잉크를 렌더링하는 데 사용되는 개체 변환을 나타내는 InkTransform 개체를 설정합니다.
SetOwnerWindow 이 컨트롤을 소유하는 창을 수정합니다.
SetPacketValuesByProperty 특정 속성의 패킷 값을 수정합니다.
설정값 X, Y 값의 배열을 사용하여 IInkStrokeDisp의 점을 설정합니다.
SetPosition 컨트롤의 위치와 크기를 수정합니다.
SetPreviewHeight 미리 보기 영역 높이를 픽셀 단위로 수정합니다.
SetRectangle 단일 호출에서 InkRectangle 개체의 요소를 설정합니다.
SetResultOnStrokes 결과를 만드는 데 사용된 스트로크에 인식 결과를 할당합니다.
SetSingleTabletIntegratedMode 잉크 수집기(InkCollector, InkOverlay 또는 InkPicture)가 하나의 태블릿에서만 잉크를 수집할 수 있습니다. 다른 태블릿의 잉크는 잉크 수집기에서 무시됩니다. (IInkCollector.SetSingleTabletIntegratedMode)
SetSingleTabletIntegratedMode 잉크 수집기(InkCollector, InkOverlay 또는 InkPicture)가 하나의 태블릿에서만 잉크를 수집할 수 있습니다. 다른 태블릿의 잉크는 잉크 수집기에서 무시됩니다. (IInkOverlay.SetSingleTabletIntegratedMode)

SetSingleTabletIntegratedMode 잉크 수집기(InkCollector, InkOverlay 또는 InkPicture)가 하나의 태블릿에서만 잉크를 수집할 수 있습니다. 다른 태블릿의 잉크는 잉크 수집기에서 무시됩니다. (IInkPicture.SetSingleTabletIntegratedMode)
SetSingleTabletMode RtS(RealTimeStylus Class) 개체의 모드를 수정하여 태블릿 PC에 연결된 디지타이저를 나타내는 태블릿 개체 하나만 입력을 수집합니다. 다른 디지타이저의 스타일러스 입력은 RealTimeStylus에서 무시됩니다.
SetTextContext 인식기 컨텍스트에 포함된 텍스트 앞과 뒤의 텍스트 문자열을 제공합니다. 잉크를 처음으로 처리하기 전에 이 함수를 호출합니다. 따라서 Process 함수를 호출하기 전에 SetTextContext 함수를 호출합니다.
SetTransform InkTransform 멤버 데이터를 수정합니다.
SetViewTransform 잉크를 렌더링하는 데 사용되는 뷰 변환을 나타내는 InkTransform 개체를 설정합니다.
SetWindowInputRectangle 잉크가 그려지는 창 사각형을 픽셀 단위로 설정합니다. (IInkCollector.SetWindowInputRectangle)
SetWindowInputRectangle 잉크가 그려지는 창 사각형을 픽셀 단위로 설정합니다. (IInkOverlay.SetWindowInputRectangle)
SetWindowInputRectangle 잉크가 그려지는 창 사각형을 픽셀 단위로 수정합니다.
SetWordList 인식할 현재 인식기 컨텍스트의 단어 목록을 설정합니다.
전단 스트로크 또는 스트로크의 잉크를 지정된 가로 및 세로 요소로 전단합니다. (IInkStrokeDisp.Shear)
전단 스트로크 또는 스트로크의 잉크를 지정된 가로 및 세로 요소로 전단합니다. (IInkStrokes.Shear)

전단 지정된 가로 및 세로 요소에 따라 InkTransform의 전단을 조정합니다.
표시 컨트롤을 표시합니다.
Split 스트로크의 지정된 위치에서 스트로크를 분할합니다.
StopBackgroundRecognition BackgroundRecognize 또는 BackgroundRecognizeWithAlternates 호출로 시작된 백그라운드 인식을 종료합니다.
StylusButtonDown 구현 플러그 인에 사용자가 스타일러스 단추를 누르고 있음을 알립니다.
StylusButtonUp 구현 플러그 인에 사용자가 스타일러스 단추를 해제했음을 알립니다.
스타일러스다운 구현 플러그 인에 태블릿 펜이 디지타이저 표면에 닿은 것을 알립니다.
StylusInRange 스타일러스가 디지타이저의 검색 범위에 진입하고 있음을 구현 플러그 인에 알릴 수 있습니다.
StylusOutOfRange 스타일러스가 디지타이저의 검색 범위를 떠났다는 것을 구현 플러그 인에 알릴 수 있습니다.
StylusUp 사용자가 태블릿 디지타이저 표면에서 태블릿 펜을 올렸다는 것을 구현 플러그 인에 알립니다.
SystemEvent 구현 플러그 인에 시스템 이벤트를 사용할 수 있음을 알 수 있습니다.
TabletAdded ITablet 개체가 시스템에 연결되면 구현 플러그 인을 알 수 있습니다.

TabletRemoved ITablet 개체가 시스템에서 제거될 때 구현 플러그 인에 알 수 있습니다.
TextInserted 태블릿 PC 입력 패널이 입력 포커스가 있는 컨트롤에 텍스트를 삽입한 경우에 발생합니다. 사용자가 입력 패널에 입력한 잉크에 대한 액세스를 제공합니다.
TextInserting 태블릿 PC 입력 패널이 입력 포커스를 사용하여 컨트롤에 텍스트를 삽입하려고 할 때 발생합니다. 사용자가 입력 패널에 입력한 잉크에 대한 액세스를 제공합니다.
ToString ToString은 Windows Vista를 기준으로 더 이상 사용할 수 없습니다.
변환 IInkStrokeDisp 개체 또는 변환의 크기 조정, 회전, 변환 및 조합을 나타낼 수 있는 InkStrokes 컬렉션에 선형 변환을 적용합니다. (IInkStrokeDisp.Transform)
변환 IInkStrokeDisp 개체 또는 변환의 크기 조정, 회전, 변환 및 조합을 나타낼 수 있는 InkStrokes 컬렉션에 선형 변환을 적용합니다. (IInkStrokes.Transform)
번역 변환에 변환을 적용합니다.
Unadvise ITextInputPanel::Advise 메서드를 통해 이전에 설정된 권고 연결을 종료합니다.
UpdateMapping dpi 또는 방향과 같은 속성을 표시할 때 플러그 인에 알림을 표시합니다.

인터페이스

_IInkEditEvents

_IMathInputControlEvents 수학 입력 컨트롤 이벤트 처리기를 노출합니다.
IDynamicRenderer RealTimeStylus Class 개체에서 데이터를 처리할 때 태블릿 펜 데이터를 실시간으로 표시합니다.
IGestureRecognizer 제스처를 인식하고 입력 큐에 제스처 데이터를 추가하여 이벤트에 반응합니다.
IHandwrittenTextInsertion 애플리케이션의 사용자 지정 텍스트 입력 코드에서 텍스트 필드와 Text Services 백업 저장소 모두에 텍스트를 삽입하는 데 사용됩니다.
IInkCollector . (IInkCollector)
IInkCursor 태블릿 커서에 대한 일반 정보를 나타냅니다.
IInkCursorButton 디바이스를 가리키고 선택하는 태블릿의 단추에 대한 일반적인 정보를 나타냅니다.
IInkCursorButtons IInkCursor 인터페이스에 대한 IInkCursorButton 개체의 컬렉션을 나타냅니다.
IInkCursors IInkCursor 개체의 컬렉션을 나타냅니다.
IInkCustomStrokes 사용자 정의 InkStrokes 컬렉션의 컬렉션을 포함합니다.
IInkDisp . (IInkDisp)
IInkDivider . (IInkDivider)

IInkDivisionResult InkDivider 개체에 포함된 스트로크 컬렉션의 레이아웃 분석을 나타냅니다.
IInkDivisionUnit IInkDivisionResult 개체 내의 단일 구조 요소를 나타냅니다.
IInkDivisionUnits IInkDivisionResult 개체에 포함된 IInkDivisionUnit 개체의 컬렉션을 포함합니다.
IInkDrawingAttributes . (IInkDrawingAttributes)
IInkEdit . (IInkEdit)
IInkExtendedProperties 애플리케이션 정의 데이터를 포함하는 IInkExtendedProperty 개체의 컬렉션을 나타냅니다.
IInkExtendedProperty 태블릿 PC 개체 모델 내의 다양한 개체에 사용자 고유의 데이터를 추가하는 기능을 나타냅니다.
IInkGesture 제스처 인식에서 반환된 제스처의 특정 속성을 쿼리하는 기능을 나타냅니다.
IInkLineInfo IInkLineInfo 인터페이스는 텍스트 잉크 개체(tInk)의 표시 속성 및 인식 결과 목록에 대한 액세스를 제공합니다.
IInkOverlay . (IInkOverlay)
IInkPicture . (IInkPicture)
IInkRecognitionAlternate 인식기 사전과 비교되는 잉크 세그먼트에 대해 가능한 단어 일치를 나타냅니다.

IInkRecognitionAlternates 잉크 세그먼트에 대해 가능한 단어 일치를 나타내는 IInkRecognitionAlternate 개체를 포함합니다.
IInkRecognitionResult 인식의 결과를 나타냅니다. 필기 잉크를 인식한 결과는 IInkRecognitionResult 개체에 반환됩니다.
IInkRecognizer 잉크 또는 필기를 처리하고 스트로크를 텍스트 또는 제스처로 변환하는 기능을 나타냅니다. 인식기는 실제 필기 인식을 수행하는 데 사용되는 InkRecognizerContext 개체를 만듭니다.
IInkRecognizer2 IInkWordList2 인터페이스에 멤버를 추가합니다.
IInkRecognizerContext . (IInkRecognizerContext)
IInkRecognizerContext2 InkRecognizerContext 클래스에 멤버를 추가합니다.
IInkRecognizerGuide . (IInkRecognizerGuide)
IInkRecognizers . (IInkRecognizers)
IInkRectangle . (IInkRectangle)
IInkRenderer . (IInkRenderer)
IInkStrokeDisp 단일 잉크 스트로크를 나타냅니다. 스트로크는 디지타이저가 알려진 잉크 표시의 좌표와 속성을 나타내는 캡처하는 속성 및 점 데이터 집합입니다.
IInkStrokes . (IInkStrokes)

IInkTablet 태블릿 디바이스 메시지 또는 이벤트를 수신하는 태블릿 PC의 디지타이저 디바이스를 나타냅니다.
IInkTablet2 IInkTablet 인터페이스를 확장합니다.
IInkTablet3 입력에 대한 멀티 터치 쿼리를 사용하도록 설정합니다.
IInkTablets . (IInkTablets)
IInkTransform . (IInkTransform)
IInkWordList . (IInkWordList)
IInkWordList2 InkWordList 클래스에 멤버를 추가합니다.
IMathInputControl 잉크 입력을 해석된 수학 출력으로 변환하는 메서드를 노출합니다.
IPenInputPanel . (IPenInputPanel)
IRealTimeStylus 디지타이저의 스타일러스 패킷 데이터를 실시간으로 처리합니다.
IRealTimeStylus2 IRealTimeStylus 인터페이스를 확장합니다.
IRealTimeStylus3 IRealTimeStylus3 인터페이스를 사용하면 실시간 스타일러스에 멀티 터치를 사용할 수 있습니다.

IRealTimeStylusSynchronization RealTimeStylus Class 개체에 대한 액세스를 동기화합니다.
IStrokeBuilder 인터페이스를 사용하여 패킷 데이터에서 스트로크를 프로그래밍 방식으로 만듭니다.
IStylusAsyncPlugin RealTimeStylus Class 개체의 비동기 플러그 인 컬렉션에 추가할 수 있는 비동기 플러그 인을 나타냅니다.
IStylusPlugin 해당 이벤트에 따라 사용자 지정 처리를 수행할 수 있도록 RealTimeStylus 클래스 이벤트의 알림을 받습니다.
IStylusSyncPlugin RealTimeStylus Class 개체의 동기 플러그 인 컬렉션에 추가할 수 있는 동기 플러그 인을 나타냅니다.
ITextInputPanel 태블릿 PC 입력 패널의 모양과 동작을 제어합니다.
ITextInputPanelEventSink ITextInputPanel 인터페이스 이벤트를 처리하는 메서드를 정의합니다.
ITextInputPanelRunInfo 텍스트 입력 패널이 현재 실행 중인지 확인하는 메서드를 제공합니다.

구조체

CHARACTER_RANGE 유니코드 점(문자) 범위를 지정합니다.
FLICK_DATA 펜 플릭에 대한 정보를 포함합니다.

FLICK_POINT 펜 터치에 대한 정보를 제공합니다.
IEC_GESTUREINFO 특정 제스처에 대한 정보를 포함합니다.
IEC_RECOGNITIONRESULTINFO IInkRecognitionResult 인터페이스 개체에 대한 정보를 포함합니다.
IEC_STROKEINFO Stroke 이벤트에 대한 정보를 포함합니다.
INKMETRIC 텍스트 잉크 개체(tInk)의 표시 속성을 지정합니다.
InkRecoGuide 더 이상 사용되지 않습니다. 인식 가이드에 대한 정보를 나타냅니다. 대신 WritingBox 속성, DrawnBox 속성, Rows 속성, Columns 속성 및 Midline 속성 [InkRecognizerGuide 클래스] 속성을 사용합니다.
LATTICE_METRICS 기준선 및 중간선 높이에 대해 설명합니다.
LINE_SEGMENT 기준선 또는 중간선과 같은 선 인식 세그먼트의 시작점과 끝점을 설명합니다.
PACKET_DESCRIPTION 특정 태블릿 인식기 컨텍스트에 대한 패킷의 내용을 설명합니다. 이 구조를 사용하여 패킷에 포함된 데이터에 액세스하지 마세요. 이 구조체는 패킷의 내용을 설명합니다.
PACKET_PROPERTY 태블릿 드라이버에서 보고하는 패킷 속성을 설명합니다.
PROPERTY_METRICS 패킷 속성의 범위와 해상도를 정의합니다.

RECO_ATTRS 인식기의 특성을 검색하거나 설치된 인식기를 검색할 때 사용할 특성을 지정합니다.
RECO_GUIDE 인식기로 잉크의 경계를 정의합니다.
RECO_LATTICE 격자의 진입점 역할을 합니다.
RECO_LATTICE_COLUMN 격자의 열을 나타냅니다.
RECO_LATTICE_ELEMENT 일반적으로 한 단어 또는 한 동아시아 문자에 해당합니다. 그러나 요소는 제스처, 도형 또는 기타 코드에 해당할 수도 있습니다.
RECO_LATTICE_PROPERTIES 속성 구조에 대한 포인터 배열을 포함합니다.
RECO_LATTICE_PROPERTY 격자에 사용되는 속성을 포함합니다.
RECO_RANGE 구조체는 사용되지 않습니다.
STROKE_RANGE InkDisp 개체의 스트로크 범위를 지정합니다.
StylusInfo 스타일러스에 대한 정보를 제공합니다.
SYSTEM_EVENT_DATA 태블릿 시스템 이벤트에 대한 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

TAPI 2.2

아티클2023. 03. 09.

TAPI 2.2 기술에 대한 개요입니다.

TAPI 2.2를 개발하려면 다음 헤더가 필요합니다.

- tapi.h
- tspi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 전화 통신 애플리케이션 프로그래밍 인터페이스

작업 Scheduler

아티클2023. 03. 16.

작업 스케줄러 기술 개요입니다.

작업 스케줄러를 개발하려면 다음 헤더가 필요합니다.

- mstask.h
- taskschd.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 작업 Scheduler

열거형

TASK_ACTION_TYPE 태스크가 수행할 수 있는 작업 유형을 정의합니다.
TASK_COMPATIBILITY 태스크 스케줄러 또는 태스크가 호환되는 AT 명령의 버전을 정의합니다.
TASK_CREATION 작업 스케줄러 서비스가 작업을 만들거나 업데이트하거나 사용하지 않도록 설정하는 방법을 정의합니다.
TASK_ENUM_FLAGS 작업 스케줄러가 등록된 작업을 통해 열거하는 방법을 정의합니다.
TASK_INSTANCES_POLICY 작업 스케줄러가 작업의 새 인스턴스를 시작할 때 작업의 기존 인스턴스를 처리하는 방법을 정의합니다.
TASK_LOGON_TYPE 작업을 실행하는 데 필요한 로그온 기술을 정의합니다.
TASK_PROCESSTOKENSID_TYPE 태스크에서 사용할 수 있는 SID(프로세스 보안 식별자) 유형을 정의합니다.

TASK_RUN_FLAGS 태스크 실행 방법을 정의합니다.
TASK_RUNLEVEL_TYPE 태스크가 실행될 권한 수준으로 지정하는 LUA 권한 상승 플래그를 정의합니다.
TASK_SESSION_STATE_CHANGE_TYPE 시작할 작업을 트리거하는 데 사용할 수 있는 터미널 서버 세션 상태 변경 종류를 정의합니다.
TASK_STATE 등록된 작업이 있을 수 있는 여러 상태를 정의합니다.
TASK_TRIGGER_TYPE 작업과 연결된 트리거 유형을 정의합니다.
TASK_TRIGGER_TYPE2 태스크에서 사용할 수 있는 트리거 유형을 정의합니다.
작업 페이지 검색할 작업 페이지의 유형을 정의합니다.

함수

활성화 Activate 메서드는 지정된 작업 항목에 대한 활성 인터페이스를 반환합니다.
AddRequiredPrivilege 작업 프로세스 토큰에 필요한 권한을 추가합니다.
AddWorkItem AddWorkItem 메서드는 태스크 일정에 작업을 추가합니다.
지우기 컬렉션에서 모든 작업을 지웁니다.

지우기 이름-값 쌍의 전체 컬렉션을 지웁니다.
지우기 컬렉션에서 모든 트리거를 지웁니다.
복제 현재 열거형과 동일한 열거형 상태를 포함하는 새 열거형 개체를 만듭니다.
연결 원격 컴퓨터에 연결하고 이 인터페이스의 모든 후속 호출을 원격 세션과 연결합니다.
만들기 컬렉션에 새 작업을 만들고 추가합니다.
만들기 컬렉션에 이름-값 쌍을 만듭니다.
만들기 작업에 대한 새 트리거를 만듭니다.
CreateFolder 관련 작업에 대한 폴더를 만듭니다.
CreateTrigger 작업 항목에 대한 트리거를 만듭니다.
삭제 Delete 메서드는 작업을 삭제합니다.
DeleteFolder 부모 폴더에서 하위 폴더를 삭제합니다.
DeleteTask 폴더에서 작업을 삭제합니다.

DeleteTrigger 작업 항목에서 트리거를 삭제합니다.
EditWorkItem 작업 항목의 작업, 일정 및 설정 속성 페이지를 표시하여 사용자가 해당 페이지에서 속성을 설정할 수 있도록 합니다.
Enum Enum 메서드는 현재 작업 폴더의 작업을 열거하는 OLE 열거자 개체에 대한 포인터를 검색합니다.
get__NewEnum 작업 컬렉션의 컬렉션 열거자를 가져옵니다.
get__NewEnum 레지스터 작업 컬렉션의 컬렉션 열거자를 가져옵니다.
get__NewEnum 실행 중인 작업 컬렉션에 대한 컬렉션 열거자를 가져옵니다.
get__NewEnum 폴더 컬렉션의 컬렉션 열거자를 가져옵니다.
get__NewEnum 이름-값 쌍 컬렉션의 컬렉션 열거자를 가져옵니다.
get__NewEnum 트리거 컬렉션의 컬렉션 열거자를 가져옵니다.
get_Actions 태스크에서 수행하는 작업 컬렉션을 가져오거나 설정합니다. (Get)
get_AllowDemandStart 실행 명령 또는 상황에 맞는 메뉴를 사용하여 작업을 시작할 수 있음을 나타내는 부울 값을 가져오거나 설정합니다. (Get)

get_AllowHardTerminate TerminateProcess를 사용하여 작업 스케줄러 서비스에서 작업을 종료할 수 있음을 나타내는 부울 값을 가져오거나 설정합니다. (Get)
get_Arguments 명령줄 작업과 연결된 인수를 가져오거나 설정합니다. (Get)
get_Attachments 전자 메일 메시지와 함께 전송되는 첨부 파일 배열에 대한 포인터를 가져오거나 설정합니다. (Get)
get_Author 작업의 작성자를 가져오거나 설정합니다. (Get)
get_Bcc 전자 메일 메시지에서 숨은 참조할 전자 메일 주소 또는 주소를 가져오거나 설정합니다. (Get)
get_Body 전자 메일 메시지가 포함된 전자 메일의 본문을 가져오거나 설정합니다. (Get)
get_Cc 전자 메일 메시지에서 참조하려는 전자 메일 주소 또는 주소를 가져오거나 설정합니다. (Get)
get_ClassId 처리기 클래스의 식별자를 가져오거나 설정합니다. (Get)
get_Compatibility 태스크가 호환되는 작업 스케줄러의 버전을 나타내는 정수 값을 가져오거나 설정합니다. (Get)
get_Connected 작업 스케줄러 서비스에 연결되어 있는지 여부를 나타내는 부울 값을 가져옵니다.
get_ConnectedDomain TargetServer 컴퓨터가 연결된 도메인의 이름을 가져옵니다.
get_ConnectedUser 작업 스케줄러 서비스에 연결된 사용자의 이름을 가져옵니다.

get_Context 작업에 대한 보안 주체의 식별자를 가져오거나 설정합니다. (Get)
get_Count 컬렉션의 작업 수를 가져옵니다.
get_Count 컬렉션에 등록된 작업 수를 가져옵니다.
get_Count 컬렉션에서 실행 중인 작업 수를 가져옵니다.
get_Count 컬렉션의 폴더 수를 가져옵니다.
get_Count 컬렉션의 이름-값 쌍 수를 가져옵니다.
get_Count 컬렉션의 트리거 수를 가져옵니다.
get_CurrentAction 실행 중인 작업이 수행하는 현재 작업의 이름을 가져옵니다.
get_Data 처리기와 연결된 추가 데이터를 가져오거나 설정합니다. (Get)
get_Data 작업과 연결된 데이터를 가져오거나 설정합니다. (Get)
get_Date 작업이 등록된 날짜와 시간을 가져오거나 설정합니다. (Get)
get_DaysInterval 일정에서 일 사이의 간격을 가져오거나 설정합니다. (Get)

get_DaysOfMonth 작업이 실행되는 월의 일을 가져오거나 설정합니다. (Get)
get_DaysOfWeek 작업이 실행되는 요일을 가져오거나 설정합니다. (Get)
get_DaysOfWeek 작업이 실행되는 요일을 가져오거나 설정합니다. (Get)
get_Deadline 정기적인 자동 유지 관리 중에 태스크를 완료하지 못한 경우 작업 스케줄러가 긴급 자동 유지 관리 중에 작업을 실행하려고 시도하는 시간을 가져오거나 설정합니다. (Get)
get_Definition 태스크의 정의를 가져옵니다.
get_Delay 시스템이 부팅된 시간과 작업이 시작된 시점 사이의 시간을 나타내는 값을 가져오거나 설정합니다. (Get)
get_Delay 이벤트가 발생한 시간과 태스크가 시작된 시점 사이의 시간을 나타내는 값을 가져오거나 설정합니다. (Get)
get_Delay 사용자가 로그온한 시간과 작업이 시작된 시점 사이의 시간을 나타내는 값을 가져오거나 설정합니다. (Get)
get_Delay 작업이 등록된 시간과 작업이 시작된 시점 사이의 시간을 가져오거나 설정합니다. (Get)
get_Delay 터미널 서버 세션 상태 변경이 감지된 후 작업을 시작하기 전에 지연이 발생하는 시간을 나타내는 값을 가져오거나 설정합니다. (ISessionStateChangeTrigger.get_Delay)
get_DeleteExpiredTaskAfter 작업 스케줄러가 만료된 후 작업을 삭제하기 전에 대기하는 시간을 가져오거나 설정합니다. (Get)

get_Description 작업에 대한 설명을 가져오거나 설정합니다. (Get)
get_DisallowStartIfOnBatteries 컴퓨터가 배터리에서 실행되는 경우 작업이 시작되지 않음을 나타내는 부울 값을 가져오거나 설정합니다. (Get)
get_DisallowStartOnRemoteAppSession RAIL(원격 애플리케이션 통합 로컬) 세션에서 실행되도록 트리거되는 경우 작업이 시작되지 않도록 지정하는 부울 값을 가져오거나 설정합니다. (Get)
get_DisplayName 보안 주체의 이름을 가져오거나 설정합니다. (Get)
get_Documentation 작업에 대한 추가 설명서를 가져오거나 설정합니다. (Get)
get_Duration 패턴이 반복되는 기간을 가져오거나 설정합니다. (Get)
get_Enabled 등록된 작업이 사용되는지 여부를 나타내는 부울 값을 가져오거나 설정합니다. (Get)
get_Enabled 작업이 사용하도록 설정되어 있음을 나타내는 부울 값을 가져오거나 설정합니다. 이 설정이 True인 경우에만 작업을 수행할 수 있습니다.
get_Enabled 트리거가 사용되는지 여부를 나타내는 부울 값을 가져오거나 설정합니다. (Get)
get_EndBoundary 트리거가 비활성화된 날짜 및 시간을 가져오거나 설정합니다. (Get)
get_EnginePID 작업을 실행하는 엔진(프로세스)의 프로세스 ID를 가져옵니다.

get_Exclusive 자동 유지 관리 중에 작업 스케줄러가 단독 모드에서 작업을 시작해야 하는지 여부를 나타냅니다. (Get)
get_ExecutionTimeLimit 작업을 완료할 수 있는 시간을 가져오거나 설정합니다. (Get)
get_ExecutionTimeLimit 이 트리거에서 시작한 작업을 실행할 수 있는 최대 시간을 가져오거나 설정합니다. (Get)
get_From 전자 메일을 보낼 전자 메일 주소를 가져오거나 설정합니다. (Get)
get_GroupId 보안 주체와 연결된 작업을 실행하는 데 필요한 사용자 그룹의 식별자를 가져오거나 설정합니다. (Get)
get_HeaderFields 보낼 전자 메일 메시지의 헤더 정보를 가져오거나 설정합니다. (Get)
get_Hidden 작업이 UI에 표시되지 않음을 나타내는 부울 값을 가져오거나 설정합니다. (Get)
get_HighestVersion 컴퓨터에서 지원하는 작업 스케줄러의 최고 버전을 나타냅니다.
get_Id 작업의 식별자를 가져오거나 설정합니다. (Get)
get_Id 네트워크 프로필을 식별하는 GUID 값을 가져오거나 설정합니다. (Get)
get_Id 보안 주체의 식별자를 가져오거나 설정합니다. (Get)
get_Id 트리거의 식별자를 가져오거나 설정합니다. (Get)

get_IdleDuration 작업을 실행하기 전에 컴퓨터가 유휴 상태여야 하는 시간을 나타내는 값을 가져오거나 설정합니다. (Get)
get_IdleSettings 컴퓨터가 유휴 상태일 때 작업 스케줄러가 작업을 수행하는 방법을 지정하는 정보를 가져오거나 설정합니다. (Get)
get_InstanceGuid 태스크의 이 인스턴스에 대한 GUID 식별자를 가져옵니다.
get_Interval 작업을 다시 시작할 때마다 걸리는 시간을 가져오거나 설정합니다. (Get)
get_Item 컬렉션에서 지정된 동작을 가져옵니다.
get_Item 컬렉션에서 지정된 등록된 작업을 가져옵니다.
get_Item 컬렉션에서 지정된 작업을 가져옵니다.
get_Item 컬렉션에서 지정된 폴더를 가져옵니다.
get_Item 컬렉션에서 지정된 이름-값 쌍을 가져옵니다.
get_Item 컬렉션에서 지정된 트리거를 가져옵니다.
get_LastRunTime 등록된 작업이 마지막으로 실행된 시간을 가져옵니다.
get_LastTaskResult 등록된 작업이 마지막으로 실행되었을 때 반환된 결과를 가져옵니다.

get_LogonType 보안 주체와 연결된 작업을 실행하는 데 필요한 보안 로그온 메서드를 가져오거나 설정합니다. (Get)
get_MaintenanceSettings 자동 유지 관리 중에 작업 스케줄러가 작업을 수행하는 데 사용하는 IMaintenanceSettingsobject에 대한 포인터를 가져오거나 설정합니다. (Get)
get_MessageBody 메시지 상자의 본문에 표시되는 메시지 텍스트를 가져오거나 설정합니다. (Get)
get_MonthsOfYear 작업이 실행되는 연도의 월을 가져오거나 설정합니다. (IMonthlyDOWTrigger.get_MonthsOfYear)
get_MonthsOfYear 작업이 실행되는 연도의 월을 가져오거나 설정합니다. (IMonthlyTrigger.get_MonthsOfYear)
get_MultipleInstances 작업 스케줄러가 작업의 여러 인스턴스를 처리하는 방법을 정의하는 정책을 가져오거나 설정합니다. (Get)
get_Name 네트워크 프로필의 이름을 가져오거나 설정합니다. (Get)
get_Name 등록된 작업의 이름을 가져옵니다.
get_Name 태스크의 이름을 가져옵니다.
get_Name 작업이 포함된 폴더를 식별하는 데 사용되는 이름을 가져옵니다.
get_Name 이름-값 쌍의 값과 연결된 이름을 가져오거나 설정합니다. (Get)

get_NetworkSettings 네트워크 프로필 식별자와 이름을 포함하는 네트워크 설정 개체를 가져오거나 설정합니다. (Get)
get_NextRunTime 등록된 작업이 다음에 실행되도록 예약된 시간을 가져옵니다.
get_NumberOfMissedRuns 등록된 작업이 예약된 실행을 놓친 횟수를 가져옵니다.
get_Path 실행 파일의 경로를 가져오거나 설정합니다. (Get)
get_Path 등록된 작업이 저장되는 경로를 가져옵니다.
get_Path 태스크가 저장되는 경로를 가져옵니다.
get_Path 폴더가 저장되는 경로를 가져옵니다.
get_Period 정기적인 자동 유지 관리 중에 태스크를 한 번 실행해야 하는 시간을 가져오거나 설정합니다. (Get)
get_Principal 작업에 대한 보안 자격 증명을 제공하는 작업의 보안 주체를 가져오거나 설정합니다. (Get)
get_Priority 작업의 우선 순위 수준을 가져오거나 설정합니다. (Get)
get_ProcessTokenSidType SID(작업 프로세스 보안 식별자) 형식을 가져오거나 설정합니다. (Get)
get_RandomDelay 트리거의 시작 시간에 임의로 추가되는 지연 시간을 가져오거나 설정합니다. (IDailyTrigger.get_RandomDelay)

get_RandomDelay 트리거의 시작 시간에 임의로 추가되는 지연 시간을 가져오거나 설정합니다. (IMonthlyDOWTrigger.get_RandomDelay)
get_RandomDelay 트리거의 시작 시간에 임의로 추가되는 지연 시간을 가져오거나 설정합니다. (IMonthlyTrigger.get_RandomDelay)
get_RandomDelay 트리거의 시작 시간에 임의로 추가되는 지연 시간을 가져오거나 설정합니다. (ITimeTrigger.get_RandomDelay)
get_RandomDelay 트리거의 시작 시간에 임의로 추가되는 지연 시간을 가져오거나 설정합니다. (IWeeklyTrigger.get_RandomDelay)
get_RegistrationInfo 작업에 대한 설명, 태스크 작성자 및 태스크가 등록된 날짜와 같이 작업을 설명하는 데 사용되는 등록 정보를 가져오거나 설정합니다. (Get)
get_Repetition 작업이 실행되는 빈도와 작업이 시작된 후 반복 패턴이 반복되는 기간을 나타내는 값을 가져오거나 설정합니다. (Get)
get_ReplyTo 회신할 전자 메일 주소를 가져오거나 설정합니다. (Get)
get_RequiredPrivilege 인덱스별로 작업의 필요한 권한을 가져옵니다.
get_RequiredPrivilegeCount 필요한 권한 배열의 권한 수를 가져옵니다.
get_RestartCount 작업 스케줄러가 작업을 다시 시작하려고 시도하는 횟수를 가져오거나 설정합니다. (Get)
get_RestartInterval 작업 스케줄러가 작업을 다시 시작할 기간을 지정하는 값을 가져오거나 설정합니다. (Get)

get_RestartOnIdle 컴퓨터가 유휴 상태로 두 번 이상 순환할 때 작업이 다시 시작되는지 여부를 나타내는 부울 값을 가져오거나 설정합니다. (Get)
get_RunLevel 보안 주체와 연결된 작업을 실행하는 데 필요한 권한 수준을 지정하는 데 사용되는 식별자를 가져오거나 설정합니다. (Get)
get_RunOnLastDayOfMonth 작업이 월의 마지막 날에 실행되었음을 나타내는 부울 값을 가져오거나 설정합니다. (Get)
get_RunOnLastWeekOfMonth 작업이 월의 마지막 주에 실행되었음을 나타내는 부울 값을 가져오거나 설정합니다. (Get)
get_RunOnlyIfIdle 컴퓨터가 유휴 상태인 경우에만 작업 스케줄러가 작업을 실행한다는 부울 값을 가져오거나 설정합니다. (Get)
get_RunOnlyIfNetworkAvailable 네트워크를 사용할 수 있는 경우에만 작업 스케줄러가 작업을 실행한다는 부울 값을 가져오거나 설정합니다. (Get)
get_SecurityDescriptor 작업의 보안 설명자를 가져오거나 설정합니다. (Get)
get_Server 전자 메일을 보내는 데 사용하는 SMTP 서버의 이름을 가져오거나 설정합니다. (Get)
get_Settings 작업 스케줄러 서비스에서 작업을 수행하는 방법을 정의하는 설정을 가져오거나 설정합니다. (Get)
get_Source 작업이 시작된 위치를 가져오거나 설정합니다. 예를 들어 작업은 구성 요소, 서비스, 애플리케이션 또는 사용자에서 발생할 수 있습니다. (Get)
get_StartBoundary 트리거가 활성화된 날짜와 시간을 가져오거나 설정합니다. (Get)

get_StartWhenAvailable 작업 스케줄러가 예약된 시간이 지나면 언제든지 작업을 시작할 수 있음을 나타내는 부울 값을 가져오거나 설정합니다. (Get)
get_State 등록된 작업의 작동 상태를 가져옵니다.
get_State 실행 중인 작업의 상태에 대한 식별자를 가져옵니다.
get_StateChange 작업 시작을 트리거하는 터미널 서버 세션 변경의 종류를 가져오거나 설정합니다. (Get)
get_StopAtDurationEnd 반복 패턴 기간이 끝날 때 실행 중인 태스크 인스턴스가 중지되는지 여부를 나타내는 부울 값을 가져오거나 설정합니다. (Get)
get_StopIfGoingOnBatteries 컴퓨터가 배터리로 이동하면 작업이 중지됨을 나타내는 부울 값을 가져오거나 설정합니다. (Get)
get_StopOnIdleEnd 작업이 완료되기 전에 유휴 상태가 종료되면 작업 스케줄러가 작업을 종료할 것임을 나타내는 부울 값을 가져오거나 설정합니다. 유휴 상태는 컴퓨터가 더 이상 유휴 상태가 되지 않을 때 종료됩니다. (Get)
get_Subject 전자 메일 메시지의 제목을 가져오거나 설정합니다. (Get)
get_Subscription 트리거를 발생시키는 이벤트를 식별하는 쿼리 문자열을 가져오거나 설정합니다. (Get)
get_TargetServer 사용자가 연결된 작업 스케줄러 서비스를 실행하는 컴퓨터의 이름을 가져옵니다.
get_Title 메시지 상자의 제목을 가져오거나 설정합니다. (Get)

get_To 전자 메일을 보낼 전자 메일 주소 또는 주소를 가져오거나 설정합니다. (Get)
get_Triggers 작업을 시작하는 데 사용되는 트리거 컬렉션을 가져오거나 설정합니다. (Get)
get_Type 동작의 형식을 가져옵니다.
get_Type 트리거의 형식을 가져옵니다.
get_URI 작업의 URI를 가져오거나 설정합니다. (Get)
get_UserId 사용자의 식별자를 가져오거나 설정합니다. (Get)
get_UserId 보안 주체와 연결된 작업을 실행하는 데 필요한 사용자 식별자를 가져오거나 설정합니다. (Get)
get_UserId 터미널 서버 세션에 대한 사용자를 가져오거나 설정합니다. 이 사용자에 대한 세션 상태 변경이 감지되면 작업이 시작됩니다. (Get)
get_UseUnifiedSchedulingEngine 통합 일정 엔진이 이 작업을 실행하는 데 활용될 것임을 나타내는 부울 값을 가져오거나 설정합니다. (Get)
get_Value 이름-값 쌍의 이름과 연결된 값을 가져오거나 설정합니다. (Get)
get_ValueQueries 명명된 XPath 쿼리의 컬렉션을 가져오거나 설정합니다. 컬렉션의 각 쿼리는 구독 속성에 지정된 구독 쿼리에서 반환된 마지막 일치 이벤트 XML에 적용됩니다. (Get)

get_Version 작업의 버전 번호를 가져오거나 설정합니다. (Get)
get_Volatile Windows가 시작될 때마다 작업이 자동으로 비활성화되는지 여부를 나타내는 부울 값을 가져오거나 설정합니다. (Get)
get_WaitTimeout 작업 스케줄러가 유휴 상태가 발생할 때까지 대기하는 시간을 나타내는 값을 가져오거나 설정합니다. (Get)
get_WakeToRun 작업을 실행할 때 작업 스케줄러가 컴퓨터를 절전 모드에서 해제하고 작업이 완료될 때까지 컴퓨터를 절전 모드에서 해제하도록 나타내는 부울 값을 가져오거나 설정합니다. (Get)
get_WeeksInterval 일정에서 주 사이의 간격을 가져오거나 설정합니다. (Get)
get_WeeksOfMonth 작업이 실행되는 월의 주를 가져오거나 설정합니다. (Get)
get_WorkingDirectory 실행 파일 또는 실행 파일에서 사용되는 파일이 포함된 디렉터리를 가져오거나 설정합니다. (Get)
get_Xml 등록된 작업에 대한 XML 형식 등록 정보를 가져옵니다.
get_XmlText XML 형식의 컬렉션 버전을 가져오거나 설정합니다. (Get)
get_XmlText 작업에 대한 등록 정보의 XML 형식 버전을 가져오거나 설정합니다. (Get)
get_XmlText 작업의 XML 형식 정의를 가져오거나 설정합니다. (Get)

get_XmlText 작업 설정의 XML 형식 정의를 가져오거나 설정합니다. (Get)
GetAccountInformation 작업 항목의 계정 이름을 검색합니다.
GetApplicationName 이 메서드는 태스크가 연결된 애플리케이션의 이름을 검색합니다.
GetComment 작업 항목에 대한 주석을 검색합니다.
GetContext 동일한 작업 인스턴스에 있는 여러 단계와 작업 간에 컨텍스트를 공유하는 데 사용됩니다.
GetCreator 작업 항목 작성자의 이름을 검색합니다.
GetErrorRetryCount 오류가 발생할 때 작업 스케줄러가 작업을 다시 시도하는 횟수를 검색합니다. 이 메서드가 구현되지 않은 경우
GetErrorRetryInterval 오류가 발생할 경우 작업 스케줄러가 작업 항목을 실행하려고 시도하는 시간 간격(분)을 검색합니다. 이 메서드가 구현되지 않은 경우
GetExitCode 마지막 실행에서 작업 항목과 연결된 실행 파일에서 반환된 마지막 종료 코드를 검색합니다. 또한 이 메서드는 작업 항목을 마지막으로 실행하려고 할 때 작업 스케줄러에 반환된 종료 코드를 반환합니다.
GetFlags 모든 유형의 작업 항목의 동작을 수정하는 플래그를 검색합니다.
GetFolder 지정된 위치에 있는 작업이 포함된 폴더를 가져옵니다.

GetFolder 등록된 작업의 폴더를 가져옵니다.
GetFolders 폴더의 모든 하위 폴더를 가져옵니다.
GetIdleWait 작업 항목에 대한 유휴 대기 시간을 검색합니다.
GetInput 작업에 대한 입력 변수를 가져옵니다.
GetInstances 현재 실행 중인 등록된 작업의 모든 인스턴스를 반환합니다.
GetMaxRunTime 이 메서드는 종료하기 전에 작업을 실행할 수 있는 최대 시간(밀리초)을 검색합니다.
GetMostRecentRunTime 작업 항목이 실행되기 시작한 가장 최근 시간을 검색합니다.
GetNextRunTime 작업 항목이 다음에 실행될 때를 검색합니다.
GetPage 이 메서드는 작업 개체와 연결된 하나 이상의 속성 시트 페이지를 검색합니다.
GetParameters 이 메서드는 태스크의 명령줄 매개 변수를 검색합니다.
GetPriority 이 메서드는 작업의 우선 순위를 검색합니다.
GetRunningTasks 실행 중인 작업의 컬렉션을 가져옵니다.

GetRunTimes 지정된 기간 동안 작업 항목 실행 시간을 검색합니다.
GetRunTimes 지정된 시간 동안 등록된 작업이 실행되도록 예약된 시간을 가져옵니다.
GetSecurityDescriptor 등록된 작업의 자격 증명으로 사용되는 보안 설명자를 가져옵니다.
GetSecurityDescriptor 폴더의 보안 설명자를 가져옵니다.
GetStatus 작업 항목의 상태를 검색합니다.
GetTargetComputer GetTargetComputer 메서드는 ITaskScheduler가 현재 대상으로 지정된 컴퓨터의 이름을 반환합니다.
GetTask 폴더의 지정된 위치에 있는 작업을 가져옵니다.
GetTaskFlags 이 메서드는 작업의 동작을 수정하는 플래그를 반환합니다.
GetTasks 폴더의 모든 작업을 가져옵니다.
GetTrigger 작업 트리거를 검색합니다.
GetTrigger GetTrigger 메서드는 현재 작업 트리거를 검색합니다.
GetTriggerCount 현재 작업 항목에 대한 트리거 수를 검색합니다.

GetTriggerString 작업 항목 트리거를 설명하는 문자열을 검색합니다.
GetTriggerString GetTriggerString 메서드는 문자열 형식으로 현재 작업 트리거를 검색합니다. 이 문자열은 작업 스케줄러 사용자 인터페이스에 "매일 오후 2시에 5/11/97부터 시작"이라는 형식으로 표시됩니다.
GetWorkingDirectory 이 메서드는 작업의 작업 디렉터리를 검색합니다.
GetWorkItemData 작업 항목과 연결된 애플리케이션 정의 데이터를 검색합니다.
IsOfType IsOfType 메서드는 개체의 형식을 검사하여 특정 인터페이스를 지원하는지 확인합니다.
NewTask 설정 및 속성으로 채워진 다음 ITaskFolder::RegisterTaskDefinition 메서드를 사용하여 등록할 빈 작업 정의 개체를 반환합니다.
NewWorkItem NewWorkItem 메서드는 작업 항목의 공간을 할당하고 해당 주소를 검색하는 새 작업 항목을 만듭니다.
다음 열거형 시퀀스에서 지정한 다음 개수의 작업을 검색합니다.
일시 중지 COM 처리기를 일시 중지하기 위해 호출됩니다.
put_Actions 태스크에서 수행하는 작업의 컬렉션을 가져오거나 설정합니다. (Put)
put_AllowDemandStart 실행 명령 또는 상황에 맞는 메뉴를 사용하여 작업을 시작할 수 있음을 나타내는 부울 값을 가져오거나 설정합니다. (Put)

put_AllowHardTerminate TerminateProcess를 사용하여 작업 스케줄러 서비스에서 작업을 종료할 수 있음을 나타내는 부울 값을 가져오거나 설정합니다. (Put)
put_Arguments 명령줄 작업과 연결된 인수를 가져오거나 설정합니다. (Put)
put_Attachments 전자 메일 메시지와 함께 전송되는 첨부 파일 배열에 대한 포인터를 가져오거나 설정합니다. (Put)
put_Author 작업의 작성자를 가져오거나 설정합니다. (Put)
put_Bcc 전자 메일 메시지에서 숨은 참조할 전자 메일 주소 또는 주소를 가져오거나 설정합니다. (Put)
put_Body 전자 메일 메시지가 포함된 전자 메일의 본문을 가져오거나 설정합니다. (Put)
put_Cc 전자 메일 메시지에서 참조하려는 전자 메일 주소 또는 주소를 가져오거나 설정합니다. (Put)
put_ClassId 처리기 클래스의 식별자를 가져오거나 설정합니다. (Put)
put_Compatibility 태스크가 호환되는 작업 스케줄러의 버전을 나타내는 정수 값을 가져오거나 설정합니다. (Put)
put_Context 작업에 대한 보안 주체의 식별자를 가져오거나 설정합니다. (Put)
put_Data 처리기와 연결된 추가 데이터를 가져오거나 설정합니다. (Put)
put_Data 작업과 연결된 데이터를 가져오거나 설정합니다. (Put)

put_Date 작업이 등록된 날짜와 시간을 가져오거나 설정합니다. (Put)
put_DaysInterval 일정에서 일 사이의 간격을 가져오거나 설정합니다. (Put)
put_DaysOfMonth 작업이 실행되는 월의 일을 가져오거나 설정합니다. (Put)
put_DaysOfWeek 작업이 실행되는 요일을 가져오거나 설정합니다. (Put)
put_DaysOfWeek 작업이 실행되는 요일을 가져오거나 설정합니다. (Put)
put_Deadline 정기적인 자동 유지 관리 중에 태스크를 완료하지 못한 경우 작업 스케줄러가 긴급 자동 유지 관리 중에 작업을 실행하려고 시도하는 시간을 가져오거나 설정합니다. (Put)
put_Delay 시스템이 부팅된 시간과 작업이 시작된 시점 사이의 시간을 나타내는 값을 가져오거나 설정합니다. (Put)
put_Delay 이벤트가 발생한 시간과 태스크가 시작된 시점 사이의 시간을 나타내는 값을 가져오거나 설정합니다. (Put)
put_Delay 사용자가 로그온한 시간과 작업이 시작된 시점 사이의 시간을 나타내는 값을 가져오거나 설정합니다. (Put)
put_Delay 작업이 등록된 시간과 작업이 시작된 시점 사이의 시간을 가져오거나 설정합니다. (Put)
put_Delay 터미널 서버 세션 상태 변경이 감지된 후 작업을 시작하기 전에 지연이 발생하는 시간을 나타내는 값을 가져오거나 설정합니다. (ISessionStateChangeTrigger.put_Delay)

put_DeleteExpiredTaskAfter 작업 스케줄러가 만료된 후 작업을 삭제하기 전에 대기하는 시간을 가져오거나 설정합니다. (Put)
put_Description 작업에 대한 설명을 가져오거나 설정합니다. (Put)
put_DisallowStartIfOnBatteries 컴퓨터가 배터리에서 실행되는 경우 작업이 시작되지 않음을 나타내는 부울 값을 가져오거나 설정합니다. (Put)
put_DisallowStartOnRemoteAppSession RAIL(원격 애플리케이션 통합 로컬) 세션에서 실행되도록 트리거되는 경우 작업이 시작되지 않도록 지정하는 부울 값을 가져오거나 설정합니다. (Put)
put_DisplayName 보안 주체의 이름을 가져오거나 설정합니다. (Put)
put_Documentation 작업에 대한 추가 설명서를 가져오거나 설정합니다. (Put)
put_Duration 패턴이 반복되는 기간을 가져오거나 설정합니다. (Put)
put_Enabled 등록된 작업이 사용되는지 여부를 나타내는 부울 값을 가져오거나 설정합니다. (Put)
put_Enabled 트리거가 사용되는지 여부를 나타내는 부울 값을 가져오거나 설정합니다. (Put)
put_EndBoundary 트리거가 비활성화된 날짜 및 시간을 가져오거나 설정합니다. (Put)
put_Exclusive 자동 유지 관리 중에 작업 스케줄러가 배타적 모드에서 작업을 시작해야 하는지 여부를 나타냅니다. (Put)

put_ExecutionTimeLimit 작업을 완료할 수 있는 시간을 가져오거나 설정합니다. (Put)
put_ExecutionTimeLimit 이 트리거에서 시작한 작업을 실행할 수 있는 최대 시간을 가져오거나 설정합니다. (Put)
put_From 전자 메일을 보낼 전자 메일 주소를 가져오거나 설정합니다. (Put)
put_GroupId 보안 주체와 연결된 작업을 실행하는 데 필요한 사용자 그룹의 식별자를 가져오거나 설정합니다. (Put)
put_HeaderFields 보낼 전자 메일 메시지의 헤더 정보를 가져오거나 설정합니다. (Put)
put_Hidden 작업이 UI에 표시되지 않음을 나타내는 부울 값을 가져오거나 설정합니다. (Put)
put_Id 작업의 식별자를 가져오거나 설정합니다. (Put)
put_Id 네트워크 프로필을 식별하는 GUID 값을 가져오거나 설정합니다. (Put)
put_Id 보안 주체의 식별자를 가져오거나 설정합니다. (Put)
put_Id 트리거의 식별자를 가져오거나 설정합니다. (Put)
put_IdleDuration 작업이 실행되기 전에 컴퓨터가 유휴 상태여야 하는 시간을 나타내는 값을 가져오거나 설정합니다. (Put)

put_IdleSettings 컴퓨터가 유휴 상태일 때 작업 스케줄러가 작업을 수행하는 방법을 지정하는 정보를 가져오거나 설정합니다. (Put)
put_Interval 태스크를 다시 시작할 때마다 걸리는 시간을 가져오거나 설정합니다. (Put)
put_LogonType 보안 주체와 연결된 작업을 실행하는 데 필요한 보안 로그온 메서드를 가져오거나 설정합니다. (Put)
put_MaintenanceSettings 자동 유지 관리 중에 작업 스케줄러가 작업을 수행하는 데 사용하는 IMaintenanceSettingsobject에 대한 포인터를 가져오거나 설정합니다. (Put)
put_MessageBody 메시지 상자의 본문에 표시되는 메시지 텍스트를 가져오거나 설정합니다. (Put)
put_MonthsOfYear 작업이 실행되는 연도의 월을 가져오거나 설정합니다. (IMonthlyDOWTrigger.put_MonthsOfYear)
put_MonthsOfYear 작업이 실행되는 연도의 월을 가져오거나 설정합니다. (IMonthlyTrigger.put_MonthsOfYear)
put_MultipleInstances 작업 스케줄러가 태스크의 여러 인스턴스를 처리하는 방법을 정의하는 정책을 가져오거나 설정합니다. (Put)
put_Name 네트워크 프로필의 이름을 가져오거나 설정합니다. (Put)
put_Name 이름-값 쌍의 값과 연결된 이름을 가져오거나 설정합니다. (Put)
put_NetworkSettings 네트워크 프로필 식별자와 이름을 포함하는 네트워크 설정 개체를 가져오거나 설정합니다. (Put)

put_Path 실행 파일의 경로를 가져오거나 설정합니다. (Put)
put_Period 정기적인 자동 유지 관리 중에 태스크를 한 번 실행해야 하는 시간을 가져오거나 설정합니다. (Put)
put_Principal 작업에 대한 보안 자격 증명을 제공하는 작업의 보안 주체를 가져오거나 설정합니다. (Put)
put_Priority 작업의 우선 순위 수준을 가져오거나 설정합니다. (Put)
put_ProcessTokenSidType SID(작업 프로세스 보안 식별자) 형식을 가져오거나 설정합니다. (Put)
put_RandomDelay 트리거의 시작 시간에 임의로 추가되는 지연 시간을 가져오거나 설정합니다. (IDailyTrigger.put_RandomDelay)
put_RandomDelay 트리거의 시작 시간에 임의로 추가되는 지연 시간을 가져오거나 설정합니다. (IMonthlyDOWTrigger.put_RandomDelay)
put_RandomDelay 트리거의 시작 시간에 임의로 추가되는 지연 시간을 가져오거나 설정합니다. (IMonthlyTrigger.put_RandomDelay)
put_RandomDelay 트리거의 시작 시간에 임의로 추가되는 지연 시간을 가져오거나 설정합니다. (ITimeTrigger.put_RandomDelay)
put_RandomDelay 트리거의 시작 시간에 임의로 추가되는 지연 시간을 가져오거나 설정합니다. (IWeeklyTrigger.put_RandomDelay)
put_RegistrationInfo 작업에 대한 설명, 작업의 작성자, 작업이 등록된 날짜 등 작업을 설명하는 데 사용되는 등록 정보를 가져오거나 설정합니다. (Put)

put_Repetition 태스크가 실행되는 빈도와 작업이 시작된 후 반복 패턴이 반복되는 기간을 나타내는 값을 가져오거나 설정합니다. (Put)
put_ReplyTo 회신할 전자 메일 주소를 가져오거나 설정합니다. (Put)
put_RestartCount 작업 스케줄러가 작업을 다시 시작하려고 시도하는 횟수를 가져오거나 설정합니다. (Put)
put_RestartInterval 작업 스케줄러가 작업을 다시 시작하는 기간을 지정하는 값을 가져오거나 설정합니다. (Put)
put_RestartOnIdle 컴퓨터가 유휴 상태로 두 번 이상 순환할 때 작업이 다시 시작되는지 여부를 나타내는 부울 값을 가져오거나 설정합니다. (Put)
put_RunLevel 보안 주체와 연결된 작업을 실행하는 데 필요한 권한 수준을 지정하는 데 사용되는 식별자를 가져오거나 설정합니다. (Put)
put_RunOnLastDayOfMonth 작업이 월의 마지막 날에 실행되었음을 나타내는 부울 값을 가져오거나 설정합니다. (Put)
put_RunOnLastWeekOfMonth 작업이 월의 마지막 주에 실행되었음을 나타내는 부울 값을 가져오거나 설정합니다. (Put)
put_RunOnlyIfIdle 컴퓨터가 유휴 상태인 경우에만 작업 스케줄러가 작업을 실행한다는 것을 나타내는 부울 값을 가져오거나 설정합니다. (Put)
put_RunOnlyIfNetworkAvailable 네트워크를 사용할 수 있는 경우에만 작업 스케줄러가 작업을 실행한다는 부울 값을 가져오거나 설정합니다. (Put)
put_SecurityDescriptor 작업의 보안 설명자를 가져오거나 설정합니다. (Put)

put_Server 전자 메일을 보내는 데 사용하는 SMTP 서버의 이름을 가져오거나 설정합니다. (Put)
put_Settings 작업 스케줄러 서비스에서 작업을 수행하는 방법을 정의하는 설정을 가져오거나 설정합니다. (Put)
put_Source 작업이 시작된 위치를 가져오거나 설정합니다. 예를 들어 작업은 구성 요소, 서비스, 애플리케이션 또는 사용자에서 발생할 수 있습니다. (Put)
put_StartBoundary 트리거가 활성화된 날짜와 시간을 가져오거나 설정합니다. (Put)
put_StartWhenAvailable 작업 스케줄러가 예약된 시간이 지나면 언제든지 작업을 시작할 수 있음을 나타내는 부울 값을 가져오거나 설정합니다. (Put)
put_StateChange 작업 시작을 트리거하는 터미널 서버 세션 변경의 종류를 가져오거나 설정합니다. (Put)
put_StopAtDurationEnd 반복 패턴 기간이 끝날 때 실행 중인 태스크 인스턴스가 중지되는지 여부를 나타내는 부울 값을 가져오거나 설정합니다. (Put)
put_StopIfGoingOnBatteries 컴퓨터가 배터리로 이동하면 작업이 중지됨을 나타내는 부울 값을 가져오거나 설정합니다. (Put)
put_StopOnIdleEnd 작업이 완료되기 전에 유휴 상태가 종료되면 작업 스케줄러가 작업을 종료할 것임을 나타내는 부울 값을 가져오거나 설정합니다. 유휴 상태는 컴퓨터가 더 이상 유휴 상태가 되지 않을 때 종료됩니다. (Put)
put_Subject 전자 메일 메시지의 제목을 가져오거나 설정합니다. (Put)
put_Subscription 트리거를 발생시키는 이벤트를 식별하는 쿼리 문자열을 가져오거나 설정합니다. (Put)

put_Title 메시지 상자의 제목을 가져오거나 설정합니다. (Put)
put_To 전자 메일을 보낼 전자 메일 주소 또는 주소를 가져오거나 설정합니다. (Put)
put_Triggers 작업을 시작하는 데 사용되는 트리거 컬렉션을 가져오거나 설정합니다. (Put)
put_URI 작업의 URI를 가져오거나 설정합니다. (Put)
put_UserId 사용자의 식별자를 가져오거나 설정합니다. (Put)
put_UserId 보안 주체와 연결된 작업을 실행하는 데 필요한 사용자 식별자를 가져오거나 설정합니다. (Put)
put_UserId 터미널 서버 세션에 대한 사용자를 가져오거나 설정합니다. 이 사용자에 대한 세션 상태 변경이 감지되면 작업이 시작됩니다. (Put)
put_UseUnifiedSchedulingEngine 통합 일정 엔진이 이 작업을 실행하는 데 활용될 것임을 나타내는 부울 값을 가져오거나 설정합니다. (Put)
put_Value 이름-값 쌍의 이름과 연결된 값을 가져오거나 설정합니다. (Put)
put_ValueQueries 명명된 XPath 쿼리의 컬렉션을 가져오거나 설정합니다. 컬렉션의 각 쿼리는 구독 속성에 지정된 구독 쿼리에서 반환된 마지막 일치 이벤트 XML에 적용됩니다. (Put)
put_Version 작업의 버전 번호를 가져오거나 설정합니다. (Put)

put_Volatile Windows가 시작될 때마다 작업이 자동으로 비활성화되는지 여부를 나타내는 부울 값을 가져오거나 설정합니다. (Put)
put_WaitTimeout 작업 스케줄러가 유휴 상태가 발생할 때까지 대기하는 시간을 나타내는 값을 가져오거나 설정합니다. (Put)
put_WakeToRun 작업을 실행할 때 작업 스케줄러가 컴퓨터를 절전 모드에서 해제하고 작업이 완료될 때까지 컴퓨터를 절전 모드에서 해제하도록 나타내는 부울 값을 가져오거나 설정합니다. (Put)
put_WeeksInterval 일정에서 주 사이의 간격을 가져오거나 설정합니다. (Put)
put_WeeksOfMonth 작업이 실행되는 월의 주를 가져오거나 설정합니다. (Put)
put_WorkingDirectory 실행 파일 또는 실행 파일에서 사용되는 파일이 포함된 디렉터리를 가져오거나 설정합니다. (Put)
put_XmlText XML 형식의 컬렉션 버전을 가져오거나 설정합니다. (Put)
put_XmlText 작업에 대한 등록 정보의 XML 형식 버전을 가져오거나 설정합니다. (Put)
put_XmlText 작업의 XML 형식 정의를 가져오거나 설정합니다. (Put)
put_XmlText 작업 설정의 XML 형식 정의를 가져오거나 설정합니다. (Put)
새로 고침 작업의 모든 로컬 인스턴스 변수를 새로 고칩니다.

RegisterTask XML을 사용하여 폴더에 새 작업을 등록(생성)하여 작업을 정의합니다.
RegisterTaskDefinition ITaskDefinition 인터페이스를 사용하여 작업을 정의하여 지정된 위치에 작업을 등록(만듭니다).
제거 컬렉션에서 지정된 작업을 제거합니다.
제거 컬렉션에서 선택한 이름-값 쌍을 제거합니다.
제거 태스크에서 사용하는 트리거 컬렉션에서 지정된 트리거를 제거합니다.
재설정 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumWorkItems.Reset)
재개 COM 처리기를 다시 시작하기 위해 호출됩니다.
실행 작업 스케줄러 서비스에 작업 항목을 실행하라는 요청을 보냅니다.
실행 등록된 작업을 즉시 실행합니다.
RunEx 지정된 플래그 및 세션 식별자를 사용하여 등록된 작업을 즉시 실행합니다.
SetAccountInformation 작업 항목을 실행하는 데 사용되는 계정 이름 및 암호를 설정합니다.
SetApplicationName 이 메서드는 현재 작업에 특정 애플리케이션을 할당합니다.

SetComment 작업 항목에 대한 주석을 설정합니다.
SetCreator 작업 항목 작성자의 이름을 설정합니다.
SetErrorRetryCount 오류가 발생할 경우 작업 스케줄러가 작업 항목을 다시 실행하려고 시도하는 횟수를 설정합니다. 이 메서드가 구현되지 않은 경우
SetErrorRetryInterval 오류가 발생한 후 작업 스케줄러가 작업 항목을 실행하려는 시도 사이의 시간 간격(분)을 설정합니다. 이 메서드가 구현되지 않은 경우
SetFlags 모든 유형의 작업 항목의 동작을 수정하는 플래그를 설정합니다.
SetIdleWait 작업 항목을 실행하기 전에 시스템이 유휴 상태여야 하는 시간(분)을 설정합니다.
SetMaxRunTime 이 메서드는 종료하기 전에 작업을 실행할 수 있는 최대 시간(밀리초)을 설정합니다.
SetOutput 작업에 대한 출력 변수를 설정합니다.
SetParameters 이 메서드는 작업에 대한 명령줄 매개 변수를 설정합니다.
SetPriority 이 메서드는 작업의 우선 순위를 설정합니다.
SetSecurityDescriptor 등록된 작업의 자격 증명으로 사용되는 보안 설명자를 설정합니다.
SetSecurityDescriptor 폴더의 보안 설명자를 설정합니다.

SetTargetComputer SetTargetComputer 메서드는 ITaskScheduler 인터페이스가 작동하는 컴퓨터를 선택하여 원격 작업 관리 및 열거를 허용합니다.
SetTaskFlags 이 메서드는 예약된 작업의 동작을 수정하는 플래그를 설정합니다.
SetTrigger SetTrigger 메서드는 작업 트리거에 대한 트리거 조건을 설정합니다.
SetWorkingDirectory 이 메서드는 작업에 대한 작업 디렉터리를 설정합니다.
SetWorkItemData 이 메서드는 작업 항목과 연결된 애플리케이션 정의 데이터를 저장합니다.
skip 열거형 시퀀스에서 지정한 다음 작업 수를 건너뜁니다.
시작 COM 처리기를 시작하기 위해 호출되었습니다.
중지 등록된 작업을 즉시 중지합니다.
중지 작업의 이 인스턴스를 중지합니다.
중지 COM 처리기를 중지하기 위해 호출되었습니다.
TaskCompleted COM 처리기가 완료되었다는 것을 작업 스케줄러에 알릴 수 있습니다.
Terminate 이 메서드는 작업 항목의 실행을 종료합니다.

UpdateStatus COM 처리기의 완료율에 대해 작업 스케줄러에 알릴 수 있습니다.

인터페이스

IAction 모든 작업 개체에서 상속되는 공통 속성을 제공합니다.
IActionCollection 태스크에서 수행하는 작업을 포함합니다.
IBootTrigger 시스템이 시작될 때 작업을 시작하는 트리거를 나타냅니다.
IComHandlerAction 처리기를 발생시키는 동작을 나타냅니다.
IDailyTrigger 일별 일정에 따라 작업을 시작하는 트리거를 나타냅니다.
IEmailAction 전자 메일 메시지를 보내는 작업을 나타냅니다.
IEnumWorkItems 예약된 작업 폴더에서 작업을 열거하는 메서드를 제공합니다.
IEventTrigger 시스템 이벤트가 발생할 때 작업을 시작하는 트리거를 나타냅니다.
IExecAction 명령줄 작업을 실행하는 작업을 나타냅니다.
IIdleSettings 컴퓨터가 유휴 상태일 때 작업 스케줄러가 작업을 수행하는 방법을 지정합니다.

IIdleTrigger 컴퓨터가 유휴 상태로 전환할 때 작업을 시작하는 트리거를 나타냅니다.
ILogonTrigger 사용자가 로그온할 때 작업을 시작하는 트리거를 나타냅니다.
IMaintenanceSettings 자동 유지 관리 중에 작업 스케줄러가 작업을 수행하는 데 사용하는 설정을 제공합니다.
IMonthlyDOWTrigger 매월 요일 일정에 따라 작업을 시작하는 트리거를 나타냅니다.
IMonthlyTrigger 월별 일정에 따라 작업을 시작하는 트리거를 나타냅니다.
INetworkSettings 작업 스케줄러 서비스에서 네트워크 프로필을 가져오는 데 사용하는 설정을 제공합니다.
IPrincipal 보안 주체에 대한 보안 자격 증명을 제공합니다.
IPrincipal2 보안 주체의 보안 자격 증명에 적용된 확장 설정을 제공합니다.
IProvideTaskPage 작업의 속성 시트 설정에 액세스하는 메서드를 제공합니다.
IRegisteredTask 작업을 즉시 실행하고, 태스크의 실행 중인 인스턴스를 가져오는 데 사용되는 메서드, 작업을 등록하는 데 사용되는 자격 증명을 가져오기 또는 설정하고, 작업을 설명하는 속성을 제공합니다.
IRegisteredTaskCollection 등록된 모든 작업을 포함합니다.
IRegistrationInfo 작업을 설명하는 데 사용할 수 있는 관리 정보를 제공합니다.

IRegistrationTrigger 작업이 등록되거나 업데이트될 때 작업을 시작하는 트리거를 나타냅니다.
IRepetitionPattern 태스크가 실행되는 빈도와 작업이 시작된 후 반복 패턴이 반복되는 기간을 정의합니다.
IRunningTask 실행 중인 작업에서 정보를 얻고 제어하는 메서드를 제공합니다.
IRunningTaskCollection 실행 중인 작업을 제어하는 데 사용되는 컬렉션을 제공합니다.
IScheduledWorkItem 특정 작업 항목을 관리하는 방법을 제공합니다.
ISessionStateChangeTrigger 콘솔 연결 또는 연결 끊기, 원격 연결 또는 연결 끊기, 워크스테이션 잠금 또는 잠금 해제 알림에 대한 작업을 트리거합니다.
IShowMessageAction 작업이 활성화될 때 메시지 상자를 표시하는 작업을 나타냅니다.
Itask 작업을 실행하고, 작업 정보를 가져오거나 설정하고, 작업을 종료하는 방법을 제공합니다. IScheduledWorkItem 인터페이스에서 파생되며 해당 인터페이스의 모든 메서드를 상속합니다.
ITaskDefinition 작업 설정, 트리거, 작업 및 등록 정보와 같은 태스크의 모든 구성 요소를 정의합니다.
ITaskFolder 폴더에 작업을 등록(만들기)하고, 폴더에서 작업을 제거하고, 폴더에서 하위 폴더를 만들거나 제거하는 데 사용되는 메서드를 제공합니다.
ITaskFolderCollection 작업을 포함하는 폴더 컬렉션에 대한 정보 및 제어를 제공합니다.

ITaskHandler COM 처리기를 관리하기 위해 작업 스케줄러 서비스에서 호출하는 메서드를 정의합니다.
ITaskHandlerStatus COM 처리기에서 작업 스케줄러에 처리기의 상태를 알리는 데 사용되는 메서드를 제공합니다.
ITaskNamedValueCollection ITaskNamedValuePair 인터페이스 이름-값 쌍의 컬렉션을 포함합니다.
ITaskNamedValuePair 이름이 값과 연결된 이름-값 쌍을 만듭니다.
ITaskScheduler 작업을 예약하는 방법을 제공합니다.
ITaskService 등록된 작업을 관리하기 위한 작업 스케줄러 서비스에 대한 액세스를 제공합니다.
ITaskSettings 작업 스케줄러 서비스에서 작업을 수행하는 데 사용하는 설정을 제공합니다.
ITaskSettings2 작업 스케줄러가 작업을 실행하는 데 사용하는 확장 설정을 제공합니다. (ITaskSettings2)
ITaskSettings3 작업 스케줄러가 작업을 실행하는 데 사용하는 확장 설정을 제공합니다. (ITaskSettings3)
ITaskTrigger 작업에 대한 트리거에 액세스하고 설정하는 방법을 제공합니다. 트리거는 작업 시작 시간, 반복 조건 및 태스크 실행 시기를 제어하는 기타 매개 변수를 지정합니다.
ITaskVariables 태스크에서 시작되는 작업 처리기 및 외부 실행 파일에 매개 변수로 전달할 수 있는 작업 변수를 정의합니다.
ITimeTrigger 특정 날짜 및 시간에 작업을 시작하는 트리거를 나타냅니다.

ITrigger 모든 트리거 개체에서 상속되는 공통 속성을 제공합니다.
ITriggerCollection 작업의 트리거를 추가, 제거 및 가져오는 데 사용되는 메서드를 제공합니다.
IWeeklyTrigger 주별 일정에 따라 작업을 시작하는 트리거를 나타냅니다.

구조체

매일 작업이 실행되는 간격(일)을 정의합니다.
MONTHLYDATE 태스크가 실행되는 월의 일을 정의합니다.
MONTHLYDOW 작업이 월, 주 및 요일별로 실행되는 날짜를 정의합니다.
TASK_TRIGGER 예약된 작업 항목을 실행하는 시간을 정의합니다.
TRIGGER_TYPE_UNION TASK_TRIGGER 구조체의 Type 멤버 내에서 트리거의 호출 일정을 정의합니다.
주간 작업 호출 사이의 간격(주)을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

전화 통신 애플리케이션 프로그래밍 인터페이스

아티클2023. 08. 22.

전화 통신 애플리케이션 프로그래밍 인터페이스 기술 개요입니다.

전화 통신 애플리케이션 프로그래밍 인터페이스를 개발하려면 다음 헤더가 필요합니다.

- mdhcp.h
- msp.h
- mspaddr.h
- mspcall.h
- mspstrm.h
- rend.h
- tapi3.h
- tapi3cc.h
- tapi3ds.h
- tapi3if.h
- termmgr.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 전화 통신 애플리케이션 프로그래밍 인터페이스

클래스

CMSPAddress CMSPAddress 클래스는 MSP 주소 개체를 구현하고 ITMSPAddress 인터페이스를 지원합니다. 이 클래스는 MSPaddr.h에 정의되어 있습니다. 다음은 클래스를 문서화합니다.
CMSPCallBase CMSPCallBase 클래스는 호출 개체의 제네릭 구현을 제공합니다. ITStreamControl 인터페이스를 지원합니다. 스트림 목록을 유지하는 간단한 기본 클래스로 사용됩니다. 이 클래스는 MSPcall.h에 정의되어 있습니다.
CMSPCallMultiGraph CMSPCallBase에서 파생된 이 클래스는 각 스트림에 필터 그래프를 사용하는 호출을 정의합니다.

이러한 추가적인 특이성을 통해 이 클래스는 CMSPCallBase보다 더 많은 호출 기능을 구현할 수 있습니다. MSPcall.h에 정의됩니다.
CMSPStream CMSPStream 클래스는 애플리케이션이 하위 스트림을 시작, 일시 중지 또는 중지하고 터미널을 선택하거나 선택 취소할 수 있도록 하는 메서드를 노출합니다. MSPstrm.h에 정의됩니다.

열거형

ACDGROUP_EVENT ACDGROUP_EVENT 열거형(tapi3.h)은 ACD 그룹 이벤트를 설명합니다.
ACDGROUP_EVENT ACDGROUP_EVENT 열거형(tapi3cc.h)은 ACD 그룹 이벤트를 설명합니다.
ACDQUEUE_EVENT ACDQUEUE_EVENT 열거형(tapi3.h)은 ACD 큐 이벤트를 설명합니다.
ACDQUEUE_EVENT ACDQUEUE_EVENT 열거형(tapi3cc.h)은 ACD 큐 이벤트를 설명합니다.
ADDRESS_CAPABILITY ADDRESS_CAPABILITY 열거형의 멤버는 ITAddressCapabilities::get_AddressCapability 메서드에서 필요한 주소 기능을 나타내는 데 사용됩니다.
ADDRESS_CAPABILITY_STRING ADDRESS_CAPABILITY_STRING 열거형은 문자열에서 설명하는 주소 기능을 검사 데 사용됩니다.
ADDRESS_EVENT ADDRESS_EVENT 열거형은 주소 이벤트를 설명합니다. ITAddressEvent::get_Event 메서드는 발생한 주소 이벤트의 유형을 나타내기 위해 이 열거형의 멤버를 반환합니다.
ADDRESS_STATE ADDRESS_STATE 열거형은 ITAddress::get_State 메서드에서 주소 상태를 검사 데 사용됩니다.
AGENT_EVENT

AGENT_EVENT 열거형(tapi3.h)은 에이전트 이벤트를 설명합니다.
AGENT_EVENT AGENT_EVENT 열거형(tapi3cc.h)은 에이전트 이벤트를 설명합니다.
AGENT_SESSION_EVENT AGENT_SESSION_EVENT 열거형(tapi3.h)은 에이전트 세션 이벤트를 설명합니다.
AGENT_SESSION_EVENT AGENT_SESSION_EVENT 열거형(tapi3cc.h)은 에이전트 세션 이벤트를 설명합니다.
AGENT_SESSION_STATE AGENT_SESSION_STATE 열거형(tapi3.h)은 ITAgentSession::get_State 및 ITAgentSession::p ut_State 메서드에서 사용하는 에이전트 세션 표시기를 정의합니다.
AGENT_SESSION_STATE 이 AGENT_SESSION_STATE 열거형(tapi3cc.h)은 ITAgentSession::get_State 및 ITAgentSession::p ut_State 메서드에서 사용하는 에이전트 세션 표시기를 정의합니다.
AGENT_STATE AGENT_STATE 열거형(tapi3.h)은 ITAgent::p ut_State 및 ITAgent::get_State 메서드에서 에이전트 상태를 설명하는 데 사용됩니다.
AGENT_STATE AGENT_STATE 열거형(tapi3cc.h)은 ITAgent::p ut_State 및 ITAgent::get_State 메서드에서 에이전트 상태를 설명하는 데 사용됩니다.
AGENTHANDLER_EVENT AGENTHANDLER_EVENT 열거형(tapi3.h)은 에이전트 처리기 이벤트를 설명합니다.
AGENTHANDLER_EVENT AGENTHANDLER_EVENT 열거형(tapi3cc.h)은 에이전트 처리기 이벤트를 설명합니다.
CALL_MEDIA_EVENT CALL_MEDIA_EVENT 열거형은 호출 미디어 이벤트를 설명합니다. ITCallMediaEvent::get_Event 메서드는 발생한 호출 미디어 이벤트의 유형을 나타내기 위해 이 열거형의 멤버를 반환합니다.
CALL_MEDIA_EVENT_CAUSE

CALL_MEDIA_EVENT_CAUSE 열거형은 ITCallMediaEvent::get_Cause 메서드에서 디바이스 시간 제한과 같은 미디어 이벤트의 원인에 대한 설명을 반환하는 데 사용됩니다.
CALL_NOTIFICATION_EVENT CALL_NOTIFICATION_EVENT 열거형은 호출 알림 이벤트를 설명합니다. ITCallNotificationEvent::get_Event 메서드는 발생한 호출 알림 이벤트의 유형을 나타내기 위해 이 열거형의 멤버를 반환합니다.
CALL_PRIVILEGE CALL_PRIVILEGE 멤버는 ITCallInfo::get_Privilege 메서드에서 반환되며 현재 애플리케이션이 현재 호출을 소유하거나 모니터링하는 시기를 나타냅니다.
CALL_STATE CALL_STATE 열거형은 ITCallInfo::get_CallState 및 ITCallStateEvent::get_State 메서드에서 사용됩니다.
CALL_STATE_EVENT_CAUSE CALL_STATE_EVENT_CAUSE 열거형은 ITCallStateEvent::get_Cause 메서드에서 반환됩니다.
CALLHUB_EVENT CALLHUB_EVENT 열거형은 CallHub 이벤트를 설명합니다. ITCallHubEvent::get_Event 메서드는 이 열거형의 멤버를 반환하여 발생한 CallHub 이벤트의 유형을 나타냅니다.
CALLHUB_STATE CALLHUB_STATE 열거형은 ITCallHub::get_State 메서드에서 반환하는 상태 표시기입니다.
CALLINFO_BUFFER CALLINFO_BUFFER 열거형은 ITCallInfo::GetCallInfoBuffer 메서드 또는 ITCallInfo::SetCallInfoBuffer 메서드에서 액세스하는 버퍼 유형을 나타냅니다.
CALLINFO_LONG CALLINFO_LONG 열거형은 LONG 형식의 호출 정보를 설정하고 가져오는 ITCallInfo 메서드에서 사용됩니다.
CALLINFO_STRING CALLINFO_STRING 열거형은 문자열 사용과 관련된 호출 정보를 설정하고 가져오는 ITCallInfo 메서드에서 사용됩니다.
CALLINFOCHANGE_CAUSE

CALLINFOCHANGE_CAUSE 열거형은 ITCallInfoChangeEvent::get_Cause 메서드에서 변경된 호출 정보 형식에 대한 설명을 반환하는 데 사용됩니다.

DIRECTORY_OBJECT_TYPE

DIRECTORY_OBJECT_TYPE 열거형은 디렉터리 개체가 회의인지 사용자인지를 설명하는 설명자입니다.

DIRECTORY_TYPE

DIRECTORY_TYPE 열거형은 디렉터리 서버의 유형을 나타내는 데 사용됩니다.

DISCONNECT_CODE

DISCONNECT_CODE 열거형은 ITBasicCallControl::D isconnect 메서드에서 사용됩니다.

FINISH_MODE

FINISH_MODE 열거형은 애플리케이션에서 필요한 호출 완료 유형을 나타내는 데 사용됩니다. TAPI DLL이 수행하는 작업은 통화 전송이 완료되었는지 또는 통화가 회의에 추가되는지에 따라 달라집니다.

FT_STATE_EVENT_CAUSE

FT_STATE_EVENT_CAUSE 열거형은 파일 터미널 이벤트의 형식을 나타냅니다.

FULLDUPLEX_SUPPORT

FULLDUPLEX_SUPPORT 열거형은 지정된 터미널이 전체 이중 작업을 지원하는지 여부를 나타내기 위해 레거시 TSP와 상호 작용하는 애플리케이션에서 사용됩니다. 이 열거형은 ITLegacyWaveSupport::IsFullDuplex 메서드에 의해 반환됩니다.

MSP_ADDRESS_EVENT

MSP_ADDRESS_EVENT 열거형(tapi3.h) 상수는 MSP_EVENT ME_ADDRESS_EVENT 때 GetEvent 메서드에 의해 MSP_EVENT_INFO 구조체 내에서 반환됩니다.

MSP_ADDRESS_EVENT

MSP_EVENT ME_ADDRESS_EVENT 경우 MSP_ADDRESS_EVENT(msp.h) 열거형 상수는 GetEvent 메서드에 의해 MSP_EVENT_INFO 구조체 내에서 반환됩니다.

MSP_CALL_EVENT

MSP_CALL_EVENT 열거형(tapi3.h) 상수는 MSP_EVENT ME_CALL_EVENT 때 GetEvent 메서드에 의해 MSP_EVENT_INFO 구조체 내에서 반환됩니다.

MSP_CALL_EVENT

MSP_EVENT ME_CALL_EVENT 때 MSP_CALL_EVENT(msp.h) 열거형 상수는 GetEvent 메서드에 의해 MSP_EVENT_INFO 구조체 내에서 반환됩니다.
MSP_CALL_EVENT_CAUSE MSP_EVENT ME_CALL_EVENT 경우 MSP_CALL_EVENT_CAUSE 열거형(tapi3.h) 상수는 GetEvent 메서드에 의해 MSP_EVENT_INFO 구조체 내에서 반환됩니다.
MSP_CALL_EVENT_CAUSE MSP_CALL_EVENT_CAUSE(msp.h) 열거형 상수는 MSP_EVENT ME_CALL_EVENT 때 GetEvent 메서드에 의해 MSP_EVENT_INFO 구조체 내에서 반환됩니다.
MSP_EVENT MSP_EVENT 열거자(tapi3.h)는 MSP_EVENT_INFO 포함된 이벤트 정보의 유형을 정의합니다.
MSP_EVENT MSP_EVENT(msp.h) 열거자는 MSP_EVENT_INFO 포함된 이벤트 정보의 유형을 정의합니다.
PHONE_BUTTON_FUNCTION PHONE_BUTTON_FUNCTION 열거형은 단추의 함수에 대한 자세한 정보를 제공합니다.
PHONE_BUTTON_MODE PHONE_BUTTON_MODE 열거형은 휴대폰 단추의 모드를 설명합니다.
PHONE_BUTTON_STATE PHONE_BUTTON_STATE 열거형은 휴대폰 단추의 상태를 설명합니다.
PHONE_EVENT PHONE_EVENT 열거형은 전화 이벤트의 유형을 나타냅니다.
PHONE_HOOK_SWITCH_DEVICE PHONE_HOOK_SWITCH_DEVICE 열거형은 휴대폰 디바이스의 스위치 후크 유형을 나타내는 데 사용됩니다.
PHONE_HOOK_SWITCH_STATE PHONE_HOOK_SWITCH_STATE 열거형은 휴대폰 후크위치 상태 표시기를 제공합니다.
PHONE_LAMP_MODE PHONE_LAMP_MODE 열거형은 전화 램프의 상태 표시기를 제공합니다.

PHONE_PRIVILEGE PHONE_PRIVILEGE 열거형은 현재 휴대폰 디바이스와 관련하여 애플리케이션의 권한 상태 나타냅니다.
PHONE_TONE PHONE_TONE 열거형은 전화 톤을 식별합니다.
PHONECAPS_BUFFER PHONECAPS_BUFFER 열거형은 버퍼에서 설명하는 전화 기능을 설정하거나 가져오는 메서드에서 사용됩니다.
PHONECAPS_LONG PHONECAPS_LONG 열거형은 long 값으로 설명된 전화 기능을 설정하거나 가져오는 메서드에서 사용됩니다.
PHONECAPS_STRING PHONECAPS_STRING 열거형은 문자열에서 설명하는 전화 기능을 설정하거나 가져오는 메서드에서 사용됩니다.
QOS_EVENT QOS_EVENT 열거형은 QOS(서비스 품질) 이벤트를 설명합니다. ITQOSEvent::get_Event 메서드는 발생한 QOS 이벤트의 유형을 나타내기 위해 이 열거형의 멤버를 반환합니다.
QOS_SERVICE_LEVEL QOS_SERVICE_LEVEL 열거형은 ITBasicCallControl::SetQOS 메서드에서 호출에 대한 서비스 품질 요구 사항을 나타내는 데 사용됩니다.
RND_ADVERTISING_SCOPE RND_ADVERTISING_SCOPE 열거형의 멤버는 회의 공지 사항의 배포 범위를 지정합니다. 값은 ITDirectoryObjectConference 인터페이스의 advertising scope 속성에 해당합니다.
TAPI_EVENT TAPI 개체에서 변경이 발생했음을 애플리케이션에 알리는 데 사용됩니다.
TAPI_GATHERTERM TAPI_GATHERTERM 열거형은 TAPI 서버가 호출에서 숫자 수집을 종료한 이유를 설명하는 데 사용됩니다.

TAPI_OBJECT_TYPE TAPI_OBJECT_TYPE 열거형은 이벤트에 관련된 TAPI 개체의 형식을 나타내는 데 사용됩니다.
TAPI_TONEMODE TAPI_TONEMODE 열거형은 선 톤을 생성할 때 사용되는 다양한 선택 항목을 설명하는 데 사용됩니다.
TAPIOBJECT_EVENT TAPIOBJECT_EVENT 열거형은 TAPI 개체 이벤트를 설명합니다. ITTAPIObjectEvent::get_Event 메서드는 이 열거형의 멤버를 반환하여 발생한 TAPI 개체 이벤트의 형식을 나타냅니다.
TERMINAL_DIRECTION TERMINAL_DIRECTION 열거형은 로컬 컴퓨터 또는 터미널의 방향 기능과 관련하여 미디어 스트림의 방향을 설명하는 데 사용됩니다.
TERMINAL_MEDIA_STATE TERMINAL_MEDIA_STATE 열거형은 파일 터미널의 상태를 나타냅니다.
TERMINAL_STATE TERMINAL_STATE 열거형은 터미널 디바이스의 현재 상태를 설명합니다. 이 열거형은 ITTerminal::get_State 메서드에서 반환됩니다.
TERMINAL_TYPE TERMINAL_TYPE 열거형은 터미널의 형식을 설명합니다. 이 열거형은 ITTerminal::get_TerminalType 메서드에서 반환됩니다.
TMGR_DIRECTION TMGR_DIRECTION 열거형은 플러그형 터미널 메서드에서 터미널 방향을 나타내는 데 사용됩니다.

Functions

추가 Add 메서드는 지정된 인덱스의 컬렉션에 새 항목을 삽입합니다.
추가

Add 메서드는 레지스트리에 터미널 정보를 추가합니다. 터미널에 대한 항목이 이미 있는 경우 메서드는 항목을 편집합니다.
추가 Add 메서드는 레지스트리에 플러그형 터미널 슈퍼클래스를 추가합니다. superclass가 이미 있는 경우 메서드는 superclass에 대한 정보를 수정합니다.
AddDirectoryObject AddDirectoryObject 메서드는 ITDirectoryObject 개체를 서버에 추가합니다. 디렉터리 또는 사용자 컴퓨터 매핑일 수 있습니다.
답변 Answer 메서드는 들어오는 호출에 응답합니다. 이 메서드는 호출 상태가 CS_OFFERING 경우에만 성공할 수 있습니다.
바인딩 Bind 메서드는 서버에 바인딩됩니다.
BlindTransfer BlindTransfer 메서드는 지정된 대상 주소에 대한 지정된 호출의 블라인드 또는 단일 단계 전송을 수행합니다. 애플리케이션은 호출의 소유자여야 합니다. 성공적으로 전송되면 호출 상태가 CS_DISCONNECTED 전환됩니다.
지우기 Clear 메서드는 CallHub에서 모든 호출 및 참가자를 제거하려고 시도합니다. 애플리케이션에 모든 호출의 연결을 끊는 데 필요한 권한이 없을 수 있습니다.
지우기 Clear 메서드는 이 개체의 모든 전달 정보를 지웁니다.
복제 IEnumACDGroup::Clone 메서드(tapi3.h)는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다.
복제 IEnumAgent::Clone 메서드(tapi3.h)는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다.
복제

IEnumAgentHandler::Clone 메서드(tapi3.h)는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다.

복제

IEnumAgentSession::Clone 메서드(tapi3.h)는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다.

복제

IEnumQueue::Clone 메서드(tapi3.h)는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다.

복제

Clone 메서드는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumAddress.Clone)

복제

Clone 메서드는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumBstr.Clone)

복제

Clone 메서드는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumCall.Clone)

복제

Clone 메서드는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumCallHub.Clone)

복제

Clone 메서드는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumCallingCard.Clone)

복제

Clone 메서드는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumLocation.Clone)

복제

Clone 메서드는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumPhone.Clone)

복제

Clone 메서드는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumPluggableSuperclassInfo.Clone)
복제 Clone 메서드는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumPluggableTerminalClassInfo.Clone)
복제 Clone 메서드는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다. (IEnumStream.Clone)
복제 Clone 메서드는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다. (IEnumSubStream.Clone)
복제 Clone 메서드는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumTerminal.Clone)
복제 Clone 메서드는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumTerminalClass.Clone)
복제 IEnumACDGroup::Clone 메서드(tapi3cc.h)는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다.
복제 IEnumAgent::Clone 메서드(tapi3cc.h)는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다.
복제 IEnumAgentHandler::Clone 메서드(tapi3cc.h)는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다.
복제 IEnumAgentSession::Clone 메서드(tapi3cc.h)는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다.
복제

IEnumQueue::Clone 메서드(tapi3cc.h)는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다.
복제 Clone 메서드는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다. (IEnumDialableAddrs.Clone)
복제 Clone 메서드는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다. (IEnumDirectory.Clone)
복제 Clone 메서드는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다. (IEnumDirectoryObject.Clone)
복제 Clone 메서드는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다. (IEnumMcastScope.Clone)
닫기 Close 메서드는 이 휴대폰 디바이스를 닫습니다. 애플리케이션이 ITPhone::Open 메서드를 호출할 때까지 휴대폰 디바이스는 닫힌 상태로 유지됩니다. 자세한 내용은 아래 설명 부분을 참조하십시오.
CompleteConnectTerminal CompleteConnectTerminal 메서드는 터미널 연결을 완료합니다.
회의 회의 방법은 현재 통화가 참가자인 회의에 상담 전화를 추가합니다. 연결된 ITCallHub 개체가 없으면 만들어집니다.
ConfigDialog ConfigDialog 메서드를 사용하면 지정된 줄 디바이스의 공급자가 사용자가 줄 디바이스와 관련된 매개 변수를 구성할 수 있도록 대화 상자를 표시합니다. 편집할 수 있는 매개 변수는 현재 디바이스에서 사용 중인 매개 변수입니다.
ConfigDialogEdit ConfigDialogEdit 메서드를 사용하면 지정된 줄 디바이스의 공급자가 사용자가 줄 디바이스와 관련된 매개 변수를 구성할 수 있도록 대화 상자를 표시합니다.

연결 Connect 메서드는 나가는 호출의 연결을 완료하려고 시도합니다.
연결 Connect 메서드는 디렉터리 서버에 대한 연결을 설정합니다.
ConnectTerminal ConnectTerminal 메서드는 필터를 연결하고 연결에 대한 핀 집합을 반환합니다. 필터 그래프에 각 내부 필터를 입력하고 내부 필터를 함께 연결하고(해당하는 경우) 연결에 대한 핀 집합을 반환합니다.
CreateAgent CreateAgent 메서드(tapi3.h)는 에이전트 개체를 만듭니다.
CreateAgent ITAgentHandler::CreateAgent 메서드(tapi3cc.h)는 에이전트 개체를 만듭니다.
CreateAgentWithID CreateAgentWithID 메서드(tapi3.h)는 에이전트 식별자를 기반으로 에이전트 개체를 만듭니다.
CreateAgentWithID ITAgentHandler::CreateAgentWithID 메서드(tapi3cc.h)는 에이전트 식별자를 기반으로 에이전트 개체를 만듭니다.
CreateCall CreateCall 메서드는 나가는 호출을 만드는 데 사용할 수 있는 새 Call 개체를 만들고 개체의 ITBasicCallControl 인터페이스에 대한 포인터를 반환합니다. 새로 만든 호출은 CS_IDLE 상태이며 미디어 또는 터미널이 선택되지 않았습니다.
CreateCustomToneObject CreateCustomToneObject 메서드는 GenerateCustomTonesByCollection 메서드와 함께 사용할 사용자 지정 톤 개체를 만듭니다.
CreateDetectToneObject CreateDetectToneObject 메서드는 DetectTonesByCollection 메서드와 함께 사용할 검색 톤 개체를 만듭니다.
CreateDirectory

CreateDirectory 메서드는 지정된 형식 및 이름의 디렉터리에 해당하는 ITDirectory 개체를 만듭니다.
CreateDirectoryObject CreateDirectoryObject 메서드는 새 ITDirectoryObject 개체를 만듭니다.
CreateDynamicTerminal CreateDynamicTerminal 메서드는 지정된 터미널 클래스, 미디어 형식 및 방향의 동적 터미널을 만듭니다.
CreateEmptyCollectionObject CreateEmptyCollectionObject 메서드는 빈 컬렉션 개체를 만듭니다. 컬렉션은 DetectTonesByCollection 메서드 또는 GenerateCustomTonesByCollection 메서드와 함께 사용할 ITDetectTone 또는 ITCustomTone 개체로 채워질 수 있습니다.
CreateForwardInfoObject CreateForwardInfoObject 메서드는 전달 정보 개체를 만들고 ITForwardInformation 인터페이스 포인터를 반환합니다.
CreateLeaseInfo CreateLeaseInfo 메서드는 RenewAddress 또는 ReleaseAddress에 대한 후속 호출에 대한 임대 정보 개체를 만듭니다.
CreateLeaseInfoFromVariant CreateLeaseInfoFromVariant 메서드는 RenewAddress 또는 ReleaseAddress에 대한 후속 호출에 대한 임대 정보 개체를 만듭니다. 이 메서드는 CreateLeaseInfo와 비슷하지만 Visual Basic과 같은 Automation 클라이언트 언어에서 사용됩니다.
CreateMSPCall ITMSPAddress::CreateMSPCall(tapi3.h) 메서드는 MSP Call 개체를 만듭니다. TAPI는 이를 기본 Call 개체에 집계하고 ITStreamControl 인터페이스를 노출합니다.
CreateMSPCall ITMSPAddress::CreateMSPCall(msp.h) 메서드는 MSP Call 개체를 만듭니다.
CreateMSPCallHelper CreateMSPCallHelper 도우미 템플릿 함수는 CreateMSPCall의 파생 클래스 구현에서 호출됩니다.
CreateSession

ITAgent::CreateSession 메서드(tapi3.h)는 입력 ACD 그룹 및 주소에 대한 새 에이전트 세션을 만듭니다.
CreateSession ITAgent::CreateSession 메서드(tapi3cc.h)는 입력 ACD 그룹 및 주소에 대한 새 에이전트 세션을 만듭니다.
CreateSessionWithPIN ITAgent::CreateSessionWithPIN 메서드(tapi3.h)는 PIN(개인 식별 번호)을 사용하여 입력 ACD 그룹 및 주소에 대한 새 에이전트 세션을 만듭니다.
CreateSessionWithPIN ITAgent::CreateSessionWithPIN 메서드(tapi3cc.h)는 PIN(개인 식별 번호)을 사용하여 입력 ACD 그룹 및 주소에 대한 새 에이전트 세션을 만듭니다.
CreateStream CreateStream 메서드는 새 미디어 스트림을 만듭니다.
CreateStreamObject CreateStreamObject 메서드는 InternalCreateStream에서 호출됩니다.
CreateSubStream CreateSubStream 메서드는 하위 스트림을 만듭니다.
CreateTerminal CreateTerminal 메서드는 동적 터미널 클래스 및 미디어를 기반으로 새 ITTerminal 개체를 만들고 초기화합니다. 터미널 클래스는 GUID로 식별됩니다. 이 메서드에 전달하려면 STRINGFromIID를 사용하여 GUID를 문자열로 변환해야 합니다.
CreateTrackTerminal CreateTrackTerminal 메서드는 지정된 미디어 유형 또는 형식 및 미디어 방향을 처리할 수 있는 멀티트랙 터미널을 만듭니다.
삭제 Delete 메서드는 레지스트리에서 터미널 클래스를 삭제합니다.
삭제 Delete 메서드는 레지스트리에서 현재 터미널 슈퍼클래스를 제거합니다.

DeleteDirectoryObject DeleteDirectoryObject 메서드는 서버에서 개체를 삭제합니다.
DetectDigits DetectDigits 메서드는 회전 펄스 또는 DTMF와 같이 현재 호출에서 검색될 숫자 형식의 식별자를 설정합니다.
DetectTones DetectTones 메서드는 호출 시 인밴드 톤 검색을 활성화하고 사용하지 않도록 설정합니다. 지정된 톤이 검색될 때마다 메시지가 애플리케이션으로 전송됩니다.
DetectTonesByCollection DetectTonesByCollection 메서드는 호출 시 인밴드 톤 검색을 활성화하고 사용하지 않도록 설정합니다. 지정된 톤이 검색될 때마다 메시지가 애플리케이션으로 전송됩니다.
DeviceSpecific DeviceSpecific 메서드를 사용하면 서비스 공급자가 다른 TAPI 함수에서 제공하지 않는 기능에 액세스할 수 있습니다.
DeviceSpecific DeviceSpecific 메서드를 사용하면 서비스 공급자가 다른 TAPI 함수에서 제공하지 않는 디바이스별 기능에 대한 액세스를 제공할 수 있습니다.
DeviceSpecificVariant DeviceSpecificVariant 메서드를 사용하면 서비스 공급자가 다른 TAPI 함수에서 제공하지 않는 기능에 대한 액세스를 제공할 수 있습니다. (ITAddress2.DeviceSpecificVariant)
DeviceSpecificVariant DeviceSpecificVariant 메서드를 사용하면 서비스 공급자가 다른 TAPI 함수에서 제공하지 않는 기능에 대한 액세스를 제공할 수 있습니다. (ITPhone.DeviceSpecificVariant)
다이얼 Dial 메서드는 지정된 주소를 다이얼합니다.
연결 끊기 Disconnect 메서드는 호출의 연결을 끊습니다. 메서드가 성공적으로 완료되면 호출 상태가 CS_DISCONNECTED 전환됩니다.

DisconnectTerminal DisconnectTerminal 메서드는 내부 필터의 연결을 끊고 필터 그래프에서 제거합니다.
DispatchGraphEvent DispatchGraphEvent 메서드는 초기화 중에 RegisterWaitForSingleObject 함수에 게시된 정적 메서드입니다.
EnableAutoRefresh EnableAutoRefresh 메서드는 호출된 후 생성된 디렉터리 개체에 대해 자동 새로 고침을 사용하도록 설정합니다. 동적 서버에만 적용됩니다.
EnumerateACDGroups EnumerateACDGroups 메서드(tapi3.h)는 현재 에이전트 처리기와 연결된 ACD 그룹을 열거합니다.
EnumerateACDGroups ITAgentHandler::EnumerateACDGroups 메서드(tapi3cc.h)는 현재 에이전트 처리기와 연결된 ACD 그룹을 열거합니다.
EnumerateAddresses EnumerateAddresses 메서드는 휴대폰을 사용할 수 있는 주소를 열거합니다. 이 메서드를 실행하기 전에 애플리케이션에서 ITPhone::Open을 호출할 필요가 없습니다.
EnumerateAddresses EnumerateAddresses 메서드는 현재 사용할 수 있는 주소를 열거합니다. C 및 C++ 애플리케이션에 대해 제공됩니다. Visual Basic으로 작성된 애플리케이션과 같은 Automation 클라이언트 애플리케이션은 get_Addresses 메서드를 사용해야 합니다.
EnumerateAddresses EnumerateAddresses 메서드는 이 임대 또는 임대 요청의 대상이 되는 멀티캐스트 주소의 컬렉션을 가져옵니다. 이 메서드는 주로 C++ 프로그래머용입니다. Visual Basic 및 기타 스크립팅 언어는 대신 get_Addresses 사용합니다.
EnumerateAgentHandlers ITTAPICallCenter::EnumerateAgentHandlers(tapi3.h) 메서드는 현재 콜 센터와 연결된 에이전트 처리기를 열거합니다.
EnumerateAgentHandlers ITTAPICallCenter::EnumerateAgentHandlers 메서드(tapi3cc.h)는 현재 콜 센터와 연결된 에이전트 처리기를 열거합니다.

EnumerateAgentSessions ITAgent::EnumerateAgentSessions 메서드(tapi3.h)는 현재 에이전트 세션을 열거합니다.
EnumerateAgentSessions ITAgent::EnumerateAgentSessions 메서드(tapi3cc.h)는 현재 에이전트 세션을 열거합니다.
EnumerateCallHubs EnumerateCallHubs 메서드는 현재 사용 가능한 호출 허브를 열거합니다. C 및 C++ 애플리케이션에 대해 제공됩니다. Visual Basic으로 작성된 애플리케이션과 같은 Automation 클라이언트 애플리케이션은 get_Callhubs 메서드를 사용해야 합니다.
EnumerateCallingCards EnumerateCallingCards 메서드는 주소와 연결된 호출 카드를 열거합니다. 이 메서드는 C 및 C++ 애플리케이션에 대해 제공됩니다. Visual Basic으로 작성된 애플리케이션과 같은 Automation 클라이언트 애플리케이션은 get_CallingCards 메서드를 사용해야 합니다.
EnumerateCalls EnumerateCalls 메서드는 현재 주소에 대한 호출을 열거합니다. 이 메서드는 C 및 C++ 애플리케이션에 대해 제공됩니다. Visual Basic으로 작성된 애플리케이션과 같은 Automation 클라이언트 애플리케이션은 get_Calls 메서드를 사용해야 합니다.
EnumerateCalls EnumerateCalls 메서드는 현재 호출 허브와 연결된 호출을 열거합니다. 이 메서드는 C 및 C++ 애플리케이션에 대해 제공됩니다. Visual Basic으로 작성된 애플리케이션과 같은 Automation 클라이언트 애플리케이션은 get_Calls 메서드를 사용해야 합니다.
EnumerateCallTreatments EnumerateCallTreatments 메서드는 호출 처리를 가져옵니다. 이 메서드는 C/C++ 및 Java로 작성된 애플리케이션에 대해 제공됩니다.
EnumerateCompletionMessages EnumerateCompletionMessages 메서드는 완료 메시지를 가져옵니다. 이 메서드는 C/C++ 및 Java로 작성된 애플리케이션에 대해 제공됩니다.
EnumerateDefaultDirectories EnumerateDefaultDirectories 메서드는 구성된 모든 기본 디렉터리를 열거합니다. 이 메서드는 get_DefaultDirectories 비슷하지만 C/C++용으로 설계되었습니다.
EnumerateDeviceClasses

EnumerateDeviceClasses 메서드는 디바이스 클래스를 가져옵니다. 이 메서드는 C/C++ 및 Java로 작성된 애플리케이션에 대해 제공됩니다.
EnumerateDialableAddrs EnumerateDialableAddrs 메서드는 디렉터리에서 지정된 형식의 모든 전화 걸기 가능한 주소의 열거자를 만듭니다.
EnumerateDirectoryObjects EnumerateDirectoryObjects 메서드는 지정된 형식 및 이름의 디렉터리 개체 열거형을 만듭니다.
EnumerateDynamicTerminalClasses EnumerateDynamicTerminalClasses 메서드는 지원되는 현재 사용 가능한 동적 터미널 클래스를 열거합니다.
EnumerateLocations EnumerateLocations 메서드는 현재 사용 가능한 주소 위치를 열거합니다. 이 메서드는 C 및 C++ 애플리케이션에 대해 제공됩니다. Visual Basic으로 작성된 애플리케이션과 같은 Automation 클라이언트 애플리케이션은 get_Locations 메서드를 사용해야 합니다.
EnumeratePhones EnumeratePhones 메서드는 이 주소와 함께 사용할 수 있는 휴대폰 디바이스에 해당하는 휴대폰 개체를 열거합니다.
EnumeratePhones EnumeratePhones 메서드는 휴대폰 디바이스에 해당하는 휴대폰 개체를 열거합니다. 주소와 함께 사용할 수 있는 휴대폰이 없는 경우 이 메서드는 빈 열거형을 생성하고 S_OK 반환합니다.
EnumeratePluggableSuperclasses EnumeratePluggableSuperclasses 메서드는 현재 시스템에 등록된 플러그형 터미널 슈퍼클래스를 열거합니다.
EnumeratePluggableTerminalClasses EnumeratePluggableTerminalClasses 메서드는 지정된 슈퍼클래스에 등록된 플러그형 터미널 클래스를 열거합니다.
EnumeratePreferredAddresses EnumeratePreferredAddresses 메서드는 전화 개체의 기본 주소를 열거합니다. 이 메서드를 실행하기 전에 애플리케이션에서 ITPhone::Open을 호출할 필요가 없습니다.
EnumeratePreferredPhones

EnumeratePreferredPhones 메서드는 이 주소에 대한 기본 전화 개체를 열거합니다.
EnumeratePrivateTAPIObjects ITTAPI 인터페이스의 EnumeratePrivateTAPIObjects 메서드는 구현되지 않으며 E_NOTIMPL 반환합니다.
EnumerateQueues ITACDGroup::EnumerateQueues 메서드(tapi3.h)는 현재 ACD 그룹에 있는 큐를 열거합니다. 이 메서드는 C 및 C++ 애플리케이션에 대해 제공됩니다.
EnumerateQueues ITACDGroup::EnumerateQueues 메서드(tapi3cc.h)는 현재 ACD 그룹에 있는 큐를 열거합니다.
EnumerateScopes EnumerateScopes 메서드는 사용 가능한 멀티캐스트 범위의 열거형을 만듭니다. 이 메서드는 주로 C++ 프로그래머용입니다. Visual Basic 및 기타 스크립팅 언어는 대신 get_Scopes 사용합니다.
EnumerateSelectedCalls EnumerateSelectedCalls 메서드는 현재 이 휴대폰에서 선택된 호출을 나타내는 열거자 개체를 검색합니다. 자세한 내용은 ITAutomatedPhoneControl::SelectCall을 참조하세요.
EnumerateStaticTerminals EnumerateStaticTerminals 메서드는 주소와 연결된 현재 사용 가능한 정적 터미널을 열거합니다.
EnumerateStreams EnumerateStreams 메서드는 현재 사용 가능한 미디어 스트림을 열거합니다. C 및 C++ 애플리케이션에 제공됩니다. Visual Basic과 같은 Automation 클라이언트 애플리케이션은 get_Streams 메서드를 사용해야 합니다.
EnumerateSubStreams EnumerateSubStreams 메서드는 현재 사용 가능한 미디어 하위 스트림을 열거합니다. C 및 C++ 애플리케이션에 제공됩니다. Visual Basic과 같은 Automation 클라이언트 애플리케이션은 get_SubStreams 메서드를 사용해야 합니다.
EnumerateTerminalClasses EnumerateTerminalClasses 메서드는 현재 터미널 슈퍼클래스에 대한 모든 터미널 클래스를 나열합니다.
열거형터미널

EnumerateTerminals 메서드는 휴대폰과 연결된 터미널의 열거형을 검색합니다. 이 메서드를 실행하기 전에 애플리케이션에서 ITPhone::Open을 호출할 필요가 없습니다.
열거형터미널 EnumerateTerminals 메서드는 스트림에서 선택한 터미널을 열거합니다. C 및 C++ 애플리케이션에 제공됩니다. Visual Basic과 같은 Automation 클라이언트 애플리케이션은 get_Terminals 메서드를 사용해야 합니다.
열거형터미널 EnumerateTerminals 메서드는 하위 스트림에서 선택한 터미널을 열거합니다. C 및 C++ 애플리케이션에 제공됩니다. Visual Basic과 같은 Automation 클라이언트 애플리케이션은 get_Terminals 메서드를 사용해야 합니다.
EnumerateTrackTerminals EnumerateTrackTerminals 메서드는 이 메서드가 호출된 멀티트랙 터미널에 포함된 터미널이 포함된 열거형을 만들고 반환합니다.
EnumerateUsableAddresses EnumerateUsableAddresses 메서드(tapi3.h)는 이 에이전트 처리기에서 ACD 호출을 수신하는 데 사용할 수 있는 주소를 열거합니다.
EnumerateUsableAddresses ITAgentHandler::EnumerateUsableAddresses 메서드(tapi3cc.h)는 이 에이전트 처리기에서 ACD 호출을 수신하는 데 사용할 수 있는 주소를 열거합니다.
이벤트 Event 메서드는 비동기 이벤트 알림에 대한 응답을 결정하기 위해 TAPI에서 호출됩니다.
FinalRelease FinalRelease 메서드는 필터 그래프 포인터를 해제합니다.
마침 Finish 메서드는 회의 또는 이전을 완료하기 위해 상담 전화에서 호출됩니다.
FireEvent ITPluggableTerminalEventSink::FireEvent(tapi3.h) 메서드는 플러그형 터미널의 변경 사항을 클라이언트 애플리케이션에 알린 메시지를 생성합니다.
FireEvent

ITPluggableTerminalEventSink::FireEvent(msp.h) 메서드는 플러그형 터미널의 변경 사항을 클라이언트 애플리케이션에 알린 메시지를 생성합니다.
앞으로 Forward 메서드는 ITForwardInformation에 포함된 전달 지침에 따라 주소로 향하는 호출을 전달합니다. pForwardInfo가 NULL로 설정된 경우 전달이 취소됩니다.
GatherDigits GatherDigits 메서드는 지정된 호출에서 숫자 수집을 시작합니다. 애플리케이션은 수집할 최대 자릿수를 지정합니다.
GenerateCustomTones GenerateCustomTones 메서드는 지정된 사용자 지정 톤을 생성합니다.
GenerateCustomTonesByCollection GenerateCustomTonesByCollection 메서드는 지정된 사용자 지정 톤을 생성합니다.
GenerateDigits GenerateDigits 메서드를 사용하면 현재 호출에서 숫자가 출력됩니다.
GenerateDigits2 GenerateDigits2 메서드를 사용하면 현재 호출에서 숫자가 출력됩니다. 이 메서드는 duration 매개 변수를 추가하여 ITLegacyCallMediaControl::GenerateDigits 메서드를 확장합니다.
GenerateTone GenerateTone 메서드는 지정된 톤을 생성합니다.
get__NewEnum get__NewEnum 메서드는 컬렉션에 대한 열거자를 가져옵니다.
get_ACDCallRate get_ACDCallRate 메서드(tapi3.h)는 시간당 호출에서 에이전트 세션당 모든 속도를 가져옵니다.
get_ACDCallRate ITAgentSession::get_ACDCallRate 메서드(tapi3cc.h)는 에이전트 세션당 모든 속도를 시간당 호출로 가져옵니다.
get_ACDGroup

get_ACDGroup 메서드(tapi3.h)는 이 세션과 연결된 ACD 그룹을 가져옵니다.
get_ACDGroup ITAgentSession::get_ACDGroup 메서드(tapi3cc.h)는 이 세션과 연결된 ACD 그룹을 가져옵니다.
get_ACDGroups get_ACDGroups 메서드(tapi3.h)는 현재 에이전트 처리기와 연결된 ACD 그룹의 컬렉션을 만듭니다.
get_ACDGroups ITAgentHandler::get_ACDGroups 메서드(tapi3cc.h)는 현재 에이전트 처리기와 연결된 ACD 그룹의 컬렉션을 만듭니다.
get_Address ITAgentSession::get_Address(tapi3.h) 메서드는 이 세션과 연결된 ITAddress 인터페이스에 대한 포인터를 가져옵니다.
get_Address get_Address 메서드는 이벤트에 관련된 Address 개체의 ITAddress 인터페이스에 대한 포인터를 가져옵니다.
get_Address get_Address 메서드는 이벤트에 관련된 ITAddress 개체에 대한 포인터를 가져옵니다.
get_Address get_Address 메서드는 Address 개체의 ITAddress 인터페이스에 대한 포인터를 가져옵니다.
get_Address get_Address 메서드는 이벤트에 관련된 프라이빗 개체에 대한 ITAddress 인터페이스 포인터를 가져옵니다.
get_Address get_Address 메서드는 이벤트가 발생한 Address 개체에 대한 포인터를 가져옵니다.
get_Address ITAgentSession::get_Address 메서드(tapi3cc.h)는 이 세션과 연결된 ITAddress 인터페이스에 대한 포인터를 가져옵니다.
get_AddressCapability

get_AddressCapability 메서드는 지정된 ADDRESS_CAPABILITY 대한 기능 값을 가져옵니다.
get_AddressCapabilityString get_AddressCapabilityString 메서드는 지정된 ADDRESS_CAPABILITY_STRING 대한 기능 문자열을 가져옵니다.
get_AddressCount get_AddressCount 메서드는 이 임대에서 요청되거나 부여된 주소 수를 가져옵니다.
get_Addresses get_Addresses 메서드는 휴대폰을 사용할 수 있는 주소 컬렉션을 반환합니다. 이 메서드를 실행하기 전에 애플리케이션에서 ITPhone::Open을 호출할 필요가 없습니다.
get_Addresses get_Addresses 메서드는 현재 사용할 수 있는 주소 컬렉션을 만듭니다. Visual Basic으로 작성된 애플리케이션과 같은 Automation 클라이언트 애플리케이션에 대해 제공됩니다. C 및 C++ 애플리케이션은 EnumerateAddresses 메서드를 사용해야 합니다.
get_Addresses get_Addresses 메서드는 이 임대 또는 임대 요청의 대상이 되는 멀티캐스트 주소의 컬렉션을 가져옵니다. 이 메서드는 주로 Visual Basic 및 기타 스크립팅 언어용입니다. C++ 프로그래머는 EnumerateAddresses를 대신 사용합니다.
get_AddressHandle get_AddressHandle 메서드는 MSP 주소 핸들을 가져옵니다.
get_AddressName get_AddressName 메서드는 주소의 표시 가능한 이름을 가져옵니다.
get_AdvertisingScope get_AdvertisingScope 방법은 광고 scope 가져옵니다.
get_Agent ITAgentEvent::get_Agent 메서드(tapi3.h)는 이벤트가 발생한 에이전트에 대한 인터페이스를 가져옵니다.
get_Agent ITAgentSession::get_Agent(tapi3.h) 메서드는 이 세션과 연결된 ITAgent 인터페이스에 대한 포인터를 가져옵니다.

get_Agent ITAgentEvent::get_Agent 메서드(tapi3cc.h)는 이벤트가 발생한 에이전트에 대한 인터페이스를 가져옵니다.
get_Agent ITAgentSession::get_Agent 메서드(tapi3cc.h)는 이 세션과 연결된 ITAgent 인터페이스에 대한 포인터를 가져옵니다.
get_AgentHandler get_AgentHandler 메서드(tapi3.h)는 ITAgentHandler 인터페이스 포인터를 가져옵니다.
get_AgentHandler ITAgentHandlerEvent::get_AgentHandler 메서드(tapi3cc.h)는 ITAgentHandler 인터페이스 포인터를 가져옵니다.
get_AgentHandlers ITTAPICallCenter::get_AgentHandlers(tapi3.h) 메서드는 현재 콜 센터와 연결된 에이전트 처리기의 컬렉션을 만듭니다.
get_AgentHandlers ITTAPICallCenter::get_AgentHandlers 메서드(tapi3cc.h)는 현재 콜 센터와 연결된 에이전트 처리기의 컬렉션을 만듭니다.
get_AgentSessions ITAgent::get_AgentSessions 메서드(tapi3.h)는 현재 에이전트 세션의 컬렉션을 만듭니다.
get_AgentSessions ITAgent::get_AgentSessions 메서드(tapi3cc.h)는 현재 에이전트 세션의 컬렉션을 만듭니다.
get_AppName get_AppName 메서드는 애플리케이션의 이름을 가져옵니다.
get_AppSpecific get_AppSpecific 메서드는 검색할 톤을 식별하는 애플리케이션 정의 태그를 검색합니다.
get_AppSpecific get_AppSpecific 메서드는 톤 감지 이벤트와 연결된 톤을 식별하는 애플리케이션 정의 태그를 가져옵니다.

get_AudioFormatForScripting get_AudioFormatForScripting 메서드는 오디오 스크립팅 형식을 가져옵니다.
get_AutoDialtone get_AutoDialtone 메서드는 AutoDialtone 속성의 현재 값을 검색합니다.
get_AutoEndOfNumberTimeout get_AutoEndOfNumberTimeout 메서드는 AutoEndOfNumberTimeout 속성의 현재 값을 검색합니다. 속성은 전체 숫자가 수집되었다고 가정하기 전에 숫자를 누른 후 대기하는 시간을 지정합니다.
get_AutoKeypadTones get_AutoKeypadTones 메서드는 이 휴대폰의 AutoKeypadTones 속성을 가져옵니다. 이 기능을 사용하도록 설정하면 키패드 단추를 누를 때마다 숫자 톤이 자동으로 재생됩니다.
get_AutoKeypadTonesMinimumDuration get_AutoKeypadTonesMinimumDuration 메서드는 AutoKeypadTonesMinimumDuration 속성의 현재 값을 검색합니다. 속성은 PBS_DOWN 키패드 톤을 재생할 기간을 지정합니다.
get_AutoStopRingOnOffHook get_AutoStopRingOnOffHook 메서드는 AutoStopRingOnOffHook 속성의 현재 값을 검색합니다.
get_AutoStopTonesOnOnHook get_AutoStopTonesOnOnHook 메서드는 AutoStopTonesOnHook 속성의 현재 값을 검색합니다.
get_AutoVolumeControl get_AutoVolumeControl 메서드는 AutoVolumeControl 속성의 현재 값을 검색합니다.
get_AutoVolumeControlRepeatDelay get_AutoVolumeControlRepeatDelay 메서드는 AutoVolumeControlRepeatDelay 속성의 현재 값을 검색합니다. 속성은 볼륨 단추가 누른 상태에서 반복되기 전에 지연 시간(밀리초)을 지정합니다.
get_AutoVolumeControlRepeatPeriod get_AutoVolumeControlRepeatPeriod 메서드는 AutoVolumeControlRepeatPeriod 속성의 현재 값을 검색합니다. 속성은 볼륨 단추를 누를 때 단추가 반복되는 기간을 밀리초(밀리초)로 제어합니다.
get_AutoVolumeControlStep get_AutoVolumeControlStep 메서드는 AutoVolumeControlStep 속성의 현재 값을 검색합니다. 속성은 볼륨 단추를 누를 때 휴대폰 볼륨이 조정되는 양을 결정합니다.

get_AverageCallTime ITAgentSession::get_AverageCallTime(tapi3.h) 메서드는 이 에이전트 세션 동안 ACD 호출당 소요된 평균 시간(초)을 가져옵니다.
get_AverageCallTime ITAgentSession::get_AverageCallTime 메서드(tapi3cc.h)는 이 에이전트 세션 동안 ACD 호출당 소요된 평균 시간(초)을 가져옵니다.
get_AverageTalkTime ITAgentSession::get_AverageTalkTime(tapi3.h) 메서드는 이 에이전트 세션 동안(이 에이전트가) ACD 호출당 통신하는 데 소요된 평균 시간을 가져옵니다.
get_AverageTalkTime ITAgentSession::get_AverageTalkTime 메서드(tapi3cc.h)는 이 에이전트 세션 동안 ACD 호출당 통신하는 데 소요된 평균 시간(초)을 가져옵니다.
get_AverageTimeToAnswer ITAgentSession::get_AverageTimeToAnswer(tapi3.h) 메서드는 호출이 응답되기 위해 대기한 평균 시간(초)을 가져옵니다.
get_AverageTimeToAnswer ITAgentSession::get_AverageTimeToAnswer 메서드(tapi3cc.h)는 호출이 응답되기 위해 대기한 평균 시간(초)을 가져옵니다.
get_AverageWaitTime ITQueue::get_AverageWaitTime(tapi3.h) 메서드는 현재 측정 기간 동안 호출에 대한 큐의 평균 시간(초)을 가져옵니다.
get_AverageWaitTime ITQueue::get_AverageWaitTime 메서드(tapi3cc.h)는 현재 측정 기간 동안 호출에 대한 큐의 평균 시간(초)을 가져옵니다.
get_AverageWrapUpTime ITAgentSession::get_AverageWrapUpTime(tapi3.h) 메서드는 이 에이전트 세션 동안 래핑(통화 후 작업)에 소요된 ACD 호출당 평균 시간을 가져옵니다.
get_AverageWrapUpTime ITAgentSession::get_AverageWrapUpTime 메서드(tapi3cc.h)는 이 에이전트 세션 동안 래핑에 소요된 ACD 호출당 평균 시간(초)을 가져옵니다.

get_AvgBytesPerSec get_AvgBytesPerSec 메서드는 WAVEFORMATEX 구조체에서 nAvgBytesPerSec 멤버의 값을 반환합니다.
get_Balance get_Balance 메서드는 균형을 가져옵니다. 이 메서드는 TAPI 3.0 이상과 함께 제공되는 터미널에 대해 구현되지 않습니다.
get_BitsPerSample get_BitsPerSample 메서드는 WAVEFORMATEX 구조체에서 wBitsPerSample 멤버의 값을 반환합니다.
get_BlockAlign get_BlockAlign 메서드는 WAVEFORMATEX 구조체에서 nBlockAlign 멤버의 값을 반환합니다.
get_ButtonFunction get_ButtonFunction 메서드는 특정 단추와 연결된 단추 함수를 검색합니다.
get_ButtonLampId get_ButtonLampId 메서드는 이벤트를 트리거한 단추 또는 램프를 나타내는 긴 값을 반환합니다. 이 정보는 ITPhoneEvent::get_Event PE_LAMPMODE 또는 PE_BUTTON 반환하는 경우에만 사용할 수 있습니다.
get_ButtonMode get_ButtonMode 메서드는 특정 단추와 연결된 단추 모드를 검색합니다.
get_ButtonState get_ButtonState 메서드는 특정 단추와 연결된 단추 상태를 검색합니다.
get_ButtonState get_ButtonState 메서드는 단추가 전환된 상태를 지정하는 PHONE_BUTTON_STATE 값을 반환합니다. 이 정보는 ITPhoneEvent::get_Event 메서드가 PE_BUTTON 반환하는 경우에만 사용할 수 있습니다.
get_ButtonText get_ButtonText 메서드는 특정 단추와 연결된 단추 텍스트를 검색합니다.
get_CadenceOff

get_CadenceOff 메서드는 생성할 사용자 지정 톤 주기의 "off" 기간을 검색합니다.
get_CadenceOn get_CadenceOn 메서드는 생성할 사용자 지정 톤 주기의 "on" 기간을 검색합니다.
get_Call get_Call 메서드는 이벤트에 관련된 호출 개체에 대한 ITCallInfo 인터페이스 포인터에 대한 포인터를 가져옵니다.
get_Call get_Call 메서드는 터미널 이벤트에 관련된 호출 개체에 대한 ITCallInfo 인터페이스에 대한 포인터를 반환합니다.
get_Call get_Call 메서드는 이벤트가 발생한 호출의 ITCallInfo 인터페이스에 대한 포인터를 반환합니다. (ITCallHubEvent.get_Call)
get_Call get_Call 메서드는 호출 정보가 변경된 ITCallInfo 인터페이스를 반환합니다.
get_Call get_Call 메서드는 이 이벤트와 연결된 호출 개체에 대한 ITCallInfo 인터페이스 포인터를 가져옵니다.
get_Call get_Call 메서드는 호출 이벤트가 발생한 ITCallInfo 인터페이스를 반환합니다.
get_Call get_Call 메서드는 이벤트가 발생한 호출에 대한 호출 정보 인터페이스에 대한 포인터를 가져옵니다. (ITCallStateEvent.get_Call)
get_Call get_Call 메서드는 이벤트가 발생한 호출에 대한 ITCallInfo 인터페이스에 대한 포인터를 가져옵니다.
get_Call get_Call 메서드는 이벤트가 필요한 호출에 대한 ITCallInfo 인터페이스 포인터를 반환합니다.
get_Call

get_Call 메서드는 이벤트가 발생한 호출 개체의 호출 정보 인터페이스에 대한 포인터를 가져옵니다.
get_Call get_Call 메서드는 이벤트가 발생한 호출에 대한 호출 정보 인터페이스에 대한 포인터를 가져옵니다. (ITFileTerminalEvent.get_Call)
get_Call get_Call 메서드는 이벤트에 관련된 호출 개체에 대한 ITCallInfo 인터페이스에 대한 포인터를 반환합니다.
get_Call get_Call 메서드는 이벤트가 발생한 호출의 ITCallInfo 인터페이스에 대한 포인터를 반환합니다. (ITPrivateEvent.get_Call)
get_Call get_Call 메서드는 QOS 이벤트가 발생한 호출에 대한 ITCallInfo 인터페이스에 대한 포인터를 가져옵니다.
get_Call get_Call 메서드는 톤 검색 이벤트가 발생한 호출 개체의 호출 정보 인터페이스에 대한 포인터를 가져옵니다.
get_Call get_Call 메서드는 이벤트가 발생한 호출 개체에 대한 인터페이스 포인터를 검색합니다.
get_Call get_Call 메서드는 터미널 이벤트에 관련된 호출에 대한 ITCallInfo 인터페이스 포인터를 반환합니다.
get_CallbackInstance get_CallbackInstance 메서드는 이 이벤트와 연결된 콜백 instance 대한 포인터를 가져옵니다. (ITCallInfoChangeEvent.get_CallbackInstance)
get_CallbackInstance get_CallbackInstance 메서드는 이 이벤트와 연결된 콜백 instance 대한 포인터를 가져옵니다. (ITCallNotificationEvent.get_CallbackInstance)
get_CallbackInstance

get_CallbackInstance 메서드는 이 이벤트와 연결된 콜백 instance 대한 포인터를 가져옵니다. (ITCallStateEvent.get_CallbackInstance)
get_CallbackInstance get_CallbackInstance 메서드는 이벤트와 연결된 콜백 instance 대한 포인터를 가져옵니다. (ITDigitDetectionEvent.get_CallbackInstance)
get_CallbackInstance get_CallbackInstance 메서드는 이벤트와 연결된 콜백 instance 대한 포인터를 가져옵니다. (ITDigitGenerationEvent.get_CallbackInstance)
get_CallbackInstance get_CallbackInstance 메서드는 이벤트를 처리할 애플리케이션의 콜백 함수에 대한 포인터를 가져옵니다. (ITDigitsGatheredEvent.get_CallbackInstance)
get_CallbackInstance get_CallbackInstance 메서드는 이벤트와 연결된 콜백 instance 대한 포인터를 가져옵니다. (ITTAPIObjectEvent.get_CallbackInstance)
get_CallbackInstance get_CallbackInstance 메서드는 이벤트를 처리할 애플리케이션의 콜백 함수에 대한 포인터를 가져옵니다. (ITToneDetectionEvent.get_CallbackInstance)
get_CalledParty get_CalledParty 메서드는 호출된 파티를 가져옵니다.
get_CallHub get_CallHub 메서드는 이벤트가 발생한 ITCallHub 인터페이스에 대한 포인터를 반환합니다. (ITCallHubEvent.get_CallHub)
get_CallHub get_CallHub 메서드는 CallHub 개체의 ITCallHub 인터페이스에 대한 포인터를 가져옵니다.
get_CallHub get_CallHub 메서드는 이벤트가 발생한 ITCallHub 인터페이스에 대한 포인터를 반환합니다. (ITPrivateEvent.get_CallHub)
get_CallHubs get_CallHubs 메서드는 현재 사용 가능한 호출 허브의 컬렉션을 만듭니다. Visual Basic으로 작성된

애플리케이션과 같은 Automation 클라이언트 애플리케이션에 대해 제공됩니다. C 및 C++ 애플리케이션은 EnumerateCallHubs 메서드를 사용해야 합니다.
get_CallInfoBuffer get_CallInfoBuffer 메서드는 사용자-사용자 정보와 같이 버퍼가 필요한 호출 정보 항목을 가져옵니다.
get_CallInfoLong get_CallInfoLong 메서드는 전달자 모드와 같이 long에서 설명하는 호출 정보 항목을 가져옵니다.
get_CallInfoString get_CallInfoString 메서드는 표시 가능한 주소와 같이 문자열에서 설명하는 호출 정보 항목을 가져옵니다.
get_CallingCards get_CallingCards 메서드는 주소와 연결된 호출 카드 컬렉션을 만듭니다.
get_Calls get_Calls 메서드는 주소에서 현재 활성 상태인 호출 컬렉션을 만듭니다. 이 메서드는 Visual Basic으로 작성된 애플리케이션과 같은 Automation 클라이언트 애플리케이션에 대해 제공됩니다. C 및 C++ 애플리케이션은 EnumerateCalls 메서드를 사용해야 합니다.
get_Calls get_Calls 메서드는 현재 호출 허브와 연결된 호출 컬렉션을 만듭니다. 이 메서드는 Visual Basic으로 작성된 애플리케이션과 같은 Automation 클라이언트 애플리케이션에 대해 제공됩니다. C 및 C++ 애플리케이션은 EnumerateCalls 메서드를 사용해야 합니다.
get_CallState get_CallState 메서드는 CS_IDLE 같은 현재 호출 상태에 대한 포인터를 가져옵니다.
get_CallTreatments get_CallTreatments 메서드는 호출 처리를 가져옵니다. 이 메서드는 Visual Basic 및 스크립팅 언어로 작성된 애플리케이션과 같은 Automation 클라이언트 애플리케이션에 대해 제공됩니다.
get_CancelCallWaitingCode get_CancelCallWaitingCode 메서드는 통화 대기를 취소하기 위해 전화 걸기 문자열 앞에 접두사를 지정해야 하는 다이얼 숫자와 한정자를 가져옵니다.
get_CardName

get_CardName 메서드는 호출 카드 대한 이름을 가져옵니다.
get_Cause get_Cause 메서드는 변경에 대한 CALLINFOCHANGE_CAUSE 설명을 가져옵니다.
get_Cause get_Cause 메서드는 렌더러 디바이스의 시간 제한과 같은 호출 미디어 이벤트의 원인을 가져옵니다.
get_Cause get_Cause 메서드는 이 이벤트와 연결된 원인을 가져옵니다. (ITCallStateEvent.get_Cause)
get_Cause get_Cause 메서드는 이 이벤트와 연결된 원인을 가져옵니다. (ITFileTerminalEvent.get_Cause)
get_Channels get_Channels 메서드는 WAVEFORMATEX 구조체의 nChannels 멤버에 대한 값을 반환합니다.
get_CityCode get_CityCode 메서드는 도시에 대한 코드를 가져옵니다.
get_CLSID get_CLSID 메서드는 터미널을 CoCreateInstance하는 데 사용되는 CLSID를 가져옵니다. (ITPluggableTerminalClassInfo.get_CLSID)
get_CLSID get_CLSID 메서드는 터미널을 CoCreateInstance하는 데 사용되는 CLSID를 가져옵니다. (ITPluggableTerminalSuperclassInfo.get_CLSID)
get_CLSID get_CLSID 메서드는 터미널을 CoCreateInstance하는 데 사용되는 CLSID를 가져옵니다. (ITPluggableTerminalClassRegistration.get_CLSID)
get_CLSID get_CLSID 메서드는 터미널을 CoCreateInstance하는 데 사용되는 CLSID를 가져옵니다. (ITPluggableTerminalSuperclassRegistration.get_CLSID)
get_Comment

get_Comment 메서드는 주석을 가져옵니다.
get_Company get_Company 메서드는 이 플러그형 터미널을 발급한 회사의 이름을 가져옵니다. (ITPluggableTerminalClassInfo.get_Company)
get_Company get_Company 메서드는 이 플러그형 터미널을 발급한 회사의 이름을 가져옵니다. (ITPluggableTerminalClassRegistration.get_Company)
get_CompletionMessages get_CompletionMessages 완료 메시지를 가져옵니다. 이 메서드는 Visual Basic 및 스크립팅 언어로 작성된 애플리케이션과 같은 Automation 클라이언트 애플리케이션에 대해 제공됩니다.
get_ControllingTerminal get_ControllingTerminal 메서드는 현재 트랙을 제어하는 멀티트랙 터미널을 반환합니다.
get_Count get_Count 메서드는 컬렉션의 항목 수를 가져옵니다.
get_CountryCode get_CountryCode 메서드는 국가/지역 코드를 가져옵니다.
get_CountryID get_CountryID 메서드는 국가/지역의 식별자를 가져옵니다.
get_CurrentCallsQueued ITQueue::get_CurrentCallsQueued(tapi3.h) 메서드는 현재 대기 중인 들어오는 호출 수를 가져옵니다.
get_CurrentCallsQueued ITQueue::get_CurrentCallsQueued 메서드(tapi3cc.h)는 현재 대기 중인 수신 호출 수를 가져옵니다.
get_CurrentCountryCode get_CurrentCountryCode 메서드는 현재 국가/지역 코드를 가져옵니다.
get_CurrentForwardInfo get_CurrentForwardInfo 메서드는 현재 전달 정보 개체에 대한 포인터를 가져옵니다.

get_CurrentLongestWaitTime ITQueue::get_CurrentLongestWaitTime(tapi3.h) 메서드는 현재 측정 기간 동안 현재 호출이 대기한 가장 긴 시간을 가져옵니다.
get_CurrentLongestWaitTime ITQueue::get_CurrentLongestWaitTime 메서드(tapi3cc.h)는 현재 측정 기간 동안 현재 호출이 큐에서 대기한 가장 긴 시간을 가져옵니다.
get_DefaultDirectories get_DefaultDirectories 메서드는 구성된 모든 기본 디렉터리를 열거합니다. 이 메서드는 EnumerateDefaultDirectories와 유사하지만 Visual Basic 및 기타 스크립팅 언어에서 사용하도록 제공됩니다.
get_DefaultObjectTTL get_DefaultObjectTTL 메서드는 만든 개체에 대한 TTL(기본 TTL) 값을 초 단위로 가져옵니다. 동적 서버에만 적용됩니다.
get_Description get_Description 메서드는 회의에 대한 설명을 가져옵니다.
get_DestAddress get_DestAddress 메서드는 대상 주소를 가져옵니다.
get_DestinationCountryCode 호출 대상에 대한 국가/지역 코드를 검색합니다.
get_DeviceClasses get_DeviceClasses 메서드는 디바이스 클래스를 가져옵니다. 이 메서드는 Visual Basic 및 스크립팅 언어로 작성된 애플리케이션과 같은 Automation 클라이언트 애플리케이션에 대해 제공됩니다.
get_DialableAddress get_DialableAddress 메서드는 이 주소에 연결하는 데 사용할 수 있는 BSTR을 가져옵니다. BSTR은 다른 애플리케이션이 이 주소에 연결하는 데 사용하는 대상 주소 문자열(예: 전화 번호 또는 전자 메일 이름)에 해당합니다.
get_DialableAddrs get_DialableAddrs 메서드는 디렉터리에서 지정된 형식의 모든 전화 걸기 주소를 가져옵니다. 이 메서드는 EnumerateDialableAddrs와 동일한 함수를 수행하지만 Visual Basic과 같은 스크립팅 언어에서 사용됩니다.

get_DialableString get_DialableString 메서드는 전화 걸기 가능 번호를 포함하는 문자열을 가져옵니다. 일반적으로 이 숫자는 ITAddress::CreateCall에서 pDestAddress 인수로 사용됩니다.
get_Digit get_Digit 메서드는 감지된 숫자에 대한 부호 없는 char 포인터를 검색합니다.
get_DigitMode get_DigitMode 메서드는 줄 숫자 모드(예: LINEDIGITMODE_DTMF)의 표시기를 가져옵니다.
get_Digits get_Digits 메서드는 호출에 대해 수집된 숫자를 가져옵니다.
get_Direction get_Direction 메서드는 터미널에서 지원하는 방향을 가져옵니다. (ITPluggableTerminalClassInfo.get_Direction)
get_Direction get_Direction 메서드는 스트림의 터미널 방향을 가져옵니다.
get_Direction get_Direction 메서드는 터미널에 대한 미디어 스트림 방향의 TERMINAL_DIRECTION 설명자를 가져옵니다.
get_Direction get_Direction 메서드는 터미널에서 지원하는 방향을 가져옵니다. (ITPluggableTerminalClassRegistration.get_Direction)
get_DirectionsInUse get_DirectionsInUse 메서드는 멀티트랙 터미널에서 현재 관리되는 모든 트랙의 방향을 반환합니다.
get_DirectoryObjects get_DirectoryObjects 메서드는 특정 조건과 일치하는 지정된 디렉터리의 개체 컬렉션을 가져옵니다. 이 메서드는 EnumerateDirectoryObjects와 동일한 함수를 수행하지만 Visual Basic 및 기타 스크립팅 언어에서 사용됩니다.
get_DirectoryType

get_DirectoryType 메서드는 디렉터리 형식의 DIRECTORY_TYPE 표시기를 가져옵니다.
get_Display get_Display 메서드는 휴대폰의 디스플레이를 가져옵니다. TAPI에서 디스플레이는 단순히 NxM 문자 버퍼입니다.
get_DisplayableString get_DisplayableString 메서드는 표시 가능한 버전의 전화 걸기 가능 번호를 포함하는 문자열을 가져옵니다.
get_DisplayName get_DisplayName 메서드는 디렉터리의 표시 가능한 이름을 가져옵니다.
get_DoNotDisturb get_DoNotDisturb 메서드는 주소에서 방해 금지 기능의 현재 상태 가져옵니다. 방해 금지 기능은 모든 주소에서 사용할 수 없습니다.
get_Duration get_Duration 메서드는 TAPI Server에서 톤 이벤트를 생성하기 전에 톤이 있어야 하는 기간을 검색합니다.
get_DynamicTerminalClasses get_DynamicTerminalClasses 메서드는 현재 사용 가능한 동적 터미널의 컬렉션을 만듭니다.
get_EmptyAudioFormatForScripting get_EmptyAudioFormatForScripting 메서드는 모든 필드가 0으로 설정된 ITScriptableAudioFormat 인터페이스를 가져오는 데 사용됩니다.
get_Error get_Error 메서드는 터미널 이벤트와 연결된 오류의 HRESULT 캐스트를 반환합니다.
get_Error get_Error 메서드는 미디어 이벤트(있는 경우)와 관련된 오류를 가져옵니다.
get_Error get_Error 메서드는 이벤트에 대한 오류 코드를 가져옵니다.
get_Error

get_Error 메서드는 이벤트에 관련된 오류 코드의 HRESULT 캐스트를 반환합니다.
get_Error get_Error 메서드는 터미널 이벤트에 관련된 오류 코드의 HRESULT 캐스트를 가져옵니다.
get_Event ITACDGroupEvent::get_Event 메서드(tapi3.h)는 새 ACD 그룹이 추가되었음을 나타내는 이벤트의 설명자를 가져옵니다.
get_Event ITAgentEvent::get_Event 메서드(tapi3.h)는 발생한 이벤트의 AGENT_EVENT 설명자를 가져옵니다.
get_Event get_Event 메서드(tapi3.h)는 발생한 이벤트에 대한 설명을 가져옵니다.
get_Event get_Event 메서드(tapi3.h)는 발생한 이벤트의 AGENT_SESSION_EVENT 설명자를 가져옵니다.
get_Event ITQueueEvent::get_Event(tapi3.h) 메서드는 발생한 이벤트의 설명자를 가져옵니다.
get_Event get_Event 메서드는 이벤트의 ADDRESS_EVENT 설명자를 가져옵니다.
get_Event get_Event 메서드는 발생한 이벤트에 대한 CALLHUB_EVENT 열거형 설명에 대한 포인터를 반환합니다.
get_Event get_Event 메서드는 호출 미디어 이벤트 표시기를 가져옵니다.
get_Event get_Event 메서드는 애플리케이션이 이벤트가 발생한 호출을 소유하거나 모니터링하는지에 대한 CALL_NOTIFICATION_EVENT 설명을 반환합니다.
get_Event get_Event 메서드는 발생한 전화 이벤트 유형을 지정하는 PHONE_EVENT 값을 반환합니다.

get_Event get_Event 메서드는 QOS_EVENT 표시기를 가져옵니다.
get_Event get_Event 메서드는 비동기 이벤트 알림과 관련된 정보를 가져옵니다. 애플리케이션은 TAPIOBJECT_EVENT 사용하여 신호를 받고 있는 이벤트 유형을 확인합니다.
get_Event ITACDGroupEvent::get_Event 메서드(tapi3cc.h)는 새 ACD 그룹이 추가되었음을 나타내는 이벤트의 설명자를 가져옵니다.
get_Event ITAgentEvent::get_Event 메서드(tapi3cc.h)는 발생한 이벤트의 AGENT_EVENT 설명자를 가져옵니다.
get_Event ITAgentHandlerEvent::get_Event 메서드(tapi3cc.h)는 발생한 이벤트에 대한 설명을 가져옵니다. (ITAgentHandlerEvent.get_Event)
get_Event ITAgentSessionEvent::get_Event 메서드(tapi3cc.h)는 발생한 이벤트의 AGENT_SESSION_EVENT 설명자를 가져옵니다.
get_Event ITQueueEvent::get_Event 메서드(tapi3cc.h)는 발생한 이벤트의 설명자를 가져옵니다.
get_EventCode get_EventCode 메서드는 공급자별 이벤트 설명자에 대한 포인터를 반환합니다.
get_EventFilter get_EventFilter 메서드는 현재 주소 및 지정된 TAPI_EVENT 값에 대한 현재 필터 설정을 검색합니다.
get_EventFilter get_EventFilter 메서드는 이 호출에 적용되는 이벤트 필터 정보를 가져옵니다.
get_EventFilter get_EventFilter 메서드는 현재 이벤트 필터 마스크를 가져옵니다. 마스크는 TAPI_EVENT 열거형의 일련의 ORed 멤버입니다.

get_EventInterface get_EventInterface 메서드는 프라이빗 개체의 이벤트 처리기의 IDispatch 인터페이스에 대한 포인터를 반환합니다.
get_FileName get_FileName 메서드는 이 터미널에서 기록하는 데 사용된 파일의 이름을 검색합니다.
get_FinalDisposition ITQueue::get_FinalDisposition(tapi3.h) 메서드는 현재 측정 기간 동안 호출 가이드의 맨 아래에 도달하는 총 호출 수를 가져옵니다.
get_FinalDisposition ITQueue::get_FinalDisposition 메서드(tapi3cc.h)는 현재 측정 기간 동안 호출 가이드의 맨 아래에 도달하는 총 호출 수를 가져옵니다.
get_Format get_Format 메서드는 파일 터미널의 형식을 가져옵니다.
get_FormatTag get_FormatTag 메서드는 WAVEFORMATEX 구조체의 wFormatTag 멤버에 대한 값을 반환합니다.
get_ForwardTypeCaller get_ForwardTypeCaller 메서드는 지정된 전달 모드에 대한 호출자 유형을 가져옵니다.
get_ForwardTypeCallerAddressType get_ForwardTypeCallerAddressType 메서드는 지정된 전달 형식에 대한 호출자 주소 형식을 가져옵니다.
get_ForwardTypeDestination get_ForwardTypeDestination 메서드는 전달 모드의 대상을 가져옵니다.
get_ForwardTypeDestinationAddressType get_ForwardTypeDestinationAddressType 메서드는 지정된 전달 형식의 대상 주소 유형을 가져옵니다.
get_Frequency get_Frequency 메서드는 생성할 톤 구성 요소의 빈도를 검색합니다.

get_Frequency get_Frequency 메서드는 TAPI Server에서 톤 이벤트를 생성하는 톤 빈도를 검색합니다.
get_GatherTermination get_GatherTermination 메서드는 TAPI 서버가 호출에서 숫자 수집을 종료한 이유를 가져옵니다.
get_GenerationTermination get_GenerationTermination 메서드는 생성된 숫자 계열의 끝을 나타내는 숫자 또는 숫자를 가져옵니다.
get_Group ITACDGroupEvent::get_Group 메서드(tapi3.h)는 이벤트가 발생한 그룹에 대한 ITACDGroup 인터페이스 포인터를 가져옵니다.
get_Group ITACDGroupEvent::get_Group 메서드(tapi3cc.h)는 이벤트가 발생한 그룹에 대한 ITACDGroup 인터페이스 포인터를 가져옵니다.
get_HookSwitchDevice get_HookSwitchDevice 메서드는 상태를 변경한 후크위치 디바이스를 지정하는 PHONE_HOOK_SWITCH_DEVICE 값을 반환합니다. 이 정보는 ITPhoneEvent::get_Event 메서드가 PE_HOOKSWITCH 반환하는 경우에만 사용할 수 있습니다.
get_HookSwitchState get_HookSwitchState 메서드는 휴대폰의 특정 후크위치 디바이스에 대한 현재 후크위치 상태를 검색합니다.
get_HookSwitchState get_HookSwitchState 메서드는 hookswitch가 전환된 상태를 지정하는 PHONE_HOOK_SWITCH_STATE 값을 반환합니다. 이 정보는 ITPhoneEvent::get_Event 메서드가 PE_HOOKSWITCH 반환하는 경우에만 사용할 수 있습니다.
get_ID ITAgent::get_ID 메서드(tapi3.h)는 에이전트의 ID를 가져옵니다.
get_ID ITAgent::get_ID 메서드(tapi3cc.h)는 에이전트의 ID를 가져옵니다.

get_InterfaceID get_InterfaceID 메서드는 이 scope 게시한 서버가 있는 인터페이스를 식별하는 이 scope 인터페이스 식별자를 가져옵니다. 일반적으로 인터페이스의 네트워크 주소입니다.
get_InternationalDialingRule get_InternationalDialingRule 메서드는 이 통화 카드 대한 국제 전화 걸기 규칙을 가져옵니다.
get_IPPhonePrimary get_IPPhonePrimary 메서드는 사용자의 기본 IP 전화인 컴퓨터의 이름을 가져옵니다.
get_IsDynamic get_IsDynamic 메서드는 서버의 개체를 새로 고쳐야 하는지 여부를 나타내는 표시기를 가져옵니다.
get_IsEncrypted get_IsEncrypted 메서드는 회의 암호화 여부를 가져옵니다.
get_Item 인덱스가 지정된 get_Item 메서드는 컬렉션의 항목을 반환합니다.
get_LampMode get_LampMode 메서드는 지정된 램프의 현재 램프 모드를 가져옵니다.
get_LeaseStartTime get_LeaseStartTime 메서드는 임대의 시작 시간을 가져옵니다.
get_LeaseStopTime get_LeaseStopTime 메서드는 임대 중지 시간을 가져옵니다.
get_LocalAccessCode get_LocalAccessCode 메서드는 로컬 액세스 코드를 가져옵니다.
get_LocationName get_LocationName 메서드는 위치 이름을 가져옵니다.
get_Locations get_Locations 메서드는 현재 사용 가능한 주소 위치의 컬렉션을 만듭니다. 이 메서드는 Visual

Basic으로 작성된 애플리케이션과 같은 Automation 클라이언트 애플리케이션에 대해 제공됩니다. C 및 C++ 애플리케이션은 EnumerateLocations 메서드를 사용해야 합니다.
get_LongDistanceAccessCode get_LongDistanceAccessCode 메서드는 장거리에 액세스하는 코드를 가져옵니다.
get_LongDistanceDialingRule get_LongDistanceDialingRule 메서드는 이 호출 카드 대한 장거리 전화 걸기 규칙을 가져옵니다.
get_LongestEverWaitTime ITQueue::get_LongestEverWaitTime(tapi3.h) 메서드는 현재 측정 기간 동안 큐에서 호출이 대기한 가장 긴 시간(초)을 가져옵니다.
get_LongestEverWaitTime ITQueue::get_LongestEverWaitTime 메서드(tapi3cc.h)는 현재 측정 기간 동안 큐에서 호출이 대기한 가장 긴 시간(초)을 가져옵니다.
get_LongestTimeToAnswer ITAgentSession::get_LongestTimeToAnswer(tapi3.h) 메서드는 호출이 응답되기를 기다리는 가장 긴 시간(초)을 가져옵니다.
get_LongestTimeToAnswer ITAgentSession::get_LongestTimeToAnswer 메서드(tapi3cc.h)는 호출이 응답 대기 중인 가장 긴 시간(초)을 가져옵니다.
get_lParam1 get_lParam1 메서드는 지정된 주소 디바이스와 관련된 세 개의 버퍼 중 첫 번째 버퍼를 검색합니다. 이러한 버퍼의 정확한 콘텐츠와 의미는 공급자에 의해 정의됩니다.
get_lParam1 get_lParam1 메서드는 지정된 전화 장치와 관련된 세 개의 버퍼 중 첫 번째 버퍼를 검색합니다. 이러한 버퍼의 정확한 콘텐츠와 의미는 공급자에 의해 정의됩니다.
get_lParam2 get_lParam2 메서드는 지정된 주소 디바이스와 관련된 세 개의 버퍼 중 두 번째 버퍼를 검색합니다. 이러한 버퍼의 정확한 콘텐츠와 의미는 공급자에 의해 정의됩니다.
get_lParam2

get_lParam2 메서드는 지정된 전화 장치와 관련된 세 개의 버퍼 중 두 번째 버퍼를 검색합니다. 이러한 버퍼의 정확한 콘텐츠와 의미는 공급자에 의해 정의됩니다.
get_lParam3 get_lParam3 메서드는 지정된 주소 디바이스와 관련된 세 개의 버퍼 중 세 번째 버퍼를 검색합니다. 이러한 버퍼의 정확한 콘텐츠와 의미는 공급자에 의해 정의됩니다.
get_lParam3 get_lParam3 메서드는 지정된 전화 장치와 관련된 세 개의 버퍼 중 세 번째 버퍼를 검색합니다. 이러한 버퍼의 정확한 콘텐츠와 의미는 공급자에 의해 정의됩니다.
get_MeasurementPeriod get_MeasurementPeriod 메서드(tapi3.h)는 스위치 및/또는 구현이 정보를 저장하고 계산하는 측정 기간을 가져옵니다.
get_MeasurementPeriod ITQueue::get_MeasurementPeriod(tapi3.h) 메서드는 스위치 및/또는 구현이 정보를 저장하고 계산하는 측정 기간을 가져옵니다.
get_MeasurementPeriod ITAgent::get_MeasurementPeriod 메서드(tapi3cc.h)는 스위치 및/또는 구현이 정보를 저장하고 계산하는 측정 기간을 가져옵니다.
get_MeasurementPeriod ITQueue::get_MeasurementPeriod 메서드(tapi3cc.h)는 스위치 및/또는 구현이 정보를 저장하고 계산하는 측정 기간을 가져옵니다.
get_MediaFormat ITAMMediaFormat::get_MediaFormat(tapi3.h) 메서드는 미디어 형식을 가져옵니다.
get_MediaFormat ITAMMediaFormat::get_MediaFormat 메서드(tapi3ds.h)는 미디어 형식을 가져옵니다.
get_MediaState get_MediaState 메서드는 파일 터미널에서 미디어의 현재 상태를 가져옵니다.
get_MediaType get_MediaType 메서드는 미디어 형식 표시기를 가져옵니다.

get_MediaType get_MediaType 메서드는 스트림의 미디어 형식을 가져옵니다.
get_MediaType get_MediaType 메서드는 이 터미널에서 지원하는 미디어를 결정합니다.
get_MediaTypes get_MediaTypes 메서드는 현재 주소에서 지원되는 미디어 형식 또는 형식을 가져옵니다.
get_MediaTypes get_MediaTypes 메서드는 터미널에서 지원하는 미디어 형식을 가져옵니다. (ITPluggableTerminalClassInfo.get_MediaTypes)
get_MediaTypes get_MediaTypes 메서드는 터미널에서 지원하는 미디어 형식을 가져옵니다. (ITPluggableTerminalClassRegistration.get_MediaTypes)
get_MediaTypesInUse get_MediaTypesInUse 메서드는 멀티트랙 터미널에서 현재 관리되는 모든 트랙의 미디어 형식(비트 ORed)을 반환합니다.
get_MessageWaiting get_MessageWaiting 메서드는 주소에 대기 중인 메시지가 있는지 확인합니다.
get_Name ITACDGroup::get_Name 메서드(tapi3.h)는 ACD 그룹 이름을 가져옵니다. 이 문자열은 그룹의 표시 가능한 이름일 수 있습니다.
get_Name get_Name 메서드(tapi3.h)는 에이전트 처리기 이름을 가져옵니다.
get_Name ITQueue::get_Name(tapi3.h) 메서드는 큐 이름을 가져옵니다.
get_Name get_Name 메서드는 터미널의 이름을 가져옵니다. (ITPluggableTerminalClassInfo.get_Name)

get_Name get_Name 메서드는 터미널의 이름을 가져옵니다. (ITPluggableTerminalSuperclassInfo.get_Name)
get_Name get_Name 메서드는 스트림의 이름을 나타내는 BSTR을 가져옵니다. 이 이름은 정보 또는 표시 목적으로 사용됩니다.
get_Name get_Name 메서드는 터미널의 설명이 포함된 이름을 가져옵니다. 이름은 사용자에게 사용할 수 있어야 합니다.
get_Name get_Name 메서드는 터미널의 이름을 가져옵니다. (ITPluggableTerminalClassRegistration.get_Name)
get_Name get_Name 메서드는 터미널 슈퍼클래스에 대한 이름을 가져옵니다.
get_Name ITACDGroup::get_Name 메서드(tapi3cc.h)는 ACD 그룹 이름을 가져옵니다. 이 문자열은 그룹의 표시 가능한 이름일 수 있습니다.
get_Name ITAgentHandler::get_Name 메서드(tapi3cc.h)는 에이전트 처리기 이름을 가져옵니다.
get_Name ITQueue::get_Name 메서드(tapi3cc.h)는 큐 이름을 가져옵니다.
get_Name get_Name 메서드는 디렉터리 개체의 이름을 가져옵니다.
get_NumberGathered get_NumberGathered 메서드는 수집된 전화 번호를 지정하는 BSTR 값을 반환합니다. 이 정보는 ITPhoneEvent::get_Event 메서드가 PE_NUMBERGATHERED 반환하는 경우에만 사용할 수 있습니다.
get_NumberOfACDCalls get_NumberOfACDCalls 메서드(tapi3.h)는 모든 세션에서 이 에이전트가 처리하는 ACD 호출 수를 가져옵니다.

get_NumberOfACDCalls ITAgent::get_NumberOfACDCalls 메서드(tapi3cc.h)는 모든 세션에서 이 에이전트가 처리하는 ACD 호출 수를 가져옵니다.
get_NumberOfCalls ITAgentSession::get_NumberOfCalls(tapi3.h) 메서드는 이 세션 중에 이 에이전트가 처리한 ACD 호출 수를 가져옵니다.
get_NumberOfCalls ITAgentSession::get_NumberOfCalls 메서드(tapi3cc.h)는 이 세션 중에 이 에이전트가 처리한 ACD 호출 수를 가져옵니다.
get_NumberOfDigits get_NumberOfDigits 메서드는 기존 카드 번호의 숫자 수를 가져옵니다.
get_NumberOfIncomingCalls get_NumberOfIncomingCalls 메서드(tapi3.h)는 이 에이전트가 처리하는 들어오는 비 ACD 호출 수를 가져옵니다.
get_NumberOfIncomingCalls ITAgent::get_NumberOfIncomingCalls 메서드(tapi3cc.h)는 이 에이전트가 처리하는 들어오는 비 ACD 호출 수를 가져옵니다.
get_NumberOfOutgoingCalls get_NumberOfOutgoingCalls 메서드(tapi3.h)는 이 에이전트에서 처리하는 나가는 비 ACD 호출 수를 가져옵니다.
get_NumberOfOutgoingCalls ITAgent::get_NumberOfOutgoingCalls 메서드(tapi3cc.h)는 이 에이전트에서 처리하는 나가는 비 ACD 호출 수를 가져옵니다.
get_NumCalls get_NumCalls 메서드는 현재 CallHub에 있는 호출 수를 가져옵니다.
get_NumRingsNoAnswer get_NumRingsNoAnswer 메서드는 응답 조건이 없는 것으로 가정된 링 수를 검색합니다.
get_ObjectType

get_ObjectType 메서드는 개체의 DIRECTORY_OBJECT_TYPE 설명자를 가져옵니다.
get_Options get_Options 메서드는 이 주소 및 카드 대한 번역 옵션을 가져옵니다.
get_Options get_Options 메서드는 현재 위치가 펄스 또는 톤 전화 걸기를 지원하는지 여부를 나타내는 표시기를 가져옵니다.
get_Originator get_Originator 메서드는 보낸 사람의 사용자 이름을 가져옵니다.
get_OverallCallRate get_OverallCallRate 메서드(tapi3.h)는 모든 세션에서 에이전트의 호출 속도를 가져옵니다.
get_OverallCallRate ITAgent::get_OverallCallRate 메서드(tapi3cc.h)는 모든 세션에서 에이전트의 호출 속도를 가져옵니다.
get_PermanentCardID get_PermanentCardID 메서드는 카드 식별하는 영구 식별자를 가져옵니다.
get_PermanentLocationID get_PermanentLocationID 메서드는 영구 위치 식별자를 가져옵니다.
get_Phone get_Phone 메서드는 휴대폰 디바이스 이벤트에 대한 ITPhone 인터페이스 포인터를 검색합니다.
get_Phone get_Phone 메서드는 이 이벤트를 발생시킨 휴대폰 개체의 ITPhone 인터페이스에 대한 포인터를 반환합니다.
get_Phone get_Phone 메서드는 이 TAPI 개체 이벤트를 발생시킨 휴대폰 개체의 ITPhone 인터페이스에 대한 포인터를 반환합니다.
get_PhoneCapsBuffer

get_PhoneCapsBuffer 메서드는 전달된 PHONECAPS_BUFFER 열거형을 기반으로 휴대폰에 대한 버퍼 기능/정보를 가져옵니다.
get_PhoneCapsLong get_PhoneCapsLong 메서드는 전달된 PHONECAPS_LONG 열거형에 따라 휴대폰의 DWORD 기능을 가져옵니다. 이 메서드를 실행하기 전에 애플리케이션에서 ITPhone::Open을 호출할 필요가 없습니다.
get_PhoneCapsString get_PhoneCapsString 메서드는 전달된 PHONECAPS_STRING 열거형에 따라 휴대폰에 대한 문자열 기능/정보를 가져옵니다. 이 메서드를 실행하기 전에 애플리케이션에서 ITPhone::Open을 호출할 필요가 없습니다.
get_PhoneHandlingEnabled get_PhoneHandlingEnabled 메서드는 PhoneHandlingEnabled 속성의 현재 값을 검색합니다.
get_Phones get_Phones 메서드는 이 주소와 함께 사용할 수 있는 휴대폰 디바이스에 해당하는 전화 개체의 ITCollection에 대한 VARIANT 포인터를 반환합니다.
get_Phones get_Phones 메서드는 휴대폰 디바이스에 해당하는 휴대폰 개체를 열거합니다. 주소와 함께 사용할 수 있는 휴대폰이 없는 경우 이 메서드는 빈 컬렉션을 생성하고 S_OK 반환합니다.
get_PlayList get_PlayList 메서드는 재생할 파일 목록을 가져옵니다.
get_PluggableSuperclasses get_PluggableSuperclasses 메서드는 ITPluggableTerminalSuperclassInfo 슈퍼클래스 정보 인터페이스 포인터의 컬렉션을 반환합니다.
get_PluggableTerminalClasses get_PluggableTerminalClasses 메서드는 ITPluggableTerminalClassInfo 터미널 클래스 정보 인터페이스 포인터의 컬렉션을 반환합니다.
get_Port get_Port 메서드는 지정된 ILS 디렉터리의 서버에 연결하는 데 사용되는 포트 번호를 가져옵니다.
get_PreferredAddresses

get_PreferredAddresses 메서드는 휴대폰에서 사용하는 데 선호되는 주소 컬렉션을 반환합니다. 이 메서드를 실행하기 전에 애플리케이션에서 ITPhone::Open을 호출할 필요가 없습니다.
get_PreferredCardID get_PreferredCardID 메서드는 현재 위치에서 전화를 걸 때 기본 호출 카드 식별자를 가져옵니다.
get_PreferredPhones get_PreferredPhones 메서드는 이 주소에 사용하기 위해 선호되는 휴대폰 디바이스에 해당하는 휴대폰 개체의 컬렉션을 반환합니다.
get_PrivateTAPIObjects ITTAPI 인터페이스의 get_PrivateTAPIObjects 메서드는 구현되지 않으며 E_NOTIMPL 반환합니다.
get_Privilege get_Privilege 메서드는 현재 호출에 대한 애플리케이션의 호출 권한(예: CP_MONITOR)을 가져옵니다.
get_Privilege get_Privilege 메서드는 열려 있는 휴대폰의 권한을 검색합니다.
get_Protocol get_Protocol 메서드는 프로토콜 식별을 가져옵니다.
get_Queue ITQueueEvent::get_Queue(tapi3.h) 메서드는 이벤트가 발생한 큐에 대한 포인터를 가져옵니다.
get_Queue ITQueueEvent::get_Queue 메서드(tapi3cc.h)는 이벤트가 발생한 큐에 대한 포인터를 가져옵니다.
get_Queues Automation 클라이언트 애플리케이션에 대한 ITACDGroup::get_Queues 메서드(tapi3.h)는 현재 ACD 그룹과 연결된 큐 컬렉션을 만듭니다.
get_Queues ITACDGroup::get_Queues 메서드(tapi3cc.h)는 현재 ACD 그룹과 연결된 큐 컬렉션을 만듭니다.
get_RegistrationInstance get_RegistrationInstance 메서드는 등록 instance 가져옵니다.

get_RequestID get_RequestID 메서드는 임대에 대한 요청 ID를 가져옵니다.
get_RequestMode get_RequestMode 메서드는 요청 모드를 가져옵니다.
get_Ringer get_Ringer 메서드는 휴대폰이 현재 이 인터페이스에서 StartRinger 메서드에 의해 시작된 들어오는 링을 수행하고 있는지 여부를 나타내는 부울 값을 반환합니다.
get_RingMode get_RingMode 메서드는 휴대폰의 현재 링 모드를 검색합니다.
get_RingMode get_RingMode 메서드는 휴대폰이 전환된 링 모드를 지정하는 긴 값을 반환합니다. 이 정보는 ITPhoneEvent::get_Event 메서드가 PE_RINGMODE 반환하는 경우에만 사용할 수 있습니다.
get_RingVolume get_RingVolume 메서드는 휴대폰의 현재 링 볼륨을 검색합니다.
get_SameAreaDialingRule get_SameAreaDialingRule 메서드는 동일한 지역 코드 내의 호출에 대한 전화 걸기 규칙을 가져옵니다.
get_SamplesPerSec get_SamplesPerSec 메서드는 WAVEFORMATEX 구조체의 nSamplesPerSec 멤버에 대한 값을 반환합니다.
get_ScopeDescription get_ScopeDescription 메서드는 이 scope 연결된 텍스트 설명을 가져옵니다. 설명은 scope 목적이나 의미를 명확히 하는 데만 사용되며 후속 호출에는 필요하지 않습니다.
get_ScopeID get_ScopeID 메서드는 멀티캐스트 주소의 scope 대한 식별자를 가져옵니다.
get_Scopes get_Scopes 메서드는 사용 가능한 IMcast 범위 컬렉션을 만듭니다. 이 메서드는 EnumerateScopes와 비슷하지만 Visual Basic과 같은 스크립팅 언어에서 사용됩니다.

get_SecurityDescriptor get_SecurityDescriptor 메서드는 현재 보안 권한을 설명하는 디렉터리 서비스 보안 설명자 개체에 대한 IDispatch 포인터를 가져옵니다.
get_SelectedCalls get_SelectedCalls 메서드는 현재 이 휴대폰에서 선택된 호출을 나타내는 컬렉션 개체에 대한 포인터가 포함된 VARIANT를 검색합니다. 자세한 내용은 ITAutomatedPhoneControl::SelectCall을 참조하세요.
get_ServerAddress get_ServerAddress 메서드는 이 임대를 부여하는 멀티캐스트 서버의 주소를 나타내는 문자열을 가져옵니다.
get_ServerID get_ServerID 메서드는 이 scope 연결된 서버 ID를 가져옵니다.
get_ServiceProviderName get_ServiceProviderName 메서드는 이 address:_for example, Unimodem 서비스 공급자의 경우 Unimdm.tsp, H323 서비스 공급자의 경우 H323.tsp를 지원하는 TSP(전화 통신 서비스 공급자)의 이름을 가져옵니다.
get_Session get_Session 메서드(tapi3.h)는 이벤트가 발생한 ITAgentSession에 대한 포인터를 가져옵니다.
get_Session ITAgentSessionEvent::get_Session 메서드(tapi3cc.h) 메서드는 이벤트가 발생한 ITAgentSession에 대한 포인터를 가져옵니다.
get_SessionDuration ITAgentSession::get_SessionDuration(tapi3.h) 메서드는 에이전트 세션의 기간을 초 단위로 가져옵니다.
get_SessionDuration ITAgentSession::get_SessionDuration 메서드(tapi3cc.h)는 에이전트 세션의 기간을 초 단위로 가져옵니다.
get_SessionStartTime ITAgentSession::get_SessionStartTime(tapi3.h) 메서드는 세션이 만들어진 시간을 가져옵니다.

get_SessionStartTime ITAgentSession::get_SessionStartTime(tapi3cc.h) 메서드는 세션이 만들어진 시간을 가져옵니다.
get_StartTime get_StartTime 메서드는 회의의 시작 시간을 가져옵니다.
get_State get_State 메서드(tapi3.h)는 에이전트 세션의 상태를 가져옵니다.
get_State ITAgentSession::get_State(tapi3.h) 메서드는 이 세션의 현재 상태를 가져옵니다.
get_State get_State 메서드는 pAddressState에서 주소의 현재 상태를 가져옵니다.
get_State get_State 메서드는 CallHub의 현재 상태를 가져옵니다.
get_State get_State 메서드는 새 호출 상태에 대한 정보를 가져옵니다.
get_State get_State 메서드는 새 파일 터미널 상태에 대한 정보를 가져옵니다.
get_State get_State 메서드는 터미널의 현재 상태를 가져옵니다.
get_State ITAgent::get_State 메서드(tapi3cc.h)는 에이전트 세션의 상태를 가져옵니다.
get_State ITAgentSession::get_State 메서드(tapi3cc.h)는 이 세션의 현재 상태를 가져옵니다.
get_StaticTerminals get_StaticTerminals 메서드는 현재 사용 가능한 정적 터미널의 컬렉션을 만듭니다.
get_StopTime

get_StopTime 메서드는 회의 중지 시간을 가져옵니다. 종료 시간이 0이면 세션이 바인딩되지 않습니다.
get_Stream get_Stream 메서드는 호출 미디어 이벤트와 연결된 ITStream 인터페이스에 대한 포인터를 가져옵니다.
get_Stream get_Stream 메서드는 현재 하위 스트림에 대한 ITStream 인터페이스에 대한 포인터를 검색합니다.
get_Streams get_Streams 메서드는 현재 호출에서 사용할 수 있는 미디어 스트림의 컬렉션을 만듭니다. Visual Basic으로 작성된 애플리케이션과 같은 Automation 클라이언트 애플리케이션에 대해 제공됩니다. C 및 C++ 애플리케이션은 EnumerateStreams 메서드를 사용해야 합니다.
get_SubStreams get_SubStreams 메서드는 현재 사용 가능한 하위 스트림의 컬렉션을 만듭니다. Visual Basic으로 작성된 애플리케이션과 같은 Automation 클라이언트 애플리케이션에 대해 제공됩니다. C 및 C++ 애플리케이션은 EnumerateSubStreams 메서드를 사용해야 합니다.
get_TAPIObject get_TAPIObject 메서드는 이 주소를 소유하는 TAPI 개체에 대한 포인터를 가져옵니다.
get_TAPIObject get_TAPIObject 메서드는 이벤트가 발생한 TAPI 개체에 대한 포인터를 가져옵니다.
get_Terminal get_Terminal 메서드는 이벤트와 연결된 ITTerminal 인터페이스에 대한 포인터를 가져옵니다.
get_Terminal get_Terminal 메서드는 이벤트가 발생한 터미널의 ITTerminal 인터페이스에 대한 포인터를 반환합니다.
get_Terminal get_Terminal 메서드는 이벤트와 연결된 터미널을 가져옵니다.
get_Terminal get_Terminal 메서드는 이 이벤트를 생성한 파일 터미널을 반환합니다.

get_Terminal get_Terminal 메서드는 이벤트가 발생한 톤 터미널에 대한 ITTerminal 포인터를 반환합니다.
get_Terminal get_Terminal 메서드는 이벤트에 관련된 터미널 개체에 대한 ITTerminal 인터페이스 포인터를 가져옵니다.
get_TerminalClass get_TerminalClass 메서드는 터미널의 터미널 클래스를 가져옵니다. (ITPluggableTerminalClassInfo.get_TerminalClass)
get_TerminalClass get_TerminalClass 메서드는 터미널의 터미널 클래스를 가져옵니다.
get_TerminalClass get_TerminalClass 메서드는 터미널의 터미널 클래스를 가져옵니다. (ITPluggableTerminalClassRegistration.get_TerminalClass)
get_TerminalClasses get_TerminalClasses 메서드는 이 플러그형 터미널 슈퍼클래스에 대한 터미널 클래스를 가져옵니다.
get_Terminals get_Terminals 메서드는 휴대폰과 연결된 터미널 컬렉션을 검색합니다. 이 메서드를 실행하기 전에 애플리케이션에서 ITPhone::Open을 호출할 필요가 없습니다.
get_Terminals get_Terminals 메서드는 현재 스트림과 연결된 터미널 컬렉션을 만듭니다. Visual Basic으로 작성된 애플리케이션과 같은 Automation 클라이언트 애플리케이션에 대해 제공됩니다. C 및 C++ 애플리케이션은 EnumerateTerminals 메서드를 사용해야 합니다.
get_Terminals get_Terminals 메서드는 현재 하위 스트림과 연결된 터미널의 컬렉션을 만듭니다. Visual Basic으로 작성된 애플리케이션과 같은 Automation 클라이언트 애플리케이션에 대해 제공됩니다. C 및 C++ 애플리케이션은 EnumerateTerminals 메서드를 사용해야 합니다.
get_TerminalType get_TerminalType 메서드는 터미널의 TERMINAL_TYPE 가져옵니다.

get_TickCount get_TickCount 메서드는 숫자 수집이 완료된 "틱 수"(Windows가 시작된 이후의 시간(밀리초))를 가져옵니다.
get_TickCount get_TickCount 메서드는 숫자 생성이 완료된 "틱 수"(Windows가 시작된 이후의 시간(밀리초))를 가져옵니다.
get_TickCount get_TickCount 메서드는 숫자 수집이 완료된 "틱 수"(Windows가 시작된 이후의 시간(밀리초))를 가져옵니다.
get_TickCount get_TickCount 메서드는 톤이 검색된 "틱 수"(Windows가 시작된 이후의 시간(밀리초))를 가져옵니다.
get_TollPrefixList get_TollPrefixList 메서드는 통행료 접두사 목록을 가져옵니다.
get_Tone get_Tone 메서드는 휴대폰이 현재 재생 중인 톤 유형을 나타내는 PHONE_TONE 열거형 값을 반환합니다.
get_TotalACDCallTime get_TotalACDCallTime 메서드(tapi3.h)는 이 에이전트의 ACD 호출에 소요된 시간(초)을 가져옵니다(모든 세션에서).
get_TotalACDCallTime ITAgent::get_TotalACDCallTime 메서드(tapi3cc.h)는 모든 세션에서 이 에이전트의 ACD 호출에 소요된 시간(초)을 가져옵니다.
get_TotalACDTalkTime get_TotalACDTalkTime 메서드(tapi3.h)는 이 에이전트가 ACD 호출에서 통신하는 데 소요된 시간(초)을 가져옵니다(모든 세션에서).
get_TotalACDTalkTime ITAgent::get_TotalACDTalkTime 메서드(tapi3cc.h)는 모든 세션에서 이 에이전트의 ACD 호출에서 통신하는 데 소요된 시간(초)을 가져옵니다.

get_TotalCallsAbandoned ITQueue::get_TotalCallsAbandoned(tapi3.h) 메서드는 현재 측정 기간 동안 중단된 호출 수를 가져옵니다.
get_TotalCallsAbandoned ITQueue::get_TotalCallsAbandoned 메서드(tapi3cc.h)는 현재 측정 기간 동안 중단된 호출 수를 가져옵니다.
get_TotalCallsFlowedIn ITQueue::get_TotalCallsFlowedIn(tapi3.h) 메서드는 현재 측정 기간 동안 이 큐로 전달된 총 호출 수를 가져옵니다.
get_TotalCallsFlowedIn ITQueue::get_TotalCallsFlowedIn 메서드(tapi3cc.h)는 현재 측정 기간 동안 이 큐로 전달된 총 호출 수를 가져옵니다.
get_TotalCallsFlowedOut ITQueue::get_TotalCallsFlowedOut(tapi3.h) 메서드는 현재 측정 기간 동안 이 큐에서 발생한 총 호출 수를 가져옵니다.
get_TotalCallsFlowedOut ITQueue::get_TotalCallsFlowedOut 메서드(tapi3cc.h)는 현재 측정 기간 동안 이 큐에서 발생한 총 호출 수를 가져옵니다.
get_TotalCallsQueued ITQueue::get_TotalCallsQueued(tapi3.h) 메서드는 현재 측정 기간 동안 이 큐에 대해 들어오는 총 호출 수를 가져옵니다.
get_TotalCallsQueued ITQueue::get_TotalCallsQueued 메서드(tapi3cc.h)는 현재 측정 기간 동안 이 큐에 대해 들어오는 총 호출 수를 가져옵니다.
get_TotalCallTime ITAgentSession::get_TotalCallTime(tapi3.h) 메서드는 이 에이전트 세션 동안 ACD 호출에 소요된 시간(초)을 가져옵니다(이 에이전트에 의해).
get_TotalCallTime ITAgentSession::get_TotalCallTime 메서드(tapi3cc.h)는 이 에이전트 세션 동안 ACD 호출에 소요된 시간(초)을 가져옵니다(이 에이전트에 의해).

get_TotalTalkTime ITAgentSession::get_TotalTalkTime(tapi3.h) 메서드는 이 세션 중에 ACD 호출에서 통신하는 이 에이전트가 소요한 시간(초)을 가져옵니다.
get_TotalTalkTime ITAgentSession::get_TotalTalkTime 메서드(tapi3cc.h)는 이 세션 동안 ACD 호출에서 통신하는 이 에이전트가 보낸 시간(초)을 가져옵니다.
get_TotalWrapUpTime get_TotalWrapUpTime 메서드(tapi3.h)는 이 에이전트가 ACD 호출 래핑(통화 후 작업)에 소요한 시간(초)을 가져옵니다(모든 세션).
get_TotalWrapUpTime ITAgentSession::get_TotalWrapUpTime(tapi3.h) 메서드는 이 에이전트 세션 동안 ACD 호출 래핑(통화 후 작업)에 소요된 시간(초)을 가져옵니다.
get_TotalWrapUpTime ITAgent::get_TotalWrapUpTime 메서드(tapi3cc.h)는 모든 세션에서 이 에이전트가 ACD 호출 래핑에 소요한 시간(초)을 가져옵니다.
get_TotalWrapUpTime ITAgentSession::get_TotalWrapUpTime 메서드(tapi3cc.h)는 이 에이전트 세션 동안 ACD 호출 래핑(통화 후 작업)에 소요된 시간(초)을 가져옵니다.
get_Track get_Track 메서드는 이 이벤트를 생성한 트랙 터미널을 반환합니다.
get_TrackTerminals get_TrackTerminals 메서드는 이 메서드가 호출된 멀티트랙 터미널에 포함된 터미널이 포함된 컬렉션을 만들고 반환합니다.
get_TranslationResults get_TranslationResults 메서드는 번역 작업의 결과를 가져옵니다.
get_TTL get_TTL 메서드는 이 임대와 연결된 TTL(Time to Live) 값을 가져옵니다.
get_TTL

get_TTL 메서드는 멀티캐스트 scope 대한 TL(Time to Live) 값을 가져옵니다.
get_Url get_Url 메서드는 URL을 가져옵니다.
get_UsableAddresses get_UsableAddresses 메서드(tapi3.h)는 이 에이전트 처리기에서 ACD 호출을 수신하는 데 사용할 수 있는 주소 컬렉션을 만듭니다.
get_UsableAddresses ITAgentHandler::get_UsableAddresses 메서드(tapi3cc.h)는 이 에이전트 처리기에서 ACD 호출을 수신하는 데 사용할 수 있는 주소 컬렉션을 만듭니다.
get_User get_User 메서드(tapi3.h)는 운영 체제 사용자 로그인 또는 전자 메일 이름과 동일한 에이전트 사용자 이름을 가져옵니다.
get_User ITAgent::get_User 메서드(tapi3cc.h)는 운영 체제 사용자 로그인 또는 전자 메일 이름과 동일한 에이전트 사용자 이름을 가져옵니다.
get_Version get_Version 메서드는 터미널 버전을 가져옵니다. (ITPluggableTerminalClassInfo.get_Version)
get_Version get_Version 메서드는 터미널 버전을 가져옵니다. (ITPluggableTerminalClassRegistration.get_Version)
get_Volume get_Volume 메서드는 볼륨을 가져옵니다.
get_Volume get_Volume 메서드는 톤을 생성할 볼륨 수준을 검색합니다.
get_WaveId get_WaveId 메서드는 이 터미널을 구현하는 데 사용되는 오디오 디바이스의 웨이브 ID를 반환합니다.
GetAllocateBuffers

ITAllocatorProperties::GetAllocateBuffers(tapi3.h) 메서드는 현재 할당자 버퍼를 검색할 수 있는지 여부를 결정합니다.
GetAllocateBuffers ITAllocatorProperties::GetAllocateBuffers 메서드(tapi3ds.h)는 현재 할당자 버퍼를 검색할 수 있는지 여부를 결정합니다.
GetAllocatorProperties ITAllocatorProperties::GetAllocatorProperties(tapi3.h) 메서드는 연결 후 할당자 속성에 대한 값을 가져오고 협상된 값을 제공합니다.
GetAllocatorProperties GetAllocatorProperties 메서드(tapi3ds.h)는 연결 후 할당자 속성에 대한 현재 값을 가져오고 협상된 값을 제공합니다.
GetBufferSize ITAllocatorProperties::GetBufferSize(tapi3.h) 메서드는 할당자 버퍼의 크기를 가져옵니다.
GetBufferSize ITAllocatorProperties::GetBufferSize 메서드(tapi3ds.h)는 할당자 버퍼의 크기를 가져옵니다.
GetCallInfoBuffer GetCallInfoBuffer 메서드는 사용자-사용자 정보와 같이 버퍼가 필요한 호출 정보 항목을 가져옵니다. Visual Basic으로 작성된 애플리케이션과 같은 Automation 클라이언트 애플리케이션은 ITCallInfo::get_CallInfoBuffer 메서드를 사용해야 합니다.
GetCallMediaTypes GetCallMediaTypes 메서드는 MSP에서 지원하는 미디어 형식을 가져옵니다. 파생 클래스는 지원되는 미디어 형식을 나타내기 위해 이 메서드를 재정의해야 합니다. 관련 TAPIMEDIATYPE 상수는 함께 ORed하고 반환되어야 합니다.
GetDefaultStaticTerminal GetDefaultStaticTerminal 메서드는 지정된 미디어 형식에 대한 기본 정적 터미널을 가져옵니다.
GetDevConfig GetDevConfig 메서드는 불투명 데이터 구조를 반환합니다.
GetDynamicTerminalClasses GetDynamicTerminalClasses 메서드는 미디어 형식 집합에 대한 터미널 클래스 목록을 가져옵니다.

GetDynamicTerminalClasses GetDynamicTerminalClasses 메서드는 래퍼 메서드에 의해 호출되어 이 주소에서 사용할 수 있는 동적 터미널 클래스의 배열을 가져옵니다.
GetEvent ITMSPAddress::GetEvent(tapi3.h) 메서드는 이벤트 정보를 검색합니다.
GetEvent ITMSPAddress::GetEvent(msp.h) 메서드는 이벤트 정보를 검색합니다.
GetForwardType GetForwardType 메서드는 전달 모드를 가져옵니다.
GetForwardType2 GetForwardType2 메서드는 호출자 주소로 지정된 현재 전달 모드를 가져옵니다.
GetID GetID 메서드는 현재 주소와 연결된 지정된 디바이스 클래스에 대한 디바이스 식별자를 반환합니다.
GetID GetID 메서드는 현재 호출과 연결된 디바이스의 식별자를 가져옵니다.
GetIDAsVariant GetIDAsVariant 메서드는 현재 호출과 연결된 디바이스의 식별자를 가져옵니다.
GetPhoneCapsBuffer GetPhoneCapsBuffer 메서드는 전달된 PHONECAPS_BUFFER 열거형을 기반으로 휴대폰에 대한 버퍼 기능/정보를 가져옵니다.
GetPhoneFromTerminal GetPhoneFromTerminal 메서드는 터미널과 연결된 전화 개체를 반환합니다. 하나의 휴대폰만 터미널에 연결할 수 있습니다.
GetPluggableSuperclasses GetPluggableSuperclasses 메서드는 레지스트리의 모든 플러그형 터미널 슈퍼클래스에 대한 공용 CLSID를 가져옵니다.

GetPluggableTerminalClasses GetPluggableTerminalClasses 메서드는 터미널 슈퍼클래스에 등록된 모든 플러그형 터미널에 대한 터미널 클래스를 나열합니다.
GetState GetState 메서드는 MSPCall 개체에 의해 호출됩니다. 스트림의 현재 상태 반환합니다. 기본 구현은 E_NOTIMPL 반환합니다.
GetStaticTerminals GetStaticTerminals 메서드는 래퍼 메서드(get_StaticTerminals 및 EnumerateStaticTerminals)에 의해 호출되어 이 주소에서 사용할 수 있는 정적 터미널 배열을 가져옵니다.
GetTerminalClassInfo GetTerminalClassInfo 메서드는 특정 터미널에 대한 레지스트리에서 모든 정보를 가져옵니다.
GetTerminalSuperclassInfo GetTerminalSuperclassInfo 메서드는 현재 터미널 슈퍼클래스에 대한 레지스트리에서 클래스 정보를 가져옵니다.
HandleGraphEvent HandleGraphEvent 메서드는 DispatchGraphEvent 정적 메서드에 의해 호출되어 호출 개체가 이벤트를 처리할 instance 있습니다.
HandleStreamEvent HandleStreamEvent 메서드는 TAPI에 이벤트를 보내기 위해 스트림에서 호출됩니다.
HandleTSPData 파생 호출 개체에서 HandleTSPData 메서드를 호출하여 스트림이 TSP 명령을 처리할 수 있도록 할 수 있습니다.
HandoffDirect HandoffDirect 메서드는 다른 애플리케이션에 대한 호출을 해제합니다. 이는 애플리케이션에 더 이상 호출 소유권이 필요하지 않음을 나타냅니다.
HandoffIndirect HandoffIndirect 메서드는 호출의 미디어 유형에 따라 다른 애플리케이션에 대한 호출을 해제합니다.

보류 Hold 메서드는 보류에서 호출을 배치하거나 제거합니다.
Init Init 메서드는 MSP 호출 개체를 초기화하기 위해 MSP 주소 개체(CreateMSPCall 메서드)에서 호출됩니다. 파생 클래스는 전달된 정보를 사용하여 멤버를 초기화해야 합니다.
Init Init 메서드는 MSP 호출 개체를 초기화하기 위해 MSP 주소 개체(CreateMSPCall 메서드)에서 호출됩니다.
Init Init 메서드는 스트림을 만들 때 MSPCall에 의해 호출됩니다. 멤버를 초기화하고, 호출 개체에서 MSPCallAddRef를 호출하고, 필터 그래프의 다양한 인터페이스를 쿼리합니다.
초기화 MSP가 로드될 때 ITMSPAddress::Initialize(tapi3.h) 메서드가 호출됩니다.
초기화 Initialize 메서드는 TAPI를 초기화합니다. 다른 TAPI 3 메서드를 호출하기 전에 이 메서드를 호출해야 합니다. 애플리케이션은 TAPI 세션을 종료할 때 Shutdown 메서드를 호출해야 합니다.
초기화 MSP가 로드될 때 ITMSPAddress::Initialize(msp.h) 메서드가 호출됩니다.
InitializeDynamic InitializeDynamic 메서드는 플러그형 터미널에 대한 기본 터미널 개체 만들기를 수행합니다.
InternalCreateStream InternalCreateStream 메서드는 CreateStream에서 스트림 개체를 만들기 위해 호출됩니다(호출자는 인수 검사를 수행). CreateStreamObject를 사용하여 스트림 개체를 만들고 초기화해야 합니다.
InternalCreateStream InternalCreateStream 메서드는 CreateStream에서 스트림 개체를 만들기 위해 호출됩니다(호출자는 인수 검사를 수행).
IsFullDuplex IsFullDuplex 메서드는 주소가 웨이브 디바이스를 지원하는지 여부를 나타냅니다.

IsValidSetOfMediaTypes IsValidSetOfMediaTypes 메서드는 지정된 미디어 형식이 0이 아니고 지정된 마스크에 있는지 확인합니다.
lineAccept lineAccept 함수는 지정된 제공된 호출을 수락합니다. 필요에 따라 지정된 사용자-사용자 정보를 통화 당사자에게 보낼 수 있습니다.
lineAddProvider lineAddProvider 함수(tapi.h)는 새로운 전화 통신 서비스 공급자를 전화 통신 시스템에 설치합니다.
lineAddProviderA lineAddProvider 함수는 전화 통신 시스템에 새 전화 통신 서비스 공급자를 설치합니다. (lineAddProviderA)
lineAddProviderW lineAddProviderW(유니코드) 함수(tapi.h)는 전화 통신 시스템에 새 전화 통신 서비스 공급자를 설치합니다.
lineAddToConference lineAddToConference 함수는 hConsultCall로 지정된 호출을 hConfCall로 지정된 전화 회의에 추가합니다.
lineAgentSpecific lineAgentSpecific 함수를 사용하면 애플리케이션이 주소와 연결된 에이전트 처리기의 독점 처리기 관련 함수에 액세스할 수 있습니다.
lineAnswer lineAnswer 함수는 지정된 제품 호출에 응답합니다.
lineBlindTransfer lineBlindTransfer 함수(tapi.h)는 지정된 대상 주소에 대한 지정된 호출의 블라인드 또는 단일 단계 전송을 수행합니다.
lineBlindTransferA lineBlindTransfer 함수는 지정된 대상 주소에 대한 지정된 호출의 블라인드 또는 단일 단계 전송을 수행합니다. (lineBlindTransferA)

lineBlindTransferW lineBlindTransfer(유니코드) 함수(tapi.h)는 지정된 대상 주소에 대한 지정된 호출의 블라인드 또는 단일 단계 전송을 수행합니다.
LINECALLBACK lineCallback 함수는 애플리케이션에서 제공하는 함수 이름의 자리 표시자입니다.
lineClose lineClose 함수는 지정된 열린 줄 디바이스를 닫습니다.
lineCompleteCall lineCompleteCall 함수는 정상적으로 연결할 수 없는 호출을 대신 완료하는 방법을 지정합니다.
lineCompleteTransfer lineCompleteTransfer 함수는 상담 통화에 연결된 당사자에게 지정된 호출의 전송을 완료합니다.
lineConfigDialog lineConfigDialog 함수(tapi.h)를 사용하면 지정된 줄 디바이스의 공급자가 대화 상자를 표시합니다.
lineConfigDialogA lineConfigDialog 함수를 사용하면 지정된 줄 디바이스의 공급자가 사용자가 줄 디바이스와 관련된 매개 변수를 구성할 수 있도록 대화 상자(애플리케이션의 hwndOwner에 연결됨)를 표시합니다. (lineConfigDialogA)
lineConfigDialogEdit lineConfigDialogEdit 함수(tapi.h)를 사용하면 지정된 줄 디바이스의 공급자가 대화 상자를 표시합니다.
lineConfigDialogEditA lineConfigDialogEdit 함수를 사용하면 지정된 줄 디바이스의 공급자가 대화 상자(애플리케이션의 hwndOwner에 연결됨)를 표시하여 사용자가 줄 디바이스와 관련된 매개 변수를 구성할 수 있습니다. (lineConfigDialogEditA)
lineConfigDialogEditW lineConfigDialogEditW(유니코드) 함수(tapi.h)를 사용하면 지정된 줄 디바이스의 공급자가 대화 상자를 표시합니다.
lineConfigDialogW

lineConfigDialogW(유니코드) 함수(tapi.h)를 사용하면 지정된 줄 디바이스의 공급자가 대화 상자를 표시합니다.
lineConfigProvider lineConfigProvider 함수를 사용하면 서비스 공급자가 구성 대화 상자를 표시합니다.
lineCreateAgentA lineCreateAgent 함수는 새 에이전트 개체를 만듭니다. LINEPROXYREQUEST_CREATEAGENT 형식의 LINEPROXYREQUEST 구조를 참조하여 등록된 프록시 함수 처리기로 보낼 LINE_PROXYREQUEST 메시지를 생성합니다. (ANSI)
lineCreateAgentSessionA lineCreateAgentSession 함수는 새 AgentSession 개체를 만듭니다. LINEPROXYREQUEST_CREATEAGENTSESSION 형식의 LINEPROXYREQUEST 구조를 참조하여 등록된 프록시 함수 처리기로 보낼 LINE_PROXYREQUEST 메시지를 생성합니다. (ANSI)
lineCreateAgentSessionW lineCreateAgentSession 함수는 새 AgentSession 개체를 만듭니다. LINEPROXYREQUEST_CREATEAGENTSESSION 형식의 LINEPROXYREQUEST 구조를 참조하여 등록된 프록시 함수 처리기로 보낼 LINE_PROXYREQUEST 메시지를 생성합니다. (유니코드)
lineCreateAgentW lineCreateAgent 함수는 새 에이전트 개체를 만듭니다. LINEPROXYREQUEST_CREATEAGENT 형식의 LINEPROXYREQUEST 구조를 참조하여 등록된 프록시 함수 처리기로 보낼 LINE_PROXYREQUEST 메시지를 생성합니다. (유니코드)
lineDeallocateCall 지정된 호출 핸들의 할당을 취소합니다.
lineDevSpecific lineDevSpecific 함수를 사용하면 서비스 공급자가 다른 TAPI 함수에서 제공하지 않는 기능에 대한 액세스를 제공할 수 있습니다.
lineDevSpecificFeature lineDevSpecificFeature 함수를 사용하면 서비스 공급자가 다른 TAPI 함수에서 제공하지 않는 기능에 액세스할 수 있습니다.
lineDial lineDial 함수(tapi.h)는 지정된 통화에서 지정된 전화 걸기 가능 번호로 전화를 켭니다.

lineDialA lineDial 함수는 지정된 통화에서 지정된 전화 걸기 가능 번호로 전화를 켭니다. (lineDialA)
lineDialW lineDialW(유니코드) 함수(tapi.h)는 지정된 통화에서 지정된 전화 걸기 가능 번호로 전화를 켭니다.
lineDrop lineDrop 함수는 지정된 호출을 삭제하거나 연결을 끊습니다. 애플리케이션에는 호출 연결 끊기의 일부로 전송할 사용자-사용자 정보를 지정하는 옵션이 있습니다.
lineForward lineForward 함수(tapi.h)는 지정된 전달 지침에 따라 지정된 줄의 지정된 주소로 향하는 호출을 전달합니다.
lineForwardA lineForward 함수는 지정된 전달 지침에 따라 지정된 줄의 지정된 주소로 향하는 호출을 전달합니다. (lineForwardA)
lineForwardW lineForwardW(유니코드) 함수는 지정된 전달 지침에 따라 지정된 줄의 지정된 주소로 향하는 호출을 전달합니다.
lineGatherDigits lineGatherDigits 함수(tapi.h)는 지정된 호출에서 버퍼링된 숫자 수집을 시작합니다.
lineGatherDigitsA lineGatherDigits 함수는 지정된 호출에서 버퍼링된 숫자 수집을 시작합니다. 애플리케이션은 수집할 숫자와 최대 자릿수를 배치할 버퍼를 지정합니다. (lineGatherDigitsA)
lineGatherDigitsW lineGatherDigitsW(유니코드) 함수는 지정된 호출에서 버퍼링된 숫자 수집을 시작합니다.
lineGenerateDigits lineGenerateDigits 함수(tapi.h)는 지정된 신호 모드를 사용하여 지정된 호출에서 지정된 숫자를 인밴드 톤으로 생성합니다.
lineGenerateDigitsA

지정된 신호 모드를 사용하여 지정된 호출에서 지정된 숫자를 인밴드 톤으로 생성합니다. (lineGenerateDigitsA)
lineGenerateDigitsW lineGenerateDigitsW(유니코드) 함수는 지정된 신호 모드를 사용하여 호출에서 지정된 숫자를 인밴드 톤으로 생성하기 시작합니다.
lineGenerateTone lineGenerateTone 함수는 지정된 호출에 대해 지정된 인밴드 톤을 생성합니다.
lineGetAddressCaps lineGetAddressCaps 함수(tapi.h)는 지정된 줄 디바이스에서 지정된 주소를 쿼리하여 전화 통신 기능을 확인합니다.
lineGetAddressCapsA lineGetAddressCaps 함수는 지정된 줄 디바이스에서 지정된 주소를 쿼리하여 전화 통신 기능을 확인합니다. (lineGetAddressCapsA)
lineGetAddressCapsW lineGetAddressCapsW(유니코드) 함수(tapi.h)는 지정된 줄 디바이스에서 지정된 주소를 쿼리하여 전화 통신 기능을 확인합니다.
lineGetAddressID lineGetAddressID 함수(tapi.h)는 지정된 줄에서 주소와 연결된 주소 식별자를 다른 형식으로 반환합니다.
lineGetAddressIDA lineGetAddressID 함수는 지정된 줄에서 주소와 연결된 주소 식별자를 다른 형식으로 반환합니다. (lineGetAddressIDA)
lineGetAddressIDW lineGetAddressIDW(유니코드) 함수(tapi.h)는 지정된 줄에서 주소와 연결된 주소 식별자를 다른 형식으로 반환합니다.
lineGetAddressStatus lineGetAddressStatus 함수(tapi.h)를 사용하면 애플리케이션이 현재 상태 대해 지정된 주소를 쿼리할 수 있습니다.
lineGetAddressStatusA

lineGetAddressStatus 함수를 사용하면 애플리케이션이 현재 상태 대해 지정된 주소를 쿼리할 수 있습니다. (lineGetAddressStatusA)
lineGetAddressStatusW lineGetAddressStatusW 함수(tapi.h)를 사용하면 애플리케이션이 현재 상태 대해 지정된 주소를 쿼리할 수 있습니다.
lineGetAgentActivityListA lineGetAgentActivityList 함수는 lineSetAgentActivity를 사용하여 현재 에이전트가 실제로 수행하는 함수를 나타내기 위해 애플리케이션에서 선택할 수 있는 활동의 ID를 가져옵니다. (ANSI)
lineGetAgentActivityListW lineGetAgentActivityList 함수는 lineSetAgentActivity를 사용하여 현재 에이전트가 실제로 수행하는 함수를 나타내기 위해 애플리케이션에서 선택할 수 있는 활동의 ID를 가져옵니다. (유니코드)
lineGetAgentCapsA lineGetAgentCaps 함수는 지정된 줄 디바이스에서 지원되는 에이전트 관련 기능을 가져옵니다. 특정 에이전트의 이름이 지정된 경우 기능에는 에이전트가 로그인할 수 있는 ACD 그룹 목록이 포함됩니다. (ANSI)
lineGetAgentCapsW lineGetAgentCaps 함수는 지정된 줄 디바이스에서 지원되는 에이전트 관련 기능을 가져옵니다. 특정 에이전트의 이름이 지정된 경우 기능에는 에이전트가 로그인할 수 있는 ACD 그룹 목록이 포함됩니다. (유니코드)
lineGetAgentGroupListA lineGetAgentGroupList 함수는 현재 워크스테이션에 로그인한 에이전트가 자동 호출 배포자에 로그인할 수 있는 에이전트 그룹의 ID(큐, 감독자, 기술 수준 등의 조합)를 가져옵니다. (ANSI)
lineGetAgentGroupListW lineGetAgentGroupList 함수는 현재 워크스테이션에 로그인한 에이전트가 자동 호출 배포자에 로그인할 수 있는 에이전트 그룹의 ID(큐, 감독자, 기술 수준 등의 조합)를 가져옵니다. (유니코드)
lineGetAgentInfo lineGetAgentInfo 함수는 특정 에이전트 핸들과 연결된 ACD 정보를 포함하는 구조를 반환합니다.
lineGetAgentSessionInfo lineGetAgentSessionInfo 함수는 특정 에이전트 세션 핸들과 연결된 ACD 정보를 포함하는 구조를 반환합니다.

lineGetAgentSessionList lineGetAgentSessionList 함수는 지정된 에이전트에 대해 만든 에이전트 세션 목록을 반환합니다.
lineGetAgentStatusA lineGetAgentStatus 함수는 지정된 주소에서 에이전트 관련 상태 가져옵니다. (ANSI)
lineGetAgentStatusW lineGetAgentStatus 함수는 지정된 주소에서 에이전트 관련 상태 가져옵니다. (유니코드)
lineGetAppPriority lineGetAppPriority 함수(tapi.h)를 사용하면 애플리케이션이 특정 미디어 모드의 핸드오프 우선 순위 목록에 있는지 여부를 확인할 수 있습니다.
lineGetAppPriorityA 애플리케이션이 특정 미디어 모드 또는 보조 전화 통신 요청 모드의 핸드오프 우선 순위 목록에 있는지 여부와 우선 순위 목록에서 해당 위치를 확인할 수 있도록 합니다. (lineGetAppPriorityA)
lineGetAppPriorityW lineGetAppPriorityW(유니코드) 함수를 사용하면 애플리케이션이 특정 미디어 모드의 핸드오프 우선 순위 목록에 있는지 여부를 확인할 수 있습니다.
lineGetCallInfo lineGetCallInfo 함수(tapi.h)를 사용하면 애플리케이션이 지정된 호출에 대한 고정된 정보를 가져올 수 있습니다.
lineGetCallInfoA lineGetCallInfo 함수를 사용하면 애플리케이션이 지정된 호출에 대한 고정 정보를 가져올 수 있습니다. (lineGetCallInfoA)
lineGetCallInfoW lineGetCallInfoW(유니코드) 함수(tapi.h)를 사용하면 애플리케이션이 지정된 호출에 대한 고정 정보를 가져올 수 있습니다.
lineGetCallStatus lineGetCallStatus 함수는 지정된 호출의 현재 상태 반환합니다.
lineGetConfRelatedCalls

lineGetConfRelatedCalls 함수는 지정된 호출과 동일한 전화 회의 호출의 일부인 호출 핸들 목록을 반환합니다.
lineGetCountry lineGetCountry 함수(tapi.h)는 저장된 전화 걸기 규칙 및 지정된 국가/지역과 관련된 기타 정보를 가져옵니다.
lineGetCountryA lineGetCountry 함수는 저장된 전화 걸기 규칙 및 지정된 국가/지역, 국가/지역 목록의 첫 번째 국가/지역 또는 모든 국가/지역과 관련된 기타 정보를 가져옵니다. (lineGetCountryA)
lineGetCountryW lineGetCountryW(유니코드) 함수는 저장된 전화 걸기 규칙 및 지정된 국가/지역과 관련된 기타 정보를 가져옵니다.
lineGetDevCaps lineGetDevCaps 함수(tapi.h)는 지정된 라인 디바이스를 쿼리하여 전화 통신 기능을 확인합니다.
lineGetDevCapsA lineGetDevCaps 함수는 지정된 줄 디바이스를 쿼리하여 전화 통신 기능을 확인합니다. 반환된 정보는 줄 디바이스의 모든 주소에 유효합니다. (lineGetDevCapsA)
lineGetDevCapsW lineGetDevCapsW(유니코드) 함수(tapi.h)는 지정된 줄 디바이스를 쿼리하여 전화 통신 기능을 확인합니다.
lineGetDevConfig lineGetDevConfig 함수(tapi.h)는 줄 및 디바이스 클래스와 관련된 내용인 "불투명" 데이터 구조 개체를 반환합니다.
lineGetDevConfigA lineGetDevConfig 함수는 줄(서비스 공급자) 및 디바이스 클래스와 관련된 내용인 "불투명" 데이터 구조 개체를 반환합니다. (lineGetDevConfigA)
lineGetDevConfigW lineGetDevConfigW(유니코드) 함수는 줄 및 디바이스 클래스와 관련된 내용인 "불투명" 데이터 구조 개체를 반환합니다.
lineGetGroupListA

lineGetGroupList 함수는 ACD 시스템에서 사용할 수 있는 ACD 그룹 목록을 반환합니다. (ANSI)
lineGetGroupListW lineGetGroupList 함수는 ACD 시스템에서 사용할 수 있는 ACD 그룹 목록을 반환합니다. (유니코드)
lineGetIcon lineGetIcon 함수(tapi.h)를 사용하면 애플리케이션이 사용자에게 표시할 서비스 라인 디바이스별(또는 공급자별) 아이콘을 검색할 수 있습니다.
lineGetIconA lineGetIcon 함수를 사용하면 애플리케이션이 사용자에게 표시할 서비스 라인 디바이스별(또는 공급자별) 아이콘을 검색할 수 있습니다. (lineGetIconA)
lineGetIconW lineGetIconW(유니코드) 함수를 사용하면 애플리케이션이 사용자에게 표시할 서비스 라인 디바이스별 아이콘을 검색할 수 있습니다.
lineGetID lineGetID 함수(tapi.h)는 선택한 줄, 주소 또는 호출과 연결된 지정된 디바이스 클래스에 대한 디바이스 식별자를 반환합니다.
lineGetIDA lineGetID 함수는 선택한 줄, 주소 또는 호출과 연결된 지정된 디바이스 클래스에 대한 디바이스 식별자를 반환합니다. (lineGetIDA)
lineGetIDW lineGetIDW(유니코드) 함수(tapi.h)는 선택한 줄, 주소 또는 호출과 연결된 지정된 디바이스 클래스에 대한 디바이스 식별자를 반환합니다.
lineGetLineDevStatus lineGetLineDevStatus 함수(tapi.h)를 사용하면 애플리케이션이 현재 상태 대해 지정된 열린 줄 디바이스를 쿼리할 수 있습니다.
lineGetLineDevStatusA lineGetLineDevStatus 함수를 사용하면 애플리케이션이 현재 상태 대해 지정된 열린 줄 디바이스를 쿼리할 수 있습니다. (lineGetLineDevStatusA)
lineGetLineDevStatusW

lineGetLineDevStatusW(유니코드) 함수를 사용하면 애플리케이션이 현재 상태 대해 지정된 열린 줄 디바이스를 쿼리할 수 있습니다.
lineGetMessage lineGetMessage 함수는 이벤트 핸들 알림 메커니즘을 사용하는 애플리케이션에 배달하기 위해 큐에 대기 중인 다음 TAPI 메시지를 반환합니다(자세한 내용은 lineInitializeEx 참조).
lineGetNewCalls lineGetNewCalls 함수는 애플리케이션에 현재 핸들이 없는 지정된 줄 또는 주소의 호출에 대한 호출 핸들을 반환합니다. 애플리케이션에는 이러한 호출에 대한 모니터 권한이 부여됩니다.
lineGetNumRings lineGetNumRings 함수는 호출에 응답하기 전에 지정된 주소에서 들어오는 호출이 울리는 링 수를 결정합니다.
lineGetProviderList lineGetProviderList 함수(tapi.h)는 현재 전화 통신 시스템에 설치된 서비스 공급자 목록을 반환합니다.
lineGetProviderListA lineGetProviderList 함수는 현재 전화 통신 시스템에 설치된 서비스 공급자 목록을 반환합니다. (lineGetProviderListA)
lineGetProviderListW lineGetProviderListW(유니코드) 함수(tapi.h)는 현재 전화 통신 시스템에 설치된 서비스 공급자 목록을 반환합니다.
lineGetProxyStatus lineGetProxyStatus 함수는 지정된 디바이스에 대해 현재 서비스되고 있는 프록시 요청 형식 목록을 반환합니다.
lineGetQueueInfo lineGetQueueInfo 함수는 특정 큐와 연결된 ACD 정보를 포함하는 구조를 반환합니다.
lineGetQueueListA lineGetQueueList 함수는 특정 ACD 그룹과 연결된 큐 목록을 반환합니다. (ANSI)
lineGetQueueListW lineGetQueueList 함수는 특정 ACD 그룹과 연결된 큐 목록을 반환합니다. (유니코드)

lineGetRequest lineGetRequest 함수(tapi.h)는 지정된 요청 모드에 대한 다음 프록시별 요청을 검색합니다.
lineGetRequestA 지정된 요청 모드에 대한 다음 프록시별 요청을 검색합니다. (lineGetRequestA)
lineGetRequestW lineGetRequestW(유니코드) 함수(tapi.h)는 지정된 요청 모드에 대한 다음 프록시별 요청을 검색합니다.
lineGetStatusMessages lineGetStatusMessages 함수를 사용하면 애플리케이션이 지정된 줄 또는 해당 주소에 대한 상태 변경 내용과 관련된 이벤트에 대해 수신하도록 설정된 알림 메시지를 쿼리할 수 있습니다.
lineGetTranslateCaps lineGetTranslateCaps 함수(tapi.h)는 주소 변환 기능을 반환합니다.
lineGetTranslateCapsA lineGetTranslateCaps 함수는 주소 변환 기능을 반환합니다. (lineGetTranslateCapsA)
lineGetTranslateCapsW lineGetTranslateCapsW(유니코드) 함수(tapi.h)는 주소 변환 기능을 반환합니다.
lineHandoff lineHandoff 함수(tapi.h)는 다른 애플리케이션에 대해 지정된 호출의 소유권을 부여합니다.
lineHandoffA lineHandoff 함수는 다른 애플리케이션에 대해 지정된 호출의 소유권을 부여합니다. 애플리케이션은 파일 이름으로 직접 지정하거나 지정된 미디어 모드의 호출을 처리하는 우선 순위가 가장 높은 애플리케이션으로 간접적으로 지정할 수 있습니다. (lineHandoffA)
lineHandoffW lineHandoffW(유니코드) 함수(tapi.h)는 다른 애플리케이션에 대해 지정된 호출의 소유권을 제공합니다.
lineHold lineHold 함수는 지정된 호출을 보류합니다.

lineInitialize lineInitialize 함수는 사용되지 않습니다. API 버전 1.3 및 1.4를 사용하는 애플리케이션과의 이전 버전과의 호환성을 위해 Tapi.dll 및 Tapi32.dll 계속 내보냅니다.
lineInitializeExA lineInitializeEx 함수는 줄 추상화의 후속 사용을 위해 애플리케이션의 TAPI 사용을 초기화합니다. (ANSI)
lineInitializeExW lineInitializeEx 함수는 줄 추상화의 후속 사용을 위해 애플리케이션의 TAPI 사용을 초기화합니다. (유니코드)
lineMakeCall lineMakeCall 함수(tapi.h)는 지정된 줄에서 지정된 대상 주소에 대한 호출을 배치합니다.
lineMakeCallA lineMakeCall 함수는 지정된 줄에서 지정된 대상 주소에 대한 호출을 배치합니다. 필요에 따라 기본 호출 설정 매개 변수 외에 다른 항목이 요청되면 호출 매개 변수를 지정할 수 있습니다. (lineMakeCallA)
lineMakeCallW lineMakeCallW(유니코드) 함수(tapi.h)는 지정된 줄에서 지정된 대상 주소에 대한 호출을 배치합니다.
lineMonitorDigits lineMonitorDigits 함수는 호출 시 수신된 숫자의 버퍼링되지 않은 검색을 사용하도록 설정하고 사용하지 않도록 설정합니다. 지정된 숫자 모드의 숫자가 검색될 때마다 검색된 숫자를 나타내는 메시지가 애플리케이션으로 전송됩니다.
lineMonitorMedia lineMonitorMedia 함수는 지정된 호출에서 미디어 유형(모드)의 검색을 활성화하고 사용하지 않도록 설정합니다. 미디어 유형이 검색되면 메시지가 애플리케이션으로 전송됩니다. 자세한 내용은 ITLegacyCallMediaControl::MonitorMedia를 참조하세요.
lineMonitorTones lineMonitorTones 함수는 호출 시 인밴드 톤 검색을 활성화하고 사용하지 않도록 설정합니다. 지정된 톤이 검색될 때마다 메시지가 애플리케이션으로 전송됩니다.
lineNegotiateAPIVersion

lineNegotiateAPIVersion 함수를 사용하면 애플리케이션에서 사용할 API 버전을 협상할 수 있습니다.
lineNegotiateExtVersion lineNegotiateExtVersion 함수를 사용하면 애플리케이션이 지정된 줄 디바이스에서 사용할 확장 버전을 협상할 수 있습니다. 애플리케이션이 확장을 지원하지 않는 경우 이 작업을 호출할 필요가 없습니다.
lineOpen lineOpen 함수(tapi.h)는 디바이스 식별자에 의해 지정된 줄 디바이스를 열고 해당 열린 줄 디바이스에 대한 줄 핸들을 반환합니다.
lineOpenA lineOpen 함수는 디바이스 식별자에 의해 지정된 줄 디바이스를 열고 열린 해당 줄 디바이스에 대한 줄 핸들을 반환합니다. 이 줄 핸들은 라인 디바이스의 후속 작업에 사용됩니다. (lineOpenA)
lineOpenW lineOpenW(유니코드) 함수(tapi.h)는 디바이스 식별자가 지정한 줄 디바이스를 열고 해당 열린 줄 디바이스에 대한 줄 핸들을 반환합니다.
linePark linePark 함수(tapi.h)는 지정된 공원 모드에 따라 지정된 호출을 파킹합니다.
lineParkA linePark 함수는 지정된 대기 모드에 따라 지정된 호출을 주차합니다. (lineParkA)
lineParkW lineParkW(유니코드) 함수(tapi.h)는 지정된 대기 모드에 따라 지정된 호출을 주차합니다.
linePickup linePickup 함수(tapi.h)는 지정된 대상 주소에서 호출 경고를 선택하고 선택한 호출에 대한 호출 핸들을 반환합니다.
linePickupA linePickup 함수는 지정된 대상 주소에서 호출 경고를 선택하고 선택한 호출에 대한 호출 핸들을 반환합니다. (linePickupA)
linePickupW

linePickupW(유니코드) 함수(tapi.h)는 지정된 대상 주소에서 호출 경고를 선택하고 선택한 호출에 대한 호출 핸들을 반환합니다.
linePrepareAddToConference linePrepareAddToConference 함수(tapi.h)는 다른 파티의 추가를 위해 기존 전화 회의를 준비합니다.
linePrepareAddToConferenceA linePrepareAddToConference 함수는 다른 파티의 추가를 위해 기존 전화 회의를 준비합니다. (linePrepareAddToConferenceA)
linePrepareAddToConferenceW linePrepareAddToConferenceW(유니코드) 함수(tapi.h)는 다른 파티의 추가를 위해 기존 전화 회의를 준비합니다.
lineProxyMessage lineProxyMessage 함수는 등록된 프록시 요청 처리기에서 해당 역할과 관련된 TAPI 메시지를 생성하는 데 사용됩니다.
lineProxyResponse 서버의 ACD 에이전트 처리기와 같이 등록된 프록시 처리기의 프록시 요청 완료를 나타냅니다.
lineRedirect lineRedirect 함수(tapi.h)는 지정된 제품 호출을 지정된 대상 주소로 리디렉션합니다.
lineRedirectA lineRedirect 함수는 지정된 제품 호출을 지정된 대상 주소로 리디렉션합니다. (lineRedirectA)
lineRedirectW lineRedirectW(유니코드) 함수(tapi.h)는 지정된 제품 호출을 지정된 대상 주소로 리디렉션합니다.
lineRegisterRequestRecipient lineRegisterRequestRecipient 함수는 호출하는 애플리케이션을 지정된 요청 모드에 대한 요청 수신자로 등록합니다.
lineReleaseUserUserInfo lineReleaseUserUserInfo 함수는 애플리케이션이 LINECALLINFO 구조에 포함된 사용자-사용자 정보를 처리했음을 서비스 공급자에게 알릴 수 있습니다.

lineRemoveFromConference lineRemoveFromConference 함수는 현재 속한 전화 회의 호출에서 지정된 호출을 제거합니다. 전화 회의의 나머지 통화는 영향을 받지 않습니다.
lineRemoveProvider lineRemoveProvider 함수는 전화 통신 시스템에서 기존 전화 통신 서비스 공급자를 제거합니다.
lineSecureCall lineSecureCall 함수는 호출의 미디어 스트림에 영향을 줄 수 있는 중단 또는 간섭으로부터 호출을 보호합니다.
lineSendUserUserInfo lineSendUserUserInfo 함수는 지정된 호출에서 원격 당사자에게 사용자-사용자 정보를 보냅니다.
lineSetAgentActivity lineSetAgentActivity 함수는 특정 주소와 연결된 에이전트 활동 코드를 설정합니다.
lineSetAgentGroup lineSetAgentGroup 함수는 에이전트가 특정 주소에 로그인되는 에이전트 그룹을 설정합니다.
lineSetAgentMeasurementPeriod lineSetAgentMeasurementPeriod 함수는 특정 에이전트와 연결된 측정 기간을 설정합니다.
lineSetAgentSessionState lineSetAgentSessionState 함수는 특정 에이전트 세션 핸들과 연결된 에이전트 세션 상태를 설정합니다.
lineSetAgentState lineSetAgentState 함수는 특정 주소와 연결된 에이전트 상태를 설정합니다.
lineSetAgentStateEx lineSetAgentStateEx 함수는 특정 에이전트 핸들과 연결된 에이전트 상태를 설정합니다.
lineSetAppPriority lineSetAppPriority 함수(tapi.h)를 사용하면 애플리케이션이 핸드오프 우선 순위 목록에서 우선 순위를 설정할 수 있습니다.

lineSetAppPriorityA 애플리케이션이 특정 미디어 유형 또는 보조 전화 통신 요청 모드의 핸드오프 우선 순위 목록에서 우선 순위를 설정하거나 우선 순위 목록에서 자신을 제거할 수 있도록 합니다. (lineSetAppPriorityA)
lineSetAppPriorityW lineSetAppPriorityW(유니코드) 함수(tapi.h)를 사용하면 애플리케이션이 핸드오프 우선 순위 목록에서 우선 순위를 설정할 수 있습니다.
lineSetAppSpecific lineSetAppSpecific 함수를 사용하면 애플리케이션이 지정된 호출의 호출 정보 레코드의 애플리케이션별 필드를 설정할 수 있습니다.
lineSetCallData lineSetCallData 함수는 LINECALLINFO에서 CallData 멤버를 설정합니다.
lineSetCallParams lineSetCallParams 함수를 사용하면 애플리케이션이 전달자 모드 및/또는 기존 호출의 속도 매개 변수를 변경할 수 있습니다.
lineSetCallPrivilege lineSetCallPrivilege 함수는 애플리케이션의 권한을 지정된 권한으로 설정합니다.
lineSetCallQualityOfService lineSetCallQualityOfService 함수를 사용하면 애플리케이션이 기존 호출에 대한 서비스 매개 변수(예약된 용량 및 성능 보장)의 품질을 변경하려고 시도할 수 있습니다.
lineSetCallTreatment lineSetCallTreatment 함수는 응답이 없거나 수신 대기 중인 통화에서 파티 소리를 설정합니다. 기본 매개 변수 유효성 검사를 제외하고 서비스 공급자에 대한 TAPI의 직선 통과입니다.
lineSetCurrentLocation lineSetCurrentLocation 함수는 주소 변환의 컨텍스트로 사용되는 위치를 설정합니다.
lineSetDevConfig lineSetDevConfig 함수(tapi.h)를 사용하면 애플리케이션이 라인 디바이스에서 미디어 스트림 디바이스의 구성을 이전 설정으로 복원할 수 있습니다.

lineSetDevConfigA lineSetDevConfig 함수를 사용하면 애플리케이션이 라인 디바이스의 미디어 스트림 디바이스 구성을 lineGetDevConfig를 사용하여 이전에 가져온 설정으로 복원할 수 있습니다. (lineSetDevConfigA)
lineSetDevConfigW lineSetDevConfigW(유니코드) 함수(tapi.h)를 사용하면 애플리케이션이 라인 디바이스에서 미디어 스트림 디바이스의 구성을 이전 설정으로 복원할 수 있습니다.
lineSetLineDevStatus lineSetLineDevStatus 함수는 선 디바이스를 상태 설정합니다.
lineSetMediaControl lineSetMediaControl 함수는 지정된 줄, 주소 또는 호출과 연결된 미디어 스트림에서 제어 작업을 사용하거나 사용하지 않도록 설정합니다.
lineSetMediaMode lineSetMediaMode 함수는 LINECALLINFO 구조에서 지정된 호출의 미디어 형식을 설정합니다. 자세한 내용은 ITLegacyCallMediaControl::SetMediaType을 참조하세요.
lineSetNumRings lineSetNumRings 함수는 들어오는 호출에 응답하기 전에 발생해야 하는 링 수를 설정합니다.
lineSetQueueMeasurementPeriod lineSetQueueMeasurementPeriod 함수는 특정 큐와 연결된 측정 기간을 설정합니다.
lineSetStatusMessages lineSetStatusMessages 함수를 사용하면 애플리케이션이 지정된 줄 또는 해당 주소에 대한 상태 변경 내용과 관련된 이벤트에 대해 수신할 알림 메시지를 지정할 수 있습니다.
lineSetTerminal lineSetTerminal 함수를 사용하면 애플리케이션이 지정된 줄, 주소 또는 호출과 관련된 터미널 정보를 라우팅할 터미널 정보를 지정할 수 있습니다.
lineSetTollList lineSetTollList 함수(tapi.h)는 통행료 목록을 조작합니다.
lineSetTollListA lineSetTollList 함수는 통행료 목록을 조작합니다. (lineSetTollListA)

lineSetTollListW lineSetTollListW(유니코드) 함수(tapi.h)는 통행료 목록을 조작합니다.
lineSetupConference lineSetupConference 함수(tapi.h)는 타사 추가를 위한 전화 회의를 설정합니다.
lineSetupConferenceA lineSetupConference 함수는 타사 추가를 위한 전화 회의를 설정합니다. (lineSetupConferenceA)
lineSetupConferenceW lineSetupConferenceW(유니코드) 함수(tapi.h)는 타사 추가를 위한 전화 회의를 설정합니다.
lineSetupTransfer lineSetupTransfer 함수(tapi.h)는 hCall 매개 변수로 지정된 호출의 전송을 시작합니다.
lineSetupTransferA lineSetupTransfer 함수는 hCall 매개 변수로 지정된 호출의 전송을 시작합니다. (lineSetupTransferA)
lineSetupTransferW lineSetupTransferW(유니코드) 함수(tapi.h)는 hCall 매개 변수로 지정된 호출의 전송을 시작합니다.
lineShutdown lineShutdown 함수는 API의 선 추상화에 대한 애플리케이션의 사용을 종료합니다.
lineSwapHold lineSwapHold 함수는 지정된 활성 호출을 상담 보류 시 지정된 호출로 바꿉니다.
lineTranslateAddress lineTranslateAddress 함수(tapi.h)는 지정된 주소를 다른 형식으로 변환합니다.
lineTranslateAddressA lineTranslateAddress 함수는 지정된 주소를 다른 형식으로 변환합니다. (lineTranslateAddressA)
lineTranslateAddressW lineTranslateAddressW(유니코드) 함수(tapi.h)는 지정된 주소를 다른 형식으로 변환합니다.
lineTranslateDialog

lineTranslateDialog 함수(tapi.h)는 사용자가 전화 번호 위치를 변경하고, 호출 카드 매개 변수를 조정하고, 효과를 볼 수 있는 애플리케이션 모달 대화 상자를 표시합니다.
lineTranslateDialogA lineTranslateDialog 함수는 사용자가 전화를 걸 전화 번호의 현재 위치를 변경하고, 위치를 조정하고, 카드 매개 변수를 호출하고, 효과를 볼 수 있는 애플리케이션 모달 대화 상자를 표시합니다. (lineTranslateDialogA)
lineTranslateDialogW lineTranslateDialogW(유니코드) 함수(tapi.h)에는 사용자가 전화 번호 위치를 변경하고 호출 카드 매개 변수를 조정하고 효과를 볼 수 있는 애플리케이션 모달 대화 상자가 표시됩니다.
lineUncompleteCall lineUncompleteCall 함수는 지정된 줄에서 지정된 호출 완료 요청을 취소합니다.
lineUnhold lineUnhold 함수는 지정된 보류된 호출을 검색합니다.
lineUnpark lineUnpark 함수(tapi.h)는 지정된 주소에 주차된 호출을 검색하고 해당 호출 핸들을 반환합니다.
lineUnparkA lineUnpark 함수는 지정된 주소에 대기된 호출을 검색하고 해당 호출 핸들을 반환합니다. (lineUnparkA)
lineUnparkW lineUnparkW(유니코드) 함수(tapi.h)는 지정된 주소에 주차된 호출을 검색하고 해당 호출 핸들을 반환합니다.
MakeCall MakeCall 메서드는 지정된 당사자를 호출합니다.
ModifyDirectoryObject ModifyDirectoryObject 메서드는 디렉터리 수정 내용을 서버에 커밋합니다.
MonitorMedia MonitorMedia 메서드는 현재 호출에서 지정된 미디어 형식에 대한 모니터링을 설정합니다.
MSPAddressAddRef

MSPAddressAddRef 메서드는 주소에 대한 프라이빗 AddRef 메서드입니다.
MSPAddressRelease MSPAddressRelease 메서드는 주소에 대한 프라이빗 Release 메서드입니다.
MSPCallAddRef MSPCallAddRef 메서드는 호출 개체에 대한 프라이빗 AddRef 메서드입니다.
MSPCallRelease MSPCallRelease 메서드는 호출 개체에 대한 프라이빗 Release 메서드입니다.
NegotiateExtVersion NegotiateExtVersion 메서드를 사용하면 애플리케이션이 지정된 줄 디바이스에서 사용할 확장 버전을 협상할 수 있습니다. 애플리케이션이 공급자별 확장을 지원하지 않는 경우 이 메서드를 호출할 필요가 없습니다.
NegotiateExtVersion NegotiateExtVersion 메서드를 사용하면 애플리케이션이 지정된 전화 장치에서 사용할 확장 버전을 협상할 수 있습니다. 애플리케이션이 공급자별 확장을 지원하지 않는 경우 이 작업을 호출할 필요가 없습니다.
다음 IEnumACDGroup::Next 메서드(tapi3.h)는 열거형 시퀀스에서 지정된 다음 개수의 요소를 가져옵니다.
다음 IEnumAgent::Next 메서드(tapi3.h)는 열거형 시퀀스에서 지정된 다음 개수의 요소를 가져옵니다.
다음 IEnumAgentHandler::Next 메서드(tapi3.h)는 열거형 시퀀스에서 지정된 다음 개수의 요소를 가져옵니다.
다음 IEnumAgentSession::Next 메서드(tapi3.h)는 열거형 시퀀스에서 지정된 다음 개수의 요소를 가져옵니다.
다음 IEnumQueue::Next 메서드(tapi3.h)는 열거형 시퀀스에서 지정된 다음 개수의 요소를 가져옵니다.

다음 Next 메서드는 열거형 시퀀스에서 지정한 다음 개수의 요소를 가져옵니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumAddress.Next)
다음 Next 메서드는 열거형 시퀀스에서 지정한 다음 개수의 요소를 가져옵니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumBstr.Next)
다음 Next 메서드는 열거형 시퀀스에서 지정한 다음 개수의 요소를 가져옵니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumCall.Next)
다음 Next 메서드는 열거형 시퀀스에서 지정한 다음 개수의 요소를 가져옵니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumCallHub.Next)
다음 Next 메서드는 열거형 시퀀스에서 지정한 다음 개수의 요소를 가져옵니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumCallingCard.Next)
다음 Next 메서드는 열거형 시퀀스에서 지정한 다음 개수의 요소를 가져옵니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumLocation.Next)
다음 Next 메서드는 열거형 시퀀스에서 지정한 다음 개수의 요소를 가져옵니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumPhone.Next)
다음 Next 메서드는 열거형 시퀀스에서 지정한 다음 개수의 요소를 가져옵니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumPluggableSuperclassInfo.Next)
다음 Next 메서드는 열거형 시퀀스에서 지정한 다음 개수의 요소를 가져옵니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumPluggableTerminalClassInfo.Next)
다음 Next 메서드는 열거형 시퀀스에서 지정한 다음 개수의 요소를 가져옵니다. (IEnumStream.Next)

다음 Next 메서드는 열거형 시퀀스에서 지정한 다음 개수의 요소를 가져옵니다. (IEnumSubStream.Next)
다음 Next 메서드는 열거형 시퀀스에서 지정한 다음 개수의 요소를 가져옵니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumTerminal.Next)
다음 Next 메서드는 열거형 시퀀스에서 지정한 다음 개수의 요소를 가져옵니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumTerminalClass.Next)
다음 IEnumACDGroup::Next 메서드(tapi3cc.h)는 열거형 시퀀스에서 지정된 다음 개수의 요소를 가져옵니다.
다음 IEnumAgentHandler::Next 메서드(tapi3cc.h)는 열거형 시퀀스에서 지정된 다음 개수의 요소를 가져옵니다.
다음 IEnumAgentHandler::Next 메서드(tapi3cc.h)는 열거형 시퀀스에서 지정된 다음 개수의 요소를 가져옵니다.
다음 IEnumAgentSession::Next 메서드(tapi3cc.h)는 열거형 시퀀스에서 지정된 다음 개수의 요소를 가져옵니다.
다음 IEnumQueue::Next 메서드(tapi3cc.h)는 열거형 시퀀스에서 지정된 다음 개수의 요소를 가져옵니다.
다음 Next 메서드는 열거형 시퀀스에서 지정한 다음 개수의 요소를 가져옵니다. (IEnumDialableAddrs.Next)
다음 Next 메서드는 열거형 시퀀스에서 지정한 다음 개수의 요소를 가져옵니다. (IEnumDirectory.Next)
다음

Next 메서드는 열거형 시퀀스에서 지정한 다음 개수의 요소를 가져옵니다. (IEnumDirectoryObject.Next)
다음 Next 메서드는 열거형 시퀀스에서 지정한 다음 개수의 요소를 가져옵니다. (IEnumMcastScope.Next)
열기 Open 메서드가 이 휴대폰 디바이스를 엽니다. 응용 프로그램이 ITPhone::Close를 호출하거나 TAPI가 종료될 때까지 휴대폰 디바이스가 열린 상태로 유지됩니다.
ParkDirect ParkDirect 메서드는 호출을 지정된 주소에 파킹합니다.
ParkIndirect ParkIndirect 메서드는 호출을 파킹하고 주차된 주소를 반환합니다.
일시 중지 Pause 메서드는 파일의 현재 위치에 남아 있는 작업을 일시 중지합니다.
PauseStream PauseStream 메서드는 스트림을 일시 중지합니다.
PauseSubStream PauseSubStream 메서드는 하위 스트림을 일시 중지합니다.
PHONECALLBACK phoneCallback 함수는 애플리케이션 제공 함수 이름의 자리 표시자입니다.
phoneClose phoneClose 함수는 지정된 열린 휴대폰 디바이스를 닫습니다.
phoneConfigDialog phoneConfigDialog 함수(tapi.h)를 사용하면 지정된 디바이스의 공급자가 모달 대화 상자를 표시하여 사용자가 관련 매개 변수를 구성할 수 있습니다.
phoneConfigDialogA

phoneConfigDialog 함수를 사용하면 지정된 휴대폰 디바이스의 공급자가 사용자가 휴대폰 디바이스와 관련된 매개 변수를 구성할 수 있는 모달 대화 상자를 표시합니다. (phoneConfigDialogA)
phoneConfigDialogW phoneConfigDialogW(유니코드) 함수(tapi.h)를 사용하면 지정된 디바이스의 공급자가 모달 대화 상자를 표시하여 사용자가 관련 매개 변수를 구성할 수 있습니다.
phoneDevSpecific phoneDevSpecific 함수는 다른 TAPI 함수에 설명되지 않은 기능을 제공하기 위해 전화 통신 API 구현을 사용하도록 설정하는 일반적인 확장 메커니즘으로 사용됩니다. 이러한 확장의 의미는 디바이스에 따라 다릅니다.
phoneGetButtonInfo phoneGetButtonInfo 함수(tapi.h)는 지정된 단추에 대한 정보를 반환합니다.
phoneGetButtonInfoA phoneGetButtonInfo 함수는 지정된 단추에 대한 정보를 반환합니다. (phoneGetButtonInfoA)
phoneGetButtonInfoW phoneGetButtonInfoW(유니코드) 함수(tapi.h)는 지정된 단추에 대한 정보를 반환합니다.
phoneGetData phoneGetData 함수는 열린 휴대폰 디바이스의 지정된 위치에서 지정된 버퍼로 정보를 업로드합니다.
phoneGetDevCaps phoneGetDevCaps 함수(tapi.h)는 지정된 전화 장치를 쿼리하여 전화 통신 기능을 확인합니다.
phoneGetDevCapsA phoneGetDevCaps 함수는 지정된 전화 장치를 쿼리하여 전화 통신 기능을 확인합니다. (phoneGetDevCapsA)
phoneGetDevCapsW phoneGetDevCapsW(유니코드) 함수(tapi.h)는 지정된 전화 장치를 쿼리하여 전화 통신 기능을 확인합니다.
phoneGetDisplay phoneGetDisplay 함수는 지정된 휴대폰 디스플레이의 현재 콘텐츠를 반환합니다.

phoneGetGain phoneGetGain 함수는 지정된 휴대폰 후크위치 디바이스의 마이크 게인 설정을 반환합니다.
phoneGetHookSwitch phoneGetHookSwitch 함수는 지정된 개방형 휴대폰 디바이스의 현재 후크위치 모드를 반환합니다.
phoneGetIcon phoneGetIcon 함수(tapi.h)를 사용하면 애플리케이션이 사용자에게 표시할 수 있는 서비스 전화 디바이스별(또는 공급자별) 아이콘을 검색할 수 있습니다.
phoneGetIconA phoneGetIcon 함수를 사용하면 애플리케이션이 사용자에게 표시할 수 있는 서비스 전화 디바이스별(또는 공급자별) 아이콘을 검색할 수 있습니다. (phoneGetIconA)
phoneGetIconW phoneGetIconW(유니코드) 함수(tapi.h)를 사용하면 애플리케이션이 사용자에게 표시할 수 있는 서비스 전화 디바이스별(또는 공급자별) 아이콘을 검색할 수 있습니다.
phoneGetID phoneGetID 함수(tapi.h)는 지정된 휴대폰 디바이스와 연결된 지정된 디바이스 클래스에 대한 디바이스 식별자를 반환합니다.
phoneGetIDA phoneGetID 함수는 지정된 전화 디바이스와 연결된 지정된 디바이스 클래스에 대한 디바이스 식별자를 반환합니다. (phoneGetIDA)
phoneGetIDW phoneGetIDW(유니코드) 함수(tapi.h)는 지정된 휴대폰 디바이스와 연결된 지정된 디바이스 클래스에 대한 디바이스 식별자를 반환합니다.
phoneGetLamp phoneGetLamp 함수는 지정된 램프의 현재 램프 모드를 반환합니다.
phoneGetMessage phoneGetMessage 함수는 이벤트 핸들 알림 메커니즘을 사용하는 애플리케이션에 배달을 위해 대기 중인 다음 TAPI 메시지를 반환합니다(자세한 내용은 phoneInitializeEx 참조).
phoneGetRing

phoneGetRing 함수를 사용하면 애플리케이션이 현재 링 모드와 같이 지정된 개방형 휴대폰 디바이스를 쿼리할 수 있습니다.
phoneGetStatus phoneGetStatus 함수(tapi.h)를 사용하면 애플리케이션이 전체 상태 대해 지정된 개방형 휴대폰 디바이스를 쿼리할 수 있습니다.
phoneGetStatusA phoneGetStatus 함수를 사용하면 애플리케이션이 전체 상태 대해 지정된 개방형 휴대폰 디바이스를 쿼리할 수 있습니다. (phoneGetStatusA)
phoneGetStatusMessages phoneGetStatusMessages 함수는 지정된 휴대폰 디바이스에서 애플리케이션에 대한 콜백을 생성하는 전화 상태 변경 내용을 반환합니다.
phoneGetStatusW phoneGetStatusW(유니코드) 함수(tapi.h)를 사용하면 애플리케이션이 전체 상태 대해 지정된 개방형 휴대폰 디바이스를 쿼리할 수 있습니다.
phoneGetVolume phoneGetVolume 함수는 지정된 휴대폰의 후크위치 디바이스의 볼륨 설정을 반환합니다.
phoneInitialize phoneInitialize 함수는 사용되지 않습니다. TAPI 버전 1.3 및 1.4를 사용하는 애플리케이션과의 이전 버전과의 호환성을 위해 Tapi.dll 및 Tapi32.dll 계속 내보냅니다.
phoneInitializeExA phoneInitializeEx 함수는 휴대폰 추상화의 후속 사용을 위해 애플리케이션의 TAPI 사용을 초기화합니다. (ANSI)
phoneInitializeExW phoneInitializeEx 함수는 휴대폰 추상화의 후속 사용을 위해 애플리케이션의 TAPI 사용을 초기화합니다. (유니코드)
phoneNegotiateAPIVersion phoneNegotiateAPIVersion을 사용하면 애플리케이션이 지정된 전화 디바이스에 사용할 API 버전을 협상할 수 있습니다.
phoneNegotiateExtVersion phoneNegotiateExtVersion 함수를 사용하면 애플리케이션이 지정된 휴대폰 디바이스에서 사용할

확장 버전을 협상할 수 있습니다. 애플리케이션이 확장을 지원하지 않는 경우 이 작업을 호출할 필요가 없습니다.
phoneOpen phoneOpen 함수는 지정된 휴대폰 디바이스를 엽니다.
phoneSetButtonInfo phoneSetButtonInfo 함수(tapi.h)는 지정된 휴대폰의 지정된 단추에 대한 정보를 설정합니다.
phoneSetButtonInfoA phoneSetButtonInfo 함수는 지정된 휴대폰의 지정된 단추에 대한 정보를 설정합니다. (phoneSetButtonInfoA)
phoneSetButtonInfoW phoneSetButtonInfoW(유니코드) 함수(tapi.h)는 지정된 휴대폰의 지정된 단추에 대한 정보를 설정합니다.
phoneSetData phoneSetData 함수는 지정된 버퍼의 정보를 선택한 데이터 식별자의 열린 휴대폰 디바이스에 다운로드합니다.
phoneSetDisplay phoneSetDisplay 함수를 사용하면 지정된 열린 휴대폰 디바이스에 지정된 문자열이 표시됩니다.
phoneSetGain phoneSetGain 함수는 지정된 후크위치 디바이스의 마이크 게인을 지정된 게인 수준으로 설정합니다.
phoneSetHookSwitch phoneSetHookSwitch 함수는 지정된 열린 휴대폰의 후크위치 디바이스의 후크 상태를 지정된 모드로 설정합니다. 나열된 후크위치 디바이스의 hookswitch 상태만 영향을 받습니다.
phoneSetLamp phoneSetLamp 함수를 사용하면 지정된 램프 모드에서 지정된 열린 휴대폰 디바이스에서 지정된 램프가 켜집니다.
phoneSetRing phoneSetRing 함수는 지정된 링 모드 및 볼륨을 사용하여 지정된 열린 휴대폰 디바이스를 울립니다.

phoneSetStatusMessages phoneSetStatusMessages 함수를 사용하면 애플리케이션이 선택한 상태 이벤트에 대해 지정된 휴대폰 디바이스를 모니터링할 수 있습니다.
phoneSetVolume phoneSetVolume 함수는 지정된 후크위치 디바이스의 스피커 구성 요소 볼륨을 지정된 수준으로 설정합니다.
phoneShutdown phoneShutdown 함수는 TAPI의 휴대폰 추상화에 대한 애플리케이션의 사용을 종료합니다.
픽업 Pickup 메서드는 지정된 그룹 식별에서 호출 경고를 선택합니다.
PostEvent POSTEvent 메서드는 MSPCall에 의해 호출되어 이벤트를 TAPI3에 게시합니다. 이 메서드는 이벤트 목록의 끝에 이벤트를 두고 TAPI3 신호를 보냅니다. 이벤트 목록을 잠가 입니다.
ProcessGraphEvent MSPCall.h에 정의된 ProcessGraphEvent 메서드는 MSP 작업자 스레드에서 호출됩니다.
ProcessGraphEvent ProcessGraphEvent 메서드는 스트림이 그래프 이벤트를 처리할 수 있도록 MSPCall 개체에 의해 호출됩니다.
put_AdvertisingScope put_AdvertisingScope 메서드는 광고 scope 설정합니다.
put_AppSpecific put_AppSpecific 메서드는 감지할 톤을 식별하는 애플리케이션 정의 태그를 설정합니다.
put_AudioFormatForScripting put_AudioFormatForScripting 메서드는 오디오 스크립팅 형식을 설정합니다.
put_AutoDialtone put_AutoDialtone 메서드는 AutoDialtone 속성의 값을 설정합니다.

put_AutoEndOfNumberTimeout put_AutoEndOfNumberTimeout 메서드는 AutoEndOfNumberTimeout 속성의 값을 설정합니다. 속성은 전체 숫자가 수집되었다고 가정하기 전에 숫자를 누른 후 대기하는 시간을 지정합니다.
put_AutoKeypadTones put_AutoKeypadTones 메서드는 이 휴대폰의 AutoKeypadTones 속성을 설정합니다. 이 기능을 사용하도록 설정하면 키패드 단추를 누를 때마다 숫자 톤이 자동으로 재생됩니다.
put_AutoKeypadTonesMinimumDuration put_AutoKeypadTonesMinimumDuration 메서드는 AutoKeypadTonesMinimumDuration 속성의 값을 설정합니다. 속성은 PBS_DOWN 키패드 톤을 재생할 기간을 지정합니다.
put_AutoStopRingOnOffHook put_AutoStopRingOnOffHook 메서드는 AutoStopRingOnOffHook 속성을 설정합니다. 이 기능을 사용하도록 설정하면 휴대폰이 꺼져 있으면 휴대폰에서 생성된 수신 링이 종료됩니다 (ITAutomatedPhoneControl::StopRinger에 대한 호출을 통해).
put_AutoStopTonesOnOnHook put_AutoStopTonesOnOnHook 메서드는 이 휴대폰의 AutoStopTonesOnOnHook 속성을 설정합니다.
put_AutoVolumeControl put_AutoVolumeControl 메서드는 이 휴대폰의 AutoVolumeControl 속성을 설정합니다.
put_AutoVolumeControlRepeatDelay put_AutoVolumeControlRepeatDelay 메서드는 AutoVolumeControlRepeatDelay 속성을 설정합니다. 속성은 볼륨 단추가 누른 상태에서 반복되기 전에 지연 시간(밀리초)을 지정합니다.
put_AutoVolumeControlRepeatPeriod put_AutoVolumeControlRepeatPeriod 메서드는 AutoVolumeControlRepeatPeriod 속성을 설정합니다. 속성은 볼륨 단추를 누를 때 단추가 반복되는 기간을 밀리초(밀리초)로 제어합니다.
put_AutoVolumeControlStep put_AutoVolumeControlStep 메서드는 AutoVolumeControlStep 속성을 설정합니다. 속성은 볼륨 단추를 누를 때 휴대폰 볼륨이 조정되는 양을 결정합니다.
put_AvgBytesPerSec put_AvgBytesPerSec 메서드는 WAVEFORMATEX 구조체에서 nAvgBytesPerSec 멤버를 설정합니다.

put_Balance put_Balance 메서드는 잔액을 설정합니다. 이 메서드는 TAPI 3.0 이상과 함께 제공되는 터미널에 대해 구현되지 않습니다.
put_BitsPerSample put_BitsPerSample 메서드는 WAVEFORMATEX 구조체에서 wBitsPerSample 멤버를 설정합니다.
put_BlockAlign put_BlockAlign 메서드는 WAVEFORMATEX 구조체에서 nBlockAlign 멤버를 설정합니다.
put_ButtonFunction put_ButtonFunction 메서드는 단추 함수를 설정합니다.
put_ButtonMode put_ButtonMode 메서드는 단추 모드를 설정합니다.
put_ButtonText put_ButtonText 메서드는 단추 텍스트를 설정합니다.
put_CadenceOff put_CadenceOff 메서드는 생성할 사용자 지정 톤 주기의 "off" 기간을 설정합니다.
put_CadenceOn put_CadenceOn 메서드는 생성할 사용자 지정 톤 주기의 "켜기" 기간을 설정합니다.
put_CallInfoBuffer put_CallInfoBuffer 메서드는 사용자-사용자 정보와 같이 버퍼가 필요한 호출 정보 항목을 설정합니다.
put_CallInfoLong put_CallInfoLong 메서드는 전달자 모드와 같이 long에서 설명하는 호출 정보 항목을 설정합니다.
put_CallInfoString put_CallInfoString 메서드는 표시 가능한 주소와 같이 문자열에서 설명하는 호출 정보 항목을 설정합니다.
put_Channels

put_Channels 메서드는 WAVEFORMATEX 구조체에서 nChannels 멤버를 설정합니다.
put_CLSID put_CLSID 메서드는 터미널을 CoCreateInstance하는 데 사용되는 CLSID를 설정합니다. (ITPluggableTerminalClassRegistration.put_CLSID)
put_CLSID put_CLSID 메서드는 터미널을 CoCreateInstance하는 데 사용되는 CLSID를 설정합니다. (ITPluggableTerminalSuperclassRegistration.put_CLSID)
put_Company put_Company 메서드는 이 플러그형 터미널을 발급한 회사의 이름을 설정합니다.
put_DefaultObjectTTL put_DefaultObjectTTL 메서드는 만든 개체에 대한 TTL(기본 TTL) 값을 초 단위로 설정합니다. 동적 서버에만 적용됩니다. 최소값은 300초입니다.
put_Description put_Description 메서드는 회의 설명을 설정합니다.
put_Direction put_Direction 메서드는 터미널에서 지원하는 방향을 설정합니다.
put_DoNotDisturb put_DoNotDisturb 메서드는 방해 금지 상태 설정합니다. 방해 금지 기능은 모든 주소에서 사용할 수 없습니다.
put_Duration put_Duration 메서드는 TAPI Server에서 톤 이벤트를 생성하기 전에 톤이 있어야 하는 기간을 설정합니다.
put_EventFilter put_EventFilter 메서드는 현재 주소에 대한 이벤트 필터를 설정합니다. 이벤트 필터가 설정되지 않은 경우 애플리케이션은 이 주소의 이벤트에 대한 알림을 받지 않습니다.
put_EventFilter put_EventFilter 메서드는 현재 호출에 대한 이벤트 필터를 설정합니다.
put_EventFilter

put_EventFilter 메서드는 이벤트 필터 마스크를 설정합니다. 마스크는 TAPI_EVENT 열거형의 일련의 ORed 멤버입니다.
put_FileName put_FileName 메서드는 기록할 파일의 이름을 설정합니다.
put_Format put_Format 메서드는 트랙의 형식 형식을 설정합니다.
put_FormatTag put_FormatTag 메서드는 WAVEFORMATEX 구조체에서 wFormatTag 멤버를 설정합니다.
put_Frequency put_Frequency 메서드는 생성할 톤 구성 요소의 빈도를 설정합니다.
put_Frequency put_Frequency 메서드는 TAPI Server에서 톤 이벤트를 생성해야 하는 톤 빈도를 설정합니다.
put_HookSwitchState put_HookSwitchState 메서드는 휴대폰의 특정 후크위치 디바이스에 대한 현재 후크위치 상태를 설정합니다.
put_IPPhonePrimary put_IPPhonePrimary 메서드는 컴퓨터의 이름을 사용자의 기본 IP 휴대폰으로 설정합니다.
put_IsEncrypted put_IsEncrypted 메서드는 회의 암호화 여부를 설정합니다.
put_LampMode put_LampMode 메서드는 지정된 램프의 현재 램프 모드를 설정합니다.
put_LeaseStartTime put_LeaseStartTime 메서드는 임대의 시작 시간을 설정합니다. 이 메서드는 put_LeaseStopTime 함께 IMcastAddressAllocation::CreateLeaseInfo를 호출하지 않고 임대를 갱신할 수 있습니다.
put_LeaseStopTime put_LeaseStopTime 메서드는 임대의 중지 시간을 설정합니다. 이 메서드는 put_LeaseStartTime 함께 IMcastAddressAllocation::CreateLeaseInfo를 호출하지 않고 임대를 갱신할 수 있습니다.

put_MeasurementPeriod put_MeasurementPeriod 메서드(tapi3.h)는 스위치 및/또는 구현이 저장하고 정보를 계산하는 기간(초)을 설정합니다.
put_MeasurementPeriod ITQueue::p ut_MeasurementPeriod(tapi3.h) 메서드는 스위치 및/또는 구현이 정보를 저장하고 계산하는 마침표(초)를 설정합니다.
put_MeasurementPeriod ITAgent::p ut_MeasurementPeriod 메서드(tapi3cc.h)는 스위치 및/또는 구현이 정보를 저장하고 계산하는 마침표(초)를 설정합니다.
put_MeasurementPeriod ITQueue::p ut_MeasurementPeriod 메서드(tapi3cc.h)는 스위치 및/또는 구현이 정보를 저장하고 계산하는 마침표(초)를 설정합니다.
put_MediaFormat ITAMMediaFormat::p ut_MediaFormat(tapi3.h) 메서드는 미디어 형식을 설정합니다.
put_MediaFormat ITAMMediaFormat::p ut_MediaFormat 메서드(tapi3ds.h)는 미디어 형식을 설정합니다.
put_MediaTypes put_MediaTypes 메서드는 터미널에서 지원하는 미디어 형식을 설정합니다.
put_MessageWaiting put_MessageWaiting 메서드는 주소에서 대기 중인 메시지의 상태 설정합니다.
put_Name put_Name 메서드는 등록 중인 터미널 클래스의 이름을 설정합니다.
put_Name put_Name 메서드는 터미널 슈퍼클래스에 대한 이름을 설정합니다.
put_Name put_Name 메서드는 디렉터리 개체의 이름을 설정합니다.

put_NumRingsNoAnswer put_NumRingsNoAnswer 메서드는 응답 없음 조건이 가정된 링 수를 설정합니다.
put_Originator put_Originator 메서드는 보낸 사람의 사용자 이름을 설정합니다.
put_PhoneHandlingEnabled put_PhoneHandlingEnabled 메서드는 PhoneHandlingEnabled 속성을 설정합니다.
put_PlayList put_PlayList 메서드는 재생할 파일 목록이 있는 파일 재생 터미널을 제공합니다.
put_Port put_Port 메서드는 지정된 ILS 디렉터리의 서버에 연결하는 데 사용되는 포트 번호를 설정합니다.
put_RingMode put_RingMode 메서드는 휴대폰이 링 모드를 변경해 줄 것을 요청합니다.
put_RingVolume put_RingVolume 메서드는 휴대폰이 벨소리 볼륨을 변경해줄 것을 요청합니다.
put_SamplesPerSec put_SamplesPerSec 메서드는 WAVEFORMATEX 구조체에서 nSamplesPerSec 멤버를 설정합니다.
put_SecurityDescriptor put_SecurityDescriptor 메서드는 현재 보안 권한을 설명하는 디렉터리 서비스 보안 설명자 개체에 대한 IDispatch 포인터를 설정합니다.
put_StartTime put_StartTime 메서드는 회의의 시작 시간을 설정합니다.
put_State put_State 메서드(tapi3.h)는 에이전트 세션의 상태를 설정합니다.
put_State ITAgentSession::p ut_State(tapi3.h) 메서드는 에이전트 세션의 상태를 설정합니다.

put_State ITAgent::p ut_State 메서드(tapi3cc.h)는 에이전트 세션의 상태를 설정합니다.
put_State ITAgentSession::p ut_State 메서드(tapi3cc.h)는 에이전트 세션의 상태를 설정합니다.
put_StopTime put_StopTime 메서드는 회의 중지 시간을 설정합니다. 종료 시간이 0이면 세션이 제한되지 않습니다.
put_TerminalClass put_TerminalClass 메서드는 터미널의 터미널 클래스를 설정합니다.
put_Url put_Url 메서드는 URL을 설정합니다.
put_Version put_Version 메서드는 터미널 버전을 설정합니다.
put_Volume put_Volume 메서드는 볼륨을 설정합니다.
put_Volume put_Volume 메서드는 톤을 생성할 볼륨 수준을 설정합니다.
QueryDispatchInterface QueryDispatchInterface 메서드는 해당 GUID 및 개체에 있는 다른 인터페이스의 디스패치 포인터가 지정된 개체의 다른 인터페이스에 대한 디스패치 포인터를 반환합니다.
QueryMediaType QueryMediaType 메서드는 현재 주소와 연결된 서비스 공급자가 IMediaType으로 표시된 미디어 형식 또는 형식을 지원하는지 여부를 나타냅니다.
ReceiveTSPAddressData ReceiveTSPAddressData 메서드는 TSP 데이터 메시지가 특정 호출이 아닌 주소로 처리될 때 호출됩니다. MSP는 주소별 TSP 메시지를 처리하려는 경우 이 메서드를 재정의해야 합니다.

ReceiveTSPCallData ReceiveTSPCallData 메서드는 MSP 주소 개체의 ReceiveTSPData 메서드에 의해 호출되어 TSP 데이터를 올바른 호출로 디스패치합니다.
ReceiveTSPData TSP가 MSP에 비동기 데이터를 보낼 때 TAPI 3에서 ITMSPAddress::ReceiveTSPData(tapi3.h) 메서드를 호출합니다.
ReceiveTSPData TSP가 MSP에 비동기 데이터를 보낼 때 TAPI 3에서 ITMSPAddress::ReceiveTSPData(msp.h) 메서드를 호출합니다.
RefreshDirectoryObject RefreshDirectoryObject 메서드는 서버의 개체에 대한 TTL(Time to Live)을 새로 고칩니다. 동적 서버에만 적용됩니다.
RegisterCallNotifications RegisterCallNotifications 메서드는 애플리케이션이 수신할 새 호출 알림을 설정합니다. 애플리케이션은 각 주소에 대해 메서드를 호출하여 처리할 수 있는 미디어 유형 또는 형식을 나타내고 요청하는 권한을 지정해야 합니다.
RegisterRequestRecipient RegisterRequestRecipient 메서드는 애플리케이션 instance 보조 전화 통신 요청을 처리하기에 적합한 것으로 등록합니다.
RegisterSink ITPluggableTerminalEventSinkRegistration::RegisterSink(tapi3.h) 메서드는 플러그형 터미널 이벤트 알림에 대한 애플리케이션을 등록합니다.
RegisterSink ITPluggableTerminalEventSinkRegistration::RegisterSink(msp.h) 메서드는 플러그형 터미널 이벤트 알림에 대한 애플리케이션을 등록합니다.
RegisterWaitEvent RegisterWaitEvent 메서드는 호출 개체의 중요한 섹션 내에서만 호출해야 합니다.
ReleaseAddress ReleaseAddress 메서드는 이전에 가져온 임대를 해제합니다.

ReleaseUserUserInfo ReleaseUserUserInfo 메서드는 애플리케이션이 ITCallInfo::GetCallInfoBuffer 메서드에서 가져온 사용자-사용자 정보를 처리했음을 서비스 공급자에게 알릴 수 있습니다.
제거 Remove 메서드는 지정된 인덱스의 컬렉션에서 항목을 삭제합니다.
RemoveFromConference RemoveFromConference 메서드는 전화 회의와 관련된 경우 전화 회의에서 호출을 제거합니다.
RemoveStream RemoveStream 메서드는 미디어 스트림을 제거합니다.
RemoveStream (Interface RemoveStream) RemoveStream 메서드는 호출에서 스트림을 제거하기 위해 애플리케이션에서 호출됩니다. (CMSPCallBase.RemoveStream)
RemoveStream (Interface RemoveStream) RemoveStream 메서드는 호출에서 스트림을 제거하기 위해 애플리케이션에서 호출됩니다. (CMSPCallMultiGraph.RemoveStream)
RemoveSubStream RemoveSubStream 메서드는 하위 스트림을 제거합니다.
RemoveTrackTerminal RemoveTrackTerminal 메서드는 메서드가 호출된 멀티트랙 터미널에 속하는 트랙 터미널 컬렉션에서 지정된 터미널을 제거합니다.
RenewAddress RenewAddress 메서드는 주소 임대를 갱신합니다. CreateLeaseInfo를 호출하여 갱신 요청의 매개 변수를 지정한 다음, 이 메서드를 호출하여 요청을 수행합니다.
RequestAddress RequestAddress 메서드는 하나 이상의 멀티캐스트 주소에 대한 새 임대를 가져옵니다. EnumerateScopes 또는 get_Scopes 메서드를 먼저 호출해야 합니다.
RequestTerminal

RequestTerminal 메서드는 필요한 클래스, 미디어 및 방향을 고려하여 적합한 터미널을 가져옵니다.
Reset IEnumACDGroup::Reset 메서드(tapi3.h)는 열거형 시퀀스를 시작 부분으로 다시 설정합니다.
Reset IEnumAgent::Reset 메서드(tapi3.h)는 열거형 시퀀스를 시작 부분으로 다시 설정합니다.
Reset IEnumAgentHandler::Reset 메서드(tapi3.h)는 열거형 시퀀스를 시작 부분으로 다시 설정합니다.
Reset IEnumAgentSession::Reset 메서드(tapi3.h)는 열거형 시퀀스를 시작 부분으로 다시 설정합니다.
Reset IEnumQueue::Reset 메서드(tapi3.h)는 열거형 시퀀스를 시작 부분으로 다시 설정합니다.
Reset Reset 메서드는 열거형 시퀀스의 시작 부분으로 다시 설정됩니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumAddress.Reset)
Reset Reset 메서드는 열거형 시퀀스의 시작 부분으로 다시 설정됩니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumBstr.Reset)
Reset Reset 메서드는 열거형 시퀀스의 시작 부분으로 다시 설정됩니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumCall.Reset)
Reset Reset 메서드는 열거형 시퀀스의 시작 부분으로 다시 설정됩니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumCallHub.Reset)
Reset Reset 메서드는 열거형 시퀀스의 시작 부분으로 다시 설정됩니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumCallingCard.Reset)
Reset

Reset 메서드는 열거형 시퀀스의 시작 부분으로 다시 설정됩니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumLocation.Reset)
Reset Reset 메서드는 열거형 시퀀스를 시작 부분으로 다시 설정합니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumPhone.Reset)
Reset Reset 메서드는 열거형 시퀀스를 시작 부분으로 다시 설정합니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumPluggableSuperclassInfo.Reset)
Reset Reset 메서드는 열거형 시퀀스를 시작 부분으로 다시 설정합니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumPluggableTerminalClassInfo.Reset)
Reset Reset 메서드는 열거형 시퀀스의 시작 부분으로 다시 설정됩니다. (IEnumStream.Reset)
Reset Reset 메서드는 열거형 시퀀스의 시작 부분으로 다시 설정됩니다. (IEnumSubStream.Reset)
Reset Reset 메서드는 열거형 시퀀스의 시작 부분으로 다시 설정됩니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumTerminal.Reset)
Reset Reset 메서드는 열거형 시퀀스의 시작 부분으로 다시 설정됩니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumTerminalClass.Reset)
Reset IEnumACDGroup::Reset 메서드(tapi3cc.h)는 열거형 시퀀스를 시작 부분으로 다시 설정합니다.
Reset IEnumAgentHandler::Reset 메서드(tapi3cc.h)는 열거형 시퀀스를 시작 부분으로 다시 설정합니다.
Reset IEnumAgentHandler::Next 메서드(tapi3cc.h)는 열거형 시퀀스를 시작 부분으로 다시 설정합니다.
Reset

IEnumAgentSession::Reset 메서드(tapi3cc.h)는 열거형 시퀀스를 시작 부분으로 다시 설정합니다.
Reset IEnumQueue::Reset 메서드(tapi3cc.h)는 열거형 시퀀스를 시작 부분으로 다시 설정합니다.
Reset Reset 메서드는 열거형 시퀀스의 시작 부분으로 다시 설정됩니다. (IEnumDialableAddrs.Reset)
Reset Reset 메서드는 열거형 시퀀스의 시작 부분으로 다시 설정됩니다. (IEnumDirectory.Reset)
Reset Reset 메서드는 열거형 시퀀스의 시작 부분으로 다시 설정됩니다. (IEnumDirectoryObject.Reset)
Reset Reset 메서드는 열거형 시퀀스의 시작 부분으로 다시 설정됩니다. (IEnumMcastScope.Reset)
RunRenderFilter RunRenderFilter 메서드는 터미널에서 가장 오른쪽 렌더링 필터를 시작합니다. 동적 필터 그래프에 필요합니다.
SelectCall SelectCall 메서드는 pCall 매개 변수가 가리키는 Call 개체에 현재 전화 개체를 선택합니다.
선택터미널 SelectTerminal 메서드는 스트림에 대한 ITTerminal 개체를 선택합니다.
선택터미널 SelectTerminal 메서드는 하위 스트림에 대한 ITTerminal 개체를 선택합니다. 자세한 내용은 ITStream::SelectTerminal 아래의 설명 섹션을 참조하세요.
SelectTerminalOnCall SelectTerminalOnCall 메서드는 호출에 대한 터미널을 선택합니다.
SetAllocateBuffers ITAllocatorProperties::SetAllocateBuffers(tapi3.h) 메서드는 현재 할당자 버퍼를 설정해야 하는지 여부를 결정합니다.

SetAllocateBuffers ITAllocatorProperties::SetAllocateBuffers 메서드(tapi3ds.h)는 현재 할당자 버퍼를 설정해야 하는지 여부를 결정합니다.
SetAllocatorProperties ITAllocatorProperties::SetAllocatorProperties(tapi3.h) 메서드는 연결 전에 호출되어야 하며 필터 협상 중에 MSP가 이러한 값을 사용하도록 강제합니다.
SetAllocatorProperties ITAllocatorProperties::SetAllocatorProperties 메서드(tapi3ds.h)는 필터 협상 중에 MSP가 입력한 값을 사용하도록 강제합니다.
SetApplicationPriority SetApplicationPriority 메서드를 사용하면 애플리케이션이 특정 미디어 유형 또는 보조 전화 통신 요청 모드의 핸드오프 우선 순위 목록에서 우선 순위를 설정하거나 우선 순위 목록에서 자신을 제거할 수 있습니다.
SetAssistedTelephonyPriority SetAssistedTelephonyPriority 메서드는 지원되는 전화 통신 요청을 처리하도록 애플리케이션 우선 순위를 설정합니다.
SetBufferSize ITAllocatorProperties::SetBufferSize(tapi3.h) 메서드는 할당자 버퍼의 크기를 설정합니다.
SetBufferSize ITAllocatorProperties::SetBufferSize 메서드(tapi3ds.h)는 할당자 버퍼의 크기를 설정합니다.
SetCallHubTracking SetCallHubTracking 메서드는 CallHub 추적을 사용하거나 사용하지 않도록 설정합니다.
SetCallInfoBuffer SetCallInfoBuffer 메서드는 사용자-사용자 정보와 같이 버퍼가 필요한 호출 정보 항목을 설정합니다. Visual Basic으로 작성된 애플리케이션과 같은 Automation 클라이언트 애플리케이션은 ITCallInfo::p ut_CallInfoBuffer 메서드를 사용해야 합니다.
SetDevConfig SetDevConfig 함수를 사용하면 애플리케이션이 회선 디바이스의 미디어 스트림 디바이스 구성을 GetDevConfig를 사용하여 이전에 가져온 설정으로 복원할 수 있습니다.

SetDisplay SetDisplay 메서드는 휴대폰 디스플레이의 지정된 행과 열에 표시되는 내용을 설정합니다.
SetForwardType SetForwardType 메서드는 전달 모드와 대상을 호출자 주소별로 설정합니다.
SetForwardType2 SetForwardType2 메서드는 호출자 주소로 지정된 현재 전달 모드를 설정합니다.
SetMediaType SetMediaType 메서드는 LINECALLINFO 구조에서 현재 호출에 대한 미디어 형식을 설정합니다. 자세한 내용은 lineSetMediaMode를 참조하세요.
SetQOS SetQOS 메서드는 호출에 대한 서비스 수준의 품질을 설정합니다.
종료 MSP가 언로드될 때 ITMSPAddress::Shutdown(tapi3.h) 메서드가 호출됩니다. 종료는 주소 개체당 한 번 호출됩니다.
종료 Shutdown 메서드는 TAPI 세션을 종료합니다.
종료 MSP가 언로드될 때 ITMSPAddress::Shutdown(msp.h) 메서드가 호출됩니다. 종료는 주소 개체당 한 번 호출됩니다.
종료 ShutDown 메서드는 호출을 종료하기 위해 MSPAddress 개체(ShutdownMSPCall 메서드)에 의해 호출됩니다.
종료 ShutDown 메서드는 MSP 주소 개체(ShutdownMSPCall 메서드)에서 호출되어 MSP 호출 개체를 종료합니다.
종료 ShutDown 메서드는 MSPCall 개체에 의해 호출됩니다. UnselectTerminal을 통해 모든 터미널 개체

의 선택을 취소합니다. 또한 호출 개체에서 MSPCallRelease를 호출합니다. 순환 리카운트 중단에 필요합니다.
ShutdownMSPCall 호출 개체가 제거될 때 ITMSPAddress::ShutdownMSPCall(tapi3.h) 메서드가 호출됩니다.
ShutdownMSPCall 호출 개체가 제거될 때 ITMSPAddress::ShutdownMSPCall(msp.h) 메서드가 호출됩니다.
ShutdownMSPCallHelper ShutdownMSPCallHelper 도우미 템플릿 함수는 파생 클래스의 ShutdownMSPCall 구현에서 호출됩니다.
Skip IEnumACDGroup::Skip 메서드(tapi3.h)는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다.
Skip IEnumAgent::Skip 메서드(tapi3.h)는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다.
Skip IEnumAgentHandler::Skip 메서드(tapi3.h)는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다.
Skip IEnumAgentSession::Skip 메서드(tapi3.h)는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다.
Skip IEnumQueue::Skip 메서드(tapi3.h)는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다.
Skip Skip 메서드는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumAddress.Skip)
Skip Skip 메서드는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumBstr.Skip)

Skip Skip 메서드는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumCall.Skip)
Skip Skip 메서드는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumCallHub.Skip)
Skip Skip 메서드는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumCallingCard.Skip)
Skip Skip 메서드는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumLocation.Skip)
Skip Skip 메서드는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumPhone.Skip)
Skip Skip 메서드는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumPluggableSuperclassInfo.Skip)
Skip Skip 메서드는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumPluggableTerminalClassInfo.Skip)
Skip Skip 메서드는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다. (IEnumStream.Skip)
Skip Skip 메서드는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다. (IEnumSubStream.Skip)
Skip Skip 메서드는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumTerminal.Skip)

Skip Skip 메서드는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다. 이 메서드는 Visual Basic 및 스크립팅 언어에서 숨겨집니다. (IEnumTerminalClass.Skip)
Skip IEnumACDGroup::Skip 메서드(tapi3cc.h)는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다.
Skip IEnumAgent::Skip 메서드(tapi3cc.h)는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다.
Skip IEnumAgentHandler::Skip 메서드(tapi3cc.h)는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다.
Skip IEnumAgentSession 인터페이스(tapi3cc.h)는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다.
Skip IEnumQueue::Skip 메서드(tapi3cc.h)는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다.
Skip Skip 메서드는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다. (IEnumDialableAddrs.Skip)
Skip Skip 메서드는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다. (IEnumDirectory.Skip)
Skip Skip 메서드는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다. (IEnumDirectoryObject.Skip)
Skip Skip 메서드는 열거형 시퀀스에서 지정된 다음 개수의 요소를 건너뜁니다. (IEnumMcastScope.Skip)
시작

Start 메서드는 현재 위치에서 작업을 시작합니다.
StartRinger StartRinger 메서드는 링 모드와 링 기간을 지정하여 휴대폰의 벨소리를 시작합니다.
StartStream StartStream 메서드는 스트림을 시작합니다.
StartSubStream StartSubStream 메서드는 하위 스트림을 시작합니다. 자세한 내용은 ITStream::StartStream 아래의 설명 섹션을 참조하세요.
StartTone StartTone 메서드는 컨트롤 톤을 보냅니다.
중지 Stop 메서드는 현재 작업을 중지하고 현재 위치를 파일의 시작 부분으로 설정합니다.
StopRenderFilter StopRenderFilter 메서드는 터미널에서 가장 오른쪽 렌더링 필터를 중지합니다. 동적 필터 그래프에 필요합니다.
StopRinger StopRinger 메서드는 현재 생성되는 들어오는 링을 중단합니다. 링이 생성되지 않으면 메서드는 아무 작업도 수행하지 않고 성공을 반환합니다.
StopStream StopStream 메서드는 스트림을 중지합니다.
StopSubStream StopSubStream 메서드는 하위 스트림을 중지합니다. 자세한 내용은 ITStream::StopStream을 참조하세요.
StopTone StopTone 메서드는 현재 재생 중인 모든 톤을 중단합니다. 톤이 재생되지 않으면 메서드는 아무 작업도 수행하지 않고 성공을 반환합니다.
SwapHold

SwapHold 메서드는 호출(활성 상태)을 지정된 통화 보류 상태로 교환합니다.
TAPIERROR_FORMATMESSAGE TAPIERROR_FORMATMESSAGE 매크로는 FormatMessage 함수에서 사용할 수 있는 표준 TAPI 오류 코드에 대한 식별자를 생성합니다.
tapiGetLocationInfo tapiGetLocationInfo 함수(tapi.h)는 사용자가 전화 통신 제어판 위치 매개 변수에 설정한 국가, 지역 및 도시 코드를 반환합니다.
tapiGetLocationInfoA tapiGetLocationInfo 함수는 사용자가 전화 통신 제어판 현재 위치 매개 변수에 설정한 국가 또는 지역 코드 및 도시(영역) 코드를 반환합니다. (tapiGetLocationInfoA)
tapiGetLocationInfoW tapiGetLocationInfoW(유니코드) 함수(tapi.h)는 사용자가 전화 통신 제어판 위치 매개 변수에 설정한 국가, 지역 및 도시 코드를 반환합니다.
tapiRequestDrop tapiRequestMediaCall에 대한 이전 호출에서 수행한 호출 요청을 닫습니다.
tapiRequestMakeCall tapiRequestMakeCall 함수(tapi.h)는 음성 통화의 설정을 요청합니다.
tapiRequestMakeCallA tapiRequestMakeCall 함수는 음성 통화의 설정을 요청합니다. 호출 관리자 애플리케이션은 요청 애플리케이션을 대신하여 호출을 설정한 다음, 사용자의 call-manager 애플리케이션에 의해 제어됩니다. (tapiRequestMakeCallA)
tapiRequestMakeCallW tapiRequestMakeCallW(유니코드) 함수(tapi.h)는 음성 통화의 설정을 요청합니다.
전달 Transfer 메서드는 현재 호출을 대상 주소로 전송합니다.
TranslateAddress TranslateAddress 메서드는 주소 번역 정보 인터페이스를 만듭니다.
TranslateDialog

TranslateDialog 메서드는 사용자가 전화를 걸 전화 번호의 현재 위치를 변경하고, 위치를 조정하고, 카드 매개 변수를 호출하고, 효과를 볼 수 있는 애플리케이션 모달 대화 상자를 표시합니다.
언파크 Unpark 메서드는 park에서 호출을 가져옵니다.
UnregisterNotifications UnregisterNotifications 메서드는 ITTAPI::RegisterCallNotifications를 사용하여 수행된 들어오는 호출 알림 등록을 제거합니다.
UnregisterSink ITPluggableTerminalEventSinkRegistration::UnregisterSink(tapi3.h) 메서드는 플러그형 터미널 이벤트에 대한 이벤트 등록을 지웁니다.
UnregisterSink ITPluggableTerminalEventSinkRegistration::UnregisterSink(msp.h) 메서드는 플러그형 터미널 이벤트에 대한 이벤트 등록을 지웁니다.
UnregisterWaitEvent UnregisterWaitEvent 메서드는 UnregisterWait 함수를 호출하여 스레드 풀에 지정된 인덱스에서 대기 블록으로 표시된 핸들에 대한 대기를 중지하도록 지시합니다.
UnselectCall UnselectCall 메서드는 이 전화 개체에서 지정된 통화를 제거하여 통화 개체에 대한 전화 개체의 참조를 해제합니다.
UnselectTerminal UnselectTerminal 메서드는 스트림에서 터미널을 선택 취소하고 이 스트림에 대한 스트리밍을 중지합니다.
UnselectTerminal UnselectTerminal 메서드는 하위 스트림에서 터미널을 선택 취소합니다.
UnselectTerminalOnCall UnselectTerminalOnCall 메서드는 호출에서 터미널을 선택 취소합니다.
UpdateTerminalList UpdateTerminalList 메서드는 MSP의 정적 터미널 목록을 채웁니다.

인터페이스

IEnumACDGroup IEnumACDGroup(tapi3.h) 인터페이스는 ITACDGroup 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다.
IEnumACDGroup IEnumACDGroup 인터페이스(tapi3cc.h)는 ITACDGroup 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다.
IEnumAddress IEnumAddress 인터페이스는 ITAddress 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다. ITTAPI::EnumerateAddresses 및 ITAgentHandler::EnumerateUsableAddresses 메서드는 IEnumAddress에 대한 포인터를 반환합니다.
IEnumAgent IEnumAgent(tapi3.h)는 ITAgent에 대한 열거자 인터페이스입니다.
IEnumAgent IEnumAgent(tapi3cc.h)는 ITAgent의 열거자 인터페이스입니다.
IEnumAgentHandler IEnumAgentHandler(tapi3.h) 인터페이스는 ITAgentHandler 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다.
IEnumAgentHandler IEnumAgentHandler 인터페이스(tapi3cc.h)는 ITAgentHandler 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다.
IEnumAgentSession IEnumAgentSession(tapi3.h) 인터페이스는 ITAgentSession 인터페이스에 대한 COM 표준 열거형 메서드를 제공합니다.
IEnumAgentSession IEnumAgentSession 인터페이스(tapi3cc.h)는 ITAgentSession 인터페이스에 대한 COM 표준 열거형 메서드를 제공합니다.
IEnumBstr

IEnumBstr 인터페이스는 BSTR 문자열을 열거하는 COM 표준 메서드를 제공합니다.
IEnumCall IEnumCall 인터페이스는 ITCallInfo 인터페이스에 대한 COM 표준 열거형 메서드를 제공합니다. ITCallHub::EnumerateCalls 및 ITAddress::EnumerateCalls 메서드는 IEnumCall에 대한 포인터를 반환합니다.
IEnumCallHub IEnumCallHub 인터페이스는 ITCallHub 인터페이스에 대한 COM 표준 열거형 메서드를 제공합니다. ITTAPI::EnumerateCallHubs 메서드는 IEnumCallHub에 대한 포인터를 반환합니다.
IEnumCallingCard IEnumCallingCard 인터페이스는 ITCallingCard 인터페이스에 대한 COM 표준 열거형 메서드를 제공합니다. ITAddressTranslation::EnumerateCallingCards 메서드는 IEnumCallingCard에 대한 포인터를 반환합니다.
IEnumDialableAddrs IEnumDialableAddrs 인터페이스는 COM 표준 열거형 메서드를 제공하여 디렉터리에서 사용 가능한 전화 걸기 주소를 검색하고 사용합니다. ITDirectoryObject::EnumerateDialableAddrs 메서드는 이 인터페이스에 대한 포인터를 반환합니다.
IEnumDirectory IEnumDirectory 인터페이스는 ITDirectory 인터페이스에 대한 COM 표준 열거형 메서드를 제공합니다. ITRendezvous::EnumerateDefaultDirectories 메서드는 IEnumDirectory에 대한 포인터를 반환합니다.
IEnumDirectoryObject IEnumDirectoryObject 인터페이스는 ITDirectoryObject 인터페이스에 대한 COM 표준 열거형 메서드를 제공합니다. ITDirectory::EnumerateDirectoryObjects 메서드는 IEnumDirectoryObject에 대한 포인터를 반환합니다.
IEnumLocation IEnumLocation 인터페이스는 ITLocationInfo 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다. ITAddressTranslation::EnumerateLocations 메서드는 IEnumLocation에 대한 포인터를 반환합니다.
IEnumMcastScope IEnumMcastScope 인터페이스는 IMcastScope 인터페이스에 대한 COM 표준 열거형 메서드를 제공합니다. IMcastAddressAllocation::EnumerateScopes 메서드는 IEnumMcastScope에 대한 포인터를 반환합니다.

IEnumPhone IEnumPhone 인터페이스는 ITPhone 인터페이스에 대한 COM 표준 열거형 메서드를 제공합니다. ITAddress2::EnumeratePhones 및 ITTAPI2::EnumeratePhones 메서드는 IEnumPhone에 대한 포인터를 반환합니다.
IEnumPluggableSuperclassInfo IEnumPluggableSuperclassInfo 인터페이스는 ITPluggableTerminalSuperclassInfo 인터페이스에 대한 COM 표준 열거형 메서드를 제공합니다. ITTerminalSupport2::EnumeratePluggableSuperclasses 메서드는 IEnumPluggableSuperclassInfo에 대한 포인터를 반환합니다.
IEnumPluggableTerminalClassInfo IEnumPluggableTerminalClassInfo 인터페이스는 ITPluggableTerminalClassInfo 인터페이스에 대한 COM 표준 열거형 메서드를 제공합니다. ITTerminalSupport2::EnumeratePluggableTerminalClasses 메서드는 IEnumPluggableTerminalClassInfo에 대한 포인터를 반환합니다.
IEnumQueue IEnumQueue(tapi3.h) 인터페이스는 ITQueue 인터페이스에 대한 COM 표준 열거형 메서드를 제공합니다.
IEnumQueue IEnumQueue 인터페이스(tapi3cc.h)는 ITQueue 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다.
IEnumStream IEnumStream 인터페이스는 ITStream 인터페이스에 대한 COM 표준 열거형 메서드를 제공합니다. ITStreamControl::EnumerateStreams 및 ITParticipant::EnumerateStreams 메서드는 IEnumStream에 대한 포인터를 반환합니다.
IEnumSubStream IEnumSubStream 인터페이스는 ITSubStream 인터페이스에 대한 COM 표준 열거형 메서드를 제공합니다. ITSubStreamControl::EnumerateSubStreams 메서드는 IEnumSubStream에 대한 포인터를 반환합니다.
IEnumTerminal IEnumTerminal 인터페이스는 ITTerminal 인터페이스에 대한 COM 표준 열거형 메서드를 제공합니다.
IEnumTerminalClass IEnumTerminalClass 인터페이스는 사용 가능한 동적 터미널 클래스를 검색하고 사용하는 COM 표

준 열거형 메서드를 제공합니다. ITTerminalSupport::EnumerateDynamicTerminalClasses 메서드는 이 인터페이스에 대한 포인터를 반환합니다.
IMcastAddressAllocation IMcastAddressAllocation은 멀티캐스트 주소 할당을 위한 기본 인터페이스입니다. 애플리케이션은 이 인터페이스에서 COM CoCreateInstance 함수를 호출하여 멀티캐스트 클라이언트 인터페이스 개체를 만듭니다.
IMcastLeaseInfo IMcastLeaseInfo 인터페이스는 멀티캐스트 주소 할당과 관련된 정보를 얻거나 설정할 수 있는 메서드를 노출합니다. IMcastAddressAllocation::CreateLeaseInfo를 호출하여 IMcastLease 개체를 만듭니다.
IMcastScope IMcastAddressAllocation::EnumerateScopes 또는 IMcastAddressAllocation::get_Scopes 호출하여 IMcastScope 인터페이스를 가져옵니다.
ITACDGroup ITACDGroup(tapi3.h) 개체는 ACD 파일럿, 분할 또는 그룹을 반영합니다. ACD는 스위칭 시스템 내에서 호출을 큐에 대기하고 배포하는 메커니즘입니다.
ITACDGroup ITACDGroup 인터페이스(tapi3cc.h)는 전환 시스템 내에서 호출을 큐에 대기하고 배포하는 ACD(자동 호출 분포) 메커니즘을 처리합니다.
ITACDGroupEvent ITACDGroupEvent(tapi3.h) 인터페이스에는 ACD(자동 호출 배포) 그룹 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITACDGroupEvent ITACDGroupEvent 인터페이스(tapi3cc.h)에는 ACD(자동 호출 배포) 그룹 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITAddress ITAddress 인터페이스는 Address 개체의 기본 인터페이스입니다. 애플리케이션은 이 인터페이스를 사용하여 Address 개체에 대한 정보를 얻고 사용합니다.
ITAddress2 ITAddress2 인터페이스는 ITAddress 인터페이스에서 파생됩니다. ITAddress2는 전화 디바이스를 지

원하기 위해 Address 개체에 메서드를 추가합니다. TAPI 3.1에서 열거된 모든 Address 개체는 이 인터페이스를 자동으로 구현합니다.
ITAddressCapabilities ITAddressCapabilities 인터페이스는 주소의 기능에 대한 정보를 가져오는 데 사용됩니다. Address 개체에 있으며 애플리케이션은 Address 개체에서 QueryInterface를 호출하여 액세스할 수 있습니다.
ITAddressDeviceSpecificEvent ITAddressDeviceSpecificEvent는 애플리케이션이 디바이스 관련 이벤트에 대한 정보를 검색할 수 있도록 하는 메서드를 노출합니다.
ITAddressEvent ITAddressEvent 인터페이스에는 주소 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITAddressTranslation ITAddressTranslation 인터페이스는 호출 주소를 다른 형식으로 변환할 수 있는 메서드를 제공합니다. 예를 들어 애플리케이션은 전화를 걸기 전에 주소를 정식 주소에서 전화 걸기로 변환해야 할 수 있습니다.
ITAddressTranslationInfo 주소 변환 데이터를 확인하는 데 사용됩니다.
ITAgent ITAgent(tapi3.h) 인터페이스는 IDispatch 인터페이스에서 상속됩니다. 에이전트는 콜 센터의 핵심입니다.
ITAgent ITAgent 인터페이스(tapi3cc.h)는 들어오는 호출을 수신 및 처리하고 고객 또는 잠재 고객에게 발신 통화를 하는 에이전트 개체를 처리합니다.
ITAgentEvent ITAgentEvent(tapi3.h) 인터페이스에는 에이전트 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITAgentEvent ITAgentEvent 인터페이스(tapi3cc.h)에는 에이전트 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.

ITAgentHandler ITAgentHandler(tapi3.h) 인터페이스는 에이전트 개체를 만들고 ACD(자동 호출 배포) 그룹을 열거하는 메서드를 제공합니다.
ITAgentHandler ITAgentHandler 인터페이스(tapi3cc.h)는 에이전트 개체를 만들고 ACD(자동 호출 배포) 그룹을 열거하는 메서드를 제공합니다.
ITAgentHandlerEvent ITAgentHandlerEvent(tapi3.h) 인터페이스에는 에이전트 처리기 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITAgentHandlerEvent ITAgentHandlerEvent 인터페이스(tapi3cc.h)에는 에이전트 처리기 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITAgentSession ITAgentSession(tapi3.h)의 메서드를 사용하면 애플리케이션이 통계를 검색할 수 있습니다. 에이전트 세션은 에이전트, 그룹 및 주소 간의 연결을 나타냅니다.
ITAgentSession ITAgentSession 인터페이스(tapi3cc.h)는 에이전트, 그룹 및 주소 간의 연결을 나타내는 에이전트 세션을 처리합니다.
ITAgentSessionEvent ITAgentSessionEvent 인터페이스(tapi3.h)에는 에이전트 세션 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITAgentSessionEvent ITAgentSessionEvent 인터페이스(tapi3cc.h)에는 에이전트 세션 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITAllocatorProperties ITAllocatorProperties 인터페이스(tapi3.h)는 MST(미디어 스트리밍 터미널)의 버퍼 할당자 속성을 최종 사용자 또는 서버 애플리케이션에 노출합니다.
ITAllocatorProperties ITAllocatorProperties 인터페이스(tapi3ds.h)는 MST(미디어 스트리밍 터미널)의 버퍼 할당자 속성을 최종 사용자 또는 서버 애플리케이션에 노출합니다.

ITAMMediaFormat ITAMMediaFormat 인터페이스(tapi3.h)는 DirectShow 미디어 형식을 설정하고 가져옵니다.
ITAMMediaFormat ITAMMediaFormat 인터페이스(tapi3ds.h)는 DirectShow 미디어 형식을 설정하고 가져옵니다.
ITASRTerminalEvent ITASRTerminalEvent 인터페이스에는 발생한 자동 음성 인식 터미널 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITAutomatedPhoneControl ITAutomatedPhoneControl은 TAPI 전화 개체에 의해 노출되는 완전히 OLE 자동화 가능하고 스크립트 가능한 인터페이스입니다.
ITBasicAudioTerminal ITBasicAudioTerminal 인터페이스는 애플리케이션이 터미널의 기본 사운드 특성을 제어할 수 있는 메서드를 제공합니다.
ITBasicCallControl ITBasicCallControl 인터페이스는 애플리케이션에서 호출 개체에 대한 기본 전화 통신 작업을 연결, 응답 및 수행하는 데 사용됩니다.
ITBasicCallControl2 ITBasicCallControl2 인터페이스는 ITBasicCallControl 인터페이스의 확장입니다.
ITCallHub ITCallHub 인터페이스는 CallHub 개체와 관련된 정보를 검색하는 메서드를 제공합니다. IEnumCallHub::Next 및 ITTapi::get_CallHubs 메서드는 ITCallHub 인터페이스를 만듭니다.
ITCallHubEvent ITCallHubEvent 인터페이스에는 CallHub 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITCallInfo ITCallInfo 인터페이스는 Call 개체와 관련된 다양한 정보를 가져오고 설정합니다. ITAddress::get_Calls 및 IEnumCall::Next 메서드는 ITCallInfo 인터페이스를 만듭니다.
ITCallInfo2

ITCallInfo2 인터페이스는 ITCallInfo 인터페이스의 확장입니다. ITCallInfo2는 애플리케이션이 호출별로 이벤트 필터링을 설정할 수 있는 추가 메서드를 제공합니다.
ITCallInfoChangeEvent ITCallInfoChangeEvent 인터페이스에는 호출 정보 변경 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITCallingCard ITCallingCard 인터페이스는 전화 통신 통화 카드와 관련된 정보를 검색하는 메서드를 제공합니다.
ITCallMediaEvent ITCallMediaEvent 인터페이스에는 미디어 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITCallNotificationEvent ITCallNotificationEvent 인터페이스에는 호출 알림 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITCallStateEvent ITCallStateEvent 인터페이스에는 호출 상태 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITCollection ITCollection 인터페이스를 사용하면 Visual Basic으로 작성된 애플리케이션과 같은 Automation 클라이언트 애플리케이션이 컬렉션 정보를 검색할 수 있습니다.
ITCollection2 ITCollection2 인터페이스는 ITCollection 인터페이스의 확장입니다. ITCollection2는 컬렉션을 수정하기 위한 추가 메서드를 노출합니다.
ITCustomTone ITCustomTone 인터페이스는 일부 휴대폰 집합에서 사용할 수 있는 사용자 지정 톤을 자세히 제어할 수 있는 메서드를 노출합니다.
ITDetectTone ITDetectTone 인터페이스는 애플리케이션이 TAPI Server에서 톤 이벤트를 생성해야 하는 톤 및 톤 특성을 지정할 수 있도록 하는 메서드를 노출합니다.
ITDigitDetectionEvent

ITDigitDetectionEvent 인터페이스에는 DTMF 숫자 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITDigitGenerationEvent ITDigitGenerationEvent 인터페이스에는 숫자 생성 이벤트를 설명하는 메서드가 포함되어 있습니다.
ITDigitsGatheredEvent ITDigitsGatheredEvent 인터페이스는 TAPI 서버가 서버가 애플리케이션에 필요한 숫자를 수집했음을 나타내는 이벤트를 보낼 때 애플리케이션이 데이터를 검색할 수 있도록 하는 메서드를 노출합니다.
ITDirectory ITDirectory 인터페이스는 특정 디렉터리에 해당하는 Directory 개체에 의해 노출됩니다.
ITDirectoryObject ITDirectoryObject 인터페이스는 ITDirectory 인터페이스를 사용하여 추가 및 삭제할 수 있는 모든 개체에서 지원하는 공통 인터페이스입니다.
ITDirectoryObjectConference ITDirectoryObjectConference 인터페이스는 애플리케이션이 회의 세부 정보를 설정하고 가져올 수 있도록 하는 메서드를 제공합니다. ITDirectoryObjectConference 인터페이스는 ITDirectoryObject에서 QueryInterface를 호출하여 만들어집니다.
ITDirectoryObjectUser ITDirectoryObjectUser 인터페이스는 User 개체에서 지원하는 공통 인터페이스입니다. 이 인터페이스는 ITDirectoryObject에서 QueryInterface를 호출하여 만듭니다.
ITDispatchMapper ITDispatchMapper 인터페이스를 사용하면 한 인터페이스의 디스패치 포인터와 다른 인터페이스의 GUID를 고려하여 애플리케이션이 개체에서 다른 인터페이스의 디스패치 포인터를 검색할 수 있습니다.
ITFileTerminalEvent ITFileTerminalEvent 인터페이스에는 발생한 파일 터미널 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITFileTrack ITFileTrack 인터페이스는 애플리케이션이 파일 터미널 트랙과 관련된 정보를 가져와서 설정할 수

있도록 하는 메서드를 노출합니다. ITFileTerminalEvent::get_Track 메서드는 ITFileTrack 인터페이스를 만듭니다.
ITForwardInformation ITForwardInformation 인터페이스는 착신 전환의 설정 및 구현을 위한 메서드를 제공합니다.
ITForwardInformation2 ITForwardInformation2 인터페이스는 전달 정보를 제어하기 위한 추가 메서드를 제공하는 메서드를 노출합니다. 기본 전달 제어 메서드는 ITForwardInformation을 참조하세요.
ITILSConfig ITILSConfig 인터페이스를 사용하면 ILS 디렉터리를 구성할 수 있습니다.
ITLegacyAddressMediaControl ITLegacyAddressMediaControl 인터페이스는 디바이스 및 해당 구성에 직접 액세스해야 하는 레거시 애플리케이션을 지원하기 위해 제공됩니다. Address 개체에 의해 노출되며 ITAddress에서 QueryInterface를 호출하여 만들 수 있습니다.
ITLegacyAddressMediaControl2 ITLegacyAddressMediaControl2 인터페이스는 ITLegacyAddressMediaControl 인터페이스에서 파생됩니다. ITLegacyAddressMediaControl2는 줄 디바이스와 관련된 매개 변수의 구성을 허용하는 추가 메서드를 제공합니다.
ITLegacyCallMediaControl ITLegacyCallMediaControl 인터페이스는 디바이스와 직접 통신해야 하는 레거시 애플리케이션을 지원합니다. 이 인터페이스는 호출 개체에 노출되며 ITBasicCallControl에서 QueryInterface를 호출하여 만들 수 있습니다.
ITLegacyCallMediaControl2 ITLegacyCallMediaControl2 인터페이스는 ITLegacyCallMediaControl 인터페이스의 확장입니다. ITLegacyCallMediaControl2는 주로 톤 감지 및 생성을 위한 추가 메서드를 제공합니다.
ITLegacyWaveSupport ITLegacyWaveSupport 인터페이스를 사용하면 애플리케이션이 Wave API를 사용하여 레거시 TSP(PRE-TAPI 3)에서 만든 터미널을 제어할 수 있는지 여부를 검색할 수 있습니다.
ITLocationInfo ITLocationInfo 인터페이스는 통화 당사자의 위치와 관련된 정보를 가져오는 데 사용됩니다. 제어판 아래의 전화 통신 애플릿을 사용하여 입력되는 위치 정보입니다.

ITMediaControl ITMediaControl 인터페이스는 미디어 파일 터미널에 대한 일반 인터페이스입니다. 인터페이스는 애플리케이션이 재생과 같은 현재 작업을 시작, 중지 또는 일시 중지할 수 있도록 하는 메서드를 노출합니다.
ITMediaPlayback ITMediaPlayback 인터페이스는 애플리케이션이 재생할 파일 목록을 설정하고 가져올 수 있도록 하는 재생 관련 메서드를 제공합니다. 이 인터페이스는 ITTerminal에서 QueryInterface를 호출하여 만듭니다.
ITMediaRecord ITMediaRecord 인터페이스는 애플리케이션이 기록할 파일의 이름을 설정하고 가져올 수 있도록 하는 기록 관련 메서드를 제공합니다.
ITMediaSupport ITMediaSupport 인터페이스는 애플리케이션이 이 인터페이스를 노출하는 Address 개체에 대한 미디어 지원 기능을 검색할 수 있도록 하는 메서드를 제공합니다.
ITMSPAddress ITMSPAddress 인터페이스(tapi3.h)는 MSP에 의해 구현되며 TAPI DLL에 대한 미디어 서비스 공급자를 나타냅니다.
ITMSPAddress ITMSPAddress(msp.h) 인터페이스는 MSP에 의해 구현되며 TAPI DLL에 대한 미디어 서비스 공급자를 나타냅니다.
ITMultiTrackTerminal 이 ITMultiTrackTerminal 인터페이스는 모든 멀티트랙 터미널에 노출됩니다. 인터페이스에는 트랙을 열거, 생성 및 제거하는 메서드가 포함됩니다. ITMultiTrackTerminal 인터페이스는 ITTerminal에서 QueryInterface를 호출하여 만듭니다.
ITPhone ITPhone 인터페이스는 TAPI 3.1 개체 모델의 새 Phone 개체에 대한 기본 인터페이스입니다.
ITPhoneDeviceSpecificEvent ITPhoneDeviceSpecificEvent는 애플리케이션이 휴대폰 디바이스 관련 이벤트에 대한 정보를 검색할 수 있도록 하는 메서드를 노출합니다.
ITPhoneEvent

ITPhoneEvent 인터페이스에는 발생한 전화 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITPluggableTerminalClassInfo ITPluggableTerminalClassInfo 인터페이스는 애플리케이션이 플러그형 터미널과 관련된 정보를 검색할 수 있도록 하는 메서드를 노출합니다.
ITPluggableTerminalClassRegistration ITPluggableTerminalClassRegistration 인터페이스는 플러그형 터미널에 대한 레지스트리 항목을 만들고 수정하고 삭제할 수 있는 메서드를 노출합니다.
ITPluggableTerminalEventSink ITPluggableTerminalEventSink 인터페이스(tapi3.h)는 플러그형 터미널의 변경에 대해 클라이언트 애플리케이션에 알리는 메시지를 발생시키는 메서드를 제공합니다.
ITPluggableTerminalEventSink ITPluggableTerminalEventSink(msp.h) 인터페이스는 플러그형 터미널의 변경 내용을 클라이언트 애플리케이션에 알리는 메시지를 발생시키는 메서드를 제공합니다.
ITPluggableTerminalEventSinkRegistration ITPluggableTerminalEventSinkRegistration 인터페이스(tapi3.h)는 플러그형 터미널 이벤트에 대한 클라이언트 애플리케이션을 등록 및 등록 취소합니다.
ITPluggableTerminalEventSinkRegistration ITPluggableTerminalEventSinkRegistration(msp.h) 인터페이스는 플러그형 터미널 이벤트에 대한 클라이언트 애플리케이션을 등록 및 등록 취소합니다.
ITPluggableTerminalInitialization ITPluggableTerminalInitialization 인터페이스는 플러그형 터미널에 의해 구현되며 터미널 관리자가 터미널을 초기화할 수 있도록 합니다. ITPluggableTerminalInitialization 인터페이스는 ITTerminal에서 QueryInterface를 호출하여 만듭니다.
ITPluggableTerminalSuperclassInfo ITPluggableTerminalSuperclassInfo 인터페이스는 플러그형 터미널 클래스의 이름과 CLSID를 가져오는 메서드를 노출합니다.
ITPluggableTerminalSuperclassRegistration ITPluggableTerminalSuperclassRegistration 인터페이스는 터미널 슈퍼클래스(이름 및 CLSID)에 대한 정보를 가져와서 설정하는 메서드를 노출합니다.

ITPrivateEvent ITPrivateEvent 인터페이스는 애플리케이션이 프라이빗 개체에 의해 생성된 이벤트를 처리할 수 있도록 하는 메서드를 노출합니다.
ITQOSEvent ITQOSEvent 인터페이스에는 QOS(서비스 품질) 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITQueue ITQueue 인터페이스(tapi3.h)는 큐와 관련된 정보를 가져오고 설정합니다.
ITQueue ITQueue 인터페이스(tapi3cc.h)는 큐와 관련된 정보를 가져오고 설정합니다.
ITQueueEvent ITQueueEvent 인터페이스(tapi3.h)에는 ACD(자동 호출 배포) 큐 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITQueueEvent ITQueueEvent 인터페이스(tapi3cc.h)에는 ACD(자동 호출 배포) 큐 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITRendezvous ITRendezvous 인터페이스는 랑데부 컨트롤에 대한 기본 인터페이스입니다. 애플리케이션은 이 인터페이스에서 COM CoCreateInstance 함수를 호출하여 Rendezvous 개체를 만듭니다.
ITRequest ITRequest 인터페이스를 사용하면 애플리케이션에서 보조 전화 통신을 사용할 수 있습니다. 보조 전화 통신은 개발자가 전화 통신을 완전히 읽지 않아도 전화를 걸 수 있는 간단한 메커니즘으로 전화 통신 지원 애플리케이션을 제공합니다.
ITRequestEvent ITRequestEvent 인터페이스에는 애플리케이션이 보조 전화 통신 요청 이벤트를 수신하고 처리할 수 있는 메서드가 포함되어 있습니다.
ITScriptableAudioFormat ITScriptableAudioFormat 인터페이스는 스크립팅 가능한 클라이언트가 트랙의 오디오 형식을 얻거나 오디오 형식을 설정하는 데 사용됩니다. 인터페이스는 WAVEFORMATEX 구조체의 각 멤버에 대한 속성을 제공합니다.

ITStaticAudioTerminal ITStaticAudioTerminal 인터페이스는 TAPI 3.1 MSP가 모든 정적 오디오 터미널에 노출해야 하는 인터페이스입니다. 인터페이스는 휴대폰 디바이스를 지원하는 데 필요한 정적 오디오 터미널 개체의 메서드를 정의합니다.
ITStream ITStream 인터페이스는 애플리케이션이 스트림에 대한 정보를 검색할 수 있도록 하는 메서드를 노출합니다. 스트림을 시작, 일시 중지 또는 중지하려면 이고, 스트림에서 터미널을 선택하거나 선택 취소하려면 및 를 선택하여 스트림에서 선택한 터미널 목록을 가져옵니다.
ITStreamControl ITStreamControl 인터페이스는 호출의 미디어 스트리밍 기능을 나타내며 애플리케이션이 스트림을 열거, 생성 또는 제거할 수 있도록 하는 메서드를 노출합니다.
ITSubStream ITSubStream은 ITStream의 구성 요소이며, 애플리케이션에 미디어 스트리밍을 더 세밀하게 제어할 수 있습니다.
ITSubStreamControl ITSubStreamControl 인터페이스는 애플리케이션이 하위 스트림을 열거, 생성 또는 제거할 수 있도록 하는 메서드를 노출합니다. 많은 MSP가 이 인터페이스를 지원하지 않습니다.
ITTAPI ITTAPI 인터페이스는 TAPI 개체의 기본 인터페이스입니다. TAPI 개체는 CoCreateInstance에 의해 만들어집니다. CoCreateInstance에 대한 자세한 내용은 COM 설명서를 참조하세요. 다른 모든 TAPI 3 개체는 TAPI 3 자체에서 만들어집니다.
ITTAPI2 ITTAPI2 인터페이스는 ITTAPI 인터페이스에서 파생됩니다. TAPI 개체에 추가 메서드를 추가하여 휴대폰 디바이스를 지원합니다.
ITTAPICallCenter ITTAPICallCenter 인터페이스(tapi3.h)는 콜 센터 컨트롤에 진입점을 제공합니다.
ITTAPICallCenter ITTAPICallCenter 인터페이스(tapi3cc.h)는 콜 센터 컨트롤에 진입점을 제공합니다.
ITTAPIEventNotification

ITTAPIEventNotification 인터페이스는 애플리케이션이 이벤트 정보 처리를 제어할 수 있는 나가는 인터페이스입니다.
ITTAPIObjectEvent ITTAPIObjectEvent 인터페이스에는 TAPI 개체 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITTAPIObjectEvent2 ITTAPIObjectEvent2 인터페이스는 ITTAPIObjectEvent 인터페이스의 확장입니다. ITTAPIObjectEvent2는 TAPI 개체 이벤트를 발생시킨 휴대폰 개체의 ITPhone 인터페이스에 대한 포인터를 반환하는 추가 메서드를 노출합니다.
ITTerminal ITTerminal 인터페이스는 터미널 개체의 기본 인터페이스입니다.
ITTerminalControl ITTerminalControl 인터페이스는 MSP 핸들을 가져와 필터 그래프에서 터미널을 연결 및 연결 해제하고 렌더러를 실행하거나 중지하는 메서드를 제공합니다.
ITTerminalManager ITTerminalManager 인터페이스는 MSP에서 동적 터미널을 만드는 데 사용됩니다.
ITTerminalManager2 ITTerminalManager2 인터페이스는 현재 시스템에 등록된 플러그형 터미널 클래스에 대한 정보를 검색하는 메서드를 노출합니다. ITTerminalManager2는 ITTerminalManager 인터페이스에서 파생됩니다.
ITTerminalSupport ITTerminalSupport 인터페이스는 MSP가 있는 경우에만 Address 개체에 노출됩니다. 이 인터페이스의 메서드를 사용하면 애플리케이션이 사용 가능한 터미널을 검색하고/또는 터미널을 만들고 필요한 터미널 개체에 대한 포인터를 가져올 수 있습니다.
ITTerminalSupport2 ITTerminalSupport2 인터페이스는 ITTerminalSupport 인터페이스에서 파생됩니다. ITTerminalSupport2는 C, C++및 스크립팅 애플리케이션을 통해 플러그형 터미널 클래스 및 슈퍼클래스에 대한 정보 검색을 지원합니다.
ITToneDetectionEvent ITToneDetectionEvent 인터페이스는 애플리케이션이 톤 감지 이벤트에 대한 정보를 검색할 수 있도록 하는 메서드를 노출합니다.

ITToneTerminalEvent ITToneTerminalEvent 인터페이스에는 발생한 톤 터미널 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITTTSTerminalEvent ITTTSTerminalEvent 인터페이스에는 발생한 TTS(Text to Speech) 터미널 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.

구조체

LINEADDRESSCAPS LINEADDRESSCAPS 구조체는 지정된 주소의 기능을 설명합니다. lineGetAddressCaps 함수와 TSPI_lineGetAddressCaps 함수는 LINEADDRESSCAPS 구조를 반환합니다.
LINEADDRESSSTATUS LINEADDRESSSTATUS 구조체는 주소의 현재 상태 설명합니다. lineGetAddressStatus 함수와 TSPI_lineGetAddressStatus 함수는 LINEADDRESSSTATUS 구조를 반환합니다.
LINEAGENTACTIVITYENTRY LINEAGENTACTIVITYENTRY 구조는 단일 ACD 에이전트 작업을 설명합니다. LINEAGENTACTIVITYLIST 구조체에는 LINEAGENTACTIVITYENTRY 구조체의 배열이 포함될 수 있습니다.
LINEAGENTACTIVITYLIST LINEAGENTACTIVITYLIST 구조는 ACD 에이전트 활동 목록을 설명합니다. 이 구조체에는 LINEAGENTACTIVITYENTRY 구조체의 배열이 포함될 수 있습니다. lineGetAgentActivityList 함수는 LINEAGENTACTIVITYLIST 구조를 반환합니다.
LINEAGENTCAPS LINEAGENTCAPS 구조는 ACD 에이전트의 기능을 설명합니다. lineGetAgentCaps 함수는 LINEAGENTCAPS 구조를 반환합니다.
LINEAGENTENTRY LINEAGENTENTRY 구조는 개별 ACD 에이전트를 설명합니다. LINEAGENTLIST 구조체에는 LINEAGENTENTRY 구조체의 배열이 포함될 수 있습니다.

LINEAGENTGROUPENTRY LINEAGENTGROUPENTRY 구조는 ACD 에이전트 그룹에 대한 정보를 제공합니다. LINEAGENTGROUPLIST 구조체에는 LINEAGENTGROUPENTRY 구조체의 배열이 포함될 수 있습니다.
LINEAGENTGROUPLIST LINEAGENTGROUPLIST 구조는 ACD 에이전트 그룹 목록을 설명합니다. 이 구조체에는 LINEAGENTGROUPENTRY 구조체의 배열이 포함될 수 있습니다.
LINEAGENTINFO LINEAGENTINFO 구조체에는 ACD 에이전트에 대한 정보가 포함되어 있습니다. lineGetAgentInfo 함수는 LINEAGENTINFO 구조를 반환합니다.
LINEAGENTLIST LINEAGENTLIST 구조는 ACD 에이전트 목록을 설명합니다. 이 구조체에는 LINEAGENTENTRY 구조체의 배열이 포함될 수 있습니다.
LINEAGENTSESSIONENTRY LINEAGENTSESSIONENTRY 구조체는 ACD 에이전트 세션을 설명합니다. LINEAGENTSESSIONLIST 구조체에는 LINEAGENTSESSIONENTRY 구조체의 배열이 포함될 수 있습니다.
LINEAGENTSESSIONINFO LINEAGENTSESSIONINFO 구조체에는 ACD 에이전트 세션에 대한 정보가 포함됩니다. lineGetAgentSessionInfo 함수는 LINEAGENTSESSIONINFO 구조를 반환합니다.
LINEAGENTSESSIONLIST LINEAGENTSESSIONLIST 구조는 ACD 에이전트 세션 목록을 설명합니다. 이 구조체에는 LINEAGENTSESSIONENTRY 구조체의 배열이 포함될 수 있습니다. lineGetAgentSessionList 함수는 LINEAGENTSESSIONLIST 구조를 반환합니다.
LINEAGENTSTATUS LINEAGENTSTATUS 구조는 ACD 에이전트의 현재 상태 설명합니다. lineGetAgentStatus 함수는 LINEAGENTSTATUS 구조를 반환합니다.
LINEAPPINFO LINEAPPINFO 구조체에는 현재 실행 중인 애플리케이션에 대한 정보가 포함되어 있습니다. LINEDEVSTATUS 구조체에는 LINEAPPINFO 구조체의 배열이 포함될 수 있습니다.
LINECALLHUBTRACKINGINFO

LINECALLHUBTRACKINGINFO 구조에는 호출 허브에서 사용할 수 있는 추적 유형을 보고하는 정보가 포함되어 있습니다. 이 구조는 TAPI 버전 2.2 이상을 협상하는 애플리케이션에만 노출됩니다.
LINECALLINFO LINECALLINFO 구조체에는 호출에 대한 정보가 포함됩니다.
LINECALLLIST LINECALLLIST 구조는 호출 핸들 목록을 설명합니다. 이 형식의 구조체는 LineGetNewCalls 및 lineGetConfRelatedCalls 함수에 의해 반환됩니다.
LINECALLPARAMS LINECALLPARAMS 구조체는 lineMakeCall 및 TSPI_lineMakeCall 함수를 사용하여 호출할 때 제공되는 매개 변수를 설명합니다. LINECALLPARAMS 구조체는 lineOpen 함수와 같은 다른 작업에서도 매개 변수로 사용됩니다.
LINECALLSTATUS LINECALLSTATUS 구조체는 호출의 현재 상태 설명합니다.
LINECALLTREATMENTENTRY LINECALLTREATMENTENTRY 구조는 현재 통화에서 음악, 녹음된 알림 또는 무음과 같은 통화 처리 유형에 대한 정보를 제공합니다. LINEADDRESSCAPS 구조체에는 LINECALLTREATMENTENTRY 구조체의 배열이 포함될 수 있습니다.
LINECARDENTRY LINECARDENTRY 구조는 호출 카드 설명합니다. LINETRANSLATECAPS 구조체에는 LINECARDENTRY 구조체의 배열이 포함될 수 있습니다.
LINECOUNTRYENTRY 단일 국가/지역 항목에 대한 데이터를 제공합니다.
LINECOUNTRYLIST LINECOUNTRYLIST 구조는 국가/지역 목록을 설명합니다. 이 구조체는 LINECOUNTRYENTRY 구조체의 배열을 포함할 수 있습니다. LINECOUNTRYLIST는 lineGetCountry 함수에 의해 반환됩니다.
LINEDEVCAPS LINEDEVCAPS 구조는 라인 디바이스의 기능을 설명합니다. lineGetDevCaps 함수와 TSPI_lineGetDevCaps 함수는 LINEDEVCAPS 구조를 반환합니다.
LINEDEVSTATUS

LINEDEVSTATUS 구조는 라인 디바이스의 현재 상태 설명합니다. lineGetLineDevStatus 함수와 TSPI_lineGetLineDevStatus 함수는 LINEDEVSTATUS 구조를 반환합니다.
LINEDIALPARAMS LINEDIALPARAMS 구조체는 전화 걸기 관련 필드의 컬렉션을 지정합니다. lineSetCallParams 함수 또는 TSPI_lineSetCallParams 함수를 호출하여 LINEDIALPARAMS 구조를 사용하여 호출에 대한 매개 변수를 설정합니다.
LINEEXTENSIONID LINEEXTENSIONID 구조체는 확장 식별자를 설명합니다.
LINEFORWARD LINEFORWARD 구조체는 전달 지침의 항목을 설명합니다. LINEFORWARDLIST 및 LINEADDRESSSTATUS 구조체에는 LINEFORWARD 구조체의 배열이 포함될 수 있습니다.
LINEFORWARDLIST LINEFORWARDLIST 구조체는 전달 지침 목록을 설명합니다. 이 구조체는 LINEFORWARD 구조체의 배열을 포함할 수 있습니다. lineForward 및 TSPI_lineForward 함수는 LINEFORWARDLIST 구조를 사용합니다.
LINEGENERATETONE LINEGENERATETONE 구조체에는 생성할 톤에 대한 정보가 포함되어 있습니다. 이 구조체는 lineGenerateTone 및 TSPI_lineGenerateTone 함수에서 사용됩니다.
LINEINITIALIZEEXPARAMS LINEINITIZALIZEEXPARAMS 구조는 LINEINITIALIZEEX를 사용하여 호출할 때 제공되는 매개 변수를 설명합니다.
LINELOCATIONENTRY 주소 변환 컨텍스트를 제공하는 데 사용되는 위치를 설명합니다.
LINEMEDIACONTROLCALLSTATE LINEMEDIACONTROLCALLSTATE 구조체는 하나 이상의 호출 상태로 전환을 검색할 때 실행할 미디어 작업을 설명합니다. lineSetMediaControl 및 TSPI_lineSetMediaControl 함수는 이 구조를 사용합니다.
LINEMEDIACONTROLDIGIT LINEMEDIACONTROLDIGIT 구조체는 숫자를 검색할 때 실행할 미디어 작업을 설명합니다. 배열의 항목으로 사용됩니다. lineSetMediaControl 및 TSPI_lineSetMediaControl 함수는 이 구조를 사용합니다.

LINEMEDIACONTROLMEDIA LINEMEDIACONTROLMEDIA 구조는 미디어 형식 변경을 검색할 때 실행할 미디어 작업을 설명합니다. 배열의 항목으로 사용됩니다. lineSetMediaControl 및 TSPI_lineSetMediaControl 함수는 이 구조를 사용합니다.
LINEMEDIACONTROLTONE LINEMEDIACONTROLTONE 구조체는 톤이 검색될 때 실행할 미디어 동작을 설명합니다. 배열의 항목으로 사용됩니다. lineSetMediaControl 및 TSPI_lineSetMediaControl 함수는 이 구조를 사용합니다.
LINEMESSAGE LINEMESSAGE 구조에는 애플리케이션이 현재 열려 있는 줄의 상태 변경을 지정하는 매개 변수 값이 포함되어 있습니다. lineGetMessage 함수는 LINEMESSAGE 구조를 반환합니다.
LINEMONITORTONE LINEMONITORTONE 구조체는 모니터링할 톤을 설명합니다. 배열의 항목으로 사용됩니다. lineMonitorTones 및 TSPI_lineMonitorTones 함수는 이 구조를 사용합니다.
LINEPROVIDERENTRY LINEPROVIDERENTRY 구조는 단일 서비스 공급자 항목에 대한 정보를 제공합니다. 이러한 구조체의 배열은 lineGetProviderList 함수에서 반환하는 LINEPROVIDERLIST 구조체의 일부로 반환됩니다.
LINEPROVIDERLIST LINEPROVIDERLIST 구조는 서비스 공급자 목록을 설명합니다. 이 형식의 구조체는 lineGetProviderList 함수에 의해 반환됩니다. LINEPROVIDERLIST 구조체에는 LINEPROVIDERENTRY 구조체의 배열이 포함될 수 있습니다.
LINEPROXYREQUEST LINEPROXYREQUEST 구조체에는 프록시 요청을 만드는 애플리케이션의 매개 변수 값이 포함되어 있습니다. 여러 TAPI 콜 센터 함수는 LINEPROXYREQUEST 구조를 참조하는 LINE_PROXYREQUEST 메시지를 생성합니다.
LINEPROXYREQUESTLIST LINEPROXYREQUESTLIST 구조체는 프록시 요청 목록을 설명합니다. lineGetProxyStatus 함수는 LINEPROXYREQUESTLIST 구조를 반환합니다.
LINEQUEUEENTRY LINEQUEUEENTRY 구조는 단일 큐 항목에 대한 정보를 제공합니다. LINEQUEUELIST 구조체에는

LINEQUEUEENTRY 구조체의 배열이 포함될 수 있습니다. 이 구조에는 TAPI 3.0 버전 협상이 필요합니다.
LINEQUEUEINFO LINEQUEUEINFO 구조체는 라인 디바이스의 큐에 대한 정보를 제공합니다. lineGetQueueInfo 함수는 LINEQUEUEINFO 구조를 반환합니다. 이 구조에는 TAPI 3.0 버전 협상이 필요합니다.
LINEQUEUELIST LINEQUEUELIST 구조는 큐 목록을 설명합니다. 이 구조체는 LINEQUEUEENTRY 구조체의 배열을 포함할 수 있습니다. lineGetQueueList 함수는 LINEQUEUELIST 구조를 반환합니다. LINEQUEUELIST에는 TAPI 3.0 버전 협상이 필요합니다.
LINEREQMAKECALL LINEREQMAKECALL 구조체는 lineGetRequest 함수에 대한 호출로 시작된 요청을 설명합니다.
LINEREQMEDIACALL lineGetRequest 함수에 대한 호출로 시작된 요청을 설명합니다. 이 데이터 구조는 사용되지 않으므로 사용하면 안 됩니다.
LINETERMCAPS LINETERMCAPS 구조는 라인 터미널 디바이스의 기능을 설명합니다. LINEDEVCAPS 구조체에는 LINETERMCAPS 구조체의 배열이 포함될 수 있습니다.
LINETRANSLATECAPS LINETRANSLATECAPS 구조체는 주소 변환 기능을 설명합니다.
LINETRANSLATEOUTPUT LINETRANSLATEOUTPUT 구조체는 주소 변환의 결과를 설명합니다. lineTranslateAddress 함수는 이 구조를 사용합니다.
MSP_EVENT_INFO MSP_EVENT_INFO 구조체(tapi3.h)는 GetEvent 메서드에서 반환되는 이벤트 형식을 정의합니다.
MSP_EVENT_INFO MSP_EVENT_INFO(msp.h) 구조체는 GetEvent 메서드에서 반환되는 이벤트 형식을 정의합니다.
MSPEVENTITEM MSPEVENTITEM 구조에는 MSP 이벤트에 대한 링크 포인터 및 이벤트 유형이 포함됩니다.

PHONEBUTTONINFO PHONEBUTTONINFO 구조에는 휴대폰 디바이스의 단추에 대한 정보가 포함되어 있습니다. 이 구조체는 여러 TAPI 및 TSPI 함수에서 사용됩니다.
PHONECAPS PHONECAPS 구조는 휴대폰 디바이스의 기능을 설명합니다. phoneGetDevCaps 및 TSPI_phoneGetDevCaps 함수는 이 구조를 반환합니다.
PHONEEXTENSIONID PHONEEXTENSIONID 구조체는 확장 식별자를 설명합니다.
PHONEINITIALIZEEXPARAMS PHONEINITIALIZEEXPARAMS 구조체에는 애플리케이션과 TAPI 간의 연결을 설정하는 데 사용되는 매개 변수가 포함되어 있습니다. 예를 들어 애플리케이션에서 선택한 이벤트 알림 메커니즘입니다. phoneInitializeEx 함수는 이 구조를 사용합니다.
PHONEMESSAGE PHONEMESSAGE 구조에는 애플리케이션에 배달하기 위해 대기하는 다음 메시지가 포함됩니다. phoneGetMessage 함수는 이 구조를 반환합니다.
PHONESTATUS PHONESTATUS 구조는 휴대폰 디바이스의 현재 상태 설명합니다. phoneGetStatus 및 TSPI_phoneGetStatus 함수는 이 구조를 반환합니다.
TAPI_CUSTOMTONE TAPI_CUSTOMTONE 구조에는 사용자 지정 톤을 정의하는 매개 변수가 포함되어 있습니다.
TAPI_DETECTTONE TAPI_DETECTTONE 구조는 모니터링할 톤을 설명합니다. 배열의 항목으로 사용됩니다.
VARSTRING VARSTRING 구조체는 변함없이 크기가 조정된 문자열을 반환하는 데 사용됩니다. 줄 디바이스 클래스와 휴대폰 디바이스 클래스 모두에서 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

전화 통신 서비스 공급자

아티클2023. 03. 09.

전화 통신 서비스 공급자 기술 개요입니다.

전화 통신 서비스 공급자 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 전화 통신 애플리케이션 프로그래밍 인터페이스

텍스트 서비스 프레임워크 비교

아티클2023. 03. 11.

Text Services Framework 기술 개요입니다.

Text Services 프레임워크를 개발하려면 다음 헤더가 필요합니다.

- ctffunc.h
- ctfspui.h
- ctfutb.h
- inputscope.h
- msaatext.h
- msctf.h
- msctfmonitorapi.h
- textstor.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 텍스트 서비스 프레임워크 비교

열거형

InputScope InputScope 열거형에는 지정된 필드에 적용되는 입력 범위를 지정하는 값이 포함됩니다.
TF_DA_ATTR_INFO TF_DA_ATTR_INFO 열거형의 요소는 TF_DISPLAYATTRIBUTE 구조에서 텍스트 변환 데이터를 지정하는 데 사용됩니다.
TF_DA_COLORTYPE TF_DA_COLORTYPE 열거형의 요소는 TF_DA_COLOR 구조에 포함된 색의 형식을 지정합니다.
TF_DA_LINESTYLE TF_DA_LINESTYLE 열거형 요소는 TF_DA_COLOR 구조체에서 표시 특성의 밑줄 스타일을 지정합니다.
TfActiveSelEnd TfActiveSelEnd 열거형의 요소는 선택한 텍스트 범위의 끝을 지정합니다.

TfAnchor TfAnchor 열거형의 요소는 ITfRange 개체의 시작 앵커 또는 끝 앵커를 지정합니다.
TfCandidateResult TfCandidateResult 열거형의 요소는 ITfCandidateList::SetResult 메서드와 함께 사용하여 지정된 후보 문자열에서 수행된 재연결 작업의 결과를 지정합니다.
TfGravity TfGravity 열거형의 요소는 ITfRange 개체의 앵커와 연결된 중력 유형을 지정합니다.
TfIntegratableCandidateListSelectionStyle TfIntegratableCandidateListSelectionStyle 열거형의 요소는 통합 가능한 후보 목록 선택 스타일을 지정합니다.
TfLayoutCode TfLayoutCode 열거형의 요소는 ITfTextLayoutSink::OnLayoutChange 알림의 레이아웃 변경 유형을 지정합니다.
TfLBBalloonStyle TfLBBalloonStyle 열거형의 요소는 언어 막대 풍선 스타일을 지정하는 데 사용됩니다.
TfLBIClick TfLBIClick 열거형의 요소는 도구 모음 항목을 클릭하는 데 사용된 마우스 단추를 지정합니다.
TfSapiObject TfSapiObject 열거형의 요소는 ITfFnGetSAPIObject::Get 메서드와 함께 사용되어 SAPI(Speech API) 개체의 특정 형식을 지정합니다.
TfShiftDir TfShiftDir 열거형의 요소는 범위 앵커가 이동되는 방향을 지정합니다.
TKBLayoutType TKBLayoutType 열거형의 요소는 ITfFnGetPreferredTouchKeyboardLayout::GetLayout 호출에서 IME에 의해 전달되어 레이아웃 유형을 지정합니다.
TsActiveSelEnd TsActiveSelEnd 열거형의 요소는 현재 텍스트 저장소 선택 영역의 끝을 지정합니다.

TsGravity TsGravity 열거형의 요소는 IAnchor 개체와 연결된 중력 유형을 지정합니다.
TsLayoutCode TsLayoutCode 열거형의 요소는 ITextStoreACPSink::OnLayoutChange 또는 ITextStoreAnchorSink::OnLayoutChange 알림에서 레이아웃 변경 유형을 지정하는 데 사용됩니다.
TsRunType TsRunType 열거형의 요소는 텍스트 실행이 표시되거나 숨겨지거나 텍스트 실행에 포함된 프라이빗 데이터 형식인지 여부를 지정합니다.
TsShiftDir TsShiftDir 열거형의 요소는 앵커를 이동할 방향을 지정합니다.

Functions

중단 ITfCandidateListUIElementBehavior::Abort 메서드는 후보 목록을 닫습니다. 현재 선택이 완료될 것이라는 보장은 없습니다.
활성화 ITfTextInputProcessor::Activate 메서드
활성화 ITfThreadMgr::Activate 메서드
활성화 호출 스레드에 대한 TSF를 활성화합니다.
ActivateEx ITfTextInputProcessorEx::ActivateEx 메서드는 사용자 세션이 시작될 때 텍스트 서비스를 활성화합니다. 텍스트 서비스가 ITfTextInputProcessorEx를 구현하고 ActivateEx가 호출되면 ITfTextInputProcessor::Activate가 호출되지 않습니다.

ActivateEx TSF가 활성화되는 방법을 지정하는 플래그를 사용하여 호출 스레드에 대한 TSF를 초기화하고 활성화합니다.
ActivateEx ITfThreadMgrEx::ActivateEx 메서드는 애플리케이션에서 호출 스레드에 대한 TSF를 초기화하고 활성화하는 데 사용됩니다. ITfThreadMgr::Activate와 달리 ITfThreadMgrEx::ActivateEx는 플래그를 사용하여 TSF가 활성화되는 방법을 지정할 수 있습니다.
ActivateLanguageProfile ITfInputProcessorProfiles::ActivateLanguageProfile 메서드
ActivateProfile ITfInputProcessorProfileMgr::ActivateProfile 메서드는 지정된 텍스트 서비스의 프로필 또는 키보드 레이아웃을 활성화합니다.
AddItem ITfLangBarItemMgr::AddItem 메서드
AddLanguageProfile ITfInputProcessorProfiles::AddLanguageProfile 메서드
AddMenuItem ITfMenu::AddMenuItem 메서드
AdjustForInsert ITfRange::AdjustForInsert 메서드는 텍스트 삽입을 위해 조정할 텍스트 범위를 확장하거나 축소합니다.
AdviseEventSink ITfLangBarMgr::AdviseEventSink 메서드는 언어 모음 이벤트에 대한 싱크를 조언합니다.
AdviseItemSink ITfLangBarItemMgr::AdviseItemSink 메서드
AdviseItemsSink ITfLangBarItemMgr::AdviseItemsSink 메서드

AdviseKeyEventSink ITfKeystrokeMgr::AdviseKeyEventSink 메서드
AdviseMouseSink ITfMouseTracker::AdviseMouseSink 메서드
AdviseMouseSink ITfMouseTrackerACP::AdviseMouseSink 메서드
AdviseSingleSink ITfSourceSingle::AdviseSingleSink 메서드
AdviseSink ITextStoreACP::AdviseSink 메서드는 ITextStoreACPSink 인터페이스에서 새 advise 싱크를 설치하거나 기존 advise 싱크를 수정합니다. 싱크 인터페이스는 펑크 매개 변수에 의해 지정됩니다.
AdviseSink ITextStoreACPSink 인터페이스에서 새 advise 싱크를 설치하거나 기존 조언 싱크를 수정합니다. 싱크 인터페이스는 펑크 매개 변수에 의해 지정됩니다.
AdviseSink ITextStoreAnchor::AdviseSink 메서드는 ITextStoreAnchorSink 인터페이스에서 새 advise 싱크를 설치하거나 기존 조언 싱크를 수정합니다.
AdviseSink ITfSource::AdviseSink 메서드
AssociateFocus ITfThreadMgr::AssociateFocus 메서드
BeginUIElement ITfUIElementMgr::BeginUIElement 메서드는 UI를 표시하기 전에 텍스트 서비스에서 호출됩니다. 반환된 값은 텍스트 서비스의 UI를 표시할지 여부를 결정합니다.
BeginUIElement ITfUIElementSink::BeginUIElement 메서드는 UIElement가 시작될 때 호출됩니다. 이 싱크는 텍스트 서비스가 UI 요소를 그리거나 그릴 수 없도록 할 수 있습니다.

ChangeCurrentLanguage ITfInputProcessorProfiles::ChangeCurrentLanguage 메서드
지우기 ITfProperty::Clear 메서드
ClearChangeHistory IAnchor::ClearChangeHistory 메서드는 앵커 변경 기록 플래그를 지웁니다.
ClearCompartment ITfCompartmentMgr::ClearCompartment 메서드
복제 IAnchor::Clone 메서드는 현재 앵커와 동일한 위치와 동일한 중력에 배치된 새 앵커 개체를 생성합니다.
복제 IEnumITfCompositionView::Clone 메서드
복제 IEnumTfContexts::Clone 메서드
복제 IEnumTfDisplayAttributeInfo::Clone 메서드
복제 IEnumTfDocumentMgrs::Clone 메서드
복제 IEnumTfFunctionProviders::Clone 메서드
복제 IEnumTfInputProcessorProfiles::Clone 메서드는 열거자 개체의 복사본을 만듭니다.
복제 IEnumTfLanguageProfiles::Clone 메서드

복제 IEnumTfProperties::Clone 메서드
복제 IEnumTfPropertyValue::Clone 메서드
복제 IEnumTfRanges::Clone 메서드
복제 IEnumTfUIElements::Clone 메서드는 열거자 개체의 복사본을 만듭니다.
복제 ITfPropertyStore::Clone 메서드
복제 ITfRange::Clone 메서드는 이 텍스트 범위를 복제합니다.
복제 IEnumTfCandidates::Clone 메서드
복제 IEnumTfLatticeElements::Clone 메서드
복제 IEnumTfLangBarItems::Clone 메서드
축소 ITfRange::Collapse 메서드는 시작 앵커와 끝 앵커를 동일한 위치로 이동하여 텍스트 범위를 지웁니다.
비교 IAnchor::Compare 메서드는 텍스트 스트림 내에서 두 앵커의 상대 위치를 비교합니다.
CompareEnd ITfRange::CompareEnd 메서드는 이 텍스트 범위의 끝 앵커 위치를 다른 범위의 앵커와 비교합니다.

CompareStart ITfRange::CompareStart 메서드는 이 텍스트 범위의 시작 앵커 위치를 다른 범위의 앵커와 비교합니다.
CreateContext ITfDocumentMgr::CreateContext 메서드
CreateDocumentMgr ITfThreadMgr::CreateDocumentMgr 메서드
CreateDocumentMgr 문서 관리자 개체를 만듭니다.
CreatePropertyStore ITfCreatePropertyStore::CreatePropertyStore 메서드
CreateRange ITextStoreACPServices::CreateRange 메서드
CreateRange ITfContextOwnerServices::CreateRange 메서드는 지정된 문자 위치에 따라 범위가 지정된 새 범위를 만듭니다.
CreateRangeBackup ITfContext::CreateRangeBackup 메서드
비활성화 ITfTextInputProcessor::D eactivate 메서드
비활성화 ITfThreadMgr::D eactivate 메서드
비활성화 호출 스레드에 대한 TSF를 비활성화합니다.

DeactivateProfile ITfInputProcessorProfileMgr::D eactivateProfile 메서드는 지정된 텍스트 서비스의 프로필 또는 키보드 레이아웃을 비활성화합니다.
DisableSystemKeystrokeFeed ITfConfigureSystemKeystrokeFeed::D isableSystemKeystrokeFeed 메서드
나누기 ITfPropertyStore::D ivide 메서드
DoEditSession ITfEditSession::D oEditSession 메서드
DoReverseConversion 지정된 문자열의 역방향 변환을 수행합니다.
DrawBitmap ITfLangBarItemBitmap::D rawBitmap 메서드
DrawBitmap ITfLangBarItemBitmapButton::D rawBitmap 메서드
EnableLanguageProfile ITfInputProcessorProfiles::EnableLanguageProfile 메서드
EnableLanguageProfileByDefault ITfInputProcessorProfiles::EnableLanguageProfileByDefault 메서드
EnableSystemKeystrokeFeed ITfConfigureSystemKeystrokeFeed::EnableSystemKeystrokeFeed 메서드
EndComposition ITfComposition::EndComposition 메서드
EndUIElement UI 요소가 숨겨지면 텍스트 서비스에서 ITfUIElementMgr::EndUIElement 메서드를 호출합니다.

EndUIElement UIElement가 완료되면 ITfUIElementSink::EndUIElement 메서드가 호출됩니다.
EnumCandidates ITfCandidateList::EnumCandidates 메서드
EnumCategoriesInItem ITfCategoryMgr::EnumCategoriesInItem 메서드
EnumCompartments ITfCompartmentMgr::EnumCompartments 메서드는 구획 관리자 내 구획의 GUID를 포함하는 열거자를 가져옵니다.
EnumCompositions ITfContextComposition::EnumCompositions 메서드
EnumContexts ITfDocumentMgr::EnumContexts 메서드
EnumDisplayAttributeInfo ITfDisplayAttributeMgr::EnumDisplayAttributeInfo 메서드
EnumDisplayAttributeInfo ITfDisplayAttributeProvider::EnumDisplayAttributeInfo 메서드
EnumDocumentMgrs ITfThreadMgr::EnumDocumentMgrs 메서드
EnumDocumentMgrs 호출 스레드 내의 모든 문서 관리자에 대한 열거자를 반환합니다.
EnumFunctionProviders ITfThreadMgr::EnumFunctionProviders 메서드
EnumFunctionProviders 호출 스레드에 등록된 모든 함수 공급자에 대한 열거자를 가져옵니다.

EnumInputProcessorInfo ITfInputProcessorProfiles::EnumInputProcessorInfo 메서드
EnumItems ITfLangBarItemMgr::EnumItems 메서드
EnumItemsInCategory ITfCategoryMgr::EnumItemsInCategory 메서드
EnumLanguageProfiles ITfInputProcessorProfiles::EnumLanguageProfiles 메서드
EnumLatticeElements ITfLMLattice::EnumLatticeElements 메서드
EnumProfiles ITfInputProcessorProfileMgr::EnumProfiles 메서드는 열거할 프로필을 반환합니다.
EnumProperties ITfContext::EnumProperties 메서드
EnumRanges ITfReadOnlyProperty::EnumRanges 메서드
EnumUIElements ITfUIElementMgr::EnumUIElements 메서드는 IEnumTfUIElements 인터페이스 포인터를 반환하여 ITfUIElement를 열거합니다.
EnumWordList ITfInputScope2::EnumWordList 메서드
Finalize ITfCandidateListUIElementBehavior::Finalize 메서드는 현재 선택을 마무리하고 후보 목록을 닫습니다.
FinalizeExactCompositionString 현재 사용자에게 표시된 값을 사용하여 현재 컴퍼지션을 마무리합니다.

FindClosestCategory ITfCategoryMgr::FindClosestCategory 메서드
FindComposition ITfContextComposition::FindComposition 메서드
FindNextAttrTransition ITextStoreACP::FindNextAttrTransition 메서드는 특성 값에서 전환이 발생하는 문자 위치를 결정합니다. 검사할 지정된 특성이 애플리케이션에 종속되어 있습니다.
FindNextAttrTransition 특성 값에서 전환이 발생하는 문자 위치를 결정합니다. 검사할 지정된 특성이 애플리케이션에 종속되어 있습니다.
FindNextAttrTransition ITextStoreAnchor::FindNextAttrTransition 메서드는 특성 값에서 전환이 발생하는 텍스트 스트림의 위치를 찾습니다. 검사할 지정된 특성이 애플리케이션에 종속되어 있습니다.
FindRange ITfProperty::FindRange 메서드
ForceLoadProperty ITextStoreACPServices::ForceLoadProperty 메서드
ForceLoadProperty ITfContextOwnerServices::ForceLoadProperty 메서드
가져오기 ITfFnGetSAPIObject::Get 메서드
GetACPFromPoint ITextStoreACP::GetACPFromPoint 메서드는 화면 좌표의 점을 애플리케이션 문자 위치로 변환합니다.
GetACPFromPoint 화면의 점 좌표를 애플리케이션 문자 위치로 변환합니다.

GetACPFromPoint ITfContextOwner::GetACPFromPoint 메서드는 화면 좌표의 점을 애플리케이션 문자 위치로 변환합니다.
GetActiveFlags 호출 스레드의 활성 플래그를 가져옵니다.
GetActiveFlags ITfThreadMgrEx::GetActiveFlags 메서드는 TSF가 활성 상태인 플래그를 반환합니다.
GetActiveLanguageProfile ITfInputProcessorProfiles::GetActiveLanguageProfile 메서드
GetActiveProfile 이 메서드는 현재 활성 프로필을 반환합니다.
GetActiveView ITextStoreACP::GetActiveView 메서드는 현재 활성 보기를 지정하는 TsViewCookie 데이터 형식을 반환합니다.
GetActiveView 현재 활성 보기를 나타내는 TsViewCookie를 가져옵니다.
GetActiveView ITextStoreAnchor::GetActiveView 메서드는 현재 활성 보기를 지정하는 TsViewCookie 데이터 형식을 반환합니다. TSF는 단일 활성 보기만 지원하므로 지정된 텍스트 저장소는 항상 동일한 TsViewCookie 데이터 형식을 반환해야 합니다.
GetActiveView ITfContext::GetActiveView 메서드
GetAlternates 지정된 텍스트 범위에 대한 대체 문자열 목록을 반환합니다.
GetAnchorFromPoint ITextStoreAnchor::GetAnchorFromPoint 메서드는 화면 좌표의 점을 해당 위치에 배치된 앵커로 변환합니다.

GetAppProperty ITfContext::GetAppProperty 메서드
GetAttribute ITfContextOwner::GetAttribute 메서드는 지원되는 특성의 값을 반환합니다. 특성이 지원되지 않으면 pvarValue 매개 변수가 VT_EMPTY 설정됩니다.
GetAttributeInfo ITfDisplayAttributeInfo::GetAttributeInfo 메서드
GetBalloonInfo ITfLangBarItemBalloon::GetBalloonInfo 메서드
GetBase ITfDocumentMgr::GetBase 메서드
GetCandidate ITfCandidateList::GetCandidate 메서드
GetCandidateNum ITfCandidateList::GetCandidateNum 메서드
GetChangeHistory IAnchor::GetChangeHistory 메서드는 앵커 바로 앞이나 뒤에 발생한 삭제 기록을 가져옵니다.
GetClientId ITfClientId::GetClientId 메서드
GetCompartment ITfCompartmentMgr::GetCompartment 메서드
GetContext ITfRange::GetContext 메서드
GetContext 이 메서드는 이 읽기 정보 UI의 대상 ITfContext를 반환합니다.

GetContext ITfReadOnlyProperty::GetContext 메서드
GetCount ITfCandidateListUIElement::GetCount 메서드는 후보 문자열의 수를 반환합니다.
GetCurrentLanguage ITfInputProcessorProfiles::GetCurrentLanguage 메서드
GetCurrentPage ITfCandidateListUIElement::GetCurrentPage 메서드는 현재 페이지를 반환합니다.
GetData ITfPropertyStore::GetData 메서드
GetDataType ITfPropertyStore::GetDataType 메서드
GetDefaultLanguageProfile ITfInputProcessorProfiles::GetDefaultLanguageProfile 메서드
GetDescription ITfDisplayAttributeInfo::GetDescription 메서드
GetDescription ITfFunctionProvider::GetDescription 메서드
GetDescription ITfUIElement::GetDescription 메서드는 UI 요소에 대한 설명을 반환합니다.
GetDisplayAttributeInfo ITfDisplayAttributeMgr::GetDisplayAttributeInfo 메서드
GetDisplayAttributeInfo ITfDisplayAttributeProvider::GetDisplayAttributeInfo 메서드

GetDisplayName ITfFunction::GetDisplayName 메서드
GetDocumentMgr ITfCandidateListUIElement::GetDocumentMgr 메서드는 이 UI의 대상 문서 관리자를 반환합니다.
GetDocumentMgr ITfContext::GetDocumentMgr 메서드
GetDocumentMgr ITfTransitoryExtensionUIElement::GetDocumentMgr 메서드는 일시적인 문서 관리자의 포인터를 반환합니다.
GetEmbedded 포함된 문서를 가져옵니다. (ITextStoreACP.GetEmbedded)
GetEmbedded 포함된 문서를 가져옵니다. (ITextStoreACP2.GetEmbedded)
GetEmbedded ITextStoreAnchor::GetEmbedded 메서드는 텍스트 스트림에서 포함된 개체를 가져옵니다.
GetEmbedded ITfRange::GetEmbedded 메서드는 텍스트 스트림의 TS_CHAR_EMBEDDED 문자에 해당하는 콘텐츠를 가져옵니다. 텍스트 범위의 시작 앵커는 관심 있는 문자 바로 앞에 배치됩니다.
GetEnd ITextStoreAnchor::GetEnd 메서드는 텍스트 스트림의 끝에 배치된 앵커를 반환합니다.
GetEnd ITfContext::GetEnd 메서드
GetEndACP ITextStoreACP::GetEndACP 메서드는 문서의 문자 수를 반환합니다.
GetEndACP 문서의 문자 수를 가져옵니다.

GetErrorIndex 이 메서드는 입력 오류가 발생하는 char 인덱스를 반환합니다.
GetExtent ITfRangeACP::GetExtent 메서드
GetFocus ITfThreadMgr::GetFocus 메서드
GetFocus 입력 포커스가 있는 문서 관리자를 반환합니다.
GetForeground ITfKeystrokeMgr::GetForeground 메서드
GetFormattedText ITextStoreACP::GetFormattedText 메서드는 지정된 텍스트 문자열에 대한 서식이 지정된 텍스트 데이터를 반환합니다. 호출자는 이 메서드를 호출하기 전에 문서에 대한 읽기/쓰기 잠금이 있어야 합니다.
GetFormattedText 지정된 텍스트 문자열에 대한 서식이 지정된 텍스트 데이터를 가져옵니다. 호출자는 이 메서드를 호출하기 전에 문서에 대한 읽기/쓰기 잠금이 있어야 합니다.
GetFormattedText ITextStoreAnchor::GetFormattedText 메서드는 텍스트 스트림에서 서식이 지정된 텍스트 정보를 반환합니다.
GetFormattedText ITfRange::GetFormattedText 메서드는 텍스트 범위 내에 포함된 서식이 지정된 콘텐츠를 가져옵니다. 콘텐츠는 IDataObject 인터페이스를 지원하는 개체에 패키지됩니다.
GetFunction ITfFunctionProvider::GetFunction 메서드
GetFunctionProvider ITfThreadMgr::GetFunctionProvider 메서드

GetFunctionProvider 지정된 함수 공급자 개체를 가져옵니다.
GetGlobalCompartment ITfThreadMgr::GetGlobalCompartment 메서드
GetGlobalCompartment 전역 구획 관리자 개체를 가져옵니다.
GetGravity IAnchor::GetGravity 메서드는 IAnchor 개체에서 앵커의 중력을 검색합니다.
GetGravity ITfRange::GetGravity 메서드
GetGUID ITfCategoryMgr::GetGUID 메서드
GetGUID ITfDisplayAttributeInfo::GetGUID 메서드
GetGUID ITfUIElement::GetGUID 메서드는 이 UI 요소의 고유 ID를 반환합니다.
GetGUIDDescription ITfCategoryMgr::GetGUIDDescription 메서드
GetGUIDDWORD ITfCategoryMgr::GetGUIDDWORD 메서드
GetIcon ITfLangBarItemButton::GetIcon 메서드
GetIconMode ITfSystemDeviceTypeLangBarItem::GetIconMode 메서드

GetIndex ITfCandidateString::GetIndex 메서드
GetInfo ITfLangBarItem::GetInfo 메서드
GetInputProcessorProfiles ITfLangBarMgr::GetInputProcessorProfiles 메서드
GetInputScopes ITfInputScope::GetInputScopes 메서드
GetItem ITfLangBarItemMgr::GetItem 메서드
GetItemFloatingRect ITfLangBarEventSink::GetItemFloatingRect 메서드
GetItemFloatingRect ITfLangBarItemMgr::GetItemFloatingRect 메서드
GetItemNum ITfLangBarItemMgr::GetItemNum 메서드
GetItems ITfLangBarItemMgr::GetItems 메서드
GetItemsStatus ITfLangBarItemMgr::GetItemsStatus 메서드
GetItemText ITfSystemLangBarItemText::GetItemText 메서드는 시스템 언어 모음 메뉴에 표시되는 텍스트를 가져옵니다.
GetLanguageList ITfInputProcessorProfiles::GetLanguageList 메서드

GetLanguageProfileDescription ITfInputProcessorProfiles::GetLanguageProfileDescription 메서드
GetLayout IME가 활성화되어 있는 동안 IME가 터치 키보드를 표시하도록 지시하는 레이아웃의 터치 키보드 레이아웃 식별자를 가져옵니다.
GetLength 목록에서 키 입력 시퀀스 수를 검색합니다.
GetMaxReadingStringLength ITfReadingInformationUIElement::GetMaxReadingStringLength 메서드는 읽기 정보 UI의 최대 문자열 수를 반환합니다.
GetMessageA ITfMessagePump::GetMessageA 메서드
GetMessageW ITfMessagePump::GetMessageW 메서드
GetOwnerClsid ITfCompositionView::GetOwnerClsid 메서드
GetPageIndex ITfCandidateListUIElement::GetPageIndex 메서드는 목록의 페이지 인덱스를 반환합니다.
GetPhrase ITfInputScope::GetPhrase 메서드
GetPreferredSize ITfLangBarItemBalloon::GetPreferredSize 메서드
GetPreferredSize ITfLangBarItemBitmap::GetPreferredSize 메서드
GetPreferredSize ITfLangBarItemBitmapButton::GetPreferredSize 메서드

GetPreservedKey ITfKeystrokeMgr::GetPreservedKey 메서드
GetPreservedKeyDescription ITfKeystrokeMgr::GetPreservedKeyDescription 메서드
Getprofile ITfInputProcessorProfileMgr::GetProfile 메서드는 지정된 텍스트 서비스의 프로필 또는 키보드 레이아웃에 대한 정보를 TF_INPUTPROCESSORPROFILE 구조체로 반환합니다.
Getproperty ITfContext::GetProperty 메서드
GetPropertyRangeCreator ITfPropertyStore::GetPropertyRangeCreator 메서드
GetRange ITfComposition::GetRange 메서드
GetRange ITfCompositionView::GetRange 메서드
GetRangeFromPoint ITfContextView::GetRangeFromPoint 메서드는 점(화면 좌표)을 해당 위치에 배치된 빈 텍스트 범위로 변환합니다.
GetReconversion ITfFnLMProcessor::GetReconversion 메서드
GetReconversion ITfFnReconversion::GetReconversion 메서드
GetRegularExpression ITfInputScope::GetRegularExpression 메서드
GetReverseConversion 역방향 변환을 수행할 수 있는 ITfReverseConversion 개체를 검색합니다.

GetScreenExt ITextStoreACP::GetScreenExt 메서드는 텍스트 스트림이 렌더링되는 디스플레이 화면의 경계 상자 화면 좌표를 반환합니다.
GetScreenExt 텍스트 스트림이 렌더링되는 디스플레이 화면의 경계 상자 화면 좌표를 가져옵니다.
GetScreenExt ITextStoreAnchor::GetScreenExt 메서드는 텍스트 스트림이 렌더링되는 디스플레이 화면의 경계 상자 화면 좌표를 반환합니다.
GetScreenExt ITfContextOwner::GetScreenExt 메서드는 텍스트 스트림이 렌더링되는 디스플레이 화면의 경계 상자를 화면 좌표로 반환합니다.
GetScreenExt ITfContextView::GetScreenExt 메서드는 문서 디스플레이의 경계 상자를 화면 좌표로 반환합니다.
GetSearchCandidates IME 관련 메시지 또는 이벤트를 생성하지 않고 지정된 문자열에 대한 변환 후보 목록을 가져옵니다.
GetSelection ITextStoreACP::GetSelection 메서드는 문서에서 텍스트 선택 영역의 문자 위치를 반환합니다. 이 메서드는 여러 텍스트 선택을 지원합니다. 호출자는 이 메서드를 호출하기 전에 문서에 대한 읽기 전용 잠금이 있어야 합니다.
GetSelection 문서에서 텍스트 선택 영역의 문자 위치를 가져옵니다. 이 메서드는 여러 텍스트 선택을 지원합니다. 호출자는 이 메서드를 호출하기 전에 문서에 대한 읽기 전용 잠금이 있어야 합니다.
GetSelection ITextStoreAnchor::GetSelection 메서드는 텍스트 스트림에서 텍스트 선택 영역의 오프셋을 반환합니다. 이 메서드는 여러 텍스트 선택을 지원합니다. 호출자는 이 메서드를 호출하기 전에 문서에 대한 읽기 전용 잠금이 있어야 합니다.
GetSelection ITfCandidateListUIElement::GetSelection 메서드는 후보 목록의 현재 선택 항목을 반환합니다.

GetSelection ITfContext::GetSelection 메서드
GetSelectionStatus ITfEditRecord::GetSelectionStatus 메서드
GetSelectionStyle 선택 스타일을 검색합니다.
GetShowFloatingStatus ITfLangBarMgr::GetShowFloatingStatus 메서드
GetSRGS ITfInputScope::GetSRGS 메서드
GetStart ITextStoreAnchor::GetStart 메서드는 텍스트 스트림의 시작 부분에 배치된 앵커를 반환합니다.
GetStart ITfContext::GetStart 메서드
GetStatus ITextStoreACP::GetStatus 메서드는 문서 상태를 가져옵니다. 문서 상태는 TS_STATUS 구조를 통해 반환됩니다.
GetStatus 문서 상태를 가져옵니다. 문서 상태는 TS_STATUS 구조를 통해 반환됩니다.
GetStatus ITextStoreAnchor::GetStatus 메서드는 문서 상태를 가져옵니다. 문서 상태는 TS_STATUS 구조를 통해 반환됩니다.
GetStatus ITfContext::GetStatus 메서드

GetStatus ITfContextOwner::GetStatus 메서드는 문서의 상태를 가져옵니다. 문서 상태는 TS_STATUS 구조를 통해 반환됩니다.
GetStatus ITfFnPropertyUIStatus::GetStatus 메서드
GetStatus ITfLangBarItem::GetStatus 메서드
GetString ITfCandidateListUIElement::GetString 메서드는 인덱스의 문자열을 반환합니다.
GetString 이 메서드는 읽기 정보 UI의 문자열을 반환합니다.
GetString 지정된 인덱스에서 키 입력 시퀀스를 검색합니다.
GetString 도구 설명의 문자열을 반환합니다.
GetString ITfCandidateString::GetString 메서드
GetSubstituteKeyboardLayout ITfInputProcessorProfileSubstituteLayout::GetSubstituteKeyboardLayout 메서드
GetText ITextStoreACP::GetText 메서드는 지정된 문자 위치에 있는 텍스트에 대한 정보를 반환합니다. 이 메서드는 표시 및 숨겨진 텍스트를 반환하고 포함된 데이터가 텍스트에 연결되어 있는지를 나타냅니다.
GetText 지정된 문자 위치에 있는 텍스트에 대한 정보를 가져옵니다. 이 메서드는 표시 및 숨겨진 텍스트를 반환하고 포함된 데이터가 텍스트에 연결되어 있는지를 나타냅니다.

GetText ITextStoreAnchor::GetText 메서드는 지정된 앵커 위치에서 텍스트에 대한 정보를 반환합니다. 이 메서드는 표시 및 숨겨진 텍스트를 반환하고 포함된 데이터가 텍스트에 연결되어 있는지를 나타냅니다.
GetText ITfRange::GetText 메서드는 이 텍스트 범위에서 다루는 콘텐츠를 가져옵니다.
GetText ITfLangBarItemBitmapButton::GetText 메서드
GetText ITfLangBarItemButton::GetText 메서드
GetTextAndPropertyUpdates ITfEditRecord::GetTextAndPropertyUpdates 메서드
GetTextExt ITextStoreACP::GetTextExt 메서드는 지정된 문자 위치에 있는 텍스트의 경계 상자를 화면 좌표로 반환합니다. 호출자는 이 메서드를 호출하기 전에 문서에 대한 읽기 전용 잠금이 있어야 합니다.
GetTextExt 지정된 문자 위치에 있는 텍스트의 경계 상자를 화면 좌표로 가져옵니다. 호출자는 이 메서드를 호출하기 전에 문서에 대한 읽기 전용 잠금이 있어야 합니다.
GetTextExt ITextStoreAnchor::GetTextExt 메서드는 텍스트 범위의 경계 상자를 화면 좌표로 반환합니다. 호출자는 이 메서드를 호출하기 전에 문서에 대한 읽기 전용 잠금이 있어야 합니다.
GetTextExt ITfContextOwner::GetTextExt 메서드는 지정된 문자 위치에 있는 텍스트의 경계 상자를 화면 좌표로 반환합니다. 호출자는 이 메서드를 호출하기 전에 문서에 대한 읽기 전용 잠금이 있어야 합니다.
GetTextExt ITfContextView::GetTextExt 메서드는 텍스트 범위의 경계 상자를 화면 좌표로 반환합니다.
GetThreadLangBarItemMgr ITfLangBarMgr::GetThreadLangBarItemMgr 메서드

GetThreadMarshalInterface ITfLangBarMgr::GetThreadMarshalInterface 메서드
GetTooltipString ITfLangBarItem::GetTooltipString 메서드
GetTop ITfDocumentMgr::GetTop 메서드
GetType ITfFunctionProvider::GetType 메서드
GetType ITfPropertyStore::GetType 메서드
GetType ITfReadOnlyProperty::GetType 메서드
GetUIElement ITfUIElementMgr::GetUIElement 메서드는 요소 ID의 ITfUIElement 인터페이스를 가져옵니다.
GetUpdatedFlags ITfCandidateListUIElement::GetUpdatedFlags 메서드는 이 요소의 업데이트된 부분을 알려주는 플래그를 반환합니다.
GetUpdatedFlags 이 메서드는 이 요소의 업데이트된 부분을 알려주는 플래그를 반환합니다.
GetValue ITfCompartment::GetValue 메서드
GetValue ITfReadOnlyProperty::GetValue 메서드
GetWnd ITextStoreACP::GetWnd 메서드는 현재 문서에 해당하는 창에 핸들을 반환합니다.

GetWnd ITextStoreAnchor::GetWnd 메서드는 현재 텍스트 스트림에 해당하는 창에 핸들을 반환합니다.
GetWnd ITfContextOwner::GetWnd 메서드는 핸들을 현재 문서에 해당하는 창으로 반환합니다.
GetWnd ITfContextView::GetWnd 메서드는 현재 문서에 해당하는 창에 핸들을 반환합니다.
Getxml ITfInputScope::GetXML 메서드
초기화 ITfSpeechUIServer::Initialize 메서드
InitLocalMsCtfMonitor InitLocalMsCtfMonitor 함수는 현재 데스크톱에서 TextServicesFramework를 초기화하고 필요한 경우 부동 언어 모음을 준비합니다. 이 함수는 앱의 데스크톱에서 호출되어야 합니다.
InitMenu 단추 항목에 TF_LBI_STYLE_BTN_MENU 스타일이 없는 경우 이 메서드는 사용되지 않습니다. (ITfLangBarItemBitmapButton.InitMenu)
InitMenu 단추 항목에 TF_LBI_STYLE_BTN_MENU 스타일이 없는 경우 이 메서드는 사용되지 않습니다. (ITfLangBarItemButton.InitMenu)
InitMenu ITfSystemLangBarItemSink::InitMenu 메서드
InsertEmbedded 지정된 문자에 포함된 개체를 삽입합니다. (ITextStoreACP.InsertEmbedded)
InsertEmbedded 지정된 문자에 포함된 개체를 삽입합니다. (ITextStoreACP2.InsertEmbedded)

InsertEmbedded ITextStoreAnchor::InsertEmbedded 메서드
InsertEmbedded ITfRange::InsertEmbedded 메서드는 텍스트 범위의 시작 앵커 위치에 개체를 삽입합니다.
InsertEmbeddedAtSelection ITextStoreACP::InsertEmbeddedAtSelection 메서드는 삽입 지점 또는 선택 영역에 IDataObject 개체를 삽입합니다. 이 메서드를 호출하는 클라이언트는 문서에 IDataObject 개체를 삽입하기 전에 읽기/쓰기 잠금이 있어야 합니다.
InsertEmbeddedAtSelection 삽입 지점 또는 선택 영역에 IDataObject를 삽입합니다. 이 메서드를 호출하는 클라이언트는 문서에 IDataObject 개체를 삽입하기 전에 읽기/쓰기 잠금이 있어야 합니다.
InsertEmbeddedAtSelection ITextStoreAnchor::InsertEmbeddedAtSelection 메서드는 삽입 지점 또는 선택 영역에 IDataObject 개체를 삽입합니다. 이 메서드를 호출하는 클라이언트는 텍스트 스트림에 IDataObject를 삽입하기 전에 읽기/쓰기 잠금이 있어야 합니다.
InsertEmbeddedAtSelection ITfInsertAtSelection::InsertEmbeddedAtSelection 메서드는 선택 영역 또는 삽입 지점에 IDataObject 개체를 삽입합니다.
InsertTextAtSelection ITextStoreACP::InsertTextAtSelection 메서드는 삽입 지점 또는 선택 영역에 텍스트를 삽입합니다. 호출자는 텍스트를 삽입하기 전에 문서에 대한 읽기/쓰기 잠금이 있어야 합니다.
InsertTextAtSelection 삽입 지점 또는 선택 영역에 텍스트를 삽입합니다. 호출자는 텍스트를 삽입하기 전에 문서에 대한 읽기/쓰기 잠금이 있어야 합니다.
InsertTextAtSelection ITextStoreAnchor::InsertTextAtSelection 메서드
InsertTextAtSelection ITfInsertAtSelection::InsertTextAtSelection 메서드

InvokeFunc ITfFnLMProcessor::InvokeFunc 메서드
InvokeKey ITfFnLMProcessor::InvokeKey 메서드
InWriteSession ITfContext::InWriteSession 메서드
IsEmpty ITfRange::IsEmpty 메서드는 시작 및 끝 앵커가 동일한 위치를 차지하기 때문에 텍스트 범위가 비어 있는지 확인합니다.
IsEnabledLanguageProfile ITfInputProcessorProfiles::IsEnabledLanguageProfile 메서드
IsEqual IAnchor::IsEqual 메서드는 텍스트 스트림 내에서 두 개의 앵커를 평가하고 앵커 위치의 같음 또는 같지 않음을 지정하는 부울 값을 반환합니다.
IsEqualEnd ITfRange::IsEqualStart 메서드는 이 텍스트 범위의 끝 앵커가 지정된 다른 범위의 앵커와 일치하는지 확인합니다.
IsEqualStart ITfRange::IsEqualStart 메서드는 이 텍스트 범위의 시작 앵커가 지정된 다른 범위의 앵커와 일치하는지 확인합니다.
IsEqualTfGuidAtom ITfCategoryMgr::IsEqualTfGuidAtom 메서드
IsPreservedKey ITfKeystrokeMgr::IsPreservedKey 메서드
IsProfileAvailableForLang ITfFnLangProfileUtil::IsProfileAvailableForLang 메서드

IsShown ITfUIElement::IsShown 메서드는 현재 텍스트 서비스에서 UI를 표시하는 경우 true를 반환합니다. 그렇지 않으면 false입니다.
IsStoreSerializable ITfCreatePropertyStore::IsStoreSerializable 메서드
IsThreadFocus ITfThreadMgr::IsThreadFocus 메서드
IsThreadFocus 호출 스레드에 TSF 입력 포커스가 있는지 여부를 확인합니다.
IsVerticalOrderP 참조 이 메서드는 UI를 세로 순서로 표시하려는 경우 반환합니다.
KeyDown ITfKeystrokeMgr::KeyDown 메서드
KeyUp ITfKeystrokeMgr::KeyUp 메서드
LoadProperty ITfPersistentPropertyLoaderACP::LoadProperty 메서드
다음 IEnumITfCompositionView::Next 메서드
다음 IEnumTfContexts::Next 메서드
다음 IEnumTfDisplayAttributeInfo::Next 메서드
다음 IEnumTfDocumentMgrs::Next 메서드

다음 IEnumTfFunctionProviders::Next 메서드
다음 IEnumTfInputProcessorProfiles::Next 메서드는 현재 위치에서 열거형 시퀀스의 지정된 요소 수를 가져옵니다.
다음 IEnumTfLanguageProfiles::Next 메서드
다음 IEnumTfProperties::Next 메서드
다음 IEnumTfPropertyValue::Next 메서드
다음 IEnumTfRanges::Next 메서드
다음 IEnumTfUIElements::Next 메서드는 현재 위치에서 열거형 시퀀스의 지정된 요소 수를 가져옵니다.
다음 IEnumTfCandidates::Next 메서드
다음 IEnumTfLatticeElements::Next 메서드
다음 IEnumTfLangBarItems::Next 메서드
OnActivated ITfActiveLanguageProfileNotifySink::OnActivated 메서드
OnActivated ITfInputProcessorProfileActivationSink::OnActivated 메서드는 입력 프로세서 프로필이 활성화되거나 비활성화될 때 호출되는 콜백입니다.

OnAttributeChange ITfContextOwnerServices::OnAttributeChange 메서드
OnAttrsChange ITextStoreACPSink::OnAttrsChange 메서드
OnAttrsChange 하나 이상의 텍스트 특성 값이 변경되면 ITextStoreAnchorSink::OnAttrsChange 메서드가 호출됩니다.
Onchange ITfCompartmentEventSink::OnChange 메서드
OnCleanupContext ITfCleanupContextSink::OnCleanupContext 메서드
Onclick ITfLangBarItemBalloon::OnClick 메서드
Onclick ITfLangBarItemBitmap::OnClick 메서드
Onclick 단추 항목에 TF_LBI_STYLE_BTN_BUTTON 스타일이 없는 경우이 메서드를 사용 하지 않습니다. (ITfLangBarItemBitmapButton.OnClick)
Onclick 단추 항목에 TF_LBI_STYLE_BTN_BUTTON 스타일이 없는 경우이 메서드를 사용 하지 않습니다. (ITfLangBarItemButton.OnClick)
OnCompositionTerminated ITfCompositionSink::OnCompositionTerminated 메서드
OnEndCleanupContext ITfCleanupContextDurationSink::OnEndCleanupContext 메서드

OnEndComposition ITfContextOwnerCompositionSink::OnEndComposition 메서드
OnEndEdit ITfTextEditSink::OnEndEdit 메서드
OnEndEditTransaction ITextStoreACPSink::OnEndEditTransaction 메서드
OnEndEditTransaction ITextStoreAnchorSink::OnEndEditTransaction 메서드
OnEndEditTransaction ITfEditTransactionSink::OnEndEditTransaction 메서드
OnInitDocumentMgr ITfThreadMgrEventSink::OnInitDocumentMgr 메서드
OnKeyDown ITfContextKeyEventSink::OnKeyDown 메서드
OnKeyDown ITfKeyEventSink::OnKeyDown 메서드
OnKeyDown 키 누르기를 처리합니다.
OnKeyTraceDown ITfKeyTraceEventSink::OnKeyTraceDown 메서드
OnKeyTraceUp ITfKeyTraceEventSink::OnKeyTraceUp 메서드
OnKeyUp ITfContextKeyEventSink::OnKeyUp 메서드

OnKeyUp ITfKeyEventSink::OnKeyUp 메서드
OnKillThreadFocus ITfThreadFocusSink::OnKillThreadFocus 메서드
OnLanguageChange ITfLanguageProfileNotifySink::OnLanguageChange 메서드
OnLanguageChanged ITfLanguageProfileNotifySink::OnLanguageChanged 메서드
OnLatticeUpdate ITfFnAdviseText::OnLatticeUpdate 메서드
OnLayoutChange ITextStoreACPSink::OnLayoutChange 메서드
OnLayoutChange ITextStoreAnchorSink::OnLayoutChange 메서드는 문서의 레이아웃(화면 표현)이 변경되면 호출됩니다.
OnLayoutChange ITfContextOwnerServices::OnLayoutChange 메서드는 컴퍼지션 중에 텍스트 스트림의 화면 표현이 업데이트되면 컨텍스트 소유자에 의해 호출됩니다.
OnLayoutChange ITfTextLayoutSink::OnLayoutChange 메서드
OnLockGranted ITextStoreACPSink::OnLockGranted 메서드
OnLockGranted ITextStoreAnchorSink::OnLockGranted 메서드

OnMenuSelect 단추 항목에 TF_LBI_STYLE_BTN_MENU 스타일이 없는 경우이 메서드를 사용 하지 않습니다. (ITfLangBarItemBitmapButton.OnMenuSelect)
OnMenuSelect 단추 항목에 TF_LBI_STYLE_BTN_MENU 스타일이 없는 경우이 메서드를 사용 하지 않습니다. (ITfLangBarItemButton.OnMenuSelect)
OnMenuSelect ITfSystemLangBarItemSink::OnMenuSelect 메서드
OnModalInput ITfLangBarEventSink::OnModalInput 메서드
OnMouseEvent ITfMouseSink::OnMouseEvent 메서드
OnPopContext ITfThreadMgrEventSink::OnPopContext 메서드
OnPreservedKey ITfKeyEventSink::OnPreservedKey 메서드
OnPushContext ITfThreadMgrEventSink::OnPushContext 메서드
OnSelectionChange ITextStoreACPSink::OnSelectionChange 메서드
OnSelectionChange 텍스트 스트림 내의 선택 영역이 변경되면 ITextStoreAnchorSink::OnSelectionChange 메서드가 호출됩니다. ITextStoreAnchor::GetSelection에 대한 잠재적 호출의 반환 값이 변경될 때마다 이 메서드를 호출해야 합니다.
OnSetFocus ITfKeyEventSink::OnSetFocus 메서드

OnSetFocus ITfThreadMgrEventSink::OnSetFocus 메서드
OnSetFocus ITfLangBarEventSink::OnSetFocus 메서드
OnSetThreadFocus ITfThreadFocusSink::OnSetThreadFocus 메서드
OnStartCleanupContext ITfCleanupContextDurationSink::OnStartCleanupContext 메서드
OnStartComposition ITfContextOwnerCompositionSink::OnStartComposition 메서드
OnStartEditTransaction ITextStoreACPSink::OnStartEditTransaction 메서드
OnStartEditTransaction ITextStoreAnchorSink::OnStartEditTransaction 메서드
OnStartEditTransaction ITfEditTransactionSink::OnStartEditTransaction 메서드
OnStatusChange ITextStoreACPSink::OnStatusChange 메서드
OnStatusChange ITextStoreAnchorSink::OnStatusChange 메서드
OnStatusChange ITfContextOwnerServices::OnStatusChange 메서드는 ITfContextOwner::GetStatus 메서드에서 반환된 TS_STATUS 구조체의 dwDynamicFlags 멤버가 변경될 때 컨텍스트 소유자에 의해 호출됩니다.
OnStatusChange ITfStatusSink::OnStatusChange 메서드

OnTestKeyDown ITfContextKeyEventSink::OnTestKeyDown 메서드
OnTestKeyDown ITfKeyEventSink::OnTestKeyDown 메서드
OnTestKeyUp ITfContextKeyEventSink::OnTestKeyUp 메서드
OnTestKeyUp ITfKeyEventSink::OnTestKeyUp 메서드
OnTextChange ITextStoreACPSink::OnTextChange 메서드
OnTextChange ITextStoreAnchorSink::OnTextChange 메서드
OnTextUpdate ITfFnAdviseText::OnTextUpdate 메서드
OnTextUpdated ITfPropertyStore::OnTextUpdated 메서드
OnThreadItemChange ITfLangBarEventSink::OnThreadItemChange 메서드
OnThreadTerminate ITfLangBarEventSink::OnThreadTerminate 메서드
OnTransitoryExtensionUpdated ITfTransitoryExtensionSink::OnTransitoryExtensionUpdated 메서드
OnUninitDocumentMgr ITfThreadMgrEventSink::OnUninitDocumentMgr 메서드

OnUpdate ITfLangBarItemSink::OnUpdate 메서드
OnUpdateComposition ITfContextOwnerCompositionSink::OnUpdateComposition 메서드
OnUpdated ITfPreservedKeyNotifySink::OnUpdated 메서드
OnUpdateInfo ITfDisplayAttributeMgr::OnUpdateInfo 메서드
OnUpdateInfo ITfDisplayAttributeNotifySink::OnUpdateInfo 메서드
OnWindowClosed IME UI를 닫은 후 TSF에서 호출됩니다.
OnWindowClosing IME UI를 닫기 전에 TSF에서 호출합니다.
OnWindowOpened IME UI를 연 후 TSF에서 호출됩니다.
OnWindowOpening IME UI를 열기 전에 TSF에서 호출합니다.
OnWindowUpdated 열린 IME UI의 크기 조정 및/또는 재배치 후 TSF에서 호출됩니다.
OnWindowUpdating 열린 IME UI의 크기를 조정 및/또는 재배치하기 전에 TSF에서 호출합니다.
PeekMessageA ITfMessagePump::P eekMessageA 메서드

PeekMessageW ITfMessagePump::P eekMessageW 메서드
재생 ITfFnPlayBack::P lay 메서드
팝 ITfDocumentMgr::P op 메서드
PreserveKey ITfKeystrokeMgr::P reserveKey 메서드
ProcessLattice ITfFnLMInternal::P rocessLattice 메서드
Push ITfDocumentMgr::P ush 메서드
QueryInsert ITextStoreACP::QueryInsert 메서드는 지정된 시작 및 끝 문자 위치가 유효한지 여부를 결정합니다.
QueryInsert 지정된 시작 및 끝 문자 위치가 유효한지 여부를 확인합니다. 편집을 실행하기 전에 이 메서드를 사용하여 문서에 대한 편집을 조정합니다. 이 메서드는 문서 범위 밖에 있는 값을 반환해서는 안됩니다.
QueryInsert ITextStoreAnchor::QueryInsert 메서드는 지정된 시작 및 끝 앵커가 유효한지 여부를 결정합니다. 편집을 실행하기 전에 이 메서드를 사용하여 문서에 대한 편집을 조정합니다. 이 메서드는 문서 범위 밖에 있는 값을 반환해서는 안됩니다.
QueryInsertEmbedded 지정한 개체를 문서에 삽입할 수 있는지 여부를 나타내는 값을 가져옵니다. (ITextStoreACP.QueryInsertEmbedded)
QueryInsertEmbedded 지정한 개체를 문서에 삽입할 수 있는지 여부를 나타내는 값을 가져옵니다. (ITextStoreACP2.QueryInsertEmbedded)

QueryInsertEmbedded ITextStoreAnchor::QueryInsertEmbedded 메서드
QueryInsertEmbedded ITfQueryEmbedded::QueryInsertEmbedded 메서드
QueryKey ITfFnLMProcessor::QueryKey 메서드
QueryLangID ITfFnLMProcessor::QueryLangID 메서드
QueryRange ITfFnLMProcessor::QueryRange 메서드
QueryRange ITfFnPlayBack::QueryRange 메서드
QueryRange ITfFnReconversion::QueryRange 메서드는 다시 변환이 적용되는 텍스트 범위를 가져옵니다.
QueryType ITfLMLattice::QueryType 메서드
다시 변환 ITfFnLMProcessor::Reconvert 메서드
다시 변환 ITfFnReconversion::Reconvert 메서드
등록 ITfInputProcessorProfiles::Register 메서드
RegisterActiveProfiles ITfFnLangProfileUtil::RegisterActiveProfiles 메서드

RegisterCategory ITfCategoryMgr::RegisterCategory 메서드
RegisterGUID ITfCategoryMgr::RegisterGUID 메서드
RegisterGUIDDescription ITfCategoryMgr::RegisterGUIDDescription 메서드
RegisterGUIDDWORD ITfCategoryMgr::RegisterGUIDDWORD 메서드
RegisterProfile ITfInputProcessorProfileMgr::RegisterProfile 메서드는 텍스트 서비스와 프로필을 등록합니다.
ReleaseInputProcessor ITfInputProcessorProfileMgr::ReleaseInputProcessor 메서드는 지정된 CLSID의 텍스트 서비스에 속하는 프로필을 비활성화하고 ITfTextInputProcessorEx 인터페이스의 인스턴스를 해제합니다.
RemoveItem ITfLangBarItemMgr::RemoveItem 메서드
RemoveLanguageProfile ITfInputProcessorProfiles::RemoveLanguageProfile 메서드
RequestAttrsAtPosition 지정된 문자 위치에 있는 텍스트 특성을 가져옵니다. (ITextStoreACP.RequestAttrsAtPosition)
RequestAttrsAtPosition 지정된 문자 위치에 있는 텍스트 특성을 가져옵니다. (ITextStoreACP2.RequestAttrsAtPosition)
RequestAttrsAtPosition ITextStoreAnchor::RequestAttrsAtPosition 메서드
RequestAttrsTransitioningAtPosition 지정된 문자 위치에서 전환되는 텍스트 특성을 가져옵니다. (ITextStoreACP.RequestAttrsTransitioningAtPosition)

RequestAttrsTransitioningAtPosition 지정된 문자 위치에서 전환되는 텍스트 특성을 가져옵니다. (ITextStoreACP2.RequestAttrsTransitioningAtPosition)
RequestAttrsTransitioningAtPosition ITextStoreAnchor::RequestAttrsTransitioningAtPosition 메서드
RequestEditSession ITfContext::RequestEditSession 메서드
RequestLock ITextStoreACP::RequestLock 메서드는 문서를 수정하기 위해 문서 잠금을 제공하기 위해 TSF 관리자가 호출합니다. 이 메서드는 ITextStoreACPSink::OnLockGranted 메서드를 호출하여 문서 잠금을 만듭니다.
RequestLock 문서를 수정하기 위해 문서 잠금을 제공하기 위해 TSF 관리자가 호출합니다. 이 메서드는 OnLockGranted 메서드를 호출하여 문서 잠금을 만듭니다.
RequestLock ITextStoreAnchor::RequestLock 메서드
RequestSupportedAttrs 문서에서 지원되는 특성을 가져옵니다. (ITextStoreACP.RequestSupportedAttrs)
RequestSupportedAttrs 문서에서 지원되는 특성을 가져옵니다. (ITextStoreACP2.RequestSupportedAttrs)
RequestSupportedAttrs ITextStoreAnchor::RequestSupportedAttrs 메서드
재설정 IEnumITfCompositionView::Reset 메서드
재설정 IEnumTfContexts::Reset 메서드

재설정 IEnumTfDisplayAttributeInfo::Reset 메서드
재설정 IEnumTfDocumentMgrs::Reset 메서드
재설정 IEnumTfFunctionProviders::Reset 메서드
재설정 IEnumTfInputProcessorProfiles::Reset 메서드는 현재 위치를 열거형 시퀀스의 시작 부분으로 이동하여 열거자 개체를 다시 설정합니다.
재설정 IEnumTfLanguageProfiles::Reset 메서드
재설정 IEnumTfProperties::Reset 메서드
재설정 IEnumTfPropertyValue::Reset 메서드
재설정 IEnumTfRanges::Reset 메서드
재설정 IEnumTfUIElements::Reset 메서드는 현재 위치를 열거형 시퀀스의 시작 부분으로 이동하여 열거자 개체를 다시 설정합니다.
재설정 ITfDisplayAttributeInfo::Reset 메서드
재설정 IEnumTfCandidates::Reset 메서드
재설정 IEnumTfLatticeElements::Reset 메서드

재설정 IEnumTfLangBarItems::Reset 메서드
복원 ITfRangeBackup::Restore 메서드
RestoreLastFocus ITfLangBarMgr::RestoreLastFocus 메서드
ResumeKeystrokeHandling 일시 중단된 키 입력 처리를 다시 시작합니다.
RetrieveRequestedAttrs 특성 요청 메서드에 대한 호출에서 반환되는 특성을 가져옵니다. (ITextStoreACP.RetrieveRequestedAttrs)
RetrieveRequestedAttrs 특성 요청 메서드에 대한 호출에서 반환되는 특성을 가져옵니다. (ITextStoreACP2.RetrieveRequestedAttrs)
RetrieveRequestedAttrs ITextStoreAnchor::RetrieveRequestedAttrs 메서드
직렬화 ITextStoreACPServices::Serialize 메서드
직렬화 ITfContextOwnerServices::Serialize 메서드는 텍스트 범위에서 속성을 가져오고 속성 데이터를 스트림 개체에 씁니다. 이렇게 하면 애플리케이션이 속성 데이터를 저장할 수 있습니다(예: 파일에 데이터를 쓸 때).
직렬화 ITfPropertyStore::Serialize 메서드
SetAttributeInfo ITfDisplayAttributeInfo::SetAttributeInfo 메서드

SetChangeHistoryMask 이 메서드는 구현되지 않았습니다.
SetDefaultLanguageProfile ITfInputProcessorProfiles::SetDefaultLanguageProfile 메서드
SetExtent ITfRangeACP::SetExtent 메서드
Setfocus ITfThreadMgr::SetFocus 메서드
Setfocus 입력 포커스를 지정된 문서 관리자로 설정합니다.
SetGravity IAnchor::SetGravity 메서드
SetGravity ITfRange::SetGravity 메서드
SetIcon ITfSystemLangBarItem::SetIcon 메서드
SetIconMode ITfSystemDeviceTypeLangBarItem::SetIconMode 메서드
SetInputScope 지정된 창에 대한 입력 범위를 설정합니다.
SetInputScopes 지정된 창에 대한 하나의 입력 범위, 여러 입력 범위, 하나의 구 목록, 정규식 및/또는 SRGS(Speech Recognition Grammar Specification) 규칙의 조합을 설정합니다.
SetInputScopes2 인터페이스의 참조를 지우려면 창이 제거되기 전에 애플리케이션이 IS_DEFAULT SetInputScope를 호출해야 합니다.

SetInputScopeXML 사용하지 마십시오. (SetInputScopeXML)
SetIntegrationStyle 통합 스타일을 설정합니다.
SetItemText ITfSystemLangBarItemText::SetItemText 메서드는 시스템 언어 모음 메뉴에 표시되는 텍스트를 수정합니다.
SetLanguageProfileDisplayName 재배포 가능 패키지:windows 2000에서 TSF 1.0을 _Requires.
SetModalInput ITfLangBarMgr::SetModalInput 메서드
SetPageIndex ITfCandidateListUIElement::SetPageIndex 메서드는 페이지 인덱스를 설정합니다.
SetPreservedKeyDescription ITfKeystrokeMgr::SetPreservedKeyDescription 메서드
SetResult ITfCandidateList::SetResult 메서드
SetResult 사용자가 후보를 선택할 때 텍스트 서비스 또는 IME에 기록 데이터를 제공합니다.
SetSelection ITextStoreACP::SetSelection 메서드는 문서 내의 텍스트를 선택합니다. 이 메서드를 호출하기 전에 애플리케이션에 문서에 대한 읽기/쓰기 잠금이 있어야 합니다.
SetSelection 문서 내의 텍스트를 선택합니다. 이 메서드를 호출하기 전에 애플리케이션에 문서에 대한 읽기/쓰기 잠금이 있어야 합니다.

SetSelection ITextStoreAnchor::SetSelection 메서드
SetSelection ITfCandidateListUIElementBehavior::SetSelection 메서드는 후보 목록의 선택을 설정합니다.
SetSelection ITfContext::SetSelection 메서드
SetStatus ITfFnPropertyUIStatus::SetStatus 메서드
Settext ITextStoreACP::SetText 메서드는 텍스트 선택을 제공된 문자 위치로 설정합니다.
Settext 텍스트 선택을 제공된 문자 위치로 설정합니다.
Settext ITextStoreAnchor::SetText 메서드는 제공된 두 앵커 위치 간에 텍스트 선택을 설정합니다.
Settext ITfRange::SetText 메서드는 텍스트 범위에서 다루는 콘텐츠를 대체합니다.
SetTooltipString ITfSystemLangBarItem::SetTooltipString 메서드
SetValue ITfCompartment::SetValue 메서드
SetValue ITfProperty::SetValue 메서드
SetValueStore ITfProperty::SetValueStore 메서드

Shift IAnchor::Shift 메서드는 텍스트 스트림 내에서 앵커를 앞뒤로 이동합니다.
ShiftEnd ITfComposition::ShiftEnd 메서드
ShiftEnd ITfRange::ShiftEnd 메서드
ShiftEndRegion ITfRange::ShiftEndRegion 메서드
ShiftEndToRange ITfRange::ShiftEndToRange 메서드
ShiftRegion IAnchor::ShiftRegion 메서드
ShiftStart ITfComposition::ShiftStart 메서드
ShiftStart ITfRange::ShiftStart 메서드
ShiftStartRegion ITfRange::ShiftStartRegion 메서드
ShiftStartToRange ITfRange::ShiftStartToRange 메서드
ShiftTo IAnchor::ShiftTo 메서드는 현재 앵커를 다른 앵커와 동일한 위치로 이동합니다.
표시 ITfUIElement::Show 메서드는 이 UI 요소의 텍스트 서비스의 UI를 표시합니다.

표시 ITfFnConfigure::Show 메서드
표시 ITfFnConfigureRegisterEudc::Show 메서드는 EUDC 키 시퀀스 레지스터 UI를 표시합니다.
표시 ITfFnConfigureRegisterWord::Show 메서드
표시 ITfFnShowHelp::Show 메서드
표시 ITfLangBarItem::Show 메서드
ShowCandidateNumbers 후보 번호를 표시할지 여부를 지정합니다.
ShowFloating ITfLangBarEventSink::ShowFloating 메서드
ShowFloating ITfLangBarMgr::ShowFloating 메서드
ShowUI ITfSpeechUIServer::ShowUI 메서드
축소 ITfPropertyStore::Shrink 메서드
SimulatePreservedKey ITfKeystrokeMgr::SimulatePreservedKey 메서드
skip IEnumITfCompositionView::Skip 메서드

skip IEnumTfContexts::Skip 메서드
skip IEnumTfDisplayAttributeInfo::Skip 메서드
skip IEnumTfDocumentMgrs::Skip 메서드
skip IEnumTfFunctionProviders::Skip 메서드
skip IEnumTfInputProcessorProfiles::Skip 메서드는 지정된 수의 요소에 따라 열거형 시퀀스에서 현재 위치를 앞으로 이동합니다.
skip IEnumTfLanguageProfiles::Skip 메서드
skip IEnumTfProperties::Skip 메서드
skip IEnumTfPropertyValue::Skip 메서드
skip IEnumTfRanges::Skip 메서드
skip IEnumTfUIElements::Skip 메서드는 현재 위치에서 열거형 시퀀스의 지정된 요소 수를 가져옵니다.
skip IEnumTfCandidates::Skip 메서드
skip IEnumTfLatticeElements::Skip 메서드

skip IEnumTfLangBarItems::Skip 메서드
StartComposition ITfContextComposition::StartComposition 메서드
SubstituteKeyboardLayout ITfInputProcessorProfiles::SubstituteKeyboardLayout 메서드
SuspendKeystrokeHandling 키 입력 처리를 일시 중단합니다.
SystemDisableMSAA ITfMSAAControl::SystemDisableMSAA 메서드
SystemEnableMSAA ITfMSAAControl::SystemEnableMSAA 메서드
TakeOwnership ITfContextComposition::TakeOwnership 메서드
TerminateComposition ITfContextOwnerCompositionServices::TerminateComposition 메서드
TestKeyDown ITfKeystrokeMgr::TestKeyDown 메서드
TestKeyUp ITfKeystrokeMgr::TestKeyUp 메서드
TF_CreateCategoryMgr TF_CreateCategoryMgr 함수는 COM을 초기화할 필요 없이 범주 관리자 개체를 만듭니다. 호출 스레드가 MSCTF.DLL 소유하는 개체에 대한 참조 수를 유지해야 하므로 신중하게 사용해야 합니다.
TF_CreateDisplayAttributeMgr TF_CreateDisplayAttributeMgr 함수는 COM을 초기화하지 않고도 표시 특성 관리자 개체를 만드는 데 사용됩니다.

TF_CreateInputProcessorProfiles TF_CreateInputProcessorProfiles 함수는 COM을 초기화할 필요 없이 입력 프로세서 프로필 개체를 만드는 데 사용됩니다.
TF_CreateLangBarItemMgr TF_CreateLangBarItemMgr 함수는 COM을 초기화하지 않고도 언어 모음 항목 관리자 개체를 만드는 데 사용됩니다.
TF_CreateLangBarMgr TF_CreateLangBarMgr 함수는 COM을 초기화하지 않고도 언어 모음 관리자 개체를 만듭니다. 호출 프로세스는 Msctf.dll 소유하는 개체에 대한 적절한 참조 수를 유지해야 하므로 이 메서드를 사용하지 않는 것이 좋습니다.
TF_CreateThreadMgr TF_CreateThreadMgr 함수는 COM을 초기화할 필요 없이 스레드 관리자 개체를 만듭니다. 호출 프로세스는 Msctf.dll 소유하는 개체에 대한 적절한 참조 수를 유지해야 하므로 이 메서드를 사용하지 않는 것이 좋습니다.
TF_GetThreadMgr TF_GetThreadMgr 함수는 호출 스레드 내에서 이전에 만든 스레드 관리자 개체의 복사본을 가져옵니다.
TrackProperties ITfContext::TrackProperties 메서드
UnadviseEventSink ITfLangBarMgr::UnadviseEventSink 메서드
UnadviseItemSink ITfLangBarItemMgr::UnadviseItemSink 메서드
UnadviseItemsSink ITfLangBarItemMgr::UnadviseItemsSink 메서드
UnadviseKeyEventSink ITfKeystrokeMgr::UnadviseKeyEventSink 메서드

UnadviseMouseSink ITfMouseTracker::UnadviseMouseSink 메서드
UnadviseMouseSink ITfMouseTrackerACP::UnadviseMouseSink 메서드
UnadviseSingleSink ITfSourceSingle::UnadviseSingleSink 메서드
UnadviseSink ITextStoreACP::UnadviseSink 메서드는 TSF 관리자의 알림이 더 이상 필요하지 않음을 나타내기 위해 애플리케이션에서 호출됩니다. TSF 관리자는 싱크 인터페이스를 해제하고 알림을 중지합니다.
UnadviseSink TSF 관리자의 알림이 더 이상 필요하지 않음을 나타내기 위해 애플리케이션에서 호출됩니다. TSF 관리자는 싱크 인터페이스를 해제하고 알림을 중지합니다.
UnadviseSink ITextStoreAnchor::UnadviseSink 메서드
UnadviseSink ITfSource::UnadviseSink 메서드
UninitLocalMsCtfMonitor UninitLocalMsCtfMonitor 함수는 현재 데스크톱에서 TextServicesFramework를 초기화하지 않습니다.
UnpreserveKey ITfKeystrokeMgr::UnpreserveKey 메서드
등록 취소 ITfInputProcessorProfiles::Unregister 메서드
UnregisterCategory ITfCategoryMgr::UnregisterCategory 메서드

UnregisterGUIDDescription ITfCategoryMgr::UnregisterGUIDDescription 메서드
UnregisterGUIDDWORD ITfCategoryMgr::UnregisterGUIDDWORD 메서드
UnregisterProfile ITfInputProcessorProfileMgr::UnregisterProfile 메서드는 텍스트 서비스와 프로필을 등록 취소합니다.
비직렬화 ITextStoreACPServices::Unserialize 메서드
비직렬화 ITfContextOwnerServices::Unserialize 메서드
UpdateBalloon ITfFnBalloon::UpdateBalloon 메서드
UpdateBalloon ITfSpeechUIServer::UpdateBalloon 메서드
UpdateUIElement UI 요소를 업데이트해야 하는 경우 텍스트 서비스에서 ITfUIElementMgr::UpdateUIElement 메서드를 호출합니다.
UpdateUIElement ITfUIElementSink::UpdateUIElement 메서드는 UIElement의 콘텐츠가 업데이트되면 호출됩니다.

인터페이스

IAnchor IAnchor 인터페이스는 TSF 관리자에 의해 구현됩니다. Microsoft Active Accessibility 클라이언트는 IAnchor 앵커 개체를 사용하여 텍스트 스트림 내의 텍스트 범위를 구분합니다.

IEnumITfCompositionView IEnumITfCompositionView 인터페이스는 TSF 관리자에 의해 구현되어 컴퍼지션 뷰 개체의 열거형을 제공합니다.
IEnumTfCandidates IEnumTfCandidates 인터페이스는 텍스트 서비스에 의해 구현되며 TSF 관리자가 후보 문자열 개체의 열거형을 제공하는 데 사용됩니다.
IEnumTfContexts IEnumTfContexts 인터페이스는 컨텍스트 개체의 열거형을 제공하기 위해 TSF 관리자에 의해 구현됩니다.
IEnumTfContextViews 구현되지 않았습니다. (IEnumTfContextViews)
IEnumTfDisplayAttributeInfo IEnumTfDisplayAttributeInfo 인터페이스는 디스플레이 특성 정보 개체의 열거형을 제공하기 위해 TSF 관리자에 의해 구현됩니다.
IEnumTfDocumentMgrs IEnumTfDocumentMgrs 인터페이스는 문서 관리자 개체의 열거형을 제공하기 위해 TSF 관리자에 의해 구현됩니다.
IEnumTfFunctionProviders IEnumTfFunctionProviders 인터페이스는 함수 공급자 개체의 열거형을 제공하기 위해 TSF 관리자에 의해 구현됩니다.
IEnumTfInputProcessorProfiles IEnumTfInputProcessorProfiles 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션 또는 텍스트 서비스에서 사용됩니다. ITfInputProcessorProfileMgr::EnumProfiles에서 이 인터페이스를 검색하고 등록된 프로필을 열거할 수 있습니다.
IEnumTfLangBarItems IEnumTfLangBarItems 인터페이스는 TSF 관리자가 언어 모음 항목 개체의 열거형을 제공하기 위해 구현합니다.
IEnumTfLanguageProfiles IEnumTfLanguageProfiles 인터페이스는 언어 프로필의 열거형을 제공하기 위해 TSF 관리자에 의해 구현됩니다.

IEnumTfLatticeElements IEnumTfLatticeElements 인터페이스는 격자 요소의 열거형을 제공하기 위해 TSF 관리자에 의해 구현됩니다.
IEnumTfProperties IEnumTfProperties 인터페이스는 속성 개체의 열거형을 제공하기 위해 TSF 관리자에 의해 구현됩니다.
IEnumTfPropertyValue IEnumTfPropertyValue 인터페이스는 TSF 관리자에 의해 구현되어 속성 값의 열거형을 제공합니다.
IEnumTfRanges IEnumTfRanges 인터페이스는 범위 개체의 열거형을 제공하기 위해 TSF 관리자에 의해 구현됩니다.
IEnumTfUIElements IEnumTfUIElements 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션 또는 텍스트 서비스에서 사용됩니다. ITfUIElementMgr::EnumUIElements에서 이 인터페이스를 검색하고 등록된 UI 요소를 열거할 수 있습니다.
ITextStoreACP ITextStoreACP 인터페이스는 애플리케이션에서 구현되며 TSF 관리자가 TSF의 텍스트 스트림 또는 텍스트 저장소를 조작하는 데 사용됩니다.
ITextStoreACP2 ITextStoreACP2 인터페이스는 애플리케이션에서 구현되며 TSF 관리자가 TSF의 텍스트 스트림 또는 텍스트 저장소를 조작하는 데 사용됩니다.
ITextStoreACPServices ITextStoreACPServices 인터페이스는 ACP 기반 애플리케이션에 다양한 서비스를 제공하기 위해 TSF 관리자에 의해 구현됩니다.
ITextStoreACPSink ITextStoreACPSink 인터페이스는 TSF 관리자에 의해 구현되며 ACP 기반 애플리케이션에서 특정 이벤트가 발생할 때 관리자에게 알리는 데 사용됩니다. 관리자는 ITextStoreACP::AdviseSink를 호출하여 이 조언 싱크를 설치합니다.
ITextStoreAnchor ITextStoreAnchor 인터페이스는 Microsoft Active Accessibility 클라이언트에 의해 구현되며 TSF 관리자가 텍스트 스트림을 조작하는 데 사용됩니다.

ITextStoreAnchorSink ITextStoreAnchorSink 인터페이스는 TSF 관리자에 의해 구현되며 앵커 기반 애플리케이션에서 특정 이벤트가 발생할 때 관리자에게 알리는 데 사용됩니다. 관리자는 ITextStoreAnchor::AdviseSink를 호출하여 이 조언 싱크를 설치합니다.
ITfActiveLanguageProfileNotifySink ITfActiveLanguageProfileNotifySink 인터페이스는 활성 언어 또는 텍스트 서비스가 변경될 때 알림을 받기 위해 애플리케이션에서 구현합니다.
ITfCandidateList ITfCandidateList 인터페이스는 텍스트 서비스에서 구현되며 TSF 관리자 또는 클라이언트(애플리케이션 또는 기타 텍스트 서비스)에서 후보 문자열 개체를 가져오고 조작하는 데 사용됩니다.
ITfCandidateListUIElement ITfCandidateListUIElement 인터페이스는 후보 목록 UI가 있는 텍스트 서비스에 의해 구현됩니다.
ITfCandidateListUIElementBehavior 이 인터페이스는 후보 목록 UI가 있는 텍스트 서비스에 의해 구현되며 해당 UI는 애플리케이션에서 제어할 수 있습니다. 애플리케이션은 ITfUIElement에서 이 인터페이스를 QI로 지정하고 후보 목록 동작을 제어합니다.
ITfCandidateString ITfCandidateString 인터페이스는 텍스트 서비스에서 구현되며 TSF 관리자 또는 클라이언트에서 후보 문자열 개체에 대한 정보를 가져오는 데 사용됩니다.
ITfCategoryMgr ITfCategoryMgr 인터페이스는 텍스트 서비스에 대한 개체의 범주를 관리합니다. TSF 관리자는 이 인터페이스를 구현합니다.
ITfCleanupContextDurationSink ITfCleanupContextDurationSink 인터페이스는 컨텍스트 정리 작업이 수행될 때 알림을 수신하기 위해 텍스트 서비스에 의해 구현됩니다.
ITfCleanupContextSink ITfCleanupContextSink 인터페이스는 컨텍스트 정리 작업이 발생할 때 알림을 수신하도록 텍스트 서비스에 의해 구현됩니다. 이 알림 싱크는 IID_ITfCleanupContextSink ITfSourceSingle::AdviseSingleSink를 호출하여 설치됩니다.

ITfClientId ITfClientId 인터페이스는 TSF 관리자에 의해 구현됩니다. 이 인터페이스는 TSF 개체에 대한 클라이언트 식별자를 가져오는 데 사용됩니다. 이 인터페이스의 인스턴스는 IID_ITfClientId 사용하여 스레드 관리자를 쿼리하여 가져옵니다.
ITfCompartment ITfCompartment 인터페이스는 TSF 관리자에 의해 구현되며 클라이언트(애플리케이션 및 텍스트 서비스)에서 TSF 구획에서 데이터를 가져오고 설정하는 데 사용됩니다.
ITfCompartmentEventSink ITfCompartmentEventSink 인터페이스는 클라이언트(애플리케이션 또는 텍스트 서비스)에서 구현되며, TSF 관리자가 구획 데이터가 변경될 때 클라이언트에 알리는 데 사용됩니다.
ITfCompartmentMgr ITfCompartmentMgr 인터페이스는 TSF 관리자에 의해 구현되며 클라이언트(애플리케이션 및 텍스트 서비스)에서 TSF 구획을 가져오고 조작하는 데 사용됩니다.
ITfComposition ITfComposition 인터페이스는 TSF 관리자에 의해 구현되며 텍스트 서비스에서 컴퍼지션에 대한 데이터를 가져오고 종료하는 데 사용됩니다. 이 인터페이스의 인스턴스는 ITfContextComposition::StartComposition 메서드에 의해 제공됩니다.
ITfCompositionSink ITfCompositionSink 인터페이스는 컴퍼지션이 종료될 때 알림을 수신하기 위해 텍스트 서비스에 의해 구현됩니다.
ITfCompositionView ITfCompositionView 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션에서 컴퍼지션 뷰에 대한 데이터를 가져오는 데 사용됩니다. 이 인터페이스의 인스턴스는 ITfContextOwnerCompositionSink 메서드 중 하나에서 제공됩니다.
ITfConfigureSystemKeystrokeFeed ITfConfigureSystemKeystrokeFeed 인터페이스는 키 입력 처리를 사용하거나 사용하지 않도록 설정하기 위해 TSF 관리자에 의해 구현됩니다.
ITfContext ITfContext 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션 및 텍스트 서비스에서 편집 컨텍스트에 액세스하는 데 사용됩니다.

ITfContextComposition ITfContextComposition 인터페이스는 TSF 관리자에 의해 구현되며 텍스트 서비스에서 컴퍼지션을 만들고 조작하는 데 사용됩니다. 이 인터페이스의 인스턴스는 IID_ITfContextComposition ITfContext::QueryInterface에서 제공됩니다.
ITfContextKeyEventSink ITfContextKeyEventSink 인터페이스는 입력 컨텍스트에서 발생하는 키보드 이벤트 알림을 수신하기 위해 텍스트 서비스에 의해 구현됩니다.
ITfContextOwner ITfContextOwner 인터페이스는 텍스트 저장소 없이 텍스트 입력을 수신하도록 애플리케이션 또는 텍스트 서비스에 의해 구현됩니다. 이 인터페이스의 인스턴스는 애플리케이션이 ITfSource::AdviseSink 메서드를 호출할 때 가져옵니다.
ITfContextOwnerCompositionServices ITfContextOwnerCompositionServices 인터페이스는 TSF 관리자에 의해 구현되며 컨텍스트 소유자가 텍스트 서비스에서 만든 컴퍼지션을 조작하는 데 사용됩니다.
ITfContextOwnerCompositionSink ITfContextOwnerCompositionSink 인터페이스는 컴퍼지션 관련 알림을 수신하기 위해 애플리케이션에 의해 구현됩니다.
ITfContextOwnerServices ITfContextOwnerServices 인터페이스는 관리자가 구현하며 컨텍스트 소유자 역할을 하는 텍스트 서비스 또는 애플리케이션에서 사용됩니다.
ITfContextView ITfContextView 인터페이스는 TSF 관리자에 의해 구현되며 클라이언트(애플리케이션 또는 텍스트 서비스)에서 컨텍스트 뷰에 대한 정보를 가져오는 데 사용됩니다.
ITfCreatePropertyStore ITfCreatePropertyStore 인터페이스는 속성 저장소 데이터의 지속성을 지원하기 위해 텍스트 서비스에 의해 구현됩니다.
ITfDisplayAttributeInfo ITfDisplayAttributeInfo 인터페이스는 표시 특성 데이터를 제공하기 위해 텍스트 서비스에 의해 구현됩니다. 이 인터페이스는 텍스트가 표시되는 방식을 결정해야 하는 모든 구성 요소(가장 자주 애플리케이션)에서 사용됩니다.

ITfDisplayAttributeMgr ITfDisplayAttributeMgr 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션에서 디스플레이 특성을 가져오고 열거하는 데 사용됩니다. 개별 표시 특성은 ITfDisplayAttributeInfo 인터페이스를 통해 액세스됩니다.
ITfDisplayAttributeNotifySink ITfDisplayAttributeNotifySink 인터페이스는 디스플레이 특성 정보가 업데이트될 때 알림을 수신하기 위해 애플리케이션에서 구현됩니다.
ITfDisplayAttributeProvider ITfDisplayAttributeProvider 인터페이스는 텍스트 서비스에 의해 구현되며 TSF 관리자가 개별 표시 특성 정보 개체를 열거하고 가져오는 데 사용됩니다.
ITfDocumentMgr ITfDocumentMgr 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션 또는 텍스트 서비스에서 텍스트 컨텍스트를 만들고 관리하는 데 사용됩니다. 이 인터페이스의 인스턴스를 가져오려면 ITfThreadMgr::CreateDocumentMgr을 호출합니다.
ITfEditRecord ITfEditRecord 인터페이스는 TSF 관리자에 의해 구현되며 텍스트 편집 싱크에서 편집 세션 중에 변경된 내용을 확인하는 데 사용됩니다.
ITfEditSession ITfEditSession 인터페이스는 텍스트 서비스에 의해 구현되며 TSF 관리자가 컨텍스트의 텍스트 및 속성을 읽고/또는 수정하는 데 사용됩니다.
ITfEditTransactionSink ITfEditTransactionSink 인터페이스는 텍스트 서비스에 의해 구현되며 TSF 관리자가 트랜잭션 편집을 지원하는 데 사용됩니다.
ITfFnAdviseText ITfFnAdviseText 인터페이스는 텍스트 서비스에 의해 구현되고 TSF 관리자가 컨텍스트의 텍스트 또는 격자 요소가 변경될 때 알림을 제공하는 데 사용됩니다.
ITfFnBalloon ITfFnBalloon 인터페이스는 텍스트 서비스에 의해 구현되며 애플리케이션 또는 다른 텍스트 서비스에서 텍스트 서비스가 언어 모음에 추가하는 풍선 항목을 업데이트하는 데 사용됩니다.

ITfFnConfigure ITfFnConfigure 인터페이스는 텍스트 서비스가 구성 대화 상자를 표시할 수 있도록 Text Services 제어판 애플리케이션을 사용하도록 텍스트 서비스에 의해 구현됩니다.
ITfFnConfigureRegisterEudc ITfFnConfigureRegisterEudc 인터페이스는 지정된 EUDC에 대한 키 시퀀스를 등록하는 UI를 제공하기 위해 텍스트 서비스에 의해 구현됩니다.
ITfFnConfigureRegisterWord ITfFnConfigureRegisterWord 인터페이스는 텍스트 서비스에서 구현되어 IME(Active Input Method Editor)가 텍스트 서비스에 단어 등록 대화 상자를 표시하도록 합니다.
ITfFnGetLinguisticAlternates ITfFnGetLinguisticAlternates 인터페이스는 텍스트 서비스 및/또는 TSF 관리자에 의해 구현되어 매개 변수로 전달된 지정된 범위 내의 텍스트에 대한 언어 대체를 제공합니다.
ITfFnGetPreferredTouchKeyboardLayout ITfFnGetPreferredTouchKeyboardLayout 인터페이스는 받은 편지함 Windows 8 터치 키보드에서 지원하는 특정 키보드 레이아웃의 사용을 지정하기 위해 텍스트 서비스에 의해 구현됩니다.
ITfFnGetSAPIObject ITfFnGetSAPIObject 인터페이스는 SAPI(Speech API) 텍스트 서비스에 의해 구현됩니다. 이 인터페이스는 TSF 관리자 또는 클라이언트(애플리케이션 또는 기타 텍스트 서비스)에서 다양한 SAPI 개체를 가져오는 데 사용됩니다.
ITfFnLangProfileUtil ITfFnLangProfileUtil 인터페이스는 음성 텍스트 서비스에 의해 구현되며 음성 텍스트 서비스에 대한 유틸리티 메서드를 제공하는 데 사용됩니다.
ITfFnLMInternal ITfFnLMInternal 인터페이스는 사용되지 않습니다.
ITfFnLMProcessor ITfFnLMProcessor 인터페이스는 언어 모델 텍스트 서비스에서 구현되며 애플리케이션 또는 텍스트 서비스에서 대체 언어 모델 처리를 사용하도록 설정하는 데 사용됩니다.

ITfFnPlayBack ITfFnPlayBack 인터페이스는 SAPI(Speech API) 텍스트 서비스에 의해 구현됩니다. 이 인터페이스는 TSF 관리자 또는 클라이언트(애플리케이션 또는 기타 텍스트 서비스)에서 음성 입력 텍스트의 오디오 데이터를 제어하는 데 사용됩니다.
ITfFnPropertyUIStatus ITfFnPropertyUIStatus 인터페이스는 텍스트 서비스에 의해 구현되며 애플리케이션 또는 텍스트 서비스에서 텍스트 서비스 속성 UI의 상태를 가져오고 설정하는 데 사용됩니다.
ITfFnReconversion ITfFnReconversion 인터페이스는 텍스트 서비스에 의해 구현되며 TSF 관리자 또는 클라이언트에서 텍스트 서비스에서 제공하는 텍스트의 재전송을 지원하는 데 사용됩니다.
ITfFnSearchCandidateProvider IME(입력 방법 편집기)에서 통합 검색 환경을 사용하도록 설정합니다.
ITfFnShowHelp ITfFnShowHelp 인터페이스는 언어 모음이 언어 모음 도움말 메뉴에 텍스트 서비스에 대한 도움말 명령을 배치할 수 있도록 텍스트 서비스에 의해 구현됩니다.
ITfFunction ITfFunction 인터페이스는 개별 함수 인터페이스의 기본 인터페이스입니다.
ITfFunctionProvider ITfFunctionProvider 인터페이스는 다양한 함수 개체를 제공하기 위해 애플리케이션 또는 텍스트 서비스에 의해 구현됩니다.
ITfInputProcessorProfileActivationSink ITfInputProcessorProfileActivationSink 인터페이스는 프로필이 변경될 때 알림을 수신하기 위해 애플리케이션에 의해 구현됩니다.
ITfInputProcessorProfileMgr ITfInputProcessorProfileMgr 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션 또는 텍스트 서비스에서 하나 이상의 텍스트 서비스의 언어 프로필을 조작하는 데 사용됩니다.
ITfInputProcessorProfiles ITfInputProcessorProfiles 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션 또는 텍스트 서비스에서 하나 이상의 텍스트 서비스의 언어 프로필을 조작하는 데 사용됩니다.

ITfInputProcessorProfilesEx 이 인터페이스는 TSF 관리자에 의해 구현되며 텍스트 서비스 또는 애플리케이션에서 언어 프로필의 표시 설명을 설정하는 데 사용됩니다.
ITfInputProcessorProfileSubstituteLayout 이 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션 또는 텍스트 서비스에서 텍스트 서비스 프로필의 대체 입력 로캘 식별자(키보드 레이아웃)를 조작하는 데 사용됩니다.
ITfInputScope ITfInputScope 인터페이스는 텍스트 입력 프로세서에서 창과 연결된 문서 컨텍스트를 나타내는 InputScope 값을 가져오는 데 사용됩니다.
ITfInputScope2 ITfInputScope2 인터페이스는 텍스트 입력 프로세서에서 IEnumString 인터페이스 포인터를 가져오는 데 사용되며, 이 IEnumString 인터페이스는 애플리케이션이 이 컨텍스트에 대해 지정한 단어 목록을 열거합니다.
ITfInsertAtSelection ITfInsertAtSelection 인터페이스는 관리자가 구현하며 텍스트 서비스에서 텍스트 또는 포함된 개체를 컨텍스트에 삽입하는 데 사용됩니다. 텍스트 서비스는 ITfContext::QueryInterface를 호출하여 이 인터페이스를 가져옵니다.
ITfIntegratableCandidateListUIElement 텍스트 서비스 및 IME(입력 메서드 편집기)가 UI 없는 모드 동작을 조정할 수 있도록 합니다.
ITfKeyEventSink ITfKeyEventSink 인터페이스는 키보드 및 포커스 이벤트 알림을 수신하기 위해 텍스트 서비스에 의해 구현됩니다. 이 이벤트 싱크를 설치하려면 ITfKeystrokeMgr::AdviseKeyEventSink를 호출합니다.
ITfKeystrokeMgr ITfKeystrokeMgr 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션 및 텍스트 서비스에서 키보드 관리자와 상호 작용하는 데 사용됩니다.
ITfKeyTraceEventSink ITfKeyTraceEventSink 인터페이스는 대상에 의해 이벤트가 처리되기 전에 키 스트로크 이벤트 알림을 수신하기 위해 애플리케이션 또는 텍스트 서비스에 의해 구현됩니다.

ITfLangBarEventSink ITfLangBarEventSink 인터페이스는 애플리케이션 또는 텍스트 서비스에 의해 구현되며 언어 모음에서 언어 모음에서 발생하는 특정 이벤트의 알림을 제공하는 데 사용됩니다.
ITfLangBarItem ITfLangBarItem 인터페이스는 언어 모음 항목 공급자에 의해 구현되며 언어 모음 관리자가 언어 모음 항목에 대한 자세한 정보를 가져오는 데 사용됩니다.
ITfLangBarItemBalloon ITfLangBarItemBalloon 인터페이스는 애플리케이션 또는 텍스트 서비스에 의해 구현되며 언어 모음 관리자가 언어 모음의 풍선 항목과 관련된 정보를 가져오는 데 사용됩니다.
ITfLangBarItemBitmap ITfLangBarItemBitmap 인터페이스는 애플리케이션 또는 텍스트 서비스에 의해 구현되며 언어 모음 관리자가 언어 모음의 비트맵 항목과 관련된 정보를 가져오는 데 사용됩니다.
ITfLangBarItemBitmapButton ITfLangBarItemBitmapButton 인터페이스는 언어 막대 비트맵 단추 공급자에 의해 구현되며 언어 모음 관리자가 언어 모음의 비트맵 단추 항목과 관련된 정보를 가져오는 데 사용됩니다.
ITfLangBarItemButton ITfLangBarItemButton 인터페이스는 언어 모음 단추 공급자에 의해 구현되며 언어 모음 관리자가 언어 모음의 단추 항목에 대한 정보를 가져오는 데 사용됩니다.
ITfLangBarItemMgr ITfLangBarItemMgr 인터페이스는 언어 모음에서 구현되며 텍스트 서비스에서 언어 모음의 항목을 관리하는 데 사용됩니다.
ITfLangBarItemSink ITfLangBarItemSink 인터페이스는 언어 모음에서 구현되며 언어 모음 항목 공급자가 언어 모음에 언어 모음 항목 변경 내용을 알리는 데 사용됩니다.
ITfLangBarMgr ITfLangBarMgr 인터페이스는 TSF 관리자에 의해 구현되며 텍스트 서비스에서 이벤트 싱크 알림을 관리하고 부동 언어 모음 표시 설정을 구성하는 데 사용됩니다. 인터페이스 ID가 IID_ITfLangBarMgr.

ITfLanguageProfileNotifySink ITfLanguageProfileNotifySink 인터페이스는 언어 프로필이 변경될 때 알림을 수신하기 위해 애플리케이션에 의해 구현됩니다.
ITfLMLattice ITfLMLattice 인터페이스는 격자 요소 속성에 대한 정보를 제공하기 위해 음성 텍스트 서비스에 의해 구현되며 클라이언트(애플리케이션 또는 기타 텍스트 서비스)에서 사용됩니다.
ITfMenu ITfMenu 인터페이스는 언어 모음에 의해 구현되며 언어 모음 단추 공급자가 언어 모음에서 단추에 대해 표시할 메뉴에 항목을 추가하는 데 사용됩니다.
ITfMessagePump ITfMessagePump 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션에서 애플리케이션 메시지 큐에서 메시지를 가져오는 데 사용됩니다.
ITfMouseSink ITfMouseSink 인터페이스는 마우스 이벤트 알림을 수신하기 위해 텍스트 서비스에 의해 구현됩니다. 마우스 이벤트 싱크는 ITfMouseTracker 인터페이스 중 하나의 ITfMouseTracker::AdviseMouseSink 메서드와 함께 설치됩니다.
ITfMouseTracker ITfMouseTracker 인터페이스는 TSF 관리자에 의해 구현되며 텍스트 서비스에서 마우스 이벤트 알림 싱크를 관리하는 데 사용됩니다. 이 인터페이스의 인스턴스는 IID_ITfMouseTracker ITfContext 개체를 쿼리하여 가져옵니다.
ITfMouseTrackerACP ITfMouseTrackerACP 인터페이스는 마우스 이벤트 싱크를 지원하기 위해 애플리케이션에 의해 구현됩니다.
ITfMSAAControl ITfMSAAControl 인터페이스는 TSF에서 불필요한 오버헤드를 방지하기 위해 Microsoft Active Accessibility에서 TSF 컨트롤에서 문서를 추가하거나 제거하는 데 사용됩니다. 이 인터페이스는 다른 애플리케이션에서 사용하지 않는 것이 좋습니다.
ITfPersistentPropertyLoaderACP ITfPersistentPropertyLoaderACP 인터페이스는 애플리케이션에서 구현되며 TSF 관리자가 속성을 비동기적으로 로드하는 데 사용합니다.

ITfPreservedKeyNotifySink ITfPreservedKeyNotifySink 인터페이스는 키가 보존되거나, 예약되지 않거나, 유지된 키 설명이 변경될 때 알림을 수신하기 위해 애플리케이션 또는 TSF 텍스트 서비스에 의해 구현됩니다.
ITfProperty ITfProperty 인터페이스는 TSF 관리자에 의해 구현되며 클라이언트(애플리케이션 또는 텍스트 서비스)에서 속성 값을 수정하는 데 사용됩니다.
ITfPropertyStore ITfPropertyStore 인터페이스는 텍스트 서비스에 의해 구현되며 TSF 관리자가 비정적 속성 값을 제공하는 데 사용됩니다. 이 인터페이스의 인스턴스는 ITfProperty::SetValueStore에 전달됩니다.
ITfQueryEmbedded ITfQueryEmbedded 인터페이스는 TSF 관리자에 의해 구현되며 텍스트 서비스에서 컨텍스트가 포함된 개체를 허용할 수 있는지 여부를 결정하는 데 사용됩니다.
ITfRange ITfRange 인터페이스는 텍스트 서비스 및 애플리케이션에서 지정된 컨텍스트 내에서 텍스트를 참조하고 조작하는 데 사용됩니다. 인터페이스 ID가 IID_ITfRange.
ITfRangeACP ITfRangeACP 인터페이스는 TSF 관리자에 의해 구현되며 ACP(애플리케이션 문자 위치) 기반 애플리케이션에서 범위 개체에 액세스하고 조작하는 데 사용됩니다.
ITfRangeBackup ITfRangeBackup 인터페이스는 TSF 관리자에 의해 구현되며 텍스트 서비스에서 범위 개체에 포함된 데이터의 백업 복사본을 만드는 데 사용됩니다.
ITfReadingInformationUIElement ITfCandidateListUIElement 인터페이스는 가까운 케어에서 정보 UI를 읽기 위한 UI가 있는 텍스트 서비스에 의해 구현됩니다.
ITfReadOnlyProperty ITfReadOnlyProperty 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션 또는 텍스트 서비스에서 속성 데이터를 가져오는 데 사용됩니다.
ITfReverseConversion 지정된 문자열의 역방향 변환을 수행합니다.

ITfReverseConversionList 지정된 문자열을 만드는 데 필요한 키 입력 시퀀스의 목록을 나타냅니다.
ITfReverseConversionMgr 역방향 변환을 수행하는 데 사용되는 ITfReverseConversion 개체에 대한 액세스를 제공합니다.
ITfSource ITfSource 인터페이스는 TSF 관리자에 의해 구현됩니다. 애플리케이션 및 텍스트 서비스에서 advise 싱크를 설치하고 제거하는 데 사용됩니다.
ITfSourceSingle ITfSourceSingle 인터페이스는 TSF 관리자에 의해 구현됩니다.
ITfSpeechUIServer ITfSpeechUIServer 인터페이스는 TSF 언어 모음에서 음성 관련 사용자 인터페이스를 관리합니다.
ITfStatusSink ITfStatusSink 인터페이스는 전역 문서 상태 변경을 지원합니다. 이 조언 싱크는 IID_ITfStatusSink ITfSource::AdviseSink를 호출하여 설치됩니다. 텍스트 서비스는 필요에 따라 이 인터페이스를 구현할 수 있습니다.
ITfSystemDeviceTypeLangBarItem ITfSystemDeviceTypeLangBarItem 인터페이스는 시스템 언어 모음 항목에 의해 구현되며 애플리케이션 또는 텍스트 서비스에서 시스템 항목이 아이콘을 표시하는 방법을 제어하는 데 사용됩니다.
ITfSystemLangBarItem ITfSystemLangBarItem 인터페이스는 시스템 언어 모음 메뉴에서 구현되며 시스템 언어 모음 확장에서 메뉴에 표시되는 아이콘 및/또는 도구 설명 문자열을 수정하는 데 사용됩니다.
ITfSystemLangBarItemSink ITfSystemLangBarItemSink 인터페이스는 시스템 언어 모음 메뉴 확장에 의해 구현되며 기존 시스템 언어 모음 메뉴에 메뉴 항목을 추가할 수 있도록 시스템 언어 모음 메뉴(호스트)에서 사용됩니다.
ITfSystemLangBarItemText ITfSystemLangBarItemText 인터페이스는 시스템 언어 모음에서 구현되며 시스템 언어 모음 확장에서 메뉴에 표시되는 설명을 수정하는 데 사용됩니다.

ITfTextEditSink ITfTextEditSink 인터페이스는 읽기/쓰기 액세스가 포함된 편집 세션의 완료를 지원합니다.
ITfTextInputProcessor ITfTextInputProcessor 인터페이스는 텍스트 서비스에 의해 구현되며 TSF 관리자가 텍스트 서비스를 활성화하고 비활성화하는 데 사용됩니다.
ITfTextInputProcessorEx ITfTextInputProcessorEx 인터페이스는 텍스트 서비스에 의해 구현되며 TSF 관리자가 텍스트 서비스를 활성화하고 비활성화하는 데 사용됩니다.
ITfTextLayoutSink ITfTextLayoutSink 인터페이스는 애플리케이션의 컨텍스트 레이아웃 변경을 지원합니다. IID_ITfTextLayoutSink ITfSource::AdviseSink를 호출하여 이 조언 싱크를 설치합니다. 텍스트 서비스는 필요에 따라 이 인터페이스를 구현할 수 있습니다.
ITfThreadFocusSink ITfThreadFocusSink 인터페이스는 스레드가 UI 포커스를 받거나 분실할 때 알림을 수신하도록 애플리케이션 또는 TSF 텍스트 서비스에 의해 구현됩니다.
ITfThreadMgr ITfThreadMgr은 TSF 관리자가 구현하는 기본 개체를 정의합니다. ITfThreadMgr은 애플리케이션 및 텍스트 서비스에서 텍스트 서비스를 활성화 및 비활성화하고, 문서 관리자를 만들고, 문서 컨텍스트 포커스를 유지하는 데 사용됩니다.
ITfThreadMgr2 ITfThreadMgr2는 TSF 관리자가 구현하는 기본 개체를 정의합니다. ITfThreadMgr2는 애플리케이션 및 텍스트 서비스에서 텍스트 서비스를 활성화 및 비활성화하고, 문서 관리자를 만들고, 문서 컨텍스트 포커스를 유지하는 데 사용됩니다.
ITfThreadMgrEventSink ITfThreadMgrEventSink 인터페이스는 애플리케이션 또는 TSF 텍스트 서비스에 의해 구현되어 특정 스레드 관리자 이벤트의 알림을 받습니다. TSF 관리자 ITfSource::AdviseSink에 IID_ITfThreadMgrEventSink 호출하여 이 조언 싱크를 설치합니다.
ITfThreadMgrEx ITfThreadMgrEx 인터페이스는 애플리케이션에서 일부 플래그를 사용하여 텍스트 서비스를 활성화하는 데 사용됩니다. ITfThreadMgrEx는 ITfThreadMgr에서 QI로 가져올 수 있습니다.

ITfToolTipUIElement ITfToolTipUIElement 인터페이스는 UI에 도구 설명을 표시하려는 텍스트 서비스에 의해 구현됩니다.
ITfTransitoryExtensionSink ITfTransitoryExtensionSink 인터페이스는 Transitory Extension dim을 사용하는 애플리케이션에 의해 구현됩니다. 애플리케이션은 이 싱크 인터페이스를 사용하여 일시적인 확장에서 발생한 변경 내용을 추적할 수 있습니다.
ITfTransitoryExtensionUIElement ITfTransitoryExtensionUIElement 인터페이스는 일시적인 확장의 UI를 제공하는 TSF 관리자에 의해 구현됩니다.
ITfUIElement ITfUIElement 인터페이스는 UIElement 개체의 기본 인터페이스이며 텍스트 서비스에 의해 구현됩니다.
ITfUIElementMgr ITfUIElementMgr 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션 또는 텍스트 서비스에서 사용됩니다. 애플리케이션 및 텍스트 서비스는 IID_ITfUIElementMgr ITfThreadMgr::QueryInterface에서 이 인터페이스를 가져올 수 있습니다.
ITfUIElementSink ITfUIElementSink 인터페이스는 UI 요소가 변경될 때 알림을 수신하기 위해 애플리케이션에 의해 구현됩니다.
IUIManagerEventSink IUIManagerEventSink 인터페이스는 IME UI 통합을 지원하는 앱에 의해 구현되어 IME UI 모양에 대한 알림을 받습니다.

구조체

TF_DA_COLOR TF_DA_COLOR 구조체에는 텍스트 범위의 표시 특성에 사용되는 색 데이터가 포함되어 있습니다.

TF_DISPLAYATTRIBUTE TF_DISPLAYATTRIBUTE 구조에는 텍스트를 렌더링하기 위한 표시 특성 데이터가 포함됩니다.
TF_HALTCOND TF_HALTCOND 구조체는 범위 이동 조건을 포함하는 데 사용됩니다.
TF_INPUTPROCESSORPROFILE 이 구조에는 입력 프로세서 프로필에 대한 데이터가 포함됩니다.
TF_LANGBARITEMINFO TF_LANGBARITEMINFO 구조체는 언어 모음 항목에 대한 정보를 보관하는 데 사용됩니다.
TF_LANGUAGEPROFILE TF_LANGUAGEPROFILE 구조에는 언어 프로필에 대한 정보가 포함됩니다.
TF_LBBALLOONINFO TF_LBBALLOONINFO 구조에는 언어 막대 풍선 항목에 대한 정보가 포함됩니다.
TF_LMLATTELEMENT TF_LMLATTELEMENT 구조체에는 격자 요소에 대한 정보가 포함됩니다. 격자 요소는 음성 인식에 사용됩니다. 이 구조체는 IEnumTfLatticeElements::Next 메서드와 함께 사용됩니다.
TF_PERSISTENT_PROPERTY_HEADER_ACP TF_PERSISTENT_PROPERTY_HEADER_ACP 구조체는 속성 헤더 데이터를 제공하는 데 사용됩니다.
TF_PRESERVEDKEY TF_PRESERVEDKEY 구조체는 유지된 키를 나타냅니다.
TF_PROPERTYVAL TF_PROPERTYVAL 구조에는 속성 값 데이터가 포함됩니다. 이 구조체는 IEnumTfPropertyValue::Next 메서드와 함께 사용됩니다.
TF_SELECTION TF_SELECTION 구조에는 텍스트 선택 데이터가 포함됩니다.
TF_SELECTIONSTYLE TF_SELECTIONSTYLE 구조체는 선택 영역의 스타일을 나타냅니다.

TS_ATTRVAL TS_ATTRVAL 구조에는 문서 특성 값이 포함됩니다.
TS_RUNINFO TS_RUNINFO 구조체는 텍스트 실행 데이터의 속성을 지정합니다.
TS_SELECTION_ACP TS_SELECTION_ACP 구조에는 ACP 기반 텍스트 선택 데이터가 포함됩니다.
TS_SELECTION_ANCHOR TS_SELECTION_ANCHOR 구조에는 앵커 기반 텍스트 선택 데이터가 포함됩니다.
TS_SELECTIONSTYLE TS_SELECTIONSTYLE 구조체는 선택 영역의 스타일을 나타냅니다.
TS_STATUS TS_STATUS 구조에는 문서 상태 데이터가 포함됩니다.
TS_TEXTCHANGE TS_TEXTCHANGE 구조에는 텍스트 변경 데이터가 포함됩니다.

Windows 셸

Windows Shell 기술 개요입니다.

Windows 셸을 개발하려면 다음 헤더가 필요합니다.

- appmgmt.h
- appnotify.h
- cpl.h
- credentialprovider.h
- dimm.h
- imagetranscode.h
- inputpanelconfiguration.h
- intsafe.h
- intshcut.h
- mobsync.h
- objectarray.h
- pathcch.h
- profinfo.h
- propkeydef.h
- scrnsave.h
- shappmgr.h
- shdeprecated.h
- shidfact.h
- shimgdata.h
- shlwapi.h
- shtypes.h
- storageprovider.h
- syncmgr.h
- thumbcache.h
- thumbnailstreamcache.h
- tlogstg.h
- userenv.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 셸

클래스

테이블 확장

CItemIDFactory 셸 데이터 원본과 상호 작용하기 위한 메서드를 노출합니다.

열거 목록

테이블 확장

_BROWSERFRAMEOPTIONS 메서드 IBrowserFrameOptions::GetFrameOptions와 함께 사용됩니다.
_EXPCMDSTATE EXPCMDSTATE 값은 셸 항목의 명령 상태를 나타냅니다.
_EXPLORERPANESTATE 지정된 Windows 탐색기 창의 현재 상태를 가져오기 위해 IExplorerPaneVisibility::GetPaneState에서 사용하는 플래그를 나타냅니다.
_FILEOPENDIALOGOPTIONS 열기 또는 저장 대화 상자에서 사용할 수 있는 옵션 집합을 정의합니다.
_KF_DEFINITION_FLAGS 알려진 특정 폴더 동작을 지정하는 플래그입니다. KNOWNFOLDER_DEFINITION 구조와 함께 사용됩니다.
_KF_REDIRECT_FLAGS IKnownFolderManager::Redirect에서 사용하는 플래그를 사용하여 리디렉션된 폴더에 대한 사용 권한 및 소유권과 같은 알려진 폴더 리디렉션의 세부 정보를 지정합니다.
_KF_REDIRECTION_CAPABILITIES 알려진 폴더의 현재 리디렉션 기능을 지정하는 플래그입니다. IKnownFolder::GetRedirectionCapabilities에서 사용됩니다.
_NSTCITEMSTATE 트리 항목의 상태를 지정합니다. 이러한 값은 INameSpaceTreeControl 인터페이스의 메서드에서 사용됩니다.

_NSTCSTYLE 지정된 네임스페이스 트리 컨트롤의 특징을 설명합니다.
_SHCONTF 열거형에 포함된 항목의 유형을 결정합니다. 이러한 값은 IShellFolder::EnumObjects 메서드와 함께 사용됩니다.
_SHGDNF IShellFolder::GetDisplayNameOf 및 IShellFolder::SetNameOf 메서드와 함께 사용되는 값을 정의하여 해당 메서드에서 사용하는 파일 또는 폴더 이름의 형식을 지정합니다.
_SICHINTF 두 셸 항목을 비교하는 방법을 결정하는 데 사용됩니다. IShellItem::Compare는 이 열거형 형식을 사용합니다.
_SPBEGINF IActionProgress::Begin에서 사용되는 이러한 상수는 사용하거나 사용하지 않도록 설정할 특정 UI 작업을 지정합니다.
_SVGIO IFolderView::Items, IFolderView::ItemCount 및 IShellView::GetItemObject 메서드와 함께 사용하여 컬렉션의 항목을 제한하거나 제어합니다.
_SVSIF 적용할 선택 유형을 지정하기 위해 IFolderView, IFolderView2, IShellView 및 IShellView2에서 사용하는 플래그를 나타냅니다.
_TRANSFER_SOURCE_FLAGS ITransferSource 및 ITransferDestination 인터페이스의 메서드에서 파일 작업을 제어하는 데 사용됩니다.
APPACTIONFLAGS 애플리케이션 게시자가 지원하는 애플리케이션 관리 작업을 지정합니다. 이러한 플래그는 IShellApp::GetPossibleActions에 전달되는 비트 마스크입니다.
APPINFODATAFLAGS IShellApp::GetAppInfo에서 반환할 애플리케이션 정보를 지정합니다. 이러한 플래그는 APPINFODATA 구조체의 dwMask 멤버에 사용되는 비트 마스크입니다.

APPLICATION_VIEW_ORIENTATION 창(앱 보기)의 표시 방향 모드 집합을 정의합니다. IApplicationDesignModeSettings2::GetApplicationViewOrientation 및 IApplicationDesignModeSettings2::SetApplicationViewOrientation에 사용됩니다.
APPLICATION_VIEW_SIZE_PREFERENCE 가능한 일반 창(앱 보기) 크기 기본 설정 집합을 정의합니다. ILaunchSourceViewSizePreference::GetSourceViewSizePreference 및 ILaunchTargetViewSizePreference::GetTargetViewSizePreference에서 사용됩니다.
APPLICATION_VIEW_STATE Windows 스토어 앱의 현재 뷰 상태를 나타냅니다. IApplicationDesignModeSettings::SetApplicationViewState 및 IApplicationDesignModeSettings::IsApplicationViewStateSupported에 사용됩니다.
ASSOCDATA IQueryAssociations::GetData에서 반환할 데이터 형식을 정의하는 데 사용됩니다.
ASSOCIATIONLEVEL 파일 이름 확장명 기본 연결의 원본을 지정합니다. IApplicationAssociationRegistration 인터페이스의 메서드에서 사용됩니다.
ASSOCIATIONTYPE 애플리케이션에 대한 연결 유형을 지정합니다. IApplicationAssociationRegistration 인터페이스의 메서드에서 사용됩니다.
ASSOCKEY IQueryAssociations::GetKey에서 반환할 키의 형식을 지정합니다.
ASSOCSTR IQueryAssociations::GetString에서 반환할 문자열의 형식을 정의하는 데 사용됩니다.
ATTACHMENT_ACTION IAttachmentExecute::P rompt와 함께 사용할 플래그 집합을 제공하여 사용자 확인 시 수행할 작업을 나타냅니다.
ATTACHMENT_PROMPT 표시할 프롬프트 UI의 유형을 나타내기 위해 IAttachmentExecute::P rompt와 함께 사용할 플래그 집합을 제공합니다.

AUTOCOMPLETELISTOPTIONS 자동 완성 목록에 대해 열거되는 개체를 지정합니다.
AUTOCOMPLETEOPTIONS 자동 완성을 둘러싼 옵션에 대해 IAutoComplete2::GetOptions 및 IAutoComplete2::SetOptions에서 사용하는 값을 지정합니다.
BNSTATE Deprecated. IBrowserService::SetNavigateState 및 IBrowserService::GetNavigateState에서 탐색 상태를 지정하는 데 사용됩니다.
CATEGORYINFO_FLAGS CATEGORY_INFO 구조체에 사용할 플래그 집합을 제공합니다.
CATSORT_FLAGS 범주 데이터를 정렬하는 메서드를 지정합니다.
CM_ENUM_FLAGS IColumnManager 인터페이스의 멤버가 요청 중인 열 집합을 지정하는 데 사용되며 현재 표시되는 열 집합 중 하나 또는 모두만 지정합니다.
CM_MASK IColumnManager::SetColumnInfo를 호출하는 동안 설정해야 하는 CM_COLUMNINFO 구조체의 값을 나타냅니다.
CM_SET_WIDTH_VALUE 너비 값을 픽셀 단위로 지정하고 기본값 및 자동 크기 조정에 대한 특별 지원을 포함합니다. CM_COLUMNINFO 구조를 통해 IColumnManager 인터페이스의 멤버에 의해 사용됩니다.
CM_STATE 열 상태 값을 지정합니다. CM_COLUMNINFO 구조를 통해 IColumnManager 인터페이스의 멤버에 의해 사용됩니다.
CREDENTIAL_PROVIDER_ACCOUNT_OPTIONS 자격 증명 공급자가 "다른 사용자" 타일과 연결하기 위해 반환해야 하는 자격 증명 유형을 나타냅니다. ICredentialProviderUserArray_GetAccountOptions 사용됩니다.
CREDENTIAL_PROVIDER_CREDENTIAL_FIELD_OPTIONS

로그온 또는 자격 증명 UI의 단일 필드에 대한 사용자 지정 옵션을 제공합니다.
CREDENTIAL_PROVIDER_FIELD_INTERACTIVE_STATE 필드의 상태 및 사용자가 필드와 상호 작용하는 방법을 설명합니다. 자격 증명 공급자는 다양한 대화형 상태로 필드를 표시할 수 있습니다.
CREDENTIAL_PROVIDER_FIELD_STATE 자격 증명 UI에서 단일 필드의 상태를 지정합니다.
CREDENTIAL_PROVIDER_FIELD_TYPE 자격 증명 필드의 형식을 지정합니다. CREDENTIAL_PROVIDER_FIELD_DESCRIPTOR 사용됩니다.
CREDENTIAL_PROVIDER_GET_SERIALIZATION_RESPONSE 자격 증명 공급자가 자격 증명을 직렬화하려고 할 때의 응답을 설명합니다.
CREDENTIAL_PROVIDER_STATUS_ICON 표시할 상태 아이콘을 나타냅니다.
CREDENTIAL_PROVIDER_USAGE_SCENARIO 자격 증명 공급자가 지원되는 시나리오를 선언합니다. CPUS(자격 증명 공급자 사용 시나리오)를 사용하면 자격 증명 공급자가 시나리오 간에 고유한 열거형 동작 및 UI 필드 설정을 제공할 수 있습니다.
DATAOBJ_GET_ITEM_FLAGS SHGetItemFromDataObject 함수에서 소스 개체 처리와 관련된 옵션을 지정하는 데 사용하는 값입니다.
DEF_SHARE_ID ISharingConfigurationManager 인터페이스의 메서드에 의해 수행되는 폴더를 지정하는 값입니다.
DEFAULT_FOLDER_MENU_RESTRICTIONS . (DEFAULT_FOLDER_MENU_RESTRICTIONS)
DEFAULTSAVEFOLDERTYPE 기본 저장 위치를 지정합니다.
DESKTOP_WALLPAPER_POSITION 바탕 화면 배경 화면을 표시하는 방법을 지정합니다.
DEVICE_SCALE_FACTOR

스푸핑된 디바이스 배율 인수를 백분율로 나타냅니다. IApplicationDesignModeSettings::SetApplicationViewState 및 IApplicationDesignModeSettings::IsApplicationViewStateSupported에 사용됩니다.
DISPLAY_DEVICE_TYPE 디바이스가 기본 또는 몰입형 디스플레이 유형인지 여부를 나타냅니다.
DROPIMAGETYPE DROPDESCRIPTION 구조와 함께 드롭 이미지를 지정하는 데 사용되는 값입니다.
EXPLORER_BROWSER_FILL_FLAGS 이러한 플래그는 IExplorerBrowser::FillFromObject와 함께 사용됩니다.
EXPLORER_BROWSER_OPTIONS 이러한 플래그는 IExplorerBrowser::GetOptions 및 IExplorerBrowser::SetOptions와 함께 사용됩니다.
FDAP 목록 배치를 지정합니다.
FDE_OVERWRITE_RESPONSE 일반 파일 대화 상자를 사용하여 저장 작업 중에 덮어쓰기 요청에 대한 애플리케이션의 응답을 나타내기 위해 IFileDialogEvents::OnOverwrite 메서드에서 사용하는 값을 지정합니다.
FDE_SHAREVIOLATION_RESPONSE 파일을 열거나 저장할 때 발생하는 공유 위반에 대한 애플리케이션의 응답을 나타내기 위해 IFileDialogEvents::OnShareViolation 메서드에서 사용하는 값을 지정합니다.
FFFP_MODE 일치 조건을 설명합니다. IKnownFolderManager 인터페이스의 메서드에서 사용됩니다.
FILE_USAGE_TYPE 사용 중인 파일을 사용하는 방법을 나타내기 위해 IFileIsInUse::GetUsage에서 사용하는 상수입니다.
FILETYPEATTRIBUTEFLAGS 파일 연결 PROGID 레지스트리 키의 EditFlags 값에 사용되는 FILETYPEATTRIBUTEFLAGS 상수입니다.
FOLDER_ENUM_MODE

IObjectWithFolderEnumMode::GetMode 및 IObjectWithFolderEnumMode::SetMode 메서드에서 폴더의 표시 모드를 가져와서 설정하는 데 사용됩니다.
FOLDERFLAGS 폴더 뷰 옵션을 지정하는 플래그 집합입니다. 플래그는 서로 독립적이며 모든 조합에서 사용할 수 있습니다.
FOLDERLOGICALVIEWMODE IFolderViewSettings::GetViewMode 및 ISearchFolderItemFactory::SetFolderLogicalViewMode에서 뷰 모드를 설명하는 데 사용됩니다.
FOLDERVIEWMODE 폴더 뷰 유형을 지정합니다.
FOLDERVIEWOPTIONS IFolderViewOptions 인터페이스의 메서드에서 Windows 7 이상 시스템에서 기본적으로 지원되지 않는 Windows Vista 옵션을 활성화하고 새 Windows 7 옵션을 비활성화하는 데 사용됩니다.
IESHORTCUTFLAGS 브라우저에서 바로 가기를 처리하는 방법을 지정합니다.
KF_CATEGORY 알려진 폴더 시스템에 등록된 폴더를 분류할 수 있는 범주를 나타내는 값입니다.
KNOWN_FOLDER_FLAG 알려진 폴더에 대한 특수 검색 옵션을 지정하는 상수를 정의합니다. 이러한 값은 병렬 의미가 있는 CSIDL 값을 대체합니다.
LIBRARYFOLDERFILTER 폴더 항목을 필터링하는 옵션을 정의합니다.
LIBRARYMANAGEDIALOGOPTIONS SHShowManageLibraryUI에서 라이브러리를 저장할 때 이름 충돌을 처리하는 옵션을 정의하는 데 사용됩니다.
LIBRARYOPTIONFLAGS 라이브러리 옵션을 지정합니다.
LIBRARYSAVEFLAGS

라이브러리를 저장할 때 이름 충돌을 처리하는 옵션을 지정합니다.
MIMEASSOCIATIONDIALOG_IN_FLAGS MIMEAssociationDialog 함수와 함께 사용하여 실행 방법을 결정합니다.
MONITOR_APP_VISIBILITY 디스플레이에 Windows 스토어 앱 대신 데스크톱 창이 표시되는지 여부를 지정합니다.
NSTCFOLDERCAPABILITIES 트리 항목의 상태를 지정합니다. 이러한 값은 INameSpaceTreeControlFolderCapabilities 인터페이스의 메서드에서 사용됩니다.
NSTCSTYLE2 INameSpaceTreeControl2의 메서드에서 셸 네임스페이스 트리 뷰에서 확장된 표시 스타일을 지정하는 데 사용됩니다.
NWMF INewWindowManager::EvaluateNewWindow에서 사용하는 플래그입니다. 이러한 값은 팝업 창을 표시할지 여부를 결정하는 요소입니다.
PACKAGE_EXECUTION_STATE . (PACKAGE_EXECUTION_STATE)
인식 파일의 인식된 형식을 지정합니다. 이 상수 집합은 AssocGetPerceivedType 함수에 사용됩니다.
PUBAPPINFOFLAGS PUBAPPINFO 구조체에서 유효한 멤버를 지정합니다. 이러한 플래그는 dwMask 멤버에 설정된 비트 마스크이며 IPublishedApp::GetPublishedAppInfo에 전달됩니다.
QUERY_USER_NOTIFICATION_STATE 알림 보내기의 적정성과 관련하여 현재 사용자의 컴퓨터 상태를 지정합니다. SHQueryUserNotificationState에서 사용됩니다.
제한 이러한 플래그는 SHRestricted 함수와 함께 사용됩니다.
SCALE_CHANGE_FLAGS

발생한 크기 조정 변경을 나타내는 데 사용되는 플래그입니다.
SCNRT_STATUS SHChangeNotifyRegisterThread에 대해 비동기 레지스터 및 등록 취소를 사용하거나 사용하지 않도록 설정할지 여부를 나타냅니다.
SECURELOCKCODE Deprecated. 이 열거형은 BASEBROWSERDATA 구조체에서 기본 브라우저의 잠금 아이콘 상태를 나타내는 데 사용됩니다.
파편 사용 통계가 추적되는 항목을 식별하기 위해 pv 매개 변수에서 SHAddToRecentDocs에서 전달한 데이터의 해석을 나타냅니다.
SHARE_ROLE 사용자 또는 공용 폴더에 할당된 액세스 권한을 지정합니다. CreateShare 및 GetSharePermissions에 사용됩니다.
SHCOLSTATE 속성을 처리하는 방법을 설명합니다. 이러한 값은 Shtypes.h에 정의됩니다.
SHELL_LINK_DATA_FLAGS 옵션 설정을 지정합니다. IShellLinkDataList::GetFlags 및 IShellLinkDataList::SetFlags와 함께 사용됩니다.
SHELL_UI_COMPONENT 셸에 필요한 UI 구성 요소의 유형을 식별합니다.
ShellFolderViewOptions ViewOptions 속성에서 반환되는 보기 옵션을 지정합니다.
ShellSpecialFolderConstants 특수 폴더를 식별하는 고유한 시스템 독립적 값을 지정합니다.
ShellWindowFindWindowOptions 셸 창 컬렉션에서 창을 찾는 옵션을 지정합니다.
ShellWindowTypeConstants 셸 창의 유형을 지정합니다.

SHGLOBALCOUNTER 다양한 전역 카운터 또는 공유 변수에 대한 식별자입니다. 각 전역 카운터는 SHGlobalCounterIncrement 및 SHGlobalCounterDecrement를 사용하여 증가하거나 감소할 수 있습니다.
SHREGDEL_FLAGS 항목을 삭제할 기본 키를 나타내는 값 집합을 제공합니다.
SHREGENUM_FLAGS 열거형에 사용할 기본 키를 나타내는 값 집합을 제공합니다.
SHSTOCKICONID SHGetStockIconInfo에서 검색할 주식 시스템 아이콘을 식별하는 데 사용됩니다.
SIGDN IShellItem::GetDisplayName 및 SHGetNameFromIDList를 통해 검색할 항목의 표시 이름 양식을 요청합니다.
SPACTION IActionProgress 인터페이스를 사용하여 진행률을 사용자에게 표시해야 하는 수행 중인 작업에 대해 설명합니다.
SPTEXT IActionProgress 인터페이스에 제공되는 설명 텍스트의 형식을 지정합니다.
STPFLAG ITaskbarList4::SetTabProperties 메서드에서 탭 속성을 지정하는 데 사용됩니다.
SVUIA_STATUS IBrowserService2::_UIActivateView 메서드와 함께 사용하여 브라우저 보기의 상태를 설정합니다.
SYNCMGR_CANCEL_REQUEST 동기화를 취소하라는 사용자의 요청에 대해 설명합니다.
SYNCMGR_CONFLICT_ITEM_TYPE 충돌 항목 유형에 대해 설명합니다.
SYNCMGR_CONTROL_FLAGS

ISyncMgrControl의 특정 메서드에 대해 요청된 작업을 수행하는 방법을 지정합니다.
SYNCMGR_EVENT_FLAGS 동기화 이벤트에 대한 플래그를 지정합니다.
SYNCMGR_EVENT_LEVEL 동기화 센터에 보고되는 이벤트의 유형을 지정합니다.
SYNCMGR_HANDLER_CAPABILITIES 처리기에 대해 수행할 수 있는 작업에 대한 처리기의 기능을 지정합니다.
SYNCMGR_HANDLER_POLICIES 기본 정책에서 벗어나는 동기화 처리기에서 지정한 정책을 열거합니다.
SYNCMGR_HANDLER_TYPE 처리기의 형식을 지정합니다. ISyncMgrHandlerInfo::GetType에서 사용됩니다.
SYNCMGR_ITEM_CAPABILITIES 항목에 대해 수행할 수 있는 작업을 지정합니다.
SYNCMGR_ITEM_POLICIES 그룹 정책에서 사용하도록 설정하거나 사용하지 않도록 설정할 수 있는 방법을 제어하는 항목의 정책을 지정합니다.
SYNCMGR_PRESENTER_CHOICE 사용자가 동기화 관리자 충돌 해결에 대해 어떤 선택을 하는지 설명합니다. ISyncMgrConflictPresenter에서 사용됩니다.
SYNCMGR_PRESENTER_NEXT_STEP 동기화 관리자 충돌 해결에서 발생하는 다음 단계를 설명합니다. ISyncMgrConflictPresenter에서 사용됩니다.
SYNCMGR_PROGRESS_STATUS 동기화 프로세스의 현재 진행 상태를 지정합니다. ISyncMgrSyncCallback::ReportProgress에서 사용됩니다.
SYNCMGR_RESOLUTION_ABILITIES 수행할 기능 및 충돌 해결 활동을 나타냅니다. ISyncMgrResolutionHandler::QueryAbilities와 함께 사용됩니다.

SYNCMGR_RESOLUTION_FEEDBACK 동기화 관리자 확인 피드백에 대해 설명합니다. ISyncMgrResolutionHandler에서 사용됩니다.
SYNCMGR_SYNC_CONTROL_FLAGS ISyncMgrControl::StartHandlerSync 및 ISyncMgrControl::StartItemSync에서 사용하는 플래그를 나타냅니다.
SYNCMGRFLAG SYNCMGRFLAG 열거형 값은 동기화 이벤트가 시작된 방법을 나타내기 위해 ISyncMgrSynchronize::Initialize 메서드에 사용됩니다.
SYNCMGRHANDLERFLAGS SYNCMGRHANDLERINFO 구조체에서 현재 처리기에 적용되는 플래그로 사용됩니다.
SYNCMGRINVOKEFLAGS SYNCMGRINVOKEFLAGS 열거형 값은 ISyncMgrSynchronizeInvoke::UpdateItems 메서드에서 동기화 관리자를 호출하는 방법을 지정합니다.
SYNCMGRITEMFLAGS SYNCMGRITEM 구조체의 현재 항목에 대한 정보를 지정합니다.
SYNCMGRLOGLEVEL SYNCMGRLOGLEVEL 열거형 값은 ISyncMgrSynchronizeCallback::LogError 메서드에서 사용할 오류 수준을 지정합니다.
SYNCMGRREGISTERFLAGS SYNCMGRREGISTERFLAGS 열거형 값은 ISyncMgrRegister 인터페이스의 메서드에서 처리기가 알림을 받도록 등록된 이벤트를 식별하는 데 사용됩니다.
SYNCMGRSTATUS ISyncMgrSynchronize::SetItemStatus 메서드에서 항목의 업데이트된 상태를 지정하는 데 사용됩니다.
tagDESKBANDCID 이러한 명령 ID는 IOleCommandTarget::Exec를 사용하여 밴드 개체의 컨테이너로 보낼 수 있습니다.
tagSFBS_FLAGS StrFormatByteSizeEx 함수가 재생되지 않은 숫자의 반올림을 처리하는 방법을 지정합니다.

THUMBBUTTONFLAGS THUMBBUTTON에서 단추의 특정 상태 및 동작을 제어하는 데 사용됩니다.
THUMBBUTTONMASK THUMBBUTTON 구조체에서 유효한 데이터를 포함하는 해당 구조체의 멤버를 지정하는 데 사용됩니다.
ThumbnailStreamCacheOptions IThumbnailStreamCache 인터페이스에서 사용하는 캐시 옵션을 정의합니다.
TRANSLATEURL_IN_FLAGS TRANSLATEURL_IN_FLAGS 열거형 값은 TranslateURL 함수와 함께 사용하여 실행 방법을 결정합니다.
UNDOCK_REASON 도킹된 접근성 앱 창이 도킹 해제된 이유를 나타내는 값입니다. IAccessibilityDockingServiceCallback::Undocked에서 사용됩니다.
URL_SCHEME URL 구성표를 지정하는 데 사용됩니다.
URLASSOCIATIONDIALOG_IN_FLAGS URLASSOCIATIONDIALOG_IN_FLAGS 열거형 값은 URLAssociationDialog와 함께 사용하여 실행 방법을 결정합니다.
VPCOLORFLAGS 색의 사용을 지정합니다. IVisualProperties 메서드에서 사용됩니다.
VPWATERMARKFLAGS 워터마크 플래그를 지정합니다. IVisualProperties::SetWatermark에서 사용됩니다.
WTS_CONTEXTFLAGS 썸네일 추출의 컨텍스트를 지정합니다. IThumbnailSettings::SetContext에서 사용됩니다.
WTS_FLAGS IThumbnailCache::GetThumbnail에서 썸네일 이미지의 추출 및 표시 옵션을 지정하는 데 사용되는 값입니다.

Functions

테이블 확장

_CancelPendingNavigationAsync Deprecated. 파생 클래스가 보류 중인 탐색을 기본 클래스에 취소하도록 요청할 수 있습니다.
_CancelPendingView Deprecated. 파생 클래스가 보류 중인 뷰를 기본 클래스에 취소하도록 요청할 수 있도록 합니다.
_CloseAndReleaseToolbars Deprecated. 파생 클래스에서 호스트하는 브라우저 도구 모음의 닫기를 요청합니다.
_DisableModeless Deprecated. 파생 클래스에서 모달 UI가 표시되는지 여부를 기본 클래스에 요청할 수 있습니다. 모달 UI는 브라우저 창에서 최소화 및 닫기 동작을 차단합니다.
_ExecChildren Deprecated. 파생 클래스가 기본 클래스를 사용하는 대신 IOleCommandTarget::Exec 메서드를 통해 명령을 직접 실행하도록 설정합니다.
_FindTBar Deprecated. COM(구성 요소 개체 모델) ID 규칙을 기반으로 브라우저 도구 모음 항목의 인덱스 반환
_get_itbLastFocus Deprecated. 포커스가 있는 마지막 도구 모음 또는 보기의 ID를 가져옵니다.
_GetBorderDWHelper Deprecated. GetBorderDW 구현을 위한 도우미 메서드입니다.
_GetEffectiveClientArea Deprecated. IBrowserService2::_GetViewBorderRect 사용하여 브라우저 보기의 차원을 협상합니다.
_GetToolbarCount Deprecated. 브라우저 창의 도구 모음 수를 반환합니다.
_GetToolbarItem

Deprecated. 도구 모음에서 특정 항목을 가져옵니다.
_GetViewBorderRect Deprecated. IBrowserService2::_GetEffectiveClientArea 사용하여 브라우저 보기의 크기와 위치를 협상합니다.
_초기화 Deprecated. 기본 클래스와 파생 클래스 간의 상태 초기화를 조정합니다.
_LoadToolbars Deprecated. 브라우저 도구 모음의 저장된 상태를 로드합니다.
_MaySaveChanges Deprecated. 기본 클래스가 브라우저 뷰를 닫기 전에 변경 내용을 저장해야 하는지 여부를 확인할 수 있습니다.
_NavigateToPidl Deprecated. 기본 클래스를 동기적으로 새 위치로 이동합니다.
_NewEnum Shell 창 컬렉션에 대한 열거자를 검색합니다.
_OnFocusChange Deprecated. 파생 클래스의 브라우저 도구 모음과 해당 뷰 간에 포커스가 이동하면 기본 클래스와 파생 클래스 간의 포커스를 조정합니다.
_PauseOrResumeView Deprecated. 파생 클래스에서 기본 클래스를 일시 중지(예: 최소화 작업 전)하거나 브라우저 보기를 다시 시작하도록 요청할 수 있습니다.
_PositionViewWindow Deprecated. 뷰 크기 협상에 사용됩니다. 이 메서드는 사용 가능한 차원을 결정한 후 _UpdateViewRectSize 의해 호출됩니다.
_put_itbLastFocus Deprecated. 포커스가 있는 마지막 도구 모음 또는 마지막 보기를 설정합니다.
_ResizeAllBorders

Deprecated. (IBrowserService4._ResizeAllBorders)
_ResizeNextBorder Deprecated. 도구 모음의 추가 또는 제거에 따라 브라우저 보기의 테두리 크기를 조정합니다.
_ResizeNextBorderHelper Deprecated. IBrowserService2::_ResizeNextBorder 구현에서 사용하는 도우미 메서드입니다.
_ResizeView Deprecated. IBrowserService2::_UpdateViewRectSize 호출한 다음 IOleInPlaceActiveObject::ResizeBorder를 사용하여 브라우저 보기를 업데이트합니다.
_SaveToolbars Deprecated. 브라우저 도구 모음의 상태를 저장합니다.
_SendChildren Deprecated. 파생 클래스에서 기본 클래스를 사용하는 대신 SendMessage 함수를 통해 직접 메시지를 보낼 수 있습니다.
_SetFocus Deprecated. 도구 모음 또는 브라우저의 보기 창에 포커스를 설정합니다. TranslateAcceleratorSB를 통해 또는 IBrowserService2::v_MayGetNextToolbarFocus 실패할 때 가속기를 변환할 때 호출됩니다.
_SwitchActivationNow Deprecated. 현재 및 보류 중인 브라우저 보기 간에 전환하는 동안 상태 업데이트를 조정합니다.
_TryShell2Rename Deprecated. 브라우저가 리디렉션될 때 현재 브라우저 보기의 이름 바꾸기를 조정합니다.
_UIActivateView Deprecated. 파생 클래스에서 기본 클래스가 브라우저 보기를 업데이트하도록 요청할 수 있습니다.
_UpdateViewRectSize Deprecated. 브라우저 보기 크기 협상에 관련된 다른 함수에 허용되는 브라우저 보기 크기가 변경되었음을 알리기 위해 호출됩니다.
AbortList

변경 내용을 커밋하지 않고 ICustomDestinationList::BeginList에서 시작한 점프 목록 빌드 세션을 중단합니다.
Activate 처리기가 활성화 또는 비활성화되도록 요청합니다. 활성 처리기를 동기화할 수 있습니다. 비활성 처리기는 사용할 수 없습니다.
ActivateApplication 현재 세션에서 일반 시작 계약(Windows.Launch)에 대해 지정된 Windows 스토어 앱을 활성화합니다.
ActivateBackgroundTask 지정된 백그라운드 작업을 활성화합니다.
ActivateForFile 파일 계약에 대해 지정된 Windows 스토어 앱을 활성화합니다(Windows.File).
ActivateForProtocol 프로토콜 계약에 대해 지정된 Windows 스토어 앱을 활성화합니다(Windows.Protocol).
ActivateHandler 처리기를 활성화하거나 비활성화합니다.
ActivatePendingView Deprecated. 브라우저가 현재 보기와 보류 중인 보기 사이를 전환하는 동안 상태 업데이트를 조정합니다.
ActivateTab 작업 표시줄에서 항목을 활성화합니다. 창이 실제로 활성화되지 않았습니다. 작업 표시줄에 있는 창의 항목은 활성으로만 표시됩니다.
ActivateView Deprecated. (IBrowserService4.ActivateView)
AddBand 밴드 사이트 개체에 밴드를 추가합니다.
AddCheckButton 대화 상자에 확인 단추(확인란)를 추가합니다.

AddComboBox 대화 상자에 콤보 상자를 추가합니다.
AddControlItem 대화 상자의 컨테이너 컨트롤에 항목을 추가합니다.
AddCreateFlags CreateProcess 호출에 포함될 추가 플래그를 설정합니다.
AddDataBlock 링크에 데이터 블록을 추가합니다.
AddEditBox 대화 상자에 편집 상자 컨트롤을 추가합니다.
AddEntry Deprecated. 보류 중인 탐색에 대한 새 항목을 여행 로그에 추가합니다.
AddFolder 라이브러리에 폴더를 추가합니다.
AddFromArray IObjectArray에 포함된 개체를 컬렉션에 추가합니다.
AddIDList PIDL(항목 식별자 목록)에 대한 포인터를 결과 폴더에 삽입합니다.
AddItem 결과 폴더에 항목을 추가합니다.
AddItemToSession 현재 동기화 중인 항목 집합에 지정된 항목을 추가합니다.
AddMenu 대화 상자에 메뉴를 추가합니다.
AddObject

AddObject를 변경하거나 사용할 수 없습니다.
AddObject 컬렉션에 단일 개체를 추가합니다.
AddPages 배열을 해당 페이지를 나타내는 PROPSHEETPAGE 구조체에 핸들을 채워 확장 페이지를 마법사에 추가합니다.
AddPages 셸에서 파일 개체에 대해 표시하는 속성 시트에 하나 이상의 페이지를 추가합니다. 셸은 파일 형식에 등록된 각 속성 시트 처리기에 대해 이 메서드를 호출합니다.
AddPlace 사용자가 항목을 열거나 저장할 수 있는 위치 목록에 폴더를 추가합니다.
AddPropertySheetPages 보기 메뉴에서 옵션 속성 시트에 페이지를 추가할 수 있습니다.
AddPushButton 대화 상자에 단추를 추가합니다.
AddRadioButtonList 대화 상자에 옵션 단추(라디오 단추라고도 함) 그룹을 추가합니다.
AddSeparator 컨트롤을 시각적으로 분리할 수 있도록 대화 상자에 구분 기호를 추가합니다.
AddStartupFlags CreateProcess에 전달된 STARTUPINFO 구조에 포함될 추가 플래그입니다.
AddTab 작업 표시줄에 항목을 추가합니다.
AddTask 스케줄러의 백그라운드 큐에 작업을 추가합니다.
AddText

대화 상자에 텍스트 콘텐츠를 추가합니다.
AddToolbar 지정된 IDockingWindow 개체를 프레임에 추가합니다.
AddUserTasks 사용자 지정 점프 목록의 작업 범주에 포함할 항목을 지정합니다.
AdvanceSlideshow 지정된 모니터의 배경 화면을 슬라이드 쇼의 다음 이미지로 전환합니다.
조언하다 디스플레이 변경 알림을 받도록 조언 싱크 개체를 등록합니다.
조언하다 이벤트 콜백을 위해 IExplorerBrowser와의 연결을 시작합니다.
조언하다 대화 상자에서 오는 이벤트를 수신 대기하는 이벤트 처리기를 할당합니다.
조언하다 처리기가 모든 작업에 대한 상태 및 오류 정보를 제공할 수 있도록 합니다.
조언하다 이벤트가 입력 창을 트리거할 때 창을 대신하여 알림을 받도록 앱의 입력 창 처리기 개체를 등록합니다. 이 메서드는 ICoreWindow를 구현하는 개체를 통해 해당 창을 참조한다는 점에서 AdviseWithHWND와 다릅니다.
조언하다 파일 작업의 상태에 대한 알림에 대한 권고 연결을 설정합니다. (ITransferDestination.Advise)
조언하다 파일 작업의 상태에 대한 알림에 대한 권고 연결을 설정합니다. (ITransferSource.Advise)
조언하다 자격 증명 공급자가 콜백 인터페이스를 통해 로그온 UI 또는 자격 증명 UI에서 이벤트를 시작할 수 있습니다.

조언하다 자격 증명이 콜백 인터페이스를 통해 로그온 UI 또는 자격 증명 UI에서 이벤트를 시작할 수 있도록 합니다. 이 메서드는 ICredentialProviderCredential 인터페이스의 다른 메서드 앞에 호출해야 합니다.
AdviseWithHWND 이벤트가 입력 창을 트리거할 때 창을 대신하여 알림을 받도록 앱의 입력 창 처리기 개체를 등록합니다. 이 메서드는 HWND를 통해 창을 참조한다는 점에서 Advise와 다릅니다.
AllowAutoPlay 사용자가 삽입한 미디어를 재생할지 여부와 어떤 제한을 사용하는지 여부를 결정합니다.
AllowViewResize Deprecated. 뷰 크기 조정을 허용할지 여부를 기본 클래스에 알릴 수 있습니다.
레코드 추가 관리되는 개체의 컬렉션에 개체를 추가합니다.
AppendCategory 사용자 지정 점프 목록에 포함할 사용자 지정 범주 및 해당 범주에 포함된 대상을 정의합니다.
AppendFieldComboBoxItem 로그온 UI 또는 자격 증명 UI에 콤보 상자에 추가된 항목이 필요하고 UI를 업데이트해야 한다고 통신합니다.
AppendKnownCategory 자주 사용하는 항목 또는 최근 항목 범주를 사용자 지정 점프 목록에 포함되도록 지정합니다.
AppendRoot 트리의 루트 목록에 셸 항목을 추가합니다.
APPLET_PROC 제어판 애플리케이션의 진입점 역할을 합니다. 라이브러리 정의 콜백 함수입니다.
ApplyProperties 셸의 복사 엔진을 사용하여 항목에 속성 집합을 적용합니다.
ApplyPropertiesToItem

속성 값을 설정할 단일 항목을 선언합니다.
ApplyPropertiesToItem 항목에 속성 변경 내용 집합을 적용합니다.
ApplyPropertiesToItems 공통 속성 값 집합을 적용할 항목 집합을 선언합니다.
AppVisibilityOnMonitorChanged 디스플레이 모드가 변경되었음을 클라이언트에 알깁니다.
ArePrintersShared 이 컴퓨터에 연결된 프린터가 공유되는지 여부를 확인합니다.
AreSnapshotsAvailable 이전 시간에 시스템에서 기록한 Windows Server 2003 볼륨 이미지의 가용성에 대한 쿼리입니다.
ArrangeGrid 이동된 아이콘이 보이지 않는 눈금에 맞도록 정렬합니다.
AssocCreate IQueryAssociations 개체에 대한 포인터를 반환합니다.
AssocCreateForClasses IQueryAssociations 인터페이스를 구현하는 개체를 검색합니다.
AssocGetDetailsOfPropKey 네임스페이스 확장명에서 제공하는 파일 연결 정보를 사용하여 지정된 속성 키의 값을 검색합니다.
AssocGetPerceivedType 확장자를 기반으로 파일의 인식된 형식을 검색합니다.
AssocIsDangerous 파일 형식이 잠재적인 보안 위험으로 간주되는지 여부를 결정합니다.
AssocQueryKeyA 레지스트리에서 파일 또는 프로토콜 연결과 관련된 키를 검색하고 검색합니다. (ANSI)

AssocQueryKeyW 레지스트리에서 파일 또는 프로토콜 연결과 관련된 키를 검색하고 검색합니다. (유니코드)
AssocQueryStringA 레지스트리에서 파일 또는 프로토콜 연결 관련 문자열을 검색하고 검색합니다. (AssocQueryStringA)
AssocQueryStringByKeyA 레지스트리에서 지정된 키부터 파일 연결 관련 문자열을 검색하고 검색합니다. (ANSI)
AssocQueryStringByKeyW 레지스트리에서 지정된 키부터 파일 연결 관련 문자열을 검색하고 검색합니다. (유니코드)
AssocQueryStringW 레지스트리에서 파일 또는 프로토콜 연결 관련 문자열을 검색하고 검색합니다. (AssocQueryStringW)
AutoArrange 자동 정렬을 변경하거나 사용할 수 없습니다.
BeforeCoCreateInstance . (IHandlerActivationHost.BeforeCoCreateInstance)
BeforeCreateProcess . (IHandlerActivationHost.BeforeCreateProcess)
시작 진행률을 사용자에게 표시해야 하는 작업이 시작되면 호출됩니다.
BeginFieldUpdates 로그온 또는 자격 증명 UI의 필드에 대한 일괄 업데이트를 시작합니다.
BeginList 사용자 지정 점프 목록에 대한 빌드 세션을 시작합니다.
BindToConflict IID로 지정된 특정 충돌에 바인딩합니다.
BindToHandler

처리기 ID 값(BHID)에 지정된 대로 항목의 처리기에 바인딩합니다.
BindToHandler 지정된 처리기를 사용하여 개체에 바인딩합니다.
BindToHandler 동기화 센터에서 호출할 때 지정된 동기화 처리기를 인스턴스화합니다.
BindToObject 처리기(일반적으로 특정 항목에 대해 IShellFolder를 구현하는 Shell 폴더 개체)를 검색합니다. 처리기의 생성을 제어하는 선택적 매개 변수는 바인딩 컨텍스트에서 전달됩니다.
BindToStorage 개체의 스토리지 인터페이스에 대한 포인터를 요청합니다.
BrowseObject Windows 탐색기에서 다른 폴더로 이동하도록 알릴 수 있습니다.
BrowseToIDList PIDL(항목 식별자 목록)에 대한 포인터를 찾습니다.
BrowseToObject 개체를 찾습니다.
타다 준비 영역에서 쓰기 가능한 CD로 데이터를 복사하도록 지시합니다.
ByteToChar BYTE 형식의 값을 CHAR 형식의 값으로 변환합니다.
ByteToInt8 BYTE 형식의 값을 INT8 형식의 값으로 변환합니다.
CacheOLEServer Deprecated. 다시 사용할 때 서버를 다시 로드하지 않도록 외부 개체에 대한 참조를 캐시합니다.
콜백 상황에 맞는 메뉴에 콜백 함수를 사용하도록 설정합니다.

CallbackSM 메뉴 밴드 개체에서 메시지를 받습니다.
CanCategorizeOnSCID 열을 범주로 사용할 수 있는지 여부를 결정합니다.
CancelIo 보류 중인 모든 입력/출력(I/O) 작업을 취소된 것으로 표시합니다.
CanContinue 동기화가 취소되었는지 여부를 확인합니다.
CanNavigateNow Deprecated. 탐색이 현재 허용되는지 여부를 나타내는 값을 반환합니다.
CanRenderComposited 책상 밴드가 반투명으로 표시할 수 있음을 나타냅니다.
CaptureThumbnail IHTMLDocument2 문서의 썸네일 표현을 검색합니다.
CDefFolderMenu_Create2 선택한 파일 폴더 개체 그룹에 대한 상황에 맞는 메뉴를 만듭니다.
CheckPolicy 첨부 파일의 실행 정책에 따라 결정을 내리는 데 사용할 수 있는 부울 테스트를 제공합니다.
ChrCmpIA 두 문자 간의 비교를 수행합니다. 비교는 대/소문자를 구분하지 않습니다. (ANSI)
ChrCmpIW 두 문자 간의 비교를 수행합니다. 비교는 대/소문자를 구분하지 않습니다. (유니코드)
CIDLData_CreateFromIDArray CIDLData_CreateFromIDArray 변경되거나 사용할 수 없습니다.
지우기

컬렉션에서 모든 개체를 제거합니다. (IObjectCollection.Clear)
ClearClientData 대화 상자에 지속형 상태 정보를 모두 지우도록 지시합니다.
ClearClientState 클라이언트의 GUID를 기반으로 하는 저장된 상태를 제거합니다. 예를 들어 특정 파일 형식에 대해 프롬프트를 다시 표시해서는 안 됨을 나타내는 확인란을 기반으로 하는 설정이 있을 수 있습니다.
ClearUserAssociations 현재 사용자에 대한 모든 사용자별 연결을 제거합니다. 이로 인해 컴퓨터 기본값(있는 경우)으로의 회귀가 발생합니다. Windows 8에서는 사용할 수 없습니다.
클론 현재 구현되지 않았습니다. (IEnumExplorerCommand.Clone)
클론 현재 상태를 유지하기 위해 열거자 개체의 중복을 요청하는 데 사용됩니다.
클론 현재 항목과 동일한 내용 및 상태를 사용하여 새 항목 열거형 개체를 만듭니다. (IEnumFullIDList.Clone)
클론 현재 항목과 동일한 내용 및 상태를 사용하여 새 항목 열거형 개체를 만듭니다. (IEnumIDList.Clone)
클론 구현되지 않았습니다. 구현되지 않았습니다.
클론 리소스 열거자를 복제합니다.
클론 현재 열거형의 복사본을 가져옵니다.
클론 사용되지 않습니다. IEnumSyncMgrConflict 개체를 복제합니다.
클론

IEnumSyncMgrEvents 개체를 복제합니다.
클론 사용되지 않습니다. IEnumSyncMgrSyncItems 개체를 복제합니다.
클론 동일한 목록을 반복할 현재 열거자와 동일한 상태를 가진 다른 항목 열거자를 만듭니다. 이 메서드를 사용하면 나중에 해당 지점으로 돌아가기 위해 열거형 시퀀스의 점을 기록할 수 있습니다.
클론 Deprecated. 현재 여행 로그의 내용을 복제합니다.
CloneFrame 현재 이미지 또는 프레임의 복제본을 검색합니다.
닫기 이미지 캐시를 닫습니다.
닫기 대화 상자를 닫습니다.
CloseDW 도킹 창 개체가 프레임에서 제거될 예정임을 알립니다. 현재 도킹 창 개체는 영구 정보를 저장해야 합니다.
CloseFile 현재 사용 중인 파일을 닫습니다.
CollapseAll 지정된 트리의 모든 항목을 축소합니다.
ColorAdjustLuma RGB 값의 광도를 변경합니다. 색조와 채도는 영향을 받지 않습니다.
ColorHLSToRGB 색을 HLS(색조 채도)에서 RGB 형식으로 변환합니다.
ColorRGBToHLS 색을 RGB에서 HLS(색조 채도) 형식으로 변환합니다.

ColumnClick 열을 다시 정렬합니다.
CommandLineToArgvW 유니코드 명령줄 문자열을 구문 분석하고 표준 C 런타임 argv 및 argc 값과 유사한 방식으로 이러한 인수의 수와 함께 명령줄 인수에 대한 포인터 배열을 반환합니다.
CommandLinkClicked 로그온 UI 및 자격 증명 UI가 링크를 클릭했음을 나타낼 수 있도록 합니다.
커밋 pdwLoc에서 지정한 내용을 스토리지에 씁니다.
커밋 라이브러리 업데이트를 기존 라이브러리 설명 파일에 커밋합니다.
CommitItem 지정된 항목을 처리기 동기화 집합의 멤버로 확인하고 UI에 표시되어야 하는지 확인합니다.
CommitList ICustomDestinationList::BeginList에 대한 호출로 시작된 점프 목록이 완료되고 표시할 준비가 되었음을 선언합니다.
비교 두 IShellItem 개체를 비교합니다.
CompareCategory 항목 식별자 목록에서 두 항목의 상대 순서를 결정하므로 UI에서 결정합니다.
CompareIDs 항목 식별자 목록이 지정된 경우 두 파일 개체 또는 폴더의 상대 순서를 결정합니다.
ComputeApplicationSize 스푸핑된 설정의 현재 집합에 따라 Windows 스토어 앱의 크기를 가져옵니다.
ComputerInfoChanged

지원되지 않습니다. 이 인터페이스는 이후 버전의 Windows에 없을 수 있습니다. (IComputerInfoChangeNotify.ComputerInfoChanged)
ConfirmEncryptionLoss 이 작업에 대해 암호화 손실이 허용되는지 확인하는 메시지를 사용자에게 표시합니다.
ConfirmOverwrite 기존 항목을 덮어쓸 수 있음을 확인하는 메시지를 사용자에게 표시합니다.
연결 IConnectableCredentialProviderCredential 개체를 연결합니다. 이 메서드는 사용자가 사전Logon-Access 공급자 화면 내에서 제출 단추를 클릭한 후 ICredentialProviderCredential::GetSerialization이 호출되기 전에 호출됩니다.
ConnectToConnectionPoint 클라이언트의 싱크와 연결 지점 컨테이너 간의 연결을 설정하거나 종료합니다.
CopyCallback 셸에서 폴더 또는 프린터 개체를 이동, 복사, 삭제 또는 이름을 바꿀 수 있는지 여부를 결정합니다. (ANSI)
CopyCallback 셸에서 폴더 또는 프린터 개체를 이동, 복사, 삭제 또는 이름을 바꿀 수 있는지 여부를 결정합니다. (유니코드)
CopyDataBlock 링크의 데이터 블록 복사본을 검색합니다.
CopyItem 지정된 대상에 복사할 단일 항목을 선언합니다.
CopyItems 지정된 대상에 복사할 항목 집합을 선언합니다.
CountEntries Deprecated. 이동 로그의 항목 수를 생성합니다.
CountTasks 스케줄러의 큐에서 소유자 ID가 동일한 작업을 계산합니다.

CreateAppContainerProfile AppContainer에 대한 사용자별 앱별 프로필을 만듭니다.
CreateBand 지정된 밴드를 만듭니다.
CreateBrowserPropSheetExt Deprecated. 파생 클래스에서 기본 클래스에 Folder Options 속성 시트를 추가할 수 있습니다.
CreateCategory 범주 개체를 만듭니다.
CreateEnumIDListFromContents 보기의 내용에서 ID 목록의 열거자를 만듭니다.
CreateEnvironmentBlock 지정된 사용자의 환경 변수를 검색합니다. 그런 다음 이 블록을 CreateProcessAsUser 함수에 전달할 수 있습니다.
CreateImageFromFile 지정된 파일을 기반으로 IShellImageData 인터페이스의 인스턴스를 만듭니다.
CreateImageFromStream 지정된 파일 스트림을 기반으로 IShellImageData 인터페이스의 인스턴스를 만듭니다.
CreateInvoker 현재 선택 영역에서 연결된 처리기를 호출할 수 있도록 하는 개체를 검색합니다. 호출자에는 현재 선택이 지원되는지 여부를 확인하는 기능이 포함됩니다.
CreateIShellImageData IShellImageData 인터페이스의 인스턴스를 만듭니다.
CreateItem 지정된 파일을 만듭니다.
CreateItemID 제공된 데이터에서 ItemID를 만듭니다.

CreateObject 지정된 클래스의 로컬 개체를 만들고 개체의 지정된 인터페이스에 대한 포인터를 반환합니다.
CreateProfile 새 사용자 프로필을 만듭니다.
CreateResource 지정된 리소스를 만듭니다.
createSession 동기화 센터에 지정된 항목의 동기화가 시작되었음을 알립니다.
CreateShare 사용자 또는 공용 폴더를 공유합니다. 폴더가 이미 공유된 경우 이 메서드는 공유 상태를 업데이트합니다.
CreateViewObject 폴더 개체에서 정보를 가져오거나 상호 작용하는 데 사용할 수 있는 개체를 요청합니다.
CreateViewWindow 보기 창을 만듭니다. Windows 탐색기의 오른쪽 창 또는 폴더 창의 클라이언트 창일 수 있습니다.
CreateViewWindow Deprecated. 새 브라우저 보기 창을 만들 때 상태 업데이트를 조정합니다.
CreateViewWindow2 새 셸 보기 창 만들기를 요청하는 데 사용됩니다. Windows 탐색기의 오른쪽 창 또는 폴더 창의 클라이언트 창일 수 있습니다.
CreateViewWindow3 새 셸 보기 창 만들기를 요청합니다. 보기는 Windows 탐색기의 오른쪽 창 또는 폴더 창의 클라이언트 창일 수 있습니다. 이 메서드는 CreateViewWindow2를 대체합니다.
CredentialsChanged 로그온 UI 또는 자격 증명 UI에 열거된 자격 증명 목록이 변경되었음을 알릴 수 있습니다.
DAD_AutoScroll 이미지를 끌 때 창을 스크롤합니다.

DAD_DragEnterEx 끌기 작업 중에 지정된 창에 대한 업데이트를 잠그고 창 내의 지정된 위치에 끌기 이미지를 표시합니다. (DAD_DragEnterEx)
DAD_DragEnterEx2 끌어서 놓기 작업 중에 지정된 창에 대한 업데이트를 잠그고 끌어서 놓기 이미지를 창 내의 지정된 위치에 표시합니다.
DAD_DragLeave DAD_DragEnterEx 함수에 의해 잠긴 창의 잠금을 해제합니다.
DAD_DragMove 끌어서 놓기 작업 중에 끌 이미지를 이동합니다.
DAD_SetDragImage 끌기 이미지를 설정합니다.
DAD_ShowDragImage 끌 이미지를 표시하거나 숨깁니다. (DAD_ShowDragImage)
디코딩 이미지 파일을 디코딩하고 상태를 설정합니다.
DEFINE_PROPERTYKEY FMTID(형식 식별자) 및 PID(속성 식별자)를 속성 키를 나타내는 PROPERTYKEY 구조체로 압축하는 데 사용됩니다.
DefScreenSaverProc 화면 보호기 애플리케이션에서 처리하지 않는 모든 메시지에 대한 기본 처리를 제공합니다.
DefSubclassProc 창의 하위 클래스 체인에서 다음 처리기를 호출합니다. 하위 클래스 체인의 마지막 처리기는 창에 대한 원래 창 프로시저를 호출합니다.
Delete 동기화 항목을 삭제합니다.

DeleteAppContainerProfile 지정된 사용자별 앱별 프로필을 삭제합니다.
DeleteFieldComboBoxItem 콤보 상자에서 항목을 삭제하고 UI를 업데이트해야 한다는 로그온 UI 또는 자격 증명 UI와 통신합니다.
DeleteItem 삭제할 단일 항목을 선언합니다.
DeleteItems 삭제할 항목 집합을 선언합니다.
DeleteList 지정된 애플리케이션에 대한 사용자 지정 점프 목록을 삭제합니다.
DeleteLogError 동기화 관리자 상태 대화 상자의 오류 탭에서 이전에 기록된 ErrorInformation, 경고 또는 오류 메시지를 삭제하기 위해 등록된 애플리케이션의 처리기에서 호출됩니다.
DeleteProfileA 지정된 컴퓨터에서 사용자 프로필 및 모든 사용자 관련 설정을 삭제합니다. 사용자의 프로필을 삭제하려면 호출자에게 관리자 권한이 있어야 합니다. (ANSI)
DeleteProfileW 지정된 컴퓨터에서 사용자 프로필 및 모든 사용자 관련 설정을 삭제합니다. 사용자의 프로필을 삭제하려면 호출자에게 관리자 권한이 있어야 합니다. (유니코드)
DeleteShare 사용자 또는 공용 폴더에서 공유를 제거합니다.
DeleteTab 작업 표시줄에서 항목을 삭제합니다.
DeriveAppContainerSidFromAppContainerName 지정된 프로필의 SID를 가져옵니다.
DeriveRestrictedAppContainerSidFromAppContainerSidAndRestrictedName

DeriveRestrictedAppContainerSidFromAppContainerSidAndRestrictedName은 나중에 사용하도록 예약되어 있습니다.
DeskBandRegistrationChanged deskband 등록 캐시를 새로 고칩니다.
Destroy 브라우저를 삭제합니다.
DestroyBand 지정된 밴드를 제거합니다.
DestroyEnvironmentBlock CreateEnvironmentBlock 함수에서 만든 환경 변수를 해제합니다.
DestroyViewWindow 보기 창을 삭제합니다.
분리 ISharedBitmap 개체에 포함된 비트맵을 검색하고 포함된 비트맵이 공유 메모리에 있는 경우 복사본을 반환합니다.
DisableDebugging 지정된 패키지의 프로세스에 대한 디버그 모드를 사용하지 않도록 설정합니다.
DiscardEdit 이미지에 대한 편집 내용을 삭제합니다.
분리하다 IConnectableCredentialProviderCredential 개체의 연결을 끊습니다.
DisplayName IShellImageData가 파일 경로에서 초기화된 경우 파일의 이름을 가져옵니다. 그렇지 않으면 데이터 스트림의 이름을 가져옵니다.
DisplayParseError Deprecated. IBrowserService::IEParseDisplayName에서 성공적으로 구문 분석하지 못한 URL을 표시합니다.

DLLGETVERSIONPROC 애플리케이션이 DLL 관련 버전 정보를 가져올 수 있도록 많은 Windows 셸 DLL에 의해 구현됩니다.
DllInstall DLL에 대한 설치 및 설정을 처리합니다.
DoContextMenuPopup DoContextMenuPopup은 Windows Server 2003을 기준으로 더 이상 사용할 수 없습니다.
DoEnvironmentSubstA 하나 이상의 환경 변수에 대한 참조가 포함된 입력 문자열을 구문 분석하고 완전히 확장된 값으로 대체합니다. (ANSI)
DoEnvironmentSubstW 하나 이상의 환경 변수에 대한 참조가 포함된 입력 문자열을 구문 분석하고 완전히 확장된 값으로 대체합니다. (유니코드)
DoPreview 미리 보기 처리기가 이전 Initialize 메서드 호출에 지정된 원본에서 데이터를 로드하고 미리 보기 창에 렌더링을 시작하도록 지시합니다.
DoRename 현재 선택 영역에서 이름 바꾸기 작업을 시작합니다.
DragAcceptFiles 창에서 삭제된 파일을 허용하는지 여부를 등록합니다.
DragEnter 끌어서 놓기 대상의 IDropTarget::D ragEnter 메서드가 호출되었음을 끌기 이미지 관리자에게 알립니다.
DragFinish 애플리케이션에 파일 이름을 전송하는 데 사용하기 위해 시스템이 할당한 메모리를 해제합니다.
DragLeave 끌어서 놓기 대상의 IDropTarget::D ragLeave 메서드가 호출되었음을 끌기 이미지 관리자에게 알립니다.
DragOver

끌어서 놓기 대상의 IDropTarget::D ragOver 메서드가 호출되었음을 끌기 이미지 관리자에게 알립니다.
DragQueryFileA 끌어서 놓기 작업의 성공으로 인해 삭제된 파일의 이름을 검색합니다. (ANSI)
DragQueryFileW 끌어서 놓기 작업의 성공으로 인해 삭제된 파일의 이름을 검색합니다. (유니코드)
DragQueryPoint 끌어서 놓기 작업 중에 파일을 삭제할 때 마우스 포인터의 위치를 검색합니다.
그리다 디코딩된 이미지를 그립니다.
DriveType DriveType 함수는 드라이브 번호에 따라 드라이브 유형을 결정합니다. (DriveType 함수)
DriveType DriveType 함수는 드라이브 번호에 따라 드라이브 유형을 결정합니다. (DriveType)
삭제 끌어서 놓기 대상의 IDropTarget::D rop 메서드가 호출되었음을 끌기 이미지 관리자에게 알립니다.
DuplicateIcon 지정된 아이콘의 복제본을 만듭니다.
DWordPtrAdd DWORD_PTR 형식의 두 값을 추가합니다.
DWordPtrMult DWORD_PTR 형식의 한 값을 다른 값으로 곱합니다.
DWordPtrSub DWORD_PTR 형식의 한 값을 다른 값에서 뺍니다.
Enable 자동 완성을 사용하거나 사용하지 않도록 설정합니다.

Enable 바탕 화면 배경을 사용하거나 사용하지 않도록 설정합니다.
Enable 활성 처리기를 사용하거나 사용하지 않도록 설정하도록 요청합니다. 활성화된 처리기를 동기화할 수 있으며 비활성화된 처리기는 동기화할 수 없습니다.
Enable 동기화 항목을 사용하거나 사용하지 않도록 설정합니다.
EnableDebugging 지정된 패키지의 프로세스에 대해 디버그 모드를 사용하도록 설정합니다.
EnableFocusTracking 클라이언트 프로세스가 터치 키보드의 호출 및 해제 의미 체계를 제어하는 Windows 스토어 앱에 대한 포커스 추적 메커니즘을 옵트인할 수 있습니다.
EnableHandler 처리기를 사용하거나 사용하지 않도록 설정합니다.
EnableItem 지정된 처리기에서 관리하는 동기화 항목을 사용하거나 사용하지 않도록 설정합니다.
EnableModeless 모덜리스 대화 상자를 사용하거나 사용하지 않도록 설정합니다. 이 메서드는 현재 구현되지 않았습니다.
EnableModeless 모든 대화 상자가 PrepareForSync 및 Synchronize 메서드 내에서 표시되는 전후에 등록된 애플리케이션에서 호출됩니다.
EnableModelessSB Windows 탐색기에서 모덜리스 대화 상자를 사용하거나 사용하지 않도록 설정합니다.
EnableOpenDropDown 대화 상자의 열기 또는 저장 단추에서 드롭다운 목록을 사용하도록 설정합니다.
끝

이 진행률 구현과 관련된 작업이 종료되었음을 나타냅니다.
EndFieldUpdates BeginFieldUpdates에서 시작한 일괄 처리 업데이트를 완료하고 커밋합니다.
EndOperation 데이터 개체에 비동기 데이터 추출이 종료되었음을 알 수 있습니다.
EndVisualGroup 대화 상자에서 시각적 개체 그룹에 요소 추가를 중지합니다.
EnsureItemVisible 지정된 항목이 표시되는지 확인합니다.
EnterFolder 네임스페이스 워크 중에 폴더를 입력하려고 할 때 호출됩니다. 검색된 항목의 초기화에는 이 메서드를 사용합니다.
EnterFolder 폴더가 파일 작업의 대상임을 알 수 있습니다.
EnumApps 지정된 범주에 대해 애플리케이션 게시자가 게시한 모든 애플리케이션을 열거하기 위한 열거자를 만듭니다.
EnumBands 밴드 사이트의 밴드를 열거합니다.
EnumCategories 범주를 나타내는 GUID 목록의 열거자를 가져옵니다.
EnumConflicts 제공된 동기화 처리기 및 동기화 항목으로 범위가 지정된 충돌을 열거합니다.
EnumerateBackgroundTasks 지정된 패키지에서 제공하는 백그라운드 작업을 가져옵니다.
EnumItems

배열에 있는 항목의 열거자를 가져옵니다.
EnumObjects 항목 식별자 열거형 개체를 만들고 해당 IEnumIDList 인터페이스를 반환하여 클라이언트가 폴더의 내용을 확인할 수 있도록 합니다. 그런 다음 해당 인터페이스에서 지원하는 메서드를 사용하여 폴더의 내용을 열거할 수 있습니다.
EnumReady 뷰의 항목 열거가 완료되었음을 구현자에게 알립니다.
EnumResources 리소스 열거자 개체를 가져옵니다.
EnumSearches 클라이언트가 사용 가능한 검색 개체를 열거할 수 있는 인터페이스에 대한 포인터를 요청합니다.
EnumSubCommands 명령의 하위 명령에 대한 열거자를 검색합니다.
EnumSyncMgrItems 등록된 애플리케이션에서 처리하는 항목에 대한 ISyncMgrEnumItems 인터페이스를 가져옵니다.
EstablishConnection 네트워크 연결이 필요할 때 등록된 애플리케이션의 처리기에서 호출됩니다.
EvaluateNewWindow 표시하려는 새 창에 대한 데이터를 허용하고 사용자의 기본 설정에 따라 해당 창을 열 수 있는지 여부를 결정합니다.
실행 첨부 파일에서 작업을 실행합니다.
실행 선택한 항목에서 동사를 호출합니다. 이 인터페이스의 다른 메서드를 호출한 후 이 메서드를 호출합니다.
넓히다 자동 완성 클라이언트가 해당 네임스페이스에서 지정된 항목과 연결된 후보 문자열을 생성할 것을 요청합니다.

ExpandEnvironmentStringsForUserA 지정된 사용자에 대해 설정된 환경 블록을 사용하여 원본 문자열을 확장합니다. (ANSI)
ExpandEnvironmentStringsForUserW 지정된 사용자에 대해 설정된 환경 블록을 사용하여 원본 문자열을 확장합니다. (유니코드)
추출 지정된 위치에서 아이콘 이미지를 추출합니다. (ANSI)
추출 지정된 위치에서 아이콘 이미지를 추출합니다. (유니코드)
추출 셸 폴더의 항목과 같은 개체에서 이미지를 요청합니다.
ExtractAssociatedIconA 파일에 리소스로 저장된 아이콘 또는 파일의 연결된 실행 파일에 저장된 아이콘에 대한 핸들을 가져옵니다. (ANSI)
ExtractAssociatedIconExA ExtractAssociatedIconEx를 변경하거나 사용할 수 없습니다. (ANSI)
ExtractAssociatedIconExW ExtractAssociatedIconEx를 변경하거나 사용할 수 없습니다. (유니코드)
ExtractAssociatedIconW 파일에 리소스로 저장된 아이콘 또는 파일의 연결된 실행 파일에 저장된 아이콘에 대한 핸들을 가져옵니다. (유니코드)
ExtractIconA 지정된 실행 파일, DLL 또는 아이콘 파일에서 아이콘에 대한 핸들을 가져옵니다. 크거나 작은 아이콘에 대한 핸들 배열을 검색하려면 ExtractIconEx 함수를 사용합니다. (ANSI)
ExtractIconExA ExtractIconEx 함수는 지정된 실행 파일, DLL 또는 아이콘 파일에서 추출된 크거나 작은 아이콘에 대한 핸들 배열을 만듭니다. (ANSI)

ExtractIconExW ExtractIconEx 함수는 지정된 실행 파일, DLL 또는 아이콘 파일에서 추출된 크거나 작은 아이콘에 대한 핸들 배열을 만듭니다. (유니코드)
ExtractIconW 지정된 실행 파일, DLL 또는 아이콘 파일에서 아이콘에 대한 핸들을 가져옵니다. 크거나 작은 아이콘에 대한 핸들 배열을 검색하려면 ExtractIconEx 함수를 사용합니다. (유니코드)
FileFailure 오류가 있고 사용자 상호 작용이 필요할 때 호출됩니다.
FillFromObject 결과 폴더를 만들고 항목으로 채웁니다.
필터 자격 증명 공급자 목록을 자격 증명 타일을 제공할 수 있는지 여부를 평가합니다.
FilterView 현재 보기의 내용에 대한 필터를 설정합니다.
FindCIE4ConnectionPoint Deprecated. 브라우저 이벤트에 대한 연결점을 가져옵니다.
FindConnectionPoint Deprecated. 브라우저 이벤트에 대한 연결점을 가져옵니다.
FindExecutableA 특정 문서 파일과 연결된 실행 파일(.exe) 파일의 이름과 핸들을 검색합니다. (ANSI)
FindExecutableW 특정 문서 파일과 연결된 실행 파일(.exe) 파일의 이름과 핸들을 검색합니다. (유니코드)
FindFolderFromIDList IDList를 기반으로 알려진 폴더를 나타내는 개체를 가져옵니다. 개체를 사용하면 특정 폴더 속성을 쿼리하고, 폴더의 현재 경로를 가져와서 폴더를 다른 위치로 리디렉션하고, 폴더의 경로를 ITEMIDLIST로 가져올 수 있습니다.

FindFolderFromPath 파일 시스템 경로를 기반으로 알려진 폴더를 나타내는 개체를 가져옵니다.
FindToolbar 도구 모음 프레임에서 지정된 IDockingWindow 개체를 찾아 해당 개체에 대한 인터페이스 포인터를 반환합니다.
FindTravelEntry Deprecated. 특정 여행 항목이 여행 로그에 있는지 여부를 확인합니다.
FindWindowSW 셸 창 컬렉션에서 창을 찾아 창의 핸들 및 IDispatch 인터페이스를 반환합니다.
FinishOperations IFileOperation 호출에 의해 수행된 마지막 작업이 완료된 후 호출자 구현 작업을 수행합니다.
FolderIdFromCsidl 레거시 CSIDL 값과 동일한 KNOWNFOLDERID를 가져옵니다.
FolderIdToCsidl 지정된 KNOWNFOLDERID와 동일한 레거시 CSIDL 값을 가져옵니다.
ForwardViewMsg Deprecated. SENDMessage hWnd 매개 변수로 BASEBROWSERDATA 구조체의 _hwndView 멤버를 사용하여 뷰에서 받은 메시지를 사용하여 SendMessage 함수를 호출합니다.
FoundItem 네임스페이스를 탐색하는 동안 네임스페이스에서 개체를 찾을 때 호출됩니다. 이 메서드를 구현하는 클래스의 기본 작업 함수로 사용합니다. 이 메서드 내에서 필요에 따라 작업을 수행합니다.
FreeConfirmConflictItem CONFIRM_CONFLICT_ITEM 구조에 할당된 리소스를 해제합니다.
FreeIDListArray PIDL(항목 식별자 목록) 목록 배열에 대한 포인터에서 사용하는 메모리를 해제합니다.
FreeIDListArrayChild

자식 항목 ID에 대한 포인터 배열의 메모리 공간을 해제합니다. 이렇게 하면 배열 내의 PITEMID_CHILDs 배열 자체와 둘 다 해제됩니다.
FreeIDListArrayFull PIDL(항목 식별자 목록) 배열에 대한 포인터의 메모리 공간을 해제합니다. 그러면 배열 내의 PIDLIST_ABSOLUTEs 및 배열 자체가 모두 해제됩니다.
FreeKnownFolderDefinitionFields IKnownFolder::GetFolderDefinition의 결과에서 할당된 필드를 해제합니다.
get_Application 애플리케이션 개체를 가져옵니다.
get_Count Shell 창 컬렉션의 창 수를 가져옵니다.
get_CurrentViewMode 현재 폴더의 현재 보기 모드를 가져옵니다.
get_FocusedItem 입력 포커스가 있는 항목을 나타내는 FolderItem 개체를 가져옵니다.
get_Folder 뷰를 나타내는 Folder 개체를 가져옵니다.
get_FolderFlags 현재 폴더의 설정을 가져옵니다.
get_GroupBy 폴더 뷰를 그룹화하기 위해 사용되는 열을 가져옵니다.
get_IconSize 현재 폴더의 아이콘 크기 설정을 가져옵니다.
get_Parent 구현되지 않았습니다. (IShellFolderViewDual.get_Parent)
get_Script

뷰에 대한 스크립팅 개체를 가져옵니다.
get_SortColumns 현재 폴더를 정렬하는 데 사용되는 열의 이름을 가져옵니다.
get_ViewOptions 뷰의 현재 옵션을 나타내는 플래그 집합을 가져옵니다.
GetAcceptLanguagesA 언어 기본 설정을 지정할 때 웹 사이트에 사용되는 문자열을 검색합니다. (ANSI)
GetAcceptLanguagesW 언어 기본 설정을 지정할 때 웹 사이트에 사용되는 문자열을 검색합니다. (유니코드)
GetAccountOptions 로컬 또는 Microsoft 계정의 "다른 사용자" 타일이 로그온 또는 자격 증명 UI에 표시되는지 여부를 나타내는 값을 검색합니다.
GetAllUsersProfileDirectoryA 모든 사용자가 공유하는 프로그램 데이터를 포함하는 디렉터리의 루트 경로를 검색합니다. (ANSI)
GetAllUsersProfileDirectoryW 모든 사용자가 공유하는 프로그램 데이터를 포함하는 디렉터리의 루트 경로를 검색합니다. (유니코드)
GetAnyOperationsAborted IFileOperation::P erformOperations를 호출하여 시작된 파일 작업이 완료되기 전에 중지되었는지 여부를 나타내는 값을 가져옵니다. 사용자 작업 또는 시스템에서 자동으로 작업을 중지할 수 있습니다.
GetAppContainerFolderPath 지정된 앱 컨테이너에 대한 로컬 앱 데이터 폴더의 경로를 가져옵니다.
GetAppContainerRegistryLocation 앱 컨테이너와 연결된 레지스트리 스토리지의 위치를 가져옵니다.
GetAppID 선언된 경우 파일 형식 처리기의 명시적 애플리케이션 사용자 모델 ID(AppUserModelID)를 검색합니다.
GetAppInfo

애플리케이션에 대한 일반 정보를 가져옵니다.
GetApplicationDisplayName 처리기를 구현한 애플리케이션의 표시 이름을 검색합니다.
GetApplicationIconReference 처리기를 구현한 애플리케이션의 아이콘을 검색합니다.
GetApplicationPublisher 처리기를 구현한 애플리케이션 게시자의 이름을 검색합니다.
GetApplicationSizeBounds 이 메서드는 애플리케이션에서 지원하는 크기 범위를 검색합니다.
GetApplicationViewOrientation 애플리케이션 디자인 모드 창의 방향을 가져옵니다.
GetAppName 파일을 사용하는 애플리케이션의 이름을 검색합니다.
GetAppUserModelId 원본 애플리케이션에서 AppUserModelId를 검색합니다.
GetAppVisibilityOnMonitor 지정된 모니터의 현재 모드를 쿼리합니다.
GetArguments Shell 링크 개체와 연결된 명령줄 인수를 가져옵니다. (ANSI)
GetArguments Shell 링크 개체와 연결된 명령줄 인수를 가져옵니다. (유니코드)
GetArrangeParam 뷰가 정렬된 방식인 뷰의 정렬 매개 변수를 가져옵니다.
GetAsyncMode 데이터 개체가 비동기 데이터 추출을 지원하는지 여부를 확인하기 위해 놓기 대상에 의해 호출됩니다.

GetAt 배열에서 지정된 사용자를 검색합니다.
GetAt 지정된 개체의 인터페이스에 대한 포인터를 제공합니다. 개체와 인터페이스는 인덱스 및 인터페이스 ID로 지정됩니다.
GetAttributes IShellItem 개체의 요청된 특성 집합을 가져옵니다.
GetAttributes IShellItemArray에 포함된 항목 집합의 특성을 가져옵니다.
GetAttributes 리소스 특성을 가져옵니다.
GetAttributesOf IShellFolder가 나타내는 개체에 포함된 하나 이상의 파일 또는 폴더 개체의 특성을 가져옵니다.
GetAutoArrange GetAutoArrange를 변경하거나 사용할 수 없습니다.
GetAutoArrange 폴더 자동 정렬 모드의 현재 상태를 가져옵니다.
GetAvailableSize 접근성 창을 표시하기 위해 특정 화면에서 사용할 수 있는 차원을 검색합니다.
GetBackgroundColor 이미지가 표시되지 않거나 바탕 화면 배경이 비활성화된 경우 바탕 화면에 표시되는 색을 검색합니다. 이 색은 바탕 화면 배경 화면이 전체 화면을 채우지 않을 때 테두리로도 사용됩니다.
GetBandInfo 밴드 개체의 상태 정보를 가져옵니다.
GetBandInfoSFB IShellFolderBand 개체와 관련된 정보를 가져오고 BANDINFOSFB 구조체에 배치합니다.

GetBandObject 밴드 사이트에서 지정된 밴드 개체를 가져옵니다.
GetBandSiteInfo 밴드 사이트의 밴드에 대한 정보를 가져옵니다.
GetBaseBrowserData Deprecated. 상태를 결정하기 위해 기본 클래스가 소유한 보호된 요소를 포함하는 읽기 전용 구조를 가져옵니다.
GetBitmapValue 비트맵 필드가 있는 자격 증명에서 비트맵 데이터를 검색할 수 있습니다.
GetBool 지정된 속성 키의 부울 값을 가져옵니다.
GetBorderDW 지정된 IDockingWindow 개체에 할당된 테두리 공간을 가져옵니다.
GetBrowserByIndex Deprecated. 지정된 인덱스를 사용하여 브라우저를 검색합니다.
GetBrowserIndex Deprecated. 창 계층 구조에서 브라우저의 인덱스를 검색합니다.
GetCachedSlowAppInfo 느린 원본에서 시작된 애플리케이션에 정보를 반환합니다. IShellApp::GetSlowAppInfo와 달리 이 메서드는 캐시된 정보를 반환할 수 있습니다. 이 메서드는 게시된 애플리케이션에 적용되지 않습니다.
GetCancelEvent 작업이 취소될 때 전송될 이벤트를 검색합니다.
GetCancelInformation 사용자가 진행률 UI에서 취소를 선택할 때 게시되는 정보를 반환합니다.
GetCancelledPage

사용자가 마법사 확장을 통해 탐색을 취소할 때 호출됩니다. 사용자가 마법사 확장에 있는 동안 탐색을 취소할 때 표시할 마법사 페이지를 나타내는 PROPSHEETPAGE의 핸들을 가져옵니다.
GetCanonicalName Windows 탐색기 명령의 GUID를 가져옵니다.
GetCapabilities 파일을 닫을 수 있는지 여부와 UI가 파일을 사용하는 애플리케이션의 창으로 전환할 수 있는지 여부를 결정합니다.
GetCapabilities 처리기의 정의된 기능을 설명하는 플래그 집합을 가져옵니다.
GetCapabilities 항목의 정의된 기능을 설명하는 플래그 집합을 가져옵니다.
GetCategories 애플리케이션 게시자가 제공하는 범주를 나열하는 구조를 검색합니다.
GetCategory 식별자 목록과 연결된 범주 목록을 가져옵니다.
GetCategory 선택한 폴더의 범주(가상, 고정, 공통 또는 사용자 단위)를 검색합니다.
GetCategoryForSCID 지정된 셸 열에 사용할 범주를 나타내는 GUID를 가져옵니다.
GetCategoryInfo UI에 표시할 기본 표시 및 텍스트와 같은 범주에 대한 정보를 가져옵니다.
GetCategoryName 지정된 범주의 이름을 가져옵니다.
GetCheckboxValue 확인란 값을 검색합니다.
GetCheckButtonState

대화 상자에서 확인 단추(확인란)의 현재 상태를 가져옵니다.
GetCheckState 트리 뷰 컨트롤에서 확인란 항목의 상태를 가져옵니다.
GetClient 클라이언트 개체를 가져옵니다.
GetCLSID 지정된 속성 키의 CLSID(클래스 식별자) 값을 가져옵니다.
GetCodePage ANSI 코드 페이지의 숫자 값(코드 페이지 식별자)을 검색합니다.
GetColor 지정한 대로 색을 가져옵니다.
GetColumnCount 표시되는 열 또는 전체 열 집합에 대한 열 수를 가져옵니다.
GetColumnInfo 열에 대한 정보를 요청합니다.
GetColumnInfo 각 열:_width, 표시 유형, 표시 이름 및 상태에 대한 정보를 가져옵니다.
GetColumnPropertyList 열거 열에 해당하는 순서가 지정된 열 목록을 가져옵니다.
GetColumns 뷰에서 지원하는 열을 나타내는 PROPERTYKEY 구조체의 배열을 가져옵니다. 모든 열 또는 현재 표시되는 열만 포함합니다.
GetComboBoxValueAt 지정된 인덱스에서 콤보 상자 항목의 문자열 레이블을 가져옵니다.
GetComboBoxValueCount 지정된 콤보 상자의 항목 수를 가져오고 초기 선택 항목을 지정합니다.

GetCommand 지정된 탐색기 명령 인스턴스를 가져옵니다.
GetCommands 지정된 Explorer 명령 열거자 인스턴스를 가져옵니다.
GetCommandString 도움말 문자열과 명령의 언어 독립적 이름 또는 정식 이름을 포함하여 바로 가기 메뉴 명령에 대한 정보를 가져옵니다.
GetComment 처리기에 대한 설명이 포함된 문자열을 가져옵니다.
GetComment 항목에 대한 설명이 포함된 문자열을 가져옵니다.
GetCompositionState 컴퍼지션 상태를 가져옵니다.
GetCondition 검색 상자의 내용을 ICondition 개체로 검색합니다.
GetConflictIdInfo 충돌 저장소 내에서 충돌을 식별하는 정보를 가져옵니다.
GetConflictIDList 충돌을 해당 IShellItem에 매핑합니다.
GetContext 처리기에서 속성을 표시하거나 상황에 맞는 메뉴 동작을 실행하는 데 사용할 수 있는 컨텍스트 개체를 가져옵니다.
GetControlItemState 대화 상자에 있는 컨테이너 컨트롤에 있는 항목의 현재 상태를 가져옵니다.
GetControlState 지정된 컨트롤의 현재 표시 유형 및 사용 상태를 가져옵니다.

GetControlStyle 네임스페이스 개체의 트리 뷰 컨트롤에 대해 설정된 표시 스타일을 가져옵니다.
GetControlStyle2 네임스페이스 개체의 트리 뷰 컨트롤에 대한 확장 표시 스타일 집합을 가져옵니다.
GetControlWindow 브라우저 컨트롤에 대한 창 핸들을 가져옵니다.
GetCount 지정된 IShellItem 배열의 항목 수를 가져옵니다.
GetCount 사용자 배열의 ICredentialProviderUser 개체 수를 검색합니다.
GetCount 충돌 항목 수를 가져옵니다.
GetCount 항목 수를 가져옵니다.
GetCount 저장소의 충돌 횟수를 가져옵니다.
GetCount 컬렉션의 개체 수를 제공합니다.
GetCreateFlags CreateProcess에 전달될 추가 플래그를 가져옵니다.
GetCredentialAt 특정 자격 증명을 가져옵니다.
GetCredentialCount 이 자격 증명 공급자에서 사용 가능한 자격 증명 수를 가져옵니다.
GetCurFolder

폴더 개체의 ITEMIDLIST를 가져옵니다.
GetCurrentFilter 현재 필터를 유니코드 문자열로 가져옵니다.
GetCurrentFolderFlags 현재 적용된 폴더 플래그를 가져옵니다.
GetCurrentInfo 현재 폴더 설정을 가져옵니다.
GetCurrentPage 다중 페이지 이미지의 현재 페이지를 가져옵니다.
GetCurrentProcessExplicitAppUserModelID 현재 프로세스에 대한 애플리케이션 정의 명시적 애플리케이션 사용자 모델 ID(AppUserModelID)를 검색합니다.
GetCurrentSelection 대화 상자에서 사용자의 현재 선택 영역을 가져옵니다.
GetCurrentView 브라우저의 현재 보기에 대한 인터페이스를 가져옵니다.
GetCurrentView 가장 최근 제어판 보기:_Classic 보기 또는 범주 보기를 가져옵니다.
GetCurrentViewMode 폴더의 현재 보기 모드를 나타내는 값을 포함하는 주소를 가져옵니다.
GetData 레지스트리에서 파일 또는 프로토콜 연결 관련 이진 데이터를 검색하고 검색합니다.
GetDataFormatFromPath 확장에 따라 파일의 형식을 결정합니다.
GetDataFromIDList

IDList의 첫 번째 ItemID에서 클라이언트 제공 구조체에 대한 읽기 전용 포인터를 가져옵니다. (오버로드 1/2)
GetDataFromIDList IDList의 첫 번째 ItemID에서 클라이언트 제공 구조체에 대한 읽기 전용 포인터를 가져옵니다. (오버로드 2/2)
GetDataObject 현재 DataPackage 개체의 IDataObject 표현을 가져옵니다.
GetDateStamp 이미지가 마지막으로 수정된 날짜를 요청합니다. 이 메서드를 사용하면 셸에서 캐시된 이미지가 오래된지 여부를 확인할 수 있습니다.
GetDefaultBandWidth 밴드를 추가할 때 밴드가 처음에 기본 너비를 설정하는 데 사용하는 대역 너비를 가져옵니다.
GetDefaultCategory 폴더가 기본 그룹화 재정의를 사용하도록 설정합니다.
GetDefaultColumn 기본 정렬 및 표시 열을 가져옵니다.
GetDefaultColumnState 지정된 열의 기본 상태를 가져옵니다.
GetDefaultDestinationName 셸 항목의 기본 이름을 가져옵니다.
GetDefaultMenuText 기본 바로 가기 메뉴 텍스트를 가져오기 위해 셸 보기에서 호출됩니다.
GetDefaultSaveFolder 라이브러리에서 저장 작업에 사용하는 기본 대상 폴더를 검색합니다.
GetDefaultSearchGUID 폴더에 대한 기본 검색 개체의 GUID(Globally Unique Identifier)를 반환합니다.

GetDefaultSpacing 주변 공백을 포함하여 항목의 기본 너비(x) 및 높이(y) 측정값을 포함하는 POINT 구조체에 대한 포인터를 가져옵니다.
GetDefaultUserProfileDirectoryA 기본 사용자 프로필의 루트 경로를 검색합니다. (ANSI)
GetDefaultUserProfileDirectoryW 기본 사용자 프로필의 루트 경로를 검색합니다. (유니코드)
GetDelay 애니메이션의 현재 프레임에 대한 지연 값을 가져옵니다.
GetDelayedPropertyStore 지정된 대로 IPropertyStore 인터페이스 개체를 가져옵니다.
GetDescription UI에 표시할 수 있는 디바이스 유형별 그룹화와 같은 범주의 이름을 가져옵니다.
GetDescription Shell 링크 개체에 대한 설명 문자열을 가져옵니다. (ANSI)
GetDescription Shell 링크 개체에 대한 설명 문자열을 가져옵니다. (유니코드)
GetDescription 이벤트 설명을 가져옵니다.
GetDestinationStream 복사할 파일의 새 버전을 수신하는 빈 스트림을 가져옵니다.
GetDetailsEx Shell 폴더의 항목에서 FMTID(속성 집합 식별자) 및 PID(속성 식별자)로 식별되는 자세한 정보를 가져옵니다.
GetDetailsOf Shell 폴더의 항목에 대한 자세한 정보를 가져옵니다.

GetDetailsOf Shell 폴더의 항목에 대한 열 인덱스로 식별되는 자세한 정보를 가져옵니다.
GetDirectory 현재 작업 디렉터리를 가져옵니다.
GetDisplayName IShellItem 개체의 표시 이름을 가져옵니다.
GetDisplayNameOf 지정된 파일 개체 또는 하위 폴더의 표시 이름을 검색합니다.
GetDpiForShellUIComponent 현재 배율 인수 및 PROCESS_DPI_AWARENESS 기반으로 SHELL_UI_COMPONENT 차지하는 dpi(인치당 점수)를 검색합니다.
GetDragPoint 현재 끌어서 놓기 작업이 시작된 지점을 가져옵니다.
GetDropDownStatus 자동 완성 드롭다운 목록의 현재 표시 상태를 가져옵니다.
GetDropPoint 현재 끌어서 놓기 작업이 종료된 지점을 가져옵니다.
GetDynamicInfo 자동 실행 대화 상자가 표시되기 전에 특정 처리기가 표시될지 여부를 확인하기 위해 시스템에서 호출됩니다.
GetEditBoxText 편집 상자 컨트롤의 현재 텍스트를 가져옵니다.
GetEncoderParams 인코더 매개 변수의 현재 집합을 가져옵니다.
GetEntry 캐시의 이미지에 대한 핸들을 가져옵니다.

GetEnum 이 메서드는 구현되지 않습니다. (IQueryAssociations.GetEnum)
GetEnumFlags 클라이언트에서 셸 폴더에 열거할 개체의 클래스를 지정할 수 있습니다. SHBrowseForFolder와 함께 사용하는 경우 대화 상자 트리 뷰에 표시되어야 하는 항목의 클래스 또는 클래스와 그렇지 않은 클래스를 지정합니다.
GetEnumFlagsForItem 클라이언트가 뷰에 포함되도록 셸 항목에 열거해야 하는 개체 클래스를 지정할 수 있습니다.
GetEvent 지정된 이벤트 개체를 가져옵니다.
GetEventCount 이벤트 수를 가져옵니다.
GetEventEnumerator 처리기 이벤트에 대한 열거자를 가져옵니다.
GetEventID 이벤트 ID를 가져옵니다.
GetExtension 바로 가기 개체의 파일 이름 확장명을 가져옵니다. (ANSI)
GetExtension 바로 가기 개체의 파일 이름 확장명을 가져옵니다. (유니코드)
GetFieldDescriptorAt 지정된 필드를 설명하는 메타데이터를 가져옵니다.
GetFieldDescriptorCount 이 공급자의 자격 증명을 표시하는 데 필요한 필드 수를 검색합니다.
GetFieldOptions

로그온 또는 자격 증명 UI에서 지정된 필드에 대한 현재 옵션 집합을 검색합니다. 자격 증명 공급자 프레임워크에서 호출됩니다.
GetFieldState 필드 상태를 검색합니다. 로그온 UI 및 자격 증명 UI는 이 정보를 사용하여 자격 증명 필드에 대한 정보를 얻고 이 정보를 사용자 타일에 표시합니다.
GetFileID 현재 파일의 고유 파일 식별자를 가져옵니다.
GetFileName 대화 상자의 파일 이름 편집 상자에 현재 입력한 텍스트를 검색합니다.
GetFileNameFromBrowse GetFileNameFromBrowse 함수는 사용자가 열 파일의 드라이브, 디렉터리 및 이름을 지정할 수 있도록 열기 대화 상자를 만듭니다.
GetFileNameFromBrowse GetFileNameFromBrowse 함수는 사용자가 열 파일의 드라이브, 디렉터리 및 이름을 지정할 수 있도록 열기 대화 상자를 만듭니다. (GetFileNameFromBrowse)
GetFileOverlayInfo 지정된 특성이 있는 지정된 파일에 대한 아이콘 오버레이 또는 아이콘 이미지의 인덱스 또는 아이콘 이미지를 가져옵니다.
GetFileTime 지정된 속성 키의 날짜 및 시간 값을 가져옵니다.
GetFileTypeIndex 현재 선택한 파일 형식을 가져옵니다.
GetFindData WIN32_FIND_DATA 구조에 저장된 파일 시스템 정보를 가져옵니다.
GetFirstPage 마법사 확장의 첫 번째 페이지에 대한 핸들을 가져옵니다.
GetFlags

Windows 탐색기 명령과 연결된 플래그를 가져옵니다.
GetFlags 현재 옵션 설정을 가져옵니다.
GetFlags 이벤트 플래그를 가져옵니다.
GetFlags Deprecated. 현재 브라우저 플래그 집합을 검색합니다.
GetFocusedItem 현재 폴더 뷰에 포커스가 있는 항목의 인덱스입니다.
GetFolder 바로 가기 개체의 폴더 이름을 가져옵니다. (ANSI)
GetFolder 바로 가기 개체의 폴더 이름을 가져옵니다. (유니코드)
GetFolder 대화 상자에서 현재 선택된 폴더를 가져오거나 대화 상자가 현재 표시되지 않으면 대화 상자를 열 때 선택할 폴더를 가져옵니다.
GetFolder 폴더 개체를 가져옵니다.
GetFolder KNOWNFOLDERID로 식별된 알려진 폴더를 나타내는 개체를 가져옵니다.
GetFolderByName 정식 이름으로 식별되는 알려진 폴더를 나타내는 개체를 가져옵니다.
GetFolderCapabilities System.IsPinnedToNameSpaceTree 속성 키 값을 통해 필터링할 폴더의 기능을 가져오고 알림 등록 상태를 변경합니다.
GetFolderDefinition

폴더의 범주, 이름, 경로, 설명, 도구 설명, 아이콘 및 기타 속성을 포함하는 알려진 폴더의 정의 요소를 포함하는 구조를 검색합니다.
GetFolderFlags 폴더 뷰 옵션 플래그를 가져옵니다.
GetFolderIds 등록된 모든 알려진 폴더 ID의 배열을 가져옵니다. 알려진 모든 폴더를 열거하는 데 사용할 수 있습니다.
GetFolders 라이브러리에 포함된 자식 폴더 집합을 가져옵니다.
GetFolderSetData Deprecated. 폴더 정보를 포함하는 구조를 가져옵니다.
GetFolderTargetInfo 폴더 바로 가기의 대상 폴더의 위치 및 특성을 제공합니다.
GetFolderType 폴더 유형을 검색합니다.
GetFolderType 라이브러리의 폴더 유형을 가져옵니다.
GetFolderViewOptions 보기에 대한 현재 옵션 집합을 검색합니다.
GetFont 글꼴에 설정된 현재 특성을 가져옵니다.
GetFormat 비트맵 이미지의 알파 형식을 검색합니다.
GetForWindow 지정된 창에 대한 DataTransferManager 인스턴스를 가져옵니다.
GetFrameOptions 사용 가능한 브라우저 프레임 보기 옵션을 검색합니다.

GetGroupBy 폴더 표시의 항목을 그룹화하는 데 사용되는 속성 및 정렬 순서를 검색합니다.
GetGroupByProperty 그룹화 속성을 가져옵니다.
GetGroupSubsetCount 그룹의 하위 집합에 대해 표시되는 행 수를 가져옵니다.
GetGroupSubsetCount 표시되는 행의 그룹 수를 가져옵니다.
GetHandlerEnumerator 사용자가 노출하고 관리하는 동기화 처리기의 ID에 대한 액세스를 제공하는 열거자를 가져옵니다.
GetHandlerID 이벤트가 기록된 처리기의 ID를 가져옵니다.
GetHandlerInfo 처리기를 설명하는 속성을 가져옵니다.
GetHandlerInfo 처리기 정보를 가져옵니다.
GetHandlerRegistrationInfo 현재 등록 정보를 가져오기 위해 등록된 애플리케이션의 처리기에서 호출됩니다.
GetHistoryObject Deprecated. 브라우저의 기록 개체를 나타내는 IOleObject를 검색합니다.
GetHotkey Shell 링크 개체의 바로 가기 키(핫 키)를 가져옵니다. (ANSI)
GetHotkey Shell 링크 개체의 바로 가기 키(핫 키)를 가져옵니다. (유니코드)
GetIcon

지정된 Windows 탐색기 명령 항목과 연결된 아이콘의 아이콘 리소스 문자열을 가져옵니다.
GetIcon 라이브러리의 기본 아이콘을 가져옵니다.
GetIconLocation 아이콘의 위치 및 인덱스입니다. (ANSI)
GetIconLocation 아이콘의 위치 및 인덱스입니다. (유니코드)
GetIconLocation 애플리케이션과 연결된 아이콘의 위치를 검색합니다.
GetIconLocation Shell 링크 개체의 아이콘 위치(경로 및 인덱스)를 가져옵니다. (ANSI)
GetIconLocation Shell 링크 개체의 아이콘 위치(경로 및 인덱스)를 가져옵니다. (유니코드)
GetIconOf 특정 폴더 내의 개체에 대한 아이콘을 가져옵니다.
GetIconSize 폴더 아이콘 크기를 가져옵니다.
GetId 선택한 폴더의 ID를 가져옵니다.
GetIDArrayResult INamespaceWalk::Walk에서 시작한 네임스페이스 워크 중에 찾은 개체 목록을 가져옵니다.
GetIDList IDList(ITEMIDLIST) 양식에서 셸 네임스페이스 폴더의 위치를 가져옵니다.
GetIDList 항목 식별자 목록을 가져옵니다.

GetIDList 검색 폴더를 ITEMIDLIST로 가져옵니다.
GetIDList ShellI 링크 개체의 대상에 대한 항목 식별자 목록을 가져옵니다. (ANSI)
GetIDList ShellI 링크 개체의 대상에 대한 항목 식별자 목록을 가져옵니다. (유니코드)
GetImage IShellItem을 나타내는 HBITMAP를 가져옵니다.
GetInfoFlags 항목의 정보 플래그를 가져옵니다. 이 메서드는 현재 사용되지 않습니다.
GetInfoTip 항목의 정보 팁 텍스트를 가져옵니다.
GetInt32 지정된 속성 키의 Int32 값을 가져옵니다.
GetItem 지정된 항목을 나타내는 개체를 검색합니다.
GetItem IParseAndCreateItem::GetItem 메서드
GetItem 이 항목과 관련된 IShellItem을 가져옵니다.
GetItem 지정된 충돌 데이터 항목을 가져옵니다.
GetItem 성공하면 지정된 항목에 대한 결과 정보를 가져옵니다.
GetItemAt

IShellItemArray의 지정된 인덱스에서 항목을 가져옵니다.
GetItemChoice 사용자가 유지하려는 항목의 인덱스를 가져옵니다.
GetItemChoiceCount 사용자가 유지할 항목 수를 가져옵니다.
GetItemCustomState 지정된 셸 항목과 연결된 확인란의 상태를 가져옵니다.
GetItemData 지정된 파일에 대한 열 데이터를 요청합니다.
GetItemHeight 항목 높이를 가져옵니다.
GetItemID 이벤트가 기록된 항목의 ID를 가져옵니다.
GetItemID 동기화 항목의 고유 ID를 가져옵니다.
GetItemIDList 관련된 항목의 PIDL(항목 식별자 목록)에 대한 포인터를 가져옵니다.
GetItemInfo 동기화 항목의 속성을 가져옵니다.
GetItemObject 뷰에 표시되는 데이터를 참조하는 인터페이스를 가져옵니다.
GetItemObject 등록된 애플리케이션이 처리하는 지정된 항목에 대한 인터페이스를 가져옵니다.
GetItemPosition 폴더 보기에 있는 항목의 위치를 가져옵니다.

GetItemRect 지정된 항목의 크기와 위치를 설명하는 RECT 구조를 가져옵니다.
GetItemsArray 충돌 항목 배열을 검색합니다.
GetItemSpacing 작은 보기 모드와 큰 보기 모드의 간격만 가져옵니다.
GetItemState 셸 항목에 대한 상태 정보를 가져옵니다.
GetIterationInfo 충돌 집합에서 해결되는 충돌에 대한 정보를 가져옵니다.
GetJunctionCLSID 항목이 접합 지점인 경우 항목에 대해 IShellFolder를 구현하는 개체의 CLSID(클래스 식별자)를 가져옵니다.
GetKey 레지스트리에서 파일 또는 프로토콜 연결 관련 키를 검색하고 검색합니다.
GetLastPage 마법사 확장 페이지의 마지막 페이지에 대한 핸들을 가져옵니다.
GetLastSyncTime 처리기가 마지막으로 동기화된 날짜 및 시간을 가져옵니다.
GetLastSyncTime 항목이 마지막으로 동기화된 날짜 및 시간을 가져옵니다.
GetLevel 이벤트의 로그 수준을 가져옵니다.
GetLinkReference 이벤트에 대한 핫 링크에 대한 참조를 가져옵니다. 핫 링크는 사용자가 작업을 실행하기 위해 클릭할 수 있는 표시된 속성입니다. 이를 통해 처리기는 사용자가 폴더에서 한눈에 볼 수 있는 사용 가능한 작업을 표시할 수 있습니다.

GetLinkText 이벤트에 대한 핫 링크의 텍스트를 가져옵니다.
GetList 점프 목록의 최근 항목 또는 자주 사용하는 범주에 나열된 대상의 컬렉션을 나타내는 개체를 검색합니다.
GetLocation 추출할 이미지의 경로를 가져옵니다.
GetMaxLength 호출되는 네임스페이스의 특정 이름에 허용되는 최대 문자 수를 반환합니다.
GetMenu IShellMenu::SetMenu를 호출하여 설정된 메뉴 정보를 가져옵니다.
GetMenuContextHelpId 지정된 메뉴와 연결된 도움말 컨텍스트 식별자를 검색합니다.
GetMenuInfo IShellMenu::Initialize 메서드에서 정보를 가져옵니다.
GetMenuPosFromID GetMenuPosFromID를 변경하거나 사용할 수 없습니다.
GetMenuRestrictions . (IDefaultFolderMenuInitialize.GetMenuRestrictions)
GetMilliseconds 진행률 대화 상자의 경과된 시간 및 남은 시간을 가져옵니다.
GetMode 구문 분석된 항목의 열거 모드를 검색합니다.
GetMonitor 시작 중인 애플리케이션의 대상 모니터를 검색합니다.
GetMonitorDevicePathAt

시스템 모니터 중 하나의 고유 ID를 검색합니다.
GetMonitorDevicePathCount 시스템과 연결된 모니터 수를 검색합니다.
GetMonitorRECT 지정된 모니터의 표시 사각형을 검색합니다.
GetName 확장명 없이 바로 가기 개체의 파일 이름을 가져옵니다. (ANSI)
GetName 확장명 없이 바로 가기 개체의 파일 이름을 가져옵니다. (유니코드)
GetName 파일 형식과 연결된 실행 파일의 전체 경로 및 파일 이름을 검색합니다.
GetName 이벤트의 이름을 가져옵니다. 이 문자열은 이벤트의 간단한 이름 또는 간단한 요약일 수 있습니다. 폴더 및 이벤트의 속성 시트에 표시됩니다.
GetName 처리기의 표시 이름을 가져옵니다.
GetName 동기화 항목의 UI 표시 이름을 가져옵니다.
GetNameAt 속성 저장소의 지정된 인덱스에서 속성의 이름을 가져옵니다.
GetNameCount 속성 저장소의 속성 이름 수를 가져옵니다.
GetNamedValue 명명된 속성 저장소에서 명명된 속성의 값을 가져옵니다.
GetNavigateState Deprecated. 브라우저의 현재 탐색 상태를 검색합니다.

GetNextItem 요청된 메서드에 따라 트리의 다음 항목을 검색합니다.
GetNextPage 사용자가 마법사 확장 페이지를 지나 앞으로 이동할 때 호출됩니다. 마법사 확장 페이지 바로 다음 마법사 페이지를 나타내는 PROPSHEETPAGE의 핸들을 가져옵니다.
GetNumberOfApps 더 이상 사용되지 않음. 프로그램 제어판 애플리케이션 추가/제거 클라이언트는 E_NOTIMPL 반환할 수 있습니다.
GetNumberOfCategories 더 이상 사용되지 않음. 프로그램 추가/제거 제어판 애플리케이션의 클라이언트는 E_NOTIMPL 반환할 수 있습니다.
GetObject 보기에서 항목을 가져옵니다.
GetObject 처리기와 관련된 특정 형식의 개체를 만듭니다.
GetObject 항목과 관련된 특정 형식의 개체를 만듭니다.
GetObjectCount 폴더 보기의 항목 수를 가져옵니다.
GetOleObject Deprecated. 브라우저에 대한 IOleObject를 검색합니다.
GetOperationStatus 진행률 대화 상자의 작업 상태를 가져옵니다.
GetOptions 현재 자동 완성 옵션을 가져옵니다. (IACList2.GetOptions)
GetOptions

현재 자동 완성 옵션을 가져옵니다. (IAutoComplete2.GetOptions)
GetOptions 현재 브라우저 옵션을 가져옵니다.
GetOptions 대화 동작을 제어하도록 설정된 현재 플래그를 가져옵니다.
GetOptions 라이브러리의 옵션을 가져옵니다.
GetOverlayIconIndex 시스템 이미지 목록에서 아이콘 오버레이의 인덱스입니다.
GetOverlayIndex 시스템 이미지 목록의 오버레이 인덱스입니다.
GetOverlayInfo 아이콘 오버레이 비트맵의 위치를 제공합니다.
GetPackageExecutionState 지정된 패키지의 현재 실행 상태를 반환합니다.
GetPageCount 다중 페이지 이미지의 페이지 수를 가져옵니다.
GetPalette Deprecated. 브라우저의 색상표를 검색합니다.
GetPaneState 지정된 Windows 탐색기 창의 표시 상태를 가져옵니다.
GetParent IShellItem 개체의 상위 개체를 가져옵니다.
GetParentAndItem 항목의 부모 및 부모의 자식 ID를 가져옵니다.

GetParentSite Deprecated. 브라우저 부모의 현재 위치 클라이언트 사이트를 검색합니다.
GetPath 알려진 폴더의 경로를 문자열로 검색합니다.
GetPath 지정된 제어판 항목의 경로를 가져옵니다.
GetPath Shell 링크 개체 대상의 경로 및 파일 이름을 가져옵니다. (ANSI)
GetPath Shell 링크 개체 대상의 경로 및 파일 이름을 가져옵니다. (유니코드)
GetPidl Deprecated. PIDL(항목 식별자 목록)에 대한 현재 포인터의 복사본을 검색합니다.
GetPidl Deprecated. 이동 항목과 연결된 PIDL(항목 식별자 목록)에 대한 포인터를 가져옵니다.
GetPinnedWindow 현재 고정된 창(있는 경우)에 대한 핸들을 가져옵니다.
GetPixelFormat 이미지의 픽셀 형식을 가져옵니다.
GetPolicies 처리기에서 설정한 정책을 설명하는 플래그 집합을 가져옵니다.
GetPolicies 항목에서 설정한 정책을 설명하는 플래그 집합을 가져옵니다.
GetPosition 바탕 화면 배경 이미지의 현재 표시 값을 검색합니다.
GetPossibleActions

애플리케이션에 허용되는 관리 작업의 비트 마스크를 가져옵니다.
GetPresenterChoice 선택한 종류와 집합의 모든 후속 충돌에 선택을 적용할지 여부를 가져옵니다.
GetPresenterNextStep 동기화 관리자 충돌 해결의 다음 단계로 발표자가 수행할 작업을 가져옵니다.
GetPreviousPage 사용자가 마법사 확장에서 뒤로 이동할 때 호출됩니다. 마법사 확장 페이지 앞에 있는 마법사 페이지를 나타내는 PROPSHEETPAGE의 핸들을 가져옵니다.
GetPriority 아이콘 오버레이의 우선 순위를 지정합니다.
GetProcessReference SetProcessReference에서 제공하는 프로세스별 개체를 검색하여 참조 수를 증가시켜 프로세스를 활성 상태로 유지합니다.
GetProfilesDirectoryA 사용자 프로필이 저장되는 루트 디렉터리의 경로를 검색합니다. (ANSI)
GetProfilesDirectoryW 사용자 프로필이 저장되는 루트 디렉터리의 경로를 검색합니다. (유니코드)
GetProfileType 현재 사용자에 대해 로드된 프로필의 유형을 검색합니다.
GetProgID 개체와 연결된 ProgID를 검색합니다.
GetProperties 저장된 항목 또는 저장 프로세스의 항목에 대한 속성 값 집합을 검색합니다.
GetProperties 이미지의 속성에 액세스할 수 있는 IPropertySetStorage를 가져옵니다.
GetProperty

지정된 속성 키에서 PROPVARIANT 구조를 가져옵니다.
GetProperty 속성 키가 지정된 경우 충돌 속성을 가져옵니다.
GetPropertyDescriptionList 속성 키에 대한 참조가 지정된 속성 설명 목록 개체를 가져옵니다.
GetPropertyDescriptionList 셸 항목 배열의 항목에 대한 속성 설명 목록을 가져옵니다.
GetPropertyFromIDList 키를 사용하여 IDList 내의 IPropertyStore에서 변형으로 속성을 가져옵니다. (오버로드 3/4)
GetPropertyFromIDList 키를 사용하여 IDList 내의 IPropertyStore에서 변형으로 속성을 가져옵니다. (오버로드 2/4)
GetPropertyFromIDList 키를 사용하여 IDList 내의 IPropertyStore에서 변형으로 속성을 가져옵니다. (오버로드 4/4)
GetPropertyFromIDList 키를 사용하여 IDList 내의 IPropertyStore에서 변형으로 속성을 가져옵니다. (오버로드 1/4)
GetPropertyHandlerFromFileId 제공된 파일 식별자와 연결된 IStorageProviderPropertyHandler의 인스턴스를 가져옵니다.
GetPropertyHandlerFromPath 제공된 경로와 연결된 IStorageProviderPropertyHandler의 인스턴스를 가져옵니다.
GetPropertyHandlerFromUri 제공된 URI와 연결된 IStorageProviderPropertyHandler의 인스턴스를 가져옵니다.
GetPropertyKey 속성 키를 가져옵니다.
GetPropertyStorage 속성 저장소 인스턴스에서 직렬화된 속성 스토리지 데이터를 가져옵니다.

GetPropertyStorage 메타데이터를 저장하는 데 사용되는 직렬화된 속성 스토리지에 대한 읽기 전용 포인터를 가져옵니다.
GetPropertyStorageBuffer 속성 저장소 인스턴스에서 serialize된 속성 스토리지 버퍼를 가져옵니다.
GetPropertyStorageFromIDList 첫 번째 ItemID와 연결된 직렬화된 속성 스토리지를 기반으로 IPropertyStore의 인스턴스를 만듭니다.
GetPropertyStorageSize 속성 저장소 인스턴스에서 직렬화된 속성 스토리지 데이터의 크기를 가져옵니다.
GetPropertyStore 지정된 속성 저장소 플래그에 대한 속성 저장소 개체를 가져옵니다.
GetPropertyStore 속성 저장소를 가져옵니다.
GetPropertyStoreForKeys 지정된 속성 키에 대한 속성 저장소 개체를 가져옵니다.
GetPropertyStoreWithCreateObject CoCreateInstance 대신 지정된 ICreateObject를 사용하여 이 메서드가 호출되는 셸 항목과 연결된 속성 처리기의 인스턴스를 만듭니다.
GetProviderID 이 사용자에 대한 계정 공급자의 ID를 검색합니다.
GetPublishedAppInfo 애플리케이션 게시자가 게시한 애플리케이션에 대한 게시 관련 정보를 가져옵니다.
GetRawDataFormat 이미지의 형식을 식별하는 GUID를 검색합니다.
GetRecorderDriveLetter 쓰기 사용으로 표시된 CD 드라이브의 드라이브 문자를 가져옵니다.

GetRedirectionCapabilities 알려진 폴더의 경로를 새 값으로 설정할 수 있는지 또는 해당 리디렉션에 적용되는 특정 제한 사항 또는 금지 사항을 나타내는 값을 가져옵니다.
GetReferent 바로 가기 개체의 참조를 가져옵니다. (ANSI)
GetReferent 바로 가기 개체의 참조를 가져옵니다. (유니코드)
GetRemovedDestinations 이 사용자 지정 점프 목록을 바꾸려는 기존 점프 목록에서 사용자가 제거한 대상의 현재 목록을 검색합니다.
GetReservedOverlayInfo 예약된 오버레이 중 하나에서 지정된 특성이 있는 지정된 파일의 아이콘 오버레이 또는 아이콘 이미지의 인덱스를 가져옵니다.
GetResolution 이미지의 해상도(dpi(인치당 점)를 가져옵니다.
GetResolutionHandler 충돌에 대한 해결 처리기를 가져옵니다.
GetResourceDescription 리소스 설명을 가져옵니다.
GetResult 대화 상자에서 사용자가 선택한 항목을 가져옵니다.
GetResults 여러 항목을 선택할 수 있는 대화 상자에서 사용자의 선택을 가져옵니다.
GetResults RSS 또는 Atom 형식으로 서식이 지정된 OpenSearch 데이터 원본에서 검색 결과를 반환합니다.
GetRootItems

루트 항목의 배열을 가져옵니다.
GetScaleFactorForDevice 디스플레이 디바이스의 기본 배율 인수를 가져옵니다.
GetScaleFactorForMonitor 특정 모니터의 배율 인수를 가져옵니다. 이 함수는 GetScaleFactorForDevice를 대체합니다.
GetSearchStyle 자동 아키텍처가 수행되는 방법을 결정하는 레지스트리 설정을 재정의합니다.
GetSearchText 브라우저의 주소 표시줄에 있는 텍스트를 검색합니다.
GetSearchUrl 검색 중인 URL을 검색합니다.
GetSectorSize 현재 사용 중인 디스크의 섹터당 바이트 수를 검색합니다. 버퍼되지 않은 입력/출력(I/O)을 사용하는 경우 적절한 바이트 맞춤을 보장하기 위해 디스크에서 읽는 섹터의 크기를 알고 있는 것이 중요합니다.
GetSelectedControlItem 대화 상자의 지정된 컨테이너 컨트롤에서 특정 항목을 가져옵니다.
GetSelectedCount 선택한 보기의 항목 수를 가져옵니다.
GetSelectedItem 지정된 인덱스 또는 그 뒤에 현재 선택된 항목을 찾습니다.
GetSelectedItems 대화 상자에서 현재 선택한 항목을 가져옵니다. 이러한 항목은 보기에서 선택한 항목이거나 파일 이름 편집 상자에서 선택한 텍스트일 수 있습니다.
GetSelectedItems 선택한 셸 항목의 배열을 가져옵니다.
GetSelectedObjects

선택한 뷰에 있는 개체의 배열과 해당 개체의 수를 가져옵니다.
GetSelection 현재 선택 영역을 IShellItemArray로 가져옵니다.
GetSelection 선택한 항목이 포함된 셸 항목 배열을 가져옵니다.
GetSelectionMarkedItem IFolderView::SelectItem의 SVSI_SELECTIONMARK 사용하여 표시한 폴더 뷰의 항목 인덱스입니다.
GetSelectionState 확인 상태를 포함한 선택 상태를 가져옵니다.
GetSerialization 이 자격 증명을 기본 인증 엔진에 제출하려는 시도에 대한 응답으로 호출됩니다.
GetSetCodePage Deprecated. 새 문자 코드 페이지를 설정하고 이전 코드 페이지에 대한 포인터를 검색합니다.
GetSharedBitmap ISharedBitmap 개체에 포함된 비트맵을 검색합니다.
GetSharePermissions ACE(모든 사용자 액세스 제어 항목)의 사용자 또는 공용 폴더와 현재 연결된 액세스 권한을 가져옵니다.
GetShellFolder 메뉴 밴드가 찾아보도록 설정된 폴더를 가져옵니다.
GetShellItem 셸 항목(IShellItem 또는 파생 인터페이스) 형식으로 셸 네임스페이스에서 알려진 폴더의 위치를 검색합니다.
GetShellItem 검색 폴더를 IShellItem으로 가져옵니다.
GetShowCmd Shell 링크 개체에 대한 show 명령을 가져옵니다. (ANSI)

GetShowCmd ShellI 링크 개체에 대한 show 명령을 가져옵니다. (유니코드)
GetSid 사용자의 SID(보안 식별자)를 검색합니다. (ICredentialProviderUser.GetSid)
GetSize 원본 크기를 가져옵니다.
GetSize ISharedBitmap 개체에 포함된 비트맵의 크기를 검색합니다.
GetSize 이미지 파일의 크기를 가져옵니다.
GetSlideshow 슬라이드 쇼 이미지가 저장되는 디렉터리의 경로를 가져옵니다.
GetSlideshowOptions 순서 섞기 및 타이밍에 대한 현재 바탕 화면 배경 화면 슬라이드 쇼 설정을 가져옵니다.
GetSlowAppInfo 느린 원본에서 시작된 애플리케이션에 정보를 반환합니다. 이 메서드는 게시된 애플리케이션에 적용되지 않습니다.
GetSortColumnCount 뷰에 현재 적용된 정렬 열 수를 가져옵니다.
GetSortColumns 뷰에 현재 적용된 정렬 열을 가져옵니다.
GetSortColumns 열 정렬 정보를 가져옵니다.
GetSourceViewSizePreference 애플리케이션이 시작된 후 애플리케이션의 보기 크기 기본 설정을 검색합니다.

GetSourceViewToPosition 원본 애플리케이션 창의 위치를 검색합니다.
GetSpacing 항목의 너비(x) 및 높이(y) 차원(주변 공백 포함)을 포함하는 POINT 구조를 가져옵니다.
GetState 지정된 Windows 탐색기 명령 항목과 연결된 상태 정보를 가져옵니다.
GetState 지정된 셸 항목과 연결된 명령 상태를 가져옵니다.
GetState 채워진 SMDATA 구조를 가져옵니다.
GetStatus 슬라이드 쇼의 현재 상태를 가져옵니다.
GetString 지정된 속성 키의 문자열 값을 가져옵니다.
GetString 레지스트리에서 파일 또는 프로토콜 연결 관련 문자열을 검색하고 검색합니다. (IQueryAssociations.GetString)
GetStringValue 텍스트 필드가 있는 자격 증명에서 텍스트를 검색할 수 있습니다.
GetStringValue 입력 값을 기반으로 ICredentialProviderUser 개체에서 문자열 속성을 검색합니다.
GetSubmitButtonValue 제출 단추가 로그온 UI 옆에 배치되어야 하는 필드의 식별자를 검색합니다.
GetSupportedActionTypes CD 쓰기 확장 프로그램에 대해 지원되는 데이터 형식을 결정합니다.

GetSwitchToHWND 파일을 사용하는 애플리케이션의 최상위 창 핸들을 검색합니다.
GetSyncItem 지정된 동기화 항목을 가져옵니다.
GetSyncItemCount 컨테이너의 동기화 항목 수를 가져옵니다.
GetSyncItemEnumerator 처리기의 동기화 항목을 열거하는 인터페이스를 가져옵니다.
GetTargetViewSizePreference 시작 중인 애플리케이션의 기본 보기 크기를 검색합니다.
GetText 검색 상자의 내용을 일반 텍스트로 검색합니다.
GetThumbnail 지정된 셸 항목에 대해 캐시된 썸네일을 가져옵니다.
GetThumbnail 썸네일 이미지 및 알파 형식을 가져옵니다.
GetThumbnailByID 해당 ID가 지정된 경우 썸네일 캐시에서 썸네일을 가져옵니다.
GetThumbnailHandler 지정된 항목의 축소판 그림에 대해 요청된 썸네일 처리기를 가져옵니다.
GetThumbnailStream 썸네일 스트림을 가져옵니다. 이 메서드는 내부 전용이며 사진 애플리케이션에서만 호출할 수 있습니다.
GetTime 생성 시간을 가져옵니다.
GetTimes

파일 시간을 가져옵니다.
GetTitle 지정된 Windows 탐색기 명령 항목을 시작하는 단추 또는 메뉴 항목의 제목 텍스트를 가져옵니다.
GetTitle Deprecated. 브라우저 창의 제목을 검색합니다.
GetToolTip 지정된 Windows 탐색기 명령 항목과 연결된 도구 설명 문자열을 가져옵니다.
GetToolTipText Deprecated. UI에서 유니코드 표시 문자열로 사용되는 이동 항목에 대한 도구 설명 텍스트를 가져옵니다.
GetTransferManifest 온라인 인쇄 마법사 또는 네트워크 위치 추가 마법사와 같이 게시 마법사에서 수행하는 파일 전송 작업에 대한 전송 매니페스트를 가져옵니다.
GetTravelEntry Deprecated. 현재 항목의 위치를 기준으로 여행 로그의 이동 항목을 가져옵니다.
GetTravelLog Deprecated. 브라우저의 ITravelLog를 검색합니다.
GetType 동기화 센터의 처리기 유형을 가져옵니다.
GetTypeLabel 처리기 형식에 대한 레이블을 가져옵니다. 일반적으로 디바이스의 모델 또는 해당하는 처리기별 ID 문자열을 제공합니다.
GetTypeLabel 항목 형식에 대한 레이블을 가져옵니다. 일반적으로 디바이스의 모델 또는 해당 항목별 ID 문자열을 제공합니다.
GetUIMode IExplorerCommand 기반 Shell 동사 처리기가 애플리케이션이 호출된 호스트 구성 요소의 UI 모드를 쿼리할 수 있도록 합니다.

GetUIName 애플리케이션의 표시 이름을 검색합니다.
GetUInt32 지정된 속성 키의 UInt32 값을 가져옵니다.
GetUInt64 지정된 속성 키의 UInt64 값을 가져옵니다.
GetUIObjectOf 지정된 파일 개체 또는 폴더에 대한 작업을 수행하는 데 사용할 수 있는 개체를 가져옵니다.
GetUsage 사용 중인 파일이 사용되는 방식을 나타내는 값을 가져옵니다.
GetUserProfileDirectoryA 지정된 사용자 프로필의 루트 디렉터리에 대한 경로를 검색합니다. (ANSI)
GetUserProfileDirectoryW 지정된 사용자 프로필의 루트 디렉터리에 대한 경로를 검색합니다. (유니코드)
GetUserSid 이 자격 증명과 연결된 사용자의 SID(보안 식별자)를 검색합니다.
GetValidCharacters 호출되는 네임스페이스에서 유효하거나 잘못된 각 문자가 포함된 문자열을 로드합니다.
GetValue 현재 애플리케이션 호스트 환경이 데스크톱 또는 몰입형 모드에 있는지 여부를 확인합니다.
GetValue 사용자에 대해 지정된 속성 값 집합을 검색합니다.
GetView 이 IShellView2 구현에서 지원하는 다른 모든 VID(보기 식별자)와 함께 현재 또는 기본 셸 보기를 요청합니다.

GetViewFlags 공통 대화 브라우저에 대해 특별한 사용자 지정이 필요한지 보기에서 결정해야 하는 경우 호출됩니다.
GetViewMode 폴더의 논리 뷰 모드를 가져옵니다.
GetViewModeAndIconSize 보기에 적용된 현재 보기 모드 및 아이콘 크기를 가져옵니다.
GetViewProperty 뷰의 캐시에서 지정된 속성 키의 속성 값을 가져옵니다.
GetViewRect Deprecated. 창의 허용된 크기를 협상하는 데 사용되는 값을 검색합니다.
GetViewStateStream 뷰별 상태 정보의 스토리지에 사용할 수 있는 IStream 인터페이스를 가져옵니다.
GetViewWindow Deprecated. IBrowserService2::CreateViewWindow에서 만든 브라우저 보기 창에 대한 직접 액세스를 제공합니다.
GetVisibleItem 뷰에서 지정된 인덱스와 관련하여 표시되는 다음 항목을 가져옵니다.
GetWallpaper 현재 바탕 화면 배경 화면을 가져옵니다.
GetWindowContext 미리 보기 호스트의 바로 가기 키 목록을 가져옵니다.
GetWindowContextHelpId 지정된 창과 연결된 도움말 컨텍스트 식별자(있는 경우)를 검색합니다.
GetWindowDesktopId 제공된 최상위 창을 호스팅하는 가상 데스크톱의 식별자를 가져옵니다.

GetWindowSubclass 지정된 창 하위 클래스 콜백에 대한 참조 데이터를 검색합니다.
GetWorkingDirectory Shell 링크 개체의 작업 디렉터리 이름을 가져옵니다. (ANSI)
GetWorkingDirectory Shell 링크 개체의 작업 디렉터리 이름을 가져옵니다. (유니코드)
GroupChildWithParent ISuspensionDependencyManager::GroupChildWithParent 메서드
HandleEvent 애플리케이션이 처리하도록 등록된 형식의 콘텐츠가 없는 자동 실행 디바이스 이벤트를 처리합니다.
HandleEventWithContent 구현되지 않았습니다. (IHWEventHandler.HandleEventWithContent)
HandleEventWithHWND 애플리케이션이 처리하도록 등록되지 않은 콘텐츠 형식이 포함된 자동 실행 디바이스 이벤트를 처리합니다. 이 메서드는 프로세스에 상승된 권한이 필요한 경우 UI를 표시할 수 있도록 소유자 창에 대한 핸들을 제공합니다.
HandleMenuMsg IContextMenu 인터페이스의 클라이언트 개체가 소유자가 그린 메뉴 항목과 연결된 메시지를 처리할 수 있도록 합니다.
HandleMenuMsg2 IContextMenu3 인터페이스의 클라이언트 개체가 소유자가 그린 메뉴 항목과 연결된 메시지를 처리할 수 있도록 허용합니다.
HandleRename 항목의 식별자를 변경하는 데 사용됩니다.
HasFocusIO 개체의 창 중 하나에 키보드 포커스가 있는지 여부를 확인합니다.

HashData 데이터 배열을 해시합니다.
HasRecordableDrive 시스템에서 쓰기 기능이 있는 CD 드라이브를 검사하고, 해당 드라이브가 발견되면 TRUE를 반환합니다.
HasUserCancelled 사용자가 작업을 취소했는지 여부를 확인합니다.
HideDeskBand 지정된 책상 밴드를 숨깁니다.
숨기기 입력 창이 디스플레이에서 나가려고 할 때 호출됩니다.
HitTest 지정된 지점이 있는 항목(있는 경우)을 검색합니다.
HrInit 작업 표시줄 목록 개체를 초기화합니다. 다른 ITaskbarList 메서드를 호출하려면 먼저 이 메서드를 호출해야 합니다.
IDListContainerIsConsistent IDList의 컨테이너 구조가 유효한지 확인합니다.
IEGetDisplayName Deprecated. PIDL(항목 식별자 목록)에 대한 포인터에 해당하는 URL을 검색합니다.
IEParseDisplayName Deprecated. URL을 PIDL(항목 식별자 목록)에 대한 포인터로 구문 분석합니다. (IBrowserService.IEParseDisplayName)
IEParseDisplayNameEx Deprecated. URL을 PIDL(항목 식별자 목록)에 대한 포인터로 구문 분석합니다. (IBrowserService3.IEParseDisplayNameEx)
IID_PPV_ARGS

사용되는 인터페이스 포인터의 형식에 따라 요청된 인터페이스의 IID 값을 자동으로 제공하여 인터페이스 포인터를 검색하는 데 사용됩니다. 이렇게 하면 컴파일 시간에 전달된 값의 형식을 확인하여 일반적인 코딩 오류를 방지할 수 있습니다.
ILAppendID ITEMIDLIST 구조체에 SHITEMID 구조체를 추가하거나 앞에 추가합니다.
ILClone ITEMIDLIST 구조를 복제합니다.
ILCloneChild 자식 ITEMIDLIST 구조를 복제합니다.
ILCloneFirst ITEMIDLIST 구조체의 첫 번째 SHITEMID 구조를 복제합니다.
ILCloneFull 전체 또는 절대 ITEMIDLIST 구조를 복제합니다.
ILCombine 두 ITEMIDLIST 구조를 결합합니다.
ILCreateFromPath ILCreateFromPath 함수는 지정된 파일 경로와 연결된 ITEMIDLIST 구조를 반환합니다.
ILCreateFromPathA 지정된 파일 경로와 연결된 ITEMIDLIST 구조를 반환합니다. (ILCreateFromPathA)
ILCreateFromPathW ILCreateFromPathW(유니코드) 함수는 지정된 파일 경로와 연결된 ITEMIDLIST 구조를 반환합니다.
ILFindChild 지정된 ITEMIDLIST 구조체가 다른 ITEMIDLIST 구조체의 자식인지 여부를 확인합니다.
ILFindLastID ITEMIDLIST 구조체의 마지막 SHITEMID 구조체에 대한 포인터를 반환합니다.
ILFree

셸에서 할당한 ITEMIDLIST 구조를 해제합니다.
ILGetNext ITEMIDLIST 구조체에서 다음 SHITEMID 구조를 검색합니다. (ILGetNext)
ILGetSize ITEMIDLIST 구조체의 크기(바이트)를 반환합니다.
ILIsAligned 32비트 아키텍처의 DWORD 및 64비트 아키텍처의 QWORD인 포인터 경계에 상수 ITEMIDLIST가 정렬되는지 확인합니다.
ILIsChild PIDL(항목 식별자 목록)에 대한 포인터가 정확히 하나의 SHITEMID가 있는 PIDL인 자식 PIDL인지 확인합니다.
ILIsEmpty ITEMIDLIST 구조체가 비어 있는지 확인합니다.
ILIsEqual 두 ITEMIDLIST 구조체가 이진 비교에서 같은지 테스트합니다.
ILIsParent ITEMIDLIST 구조체가 다른 ITEMIDLIST 구조체의 부모인지 여부를 테스트합니다.
ILLoadFromStream Deprecated. 스트림에서 ITEMIDLIST 구조를 로드합니다.
ILLoadFromStreamEx 이 함수는 변경되거나 사용할 수 없습니다.
ILNext ITEMIDLIST 구조체에서 다음 SHITEMID 구조를 검색합니다. (ILNext)
ILRemoveLastID ITEMIDLIST 구조체에서 마지막 SHITEMID 구조를 제거합니다.
ILSaveToStream

ITEMIDLIST 구조를 스트림에 저장합니다.
ILSkip 정렬되지 않은 상수의 상대 ITEMIDLIST 구조에서 지정된 바이트 수를 건너뜁니다.
IncludeItem 지정된 셸 항목 상태를 보기에 포함하도록 설정합니다.
IncludeItems 보기에서 메뉴에 표시된 항목을 필터링할 수 있습니다.
IncludeObject 공통 대화 상자에서 보기가 표시하는 개체를 필터링할 수 있습니다.
InetIsOffline 시스템이 인터넷에 연결되어 있는지 여부를 확인합니다.
Init 자동 완성 개체를 초기화합니다.
Init 동기화 센터에서 연결할 이벤트를 제공할 수 있으므로 ISyncMgrUIOperation::Run에서 작동할 이벤트를 알 수 있습니다.
Init IQueryAssociations 인터페이스를 초기화하고 루트 키를 적절한 ProgID로 설정합니다.
초기화 IFolderView 개체를 호스트하는 개체를 초기화합니다.
초기화 IHWEventHandler 인터페이스의 구현을 포함하는 개체를 초기화합니다.
초기화 전송할 파일, 사용할 설정 및 만들 마법사 유형을 사용하여 게시 마법사 개체를 초기화합니다.
초기화 IColumnProvider 인터페이스를 초기화합니다.

초기화 파일 경로를 사용하여 처리기를 초기화합니다.
초기화 스트림을 사용하여 처리기를 초기화합니다.
초기화 작업 진행률 대화 상자에 대한 세부 정보를 제공합니다.
초기화 . (IDefaultFolderMenuInitialize.Initialize)
초기화 탐색할 브라우저를 준비합니다.
초기화 애플리케이션에서 지정한 명령 이름 및 등록된 속성을 사용하여 IExplorerCommandState, IExecuteCommand 또는 IDropTarget의 구현을 공유하는 개체를 초기화합니다.
초기화 바인딩 컨텍스트를 사용하여 처리기를 초기화합니다.
초기화 IShellItem을 사용하여 처리기를 초기화합니다.
초기화 IPropertyStore를 사용하여 처리기를 초기화합니다.
초기화 데스크톱 앱에서 사용되는 Windows 런타임 개체에서 사용할 소유자 창을 지정합니다.
초기화 INameSpaceTreeControl 개체를 초기화합니다.
초기화 전달된 정보에 따라 셸 폴더 개체를 초기화하도록 지시합니다.

초기화 원격 컴퓨터에서 호출된 네임스페이스 확장을 초기화하거나 열거할 때 Windows 탐색기 또는 Windows Internet Explorer에서 사용됩니다.
초기화 속성 시트 확장, 바로 가기 메뉴 확장 또는 끌어서 놓기 처리기를 초기화합니다.
초기화 메뉴 밴드를 초기화합니다.
초기화 등록된 애플리케이션 처리기에서 동기화 관리자가 호출하여 처리기가 동기화 이벤트를 처리하는지 여부를 확인합니다.
InitializeBitmap 지정된 비트맵을 사용하여 새 ISharedBitmap 개체를 초기화합니다.
InitializeDownloadManager Deprecated. 기본 클래스에서 다운로드 관리자를 사용하도록 설정합니다.
InitializeEx 폴더를 초기화하고 네임스페이스에서 해당 위치를 지정합니다. 폴더가 바로 가기인 경우 이 메서드는 대상 폴더의 위치도 지정합니다.
InitializeFromBitmap 창 없는 컨트롤에 대한 끌기 이미지 관리자를 초기화합니다.
InitializeFromWindow 창을 사용하여 컨트롤에 대한 끌기 이미지 관리자를 초기화합니다.
InitializeProgressDialog 네임스페이스 워크 중에 표시되는 진행률 대화 상자의 창 제목 및 취소 단추 텍스트를 초기화합니다.
InitializeSFB IShellFolderBand 개체를 초기화합니다.
InitializeTransitionSite

Deprecated. 브라우저 보기 창에서 전환을 사용하도록 설정합니다.
InitializeTravelLog Deprecated. 파생 클래스가 새 창에서 사용할 탐색 레코드를 지정할 수 있도록 허용합니다.
InitNetworkAddressControl 네트워크 주소 제어 창 클래스를 초기화합니다.
InitWizard 동기화 일정 마법사를 초기화합니다.
InOperation 대상에서 데이터를 비동기적으로 추출하는지 여부를 확인하기 위해 드롭 소스에서 호출됩니다.
InsertItem ITEMIDLIST 구조체를 이러한 구조 목록으로 추가합니다.
InsertMenuEntries Deprecated. 지정된 메뉴에 항목을 삽입합니다.
InsertMenusSB 확장 네임스페이스를 보거나 사용할 때 표시되는 복합 메뉴에 해당 메뉴 그룹을 컨테이너에 삽입할 수 있습니다.
InsertRoot 트리의 루트 항목에 셸 항목을 삽입합니다.
설치 애플리케이션 게시자가 게시한 애플리케이션을 설치합니다. 이 메서드는 사용자가 제어판의 프로그램 추가/제거에서 나중에 추가 또는 추가를 선택할 때 호출됩니다.
Install2 여러 창이 동일한 스레드에서 활성화되지 않도록 하면서 애플리케이션 게시자가 게시한 애플리케이션을 설치합니다.
Int8Add INT8 형식의 두 값을 추가합니다.

Int8Mult INT8 형식의 두 값을 곱합니다.
Int8Sub INT8 형식의 한 값을 다른 값에서 뺍니다.
Int8ToUChar INT8 형식의 값을 UCHAR 형식의 값으로 변환합니다.
Int8ToUInt INT8 형식의 값을 UINT 형식의 값으로 변환합니다.
Int8ToUInt8 INT8 형식의 값을 UINT8 형식의 값으로 변환합니다.
Int8ToUIntPtr INT8 형식의 값을 UINT_PTR 형식의 값으로 변환합니다.
Int8ToULong INT8 형식의 값을 ULONG 형식의 값으로 변환합니다.
Int8ToULongLong INT8 형식의 값을 ULONGLONG 형식의 값으로 변환합니다.
Int8ToULongPtr INT8 형식의 값을 ULONG_PTR 형식의 값으로 변환합니다.
Int8ToUShort INT8 형식의 값을 USHORT 형식의 값으로 변환합니다.
IntAdd INT 형식의 두 값을 추가합니다.
IntlStrEqNA 지역화된 두 문자열의 시작 부분에서 지정된 수의 문자에 대해 대/소문자를 구분하여 비교합니다. (ANSI)
IntlStrEqNIA

지역화된 두 문자열의 시작 부분에서 지정된 수의 문자를 대/소문자를 구분하지 않는 비교를 수행합니다. (ANSI)
IntlStrEqNIW 지역화된 두 문자열의 시작 부분에서 지정된 수의 문자를 대/소문자를 구분하지 않는 비교를 수행합니다. (유니코드)
IntlStrEqNW 지역화된 두 문자열의 시작 부분에서 지정된 수의 문자에 대해 대/소문자를 구분하여 비교합니다. (유니코드)
IntlStrEqWorkerA 지역화된 두 문자열의 시작 부분에서 지정된 문자 수를 비교합니다. (ANSI)
IntlStrEqWorkerW 지역화된 두 문자열의 시작 부분에서 지정된 문자 수를 비교합니다. (유니코드)
IntMult INT 형식의 두 값을 곱합니다.
IntPtrAdd INT_PTR 형식의 두 값을 추가합니다.
IntPtrMult INT_PTR 형식의 두 값을 곱합니다.
IntPtrSub INT_PTR 형식의 한 값을 다른 값에서 뺍니다.
IntPtrToChar 형식 INT_PTR 값을 CHAR 형식의 값으로 변환합니다.
IntPtrToInt 형식 INT_PTR 값을 INT 형식의 값으로 변환합니다.
IntPtrToInt8 INT_PTR 형식의 값을 INT8 형식의 값으로 변환합니다.

IntPtrToLong INT_PTR 형식의 값을 LONG 형식의 값으로 변환합니다.
IntPtrToLongPtr 형식 INT_PTR 값을 형식 LONG_PTR 값으로 변환합니다.
IntPtrToShort INT_PTR 형식의 값을 SHORT 형식의 값으로 변환합니다.
IntPtrToUChar INT_PTR 형식의 값을 UCHAR 형식의 값으로 변환합니다.
IntPtrToUInt INT_PTR 형식의 값을 UINT 형식의 값으로 변환합니다.
IntPtrToUInt8 INT_PTR 형식의 값을 UINT8 형식의 값으로 변환합니다.
IntPtrToUIntPtr 형식 INT_PTR 값을 UINT_PTR 형식의 값으로 변환합니다.
IntPtrToULong INT_PTR 형식의 값을 ULONG 형식의 값으로 변환합니다.
IntPtrToULongLong INT_PTR 형식의 값을 ULONGLONG 형식의 값으로 변환합니다.
IntPtrToULongPtr 형식 INT_PTR 값을 ULONG_PTR 형식의 값으로 변환합니다.
IntPtrToUShort INT_PTR 형식의 값을 USHORT 형식의 값으로 변환합니다.
IntSub INT 형식의 한 값을 다른 값에서 뺍니다.
IntToChar

INT 형식의 값을 CHAR 형식의 값으로 변환합니다.
IntToInt8 INT 형식의 값을 INT8 형식의 값으로 변환합니다.
IntToShort INT 형식의 값을 SHORT 형식의 값으로 변환합니다.
IntToUChar INT 형식의 값을 UCHAR 형식의 값으로 변환합니다.
IntToUInt INT 형식의 값을 UINT 형식의 값으로 변환합니다.
IntToUInt8 INT 형식의 값을 UINT8 형식의 값으로 변환합니다.
IntToULong INT 형식의 값을 ULONG 형식의 값으로 변환합니다.
IntToULongLong INT 형식의 값을 UINT_PTR 형식의 값으로 변환합니다.
IntToUShort INT 형식의 값을 USHORT 형식의 값으로 변환합니다.
InvalidateItem 메뉴 밴드에서 항목을 다시 그립니다.
호출 연결된 처리기를 직접 호출합니다.
호출 연결된 애플리케이션 처리기를 호출합니다.
호출 Windows 탐색기 명령을 호출합니다.

호출 Deprecated. 해당 페이지로 이동하여 이동 항목을 호출합니다.
InvokeCommand 바로 가기 메뉴 항목과 연결된 명령을 수행합니다.
InvokeVerbOnSelection 현재 선택 영역에서 지정된 동사를 호출합니다.
IsActive 처리기를 동기화할 수 있는지 여부를 나타내는 값을 가져옵니다.
IsAnimated 이미지에 애니메이션 효과를 주는지 여부를 결정합니다.
IsApplicationViewStateSupported 특정 스푸핑된 표시 크기 및 배율 인수 설정에 대해 특정 애플리케이션 보기 상태가 지원되는지 여부를 결정합니다.
IsBkDropTarget IsBkDropTarget을 변경하거나 사용할 수 없습니다.
IsCharSpaceA 문자가 공백을 나타내는지 여부를 결정합니다. (ANSI)
IsCharSpaceW 문자가 공백을 나타내는지 여부를 결정합니다. (유니코드)
IsConnected 처리기(일반적으로 일부 유형의 외부 디바이스)가 연결되어 있는지 여부를 나타내는 값을 가져옵니다.
IsConnected 항목(일반적으로 일부 유형의 외부 디바이스)이 연결되어 있는지 여부를 나타내는 값을 생성합니다.
IsControlWindowShown Deprecated. 지정된 프레임 컨트롤이 현재 표시되는지 여부를 나타내는 값을 검색합니다.

IsDecoded IShellImageData::D ecode를 호출하여 이미지가 디코딩되었는지 여부를 확인합니다. 이미지가 처음 디코딩되지 않은 경우 많은 작업이 실패 코드를 반환합니다.
IsDelegateFolder 팩터리에서 대리자 폴더인지 여부를 지정하는 부울 값을 가져옵니다.
IsDeskBandShown 책상 밴드가 표시되는지 여부를 나타냅니다.
IsDropOnSource 현재 끌어서 놓기 또는 잘라내기 및 붙여넣기 작업의 대상이 원본과 같은지 확인합니다.
IsEditable 이미지를 편집할 수 있는지 여부를 결정합니다.
사용 가능 처리기를 사용할 수 있는지 여부를 나타내는 값을 가져옵니다.
사용 가능 항목을 사용할 수 있는지 여부를 나타내는 값을 생성합니다.
IsEntryInStore 이미지가 저장소에 있는지 확인합니다.
IsEqualPropertyKey 두 PROPERTYKEY 구조체의 멤버를 비교하고 같은지 여부를 반환합니다.
IsInstalled 지정된 애플리케이션이 현재 설치되어 있는지 여부를 나타내는 값을 가져옵니다.
IsInternetESCEnabled Windows Internet Explorer가 향상된 보안 구성에 있는지 여부를 확인합니다.
IsLauncherVisible 시작 화면이 표시되는지 여부를 나타내는 값을 가져옵니다.

IsMember 로컬 컴퓨터가 HomeGroup의 구성원인지 여부를 확인합니다.
IsMemberOf 셸 개체의 아이콘에 아이콘 오버레이를 추가할지 여부를 지정합니다.
IsMenuMessage 메시지 펌프는 이 메서드를 호출하여 메시지가 COM(구성 요소 개체 모델) 개체로 리디렉션되어야 하는지 확인합니다.
IsMoveInSameFolder 이 뷰가 현재 끌어서 놓기 또는 잘라내기 및 붙여넣기 작업(놓기 대상 개체에서 사용됨)을 제공하는지 확인합니다.
IsMultipage 이미지가 여러 페이지 TIFF(태그가 지정된 이미지 파일 형식) 이미지인지 여부를 확인합니다.
IsNetDrive 드라이브가 네트워크 드라이브인지 여부를 테스트합니다.
IsOS 지정된 운영 체제 및 운영 체제 기능을 확인합니다.
IsPrintable 이미지를 인쇄할 수 있는지 여부를 결정합니다.
IsRecommended 애플리케이션이 쿼리된 파일 형식에 대한 권장 처리기로 등록되었는지 여부를 나타냅니다.
IsRunning 썸네일 추출과 같은 작업의 상태에 대한 정보를 요청합니다.
IsTransparent 이미지가 투명한지 여부를 확인합니다.
IStream_Copy 스트림을 다른 스트림에 복사합니다.

IStream_Read 지정된 스트림에서 바이트를 읽고 모든 바이트를 성공적으로 읽었는지 여부를 나타내는 값을 반환합니다.
IStream_ReadPidl IStream 개체에서 PIDLIST_RELATIVE 개체로 PIDL(항목 식별자 목록)에 대한 포인터를 읽습니다.
IStream_ReadStr 스트림에서 읽고 문자열에 씁니다.
IStream_Reset 지정된 스트림의 검색 위치를 스트림의 시작 부분으로 이동합니다.
IStream_Size 지정된 스트림의 크기(바이트)를 검색합니다.
IStream_Write 버퍼에서 지정된 스트림에 알 수 없는 형식의 데이터를 씁니다.
IStream_WritePidl PCUIDLIST_RELATIVE 개체의 PIDL(항목 식별자 목록)에 대한 포인터를 IStream 개체에 씁니다.
IStream_WriteStr 문자열에서 읽고 스트림에 씁니다.
IsUserAnAdmin IsUserAnAdmin을 변경하거나 사용할 수 없습니다.
IsVector 이미지가 벡터 이미지인지 여부를 확인합니다.
IsWindowOnCurrentVirtualDesktop 제공된 창이 현재 활성 가상 데스크톱에 있는지 여부를 나타냅니다.
항목 지정된 인덱스로 등록된 셸 창을 반환합니다.
항목

폴더 뷰에서 인덱스별로 특정 항목의 식별자를 가져옵니다.
ItemCount 폴더의 항목 수를 가져옵니다. 모든 항목의 수 또는 선택한 항목 수와 같은 하위 집합일 수 있습니다.
ItemPostPaint 네임스페이스 트리 컨트롤의 항목을 그린 후 호출됩니다.
ItemPrePaint 네임스페이스 트리 컨트롤의 항목을 그리기 전에 호출됩니다.
항목 폴더 뷰의 항목 컬렉션을 기반으로 하는 열거형 개체의 주소를 가져옵니다.
IUnknown_AtomicRelease COM(구성 요소 개체 모델) 포인터를 해제하고 NULL로 설정합니다.
IUnknown_GetSite 지정된 개체의 IObjectWithSite::GetSite 메서드를 호출합니다.
IUnknown_GetWindow GetWindow 메서드가 있는 다양한 인터페이스를 쿼리하여 COM(구성 요소 개체 모델) 개체에서 창 핸들을 검색하려고 시도합니다.
IUnknown_QueryService 지정된 개체에서 서비스에 대한 인터페이스를 검색합니다.
IUnknown_Set COM(구성 요소 개체 모델) 인터페이스 포인터의 값을 변경하고 이전 인터페이스를 해제합니다.
IUnknown_SetSite IObjectWithSite::SetSite 메서드를 호출하여 지정된 개체의 사이트를 설정합니다.
KeepItems 전달된 셸 항목을 유지합니다.
KeepOther 충돌하는 버전을 일반적으로 원본의 병합된 버전인 다른 셸 항목으로 바꿉니다.

KeepRecent 최신 복사본을 유지합니다.
죽이다 작업을 중지할 것을 요청합니다.
LaunchAdvancedAssociationUI 사용자가 pszAppRegName에 지정된 애플리케이션에 대한 연결을 사용자 지정할 수 있는 고급 연결 대화 상자를 시작합니다.
LauncherVisibilityChange 시작 화면의 표시 유형이 변경되었음을 클라이언트에 알깁니다.
LeaveFolder 네임스페이스가 폴더를 통과한 후 호출됩니다. INamespaceWalkCB::EnterFolder 또는 INamespaceWalkCB::FoundItem에서 수행한 작업 다음에 필요한 정리를 수행하려면 이 메서드를 사용합니다.
LeaveFolder 폴더가 더 이상 파일 작업의 대상이 아닙니다는 알림을 보냅니다.
LinkItem 구현되지 않았습니다. (ITransferSource.LinkItem)
LoadLibraryFromItem 지정된 라이브러리 정의 파일에서 라이브러리를 로드합니다.
LoadLibraryFromKnownFolder KNOWNFOLDERID에서 참조하는 라이브러리를 로드합니다.
LoadNonloadedOverlayIdentifiers 현재 로드되지 않은 등록된 오버레이 식별자 또는 처리기를 로드합니다.
LoadUserProfileA 지정된 사용자의 프로필을 로드합니다. 프로필은 로컬 사용자 프로필 또는 로밍 사용자 프로필일 수 있습니다. (ANSI)

LoadUserProfileW 지정된 사용자의 프로필을 로드합니다. 프로필은 로컬 사용자 프로필 또는 로밍 사용자 프로필일 수 있습니다. (유니코드)
위치 입력 창의 현재 위치를 가져옵니다.
LogError 동기화 관리자 상태 대화 상자의 오류 탭에 정보, 경고 또는 오류 메시지를 기록하기 위해 등록된 애플리케이션에서 호출합니다.
LongAdd LONG 형식의 두 값을 추가합니다.
LongLongAdd LONGLONG 형식의 두 값을 추가합니다.
LongLongMult LONGLONG 형식의 두 값을 곱합니다.
LongLongSub LONGLONG 형식의 값을 다른 값에서 뺍니다.
LongLongToChar LONGLONG 형식의 값을 CHAR 형식의 값으로 변환합니다.
LongLongToInt LONGLONG 형식의 값을 INT 형식의 값으로 변환합니다.
LongLongToInt8 LONGLONG 형식의 값을 INT8 형식의 값으로 변환합니다.
LongLongToIntPtr LONGLONG 형식의 값을 INT_PTR 형식의 값으로 변환합니다.
LongLongToLong LONGLONG 형식의 값을 LONG 형식의 값으로 변환합니다.

LongLongToLongPtr LONGLONG 형식의 값을 LONG_PTR 형식의 값으로 변환합니다.
LongLongToShort LONGLONG 형식의 값을 SHORT 형식의 값으로 변환합니다.
LongLongToUChar LONGLONG 형식의 값을 UCHAR 형식의 값으로 변환합니다.
LongLongToUInt LONGLONG 형식의 값을 UINT 형식의 값으로 변환합니다.
LongLongToUInt8 LONGLONG 형식의 값을 UINT8 형식의 값으로 변환합니다.
LongLongToULong LONGLONG 형식의 값을 ULONG 형식의 값으로 변환합니다.
LongLongToULongLong LONGLONG 형식의 값을 ULONGLONG 형식의 값으로 변환합니다.
LongLongToUShort LONGLONG 형식의 값을 USHORT 형식의 값으로 변환합니다.
LongMult LONG 형식의 두 값을 곱합니다.
LongPtrAdd LONG_PTR 형식의 두 값을 추가합니다.
LongPtrMult LONG_PTR 형식의 두 값을 곱합니다.
LongPtrSub LONG_PTR 형식의 한 값을 다른 값에서 뺍니다.
LongPtrToChar

형식 LONG_PTR 값을 CHAR 형식의 값으로 변환합니다.
LongPtrToInt LONG_PTR 형식의 값을 INT 형식의 값으로 변환합니다.
LongPtrToInt8 LONG_PTR 형식의 값을 INT8 형식의 값으로 변환합니다.
LongPtrToIntPtr 형식 LONG_PTR 값을 INT_PTR 형식의 값으로 변환합니다.
LongPtrToLong LONG_PTR 형식의 값을 LONG 형식의 값으로 변환합니다.
LongPtrToShort LONG_PTR 형식의 값을 SHORT 형식의 값으로 변환합니다.
LongPtrToUChar LONG_PTR 형식의 값을 UCHAR 형식의 값으로 변환합니다.
LongPtrToUInt LONG_PTR 형식의 값을 UINT 형식의 값으로 변환합니다.
LongPtrToUInt8 LONG_PTR 형식의 값을 UINT8 형식의 값으로 변환합니다.
LongPtrToUIntPtr 형식 LONG_PTR 값을 형식 UINT_PTR 값으로 변환합니다.
LongPtrToULong LONG_PTR 형식의 값을 ULONG 형식의 값으로 변환합니다.
LongPtrToULongLong LONG_PTR 형식의 값을 ULONGLONG 형식의 값으로 변환합니다.
LongPtrToULongPtr 형식 LONG_PTR 값을 형식 ULONG_PTR 값으로 변환합니다.

LongPtrToUShort LONG_PTR 형식의 값을 USHORT 형식의 값으로 변환합니다.
LongSub LONG 형식의 값을 다른 값에서 뺍니다.
LongToChar LONG 형식의 값을 CHAR 형식의 값으로 변환합니다.
LongToInt LONG 형식의 값을 INT 형식의 값으로 변환합니다.
LongToInt8 LONG 형식의 값을 INT8 형식의 값으로 변환합니다.
LongToIntPtr LONG 형식의 값을 INT_PTR 형식의 값으로 변환합니다.
LongToShort LONG 형식의 값을 SHORT 형식의 값으로 변환합니다.
LongToUChar LONG 형식의 값을 UCHAR 형식의 값으로 변환합니다.
LongToUInt LONG 형식의 값을 UINT 형식의 값으로 변환합니다.
LongToUInt8 LONG 형식의 값을 UINT8 형식의 값으로 변환합니다.
LongToUIntPtr LONG 형식의 값을 UINT_PTR 형식의 값으로 변환합니다.
LongToULong LONG 형식의 값을 ULONG 형식의 값으로 변환합니다.
LongToULongLong

LONG 형식의 값을 ULONGLONG 형식의 값으로 변환합니다.
LongToULongPtr LONG 형식의 값을 ULONG_PTR 형식의 값으로 변환합니다.
LongToUShort LONG 형식의 값을 USHORT 형식의 값으로 변환합니다.
LPFNDFMCALLBACK LPFNDFMCALLBACK을 변경하거나 사용할 수 없습니다.
LPFNVIEWCALLBACK 시스템 폴더 뷰 개체에서 사용하는 콜백 함수의 프로토타입을 정의합니다. 이 함수는 기본적으로 IShellFolderViewCB의 기능을 복제합니다.
MakeDefault 애플리케이션을 이 파일 형식의 기본 애플리케이션으로 설정합니다.
MAKEDLLVERULL DLL 버전 정보를 ULONGLONG 값으로 압축하는 데 사용됩니다.
MakeProminent 추가된 다른 컨트롤과 비교하여 눈에 띄는 컨트롤을 대화 상자에 배치합니다.
MapColumnToSCID 열을 FMTID(적절한 속성 집합 ID) 및 PID(속성 ID)로 변환합니다.
MarkForDelete 삭제 표시입니다.
MarkFullscreenWindow 창을 전체 화면으로 표시합니다.
병합 IFileSyncMergeHandler::Merge 메서드
MessageSFVCB 시스템 폴더 뷰 개체와 시스템 폴더 뷰 콜백 개체 간의 통신을 허용합니다.

MIMEAssociationDialogA 등록되지 않은 MIME 콘텐츠 형식 대화 상자를 실행합니다. Windows XP SP2(서비스 팩 2) 이상:_This 함수는 더 이상 지원되지 않습니다. (ANSI)
MIMEAssociationDialogW 등록되지 않은 MIME 콘텐츠 형식 대화 상자를 실행합니다. Windows XP SP2(서비스 팩 2) 이상:_This 함수는 더 이상 지원되지 않습니다. (유니코드)
MoveIcons 이 메서드는 구현되지 않습니다. (IShellFolderView.MoveIcons)
MoveItem 지정된 대상으로 이동할 단일 항목을 선언합니다.
MoveItem 볼륨/네임스페이스 내에서 항목을 이동하여 새 위치에서 IShellItem을 반환합니다.
MoveItems 지정된 대상으로 이동할 항목 집합을 선언합니다.
MoveWindowToDesktop 창을 지정된 가상 데스크톱으로 이동합니다.
NavigateToPidl Deprecated. 브라우저를 PIDL(항목 식별자 목록)에 대한 포인터로 표시된 위치로 이동합니다.
NetAddr_DisplayErrorTip 네트워크 주소 컨트롤과 연결된 풍선 팁에 오류 메시지를 표시합니다.
NetAddr_GetAddress 네트워크 주소가 지정된 형식 및 형식을 준수하는지 여부를 나타냅니다.
NetAddr_GetAllowType 지정된 네트워크 주소 컨트롤에서 허용하는 네트워크 주소 유형을 검색합니다.
NetAddr_SetAllowType 지정된 네트워크 주소 컨트롤에서 허용하는 네트워크 주소 유형을 설정합니다.

NewItem 지정된 위치에 만들 새 항목을 선언합니다.
다음 지정된 개수의 요소를 검색합니다.
다음 현재 요소를 직접 따르는 지정된 개수의 요소를 검색합니다.
다음 하나 이상의 검색 개체에 대한 정보를 요청하는 데 사용됩니다.
다음 지정된 수의 IDLIST_ABSOLUTE 항목을 검색합니다.
다음 열거형 시퀀스에서 지정된 수의 항목 식별자를 검색하고 검색된 항목 수만큼 현재 위치를 앞으로 이동합니다.
다음 지정한 다음 개체의 수와 형식을 가져옵니다.
다음 다음 SHELL_ITEM_RESOURCE 구조를 가져옵니다.
다음 열거형에서 하나 이상의 IShellItem 인터페이스 배열을 가져옵니다.
다음 열거형의 다음 IPublishedApp 개체를 가져옵니다.
다음 충돌 저장소에서 충돌의 다음 일괄 처리를 가져옵니다.
다음 이벤트 저장소에서 이벤트의 다음 일괄 처리를 가져옵니다.

다음 처리기에서 동기화 항목의 다음 일괄 처리를 가져옵니다.
다음 열거자 목록의 다음 셀트 요소를 열거하고 pceltFetched의 열거된 요소의 실제 수와 함께 rgelt로 반환합니다.
NextFrame 애니메이션 이미지의 다음 프레임으로 전환합니다.
NextPage 다중 페이지 이미지의 다음 페이지로 전환합니다. 연결된 모든 애니메이션이 다시 설정됩니다.
알림 이벤트가 발생했음을 호스팅하는 일반 대화 상자에 알리기 위해 셸 보기에서 호출됩니다.
NotifyRedirect Deprecated. 필요한 경우 탐색하여 브라우저를 지정된 포인터로 PIDL(항목 식별자 목록)으로 업데이트합니다. 이 메서드는 페이지가 리디렉션될 때 호출됩니다.
오프라인 Deprecated. 기본 클래스와 파생 클래스 간의 상태를 동기화하여 브라우저의 오프라인 상태를 확인하고 업데이트합니다.
OnActivated 셸 창의 활성화 상태가 변경되면 발생합니다.
OnAfterContextMenu 상황에 맞는 메뉴가 표시된 후 호출됩니다.
OnAfterExpand IShellItem이 확장된 후 호출됩니다.
OnBalloonUserClick 사용자가 풍선을 클릭할 때 호출됩니다. 애플리케이션은 클릭 중인 풍선에 적합한 동작으로 응답할 수 있습니다.

OnBeforeContextMenu 상황에 맞는 메뉴가 표시되기 전에 호출됩니다. 을 사용하면 클라이언트에서 메뉴 항목을 추가할 수 있습니다.
OnBeforeExpand IShellItem이 확장되기 전에 호출됩니다.
OnBeforeItemDelete IShellItem 및 모든 자식이 삭제되기 전에 호출됩니다.
OnBeforeStateImageChange 지정된 IShellItem의 상태 아이콘이 변경되기 전에 호출됩니다.
OnBeginLabelEdit IShellItem이 편집 모드로 전환되기 전에 호출됩니다.
OnButtonClicked 사용자가 명령 단추를 클릭할 때 호출됩니다.
OnChange 네임스페이스 확장에 해당 항목에 영향을 주는 이벤트가 발생했음을 알립니다.
OnCheckButtonToggled 사용자가 확인 단추의 상태를 변경할 때 호출됩니다(확인란).
OnColumnClicked IShellView 인터페이스에서 지정된 열을 클릭한 후 호출됩니다.
OnCommand Deprecated. WM_COMMAND 메시지를 받으면 기본 클래스에서 파생 클래스를 호출합니다. 파생 클래스는 메시지를 처리합니다.
OnContextMenu 사용자가 알림 영역에서 아이콘을 마우스 오른쪽 단추로 클릭하거나 Shift+F10을 누르면 호출됩니다. 애플리케이션은 응답에 해당 상황에 맞는 메뉴를 표시해야 합니다.
OnControlActivating

EnableOpenDropDown 또는 도구 메뉴를 통해 사용자 지정된 열기 단추 드롭다운 목록이 해당 콘텐츠를 표시하려고 할 때 호출됩니다.
OnCreate Deprecated. WM_CREATE 메시지를 받으면 기본 클래스에서 파생 클래스를 호출합니다. 파생 클래스는 메시지를 처리합니다.
OnCreated 프레임에 대해 새 셸 창을 만들 때 발생합니다.
OnCreating 만들 프로세스의 매개 변수를 수정할 수 있습니다.
OnCreatingWindow 창이 만들어지면 호출됩니다. Advise가 호출된 후 자격 증명이 부모 창의 HWND를 검색할 수 있도록 합니다.
OnDefaultCommand 사용자가 보기를 두 번 클릭하거나 Enter 키를 누를 때 호출됩니다.
OnDestroy Deprecated. WM_DESTROY 메시지를 받으면 기본 클래스에서 파생 클래스를 호출합니다. 파생 클래스는 메시지를 처리합니다.
OnDoDefaultAccessibilityAction 셸 항목에 대한 기본 접근성 작업을 호출합니다.
OnDragEnter 끌기 입력 시 호출되어 지정된 대로 끌기 효과를 설정합니다.
OnDragLeave 지정된 항목에 대해 끌어서 둡니다.
OnDragOver 끌어서 놓기를 호출하여 지정된 대로 끌기 효과를 설정합니다.
OnDragPosition 트리의 동일한 수준(동일한 부모 폴더 내) 내에서 항목을 끌 때 호출됩니다.

OnDrop 드롭 시 호출되어 지정된 대로 삭제 효과를 설정합니다.
OnDropPosition 트리의 동일한 수준(동일한 부모 폴더 내) 내에서 항목을 삭제할 때 호출됩니다.
OnEndLabelEdit IShellItem이 편집 모드를 벗어나면 호출됩니다.
OnFileOk 대화 상자가 결과를 반환하기 직전에 호출됩니다.
OnFocusChangeIS 포커스가 변경되었음을 브라우저에 알립니다.
OnFolderChange 사용자가 새 폴더로 이동할 때 호출됩니다.
OnFolderChanging IFileDialogEvents::OnFolderChange 전에 호출됩니다. 이렇게 하면 구현자가 특정 위치에 대한 탐색을 중지할 수 있습니다.
OnFrameWindowActivateBS Deprecated. 활성화되거나 비활성화되는 하위 프레임 창에 대한 응답으로 기본 클래스에서 파생 클래스를 호출합니다. 파생 클래스는 작업에 대한 응답으로 수행할 작업을 결정합니다.
OnGetAccessibilityRole 셀 항목에 대한 접근성 역할을 가져옵니다.
OnGetDefaultAccessibilityAction 셀 항목에 대한 기본 접근성 동작을 가져옵니다.
OnGetToolTip 도구 설명을 제공할 수 있습니다.
OnHttpEquiv

Deprecated. 문서 개체가 IOleCommandTarget::Exec를 통해 OLECMDID_HTTPEQUIV 또는 OLECMDID_HTTPEQUIV_DONE 명령을 실행하여 HTTP-EQUIV 메타태그에 응답할 때 호출됩니다.
OnInvoke Deprecated. 앰비언트 속성을 가져옵니다.
OnInvoke 구현되지 않았습니다. (IExpDispSupportXP.OnInvoke)
OnItemAdded IShellItem이 추가된 후 호출합니다.
OnItemClick 사용자가 마우스의 단추를 클릭할 때 호출됩니다.
OnItemDeleted IShellItem이 삭제된 후 호출합니다.
OnItemSelected 콤보 상자에서 항목을 선택하거나, 사용자가 옵션 단추(라디오 단추라고도 함)를 클릭하거나 도구 메뉴에서 항목을 선택할 때 호출됩니다.
OnItemStateChanged 구현되지 않았습니다. (INameSpaceTreeControlEvents.OnItemStateChanged)
OnItemStateChanging 항목 상태가 변경되기 전에 호출합니다.
OnKeyboardInput 사용자가 키보드에서 키를 누를 때 호출됩니다.
OnLeftClick 사용자가 알림 영역에서 아이콘을 클릭할 때 호출됩니다. 애플리케이션은 응답으로 일부 사용자 지정 UI를 시작할 수 있습니다.
OnNavigate 셸 창을 새 위치로 이동할 때 발생합니다.

OnNavigationComplete Explorer 브라우저가 셸 폴더로 성공적으로 이동했음을 클라이언트에 알립니다.
OnNavigationFailed Explorer 브라우저가 셸 폴더로 이동하지 못했음을 클라이언트에 알립니다.
OnNavigationPending 셸 폴더에 대한 보류 중인 탐색기 브라우저 탐색을 클라이언트에 알립니다.
OnNotify Deprecated. WM_NOTIFY 메시지를 받으면 기본 클래스에서 파생 클래스를 호출합니다. 파생 클래스는 메시지를 처리합니다.
OnOverwrite 사용자가 파일을 덮어쓰도록 선택할 때 저장 대화 상자에서 호출됩니다.
OnPictureChange 사용자 계정을 나타내는 그림이 변경될 때 알림을 보내기 위해 호출됩니다.
OnPosRectChangeDB 사각형이 변경되었음을 개체에 알립니다.
OnPreViewCreated IShellView 인터페이스에서 지정된 미리 보기를 만든 후 호출됩니다.
OnPropertyItemCommit 구현되지 않았습니다. (INameSpaceTreeControlEvents.OnPropertyItemCommit)
OnSelect 선택 알림을 처리합니다.
OnSelectionChange 사용자가 대화 상자 보기에서 선택을 변경할 때 호출됩니다.
OnSelectionChanged 선택 영역이 변경되면 호출합니다.

OnSetFocus Deprecated. WM_SETFOCUS 메시지를 받으면 기본 클래스에서 파생 클래스를 호출합니다. 파생 클래스는 메시지를 처리합니다.
OnSetTitle 사이트 개체의 제목을 설정하거나 검색합니다.
OnShareViolation 애플리케이션이 열기 또는 저장 작업에서 발생하는 공유 위반에 응답할 수 있도록 합니다.
OnSize Deprecated. WM_SIZE 메시지를 받으면 기본 클래스에서 파생 클래스를 호출합니다. 파생 클래스는 메시지를 처리합니다.
OnStateChange uChange 매개 변수로 식별된 상태가 IShellView 인터페이스에서 변경된 후에 호출됩니다.
OnStateChanged Windows 스토어 앱 디버깅 중에 패키지 상태가 변경되면 호출됩니다.
OnTranslateAccelerator Deprecated. pMsg에서 설명하고 grfModifiers의 플래그에 의해 수정된 키 입력을 처리하도록 제어 사이트에 지시합니다.
OnTranslateAccelerator 구현되지 않았습니다. (IExpDispSupportXP.OnTranslateAccelerator)
OnTypeChange 처음 선택한 파일 형식을 애플리케이션에 알리기 위해 대화 상자가 열릴 때 호출됩니다.
OnViewCreated 탐색기 브라우저의 보기가 만들어졌으며 수정할 수 있음을 클라이언트에 알립니다.
OnViewWindowActive 보기 창 또는 자식 창 중 하나가 포커스를 가져오거나 활성화될 때 셸 보기에서 호출됩니다.
열기

저장소를 열고 잠금을 반환합니다.
열기 필요에 따라 지정된 제어판 항목을 특정 페이지로 엽니다.
OpenItem 복사할 항목을 엽니다. 리소스에 대해 열거할 수 있는 개체를 반환합니다(IShellItemResources).
OpenRegStream OpenRegStream을 변경하거나 사용할 수 없습니다. 대신 SHOpenRegStream2 또는 SHOpenRegStream을 사용합니다.
OpenResource 지정된 리소스를 엽니다.
OverlappedResult 겹치는 작업의 결과를 검색합니다.
OverlayIndexFromImageIndex 지정된 아이콘 이미지에 대한 오버레이 이미지의 인덱스입니다. 아이콘에 대한 오버레이를 찾을 수 없는 경우 오버레이를 추가할 수 있습니다.
PageInThumbnail . (IThumbnailCachePrimer.PageInThumbnail)
PAPPSTATE_CHANGE_ROUTINE 앱이 일시 중단된 상태를 입력하거나 종료할 때 앱에 알려주는 앱 정의 콜백 함수를 지정합니다.
ParseDisplayName 파일 개체 또는 폴더의 표시 이름을 항목 식별자 목록으로 변환합니다.
ParseURLA URL의 기본적인 구문 분석을 수행합니다. (ANSI)
ParseURLW URL의 기본적인 구문 분석을 수행합니다. (유니코드)
PathAddBackslashA

문자열의 끝에 백슬래시를 추가하여 경로에 대한 올바른 구문을 만듭니다. (PathAddBackslashA)
PathAddBackslashW 문자열의 끝에 백슬래시를 추가하여 경로에 대한 올바른 구문을 만듭니다. (PathAddBackslashW)
PathAddExtensionA 경로 문자열에 파일 이름 확장명을 추가합니다. (ANSI)
PathAddExtensionW 경로 문자열에 파일 이름 확장명을 추가합니다. (유니코드)
PathAllocCanonicalize 경로 문자열을 정식 형식으로 변환합니다. 이 함수는 힙의 결과를 반환한다는 점에서 PathCchCanonicalize 및 PathCchCanonicalizeEx와 다릅니다.
PathAllocCombine 두 경로 조각을 단일 경로로 연결합니다.
PathAppendA 한 경로를 다른 경로의 끝에 추가합니다. (ANSI)
PathAppendW 한 경로를 다른 경로의 끝에 추가합니다. (유니코드)
PathBuildRootA 지정된 드라이브 번호에서 루트 경로를 만듭니다. (ANSI)
PathBuildRootW 지정된 드라이브 번호에서 루트 경로를 만듭니다. (유니코드)
PathCanonicalizeA "." 및 ".."와 같은 탐색 요소를 제거하여 경로를 간소화하여 직접적이고 올바른 형식의 경로를 생성합니다. (ANSI)
PathCanonicalizeW "." 및 ".."와 같은 탐색 요소를 제거하여 경로를 간소화하여 직접적이고 올바른 형식의 경로를 생성합니다. (유니코드)

PathCchAddBackslash 문자열의 끝에 백슬래시를 추가하여 경로에 대한 올바른 구문을 만듭니다. (PathCchAddBackslash)
PathCchAddBackslashEx 문자열의 끝에 백슬래시를 추가하여 경로에 대한 올바른 구문을 만듭니다. (PathCchAddBackslashEx)
PathCchAddExtension 경로 문자열에 파일 이름 확장명을 추가합니다. 이 함수는 "\", "\?" 경로를 허용한다는 점에서 PathAddExtension과 다릅니다. 및 "\?\UNC" 접두사.
PathCchAppend 한 경로를 다른 경로의 끝에 추가합니다. 이 함수는 길이가 MAX_PATH 최종 경로로 제한된다는 점에서 PathCchAppendEx와 다릅니다. 이 함수는 "\", "\?" 경로를 허용한다는 점에서 PathAppend와 다릅니다. 및 "\?\UNC" 접두사.
PathCchAppendEx 한 경로를 다른 경로의 끝에 추가합니다. 이 함수는 더 긴 최종 경로를 생성할 수 있다는 점에서 PathCchAppend와 다릅니다. 이 함수는 "\", "\?" 경로를 허용한다는 점에서 PathAppend와 다릅니다. 및 "\?\UNC" 접두사.
PathCchCanonicalize 경로 문자열을 정식 형식으로 변환합니다. 이 함수는 길이가 MAX_PATH 최종 경로로 제한된다는 점에서 PathCchCanonicalizeEx와 다릅니다. 이 함수는 호출자가 스택에 저장된 반환된 문자열의 크기를 선언해야 한다는 점에서 PathAllocCanonicalize와 다릅니다. 이 함수는 "\", "\?" 경로를 허용한다는 점에서 PathCanonicalize와 다릅니다. 및 "\?\UNC" 접두사.
PathCchCanonicalizeEx "." 및 ".."와 같은 탐색 요소를 제거하여 경로를 간소화하여 직접적이고 올바른 형식의 경로를 생성합니다. 이 함수는 더 긴 최종 경로를 생성할 수 있다는 점에서 PathCchCanonicalize와 다릅니다. 이 함수는 호출자가 스택에 저장된 반환된 문자열의 크기를 선언해야 한다는 점에서 PathAllocCanonicalize와 다릅니다. 이 함수는 "\", "\?" 경로를 허용한다는 점에서 PathCanonicalize와 다릅니다. 및 "\?\UNC" 접두사.
PathCchCombine 두 경로 조각을 단일 경로로 결합합니다. (PathCchCombine)
PathCchCombineEx 두 경로 조각을 단일 경로로 결합합니다. (PathCchCombineEx)
PathCchFindExtension

경로를 검색하여 파일 이름 확장명(예: ".exe" 또는 ".ini")을 찾습니다.
PathCchIsRoot 경로 문자열이 볼륨의 루트를 참조하는지 여부를 결정합니다. 이 함수는 "\", "\?#0034; 경로를 허용한다는 점에서 PathIsRoot와 다릅니다. 및 "\?\UNC" 접두사.
PathCchRemoveBackslash 경로 문자열의 끝에서 후행 백슬래시를 제거합니다. 이 함수는 "\", "\?" 경로를 허용한다는 점에서 PathRemoveBackslash와 다릅니다. 및 "\?\UNC" 접두사.
PathCchRemoveBackslashEx 경로 문자열의 끝에서 후행 백슬래시를 제거합니다. 이 함수는 문자열의 새 끝에 대한 포인터를 반환하고 버퍼에 남아 있는 사용되지 않은 문자 수를 보고할 수 있다는 점에서 PathCchRemoveBackslash와 다릅니다. 이 함수는 "\", "\?" 경로를 허용한다는 점에서 PathRemoveBackslash와 다릅니다. 및 "\?\UNC" 접두사.
PathCchRemoveExtension 경로에서 파일 이름 확장명을 제거합니다(있는 경우). 이 함수는 "\", "\?" 경로를 허용한다는 점에서 PathRemoveExtension과 다릅니다. 및 "\?\UNC" 접두사.
PathCchRemoveFileSpec 해당 요소가 파일 이름인지 디렉터리 이름인지 여부에 관계없이 경로 문자열의 마지막 요소를 제거합니다.
PathCchRenameExtension 경로 문자열의 끝에 있는 파일 이름의 확장명을 새 확장명으로 바꿉니다.
PathCchSkipRoot 드라이브 문자 또는 UNC(유니버설 명명 규칙) 서버/공유 경로 요소 다음에 있는 경로의 첫 번째 문자에 대한 포인터를 검색합니다. 이 함수는 "\", "\?" 경로를 허용한다는 점에서 PathSkipRoot와 다릅니다. 및 "\?\UNC" 접두사.
PathCchStripPrefix "\?"를 제거합니다. 파일 경로의 접두사(있는 경우)입니다.
PathCchStripToRoot 루트 정보를 제외한 경로의 모든 파일 및 디렉터리 요소를 제거합니다. 이 함수는 "\", "\?" 경로를 허용한다는 점에서 PathStripToRoot와 다릅니다. 및 "\?\UNC" 접두사.
PathCleanupSpec PathCleanupSpec을 변경하거나 사용할 수 없습니다.

PathCombineA 올바르게 구성된 경로를 나타내는 두 문자열을 하나의 경로로 연결합니다. 또한 상대 경로 요소를 연결합니다. (ANSI)
PathCombineW 올바르게 구성된 경로를 나타내는 두 문자열을 하나의 경로로 연결합니다. 또한 상대 경로 요소를 연결합니다. (유니코드)
PathCommonPrefixA 두 경로를 비교하여 공통 접두사를 공유하는지 확인합니다. 접두사는 다음 형식 중 하나입니다. _"C:\", ".", "..", ".. \". (ANSI)
PathCommonPrefixW 두 경로를 비교하여 공통 접두사를 공유하는지 확인합니다. 접두사는 다음 형식 중 하나입니다. _"C:\", ".", "..", ".. \". (유니코드)
PathCompactPathA 경로 구성 요소를 줄임표로 바꿔 지정된 픽셀 너비 내에 맞게 파일 경로를 자립니다. (ANSI)
PathCompactPathExA 경로 구성 요소를 줄임표로 바꿔 특정 문자 수 내에 맞게 경로를 자립니다. (ANSI)
PathCompactPathExW 경로 구성 요소를 줄임표로 바꿔 특정 문자 수 내에 맞게 경로를 자립니다. (유니코드)
PathCompactPathW 경로 구성 요소를 줄임표로 바꿔 지정된 픽셀 너비 내에 맞게 파일 경로를 자립니다. (유니코드)
PathCreateFromUrlA 파일 URL을 Microsoft MS-DOS 경로로 변환합니다. (ANSI)
PathCreateFromUrlAlloc 파일 URL에서 경로를 만듭니다.
PathCreateFromUrlW 파일 URL을 Microsoft MS-DOS 경로로 변환합니다. (유니코드)

PathFileExistsA 파일 또는 폴더와 같은 파일 시스템 개체의 경로가 유효한지 여부를 확인합니다. (ANSI)
PathFileExistsW 파일 또는 폴더와 같은 파일 시스템 개체의 경로가 유효한지 여부를 확인합니다. (유니코드)
PathFindExtensionA 확장에 대한 경로를 검색합니다. (ANSI)
PathFindExtensionW 확장에 대한 경로를 검색합니다. (유니코드)
PathFindFileNameA 파일 이름을 검색합니다. (ANSI)
PathFindFileNameW 파일 이름을 검색합니다. (유니코드)
PathFindNextComponentA 경로를 구문 분석하고 첫 번째 백슬래시 뒤에 있는 해당 경로의 부분을 반환합니다. (ANSI)
PathFindNextComponentW 경로를 구문 분석하고 첫 번째 백슬래시 뒤에 있는 해당 경로의 부분을 반환합니다. (유니코드)
PathFindOnPathA 파일을 검색합니다. (ANSI)
PathFindOnPathW 파일을 검색합니다. (유니코드)
PathFindSuffixArrayA 지정된 파일 이름에 접미사 목록 중 하나가 있는지 여부를 확인합니다. (ANSI)
PathFindSuffixArrayW 지정된 파일 이름에 접미사 목록 중 하나가 있는지 여부를 확인합니다. (유니코드)
PathGetArgsA

지정된 경로 내에서 명령줄 인수를 찾습니다. (ANSI)
PathGetArgsW 지정된 경로 내에서 명령줄 인수를 찾습니다. (유니코드)
PathGetCharTypeA 경로와 관련하여 문자 형식을 결정합니다. (ANSI)
PathGetCharTypeW 경로와 관련하여 문자 형식을 결정합니다. (유니코드)
PathGetDriveNumberA 'A'에서 'Z' 범위 내의 드라이브 문자 경로를 검색하고 해당 드라이브 번호를 반환합니다. (ANSI)
PathGetDriveNumberW 'A'에서 'Z' 범위 내의 드라이브 문자 경로를 검색하고 해당 드라이브 번호를 반환합니다. (유니코드)
PathGetShortPath PathGetShortPath를 변경하거나 사용할 수 없습니다.
PathIsContentTypeA 파일의 등록된 콘텐츠 형식이 지정된 콘텐츠 형식과 일치하는지 여부를 확인합니다. 이 함수는 지정된 파일 형식의 콘텐츠 형식을 가져오고 해당 문자열을 pszContentType과 비교합니다. 비교는 대/소문자를 구분하지 않습니다. (ANSI)
PathIsContentTypeW 파일의 등록된 콘텐츠 형식이 지정된 콘텐츠 형식과 일치하는지 여부를 확인합니다. 이 함수는 지정된 파일 형식의 콘텐츠 형식을 가져오고 해당 문자열을 pszContentType과 비교합니다. 비교는 대/소문자를 구분하지 않습니다. (유니코드)
PathIsDirectoryA 경로가 유효한 디렉터리인지 확인합니다. (ANSI)
PathIsDirectoryEmptyA 지정된 경로가 빈 디렉터리인지 여부를 확인합니다. (ANSI)
PathIsDirectoryEmptyW 지정된 경로가 빈 디렉터리인지 여부를 확인합니다. (유니코드)

PathIsDirectoryW 경로가 유효한 디렉터리인지 확인합니다. (유니코드)
PathIsExe PathIsExe를 변경하거나 사용할 수 없습니다.
PathIsFileSpecA 경로 구분 문자(예: ':' 또는 '')를 검색합니다. 경로 구분 문자가 없는 경우 경로는 파일 사양 경로로 간주됩니다.
PathIsFileSpecW 경로 구분 문자(예: ':' 또는 '\')를 검색합니다. 경로 구분 문자가 없는 경우 경로는 파일 사양 경로로 간주됩니다.
PathIsHTMLFileA 파일이 HTML 파일인지 여부를 확인합니다. 파일의 확장명용으로 등록된 콘텐츠 형식에 따라 결정됩니다. (ANSI)
PathIsHTMLFileW 파일이 HTML 파일인지 여부를 확인합니다. 파일의 확장명용으로 등록된 콘텐츠 형식에 따라 결정됩니다. (유니코드)
PathIsLFNFileSpecA 파일 이름이 긴 형식인지 여부를 확인합니다. (ANSI)
PathIsLFNFileSpecW 파일 이름이 긴 형식인지 여부를 확인합니다. (유니코드)
PathIsNetworkPathA 경로 문자열이 네트워크 리소스를 나타내는지 여부를 결정합니다. (ANSI)
PathIsNetworkPathW 경로 문자열이 네트워크 리소스를 나타내는지 여부를 결정합니다. (유니코드)
PathIsPrefixA 경로를 검사하여 pszPrefix에서 전달된 접두사 값으로 시작하는지 확인합니다. (ANSI)

PathIsPrefixW 경로를 검사하여 pszPrefix에서 전달된 접두사 값으로 시작하는지 확인합니다. (유니코드)
PathIsRelativeA 경로를 검색하고 상대 경로인지 확인합니다. (ANSI)
PathIsRelativeW 경로를 검색하고 상대 경로인지 확인합니다. (유니코드)
PathIsRootA 경로 문자열이 볼륨의 루트를 참조하는지 여부를 결정합니다. (ANSI)
PathIsRootW 경로 문자열이 볼륨의 루트를 참조하는지 여부를 결정합니다. (유니코드)
PathIsSameRootA 두 경로를 비교하여 공통 루트 구성 요소가 있는지 확인합니다. (ANSI)
PathIsSameRootW 두 경로를 비교하여 공통 루트 구성 요소가 있는지 확인합니다. (유니코드)
PathIsSlowA PathIsSlow를 변경하거나 사용할 수 없습니다. (ANSI)
PathIsSlowW PathIsSlow를 변경하거나 사용할 수 없습니다. (유니코드)
PathIsSystemFolderA 기존 폴더에 시스템 폴더로 만드는 특성이 포함되어 있는지 확인합니다. 또는 이 함수는 특정 특성이 폴더를 시스템 폴더로 한정하는지를 나타냅니다. (ANSI)
PathIsSystemFolderW 기존 폴더에 시스템 폴더로 만드는 특성이 포함되어 있는지 확인합니다. 또는 이 함수는 특정 특성이 폴더를 시스템 폴더로 한정하는지를 나타냅니다. (유니코드)
PathIsUNCA

드라이브 문자를 기반으로 하는 경로가 아니라 경로 문자열이 유효한 UNC(범용 명명 규칙) 경로인지 여부를 확인합니다. (ANSI)
PathIsUNCEx 드라이브 문자를 기반으로 하는 경로가 아니라 경로 문자열이 유효한 UNC(범용 명명 규칙) 경로인지 여부를 확인합니다. 이 함수는 경로에서 서버 이름을 추출할 수도 있다는 점에서 PathIsUNC와 다릅니다.
PathIsUNCServerA 문자열이 서버 경로에 대해서만 유효한 UNC(범용 명명 규칙)인지 확인합니다. (ANSI)
PathIsUNCServerShareA 문자열이 유효한 UNC(유니버설 명명 규칙) 공유 경로인 \server\share인지 확인합니다. (ANSI)
PathIsUNCServerShareW 문자열이 유효한 UNC(유니버설 명명 규칙) 공유 경로인 \server\share인지 확인합니다. (유니코드)
PathIsUNCServerW 문자열이 서버 경로에 대해서만 유효한 UNC(범용 명명 규칙)인지 확인합니다. (유니코드)
PathIsUNCW 드라이브 문자를 기반으로 하는 경로가 아니라 경로 문자열이 유효한 UNC(범용 명명 규칙) 경로인지 여부를 확인합니다. (유니코드)
PathIsURLA 지정된 문자열을 테스트하여 유효한 URL 형식을 준수하는지 확인합니다. (ANSI)
PathIsURLW 지정된 문자열을 테스트하여 유효한 URL 형식을 준수하는지 확인합니다. (유니코드)
PathMakePrettyA 모든 대문자 경로를 모든 소문자로 변환하여 경로에 일관된 모양을 지정합니다. (ANSI)
PathMakePrettyW 모든 대문자 경로를 모든 소문자로 변환하여 경로에 일관된 모양을 지정합니다. (유니코드)
PathMakeSystemFolderA 기존 폴더에 시스템 폴더가 될 적절한 특성을 제공합니다. (ANSI)

PathMakeSystemFolderW 기존 폴더에 시스템 폴더가 될 적절한 특성을 제공합니다. (유니코드)
PathMakeUniqueName 템플릿에서 고유한 경로 이름을 만듭니다.
PathMatchSpecA Microsoft MS-DOS 와일드카드 일치 형식을 사용하여 문자열을 검색합니다. (ANSI)
PathMatchSpecExA 경로의 파일 이름을 하나 이상의 파일 이름 패턴과 일치합니다. (ANSI)
PathMatchSpecExW 경로의 파일 이름을 하나 이상의 파일 이름 패턴과 일치합니다. (유니코드)
PathMatchSpecW Microsoft MS-DOS 와일드카드 일치 형식을 사용하여 문자열을 검색합니다. (유니코드)
PathParseIconLocationA 파일 위치 및 아이콘 인덱스가 포함된 파일 위치 문자열을 구문 분석하고 별도의 값을 반환합니다. (ANSI)
PathParseIconLocationW 파일 위치 및 아이콘 인덱스가 포함된 파일 위치 문자열을 구문 분석하고 별도의 값을 반환합니다. (유니코드)
PathProcessCommand Deprecated. 명령줄이 포함된 문자열을 처리하고 필요한 경우 인수가 연결된 적절한 따옴표 붙은 문자열을 생성합니다.
PathQuoteSpacesA 공백의 경로를 검색합니다. 공백이 있으면 전체 경로가 따옴표로 묶입니다. (ANSI)
PathQuoteSpacesW 공백의 경로를 검색합니다. 공백이 있으면 전체 경로가 따옴표로 묶입니다. (유니코드)
PathRelativePathToA 한 파일 또는 폴더에서 다른 파일로의 상대 경로를 만듭니다. (ANSI)

PathRelativePathToW 한 파일 또는 폴더에서 다른 파일로의 상대 경로를 만듭니다. (유니코드)
PathRemoveArgsA 지정된 경로에서 인수를 제거합니다. (ANSI)
PathRemoveArgsW 지정된 경로에서 인수를 제거합니다. (유니코드)
PathRemoveBackslashA 지정된 경로에서 후행 백슬래시를 제거합니다. (ANSI)
PathRemoveBackslashW 지정된 경로에서 후행 백슬래시를 제거합니다. (유니코드)
PathRemoveBlanksA 문자열에서 선행 및 후행 공백을 모두 제거합니다. (ANSI)
PathRemoveBlanksW 문자열에서 선행 및 후행 공백을 모두 제거합니다. (유니코드)
PathRemoveExtensionA 경로에서 파일 이름 확장명을 제거합니다(있는 경우). (ANSI)
PathRemoveExtensionW 경로에서 파일 이름 확장명을 제거합니다(있는 경우). (유니코드)
PathRemoveFileSpecA 경로에 있는 경우 후행 파일 이름 및 백슬래시를 제거합니다. (ANSI)
PathRemoveFileSpecW 경로에 있는 경우 후행 파일 이름 및 백슬래시를 제거합니다. (유니코드)
PathRenameExtensionA 파일 이름의 확장명을 새 확장명으로 바꿉니다. 파일 이름에 확장명을 포함하지 않으면 확장명은 문자열의 끝에 연결됩니다. (ANSI)

PathRenameExtensionW 파일 이름의 확장명을 새 확장명으로 바꿉니다. 파일 이름에 확장명을 포함하지 않으면 확장명은 문자열의 끝에 연결됩니다. (유니코드)
PathResolve PathResolve를 변경하거나 사용할 수 없습니다.
PathSearchAndQualifyA 지정된 경로의 형식이 올바르게 지정되고 정규화되었는지 여부를 확인합니다. (ANSI)
PathSearchAndQualifyW 지정된 경로의 형식이 올바르게 지정되고 정규화되었는지 여부를 확인합니다. (유니코드)
PathSetDlgItemPathA PathCompactPath를 사용하여 경로가 컨트롤에 맞는지 확인하여 창 또는 대화 상자에서 자식 컨트롤의 텍스트를 설정합니다. (ANSI)
PathSetDlgItemPathW PathCompactPath를 사용하여 경로가 컨트롤에 맞는지 확인하여 창 또는 대화 상자에서 자식 컨트롤의 텍스트를 설정합니다. (유니코드)
PathSkipRootA 드라이브 문자 또는 UNC(유니버설 명명 규칙) 서버/공유 경로 요소 다음에 있는 경로의 첫 번째 문자에 대한 포인터를 검색합니다. (ANSI)
PathSkipRootW 드라이브 문자 또는 UNC(유니버설 명명 규칙) 서버/공유 경로 요소 다음에 있는 경로의 첫 번째 문자에 대한 포인터를 검색합니다. (유니코드)
PathStripPathA 정규화된 경로 및 파일의 경로 부분을 제거합니다. (ANSI)
PathStripPathW 정규화된 경로 및 파일의 경로 부분을 제거합니다. (유니코드)
PathStripToRootA 루트 정보를 제외한 경로의 모든 파일 및 디렉터리 요소를 제거합니다. (ANSI)

PathStripToRootW 루트 정보를 제외한 경로의 모든 파일 및 디렉터리 요소를 제거합니다. (유니코드)
PathUndecorateA 경로 문자열에서 장식을 제거합니다. (ANSI)
PathUndecorateW 경로 문자열에서 장식을 제거합니다. (유니코드)
PathUnExpandEnvStringsA 정규화된 경로의 특정 폴더 이름을 연결된 환경 문자열로 바꿉니다. (ANSI)
PathUnExpandEnvStringsW 정규화된 경로의 특정 폴더 이름을 연결된 환경 문자열로 바꿉니다. (유니코드)
PathUnmakeSystemFolderA 시스템 폴더로 만드는 폴더에서 특성을 제거합니다. 이 폴더는 실제로 파일 시스템에 있어야 합니다. (ANSI)
PathUnmakeSystemFolderW 시스템 폴더로 만드는 폴더에서 특성을 제거합니다. 이 폴더는 실제로 파일 시스템에 있어야 합니다. (유니코드)
PathUnquoteSpacesA 경로의 시작과 끝에서 따옴표를 제거합니다. (ANSI)
PathUnquoteSpacesW 경로의 시작과 끝에서 따옴표를 제거합니다. (유니코드)
PathYetAnotherMakeUniqueName 기존 파일 이름을 기반으로 고유한 파일 이름을 만듭니다.
PauseTimer 지원되지 않습니다. (IFileOperationProgressSink.PauseTimer)
PauseTimer 진행률 대화 상자 타이머를 일시 중지합니다.

PerformOperations 선택한 모든 작업을 실행합니다.
PerUserInit 내 문서 및 기타 특수 폴더를 만들고, 필요에 따라 초기화하고, 내 문서에 대한 보내기 바로 가기 메뉴 항목을 만듭니다.
PickIconDlg PickIconDlg를 변경하거나 사용할 수 없습니다.
PlaySound 알림과 함께 소리를 재생합니다. (IUserNotification2.PlaySound)
PlaySound 알림과 함께 소리를 재생합니다. (IUserNotification.PlaySound)
팝업 지정된 화면 위치에서 바로 가기 메뉴를 호출합니다.
팝업 특정 위치에 모달 팝업 메뉴를 표시합니다.
PopupItemMenu 지정된 항목에 대한 바로 가기 메뉴를 만들고 선택한 명령 문자열을 반환합니다.
PostCopyItem 각 항목에 대한 복사 프로세스가 완료된 후 호출자 구현 작업을 수행합니다.
PostDeleteItem 각 항목에 대한 삭제 프로세스가 완료된 후 호출자 구현 작업을 수행합니다.
PostMoveItem 각 항목에 대한 이동 프로세스가 완료된 후 호출자 구현 작업을 수행합니다.
PostNewItem 새 항목을 만든 후 호출자 구현 작업을 수행합니다.

PostPaint 네임스페이스 트리 컨트롤을 그린 후 호출됩니다.
PostRenameItem 각 항목에 대한 이름 바꾸기 프로세스가 완료된 후 호출자 구현 작업을 수행합니다.
PreCopyItem 각 항목에 대한 복사 프로세스가 시작되기 전에 호출자 구현 작업을 수행합니다.
PreDeleteItem 각 항목에 대한 삭제 프로세스가 시작되기 전에 호출자 구현 작업을 수행합니다.
PreMoveItem 각 항목에 대한 이동 프로세스가 시작되기 전에 호출자 구현 작업을 수행합니다.
PreNewItem 새 항목을 만드는 프로세스가 시작되기 전에 호출자 구현 작업을 수행합니다.
프리페인트 네임스페이스 트리 컨트롤을 그리기 전에 호출됩니다.
PrepareForSync 등록된 애플리케이션에서 사용자 인터페이스를 표시하고 ISyncMgrSynchronize::Synchronize 메서드가 호출되기 전에 필요한 초기화를 수행할 수 있습니다.
PrepareForSyncCompleted PrepareForSync 메서드가 완료된 후 애플리케이션의 등록된 처리기에서 호출됩니다.
PreRenameItem 각 항목의 이름 바꾸기 프로세스가 시작되기 전에 호출자 구현 작업을 수행합니다.
PresentConflict 사용자에게 충돌을 표시합니다.
PrevPage 다중 페이지 이미지의 이전 페이지로 전환합니다. 연결된 모든 애니메이션이 다시 설정됩니다.

PrintTo 파일을 인쇄합니다. (ANSI)
PrintTo 파일을 인쇄합니다. (유니코드)
ProcessAttachDetach Deprecated. 항상 S_OK 반환합니다.
ProfferService 동일한 호스트의 다른 개체에서 서비스를 사용할 수 있도록 합니다.
진행률 등록된 애플리케이션에서 호출하여 진행률 정보를 업데이트하고 작업을 계속할지 여부를 결정합니다.
프롬프트 사용자에게 프롬프트 UI를 표시합니다.
PropertyFailure 파일 속성과 사용자 상호 작용이 필요한 오류가 있을 때 호출됩니다.
ProposeItem 이전에 열거된 항목 집합에 새 항목을 추가할 것을 제안합니다.
PtrdiffTAdd ptrdiff_t 형식의 두 값을 추가합니다.
PtrdiffTMult ptrdiff_t 형식의 두 값을 곱합니다.
PtrdiffTSub ptrdiff_t 형식의 한 값을 다른 값에서 뺍니다.
put_CurrentViewMode 현재 폴더의 현재 보기 모드를 설정합니다.
put_FolderFlags

현재 폴더 설정을 설정합니다.
put_GroupBy 폴더 뷰를 그룹화할 때 사용되는 열을 설정합니다.
put_IconSize 현재 폴더의 아이콘 크기 설정을 설정합니다.
put_SortColumns 정렬할 열의 이름을 설정합니다.
PutBaseBrowserData Deprecated. 기본 클래스의 보호된 멤버에 대한 읽기/쓰기 액세스를 허용하는 구조를 가져옵니다. 그러나 해당 상태는 기본 브라우저에서만 업데이트해야 합니다.
QISearch IUnknown::QueryInterface 메서드의 테이블 기반 구현입니다.
QueryAbilities 충돌 발표자가 표시할 옵션을 결정합니다.
QueryAbort 디코딩, 그리기 또는 크기 조정과 같은 IShellImageData 프로세스를 중단합니다. 콜백 메서드입니다.
QueryActiveShellView 현재 활성(표시된) Shell 뷰 개체를 검색합니다.
QueryAppIsDefault 애플리케이션이 지정된 애플리케이션 수준 및 형식에 대해 등록된 기본 연결을 소유하는지 여부를 결정합니다. Windows 8에서는 사용할 수 없습니다.
QueryAppIsDefaultAll 애플리케이션이 지정된 애플리케이션 수준에 대해 등록된 모든 기본 연결을 소유하는지 여부를 결정합니다. Windows 8에서는 사용할 수 없습니다.
QueryBand 밴드 사이트의 밴드에 대한 정보를 가져옵니다.

QueryCancel 작업이 취소되는지 여부에 대한 정보를 제공합니다.
QueryContextMenu 바로 가기 메뉴에 명령을 추가합니다.
QueryContinue 이 인터페이스의 현재 인스턴스와 연결된 작업이 계속되어야 하는 경우 S_OK 반환합니다.
QueryCurrentDefault 지정된 연결 형식에 대한 기본 애플리케이션을 결정합니다. 이 애플리케이션은 해당 형식에 대해 ShellExecute에서 시작한 기본 애플리케이션입니다.
QueryFocus GetFocus 함수를 호출하지 않도록 HWND를 반환하도록 미리 보기 처리기를 지시합니다.
QueryForAdditionalItems 동기화할 보류 중인 요청이 있는 항목 집합의 열거자를 검색합니다. 현재 동기화가 완료된 후 동기화되는 항목 집합입니다.
QueryObject 지정된 개체에 대한 쿼리입니다.
QuerySupport QuerySupport를 변경하거나 사용할 수 없습니다.
ReadAsync 스트림에서 비동기적으로 정보를 읽습니다. 예를 들어 셸은 비동기적으로 전송할 때 파일 항목에 이 인터페이스를 구현합니다.
ReadCabinetState ReadCabinetState를 변경하거나 사용할 수 없습니다.
ReadPropertyNPB 명명된 속성 모음에서 속성을 읽도록 합니다.
RealDriveType

RealDriveType을 변경하거나 사용할 수 없습니다.
재정렬 다시 정렬은 변경되거나 사용할 수 없습니다.
RecompressImage 이미지를 다시 압축합니다. ImageRecompress 개체에서 구현된 이 메서드는 품질 지정을 사용하여 x 및 y 차원을 허용합니다. 메서드는 지정된 크기로 다시 압축된 새 이미지를 포함하는 스트림을 만듭니다.
RecycleItem 항목을 제공된 재활용 위치로 재활용하고 새 위치에 항목을 반환합니다.
Redirect 일반 및 사용자별 폴더에 대한 폴더 요청을 리디렉션합니다.
새로 고침 사용자 입력에 대한 응답으로 보기의 콘텐츠를 새로 고칩니다.
RefreshObject RefreshObject를 변경하거나 사용할 수 없습니다.
RefreshOverlayImages 오버레이 캐시, 이미지 목록 또는 둘 다를 새로 고칩니다.
등록기 열린 창을 셸 창으로 등록합니다. 창은 핸들로 지정됩니다.
RegisterAbort 필요에 따라 이전 개체에 대한 포인터를 반환하여 콜백 중단 개체를 설정합니다.
RegisterAppStateChangeNotification 앱이 콜백 함수를 등록할 수 있도록 하며, 이를 통해 라이브러리가 일시 중단된 상태로 전환되거나 중단되었음을 알 수 있습니다.
RegisterAsChild ISuspensionDependencyManager::RegisterAsChild 메서드
RegisterDialogClasses

화면 보호기 구성 대화 상자에 필요한 비표준 창 클래스를 등록합니다.
RegisterFolder 레지스트리에 알려진 새 폴더를 추가합니다. 알려진 폴더 시스템에 자신의 폴더 중 하나를 추가하는 ISV(독립 소프트웨어 공급업체)에서 특히 사용됩니다.
RegisterForPackageStateChanges 패키지 상태 변경 알림을 등록합니다.
RegisterPending 보류 중인 창을 셸 창으로 등록합니다. 창은 절대 PIDL로 지정됩니다.
RegisterScaleChangeEvent 크기 조정이 변경되었을 때 트리거되는 이벤트를 등록합니다. 이 함수는 RegisterScaleChangeNotifications를 대체합니다.
RegisterScaleChangeNotifications 크기 조정 정보가 변경될 때 콜백을 받을 창을 등록합니다.
RegisterSyncMgrHandler 처리기에 동기화할 항목이 있는 경우 동기화 관리자에 처리기를 등록합니다.
RegisterTab 애플리케이션의 작업 표시줄 그룹 플라이아웃에 표시할 새 탭 또는 문서 축소판 그림이 제공되었음을 작업 표시줄에 알립니다.
RegisterWindow Deprecated. 브라우저 창 목록에 브라우저를 등록합니다.
ReleaseShellView Deprecated. 기본 클래스와 해당 파생 클래스 간의 뷰 수명을 조정합니다.
Remove 관리되는 개체 컬렉션에서 개체를 제거합니다.
removeAll 결과 폴더에서 모든 항목을 제거합니다.

removeAll 결과 폴더에서 모든 항목을 제거합니다.
RemoveAllControlItems 구현되지 않았습니다. (IFileDialogCustomize.RemoveAllControlItems)
RemoveAllDestinations 애플리케이션의 점프 목록의 최근 및 자주 사용하는 범주에서 모든 대상 항목을 지웁니다.
RemoveAllRoots 트리에서 모든 루트와 자식을 제거합니다.
RemoveBackReferences 개체가 보유한 모든 뒤로 참조를 제거합니다.
RemoveBand 밴드 사이트에서 밴드를 제거합니다.
RemoveConflicts 충돌 ID로 지정된 충돌 집합을 저장소에서 삭제합니다.
RemoveControlItem 대화 상자의 컨테이너 컨트롤에서 항목을 제거합니다.
RemoveDataBlock 링크에서 데이터 블록을 제거합니다.
RemoveDestination 점프 목록의 최근 및 자주 사용하는 범주에서 단일 대상을 제거합니다.
RemoveEvent 지정된 대로 이벤트를 제거합니다.
RemoveFolder 라이브러리에서 폴더를 제거합니다.
RemoveFromList

Windows Vista:_Removes 시작 메뉴의 왼쪽 위 위치에 있는 목록인 시작 메뉴 고정 목록의 항목입니다.
RemoveFromSyncSet 충돌을 삭제하고 동기화에서 IShellItem을 제거합니다.
RemoveIDList 결과 폴더에서 PIDL(항목 식별자 목록)에 대한 포인터를 제거합니다.
RemoveItem 결과 폴더에서 항목을 제거합니다.
RemoveItem 항목을 휴지통으로 이동하지 않고 항목을 제거합니다.
RemoveMenusSB 컨테이너가 현재 위치 복합 메뉴에서 해당 메뉴 요소를 제거하고 연결된 모든 리소스를 해제할 수 있도록 허용합니다.
RemoveObject RemoveObject를 변경하거나 사용할 수 없습니다.
RemoveObjectAt 컬렉션에서 지정된 단일 개체를 제거합니다.
RemovePropertyNPB 명명된 속성 모음에서 속성을 제거합니다.
RemoveRoot 트리에서 루트 및 해당 자식을 제거합니다.
RemoveTasks 스케줄러의 백그라운드 큐에서 작업을 제거합니다.
RemoveToolbar 도구 모음 프레임에서 지정된 IDockingWindow를 제거합니다.
RemoveWindowSubclass 창에서 하위 클래스 콜백을 제거합니다.

RenameItem 새 표시 이름을 부여할 단일 항목을 선언합니다.
RenameItem 새 이름으로 IShellItem을 반환하여 항목의 이름을 변경합니다.
RenameItems 새 표시 이름을 부여할 항목 집합을 선언합니다. 모든 항목에는 동일한 이름이 지정됩니다.
ReplaceFrame 현재 프레임을 새 이미지로 바꿉니다.
ReplacePage 제어판 개체의 속성 시트에 있는 페이지를 대체합니다.
ReportEvent 동기화 중인 항목에 대한 동기화 결과 폴더에 추가할 이벤트를 제공합니다.
ReportManualSync 동기화 센터 UI 외부에서 수동으로 요청된 동기화 작업이 수행되고 있다고 보고합니다.
ReportProgress 동기화 센터에 단일 동기화 항목의 동기화 진행률을 보고합니다.
ReportResult 수신된 오류 상태 코드를 사용자가 읽을 수 있는 적절한 메시지로 변환합니다.
RequestBorderSpaceDW IDockingWindow 개체의 테두리 공간에 대한 요청을 승인, 수정 또는 거부합니다. SetBorderSpaceDW 메서드가 호출될 때까지 테두리 공간이 할당되지 않습니다.
RequireTouchInEditControl 터치 키보드가 호출되기 전에 편집 필드에서 명시적 사용자 탭이 필요합니다.
재설정 열거형을 0으로 다시 설정합니다.

재설정 열거형 인덱스(0)를 다시 설정하는 데 사용됩니다.
재설정 열거형 시퀀스의 시작 부분에 열거자를 반환합니다.
재설정 열거형 시퀀스의 시작 부분으로 돌아갑니다.
재설정 열거형 인덱스를 0으로 다시 설정합니다. (IEnumObjects.Reset)
재설정 열거형 인덱스를 0으로 다시 설정합니다. (IEnumResources.Reset)
재설정 열거형에서 검색된 IShellItem 인터페이스의 내부 수를 다시 설정합니다.
재설정 IPublishedApp 개체의 열거형을 첫 번째 항목으로 다시 설정합니다.
재설정 열거형의 현재 위치를 0으로 다시 설정합니다.
재설정 열거형의 현재 위치를 0으로 다시 설정합니다.
재설정 열거형의 현재 위치를 0으로 다시 설정합니다.
재설정 요소 목록의 시작 부분에 자신을 배치하도록 열거자를 지시합니다.
ResetCancel 취소가 완료된 후 진행률 대화 상자를 다시 설정합니다.
ResetEnumerator

목록이 표시되면 자동 완성 개체가 제안 목록을 새로 고치도록 합니다.
ResetTimer 지원되지 않습니다. (IFileOperationProgressSink.ResetTimer)
ResetTimer 진행률 대화 상자 타이머를 0으로 다시 설정합니다.
ResizeBorderDW 도킹 창 개체에 프레임의 테두리 공간이 변경되었음을 알립니다. 이 메서드에 대한 응답으로 테두리 공간이 없거나 변경이 필요하지 않은 경우에도 IDockingWindow 구현은 SetBorderSpaceDW를 호출해야 합니다.
해결하다 셸 링크가 이동되거나 이름이 바뀐 경우에도 셸 링크의 대상을 찾으려고 시도합니다. (ANSI)
해결하다 셸 링크가 이동되거나 이름이 바뀐 경우에도 셸 링크의 대상을 찾으려고 시도합니다. (유니코드)
해결하다 자체 동기화 처리기를 사용하여 충돌을 해결하고 UI를 제어합니다.
ResolveFolder 폴더가 이동되거나 이름이 바뀐 경우에도 라이브러리 폴더의 대상 위치를 확인합니다.
ResolveShellLink 폴더 개체가 Shell 링크를 확인하라는 요청입니다.
RestartDialog 사용자에게 Windows를 다시 시작하라는 메시지를 표시하는 대화 상자를 표시합니다. 사용자가 단추를 클릭하면 함수는 ExitWindowsEx를 호출하여 Windows를 다시 시작하려고 시도합니다.
RestartDialogEx 사용자에게 Windows를 다시 시작하도록 요청하는 대화 상자를 표시합니다. 사용자가 단추를 클릭하면 함수는 ExitWindowsEx를 호출하여 Windows를 다시 시작하려고 시도합니다.
결과 StartHandlerSync 또는 StartItemSync 호출의 결과를 가져옵니다.

Resume 현재 일시 중단된 경우 패키지의 프로세스를 다시 시작합니다.
Resume 작업이 다시 시작되도록 요청합니다.
ResumeTimer 지원되지 않습니다. (IFileOperationProgressSink.ResumeTimer)
ResumeTimer 진행률 대화 상자 타이머를 다시 시작합니다.
RetrieveProperties 동기화 엔진에서 관리하는 속성을 가져옵니다.
되돌릴 Deprecated. 탐색에 실패한 경우 ITravelLog::AddEntry의 결과를 삭제하여 현재 항목으로 되돌려 줍니다.
취소 셸 창의 등록을 취소하고 셸 창 컬렉션에서 창을 제거합니다.
RevokeScaleChangeNotifications 창의 등록을 취소하여 정보 크기 조정이 변경될 때 콜백을 수신하지 못하게 합니다.
RevokeService IProfferService::P rofferService를 통해 이전에 다른 개체에서 사용할 수 있었던 서비스를 사용할 수 없게 만듭니다.
돌다 이미지를 90도씩 회전합니다.
실행 지원되지 않습니다. (IShellRunDll.Run)
실행 작업이 시작되도록 요청합니다.

실행 동기화 센터에서 요청하면 처리기 또는 동기화 항목에 대한 UI의 실제 표시를 수행합니다.
RunGadget 바탕 화면에 설치된 가젯을 추가합니다.
저장 첨부 파일을 저장합니다.
저장 라이브러리를 새 라이브러리 설명(*.library-ms) 파일에 저장합니다.
SaveInKnownFolder 라이브러리를 지정된 알려진 폴더의 새 파일에 저장합니다.
SaveProperties 파일 또는 폴더와 연결된 속성을 저장합니다.
SaveViewState 이후 검색 세션 중에 현재 상태를 복원할 수 있도록 셸의 보기 설정을 저장합니다.
SaveViewState Deprecated. (IBrowserService4.SaveViewState)
SaveWithUI 저장 작업이 실패하면 사용자에게 설명 오류 UI를 표시합니다.
Scale 이미지의 크기를 조정합니다.
ScreenSaverConfigureDialog 화면 보호기 구성 대화 상자로 전송된 메시지를 받습니다. 사용자 구성을 허용하는 화면 보호기에서 이 함수를 정의해야 합니다.
ScreenSaverProc 지정된 화면 보호기 창으로 전송된 메시지를 받습니다.

Select IShellFolderView::Select를 변경하거나 사용할 수 없습니다.
SelectAndEditItem INewMenuClient::SelectAndEditItem 메서드
SelectAndPositionItem 셸 뷰에서 항목을 선택하고 배치합니다.
SelectAndPositionItems 폴더 보기에 표시되는 항목의 선택 및 위치를 허용합니다.
SelectedItems 뷰에서 선택한 모든 항목을 나타내는 FolderItems 개체를 가져옵니다.
SelectItem 보기에서 항목의 선택 상태를 설정합니다.
SelectItem 폴더 보기에서 항목을 선택합니다.
SelectItem 셸 보기 창 내에서 하나 이상의 항목의 선택 상태를 변경합니다.
SelectItemRelative 현재 항목을 기준으로 항목을 선택합니다.
SelectPage 다중 페이지 이미지에서 지정된 페이지를 선택합니다.
SendControlMsg Windows 탐색기 창의 도구 모음 또는 상태 표시줄에 컨트롤 메시지를 보냅니다.
SetAcceleratorMenu Deprecated. TranslateAcceleratorSB 호출에 사용할 수 있는 메뉴 가속기를 정의하기 위해 파생 클래스에 의해 구현됩니다.

SetAccelerators 액셀러레이터를 설정합니다.
SetAccessibleName IAccessible::get_accName 개체의 Name 속성을 가져오는 데 사용할 접근성 도구에 의해 검색되는 텍스트를 설정합니다.
SetActivateState Deprecated. 브라우저 보기 창이 활성화된 상태인지 여부를 추적하는 BASEBROWSERDATA 구조체의 _uActivateState 멤버 값을 업데이트합니다. 파생 클래스는 기본 클래스를 호출합니다.
SetActiveAlt 작업 표시줄 항목을 활성으로 표시하지만 시각적으로 활성화하지는 않습니다.
SetAdjacentDisplayEdges 애플리케이션 창이 에뮬레이트된 디스플레이의 가장자리에 인접할지 여부를 설정합니다.
SetAnimation 대화 상자에서 실행되는 Audio-Video AVI(인터리브) 클립을 지정합니다.
SetAppAsDefault 애플리케이션의 게시자가 현재 기본값과 일치하는 경우 애플리케이션을 지정된 확장 또는 프로토콜의 기본값으로 설정합니다. 자세한 내용은 기본 프로그램을 참조하세요. Windows 8에서는 사용할 수 없습니다.
SetAppAsDefaultAll 애플리케이션에 대해 모든 형식의 등록된 모든 연결에 대한 기본값으로 애플리케이션을 설정합니다. Windows 8에서는 사용할 수 없습니다.
SetAppID 작업 표시줄 단추의 점프 목록에서 이 인터페이스의 메서드가 대상을 제거하는 애플리케이션에 대한 고유한 애플리케이션 사용자 모델 ID(AppUserModelID)를 지정합니다. 이 메서드는 선택 사항입니다.
SetAppID 대상 목록을 검색하는 애플리케이션에 대한 고유한 애플리케이션 사용자 모델 ID(AppUserModelID)를 지정합니다. 이 메서드는 선택 사항입니다.
SetAppID 작업 표시줄 단추가 이 인터페이스의 메서드를 통해 빌드된 사용자 지정 점프 목록을 보유하는 애플리케이

션에 대한 고유한 애플리케이션 사용자 모델 ID(AppUserModelID)를 지정합니다. 이 메서드는 선택 사항입니다.
SetAppID 개체를 특정 파일 형식의 처리기로 식별하는 고유한 애플리케이션 정의 애플리케이션 사용자 모델 ID(AppUserModelID)를 지정합니다. 이 메서드는 동적 AppUserModelID가 필요한 애플리케이션에서 사용됩니다.
SetApplicationViewMinWidth 애플리케이션 디자인 모드 창의 원하는 최소 너비를 설정합니다.
SetApplicationViewOrientation 디자인 모드 창에 사용되는 창 방향을 설정합니다.
SetApplicationViewState 디자인 모드에서 실행되는 Windows 스토어 앱에 사용할 스푸핑된 애플리케이션 보기 상태(전체 화면 가로, 전체 화면 세로, 채우기 또는 맞춤)를 설정합니다.
SetArguments Shell 링크 개체의 명령줄 인수를 설정합니다. (ANSI)
SetArguments Shell 링크 개체의 명령줄 인수를 설정합니다. (유니코드)
SetAsDefFolderSettings Deprecated. 폴더의 현재 보기 모드를 모든 폴더의 기본 보기 모드로 설정합니다. 폴더 옵션 대화 상자에서 사용됩니다.
SetAsyncMode 데이터 개체가 비동기 데이터 추출을 지원하는지 여부를 지정하기 위해 드롭 소스에서 호출됩니다.
SetAutomationObject IShellView의 내부 자동화 개체를 바꿉니다.
SetBackgroundColor 이미지가 표시되지 않거나 바탕 화면 배경이 비활성화된 경우 바탕 화면에 표시되는 색을 설정합니다. 이 색은 바탕 화면 배경 화면이 전체 화면을 채우지 않을 때 테두리로도 사용됩니다.
SetBackgroundColor

미리 보기 처리기의 배경색을 설정합니다.
SetBalloonInfo 풍선 알림에 표시할 정보를 설정합니다. (IUserNotification2.SetBalloonInfo)
SetBalloonInfo 풍선 알림에 표시할 정보를 설정합니다. (IUserNotification.SetBalloonInfo)
SetBalloonRetry 첫 번째 시도가 실패할 때 사용자 정보를 표시하려는 조건을 지정합니다. (IUserNotification2.SetBalloonRetry)
SetBalloonRetry 첫 번째 시도가 실패할 때 사용자 정보를 표시하려는 조건을 지정합니다. (IUserNotification.SetBalloonRetry)
SetBandAvailability 지정된 대역의 가용성을 설정합니다.
SetBandInfoSFB BANDINFOSFB 구조체의 정보를 사용하여 IShellFolderBand 개체에 대한 밴드 정보를 설정합니다.
SetBandSiteInfo 밴드 사이트에 대한 정보를 설정합니다.
SetBandState 밴드 사이트에서 밴드 상태를 설정합니다.
SetBorderSpaceDW IDockingWindow 개체에 대한 테두리 공간을 할당하고 예약합니다.
SetCallback IShellFolderView::SetCallback을 변경하거나 사용할 수 없습니다.
SetCancelButtonLabel 일반 파일 대화 상자의 취소 단추에서 기본 텍스트 "취소"를 바꿉니다.
SetCancelInformation

사용자가 진행률 UI에서 취소를 선택할 때 게시되는 정보를 설정합니다.
SetCancelMsg 사용자가 작업을 취소하는 경우 표시할 메시지를 설정합니다.
SetCascade cascade 폴더를 설정합니다.
SetCheckboxValue 로그온 UI 및 자격 증명 UI를 사용하여 확인란 값이 변경되었음을 나타냅니다.
SetCheckButtonState 대화 상자에서 확인 단추(확인란)의 상태를 설정합니다.
SetCheckState 트리 뷰 컨트롤에서 확인란 항목의 상태를 설정합니다.
SetClient punkClient로 지정된 클라이언트를 설정합니다.
SetClientGuid 클라이언트의 GUID를 지정하고 저장합니다.
SetClientGuid 호출 애플리케이션이 GUID를 대화 상자의 지속형 상태와 연결할 수 있도록 합니다.
SetClientTitle 프롬프트 창의 제목을 지정하고 저장합니다.
SetClipboard SetClipboard를 변경하거나 사용할 수 없습니다.
SetCodePage ANSI 코드 페이지의 숫자 값을 지정된 코드 페이지 식별자로 설정합니다.
SetCollectedProperties 저장 대화 상자에서 수집할 속성을 지정합니다.

SetColor 지정된 대로 색을 설정합니다.
SetColumnInfo 지정된 열의 상태를 설정합니다.
SetColumns 표시할 뷰의 열 컬렉션을 설정합니다.
SetComboBoxSelectedValue 로그온 UI 및 자격 증명 UI를 사용하여 콤보 상자 값이 선택되었음을 나타냅니다.
SetCompositionState 컴퍼지션 상태를 설정합니다.
SetCondition 검색의 ICondition을 설정합니다. 이 메서드가 호출되지 않으면 결과 검색에 필터가 적용되지 않습니다.
SetContext 썸네일 공급자가 사용자의 컨텍스트와 관련된 썸네일을 반환할 수 있도록 합니다.
SetContractDelegateWindow 기본 포그라운드 창이 아닌 앱 창을 앱의 계약과 연결합니다. 네이티브 C++에서 Windows 스토어 앱을 작성하는 개발자인 경우 이 함수를 사용합니다.
SetControlItemState 대화 상자에 있는 컨테이너 컨트롤에 있는 항목의 현재 상태를 설정합니다.
SetControlItemText 컨트롤 항목의 텍스트를 설정합니다. 예를 들어 라디오 단추 또는 메뉴의 항목과 함께 제공되는 텍스트입니다.
SetControlLabel 단추 텍스트 또는 편집 상자 레이블과 같이 컨트롤과 연결된 텍스트를 설정합니다.
SetControlState 지정된 컨트롤의 현재 표시 유형 및 사용 상태를 설정합니다.

SetControlStyle 네임스페이스 개체의 트리 뷰 컨트롤에 대한 표시 스타일을 설정합니다.
SetControlStyle2 네임스페이스 개체의 treeview 컨트롤에 대한 확장된 표시 스타일을 설정합니다.
SetCreateFlags CreateProcess 호출에 포함될 플래그를 설정합니다.
SetCurrentFolderFlags 지정된 폴더 플래그를 설정하고 적용합니다.
SetCurrentProcessExplicitAppUserModelID 작업 표시줄에 대한 현재 프로세스를 식별하는 고유한 애플리케이션 정의 애플리케이션 사용자 모델 ID(AppUserModelID)를 지정합니다. 이 식별자를 사용하면 애플리케이션이 단일 작업 표시줄 단추 아래에 연결된 프로세스와 창을 그룹화할 수 있습니다.
SetCurrentViewMode 선택한 폴더의 보기 모드를 설정합니다.
SetDataObject IDataObject 인스턴스를 Windows 런타임 DataPackage로 래핑합니다.
SetDefaultExtension 파일 이름에 추가할 기본 확장명을 설정합니다.
SetDefaultFolder 최근에 사용한 폴더 값이 없는 경우 기본값으로 사용되는 폴더를 설정합니다.
SetDefaultIcon 기본 아이콘을 설정합니다.
SetDefaultSaveFolder 라이브러리에서 저장 작업에 사용할 기본 대상 폴더를 설정합니다.
SetDescription Shell 링크 개체에 대한 설명을 설정합니다. 설명은 애플리케이션 정의 문자열일 수 있습니다. (ANSI)

SetDescription Shell 링크 개체에 대한 설명을 설정합니다. 설명은 애플리케이션 정의 문자열일 수 있습니다. (유니코드)
SetDeselected 자격 증명이 선택 영역을 잃을 때 호출됩니다.
SetDirectory 현재 작업 디렉터리를 설정합니다.
SetDirectory 새 작업 디렉터리를 설정합니다.
SetDisplayName 지정된 대로 검색 폴더 표시 이름을 설정합니다.
SetEditBoxText 대화 상자에 있는 편집 상자 컨트롤의 텍스트를 설정합니다.
SetEmptyText 기본 빈 텍스트를 설정합니다.
SetEncoderParams 인코더 매개 변수를 설정합니다.
SetEnumReadyCallback 초기 뷰 열거가 완료되면 알림을 받을 뷰에 콜백을 설정합니다.
SetEnvironmentVariable 생성된 프로세스의 환경에서 변수를 설정합니다.
SetErrorURL 마법사 확장 페이지를 탐색하는 동안 오류가 발생할 때 표시되는 페이지의 URL을 지정합니다.
SetExtendedTileViewProperties 항목의 확장 타일 속성 목록을 설정합니다.
SetFieldBitmap

타일 이미지 필드가 변경되었으며 UI를 업데이트해야 한다는 로그온 UI 또는 자격 증명 UI와 통신합니다.
SetFieldCheckbox 확인란 필드가 변경되었으며 UI를 업데이트해야 한다는 로그온 UI 또는 자격 증명 UI와 통신합니다.
SetFieldComboBoxSelectedItem 콤보 상자에서 선택한 항목이 변경되었으며 UI를 업데이트해야 한다는 로그온 UI 또는 자격 증명 UI와 통신합니다.
SetFieldInteractiveState 로그온 UI 또는 자격 증명 UI와 통신하여 필드의 대화형 상태가 변경되었으며 UI를 업데이트해야 합니다.
SetFieldOptions 로그온 또는 자격 증명 UI의 지정된 필드에 "암호 표시" 문자 모양이 표시될지 또는 전자 메일 주소를 받을지 여부를 지정합니다.
SetFieldState 로그온 UI 또는 자격 증명 UI와 통신하여 필드 상태가 변경되었으며 UI를 업데이트해야 합니다.
SetFieldString 로그온 UI 또는 자격 증명 UI와 통신하여 필드와 연결된 문자열이 변경되었으며 UI를 업데이트해야 합니다.
SetFieldSubmitButton 자격 증명을 사용하여 제출 단추가 인접하게 표시되는 필드를 설정할 수 있습니다.
SetFileID 현재 파일의 고유 파일 식별자를 설정합니다.
SetFileName 파일의 제안된 이름을 지정하고 저장합니다.
SetFileName 해당 대화 상자가 열릴 때 파일 이름 편집 상자에 표시되는 파일 이름을 설정합니다.
SetFileNameLabel 파일 이름 편집 상자 옆에 있는 레이블의 텍스트를 설정합니다.
SetFileTypeIndex

대화 상자에서 선택한 대로 표시되는 파일 형식을 설정합니다.
SetFileTypes 대화 상자에서 열거나 저장할 수 있는 파일 형식을 설정합니다.
SetFilter SetFilter는 Windows 7을 기준으로 더 이상 사용할 수 없습니다.
SetFilter 클라이언트가 호스트에 IUnknown 인터페이스 포인터를 전달할 수 있도록 호스트에 의해 노출됩니다.
SetFindData 파일 시스템 정보를 WIN32_FIND_DATA 구조에 저장합니다. 이 정보는 ParseDisplayName에서 사용됩니다.
SetFlags IDragSourceHelper 개체에 대한 끌어서 놓기 작업의 특성을 설정합니다.
SetFlags 속성 저장소 개체를 읽기 전용 상태와 읽기/쓰기 상태 간에 전환합니다.
SetFlags GIL_XXX 플래그를 설정합니다.
SetFlags 현재 옵션 설정을 설정합니다.
SetFlags Deprecated. 브라우저 상태를 나타내는 플래그를 설정합니다.
SetFocus 미리 보기 처리기가 포커스를 자체로 설정하도록 지시합니다.
SetFolder 바로 가기 개체의 폴더 이름을 설정합니다. (ANSI)
SetFolder 바로 가기 개체의 폴더 이름을 설정합니다. (유니코드)

SetFolder 이전 사용자 동작에 관계없이 대화 상자가 열릴 때 항상 선택된 폴더를 설정합니다.
SetFolderLogicalViewMode 폴더 논리 뷰 모드를 설정합니다. 기본 설정은 ISearchFolderItemFactory::SetFolderTypeID 메서드에 의해 설정된 FolderTypeID를 기반으로 합니다.
SetFolderSettings 현재 보기의 폴더 설정을 설정합니다.
SetFolderType 라이브러리의 폴더 유형을 설정합니다.
SetFolderTypeID 지정된 대로 검색 폴더 유형 ID를 설정합니다.
SetFolderViewOptions 뷰에 대해 지정된 옵션을 설정합니다.
SetFont 글꼴의 특성을 설정합니다.
SetFont 미리 보기 처리기 내에서 텍스트에 사용할 글꼴 특성을 설정합니다.
SetGroupBy 지정된 속성 키와 방향을 사용하여 뷰를 그룹화합니다.
SetGroupColumn 지정된 대로 그룹 열을 설정합니다. 그룹 열을 지정하지 않으면 그룹화가 발생하지 않습니다.
SetGroupSubsetCount 그룹 하위 집합을 켜고 각 그룹에 표시되는 항목 행 수를 설정합니다.
SetHandlerClsid . (IDefaultFolderMenuInitialize.SetHandlerClsid)

SetHandlerProgressText 처리기가 동기화를 수행하는 동안 처리기에 대한 정보 필드의 콘텐츠를 설정합니다.
SetHistoryObject Deprecated. 브라우저의 기록 개체를 설정합니다.
SetHotkey Shell 링크 개체의 바로 가기 키(핫 키)를 설정합니다. (ANSI)
SetHotkey Shell 링크 개체의 바로 가기 키(핫 키)를 설정합니다. (유니코드)
SetHotKey 애플리케이션의 핫 키를 설정합니다.
SetIcon 라이브러리의 기본 아이콘을 설정합니다.
SetIconInfo 특정 사용자 정보와 연결된 알림 영역 아이콘을 설정합니다. (IUserNotification2.SetIconInfo)
SetIconInfo 특정 사용자 정보와 연결된 알림 영역 아이콘을 설정합니다. (IUserNotification.SetIconInfo)
SetIconLocation Shell 링크 개체의 아이콘 위치(경로 및 인덱스)를 설정합니다. (ANSI)
SetIconLocation Shell 링크 개체의 아이콘 위치(경로 및 인덱스)를 설정합니다. (유니코드)
SetIconSize 지정된 대로 검색 폴더 아이콘 크기를 설정합니다. 기본 설정은 ISearchFolderItemFactory::SetFolderTypeID 메서드에 의해 설정된 FolderTypeID를 기반으로 합니다.
SetIDList 지속형 항목 식별자 목록을 설정합니다.

SetIDList 셸 링크 개체의 PIDL(항목 식별자 목록)에 대한 포인터를 설정합니다. (ANSI)
SetIDList 셸 링크 개체의 PIDL(항목 식별자 목록)에 대한 포인터를 설정합니다. (유니코드)
SetInitialURL 호스트된 마법사에서 초기 서버 제공 HTML 페이지의 URL을 설정합니다.
SetIsOnLockScreen 이 메서드는 디자인 모드에서 애플리케이션이 Windows 8 잠금 화면에 정보를 표시할 수 있는지 여부를 결정합니다.
setItem IParseAndCreateItem::SetItem 메서드
SetItemAlloc 대리자 폴더에 항목 ID를 할당하고 해제하는 데 사용되는 IMalloc 인터페이스를 제공합니다.
SetItemAlloc CItemIDFactory에 항목 ID를 할당하고 해제하는 데 사용되는 IMalloc 인터페이스를 제공합니다.
SetItemChoices 사용자가 유지할 항목을 나타내는 인덱스 배열을 설정합니다. 이 메서드는 사용자가 동일한 처리기에서 동일한 형식의 선택한 모든 충돌에 동일한 작업을 적용하도록 선택할 때 사용됩니다.
SetItemCustomState 셸 항목과 연결된 확인란의 상태를 설정합니다.
SetItemHeight 지정된 항목 높이를 설정합니다.
SetItemPos 지정된 항목의 위치를 설정합니다.
SetItemState 셸 항목에 대한 상태 정보를 설정합니다.

SetItemStatus 다음 두 경우에서 항목의 상태를 변경하기 위해 등록된 애플리케이션의 처리기에서 동기화 관리자가 호출합니다. _between 처리기가 ISyncMgrSynchronize::P repareForSync 메서드에서 반환되고 PrepareForSyncCompleted 콜백 메서드를 호출한 시간입니다. 또는 처리기가 ISyncMgrSynchronize::Synchronize 메서드에서 반환되었지만 아직 SynchronizeCompleted 콜백 메서드를 호출하지 않은 후
SetJunctionCLSID 현재 항목이 접합점인 경우 IShellFolder를 구현하는 개체의 CLSID(클래스 식별자)를 설정합니다.
SetKey "DefaultIcon" 값을 로드할 레지스트리 키를 설정합니다.
SetKeyState Ctrl 키와 Shift 키의 현재 상태에 따라 값을 설정합니다.
SetLine 진행률 대화 상자에 메시지를 표시합니다.
SetLocalPath 파일의 경로를 설정하고 저장합니다.
SetMenu 메뉴 밴드에 정적 메뉴를 추가합니다.
SetMenuContextHelpId 도움말 컨텍스트 식별자를 메뉴와 연결합니다.
SetMenuRestrictions . (IDefaultFolderMenuInitialize.SetMenuRestrictions)
SetMenuSB 보기 창에 복합 메뉴를 설치합니다.
SetMenuToolbar 메뉴 밴드에 메뉴를 추가합니다.

SetMode 구문 분석된 항목의 열거형 모드를 설정합니다.
SetMode 진행률 대화 상자 작업 모드를 설정합니다.
SetNamedValue 명명된 속성의 값을 설정합니다.
SetNameOf 파일 개체 또는 하위 폴더의 표시 이름을 설정하여 프로세스의 항목 식별자를 변경합니다.
SetNativeDisplayOrientation 디자인 모드 창에 대한 에뮬레이트된 디스플레이의 방향을 설정합니다.
SetNativeDisplaySize 디자인 모드에서 실행되는 Windows 스토어 앱에 사용할 스푸핑된 네이티브 디스플레이 크기를 설정합니다.
SetNavigateState Deprecated. 현재 탐색 상태를 설정합니다. 이 메서드는 커서 및 애니메이션에 영향을 줍니다.
SetNavigationRoot 저장 대화 상자의 찾아보기 폴더 옵션과 같이 네임스페이스 검색을 시작할 최상위 위치를 지정합니다. 사용자는 이 위치 위로 이동할 수 없습니다.
SetNoIcons 아이콘이 폴더 밴드에 표시되는지 여부를 설정합니다.
SetNormalIcon 표준 아이콘을 설정합니다.
SetNoShowUI 선택한 셸 항목과 연결된 UI를 표시할지 여부를 나타냅니다.
SetNoText 텍스트가 폴더 밴드에 표시되는지 여부를 설정합니다.

SetObjectCount SetObjectCount를 더 이상 사용할 수 없습니다.
SetObscured 도구 모음의 가려진 항목을 메뉴의 항목과 함께 조정합니다.
SetOkButtonLabel 열기 또는 저장 단추의 텍스트를 설정합니다.
SetOpenIcon 컨테이너가 "열린" 모양을 지정할 수 있도록 하는 아이콘을 설정합니다.
SetOperation 진행률 대화 상자 작업이 발생하는지, 플라이트 전 모드인지 실행 취소 모드인지를 설정합니다.
SetOperationFlags 현재 작업에 대한 매개 변수를 설정합니다.
SetOptions 현재 자동 완성 옵션을 설정합니다. (IACList2.SetOptions)
SetOptions 현재 자동 완성 옵션을 설정합니다. (IAutoComplete2.SetOptions)
SetOptions 현재 브라우저 옵션을 설정합니다.
SetOptions 대화 상자의 동작을 제어하는 플래그를 설정합니다.
SetOptions 라이브러리 옵션을 설정합니다.
SetOverlayIcon 작업 표시줄 단추에 오버레이를 적용하여 사용자에게 애플리케이션 상태 또는 알림을 나타냅니다.
SetOwner

Deprecated. 서비스 개체에 대한 소유자 참조를 선언합니다.
SetOwnerWindow 진행률 및 대화 상자 창에 대한 부모 또는 소유자 창을 설정합니다.
SetParameters 동사에 대한 매개 변수 값을 제공합니다.
SetParentAndItem 항목의 부모 및 부모의 자식 ID를 설정합니다.
SetPath 알려진 폴더에 새 경로를 할당합니다.
SetPath Shell 링크 개체의 대상에 대한 경로 및 파일 이름을 설정합니다. (ANSI)
SetPath Shell 링크 개체의 대상에 대한 경로 및 파일 이름을 설정합니다. (유니코드)
SetPinnedWindow 고정된 창을 설정합니다. 사용자가 볼 새 파일을 선택하면 셸은 파일 뷰어에게 새 창을 만드는 대신 고정된 창에 새 파일을 표시하도록 지시합니다.
SetPoints SetPoint를 변경하거나 사용할 수 없습니다.
SetPosition 바탕 화면 배경 화면 이미지의 표시 옵션을 설정하여 이미지를 가운데에 배치할지, 바일링할지 또는 확장해야 하는지를 결정합니다.
SetPosition 표시에 사용되는 점의 좌표를 설정합니다.
SetPresenterChoice 동기화 관리자 충돌 해결에 대해 선택한 종류와 집합의 모든 후속 충돌에 선택 사항을 적용할지 여부를 설정합니다.

SetPresenterNextStep 발표자가 수행하려는 작업을 동기화 관리자 충돌 해결의 다음 단계로 설정합니다.
SetProcessReference 호스트 셸 확장 및 기타 구성 요소가 호스트 프로세스가 조기에 닫히지 않도록 하는 COM(구성 요소 개체 모델) 개체를 제공합니다.
SetProgID 개체의 ProgID를 설정합니다.
SetProgress 진행률 대화 상자를 작업의 현재 상태로 업데이트합니다. (IProgressDialog.SetProgress)
SetProgress64 진행률 대화 상자를 작업의 현재 상태로 업데이트합니다. (IProgressDialog.SetProgress64)
SetProgressCallback ISyncMgrSynchronizeCallback 인터페이스를 설정합니다. 등록된 애플리케이션은 이 콜백 인터페이스를 사용하여 ISyncMgrSynchronize::P repareForSync 및 ISyncMgrSynchronize::Synchronize 메서드 내에서 상태 정보를 제공합니다.
SetProgressDialog 작업의 진행률을 표시하는 데 사용되는 대화 상자를 지정합니다.
SetProgressMessage 구현되지 않았습니다. (IFileOperation.SetProgressMessage)
SetProgressState 작업 표시줄 단추에 표시되는 진행률 표시기의 유형과 상태를 설정합니다.
SetProgressValue 작업 표시줄 단추에 호스트된 진행률 표시줄을 표시하거나 업데이트하여 전체 작업의 완료된 특정 백분율을 표시합니다.
SetProperties 항목 또는 항목에 설정할 속성 및 값 집합을 선언합니다.

SetProperties 저장되는 항목에 사용할 기본값을 정의하는 속성 저장소를 제공합니다.
SetProperties 항목에 적용해야 하는 속성을 설정합니다.
SetPropertyBag 속성 모음의 이름을 설정합니다.
SetPropertyKey 속성 키를 설정합니다.
SetPropertyStorage 지정된 직렬화된 속성 스토리지 데이터에서 속성 저장소 인스턴스를 초기화합니다.
SetProviderFilter 배열의 사용자 집합을 로컬 계정 또는 Microsoft 계정으로 제한합니다.
SetRect 브라우저에서 만든 보기 창의 크기와 위치를 설정합니다.
SetRect 미리 보기 처리기가 그리는 부모 hwnd 내의 영역을 변경하도록 지시합니다.
SetRedraw 보기를 다시 그릴 수 있도록 허용하거나 다시 그려지는 것을 방지합니다.
SetRedraw 다시 그리기를 켜고 끕니다.
SetReferent 바로 가기 개체의 참조를 설정합니다. (ANSI)
SetReferent 바로 가기 개체의 참조를 설정합니다. (유니코드)
SetReferrer

참조 파일에 따라 첨부 파일과 연결된 보안 영역을 설정합니다.
SetReferrer Deprecated. 새 창을 만들 때 영역 검사에 사용되는 PIDL(항목 식별자 목록)에 대한 포인터를 설정합니다.
SetRelativePath Shell 링크 개체의 상대 경로를 설정합니다. (ANSI)
SetRelativePath Shell 링크 개체의 상대 경로를 설정합니다. (유니코드)
SetSaveAsItem 다른 이름으로 저장 대화 상자에서 초기 항목으로 사용할 항목을 설정합니다.
SetScaleFactor 디자인 모드에서 실행되는 Windows 스토어 앱에 사용할 스푸핑된 디바이스 배율 인수를 설정합니다.
SetScope 지정된 대로 검색 범위를 설정합니다.
SetSelected 자격 증명을 선택할 때 호출됩니다. 구현자가 로그온 특성을 설정할 수 있도록 합니다.
SetSelectedControlItem 옵션 단추 그룹 또는 대화 상자에 있는 콤보 상자에서 특정 항목의 선택한 상태를 설정합니다.
SetSelection 선택 영역에 포함된 항목을 지정하는 셸 항목 배열을 제공합니다.
SetSerialization 자격 증명 공급자의 serialization 특성을 설정합니다.
SetShellFolder 찾아볼 메뉴 밴드에 대한 폴더를 지정합니다.
SetShortcutIcon 개체의 바로 가기 아이콘을 설정합니다.

SetShowCmd Shell 링크 개체에 대한 show 명령을 설정합니다. show 명령은 창의 초기 표시 상태를 설정합니다. (ANSI)
SetShowCmd Shell 링크 개체에 대한 show 명령을 설정합니다. show 명령은 창의 초기 표시 상태를 설정합니다. (유니코드)
SetShowWindow 지정된 창의 시각적 상태를 설정합니다.
SetSlideshow 바탕 화면 배경 화면 슬라이드 쇼에 사용할 이미지를 지정합니다.
SetSlideshowOptions 순서 섞기 및 타이밍에 대한 바탕 화면 배경 화면 슬라이드 쇼 설정을 설정합니다.
SetSortColumns 지정된 정렬 열을 기준으로 뷰를 설정하고 정렬합니다.
SetSortColumns 지정된 대로 정렬 열 방향 목록을 만듭니다.
SetSource 파일 전송 원본에 대한 대체 경로 또는 URL을 설정합니다.
SetStacks 지정된 대로 스택 키 목록을 만듭니다. 이 메서드가 호출되지 않으면 기본적으로 폴더가 쌓이지 않습니다.
SetStatusMessage 자격 증명 공급자가 IConnectableCredentialProviderCredential::Connect를 완료하려고 할 때 상태 메시지를 설정할 수 있도록 합니다.
SetStatusTextSB 컨테이너의 프레임 창 상태 표시줄에 있는 현재 위치 개체에 대한 상태 텍스트를 설정하고 표시합니다.
SetStringValue

로그온 UI 또는 자격 증명 UI가 사용자가 입력할 때 CPFT_EDIT_TEXT 필드의 텍스트를 업데이트할 수 있도록 합니다.
SetSubMenu 지정된 메뉴 모음 인터페이스를 호출하는 애플리케이션 개체 인터페이스의 하위 메뉴로 설정합니다.
SetTabActive 탭 또는 문서 창이 활성 창으로 설정되었음을 작업 표시줄에 알립니다.
SetTabOrder TDI(Tabbed-document interface) 또는 MDI(다중 문서 인터페이스) 애플리케이션의 그룹 플라이아웃에 새 썸네일을 삽입하거나 기존 썸네일을 애플리케이션 그룹의 새 위치로 이동합니다.
SetTabProperties 탭에서 주 애플리케이션 프레임 창 또는 탭 창을 축소판 그림으로 사용할지 또는 특정 상황에서 피킹 기능에서 사용할지 여부를 지정할 수 있습니다.
SetTargetSessionId 세션 식별자를 설정합니다.
SetText 보기에 항목이 없을 때 사용할 기본 텍스트를 설정합니다.
SetTextColor 미리 보기 처리기 내에서 텍스트의 색을 설정합니다.
SetTheme 지정된 테마를 설정합니다.
SetTheme 현재 창에 대해서만 바탕 화면 테마를 설정합니다.
SetThumbnailClip 작업 표시줄에 해당 창의 축소판 그림으로 표시할 창의 클라이언트 영역 일부를 선택합니다.
SetThumbnailStream 썸네일 스트림을 설정합니다. 이 메서드는 내부 전용이며 사진 애플리케이션에서만 호출할 수 있습니다.

SetThumbnailTooltip 마우스 포인터가 작업 표시줄 단추 플라이아웃의 개별 미리 보기에 놓일 때 표시되는 도구 설명의 텍스트를 지정하거나 업데이트합니다.
SetTileViewProperties 항목의 타일 속성 목록을 설정합니다.
SetTimes 파일 시간을 설정합니다.
SetTitle 진행률 대화 상자의 제목을 설정합니다.
SetTitle CreateProcess를 전달할 제목을 설정합니다.
SetTitle 대화 상자의 제목을 설정합니다.
SetTitle Deprecated. 브라우저 창의 제목을 설정합니다.
SetToolbarItems Windows 탐색기의 도구 모음에 도구 모음 항목을 추가합니다.
SetTopBrowser Deprecated. 기본 클래스가 최상위 브라우저 인스턴스가 되면 이를 알립니다.
SetUsageScenario 자격 증명 공급자가 유효한 시나리오를 정의합니다. 자격 증명 공급자가 초기화될 때마다 호출됩니다.
SetUserArray 로그온 또는 자격 증명 UI를 초기화하는 동안 시스템에서 호출되어 해당 UI에 표시할 사용자 집합을 검색합니다.
SetViewModeAndIconSize 보기 모드 및 이미지 크기를 설정하고 적용합니다.

SetViewProperty 뷰의 속성 캐시에 있는 항목의 속성을 캐시합니다.
SetVisibleColumns PROPERTYKEY 구조의 배열이 지정된 경우 열이 모두 표시되는 새 열 목록을 만듭니다. 기본값은 FolderTypeID를 기반으로 합니다.
SetWallpaper 바탕 화면 배경 화면을 설정합니다.
SetWatermark 워터마크로 사용할 비트맵을 제공합니다.
SetWindow 미리 보기 창의 부모 창과 미리 보기 창에 사용할 부모 내 영역을 설정합니다.
SetWindowContextHelpId 도움말 컨텍스트 식별자를 지정된 창에 연결합니다.
SetWindowSubclass 창 하위 클래스 콜백을 설치하거나 업데이트합니다.
SetWorkingDirectory Shell 링크 개체의 작업 디렉터리 이름을 설정합니다. (ANSI)
SetWorkingDirectory Shell 링크 개체의 작업 디렉터리 이름을 설정합니다. (유니코드)
SHAddFolderPathToLibrary 라이브러리에 폴더를 추가합니다.
SHAddFromPropSheetExtArray SHCreatePropSheetExtArray에서 만든 속성 시트 확장 배열에 페이지를 추가합니다.
SHAddToRecentDocs 가장 최근에 가장 자주 사용되는 항목을 추적하기 위해 항목에 액세스했음을 시스템에 알립니다. 이 함수를 사용하여 모든 사용량 현황 데이터를 지울 수도 있습니다.

SHAlloc 셸의 힙에서 메모리를 할당합니다.
SHAllocShared SHAllocShared를 변경하거나 사용할 수 없습니다.
SHAnsiToAnsi ANSI 문자열을 복사합니다.
SHAnsiToUnicode ANSI 코드 페이지에서 유니코드 코드 페이지로 문자열을 변환합니다.
SHAppBarMessage 앱바 메시지를 시스템에 보냅니다.
ShareExists 사용자 또는 공용 폴더가 공유되는지 여부를 쿼리합니다.
SharePrinters 컴퓨터에 연결된 모든 로컬 프린터를 공유하여 네트워크의 다른 컴퓨터에서 검색할 수 있도록 합니다.
SHAssocEnumHandlers 지정된 파일 이름 확장명 처리기 집합에 대한 열거형 개체를 반환합니다.
SHAssocEnumHandlersForProtocolByApplication 지정된 프로토콜과 연결된 처리기에 대한 액세스를 제공하는 열거형 인터페이스를 가져옵니다.
SHAutoComplete 자동 완성을 사용하여 URL 또는 파일 시스템 경로를 완료하는 데 도움이 되도록 시스템 편집 컨트롤에 지시합니다.
SHBindToFolderIDListParent 폴더 형식으로 지정된 셸 네임스페이스 항목과 해당 폴더를 기준으로 하는 항목 식별자 목록을 지정하면 이 함수는 네임스페이스 항목의 부모에 바인딩되고 선택적으로 항목 식별자 목록의 최종 구성 요소에 대한 포인터를 반환합니다.
SHBindToFolderIDListParentEx

호출자가 바인딩 컨텍스트를 지정할 수 있도록 하여 SHBindToFolderIDListParent 함수를 확장합니다.
SHBindToObject 셸 네임스페이스 IShellFolder::BindToObject 메서드를 사용하여 지정된 개체를 검색하고 바인딩합니다.
SHBindToParent PIDL(정규화된 항목 식별자 목록)에 대한 포인터를 사용하고 부모 개체에 대해 지정된 인터페이스 포인터를 반환합니다.
SHBrowseForFolderA 사용자가 Shell 폴더를 선택할 수 있는 대화 상자를 표시합니다. (ANSI)
SHBrowseForFolderW 사용자가 Shell 폴더를 선택할 수 있는 대화 상자를 표시합니다. (유니코드)
SHChangeNotification_Lock Shell 변경 알림 이벤트와 연결된 공유 메모리를 잠깁니다.
SHChangeNotification_Unlock 변경 알림에 대한 공유 메모리 잠금을 해제합니다.
SHChangeNotify 애플리케이션이 수행한 이벤트 시스템에 알합니다. 애플리케이션은 셸에 영향을 줄 수 있는 작업을 수행하는 경우 이 함수를 사용해야 합니다.
SHChangeNotifyDeregister SHChangeNotify 메시지를 수신하지 못하도록 클라이언트의 창 프로세스를 등록 취소합니다.
SHChangeNotifyRegister 파일 시스템에서 알림을 지원하는 경우 파일 시스템 또는 셸에서 알림을 받을 창을 등록합니다.
SHChangeNotifyRegisterThread 스레드의 비동기 레지스터 및 등록 취소를 사용하도록 설정합니다.
SHCloneSpecialIDList SHCloneSpecialIDList를 변경하거나 사용할 수 없습니다. 대신 SHGetSpecialFolderLocation을 사용합니다.
SHCLSIDFromString

CLSID(클래스 식별자)의 문자열 형식을 사용하고 해당 CLSID를 만듭니다.
SHCoCreateInstance SHCoCreateInstance를 변경하거나 사용할 수 없습니다. 대신 CoCreateInstance를 사용합니다.
SHCopyKeyA 원본 하위 키의 하위 키와 값을 대상 키에 재귀적으로 복사합니다. SHCopyKey는 키의 보안 특성을 복사하지 않습니다. (ANSI)
SHCopyKeyW 원본 하위 키의 하위 키와 값을 대상 키에 재귀적으로 복사합니다. SHCopyKey는 키의 보안 특성을 복사하지 않습니다. (유니코드)
SHCreateAssociationRegistration Windows에서 제공하는 인터페이스의 스톡 구현을 기반으로 IApplicationAssociationRegistration 개체를 만듭니다.
SHCreateDataObject 부모 폴더에 데이터 개체를 만듭니다.
SHCreateDefaultContextMenu Shell의 기본 상황에 맞는 메뉴 구현을 나타내는 개체를 만듭니다.
SHCreateDefaultExtractIcon IDefaultExtractIconInit 인터페이스를 통해 기본값을 추가로 구성할 수 있는 표준 아이콘 추출기를 만듭니다.
SHCreateDefaultPropertiesOp 아직 설정되지 않은 셸 항목의 기본 속성을 설정하는 파일 작업을 만듭니다.
SHCreateDirectory 새 파일 시스템 폴더를 만듭니다.
SHCreateDirectoryExA 선택적 보안 특성을 사용하여 새 파일 시스템 폴더를 만듭니다. (ANSI)
SHCreateDirectoryExW 선택적 보안 특성을 사용하여 새 파일 시스템 폴더를 만듭니다. (유니코드)

SHCreateFileExtractIconA SHCreateFileExtractIcon을 변경하거나 사용할 수 없습니다. (ANSI)
SHCreateFileExtractIconW SHCreateFileExtractIcon을 변경하거나 사용할 수 없습니다. (유니코드)
SHCreateItemFromIDList PIDL(항목 식별자 목록)에 대한 포인터에서 셸 항목 개체를 만들고 초기화합니다. 결과 셸 항목 개체는 IShellItem 인터페이스를 지원합니다.
SHCreateItemFromParsingName 구문 분석 이름에서 Shell 항목 개체를 만들고 초기화합니다.
SHCreateItemFromRelativeName 상대 구문 분석 이름에서 Shell 항목 개체를 만들고 초기화합니다.
SHCreateItemInKnownFolder 알려진 폴더 내에 있는 단일 파일에 대한 Shell 항목 개체를 만듭니다.
SHCreateItemWithParent 부모 폴더와 자식 항목 ID가 지정된 셸 항목을 만듭니다.
SHCreateLibrary IShellLibrary 개체를 만듭니다.
SHCreateMemStream CreateStreamOnHGlobal과 유사한 프로세스를 사용하여 메모리 스트림을 만듭니다.
SHCreateProcessAsUserW 지정된 실행 파일을 실행하기 위해 새 사용자 모드 프로세스 및 기본 스레드를 만듭니다.
SHCreatePropSheetExtArray SHCreatePropSheetExtArray를 변경하거나 사용할 수 없습니다.
SHCreateQueryCancelAutoPlayMoniker Deprecated. QueryCancelAutoPlay 클래스 모니커를 만든 다음, ROT(실행 중인 개체 테이블)에 IQueryCancelAutoPlay 처리기를 등록하는 데 사용할 수 있습니다.

SHCreateShellFolderView 기본 Shell 폴더 뷰 개체(DefView)의 새 인스턴스를 만듭니다.
SHCreateShellFolderViewEx 기본 Shell 폴더 뷰 개체의 새 인스턴스를 만듭니다. 이 함수 대신 SHCreateShellFolderView를 사용하는 것이 좋습니다.
SHCreateShellItem IShellItem 개체를 만듭니다.
SHCreateShellItemArray Shell 항목 배열 개체를 만듭니다.
SHCreateShellItemArrayFromDataObject 데이터 개체에서 Shell 항목 배열 개체를 만듭니다.
SHCreateShellItemArrayFromIDLists ITEMIDLIST 구조 목록에서 셸 항목 배열 개체를 만듭니다.
SHCreateShellItemArrayFromShellItem 단일 셸 항목에서 한 요소의 배열을 만듭니다.
SHCreateShellPalette 지정된 디바이스 컨텍스트에 대한 하프톤 팔레트를 만듭니다.
SHCreateStdEnumFmtEtc SHCreateStdEnumFmtEtc를 변경하거나 사용할 수 없습니다.
SHCreateStreamOnFileA SHCreateStreamOnFile을 변경하거나 사용할 수 없습니다. 대신 SHCreateStreamOnFileEx를 사용합니다. (ANSI)
SHCreateStreamOnFileEx 파일을 열거나 만들고 해당 파일을 읽거나 쓸 스트림을 검색합니다.
SHCreateStreamOnFileW

SHCreateStreamOnFile을 변경하거나 사용할 수 없습니다. 대신 SHCreateStreamOnFileEx를 사용합니다. (유니코드)
SHCreateThread 스레드를 만듭니다.
SHCreateThreadRef COM(Component Object Model) 개체에 대한 스레드별 참조를 만듭니다.
SHCreateThreadWithHandle 새 스레드를 만들고 해당 핸들을 검색합니다.
SHDefExtractIconA 파일에서 아이콘을 추출하는 기본 처리기를 제공합니다. (ANSI)
SHDefExtractIconW 파일에서 아이콘을 추출하는 기본 처리기를 제공합니다. (유니코드)
SHDeleteEmptyKeyA 빈 키를 삭제합니다. (ANSI)
SHDeleteEmptyKeyW 빈 키를 삭제합니다. (유니코드)
SHDeleteKeyA 하위 키와 모든 하위 항목을 삭제합니다. 이 함수는 레지스트리에서 키와 모든 키 값을 제거합니다. (ANSI)
SHDeleteKeyW 하위 키와 모든 하위 항목을 삭제합니다. 이 함수는 레지스트리에서 키와 모든 키 값을 제거합니다. (유니코드)
SHDeleteValueA 지정된 레지스트리 키에서 명명된 값을 삭제합니다. (ANSI)
SHDeleteValueW 지정된 레지스트리 키에서 명명된 값을 삭제합니다. (유니코드)
SHDestroyPropSheetExtArray

SHCreatePropSheetExtArray에서 만든 배열을 가리키는 속성 시트 처리기를 해제합니다.
SHDoDragDrop 끌어서 놓기 작업을 실행합니다. 요청 시 원본 끌어서 만들기와 끌기 이미지를 지원합니다.
Shell_GetCachedImageIndex Shell_GetCachedImageIndex 변경되거나 사용할 수 없습니다.
Shell_GetCachedImageIndexA Shell_GetCachedImageIndex 변경되거나 사용할 수 없습니다. (ANSI)
Shell_GetCachedImageIndexW Shell_GetCachedImageIndex 변경되거나 사용할 수 없습니다. (유니코드)
Shell_GetImageLists 크고 작은 아이콘에 대한 시스템 이미지 목록을 검색합니다.
Shell_MergeMenus Shell_MergeMenus 변경되거나 사용할 수 없습니다.
Shell_NotifyIconA 작업 표시줄의 상태 영역에 메시지를 보냅니다. (ANSI)
Shell_NotifyIconGetRect 알림 아이콘의 경계 사각형의 화면 좌표를 가져옵니다.
Shell_NotifyIconW 작업 표시줄의 상태 영역에 메시지를 보냅니다. (유니코드)
ShellAboutA ShellAbout 대화 상자를 표시합니다. (ANSI)
ShellAboutW ShellAbout 대화 상자를 표시합니다. (유니코드)
ShellExecuteA 지정된 파일에 대한 작업을 수행합니다. (ShellExecuteA)

ShellExecuteExA 지정된 파일에 대한 작업을 수행합니다. (ShellExecuteExA)
ShellExecuteExW 지정된 파일에 대한 작업을 수행합니다. (ShellExecuteExW)
ShellExecuteW 지정된 파일에 대한 작업을 수행합니다. (ShellExecuteW)
ShellMessageBoxA ShellMessageBox를 변경하거나 사용할 수 없습니다. (ANSI)
ShellMessageBoxW ShellMessageBox를 변경하거나 사용할 수 없습니다. (유니코드)
SHEmptyRecycleBinA 지정된 드라이브에서 휴지통을 비웁니다. (ANSI)
SHEmptyRecycleBinW 지정된 드라이브에서 휴지통을 비웁니다. (유니코드)
SHEnumerateUnreadMailAccountsA 읽지 않은 전자 메일이 있는 사용자 계정을 열거합니다. (ANSI)
SHEnumerateUnreadMailAccountsW 읽지 않은 전자 메일이 있는 사용자 계정을 열거합니다. (유니코드)
SHEnumKeyExA 지정된 열린 레지스트리 키의 하위 키를 열거합니다. (SHEnumKeyExA)
SHEnumKeyExW 지정된 열린 레지스트리 키의 하위 키를 열거합니다. (SHEnumKeyExW)
SHEnumValueA 지정된 열린 레지스트리 키의 값을 열거합니다. (ANSI)
SHEnumValueW

지정된 열린 레지스트리 키의 값을 열거합니다. (유니코드)
SHEvaluateSystemCommandTemplate CreateProcess 또는 ShellExecute 호출에 사용되는 매개 변수의 엄격한 유효성 검사를 적용합니다.
SHFileOperationA 파일 시스템 개체를 복사, 이동, 이름 바꾸기 또는 삭제합니다. 이 함수는 Windows Vista에서 IFileOperation으로 대체되었습니다.
SHFileOperationW 파일 시스템 개체를 복사, 이동, 이름 바꾸기 또는 삭제합니다. Windows Vista 이상 릴리스에서는 이 함수 대신 IFileOperation을 사용하는 것이 좋습니다.
SHFind_InitMenuPopup SHFind_InitMenuPopup 변경되거나 사용할 수 없습니다.
SHFindFiles SHFindFiles를 변경하거나 사용할 수 없습니다.
SHFlushSFCache SHFlushSFCache를 변경하거나 사용할 수 없습니다.
SHFormatDateTimeA SHFormatDateTime을 변경하거나 사용할 수 없습니다. (ANSI)
SHFormatDateTimeW SHFormatDateTime을 변경하거나 사용할 수 없습니다. (유니코드)
SHFormatDrive SHFormatDrive를 변경하거나 사용할 수 없습니다.
SHFree SHAlloc에서 할당한 메모리를 해제합니다.
SHFreeNameMappings SHFileOperation 함수에 의해 검색된 파일 이름 매핑 개체를 해제합니다.
SHFreeShared

SHFreeShared를 변경하거나 사용할 수 없습니다.
SHGetAssocKeys IQueryAssociations 개체와 연결된 클래스 하위 키의 배열을 검색합니다.
SHGetAttributesFromDataObject SHGetAttributesFromDataObject를 변경하거나 사용할 수 없습니다.
SHGetDataFromIDListA 상대 식별자 목록에서 확장 속성 데이터를 검색합니다. (ANSI)
SHGetDataFromIDListW 상대 식별자 목록에서 확장 속성 데이터를 검색합니다. (유니코드)
SHGetDesktopFolder 셸 네임스페이스의 루트인 데스크톱 폴더에 대한 IShellFolder 인터페이스를 검색합니다.
SHGetDiskFreeSpaceExA 디스크 볼륨에 대한 디스크 공간 정보를 검색합니다. (ANSI)
SHGetDiskFreeSpaceExW 디스크 볼륨에 대한 디스크 공간 정보를 검색합니다. (유니코드)
SHGetDriveMedia 지정된 드라이브에 있는 미디어의 형식을 반환합니다.
SHGetFileInfoA 파일, 폴더, 디렉터리 또는 드라이브 루트와 같은 파일 시스템의 개체에 대한 정보를 검색합니다. (ANSI)
SHGetFileInfoW 파일, 폴더, 디렉터리 또는 드라이브 루트와 같은 파일 시스템의 개체에 대한 정보를 검색합니다. (유니코드)
SHGetFolderLocation Deprecated. 폴더의 경로를 ITEMIDLIST 구조체로 검색합니다.
SHGetFolderPathA Deprecated. (SHGetFolderPathA)

SHGetFolderPathAndSubDirA 폴더의 경로를 가져오고 사용자가 제공한 하위 폴더 경로를 추가합니다. (ANSI)
SHGetFolderPathAndSubDirW 폴더의 경로를 가져오고 사용자가 제공한 하위 폴더 경로를 추가합니다. (유니코드)
SHGetFolderPathW Deprecated. (SHGetFolderPathW)
SHGetIconOverlayIndexA 시스템 이미지 목록에서 오버레이 아이콘의 인덱스 값을 반환합니다. (ANSI)
SHGetIconOverlayIndexW 시스템 이미지 목록에서 오버레이 아이콘의 인덱스 값을 반환합니다. (유니코드)
SHGetIDListFromObject 개체의 PIDL(항목 식별자 목록)에 대한 포인터를 검색합니다.
SHGetImageList 이미지 목록을 검색합니다.
SHGetInstanceExplorer 호스트 셸 확장 및 기타 구성 요소가 호스트 프로세스가 조기에 닫히지 않도록 하는 인터페이스를 검색합니다.
SHGetInverseCMAP 하프톤 팔레트의 역색 테이블 매핑을 검색합니다.
SHGetItemFromDataObject IDataObject에서 지정한 항목을 기반으로 IShellItem 또는 관련 개체를 만듭니다.
SHGetItemFromObject 개체에 대한 IShellItem을 검색합니다.
SHGetKnownFolderIDList 알려진 폴더의 경로를 ITEMIDLIST 구조체로 검색합니다.

SHGetKnownFolderItem 알려진 폴더를 나타내는 IShellItem 개체를 검색합니다.
SHGetKnownFolderPath 폴더의 KNOWNFOLDERID로 식별된 알려진 폴더의 전체 경로를 검색합니다.
SHGetLocalizedName Shell 폴더에 있는 파일의 지역화된 이름을 검색합니다.
SHGetMalloc 셸의 IMalloc 인터페이스에 대한 포인터를 검색합니다.
SHGetNameFromIDList IDList로 식별된 항목의 표시 이름을 검색합니다.
SHGetNewLinkInfoA 바로 가기의 제안된 대상에 따라 새 바로 가기의 이름을 만듭니다. 이 함수는 바로 가기를 만들지 않고 이름만 만듭니다. (ANSI)
SHGetNewLinkInfoW 바로 가기의 제안된 대상에 따라 새 바로 가기의 이름을 만듭니다. 이 함수는 바로 가기를 만들지 않고 이름만 만듭니다. (유니코드)
SHGetPathFromIDListA 항목 식별자 목록을 파일 시스템 경로로 변환합니다. (ANSI)
SHGetPathFromIDListEx 항목 식별자 목록을 파일 시스템 경로로 변환합니다. 이 함수는 문자열 버퍼의 초기 크기를 설정하고 아래 옵션을 선언할 수 있도록 하여 SHGetPathFromIDList를 확장합니다.
SHGetPathFromIDListW 항목 식별자 목록을 파일 시스템 경로로 변환합니다. (유니코드)
SHGetRealIDL SHGetRealIDL을 변경하거나 사용할 수 없습니다.
SHGetSetFolderCustomSettings

SHGetSetFolderCustomSettings를 변경하거나 사용할 수 없습니다.
SHGetSetSettings SHGetSetSettings를 변경하거나 사용할 수 없습니다.
SHGetSettings 현재 셸 옵션 설정을 검색합니다.
SHGetSpecialFolderLocation SHGetSpecialFolderLocation은 지원되지 않으며 나중에 변경되거나 사용할 수 없습니다. 대신 SHGetFolderLocation을 사용합니다.
SHGetSpecialFolderPathA SHGetSpecialFolderPath는 지원되지 않습니다. 대신 ShGetFolderPath를 사용합니다. (ANSI)
SHGetSpecialFolderPathW SHGetSpecialFolderPath는 지원되지 않습니다. 대신 ShGetFolderPath를 사용합니다. (유니코드)
SHGetStockIconInfo 시스템 정의 셸 아이콘에 대한 정보를 검색합니다.
SHGetTemporaryPropertyForItem 지정된 항목에 대한 임시 속성을 검색합니다. 임시 속성은 항목에 다시 유지되지 않고 IShellItem 개체의 수명 동안만 속성을 보유하는 읽기/쓰기 저장소입니다.
SHGetThreadRef SHSetThreadRef에서 설정한 스레드별 개체 참조를 검색합니다.
SHGetUnreadMailCountA 모든 전자 메일 계정에 대해 지정된 사용자의 읽지 않은 메시지 수를 검색합니다. (ANSI)
SHGetUnreadMailCountW 모든 전자 메일 계정에 대해 지정된 사용자의 읽지 않은 메시지 수를 검색합니다. (유니코드)
SHGetValueA 레지스트리 값을 검색합니다. (SHGetValueA)
SHGetValueW

레지스트리 값을 검색합니다. (SHGetValueW)
SHGetViewStatePropertyBag SHGetViewStatePropertyBag을 변경하거나 사용할 수 없습니다.
SHGlobalCounterDecrement 전역 카운터를 감소합니다.
SHGlobalCounterGetValue 전역 카운터의 현재 값을 가져옵니다.
SHGlobalCounterIncrement 전역 카운터를 증분합니다.
SHHandleUpdateImage SHHandleUpdateImage를 변경하거나 사용할 수 없습니다.
SHILCreateFromPath SHILCreateFromPath를 변경하거나 사용할 수 없습니다.
SHInvokePrinterCommandA 프린터 개체에서 명령을 실행합니다. (ANSI)
SHInvokePrinterCommandW 프린터 개체에서 명령을 실행합니다. (유니코드)
SHIsFileAvailableOffline 파일 또는 폴더를 오프라인으로 사용할 수 있는지 여부를 결정합니다. 또한 이 함수는 네트워크에서, 로컬 오프라인 파일 캐시에서 또는 두 위치에서 파일을 열지 여부를 결정합니다.
SHIsLowMemoryMachine 지원되지 않습니다. (SHIsLowMemoryMachine)
SHLimitInputEdit 편집 컨트롤의 유효한 문자에 대한 제한을 설정합니다.
SHLoadIndirectString

해당 리소스를 간접 문자열 형식('@' 기호로 시작하는 문자열)으로 지정하면 지정된 텍스트 리소스를 추출합니다.
SHLoadInProc Shell 프로세스의 컨텍스트 내에서 지정된 개체 클래스의 인스턴스를 만듭니다. Windows Vista 이상:_This 함수가 비활성화되어 E_NOTIMPL 반환합니다.
SHLoadLibraryFromItem 지정된 라이브러리 정의 파일에서 IShellLibrary 개체를 만들고 로드합니다.
SHLoadLibraryFromKnownFolder 지정된 알려진 폴더 ID에 대한 IShellLibrary 개체를 만들고 로드합니다.
SHLoadLibraryFromParsingName 지정된 경로에 대한 IShellLibrary 개체를 만들고 로드합니다.
SHLoadNonloadedIconOverlayIdentifiers 오버레이 정보가 필요한 다음 작업 중에 생성에 실패했거나 시작 시 생성에 존재하지 않는 아이콘 오버레이 식별자를 로드해야 한다는 신호를 Shell에 알릴 수 있습니다. 이미 로드된 식별자는 영향을 받지 않습니다.
SHLoadOLE Deprecated. 이전 버전과의 호환성을 위해 제공됩니다.
SHLocalStrDupA 새로 할당된 메모리에서 문자열의 복사본을 만듭니다. (SHLocalStrDupA)
SHLocalStrDupW 새로 할당된 메모리에서 문자열의 복사본을 만듭니다. (SHLocalStrDupW)
SHLockShared SHLockShared를 변경하거나 사용할 수 없습니다.
SHMapPIDLToSystemImageListIndex SHMapPIDLToSystemImageListIndex를 변경하거나 사용할 수 없습니다.
SHMessageBoxCheckA SHMessageBoxCheck를 변경하거나 사용할 수 없습니다. (ANSI)

SHMessageBoxCheckW SHMessageBoxCheck를 변경하거나 사용할 수 없습니다. (유니코드)
SHMultiFileProperties 파일 집합에 대한 병합된 속성 시트를 표시합니다. 모든 파일에 공통적인 속성 값이 표시되고 다른 속성 값은 문자열(여러 값)을 표시합니다.
SHObjectProperties SHObjectProperties를 변경하거나 사용할 수 없습니다.
SHOpenFolderAndSelectItems 특정 폴더에 지정된 항목이 선택된 Windows 탐색기 창을 엽니다.
SHOpenPropSheetA SHOpenPropSheet를 변경하거나 사용할 수 없습니다. (ANSI)
SHOpenPropSheetW SHOpenPropSheet를 변경하거나 사용할 수 없습니다. (유니코드)
SHOpenRegStream2A 레지스트리 값을 열고 값을 읽거나 쓰는 데 사용할 수 있는 스트림을 제공합니다. 이 함수는 SHOpenRegStream을 대체합니다. (ANSI)
SHOpenRegStream2W 레지스트리 값을 열고 값을 읽거나 쓰는 데 사용할 수 있는 스트림을 제공합니다. 이 함수는 SHOpenRegStream을 대체합니다. (유니코드)
SHOpenRegStreamA Deprecated. (SHOpenRegStreamA)
SHOpenRegStreamW Deprecated. (SHOpenRegStreamW)
SHOpenWithDialog [함께 열기] 대화 상자를 표시합니다.
ShortAdd

SHORT 형식의 두 값을 추가합니다.
ShortMult SHORT 형식의 두 값을 곱합니다.
ShortSub SHORT 형식의 한 값을 다른 값에서 뺍니다.
ShortToChar SHORT 형식의 값을 CHAR 값으로 변환합니다.
ShortToDWordPtr SHORT 형식의 값을 DWORD_PTR 형식의 값으로 변환합니다.
ShortToInt8 SHORT 형식의 값을 INT8 형식의 값으로 변환합니다.
ShortToUChar SHORT 형식의 값을 UCHAR 값으로 변환합니다.
ShortToUInt SHORT 형식의 값을 UINT 형식의 값으로 변환합니다.
ShortToUInt8 SHORT 형식의 값을 UINT8 형식의 값으로 변환합니다.
ShortToUIntPtr SHORT 형식의 값을 UINT_PTR 형식의 값으로 변환합니다.
ShortToULong SHORT 형식의 값을 ULONG 형식의 값으로 변환합니다.
ShortToULongLong SHORT 형식의 값을 ULONGLONG 형식의 값으로 변환합니다.
ShortToULongPtr SHORT 형식의 값을 ULONG_PTR 형식의 값으로 변환합니다.

ShortToUShort SHORT 형식의 값을 USHORT 형식의 값으로 변환합니다.
ShouldShow 필터를 통해 개별 항목을 허용할지 여부와 차단해야 하는 항목을 지정합니다.
표시 풍선 스타일 도구 설명에 사용자 정보를 표시합니다.
표시 파일을 표시합니다. (ANSI)
표시 파일을 표시합니다. (유니코드)
표시 끌기 이미지 관리자에게 끌기 이미지를 표시하거나 숨기도록 알릴 수 있습니다.
표시 모달 창을 시작합니다.
표시 알림을 표시합니다.
ShowContactCardForWindow 지정한 창에 연락처의 UI를 표시합니다.
ShowControlWindow Deprecated. 다양한 프레임 컨트롤을 표시하거나 숨깁니다.
ShowDeskBand 지정된 책상 밴드를 표시합니다.
ShowDW 도킹 창 개체에 자신을 표시하거나 숨기도록 지시합니다.
ShowError

사용자가 오류 탭에서 연결된 메시지를 두 번 클릭할 때 등록된 애플리케이션 처리기에서 동기화 관리자가 호출합니다.
ShowErrorCompleted PrepareForSync 작업이 완료되기 전이나 후에 등록된 애플리케이션의 처리기에서 호출됩니다.
페이지 입력 창이 표시되기 전에 호출되어 앱 창에서 사용 가능한 축소된 화면 공간에 대한 응답으로 UI에 필요한 조정을 수행할 수 있도록 합니다.
ShowInitialize 파일 뷰어가 파일을 표시할 수 있는지 여부를 확인할 수 있습니다. (ANSI)
ShowInitialize 파일 뷰어가 파일을 표시할 수 있는지 여부를 확인할 수 있습니다. (유니코드)
ShowProperties 사용자가 선택 대화 상자에서 항목을 선택한 다음 속성 단추를 클릭하면 동기화 관리자가 호출합니다.
ShowPropertiesCompleted ShowProperties 작업이 완료되기 전이나 후에 등록된 애플리케이션의 처리기에서 호출됩니다.
ShowResolveConflictUIAsync IFileSyncMergeHandler::ShowResolveConflictUIAsync 메서드
ShowShareUIForWindow 지정된 창의 콘텐츠를 공유하기 위한 UI를 표시합니다.
ShowSharingWizard 사용자가 홈 그룹을 만들 수 있는 마법사를 표시한 다음 마법사를 통해 사용자가 선택한 공유 옵션을 검색합니다.
SHParseDisplayName 셸 네임스페이스 개체의 표시 이름을 항목 식별자 목록으로 변환하고 개체의 특성을 반환합니다. 이 함수는 문자열을 PIDL(항목 식별자 목록)에 대한 포인터로 변환하는 기본 메서드입니다.
SHPathPrepareForWriteA 경로가 있는지 확인합니다. (ANSI)

SHPathPrepareForWriteW 경로가 있는지 확인합니다. (유니코드)
SHQueryInfoKeyA 지정된 레지스트리 키에 대한 정보를 검색합니다. (ANSI)
SHQueryInfoKeyW 지정된 레지스트리 키에 대한 정보를 검색합니다. (유니코드)
SHQueryRecycleBinA 지정된 드라이브에 대한 휴지통의 크기와 휴지통의 항목 수를 검색합니다. (ANSI)
SHQueryRecycleBinW 지정된 드라이브에 대한 휴지통의 크기와 휴지통의 항목 수를 검색합니다. (유니코드)
SHQueryUserNotificationState 현재 사용자의 컴퓨터 상태를 확인하여 알림 보내기가 적절한지 여부를 확인합니다.
SHQueryValueExA 레지스트리 키를 열고 특정 값을 쿼리합니다. (ANSI)
SHQueryValueExW 레지스트리 키를 열고 특정 값을 쿼리합니다. (유니코드)
SHRegCloseUSKey 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 사용자별 레지스트리 하위 키에 대한 핸들을 닫습니다.
SHRegCreateUSKeyA 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 레지스트리 하위 키를 만들거나 엽니다. (ANSI)
SHRegCreateUSKeyW 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 레지스트리 하위 키를 만들거나 엽니다. (유니코드)
SHRegDeleteEmptyUSKeyA

사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 빈 레지스트리 하위 키를 삭제합니다. (ANSI)
SHRegDeleteEmptyUSKeyW 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 빈 레지스트리 하위 키를 삭제합니다. (유니코드)
SHRegDeleteUSValueA 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 레지스트리 하위 키 값을 삭제합니다. (ANSI)
SHRegDeleteUSValueW 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 레지스트리 하위 키 값을 삭제합니다. (유니코드)
SHRegDuplicateHKey 레지스트리 키의 HKEY 핸들을 복제합니다.
SHRegEnumUSKeyA 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 레지스트리 하위 키의 하위 키를 열거합니다. (ANSI)
SHRegEnumUSKeyW 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 레지스트리 하위 키의 하위 키를 열거합니다. (유니코드)
SHRegEnumUSValueA 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 지정된 레지스트리 하위 키의 값을 열거합니다. (ANSI)
SHRegEnumUSValueW 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 지정된 레지스트리 하위 키의 값을 열거합니다. (유니코드)
SHRegGetBoolUSValueA 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)의 레지스트리 하위 키에서 부울 값을 검색합니다. (ANSI)
SHRegGetBoolUSValueW

사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)의 레지스트리 하위 키에서 부울 값을 검색합니다. (유니코드)
SHRegGetBoolValueFromHKCUHKLM 레지스트리 키 값을 평가하고 값이 있는지와 예상 상태가 실제 상태와 일치하는지 여부를 반영하는 부울 값을 반환합니다.
SHRegGetIntW 레지스트리에서 숫자 문자열 값을 읽고 정수로 변환합니다.
SHRegGetPathA 레지스트리에서 파일 경로를 검색하여 필요에 따라 환경 변수를 확장합니다. (ANSI)
SHRegGetPathW 레지스트리에서 파일 경로를 검색하여 필요에 따라 환경 변수를 확장합니다. (유니코드)
SHRegGetUSValueA 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)의 레지스트리 하위 키에서 값을 검색합니다. (ANSI)
SHRegGetUSValueW 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)의 레지스트리 하위 키에서 값을 검색합니다. (유니코드)
SHRegGetValueA 레지스트리 값을 검색합니다. (SHRegGetValueA)
SHRegGetValueFromHKCUHKLM 레지스트리에서 지정된 정보를 가져옵니다.
SHRegGetValueW 레지스트리 값을 검색합니다. (SHRegGetValueW)
SHRegOpenUSKeyA 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 레지스트리 하위 키를 엽니다. (ANSI)
SHRegOpenUSKeyW

사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 레지스트리 하위 키를 엽니다. (유니코드)
SHRegQueryInfoUSKeyA 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 지정된 레지스트리 하위 키에 대한 정보를 검색합니다. (ANSI)
SHRegQueryInfoUSKeyW 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 지정된 레지스트리 하위 키에 대한 정보를 검색합니다. (유니코드)
SHRegQueryUSValueA 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 열린 레지스트리 하위 키와 연결된 지정된 이름의 형식 및 데이터를 검색합니다. (ANSI)
SHRegQueryUSValueW 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 열린 레지스트리 하위 키와 연결된 지정된 이름의 형식 및 데이터를 검색합니다. (유니코드)
SHRegSetPathA 파일 경로를 사용하고 폴더 이름을 환경 문자열로 바꾸고 결과 문자열을 레지스트리에 배치합니다. (ANSI)
SHRegSetPathW 파일 경로를 사용하고 폴더 이름을 환경 문자열로 바꾸고 결과 문자열을 레지스트리에 배치합니다. (유니코드)
SHRegSetUSValueA 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 레지스트리 하위 키 값을 설정합니다. (ANSI)
SHRegSetUSValueW 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 레지스트리 하위 키 값을 설정합니다. (유니코드)
SHRegSetValue 지원되지 않습니다. (SHRegSetValue)
SHRegWriteUSValueA

사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)의 레지스트리 하위 키에 값을 씁니다. (ANSI)
SHRegWriteUSValueW 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)의 레지스트리 하위 키에 값을 씁니다. (유니코드)
SHReleaseThreadRef 스레드 프로시저가 반환되기 전에 스레드 참조를 해제합니다.
SHRemoveFolderPathFromLibrary 라이브러리에서 폴더를 제거합니다.
SHRemoveLocalizedName Shell 폴더에 있는 파일의 지역화된 이름을 제거합니다.
SHReplaceFromPropSheetExtArray 페이지를 바꾸기 위해 속성 시트 확장 배열의 각 속성 시트를 요청합니다. 각 페이지는 최대 하나의 대체가 허용됩니다.
SHResolveFolderPathInLibrary 이동되거나 이름이 바뀐 라이브러리 폴더의 대상 위치를 확인하려고 시도합니다.
SHResolveLibrary 라이브러리의 모든 위치(이동 또는 이름이 변경된 위치)도 확인합니다.
SHRestricted SHRestricted는 변경되거나 사용할 수 없습니다.
SHRunControlPanel 제어판 항목을 엽니다.
SHSaveLibraryInFolderPath IShellLibrary 개체를 디스크에 저장합니다.
SHSendMessageBroadcastA 시스템의 모든 최상위 창에 메시지를 보냅니다. (ANSI)

SHSendMessageBroadcastW 시스템의 모든 최상위 창에 메시지를 보냅니다. (유니코드)
SHSetDefaultProperties 셸 항목에 기본 속성 집합을 적용합니다.
SHSetFolderPathA Deprecated. CSIDL로 식별되는 시스템 폴더에 새 경로를 할당합니다. (ANSI)
SHSetFolderPathW Deprecated. CSIDL로 식별되는 시스템 폴더에 새 경로를 할당합니다. (유니코드)
SHSetInstanceExplorer 호스트 셸 확장 및 기타 구성 요소가 호스트 프로세스가 조기에 닫히지 않도록 하는 인터페이스를 제공합니다.
SHSetKnownFolderPath 알려진 폴더를 새 위치로 리디렉션합니다.
SHSetLocalizedName Shell 폴더에 있는 파일의 지역화된 이름을 설정합니다.
SHSetTemporaryPropertyForItem 지정된 항목에 대한 임시 속성을 설정합니다. 임시 속성은 항목에 다시 쓰는 대신 IShellItem 개체의 수명 동안만 속성을 보유하는 읽기/쓰기 저장소에 유지됩니다.
SHSetThreadRef COM(Component Object Model) 개체에 대한 스레드별 참조를 저장합니다. 이렇게 하면 호출자가 스레드의 수명을 제어하여 호출자가 준비되기 전에 Windows에서 스레드를 종료하지 않도록 할 수 있습니다.
SHSetUnreadMailCountA 레지스트리에 지정된 전자 메일 계정에 대한 현재 사용자의 읽지 않은 메시지 수를 저장합니다. (ANSI)
SHSetUnreadMailCountW 레지스트리에 지정된 전자 메일 계정에 대한 현재 사용자의 읽지 않은 메시지 수를 저장합니다. (유니코드)
SHSetValueA

레지스트리 키의 값을 설정합니다. (ANSI)
SHSetValueW 레지스트리 키의 값을 설정합니다. (유니코드)
SHShellFolderView_Message SHShellFolderView_Message 변경되거나 사용할 수 없습니다.
SHShowManageLibraryUI 사용자가 라이브러리 폴더 및 기본 저장 위치를 관리할 수 있도록 하는 라이브러리 관리 대화 상자를 표시합니다.
SHSimpleIDListFromPath Deprecated. 경로를 전달할 때 ITEMIDLIST 구조체에 대한 포인터를 반환합니다.
SHSkipJunction 바인딩 컨텍스트를 검사하여 특정 구성 요소 개체에 바인딩하는 것이 안전한지 확인합니다.
SHStartNetConnectionDialogA SHStartNetConnectionDialog를 변경하거나 사용할 수 없습니다. (ANSI)
SHStartNetConnectionDialogW SHStartNetConnectionDialog를 변경하거나 사용할 수 없습니다. (유니코드)
SHStrDupA 새로 할당된 메모리에서 문자열의 복사본을 만듭니다. (SHStrDupA)
SHStrDupW 새로 할당된 메모리에서 문자열의 복사본을 만듭니다. (SHStrDupW)
SHStripMneumonicA 문자열에서 니모닉 마커를 제거합니다. (ANSI)
SHStripMneumonicW 문자열에서 니모닉 마커를 제거합니다. (유니코드)
SHTestTokenMembership

CheckTokenMembership을 사용하여 지정된 토큰이 지정된 RID를 사용하는 로컬 그룹의 구성원인지 여부를 테스트합니다.
SHUnicodeToAnsi 문자열을 유니코드 코드 페이지에서 ANSI 코드 페이지로 변환합니다.
SHUnicodeToUnicode 유니코드 문자열을 복사합니다.
SHUnlockShared SHUnlockShared를 변경하거나 사용할 수 없습니다.
SHUpdateImageA 시스템 이미지 목록의 이미지가 변경되었음을 셸에 알깁니다. (ANSI)
SHUpdateImageW 시스템 이미지 목록의 이미지가 변경되었음을 셸에 알깁니다. (유니코드)
SHValidateUNC SHValidateUNC를 변경하거나 사용할 수 없습니다.
SignalFileOpen SignalFileOpen을 변경하거나 사용할 수 없습니다.
SizeTAdd size_t 형식의 두 값을 추가합니다.
SizeTMult size_t 형식의 한 값을 다른 값으로 곱합니다. (SizeTMult)
SizeTSub size_t 형식의 한 값을 다른 값에서 뺍니다.
건너뛰기 현재 구현되지 않았습니다. (IEnumExplorerCommand.Skip)
건너뛰기 지정된 수의 개체를 건너뜁니다.

건너뛰기 지정된 수의 IDLIST_ABSOLUTE 항목을 건너뜁니다.
건너뛰기 열거형 시퀀스에서 지정된 개수의 요소를 건너뜁니다.
건너뛰기 지정된 수의 개체를 건너뜁니다.
건너뛰기 지정된 수의 리소스를 건너뜁니다.
건너뛰기 열거형에서 지정된 수의 IShellItem 인터페이스를 건너뜁니다. 인터페이스를 검색할 때 사용됩니다.
건너뛰기 열거형에서 지정된 충돌 횟수를 건너뜁니다.
건너뛰기 열거형에서 지정된 수의 이벤트를 건너뜁니다.
건너뛰기 지정된 항목 수를 열거형에서 앞으로 건너뜁니다.
건너뛰기 ISyncMgrEnumItems::Next에 대한 다음 호출이 해당 요소를 반환하지 않도록 열거형의 다음 celt 요소를 건너뛰도록 열거자에게 지시합니다.
SoftwareUpdateMessageBox 애플리케이션이 업데이트되었음을 사용자에게 알리는 데 사용할 수 있는 표준 메시지 상자를 표시합니다.
SSIZETAdd 두 개의 SSIZE_T 값을 함께 추가합니다.
SSIZETMult 한 SSIZE_T 값을 다른 값으로 곱합니다.

SSIZETSub 한 SSIZE_T 값을 다른 값에서 뺍니다.
StartHandlerSync 특정 처리기에서 관리하는 모든 항목의 동기화를 시작합니다.
StartItemSync 특정 처리기에서 관리하는 지정된 항목의 동기화를 시작합니다.
StartOperation 비동기 데이터 추출이 시작 중임을 나타내기 위해 놓기 대상에 의해 호출됩니다.
StartOperations 특정 파일 작업을 수행하기 전에 호출자 구현 작업을 수행합니다.
StartProgressDialog 진행률 대화 상자를 시작합니다.
StartProgressDialog 지정된 진행률 대화 상자를 시작합니다.
StartServicing 지정된 패키지와 연결된 앱의 비 백그라운드 부분을 일시 중단 및 종료하고 패키지와 연결된 백그라운드 작업을 취소합니다.
StartSessionRedirection 지정된 사용자 세션에서 지정된 패키지에 대한 백그라운드 작업이 활성화되도록 합니다.
StartSyncAll 모든 처리기에서 관리하는 모든 항목을 동기화합니다.
StartVisualGroup 대화 상자에서 시각적 개체 그룹을 선언합니다. "add" 메서드에 대한 후속 호출은 해당 요소를 이 그룹에 추가합니다.
상태 현재 작업의 릴리스 상태 및 백그라운드 스레드 시간 제한을 설정합니다.

StgMakeUniqueName 템플릿에서 스트림 또는 스토리지 개체의 고유한 이름을 만듭니다.
중지 진행률 대화 상자를 중지합니다.
StopHandlerSync 지정된 처리기의 동기화를 중지합니다.
StopItemSync 특정 처리기에서 관리하는 지정된 항목의 동기화를 중지합니다.
StopProgressDialog 진행률 대화 상자를 중지하고 화면에서 제거합니다.
StopProgressDialog 현재 진행률 대화 상자를 중지합니다.
StopServicing StartServicing 메서드를 호출하여 시작한 이전 서비스 작업을 완료합니다.
StopSessionRedirection 지정된 패키지에 대한 백그라운드 작업의 리디렉션을 중지합니다.
StopSharingPrinters 컴퓨터에 연결된 모든 로컬 공유 프린터 공유를 중지합니다.
StopSyncAll 모든 처리기에서 관리하는 모든 항목의 동기화를 중지합니다.
StopWatchFlush StopWatchFlush를 변경하거나 사용할 수 없습니다.
StopWatchMode StopWatchMode를 변경하거나 사용할 수 없습니다.
StrCatBuffA

한 문자열의 문자를 복사하여 다른 문자열의 끝에 추가합니다. (ANSI)
StrCatBuffW 한 문자열의 문자를 복사하여 다른 문자열의 끝에 추가합니다. (유니코드)
StrCatChainW 두 유니코드 문자열을 연결합니다. 동일한 버퍼에 반복 연결이 필요한 경우 사용됩니다.
StrCatW 한 문자열을 다른 문자열에 추가합니다.
StrChrA 문자열에서 지정된 문자와 일치하는 문자가 처음 나타나는지 검색합니다. 비교 시 대/소문자를 구분합니다. (ANSI)
StrChrIA 문자열에서 지정된 문자와 일치하는 문자가 처음 나타나는지 검색합니다. 비교는 대/소문자를 구분하지 않습니다. (ANSI)
StrChrIW 문자열에서 지정된 문자와 일치하는 문자가 처음 나타나는지 검색합니다. 비교는 대/소문자를 구분하지 않습니다. (유니코드)
StrChrNIW 문자열에서 지정된 문자의 첫 번째 항목을 검색합니다. 비교는 대/소문자를 구분하지 않습니다.
StrChrNW 문자열에서 지정된 문자의 첫 번째 항목을 검색합니다. 비교 시 대/소문자를 구분합니다.
StrChrW 문자열에서 지정된 문자와 일치하는 문자가 처음 나타나는지 검색합니다. 비교 시 대/소문자를 구분합니다. (유니코드)
StrCmpCA ASCII(C 런타임) 데이터 정렬 규칙을 사용하여 문자열을 비교합니다. 비교 시 대/소문자를 구분합니다. (ANSI)
StrCmpCW

ASCII(C 런타임) 데이터 정렬 규칙을 사용하여 문자열을 비교합니다. 비교 시 대/소문자를 구분합니다. (유니코드)
StrCmpICA ASCII(C 런타임) 데이터 정렬 규칙을 사용하여 두 문자열을 비교합니다. 비교는 대/소문자를 구분하지 않습니다. (ANSI)
StrCmpICW ASCII(C 런타임) 데이터 정렬 규칙을 사용하여 두 문자열을 비교합니다. 비교는 대/소문자를 구분하지 않습니다. (유니코드)
StrCmpIW 두 문자열을 비교하여 동일한지 확인합니다. 비교는 대/소문자를 구분하지 않습니다.
StrCmpLogicalW 두 유니코드 문자열을 비교합니다. 문자열의 숫자는 텍스트가 아닌 숫자 콘텐츠로 간주됩니다. 이 테스트는 대/소문자를 구분하지 않습니다.
StrCmpNA 두 문자열의 시작 부분에서 지정된 수의 문자를 비교하여 동일한지 확인합니다. 비교 시 대/소문자를 구분합니다. StrNCmp 매크로는 이름에서만 이 함수와 다릅니다. (ANSI)
StrCmpNCA ASCII(C 런타임) 데이터 정렬 규칙을 사용하여 두 문자열의 시작 부분에서 지정된 문자 수를 비교합니다. 비교 시 대/소문자를 구분합니다. (ANSI)
StrCmpNCW ASCII(C 런타임) 데이터 정렬 규칙을 사용하여 두 문자열의 시작 부분에서 지정된 문자 수를 비교합니다. 비교 시 대/소문자를 구분합니다. (유니코드)
StrCmpNIA 두 문자열의 시작 부분에서 지정된 수의 문자를 비교하여 동일한지 확인합니다. 비교는 대/소문자를 구분하지 않습니다. StrNCmpI 매크로는 이름에서만 이 함수와 다릅니다. (ANSI)
StrCmpNICA ASCII(C 런타임) 데이터 정렬 규칙을 사용하여 두 문자열의 시작 부분에서 지정된 문자 수를 비교합니다. 비교는 대/소문자를 구분하지 않습니다. (ANSI)
StrCmpNICW

ASCII(C 런타임) 데이터 정렬 규칙을 사용하여 두 문자열의 시작 부분에서 지정된 문자 수를 비교합니다. 비교는 대/소문자를 구분하지 않습니다. (유니코드)
StrCmpNIW 두 문자열의 시작 부분에서 지정된 수의 문자를 비교하여 동일한지 확인합니다. 비교는 대/소문자를 구분하지 않습니다. StrNCmpI 매크로는 이름에서만 이 함수와 다릅니다. (유니코드)
StrCmpNW 두 문자열의 시작 부분에서 지정된 수의 문자를 비교하여 동일한지 확인합니다. 비교 시 대/소문자를 구분합니다. StrNCmp 매크로는 이름에서만 이 함수와 다릅니다. (유니코드)
StrCmpW 두 문자열을 비교하여 동일한지 확인합니다. 비교 시 대/소문자를 구분합니다.
StrCpyNW 한 문자열의 시작 부분에서 다른 문자열로 지정된 수의 문자를 복사합니다. 참고 이 함수나 StrNCpy 매크로는 사용하지 마세요.
StrCpyW 한 문자열을 다른 문자열에 복사합니다. (StrCpyW)
StrCSpnA 문자열에서 문자 그룹의 첫 번째 항목을 검색합니다. 검색 방법은 대/소문자를 구분하며 종료되는 NULL 문자는 검색 패턴 일치 내에 포함됩니다. (ANSI)
StrCSpnIA 문자열에서 문자 그룹의 첫 번째 항목을 검색합니다. 검색 방법은 대/소문자를 구분하지 않으며 종료되는 NULL 문자는 검색 패턴 일치 내에 포함됩니다. (ANSI)
StrCSpnIW 문자열에서 문자 그룹의 첫 번째 항목을 검색합니다. 검색 방법은 대/소문자를 구분하지 않으며 종료되는 NULL 문자는 검색 패턴 일치 내에 포함됩니다. (유니코드)
StrCSpnW 문자열에서 문자 그룹의 첫 번째 항목을 검색합니다. 검색 방법은 대/소문자를 구분하며 종료되는 NULL 문자는 검색 패턴 일치 내에 포함됩니다. (유니코드)
StrDupA 문자열을 복제합니다. (ANSI)

StrDupW 문자열을 복제합니다. (유니코드)
StrFormatByteSize64A 숫자 값을 크기에 따라 크기 값(바이트, 킬로바이트, 메가바이트 또는 기가바이트)으로 표현되는 숫자를 나타내는 문자열로 변환합니다.
StrFormatByteSizeA 숫자 값을 크기에 따라 크기 값(바이트, 킬로바이트, 메가바이트 또는 기가바이트)으로 표현되는 숫자를 나타내는 문자열로 변환합니다. 한 매개 변수 형식의 StrFormatByteSizeW와 다릅니다.
StrFormatByteSizeEx 크기에 따라 숫자 값을 바이트, 킬로바이트, 메가바이트 또는 기가바이트 단위의 숫자를 나타내는 문자열로 변환합니다.
StrFormatByteSizeW 숫자 값을 크기에 따라 크기 값(바이트, 킬로바이트, 메가바이트 또는 기가바이트)으로 표현되는 숫자를 나타내는 문자열로 변환합니다. 한 매개 변수 형식의 StrFormatByteSizeA와 다릅니다.
StrFormatKBSizeA 숫자 값을 크기 값(킬로바이트)으로 표현되는 숫자를 나타내는 문자열로 변환합니다. (ANSI)
StrFormatKBSizeW 숫자 값을 크기 값(킬로바이트)으로 표현되는 숫자를 나타내는 문자열로 변환합니다. (유니코드)
StrFromTimeIntervalA 밀리초 단위로 지정된 시간 간격을 문자열로 변환합니다. (ANSI)
StrFromTimeIntervalW 밀리초 단위로 지정된 시간 간격을 문자열로 변환합니다. (유니코드)
StrIsIntlEqualA 두 문자열의 시작 부분에서 지정된 수의 문자를 비교하여 문자가 같은지 확인합니다. (ANSI)
StrIsIntlEqualW 두 문자열의 시작 부분에서 지정된 수의 문자를 비교하여 문자가 같은지 확인합니다. (유니코드)

StrNCatA 한 문자열의 시작부터 다른 문자열의 끝까지 지정된 수의 문자를 추가합니다. (ANSI)
StrNCatW 한 문자열의 시작부터 다른 문자열의 끝까지 지정된 수의 문자를 추가합니다. (유니코드)
StrPBrkA 문자열에서 지정된 버퍼에 포함된 문자의 첫 번째 발생을 검색합니다. 이 검색에는 종료 null 문자가 포함되지 않습니다. (ANSI)
StrPBrkW 문자열에서 지정된 버퍼에 포함된 문자의 첫 번째 발생을 검색합니다. 이 검색에는 종료 null 문자가 포함되지 않습니다. (유니코드)
StrRChrA 문자열에서 지정된 문자의 마지막 항목을 검색합니다. 비교 시 대/소문자를 구분합니다. (ANSI)
StrRChrIA 문자열에서 지정된 문자의 마지막 항목을 검색합니다. 비교는 대/소문자를 구분하지 않습니다. (ANSI)
StrRChrIW 문자열에서 지정된 문자의 마지막 항목을 검색합니다. 비교는 대/소문자를 구분하지 않습니다. (유니코드)
StrRChrW 문자열에서 지정된 문자의 마지막 항목을 검색합니다. 비교 시 대/소문자를 구분합니다. (유니코드)
StrRetToBSTR 문자열을 포함하거나 가리키는 IShellFolder::GetDisplayNameOf에서 반환된 STRRET 구조체를 수락하고 해당 문자열을 BSTR로 반환합니다.
StrRetToBufA IShellFolder::GetDisplayNameOf에서 반환된 STRRET 구조를 문자열로 변환하고 결과를 버퍼에 배치합니다. (ANSI)
StrRetToBufW IShellFolder::GetDisplayNameOf에서 반환된 STRRET 구조를 문자열로 변환하고 결과를 버퍼에 배치합니다. (유니코드)

StrRetToStrA IShellFolder::GetDisplayNameOf에서 반환된 STRRET 구조를 사용하고 표시 이름을 포함하는 할당된 문자열에 대한 포인터를 반환합니다. (ANSI)
StrRetToStrW IShellFolder::GetDisplayNameOf에서 반환된 STRRET 구조를 사용하고 표시 이름을 포함하는 할당된 문자열에 대한 포인터를 반환합니다. (유니코드)
StrRStrIA 문자열 내에서 지정된 부분 문자열의 마지막 항목을 검색합니다. 비교는 대/소문자를 구분하지 않습니다. (ANSI)
StrRStrIW 문자열 내에서 지정된 부분 문자열의 마지막 항목을 검색합니다. 비교는 대/소문자를 구분하지 않습니다. (유니코드)
StrSpnA 지정된 버퍼에 포함된 문자로 구성된 문자열 내의 부분 문자열 길이를 가져옵니다. (ANSI)
StrSpnW 지정된 버퍼에 포함된 문자로 구성된 문자열 내의 부분 문자열 길이를 가져옵니다. (유니코드)
StrStrA 문자열 내에서 첫 번째 부분 문자열을 찾습니다. 비교 시 대/소문자를 구분합니다. (StrStrA)
StrStrIA 문자열 내에서 첫 번째 부분 문자열을 찾습니다. 비교는 대/소문자를 구분하지 않습니다. (ANSI)
StrStrIW 문자열 내에서 첫 번째 부분 문자열을 찾습니다. 비교는 대/소문자를 구분하지 않습니다. (유니코드)
StrStrNIW 문자열 내에서 첫 번째 부분 문자열을 찾습니다. 비교는 대/소문자를 구분하지 않습니다.
StrStrNW 문자열 내에서 첫 번째 부분 문자열을 찾습니다. 비교 시 대/소문자를 구분합니다. (StrStrNW)

StrStrW 문자열 내에서 첫 번째 부분 문자열을 찾습니다. 비교 시 대/소문자를 구분합니다. (StrStrW)
StrToInt64ExA 10진수 또는 16진수 값을 나타내는 문자열을 64비트 정수로 변환합니다. (ANSI)
StrToInt64ExW 10진수 또는 16진수 값을 나타내는 문자열을 64비트 정수로 변환합니다. (유니코드)
StrToIntA 10진수 값을 나타내는 문자열을 정수로 변환합니다. StrToLong 매크로는 이 함수와 동일합니다. (ANSI)
StrToIntExA 10진수 또는 16진수를 나타내는 문자열을 정수로 변환합니다. (ANSI)
StrToIntExW 10진수 또는 16진수를 나타내는 문자열을 정수로 변환합니다. (유니코드)
StrToIntW 10진수 값을 나타내는 문자열을 정수로 변환합니다. StrToLong 매크로는 이 함수와 동일합니다. (유니코드)
StrTrimA 문자열에서 지정된 선행 및 후행 문자를 제거합니다. (ANSI)
StrTrimW 문자열에서 지정된 선행 및 후행 문자를 제거합니다. (유니코드)
SUBCLASSPROC RemoveWindowSubclass 및 SetWindowSubclass에서 사용하는 콜백 함수의 프로토타입을 정의합니다.
SubStreamFailure 보조 스트림과 관련된 오류가 있고 사용자 상호 작용이 필요할 때 호출됩니다.
SupportsResource 항목이 지정된 리소스를 지원하는지 여부를 검색합니다.
SupportsSelection

호출자가 해당 선택을 지원하는지 여부를 결정합니다.
일시 중단 현재 실행 중인 패키지의 프로세스를 일시 중단합니다.
일시 중단 작업이 일시 중단되도록 요청합니다.
동기화 처리기의 동기화 항목 선택 항목의 동기화를 시작합니다.
동기화 사용자가 동기화할 등록된 애플리케이션을 선택한 후 선택한 각 그룹에 대해 동기화 관리자가 한 번 호출합니다.
SynchronizeCompleted Synchronize 메서드가 완료되면 애플리케이션에서 호출됩니다.
TerminateAllProcesses 지정된 패키지에 대한 모든 프로세스를 종료합니다.
ThumbBarAddButtons 작업 표시줄 단추 플라이아웃에서 창의 축소판 그림 이미지에 지정된 단추 집합이 있는 축소판 그림 도구 모음을 추가합니다.
ThumbBarSetImageList 작업 표시줄 단추 플라이아웃에서 창의 축소판 그림 이미지에 포함된 도구 모음에 대한 단추 이미지가 포함된 이미지 목록을 지정합니다.
ThumbBarUpdateButtons 창의 현재 상태에 필요한 대로 썸네일 도구 모음에서 단추를 표시, 사용, 사용 안 함 또는 숨깁니다. 썸네일 도구 모음은 작업 표시줄 단추 플라이아웃에 있는 창의 썸네일 이미지에 포함된 도구 모음입니다.
타이머 진행률 대화 상자 타이머를 0으로 다시 설정합니다.
TranscodeImage 이미지를 JPEG 또는 BMP(비트맵) 이미지 형식으로 변환합니다.

번역하기 브라우저에서 URL 주소의 프로토콜을 확인할 수 없는 경우 브라우저에서 호출됩니다.
TranslateAccelerator 미리 보기 처리기가 실행 중인 프로세스의 메시지 펌프에서 전달된 키 입력을 처리하도록 미리 보기 처리기를 지시합니다.
TranslateAccelerator 미리 보기 처리기에서 전달된 바로 가기 키를 처리하도록 호스트에 지시합니다.
TranslateAccelerator 네임스페이스 확장의 보기에 포커스가 있을 때 바로 가기 키(가속기) 키 스트로크를 변환합니다.
TranslateAcceleratorGlobal 입력 개체가 UI에서 활성화되지 않은 경우에도 키보드에 응답할 수 있도록 전역 가속기를 처리합니다.
TranslateAcceleratorIO 개체가 키보드 가속기를 처리할 수 있도록 합니다.
TranslateAcceleratorSB 보기가 활성 상태인 동안 브라우저 프레임에 대한 가속기 키 입력을 변환합니다.
TranslateMenuMessage COM(Component Object Model) 개체에 대한 메시지를 변환합니다.
TranslateURLA 지정된 URL 문자열에 일반 번역을 적용하여 새 URL 문자열을 만듭니다. (ANSI)
TranslateURLW 지정된 URL 문자열에 일반 번역을 적용하여 새 URL 문자열을 만듭니다. (유니코드)
TranslateWithSearchContext 브라우저에서 URL 주소의 프로토콜을 확인할 수 없는 경우 브라우저에서 호출됩니다. 이 메서드는 검색 컨텍스트를 사용하여 프로토콜을 확인합니다.
여행하다 Deprecated. 현재 항목의 위치를 기준으로 이동 로그의 이동 항목으로 이동합니다.

TreeAdvise 클라이언트가 컨트롤에 등록할 수 있도록 합니다.
TreeUnadvise 클라이언트가 컨트롤을 사용하여 등록을 취소할 수 있도록 합니다.
TriggerEdgeGesture 스푸핑된 에지 제스처 이벤트를 호출자 스레드의 프록시 코어 창으로 보냅니다. 이 제스처는 앱이 지원하는 경우 앱의 앱 표시줄을 전환합니다. 호출자는 에지 제스처를 트리거한 입력 유형을 지정할 수 있습니다.
UIActivate 셸 뷰 자체에 의해 발생하지 않는 이벤트에 의해 보기 창의 활성화 상태가 변경될 때 호출됩니다. 예를 들어 트리에 포커스가 있을 때 TAB 키를 누르면 보기에 포커스가 지정되어야 합니다.
UIActivateIO UI는 개체를 활성화하거나 비활성화합니다.
UInt8Add UINT8 형식의 두 값을 추가합니다.
UInt8Mult UINT8 형식의 두 값을 곱합니다.
UInt8Sub UINT8 형식의 한 값을 다른 값에서 뺍니다.
UInt8ToChar UINT8 형식의 값을 CHAR 형식의 값으로 변환합니다.
UInt8ToInt8 UINT8 형식의 값을 INT8 형식의 값으로 변환합니다.
UIntAdd UINT 형식의 두 값을 추가합니다.
UIntMult UINT 형식의 한 값을 다른 값으로 곱합니다.

UIntPtrAdd UINT_PTR 형식의 두 값을 추가합니다.
UIntPtrMult UINT_PTR 형식의 한 값을 다른 값으로 곱합니다.
UIntPtrSub UINT_PTR 형식의 한 값을 다른 값에서 뺍니다.
UIntPtrToChar 형식 UINT_PTR 값을 CHAR 형식의 값으로 변환합니다.
UIntPtrToInt SIZE_T 형식의 값을 INT 형식의 값으로 변환합니다.
UIntPtrToInt16 UINT_PTR 형식의 값을 INT16 형식의 값으로 변환합니다.
UIntPtrToInt8 UINT_PTR 형식의 값을 INT8 형식의 값으로 변환합니다.
UIntPtrToIntPtr 형식 UINT_PTR 값을 INT_PTR 형식의 값으로 변환합니다.
UIntPtrToLong size_t 형식의 값을 LONG 형식의 값으로 변환합니다.
UIntPtrToLongLong UINT_PTR 형식의 값을 LONGLONG 형식의 값으로 변환합니다.
UIntPtrToLongPtr 형식 UINT_PTR 값을 형식 LONG_PTR 값으로 변환합니다.
UIntPtrToShort UINT_PTR 형식의 값을 SHORT 형식의 값으로 변환합니다.
UIntPtrToUChar

UINT_PTR 형식의 값을 UCHAR 형식의 값으로 변환합니다.
UIntPtrToUInt UINT_PTR 형식의 값을 UINT 형식의 값으로 변환합니다.
UIntPtrToUInt16 UINT_PTR 형식의 값을 UINT16 형식의 값으로 변환합니다.
UIntPtrToUInt8 UINT_PTR 형식의 값을 UINT8 형식의 값으로 변환합니다.
UIntPtrToULong UINT_PTR 형식의 값을 ULONG 형식의 값으로 변환합니다.
UIntPtrToUShort UINT_PTR 형식의 값을 USHORT 형식의 값으로 변환합니다.
UIntSub UINT 형식의 한 값을 다른 값에서 뺍니다.
UIntToChar UINT 형식의 값을 CHAR 형식의 값으로 변환합니다.
UIntToInt UINT 형식의 값을 INT 형식의 값으로 변환합니다.
UIntToInt8 UINT 형식의 값을 INT8 형식의 값으로 변환합니다.
UIntToIntPtr UINT 형식의 값을 INT_PTR 형식의 값으로 변환합니다.
UIntToLong UINT 형식의 값을 LONG 형식의 값으로 변환합니다.
UIntToLongPtr UINT 형식의 값을 LONG_PTR 형식의 값으로 변환합니다.

UIntToShort UINT 형식의 값을 SHORT 형식의 값으로 변환합니다.
UIntToUChar UINT 형식의 값을 UCHAR 형식의 값으로 변환합니다.
UIntToUInt8 UINT 형식의 값을 UINT8 형식의 값으로 변환합니다.
UIntToUShort UINT 형식의 값을 USHORT 형식의 값으로 변환합니다.
ULongAdd ULONG 형식의 두 값을 추가합니다.
ULongLongAdd SIZE_T 형식의 두 값을 추가합니다.
ULongLongMult size_t 형식의 한 값을 다른 값으로 곱합니다. S
ULongLongSub SIZE_T 형식의 한 값을 다른 값에서 뺍니다.
ULongLongToChar ULONGLONG 형식의 값을 CHAR 형식의 값으로 변환합니다.
ULongLongToInt ULONGLONG 형식의 값을 INT 형식의 값으로 변환합니다.
ULongLongToInt8 ULONGLONG 형식의 값을 INT8 형식의 값으로 변환합니다.
ULongLongToLong ULONGLONG 형식의 값을 LONG 형식의 값으로 변환합니다.
ULongLongToLongLong

ULONGLONG 형식의 값을 INT_PTR 형식의 값으로 변환합니다.
ULongLongToLongPtr ULONGLONG 형식의 값을 LONG_PTR 형식의 값으로 변환합니다.
ULongLongToShort ULONGLONG 형식의 값을 SHORT 형식의 값으로 변환합니다.
ULongLongToUChar ULONGLONG 형식의 값을 UCHAR 형식의 값으로 변환합니다.
ULongLongToUInt ULONGLONG 형식의 값을 UINT 형식의 값으로 변환합니다.
ULongLongToUInt8 ULONGLONG 형식의 값을 UINT8 형식의 값으로 변환합니다.
ULongLongToUIntPtr ULONGLONG 형식의 값을 UINT_PTR 형식의 값으로 변환합니다.
ULongLongToULong ULONGLONG 형식의 값을 ULONG 형식의 값으로 변환합니다.
ULongLongToULongPtr ULONGLONG 형식의 값을 ULONG_PTR 형식의 값으로 변환합니다.
ULongLongToUShort ULONGLONG 형식의 값을 USHORT 형식의 값으로 변환합니다.
ULongMult ULONG 형식의 한 값을 다른 값으로 곱합니다.
ULongPtrAdd ULONG_PTR 형식의 두 값을 추가합니다.
ULongPtrMult ULONG_PTR 형식의 한 값을 다른 값으로 곱합니다.

ULongPtrSub ULONG_PTR 형식의 한 값을 다른 값에서 뺍니다.
ULongPtrToChar 형식 ULONG_PTR 값을 CHAR 형식의 값으로 변환합니다.
ULongPtrToInt 형식 size_t 값을 INT 형식의 값으로 변환합니다.
ULongPtrToInt8 ULONG_PTR 형식의 값을 INT8 형식의 값으로 변환합니다.
ULongPtrToIntPtr 형식 ULONG_PTR 값을 INT_PTR 형식의 값으로 변환합니다.
ULongPtrToLong ULONG_PTR 형식의 값을 LONG 형식의 값으로 변환합니다.
ULongPtrToLongLong SIZE_T 형식의 값을 INT64 형식의 값으로 변환합니다.
ULongPtrToLongPtr 형식 ULONG_PTR 값을 LONG_PTR 형식의 값으로 변환합니다.
ULongPtrToShort ULONG_PTR 형식의 값을 SHORT 형식의 값으로 변환합니다.
ULongPtrToUChar ULONG_PTR 형식의 값을 UCHAR 형식의 값으로 변환합니다.
ULongPtrToUInt ULONG_PTR 형식의 값을 UINT 형식의 값으로 변환합니다.
ULongPtrToUInt8 ULONG_PTR 형식의 값을 UINT8 형식의 값으로 변환합니다.
ULongPtrToUIntPtr

형식 ULONG_PTR 값을 UINT_PTR 형식의 값으로 변환합니다.
ULongPtrToULong ULONG_PTR 형식의 값을 ULONG 형식의 값으로 변환합니다.
ULongPtrToUShort ULONG_PTR 형식의 값을 USHORT 형식의 값으로 변환합니다.
ULongSub 다른 값에서 ULONG 형식의 값을 뺍니다.
ULongToChar ULONG 형식의 값을 CHAR 형식의 값으로 변환합니다.
ULongToInt ULONG 형식의 값을 INT 형식의 값으로 변환합니다.
ULongToInt8 ULONG 형식의 값을 INT8 형식의 값으로 변환합니다.
ULongToIntPtr ULONG 형식의 값을 INT_PTR 형식의 값으로 변환합니다.
ULongToLong ULONG 형식의 값을 LONG 형식의 값으로 변환합니다.
ULongToLongPtr ULONG 형식의 값을 LONG_PTR 형식의 값으로 변환합니다.
ULongToShort ULONG 형식의 값을 SHORT 형식의 값으로 변환합니다.
ULongToUChar ULONG 형식의 값을 UCHAR 형식의 값으로 변환합니다.
ULongToUInt ULONG 형식의 값을 UINT 형식의 값으로 변환합니다.

ULongToUInt8 ULONG 형식의 값을 UINT8 형식의 값으로 변환합니다.
ULongToUIntPtr ULONG 형식의 값을 UINT_PTR 형식의 값으로 변환합니다.
ULongToUShort ULONG 형식의 값을 USHORT 형식의 값으로 변환합니다.
Unadvise Advise를 사용하여 이전에 설정한 연결을 취소합니다.
Unadvise 권고 연결을 종료합니다. (IExplorerBrowser.Unadvise)
Unadvise IFileDialog::Advise 메서드를 통해 연결된 이벤트 처리기를 제거합니다.
Unadvise IFileOperation::Advise를 통해 이전에 설정된 권고 연결을 종료합니다.
Unadvise 더 이상 알림을 받지 않도록 앱의 입력 창 처리기 개체를 등록 취소합니다.
Unadvise 권고 연결을 종료합니다. (ITransferDestination.Unadvise)
Unadvise 권고 연결을 종료합니다. (ITransferSource.Unadvise)
UnAdvise 로그온 UI 또는 자격 증명 UI에서 이벤트 콜백이 더 이상 허용되지 않는다고 자격 증명 공급자에게 조언하는 데 사용됩니다.
UnAdvise 로그온 UI 또는 자격 증명 UI에서 이벤트 콜백이 더 이상 허용되지 않는다는 것을 자격 증명에 알려 주는 데 사용됩니다.

UngroupChildFromParent ISuspensionDependencyManager::UngroupChildFromParent 메서드
언로드 미리 보기 처리기에 미리 보기 렌더링을 중단하고 초기화 중에 전달된 항목에 따라 할당된 모든 리소스를 해제하도록 지시합니다.
UnloadUserProfile LoadUserProfile 함수에 의해 로드된 사용자의 프로필을 언로드합니다. 호출자는 컴퓨터에 대한 관리 권한이 있어야 합니다. 자세한 내용은 LoadUserProfile 함수의 설명 섹션을 참조하세요.
UnregisterAppStateChangeNotification RegisterAppStateChangeNotification을 통해 등록된 변경 알림을 취소합니다.
UnregisterFolder 알려진 폴더를 레지스트리에서 제거하면 알려진 폴더 시스템에 알 수 없습니다. 이 메서드는 폴더 자체를 제거하지 않습니다.
UnregisterForPackageStateChanges RegisterForPackageStateChanges에 대한 이전 호출과 연결된 패키지 상태 변경 알림 수신을 중지합니다.
UnregisterScaleChangeEvent RegisterScaleChangeEvent를 통해 등록된 크기 조정 변경 이벤트에 대한 등록을 취소합니다. 이 함수는 RevokeScaleChangeNotifications를 대체합니다.
UnregisterSyncMgrHandler 등록에서 처리기의 CLSID(클래스 식별자)를 제거합니다. 더 이상 동기화할 항목이 없으면 처리기가 이를 호출해야 합니다.
UnregisterTab 애플리케이션에서 해당 탭 또는 문서를 닫을 때 애플리케이션의 미리 보기 그룹에서 축소판 그림을 제거합니다.
예약 취소 애플리케이션 게시자가 게시한 애플리케이션의 설치를 취소합니다.
Update 이 항목의 캐시된 정보가 업데이트되었는지 확인합니다.

Update 제공된 IBindCtx에서 지정한 매개 변수를 기반으로 제공된 자식 ITEMIDLIST를 업데이트합니다.
Update Deprecated. 이동 항목을 업데이트합니다.
UpdateAll 프로그래밍 방식으로 모든 항목에 대한 업데이트를 시작합니다.
UpdateBackForwardState Deprecated. 브라우저의 뒤로 및 앞으로 단추의 상태를 업데이트합니다.
UpdateConflicts 동기화 센터에 특정 처리기 또는 항목에 대한 충돌이 추가되었음을 알립니다.
UpdateEntry Deprecated. 보류 중인 탐색을 준비하기 위해 현재 항목의 브라우저 상태를 저장합니다.
UpdateEvents 특정 처리기 또는 항목에 대한 이벤트가 추가되었음을 동기화 센터에 알립니다.
UpdateExternal Deprecated. IHlinkFrame을 통해 현재 프로시저에서 시작된 항목을 업데이트합니다.
UpdateHandler 동기화 센터에 처리기가 관리하는 항목을 다시 열거하도록 지시하거나 처리기의 속성이 변경되었다는 사실을 알립니다.
UpdateHandlerCollection 동기화 센터에 처리기 컬렉션을 다시 열거하도록 지시하거나 처리기 컬렉션의 처리기 속성이 변경되었다는 사실을 알릴 수 있습니다.
UpdateItem 동기화 항목의 속성이 변경되었다는 것을 동기화 센터에 알립니다.
UpdateItems 프로그래밍 방식으로 지정된 항목에 대한 업데이트를 시작합니다.

UpdateLocations 현재 진행률 대화 상자에서 원본과 대상을 나타내는 텍스트 요소를 지정하기 위해 호출됩니다.
UpdateObject UpdateObject를 변경하거나 사용할 수 없습니다.
UpdateProgress 작업의 진행률을 UI로 업데이트합니다.
UpdateProgress 총 작업량과 관련하여 현재 수행된 총 작업량의 추정치를 제공합니다.
UpdateProgress 지정된 대로 현재 진행률 대화 상자를 업데이트합니다.
UpdateProgress UI에서 전송 진행 상태를 업데이트합니다.
UpdateRemoteCredential 원격 세션에서 자격 증명을 업데이트합니다.
UpdateSecureLockIcon Deprecated. BASEBROWSERDATA 구조체의 _eSecureLockIcon 멤버 값을 업데이트합니다. 이 멤버는 보안 사이트를 나타내는 아이콘을 추적하고 UI에서 아이콘 자체를 업데이트합니다.
UpdateText 작업과 연결된 설명 텍스트가 변경될 경우 호출됩니다.
UpdateTransferState 전송 상태를 업데이트합니다.
UpdateWindowList Deprecated. 창 목록에서 PIDL(항목 식별자 목록)에 대한 포인터를 업데이트하도록 브라우저에 지시합니다. 이 메서드는 탐색 후에 호출됩니다.
UrlApplySchemeA 지정된 URL 문자열에 대한 구성표를 결정하고 적절한 접두사를 가진 문자열을 반환합니다. (ANSI)

UrlApplySchemeW 지정된 URL 문자열에 대한 구성표를 결정하고 적절한 접두사를 가진 문자열을 반환합니다. (유니코드)
URLAssociationDialogA 등록되지 않은 URL 프로토콜 대화 상자를 호출합니다. (ANSI)
URLAssociationDialogW 등록되지 않은 URL 프로토콜 대화 상자를 호출합니다. (유니코드)
UrlCanonicalizeA URL 문자열을 정식 형식으로 변환합니다. (ANSI)
UrlCanonicalizeW URL 문자열을 정식 형식으로 변환합니다. (유니코드)
UrlCombineA 상대 URL 및 해당 기준이 제공되면 정식 형식으로 URL을 반환합니다. (ANSI)
UrlCombineW 상대 URL 및 해당 기준이 제공되면 정식 형식으로 URL을 반환합니다. (유니코드)
UrlCompareA 두 URL 문자열을 대/소문자를 구분하여 비교합니다. (ANSI)
UrlCompareW 두 URL 문자열을 대/소문자를 구분하여 비교합니다. (유니코드)
UrlCreateFromPathA Microsoft MS-DOS 경로를 정식 URL로 변환합니다. (ANSI)
UrlCreateFromPathW Microsoft MS-DOS 경로를 정식 URL로 변환합니다. (유니코드)
UrlEscapeA 인터넷을 통해 전송하는 동안 변경될 수 있는 URL의 문자 또는 서로게이트 쌍("안전하지 않은" 문자)을 해당 이스케이프 시퀀스로 변환합니다. (ANSI)

UrlEscapeSpaces 공백 문자를 해당 이스케이프 시퀀스로 변환하는 매크로입니다.
UrlEscapeW 인터넷을 통해 전송하는 동안 변경될 수 있는 URL의 문자 또는 서로게이트 쌍("안전하지 않은" 문자)을 해당 이스케이프 시퀀스로 변환합니다. (유니코드)
UrlFixupW UrlFixupW를 변경하거나 사용할 수 없습니다.
UrlGetLocationA URL에서 위치를 검색합니다. (ANSI)
UrlGetLocationW URL에서 위치를 검색합니다. (유니코드)
UrlGetPartA URL 문자열을 수락하고 해당 URL의 지정된 부분을 반환합니다. (ANSI)
UrlGetPartW URL 문자열을 수락하고 해당 URL의 지정된 부분을 반환합니다. (유니코드)
UrlHashA URL 문자열을 해시합니다. (ANSI)
UrlHashW URL 문자열을 해시합니다. (유니코드)
UrlIsA URL이 지정된 형식인지 여부를 테스트합니다. (ANSI)
UrlIsFileUrlA URL을 테스트하여 파일 URL인지 확인합니다. (ANSI)
UrlIsFileUrlW URL을 테스트하여 파일 URL인지 확인합니다. (유니코드)

UrlIsNoHistoryA URL이 브라우저가 일반적으로 탐색 기록에 포함하지 않는 URL인지 여부를 반환합니다. (ANSI)
UrlIsNoHistoryW URL이 브라우저가 일반적으로 탐색 기록에 포함하지 않는 URL인지 여부를 반환합니다. (유니코드)
UrlIsOpaqueA URL이 불투명한지 여부를 반환합니다. (ANSI)
UrlIsOpaqueW URL이 불투명한지 여부를 반환합니다. (유니코드)
UrlIsW URL이 지정된 형식인지 여부를 테스트합니다. (유니코드)
UrlUnescapeA 이스케이프 시퀀스를 다시 일반 문자로 변환합니다. (ANSI)
UrlUnescapeInPlace 이스케이프 시퀀스를 다시 일반 문자로 변환하고 원래 문자열을 덮어씁니다.
UrlUnescapeW 이스케이프 시퀀스를 다시 일반 문자로 변환합니다. (유니코드)
UShortAdd USHORT 형식의 두 값을 추가합니다.
UShortMult USHORT 형식의 한 값을 다른 값으로 곱합니다.
UShortSub USHORT 형식의 한 값을 다른 값에서 뺍니다.
UShortToChar USHORT 형식의 값을 CHAR 형식의 값으로 변환합니다.
UShortToInt8

USHORT 형식의 값을 INT8 형식의 값으로 변환합니다.
UShortToShort USHORT 형식의 값을 SHORT 형식의 값으로 변환합니다.
UShortToUChar USHORT 형식의 값을 UCHAR 형식의 값으로 변환합니다.
UShortToUInt8 USHORT 형식의 값을 UINT8 형식의 값으로 변환합니다.
v_CheckZoneCrossing Deprecated. 한 페이지에서 다른 페이지로 이동할 때 영역 교차의 유효성을 검사하기 위해 기본 클래스에서 호출됩니다.
v_GetViewStream Deprecated. 뷰 상태를 로드하거나 저장하는 데 사용되는 스트림을 반환합니다.
v_MayGetNextToolbarFocus Deprecated. TranslateAcceleratorSB를 통해 가속기를 번역하고 보기와 브라우저 도구 모음 간의 포커스 주기를 확인할 때 사용됩니다.
v_MayTranslateAccelerator Deprecated. 기본 클래스에 키보드 니모닉 번역을 진행하도록 지시하기 위해 파생 클래스에서 호출됩니다.
v_ShowHideChildWindows Deprecated. 파생 클래스에서 크기 조정 이벤트 후 자식 창을 업데이트할 수 있습니다.
걷다 지정된 루트에서 지정된 깊이까지 네임스페이스의 재귀 연습을 시작합니다.
WalkComplete 네임스페이스 워크 중에 수집된 데이터를 제거합니다.
어느 플랫폼 어떤Platform을 변경하거나 사용할 수 없습니다.
Win32DeleteFile

Win32DeleteFile을 변경하거나 사용할 수 없습니다.
WinHelpA Windows 도움말(Winhelp.exe)을 시작하고 애플리케이션에서 요청한 도움말의 특성을 나타내는 추가 데이터를 전달합니다. (ANSI)
WinHelpW Windows 도움말(Winhelp.exe)을 시작하고 애플리케이션에서 요청한 도움말의 특성을 나타내는 추가 데이터를 전달합니다. (유니코드)
WndProcBS Deprecated. 파생 클래스가 기본 클래스의 WinProc 함수를 호출하도록 허용합니다.
wnsprintfA 가변 길이 인수 목록을 사용하여 인수 값을 printf 스타일 형식 문자열로 반환합니다. (ANSI)
wnsprintfW 가변 길이 인수 목록을 사용하여 인수 값을 printf 스타일 형식 문자열로 반환합니다. (유니코드)
WriteAsync 정보를 비동기적으로 스트림에 씁니다. 예를 들어 셸은 파일 항목을 비동기적으로 전송할 때 이 메서드를 파일 항목에 구현합니다.
WriteCabinetState WriteCabinetState를 변경하거나 사용할 수 없습니다.
WritePropertyNPB 명명된 속성 모음에 속성을 저장합니다.
wvnsprintfA 인수 목록을 가져와 인수 값을 printf 스타일 형식 문자열로 반환합니다. (ANSI)
wvnsprintfW 인수 목록을 가져와 인수 값을 printf 스타일 형식 문자열로 반환합니다. (유니코드)
YouAreAReplica Reconcile 메서드를 통해 후속 조정의 대상이 될 수 있음을 개체에 알깁니다.

Interfaces

테이블 확장

IAccessibleObject 접근성 애플리케이션에서 사용할 수 있는 메서드를 노출합니다.
IACList 후보 문자열이 계층 구조로 구성될 때 자동 완성의 효율성을 향상시키는 메서드를 노출합니다.
IACList2 자동 완성 개체의 클라이언트가 옵션 플래그를 검색하고 설정할 수 있도록 IACList 인터페이스를 확장합니다.
IActionProgress 진행률 기반 작업이 상속할 수 있는 추상 기본 클래스를 나타냅니다.
IActionProgressDialog 진행률 대화 상자를 초기화하고 중지하는 메서드를 노출합니다.
IApplicationActivationManager 시작, 파일 및 프로토콜 확장에 대한 Windows 스토어 앱을 활성화하는 메서드를 제공합니다. 일반적으로 디버거 및 디자인 도구에서 이 인터페이스를 사용합니다.
IApplicationAssociationRegistration 특정 파일 연결 형식에 대한 기본 애플리케이션을 쿼리하고 설정하는 메서드와 특정 연결 수준에서 프로토콜을 노출합니다.
IApplicationAssociationRegistrationUI 사용자가 연결을 사용자 지정할 수 있는 고급 연결 대화 상자를 시작하는 메서드를 노출합니다.
IApplicationDesignModeSettings 개발 도구 애플리케이션이 실제 하드웨어 없이 다양한 폼 팩터에 대한 디자인 모드에서 실행되는 Windows 스토어 앱을 테스트하기 위해 네이티브 디스플레이 해상도, 디바이스 배율 인수 및 애플리케이션 뷰 상태와 같은 시스템 및 사용자 상태를 동적으로 스푸핑할 수 있도록 합니다. 또한 일반적으로 사용자가 제어하는 상태의 변경 내용을 테스트하여 다양한 시나리오에서 Windows 스토어 앱을 테스트할 수 있습니다.
IApplicationDesignModeSettings2

개발 도구 애플리케이션이 실제 하드웨어 없이 다양한 폼 팩터에 대한 디자인 모드에서 실행되는 Windows 스토어 앱을 테스트하기 위해 Windows 스토어 앱에 보고되는 네이티브 디스플레이 해상도, 디바이스 배율 인수 및 애플리케이션 보기 레이아웃과 같은 시스템 및 사용자 상태를 동적으로 제어할 수 있습니다. 또한 일반적으로 사용자가 제어하는 상태의 변경 내용을 테스트하여 다양한 시나리오에서 Windows 스토어 앱을 테스트할 수 있습니다.
IApplicationDestinations 애플리케이션이 점프 목록의 최근 또는 자주 사용하는 범주에서 하나 또는 모든 대상을 제거할 수 있도록 하는 메서드를 노출합니다.
IApplicationDocumentLists 애플리케이션이 점프 목록에서 최근 또는 자주 사용하는 범주의 콘텐츠를 검색할 수 있도록 하는 메서드를 노출합니다.
IAppPublisher 제어판에서 프로그램 추가/제거를 통해 애플리케이션을 게시하는 메서드를 노출합니다. 이 용도로 구현된 주 인터페이스입니다.
IAppVisibility 디스플레이에 유니버설 Windows 플랫폼 앱이 표시되는지 여부를 확인하는 기능을 제공합니다.
IAppVisibilityEvents 애플리케이션에서 디스플레이의 상태 변경 내용 및 시작 화면 표시 유형 변경에 대한 알림을 받을 수 있습니다.
IAssocHandler 파일 연결 대화 상자 또는 메뉴를 사용하여 작업에 대한 메서드를 노출합니다.
IAssocHandlerInvoker 연결된 애플리케이션 처리기를 호출하는 메서드를 노출합니다.
IAttachmentExecute 클라이언트 애플리케이션과 함께 작동하는 메서드를 노출하여 전자 메일 및 메시징 첨부 파일을 통해 안전한 다운로드 및 파일 교환을 제공하는 사용자 환경을 제공합니다.
IAutoComplete 자동 완성 개체(CLSID_AutoComplete)에 의해 노출됩니다. 이 인터페이스를 사용하면 애플리케이션에서 개체를 초기화, 사용 및 사용하지 않도록 설정할 수 있습니다.
IAutoComplete2

IAutoComplete를 확장합니다. 이 인터페이스를 사용하면 자동 완성 개체의 클라이언트가 자동 완성 작동 방식을 제어하는 여러 옵션을 검색하고 설정할 수 있습니다.
IAutoCompleteDropDown 편집 컨트롤에서 사용자가 입력한 문자열에 대한 가능한 완성을 포함하는 자동 완성 드롭다운 목록의 표시 상태를 클라이언트가 다시 설정하거나 쿼리할 수 있도록 하는 메서드를 노출합니다.
IBandHost 밴드를 만들고 삭제하고 가용성을 지정하는 메서드를 노출합니다.
IBandSite 밴드 개체를 제어하는 메서드를 노출합니다.
IBrowserFrameOptions 브라우저 또는 호스트가 IShellView에 지원되는 보기 동작 종류를 요청할 수 있습니다.
IBrowserService Deprecated. (IBrowserService)
IBrowserService2 Deprecated. (IBrowserService2)
IBrowserService3 Deprecated. (IBrowserService3)
IBrowserService4 Deprecated. (IBrowserService4)
ICategorizer 항목 식별자 목록에 대한 정보를 가져오는 데 사용되는 메서드를 노출합니다.
ICategoryProvider IShellFolder에 등록된 범주 목록을 노출합니다.
ICDBurn 시스템에 CD에 쓰기 위한 하드웨어, CD 기록기 디바이스의 드라이브 문자가 있는지 여부를 확인하고 프로그래밍 방식으로 CD 쓰기 세션을 시작하는 메서드를 노출합니다.

ICDBurnExt ICDBurnExt를 변경하거나 사용할 수 없습니다.
IColumnManager Windows 탐색기 세부 정보 보기에서 열을 검사하고 조작할 수 있는 메서드를 노출합니다. 각 열은 속성 이름을 지정하는 PROPERTYKEY 구조체에서 참조됩니다.
IColumnProvider Windows 탐색기 세부 정보 보기에서 사용자 지정 열을 추가하도록 설정하는 메서드를 노출합니다.
ICommDlgBrowser 셸 브라우저를 호스트할 때 사용할 공통 파일 대화 상자에 의해 노출됩니다.
ICommDlgBrowser2 ICommDlgBrowser의 기능을 확장합니다. 이 인터페이스는 셸 브라우저를 호스트할 때 공통 파일 대화 상자에 의해 노출됩니다. IShellBrowser 개체에서 QueryInterface를 호출하여 ICommDlgBrowser2에 대한 포인터를 가져올 수 있습니다.
ICommDlgBrowser3 ICommDlgBrowser2의 기능을 확장하고 셸 브라우저를 호스트할 때 공통 파일 대화 상자에서 사용합니다.
IComputerInfoChangeNotify 지원되지 않습니다. 이 인터페이스는 이후 버전의 Windows에 없을 수 있습니다. (IComputerInfoChangeNotify)
IConnectableCredentialProviderCredential IConnectableCredentialProviderCredential 개체를 연결하고 연결을 끊는 메서드를 노출합니다.
IContactManagerInterop 여러 창을 관리하는 앱에서 ContactManager 메서드에 액세스할 수 있습니다.
IContextMenu Shell 개체와 연결된 바로 가기 메뉴를 만들거나 병합하는 메서드를 노출합니다.
IContextMenu2 Shell 개체와 연결된 바로 가기(상황에 맞는) 메뉴를 만들거나 병합하는 메서드를 노출합니다. 클라이언트 개체가 소유자가 그린 메뉴 항목과 연결된 메시지를 처리할 수 있는 메서드를 추가하여 IContextMenu를 확장합니다.

IContextMenu3 Shell 개체와 연결된 바로 가기 메뉴를 만들거나 병합하는 메서드를 노출합니다. 클라이언트 개체가 소유자가 그린 메뉴 항목과 연결된 메시지를 처리하고 해당 메시지 처리의 반환 값을 수락하여 IContextMenu2를 확장할 수 있습니다.
IContextMenuCB 상황에 맞는 메뉴의 콜백을 사용하도록 설정하는 메서드를 노출합니다. 예를 들어 상승이 필요한 menuItem에 방패 아이콘을 추가합니다.
IContextMenuSite SHCreateShellFolderView를 사용하여 만든 기본 폴더 뷰에 의해 구현됩니다.
ICopyHookA 복사 후크 처리기를 만드는 메서드를 노출합니다. (ANSI)
ICopyHookW 복사 후크 처리기를 만드는 메서드를 노출합니다. (유니코드)
ICreateObject 지정된 클래스의 개체를 만드는 메서드를 노출합니다.
ICreateProcessInputs ICreatingProcess 인터페이스에서 생성 중인 프로세스의 일부 매개 변수를 변경하는 데 사용됩니다.
ICreatingProcess ShellExecuteEx 및 IContextMenu에서 호출자가 생성되는 프로세스의 일부 매개 변수를 변경할 수 있도록 하는 데 사용됩니다.
ICredentialProvider 자격 증명 공급자의 설정 및 조작에 사용되는 메서드를 노출합니다. 모든 자격 증명 공급자는 이 인터페이스를 구현해야 합니다.
ICredentialProviderCredential 자격 증명을 처리할 수 있도록 하는 메서드를 노출합니다.
ICredentialProviderCredential2 사용자의 SID(보안 식별자)를 검색하는 메서드를 추가하여 ICredentialProviderCredential 인터페이스를 확장합니다. 자격 증명은 해당 사용자와 연결되며 사용자의 타일 아래에 그룹화할 수 있습니다.

ICredentialProviderCredentialEvents 로그온 UI 또는 자격 증명 UI의 상태 또는 텍스트 변경 이벤트를 알리기 위해 자격 증명에서 사용하는 비동기 콜백 메커니즘을 제공합니다.
ICredentialProviderCredentialEvents2 Logon UI 또는 자격 증명 UI에서 필드의 일괄 업데이트를 사용하도록 설정하는 메서드를 추가하여 ICredentialProviderCredentialEvents 인터페이스를 확장합니다.
ICredentialProviderCredentialWithFieldOptions 자격 증명 공급자 프레임워크가 로그온 또는 자격 증명 UI에서 필드 옵션을 사용자 지정했는지 여부를 결정할 수 있도록 하는 메서드를 제공합니다.
ICredentialProviderEvents 자격 증명 공급자가 자격 증명 목록 또는 해당 필드의 변경 내용을 알리기 위해 사용하는 비동기 콜백 메커니즘을 제공합니다.
ICredentialProviderFilter 런타임에 사용할 수 있는 정보를 기반으로 자격 증명 공급자를 동적으로 필터링하는 데 사용됩니다.
ICredentialProviderSetUserArray 자격 증명 공급자가 로그온 또는 자격 증명 UI에 표시될 사용자 집합을 수신할 수 있도록 하는 메서드를 제공합니다.
ICredentialProviderUser 로그온 또는 자격 증명 UI에 포함된 개별 사용자의 특정 속성을 검색하는 데 사용되는 메서드를 제공합니다.
ICredentialProviderUserArray 로그온 또는 자격 증명 UI에 표시될 사용자 집합을 나타냅니다. 이 정보를 사용하면 자격 증명 공급자가 집합을 열거하여 각 사용자에 대한 속성 정보를 검색하여 필드를 채우거나 집합을 필터링할 수 있습니다.
ICurrentItem 항목에 대해 IShellFolder::BindToObject를 호출하여 가져옵니다. 항목이 이전 시간에 항목의 스냅샷을 나타내는 경우 이 인터페이스는 항목의 현재 버전을 가져옵니다.
ICurrentWorkingDirectory 클라이언트가 개체의 현재 작업 디렉터리를 검색하거나 설정할 수 있도록 하는 메서드를 노출합니다.

ICustomDestinationList 애플리케이션이 작업 표시줄에 표시할 대상 및 작업을 비롯한 사용자 지정 점프 목록을 제공할 수 있도록 하는 메서드를 노출합니다.
IDataObjectAsyncCapability 일반적으로 동기식 인터페이스가 비동기적으로 작동하도록 설정합니다.
IDataObjectProvider DataPackage가 상호 운용성을 지원하는 데 사용하는 DataPackage 개체의 IDataObject 인터페이스를 설정하거나 검색할 수 있는 메서드를 제공합니다. DataPackage 개체는 앱에서 다른 앱에 데이터를 제공하는 데 사용됩니다.
IDataTransferManagerInterop 여러 창을 관리하는 Windows 스토어 앱에서 DataTransferManager 메서드에 액세스할 수 있습니다.
IDefaultExtractIconInit 개체와 연결된 기본 아이콘을 설정하는 메서드를 노출합니다.
IDefaultFolderMenuInitialize 바로 가기 메뉴 정보를 가져와서 설정하는 데 사용되는 메서드를 제공합니다. 이 정보는 DEFCONTEXTMENU 구조를 통해 SHCreateDefaultContextMenu에 제공된 것과 동일합니다.
IDefViewFrame IUnknown 기능에만 사용됩니다. 그것은 자신의 방법이 없습니다.
IDelayedPropertyStoreFactory 속성 액세스 속도가 느린 경우 지정된 IPropertyStore 개체를 만드는 메서드를 노출합니다.
IDelegateFolder 할당하는 데 필요한 IMalloc 인터페이스를 대리자 폴더에 부여하고 항목 ID를 해제하는 메서드를 노출합니다.
IDelegateItem 항목 경로의 즉시 기본 표현을 가져오는 데 사용됩니다.
IDeskBand 밴드 개체에 대한 정보를 가져오는 데 사용됩니다.

IDeskBand2 deskband 개체에서 반투명 효과를 사용하도록 설정하고 쿼리하는 메서드를 노출합니다.
IDeskBandInfo 기본 Desk 대역폭을 가져오는 메서드를 노출합니다.
IDeskBar 데스크바 조작을 사용하도록 설정하는 메서드를 노출합니다.
IDesktopGadget 설치된 가젯을 사용자의 데스크톱에 프로그래밍 방식으로 추가할 수 있는 메서드를 노출합니다.
IDesktopWallpaper 바탕 화면 배경 화면을 관리하는 방법을 제공합니다. (IDesktopWallpaper)
IDestinationStreamFactory 속성에 변경 내용을 적용하기 전에 스트림 또는 파일을 수동으로 복사하는 메서드를 노출합니다.
IDisplayItem UI에 표시될 항목 이름과 같은 표시 속성을 가져오는 데 사용할 현재 항목의 버전을 찾는 메서드를 노출합니다.
IDockingWindow 표시, 숨기기 및 임박한 제거를 포함하여 도킹 창 개체에 변경 내용을 알리는 메서드를 노출합니다. 이 인터페이스는 Windows 탐색기 창의 테두리 공간 내에 도킹할 수 있는 창 개체에 의해 구현됩니다.
IDockingWindowFrame 프레임에 IDockingWindow 개체의 추가를 지원하는 메서드를 노출합니다. 브라우저에서 구현됩니다.
IDockingWindowSite 하나 이상의 IDockingWindow 개체에 대한 테두리 공간을 관리하는 메서드를 노출합니다. 이 인터페이스는 브라우저에서 구현되며 IOleInPlaceUIWindow 인터페이스와 유사합니다.
IDocViewSite IShellView 인터페이스에서 사이트 개체로 사용됩니다.
IDragSourceHelper

애플리케이션이 셸 끌어서 놓기 작업 중에 표시할 이미지를 지정할 수 있도록 셸에 의해 노출됩니다.
IDragSourceHelper2 IDragSourceHelper에 기능을 추가하는 메서드를 노출합니다. 이 메서드는 IDragSourceHelper 개체에 대한 끌어서 놓기 작업의 특성을 설정합니다.
IDropTargetHelper 이미지가 대상 창 위에 있는 동안 놓기 대상이 끌기 이미지를 표시할 수 있도록 하는 메서드를 노출합니다.
IDynamicHWHandler 자동 실행에서 호출됩니다. 사용자에게 표시하기 전에 등록된 처리기에 대한 동적 정보를 가져오는 메서드를 노출합니다.
IEnumAssocHandlers 특정 파일 이름 확장자와 연결된 처리기 컬렉션을 열거할 수 있는 메서드를 노출합니다.
IEnumerableView 뷰의 내용을 열거하고 열거 완료 시 콜백에서 알림을 받는 메서드를 노출합니다. 이 인터페이스를 사용하면 보기의 클라이언트가 뷰의 폴더 콘텐츠 목록을 공유하려고 할 수 있습니다.
IEnumExplorerCommand IExplorerCommandProvider에서 제공합니다. 이 인터페이스에는 명령 모음에 넣을 명령의 열거형이 포함되어 있습니다.
IEnumExtraSearch 클라이언트가 폴더에 사용 가능한 검색 개체를 결정하는 데 사용하는 표준 OLE 열거자입니다.
IEnumFullIDList Shell 폴더에 있는 항목의 PIDL(항목 식별자 목록)에 대한 포인터를 열거하는 표준 메서드 집합을 노출합니다.
IEnumIDList Shell 폴더에 있는 항목의 PIDL(항목 식별자 목록)에 대한 포인터를 열거하는 데 사용되는 표준 메서드 집합을 노출합니다.
IEnumObjects 알 수 없는 개체를 열거하는 메서드를 노출합니다.
IEnumPublishedApps

게시된 애플리케이션을 제어판의 프로그램 추가/제거에 열거하는 메서드를 노출합니다. 이 인터페이스를 노출하는 개체는 IAppPublisher::EnumApps를 통해 요청됩니다.
IEnumReadyCallback 열거가 완료되면 뷰가 구현자에게 알릴 수 있도록 하는 메서드를 노출합니다.
IEnumResources 리소스 열거형 메서드를 노출합니다.
IEnumShellItems IShellItem 인터페이스의 열거형을 노출합니다. 이 인터페이스는 일반적으로 IEnumShellItems 메서드를 호출하여 가져옵니다.
IEnumSyncMgrConflict 충돌 열거형 메서드를 노출합니다.
IEnumSyncMgrEvents 동기화 이벤트 열거형 메서드를 노출합니다.
IEnumSyncMgrSyncItems 처리기에서 관리하는 동기화 항목 개체를 열거하는 메서드를 노출합니다.
IExecuteCommand 명령 동사와 관련된 지정된 상태 또는 매개 변수를 설정하는 메서드와 해당 동사를 호출하는 메서드를 노출합니다.
IExecuteCommandApplicationHostEnvironment 애플리케이션이 호스트가 데스크톱 또는 몰입형 모드인지를 확인할 수 있는 단일 메서드를 제공합니다.
IExecuteCommandHost IExplorerCommand 기반 Shell 동사 처리기가 애플리케이션이 호출된 호스트 구성 요소의 UI 모드를 쿼리할 수 있도록 하는 메서드를 제공합니다.
IExpDispSupport Deprecated. 속성 검색, 키보드 가속기 변환 및 특정 이벤트에 대한 연결 지점의 결정을 허용하는 메서드를 노출합니다. (IExpDispSupport)
IExpDispSupportXP

Deprecated. 속성 검색, 키보드 가속기 변환 및 특정 이벤트에 대한 연결 지점의 결정을 허용하는 메서드를 노출합니다. (IExpDispSupportXP)
IExplorerBrowser IExplorerBrowser는 탐색하거나 데이터 개체의 보기를 호스트할 수 있는 브라우저 개체입니다. 완전한 기능을 갖춘 브라우저 개체로서 자동 이동 로그도 지원합니다.
IExplorerBrowserEvents 탐색기 브라우저 탐색 및 생성 이벤트 보기 알림에 대한 메서드를 노출합니다.
IExplorerCommand 명령 모양을 얻거나, 하위 명령을 열거하거나, 명령을 호출하는 메서드를 노출합니다.
IExplorerCommandProvider 탐색기 명령 및 명령 열거자를 만드는 메서드를 노출합니다.
IExplorerCommandState 명령 상태를 검색할 수 있는 단일 메서드를 노출합니다.
IExplorerPaneVisibility IShellFolder 구현에서 Windows 탐색기에서 표시되는 창에 대한 제안을 보기에 제공하는 데 사용됩니다.
IExtractIconA 클라이언트가 폴더의 개체 중 하나에 연결된 아이콘을 검색할 수 있도록 하는 메서드를 노출합니다. (ANSI)
IExtractIconW 클라이언트가 폴더의 개체 중 하나에 연결된 아이콘을 검색할 수 있도록 하는 메서드를 노출합니다. (유니코드)
IExtractImage Shell 폴더에서 썸네일 이미지를 요청하는 메서드를 노출합니다.
IExtractImage2 IExtractImage의 기능을 확장합니다.
IFileDialog 일반 파일 대화 상자에서 결과를 초기화, 표시 및 가져오는 메서드를 노출합니다.

IFileDialog2 호출자가 일반 파일 대화 상자에서 검색할 수 있는 제한된 특정 위치의 이름을 지정하고 취소 단추에 레이블로 표시할 대체 텍스트를 지정할 수 있는 메서드를 제공하여 IFileDialog 인터페이스를 확장합니다.
IFileDialogControlEvents 애플리케이션이 공통 파일 대화 상자에 추가한 컨트롤과 관련된 이벤트에 대해 애플리케이션에 알림을 받을 수 있도록 하는 메서드를 노출합니다.
IFileDialogCustomize 애플리케이션이 공통 파일 대화 상자에 컨트롤을 추가할 수 있도록 하는 메서드를 노출합니다.
IFileDialogEvents 공통 파일 대화 상자 내의 이벤트 알림을 허용하는 메서드를 노출합니다.
IFileIsInUse 다른 애플리케이션에서 사용 중인 파일에 대한 정보를 얻거나 닫기 위해 호출할 수 있는 메서드를 노출합니다.
IFileOpenDialog 열려 있는 대화 상자와 관련된 메서드를 추가하여 IFileDialog 인터페이스를 확장합니다.
IFileOperation 진행률 및 오류 대화 상자를 제공하는 메서드뿐만 아니라 셸 항목을 복사, 이동, 이름 바꾸기, 만들기 및 삭제하는 메서드를 노출합니다. 이 인터페이스는 SHFileOperation 함수를 대체합니다.
IFileOperationProgressSink IFileOperation 호출자가 해당 인터페이스를 통해 수행하는 작업의 세부 정보를 모니터링하는 데 사용하는 풍부한 알림 시스템을 제공하는 메서드를 노출합니다.
IFileSaveDialog 파일과 함께 유지할 메타데이터 컬렉션을 지원하는 메서드를 포함하는 저장 대화 상자와 관련된 메서드를 추가하여 IFileDialog 인터페이스를 확장합니다.
IFileSyncMergeHandler . (IFileSyncMergeHandler)
IFileSystemBindData

IShellFolder::P arseDisplayName에 대한 호출을 최적화하기 위한 파일 시스템 정보를 저장하는 메서드를 노출합니다.
IFileSystemBindData2 IShellFolder::P arseDisplayName에 대한 호출을 최적화하기 위한 파일 시스템 정보를 저장하는 IFileSystemBindData를 확장합니다. 이 인터페이스는 기능 집합을 추가하거나 파일 ID 또는 CLSID(접합 클래스 식별자)를 가져옵니다.
IFileViewerA 등록된 파일 뷰어가 파일을 표시하거나 인쇄해야 할 때 알림을 받을 수 있도록 하는 인터페이스를 지정하는 메서드를 노출합니다. (ANSI)
IFileViewerSite 파일 뷰어가 현재 고정된 창에 대한 핸들을 검색하거나 고정된 새 창을 설정할 수 있도록 하는 인터페이스를 지정하는 메서드를 노출합니다.
IFileViewerW 등록된 파일 뷰어가 파일을 표시하거나 인쇄해야 할 때 알림을 받을 수 있도록 하는 인터페이스를 지정하는 메서드를 노출합니다. (유니코드)
IFolderBandPriv IFolderBandPriv는 요구 사항 섹션에 지정된 운영 체제에서 사용할 수 있습니다. 이후 버전에서는 변경되거나 사용할 수 없습니다.
IFolderFilter 서버 애플리케이션에 의해 Shell 폴더의 열거형을 필터링하는 방법을 지정하기 위해 클라이언트에서 노출합니다.
IFolderFilterSite 클라이언트가 셸 폴더 열거형을 필터링하는 방법을 지정할 수 있도록 호스트에서 내보냅니다.
IFolderView 폴더의 표시 옵션에 대한 정보를 검색하고, 해당 폴더에서 지정된 항목을 선택하고, 폴더의 보기 모드를 설정하는 메서드를 노출합니다. (IFolderView)
IFolderView2 폴더의 표시 옵션에 대한 정보를 검색하고, 해당 폴더에서 지정된 항목을 선택하고, 폴더의 보기 모드를 설정하는 메서드를 노출합니다. (IFolderView2)
IFolderViewHost

창에서 IFolderView 개체를 호스트하는 메서드를 노출합니다.
IFolderViewOptions Windows 7 이상 보기와 관련된 폴더 보기 옵션을 제어할 수 있는 메서드를 노출합니다.
IFolderViewSettings 폴더 보기 설정을 가져오는 메서드를 노출합니다.
IFrameworkInputPane 입력 창의 상태 변경 내용 및 위치를 앱에 알릴 수 있는 메서드를 제공합니다.
IFrameworkInputPaneHandler 입력 창(화상 키보드 또는 필기 패널)이 표시되거나 숨겨질 때 앱에 알림을 제공할 수 있습니다. 이렇게 하면 입력 창에 의해 입력 영역(예: 텍스트 상자)이 가려지지 않도록 앱 창이 표시를 조정할 수 있습니다.
IHandlerActivationHost . (IHandlerActivationHost)
IHandlerInfo IHandlerActivationHost 인터페이스의 메서드에 처리기에 대한 정보를 제공하는 메서드를 제공합니다.
IHomeGroup 컴퓨터의 HomeGroup 멤버 자격 상태를 확인하고 공유 마법사를 표시하는 메서드를 노출합니다.
IHWEventHandler 등록된 미디어 형식의 처리를 구현하기 위해 자동 실행에서 호출됩니다.
IHWEventHandler2 IHWEventHandler 인터페이스를 확장하여 디바이스 처리기의 UAC(사용자 계정 컨트롤) 상승 문제를 해결합니다.
IIdentityName 두 항목을 비교하여 동일한지 확인하는 메서드를 노출합니다.
IImageRecompress 이미지를 다시 압축하는 메서드를 노출합니다.
IInitializeCommand

애플리케이션에서 지정한 명령 이름 및 등록된 속성을 사용하여 IExplorerCommandState, IExecuteCommand 또는 IDropTarget을 구현하는 개체를 초기화하는 데 사용되는 단일 메서드를 노출합니다.
IInitializeWithBindCtx 바인딩 컨텍스트를 사용하여 속성 처리기, 썸네일 처리기 또는 미리 보기 처리기와 같은 처리기를 초기화하는 메서드를 노출합니다.
IInitializeWithFile 파일 경로를 사용하여 속성 처리기, 썸네일 처리기 또는 미리 보기 처리기와 같은 처리기를 초기화하는 메서드를 노출합니다.
IInitializeWithItem IShellItem을 사용하여 속성 처리기, 썸네일 처리기 또는 미리 보기 처리기와 같은 처리기를 초기화하는 데 사용되는 메서드를 노출합니다.
IInitializeWithPropertyStore 속성 저장소를 사용하여 속성 처리기, 썸네일 처리기 또는 미리 보기 처리기와 같은 처리기를 초기화하는 메서드를 노출합니다.
IInitializeWithStream 스트림을 사용하여 속성 처리기, 썸네일 처리기 또는 미리 보기 처리기와 같은 처리기를 초기화하는 메서드를 노출합니다.
IInitializeWithWindow 클라이언트가 데스크톱 애플리케이션에서 사용되는 Windows 런타임 개체에 소유자 창을 제공할 수 있는 메서드를 노출합니다.
IInputObject 셸에 포함된 사용자 입력 개체에 대한 UI 활성화 및 프로세스 가속기를 변경하는 메서드를 노출합니다.
IInputObject2 전역 가속기를 처리하여 IInputObject를 확장하는 메서드를 노출합니다.
IInputObjectSite Shell에 포함된 사용자 입력 개체에 대한 포커스 변경 내용을 전달하는 데 사용되는 메서드를 노출합니다.
IInputPanelConfiguration 데스크톱 앱이 Windows 스토어 앱에서 사용되는 포커스 추적 메커니즘을 옵트인하는 기능을 제공합니다.

IInputPanelInvocationConfiguration Windows 스토어 앱에서 자동 호출 동작을 옵트아웃할 수 있습니다.
IInsertItem IInsertItem을 변경하거나 사용할 수 없습니다.
IIOCancelInformation 진행률 대화 상자에서 프로세스 스레드에 취소 창 메시지를 게시하는 메서드를 노출합니다.
IItemNameLimits 유효하고 잘못된 문자 목록 또는 네임스페이스의 이름 최대 길이를 검색합니다. 유효성 검사 구문 분석 및 변환에 이 인터페이스를 사용합니다.
IKnownFolder 애플리케이션이 알려진 폴더의 범주, 형식, GUID, PIDL(항목 식별자 목록) 값, 리디렉션 기능 및 정의에 대한 포인터에 대한 정보를 검색할 수 있도록 하는 메서드를 노출합니다.
IKnownFolderManager 기존 알려진 폴더를 만들거나 열거하거나 관리하는 메서드를 노출합니다.
ILaunchSourceAppUserModelId AppUserModelId를 검색하는 메서드를 제공합니다.
ILaunchSourceViewSizePreference 원본 애플리케이션에 대한 정보를 검색하는 메서드를 제공합니다.
ILaunchTargetMonitor . (ILaunchTargetMonitor)
ILaunchTargetViewSizePreference 새 애플리케이션 창에 대한 기본 보기 크기를 검색하는 방법을 제공합니다.
IMenuBand COM(Component Object Model) 개체가 적절한 메시지를 받고 번역할 수 있도록 하는 메서드를 노출합니다.
IMenuPopup

IMenuPopup을 변경하거나 사용할 수 없습니다.
IModalWindow 모달 창을 나타내는 메서드를 노출합니다.
INamedPropertyBag 개체가 해당 속성을 저장할 수 있는 지정된 속성 모음을 개체에 제공하는 메서드를 노출합니다.
INamedPropertyStore 명명된 속성을 가져와서 설정하는 메서드를 노출합니다.
INameSpaceTreeAccessible 네임스페이스 트리 컨트롤에서 셸 항목에 대한 접근성 작업을 수행하는 메서드를 노출합니다.
INameSpaceTreeControl 셸 항목 트리에서 노드를 보고 조작하는 데 사용되는 메서드를 노출합니다.
INameSpaceTreeControl2 셸 네임스페이스 항목에 사용할 treeview 컨트롤의 표시 스타일을 가져오고 설정하는 메서드를 제공하여 INameSpaceTreeControl 인터페이스를 확장합니다.
INameSpaceTreeControlCustomDraw 사용자가 사용자 지정 네임스페이스 트리 컨트롤 및 해당 항목을 그릴 수 있도록 하는 메서드를 노출합니다.
INameSpaceTreeControlDropHandler 끌어서 놓기에 대한 처리기 메서드를 노출합니다.
INameSpaceTreeControlEvents INameSpaceTreeControl 이벤트를 처리하기 위한 메서드를 노출합니다.
INameSpaceTreeControlFolderCapabilities 폴더의 System.IsPinnedToNameSpaceTree 필터링 지원 상태를 검색하는 단일 메서드를 노출합니다.
INamespaceWalk 지정된 루트 노드에서 네임스페이스를 걷는 메서드를 노출합니다. 연습의 깊이를 지정하고, 이동한 모든 노드의 ID를 포함하는 선택적 배열이 반환됩니다.

INamespaceWalkCB INamespaceWalk와 함께 사용되는 메서드를 노출하는 콜백 인터페이스입니다.
INamespaceWalkCB2 네임스페이스 워크를 완료하는 데 필요한 메서드를 사용하여 INamespaceWalkCB를 확장합니다. 이 메서드는 연습 중에 수집된 데이터를 제거합니다.
INewMenuClient Windows 7 메뉴에서 항목을 조작할 수 있는 메서드를 노출합니다.
INewShortcutHookA 새 인터넷 바로 가기를 만드는 메서드를 노출합니다. (ANSI)
INewShortcutHookW 새 인터넷 바로 가기를 만드는 메서드를 노출합니다. (유니코드)
INewWindowManager 팝업 창을 제어할 수 있도록 다른 창에서 시작된 창을 표시할지 차단할지 여부를 결정하는 메서드를 노출합니다.
INotifyReplica 개체의 작성자에게 개체가 후속 조정의 대상이 될 수 있음을 알리는 수단을 제공하는 메서드를 노출합니다. 서류 가방 조정자는 이 인터페이스를 구현하는 역할을 담당합니다.
IObjectArray 클라이언트가 IUnknown을 지원하는 개체 컬렉션의 항목에 액세스할 수 있도록 하는 메서드를 노출합니다.
IObjectCollection 클라이언트가 컬렉션에서 IUnknown을 지원하는 개체를 추가하고 제거할 수 있는 메서드를 제공하여 IObjectArray 인터페이스를 확장합니다.
IObjectProvider 다른 개체의 GUID로 명명된 개체를 검색하는 메서드를 노출합니다. QueryService와 달리 이 인터페이스는 해당 기능을 다른 개체에 위임하지 않습니다.
IObjectWithAppUserModelID 사용자 지정 IAssocHandler 개체의 구현자가 AppUserModelID(명시적 애플리케이션 사용자 모델 ID)에 대한 액세스를 제공할 수 있도록 하는 메서드를 노출합니다.

IObjectWithBackReferences 개체가 보유한 뒤로 참조와 상호 작용하기 위한 메서드를 제공합니다.
IObjectWithCancelEvent 지원되지 않습니다. 호출된 개체가 작업의 취소를 나타내는 신호를 받는 이벤트를 호출자에게 제공합니다.
IObjectWithFolderEnumMode 구문 분석된 항목의 열거 모드를 가져와서 설정하는 메서드를 노출합니다.
IObjectWithProgID 개체와 연결된 ProgID에 대한 액세스를 제공하는 메서드를 노출합니다.
IObjectWithPropertyKey 속성 키를 가져오고 설정하기 위한 메서드를 노출합니다.
IObjectWithSelection 셸 항목 배열이 나타내는 선택한 항목을 가져오거나 설정하는 메서드를 노출합니다.
IObjMgr 클라이언트가 서버 개체에서 관리하는 개체 컬렉션에서 개체를 추가하거나 제거할 수 있도록 하는 메서드를 노출합니다.
IOpenControlPanel 제어판의 뷰 상태, 개별 제어판 항목의 경로를 검색하고 제어판 자체 또는 개별 제어판 항목을 여는 메서드를 노출합니다.
IOpenSearchSource 사용자 지정 클라이언트 쪽 OpenSearch 데이터 원본에서 검색 결과를 가져오는 메서드를 노출합니다.
IOperationsProgressDialog 진행률 대화 상자를 가져와서 설정하고 쿼리하는 메서드를 노출합니다.
IPackageDebugSettings 디버거 개발자가 일시 중단 또는 재개와 같은 Windows 스토어 앱의 수명 주기를 제어할 수 있습니다.
IPackageExecutionStateChangeNotification Windows 스토어 앱 디버깅 중에 패키지 상태 변경 알림을 받을 수 있습니다.

IParentAndItem 부모 및 부모의 자식 ID를 가져오고 설정하는 메서드를 노출합니다. IParentAndItem은 일반적으로 IShellItems에서 구현되지만 IShellItem과는 관련이 없습니다.
IParseAndCreateItem IParseAndCreateItem 인터페이스
IPersistFolder Shell 폴더 개체를 초기화하는 메서드를 노출합니다.
IPersistFolder2 Shell 폴더 개체에서 정보를 가져오는 메서드를 노출합니다.
IPersistFolder3 폴더 개체가 폴더 바로 가기의 기본이 아닌 처리를 구현할 수 있도록 하여 IPersistFolder 및 IPersistFolder2 인터페이스를 확장합니다.
IPersistIDList 항목 식별자 목록을 유지하는 데 사용되는 메서드를 노출합니다.
IPersistSerializedPropStorage 나중에 사용할 수 있도록 직렬화된 속성 스토리지 데이터를 보관하고 지속형 데이터를 새 속성 저장소 인스턴스로 복원하는 메서드를 노출합니다. (IPersistSerializedPropStorage)
IPersistSerializedPropStorage2 나중에 사용할 수 있도록 직렬화된 속성 스토리지 데이터를 보관하고 지속형 데이터를 새 속성 저장소 인스턴스로 복원하는 메서드를 노출합니다. (IPersistSerializedPropStorage2)
IPreviewHandler 다양한 미리 보기 표시를 위한 메서드를 노출합니다.
IPreviewHandlerFrame 미리 보기 처리기가 호스트에 바로 가기 키를 전달할 수 있도록 합니다. 이 인터페이스는 바로 가기 키 목록을 검색하고 호스트가 바로 가기 키를 처리하도록 지시합니다.
IPreviewHandlerVisuals 미리 보기 처리기에 색 및 글꼴 정보를 적용하는 메서드를 노출합니다.

IPreviewItem 미리 보기 창에 표시될 항목을 식별합니다.
IPreviousVersionsInfo Windows Server 2003에서 제공하는 섀도 복사본 기술로 되돌리기 위해 저장된 이전 버전의 서버 파일 또는 폴더를 확인하는 메서드를 노출합니다.
IProfferService 개체가 동일한 호스트의 다른 개체에 서비스를 제공하는 일반적인 메커니즘을 노출합니다.
IProgressDialog 애플리케이션에서 진행률 대화 상자를 표시하는 옵션을 제공하는 메서드를 노출합니다.
IPublishedApp 제어판에서 프로그램 추가/제거에 애플리케이션을 나타내는 메서드를 노출합니다.
IPublishedApp2 추가 설치 방법을 제공하여 IPublishedApp 인터페이스를 확장합니다.
IPublishingWizard 온라인 인쇄 마법사, 웹 게시 마법사 및 네트워크 위치 추가 마법사를 사용하기 위한 메서드를 노출합니다. Windows Vista에서 IPublishingWizard는 더 이상 웹 게시 마법사 또는 온라인 인쇄 마법사를 지원하지 않습니다.
IQueryAssociations 파일 형식 또는 프로토콜을 정의하고 애플리케이션과 연결하여 레지스트리에 저장된 정보를 검색하는 프로세스를 간소화하는 메서드를 노출합니다.
IQueryCancelAutoPlay 자동 실행 또는 자동 실행을 프로그래밍 방식으로 재정의하는 메서드를 노출합니다. 이렇게 하면 미디어를 삽입할 때 시작되는 콘텐츠의 위치와 형식을 사용자 지정할 수 있습니다.
IQueryCodePage ANSI 코드 페이지의 숫자 값(코드 페이지 식별자)을 가져오고 설정합니다.
IQueryContinue 작업을 계속할 수 있는 권한을 클라이언트에 쿼리하는 개체에 대한 간단한 표준 메커니즘을 제공하는 메서드를 노출합니다.

IQueryContinueWithStatus 이러한 시도를 계속해야 하는지 확인하기 위해 네트워크에 연결을 시도하는 동안 자격 증명 공급자가 QueryContinue를 호출하는 표준 메커니즘을 제공하는 메서드를 노출합니다.
IQueryInfo 셸이 IShellFolder 구현에 상주하는 항목에 대한 플래그 및 정보 팁 정보를 검색하는 데 사용하는 메서드를 노출합니다. 정보 팁은 일반적으로 도구 설명 컨트롤 내에 표시됩니다.
IRegTreeItem Tree-View 컨트롤 창 스타일 플래그가 설정된 트리 뷰 컨트롤의 항목 상태를 검색하고 설정하는 메서드를 노출합니다.
IRelatedItem 특정 관계를 사용하여 관련 항목을 파생시키는 메서드를 노출합니다.
IRemoteComputer 원격 개체에서 호출될 때 네임스페이스 확장을 열거하거나 초기화하는 메서드를 노출합니다. 예를 들어 이 인터페이스는 원격 프린터 가상 폴더를 초기화하는 데 사용됩니다.
IResolveShellLink 애플리케이션에서 셸 폴더 개체가 해당 항목 중 하나에 대한 링크를 확인하도록 요청할 수 있는 메서드를 노출합니다.
IResultsFolder 데이터 개체의 항목을 보유하는 메서드를 노출합니다.
IRunnableTask 백그라운드 스레드에서 작업을 수행할 수 있도록 개체에 의해 노출될 수 있는 자유 스레드 인터페이스입니다.
ISearchBoxInfo 호출자가 검색 상자에 입력한 정보를 검색할 수 있도록 하는 메서드를 노출합니다.
ISearchContext 사용자 지정 정보를 검색 후크에 채널하는 메서드를 노출합니다.
ISearchFolderItemFactory 검색 폴더를 만들고 수정하는 메서드를 노출합니다.

ISharedBitmap 비트맵에 액세스하기 위한 메모리 효율적인 메서드를 노출합니다. 이 인터페이스는 HBITMAP 개체에 대한 씬 래퍼로 사용되므로 해당 개체를 참조 개수로 계산하고 기본 데이터가 변경되지 않도록 보호할 수 있습니다.
ISharingConfigurationManager 사용자(C:\Users) 또는 공용(C:\Users\Public) 폴더에 대한 컴퓨터의 기본 공유 설정에 대한 정보를 설정하고 검색하는 메서드를 노출합니다. 프린터 공유를 제어할 수 있는 메서드 집합도 노출합니다.
IShellApp 애플리케이션에 대한 일반적인 정보를 제공하는 메서드를 프로그램 추가/제거 애플리케이션에 노출합니다.
IShellBrowser 셸 뷰의 호스트(IShellView를 구현하는 개체)에 의해 구현됩니다. 호스팅하는 보기 및 탐색기 창의 컨텍스트에서 실행되는 다른 개체에 대한 서비스를 제공하는 메서드를 노출합니다.
IShellChangeNotify 항목의 ID가 변경될 때 셸 네임스페이스 확장에 알립니다.
IShellDetails 폴더의 항목에 대한 자세한 정보를 제공하기 위해 Shell 폴더에 의해 노출됩니다.
IShellExtInit 속성 시트, 바로 가기 메뉴 및 끌어서 놓기 처리기(기본값이 아닌 끌어서 놓기 작업 중 바로 가기 메뉴에 항목을 추가하는 확장)에 대한 셸 확장을 초기화하는 메서드를 노출합니다.
IShellFolder 모든 Shell 네임스페이스 폴더 개체에 의해 노출되는 해당 메서드는 폴더를 관리하는 데 사용됩니다.
IShellFolder2 IShellFolder의 기능을 확장합니다. 해당 메서드는 Shell 폴더의 내용에 대한 다양한 정보를 제공합니다.
IShellFolderBand IShellFolderBand를 변경하거나 사용할 수 없습니다.
IShellFolderView 셸 폴더 뷰를 조작하는 메서드를 노출합니다.

IShellFolderViewCB 시스템 폴더 뷰 개체(SHCreateShellFolderView를 통해 반환된 IShellView 개체)를 사용하여 구현된 폴더 보기와 Windows 탐색기 간의 통신을 허용하는 메서드를 노출하여 폴더 뷰에 이벤트를 알리고 그에 따라 보기를 수정할 수 있도록 합니다.
IShellFolderViewDual 보기를 수정하고 현재 폴더의 항목을 선택하는 메서드를 노출합니다. (IShellFolderViewDual)
IShellFolderViewDual2 보기를 수정하고 현재 폴더의 항목을 선택하는 메서드를 노출합니다. (IShellFolderViewDual2)
IShellFolderViewDual3 현재 폴더 뷰를 수정하는 메서드를 노출합니다.
IShellIcon IShellFolder 개체에 대한 아이콘 인덱스를 가져오는 메서드를 노출합니다.
IShellIconOverlay 네임스페이스 확장에서 포함하는 개체에 대한 아이콘 오버레이를 지정하는 데 사용되는 메서드를 노출합니다.
IShellIconOverlayIdentifier 아이콘 오버레이 처리기와 셸 간의 모든 통신을 처리하는 메서드를 노출합니다.
IShellIconOverlayManager IShellIconOverlayManager를 변경하거나 사용할 수 없습니다.
IShellImageData 이미지 데이터를 표시, 조작 및 설명하는 메서드 및 속성을 노출합니다.
IShellImageDataAbort IShellImageData 프로세스를 중단하는 데 사용되는 단일 메서드를 노출합니다.
IShellImageDataFactory 다양한 이미지 원본을 기반으로 IShellImageData 인스턴스를 만드는 메서드를 노출합니다.
IShellImageStore

Deprecated. 이미지 캐시를 조작하는 메서드를 노출합니다.
IShellItem 셸 항목에 대한 정보를 검색하는 메서드를 노출합니다. IShellItem 및 IShellItem2는 새 코드에서 항목의 기본 표현입니다.
IShellItem2 항목의 다양한 속성 값을 검색하는 메서드를 사용하여 IShellItem을 확장합니다. IShellItem 및 IShellItem2는 새 코드에서 항목의 기본 표현입니다.
IShellItemArray 셸 항목 배열을 만들고 조작하는 메서드를 노출합니다.
IShellItemFilter 서버 애플리케이션에 의해 셸 항목의 열거형을 필터링하는 방법을 지정하기 위해 클라이언트에 의해 노출됩니다.
IShellItemImageFactory 셸 항목에 대한 아이콘 또는 썸네일을 반환하는 메서드를 노출합니다. 요청된 항목에 대해 미리 보기 또는 아이콘을 사용할 수 없는 경우 셸에서 클래스별 아이콘을 제공할 수 있습니다.
IShellItemResources 셸 항목 리소스를 조작하고 쿼리하는 메서드를 노출합니다.
IShellLibrary 라이브러리를 만들고 관리하기 위한 메서드를 노출합니다.
IShellLinkA 셸 링크를 만들고, 수정하고, 확인하는 메서드를 노출합니다. (ANSI)
IShellLinkDataList 애플리케이션이 셸 링크에 추가 데이터 블록을 연결할 수 있도록 하는 메서드를 노출합니다. 이러한 메서드는 데이터 블록을 추가, 복사 또는 제거합니다.
IShellLinkW 셸 링크를 만들고, 수정하고, 확인하는 메서드를 노출합니다. (유니코드)
IShellMenu

시작 메뉴 및 즐겨찾기 메뉴와 같은 셸 메뉴와 상호 작용하는 메서드를 노출합니다.
IShellMenuCallback 메뉴 밴드에서 메시지를 받는 메서드를 노출하는 콜백 인터페이스입니다.
IShellPropSheetExt 속성 시트 처리기가 파일 개체에 대해 표시된 속성 시트의 페이지를 추가하거나 바꿀 수 있도록 하는 메서드를 노출합니다.
IShellRunDll 지원되지 않습니다. (IShellRunDll)
IShellService Deprecated. IShellService는 특정 인터페이스를 구현하는 서비스 구성 요소가 Windows Internet Explorer 및 Windows Explorer와 같은 여러 클라이언트 간에 공유되는 경우 소유권을 선언하는 한 가지 메서드를 노출합니다.
IShellTaskScheduler IShellTaskScheduler를 변경하거나 사용할 수 없습니다.
IShellView Windows 탐색기 또는 폴더 창에 보기를 표시하는 메서드를 노출합니다.
IShellView2 IShellView의 기능을 확장합니다.
IShellView3 IShellView2::CreateViewWindow2를 대체하는 메서드를 제공하여 IShellView2의 기능을 확장합니다.
IShellWindows 열린 Shell 창의 컬렉션에 대한 액세스를 제공합니다.
IStartMenuPinnedList 시작 메뉴 또는 작업 표시줄에서 애플리케이션 바로 가기를 고정 해제하는 메서드를 노출합니다.
IStorageProviderHandler 특정 파일 또는 폴더와 연결된 IStorageProviderPropertyHandler를 검색합니다.

IStorageProviderPropertyHandler 파일 또는 폴더와 연결된 속성의 컬렉션을 제공합니다.
IStreamAsync 비동기 스트림에 입력/출력(I/O)을 관리하는 메서드를 노출합니다.
IStreamUnbufferedInfo 바이트 맞춤을 지원하기 위해 섹터 크기를 결정하는 메서드를 노출합니다.
ISuspensionDependencyManager . (ISuspensionDependencyManager)
ISyncMgrConflict 충돌 저장소에서 검색된 충돌에 대한 정보를 제공하고 충돌을 해결할 수 있도록 하는 메서드를 노출합니다.
ISyncMgrConflictFolder 충돌 개체의 충돌 ID 목록을 가져오는 메서드를 노출합니다.
ISyncMgrConflictItems 충돌 항목 데이터 및 항목 수를 가져오는 메서드를 노출합니다.
ISyncMgrConflictPresenter 사용자에게 충돌을 표시하는 메서드를 노출합니다.
ISyncMgrConflictResolutionItems 항목 정보 및 항목 수를 가져오는 메서드를 노출합니다.
ISyncMgrConflictResolveInfo 동기화 관리자 충돌 해결에 대한 정보를 가져와서 설정하는 메서드를 노출합니다.
ISyncMgrConflictStore 처리기가 Conflicts 폴더에 표시되는 충돌을 제공할 수 있도록 하는 메서드를 노출합니다.
ISyncMgrControl 애플리케이션 또는 처리기가 동기화를 시작하거나 중지하거나, 동기화 센터에 처리기 또는 항목 집합의 변경 내용을 알리거나, 속성 값의 변경 내용을 알릴 수 있는 메서드를 노출합니다.

ISyncMgrEnumItems SYNCMGRITEM 구조의 배열을 통해 열거하는 메서드를 노출합니다.
ISyncMgrEvent 이벤트 저장소에서 데이터를 검색하는 메서드를 노출합니다. 이벤트 저장소를 사용하면 동기화 센터에서 저장소에 있는 모든 이벤트의 열거자를 가져올 뿐만 아니라 개별 이벤트를 검색할 수 있습니다.
ISyncMgrEventLinkUIOperation 동기화 결과 폴더에서 이벤트 링크를 클릭할 때 호출되는 메서드를 제공합니다.
ISyncMgrEventStore 처리기가 기본 동기화 센터 이벤트 저장소를 사용하는 대신 자체 이벤트 저장소를 제공하고 자체 동기화 이벤트를 관리할 수 있도록 하는 메서드를 노출합니다. 이러한 이벤트는 동기화 결과 폴더에 표시됩니다.
ISyncMgrHandler 동기화 처리기에서 구현된 기본 인터페이스를 구성하는 메서드를 노출합니다.
ISyncMgrHandlerCollection 동기화 처리기 ID의 열거자를 제공하고 해당 동기화 처리기를 인스턴스화하는 메서드를 노출합니다.
ISyncMgrHandlerInfo 처리기가 동기화 센터에 속성 및 상태 정보를 제공할 수 있도록 하는 메서드를 노출합니다.
ISyncMgrRegister 애플리케이션이 동기화 관리자에 등록할 수 있도록 메서드를 노출합니다. ISyncMgrRegister 인터페이스를 통해 또는 레지스트리에 직접 등록하여 이 작업을 수행할 수 있습니다.
ISyncMgrResolutionHandler 동기화 충돌을 관리하는 메서드를 노출합니다. 이 인터페이스를 구현하여 동기화 충돌 처리기를 생성합니다. 충돌 해결 UI(사용자 인터페이스)는 이 인터페이스를 호출하여 사용자에게 표시되는 충돌을 해결합니다.
ISyncMgrScheduleWizardUIOperation 처리기가 처리기의 동기화 일정 마법사를 표시할 수 있도록 하는 메서드를 노출합니다.
ISyncMgrSessionCreator 처리기 또는 외부 애플리케이션이 동기화가 시작되었음을 동기화 센터에 알리고 진행률 및 이벤트를 보고할 수 있는 단일 메서드를 노출합니다.

ISyncMgrSyncCallback 동기화 프로세스에서 진행률 및 이벤트를 동기화 센터에 보고하거나 프로세스가 취소되었는지 여부를 쿼리할 수 있는 메서드를 노출합니다.
ISyncMgrSynchronize 등록된 애플리케이션 또는 서비스가 동기화 관리자로부터 알림을 받을 수 있도록 하는 메서드를 노출합니다.
ISyncMgrSynchronizeCallback 동기화 프로세스를 관리하는 메서드를 노출합니다.
ISyncMgrSynchronizeInvoke 등록된 애플리케이션이 동기화 관리자를 호출하여 항목을 업데이트할 수 있도록 하는 메서드를 노출합니다.
ISyncMgrSyncItem 처리기가 동기화 항목을 독립적인 개체로 관리할 수 있도록 단일 동기화 항목에서 동작하고 정보를 검색하는 메서드를 노출합니다.
ISyncMgrSyncItemContainer 포함된 항목에 대한 정보를 처리기에 제공하는 메서드를 노출합니다.
ISyncMgrSyncItemInfo 단일 동기화 항목에 대한 속성 및 상태 정보를 제공하는 메서드를 노출합니다.
ISyncMgrSyncResult ISyncMgrControl을 호출하는 애플리케이션이 ISyncMgrControl::StartHandlerSync 또는 ISyncMgrControl::StartItemSync 호출의 결과를 가져오는 데 사용할 수 있는 메서드를 노출합니다.
ISyncMgrUIOperation 동기화 센터에서 요청할 때 동기화 처리기 또는 동기화 항목이 UI 개체를 표시할 수 있는 메서드를 노출합니다.
ITaskbarList 작업 표시줄을 제어하는 메서드를 노출합니다. 작업 표시줄에서 항목을 동적으로 추가, 제거 및 활성화할 수 있습니다.
ITaskbarList2

창을 전체 화면 표시로 표시하는 메서드를 노출하여 ITaskbarList 인터페이스를 확장합니다.
ITaskbarList3 Windows 7에 추가된 통합 시작 및 전환 작업 표시줄 단추 기능을 지원하는 메서드를 노출하여 ITaskbarList2를 확장합니다.
ITaskbarList4 호출자가 탭 미리 보기 및 피킹 기능에 대한 두 개의 속성 값을 제어할 수 있는 메서드를 제공하여 ITaskbarList3을 확장합니다.
IThumbnailCache 애플리케이션 간에 공유되는 시스템 미리 보기 캐시에 대한 메서드를 노출합니다.
IThumbnailCachePrimer . (IThumbnailCachePrimer)
IThumbnailCapture HTML 배경 화면의 썸네일 표현을 가져오는 메서드를 노출합니다.
IThumbnailHandlerFactory 항목의 썸네일 처리기를 검색하는 메서드를 노출합니다. 자식 IDList에 사용되는 추출기를 지정하려면 이 인터페이스를 구현합니다.
IThumbnailProvider 썸네일 이미지를 가져오기 위한 메서드를 노출하며 썸네일 처리기를 위해 구현됩니다. 이 인터페이스를 구현하는 개체도 IInitializeWithStream을 구현해야 합니다.
IThumbnailSettings 썸네일 공급자가 썸네일 요청의 사용자 컨텍스트를 확인할 수 있도록 하는 메서드를 제공합니다.
IThumbnailStreamCache 썸네일 스트림을 가져오거나 설정합니다. 이 인터페이스는 내부 전용이며 사진 애플리케이션에서만 호출할 수 있습니다.
ITrackShellMenu 메뉴와 함께 도구 모음 단추를 조정하고 팝업 메뉴를 표시하는 기능을 제공하여 IShellMenu 인터페이스를 확장하는 메서드를 노출합니다.
ITranscodeImage

Windows에서 지원하는 모든 이미지 형식에서 JPEG 또는 BMP(비트맵) 이미지 형식으로 변환할 수 있는 메서드를 노출합니다.
ITransferAdviseSink 상태 수집 및 실패 정보를 지원하는 메서드를 노출합니다.
ITransferDestination 복사 또는 이동 작업에 대한 대상 셸 항목을 만드는 메서드를 노출합니다. 이 인터페이스는 ITransferDestination::Advise 메서드를 제공하여 파일 작업을 보다 자세히 제어할 수 있도록 제공됩니다.
ITransferMediumItem 복사 엔진에서 QueryInterface를 호출하여 인터페이스 ITransferDestination 또는 인터페이스 ITransferSource에 대한 포인터를 반환할 항목을 가져오는 데 사용됩니다. 복사, 이동 또는 삭제 작업을 위해 이러한 인터페이스를 쿼리하고 열거할 수 있습니다.
ITransferSource 복사, 이동, 재활용 등을 포함하여 IShellItem을 조작하는 메서드를 노출합니다. 이 인터페이스는 ITransferSource::Advise 메서드를 제공하여 파일 작업에 대한 더 많은 제어를 제공하도록 제공됩니다.
ITravelEntry Deprecated. 브라우저의 여행 기록에서 개별 항목을 식별, 호출 및 업데이트하는 메서드를 노출합니다.
ITravelLog Deprecated. 브라우저에서 이동 레코드를 유지 관리하고 조작하는 메서드를 노출합니다.
ITrayDeskBand 데스크밴드를 표시, 숨기기 및 쿼리하는 메서드를 노출합니다.
IUpdateIDList 폴더 개체의 자식에 대한 ITEMIDLIST를 업데이트하는 메서드를 제공합니다.
IURLSearchHook 브라우저에서 알 수 없는 URL 프로토콜의 주소를 변환하는 데 사용하는 메서드를 노출합니다.
IURLSearchHook2 검색 컨텍스트 개체를 사용하여 알 수 없는 URL 프로토콜의 주소를 변환하기 위해 브라우저에서 사용하는 메서드를 노출합니다.
IUserAccountChangeCallback

사용자 계정을 나타내는 그림이 변경될 때 호출되는 메서드를 노출합니다.
IUserNotification 알림 정보를 설정한 다음 작업 표시줄의 알림 영역과 함께 표시되는 풍선에서 사용자에게 해당 알림을 표시하는 메서드를 노출합니다. (IUserNotification)
IUserNotification2 알림 정보를 설정한 다음 작업 표시줄의 알림 영역과 함께 표시되는 풍선에서 사용자에게 해당 알림을 표시하는 메서드를 노출합니다. (IUserNotification2)
IUserNotificationCallback 알림 풍선에서 마우스 클릭 또는 바로 가기 메뉴 액세스를 처리하기 위한 메서드를 노출합니다. IUserNotification2::Show와 함께 사용됩니다.
IUseToBrowseItem 이 인터페이스는 지원되지 않습니다. 사용하지 마세요.
IViewStateIdentityItem 뷰 사용자 지정이 기억될 항목인 정식 지속성 항목을 제공합니다.
IVirtualDesktopManager 애플리케이션이 가상 작업 영역을 형성하는 창 그룹과 상호 작용할 수 있도록 하는 메서드를 노출합니다.
IVisualProperties 시각적 속성을 설정하고 가져오는 메서드를 노출합니다.
IWebWizardExtension 마법사 확장의 초기 URL 및 오류 발생 시 특정 URL을 설정하는 메서드를 노출하여 IWizardExtension 인터페이스를 확장합니다.
IWizardExtension 서버 쪽 콘텐츠 페이지를 호스트하는 웹 게시 마법사 및 온라인 인쇄 순서 마법사와 같은 마법사에서 사용됩니다. 이 인터페이스는 지원되는 확장 페이지를 지정하고 해당 페이지로 들어오고 나가는 메서드를 노출합니다.
IWizardSite 마법사 확장에서 자신과 마법사의 나머지 부분 사이의 테두리를 탐색하는 데 사용하는 메서드를 노출합니다.

구조

테이블 확장

AASHELLMENUFILENAME 메뉴 파일 이름에 대한 정보를 포함하는 가변 크기 구조체입니다.
AASHELLMENUITEM 메뉴 항목에 대한 정보를 포함합니다. (AASHELLMENUITEM)
APPBARDATA 시스템 앱바 메시지에 대한 정보를 포함합니다.
APPCATEGORYINFO 제어판에서 프로그램 추가/제거에 대한 애플리케이션 범주 정보를 제공합니다. APPCATEGORYINFOLIST 구조는 애플리케이션 게시자에 대한 범주의 전체 목록을 만드는 데 사용됩니다.
APPCATEGORYINFOLIST 애플리케이션 게시자에서 제어판의 프로그램 추가/제거에 지원되는 애플리케이션 범주 목록을 제공합니다.
APPINFODATA 프로그램 제어판 추가/제거 유틸리티에 게시된 애플리케이션에 대한 정보를 제공합니다.
ASSOCIATIONELEMENT AssocCreateForClasses에서 지정된 파일 연결에 대한 IQueryAssociations 인터페이스를 검색하는 데 사용하는 정보를 정의합니다.
AUTO_SCROLL_DATA AUTO_SCROLL_DATA 변경되거나 사용할 수 없습니다.
BANDINFOSFB 폴더 밴드에 대한 정보를 포함합니다. 이 구조체는 IShellFolderBand::GetBandInfoSFB 및 IShellFolderBand::SetBandInfoSFB 메서드와 함께 사용됩니다.
BANDSITEINFO 밴드 사이트에 대한 정보를 포함합니다. 이 구조체는 IBandSite::GetBandSiteInfo 및 IBandSite::SetBandSiteInfo 메서드와 함께 사용됩니다.

BASEBROWSERDATALH BASEBROWSERDATALH 구조체에는 기본 클래스의 보호된 멤버가 포함됩니다. (BASEBROWSERDATALH 구조체)
BASEBROWSERDATAXP BASEBROWSERDATAXP 구조체에는 기본 클래스의 보호된 멤버가 포함됩니다. (BASEBROWSERDATAXP 구조체)
BROWSEINFOA SHBrowseForFolder 함수에 대한 매개 변수를 포함하고 사용자가 선택한 폴더에 대한 정보를 받습니다. (ANSI)
BROWSEINFOW SHBrowseForFolder 함수에 대한 매개 변수를 포함하고 사용자가 선택한 폴더에 대한 정보를 받습니다. (유니코드)
캐비닛스테이트 CABINETSTATE는 변경되거나 사용할 수 없습니다.
CATEGORY_INFO 범주 정보를 포함합니다. 구성 요소 범주는 CATID(공통 범주 식별자)를 공유하는 COM(구성 요소 개체 모델) 클래스의 그룹입니다.
CIDA 하나 이상의 셸 네임스페이스 개체의 PIDL(항목 식별자 목록)에 포인터를 전송하기 위해 CFSTR_SHELLIDLIST 클립보드 형식과 함께 사용됩니다.
CM_COLUMNINFO 열 정보를 정의합니다. IColumnManager 인터페이스의 멤버에서 사용됩니다.
CMINVOKECOMMANDINFO IContextMenu::InvokeCommand에서 바로 가기 메뉴 명령을 호출하는 데 필요한 정보를 포함합니다.
CMINVOKECOMMANDINFOEX 바로 가기 메뉴 명령에 대한 확장 정보를 포함합니다. 이 구조는 유니코드 값을 사용할 수 있는 CMINVOKECOMMANDINFO의 확장 버전입니다.
COMDLG_FILTERSPEC

일반적으로 요소를 필터링하는 데 사용됩니다.
구성 요소 구성 요소에 대한 정보를 보관하기 위해 Windows 2000에서 사용됩니다. 이 구조체는 IE4COMPONENT 구조를 대체합니다.
COMPONENTSOPT 데스크톱 항목 옵션을 포함합니다.
COMPPOS 구성 요소의 위치 및 크기에 대한 정보를 보유합니다.
COMPSTATEINFO Windows 2000에서 구성 요소의 상태에 대한 정보를 보관하는 데 사용됩니다.
CONFIRM_CONFLICT_ITEM 충돌 항목 구조를 정의합니다.
CONFIRM_CONFLICT_RESULT_INFO 충돌 결과 정보 구조를 정의합니다.
CPLINFO 제어판 애플리케이션에서 지원하는 대화 상자에 대한 리소스 정보 및 애플리케이션 정의 값을 포함합니다. (CPLINFO)
CREDENTIAL_PROVIDER_CREDENTIAL_SERIALIZATION 자격 증명에 대한 세부 정보를 포함합니다.
CREDENTIAL_PROVIDER_FIELD_DESCRIPTOR 자격 증명의 단일 필드를 설명합니다. 예를 들어 문자열 또는 사용자 이미지입니다.
CSFV SHCreateShellFolderViewEx 함수와 함께 사용됩니다.
DATABLOCK_HEADER IShellLinkDataList에서 사용하는 일부 추가 데이터 구조의 헤더 역할을 합니다.
DEFCONTEXTMENU

SHCreateDefaultContextMenu에서 사용하는 상황에 맞는 메뉴 정보를 포함합니다.
DELEGATEITEMID 표준 ITEMIDLIST 구조 대신 대리자 폴더에서 사용됩니다.
DESKBANDINFO 밴드 개체에 대한 정보를 받습니다. 이 구조체는 사용되지 않는 IDeskBand::GetBandInfo 메서드와 함께 사용됩니다.
DETAILSINFO 셸 폴더 항목에 대한 세부 정보를 포함합니다. SFVM_GETDETAILSOF 알림과 함께 사용됩니다.
DFMICS DFM_INVOKECOMMANDEX 사용되는 추가 인수를 포함합니다.
DLLVERSIONINFO DLL 관련 버전 정보를 받습니다.
DLLVERSIONINFO2 DLL 관련 버전 정보를 받습니다. DllGetVersion 함수와 함께 사용됩니다.
DROPDESCRIPTION drop 개체에 대한 이미지 및 함께 제공되는 텍스트를 설명합니다.
DROPFILES CF_HDROP 클립보드 형식을 정의합니다. 다음 데이터는 파일 이름의 이중 null로 끝나는 목록입니다.
EXP_DARWIN_LINK IShellLinkDataList에서 사용하는 추가 데이터 블록을 보유합니다. 링크의 Windows Installer ID를 보유합니다.
EXP_PROPERTYSTORAGE 셸 링크 상태에 대한 정보를 저장합니다. 이 구조는 EXP_PROPERTYSTORAGE_SIG 태그가 지정된 추가 데이터 섹션에 사용됩니다.
EXP_SPECIAL_FOLDER IShellLinkDataList에서 사용하는 추가 데이터 블록을 보유합니다. 특수 폴더 정보를 보유합니다.

EXP_SZ_LINK IShellLinkDataList에서 사용하는 추가 데이터 블록을 보유합니다. 아이콘 또는 대상에 대해 확장 가능한 환경 문자열을 보유합니다.
EXTRASEARCH IEnumExtraSearch 열거자 개체에서 셸 폴더 개체에서 지원하는 검색 개체에 대한 정보를 반환하는 데 사용됩니다.
FILE_ATTRIBUTES_ARRAY CFSTR_FILE_ATTRIBUTES_ARRAY 대한 클립보드 형식 정의를 포함합니다.
FILEDESCRIPTORA Microsoft ActiveX 끌어서 놓기 작업 중에 클립보드를 통해 복사되는 파일의 속성을 설명합니다. (ANSI)
FILEDESCRIPTORW Microsoft ActiveX 끌어서 놓기 작업 중에 클립보드를 통해 복사되는 파일의 속성을 설명합니다. (유니코드)
FILEGROUPDESCRIPTORA CF_FILEGROUPDESCRIPTOR 클립보드 형식을 정의합니다. (ANSI)
FILEGROUPDESCRIPTORW CF_FILEGROUPDESCRIPTOR 클립보드 형식을 정의합니다. (유니코드)
FOLDERSETDATA Deprecated. IBrowserService2::GetFolderSetData에 사용되는 데이터입니다.
FOLDERSETTINGS 폴더 보기 정보를 포함합니다.
FVSHOWINFO 파일 뷰어가 파일을 표시하는 데 사용하는 정보를 포함합니다.
HELPINFO 상황에 맞는 도움말이 요청된 항목에 대한 정보를 포함합니다.
HELPWININFOA

기본 또는 보조 도움말 창의 크기와 위치를 포함합니다. 애플리케이션은 HELP_SETWINPOS 값으로 WinHelp 함수를 호출하여 이 정보를 설정할 수 있습니다. (ANSI)
HELPWININFOW 기본 또는 보조 도움말 창의 크기와 위치를 포함합니다. 애플리케이션은 HELP_SETWINPOS 값으로 WinHelp 함수를 호출하여 이 정보를 설정할 수 있습니다. (유니코드)
IE4COMPONENT Microsoft Internet Explorer 4.0 및 Microsoft Internet Explorer 4.01에서 구성 요소에 대한 정보를 보관하는 데 사용됩니다. Windows 2000에서는 COMPONENT 구조체로 대체됩니다.
ITEMIDLIST 항목 식별자 목록을 포함합니다.
ITEMSPACING 디스플레이:_small 및 대형에 사용할 수 있는 아이콘 간격의 가능한 두 가지 크기의 크기를 저장합니다. IShellFolderView::GetItemSpacing에서 사용됩니다.
KNOWNFOLDER_DEFINITION 알려진 폴더의 세부 정보를 정의합니다.
LOGFONTA ANSI(LOGFONTA) 구조체는 글꼴의 특성을 정의합니다. (LOGFONTA 구조체)
LOGFONTA ANSI(LOGFONTA) 구조체는 글꼴의 문자 셀 또는 문자 특성을 정의합니다. (LOGFONTA)
LOGFONTW LOGFONTW(유니코드) 구조체는 글꼴의 특성을 정의합니다. (LOGFONTW 구조체)
LOGFONTW LOGFONTW(유니코드) 구조체는 글꼴의 문자 셀 또는 문자 특성을 정의합니다. (LOGFONTW)
MULTIKEYHELPA 검색할 키워드와 Windows 도움말에서 검색할 키워드 테이블을 지정합니다. (ANSI)
MULTIKEYHELPW 검색할 키워드와 Windows 도움말에서 검색할 키워드 테이블을 지정합니다. (유니코드)

NC_ADDRESS 네트워크 주소를 설명하는 정보를 포함합니다.
NEWCPLINFOA 제어판 애플리케이션에서 지원하는 대화 상자에 대한 리소스 정보 및 애플리케이션 정의 값을 포함합니다. (NEWCPLINFOA)
NEWCPLINFOW 제어판 애플리케이션에서 지원하는 대화 상자에 대한 리소스 정보 및 애플리케이션 정의 값을 포함합니다. (NEWCPLINFOW)
NOTIFYICONDATAA 시스템에서 알림 영역에 알림을 표시해야 하는 정보를 포함합니다. Shell_NotifyIcon 사용됩니다. (ANSI)
NOTIFYICONDATAW 시스템에서 알림 영역에 알림을 표시해야 하는 정보를 포함합니다. Shell_NotifyIcon 사용됩니다. (유니코드)
NOTIFYICONIDENTIFIER 경계 사각형을 검색할 아이콘을 식별하기 위해 Shell_NotifyIconGetRect 사용하는 정보를 포함합니다.
NRESARRAY CF_NETRESOURCE 클립보드 형식을 정의합니다.
NSTCCUSTOMDRAW INameSpaceTreeControlCustomDraw 메서드에서 사용하는 사용자 지정 그리기 구조체입니다.
NT_CONSOLE_PROPS IShellLinkDataList에서 사용하는 추가 데이터 블록을 보유합니다. 콘솔 속성을 보유합니다.
NT_FE_CONSOLE_PROPS IShellLinkDataList에서 사용하는 추가 데이터 블록을 보유합니다. 콘솔의 코드 페이지가 있습니다.
OPEN_PRINTER_PROPS_INFOA 프린터의 속성 페이지에서 특정 속성 시트를 식별하고 해당 속성 시트가 모달이어야 하는지 여부를 식별합니다. 필요에 따라 SHInvokePrinterCommand 함수와 함께 사용됩니다. (ANSI)

OPEN_PRINTER_PROPS_INFOW 프린터의 속성 페이지에서 특정 속성 시트를 식별하고 해당 속성 시트가 모달이어야 하는지 여부를 식별합니다. 필요에 따라 SHInvokePrinterCommand 함수와 함께 사용됩니다. (유니코드)
OPENASINFO SHOpenWithDialog 함수에 대한 정보를 저장합니다.
겹쳐진 비동기(겹치는) 입력/출력(I/O)에 사용되는 정보를 포함합니다.
PARSEDURLA ParseURL 함수에서 구문 분석된 URL을 반환하는 데 사용됩니다. (ANSI)
PARSEDURLW ParseURL 함수에서 구문 분석된 URL을 반환하는 데 사용됩니다. (유니코드)
PERSIST_FOLDER_TARGET_INFO 바로 가기 폴더의 대상 폴더 및 해당 특성을 지정합니다. 이 구조체는 IPersistFolder3::GetFolderTargetInfo 및 IPersistFolder3::InitializeEx에서 사용됩니다.
PREVIEWHANDLERFRAMEINFO 액셀러레이터 테이블 구조입니다. IPreviewHandlerFrame::GetWindowContext에서 사용됩니다.
PROFILEINFOA 사용자 프로필을 로드하거나 언로드할 때 사용되는 정보를 포함합니다. (ANSI)
PROFILEINFOW 사용자 프로필을 로드하거나 언로드할 때 사용되는 정보를 포함합니다. (유니코드)
PUBAPPINFO 애플리케이션 게시자에서 제어판의 프로그램 추가/제거에 게시된 애플리케이션에 대한 정보를 제공합니다.
QCMINFO 메뉴 항목을 Windows 탐색기 메뉴로 병합하기 위한 정보를 포함합니다.
QITAB

QISearch 함수에서 단일 인터페이스를 설명하는 데 사용됩니다.
SERIALIZEDPROPERTYVALUE 직렬화된 PROPVARIANT 구조를 나타내는 임의 형식의 메모리 범위입니다.
SFV_CREATE 이 구조체는 SHCreateShellFolderView 함수와 함께 사용됩니다.
SFV_SETITEMPOS 항목의 위치 정보를 저장합니다. 메시지 SFVM_SETITEMPOS 함께 사용됩니다.
SFVM_HELPTOPIC_DATA HTML 도움말 파일의 이름과 해당 파일의 토픽을 포함합니다. SFVM_GETHELPTOPIC 알림과 함께 사용됩니다. 이 구조에는 유니코드 문자열이 필요합니다.
SFVM_PROPPAGE_DATA 개체의 속성 시트에 추가할 페이지의 세부 정보를 포함합니다.
SHARDAPPIDINFO SHAddToRecentDocs에서 항목(이 경우 IShellItem)과 연결된 프로세스를 식별하는 데 사용하는 데이터를 포함합니다.
SHARDAPPIDINFOIDLIST SHAddToRecentDocs에서 항목(이 경우 PIDL(항목 식별자 목록)에 대한 절대 포인터 및 연결된 프로세스를 식별하는 데 사용하는 데이터를 포함합니다.
SHARDAPPIDINFOLINK 이 경우 IShellLink를 통해 항목과 연결된 프로세스를 식별하기 위해 SHAddToRecentDocs에서 사용하는 데이터를 포함합니다.
SHChangeDWORDAsIDList SHChangeDWORDAsIDList를 변경하거나 사용할 수 없습니다.
SHChangeNotifyEntry 변경 알림에 대한 정보를 포함하고 받습니다. 이 구조는 SHChangeNotifyRegister 함수 및 SFVM_QUERYFSNOTIFY 알림과 함께 사용됩니다.
SHChangeProductKeyAsIDList

SHChangeProductKeyAsIDList 구조체
SHChangeUpdateImageIDList SHChangeUpdateImageIDList를 변경하거나 사용할 수 없습니다.
SHCOLUMNDATA 특정 파일을 식별하는 정보를 포함합니다. 특정 파일에 대한 데이터를 요청할 때 IColumnProvider::GetItemData에서 사용됩니다.
SHCOLUMNINFO 열의 속성에 대한 정보를 포함합니다. IColumnProvider::GetColumnInfo에서 사용됩니다.
SHCOLUMNINIT IColumnProvider::Initialize에 초기화 정보를 전달합니다.
SHCREATEPROCESSINFOW 프로세스를 만드는 데 SHCreateProcessAsUserW에 필요한 정보를 포함합니다.
SHDESCRIPTIONID SHGetDataFromIDList 호출에 대한 응답으로 항목 데이터를 받습니다.
SHDRAGIMAGE 끌기 이미지를 만드는 데 필요한 정보를 포함합니다.
SHELL_ITEM_RESOURCE 셸 항목 리소스를 정의합니다.
SHELLDETAILS Shell 폴더의 항목에 대한 자세한 정보를 보고합니다.
SHELLEXECUTEINFOA ShellExecuteEx에서 사용하는 정보를 포함합니다. (ANSI)
SHELLEXECUTEINFOW ShellExecuteEx에서 사용하는 정보를 포함합니다. (유니코드)
SHELLFLAGSTATE 현재 셸 설정을 나타내는 플래그 집합을 포함합니다. 이 구조체는 SHGetSettings 함수와 함께 사용됩니다.

SHELLSTATEA 셸 상태에 대한 설정을 포함합니다. 이 구조체는 SHGetSetSettings 함수와 함께 사용됩니다. (ANSI)
SHELLSTATEW 셸 상태에 대한 설정을 포함합니다. 이 구조체는 SHGetSetSettings 함수와 함께 사용됩니다. (유니코드)
SHFILEINFOA 파일 개체에 대한 정보를 포함합니다. (ANSI)
SHFILEINFOW 파일 개체에 대한 정보를 포함합니다. (유니코드)
SHFILEOPSTRUCTA SHFileOperation 함수가 파일 작업을 수행하는 데 사용하는 정보를 포함합니다. (ANSI)
SHFILEOPSTRUCTW SHFileOperation 함수가 파일 작업을 수행하는 데 사용하는 정보를 포함합니다. (유니코드)
SHFOLDERCUSTOMSETTINGS 사용자 지정 폴더 설정을 보유합니다. 이 구조체는 SHGetSetFolderCustomSettings 함수와 함께 사용됩니다.
SHITEMID 항목 식별자를 정의합니다.
SHNAMEMAPPINGA SHFileOperation 함수에서 이동, 복사 또는 이름을 바꾼 각 파일에 대한 이전 및 새 경로 이름을 포함합니다. (ANSI)
SHNAMEMAPPINGW SHFileOperation 함수에서 이동, 복사 또는 이름을 바꾼 각 파일에 대한 이전 및 새 경로 이름을 포함합니다. (유니코드)
SHQUERYRBINFO SHQueryRecycleBin 함수에서 검색한 크기 및 항목 개수 정보를 포함합니다.
슈스톡코닌포

주식 셸 아이콘을 검색하는 데 사용되는 정보를 받습니다. 이 구조체는 SHGetStockIconInfo 호출에 사용됩니다.
SLOWAPPINFO 제어판에서 프로그램 추가/제거에 대한 특수 애플리케이션 정보를 제공합니다. 이 구조는 게시된 애플리케이션에는 적용되지 않습니다.
SMCSHCHANGENOTIFYSTRUCT 변경 알림에 대한 정보를 포함합니다. IShellMenuCallback::CallbackSM에서 사용됩니다.
SMDATA 메뉴 밴드의 정보를 포함합니다.
SMINFO 메뉴 밴드의 항목에 대한 정보를 포함합니다.
SOFTDISTINFO 소프트웨어 업데이트에 대한 정보를 포함합니다.
SORTCOLUMN 폴더 보기에 표시되는 열을 정렬하는 방법에 대한 정보를 저장합니다.
STRRET IShellFolder 인터페이스 메서드에서 반환된 문자열을 포함합니다.
SV2CVW2_PARAMS IShellView2::CreateViewWindow2 메서드에 대한 매개 변수를 보유합니다.
SYNCMGR_CONFLICT_ID_INFO 충돌 ID 정보 구조를 설명합니다.
SYNCMGRHANDLERINFO ISyncMgrSynchronize::GetHandlerInfo 메서드에서 사용할 처리기에 대한 정보를 제공합니다.
SYNCMGRITEM ISyncMgrEnumItems 인터페이스에서 열거되는 항목에 대한 정보를 제공합니다.
SYNCMGRLOGERRORINFO

ISyncMgrSynchronizeCallback::LogError 메서드에서 사용하기 위한 오류 정보를 제공합니다.
SYNCMGRPROGRESSITEM 동기화가 진행되는 동안 상태 정보를 제공합니다. 이 구조는 ISyncMgrSynchronizeCallback::P rogress 메서드와 함께 사용되며 단일 동기화 항목에 해당합니다.
TBINFO SFVM_GETBUTTONINFO 알림과 함께 도구 모음에 추가할 단추 수와 추가 방법을 지정하는 데 사용됩니다.
THUMBBUTTON ITaskbarList3 인터페이스의 메서드에서 창의 썸네일 표현에 포함된 도구 모음에 사용되는 단추를 정의하는 데 사용됩니다.
TOOLBARITEM Deprecated. IBrowserService2::_GetToolbarItem, IBrowserService2::v_MayGetNextToolbarFocus 및 IBrowserService2::_SetFocus 도구 모음 항목을 정의하는 데 사용되는 데이터입니다.
WALLPAPEROPT 배경 화면 표시 옵션을 포함합니다. IActiveDesktop 인터페이스의 멤버와 함께 사용됩니다.
WINDOWDATA 창 데이터를 저장합니다.
WTS_THUMBNAILID 시스템 썸네일 캐시의 축소판 그림에 대한 고유 식별자를 포함합니다.

Last updated on 2025. 10. 08.

클래식 데스크톱 애플리케이션에 대한 타일, 배지 및 알림

아티클2023. 08. 22.

클래식 데스크톱 애플리케이션 기술에 대한 타일, 배지 및 알림 개요입니다.

클래식 데스크톱 애플리케이션에 대한 타일, 배지 및 알림을 개발하려면 다음 헤더가 필요합니다.

- notificationactivationcallback.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 클래식 데스크톱 애플리케이션에 대한 타일, 배지 및 알림

Functions

활성화 사용자가 알림 센터에서 알림과 상호 작용할 때 호출됩니다.

인터페이스

INotificationActivationCallback 알림 센터에서 알림을 통해 앱이 트리거될 때 알림 메시지를 받습니다.

구조체

NOTIFICATION_USER_INPUT_DATA 사용자가 알림 센터에서 알림 메시지와 상호 작용하는 방법에 대한 정보를 포함합니다. 이 구조체는 Activate에서 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

타이틀 호출 가능 UI

아티클2023. 04. 27.

타이틀 호출 가능 UI 기술 개요입니다.

타이틀 호출 가능 UI를 개발하려면 다음 헤더가 필요합니다.

- gamingtcui.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 타이틀 호출 가능 UI

열거형

KnownGamingPrivileges 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (KnownGamingPrivileges)

Functions

CheckGamingPrivilegeSilently 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (CheckGamingPrivilegeSilently)
CheckGamingPrivilegeWithUI 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (CheckGamingPrivilegeWithUI)
GameUICompletionRoutine 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (GameUICompletionRoutine)
PlayerPickerUICompletionRoutine 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (PlayerPickerUICompletionRoutine)
ProcessPendingGameUI 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (ProcessPendingGameUI)

ShowChangeFriendRelationshipUI 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (ShowChangeFriendRelationshipUI)
ShowChangeFriendRelationshipUIForUser 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (ShowChangeFriendRelationshipUIForUser)
ShowGameInviteUI 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (ShowGameInviteUI)
ShowPlayerPickerUI 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (ShowPlayerPickerUI)
ShowProfileCardUI 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (ShowProfileCardUI)
ShowProfileCardUIForUser 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (ShowProfileCardUIForUser)
ShowTitleAchievementsUI 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (ShowTitleAchievementsUI)
TryCancelPendingGameUI 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (TryCancelPendingGameUI)

도구 도움말 라이브러리

아티클2023. 08. 22.

도구 도움말 라이브러리 기술 개요입니다.

도구 도움말 라이브러리를 개발하려면 다음 헤더가 필요합니다.

- tlhelp32.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 도구 도움말 라이브러리

Functions

CreateToolhelp32Snapshot 이러한 프로세스에서 사용되는 힙, 모듈 및 스레드뿐만 아니라 지정된 프로세스의 스냅샷 사용합니다.
Heap32First 프로세스에서 할당한 힙의 첫 번째 블록에 대한 정보를 검색합니다.
Heap32ListFirst 지정된 프로세스에 의해 할당된 첫 번째 힙에 대한 정보를 검색합니다.
Heap32ListNext 프로세스에서 할당한 다음 힙에 대한 정보를 검색합니다.
Heap32Next 프로세스에서 할당한 힙의 다음 블록에 대한 정보를 검색합니다.
Module32First Module32First 함수(tlhelp32.h)는 프로세스와 연결된 첫 번째 모듈에 대한 정보를 검색합니다.
Module32FirstW Module32FirstW(유니코드) 함수(tlhelp32.h)는 프로세스와 연결된 첫 번째 모듈에 대한 정보를 검색합니다.

Module32Next Module32Next 함수(tlhelp32.h)는 프로세스 또는 스레드와 연결된 다음 모듈에 대한 정보를 검색합니다.
Module32NextW Module32NextW(유니코드) 함수(tlhelp32.h)는 프로세스 또는 스레드와 연결된 다음 모듈에 대한 정보를 검색합니다.
Process32First Process32First 함수(tlhelp32.h)는 시스템 스냅샷 발생한 첫 번째 프로세스에 대한 정보를 검색합니다.
Process32FirstW Process32FirstW(유니코드) 함수(tlhelp32.h)는 시스템 스냅샷 발생한 첫 번째 프로세스에 대한 정보를 검색합니다.
Process32Next Process32Next 함수(tlhelp32.h)는 시스템 스냅샷 기록된 다음 프로세스에 대한 정보를 검색합니다.
Process32NextW Process32NextW(유니코드) 함수(tlhelp32.h)는 시스템 스냅샷 기록된 다음 프로세스에 대한 정보를 검색합니다.
Thread32First 시스템 스냅샷 발생한 프로세스의 첫 번째 스레드에 대한 정보를 검색합니다.
Thread32Next 시스템 메모리 스냅샷 발생한 프로세스의 다음 스레드에 대한 정보를 검색합니다.
Toolhelp32ReadProcessMemory 다른 프로세스에 할당된 메모리를 애플리케이션 제공 버퍼에 복사합니다.

구조체

HEAPENTRY32

검사 중인 힙의 한 항목(블록)을 설명합니다.
HEAPLIST32 지정된 프로세스에서 사용하는 힙을 열거하는 목록의 항목을 설명합니다.
MODULEENTRY32 MODULEENTRY32(tlhelp32.h) 구조체는 지정된 프로세스에 속하는 모듈 목록의 항목을 설명합니다.
MODULEENTRY32W MODULEENTRY32W(유니코드) 구조체(tlhelp32.h)는 지정된 프로세스에 속하는 모듈 목록의 항목을 설명합니다.
PROCESSENTRY32 PROCESSENTRY32(tlhelp32.h) 구조체는 스냅샷 수행되었을 때 시스템 주소 공간에 있는 프로세스 목록의 항목을 설명합니다.
PROCESSENTRY32W PROCESSENTRY32W(유니코드) 구조체(tlhelp32.h)는 스냅샷 수행되었을 때 시스템 주소 공간에 있는 프로세스 목록의 항목을 설명합니다.
THREADENTRY32 스냅샷 수행되었을 때 시스템에서 실행되는 스레드 목록의 항목을 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

터치 적중 테스트

아티클2023. 08. 22.

터치 적중 테스트 기술 개요입니다.

터치 적중 테스트 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 터치 적중 테스트

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

터치 삽입

아티클2023. 03. 09.

터치 삽입 기술의 개요입니다.

터치 삽입 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 터치 삽입

터치 입력

아티클2023. 08. 22.

터치 입력 기술 개요입니다.

터치 입력을 개발하려면 다음 헤더가 필요합니다.

- manipulations.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 터치 입력

열거형

MANIPULATION_PROCESSOR_MANIPULATIONS MANIPULATION_PROCESSOR_MANIPULATIONS 열거형은 대상 개체에 적용할 수 있는 다양한 종류의 조작입니다.

Functions

CloseGestureInfoHandle 제스처 정보 핸들과 연결된 리소스를 닫습니다.
CloseTouchInputHandle 터치 입력 핸들을 닫고, 연결된 프로세스 메모리를 해제하고, 핸들을 무효화합니다.
완료 Complete 메서드는 현재 조작을 완료하고 관성 프로세서에서 관성 상태를 중지합니다.
CompleteManipulation CompleteManipulation 메서드는 개발자가 조작을 종료하도록 선택할 때 호출됩니다.
CompleteTime

지정된 틱에서 현재 조작을 완료하고 관성 프로세서에서 관성을 중지한 다음 ManipulationCompleted 이벤트를 발생합니다.
get_BoundaryBottom BoundaryBottom 속성은 대상 개체가 이동할 수 있는 화면 아래쪽까지의 범위를 제한합니다. (Get)
get_BoundaryLeft BoundaryLeft 속성은 대상 개체가 이동할 수 있는 화면 왼쪽까지의 범위를 제한합니다. (Get)
get_BoundaryRight BoundaryRight 속성은 대상 개체가 이동할 수 있는 화면 오른쪽까지의 범위를 제한합니다. (Get)
get_BoundaryTop BoundaryTop 속성은 대상 개체가 이동할 수 있는 화면 위쪽까지의 범위를 제한합니다. (Get)
get_DesiredAngularDeceleration DesiredAngularDeceleration 속성은 대상 개체가 제곱당 라디안으로 회전을 중지하는 원하는 속도를 지정합니다. (Get)
get_DesiredDeceleration DesiredDeceleration 속성은 번역 작업이 감속되는 원하는 속도를 지정합니다. (Get)
get_DesiredDisplacement DesiredDisplacement 속성은 개체가 이동할 원하는 거리를 지정합니다. (Get)
get_DesiredExpansion DesiredExpansion 속성은 개체의 평균 반경에서 원하는 변경 사항을 지정합니다. (Get)
get_DesiredExpansionDeceleration DesiredExpansionDeceleration 속성은 개체 확장이 중지되는 속도를 지정합니다. (Get)
get_DesiredRotation DesiredRotation 속성은 현재 관성 프로세서 개체가 대상 개체를 라디안으로 조작할 정도를 지정합니다. (Get)
get_ElasticMarginBottom ElasticMarginBottom 속성은 대상 개체를 튀기는 아래쪽 영역을 지정합니다. (Get)

get_ElasticMarginLeft ElasticMarginLeft 속성은 대상 개체를 튀기는 가장 왼쪽 영역을 지정합니다. (Get)
get_ElasticMarginRight ElasticMarginRight 속성은 대상 개체를 튀기는 가장 오른쪽 영역을 지정합니다. (Get)
get_ElasticMarginTop ElasticMarginTop 속성은 대상 개체를 튀기는 가장 맨 위 영역을 지정합니다. (Get)
get_InitialAngularVelocity InitialAngularVelocity 속성은 이동이 시작될 때 대상의 회전(각도) 속도를 지정합니다. (Get)
get_InitialExpansionVelocity InitialExpansionVelocity 속성은 대상이 관성의 영향을 받은 경우 대상의 반지름 확장 속도를 지정합니다. (Get)
get_InitialOriginX InitialOriginX 속성은 관성을 가진 대상의 시작 가로 위치를 지정합니다. (Get)
get_InitialOriginY InitialOriginY 속성은 관성을 가진 대상의 시작 세로 위치를 지정합니다. (Get)
get_InitialRadius InitialRadius 속성은 개체가 변경되기 전까지 대상의 가장자리에서 중심까지의 거리를 지정합니다. (Get)
get_InitialTimestamp InitialTimestamp 속성은 관성 대상 개체의 시작 타임스탬프를 지정합니다. (Get)
get_InitialVelocityX InitialVelocityX 속성은 가로 축에서 대상 개체의 초기 이동을 지정합니다. (Get)
get_InitialVelocityY InitialVelocityY 속성은 세로 축에서 대상 개체의 초기 이동을 지정합니다. (Get)
get_MinimumScaleRotateRadius 조작을 트리거하기 위해 눈금 또는 회전 제스처의 거리 접점 크기를 지정합니다. (Get)

get_PivotPointX PivotPointX 속성은 개체의 가로 가운데입니다. (Get)
get_PivotPointY PivotPointY 속성은 개체의 세로 가운데입니다. (Get)
get_PivotRadius PivotRadius 속성은 한 손가락 조작에 사용되는 회전의 양을 결정하는 데 사용됩니다. (Get)
get_SupportedManipulations SupportedManipulations 속성은 개체에서 지원되는 조작을 나타내는 데 사용됩니다. (Get)
GetAngularVelocity GetAngularVelocity 메서드는 대상 개체가 이동하는 회전 속도를 계산합니다.
GetExpansionVelocity GetExpansionVelocity 메서드는 대상 개체가 확장되는 속도를 계산합니다.
GetGestureConfig 창에서 Windows Touch 제스처 메시지가 전송되는 구성을 검색합니다.
GetGestureExtraArgs GESTUREINFO 핸들에서 제스처에 대한 추가 정보를 검색합니다.
GetGestureInfo 제스처 정보에 대한 핸들이 지정된 GESTUREINFO 구조를 검색합니다.
GetTouchInputInfo 특정 터치 입력 핸들과 연결된 터치 입력에 대한 자세한 정보를 검색합니다.
GetVelocityX 대상 개체의 가로 속도를 계산하고 반환합니다.
GetVelocityY 세로 속도를 계산하고 반환합니다.
GID_ROTATE_ANGLE_FROM_ARGUMENT

GID_ROTATE_ANGLE_FROM_ARGUMENT 매크로는 WM_GESTURE 구조체에서 값을 받을 때 GID_ROTATE ullArgument 값을 해석하는 데 사용됩니다.
GID_ROTATE_ANGLE_TO_ARGUMENT 회전 제스처 메시지의 인수로 라디안 값을 변환합니다.
IsTouchWindow 지정된 창이 터치 가능인지 여부를 확인하고 필요에 따라 창의 터치 기능에 대해 설정된 한정자 플래그를 검색합니다.
ManipulationCompleted 조작 또는 관성 완료 시 이벤트를 처리합니다.
ManipulationDelta 조작된 개체가 변경될 때 발생하는 이벤트를 처리합니다.
ManipulationStarted 조작 또는 관성 시작 시 에 대한 이벤트를 처리합니다.
처리 Process 메서드는 계산을 수행하고 외삽이 완료되었는지 여부에 따라 Started, Delta 또는 Completed 이벤트를 발생할 수 있습니다. 이전 틱에서 외삽이 완료된 경우 메서드는 no-op입니다.
ProcessDown ProcessDown 메서드는 터치 다운 데이터를 대상과 연결된 조작 프로세서에 공급합니다.
ProcessDownWithTime 피드는 타임스탬프를 포함한 데이터를 대상과 연결된 조작 프로세서에 터치 다운합니다.
ProcessMove ProcessMove 메서드는 대상 개체의 이동 데이터를 해당 조작 프로세서에 공급합니다.
ProcessMoveWithTime 대상 개체에 대한 타임스탬프를 포함한 이동 데이터를 조작 프로세서에 공급합니다.
ProcessTime ProcessTime 메서드는 지정된 틱에 대한 계산을 수행하고 외삽이 완료되었는지 여부에 따라

Started, Delta 또는 Completed 이벤트를 발생할 수 있습니다. 이전 틱에서 외삽이 완료된 경우 메서드는 no-op입니다.
ProcessUp ProcessUp 메서드는 터치업 시퀀스를 위해 대상의 조작 프로세서에 데이터를 공급합니다.
ProcessUpWithTime 터치업 시퀀스를 위해 타임스탬프를 포함한 데이터를 대상의 조작 프로세서에 공급합니다.
put_BoundaryBottom BoundaryBottom 속성은 대상 개체가 이동할 수 있는 화면 아래쪽까지의 범위를 제한합니다. (Put)
put_BoundaryLeft BoundaryLeft 속성은 대상 개체가 이동할 수 있는 화면 왼쪽까지의 범위를 제한합니다. (Put)
put_BoundaryRight BoundaryRight 속성은 대상 개체가 이동할 수 있는 화면 오른쪽까지의 범위를 제한합니다. (Put)
put_BoundaryTop BoundaryTop 속성은 대상 개체가 이동할 수 있는 화면 위쪽까지의 범위를 제한합니다. (Put)
put_DesiredAngularDeceleration DesiredAngularDeceleration 속성은 대상 개체가 제곱당 라디안으로 회전을 중지하는 원하는 속도를 지정합니다. (Put)
put_DesiredDeceleration DesiredDeceleration 속성은 번역 작업이 감속되는 원하는 속도를 지정합니다. (Put)
put_DesiredDisplacement DesiredDisplacement 속성은 개체가 이동할 원하는 거리를 지정합니다. (Put)
put_DesiredExpansion DesiredExpansion 속성은 개체의 평균 반경에서 원하는 변경 사항을 지정합니다. (Put)
put_DesiredExpansionDeceleration DesiredExpansionDeceleration 속성은 개체 확장이 중지되는 속도를 지정합니다. (Put)
put_DesiredRotation

DesiredRotation 속성은 현재 관성 프로세서 개체가 대상 개체를 라디안으로 조작할 정도를 지정합니다. (Put)
put_ElasticMarginBottom ElasticMarginBottom 속성은 대상 개체를 튀기는 아래쪽 영역을 지정합니다. (Put)
put_ElasticMarginLeft ElasticMarginLeft 속성은 대상 개체를 튀기는 가장 왼쪽 영역을 지정합니다. (Put)
put_ElasticMarginRight ElasticMarginRight 속성은 대상 개체를 튀기는 가장 오른쪽 영역을 지정합니다. (Put)
put_ElasticMarginTop ElasticMarginTop 속성은 대상 개체를 튀기는 가장 맨 위 영역을 지정합니다. (Put)
put_InitialAngularVelocity InitialAngularVelocity 속성은 이동이 시작될 때 대상의 회전(각도) 속도를 지정합니다. (Put)
put_InitialExpansionVelocity InitialExpansionVelocity 속성은 대상이 관성의 영향을 받은 경우 대상의 반지름 확장 속도를 지정합니다. (Put)
put_InitialOriginX InitialOriginX 속성은 관성을 가진 대상의 시작 가로 위치를 지정합니다. (Put)
put_InitialOriginY InitialOriginY 속성은 관성을 가진 대상의 시작 세로 위치를 지정합니다. (Put)
put_InitialRadius InitialRadius 속성은 개체가 변경되기 전까지 대상 가장자리에서 중심까지의 거리를 지정합니다. (Put)
put_InitialTimestamp InitialTimestamp 속성은 관성 있는 대상 개체의 시작 타임스탬프를 지정합니다. (Put)
put_InitialVelocityX InitialVelocityX 속성은 가로 축에서 대상 개체의 초기 이동을 지정합니다. (Put)

put_InitialVelocityY InitialVelocityY 속성은 세로 축에서 대상 개체의 초기 이동을 지정합니다. (Put)
put_MinimumScaleRotateRadius 조작을 트리거하는 데 필요한 크기 조정 또는 회전 제스처의 거리 접점 크기를 지정합니다. (Put)
put_PivotPointX PivotPointX 속성은 개체의 가로 중심입니다. (Put)
put_PivotPointY PivotPointY 속성은 개체의 세로 중심입니다. (Put)
put_PivotRadius PivotRadius 속성은 한 손가락 조작에 사용되는 회전 수를 결정하는 데 사용됩니다. (Put)
put_SupportedManipulations SupportedManipulations 속성은 개체에서 지원되는 조작을 나타내는 데 사용됩니다. (Put)
RegisterTouchWindow 창을 터치 가능으로 등록합니다.
Reset Reset 메서드는 초기 타임스탬프를 사용하여 프로세서를 초기화하고 관성으로 다시 시작합니다.
SetGestureConfig Windows Touch 제스처에 대한 창에서 전송되는 메시지를 구성합니다.
TOUCH_COORD_TO_PIXEL 터치 좌표를 픽셀로 변환합니다.
UnregisterTouchWindow 창을 더 이상 터치 가능하지 않은 것으로 등록합니다.

인터페이스

_IManipulationEvents 조작 및 관성 이벤트를 처리합니다.
IInertiaProcessor IInertiaProcessor 인터페이스는 Windows Touch의 개체 동작에 대한 계산을 처리합니다.
IManipulationProcessor IManipulationProcessor는 멀티 터치 입력을 모니터링하고 응답하는 기능을 제공합니다.

구조체

GESTURECONFIG 제스처 메시지 및 이 구성의 형식을 사용하도록 설정하기 위한 구성을 가져오고 설정합니다.
GESTUREINFO 제스처에 대한 정보를 저장합니다.
GESTURENOTIFYSTRUCT WM_GESTURENOTIFY 메시지와 함께 전송되는 경우 제스처에 대한 정보를 전달합니다.
TOUCHINPUT 터치 입력에 대한 데이터를 캡슐화합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

TPM 기본 서비스

아티클2023. 08. 22.

TPM 기본 서비스 기술 개요입니다.

TPM 기본 서비스를 개발하려면 다음 헤더가 필요합니다.

- tbs.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- TPM 기본 서비스

Functions

Tbsi_Context_Create TBS에 명령을 전달하는 데 사용할 수 있는 컨텍스트 핸들을 만듭니다.
Tbsi_Get_OwnerAuth 로컬 레지스트리에서 정보를 사용할 수 있는 경우 TPM의 소유자 권한 부여를 검색합니다.
Tbsi_Get_TCG_Log TCG 로그라고도 하는 최신 WBCL(Windows 부팅 구성 로그)을 검색합니다.
Tbsi_Get_TCG_Log_Ex 지정된 형식의 TCG 로그라고도 하는 WBCL(Windows 부팅 구성 로그)을 가져옵니다.
Tbsi_GetDeviceInfo 컴퓨터에서 TPM의 버전을 가져옵니다.
Tbsi_Physical_Presence_Command TBS를 통해 물리적 현재 상태 ACPI 명령을 드라이버에 전달합니다.
Tbsi_Revoke_Attestation ELAM 드라이버가 정책 위반(예: 루트킷)을 감지하면 PCR을 무효화합니다.
Tbsip_Cancel_Commands

지정된 컨텍스트에 대한 모든 미해결 명령을 취소합니다.
Tbsip_Context_Close 컨텍스트 핸들을 닫습니다. 이 핸들은 TBS의 컨텍스트와 연결된 리소스를 해제하고 TBS와 통신하는 데 사용되는 바인딩 핸들을 닫습니다.
Tbsip_Submit_Command 처리를 위해 TPM(신뢰할 수 있는 플랫폼 모듈) 명령을 TBS(TPM 기본 서비스)에 제출합니다.

구조체

TBS_CONTEXT_PARAMS TBS 컨텍스트 구현의 버전을 지정합니다.
TBS_CONTEXT_PARAMS2 TBS 컨텍스트 구현의 버전을 지정합니다. 애플리케이션이 두 버전의 TPM에서 작동하는 경우 이 구조를 사용해야 합니다.
TPM_DEVICE_INFO TPM 버전에 대한 정보를 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

TraceLogging

아티클2023. 03. 11.

TraceLogging 기술 개요입니다.

TraceLogging을 개발하려면 다음 헤더가 필요합니다.

- traceloggingactivity.h
- traceloggingprovider.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- TraceLogging

클래스

TraceLoggingActivity 활동 중에 ETW 이벤트 로깅을 지원합니다. 모든 이벤트는 수동으로 태그를 지정하거나 중첩해야 합니다.
TraceLoggingThreadActivity 활동 중에 ETW 이벤트 로깅을 지원합니다. 이벤트는 이 활동에서 자동으로 태그가 지정되거나 중첩됩니다.
TraceLoggingThreadActivityIdSetter ETW가 해당 스레드의 모든 이벤트를 활동 ID로 표시할 수 있도록 작업 ID를 사용하여 스레드에 태그를 지정합니다.

함수

~TraceLoggingThreadActivityIdSetter 원래 활동 ID를 스레드로 복원합니다.

공급 기업 TraceLoggingActivity::P rovider(traceloggingactivity.h)는 이 활동과 연결된 TraceLogging 공급자에 핸들을 반환합니다.
공급 기업 TraceLoggingThreadActivity::P rovider(traceloggingactivity.h)는 이 활동과 연결된 TraceLogging 공급자에 핸들을 반환합니다.
SetRelatedActivity 이 TraceLoggingActivity 개체에 대한 관련 작업을 설정합니다.
SetRelatedActivityId TraceLoggingActivity::SetRelatedActivityId는 활동의 고유 식별자를 사용하여 이 TraceLoggingActivity 개체에 대한 관련 활동을 설정합니다.
SetRelatedActivityId TraceLoggingActivity::SetRelatedActivityId(const GUID)는 활동의 고유 식별자를 사용하여 이 TraceLoggingActivity 개체에 대한 관련 작업을 설정합니다.
TRACELOGGING_DECLARE_PROVIDER TraceLogging 공급자에 대한 핸들을 전달 선언합니다.
TRACELOGGING_DEFINE_PROVIDER TraceLogging 공급자에 대한 핸들을 정의합니다.
TRACELOGGING_DEFINE_PROVIDER_STORAGE 사용자가 정의할 TraceLogging 공급자 핸들에 대한 정적 스토리지를 예약합니다. 이 매크로보다 TRACELOGGING_DEFINE_PROVIDER 것이 좋습니다.
TraceLoggingActivity 새 TraceLoggingActivity 개체를 만듭니다. (오버로드 1/2)
TraceLoggingActivity 새 TraceLoggingActivity 개체를 만듭니다. (오버로드 2/2)
TraceLoggingBinary 이벤트에 이진 데이터가 있는 필드를 추가하는 TraceLogging 래퍼 매크로입니다.

TraceLoggingChannel 이벤트의 채널을 설정하는 TraceLogging 래퍼 매크로입니다.
TraceLoggingCustom 사용자 지정 serializer를 사용하여 압축된 필드를 이벤트에 추가하는 TraceLogging 래퍼 매크로입니다.
TraceLoggingCustomAttribute 이벤트에 대한 사용자 지정 정보를 PDB에 추가하는 TraceLogging 래퍼 매크로입니다.
TraceLoggingDescription 이벤트에 대한 설명을 설정하는 TraceLogging 래퍼 매크로입니다.
TraceLoggingEventTag 이벤트에 대한 이벤트 태그를 설정하는 TraceLogging 래퍼 매크로입니다.
TraceLoggingFunction 현재 함수의 이름을 따서 명명된 TraceLoggingThreadActivity를 만들고 활동에 대한 시작 이벤트를 씁니다. 중지 작업은 현재 범위의 끝에 기록됩니다.
TraceLoggingKeyword 이벤트에 대한 키워드를 설정하는 TraceLogging 래퍼 매크로입니다.
TraceLoggingLevel 이벤트의 수준을 설정하는 TraceLogging 래퍼 매크로
TraceLoggingOpcode 이벤트에 대한 opcode를 설정하는 TraceLogging 래퍼 매크로
TraceLoggingOptionGroup 공급자 그룹을 지정하기 위해 TRACELOGGING_DEFINE_PROVIDER 사용할 추적 로깅 매크로입니다.
TraceLoggingRegisterEx ETW 사용 콜백을 지정하여 이벤트를 기록하는 데 사용할 수 있도록 TraceLogging 공급자를 등록합니다.

TraceLoggingSocketAddress 소켓 주소가 있는 필드를 이벤트에 추가하는 TraceLogging 래퍼 매크로입니다.
TraceLoggingStruct 이벤트에 다른 필드가 포함된 필드를 추가하는 TraceLogging 래퍼 매크로입니다.
TraceLoggingThreadActivity TraceLoggingThreadActivity 클래스의 새 인스턴스를 초기화합니다. (오버로드 2/2)
TraceLoggingThreadActivity TraceLoggingThreadActivity 클래스의 새 인스턴스를 초기화합니다. (오버로드 1/2)
TraceLoggingThreadActivityIdSetter 새 TraceLoggingThreadActivityIdSetter 개체를 만듭니다. (오버로드 2/2)
TraceLoggingThreadActivityIdSetter 새 TraceLoggingThreadActivityIdSetter 개체를 만듭니다. (오버로드 1/2)
TraceLoggingValue 이벤트에 자동으로 추론된 형식의 필드를 추가하는 C++에 대한 TraceLogging 래퍼 매크로입니다.
TraceLoggingWrite TraceLogging 이벤트를 내보낸다.
TraceLoggingWriteActivity 지정된 활동 ID를 사용하여 TraceLogging 이벤트를 내보낸다.
TraceLoggingWriteStart 활동을 시작하고 시작 이벤트를 기록합니다.
TraceLoggingWriteStop 활동을 중지하고 중지 이벤트를 기록합니다.
TraceLoggingWriteTagged 연결된 ETW 활동 ID를 사용하여 이벤트를 기록합니다.

UPnP API

아티클2023. 03. 16.

UPnP API 기술 개요입니다.

UPnP API를 개발하려면 다음 헤더가 필요합니다.

- upnp.h
- upnphost.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- UPnP API

Functions

중단 Abort 메서드는 IUPnPDescriptionDocument::LoadAsync에서 시작된 비동기 로드 작업을 중지합니다.
AddCallback AddCallback 메서드는 UPnP 프레임워크에 애플리케이션의 콜백을 등록합니다.
AddRequestHeaders 제공된 HTTP 헤더를 HTTP 요청에 추가합니다.
조언 Advise 메서드는 호스트된 서비스에서 이벤트 수신을 시작하기 위해 디바이스 호스트에서 호출됩니다.
AsyncOperationComplete AsyncOperationComplete 콜백 메서드는 비동기 I/O 작업의 완료에 대한 알림을 제공합니다.
BeginInvokeAction BeginInvokeAction 메서드는 비동기 모드의 디바이스에서 작업을 호출합니다. 또한 지연된 SCPD 다운로드 및 이벤트 구독이 옵트인되고 아직 수행되지 않은 경우 이 메서드는 SCPD 다운로드를 시작합니다.

BeginQueryStateVariable BeginQueryStateVariable 메서드는 특정 서비스의 상태 변수 값에 대한 비동기 요청을 시작합니다.
BeginSCPDDownload BeginSCPDDownload 메서드는 SCPD(서비스 제어 프로토콜 설명) 문서의 비동기 다운로드를 시작합니다.
BeginSubscribeToEvents BeginSubscribeToEvents는 비동기 모드에서 이벤트 구독을 시작하고 UPnP 프레임워크에 애플리케이션 콜백을 등록합니다.
CancelAsyncFind CancelAsyncFind 메서드는 비동기 검색을 취소합니다.
CancelAsyncOperation CancelAsyncOperation 메서드는 BeginInvokeAction, BeginQueryStateVariable, BeginSubscribeToEvents 또는 BeginSCPDDownload 메서드에 의해 시작된 보류 중인 비동기 작업을 취소합니다.
CreateAsyncFind CreateAsyncFind 메서드는 비동기 검색 작업을 만듭니다.
DeviceAdded DeviceAdded 메서드는 디바이스가 네트워크에 추가되었음을 애플리케이션에 알리기 위해 UPnP 프레임워크에 의해 호출됩니다.
DeviceAddedWithInterface DeviceAddedWithInterface 메서드는 UPnP 프레임워크에 의해 호출되어 디바이스가 네트워크에 추가되었음을 애플리케이션에 알립니다.
DeviceByUDN DeviceByUDN 메서드는 로드된 설명 문서에 포함된 지정된 UDN(고유 디바이스 이름)을 사용하여 디바이스를 반환합니다.
DeviceRemoved DeviceRemoved 메서드는 UPnP 프레임워크에서 호출되어 디바이스가 네트워크에서 제거되었음을 애플리케이션에 알립니다.

EndInvokeAction EndInvokeAction 메서드는 이전 BeginInvokeAction 작업의 결과를 검색하고 결과 출력 인수를 검색합니다.
EndQueryStateVariable EndQueryStateVariable 메서드는 이전 BeginQueryStateVariable 작업의 결과를 검색하고 결과 서비스별 상태 변수 값을 검색합니다.
EndSCPDDownload EndSCPDDownload 메서드는 SCPD(서비스 제어 프로토콜 설명) 문서의 이전 비동기 다운로드 결과를 검색합니다.
EndSubscribeToEvents EndSubscribeToEvents 메서드는 이전 BeginSubscribeToEvents 작업의 결과를 검색합니다.
FindByType FindByType 메서드는 디바이스 유형 또는 서비스 유형별로 디바이스를 동기적으로 검색합니다.
FindByUDN FindByUDN 메서드는 UDN(고유한 디바이스 이름)으로 디바이스를 동기적으로 검색합니다.
get__NewEnum _NewEnum 속성은 컬렉션에 대한 IEnumVARIANT 또는 IEnumUnknown 열거자 인터페이스를 지정합니다. (IUPnPDevices.get__NewEnum)
get__NewEnum _NewEnum 속성은 컬렉션에 대한 IEnumVARIANT 또는 IEnumUnknown 열거자 인터페이스를 지정합니다. (IUPnPServices.get__NewEnum)
get_Children Children 속성은 디바이스의 모든 자식 디바이스를 지정합니다. 디바이스는 IUPnPDevices 컬렉션에 저장됩니다.
get_Count Count 속성은 컬렉션의 디바이스 수를 지정합니다.
get_Count Count 속성은 컬렉션의 서비스 수를 지정합니다.

get_Description Description 속성은 디바이스 기능에 대한 사람이 읽을 수 있는 요약을 지정합니다.
get_FriendlyName FriendlyName 속성은 디바이스의 디바이스 표시 이름을 지정합니다.
get_HasChildren HasChildren 속성은 디바이스에 자식 디바이스가 있는지 여부를 지정합니다.
get_Id Id 속성은 서비스의 서비스 ID를 지정합니다.
get_IsRootDevice IsRootDevice 속성은 디바이스가 디바이스 트리의 최상위 디바이스인지 여부를 지정합니다.
get_Item Item 속성은 컬렉션에서 UDN으로 식별되는 디바이스에 대한 IUPnPDevice 인터페이스를 지정합니다.
get_Item Item 속성은 컬렉션에서 서비스 ID로 식별되는 서비스에 대한 IUPnPService 인터페이스를 지정합니다.
get_LastTransportStatus 이벤트 변수와 관련된 쿼리의 경우 LastTransportStatus 속성은 마지막 IUPnPService::InvokeAction 작업의 HTTP 상태를 지정합니다.
get_LoadResult LoadResult 속성은 완료된 로드 작업의 성공 또는 실패 코드를 지정합니다.
get_ManufacturerName ManufacturerName 속성은 사람이 읽을 수 있는 디바이스의 제조업체 이름 형식을 지정합니다.
get_ManufacturerURL ManufacturerURL 속성은 제조업체의 웹 사이트에 대한 URL을 지정합니다.

get_ModelName ModelName 속성은 사람이 읽을 수 있는 디바이스의 모델 이름 형식을 지정합니다.
get_ModelNumber ModelNumber 속성은 사람이 읽을 수 있는 디바이스의 모델 번호 형식을 지정합니다.
get_ModelURL ModelURL 속성은 디바이스에 대한 모델별 정보를 포함하는 웹 페이지의 URL을 지정합니다.
get_ParentDevice ParentDevice 속성은 디바이스의 부모를 지정합니다.
get_PresentationURL PresentationURL 속성은 디바이스를 제어하는 웹 페이지의 프레젠테이션 URL을 지정합니다.
get_ReadyState ReadyState 속성은 문서 로드 작업의 상태를 지정합니다.
get_RootDevice RootDevice 속성은 디바이스 트리에서 최상위 디바이스를 지정합니다. 루트 디바이스는 물리적 개체를 나타냅니다.
get_SerialNumber SerialNumber 속성은 사람이 읽을 수 있는 디바이스 일련 번호 형식을 지정합니다.
get_Services Services 속성은 디바이스에서 제공하는 서비스 목록을 지정합니다.
get_ServiceTypeIdentifier ServiceTypeIdentifier 속성은 디바이스에 대한 서비스 유형 식별자를 지정합니다.
get_Type Type 메서드는 디바이스에 대한 디바이스 유형 URI(Uniform Resource Identifier)를 지정합니다.
get_UniqueDeviceName UniqueDeviceName 속성은 디바이스의 고유한 UDN(디바이스 이름)을 지정합니다. UDN은 고유합니다. 두 디바이스가 동일한 UDN을 가질 수 없습니다.

get_UPC UPC 속성은 사람이 읽을 수 있는 제품 코드 형식을 지정합니다.
GetAddressFamily GetAddressFamily 메서드는 Device Finder 개체의 주소 패밀리 플래그의 현재 값을 검색합니다.
GetDocument UPnP 디바이스에 대한 XML 디바이스 설명 문서를 검색합니다.
GetDocument GetDocument 메서드는 서비스 개체에 대한 SCPD(서비스 제어 프로토콜 설명) 문서를 검색합니다.
GetDocumentURL GetDocumentURL 메서드는 디바이스 설명 문서를 로드할 수 있는 URL을 반환합니다.
GetDocumentURL GetDocumentURL 메서드는 서비스 개체에 대한 SCPD(서비스 제어 프로토콜 설명) URL을 검색합니다. UPnP 제어 지점은 이 URL을 사용하여 전체 SCPD 문서를 다운로드할 수 있습니다.
GetDwordValue GetDwordValue 메서드는 요청 또는 요청자에 대한 정보를 제공하는 4 바이트 값을 가져옵니다.
GetGuidValue GetGuidValue 메서드는 현재 지원되지 않습니다.
GetServiceObject GetServiceObject 메서드는 특정 서비스 개체에 대한 IDispatch 포인터를 가져오는 데 사용됩니다. 디바이스 호스트는 서비스에 대한 요청을 처음 수신할 때 서비스당 한 번씩 이 메서드를 호출합니다.
GetStringValue GetStringValue 메서드는 요청 또는 요청자에 대한 정보를 제공하는 문자열을 가져옵니다.
GetUniqueDeviceName GetUniqueDeviceName 메서드는 지정된 디바이스에 대한 UDN을 검색합니다.

IconURL IconURL 메서드는 지정된 형식의 아이콘을 로드할 수 있는 URL을 반환합니다.
초기화 Initialize 메서드는 디바이스를 초기화하는 데 사용됩니다. 디바이스 호스트는 이 메서드를 호출합니다.
InvokeAction 디바이스에서 메서드를 호출합니다.
로드 Load 메서드는 문서를 동기적으로 로드합니다. 이 메서드는 부하 작업이 완료될 때까지 호출자에게 제어를 반환하지 않습니다.
Loadasync LoadAsync 메서드는 문서를 비동기적으로 로드합니다. 이 메서드는 호출자에게 컨트롤을 즉시 반환하고 지정된 콜백을 사용하여 작업이 완료되면 호출자에게 알립니다.
LoadComplete LoadComplete 메서드는 UPnP 프레임워크가 디바이스 설명 로드를 완료했을 때 호출됩니다.
OnStateChanged OnStateChanged 메서드는 변경된 상태 변수의 DISPID 목록을 사용하여 디바이스 호스트에 이벤트를 보냅니다. 디바이스 호스트는 변경된 각 상태 변수에 대한 새 값을 얻으려면 서비스 개체를 쿼리해야 합니다.
OnStateChangedSafe OnStateChangedSafe 메서드는 변경된 DISPID 목록을 사용하여 디바이스 호스트에 이벤트를 보냅니다. 디바이스 호스트는 변경된 각 상태 변수에 대한 새 값을 얻으려면 서비스 개체를 쿼리해야 합니다.
QueryStateVariable QueryStateVariable 메서드는 지정된 서비스의 상태 변수 값을 반환합니다.
RegisterDevice RegisterDevice 메서드는 디바이스 호스트에 디바이스를 등록합니다. 디바이스 정보는 디바이스 호스트에 의해 저장됩니다. 그런 다음 디바이스 호스트는 디바이스 식별자를 반환하고 네트워크에서 디바이스를 게시하고 알릴 수 있습니다.

RegisterDeviceProvider RegisterDeviceProvider 메서드는 디바이스 호스트에 디바이스 공급자를 등록합니다. 디바이스 공급자가 네트워크에 게시되지 않습니다. 대신 디바이스를 동적으로 만들고 RegisterRunningDevice를 사용하여 등록합니다.
RegisterRunningDevice RegisterRunningDevice 메서드는 실행 중인 디바이스를 디바이스 호스트에 등록합니다.
ReregisterDevice ReregisterDevice 메서드는 디바이스 호스트에 디바이스를 다시 등록합니다. 디바이스 정보는 디바이스 호스트에 의해 저장됩니다. 그런 다음 디바이스 호스트는 디바이스 식별자를 반환하고 네트워크에서 디바이스를 게시하고 알릴 수 있습니다.
ReregisterRunningDevice ReregisterRunningDevice 메서드는 실행 중인 디바이스를 디바이스 호스트에 다시 등록합니다.
RootDevice RootDevice 메서드는 현재 로드된 문서의 디바이스 트리의 루트 디바이스를 반환합니다.
SearchComplete SearchComplete 메서드는 UPnP 프레임워크에 의해 호출되어 네트워크 디바이스에 대한 초기 검색이 완료되었음을 애플리케이션에 알립니다.
ServiceInstanceDied ServiceInstanceDied 메서드는 서비스가 더 이상 이벤트를 보내지 않을 때 호출됩니다.
SetAddressFamily SetAddressFamily 메서드는 이 플래그를 사용하여 찾은 디바이스를 필터링하는 Device Finder 개체의 주소 패밀리 플래그를 설정합니다.
SetServiceEnumProperty SetServiceEnumProperty 메서드는 IUPnPServices 개체에서 열거된 IUPnPService 개체에 대해 지연된 SCPD(서비스 제어 프로토콜 설명) 다운로드 및 이벤트 구독에 대한 옵트인을 나타내는 데 사용됩니다.
시작 Start 메서드는 디바이스 공급자를 시작합니다. 디바이스 호스트는 디바이스 공급자를 로드한 후 이 메서드를 호출합니다. 이 메서드는 디바이스 공급자에 필요한 초기화를 수행합니다.

StartAsyncFind StartAsyncFind 메서드는 비동기 검색 작업을 시작합니다.
StateVariableChanged StateVariableChanged 메서드는 상태 변수가 변경될 때 호출됩니다.
중지 Stop 메서드는 디바이스 공급자를 중지합니다.
Unadvise Unadvise 메서드는 이벤트 수신을 중지하기 위해 디바이스 호스트에서 호출됩니다. 디바이스 호스트는 IUPnPEventSource::Advise 메서드를 호출할 때와 동일한 포인터를 전달합니다.
UnregisterDevice UnregisterDevice 메서드는 디바이스 호스트에서 디바이스를 등록 취소합니다. 디바이스가 일시적으로 또는 영구적으로 등록 취소됩니다.
UnregisterDeviceProvider UnregisterDeviceProvider 메서드는 디바이스 호스트에서 디바이스 공급자의 등록을 영구적으로 취소하고 언로드합니다. IUPnPDeviceProvider::Stop 메서드가 호출됩니다.

인터페이스

IUPnPAddressFamilyControl IUPnPAddressFamilyControl 인터페이스는 Device Finder 개체의 주소 패밀리 플래그에 액세스합니다.
IUPnPAsyncResult IUPnPAsyncResult 인터페이스는 완료된 비동기 I/O 작업의 UPnP 제어 지점에 알리는 데 사용됩니다.
IUPnPDescriptionDocument IUPnPDescriptionDocument 인터페이스를 사용하면 애플리케이션에서 디바이스 설명을 로드할 수 있습니다.

IUPnPDescriptionDocumentCallback IUPnPDescriptionDocumentCallback 인터페이스를 사용하면 UPnP 프레임워크가 비동기 로드 작업의 결과를 애플리케이션에 전달할 수 있습니다.
IUPnPDevice IUPnPDevice 인터페이스를 사용하면 애플리케이션이 특정 디바이스에 대한 정보를 검색할 수 있습니다.
IUPnPDeviceControl IUPnPDeviceControl 인터페이스는 디바이스 및 해당 서비스 개체에 대한 관리의 중심 지점입니다.
IUPnPDeviceDocumentAccess IUPnPDeviceDocumentAccess 인터페이스를 사용하면 애플리케이션이 디바이스 설명 문서의 URL을 가져올 수 있습니다.
IUPnPDeviceDocumentAccessEx 특정 디바이스에 대한 전체 XML 디바이스 설명 문서를 가져오는 방법을 제공합니다.
IUPnPDeviceFinder IUPnPDeviceFinder 인터페이스를 사용하면 애플리케이션에서 디바이스를 찾을 수 있습니다.
IUPnPDeviceFinderAddCallbackWithInterface IUPnPDeviceFinderAddCallbackWithInterface 인터페이스를 사용하면 UPnP 프레임워크가 애플리케이션과 통신할 수 있습니다.
IUPnPDeviceFinderCallback IUPnPDeviceFinderCallback 인터페이스를 사용하면 UPnP 프레임워크가 비동기 검색 결과를 애플리케이션에 전달할 수 있습니다.
IUPnPDeviceProvider IUPnPDeviceProvider 인터페이스를 사용하면 디바이스 공급자가 처리를 시작하고 중지할 수 있습니다.
IUPnPDevices IUPnPDevices 인터페이스는 디바이스 컬렉션을 열거합니다.

IUPnPEventSink IUPnPEventSink 인터페이스를 사용하면 호스트된 서비스가 디바이스 호스트에 이벤트 알림을 보낼 수 있습니다.
IUPnPEventSource IUPnPEventSource 인터페이스를 사용하면 디바이스 호스트가 호스트된 서비스에 대한 이벤트 구독을 관리할 수 있습니다.
IUPnPHttpHeaderControl 호출자가 디바이스에 대한 HTTP 요청으로 전송되는 추가 HTTP 헤더를 지정할 수 있도록 합니다.
IUPnPRegistrar IUPnPRegistrar 인터페이스는 디바이스 호스트의 컨텍스트에서 실행되는 디바이스를 등록합니다.
IUPnPRemoteEndpointInfo IUPnPRemoteEndpointInfo 인터페이스를 사용하면 호스트된 디바이스가 요청자(즉, 제어점) 및 요청에 대한 정보를 가져올 수 있습니다.
IUPnPReregistrar IUPnPReregistrar 인터페이스를 사용하면 애플리케이션이 디바이스 호스트에 UPnP 기반 디바이스를 다시 등록할 수 있습니다.
IUPnPService IUPnPService 인터페이스를 사용하면 애플리케이션이 상태 변수를 쿼리하고 서비스 인스턴스에 대한 작업을 호출할 수 있습니다.
IUPnPServiceAsync 이 인터페이스를 사용하여 상태 변수를 비동기적으로 쿼리하고 서비스 인스턴스에서 작업을 호출합니다.
IUPnPServiceCallback IUPnPServiceCallback 인터페이스는 서비스 개체의 클라이언트에 이벤트 알림을 보내는 데 사용됩니다.
IUPnPServiceDocumentAccess 이 인터페이스를 사용하여 SCPD(서비스 제어 프로토콜 설명) 문서를 검색하여 UPnP 제어 지점 애플리케이션에 제공하여 서비스에서 지원하는 작업을 노출하고 상태 변수에 대한 정보를 제공합니다.

IUPnPServiceEnumProperty 이 인터페이스를 사용하여 IUPnPServices 개체에서 열거된 IUPnPService 개체에서 SCPD(서비스 제어 프로토콜 설명) 다운로드 및 이벤트 구독을 지연합니다.
IUPnPServices IUPnPServices 인터페이스는 서비스 컬렉션을 열거합니다.

USB 드라이버 참조

아티클2023. 08. 22.

USB 드라이버 참조 기술 개요입니다.

USB 드라이버 참조를 개발하려면 다음 헤더가 필요합니다.

- genericusbfnioctl.h
- usbuser.h
- winusb.h
- winusbio.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- USB 드라이버 참조

IOCTLs

IOCTL_GENERICUSBFN_ACTIVATE_USB_BUS 이 IOCTL(I/O 제어 코드)은 사용자 모드 서비스 또는 애플리케이션에서 전송되어 GenericUSBFn.sys USB(유니버설 직렬 버스)를 활성화하도록 알립니다. 활성화되면 버스는 버스 이벤트를 처리하고 트래픽을 처리할 준비가 됩니다.
IOCTL_GENERICUSBFN_BUS_EVENT_NOTIFICATION 이 IOCTL(I/O 제어 코드)은 USB(유니버설 직렬 버스) 이벤트에 등록하기 위해 사용자 모드 서비스 또는 애플리케이션에 의해 전송됩니다.
IOCTL_GENERICUSBFN_CONTROL_STATUS_HANDSHAKE_IN 이 IOCTL(I/O 제어 코드)은 사용자 모드 서비스 또는 애플리케이션에서 엔드포인트 0의 핸드셰이크를 IN 방향으로 상태 길이가 0인 컨트롤을 요청하기 위해 전송됩니다.
IOCTL_GENERICUSBFN_CONTROL_STATUS_HANDSHAKE_OUT 이 IOCTL(I/O 제어 코드)은 사용자 모드 서비스 또는 애플리케이션에서 OUT 방향으로 엔드포인트 0에서 핸드셰이크를 상태 길이가 0인 컨트롤을 완료하기 위해 전송됩니다.
IOCTL_GENERICUSBFN_DEACTIVATE_USB_BUS 이 IOCTL 코드는 nevtot가 지원됩니다.

IOCTL_GENERICUSBFN_GET_CLASS_INFO 이 IOCTL(I/O 제어 코드)은 레지스트리에 구성된 디바이스의 사용 가능한 파이프에 대한 정보를 검색하기 위해 사용자 모드 서비스 또는 애플리케이션에서 전송됩니다.
IOCTL_GENERICUSBFN_GET_CLASS_INFO_EX 이 IOCTL(I/O 제어 코드)은 레지스트리에 구성된 디바이스의 사용 가능한 파이프에 대한 정보를 검색하기 위해 사용자 모드 서비스 또는 애플리케이션에서 전송됩니다.
IOCTL_GENERICUSBFN_GET_INTERFACE_DESCRIPTOR_SET 이 IOCTL(I/O 제어 코드)은 사용자 모드 서비스 또는 애플리케이션에서 전송되어 디바이스의 함수에 대해 설정된 전체 인터페이스 설명자를 가져옵니다. 이 IOCTL 요청은 전체 디바이스에 대해 설정된 인터페이스 설명자를 검색하지 않습니다. 디바이스의 함수에 대해 설정된 USB(유니버설 직렬 버스) 인터페이스 설명자입니다.
IOCTL_GENERICUSBFN_GET_PIPE_STATE 이 IOCTL(I/O 제어 코드)은 사용자 모드 서비스 또는 애플리케이션에 의해 전송되어 지정된 USB(유니버설 직렬 버스) 파이프의 상태를 가져옵니다.
IOCTL_GENERICUSBFN_REGISTER_USB_STRING 이 IOCTL(I/O 제어 코드)은 문자열 설명자를 등록하기 위해 사용자 모드 서비스 또는 애플리케이션에 의해 전송됩니다. USB(유니버설 직렬 버스) 문자열 설명자입니다.
IOCTL_GENERICUSBFN_SET_PIPE_STATE 이 IOCTL(I/O 제어 코드)은 사용자 모드 서비스 또는 애플리케이션에 의해 전송되어 지정된 USB(유니버설 직렬 버스) 파이프의 상태를 설정합니다.
IOCTL_GENERICUSBFN_TRANSFER_IN 이 IOCTL(I/O 제어 코드)은 입력 버퍼의 지정된 파이프 ID에 해당하는 엔드포인트에서 IN 방향 전송을 실행하기 위해 사용자 모드 서비스 또는 애플리케이션에 의해 전송됩니다. (IOCTL_GENERICUSBFN_TRANSFER_IN)
IOCTL_GENERICUSBFN_TRANSFER_IN_APPEND_ZERO_PKT 이 IOCTL(I/O 제어 코드)은 입력 버퍼의 지정된 파이프 ID에 해당하는 엔드포인트에서 IN 방향 전송을 실행하기 위해 사용자 모드 서비스 또는 애플리케이션에 의해 전송됩니다. (IOCTL_GENERICUSBFN_TRANSFER_IN_APPEND_ZERO_PKT)
IOCTL_GENERICUSBFN_TRANSFER_OUT 이 IOCTL(I/O 제어 코드)은 입력 버퍼의 지정된 파이프 ID에 해당하는 엔드포인트에서 OUT 방향 전송을 실행하기 위해 사용자 모드 서비스 또는 애플리케이션에 의해 전송됩니다.

IOCTL_GET_HCD_DRIVERKEY_NAME IOCTL_GET_HCD_DRIVERKEY_NAME I/O 제어 요청은 USB 호스트 컨트롤러 드라이버에 대한 레지스트리의 드라이버 키 이름을 검색합니다.
IOCTL_USB_DIAGNOSTIC_MODE_OFF IOCTL_USB_DIAGNOSTIC_MODE_OFF I/O 컨트롤은 더 이상 사용되지 않습니다. 사용하지 마십시오.
IOCTL_USB_DIAGNOSTIC_MODE_ON IOCTL_USB_DIAGNOSTIC_MODE_ON I/O 컨트롤은 더 이상 사용되지 않습니다. 사용하지 마십시오.
IOCTL_USB_GET_ROOT_HUB_NAME IOCTL_USB_GET_ROOT_HUB_NAME I/O 컨트롤 요청은 루트 허브의 기호 링크 이름을 검색하기 위해 USB_ROOT_HUB_NAME 구조체와 함께 사용됩니다. IOCTL_USB_GET_ROOT_HUB_NAME 사용자 모드 I/O 제어 요청입니다.
IOCTL_USB_USER_REQUEST IOCTL_USB_USER_REQUEST I/O 제어 요청은 사용자 모드 애플리케이션과 커널 모드 드라이버 모두에서 사용할 수 있습니다.

열거형

USB_USER_ERROR_CODE USB_USER_ERROR_CODE 열거형은 USB 사용자 모드가 실패할 때 보고하는 오류 코드를 나열합니다.
WDMUSB_POWER_STATE WDMUSB_POWER_STATE 열거형은 호스트 컨트롤러 또는 루트 허브의 전원 상태를 나타냅니다.

Functions

WinUsb_AbortPipe

WinUsb_AbortPipe 함수는 파이프에 대해 보류 중인 모든 전송을 중단합니다. 동기 작업입니다.
WinUsb_ControlTransfer WinUsb_ControlTransfer 함수는 기본 제어 엔드포인트를 통해 제어 데이터를 전송합니다.
WinUsb_FlushPipe WinUsb_FlushPipe 함수는 파이프에 캐시된 모든 데이터를 삭제합니다. 동기 작업입니다.
WinUsb_Free WinUsb_Free 함수는 할당된 모든 리소스를 WinUsb_Initialize 해제합니다. 동기 작업입니다.
WinUsb_GetAdjustedFrameNumber WinUsb_GetAdjustedFrameNumber 함수는 프레임 번호 값 및 타임스탬프를 기반으로 현재 USB 프레임 번호를 계산합니다.
WinUsb_GetAssociatedInterface WinUsb_GetAssociatedInterface 함수는 연결된 인터페이스에 대한 핸들을 검색합니다. 동기 작업입니다.
WinUsb_GetCurrentAlternateSetting WinUsb_GetCurrentAlternateSetting 함수는 인터페이스에 대한 현재 대체 인터페이스 설정을 가져옵니다. 동기 작업입니다.
WinUsb_GetCurrentFrameNumber WinUsb_GetCurrentFrameNumber 함수는 버스의 현재 프레임 번호를 가져옵니다.
WinUsb_GetCurrentFrameNumberAndQpc WinUsb_GetCurrentFrameNumberAndQpc 함수는 프레임 및 마이크로프레임과 동기화된 QPC(시스템 쿼리 성능 카운터) 값을 검색합니다.
WinUsb_GetDescriptor WinUsb_GetDescriptor 함수는 요청된 설명자를 반환합니다. 동기 작업입니다.
WinUsb_GetOverlappedResult WinUsb_GetOverlappedResult 함수는 지정된 파일에서 겹치는 작업의 결과를 검색합니다.
WinUsb_GetPipePolicy

WinUsb_GetPipePolicy 함수는 디바이스의 엔드포인트와 연결된 특정 파이프에 대한 정책을 검색합니다. 동기 작업입니다.
WinUsb_GetPowerPolicy WinUsb_GetPowerPolicy 함수는 디바이스의 전원 정책을 검색합니다. 동기 작업입니다.
WinUsb_Initialize WinUsb_Initialize 함수는 파일 핸들로 지정된 디바이스에 대한 WinUSB 핸들을 만듭니다.
WinUsb_QueryDeviceInformation WinUsb_QueryDeviceInformation 함수는 WinUSB 인터페이스 핸들과 연결된 물리적 디바이스에 대한 정보를 가져옵니다.
WinUsb_QueryInterfaceSettings WinUsb_QueryInterfaceSettings 함수는 특정 인터페이스 핸들에 대해 지정된 대체 인터페이스 설정에 대한 인터페이스 설명자를 검색합니다.
WinUsb_QueryPipe WinUsb_QueryPipe 함수는 인터페이스에 대해 지정된 엔드포인트 및 연결된 파이프에 대한 정보를 검색합니다.
WinUsb_QueryPipeEx WinUsb_QueryPipeEx 함수는 인터페이스에 대해 지정된 엔드포인트 및 연결된 파이프에 대한 확장 정보를 검색합니다.
WinUsb_ReadIsochPipe WinUsb_ReadIsochPipe 함수는 등시 OUT 엔드포인트에서 데이터를 읽습니다.
WinUsb_ReadIsochPipeAsap WinUsb_ReadIsochPipeAsap 함수는 등시 OUT 엔드포인트에서 데이터를 읽는 요청을 제출합니다.
WinUsb_ReadPipe WinUsb_ReadPipe 함수는 지정된 파이프에서 데이터를 읽습니다.
WinUsb_RegisterIsochBuffer WinUsb_RegisterIsochBuffer 함수는 등시 전송에 사용할 버퍼를 등록합니다.
WinUsb_ResetPipe

WinUsb_ResetPipe 함수는 데이터 토글을 재설정하고 파이프의 중단 조건을 지웁니다.
WinUsb_SetCurrentAlternateSetting WinUsb_SetCurrentAlternateSetting 함수는 인터페이스의 대체 설정을 설정합니다.
WinUsb_SetPipePolicy WinUsb_SetPipePolicy 함수는 디바이스의 엔드포인트와 연결된 특정 파이프에 대한 정책을 설정합니다. 동기 작업입니다.
WinUsb_SetPowerPolicy WinUsb_SetPowerPolicy 함수는 디바이스에 대한 전원 정책을 설정합니다.
WinUsb_StartTrackingForTimeSync WinUsb_StartTrackingForTimeSync 함수는 USB 드라이버 스택에서 USB 버스 프레임 및 마이크로프레임에 대한 연결된 시스템 QPC 시간을 가져오는 시간 동기화 기능을 시작합니다.
WinUsb_StopTrackingForTimeSync WinUsb_StopTrackingForTimeSync 함수는 USB 버스 프레임 및 마이크로프레임에 대한 연결된 시스템 QPC 시간을 가져오는 USB 드라이버 스택의 시간 동기화 기능을 맨 위에 놓습니다.
WinUsb_UnregisterIsochBuffer WinUsb_UnregisterIsochBuffer 함수는 등시 전송에 할당된 WinUsb_RegisterIsochBuffer 모든 리소스를 해제합니다. 동기 작업입니다.
WinUsb_WriteIsochPipe WinUsb_WriteIsochPipe 함수는 지정된 프레임 번호에서 시작하여 호출자가 제공한 버퍼의 내용을 등시 OUT 엔드포인트에 씁니다.
WinUsb_WriteIsochPipeAsap WinUsb_WriteIsochPipeAsap 버퍼의 콘텐츠를 등시 OUT 엔드포인트에 쓰기 위한 요청을 제출합니다.
WinUsb_WritePipe WinUsb_WritePipe 함수는 파이프에 데이터를 씁니다.

구조체

USB_BANDWIDTH_INFO USB_BANDWIDTH_INFO 구조는 할당된 대역폭에 대한 정보를 검색하기 위해 IOCTL_USB_USER_REQUEST I/O 컨트롤 요청과 함께 사용됩니다.
USB_BUS_STATISTICS_0 USB_BUS_STATISTICS_0 구조는 버스 통계를 검색하기 위해 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용됩니다.
USB_CLOSE_RAW_DEVICE_PARAMETERS 이 구조는 지원되지 않습니다. USB_CLOSE_RAW_DEVICE_PARAMETERS 구조는 버스의 디바이스에 대한 원시 액세스를 닫기 위해 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용됩니다.
USB_CONTROLLER_INFO_0 USB_CONTROLLER_INFO_0 구조는 USB 호스트 컨트롤러에 대한 정보를 검색하기 위해 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용됩니다.
USB_DRIVER_VERSION_PARAMETERS USB_DRIVER_VERSION_PARAMETERS 구조는 IOCTL_USB_USER_REQUEST I/O 컨트롤 요청과 함께 버전 정보를 검색하는 데 사용됩니다.
USB_PASS_THRU_PARAMETERS USB_PASS_THRU_PARAMETERS 구조는 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용하여 공급업체별 명령을 호스트 컨트롤러 미니포트 드라이버에 전달합니다.
USB_POWER_INFO USB_POWER_INFO 구조는 호스트 컨트롤러 전원 정책이 표시된 시스템 전원 상태에 대해 지정하는 디바이스 전원 상태를 검색하기 위해 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용됩니다.
USB_UNICODE_NAME USB_UNICODE_NAME 구조에는 기호 링크 이름을 지정하는 유니코드 문자열이 포함되어 있습니다.
USBUSER_BANDWIDTH_INFO_REQUEST USBUSER_BANDWIDTH_INFO_REQUEST 구조는 할당된 대역폭에 대한 정보를 검색하기 위해 IOCTL_USB_USER_REQUEST I/O 컨트롤 요청과 함께 사용됩니다.
USBUSER_BUS_STATISTICS_0_REQUEST

USBUSER_BUS_STATISTICS_0_REQUEST 구조는 버스 통계를 검색하기 위해 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용됩니다.
USBUSER_CONTROLLER_UNICODE_NAME USBUSER_CONTROLLER_UNICODE_NAME 구조는 USB 호스트 컨트롤러 드라이버 키 이름을 검색하기 위해 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용됩니다.
USBUSER_GET_DRIVER_VERSION USBUSER_GET_DRIVER_VERSION 구조는 IOCTL_USB_USER_REQUEST I/O 컨트롤 요청과 함께 드라이버 및 인터페이스 버전 정보를 읽는 데 사용됩니다.
USBUSER_PASS_THRU_REQUEST USBUSER_PASS_THRU_REQUEST 구조는 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용하여 공급업체별 명령을 호스트 컨트롤러 미니포트 드라이버에 보냅니다.
USBUSER_POWER_INFO_REQUEST USBUSER_POWER_INFO_REQUEST 구조는 호스트 컨트롤러 및 루트 허브의 전원 상태와 특정 시스템 상태의 관계에 관한 전원 정책 정보를 검색하기 위해 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용됩니다.
USBUSER_REQUEST_HEADER USBUSER_REQUEST_HEADER 구조는 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용하여 USB 호스트 컨트롤러 드라이버에 사용자 모드 요청을 보냅니다.
WINUSB_PIPE_INFORMATION WINUSB_PIPE_INFORMATION 구조에는 WinUsb_QueryPipe 루틴에서 검색하는 파이프 정보가 포함됩니다.
WINUSB_PIPE_INFORMATION_EX WINUSB_PIPE_INFORMATION_EX 구조에는 WinUsb_QueryPipeEx 루틴에서 검색하는 파이프 정보가 포함됩니다.
WINUSB_SETUP_PACKET WINUSB_SETUP_PACKET 구조는 USB 설정 패킷을 설명합니다.

피드백

이 페이지가 도움이 되었나요?

Yes No

Microsoft Q&A에서 도움말 보기

사용자 액세스 로깅

아티클2023. 03. 09.

사용자 액세스 로깅 기술 개요입니다.

사용자 액세스 로깅을 개발하려면 다음 헤더가 필요합니다.

- ual.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 사용자 액세스 로깅

함수

UalInstrument UAL_DATA_BLOB 구조의 정보를 사용하여 지정된 데이터를 UAL(사용자 액세스 로깅) 프레임워크에 기록합니다.
UalRegisterProduct UAL(사용자 액세스 로깅)에 제품을 등록합니다.
UalStart UAL(사용자 액세스 로깅) 세션을 시작합니다.
UalStop UAL(사용자 액세스 로깅) 세션을 중지합니다.

구조체

UAL_DATA_BLOB UAL(사용자 액세스 로깅) 세션에 대한 정보를 지정합니다.

가상 하드 디스크

아티클2023. 03. 09.

가상 하드 디스크 기술 개요입니다.

가상 하드 디스크를 개발하려면 다음 헤더가 필요합니다.

- virtdisk.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 가상 스토리지

가상 스토리지

아티클2023. 03. 09.

Virtual Storage 기술에 대한 개요입니다.

Virtual Storage 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 가상 스토리지

열거형

APPLY_SNAPSHOT_VHDSET_FLAG ApplySnapshotVhdSet 함수의 동작에 영향을 주는 플래그를 포함합니다.
APPLY_SNAPSHOT_VHDSET_VERSION ApplySnapshotVhdSet 함수에 사용할 수 있는 매개 변수 버전을 열거합니다.
ATTACH_VIRTUAL_DISK_FLAG 가상 디스크 연결 요청 플래그를 포함합니다.
ATTACH_VIRTUAL_DISK_VERSION VHD 함수 호출에 사용할 VHD(가상 하드 디스크) ATTACH_VIRTUAL_DISK_PARAMETERS 구조체의 버전을 포함합니다.
COMPACT_VIRTUAL_DISK_FLAG 가상 디스크 압축 요청 플래그를 포함합니다.
COMPACT_VIRTUAL_DISK_VERSION VHD 함수 호출에 사용할 VHD(가상 하드 디스크) COMPACT_VIRTUAL_DISK_PARAMETERS 구조체의 버전을 포함합니다.
CREATE_VIRTUAL_DISK_FLAG VHD(가상 하드 디스크) 만들기 플래그를 포함합니다.

CREATE_VIRTUAL_DISK_VERSION 가상 디스크 함수 호출에 사용할 가상 디스크 CREATE_VIRTUAL_DISK_PARAMETERS 구조체의 버전을 포함합니다.
DELETE_SNAPSHOT_VHDSET_FLAG DeleteSnapshotVhdSet 함수의 동작에 영향을 주는 플래그를 포함합니다.
DELETE_SNAPSHOT_VHDSET_VERSION 가상 디스크 함수 호출에 사용할 DELETE_SNAPHSOT_VHDSET_PARAMETERS 구조의 버전을 포함합니다.
DEPENDENT_DISK_FLAG VHD(가상 하드 디스크) 종속성 정보 플래그를 포함합니다.
DETACH_VIRTUAL_DISK_FLAG 가상 디스크 분리 요청 플래그를 포함합니다.
EXPAND_VIRTUAL_DISK_FLAG VHD(가상 하드 디스크) 확장 요청 플래그를 포함합니다.
EXPAND_VIRTUAL_DISK_VERSION 가상 디스크 함수 호출에 사용할 가상 디스크 EXPAND_VIRTUAL_DISK_PARAMETERS 구조체의 버전을 포함합니다.
GET_STORAGE_DEPENDENCY_FLAG VHD(가상 하드 디스크) 스토리지 종속성 요청 플래그를 포함합니다.
GET_VIRTUAL_DISK_INFO_VERSION 검색할 수 있는 VHD(가상 하드 디스크) 정보의 종류를 포함합니다.
MERGE_VIRTUAL_DISK_FLAG VHD(가상 하드 디스크) 병합 요청 플래그를 포함합니다.
MERGE_VIRTUAL_DISK_VERSION VHD 함수 호출에 사용할 VHD(가상 하드 디스크) MERGE_VIRTUAL_DISK_PARAMETERS 구조체의 버전을 포함합니다.

MIRROR_VIRTUAL_DISK_FLAG VHD(가상 하드 디스크) 미러 요청 플래그를 포함합니다.
MIRROR_VIRTUAL_DISK_VERSION MirrorVirtualDisk 함수에서 사용하는 가상 디스크 MIRROR_VIRTUAL_DISK_PARAMETERS 구조체의 버전을 포함합니다.
MODIFY_VHDSET_FLAG ModifyVhdSet 함수의 동작에 영향을 주는 플래그를 포함합니다.
MODIFY_VHDSET_VERSION 가상 디스크 함수 호출에 사용할 MODIFY_VHDSET_PARAMETERS 구조의 버전을 포함합니다.
OPEN_VIRTUAL_DISK_FLAG VHD(가상 하드 디스크) 또는 CD 또는 ISO(DVD 이미지 파일) 열기 요청 플래그를 포함합니다.
OPEN_VIRTUAL_DISK_VERSION 가상 디스크 함수 호출에 사용할 가상 디스크 OPEN_VIRTUAL_DISK_PARAMETERS 구조체의 버전을 포함합니다.
RAW_SCSI_VIRTUAL_DISK_FLAG RawSCSIVirtualDisk 함수의 동작에 영향을 주는 플래그를 포함합니다.
RAW_SCSI_VIRTUAL_DISK_VERSION 가상 디스크 함수 호출에 사용할 RAW_SCSI_VIRTUAL_DISK_PARAMETERS 구조의 버전을 포함합니다.
RESIZE_VIRTUAL_DISK_FLAG ResizeVirtualDisk 함수에 사용할 수 있는 플래그를 열거합니다.
RESIZE_VIRTUAL_DISK_VERSION ResizeVirtualDisk 함수에 사용할 수 있는 매개 변수 버전을 열거합니다.
SET_VIRTUAL_DISK_INFO_VERSION VHD 함수 호출에 사용할 가상 디스크 SET_VIRTUAL_DISK_INFO 구조체의 버전을 포함합니다.

STORAGE_DEPENDENCY_INFO_VERSION VHD 함수 호출에 사용할 VHD(가상 하드 디스크) STORAGE_DEPENDENCY_INFO 구조체의 버전을 포함합니다.
TAKE_SNAPSHOT_VHDSET_FLAG TakeSnapshotVhdSet 함수의 동작에 영향을 주는 플래그를 포함합니다.
TAKE_SNAPSHOT_VHDSET_VERSION TakeSnapshotVhdSet 함수에 사용할 수 있는 매개 변수 버전을 열거합니다.

함수

AddVirtualDiskParent OPEN_VIRTUAL_DISK_FLAG_CUSTOM_DIFF_CHAIN 플래그를 사용하여 열린 가상 디스크에 부모를 연결합니다.
ApplySnapshotVhdSet VHD 집합 파일에 대해 현재 가상 디스크의 스냅샷을 적용합니다.
AttachVirtualDisk 첨부 파일을 수행하기 위해 적절한 VHD 공급자를 찾아 VHD(가상 하드 디스크) 또는 CD 또는 DVD 이미지 파일(ISO)을 연결합니다.
BreakMirrorVirtualDisk 이전에 시작된 미러 작업을 중단하고 미러를 활성 가상 디스크로 설정합니다.
CompactVirtualDisk VHD(가상 하드 디스크) 백업 저장소 파일의 크기를 줄입니다.
CreateVirtualDisk 기본 매개 변수를 사용하거나 기존 가상 디스크 또는 실제 디스크를 사용하여 VHD(가상 하드 디스크) 이미지 파일을 만듭니다.
DeleteSnapshotVhdSet VHD 집합 파일에서 스냅샷을 삭제합니다.

DeleteVirtualDiskMetadata 가상 디스크에서 메타데이터를 삭제합니다.
DetachVirtualDisk 작업을 수행하기 위해 적절한 가상 디스크 공급자를 찾아 VHD(가상 하드 디스크) 또는 CD 또는 DVD 이미지 파일(ISO)을 분리합니다.
EnumerateVirtualDiskMetadata 가상 디스크와 연결된 메타데이터를 열거합니다.
ExpandVirtualDisk 고정되거나 동적으로 확장 가능한 VHD(가상 하드 디스크)의 크기를 늘입니다.
GetStorageDependencyInformation VHD(가상 하드 디스크) 또는 CD 또는 DVD 이미지 파일(ISO) 또는 해당 디스크와 해당 부모 디스크 또는 볼륨에 포함된 볼륨 간의 관계를 반환합니다.
GetVirtualDiskInformation VHD에 대한 정보를 검색합니다.
GetVirtualDiskMetadata 가상 디스크에서 지정된 메타데이터를 검색합니다.
GetVirtualDiskOperationProgress 비동기 VHD(가상 하드 디스크) 작업의 진행률을 확인합니다.
GetVirtualDiskPhysicalPath VHD(가상 하드 디스크) 또는 CD 또는 DVD 이미지 파일(ISO)이 포함된 물리적 디바이스 개체의 경로를 검색합니다.
MergeVirtualDisk 차이점 보관용 체인의 자식 VHD(가상 하드 디스크)를 체인에 있는 하나 이상의 부모 가상 디스크와 병합합니다.
MirrorVirtualDisk 가상 디스크에 대한 미러 작업을 시작합니다.

ModifyVhdSet 가상 디스크 파일의 내부 콘텐츠를 수정합니다. 활성 리프를 설정하거나 스냅샷 항목을 수정하는 데 사용할 수 있습니다.
OpenVirtualDisk 사용할 VHD(가상 하드 디스크) 또는 CD 또는 DVD 이미지 파일(ISO)을 엽니다.
QueryChangesVirtualDisk RCT(복원력 있는 변경 내용 추적)에 의해 추적되는 VHD(가상 하드 디스크)의 지정된 영역에 대한 변경 내용에 대한 정보를 검색합니다.
RawSCSIVirtualDisk 포함된 SCSI 요청을 가상 하드 디스크에 직접 발급합니다.
ResizeVirtualDisk 가상 디스크의 크기를 조정합니다.
SetVirtualDiskInformation VHD(가상 하드 디스크)에 대한 정보를 설정합니다.
SetVirtualDiskMetadata 가상 디스크에 대한 메타데이터 항목을 설정합니다.
TakeSnapshotVhdSet VHD 집합 파일에 대한 현재 가상 디스크의 스냅샷을 만듭니다.

구조체

APPLY_SNAPSHOT_VHDSET_PARAMETERS 적용할 새 스냅샷에 대한 정보를 나타내는 스냅샷 매개 변수를 포함합니다.
ATTACH_VIRTUAL_DISK_PARAMETERS VHD(가상 하드 디스크) 연결 요청 매개 변수를 포함합니다.

COMPACT_VIRTUAL_DISK_PARAMETERS VHD(가상 하드 디스크) 압축 매개 변수를 포함합니다.
CREATE_VIRTUAL_DISK_PARAMETERS 새로 만든 가상 디스크에 대한 제어 및 정보를 제공하는 VHD(가상 하드 디스크) 만들기 매개 변수를 포함합니다.
DELETE_SNAPSHOT_VHDSET_PARAMETERS VHD 집합에서 삭제할 스냅샷을 지정하는 스냅샷 삭제 매개 변수를 포함합니다.
EXPAND_VIRTUAL_DISK_PARAMETERS 가상 디스크 확장 요청 매개 변수를 포함합니다.
GET_VIRTUAL_DISK_INFO VHD(가상 하드 디스크) 정보를 포함합니다.
MERGE_VIRTUAL_DISK_PARAMETERS VHD(가상 하드 디스크) 병합 요청 매개 변수를 포함합니다.
MIRROR_VIRTUAL_DISK_PARAMETERS VHD(가상 하드 디스크) 미러 요청 매개 변수를 포함합니다.
MODIFY_VHDSET_PARAMETERS VHD 집합을 변경하는 방법을 나타내는 VHD 집합 수정 매개 변수를 포함합니다.
OPEN_VIRTUAL_DISK_PARAMETERS 가상 디스크 열기 요청 매개 변수를 포함합니다.
QUERY_CHANGES_VIRTUAL_DISK_RANGE RCT(복원력 있는 변경 내용 추적)에 의해 추적된 대로 변경된 VHD(가상 하드 디스크)의 영역을 식별합니다.
RAW_SCSI_VIRTUAL_DISK_PARAMETERS 원시 SCSI 가상 디스크 요청 매개 변수를 포함합니다.
RAW_SCSI_VIRTUAL_DISK_RESPONSE 원시 SCSI 가상 디스크 응답 매개 변수를 포함합니다.

RESIZE_VIRTUAL_DISK_PARAMETERS ResizeVirtualDisk 함수에 대한 매개 변수를 포함합니다.
SET_VIRTUAL_DISK_INFO SetVirtualDiskInformation 함수를 호출하여 VHD 속성을 설정할 때 사용할 VHD(가상 하드 디스크) 정보를 포함합니다.
STORAGE_DEPENDENCY_INFO VHD(가상 하드 디스크) 스토리지 종속성 정보를 포함합니다.
STORAGE_DEPENDENCY_INFO_TYPE_1 유형 1에 대한 VHD(가상 하드 디스크) 스토리지 종속성 정보를 포함합니다.
STORAGE_DEPENDENCY_INFO_TYPE_2 형식 2에 대한 VHD 또는 ISO 스토리지 종속성 정보를 포함합니다.
TAKE_SNAPSHOT_VHDSET_PARAMETERS 만들 새 스냅샷에 대한 정보를 나타내는 스냅샷 매개 변수를 포함합니다.
VIRTUAL_DISK_PROGRESS GetVirtualDiskOperationProgress 함수에서 사용하는 현재 VHD(가상 하드 디스크) 작업에 대한 진행률 및 결과 데이터를 포함합니다.
VIRTUAL_STORAGE_TYPE 가상 스토리지 디바이스의 유형 및 공급자(공급업체)를 포함합니다.

w_graph_fx

W_Graph_Fx 기술 개요입니다.

W_Graph_Fx 기술은 헤더와 연결되지 않습니다.

Last updated on 2025. 10. 01.

w_ui_comp

W_Ui_Comp 기술 개요입니다.

W_Ui_Comp 기술은 헤더와 연결되지 않습니다.

Last updated on 2025. 10. 01.

WebAuthn

아티클2023. 03. 16.

WebAuthn 및 CTAP 사양의 일부로 Windows Hello 및 외부 보안 키와 통신하기 위한 API를 Win32 앱에 제공하는 WebAuthn 기술의 개요입니다.

WebAuthn API를 사용하여 개발하려면 다음 헤더가 필요합니다.

- webauthn.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- WebAuthn
- CTAP

함수

WebAuthNAuthenticatorGetAssertion 사용자가 특정 트랜잭션에 동의한 인증자의 어설션을 나타내는 어설션 서명을 생성합니다.
WebAuthNAuthenticatorMakeCredential 관리 인증자에 바인딩된 공개 키 자격 증명 소스를 만들고 자격 증명 프라이빗 키와 연결된 자격 증명 공개 키를 반환합니다.
WebAuthNCancelCurrentOperation 인증자 세션에서 현재 진행 중인 작업을 종료합니다.
WebAuthNDeletePlatformCredential 인증자에 저장된 자격 증명 소스를 제거합니다.
WebAuthNFreeAssertion 이전에 할당된 WebAuthN 어설션을 해제합니다.
WebAuthNFreeCredentialAttestation 이전에 할당된 자격 증명 증명을 해제합니다.

WebAuthNFreePlatformCredentialList 플랫폼 자격 증명 목록에 대한 할당을 해제합니다.
WebAuthNGetApiVersionNumber API 버전 번호를 가져옵니다.
WebAuthNGetCancellationId 취소된 작업의 취소 ID를 가져옵니다.
WebAuthNGetErrorName 오류 코드의 오류 이름을 가져옵니다.
WebAuthNGetPlatformCredentialList 저장된 자격 증명 목록을 가져옵니다.
WebAuthNGetW3CExceptionDOMError 마지막으로 실패한 작업에 대한 W3C DOM 오류 코드를 가져옵니다.
WebAuthNIsUserVerifyingPlatformAuthenticatorAvailable 플랫폼 인증자 서비스를 사용할 수 있는지 여부를 확인합니다.

구조체

WEBAUTHN_ASSERTION 어설션을 확인하는 데 필요한 데이터를 포함하는 구조체입니다.
WEBAUTHN_AUTHENTICATOR_GET_ASSERTION_OPTIONS 어설션을 가져오는 옵션이 포함된 구조체입니다.
WEBAUTHN_AUTHENTICATOR_MAKE_CREDENTIAL_OPTIONS WebAuthNAuthenticatorMakeCredential 작업에 대한 옵션입니다.
WEBAUTHN_CLIENT_DATA 인증자로 전송되는 클라이언트 데이터를 포함하는 구조체입니다.

WEBAUTHN_COMMON_ATTESTATION 증명에 대한 공통 데이터를 포함하는 구조체입니다.
WEBAUTHN_COSE_CREDENTIAL_PARAMETER COSE 자격 증명 매개 변수 정보를 포함하는 구조체입니다.
WEBAUTHN_COSE_CREDENTIAL_PARAMETERS 자격 증명에 대한 COSE 매개 변수를 포함하는 구조체입니다.
WEBAUTHN_CRED_BLOB_EXTENSION 자격 증명에 대한 Blob 확장 데이터입니다.
WEBAUTHN_CRED_PROTECT_EXTENSION_IN 자격 증명 보호 확장 정보를 포함합니다.
WEBAUTHN_CRED_WITH_HMAC_SECRET_SALT SALT 값이 있는 자격 증명을 포함하는 구조체입니다.
WEBAUTHN_CREDENTIAL 자격 증명에 대한 정보를 포함합니다.
WEBAUTHN_CREDENTIAL_ATTESTATION 자격 증명에 대한 증명 데이터를 포함합니다.
WEBAUTHN_CREDENTIAL_DETAILS 자격 증명에 대한 데이터를 포함합니다.
WEBAUTHN_CREDENTIAL_DETAILS_LIST 자격 증명 목록입니다.
WEBAUTHN_CREDENTIAL_EX 추가 정보가 포함된 자격 증명에 대한 데이터입니다.
WEBAUTHN_CREDENTIAL_LIST 사용자가 인증자에 등록한 자격 증명 목록입니다.

WEBAUTHN_CREDENTIALS 사용자가 인증자에 등록한 자격 증명의 배열입니다.
WEBAUTHN_EXTENSION 확장에 대한 정보를 포함합니다.
WEBAUTHN_EXTENSIONS 확장 목록을 포함합니다.
WEBAUTHN_GET_CREDENTIALS_OPTIONS WebAuthNGetPlatformCredentialsList 함수에 대한 옵션을 포함합니다.
WEBAUTHN_HMAC_SECRET_SALT Hmac-Secret에 대한 SALT 값을 포함합니다.
WEBAUTHN_HMAC_SECRET_SALT_VALUES HMAC 비밀에 대한 SALT 값을 포함합니다.
WEBAUTHN_RP_ENTITY_INFORMATION 신뢰 당사자에 대한 정보입니다.
WEBAUTHN_USER_ENTITY_INFORMATION 사용자 엔터티에 대한 정보입니다.
WEBAUTHN_X5C X.509 인증서 체인입니다.

Webdav

아티클2023. 08. 22.

WebDAV 기술 개요입니다.

WebDAV를 개발하려면 다음 헤더가 필요합니다.

- davclnt.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Webdav

열거형

AUTHNEXTSTEP DavAuthCallback 콜백 함수를 성공적으로 호출한 후 WebDAV 클라이언트가 수행해야 하는 다음 작업을 지정합니다.

Functions

DavAddConnection WebDAV 서버 또는 WebDAV 서버의 원격 파일 또는 디렉터리에 대한 보안 연결을 만듭니다.
DavCancelConnectionsToServer WebDAV 서버 또는 WebDAV 서버의 원격 파일 또는 디렉터리에 대한 모든 연결을 닫습니다.
DavDeleteConnection DavAddConnection 함수를 사용하여 만든 연결을 닫습니다.
DavFlushFile 원격 파일의 로컬 버전에서 WebDAV 서버로 데이터를 플러시합니다.
DavGetExtendedError WebDAV 서버가 이전에 실패한 I/O 작업에 대해 반환한 확장 오류 코드 정보를 검색합니다.

DavGetHTTPFromUNCPath 지정된 UNC 경로를 동등한 HTTP 경로로 변환합니다.
DavGetTheLockOwnerOfTheFile WebDAV 서버에 잠긴 파일의 파일 잠금 소유자를 반환합니다.
DavGetUNCFromHTTPPath 지정된 HTTP 경로를 동등한 UNC 경로로 변환합니다.
DavInvalidateCache WebDAV 서버의 원격 파일에 대한 로컬 캐시의 콘텐츠를 무효화합니다.
DavRegisterAuthCallback WebDAV 클라이언트가 사용자에게 자격 증명을 묻는 메시지를 표시하는 데 사용할 수 있는 애플리케이션 정의 콜백 함수를 등록합니다.
DavUnregisterAuthCallback WebDAV 클라이언트가 사용자에게 자격 증명을 묻는 메시지를 표시하는 데 사용하는 등록된 콜백 함수의 등록을 취소합니다.
PFNDAVAUTHCALLBACK WebDAV 클라이언트는 애플리케이션 정의 DavAuthCallback 콜백 함수를 호출하여 사용자에게 자격 증명을 묻는 메시지를 표시합니다.
PFNDAVAUTHCALLBACK_FREECRED WebDAV 클라이언트는 애플리케이션 정의 DavFreeCredCallback 콜백 함수를 호출하여 DavAuthCallback 콜백 함수에서 검색한 자격 증명 정보를 해제합니다.

구조체

DAV_CALLBACK_AUTH_BLOB DavAuthCallback 콜백 함수에서 검색한 인증 BLOB을 저장합니다.
DAV_CALLBACK_AUTH_UNP DavAuthCallback 콜백 함수에서 검색한 사용자 이름 및 암호 정보를 저장합니다.

DAV_CALLBACK_CRED DavAuthCallback 콜백 함수에서 검색한 사용자 자격 증명 정보를 저장합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

WebSocket 프로토콜 구성 요소 API

아티클2023. 03. 16.

WebSocket 프로토콜 구성 요소 API 기술에 대한 개요입니다.

WebSocket 프로토콜 구성 요소 API를 개발하려면 다음 헤더가 필요합니다.

- websocket.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- WebSocket 프로토콜 구성 요소 API

열거형

WEB_SOCKET_ACTION WebSocket 애플리케이션에서 수행할 작업을 지정합니다.
WEB_SOCKET_ACTION_QUEUE WebSocketGetAction에서 반환되는 작업 유형을 지정합니다.
WEB_SOCKET_BUFFER_TYPE WebSocket 프레임 헤더를 생성하는 데 사용되는 비트 값을 지정합니다.
WEB_SOCKET_CLOSE_STATUS WebSocket 닫기 상태를 지정합니다.
WEB_SOCKET_PROPERTY_TYPE WebSocket 속성 형식을 지정합니다.

함수

WebSocketAbortHandle WebSocketCreateClientHandle 또는 WebSocketCreateServerHandle에서 만든 WebSocket 세션 핸들을 중단합니다.

WebSocketBeginClientHandshake 클라이언트 쪽 핸드셰이크를 시작합니다.
WebSocketBeginServerHandshake 서버 쪽 핸드셰이크를 시작합니다.
WebSocketCompleteAction WebSocketGetAction에서 시작한 작업을 완료합니다.
WebSocketCreateClientHandle 클라이언트 쪽 WebSocket 세션 핸들을 만듭니다.
WebSocketCreateServerHandle 서버 쪽 WebSocket 세션 핸들을 만듭니다.
WebSocketDeleteHandle WebSocketCreateClientHandle 또는 WebSocketCreateServerHandle에서 만든 WebSocket 세션 핸들을 삭제합니다.
WebSocketEndClientHandshake 클라이언트 쪽 핸드셰이크를 완료합니다.
WebSocketEndServerHandshake 서버 쪽 핸드셰이크를 완료합니다.
WebSocketGetAction WebSocketSend, WebSocketReceive 또는 WebSocketCompleteAction에 대한 호출에서 작업을 반환합니다.
WebSocketGetGlobalProperty 단일 WebSocket 속성을 가져옵니다.
WebSocketReceive 프로토콜 구성 요소 작업 큐에 수신 작업을 추가합니다.
WebSocketSend 프로토콜 구성 요소 작업 큐에 보내기 작업을 추가합니다.

구조체

WEB_SOCKET_BUFFER 특정 WebSocket 작업에 대한 데이터를 포함합니다.
WEB_SOCKET_HTTP_HEADER HTTP 헤더를 포함합니다.
WEB_SOCKET_PROPERTY 단일 WebSocket 속성을 포함합니다.

WhqlProvider 공급자

아티클2023. 03. 09.

WhqlProvider 공급자 기술 개요입니다.

WhqlProvider 공급자 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- WhqlProvider 공급자

창 스테이션 및 데스크톱

아티클2023. 08. 22.

창 스테이션 및 데스크톱 기술 개요입니다.

창 스테이션 및 데스크톱 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 창 스테이션 및 데스크톱

Functions

CloseDesktop 데스크톱 개체에 대한 열린 핸들을 닫습니다.
CloseWindowStation 열린 창 스테이션 핸들을 닫습니다.
CreateDesktopA 새 데스크톱을 만들고, 호출 프로세스의 현재 창 스테이션과 연결하고, 호출 스레드에 할당합니다. (ANSI)
CreateDesktopExA 지정된 힙을 사용하여 새 데스크톱을 만들고, 호출 프로세스의 현재 창 스테이션에 연결하고, 호출 스레드에 할당합니다. (ANSI)
CreateDesktopExW 지정된 힙을 사용하여 새 데스크톱을 만들고, 호출 프로세스의 현재 창 스테이션에 연결하고, 호출 스레드에 할당합니다. (유니코드)
CreateDesktopW 새 데스크톱을 만들고, 호출 프로세스의 현재 창 스테이션과 연결하고, 호출 스레드에 할당합니다. (유니코드)
CreateWindowStationA 창 스테이션 개체를 만들고 호출 프로세스와 연결한 다음 현재 세션에 할당합니다. (ANSI)

CreateWindowStationW 창 스테이션 개체를 만들고 호출 프로세스와 연결한 다음 현재 세션에 할당합니다. (유니코드)
EnumDesktopsA 호출 프로세스의 지정된 창 스테이션과 연결된 모든 데스크톱을 열거합니다. 함수는 각 데스크톱의 이름을 애플리케이션 정의 콜백 함수에 차례로 전달합니다. (ANSI)
EnumDesktopsW 호출 프로세스의 지정된 창 스테이션과 연결된 모든 데스크톱을 열거합니다. 함수는 각 데스크톱의 이름을 애플리케이션 정의 콜백 함수에 차례로 전달합니다. (유니코드)
EnumDesktopWindows 지정된 데스크톱과 연결된 모든 최상위 창을 열거합니다. 핸들을 각 창에 차례로 애플리케이션 정의 콜백 함수로 전달합니다.
EnumWindowStationsA 현재 세션의 모든 창 스테이션을 열거합니다. 함수는 각 창 스테이션의 이름을 애플리케이션 정의 콜백 함수에 차례로 전달합니다. (ANSI)
EnumWindowStationsW 현재 세션의 모든 창 스테이션을 열거합니다. 함수는 각 창 스테이션의 이름을 애플리케이션 정의 콜백 함수에 차례로 전달합니다. (유니코드)
GetProcessWindowStation 호출 프로세스에 대한 현재 창 스테이션에 대한 핸들을 검색합니다.
GetThreadDesktop 지정된 스레드에 할당된 데스크톱에 대한 핸들을 검색합니다.
GetUserObjectInformationA 지정된 창 스테이션 또는 데스크톱 개체에 대한 정보를 검색합니다. (ANSI)
GetUserObjectInformationW 지정된 창 스테이션 또는 데스크톱 개체에 대한 정보를 검색합니다. (유니코드)
OpenDesktopA 지정된 데스크톱 개체를 엽니다. (ANSI)

OpenDesktopW 지정된 데스크톱 개체를 엽니다. (유니코드)
OpenInputDesktop 사용자 입력을 수신하는 데스크톱을 엽니다.
OpenWindowStationA 지정된 창 스테이션을 엽니다. (ANSI)
OpenWindowStationW 지정된 창 스테이션을 엽니다. (유니코드)
SetProcessWindowStation 지정된 창 스테이션을 호출 프로세스에 할당합니다.
SetThreadDesktop 호출 스레드에 지정된 데스크톱을 할당합니다. 데스크톱의 모든 후속 작업은 데스크톱에 부여된 액세스 권한을 사용합니다.
SetUserObjectInformationA 지정된 창 스테이션 또는 데스크톱 개체에 대한 정보를 설정합니다. (ANSI)
SetUserObjectInformationW 지정된 창 스테이션 또는 데스크톱 개체에 대한 정보를 설정합니다. (유니코드)
SwitchDesktop 지정된 데스크톱을 표시하고 활성화합니다. 이렇게 하면 데스크톱이 사용자로부터 입력을 받을 수 있습니다.

구조체

USEROBJECTFLAGS 창 스테이션 또는 데스크톱 핸들에 대한 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Windows 접근성 기능

아티클2023. 03. 11.

Windows 접근성 기능 기술 개요입니다.

Windows 접근성 기능을 개발하려면 다음 헤더가 필요합니다.

- oleacc.h
- uiautomationclient.h
- uiautomationcore.h
- uiautomationcoreapi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 접근성 기능

열거형

ActiveEnd 현재 선택한 텍스트를 나타내는 텍스트 범위를 기준으로 캐럿의 위치를 나타내는 SelectionActiveEnd 텍스트 특성에 사용할 수 있는 값을 포함합니다.
AnimationStyle AnimationStyle 텍스트 특성에 대한 값을 포함합니다.
AsyncContentLoadedState 콘텐츠의 비동기 로드 진행률을 설명하는 값을 포함합니다.
AutomationElementMode AutomationElementMode(uiautomationcoreapi.h) 열거형에는 UI 자동화 요소를 반환할 때 사용할 참조 유형을 지정하는 값이 포함되어 있습니다.
AutomationElementMode AutomationElementMode(uiautomationclient.h) 열거형에는 UI 자동화 요소를 반환할 때 사용할 참조 형식을 지정하는 값이 포함되어 있습니다.
AutomationIdentifierType UiaLookupId 함수에 사용되는 값을 포함합니다.

Bulletstyle BulletStyle 텍스트 특성에 대한 값을 포함합니다.
CapStyle CapStyle 텍스트 특성의 값을 지정하는 값을 포함합니다.
CaretBidiMode CaretBidiMode 텍스트 특성에 대한 가능한 값을 포함하며, 이는 caret가 왼쪽에서 오른쪽으로 또는 오른쪽에서 왼쪽으로 흐르는 텍스트에 있는지 여부를 나타냅니다.
CaretPosition CaretPosition 텍스트 특성에 사용할 수 있는 값을 포함하며, 이는 텍스트 범위의 텍스트 줄을 기준으로 하는 caret의 위치를 나타냅니다.
CoalesceEventsOptions 액세스 가능한 기술 클라이언트가 모든 이벤트를 수신하는지 또는 중복 이벤트가 검색되고 필터링되는 하위 집합을 수신하는지 여부를 나타내는 CoalesceEvents 속성에 대한 가능한 값을 포함합니다.
ConditionType UiaCondition 형식을 지정하는 값을 포함합니다.
ConnectionRecoveryBehaviorOptions Access 기술 클라이언트가 공급자가 응답하지 않는 경우 공급자 요청 시간 제한을 조정하는지 여부를 나타내는 ConnectionRecoveryBehavior 속성에 대한 가능한 값을 포함합니다.
DockPosition Dock 컨트롤 패턴으로 표시되는 도킹 창의 위치를 지정하는 값을 포함합니다.
EventArgsType UiaEventArgs 구조체에서 설명하는 이벤트 형식을 지정하는 값을 포함합니다.
ExpandCollapseState 확장 및 축소할 수 있는 UI 요소의 상태를 지정하는 값을 포함합니다.
FillType FillType 특성에 대한 값을 포함합니다.

FlowDirections TextFlowDirections 텍스트 특성에 대한 값을 포함합니다.
LiveSetting LiveSetting 속성에 사용할 수 있는 값을 포함합니다. 이 속성은 라이브 지역의 일부인 공급자 요소에 의해 구현됩니다.
NavigateDirection Microsoft UI 자동화 트리 내에서 탐색 방향을 지정하는 데 사용되는 값을 포함합니다.
NormalizeState UiaGetUpdatedCache의 동작을 지정하는 값을 포함합니다.
NotificationKind 알림 이벤트의 유형을 나타내는 값과 이벤트 처리에 대한 힌트를 수신기에 정의합니다.
NotificationProcessing 알림을 처리하는 방법을 나타내는 값을 정의합니다.
OrientationType 컨트롤의 방향을 지정하는 값을 포함합니다.
OutlineStyles OutlineStyle 텍스트 특성에 대한 값을 포함합니다.
PropertyConditionFlags PropertyConditionFlags(uiautomationcoreapi.h) 열거형에는 속성 조건을 만드는 데 사용되는 값이 포함됩니다.
PropertyConditionFlags PropertyConditionFlags(uiautomationclient.h) 열거형에는 속성 조건을 만드는 데 사용되는 값이 포함됩니다.
ProviderOptions UI 자동화 공급자의 유형을 지정하는 값을 포함합니다. IRawElementProviderSimple::ProviderOptions 속성은 이 열거형을 사용합니다.

ProviderType 클라이언트 쪽(프록시) UI 자동화 공급자의 형식을 지정하는 값을 포함합니다.
RowOrColumnMajor 테이블의 데이터를 주로 행 단위로 읽을지 아니면 열 단위로 읽을지를 지정하는 값을 포함합니다.
SayAsInterpretAs 텍스트 음성 변환 엔진이 특정 데이터를 해석하는 방법을 나타내는 값을 정의합니다.
ScrollAmount 스크롤할 방향과 거리를 지정하는 값을 포함합니다.
StructureChangeType Microsoft UI 자동화 트리 구조의 변경 유형을 지정하는 값을 포함합니다.
SupportedTextSelection 지원되는 텍스트 선택 특성을 지정하는 값을 포함합니다.
SynchronizedInputType 동기화된 입력의 형식을 지정하는 값이 들어 있습니다.
TextDecorationLineStyle OverlineStyle, StrikethroughStyle 및 UnderlineStyle 텍스트 특성을 지정하는 값을 포함합니다.
TextEditChangeType 텍스트 편집 이벤트가 발생하거나 처리될 때 컨트롤에서 수행되는 텍스트 편집 변경에 대해 설명합니다.
TextPatternRangeEndpoint 텍스트 범위의 엔드포인트를 지정하는 값을 포함합니다.
TextUnit 탐색을 위해 텍스트 단위를 지정하는 값을 포함합니다.
ToggleState 토글 컨트롤 패턴을 구현하는 Microsoft UI 자동화 요소의 토글 상태를 지정하는 값을 포함합니다.

TreeScope TreeScope(uiautomationcoreapi.h) 열거형에는 Microsoft UI 자동화 트리에서 다양한 작업의 범위를 지정하는 값이 포함되어 있습니다.
TreeScope TreeScope(uiautomationclient.h) 열거형에는 Microsoft UI 자동화 트리에서 다양한 작업의 범위를 지정하는 값이 포함되어 있습니다.
TreeTraversalOptions TreeTraversalOptions(uiautomationcoreapi.h) 열거형은 트리 탐색 순서를 사용자 지정하는 데 사용할 수 있는 값을 정의합니다.
TreeTraversalOptions TreeTraversalOptions(uiautomationclient.h) 열거형은 트리 탐색 순서를 사용자 지정하는 데 사용할 수 있는 값을 정의합니다.
UIAutomationType Microsoft UI 자동화 데이터 형식을 나타내는 데 사용되는 값을 포함합니다.
VisualEffects VisualEffects 특성에 대한 값을 포함합니다.
WindowInteractionState 사용자 상호 작용을 위해 창의 현재 상태를 지정하는 값을 포함합니다.
WindowVisualState 창의 표시 상태를 지정하는 값을 포함합니다.
ZoomUnit 지정된 단위로 컨트롤의 뷰포트를 확대/축소하는 IUIAutomationTransformPattern2::ZoomByUnit 메서드에 사용할 수 있는 값을 포함합니다.

Functions

accDoDefaultAction IAccessible::accDoDefaultAction 메서드는 지정된 개체의 기본 작업을 수행합니다. 모든 개체마다 기본 작업이 있는 것은 아닙니다.
AccessibleChildren 액세스 가능한 컨테이너 개체 내에서 각 자식의 자식 ID 또는 IDispatch를 검색합니다.
AccessibleObjectFromEvent 클라이언트의 이벤트 후크 함수에서 현재 처리 중인 이벤트를 생성한 개체에 대한 IAccessible 인터페이스의 주소를 검색합니다.
AccessibleObjectFromID AccessibleObjectFromID 메서드는 지정된 개체 ID와 연결된 인터페이스에 대한 IAccessibleinterface 포인터를 검색합니다. Oleacc.dll 이 메서드를 사용하여 다른 코드에서 제공하는 프록시에 대한 IAccessible 인터페이스 포인터를 가져옵니다.
AccessibleObjectFromPoint 화면의 지정된 지점에 표시되는 개체에 대한 IAccessible 인터페이스 포인터의 주소를 검색합니다.
AccessibleObjectFromWindow 지정된 창과 연결된 개체에 대해 지정된 인터페이스의 주소를 검색합니다.
accHitTest IAccessible::accHitTest 메서드는 화면의 특정 지점에 표시되는 자식 요소 또는 자식 개체를 검색합니다.
accLocation IAccessible::accLocation 메서드는 지정된 개체의 현재 화면 위치를 검색합니다. 모든 시각적 개체는 이 메서드를 지원해야 합니다. 소리 개체는 이 메서드를 지원하지 않습니다.
accNavigate IAccessible::accNavigate 메서드는 컨테이너 내의 다른 UI 요소로 트래버스하고 개체를 검색합니다. 이 메서드는 선택 사항입니다.
AccNotifyTouchInteraction AT(보조 기술) 애플리케이션이 사용자의 터치 제스처로 인해 Windows Automation API(예: Microsoft UI 자동화)를 통해 UI와 상호 작용하고 있음을 시스템에 알릴 수 있습니다.

accSelect IAccessible::accSelect 메서드는 선택 영역을 수정하거나 지정된 개체의 키보드 포커스를 이동합니다. 선택하거나 키보드 포커스를 받는 모든 개체는 이 메서드를 지원해야 합니다.
AccSetRunningUtilityState AT(보조 기술) 애플리케이션의 현재 상태가 일반적으로 시스템에서 제공하는 기능에 영향을 미치는지 여부를 나타내는 시스템 값을 설정합니다.
AcquireObjectIdRange 컨트롤 호스트에서 개체 ID의 범위를 획득하고 특정 창 없는 컨트롤에 의해 예약된 것으로 표시합니다.
AddActiveTextPositionChangedEventHandler 현재 텍스트 위치가 변경되는 시기를 처리하는 메서드를 등록합니다.
AddActiveTextPositionChangedEventHandler 활성 텍스트 위치가 변경되는 시기를 처리하는 메서드를 이벤트 처리기 그룹에 등록합니다.
AddAutomationEvent Microsoft UI 자동화 이벤트를 발생합니다.
AddAutomationEventHandler Microsoft UI 자동화 이벤트를 처리하는 메서드를 등록합니다. 참고 이벤트 처리기를 구현하기 전에 스레딩 문제 이해에 설명된 스레딩 문제에 대해 잘 알고 있어야 합니다.
AddAutomationEventHandler Microsoft UI 자동화 이벤트를 처리하는 메서드를 등록합니다.
AddAutomationPropertyChangedEvent 속성 변경 이벤트를 발생합니다.
AddChangesEventHandler 변경 이벤트를 처리하는 메서드를 등록합니다. 참고 이벤트 처리기를 구현하기 전에 스레딩 문제 이해에 설명된 스레딩 문제에 대해 잘 알고 있어야 합니다.
AddChangesEventHandler 변경 이벤트를 처리하는 메서드를 등록합니다.

AddEventHandlerGroup CreateEventHandlerGroup에 지정된 이벤트 처리기 메서드 컬렉션을 등록합니다.
AddFocusChangedEventHandler 포커스 변경 이벤트를 처리하는 메서드를 등록합니다. 참고 이벤트 처리기를 구현하기 전에 스레딩 문제 이해에 설명된 스레딩 문제에 대해 잘 알고 있어야 합니다.
AddNotificationEventHandler 알림 이벤트를 처리하는 메서드를 등록합니다. 참고 이벤트 처리기를 구현하기 전에 스레딩 문제 이해에 설명된 스레딩 문제에 대해 잘 알고 있어야 합니다.
AddNotificationEventHandler 알림 이벤트를 처리하는 메서드를 등록합니다.
AddPattern 캐시 요청에 컨트롤 패턴을 추가합니다.
AddProperty 캐시 요청에 속성을 추가합니다.
AddPropertyChangedEventHandler 속성 변경 이벤트의 배열과 처리되는 메서드를 등록합니다.
AddPropertyChangedEventHandler 속성 변경 이벤트를 처리하는 메서드를 등록합니다.
AddPropertyChangedEventHandlerNativeArray 속성 변경 이벤트의 네이티브 배열을 처리하는 메서드를 등록합니다.
AddStructureChangedEvent UI 자동화 트리의 구조가 변경되었음을 클라이언트에 알리는 이벤트를 발생합니다.
AddStructureChangedEventHandler 구조 변경 이벤트를 처리하는 메서드를 등록합니다. 참고 이벤트 처리기를 구현하기 전에 스레딩 문제 이해에 설명된 스레딩 문제에 대해 잘 알고 있어야 합니다.

AddStructureChangedEventHandler 구조 변경 이벤트를 처리하는 메서드를 등록합니다.
AddTextEditTextChangedEventHandler 프로그래밍 방식 텍스트 편집 이벤트를 처리하는 메서드를 등록합니다. 참고 이벤트 처리기를 구현하기 전에 스레딩 문제 이해에 설명된 스레딩 문제에 대해 잘 알고 있어야 합니다.
AddTextEditTextChangedEventHandler 프로그래밍 방식 텍스트 편집 이벤트를 처리하는 메서드를 등록합니다.
AddToSelection 선택한 항목 컬렉션에 현재 요소를 추가합니다. (ISelectionItemProvider.AddToSelection)
AddToSelection 선택한 텍스트의 여러 연결되지 않은 범위를 지원하는 컨트롤에서 선택한 텍스트 범위의 컬렉션에 텍스트 범위를 추가합니다. (ITextRangeProvider.AddToSelection)
AddToSelection 선택한 항목 컬렉션에 현재 요소를 추가합니다. (IUIAutomationSelectionItemPattern.AddToSelection)
AddToSelection 선택한 텍스트의 여러 연결되지 않은 범위를 지원하는 컨트롤에서 선택한 텍스트 범위의 컬렉션에 텍스트 범위를 추가합니다. (IUIAutomationTextRange.AddToSelection)
AdviseEventAdded UI 자동화 클라이언트가 속성 변경 이벤트를 포함하여 특정 이벤트를 수신 대기하기 시작하면 Microsoft UI 자동화 공급자에게 알합니다.
AdviseEventRemoved UI 자동화 클라이언트가 속성 변경 이벤트를 포함하여 특정 이벤트 수신 대기를 중지하면 Microsoft UI 자동화 공급자에게 알합니다.
BuildUpdatedCache 업데이트된 캐시를 사용하여 새 UI 자동화 요소를 검색합니다.

CallMethod 클라이언트 래퍼는 이 CallMethod 함수를 호출하고 매개 변수를 포인터 배열로 지정하여 메서드를 구현합니다.
취소 입력 수신 대기를 취소합니다.
취소 Microsoft UI 자동화 공급자가 마우스 또는 키보드 입력에 대한 수신을 중지하도록 합니다.
CheckNotSupported 제공된 VARIANT를 검사하여 지원되지 않는 식별자가 포함되어 있는지 확인합니다.
ClearHmenuProps 이 메서드는 ClearProps를 래핑하고 HMENU 기반 액세스 가능 요소에 주석을 지정하는 호출자에게 편리한 진입점을 제공합니다.
ClearHwndProps 이 메서드는 SetPropValue, SetPropServer 및 ClearProps를 래핑하고 HWND 기반 액세스 가능 요소에 주석을 추가한 호출자에게 편리한 진입점을 제공합니다.
ClearProps 서버는 ClearProps를 사용하여 이전에 주석을 추가한 액세스 가능한 요소의 속성으로 기본값을 복원합니다.
ClearTable 프록시 팩터리 테이블에서 모든 항목을 제거합니다.
복제 원래 ITextRangeProvider와 동일하고 원본의 모든 속성을 상속하는 새 ITextRangeProvider를 반환합니다.
복제 캐시 요청의 복사본을 만듭니다.
복제 원본과 동일한 새 IUIAutomationTextRange를 검색하고 원본의 모든 속성을 상속합니다.

닫기 창을 닫으려고 시도합니다.
닫기 창을 닫습니다.
CoCreateLocally 클라이언트는 ICoCreateLocally::CoCreateLocally를 호출하여 서버 개체와 동일한 컨텍스트에서 도우미 개체를 만듭니다.
축소 이 요소의 모든 자식 노드, 컨트롤 또는 콘텐츠를 숨깁니다.
축소 요소의 모든 자식 노드, 컨트롤 또는 콘텐츠를 숨깁니다.
비교 이 텍스트 범위의 엔드포인트가 다른 텍스트 범위와 같은지 여부를 지정하는 값을 검색합니다. (ITextRangeProvider.Compare)
비교 이 텍스트 범위의 엔드포인트가 다른 텍스트 범위와 같은지 여부를 지정하는 값을 검색합니다. (IUIAutomationTextRange.Compare)
CompareElements 두 UI 자동화 요소를 비교하여 동일한 기본 UI 요소를 나타내는지 여부를 확인합니다.
CompareEndpoints 두 텍스트 범위의 엔드포인트이 동일한지 여부를 나타내는 값을 반환합니다.
CompareEndpoints 이 텍스트 범위의 시작 또는 끝점이 다른 텍스트 범위의 시작 또는 끝점과 같은지 여부를 지정하는 값을 검색합니다.
CompareRuntimeIds ID(런타임 식별자)가 포함된 두 정수 배열을 비교하여 해당 콘텐츠가 동일하고 동일한 UI 요소에 속하는지 여부를 확인합니다.

ComposeHmenuIdentityString 호출자는 ComposeHmenuIdentityString을 사용하여 HMENU 기반 액세스 가능 요소에 대한 ID 문자열을 검색합니다.
ComposeHwndIdentityString 호출자는 ComposeHwndIdentityString을 사용하여 ID 문자열을 검색합니다.
ConvertReturnedElement 속성 값으로 반환된 요소의 IAccessibleEx 인터페이스를 검색합니다.
ConvertValueToString 클라이언트는 IAccDictionary::ConvertValueToString 메서드를 호출하여 값을 지역화된 문자열로 변환합니다.
CreateAndCondition 두 조건 모두와 일치하는 요소를 선택하는 조건을 만듭니다.
CreateAndConditionFromArray 여러 조건에 따라 요소를 선택하는 조건을 만듭니다. 모두 true여야 합니다.
CreateAndConditionFromNativeArray 모두 true여야 하는 여러 조건에 따라 네이티브 배열에서 요소를 선택하는 조건을 만듭니다.
CreateCacheRequest 캐시 요청을 만듭니다.
CreateClientWrapper 클라이언트 애플리케이션이 사용자 지정 컨트롤 패턴과 상호 작용할 수 있도록 하는 개체를 만듭니다.
CreateEventHandlerGroup 하나 이상의 이벤트 수신기를 단일 메서드 호출에 등록합니다.
CreateFalseCondition 항상 false인 조건을 만듭니다.

CreateNotCondition 지정된 조건의 음수인 조건을 만듭니다.
CreateOrCondition 두 조건 중 하나가 true인 경우 일치 항목이 있는 두 조건의 조합을 만듭니다.
CreateOrConditionFromArray 조건이 true인 경우 일치 항목이 있는 둘 이상의 조건 조합을 만듭니다.
CreateOrConditionFromNativeArray 조건 중 하나가 true인 경우 일치 항목이 있는 둘 이상의 조건 조합을 만듭니다.
CreatePropertyCondition 지정된 값을 가진 속성이 있는 요소를 선택하는 조건을 만듭니다.
CreatePropertyConditionEx 선택적 플래그를 사용하여 지정된 값을 가진 속성이 있는 요소를 선택하는 조건을 만듭니다.
CreateProvider UI 요소에 대한 Microsoft UI 자동화 지원을 제공하는 프록시 개체를 만듭니다.
CreateProxyFactoryEntry 프록시 팩터리 개체의 새 인스턴스를 만듭니다.
CreateStdAccessibleObject 시스템 제공 사용자 인터페이스 요소의 지정된 형식의 메서드 및 속성을 사용하여 액세스 가능한 개체를 만듭니다.
CreateStdAccessibleProxyA 시스템 제공 사용자 인터페이스 요소의 지정된 클래스에 대한 속성과 메서드가 있는 액세스 가능한 개체를 만듭니다. (ANSI)
CreateStdAccessibleProxyW 시스템 제공 사용자 인터페이스 요소의 지정된 클래스에 대한 속성과 메서드가 있는 액세스 가능한 개체를 만듭니다. (유니코드)

CreateTreeWalker Microsoft UI 자동화 트리를 트래버스하는 데 사용할 수 있는 트리 워커 개체를 검색합니다.
CreateTrueCondition 모든 요소를 선택하는 미리 정의된 조건을 검색합니다.
DecomposeHmenuIdentityString 이 메서드를 사용하여 ID 문자열로 식별되는 액세스 가능한 요소의 HMENU, 개체 ID 및 자식 ID를 확인합니다.
DecomposeHwndIdentityString 이 메서드를 사용하여 ID 문자열로 식별된 액세스 가능한 요소의 HWND, 개체 ID 및 자식 ID를 확인합니다.
Dispatch 메서드 또는 속성 getter를 사용자 지정 컨트롤 패턴 공급자로 디스패치합니다.
DockPattern_SetDockPosition 요청된 dockPosition에서 UI 자동화 요소를 도킹 컨테이너 내에 도킹합니다.
DoDefaultAction 컨트롤에 대한 기본 작업을 수행합니다.
DoDefaultAction 요소에 대한 Microsoft Active Accessibility 기본 작업을 수행합니다. (IUIAutomationLegacyIAccessiblePattern.DoDefaultAction)
ElementFromHandle 지정된 창에 대한 UI 자동화 요소를 검색합니다.
ElementFromHandleBuildCache 지정된 창에 대한 UI 자동화 요소를 검색하고, 요청된 속성 및 컨트롤 패턴을 프리페치하고, 프리페치된 항목을 캐시에 저장합니다.
ElementFromIAccessible Microsoft Active Accessibility 서버에서 지정된 액세스 가능한 개체에 대한 UI 자동화 요소를 검색합니다.

ElementFromIAccessibleBuildCache Microsoft Active Accessibility 서버에서 지정된 액세스 가능한 개체에 대한 UI 자동화 요소를 검색하고, 요청된 속성 및 컨트롤 패턴을 프리페치하고, 프리페치된 항목을 캐시에 저장합니다.
ElementFromPoint 바탕 화면의 지정된 지점에서 UI 자동화 요소를 검색합니다.
ElementFromPointBuildCache 바탕 화면의 지정된 지점에서 UI 자동화 요소를 검색하고, 요청된 속성 및 컨트롤 패턴을 프리페치하고, 프리페치된 항목을 캐시에 저장합니다.
ElementProviderFromPoint 이 조각의 지정된 지점에 있는 요소의 공급자를 검색합니다.
확장 컨트롤의 모든 자식 노드, 컨트롤 또는 내용을 표시합니다.
확장 요소의 모든 자식 노드, 컨트롤 또는 콘텐츠를 표시합니다.
ExpandCollapsePattern_Collapse UI 자동화 요소의 모든 하위 노드, 컨트롤 또는 콘텐츠를 숨깁니다.
ExpandCollapsePattern_Expand 더 많은 정보를 표시할 수 있도록 화면에서 컨트롤을 확장합니다.
ExpandToEnclosingUnit 지정한 텍스트 단위로 텍스트 범위를 정규화합니다. 범위가 지정된 단위보다 작으면 확장되거나 지정된 단위보다 길면 줄입니다. (ITextRangeProvider.ExpandToEnclosingUnit)
ExpandToEnclosingUnit 지정한 텍스트 단위로 텍스트 범위를 정규화합니다. 범위가 지정된 단위보다 작으면 확장되거나 지정된 단위보다 길면 줄입니다. (IUIAutomationTextRange.ExpandToEnclosingUnit)
Findall 지정된 조건을 충족하는 모든 UI 자동화 요소를 반환합니다.

FindAllBuildCache 지정된 조건을 충족하는 모든 UI 자동화 요소를 반환하고, 요청된 속성 및 컨트롤 패턴을 프리페치하고, 프리페치된 항목을 캐시에 저장합니다.
FindAllWithOptions 지정된 순서로 일치하는 모든 요소를 찾습니다.
FindAllWithOptionsBuildCache 지정된 순서로 일치하는 모든 요소를 찾은 다음 해당 속성과 패턴도 캐시합니다.
FindAttribute 지정된 텍스트 특성 값이 있는 텍스트 범위 하위 집합을 반환합니다.
FindAttribute 지정된 텍스트 특성 값이 있는 텍스트 범위 하위 집합을 검색합니다.
FindFirst 지정된 조건과 일치하는 첫 번째 자식 또는 하위 요소를 검색합니다.
FindFirstBuildCache 지정된 조건과 일치하는 첫 번째 자식 또는 하위 요소를 검색하고, 요청된 속성 및 컨트롤 패턴을 프리페치하고, 프리페치된 항목을 캐시에 저장합니다.
FindFirstWithOptions 지정된 순서로 일치하는 첫 번째 요소를 찾습니다.
FindFirstWithOptionsBuildCache 지정된 순서로 일치하는 첫 번째 요소를 찾은 다음 해당 속성과 패턴도 캐시합니다.
FindItemByProperty 지정된 속성 값에 따라 포함하는 요소 내의 요소를 검색합니다. (IItemContainerProvider.FindItemByProperty)
FindItemByProperty 지정된 속성 값에 따라 포함하는 요소 내의 요소를 검색합니다. (IUIAutomationItemContainerPattern.FindItemByProperty)

Findtext 지정된 텍스트를 포함하는 텍스트 범위 하위 집합을 반환합니다.
Findtext 지정된 텍스트가 포함된 텍스트 범위 하위 집합을 검색합니다.
get_accChild IAccessible::get_accChild 메서드는 지정된 자식에 대한 IDispatch(있는 경우)를 검색합니다. 모든 개체는 이 속성을 지원해야 합니다.
get_accChildCount IAccessible::get_accChildCount 메서드는 이 개체에 속하는 자식 수를 검색합니다. 모든 개체는 이 속성을 지원해야 합니다.
get_accDefaultAction IAccessible::get_accDefaultAction 메서드는 개체의 기본 동작을 나타내는 문자열을 검색합니다. 모든 개체마다 기본 작업이 있는 것은 아닙니다.
get_accDescription IAccessible::get_accDescription 메서드는 지정된 개체의 시각적 모양을 설명하는 문자열을 검색합니다. 모든 개체에 설명이 있는 것은 아닙니다.
get_accFocus IAccessible::get_accFocus 메서드는 키보드 포커스가 있는 개체를 검색합니다. 키보드 포커스를 받을 수 있는 모든 개체는 이 속성을 지원해야 합니다.
get_accHelp IAccessible::get_accHelp 메서드는 개체의 도움말 속성 문자열을 검색합니다. 모든 개체가 이 속성을 지원하지는 않습니다.
get_accHelpTopic IAccessible::get_accHelpTopic 메서드는 지정된 개체와 연결된 WinHelp 파일의 전체 경로를 검색합니다. 또한 해당 파일 내에서 적절한 토픽의 식별자를 검색합니다.
get_accKeyboardShortcut IAccessible::get_accKeyboardShortcut 메서드는 지정된 개체의 바로 가기 키 또는 니모닉이라고도 하는 액세스 키를 검색합니다. 바로 가기 키 또는 액세스 키가 있는 모든 개체는 이 속성을 지원합니다.

get_accName IAccessible::get_accName 메서드는 지정된 개체의 이름을 검색합니다. 모든 개체는 이 속성을 지원합니다.
get_accParent IAccessible::get_accParent 메서드는 개체 부모의 IDispatch를 검색합니다. 모든 개체는 이 속성을 지원합니다.
get_accRole IAccessible::get_accRole 메서드는 지정된 개체의 역할을 설명하는 정보를 검색합니다. 모든 개체는 이 속성을 지원합니다.
get_accSelection IAccessible::get_accSelection 메서드는 이 개체의 선택한 자식을 검색합니다. 선택을 지원하는 모든 개체는 이 속성을 지원해야 합니다.
get_accState IAccessible::get_accState 메서드는 지정된 개체의 현재 상태를 검색합니다. 모든 개체는 이 속성을 지원합니다.
get_accValue IAccessible::get_accValue 메서드는 지정된 개체의 값을 검색합니다. 모든 개체에 값이 있는 것은 아닙니다.
get_AllowSubstringMatch 프록시에서 부분 문자열 일치를 허용하는지 여부를 지정하는 값을 설정하거나 검색합니다. (Get)
get_AnnotationTypeId 이 주석의 주석 형식 식별자입니다.
get_AnnotationTypeName 이 주석 형식의 이름입니다.
get_Author 주석 작성자의 이름입니다.

get_AutomationElementMode 반환된 요소에 기본 UI에 대한 전체 참조가 포함되어 있는지 또는 캐시된 정보만 포함되는지 여부를 나타냅니다. (Get)
get_AutoSetFocus UI 자동화 컨트롤 패턴 메서드에 대한 호출이 자동으로 대상 요소에 포커스를 설정할지 여부를 지정합니다. (Get)
get_BooleanValue 조건:_either TRUE 또는 FALSE 값을 검색합니다.
get_BoundingRectangle 이 요소의 경계 사각형을 지정합니다.
get_CachedAcceleratorKey 요소의 캐시된 가속기 키를 검색합니다.
get_CachedAccessKey 요소의 캐시된 액세스 키 문자를 검색합니다.
get_CachedAnnotationObjects 주석, 머리글, 바닥글 등 이 요소와 연결된 주석 개체의 캐시된 목록을 반환합니다.
get_CachedAnnotationTypeId 이 주석의 형식을 식별하는 캐시된 값을 검색합니다.
get_CachedAnnotationTypeName 이 주석 형식의 캐시된 지역화된 이름을 검색합니다.
get_CachedAnnotationTypes 주석, 머리글, 바닥글 등 이 요소와 연결된 주석 형식의 캐시된 목록을 반환합니다.
get_CachedAriaProperties 요소의 캐시된 ARIA(Access Rich Internet Applications) 속성을 검색합니다.
get_CachedAriaRole 요소의 캐시된 ARIA(Access Rich Internet Applications) 역할을 검색합니다.

get_CachedAuthor 주석 작성자의 캐시된 이름을 검색합니다.
get_CachedAutomationId 요소의 캐시된 Microsoft UI 자동화 식별자를 검색합니다.
get_CachedBoundingRectangle 요소를 완전히 묶는 사각형의 캐시된 좌표를 검색합니다.
get_CachedCanMaximize 창을 최대화할 수 있는지 여부를 나타내는 캐시된 값을 검색합니다.
get_CachedCanMinimize 창을 최소화할 수 있는지 여부를 나타내는 캐시된 값을 검색합니다.
get_CachedCanMove 요소를 이동할 수 있는지 여부를 나타내는 캐시된 값을 검색합니다.
get_CachedCanResize 요소의 크기를 조정할 수 있는지 여부를 나타내는 캐시된 값을 검색합니다.
get_CachedCanRotate 요소를 회전할 수 있는지 여부를 나타내는 캐시된 값을 검색합니다.
get_CachedCanSelectMultiple 컨테이너에서 둘 이상의 항목을 한 번에 선택할 수 있는지 여부를 나타내는 캐시된 값을 검색합니다.
get_CachedCanZoom 컨트롤이 뷰포트의 확대/축소를 지원하는지 여부를 나타내는 캐시된 값을 검색합니다.
get_CachedChildId 요소에 대해 캐시된 Microsoft Active Accessibility 자식 식별자를 검색합니다.
get_CachedClassName 요소의 캐시된 클래스 이름을 검색합니다.

get_CachedColumn 그리드 항목이 포함된 열의 캐시된 인덱스(0부터 시작하는 인덱스)를 검색합니다.
get_CachedColumnCount 표에서 캐시된 열 수를 검색합니다.
get_CachedColumnSpan 그리드 항목에 의해 확장된 캐시된 열 수를 검색합니다.
get_CachedContainingGrid 그리드 항목이 포함된 캐시된 요소를 검색합니다.
get_CachedControllerFor 이 요소가 컨트롤러 역할을 하는 UI 자동화 요소의 캐시된 배열을 검색합니다.
get_CachedControlType 요소의 컨트롤 형식을 나타내는 캐시된 값을 검색합니다.
get_CachedCulture 요소와 연결된 문화권을 나타내는 캐시된 값을 검색합니다.
get_CachedCurrentSelectedItem 현재 선택한 항목을 나타내는 캐시된 IUIAutomationElement 개체를 가져옵니다.
get_CachedCurrentView 컨트롤의 현재 뷰에 대한 캐시된 컨트롤별 식별자를 검색합니다.
get_CachedDateTime 이 주석이 만들어진 캐시된 날짜 및 시간을 검색합니다.
get_CachedDefaultAction 요소에 대한 Microsoft Active Accessibility 캐시된 기본 작업을 검색합니다.
get_CachedDescribedBy 이 요소를 설명하는 캐시된 요소 배열을 검색합니다.

get_CachedDescription 요소의 캐시된 Microsoft Active Accessibility 설명을 검색합니다.
get_CachedDockPosition 도킹 컨테이너 내에서 이 요소의 캐시된 도크 위치를 검색합니다.
get_CachedDropEffect 사용자가 끌어서 놓기 작업의 일부로 이 요소를 삭제할 때 발생하는 작업을 나타내는 캐시된 지역화된 문자열을 검색합니다.
get_CachedDropEffects 사용자가 끌어서 놓기 작업의 일부로 이 요소를 삭제할 때 발생할 수 있는 전체 효과 집합을 열거하는 지역화된 문자열의 캐시된 배열을 검색합니다.
get_CachedDropTargetEffect 사용자가 이 놓기 대상에서 잡은 요소를 삭제할 때 발생하는 작업을 설명하는 캐시된 지역화된 문자열을 검색합니다.
get_CachedDropTargetEffects 사용자가 끌어서 놓기 작업의 일부로 이 놓기 대상에서 잡기 요소를 삭제할 때 발생할 수 있는 전체 효과 집합을 열거하는 지역화된 문자열의 캐시된 배열을 검색합니다.
get_CachedExpandCollapseState 요소의 확장 또는 축소 상태를 나타내는 캐시된 값을 검색합니다.
get_CachedExtendedProperties 문서의 요소에 대한 확장 속성 목록이 포함된 캐시된 지역화된 문자열을 검색합니다.
get_CachedFillColor 문서에서 요소의 캐시된 채우기 색을 검색합니다.
get_CachedFillPatternColor 문서의 요소를 채우는 데 사용되는 패턴의 캐시된 색을 검색합니다.
get_CachedFirstSelectedItem 선택한 항목 그룹의 첫 번째 항목을 나타내는 캐시된 IUIAutomationElement 개체를 가져옵니다.

get_CachedFlowsFrom 현재 요소 앞의 읽기 순서를 나타내는 캐시된 요소 배열을 검색합니다.
get_CachedFlowsTo 현재 요소 뒤의 읽기 순서를 나타내는 캐시된 요소 배열을 검색합니다.
get_CachedFormula 이 셀에 대해 캐시된 수식을 검색합니다.
get_CachedFrameworkId 요소와 연결된 기본 UI 프레임워크의 캐시된 이름을 검색합니다.
get_CachedFullDescription 자동화 요소에 대한 캐시된 전체 설명을 가져옵니다.
get_CachedHasKeyboardFocus 요소에 키보드 포커스가 있는지 여부를 나타내는 캐시된 값입니다.
get_CachedHeadingLevel Automation 요소의 캐시된 제목 수준을 가져옵니다.
get_CachedHelp 요소에 대해 캐시된 Microsoft Active Accessibility 도움말 문자열을 검색합니다.
get_CachedHelpText 요소의 캐시된 도움말 텍스트를 검색합니다.
get_CachedHorizontallyScrollable 요소가 가로로 스크롤할 수 있는지 여부를 나타내는 캐시된 값을 검색합니다.
get_CachedHorizontalScrollPercent 캐시된 가로 스크롤 위치를 검색합니다.
get_CachedHorizontalViewSize 스크롤 가능한 요소의 볼 수 있는 영역의 캐시된 가로 크기를 검색합니다.

get_CachedIsContentElement 요소가 콘텐츠 요소인지 여부를 나타내는 캐시된 값입니다.
get_CachedIsControlElement 요소가 컨트롤 요소인지 여부를 나타내는 캐시된 값을 검색합니다.
get_CachedIsDataValidForForm 요소에 양식에 대한 유효한 데이터가 포함되어 있는지 여부를 나타내는 캐시된 값을 검색합니다.
get_CachedIsDialog 캐시된 요소의 대화 상자 창 표시기를 검색합니다.
get_CachedIsEnabled 요소가 사용되는지 여부를 나타내는 캐시된 값을 검색합니다.
get_CachedIsGrabbed 이 요소가 끌어서 놓기 작업의 일부로 잡았는지 여부를 나타내는 캐시된 값을 검색합니다.
get_CachedIsKeyboardFocusable 요소가 키보드 포커스를 허용할 수 있는지 여부를 나타내는 캐시된 값을 검색합니다.
get_CachedIsModal 창이 모달인지 여부를 나타내는 캐시된 값을 검색합니다.
get_CachedIsOffscreen 요소가 화면 끄기인지 여부를 나타내는 캐시된 값을 검색합니다.
get_CachedIsPassword 요소에 위장된 암호가 포함되어 있는지 여부를 나타내는 캐시된 값을 검색합니다.
get_CachedIsPeripheral 요소에 대해 캐시된 주변 장치 UI 표시기를 검색합니다.
get_CachedIsReadOnly 요소의 값을 변경할 수 있는지 여부를 나타내는 캐시된 값을 검색합니다.

get_CachedIsReadOnly 요소의 값이 읽기 전용인지 여부를 나타내는 캐시된 값을 검색합니다.
get_CachedIsRequiredForForm 폼에서 요소를 작성해야 하는지 여부를 나타내는 캐시된 값을 검색합니다.
get_CachedIsSelected 이 항목이 선택되었는지 여부를 나타내는 캐시된 값입니다.
get_CachedIsSelectionRequired 항상 하나 이상의 항목을 선택해야 하는지 여부를 나타내는 캐시된 값을 검색합니다.
get_CachedIsTopmost 창이 z 순서의 최상위 요소인지 여부를 나타내는 캐시된 값을 검색합니다.
get_CachedItemCount 선택한 항목 수를 나타내는 캐시된 정수 값을 가져옵니다.
get_CachedItemStatus 요소 내의 항목 상태에 대한 캐시된 설명을 검색합니다.
get_CachedItemType 요소가 나타내는 항목의 형식을 설명하는 캐시된 문자열을 검색합니다.
get_CachedKeyboardShortcut 요소의 캐시된 Microsoft Active Accessibility 바로 가기 키 속성을 검색합니다.
get_CachedLabeledBy 이 요소의 텍스트 레이블을 포함하는 캐시된 요소를 검색합니다.
get_CachedLandmarkType Automation 요소에 대해 캐시된 랜드마크 형식 ID를 가져옵니다.
get_CachedLargeChange PAGE DOWN 키를 누를 때와 같이 큰 변경이 있을 때 캐시에서 컨트롤 값에 추가되거나 빼는 값을 검색합니다.

get_CachedLastSelectedItem 선택한 항목 그룹의 마지막 항목을 나타내는 캐시된 IUIAutomationElement 개체를 가져옵니다.
get_CachedLevel 요소의 수준(계층 구조)에 대해 캐시된 1 기반 정수를 반환합니다.
get_CachedLiveSetting 요소의 콘텐츠가 변경될 때 요소가 보내는 알림 유형(있는 경우)을 나타내는 캐시된 값을 검색합니다.
get_CachedLocalizedControlType 요소의 컨트롤 형식에 대한 캐시된 지역화된 설명을 검색합니다.
get_CachedLocalizedLandmarkType Automation 요소에 대해 캐시된 지역화된 랜드마크 형식을 포함하는 문자열을 가져옵니다.
get_CachedMaximum 컨트롤의 캐시된 최대값을 검색합니다.
get_CachedMinimum 컨트롤의 캐시된 최소값을 검색합니다.
get_CachedName 요소의 캐시된 이름을 검색합니다.
get_CachedName 요소의 캐시된 Microsoft Active Accessibility 이름 속성을 검색합니다.
get_CachedNativeWindowHandle 요소의 캐시된 창 핸들을 검색합니다.
get_CachedOptimizeForVisualContent 공급자가 표시되는 요소만 노출하는지 여부를 나타내는 캐시된 값을 검색합니다.
get_CachedOrientation 요소의 방향을 나타내는 캐시된 값을 검색합니다.

get_CachedPositionInSet 요소 집합의 서수 위치에 대해 캐시된 1 기반 정수 값을 반환합니다.
get_CachedProcessId 요소를 호스트하는 프로세스의 캐시된 ID를 검색합니다.
get_CachedProviderDescription 이 요소에 대한 공급자의 캐시된 설명을 검색합니다.
get_CachedRole 요소의 캐시된 Microsoft Active Accessibility 역할을 검색합니다.
get_CachedRow 항목이 포함된 행의 캐시된 인덱스(0부터 시작하는 인덱스)를 검색합니다.
get_CachedRowCount 표에서 캐시된 행 수를 검색합니다.
get_CachedRowOrColumnMajor 테이블에 대해 캐시된 기본 순회 방향을 검색합니다.
get_CachedRowSpan 그리드 항목에 걸쳐 있는 캐시된 행 수를 검색합니다.
get_CachedSelectionContainer IUIAutomationSelectionPattern을 지원하는 캐시된 요소를 검색하고 이 항목의 컨테이너 역할을 합니다.
get_CachedShape 문서에서 요소의 캐시된 모양을 검색합니다.
get_CachedSizeOfSet 요소가 있는 집합의 크기에 대해 캐시된 1 기반 정수를 반환합니다.
get_CachedSmallChange 화살표 키를 누를 때와 같이 약간 변경될 때 컨트롤 값에 추가되거나 컨트롤 값에서 빼는 값을 캐시에서 검색합니다.

get_CachedState 요소에 대해 캐시된 Microsoft Active Accessibility 상태 식별자를 검색합니다.
get_CachedStyleId 문서의 요소와 연결된 비주얼 스타일의 캐시된 식별자를 검색합니다.
get_CachedStyleName 문서의 요소와 연결된 비주얼 스타일의 캐시된 이름을 검색합니다.
get_CachedTarget 주석이 추가되는 캐시된 요소를 검색합니다.
get_CachedToggleState 컨트롤의 캐시된 상태를 검색합니다.
get_CachedValue 캐시된 Microsoft Active Accessibility 값 속성을 검색합니다.
get_CachedValue 컨트롤의 캐시된 값을 검색합니다.
get_CachedValue 요소의 캐시된 값을 검색합니다.
get_CachedVerticallyScrollable 요소가 세로로 스크롤할 수 있는지 여부를 나타내는 캐시된 값을 검색합니다.
get_CachedVerticalScrollPercent 캐시된 세로 스크롤 위치를 검색합니다.
get_CachedVerticalViewSize 스크롤 가능한 요소의 볼 수 있는 영역의 캐시된 세로 크기를 검색합니다.
get_CachedWindowInteractionState 사용자 상호 작용을 위해 창의 현재 상태를 나타내는 캐시된 값을 검색합니다.

get_CachedWindowVisualState 창의 시각적 상태를 나타내는 캐시된 값을 검색합니다. 즉, 정상, 최대화 또는 최소화 상태인지 여부입니다.
get_CachedZoomLevel 컨트롤 뷰포트의 캐시된 확대/축소 수준을 검색합니다.
get_CachedZoomMaximum 컨트롤 뷰포트의 캐시된 최대 확대/축소 수준을 검색합니다.
get_CachedZoomMinimum 컨트롤 뷰포트의 캐시된 최소 확대/축소 수준을 검색합니다.
get_CanCheckBaseClass 프록시 팩터리를 검색할 때 기본 클래스를 검사할 수 있는지 여부를 지정하는 값을 설정하거나 검색합니다. (Get)
get_CanMaximize 창을 최대화할 수 있는지 여부를 나타냅니다. (IWindowProvider.get_CanMaximize)
get_CanMinimize 창을 최소화할 수 있는지 여부를 나타냅니다. (IWindowProvider.get_CanMinimize)
get_CanMove 컨트롤을 이동할 수 있는지 여부를 나타냅니다.
get_CanResize 컨트롤의 크기를 조정할 수 있는지 여부를 나타냅니다.
get_CanRotate 컨트롤을 회전할 수 있는지 여부를 나타냅니다.
get_CanSelectMultiple Microsoft UI 자동화 공급자가 둘 이상의 자식 요소를 동시에 선택할 수 있는지 여부를 나타냅니다.

get_CanZoom 컨트롤이 뷰포트의 확대/축소를 지원하는지 여부를 나타냅니다. (ITransformProvider2.get_CanZoom)
get_ChildCount 이 "and" 조건을 구성하는 조건 수를 검색합니다.
get_ChildCount 이 "or" 조건을 구성하는 조건 수를 검색합니다.
get_ChildId 이 요소의 자식 식별자를 지정합니다.
get_ClassName 프록시 팩터리에서 제공하는 창 클래스의 이름을 설정하거나 검색합니다. (Get)
get_CoalesceEvents 액세스 가능한 기술 클라이언트가 모든 이벤트를 수신하는지 또는 중복 이벤트가 검색되고 필터링되는 하위 집합을 가져오거나 설정합니다. (Get)
get_Column 이 셀 또는 항목이 포함된 열의 서수 수를 지정합니다.
get_ColumnCount 표에 있는 열의 총 수를 지정합니다.
get_ColumnSpan 이 셀 또는 항목에 걸쳐 있는 열 수를 지정합니다.
get_Condition UI 자동화 트리의 보기를 정의하는 조건을 검색합니다.
get_ConnectionRecoveryBehavior 공급자가 응답하지 않는 경우 액세스 가능한 기술 클라이언트가 공급자 요청 시간 제한을 조정하는지 여부를 나타냅니다. (Get)

get_ConnectionTimeout UI 자동화 공급자가 자동화 요소에 대한 클라이언트 요청에 응답할 때까지 대기하는 시간을 지정합니다. (Get)
get_ContainingGrid IGridProvider를 구현하고 이 셀 또는 항목의 컨테이너를 나타내는 UI 자동화 공급자를 지정합니다.
get_ContentViewCondition 콘텐츠 요소를 선택하는 미리 정의된 IUIAutomationCondition 인터페이스를 검색합니다.
get_ContentViewWalker 콘텐츠 요소를 검색하는 데 사용되는 IUIAutomationTreeWalker 인터페이스를 검색합니다.
get_ControlViewCondition 컨트롤 요소를 선택하는 미리 정의된 IUIAutomationCondition 인터페이스를 검색합니다.
get_ControlViewWalker 컨트롤 요소를 검색하는 데 사용되는 IUIAutomationTreeWalker 인터페이스를 검색합니다.
get_Count 프록시 팩터리 테이블의 항목 수를 검색합니다.
get_CurrentAcceleratorKey 요소의 액셀러레이터 키를 검색합니다.
get_CurrentAccessKey 요소의 액세스 키 문자를 검색합니다.
get_CurrentAnnotationObjects 주석, 머리글, 바닥글 등 이 요소와 연결된 주석 개체의 현재 목록을 반환합니다.
get_CurrentAnnotationTypeId 주석의 형식을 식별하는 값을 검색합니다.
get_CurrentAnnotationTypeName 이 주석 형식의 지역화된 이름을 검색합니다.

get_CurrentAnnotationTypes 주석, 머리글, 바닥글 등 이 요소와 연결된 주석 형식의 현재 목록을 반환합니다.
get_CurrentAriaProperties 요소의 ARIA(Access Rich Internet Applications) 속성을 검색합니다.
get_CurrentAriaRole 요소의 ARIA(Access Rich Internet Applications) 역할을 검색합니다.
get_CurrentAuthor 주석 작성자의 이름을 검색합니다.
get_CurrentAutomationId 요소의 Microsoft UI 자동화 식별자를 검색합니다.
get_CurrentBoundingRectangle 요소를 완전히 묶는 사각형의 좌표를 검색합니다.
get_CurrentCanMaximize 창을 최대화할 수 있는지 여부를 나타냅니다. (IUIAutomationWindowPattern.get_CurrentCanMaximize)
get_CurrentCanMinimize 창을 최소화할 수 있는지 여부를 나타냅니다. (IUIAutomationWindowPattern.get_CurrentCanMinimize)
get_CurrentCanMove 요소를 이동할 수 있는지 여부를 나타냅니다.
get_CurrentCanResize 요소 크기를 조정할 수 있는지 여부를 나타냅니다.
get_CurrentCanRotate 요소를 회전할 수 있는지 여부를 나타냅니다.
get_CurrentCanSelectMultiple 컨테이너에서 한 번에 둘 이상의 항목을 선택할 수 있는지 여부를 나타냅니다.

get_CurrentCanZoom 컨트롤이 뷰포트의 확대/축소를 지원하는지 여부를 나타냅니다. (IUIAutomationTransformPattern2.get_CurrentCanZoom)
get_CurrentChildId 요소에 대한 Microsoft Active Accessibility 자식 식별자를 검색합니다.
get_CurrentClassName 요소의 클래스 이름을 검색합니다.
get_CurrentColumn 항목이 포함된 열의 인덱스(0부터 시작하는 인덱스)를 검색합니다.
get_CurrentColumnCount 표의 열 수입니다.
get_CurrentColumnSpan 그리드 항목에 걸쳐 있는 열 수를 검색합니다.
get_CurrentContainingGrid 그리드 항목이 포함된 요소를 검색합니다.
get_CurrentControllerFor 이 요소가 컨트롤러 역할을 하는 요소의 배열을 검색합니다.
get_CurrentControlType 요소의 컨트롤 형식을 검색합니다.
get_CurrentCulture 요소의 문화권 식별자를 검색합니다.
get_CurrentCurrentSelectedItem 현재 선택한 항목을 나타내는 IUIAutomationElement 개체를 가져옵니다.
get_CurrentCurrentView 컨트롤의 현재 뷰에 대한 컨트롤별 식별자를 검색합니다.

get_CurrentDateTime 이 주석이 만들어진 날짜와 시간을 검색합니다.
get_CurrentDefaultAction 요소에 대한 Microsoft Active Accessibility 현재 기본 작업을 검색합니다.
get_CurrentDescribedBy 이 요소를 설명하는 요소의 배열을 검색합니다.
get_CurrentDescription 요소의 Microsoft 활성 접근성 설명을 검색합니다.
get_CurrentDockPosition 도킹 컨테이너 내에서 이 요소의 도킹 위치를 검색합니다.
get_CurrentDropEffect 사용자가 끌어서 놓기 작업의 일부로 이 요소를 삭제할 때 발생하는 작업을 나타내는 지역화된 문자열을 검색합니다.
get_CurrentDropEffects 끌어서 놓기 작업의 일부로 이 요소가 발생할 수 있는 전체 효과 집합을 열거하는 지역화된 문자열 배열을 검색합니다.
get_CurrentDropTargetEffect 사용자가 이 놓기 대상에서 잡은 요소를 삭제할 때 발생하는 작업을 설명하는 지역화된 문자열을 검색합니다.
get_CurrentDropTargetEffects 사용자가 끌어서 놓기 작업의 일부로 이 놓기 대상에서 끌어서 놓기 요소를 삭제할 때 발생할 수 있는 전체 효과 집합을 열거하는 지역화된 문자열 배열을 검색합니다. (IUIAutomationDropTargetPattern.get_CurrentDropTargetEffects)
get_CurrentExpandCollapseState 요소의 상태(확장 또는 축소됨)를 나타내는 값을 검색합니다.
get_CurrentExtendedProperties 문서의 요소에 대한 확장 속성 목록이 포함된 지역화된 문자열을 검색합니다.

get_CurrentFillColor 문서에서 요소의 채우기 색을 검색합니다.
get_CurrentFillPatternColor 문서의 요소를 채우는 데 사용되는 패턴의 색을 검색합니다.
get_CurrentFirstSelectedItem 선택한 항목 그룹의 첫 번째 항목을 나타내는 IUIAutomationElement 개체를 가져옵니다.
get_CurrentFlowsFrom 현재 요소 앞의 읽기 순서를 나타내는 요소 배열을 검색합니다.
get_CurrentFlowsTo 현재 요소 뒤의 읽기 순서를 나타내는 요소 배열을 검색합니다.
get_CurrentFormula 이 셀의 수식을 검색합니다.
get_CurrentFrameworkId 기본 UI 프레임워크의 이름을 검색합니다.
get_CurrentFullDescription 자동화 요소에 대한 현재 전체 설명을 가져옵니다.
get_CurrentHasKeyboardFocus 요소에 키보드 포커스가 있는지 여부를 나타냅니다.
get_CurrentHeadingLevel Automation 요소의 현재 제목 수준을 가져옵니다.
get_CurrentHelp 요소에 대한 Microsoft Active Accessibility 도움말 문자열을 검색합니다.
get_CurrentHelpText 요소의 도움말 텍스트를 검색합니다.

get_CurrentHorizontallyScrollable 요소가 가로로 스크롤할 수 있는지 여부를 나타냅니다.
get_CurrentHorizontalScrollPercent 가로 스크롤 위치를 검색합니다.
get_CurrentHorizontalViewSize 스크롤 가능한 요소의 볼 수 있는 영역의 가로 크기를 검색합니다.
get_CurrentIsContentElement 요소가 콘텐츠 요소인지 여부를 나타냅니다.
get_CurrentIsControlElement 요소가 컨트롤 요소인지 여부를 나타냅니다.
get_CurrentIsDataValidForForm 요소에 양식에 대한 유효한 데이터가 포함되어 있는지 여부를 나타냅니다.
get_CurrentIsDialog 요소에 대한 대화 상자 창 표시기인 현재를 검색합니다.
get_CurrentIsEnabled 요소를 사용할 수 있는지 여부를 나타냅니다.
get_CurrentIsGrabbed 사용자가 끌어서 놓기 작업의 일부로 이 요소를 잡았는지 여부를 나타냅니다.
get_CurrentIsKeyboardFocusable 요소가 키보드 포커스를 허용할 수 있는지 여부를 나타냅니다.
get_CurrentIsModal 창이 모달인지 여부를 나타냅니다. (IUIAutomationWindowPattern.get_CurrentIsModal)
get_CurrentIsOffscreen 요소가 화면 끄기인지 여부를 나타냅니다.

get_CurrentIsPassword 요소에 위장된 암호가 포함되어 있는지 여부를 나타냅니다.
get_CurrentIsPeripheral 요소의 현재 주변 UI 표시기를 검색합니다.
get_CurrentIsReadOnly 요소의 값을 변경할 수 있는지 여부를 나타냅니다.
get_CurrentIsReadOnly 요소 값이 읽기 전용인지 여부를 나타냅니다.
get_CurrentIsRequiredForForm 양식에서 요소를 작성해야 하는지 여부를 나타냅니다.
get_CurrentIsSelected 이 항목이 선택되었는지 여부를 나타냅니다.
get_CurrentIsSelectionRequired 항상 하나 이상의 항목을 선택해야 하는지 여부를 나타냅니다.
get_CurrentIsTopmost 창이 z 순서에서 맨 위 요소인지 여부를 나타냅니다. (IUIAutomationWindowPattern.get_CurrentIsTopmost)
get_CurrentItemCount 선택한 항목 수를 나타내는 정수 값을 가져옵니다.
get_CurrentItemStatus 요소의 항목 상태에 대한 설명을 검색합니다.
get_CurrentItemType 요소가 나타내는 UI 항목의 형식에 대한 설명을 검색합니다.
get_CurrentKeyboardShortcut 요소에 대한 Microsoft Active Accessibility 바로 가기 키 속성을 검색합니다.

get_CurrentLabeledBy 이 요소의 텍스트 레이블이 포함된 요소를 검색합니다.
get_CurrentLandmarkType Automation 요소의 현재 랜드마크 형식 ID를 가져옵니다.
get_CurrentLargeChange PAGE DOWN 키를 누를 때와 같이 큰 변경이 있을 때 컨트롤 값에 추가되거나 빼는 값을 검색합니다.
get_CurrentLastSelectedItem 선택한 항목 그룹의 마지막 항목을 나타내는 IUIAutomationElement 개체를 가져옵니다.
get_CurrentLevel 요소의 수준(계층 구조)에 대한 현재 1 기반 정수 값을 반환합니다.
get_CurrentLiveSetting 요소의 콘텐츠가 변경되면 요소가 보내는 알림 유형(있는 경우)을 나타냅니다.
get_CurrentLocalizedControlType 요소의 컨트롤 형식에 대한 지역화된 설명을 검색합니다.
get_CurrentLocalizedLandmarkType Automation 요소에 대한 현재 지역화된 랜드마크 형식이 포함된 문자열을 가져옵니다.
get_CurrentMaximum 컨트롤의 최대값을 검색합니다.
get_CurrentMinimum 컨트롤의 최소값을 검색합니다.
get_CurrentName 요소의 이름을 검색합니다.
get_CurrentName 요소의 Microsoft Active Accessibility 이름 속성을 검색합니다.

get_CurrentNativeWindowHandle 요소의 창 핸들을 검색합니다.
get_CurrentOptimizeForVisualContent 공급자가 표시되는 요소만 노출하는지 여부를 나타냅니다.
get_CurrentOrientation 요소의 방향을 나타내는 값을 검색합니다.
get_CurrentPositionInSet 요소 집합의 서수 위치에 대한 현재 1 기반 정수 값을 반환합니다.
get_CurrentProcessId 요소를 호스트하는 프로세스의 식별자를 검색합니다.
get_CurrentProviderDescription 이 요소의 공급자에 대한 설명을 검색합니다.
get_CurrentRole 요소의 Microsoft 활성 접근성 역할 식별자를 검색합니다.
get_CurrentRow 그리드 항목을 포함하는 행의 인덱스(0부터 시작하는 인덱스)를 검색합니다.
get_CurrentRowCount 표의 행 수를 검색합니다.
get_CurrentRowOrColumnMajor 테이블의 기본 탐색 방향을 검색합니다.
get_CurrentRowSpan 그리드 항목에 걸쳐 있는 행 수를 검색합니다.
get_CurrentSelectedItem 현재 선택한 항목을 가져옵니다.

get_CurrentSelectionContainer IUIAutomationSelectionPattern을 지원하는 요소를 검색하고 이 항목의 컨테이너 역할을 합니다.
get_CurrentShape 문서에서 요소의 모양을 검색합니다.
get_CurrentSizeOfSet 요소가 있는 집합의 크기에 대한 현재 1 기반 정수 값을 반환합니다.
get_CurrentSmallChange 화살표 키를 누를 때와 같이 약간 변경될 때 컨트롤 값에 추가되거나 빼는 값을 검색합니다.
get_CurrentState 요소에 대한 Microsoft Active Accessibility 상태 식별자를 검색합니다.
get_CurrentStyleId 문서의 요소와 연결된 비주얼 스타일의 식별자를 검색합니다.
get_CurrentStyleName 문서의 요소와 연결된 비주얼 스타일의 이름을 검색합니다.
get_CurrentTarget 주석이 추가되는 요소를 검색합니다.
get_CurrentToggleState 컨트롤의 상태를 검색합니다.
get_CurrentValue Microsoft Active Accessibility 값 속성을 검색합니다.
get_CurrentValue 컨트롤의 값을 검색합니다.
get_CurrentValue 요소의 값을 검색합니다.

get_CurrentVerticallyScrollable 요소가 세로로 스크롤할 수 있는지 여부를 나타냅니다.
get_CurrentVerticalScrollPercent 세로 스크롤 위치를 검색합니다.
get_CurrentVerticalViewSize 스크롤 가능한 요소의 볼 수 있는 영역의 세로 크기를 검색합니다.
get_CurrentView 컨트롤이 정보 또는 자식 컨트롤을 표시하는 데 사용하는 현재 보기를 식별합니다.
get_CurrentWindowInteractionState 사용자 상호 작용을 위해 창의 현재 상태를 검색합니다.
get_CurrentWindowVisualState 창의 시각적 상태를 검색합니다. 즉, 정상, 최대화 또는 최소화 상태인지 여부입니다.
get_CurrentZoomLevel 컨트롤 뷰포트의 확대/축소 수준을 검색합니다.
get_CurrentZoomMaximum 컨트롤 뷰포트의 최대 확대/축소 수준을 검색합니다.
get_CurrentZoomMinimum 컨트롤 뷰포트의 최소 확대/축소 수준을 검색합니다.
get_DateTime 이 주석을 만든 날짜와 시간입니다.
get_DefaultAction 이 요소의 기본 동작에 대한 설명을 포함합니다.
get_Description 이 요소에 대한 설명을 포함합니다.

get_DockPosition 이 요소의 현재 도킹 위치를 나타냅니다.
get_DocumentRange 문서의 주 텍스트를 둘러싸는 텍스트 범위를 검색합니다. (ITextProvider.get_DocumentRange)
get_DocumentRange 문서의 주 텍스트를 둘러싸는 텍스트 범위를 검색합니다. (IUIAutomationTextPattern.get_DocumentRange)
get_DropEffect 끌어서 놓기 작업의 일부로 이 요소가 삭제될 때 발생하는 작업을 나타내는 지역화된 문자열을 검색합니다.
get_DropEffects 끌어서 놓기 작업의 일부로 이 요소를 삭제할 때 발생할 수 있는 전체 효과 집합을 열거하는 지역화된 문자열 배열을 검색합니다.
get_DropTargetEffect 사용자가 이 놓기 대상에서 잡은 요소를 삭제할 때 발생하는 효과를 설명하는 지역화된 문자열을 검색합니다.
get_DropTargetEffects 사용자가 끌어서 놓기 작업의 일부로 이 놓기 대상에서 끌어서 놓기 요소를 삭제할 때 발생할 수 있는 전체 효과 집합을 열거하는 지역화된 문자열 배열을 검색합니다. (IDropTargetProvider.get_DropTargetEffects)
get_ExpandCollapseState 컨트롤의 확장 또는 축소 상태를 나타냅니다.
get_ExtendedProperties 이 컨트롤 패턴에 포함되지 않지만 사용자에게 유용할 수 있는 문서 콘텐츠에 대한 정보를 제공하는 추가 속성을 포함합니다.
get_FillColor 문서에 있는 요소의 채우기 색을 지정합니다.

get_FillPatternColor 문서의 요소를 채우는 데 사용되는 패턴의 색을 지정합니다.
get_FillPatternStyle 문서에서 요소의 채우기 패턴 스타일을 지정합니다.
get_FirstSelectedItem 선택한 항목 그룹의 첫 번째 항목을 가져옵니다.
get_Formula 이 스프레드시트 셀의 수식을 지정합니다.
get_FragmentRoot 조각의 루트 노드를 지정합니다.
get_Help 이 요소에 대한 도움말 정보를 포함하는 문자열을 지정합니다.
get_HorizontallyScrollable 컨트롤이 가로로 스크롤할 수 있는지 여부를 나타냅니다.
get_HorizontalScrollPercent 가로 스크롤 위치를 지정합니다.
get_HorizontalViewSize 볼 수 있는 영역의 가로 크기를 지정합니다.
get_HostRawElementProvider 이 요소의 호스트 공급자를 지정합니다.
get_ImageName 프록시 팩터리의 이미지 이름을 설정하거나 검색합니다. (Get)
get_IsGrabbed 요소를 끌어서 놓기 작업의 일부로 잡았는지 여부를 나타냅니다.

get_IsModal 창이 모달인지 여부를 나타냅니다. (IWindowProvider.get_IsModal)
get_IsReadOnly 컨트롤의 값이 읽기 전용인지 여부를 나타냅니다. (IRangeValueProvider.get_IsReadOnly)
get_IsReadOnly 컨트롤의 값이 읽기 전용인지 여부를 나타냅니다. (IValueProvider.get_IsReadOnly)
get_IsSelected 항목이 선택되었는지 여부를 나타냅니다.
get_IsSelectionRequired Microsoft UI 자동화 공급자가 하나 이상의 자식 요소를 선택해야 하는지 여부를 나타냅니다.
get_IsTopmost 창이 z 순서에서 맨 위 요소인지 여부를 나타냅니다. (IWindowProvider.get_IsTopmost)
get_ItemCount 선택한 항목의 수를 가져옵니다.
get_KeyboardShortcut 이 요소의 바로 가기 키를 지정합니다.
get_LargeChange PAGE DOWN 키를 누를 때와 같이 큰 변경이 있을 때 IRangeValueProvider::Value 속성에 추가되거나 뺄 값을 지정합니다.
get_LastSelectedItem 선택한 항목 그룹의 마지막 항목을 가져옵니다.
get_Length 컬렉션의 요소 수를 검색합니다. (IUIAutomationElementArray.get_Length)
get_Length 컬렉션의 텍스트 범위 수를 검색합니다.

get_Maximum 컨트롤에서 지원하는 최대 범위 값을 지정합니다.
get_Minimum 컨트롤에서 지원하는 최소 범위 값을 지정합니다.
get_Name 이 요소의 이름을 지정합니다.
get_NeedsAdviseEvents 애플리케이션이 이벤트를 등록할 때 프록시에 알림을 받아야 하는지 여부를 지정하는 값을 설정하거나 검색합니다. (Get)
get_PropertyConditionFlags 조건이 적용되는 방법을 지정하는 플래그 집합을 검색합니다.
get_PropertyId 이 조건의 기반이 되는 속성의 식별자를 검색합니다.
get_PropertyValue 조건이 true이면 일치해야 하는 속성 값을 검색합니다.
get_ProviderOptions Microsoft UI 자동화 공급자 유형(예: 클라이언트 쪽(프록시) 또는 서버 쪽 공급자)을 지정합니다.
get_ProxyFactory 이 항목과 연결된 프록시 팩터리를 검색합니다.
get_ProxyFactoryId 프록시 팩터리의 식별자를 검색합니다.
get_ProxyFactoryMapping Window 클래스 이름 및 연결된 데이터를 개별 프록시 팩터리에 매핑하는 것을 나타내는 개체를 검색합니다.

get_RawViewCondition 필터링되지 않은 보기에서 모든 UI 요소를 선택하는 미리 정의된 IUIAutomationCondition 인터페이스를 검색합니다.
get_RawViewWalker Microsoft UI 자동화 트리의 필터링되지 않은 뷰를 트래버스하는 데 사용되는 트리 워커 개체를 검색합니다.
get_ReservedMixedAttributeValue 혼합 특성인 텍스트 특성을 나타내는 정적 토큰 개체를 검색합니다.
get_ReservedNotSupportedValue 지원되지 않는 속성 또는 텍스트 특성을 나타내는 정적 토큰 개체를 검색합니다.
get_Role 이 요소의 역할 식별자를 지정합니다.
get_Row 이 셀 또는 항목을 포함하는 행의 서수 번호를 지정합니다.
get_RowCount 표에 있는 행의 총 수를 지정합니다.
get_RowOrColumnMajor 테이블에 대한 순회의 기본 방향을 지정합니다.
get_RowSpan 이 셀 또는 항목에 걸쳐 있는 행 수를 지정합니다.
get_SelectionContainer ISelectionProvider를 구현하고 호출 개체의 컨테이너 역할을 하는 공급자를 지정합니다.
get_Shape 문서의 요소 모양을 지정합니다.

get_SmallChange 화살표 키를 누를 때와 같이 작은 변경이 수행될 때 IRangeValueProvider::Value 속성에 추가되거나 빼는 값을 지정합니다.
get_State 이 요소의 상태를 지정합니다.
get_StyleId 문서에서 요소의 비주얼 스타일을 식별합니다.
get_StyleName 문서에 있는 요소의 비주얼 스타일 이름을 지정합니다.
get_SupportedTextSelection 컨트롤에서 지원하는 텍스트 선택 유형을 지정하는 값을 검색합니다. (ITextProvider.get_SupportedTextSelection)
get_SupportedTextSelection 컨트롤에서 지원하는 텍스트 선택 유형을 지정하는 값을 검색합니다. (IUIAutomationTextPattern.get_SupportedTextSelection)
get_Target 주석이 추가되는 UI 자동화 요소입니다.
get_TextContainer 텍스트 컨트롤 패턴을 지원하는 이 요소의 가장 가까운 상위 공급자를 검색합니다.
get_TextContainer Text 컨트롤 패턴을 지원하는 이 요소의 가장 가까운 상위 요소를 검색합니다.
get_TextRange 이 자식 요소를 묶는 텍스트 범위를 검색합니다. (ITextChildProvider.get_TextRange)
get_TextRange 이 자식 요소를 묶는 텍스트 범위를 검색합니다. (IUIAutomationTextChildPattern.get_TextRange)

get_ToggleState 컨트롤의 토글 상태를 지정합니다.
get_TransactionTimeout UI 자동화 자동화 요소에 대한 정보에 대한 클라이언트 요청에 공급자가 응답할 때까지 대기하는 시간을 지정합니다. (Get)
get_TreeFilter 캐싱할 때 사용되는 UI 자동화 요소 트리의 보기를 지정합니다. (Get)
get_TreeScope 캐싱 범위를 지정합니다. (Get)
get_Value 이 요소의 값을 지정합니다.
get_Value 컨트롤의 값을 지정합니다.
get_Value 컨트롤의 값입니다.
get_VerticallyScrollable 컨트롤이 세로로 스크롤할 수 있는지 여부를 나타냅니다.
get_VerticalScrollPercent 세로 스크롤 위치를 지정합니다.
get_VerticalViewSize 볼 수 있는 영역의 세로 크기를 지정합니다.
get_WindowInteractionState 사용자 상호 작용을 위해 창의 현재 상태를 지정합니다.
get_WindowVisualState 창의 시각적 상태를 지정합니다. 즉, 창이 정상(복원됨), 최소화 또는 최대화되었는지 여부입니다.

get_ZoomLevel 요소의 현재 확대/축소 수준을 검색합니다.
get_ZoomMaximum 요소의 최대 확대/축소 수준을 검색합니다.
get_ZoomMinimum 요소의 최소 확대/축소 수준을 검색합니다.
GetActiveComposition 활성 컴퍼지션을 반환합니다. (ITextEditProvider.GetActiveComposition)
GetActiveComposition 활성 컴퍼지션을 반환합니다. (IUIAutomationTextEditPattern.GetActiveComposition)
GetAdjacentFragment 이 컨트롤 사이트에서 소유한 창 없는 Microsoft ActiveX 컨트롤에 인접한 조각에 대한 조각 포인터를 검색합니다.
GetAnnotationObjects 이 스프레드시트 셀과 연결된 주석을 나타내는 개체 배열을 검색합니다.
GetAnnotationTypes 이 스프레드시트 셀과 연결된 주석의 형식을 나타내는 주석 형식 식별자의 배열을 검색합니다. (ISpreadsheetItemProvider.GetAnnotationTypes)
GetAttributeValue 텍스트 범위에서 지정된 텍스트 특성의 값을 검색합니다.
GetAttributeValue 전체 텍스트 범위에서 지정된 텍스트 특성의 값을 검색합니다.
GetAttributeValues 단일 프로세스 간 호출에서 텍스트 범위에 대해 요청된 모든 텍스트 특성 값을 반환합니다. 이는 GetAttributeValue를 호출하는 것과 동일합니다. 단 하나의 값이 아닌 여러 값을 검색할 수 있다는 점을 제외하면 됩니다.

GetBoundingRectangles 텍스트 범위에서 완전히 표시되거나 부분적으로 표시되는 각 텍스트 줄의 경계 사각형 컬렉션을 검색합니다. (ITextRangeProvider.GetBoundingRectangles)
GetBoundingRectangles 텍스트 범위에서 완전히 표시되거나 부분적으로 표시되는 각 텍스트 줄의 경계 사각형 컬렉션을 검색합니다. (IUIAutomationTextRange.GetBoundingRectangles)
GetBuildVersion 클라이언트는 IVersionInfo::GetBuildVersion을 호출하여 지정된 구성 요소에 대한 빌드 정보를 검색합니다.
GetCachedAnnotationObjects 이 스프레드시트 셀과 연결된 주석을 나타내는 캐시된 요소 배열을 검색합니다.
GetCachedAnnotationTypes 이 스프레드시트 셀과 연결된 주석의 형식을 나타내는 주석 형식 식별자의 캐시된 배열을 검색합니다.
GetCachedChildren 이 UI 자동화 요소의 캐시된 자식 요소를 검색합니다.
GetCachedColumnHeaderItems 테이블 항목 또는 셀과 연결된 캐시된 열 헤더를 검색합니다.
GetCachedColumnHeaders 테이블의 모든 열 머리글을 나타내는 캐시된 UI 자동화 요소 컬렉션을 검색합니다.
GetCachedGrabbedItems 사용자가 끌기 작업의 일부로 끌 항목의 전체 집합을 나타내는 캐시된 요소 컬렉션을 검색합니다.
GetCachedParent 캐시에서 이 UI 자동화 요소의 부모를 검색합니다.
GetCachedPattern 캐시에서 이 UI 자동화 요소의 지정된 컨트롤 패턴에 대한 IUnknown 인터페이스를 검색합니다.

GetCachedPatternAs 이 UI 자동화 요소의 캐시에서 지정된 패턴의 컨트롤 패턴 인터페이스를 검색합니다.
GetCachedPropertyValue 이 UI 자동화 요소의 캐시에서 속성 값을 검색합니다.
GetCachedPropertyValueEx 필요에 따라 기본값을 무시하고 이 UI 자동화 요소의 캐시에서 속성 값을 검색합니다.
GetCachedRowHeaderItems 테이블 항목 또는 셀과 연결된 캐시된 행 헤더를 검색합니다.
GetCachedRowHeaders 테이블의 모든 행 헤더를 나타내는 캐시된 UI 자동화 요소 컬렉션을 검색합니다.
GetCachedSelection 이 요소의 선택한 자식을 식별하는 캐시된 Microsoft Active Accessibility 속성을 검색합니다.
GetCachedSelection 컨테이너에서 캐시된 선택한 요소를 검색합니다.
GetCachedSupportedViews 캐시에서 컨트롤별 뷰 식별자 컬렉션을 검색합니다.
GetCaretRange 텍스트 기반 컨트롤에 속하는 caret의 위치에 길이가 0인 텍스트 범위를 제공합니다.
GetCaretRange 텍스트 기반 컨트롤에 속하는 caret의 위치에서 길이가 0인 텍스트 범위를 검색합니다.
Getchild 이 조건이 음수인 조건을 검색합니다.
GetChildren 텍스트 범위 안에 있는 모든 포함된 개체의 컬렉션을 검색합니다. (ITextRangeProvider.GetChildren)

GetChildren 이 "and" 조건을 구성하는 조건을 검색합니다.
GetChildren 이 "or" 조건을 구성하는 조건을 검색합니다.
GetChildren 텍스트 범위 안에 있는 모든 포함된 개체의 컬렉션을 검색합니다. (IUIAutomationTextRange.GetChildren)
GetChildrenAsNativeArray 이 "and" 조건을 구성하는 조건을 일반 배열로 검색합니다.
GetChildrenAsNativeArray 이 "or" 조건을 구성하는 조건을 일반 배열로 검색합니다.
GetChildrenBuildCache 단일 교차 프로세스 호출에서 텍스트 범위의 요소에 대한 자식 및 제공 속성 및 패턴을 반환합니다. 이는 GetChildren를 호출하는 것과 동일하지만 표준 빌드 캐시 패턴을 추가합니다.
GetClickablePoint 클릭할 수 있는 요소의 점을 검색합니다.
GetColumnHeaderItems 테이블 항목 또는 셀과 연결된 모든 열 머리글을 나타내는 Microsoft UI 자동화 공급자의 컬렉션을 검색합니다.
GetColumnHeaders 테이블의 모든 열 머리글을 나타내는 Microsoft UI 자동화 공급자의 컬렉션을 가져옵니다.
GetComponentDescription 클라이언트는 이 메서드를 호출하여 구성 요소에 대한 설명을 검색합니다.
GetConversionTarget 현재 변환 대상 범위를 반환합니다. (ITextEditProvider.GetConversionTarget)

GetConversionTarget 현재 변환 대상 범위를 반환합니다. (IUIAutomationTextEditPattern.GetConversionTarget)
GetCurrentAnnotationObjects 이 스프레드시트 셀과 연결된 주석을 나타내는 요소 배열을 검색합니다.
GetCurrentAnnotationTypes 이 스프레드시트 셀과 연결된 주석의 형식을 나타내는 주석 형식 식별자의 배열을 검색합니다. (IUIAutomationSpreadsheetItemPattern.GetCurrentAnnotationTypes)
GetCurrentColumnHeaderItems 테이블 항목 또는 셀과 연결된 열 머리글을 검색합니다.
GetCurrentColumnHeaders 테이블의 모든 열 머리글을 나타내는 UI 자동화 요소의 컬렉션을 검색합니다.
GetCurrentGrabbedItems 사용자가 끌기 작업의 일부로 끌고 있는 항목의 전체 집합을 나타내는 요소 컬렉션을 검색합니다.
GetCurrentMetadataValue 정보를 해석하는 방법을 나타내는 UI 자동화 요소에서 메타데이터를 가져옵니다. (IUIAutomationElement7.GetCurrentMetadataValue)
GetCurrentPattern 이 UI 자동화 요소에서 지정된 컨트롤 패턴의 IUnknown 인터페이스를 검색합니다.
GetCurrentPatternAs 이 UI 자동화 요소에서 지정된 패턴의 컨트롤 패턴 인터페이스를 검색합니다.
GetCurrentPropertyValue 이 UI 자동화 요소에 대한 속성의 현재 값을 검색합니다.
GetCurrentPropertyValueEx 필요에 따라 기본값을 무시하고 이 UI 자동화 요소의 속성 값을 검색합니다.
GetCurrentRowHeaderItems 테이블 항목 또는 셀과 연결된 행 머리글을 검색합니다.

GetCurrentRowHeaders 테이블의 모든 행 머리글을 나타내는 UI 자동화 요소의 컬렉션을 검색합니다.
GetCurrentSelection 이 요소의 선택한 자식을 식별하는 Microsoft Active Accessibility 속성을 검색합니다.
GetCurrentSelection 컨테이너에서 선택한 요소를 검색합니다.
GetCurrentSupportedViews 컨트롤별 보기 식별자의 컬렉션을 검색합니다. (IUIAutomationMultipleViewPattern.GetCurrentSupportedViews)
GetDocuments 클라이언트는 IAccClientDocMgr::GetDocuments를 호출하여 Microsoft Active Accessibility 런타임에 등록된 모든 문서 목록을 가져옵니다.
GetElement 컬렉션에서 Microsoft UI 자동화 요소를 검색합니다.
GetElement 컬렉션에서 텍스트 범위를 검색합니다.
GetEmbeddedAccessibles 이 공급자가 호스트하는 창 없는 Microsoft ActiveX 컨트롤의 IAccessible 인터페이스 포인터를 검색합니다.
GetEmbeddedFragmentRoots Microsoft UI 자동화 공급자 구현이 있고 IAccessibleHostingElementProviders 인터페이스를 구현하는 Microsoft Active Accessibility 개체에서 호스트되는 모든 창 없는 Microsoft ActiveX 컨트롤의 Microsoft Active Accessibility 공급자를 검색합니다.
GetEmbeddedFragmentRoots 현재 요소에 루트된 Microsoft UI 자동화 트리에 포함된 루트 조각의 배열을 검색합니다.
GetEnclosingElement 텍스트 범위를 둘러싸는 가장 안쪽 요소를 반환합니다.

GetEnclosingElement 텍스트 범위를 둘러싸는 가장 안쪽의 UI 자동화 요소를 반환합니다.
GetEnclosingElementBuildCache 단일 프로세스 간 호출에서 텍스트 범위의 요소에 대한 바깥쪽 요소와 제공된 속성 및 패턴을 가져옵니다. GetEnclosingElement를 호출하는 것과 동일하지만 표준 빌드 캐시 패턴을 추가합니다.
GetEntry 프록시 팩터리 테이블에서 항목을 검색합니다.
GetFirstChildElement 지정된 UI 자동화 요소의 첫 번째 자식 요소를 검색합니다.
GetFirstChildElementBuildCache 지정된 UI 자동화 요소의 첫 번째 자식 요소를 검색하고 속성 및 컨트롤 패턴을 캐시합니다.
GetFocus 이 조각에서 입력 포커스가 있는 요소를 검색합니다.
GetFocused 클라이언트는 IAccClientDocMgr::GetFocused 메서드를 호출하여 포커스가 있는 문서에 대한 포인터에 액세스합니다.
GetFocusedElement 입력 포커스가 있는 UI 자동화 요소를 검색합니다.
GetFocusedElementBuildCache 입력 포커스가 있는 UI 자동화 요소를 검색하고, 요청된 속성 및 컨트롤 패턴을 프리페치하고, 프리페치된 항목을 캐시에 저장합니다.
GetGrabbedItems 끌기 작업의 일부로 끌 요소의 컬렉션을 검색합니다.
GetIAccessible LegacyIAccessible 컨트롤 패턴을 지원하는 UI 자동화 요소에 해당하는 액세스 가능한 개체를 검색합니다.

GetIAccessible Microsoft UI 자동화 요소에 해당하는 IAccessible 개체를 검색합니다.
GetIAccessiblePair 이 항목에 대한 IAccessible 인터페이스 및 자식 ID를 검색합니다.
GetIdentityString 액세스 가능한 요소를 고유하게 식별하는 바이트 문자열(ID 문자열)을 검색합니다.
GetImplementationID 클라이언트는 IVersionInfo::GetImplementationID를 호출하여 구성 요소에 대한 고유 식별자를 검색합니다.
GetInstanceDescription 클라이언트는 인스턴스에 대한 설명을 검색하기 위해 이 메서드를 호출합니다. 활성 접근성 Text Services는 더 이상 사용되지 않습니다.
GetItem 지정된 셀에 대한 Microsoft UI 자동화 공급자를 검색합니다.
GetItem 그리드의 항목을 나타내는 UI 자동화 요소를 검색합니다.
GetItemByName 지정한 이름을 가진 스프레드시트 셀을 나타내는 UI 자동화 요소를 노출합니다.
GetItemByName 지정한 이름을 가진 스프레드시트 셀을 나타내는 UI 자동화 요소를 검색합니다.
GetLastChildElement 지정된 UI 자동화 요소의 마지막 자식 요소를 검색합니다.
GetLastChildElementBuildCache 지정된 UI 자동화 요소의 마지막 자식 요소를 검색하고 속성 및 컨트롤 패턴을 캐시합니다.

GetLocalizedString 클라이언트는 IAccDictionary::GetLocalizedString 메서드를 호출하여 모든 시스템 속성 및 해당 값에 대해 지역화된 문자열을 가져옵니다.
GetMetadataValue 정보를 해석하는 방법을 나타내는 UI 자동화 요소에서 메타데이터를 가져옵니다. (IRawElementProviderSimple3.GetMetadataValue)
GetMnemonicString 니모닉 문자열을 검색합니다. 활성 접근성 Text Services는 더 이상 사용되지 않습니다.
GetNextSiblingElement 지정된 UI 자동화 요소의 다음 형제 요소를 검색합니다.
GetNextSiblingElementBuildCache 지정된 UI 자동화 요소의 다음 형제 요소를 검색하고 속성 및 컨트롤 패턴을 캐시합니다.
GetObjectForChild 이 요소의 지정된 자식을 나타내는 IAccessibleEx 인터페이스를 검색합니다.
GetObjectIdForProvider Microsoft UI 자동화 구현하는 포함된 창 없는 Microsoft ActiveX 컨트롤과 연결된 개체 ID를 검색합니다.
GetOleaccVersionInfo Microsoft Active Accessibility 파일 Oleacc.dll 버전 번호 및 빌드 번호를 검색합니다.
GetOverrideProviderForHwnd 지정된 요소에 대한 UI 자동화 공급자를 가져옵니다.
GetParentAccessible 접근성 트리에서 창 없는 Microsoft ActiveX 컨트롤의 부모에 대한 IAccessible 포인터를 검색합니다.
GetParentElement 지정된 UI 자동화 요소의 부모 요소를 검색합니다.

GetParentElementBuildCache 지정된 UI 자동화 요소의 부모 요소를 검색하고 속성 및 컨트롤 패턴을 캐시합니다.
GetParentTerm 클라이언트는 IAccDictionary::GetParentTerm 메서드를 호출하여 개체 계층 트리를 탐색합니다. 이 메서드는 지정된 속성의 부모 개체를 반환합니다.
GetPatternProgrammaticName 컨트롤 패턴의 등록된 프로그래밍 이름을 검색합니다.
GetPatternProvider Microsoft UI 자동화 요소에서 컨트롤 패턴을 지원하는 개체에 대한 포인터를 검색합니다.
GetPreviousSiblingElement 지정된 UI 자동화 요소의 이전 형제 요소를 검색합니다.
GetPreviousSiblingElementBuildCache 지정된 UI 자동화 요소의 이전 형제 요소를 검색하고 속성 및 컨트롤 패턴을 캐시합니다.
Getproperty 클라이언트 래퍼 개체는 이 함수를 호출하고 인덱스별로 속성을 지정하여 IUIAutomation::get_CurrentX 및 IUIAutomationElement::get_CachedX 메서드를 구현합니다.
GetPropertyProgrammaticName 속성의 등록된 프로그래밍 이름을 검색합니다.
GetPropertyValue Microsoft UI 자동화 공급자가 지원하는 속성의 값을 검색합니다.
GetPropValue 액세스 가능한 요소의 속성 값을 검색합니다.
GetRoleTextA 지정된 역할 값에 대한 개체의 역할을 설명하는 지역화된 문자열을 검색합니다. (ANSI)
GetRoleTextW 지정된 역할 값에 대한 개체의 역할을 설명하는 지역화된 문자열을 검색합니다. (유니코드)

GetRootElement 바탕 화면을 나타내는 UI 자동화 요소를 검색합니다.
GetRootElementBuildCache 데스크톱을 나타내는 UI 자동화 요소를 검색하고, 요청된 속성 및 컨트롤 패턴을 프리페치하고, 프리페치된 항목을 캐시에 저장합니다.
GetRowHeaderItems 테이블 항목 또는 셀과 연결된 모든 행 머리글을 나타내는 Microsoft UI 자동화 공급자의 컬렉션을 검색합니다.
GetRowHeaders 테이블의 모든 행 머리글을 나타내는 Microsoft UI 자동화 공급자의 컬렉션을 가져옵니다.
GetRuntimeId 이 요소의 런타임 식별자를 검색합니다.
GetRuntimeId 요소의 런타임 식별자를 검색합니다.
GetRuntimeId UI 요소에 할당된 고유 식별자를 검색합니다.
GetRuntimeIdPrefix 창 없는 Microsoft ActiveX 컨트롤 사이트에 고유한 Microsoft UI 자동화 런타임 ID를 검색합니다.
GetSelection 컨트롤에서 선택한 항목 또는 항목을 검색합니다.
GetSelection 선택한 각 자식 요소에 대한 Microsoft UI 자동화 공급자를 검색합니다.
GetSelection 텍스트 기반 컨트롤에서 현재 선택한 텍스트를 나타내는 텍스트 범위의 컬렉션을 검색합니다. (ITextProvider.GetSelection)

GetSelection 텍스트 기반 컨트롤에서 현재 선택한 텍스트를 나타내는 텍스트 범위의 컬렉션을 검색합니다. (IUIAutomationTextPattern.GetSelection)
GetStateTextA 미리 정의된 단일 상태 비트 플래그에 대한 개체의 상태를 설명하는 지역화된 문자열을 검색합니다. 상태 값은 하나 이상의 비트 플래그의 조합이므로 클라이언트는 이 함수를 두 번 이상 호출하여 모든 상태 문자열을 검색합니다. (ANSI)
GetStateTextW 미리 정의된 단일 상태 비트 플래그에 대한 개체의 상태를 설명하는 지역화된 문자열을 검색합니다. 상태 값은 하나 이상의 비트 플래그의 조합이므로 클라이언트는 이 함수를 두 번 이상 호출하여 모든 상태 문자열을 검색합니다. (유니코드)
GetSubcomponentCount 클라이언트는 IVersionInfo::GetSubcomponentCount를 호출하여 버전 정보가 반환되는 하위 구성 요소의 수를 결정합니다.
GetSupportedViews 컨트롤별 보기 식별자의 컬렉션을 검색합니다. (IMultipleViewProvider.GetSupportedViews)
Gettable 프록시 팩터리 테이블의 모든 항목을 검색합니다.
GetText 범위의 일반 텍스트를 검색합니다.
GetText 텍스트 범위의 일반 텍스트를 반환합니다.
GetUnderlyingObjectModel 공급자의 기본 개체 모델에 액세스하는 데 사용되는 인터페이스를 검색합니다. (IObjectModelProvider.GetUnderlyingObjectModel)
GetUnderlyingObjectModel 공급자의 기본 개체 모델에 액세스하는 데 사용되는 인터페이스를 검색합니다. (IUIAutomationObjectModelPattern.GetUnderlyingObjectModel)

GetViewName 컨트롤별 뷰 이름을 검색합니다. (IMultipleViewProvider.GetViewName)
GetViewName 컨트롤별 뷰 이름을 검색합니다. (IUIAutomationMultipleViewPattern.GetViewName)
GetVisibleRanges 각 텍스트 범위가 표시되는 텍스트의 연속된 범위를 나타내는 텍스트 기반 컨트롤에서 연결이 끊긴 텍스트 범위의 배열을 검색합니다. (ITextProvider.GetVisibleRanges)
GetVisibleRanges 각 텍스트 범위가 표시되는 텍스트의 연속된 범위를 나타내는 텍스트 기반 컨트롤에서 연결이 끊긴 텍스트 범위의 배열을 검색합니다. (IUIAutomationTextPattern.GetVisibleRanges)
GetWinEventsForAutomationEvent 특정 Microsoft UI 자동화 이벤트에 매핑되는 WinEvents 목록을 검색합니다. 이 프록시가 나타내는 요소가 나열된 WinEvents 중 하나를 발생하면 프록시가 이를 처리합니다.
GridPattern_GetItem 그리드의 항목에 대한 노드를 가져옵니다.
HandleActiveTextPositionChangedEvent Microsoft UI 자동화 활성 텍스트 위치 변경 이벤트를 처리합니다.
HandleAutomationEvent Microsoft UI 자동화 이벤트를 처리합니다.
HandleChangesEvent 하나 이상의 Microsoft UI 자동화 변경 이벤트를 처리합니다.
HandleFocusChangedEvent 키보드 포커스가 다른 UI 자동화 요소로 이동할 때 발생하는 이벤트를 처리합니다.
HandleNotificationEvent Microsoft UI 자동화 알림 이벤트를 처리합니다.

HandlePropertyChangedEvent Microsoft UI 자동화 속성 변경 이벤트를 처리합니다.
HandleStructureChangedEvent Microsoft UI 자동화 트리 구조가 변경될 때 발생하는 이벤트를 처리합니다.
HandleTextEditTextChangedEvent 텍스트 편집 컨트롤에 대한 Microsoft UI 자동화 공급자가 프로그래밍 방식 텍스트 변경을 보고할 때 발생하는 이벤트를 처리합니다.
InsertEntries 프록시 팩터리 테이블에 항목을 삽입합니다.
InsertEntry 프록시 팩터리 테이블에 항목을 삽입합니다.
IntNativeArrayToSafeArray 정수 배열을 SAFEARRAY로 변환합니다.
IntSafeArrayToNativeArray 정수의 SAFEARRAY를 배열로 변환합니다.
Invoke 요청을 보내 컨트롤을 활성화하고 해당하는 명확한 단일 작업을 시작합니다. (IInvokeProvider.Invoke)
Invoke 단추 클릭과 같은 컨트롤의 동작을 호출합니다.
InvokePattern_Invoke 요청을 보내 컨트롤을 활성화하고 해당하는 명확한 단일 작업을 시작합니다. (InvokePattern_Invoke)
IsWinEventHookInstalled 지정된 이벤트에 대한 알림을 받을 수 있는 설치된 WinEvent 후크가 있는지 여부를 확인합니다.

ItemContainerPattern_FindItemByProperty 지정된 속성 값을 기반으로 포함하는 노드 내의 노드를 검색합니다.
LegacyIAccessiblePattern_DoDefaultAction 요소에 대한 Microsoft Active Accessibility 기본 작업을 수행합니다. (LegacyIAccessiblePattern_DoDefaultAction)
LegacyIAccessiblePattern_GetIAccessible UI 자동화 요소에 해당하는 IAccessible 개체를 검색합니다.
LegacyIAccessiblePattern_Select Microsoft 활성 접근성 선택을 수행합니다. (LegacyIAccessiblePattern_Select)
LegacyIAccessiblePattern_SetValue 노드에 대한 Microsoft Active Accessibility 값 속성을 설정합니다.
LocalInit 로컬 개체에 대한 정보를 반환하기 위해 클라이언트에서 구현합니다. 활성 접근성 Text Services는 더 이상 사용되지 않습니다.
LookupByHWND 클라이언트는 IAccClientDocMgr::LookupByHWND를 호출하여 문서에 대한 HWND를 제공하여 문서를 가져옵니다.
LookupByPoint 클라이언트는 IAccClientDocMgr::LookupByPoint를 호출하여 문서 내의 지점에서 문서 개체를 가져옵니다.
LookupMnemonicTerm 클라이언트는 IAccDictionary::LookupMnemonicTerm 메서드를 호출하여 지정된 니모닉 문자열에 대한 속성을 찾습니다.
LresultFromObject 핸들과 유사한 참조를 지정된 개체에 반환합니다. 서버는 WM_GETOBJECT 처리할 때 이 참조를 반환합니다.
이동 텍스트 범위를 지정한 텍스트 단위 수만큼 앞으로 또는 뒤로 이동합니다.

이동 컨트롤을 이동합니다.
이동 텍스트 범위를 지정한 텍스트 단위 수만큼 앞으로 또는 뒤로 이동합니다.
이동 UI 자동화 요소를 이동합니다.
MoveEndpointByRange 현재 텍스트 범위의 한 엔드포인트를 두 번째 텍스트 범위의 지정된 엔드포인트로 이동합니다. (ITextRangeProvider.MoveEndpointByRange)
MoveEndpointByRange 현재 텍스트 범위의 한 엔드포인트를 두 번째 텍스트 범위의 지정된 엔드포인트로 이동합니다. (IUIAutomationTextRange.MoveEndpointByRange)
MoveEndpointByUnit 텍스트 범위의 한 엔드포인트를 문서 범위 내에서 지정된 수의 TextUnit 단위로 이동합니다.
MoveEndpointByUnit 텍스트 범위의 한 엔드포인트을 문서 범위 내에서 지정된 텍스트 단위 수만큼 이동합니다.
MultipleViewPattern_GetViewName 컨트롤별 뷰 이름을 검색합니다. (MultipleViewPattern_GetViewName)
MultipleViewPattern_SetCurrentView 컨트롤을 다른 레이아웃으로 설정합니다.
탐색 UI 자동화 트리 내에서 지정된 방향으로 Microsoft UI 자동화 요소를 검색합니다.
탐색 논리 UI 트리 내에서 지정된 방향으로 다음 요소를 가져옵니다.

NewDocument 서버 애플리케이션은 IAccServerDocMgr::NewDocument 메서드를 사용할 수 있을 때 호출합니다. 어댑터는 래핑된 문서를 만들고 저장소에 등록하므로 클라이언트는 문서의 텍스트에 대한 정보에 액세스할 수 있습니다.
NormalizeElement 트리 뷰에서 지정된 Microsoft UI 자동화 요소에 가장 가까운 상위 요소를 검색합니다.
NormalizeElementBuildCache 트리 뷰에서 지정된 Microsoft UI 자동화 요소에 가장 가까운 상위 요소를 검색하고, 요청된 속성 및 컨트롤 패턴을 프리페치하고, 프리페치된 항목을 캐시에 저장합니다.
NotifyWinEvent 미리 정의된 이벤트가 발생한 시스템에 신호를 보냅니다. 클라이언트 애플리케이션이 이벤트에 대한 후크 함수를 등록한 경우 시스템은 클라이언트의 후크 함수를 호출합니다.
ObjectFromLresult 이전에 생성된 개체 참조를 기반으로 액세스 가능한 개체에 대해 요청된 인터페이스 포인터를 검색합니다.
OnDocumentFocus Text Services Framework를 사용하는 애플리케이션은 IAccServerDocMgr::OnDocumentFocus를 호출하여 문서가 포커스를 가져오거나 잃을 때 Microsoft Active Accessibility 런타임에 알립니다.
PollForPotentialSupportedPatterns UI 자동화 요소에서 지원될 수 있는 컨트롤 패턴을 검색합니다.
PollForPotentialSupportedProperties UI 자동화 요소에서 지원될 수 있는 속성을 검색합니다.
put_accName IAccessible::p ut_accName 메서드는 더 이상 지원되지 않습니다. 클라이언트 애플리케이션은 SetWindowText 함수와 같은 컨트롤 관련 해결 방법을 사용해야 합니다. 서버는 E_NOTIMPL 반환해야 합니다.
put_accValue IAccessible::p ut_accValue 메서드는 지정된 개체의 값을 설정합니다. 모든 개체에 값이 있는 것은 아닙니다.

put_AllowSubstringMatch 프록시에서 부분 문자열 일치를 허용하는지 여부를 지정하는 값을 설정하거나 검색합니다. (Put)
put_AutomationElementMode 반환된 요소에 기본 UI에 대한 전체 참조가 포함되어 있는지 또는 캐시된 정보만 포함되는지 여부를 나타냅니다. (Put)
put_AutoSetFocus UI 자동화 컨트롤 패턴 메서드에 대한 호출이 자동으로 대상 요소에 포커스를 설정할지 여부를 지정합니다. (Put)
put_CanCheckBaseClass 프록시 팩터리를 검색할 때 기본 클래스를 확인할 수 있는지 여부를 지정하는 값을 설정하거나 검색합니다. (Put)
put_ClassName 프록시 팩터리에서 제공하는 창 클래스의 이름을 설정하거나 검색합니다. (Put)
put_CoalesceEvents 액세스 가능한 기술 클라이언트가 모든 이벤트를 수신하는지 또는 중복 이벤트가 검색되고 필터링되는 하위 집합을 가져오거나 설정합니다. (Put)
put_ConnectionRecoveryBehavior 공급자가 응답하지 않는 경우 액세스 가능한 기술 클라이언트가 공급자 요청 시간 제한을 조정하는지 여부를 나타냅니다. (Put)
put_ConnectionTimeout UI 자동화 공급자가 자동화 요소에 대한 클라이언트 요청에 응답할 때까지 대기하는 시간을 지정합니다. (Put)
put_ImageName 프록시 팩터리의 이미지 이름을 설정하거나 검색합니다. (Put)
put_NeedsAdviseEvents 애플리케이션이 이벤트를 등록할 때 프록시에 알림을 받아야 하는지 여부를 지정하는 값을 설정하거나 검색합니다. (Put)

put_TransactionTimeout UI 자동화 자동화 요소에 대한 정보를 위해 공급자가 클라이언트 요청에 응답할 때까지 대기하는 시간을 지정합니다. (Put)
put_TreeFilter 캐싱할 때 사용되는 UI 자동화 요소 트리의 보기를 지정합니다. (Put)
put_TreeScope 캐싱 범위를 지정합니다. (Put)
QueryObjectIdRanges 특정 창 없는 Microsoft ActiveX 컨트롤이 예약한 개체 ID 범위를 검색합니다.
RangeFromAnnotation 지정된 주석 요소와 연결된 주석의 대상인 텍스트가 포함된 텍스트 범위를 노출합니다.
RangeFromAnnotation 지정된 주석 요소와 연결된 주석의 대상인 텍스트가 포함된 텍스트 범위를 검색합니다.
RangeFromChild 이미지, 하이퍼링크, 기타 포함된 개체 등의 자식 요소를 둘러싸는 텍스트 범위를 검색합니다.
RangeFromChild 이미지, 하이퍼링크, Microsoft Excel 스프레드시트 또는 기타 포함된 개체와 같은 자식 요소를 포함하는 텍스트 범위를 검색합니다.
RangeFromPoint 지정된 화면 좌표와 가장 가까운 중복 제거 텍스트 범위(빈 범위)를 반환합니다.
RangeFromPoint 지정된 화면 좌표에 가장 가까운 퇴화(빈) 텍스트 범위를 검색합니다. (IUIAutomationTextPattern.RangeFromPoint)
RangeValuePattern_SetValue 숫자 범위가 있는 컨트롤의 값을 설정합니다.

실현 가상 항목을 UI 자동화 요소로 완전히 액세스할 수 있도록 만듭니다. (IVirtualizedItemProvider.Realize)
실현 가상화된 항목에 대한 전체 UI 자동화 요소를 만듭니다.
RectToVariant 사각형의 좌표를 포함하는 VARIANT를 만듭니다.
RegisterEvent 타사 Microsoft UI 자동화 이벤트를 등록합니다.
RegisterPattern 타사 컨트롤 패턴을 등록합니다.
RegisterPointerInputTarget 호출자가 지정된 형식의 모든 포인터 입력이 리디렉션되는 대상 창을 등록할 수 있습니다.
RegisterPointerInputTargetEx RegisterPointerInputTargetEx를 변경하거나 사용할 수 없습니다. 대신 RegisterPointerInputTarget을 사용합니다.
RegisterProperty 타사 속성을 등록합니다.
ReleaseObjectIdRange IAccessibleWindowlessSite::AcquireObjectIdRange 메서드에 대한 이전 호출로 얻은 개체 ID 범위를 해제합니다.
RemoveActiveTextPositionChangedEventHandler 활성 텍스트 위치 변경 이벤트 처리기를 제거합니다.
RemoveAllEventHandlers 등록된 모든 Microsoft UI 자동화 이벤트 처리기를 제거합니다.

RemoveAutomationEventHandler 지정된 UI 자동화 이벤트 처리기를 제거합니다.
RemoveChangesEventHandler 변경 이벤트 처리기를 제거합니다.
RemoveEntry 프록시 팩터리 테이블에서 항목을 제거합니다.
RemoveEventHandlerGroup 지정된 UI 자동화 이벤트 처리기 그룹을 비동기적으로 제거합니다.
RemoveFocusChangedEventHandler 포커스가 변경된 이벤트 처리기를 제거합니다.
RemoveFromSelection 선택한 항목 컬렉션에서 현재 요소를 제거합니다.
RemoveFromSelection 선택한 텍스트의 연결되지 않은 여러 범위를 지원하는 컨트롤의 선택한 텍스트 범위 컬렉션에서 텍스트 범위를 제거합니다.
RemoveFromSelection 선택 영역에서 이 요소를 제거합니다.
RemoveFromSelection 여러 개의 연결되지 않은 선택을 지원하는 텍스트 컨테이너에서 선택한 텍스트의 기존 컬렉션에서 텍스트 범위를 제거합니다.
RemoveNotificationEventHandler 알림 이벤트 처리기를 제거합니다.
RemovePropertyChangedEventHandler 속성 변경 이벤트 처리기를 제거합니다.
RemoveStructureChangedEventHandler 구조 변경 이벤트 처리기를 제거합니다.

RemoveTextEditTextChangedEventHandler 프로그래밍 방식 텍스트 편집 이벤트 처리기를 제거합니다.
크기 조정 컨트롤의 크기를 조정합니다.
크기 조정 UI 자동화 요소의 크기를 조정합니다.
RespondToWinEvent WinEvent를 처리합니다.
RestoreDefaultTable 프록시 팩터리 기본 테이블을 복원합니다.
RevokeDocument 서버 애플리케이션은 IAccServerDocMgr::RevokeDocument 메서드를 호출하여 Microsoft Active Accessibility 런타임에 문서를 더 이상 사용할 수 없음을 알립니다. RevokeDocument를 호출하면 클라이언트가 문서를 볼 수 없도록 저장소에서 제거됩니다.
회전 컨트롤을 회전합니다.
회전 UI 자동화 요소를 회전합니다.
SafeArrayToRectNativeArray 사각형 좌표를 포함하는 SAFEARRAY를 RECT 형식의 배열로 변환합니다.
스크롤 콘텐츠 영역의 표시 영역을 가로 및 세로로 스크롤합니다. (IScrollProvider.Scroll)
스크롤 콘텐츠 영역의 표시 영역을 가로 및 세로로 스크롤합니다. (IUIAutomationScrollPattern.Scroll)

ScrollIntoView 컨테이너의 표시 영역(뷰포트) 내에 컨트롤을 표시하기 위해 컨테이너 개체의 내용 영역을 스크롤합니다.
ScrollIntoView 뷰포트에 텍스트 범위가 보일 때까지 텍스트 컨트롤을 세로로 스크롤합니다.
ScrollIntoView 컨테이너 개체의 콘텐츠 영역을 스크롤하여 컨테이너의 표시되는 영역(뷰포트) 내에 UI 자동화 요소를 표시합니다.
ScrollIntoView 텍스트 컨트롤이 뷰포트에 텍스트 범위가 표시될 때까지 스크롤되도록 합니다.
ScrollItemPattern_ScrollIntoView 컨테이너의 표시되는 영역(뷰포트) 내에 UI 자동화 요소를 표시하기 위해 컨테이너 개체의 콘텐츠 영역을 스크롤합니다.
ScrollPattern_Scroll 지정한 ScrollAmount 콘텐츠 영역의 현재 표시되는 영역을 가로, 세로 또는 둘 다 스크롤합니다.
ScrollPattern_SetScrollPercent 컨테이너를 가로, 세로 또는 둘 다의 특정 위치로 스크롤합니다.
Select 요소를 선택합니다.
Select 선택한 항목을 선택 취소한 다음 현재 요소를 선택합니다.
Select 이 텍스트 범위에 해당하는 텍스트 범위를 선택하고 이전 선택 영역을 제거합니다. (ITextRangeProvider.Select)
Select Microsoft 활성 접근성 선택을 수행합니다. (IUIAutomationLegacyIAccessiblePattern.Select)

Select 선택된 모든 항목의 선택을 취소한 다음 현재 요소를 선택합니다.
Select 이 텍스트 범위에 해당하는 텍스트 범위를 선택하고 이전 선택 영역을 제거합니다. (IUIAutomationTextRange.Select)
SelectionItemPattern_AddToSelection 선택되지 않은 요소를 컨트롤의 선택 영역에 추가합니다.
SelectionItemPattern_RemoveFromSelection 선택 컨테이너의 선택 영역에서 요소를 제거합니다.
SelectionItemPattern_Select 선택 컨테이너에서 요소를 선택합니다.
SetCurrentView 현재 컨트롤별 뷰를 설정합니다.
SetCurrentView 컨트롤의 보기를 설정합니다.
SetDockPosition 이 요소의 도킹 위치를 설정합니다.
SetDockPosition 이 요소의 dock 위치를 설정합니다.
Setfocus 포커스를 이 요소로 설정합니다.
Setfocus 키보드 포커스를 이 UI 자동화 요소로 설정합니다.

SetHmenuProp 이 메서드는 SetPropValue를 래핑하여 HMENU 기반 액세스 가능 요소에 주석을 지정하는 호출자에게 편리한 진입점을 제공합니다. 새 값이 문자열인 경우 IAccPropServices::SetHmenuPropStr을 대신 사용할 수 있습니다.
SetHmenuPropServer 이 메서드는 SetPropServer를 래핑하여 HMENU 기반 액세스 가능 요소에 주석을 지정하는 호출자에게 편리한 진입점을 제공합니다.
SetHmenuPropStr 이 메서드는 SetPropValue를 래핑하여 HMENU 기반 액세스 가능 요소에 주석을 지정하는 호출자에게 보다 편리한 진입점을 제공합니다.
SetHwndProp 이 메서드는 SetPropValue를 래핑하여 HWND 기반 액세스 가능 요소에 주석을 지정하는 호출자에게 편리한 진입점을 제공합니다. 새 값이 문자열인 경우 IAccPropServices::SetHwndPropStr를 대신 사용할 수 있습니다.
SetHwndPropServer 이 메서드는 SetPropServer를 래핑하여 HWND 기반 액세스 가능 요소에 주석을 지정하는 호출자에게 편리한 진입점을 제공합니다.
SetHwndPropStr 이 메서드는 SetPropValue를 래핑하여 HWND 기반 액세스 가능 요소에 주석을 지정하는 호출자에게 보다 편리한 진입점을 제공합니다.
SetPropServer 서버는 SetPropServer를 사용하여 액세스 가능한 요소에 대한 속성 배열에 주석을 추가하는 데 사용할 콜백 개체를 지정합니다.
SetPropValue SetPropValue를 사용하여 주석을 추가할 액세스 가능한 요소를 식별하고, 주석을 추가할 속성을 지정하고, 해당 속성에 대한 새 값을 제공합니다.
SetScrollPercent 컨트롤 내의 가로 및 세로 스크롤 위치를 전체 콘텐츠 영역의 백분율로 설정합니다.
SetScrollPercent 가로 및 세로 스크롤 위치를 UI 자동화 요소 내의 총 콘텐츠 영역의 백분율로 설정합니다.

설정할 프록시 팩터리 테이블을 설정합니다.
SetValue 컨트롤의 문자열 값을 설정합니다.
SetValue 컨트롤의 값을 설정합니다. (IRangeValueProvider.SetValue)
SetValue 컨트롤 값을 설정합니다.
SetValue 요소에 대한 Microsoft Active Accessibility 값 속성을 설정합니다.
SetValue 컨트롤의 값을 설정합니다. (IUIAutomationRangeValuePattern.SetValue)
SetValue 요소의 값을 설정합니다.
SetVisualState 창의 표시 상태를 변경합니다. 예를 들어 최소화하거나 최대화합니다.
SetWindowVisualState 창을 최소화, 최대화 또는 복원합니다.
SetWinEventHook 이벤트 범위에 대한 이벤트 후크 함수를 설정합니다.
SetWinEventsForAutomationEvent Microsoft UI 자동화 이벤트를 WinEvents에 매핑합니다.
ShowContextMenu 프로그래밍 방식으로 대상 요소의 상황에 맞는 메뉴를 호출합니다. (IRawElementProviderSimple2.ShowContextMenu)

ShowContextMenu 프로그래밍 방식으로 대상 요소의 상황에 맞는 메뉴를 호출합니다. (ITextRangeProvider2.ShowContextMenu)
ShowContextMenu 프로그래밍 방식으로 대상 요소의 상황에 맞는 메뉴를 호출합니다. (IUIAutomationElement3.ShowContextMenu)
ShowContextMenu 프로그래밍 방식으로 대상 텍스트 범위에서 상황에 맞는 메뉴를 호출합니다.
StartListening 지정된 형식의 입력 수신 대기를 시작합니다.
StartListening Microsoft UI 자동화 공급자가 마우스 또는 키보드 입력을 수신 대기하도록 합니다.
SynchronizedInputPattern_Cancel UI 자동화 공급자가 마우스 또는 키보드 입력에 대한 수신을 중지하도록 합니다.
SynchronizedInputPattern_StartListening UI 자동화 공급자가 마우스 또는 키보드 입력을 수신 대기하도록 합니다.
TextPattern_get_DocumentRange 전체 문서의 텍스트 범위를 가져옵니다.
TextPattern_get_SupportedTextSelection 텍스트 컨테이너의 내용을 선택하고 선택 취소할 수 있는지 여부를 확인합니다.
TextPattern_GetSelection 텍스트 패턴을 지원하는 텍스트 컨테이너에서 선택한 텍스트의 현재 범위를 가져옵니다.
TextPattern_GetVisibleRanges 텍스트 컨테이너에서 각 텍스트 범위가 부분적으로 표시되는 첫 번째 줄로 시작되고 부분적으로 표시되는 마지막 줄의 끝까지 이어지는 서로 분리된 텍스트 범위의 배열을 검색합니다.

TextPattern_RangeFromChild 지정된 노드에 걸쳐 있는 텍스트 범위를 가져옵니다.
TextPattern_RangeFromPoint 지정된 화면 좌표에 가장 가까운 퇴화(빈) 텍스트 범위를 검색합니다. (TextPattern_RangeFromPoint)
TextRange_AddToSelection 호출하는 텍스트 범위 시작 및 끝 엔드포인트에 해당하는 보조 텍스트를 강조 표시하여 여러 개의 연결되지 않은 선택을 지원하는 텍스트 컨테이너의 강조 표시된 텍스트의 기존 컬렉션에 추가합니다.
TextRange_Clone 텍스트 범위를 복사합니다.
TextRange_Compare 두 텍스트 범위를 비교합니다.
TextRange_CompareEndpoints 두 텍스트 범위에 동일한 엔드포인트가 있는지 여부를 나타내는 값을 반환합니다.
TextRange_ExpandToEnclosingUnit 텍스트 범위를 문자, Word, 줄 또는 페이지와 같은 더 크거나 작은 단위로 확장합니다.
TextRange_FindAttribute 지정된 텍스트 특성을 지원하는 텍스트의 첫 번째 부분을 지정된 방향으로 검색합니다.
TextRange_FindText 클라이언트에서 검색하는 텍스트가 포함된 지정된 방향으로 첫 번째 텍스트 범위를 반환합니다.
TextRange_GetAttributeValue 텍스트 범위에 대한 텍스트 특성의 값을 가져옵니다.
TextRange_GetBoundingRectangles 범위를 묶을 수 있는 경계 사각형의 최소 개수(줄당 직사각형 1개)를 검색합니다.

TextRange_GetChildren 지정된 텍스트 범위 내에 포함된 모든 UI 자동화 요소를 반환합니다.
TextRange_GetEnclosingElement 범위를 포함하는 다음으로 가장 작은 공급자에 대한 노드를 반환합니다.
TextRange_GetText 텍스트 범위의 텍스트를 지정된 수의 문자까지 반환합니다.
TextRange_Move 클라이언트에서 요청한 지정된 단위 수를 텍스트 범위로 이동합니다.
TextRange_MoveEndpointByRange 한 범위의 엔드포인트를 다른 범위의 엔드포인트로 이동합니다.
TextRange_MoveEndpointByUnit 범위의 엔드포인트를 지정된 단위 수로 이동합니다.
TextRange_RemoveFromSelection 여러 개의 연결되지 않은 선택을 지원하는 텍스트 컨테이너에 있는 선택한 텍스트의 기존 컬렉션에서 호출 텍스트 범위 TextPatternRangeEndpoint_Start 및 TextPatternRangeEndpoint_End 엔드포인트에 해당하는 선택한 텍스트를 제거합니다.
TextRange_ScrollIntoView 지정된 범위가 뷰포트에 표시되도록 텍스트를 스크롤합니다.
TextRange_Select 텍스트 범위를 선택합니다.
Toggle 컨트롤의 설정/해제 상태를 순환합니다.
Toggle 컨트롤의 토글 상태를 순환합니다.
TogglePattern_Toggle 컨트롤을 지원되는 다음 상태로 전환합니다.

TransformPattern_Move 요소를 화면의 지정된 위치로 이동합니다.
TransformPattern_Resize 화면의 요소 크기를 조정합니다.
TransformPattern_Rotate 화면에서 요소를 회전합니다.
UiaAddEvent UI 자동화 트리의 노드에서 이벤트에 대한 수신기를 추가합니다.
UiaClientsAreListening 클라이언트 애플리케이션이 Microsoft UI 자동화 이벤트를 구독하는지 여부를 나타내는 값을 가져옵니다.
UiaDisconnectAllProviders 호출 프로세스와 연결된 모든 공급자가 보유한 모든 Microsoft UI 자동화 리소스를 해제합니다.
UiaDisconnectProvider 특정 공급자가 Microsoft UI 자동화 개체에 보유하는 모든 참조를 해제합니다.
UiaEventAddWindow 이벤트 수신기에 창을 추가합니다.
UiaEventCallback 클라이언트가 구독한 이벤트가 발생할 때 UI 자동화 호출되는 클라이언트 구현 함수입니다.
UiaEventRemoveWindow 이벤트 수신기에서 창을 제거합니다.
UiaFind 검색 조건과 일치하는 하나 이상의 UI 자동화 노드를 검색합니다.
UiaGetErrorDescription 클라이언트에 전달할 수 있도록 오류 문자열을 가져옵니다. 이 메서드는 클라이언트에서 직접 사용되지 않습니다.

UiaGetPatternProvider 컨트롤 패턴을 검색합니다.
UiaGetPropertyValue UI 자동화 속성의 값을 검색합니다.
UiaGetReservedMixedAttributeValue Microsoft UI 자동화 텍스트 특성의 값이 텍스트 범위 내에서 달라진다는 것을 나타내는 예약된 값을 검색합니다.
UiaGetReservedNotSupportedValue Microsoft UI 자동화 속성 또는 텍스트 특성이 지원되지 않음을 나타내는 예약된 값을 검색합니다.
UiaGetRootNode 루트 UI 자동화 노드를 검색합니다.
UiaGetRuntimeId UI 자동화 노드의 런타임 식별자를 검색합니다.
UiaGetUpdatedCache 속성 값 및 컨트롤 패턴의 캐시를 업데이트.
UiaHasServerSideProvider 창에 Microsoft UI 자동화 공급자 구현이 있는지 확인합니다.
UiaHostProviderFromHwnd 창에 대한 호스트 공급자를 가져옵니다.
UiaHPatternObjectFromVariant VARIANT 형식에서 컨트롤 패턴 개체를 가져옵니다.
UiaHTextRangeFromVariant VARIANT 형식에서 텍스트 범위를 가져옵니다.
UiaHUiaNodeFromVariant VARIANT 형식에서 HUIANODE를 가져옵니다.

UiaIAccessibleFromProvider Microsoft UI 자동화 공급자를 대신하여 Microsoft Active Accessibility 데이터를 제공하는 IAccessible 구현을 검색합니다.
UiaLookupId PROPERTYID, PATTERNID, CONTROLTYPEID, TEXTATTRIBUTEID 또는 EVENTID가 필요한 메서드에서 사용할 수 있는 정수 식별자를 가져옵니다.
UiaNavigate UI 자동화 트리에서 탐색하고 필요에 따라 캐시된 정보를 검색합니다.
UiaNodeFromFocus 현재 입력 포커스가 있는 UI 요소의 UI 자동화 노드를 검색합니다.
UiaNodeFromHandle 창과 연결된 UI 자동화 노드를 검색합니다.
UiaNodeFromPoint 지정된 지점에서 요소의 UI 자동화 노드를 검색합니다.
UiaNodeFromProvider 원시 요소 공급자에 대한 UI 자동화 노드를 검색합니다.
UiaNodeRelease 메모리에서 노드를 삭제합니다.
UiaPatternRelease 메모리에서 할당된 패턴 개체를 삭제합니다.
UiaProviderCallback 요소에 대한 클라이언트 쪽 공급자를 가져오기 위해 UI 자동화 호출되는 애플리케이션 정의 함수입니다.
UiaProviderForNonClient 창의 전체 비 클라이언트 영역 또는 창의 비 클라이언트 영역에 있는 컨트롤에 대한 공급자를 가져옵니다.

UiaProviderFromIAccessible 지정된 Microsoft Active Accessibility 개체를 기반으로 Microsoft UI 자동화 공급자를 만듭니다.
UiaRaiseActiveTextPositionChangedEvent 공급자가 텍스트 컨트롤에 프로그래밍 방식으로 변경된 텍스트가 있음을 Microsoft UI 자동화 코어에 알리기 위해 호출됩니다. (UiaRaiseActiveTextPositionChangedEvent)
UiaRaiseAsyncContentLoadedEvent Microsoft UI 자동화 코어에 콘텐츠가 비동기적으로 로드되고 있음을 알리기 위해 공급자가 호출합니다.
UiaRaiseAutomationEvent 수신기에 이벤트를 알 수 있습니다.
UiaRaiseAutomationPropertyChangedEvent 공급자가 호출하여 요소 속성이 변경되었음을 Microsoft UI 자동화 코어에 알립니다.
UiaRaiseChangesEvent 공급자가 Microsoft UI 자동화 핵심에 변경이 발생했음을 알리기 위해 호출됩니다.
UiaRaiseNotificationEvent 공급자가 알림 이벤트를 시작하기 위해 호출합니다.
UiaRaiseStructureChangedEvent Microsoft UI 자동화 코어에 트리 구조가 변경되었음을 알리기 위해 공급자가 호출합니다.
UiaRaiseTextEditTextChangedEvent 공급자가 텍스트 컨트롤에 프로그래밍 방식으로 변경된 텍스트가 있음을 Microsoft UI 자동화 코어에 알리기 위해 호출됩니다. (UiaRaiseTextEditTextChangedEvent)
UiaRectIsEmpty 사각형에 모든 좌표가 0으로 설정되어 있는지 여부를 지정하는 부울 값을 가져옵니다.
UiaRectSetEmpty UiaRect 구조체의 요소를 0으로 설정합니다.

UiaRegisterProviderCallback 요소에 대한 공급자를 얻기 위해 UI 자동화 호출되는 애플리케이션 정의 메서드를 등록합니다.
UiaRemoveEvent UI 자동화 트리의 노드에서 이벤트에 대한 수신기를 제거합니다.
UiaReturnRawElementProvider 창에 대한 UI 자동화 공급자에 대한 인터페이스를 가져옵니다.
UiaSetFocus 입력 포커스를 UI의 지정된 요소로 설정합니다.
UiaTextRangeRelease 메모리에서 할당된 텍스트 범위 개체를 삭제합니다.
UnhookWinEvent SetWinEventHook에 대한 이전 호출에서 만든 이벤트 후크 함수를 제거합니다.
UnregisterPointerInputTarget 호출자가 지정된 형식의 모든 포인터 입력이 리디렉션되는 대상 창의 등록을 취소할 수 있습니다.
UnregisterPointerInputTargetEx UnregisterPointerInputTargetEx를 변경하거나 사용할 수 없습니다. 대신 UnregisterPointerInputTarget을 사용합니다.
ValuePattern_SetValue 요소의 텍스트 값을 설정합니다.
VariantToRect 사각형 좌표를 포함하는 VARIANT를 RECT로 변환합니다.
VirtualizedItemPattern_Realize 가상 항목을 UI 자동화 요소로 완전히 액세스할 수 있도록 만듭니다. (VirtualizedItemPattern_Realize)

WaitForInputIdle 지정된 시간 동안 또는 연결된 프로세스가 유휴 상태가 될 때까지 중 먼저 완료되는 시간 동안 코드 호출을 차단합니다. (IWindowProvider.WaitForInputIdle)
WaitForInputIdle 지정된 시간 동안 또는 연결된 프로세스가 유휴 상태가 될 때까지 중 먼저 완료되는 시간 동안 코드 호출을 차단합니다. (IUIAutomationWindowPattern.WaitForInputIdle)
WindowFromAccessibleObject IAccessible 인터페이스의 특정 인스턴스에 해당하는 창 핸들을 검색합니다.
WindowPattern_Close 열려 있는 창을 닫습니다.
WindowPattern_SetWindowVisualState 창의 시각적 상태를 설정합니다. 예를 들어 창을 최대화합니다.
WindowPattern_WaitForInputIdle 지정된 시간 동안 또는 연결된 프로세스가 유휴 상태가 될 때까지 중 먼저 완료되는 시간 동안 코드 호출을 차단합니다. (WindowPattern_WaitForInputIdle)
WINEVENTPROC 액세스 가능한 개체에 의해 생성된 이벤트에 대한 응답으로 시스템이 호출하는 애플리케이션 정의 콜백(또는 후크) 함수입니다.
Zoom 컨트롤의 뷰포트를 확대/축소합니다. (ITransformProvider2.Zoom)
Zoom 컨트롤의 뷰포트를 확대/축소합니다. (IUIAutomationTransformPattern2.Zoom)
ZoomByUnit 지정된 논리 단위로 컨트롤의 뷰포트를 확대/축소합니다.
ZoomByUnit 컨트롤의 뷰포트를 지정된 단위로 확대/축소합니다.

인터페이스

IAccClientDocMgr 클라이언트 애플리케이션이 문서를 검색하는 메서드를 노출합니다.
IAccDictionary 문자열 조작을 위한 메서드를 노출합니다.
IAccessible 클라이언트 애플리케이션에서 사용자 인터페이스 요소와 해당 자식에 액세스할 수 있도록 하는 메서드 및 속성을 노출합니다.
IAccessibleEx Microsoft UI 자동화 호출된 메서드를 노출하여 Microsoft Active Accessibility를 지원하는 컨트롤에 대한 추가 정보를 검색합니다.
IAccessibleHandler 개체 ID에서 액세스 가능한 요소를 검색하는 메서드를 노출합니다.
IAccessibleHostingElementProviders Microsoft Active Accessibility 개체는 개체가 Microsoft UI 자동화 구현하는 창 없는 Microsoft ActiveX 컨트롤을 포함하는 접근성 트리의 루트인 경우 이 인터페이스를 구현합니다.
IAccessibleWindowlessSite Microsoft ActiveX 컨트롤 사이트는 이 인터페이스를 구현하여 Microsoft Active Accessibility 구현이 있는 창 없는 ActiveX 컨트롤이 접근성을 표현할 수 있도록 합니다.
IAccIdentity 액세스 가능한 요소에 대한 고유 식별자를 제공하는 메서드를 노출합니다.
IAccPropServer 액세스 가능한 요소의 속성 값을 검색하는 메서드를 노출합니다.
IAccPropServices 액세스 가능한 요소에 주석을 추가하고 ID 문자열을 조작하기 위한 메서드를 노출합니다.

IAccServerDocMgr 클라이언트 애플리케이션에서 문서에 액세스할 수 있도록 하는 메서드를 노출합니다.
IAnnotationProvider 문서에서 주석의 속성을 노출합니다.
ICoCreatedLocally 로컬 개체에 대한 정보를 반환하는 메서드를 노출합니다.
ICoCreateLocally 클라이언트 애플리케이션이 서버 컨텍스트에서 도우미 개체를 만들 수 있도록 하는 메서드를 노출합니다.
IDockProvider 도킹 컨테이너의 요소에 대한 액세스를 제공합니다.
IDragProvider Microsoft UI 자동화 요소가 끌어서 놓기 작업의 일부로 끌 수 있는 요소로 자신을 설명할 수 있도록 합니다.
IDropTargetProvider Microsoft UI 자동화 요소가 끌어서 놓기 작업의 일부로 끌어온 요소의 드롭을 받을 수 있는 요소로 자신을 설명할 수 UI 자동화.
IExpandCollapseProvider 콘텐츠를 표시하기 위해 시각적으로 확장되고 축소되어 콘텐츠를 숨기는 컨트롤에 대한 액세스를 제공합니다.
IGridItemProvider IGridProvider를 구현하는 컨테이너의 개별 자식 컨트롤에 대한 액세스를 제공합니다.
IGridProvider 키보드를 사용하여 트래버스할 수 있는 2차원 논리 좌표계(즉, Microsoft UI 자동화 클라이언트가 인접한 컨트롤로 이동할 수 있음)로 구성된 자식 요소 컬렉션의 컨테이너 역할을 하는 컨트롤에 대한 액세스를 제공합니다.

IInvokeProvider 명확한 단일 작업을 시작하거나 수행하고 활성화될 때 상태를 유지 관리하지 않는 컨트롤에 대한 액세스를 제공합니다.
IItemContainerProvider 가상 목록 보기와 같은 다른 컨트롤의 컨테이너 역할을 하는 컨트롤에 대한 액세스를 제공합니다.
ILegacyIAccessibleProvider Microsoft UI 자동화 클라이언트가 Microsoft Active Accessibility 요소의 기본 IAccessible 구현에 액세스할 수 있도록 합니다.
IMultipleViewProvider 동일한 정보 또는 자식 컨트롤 집합의 여러 표현을 제공하고 전환할 수 있는 컨트롤에 대한 액세스를 제공합니다.
IObjectModelProvider 컨트롤 또는 애플리케이션에서 구현하는 기본 개체 모델에 대한 액세스를 제공합니다. (IObjectModelProvider)
IProxyProviderWinEventHandler WinEvents를 처리하기 위해 프록시 공급자가 구현하는 메서드를 노출합니다.
IProxyProviderWinEventSink 이벤트를 발생 하려면 프록시 공급자에서 사용 하는 메서드를 노출 합니다.
IRangeValueProvider 범위 내의 값으로 설정할 수 있는 컨트롤에 대한 액세스를 제공합니다.
IRawElementProviderAdviseEvents Microsoft UI 자동화 클라이언트 애플리케이션이 해당 조각의 이벤트 수신 대기를 시작하거나 종료할 때 조각의 루트 요소에 알리기 위해 호출되는 메서드를 노출합니다.
IRawElementProviderFragment 목록 상자 또는 목록 항목과 같이 여러 수준 깊이의 구조에 속하는 UI 요소에 메서드 및 속성을 노출합니다. Microsoft UI 자동화 공급자가 구현합니다.
IRawElementProviderFragmentRoot 루트 요소에 대한 메서드 및 속성을 조각에 노출합니다.

IRawElementProviderHostingAccessibles 이 인터페이스는 공급자가 Microsoft Active Accessibility를 지원하는 창 없는 컨트롤을 포함하는 접근성 트리의 루트인 경우 Microsoft UI 자동화 공급자에 의해 구현됩니다.
IRawElementProviderHwndOverride 조각의 UI 자동화 트리 내에서 창 기반 요소의 위치를 변경할 수 있는 메서드를 노출합니다.
IRawElementProviderSimple 간단한 UI 요소를 노출하는 메서드 및 속성을 정의합니다.
IRawElementProviderSimple2 프로그래밍 방식으로 상황에 맞는 메뉴를 호출할 수 있도록 IRawElementProviderSimple 인터페이스를 확장합니다.
IRawElementProviderSimple3 IRawElementProviderSimple2 인터페이스를 확장하여 접근성 있는 기술이 기본 콘텐츠 형식을 말해야 하는 방법에 대한 메타데이터를 검색할 수 있도록 합니다.
IRawElementProviderWindowlessSite Microsoft ActiveX 컨트롤 사이트는 이 인터페이스를 구현하여 Microsoft UI 자동화 지원 ActiveX 컨트롤이 접근성을 표현할 수 있도록 합니다.
IScrollItemProvider IScrollProvider를 구현하는 컨테이너의 개별 자식 컨트롤에 대한 액세스를 제공합니다.
IScrollProvider 자식 개체 컬렉션에 대해 스크롤 가능한 컨테이너 역할을 하는 컨트롤에 대한 액세스를 제공합니다.
ISelectionItemProvider ISelectionProvider를 구현하는 컨테이너의 선택 가능한 개별 자식 컨트롤에 대한 액세스를 제공합니다.
ISelectionProvider 선택 가능한 개별 자식 항목 컬렉션의 컨테이너 역할을 하는 컨트롤에 대한 액세스를 제공합니다.
ISelectionProvider2 선택한 항목에 대한 정보를 제공하도록 ISelectionItemProvider 인터페이스를 확장합니다.

ISpreadsheetItemProvider 스프레드시트의 항목(셀)에 대한 정보에 대한 액세스를 제공합니다.
ISpreadsheetProvider 스프레드시트의 항목(셀)에 대한 액세스를 제공합니다.
IStylesProvider 문서 내용과 연결된 비주얼 스타일에 대한 액세스를 제공합니다.
ISynchronizedInputProvider Microsoft UI 자동화 클라이언트 애플리케이션에서 마우스 또는 키보드 입력을 특정 UI 요소로 보낼 수 있습니다.
ITableItemProvider ITableProvider를 구현하는 컨테이너의 자식 컨트롤에 대한 액세스를 제공합니다.
ITableProvider 자식 요소 컬렉션의 컨테이너 역할을 하는 컨트롤에 대한 액세스를 제공합니다.
ITextChildProvider 다른 텍스트 기반 컨트롤의 자식 또는 하위 항목인 텍스트 기반 컨트롤(또는 텍스트에 포함된 개체)에 대한 액세스를 제공합니다.
ITextEditProvider Microsoft UI 자동화 공급자가 프로그래밍 방식 텍스트 편집 작업을 노출할 수 있도록 ITextProvider 인터페이스를 확장합니다.
ITextProvider 텍스트를 포함하는 컨트롤에 대한 액세스를 제공합니다.
ITextProvider2 ITextProvider 인터페이스를 확장하여 Microsoft UI 자동화 공급자가 주석의 대상인 텍스트 콘텐츠와 공급자에 속한 케어에 대한 정보를 노출할 수 있도록 합니다.
ITextRangeProvider ITextProvider 또는 ITextProvider2를 구현하는 텍스트 컨테이너의 연속 텍스트 범위에 대한 액세스를 제공합니다.

ITextRangeProvider2 Microsoft UI 자동화 공급자가 상황에 맞는 메뉴를 호출할 수 있도록 ITextRangeProvider 인터페이스를 확장합니다.
IToggleProvider 상태 집합을 순환하고 설정된 후 상태를 유지할 수 있는 컨트롤에 대한 액세스를 제공합니다.
ITransformProvider 2차원 공간 내에서 이동, 크기 조정 및/또는 회전할 수 있는 컨트롤에 대한 액세스를 제공합니다.
ITransformProvider2 ITransformProvider 인터페이스를 확장하여 Microsoft UI 자동화 공급자가 컨트롤의 뷰포트 확대/축소 기능을 지원하기 위해 속성을 노출할 수 있도록 합니다.
IUIAutomation Microsoft UI 자동화 클라이언트 애플리케이션에서 UI 자동화 요소를 검색, 액세스 및 필터링할 수 있도록 하는 메서드를 노출합니다.
IUIAutomation2 IUIAutomation 인터페이스를 확장하여 Microsoft UI 자동화 기능을 제어하는 추가 메서드를 노출합니다.
IUIAutomation3 IUIAutomation2 인터페이스를 확장하여 Microsoft UI 자동화 기능을 제어하는 추가 메서드를 노출합니다.
IUIAutomation4 IUIAutomation3 인터페이스를 확장하여 Microsoft UI 자동화 기능을 제어하는 추가 메서드를 노출합니다.
IUIAutomation5 IUIAutomation4 인터페이스를 확장하여 Microsoft UI 자동화 기능을 제어하는 추가 메서드를 노출합니다.
IUIAutomation6 IUIAutomation5 인터페이스를 확장하여 Microsoft UI 자동화 기능을 제어하는 추가 메서드를 노출합니다.

IUIAutomationActiveTextPositionChangedEventHandler 활성 텍스트 위치가 변경되면 발생하는 Microsoft UI 자동화 이벤트를 처리하는 메서드를 노출합니다.
IUIAutomationAndCondition Microsoft UI 자동화 클라이언트 애플리케이션에서 AND 기반 속성 조건에 대한 정보를 검색하는 데 사용할 수 있는 속성 및 메서드를 노출합니다.
IUIAutomationAnnotationPattern 문서에서 주석의 속성에 대한 액세스를 제공합니다.
IUIAutomationBoolCondition TRUE(모든 요소 선택) 또는 FALSE(요소가 선택되지 않음)일 수 있는 조건을 나타냅니다.
IUIAutomationCacheRequest 캐시 요청의 속성 및 메서드를 노출합니다. 클라이언트 애플리케이션은 이 인터페이스를 사용하여 Microsoft UI 자동화 요소를 가져올 때 캐시할 속성 및 컨트롤 패턴을 지정합니다.
IUIAutomationChangesEventHandler 하나 이상의 Microsoft UI 자동화 변경 이벤트를 처리하는 메서드를 노출합니다.
IUIAutomationCondition UI 자동화 트리에서 요소를 검색할 때 필터링에 사용되는 조건에 대한 기본 인터페이스입니다.
IUIAutomationCustomNavigationPattern 사용자 지정 탐색 순서를 지원하는 컨트롤에 Microsoft UI 자동화 클라이언트의 액세스를 지원하는 메서드를 노출합니다.
IUIAutomationDockPattern 자식 요소가 서로 상대적으로 가로 및 세로로 정렬될 수 있도록 하는 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationDragPattern 끌어서 놓기 작업의 일부로 끌 수 있는 요소에 대해 UI 자동화 공급자가 노출하는 정보에 대한 액세스를 제공합니다.

IUIAutomationDropTargetPattern 끌어서 놓기 작업의 놓기 대상이 될 수 있는 요소에 대해 Microsoft UI 자동화 공급자가 노출하는 끌어서 놓기 정보에 대한 액세스를 제공합니다.
IUIAutomationElement UI 항목을 나타내는 UI 자동화 요소에 대한 메서드 및 속성을 노출합니다.
IUIAutomationElement2 IUIAutomationElement 인터페이스를 확장합니다.
IUIAutomationElement3 IUIAutomationElement2 인터페이스를 확장합니다.
IUIAutomationElement4 IUIAutomationElement3 인터페이스를 확장합니다.
IUIAutomationElement5 IUIAutomationElement4 인터페이스를 확장하여 현재 및 캐시된 랜드마크 데이터에 대한 액세스를 제공합니다.
IUIAutomationElement6 IUIAutomationElement5 인터페이스를 확장하여 현재 및 캐시된 전체 설명에 대한 액세스를 제공합니다.
IUIAutomationElement7 IUIAutomationElement6 인터페이스를 확장합니다.
IUIAutomationElement8 IUIAutomationElement7 인터페이스를 확장합니다.
IUIAutomationElement9 IUIAutomationElement8 인터페이스를 확장합니다.
IUIAutomationElementArray UI 자동화 요소의 컬렉션을 나타냅니다.

IUIAutomationEventHandler Microsoft UI 자동화 이벤트를 처리하는 메서드를 노출합니다.
IUIAutomationEventHandlerGroup IUIAutomation6에 정의된 CreateEventHandlerGroup 및 AddEventHandlerGroup 메서드를 통해 대량 등록을 위해 컬렉션에 하나 이상의 이벤트를 추가하는 메서드를 노출합니다.
IUIAutomationExpandCollapsePattern 시각적으로 확장하여 콘텐츠를 표시하고 축소하여 콘텐츠를 숨길 수 있는 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationFocusChangedEventHandler 키보드 포커스가 다른 UI 자동화 요소로 이동할 때 발생하는 이벤트를 처리하는 메서드를 노출합니다.
IUIAutomationGridItemPattern IUIAutomationGridPattern 인터페이스를 지원하는 그리드 스타일 컨테이너의 자식 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationGridPattern 행과 열로 트래버스할 수 있는 2차원 논리 좌표계로 구성된 자식 컨트롤 컬렉션의 컨테이너 역할을 하는 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationInvokePattern 클라이언트 애플리케이션이 컨트롤의 동작(일반적으로 단추)을 호출할 수 있도록 하는 메서드를 노출합니다.
IUIAutomationItemContainerPattern 가상 목록과 같은 컨테이너에서 항목을 검색하는 메서드를 노출합니다.
IUIAutomationLegacyIAccessiblePattern Microsoft UI 자동화 클라이언트가 MSAA(Microsoft Active Accessibility) 서버에서 UI 정보를 검색할 수 있도록 하는 메서드 및 속성을 노출합니다.
IUIAutomationMultipleViewPattern 동일한 정보 또는 자식 컨트롤 집합의 여러 표현 간에 전환할 수 있는 컨트롤에 대한 액세스를 제공합니다.

IUIAutomationNotCondition 다른 조건의 음수인 조건을 나타냅니다.
IUIAutomationNotificationEventHandler Microsoft UI 자동화 알림 이벤트를 처리하는 메서드를 노출합니다.
IUIAutomationObjectModelPattern 컨트롤 또는 애플리케이션에서 구현하는 기본 개체 모델에 대한 액세스를 제공합니다. (IUIAutomationObjectModelPattern)
IUIAutomationOrCondition 여러 조건으로 구성된 조건을 나타내며, 그 중 하나 이상이 true여야 합니다.
IUIAutomationPatternHandler 클라이언트 API 래퍼 개체를 반환하고 실제 공급자 인스턴스에 대한 속성 및 메서드 요청을 취소합니다.
IUIAutomationPatternInstance 컨트롤 패턴 개체를 나타냅니다. 클라이언트 API 래퍼는 이 인터페이스를 사용하여 GetProperty 및 CallMethod 메서드 측면에서 모든 속성 및 메서드 호출을 구현합니다.
IUIAutomationPropertyChangedEventHandler 속성이 변경될 때 발생하는 Microsoft UI 자동화 이벤트를 처리하는 메서드를 노출합니다.
IUIAutomationPropertyCondition UI 자동화 요소를 찾는 데 사용되는 속성 값을 기반으로 하는 조건을 나타냅니다.
IUIAutomationProxyFactory UI 자동화 대한 기본 지원이 없는 UI 요소에 대한 Microsoft UI 자동화 공급자를 만드는 개체의 속성과 메서드를 노출합니다. 이 인터페이스는 프록시에 의해 구현됩니다.
IUIAutomationProxyFactoryEntry Microsoft UI 자동화 유지 관리하는 테이블의 프록시 팩터리를 나타내며 클라이언트 애플리케이션에서 IUIAutomationProxyFactory 개체와 상호 작용하는 데 사용할 수 있는 속성과 메서드를 노출합니다.

IUIAutomationProxyFactoryMapping 프록시 팩터리 테이블에 대한 속성 및 메서드를 노출합니다. 각 테이블 항목은 IUIAutomationProxyFactoryEntry 인터페이스로 표시됩니다. 항목은 시스템에서 프록시를 사용하려고 시도하는 순서대로 되어 있습니다.
IUIAutomationRangeValuePattern 값 범위를 표시하는 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationRegistrar 새 컨트롤 패턴, 속성 및 이벤트를 등록하는 메서드를 노출합니다.
IUIAutomationScrollItemPattern 스크롤 가능한 보기의 항목을 보기의 표시되는 부분에 배치할 수 있도록 하는 메서드를 노출합니다.
IUIAutomationScrollPattern 자식 요소 컬렉션의 스크롤 가능 컨테이너 역할을 하는 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationSelectionItemPattern IUIAutomationSelectionPattern을 지원하는 컨테이너 컨트롤의 선택 가능한 자식 항목에 대한 액세스를 제공합니다.
IUIAutomationSelectionPattern 선택 가능한 자식 항목이 포함된 컨트롤에 대한 액세스를 제공합니다. 이 요소의 자식은 IUIAutomationSelectionItemPattern을 지원합니다.
IUIAutomationSelectionPattern2 IUIAutomationSelectionPattern 인터페이스를 확장하여 선택한 항목에 대한 정보를 제공합니다.
IUIAutomationSpreadsheetItemPattern 클라이언트 애플리케이션이 스프레드시트의 항목(셀)에 대한 정보를 검색할 수 있도록 합니다.
IUIAutomationSpreadsheetPattern 클라이언트 애플리케이션이 스프레드시트의 항목(셀)에 액세스할 수 있도록 합니다.
IUIAutomationStructureChangedEventHandler Microsoft UI 자동화 트리 구조가 변경될 때 발생하는 이벤트를 처리하는 메서드를 노출합니다.

IUIAutomationStylesPattern Microsoft UI 자동화 클라이언트가 문서의 요소와 연결된 비주얼 스타일을 검색할 수 있도록 합니다.
IUIAutomationSynchronizedInputPattern 컨트롤의 키보드 또는 마우스 입력에 대한 액세스를 제공합니다.
IUIAutomationTableItemPattern IUIAutomationTablePattern을 지원하는 컨테이너의 자식 요소에 대한 액세스를 제공합니다.
IUIAutomationTablePattern 자식 요소 컬렉션의 컨테이너 역할을 하는 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationTextChildPattern 다른 텍스트 기반 컨트롤의 자식 또는 하위 항목인 텍스트 기반 컨트롤(또는 텍스트에 포함된 개체)에 대한 액세스를 제공합니다.
IUIAutomationTextEditPattern 예를 들어 자동 고침을 수행하거나 IME(입력기)를 통한 입력 작성을 지원하는 컨트롤과 같이, 텍스트를 수정하는 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationTextEditTextChangedEventHandler Microsoft UI 자동화 텍스트 편집 컨트롤에서 텍스트 변경 이벤트를 보고할 때 발생하는 이벤트를 처리하는 메서드를 노출합니다.
IUIAutomationTextPattern 텍스트가 포함된 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationTextPattern2 IUIAutomationTextPattern 인터페이스를 확장합니다.
IUIAutomationTextRange IUIAutomationTextPattern 인터페이스를 지원하는 컨테이너의 연속 텍스트 범위에 대한 액세스를 제공합니다. 클라이언트 애플리케이션은 IUIAutomationTextRange 인터페이스를 사용하여 텍스트 범위에서 포함된 개체를 선택, 비교 및 검색할 수 있습니다.

IUIAutomationTextRange2 IUIAutomationTextRange 인터페이스를 확장하여 Microsoft UI 자동화 클라이언트가 프로그래밍 방식으로 상황에 맞는 메뉴를 호출할 수 있도록 합니다.
IUIAutomationTextRange3 텍스트 범위의 기본 서식 있는 텍스트 데이터에 더 빠르게 액세스할 수 있도록 IUIAutomationTextRange2 인터페이스를 확장합니다.
IUIAutomationTextRangeArray IUIAutomationTextRange 개체의 컬렉션을 나타냅니다.
IUIAutomationTogglePattern 상태 집합을 순환하고 설정된 후 상태를 유지할 수 있는 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationTransformPattern 이동, 크기 조정 또는 회전할 수 있는 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationTransformPattern2 IUIAutomationTransformPattern 인터페이스를 확장하여 Microsoft UI 자동화 클라이언트가 프로그래밍 방식으로 컨트롤의 뷰포트 확대/축소 기능에 액세스할 수 있도록 합니다.
IUIAutomationTreeWalker 클라이언트 애플리케이션이 데스크톱에서 UI 자동화 요소를 보고 탐색하는 데 사용하는 UI 자동화 속성 및 메서드를 노출합니다.
IUIAutomationValuePattern 범위에 포함되지 않고 문자열로 나타낼 수 있는 값이 포함된 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationVirtualizedItemPattern Microsoft UI 자동화 트리의 자리 표시자 자동화 요소로 표시되는 항목인 가상화된 항목을 나타냅니다.
IUIAutomationWindowPattern 창의 기본 기능에 대한 액세스를 제공합니다.
IValueProvider 범위에 포함되지 않고 문자열로 나타낼 수 있는 내장 값이 있는 컨트롤에 대한 액세스를 제공합니다.

IVersionInfo 액세스 가능한 요소에 대한 버전 정보를 제공하는 메서드를 노출합니다.
IVirtualizedItemProvider Microsoft UI 자동화 트리에서 자리 표시자 자동화 요소로 표시되는 항목인 가상화된 항목에 대한 액세스를 제공합니다.
IWindowProvider 컨트롤의 기본 창 기반 기능에 대한 액세스를 제공합니다.

구조체

ACCESSTIMEOUT 접근성 기능과 관련된 제한 시간에 대한 정보를 포함합니다.
ExtendedProperty 확장 속성에 대한 정보를 포함합니다.
FILTERKEYS 장애가 있는 사용자가 키보드 반복 속도(RepeatKeys), 수용 지연(SlowKeys) 및 반송률(BounceKeys)을 설정할 수 있도록 하는 FilterKeys 접근성 기능에 대한 정보를 포함합니다.
HIGHCONTRASTA 고대비 접근성 기능에 대한 정보를 포함합니다. (ANSI)
HIGHCONTRASTW 고대비 접근성 기능에 대한 정보를 포함합니다. (유니코드)
마우스 MouseKeys 접근성 기능에 대한 정보를 포함합니다.
MSAAMENUINFO 서버 개발자가 소유자가 그린 메뉴 항목의 이름을 노출하는 데 사용됩니다.

SERIALKEYSA 직렬 포트에 연결된 통신 지원의 데이터를 시스템이 키보드 및 마우스 입력을 시뮬레이션하는 명령으로 해석하는 SerialKeys 접근성 기능에 대한 정보를 포함합니다. (ANSI)
SERIALKEYSW 직렬 포트에 연결된 통신 지원의 데이터를 시스템이 키보드 및 마우스 입력을 시뮬레이션하는 명령으로 해석하는 SerialKeys 접근성 기능에 대한 정보를 포함합니다. (유니코드)
SOUNDSENTRYA SoundSentry 접근성 기능에 대한 정보를 포함합니다. SoundSentry 기능이 켜지면 컴퓨터는 소리가 생성될 때만 시각적 표시를 표시합니다. (ANSI)
SOUNDSENTRYW SoundSentry 접근성 기능에 대한 정보를 포함합니다. SoundSentry 기능이 켜지면 컴퓨터는 소리가 생성될 때만 시각적 표시를 표시합니다. (유니코드)
고정키 StickyKeys 접근성 기능에 대한 정보를 포함합니다.
토글키 ToggleKeys 접근성 기능에 대한 정보를 포함합니다.
UiaAndOrCondition 참고: 이 구조는 더 이상 사용되지 않습니다. 복잡한 조건에 대한 정보를 포함합니다.
UiaAsyncContentLoadedEventArgs 참고: 이 구조는 더 이상 사용되지 않습니다. 콘텐츠가 UI 요소에 의해 비동기적으로 로드될 때 발생하는 이벤트에 대한 정보를 포함합니다.
UiaCacheRequest 참고: 이 구조는 더 이상 사용되지 않습니다. UI 자동화 요소에 대한 데이터를 캐시하는 요청에 대한 정보를 포함합니다.
UiaChangeInfo 발생한 UI 자동화 변경에 대한 데이터를 포함합니다.
UiaCondition 참고: 이 구조는 더 이상 사용되지 않습니다. 조건에 대한 정보를 포함합니다.

UiaEventArgs 참고: 이 구조는 더 이상 사용되지 않습니다. Microsoft UI 자동화 이벤트에 대한 정보를 포함합니다.
UiaFindParams 참고: 이 구조는 더 이상 사용되지 않습니다. UiaFind 함수에 사용되는 매개 변수를 포함합니다.
UiaNotCondition 참고: 이 구조는 더 이상 사용되지 않습니다. 음수 조건에 대한 정보를 포함합니다.
UiaPoint 점의 좌표를 포함합니다.
UiaPropertyChangedEventArgs 참고: 이 구조는 더 이상 사용되지 않습니다. Microsoft UI 자동화 요소 속성 변경이 발생할 때 발생하는 이벤트에 대한 정보를 포함합니다.
UiaPropertyCondition 참고: 이 구조는 더 이상 사용되지 않습니다. 일치하는 속성이 있는 UI 자동화 요소를 찾는 데 사용되는 조건에 대한 정보를 포함합니다.
UiaRect 사각형의 위치와 크기를 화면 좌표로 포함합니다.
UiaStructureChangedEventArgs 참고: 이 구조는 더 이상 사용되지 않습니다. Microsoft UI 자동화 트리의 구조가 변경될 때 발생하는 이벤트에 대한 정보를 포함합니다.
UIAutomationEventInfo 사용자 지정 이벤트에 대한 정보를 포함합니다.
UIAutomationMethodInfo 사용자 지정 컨트롤 패턴에서 지원하는 메서드에 대한 정보를 포함합니다.
UIAutomationParameter 사용자 지정 컨트롤 패턴의 매개 변수에 대한 정보를 포함합니다.

UIAutomationPatternInfo 사용자 지정 컨트롤 패턴에 대한 정보를 포함합니다.
UIAutomationPropertyInfo 사용자 지정 속성에 대한 정보를 포함합니다.
UiaWindowClosedEventArgs 참고: 이 구조는 더 이상 사용되지 않습니다. 하나 이상의 창이 닫히면 발생하는 이벤트에 대한 정보를 포함합니다.

Windows 주소록

Windows 주소록 기술 개요입니다.

Windows 주소록을 개발하려면 다음 헤더가 필요합니다.

- wabapi.h
- wabdefs.h
- wabiab.h
- wabtags.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 주소록

열거 목록

테이블 확장

성별 사용하지 마세요. Gender 열거형은 PR_GENDER 속성에 사용할 수 있는 값을 지정합니다.

Functions

테이블 확장

AllocateBuffer WAB(Windows 주소록) 메서드에 전달되는 버퍼에 대한 메모리를 할당합니다. 버퍼는 IWABObject::FreeBuffer를 사용하여 해제해야 하며 IWABObject::AllocateMore를 사용하여 다시 할당할 수 있습니다.
AllocateMore 이전에 IWABObject::AllocateBuffer 메서드를 사용하여 할당된 다른 버퍼에 연결된 메모리 버퍼를 할당합니다.
백업 이 메서드는 구현되지 않습니다. (IWABObject.Backup)

찾기 WAB(Windows 주소록) 찾기 대화 상자를 시작합니다.
freeBuffer IWABObject::AllocateBuffer 또는 다른 WAB(Windows 주소록) 메서드를 사용하여 할당된 메모리를 해제합니다.
GetLastError 이 메서드는 구현되지 않습니다. (IWABObject.GetLastError)
GetMe "ME"로 지정된 개체의 항목 식별자를 검색합니다.
가져오기 .wab 파일을 사용자의 주소록으로 가져옵니다.
LDAPUrl LDAP(Lightweight Directory Access Protocol) URL을 처리하고 URL에서 가져온 결과를 표시합니다.
SetMe 특정 연락처를 ME 개체로 지정합니다.
VCardCreate 지정된 MailUser 개체의 속성을 vCard 파일로 변환합니다.
VCardDisplay vCard 파일에 속성을 표시합니다.
VCardRetrieve vCard 파일을 읽고 vCard 속성을 포함하는 MailUser 개체를 만듭니다.
WABOpen 사용하지 마세요. 여러 개체 인터페이스를 통해 주소록에 대한 액세스를 제공합니다. 루트 인터페이스는 IAddrBook의 MAPI 구현 하위 집합인 IAddrBook입니다.
WABOpenEx WABOpenEx는 Windows Vista를 기준으로 더 이상 사용할 수 없습니다.

Interfaces

테이블 확장

IABContainer 사용하지 마세요. 이 인터페이스는 주소록 컨테이너에 대한 액세스를 제공합니다. 애플리케이션은 인터페이스의 메서드를 호출하여 이름 확인을 수행하고 받는 사람을 만들고, 복사하고, 삭제합니다.
IDistList 사용하지 마세요. 이 인터페이스는 수정 가능한 주소록 컨테이너의 메일 그룹 액세스를 제공하는 데 사용됩니다. 인터페이스는 이름 확인을 수행하는 것 외에도 메일 목록을 만들고, 복사하고, 삭제하는 메서드를 제공합니다.
IMailUser 사용하지 마세요. 이 인터페이스는 메일 사용자 개체에 대한 액세스를 제공합니다.
IMAPITable 사용하지 마세요. 이 인터페이스는 WAB(Windows 주소록) 컨테이너 및 배포 목록의 콘텐츠 테이블에 사용됩니다.
IWABExtInit 사용하지 마세요. 이 인터페이스는 표시되는 WAB(Windows Address Book) 개체(예: 속성 시트 또는 상황에 맞는 메뉴)를 나타냅니다.
IWABObject 사용하지 마세요. 이 인터페이스는 메모리 할당 함수 및 데이터베이스 유지 관리 함수에 대한 함수 포인터를 포함하는 WAB(Windows Address Book) 개체에 대한 액세스를 제공합니다.

구조

테이블 확장

SRestriction 사용하지 마세요. 테이블 보기를 특정 행으로 제한하는 필터에 대해 설명합니다.
ADRENTRY

ADRENTRY(wabdefs.h) - 사용하지 마세요. 하나 이상의 받는 사람에 속하는 0개 이상의 속성을 설명합니다.
ADRLIST ADRLIST(wabdefs.h) - 사용하지 마세요. 하나 이상의 받는 사람에 속하는 0개 이상의 속성을 설명합니다. G
ADRPARM 사용하지 마세요. 일반 주소 대화 상자의 표시 및 동작에 대해 설명합니다.
ENTRYID 사용하지 마세요. MAPI 개체에 대한 항목 식별자 정보를 포함합니다.
SBinaryArray 사용하지 마세요. MAPI 개체를 나타내는 항목 식별자의 배열입니다. SBinaryArray와 동일한 구현을 사용합니다.
SPropProblem 사용하지 마세요. 속성과 관련된 작업과 관련된 오류를 설명합니다.
SPropProblemArray 사용하지 마세요. 하나 이상의 SPropProblem 구조체 배열을 포함합니다.
SPropTagArray 사용하지 마세요. 속성 태그의 배열을 포함합니다.
SPropValue 사용하지 마세요. 속성 태그 값을 포함합니다.
슬로우 (동음이의) 사용하지 마세요. 특정 개체에 대해 선택한 속성이 포함된 테이블의 행에 대해 설명합니다.
SRowSet 사용하지 마세요. SRow 구조의 배열을 포함합니다. 각 SRow 구조는 테이블의 행을 설명합니다.
SSortOrder 사용하지 마세요. 정렬 키로 사용할 열과 정렬 방향을 모두 설명하는 테이블의 행을 정렬하는 방법을 정의합니다.
SSortOrderSet

사용하지 마세요. 표준 또는 분류된 정렬에 사용할 테이블의 키 컬렉션을 정의합니다.
WAB_PARAM 사용하지 마세요. WABOpen에 전달할 입력 정보를 포함합니다.
WABEXTDISPLAY 사용하지 마세요. WAB(Windows 주소록)에서 사용자의 IContextMenu 인터페이스 및 IShellPropSheetExt 인터페이스 구현을 초기화하는 데 사용됩니다.
WABIMPORTPARAM 사용하지 마세요. .wab 파일 가져오기에 대한 정보를 제공하는 가져오기에 전달된 구조체입니다.

Last updated on 2025. 10. 08.

Windows 및 메시지

아티클2023. 03. 16.

Windows 및 메시지 기술 개요입니다.

Windows 및 메시지 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 및 메시지

Functions

AdjustWindowRect 원하는 클라이언트 사각형 크기에 따라 창 사각형의 필요한 크기를 계산합니다. 그런 다음 창 사각형을 CreateWindow 함수에 전달하여 클라이언트 영역이 원하는 크기인 창을 만들 수 있습니다.
AdjustWindowRectEx 클라이언트 사각형의 원하는 크기에 따라 창 사각형의 필요한 크기를 계산합니다. 그런 다음 창 사각형을 CreateWindowEx 함수에 전달하여 클라이언트 영역이 원하는 크기인 창을 만들 수 있습니다.
AllowSetForegroundWindow SetForegroundWindow 함수를 사용하여 지정된 프로세스가 포그라운드 창을 설정할 수 있도록 합니다. 호출 프로세스는 이미 포그라운드 창을 설정할 수 있어야 합니다. 자세한 내용은 이 항목의 뒷부분에 나오는 주의를 참조하십시오.
AnimateWindow 창을 표시하거나 숨길 때 특수 효과를 생성할 수 있습니다. 애니메이션에는 _roll, 슬라이드, 축소 또는 확장, 알파 혼합 페이드의 네 가지 유형이 있습니다.
AnyPopup 소유, 표시, 최상위 팝업 또는 겹치는 창이 화면에 있는지 여부를 나타냅니다. 함수는 호출 애플리케이션의 클라이언트 영역뿐만 아니라 전체 화면을 검색합니다.
ArrangeIconicWindows 지정된 부모 창의 최소화된(상징적인) 자식 창을 모두 정렬합니다.

BeginDeferWindowPos 다중 창 위치 구조에 대한 메모리를 할당하고 구조체에 핸들을 반환합니다.
BringWindowToTop 지정된 창을 Z 순서의 맨 위로 가져옵니다. 창이 최상위 창이면 활성화됩니다. 창이 자식 창인 경우 자식 창과 연결된 최상위 부모 창이 활성화됩니다.
BroadcastSystemMessage 지정된 받는 사람에게 메시지를 보냅니다. (BroadcastSystemMessageW)
BroadcastSystemMessageA 지정된 받는 사람에게 메시지를 보냅니다. (BroadcastSystemMessageA)
BroadcastSystemMessageExA 지정된 받는 사람에게 메시지를 보냅니다. (BroadcastSystemMessageExA)
BroadcastSystemMessageExW 지정된 받는 사람에게 메시지를 보냅니다. (BroadcastSystemMessageExW)
BroadcastSystemMessageW 지정된 받는 사람에게 메시지를 보냅니다. (BroadcastSystemMessageW)
CalculatePopupWindowPosition 지정된 앵커 지점, 팝업 창 크기, 플래그 및 선택적 제외 사각형을 사용하여 적절한 팝업 창 위치를 계산합니다.
CallMsgFilterA 지정된 메시지 및 후크 코드를 WH_SYSMSGFILTER 및 WH_MSGFILTER 후크와 연결된 후크 프로시저에 전달합니다. (ANSI)
CallMsgFilterW 지정된 메시지 및 후크 코드를 WH_SYSMSGFILTER 및 WH_MSGFILTER 후크와 연결된 후크 프로시저에 전달합니다. (유니코드)
CallNextHookEx 후크 정보를 현재 후크 체인의 다음 후크 프로시저에 전달합니다. 후크 프로시저는 후크 정보를 처리하기 전이나 후에 이 함수를 호출할 수 있습니다.

CallWindowProcA 지정된 창 프로시저에 메시지 정보를 전달합니다. (ANSI)
CallWindowProcW 지정된 창 프로시저에 메시지 정보를 전달합니다. (유니코드)
CascadeWindows 지정된 부모 창의 지정된 자식 창을 계단식으로 배열합니다.
ChangeWindowMessageFilter UIPI(사용자 인터페이스 권한 격리) 메시지 필터에서 메시지를 추가하거나 제거합니다.
ChangeWindowMessageFilterEx 지정된 창에 대한 UIPI(사용자 인터페이스 권한 격리) 메시지 필터를 수정합니다.
ChildWindowFromPoint 부모 창에 속한 자식 창에 지정된 점을 포함하는 자식 창이 있는 경우 결정합니다. 검색은 바로 자식 창으로 제한됩니다. 손자 손녀와 더 깊은 하위 창은 검색되지 않습니다.
ChildWindowFromPointEx 지정된 부모 창에 속한 자식 창에 지정된 점이 포함된 자식 창이 있는 경우 결정합니다.
CloseWindow 지정된 창을 최소화(하지만 삭제하지는 않음).
CreateMDIWindowA MDI(다중 문서 인터페이스) 자식 창을 만듭니다. (ANSI)
CreateMDIWindowW MDI(다중 문서 인터페이스) 자식 창을 만듭니다. (유니코드)
CreateWindowA 겹치는, 팝업 또는 자식 창을 만듭니다. (ANSI)
CreateWindowExA 확장 창 스타일을 사용하여 겹치는, 팝업 또는 자식 창을 만듭니다. 그렇지 않으면 이 함수는 CreateWindow 함수와 동일합니다. (ANSI)

CreateWindowExW 확장 창 스타일을 사용하여 겹치는, 팝업 또는 자식 창을 만듭니다. 그렇지 않으면 이 함수는 CreateWindow 함수와 동일합니다. (유니코드)
CreateWindowW 겹치는, 팝업 또는 자식 창을 만듭니다. (유니코드)
DeferWindowPos 지정된 창에 대해 지정된 다중 창 위치 구조를 업데이트.
DefFrameProcA MDI(다중 문서 인터페이스) 프레임 창의 창 프로시저가 처리되지 않는 모든 창 메시지에 대한 기본 처리를 제공합니다. (ANSI)
DefFrameProcW MDI(다중 문서 인터페이스) 프레임 창의 창 프로시저가 처리되지 않는 모든 창 메시지에 대한 기본 처리를 제공합니다. (유니코드)
DefMDIChildProcA MDI(다중 문서 인터페이스) 자식 창의 창 프로시저가 처리되지 않는 모든 창 메시지에 대한 기본 처리를 제공합니다. (ANSI)
DefMDIChildProcW MDI(다중 문서 인터페이스) 자식 창의 창 프로시저가 처리되지 않는 모든 창 메시지에 대한 기본 처리를 제공합니다. (유니코드)
DefWindowProcA 기본 창 프로시저를 호출하여 애플리케이션이 처리하지 않는 모든 창 메시지에 대한 기본 처리를 제공합니다. (ANSI)
DefWindowProcW 기본 창 프로시저를 호출하여 애플리케이션이 처리하지 않는 모든 창 메시지에 대한 기본 처리를 제공합니다. (유니코드)
DeregisterShellHookWindow 셸 후크 메시지를 수신하도록 등록된 지정된 셸 창을 등록 취소합니다.

DestroyWindow 지정된 창을 삭제합니다.
DispatchMessage 창 프로시저에 메시지를 디스패치합니다. 일반적으로 GetMessage 함수에서 검색한 메시지를 디스패치하는 데 사용됩니다. (DispatchMessageW)
DispatchMessageA 창 프로시저에 메시지를 디스패치합니다. 일반적으로 GetMessage 함수에서 검색한 메시지를 디스패치하는 데 사용됩니다. (DispatchMessageA)
DispatchMessageW 창 프로시저에 메시지를 디스패치합니다. 일반적으로 GetMessage 함수에서 검색한 메시지를 디스패치하는 데 사용됩니다. (DispatchMessageW)
EndDeferWindowPos 단일 화면 새로 고침 주기에서 하나 이상의 창의 위치와 크기를 동시에 업데이트합니다.
EndTask 지정된 창을 강제로 닫습니다.
EnumChildWindows 각 자식 창에 핸들을 차례로 애플리케이션 정의 콜백 함수에 전달하여 지정된 부모 창에 속하는 자식 창을 열거합니다.
EnumPropsA 지정된 콜백 함수에 하나씩 전달하여 창의 속성 목록에 있는 모든 항목을 열거합니다. EnumProps는 마지막 항목이 열거되거나 콜백 함수가 FALSE를 반환할 때까지 계속됩니다. (ANSI)
EnumPropsExA 지정된 콜백 함수에 하나씩 전달하여 창의 속성 목록에 있는 모든 항목을 열거합니다. EnumPropsEx는 마지막 항목이 열거되거나 콜백 함수가 FALSE를 반환할 때까지 계속됩니다. (ANSI)
EnumPropsExW 지정된 콜백 함수에 하나씩 전달하여 창의 속성 목록에 있는 모든 항목을 열거합니다. EnumPropsEx는 마지막 항목이 열거되거나 콜백 함수가 FALSE를 반환할 때까지 계속됩니다. (유니코드)

EnumPropsW 지정된 콜백 함수에 하나씩 전달하여 창의 속성 목록에 있는 모든 항목을 열거합니다. EnumProps는 마지막 항목이 열거되거나 콜백 함수가 FALSE를 반환할 때까지 계속됩니다. (유니코드)
EnumThreadWindows 각 창에 핸들을 차례로 애플리케이션 정의 콜백 함수에 전달하여 스레드와 연결된 모든 비자식 창을 열거합니다.
EnumWindows 각 창에 핸들을 차례로 애플리케이션 정의 콜백 함수에 전달하여 화면의 모든 최상위 창을 열거합니다. EnumWindows는 마지막 최상위 창이 열거되거나 콜백 함수가 FALSE를 반환할 때까지 계속됩니다.
FindWindowA 클래스 이름과 창 이름이 지정된 문자열과 일치하는 최상위 창에 대한 핸들을 검색합니다. 이 함수는 자식 창을 검색하지 않습니다. 이 함수는 대/소문자를 구분하는 검색을 수행하지 않습니다. (ANSI)
FindWindowExA 클래스 이름과 창 이름이 지정된 문자열과 일치하는 창에 대한 핸들을 검색합니다. 함수는 지정된 자식 창 다음에 있는 자식 창부터 시작하여 자식 창을 검색합니다. 이 함수는 대/소문자를 구분하는 검색을 수행하지 않습니다. (ANSI)
FindWindowExW 클래스 이름과 창 이름이 지정된 문자열과 일치하는 창에 대한 핸들을 검색합니다. 함수는 지정된 자식 창 다음에 있는 자식 창부터 시작하여 자식 창을 검색합니다. 이 함수는 대/소문자를 구분하는 검색을 수행하지 않습니다. (유니코드)
FindWindowW 클래스 이름과 창 이름이 지정된 문자열과 일치하는 최상위 창에 대한 핸들을 검색합니다. 이 함수는 자식 창을 검색하지 않습니다. 이 함수는 대/소문자를 구분하는 검색을 수행하지 않습니다. (유니코드)
GET_X_LPARAM 지정된 LPARAM 값에서 서명된 x 좌표를 검색합니다.
GET_Y_LPARAM 지정된 LPARAM 값에서 서명된 y 좌표를 검색합니다.

GetAltTabInfoA 지정된 창이 애플리케이션 전환(ALT+TAB) 창인 경우 상태 정보를 검색합니다. (ANSI)
GetAltTabInfoW 지정된 창이 애플리케이션 전환(ALT+TAB) 창인 경우 상태 정보를 검색합니다. (유니코드)
GetAncestor 지정된 창의 상위 항목에 대한 핸들을 검색합니다.
GetClassInfoA 창 클래스에 대한 정보를 검색합니다. (ANSI)
GetClassInfoExA 창 클래스와 연결된 작은 아이콘에 대한 핸들을 포함하여 창 클래스에 대한 정보를 검색합니다. GetClassInfo 함수는 작은 아이콘에 대한 핸들을 검색하지 않습니다. (ANSI)
GetClassInfoExW 창 클래스와 연결된 작은 아이콘에 대한 핸들을 포함하여 창 클래스에 대한 정보를 검색합니다. GetClassInfo 함수는 작은 아이콘에 대한 핸들을 검색하지 않습니다. (유니코드)
GetClassInfoW 창 클래스에 대한 정보를 검색합니다. (유니코드)
GetClassLongA 지정된 창과 연결된 WNDCLASSEX 구조체에서 지정된 32비트(DWORD) 값을 검색합니다. (ANSI)
GetClassLongPtrA 지정된 창과 연결된 WNDCLASSEX 구조체에서 지정된 값을 검색합니다. (ANSI)
GetClassLongPtrW 지정된 창과 연결된 WNDCLASSEX 구조체에서 지정된 값을 검색합니다. (유니코드)
GetClassLongW 지정된 창과 연결된 WNDCLASSEX 구조체에서 지정된 32비트(DWORD) 값을 검색합니다. (유니코드)

GetClassName 지정된 창이 속한 클래스의 이름을 검색합니다. (GetClassNameW)
GetClassNameA 지정된 창이 속한 클래스의 이름을 검색합니다. (GetClassNameA)
GetClassNameW 지정된 창이 속한 클래스의 이름을 검색합니다. (GetClassNameW)
GetClassWord 지정된 오프셋의 16비트(WORD) 값을 지정된 창이 속한 창 클래스의 추가 클래스 메모리로 검색합니다.
GetClientRect 창 클라이언트 영역의 좌표를 검색합니다.
GetDesktopWindow 바탕 화면 창에 대한 핸들을 검색합니다. 바탕 화면 창은 전체 화면을 덮습니다. 바탕 화면 창은 다른 창이 그려진 맨 위에 있는 영역입니다.
GetForegroundWindow 전경 창(사용자가 현재 작업 중인 창)에 대한 핸들을 검색합니다. 시스템은 다른 스레드보다 포그라운드 창을 만드는 스레드에 약간 더 높은 우선 순위를 할당합니다.
GetGUIThreadInfo 활성 창 또는 지정된 GUI 스레드에 대한 정보를 검색합니다.
GetInputState 호출 스레드의 메시지 큐에 마우스 단추 또는 키보드 메시지가 있는지 여부를 결정합니다.
GetLastActivePopup 지정된 창이 소유한 팝업 창이 가장 최근에 활성화되었는지 확인합니다.
GetLayeredWindowAttributes 계층적 창의 불투명도 및 투명도 색상 키를 검색합니다.

GetMessage 호출 스레드의 메시지 큐에서 메시지를 검색합니다. 함수는 게시된 메시지를 검색할 수 있게 될 때까지 들어오는 보낸 메시지를 디스패치합니다. (GetMessageW)
GetMessageA 호출 스레드의 메시지 큐에서 메시지를 검색합니다. 함수는 게시된 메시지를 검색할 수 있게 될 때까지 들어오는 보낸 메시지를 디스패치합니다. (GetMessageA)
GetMessageExtraInfo 현재 스레드에 대한 추가 메시지 정보를 검색합니다. 추가 메시지 정보는 현재 스레드의 메시지 큐와 연결된 애플리케이션 또는 드라이버 정의 값입니다.
GetMessagePos GetMessage 함수에서 검색한 마지막 메시지의 커서 위치를 검색합니다.
GetMessageTime GetMessage 함수에서 검색한 마지막 메시지의 메시지 시간을 검색합니다.
GetMessageW 호출 스레드의 메시지 큐에서 메시지를 검색합니다. 함수는 게시된 메시지를 검색할 수 있게 될 때까지 들어오는 보낸 메시지를 디스패치합니다. (GetMessageW)
GetNextWindow Z-Order의 다음 또는 이전 창에 대한 핸들을 검색합니다. 다음 창은 지정된 창 아래에 있습니다. 이전 창이 위에 있습니다.
GetParent 지정된 창의 부모 또는 소유자에 대한 핸들을 검색합니다.
GetProcessDefaultLayout 부모 또는 소유자 없이 창을 만들 때 사용되는 기본 레이아웃을 검색합니다.
GetPropA 지정된 창의 속성 목록에서 데이터 핸들을 검색합니다. 문자열은 검색할 핸들을 식별합니다. SetProp 함수에 대한 이전 호출을 통해 문자열과 핸들을 속성 목록에 추가해야 합니다. (ANSI)

GetPropW 지정된 창의 속성 목록에서 데이터 핸들을 검색합니다. 문자열은 검색할 핸들을 식별합니다. SetProp 함수에 대한 이전 호출을 통해 문자열과 핸들을 속성 목록에 추가해야 합니다. (유니코드)
GetQueueStatus 호출 스레드의 메시지 큐에 있는 메시지 유형을 검색합니다.
GetShellWindow 셸의 바탕 화면 창에 대한 핸들을 검색합니다.
GetSysColor 지정된 표시 요소의 현재 색을 검색합니다.
GetSystemMetrics 지정된 시스템 메트릭 또는 시스템 구성 설정을 검색합니다.
GetTitleBarInfo 지정된 제목 표시줄에 대한 정보를 검색합니다.
GetTopWindow 지정된 부모 창과 연결된 자식 창의 Z 순서를 검사하고 Z 순서 맨 위에 있는 자식 창에 대한 핸들을 검색합니다.
GetWindow 지정된 창에 대한 지정한 관계(Z-Order 또는 소유자)가 있는 창에 대한 핸들을 검색합니다.
GetWindowDisplayAffinity 지정된 창에 대한 모든 프로세스에서 현재 표시 선호도 설정을 검색합니다.
GetWindowInfo 지정된 창에 대한 정보를 검색합니다. (GetWindowInfo)
GetWindowLongA 지정된 창에 대한 정보를 검색합니다. (GetWindowLongA)

GetWindowLongPtrA 지정된 창에 대한 정보를 검색합니다. 또한 함수는 지정된 오프셋의 값을 추가 창 메모리로 검색합니다. (ANSI)
GetWindowLongPtrW 지정된 창에 대한 정보를 검색합니다. 또한 함수는 지정된 오프셋의 값을 추가 창 메모리로 검색합니다. (유니코드)
GetWindowLongW 지정된 창에 대한 정보를 검색합니다. (GetWindowLongW)
GetWindowModuleFileNameA 지정된 창 핸들과 연결된 모듈의 전체 경로 및 파일 이름을 검색합니다. (ANSI)
GetWindowModuleFileNameW 지정된 창 핸들과 연결된 모듈의 전체 경로 및 파일 이름을 검색합니다. (유니코드)
GetWindowPlacement 표시 상태와 지정된 창의 복원, 최소화 및 최대화된 위치를 검색합니다.
GetWindowRect 지정된 창의 경계 사각형 크기를 검색합니다. 차원은 화면의 왼쪽 위 모서리를 기준으로 하는 화면 좌표로 제공됩니다.
GetWindowTextA 지정된 창의 제목 표시줄 텍스트(있는 경우)를 버퍼에 복사합니다. 지정한 창이 컨트롤이면 컨트롤의 텍스트가 복사됩니다. 그러나 GetWindowText는 다른 애플리케이션에서 컨트롤의 텍스트를 검색할 수 없습니다. (ANSI)
GetWindowTextLengthA 창에 제목 표시줄이 있는 경우 지정된 창의 제목 표시줄 텍스트의 길이를 문자 단위로 검색합니다. (ANSI)
GetWindowTextLengthW 창에 제목 표시줄이 있는 경우 지정된 창의 제목 표시줄 텍스트의 길이를 문자 단위로 검색합니다. (유니코드)

GetWindowTextW 지정된 창의 제목 표시줄 텍스트(있는 경우)를 버퍼에 복사합니다. 지정한 창이 컨트롤이면 컨트롤의 텍스트가 복사됩니다. 그러나 GetWindowText는 다른 애플리케이션에서 컨트롤의 텍스트를 검색할 수 없습니다. (유니코드)
GetWindowThreadProcessId 지정된 창을 만든 스레드의 식별자와 필요에 따라 창을 만든 프로세스의 식별자를 검색합니다.
GetWindowWord 지정된 오프셋의 16비트(**DWORD**) 값을 추가 창 메모로 검색합니다.
HOOKPROC SetWindowsHookEx 함수와 함께 사용되는 애플리케이션 정의 또는 라이브러리 정의 콜백 함수입니다. 시스템은 SendMessage 함수가 호출된 후 이 함수를 호출합니다. 후크 프로시저는 메시지를 검사할 수 있습니다. 수정할 수 없습니다.
InSendMessage 현재 창 프로시저가 SendMessage 함수를 호출하여 다른 스레드(동일한 프로세스 또는 다른 프로세스)에서 보낸 메시지를 처리하고 있는지 여부를 확인합니다.
InSendMessageEx 현재 창 프로시저가 다른 스레드(동일한 프로세스 또는 다른 프로세스)에서 보낸 메시지를 처리하고 있는지 여부를 확인합니다.
InternalGetWindowText 지정된 창의 제목 표시줄 텍스트(있는 경우)를 버퍼에 복사합니다.
IsChild 창이 지정된 부모 창의 자식 창인지 하위 창인지 여부를 결정합니다.
IsGUIThread 호출 스레드가 이미 GUI 스레드인지 여부를 확인합니다. 필요에 따라 스레드를 GUI 스레드로 변환할 수도 있습니다.
IsHungAppWindow 시스템에서 지정된 애플리케이션이 응답하지 않는 것으로 간주하는지 여부를 결정합니다.

IsIconic 지정된 창이 최소화되는지 여부를 결정합니다(아이콘).
IsProcessDPIAware IsProcessDPIAware를 변경하거나 사용할 수 없습니다. 대신 GetProcessDPIAwareness를 사용합니다.
IsWindow 지정된 창 핸들이 기존 창을 식별하는지 여부를 확인합니다.
IsWindowUnicode 지정된 창이 네이티브 유니코드 창인지 여부를 확인합니다.
IsWindowVisible 지정된 창의 표시 여부를 결정합니다.
IsZoomed 창이 최대화되는지 여부를 결정합니다.
KillTimer 지정된 타이머를 삭제합니다.
LockSetForegroundWindow 포그라운드 프로세스는 LockSetForegroundWindow 함수를 호출하여 SetForegroundWindow 함수에 대한 호출을 사용하지 않도록 설정할 수 있습니다.
LogicalToPhysicalPoint 창에 있는 점의 논리적 좌표를 실제 좌표로 변환합니다.
MAKELPARAM 메시지에서 lParam 매개 변수로 사용할 값을 만듭니다. 매크로는 지정된 값을 연결합니다.
MAKELRESULT 창 프로시저에서 반환 값으로 사용할 값을 만듭니다. 매크로는 지정된 값을 연결합니다.
MAKEWPARAM 메시지에서 wParam 매개 변수로 사용할 값을 만듭니다. 매크로는 지정된 값을 연결합니다.

MoveWindow 지정된 창의 위치와 크기를 변경합니다.
OpenIcon 최소화된(상징적) 창을 이전 크기와 위치로 복원합니다. 그러면 창이 활성화됩니다.
PeekMessageA 큐에 추가되지 않은 들어오는 메시지를 디스패치하고, 게시된 메시지에 대한 스레드 메시지 큐를 확인하고, 메시지를 검색합니다(있는 경우). (ANSI)
PeekMessageW 큐에 추가되지 않은 들어오는 메시지를 디스패치하고, 게시된 메시지에 대한 스레드 메시지 큐를 확인하고, 메시지를 검색합니다(있는 경우). (유니코드)
PhysicalToLogicalPoint 창에 있는 점의 실제 좌표를 논리 좌표로 변환합니다.
PostMessageA 지정된 창을 만든 스레드와 연결된 메시지 큐에 메시지를 배치(게시)하고 스레드가 메시지를 처리할 때까지 기다리지 않고 반환합니다. (ANSI)
PostMessageW 지정된 창을 만든 스레드와 연결된 메시지 큐에 메시지를 배치(게시)하고 스레드가 메시지를 처리할 때까지 기다리지 않고 반환합니다. (유니코드)
PostQuitMessage 스레드가 종료(종료)를 요청했음을 시스템에 나타냅니다. 일반적으로 WM_DESTROY 메시지에 대한 응답으로 사용됩니다.
PostThreadMessageA 지정된 스레드의 메시지 큐에 메시지를 게시합니다. 스레드가 메시지를 처리할 때까지 기다리지 않고 반환됩니다. (ANSI)
PostThreadMessageW 지정된 스레드의 메시지 큐에 메시지를 게시합니다. 스레드가 메시지를 처리할 때까지 기다리지 않고 반환됩니다. (유니코드)

PROPENUMPROCA EnumProps 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (ANSI)
PROPENUMPROCEXA EnumPropsEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (ANSI)
PROPENUMPROCEXW EnumPropsEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (유니코드)
PROPENUMPROCW EnumProps 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (유니코드)
RealChildWindowFromPoint 지정된 지점에서 자식 창에 대한 핸들을 검색합니다. 검색은 바로 자식 창으로 제한됩니다. 손자와 더 깊은 하위 창은 검색되지 않습니다.
RealGetWindowClassA 창 형식을 지정하는 문자열을 검색합니다. (ANSI)
RealGetWindowClassW 창 형식을 지정하는 문자열을 검색합니다. (유니코드)
RegisterClassA CreateWindow 또는 CreateWindowEx 함수에 대한 호출에서 후속 사용을 위해 창 클래스를 등록합니다. (RegisterClassA)
RegisterClassExA CreateWindow 또는 CreateWindowEx 함수에 대한 호출에서 후속 사용을 위해 창 클래스를 등록합니다. (RegisterClassExA)
RegisterClassExW CreateWindow 또는 CreateWindowEx 함수에 대한 호출에서 후속 사용을 위해 창 클래스를 등록합니다. (RegisterClassExW)
RegisterClassW CreateWindow 또는 CreateWindowEx 함수에 대한 호출에서 후속 사용을 위해 창 클래스를 등록합니다. (RegisterClassW)

RegisterShellHookWindow 지정된 셸 창을 등록하여 셸 애플리케이션에 유용한 이벤트 또는 알림에 대한 특정 메시지를 받습니다.
RegisterWindowMessageA 시스템 전체에서 고유하도록 보장되는 새 창 메시지를 정의합니다. 메시지 값은 메시지를 보내거나 게시할 때 사용할 수 있습니다. (ANSI)
RegisterWindowMessageW 시스템 전체에서 고유하도록 보장되는 새 창 메시지를 정의합니다. 메시지 값은 메시지를 보내거나 게시할 때 사용할 수 있습니다. (유니코드)
RemovePropA 지정된 창의 속성 목록에서 항목을 제거합니다. 지정된 문자열은 제거할 항목을 식별합니다. (ANSI)
RemovePropW 지정된 창의 속성 목록에서 항목을 제거합니다. 지정된 문자열은 제거할 항목을 식별합니다. (유니코드)
ReplyMessage SendMessage 함수가 다른 스레드에서 보낸 메시지에 회신합니다.
SENDASYNCPROC SendMessageCallback 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
SendMessage 지정된 메시지를 창 또는 창으로 보냅니다. SendMessage 함수는 지정된 창에 대한 창 프로시저를 호출하고 창 프로시저가 메시지를 처리할 때까지 반환되지 않습니다. (SendMessageW)
SendMessageA 지정된 메시지를 창 또는 창으로 보냅니다. SendMessage 함수는 지정된 창에 대한 창 프로시저를 호출하고 창 프로시저가 메시지를 처리할 때까지 반환되지 않습니다. (SendMessageA)
SendMessageCallbackA 지정된 메시지를 창 또는 창으로 보냅니다. (SendMessageCallbackA)
SendMessageCallbackW 지정된 메시지를 창 또는 창으로 보냅니다. (SendMessageCallbackW)

SendMessageTimeoutA 지정된 메시지를 하나 이상의 창으로 보냅니다. (ANSI)
SendMessageTimeoutW 지정된 메시지를 하나 이상의 창으로 보냅니다. (유니코드)
SendMessageW 지정된 메시지를 창 또는 창으로 보냅니다. SendMessage 함수는 지정된 창에 대한 창 프로시저를 호출하고 창 프로시저가 메시지를 처리할 때까지 반환되지 않습니다. (SendMessageW)
SendNotifyMessageA 지정된 메시지를 창 또는 창으로 보냅니다. (SendNotifyMessageA)
SendNotifyMessageW 지정된 메시지를 창 또는 창으로 보냅니다. (SendNotifyMessageW)
SetAdditionalForegroundBoostProcesses SetAdditionalForegroundBoostProcesses는 여러 프로세스가 데이터 또는 렌더링으로 포그라운드 환경에 기여하는 다중 프로세스 애플리케이션 모델을 사용하는 애플리케이션을 지원하는 성능 지원 API입니다.
SetClassLongA 지정된 오프셋의 지정된 32비트(long) 값을 지정된 창이 속한 클래스의 추가 클래스 메모리 또는 WNDCLASSEX 구조체로 바꿉니다. (ANSI)
SetClassLongPtrA 지정한 창이 속한 클래스의 추가 클래스 메모리 또는 WNDCLASSEX 구조체에서 지정된 오프셋의 지정된 값을 바꿉니다. (ANSI)
SetClassLongPtrW 지정한 창이 속한 클래스의 추가 클래스 메모리 또는 WNDCLASSEX 구조체에서 지정된 오프셋의 지정된 값을 바꿉니다. (유니코드)
SetClassLongW 지정된 오프셋의 지정된 32비트(long) 값을 지정된 창이 속한 클래스의 추가 클래스 메모리 또는 WNDCLASSEX 구조체로 바꿉니다. (유니코드)

SetClassWord 지정된 오프셋의 16비트(WORD) 값을 지정된 창이 속한 창 클래스의 추가 클래스 메모리로 바꿉니다.
SetCoalescableTimer 지정된 제한 시간 값과 병합 허용 오차 지연을 사용하여 타이머를 만듭니다.
SetForegroundWindow 지정된 창을 만든 스레드를 포그라운드로 가져오고 창을 활성화합니다.
SetLayeredWindowAttributes 계층적 창의 불투명도 및 투명도 색상 키를 설정합니다.
SetMessageExtraInfo 현재 스레드에 대한 추가 메시지 정보를 설정합니다.
SetParent 지정된 자식 창의 부모 창을 변경합니다.
SetProcessDefaultLayout 현재 실행 중인 프로세스에 대해서만 부모 또는 소유자 없이 창을 만들 때 기본 레이아웃을 변경합니다.
SetProcessDPIAware SetProcessDPIAware를 변경하거나 사용할 수 없습니다. 대신 SetProcessDPIAwareness를 사용합니다.
SetPropA 새 항목을 추가하거나 지정된 창의 속성 목록에서 기존 항목을 변경합니다. (ANSI)
SetPropW 새 항목을 추가하거나 지정된 창의 속성 목록에서 기존 항목을 변경합니다. (유니코드)
SetSysColors 지정된 표시 요소의 색을 설정합니다.

SetTimer 지정된 시간 제한 값을 사용하여 타이머를 만듭니다.
SetWindowDisplayAffinity 디스플레이 선호도 설정을 창과 연결된 hWnd의 커널 모드에 저장합니다.
SetWindowLongA 지정된 창의 특성을 변경합니다. 또한 함수는 지정된 오프셋의 32비트(long) 값을 추가 창 메모리로 설정합니다. (ANSI)
SetWindowLongPtrA 지정된 창의 특성을 변경합니다. (ANSI)
SetWindowLongPtrW 지정된 창의 특성을 변경합니다. (유니코드)
SetWindowLongW 지정된 창의 특성을 변경합니다. 또한 함수는 지정된 오프셋의 32비트(long) 값을 추가 창 메모리로 설정합니다. (유니코드)
SetWindowPlacement 표시 상태와 지정된 창의 복원, 최소화 및 최대화된 위치를 설정합니다.
SetWindowPos 자식, 팝업 또는 최상위 창의 크기, 위치 및 Z 순서를 변경합니다. 이러한 창은 화면의 모양에 따라 정렬됩니다. 맨 위 창은 가장 높은 순위를 받고 Z 순서의 첫 번째 창입니다.
SetWindowsHookExA 애플리케이션 정의 후크 프로시저를 후크 체인에 설치합니다. (ANSI)
SetWindowsHookExW 애플리케이션 정의 후크 프로시저를 후크 체인에 설치합니다. (유니코드)
SetWindowTextA 지정된 창 제목 표시줄의 텍스트를 변경합니다(있는 경우). 지정한 창이 컨트롤이면 컨트롤의 텍스트가 변경됩니다. 그러나 SetWindowText는 다른 애플리케이션에서 컨트롤의 텍스트를 변경할 수 없습니다. (ANSI)

SetWindowTextW 지정된 창 제목 표시줄의 텍스트를 변경합니다(있는 경우). 지정한 창이 컨트롤이면 컨트롤의 텍스트가 변경됩니다. 그러나 SetWindowText는 다른 애플리케이션에서 컨트롤의 텍스트를 변경할 수 없습니다. (유니코드)
ShowOwnedPopups 지정된 창이 소유한 모든 팝업 창을 표시하거나 숨깁니다.
Showwindow 지정된 창의 표시 상태를 설정합니다.
ShowWindowAsync 작업이 완료 될 때까지 기다리지 않고 창의 표시 상태를 설정합니다.
SoundSentry 시각적 신호를 트리거하여 소리가 재생되고 있음을 나타냅니다.
SwitchToThisWindow 포커스를 지정된 창으로 전환하여 포그라운드로 가져옵니다.
SystemParametersInfoA 시스템 전체 매개 변수 중 하나의 값을 검색하거나 설정합니다. (ANSI)
SystemParametersInfoW 시스템 전체 매개 변수 중 하나의 값을 검색하거나 설정합니다. (유니코드)
TileWindows 지정된 부모 창의 지정된 자식 창을 타일로 표시합니다.
TIMERPROC WM_TIMER 메시지를 처리하는 애플리케이션 정의 콜백 함수입니다. TIMERPROC 형식은 이 콜백 함수에 대한 포인터를 정의합니다. TimerProc은 애플리케이션 정의 함수 이름의 자리 표시자입니다.
TranslateMDISysAccel 지정된 MDI 클라이언트 창과 연결된 MDI(다중 문서 인터페이스) 자식 창의 창 메뉴 명령에 대한 가속기 키 입력을 처리합니다.

TranslateMessage 가상 키 메시지를 문자 메시지로 변환합니다. 문자 메시지는 다음에 스레드가 GetMessage 또는 PeekMessage 함수를 호출할 때 읽을 호출 스레드의 메시지 큐에 게시됩니다.
UnhookWindowsHookEx SetWindowsHookEx 함수에 의해 후크 체인에 설치된 후크 프로시저를 제거합니다.
UnregisterClassA 창 클래스의 등록을 취소하여 클래스에 필요한 메모리를 해제합니다. (ANSI)
UnregisterClassW 창 클래스의 등록을 취소하여 클래스에 필요한 메모리를 해제합니다. (유니코드)
UpdateLayeredWindow 계층적 창의 위치, 크기, 모양, 내용 및 투명도를 업데이트합니다.
WaitMessage 스레드에 메시지 큐에 다른 메시지가 없는 경우 다른 스레드에 대한 제어를 생성합니다. WaitMessage 함수는 스레드를 일시 중단하고 새 메시지가 스레드의 메시지 큐에 배치될 때까지 반환되지 않습니다.
WindowFromPhysicalPoint 지정된 물리적 지점이 포함된 창에 대한 핸들을 검색합니다.
WindowFromPoint 지정된 점을 포함하는 창에 대한 핸들을 검색합니다.
WinMain 그래픽 Windows 기반 애플리케이션에 대한 사용자가 제공한 진입점입니다.
Wndproc 애플리케이션에서 정의하는 콜백 함수는 창으로 전송된 메시지를 처리합니다.

구조체

ALTTABINFO 애플리케이션 전환(ALT+TAB) 창에 대한 상태 정보를 포함합니다.
ANIMATIONINFO 사용자 작업과 관련된 애니메이션 효과에 대해 설명합니다.
AUDIODESCRIPTION 오디오 설명과 관련된 정보를 포함합니다. 이 구조체는 SPI_GETAUDIODESCRIPTION 또는 SPI_SETAUDIODESCRIPTION 작업 값을 지정할 때 SystemParametersInfo 함수와 함께 사용됩니다.
BSMINFO BroadcastSystemMessageEx의 요청을 거부한 창에 대한 정보를 포함합니다.
CBT_CREATEWNDA 창을 만들기 전에 WH_CBT 후크 프로시저 CBTProc에 전달된 정보를 포함합니다. (ANSI)
CBT_CREATEWNDW 창을 만들기 전에 WH_CBT 후크 프로시저 CBTProc에 전달된 정보를 포함합니다. (유니코드)
CBTACTIVATESTRUCT 창이 활성화되기 전에 WH_CBT 후크 프로시저 CBTProc에 전달된 정보를 포함합니다.
CHANGEFILTERSTRUCT ChangeWindowMessageFilterEx 함수를 호출하여 얻은 확장된 결과 정보를 포함합니다.
CLIENTCREATESTRUCT MDI 클라이언트 창의 메뉴 및 첫 번째 MDI(다중 문서 인터페이스) 자식 창에 대한 정보를 포함합니다.
CREATESTRUCTA 애플리케이션의 창 프로시저에 전달되는 초기화 매개 변수를 정의합니다. 이러한 멤버는 CreateWindowEx 함수의 매개 변수와 동일합니다. (ANSI)
CREATESTRUCTW 애플리케이션의 창 프로시저에 전달되는 초기화 매개 변수를 정의합니다. 이러한 멤버는 CreateWindowEx 함수의 매개 변수와 동일합니다. (유니코드)

CWPRETSTRUCT WH_CALLWNDPROCRET 후크 프로시저 CallWndRetProc에 전달되는 메시지 매개 변수를 정의합니다.
CWPSTRUCT WH_CALLWNDPROC 후크 프로시저 CallWndProc에 전달되는 메시지 매개 변수를 정의합니다.
DEBUGHOOKINFO WH_DEBUG 후크 프로시저인 DebugProc에 전달되는 디버깅 정보를 포함합니다.
EVENTMSG 시스템 메시지 큐로 전송된 하드웨어 메시지에 대한 정보를 포함합니다. 이 구조는 JournalPlaybackProc 콜백 함수에 대한 메시지 정보를 저장하는 데 사용됩니다.
GUITHREADINFO GUI 스레드에 대한 정보를 포함합니다.
KBDLLHOOKSTRUCT 하위 수준 키보드 입력 이벤트에 대한 정보를 포함합니다.
MDICREATESTRUCTA MDI(다중 문서 인터페이스) 자식 창의 클래스, 제목, 소유자, 위치 및 크기에 대한 정보를 포함합니다. (ANSI)
MDICREATESTRUCTW MDI(다중 문서 인터페이스) 자식 창의 클래스, 제목, 소유자, 위치 및 크기에 대한 정보를 포함합니다. (유니코드)
MINIMIZEDMETRICS 최소화된 창과 연결된 확장 가능한 메트릭을 포함합니다.
MINMAXINFO 창의 최대 크기 및 위치와 최소 및 최대 추적 크기에 대한 정보를 포함합니다.
MOUSEHOOKSTRUCT WH_MOUSE 후크 프로시저 MouseProc에 전달된 마우스 이벤트에 대한 정보를 포함합니다.

MOUSEHOOKSTRUCTEX WH_MOUSE 후크 프로시저 MouseProc에 전달된 마우스 이벤트에 대한 정보를 포함합니다. 휠 이동 또는 X 단추 사용에 대한 정보를 포함하는 MOUSEHOOKSTRUCT 구조체의 확장입니다.
메시지 스레드 메시지 큐의 메시지 정보를 포함합니다.
MSLLHOOKSTRUCT 하위 수준 마우스 입력 이벤트에 대한 정보를 포함합니다.
NCCALCSIZE_PARAMS 애플리케이션이 WM_NCCALCSIZE 메시지를 처리하는 동안 창의 클라이언트 영역의 크기, 위치 및 유효한 콘텐츠를 계산하는 데 사용할 수 있는 정보를 포함합니다.
NONCLIENTMETRICSA 축소되지 않은 창의 비클라이언트 영역과 연결된 확장 가능한 메트릭을 포함합니다. (ANSI)
NONCLIENTMETRICSW 축소되지 않은 창의 비클라이언트 영역과 연결된 확장 가능한 메트릭을 포함합니다. (유니코드)
STYLESTRUCT 창에 대한 스타일을 포함합니다.
TITLEBARINFO 제목 표시줄 정보를 포함합니다.
TITLEBARINFOEX 제목 표시줄의 각 요소의 좌표를 포함하여 TITLEBARINFO 구조에 설명된 정보를 확장합니다.
UPDATELAYEREDWINDOWINFO UpdateLayeredWindowIndirect에서 계층화된 창에 대한 위치, 크기, 모양, 콘텐츠 및 반투명 정보를 제공하는 데 사용됩니다.
WINDOWINFO 창 정보를 포함합니다.

창 위치 화면에 창의 배치에 대한 정보를 포함합니다.
WINDOWPOS 창의 크기와 위치에 대한 정보를 포함합니다.
WNDCLASSA RegisterClass 함수에 의해 등록된 창 클래스 특성을 포함합니다. (ANSI)
WNDCLASSEXA 창 클래스 정보를 포함합니다. (ANSI)
WNDCLASSEXW 창 클래스 정보를 포함합니다. (유니코드)
WNDCLASSW RegisterClass 함수에 의해 등록된 창 클래스 특성을 포함합니다. (유니코드)

Windows 애니메이션 관리자

아티클2023. 03. 11.

Windows 애니메이션 관리자 기술 개요입니다.

Windows 애니메이션 관리자를 개발하려면 다음 헤더가 필요합니다.

- uianimation.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 애니메이션 관리자

열거형

UI_ANIMATION_DEPENDENCIES 지정된 입력에 따라 보간기의 측면을 정의합니다.
UI_ANIMATION_IDLE_BEHAVIOR 애니메이션 관리자가 유휴 상태일 때 타이머의 동작을 정의합니다.
UI_ANIMATION_MANAGER_STATUS 애니메이션 관리자의 활동 상태를 정의합니다.
UI_ANIMATION_MODE 애니메이션 모드를 정의합니다.
UI_ANIMATION_PRIORITY_EFFECT 우선 순위 비교가 false를 반환하는 경우 스토리보드에 대한 잠재적 효과를 정의합니다.
UI_ANIMATION_REPEAT_MODE 루프 반복에 대한 패턴을 정의합니다.
UI_ANIMATION_ROUNDING_MODE 애니메이션 변수의 값이 부동 소수점 형식에서 정수 형식으로 변환될 때 사용할 반올림 모드를 정의합니다.

UI_ANIMATION_SCHEDULING_RESULT 스토리보드 일정에 대한 결과를 정의합니다.
UI_ANIMATION_SLOPE 애니메이션 기울기 특성을 정의합니다.
UI_ANIMATION_STORYBOARD_STATUS 스토리보드의 상태를 정의합니다.
UI_ANIMATION_TIMER_CLIENT_STATUS 타이머의 클라이언트에 대한 활동 상태를 정의합니다.
UI_ANIMATION_UPDATE_RESULT 애니메이션 업데이트에 대한 결과를 정의합니다.

Functions

포기 스토리보드를 종료하고, 관련된 모든 애니메이션 변수를 해제하고, 일정에서 스토리보드를 제거합니다. (IUIAnimationStoryboard.Abandon)
포기 스토리보드를 종료하고, 관련된 모든 애니메이션 변수를 해제하고, 일정에서 스토리보드를 제거합니다. (IUIAnimationStoryboard2.Abandon)
AbandonAllStoryboards 모든 활성 스토리보드를 중단합니다. (IUIAnimationManager.AbandonAllStoryboards)
AbandonAllStoryboards 모든 활성 스토리보드를 중단합니다. (IUIAnimationManager2.AbandonAllStoryboards)
AddCubic 전환 곡선의 모양을 설명하는 입방 다항식 세그먼트를 애니메이션 함수에 추가합니다.

AddKeyframeAfterTransition 지정된 전환이 끝날 때 키 프레임을 추가합니다. (IUIAnimationStoryboard.AddKeyframeAfterTransition)
AddKeyframeAfterTransition 지정된 전환이 끝날 때 키 프레임을 추가합니다. (IUIAnimationStoryboard2.AddKeyframeAfterTransition)
AddKeyframeAtOffset 기존 키 프레임에서 지정된 오프셋에 키 프레임을 추가합니다. (IUIAnimationStoryboard.AddKeyframeAtOffset)
AddKeyframeAtOffset 기존 키 프레임에서 지정된 오프셋에 키 프레임을 추가합니다. (IUIAnimationStoryboard2.AddKeyframeAtOffset)
AddSinusoidal 전환 곡선의 모양을 설명하는 부비동 세그먼트를 애니메이션 함수에 추가합니다.
AddTransition 스토리보드에 전환을 추가합니다. (IUIAnimationStoryboard.AddTransition)
AddTransition 스토리보드에 전환을 추가합니다. (IUIAnimationStoryboard2.AddTransition)
AddTransitionAtKeyframe 지정된 키 프레임에서 시작하는 전환을 추가합니다. (IUIAnimationStoryboard.AddTransitionAtKeyframe)
AddTransitionAtKeyframe 지정된 키 프레임에서 시작하는 전환을 추가합니다. (IUIAnimationStoryboard2.AddTransitionAtKeyframe)
AddTransitionBetweenKeyframes 두 키 프레임 간에 전환을 추가합니다. (IUIAnimationStoryboard.AddTransitionBetweenKeyframes)
AddTransitionBetweenKeyframes 두 키 프레임 간에 전환을 추가합니다. (IUIAnimationStoryboard2.AddTransitionBetweenKeyframes)

ClearTimerClientEventHandler 타이머 클라이언트 상태 변경 이벤트에 대한 처리기를 지웁니다.
결론 진행 중인 키 프레임 루프의 현재 반복을 완료하고(루프가 UI_ANIMATION_REPEAT_INDEFINITELY 설정된 경우), 루프를 종료하고 스토리보드를 계속 진행합니다. (IUIAnimationStoryboard.Conclude)
결론 진행 중인 키 프레임 루프의 현재 반복을 완료하고(루프가 UI_ANIMATION_REPEAT_INDEFINITELY 설정된 경우), 루프를 종료하고 스토리보드를 계속 진행합니다. (IUIAnimationStoryboard2.Conclude)
CreateAccelerateDecelerateTransition 가속 감속 전환을 만듭니다.
CreateAccelerateDecelerateTransition 가속 감속 스칼라 전환을 만듭니다.
CreateAnimationVariable 새 애니메이션 변수를 만듭니다. (IUIAnimationManager.CreateAnimationVariable)
CreateAnimationVariable 새 애니메이션 변수를 만듭니다. (IUIAnimationManager2.CreateAnimationVariable)
CreateAnimationVectorVariable 지정된 각 차원에 대한 새 애니메이션 변수를 만듭니다.
CreateConstantTransition 상수 전환을 만듭니다.
CreateConstantTransition 상수 스칼라 전환을 만듭니다.
CreateCubicBezierLinearTransition 입방형 3차원 선형 스칼라 전환을 만듭니다.

CreateCubicBezierLinearVectorTransition 지정된 각 차원에 대한 입방형 베지어 선형 벡터 전환을 만듭니다.
CreateCubicTransition 입방 전환을 만듭니다.
CreateCubicTransition 입방 스칼라 전환을 만듭니다.
CreateCubicVectorTransition 지정된 각 차원에 대한 입방 벡터 전환을 만듭니다.
CreateDiscreteTransition 불연속 전환을 만듭니다.
CreateDiscreteTransition 불연속 스칼라 전환을 만듭니다.
CreateDiscreteVectorTransition 지정된 각 차원에 대한 불연속 벡터 전환을 만듭니다.
CreateInstantaneousTransition 즉각적인 전환을 만듭니다.
CreateInstantaneousTransition 즉각적인 스칼라 전환을 만듭니다.
CreateInstantaneousVectorTransition 지정된 각 차원에 대해 즉각적인 벡터 전환을 만듭니다.
CreateLinearTransition 선형 전환을 만듭니다.
CreateLinearTransition 선형 스칼라 전환을 만듭니다.

CreateLinearTransitionFromSpeed 선형 속도 전환을 만듭니다.
CreateLinearTransitionFromSpeed 선형 속도 스칼라 전환을 만듭니다.
CreateLinearVectorTransition 지정된 차원에 선형 벡터 전환을 만듭니다.
CreateLinearVectorTransitionFromSpeed 지정된 차원에 선형 속도 벡터 전환을 만듭니다.
CreateParabolicTransitionFromAcceleration 포물선 가속 전환을 만듭니다.
CreateParabolicTransitionFromAcceleration 포물선 가속 스칼라 전환을 만듭니다.
CreateReversalTransition 반전 전환을 만듭니다.
CreateReversalTransition 반전 스칼라 전환을 만듭니다.
CreateSinusoidalTransitionFromRange 지정된 진동 범위를 사용하여 부비동 범위 전환을 만듭니다.
CreateSinusoidalTransitionFromRange 지정된 진동 범위를 사용하여 부비동 범위 스칼라 전환을 만듭니다.
CreateSinusoidalTransitionFromVelocity 초기 속도에 따라 진폭이 결정되는 부비동 속도 전환을 만듭니다.
CreateSinusoidalTransitionFromVelocity 진폭이 초기 속도에 따라 결정되는 부비동 스칼라 전환을 만듭니다.

CreateSmoothStopTransition 부드러운 중지 전환을 만듭니다.
CreateSmoothStopTransition 부드러운 중지 스칼라 전환을 만듭니다.
CreateStoryboard 새 스토리보드를 만듭니다. (IUIAnimationManager.CreateStoryboard)
CreateStoryboard 새 스토리보드를 만듭니다. (IUIAnimationManager2.CreateStoryboard)
CreateTransition 사용자 지정 보간기에서 전환을 만듭니다.
CreateTransition 지정된 차원에 대한 사용자 지정 보간기에서 전환을 만듭니다.
사용 안 함 애니메이션 타이머를 사용하지 않도록 설정합니다.
사용 애니메이션 타이머를 사용하도록 설정합니다.
EstimateNextEventTime 다음 애니메이션 이벤트 전 시간 간격의 예상값을 검색합니다.
마침 지정된 시간 내에 스토리보드를 완료하고 필요한 경우 스토리보드를 압축합니다. (IUIAnimationStoryboard.Finish)
마침 지정된 시간 내에 스토리보드를 완료하고 필요한 경우 스토리보드를 압축합니다. (IUIAnimationStoryboard2.Finish)

FinishAllStoryboards 지정된 시간 간격 내에 모든 활성 스토리보드를 완료합니다. (IUIAnimationManager.FinishAllStoryboards)
FinishAllStoryboards 지정된 시간 간격 내에 모든 활성 스토리보드를 완료합니다. (IUIAnimationManager2.FinishAllStoryboards)
GetCurrentStoryboard 현재 애니메이션 변수에 애니메이션 효과를 주는 스토리보드를 가져옵니다.
GetCurrentStoryboard 애니메이션 변수의 활성 스토리보드를 가져옵니다.
GetCurve 애니메이션 변수의 애니메이션 곡선을 가져옵니다.
GetDependencies SetInitialValueAndVelocity에 전달된 초기 값 또는 속도에 종속되거나 SetDuration에 전달된 기간에 따라 달라지는 보간기의 측면을 가져옵니다.
GetDependencies 지정된 차원의 경우 GetDependencies는 IUIAnimationInterpolator2::SetInitialValueAndVelocity 메서드에 전달되는 초기 값 또는 속도 또는 IUIAnimationInterpolator2::SetDuration 메서드에 전달되는 기간에 따라 보간기의 측면을 검색합니다.
GetDimension 보간이 필요한 차원 수를 가져옵니다.
GetDimension 애니메이션 변수에 전환이 지정된 차원의 수를 가져옵니다.
GetDimension 애니메이션 변수에 애니메이션 효과를 적용할 차원 수를 가져옵니다.
GetDuration 전환 기간을 가져옵니다.

GetDuration 지정된 차원의 전환 기간을 가져옵니다.
GetDuration 전환 기간을 가져옵니다. (IUIAnimationTransition.GetDuration)
GetDuration 전환 기간을 가져옵니다. (IUIAnimationTransition2.GetDuration)
GetElapsedTime 스토리보드가 재생되기 시작한 이후 경과된 시간을 가져옵니다. (IUIAnimationStoryboard.GetElapsedTime)
GetElapsedTime 스토리보드가 재생되기 시작한 이후 경과된 시간을 가져옵니다. (IUIAnimationStoryboard2.GetElapsedTime)
GetFinalIntegerValue 애니메이션 변수의 최종 값을 정수로 가져옵니다. 현재 예약된 모든 애니메이션이 완료된 후의 값입니다.
GetFinalIntegerValue 애니메이션 변수의 최종 정수 값을 가져옵니다. 현재 예약된 모든 애니메이션이 완료된 후의 값입니다.
GetFinalIntegerVectorValue 지정된 차원에 대한 애니메이션 변수의 최종 정수 값을 가져옵니다. 현재 예약된 모든 애니메이션이 완료된 후의 값입니다.
GetFinalValue 전환이 끝날 때 최종 값을 가져옵니다.
GetFinalValue 지정된 차원의 전환 끝에 있는 최종 값을 가져옵니다.
GetFinalValue 애니메이션 변수의 최종 값을 가져옵니다. 현재 예약된 모든 애니메이션이 완료된 후의 값입니다. (IUIAnimationVariable.GetFinalValue)

GetFinalValue 애니메이션 변수의 최종 값을 가져옵니다. 현재 예약된 모든 애니메이션이 완료된 후의 값입니다. (IUIAnimationVariable2.GetFinalValue)
GetFinalVectorValue 지정된 차원에 대한 애니메이션 변수의 최종 값을 가져옵니다. 현재 예약된 모든 애니메이션이 완료된 후의 값입니다.
GetIntegerValue 애니메이션 변수의 현재 값을 정수로 가져옵니다.
GetIntegerValue 애니메이션 변수의 정수 값을 가져옵니다.
GetIntegerVectorValue 지정된 차원에 대한 애니메이션 변수의 정수 값을 가져옵니다.
GetPreviousIntegerValue 애니메이션 변수의 이전 값을 정수로 가져옵니다. 가장 최근 업데이트 전에 애니메이션 변수의 값입니다.
GetPreviousIntegerValue 지정된 차원에 있는 애니메이션 변수의 이전 정수 값을 가져옵니다. 가장 최근 업데이트 전에 애니메이션 변수의 값입니다.
GetPreviousIntegerVectorValue 지정된 차원에 대한 애니메이션 변수의 이전 정수 값을 가져옵니다. 가장 최근 업데이트 전에 애니메이션 변수의 값입니다.
GetPreviousValue 애니메이션 변수의 이전 값을 가져옵니다. 가장 최근 업데이트 전에 애니메이션 변수의 값입니다. (IUIAnimationVariable.GetPreviousValue)
GetPreviousValue 애니메이션 변수의 이전 값을 가져옵니다. 가장 최근 업데이트 전에 애니메이션 변수의 값입니다. (IUIAnimationVariable2.GetPreviousValue)

GetPreviousVectorValue 지정된 차원에 대한 애니메이션 변수의 이전 값을 가져옵니다. 가장 최근 업데이트 전에 애니메이션 변수의 값입니다.
GetPrimitiveInterpolation 지정된 애니메이션 곡선의 기본 보간을 생성합니다.
GetStatus 애니메이션 관리자의 상태를 가져옵니다. (IUIAnimationManager.GetStatus)
GetStatus 애니메이션 관리자의 상태를 가져옵니다. (IUIAnimationManager2.GetStatus)
GetStatus 스토리보드의 상태를 가져옵니다. (IUIAnimationStoryboard.GetStatus)
GetStatus 스토리보드의 상태를 가져옵니다. (IUIAnimationStoryboard2.GetStatus)
GetStoryboardFromTag 지정된 태그가 있는 스토리보드를 가져옵니다. (IUIAnimationManager.GetStoryboardFromTag)
GetStoryboardFromTag 지정된 태그가 있는 스토리보드를 가져옵니다. (IUIAnimationManager2.GetStoryboardFromTag)
GetTag 스토리보드의 태그를 가져옵니다. (IUIAnimationStoryboard.GetTag)
GetTag 스토리보드의 태그를 가져옵니다. (IUIAnimationStoryboard2.GetTag)
GetTag 애니메이션 변수의 태그를 가져옵니다.
GetTag 애니메이션 변수의 태그를 가져옵니다.

Gettime 현재 시간을 가져옵니다.
GetValue 애니메이션 변수의 현재 값을 가져옵니다.
GetValue 애니메이션 변수의 값을 가져옵니다.
GetVariableFromTag 지정된 태그가 있는 애니메이션 변수를 가져옵니다. (IUIAnimationManager.GetVariableFromTag)
GetVariableFromTag 지정된 태그가 있는 애니메이션 변수를 가져옵니다. (IUIAnimationManager2.GetVariableFromTag)
GetVectorCurve 지정된 차원에 대한 애니메이션 변수의 애니메이션 곡선을 가져옵니다.
GetVectorValue 지정된 차원의 애니메이션 변수 값을 가져옵니다.
HasPriority 새 스토리보드가 예약된 스토리보드보다 우선 순위가 있는지 여부를 결정합니다.
HasPriority 예약된 스토리보드와 새 스토리보드 간의 상대적 우선 순위를 결정합니다.
HoldVariable 스토리보드가 끝날 때까지 지정된 애니메이션 변수를 최종 값으로 유지하도록 스토리보드에 지시합니다. (IUIAnimationStoryboard.HoldVariable)
HoldVariable 스토리보드가 끝날 때까지 지정된 애니메이션 변수를 최종 값으로 유지하도록 스토리보드에 지시합니다. (IUIAnimationStoryboard2.HoldVariable)
InterpolateValue 지정된 오프셋에서 애니메이션 변수의 값을 보간합니다.

InterpolateValue 지정된 오프셋 및 지정된 차원의 애니메이션 변수 값을 보간합니다.
InterpolateVelocity 지정된 오프셋에서 속도 또는 변경 속도를 보간합니다.
InterpolateVelocity 지정된 차원의 지정된 오프셋에서 속도 또는 변경 속도를 보간합니다.
IsDurationKnown 전환의 기간이 현재 알려진지 여부를 결정합니다.
IsDurationKnown 전환 기간을 알 수 있는지 여부를 결정합니다.
IsEnabled 타이머가 현재 사용하도록 설정되어 있는지 여부를 확인합니다.
OnCurveChanged 애니메이션 변수의 애니메이션 곡선이 변경되면 발생하는 이벤트를 처리합니다.
OnIntegerValueChanged 애니메이션 변수의 값이 변경되면 발생하는 이벤트를 처리합니다. (IUIAnimationVariableIntegerChangeHandler.OnIntegerValueChanged)
OnIntegerValueChanged 지정된 차원에서 애니메이션 변수의 정수 값이 변경될 때 발생하는 이벤트를 처리합니다.
OnLoopIterationChanged 스토리보드 내의 루프가 새 반복을 시작할 때 발생하는 루프 반복 변경 이벤트를 처리합니다.
OnManagerStatusChanged 애니메이션 관리자의 상태 변경을 처리합니다. (IUIAnimationManagerEventHandler.OnManagerStatusChanged)

OnManagerStatusChanged 애니메이션 관리자의 상태 변경을 처리합니다. (IUIAnimationManagerEventHandler2.OnManagerStatusChanged)
OnPostUpdate 애니메이션 업데이트가 완료된 후 발생하는 이벤트를 처리합니다.
OnPreUpdate 애니메이션 업데이트가 시작되기 전에 발생하는 이벤트를 처리합니다.
OnRenderingTooSlow 애니메이션의 렌더링 프레임 속도가 바람직한 최소 프레임 속도보다 낮을 때 발생하는 이벤트를 처리합니다.
OnStoryboardStatusChanged 스토리보드의 상태가 변경되면 발생하는 이벤트를 처리합니다.
OnStoryboardStatusChanged 스토리보드 상태 변경 이벤트를 처리합니다.
OnStoryboardUpdated 스토리보드를 업데이트할 때 발생하는 이벤트를 처리합니다.
OnStoryboardUpdated 스토리보드 업데이트 이벤트를 처리합니다.
OnTimerClientStatusChanged 타이머의 클라이언트 상태가 변경되면 발생하는 이벤트를 처리합니다.
OnUpdate 타이머의 업데이트 이벤트를 처리합니다.
OnValueChanged 애니메이션 변수의 값이 변경되면 발생하는 이벤트를 처리합니다. (IUIAnimationVariableChangeHandler.OnValueChanged)

OnValueChanged 지정된 차원의 애니메이션 변수 값이 변경될 때 발생하는 이벤트를 처리합니다.
일시 중지 모든 애니메이션을 일시 중지합니다. (IUIAnimationManager.Pause)
일시 중지 모든 애니메이션을 일시 중지합니다. (IUIAnimationManager2.Pause)
RepeatBetweenKeyframes 지정된 두 키 프레임 사이에 루프를 만듭니다.
RepeatBetweenKeyframes 두 키 프레임 사이에 루프를 만듭니다.
재개 모든 애니메이션을 다시 시작합니다. (IUIAnimationManager.Resume)
재개 모든 애니메이션을 다시 시작합니다. (IUIAnimationManager2.Resume)
일정 스토리보드가 재생되도록 예약하도록 지시합니다. (IUIAnimationStoryboard.Schedule)
일정 스토리보드가 재생되도록 예약하도록 지시합니다. (IUIAnimationStoryboard2.Schedule)
ScheduleTransition 단일 전환 스토리보드를 만들고 예약합니다. (IUIAnimationManager.ScheduleTransition)
ScheduleTransition 단일 전환 스토리보드를 만들고 예약합니다. (IUIAnimationManager2.ScheduleTransition)
SetAnimationMode 애니메이션 모드를 설정합니다. (IUIAnimationManager.SetAnimationMode)

SetAnimationMode 애니메이션 모드를 설정합니다. (IUIAnimationManager2.SetAnimationMode)
SetCancelPriorityComparison 예약된 스토리보드를 취소할 수 있는지 여부를 결정하기 위해 호출할 우선 순위 비교 처리기를 설정합니다.
SetCancelPriorityComparison 예약된 스토리보드를 취소할 수 있는지 여부를 결정하는 우선 순위 비교 처리기를 설정합니다.
SetCompressPriorityComparison 예약된 스토리보드를 압축할 수 있는지 여부를 결정하기 위해 호출할 우선 순위 비교 처리기를 설정합니다.
SetCompressPriorityComparison 예약된 스토리보드를 압축할 수 있는지 여부를 결정하는 우선 순위 비교 처리기를 설정합니다.
SetConcludePriorityComparison 예약된 스토리보드를 종료할 수 있는지 여부를 결정하기 위해 호출할 우선 순위 비교 처리기를 설정합니다.
SetConcludePriorityComparison 예약된 스토리보드를 종료할 수 있는지 여부를 결정하는 우선 순위 비교 처리기를 설정합니다.
SetDefaultLongestAcceptableDelay 허용되는 기본 애니메이션 지연을 설정합니다. 스토리보드가 시작되기까지 경과할 수 있는 시간입니다. (IUIAnimationManager.SetDefaultLongestAcceptableDelay)
SetDefaultLongestAcceptableDelay 허용되는 기본 애니메이션 지연을 설정합니다. 스토리보드가 시작되기까지 경과할 수 있는 시간입니다. (IUIAnimationManager2.SetDefaultLongestAcceptableDelay)
SetDuration 전환 기간을 설정합니다.
SetDuration 지정된 차원의 전환 기간을 설정합니다.

SetFrameRateThreshold 렌더링 속도가 너무 느리다는 것을 타이머가 애플리케이션에 알릴 프레임 속도를 설정합니다.
SetInitialValue 전환의 초기 값을 설정합니다.
SetInitialValue 전환의 초기 값을 설정합니다.
SetInitialValueAndVelocity 전환 시작 시 초기 값 및 속도를 설정합니다.
SetInitialValueAndVelocity 지정된 차원에 대한 전환의 초기 값 및 속도를 설정합니다.
SetInitialVectorValue 애니메이션 변수에서 지정된 각 차원에 대한 전환의 초기 값을 설정합니다.
SetInitialVectorVelocity 애니메이션 변수에서 지정된 각 차원에 대한 전환의 초기 속도를 설정합니다.
SetInitialVelocity 전환의 초기 속도를 설정합니다.
SetInitialVelocity 전환의 초기 속도를 설정합니다.
SetLongestAcceptableDelay 예약된 스토리보드가 시작되기 전에 허용되는 가장 긴 지연 시간을 설정합니다. (IUIAnimationStoryboard.SetLongestAcceptableDelay)
SetLongestAcceptableDelay 예약된 스토리보드가 시작되기 전에 허용되는 가장 긴 지연 시간을 설정합니다. (IUIAnimationStoryboard2.SetLongestAcceptableDelay)

SetLowerBound 애니메이션 변수의 하한(바닥)을 설정합니다. 애니메이션 변수의 값이 지정된 값보다 낮아서는 안 됩니다.
SetLowerBound 애니메이션 변수 값의 하한(바닥)을 설정합니다. 애니메이션 변수의 값이 지정된 값보다 낮아서는 안 됩니다.
SetLowerBoundVector 애니메이션 변수에 대해 지정된 각 차원의 하한(바닥) 값을 설정합니다. 각 애니메이션 변수의 값이 하한 아래로 떨어지지 않아야 합니다.
SetManagerEventHandler 애니메이션 관리자 상태 업데이트에 대한 처리기를 지정합니다. (IUIAnimationManager.SetManagerEventHandler)
SetManagerEventHandler 애니메이션 관리자 상태 업데이트에 대한 처리기를 지정합니다. (IUIAnimationManager2.SetManagerEventHandler)
SetRoundingMode 애니메이션 변수의 반올림 모드를 지정합니다.
SetRoundingMode 애니메이션 변수의 반올림 모드를 설정합니다.
SetSkipDuration 애니메이션 효과를 시작할 스토리보드의 시작 부분부터 오프셋을 지정합니다.
SetStoryboardEventHandler 스토리보드 이벤트에 대한 처리기를 지정합니다. (IUIAnimationStoryboard.SetStoryboardEventHandler)
SetStoryboardEventHandler 스토리보드 이벤트에 대한 처리기를 지정합니다. (IUIAnimationStoryboard2.SetStoryboardEventHandler)

SetTag 스토리보드의 태그를 설정합니다. (IUIAnimationStoryboard.SetTag)
SetTag 스토리보드의 태그를 설정합니다. (IUIAnimationStoryboard2.SetTag)
SetTag 애니메이션 변수의 태그를 설정합니다.
SetTag 애니메이션 변수의 태그를 설정합니다.
SetTimerClientEventHandler 타이머 클라이언트 상태 변경 이벤트에 대한 처리기를 지정합니다.
SetTimerEventHandler 타이머 이벤트 처리기를 지정합니다.
SetTimerUpdateHandler 타이머 업데이트 처리기를 지정합니다.
SetTrimPriorityComparison 예약된 스토리보드를 트리밍할 수 있는지 여부를 결정하기 위해 호출할 우선 순위 비교 처리기를 설정합니다.
SetTrimPriorityComparison 예약된 스토리보드를 트리밍할 수 있는지 여부를 결정하는 우선 순위 비교 처리기를 설정합니다.
SetUpperBound 애니메이션 변수의 상한(천장)을 설정합니다. 애니메이션 변수의 값이 지정된 값보다 높아서는 안 됩니다.
SetUpperBound 애니메이션 변수 값의 상한(최대값)을 설정합니다. 애니메이션 변수의 값이 지정된 값보다 높아서는 안 됩니다.

SetUpperBoundVector 애니메이션 변수에 대해 지정된 각 차원의 상한(최대값) 값을 설정합니다. 각 애니메이션 변수의 값은 상한보다 높아서는 안 됩니다.
SetVariableChangeHandler 변수 변경 처리기를 지정합니다. 이 처리기는 애니메이션 변수의 값에 대한 변경 내용에 대해 알림을 받습니다.
SetVariableChangeHandler 애니메이션 변수의 값을 변경하는 처리기를 지정합니다.
SetVariableCurveChangeHandler 애니메이션 변수의 애니메이션 곡선 변경에 대한 처리기를 지정합니다.
SetVariableIntegerChangeHandler 정수 변수 변경 처리기를 지정합니다. 이 처리기는 애니메이션 변수의 정수 값에 대한 변경 내용에 대해 알림을 받습니다.
SetVariableIntegerChangeHandler 애니메이션 변수의 정수 값 변경에 대한 처리기를 지정합니다.
종료 애니메이션 관리자와 관련된 모든 개체를 종료합니다. (IUIAnimationManager.Shutdown)
종료 애니메이션 관리자와 관련된 모든 개체를 종료합니다. (IUIAnimationManager2.Shutdown)
업데이트 모든 애니메이션 변수의 값을 업데이트. (IUIAnimationManager.Update)
업데이트 모든 애니메이션 변수의 값을 업데이트. (IUIAnimationManager2.Update)

인터페이스

IUIAnimationInterpolator 사용자 지정 보간기를 만드는 메서드를 정의합니다.
IUIAnimationInterpolator2 사용자 지정 보간기를 만들기 위한 메서드를 정의하는 IUIAnimationInterpolator 인터페이스를 확장합니다. IUIAnimationInterpolator2는 지정된 차원의 보간을 지원합니다.
IUIAnimationLoopIterationChangeHandler2 스토리보드 루프 반복 이벤트를 처리하는 메서드를 정의합니다.
IUIAnimationManager 애니메이션을 만들고 관리하기 위한 중앙 인터페이스를 제공하는 애니메이션 관리자를 정의합니다.
IUIAnimationManager2 여러 차원에서 애니메이션을 만들고 관리하기 위한 중앙 인터페이스를 제공하는 애니메이션 관리자를 정의합니다.
IUIAnimationManagerEventHandler 애니메이션 관리자에 대한 상태 업데이트를 처리하는 메서드를 정의합니다.
IUIAnimationManagerEventHandler2 애니메이션 관리자에 대한 업데이트를 처리하는 메서드를 정의합니다.
IUIAnimationPrimitiveInterpolation 사용자 지정 보간기가 입방 다항식 곡선의 형태로 애니메이션 관리자로 전환 정보를 제공할 수 있도록 하는 메서드를 정의합니다.
IUIAnimationPriorityComparison 애니메이션 관리자가 일정 충돌을 해결하는 데 사용하는 우선 순위 비교 메서드를 정의합니다.
IUIAnimationPriorityComparison2 우선 순위 비교를 통해 일정 충돌을 해결하는 메서드를 정의합니다.
IUIAnimationStoryboard 서로 상대적으로 동기화되는 전환 그룹을 포함하는 스토리보드를 정의합니다.

IUIAnimationStoryboard2 서로 상대적으로 동기화되는 전환 그룹을 포함하는 스토리보드를 정의합니다. 이 섹션에서 TopicDescriptionAbandon MethodTerminates the storyboard, release all related animation variables, and remove the storyboard from the schedule. AddKeyframeAfterTransition MethodAdds는 지정된 전환이 끝날 때 키 프레임을 만듭니다. AddKeyframeAtOffset MethodAdds는 기존 키 프레임의 지정된 오프셋에 있는 키 프레임입니다. AddTransition MethodAdds 스토리보드로의 전환입니다. AddTransitionAtKeyframe MethodAdds는 지정된 키 프레임에서 시작하는 전환을 반환합니다. AddTransitionBetweenKeyframes MethodAdds는 두 키 프레임 간의 전환을 지원합니다. 진행 중인 키 프레임 루프의 현재 반복(루프가 UI_ANIMATION_REPEAT_INDEFINITELY 설정됨)을 종료하고, 루프를 종료하고, 스토리보드를 계속 진행합니다. Finish MethodFinishes the storyboard within the specified time, compressing the storyboard if if necessary. GetElapsedTime MethodGets는 스토리보드 재생을 시작한 이후 경과된 시간을 지정합니다. GetStatus MethodGets는 스토리보드의 상태를 설정합니다. GetTag MethodGets는 스토리보드에 대한 태그를 설정합니다. HoldVariable MethodDirects storyboard는 스토리보드가 끝날 때까지 지정된 애니메이션 변수를 최종 값으로 유지합니다. RepeatBetweenKeyframes MethodCreates a loop between two keyframes. Schedule MethodDirects the storyboard to scheduleself for play. SetSkipDuration MethodSpecifs는 애니메이션 효과를 시작할 스토리보드의 시작 부분에서 오프셋을 지정합니다. SetLongestAcceptableDelay 메서드 집합은 예약된 스토리보드가 시작되기 전에 허용되는 가장 긴 지연 시간을 설정합니다. SetStoryboardEventHandler MethodSpecifs는 스토리보드 이벤트에 대한 처리기를 지정합니다. SetTag 메서드는 스토리보드의 태그를 설정합니다. .
IUIAnimationStoryboardEventHandler 스토리보드의 상태 및 업데이트 이벤트를 처리하는 메서드를 정의합니다.
IUIAnimationStoryboardEventHandler2 스토리보드 이벤트를 처리하는 메서드를 정의합니다.
IUIAnimationTimer 애니메이션 타이밍을 관리하기 위한 서비스를 제공하는 애니메이션 타이머를 정의합니다.
IUIAnimationTimerClientEventHandler 타이머 클라이언트 상태의 변경 내용과 관련된 이벤트를 처리하는 메서드를 정의합니다.
IUIAnimationTimerEventHandler 타이밍 이벤트를 처리하는 메서드를 정의합니다.
IUIAnimationTimerUpdateHandler 타이밍 업데이트 이벤트를 처리하는 메서드를 정의합니다.

IUIAnimationTransition 시간이 지남에 따라 애니메이션 변수가 어떻게 변하는지 결정하는 전환을 정의합니다.
IUIAnimationTransition2 전환을 정의하는 IUIAnimationTransition 인터페이스를 확장합니다. IUIAnimationTransition2 전환은 지정된 차원의 시간에 따라 애니메이션 변수가 변경되는 방식을 결정합니다.
IUIAnimationTransitionFactory 사용자 지정 보간기에서 전환을 만드는 메서드를 정의합니다.
IUIAnimationTransitionFactory2 사용자 지정 보간기에서 전환을 만드는 메서드를 정의합니다. 는 지정된 차원의 전환 생성을 지원합니다.
IUIAnimationTransitionLibrary 표준 전환 라이브러리를 정의합니다.
IUIAnimationTransitionLibrary2 지정된 차원에 대한 표준 전환 라이브러리를 정의합니다.
IUIAnimationVariable 애니메이션할 수 있는 시각적 요소를 나타내는 애니메이션 변수를 정의합니다.
IUIAnimationVariable2 여러 차원에서 애니메이션할 수 있는 시각적 요소를 나타내는 애니메이션 변수를 정의합니다.
IUIAnimationVariableChangeHandler 애니메이션 변수 업데이트와 관련된 이벤트를 처리하는 메서드를 정의합니다.
IUIAnimationVariableChangeHandler2 애니메이션 변수 업데이트 이벤트를 처리하는 메서드를 정의합니다. IUIAnimationVariableChangeHandler2는 지정된 차원에서 발생하는 이벤트를 처리합니다.
IUIAnimationVariableCurveChangeHandler2 애니메이션 곡선 업데이트 이벤트를 처리하는 메서드를 정의합니다.

IUIAnimationVariableIntegerChangeHandler 애니메이션 변수 업데이트 이벤트를 처리하는 메서드를 정의합니다.
IUIAnimationVariableIntegerChangeHandler2 애니메이션 변수 업데이트 이벤트를 처리하는 메서드를 정의합니다. IUIAnimationVariableIntegerChangeHandler2는 지정된 차원에서 발생하는 이벤트를 처리합니다.

구조체

__MIDL___MIDL_itf_UIAnimation_0000_0002_0003 스토리보드 내의 시간 오프셋을 나타내는 키 프레임을 정의합니다.

Windows 생체 인식 프레임워크

아티클2023. 03. 09.

Windows 생체 인식 프레임워크 기술 개요입니다.

Windows 생체 인식 프레임워크를 개발하려면 다음 헤더가 필요합니다.

- winbio.h
- winbio_adapter.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 생체 인식 프레임워크

열거형

WINBIO_ASYNC_NOTIFICATION_METHOD 비동기 작업에 대한 완료 알림을 클라이언트 애플리케이션에 전달하는 방법을 지정하는 상수를 정의합니다.

Functions

PIBIO_ENGINE_ACCEPT_SAMPLE_DATA_FN 원시 생체 인식 샘플을 수락하고 기능 집합을 추출합니다.
PIBIO_ENGINE_ACTIVATE_FN 엔진 어댑터에 센서 구성 요소를 유휴 상태로 가져오는 데 필요한 모든 작업을 수행할 수 있는 기회를 제공합니다.
PIBIO_ENGINE_ATTACH_FN 생체 인식 단위의 처리 파이프라인에 엔진 어댑터를 추가합니다.
PIBIO_ENGINE_CHECK_FOR_DUPLICATE_FN 파이프라인의 새 템플릿이 템플릿과 연결된 ID에 관계없이 데이터베이스에 이미 저장된 템플릿을 복제하는지 여부를 결정합니다.

PIBIO_ENGINE_CLEAR_CONTEXT_FN 새 작업을 위해 생체 인식 단위의 처리 파이프라인을 준비합니다. 전자
PIBIO_ENGINE_COMMIT_ENROLLMENT_FN 등록 개체를 마무리하고 템플릿으로 변환한 다음 데이터베이스에 템플릿을 저장합니다.
PIBIO_ENGINE_CONTROL_UNIT_FN 상승된 권한이 필요하지 않은 공급업체 정의 제어 작업을 수행합니다. 전자
PIBIO_ENGINE_CONTROL_UNIT_PRIVILEGED_FN 상승된 권한이 필요한 공급업체 정의 제어 작업을 수행합니다. 전자
PIBIO_ENGINE_CREATE_ENROLLMENT_FN 생체 인식 단위 파이프라인에서 등록 개체를 초기화합니다.
PIBIO_ENGINE_CREATE_KEY_FN HMAC 키를 센서에 푸시하기 위해 Windows 생체 인식 프레임워크에서 호출됩니다. 프레임워크가 EngineAdapterIdentifyFeatureSetSecure를 호출하면 반환된 키 식별자가 생체 인식 단위로 다시 전달됩니다.
PIBIO_ENGINE_DEACTIVATE_FN 엔진 어댑터에 센서 구성 요소를 유휴 상태로 두는 데 필요한 모든 작업을 수행할 수 있는 기회를 제공합니다.
PIBIO_ENGINE_DETACH_FN 파이프라인에 연결된 어댑터 관련 리소스를 해제합니다. 전자
PIBIO_ENGINE_DISCARD_ENROLLMENT_FN 파이프라인에서 중간 등록 상태 정보를 삭제합니다.
PIBIO_ENGINE_EXPORT_ENGINE_DATA_FN 표준 생체 인식 정보 레코드의 엔진에서 가장 최근에 처리된 기능 집합 또는 템플릿의 복사본을 검색합니다.
PIBIO_ENGINE_GET_ENROLLMENT_HASH_FN 파이프라인에서 완료된 등록 템플릿의 해시를 검색합니다.

PIBIO_ENGINE_GET_ENROLLMENT_STATUS_FN 등록 개체를 파이프라인에 커밋할 준비가 되었는지 여부를 결정합니다.
PIBIO_ENGINE_IDENTIFY_ALL_FN 현재 카메라 프레임에 있는 모든 사람의 ID를 결정합니다.
PIBIO_ENGINE_IDENTIFY_FEATURE_SET_FN 현재 기능 집합에서 템플릿을 빌드하고 데이터베이스에서 일치하는 템플릿을 찾습니다.
PIBIO_ENGINE_IDENTIFY_FEATURE_SET_SECURE_FN Windows 생체 인식 프레임워크에서 호출하여 현재 기능 집합에서 템플릿을 빌드하고 데이터베이스에서 일치하는 템플릿을 찾습니다.
PIBIO_ENGINE_NOTIFY_POWER_CHANGE_FN 컴퓨터 전원 상태의 변경에 대한 알림을 받고 그에 따라 엔진 어댑터를 준비합니다.
PIBIO_ENGINE_PIPELINE_CLEANUP_FN 엔진 어댑터에 스토리지 어댑터의 도움이 필요한 정리를 수행할 수 있는 기회를 제공합니다.
PIBIO_ENGINE_PIPELINE_INIT_FN 엔진 어댑터에 불완전하게 남아 있는 초기화를 수행할 수 있는 기회를 제공합니다.
PIBIO_ENGINE_QUERY_CALIBRATION_DATA_FN 엔진 어댑터에서 캡처 후 보정 데이터 집합을 가져옵니다.
PIBIO_ENGINE_QUERY_EXTENDED_ENROLLMENT_STATUS_FN WINBIO_PROPERTY_EXTENDED_ENROLLMENT_STATUS 속성을 쿼리합니다.
PIBIO_ENGINE_QUERY_EXTENDED_INFO_FN 생체 인식 엔진 구성 요소의 기능 및 제한 사항을 결정합니다.
PIBIO_ENGINE_QUERY_HASH_ALGORITHMS_FN 엔진 어댑터에서 지원하는 해시 알고리즘을 나타내는 개체 식별자의 배열을 검색합니다.
PIBIO_ENGINE_QUERY_INDEX_VECTOR_SIZE_FN 엔진 어댑터에서 사용하는 인덱스 벡터의 크기를 검색합니다.

PIBIO_ENGINE_QUERY_PREFERRED_FORMAT_FN 엔진 어댑터에서 선호하는 입력 데이터 형식을 결정합니다.
PIBIO_ENGINE_QUERY_SAMPLE_HINT_FN 등록 템플릿을 생성하는 데 엔진 어댑터에 필요한 올바른 샘플 수를 검색합니다.
PIBIO_ENGINE_REFRESH_CACHE_FN 엔진 어댑터에 메모리에 보관할 수 있는 캐시된 템플릿을 삭제해야 한다고 알립니다.
PIBIO_ENGINE_SELECT_CALIBRATION_FORMAT_FN Windows 생체 인식 프레임워크에서 호출하여 엔진 어댑터에서 사용하려는 센서 어댑터의 보정 형식을 결정합니다.
PIBIO_ENGINE_SET_ACCOUNT_POLICY_FN 엔진 어댑터에서 사용하는 확장 기본 및 사용자별 스푸핑 방지 정책을 설정합니다.
PIBIO_ENGINE_SET_ENROLLMENT_PARAMETERS_FN 엔진 어댑터에 등록 작업에 대한 추가 정보를 제공합니다.
PIBIO_ENGINE_SET_ENROLLMENT_SELECTOR_FN 엔진 어댑터에 현재 등록 작업을 추적할 사람을 알려줍니다.
PIBIO_ENGINE_SET_HASH_ALGORITHM_FN 후속 작업에서 사용할 해시 알고리즘을 선택합니다.
PIBIO_ENGINE_UPDATE_ENROLLMENT_FN 등록 개체에 현재 기능 집합을 추가합니다.
PIBIO_ENGINE_VERIFY_FEATURE_SET_FN 현재 기능 집합의 템플릿을 데이터베이스의 특정 템플릿과 비교합니다.
PIBIO_SENSOR_ACCEPT_CALIBRATION_DATA_FN 엔진 어댑터에서 센서 어댑터로 보정 데이터를 전달합니다.
PIBIO_SENSOR_ACTIVATE_FN 센서 어댑터에게 센서 구성 요소를 유휴 상태에서 해제하는 데 필요한 모든 작업을 수행할 수 있는 기회를 제공합니다.

PIBIO_SENSOR_ATTACH_FN 생체 인식 단위의 처리 파이프라인에 센서 어댑터를 추가합니다.
PIBIO_SENSOR_CANCEL_FN 보류 중인 모든 센서 작업을 취소합니다.
PIBIO_SENSOR_CLEAR_CONTEXT_FN PIBIO_SENSOR_CLEAR_CONTEXT_FN(winbio_adapter.h)는 새 작업을 위해 생체 인식 단위의 처리 파이프라인을 준비합니다.
PIBIO_SENSOR_CONTROL_UNIT_FN PIBIO_SENSOR_CONTROL_UNIT_FN(winbio_adapter.h)는 상승된 권한이 필요하지 않은 공급업체 정의 제어 작업을 수행합니다.
PIBIO_SENSOR_CONTROL_UNIT_PRIVILEGED_FN PIBIO_SENSOR_CONTROL_UNIT_PRIVILEGED_FN 콜백 함수(winbio_adapter.h)는 상승된 권한이 필요한 공급업체 정의 제어 작업을 수행합니다.
PIBIO_SENSOR_DEACTIVATE_FN 센서 어댑터에 센서 구성 요소를 유휴 상태로 두는 데 필요한 모든 작업을 수행할 수 있는 기회를 제공합니다.
PIBIO_SENSOR_DETACH_FN 파이프라인에 연결된 어댑터 특정 리소스를 해제합니다.
PIBIO_SENSOR_EXPORT_SENSOR_DATA_FN 표준 WINBIO_BIR 구조 형식의 가장 최근에 캡처된 생체 인식 샘플을 검색합니다.
PIBIO_SENSOR_FINISH_CAPTURE_FN PIBIO_SENSOR_FINISH_CAPTURE_FN 콜백(winbio_adapter.h)은 센서 표시기가 켜져 있는지 여부를 나타내는 값을 검색합니다.
PIBIO_SENSOR_GET_INDICATOR_STATUS_FN PIBIO_SENSOR_GET_INDICATOR_STATUS_FN 콜백 함수(winbio_adapter.h)는 센서 표시기가 켜져 있는지 여부를 나타내는 값을 검색합니다.
PIBIO_SENSOR_NOTIFY_POWER_CHANGE_FN 컴퓨터 전원 상태의 변경에 대한 알림을 받고 그에 따라 센서 어댑터를 준비합니다.

PIBIO_SENSOR_PIPELINE_CLEANUP_FN 센서 어댑터에 엔진 또는 스토리지 어댑터 구성 요소의 도움이 필요한 정리를 수행할 수 있는 기회를 제공합니다.
PIBIO_SENSOR_PIPELINE_INIT_FN 센서 어댑터에 불완전하게 유지되고 엔진 또는 스토리지 어댑터 구성 요소의 도움이 필요한 초기화를 수행할 수 있는 기회를 제공합니다.
PIBIO_SENSOR_PUSH_DATA_TO_ENGINE_FN 샘플 버퍼의 현재 내용을 엔진 어댑터에서 사용할 수 있도록 합니다.
PIBIO_SENSOR_QUERY_CALIBRATION_FORMATS_FN 센서 어댑터에서 지원하는 보정 형식 집합을 결정합니다.
PIBIO_SENSOR_QUERY_EXTENDED_INFO_FN 생체 인식 센서 구성 요소의 기능 및 제한 사항을 결정합니다.
PIBIO_SENSOR_QUERY_STATUS_FN 센서 디바이스의 현재 상태에 대한 정보를 검색합니다.
PIBIO_SENSOR_RESET_FN 센서를 다시 초기화합니다.
PIBIO_SENSOR_SET_CALIBRATION_FORMAT_FN 센서 어댑터에 엔진 어댑터에서 특정 보정 데이터 형식을 선택했음을 알 수 있습니다.
PIBIO_SENSOR_SET_INDICATOR_STATUS_FN 센서 표시기를 켜거나 끕니다.
PIBIO_SENSOR_SET_MODE_FN 센서 어댑터 모드를 설정합니다.
PIBIO_SENSOR_START_CAPTURE_FN 비동기 생체 인식 캡처를 시작합니다.

PIBIO_STORAGE_ACTIVATE_FN 스토리지 어댑터에 스토리지 구성 요소를 유휴 상태에서 벗어나는 데 필요한 모든 작업을 수행할 수 있는 기회를 제공합니다.
PIBIO_STORAGE_ADD_RECORD_FN 데이터베이스에 템플릿을 추가합니다.
PIBIO_STORAGE_ATTACH_FN 생체 인식 단위의 처리 파이프라인에 스토리지 어댑터를 추가합니다.
PIBIO_STORAGE_CLEAR_CONTEXT_FN PIBIO_STORAGE_CLEAR_CONTEXT_FN 콜백 함수(winbio_adapter.h)는 새 작업을 위해 생체 인식 단위의 처리 파이프라인을 준비합니다.
PIBIO_STORAGE_CLOSE_DATABASE_FN 파이프라인과 연결된 데이터베이스를 닫고 관련된 모든 리소스를 해제합니다.
PIBIO_STORAGE_CONTROL_UNIT_FN PIBIO_STORAGE_CONTROL_UNIT_FN 콜백 함수(winbio_adapter.h)는 상승된 권한이 필요하지 않은 공급업체 정의 제어 작업을 수행합니다.
PIBIO_STORAGE_CONTROL_UNIT_PRIVILEGED_FN PIBIO_STORAGE_CONTROL_UNIT_PRIVILEGED_FN 콜백 함수(winbio_adapter.h)는 상승된 권한이 필요한 공급업체 정의 제어 작업을 수행합니다.
PIBIO_STORAGE_CREATE_DATABASE_FN 새 데이터베이스를 만들고 구성합니다.
PIBIO_STORAGE_DEACTIVATE_FN 스토리지 어댑터에 스토리지 구성 요소를 유휴 상태로 두는 데 필요한 모든 작업을 수행할 수 있는 기회를 제공합니다.
PIBIO_STORAGE_DELETE_RECORD_FN 데이터베이스에서 하나 이상의 템플릿을 삭제합니다.
PIBIO_STORAGE_DETACH_FN 파이프라인에 연결된 어댑터별 리소스를 해제합니다. S

PIBIO_STORAGE_ERASE_DATABASE_FN 데이터베이스를 지우고 삭제할 수 있는 것으로 표시합니다.
PIBIO_STORAGE_FIRST_RECORD_FN 결과 집합 커서를 집합의 첫 번째 레코드에 배치합니다.
PIBIO_STORAGE_GET_CURRENT_RECORD_FN 파이프라인 결과 집합에서 현재 레코드의 내용을 검색합니다.
PIBIO_STORAGE_GET_DATA_FORMAT_FN 파이프라인과 연결된 현재 데이터베이스에서 사용하는 형식 및 버전 정보를 검색합니다.
PIBIO_STORAGE_GET_DATABASE_SIZE_FN 데이터베이스 크기 및 사용 가능한 공간을 검색합니다.
PIBIO_STORAGE_GET_RECORD_COUNT_FN 파이프라인 결과 집합의 템플릿 레코드 수를 검색합니다.
PIBIO_STORAGE_NEXT_RECORD_FN 결과 집합 커서를 하나의 레코드로 진행합니다.
PIBIO_STORAGE_NOTIFY_POWER_CHANGE_FN 컴퓨터 전원 상태의 변경에 대한 알림을 받고 그에 따라 스토리지 어댑터를 준비합니다.
PIBIO_STORAGE_OPEN_DATABASE_FN 스토리지 어댑터에서 사용할 데이터베이스를 엽니다.
PIBIO_STORAGE_PIPELINE_CLEANUP_FN 스토리지 어댑터에 템플릿 데이터베이스를 닫을 준비를 위해 정리를 수행할 수 있는 기회를 제공합니다.
PIBIO_STORAGE_PIPELINE_INIT_FN 스토리지 어댑터에 불완전하게 남아 있는 초기화를 수행할 수 있는 기회를 제공합니다.
PIBIO_STORAGE_QUERY_BY_CONTENT_FN 지정된 인덱스 벡터와 연결된 템플릿에 대해 현재 열려 있는 데이터베이스를 쿼리합니다.

PIBIO_STORAGE_QUERY_BY_SUBJECT_FN 지정된 ID 및 하위 요소와 연결된 템플릿에 대해 현재 열려 있는 데이터베이스를 쿼리합니다.
PIBIO_STORAGE_QUERY_EXTENDED_INFO_FN 생체 인식 스토리지 구성 요소의 기능 및 제한 사항을 결정합니다.
PWINBIO_ASYNC_COMPLETION_CALLBACK WinBioAsyncOpenSession 또는 WinBioAsyncOpenFramework를 사용하여 시작된 비동기 작업이 완료되었음을 클라이언트 애플리케이션에 알립니다.
PWINBIO_CAPTURE_CALLBACK 비동기 WinBioCaptureSampleWithCallback 함수의 결과를 반환합니다.
PWINBIO_ENROLL_CAPTURE_CALLBACK 비동기 WinBioEnrollCaptureWithCallback 함수의 결과를 반환하기 위해 Windows 생체 인식 프레임워크에서 호출됩니다.
PWINBIO_EVENT_CALLBACK 비동기 WinBioRegisterEventMonitor 함수의 결과를 반환합니다.
PWINBIO_IDENTIFY_CALLBACK 비동기 WinBioIdentifyWithCallback 함수의 결과를 반환합니다.
PWINBIO_LOCATE_SENSOR_CALLBACK 비동기 WinBioLocateSensorWithCallback 함수의 결과를 반환합니다.
PWINBIO_VERIFY_CALLBACK 비동기 WinBioVerifyWithCallback 함수의 결과를 반환합니다.
WbioQueryEngineInterface 엔진 어댑터의 WINBIO_ENGINE_INTERFACE 구조에 대한 포인터를 검색합니다.
WbioQuerySensorInterface 센서 어댑터의 WINBIO_SENSOR_INTERFACE 구조에 대한 포인터를 검색합니다.
WbioQueryStorageInterface 스토리지 어댑터의 WINBIO_STORAGE_INTERFACE 구조에 대한 포인터를 검색합니다.

WinBioAcquireFocus 창 포커스를 가져옵니다.
WinBioAsyncEnumBiometricUnits 입력 인자 유형과 일치하는 연결된 모든 생체 인식 단위를 비동기적으로 열거합니다.
WinBioAsyncEnumDatabases 지정된 형식과 일치하는 등록된 모든 데이터베이스를 비동기적으로 열거합니다.
WinBioAsyncEnumServiceProviders 설치된 생체 인식 서비스 공급자에 대한 정보를 비동기적으로 반환합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioAsyncMonitorFrameworkChanges 생체 인식 프레임워크의 변경 내용에 대한 비동기 모니터를 시작합니다.
WinBioAsyncOpenFramework 생체 인식 프레임워크에 대한 핸들을 엽니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioAsyncOpenSession 생체 인식 서비스 공급자와 하나 이상의 생체 인식 단위에 비동기적으로 연결합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioCancel 지정된 세션에 대해 보류 중인 모든 생체 인식 작업을 취소합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioCaptureSample 생체 인식 샘플을 캡처하고 원시 또는 처리된 데이터로 BIR(생체 인식 정보 레코드)을 채웁니다.
WinBioCaptureSampleWithCallback 생체 인식 샘플을 비동기적으로 캡처하고 원시 또는 처리된 데이터를 BIR(생체 인식 정보 레코드)에 반환합니다.
WinBioCloseFramework WinBioAsyncOpenFramework를 사용하여 이전에 연 프레임워크 핸들을 닫습니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.

WinBioCloseSession 생체 인식 세션을 닫고 관련 리소스를 해제합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioControlUnit 호출자가 생체 인식 단위에서 공급업체 정의 제어 작업을 수행할 수 있도록 허용합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioControlUnitPrivileged 호출자가 생체 인식 단위에서 권한 있는 공급업체 정의 제어 작업을 수행할 수 있도록 허용합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioDeleteTemplate 템플릿 저장소에서 생체 인식 템플릿을 삭제합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioEnrollBegin 생체 인식 등록 순서를 시작하고 빈 생체 인식 템플릿을 만듭니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioEnrollCapture 생체 인식 샘플을 캡처하고 템플릿에 추가합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioEnrollCaptureWithCallback 생체 인식 샘플을 비동기적으로 캡처하여 템플릿에 추가합니다.
WinBioEnrollCommit 보류 중인 생체 인식 템플릿을 마무리하고 등록에 사용되는 생체 인식 단위와 연결된 데이터베이스에 저장합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioEnrollDiscard 등록 순서를 종료하고 보류 중인 생체 인식 템플릿을 삭제합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioEnrollSelect 여러 개인을 나타내는 데이터가 샘플 버퍼에 있을 때 등록하려는 개인을 지정합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.

WinBioEnumBiometricUnits 입력 형식과 일치하는 연결된 모든 생체 인식 단위를 열거합니다.
WinBioEnumDatabases 지정된 형식과 일치하는 등록된 모든 데이터베이스를 열거합니다.
WinBioEnumEnrollments 지정된 ID 및 생체 인식 단위에 등록된 생체 인식 하위 요소를 검색합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioEnumServiceProviders 설치된 생체 인식 서비스 공급자에 대한 정보를 검색합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioFree Windows 생체 인식 프레임워크 API 함수에 대한 이전 호출을 통해 클라이언트 애플리케이션에 할당된 메모리를 해제합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioGetCredentialState 지정된 사용자에 대해 자격 증명이 설정되었는지 여부를 지정하는 값을 검색합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioGetDomainLogonSetting 사용자가 생체 인식 정보를 사용하여 도메인에 로그온할 수 있는지 여부를 지정하는 값을 검색합니다.
WinBioGetEnabledSetting Windows 생체 인식 프레임워크를 현재 사용할 수 있는지 여부를 지정하는 값을 검색합니다.
WinBioGetEnrolledFactors 지정된 사용자가 컴퓨터에 있는 생체 인식 등록에 대한 정보를 가져옵니다.
WinBioGetLogonSetting 사용자가 생체 인식 정보를 사용하여 로그온할 수 있는지 여부를 나타내는 값을 검색합니다.

WinBioGetProperty 세션, 단위 또는 템플릿 속성을 검색합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioIdentify 생체 인식 샘플을 캡처하고 기존 생체 인식 템플릿과 일치하는지 여부를 결정합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioIdentifyWithCallback 생체 인식 샘플을 비동기적으로 캡처하고 기존 생체 인식 템플릿과 일치하는지 여부를 결정합니다.
WinBioLocateSensor 사용자가 대화형으로 선택한 생체 인식 단위의 ID 번호를 검색합니다.
WinBioLocateSensorWithCallback 사용자가 대화형으로 선택한 생체 인식 단위의 ID 번호를 비동기적으로 검색합니다.
WinBioLockUnit 단일 세션에서만 사용할 수 있는 생체 인식 단위를 잠급 수 있습니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioLogonIdentifiedUser 생체 인식 세션에서 마지막으로 성공한 식별 작업과 연결된 계정으로 빠르게 전환합니다.
WinBioMonitorPresence 지정된 생체 인식 단위에 대한 얼굴 인식 또는 홍채 모니터링 메커니즘을 켭니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioOpenSession 생체 인식 서비스 공급자 및 하나 이상의 생체 인식 단위에 연결합니다.
WinBioRegisterEventMonitor 열린 세션과 연결된 서비스 공급자로부터 이벤트 알림을 수신하도록 콜백 함수를 등록합니다.
WinBioReleaseFocus 창 포커스를 해제합니다.

WinBioRemoveAllCredentials 저장소에서 모든 자격 증명을 제거합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioRemoveAllDomainCredentials 저장소에서 현재 도메인에 대한 모든 사용자 자격 증명을 제거합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioRemoveCredential 지정된 사용자의 생체 인식 로그온 자격 증명을 삭제합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioSetCredential 현재 사용자의 생체 인식 로그온 자격 증명을 저장합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioSetProperty 생체 인식 세션, 단위, 템플릿 또는 계정과 연결된 표준 속성의 값을 설정합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioUnlockUnit 지정된 생체 인식 단위에서 세션 잠금을 해제합니다.
WinBioUnregisterEventMonitor 열린 생체 인식 세션과 연결된 서비스 공급자의 이벤트 알림을 취소합니다.
WinBioVerify 생체 인식 샘플을 캡처하고 샘플이 지정된 사용자 ID에 해당하는지 여부를 결정합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioVerifyWithCallback 비동기적으로 생체 인식 샘플을 캡처하고 샘플이 지정된 사용자 ID에 해당하는지 여부를 결정합니다.
WinBioWait 세션에 대해 보류 중인 모든 생체 인식 작업이 완료되거나 취소될 때까지 호출자 실행을 차단합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.

구조체

WINBIO_ENGINE_INTERFACE 사용자 지정 엔진 어댑터 함수에 대한 포인터를 포함합니다.
WINBIO_SENSOR_INTERFACE 사용자 지정 센서 어댑터 함수에 대한 포인터를 포함합니다.
WINBIO_STORAGE_INTERFACE 사용자 지정 스토리지 어댑터 함수에 대한 포인터를 포함합니다.
WINBIO_ADAPTER_INTERFACE_VERSION 엔진, 센서 및 스토리지 어댑터 인터페이스 테이블에 사용되는 주 버전 및 부 버전 번호를 포함합니다.
WINBIO_ASYNC_RESULT 비동기 작업의 결과를 포함합니다.
WINBIO_PIPELINE 단일 생체 인식 단위의 센서, 엔진 및 스토리지 어댑터 구성 요소에서 사용하는 공유 컨텍스트 정보를 포함합니다.
WINBIO_STORAGE_RECORD 표준 형식의 생체 인식 템플릿 및 관련 데이터를 포함합니다.

Windows 클러스터링

아티클2023. 03. 09.

Windows 클러스터링 기술 개요입니다.

Windows 클러스터링을 개발하려면 다음 헤더가 필요합니다.

- cluadmex.h
- clusapi.h
- msclus.h
- resapi.h
- smbclnt.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 클러스터링

열거형

CLCTL_CODES CLCTL_CODES(clusapi.h)는 제어 코드가 수행할 수 있는 작업을 열거합니다.
CLCTL_CODES CLCTL_CODES(msclus.h)는 제어 코드가 수행할 수 있는 작업을 열거합니다.
CLUS_CHARACTERISTICS CLUS_CHARACTERISTICS(clusapi.h)는 리소스 종류 및 리소스의 특성을 열거합니다.
CLUS_CHARACTERISTICS CLUS_CHARACTERISTICS(msclus.h)는 리소스 종류 및 리소스의 특성을 열거합니다.
CLUS_FLAGS CLUS_FLAGS(clusapi.h)는 리소스 또는 그룹을 핵심 리소스로 식별합니다.
CLUS_FLAGS CLUS_FLAGS(msclus.h)는 리소스 또는 그룹을 핵심 리소스로 식별합니다.

CLUS_GROUP_START_SETTING CLUS_GROUP_START_SETTING(clusapi.h)는 클러스터 그룹에 대한 시작 설정을 열거합니다.
CLUS_GROUP_START_SETTING CLUS_GROUP_START_SETTING(msclus.h)는 클러스터 그룹에 대한 시작 설정을 열거합니다.
CLUS_RESSUBCLASS CLUS_RESSUBCLASS(clusapi.h)는 공유 리소스를 관리하는 리소스 하위 클래스를 식별합니다.
CLUS_RESSUBCLASS CLUS_RESSUBCLASS 열거형(msclus.h)은 공유 리소스를 관리하는 리소스 하위 클래스를 식별합니다.
CLUS_RESSUBCLASS_NETWORK CLUS_RESSUBCLASS_NETWORK(clusapi.h)는 IP 주소 공급자를 관리하는 리소스 하위 클래스를 식별합니다.
CLUS_RESSUBCLASS_NETWORK CLUS_RESSUBCLASS_NETWORK 열거형(msclus.h)은 IP 주소 공급자를 관리하는 리소스 하위 클래스를 식별합니다.
CLUS_RESSUBCLASS_STORAGE CLUS_RESSUBCLASS_STORAGE(clusapi.h)는 공유 버스를 관리하는 리소스 하위 클래스를 식별합니다.
CLUS_RESSUBCLASS_STORAGE CLUS_RESSUBCLASS_STORAGE 열거형(msclus.h)은 공유 버스를 관리하는 리소스 하위 클래스를 식별합니다.
CLUSCTL_CLUSTER_CODES CLUSCTL_CLUSTER_CODES(clusapi.h)는 ClusterControl 함수에서 사용하는 클러스터 제어 코드를 열거합니다.
CLUSCTL_CLUSTER_CODES CLUSCTL_CLUSTER_CODES 열거형(msclus.h)은 ClusterControl 함수에서 사용하는 클러스터 제어 코드를 열거합니다.

CLUSCTL_GROUP_CODES CLUSCTL_GROUP_CODES(clusapi.h)는 groupcontrol 코드를 열거합니다.
CLUSCTL_GROUP_CODES CLUSCTL_GROUP_CODES 열거형(msclus.h)은 그룹 제어 코드를 열거합니다.
CLUSCTL_GROUPSET_CODES CLUSCTL_GROUPSET_CODES(clusapi.h)는 ClusterGroupSetControl 함수에서 사용하는 groupset 제어 코드를 열거합니다.
CLUSCTL_GROUPSET_CODES CLUSCTL_GROUPSET_CODES 열거형(msclus.h)은 ClusterGroupSetControl 함수에서 사용하는 groupset 컨트롤 코드를 열거합니다.
CLUSCTL_NETINTERFACE_CODES CLUSCTL_NETINTERFACE_CODES(clusapi.h)는 네트워크 인터페이스 제어 코드를 열거합니다.
CLUSCTL_NETINTERFACE_CODES CLUSCTL_NETINTERFACE_CODES 열거형(msclus.h)은 네트워크 인터페이스 제어 코드를 열거합니다.
CLUSCTL_NETWORK_CODES CLUSCTL_NETWORK_CODES(clusapi.h)는 네트워크 제어 코드를 열거합니다.
CLUSCTL_NETWORK_CODES CLUSCTL_NETWORK_CODES 열거형(msclus.h)은 네트워크 제어 코드를 열거합니다.
CLUSCTL_NODE_CODES CLUSCTL_NODE_CODES(clusapi.h)는 노드 제어 코드를 열거합니다.
CLUSCTL_NODE_CODES CLUSCTL_NODE_CODES 열거형(msclus.h)은 노드 제어 코드를 열거합니다.
CLUSCTL_RESOURCE_CODES CLUSCTL_RESOURCE_CODES(clusapi.h)는 리소스 제어 코드를 열거합니다.

CLUSCTL_RESOURCE_CODES CLUSCTL_RESOURCE_CODES 열거형(msclus.h)은 리소스 제어 코드를 열거합니다.
CLUSCTL_RESOURCE_TYPE_CODES CLUSCTL_RESOURCE_TYPE_CODES(clusapi.h)는 리소스 형식 제어 코드를 열거합니다.
CLUSCTL_RESOURCE_TYPE_CODES CLUSCTL_RESOURCE_TYPE_CODES 열거형(msclus.h)은 리소스 종류 제어 코드를 열거합니다.
CLUSGROUP_TYPE CLUSGROUP_TYPE(clusapi.h)는 만들 클러스터 그룹의 유형을 지정합니다.
CLUSGROUP_TYPE CLUSGROUP_TYPE 열거형(msclus.h)은 만들 클러스터 그룹의 유형을 지정합니다.
CLUSPROP_IPADDR_ENABLENETBIOS CLUSPROP_DWORD 구조와 함께 사용하는 경우 IP 주소 리소스의 EnableNetBIOS 속성 기능을 사용하거나 사용하지 않도록 설정합니다.
CLUSPROP_PIFLAGS CLUSPROP_PIFLAGS(clusapi.h)는 디스크 파티션 정보를 나타냅니다. 열거형 플래그는 스토리지 클래스 리소스인 디스크 파티션의 특정 속성을 식별합니다.
CLUSPROP_PIFLAGS CLUSPROP_PIFLAGS 열거형(msclus.h)은 디스크 파티션 정보를 나타냅니다.
CLUSTER_CHANGE CLUSTER_CHANGE(clusapi.h)는 반환된 알림의 유형을 설명합니다.
CLUSTER_CHANGE CLUSTER_CHANGE 열거형(msclus.h)은 반환된 알림의 유형을 설명합니다.
CLUSTER_CHANGE_CLUSTER_V2 CLUSTER_CHANGE_CLUSTER_V2(clusapi.h)는 클러스터에 대해 생성된 알림 목록을 정의합니다.

CLUSTER_CHANGE_CLUSTER_V2 CLUSTER_CHANGE_CLUSTER_V2 열거형(msclus.h)은 클러스터에 대해 생성된 알림 목록을 정의합니다.
CLUSTER_CHANGE_GROUP_V2 CLUSTER_CHANGE_GROUP_V2(clusapi.h)는 그룹에 대해 생성된 알림 목록을 정의합니다.
CLUSTER_CHANGE_GROUP_V2 CLUSTER_CHANGE_GROUP_V2 열거형(msclus.h)은 그룹에 대해 생성된 알림 목록을 정의합니다.
CLUSTER_CHANGE_GROUPSET_V2 CLUSTER_CHANGE_GROUPSET_V2(clusapi.h)는 그룹 집합에 대해 생성된 알림 목록을 정의합니다.
CLUSTER_CHANGE_GROUPSET_V2 CLUSTER_CHANGE_GROUPSET_V2 열거형(msclus.h)은 그룹 집합에 대해 생성된 알림 목록을 정의합니다.
CLUSTER_CHANGE_NETINTERFACE_V2 CLUSTER_CHANGE_NETINTERFACE_V2(clusapi.h)는 클러스터 네트워크 인터페이스에 대해 생성된 알림 집합을 정의합니다.
CLUSTER_CHANGE_NETINTERFACE_V2 CLUSTER_CHANGE_NETINTERFACE_V2 열거형(msclus.h)은 클러스터 네트워크 인터페이스에 대해 생성된 알림 집합을 정의합니다.
CLUSTER_CHANGE_NETWORK_V2 CLUSTER_CHANGE_NETWORK_V2(clusapi.h)는 클러스터 네트워크에 대해 생성된 알림을 정의합니다.
CLUSTER_CHANGE_NETWORK_V2 CLUSTER_CHANGE_NETWORK_V2 열거형(msclus.h)은 클러스터 네트워크에 대해 생성된 알림을 정의합니다.
CLUSTER_CHANGE_NODE_UPGRADE_PHASE_V2 CLUSTER_CHANGE_NODE_UPGRADE_PHASE_V2(clusapi.h)는 클러스터 노드 업그레이드에 대해 생성된 알림을 정의합니다.

CLUSTER_CHANGE_NODE_UPGRADE_PHASE_V2 CLUSTER_CHANGE_NODE_UPGRADE_PHASE_V2 열거형(msclus.h)은 클러스터 노드 업그레이드에 대해 생성된 알림을 정의합니다.
CLUSTER_CHANGE_NODE_V2 CLUSTER_CHANGE_NODE_V2(clusapi.h)는 클러스터 노드에 대해 생성된 알림을 정의합니다.
CLUSTER_CHANGE_NODE_V2 CLUSTER_CHANGE_NODE_V2 열거형(msclus.h)은 클러스터 노드에 대해 생성된 알림을 정의합니다.
CLUSTER_CHANGE_QUORUM_V2 CLUSTER_CHANGE_QUORUM_V2(clusapi.h)는 쿼럼 관련 정보에 대해 생성된 알림을 정의합니다.
CLUSTER_CHANGE_QUORUM_V2 CLUSTER_CHANGE_QUORUM_V2 열거형(msclus.h)은 쿼럼 관련 정보에 대해 생성되는 알림을 정의합니다.
CLUSTER_CHANGE_REGISTRY_V2 CLUSTER_CHANGE_REGISTRY_V2(clusapi.h)는 레지스트리 키에 대해 생성된 알림을 정의합니다.
CLUSTER_CHANGE_REGISTRY_V2 CLUSTER_CHANGE_REGISTRY_V2 열거형(msclus.h)은 레지스트리 키에 대해 생성된 알림을 정의합니다.
CLUSTER_CHANGE_RESOURCE_TYPE_V2 CLUSTER_CHANGE_RESOURCE_TYPE_V2(clusapi.h)는 리소스 종류에 대해 생성된 알림 집합을 정의합니다.
CLUSTER_CHANGE_RESOURCE_TYPE_V2 CLUSTER_CHANGE_RESOURCE_TYPE_V2 열거형(msclus.h)은 리소스 종류에 대해 생성된 알림 집합을 정의합니다.
CLUSTER_CHANGE_RESOURCE_V2 CLUSTER_CHANGE_RESOURCE_V2(clusapi.h)는 리소스에 대해 생성된 알림 목록을 정의합니다.

CLUSTER_CHANGE_RESOURCE_V2 CLUSTER_CHANGE_RESOURCE_V2 열거형(msclus.h)이 리소스에 대해 생성된 알림 목록을 정의하는 방법을 알아봅니다.
CLUSTER_CHANGE_SHARED_VOLUME_V2 CLUSTER_CHANGE_SHARED_VOLUME_V2(clusapi.h)는 클러스터 공유 볼륨에 대해 생성된 알림을 정의합니다.
CLUSTER_CHANGE_SHARED_VOLUME_V2 CLUSTER_CHANGE_SHARED_VOLUME_V2 열거형(msclus.h)은 클러스터 공유 볼륨에 대해 생성된 알림을 정의합니다.
CLUSTER_CHANGE_SPACEPORT_V2 CLUSTER_CHANGE_SPACEPORT_V2(clusapi.h)는 열거형입니다.
CLUSTER_CHANGE_SPACEPORT_V2 CLUSTER_CHANGE_SPACEPORT_V2 열거형(msclus.h)은 장애 조치(failover) 클러스터와 관련이 있으며 자세한 내용은 확인할 수 있습니다.
CLUSTER_CONTROL_OBJECT CLUSTER_CONTROL_OBJECT(clusapi.h)는 컨트롤 코드가 적용되는 클러스터 개체의 형식을 나타내는 컨트롤 코드의 8비트 개체 구성 요소입니다. 자세한 내용은 제어 코드 아키텍처를 참조하세요.
CLUSTER_CONTROL_OBJECT CLUSTER_CONTROL_OBJECT(msclus.h)는 컨트롤 코드가 적용되는 클러스터 개체 형식을 나타내는 컨트롤 코드의 8비트 개체 구성 요소입니다.
CLUSTER_CSV_VOLUME_FAULT_STATE CSV(클러스터 공유 볼륨)에 대한 다양한 오류 상태를 정의합니다.
CLUSTER_ENUM CLUSTER_ENUM(clusapi.h)는 열거되는 클러스터 개체의 형식을 설명합니다.
CLUSTER_ENUM CLUSTER_ENUM 열거형(msclus.h)은 열거되는 클러스터 개체의 형식을 설명합니다.

CLUSTER_GROUP_AUTOFAILBACK_TYPE CLUSTER_GROUP_AUTOFAILBACK_TYPE(clusapi.h)는 AutoFailbackType 그룹 공통 속성에서 장애 조치(failover) 후 해당 노드가 다시 온라인 상태가 될 때 기본 설정 소유자로 식별된 노드로 그룹을 장애 복구할지 여부를 지정하는 데 사용됩니다.
CLUSTER_GROUP_AUTOFAILBACK_TYPE AutoFailbackType 그룹 공통 속성은 CLUSTER_GROUP_AUTOFAILBACK_TYPE 사용하여 그룹이 다시 온라인 상태가 된 후 노드 소유자에게 장애 복구(failback)해야 하는지를 지정합니다.
CLUSTER_GROUP_ENUM CLUSTER_GROUP_ENUM(clusapi.h)는 ClusterGroupEnum 및 ClusterGroupOpenEnum 함수에 의해 열거되는 클러스터 개체의 형식을 설명합니다.
CLUSTER_GROUP_ENUM CLUSTER_GROUP_ENUM 열거형(msclus.h)은 ClusterGroupEnum 및 ClusterGroupOpenEnum 함수에 의해 열거되는 클러스터 개체의 형식을 설명합니다.
CLUSTER_GROUP_PRIORITY CLUSTER_GROUP_PRIORITY(clusapi.h)는 그룹의 우선 순위 수준을 지정합니다.
CLUSTER_GROUP_PRIORITY CLUSTER_GROUP_PRIORITY 열거형(msclus.h)은 그룹의 우선 순위 수준을 지정합니다. (CLUSTER_GROUP_PRIORITY)
CLUSTER_GROUP_STATE CLUSTER_GROUP_STATE(clusapi.h)는 그룹의 가능한 상태를 열거합니다.
CLUSTER_GROUP_STATE CLUSTER_GROUP_STATE 열거형(msclus.h)은 그룹의 가능한 상태를 열거합니다.
CLUSTER_MGMT_POINT_TYPE CLUSTER_MGMT_POINT_TYPE(clusapi.h)는 클러스터에 대한 관리 지점의 유형을 지정합니다.
CLUSTER_MGMT_POINT_TYPE CLUSTER_MGMT_POINT_TYPE 열거형(msclus.h)은 클러스터에 대한 관리 지점의 유형을 지정합니다.

CLUSTER_NETINTERFACE_STATE CLUSTER_NETINTERFACE_STATE(clusapi.h)는 네트워크 인터페이스 상태의 가능한 값을 열거합니다.
CLUSTER_NETINTERFACE_STATE CLUSTER_NETINTERFACE_STATE 열거형(msclus.h)은 네트워크 인터페이스 상태의 가능한 값을 열거합니다.
CLUSTER_NETWORK_ENUM CLUSTER_NETWORK_ENUM(clusapi.h)는 ClusterNetworkEnum 및 ClusterNetworkOpenEnum 함수에 의해 열거되는 클러스터 개체의 형식을 설명합니다.
CLUSTER_NETWORK_ENUM CLUSTER_NETWORK_ENUM(msclus.h)는 ClusterNetworkEnum 및 ClusterNetworkOpenEnum 함수에 의해 열거되는 클러스터 개체의 형식을 설명합니다.
CLUSTER_NETWORK_ROLE CLUSTER_NETWORK_ROLE(clusapi.h)는 클러스터에서 네트워크가 수행하는 역할을 설명합니다.
CLUSTER_NETWORK_ROLE CLUSTER_NETWORK_ROLE 열거형(msclus.h)은 클러스터에서 네트워크가 수행하는 역할을 설명합니다.
CLUSTER_NETWORK_STATE CLUSTER_NETWORK_STATE(clusapi.h)는 네트워크 상태의 가능한 값을 열거합니다.
CLUSTER_NETWORK_STATE CLUSTER_NETWORK_STATE 열거형(msclus.h)은 네트워크 상태의 가능한 값을 열거합니다.
CLUSTER_NODE_DRAIN_STATUS CLUSTER_NODE_DRAIN_STATUS(clusapi.h)는 노드 드레이닝 상태의 가능한 값을 열거합니다.
CLUSTER_NODE_DRAIN_STATUS CLUSTER_NODE_DRAIN_STATUS 열거형(msclus.h)은 노드 드레이닝 상태의 가능한 값을 열거합니다.
CLUSTER_NODE_ENUM CLUSTER_NODE_ENUM(clusapi.h)는 ClusterNodeEnum 및 ClusterNodeOpenEnum 함수에 의해 열거되는 클러스터 개체의 형식을 설명합니다.

CLUSTER_NODE_ENUM CLUSTER_NODE_ENUM(msclus.h)는 ClusterNodeEnum 및 ClusterNodeOpenEnum 함수에 의해 열거되는 클러스터 개체의 형식을 설명합니다.
CLUSTER_NODE_RESUME_FAILBACK_TYPE ResumeClusterNodeEx 함수에 의해 일시 중지된 상태의 클러스터 노드가 다시 시작될 때 사용할 장애 복구 유형을 지정합니다.
CLUSTER_NODE_STATE CLUSTER_NODE_STATE(clusapi.h)는 클러스터 노드의 상태를 설명합니다.
CLUSTER_NODE_STATE CLUSTER_NODE_STATE 열거형(msclus.h)은 클러스터 노드의 상태를 설명합니다.
CLUSTER_NODE_STATUS CLUSTER_NODE_STATUS(clusapi.h)는 클러스터 노드의 상태를 설명합니다.
CLUSTER_NODE_STATUS CLUSTER_NODE_STATUS 열거형(msclus.h)은 클러스터 노드의 상태를 설명합니다.
CLUSTER_NOTIFICATIONS_VERSION CLUSTER_NOTIFICATIONS_VERSION(clusapi.h)는 다양한 버전의 클러스터 알림 열거형을 정의합니다.
CLUSTER_NOTIFICATIONS_VERSION CLUSTER_NOTIFICATIONS_VERSION 열거형(msclus.h)은 다양한 버전의 클러스터 알림 열거형을 정의합니다.
CLUSTER_OBJECT_TYPE CLUSTER_OBJECT_TYPE(clusapi.h)는 알림이 요청되거나 생성되는 개체의 형식을 정의합니다.
CLUSTER_OBJECT_TYPE CLUSTER_OBJECT_TYPE 열거형(msclus.h)은 알림이 요청되거나 생성되는 개체의 형식을 정의합니다.
CLUSTER_PROPERTY_FORMAT CLUSTER_PROPERTY_FORMAT(clusapi.h)는 속성 목록에서 속성 값의 데이터 형식을 지정합니다.

CLUSTER_PROPERTY_FORMAT CLUSTER_PROPERTY_FORMAT 열거형(msclus.h)은 속성 목록에서 속성 값의 데이터 형식을 지정합니다.
CLUSTER_PROPERTY_SYNTAX CLUSTER_PROPERTY_SYNTAX(clusapi.h)는 속성 목록의 구문 구조에 대해 가능한 값을 제공합니다.
CLUSTER_PROPERTY_SYNTAX CLUSTER_PROPERTY_SYNTAX 열거형(msclus.h)은 속성 목록의 구문 구조에 대해 가능한 값을 제공합니다.
CLUSTER_PROPERTY_TYPE CLUSTER_PROPERTY_TYPE(clusapi.h)는 클러스터 속성 목록에서 지원하는 속성 형식을 정의합니다.
CLUSTER_PROPERTY_TYPE CLUSTER_PROPERTY_TYPE 열거형(msclus.h)은 클러스터 속성 목록에서 지원하는 속성 형식을 정의합니다.
CLUSTER_QUORUM_TYPE CLUSTER_QUORUM_TYPE(clusapi.h)는 만들 쿼럼 리소스의 유형입니다.
CLUSTER_QUORUM_TYPE CLUSTER_QUORUM_TYPE 열거형(msclus.h)은 만들 쿼럼 리소스의 형식입니다.
CLUSTER_QUORUM_VALUE CLUSTER_QUORUM_VALUE(clusapi.h)는 CLUSCTL_CLUSTER_CHECK_VOTER_DOWN 또는 CLUSCTL_CLUSTER_CHECK_VOTER_EVICT 제어 코드를 사용하여 ClusterControl 함수에서 반환한 값을 열거합니다.
CLUSTER_QUORUM_VALUE CLUSTER_QUORUM_VALUE(msclus.h)는 ClusterControl 함수에서 반환하는 값을 두 개의 특정 제어 코드로 열거합니다.
CLUSTER_REG_COMMAND CLUSTER_REG_COMMAND(clusapi.h)는 클러스터 조인을 시도할 때 로컬 노드가 수행할 수 있는 클러스터 레지스트리 명령을 열거합니다.

CLUSTER_REG_COMMAND CLUSTER_REG_COMMAND(msclus.h)는 클러스터 조인을 시도할 때 로컬 노드가 수행할 수 있는 클러스터 레지스트리 명령을 열거합니다.
CLUSTER_RESOURCE_APPLICATION_STATE 리소스 애플리케이션 상태를 열거합니다.
CLUSTER_RESOURCE_CLASS CLUSTER_RESOURCE_CLASS(clusapi.h)는 리소스의 클래스를 정의합니다.
CLUSTER_RESOURCE_CLASS CLUSTER_RESOURCE_CLASS 열거형(msclus.h)은 리소스의 클래스를 정의합니다.
CLUSTER_RESOURCE_CREATE_FLAGS CLUSTER_RESOURCE_CREATE_FLAGS(clusapi.h)는 지정된 리소스를 할당할 리소스 모니터를 결정합니다.
CLUSTER_RESOURCE_CREATE_FLAGS CLUSTER_RESOURCE_CREATE_FLAGS 열거형(msclus.h)은 지정된 리소스를 할당할 리소스 모니터를 결정합니다.
CLUSTER_RESOURCE_EMBEDDED_FAILURE_ACTION CLUSTER_RESOURCE_EMBEDDED_FAILURE_ACTION(clusapi.h)는 리소스에 포함된 오류가 있을 때 수행할 수 있는 다양한 작업을 지정합니다.
CLUSTER_RESOURCE_EMBEDDED_FAILURE_ACTION CLUSTER_RESOURCE_EMBEDDED_FAILURE_ACTION 열거형(msclus.h)은 리소스에 포함된 오류가 있을 때 수행할 수 있는 다양한 작업을 지정합니다.
CLUSTER_RESOURCE_ENUM CLUSTER_RESOURCE_ENUM(clusapi.h)는 ClusterResourceEnum 또는 ClusterResourceOpenEnum 함수에 의해 열거되는 클러스터 개체의 형식을 설명합니다.
CLUSTER_RESOURCE_ENUM CLUSTER_RESOURCE_ENUM(msclus.h)는 ClusterResourceEnum 또는 ClusterResourceOpenEnum 함수에 의해 열거되는 클러스터 개체의 형식을 설명합니다.

CLUSTER_RESOURCE_RESTART_ACTION CLUSTER_RESOURCE_RESTART_ACTION(clusapi.h)는 RestartAction 리소스 공용 속성에서 리소스가 실패할 경우 클러스터 서비스에서 수행할 작업을 지정하는 데 사용됩니다.
CLUSTER_RESOURCE_RESTART_ACTION CLUSTER_RESOURCE_RESTART_ACTION(msclus.h)는 RestartAction 리소스 공용 속성에서 리소스가 실패할 경우 클러스터 서비스의 작업을 지정하는 데 사용됩니다.
CLUSTER_RESOURCE_STATE CLUSTER_RESOURCE_STATE(clusapi.h) 열거형은 리소스의 작동 상태를 설명합니다.
CLUSTER_RESOURCE_STATE CLUSTER_RESOURCE_STATE 열거형(msclus.h)은 리소스의 작동 상태를 설명합니다.
CLUSTER_RESOURCE_STATE_CHANGE_REASON CLUSTER_RESOURCE_STATE_CHANGE_REASON(clusapi.h)는 제어 코드에서 리소스 상태 변경 이유를 설명하는 데 사용됩니다.
CLUSTER_RESOURCE_STATE_CHANGE_REASON CLUSTER_RESOURCE_STATE_CHANGE_REASON(msclus.h)는 리소스 상태 변경 이유를 설명하기 위해 두 개의 특정 제어 코드에서 사용됩니다.
CLUSTER_RESOURCE_TYPE_ENUM CLUSTER_RESOURCE_TYPE_ENUM ClusterResourceTypeEnum 및 ClusterResourceTypeOpenEnum 함수에 의해 열거되는 클러스터 개체의 형식을 설명합니다.
CLUSTER_RESOURCE_TYPE_ENUM CLUSTER_RESOURCE_TYPE_ENUM 열거형(msclus.h)은 두 가지 특정 함수로 열거되는 클러스터 개체의 형식을 설명합니다.
CLUSTER_ROLE 표준 클러스터 역할의 이름을 포함합니다.
CLUSTER_ROLE_STATE ResUtilGetClusterRoleState 함수에 대한 잠재적 반환 값을 정의합니다.

CLUSTER_SETUP_PHASE ClusterSetupProgressCallback 함수에서 클러스터 설치 프로세스의 현재 단계를 식별하는 데 사용됩니다.
CLUSTER_SETUP_PHASE_SEVERITY 클러스터 설치 프로세스의 현재 단계의 심각도를 설명합니다.
CLUSTER_SETUP_PHASE_TYPE 클러스터 설치 프로세스의 진행 상황을 설명합니다.
CLUSTER_SHARED_VOLUME_BACKUP_STATE CSV 백업 상태를 설명합니다.
CLUSTER_SHARED_VOLUME_SNAPSHOT_STATE CLUSTER_SHARED_VOLUME_SNAPSHOT_STATE(clusapi.h)는 공유 볼륨에 대한 다양한 스냅샷 상태를 지정합니다.
CLUSTER_SHARED_VOLUME_SNAPSHOT_STATE CLUSTER_SHARED_VOLUME_SNAPSHOT_STATE 열거형(msclus.h)은 공유 볼륨에 대한 다양한 스냅샷 상태를 지정합니다.
CLUSTER_SHARED_VOLUME_STATE 클러스터 공유 볼륨의 상태를 정의합니다.
CLUSTER_UPGRADE_PHASE 클러스터에서 운영 체제의 롤링 업그레이드 상태를 설명합니다. 이 열거형은 ClusterUpgradeProgressCallback 콜백 함수에서 사용됩니다.
FAILURE_TYPE 클러스터 리소스에 대한 오류 유형을 정의합니다.
FILESHARE_CHANGE_ENUM 파일 공유 이벤트 알림 목록의 항목을 설명하기 위해 FILESHARE_CHANGE 구조에서 사용할 수 있는 변경 이벤트를 포함합니다.
LOG_LEVEL LogEvent 콜백 함수에 전달된 로그 이벤트의 심각도를 나타냅니다.

MAINTENANCE_MODE_TYPE_ENUM 유지 관리를 위해 표시될 때 스토리지 클래스 리소스를 배치할 수 있는 가능한 상태를 정의합니다.
NODE_CLUSTER_STATE NODE_CLUSTER_STATE(clusapi.h)는 클러스터의 상태를 나타냅니다. GetNodeClusterState 함수는 이 열거형을 사용합니다.
NODE_CLUSTER_STATE NODE_CLUSTER_STATE 열거형(msclus.h)은 클러스터의 상태를 나타냅니다.
PLACEMENT_OPTIONS 클러스터 배치 옵션을 정의합니다.
RESDLL_CONTEXT_OPERATION_TYPE GET_OPERATION_CONTEXT_PARAMS 구조체에 대한 다양한 유형의 컨텍스트 작업을 지정합니다.
RESOURCE_EXIT_STATE 리소스의 가능한 종료 상태를 열거합니다.
RESOURCE_MONITOR_STATE RESOURCE_MONITOR_STATE 열거형은 애플리케이션이 실행 중일 때 발생하는 계측 작업의 보기를 제공합니다.
SR_DISK_REPLICATION_ELIGIBLE 클러스터 노드의 디스크가 복제에 적합하거나 부적격할 수 있는 다양한 이유를 지정합니다.
SR_REPLICATED_DISK_TYPE SR_RESOURCE_TYPE_REPLICATED_DISK 구조체에 대해 복제된 디스크 유형을 지정합니다.
VM_RESDLL_CONTEXT 수행할 가상 머신에 대한 작업을 포함합니다.

Functions

AddClusterGroupDependency 두 클러스터 그룹 간에 종속성을 추가합니다.
AddClusterGroupSetDependency 두 클러스터 그룹 집합 간의 종속성을 추가합니다.
AddClusterGroupToGroupSetDependency 클러스터 그룹과 클러스터 그룹 집합 간의 종속성을 추가합니다.
AddClusterNode 기존 클러스터에 노드를 추가합니다.
AddClusterResourceDependency 두 리소스 간의 종속성 관계를 만듭니다.
AddClusterResourceNode 리소스를 실행할 수 있는 가능한 노드 목록에 노드를 추가합니다.
AddContextMenuItems 클러스터 개체에 대한 상황에 맞는 메뉴 항목을 만들고 장애 조치(failover) 클러스터 관리자 상황에 맞는 메뉴에 항목을 추가할 수 있습니다.
AddExtensionMenuItem 장애 조치(failover) 클러스터 관리자 상황에 맞는 메뉴에 메뉴 항목을 추가합니다.
AddPropertySheetPage 장애 조치(failover) 클러스터 관리자 속성 시트에 속성 페이지를 추가합니다.
AddResourceToClusterSharedVolumes 클러스터 공유 볼륨에 스토리지를 추가합니다.
AddWizard97Page 장애 조치(failover) 클러스터 애플리케이션 마법사와 같은 Wizard97 마법사에 Wizard97 속성 페이지를 추가합니다.
AddWizardPage 장애 조치(failover) 클러스터 관리자 마법사에 속성 페이지를 추가합니다.

ALIGN_CLUSPROP 값 목록 내에서 구조체를 올바르게 맞춥니다.
BackupClusterDatabase 클러스터 데이터베이스 및 모든 레지스트리 검사점의 백업을 만듭니다.
CancelClusterGroupOperation 클라이언트가 그룹에 대해 보류 중인 MoveClusterGroup 또는 MoveClusterGroupEx 작업을 취소할 수 있도록 합니다. 그런 다음 그룹은 영구 상태로 돌아갑니다.
CanResourceBeDependent 한 리소스가 다른 리소스에 종속될 수 있는지 여부를 결정합니다.
ChangeClusterResourceGroup 한 그룹에서 다른 그룹으로 리소스를 이동합니다.
CloseCluster 클러스터 핸들을 닫습니다.
CloseClusterCryptProvider CSP(암호화 서비스 공급자)에 대한 핸들을 닫습니다. PCLOSE_CLUSTER_CRYPT_PROVIDER 형식은 이 함수에 대한 포인터를 정의합니다.
CloseClusterGroup 그룹 핸들을 닫습니다.
CloseClusterGroupSet OpenClusterGroupSet에서 반환된 그룹 집합 핸들을 닫습니다.
CloseClusterNetInterface 네트워크 인터페이스 핸들을 닫습니다.
CloseClusterNetwork 네트워크 핸들을 닫습니다.
CloseClusterNode 노드 핸들을 닫습니다.

CloseClusterNotifyPort CreateClusterNotifyPort를 통해 설정된 알림 포트를 닫습니다.
CloseClusterResource 리소스 핸들을 닫습니다.
CLRES_V1_FUNCTION_TABLE Resource API 버전 1.0에 대한 함수 테이블을 초기화합니다.
CLRES_V2_FUNCTION_TABLE_SET Resource API 버전 2.0에 대한 함수 테이블을 초기화합니다.
CLRES_V3_FUNCTION_TABLE_SET Resource API 버전 3.0에 대한 함수 테이블을 초기화합니다.
CLRES_V4_FUNCTION_TABLE_SET Resource API 버전 4.0에 대한 함수 테이블을 초기화합니다.
ClusAddClusterHealthFault ClusAddClusterHealthFault 함수는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부인 상태 오류 유틸리티 함수입니다.
CLUSCTL_GET_ACCESS_MODE 제어 코드에서 액세스 코드를 추출합니다. 제어 코드를 구성하는 다양한 코드에 대한 자세한 내용은 제어 코드 아키텍처를 참조하세요.
CLUSCTL_GET_CONTROL_FUNCTION 제어 코드에서 작업 코드를 추출합니다. 제어 코드를 구성하는 다양한 코드에 대한 자세한 내용은 제어 코드 아키텍처를 참조하세요.
CLUSCTL_GET_CONTROL_OBJECT 컨트롤 코드에서 개체 코드를 추출합니다. 제어 코드를 구성하는 다양한 코드에 대한 자세한 내용은 제어 코드 아키텍처를 참조하세요.
CLUSCTL_GET_USER 컨트롤 코드에서 사용자 비트를 검색합니다. 자세한 내용은 제어 코드 아키텍처를 참조하세요.

CLUSCTL_USER_CODE 올바른 형식의 사용자 정의 컨트롤 코드를 생성합니다. 컨트롤 코드의 비트 레이아웃에 대한 자세한 내용은 제어 코드 아키텍처를 참조하세요.
ClusGetClusterHealthFaults ClusGetClusterHealthFaults 함수는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부인 상태 오류 유틸리티 함수입니다.
CLUSPROP_BINARY_DECLARE rgb 멤버가 호출자가 결정한 크기로 설정된 CLUSPROP_BINARY 구조를 만듭니다.
CLUSPROP_PROPERTY_NAME_DECLARE sz 멤버가 호출자에 의해 결정된 크기로 설정된 CLUSPROP_PROPERTY_NAME 구조를 만듭니다.
CLUSPROP_SZ_DECLARE sz 멤버가 호출자에 의해 결정된 크기로 설정된 CLUSPROP_SZ 구조를 만듭니다.
ClusRemoveClusterHealthFault ClusRemoveClusterHealthFault 함수는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부인 상태 오류 유틸리티 함수입니다.
CLUSTER_GET_MAJOR_VERSION 클러스터 서비스 버전 번호의 주 버전 부분을 추출합니다.
CLUSTER_GET_MINOR_VERSION 클러스터 서비스 버전 번호의 부 버전 부분을 추출합니다.
CLUSTER_GET_UPGRADE_VERSION 클러스터 서비스 버전 번호의 업그레이드 버전 부분을 추출합니다.
CLUSTER_MAKE_VERSION 클러스터 서비스 버전 번호를 만듭니다.
ClusterAddGroupToGroupSet 클러스터의 그룹 집합에 지정된 그룹을 추가합니다.

ClusterClearBackupStateForSharedVolume 클러스터 공유 볼륨의 백업 상태를 지웁니다.
ClusterCloseEnum ClusterOpenEnum에서 원래 연 클러스터 열거형 핸들을 닫습니다.
ClusterCloseEnumEx ClusterOpenEnumEx 함수에서 연 열거형에 대한 핸들을 닫습니다.
ClusterControl 클러스터에 영향을 주는 작업을 시작합니다.
ClusterDecrypt CSP(암호화 서비스 공급자)에 대한 검사점 데이터 암호를 해독합니다.
ClusterEncrypt CSP(암호화 서비스 공급자)에 대한 검사점 데이터를 암호화합니다.
ClusterEnum 클러스터의 클러스터 개체를 열거하고 호출할 때마다 한 개체의 이름을 반환합니다.
ClusterEnumEx 클러스터의 개체를 열거한 다음 클러스터 개체의 이름과 속성을 가져옵니다.
ClusterGetEnumCount 클러스터 열거형 핸들과 연결된 클러스터 개체의 수를 반환합니다.
ClusterGetEnumCountEx 클러스터 열거형 핸들과 연결된 클러스터 개체의 수를 반환합니다.
ClusterGetVolumeNameForVolumeMountPoint ClusterGetVolumeNameForVolumeMountPoint를 변경하거나 사용할 수 없습니다. 대신 GetVolumeNameForVolumeMountPoint를 사용합니다.
ClusterGetVolumePathName ClusterGetVolumePathName을 변경하거나 사용할 수 없습니다. 대신 GetVolumePathName을 사용합니다.

ClusterGroupCloseEnum 그룹 열거형 핸들을 닫습니다.
ClusterGroupCloseEnumEx 열거형을 닫고 hGroupEnumEx 핸들이 보유한 모든 메모리를 해제합니다.
ClusterGroupControl 그룹에 영향을 주는 작업을 시작합니다. 수행되는 작업은 dwControlCode 매개 변수에 전달된 제어 코드에 따라 달라집니다.
ClusterGroupEnum 그룹의 기본 소유자인 그룹 또는 노드의 리소스를 열거하고 각 호출과 함께 리소스 또는 노드의 이름을 반환합니다.
ClusterGroupEnumEx 열거형의 항목을 검색합니다.
ClusterGroupGetEnumCount 그룹 열거형 핸들과 연결된 클러스터 개체의 수를 반환합니다.
ClusterGroupGetEnumCountEx 열거형의 요소 수를 반환합니다.
ClusterGroupOpenEnum 그룹의 리소스 및/또는 기본 설정 소유자 목록에 포함된 노드를 반복하기 위한 열거자를 엽니다.
ClusterGroupOpenEnumEx 그룹 열거형에 대한 핸들을 엽니다.
ClusterGroupSetCloseEnum 그룹 집합에 대한 열린 열거형을 닫습니다.
ClusterGroupSetControl 그룹 집합에 영향을 주는 작업을 시작합니다.
ClusterGroupSetEnum 다음 열거 가능 개체를 반환합니다.

ClusterGroupSetGetEnumCount 열거자의 컬렉션에 포함된 항목 수를 가져옵니다.
ClusterGroupSetOpenEnum 클러스터에 대한 그룹 집합의 열거형을 시작합니다.
ClusterIsPathOnSharedVolume 경로가 클러스터 공유 볼륨에 있는지 여부를 확인합니다.
ClusterNetInterfaceCloseEnum 네트워크 인터페이스 열거형 핸들을 닫습니다.
ClusterNetInterfaceControl 네트워크 인터페이스에 영향을 주는 작업을 시작합니다. 수행되는 작업은 dwControlCode 매개 변수에 전달된 제어 코드에 따라 달라집니다.
ClusterNetInterfaceEnum 클러스터에 설치된 네트워크 인터페이스를 열거하고 각 호출과 함께 하나의 이름을 반환합니다.
ClusterNetInterfaceOpenEnum 설치된 네트워크 인터페이스를 반복하기 위한 열거자를 엽니다.
ClusterNetworkCloseEnum 네트워크 열거형 핸들을 닫습니다.
ClusterNetworkControl 네트워크에서 작업을 시작합니다. 수행되는 작업은 dwControlCode 매개 변수에 전달된 제어 코드에 따라 달라집니다.
ClusterNetworkEnum 네트워크에서 클러스터 개체를 열거하고 호출할 때마다 한 개체의 이름을 반환합니다.
ClusterNetworkGetEnumCount 네트워크 열거형 핸들과 연결된 클러스터 개체의 수를 반환합니다.
ClusterNetworkOpenEnum 네트워크의 개체를 반복하기 위한 열거자를 엽니다.

ClusterNodeCloseEnum 노드 열거형 핸들을 닫습니다. (ClusterNodeCloseEnum)
ClusterNodeCloseEnumEx 노드 열거형 핸들을 닫습니다. (ClusterNodeCloseEnumEx)
ClusterNodeControl 노드에 영향을 주는 작업을 시작합니다. 수행되는 작업은 dwControlCode 매개 변수에 전달된 제어 코드에 따라 달라집니다.
ClusterNodeEnum 노드에 설치된 네트워크 인터페이스 또는 그룹을 열거하고 각 호출에 대한 각 이름을 반환합니다.
ClusterNodeEnumEx CLUSTER_ENUM_ITEM 열거형에서 지정된 클러스터 노드를 검색합니다.
ClusterNodeGetEnumCount 노드 열거형 핸들과 연결된 클러스터 개체의 수를 반환합니다.
ClusterNodeGetEnumCountEx 노드 열거형 핸들과 연결된 클러스터 개체의 수를 반환합니다.
ClusterNodeOpenEnum 노드에 설치된 네트워크 인터페이스 또는 그룹을 반복하기 위한 열거자를 엽니다.
ClusterNodeOpenEnumEx 노드에 설치된 네트워크 인터페이스 또는 그룹을 반복할 수 있는 열거자를 엽니다.
ClusterOpenEnum 클러스터의 클러스터 개체를 반복하기 위한 열거자를 엽니다.
ClusterOpenEnumEx 개체를 반복하기 위해 클러스터에 대한 핸들을 엽니다.
ClusterPrepareSharedVolumeForBackup ClusterPrepareSharedVolumeForBackup을 변경하거나 사용할 수 없습니다.

ClusterRegBatchAddCommand 클러스터 레지스트리 키에서 실행될 일괄 처리에 명령을 추가합니다.
ClusterRegBatchCloseNotification 일괄 처리 알림과 연결된 메모리를 해제합니다.
ClusterRegBatchReadCommand 일괄 처리 알림에서 명령을 읽습니다.
ClusterRegCloseBatch ClusterRegCreateBatch 함수에서 만든 일괄 처리를 실행하거나 무시합니다. (ClusterRegCloseBatch)
ClusterRegCloseBatchEx ClusterRegCreateBatch 함수에서 만든 일괄 처리를 실행하거나 무시합니다. (ClusterRegCloseBatchEx)
ClusterRegCloseBatchNotifyPort ClusterRegCreateBatchNotifyPort 함수에서 만든 일괄 처리 알림 포트에 대한 구독을 닫습니다.
ClusterRegCloseKey 클러스터 데이터베이스 키의 핸들을 해제합니다.
ClusterRegCloseReadBatch 읽기 일괄 처리를 실행하고 읽기 일괄 처리 실행의 결과를 반환합니다. (ClusterRegCloseReadBatch)
ClusterRegCloseReadBatchEx 읽기 일괄 처리를 실행하고 읽기 일괄 처리 실행의 결과를 반환합니다. (ClusterRegCloseReadBatchEx)
ClusterRegCloseReadBatchReply 읽기 일괄 처리 결과 핸들을 닫고 연결된 메모리를 해제합니다.
ClusterRegCreateBatch 클러스터 레지스트리 키에서 명령을 실행할 일괄 처리를 만듭니다.

ClusterRegCreateBatchNotifyPort 일괄 처리 알림 포트에 대한 구독을 만듭니다.
ClusterRegCreateKey 지정된 클러스터 데이터베이스 키를 만듭니다. 키가 데이터베이스에 이미 있는 경우 ClusterRegCreateKey는 변경하지 않고 엽니다.
ClusterRegCreateReadBatch 클러스터 레지스트리 키에서 읽기 명령을 실행하는 읽기 일괄 처리에 대한 핸들을 만듭니다.
ClusterRegDeleteKey 클러스터 데이터베이스 키를 삭제합니다.
ClusterRegDeleteValue 클러스터 데이터베이스 키에서 명명된 값을 제거합니다.
ClusterRegEnumKey 열려 있는 클러스터 데이터베이스 키의 하위 키를 열거합니다.
ClusterRegEnumValue 열려 있는 클러스터 데이터베이스 키의 값을 열거합니다.
ClusterRegGetBatchNotification 일괄 처리 알림을 가져옵니다.
ClusterRegGetKeySecurity 지정된 클러스터 데이터베이스 키를 보호하는 보안 설명자의 복사본을 반환합니다.
ClusterRegOpenKey 기존 클러스터 데이터베이스 키를 엽니다.
ClusterRegQueryInfoKey 클러스터 데이터베이스 키에 대한 정보를 반환합니다.
ClusterRegQueryValue 열려 있는 클러스터 데이터베이스 키의 값과 연결된 이름, 형식 및 데이터 구성 요소를 반환합니다.

ClusterRegReadBatchAddCommand 클러스터 레지스트리 키에서 실행되는 일괄 처리에 읽기 명령을 추가합니다.
ClusterRegReadBatchReplyNextCommand 읽기 일괄 처리 결과에서 다음 명령을 읽습니다.
ClusterRegSetKeySecurity 클러스터 데이터베이스 키에 대한 보안 특성을 설정합니다.
ClusterRegSetValue 클러스터 데이터베이스 키의 값을 설정합니다.
ClusterRegSyncDatabase 클러스터 데이터베이스를 클러스터와 동기화합니다.
ClusterRemoveGroupFromGroupSet 현재 멤버인 그룹 집합에서 지정된 그룹을 제거합니다.
ClusterResourceCloseEnum 리소스 열거형 핸들을 닫습니다.
ClusterResourceCloseEnumEx 리소스 열거형에 대한 핸들을 닫습니다.
ClusterResourceControl 리소스에 영향을 주는 작업을 시작합니다. 수행되는 작업은 dwControlCode 매개 변수에 전달된 제어 코드에 따라 달라집니다.
ClusterResourceControlAsUser 리소스에 영향을 주는 작업을 시작합니다.
ClusterResourceEnum 리소스의 종속 리소스, 노드 또는 둘 다를 열거합니다.
ClusterResourceEnumEx 리소스를 열거한 다음 현재 종속 리소스 또는 노드에 대한 포인터를 반환합니다.

ClusterResourceGetEnumCount 리소스 열거형 핸들과 연결된 클러스터 개체의 수를 반환합니다.
ClusterResourceGetEnumCountEx 리소스 열거형 핸들과 연결된 클러스터 개체의 수를 반환합니다.
ClusterResourceOpenEnum 리소스의 종속성 및 노드를 반복하기 위한 열거자를 엽니다.
ClusterResourceOpenEnumEx 리소스의 종속성 및 노드를 반복할 수 있는 리소스 열거형에 대한 핸들을 엽니다.
ClusterResourceTypeCloseEnum 리소스 종류 열거형 핸들을 닫습니다.
ClusterResourceTypeControl 리소스 종류에 영향을 주는 작업을 시작합니다. 수행되는 작업은 dwControlCode 매개 변수에 전달된 제어 코드에 따라 달라집니다.
ClusterResourceTypeControlAsUser 리소스 종류에 영향을 주는 작업을 시작합니다.
ClusterResourceTypeEnum 리소스 유형의 가능한 소유자 노드 또는 리소스를 열거합니다.
ClusterResourceTypeGetEnumCount resource_type 열거형 핸들과 연결된 클러스터 개체 수를 반환합니다.
ClusterResourceTypeOpenEnum 리소스 유형의 가능한 소유자 노드 또는 리소스를 반복하기 위한 열거자를 엽니다.
ClusterSetAccountAccess 클러스터에 대한 ACL(계정 액세스 목록)을 업데이트.
ClusterSharedVolumeSetSnapshotState 클러스터 공유 볼륨의 스냅샷 상태를 업데이트.

ClusterUpgradeFunctionalLevel 클러스터에서 운영 체제의 롤링 업그레이드를 시작합니다. PCLUSAPI_CLUSTER_UPGRADE 이 함수에 대한 포인터를 정의합니다.
ClusWorkerCheckTerminate 작업자 스레드가 가능한 한 빨리 종료되어야 하는지 여부를 결정합니다. PCLUSAPIClusWorkerCheckTerminate 형식은 이 함수에 대한 포인터를 정의합니다.
ClusWorkerCreate 작업자 스레드를 만듭니다. PCLUSAPI_CLUS_WORKER_CREATE 형식은 이 함수에 대한 포인터를 정의합니다.
ClusWorkersTerminate 여러 작업자 스레드가 지정된 시간 제한까지 종료될 때까지 기다립니다.
ClusWorkerTerminate 작업자 스레드가 지정된 시간 제한까지 종료될 때까지 기다립니다. (ClusWorkerTerminate)
ClusWorkerTerminateEx 작업자 스레드가 지정된 시간 제한까지 종료될 때까지 기다립니다. (ClusWorkerTerminateEx)
CreateCluster 클러스터를 만들고 시작합니다.
CreateClusterCNOless 클러스터 이름 및 IP 주소 리소스 없이 클러스터를 만듭니다.
CreateClusterGroup 클러스터에 그룹을 추가하고 새로 추가된 그룹에 핸들을 반환합니다.
CreateClusterGroupEx 단일 작업에서 CLUSTER_CREATE_GROUP_INFO 구조에 지정된 옵션을 사용하여 새 클러스터 그룹을 만듭니다.
CreateClusterGroupSet 클러스터에 그룹 집합을 추가하고 새로 추가된 그룹 집합에 핸들을 반환합니다.

CreateClusterNameAccount 클러스터 이름 리소스를 만든 다음 클러스터를 호스트하는 컴퓨터가 도메인의 멤버가 아닌 경우에도 클러스터를 도메인에 추가합니다.
CreateClusterNotifyPort 알림 포트를 만들거나 수정합니다. 알림 포트에 대한 자세한 내용은 클러스터 이벤트 수신을 참조하세요.
CreateClusterNotifyPortV2 알림 포트를 만들거나 수정합니다. 알림 포트에 대한 자세한 내용은 클러스터 이벤트 수신을 참조하세요.
CreateClusterResource 클러스터에 리소스를 만듭니다.
CreateClusterResourceType 클러스터에 리소스 유형을 만듭니다.
CreatePropertySheetPages 클러스터 개체에 대한 속성 페이지를 만들고 장애 조치(failover) 클러스터 관리자 속성 시트에 추가합니다.
CreateWizard97Pages Wizard97 속성 페이지를 만들고 장애 조치(failover) 클러스터 관리자 마법사에 추가할 수 있습니다.
CreateWizardPages 마법사 페이지를 만들고 장애 조치(failover) 클러스터 관리자의 새 리소스 마법사 또는 클러스터 애플리케이션 마법사에 추가할 수 있습니다.
DeleteClusterGroup 클러스터에서 오프라인 및 빈 그룹을 제거합니다.
DeleteClusterGroupSet 클러스터에서 지정된 그룹 집합을 삭제합니다.
DeleteClusterResource 클러스터에서 오프라인 리소스를 제거합니다.

DeleteClusterResourceType 클러스터에서 리소스 유형을 제거합니다.
DestroyCluster 클러스터를 제거합니다.
DestroyClusterGroup 클러스터에서 지정된 그룹을 삭제합니다.
EnableNext 현재 페이지가 마지막 페이지인지 여부에 따라 Wizard97 마법사 페이지에서 다음 또는 마침 단추를 사용하거나 사용하지 않도록 설정합니다.
EnableNext 현재 페이지가 마지막 페이지인지 여부에 따라 장애 조치(failover) 클러스터 관리자 마법사 페이지에서 다음 또는 마침 단추를 사용하거나 사용하지 않도록 설정합니다.
EvictClusterNode 클러스터 데이터베이스에서 노드를 삭제합니다.
EvictClusterNodeEx 클러스터에서 노드를 제거하고 노드에서 정리 작업을 시작합니다.
FailClusterResource 리소스 오류를 시작합니다.
FreeClusterCrypt FreeClusterCrypt 함수는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부인 상태 오류 유틸리티 함수입니다.
FreeClusterHealthFault FreeClusterHealthFault 함수는 Windows 클러스터링에서 사용되는 resapi.h 헤더의 일부인 상태 오류 유틸리티 함수입니다.
FreeClusterHealthFaultArray FreeClusterHealthFaultArray 함수는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부인 상태 오류 유틸리티 함수입니다.

GetClusterFromGroup 그룹과 연결된 클러스터에 대한 핸들을 반환합니다.
GetClusterFromGroupSet GetClusterFromGroupSet 함수(clusapi.h)는 그룹 집합과 연결된 클러스터에 대한 핸들을 반환합니다.
GetClusterFromNetInterface 네트워크 인터페이스와 연결된 클러스터에 대한 핸들을 반환합니다.
GetClusterFromNetwork 네트워크와 연결된 클러스터에 대한 핸들을 반환합니다.
GetClusterFromNode 노드와 연결된 클러스터에 대한 핸들을 반환합니다.
GetClusterFromResource 리소스와 연결된 클러스터에 대한 핸들을 반환합니다.
GetClusterGroupKey 그룹에 대한 클러스터 데이터베이스 하위 트리의 루트를 엽니다.
GetClusterGroupState 그룹의 현재 상태를 반환합니다.
GetClusterHandle 클러스터에 대한 핸들을 반환합니다.
GetClusterInformation 클러스터의 이름 및 버전을 검색합니다.
GetClusterKey 클러스터에 대한 클러스터 데이터베이스 하위 트리의 루트를 엽니다.
GetClusterName 클러스터의 이름을 반환합니다. (IGetClusterDataInfo.GetClusterName)

GetClusterName 클러스터의 이름을 반환합니다. (IGetClusterUIInfo.GetClusterName)
GetClusterNetInterface 클러스터의 네트워크에 대한 노드 인터페이스의 이름을 반환합니다.
GetClusterNetInterfaceKey 네트워크 인터페이스 개체에 대한 클러스터 데이터베이스 하위 트리의 루트를 엽니다.
GetClusterNetInterfaceState 네트워크 인터페이스의 현재 상태를 반환합니다.
GetClusterNetworkId 네트워크의 식별자를 반환합니다.
GetClusterNetworkKey 네트워크에 대한 클러스터 데이터베이스 하위 트리의 루트를 엽니다.
GetClusterNetworkState 네트워크의 현재 상태입니다.
GetClusterNodeId 클러스터 노드의 고유 식별자를 반환합니다.
GetClusterNodeKey 노드에 대한 클러스터 데이터베이스 하위 트리의 루트를 엽니다.
GetClusterNodeState 노드의 현재 상태를 반환합니다.
GetClusterNotify 알림 포트에 대해 저장된 다음 알림 이벤트와 관련된 정보입니다.
GetClusterNotifyV2 알림 포트에 대한 다음 알림 이벤트에 대한 정보를 검색합니다.

GetClusterQuorumResource 클러스터의 쿼럼 리소스의 이름을 반환합니다.
GetClusterResourceDependencyExpression 지정된 리소스와 연결된 종속성 식을 검색합니다.
GetClusterResourceKey 리소스에 대한 클러스터 데이터베이스 하위 트리의 루트를 엽니다.
GetClusterResourceNetworkName 리소스가 종속된 네트워크 이름 리소스의 Name 프라이빗 속성을 검색합니다.
GetClusterResourceState 리소스의 현재 상태를 반환합니다.
GetClusterResourceTypeKey 리소스 종류에 대한 클러스터 데이터베이스 하위 트리의 루트를 엽니다.
GetCurrentClusterNodeId 현재 클러스터 노드의 고유 식별자를 반환합니다.
GetFont 속성 및 마법사 페이지에 표시할 글꼴에 대한 핸들을 반환합니다.
GetGroupHandle 그룹에 대한 핸들을 반환합니다.
GetIcon 속성 및 마법사 페이지의 왼쪽 위 모서리에 사용할 아이콘의 핸들을 반환합니다.
GetLocale 속성 및 마법사 페이지에 사용할 로캘 식별자를 반환합니다.
GetNetInterfaceHandle 노드에 대한 핸들을 검색합니다.

GetNetworkHandle 네트워크에 대한 핸들을 검색합니다.
GetNodeClusterState 클러스터 서비스가 노드에 설치되어 실행 중인지 여부를 결정합니다.
GetNodeHandle 노드에 대한 핸들을 반환합니다.
GetNotifyEventHandle 알림 이벤트에 대한 핸들을 검색합니다.
GetObjectCount 선택한 클러스터 개체의 개수를 반환합니다.
GetObjectName 클러스터 개체의 이름을 반환합니다.
GetObjectType 클러스터 개체의 형식을 반환합니다.
GetResourceHandle 리소스에 대한 핸들을 반환합니다.
GetResourceNetworkName 리소스가 의존하는 네트워크 이름 리소스에서 관리하는 네트워크의 이름을 반환합니다.
GetResourceTypeName 리소스의 형식을 반환합니다.
InitializeClusterHealthFault InitializeClusterHealthFault 함수는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부인 상태 오류 유틸리티 함수입니다.
InitializeClusterHealthFaultArray InitializeClusterHealthFaultArray 함수는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부인 상태 오류 유틸리티 함수입니다.

InvokeCommand 사용자가 상황에 맞는 메뉴 항목을 선택할 때 실행되는 프로시저를 구현할 수 있습니다.
IsFileOnClusterSharedVolume 파일이 클러스터 공유 볼륨에 있는지 여부를 지정합니다.
LPRESOURCE_CALLBACK LPRESOURCE_CALLBACK 콜백 함수는 다른 함수에서 액세스할 수 있으며 첫 번째 함수가 완료된 후에 호출됩니다. (LPRESOURCE_CALLBACK)
LPRESOURCE_CALLBACK_EX LPRESOURCE_CALLBACK_EX 콜백 함수는 다른 함수에서 액세스할 수 있으며 첫 번째 함수가 완료된 후에 호출됩니다. (LPRESOURCE_CALLBACK_EX)
MoveClusterGroup 그룹 및 모든 해당 리소스를 한 노드에서 다른 노드로 이동합니다.
MoveClusterGroupEx 플래그 및 버퍼를 추가하여 기존 MoveClusterGroup 메서드를 확장합니다.
OfflineClusterGroup 그룹을 오프라인으로 전환합니다.
OfflineClusterGroupEx OfflineClusterGroup 메서드를 확장합니다.
OfflineClusterResource 리소스를 오프라인으로 전환합니다.
OfflineClusterResourceEx OfflineClusterResource 메서드를 확장합니다.
OnlineClusterGroup 그룹을 온라인으로 가져옵니다. (OnlineClusterGroup)
OnlineClusterGroupEx 그룹을 온라인으로 가져옵니다. (OnlineClusterGroupEx)

OnlineClusterResource 오프라인 또는 실패한 리소스를 온라인으로 가져옵니다. (OnlineClusterResource)
OnlineClusterResourceEx 오프라인 또는 실패한 리소스를 온라인으로 가져옵니다. (OnlineClusterResourceEx)
OpenCluster 클러스터에 대한 연결을 열고 클러스터에 대한 핸들을 반환합니다. (OpenCluster)
OpenClusterCryptProvider 클러스터 리소스에 대한 검사점 데이터 암호화를 관리하기 위해 CSP(암호화 서비스 공급자)에 대한 핸들을 엽니다. POPEN_CLUSTER_CRYPT_PROVIDER 형식은 이 함수에 대한 포인터를 정의합니다.
OpenClusterEx 클러스터에 대한 연결을 열고 클러스터에 대한 핸들을 반환합니다. (OpenClusterEx)
OpenClusterGroup 장애 조치(failover) 클러스터 그룹을 열고 해당 그룹에 대한 핸들을 반환합니다. (OpenClusterGroup)
OpenClusterGroupEx 장애 조치(failover) 클러스터 그룹을 열고 해당 그룹에 대한 핸들을 반환합니다. (OpenClusterGroupEx)
OpenClusterGroupSet 지정된 그룹 집합에 대한 핸들을 엽니다.
OpenClusterNetInterface 네트워크 인터페이스에 대한 핸들을 엽니다. (OpenClusterNetInterface)
OpenClusterNetInterfaceEx 네트워크 인터페이스에 대한 핸들을 엽니다. (OpenClusterNetInterfaceEx)
OpenClusterNetwork 네트워크에 대한 연결을 열고 네트워크에 대한 핸들을 반환합니다. (OpenClusterNetwork)

OpenClusterNetworkEx 네트워크에 대한 연결을 열고 네트워크에 대한 핸들을 반환합니다. (OpenClusterNetworkEx)
OpenClusterNode 노드를 열고 노드에 대한 핸들을 반환합니다. (OpenClusterNode)
OpenClusterNodeEx 노드를 열고 노드에 대한 핸들을 반환합니다. (OpenClusterNodeEx)
OpenClusterResource 리소스를 열고 해당 리소스에 대한 핸들을 반환합니다. (OpenClusterResource)
OpenClusterResourceEx 리소스를 열고 해당 리소스에 대한 핸들을 반환합니다. (OpenClusterResourceEx)
PARBITRATE_ROUTINE 노드에서 쿼럼 리소스의 소유권을 다시 얻을 수 있습니다.
PauseClusterNode 노드가 클러스터 활동을 일시적으로 일시 중단할 것을 요청합니다. PCLUSAPI_PAUSE_CLUSTER_NODE 형식은 이 함수에 대한 포인터를 정의합니다.
PauseClusterNodeEx 노드가 클러스터 활동을 일시적으로 일시 중단하는 것을 요청합니다.
PBEGIN_RESCALL_AS_USER_ROUTINE PBEGIN_RESCALL_AS_USER_ROUTINE 형식은 이 콜백 함수에 대한 포인터를 정의합니다.
PBEGIN_RESCALL_ROUTINE 리소스 제어 코드에 대한 호출을 시작합니다. PBEGIN_RESCALL_ROUTINE 형식은 이 콜백 함수에 대한 포인터를 정의합니다.
PBEGIN_RESTYPECALL_AS_USER_ROUTINE PBEGIN_RESTYPECALL_AS_USER_ROUTINE 형식은 이 콜백 함수에 대한 포인터를 정의합니다.

PBEGIN_RESTYPECALL_ROUTINE 리소스 제어 코드에 대한 호출을 시작합니다. PBEGIN_RESTYPECALL_ROUTINE 형식은 이 콜백 함수에 대한 포인터를 정의합니다.
PCANCEL_ROUTINE 리소스에 대한 작업을 취소합니다.
PCHANGE_RES_TYPE_PROCESS_FOR_DUMPS PCHANGE_RES_TYPE_PROCESS_FOR_DUMPS 형식은 이 함수에 대한 포인터를 정의합니다.
PCHANGE_RESOURCE_PROCESS_FOR_DUMPS PCHANGE_RESOURCE_PROCESS_FOR_DUMPS 형식은 이 함수에 대한 포인터를 정의합니다.
PCLOSE_ROUTINE 리소스를 닫습니다.
PCLUSAPI_CLUS_WORKER_TERMINATE 작업자 스레드를 종료합니다. PCLUSAPI_CLUS_WORKER_TERMINATE 형식은 이 함수에 대한 포인터를 정의합니다.
PCLUSTER_SETUP_PROGRESS_CALLBACK 클러스터 설정 진행에 대한 정기적인 업데이트를 수신하는 콜백 함수입니다.
PCLUSTER_UPGRADE_PROGRESS_CALLBACK 클러스터에서 운영 체제의 롤링 업그레이드에 대한 상태 정보를 검색합니다. PCLUSTER_UPGRADE_PROGRESS_CALLBACK 형식은 이 함수에 대한 포인터를 정의합니다.
PEND_CONTROL_CALL 리소스 제어 코드 작업이 완료되면 호출됩니다. PEND_CONTROL_CALL 형식은 이 함수에 대한 포인터를 정의합니다.
PEND_TYPE_CONTROL_CALL 리소스 유형 제어 코드 작업이 완료되면 호출됩니다. PEND_TYPE_CONTROL_CALL 형식은 이 함수에 대한 포인터를 정의합니다.
PEXTEND_RES_CONTROL_CALL 리소스 제어 코드에 대한 호출에 대한 시간 제한을 확장합니다. PEXTEND_RES_CONTROL_CALL 형식은 이 함수에 대한 포인터를 정의합니다.

PEXTEND_RES_TYPE_CONTROL_CALL 리소스 유형 제어 코드에 대한 호출에 대한 시간 제한을 확장합니다. PEXTEND_RES_TYPE_CONTROL_CALL 형식은 이 함수에 대한 포인터를 정의합니다.
PIS_ALIVE_ROUTINE 리소스를 사용할 수 있는지 여부를 결정합니다.
PLOG_EVENT_ROUTINE 클러스터 로그에 이벤트를 기록합니다.
PLOOKS_ALIVE_ROUTINE 리소스를 사용할 수 있는 것으로 보이는지 여부를 결정합니다.
POFFLINE_ROUTINE POFFLINE_ROUTINE 콜백 함수는 정리 처리가 완료된 후 리소스를 사용할 수 없음으로 표시합니다.
POFFLINE_V2_ROUTINE POFFLINE_V2_ROUTINE 콜백 함수는 정리 처리가 완료된 후 리소스를 사용할 수 없음으로 표시합니다.
PONLINE_ROUTINE PONLINE_ROUTINE 콜백 함수는 리소스를 사용할 수 있는 것으로 표시합니다. (PONLINE_ROUTINE)
PONLINE_V2_ROUTINE PONLINE_V2_ROUTINE 콜백 함수는 리소스를 사용할 수 있는 것으로 표시합니다. (PONLINE_V2_ROUTINE)
POPEN_ROUTINE POPEN_ROUTINE 콜백 함수는 리소스를 엽니다. POPEN_ROUTINE 형식은 이 함수에 대한 포인터를 정의합니다.
POPEN_V2_ROUTINE POPEN_V2_ROUTINE 콜백 함수는 리소스를 엽니다. POPEN_V2_ROUTINE 형식은 이 함수에 대한 포인터를 정의합니다.
PQUORUM_RESOURCE_LOST 쿼럼 리소스의 제어가 손실되었을 때 호출됩니다.

PRAISE_RES_TYPE_NOTIFICATION TBD. PRAISE_RES_TYPE_NOTIFICATION 형식은 이 함수에 대한 포인터입니다.
PRELEASE_ROUTINE 중재에서 쿼럼 리소스를 해제합니다.
PRESOURCE_CONTROL_ROUTINE 리소스에 적용되는 작업을 수행합니다.
PRESOURCE_TYPE_CONTROL_ROUTINE 리소스 종류에 적용되는 작업을 수행합니다.
PSET_INTERNAL_STATE 리소스의 내부 상태를 설정합니다.
PSET_RESOURCE_INMEMORY_NODELOCAL_PROPERTIES_ROUTINE PSET_RESOURCE_INMEMORY_NODELOCAL_PROPERTIES_ROUTINE 콜백 함수는 리소스 유형 제어 코드 작업이 완료되면 호출됩니다.
PSET_RESOURCE_LOCKED_MODE_ROUTINE 잠긴 모드가 리소스에 대해 구성된 보고서입니다.
PSET_RESOURCE_STATUS_ROUTINE PSET_RESOURCE_STATUS_ROUTINE 콜백 함수는 리소스의 상태를 업데이트하기 위해 호출됩니다.
PSET_RESOURCE_STATUS_ROUTINE_EX PSET_RESOURCE_STATUS_ROUTINE_EX 콜백 함수는 리소스의 상태를 업데이트하기 위해 호출됩니다. (PSET_RESOURCE_STATUS_ROUTINE_EX)
PSIGNAL_FAILURE_ROUTINE 리소스 인스턴스에 오류가 있었다고 보고합니다. PSIGNAL_FAILURE_ROUTINE 형식은 이 함수에 대한 포인터를 정의합니다.
PSTARTUP_EX_ROUTINE 함수 테이블과 버전 번호를 포함하는 구조를 반환하여 리소스 DLL을 로드합니다.

PSTARTUP_ROUTINE 함수 테이블과 버전 번호를 포함하는 구조를 반환하여 리소스 DLL을 로드합니다.
PTERMINATE_ROUTINE 정리 처리가 완료될 때까지 기다리지 않고 리소스를 사용할 수 없음으로 즉시 표시합니다.
PWORKER_START_ROUTINE 지정된 콜백 루틴을 사용하여 작업자 스레드를 초기화합니다. PWORKER_START_ROUTINE 형식은 이 함수에 대한 포인터를 정의합니다.
RegisterAppInstance 프로세스에 대한 AppInstance ID를 등록합니다.
RegisterClusterNotify 알림 포트에 대해 저장된 이벤트 목록에 이벤트 유형을 추가합니다.
RegisterClusterNotifyV2 이벤트 유형에 알림 키를 추가하여 알림 포트에 이벤트 유형을 등록합니다.
RegisterClusterResourceTypeNotifyV2 클러스터 알림 포트에 알림 유형을 추가합니다.
RemoveClusterGroupDependency 두 클러스터 그룹 간의 종속성을 제거합니다.
RemoveClusterGroupSetDependency 그룹 집합의 종속성 식에서 그룹 집합을 제거합니다.
RemoveClusterGroupToGroupSetDependency 그룹의 종속성 식에서 그룹 집합을 제거합니다.
RemoveClusterResourceDependency 두 리소스 간의 종속성 관계를 제거합니다.
RemoveClusterResourceNode 리소스를 호스트할 수 있는 노드 목록에서 노드를 제거합니다.

RemoveResourceFromClusterSharedVolumes 클러스터 공유 볼륨에서 스토리지를 제거합니다.
RestartClusterResource 클러스터 리소스를 다시 시작합니다.
RestoreClusterDatabase 클러스터 데이터베이스를 복원하고 함수가 호출되는 노드에서 클러스터 서비스를 다시 시작합니다. 이 노드를 복원 노드라고합니다.
ResumeClusterNode 일시 중지된 노드가 클러스터 활동을 다시 시작하라는 요청입니다. PCLUSAPI_RESUME_CLUSTER_NODE 형식은 이 함수에 대한 포인터를 정의합니다.
ResumeClusterNodeEx 지정된 장애 복구(failback) 작업을 시작한 다음, 일시 중지된 노드가 클러스터 작업을 다시 시작합니다.
ResUtilAddUnknownProperties 클러스터 데이터베이스에서 알 수 없는 속성 집합을 검색하고 속성 목록의 끝에 추가합니다.
ResUtilCreateDirectoryTree 이미 존재하는 디렉터리를 건너뛰고 경로에 지정된 모든 디렉터리를 만듭니다. PRESUTIL_CREATE_DIRECTORY_TREE 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilDupParameterBlock 한 매개 변수 블록에서 다른 매개 변수 블록으로 데이터의 멤버 단위 복사본을 수행합니다.
ResUtilDupString null로 끝나는 유니코드 문자열을 복제합니다.
ResUtilEnumPrivateProperties 클러스터 개체의 개인 속성 이름을 검색합니다. PRESUTIL_ENUM_PRIVATE_PROPERTIES 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilEnumProperties 클러스터 개체의 속성 이름을 열거합니다. PRESUTIL_ENUM_PROPERTIES 형식은 이 함수에 대한 포인터를 정의합니다.

ResUtilEnumResources 로컬 클러스터의 모든 리소스를 열거하고 각 리소스에 대해 사용자 정의 작업을 시작합니다. PRESUTIL_ENUM_RESOURCES 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilEnumResourcesEx 지정된 클러스터의 모든 리소스를 열거하고 각 리소스에 대해 사용자 정의 작업을 시작합니다. PRESUTIL_ENUM_RESOURCES_EX 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilEnumResourcesEx2 지정된 클러스터의 모든 리소스를 열거하고 각 리소스에 대해 사용자 정의 작업을 시작합니다. PRESUTIL_ENUM_RESOURCES_EX2 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilExpandEnvironmentStrings 환경 변수에 대한 확장되지 않은 참조가 포함된 문자열을 확장합니다. PRESUTIL_EXPAND_ENVIRONMENT_STRINGS 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilFindBinaryProperty 속성 목록에서 지정된 이진 속성을 찾고 속성 값을 반환할 수도 있습니다. PRESUTIL_FIND_BINARY_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilFindDependentDiskResourceDriveLetter 리소스의 실제 디스크 종속성과 연결된 드라이브 문자를 검색합니다. PRESUTIL_FIND_DEPENDENT_DISK_RESOURCE_DRIVE_LETTER 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilFindDwordProperty 속성 목록에서 부호 없는 long 속성 값을 찾습니다. PRESUTIL_FIND_DWORD_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilFindExpandedSzProperty 속성 목록에서 확장된 문자열 속성 값을 찾습니다. PRESUTIL_FIND_EXPANDED_SZ_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilFindExpandSzProperty 속성 목록에서 확장 가능한 문자열 속성을 찾습니다. PRESUTIL_FIND_EXPAND_SZ_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilFindFileTimeProperty TBD. PRESUTIL_FIND_FILETIME_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.

ResUtilFindLongProperty 속성 목록에서 서명된 long 속성 값을 찾습니다. PRESUTIL_FIND_LONG_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilFindMultiSzProperty 속성 목록에서 여러 문자열 속성을 찾습니다. PRESUTIL_FIND_MULTI_SZ_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilFindSzProperty 속성 목록에서 문자열 속성을 찾습니다. PRESUTIL_FIND_SZ_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilFindULargeIntegerProperty 속성 목록에서 큰 정수 속성 값을 가져옵니다. PRESUTIL_FIND_ULARGEINTEGER_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilFreeEnvironment ResUtilGetEnvironmentWithNetName을 사용하여 만든 환경 변수 블록을 삭제합니다. PRESUTIL_FREE_ENVIRONMENT 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilFreeParameterBlock ResUtilDupParameterBlock에 의해 매개 변수 블록에 할당된 메모리의 할당을 취소합니다.
ResUtilGetAllProperties 클러스터 개체에 대한 모든 기본 속성과 알 수 없는 속성을 포함하는 속성 목록을 반환합니다. PRESUTIL_GET_ALL_PROPERTIES 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetBinaryProperty 속성 목록에서 이진 속성을 검색하고 목록의 다음 속성에 대한 포인터를 진행합니다. PRESUTIL_GET_BINARY_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetBinaryValue 클러스터 데이터베이스에서 이진 값을 반환합니다.
ResUtilGetClusterRoleState 특정 역할이 클러스터에 할당되었는지 여부를 결정합니다.

ResUtilGetCoreClusterResources 핵심 네트워크 이름, IP 주소 및 쿼럼 리소스에 대한 핸들을 반환합니다. PRESUTIL_GET_CORE_CLUSTER_RESOURCES 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetCoreClusterResourcesEx 코어, 네트워크 이름, IP 주소 및 쿼럼 리소스에 대한 핸들을 반환합니다. PRESUTIL_GET_CORE_CLUSTER_RESOURCES_EX 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetDwordProperty 속성 목록에서 DWORD 속성을 검색하고 목록의 다음 속성에 대한 포인터를 앞으로 이동합니다. PRESUTIL_GET_DWORD_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetDwordValue 클러스터 데이터베이스에서 숫자 값을 반환합니다.
ResUtilGetEnvironmentWithNetName 리소스가 클러스터 네트워크 이름을 사용하여 위치를 식별하도록 리소스에 대한 환경 데이터를 조정합니다.
ResUtilGetExpandSzValue 클러스터 데이터베이스에서 확장 가능한 문자열 값을 반환합니다.
ResUtilGetFileTimeProperty TBD. PRESUTIL_GET_FILETIME_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetLongProperty TBD. PRESUTIL_GET_LONG_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetMultiSzProperty 속성 목록에서 여러 문자열 속성을 검색하고 목록의 다음 속성에 대한 포인터를 앞으로 이동합니다. PRESUTIL_GET_MULTI_SZ_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetMultiSzValue 클러스터 데이터베이스에서 여러 문자열 값을 반환합니다.
ResUtilGetPrivateProperties 클러스터 개체에 대한 프라이빗 속성을 반환합니다. PRESUTIL_GET_PRIVATE_PROPERTIES 형식은 이 함수에 대한 포인터를 정의합니다.

ResUtilGetProperties 클러스터 데이터베이스에서 속성 테이블에 지정된 속성을 검색하고 속성 목록에 반환합니다. PRESUTIL_GET_PROPERTIES 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetPropertiesToParameterBlock 클러스터 데이터베이스에서 속성 테이블에 지정된 속성을 검색하고 매개 변수 블록에 반환합니다.
ResUtilGetProperty 클러스터 데이터베이스에서 지정된 속성을 반환합니다. PRESUTIL_GET_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetPropertyFormats 지정된 속성 집합의 형식을 설명하는 속성 서식 목록을 반환합니다. PRESUTIL_GET_PROPERTY_FORMATS 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetPropertySize 지정된 속성에 필요한 총 바이트 수를 반환합니다.
ResUtilGetQwordValue ResUtilGetQwordValue 함수는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부인 상태 오류 유틸리티 함수입니다.
ResUtilGetResourceDependency 지정된 리소스의 종속성을 열거하고 지정된 형식의 종속성에 대한 핸들을 반환합니다. PRESUTIL_GET_RESOURCE_DEPENDENCY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetResourceDependencyByClass 지정된 클러스터에서 지정된 리소스의 종속성을 열거하고 지정된 리소스 클래스와 일치하는 종속성에 대한 핸들을 반환합니다. PRESUTIL_GET_RESOURCE_DEPENDENCY_BY_CLASS 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetResourceDependencyByClassEx 지정된 클러스터에서 지정된 리소스의 종속성을 열거하고 지정된 리소스 클래스와 일치하는 종속성에 대한 핸들을 반환합니다. PRESUTIL_GET_RESOURCE_DEPENDENCY_BY_CLASS_EX 형식은 이 함수에 대한 포인터를 정의합니다.

ResUtilGetResourceDependencyByName 지정된 클러스터에서 지정된 리소스의 종속성을 열거하고 지정된 형식의 종속성에 대한 핸들을 반환합니다. PRESUTIL_GET_RESOURCE_DEPENDENCY_BY_NAME 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetResourceDependencyByNameEx 지정된 클러스터에서 지정된 리소스의 종속성을 열거하고 지정된 형식의 종속성에 대한 핸들을 반환합니다. PRESUTIL_GET_RESOURCE_DEPENDENCY_BY_NAME_EX 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetResourceDependencyEx 지정된 리소스의 종속성을 열거하고 지정된 형식의 종속성에 대한 핸들을 반환합니다. PRESUTIL_GET_RESOURCE_DEPENDENCY_EX 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetResourceDependentIPAddressProps 지정된 리소스에 대해 발견된 첫 번째 IP 주소 종속성의 개인 속성을 검색합니다. PRESUTIL_GET_RESOURCE_DEPENDENTIP_ADDRESS_PROPS 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetResourceName 리소스의 이름을 반환합니다. PRESUTIL_GET_RESOURCE_NAME 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetResourceNameDependency 로컬 클러스터에서 지정된 리소스의 종속성을 열거하고 지정된 리소스 유형의 종속성에 대한 핸들을 반환합니다. PRESUTIL_GET_RESOURCE_NAME_DEPENDENCY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetResourceNameDependencyEx 로컬 클러스터에서 지정된 리소스의 종속성을 열거하고 지정된 리소스 유형의 종속성에 대한 핸들을 반환합니다. PRESUTIL_GET_RESOURCE_NAME_DEPENDENCY_EX 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetSzProperty 속성 목록에서 문자열 속성을 검색하고 목록의 다음 속성에 대한 포인터를 앞으로 이동합니다. PRESUTIL_GET_SZ_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetSzValue 클러스터 데이터베이스에서 문자열 값을 반환합니다.

ResUtilInitializeResourceStatus RESOURCE_STATUS 구조를 초기화합니다.
ResUtilIsPathValid 경로가 구문상 유효한지 여부를 확인합니다.
ResUtilIsResourceClassEqual 지정된 리소스의 리소스 클래스가 지정된 리소스 클래스와 같은지 테스트합니다. PRESUTIL_IS_RESOURCE_CLASS_EQUAL 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilLeftPaxosIsLessThanRight 지정된 Paxos 태그에 지정된 다른 Paxos 태그보다 오래된 클러스터 구성 정보가 포함되어 있는지 여부를 나타냅니다.
ResUtilPaxosComparer 두 Paxos 태그를 비교하고 동일한 값이 있는지 여부를 나타냅니다.
ResUtilPropertyListFromParameterBlock 속성 테이블 및 매개 변수 블록에서 속성 목록을 생성합니다.
ResUtilRemoveResourceServiceEnvironment 서비스에서 환경 데이터를 제거합니다. 이 함수는 리소스 DLL에서 호출해야 합니다. PRESUTIL_REMOVE_RESOURCE_SERVICE_ENVIRONMENT 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilResourcesEqual 두 개의 리소스 핸들이 동일한 리소스를 나타내는지 테스트합니다. PRESUTIL_RESOURCES_EQUAL 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilResourceTypesEqual 리소스 종류가 지정된 리소스의 리소스 종류 이름과 일치하는지 테스트합니다. PRESUTIL_RESOURCE_TYPES_EQUAL 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilSetBinaryValue 클러스터 데이터베이스에서 이진 값을 설정합니다.

ResUtilSetDwordValue 클러스터 데이터베이스의 숫자 값을 설정합니다. PRESUTIL_SET_DWORD_VALUE 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilSetExpandSzValue 클러스터 데이터베이스에서 확장 가능한 문자열 값을 설정합니다. PRESUTIL_SET_EXPAND_SZ_VALUE 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilSetMultiSzValue 클러스터 데이터베이스에서 여러 문자열 값을 설정합니다. PRESUTIL_SET_MULTI_SZ_VALUE 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilSetPrivatePropertyList 클러스터 개체의 프라이빗 속성을 설정합니다.
ResUtilSetPropertyParameterBlock 매개 변수 블록에서 클러스터 데이터베이스의 속성을 설정합니다. (ResUtilSetPropertyParameterBlock)
ResUtilSetPropertyParameterBlockEx 매개 변수 블록에서 클러스터 데이터베이스의 속성을 설정합니다. (ResUtilSetPropertyParameterBlockEx)
ResUtilSetPropertyTable 속성 테이블의 속성 목록을 기반으로 클러스터 데이터베이스의 속성을 설정합니다.
ResUtilSetPropertyTableEx 속성 테이블의 속성 목록을 기반으로 클러스터 데이터베이스의 속성을 설정합니다.
ResUtilSetQwordValue TBD. PRESUTIL_SET_QWORD_VALUE 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilSetResourceServiceEnvironment 서비스에서 클러스터 네트워크 이름을 사용하여 해당 위치를 식별할 수 있도록 서비스에 대한 환경 데이터를 조정합니다. 이 함수는 리소스 DLL에서 호출해야 합니다. PRESUTIL_SET_RESOURCE_SERVICE_ENVIRONMENT 형식은 이 함수에 대한 포인터를 정의합니다.

ResUtilSetResourceServiceStartParameters 지정된 서비스의 시작 매개 변수를 조정하여 클러스터 리소스로 올바르게 작동합니다. 리소스 DLL에서 호출해야 합니다. PRESUTIL_SET_RESOURCE_SERVICE_START_PARAMETERS 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilSetResourceServiceStartParametersEx 지정된 서비스의 시작 매개 변수를 조정하여 클러스터 리소스로 올바르게 작동합니다. 리소스 DLL에서 호출해야 합니다. PRESUTIL_SET_RESOURCE_SERVICE_START_PARAMETERS_EX 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilSetSzValue 클러스터 데이터베이스에서 문자열 값을 설정합니다. PRESUTIL_SET_SZ_VALUE 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilSetUnknownProperties 클러스터 데이터베이스에 클러스터 개체의 알 수 없는 속성을 저장합니다.
ResUtilSetValueEx 클러스터 데이터베이스의 값을 설정합니다.
ResUtilStartResourceService 서비스를 시작합니다. PRESUTIL_START_RESOURCE_SERVICE 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilStopResourceService 명명된 서비스를 중지합니다. PRESUTIL_STOP_RESOURCE_SERVICE 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilStopService 핸들로 식별되는 서비스를 중지합니다. PRESUTIL_STOP_SERVICE 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilTerminateServiceProcessFromResDll 리소스 DLL에서 클러스터 리소스로 관리되는 서비스의 프로세스를 종료하려고 시도합니다. PRESUTIL_TERMINATE_SERVICE_PROCESS_FROM_RES_DLL 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilVerifyPrivatePropertyList 속성 목록의 형식이 올바른지 확인합니다.

ResUtilVerifyPropertyTable 속성 테이블을 사용하여 속성 목록의 형식이 올바른지 확인합니다.
ResUtilVerifyResourceService 명명된 서비스가 시작되거나 현재 실행 중인지 확인합니다. PRESUTIL_VERIFY_RESOURCE_SERVICE 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilVerifyService 핸들로 식별된 서비스가 시작 중인지 현재 실행 중인지 확인합니다. PRESUTIL_VERIFY_SERVICE 형식은 이 함수에 대한 포인터를 정의합니다.
SetAppInstanceCsvFlags 애플리케이션 인스턴스의 연결에 영향을 주는 플래그를 설정합니다.
SetClusterGroupName 그룹의 이름을 설정합니다.
SetClusterGroupNodeList 그룹의 기본 노드 목록을 설정합니다.
SetClusterGroupSetDependencyExpression 클러스터 그룹 집합에 대한 종속성 식을 설정합니다.
SetClusterName 클러스터의 이름을 설정합니다.
SetClusterNetworkName 네트워크의 이름을 설정합니다.
SetClusterNetworkPriorityOrder 클러스터 노드 간의 내부 통신에 사용되는 네트워크 집합의 우선 순위 순서를 설정합니다.
SetClusterQuorumResource 리소스를 클러스터의 쿼럼 리소스로 설정합니다.

SetClusterResourceDependencyExpression hResource에서 참조하는 리소스와 연결할 종속성 식을 지정합니다. 리소스에 대한 기존 종속성 관계를 덮어씁니다.
SetClusterResourceName 리소스의 이름을 설정합니다.
SetClusterServiceAccountPassword 사용 가능한 모든 클러스터 노드에서 클러스터 서비스 사용자 계정의 암호를 변경합니다.
SetGroupDependencyExpression 클러스터 그룹에 대한 종속성 식을 설정합니다.

인터페이스

IGetClusterDataInfo IGetClusterDataInfo 인터페이스는 장애 조치(failover) 클러스터 관리자 확장에 의해 호출되어 클러스터에 대한 정보를 검색합니다.
IGetClusterGroupInfo IGetClusterGroupInfo 인터페이스는 장애 조치(failover) 클러스터 관리자 확장에 의해 호출되어 그룹에 대한 정보를 검색합니다.
IGetClusterNetInterfaceInfo IGetClusterNetInterfaceInfo 인터페이스는 장애 조치(failover) 클러스터 관리자 확장에 의해 호출되어 네트워크 인터페이스에 대한 정보를 검색합니다.
IGetClusterNetworkInfo 장애 조치(failover) 클러스터 관리자 확장에 의해 호출되어 네트워크에 대한 정보를 검색합니다.
IGetClusterNodeInfo IGetClusterNodeInfo 인터페이스는 장애 조치(failover) 클러스터 관리자 확장에 의해 호출되어 노드에 대한 정보를 검색합니다.

IGetClusterObjectInfo 클러스터 개체에 대한 정보를 검색하기 위해 장애 조치(failover) 클러스터 관리자 확장에 의해 호출됩니다.
IGetClusterResourceInfo 리소스에 대한 정보를 검색하기 위해 장애 조치(failover) 클러스터 관리자 확장에 의해 호출됩니다.
IGetClusterUIInfo 장애 조치(failover) 클러스터 관리자의 사용자 인터페이스에 대한 정보를 검색하기 위해 장애 조치(failover) 클러스터 관리자 확장에 의해 호출됩니다.
ISClusPartitionEx 실제 디스크 리소스의 파티션에 대한 확장된 정보를 제공합니다.
IWCContextMenuCallback IWCContextMenuCallback 인터페이스는 장애 조치(failover) 클러스터 관리자 확장에 의해 호출되어 장애 조치(failover) 클러스터 관리자 상황에 맞는 메뉴에 항목을 추가합니다.
IWCPropertySheetCallback IWCPropertySheetCallback 인터페이스는 장애 조치(failover) 클러스터 관리자 확장에 의해 호출되어 장애 조치(failover) 클러스터 관리자 속성 시트에 속성 페이지를 추가합니다.
IWCWizard97Callback IWCWizard97Callback 인터페이스는 장애 조치(failover) 클러스터 관리자 확장에 의해 호출되어 Wizard97 속성 페이지를 Wizard97 마법사(예: 클러스터 애플리케이션 마법사)에 추가합니다.
IWCWizardCallback IWCWizardCallback 인터페이스는 장애 조치(failover) 클러스터 관리자 확장에 의해 호출되어 장애 조치(failover) 클러스터 관리자 마법사에 속성 페이지를 추가하고 탐색을 관리합니다.
IWEExtendContextMenu IWEExtendContextMenu 인터페이스를 구현하여 클러스터 개체에 대한 장애 조치(failover) 클러스터 관리자 상황에 맞는 메뉴를 확장합니다.
IWEExtendPropertySheet IWEExtendPropertySheet 인터페이스를 구현하여 클러스터 개체에 대한 속성 시트 페이지를 만들고 장애 조치(failover) 클러스터 관리자 속성 시트에 추가합니다.

IWEExtendWizard IWEExtendWizard 인터페이스를 구현하여 장애 조치(failover) 클러스터 관리자의 새 리소스 마법사 또는 클러스터 애플리케이션 마법사에 마법사 페이지를 추가합니다.
IWEExtendWizard97 IWEExtendWizard97 인터페이스를 구현하여 마법사97 스타일 마법사 페이지를 장애 조치(failover) 클러스터 관리자 마법사에 추가합니다.
IWEInvokeCommand 장애 조치(failover) 클러스터 관리자는 사용자가 IWEExtendContextMenu 인터페이스를 사용하여 만든 상황에 맞는 메뉴 항목을 선택할 때 IWEInvokeCommand 인터페이스의 구현을 호출합니다.

구조체

CLRES_CALLBACK_FUNCTION_TABLE StartupEx 콜백 함수의 함수 테이블을 나타냅니다.
CLRES_FUNCTION_TABLE Resource API의 모든 버전에 대한 함수 테이블을 설명합니다.
CLRES_V1_FUNCTIONS Startup을 제외한 모든 Resource API 버전 1.0 진입점에 대한 포인터를 포함합니다.
CLRES_V2_FUNCTIONS StartupEx를 제외한 모든 Resource API 버전 2.0 진입점에 대한 포인터를 포함합니다.
CLRES_V3_FUNCTIONS StartupEx를 제외한 모든 Resource API 버전 3.0 진입점에 대한 포인터를 포함합니다.
CLRES_V4_FUNCTIONS StartupEx를 제외한 모든 Resource API 버전 4.0 진입점에 대한 포인터를 포함합니다.
CLUS_CHKDSK_INFO Chkdsk 작업에 대한 정보를 나타냅니다.

CLUS_CSV_MAINTENANCE_MODE_INFO CSV(클러스터 공유 볼륨)에서 유지 관리 모드를 사용하거나 사용하지 않도록 설정합니다.
CLUS_CSV_VOLUME_INFO CSV(클러스터 공유 볼륨)에 대한 정보를 나타냅니다.
CLUS_CSV_VOLUME_NAME CSV(클러스터 공유 볼륨)의 이름을 나타냅니다.
CLUS_DISK_NUMBER_INFO 실제 디스크의 디스크 번호에 대한 정보를 나타냅니다.
CLUS_DNN_LEADER_STATUS Scale-Out 파일 서버에 대한 DNN(분산 네트워크 이름) 리소스의 상태를 나타냅니다.
CLUS_DNN_SODAFS_CLONE_STATUS Scale-Out 파일 서버 복제본의 상태를 나타냅니다.
CLUS_FORCE_QUORUM_INFO 대부분의 노드 클러스터에서 쿼럼을 설정하기에 충분한 노드 목록에 대한 정보를 지정합니다.
CLUS_FTSET_INFO FT(내결함성) 집합에 대한 정보를 포함합니다. 이 구조체는 CLUSPROP_FTSET_INFO 구조체에서 값 목록에 항목을 만드는 데 사용됩니다.
CLUS_MAINTENANCE_MODE_INFO 클러스터 노드에서 유지 관리 모드를 사용하거나 사용하지 않도록 설정합니다.
CLUS_MAINTENANCE_MODE_INFOEX 스토리지 클래스 리소스에 대한 확장된 유지 관리 모드 설정을 나타냅니다.
CLUS_NETNAME_IP_INFO_ENTRY NetName 리소스의 IP 정보를 나타냅니다.
CLUS_NETNAME_IP_INFO_FOR_MULTICHANNEL 다중 채널이 사용하도록 설정된 NetName 리소스의 IP 정보를 나타냅니다.

CLUS_NETNAME_PWD_INFO 컴퓨터 이름과 연결된 보안 주체를 다시 설정하기 위한 정보를 제공합니다.
CLUS_NETNAME_VS_TOKEN_INFO 토큰을 요청하는 데 필요한 데이터를 포함합니다. CLUSCTL_RESOURCE_NETNAME_GET_VIRTUAL_SERVER_TOKEN 제어 코드의 입력 매개 변수로 사용됩니다.
CLUS_PARTITION_INFO 스토리지 클래스 리소스 볼륨 및 파일 시스템을 설명하는 데이터를 포함합니다. CLUSPROP_PARTITION_INFO 구조체의 데이터 멤버로 사용되며 일부 제어 코드 작업의 반환 값으로 사용됩니다.
CLUS_PARTITION_INFO_EX 스토리지 클래스 리소스 볼륨 및 파일 시스템에 대해 설명합니다.
CLUS_PARTITION_INFO_EX2 스토리지 클래스 리소스의 디스크 파티션 정보를 설명합니다.
CLUS_PROVIDER_STATE_CHANGE_INFO 공급자 리소스의 상태에 대한 데이터를 포함합니다.
CLUS_RESOURCE_CLASS_INFO 리소스 클래스 데이터를 포함합니다. CLUSPROP_RESOURCE_CLASS_INFO 구조체의 데이터 멤버로 사용되며 일부 제어 코드 작업의 반환 값으로 사용됩니다.
CLUS_SCSI_ADDRESS SCSI 주소 데이터를 포함합니다. CLUSPROP_SCSI_ADDRESS 구조체의 데이터 멤버로 사용되며 일부 제어 코드 작업의 반환 값으로 사용됩니다.
CLUS_SHARED_VOLUME_BACKUP_MODE CSV의 백업 모드를 설명합니다.
CLUS_STARTING_PARAMS 노드의 클러스터 서비스 시작 시도가 클러스터를 구성하거나 조인하려는 시도를 나타내는지 여부와 노드가 이전에 이 버전의 클러스터 서비스를 시작하려고 시도했는지 여부를 나타냅니다.

CLUS_STORAGE_GET_AVAILABLE_DRIVELETTERS 노드에서 사용할 수 있는 드라이버 문자의 비트 마스크를 포함합니다. CLUSCTL_RESOURCE_TYPE_STORAGE_GET_DRIVELETTERS 컨트롤 코드의 반환 값으로 사용됩니다.
CLUS_STORAGE_REMAP_DRIVELETTER 노드의 디스크 드라이브에 대한 기존 및 대상 드라이브 문자를 식별합니다.
CLUS_STORAGE_SET_DRIVELETTER 스토리지 클래스 리소스와 연결된 디스크 파티션에 대한 드라이브 문자 정보를 제공합니다.
CLUS_WORKER 작업자 스레드에 대한 정보를 포함합니다.
CLUSCTL_GROUP_GET_LAST_MOVE_TIME_OUTPUT 그룹을 다른 노드로 마지막으로 이동한 시간에 대한 정보를 지정합니다.
CLUSCTL_RESOURCE_STATE_CHANGE_REASON_STRUCT 리소스 상태 변경 이유를 제공하기 위해 CLUSCTL_RESOURCE_STATE_CHANGE_REASON 제어 코드와 함께 전송됩니다.
CLUSCTL_RESOURCE_TYPE_STORAGE_GET_AVAILABLE_DISKS_EX2_INPUT CLUSCTL_RESOURCE_TYPE_STORAGE_GET_AVAILABLE_DISKS_EX2_INT 제어 코드의 입력 버퍼를 나타냅니다.
CLUSPROP_BINARY 이진 데이터 값을 설명합니다.
CLUSPROP_BUFFER_HELPER 속성 목록 또는 값 목록을 작성하거나 구문 분석하는 데 사용됩니다.
CLUSPROP_DWORD 디스크의 실제 드라이브를 식별하는 숫자 값에 대해 설명합니다.
CLUSPROP_FILETIME 파일의 날짜 및 타임스탬프를 설명합니다.

CLUSPROP_FTSET_INFO FT(내결함성) 집합에 대한 정보를 포함합니다. 값 목록의 항목으로 사용되며 CLUSPROP_VALUE 및 CLUS_FTSET_INFO 구조로 구성됩니다.
CLUSPROP_LARGE_INTEGER 부 서명된 큰 정수에 대해 설명합니다.
CLUSPROP_LIST 속성 목록의 시작 부분에 액세스합니다.
CLUSPROP_LONG 서명된 LONG 데이터에 대해 설명합니다.
CLUSPROP_PARTITION_INFO 스토리지 클래스 리소스와 관련된 정보를 포함합니다.
CLUSPROP_PARTITION_INFO_EX CLUSPROP_PARTITION_INFO_EX 구조에는 스토리지 클래스 리소스와 관련된 정보가 포함됩니다.
CLUSPROP_PARTITION_INFO_EX2 스토리지 클래스 리소스에 대한 파티션 정보를 포함하는 값 목록 항목입니다. 이 구조는 입력이며 CLUSCTL_RESOURCE_STORAGE_GET_DISK_INFO_EX2 제어 코드의 반환 값입니다.
CLUSPROP_REQUIRED_DEPENDENCY 다른 리소스의 필수 종속성인 리소스에 대해 설명합니다. 이 공용 구조체는 CLUSCTL_RESOURCE_GET_REQUIRED_DEPENDENCIES 또는 CLUSCTL_RESOURCE_TYPE_GET_REQUIRED_DEPENDENCIES 제어 코드 작업에서 반환된 값 목록의 값으로 사용됩니다.
CLUSPROP_RESOURCE_CLASS 리소스 클래스에 대해 설명합니다.
CLUSPROP_RESOURCE_CLASS_INFO 리소스 클래스와 관련된 정보를 설명합니다.
CLUSPROP_SCSI_ADDRESS SCSI 디바이스의 주소를 설명합니다.

CLUSPROP_SECURITY_DESCRIPTOR 보안 설명자를 설명합니다.
CLUSPROP_SYNTAX 데이터 값의 형식과 형식을 설명합니다. CLUSPROP_VALUE 구조체의 구문 멤버로 사용됩니다.
CLUSPROP_SZ 여러 NULL 종료 유니코드 문자열에 대해 설명합니다.
CLUSPROP_ULARGE_INTEGER 부호 없는 큰 정수에 대해 설명합니다.
CLUSPROP_VALUE 값 목록에 사용되는 데이터 값의 구문과 길이를 설명합니다. CLUSPROP_VALUE 구조체는 CLUSPROP_BINARY 및 CLUSPROP_SZ 같은 특정 형식의 데이터를 설명하는 모든 구조체에서 제네릭 헤더로 사용됩니다.
CLUSPROP_WORD 숫자 데이터를 설명합니다.
CLUSTER_BATCH_COMMAND 현재 일괄 처리 명령 데이터가 ClusterRegBatchReadCommand 함수로 전송되는 순서를 나타냅니다.
CLUSTER_CREATE_GROUP_INFO CLUSTER_CREATE_GROUP_INFO 호출자는 새 그룹을 만들 때 추가 속성을 제공할 수 있습니다.
CLUSTER_CREATE_GROUP_INFO CLUSTER_CREATE_GROUP_INFO 구조를 사용하면 호출자가 새 그룹을 만들 때 추가 속성을 제공할 수 있습니다.
CLUSTER_ENUM_ITEM CLUSTER_ENUM_ITEM 클러스터 개체의 속성을 포함하며 ClusterEnumEx 및 ClusterNodeEnumEx 함수에서 클러스터를 열거하는 데 사용됩니다.
CLUSTER_ENUM_ITEM CLUSTER_ENUM_ITEM 구조에는 클러스터 개체의 속성이 포함되어 있으며 ClusterEnumEx 및 ClusterNodeEnumEx 함수에서 클러스터를 열거하는 데 사용됩니다.

CLUSTER_GROUP_ENUM_ITEM CLUSTER_GROUP_ENUM_ITEM(clusapi.h)는 클러스터 그룹의 속성을 포함하며 ClusterGroupEnumEx 함수에서 클러스터 그룹을 열거하는 데 사용됩니다.
CLUSTER_GROUP_ENUM_ITEM CLUSTER_GROUP_ENUM_ITEM 구조에는 클러스터 그룹의 속성이 포함됩니다. (CLUSTER_GROUP_ENUM_ITEM)
CLUSTER_HEALTH_FAULT CLUSTER_HEALTH_FAULT 구조는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부입니다.
CLUSTER_HEALTH_FAULT_ARRAY CLUSTER_HEALTH_FAULT_ARRAY 구조는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부입니다.
CLUSTER_IP_ENTRY 클러스터의 IP 주소를 설명합니다.
CLUSTER_MEMBERSHIP_INFO CLUSTER_MEMBERSHIP_INFO(clusapi.h) 구조체는 클러스터의 멤버 자격 정보를 나타냅니다.
CLUSTER_MEMBERSHIP_INFO CLUSTER_MEMBERSHIP_INFO 구조는 클러스터의 멤버 자격 정보를 나타냅니다. (CLUSTER_MEMBERSHIP_INFO)
CLUSTER_READ_BATCH_COMMAND 읽기 일괄 처리의 단일 명령에 대한 결과를 나타냅니다.
CLUSTER_RESOURCE_ENUM_ITEM CLUSTER_RESOURCE_ENUM_ITEM 클러스터 리소스의 속성을 나타내며 ClusterResourceEnumEx 함수에서 클러스터 리소스를 열거하는 데 사용됩니다.
CLUSTER_RESOURCE_ENUM_ITEM CLUSTER_RESOURCE_ENUM_ITEM 구조체는 클러스터 리소스의 속성을 나타내며 ClusterResourceEnumEx 함수에서 클러스터 리소스를 열거하는 데 사용됩니다.

CLUSTER_SET_PASSWORD_STATUS SetClusterServiceAccountPassword 함수에서 각 클러스터 노드에 대한 클러스터 서비스 사용자 계정 암호 변경 결과를 반환하는 데 사용됩니다.
CLUSTER_SHARED_VOLUME_STATE_INFO CSV(클러스터 공유 볼륨)의 상태에 대한 정보를 나타냅니다. (CLUSTER_SHARED_VOLUME_STATE_INFO)
CLUSTER_SHARED_VOLUME_STATE_INFO_EX CSV(클러스터 공유 볼륨)의 상태에 대한 정보를 나타냅니다. (CLUSTER_SHARED_VOLUME_STATE_INFO_EX)
CLUSTER_VALIDATE_CSV_FILENAME 유효성 검사 작업 중 CSV(클러스터 공유 볼륨)를 나타냅니다.
CLUSTER_VALIDATE_DIRECTORY CLUSTER_VALIDATE_DIRECTORY(clusapi.h) 구조체는 유효성을 검사하기 위해 디렉터리에 전달되며, 자세한 내용은 확인할 수 있습니다.
CLUSTER_VALIDATE_NETNAME CLUSTER_VALIDATE_NETNAME(clusapi.h) 구조체는 유효성을 검사하기 위해 네트워크 이름을 전달하며, 더 많은 정보를 확인할 수 있습니다.
CLUSTER_VALIDATE_PATH CLUSTER_VALIDATE_PATH(clusapi.h) 구조체는 유효성을 검사하기 위해 네트워크 이름을 전달하며 더 많은 정보를 확인할 수 있습니다.
CLUSTERVERSIONINFO 노드에 로컬로 설치된 클러스터 서비스 버전에 대한 정보를 설명합니다.
CLUSTERVERSIONINFO_NT4 CLUSTERVERSIONINFO_NT4(clusapi.h) 구조체는 여러 필드 형식과 관련이 있으며 자세한 내용은 결정해야 합니다.
CREATE_CLUSTER_CONFIG 초기 클러스터 구성을 정의합니다.

CREATE_CLUSTER_NAME_ACCOUNT CreateClusterNameAccount 함수에서 클러스터를 도메인에 추가하는 데 사용하는 클러스터 이름 리소스 및 도메인 자격 증명에 대해 설명합니다. PCREATE_CLUSTER_NAME_ACCOUNT 이 구조체에 대한 포인터를 정의합니다.
FILESHARE_CHANGE 이벤트 알림 목록의 항목 형식에 대해 설명합니다.
FILESHARE_CHANGE_LIST 파일 서버 리소스에서 관리하는 파일 공유에 대한 이벤트 알림 목록을 설명합니다.
GET_OPERATION_CONTEXT_PARAMS CLUSCTL_RESOURCE_GET_OPERATION_CONTEXT 제어 코드의 입력으로 사용되는 컨텍스트 매개 변수를 나타냅니다.
GROUP_FAILURE_INFO GROUP_FAILURE_INFO(clusapi.h) 구조체는 그룹 오류에 대한 장애 조치(failover) 시도에 대한 정보를 나타냅니다.
GROUP_FAILURE_INFO GROUP_FAILURE_INFO 구조체는 그룹 오류에 대한 장애 조치(failover) 시도에 대한 정보를 나타냅니다. (GROUP_FAILURE_INFO)
GROUP_FAILURE_INFO_BUFFER GROUP_FAILURE_INFO_BUFFER(clusapi.h) 구조체는 GROUP_FAILURE_INFO 구조체에 대한 버퍼를 나타냅니다.
GROUP_FAILURE_INFO_BUFFER GROUP_FAILURE_INFO_BUFFER 구조체는 GROUP_FAILURE_INFO 구조체에 대한 버퍼를 나타냅니다. (GROUP_FAILURE_INFO_BUFFER)
MONITOR_STATE MONITOR_STATE 구조는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부입니다.
NOTIFY_FILTER_AND_TYPE NOTIFY_FILTER_AND_TYPE(clusapi.h) 구조체는 CreateClusterNotifyPortV2 함수에서 만든 알림 포트에 대한 필터를 나타냅니다.

NOTIFY_FILTER_AND_TYPE NOTIFY_FILTER_AND_TYPE 구조는 CreateClusterNotifyPortV2 함수에서 만든 알림 포트에 대한 필터를 나타냅니다.
PaxosTagCStruct 클러스터에서 파일 공유 감시를 사용하는 경우 노드의 클러스터 구성 버전에 대한 정보를 저장하는 클러스터 노드의 Paxos 태그 값을 포함합니다.
POST_UPGRADE_VERSION_INFO 클러스터 서비스에 대한 업그레이드 후 상태 정보를 나타냅니다.
RESOURCE_FAILURE_INFO RESOURCE_FAILURE_INFO(clusapi.h) 구조체는 리소스에 대한 장애 조치(failover) 시도에 대한 정보를 나타냅니다.
RESOURCE_FAILURE_INFO RESOURCE_FAILURE_INFO 구조체는 리소스에 대한 장애 조치(failover) 시도에 대한 정보를 나타내며 RESOURCE_FAILURE_INFO_BUFFER 구조체에서 사용됩니다.
RESOURCE_FAILURE_INFO_BUFFER RESOURCE_FAILURE_INFO_BUFFER(clusapi.h) 구조체는 리소스 오류에 대한 버퍼를 나타냅니다.
RESOURCE_FAILURE_INFO_BUFFER RESOURCE_FAILURE_INFO_BUFFER 구조체는 리소스 오류에 대한 버퍼를 나타냅니다. (RESOURCE_FAILURE_INFO_BUFFER)
RESOURCE_STATUS 온라인 상태가 되거나 오프라인으로 전환되는 리소스에 대한 정보를 포함합니다. 이 구조는 콜백 함수 SetResourceStatus에 대한 매개 변수로 사용됩니다.
RESOURCE_STATUS_EX 온라인 상태가 되거나 오프라인으로 전환되는 리소스에 대한 정보를 포함합니다. 이 구조는 콜백 함수 SetResourceStatusEx에 대한 매개 변수로 사용됩니다.
RESOURCE_TERMINAL_FAILURE_INFO_BUFFER RESOURCE_TERMINAL_FAILURE_INFO_BUFFER(clusapi.h) 구조체는 리소스의 터미널 오류에 대한 버퍼를 나타냅니다.

RESOURCE_TERMINAL_FAILURE_INFO_BUFFER RESOURCE_TERMINAL_FAILURE_INFO_BUFFER 구조체는 리소스의 터미널 오류에 대한 버퍼를 나타냅니다.
RESUTIL_FILETIME_DATA FILETIME에 허용되는 기본값, 최대값 및 최소값을 설명합니다.
RESUTIL_LARGEINT_DATA 부 서명된 큰 정수에 허용되는 기본값, 최대값 및 최소값을 설명합니다.
RESUTIL_PROPERTY_ITEM 클러스터 개체 속성에 대한 정보를 포함합니다. RESUTIL_PROPERTY_ITEM 구조의 배열은 속성 작업에 사용할 수 있는 속성 테이블을 형성합니다.
RESUTIL_ULARGEINT_DATA 부호 없는 큰 정수에 허용되는 기본값, 최대값 및 최소값을 설명합니다.
SR_RESOURCE_TYPE_DISK_INFO 디스크가 복제에 적합한지 여부를 나타내는 정보 집합에 대해 설명합니다.
SR_RESOURCE_TYPE_ELIGIBLE_DISKS_RESULT 스토리지 복제를 위한 리소스 유형 제어 코드 작업에서 검색한 데이터 디스크 집합에 대해 설명합니다.
SR_RESOURCE_TYPE_QUERY_ELIGIBLE_LOGDISKS 지정된 데이터 디스크의 로그 디스크로 사용할 수 있는 검색된 디스크 집합에 대해 설명합니다.
SR_RESOURCE_TYPE_QUERY_ELIGIBLE_SOURCE_DATADISKS 복제를 위한 원본 사이트로 사용할 수 있는 검색된 데이터 디스크 집합에 대해 설명합니다.
SR_RESOURCE_TYPE_QUERY_ELIGIBLE_TARGET_DATADISKS 복제를 위한 대상 사이트로 사용할 수 있는 검색된 데이터 디스크 집합에 대해 설명합니다.
SR_RESOURCE_TYPE_REPLICATED_DISK 복제된 디스크를 나타냅니다.

SR_RESOURCE_TYPE_REPLICATED_DISKS_RESULT 복제된 디스크의 검색된 집합에 대해 설명합니다.
SR_RESOURCE_TYPE_REPLICATED_PARTITION_ARRAY 디스크에 복제된 모든 파티션을 나열합니다.
SR_RESOURCE_TYPE_REPLICATED_PARTITION_INFO 복제된 파티션에 대해 설명합니다.
WitnessTagHelper PaxosTagCStruct 구조체의 유효성을 검사하는 데 사용되는 정보를 포함합니다.
WitnessTagUpdateHelper PaxosTagCStruct 구조체를 업데이트하고 유효성을 검사하는 데 사용되는 정보를 포함합니다.

Windows 색 시스템

아티클2023. 03. 09.

Windows Color System 기술 개요입니다.

Windows Color System을 개발하려면 다음 헤더가 필요합니다.

- wcsplugin.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 색 시스템

열거형

BMFORMAT **BMFORMAT** 열거형 형식의 값은 여러 WCS 함수에서 특정 비트맵이 있는 형식을 나타내는 데 사용됩니다.
COLORDATATYPE WCS 함수에서 벡터 콘텐츠의 데이터 형식을 나타내는 데 사용됩니다.
COLORPROFILESUBTYPE 색 프로필의 하위 형식을 지정합니다.
COLORPROFILETYPE 색 프로필의 유형을 지정합니다.
COLORTYPE **COLORTYPE** 열거형의 값은 여러 WCS 함수에서 사용됩니다. **COLOR** 형식의 변수는 **COLORTYPE** 열거형으로 열거된 색 공간에 정의됩니다.
WCS_PROFILE_MANAGEMENT_SCOPE 프로필 관리 작업의 범위를 지정합니다(예: 디바이스와 프로필 연결).

Functions

AssociateColorProfileWithDeviceA 지정된 색 프로필을 지정된 디바이스와 연결합니다. (ANSI)
AssociateColorProfileWithDeviceW 지정된 색 프로필을 지정된 디바이스와 연결합니다. (유니코드)
CheckBitmapBits 지정된 비트맵의 픽셀이 지정된 변환의 출력 영역 내에 있는지 여부를 확인합니다.
CheckColors 배열의 색이 지정된 변환의 출력 영역 내에 있는지 여부를 결정합니다.
CheckColorsInGamut CheckColorsInGamut 함수는 지정된 RGB 삼중 집합이 지정된 디바이스의 출력 영역 내에 있는지 여부를 결정합니다. RGB 삼중은 입력 논리 색 공간에서 해석됩니다.
CloseColorProfile 열린 프로필 핸들을 닫습니다.
CMCheckColors 지정된 색이 지정된 변환의 출력 영역 내에 있는지 여부를 결정합니다.
CMCheckColorsInGamut 지정된 RGB 트리플이 지정된 변환의 출력 영역 안에 있는지 여부를 확인합니다.
CMCheckRGB 출력 영역과 비트맵 색을 확인합니다.
CMConvertColorNameToIndex 명명된 색 공간의 색 이름을 색 프로필의 인덱스 번호로 변환합니다.
CMConvertIndexToColorName 색 공간의 인덱스를 명명된 색 공간의 이름 배열로 변환합니다. (CMConvertIndexToColorName)
CMCreateDeviceLinkProfile ICC 프로필 형식 사양에서 국제 색 컨소시엄에서 지정한 형식으로 디바이스 링크 프로필을 만듭니다.

CMCreateMultiProfileTransform 프로필 배열 또는 단일 디바이스 링크 프로필을 수락하고 색 변환을 만듭니다. 이 변환은 첫 번째 프로필에 지정된 색 공간에서 두 번째 프로필의 색 공간 등으로의 매핑입니다.
CMCreateProfile **LOGCOLORSPACEA** 구조체에서 표시 색 프로필을 만듭니다.
CMCreateProfileW **LOGCOLORSPACEW** 구조체에서 표시 색 프로필을 만듭니다.
CMCreateTransform 더 이상 사용되지 않습니다. 이 API는 더 이상 사용되지 않으므로 대체 API가 없습니다. 대체 CMM 모듈의 개발자는 이를 구현할 필요가 없습니다. (CMCreateTransform)
CMCreateTransformExt 변환을 만드는 방법을 정의하는 플래그 집합을 사용하여 입력 **LOGCOLORSPACEA** 에서 선택적 대상 공간으로 매핑한 다음 출력 디바이스로 매핑하는 색 변환을 만듭니다.
CMCreateTransformExtW 변환을 만드는 방법을 정의하는 플래그 집합을 사용하여 입력 **LOGCOLORSPACEW** 에서 선택적 대상 공간으로 매핑한 다음 출력 디바이스로 매핑하는 색 변환을 만듭니다.
CMCreateTransformW 더 이상 사용되지 않습니다. 이 API는 더 이상 사용되지 않으므로 대체 API가 없습니다. 대체 CMM 모듈의 개발자는 이를 구현할 필요가 없습니다. (CMCreateTransformW)
CMDeleteTransform 지정된 색 변환을 삭제하고 연결된 모든 메모리를 해제합니다.
CMGetInfo CMM(색 관리 모듈)에 대한 다양한 정보를 검색합니다.
CMGetNamedProfileInfo 지정된 명명된 색 프로필에 대한 정보를 검색합니다.
CMGetPS2ColorRenderingDictionary CMGetPS2ColorRenderingDictionary 및 이와 연결된 추가 매개 변수를 결정해야 합니다.

CMGetPS2ColorRenderingIntent 프로필에서 PostScript 수준 2 색 렌더링 의도 를 검색합니다.
CMGetPS2ColorSpaceArray CMGetPS2ColorSpaceArray 및 관련 매개 변수, 반환 및 설명이 결정되어야 합니다.
CMIsProfileValid 지정된 프로필이 색 관리에 사용할 수 있는 유효한 ICC 프로필인지 여부를 보고합니다.
CMTranslateColors 색 변환을 사용하여 원본 색 공간에서 대상 색 공간 으로 색 배열을 변환합니다.
CMTranslateRGB 애플리케이션에서 제공하는 RGBQuad를 디바이스 색 공간으로 변환합니다.
CMTranslateRGB 색 변환을 사용하여 비트맵을 한 색 공간에서 다른 색 공간 으로 변환합니다.
CMTranslateRGBsExt 정의된 형식의 비트맵을 다른 정의된 형식으로 변환하고 지정된 경우 콜백 함수를 주기적으로 호출하여 진행 상황을 보고하고 호출 애플리케이션이 번역을 종료하도록 허용합니다.
Cmyk CMYK 매크로는 지정된 시안, 마젠타, 노랑 및 검은색 값을 결합하여 CMYK 색 값을 만듭니다.
ColorCorrectPalette ColorCorrectPalette 함수는 지정된 디바이스 컨텍스트에서 WCS 1.0 매개 변수를 사용하여 팔레트의 항목을 수정합니다.
ColorimetricToDeviceColors 지정된 수의 색, 채널, 디바이스 색 및 독점 플러그 인 알고리즘에 대한 응답으로 적절한 XYZ 색을 반환합니다. (IDeviceModelPlugIn.ColorimetricToDeviceColors)
ColorimetricToDeviceColorsWithBlack 들어오는 수의 색, 채널, 검은색 정보, Commission Internationale l'Eclairge XYZ(CIEXYZ) 색 및 독점 플러그 인 알고리즘에 대한 응답으로 적절한 디바이스 색을 반환합니다.

ColorMatchToTarget ColorMatchToTarget 함수를 사용하면 대상 디바이스에 표시되는 색을 미리 볼 수 있습니다.
ColorProfileAddDisplayAssociation ColorProfileAddDisplayAssociation은 설치된 색 프로필을 지정된 범위의 지정된 디스플레이와 연결합니다.
ColorProfileGetDisplayDefault ColorProfileGetDisplayDefault는 지정된 범위의 지정된 디스플레이에 대한 기본 색 프로필을 가져옵니다.
ColorProfileGetDisplayList ColorProfileGetDisplayList는 지정된 범위에서 지정된 디스플레이와 연결된 프로필 목록을 검색합니다.
ColorProfileGetDisplayUserScope ColorProfileGetDisplayUserScope는 제공된 디스플레이의 현재 선택된 색 프로필 범위(사용자 또는 시스템)를 가져옵니다.
ColorProfileRemoveDisplayAssociation ColorProfileRemoveDisplayAssociation은 지정된 범위의 지정된 디스플레이에서 설치된 색 프로필을 분리합니다.
ColorProfileSetDisplayDefaultAssociation ColorProfileSetDisplayDefaultAssociation은 지정된 범위에서 지정된 디스플레이의 기본 프로필로 설치된 색 프로필을 설정합니다.
ConvertColorNameToIndex 명명된 색 공간의 색 이름을 ICC(International Color Consortium) 색 프로필의 인덱스 번호로 변환합니다.
ConvertIndexToColorName 색 공간의 인덱스를 명명된 색 공간의 이름 배열로 변환합니다. (ConvertIndexToColorName)
CreateColorSpaceA CreateColorSpace 함수는 논리적 색 공간을 만듭니다. (ANSI)

CreateColorSpaceW CreateColorSpace 함수는 논리적 색 공간을 만듭니다. (유니코드)
CreateColorTransformA 애플리케이션에서 색 관리를 수행하는 데 사용할 수 있는 색 변환을 만듭니다. (ANSI)
CreateColorTransformW 애플리케이션에서 색 관리를 수행하는 데 사용할 수 있는 색 변환을 만듭니다. (유니코드)
CreateDeviceLinkProfile 지정된 의도를 사용하여 색상 프로필 집합에서 ICC(International Color Consortium) *디바이스 링크 프로필을* 만듭니다.
CreateMultiProfileTransform 프로필 배열 또는 단일 디바이스 링크 프로필을 수락하고 애플리케이션에서 색 매핑을 수행하는 데 사용할 수 있는 색 변환을 만듭니다.
CreateProfileFromLogColorSpaceA 논리적 색 공간을디바이스 프로필로 변환합니다. (ANSI)
CreateProfileFromLogColorSpaceW 논리적 색 공간을디바이스 프로필로 변환합니다. (유니코드)
DeleteColorSpace DeleteColorSpace 함수는 지정된 색 공간을 제거하고 삭제합니다.
DeleteColorTransform 지정된 색 변환을 삭제합니다.
DeviceToColorimetricColors 지정된 수의 색, 채널, 디바이스 색 및 전용 플러그 인 알고리즘에 대한 응답으로 적절한 XYZ 색을 반환합니다. (IDeviceModelPlugIn.DeviceToColorimetricColors)
DisassociateColorProfileFromDeviceA 지정된 컴퓨터에서 지정된 디바이스와 지정된 색 프로필을 연결 해제합니다. (ANSI)

DisassociateColorProfileFromDeviceW 지정된 컴퓨터에서 지정된 디바이스와 지정된 색 프로필을 연결 해제합니다. (유니코드)
EnumColorProfilesA 지정된 열거형 조건을 충족하는 모든 프로필을 열거합니다. (ANSI)
EnumColorProfilesW 지정된 열거형 조건을 충족하는 모든 프로필을 열거합니다. (유니코드)
EnumICMProfilesA EnumICMProfiles 함수는 시스템에서 지정된 디바이스 컨텍스트에 대해 지원하는 다양한 출력 색 프로필을 열거합니다. (ANSI)
EnumICMProfilesW EnumICMProfiles 함수는 시스템에서 지정된 디바이스 컨텍스트에 대해 지원하는 다양한 출력 색 프로필을 열거합니다. (유니코드)
GetCMMInfo 지정된 색 변환을 만든 CMM(색 관리 모듈)에 대한 다양한 정보를 검색합니다.
GetColorDirectoryA 지정된 컴퓨터에서 Windows COLOR 디렉터리의 경로를 검색합니다. (ANSI)
GetColorDirectoryW 지정된 컴퓨터에서 Windows COLOR 디렉터리의 경로를 검색합니다. (유니코드)
GetColorProfileElement 지정된 색 프로필의 지정된 태그가 지정된 프로필 요소에서 버퍼로 데이터를 복사합니다.
GetColorProfileElementTag 지정된 ICC(International Color Consortium) 색 프로필의 태그 테이블에서 *dwIndex* 로 지정된 태그 이름을 검색합니다. 여기서 *dwIndex* 는 해당 테이블에 대한 하나의 기반 인덱스입니다.
GetColorProfileFromHandle 열려 있는 색 프로필에 대한 핸들이 제공되면 **GetColorProfileFromHandle** 함수는 프로필의 내용을 애플리케이션에서 제공하는 버퍼에 복사합니다. 핸들이 WCS(Windows Color System) 핸들인 경우 DMP가 반환되고 HPROFILE과 연결된 CAMP 및 GMMP가 무시됩니다.

GetColorProfileHeader ICC 색 프로필 또는 WCS XML 프로필에서 ICC 헤더 구조를 검색하거나 파생합니다. 드라이버와 애플리케이션은 **TRUE** 를 반환한다고 가정해야 하며, 이는 올바르게 구조화된 헤더가 반환됨을 나타냅니다. 각 태그는 레거시 ICM2 API 또는 XML 스키마 API를 사용하여 독립적으로 유효성을 검사해야 합니다.
GetColorSpace GetColorSpace 함수는 지정된 디바이스 컨텍스트에서 입력 색 공간에 대한 핸들을 검색합니다.
GetCountColorProfileElements 지정된 색 프로필에서 태그가 지정된 요소의 수를 검색합니다.
GetCValue GetCValue 매크로는 CMYK 색 값에서 시안 색 값을 검색합니다.
GetDeviceGammaRamp GetDeviceGammaRamp 함수는 하드웨어에서 다운로드 가능한 감마 램프를 지원하는 드라이버가 있는 직접 색 디스플레이 보드의 감마 램프를 가져옵니다.
GetGamutBoundaryMesh 플러그 인에서 삼각형 메시를 반환합니다. 이 함수는 GamutBoundaryDescription을 계산하는 데 사용됩니다.
GetGamutBoundaryMeshSize GetGamutBoundaryMesh 함수에 필요한 데이터 구조 크기를 반환합니다.
GetICMProfileA GetICMProfile 함수는 지정된 디바이스 컨텍스트에 대한 현재 출력 색 프로필의 파일 이름을 검색합니다. (ANSI)
GetICMProfileW GetICMProfile 함수는 지정된 디바이스 컨텍스트에 대한 현재 출력 색 프로필의 파일 이름을 검색합니다. (유니코드)
GetKValue GetKValue 매크로는 CMYK 색 값에서 검은색 값을 검색합니다.

GetLogColorSpaceA GetLogColorSpace 함수는 지정된 핸들로 식별되는 색 공간 정의를 검색합니다. (ANSI)
GetLogColorSpaceW GetLogColorSpace 함수는 지정된 핸들로 식별되는 색 공간 정의를 검색합니다. (유니코드)
GetMValue GetMValue 매크로는 CMYK 색 값에서 마젠타 색 값을 검색합니다.
GetNamedProfileInfo 첫 번째 매개 변수에 지정된 ICC(International Color Consortium) 명명된 색 프로필에 대한 정보를 검색합니다.
GetNeutralAxis IDeviceModelPlugIn::GetNeutralAxis는 디바이스의 중립 축을 따라 샘플 요소의 XYZ 색 측정을 반환합니다.
GetNeutralAxisSize IDeviceModelPlugIn::GetNeutralAxisSize 함수는 GetNeutralAxis 함수에서 반환하는 중립 축을 따라 데이터 요소 수를 반환합니다.
GetNumChannels 매개 변수 pNumChannels의 디바이스 채널 수를 반환합니다.
GetPrimarySamples 기본 샘플의 측정 색을 반환합니다.
GetPS2ColorRenderingDictionary 지정된 ICC 색 프로필에서 PostScript 수준 2 색 렌더링 사전을 검색합니다.
GetPS2ColorRenderingIntent ICC 색 프로필에서 PostScript 수준 2 색 렌더링 의도 를 검색합니다.
GetPS2ColorSpaceArray ICC 색 프로필에서 PostScript 수준 2 색 공간 배열을 검색합니다.

GetStandardColorSpaceProfileA 지정된 표준 색 공간에 대해 등록된 색 프로필을 검색 합니다. (ANSI)
GetStandardColorSpaceProfileW 지정된 표준 색 공간에 대해 등록된 색 프로필을 검색 합니다. (유니코드)
GetYValue GetYValue 매크로는 CMYK 색 값에서 노란색 색 값을 검색합니다.
ICMENUMPROCA EnumICMProfilesProcCallback 콜백은 EnumICMProfiles에서 색 프로필 데이터를 처리하는 애플리케이션 정의 콜백 함수입니다. (ANSI)
ICMENUMPROCW EnumICMProfilesProcCallback 콜백은 EnumICMProfiles에서 색 프로필 데이터를 처리하는 애플리케이션 정의 콜백 함수입니다. (유니코드)
초기화 전체 디바이스 모델 플러그 인을 입력으로 포함하는 Stream에 대한 포인터를 사용하고 플러그 인에 필요한 내부 매개 변수를 초기화합니다.
초기화 지정된 원본 및 대상 영역 경계 설명과 선택적 원본 및 대상 디바이스 모델 플러그 인을 사용하여 GMMP(영역 맵 모델 프로필)를 초기화합니다.
InstallColorProfileA 지정된 컴퓨터에 사용할 지정된 프로필을 설치합니다. 프로필도 COLOR 디렉터리에 복사됩니다. (ANSI)
InstallColorProfileW 지정된 컴퓨터에 사용할 지정된 프로필을 설치합니다. 프로필도 COLOR 디렉터리에 복사됩니다. (유니코드)
IsColorProfileTagPresent 지정된 ICC(International Color Consortium) 태그가 지정된 색 프로필에 있는지 여부를 보고합니다.

IsColorProfileValid 지정된 프로필이 유효한 ICC(International Color Consortium) 프로필인지 아니면 색 관리에 사용할 수 있는 유효한 WCS(Windows Color System) 프로필 핸들인지 확인할 수 있습니다.
OpenColorProfileA 지정된 색 프로필에 대한 핸들을 만듭니다. 그런 다음 핸들을 다른 프로필 관리 함수에서 사용할 수 있습니다. (ANSI)
OpenColorProfileW 지정된 색 프로필에 대한 핸들을 만듭니다. 그런 다음 핸들을 다른 프로필 관리 함수에서 사용할 수 있습니다. (유니코드)
PBMCALLBACKFN TBD(PBMCALLBACKFN)
PCMSCALLBACKA *PCMSCALLBACKA*(또는 **ApplyCallbackFunction**)는 SetupColorMatchingW 함수가 표시하는 대화 상자가 실행되는 동안 WCS 구성 데이터를 업데이트하는 콜백 함수입니다.
PCMSCALLBACKW *PCMSCALLBACKW*(또는 **ApplyCallbackFunction**)는 SetupColorMatchingW 함수가 표시하는 대화 상자가 실행되는 동안 WCS 구성 데이터를 업데이트하는 콜백 함수입니다.
RegisterCMMA 지정된 식별 값을 지정된 색 관리 모듈 CMM DLL(동적 연결 라이브러리)과 연결합니다. 이 ID가 색 프로필에 표시되면 Windows에서 해당 CMM을 찾아 변환을 만들 수 있습니다. (ANSI)
RegisterCMMW 지정된 식별 값을 지정된 색 관리 모듈 CMM DLL(동적 연결 라이브러리)과 연결합니다. 이 ID가 색 프로필에 표시되면 Windows에서 해당 CMM을 찾아 변환을 만들 수 있습니다. (유니코드)
SelectCMM 사용할 기본 설정 CMM(색 관리 모듈)을 선택할 수 있습니다.
SetColorProfileElement ICC 색 프로필에서 태그가 지정된 프로필 요소의 요소 데이터를 설정합니다.

SetColorProfileElementReference 지정된 ICC 색 프로필에서 기존 태그와 동일한 데이터를 참조하는 새 태그를 만듭니다.
SetColorProfileElementSize ICC 색 프로필에서 태그가 지정된 요소의 크기를 설정합니다.
SetColorProfileHeader 지정된 ICC 색 프로필에서 헤더 데이터를 설정합니다.
SetColorSpace SetColorSpace 함수는 지정된 디바이스 컨텍스트에 대한 입력 색 공간을 정의합니다.
SetDeviceGammaRamp SetDeviceGammaRamp 함수는 하드웨어에서 다운로드 가능한 감마 램프를 지원하는 드라이버가 있는 직접 색 디스플레이 보드의 감마 램프를 설정합니다.
SetICMMode SetICMMode 함수를 사용하면 지정된 DC(디바이스 컨텍스트)에서 이미지 색 관리를 사용하거나 사용하지 않도록 설정하거나 쿼리할 수 있습니다.
SetICMProfileA SetICMProfile 함수는 지정된 색 프로필을 지정된 DC(디바이스 컨텍스트)의 출력 프로필로 설정합니다. (ANSI)
SetICMProfileW SetICMProfile 함수는 지정된 색 프로필을 지정된 DC(디바이스 컨텍스트)의 출력 프로필로 설정합니다. (유니코드)
SetStandardColorSpaceProfileA 지정된 표준 색 공간에 대해 지정된 프로필을 등록합니다. GetStandardColorSpaceProfileW를 사용하여 프로필을 쿼리할 수 있습니다. (ANSI)
SetStandardColorSpaceProfileW 지정된 표준 색 공간에 대해 지정된 프로필을 등록합니다. GetStandardColorSpaceProfileW를 사용하여 프로필을 쿼리할 수 있습니다. (유니코드)

SetTransformDeviceModelInfo 플러그 인에 매개 변수를 제공하여 변환 시퀀스에서 특정 플러그 인이 발생하는 위치를 결정합니다.
SetupColorMatchingA 사용자가 색 관리를 사용할지 여부를 선택할 수 있는 색 관리 대화 상자를 만들고, 이 경우 사용된 색 프로필과 렌더링 의도를 제어할 수 있습니다. (ANSI)
SetupColorMatchingW 사용자가 색 관리를 사용할지 여부를 선택할 수 있는 색 관리 대화 상자를 만들고, 이 경우 사용된 색 프로필과 렌더링 의도를 제어할 수 있습니다. (유니코드)
SourceToDestinationAppearanceColors 지정된 수의 색 및 CIEJCh 색에 대한 응답으로 적절한 영역 매핑 모양 색을 반환합니다.
TranslateBitmapBits 요청된 형식으로 다른 비트맵을 생성하도록 정의된 형식을 갖는 비트맵의 색을 변환합니다.
TranslateColors 색 변환에 정의된 대로 원본 색 공간에서 대상 색 공간으로 색 배열을 변환합니다.
UninstallColorProfileA 지정된 컴퓨터에서 지정된 색 프로필을 제거합니다. 연결된 파일은 선택적으로 시스템에서 삭제됩니다. (ANSI)
UninstallColorProfileW 지정된 컴퓨터에서 지정된 색 프로필을 제거합니다. 연결된 파일은 선택적으로 시스템에서 삭제됩니다. (유니코드)
UNREgisterCMMA 지정된 CMM DLL(색 관리 모듈 동적 연결 라이브러리)에서 지정된 ID 값을 분리합니다. (ANSI)
UnregisterCMMW 지정된 CMM DLL(색 관리 모듈 동적 연결 라이브러리)에서 지정된 ID 값을 분리합니다. (유니코드)
UpdateICMRegKeyA UpdateICMRegKey 함수는 시스템의 색 프로필 및 색 관리 모듈을 관리합니다. (ANSI)

UpdateICMRegKeyW UpdateICMRegKey 함수는 시스템의 색 프로필 및 색 관리 모듈을 관리합니다. (유니코드)
WcsAssociateColorProfileWithDevice WcsAssociateColorProfileWithDevice는 지정된 WCS 색 프로필을 지정된 디바이스와 연결합니다.
WcsCheckColors 배열의 색이 지정된 WCS 색 변환의 출력 영역 내에 있는지 여부를 결정합니다.
WcsCreateIccProfile WCS 프로필을 ICC(International Color Consortium) 프로필로 변환합니다.
WcsDisassociateColorProfileFromDevice 컴퓨터의 지정된 디바이스에서 지정된 WCS 색 프로필을 연결 해제합니다.
WcsEnumColorProfiles 지정된 프로필 관리 범위의 열거 조건을 충족하는 모든 색 프로필을 열거합니다.
WcsEnumColorProfilesSize **WcsEnumColorProfiles** 함수가 색 프로필을 열거하는 데 필요한 버퍼의 크기(바이트)를 반환합니다.
WcsGetCalibrationManagementState 디스플레이 보정 상태의 시스템 관리를 사용할 수 있는지 여부를 결정합니다.
WcsGetDefaultColorProfile 디바이스에 대한 기본 색 프로필을 검색하거나 디바이스가 지정되지 않은 경우 디바이스 독립적 기본값에 대해 검색합니다.
WcsGetDefaultColorProfileSize 디바이스에 대한 기본 색 프로필 이름(**NULL** 종결자 포함)의 크기를 바이트 단위로 반환합니다.
WcsGetDefaultRenderingIntent 지정된 프로필 관리 범위에서 기본 렌더링 의도를 검색합니다.

WcsGetUsePerUserProfiles 사용자가 지정된 디바이스에 대해 사용자별 프로필 연결 목록을 사용하도록 선택했는지 여부를 결정합니다.
WcsOpenColorProfileA 지정된 색 프로필에 대한 핸들을 만듭니다. (ANSI)
WcsOpenColorProfileW 지정된 색 프로필에 대한 핸들을 만듭니다. (유니코드)
WcsSetCalibrationManagementState 디스플레이 보정 상태의 시스템 관리를 사용하거나 사용하지 않도록 설정합니다.
WcsSetDefaultColorProfile 지정된 프로필 관리 범위에서 지정된 프로필 형식의 기본 색 프로필 이름을 설정합니다.
WcsSetDefaultRenderingIntent 지정된 프로필 관리 범위에서 기본 렌더링 의도를 설정합니다.
WcsSetUsePerUserProfiles 사용자가 지정된 디바이스에 대해 사용자별 프로필 연결 목록을 사용할지 여부를 지정할 수 있습니다.
WcsTranslateColors 색 변환에 정의된 대로 원본 색 공간에서 대상 색 공간으로 색 배열을 변환합니다.

인터페이스

IDeviceModelPlugIn IDeviceModelPlugIn COM(구성 요소 개체 모델) 인터페이스에 대해 정의된 메서드에 대해 설명합니다.
IGamutMapModelPlugIn IGamutMapModelPlugIn COM(구성 요소 개체 모델) 인터페이스에 대해 정의된 메서드에 대해 설명합니다.

구조체

BlackInformation 검은색 채널이 있는 디바이스 모델에 대한 정보를 포함합니다.
CIEXYZ CIEXYZ 구조체에는 지정된 색 공간에서 특정 색의 x, y 및 z 좌표가 포함됩니다.
CIEXYZTRIPLE CIEXYZTRIPLE 구조체에는 지정된 논리 색 공간에 대한 빨간색, 녹색 및 파란색 엔드포인트에 해당하는 세 가지 색의 x,y 및 z 좌표가 포함됩니다.
CMYKCOLOR CMYKCOLOR 구조체에 대한 설명입니다.
색 COLOR 공용 구조체에 대한 설명입니다.
COLORMATCHSETUPA **COLORMATCHSETUP** 구조에는 **SetupColorMatchingW** 함수가 **ColorManagement** 대화 상자를 초기화하는 데 사용하는 정보가 포함되어 있습니다. (ANSI)
COLORMATCHSETUPW **COLORMATCHSETUP** 구조에는 **SetupColorMatchingW** 함수가 **ColorManagement** 대화 상자를 초기화하는 데 사용하는 정보가 포함되어 있습니다. (유니코드)
ENUMTYPEA 프로필 열거형 제약 조건을 정의하는 정보를 포함합니다. (ANSI)
ENUMTYPEW 프로필 열거형 제약 조건을 정의하는 정보를 포함합니다. (유니코드)
GamutBoundaryDescription 영역 경계를 정의합니다.

GamutShell 인덱싱된 삼각형 목록으로 표시되는 영역 셸을 정의하는 정보를 포함합니다. 꼭짓점 버퍼에는 꼭짓점 데이터가 포함됩니다.
GamutShellTriangle 꼭짓점 버퍼에 액세스하기 위한 세 가지 꼭짓점 인덱스를 포함합니다.
GENERIC3CHANNEL TBD(GENERIC3CHANNEL)
GRAYCOLOR GRAYCOLOR 구조체에 대한 설명입니다.
HiFiCOLOR HiFiCOLOR 구조체에 대한 설명입니다.
JabColorF JabColorF(wcsplugin.h)는 구조체입니다.
JChColorF JChColorF(wcsplugin.h)는 구조체입니다.
LabCOLOR TBD(LabCOLOR)
LOGCOLORSPACEA LOGCOLORSPACE 구조에는 논리적 색 공간을 정의하는 정보가 포함되어 있습니다. (ANSI)
LOGCOLORSPACEW LOGCOLORSPACE 구조에는 논리적 색 공간을 정의하는 정보가 포함되어 있습니다. (유니코드)
NAMED_PROFILE_INFO **NAMED_PROFILE_INFO** 구조는 명명된 색 프로필에 대한 정보를 저장하는 데 사용됩니다.
NAMEDCOLOR TBD(NAMEDCOLOR)

PrimaryJabColors 이 구조는 Jab 좌표에 8가지 기본 색을 포함합니다.
PrimaryXYZColors 이 구조체는 XYZ 좌표에 8가지 기본 색을 포함합니다.
프로필 색 프로필을 정의하는 정보를 포함합니다.
PROFILEHEADER 디바이스 프로필 파일의 내용을 설명하는 정보를 포함합니다. 이 헤더는 디바이스 프로필 파일의 시작 부분에서 발생합니다.
RGBCOLOR TBD(RGBCOLOR)
XYZCOLOR TBD(XYZCOLOR)
XYZColorF XYZColorF(wcsplugin.h)는 구조체입니다.
YxyCOLOR TBD(YxyCOLOR)

Windows Connect Now

아티클2023. 03. 16.

Windows Connect Now 기술 개요입니다.

Windows Connect Now 개발하려면 다음 헤더가 필요합니다.

- wcndevice.h
- wcntypes.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows Connect Now

열거형

WCN_ATTRIBUTE_TYPE WCN_ATTRIBUTE_TYPE 열거형은 Wi-Fi 보호된 설치에 대해 정의된 특성 버퍼 형식을 정의합니다. 각 특성 버퍼가 차지하는 전체 크기에는 추가 4바이트(ID 2바이트, 길이 2바이트)가 포함됩니다.
WCN_PASSWORD_TYPE WCN_PASSWORD_TYPE 열거형은 WPS 세션에서 사용할 인증을 정의합니다.
WCN_SESSION_STATUS WPS 세션의 결과 상태를 정의합니다.
WCN_VALUE_TYPE_ASSOCIATION_STATE WCN_VALUE_TYPE_ASSOCIATION_STATE 열거형은 검색 요청 중에 무선 스테이션의 가능한 연결 상태를 정의합니다.
WCN_VALUE_TYPE_AUTHENTICATION_TYPE WCN_VALUE_TYPE_AUTHENTICATION_TYPE 열거형은 등록자(액세스 지점 또는 스테이션)에서 지원하는 인증 유형을 정의합니다.
WCN_VALUE_TYPE_BOOLEAN WCN_VALUE_TYPE_BOOLEAN 열거형은 디바이스 설정 및 연결 중에 발생하는 true/false 조건을 나타내는 데 사용되는 값을 정의합니다.

WCN_VALUE_TYPE_CONFIG_METHODS WCN_VALUE_TYPE_CONFIG_METHODS 열거형은 등록자 또는 등록 기관에서 지원하는 구성 메서드를 정의합니다.
WCN_VALUE_TYPE_CONFIGURATION_ERROR WCN_VALUE_TYPE_CONFIGURATION_ERROR 열거형은 WLAN을 구성하고 연결하는 동안 디바이스에 반환되는 가능한 오류 값을 정의합니다.
WCN_VALUE_TYPE_CONNECTION_TYPE WCN_VALUE_TYPE_CONNECTION_TYPE.
WCN_VALUE_TYPE_DEVICE_PASSWORD_ID WCN_VALUE_TYPE_DEVICE_PASSWORD_ID 열거형은 암호의 원본 또는 '형식'을 지정하는 값을 정의합니다.
WCN_VALUE_TYPE_ENCRYPTION_TYPE WCN_VALUE_TYPE_ENCRYPTION_TYPE 열거형은 지원되는 WLAN 암호화 유형을 정의합니다.
WCN_VALUE_TYPE_RF_BANDS WCN_VALUE_TYPE_RF_BANDS 열거형은 등록자가 검색 요청을 보낼 수 있는 가능한 무선 주파수 대역을 정의합니다.
WCN_VALUE_TYPE_VERSION 지원되는 WPS(Wi-Fi Protected Setup) 버전을 정의합니다.
WCN_VALUE_TYPE_WI_FI_PROTECTED_SETUP_STATE WCN_VALUE_TYPE_WI_FI_PROTECTED_SETUP_STATE 열거형입니다.

Functions

연결 IWCNDevice::Connect 메서드가 세션을 시작합니다.
ConnectFailed 콜백 메서드는 IWCNDevice::Connect 실패를 나타냅니다.

ConnectSucceeded 성공적인 IWCNDevice::Connect 작업을 나타내는 IWCNConnectNotify::ConnectSucceeded 콜백 메서드입니다.
GetAttribute IWCNDevice::GetAttribute 메서드는 디바이스에서 캐시된 특성을 가져옵니다.
GetIntegerAttribute GetIntegerAttribute 메서드는 디바이스에서 정수로 캐시된 특성을 가져옵니다.
GetNetworkProfile IWCNDevice::GetNetworkProfile 메서드는 디바이스에서 네트워크 프로필을 가져옵니다.
GetStringAttribute IWCNDevice::GetStringAttribute 메서드는 디바이스에서 캐시된 특성을 문자열로 가져옵니다.
GetVendorExtension GetVendorExtension 메서드는 디바이스에서 캐시된 공급업체 확장을 가져옵니다.
SetNetworkProfile IWCNDevice::SetNetworkProfile 메서드는 디바이스에 프로비전할 XML WLAN 프로필을 큐에 대기합니다. 이 메서드는 IWCNDevice::Connect 이전에만 호출할 수 있습니다.
SetPassword IWCNDevice::SetPassword 메서드는 인증 방법 값과 필요한 경우 보류 중인 세션에 사용되는 암호를 구성합니다. 이 메서드는 IWCNDevice::Connect 이전에만 호출할 수 있습니다.
SetVendorExtension IWCNDevice::SetVendorExtension 메서드는 보류 중인 세션에서 사용할 공급업체 확장을 큐에 대기합니다. 이 함수는 IWCNDevice::Connect 이전에만 호출될 수 있습니다.
Unadvise IWCNDevice::Unadvise 메서드는 IWCNDevice::Connect를 통해 이전에 설정한 콜백을 제거합니다.

인터페이스

IWCNConnectNotify Windows Connect Now 연결 세션이 완료되면 이 인터페이스를 사용하여 성공 또는 실패 알림을 받습니다.
IWCNDevice 이 인터페이스를 사용하여 디바이스를 구성하고 세션을 시작합니다.

구조체

WCN_VALUE_TYPE_PRIMARY_DEVICE_TYPE WCN_VALUE_TYPE_PRIMARY_DEVICE_TYPE 구조에는 범주, 하위 범주 및 제조업체별 OUI(조직 ID)에 따라 디바이스 유형을 식별하는 정보가 포함됩니다.
WCN_VENDOR_EXTENSION_SPEC WCN_VENDOR_EXTENSION_SPEC 구조에는 공급업체 확장을 정의하는 데이터가 포함되어 있습니다.

Windows 연결 관리자

아티클2023. 03. 11.

Windows 연결 관리자 기술 개요입니다.

Windows 연결 관리자 개발하려면 다음 헤더가 필요합니다.

- wcmapi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 연결 관리자

열거형

WCM_CONNECTION_COST 연결 비용 유형 및 플래그를 결정합니다.
WCM_CONNECTION_COST_SOURCE 연결 비용 정보를 제공하는 원본을 지정합니다.
WCM_MEDIA_TYPE 연결에 대한 미디어 유형을 지정합니다.
WCM_PROPERTY 연결의 속성을 지정합니다.

Functions

WcmFreeMemory WCM 함수에서 할당한 메모리 리소스를 해제하는 데 사용됩니다.
WcmGetProfileList 기본 설정 순서로 프로필 목록을 검색합니다.

WcmQueryProperty 지정된 WCM 속성의 값을 검색합니다.
WcmSetProfileList 프로필 목록 또는 프로필 목록의 하위 집합을 다시 정렬합니다.
WcmSetProperty WCM 속성의 값을 설정합니다.

구조체

WCM_BILLING_CYCLE_INFO 청구 주기에 대한 정보를 지정합니다.
WCM_CONNECTION_COST_DATA 연결 비용에 대한 정보를 지정합니다.
WCM_DATAPLAN_STATUS 네트워크 연결에 대한 구독 정보를 지정합니다.
WCM_POLICY_VALUE 정책의 현재 값에 대한 정보를 포함합니다.
WCM_PROFILE_INFO 특정 프로필에 대한 정보를 포함합니다.
WCM_PROFILE_INFO_LIST 기본 설정 순서로 프로필 목록을 포함합니다.
WCM_TIME_INTERVAL 시간 간격을 정의합니다.
WCM_USAGE_DATA 연결 사용과 관련된 정보를 포함합니다.

Windows 연락처

아티클2023. 03. 09.

Windows 연락처 기술 개요입니다.

Windows 연락처를 개발하려면 다음 헤더가 필요합니다.

- icontact.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 연락처

Functions

CommitChanges 이 연락처의 변경 내용을 연락처 파일에 저장합니다.
CreateArrayNode 다중 값 속성에 새 배열 노드를 만듭니다.
DeleteArrayNode 지정된 배열 항목의 데이터를 삭제합니다.
DeleteLabels 지정된 배열 항목에서 레이블을 삭제합니다.
DeleteProperty 지정된 속성의 값을 삭제합니다. 속성 수정 및 버전 데이터는 여전히 IContactPropertyCollection을 사용하여 열거할 수 있습니다.
GetBinary IStream 인터페이스 [Structured Storage]를 사용하여 속성의 이진 데이터를 검색합니다.
GetContactCollection 알려진 모든 연락처를 포함하는 IContactCollection 개체를 반환합니다.

GetContactID 로컬 컴퓨터 고유 연락처 ID를 검색합니다.
GetCurrent 열거형에서 현재 연락처를 검색합니다.
Getdate 지정된 속성의 날짜 및 시간 값을 호출자의 FILETIME 구조로 검색합니다. 모든 시간은 UTC(협정 세계시)로 저장되고 반환됩니다.
GetLabels 지정된 배열 요소 이름의 레이블을 검색합니다.
GetMeContact 'me'의 로컬 사용자 계정 개념을 검색합니다.
GetPath 이 연락처를 로드하는 데 사용되는 파일 시스템 경로를 검색합니다.
GetPropertyArrayElementID 속성 배열에서 지정된 요소의 고유 ID를 검색합니다.
GetPropertyCollection 현재 연락처에 대한 IContactPropertyCollection을 반환합니다. 필요에 따라 IContactPropertyCollection을 필터링하여 일부 값만 열거합니다.
GetPropertyModificationDate 열거형의 현재 속성에 대한 마지막 수정 날짜를 검색합니다. 수정하지 않으면 연락처 만들기 날짜가 반환됩니다.
GetPropertyName 열거형에서 현재 속성의 이름을 검색합니다.
GetPropertyType 열거형의 현재 속성에 대한 형식을 검색합니다.

GetPropertyVersion 열거형에서 현재 속성의 버전 번호를 검색합니다.
GetString 지정된 속성의 문자열 값을 호출자가 할당한 버퍼로 검색합니다.
초기화 연락처를 조작하는 데 사용되는 고유한 애플리케이션 이름 및 애플리케이션 버전을 사용하여 연락처 관리자를 초기화합니다.
로드 컴퓨터-로컬 연락처 ID에서 참조하는 연락처의 데이터를 사용하여 IContact 개체를 로드합니다.
MergeContactIDs 이전 연락처 ID를 새 연락처 ID와 동일한 값으로 확인합니다. IContactManager::이전 연락처 ID를 사용하여 로드하면 이제 새 연락처 ID 연락처가 로드됩니다.
다음 다음 연락처로 이동합니다.
다음 다음 속성으로 이동합니다.
재설정 열거자를 논리적 첫 번째 요소 이전으로 다시 설정합니다.
재설정 속성의 열거형을 다시 설정합니다.
SetBinary 지정된 속성의 이진 데이터를 Null로 종료된 문자열(MIME 형식) 데이터가 포함된 지정된 IStream 인터페이스 [Structured Storage]의 내용으로 설정합니다.
SetDate 지정된 속성의 날짜 및 시간 값을 지정된 FILETIME으로 설정합니다. 모든 시간은 UTC(협정 세계시)로 저장되고 반환됩니다.

SetLabels 지정된 속성의 레이블 집합에 전달된 레이블 집합을 추가합니다. 참고:_This 메서드는 중복 레이블을 확인하지 않습니다.
SetMeContact 'me'의 로컬 사용자 계정 개념을 지정된 사용자로 설정합니다.
SetString 지정된 속성의 문자열 값을 지정된 null로 끝나는 문자열의 문자열 값으로 설정합니다.

인터페이스

Icontact 사용하지 마십시오. 단일 연락처의 속성을 읽고 쓰는 메서드를 정의합니다.
IContactCollection 사용하지 마십시오. IContactManager에서 알려진 연락처를 열거합니다.
IContactManager 사용하지 마십시오. 연락처 ID 문자열을 기반으로 연락처를 검색하는 데 사용됩니다.
IContactProperties 사용하지 마십시오. IContact에서 속성을 검색, 설정, 만들기 및 제거하는 데 사용됩니다. 속성 이름 및 확장 메커니즘은 icontactproperties.h에 설명되어 있습니다.
IContactPropertyCollection 사용하지 마십시오. 레이블 또는 속성 집합을 기반으로 연락처 데이터를 필터링하는 데 사용됩니다. IContactProperties 개체로 노출된 연락처 속성을 열거합니다. 각 속성에 대해 이름, 형식, 버전 및 수정 날짜를 검색할 수 있습니다.

Windows 컨트롤

아티클2023. 06. 08.

Windows 컨트롤 기술 개요입니다.

Windows 컨트롤을 개발하려면 다음 헤더가 필요합니다.

- commctrl.h
- commoncontrols.h
- dpa_dsa.h
- prsht.h
- richedit.h
- richole.h
- shlobj.h
- textserv.h
- tom.h
- uxtheme.h
- windowsx.h
- winuser.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 컨트롤

클래스

ITextHost ITextHost 인터페이스는 텍스트 호스트 서비스를 가져오기 위해 텍스트 서비스 개체에서 사용됩니다.
ITextHost2 ITextHost2 인터페이스는 ITextHost 인터페이스를 확장합니다.
ITextServices TOM(텍스트 개체 모델)을 확장하여 창 없는 작업에 추가 기능을 제공합니다.
ITextServices2 ITextServices2 인터페이스는 ITextServices 인터페이스를 확장합니다.

열거형

BP_ANIMATIONSTYLE BP_ANIMATIONPARAMS 구조에서 애니메이션 옵션을 선언하는 데 사용됩니다.
BP_BUFFERFORMAT 버퍼의 형식을 지정합니다. BeginBufferedAnimation 및 BeginBufferedPaint에서 사용됩니다.
EC_ENDOFLINE 편집 컨트롤에서 사용하는 줄 문자의 끝을 나타냅니다.
EC_SEARCHWEB_ENTRYPOINT 웹 검색의 진입점을 나타내는 상수를 정의합니다.
FEEDBACK_TYPE 이벤트와 연결된 시각적 피드백을 지정합니다.
INPUT_MESSAGE_DEVICE_TYPE 입력 메시지를 보낸 디바이스의 유형입니다.
INPUT_MESSAGE_ORIGIN_ID 입력 메시지 원본의 ID입니다.
KHYPH 서식 있는 편집 컨트롤에서 하이픈을 만드는 방법을 지정하는 데 사용되는 값을 포함합니다. HyphenateProc 콜백 함수는 이 열거형 형식을 사용합니다.
MANCODE 수학적 영숫자 코드를 나타냅니다.
OBJECTTYPE TOM(텍스트 개체 모델) 콘텐츠에서 개체 형식을 식별하는 값을 정의합니다.
POINTER_DEVICE_CURSOR_TYPE 포인터 디바이스 커서 유형을 식별합니다.

POINTER_DEVICE_TYPE 포인터 디바이스 유형을 식별합니다.
POINTER_FEEDBACK_MODE CreateSyntheticPointerDevice에 사용할 수 있는 시각적 피드백 동작을 식별합니다.
PROPERTYORIGIN 속성이 발견된 위치를 지정하기 위해 GetThemePropertyOrigin에 의해 반환됩니다.
TEXTMODE 서식 있는 편집 컨트롤의 텍스트 모드를 나타냅니다. EM_SETTEXTMODE 및 EM_GETTEXTMODE 메시지는 이 열거형 형식을 사용합니다.
THEMESIZE 비주얼 스타일 부분에 대해 검색할 크기 값의 형식을 식별합니다.
tomConstants TOM(텍스트 개체 모델) API와 함께 사용되는 값을 정의합니다.
UNDONAMEID 실행 취소하거나 다시 실행할 수 있는 서식 있는 편집 컨트롤 작업의 형식을 나타내는 값을 포함합니다. EM_GETREDONAME 및 EM_GETUNDONAME 메시지는 이 열거형 형식을 사용하여 값을 반환합니다.
WINDOWTHEMEATTRIBUTETYPE 창에 설정할 시각적 스타일 특성의 형식을 지정합니다.

Functions

_NewEnum 이 스토리 컬렉션에 대한 IEnumVARIANT 열거자 인터페이스를 검색합니다.

ActivateAs 이전 클래스의 모든 개체를 언로드하고, 해당 개체를 새 클래스의 개체로 처리하도록 OLE에 알리고, 개체를 다시 로드하여 다른 이름으로 활성화 동작을 처리합니다. 개체를 다시 로드할 수 없는 경우 삭제됩니다.
추가 이미지 목록에 하나의 이미지 또는 여러 이미지를 추가합니다. (IImageList.Add)
추가 컬렉션의 끝에 문자열을 추가합니다.
AddMasked 이미지 목록에 하나의 이미지 또는 여러 이미지를 추가하여 지정된 비트맵에서 마스크를 생성합니다. (IImageList.AddMasked)
AddSubrange 이 범위에 하위 범위를 추가합니다.
AddTab tbAlign 형식과 지시문 스타일 tbLeader를 사용하여 변위 tbPos에 탭을 추가합니다.
Animate_Close AVI 클립을 닫습니다. 이 매크로를 사용하거나 NULL 매개 변수를 전달하여 ACM_OPEN 메시지를 명시적으로 보낼 수 있습니다.
Animate_Create 애니메이션 컨트롤을 만듭니다. Animate_Create는 CreateWindow 함수를 호출하여 애니메이션 컨트롤을 만듭니다.
Animate_IsPlaying Audio-Video 인터리브(AVI) 클립이 재생되고 있는지 확인합니다. 이 매크로를 사용하거나 ACM_ISPLAYING 메시지를 보낼 수 있습니다.
Animate_Open AVI 클립을 열고 애니메이션 컨트롤에 첫 번째 프레임을 표시합니다. 이 매크로를 사용하거나 ACM_OPEN 메시지를 명시적으로 보낼 수 있습니다.

Animate_OpenEx 지정된 모듈의 리소스에서 AVI 클립을 열고 애니메이션 컨트롤에 첫 번째 프레임을 표시합니다. 이 매크로를 사용하거나 ACM_OPEN 메시지를 명시적으로 보낼 수 있습니다.
Animate_Play 애니메이션 컨트롤에서 AVI 클립을 재생합니다. 스레드가 계속 실행되는 동안 컨트롤은 백그라운드에서 클립을 재생합니다. 이 매크로를 사용하거나 ACM_PLAY 메시지를 명시적으로 보낼 수 있습니다.
Animate_Seek 애니메이션 컨트롤을 지시하여 AVI 클립의 특정 프레임을 표시합니다. 컨트롤은 스레드가 계속 실행되는 동안 배경에 클립을 표시합니다. 이 매크로를 사용하거나 ACM_PLAY 메시지를 명시적으로 보낼 수 있습니다.
Animate_Stop 애니메이션 컨트롤에서 AVI 클립 재생을 중지합니다. 이 매크로를 사용하거나 ACM_STOP 메시지를 명시적으로 보낼 수 있습니다.
Append 컬렉션의 지정된 인덱스에 문자열을 추가합니다.
적용 이 텍스트 행 개체의 서식 특성을 연결된 ITextRange2의 지정된 행에 적용합니다.
AttachMsgFilter 편집 instance 새 메시지 필터를 연결합니다. 편집 instance 받는 모든 창 메시지는 메시지 필터로 전달됩니다.
AutoCorrectProc AutoCorrectProc 함수는 EM_SETAUTOCORRECTPROC 메시지와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
BeginBufferedAnimation 버퍼링된 애니메이션 작업을 시작합니다. 애니메이션은 지정된 시간 동안 두 버퍼의 콘텐츠 간 교차 페이드로 구성됩니다.
BeginBufferedPaint 버퍼링된 페인트 작업을 시작합니다.

BeginDrag 이미지 끌기를 시작합니다. (IImageList.BeginDrag)
BeginEditCollection 컬렉션 편집(실행 취소 그룹화라고도 함)을 켭니다.
BeginPanningFeedback 패닝 제스처의 영향을 받는 대상 창에 대한 피드백을 보내도록 시스템에 알립니다.
BufferedPaintClear 버퍼에서 지정된 사각형을 ARGB = {0,0,0,0}으로 지웁니다.
BufferedPaintInit 현재 스레드에 대해 버퍼링된 페인팅을 초기화합니다.
BufferedPaintRenderAnimation 버퍼링된 페인트 애니메이션의 다음 프레임을 그립니다.
BufferedPaintSetAlpha 알파를 지정된 사각형의 지정된 값으로 설정합니다. 알파는 대상 DC(대상 디바이스 컨텍스트)에서 버퍼와 혼합할 때 적용되는 투명도의 양을 제어합니다.
BufferedPaintStopAllAnimations 지정된 창에 대해 버퍼링된 모든 애니메이션을 중지합니다.
BufferedPaintUnInit 현재 스레드에 대한 버퍼링된 페인팅을 닫습니다. BeginBufferedPaint에 대한 호출이 더 이상 필요하지 않은 후 BufferedPaintInit에 대한 각 호출에 대해 한 번씩 호출됩니다.
BuildUpMath 범위의 선형 형식 수학을 내장형으로 변환하거나 현재 내장형을 수정합니다.
Button_Enable 단추를 사용하거나 사용하지 않도록 설정합니다.

Button_GetCheck 라디오 단추 또는 확인란의 선택 상태를 가져옵니다. 이 매크로를 사용하거나 BM_GETCHECK 메시지를 명시적으로 보낼 수 있습니다.
Button_GetIdealSize 이미지 목록이 있는 경우 텍스트와 이미지에 가장 잘 맞는 단추 크기를 가져옵니다. 이 매크로를 사용하거나 BCM_GETIDEALSIZE 메시지를 명시적으로 보낼 수 있습니다.
Button_GetImageList 단추 컨트롤에 설정된 이미지 목록을 설명하는 BUTTON_IMAGELIST 구조체를 가져옵니다. 이 매크로를 사용하거나 BCM_GETIMAGELIST 메시지를 명시적으로 보낼 수 있습니다.
Button_GetNote 명령 링크 단추와 연결된 메모의 텍스트를 가져옵니다. 이 매크로를 사용하거나 BCM_GETNOTE 메시지를 명시적으로 보낼 수 있습니다.
Button_GetNoteLength 명령 링크에 대한 설명에 표시될 수 있는 메모 텍스트의 길이를 가져옵니다. 이 매크로를 사용하거나 BCM_GETNOTELENGTH 메시지를 명시적으로 보냅니다.
Button_GetSplitInfo 지정된 분할 단추 컨트롤에 대한 정보를 가져옵니다. 이 매크로를 사용하거나 BCM_GETSPLITINFO 메시지를 명시적으로 보냅니다.
Button_GetState 단추 또는 확인란의 상태를 검색합니다. 이 매크로를 사용하거나 BM_GETSTATE 메시지를 명시적으로 보낼 수 있습니다.
Button_GetText 단추의 텍스트를 가져옵니다.
Button_GetTextLength 단추 텍스트의 문자 수를 가져옵니다.
Button_GetTextMargin 단추 컨트롤에서 텍스트를 그리는 데 사용되는 여백을 가져옵니다. 이 매크로를 사용하거나 BCM_GETTEXTMARGIN 메시지를 명시적으로 보낼 수 있습니다.

Button_SetCheck 라디오 단추 또는 확인란의 선택 상태를 설정합니다. 이 매크로를 사용하거나 BM_SETCHECK 메시지를 명시적으로 보낼 수 있습니다.
Button_SetDropDownState BS_SPLITBUTTON 스타일로 지정된 단추의 드롭다운 상태를 설정합니다. 이 매크로를 사용하거나 BCM_SETDROPDOWNSTATE 메시지를 명시적으로 보냅니다.
Button_SetElevationRequiredState 관리자 권한의 아이콘을 표시하기 위해 지정된 단추 또는 명령 링크에 대한 상승 필수 상태를 설정합니다. 이 매크로를 사용하거나 BCM_SETSHIELD 메시지를 명시적으로 보냅니다.
Button_SetImageList 단추 컨트롤에 이미지 목록을 할당합니다. 이 매크로를 사용하거나 BCM_SETIMAGELIST 메시지를 명시적으로 보낼 수 있습니다.
Button_SetNote 지정된 명령 링크 단추와 관련된 메모의 텍스트를 설정합니다. 이 매크로를 사용하거나 BCM_SETNOTE 메시지를 명시적으로 보낼 수 있습니다.
Button_SetSplitInfo 지정된 분할 단추 컨트롤에 대한 정보를 설정합니다. 이 매크로를 사용하거나 BCM_SETSPLITINFO 메시지를 명시적으로 보냅니다.
Button_SetState 단추의 강조 표시 상태를 설정합니다. 강조 상태는 사용자가 단추를 누른 것처럼 단추가 강조 표시되는지 여부를 나타냅니다. 이 매크로를 사용하거나 BM_SETSTATE 메시지를 명시적으로 보낼 수 있습니다.
Button_SetStyle 단추의 스타일을 설정합니다. 이 매크로를 사용하거나 BM_SETSTYLE 메시지를 명시적으로 보낼 수 있습니다.
Button_SetText 단추의 텍스트를 설정합니다.
Button_SetTextMargin 단추 컨트롤에서 텍스트를 그리기 위한 여백을 설정합니다. 이 매크로를 사용하거나 BCM_SETTEXTMARGIN 메시지를 명시적으로 보낼 수 있습니다.

CanChange 글꼴을 변경할 수 있는지 여부를 결정합니다.
CanChange 단락 서식을 변경할 수 있는지 여부를 결정합니다.
CanChange 이 행을 변경할 수 있는지 여부를 결정합니다.
CanEdit 지정된 범위를 편집할 수 있는지 여부를 결정합니다.
CanPaste 지정된 형식을 사용하여 데이터 개체를 현재 범위에 붙여넣을 수 있는지 여부를 결정합니다.
Cat2 두 문자열을 연결합니다.
CatTop2 컬렉션의 상위 두 문자열 사이에 텍스트를 삽입합니다.
ChangeCase Type 매개 변수에 따라 이 범위의 문자 대/소문자를 변경합니다.
CheckDlgButton 단추 컨트롤의 확인 상태를 변경합니다.
CheckRadioButton 그룹의 지정된 라디오 단추에 확인 표시를 추가(확인)하고 그룹의 다른 모든 라디오 단추에서 확인 표시를 제거합니다(지우기).
CheckTextLimit 추가할 문자 수가 최대 텍스트 제한을 초과하는지 여부를 확인합니다.
ClearAllTabs 모든 탭을 지우고 기본 탭 간격을 사용하여 간격이 같은 탭으로 되돌립니다.

복제 기존 이미지 목록을 복제합니다.
CloseThemeData 테마 데이터 핸들을 닫습니다.
축소 지정된 텍스트 범위를 범위의 시작 또는 끝에 있는 퇴화 지점으로 축소합니다.
ComboBox_AddItemData 지정된 위치에 있는 콤보 상자의 목록에 항목 데이터를 추가합니다. 이 매크로를 사용하거나 CB_ADDSTRING 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_AddString 콤보 상자의 목록에 문자열을 추가합니다.
ComboBox_DeleteString 콤보 상자의 목록에서 지정된 위치에 있는 항목을 삭제합니다. 이 매크로를 사용하거나 CB_DELETESTRING 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_Dir 콤보 상자에 표시된 목록에 이름을 추가합니다.
ComboBox_Enable 콤보 상자 컨트롤을 사용하거나 사용하지 않도록 설정합니다.
ComboBox_FindItemData 지정된 항목 데이터가 있는 콤보 상자 목록에서 첫 번째 항목을 찾습니다. 이 매크로를 사용하거나 CB_FINDSTRING 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_FindString 지정된 문자열로 시작하는 콤보 상자 목록에서 첫 번째 문자열을 찾습니다. 이 매크로를 사용하거나 CB_FINDSTRING 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_FindStringExact 대/소문자를 구분하지 않는다는 점을 제외하고 지정된 문자열과 정확히 일치하는 콤보 상자 목록의 첫 번째 문자열을 찾습니다. 이 매크로를 사용하거나 CB_FINDSTRINGEXACT 메시지를 명시적으로 보낼 수 있습니다.

ComboBox_GetCount 콤보 상자의 목록 상자에 있는 항목 수를 가져옵니다. 이 매크로를 사용하거나 CB_GETCOUNT 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_GetCueBannerText 콤보 상자의 편집 컨트롤에 표시되는 큐 배너 텍스트를 가져옵니다. 이 매크로를 사용하거나 CB_GETCUEBANNER 메시지를 명시적으로 보냅니다.
ComboBox_GetCurSel 콤보 상자에서 현재 선택된 항목의 인덱스를 가져옵니다. 이 매크로를 사용하거나 CB_GETCURSEL 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_GetDroppedControlRect 드롭다운 상태에서 콤보 상자의 화면 좌표를 검색합니다. 이 매크로를 사용하거나 CB_GETDROPPEDCONTROLRECT 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_GetDroppedState 콤보 상자 컨트롤의 드롭 목록이 표시되는지 여부를 확인합니다. 이 매크로를 사용하거나 CB_GETDROPPEDSTATE 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_GetExtendedUI 콤보 상자가 기본 UI(사용자 인터페이스)를 사용하는지 또는 확장 UI를 사용하는지 확인합니다. 이 매크로를 사용하거나 CB_GETEXTENDEDUI 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_GetItemData 콤보 상자의 지정된 목록 항목과 연결된 애플리케이션 정의 값을 가져옵니다. 이 매크로를 사용하거나 CB_GETITEMDATA 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_GetItemHeight 콤보 상자에서 목록 항목의 높이를 검색합니다. 이 매크로를 사용하거나 CB_GETITEMHEIGHT 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_GetLBText 콤보 상자의 목록에서 문자열을 가져옵니다. 이 매크로를 사용하거나 CB_GETLBTEXT 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_GetLBTextLen 콤보 상자의 목록에서 문자열의 길이를 가져옵니다. 이 매크로를 사용하거나 CB_GETLBTEXTLEN 메시지를 명시적으로 보낼 수 있습니다.

ComboBox_GetMinVisible 콤보 상자의 드롭다운 목록에 표시되는 항목의 최소 수를 가져옵니다.
ComboBox_GetText 콤보 상자 컨트롤에서 텍스트를 검색합니다.
ComboBox_GetTextLength 콤보 상자의 텍스트에서 문자 수를 가져옵니다.
ComboBox_InsertItemData 콤보 상자의 지정된 위치에 목록의 항목 데이터를 삽입합니다. 이 매크로를 사용하거나 CB_INSERTSTRING 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_InsertString 지정된 위치에 있는 콤보 상자의 목록에 문자열을 추가합니다. 이 매크로를 사용하거나 CB_INSERTSTRING 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_LimitText 사용자가 콤보 상자의 편집 컨트롤에 입력할 수 있는 텍스트의 길이를 제한합니다. 이 매크로를 사용하거나 CB_LIMITTEXT 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_ResetContent 콤보 상자의 목록 상자 및 편집 컨트롤에서 모든 항목을 제거합니다. 이 매크로를 사용하거나 CB_RESETCONTENT 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_SelectItemData 지정된 항목 데이터가 있는 항목에 대한 콤보 상자의 목록을 검색합니다. 일치하는 항목이 발견되면 해당 항목이 선택됩니다. 이 매크로를 사용하거나 CB_SELECTSTRING 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_SelectString 지정된 문자열의 문자로 시작하는 항목에 대한 콤보 상자의 목록을 검색합니다. 일치하는 항목이 발견되면 해당 항목이 선택됩니다. 이 매크로를 사용하거나 CB_SELECTSTRING 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_SetCueBannerText 콤보 상자의 편집 컨트롤에 대해 표시되는 큐 배너 텍스트를 설정합니다.

ComboBox_SetCurSel 콤보 상자에서 현재 선택된 항목을 설정합니다. 이 매크로를 사용하거나 CB_SETCURSEL 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_SetExtendedUI 기본 UI(사용자 인터페이스) 또는 CBS_DROPDOWN 또는 CBS_DROPDOWNLIST 스타일이 있는 콤보 상자의 확장 UI를 선택합니다. 이 매크로를 사용하거나 CB_SETEXTENDEDUI 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_SetItemData 콤보 상자에서 지정된 목록 항목과 연결된 애플리케이션 정의 값을 설정합니다. 이 매크로를 사용하거나 CB_SETITEMDATA 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_SetItemHeight 콤보 상자에서 목록 항목 또는 선택 필드의 높이를 설정합니다. 이 매크로를 사용하거나 CB_SETITEMHEIGHT 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_SetMinVisible 콤보 상자의 드롭다운 목록에 표시되는 항목의 최소 수를 설정합니다.
ComboBox_SetText 콤보 상자의 텍스트를 설정합니다.
ComboBox_ShowDropdown 콤보 상자에서 목록을 표시하거나 숨깁니다. 이 매크로를 사용하거나 CB_SHOWDROPDOWN 메시지를 명시적으로 보낼 수 있습니다.
ContextSensitiveHelp 서식 있는 편집 컨트롤이 컨텍스트에 민감한 도움말 모드로 전환되거나 외부로 전환되어야 하는지 여부를 나타냅니다. 리치 편집 컨트롤은 상태 변경이 발생하는 경우 현재 활성 상태 개체의 IRichEditOle::ContextSensitiveHelp 메서드를 호출합니다.
ContextSensitiveHelp 애플리케이션이 컨텍스트에 민감한 도움말 모드로 전환할지 아니면 외부로 전환해야 하는지 여부를 나타냅니다. 이 메서드는 IOleWindow::ContextSensitiveHelp에 대해 설명된 기능을 구현해야 합니다.

ConvertObject 개체를 새 형식으로 변환합니다. 이 호출은 개체를 다시 로드하지만 강제로 업데이트하지는 않습니다. 호출자는 이 작업을 수행해야 합니다.
Copy 지정된 이미지 목록에서 이미지를 복사합니다.
Copy 텍스트를 데이터 개체에 복사합니다.
CreateMappedBitmap 도구 모음에서 사용할 비트맵을 만듭니다.
CreatePropertySheetPageA 속성 시트에 대한 새 페이지를 만듭니다. (ANSI)
CreatePropertySheetPageW 속성 시트에 대한 새 페이지를 만듭니다. (유니코드)
CreateProvider 창 없는 서식 있는 편집 컨트롤의 부모에 대한 Microsoft UI 자동화 공급자 개체를 가져옵니다.
CreateStatusWindowA 일반적으로 애플리케이션의 상태를 표시하는 데 사용되는 상태 창을 만듭니다. (ANSI)
CreateStatusWindowW 일반적으로 애플리케이션의 상태를 표시하는 데 사용되는 상태 창을 만듭니다. (유니코드)
CreateSyntheticPointerDevice 호출 애플리케이션에 대한 포인터 삽입 디바이스를 구성하고 앱이 삽입할 수 있는 최대 동시 포인터 수를 초기화합니다.
CreateTextServices CreateTextServices 함수는 텍스트 서비스 개체의 인스턴스를 만듭니다. 텍스트 서비스 개체는 ITextServices 및 TOM(텍스트 개체 모델)을 비롯한 다양한 인터페이스를 지원합니다.

CreateToolbarEx 도구 모음 창을 만들고 지정된 단추를 도구 모음에 추가합니다.
CreateUpDownControl 업다운 컨트롤을 만듭니다. 참고:_This 함수는 사용되지 않습니다. 16비트 함수이며 범위와 위치에 대해 32비트 값을 처리할 수 없습니다.
잘라내기 pVar 매개 변수에 따라 일반 또는 서식 있는 텍스트를 데이터 개체 또는 클립보드로 잘라냅니다.
DateTime_CloseMonthCal DTP(날짜 및 시간 선택기) 컨트롤을 닫습니다. 이 매크로를 사용하거나 DTM_CLOSEMONTHCAL 메시지를 명시적으로 보냅니다.
DateTime_GetDateTimePickerInfo 지정된 DTP(날짜 및 시간 선택기) 컨트롤에 대한 정보를 가져옵니다.
DateTime_GetIdealSize 클리핑 없이 컨트롤을 표시하는 데 필요한 크기를 가져옵니다. 이 매크로를 사용하거나 DTM_GETIDEALSIZE 메시지를 명시적으로 보냅니다.
DateTime_GetMonthCal DTP(날짜 및 시간 선택기) 자식 월 달력 컨트롤에 대한 핸들을 가져옵니다. 이 매크로를 사용하거나 DTM_GETMONTHCAL 메시지를 명시적으로 보낼 수 있습니다.
DateTime_GetMonthCalColor DTP(날짜 및 시간 선택기) 컨트롤 내에서 월 달력의 지정된 부분에 대한 색을 가져옵니다. 이 매크로를 사용하거나 DTM_GETMCCOLOR 메시지를 명시적으로 보낼 수 있습니다.
DateTime_GetMonthCalFont DTP(날짜 및 시간 선택기) 컨트롤의 자식 월 달력 컨트롤이 현재 사용하는 글꼴을 가져옵니다. 이 매크로를 사용하거나 DTM_GETMCFONT 메시지를 명시적으로 보낼 수 있습니다.
DateTime_GetMonthCalStyle 지정된 DTP(날짜 및 시간 선택기) 컨트롤의 스타일을 가져옵니다. 이 매크로를 사용하거나 DTM_GETMCSTYLE 메시지를 명시적으로 보냅니다.

DateTime_GetRange DTP(날짜 및 시간 선택기) 컨트롤에 허용되는 현재 최소 및 최대 시스템 시간을 가져옵니다. 이 매크로를 사용하거나 DTM_GETRANGE 메시지를 명시적으로 보낼 수 있습니다.
DateTime_GetSystemtime DTP(날짜 및 시간 선택기) 컨트롤에서 현재 선택된 시간을 가져와 지정된 SYSTEMTIME 구조체에 배치합니다. 이 매크로를 사용하거나 DTM_GETSYSTEMTIME 메시지를 명시적으로 보낼 수 있습니다.
DateTime_SetFormat 지정된 형식 문자열을 기반으로 DTP(날짜 및 시간 선택기) 컨트롤의 표시를 설정합니다. 이 매크로를 사용하거나 DTM_SETFORMAT 메시지를 명시적으로 보낼 수 있습니다.
DateTime_SetMonthCalColor DTP(날짜 및 시간 선택기) 컨트롤 내에서 월 달력의 지정된 부분에 대한 색을 설정합니다. 이 매크로를 사용하거나 DTM_SETMCCOLOR 메시지를 명시적으로 보낼 수 있습니다.
DateTime_SetMonthCalFont DTP(날짜 및 시간 선택기) 컨트롤의 자식 월 달력 컨트롤에서 사용할 글꼴을 설정합니다. 이 매크로를 사용하거나 DTM_SETMCFONT 메시지를 명시적으로 보낼 수 있습니다.
DateTime_SetMonthCalStyle 지정된 DTP(날짜 및 시간 선택기) 컨트롤의 스타일을 설정합니다. 이 매크로를 사용하거나 DTM_SETMCSTYLE 메시지를 명시적으로 보냅니다.
DateTime_SetRange DTP(날짜 및 시간 선택기) 컨트롤에 허용되는 최소 및 최대 시스템 시간을 설정합니다. 이 매크로를 사용하거나 DTM_SETRANGE 메시지를 명시적으로 보낼 수 있습니다.
DateTime_SetSystemtime DTP(날짜 및 시간 선택기) 컨트롤을 지정된 날짜 및 시간으로 설정합니다. 이 매크로를 사용하거나 DTM_SETSYSTEMTIME 메시지를 명시적으로 보낼 수 있습니다.
삭제 CTRL 키를 누른 상태로 두지 않고 DELETE 및 BACKSPACE 키를 모방합니다.
DeleteObject 서식 있는 편집 컨트롤에서 개체가 삭제될 예정이라는 알림을 보냅니다. 이 멤버가 호출될 때 개체가 해제될 필요는 없습니다.

DeleteRange 지정된 범위의 내용을 삭제합니다.
DeleteSubrange 범위에서 하위 범위를 삭제합니다.
DeleteTab 지정된 변위에서 탭을 삭제합니다.
DestroyPropertySheetPage 속성 시트 페이지를 삭제합니다. 애플리케이션에서 PropertySheet 함수에 전달되지 않은 페이지에 대해 이 함수를 호출해야 합니다.
DestroySyntheticPointerDevice 지정된 포인터 삽입 디바이스를 삭제합니다.
DiscardImages 지정된 대로 목록에서 이미지를 삭제합니다.
DlgDirListA 콤보 상자의 콘텐츠를 지정된 디렉터리의 하위 디렉터리 및 파일 이름으로 바꿉니다. 파일 특성 집합을 지정하여 이름 목록을 필터링할 수 있습니다. 목록에는 필요에 따라 매핑된 드라이브가 포함될 수 있습니다. (ANSI)
DlgDirListComboBoxA 콤보 상자의 콘텐츠를 지정된 디렉터리의 하위 디렉터리 및 파일 이름으로 바꿉니다. 파일 특성 집합을 지정하여 이름 목록을 필터링할 수 있습니다. 이름 목록에는 매핑된 드라이브 문자가 포함될 수 있습니다. (ANSI)
DlgDirListComboBoxW 콤보 상자의 콘텐츠를 지정된 디렉터리의 하위 디렉터리 및 파일 이름으로 바꿉니다. 파일 특성 집합을 지정하여 이름 목록을 필터링할 수 있습니다. 이름 목록에는 매핑된 드라이브 문자가 포함될 수 있습니다. (유니코드)
DlgDirListW 콤보 상자의 콘텐츠를 지정된 디렉터리의 하위 디렉터리 및 파일 이름으로 바꿉니다. 파일 특성 집합을 지정하여 이름 목록을 필터링할 수 있습니다. 목록에는 필요에 따라 매핑된 드라이브가 포함될 수 있습니다. (유니코드)

DlgDirSelectComboBoxExA DlgDirListComboBox 함수를 사용하여 채워진 콤보 상자에서 현재 선택 항목을 검색합니다. 선택 항목은 드라이브 문자, 파일 또는 디렉터리 이름으로 해석됩니다. (ANSI)
DlgDirSelectComboBoxExW DlgDirListComboBox 함수를 사용하여 채워진 콤보 상자에서 현재 선택 항목을 검색합니다. 선택 항목은 드라이브 문자, 파일 또는 디렉터리 이름으로 해석됩니다. (유니코드)
DlgDirSelectExA 단일 선택 목록 상자에서 현재 선택 영역을 검색합니다. 목록 상자가 DlgDirList 함수에 의해 채워졌으며 선택 영역이 드라이브 문자, 파일 이름 또는 디렉터리 이름이라고 가정합니다. (ANSI)
DlgDirSelectExW 단일 선택 목록 상자에서 현재 선택 영역을 검색합니다. 목록 상자가 DlgDirList 함수에 의해 채워졌으며 선택 영역이 드라이브 문자, 파일 이름 또는 디렉터리 이름이라고 가정합니다. (유니코드)
DPA_AppendPtr DPA(동적 포인터 배열) 끝에 새 항목을 삽입합니다.
DPA_Clone DPA(동적 포인터 배열)를 복제합니다.
DPA_Create DPA(동적 포인터 배열)를 만듭니다.
DPA_CreateEx 지정된 크기 및 힙 위치를 사용하여 DPA(동적 포인터 배열)를 만듭니다.
DPA_DeleteAllPtrs DPA(동적 포인터 배열)에서 모든 항목을 제거하고 그에 따라 DPA를 축소합니다.
DPA_DeletePtr DPA(동적 포인터 배열)에서 항목을 제거합니다. 제거된 항목을 수용하기 위해 필요한 경우 DPA가 축소됩니다.
DPA_Destroy DPA(동적 포인터 배열)를 해제합니다.

DPA_DestroyCallback DPA(동적 포인터 배열)의 각 요소에서 pfnCB를 호출한 다음 DPA를 해제합니다.
DPA_EnumCallback DPA(동적 포인터 배열)를 반복하고 각 항목에서 pfnCB를 호출합니다.
DPA_FastDeleteLastPtr DPA(동적 포인터 배열)에서 마지막 포인터를 삭제합니다.
DPA_FastGetPtr DPA(동적 포인터 배열)에서 지정된 포인터의 값을 가져옵니다.
DPA_GetPtr DPA(동적 포인터 배열)에서 항목을 가져옵니다.
DPA_GetPtrCount DPA(동적 포인터 배열)의 포인터 수를 가져옵니다.
DPA_GetPtrIndex DPA(동적 포인터 배열)에 있는 일치하는 항목의 인덱스 를 가져옵니다.
DPA_GetPtrPtr DPA(동적 포인터 배열)의 내부 포인터 배열에 대한 포인터를 가져옵니다.
DPA_GetSize DPA(동적 포인터 배열)의 크기를 가져옵니다.
DPA_Grow DPA(동적 포인터 배열)의 포인터 수를 변경합니다.
DPA_InsertPtr DPA(동적 포인터 배열)의 지정된 위치에 새 항목을 삽입합니다. 필요한 경우 DPA는 새 항목을 수용하도록 확장됩니다.
DPA_LoadStream 지정된 콜백 함수를 호출하여 각 요소를 읽어 스트림에서 DPA(동적 포인터 배열)를 로드합니다.

DPA_Merge 두 DBA(동적 포인터 배열)의 내용을 결합합니다.
DPA_SaveStream 헤더를 작성한 다음 지정된 콜백 함수를 호출하여 각 요소를 작성하여 DPA(동적 포인터 배열)를 스트림에 저장합니다.
DPA_Search DPA(동적 포인터 배열)에서 항목을 찾습니다.
DPA_SetPtr DPA(동적 포인터 배열)의 항목에 값을 할당합니다.
DPA_SetPtrCount DPA(동적 포인터 배열)의 포인터 수를 설정합니다.
DPA_Sort DPA(동적 포인터 배열)의 항목을 정렬합니다.
DPA_SortedInsertPtr 지정된 기존 항목 앞이나 뒤에 새 항목을 삽입합니다.
DragEnter 끌어서 놓기 작업 중에 지정된 창의 업데이트를 잠그고 창 내의 지정된 위치에 끌기 이미지를 표시합니다. (IImageList.DragEnter)
DragLeave 지정된 창의 잠금을 해제하고 끌기 이미지를 숨겨 창의 업데이트를 활성화합니다.
DragMove 끌어서 놓기 작업 중에 끌기 이미지를 이동합니다. 이 함수는 일반적으로 WM_MOUSEMOVE 메시지에 대한 응답으로 호출됩니다. (IImageList.DragMove)
DragShowNolock 끌기 이미지를 표시하거나 숨깁니다. (IImageList.DragShowNolock)

Draw 지정된 디바이스 컨텍스트에서 이미지 목록 항목을 그립니다. (IImageList.Draw)
DrawInsert 지정한 끌기 목록 상자의 부모 창에 삽입 아이콘을 그립니다.
DrawShadowText 그림자가 있는 텍스트를 그립니다.
DrawStatusTextA DrawStatusText 함수는 테두리가 있는 상태 창 스타일로 지정된 텍스트를 그립니다. (ANSI)
DrawStatusTextW DrawStatusText 함수는 테두리가 있는 상태 창 스타일로 지정된 텍스트를 그립니다. (유니코드)
DrawThemeBackground 지정된 컨트롤 파트에 대한 시각적 스타일로 정의된 테두리 및 채우기를 그립니다.
DrawThemeBackgroundEx 지정된 컨트롤 파트에 대한 시각적 스타일로 정의된 배경 이미지를 그립니다.
DrawThemeEdge 직사각형의 시각적 스타일로 정의된 하나 이상의 모서리를 그립니다.
DrawThemeIcon 시각적 스타일로 정의된 아이콘 효과를 사용하여 이미지 목록에서 이미지를 그립니다.
DrawThemeParentBackground 부분적으로 투명하거나 알파가 혼합된 자식 컨트롤로 덮힌 부모 컨트롤의 파트를 그립니다.
DrawThemeParentBackgroundEx 부분적으로 투명하거나 알파가 혼합된 자식 컨트롤에서 부모 컨트롤이 표시되는 파트를 그리는 데 사용됩니다. WM_ERASEBKGND 메시지와 WM_PRINTCLIENT를 보냅니다.
DrawThemeText 시각적 스타일에서 정의한 색과 글꼴을 사용하여 텍스트를 그립니다.

메서드의 시각적 스타일에서 정의한 색과 글꼴을 사용하여 텍스트를 그립니다. 추가 텍스트 형식 옵션을 허용하여 DrawThemeText를 확장합니다.
DSA_AppendItem DSA(동적 구조 배열)의 끝에 새 항목을 추가합니다.
DSA_Clone DSA(동적 구조체 배열)를 복제합니다.
DSA_Create DSA(동적 구조 배열)를 만듭니다.
DSA_DeleteAllItems DSA(동적 구조 배열)에서 모든 항목을 삭제합니다.
DSA_DeleteItem DSA(동적 구조 배열)에서 항목을 삭제합니다.
DSA_Destroy DSA(동적 구조 배열)를 해제합니다.
DSA_DestroyCallback 각 항목에서 지정된 콜백 함수를 호출하여 DSA(동적 구조 배열)를 반복합니다. 배열 끝에 도달하면 DSA가 해제됩니다.
DSA_EnumCallback DSA(동적 구조 배열)를 반복하고 각 항목에서 pfnCB를 호출합니다.
DSA_GetItem DSA(동적 구조 배열)에서 요소를 가져옵니다.
DSA_GetItemCount DSA(동적 구조 배열)의 항목 수를 가져옵니다.
DSA_GetItemPtr DSA(동적 구조 배열)에서 요소에 대한 포인터를 가져옵니다.

DSA_GetSize DSA(동적 구조 배열)의 크기를 가져옵니다.
DSA_InsertItem DSA(동적 구조 배열)에 새 항목을 삽입합니다. 필요한 경우 DSA는 새 항목을 수용하도록 확장됩니다.
DSA_SetItem DSA(동적 구조 배열)에서 요소의 내용을 설정합니다.
DSA_Sort DSA(동적 구조 배열)의 항목을 정렬합니다.
Edit_CanUndo 편집 또는 서식 있는 편집 컨트롤의 실행 취소 큐에 작업이 있는지 여부를 결정합니다. 이 매크로를 사용하거나 EM_CANUNDO 메시지를 명시적으로 보낼 수 있습니다.
Edit_EmptyUndoBuffer 편집 또는 서식 있는 편집 컨트롤의 실행 취소 플래그를 다시 설정합니다. 편집 컨트롤 내의 작업을 실행 취소할 수 있을 때마다 실행 취소 플래그가 설정됩니다. 이 매크로를 사용하거나 EM_EMPTYUNDOBUFFER 메시지를 명시적으로 보낼 수 있습니다.
Edit_Enable 편집 컨트롤을 사용하거나 사용하지 않도록 설정합니다.
Edit_EnableSearchWeb "Bing을 사용하여 검색..."을 사용하거나 사용하지 않도록 설정합니다. 편집 컨트롤의 상황에 맞는 메뉴 항목입니다. 이 매크로를 사용하거나 EM_ENABLESEARCHWEB 메시지를 명시적으로 보낼 수 있습니다.
Edit_FmtLines 여러 줄 편집 컨트롤에서 검색된 텍스트에 소프트 줄 바꿈 문자가 포함되는지 여부를 결정하는 플래그를 설정합니다.
Edit_GetCaretIndex 지정된 편집 컨트롤에 대한 caret 위치의 문자 인덱스 를 가져옵니다. 이 매크로를 사용하거나 EM_GETCARETINDEX 메시지를 명시적으로 보낼 수 있습니다.

Edit_GetCueBannerText 편집 컨트롤에서 텍스트 큐 또는 팁으로 표시되는 텍스트를 가져옵니다. 이 매크로를 사용하거나 EM_GETCUEBANNER 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetEndOfLine 편집 컨트롤의 내용에 사용되는 줄 문자의 끝을 가져옵니다. 이 매크로를 사용하거나 EM_GETENDOFLINE 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetExtendedStyle 지정된 편집 컨트롤에 현재 사용 중인 확장 스타일을 가져옵니다. 이 매크로를 사용하거나 EM_GETEXTENDEDSTYLE 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetFileLine 지정된 파일(또는 논리적) 줄의 텍스트를 가져옵니다(텍스트 줄 바꿈 구분 기호는 무시됨). 이 매크로를 사용하거나 EM_GETFILELINE 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetFileLineCount 파일(또는 논리적) 줄 수를 가져옵니다(텍스트 줄 바꿈 구분 기호는 무시됨). 이 매크로를 사용하거나 EM_GETFILELINECOUNT 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetFileLineFromChar 지정된 문자 인덱스가 포함된 파일(또는 논리적) 텍스트 줄의 인덱스(텍스트 줄 바꿈 구분 기호는 무시됨)를 가져옵니다. 이 매크로를 사용하거나 EM_FILELINEFROMCHAR 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetFileLineIndex 지정된 표시줄에 따라 파일(또는 논리적) 텍스트 줄의 인덱스 를 가져옵니다. 이 매크로를 사용하거나 EM_FILELINEINDEX 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetFileLineLength 지정된 문자 인덱스에서 파일(또는 논리적) 텍스트 줄의 길이를 가져옵니다(텍스트 줄 바꿈 구분 기호는 무시됨). 이 매크로를 사용하거나 EM_FILELINELENGTH 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetFirstVisibleLine 여러 줄 편집 또는 서식 있는 편집 컨트롤에서 맨 위에 표시되는 줄의 인덱스를 가져옵니다. 이 매크로를 사용하거나 EM_GETFIRSTVISIBLELINE 메시지를 명시적으로 보낼 수 있습니다.

Edit_GetHandle 여러 줄 편집 컨트롤의 텍스트에 대해 현재 할당된 메모리에 대한 핸들을 가져옵니다. 이 매크로를 사용하거나 EM_GETHANDLE 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetHilite 이 매크로는 구현되지 않습니다. (Edit_GetHilite)
Edit_GetLine 편집 또는 서식 있는 편집 컨트롤에서 텍스트 줄을 검색합니다. 이 매크로를 사용하거나 EM_GETLINE 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetLineCount 편집 컨트롤의 텍스트에 있는 줄 수를 가져옵니다. 이 매크로를 사용하거나 EM_GETLINECOUNT 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetModify 편집 또는 서식 있는 편집 컨트롤의 수정 플래그 상태를 가져옵니다. 플래그는 컨트롤의 내용이 수정되었는지 여부를 나타냅니다. 이 매크로를 사용하거나 EM_GETMODIFY 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetPasswordChar 편집 또는 서식 있는 편집 컨트롤의 암호 문자를 가져옵니다. 이 매크로를 사용하거나 EM_GETPASSWORDCHAR 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetRect 편집 컨트롤의 서식 사각형을 가져옵니다. 이 매크로를 사용하거나 EM_GETRECT 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetSel 편집 또는 서식 있는 편집 컨트롤에서 현재 선택 항목의 시작 및 끝 문자 위치를 가져옵니다. 이 매크로를 사용하거나 EM_GETSEL 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetText 편집 컨트롤의 텍스트를 가져옵니다.
Edit_GetTextLength 편집 컨트롤의 텍스트에 있는 문자 수를 가져옵니다.

Edit_GetWordBreakProc 편집 또는 서식 있는 편집 컨트롤의 Wordwrap 함수의 주소를 검색합니다. 이 매크로를 사용하거나 EM_GETWORDBREAKPROC 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetZoom 편집 컨트롤의 현재 확대/축소 비율을 가져옵니다(확대/축소 비율은 항상 1/64에서 64 사이임). 이 매크로를 사용하거나 EM_GETZOOM 메시지를 명시적으로 보낼 수 있습니다.
Edit_HideBalloonTip 편집 컨트롤과 연결된 풍선 팁을 숨깁니다. 이 매크로를 사용하거나 EM_HIDEBALLOONTIP 메시지를 명시적으로 보낼 수 있습니다.
Edit_LimitText 편집 컨트롤에 입력할 수 있는 텍스트의 길이를 제한합니다. 이 매크로를 사용하거나 EM_LIMITTEXT 메시지를 명시적으로 보낼 수 있습니다.
Edit_LineFromChar 여러 줄 편집 또는 서식 있는 편집 컨트롤에서 지정된 문자 인덱스가 포함된 줄의 인덱스를 가져옵니다. 이 매크로를 사용하거나 EM_LINEFROMCHAR 메시지를 명시적으로 보낼 수 있습니다.
Edit_LineIndex 여러 줄 편집 또는 서식 있는 편집 컨트롤에서 지정된 줄 첫 번째 문자의 문자 인덱스를 가져옵니다. 이 매크로를 사용하거나 EM_LINEINDEX 메시지를 명시적으로 보낼 수 있습니다.
Edit_LineLength 편집 또는 서식 있는 편집 컨트롤에서 줄의 길이(문자)를 검색합니다. 이 매크로를 사용하거나 EM_LINELENGTH 메시지를 명시적으로 보낼 수 있습니다.
Edit_NoSetFocus 한 줄 편집 컨트롤이 키보드 포커스를 받지 않도록 합니다. 이 매크로를 사용하거나 EM_NOSETFOCUS 메시지를 명시적으로 보낼 수 있습니다.
Edit_ReplaceSel 편집 컨트롤 또는 서식 있는 편집 컨트롤에서 선택한 텍스트를 지정된 텍스트로 바꿉니다. 이 매크로를 사용하거나 EM_REPLACESEL 메시지를 명시적으로 보낼 수 있습니다.
Edit_Scroll 여러 줄 편집 또는 서식 있는 편집 컨트롤에서 텍스트를 세로로 스크롤합니다. 이 매크로를 사용하거나 EM_SCROLL 메시지를 명시적으로 보낼 수 있습니다.

Edit_ScrollCaret 편집 또는 서식 있는 편집 컨트롤에서 캐럿을 뷰로 스크롤합니다. 이 매크로를 사용하거나 EM_SCROLLCARET 메시지를 명시적으로 보낼 수 있습니다.
Edit_SearchWeb "Bing을 사용하여 검색..."을 호출합니다. 편집 컨트롤의 상황에 맞는 메뉴 항목입니다. 이 매크로를 사용하거나 EM_SEARCHWEB 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetCaretIndex caret를 찾을 문자 인덱스 를 설정합니다. 이 매크로를 사용하거나 EM_SETCARETINDEX 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetCueBannerText 편집 컨트롤에 대해 텍스트 큐 또는 팁으로 표시되는 텍스트를 설정합니다. 이 매크로를 사용하거나 EM_SETCUEBANNER 메시지를 명시적으로 보낼 수 있습니다. (Edit_SetCueBannerText)
Edit_SetCueBannerTextFocused 편집 컨트롤에 대해 텍스트 큐 또는 팁으로 표시되는 텍스트를 설정합니다. 이 매크로를 사용하거나 EM_SETCUEBANNER 메시지를 명시적으로 보낼 수 있습니다. (Edit_SetCueBannerTextFocused)
Edit_SetEndOfLine 편집 컨트롤의 내용에 사용되는 줄 문자의 끝을 설정합니다. 이 매크로를 사용하거나 EM_SETENDOFLINE 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetExtendedStyle 스타일 마스크를 사용하여 편집 컨트롤의 확장 스타일을 설정합니다. 이 매크로를 사용하거나 EM_SETEXTENDEDSTYLE 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetHandle 여러 줄 편집 컨트롤에서 사용할 메모리의 핸들을 설정합니다. 이 매크로를 사용하거나 EM_SETHANDLE 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetHilite 이 매크로는 구현되지 않습니다. (Edit_SetHilite)
Edit_SetModify 편집 컨트롤에 대한 수정 플래그를 설정하거나 지웁니다. 수정 플래그는 편집 컨트롤 내의 텍스트가 수정되었는지 여부를 나타냅니다. 이 매크로를 사용하거나 EM_SETMODIFY 메시지를 명시적으로 보낼 수 있습니다.

Edit_SetPasswordChar 편집 또는 서식 있는 편집 컨트롤의 암호 문자를 설정하거나 제거합니다. 암호 문자가 설정되면 사용자가 입력한 문자 대신 해당 문자가 표시됩니다. 이 매크로를 사용하거나 EM_SETPASSWORDCHAR 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetReadOnly 편집 또는 서식 있는 편집 컨트롤의 읽기 전용 스타일(ES_READONLY)을 설정하거나 제거합니다. 이 매크로를 사용하거나 EM_SETREADONLY 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetRect 편집 컨트롤의 서식 사각형을 설정합니다. 이 매크로를 사용하거나 EM_SETRECT 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetRectNoPaint 여러 줄 편집 컨트롤의 서식 사각형을 설정합니다. 이 매크로는 편집 컨트롤 창을 다시 그리지 않는다는 점을 제외하고 Edit_SetRect와 동일합니다. 이 매크로를 사용하거나 EM_SETRECTNP 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetSel 편집 또는 서식 있는 편집 컨트롤에서 문자 범위를 선택합니다. 이 매크로를 사용하거나 EM_SETSEL 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetTabStops 여러 줄 편집 또는 서식 있는 편집 컨트롤에서 탭 정지를 설정합니다. 텍스트가 컨트롤에 복사되면 텍스트의 모든 탭 문자로 인해 다음 탭 정지까지 공간이 생성됩니다. 이 매크로를 사용하거나 EM_SETTABSTOPS 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetText 편집 컨트롤의 텍스트를 설정합니다.
Edit_SetWordBreakProc 편집 컨트롤의 기본 단어 줄 바꿈 함수를 애플리케이션 정의 단어 줄 바꿈 함수로 바꿉니다. 이 매크로를 사용하거나 EM_SETWORDBREAKPROC 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetZoom 편집 컨트롤의 현재 확대/축소 비율을 설정합니다(확대/축소 비율은 항상 1/64에서 64 사이임). 이 매크로를 사용하거나 EM_SETZOOM 메시지를 명시적으로 보낼 수 있습니다.

Edit_ShowBalloonTip 편집 컨트롤과 연결된 풍선 팁을 표시합니다. 이 매크로를 사용하거나 EM_SHOWBALLOONTIP 메시지를 명시적으로 보낼 수 있습니다.
Edit_TakeFocus 한 줄 편집 컨트롤을 강제 적용하여 키보드 포커스를 받습니다. 이 매크로를 사용하거나 EM_TAKEFOCUS 메시지를 명시적으로 보낼 수 있습니다.
Edit_Undo 편집 또는 서식 있는 편집 컨트롤의 실행 취소 큐에서 마지막 작업을 실행 취소합니다. 이 매크로를 사용하거나 EM_UNDO 메시지를 명시적으로 보낼 수 있습니다.
EDITSTREAMCALLBACK EditStreamCallback 함수는 EM_STREAMIN 및 EM_STREAMOUT 메시지와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
EDITWORDBREAKPROCA EM_SETWORDBREAKPROC 메시지와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (ANSI)
EDITWORDBREAKPROCEX EditWordBreakProcEx 함수는 EM_SETWORDBREAKPROCEX 메시지와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
EDITWORDBREAKPROCW EM_SETWORDBREAKPROC 메시지와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (유니코드)
EnableScrollBar EnableScrollBar 함수는 하나 또는 두 스크롤 막대 화살표를 사용하거나 사용하지 않도록 설정합니다.
EnableThemeDialogTexture 대화 상자 창 배경의 시각적 스타일을 사용하거나 사용하지 않도록 설정합니다.
EnableTheming Windows Vista에서 Windows 7:_Enables 또는 현재 및 이후 세션의 현재 사용자에 대한 시각적 스타일을 사용하지 않도록 설정합니다. Windows 8 이상:_This 함수는 아무 것도 수행하지 않습니다. 시각적 스타일은 항상 Windows 8 이상에서 사용하도록 설정됩니다.

EncodeFunction 인수 문자열 집합이 지정된 경우 개체를 인코딩합니다.
EndBufferedAnimation 버퍼링된 애니메이션 작업의 첫 번째 프레임을 렌더링하고 애니메이션 타이머를 시작합니다.
EndBufferedPaint 버퍼링된 페인트 작업을 완료하고 연결된 버퍼링된 페인트 핸들을 해제합니다.
EndDrag 끌기 작업을 종료합니다. (IImageList.EndDrag)
EndEditCollection 컬렉션 편집(실행 취소 그룹화라고도 함)을 해제합니다.
EndKey End 키의 기능을 모방합니다.
EndOf 이 범위의 끝을 범위의 마지막 겹치는 단위 끝으로 이동합니다.
EndPanningFeedback 진행 중이거나 BeginPanningFeedback 및 UpdatePanningFeedback에 의해 설정된 기존 애니메이션을 종료합니다.
EvaluateProximityToPolygon 다각형의 점수를 가능한 터치 대상(터치 접촉 영역을 교차하는 다른 모든 다각형과 비교) 및 다각형 내의 조정된 터치 지점으로 반환합니다.
EvaluateProximityToRect 터치 접촉 영역을 교차하는 다른 모든 사각형과 사각형 내의 조정된 터치 지점에 비해 가능한 터치 대상으로 사각형의 점수를 반환합니다.
확장 포함된 부분 단위가 완전히 포함되도록 이 범위를 확장합니다.

찾기 원본 범위에서 지정한 대로 텍스트의 수학 인라인 함수를 검색합니다.
Findtext bstr에서 지정한 텍스트에 대해 최대 개수 문자를 검색합니다. 시작 위치와 방향도 Count로 지정되며 일치 조건은 Flags에 의해 지정됩니다.
FindTextEnd 범위의 End cp부터 시작하여 문자열 bstr에 대한 최대 개수 문자를 검색합니다.
FindTextStart 범위의 시작 cp(cpFirst)에서 시작하여 문자열 bstr에 대한 최대 개수 문자를 검색합니다.
FIRST_IPADDRESS IPM_GETADDRESS 메시지와 함께 검색된 압축된 IP 주소에서 필드 0 값을 추출합니다.
FlatSB_EnableScrollBar 플랫 스크롤 막대 방향 단추 중 하나 또는 둘 모두를 사용하거나 사용하지 않도록 설정합니다. 창에 대해 플랫 스크롤 막대가 초기화되지 않은 경우 이 함수는 표준 EnableScrollBar 함수를 호출합니다.
FlatSB_GetScrollInfo 플랫 스크롤 막대에 대한 정보를 가져옵니다. 창에 대해 플랫 스크롤 막대가 초기화되지 않은 경우 이 함수는 표준 GetScrollInfo 함수를 호출합니다.
FlatSB_GetScrollPos 플랫 스크롤 막대에서 엄지 위치를 가져옵니다. 창에 대해 플랫 스크롤 막대가 초기화되지 않은 경우 이 함수는 표준 GetScrollPos 함수를 호출합니다.
FlatSB_GetScrollProp 플랫 스크롤 막대의 속성을 가져옵니다. 이 함수는 이 창에 대해 InitializeFlatSB가 호출되었는지 확인하는 데에도 사용할 수 있습니다.
FlatSB_GetScrollPropPtr 플랫 스크롤 막대의 속성을 가져옵니다.
FlatSB_GetScrollRange 플랫 스크롤 막대의 스크롤 범위를 가져옵니다. 창에 대해 플랫 스크롤 막대가 초기화되지 않은 경우 이 함수는 표준 GetScrollRange 함수를 호출합니다.

FlatSB_SetScrollInfo 플랫 스크롤 막대에 대한 정보를 설정합니다. 창에 대해 플랫 스크롤 막대가 초기화되지 않은 경우 이 함수는 표준 SetScrollInfo 함수를 호출합니다.
FlatSB_SetScrollPos 플랫 스크롤 막대에서 엄지의 현재 위치를 설정합니다. 창에 대해 플랫 스크롤 막대가 초기화되지 않은 경우 이 함수는 표준 SetScrollPos 함수를 호출합니다.
FlatSB_SetScrollProp 플랫 스크롤 막대의 속성을 설정합니다.
FlatSB_SetScrollRange 플랫 스크롤 막대의 스크롤 범위를 설정합니다. 창에 대해 플랫 스크롤 막대가 초기화되지 않은 경우 이 함수는 표준 SetScrollRange 함수를 호출합니다.
FlatSB_ShowScrollBar 플랫 스크롤 막대를 표시하거나 숨깁니다. 창에 대해 플랫 스크롤 막대가 초기화되지 않은 경우 이 함수는 표준 ShowScrollBar 함수를 호출합니다.
ForceImagePresent 지정된 대로 이미지를 강제로 표시합니다.
FORWARD_WM_NOTIFY WM_NOTIFY 메시지를 보내거나 게시합니다.
FOURTH_IPADDRESS IPM_GETADDRESS 메시지와 함께 검색된 압축된 IP 주소에서 필드 3 값을 추출합니다.
중지 동결 횟수를 증분합니다.
GetActive 스토리의 활성 상태를 설정합니다. (ITextStory.GetActive)
GetActiveStory 활성 스토리를 가져옵니다. 즉, 키보드 및 마우스 입력을 수신하는 스토리입니다.

GetAlignment 현재 단락 맞춤 값을 검색합니다.
GetAlignment 행의 가로 맞춤을 가져옵니다.
GetAllCaps 문자가 모두 대문자인지 여부를 가져옵니다.
GetAnimation 애니메이션 형식을 가져옵니다.
GetAutoLigatures 자동 합자 지원이 활성 상태인지 여부를 가져옵니다.
GetAutospaceAlpha 동아시아 "autospace 사전순" 상태를 가져옵니다.
GetAutospaceNumeric 동아시아 "autospace numeric" 상태를 가져옵니다.
GetAutospaceParens 동아시아 "autospace 괄호" 상태를 가져옵니다.
GetBackColor 텍스트 배경(강조 표시) 색을 가져옵니다.
GetBkColor 이미지 목록의 현재 배경색을 가져옵니다.
GetBold 문자가 굵게 표시되는지 여부를 가져옵니다.
GetBorders 테두리 컬렉션을 가져옵니다.

GetBoundaryRectangle 창 없는 서식 있는 편집 컨트롤의 경계 사각형을 검색합니다.
GetBufferedPaintBits 버퍼가 DIB(디바이스 독립적 비트맵)인 경우 버퍼 비트맵에 대한 포인터를 검색합니다.
GetBufferedPaintDC 그림판 디바이스 컨텍스트(DC)를 가져옵니다. 이는 BeginBufferedPaint에서 검색한 것과 동일한 값입니다.
GetBufferedPaintTargetDC DC(대상 디바이스 컨텍스트)를 검색합니다.
GetBufferedPaintTargetRect BeginBufferedPaint에서 지정한 대상 사각형을 검색합니다.
GetCallback 이미지 목록 콜백 개체를 가져옵니다.
GetCallManager 호출 관리자를 가져옵니다.
GetCaretType caret 형식을 가져옵니다.
GetCch 범위의 문자 수를 가져옵니다.
GetCch 선택한 문자열 인덱스의 문자 수를 가져옵니다.
GetCellAlignment 활성 셀의 세로 맞춤을 가져옵니다.
GetCellBorderColors 활성 셀의 테두리 색을 가져옵니다.

GetCellBorderWidths 현재 셀의 테두리 너비를 가져옵니다.
GetCellColorBack 활성 셀의 배경색을 가져옵니다.
GetCellColorFore 활성 셀의 전경색을 가져옵니다.
GetCellCount 이 행의 셀 수를 가져옵니다.
GetCellCountCache 이 행에 대해 캐시된 셀 수를 가져옵니다.
GetCellIndex 매개 변수를 가져오거나 설정할 활성 셀의 인덱스 를 가져옵니다.
GetCellMargin 이 행의 셀 여백을 가져옵니다.
GetCellMergeFlags 활성 셀의 병합 플래그를 가져옵니다.
GetCells 현재 선택한 표 행 또는 열에 있는 셀의 매개 변수가 있는 셀 개체를 가져옵니다.
GetCellShading 활성 셀의 음영을 가져옵니다.
GetCellVerticalText 활성 셀의 세로 텍스트 설정을 가져옵니다.
GetCellWidth 활성 셀의 너비를 가져옵니다.

GetChar 범위의 시작 위치에 있는 문자를 가져옵니다.
GetChar2 이 범위의 끝에서 지정된 오프셋의 문자를 가져옵니다.
GetCharRep 문자 레퍼토리(쓰기 시스템)를 가져옵니다.
GetCIMSSM 입력 메시지의 원본을 검색합니다(GetCurrentInputMessageSourceInSendMessage).
GetClientRect 서식 있는 편집 컨트롤의 클라이언트 사각형을 검색합니다.
GetClientSite 새 개체를 만들 때 사용할 IOleClientSite 인터페이스를 검색합니다. 서식 있는 편집 컨트롤에 삽입된 모든 개체는 이 함수에서 반환된 클라이언트 사이트 인터페이스를 사용해야 합니다. 클라이언트 사이트는 정확히 하나의 개체와 함께 사용할 수 있습니다.
GetClipboardData 편집 컨트롤의 범위에 대한 클립보드 개체를 검색합니다.
GetClipboardData 클라이언트에서 자체 클립보드 개체를 제공할 수 있습니다.
GetColumn 현재 선택한 열의 열 속성을 가져옵니다.
GetComboBoxInfo 지정된 콤보 상자에 대한 정보를 검색합니다.
GetCompressionMode 동아시아 압축 모드를 가져옵니다.
GetContextMenu 마우스 오른쪽 단추 클릭 이벤트에서 사용할 상황에 맞는 메뉴를 애플리케이션에 쿼리합니다.

GetCookie 클라이언트 쿠키를 가져옵니다.
GetCount 이 문자 서식 컬렉션의 추가 속성 수를 가져옵니다.
GetCount 현재 범위의 활성 하위 범위를 포함하여 하위 범위의 수를 가져옵니다.
GetCount 지정된 스토리 컬렉션의 스토리 수를 검색합니다.
GetCount 문자열 컬렉션의 문자열 수를 가져옵니다.
GetCurrentInputMessageSource 입력 메시지의 원본을 검색합니다.
GetCurrentThemeName 현재 시각적 스타일의 이름을 검색하고 선택적으로 색 구성표 이름과 크기 이름을 검색합니다.
GetDefaultTabStop 기본 탭 너비를 가져옵니다.
GetDisplay 스토리에 대한 새 디스플레이를 가져옵니다.
GetDisplays 이 TOM(텍스트 개체 모델) 엔진 instance 대한 표시 컬렉션을 가져옵니다.
GetDocumentFont TOM(텍스트 개체 모델) 엔진의 이 instance 대한 기본 문자 형식 정보를 제공하는 개체를 가져옵니다.
GetDocumentPara TOM(텍스트 개체 모델) 엔진의 이 instance 대한 기본 단락 서식 정보를 제공하는 개체를 가져옵니다.

GetDoubleStrike 문자를 가운데에 이중 가로줄로 표시할지 여부를 가져옵니다.
GetDragDropEffect 클라이언트에서 삭제 작업의 효과를 지정할 수 있습니다.
GetDragImage 끌기 이미지에 사용되는 임시 이미지 목록을 가져옵니다. 또한 이 함수는 현재 끌기 위치와 끌기 위치를 기준으로 끌어온 이미지의 오프셋을 검색합니다.
GetDropCap 이 범위를 포함하는 단락의 드롭 캡 매개 변수를 가져옵니다.
GetDuplicate 이 텍스트 글꼴 개체의 중복을 가져옵니다.
GetDuplicate 지정된 단락 서식 개체의 복제본을 만듭니다. 중복 속성은 ITextPara 개체의 기본 속성입니다.
GetDuplicate 이 범위 개체의 중복을 가져옵니다.
GetDuplicate2 이 문자 형식 개체의 중복을 가져옵니다.
GetDuplicate2 이 텍스트 단락 서식 개체의 중복을 가져옵니다.
GetDuplicate2 범위 개체의 중복을 가져옵니다.
GetEastAsianFlags 동아시아 플래그를 가져옵니다.
GetEffectColor 특수 텍스트 특성에 사용되는 색을 검색합니다.

GetEffectiveClientRect 지정된 모든 컨트롤을 포함하는 클라이언트 영역의 사각형 크기를 계산합니다.
GetEffects 문자 형식 효과를 가져옵니다.
GetEffects 단락 서식 효과를 가져옵니다.
GetEffects2 추가 문자 형식 효과를 가져옵니다.
GetEmbeddedObject 지정된 범위의 시작 부분, 즉 cpFirst에서 포함된 개체에 대한 포인터를 검색합니다. 범위는 삽입 지점이거나 포함된 개체만 선택해야 합니다.
GetEmboss 문자가 엠보싱되는지 여부를 가져옵니다.
GetEnd 범위의 끝 문자 위치를 가져옵니다.
GetEngrave 문자가 각인된 문자로 표시되는지 여부를 가져옵니다.
GetFirstLineIndent 왼쪽 들여쓰기를 기준으로 단락의 첫 번째 줄을 들여쓰는 데 사용되는 양을 검색합니다. 왼쪽 들여쓰기는 첫 번째 줄을 제외한 단락의 모든 줄에 대한 들여쓰기입니다.
GetFlags 텍스트 선택 플래그를 가져옵니다.
GetFont 지정된 범위의 문자 특성이 있는 ITextFont 개체를 가져옵니다.
GetFont2 현재 범위의 문자 특성이 있는 ITextFont2 개체를 가져옵니다.

GetFontAlignment 단락 글꼴 맞춤 상태를 가져옵니다.
GetForeColor 전경색 또는 텍스트 색을 가져옵니다.
GetFormattedText 지정된 범위의 서식이 지정된 텍스트가 있는 ITextRange 개체를 가져옵니다.
GetFormattedText2 현재 범위의 서식이 지정된 텍스트가 있는 ITextRange2 개체를 가져옵니다.
GetGenerator TOM(텍스트 개체 모델) 엔진의 이름을 가져옵니다.
GetGravity 이 범위의 중력을 가져옵니다.
GetHangingPunctuation 단락이 양쪽 맞춤될 때 오른쪽 여백에 문장 부호 기호를 걸지 여부를 가져옵니다.
GetHeight 행의 높이를 가져옵니다.
GetHidden 문자를 숨길지 여부를 가져옵니다.
GetHyphenation 범위에 대해 자동 하이픈을 사용할 수 있는지 여부를 결정합니다.
GetIcon 이미지에서 아이콘을 만들고 이미지 목록의 마스크를 만듭니다.
GetIconSize 이미지 목록의 이미지 차원을 가져옵니다. 이미지 목록의 모든 이미지는 차원이 동일합니다.

GetImageCount 이미지 목록의 이미지 수를 가져옵니다.
GetImageInfo 이미지 정보를 가져옵니다.
GetImageRect 이미지의 경계 사각형을 가져옵니다.
GetImmContext TOM(텍스트 개체 모델) 호스트에서 IMM(입력 메서드 관리자) 입력 컨텍스트를 가져옵니다.
GetIndent 이 행의 들여쓰기를 가져옵니다.
GetIndex 지정된 범위 시작 문자 위치에서 Unit 매개 변수의 스토리 인덱스를 검색합니다.
GetIndex 스토리의 인덱스 가져오기
GetInlineObject 범위 활성 끝에 있는 인라인 개체의 속성을 가져옵니다.
GetInPlaceContext 현재 위치 활성화를 지원하는 데 필요한 애플리케이션 및 문서 수준 인터페이스 및 정보를 제공합니다.
GetItalic 문자가 기울임꼴인지 여부를 가져옵니다.
GetItemFlags 이미지의 플래그를 가져옵니다.
GetKeepTogether 단락 내에서 페이지 나누기가 허용되는지 여부를 결정합니다.

GetKeepTogether 이 행이 여러 페이지에서 끊어질 수 있는지 여부를 가져옵니다.
GetKeepWithNext 범위의 단락 간에 페이지 나누기가 허용되는지 여부를 결정합니다.
GetKeepWithNext 이 행이 뒤에 오는 행과 동일한 페이지에 표시되어야 하는지 여부를 가져옵니다.
GetKerning 커닝이 발생하는 최소 글꼴 크기를 가져옵니다.
GetLanguageID 언어 ID 또는 LCID(언어 코드 식별자)를 가져옵니다.
GetLeftIndent 단락의 첫 번째 줄을 제외한 모든 줄을 들여쓰는 데 사용되는 거리를 검색합니다. 거리는 왼쪽 여백을 기준으로 합니다.
GetLineSpacing 텍스트 범위의 줄 간격 값을 검색합니다.
GetLineSpacingRule 텍스트 범위에 대한 줄 간격 규칙을 검색합니다.
GetLinkCount 링크인 서식 있는 편집 컨트롤의 개체 수를 반환합니다.
GetLinkType 링크 유형을 가져옵니다.
GetListAlignment 글머리 기호 및 번호 매기기 목록에 사용할 정렬 종류를 검색합니다.
GetListBoxInfo 지정된 목록 상자에서 열당 항목 수를 검색합니다.

GetListLevelIndex 단락에 사용되는 목록 수준 인덱스를 검색합니다.
GetListStart 목록 번호 매기기 시퀀스의 시작 값 또는 코드를 검색합니다.
GetListTab 첫 줄 들여쓰기와 첫 번째 줄의 텍스트 사이의 거리인 목록 탭 설정을 검색합니다. 번호 매기기 또는 글머리 기호 텍스트는 첫 줄 들여쓰기 값에서 왼쪽 맞춤, 가운데 맞춤 또는 오른쪽 맞춤입니다.
GetListType 단락에 사용할 번호 매기기 종류를 검색합니다.
GetMainStory 기본 스토리를 가져옵니다.
GetMathFunctionType 지정된 수학 함수 이름과 연결된 수학 함수 형식을 검색합니다.
GetMathProperties 문서의 수학 속성을 가져옵니다.
GetMathZone 수학 영역이 활성 상태인지 여부를 가져옵니다.
GetModWidthPairs "쌍의 너비 감소"가 활성 상태인지 여부를 가져옵니다.
GetModWidthSpace "공백 너비 늘리기"가 활성 상태인지 여부를 가져옵니다.
GetMUILanguage 특정 프로세스에 대한 공용 컨트롤에서 현재 사용 중인 언어를 가져옵니다.
GetName 이 문서의 파일 이름을 가져옵니다. ITextDocument 기본 속성입니다.

GetName 글꼴 이름을 가져옵니다.
GetNestLevel 테이블의 중첩 수준을 가져옵니다.
GetNewStorage 클립보드에서 붙여넣거나 RTF(서식 있는 텍스트 형식) 스트림에서 읽은 새 개체에 대한 스토리지를 제공합니다.
GetNewStory 새 스토리를 가져옵니다.
GetNoLineNumber 단락 번호 매기기를 사용할지 여부를 결정합니다.
GetNotificationMode 알림 모드를 가져옵니다.
GetObject 다양한 편집 컨트롤의 개체에 대한 REOBJECT 구조체에 저장된 정보를 검색합니다.
GetObjectCount 현재 서식 있는 편집 컨트롤에 포함된 개체 수를 반환합니다.
GetOldNumbers 이전 스타일 번호가 활성 상태인지 여부를 가져옵니다.
GetOriginalSize 지정된 이미지의 원래 크기를 가져옵니다.
GetOutline 문자가 윤곽선으로 표시되는지 여부를 가져옵니다.
GetOverlapping 겹치는 텍스트가 활성 상태인지 여부를 가져옵니다.

GetOverlayImage 오버레이 마스크로 사용되는 이미지 목록에서 지정된 이미지를 검색합니다.
GetPageBreakBefore 범위의 각 단락이 새 페이지에서 시작해야 하는지 여부를 결정합니다.
GetPara 지정된 범위의 단락 특성이 있는 ITextPara 개체를 가져옵니다.
GetPara2 범위의 단락 특성이 있는 ITextPara2 개체를 가져옵니다.
GetPoint 텍스트 범위의 시작 또는 끝 문자 위치와 인트라라인 위치의 화면 좌표를 검색합니다.
GetPointerDevice 포인터 디바이스에 대한 정보를 가져옵니다.
GetPointerDeviceCursors 포인터 디바이스와 연결된 커서에 매핑되는 커서 ID를 가져옵니다.
GetPointerDeviceProperties POINTER_DEVICE_INFO 구조에 포함되지 않은 디바이스 속성을 가져옵니다.
GetPointerDeviceRects 포인터 디바이스의 x 및 y 범위(하이메트릭)와 포인터 디바이스가 매핑된 디스플레이의 x 및 y 범위(현재 해상도)를 가져옵니다.
GetPointerDevices 시스템에 연결된 포인터 디바이스에 대한 정보를 가져옵니다.
GetPosition 기준선을 기준으로 문자가 세로로 오프셋되는 양을 가져옵니다.
GetPositionSubSuper 기준선을 기준으로 첨자 또는 위 첨자 위치를 가져옵니다.

GetPreferredFont 특정 문자 레퍼토리 및 문자 위치에 대한 기본 설정 글꼴을 검색합니다.
GetProperty 속성의 값을 검색합니다.
GetProperty 지정한 속성의 값을 가져옵니다. (ITextFont2.GetProperty)
GetProperty 지정한 속성의 값을 가져옵니다. (ITextPara2.GetProperty)
GetProperty 속성의 값을 가져옵니다.
GetProperty 지정한 속성의 값을 가져옵니다. (ITextRow.GetProperty)
GetProperty 지정한 속성의 값을 가져옵니다. (ITextStory.GetProperty)
GetPropertyInfo 지정된 추가 속성의 속성 형식 및 값을 가져옵니다.
GetPropertyOverrideValue 창 없는 서식 있는 편집 컨트롤의 지정된 Microsoft UI 자동화 접근성 속성에 대한 호스트 컨테이너의 재정의 값을 검색합니다.
GetProtected 문자를 수정하려는 시도로부터 문자가 보호되는지 여부를 가져옵니다.
GetRange 스토리의 텍스트 범위 개체를 가져옵니다.
GetRawPointerDeviceData 포인터 디바이스에서 원시 입력 데이터를 가져옵니다.

GetRect 현재 범위에 대해 지정된 형식의 사각형을 검색합니다.
GetRightIndent 단락의 오른쪽 여백 들여쓰기 크기를 검색합니다.
GetRow 현재 선택한 행의 행 속성을 가져옵니다.
GetRTL 이 행의 오른쪽에서 왼쪽 방향이 있는지 여부를 가져옵니다.
GetSaved 파일이 마지막으로 저장된 이후 변경되었는지 여부를 나타내는 값을 가져옵니다.
GetScaling 글꼴 가로 배율 백분율을 가져옵니다.
GetScrollBarInfo GetScrollBarInfo 함수는 지정된 스크롤 막대에 대한 정보를 검색합니다.
GetScrollInfo GetScrollInfo 함수는 최소 및 최대 스크롤 위치, 페이지 크기, 스크롤 상자(엄지)의 위치를 포함하여 스크롤 막대의 매개 변수를 검색합니다.
GetScrollPos GetScrollPos 함수는 지정된 스크롤 막대에서 스크롤 상자(엄지)의 현재 위치를 검색합니다.
GetScrollRange GetScrollRange 함수는 지정된 스크롤 막대의 현재 최소 및 최대 스크롤 상자(엄지) 위치를 검색합니다.
GetSelection 활성 선택 영역을 가져옵니다. (ITextDocument.GetSelection)
GetSelection2 활성 선택 영역을 가져옵니다. (ITextDocument2.GetSelection2)

GetShadow 문자가 그림자 문자로 표시되는지 여부를 가져옵니다.
GetSize 글꼴 크기를 가져옵니다. (ITextFont.GetSize)
GetSmallCaps 문자가 작은 대문자인지 여부를 가져옵니다.
GetSnapToGrid 단락 줄이 전체 문서에 대해 정의할 수 있는 세로 눈금에 맞춰지는지 여부를 가져옵니다.
GetSpaceAfter 단락 아래의 세로 간격을 검색합니다.
GetSpaceBefore 단락 위의 세로 간격을 검색합니다.
GetSpaceExtension 동아시아 공간 확장 값을 가져옵니다.
GetSpacing 문자 사이의 가로 간격을 가져옵니다.
GetStart 범위의 시작 문자 위치를 가져옵니다.
GetStartPara 범위의 시작 문자 위치가 포함된 단락 시작 부분의 문자 위치를 가져옵니다.
GetStatistics 이미지 목록 통계 구조를 가져옵니다.
GetStory 특정 인덱스에 해당하는 스토리를 검색합니다.

GetStoryCount 이 문서의 스토리 수를 가져옵니다.
GetStoryLength 범위의 스토리에 있는 문자 수를 가져옵니다.
GetStoryRanges 문서의 스토리를 열거하는 데 사용되는 스토리 컬렉션 개체를 가져옵니다.
GetStoryRanges2 문서의 스토리 열거를 위한 개체를 가져옵니다.
GetStoryType 범위의 스토리 유형을 가져옵니다.
GetStrikeThrough 문자가 가운데를 통해 가로줄로 표시되는지 여부를 가져옵니다.
GetStrings 서식 있는 텍스트 문자열의 컬렉션을 가져옵니다.
GetStyle 범위에 있는 문자의 문자 스타일 핸들을 가져옵니다.
GetStyle 지정한 범위의 단락에 대한 스타일 핸들을 검색합니다.
GetSubrange 범위에서 하위 범위를 검색합니다.
GetSubscript 문자가 아래 첨자로 표시되는지 여부를 가져옵니다.
GetSuperscript 문자가 위 첨자로 표시되는지 여부를 가져옵니다.

GetTab 지정된 탭에 대한 탭 매개 변수(변위, 맞춤 및 지시선 스타일)를 검색합니다.
GetTabCount 탭 수를 검색합니다.
Gettable 현재 선택한 테이블의 테이블 속성을 가져옵니다.
GetText 이 범위의 일반 텍스트를 가져옵니다. Text 속성은 ITextRange 인터페이스의 기본 속성입니다.
GetText 지정한 변환 플래그에 따라 스토리의 텍스트를 가져옵니다.
GetText2 지정된 변환 플래그에 따라 이 범위의 텍스트를 가져옵니다.
GetThemeAnimationProperty 스토리보드 ID 및 대상 ID를 기반으로 하는 테마 애니메이션 속성을 가져옵니다.
GetThemeAnimationTransform 스토리보드 ID, 대상 ID 및 변환 인덱스 기반의 애니메이션 변환 작업을 가져옵니다.
GetThemeAppProperties 현재 애플리케이션에서 시각적 스타일이 적용되는 방식을 제어하는 속성 플래그를 검색합니다.
GetThemeBackgroundContentRect 시각적 스타일로 정의된 배경에 대한 콘텐츠 영역의 크기를 검색합니다.
GetThemeBackgroundExtent 콘텐츠 영역에 따라 시각적 스타일로 정의된 배경의 크기와 위치를 계산합니다.
GetThemeBackgroundRegion 지정된 사각형으로 둘러싸인 일반 또는 부분적으로 투명한 배경에 대한 영역을 계산합니다.

GetThemeBitmap 특정 테마, 파트, 상태 및 속성과 관련된 비트맵을 검색합니다.
GetThemeBool 테마 데이터의 SysMetrics 섹션에서 BOOL 속성 값을 검색합니다.
GetThemeColor 색 속성 값을 검색합니다.
GetThemeDocumentationProperty 지정된 테마 파일의 설명서 섹션에서 테마 속성 값을 검색합니다.
GetThemeEnumValue 열거된 유형 속성의 값을 검색합니다.
GetThemeFilename 파일 이름 속성 값을 검색합니다.
GetThemeFont 글꼴 속성 값을 검색합니다.
GetThemeInt int 속성 값을 검색합니다.
GetThemeIntList 시각적 스타일에서 int 데이터 목록을 검색합니다.
GetThemeMargins MARGINS 속성 값을 검색합니다.
GetThemeMetric 메트릭 속성의 값을 검색합니다.
GetThemePartSize 시각적 스타일로 정의된 파트의 원래 크기를 계산합니다.

GetThemePosition 위치 속성의 값을 검색합니다.
GetThemePropertyOrigin 속성에 대한 테마 속성 정의의 위치를 검색합니다.
GetThemeRect RECT 속성 값을 검색합니다.
GetThemeStream 지정된 파트, 상태 및 속성에서 시작하여 지정된 테마에 해당하는 데이터 스트림을 검색합니다.
GetThemeString 문자열 속성의 값을 검색합니다.
GetThemeSysBool 시스템 메트릭의 부울 값을 검색합니다.
GetThemeSysColor 시스템 색의 값을 검색합니다.
GetThemeSysColorBrush 시스템 색 브러시를 검색합니다.
GetThemeSysFont 시스템 글꼴의 LOGFONT를 검색합니다.
GetThemeSysInt 시스템 int의 값을 검색합니다.
GetThemeSysSize 테마 데이터에서 시스템 크기 메트릭 값을 검색합니다.
GetThemeSysString 시스템 문자열의 값을 검색합니다.

GetThemeTextExtent 시각적 스타일 글꼴로 렌더링될 때 지정된 텍스트의 크기와 위치를 계산합니다.
GetThemeTextMetrics 특정 파트의 시각적 스타일로 지정된 글꼴에 대한 정보를 검색합니다.
GetThemeTimingFunction 타이밍 함수 식별자를 기반으로 미리 정의된 타이밍 함수를 가져옵니다.
GetThemeTransitionDuration 지정된 전환의 기간을 가져옵니다.
GetTrimPunctuationAtStart 줄의 시작 부분에 문장 부호 기호의 선행 공간을 트리밍할지 여부를 가져옵니다.
GetType 텍스트 선택 유형을 가져옵니다.
GetType 이 스토리의 형식을 가져옵니다.
GetTypographyOptions 입력 체계 옵션을 가져옵니다.
GetUnderline 범위의 문자에 대한 밑줄 형식을 가져옵니다.
GetUnderlinePositionMode 밑줄 위치 모드를 가져옵니다.
GetURL 범위와 연결된 URL 텍스트를 반환합니다.
GetVersion TOM(텍스트 개체 모델) 엔진의 버전 번호를 가져옵니다.

GetWeight 범위의 문자에 대한 글꼴 가중치를 가져옵니다.
GetWidowControl 범위의 단락에 대한 과부 및 분리 컨트롤 상태를 검색합니다.
GetWindow TOM(텍스트 개체 모델) 엔진이 출력을 표시하는 데 사용하는 창의 핸들을 가져옵니다.
GetWindowFeedbackSetting 창에 대한 피드백 구성을 검색합니다.
GetWindowTheme 시각적 스타일이 적용된 창에 대한 테마 핸들을 검색합니다.
HANDLE_WM_NOTIFY WM_NOTIFY 메시지를 처리하는 함수를 호출합니다.
HandsOffStorage 서식 있는 편집 컨트롤이 지정된 개체와 연결된 스토리지 인터페이스에 대한 참조를 해제하는 경우를 나타냅니다. 이 호출은 개체의 IRichEditOle::HandsOffStorage 메서드를 호출하지 않습니다. 호출자는 이 작업을 수행해야 합니다.
Header_ClearAllFilters 지정된 헤더 컨트롤에 대한 모든 필터를 지웁니다. 이 매크로를 사용하거나 HDM_CLEARFILTER 메시지를 명시적으로 보낼 수 있습니다.
Header_ClearFilter 지정된 헤더 컨트롤에 대한 필터를 지웁니다. 이 매크로를 사용하거나 HDM_CLEARFILTER 메시지를 명시적으로 보낼 수 있습니다.
Header_CreateDragImage 기존 헤더 컨트롤 내에서 항목 이미지의 투명 버전을 만듭니다. 이 매크로를 사용하거나 HDM_CREATEDRAGIMAGE 메시지를 명시적으로 보낼 수 있습니다.
Header_DeleteItem 헤더 컨트롤에서 항목을 삭제합니다. 이 매크로를 사용하거나 HDM_DELETEITEM 메시지를 명시적으로 보낼 수 있습니다.

Header_EditFilter 필터 단추에 포커스가 있는 경우 입력 포커스를 편집 상자로 이동합니다.
Header_GetBitmapMargin 기존 헤더 컨트롤에서 비트맵의 여백 너비(픽셀)를 가져옵니다. 이 매크로를 사용하거나 HDM_GETBITMAPMARGIN 메시지를 명시적으로 보낼 수 있습니다.
Header_GetFocusedItem 포커스가 있는 헤더 컨트롤의 항목을 가져옵니다. 이 매크로를 사용하거나 HDM_GETFOCUSEDITEM 메시지를 명시적으로 보냅니다.
Header_GetImageList 기존 헤더 컨트롤에 대해 설정된 이미지 목록에 대한 핸들을 가져옵니다. 이 매크로를 사용하거나 HDM_GETIMAGELIST 메시지를 명시적으로 보낼 수 있습니다.
Header_GetItem 헤더 컨트롤의 항목에 대한 정보를 가져옵니다. 이 매크로를 사용하거나 HDM_GETITEM 메시지를 명시적으로 보낼 수 있습니다.
Header_GetItemCount 헤더 컨트롤의 항목 수를 가져옵니다. 이 매크로를 사용하거나 HDM_GETITEMCOUNT 메시지를 명시적으로 보낼 수 있습니다.
Header_GetItemDropDownRect 헤더 컨트롤의 지정된 항목에 대한 드롭다운 단추의 좌표를 가져옵니다. 헤더 컨트롤은 HDF_SPLITBUTTON 형식이어야 합니다. 이 매크로를 사용하거나 HDM_GETITEMDROPDOWNRECT 메시지를 명시적으로 보냅니다.
Header_GetItemRect 헤더 컨트롤의 지정된 항목에 대한 경계 사각형을 가져옵니다. 이 매크로를 사용하거나 HDM_GETIMRECT 메시지를 명시적으로 보낼 수 있습니다.
Header_GetOrderArray 헤더 컨트롤에 있는 항목의 현재 왼쪽에서 오른쪽 순서를 가져옵니다. 이 매크로를 사용하거나 HDM_GETORDERARRAY 메시지를 명시적으로 보낼 수 있습니다.

Header_GetOverflowRect 지정된 헤더 컨트롤에 대한 드롭다운 오버플로 영역의 좌표를 가져옵니다. 헤더 컨트롤은 HDF_SPLITBUTTON 형식이어야 합니다. 이 매크로를 사용하거나 HDM_GETOVERFLOWRECT 메시지를 명시적으로 보냅니다.
Header_GetStateImageList 기존 헤더 컨트롤 상태에 대해 설정된 이미지 목록에 대한 핸들을 가져옵니다.
Header_GetUnicodeFormat 컨트롤의 유니코드 문자 형식 플래그를 가져옵니다. 이 매크로를 사용하거나 HDM_GETUNICODEFORMAT 메시지를 명시적으로 보낼 수 있습니다.
Header_InsertItem 헤더 컨트롤에 새 항목을 삽입합니다. 이 매크로를 사용하거나 HDM_INSERTITEM 메시지를 명시적으로 보낼 수 있습니다.
Header_Layout 부모 창 내에서 헤더 컨트롤의 올바른 크기와 위치를 검색합니다. 이 매크로를 사용하거나 HDM_LAYOUT 메시지를 명시적으로 보낼 수 있습니다.
Header_OrderToIndex 헤더 컨트롤의 순서에 따라 항목의 인덱스 값을 검색합니다. 이 매크로를 사용하거나 HDM_ORDERTOINDEX 메시지를 명시적으로 보낼 수 있습니다.
Header_SetBitmapMargin 기존 헤더 컨트롤에서 비트맵의 여백 너비를 설정합니다. 이 매크로를 사용하거나 HDM_SETBITMAPMARGIN 메시지를 명시적으로 보낼 수 있습니다.
Header_SetFilterChangeTimeout 필터 특성에서 변경이 발생하는 시간과 HDN_FILTERCHANGE 알림 게시 사이의 시간 제한 간격을 설정합니다. 이 매크로를 사용하거나 HDM_SETFILTERCHANGETIMEOUT 메시지를 명시적으로 보낼 수 있습니다.
Header_SetFocusedItem 헤더 컨트롤의 지정된 항목에 포커스를 설정합니다. 이 매크로를 사용하거나 HDM_SETFOCUSEDITEM 메시지를 명시적으로 보냅니다.

Header_SetHotDivider 외부 끌어서 놓기 작업의 대상을 나타내기 위해 헤더 항목 사이의 구분선 색을 변경합니다. 이 매크로를 사용하거나 HDM_SETHOTDIVIDER 메시지를 명시적으로 보낼 수 있습니다.
Header_SetImageList 기존 헤더 컨트롤에 이미지 목록을 할당합니다. 이 매크로를 사용하거나 HDM_SETIMAGELIST 메시지를 명시적으로 보낼 수 있습니다.
Header_SetItem 헤더 컨트롤에서 지정된 항목의 특성을 설정합니다. 이 매크로를 사용하거나 HDM_SETITEM 메시지를 명시적으로 보낼 수 있습니다.
Header_SetOrderArray 헤더 항목의 왼쪽에서 오른쪽 순서를 설정합니다. 이 매크로를 사용하거나 HDM_SETORDERARRAY 메시지를 명시적으로 보낼 수 있습니다.
Header_SetStateImageList 기존 헤더 컨트롤 상태에 이미지 목록을 할당합니다.
Header_SetUnicodeFormat 컨트롤의 유니코드 문자 형식 플래그를 설정합니다.
HexToUnicode 이 범위의 끝에 있는 16진수를 유니코드 문자로 변환하고 바꿉니다.
HIMAGELIST_QueryInterface 이미지 목록의 HIMAGELIST 핸들에 해당하는 IImageList 또는 IImageList2 개체에 대한 포인터를 검색합니다.
HitTestThemeBackground 시각적 스타일로 지정된 배경의 지점에 대한 적중 테스트 코드를 검색합니다.
HomeKey 홈 키의 기능을 일반화합니다.
HyphenateProc HyphenateProc 함수는 EM_SETHYPHENATEINFO 메시지와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. Microsoft Rich Edit 컨트롤에서 하이픈 연결이 수행되는 방식을 결정합니다.

ImageList_Add 이미지 목록에 하나의 이미지 또는 여러 이미지를 추가합니다. (ImageList_Add)
ImageList_AddIcon 이미지 목록에 아이콘 또는 커서를 추가합니다. ImageList_AddIcon 함수는 ImageList_ReplaceIcon 함수를 호출합니다.
ImageList_AddMasked 이미지 목록에 하나의 이미지 또는 여러 이미지를 추가하여 지정된 비트맵에서 마스크를 생성합니다. (ImageList_AddMasked)
ImageList_BeginDrag 이미지 끌기를 시작합니다. (ImageList_BeginDrag)
ImageList_CoCreateInstance 이미지 목록의 단일 인스턴스를 만들고 해당 인스턴스에 대한 인터페이스 포인터를 반환합니다.
ImageList_Copy 지정된 이미지 목록 내의 이미지를 복사합니다.
ImageList_Create 새 이미지 목록을 만듭니다.
ImageList_Destroy 이미지 목록을 제거합니다.
ImageList_DragEnter 창 내의 지정된 위치에 끌기 이미지를 표시합니다.
ImageList_DragLeave 지정된 창의 잠금을 해제하고 끌기 이미지를 숨겨 창을 업데이트할 수 있습니다.
ImageList_DragMove 끌어서 놓기 작업 중에 끌기 이미지를 이동합니다. 이 함수는 일반적으로 WM_MOUSEMOVE 메시지에 대한 응답으로 호출됩니다. (ImageList_DragMove)

ImageList_DragShowNolock 끌기 이미지를 표시하거나 숨깁니다. (ImageList_DragShowNolock)
ImageList_Draw 지정된 디바이스 컨텍스트에서 이미지 목록 항목을 그립니다. (ImageList_Draw)
ImageList_DrawEx 지정된 디바이스 컨텍스트에서 이미지 목록 항목을 그립니다. 이 함수는 지정된 그리기 스타일을 사용하고 이미지를 지정된 색과 혼합합니다.
ImageList_DrawIndirect IMAGELISTDRAWPARAMS 구조체를 기반으로 이미지 목록 이미지를 그립니다.
ImageList_Duplicate 기존 이미지 목록의 복제본을 만듭니다.
ImageList_EndDrag 끌기 작업을 종료합니다. (ImageList_EndDrag)
ImageList_ExtractIcon ImageList_GetIcon 함수를 호출하여 이미지 목록의 이미지 및 마스크를 기반으로 아이콘 또는 커서를 만듭니다.
ImageList_GetBkColor 이미지 목록의 현재 배경색을 검색합니다.
ImageList_GetDragImage 끌기 이미지에 사용되는 임시 이미지 목록을 검색합니다. 또한 이 함수는 현재 끌기 위치와 끌기 위치를 기준으로 끌어온 이미지의 오프셋을 검색합니다.
ImageList_GetIcon 이미지에서 아이콘을 만들고 이미지 목록의 마스크를 만듭니다.
ImageList_GetIconSize 이미지 목록에서 이미지 차원을 검색합니다. 이미지 목록의 모든 이미지는 차원이 동일합니다.

ImageList_GetImageCount 이미지 목록의 이미지 수를 검색합니다.
ImageList_GetImageInfo 이미지에 대 한 정보를 검색합니다.
ImageList_LoadBitmap ImageList_LoadImage 함수를 호출하여 지정된 비트맵 리소스에서 이미지 목록을 만듭니다.
ImageList_LoadImageA 지정된 비트맵에서 이미지 목록을 만듭니다. (ANSI)
ImageList_LoadImageW 지정된 비트맵에서 이미지 목록을 만듭니다. (유니코드)
ImageList_Merge 기존의 두 이미지를 결합하여 새 이미지를 만듭니다. 또한 이 함수는 이미지를 저장할 새 이미지 목록을 만듭니다.
ImageList_Read 스트림에서 이미지 목록을 읽습니다.
ImageList_ReadEx 스트림에서 이미지 목록을 읽고 이미지 목록에 IImageList 인터페이스를 반환합니다.
ImageList_Remove 이미지 목록에서 이미지를 제거합니다. (ImageList_Remove)
ImageList_RemoveAll ImageList_Remove 함수를 호출하여 이미지 목록에서 모든 이미지를 제거합니다.
ImageList_Replace 이미지 목록에 있는 이미지를 새 이미지로 바꿉니다. (ImageList_Replace)
ImageList_ReplaceIcon 이미지를 아이콘 또는 커서로 바꿉니다. (ImageList_ReplaceIcon)

ImageList_SetBkColor 이미지 목록의 배경색을 설정합니다. 이 함수는 아이콘을 추가하거나 흑백 비트맵이 있는 ImageList_AddMasked를 사용하는 경우에만 적용됩니다. 마스크가 없으면 전체 이미지가 그려지므로 배경색이 표시되지 않습니다.
ImageList_SetDragCursorImage 지정된 이미지(일반적으로 마우스 커서 이미지)를 현재 끌기 이미지와 결합하여 새 끌기 이미지를 만듭니다.
ImageList_SetIconSize 이미지 목록의 이미지 차원을 설정하고 목록에서 모든 이미지를 제거합니다. (ImageList_SetIconSize)
ImageList_SetImageCount 기존 이미지 목록의 크기를 조정합니다. (ImageList_SetImageCount)
ImageList_SetOverlayImage 오버레이 마스크로 사용할 이미지 목록에 지정된 이미지를 추가합니다. 이미지 목록에는 버전 4.70 이하에서 최대 4개, 버전 4.71에서 최대 15개의 오버레이 마스크를 포함할 수 있습니다. 이 함수는 지정된 이미지에 오버레이 마스크 인덱스를 할당합니다.
ImageList_Write 스트림에 이미지 목록을 씁니다. (ImageList_Write)
ImageList_WriteEx 스트림에 이미지 목록을 씁니다. (ImageList_WriteEx)
ImportDataObject 클립보드 개체를 서식 있는 편집 컨트롤로 가져와 현재 선택 영역을 바꿉니다.
INDEXTOOVERLAYMASK ImageList_Draw 함수에서 사용할 수 있도록 오버레이 마스크의 인덱스를 준비합니다.
INDEXTOSTATEIMAGEMASK 트리 뷰 컨트롤 또는 목록 뷰 컨트롤이 인덱스를 사용하여 항목의 상태 이미지를 검색할 수 있도록 상태 이미지의 인덱스를 준비합니다.

InitCommonControls 특정 공용 컨트롤 창 클래스를 등록하고 초기화합니다. 이 함수는 사용되지 않습니다. 새 애플리케이션은 InitCommonControlsEx 함수를 사용해야 합니다.
InitCommonControlsEx 공용 컨트롤 DLL(Comctl32.dll)이 로드되었는지 확인하고 DLL에서 특정 공용 컨트롤 클래스를 등록합니다. 애플리케이션은 공용 컨트롤을 만들기 전에 이 함수를 호출해야 합니다.
초기화 이미지 목록을 초기화합니다.
InitializeFlatSB 특정 창에 대한 플랫 스크롤 막대를 초기화합니다.
InitializeTouchInjection 호출 애플리케이션에 대한 터치 삽입 컨텍스트를 구성하고 앱이 삽입할 수 있는 최대 동시 접점 수를 초기화합니다.
InitMUILanguage 애플리케이션이 시스템 언어와 다른 공용 컨트롤과 함께 사용할 언어를 지정할 수 있도록 합니다.
InjectSyntheticPointerInput 포인터 입력(펜 또는 터치)을 시뮬레이션합니다.
InjectTouchInput 터치 입력을 시뮬레이션합니다.
InPlaceDeactivate 서식 있는 편집 컨트롤이 현재 활성 상태 개체(있는 경우)를 비활성화하는 경우를 나타냅니다.
InRange 이 범위가 지정된 범위와 동일한 텍스트 내에 있는지 여부를 확인합니다.
삽입 연결된 ITextRange2 개체로 식별되는 위치에 행 또는 행을 삽입합니다.

InsertImage 이 범위에 이미지를 삽입합니다.
InsertNullStr 선택한 문자열 인덱스의 컬렉션에 NULL 문자열을 삽입합니다.
InsertObject 서식 있는 편집 컨트롤에 개체를 삽입합니다.
InsertTable 범위에 테이블을 삽입합니다.
InStory 이 범위의 스토리가 지정된 범위의 스토리와 같은지 여부를 결정합니다.
IsAppThemed 현재 애플리케이션의 사용자 인터페이스가 시각적 스타일을 사용하여 표시되는지 여부를 보고합니다.
IsCompositionActive DWM(바탕 화면 창 관리자) 구성 효과를 테마에 사용할 수 있는지 여부를 결정합니다.
IsDlgButtonChecked IsDlgButtonChecked 함수는 단추 컨트롤이 선택되었는지 또는 세 가지 상태의 단추 컨트롤이 선택되었는지, 선택되지 않았는지 또는 결정되지 않았는지를 결정합니다.
IsEqual 이 텍스트 글꼴 개체의 속성이 지정된 텍스트 글꼴 개체와 같은지 여부를 확인합니다. (ITextFont.IsEqual)
IsEqual 현재 범위에 지정된 범위와 동일한 속성이 있는지 확인합니다.
IsEqual 이 범위의 문자 위치와 스토리가 지정된 범위와 같은지 여부를 결정합니다.

IsEqual 두 테이블 행을 비교하여 동일한 속성이 있는지 확인합니다.
IsEqual2 이 텍스트 글꼴 개체의 속성이 지정된 텍스트 글꼴 개체와 같은지 여부를 확인합니다. (ITextFont2.IsEqual2)
IsEqual2 이 텍스트 단락 개체의 속성이 지정된 텍스트 단락 개체와 같은지 여부를 결정합니다.
IsThemeActive 현재 애플리케이션의 시각적 스타일이 활성화되어 있는지 테스트합니다.
IsThemeBackgroundPartiallyTransparent 시각적 스타일로 지정된 배경에 투명한 부분이 있는지 또는 알파가 혼합된 부분이 있는지 검색합니다.
IsThemeDialogTextureEnabled 지정된 대화창이 배경 텍스처링을 지원하는지 여부를 보고합니다.
IsThemePartDefined 시각적 스타일에 지정된 파트 및 상태에 대한 정의된 매개 변수가 있는지 여부를 검색합니다.
IsVisible 창 없는 서식 있는 편집 컨트롤이 현재 표시되는지 여부를 나타냅니다.
Item 이 스토리 컬렉션의 Indexth 스토리에 대한 ITextRange 개체를 검색합니다.
Item 문자열 컬렉션에서 선택한 인덱스용 ITextRange2 개체를 가져옵니다.
Item2 스토리 컬렉션에서 인덱스별로 스토리에 대한 ITextRange2 개체를 가져옵니다.
LBItemFromPt 목록 상자에 지정된 지점에서 항목의 인덱스를 검색합니다.

선형화 이 범위의 기본 제공 수학, 루비 및 기타 인라인 개체를 선형화된 형식으로 변환합니다.
ListBox_AddItemData 지정된 위치의 목록 상자에 항목 데이터를 추가합니다. 이 매크로를 사용하거나 LB_ADDSTRING 메시지를 명시적으로 보낼 수 있습니다.
ListBox_AddString 목록 상자에 문자열을 추가합니다.
ListBox_DeleteString 목록 상자의 지정된 위치에 있는 항목을 삭제합니다. 이 매크로를 사용하거나 LB_DELETESTRING 메시지를 명시적으로 보낼 수 있습니다.
ListBox_Dir 목록 상자에 표시된 목록에 이름을 추가합니다.
ListBox_Enable 목록 상자 컨트롤을 사용하거나 사용하지 않도록 설정합니다.
ListBox_FindItemData 목록 상자에서 지정된 항목 데이터가 있는 첫 번째 항목을 찾습니다. 이 매크로를 사용하거나 LB_FINDSTRING 메시지를 명시적으로 보낼 수 있습니다.
ListBox_FindString 지정된 문자열로 시작하는 목록 상자에서 첫 번째 문자열을 찾습니다. 이 매크로를 사용하거나 LB_FINDSTRING 메시지를 명시적으로 보낼 수 있습니다.
ListBox_FindStringExact 대/소문자를 구분하지 않는 검색 결과를 제외하고 지정된 문자열과 정확히 일치하는 첫 번째 목록 상자 문자열을 찾습니다. 이 매크로를 사용하거나 LB_FINDSTRINGEXACT 메시지를 명시적으로 보낼 수 있습니다.
ListBox_GetCaretIndex 다중 선택 목록 상자에 포커스 사각형이 있는 목록 상자 항목의 인덱스를 검색합니다. 항목을 선택하거나 선택하지 않을 수 있습니다. 이 매크로를 사용하거나 LB_GETCARETINDEX 메시지를 명시적으로 보낼 수 있습니다.

ListBox_GetCount 목록 상자에 있는 항목의 개수를 가져옵니다. 이 매크로를 사용하거나 LB_GETCOUNT 메시지를 명시적으로 보낼 수 있습니다.
ListBox_GetCurSel 단일 선택 목록 상자에서 현재 선택한 항목의 인덱스 를 가져옵니다. 이 매크로를 사용하거나 LB_GETCURSEL 메시지를 명시적으로 보낼 수 있습니다.
ListBox_GetHorizontalExtent 목록 상자에 가로 스크롤 막대가 있는 경우 목록 상자를 가로로 스크롤할 수 있는 너비(스크롤 가능한 너비)를 가져옵니다. 이 매크로를 사용하거나 LB_GETHORIZONTALEXTENT 메시지를 명시적으로 보낼 수 있습니다.
ListBox_GetItemData 지정된 목록 상자 항목과 연결된 애플리케이션 정의 값을 가져옵니다. 이 매크로를 사용하거나 LB_GETITEMDATA 메시지를 명시적으로 보낼 수 있습니다.
ListBox_GetItemHeight 목록 상자에서 항목의 높이를 검색합니다.
ListBox_GetItemRect 목록 상자에 현재 표시되는 목록 상자 항목의 경계를 이루는 사각형의 크기를 가져옵니다. 이 매크로를 사용하거나 LB_GETITEMRECT 메시지를 명시적으로 보낼 수 있습니다.
ListBox_GetSel 항목을 선택한 상태를 가져옵니다. 이 매크로를 사용하거나 LB_GETSEL 메시지를 명시적으로 보낼 수 있습니다.
ListBox_GetSelCount 다중 선택 목록 상자에서 선택한 항목 수를 가져옵니다. 이 매크로를 사용하거나 LB_GETSELCOUNT 메시지를 명시적으로 보낼 수 있습니다.
ListBox_GetSelItems 다중 선택 목록 상자에서 선택한 항목의 인덱스를 가져옵니다. 이 매크로를 사용하거나 LB_GETSELITEMS 메시지를 명시적으로 보낼 수 있습니다.
ListBox_GetText 목록 상자에서 문자열을 가져옵니다. 이 매크로를 사용하거나 LB_GETTEXT 메시지를 명시적으로 보낼 수 있습니다.

ListBox_GetTextLen 목록 상자에서 문자열의 길이를 가져옵니다. 이 매크로를 사용하거나 LB_GETTEXTLEN 메시지를 명시적으로 보낼 수 있습니다.
ListBox_GetTopIndex 목록 상자에서 가장 먼저 표시되는 항목의 인덱스를 가져옵니다. 이 매크로를 사용하거나 LB_GETTOPINDEX 메시지를 명시적으로 보낼 수 있습니다.
ListBox_InsertItemData 지정된 위치의 목록 상자에 항목 데이터를 삽입합니다. 이 매크로를 사용하거나 LB_INSERTSTRING 메시지를 명시적으로 보낼 수 있습니다.
ListBox_InsertString 지정된 위치의 목록 상자에 문자열을 추가합니다. 이 매크로를 사용하거나 LB_INSERTSTRING 메시지를 명시적으로 보낼 수 있습니다.
ListBox_ResetContent 목록 상자에서 모든 항목을 제거합니다. 이 매크로를 사용하거나 LB_RESETCONTENT 메시지를 명시적으로 보낼 수 있습니다.
ListBox_SelectItemData 목록 상자에서 지정된 항목 데이터가 있는 항목을 검색합니다. 일치하는 항목이 발견되면 해당 항목이 선택됩니다. 이 매크로를 사용하거나 LB_SELECTSTRING 메시지를 명시적으로 보낼 수 있습니다.
ListBox_SelectString 지정된 문자열의 문자로 시작하는 항목에 대한 목록 상자를 검색합니다. 일치하는 항목을 찾으면 해당 항목이 선택됩니다. 이 매크로를 사용하거나 LB_SELECTSTRING 메시지를 명시적으로 보낼 수 있습니다.
ListBox_SelItemRange 다중 선택 목록 상자에서 하나 이상의 연속 항목을 선택하거나 선택 취소합니다. 이 매크로를 사용하거나 LB_SELITEMRANGE 메시지를 명시적으로 보낼 수 있습니다.
ListBox_SetCaretIndex 포커스 사각형을 다중 선택 목록 상자의 지정된 인덱스 항목으로 설정합니다. 항목이 표시되지 않으면 보기로 스크롤됩니다. 이 매크로를 사용하거나 LB_SETCARETINDEX 메시지를 명시적으로 보낼 수 있습니다.

ListBox_SetColumnWidth 여러 열 목록 상자에 있는 모든 열의 너비를 설정합니다. 이 매크로를 사용하거나 LB_SETCOLUMNWIDTH 메시지를 명시적으로 보낼 수 있습니다.
ListBox_SetCurSel 단일 선택 목록 상자에서 현재 선택한 항목을 설정합니다. 이 매크로를 사용하거나 LB_SETCURSEL 메시지를 명시적으로 보낼 수 있습니다.
ListBox_SetHorizontalExtent 목록 상자를 가로로 스크롤할 수 있는 너비(스크롤 가능한 너비)를 설정합니다.
ListBox_SetItemData 지정된 목록 상자 항목과 연결된 애플리케이션 정의 값을 설정합니다. 이 매크로를 사용하거나 LB_SETITEMDATA 메시지를 명시적으로 보낼 수 있습니다.
ListBox_SetItemHeight 목록 상자에서 항목의 높이를 설정합니다.
ListBox_SetSel 다중 선택 목록 상자에서 항목을 선택하거나 선택 취소합니다. 이 매크로를 사용하거나 LB_SETSEL 메시지를 명시적으로 보낼 수 있습니다.
ListBox_SetTabStops 목록 상자에서 탭 중지 위치를 설정합니다. 이 매크로를 사용하거나 LB_SETTABSTOPS 메시지를 명시적으로 보낼 수 있습니다.
ListBox_SetTopIndex 목록 상자에 지정된 항목이 표시되는지 확인합니다. 이 매크로를 사용하거나 LB_SETTOPINDEX 메시지를 명시적으로 보낼 수 있습니다.
ListView_ApproximateViewRect 지정된 항목 수를 표시하는 데 필요한 대략적인 너비와 높이를 계산합니다. 이 매크로를 사용하거나 LVM_APPROXIMATEVIEWRECT 메시지를 명시적으로 보낼 수 있습니다.
ListView_Arrange 아이콘 보기에서 항목을 정렬합니다. 이 매크로를 사용하거나 LVM_ARRANGE 메시지를 명시적으로 보낼 수 있습니다.

ListView_CancelEditLabel 항목 텍스트 편집 작업을 취소합니다. 이 매크로를 사용하거나 LVM_CANCELEDITLABEL 메시지를 명시적으로 보낼 수 있습니다.
ListView_CreateDragImage 지정된 항목에 대한 끌기 이미지 목록을 만듭니다. 이 매크로를 사용하거나 LVM_CREATEDRAGIMAGE 메시지를 명시적으로 보낼 수 있습니다.
ListView_DeleteAllItems 목록 뷰 컨트롤에서 모든 항목을 제거합니다. 이 매크로를 사용하거나 LVM_DELETEALLITEMS 메시지를 명시적으로 보낼 수 있습니다.
ListView_DeleteColumn 목록 뷰 컨트롤에서 열을 제거합니다. 이 매크로를 사용하거나 LVM_DELETECOLUMN 메시지를 명시적으로 보낼 수 있습니다.
ListView_DeleteItem 목록 뷰 컨트롤에서 항목을 제거합니다. 이 매크로를 사용하거나 LVM_DELETEITEM 메시지를 명시적으로 보낼 수 있습니다.
ListView_EditLabel 지정된 목록 뷰 항목의 텍스트를 바로 편집하기 시작합니다. 메시지는 지정된 항목을 암시적으로 선택하고 포커스를 지정합니다. 이 매크로를 사용하거나 LVM_EDITLABEL 메시지를 명시적으로 보낼 수 있습니다.
ListView_EnableGroupView 목록 뷰 컨트롤의 항목이 그룹으로 표시되는지 여부를 사용 또는 사용하지 않도록 설정합니다. 이 매크로를 사용하거나 LVM_ENABLEGROUPVIEW 메시지를 명시적으로 보낼 수 있습니다.
ListView_EnsureVisible 필요한 경우 목록 뷰 컨트롤을 스크롤하여 목록 뷰 항목이 전체적으로 또는 부분적으로 표시되도록 합니다. 이 매크로를 사용하거나 LVM_ENSUREVISIBLE 메시지를 명시적으로 보낼 수 있습니다.
ListView_FindItem 지정된 특성을 가진 목록 뷰 항목을 검색합니다. 이 매크로를 사용하거나 LVM_FINDITEM 메시지를 명시적으로 보낼 수 있습니다.

ListView_GetBkColor 목록 뷰 컨트롤에서 배경색을 가져옵니다. 이 매크로를 사용하거나 LVM_GETBKCOLOR 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetBkImage 목록 뷰 컨트롤에서 배경 이미지를 가져옵니다. 이 매크로를 사용하거나 LVM_GETBKIMAGE 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetCallbackMask 목록 뷰 컨트롤의 콜백 마스크를 가져옵니다. 이 매크로를 사용하거나 LVM_GETCALLBACKMASK 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetCheckState 목록 뷰 컨트롤의 항목이 선택되어 있는지 여부를 확인합니다. 이것은 LVS_EX_CHECKBOXES 스타일이 있는 목록 뷰 컨트롤에만 사용해야 합니다.
ListView_GetColumn 목록 뷰 컨트롤 열의 특성을 가져옵니다. 이 매크로를 사용하거나 LVM_GETCOLUMN 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetColumnOrderArray 목록 뷰 컨트롤에서 열의 현재 왼쪽에서 오른쪽 순서를 가져옵니다. 이 매크로를 사용하거나 LVM_GETCOLUMNORDERARRAY 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetColumnWidth 보고서 또는 목록 뷰에서 열의 너비를 가져옵니다. 이 매크로를 사용하거나 LVM_GETCOLUMNWIDTH 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetCountPerPage 목록 또는 보고서 뷰에 있을 때 목록 뷰 컨트롤의 표시 영역에 수직으로 들어갈 수 있는 항목 수를 계산합니다. 완전히 표시되는 항목만 계산됩니다. 이 매크로를 사용하거나 LVM_GETCOUNTPERPAGE 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetEditControl 목록 보기 항목의 텍스트를 편집하는 데 사용되는 편집 컨트롤에 대한 핸들을 가져옵니다. 이 매크로를 사용하거나 LVM_GETEDITCONTROL 메시지를 명시적으로 보낼 수 있습니다.

ListView_GetEmptyText 목록 뷰 컨트롤이 비어 있을 때 표시할 텍스트를 가져옵니다. 이 매크로를 사용하거나 LVM_GETEMPTYTEXT 메시지를 명시적으로 보냅니다.
ListView_GetExtendedListViewStyle 지정된 목록 뷰 컨트롤에 현재 사용 중인 확장 스타일을 가져옵니다. 이 매크로를 사용하거나 LVM_GETEXTENDEDLISTVIEWSTYLE 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetFocusedGroup 포커스가 있는 그룹을 가져옵니다. 이 매크로를 사용하거나 LVM_GETFOCUSEDGROUP 메시지를 명시적으로 보냅니다.
ListView_GetFooterInfo 지정된 목록 뷰 컨트롤의 바닥글에 대한 정보를 가져옵니다. 이 매크로를 사용하거나 LVM_GETFOOTERINFO 메시지를 명시적으로 보냅니다.
ListView_GetFooterItem 지정된 목록 뷰 컨트롤의 바닥글 항목에 대한 정보를 가져옵니다. 이 매크로를 사용하거나 LVM_GETFOOTERITEM 메시지를 명시적으로 보냅니다.
ListView_GetFooterItemRect 목록 뷰 컨트롤에서 지정된 항목의 바닥글 좌표를 가져옵니다. 이 매크로를 사용하거나 LVM_GETFOOTERITEMRECT 메시지를 명시적으로 보냅니다.
ListView_GetFooterRect 지정된 목록 뷰 컨트롤의 바닥글 좌표를 가져옵니다. 이 매크로를 사용하거나 LVM_GETFOOTERRECT 메시지를 명시적으로 보냅니다.
ListView_GetGroupCount 그룹의 개수를 가져옵니다. 이 매크로를 사용하거나 LVM_GETGROUPCOUNT 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetGroupHeaderImageList 기존 목록 뷰 컨트롤에 대해 설정된 그룹 헤더 이미지 목록을 가져옵니다.
ListView_GetGroupInfo 그룹 정보를 가져옵니다. 이 매크로를 사용하거나 LVM_GETGROUPINFO 메시지를 명시적으로 보낼 수 있습니다.

ListView_GetGroupInfoByIndex 지정된 그룹에 대한 정보를 가져옵니다. 이 매크로를 사용하거나 LVM_GETGROUPINFOBYINDEX 메시지를 명시적으로 보냅니다.
ListView_GetGroupMetrics 그룹 표시에 대한 정보를 가져옵니다. 이 매크로를 사용하거나 LVM_GETGROUPMETRICS 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetGroupRect 지정된 그룹의 사각형을 가져옵니다. 이 매크로를 사용하거나 LVM_GETGROUPRECT 메시지를 명시적으로 보냅니다.
ListView_GetGroupState 지정된 그룹의 상태를 가져옵니다. 이 매크로를 사용하거나 LVM_GETGROUPSTATE 메시지를 명시적으로 보냅니다.
ListView_GetHeader 목록 뷰 컨트롤에서 사용하는 헤더 컨트롤에 대한 핸들을 가져옵니다. 이 매크로를 사용하거나 LVM_GETHEADER 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetHotCursor 핫 추적을 사용하도록 설정한 상태에서 포인터가 항목 위에 있을 때 사용되는 HCURSOR 값을 검색합니다. 이 매크로를 사용하거나 LVM_GETHOTCURSOR 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetHotItem 핫 항목의 인덱스를 가져옵니다. 이 매크로를 사용하거나 LVM_GETHOTITEM 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetHoverTime 항목을 선택하기 전에 항목 위로 마우스 커서를 이동해야 하는 시간을 가져옵니다. 이 매크로를 사용하거나 LVM_GETHOVERTIME 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetImageList 목록 뷰 항목을 그리는 데 사용되는 이미지 목록에 대한 핸들을 가져옵니다. 이 매크로를 사용하거나 LVM_GETIMAGELIST 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetInsertMark 삽입 지점의 위치를 가져옵니다. 이 매크로를 사용하거나 LVM_GETINSERTMARK 메시지를 명시적으로 보낼 수 있습니다.

ListView_GetInsertMarkColor 삽입 지점의 색을 가져옵니다. 이 매크로를 사용하거나 LVM_GETINSERTMARKCOLOR 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetInsertMarkRect 삽입 지점을 경계하는 사각형을 가져옵니다. 이 매크로를 사용하거나 LVM_GETINSERTMARKRECT 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetISearchString 목록 뷰 컨트롤의 증분 검색 문자열을 가져옵니다. 이 매크로를 사용하거나 LVM_GETISEARCHSTRING 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetItem 목록 뷰 항목의 특성 중 일부 또는 전부를 가져옵니다. 이 매크로를 사용하거나 LVM_GETITEM 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetItemCount 목록 뷰 컨트롤의 항목 수를 가져옵니다. 이 매크로를 사용하거나 LVM_GETITEMCOUNT 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetItemIndexRect 지정된 목록 뷰 컨트롤의 현재 뷰에서 하위 항목의 전체 또는 일부에 대한 경계 사각형을 가져옵니다. 이 매크로를 사용하거나 LVM_GETITEMINDEXRECT 메시지를 명시적으로 보냅니다.
ListView_GetItemPosition 목록 뷰 항목의 위치를 가져옵니다. 이 매크로를 사용하거나 LVM_GETITEMPOSITION 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetItemRect 현재 뷰에서 항목의 전체 또는 일부에 대한 경계 사각형을 가져옵니다. 이 매크로를 사용하거나 LVM_GETITEMRECT 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetItemSpacing 목록 뷰 컨트롤의 항목 간 간격을 결정합니다. 이 매크로를 사용하거나 LVM_GETITEMSPACING 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetItemState 목록 뷰 항목의 상태를 가져옵니다. 이 매크로를 사용하거나 LVM_GETITEMSTATE 메시지를 명시적으로 보낼 수 있습니다.

ListView_GetItemText 목록 뷰 항목 또는 하위 항목의 텍스트를 가져옵니다. 이 매크로를 사용하거나 LVM_GETITEMTEXT 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetNextItem 지정된 속성이 있고 지정된 항목에 대한 지정된 관계를 포함하는 목록 뷰 항목을 검색합니다. 이 매크로를 사용하거나 LVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetNextItemIndex 지정된 속성과 다른 특정 항목과의 관계가 있는 특정 목록 뷰 컨트롤의 항목 인덱스를 가져옵니다. 이 매크로를 사용하거나 LVM_GETNEXTITEMINDEX 메시지를 명시적으로 보냅니다.
ListView_GetNumberOfWorkAreas 목록 뷰 컨트롤의 작업 영역 수를 가져옵니다. 이 매크로를 사용하거나 LVM_GETNUMBEROFWORKAREAS 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetOrigin 목록 뷰 컨트롤의 현재 뷰 원점을 가져옵니다. 이 매크로를 사용하거나 LVM_GETORIGIN 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetOutlineColor LVS_EX_BORDERSELECT 확장 창 스타일이 설정된 경우 목록 뷰 컨트롤의 테두리 색을 가져옵니다. 이 매크로를 사용하거나 LVM_GETOUTLINECOLOR 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetSelectedColumn 선택한 열을 지정하는 정수 를 가져옵니다. 이 매크로를 사용하거나 LVM_GETSELECTEDCOLUMN 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetSelectedCount 목록 뷰 컨트롤에서 선택한 항목의 수를 결정합니다. 이 매크로를 사용하거나 LVM_GETSELECTEDCOUNT 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetSelectionMark 목록 뷰 컨트롤에서 선택 표시를 가져옵니다. 이 매크로를 사용하거나 LVM_GETSELECTIONMARK 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetStringWidth 지정된 목록 뷰 컨트롤의 현재 글꼴을 사용하여 지정된 문자열의 너비를 결정합니다. 이 매크로를 사용하거나 LVM_GETSTRINGWIDTH 메시지를 명시적으로 보낼 수 있습니다.

ListView_GetSubItemRect 목록 뷰 컨트롤의 하위 사이트를 둘러싸는 사각형에 대한 정보를 가져옵니다.
ListView_GetTextBkColor 목록 뷰 컨트롤의 텍스트 배경색을 가져옵니다. 이 매크로를 사용하거나 LVM_GETTEXTBKCOLOR 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetTextColor 목록 뷰 컨트롤의 텍스트 색을 가져옵니다. 이 매크로를 사용하거나 LVM_GETTEXTCOLOR 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetTileInfo 목록 뷰 컨트롤의 타일에 대한 정보를 가져옵니다. 이 매크로를 사용하거나 LVM_GETTILEINFO 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetTileViewInfo 타일 뷰에서 목록 뷰 컨트롤에 대한 정보를 가져옵니다. 이 매크로를 사용하거나 LVM_GETTILEVIEWINFO 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetToolTips 목록 뷰 컨트롤이 도구 설명을 표시하는 데 사용하는 도구 설명 컨트롤을 가져옵니다. 이 매크로를 사용하거나 LVM_GETTOOLTIPS 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetTopIndex 목록 또는 보고서 뷰에 있을 때 가장 많이 표시되는 항목의 인덱스를 가져옵니다. 이 매크로를 사용하거나 LVM_GETTOPINDEX 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetUnicodeFormat 컨트롤의 유니코드 문자 형식 플래그를 가져옵니다. 이 매크로를 사용하거나 LVM_GETUNICODEFORMAT 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetView 목록 뷰 컨트롤의 현재 뷰를 가져옵니다. 이 매크로를 사용하거나 LVM_GETVIEW 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetViewRect 목록 뷰 컨트롤에 있는 모든 항목의 경계 사각형을 가져옵니다. 목록 뷰는 아이콘 또는 작은 아이콘 뷰에 있어야 합니다. 이 매크로를 사용하거나 LVM_GETVIEWRECT 메시지를 명시적으로 보낼 수 있습니다.

ListView_GetWorkAreas 목록 뷰 컨트롤에서 작업 영역을 가져옵니다. 이 매크로를 사용하거나 LVM_GETWORKAREAS 메시지를 명시적으로 보낼 수 있습니다.
ListView_HasGroup 목록 뷰 컨트롤에 지정된 그룹이 있는지 여부를 확인합니다. 이 매크로를 사용하거나 LVM_HASGROUP 메시지를 명시적으로 보낼 수 있습니다.
ListView_HitTest 있는 경우 지정된 위치에 있는 목록 뷰 항목을 결정합니다. 이 매크로를 사용하거나 LVM_HITTEST 메시지를 명시적으로 보낼 수 있습니다. (ListView_HitTest)
ListView_HitTestEx 있는 경우 지정된 위치에 있는 목록 뷰 항목을 결정합니다. 이 매크로를 사용하거나 LVM_HITTEST 메시지를 명시적으로 보낼 수 있습니다. (ListView_HitTestEx)
ListView_InsertColumn 목록 뷰 컨트롤에 새 열을 삽입합니다. 이 매크로를 사용하거나 LVM_INSERTCOLUMN 메시지를 명시적으로 보낼 수 있습니다.
ListView_InsertGroup 그룹을 목록 뷰 컨트롤에 삽입합니다. 이 매크로를 사용하거나 LVM_INSERTGROUP 메시지를 명시적으로 보낼 수 있습니다.
ListView_InsertGroupSorted 순서가 지정된 그룹 목록에 그룹을 삽입합니다. 이 매크로를 사용하거나 LVM_INSERTGROUPSORTED 메시지를 명시적으로 보낼 수 있습니다.
ListView_InsertItem 목록 뷰 컨트롤에 새 항목을 삽입합니다. 이 매크로를 사용하거나 LVM_INSERTITEM 메시지를 명시적으로 보낼 수 있습니다.
ListView_InsertMarkHitTest 지정된 지점에 가장 가까운 삽입 지점을 검색합니다. 이 매크로를 사용하거나 LVM_INSERTMARKHITTEST 메시지를 명시적으로 보낼 수 있습니다.
ListView_IsGroupViewEnabled 목록 뷰 컨트롤에 그룹 보기가 사용하도록 설정되어 있는지 확인합니다. 이 매크로를 사용하거나 LVM_ISGROUPVIEWENABLED 메시지를 명시적으로 보낼 수 있습니다.

ListView_IsItemVisible 목록 뷰 컨트롤의 항목이 표시되는지 여부를 나타냅니다. 이 매크로를 사용하거나 LVM_ISITEMVISIBLE 메시지를 명시적으로 보냅니다.
ListView_MapIDToIndex 항목의 ID를 인덱스로 매핑합니다. 이 매크로를 사용하거나 LVM_MAPIDTOINDEX 메시지를 명시적으로 보낼 수 있습니다.
ListView_MapIndexToID 항목의 인덱스가 고유 ID에 매핑됩니다. 이 매크로를 사용하거나 LVM_MAPINDEXTOID 메시지를 명시적으로 보낼 수 있습니다.
ListView_MoveGroup 이 매크로는 구현되지 않습니다. (ListView_MoveGroup)
ListView_MoveItemToGroup 이 매크로는 구현되지 않습니다. (ListView_MoveItemToGroup)
ListView_RedrawItems 목록 뷰 컨트롤이 항목 범위를 다시 그리도록 합니다. 이 매크로를 사용하거나 LVM_REDRAWITEMS 메시지를 명시적으로 보낼 수 있습니다.
ListView_RemoveAllGroups 목록 뷰 컨트롤에서 모든 그룹을 제거합니다. 이 매크로를 사용하거나 LVM_REMOVEALLGROUPS 메시지를 명시적으로 보낼 수 있습니다.
ListView_RemoveGroup 목록 뷰 컨트롤에서 그룹을 제거합니다. 이 매크로를 사용하거나 LVM_REMOVEGROUP 메시지를 명시적으로 보낼 수 있습니다.
ListView_Scroll 목록 뷰 컨트롤의 콘텐츠를 스크롤합니다. 이 매크로를 사용하거나 LVM_SCROLL 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetBkColor 목록 뷰 컨트롤에서 배경색을 설정합니다. 이 매크로를 사용하거나 LVM_SETBKCOLOR 메시지를 명시적으로 보낼 수 있습니다.

ListView_SetBkImage 목록 뷰 컨트롤에서 배경 이미지를 설정합니다. 이 매크로를 사용하거나 LVM_SETBKIMAGE 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetCallbackMask 목록 뷰 컨트롤의 콜백 마스크를 변경합니다. 이 매크로를 사용하거나 LVM_SETCALLBACKMASK 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetCheckState 목록 뷰 컨트롤에서 항목을 선택하거나 선택 취소합니다. 이 매크로를 사용하거나 LVM_SETITEMSTATE 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetColumn 목록 뷰 열의 특성을 설정합니다. 이 매크로를 사용하거나 LVM_SETCOLUMN 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetColumnOrderArray 목록 뷰 컨트롤에서 열의 왼쪽에서 오른쪽 순서를 설정합니다. 이 매크로를 사용하거나 LVM_SETCOLUMNORDERARRAY 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetColumnWidth 보고서 뷰에서 열의 너비 또는 목록 뷰 모드에서 모든 열의 너비를 변경하는 데 사용됩니다. 이 매크로를 사용하거나 LVM_SETCOLUMNWIDTH 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetExtendedListViewStyle 목록 뷰 컨트롤을 위한 확장 스타일을 설정합니다. 이 매크로를 사용하거나 LVM_SETEXTENDEDLISTVIEWSTYLE 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetExtendedListViewStyleEx 스타일 마스크를 사용하여 목록 뷰 컨트롤을 위한 확장 스타일을 설정합니다. 이 매크로를 사용하거나 LVM_SETEXTENDEDLISTVIEWSTYLE 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetGroupHeaderImageList 목록 뷰 컨트롤의 그룹 머리글에 이미지 목록을 할당합니다.
ListView_SetGroupInfo 그룹 정보를 설정합니다. 이 매크로를 사용하거나 LVM_SETGROUPINFO 메시지를 명시적으로 보낼 수 있습니다.

ListView_SetGroupMetrics 그룹 표시에 대한 정보를 설정합니다. 이 매크로를 사용하거나 LVM_SETGROUPMETRICS 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetGroupState 지정된 그룹의 상태를 설정합니다.
ListView_SetHotCursor 핫 추적이 사용하도록 설정된 상태에서 포인터가 항목 위에 있을 때 목록 뷰 컨트롤이 사용하는 HCURSOR를 설정합니다. 이 매크로를 사용하거나 LVM_SETHOTCURSOR 메시지를 명시적으로 보낼 수 있습니다. 핫 추적이 사용하도록 설정되었는지 확인하려면 SystemParametersInfo를 호출합니다.
ListView_SetHotItem 목록 뷰 컨트롤에서 핫 항목을 설정합니다. 이 매크로를 사용하거나 LVM_SETHOTITEM 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetHoverTime 항목을 선택하기 전에 항목 위로 마우스 커서를 이동해야 하는 시간을 설정합니다. 이 매크로를 사용하거나 LVM_SETHOVERTIME 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetIconSpacing 목록 뷰 컨트롤의 아이콘 사이의 간격을 LVS_ICON 스타일로 설정합니다. 이 매크로를 사용하거나 LVM_SETICONSPACING 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetImageList 목록 뷰 컨트롤에 이미지 목록을 할당합니다. 이 매크로를 사용하거나 LVM_SETIMAGELIST 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetInfoTip 도구 설명 텍스트를 설정합니다. 이 매크로를 사용하거나 LVM_SETINFOTIP 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetInsertMark 삽입 지점을 정의된 위치로 설정합니다. 이 매크로를 사용하거나 LVM_SETINSERTMARK 메시지를 명시적으로 보낼 수 있습니다.

ListView_SetInsertMarkColor 삽입 지점의 색을 설정합니다. 이 매크로를 사용하거나 LVM_SETINSERTMARKCOLOR 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetItem 목록 뷰 항목의 특성 중 일부 또는 전부를 설정합니다. ListView_SetItem을 사용하여 하위 항목의 텍스트를 설정할 수도 있습니다. 이 매크로를 사용하거나 LVM_SETITEM 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetItemCount 목록 뷰 컨트롤이 지정된 수의 항목에 대한 메모리를 할당하도록 합니다. 이 매크로를 사용하거나 LVM_SETITEMCOUNT 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetItemCountEx 가상 목록 뷰에 있는 항목의 가상 번호를 설정합니다. 이 매크로를 사용하거나 LVM_SETITEMCOUNT 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetItemIndexState 지정된 목록 뷰 항목의 상태를 설정합니다. 이 매크로를 사용하거나 LVM_SETITEMINDEXSTATE 메시지를 명시적으로 보냅니다.
ListView_SetItemPosition 항목을 목록 뷰 컨트롤(아이콘 또는 작은 아이콘 뷰)의 지정된 위치로 이동합니다. 이 매크로를 사용하거나 LVM_SETITEMPOSITION 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetItemPosition32 항목을 목록 뷰 컨트롤(아이콘 또는 작은 아이콘 뷰)의 지정된 위치로 이동합니다.
ListView_SetItemState 목록 뷰 컨트롤에서 항목의 상태를 변경합니다. 이 매크로를 사용하거나 LVM_SETITEMSTATE 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetItemText 목록 뷰 항목 또는 하위 항목의 텍스트를 변경합니다. 이 매크로를 사용하거나 LVM_SETITEMTEXT 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetOutlineColor LVS_EX_BORDERSELECT 확장 창 스타일이 설정된 경우 목록 보기 컨트롤의 테두리 색을 설정합니다. 이 매크로를 사용하거나 LVM_SETOUTLINECOLOR 메시지를 명시적으로 보낼 수 있습니다.

ListView_SetSelectedColumn 선택한 열의 인덱스를 설정합니다. 이 매크로를 사용하거나 LVM_SETSELECTEDCOLUMN 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetSelectionMark 목록 뷰 컨트롤에서 선택 표시를 설정합니다. 이 매크로를 사용하거나 LVM_SETSELECTIONMARK 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetTextBkColor list-view 컨트롤에서 텍스트의 배경색을 설정합니다. 이 매크로를 사용하거나 LVM_SETTEXTBKCOLOR 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetTextColor 목록 뷰 컨트롤의 텍스트 색을 설정합니다. 이 매크로를 사용하거나 LVM_SETTEXTCOLOR 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetTileInfo 목록 뷰 컨트롤의 기존 타일에 대한 정보를 설정합니다. 이 매크로를 사용하거나 LVM_SETTILEINFO 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetTileViewInfo 목록 뷰 컨트롤이 타일 보기에서 사용하는 정보를 설정합니다. 이 매크로를 사용하거나 LVM_SETTILEVIEWINFO 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetToolTips 목록 뷰 컨트롤이 도구 설명을 표시하는 데 사용할 도구 설명 컨트롤을 설정합니다. 이 매크로를 사용하거나 LVM_SETTOOLTIPS 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetUnicodeFormat 컨트롤의 유니코드 문자 형식 플래그를 설정합니다. (ListView_SetUnicodeFormat)
ListView_SetView 목록 뷰 컨트롤의 뷰를 설정합니다. 이 매크로를 사용하거나 LVM_SETVIEW 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetWorkAreas 목록 뷰 컨트롤 내에서 작업 영역을 설정합니다. 이 매크로를 사용하거나 LVM_SETWORKAREAS 메시지를 명시적으로 보낼 수 있습니다.

ListView_SortGroups 애플리케이션 정의 비교 함수를 사용하여 목록 뷰 컨트롤 내에서 ID별로 그룹을 정렬합니다. 이 매크로를 사용하거나 LVM_SORTGROUPS 메시지를 명시적으로 보낼 수 있습니다.
ListView_SortItems 애플리케이션 정의 비교 함수를 사용하여 목록 뷰 컨트롤의 항목을 정렬합니다. 각 항목의 인덱스는 새 시퀀스를 반영하도록 변경됩니다. 이 매크로를 사용하거나 LVM_SORTITEMS 메시지를 명시적으로 보낼 수 있습니다.
ListView_SortItemsEx 애플리케이션 정의 비교 함수를 사용하여 목록 뷰 컨트롤의 항목을 정렬합니다. 각 항목의 인덱스는 새 시퀀스를 반영하도록 변경됩니다. 이 매크로를 사용하거나 LVM_SORTITEMSEX 메시지를 명시적으로 보낼 수 있습니다.
ListView_SubItemHitTest 지정된 위치에 있는 목록 뷰 항목 또는 하위 항목을 결정합니다. 이 매크로를 사용하거나 LVM_SUBITEMHITTEST 메시지를 명시적으로 보낼 수 있습니다. (ListView_SubItemHitTest)
ListView_SubItemHitTestEx 지정된 위치에 있는 목록 뷰 항목 또는 하위 항목을 결정합니다. 이 매크로를 사용하거나 LVM_SUBITEMHITTEST 메시지를 명시적으로 보낼 수 있습니다. (ListView_SubItemHitTestEx)
ListView_Update 목록 뷰 항목을 업데이트합니다. list-view 컨트롤에 LVS_AUTOARRANGE 스타일이 있는 경우 이 매크로를 사용하면 목록 보기 컨트롤이 정렬됩니다. 이 매크로를 사용하거나 LVM_UPDATE 메시지를 명시적으로 보낼 수 있습니다.
LoadIconMetric 클라이언트 지정 시스템 메트릭을 사용하여 지정된 아이콘 리소스를 로드합니다.
LoadIconWithScaleDown 아이콘을 로드합니다. 아이콘이 표준 크기가 아닌 경우 이 함수는 작은 이미지를 확대하는 대신 큰 이미지의 크기를 조정합니다.
LPFNADDPROPSHEETPAGE 속성 시트 확장에서 속성 시트에 페이지를 추가하는 데 사용하는 애플리케이션 정의 콜백 함수를 지정합니다.

LPFNPSPCALLBACKA 페이지를 만들 때와 페이지를 제거하려고 할 때 속성 시트에서 호출하는 애플리케이션 정의 콜백 함수를 지정합니다. 애플리케이션은 이 함수를 사용하여 페이지에 대한 초기화 및 정리 작업을 수행할 수 있습니다. (ANSI)
LPFNPSPCALLBACKW 페이지를 만들 때와 페이지를 제거하려고 할 때 속성 시트에서 호출하는 애플리케이션 정의 콜백 함수를 지정합니다. 애플리케이션은 이 함수를 사용하여 페이지에 대한 초기화 및 정리 작업을 수행할 수 있습니다. (유니코드)
MakeDragList 지정된 단일 선택 목록 상자를 끌기 목록 상자로 변경합니다.
MAKEIPADDRESS 4바이트 값을 IPM_SETADDRESS 메시지에 사용하기에 적합한 단일 LPARAM으로 압축합니다.
MAKEIPRANGE 2바이트 값을 IPM_SETRANGE 메시지에 사용하기에 적합한 단일 LPARAM으로 압축합니다.
MenuHelp WM_MENUSELECT 및 WM_COMMAND 메시지를 처리하고 지정된 상태 창에 현재 메뉴에 대한 도움말 텍스트를 표시합니다.
병합 기존의 두 이미지를 결합하여 새 이미지를 만듭니다. 또한 이 메서드는 이미지를 저장할 새 이미지 목록을 만듭니다.
MonthCal_GetCalendarBorder 월 달력 컨트롤의 테두리 크기(픽셀)를 가져옵니다. 이 매크로를 사용하거나 MCM_GETCALENDARBORDER 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetCalendarCount 달력 컨트롤에 현재 표시된 달력 수를 가져옵니다. 이 매크로를 사용하거나 MCM_GETCALENDARCOUNT 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetCalendarGridInfo 일정 눈금에 대한 정보를 가져옵니다.

MonthCal_GetCALID 지정된 달력 컨트롤의 현재 달력 ID를 가져옵니다. 이 매크로를 사용하거나 MCM_GETCALID 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetColor 월 달력 컨트롤의 지정된 부분에 대한 색을 검색합니다. 이 매크로를 사용하거나 MCM_GETCOLOR 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetCurrentView 월 달력 컨트롤에 대한 뷰를 가져옵니다. 이 매크로를 사용하거나 MCM_GETCURRENTVIEW 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetCurSel 현재 선택된 날짜를 검색합니다. 이 매크로를 사용하거나 MCM_GETCURSEL 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetFirstDayOfWeek 월 달력 컨트롤의 첫 번째 요일을 검색합니다. 이 매크로를 사용하거나 MCM_GETFIRSTDAYOFWEEK 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetMaxSelCount 월 달력 컨트롤에서 선택할 수 있는 최대 날짜 범위를 검색합니다. 이 매크로를 사용하거나 MCM_GETMAXSELCOUNT 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetMaxTodayWidth 월 달력 컨트롤에서 "today" 문자열의 최대 너비를 검색합니다. 여기에는 레이블 텍스트와 날짜 텍스트가 포함됩니다. 이 매크로를 사용하거나 MCM_GETMAXTODAYWIDTH 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetMinReqRect 월 달력 컨트롤에서 전체 월을 표시하는 데 필요한 최소 크기를 검색합니다. 크기 정보는 RECT 구조체의 형태로 제공됩니다. 이 매크로를 사용하거나 MCM_GETMINREQRECT 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetMonthDelta 월 달력 컨트롤의 스크롤 속도를 검색합니다. 스크롤 속도는 사용자가 스크롤 단추를 클릭할 때 컨트롤이 디스플레이를 이동하는 개월 수입니다. 이 매크로를 사용하거나 MCM_GETMONTHDELTA 메시지를 명시적으로 보낼 수 있습니다.

MonthCal_GetMonthRange 월 달력 컨트롤 표시의 상한 및 하한을 나타내는 날짜 정보(SYSTEMTIME 구조체 사용)를 검색합니다. 이 매크로를 사용하거나 MCM_GETMONTHRANGE 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetRange 월 달력 컨트롤에 설정된 최소 및 최대 허용 날짜를 검색합니다. 이 매크로를 사용하거나 MCM_GETRANGE 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetSelRange 현재 사용자가 선택한 날짜 범위의 상한과 하한을 나타내는 날짜 정보를 가져옵니다. 이 매크로를 사용하거나 MCM_GETSELRANGE 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetToday 월 달력 컨트롤에 대해 "today"로 지정된 날짜의 날짜 정보를 검색합니다. 이 매크로를 사용하거나 MCM_GETTODAY 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetUnicodeFormat 컨트롤의 유니코드 문자 형식 플래그를 검색합니다. 이 매크로를 사용하거나 MCM_GETUNICODEFORMAT 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_HitTest 화면의 지정된 지점에 있는 월 달력 컨트롤의 일부를 결정합니다. 이 매크로를 사용하거나 MCM_HITTEST 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetCalendarBorder 월 달력 컨트롤의 테두리 크기(픽셀)를 설정합니다. 이 매크로를 사용하거나 MCM_SETCALENDARBORDER 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetCALID 지정된 달력 컨트롤의 달력 ID를 설정합니다. 이 매크로를 사용하거나 MCM_SETCALID 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetColor 월 달력 컨트롤의 지정된 부분에 대한 색을 설정합니다. 이 매크로를 사용하거나 MCM_SETCOLOR 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetCurrentView 월 달력 컨트롤에 대한 뷰를 설정합니다. 이 매크로를 사용하거나 MCM_SETCURRENTVIEW 메시지를 명시적으로 보낼 수 있습니다.

MonthCal_SetCurSel 월 달력 컨트롤에 대해 현재 선택한 날짜를 설정합니다. 지정된 날짜가 표시되지 않으면 컨트롤이 디스플레이를 업데이트하여 표시합니다. 이 매크로를 사용하거나 MCM_SETCURSEL 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetDayState 달력 컨트롤 내에서 현재 표시되는 모든 달의 날짜 상태를 설정합니다. 이 매크로를 사용하거나 MCM_SETDAYSTATE 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetFirstDayOfWeek 월 달력 컨트롤의 첫 번째 요일을 설정합니다. 이 매크로를 사용하거나 MCM_SETFIRSTDAYOFWEEK 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetMaxSelCount 월 달력 컨트롤에서 선택할 수 있는 최대 일수를 설정합니다. 이 매크로를 사용하거나 MCM_SETMAXSELCOUNT 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetMonthDelta 월 달력 컨트롤에 대한 스크롤 비율을 설정합니다. 스크롤 속도는 사용자가 스크롤 단추를 클릭할 때 컨트롤이 디스플레이를 이동하는 개월 수입니다. 이 매크로를 사용하거나 MCM_SETMONTHDELTA 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetRange 월 달력 컨트롤에 허용되는 최소 및 최대 날짜를 설정합니다. 이 매크로를 사용하거나 MCM_SETRANGE 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetSelRange 월 달력 컨트롤에 대한 선택을 지정된 날짜 범위로 설정합니다. 이 매크로를 사용하거나 MCM_SETSELRANGE 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetToday 월 달력 컨트롤에 대한 "today" 선택을 설정합니다. 이 매크로를 사용하거나 MCM_SETTODAY 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetUnicodeFormat 컨트롤의 유니코드 문자 형식 플래그를 설정합니다. (MonthCal_SetUnicodeFormat)

MonthCal_SizeRectToMin 지정된 사각형에 맞는 달력 수를 계산한 다음, 해당 달력 수에 맞게 사각형이 필요한 최소 크기를 반환합니다. 이 매크로를 사용하거나 MCM_SIZERECTTOMIN 메시지를 명시적으로 보낼 수 있습니다.
이동 삽입 지점을 지정된 단위 수만큼 앞으로 또는 뒤로 이동합니다. 범위가 비일반적인 경우 범위는 Count에 따라 양쪽 끝의 삽입 지점으로 축소된 다음 이동됩니다.
MoveBoundary 선택한 문자 수에 대해 문자열의 시작 경계를 인덱스별로 이동합니다.
MoveDown 아래쪽 화살표 및 페이지 아래쪽 키의 기능을 모방합니다.
MoveEnd 범위의 끝 위치를 이동합니다.
MoveEndUntil 범위 끝의 Count 문자 내에 있는 경우 Cset에서 지정한 문자 집합에 있는 첫 번째 문자의 문자 위치로 범위의 끝을 이동합니다.
MoveEndWhile 범위의 끝을 문자 개수로 이동하거나 Cset에서 지정한 문자 집합에 있는 모든 연속 문자를 지나갑니다.
MoveLeft 왼쪽 화살표 키의 기능을 일반화합니다.
MoveRight 오른쪽 화살표 키의 기능을 일반화합니다.
MoveStart 범위의 시작 위치를 지정된 방향으로 지정된 단위 수로 이동합니다.
MoveStartUntil 문자가 시작 위치의 Count 문자 내에 있는 경우 Cset에서 지정한 문자 집합에 있는 첫 번째 문자의 위치 범위의 시작 위치를 이동합니다.

MoveStartWhile 범위의 시작 위치를 Count 문자로 이동하거나 Cset에서 지정한 문자 집합에 있는 모든 연속 문자를 지나갑니다.
MoveUntil Cset에서 지정한 문자 집합의 첫 번째 문자에 대해 최대 개수 문자 수를 검색합니다. 문자가 발견되면 범위가 해당 지점으로 축소됩니다. 검색 시작과 방향도 Count로 지정됩니다.
MoveUp 위쪽 화살표 및 Page Up 키의 기능을 모방합니다.
MoveWhile 범위의 지정된 끝에서 시작하고 문자가 Cset에 지정된 집합에 속하는 동안 및 문자 수가 Count보다 작거나 같은 경우를 검색합니다.
새로 만들기 새 문서를 엽니다.
알림 TOM(텍스트 개체 모델) 엔진 클라이언트에 특정 IME(입력 메서드 편집기) 이벤트를 알립니다.
OnTxCharFormatChange 텍스트 호스트의 기본 문자 형식을 설정합니다.
OnTxInPlaceActivate 텍스트 서비스 개체에 이 컨트롤이 현재 위치로 활성 상태임을 알립니다.
OnTxInPlaceDeactivate 텍스트 서비스 개체에 이 컨트롤이 더 이상 활성 상태가 아님을 알립니다.
OnTxParaFormatChange 텍스트 호스트의 기본 단락 형식을 설정합니다.
OnTxPropertyBitsChange 컨트롤에 대한 속성(비트로 표시)을 설정합니다.

OnTxSetCursor 텍스트 서비스 개체에 커서를 설정하도록 알립니다.
OnTxUIActivate 이제 컨트롤이 UI 활성 상태임을 텍스트 서비스 개체에 알립니다.
OnTxUIDeactivate 텍스트 서비스 개체에 컨트롤이 더 이상 UI가 활성화되지 않음을 알립니다.
열기 지정된 문서를 엽니다. 파일의 코드 페이지뿐만 아니라 액세스 및 공유 권한, 파일 만들기 및 변환을 지정하는 매개 변수가 있습니다.
OpenThemeData 창 및 관련 클래스에 대한 테마 데이터를 엽니다.
OpenThemeDataEx 지정된 테마 클래스에 대한 창과 연결된 테마 데이터를 엽니다.
PackTouchHitTestingProximityEvaluation 근접 평가 점수와 조정된 터치 포인트 좌표를 WM_TOUCHHITTESTING 콜백의 압축된 값으로 반환합니다.
Pager_ForwardMouse Pager 컨트롤에 대한 마우스 전달을 사용하거나 사용하지 않도록 설정합니다. 마우스 전달이 사용하도록 설정되면 Pager 컨트롤이 포함된 창에 WM_MOUSEMOVE 메시지를 전달합니다. 이 매크로를 사용하거나 PGM_FORWARDMOUSE 메시지를 명시적으로 보낼 수 있습니다.
Pager_GetBkColor Pager 컨트롤의 현재 배경색을 검색합니다. 이 매크로를 사용하거나 PGM_GETBKCOLOR 메시지를 명시적으로 보낼 수 있습니다.
Pager_GetBorder Pager 컨트롤의 현재 테두리 크기를 검색합니다. 이 매크로를 사용하거나 PGM_GETBORDER 메시지를 명시적으로 보낼 수 있습니다.

Pager_GetButtonSize Pager 컨트롤의 현재 단추 크기를 검색합니다. 이 매크로를 사용하거나 PGM_GETBUTTONSIZE 메시지를 명시적으로 보낼 수 있습니다.
Pager_GetButtonState Pager 컨트롤에서 지정된 단추의 상태를 검색합니다. 이 매크로를 사용하거나 PGM_GETBUTTONSTATE 메시지를 명시적으로 보낼 수 있습니다.
Pager_GetDropTarget Pager 컨트롤의 IDropTarget 인터페이스 포인터를 검색합니다. 이 매크로를 사용하거나 PGM_GETDROPTARGET 메시지를 명시적으로 보낼 수 있습니다.
Pager_GetPos Pager 컨트롤의 현재 스크롤 위치를 검색합니다. 이 매크로를 사용하거나 PGM_GETPOS 메시지를 명시적으로 보낼 수 있습니다.
Pager_RecalcSize Pager 컨트롤이 포함된 창의 크기를 다시 계산하도록 합니다. 이 매크로를 사용하면 PGN_CALCSIZE 알림이 전송됩니다. 이 매크로를 사용하거나 PGM_RECALCSIZE 메시지를 명시적으로 보낼 수 있습니다.
Pager_SetBkColor Pager 컨트롤의 현재 배경색을 설정합니다. 이 매크로를 사용하거나 PGM_SETBKCOLOR 메시지를 명시적으로 보낼 수 있습니다.
Pager_SetBorder Pager 컨트롤의 현재 테두리 크기를 설정합니다. 이 매크로를 사용하거나 PGM_SETBORDER 메시지를 명시적으로 보낼 수 있습니다.
Pager_SetButtonSize Pager 컨트롤의 현재 단추 크기를 설정합니다. 이 매크로를 사용하거나 PGM_SETBUTTONSIZE 메시지를 명시적으로 보낼 수 있습니다.
Pager_SetChild Pager 컨트롤에 포함된 창을 설정합니다.
Pager_SetPos Pager 컨트롤의 스크롤 위치를 설정합니다. 이 매크로를 사용하거나 PGM_SETPOS 메시지를 명시적으로 보낼 수 있습니다.

Pager_SetScrollInfo 시간 제한 값, 시간 제한당 라인 및 라인당 픽셀을 포함하여 Pager 컨트롤의 스크롤 매개 변수를 설정합니다. 이 매크로를 사용하거나 PGM_SETSETSCROLLINFO 메시지를 명시적으로 보낼 수 있습니다.
붙여넣기 지정된 데이터 개체의 텍스트를 붙여넣습니다.
PFNDACOMPARE DSA_Sort에서 사용하는 비교 함수의 프로토타입을 정의합니다.
PFNDACOMPARECONST 비교 중인 항목이 상수 개체일 때 DSA_Sort에서 사용하는 비교 함수의 프로토타입을 정의합니다.
PFNDAENUMCALLBACK DSA(동적 구조 배열) 및 DPA(동적 포인터 배열) 함수에서 사용하는 콜백 함수의 프로토타입을 정의합니다.
PFNDAENUMCALLBACKCONST 관련된 항목이 상수 데이터에 대한 포인터일 때 DSA(동적 구조 배열) 및 DPA(동적 포인터 배열) 함수에서 사용하는 콜백 함수의 프로토타입을 정의합니다.
PFNDPAMERGE DPA_Merge에서 사용하는 병합 함수의 프로토타입을 정의합니다.
PFNDPAMERGECONST 상수 값을 사용하여 DPA_Merge에서 사용하는 병합 함수의 프로토타입을 정의합니다.
PFNDPASTREAM DPA_LoadStream 및 DPA_SaveStream에서 사용하는 콜백 함수의 프로토타입을 정의합니다.
PFNLVGROUPCOMPARE LVGroupCompare 함수는 LVM_INSERTGROUPSORTED 및 LVM_SORTGROUPS 메시지와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PFNPROPSHEETCALLBACK 속성 시트를 만들고 초기화할 때 시스템에서 호출하는 애플리케이션 정의 콜백 함수입니다.

PFTASKDIALOGCALLBACK TaskDialogCallbackProc 함수는 TaskDialogIndirect 함수와 함께 사용되는 애플리케이션 정의 함수입니다.
PrefixTop 문자열을 컬렉션의 맨 위 문자열에 접두사로 추가합니다.
PreloadImages 지정된 대로 이미지를 미리 로드합니다.
PropertySheetA 속성 시트를 만들고 지정된 속성 시트 헤더 구조체에 정의된 페이지를 추가합니다. (ANSI)
PropertySheetW 속성 시트를 만들고 지정된 속성 시트 헤더 구조체에 정의된 페이지를 추가합니다. (유니코드)
PropSheet_AddPage 기존 속성 시트의 끝에 새 페이지를 추가합니다. 이 매크로를 사용하거나 PSM_ADDPAGE 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_Apply 하나 이상의 페이지가 변경되었으며 변경 내용의 유효성을 검사하고 기록해야 함을 나타내는 적용 단추의 선택을 시뮬레이션합니다. 이 매크로를 사용하거나 PSM_APPLY 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_CancelToClose 가장 최근의 PSN_APPLY 알림을 취소할 수 없으므로 변경한 경우에 사용됩니다. PSM_CANCELTOCLOSE 메시지를 명시적으로 보낼 수도 있습니다.
PropSheet_Changed 페이지의 정보가 변경되었음을 속성 시트에 알립니다. 이 매크로를 사용하거나 PSM_CHANGED 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_EnableWizButtons Aero 마법사에서 단추를 사용하거나 사용하지 않도록 설정합니다. 이 매크로를 사용하거나 PSM_ENABLEWIZBUTTONS 메시지를 명시적으로 보낼 수 있습니다.

PropSheet_GetCurrentPageHwnd 속성 시트의 현재 페이지 창에 대한 핸들을 검색합니다. 이 매크로를 사용하거나 PSM_GETCURRENTPAGEHWND 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_GetResult PropertySheet에 의해 모달 속성 시트에 반환된 정보를 검색하기 위해 모덜리스 속성 시트에서 사용됩니다. 이 매크로를 사용하거나 PSM_GETRESULT 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_GetTabControl 속성 시트의 탭 컨트롤에 대한 핸들을 검색합니다. 이 매크로를 사용하거나 PSM_GETTABCONTROL 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_HwndToIndex 속성 시트 페이지의 창 핸들을 가져와서 0부터 시작하는 인덱스로 반환합니다. 이 매크로를 사용하거나 PSM_HWNDTOINDEX 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_IdToIndex 속성 시트 페이지의 리소스 식별자(ID)를 가져와서 0부터 시작하는 인덱스로 반환합니다. 이 매크로를 사용하거나 PSM_IDTOINDEX 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_IndexToHwnd 속성 시트 페이지의 인덱스를 가져와서 해당 창 핸들을 반환합니다. 이 매크로를 사용하거나 PSM_INDEXTOHWND 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_IndexToId 속성 시트 페이지의 인덱스 를 가져와서 해당 리소스 식별자(ID)를 반환합니다. 이 매크로를 사용하거나 PSM_INDEXTOID 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_IndexToPage 속성 시트 페이지의 인덱스를 가져오고 해당 HPROPSHEETPAGE 핸들을 반환합니다. 이 매크로를 사용하거나 PSM_INDEXTOPAGE 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_InsertPage 기존 속성 시트에 새 페이지를 삽입합니다. 페이지는 지정된 인덱스 또는 지정된 페이지 뒤에 삽입할 수 있습니다. 이 매크로를 사용하거나 PSM_INSERTPAGE 메시지를 명시적으로 보낼 수 있습니다.

PropSheet_IsDialogMessage 속성 시트 대화 상자에 메시지를 전달하고 대화 상자에서 메시지를 처리했는지 여부를 나타냅니다. 이 매크로를 사용하거나 PSM_ISDIALOGMESSAGE 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_PageToIndex 속성 시트 페이지의 HPROPSHEETPAGE 핸들을 가져와서 해당 인덱스(0부터 시작)를 반환합니다. 이 매크로를 사용하거나 PSM_PAGETOINDEX 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_PressButton 속성 시트 단추의 선택을 시뮬레이션합니다. 이 매크로를 사용하거나 PSM_PRESSBUTTON 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_QuerySiblings 속성 시트가 각 페이지에 PSM_QUERYSIBLINGS 메시지를 보내도록 합니다. 이 매크로를 사용하거나 PSM_QUERYSIBLINGS 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_RebootSystem 변경 내용을 적용하려면 시스템을 다시 시작해야 함을 나타냅니다. 이 매크로를 사용하거나 PSM_REBOOTSYSTEM 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_RecalcPageSizes 페이지가 추가되거나 제거된 후 표준 또는 마법사 속성 시트의 페이지 크기를 다시 계산합니다. 이 매크로를 사용하거나 PSM_RECALCPAGESIZES 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_RemovePage 속성 시트에서 페이지를 제거합니다. 이 매크로를 사용하거나 PSM_REMOVEPAGE 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_RestartWindows 변경 내용을 적용하려면 Windows를 다시 시작해야 함을 나타내는 PSM_RESTARTWINDOWS 메시지를 보냅니다. 이 매크로를 사용하거나 PSM_RESTARTWINDOWS 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_SetButtonText Aero 마법사에서 단추의 텍스트를 설정합니다. 이 매크로를 사용하거나 PSM_SETBUTTONTEXT 메시지를 명시적으로 보낼 수 있습니다.

PropSheet_SetCurSel 속성 시트에서 지정된 페이지를 활성화합니다. 이 매크로를 사용하거나 PSM_SETCURSEL 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_SetCurSelByID 페이지의 리소스 식별자를 기반으로 속성 시트에서 지정된 페이지를 활성화합니다. 이 매크로를 사용하거나 PSM_SETCURSELID 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_SetFinishText 마법사에서 마침 단추의 텍스트를 설정하고, 이 단추를 표시하고 사용하도록 설정하며, 다음 및 뒤로 단추를 숨깁니다. 이 매크로를 사용하거나 PSM_SETFINISHTEXT 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_SetHeaderSubTitle 마법사 내부 페이지의 헤더에 대한 부제 텍스트를 설정합니다. 이 매크로를 사용하거나 PSM_SETHEADERSUBTITLE 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_SetHeaderTitle 마법사 내부 페이지의 헤더에 대한 제목 텍스트를 설정합니다. 이 매크로를 사용하거나 PSM_SETHEADERTITLE 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_SetNextText 마법사에서 다음 단추의 텍스트를 설정합니다. 이 매크로를 사용하거나 PSM_SETNEXTTEXT 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_SetTitle 속성 시트의 제목을 설정합니다. 이 매크로를 사용하거나 PSM_SETTITLE 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_SetWizButtons PSM_SETWIZBUTTONS 메시지를 게시하여 마법사에서 뒤로, 다음 및 마침 단추를 사용하거나 사용하지 않도록 설정합니다. 이 매크로를 사용하거나 PSM_SETWIZBUTTONS 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_ShowWizButtons 마법사에서 단추를 표시하거나 숨깁니다. 이 매크로를 사용하거나 PSM_SHOWWIZBUTTONS 메시지를 명시적으로 보낼 수 있습니다.

PropSheet_UnChanged 페이지의 정보가 이전에 저장된 상태로 되돌아갔음을 속성 시트에 알립니다. 이 매크로를 사용하거나 PSM_UNCHANGED 메시지를 명시적으로 보낼 수 있습니다.
QueryAcceptData 붙여넣기 작업 또는 끌기 이벤트 중에 붙여넣거나 끌 수 있는 데이터를 수락해야 하는지 여부를 결정합니다.
QueryInsertObject 개체를 삽입해야 하는지 여부를 애플리케이션에 쿼리합니다. 멤버는 붙여넣을 때 및 RTF(서식 있는 텍스트 형식)를 읽을 때 호출됩니다.
Range 문서의 현재 스토리에서 지정된 범위의 콘텐츠에 대한 텍스트 범위 개체를 검색합니다.
Range2 문서의 현재 스토리에 대한 새 텍스트 범위를 검색합니다.
RangeFromPoint 화면의 지정된 지점에 있거나 가장 가까운 콘텐츠의 범위를 검색합니다.
RangeFromPoint2 화면의 특정 지점에서(또는 가장 가까운) 퇴화 범위를 검색합니다.
다시 실행 지정된 수의 다시 실행 작업을 수행합니다.
RegisterPointerDeviceNotifications 창을 등록하여 WM_POINTERDEVICECHANGE, WM_POINTERDEVICEINRANGE 및 WM_POINTERDEVICEOUTOFRANGE 포인터 디바이스 알림을 처리합니다.
RegisterTouchHitTestingWindow WM_TOUCHHITTESTING 알림을 처리하는 창을 등록합니다.
ReleaseCallManager 호출 관리자를 해제합니다.

ReleaseImmContext IMM(Input Method Manager) 입력 컨텍스트를 해제합니다.
제거 이미지 목록에서 이미지를 제거합니다. (IImageList.Remove)
제거 인덱스에서 시작하여 문자열 컬렉션에서 문자열을 제거합니다.
바꾸기 이미지 목록에 있는 이미지를 새 이미지로 바꿉니다. (IImageList.Replace)
Replace2 이미지 목록의 이미지를 대체합니다.
ReplaceFromImageList 한 이미지 목록의 이미지를 다른 이미지 목록의 이미지로 대체합니다.
ReplaceIcon 이미지를 아이콘 또는 커서로 바꿉니다. (IImageList.ReplaceIcon)
Reset 문자 서식을 지정된 값으로 다시 설정합니다.
Reset 단락 서식을 선택한 기본값으로 다시 설정합니다.
Reset 행을 다시 설정합니다.
크기 조정 현재 이미지의 크기를 조정합니다.
저장 문서를 저장합니다.

SaveCompleted 가장 최근의 저장 작업이 완료된 시기와 서식 있는 편집 컨트롤이 개체에 대한 다른 스토리지를 유지해야 했음을 나타냅니다.
ScrollBar_Enable 스크롤 막대 컨트롤을 사용하거나 사용하지 않도록 설정합니다.
ScrollBar_GetPos 지정된 스크롤 막대에서 스크롤 상자(thumb)의 위치를 검색합니다.
ScrollBar_GetRange 스크롤 막대의 범위를 가져옵니다.
ScrollBar_SetPos 지정된 스크롤 막대에서 스크롤 상자(thumb)의 위치를 설정하고, 요청된 경우 스크롤 막대를 다시 그려 스크롤 상자의 새 위치를 반영합니다. (ScrollBar_SetPos)
ScrollBar_SetRange 스크롤 막대의 범위를 설정합니다.
ScrollBar_Show 스크롤 막대 컨트롤을 표시하거나 숨깁니다.
ScrollDC ScrollDC 함수는 비트의 직사각형을 수평 및 수직으로 스크롤합니다.
ScrollIntoView 지정된 범위를 보기로 스크롤합니다.
ScrollWindow ScrollWindow 함수는 지정된 창의 클라이언트 영역 콘텐츠를 스크롤합니다.
ScrollWindowEx ScrollWindowEx 함수는 지정된 창의 클라이언트 영역 콘텐츠를 스크롤합니다.
SECOND_IPADDRESS IPM_GETADDRESS 메시지와 함께 검색된 압축된 IP 주소에서 필드 1 값을 추출합니다.

Select 현재 선택 영역의 시작 및 끝 위치와 스토리 값을 이 범위의 위치로 설정합니다.
SetActive 스토리의 활성 상태를 설정합니다. (ITextStory.SetActive)
SetActiveStory 활성 스토리를 설정합니다. 즉, 키보드 및 마우스 입력을 받는 스토리입니다.
SetActiveSubrange 지정된 하위 범위를 이 범위의 활성 하위 범위로 만듭니다.
SetAlignment 단락 맞춤을 설정합니다.
SetAlignment 행의 가로 맞춤을 설정합니다.
SetAllCaps 문자가 모두 대문자인지 여부를 설정합니다.
SetAnimation 애니메이션 유형을 설정합니다.
SetAutoLigatures 자동 합자 지원이 활성 상태인지 여부를 설정합니다.
SetAutospaceAlpha 동아시아 "autospace alpha" 상태를 설정합니다.
SetAutospaceNumeric 동아시아 "autospace numeric" 상태를 설정합니다.
SetAutospaceParens 동아시아 "autospace 괄호" 상태를 설정합니다.

SetBackColor 배경색을 설정합니다.
SetBkColor 이미지 목록의 배경색을 설정합니다.
SetBold 문자가 굵게 표시되는지 여부를 설정합니다.
SetCallback 이미지 목록 콜백을 설정합니다.
SetCaretType caret 형식을 설정합니다.
SetCellAlignment 활성 셀의 세로 맞춤을 설정합니다.
SetCellBorderColors 현재 셀의 테두리 색을 설정합니다.
SetCellBorderWidths 현재 셀의 테두리 너비를 설정합니다.
SetCellColorBack 현재 셀의 배경색을 설정합니다.
SetCellColorFore 현재 셀의 전경색을 설정합니다.
SetCellCount 행의 셀 수를 설정합니다.
SetCellCountCache 행에 대해 캐시된 셀 수를 설정합니다.

SetCellIndex 활성 셀의 인덱스 를 설정합니다.
SetCellMargin 행의 셀 여백을 설정합니다.
SetCellMergeFlags 활성 셀의 병합 플래그를 설정합니다.
SetCellShading 활성 셀의 음영을 설정합니다.
SetCellVerticalText 활성 셀의 세로 텍스트 설정을 설정합니다.
SetCellWidth 활성 셀 너비를 트윅 단위로 설정합니다.
SetChar 범위의 시작 위치에 있는 문자를 설정합니다.
SetCharRep 문자 레퍼토리(쓰기 시스템)를 설정합니다.
SetCompressionMode 동아시아 압축 모드를 설정합니다.
SetCookie 클라이언트 쿠키를 설정합니다.
SetDefaultTabStop 현재 표시 위치 이외의 탭이 없을 때 사용되는 기본 탭 중지를 설정합니다.
SetDocumentFont TOM(텍스트 개체 모델) 엔진의 이 instance 대한 기본 문자 서식을 설정합니다.

SetDocumentPara TOM(텍스트 개체 모델) 엔진의 이 instance 대한 기본 단락 서식을 설정합니다.
SetDoubleStrike 문자가 가운데를 통해 이중 가로선으로 표시되는지 여부를 설정합니다.
SetDragCursorImage 지정된 이미지(일반적으로 마우스 커서 이미지)를 현재 끌기 이미지와 결합하여 새 끌기 이미지를 만듭니다.
SetDropCap 현재 범위를 포함하는 단락의 드롭 캡 매개 변수를 설정합니다.
SetDuplicate 다른 텍스트 글꼴 개체를 복사하여 문자 서식을 설정합니다.
SetDuplicate 지정된 형식을 복사하여 기존 단락의 서식을 설정합니다.
SetDuplicate2 다른 텍스트 글꼴 개체의 속성을 복사하여 이 개체의 속성을 설정합니다.
SetDuplicate2 다른 텍스트 단락 개체의 속성을 복사하여 이 개체의 속성을 설정합니다.
SetDvaspect 서식 있는 편집 컨트롤이 개체를 그리는 데 사용하는 측면을 설정합니다. 이 호출은 개체에 캐시된 그리기 정보를 변경하지 않습니다. 호출자가 이 작업을 수행해야 합니다. 호출로 인해 개체가 다시 그려지게 됩니다.
SetEffectColor 특수 텍스트 특성에 사용할 색을 지정합니다.
SetEffects 문자 서식 효과를 설정합니다.

SetEffects 단락 서식 효과를 설정합니다.
SetEffects2 추가 문자 형식 효과를 설정합니다.
SetEmboss 문자가 엠보싱되는지 여부를 설정합니다.
SetEnd 범위의 끝 위치를 설정합니다.
SetEngrave 문자가 각인된 문자로 표시되는지 여부를 설정합니다.
SetFlags 텍스트 선택 플래그를 설정합니다.
SetFont 이 범위의 문자 특성을 지정된 ITextFont 개체의 특성으로 설정합니다.
SetFont2 범위의 문자 서식 특성을 설정합니다.
SetFontAlignment 중국어, 일본어, 한국어 텍스트의 단락 글꼴 맞춤을 설정합니다.
SetForeColor 전경(텍스트) 색을 설정합니다.
SetFormattedText 이 범위 텍스트의 서식이 지정된 텍스트를 지정된 범위의 서식이 지정된 텍스트로 설정합니다.
SetFormattedText 스토리의 텍스트를 지정된 서식이 지정된 텍스트로 대체합니다.

SetFormattedText 텍스트를 서식이 지정된 텍스트로 대체합니다.
SetFormattedText2 이 범위의 텍스트를 지정된 범위의 서식이 지정된 텍스트로 설정합니다.
SetGravity 이 범위의 중력을 설정합니다.
SetHangingPunctuation 단락이 양쪽 맞춤될 때 오른쪽 여백에 문장 부호 기호를 걸지 여부를 설정합니다.
SetHeight 행의 높이를 설정합니다.
SetHidden 문자를 숨길지 여부를 설정합니다.
SetHostNames 개체가 서식 있는 편집 컨트롤에 삽입될 때 개체에 부여할 호스트 이름을 설정합니다. 호스트 이름은 서버의 사용자 인터페이스에서 열린 개체의 컨테이너 컨텍스트를 설명하는 데 사용됩니다.
SetHyphenation 범위의 단락에 대한 하이픈을 제어합니다.
SetIconSize 이미지 목록의 이미지 차원을 설정하고 목록에서 모든 이미지를 제거합니다. (IImageList.SetIconSize)
SetImageCount 기존 이미지 목록의 크기를 조정합니다. (IImageList.SetImageCount)
SetIMEInProgress IME(입력 메서드 편집기) 진행 중 플래그의 상태를 설정합니다.
SetIndent 행의 들여쓰기를 설정합니다.

SetIndents 단락의 첫 줄 들여쓰기, 왼쪽 들여쓰기 및 오른쪽 들여쓰기를 설정합니다.
SetIndex 이 범위를 스토리의 지정된 단위로 변경합니다.
SetInlineObject 퇴화 범위에 대한 인라인 개체의 속성을 설정하거나 삽입합니다.
SetItalic 문자가 기울임꼴인지 여부를 설정합니다.
SetKeepTogether 범위의 단락 내에서 페이지 나누기가 허용되는지 여부를 제어합니다.
SetKeepTogether 이 행이 여러 페이지에서 끊어질 수 있는지 여부를 설정합니다.
SetKeepWithNext 범위의 단락 간에 페이지 나누기가 허용되는지 여부를 제어합니다.
SetKeepWithNext 행이 다음 행과 동일한 페이지에 표시되어야 하는지 여부를 설정합니다.
SetKerning 커닝이 발생하는 최소 글꼴 크기를 설정합니다.
SetLanguageID 언어 ID 또는 LCID(언어 코드 식별자)를 설정합니다.
SetLineSpacing 단락 줄 간격 규칙과 단락의 줄 간격을 설정합니다.
SetLinkAvailable 개체의 플래그에서 링크 사용 가능한 비트의 값을 설정합니다.

SetListAlignment 단락에 사용되는 글머리 기호 또는 번호 매기기 텍스트의 맞춤을 설정합니다.
SetListLevelIndex 단락에 사용되는 목록 수준 인덱스를 설정합니다.
SetListStart 번호 매기기 목록의 시작 번호 또는 유니코드 값을 설정합니다.
SetListTab 첫 번째 들여쓰기와 첫 번째 줄의 텍스트 시작 사이의 거리인 목록 탭 설정을 설정합니다.
SetListType 단락에 사용할 목록 유형을 설정합니다.
SetMathProperties 문서에 사용할 수학 속성을 지정합니다.
SetMathZone 수학 영역이 활성 상태인지 여부를 설정합니다.
SetModWidthPairs "쌍의 너비 감소"가 활성 상태인지 여부를 설정합니다.
SetModWidthSpace "공백 너비 늘리기"가 활성 상태인지 여부를 설정합니다.
SetName 글꼴 이름을 설정합니다.
SetNoLineNumber 범위에서 단락의 줄 번호 매기기를 표시하지 않을지 여부를 결정합니다.
SetNotificationMode 알림 모드를 설정합니다.

SetOldNumbers 이전 스타일 번호가 활성 상태인지 여부를 설정합니다.
SetOpCp 원하는 문자 서식 특성이 있는 원본 범위의 스토리에서 문자 위치를 설정합니다.
SetOriginalSize 지정된 이미지의 원래 크기를 설정합니다.
SetOutline 문자가 윤곽선으로 표시되는지 여부를 설정합니다.
SetOverlapping 겹치는 텍스트가 활성 상태인지 여부를 설정합니다.
SetOverlayImage 오버레이 마스크로 사용되는 이미지 목록에 지정된 이미지를 추가합니다.
SetPageBreakBefore 범위의 각 단락 앞에 페이지 나누기가 있는지 여부를 제어합니다.
SetPara 이 범위의 단락 특성을 지정된 ITextPara 개체의 단락 특성으로 설정합니다.
SetPara2 범위의 단락 서식 특성을 설정합니다.
SetPoint 형식에 따라 정렬된 점(x, y)을 확장에 따라 또는 위쪽에서 지정된 지점에 따라 범위를 변경합니다.
SetPosition 기준선을 기준으로 문자가 세로로 오프셋되는 양을 설정합니다.
SetPositionSubSuper 기준선을 기준으로 첨자 또는 위 첨자의 위치를 글꼴 높이의 백분율로 설정합니다.

SetProperty 속성에 대한 새 값을 지정합니다.
SetProperty 지정한 속성의 값을 설정합니다. (ITextFont2.SetProperty)
SetProperty 속성 값을 설정합니다.
SetProperty 지정한 속성의 값을 설정합니다. (ITextRange2.SetProperty)
SetProperty 지정한 속성의 값을 설정합니다. (ITextRow.SetProperty)
SetProperty 지정한 속성의 값을 설정합니다. (ITextStory.SetProperty)
SetProtected 문자를 수정하려는 시도로부터 문자가 보호되는지 여부를 설정합니다.
SetRange 범위 엔드포인트를 지정된 값으로 조정합니다.
SetRightIndent 단락의 오른쪽 여백을 설정합니다.
SetRTL 이 행의 오른쪽에서 왼쪽 방향이 있는지 여부를 설정합니다.
SetSaved 저장된 문서 속성을 설정합니다.
SetScaling 글꼴 가로 크기 조정 백분율을 설정합니다.

SetScrollInfo SetScrollInfo 함수는 최소 및 최대 스크롤 위치, 페이지 크기, 스크롤 상자(엄지)의 위치를 포함하여 스크롤 막대의 매개 변수를 설정합니다. 또한 이 함수는 요청된 경우 스크롤 막대를 다시 그립니다.
SetScrollPos SetScrollPos 함수는 지정된 스크롤 막대에서 스크롤 상자(엄지)의 위치를 설정하고 요청 시 스크롤 막대를 다시 그려서 스크롤 상자의 새 위치를 반영합니다.
SetScrollRange SetScrollRange 함수는 지정된 스크롤 막대의 최소 및 최대 스크롤 상자 위치를 설정합니다.
SetShadow 문자가 그림자 문자로 표시되는지 여부를 설정합니다.
SetSize 글꼴 크기를 설정합니다.
SetSmallCaps 문자가 작은 대문자인지 여부를 설정합니다.
SetSnapToGrid 단락 줄이 전체 문서에 대해 정의할 수 있는 세로 눈금에 맞춰지는지 여부를 설정합니다.
SetSpaceAfter 단락 뒤에 있는 공간의 크기를 설정합니다.
SetSpaceBefore 단락 앞의 간격을 설정합니다.
SetSpaceExtension 동아시아 공간 확장 값을 설정합니다.
SetSpacing 문자 간의 가로 간격을 설정합니다.
SetStart 이 범위의 시작 부분에 대한 문자 위치를 설정합니다.

SetStrikeThrough 문자가 가운데를 통해 가로선으로 표시되는지 여부를 설정합니다.
SetStyle 범위에 있는 문자의 문자 스타일 핸들을 설정합니다.
SetStyle 범위에서 단락의 단락 스타일을 설정합니다.
SetSubscript 문자가 아래 첨자로 표시되는지 여부를 설정합니다.
SetSuperscript 문자가 위 첨자로 표시되는지 여부를 설정합니다.
SetText 이 범위의 텍스트를 설정합니다.
SetText 스토리의 텍스트를 지정된 텍스트로 바꿉니다.
SetText2 이 범위의 텍스트를 설정합니다.
SetThemeAppProperties 호출 애플리케이션에서 시각적 스타일이 구현되는 방식을 결정하는 플래그를 설정합니다.
SetTrimPunctuationAtStart 줄의 시작 부분에 있는 문장 부호 기호의 선행 공간을 트리밍할지 여부를 설정합니다.
SetType 스토리 유형을 설정합니다.
SetTypographyOptions 문서의 입력 체계 옵션을 지정합니다.

SetUnderline 범위의 문자에 대한 밑줄 형식을 설정합니다.
SetUnderlinePositionMode 밑줄 위치 모드를 설정합니다.
SetURL 이 범위의 텍스트를 지정된 URL의 텍스트로 설정합니다.
SetWeight 범위의 문자에 대한 글꼴 가중치를 설정합니다.
SetWidowControl 과부와 고아의 억제를 제어합니다.
SetWindowFeedbackSetting 창에 대한 피드백 구성을 설정합니다.
SetWindowTheme 창에서 해당 클래스가 일반적으로 사용하는 것과 다른 시각적 스타일 정보 집합을 사용하도록 합니다.
SetWindowThemeAttribute 시각적 스타일이 지정된 창에 적용되는 방식을 제어하는 특성을 설정합니다.
SetWindowThemeNonClientAttributes 비클라이언트 특성을 설정하여 시각적 스타일이 지정된 창에 적용되는 방식을 제어합니다.
ShowContainerUI 애플리케이션이 컨테이너 UI를 표시할지 여부를 나타냅니다.
ShowHideMenuCtl 지정된 메뉴 항목의 확인 표시 특성을 설정하거나 제거하고 해당 컨트롤을 표시하거나 숨깁니다.
ShowScrollBar ShowScrollBar 함수는 지정된 스크롤 막대를 표시하거나 숨깁니다.

ShutdownTextServices 텍스트 서비스 instance 호스트의 연결을 끊습니다.
StartOf 범위 끝을 범위의 첫 번째 겹치는 단위의 시작으로 이동합니다.
Static_Enable 정적 컨트롤을 사용하거나 사용하지 않도록 설정합니다.
Static_GetIcon SS_ICON 스타일이 있는 정적 컨트롤과 연결된 아이콘에 대한 핸들을 검색합니다. 이 매크로를 사용하거나 STM_GETICON 메시지를 명시적으로 보낼 수 있습니다.
Static_GetText 정적 컨트롤의 텍스트를 가져옵니다.
Static_GetTextLength 정적 컨트롤의 텍스트에 있는 문자 수를 가져옵니다.
Static_SetIcon 정적 컨트롤의 아이콘을 설정합니다. 이 매크로를 사용하거나 STM_SETICON 메시지를 명시적으로 보낼 수 있습니다.
Static_SetText 정적 컨트롤의 텍스트를 설정합니다.
Str_SetPtrW ppszCurrent를 pszNew의 복사본으로 설정하고 필요한 경우 이전 값을 해제합니다.
접미사Top 문자열을 컬렉션의 위쪽 문자열에 접미사로 붙입니다.
스왑 컬렉션의 상위 두 문자열을 교환합니다.
SysBeep 시스템 경고음을 생성합니다.

TabCtrl_AdjustRect 창 사각형이 지정된 탭 컨트롤의 표시 영역을 계산하거나 지정된 표시 영역에 해당하는 창 사각형을 계산합니다. 이 매크로를 사용하거나 TCM_ADJUSTRECT 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_DeleteAllItems 탭 컨트롤에서 모든 항목을 제거합니다. 이 매크로를 사용하거나 TCM_DELETEALLITEMS 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_DeleteItem 탭 컨트롤에서 항목을 제거합니다. 이 매크로를 사용하거나 TCM_DELETEITEM 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_DeselectAll 탭 컨트롤의 항목을 초기화하여 TCIS_BUTTONPRESSED 상태로 설정된 모든 항목을 지웁니다. 이 매크로를 사용하거나 TCM_DESELECTALL 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_GetCurFocus 탭 컨트롤에 포커스가 있는 항목의 인덱스를 반환합니다. 이 매크로를 사용하거나 TCM_GETCURFOCUS 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_GetCurSel 탭 컨트롤에서 현재 선택된 탭을 결정합니다. 이 매크로를 사용하거나 TCM_GETCURSEL 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_GetExtendedStyle 현재 탭 컨트롤에 사용 중인 확장 스타일을 검색합니다. 이 매크로를 사용하거나 TCM_GETEXTENDEDSTYLE 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_GetImageList 탭 컨트롤과 연결된 이미지 목록을 검색합니다. 이 매크로를 사용하거나 TCM_GETIMAGELIST 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_GetItem 탭 컨트롤의 탭에 대한 정보를 검색합니다. 이 매크로를 사용하거나 TCM_GETITEM 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_GetItemCount 탭 컨트롤에서 탭 수를 검색합니다. 이 매크로를 사용하거나 TCM_GETITEMCOUNT 메시지를 명시적으로 보낼 수 있습니다.

TabCtrl_GetItemRect 탭 컨트롤의 탭에 대한 경계 사각형을 검색합니다. 이 매크로를 사용하거나 TCM_GETIMRECT 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_GetRowCount 탭 컨트롤에서 탭의 현재 행 수를 검색합니다. 이 매크로를 사용하거나 TCM_GETROWCOUNT 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_GetToolTips 탭 컨트롤과 연결된 도구 설명 컨트롤에 대한 핸들을 검색합니다. 이 매크로를 사용하거나 TCM_GETTOOLTIPS 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_GetUnicodeFormat 컨트롤의 유니코드 문자 형식 플래그를 검색합니다. 이 매크로를 사용하거나 TCM_GETUNICODEFORMAT 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_HighlightItem 탭 항목의 강조 표시 상태를 설정합니다. 이 매크로를 사용하거나 TCM_HIGHLIGHTITEM 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_HitTest 지정된 화면 위치에 있는 탭(있는 경우)을 결정합니다. 이 매크로를 사용하거나 TCM_HITTEST 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_InsertItem 탭 컨트롤에 새 탭을 삽입합니다. 이 매크로를 사용하거나 TCM_INSERTITEM 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_RemoveImage 탭 컨트롤의 이미지 목록에서 이미지를 제거합니다. 이 매크로를 사용하거나 TCM_REMOVEIMAGE 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_SetCurFocus 탭 컨트롤의 지정된 탭에 포커스를 설정합니다. 이 매크로를 사용하거나 TCM_SETCURFOCUS 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_SetCurSel 탭 컨트롤에서 탭을 선택합니다. 이 매크로를 사용하거나 TCM_SETCURSEL 메시지를 명시적으로 보낼 수 있습니다.

TabCtrl_SetExtendedStyle 탭 컨트롤이 사용할 확장 스타일을 설정합니다. 이 매크로를 사용하거나 TCM_SETEXTENDEDSTYLE 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_SetImageList 탭 컨트롤에 이미지 목록을 할당합니다. 이 매크로를 사용하거나 TCM_SETIMAGELIST 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_SetItem 탭 특성의 일부 또는 전부를 설정합니다. 이 매크로를 사용하거나 TCM_SETITEM 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_SetItemExtra 탭 컨트롤에서 애플리케이션 정의 데이터용으로 예약된 탭당 바이트 수를 설정합니다. 이 매크로를 사용하거나 TCM_SETITEMEXTRA 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_SetItemSize 고정 너비 또는 소유자가 그린 탭 컨트롤에서 탭의 너비와 높이를 설정합니다. 이 매크로를 사용하거나 TCM_SETITEMSIZE 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_SetMinTabWidth 탭 컨트롤에서 항목의 최소 너비를 설정합니다. 이 매크로를 사용하거나 TCM_SETMINTABWIDTH 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_SetPadding 탭 컨트롤에서 각 탭의 아이콘과 레이블 주변의 공간(패딩)을 설정합니다. 이 매크로를 사용하거나 TCM_SETPADDING 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_SetToolTips 탭 컨트롤에 도구 설명 컨트롤을 할당합니다. 이 매크로를 사용하거나 TCM_SETTOOLTIPS 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_SetUnicodeFormat 컨트롤의 유니코드 문자 형식 플래그를 설정합니다. (TabCtrl_SetUnicodeFormat)
TaskDialog TaskDialog 함수는 작업 대화 상자를 만들고, 표시하고, 작동합니다.

TaskDialogIndirect TaskDialogIndirect 함수는 작업 대화 상자를 만들고, 표시하고, 작동합니다.
THIRD_IPADDRESS IPM_GETADDRESS 메시지와 함께 검색된 압축된 IP 주소에서 필드 2 값을 추출합니다.
TreeView_CreateDragImage 트리 뷰 컨트롤에서 지정된 항목에 대한 끌기 비트맵을 만듭니다.
TreeView_DeleteAllItems 트리 뷰 컨트롤에서 모든 항목을 삭제합니다.
TreeView_DeleteItem 트리 뷰 컨트롤에서 항목과 모든 자식을 제거합니다. TVM_DELETEITEM 메시지를 명시적으로 보낼 수도 있습니다.
TreeView_EditLabel 지정된 항목의 텍스트에 대한 현재 위치 편집을 시작하여 항목의 텍스트를 텍스트가 포함된 한 줄 편집 컨트롤로 대체합니다.
TreeView_EndEditLabelNow 트리 뷰 항목의 레이블 편집을 종료합니다. 이 매크로를 사용하거나 TVM_ENDEDITLABELNOW 메시지를 명시적으로 보낼 수 있습니다.
TreeView_EnsureVisible 필요한 경우 트리 뷰 항목이 표시되고 부모 항목이 확장되거나 트리 뷰 컨트롤이 스크롤되도록 합니다. 이 매크로를 사용하거나 TVM_ENSUREVISIBLE 메시지를 명시적으로 보낼 수 있습니다.
TreeView_Expand TreeView_Expand 매크로는 지정된 부모 항목(있는 경우)과 연결된 자식 항목 목록을 확장하거나 축소합니다. 이 매크로를 사용하거나 TVM_EXPAND 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetBkColor 컨트롤의 현재 배경색을 검색합니다. 이 매크로를 사용하거나 TVM_GETBKCOLOR 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetCheckState 지정된 항목의 확인 상태를 가져옵니다. TVM_GETITEMSTATE 메시지를 직접 사용할 수도 있습니다.

TreeView_GetChild 지정된 트리 뷰 항목의 첫 번째 자식 항목을 검색합니다. 이 매크로를 사용하거나 TVGN_CHILD 플래그와 함께 TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetCount 트리 뷰 컨트롤에서 항목 수를 검색합니다. 이 매크로를 사용하거나 TVM_GETCOUNT 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetDropHilight 끌어서 놓기 작업의 대상인 트리 뷰 항목을 검색합니다. 이 매크로를 사용하거나 TVGN_DROPHILITE 플래그와 함께 TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetEditControl 트리 뷰 항목의 텍스트를 편집하는 데 사용되는 편집 컨트롤에 대한 핸들을 검색합니다. 이 매크로를 사용하거나 TVM_GETEDITCONTROL 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetExtendedStyle 지정된 트리 뷰 컨트롤에 대한 확장 스타일을 검색합니다. 이 매크로를 사용하거나 TVM_GETEXTENDEDSTYLE 메시지를 명시적으로 보냅니다.
TreeView_GetFirstVisible 트리 뷰 컨트롤 창에서 처음 표시되는 항목을 검색합니다. 이 매크로를 사용하거나 TVGN_FIRSTVISIBLE 플래그와 함께 TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetImageList 트리 뷰 컨트롤과 연결된 일반 또는 상태 이미지 목록에 대한 핸들을 검색합니다. 이 매크로를 사용하거나 TVM_GETIMAGELIST 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetIndent 자식 항목이 부모 항목에 비해 들여쓰기가 적용된 양을 픽셀 단위로 검색합니다. 이 매크로를 사용하거나 TVM_GETINDENT 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetInsertMarkColor 트리 뷰의 삽입 마크를 그리는 데 사용되는 색을 검색합니다. 이 매크로를 사용하거나 TVM_GETINSERTMARKCOLOR 메시지를 명시적으로 보낼 수 있습니다.

TreeView_GetISearchString 트리 뷰 컨트롤에 대한 증분 검색 문자열을 검색합니다. 트리 뷰 컨트롤은 증분 검색 문자열을 사용하여 사용자가 입력한 문자를 기반으로 항목을 선택합니다. 이 매크로를 사용하거나 TVM_GETISEARCHSTRING 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetItem 트리 뷰 항목의 특성 중 일부 또는 전부를 검색합니다. 이 매크로를 사용하거나 TVM_GETITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetItemHeight 트리 뷰 항목의 현재 높이를 검색합니다. 이 매크로를 사용하거나 TVM_GETITEMHEIGHT 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetItemPartRect 항목의 지정된 부분에 대한 "적중 영역"을 구성하는 가능한 가장 큰 경계 사각형을 검색합니다. 이 매크로를 사용하거나 TVM_GETITEMPARTRECT 메시지를 명시적으로 보냅니다.
TreeView_GetItemRect 트리 뷰 항목의 경계 사각형을 검색하고 항목이 표시되는지 여부를 나타냅니다. 이 매크로를 사용하거나 TVM_GETITEMRECT 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetItemState 트리 뷰 항목의 상태 특성 중 일부 또는 전부를 검색합니다. 이 매크로를 사용하거나 TVM_GETITEMSTATE 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetLastVisible 트리 뷰 컨트롤에서 마지막으로 확장된 항목을 검색합니다. 트리 뷰 창에 표시되는 마지막 항목은 검색하지 않습니다. 이 매크로를 사용하거나 TVGN_LASTVISIBLE 플래그와 함께 TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetLineColor 현재 선 색을 가져옵니다. TVM_GETLINECOLOR 메시지를 직접 사용할 수도 있습니다.
TreeView_GetNextItem 지정된 항목과 지정된 관계를 포함하는 트리 보기 항목을 검색합니다. 이 매크로를 사용하거나, 아래에 설명된 TreeView_Get 매크로 중 하나를 사용하거나, TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.

TreeView_GetNextSelected 지정된 트리 항목에 대한 TVGN_NEXTSELECTED 관계를 포함하는 트리 뷰 항목을 검색합니다.
TreeView_GetNextSibling 트리 뷰 컨트롤에서 지정된 항목의 다음 형제 항목을 검색합니다. 이 매크로를 사용하거나 TVGN_NEXT 플래그와 함께 TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetNextVisible 트리 뷰 컨트롤에서 지정된 항목 다음에 표시되는 다음 항목을 검색합니다. 이 매크로를 사용하거나 TVGN_NEXTVISIBLE 플래그와 함께 TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetParent 지정된 트리 뷰 항목의 부모 항목을 검색합니다. 이 매크로를 사용하거나 TVGN_PARENT 플래그와 함께 TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetPrevSibling 트리 뷰 컨트롤에서 지정된 항목의 이전 형제 항목을 검색합니다. 이 매크로를 사용하거나 TVGN_PREVIOUS 플래그와 함께 TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetPrevVisible 트리 뷰 컨트롤에서 지정된 항목 앞에 표시되는 첫 번째 항목을 검색합니다. 이 매크로를 사용하거나 TVGN_PREVIOUSVISIBLE 플래그와 함께 TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetRoot 트리 뷰 컨트롤의 맨 위 또는 첫 번째 항목을 검색합니다. 이 매크로를 사용하거나 TVGN_ROOT 플래그와 함께 TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetScrollTime 트리 뷰 컨트롤의 최대 스크롤 시간을 검색합니다. 이 매크로를 사용하거나 TVM_GETSCROLLTIME 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetSelectedCount TreeView_GetSelectedCount 매크로
TreeView_GetSelection 트리 뷰 컨트롤에서 현재 선택된 항목을 검색합니다. 이 매크로를 사용하거나 TVGN_CARET 플래그와 함께 TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.

TreeView_GetTextColor 컨트롤의 현재 텍스트 색을 검색합니다. 이 매크로를 사용하거나 TVM_GETTEXTCOLOR 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetToolTips 트리 뷰 컨트롤에서 사용하는 자식 도구 설명 컨트롤에 대한 핸들을 검색합니다. 이 매크로를 사용하거나 TVM_GETTOOLTIPS 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetUnicodeFormat 컨트롤의 유니코드 문자 형식 플래그를 검색합니다. 이 매크로를 사용하거나 TVM_GETUNICODEFORMAT 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetVisibleCount 트리 뷰 컨트롤의 클라이언트 창에서 완전히 볼 수 있는 항목 수를 가져옵니다. 이 매크로를 사용하거나 TVM_GETVISIBLECOUNT 메시지를 명시적으로 보낼 수 있습니다.
TreeView_HitTest 트리 뷰 컨트롤의 클라이언트 영역을 기준으로 지정된 지점의 위치를 결정합니다. 이 매크로를 사용하거나 TVM_HITTEST 메시지를 명시적으로 보낼 수 있습니다.
TreeView_InsertItem 트리 뷰 컨트롤에 새 항목을 삽입합니다. 이 매크로를 사용하거나 TVM_INSERTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_MapAccIDToHTREEITEM 접근성 ID를 HTREEITEM에 매핑합니다. 이 매크로를 사용하거나 TVM_MAPACCIDTOHTREEITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_MapHTREEITEMToAccID HTREEITEM을 접근성 ID에 매핑합니다. 이 매크로를 사용하거나 TVM_MAPHTREEITEMTOACCID 메시지를 명시적으로 보낼 수 있습니다.
TreeView_Select 지정된 트리 뷰 항목을 선택하거나 항목을 보기로 스크롤하거나 끌어서 놓기 작업의 대상을 나타내는 데 사용되는 스타일로 항목을 다시 그립니다.

TreeView_SelectDropTarget 끌어서 놓기 작업의 대상을 나타내는 데 사용되는 스타일에서 지정된 트리 뷰 컨트롤 항목을 다시 그립니다. 이 매크로 또는 TreeView_Select 매크로를 사용하거나 TVM_SELECTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SelectItem 지정된 트리 뷰 항목을 선택합니다. 이 매크로 또는 TreeView_Select 매크로를 사용하거나 TVM_SELECTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SelectSetFirstVisible 트리 뷰 컨트롤을 세로로 스크롤하여 지정된 항목이 표시되는지 확인합니다.
TreeView_SetAutoScrollInfo 자동 스크롤 특성을 확인하는 데 사용되는 정보를 설정합니다. 이 매크로를 사용하거나 TVM_SETAUTOSCROLLINFO 메시지를 명시적으로 보냅니다.
TreeView_SetBkColor 컨트롤의 배경색을 설정합니다. 이 매크로를 사용하거나 TVM_SETBKCOLOR 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetBorder 트리 뷰 컨트롤의 항목에 대한 테두리 크기를 설정합니다. 이 매크로를 사용하거나 TVM_SETBORDER 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetCheckState 항목의 상태 이미지를 "checked" 또는 "unchecked"로 설정합니다. TVM_SETITEM 메시지를 직접 사용할 수도 있습니다.
TreeView_SetExtendedStyle 지정된 TreeView 컨트롤의 확장 스타일을 설정합니다. 이 매크로를 사용하거나 TVM_SETEXTENDEDSTYLE 메시지를 명시적으로 보냅니다.
TreeView_SetHot 트리 뷰 컨트롤에 대한 핫 항목을 설정합니다. 이 매크로를 사용하거나 TVM_SETHOT 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetImageList 트리 뷰 컨트롤에 대한 일반 또는 상태 이미지 목록을 설정하고 새 이미지를 사용하여 컨트롤을 다시 그립니다. 이 매크로를 사용하거나 TVM_SETIMAGELIST 메시지를 명시적으로 보낼 수 있습니다.

TreeView_SetIndent 트리 뷰 컨트롤의 들여쓰기 너비를 설정하고 새 너비를 반영하도록 컨트롤을 다시 그립니다. 이 매크로를 사용하거나 TVM_SETINDENT 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetInsertMark 트리 뷰 컨트롤에서 삽입 표시를 설정합니다. 이 매크로를 사용하거나 TVM_SETINSERTMARK 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetInsertMarkColor 트리 뷰의 삽입 표시를 그리는 데 사용되는 색을 설정합니다. 이 매크로를 사용하거나 TVM_SETINSERTMARKCOLOR 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetItem TreeView_SetItem 매크로는 트리 뷰 항목 특성의 일부 또는 전부를 설정합니다. 이 매크로를 사용하거나 TVM_SETITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetItemHeight 트리 뷰 항목의 높이를 설정합니다. 이 매크로를 사용하거나 TVM_SETITEMHEIGHT 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetItemState 트리 뷰 항목의 상태 특성을 설정합니다. 이 매크로를 사용하거나 TVM_SETITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetLineColor 현재 선 색을 설정합니다. TVM_SETLINECOLOR 메시지를 직접 사용할 수도 있습니다.
TreeView_SetScrollTime 트리 뷰 컨트롤의 최대 스크롤 시간을 설정합니다. 이 매크로를 사용하거나 TVM_SETSCROLLTIME 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetTextColor 컨트롤의 텍스트 색을 설정합니다. 이 매크로를 사용하거나 TVM_SETTEXTCOLOR 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetToolTips 트리 뷰 컨트롤의 자식 도구 설명 컨트롤을 설정합니다. 이 매크로를 사용하거나 TVM_SETTOOLTIPS 메시지를 명시적으로 보낼 수 있습니다.

TreeView_SetUnicodeFormat 컨트롤의 유니코드 문자 형식 플래그를 설정합니다. (TreeView_SetUnicodeFormat)
TreeView_ShowInfoTip 트리 뷰 컨트롤에서 지정된 항목에 대한 infotip을 표시합니다. 이 매크로를 사용하거나 TVM_SHOWINFOTIP 메시지를 명시적으로 보냅니다.
TreeView_SortChildren 트리 뷰 컨트롤에서 지정된 부모 항목의 자식 항목을 정렬합니다. 이 매크로를 사용하거나 TVM_SORTCHILDREN 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SortChildrenCB 항목을 비교하는 애플리케이션 정의 콜백 함수를 사용하여 트리 뷰 항목을 정렬합니다. 이 매크로를 사용하거나 TVM_SORTCHILDRENCB 메시지를 명시적으로 보낼 수 있습니다.
TxActivate 컨트롤이 활성 상태임을 텍스트 호스트에 알립니다.
TxClientToScreen 텍스트 호스트 좌표를 화면 좌표로 변환합니다.
TxCreateCaret 창 없는 서식 있는 편집 컨트롤의 캐리트에 대한 새 셰이프를 만듭니다.
TxDeactivate 이제 컨트롤이 비활성 상태임을 텍스트 호스트에 알립니다.
TxDestroyCaret 캐리트를 제거합니다(Direct2D만 해당).
TxDraw 텍스트 서비스 개체를 그립니다.
TxDrawD2D Direct2D 렌더링을 사용하여 텍스트 서비스 개체를 그립니다.

TxEnableScrollBar 텍스트 호스트 창에서 하나 또는 두 개의 스크롤 막대 화살표를 사용하거나 사용하지 않도록 설정합니다.
TxFreeTextServicesNotification 텍스트 서비스가 해제되었음을 텍스트 호스트에 알립니다.
TxGetAcceleratorPos 밑줄 가속기 문자에 사용할 특수 문자를 요청합니다.
TxGetBackStyle 텍스트 호스트의 배경 스타일을 요청합니다.
TxGetBaseLinePos 텍스트 서비스 클라이언트 사각형을 기준으로 첫 번째 가시선의 기준선 위치를 픽셀 단위로 가져옵니다. 이렇게 하면 기본 줄에서 컨트롤을 정렬할 수 있습니다.
TxGetCachedSize 텍스트 서비스에서 사용하는 캐시된 그리기 논리 크기(있는 경우)를 반환합니다. 일반적으로 ITextServices::TxDraw, ITextServices::OnTxSetCursor 등에서 사용되는 마지막 클라이언트 사각형의 크기입니다.
TxGetCharFormat 텍스트 호스트의 기본 문자 형식을 요청합니다.
TxGetClientRect 텍스트 호스트의 클라이언트 영역의 클라이언트 좌표를 검색합니다.
TxGetCurTargetX 대상 x 위치(즉, caret의 현재 가로 위치)를 가져옵니다.
TxGetDC 텍스트 호스트 창에 대한 디바이스 컨텍스트를 요청합니다.
TxGetDropTarget 텍스트 컨트롤의 놓기 대상을 가져옵니다.

TxGetEastAsianFlags IME(입력 메서드 편집기) 입력이 허용되는지 여부와 편집 스타일에 ES_SELFIME 포함되는지 여부를 가져옵니다.
TxGetEditStyle 서식 있는 편집 컨트롤이 대화 상자에 있는지 여부를 가져옵니다.
TxGetExtent HIMETRIC에서 컨트롤의 네이티브 크기를 요청합니다.
TxGetHorzExtent 텍스트 호스트 창의 가로 스크롤 범위를 가져옵니다.
TxGetHScroll 가로 스크롤 막대 정보를 반환합니다.
TxGetMaxLength 텍스트에 허용되는 텍스트 호스트의 최대 길이를 가져옵니다.
TxGetNaturalSize 컨트롤의 크기를 조정하여 콘텐츠에 맞게 조정할 수 있습니다.
TxGetNaturalSize2 컨트롤의 크기를 조정하여 콘텐츠에 맞게 조정합니다. 이 메서드는 TxGetNaturalSize와 유사하지만 텍스트 위쪽 줄의 상승도 검색합니다.
TxGetPalette 서식 있는 편집 컨트롤의 색상표를 검색합니다.
TxGetParaFormat 텍스트 호스트의 기본 단락 형식을 요청합니다.
TxGetPasswordChar 텍스트 호스트의 암호 문자를 요청합니다.
TxGetPropertyBits 텍스트 호스트에 대한 비트 속성 설정을 요청합니다.

TxGetScrollBars 텍스트 호스트에서 지원하는 스크롤 막대에 대한 정보를 요청합니다.
TxGetSelectionBarWidth HIMETRIC에서 선택 막대의 크기를 반환합니다.
TxGetSysColor 지정된 표시 요소에 대한 텍스트 호스트의 색을 검색합니다.
TxGetText 컨트롤의 모든 유니코드 일반 텍스트를 BSTR로 반환합니다.
TxGetViewInset 텍스트 호스트 창의 텍스트 주위에 있는 공백의 크기를 요청합니다.
TxGetVScroll 세로 스크롤 막대 상태 정보를 반환합니다.
TxGetWindow 서식 있는 편집 컨트롤에 대한 텍스트 호스트 창의 핸들을 검색합니다.
TxGetWindowStyles 텍스트 호스트 창의 창 스타일과 확장된 창 스타일을 검색합니다.
TxImmGetContext 텍스트 서비스 호스트와 연결된 IME(입력 메서드 편집기) 입력 컨텍스트를 검색합니다. 이 메서드는 운영 체제의 아시아어 버전에서만 사용됩니다.
TxImmReleaseContext ITextHost::TxImmGetContext 메서드에서 반환된 입력 컨텍스트를 해제하고 컨텍스트와 연결된 메모리의 잠금을 해제합니다. 이 메서드는 운영 체제의 아시아어 버전에서만 사용됩니다.
TxInvalidateRect 텍스트 호스트 창의 업데이트 영역에 추가할 텍스트 호스트의 사각형을 지정합니다.

TxIsDoubleClickPending 메시지 큐에 텍스트 호스트 창에 대해 보류 중인 WM_LBUTTONDBLCLK 메시지가 포함되어 있는지 여부를 검색합니다.
TxKillTimer 지정된 타이머를 삭제하도록 텍스트 호스트에 요청합니다.
TxNotify 텍스트 호스트에 다양한 이벤트를 알립니다.
TxQueryHitPoint 지정된 점이 텍스트 서비스 개체의 사각형 내에 있는지 여부를 테스트합니다.
TxReleaseDC ITextHost::TxGetDC 메서드에서 가져온 디바이스 컨텍스트를 해제합니다.
TxScreenToClient 화면 좌표를 텍스트 호스트 창 좌표로 변환합니다.
TxScrollWindowEx 텍스트 호스트에 지정된 클라이언트 영역의 콘텐츠를 스크롤하도록 요청합니다.
TxSendMessage 창 호스트에서 창에서 텍스트 서비스 개체로 보낸 메시지를 전달하는 데 사용됩니다.
TxSetCapture 텍스트 호스트의 창에서 마우스 캡처를 설정합니다.
TxSetCaretPos 텍스트 호스트 창에서 지정된 좌표로 캐리트 위치를 이동합니다.
TxSetCursor 텍스트 호스트의 창에 새 커서 셰이프(I-beam)를 설정합니다.
TxSetCursor2 텍스트 호스트 창에서 커서의 모양을 설정합니다.

TxSetFocus 포커스를 텍스트 호스트 창으로 설정합니다.
TxSetForegroundWindow 서식 있는 편집 컨트롤의 호스트 창을 포그라운드 창으로 설정합니다.
TxSetScrollPos 지정된 스크롤 막대에서 스크롤 상자(thumb)의 위치를 설정하고, 요청된 경우 스크롤 막대를 다시 그려 스크롤 상자의 새 위치를 반영합니다. (ITextHost.TxSetScrollPos)
TxSetScrollRange 텍스트 호스트 창에서 지정된 스크롤 막대의 최소 및 최대 위치 값을 설정합니다.
TxSetText 컨트롤의 모든 텍스트를 설정합니다.
TxSetTimer 지정된 시간 제한으로 타이머를 만들도록 텍스트 호스트에 요청합니다.
TxShowCaret 텍스트 호스트 창의 caret 위치에 있는 캐리트를 표시하거나 숨깁니다.
TxShowDropCaret 끌어서 놓기 작업의 놓기 부분(Direct2D에만 해당)을 표시하거나 숨깁니다.
TxShowScrollBar 텍스트 호스트 창에서 스크롤 막대를 표시하거나 숨깁니다.
TxViewChange 업데이트 영역이 변경되었음을 텍스트 호스트에 나타냅니다.
TypeText 이 선택 영역에서 bstr에서 지정한 문자열을 다른 사용자가 입력한 것처럼 입력합니다. 이는 기본 SetText 메서드와 비슷하지만 삽입/덮어쓰기 키 상태 및 자동 고침 및 스마트 따옴표와 같은 UI 설정에 민감합니다.

실행 취소 지정된 수의 실행 취소 작업을 수행합니다.
고정 취소 고정 횟수를 감소합니다.
UnicodeToHex 이 텍스트 범위의 시작 위치 앞에 있는 유니코드 문자를 16진수로 변환하고 선택합니다.
UninitializeFlatSB 특정 창에 대한 플랫 스크롤 막대의 초기화를 해제합니다. 지정된 창은 표준 스크롤 막대로 되돌아갑니다.
업데이트 선택 영역과 캐리트를 업데이트.
UpdatePanningFeedback 패닝 제스처로 인한 창 상태에 대해 클라이언트를 업데이트합니다. 이 함수는 BeginPanningFeedback 호출 후에만 호출할 수 있습니다.
UpdateWindow 보기가 변경되었음을 클라이언트에 알리고 TOM(텍스트 개체 모델) 엔진이 현재 위치로 활성 상태인 경우 클라이언트가 보기를 업데이트해야 합니다.

인터페이스

IImageList 이미지 목록으로 조작하고 상호 작용하는 메서드를 공개합니다.
IImageList2 이미지 목록을 조작하고 상호 작용하기 위한 추가 메서드를 제공하여 IImageList를 확장합니다.

IRichEditOle IRichEditOle 인터페이스는 풍부한 편집 컨트롤의 COM(구성 요소 개체 모델) 기능을 노출합니다. 인터페이스는 EM_GETOLEINTERFACE 메시지를 전송하여 가져올 수 있습니다. 이 인터페이스에는 다음과 같은 메서드가 있습니다.
IRichEditOleCallback IRichEditOleCallback 인터페이스는 클라이언트에서 OLE 관련 정보를 검색하기 위해 서식 있는 텍스트 편집 컨트롤에서 사용됩니다.
IRichEditUiaInformation 창 없는 풍부한 편집 컨트롤에 대한 Microsoft UI 자동화 접근성 정보를 제공합니다.
IRicheditUiaOverrides 창 없는 서식 있는 편집 컨트롤의 호스트 컨테이너가 컨트롤의 Microsoft UI 자동화 접근성 속성을 재정의할 수 있도록 합니다.
IRicheditWindowlessAccessibility 창 없는 서식 있는 편집 컨트롤의 호스트 컨테이너가 컨트롤의 부모에 대한 Microsoft UI 자동화 공급자를 가져올 수 있도록 합니다.
ITextDisplays 이 TOM(텍스트 개체 모델) 엔진 instance 대한 표시 컬렉션을 나타냅니다.
ITextDocument ITextDocument 인터페이스는 TOM(텍스트 개체 모델) 최상위 인터페이스로, 활성 상태인지 여부에 관계없이 문서의 모든 스토리에 대한 활성 선택 영역 및 범위 개체를 검색합니다.
ITextDocument2 ITextDocument 인터페이스를 확장하여 IME(입력 메서드 편집기)가 풍부한 편집 컨트롤을 구동할 수 있도록 하는 메서드와 ITextDisplays, ITextRange2, ITextFont2, ITextPara2 등과 같은 다른 인터페이스를 검색하는 메서드를 추가합니다.
ITextFont TOM(텍스트 개체 모델) 서식 있는 텍스트 범위 특성은 이중 인터페이스 ITextFont 및 ITextPara 쌍을 통해 액세스됩니다. (ITextFont)
ITextFont2 TOM(텍스트 개체 모델)에서 애플리케이션은 이중 인터페이스 ITextFont 및 ITextPara 쌍을 사용하여 텍스트 범위 특성에 액세스합니다.

ITextPara TOM(텍스트 개체 모델) 서식 있는 텍스트 범위 특성은 이중 인터페이스 ITextFont 및 ITextPara 쌍을 통해 액세스됩니다. (ITextPara)
ITextPara2 TOM(텍스트 개체 모델) 서식 있는 텍스트 범위 특성은 이중 인터페이스 ITextFont 및 ITextPara 쌍을 통해 액세스됩니다. (ITextPara2)
ITextRange ITextRange 개체는 프로그램에서 스토리에서 텍스트를 선택한 다음 해당 텍스트를 검사하거나 변경할 수 있는 강력한 편집 및 데이터 바인딩 도구입니다.
ITextRange2 ITextRange2 인터페이스는 ITextRange에서 파생되며, 해당 개체는 프로그램이 스토리에서 텍스트를 선택한 다음 해당 텍스트를 검사하거나 변경할 수 있는 강력한 편집 및 데이터 바인딩 도구입니다.
ITextRow ITextRow 인터페이스는 하나 이상의 동일한 테이블 행을 삽입하고 테이블 행 속성을 검색하고 변경하는 메서드를 제공합니다. 비인간적 행을 삽입하려면 각 행 구성에 대해 ITextRow::Insert를 호출합니다.
ITextSelection 텍스트 선택 영역은 선택 영역이 강조 표시된 텍스트 범위입니다.
ITextSelection2 현재 이 인터페이스에는 ITextRange2에서 상속된 메서드 이외의 메서드가 없습니다.
ITextStory ITextStory 인터페이스 메서드는 부모 ITextServices instance 저장된 여러 스토리에서 공유 데이터에 액세스하는 데 사용됩니다.
ITextStoryRanges ITextStoryRanges 인터페이스의 목적은 ITextDocument의 스토리를 열거하는 것입니다.
ITextStoryRanges2 ITextStoryRanges2 인터페이스는 ITextDocument의 스토리를 열거합니다.

ITextStrings ITextStrings 인터페이스는 서식 있는 텍스트를 조작하는 데 유용한 서식 있는 텍스트 문자열의 컬렉션을 나타냅니다.

구조체

BIDIOPTIONS 서식 있는 편집 컨트롤에 대한 양방향 정보를 포함합니다. 이 구조체는 컨트롤에 대한 양방향 정보를 가져오고 설정하기 위해 EM_GETBIDIOPTIONS 및 EM_SETBIDIOPTIONS 메시지에서 사용됩니다.
BP_ANIMATIONPARAMS BeginBufferedPaint에서 사용하는 BP_PAINTPARAMS 구조체에 대한 애니메이션 매개 변수를 정의합니다.
BP_PAINTPARAMS BeginBufferedPaint에 대한 페인트 작업 매개 변수를 정의합니다.
BUTTON_IMAGELIST 단추 컨트롤과 함께 사용되는 이미지 목록에 대한 정보를 포함합니다.
BUTTON_SPLITINFO 분할 단추(BS_SPLITBUTTON 및 BS_DEFSPLITBUTTON 스타일)을 정의하는 정보를 포함합니다. BCM_GETSPLITINFO 및 BCM_SETSPLITINFO 메시지와 함께 사용됩니다.
CHANGENOTIFY EN_CHANGE 알림 코드와 연결된 정보를 포함합니다. 창이 없는 서식 있는 편집 컨트롤은 컨트롤의 콘텐츠가 변경되면 호스트 창으로 이 알림을 보냅니다.
CHARFORMAT2A ANSI(CHARFORMAT2A) 구조체(richedit.h)에는 서식 있는 편집 컨트롤의 문자 서식에 대한 정보가 포함되어 있습니다.
CHARFORMAT2A 서식 있는 편집 컨트롤의 문자 서식에 대한 정보를 포함합니다. (CHARFORMAT2A)

CHARFORMAT2W CHARFORMAT2W(유니코드) 구조체(richedit.h)에는 서식 있는 편집 컨트롤의 문자 서식에 대한 정보가 포함되어 있습니다.
CHARFORMAT2W 서식 있는 편집 컨트롤의 문자 서식에 대한 정보를 포함합니다. (CHARFORMAT2W)
CHARFORMATA 서식 있는 편집 컨트롤의 문자 서식에 대한 정보를 포함합니다. (CHARFORMATA)
CHARFORMATW 서식 있는 편집 컨트롤의 문자 서식에 대한 정보를 포함합니다. (CHARFORMATW)
CHARRANGE 서식 있는 편집 컨트롤의 문자 범위를 지정합니다. cpMin 및 cpMax 멤버가 같으면 범위가 비어 있습니다. cpMin이 0이고 cpMax가 1인 경우 범위에 모든 항목이 포함됩니다.
CLIPBOARDFORMAT 클립보드 형식을 지정합니다. 이 구조는 EN_CLIPFORMAT 알림에 포함됩니다.
COLORMAP 비트맵의 색을 매핑하기 위해 CreateMappedBitmap 함수에서 사용하는 정보를 포함합니다.
COLORSCHEME 도구 모음 또는 Rebar의 단추 그리기에 대한 정보를 포함합니다.
COMBOBOXEXITEMA ComboBoxEx 컨트롤의 항목에 대한 정보를 포함합니다. (ANSI)
COMBOBOXEXITEMW ComboBoxEx 컨트롤의 항목에 대한 정보를 포함합니다. (유니코드)
COMBOBOXINFO 콤보 상자 상태 정보를 포함합니다.

COMPAREITEMSTRUCT 소유자가 그린 정렬된 목록 상자 또는 콤보 상자에서 두 항목에 대한 식별자 및 애플리케이션 제공 데이터를 제공합니다.
COMPCOLOR 구성 문자열에 대한 색 설정을 포함합니다.
DATETIMEPICKERINFO DTP(날짜 및 시간 선택기) 컨트롤에 대한 정보를 포함합니다.
DELETEITEMSTRUCT 삭제된 목록 상자 또는 콤보 상자 항목을 설명합니다.
DPASTREAMINFO PFNDPASTREAM 콜백 함수에서 사용하는 스트림 항목을 포함합니다.
DRAGLISTINFO 끌기 이벤트에 대한 정보를 포함합니다. DRAGLISTINFO를 가리키는 포인터는 끌기 목록 메시지의 lParam 매개 변수로 전달됩니다.
DRAWITEMSTRUCT 소유자 창에서 소유자 그리기 컨트롤 또는 메뉴 항목을 그리는 방법을 결정하는 데 사용하는 정보를 제공합니다.
DTBGOPTS DrawThemeBackgroundEx 함수에 대한 옵션을 정의합니다.
DTTOPTS DrawThemeTextEx 함수에 대한 옵션을 정의합니다.
EDITBALLOONTIP 단추 컨트롤과 연결된 풍선 팁에 대한 정보를 포함합니다.
EDITSTREAM 애플리케이션이 EM_STREAMIN 또는 EM_STREAMOUT 메시지의 서식 있는 편집 컨트롤에 전달하는 정보를 포함합니다. 서식 있는 편집 컨트롤은 이 정보를 사용하여 데이터 스트림을 컨트롤 안팎으로 전송합니다.

ENCORRECTTEXT 수정하기 위해 선택한 텍스트에 대한 정보를 포함합니다.
ENDCOMPOSITIONNOTIFY 다양한 편집 컨트롤의 EN_ENDCOMPOSITION 알림 코드에 대한 정보를 포함합니다.
ENDROPFILES EN_DROPFILES 알림 코드와 관련된 정보를 포함합니다. 서식 있는 편집 컨트롤은 WM_DROPFILES 메시지를 수신할 때 이 알림 코드를 보냅니다.
ENLINK 서식 있는 편집 컨트롤의 EN_LINK 알림 코드에 대한 정보를 포함합니다.
ENLOWFIRTF Microsoft Rich Edit 컨트롤의 지원되지 않는 RTF(서식 있는 텍스트 형식) 키워드(keyword) 대한 정보를 포함합니다.
ENOLEOPFAILED 실패한 작업에 대한 정보를 포함합니다.
ENPROTECTED EN_PROTECTED 알림 코드와 관련된 정보를 포함합니다. 서식 있는 편집 컨트롤은 사용자가 보호된 텍스트를 편집하려고 할 때 이 알림을 보냅니다.
ENSAVECLIPBOARD 클립보드의 개체 및 텍스트에 대한 정보를 포함합니다.
FINDTEXTA 서식 있는 편집 컨트롤의 검색 작업에 대한 정보를 포함합니다. 이 구조체는 EM_FINDTEXT 메시지와 함께 사용됩니다. (ANSI)
FINDTEXTEXA 서식 있는 편집 컨트롤에서 검색할 텍스트에 대한 정보를 포함합니다. 이 구조체는 EM_FINDTEXTEX 메시지와 함께 사용됩니다. (ANSI)
FINDTEXTEXW 서식 있는 편집 컨트롤에서 검색할 텍스트에 대한 정보를 포함합니다. 이 구조체는 EM_FINDTEXTEX 메시지와 함께 사용됩니다. (유니코드)

FINDTEXTW 서식 있는 편집 컨트롤의 검색 작업에 대한 정보를 포함합니다. 이 구조체는 EM_FINDTEXT 메시지와 함께 사용됩니다. (유니코드)
FORMATRANGE 서식 있는 편집 컨트롤이 특정 디바이스의 출력 형식을 지정하는 데 사용하는 정보입니다. 이 구조체는 EM_FORMATRANGE 메시지와 함께 사용됩니다.
GETCONTEXTMENUEX IRichEditOleCallback::GetContextMenu 메서드에 전달되는 상황에 맞는 메뉴 정보를 포함합니다.
GETTEXTEX 서식 있는 편집 컨트롤에서 텍스트를 가져오는 데 사용되는 정보를 포함합니다. 이 구조체는 EM_GETTEXTEX 메시지와 함께 사용됩니다.
GETTEXTLENGTHEX 서식 있는 편집 컨트롤의 텍스트 길이를 계산하는 방법에 대한 정보를 포함합니다. 이 구조체는 EM_GETTEXTLENGTHEX 메시지의 wParam에 전달됩니다.
HD_TEXTFILTERA 헤더 컨트롤 텍스트 필터에 대한 정보를 포함합니다. (ANSI)
HD_TEXTFILTERW 헤더 컨트롤 텍스트 필터에 대한 정보를 포함합니다. (유니코드)
HDHITTESTINFO 적중 테스트에 대한 정보를 포함합니다. 이 구조체는 HDM_HITTEST 메시지와 함께 사용되며 HD_HITTESTINFO 구조를 대체합니다.
HDITEMA 헤더 컨트롤의 항목에 대한 정보를 포함합니다. 이 구조체는 HD_ITEM 구조를 대체합니다. (ANSI)
HDITEMW 헤더 컨트롤의 항목에 대한 정보를 포함합니다. 이 구조체는 HD_ITEM 구조를 대체합니다. (유니코드)

HDLAYOUT 헤더 컨트롤의 크기와 위치를 설정하는 데 사용되는 정보를 포함합니다. HDLAYOUT은 HDM_LAYOUT 메시지와 함께 사용됩니다. 이 구조체는 HD_LAYOUT 구조를 대체합니다.
HYPHENATEINFO Microsoft Rich Edit 컨트롤의 하이픈 넣기에 대한 정보를 포함합니다.
HYPHRESULT Microsoft Rich Edit 컨트롤의 하이픈 넣기 결과에 대한 정보를 포함합니다.
IMAGEINFO IMAGEINFO 구조체에는 이미지 목록의 이미지에 대한 정보가 포함됩니다. 이 구조체는 IImageList::GetImageInfo 함수와 함께 사용됩니다.
IMAGEINFO IMAGEINFO 구조체는 이미지 목록의 이미지에 대한 정보를 포함하며 IImageList::GetImageInfo 함수와 함께 사용됩니다.
IMAGELISTDRAWPARAMS IMAGELISTDRAWPARAMS 구조체는 이미지 목록 그리기 작업에 대한 정보를 포함하며 IImageList::D raw 함수와 함께 사용됩니다.
IMAGELISTDRAWPARAMS IMAGELISTDRAWPARAMS 구조체에는 이미지 목록 그리기 작업에 대한 정보가 포함되어 있습니다. 이 구조체는 IImageList::D raw 함수와 함께 사용됩니다.
IMAGELISTSTATS 이미지 목록 통계를 포함합니다. GetStatistics에서 사용됩니다.
IMECOMPTEXT Microsoft Rich Edit 컨트롤의 IME(입력 메서드 편집기) 컴퍼지션 텍스트에 대한 정보를 포함합니다.
INITCOMMONCONTROLSEX DLL(동적 연결 라이브러리)에서 공용 컨트롤 클래스를 로드하는 데 사용되는 정보를 전달합니다. 이 구조체는 InitCommonControlsEx 함수와 함께 사용됩니다.
INPUT_INJECTION_VALUE 입력 삽입 세부 정보를 포함합니다.

INPUT_MESSAGE_SOURCE 입력 메시지의 원본에 대한 정보를 포함합니다.
INTLIST 시각적 스타일의 int 데이터 항목 목록 또는 배열을 포함합니다.
LHITTESTINFO 지정된 위치에 해당하는 링크 관련 정보를 가져오는 데 사용됩니다.
LITEM 링크 항목에 대한 정보를 설정하고 검색하는 데 사용됩니다.
LVBKIMAGEA 목록 뷰 컨트롤의 배경 이미지에 대한 정보를 포함합니다. 이 구조체는 배경 이미지 정보를 설정하고 검색하는 데 사용됩니다. (ANSI)
LVBKIMAGEW 목록 뷰 컨트롤의 배경 이미지에 대한 정보를 포함합니다. 이 구조체는 배경 이미지 정보를 설정하고 검색하는 데 사용됩니다. (유니코드)
LVCOLUMNA 보고서 뷰의 열에 대한 정보를 포함합니다. 이 구조체는 열을 만들고 조작하는 데 모두 사용됩니다. 이 구조체는 LV_COLUMN 구조체를 대체합니다. (ANSI)
LVCOLUMNW 보고서 뷰의 열에 대한 정보를 포함합니다. 이 구조체는 열을 만들고 조작하는 데 모두 사용됩니다. 이 구조체는 LV_COLUMN 구조체를 대체합니다. (유니코드)
LVFINDINFOA 목록 뷰 항목을 검색할 때 사용되는 정보를 포함합니다. 이 구조체는 LV_FINDINFO와 동일하지만 표준 명명 규칙에 맞게 이름이 변경되었습니다. (ANSI)
LVFINDINFOW 목록 뷰 항목을 검색할 때 사용되는 정보를 포함합니다. 이 구조체는 LV_FINDINFO와 동일하지만 표준 명명 규칙에 맞게 이름이 변경되었습니다. (유니코드)
LVFOOTERINFO 목록 뷰 컨트롤의 바닥글에 대한 정보를 포함합니다.

LVFOOTERITEM 바닥글 항목에 대한 정보를 포함합니다.
LVGROUP 그룹을 설정하고 검색하는 데 사용됩니다.
LVGROUPMETRICS 목록 뷰 컨트롤의 그룹 표시에 대한 정보를 포함합니다.
LVHITTESTINFO 적중 테스트에 대한 정보를 포함합니다.
LVINSERTGROUPSORTED 그룹을 정렬하는 데 사용됩니다. LVM_INSERTGROUPSORTED 와함께 사용됩니다.
LVINSERTMARK 삽입 지점을 설명하는 데 사용됩니다.
LVITEMA 목록 뷰 항목의 특성을 지정하거나 받습니다. 이 구조체는 항목 들여쓰기를 사용하도록 설정하는 새 마스크 값(LVIF_INDENT)을 지원하도록 업데이트되었습니다. 이 구조체는 LV_ITEM 구조체를 대체합니다. (ANSI)
LVITEMINDEX 목록 뷰 항목에 대한 인덱스 정보를 포함합니다.
LVITEMW 목록 뷰 항목의 특성을 지정하거나 받습니다. 이 구조체는 항목 들여쓰기를 사용하도록 설정하는 새 마스크 값(LVIF_INDENT)을 지원하도록 업데이트되었습니다. 이 구조체는 LV_ITEM 구조체를 대체합니다. (유니코드)
LVSETINFOTIP 설정할 도구 설명 텍스트에 대한 정보를 제공합니다.
LVTILEINFO 타일 뷰에 표시될 때 목록 뷰 컨트롤의 항목에 대한 정보를 제공합니다.

LVTILEVIEWINFO 타일 뷰에 표시될 때 목록 뷰 컨트롤에 대한 정보를 제공합니다.
MARGINS 시각적 스타일이 적용된 창의 여백을 정의하기 위해 GetThemeMargins 함수에 의해 반환됩니다.
MCGRIDINFO 달력 컨트롤의 일부에 대한 정보를 포함합니다.
MCHITTESTINFO 월 달력 컨트롤에 대한 적중 횟수 테스트 지점과 관련된 정보를 전달합니다. 이 구조체는 MCM_HITTEST 메시지 및 해당 MonthCal_HitTest 매크로와 함께 사용됩니다.
MEASUREITEMSTRUCT 소유자가 그린 컨트롤 또는 메뉴 항목의 차원을 시스템에 알립니다. 이렇게 하면 시스템이 컨트롤과 사용자의 상호 작용을 올바르게 처리할 수 있습니다.
MSGFILTER 키보드 또는 마우스 이벤트에 대한 정보를 포함합니다. 서식 있는 편집 컨트롤은 이 구조체를 EN_MSGFILTER 알림 코드의 일부로 부모 창에 전송하여 부모 창에서 메시지를 변경하거나 처리되지 않도록 할 수 있습니다.
NMBCDROPDOWN BCN_DROPDOWN 알림에 대한 정보를 포함합니다.
NMBCHOTITEM 단추 컨트롤 위의 마우스 놓기에 대한 정보를 포함합니다.
NMCBEDRAGBEGINA CBEN_DRAGBEGIN 알림 코드와 함께 사용되는 정보를 포함합니다. (ANSI)
NMCBEDRAGBEGINW CBEN_DRAGBEGIN 알림 코드와 함께 사용되는 정보를 포함합니다. (유니코드)
NMCBEENDEDITA ComboBoxEx 컨트롤 내에서 편집 작업의 결론에 대한 정보를 포함합니다. 이 구조체는 CBEN_ENDEDIT 알림 코드와 함께 사용됩니다. (ANSI)

NMCBEENDEDITW ComboBoxEx 컨트롤 내에서 편집 작업의 결론에 대한 정보를 포함합니다. 이 구조체는 CBEN_ENDEDIT 알림 코드와 함께 사용됩니다. (유니코드)
NMCHAR 문자 알림 메시지에 사용되는 정보를 포함합니다.
NMCOMBOBOXEXA 알림 코드와 함께 사용할 ComboBoxEx 항목에 특정한 정보를 포함합니다. (ANSI)
NMCOMBOBOXEXW 알림 코드와 함께 사용할 ComboBoxEx 항목에 특정한 정보를 포함합니다. (유니코드)
NMCUSTOMDRAW NM_CUSTOMDRAW 알림 코드에 특정한 정보를 포함합니다.
NMCUSTOMSPLITRECTINFO 분할 단추의 두 사각형에 대한 정보를 포함합니다. NM_GETCUSTOMSPLITRECT 알림과 함께 전송됩니다.
NMCUSTOMTEXT 사용자 지정 텍스트 알림에 사용되는 정보를 포함합니다.
NMDATETIMECHANGE DTP(날짜 및 시간 선택기) 컨트롤에서 발생한 변경 내용에 대한 정보가 포함되어 있습니다. 이 구조체는 DTN_DATETIMECHANGE 알림 코드와 함께 사용됩니다.
NMDATETIMEFORMATA DTP(날짜 및 시간 선택기) 컨트롤 내에서 콜백 필드를 정의하는 형식 문자열의 일부에 대한 정보가 포함되어 있습니다. (ANSI)
NMDATETIMEFORMATQUERYA DTP(날짜 및 시간 선택기) 컨트롤 콜백 필드에 대한 정보가 포함되어 있습니다. (ANSI)
NMDATETIMEFORMATQUERYW DTP(날짜 및 시간 선택기) 컨트롤 콜백 필드에 대한 정보가 포함되어 있습니다. (유니코드)

NMDATETIMEFORMATW DTP(날짜 및 시간 선택기) 컨트롤 내에서 콜백 필드를 정의하는 형식 문자열의 일부에 대한 정보가 포함되어 있습니다. (유니코드)
NMDATETIMESTRINGA DTP(날짜 및 시간 선택기) 컨트롤에서 수행된 편집 작업과 관련된 정보가 포함되어 있습니다. 이 메시지는 DTN_USERSTRING 알림 코드와 함께 사용됩니다. (ANSI)
NMDATETIMESTRINGW DTP(날짜 및 시간 선택기) 컨트롤에서 수행된 편집 작업과 관련된 정보가 포함되어 있습니다. 이 메시지는 DTN_USERSTRING 알림 코드와 함께 사용됩니다. (유니코드)
NMDATETIMEWMKEYDOWNA DTN_WMKEYDOWN 알림 코드를 설명하고 처리하는 데 사용되는 정보를 전달합니다. (ANSI)
NMDATETIMEWMKEYDOWNW DTN_WMKEYDOWN 알림 코드를 설명하고 처리하는 데 사용되는 정보를 전달합니다. (유니코드)
NMDAYSTATE MCN_GETDAYSTATE 알림 코드를 처리하는 데 필요한 정보를 전달합니다. 이 구조체의 모든 멤버는 MCN_GETDAYSTATE를 처리할 때 수신 애플리케이션이 설정해야 하는 prgDayState를 제외하고 입력용입니다.
NMHDDISPINFOA HDN_GETDISPINFO 알림 코드를 처리하는 데 사용되는 정보를 포함합니다. (ANSI)
NMHDDISPINFOW HDN_GETDISPINFO 알림 코드를 처리하는 데 사용되는 정보를 포함합니다. (유니코드)
NMHDFILTERBTNCLICK 필터 단추 클릭의 특성을 지정하거나 받습니다.
NMHDR NMHDR 구조에는 알림 메시지에 대한 정보가 포함됩니다. (NMHDR 구조체)
NMHDR NMHDR(richedit.h) 구조체에는 알림 메시지에 대한 정보가 포함됩니다.

NMHEADERA 헤더 컨트롤 알림 메시지에 대한 정보를 포함합니다. 이 구조체는 HD_NOTIFY 구조를 대체합니다. (ANSI)
NMHEADERW 헤더 컨트롤 알림 메시지에 대한 정보를 포함합니다. 이 구조체는 HD_NOTIFY 구조를 대체합니다. (유니코드)
NMIPADDRESS IPN_FIELDCHANGED 알림 코드의 정보를 포함합니다.
NMITEMACTIVATE LVN_ITEMACTIVATE 알림 코드에 대한 정보를 포함합니다.
NMKEY 주요 알림 메시지와 함께 사용되는 정보를 포함합니다.
NMLINK NMLINK에는 알림 정보가 포함됩니다. NM_CLICK 또는 NM_RETURN 메시지와 함께 이 구조체를 보냅니다.
NMLISTVIEW 목록 보기 알림 메시지에 대한 정보를 포함합니다. 이 구조체는 NM_LISTVIEW 구조체와 동일하지만 표준 명명 규칙에 맞게 이름이 변경되었습니다.
NMLVCACHEHINT 가상 목록 뷰와 함께 사용할 캐시된 항목 정보를 업데이트하는 데 사용되는 정보를 포함합니다.
NMLVCUSTOMDRAW 목록 뷰 컨트롤에서 보낸 NM_CUSTOMDRAW(목록 뷰) 알림 코드와 관련된 정보를 포함합니다.
NMLVDISPINFOA LVN_GETDISPINFO 또는 LVN_SETDISPINFO 알림 코드에 대한 정보를 포함합니다. 이 구조체는 LV_DISPINFO 구조체와 동일하지만 표준 명명 규칙에 맞게 이름이 변경되었습니다. (ANSI)
NMLVDISPINFOW LVN_GETDISPINFO 또는 LVN_SETDISPINFO 알림 코드에 대한 정보를 포함합니다. 이 구조체는 LV_DISPINFO 구조체와 동일하지만 표준 명명 규칙에 맞게 이름이 변경되었습니다. (유니코드)

NMLVEMPTYMARKUP LVN_GETEMPTYMARKUP 알림 코드와 함께 사용되는 정보를 포함합니다.
NMLVFINDITEMA 소유자가 가상 목록 보기 컨트롤에서 요청한 항목을 찾는 데 필요한 정보를 포함합니다. 이 구조체는 LVN_ODFINDITEM 알림 코드와 함께 사용됩니다. (ANSI)
NMLVFINDITEMW 소유자가 가상 목록 보기 컨트롤에서 요청한 항목을 찾는 데 필요한 정보를 포함합니다. 이 구조체는 LVN_ODFINDITEM 알림 코드와 함께 사용됩니다. (유니코드)
NMLVGETINFOTIPA 항목에 대한 도구 설명을 표시하는 데 필요한 목록 뷰 항목 정보를 포함하고 받습니다. 이 구조체는 LVN_GETINFOTIP 알림 코드와 함께 사용됩니다. (ANSI)
NMLVGETINFOTIPW 항목에 대한 도구 설명을 표시하는 데 필요한 목록 뷰 항목 정보를 포함하고 받습니다. 이 구조체는 LVN_GETINFOTIP 알림 코드와 함께 사용됩니다. (유니코드)
NMLVKEYDOWN LVN_KEYDOWN 알림 코드를 처리하는 데 사용되는 정보를 포함합니다. 이 구조체는 NMLVKEYDOWN 구조체와 동일하지만 표준 명명 규칙에 맞게 이름이 변경되었습니다.
NMLVLINK LVN_LINKCLICK 알림 코드에 대한 정보를 포함합니다.
NMLVODSTATECHANGE LVN_ODSTATECHANGED 알림 코드를 처리하는 데 사용할 정보를 포함하는 구조체입니다.
NMLVSCROLL 스크롤 작업에 대한 정보를 제공합니다.
NMMOUSE 마우스 알림 메시지에 사용되는 정보를 포함합니다.
NMOBJECTNOTIFY TBN_GETOBJECT, TCN_GETOBJECT 및 PSN_GETOBJECT 알림 코드와 함께 사용되는 정보를 포함합니다.

NMPGCALCSIZE Pager 컨트롤이 포함된 창의 스크롤 가능 영역을 계산하는 데 사용하는 정보를 포함하고 받습니다. PGN_CALCSIZE 알림과 함께 사용됩니다.
NMPGHOTITEM PGN_HOTITEMCHANGE 알림 코드와 함께 사용되는 정보를 포함합니다.
NMPGSCROLL 포함된 창을 스크롤할 때 Pager 컨트롤이 사용하는 정보를 포함하고 받습니다. PGN_SCROLL 알림과 함께 사용됩니다.
NMRBAUTOSIZE RBN_AUTOSIZE 알림 코드를 처리하는 데 사용되는 정보를 포함합니다.
NMREBAR 다양한 철근 알림을 처리하는 데 사용되는 정보를 포함합니다.
NMREBARAUTOBREAK RBN_AUTOBREAK 알림 코드와 함께 사용되는 정보를 포함합니다.
NMREBARCHEVRON RBN_CHEVRONPUSHED 알림 코드를 처리하는 데 사용되는 정보를 포함합니다.
NMREBARCHILDSIZE RBN_CHILDSIZE 알림 코드를 처리하는 데 사용되는 정보를 포함합니다.
NMREBARSPLITTER RBN_SPLITTERDRAG 알림 코드를 처리하는 데 사용되는 정보를 포함합니다.
NMSEARCHWEB EN_SEARCHWEB 알림 코드를 처리하는 데 사용되는 정보를 포함합니다.
NMSELCHANGE MCN_SELCHANGE 알림 코드를 처리하는 데 필요한 정보를 전달합니다.
NMTBCUSTOMDRAW 도구 모음 컨트롤에서 보낸 NM_CUSTOMDRAW 알림 코드와 관련된 정보를 포함합니다.

NMTBDISPINFOA 도구 모음 항목의 표시 정보를 포함하고 받습니다. 이 구조체는 TBN_GETDISPINFO 알림 코드와 함께 사용됩니다. (ANSI)
NMTBDISPINFOW 도구 모음 항목의 표시 정보를 포함하고 받습니다. 이 구조체는 TBN_GETDISPINFO 알림 코드와 함께 사용됩니다. (유니코드)
NMTBGETINFOTIPA 도구 모음 항목의 정보 팁 정보를 포함하고 받습니다. 이 구조체는 TBN_GETINFOTIP 알림 코드와 함께 사용됩니다. (ANSI)
NMTBGETINFOTIPW 도구 모음 항목의 정보 팁 정보를 포함하고 받습니다. 이 구조체는 TBN_GETINFOTIP 알림 코드와 함께 사용됩니다. (유니코드)
NMTBHOTITEM TBN_HOTITEMCHANGE 알림 코드와 함께 사용되는 정보를 포함합니다.
NMTBRESTORE 도구 모음 상태가 저장되었을 때 애플리케이션이 NMTBSAVE에 배치된 정보를 추출할 수 있습니다. TBN_RESTORE 알림 코드를 받으면 이 구조체가 애플리케이션에 전달됩니다.
NMTBSAVE TBN_SAVE 알림 코드를 받으면 이 구조체가 애플리케이션에 전달됩니다. 여기에는 현재 저장 중인 단추에 대한 정보가 있습니다. 애플리케이션은 멤버 값을 수정하여 추가 정보를 저장할 수 있습니다.
NMTCKEYDOWN 탭 컨트롤의 키 누름에 대한 정보를 포함합니다. TCN_KEYDOWN 알림 코드와 함께 사용됩니다. 이 구조체는 TC_KEYDOWN 구조체를 대체합니다.
NMTOOLBARA 도구 모음 알림 코드를 처리하는 데 사용되는 정보를 포함합니다. 이 구조체는 TBNOTIFY 구조체를 대체합니다. (ANSI)
NMTOOLBARW 도구 모음 알림 코드를 처리하는 데 사용되는 정보를 포함합니다. 이 구조체는 TBNOTIFY 구조체를 대체합니다. (유니코드)

NMTOOLTIPSCREATED NM_TOOLTIPSCREATED 알림 코드와 함께 사용되는 정보를 포함합니다.
NMTRBTHUMBPOSCHANGING 트랙바 변경 알림에 대한 정보를 포함합니다. 이 메시지는 TRBN_THUMBPOSCHANGING 알림과 함께 전송됩니다.
NMTREEVIEWA 트리 뷰 알림 메시지에 대한 정보를 포함합니다. 이 구조체는 NM_TREEVIEW 구조체와 동일하지만 현재 명명 규칙을 따르도록 이름이 변경되었습니다. (ANSI)
NMTREEVIEWW 트리 뷰 알림 메시지에 대한 정보를 포함합니다. 이 구조체는 NM_TREEVIEW 구조체와 동일하지만 현재 명명 규칙을 따르도록 이름이 변경되었습니다. (유니코드)
NMTTCUSTOMDRAW 도구 설명 컨트롤에서 보낸 NM_CUSTOMDRAW 알림 코드에 특정한 정보를 포함합니다.
NMTTDISPINFOA TTN_GETDISPINFO 알림 코드를 처리하는 데 사용되는 정보를 포함합니다. 이 구조체는 TOOLTIPTEXT 구조체를 대체합니다. (ANSI)
NMTTDISPINFOW TTN_GETDISPINFO 알림 코드를 처리하는 데 사용되는 정보를 포함합니다. 이 구조체는 TOOLTIPTEXT 구조체를 대체합니다. (유니코드)
NMTVASYNCDRAW 아이콘 또는 오버레이 트리 항목 그리기 실패 이유에 대한 설명이 포함되어 있습니다.
NMTVCUSTOMDRAW 트리 뷰 컨트롤에서 보낸 NM_CUSTOMDRAW(트리 뷰) 알림 코드와 관련된 정보를 포함합니다.
NMTVDISPINFOA 트리 뷰 항목에 대한 표시 정보를 포함하고 받습니다. 이 구조체는 TV_DISPINFO 구조체와 동일하지만 현재 명명 규칙을 따르도록 이름이 변경되었습니다. (ANSI)
NMTVDISPINFOEXA 확장된 TreeView 알림 정보와 관련된 정보를 포함합니다. (ANSI)

NMTVDISPINFOEXW 확장된 TreeView 알림 정보와 관련된 정보를 포함합니다. (유니코드)
NMTVDISPINFOW 트리 뷰 항목에 대한 표시 정보를 포함하고 받습니다. 이 구조체는 TV_DISPINFO 구조체와 동일하지만 현재 명명 규칙을 따르도록 이름이 변경되었습니다. (유니코드)
NMTVGETINFOTIPA 항목에 대한 도구 설명을 표시하는 데 필요한 트리 뷰 항목 정보를 포함하고 받습니다. 이 구조체는 TVN_GETINFOTIP 알림 코드와 함께 사용됩니다. (ANSI)
NMTVGETINFOTIPW 항목에 대한 도구 설명을 표시하는 데 필요한 트리 뷰 항목 정보를 포함하고 받습니다. 이 구조체는 TVN_GETINFOTIP 알림 코드와 함께 사용됩니다. (유니코드)
NMTVITEMCHANGE 트리 뷰 항목 변경에 대한 정보를 포함합니다. 이 구조체는 TVN_ITEMCHANGED 및 TVN_ITEMCHANGING 알림과 함께 전송됩니다.
NMTVKEYDOWN 트리 뷰 컨트롤의 키보드 이벤트에 대한 정보를 포함합니다. 이 구조체는 TVN_KEYDOWN 알림 코드와 함께 사용됩니다. 구조체는 TV_KEYDOWN 구조체와 동일하지만 현재 명명 규칙을 따르도록 이름이 변경되었습니다.
NMTVSTATEIMAGECHANGING NM_TVSTATEIMAGECHANGING 알림 코드에 대한 정보를 포함합니다.
NMUPDOWN 업다운 제어 알림 메시지와 관련된 정보를 포함합니다. NM_UPDOWN 구조체와 동일하며 이를 대체합니다.
NMVIEWCHANGE MCN_VIEWCHANGE 알림 코드를 처리하는 데 필요한 정보를 저장합니다.
개체 위치 개체 위치 정보를 포함합니다.

PARAFORMAT 서식 있는 편집 컨트롤의 단락 서식 특성에 대한 정보를 포함합니다. (PARAFORMAT)
PARAFORMAT2 서식 있는 편집 컨트롤의 단락 서식 특성에 대한 정보를 포함합니다. P
PARAFORMAT2 서식 있는 편집 컨트롤의 단락 서식 특성에 대한 정보를 포함합니다. (PARAFORMAT2)
PBRANGE 진행률 표시줄 컨트롤의 상한 및 하한에 대한 정보를 포함합니다. 이 구조체는 PBM_GETRANGE 메시지와 함께 사용됩니다.
POINTER_DEVICE_CURSOR_INFO 포인터 디바이스에 대한 커서 ID 매핑을 포함합니다.
POINTER_DEVICE_INFO 포인터 디바이스에 대한 정보를 포함합니다. 이러한 구조체의 배열은 GetPointerDevices 함수에서 반환됩니다. GetPointerDevice 함수에 대한 호출에서 단일 구조체가 반환됩니다.
POINTER_DEVICE_PROPERTY 포인터 기반 디바이스 속성(HID 사용량에 해당하는 HID(휴먼 인터페이스 디바이스) 전역 항목)을 포함합니다.
POINTER_TYPE_INFO 포인터 입력 형식에 대한 정보를 포함합니다.
PROPSHEETHEADERA_V2 ANSI(PROPSHEETHEADERA_V2) 구조체는 속성 시트의 프레임과 페이지를 정의합니다.
PROPSHEETHEADERW_V2 PROPSHEETHEADERW_V2(유니코드) 구조체는 속성 시트의 프레임과 페이지를 정의합니다.
PROPSHEETPAGEA_V2 ANSI(PROPSHEETPAGEA_V2) 구조체는 속성 시트의 페이지를 정의합니다.

PROPSHEETPAGEA_V3 ANSI(PROPSHEETPAGEA_V3) 구조체는 속성 시트의 페이지를 정의합니다.
PROPSHEETPAGEA_V4 ANSI(PROPSHEETPAGEA_V4) 구조체는 속성 시트의 페이지를 정의합니다.
PROPSHEETPAGEW_V2 PROPSHEETPAGEW_V2(유니코드) 구조체는 속성 시트의 페이지를 정의합니다.
PROPSHEETPAGEW_V3 PROPSHEETPAGEW_V3(유니코드) 구조체는 속성 시트의 페이지를 정의합니다.
PROPSHEETPAGEW_V4 PROPSHEETPAGEW_V4(유니코드) 구조체는 속성 시트의 페이지를 정의합니다.
PSHNOTIFY 속성 시트 알림 메시지에 대한 정보를 포함합니다.
PUNCTUATION 서식 있는 편집 컨트롤에 사용되는 문장 부호에 대한 정보를 포함합니다.
RBHITTESTINFO 적중 테스트 작업과 관련된 정보를 포함합니다. 이 구조체는 RB_HITTEST 메시지와 함께 사용됩니다.
REBARBANDINFOA rebar 컨트롤에서 밴드를 정의하는 정보를 포함합니다. (ANSI)
REBARBANDINFOW rebar 컨트롤에서 밴드를 정의하는 정보를 포함합니다. (유니코드)
REBARINFO rebar 컨트롤 특성을 설명하는 정보를 포함합니다.
REOBJECT 풍부한 편집 컨트롤의 OLE 또는 이미지 개체에 대한 정보를 포함합니다.

REPASTESPECIAL 붙여넣은 개체의 표시 측면이 개체의 콘텐츠 또는 개체를 나타내는 아이콘을 기반으로 해야 하는지 여부를 식별하는 정보를 포함합니다.
REQRESIZE 서식 있는 편집 컨트롤의 요청된 크기를 포함합니다. 서식 있는 편집 컨트롤은 이 구조체를 EN_REQUESTRESIZE 알림 코드의 일부로 부모 창에 보냅니다.
RICHEDIT_IMAGE_PARAMETERS EM_INSERTIMAGE 메시지에서 삽입할 이미지의 특성을 정의합니다.
SCROLLBARINFO SCROLLBARINFO 구조체에는 스크롤 막대 정보가 포함되어 있습니다.
SCROLLINFO SCROLLINFO 구조체에는 SetScrollInfo 함수(또는 SBM_SETSCROLLINFO 메시지)가 설정하거나 GetScrollInfo 함수(또는 SBM_GETSCROLLINFO 메시지)가 검색한 스크롤 막대 매개 변수가 포함되어 있습니다.
SELCHANGE EN_SELCHANGE 알림 코드와 관련된 정보를 포함합니다. 서식 있는 편집 컨트롤은 현재 선택 항목이 변경될 때 부모 창에 이 알림을 보냅니다.
SETTEXTEX 텍스트를 설정하는 데 사용할 코드 페이지(있는 경우), 텍스트가 컨트롤의 모든 텍스트를 바꿀지 선택 항목만 바꿀지 여부, 실행 취소 상태를 유지할지 여부를 지정합니다. 이 구조체는 EM_SETTEXTEX 메시지와 함께 사용됩니다.
TABLECELLPARMS 표 행에 있는 셀의 특성을 정의합니다.
TABLEROWPARMS 테이블의 행 특성을 정의합니다.
TASKDIALOG_BUTTON TASKDIALOG_BUTTON 구조에는 작업 대화 상자에 단추를 표시하는 데 사용되는 정보가 포함되어 있습니다. TASKDIALOGCONFIG 구조체에서 이 구조체를 사용합니다.

TASKDIALOGCONFIG TASKDIALOGCONFIG 구조에는 작업 대화 상자를 표시하는 데 사용되는 정보가 포함되어 있습니다. TaskDialogIndirect 함수에서 이 구조체를 사용합니다.
TBADDBITMAP 단추 이미지가 포함된 비트맵을 도구 모음에 추가합니다.
TBBUTTON 도구 모음의 단추에 대한 정보를 포함합니다.
TBBUTTONINFOA 도구 모음의 특정 단추에 대한 정보를 포함하거나 받습니다. (ANSI)
TBBUTTONINFOW 도구 모음의 특정 단추에 대한 정보를 포함하거나 받습니다. (유니코드)
TBINSERTMARK 도구 모음 컨트롤의 삽입 표시에 대한 정보를 포함합니다.
TBMETRICS 도구 모음 항목을 축소하거나 확장하는 데 사용되는 도구 모음의 메트릭을 정의합니다.
TBREPLACEBITMAP TB_REPLACEBITMAP 메시지와 함께 사용되어 도구 모음 비트맵을 다른 도구 모음 비트맵으로 바꿉니다.
TBSAVEPARAMSA TB_SAVERESTORE 메시지가 저장되는 레지스트리 위치를 지정하고 도구 모음 상태에 대한 정보를 검색합니다. (ANSI)
TBSAVEPARAMSW TB_SAVERESTORE 메시지가 저장되는 레지스트리 위치를 지정하고 도구 모음 상태에 대한 정보를 검색합니다. (유니코드)
TCHITTESTINFO 적중 테스트에 대한 정보를 포함합니다. 이 구조체는 TC_HITTESTINFO 구조체를 대체합니다.

TCITEMA 탭 항목의 특성을 지정하거나 받습니다. TCM_INSERTITEM, TCM_GETITEM 및 TCM_SETITEM 메시지와 함께 사용됩니다. 이 구조체는 TC_ITEM 구조체를 대체합니다. (ANSI)
TCITEMHEADERA 탭의 특성을 지정하거나 수신합니다. TCM_INSERTITEM, TCM_GETITEM 및 TCM_SETITEM 메시지와 함께 사용됩니다. 이 구조체는 TC_ITEMHEADER 구조체를 대체합니다. (ANSI)
TCITEMHEADERW 탭의 특성을 지정하거나 수신합니다. TCM_INSERTITEM, TCM_GETITEM 및 TCM_SETITEM 메시지와 함께 사용됩니다. 이 구조체는 TC_ITEMHEADER 구조체를 대체합니다. (유니코드)
TCITEMW 탭 항목의 특성을 지정하거나 받습니다. TCM_INSERTITEM, TCM_GETITEM 및 TCM_SETITEM 메시지와 함께 사용됩니다. 이 구조체는 TC_ITEM 구조체를 대체합니다. (유니코드)
TEXTRANGEA 서식 있는 편집 컨트롤의 텍스트 범위입니다. 이 구조체는 EM_GETTEXTRANGE 메시지로 채워집니다. lpstrText 멤버가 가리키는 버퍼는 모든 문자와 종료 Null 문자를 받을 수 있을 만큼 충분히 커야 합니다. (ANSI)
TEXTRANGEW 서식 있는 편집 컨트롤의 텍스트 범위입니다. 이 구조체는 EM_GETTEXTRANGE 메시지로 채워집니다. lpstrText 멤버가 가리키는 버퍼는 모든 문자와 종료 Null 문자를 받을 수 있을 만큼 충분히 커야 합니다. (유니코드)
TOUCH_HIT_TESTING_INPUT 터치 디지타이저에서 보고한 터치 접촉 영역에 대한 정보를 포함합니다.
TOUCH_HIT_TESTING_PROXIMITY_EVALUATION 터치 접촉 영역과 교차하는 다른 개체를 기준으로 개체가 터치 접촉 영역의 대상일 가능성이 있는지 여부를 나타내는 적중 테스트 점수를 포함합니다.
TTGETTITLE 도구 설명 컨트롤의 제목에 대한 정보를 제공합니다.
TTHITTESTINFOA 도구 설명 컨트롤에서 점이 지정된 도구의 경계 사각형에 있는지 여부를 확인하는 데 사용하는 정보를 포함합니다. 점이 사각형에 있으면 구조체가 도구에 대한 정보를 받습니다. (ANSI)

TTHITTESTINFOW 도구 설명 컨트롤에서 점이 지정된 도구의 경계 사각형에 있는지 여부를 확인하는 데 사용하는 정보를 포함합니다. 점이 사각형에 있으면 구조체가 도구에 대한 정보를 받습니다. (유니코드)
TTTOOLINFOA TOOLINFO 구조체는 도구 설명 컨트롤의 도구에 대한 정보를 포함합니다. (ANSI)
TTTOOLINFOW TOOLINFO 구조체는 도구 설명 컨트롤의 도구에 대한 정보를 포함합니다. (유니코드)
TVGETITEMPARTRECTINFO 트리 항목의 지정된 부분에 대한 "적중 영역"을 식별하기 위한 정보를 포함합니다. 구조체는 TVM_GETITEMPARTRECT 메시지 및 TreeView_GetItemPartRect 매크로와 함께 사용됩니다.
TVHITTESTINFO 트리 뷰 컨트롤을 기준으로 지점의 위치를 결정하는 데 사용되는 정보를 포함합니다.
TVINSERTSTRUCTA 트리 뷰 컨트롤에 새 항목을 추가하는 데 사용되는 정보를 포함합니다. 이 구조체는 TVM_INSERTITEM 메시지와 함께 사용됩니다. 구조체는 TV_INSERTSTRUCT 구조체와 동일하지만 현재 명명 규칙을 따르도록 이름이 변경되었습니다. (ANSI)
TVINSERTSTRUCTW 트리 뷰 컨트롤에 새 항목을 추가하는 데 사용되는 정보를 포함합니다. 이 구조체는 TVM_INSERTITEM 메시지와 함께 사용됩니다. 구조체는 TV_INSERTSTRUCT 구조체와 동일하지만 현재 명명 규칙을 따르도록 이름이 변경되었습니다. (유니코드)
TVITEMA 트리 뷰 항목의 특성을 지정하거나 받습니다. 이 구조체는 TV_ITEM 구조체와 동일하지만 현재 명명 규칙을 따르도록 이름이 변경되었습니다. 새 애플리케이션은 이 구조체를 사용해야 합니다. (ANSI)
TVITEMEXA 트리 뷰 항목의 특성을 지정하거나 받습니다. 이 구조체는 TVITEM 구조체의 향상된 기능입니다. 새 애플리케이션은 적절한 경우 이 구조체를 사용해야 합니다. (ANSI)
TVITEMEXW 트리 뷰 항목의 특성을 지정하거나 받습니다. 이 구조체는 TVITEM 구조체의 향상된 기능입니다. 새 애플리케이션은 적절한 경우 이 구조체를 사용해야 합니다. (유니코드)

TVITEMW 트리 뷰 항목의 특성을 지정하거나 받습니다. 이 구조체는 TV_ITEM 구조체와 동일하지만 현재 명명 규칙을 따르도록 이름이 변경되었습니다. 새 애플리케이션은 이 구조체를 사용해야 합니다. (유니코드)
TVSORTCB 트리 뷰 컨트롤에서 자식 항목을 정렬하는 데 사용되는 정보를 포함합니다. 이 구조체는 TVM_SORTCHILDRENCB 메시지와 함께 사용됩니다. 이 구조체는 TV_SORTCB 구조체와 동일하지만 현재 명명 규칙을 따르도록 이름이 변경되었습니다.
UDACCEL 업다운 컨트롤에 대한 가속 정보가 포함됩니다.
USAGE_PROPERTIES 모든 유형의 HID 입력 디바이스에 대한 디바이스 속성(HID 사용량에 해당하는 HID(휴먼 인터페이스 디바이스) 전역 항목)을 포함합니다.
WTA_OPTIONS 창 시각적 스타일 특성을 설정하는 데 사용되는 옵션을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Windows 데이터 액세스 구성 요소 WMI 공급자

아티클2023. 03. 09.

Windows 데이터 액세스 구성 요소 WMI 공급자 기술 개요입니다.

Windows 데이터 액세스 구성 요소 WMI 공급자 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 데이터 액세스 구성 요소 WMI 공급자

Windows Deployment Services

아티클2023. 03. 16.

Windows 배포 서비스 기술 개요입니다.

Windows 배포 서비스를 개발하려면 다음 헤더가 필요합니다.

- wdsbp.h
- wdsclientapi.h
- wdspxe.h
- wdstci.h
- wdstpdi.h
- wdstptmgmt.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows Deployment Services

열거형

__unnamed_enum_1 __unnamed_enum_1(wdsclientapi.h) 열거형은 WdsCliLog 함수에서 사용됩니다.
__unnamed_enum_2 __unnamed_enum_2(wdsclientapi.h) 열거형은 WdsCliLog 함수에서 사용됩니다.
TRANSPORTCLIENT_CALLBACK_ID 이 열거형은 WdsTransportClientRegisterCallback 함수에 의해 수신됩니다.
TRANSPORTPROVIDER_CALLBACK_ID 이 구조는 WdsTransportServerRegisterCallback 함수에서 사용됩니다.
WDSTRANSPORT_DIAGNOSTICS_COMPONENT_FLAGS 진단을 사용하도록 설정된 WDS 구성 요소를 구성합니다. WDS 진단 로그 이벤트를 시스템 이벤트 로그에 기록합니다.
WDSTRANSPORT_DISCONNECT_TYPE WDS 클라이언트가 세션에서 연결이 끊어지면 수행해야 하는 작업을 나타냅니다.

WDSTRANSPORT_FEATURE_FLAGS WDS 서버에 설치된 WDS 기능을 나타냅니다.
WDSTRANSPORT_IP_ADDRESS_SOURCE_TYPE WDS 멀티캐스트 공급자가 새 세션에 대한 멀티캐스트 주소를 가져오는 원본을 나타냅니다.
WDSTRANSPORT_IP_ADDRESS_TYPE IP 주소의 유형을 나타냅니다.
WDSTRANSPORT_NAMESPACE_TYPE 이 네임스페이스에서 다루는 개체를 전송하는 데 사용되는 멀티캐스트 세션의 유형을 결정합니다.
WDSTRANSPORT_NETWORK_PROFILE_TYPE WDS 전송 프로토콜에서 네트워크에서 데이터 전송을 최적화하는 데 사용되는 설정을 정의합니다.
WDSTRANSPORT_PROTOCOL_FLAGS WDS 전송 서버에서 지원하는 프로토콜을 지정합니다.
WDSTRANSPORT_SERVICE_NOTIFICATION 구성 변경 후 설정을 다시 읽는 등 WDS 전송 서비스에 알릴 때 수행해야 하는 작업을 지정합니다.
WDSTRANSPORT_SLOW_CLIENT_HANDLING_TYPE Windows Server 2008 R2에서 실행되는 WDS 전송 서버가 멀티캐스트 전송 속도를 저하시키는 클라이언트 컴퓨터를 처리하는 데 사용해야 하는 자동 작업의 유형을 지정합니다.
WDSTRANSPORT_TFTP_CAPABILITY WDS TFTP 서버에서 지원되는 기능을 나타냅니다.
WDSTRANSPORT_UDP_PORT_POLICY UDP 포트를 할당할 때 WDS 전송 서비스에서 사용해야 하는 정책을 지정합니다.

함수

복제 이 네임스페이스 개체의 정보를 메모리에 등록되지 않은 새 네임스페이스 개체로 복사합니다.
커밋 IWdsTransportCacheable::D irty 속성이 설정된 경우 개체 데이터 멤버를 기본 데이터 저장소에 커밋합니다. 그렇지 않으면 메서드가 아무 작업도 없이 반환됩니다.
CreateNamespace 현재 WDS 전송 서버에 등록할 수 있는 IWdsTransportNamespace 인터페이스의 개체를 만듭니다.
등록 취소 서버에서 네임스페이스를 등록 취소합니다.
DeregisterContentProvider 클라이언트 컴퓨터에서 애플리케이션을 실행하여 콘텐츠 공급자를 등록 취소할 수 있습니다. 이렇게 하면 WDS 전송 서버에서 공급자를 더 이상 사용할 수 없습니다.
DisableWdsTransportServices 모든 WDS 전송 서비스를 사용 안 함 모드로 설정합니다.
무시 IWdsTransportCacheable::D irty 속성을 지우고 개체의 IWdsTransportCacheable::Refresh 메서드를 호출하여 현재 개체 데이터를 다시 읽어 개체 데이터 멤버에 대한 모든 변경 내용을 삭제합니다.
연결 끊기 세션에서 WDS 클라이언트의 연결을 끊고 연결이 끊어지면 클라이언트가 수행해야 하는 작업을 지정합니다.
DisconnectClient 전송 세션에서 WDS 클라이언트의 연결을 끊고 연결이 끊어지면 WDS 클라이언트가 수행해야 하는 작업을 지정합니다.
EnableWdsTransportServices 모든 WDS 전송 서비스를 자동 시작 모드로 설정합니다.
get__NewEnum 이 컬렉션의 개체를 반복하는 데 사용할 수 있는 열거자에 대한 포인터를 받습니다.

get_AutoDisconnectThreshold 서버에서 사용하는 임계값 전송 속도(초당 킬로바이트)를 설정하거나 검색합니다. (Get)
get_BlockSize TFTP 세션에서 사용되는 블록 크기를 검색합니다.
get_Components WDS 클라이언트가 이벤트 로깅이 있는 WDS 전송 구성 요소를 구성할 수 있습니다. (Get)
get_Configuration 네임스페이스의 콘텐츠 공급자에 대한 구성 정보를 설정하거나 검색합니다. (Get)
get_ConfigurationManager 이 서버의 구성을 관리하는 데 사용되는 IWdsTransportConfigurationManager 인터페이스의 개체에 대한 포인터를 반환합니다.
get_Content WDS 전송 서버의 활성 전송 세션을 나타내는 IWdsTransportContent 인터페이스의 개체에 대한 포인터를 받습니다.
get_ContentProvider 네임스페이스에 대한 콘텐츠 공급자를 설정하거나 검색합니다. (Get)
get_ContentProviders IWdsTransportCollection 인터페이스의 인스턴스에 대한 포인터를 받습니다. 컬렉션에는 서버에 등록된 콘텐츠 공급자에 대한 IWdsTransportContentProvider 인터페이스의 개체가 포함됩니다.
get_Count 이 컬렉션의 개체 수를 받습니다.
get_CpuUtilization WDS 클라이언트의 CPU 사용률 비율을 받습니다.
get_CurrentFileOffset TFTP 세션에서 전송되는 현재 블록의 바이트 단위로 파일 시작 시 오프셋을 받습니다.

get_Description 콘텐츠 공급자에 대한 설명을 검색합니다.
get_Description 네임스페이스에 대한 설명을 설정하거나 검색합니다. (Get)
get_DiagnosticsPolicy Configuration Manager 진단 정책 개체에 대한 인터페이스 포인터를 받습니다. 이 개체를 사용하여 WDS 전송 서버 구성 요소가 진단 이벤트 로깅에 사용할 수 있는 진단 설정을 구성할 수 있습니다.
get_Dirty 개체 데이터가 수정되었는지 여부를 나타내는 값을 받습니다.
get_Enabled WDS 클라이언트가 서버에서 진단 로깅을 구성할 수 있도록 하는 값을 받거나 설정합니다. (Get)
get_EnableTftpVariableWindowExtension TFTP 서버에서 변수 창 확장을 사용할 수 있는지 여부를 지정하는 WDSTRANSPORT_TFTP_CAPABILITY 값을 받습니다. 이 값을 사용하면 WDS 클라이언트 컴퓨터가 변수 창 확장을 사용하도록 WDS TFTP 서버를 구성할 수 있습니다. (Get)
get_EndIpAddress WDS 클라이언트 컴퓨터가 지정된 유형의 IP 주소에 대한 멀티캐스트 IP 주소 범위의 끝을 구성할 수 있도록 합니다. (Get)
get_EndPort WDS 클라이언트 컴퓨터가 WDS 전송 서비스에서 사용하는 UDP 포트 범위의 끝을 구성할 수 있도록 합니다. (Get)
get_FileName TFTP 세션에서 전송되는 파일의 이름을 검색합니다.
get_FilePath 콘텐츠 공급자의 DLL 경로를 검색합니다.
get_FileSize 전송되는 파일의 크기(바이트)를 받습니다.

get_FriendlyName 네임스페이스의 사용자에게 친숙한 이름을 설정하거나 검색합니다. (Get)
get_Id WDS 서버에서 이 WDS 클라이언트를 식별하는 고유한 클라이언트 ID를 받습니다.
get_Id 서버에서 이 콘텐츠 개체를 식별하는 고유한 콘텐츠 ID를 받습니다.
get_Id 서버에 등록된 네임스페이스의 고유한 네임스페이스 ID를 받습니다.
get_Id 서버에서 이 세션을 식별하는 고유한 세션 ID를 받습니다.
get_InitializationRoutine 콘텐츠 공급자를 초기화하기 위해 호출된 메서드의 이름을 검색합니다.
get_InstalledFeatures 서버에 설치된 WDS 기능을 나타내는 값을 받습니다.
get_IpAddress WDS 클라이언트의 IP 주소를 포함하는 문자열 값을 받습니다.
get_IpAddress 클라이언트의 IP 주소를 포함하는 문자열 값을 받습니다.
get_IpAddressSource WDS 클라이언트 컴퓨터가 지정된 유형의 IP 주소에 대해 WDS 전송 서버가 멀티캐스트 주소를 가져오는 IP 주소 원본을 구성할 수 있도록 합니다. (Get)
get_Item 지정된 인덱스와 일치하는 개체에 대한 포인터를 받습니다.
get_JoinDuration WDS 클라이언트가 전송 세션에 조인된 이후 경과된 시간(초)을 받습니다.

get_MacAddress WDS 클라이언트의 MAC 주소를 받습니다.
get_MasterClientId 이 세션의 마스터 클라이언트를 식별하는 WDS 서버에서 할당한 고유 클라이언트 ID를 받습니다.
get_MemoryUtilization 사용 중인 WDS 클라이언트 메모리의 백분율을 받습니다.
get_MinimumClients 자동으로 전송을 시작하는 조건을 검색하거나 설정합니다. 세션에 조인된 애플리케이션 수가 이 속성의 값에 도달하면 전송이 시작됩니다. (Get)
get_MulticastSessionPolicy Configuration Manager 멀티캐스트 세션 정책 개체에 대한 인터페이스 포인터를 받습니다. 이 개체는 Windows Server 2008 R2부터 사용할 수 있는 멀티캐스트 세션 매개 변수를 구성하는 데 사용할 수 있습니다.
get_MultistreamStreamCount 서버에서 사용하는 전송당 최대 멀티캐스트 스트림 수를 받습니다. (Get)
get_Name WDS 서버에서 WDS 클라이언트의 이름을 받습니다.
get_Name 이 콘텐츠 개체가 나타내는 데이터 개체의 이름을 포함하는 문자열 값에 대한 포인터를 받습니다.
get_Name 콘텐츠 공급자의 이름을 검색합니다.
get_Name 네임스페이스의 이름을 설정하거나 검색합니다. (Get)
get_Name 이 개체가 나타내는 서버의 이름을 반환합니다.

get_Namespace 이 콘텐츠와 연결된 네임스페이스를 나타내는 IWdsTransportNamespace 인터페이스의 개체에 대한 포인터를 받습니다.
get_NamespaceManager 이 서버에서 네임스페이스를 관리하는 데 사용되는 IWdsTransportNamespaceManager 인터페이스의 개체에 대한 포인터를 반환합니다.
get_NetworkInterfaceAddress 이 전송 세션에서 사용하는 서버 네트워크 인터페이스의 MAC 주소를 받습니다.
get_NetworkInterfaceName 이 전송 세션에서 사용하는 서버 네트워크 인터페이스의 이름을 받습니다.
get_NetworkProfile 클라이언트 컴퓨터가 WDS 전송 서버에서 사용하는 네트워크 프로필을 구성할 수 있도록 합니다. (Get)
get_NetworkUtilization 사용된 WDS 클라이언트의 네트워크 대역폭 백분율을 받습니다.
get_PercentCompletion 다운로드한 현재 개체의 백분율을 받습니다.
get_Protocols WDS 서버에서 지원하는 전송 프로토콜을 나타내는 값을 받습니다.
get_Registered 네임스페이스가 서버에 등록되어 있는지 여부를 나타내는 값을 반환합니다.
get_ServicePolicy Configuration Manager 서비스 정책 개체에 대한 인터페이스 포인터를 받습니다. 이 개체는 멀티캐스트 IP 주소 원본 및 활성 네트워크 프로필과 같은 서비스 매개 변수를 구성하는 데 사용할 수 있습니다.
get_Session WDS 클라이언트가 조인되는 전송 세션을 받습니다.

get_SetupManager 이 서버의 설정 기능을 관리하는 데 사용되는 IWdsTransportSetupManager 인터페이스의 개체에 대한 포인터를 반환합니다.
get_SlowClientFallback 멀티캐스트 전송에서 느린 클라이언트의 연결을 자동으로 끊을 때 서버에서 요청한 대체 정책을 나타내는 값을 받습니다. (Get)
get_SlowClientHandling 멀티캐스트 전송 속도가 느려지는 클라이언트를 처리하기 위해 서버에서 사용하는 메서드를 나타내는 WDSTRANSPORT_SLOW_CLIENT_HANDLING_TYPE 값을 설정하거나 검색합니다. (Get)
get_StartIpAddress WDS 클라이언트 컴퓨터가 지정된 유형의 IP 주소에 대한 멀티캐스트 IP 주소 범위의 시작을 구성할 수 있도록 합니다. (Get)
get_StartPort WDS 클라이언트 컴퓨터가 WDS 전송 서비스에서 사용되는 UDP 포트 범위의 시작을 구성할 수 있도록 합니다. (Get)
get_StartTime 자동으로 전송을 시작하는 조건을 검색하거나 설정합니다. 전송은 UTC 시간이 이 속성의 값에 도달하면 시작됩니다. (Get)
get_TftpCapabilities WDS TFTP 서버에서 지원되는 WDS TFTP 기능을 나타내는 WDSTRANSPORT_TFTP_CAPABILITY 값의 마스크를 받습니다.
get_TftpManager WDS TFTP 서버를 관리하는 데 사용되는 IWdsTransportTftpManager 인터페이스의 개체에 대한 포인터를 받습니다.
get_TftpMaximumBlockSize TFTP 서버에서 사용하는 최대 블록 크기의 값을 받습니다. 이 값을 사용하면 WDS 클라이언트 컴퓨터에서 최대 블록 크기를 구성할 수 있습니다. (Get)
get_Timeout 클라이언트와 통신하는 데 사용되는 시간 제한(초)을 받습니다.

get_Tombstoned 모든 활성 세션이 완료되거나 종료될 때까지 서버가 등록 취소된 네임스페이스의 네임스페이스 개체를 메모리에 저장했는지 여부를 나타내는 값을 반환합니다.
get_TombstoneTime 서버가 등록 취소된 네임스페이스의 네임스페이스 개체를 저장한 UTC 날짜 및 시간을 반환합니다.
get_TransferRate 이 세션의 데이터 전송 속도를 초당 바이트 단위로 받습니다.
get_TransmissionStarted 서버가 이 네임스페이스에서 데이터 전송을 시작했는지 여부를 나타내는 값을 받습니다.
get_Type 애플리케이션이 이 개체의 네임스페이스 형식을 검색할 수 있도록 합니다.
get_UdpPortPolicy WDS 전송 서비스에서 사용할 UDP 포트 할당 정책을 지정하는 WDSTRANSPORT_UDP_PORT_POLICY 값을 수신합니다. 이 값을 사용하면 WDS 클라이언트 컴퓨터에서 UDP 포트 할당 정책을 구성할 수 있습니다. (Get)
get_UserIdentity 이 클라이언트와 연결된 사용자 ID를 나타내는 문자열을 받습니다.
get_Version WDS 서버의 운영 체제 버전을 나타내는 값을 받습니다.
get_WdsTransportServicesRunning WDS 전송 서비스가 서버에서 실행 중인지 여부를 나타내는 값을 받습니다.
get_WindowSize TFTP 세션에서 사용되는 현재 창 크기를 받습니다. TFTP 세션 중에 사용되는 창 크기는 가변 창 TFTP 확장을 사용할 때 ACK 패킷을 통해 클라이언트에서 업데이트할 수 있습니다.
GetWdsTransportServer WDS 전송 서버를 관리하는 데 사용할 수 있는 IWdsTransportServer 인터페이스의 개체를 만듭니다. 이 메서드는 시스템이 지정된 이름의 WDS 전송 서버에 연결할 수 있음을 확인합니다.

NotifyWdsTransportServices WDS 전송 서비스에 알림을 보냅니다. 알림 값이 적절한 WDS 전송 서비스 컨트롤로 변환된 다음 이러한 컨트롤이 적절한 서비스로 전송됩니다.
PFN_WdsCliCallback 파일 또는 이미지 전송 중에 WDS에서 진행률 알림 및 오류 메시지를 호출할 수 있는 콜백 함수를 정의합니다.
PFN_WdsCliTraceFunction WDS 클라이언트에서 디버깅 메시지를 받을 수 있는 콜백 함수를 정의합니다.
PFN_WdsTransportClientReceiveContents PFN_WdsTransportClientReceiveContents 콜백은 멀티캐스트 클라이언트에서 데이터 블록을 사용할 준비가 되었음을 나타내는 데 사용됩니다.
PFN_WdsTransportClientReceiveMetadata PFN_WdsTransportClientReceiveMetadata 소비자가 파일에 대한 메타데이터 형식 정보를 수신하도록 등록할 수 있는 선택적 콜백입니다. 이 정보는 콘텐츠 공급자가 제공하며 멀티캐스트 클라이언트 및 서버에 불투명합니다.
PFN_WdsTransportClientSessionComplete PFN_WdsTransportClientSessionCompete 콜백은 더 이상 콜백이 소비자에게 전송되지 않으며 세션이 성공적으로 완료되었거나 복구할 수 없는 오류가 발생했음을 나타내기 위해 클라이언트에서 사용됩니다.
PFN_WdsTransportClientSessionStart PFN_WdsTransportClientSessionStart 콜백은 멀티캐스트 세션의 시작 부분에 호출되어 파일에 대한 파일 크기 및 기타 서버 쪽 정보를 소비자에게 나타냅니다. (PFN_WdsTransportClientSessionStart)
PFN_WdsTransportClientSessionStartEx PFN_WdsTransportClientSessionStart 콜백은 멀티캐스트 세션의 시작 부분에 호출되어 파일에 대한 파일 크기 및 기타 서버 쪽 정보를 소비자에게 나타냅니다. (PFN_WdsTransportClientSessionStartEx)
put_AutoDisconnectThreshold 서버에서 사용하는 임계값 전송 속도(초당 킬로바이트)를 설정하거나 검색합니다. (Put)

put_Components WDS 클라이언트가 이벤트 로깅이 있는 WDS 전송 구성 요소를 구성할 수 있도록 합니다. (Put)
put_Configuration 네임스페이스의 콘텐츠 공급자에 대한 구성 정보를 설정하거나 검색합니다. (Put)
put_ContentProvider 네임스페이스에 대한 콘텐츠 공급자를 설정하거나 검색합니다. (Put)
put_Description 네임스페이스에 대한 설명을 설정하거나 검색합니다. (Put)
put_Enabled WDS 클라이언트가 서버에서 진단 로깅을 구성할 수 있도록 하는 값을 받거나 설정합니다. (Put)
put_EnableTftpVariableWindowExtension TFTP 서버에서 변수 창 확장을 사용할 수 있는지 여부를 지정하는 WDSTRANSPORT_TFTP_CAPABILITY 값을 받습니다. 이 값을 사용하면 WDS 클라이언트 컴퓨터가 변수 창 확장을 사용하도록 WDS TFTP 서버를 구성할 수 있습니다. (Put)
put_EndIpAddress WDS 클라이언트 컴퓨터가 지정된 유형의 IP 주소에 대한 멀티캐스트 IP 주소 범위의 끝을 구성할 수 있도록 합니다. (Put)
put_EndPort WDS 클라이언트 컴퓨터가 WDS 전송 서비스에서 사용하는 UDP 포트 범위의 끝을 구성할 수 있도록 합니다. (Put)
put_FriendlyName 네임스페이스의 사용자에게 친숙한 이름을 설정하거나 검색합니다. (Put)
put_IpAddressSource WDS 클라이언트 컴퓨터가 지정된 유형의 IP 주소에 대해 WDS 전송 서버가 멀티캐스트 주소를 가져오는 IP 주소 원본을 구성할 수 있도록 합니다. (Put)
put_MinimumClients 자동으로 전송을 시작하는 조건을 검색하거나 설정합니다. 세션에 조인된 애플리케이션 수가 이 속성의 값에 도달하면 전송이 시작됩니다. (Put)

put_MultistreamStreamCount 서버에서 사용하는 전송당 최대 멀티캐스트 스트림 수를 받습니다. (Put)
put_Name 네임스페이스의 이름을 설정하거나 검색합니다. (Put)
put_NetworkProfile 클라이언트 컴퓨터가 WDS 전송 서버에서 사용하는 네트워크 프로필을 구성할 수 있도록 합니다. (Put)
put_SlowClientFallback 멀티캐스트 전송에서 느린 클라이언트의 연결을 자동으로 끊을 때 서버에서 요청한 대체 정책을 나타내는 값을 받습니다. (Put)
put_SlowClientHandling 멀티캐스트 전송 속도가 느려지는 클라이언트를 처리하기 위해 서버에서 사용하는 메서드를 나타내는 WDSTRANSPORT_SLOW_CLIENT_HANDLING_TYPE 값을 설정하거나 검색합니다. (Put)
put_StartIpAddress WDS 클라이언트 컴퓨터가 지정된 유형의 IP 주소에 대한 멀티캐스트 IP 주소 범위의 시작을 구성할 수 있도록 합니다. (Put)
put_StartPort WDS 클라이언트 컴퓨터가 WDS 전송 서비스에서 사용되는 UDP 포트 범위의 시작을 구성할 수 있도록 합니다. (Put)
put_StartTime 자동으로 전송을 시작하는 조건을 검색하거나 설정합니다. 전송은 UTC 시간이 이 속성의 값에 도달하면 시작됩니다. (Put)
put_TftpMaximumBlockSize TFTP 서버에서 사용하는 최대 블록 크기의 값을 받습니다. 이 값을 사용하면 WDS 클라이언트 컴퓨터에서 최대 블록 크기를 구성할 수 있습니다. (Put)
put_UdpPortPolicy WDS 전송 서비스에서 사용할 UDP 포트 할당 정책을 지정하는 WDSTRANSPORT_UDP_PORT_POLICY 값을 수신합니다. 이 값을 사용하면 WDS 클라이언트 컴퓨터에서 UDP 포트 할당 정책을 구성할 수 있습니다. (Put)

PxeAsyncRecvDone 클라이언트 요청을 비동기적으로 처리한 결과를 전달합니다. PxeProviderRecvRequest 함수가 ERROR_IO_PENDING 반환하는 경우에만 이 함수를 호출해야 합니다.
PxeDhcpAppendOption 회신 패킷에 DHCP 옵션을 추가합니다. (PxeDhcpAppendOption)
PxeDhcpAppendOptionRaw 회신 패킷에 DHCP 옵션을 추가합니다. (PxeDhcpAppendOptionRaw)
PxeDhcpGetOptionValue DHCP 패킷에서 옵션 값을 검색합니다.
PxeDhcpGetVendorOptionValue DHCP 패킷의 공급업체별 정보 필드(43)에서 옵션 값을 검색합니다.
PxeDhcpInitialize 응답 패킷을 DHCP 회신 패킷으로 초기화합니다.
PxeDhcpIsValid 패킷이 DHCP 패킷인지 확인합니다.
PxeDhcpv6AppendOption 회신 패킷에 DHCPv6 옵션을 추가합니다. (PxeDhcpv6AppendOption)
PxeDhcpv6AppendOptionRaw 회신 패킷에 DHCPv6 옵션을 추가합니다. (PxeDhcpv6AppendOptionRaw)
PxeDhcpv6CreateRelayRepl RELAY-REPL 메시지를 생성합니다.
PxeDhcpv6GetOptionValue DHCPv6 패킷에서 옵션 값을 검색합니다.
PxeDhcpv6GetVendorOptionValue DHCPv6 패킷의 OPTION_VENDOR_OPTS(17) 필드에서 옵션 값을 검색합니다.

PxeDhcpv6Initialize 응답 패킷을 DHCPv6 회신 패킷으로 초기화합니다.
PxeDhcpv6IsValid 패킷이 유효한 DHCPv6 패킷인지 확인합니다.
PxeDhcpv6ParseRelayForw 이 함수는 공급자가 RELAY-FORW 메시지와 중첩된 OPTION_RELAY_MSG 메시지를 구문 분석하는 데 사용할 수 있습니다.
PxeGetServerInfo PXE 서버에 대한 정보를 반환합니다. (PxeGetServerInfo)
PxeGetServerInfoEx PXE 서버에 대한 정보를 반환합니다. (PxeGetServerInfoEx)
PxePacketAllocate PxeSendReply 함수를 사용하여 보낼 패킷을 할당합니다.
PxePacketFree PxePacketAllocate 함수에 의해 할당된 패킷을 해제합니다.
PxeProviderEnumClose PxeProviderEnumFirst 함수에서 연 공급자의 열거형을 닫습니다.
PxeProviderEnumFirst 등록된 공급자의 열거형을 시작합니다.
PxeProviderEnumNext 등록된 공급자를 열거합니다.
PxeProviderFreeInfo PxeProviderEnumNext 함수에 의해 할당된 메모리를 해제합니다.
PxeProviderQueryIndex 등록된 공급자 목록에서 지정된 공급자의 인덱스를 반환합니다.

PxeProviderRegister 공급자를 시스템에 등록합니다.
PxeProviderSetAttribute 공급자에 대한 특성을 지정합니다.
PxeProviderUnRegister 등록된 공급자 목록에서 공급자를 제거합니다.
PxeRegisterCallback 다른 알림 이벤트에 대한 콜백 함수를 등록합니다.
PxeSendReply 클라이언트 요청에 패킷을 보냅니다.
PxeTrace 추적 항목을 PXE 로그에 추가합니다.
새로 고침 기본 데이터 저장소에서 값을 다시 읽어 개체 데이터 멤버를 새로 고칩니다. 개체의 IWdsTransportCacheable::D irty 속성이 설정된 경우에만 허용됩니다.
새로 고침 서버의 값을 사용하여 네임스페이스의 속성 값을 다시 설정합니다.
등록 서버에 네임스페이스를 등록합니다.
RegisterContentProvider 클라이언트 컴퓨터에서 애플리케이션을 실행하여 콘텐츠 공급자 DLL을 등록할 수 있습니다. 이렇게 하면 WDS 전송 서버에서 공급자를 사용할 수 있습니다.
RestartWdsTransportServices 실행 중인 모든 WDS 전송 서비스를 중지한 다음 다시 시작합니다. 실행 중인 서비스가 없으면 메서드는 추가 작업 없이 S_OK 반환합니다.

RetrieveClients 전송 세션에 조인된 WDS 클라이언트의 컬렉션을 검색합니다.
RetrieveContents 네임스페이스와 연결된 활성 전송 콘텐츠 개체의 컬렉션을 검색합니다.
RetrieveNamespace 이름으로 IWdsTransportNamespace 인터페이스의 개체를 검색합니다. 이름은 WDS 전송 서버의 네임스페이스에 등록해야 합니다.
RetrieveNamespaces 지정된 조건과 일치하는 서버의 네임스페이스를 나타내는 IWdsTransportNamespace 인터페이스의 개체 컬렉션을 반환합니다.
RetrieveSessions 이 콘텐츠와 연결된 활성 전송 세션의 컬렉션을 검색합니다.
RetrieveTftpClients 현재 TFTP 서버에 연결된 클라이언트에 대한 IWdsTransportTftpClient 인터페이스의 개체 컬렉션을 포함하는 IWdsTransportCollection 인터페이스의 개체에 대한 포인터를 반환합니다.
StartTransmission 네임스페이스에서 전송을 시작합니다.
StartWdsTransportServices 모든 WDS 전송 서비스를 시작합니다. 이 메서드는 구성을 변경하지 않고 WDS 전송 서비스의 실행 상태를 변경하는 방법을 제공합니다.
StopWdsTransportServices 모든 WDS 전송 서비스를 중지합니다. 이 메서드는 구성을 변경하지 않고 WDS 전송 서비스의 실행 상태를 변경하는 방법을 제공합니다.
Terminate 콘텐츠에서 모든 활성 세션을 종료하고 해당 콘텐츠에 조인된 클라이언트의 연결을 끊어 이 콘텐츠의 전송을 종료합니다.

Terminate WDS 전송 서버에서 활성 세션을 종료하고 세션에 조인된 모든 WDS 클라이언트의 연결을 끊습니다.
WdsBpAddOption 패킷에 옵션을 추가합니다.
WdsBpCloseHandle 지정된 핸들을 닫습니다.
WdsBpGetOptionBuffer DHCP 패킷 옵션에 추가해야 하는 버퍼에 정보를 복사합니다.
WdsBpInitialize WDS 네트워크 부팅 프로그램에 대한 옵션을 생성합니다.
WdsBpParseInitialize 네트워크 부팅 프로그램에서 보낸 패킷에 대한 핸들을 받습니다. (WdsBpParseInitialize)
WdsBpParseInitializev6 네트워크 부팅 프로그램에서 보낸 패킷에 대한 핸들을 받습니다. (WdsBpParseInitializev6)
WdsBpQueryOption 구문 분석된 패킷에서 옵션 값을 반환합니다.
WdsCliAuthorizeSession WDS 서버의 세션을 인증된 세션으로 변환합니다.
WdsCliCancelTransfer WDS 전송 작업을 취소합니다.
WdsCliClose WDS 세션 또는 이미지에 대한 핸들을 닫고 리소스를 해제합니다.
WdsCliCreateSession WDS 서버를 사용하여 새 세션을 시작합니다.

WdsCliFindFirstImage WDS 서버에 저장된 이미지의 열거를 시작하고 첫 번째 이미지를 참조하는 찾기 핸들을 반환합니다.
WdsCliFindNextImage 찾기 핸들의 참조를 WDS 서버에 저장된 다음 이미지로 이동합니다.
WdsCliFreeStringArray 이 함수를 사용하여 WdsCliObtainDriverPackages 함수에 의해 할당되는 문자열 값 배열을 해제할 수 있습니다.
WdsCliGetDriverQueryXml 이 함수는 WdsCliObtainDriverPackagesEx 함수를 사용하여 WDS 서버에서 드라이버 패키지를 쿼리하는 데 사용할 수 있는 XML 문자열을 생성합니다.
WdsCliGetEnumerationFlags 현재 이미지 핸들에 대한 이미지 열거형 플래그를 반환합니다.
WdsCliGetImageArchitecture 현재 이미지의 프로세서 아키텍처를 반환합니다.
WdsCliGetImageDescription 현재 이미지에 대한 설명을 반환합니다.
WdsCliGetImageGroup 현재 이미지에 대한 이미지 그룹의 이름을 반환합니다.
WdsCliGetImageHalName 현재 이미지의 HAL(하드웨어 추상화 계층) 이름을 반환합니다.
WdsCliGetImageHandleFromFindHandle 이미지 열거형의 현재 이미지에 대한 이미지 핸들을 반환합니다.
WdsCliGetImageHandleFromTransferHandle 완료된 전송 핸들에서 이미지 핸들을 반환합니다. 핸들은 서버에서 클라이언트로 전송된 이미지의 로컬 복사본에 대한 것입니다.

WdsCliGetImageIndex 현재 이미지에 대한 WIM(Windows 이미징 형식) 파일 내의 인덱스 반환
WdsCliGetImageLanguage 현재 이미지의 기본 언어를 반환합니다.
WdsCliGetImageLanguages 현재 이미지에서 지원하는 언어 배열을 반환합니다.
WdsCliGetImageLastModifiedTime 현재 이미지의 마지막 수정 시간을 반환합니다.
WdsCliGetImageName 현재 이미지의 이름을 반환합니다.
WdsCliGetImageNamespace 현재 이미지의 네임스페이스를 반환합니다.
WdsCliGetImagePath 현재 이미지가 포함된 파일의 경로를 반환합니다.
WdsCliGetImageSize 현재 이미지의 크기를 반환합니다.
WdsCliGetImageVersion 현재 이미지의 버전을 반환합니다.
WdsCliGetTransferSize 현재 파일 전송의 크기를 반환합니다.
WdsCliInitializeLog WDS 클라이언트에 대한 로깅을 초기화합니다.
WdsCliLog WDS 서버에 로그 이벤트를 보냅니다.

WdsCliObtainDriverPackages 이 함수는 이 컴퓨터에서 사용할 수 있는 드라이버 패키지(INF 파일)인 WDS 이미지에서 가져옵니다.
WdsCliObtainDriverPackagesEx 이 함수는 지정된 WDS 드라이버 쿼리 XML에 적용할 수 있는 드라이버 패키지(INF 파일)를 가져옵니다.
WdsCliRegisterTrace 디버깅 메시지를 받을 콜백 함수를 등록합니다.
WdsCliTransferFile 멀티캐스트 전송 프로토콜을 사용하여 WDS 서버에서 WDS 클라이언트로 파일을 전송합니다.
WdsCliTransferImage WDS 서버에서 WDS 클라이언트로 이미지를 전송합니다.
WdsCliWaitForTransfer 이미지 또는 파일 전송이 완료되기를 기다립니다.
WdsTransportClientAddRefBuffer 멀티캐스트 클라이언트가 소유한 버퍼에 대한 참조 수를 증분합니다.
WdsTransportClientCancelSession 클라이언트의 세션과 연결된 리소스를 해제합니다. (WdsTransportClientCancelSession)
WdsTransportClientCloseSession 클라이언트의 세션과 연결된 리소스를 해제합니다. (WdsTransportClientCloseSession)
WdsTransportClientCompleteReceive 데이터 블록의 모든 처리가 완료되었으며 멀티캐스트 클라이언트가 캐시에서 이 데이터 블록을 제거하여 추가 수신을 위한 공간을 확보할 수 있음을 나타냅니다.
WdsTransportClientInitialize 멀티캐스트 클라이언트를 초기화합니다.

WdsTransportClientInitializeSession 멀티캐스트 파일 전송을 시작합니다. (WdsTransportClientInitializeSession)
WdsTransportClientQueryStatus 멀티캐스트 클라이언트에서 진행 중이거나 완전한 멀티캐스트 전송의 현재 상태를 검색합니다.
WdsTransportClientRegisterCallback 멀티캐스트 클라이언트에 콜백을 등록합니다.
WdsTransportClientReleaseBuffer 멀티캐스트 클라이언트가 소유한 버퍼의 참조 수를 줄입니다.
WdsTransportClientShutdown 멀티캐스트 클라이언트를 종료합니다.
WdsTransportClientStartSession 멀티캐스트 파일 전송을 시작합니다. (WdsTransportClientStartSession)
WdsTransportClientWaitForCompletion 멀티캐스트 세션이 완료되거나 지정된 시간 제한에 도달할 때까지 차단합니다.
WdsTransportProviderCloseContent 핸들로 지정된 콘텐츠 스트림을 닫습니다.
WdsTransportProviderCloseInstance 핸들로 지정된 콘텐츠 공급자의 인스턴스를 닫습니다.
WdsTransportProviderCompareContent 지정된 콘텐츠 이름과 인스턴스를 열린 콘텐츠 스트림과 비교하여 동일한지 확인합니다.
WdsTransportProviderCreateInstance 콘텐츠 공급자의 새 인스턴스를 엽니다.
WdsTransportProviderDumpState 전송 공급자가 해당 상태의 요약 및 기타 관련 정보를 추적 로그에 출력하도록 지시합니다.

WdsTransportProviderGetContentMetadata 열려 있는 콘텐츠 스트림에 대한 메타데이터를 검색합니다.
WdsTransportProviderGetContentSize 열린 콘텐츠 스트림의 크기를 검색합니다.
WdsTransportProviderInitialize 콘텐츠 공급자를 초기화합니다.
WdsTransportProviderOpenContent 콘텐츠 스트림의 새 정적 보기를 엽니다.
WdsTransportProviderReadContent 열려 있는 콘텐츠 스트림에서 콘텐츠를 읽습니다.
WdsTransportProviderRefreshSettings 전송 공급자에게 관련 설정을 다시 읽도록 지시합니다.
WdsTransportProviderShutdown 콘텐츠 공급자를 종료합니다.
WdsTransportProviderUserAccessCheck 사용자의 토큰에 따라 콘텐츠 스트림에 대한 액세스를 지정합니다.
WdsTransportServerAllocateBuffer 메모리에 버퍼를 할당합니다.
WdsTransportServerCompleteRead 읽기 작업의 상태를 제공합니다.
WdsTransportServerFreeBuffer 버퍼에서 사용하는 메모리를 해제합니다.
WdsTransportServerRegisterCallback 멀티캐스트 서버에 공급자 콜백을 등록합니다.

WdsTransportServerTrace 디버깅 메시지를 보냅니다. (WdsTransportServerTrace)
WdsTransportServerTraceV 디버깅 메시지를 보냅니다. (WdsTransportServerTraceV)

인터페이스

IWdsTransportCacheable 영구 데이터를 처리하는 개체에 대한 캐싱을 제공합니다. 이 인터페이스는 지속형 개체를 나타내는 다른 인터페이스에서 상속할 수 있습니다.
IWdsTransportClient WDS 전송 서버의 전송 세션에 조인된 WDS 클라이언트를 나타냅니다.
IWdsTransportCollection WDS(Windows Deployment Services) 전송 관리 개체의 컬렉션을 나타냅니다.
IWdsTransportConfigurationManager WDS 전송 서버의 구성을 관리합니다.
IWdsTransportConfigurationManager2 이 인터페이스는 IWdsTransportConfigurationManager 인터페이스에서 상속되며 Windows Server 2008 R2부터 사용할 수 있는 멀티캐스트 세션 정책과 같은 구성 설정을 사용하여 확장합니다.
IWdsTransportContent 하나 이상의 세션을 통해 네임스페이스 아래에 전송되는 콘텐츠를 나타냅니다.
IWdsTransportContentProvider 콘텐츠 공급자를 설명하는 데 사용됩니다.
IWdsTransportDiagnosticsPolicy WDS 전송 서버 구성의 진단 정책 부분을 나타냅니다.

IWdsTransportManager WDS 전송 서버를 관리합니다. 이는 WDS(Windows 배포 서비스) 전송 관리 API의 최상위 인터페이스이며 CoCreateInstance 함수를 사용하여 만들 수 있는 유일한 인터페이스입니다.
IWdsTransportMulticastSessionPolicy 이 인터페이스는 WDS 전송 서버 구성의 멀티캐스트 세션 정책 부분을 나타냅니다.
IWdsTransportNamespace WDS 전송 서버의 네임스페이스를 나타냅니다.
IWdsTransportNamespaceAutoCast WDS 전송 서버는 이 인터페이스의 네임스페이스 개체에 의해 호스트되는 멀티캐스트 세션을 만들고, 시작하고, 종료할 수 있습니다.
IWdsTransportNamespaceManager WDS 전송 서버에서 네임스페이스를 관리합니다.
IWdsTransportNamespaceScheduledCast 파생 인터페이스:_IWdsTransportNamespaceScheduledCastManualStart 및 IWdsTransportNamespaceScheduledCastAutoStart에 대한 기본 인터페이스를 나타냅니다.
IWdsTransportNamespaceScheduledCastAutoStart 관리자는 IWdsTransportNamespaceScheduledCastAutoStart 인터페이스의 개체에서 자동으로 전송을 시작하는 조건을 지정할 수 있습니다. 애플리케이션은 이러한 조건에 도달할 때까지 네임스페이스 개체의 세션을 계속 조인할 수 있습니다.
IWdsTransportNamespaceScheduledCastManualStart 관리자는 IWdsTransportNamespaceScheduledCastManualStart 인터페이스의 개체에서 전송을 시작해야 합니다. 애플리케이션은 관리자가 전송을 시작할 때까지 네임스페이스의 세션에 계속 조인할 수 있습니다.
IWdsTransportServer WDS 전송 서버를 나타냅니다. WDS 클라이언트는 이 인터페이스의 개체를 사용하여 서버의 설정, 구성 및 네임스페이스 작업을 관리할 수 있습니다.
IWdsTransportServer2 이 인터페이스는 IWdsTransportServer 인터페이스에서 상속되고 확장됩니다. Windows Server 2012 시작해 사용할 수 있습니다.

IWdsTransportServicePolicy WDS 전송 서버 구성의 서비스 정책 부분을 나타냅니다.
IWdsTransportServicePolicy2 이 인터페이스는 IWdsTransportServicePolicy 인터페이스에서 상속되고 Windows Server 2012 시작합니다.
IWdsTransportSession WDS 전송 서버의 활성 전송 세션을 나타냅니다.
IWdsTransportSetupManager WDS 전송 서버에서 설치 작업을 관리합니다.
IWdsTransportSetupManager2 이 인터페이스는 IWdsTransportSetupManager 인터페이스에서 상속되고 확장됩니다. Windows Server 2012 시작해 사용할 수 있습니다.
IWdsTransportTftpClient 이 인터페이스는 서버에서 현재 활성화된 TFTP 클라이언트 세션의 정보를 지정하는 데 사용됩니다.
IWdsTransportTftpManager 이 인터페이스는 현재 TFTP 서버에 연결된 모든 클라이언트를 검색하는 메서드를 제공합니다.

구조체

PXE_ADDRESS 패킷의 네트워크 주소를 지정합니다.
PXE_DHCP_MESSAGE PXE_DHCP_MESSAGE 구조는 Windows Deployment Services PXE 서버 API와 함께 사용할 수 있습니다.

PXE_DHCP_OPTION PXE_DHCP_OPTION 구조는 Windows Deployment Services PXE 서버 API와 함께 사용할 수 있습니다.
PXE_DHCPV6_MESSAGE DHCPV6 메시지입니다.
PXE_DHCPV6_MESSAGE_HEADER PXE_DHCPV6_MESSAGE 구조체와 PXE_DHCPV6_RELAY_MESSAGE 구조 간의 공통 필드를 설명합니다.
PXE_DHCPV6_NESTED_RELAY_MESSAGE OPTION_RELAY_MSG 메시지에 중첩된 패킷에 대해 설명합니다.
PXE_DHCPV6_OPTION PXE_DHCPV6_OPTION 구조는 Windows Deployment Services PXE 서버 API와 함께 사용할 수 있습니다.
PXE_DHCPV6_RELAY_MESSAGE DHCPV6 릴레이 메시지를 제공합니다.
PXE_PROVIDER 공급자에 대해 설명합니다.
TRANSPORTCLIENT_SESSION_INFO 이 구조는 PFN_WdsTransportClientSessionStartEx 콜백 함수에서 사용됩니다.
WDS_CLI_CRED 클라이언트 세션에 권한을 부여하는 데 사용되는 자격 증명을 포함합니다.
WDS_TRANSPORTCLIENT_REQUEST 이 구조체는 WdsTransportClientStartSession 함수에서 사용됩니다.
WDS_TRANSPORTPROVIDER_INIT_PARAMS 이 구조는 WdsTransportProviderInitialize 콜백 함수에서 사용됩니다. (WDS_TRANSPORTPROVIDER_INIT_PARAMS)

WDS_TRANSPORTPROVIDER_SETTINGS 이 구조는 WdsTransportProviderInitialize 콜백 함수에서 사용됩니다. (WDS_TRANSPORTPROVIDER_SETTINGS)

Windows 데스크톱 공유

아티클2023. 03. 09.

Windows 데스크톱 공유 기술 개요입니다.

Windows 데스크톱 공유를 개발하려면 다음 헤더가 필요합니다.

- rdpencomapi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 데스크톱 공유

열거형

ATTENDEE_DISCONNECT_REASON 참석자가 세션에서 연결이 끊어진 이유에 대한 값을 정의합니다.
CHANNEL_ACCESS_ENUM 채널에 대해 참석자에게 부여된 액세스 유형에 대한 값을 정의합니다.
CHANNEL_FLAGS 채널에서 데이터를 보내는 방법에 대한 값을 정의합니다.
CHANNEL_PRIORITY 채널에서 패킷을 보내는 데 사용되는 우선 순위에 대한 값을 정의합니다.
CTRL_LEVEL 참석자가 세션에 대해 갖는 제어 수준을 정의합니다.
RDPENCOMAPI_ATTENDEE_FLAGS 참석자 유형에 대한 값을 정의합니다.
RDPSRAPI_APP_FLAGS 애플리케이션 유형에 대한 값을 정의합니다.

RDPSRAPI_KBD_CODE_TYPE 키보드 인코딩 유형에 대한 값을 정의합니다.
RDPSRAPI_KBD_SYNC_FLAG Caps Lock 키와 같이 키보드의 상태를 제어하는 키 유형에 대한 값을 정의합니다.
RDPSRAPI_MOUSE_BUTTON_TYPE 마우스 단추 형식에 대한 값을 정의합니다.
RDPSRAPI_WND_FLAGS 창 형식에 대한 값을 정의합니다.

Functions

AddTouchInput 터치 입력에 대한 설명을 허용합니다.
AllocBuffer 스트림 버퍼를 할당하기 위해 RDP(원격 데스크톱 프로토콜) 스택에서 호출됩니다.
BeginTouchFrame 일련의 터치 입력을 수락하기 시작합니다.
닫기 세션에서 모든 참석자의 연결을 끊고 들어오는 연결의 수신을 중지합니다.
닫기 RDP(원격 데스크톱 프로토콜) 스택에서 호출되어 스트림을 닫습니다.
연결 공유자의 실제 연결을 시작합니다.
ConnectToClient 공유자가 뷰어에 연결하는 역방향 연결 모드에 사용됩니다.

ConnectUsingTransportStream 지정된 전송 스트림을 사용하여 연결합니다.
CreateInvitation 초대를 만듭니다.
CreateLogger 새 IRDPSRAPIPerfCounterLogger 개체를 만듭니다.
CreateVirtualChannel 가상 채널을 만듭니다.
연결 끊기 공유자에서 뷰어의 연결을 끊습니다.
EndTouchFrame 일련의 터치 입력을 수락하지 않습니다.
FreeBuffer GetBuffer 메서드가 호출된 후 버퍼의 보류를 해제합니다.
FreeBuffer 스트림 버퍼를 스트림에 반환하기 위해 RDP(원격 데스크톱 프로토콜) 스택에서 호출됩니다.
get__NewEnum 애플리케이션 컬렉션에 대한 열거자 인터페이스입니다.
get__NewEnum 참석자 컬렉션에 대한 열거자 인터페이스입니다.
get__NewEnum 초대 컬렉션에 대한 열거자 인터페이스입니다.
get__NewEnum 가상 채널 컬렉션에 대한 열거자 인터페이스입니다.

get__NewEnum 창 컬렉션에 대한 열거자 인터페이스입니다.
get_Application 창이 속한 애플리케이션 개체에 대한 포인터를 반환합니다.
get_ApplicationFilter IRDPSRAPIApplicationFilter 인터페이스를 구현하는 개체를 반환합니다. (IRDPSRAPISharingSession.get_ApplicationFilter)
get_ApplicationFilter IRDPSRAPIApplicationFilter 인터페이스를 구현하는 개체를 반환합니다. (IRDPSRAPIViewer.get_ApplicationFilter)
get_Applications 공유 가능한 애플리케이션 목록입니다.
get_Attendee 연결이 끊긴 참석자입니다.
get_AttendeeLimit 세션에 연결할 수 있는 최대 참석자 수입니다. (Get)
get_Attendees IRDPSRAPIAttendeeManager 인터페이스를 구현하는 개체를 반환합니다. (IRDPSRAPISharingSession.get_Attendees)
get_Attendees IRDPSRAPIAttendeeManager 인터페이스를 구현하는 개체를 반환합니다. (IRDPSRAPIViewer.get_Attendees)
get_Bpp 프레임 버퍼의 픽셀당 비트입니다.
get_Code 연결 끊기 작업의 상태입니다.

get_ColorDepth 공유 세션의 색 깊이를 검색하거나 설정합니다. (Get)
get_ConnectionString 티켓 문자열입니다.
get_ConnectivityInfo 참석자의 연결 정보를 검색합니다.
get_Context 이 속성은 RDP(원격 데스크톱 프로토콜) 스택에서 사용하도록 예약되어 있습니다. 수정하지 마십시오. (Get)
get_ControlLevel 참석자가 세션에 대해 가지고 있는 제어 수준입니다. (Get)
get_Count 컬렉션의 초대 수입니다.
get_DisconnectedText 연결이 종료되기 전에 컨트롤의 가운데에 표시되는 텍스트를 검색하거나 설정합니다. (Get)
get_Enabled 애플리케이션 필터의 사용 상태입니다. (Get)
get_Flags 공유 플래그입니다.
get_Flags RDPENCOMAPI_ATTENDEE_FLAGS 열거형 형식에 정의된 참석자별 플래그를 검색합니다.
get_Flags 스트림 버퍼에 대한 옵션을 검색하거나 지정합니다. (Get)
get_Flags 채널 플래그입니다. 이 속성은 나중에 사용할 수 있도록 예약됩니다.

get_Flags 현재 창의 플래그를 반환합니다.
get_FrameBuffer 이 세션의 프레임 버퍼를 가져옵니다.
get_GroupName 그룹 이름입니다.
get_Height 프레임 버퍼의 높이(픽셀)입니다.
get_Id 응용 프로그램 식별자입니다.
get_Id 참석자의 고유 식별자입니다.
get_Id 창의 ID를 반환합니다.
get_Invitation 참석자에게 회의에 대한 액세스 권한을 부여하는 데 사용되는 초대입니다.
get_Invitations IRDPSRAPIInvitationManager 인터페이스를 구현하는 개체를 반환합니다. (IRDPSRAPISharingSession.get_Invitations)
get_Invitations IRDPSRAPIInvitationManager 인터페이스를 구현하는 개체를 반환합니다. (IRDPSRAPIViewer.get_Invitations)
get_Item 애플리케이션 컬렉션의 항목입니다.
get_Item 참석자 컬렉션의 항목입니다.

get_Item 초대 컬렉션의 항목입니다.
get_Item 가상 채널 컬렉션의 항목입니다.
get_Item 창 컬렉션의 항목입니다.
get_LocalIP 공유자 또는 뷰어에서 통신을 위해 사용하는 로컬 IP 주소를 검색합니다.
get_LocalPort 공유자 또는 뷰어가 통신하는 데 사용하는 로컬 포트를 검색합니다.
get_Name 애플리케이션 이름입니다.
get_Name 채널 이름입니다.
get_Name 창 개체의 이름을 반환합니다.
get_Password 암호 문자열입니다.
get_PayloadOffset 콘텐츠가 시작되는 스트림 버퍼의 위치를 지정합니다. (Get)
get_PayloadSize 내부 스토리지 버퍼의 콘텐츠 크기(바이트)입니다. (Get)
get_PeerIP 공유자 또는 뷰어가 통신하는 데 사용하는 원격 IP 주소를 검색합니다.

get_PeerPort 공유자 또는 뷰어가 통신하는 데 사용되는 원격 포트를 검색합니다.
get_Priority 채널 우선 순위입니다.
get_Properties IRDPSRAPISessionProperties 인터페이스를 구현하는 개체를 검색합니다.
get_Properties IRDPSRAPISessionProperties 인터페이스를 구현하는 개체를 반환합니다.
get_Property 명명된 세션 속성을 설정하거나 가져옵니다. (Get)
get_Protocol 공유자 또는 뷰어에서 통신을 위해 사용하는 프로토콜을 검색합니다.
get_Reason 참석자의 연결이 끊어진 이유입니다.
get_RemoteName 원격 클라이언트의 이름입니다. 일반적으로 참석자의 식별 이름입니다.
get_Revoked 초대의 취소된 상태입니다. (Get)
get_Shared 공유 상태입니다. (Get)
get_Shared 창의 공유 속성을 가져오거나 설정합니다. (Get)
get_SmartSizing 뷰어 ActiveX 컨트롤의 스마트 크기 조정 속성을 가져오거나 설정합니다. (Get)

get_Storage 스트림에 대한 내부 스토리지 버퍼의 주소입니다.
get_StorageSize 스트림에 대한 내부 스토리지 버퍼의 크기(바이트)입니다.
get_VirtualChannelManager IRDPSRAPIVirtualChannelManager 인터페이스를 구현하는 개체를 검색합니다.
get_VirtualChannelManager IRDPSRAPIVirtualChannelManager 인터페이스를 구현하는 개체를 반환합니다.
get_Width 프레임 버퍼의 너비(픽셀)입니다.
get_Windows 창 목록입니다.
get_Windows 공유 가능한 창 목록입니다.
GetBuffer 버퍼에서 오디오 데이터를 가져옵니다.
GetDesktopSharedRect 공유 중인 현재 데스크톱 지역을 검색합니다.
GetFrameBufferBits 프레임의 지정된 영역에 있는 비트를 가져옵니다.
초기화 오디오 스트림을 초기화합니다.
LogValue 값을 기록합니다.

OnPasteFromClipboard 이 콜백은 공유자 컴퓨터에서 데이터를 복사하려고 할 때 발생합니다.
OnReadCompleted RDP(원격 데스크톱 프로토콜) 스택에 읽기 작업이 완료되었음을 알 수 있습니다.
OnStreamClosed RDP(원격 데스크톱 프로토콜) 스택에 연결이 닫혔다는 것을 알 수 있습니다.
OnWriteCompleted 쓰기 작업이 완료되었음을 RDP(원격 데스크톱 프로토콜) 스택에 알합니다.
열기 세션을 활성 상태로 설정합니다.
열기 RDP(원격 데스크톱 프로토콜) 스택에서 호출되어 스트림을 시작하고 RDP 스택이 이벤트 알림을 받을 준비가 되었음을 나타냅니다.
일시 중지 IRDPSRAPISharingSession::Resume가 호출될 때까지 공유자에서 모든 뷰어로 전송되는 그래픽 스트림을 일시 중지합니다.
put_AttendeeLimit 세션에 연결할 수 있는 최대 참석자 수입니다. (Put)
put_ColorDepth 공유 세션의 색 깊이를 검색하거나 설정합니다. (Put)
put_Context 이 속성은 RDP(원격 데스크톱 프로토콜) 스택에서 사용하도록 예약되어 있습니다. 수정하지 마십시오. (Put)
put_ControlLevel 참석자가 세션에 대해 가지고 있는 제어 수준입니다. (Put)

put_DisconnectedText 연결이 종료되기 전에 컨트롤의 가운데에 표시되는 텍스트를 검색하거나 설정합니다. (Put)
put_Enabled 애플리케이션 필터의 사용 상태입니다. (Put)
put_Flags 스트림 버퍼에 대한 옵션을 검색하거나 지정합니다. (Put)
put_PayloadOffset 콘텐츠가 시작되는 스트림 버퍼의 위치를 지정합니다. (Put)
put_PayloadSize 내부 스토리지 버퍼의 콘텐츠 크기(바이트)입니다. (Put)
put_Property 명명된 세션 속성을 설정하거나 가져옵니다. (Put)
put_Revoked 초대의 취소된 상태입니다. (Put)
put_Shared 공유 상태입니다. (Put)
put_Shared 창의 공유 속성을 가져오거나 설정합니다. (Put)
put_SmartSizing 뷰어 ActiveX 컨트롤의 스마트 크기 조정 속성을 가져오거나 설정합니다. (Put)
ReadBuffer 스트림 버퍼의 내용을 읽기 위해 RDP(원격 데스크톱 프로토콜) 스택에서 호출됩니다.
RequestColorDepthChange 공유자 Winlogon 사용자 세션에서 색 깊이 변경을 요청합니다.

RequestControl 공유자가 뷰어의 제어 수준을 변경하도록 요청합니다.
재개 IRDPSRAPISharingSession::P ause 또는 IRDPSRAPISharingSession::Close가 호출될 때까지 공유자에서 모든 뷰어로 전송되는 그래픽 스트림이 다시 시작되도록 합니다.
SendControlLevelChangeResponse OnControlLevelChangeResponse 이벤트를 보냅니다.
SendData 채널에서 데이터를 보냅니다.
SendKeyboardEvent 키보드 이벤트 메시지를 보냅니다.
SendMouseButtonEvent 마우스 단추 이벤트 메시지를 보냅니다.
SendMouseMoveEvent 마우스 이동 이벤트 메시지를 보냅니다.
SendMouseWheelEvent 마우스 휠 이벤트 메시지를 보냅니다.
SendSyncEvent Caps Lock 키를 누를 때와 같이 키보드 상태가 변경되었음을 나타내는 이벤트 메시지를 보냅니다.
SetAccess 참석자에 대한 채널을 사용하도록 설정합니다.
SetDesktopSharedRect 공유할 데스크톱 지역을 설정합니다.
표시 현재 창을 포그라운드로 가져옵니다.

시작 오디오 스트림을 시작합니다.
StartReverseConnectListener 공유자에서 뷰어로의 역방향 연결을 수락하기 위해 수신기를 시작하거나 공유자가 뷰어에 역방향 연결을 사용하는 연결 문자열을 가져옵니다.
중지 오디오 스트림을 중지합니다.
TerminateConnection 참석자가 나타내는 클라이언트의 연결을 끊습니다.
WriteBuffer RDP(원격 데스크톱 프로토콜) 스택에서 호출되어 스트림 버퍼의 콘텐츠를 네트워크에 씁니다.

인터페이스

_IRDPSessionEvents 이벤트가 발생할 때 알림을 받도록 이 인터페이스를 구현합니다.
IRDPSRAPIApplication 프로세스 내에서 공유 가능한 창을 그룹화합니다. 각 애플리케이션 개체에는 창 개체 목록이 포함됩니다. 애플리케이션 개체가 공유되면 모든 창이 공유됩니다.
IRDPSRAPIApplicationFilter 창 및 프로세스 수준에서 공유 데스크톱 영역을 관리합니다. 애플리케이션은 열거자를 사용하여 공유할 수 있는 세션의 개체 목록을 표시할 수 있습니다.
IRDPSRAPIApplicationList 애플리케이션 목록을 관리합니다.
IRDPSRAPIAttendee 참석자 개체는 세션에 연결하고 인증된 클라이언트의 결과로 만들어집니다. 참석자 개체가 만들어지면 참석자 목록에 자동으로 추가됩니다.

IRDPSRAPIAttendeeDisconnectInfo 참석자의 연결이 끊어진 이유에 대한 정보를 포함합니다.
IRDPSRAPIAttendeeManager 참석자 개체를 관리합니다.
IRDPSRAPIAudioStream 공동 작업 공유자 Microsoft ActiveX 컨트롤에서 공동 작업 뷰어 컨트롤로 오디오 스트림을 보낼 수 있습니다.
IRDPSRAPIClipboardUseEvents 공유자 쪽에서 이 인터페이스를 구현하여 클립보드 사용을 추적하거나 제어합니다. 클립보드 공유를 사용하도록 설정하지 않으면 이 인터페이스가 적용되지 않습니다. 속성에 설명된 SetClipboardRedirectCallback 속성에 대한 값을 설정해야 합니다.
IRDPSRAPIFrameBuffer 프레임 버퍼 크기와 형식에 대한 데이터를 제공하고 콘텐츠를 검색할 수 있도록 합니다.
IRDPSRAPIInvitation 초대를 사용하면 사용자 또는 사용자 그룹이 세션에 연결할 수 있습니다. 참석자가 세션에 연결되면 클라이언트는 티켓과 암호를 보냅니다. 이 두 가지 정보는 참석자를 인증하는 데 사용됩니다.
IRDPSRAPIInvitationManager 초대 개체를 관리합니다.
IRDPSRAPIPerfCounterLogger 클라이언트 애플리케이션이 사용자 지정 성능 로깅을 구현할 수 있도록 합니다.
IRDPSRAPIPerfCounterLoggingManager IRDPSRAPIPerfCounterLogger 개체를 관리합니다.
IRDPSRAPISessionProperties 이 인터페이스를 사용하여 세션 속성을 얻거나 설정합니다.
IRDPSRAPISharingSession 협업 세션을 시작하기 위해 애플리케이션에서 만들어야 하는 기본 개체입니다. (IRDPSRAPISharingSession)

IRDPSRAPISharingSession2 협업 세션을 시작하기 위해 애플리케이션에서 만들어야 하는 기본 개체입니다. (IRDPSRAPISharingSession2)
IRDPSRAPITcpConnectionInfo 뷰어 및 공유자 쪽에서 TCP 연결 정보를 검색하는 메서드를 지원합니다.
IRDPSRAPITransportStream 스트림을 사용하여 작업을 수행하는 메서드를 노출합니다.
IRDPSRAPITransportStreamBuffer 데이터를 보내고 받기 위해 IRDPSRAPITransportStream 인터페이스에서 만들고 사용합니다.
IRDPSRAPITransportStreamEvents 스트림 인터페이스(IRDPSRAPITransportStream)에서 호출된 메서드를 노출하여 RDP(원격 데스크톱 프로토콜) 스택에 이벤트 완료를 알립니다.
IRDPSRAPIViewer 뷰어 쪽에서 사용되는 ActiveX 인터페이스입니다.
IRDPSRAPIVirtualChannel 가상 채널을 관리합니다.
IRDPSRAPIVirtualChannelManager 가상 채널 목록을 관리합니다.
IRDPSRAPIWindow 공유 가능한 창에 대한 일대일 매핑을 나타냅니다.
IRDPSRAPIWindowList 창 목록을 관리합니다.
IRDPViewerInputSink 마우스 및 키보드 이벤트를 보내고 터치 입력을 지원합니다.

Windows 오류 보고

아티클2023. 07. 28.

Windows 오류 보고 기술에 대한 개요입니다.

Windows 오류 보고 개발하려면 다음 헤더가 필요합니다.

- errorrep.h
- werapi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 오류 보고

열거형

REPORT_STORE_TYPES 열 수 있는 WER(Windows 오류 보고) 보고서 저장소 유형을 정의합니다.
WER_CONSENT WER(가능한 Windows 오류 보고) 사용자 동의 상태를 정의합니다.
WER_DUMP_TYPE WER(가능한 Windows 오류 보고) 미니덤프 형식을 정의합니다.
WER_FILE_TYPE 미니덤프 보고서에 대한 가능한 Windows 오류 보고(WER) 파일 형식을 정의합니다.
WER_REGISTER_FILE_TYPE 등록할 수 있는 WINDOWS 오류 보고(WER) 파일 형식을 정의합니다.
WER_REPORT_TYPE WER(Windows 오류 보고) 보고서 형식을 정의합니다.
WER_REPORT_UI WER(Windows 오류 보고) 보고서의 사용자 지정 가능한 UI 문자열의 인덱스를 정의합니다.

WER_SUBMIT_RESULT WER(Windows 오류 보고) 제출 결과 옵션을 정의합니다.

Functions

AddERExcludedApplicationA 지정된 애플리케이션을 오류 보고에서 제외합니다. (ANSI)
AddERExcludedApplicationW 지정된 애플리케이션을 오류 보고에서 제외합니다. (유니코드)
PFN_WER_RUNTIME_EXCEPTION_DEBUGGER_LAUNCH WER(Windows 오류 보고)은 이 함수를 호출하여 디버거 시작 옵션 및 시작 문자열을 사용자 지정할 수 있도록 합니다.
PFN_WER_RUNTIME_EXCEPTION_EVENT WER(Windows 오류 보고)은 이 함수를 호출하여 예외 처리기가 충돌을 클레임하는지 여부를 확인합니다.
PFN_WER_RUNTIME_EXCEPTION_EVENT_SIGNATURE WER(Windows 오류 보고)은 이 함수를 호출하여 문제를 고유하게 설명하는 보고서 매개 변수를 가져옵니다.
ReportFault 자체 예외 처리를 수행하는 애플리케이션이 Microsoft에 오류를 보고할 수 있도록 합니다.
WerAddExcludedApplication 지정된 애플리케이션을 오류 보고에서 제외할 애플리케이션 목록에 추가합니다.
WerFreeString 보고서 키 문자열을 저장하는 데 사용되는 메모리를 해제합니다. 특정 보고서 키 문자열이 사용되고 더 이상 필요하지 않은 경우 WerStoreGetFirstReportKey 또는 WerStoreGetNextReportKey에 대한 각 연속 호출 후에 호출해야 합니다.
WerGetFlags

지정된 프로세스에 대한 오류 보고 설정을 검색합니다.
WerRegisterAdditionalProcess 기본 애플리케이션 프로세스와 함께 WER(Windows 오류 보고) 보고서에 포함할 프로세스를 등록합니다. 필요에 따라 추가 데이터를 가져올 등록된 프로세스 내의 스레드를 지정합니다.
WerRegisterAppLocalDump 애플리케이션에 대한 프로세스 중 하나가 응답을 중지할 때 WINDOWS 오류 보고(WER)에서 수집하는 진단 메모리 덤프의 복사본을 저장해야 하는 경로(패키지된 애플리케이션의 LocalFolder를 기준으로)를 등록합니다.
WerRegisterCustomMetadata WER(Windows 오류 보고) 오류 보고서에 대해 수집할 앱별 메타데이터(키/값 문자열 형식)를 등록합니다.
WerRegisterExcludedMemoryBlock WER(Windows 오류 보고) 오류 보고서에서 제외할 메모리 블록(일반적으로 오류 보고서에 기본적으로 포함됨)을 표시합니다.
WerRegisterFile Windows 오류 보고(WER)에서 오류 보고서를 만들 때 수집할 파일을 등록합니다.
WerRegisterMemoryBlock WINDOWS 오류 보고(WER)에서 오류 보고서를 만들 때 수집할 메모리 블록을 등록합니다.
WerRegisterRuntimeExceptionModule 크래시에 대한 WER(사용자 지정 Windows 오류 보고)을 제공하는 데 사용되는 사용자 지정 런타임 예외 처리기를 등록합니다.
WerRemoveExcludedApplication WER(Windows 오류 보고)에서 제외할 애플리케이션 목록에서 지정된 애플리케이션을 제거합니다.
WerReportAddDump 지정된 형식의 덤프를 지정된 WER(Windows 오류 보고) 보고서에 추가합니다.
WerReportAddFile 지정된 WER(Windows 오류 보고) 보고서에 파일을 추가합니다.
WerReportCloseHandle

지정된 WER(Windows 오류 보고) 보고서를 닫습니다.
WerReportCreate 애플리케이션 이벤트를 설명하는 WER(Windows 오류 보고) 보고서를 만듭니다.
WerReportHang 지정된 창에서 "응답 없음" 보고를 시작합니다.
WerReportSetParameter 지정된 WER(Windows 오류 보고) 보고서에 대한 이벤트를 고유하게 식별하는 매개 변수를 설정합니다.
WerReportSetUIOption 지정된 WER(Windows 오류 보고) 보고서에 대한 사용자 인터페이스 옵션을 설정합니다.
WerReportSubmit 지정된 WER(Windows 오류 보고) 보고서를 제출합니다.
WerSetFlags 현재 프로세스에 대한 WER(Windows 오류 보고) 보고 설정을 설정합니다.
WerSetMaxProcessHoldMilliseconds WER(Windows 오류 보고)의 최대 프로세스 보류 시간을 밀리초 단위로 설정합니다.
WerStoreClose WER(저장된 Windows 오류 보고) 보고서의 컬렉션을 닫습니다.
WerStoreGetFirstReportKey 보고서 저장소의 첫 번째 WER(Windows 오류 보고) 보고서에 대한 참조를 가져옵니다.
WerStoreGetNextReportKey 오류 보고서 저장소의 다음 WER(Windows 오류 보고) 보고서에 대한 참조를 가져옵니다.
WerStoreGetReportCount 오류 보고서 저장소의 WER(Windows 오류 보고) 보고서 수를 가져옵니다.
WerStoreGetSizeOnDisk WER(Windows 오류 보고) 오류 보고서 저장소의 크기를 바이트 단위로 가져옵니다.

WerStoreOpen 저장된 WINDOWS 오류 보고(WER) 오류 보고서의 컬렉션을 엽니다.
WerStorePurge WER(Windows 오류 보고) 보고서 저장소를 제거합니다.
WerStoreQueryReportMetadataV1 저장소의 WER(Windows 오류 보고) 보고서에 대한 메타데이터를 검색합니다.
WerStoreQueryReportMetadataV2 저장소의 WER(Windows 오류 보고) 보고서에 대한 메타데이터를 검색합니다.
WerStoreQueryReportMetadataV3 저장소의 WER(Windows 오류 보고) 보고서에 대한 메타데이터를 검색합니다.
WerStoreUploadReport WER(Windows 오류 보고) 저장소에 보고서를 업로드합니다.
WerUnregisterAdditionalProcess WINDOWS 오류 보고(WER) 오류 보고서에 포함할 추가 프로세스 목록에서 프로세스를 제거합니다.
WerUnregisterAppLocalDump WerRegisterAppLocalDump 함수를 호출하여 WER(Windows 오류 보고)이 애플리케이션에 대한 프로세스 중 하나가 응답을 중지할 때 수집한 진단 메모리 덤프의 복사본을 저장하도록 지정하여 만든 등록을 취소합니다.
WerUnregisterCustomMetadata 애플리케이션에 대한 WINDOWS 오류 보고(WER) 중에 수집되는 앱별 메타데이터의 항목을 제거합니다.
WerUnregisterExcludedMemoryBlock 이전에 제외된 것으로 표시된 메모리 블록을 제거합니다. 이 블록은 WER(Windows 오류 보고] 오류 보고서에 다시 포함됩니다.
WerUnregisterFile 현재 프로세스에 대해 생성된 WER(Windows 오류 보고) 보고서에 추가할 파일 목록에서 파일을 제거합니다.

WerUnregisterMemoryBlock 애플리케이션에 대한 WINDOWS 오류 보고(WER) 중에 수집할 데이터 목록에서 메모리 블록을 제거합니다.
WerUnregisterRuntimeExceptionModule WER(Windows 오류 보고) 예외 처리기의 등록을 제거합니다.

구조체

WER_DUMP_CUSTOM_OPTIONS WerReportAddDump 함수에서 수집할 WER(사용자 지정 Windows 오류 보고) 미니덤프 정보를 지정합니다.
WER_DUMP_CUSTOM_OPTIONS_V2 WerReportAddDump 함수에서 수집할 WER(사용자 지정 Windows 오류 보고) 미니덤프 정보를 지정합니다.
WER_DUMP_CUSTOM_OPTIONS_V3 PssCaptureSnapshot 함수를 통해 백그라운드에서 수집할 WER(사용자 지정 Windows 오류 보고) 미니덤프 정보를 지정합니다(프로세스를 일시 중지하지 않음).
WER_EXCEPTION_INFORMATION WerReportAddDump 함수에 대한 WER(Windows 오류 보고) 예외 정보를 포함합니다.
WER_REPORT_INFORMATION WerReportCreate 함수에서 사용하는 WER(Windows 오류 보고) 정보를 포함합니다.
WER_REPORT_INFORMATION_V3 WerReportCreate 함수에서 사용하는 WER(Windows 오류 보고) 정보를 포함합니다.
WER_REPORT_INFORMATION_V4 WerReportCreate 함수에서 사용하는 WER(Windows 오류 보고] 정보를 포함합니다.
WER_REPORT_INFORMATION_V5 WerReportCreate 함수에서 사용하는 WER(Windows 오류 보고) 정보를 포함합니다.

WER_REPORT_METADATA_V1 WER(Windows 오류 보고)에서 생성된 오류 보고서에 대한 정보를 포함합니다.
WER_REPORT_METADATA_V2 WER(Windows 오류 보고)에서 생성된 오류 보고서에 대한 정보를 포함합니다.
WER_REPORT_METADATA_V3 WER(Windows 오류 보고)에서 생성된 오류 보고서에 대한 정보를 포함합니다.
WER_REPORT_PARAMETER WER(Windows 오류 보고)에서 생성된 오류 보고서 매개 변수에 대한 정보를 포함합니다.
WER_REPORT_SIGNATURE WINDOWS 오류 보고(WER)에서 생성된 오류 보고서 서명에 대한 정보를 포함합니다.
WER_RUNTIME_EXCEPTION_INFORMATION 크래시 클레임 여부를 결정하는 데 사용되는 WER(Windows 오류 보고) 예외 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Windows 이벤트 수집기

아티클2023. 03. 11.

Windows 이벤트 수집기 기술 개요입니다.

Windows 이벤트 수집기를 개발하려면 다음 헤더가 필요합니다.

- evcoll.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 이벤트 수집기

열거형

EC_SUBSCRIPTION_CONFIGURATION_MODE 구독의 기본 설정을 변경하는 다양한 구성 모드를 지정합니다.
EC_SUBSCRIPTION_CONTENT_FORMAT 이벤트가 이벤트 수집기 컴퓨터로 전송되기 전에 이벤트를 보내는 컴퓨터에서 이벤트를 렌더링하는 방법을 지정합니다.
EC_SUBSCRIPTION_CREDENTIALS_TYPE 이벤트 원본과 통신할 때 사용할 자격 증명 유형을 지정합니다.
EC_SUBSCRIPTION_DELIVERY_MODE 구독에서 이벤트가 전달되는 방법을 나타내는 값을 정의합니다.
EC_SUBSCRIPTION_PROPERTY_ID 구독 구성에 사용되는 이벤트 구독 속성을 식별하는 값을 정의합니다.
EC_SUBSCRIPTION_RUNTIME_STATUS_ACTIVE_STATUS 구독과 관련하여 구독 또는 이벤트 원본의 상태를 지정합니다.
EC_SUBSCRIPTION_RUNTIME_STATUS_INFO_ID EC_SUBSCRIPTION_RUNTIME_STATUS_INFO_ID 열거형은 구독 상태 또는 구독과 관련된 특정 이벤트 원본의 상태를 가져오는 데 사용되는 값을 지정합니다.

EC_SUBSCRIPTION_TYPE 사용할 구독 유형(원본 시작 또는 수집기 시작 구독)을 지정합니다.
EC_VARIANT_TYPE EC_VARIANT_TYPE 열거형은 Windows 이벤트 수집기 함수에 사용되는 데이터 형식을 지정하는 값을 정의합니다.

함수

EcClose 다른 이벤트 수집기 함수에서 받은 핸들을 닫습니다.
EcDeleteSubscription 기존 구독을 삭제합니다.
EcEnumNextSubscription 로컬 컴퓨터에 등록된 구독의 열거를 계속합니다.
EcGetObjectArrayProperty 구독의 이벤트 원본에 대한 속성 값을 검색합니다.
EcGetObjectArraySize 구독의 이벤트 원본에 대한 속성 값 배열의 인덱스 수를 검색합니다.
EcGetSubscriptionProperty 구독 개체에서 속성 값을 검색합니다.
EcGetSubscriptionRunTimeStatus 구독 또는 구독 자체의 이벤트 원본에 대한 런타임 상태 정보를 검색합니다.
EcInsertObjectArrayElement 구독의 이벤트 원본에 대한 속성 값 배열에 빈 개체를 삽입합니다.

EcOpenSubscription 기존 구독을 열거나 새 구독을 만듭니다.
EcOpenSubscriptionEnum 로컬 컴퓨터에서 등록된 모든 구독을 열거하는 구독 열거자를 만듭니다.
EcRemoveObjectArrayElement 구독의 이벤트 원본에 대한 속성 값을 포함하는 개체 배열에서 요소를 제거합니다.
EcRetrySubscription 연결되지 않은 구독의 이벤트 원본에 대한 연결을 다시 시도합니다.
EcSaveSubscription 구독 구성 정보를 저장합니다.
EcSetObjectArrayProperty 구독의 이벤트 원본에 대한 속성 값 배열의 속성 값을 설정합니다.
EcSetSubscriptionProperty 새 값을 설정하거나 구독의 기존 값을 업데이트합니다.

구조체

EC_VARIANT 이벤트 수집기 데이터(구독 데이터) 또는 속성 값을 포함합니다.

Windows 이벤트 로그

아티클2023. 03. 11.

Windows 이벤트 로그 기술 개요입니다.

Windows 이벤트 로그를 개발하려면 다음 헤더가 필요합니다.

- winevt.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 이벤트 로그

열거형

EVT_CHANNEL_CLOCK_TYPE 이벤트 채널을 로깅할 때 사용할 타임스탬프를 지정하는 값을 정의합니다.
EVT_CHANNEL_CONFIG_PROPERTY_ID 채널의 구성 속성을 식별하는 식별자를 정의합니다.
EVT_CHANNEL_ISOLATION_TYPE 채널에 적용할 기본 액세스 권한을 정의합니다.
EVT_CHANNEL_REFERENCE_FLAGS 채널을 참조하는 방법을 지정하는 값을 정의합니다.
EVT_CHANNEL_SID_TYPE 이벤트에 이벤트를 기록한 보안 주체의 SID(보안 식별자)가 포함되는지 여부를 결정하는 값을 정의합니다.
EVT_CHANNEL_TYPE 채널의 형식을 정의합니다.
EVT_EVENT_METADATA_PROPERTY_ID 이벤트 정의의 메타데이터 속성을 식별하는 식별자를 정의합니다.

EVT_EVENT_PROPERTY_ID 검색할 쿼리 정보를 결정하는 값을 정의합니다.
EVT_EXPORTLOG_FLAGS 이벤트가 채널 또는 로그 파일에서 제공되는지 여부를 나타내는 값을 정의합니다.
EVT_FORMAT_MESSAGE_FLAGS 이벤트의 메시지 문자열을 형식으로 지정하는 값을 정의합니다.
EVT_LOG_PROPERTY_ID 채널 또는 로그 파일의 로그 파일 메타데이터 속성을 식별하는 식별자를 정의합니다.
EVT_LOGIN_CLASS 원격 컴퓨터에 연결하는 데 사용할 수 있는 연결 방법 유형을 정의합니다.
EVT_OPEN_LOG_FLAGS 채널 또는 내보낸 로그 파일을 열지 여부를 지정하는 값을 정의합니다.
EVT_PUBLISHER_METADATA_PROPERTY_ID 공급자의 메타데이터 속성을 식별하는 식별자를 정의합니다.
EVT_QUERY_FLAGS 쿼리 결과를 반환하는 방법과 채널 또는 로그 파일에 대해 쿼리하는지 여부를 지정하는 값을 정의합니다.
EVT_QUERY_PROPERTY_ID 검색할 수 있는 쿼리 정보를 식별하는 식별자를 정의합니다.
EVT_RENDER_CONTEXT_FLAGS 이벤트에서 액세스할 정보 유형을 지정하는 값을 정의합니다.
EVT_RENDER_FLAGS 렌더링할 항목을 지정하는 값을 정의합니다.
EVT_RPC_LOGIN_FLAGS 원격 컴퓨터에 연결할 때 사용자를 인증하는 데 사용할 수 있는 인증 유형을 정의합니다.

EVT_SEEK_FLAGS 검색할 결과 집합의 상대 위치를 정의합니다.
EVT_SUBSCRIBE_FLAGS 이벤트 구독을 시작할 시기를 지정하는 가능한 값을 정의합니다.
EVT_SUBSCRIBE_NOTIFY_ACTION 구독 서비스가 콜백에 제공할 수 있는 가능한 데이터 형식을 정의합니다.
EVT_SYSTEM_PROPERTY_ID 이벤트의 시스템별 속성을 식별하는 식별자를 정의합니다.
EVT_VARIANT_TYPE 변형 데이터 항목의 가능한 데이터 형식을 정의합니다.

Functions

EVT_SUBSCRIBE_CALLBACK EvtSubscribe 함수를 호출하여 쿼리와 일치하는 이벤트를 수신하는 경우 이 콜백을 구현합니다.
EvtArchiveExportedLog 지정된 로그 파일의 이벤트에 지역화된 문자열을 추가합니다.
EvtCancel 핸들에서 보류 중인 모든 작업을 취소합니다.
EvtClearLog 지정된 채널에서 모든 이벤트를 제거하고 대상 로그 파일에 씁니다.
EvtClose 열린 핸들을 닫습니다.
EvtCreateBookmark 채널에서 이벤트를 식별하는 책갈피를 만듭니다.

EvtCreateRenderContext 렌더링하려는 이벤트의 정보를 지정하는 컨텍스트를 만듭니다.
EvtExportLog 지정된 채널 또는 로그 파일에서 이벤트를 복사하고 대상 로그 파일에 씁니다.
EvtFormatMessage 메시지 문자열의 서식을 지정합니다. (EvtFormatMessage)
EvtGetChannelConfigProperty 지정된 채널 구성 속성을 가져옵니다.
EvtGetEventInfo 이벤트를 선택한 구조적 XML 쿼리와 이벤트를 포함하는 채널 또는 로그 파일을 식별하는 정보를 가져옵니다.
EvtGetEventMetadataProperty 지정된 이벤트 메타데이터 속성을 가져옵니다.
EvtGetExtendedStatus 현재 오류에 대한 확장된 오류 정보가 포함된 문자 메시지를 가져옵니다.
EvtGetLogInfo 채널 또는 로그 파일에 대한 정보를 가져옵니다.
EvtGetObjectArrayProperty 배열의 지정된 개체에서 공급자 메타데이터 속성을 가져옵니다.
EvtGetObjectArraySize 개체 배열의 요소 수를 가져옵니다.
EvtGetPublisherMetadataProperty 지정된 공급자 메타데이터 속성을 가져옵니다.
EvtGetQueryInfo 쿼리가 액세스하려고 시도한 채널 또는 로그 파일 목록을 식별하는 실행한 쿼리에 대한 정보를 가져옵니다. 또한 함수는 각 액세스의 성공 또는 실패를 나타내는 반환 코드 목록을 가져옵니다.

EvtNext 쿼리 또는 구독 결과에서 다음 이벤트를 가져옵니다.
EvtNextChannelPath 열거자에서 채널 이름을 가져옵니다.
EvtNextEventMetadata 열거자에서 이벤트 정의를 가져옵니다.
EvtNextPublisherId 열거자에서 공급자의 식별자를 가져옵니다.
EvtOpenChannelConfig 채널의 구성 속성을 읽거나 수정하는 데 사용하는 핸들을 가져옵니다.
EvtOpenChannelEnum 컴퓨터에 등록된 채널 목록을 열거하는 데 사용하는 핸들을 가져옵니다.
EvtOpenEventMetadataEnum 공급자가 정의하는 이벤트 목록을 열거하는 데 사용하는 핸들을 가져옵니다.
EvtOpenLog 채널 또는 로그 파일에 대한 정보를 가져오는 데 사용할 수 있는 채널 또는 로그 파일에 대한 핸들을 가져옵니다.
EvtOpenPublisherEnum 컴퓨터에서 등록된 공급자 목록을 열거하는 데 사용하는 핸들을 가져옵니다.
EvtOpenPublisherMetadata 지정된 공급자의 메타데이터를 읽는 데 사용하는 핸들을 가져옵니다.
EvtOpenSession 다른 Windows 이벤트 로그 함수를 호출할 때 사용할 수 있는 원격 컴퓨터에 대한 연결을 설정합니다.
EvtQuery 지정된 쿼리 조건과 일치하는 채널 또는 로그 파일에서 이벤트를 검색하는 쿼리를 실행합니다.

EvtRender 지정한 렌더링 컨텍스트에 따라 XML 조각을 렌더링합니다.
EvtSaveChannelConfig 채널 구성에 대한 변경 내용을 저장합니다.
EvtSeek 쿼리 결과 집합의 특정 이벤트를 찾습니다.
EvtSetChannelConfigProperty 채널의 지정된 구성 속성을 설정합니다.
EvtSubscribe 지정된 쿼리 조건과 일치하는 채널 또는 로그 파일에서 현재 및 이후 이벤트를 수신할 구독을 만듭니다.
EvtUpdateBookmark 지정한 이벤트를 식별하는 정보를 사용하여 책갈피를 업데이트.

구조체

EVT_RPC_LOGIN 원격 컴퓨터에 연결하는 데 사용되는 정보를 포함합니다.
EVT_VARIANT 이벤트 데이터 또는 속성 값을 포함합니다.

Windows 필터링 플랫폼

아티클2024. 06. 12.

Windows 필터링 플랫폼 기술 개요입니다.

Windows 필터링 플랫폼을 개발하려면 다음 헤더가 필요합니다.

- fwpmtypes.h
- fwpmu.h
- fwptypes.h
- fwpsu.h
- iketypes.h
- ipsectypes.h
- netiodef.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 필터링 플랫폼

열거형

테이블 확장

DL_ADDRESS_TYPE DL_ADDRESS_TYPE 열거형(netiodef.h)은 데이터 링크 계층 주소의 형식을 지정합니다.
DL_ADDRESS_TYPE DL_ADDRESS_TYPE 열거형(fwpmtypes.h)은 데이터 링크 계층 주소의 형식을 지정합니다.
FWP_AF FWP_AF 열거형 형식입니다.
FWP_CLASSIFY_OPTION_TYPE FWP_CLASSIFY_OPTION_TYPE 열거형 형식은 런타임 분류 중에 설명선 및 shim에서 사용됩니다. FWP_CLASSIFY_OPTION_TYPE 유니캐스트, 멀티캐스트 및 느슨한 소스 매핑 상태에 대한 시간 제한 옵션을 지정하고 아웃바운드 멀티캐스트 및 브로드캐스트 트래픽에 대한 상태 생성을 차단하거나 사용할 수 있도록 합니다.

FWP_DATA_TYPE FWP_VALUE0 또는 FWP_CONDITION_VALUE0structure 저장할 수 있는 데이터 형식입니다.
FWP_DIRECTION 네트워크 트래픽의 방향입니다.
FWP_ETHER_ENCAP_METHOD 이더넷 II 및 SNAP 트래픽을 캡슐화하는 방법을 지정합니다. 예약.
FWP_FILTER_ENUM_TYPE 필터 열거형 조건을 해석하는 방법을 지정합니다.
FWP_IP_VERSION IP 버전을 지정합니다. (FWP_IP_VERSION)
FWP_MATCH_TYPE 필터 조건에서 허용되는 다른 일치 형식입니다.
FWP_NETWORK_CONNECTION_POLICY_SETTING_TYPE 미정
FWP_VSWITCH_NETWORK_TYPE vSwitch의 네트워크 유형을 지정합니다.
FWPM_APPC_NETWORK_CAPABILITY_TYPE 해당 개체 또는 트래픽과 연결된 앱 컨테이너 네트워크 기능의 유형을 지정합니다.
FWPM_CHANGE_TYPE 구독자에게 변경 알림을 디스패치할 때 사용됩니다.
FWPM_CONNECTION_EVENT_TYPE 연결 개체 변경 이벤트의 유형을 지정합니다.
FWPM_ENGINE_OPTION 필터 엔진에 대한 구성 가능한 옵션입니다.
FWPM_FIELD_TYPE

필드의 데이터를 해석하는 방법에 대한 추가 정보를 제공합니다.
FWPM_NET_EVENT_TYPE 순 이벤트의 유형입니다.
FWPM_PROVIDER_CONTEXT_TYPE BFE(기본 필터링 엔진)에 저장할 수 있는 공급자 컨텍스트 유형입니다.
FWPM_SERVICE_STATE 필터 엔진의 현재 상태를 지정합니다.
FWPM_SYSTEM_PORT_TYPE FWPM_SYSTEM_PORT_TYPE 열거형 형식입니다.
FWPM_VSWITCH_EVENT_TYPE vSwitch 이벤트의 형식을 지정합니다.
FWPS_BUILTIN_LAYERS 기본 제공 런타임 필터링 계층 식별자를 지정하는 상수를 정의합니다. 각각은 크기가 64비트인 LUID(로컬 고유 식별자)로 표시됩니다.
FWPS_FIELDS_ALE_AUTH_CONNECT_V4 FWPS_LAYER_ALE_AUTH_CONNECT_V4 및 FWPS_LAYER_ALE_AUTH_CONNECT_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_AUTH_CONNECT_V6 FWPS_LAYER_ALE_AUTH_CONNECT_V6 및 FWPS_LAYER_ALE_AUTH_CONNECT_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_AUTH_LISTEN_V4 FWPS_LAYER_ALE_AUTH_LISTEN_V4 및 FWPS_LAYER_ALE_AUTH_LISTEN_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_AUTH_LISTEN_V6 FWPS_LAYER_ALE_AUTH_LISTEN_V6 및 FWPS_LAYER_ALE_AUTH_LISTEN_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_AUTH_RECV_ACCEPT_V4 FWPS_LAYER_ALE_AUTH_RECV_ACCEPT_V4 및

FWPS_LAYER_ALE_AUTH_RECV_ACCEPT_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_AUTH_RECV_ACCEPT_V6 FWPS_LAYER_ALE_AUTH_RECV_ACCEPT_V6 및 FWPS_LAYER_ALE_AUTH_RECV_ACCEPT_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_BIND_REDIRECT_V4 FWPS_LAYER_ALE_BIND_REDIRECT_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_BIND_REDIRECT_V6 FWPS_LAYER_ALE_BIND_REDIRECT_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_CONNECT_REDIRECT_V4 FWPS_LAYER_ALE_CONNECT_REDIRECT_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_CONNECT_REDIRECT_V6 FWPS_LAYER_ALE_CONNECT_REDIRECT_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_ENDPOINT_CLOSURE_V4 FWPS_LAYER_ALE_ENDPOINT_CLOSURE_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_ENDPOINT_CLOSURE_V6 FWPS_LAYER_ALE_ENDPOINT_CLOSURE_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_FLOW_ESTABLISHED_V4 FWPS_LAYER_ALE_FLOW_ESTABLISHED_V4 및 FWPS_LAYER_ALE_FLOW_ESTABLISHED_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_FLOW_ESTABLISHED_V6 FWPS_LAYER_ALE_FLOW_ESTABLISHED_V6 및 FWPS_LAYER_ALE_FLOW_ESTABLISHED_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_RESOURCE_ASSIGNMENT_V4 FWPS_LAYER_ALE_RESOURCE_ASSIGNMENT_V4 및

FWPS_LAYER_ALE_RESOURCE_ASSIGNMENT_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_RESOURCE_ASSIGNMENT_V6 FWPS_LAYER_ALE_RESOURCE_ASSIGNMENT_V6 및 FWPS_LAYER_ALE_RESOURCE_ASSIGNMENT_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_RESOURCE_RELEASE_V4 FWPS_LAYER_ALE_RESOURCE_RELEASE_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_RESOURCE_RELEASE_V6 FWPS_LAYER_ALE_RESOURCE_RELEASE_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_DATAGRAM_DATA_V4 FWPS_LAYER_ALE_RESOURCE_ASSIGNMENT_V4 및 FWPS_LAYER_ALE_RESOURCE_ASSIGNMENT_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_DATAGRAM_DATA_V6 FWPS_LAYER_ALE_RESOURCE_ASSIGNMENT_V6 및 FWPS_LAYER_ALE_RESOURCE_ASSIGNMENT_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_EGRESS_VSWITCH_ETHERNET FWPS_LAYER_EGRESS_VSWITCH_ETHERNET 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_EGRESS_VSWITCH_TRANSPORT_V4 FWPS_LAYER_EGRESS_VSWITCH_TRANSPORT_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_EGRESS_VSWITCH_TRANSPORT_V6 FWPS_LAYER_EGRESS_VSWITCH_TRANSPORT_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_IKEEXT_V4 FWPS_LAYER_IKEEXT_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.

FWPS_FIELDS_IKEEXT_V6 FWPS_LAYER_IKEEXT_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_INBOUND_ICMP_ERROR_V4 FWPS_LAYER_INBOUND_ICMP_ERROR_V4 및 FWPS_LAYER_INBOUND_ICMP_ERROR_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_INBOUND_ICMP_ERROR_V6 FWPS_LAYER_INBOUND_ICMP_ERROR_V6 및 FWPS_LAYER_INBOUND_ICMP_ERROR_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_INBOUND_IPPACKET_V4 FWPS_LAYER_INBOUND_IPPACKET_V4 및 FWPS_LAYER_INBOUND_IPPACKET_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_INBOUND_IPPACKET_V6 FWPS_LAYER_INBOUND_IPPACKET_V6 및 FWPS_LAYER_INBOUND_IPPACKET_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_INBOUND_MAC_FRAME_ETHERNET FWPS_LAYER_INBOUND_MAC_FRAME_ETHERNET 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_INBOUND_MAC_FRAME_NATIVE FWPS_LAYER_INBOUND_MAC_FRAME_ETHERNET 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_INBOUND_MAC_FRAME_NATIVE_FAST 미정
FWPS_FIELDS_INBOUND_RESERVED2 내부 사용을 위해 값이 예약된 열거형입니다.
FWPS_FIELDS_INBOUND_TRANSPORT_FAST 미정
FWPS_FIELDS_INBOUND_TRANSPORT_V4

FWPS_LAYER_INBOUND_TRANSPORT_V4 및 FWPS_LAYER_INBOUND_TRANSPORT_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_INBOUND_TRANSPORT_V6 FWPS_LAYER_INBOUND_TRANSPORT_V6 및 FWPS_LAYER_INBOUND_TRANSPORT_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_INGRESS_VSWITCH_ETHERNET FWPS_LAYER_INGRESS_VSWITCH_ETHERNET 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_INGRESS_VSWITCH_TRANSPORT_V4 FWPS_LAYER_INGRESS_VSWITCH_TRANSPORT_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_INGRESS_VSWITCH_TRANSPORT_V6 FWPS_LAYER_INGRESS_VSWITCH_TRANSPORT_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_IPFORWARD_V4 FWPS_LAYER_IPFORWARD_V4 및 FWPS_LAYER_IPFORWARD_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_IPFORWARD_V6 FWPS_LAYER_IPFORWARD_V6 및 FWPS_LAYER_IPFORWARD_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_IPSEC_KM_DEMUX_V4 FWPS_LAYER_IPSEC_KM_DEMUX_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_IPSEC_KM_DEMUX_V6 FWPS_LAYER_IPSEC_KM_DEMUX_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_IPSEC_V4 FWPS_LAYER_IPSEC_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_IPSEC_V6 FWPS_LAYER_IPSEC_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.

FWPS_FIELDS_KM_AUTHORIZATION FWPS_LAYER_KM_AUTHORIZATION 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_NAME_RESOLUTION_CACHE_V4 FWPS_LAYER_NAME_RESOLUTION_CACHE_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_NAME_RESOLUTION_CACHE_V6 FWPS_LAYER_NAME_RESOLUTION_CACHE_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_OUTBOUND_ICMP_ERROR_V4 FWPS_LAYER_OUTBOUND_ICMP_ERROR_V4 및 FWPS_LAYER_OUTBOUND_ICMP_ERROR_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_OUTBOUND_ICMP_ERROR_V6 FWPS_LAYER_OUTBOUND_ICMP_ERROR_V6 및 FWPS_LAYER_OUTBOUND_ICMP_ERROR_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_OUTBOUND_IPPACKET_V4 FWPS_LAYER_OUTBOUND_IPPACKET_V4 및 FWPS_LAYER_OUTBOUND_IPPACKET_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_OUTBOUND_IPPACKET_V6 FWPS_LAYER_OUTBOUND_IPPACKET_V6 및 FWPS_LAYER_OUTBOUND_IPPACKET_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_OUTBOUND_MAC_FRAME_ETHERNET FWPS_LAYER_OUTBOUND_MAC_FRAME_ETHERNET 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_OUTBOUND_MAC_FRAME_NATIVE FWPS_LAYER_OUTBOUND_MAC_FRAME_NATIVE 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_OUTBOUND_MAC_FRAME_NATIVE_FAST 미정

FWPS_FIELDS_OUTBOUND_NETWORK_POLICY_V4 FWPS_LAYER_OUTBOUND_NETWORK_CONNECTION_POLICY_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_OUTBOUND_NETWORK_POLICY_V6 FWPS_LAYER_OUTBOUND_NETWORK_CONNECTION_POLICY_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_OUTBOUND_TRANSPORT_FAST 미정
FWPS_FIELDS_OUTBOUND_TRANSPORT_V4 FWPS_LAYER_OUTBOUND_TRANSPORT_V4 및 FWPS_LAYER_OUTBOUND_TRANSPORT_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_OUTBOUND_TRANSPORT_V6 FWPS_LAYER_OUTBOUND_TRANSPORT_V6 및 FWPS_LAYER_OUTBOUND_TRANSPORT_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_RPC_EP_ADD FWPS_LAYER_RPC_EP_ADD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_RPC_EPMAP FWPS_LAYER_RPC_EPMAP 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_RPC_PROXY_CONN FWPS_LAYER_RPC_PROXY_CONN 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_RPC_PROXY_IF_IF FWPS_LAYER_RPC_PROXY_IF 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_RPC_UM FWPS_LAYER_RPC_UM 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_STREAM_PACKET_V4 FWPS_LAYER_STREAM_PACKET_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.

FWPS_FIELDS_STREAM_PACKET_V6 FWPS_LAYER_STREAM_PACKET_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_STREAM_V4 FWPS_LAYER_STREAM_V4 및 FWPS_LAYER_STREAM_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_STREAM_V6 FWPS_LAYER_STREAM_V6 및 FWPS_LAYER_STREAM_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
IKEEXT_AUTHENTICATION_IMPERSONATION_TYPE 인증에 인증된 인터넷 프로토콜(AuthIP)을 사용할 때 수행할 가장 유형을 지정합니다.
IKEEXT_AUTHENTICATION_METHOD_TYPE IKE(인터넷 키 교환), 인증된 인터넷 프로토콜(AuthIP) 또는 IKEv2에서 사용하는 인증 방법 유형을 지정합니다.
IKEEXT_CERT_CONFIG_TYPE 인증서 구성의 유형을 나타냅니다.
IKEEXT_CERT_CRITERIA_NAME_TYPE 인증서 선택 "주체" 조건에 사용할 수 있는 NAME 필드의 형식을 지정합니다.
IKEEXT_CIPHER_TYPE IKE(인터넷 키 교환) 및 인증된 인터넷 프로토콜(AuthIP) 메시지를 암호화하는 데 사용되는 암호화 알고리즘 유형을 지정합니다.
IKEEXT_DH_GROUP IKE(인터넷 키 교환) 및 인증된 인터넷 프로토콜(AuthIP) 키 생성에 사용되는 Diffie Hellman 그룹의 유형을 지정합니다.
IKEEXT_EM_SA_STATE 인증된 인터넷 프로토콜(AuthIP) 프로토콜의 일부인 EM(확장 모드) 협상 교환의 상태입니다.
IKEEXT_INTEGRITY_TYPE IKE(인터넷 키 교환) 및 인증된 인터넷 프로토콜(AuthIP) 메시지의 무결성 보호에 사용되는 해시 알고리즘 유형을 지정합니다.

IKEEXT_KEY_MODULE_TYPE 키 지정 모듈의 유형을 지정합니다.
IKEEXT_MM_SA_STATE 인증된 인터넷 프로토콜(AuthIP) 및 IKE(인터넷 키 교환) 프로토콜의 일부인 MM(기본 모드) 협상 교환의 상태입니다.
IKEEXT_QM_SA_STATE 인증된 인터넷 프로토콜(AuthIP) 및 IKE(인터넷 키 교환) 프로토콜의 일부인 QM(빠른 모드) 협상 교환의 상태입니다.
IKEEXT_SA_ROLE IKE(인터넷 키 교환) 및 인증된 인터넷 프로토콜(AuthIP) 기본 모드 또는 빠른 모드 협상에 대한 SA(보안 연결) 역할입니다.
INET_DISCARD_REASON 전송 계층 중 하나에서 데이터를 삭제할 수 있는 이유를 정의합니다.
IP_DISCARD_REASON 네트워크 계층 중 하나에서 데이터를 삭제할 수 있는 이유를 정의합니다.
IPSEC_AUTH_TYPE 데이터 원본 인증 및 무결성 보호를 위해 IPsec SA에서 사용되는 해시 알고리즘의 유형을 나타냅니다.
IPSEC_CIPHER_TYPE 데이터 개인 정보 보호를 위해 IPsec SA에서 사용되는 암호화 알고리즘의 유형을 나타냅니다.
IPSEC_FAILURE_POINT IPsec이 실패한 시점입니다.
IPSEC_PFS_GROUP 빠른 모드 PFS(완벽한 전달 비밀)에 사용해야 하는 Diffie Hellman 알고리즘을 지정합니다.
IPSEC_SA_CONTEXT_EVENT_TYPE0 SA(IPsec 보안 연결) 컨텍스트 변경 이벤트의 유형을 지정합니다.

IPSEC_TOKEN_MODE 토큰을 가져올 수 있는 다양한 IPsec 모드입니다.
IPSEC_TOKEN_PRINCIPAL 액세스 토큰 보안 주체입니다.
IPSEC_TOKEN_TYPE IPsec 토큰 형식입니다.
IPSEC_TRAFFIC_TYPE 설명되는 IPsec 트래픽의 형식을 지정합니다.
IPSEC_TRANSFORM_TYPE IPsec SA(보안 연결) 변환의 형식을 나타냅니다.

함수

테이블 확장

FWPM_CALLOUT_CHANGE_CALLBACK0 설명선 변경 알림 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다.
FWPM_CONNECTION_CALLBACK0 연결 개체 구독 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다.
FWPM_DYNAMIC_KEYWORD_CALLBACK0 구현하는 콜백 함수는 동적 키워드 주소(FW_DYNAMIC_KEYWORD_ADDRESS0) 개체의 변경 내용에 대한 알림과 함께 호출됩니다.
FWPM_FILTER_CHANGE_CALLBACK0 필터 변경 알림 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다.
FWPM_NET_EVENT_CALLBACK0

순 이벤트 구독 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다. (FWPM_NET_EVENT_CALLBACK0)
FWPM_NET_EVENT_CALLBACK1 순 이벤트 구독 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다. (FWPM_NET_EVENT_CALLBACK1)
FWPM_NET_EVENT_CALLBACK2 순 이벤트 구독 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다. (FWPM_NET_EVENT_CALLBACK2)
FWPM_NET_EVENT_CALLBACK3 net 이벤트 구독 프로세스에 사용자 지정 동작을 추가하기 위한 콜백 함수입니다.
FWPM_NET_EVENT_CALLBACK4 net 이벤트 구독 프로세스에 사용자 지정 동작을 추가하기 위한 콜백 함수입니다.
FWPM_PROVIDER_CHANGE_CALLBACK0 공급자 변경 알림 프로세스에 사용자 지정 동작을 추가하려면
FWPM_PROVIDER_CONTEXT_CHANGE_CALLBACK0 공급자 컨텍스트 변경 알림 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다.
FWPM_SUBLAYER_CHANGE_CALLBACK0 하위 계층 변경 알림 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다.
FWPM_SYSTEM_PORTS_CALLBACK0 시스템 포트 구독 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다.
FWPM_VSWITCH_EVENT_CALLBACK0 vSwitch 이벤트 구독 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다.
FwpmCalloutAdd0 시스템에 새 설명선 개체를 추가합니다.
FwpmCalloutCreateEnumHandle0 설명선 개체 집합을 열거하는 데 사용되는 핸들을 만듭니다.

FwpmCalloutDeleteById0 시스템에서 설명선 개체를 제거합니다. (FwpmCalloutDeleteById0)
FwpmCalloutDeleteByKey0 시스템에서 설명선 개체를 제거합니다. (FwpmCalloutDeleteByKey0)
FwpmCalloutDestroyEnumHandle0 FwpmCalloutCreateEnumHandle0에서 반환된 핸들을 해제합니다.
FwpmCalloutEnum0 설명선 열거자에서 결과의 다음 페이지를 반환합니다.
FwpmCalloutGetById0 설명선 개체를 검색합니다. (FwpmCalloutGetById0)
FwpmCalloutGetByKey0 설명선 개체를 검색합니다. (FwpmCalloutGetByKey0)
FwpmCalloutGetSecurityInfoByKey0 설명자 개체에 대한 보안 설명자의 복사본을 검색합니다.
FwpmCalloutSetSecurityInfoByKey0 설명선 개체의 보안 설명자에 지정된 보안 정보를 설정합니다.
FwpmCalloutSubscribeChanges0 특정 설명선의 변경 내용에 대한 알림 배달을 요청하는 데 사용됩니다.
FwpmCalloutSubscriptionsGet0 모든 현재 설명선 변경 알림 구독의 배열을 검색합니다.
FwpmCalloutUnsubscribeChanges0 설명선 변경 구독을 취소하고 변경 알림 수신을 중지하는 데 사용됩니다.
FwpmConnectionCreateEnumHandle0 연결 개체 집합을 열거하는 데 사용되는 핸들을 만듭니다.
FwpmConnectionDestroyEnumHandle0

FwpmConnectionCreateEnumHandle0에서 반환된 핸들을 해제합니다.
FwpmConnectionEnum0 연결 개체 열거자에서 결과의 다음 페이지를 반환합니다.
FwpmConnectionGetById0 연결 개체를 검색합니다.
FwpmConnectionGetSecurityInfo0 연결 개체 변경 이벤트에 대한 보안 설명자의 복사본을 검색합니다.
FwpmConnectionPolicyAdd0 아웃바운드 연결에 대한 표현 라우팅 정책을 구성할 수 있습니다.
FwpmConnectionPolicyDeleteByKey0 FwpmConnectionPolicyAdd0을 호출할 때 FWPM_PROVIDER_CONTEXT::p roviderContextKey에 지정된 GUID를 사용하여 만든 연결 정책을 제거합니다.
FwpmConnectionSetSecurityInfo0 연결 개체 변경 이벤트에 대한 보안 설명자에 지정된 보안 정보를 설정합니다.
FwpmConnectionSubscribe0 연결 개체의 변경 내용에 대한 알림 배달을 요청하는 데 사용됩니다.
FwpmConnectionSubscriptionsGet0 모든 현재 연결 개체 변경 알림 구독의 배열을 검색합니다.
FwpmConnectionUnsubscribe0 연결 개체 변경 이벤트 구독을 취소하고 알림 수신을 중지하는 데 사용됩니다.
FwpmDynamicKeywordSubscribe0 특정 동적 키워드 주소(FW_DYNAMIC_KEYWORD_ADDRESS0) 개체의 변경 내용에 대한 알림 배달을 요청합니다.
FwpmDynamicKeywordUnsubscribe0 특정 동적 키워드 주소(FW_DYNAMIC_KEYWORD_ADDRESS0) 개체의 변경 내용에 대한 알림 배달을 취소합니다.

FwpmEngineClose0 필터 엔진에 대한 세션을 닫습니다.
FwpmEngineGetOption0 필터 엔진 옵션을 검색합니다.
FwpmEngineGetSecurityInfo0 필터 엔진에 대한 보안 설명자의 복사본을 검색합니다.
FwpmEngineOpen0 필터 엔진에 대한 세션을 엽니다.
FwpmEngineSetOption0 필터 엔진 설정을 변경합니다.
FwpmEngineSetSecurityInfo0 필터 엔진의 보안 설명자에 지정된 보안 정보를 설정합니다.
FwpmFilterAdd0 시스템에 새 필터 개체를 추가합니다.
FwpmFilterCreateEnumHandle0 필터 개체 집합을 열거하는 데 사용되는 핸들을 만듭니다.
FwpmFilterDeleteById0 시스템에서 필터 개체를 제거합니다. (FwpmFilterDeleteById0)
FwpmFilterDeleteByKey0 시스템에서 필터 개체를 제거합니다. (FwpmFilterDeleteByKey0)
FwpmFilterDestroyEnumHandle0 FwpmFilterCreateEnumHandle0에서 반환된 핸들을 해제합니다. (FwpmFilterDestroyEnumHandle0)
FwpmFilterEnum0 필터 열거자에서 결과의 다음 페이지를 반환합니다.
FwpmFilterGetById0

필터 개체를 검색합니다. (FwpmFilterGetById0)
FwpmFilterGetByKey0 필터 개체를 검색합니다. (FwpmFilterGetByKey0)
FwpmFilterGetSecurityInfoByKey0 필터 개체에 대한 보안 설명자의 복사본을 검색합니다.
FwpmFilterSetSecurityInfoByKey0 필터 개체의 보안 설명자에서 지정된 보안 정보를 설정합니다.
FwpmFilterSubscribeChanges0 특정 필터의 변경 내용에 대한 알림 배달을 요청하는 데 사용됩니다.
FwpmFilterSubscriptionsGet0 모든 현재 필터 변경 알림 구독의 배열을 검색합니다.
FwpmFilterUnsubscribeChanges0 필터 변경 구독을 취소하고 변경 알림 수신을 중지하는 데 사용됩니다.
FwpmFreeMemory0 WFP(Windows Filtering Platform) 함수에서 할당한 메모리 리소스를 해제하는 데 사용됩니다.
FwpmGetAppIdFromFileName0 파일 이름에서 애플리케이션 식별자를 검색합니다.
FwpmIPsecTunnelAdd0 시스템에 새 IPsec(인터넷 프로토콜 보안) 터널 모드 정책을 추가합니다. (FwpmIpsecTunnelAdd0)
FwpmIPsecTunnelAdd1 시스템에 새 IPsec(인터넷 프로토콜 보안) 터널 모드 정책을 추가합니다. (FwpmIpsecTunnelAdd1)
FwpmIPsecTunnelAdd2 시스템에 새 IPsec(인터넷 프로토콜 보안) 터널 모드 정책을 추가합니다. (FwpmIPsecTunnelAdd2)
FwpmIPsecTunnelAdd3 시스템에 새 IPsec(인터넷 프로토콜 보안) 터널 모드 정책을 추가합니다.

FwpmIPsecTunnelDeleteByKey0 시스템에서 IPsec(인터넷 프로토콜 보안) 터널 모드 정책을 제거합니다.
FwpmLayerCreateEnumHandle0 레이어 개체 집합을 열거하는 데 사용되는 핸들을 만듭니다.
FwpmLayerDestroyEnumHandle0 FwpmFilterCreateEnumHandle0에서 반환된 핸들을 해제합니다. (FwpmLayerDestroyEnumHandle0)
FwpmLayerEnum0 레이어 열거자에서 결과의 다음 페이지를 반환합니다.
FwpmLayerGetById0 레이어 개체를 검색합니다. (FwpmLayerGetById0)
FwpmLayerGetByKey0 레이어 개체를 검색합니다. (FwpmLayerGetByKey0)
FwpmLayerGetSecurityInfoByKey0 계층 개체에 대한 보안 설명자의 복사본을 검색합니다.
FwpmLayerSetSecurityInfoByKey0 계층 개체의 보안 설명자에 지정된 보안 정보를 설정합니다.
FwpmNetEventCreateEnumHandle0 네트워크 이벤트 집합을 열거하는 데 사용되는 핸들을 만듭니다.
FwpmNetEventDestroyEnumHandle0 FwpmNetEventCreateEnumHandle0에서 반환된 핸들을 해제합니다.
FwpmNetEventEnum0 네트워크 이벤트 열거자에서 결과의 다음 페이지를 반환합니다. (FwpmNetEventEnum0)
FwpmNetEventEnum1 네트워크 이벤트 열거자에서 결과의 다음 페이지를 반환합니다. (FwpmNetEventEnum1)

FwpmNetEventEnum2 네트워크 이벤트 열거자에서 결과의 다음 페이지를 반환합니다. (FwpmNetEventEnum2)
FwpmNetEventEnum3 네트워크 이벤트 열거자에서 결과의 다음 페이지를 반환합니다. (FwpmNetEventEnum3)
FwpmNetEventEnum4 네트워크 이벤트 열거자에서 결과의 다음 페이지를 검색합니다.
FwpmNetEventEnum5 네트워크 이벤트 열거자에서 결과의 다음 페이지를 검색합니다.
FwpmNetEventsGetSecurityInfo0 FwpmNetEventsGetSecurityInfo0 함수는 네트워크 이벤트 개체에 대한 보안 설명자의 복사본을 검색합니다.
FwpmNetEventsSetSecurityInfo0 네트워크 이벤트 개체의 보안 설명자에 지정된 보안 정보를 설정합니다.
FwpmNetEventSubscribe0 특정 Net 이벤트에 대한 알림 배달을 요청하는 데 사용됩니다. (FwpmNetEventSubscribe0)
FwpmNetEventSubscribe1 특정 Net 이벤트에 대한 알림 배달을 요청하는 데 사용됩니다. (FwpmNetEventSubscribe1)
FwpmNetEventSubscribe2 특정 Net 이벤트에 대한 알림 배달을 요청하는 데 사용됩니다. (FwpmNetEventSubscribe2)
FwpmNetEventSubscribe3 특정 순 이벤트에 대한 알림 배달을 요청하는 데 사용됩니다.
FwpmNetEventSubscribe4 특정 순 이벤트에 대한 알림 배달을 요청하는 데 사용됩니다.
FwpmNetEventSubscriptionsGet0 모든 현재 순 이벤트 알림 구독의 배열을 검색합니다.

FwpmNetEventUnsubscribe0 순 이벤트 구독을 취소하고 알림 수신을 중지하는 데 사용됩니다.
FwpmProviderAdd0 시스템에 새 공급자를 추가합니다.
FwpmProviderContextAdd0 시스템에 새 공급자 컨텍스트를 추가합니다. (FwpmProviderContextAdd0)
FwpmProviderContextAdd1 시스템에 새 공급자 컨텍스트를 추가합니다. (FwpmProviderContextAdd1)
FwpmProviderContextAdd2 시스템에 새 공급자 컨텍스트를 추가합니다. (FwpmProviderContextAdd2)
FwpmProviderContextAdd3 시스템에 새 공급자 컨텍스트를 추가합니다.
FwpmProviderContextCreateEnumHandle0 공급자 컨텍스트 집합을 열거하는 데 사용되는 핸들을 만듭니다.
FwpmProviderContextDeleteById0 시스템에서 공급자 컨텍스트를 제거합니다.
FwpmProviderContextDeleteByKey0 시스템에서 공급자 컨텍스트를 제거합니다.
FwpmProviderContextDestroyEnumHandle0 FwpmProviderContextCreateEnumHandle0에서 반환된 핸들을 해제합니다.
FwpmProviderContextEnum0 공급자 컨텍스트 열거자에서 결과의 다음 페이지를 반환합니다. (FwpmProviderContextEnum0)
FwpmProviderContextEnum1 공급자 컨텍스트 열거자에서 결과의 다음 페이지를 반환합니다. (FwpmProviderContextEnum1)
FwpmProviderContextEnum2

공급자 컨텍스트 열거자에서 결과의 다음 페이지를 반환합니다. (FwpmProviderContextEnum2)
FwpmProviderContextEnum3 공급자 컨텍스트 열거자에서 결과의 다음 페이지를 반환합니다.
FwpmProviderContextGetById0 공급자 컨텍스트를 검색합니다. (FwpmProviderContextGetById0)
FwpmProviderContextGetById1 공급자 컨텍스트를 검색합니다. (FwpmProviderContextGetById1)
FwpmProviderContextGetById2 공급자 컨텍스트를 검색합니다. (FwpmProviderContextGetById2)
FwpmProviderContextGetById3 공급자 컨텍스트를 검색합니다.
FwpmProviderContextGetByKey0 공급자 컨텍스트를 검색합니다. (FwpmProviderContextGetByKey0)
FwpmProviderContextGetByKey1 공급자 컨텍스트를 검색합니다. (FwpmProviderContextGetByKey1)
FwpmProviderContextGetByKey2 공급자 컨텍스트를 검색합니다. (FwpmProviderContextGetByKey2)
FwpmProviderContextGetByKey3 공급자 컨텍스트를 검색합니다.
FwpmProviderContextGetSecurityInfoByKey0 공급자 컨텍스트 개체에 대한 보안 설명자의 복사본을 검색합니다.
FwpmProviderContextSetSecurityInfoByKey0 공급자 컨텍스트 개체의 보안 설명자에서 지정된 보안 정보를 설정합니다.
FwpmProviderContextSubscribeChanges0 특정 공급자 컨텍스트의 변경 내용에 대한 알림 배달을 요청하는 데 사용됩니다.

FwpmProviderContextSubscriptionsGet0 모든 현재 공급자 컨텍스트 변경 알림 구독의 배열을 검색합니다.
FwpmProviderContextUnsubscribeChanges0 공급자 컨텍스트 변경 구독을 취소하고 변경 알림 수신을 중지하는 데 사용됩니다.
FwpmProviderCreateEnumHandle0 공급자 집합을 열거하는 데 사용되는 핸들을 만듭니다.
FwpmProviderDeleteByKey0 시스템에서 공급자를 제거합니다.
FwpmProviderDestroyEnumHandle0 FwpmProviderCreateEnumHandle0에서 반환된 핸들을 해제합니다.
FwpmProviderEnum0 공급자 열거자에서 결과의 다음 페이지를 반환합니다.
FwpmProviderGetByKey0 공급자를 검색합니다.
FwpmProviderGetSecurityInfoByKey0 공급자 개체에 대한 보안 설명자의 복사본을 검색합니다.
FwpmProviderSetSecurityInfoByKey0 공급자 개체의 보안 설명자에 지정된 보안 정보를 설정합니다.
FwpmProviderSubscribeChanges0 특정 공급자의 변경 내용에 대한 알림 배달을 요청하는 데 사용됩니다.
FwpmProviderSubscriptionsGet0 모든 현재 공급자 변경 알림 구독의 배열을 검색합니다.
FwpmProviderUnsubscribeChanges0 공급자 변경 구독을 취소하고 변경 알림 수신을 중지하는 데 사용됩니다.
FwpmSessionCreateEnumHandle0

세션 개체 집합을 열거하는 데 사용되는 핸들을 만듭니다.
FwpmSessionDestroyEnumHandle0 FwpmSessionCreateEnumHandle0에서 반환된 핸들을 해제합니다.
FwpmSessionEnum0 세션 열거자에서 결과의 다음 페이지를 반환합니다.
FwpmSubLayerAdd0 시스템에 새 하위 계층을 추가합니다.
FwpmSubLayerCreateEnumHandle0 하위 계층 집합을 열거하는 데 사용되는 핸들을 만듭니다.
FwpmSubLayerDeleteByKey0 키로 시스템에서 하위 계층을 삭제합니다.
FwpmSubLayerDestroyEnumHandle0 FwpmSubLayerCreateEnumHandle0에서 반환된 핸들을 해제합니다.
FwpmSubLayerEnum0 하위 계층 열거자에서 결과의 다음 페이지를 반환합니다.
FwpmSubLayerGetByKey0 해당 키로 하위 계층을 검색합니다.
FwpmSubLayerGetSecurityInfoByKey0 하위 계층에 대한 보안 설명자의 복사본을 검색합니다.
FwpmSubLayerSetSecurityInfoByKey0 하위 계층의 보안 설명자에서 지정된 보안 정보를 설정합니다.
FwpmSubLayerSubscribeChanges0 특정 하위 계층의 변경 내용에 대한 알림 배달을 요청하는 데 사용됩니다.
FwpmSubLayerSubscriptionsGet0 현재 모든 하위 계층 변경 알림 구독의 배열을 검색합니다.

FwpmSubLayerUnsubscribeChanges0 하위 계층 변경 구독을 취소하고 변경 알림 수신을 중지하는 데 사용됩니다.
FwpmSystemPortsGet0 모든 시스템 포트 형식의 배열을 검색합니다.
FwpmSystemPortsSubscribe0 특정 시스템 포트와 관련된 알림 배달을 요청하는 데 사용됩니다.
FwpmSystemPortsUnsubscribe0 시스템 포트 구독을 취소하고 알림 수신을 중지하는 데 사용됩니다.
FwpmTransactionAbort0 현재 세션 내의 현재 트랜잭션이 중단되고 롤백되도록 합니다.
FwpmTransactionBegin0 현재 세션 내에서 명시적 트랜잭션을 시작합니다.
FwpmTransactionCommit0 현재 세션 내에서 현재 트랜잭션을 커밋합니다.
FwpmvSwitchEventsGetSecurityInfo0 vSwitch 이벤트에 대한 보안 설명자의 복사본을 검색합니다.
FwpmvSwitchEventsSetSecurityInfo0 vSwitch 이벤트에 대한 보안 설명자에서 지정된 보안 정보를 설정합니다.
FwpmvSwitchEventSubscribe0 특정 vSwitch 이벤트에 대한 알림 배달을 요청하는 데 사용됩니다.
FwpmvSwitchEventUnsubscribe0 vSwitch 이벤트 구독을 취소하고 알림 수신을 중지하는 데 사용됩니다.
FWPS_CALLOUT_BOOTTIME_CALLOUT_DELETE_NOTIFY_FN0 미정
FWPS_CALLOUT_CLASSIFY_FN0

필터 엔진은 설명선에서 처리할 데이터가 있을 때마다 설명선의 classifyFn0 설명선 함수를 호출합니다.
FWPS_CALLOUT_CLASSIFY_FN1 필터 엔진은 설명선에서 처리할 데이터가 있을 때마다 설명선의 classifyFn1 설명선 함수를 호출합니다.
FWPS_CALLOUT_CLASSIFY_FN2 필터 엔진은 설명선에서 처리할 데이터가 있을 때마다 설명선의 classifyFn2 설명선 함수를 호출합니다.
FWPS_CALLOUT_CLASSIFY_FN3 미정
FWPS_CALLOUT_FLOW_DELETE_NOTIFY_FN0 필터 엔진은 설명선의 flowDeleteFn 설명선 함수를 호출하여 설명선에서 처리 중인 데이터 흐름이 종료되고 있음을 설명선에 알립니다.
FWPS_CALLOUT_NOTIFY_FN0 필터 엔진은 설명선의 notifyFn0 설명선 함수를 호출하여 설명선 드라이버에 설명선과 연결된 이벤트에 대해 알립니다.
FWPS_CALLOUT_NOTIFY_FN1 필터 엔진은 설명선의 **notifyFn1** 설명선 함수를 호출하여 설명선 드라이버에 설명선과 연결된 이벤트에 대해 알립니다.
FWPS_CALLOUT_NOTIFY_FN2 필터 엔진은 설명선의 notifyFn2 설명선 함수를 호출하여 설명선 드라이버에 설명선과 연결된 이벤트에 대해 알립니다.
FWPS_CALLOUT_NOTIFY_FN3 미정
FWPS_IS_L2_METADATA_FIELD_PRESENT 미정
FWPS_IS_METADATA_FIELD_PRESENT 미정

FwpsAleEndpointCreateEnumHandle0 다른 ALE(애플리케이션 계층 적용) 엔드포인트 함수와 함께 엔드포인트 데이터를 열거하는 데 사용할 수 있는 핸들을 만듭니다.
FwpsAleEndpointDestroyEnumHandle0 FwpsAleEndpointCreateEnumHandle0을 호출하여 만든 엔드포인트 열거형 핸들을 삭제합니다.
FwpsAleEndpointEnum0 ALE(애플리케이션 계층 적용) 엔드포인트를 열거합니다.
FwpsAleEndpointGetById0 ALE(애플리케이션 계층 적용) 엔드포인트에 대한 정보를 검색합니다.
FwpsAleEndpointGetSecurityInfo0 ALE(애플리케이션 계층 적용) 엔드포인트 열거 세션에 대한 보안 정보를 검색합니다.
FwpsAleEndpointSetSecurityInfo0 ALE(애플리케이션 계층 적용) 엔드포인트 열거 세션에 대한 보안 정보를 설정합니다.
FwpsOpenToken0 액세스 토큰을 엽니다.
IkeextGetStatistics0 IKE(인터넷 키 교환) 및 인증된 인터넷 프로토콜(AuthIP) 통계를 검색합니다. (IkeextGetStatistics0)
IkeextGetStatistics1 IKE(인터넷 키 교환) 및 인증된 인터넷 프로토콜(AuthIP) 통계를 검색합니다. (IkeextGetStatistics1)
IkeextSaCreateEnumHandle0 IKE(인터넷 키 교환) 및 인증된 인터넷 프로토콜(AuthIP) SA(보안 연결) 개체 집합을 열거하는 데 사용되는 핸들을 만듭니다.
IkeextSaDbGetSecurityInfo0 IkeextSaDbGetSecurityInfo0 함수는 SA(보안 연결) 데이터베이스에 대한 보안 설명자의 복사본을 검색합니다.
IkeextSaDbSetSecurityInfo0

IkeextSaDbSetSecurityInfo0 함수는 IKE/AuthIP 보안 연결 데이터베이스의 보안 설명자에 지정된 보안 정보를 설정합니다.
IkeextSaDeleteById0 IkeextSaDeleteById0 함수는 데이터베이스에서 SA(보안 연결)를 제거합니다.
IkeextSaDestroyEnumHandle0 IkeextSaCreateEnumHandle0에서 반환된 핸들을 해제합니다.
IkeextSaEnum0 IKE/AuthIP SA(보안 연결) 열거자에서 결과의 다음 페이지를 반환합니다. (IkeextSaEnum0)
IkeextSaEnum1 IKE/AuthIP SA(보안 연결) 열거자에서 결과의 다음 페이지를 반환합니다. (IkeextSaEnum1)
IkeextSaEnum2 IKE/AuthIP SA(보안 연결) 열거자에서 결과의 다음 페이지를 반환합니다. (IkeextSaEnum2)
IkeextSaGetById0 데이터베이스에서 IKE/AuthIP SA(보안 연결)를 검색합니다. (IkeextSaGetById0)
IkeextSaGetById1 데이터베이스에서 IKE/AuthIP SA(보안 연결)를 검색합니다. (IkeextSaGetById1)
IkeextSaGetById2 데이터베이스에서 IKE/AuthIP SA(보안 연결)를 검색합니다. (IkeextSaGetById2)
IPSEC_KEY_MANAGER_DICTATE_KEY0 TIA(신뢰할 수 있는 중개자 에이전트)에서 협상 중인 SA에 대한 키를 지시하는 데 사용됩니다.
IPSEC_KEY_MANAGER_KEY_DICTATION_CHECK0 TIA(신뢰할 수 있는 중간 에이전트)가 협상 중인 SA의 키를 지시할지 여부를 나타냅니다.
IPSEC_KEY_MANAGER_NOTIFY_KEY0 협상 중인 SA에 대한 키를 TIA(신뢰할 수 있는 중간 에이전트)에 알리는 데 사용됩니다.
IPSEC_SA_CONTEXT_CALLBACK0 SA(IPsec 보안 연결) 컨텍스트 구독 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다.

IPsecDospGetSecurityInfo0 IPsec DoS Protection 데이터베이스에 대한 보안 설명자의 복사본을 검색합니다.
IPsecDospGetStatistics0 IPsecDospGetStatistics0 함수는 IPsec(인터넷 프로토콜 보안) DoS Protection 통계를 검색합니다.
IPsecDospSetSecurityInfo0 IPsecDospSetSecurityInfo0 함수는 IPsec DoS Protection 데이터베이스의 보안 설명자에 지정된 보안 정보를 설정합니다.
IPsecDospStateCreateEnumHandle0 IPsec DoS Protection 개체 집합을 열거하는 데 사용되는 핸들을 만듭니다.
IPsecDospStateDestroyEnumHandle0 IPsecDospStateCreateEnumHandle0에서 반환된 핸들을 해제합니다.
IPsecDospStateEnum0 IPsec DoS Protection 상태 열거자에서 결과의 다음 페이지를 반환합니다. 각 IPsec DoS Protection 상태 항목은 IPsec DoS Protection 인증 검사를 성공적으로 통과한 흐름에 해당합니다.
IPsecGetStatistics0 IPsec(인터넷 프로토콜 보안) 통계를 검색합니다. (IPsecGetStatistics0)
IPsecGetStatistics1 IPsec(인터넷 프로토콜 보안) 통계를 검색합니다. (IPsecGetStatistics1)
IPsecKeyManagerAddAndRegister0 IPsec에 TIA(신뢰할 수 있는 중간 에이전트)를 등록합니다.
IPsecKeyManagerGetSecurityInfoByKey0 키 관리자에 대한 액세스를 제어하는 보안 설명자의 복사본을 검색합니다.
IPsecKeyManagerSetSecurityInfoByKey0 키 관리자에 대한 액세스를 제어하는 보안 설명자에서 지정된 보안 정보를 설정합니다.
IPsecKeyManagersGet0 현재 신뢰할 수 있는 TIA(중간 에이전트) 목록을 반환합니다.

IPsecKeyManagerUnregisterAndDelete0 이전에 IPsec에 등록된 신뢰할 수 있는 TIA(중간 에이전트)를 등록 취소합니다.
IPsecSaContextAddInbound0 IPsecSaContextAddInbound0 함수는 기존 SA 컨텍스트에 인바운드 SA(IPsec 보안 연결) 번들을 추가합니다. 참고 IPsecSaContextAddInbound0은 Windows Vista에서 사용되는 IPsecSaContextAddInbound의 특정 구현입니다.
IPsecSaContextAddInbound1 IPsecSaContextAddInbound1 함수는 기존 SA 컨텍스트에 인바운드 SA(IPsec 보안 연결) 번들을 추가합니다. 참고 IPsecSaContextAddInbound1은 Windows 7 이상에서 사용되는 IPsecSaContextAddInbound의 특정 구현입니다.
IPsecSaContextAddOutbound0 IPsecSaContextAddOutbound0 함수는 기존 SA 컨텍스트에 아웃바운드 SA(IPsec 보안 연결) 번들을 추가합니다. 참고 IPsecSaContextAddOutbound0은 Windows Vista에서 사용되는 IPsecSaContextAddOutbound의 특정 구현입니다.
IPsecSaContextAddOutbound1 IPsecSaContextAddOutbound1 함수는 기존 SA 컨텍스트에 아웃바운드 SA(IPsec 보안 연결) 번들을 추가합니다. 참고 IPsecSaContextAddOutbound1은 Windows 7 이상에서 사용되는 IPsecSaContextAddOutbound의 특정 구현입니다.
IPsecSaContextCreate0 IPsec SA(보안 연결) 컨텍스트를 만듭니다. (IPsecSaContextCreate0)
IPsecSaContextCreate1 IPsec SA(보안 연결) 컨텍스트를 만듭니다. (IPsecSaContextCreate1)
IPsecSaContextCreateEnumHandle0 SA(IPsec 보안 연결) 컨텍스트 개체 집합을 열거하는 데 사용되는 핸들을 만듭니다.
IPsecSaContextDeleteById0 IPsec SA(보안 연결) 컨텍스트를 삭제합니다.
IPsecSaContextDestroyEnumHandle0 IPsecSaContextCreateEnumHandle0에서 반환된 핸들을 해제합니다.

IPsecSaContextEnum0 SA(IPsec 보안 연결) 컨텍스트 열거자에서 결과의 다음 페이지를 반환합니다. (IPsecSaContextEnum0)
IPsecSaContextEnum1 SA(IPsec 보안 연결) 컨텍스트 열거자에서 결과의 다음 페이지를 반환합니다. (IPsecSaContextEnum1)
IPsecSaContextExpire0 IPsec SA(보안 연결) 컨텍스트가 만료되어야 임을 나타냅니다.
IPsecSaContextGetById0 IPsec SA(보안 연결) 컨텍스트를 검색합니다. (IPsecSaContextGetById0)
IPsecSaContextGetById1 IPsec SA(보안 연결) 컨텍스트를 검색합니다. (IPsecSaContextGetById1)
IPsecSaContextGetSpi0 SA(보안 연결) 컨텍스트에 대한 SPI(보안 매개 변수 인덱스)를 검색합니다. (IPsecSaContextGetSpi0)
IPsecSaContextGetSpi1 SA(보안 연결) 컨텍스트에 대한 SPI(보안 매개 변수 인덱스)를 검색합니다. (IPsecSaContextGetSpi1)
IPsecSaContextSetSpi0 SA(보안 연결) 컨텍스트에 대한 SPI(보안 매개 변수 인덱스)를 설정합니다.
IPsecSaContextSubscribe0 특정 SA(IPsec 보안 연결) 컨텍스트와 관련된 알림 배달을 요청하는 데 사용됩니다.
IPsecSaContextSubscriptionsGet0 모든 현재 SA(IPsec 보안 연결) 변경 알림 구독의 배열을 검색합니다.
IPsecSaContextUnsubscribe0 SA(IPsec 보안 연결) 변경 구독을 취소하고 변경 알림 수신을 중지하는 데 사용됩니다.
IPsecSaContextUpdate0 IPsec SA(보안 연결) 컨텍스트를 업데이트합니다.

IPsecSaCreateEnumHandle0 IPsec(Internet Protocol Security) SA(보안 연결) 개체 집합을 열거하는 데 사용되는 핸들을 만듭니다.
IPsecSaDbGetSecurityInfo0 IPsec SA(보안 연결) 데이터베이스에 대한 보안 설명자의 복사본을 검색합니다.
IPsecSaDbSetSecurityInfo0 IPsec 보안 연결 데이터베이스의 보안 설명자에 지정된 보안 정보를 설정합니다.
IPsecSaDestroyEnumHandle0 IPsecSaCreateEnumHandle0에서 반환된 핸들을 해제합니다.
IPsecSaEnum0 SA(IPsec 보안 연결) 열거자에서 결과의 다음 페이지를 반환합니다. (IPsecSaEnum0)
IPsecSaEnum1 SA(IPsec 보안 연결) 열거자에서 결과의 다음 페이지를 반환합니다. (IPsecSaEnum1)

구조

테이블 확장

FWP_BYTE_ARRAY16 정확히 16바이트 배열을 저장합니다.
FWP_BYTE_ARRAY6 정확히 6바이트 배열을 저장합니다. 예약.
FWP_BYTE_BLOB 가변 바이트 수를 포함하는 배열을 저장합니다.
FWP_CONDITION_VALUE0 일치하는 필터를 테스트할 때 필터 조건에서 사용되는 값을 포함합니다.

FWP_RANGE0 값의 범위를 지정합니다.
FWP_TOKEN_INFORMATION FWP_TOKEN_INFORMATION 구조체는 사용자 모드 분류에 사용되는 보안 식별자 집합을 정의합니다.
FWP_V4_ADDR_AND_MASK 호스트 순서로 IPv4 주소 및 마스크를 지정합니다.
FWP_V6_ADDR_AND_MASK IPv6 주소와 마스크를 지정합니다.
FWP_VALUE0 다양한 데이터 형식 중 하나일 수 있는 데이터 값을 정의합니다.
FWPM_ACTION0 모든 필터 조건이 true인 경우 수행되는 작업을 지정합니다.
FWPM_CALLOUT_CHANGE0 구독자에게 디스패치되는 변경 알림을 지정합니다.
FWPM_CALLOUT_ENUM_TEMPLATE0 설명선 열거형을 제한하는 데 사용됩니다.
FWPM_CALLOUT_SUBSCRIPTION0 변경 알림을 구독하는 데 사용됩니다. (FWPM_CALLOUT_SUBSCRIPTION0)
FWPM_CALLOUT0 설명선과 연결된 상태를 저장합니다.
FWPM_CLASSIFY_OPTION0 FWPM_CLASSIFY_OPTION0 구조체입니다.
FWPM_CLASSIFY_OPTIONS0 FWPM_CLASSIFY_OPTIONS0 구조체는 FWPM_CLASSIFY_OPTION0 구조를 저장하는 데 사용됩니다.

FWPM_CONNECTION_ENUM_TEMPLATE0 연결 개체 열거를 제한하는 데 사용됩니다.
FWPM_CONNECTION_SUBSCRIPTION0 연결 개체에 대한 알림을 구독하는 데 사용되는 정보를 저장합니다.
FWPM_CONNECTION0 연결 개체와 연결된 상태를 저장합니다.
FWPM_DISPLAY_DATA0 개체에 대한 선택적 이름 및 선택적 설명을 저장합니다.
FWPM_FIELD0 필드에 대한 스키마 정보를 지정합니다.
FWPM_FILTER_CHANGE0 구독자에게 디스패치된 변경 알림을 저장합니다.
FWPM_FILTER_CONDITION0 수행할 작업에 대해 true여야 하는 필터 조건을 표현합니다.
FWPM_FILTER_ENUM_TEMPLATE0 필터를 열거하는 데 사용됩니다.
FWPM_FILTER_SUBSCRIPTION0 변경 알림을 구독하는 데 사용됩니다.
FWPM_FILTER0 필터와 연결된 상태를 저장합니다.
FWPM_LAYER_ENUM_TEMPLATE0 계층을 열거하는 데 사용됩니다.
FWPM_LAYER_STATISTICS0 계층과 관련된 통계를 저장합니다.
FWPM_LAYER0

계층에 대한 스키마 정보입니다.
FWPM_NET_EVENT_CAPABILITY_ALLOW0 앱 컨테이너 네트워크 기능과 관련하여 허용되는 네트워크 트래픽에 대한 정보를 포함합니다.
FWPM_NET_EVENT_CAPABILITY_DROP0 앱 컨테이너 네트워크 기능과 관련하여 삭제된 네트워크 트래픽에 대한 정보를 포함합니다.
FWPM_NET_EVENT_CLASSIFY_ALLOW0 WFP 분류 엔진에 의해 적용되는 허용된 트래픽을 설명하는 정보를 포함합니다.
FWPM_NET_EVENT_CLASSIFY_DROP_MAC0 MAC 계층 삭제 실패를 설명하는 정보를 포함합니다.
FWPM_NET_EVENT_CLASSIFY_DROP0 계층 삭제 실패를 설명하는 정보를 포함합니다. (FWPM_NET_EVENT_CLASSIFY_DROP0)
FWPM_NET_EVENT_CLASSIFY_DROP1 계층 삭제 실패를 설명하는 정보를 포함합니다. (FWPM_NET_EVENT_CLASSIFY_DROP1)
FWPM_NET_EVENT_CLASSIFY_DROP2 계층 삭제 실패를 설명하는 정보를 포함합니다. (FWPM_NET_EVENT_CLASSIFY_DROP2)
FWPM_NET_EVENT_ENUM_TEMPLATE0 net 이벤트를 열거하는 데 사용됩니다.
FWPM_NET_EVENT_HEADER0 모든 이벤트에 공통적인 정보입니다.
FWPM_NET_EVENT_HEADER1 모든 이벤트에 공통적인 정보입니다. 예약.
FWPM_NET_EVENT_HEADER2 모든 이벤트에 공통적인 정보를 포함합니다. (FWPM_NET_EVENT_HEADER2)
FWPM_NET_EVENT_HEADER3 모든 이벤트에 공통적인 정보를 포함합니다. (FWPM_NET_EVENT_HEADER3)

FWPM_NET_EVENT_IKEEXT_EM_FAILURE0 FWPM_NET_EVENT_IKEEXT_EM_FAILURE0 구조에는 IKE EM(확장 모드) 오류를 설명하는 정보가 포함되어 있습니다. 참고 FWPM_NET_EVENT_IKEEXT_EM_FAILURE0 Windows Vista에서 사용되는 FWPM_NET_EVENT_IKEEXT_EM_FAILURE 특정 구현입니다.
FWPM_NET_EVENT_IKEEXT_EM_FAILURE1 FWPM_NET_EVENT_IKEEXT_EM_FAILURE1 구조에는 IKE EM(확장 모드) 오류를 설명하는 정보가 포함되어 있습니다. FWPM_NET_EVENT_IKEEXT_EM_FAILURE1 Windows 7 이상에서 사용되는 FWPM_NET_EVENT_IKEEXT_EM_FAILURE 특정 구현입니다.
FWPM_NET_EVENT_IKEEXT_MM_FAILURE0 IKE/AuthIP MM(주 모드) 실패를 설명하는 정보를 포함합니다. (FWPM_NET_EVENT_IKEEXT_MM_FAILURE0)
FWPM_NET_EVENT_IKEEXT_MM_FAILURE1 IKE/AuthIP MM(주 모드) 실패를 설명하는 정보를 포함합니다. (FWPM_NET_EVENT_IKEEXT_MM_FAILURE1)
FWPM_NET_EVENT_IKEEXT_QM_FAILURE0 IKE/AuthIP QM(빠른 모드) 실패를 설명하는 정보를 포함합니다.
FWPM_NET_EVENT_IKEEXT_QM_FAILURE1 IKE/AuthIP QM(빠른 모드) 실패를 설명하는 정보를 포함합니다. FWPM_NET_EVENT_IKEEXT_QM_FAILURE0 사용할 수도 있습니다.
FWPM_NET_EVENT_IPSEC_DOSP_DROP0 IPsec DoS Protection 드롭 이벤트를 설명하는 정보를 포함합니다.
FWPM_NET_EVENT_IPSEC_KERNEL_DROP0 IPsec 커널 드롭 이벤트를 설명하는 정보를 포함합니다.
FWPM_NET_EVENT_LPM_PACKET_ARRIVAL0 IPSec SPI(보안 매개 변수 인덱스) 값을 설명하는 정보를 포함합니다.
FWPM_NET_EVENT_SUBSCRIPTION0 네트워크 이벤트에 대한 알림을 구독하는 데 사용되는 정보를 저장합니다.
FWPM_NET_EVENT0

모든 이벤트 유형에 대한 정보를 포함합니다. (FWPM_NET_EVENT0)
FWPM_NET_EVENT1 모든 이벤트 유형에 대한 정보를 포함합니다. (FWPM_NET_EVENT1)
FWPM_NET_EVENT2 모든 이벤트 유형에 대한 정보를 포함합니다. (FWPM_NET_EVENT2)
FWPM_NET_EVENT3 모든 이벤트 유형에 대한 정보를 포함합니다. (FWPM_NET_EVENT3)
FWPM_NET_EVENT4 모든 이벤트 유형에 대한 정보를 포함합니다. FWPM_NET_EVENT3 및 FWPM_NET_EVENT2 사용할 수 있습니다. Windows 7의 경우 FWPM_NET_EVENT1 사용할 수 있습니다. Windows Vista의 경우 FWPM_NET_EVENT0 사용할 수 있습니다.
FWPM_NET_EVENT5 모든 이벤트 유형에 대한 정보를 포함합니다. FWPM_NET_EVENT4, FWPM_NET_EVENT3 및 FWPM_NET_EVENT2 사용할 수 있습니다. Windows 7의 경우 FWPM_NET_EVENT1 사용할 수 있습니다. Windows Vista의 경우 FWPM_NET_EVENT0 사용할 수 있습니다.
FWPM_NETWORK_CONNECTION_POLICY_SETTING0 연결 정책 설정에 대한 형식 및 값 쌍을 저장합니다.
FWPM_NETWORK_CONNECTION_POLICY_SETTINGS0 해당 배열의 요소 수와 함께 FWPM_NETWORK_CONNECTION_POLICY_SETTING0 값의 배열을 저장합니다.
FWPM_PROVIDER_CHANGE0 구독자에게 디스패치된 알림을 변경합니다. (FWPM_PROVIDER_CHANGE0)
FWPM_PROVIDER_CONTEXT_CHANGE0 구독자에게 디스패치된 알림을 변경합니다. (FWPM_PROVIDER_CONTEXT_CHANGE0)
FWPM_PROVIDER_CONTEXT_ENUM_TEMPLATE0 공급자 컨텍스트를 열거하는 데 사용됩니다.
FWPM_PROVIDER_CONTEXT_SUBSCRIPTION0

변경 알림을 구독하는 데 사용됩니다. (FWPM_PROVIDER_CONTEXT_SUBSCRIPTION0)
FWPM_PROVIDER_CONTEXT0 공급자 컨텍스트와 연결된 상태를 저장합니다. (FWPM_PROVIDER_CONTEXT0)
FWPM_PROVIDER_CONTEXT1 공급자 컨텍스트와 연결된 상태를 저장합니다. (FWPM_PROVIDER_CONTEXT1)
FWPM_PROVIDER_CONTEXT2 공급자 컨텍스트와 연결된 상태를 저장합니다. (FWPM_PROVIDER_CONTEXT2)
FWPM_PROVIDER_CONTEXT3 공급자 컨텍스트와 연결된 상태를 저장합니다. FWPM_PROVIDER_CONTEXT0, FWPM_PROVIDER_CONTEXT1 및 FWPM_PROVIDER_CONTEXT2 사용할 수 있습니다.
FWPM_PROVIDER_ENUM_TEMPLATE0 공급자를 열거하는 데 사용됩니다.
FWPM_PROVIDER_SUBSCRIPTION0 변경 알림을 구독하는 데 사용됩니다. (FWPM_PROVIDER_SUBSCRIPTION0)
FWPM_PROVIDER0 정책 공급자와 연결된 상태를 저장합니다.
FWPM_SESSION_ENUM_TEMPLATE0 세션을 열거하는 데 사용됩니다.
FWPM_SESSION0 클라이언트 세션과 연결된 상태를 저장합니다.
FWPM_STATISTICS0 특정 계층에서 연결과 관련된 통계를 저장합니다.
FWPM_SUBLAYER_CHANGE0 구독자에게 디스패치된 알림을 변경합니다. (FWPM_SUBLAYER_CHANGE0)
FWPM_SUBLAYER_ENUM_TEMPLATE0 하위 계층을 열거하는 데 사용됩니다.

FWPM_SUBLAYER_SUBSCRIPTION0 변경 알림을 구독하는 데 사용됩니다. (FWPM_SUBLAYER_SUBSCRIPTION0)
FWPM_SUBLAYER0 하위 계층과 연결된 상태를 저장합니다.
FWPM_SYSTEM_PORTS_BY_TYPE0 FWPM_SYSTEM_PORTS_BY_TYPE0 구조체입니다.
FWPM_SYSTEM_PORTS0 FWPM_SYSTEM_PORTS0 구조체입니다.
FWPM_VSWITCH_EVENT_SUBSCRIPTION0 vSwitch 이벤트에 대한 알림을 구독하는 데 사용되는 정보를 저장합니다.
FWPM_VSWITCH_EVENT0 vSwitch 이벤트에 대한 정보를 포함합니다.
FWPS_INCOMING_METADATA_VALUES0 필터 엔진이 설명선의 classifyFn 설명선 함수에 전달하는 메타데이터 값을 정의합니다.
IKEEXT_AUTHENTICATION_METHOD0 IKE/AuthIP 인증에 대한 다양한 매개 변수를 지정합니다.
IKEEXT_AUTHENTICATION_METHOD1 IKE/Authip 인증에 대한 다양한 매개 변수를 지정합니다. (IKEEXT_AUTHENTICATION_METHOD1)
IKEEXT_AUTHENTICATION_METHOD2 IKE/Authip 인증에 대한 다양한 매개 변수를 지정합니다. (IKEEXT_AUTHENTICATION_METHOD2)
IKEEXT_CERT_EKUS0 인증서의 EKU(확장 키 사용) 속성에 대한 정보를 포함합니다.
IKEEXT_CERT_NAME0 인증 방법에 대한 인증서 선택 "주체" 조건을 지정합니다.
IKEEXT_CERT_ROOT_CONFIG0

IKE, AuthIP 또는 IKEv2 인증서 루트 구성을 저장합니다.
IKEEXT_CERTIFICATE_AUTHENTICATION0 인증서를 사용하여 인증하기 위한 다양한 매개 변수를 지정하는 데 사용됩니다. (IKEEXT_CERTIFICATE_AUTHENTICATION0)
IKEEXT_CERTIFICATE_AUTHENTICATION1 인증서를 사용하여 인증하기 위한 다양한 매개 변수를 지정하는 데 사용됩니다. (IKEEXT_CERTIFICATE_AUTHENTICATION1)
IKEEXT_CERTIFICATE_AUTHENTICATION2 인증서를 사용하여 인증하기 위한 다양한 매개 변수를 지정하는 데 사용됩니다. (IKEEXT_CERTIFICATE_AUTHENTICATION2)
IKEEXT_CERTIFICATE_CREDENTIAL0 인증서 인증과 관련된 자격 증명 정보를 저장하는 데 사용됩니다. (IKEEXT_CERTIFICATE_CREDENTIAL0)
IKEEXT_CERTIFICATE_CREDENTIAL1 인증서 인증과 관련된 자격 증명 정보를 저장하는 데 사용됩니다. (IKEEXT_CERTIFICATE_CREDENTIAL1)
IKEEXT_CERTIFICATE_CRITERIA0 인증 방법에 적용할 조건 집합을 포함합니다.
IKEEXT_CIPHER_ALGORITHM0 IKE/AuthIP 암호화 알고리즘에 대한 정보를 저장합니다.
IKEEXT_COMMON_STATISTICS0 IKE 및 Authip에 공통적인 다양한 통계입니다. (IKEEXT_COMMON_STATISTICS0)
IKEEXT_COMMON_STATISTICS1 IKE, Authip 및 IKEv2에 공통적인 다양한 통계입니다.
IKEEXT_COOKIE_PAIR0 IKE/Authip 쿠키 쌍을 저장하는 데 사용됩니다.
IKEEXT_CREDENTIAL_PAIR0

인증에 사용되는 자격 증명 정보를 저장하는 데 사용됩니다. (IKEEXT_CREDENTIAL_PAIR0)
IKEEXT_CREDENTIAL_PAIR1 인증에 사용되는 자격 증명 정보를 저장하는 데 사용됩니다. (IKEEXT_CREDENTIAL_PAIR1)
IKEEXT_CREDENTIAL_PAIR2 인증에 사용되는 자격 증명 정보를 저장하는 데 사용됩니다. (IKEEXT_CREDENTIAL_PAIR2)
IKEEXT_CREDENTIAL0 인증에 사용되는 자격 증명 정보를 저장하는 데 사용됩니다. (IKEEXT_CREDENTIAL0)
IKEEXT_CREDENTIAL1 인증에 사용되는 자격 증명 정보를 저장하는 데 사용됩니다. (IKEEXT_CREDENTIAL1)
IKEEXT_CREDENTIAL2 인증에 사용되는 자격 증명 정보를 저장하는 데 사용됩니다. (IKEEXT_CREDENTIAL2)
IKEEXT_CREDENTIALS0 여러 자격 증명 쌍을 저장하는 데 사용됩니다. (IKEEXT_CREDENTIALS0)
IKEEXT_CREDENTIALS1 여러 자격 증명 쌍을 저장하는 데 사용됩니다. (IKEEXT_CREDENTIALS1)
IKEEXT_CREDENTIALS2 여러 자격 증명 쌍을 저장하는 데 사용됩니다. (IKEEXT_CREDENTIALS2)
IKEEXT_EAP_AUTHENTICATION0 EAP(Extensible Authentication Protocol) 인증에 필요한 정보를 저장합니다.
IKEEXT_EM_POLICY0 AuthIP의 확장 모드 협상 정책을 저장하는 데 사용됩니다. (IKEEXT_EM_POLICY0)
IKEEXT_EM_POLICY1 AuthIP의 확장 모드 협상 정책을 저장하는 데 사용됩니다. (IKEEXT_EM_POLICY1)
IKEEXT_EM_POLICY2 AuthIP의 확장 모드 협상 정책을 저장하는 데 사용됩니다. (IKEEXT_EM_POLICY2)

IKEEXT_INTEGRITY_ALGORITHM0 IKE/AuthIP 해시 알고리즘을 저장합니다.
IKEEXT_IP_VERSION_SPECIFIC_COMMON_STATISTICS0 IKE 및 Authip에 공통적인 다양한 통계입니다. (IKEEXT_IP_VERSION_SPECIFIC_COMMON_STATISTICS0)
IKEEXT_IP_VERSION_SPECIFIC_COMMON_STATISTICS1 키 지정 모듈(IKE, Authip 및 IKEv2)에 공통적인 다양한 통계입니다.
IKEEXT_IP_VERSION_SPECIFIC_KEYMODULE_STATISTICS0 키 지정 모듈 및 IP 버전과 관련된 다양한 통계입니다.
IKEEXT_IP_VERSION_SPECIFIC_KEYMODULE_STATISTICS1 키 지정 모듈(IKE, Authip 및 IKEv2) 및 IP 버전과 관련된 다양한 통계입니다.
IKEEXT_IPV6_CGA_AUTHENTICATION0 IPV6 CGA(암호화로 생성된 주소) 인증에 대한 다양한 매개 변수를 지정하는 데 사용됩니다.
IKEEXT_KERBEROS_AUTHENTICATION0 미리 공유된 키 인증에 필요한 정보를 포함합니다. (IKEEXT_KERBEROS_AUTHENTICATION0)
IKEEXT_KERBEROS_AUTHENTICATION1 미리 공유된 키 인증에 필요한 정보를 포함합니다. (IKEEXT_KERBEROS_AUTHENTICATION1)
IKEEXT_KEYMODULE_STATISTICS0 키 지정 모듈과 관련된 다양한 통계를 포함합니다. (IKEEXT_KEYMODULE_STATISTICS0)
IKEEXT_KEYMODULE_STATISTICS1 키 지정 모듈과 관련된 다양한 통계를 포함합니다. (IKEEXT_KEYMODULE_STATISTICS1)
IKEEXT_NAME_CREDENTIAL0 자격 증명 이름 정보를 저장하는 데 사용됩니다.
IKEEXT_NTLM_V2_AUTHENTICATION0 Microsoft Windows NTLM(NT LAN Manager) V2 인증에 필요한 정보를 포함합니다.

IKEEXT_POLICY0 IKE/AuthIP 주 모드 협상 정책을 저장하는 데 사용됩니다. (IKEEXT_POLICY0)
IKEEXT_POLICY1 IKE/AuthIP 주 모드 협상 정책을 저장하는 데 사용됩니다. (IKEEXT_POLICY1)
IKEEXT_POLICY2 IKE/AuthIP 주 모드 협상 정책을 저장하는 데 사용됩니다. (IKEEXT_POLICY2)
IKEEXT_PRESHARED_KEY_AUTHENTICATION0 사전 공유 키 인증에 필요한 정보를 저장합니다. (IKEEXT_PRESHARED_KEY_AUTHENTICATION0)
IKEEXT_PRESHARED_KEY_AUTHENTICATION1 사전 공유 키 인증에 필요한 정보를 저장합니다. (IKEEXT_PRESHARED_KEY_AUTHENTICATION1)
IKEEXT_PROPOSAL0 IKE/AuthIP 기본 모드 제안을 저장하는 데 사용됩니다.
IKEEXT_RESERVED_AUTHENTICATION0 내부용으로 예약되어 있습니다.
IKEEXT_SA_DETAILS0 IKE, AuthIP 또는 IKEv2 SAS(보안 연결)를 열거할 때 반환되는 정보를 저장하는 데 사용됩니다.
IKEEXT_SA_DETAILS1 IKE, AuthIP 및 IKEv2 SAS(보안 연결)를 열거할 때 반환되는 정보를 저장하는 데 사용됩니다. (IKEEXT_SA_DETAILS1)
IKEEXT_SA_DETAILS2 IKE, AuthIP 및 IKEv2 SAS(보안 연결)를 열거할 때 반환되는 정보를 저장하는 데 사용됩니다. (IKEEXT_SA_DETAILS2)
IKEEXT_SA_ENUM_TEMPLATE0 IKE/AuthIP SA(보안 연결)를 열거하는 데 사용되는 열거형 템플릿입니다.
IKEEXT_STATISTICS0 다양한 IKE/AuthIP 통계를 저장합니다.

IKEEXT_STATISTICS1 다양한 IKE, AuthIP 및 IKEv2 통계를 저장합니다.
IKEEXT_TRAFFIC0 IKE/Authip 트래픽을 지정합니다.
IPSEC_ADDRESS_INFO0 모바일 추가 주소 정보를 저장하는 데 사용됩니다.
IPSEC_AGGREGATE_DROP_PACKET_STATISTICS0 집계 IPsec 커널 패킷 삭제 통계를 저장합니다. 참고 IPSEC_AGGREGATE_DROP_PACKET_STATISTICS0 Windows Vista에서 사용되는 IPSEC_AGGREGATE_DROP_PACKET_STATISTICS 특정 구현입니다.
IPSEC_AGGREGATE_DROP_PACKET_STATISTICS1 집계 IPsec 커널 패킷 삭제 통계를 저장합니다.
IPSEC_AGGREGATE_SA_STATISTICS0 집계 IPsec SA(커널 보안 연결) 통계를 저장합니다.
IPSEC_AH_DROP_PACKET_STATISTICS0 IPsec AH 드롭 패킷 통계를 저장합니다.
IPSEC_AUTH_AND_CIPHER_TRANSFORM0 IPsec 빠른 모드 정책에서 SA 변환을 위해 해시 및 암호화 특정 정보를 함께 저장하는 데 사용됩니다.
IPSEC_AUTH_TRANSFORM_ID0 IPsec SA(보안 연결)에서 사용되는 해시 알고리즘을 고유하게 식별하는 데 사용됩니다.
IPSEC_AUTH_TRANSFORM0 SA 변환에 대한 해시 관련 정보를 지정합니다.
IPSEC_CIPHER_TRANSFORM_ID0 IPsec SA에서 사용되는 암호화 알고리즘을 고유하게 식별하는 데 사용되는 정보를 지정합니다.
IPSEC_CIPHER_TRANSFORM0

IPsec 빠른 모드 정책에서 SA 변환에 대한 암호화 특정 정보를 저장하는 데 사용됩니다.
IPSEC_DOSP_OPTIONS0 IPsec DoS Protection에 대한 구성 매개 변수를 저장하는 데 사용됩니다.
IPSEC_DOSP_STATE_ENUM_TEMPLATE0 IPSEC_DOSP_STATE_ENUM_TEMPLATE0 구조체입니다.
IPSEC_DOSP_STATE0 IPsec DoS Protection에 대한 상태 정보를 저장하는 데 사용됩니다.
IPSEC_DOSP_STATISTICS0 IPSEC_DOSP_STATISTICS0 구조체입니다.
IPSEC_ESP_DROP_PACKET_STATISTICS0 ESP 드롭 패킷 통계를 저장합니다.
IPSEC_GETSPI0 IPSEC_GETSPI0 구조에는 IPsec 드라이버에서 SPI(보안 매개 변수 인덱스)를 요청할 때 제공해야 하는 정보가 포함되어 있습니다. 참고 IPSEC_GETSPI0 Windows Vista에서 사용되는 IPSEC_GETSPI 특정 구현입니다.
IPSEC_GETSPI1 IPSEC_GETSPI1 구조에는 IPsec 드라이버에서 SPI(보안 매개 변수 인덱스)를 요청할 때 제공해야 하는 정보가 포함되어 있습니다. 참고 IPSEC_GETSPI1 Windows 7 이상에서 사용되는 IPSEC_GETSPI 특정 구현입니다.
IPSEC_ID0 IPsec에서 인증하는 ID에 해당하는 정보를 포함합니다.
IPSEC_KEY_MANAGER_CALLBACKS0 SA 협상의 다양한 단계에서 IPsec에서 호출해야 하는 콜백 집합을 지정합니다.
IPSEC_KEY_MANAGER0 IPsec에 키 관리 콜백을 등록하는 데 사용됩니다.
IPSEC_KEYING_POLICY0

IPsec에 대해 시도할 순서가 지정되지 않은 키 지정 모듈 집합을 정의합니다. (IPSEC_KEYING_POLICY0)
IPSEC_KEYING_POLICY1 IPsec에 대해 시도할 순서가 지정되지 않은 키 지정 모듈 집합을 정의합니다. (IPSEC_KEYING_POLICY1)
IPSEC_KEYMODULE_STATE0 IPsec(인터넷 프로토콜 보안) 키 지정 모듈 관련 정보를 저장합니다.
IPSEC_PROPOSAL0 IPsec 빠른 모드 제안을 저장하는 데 사용됩니다.
IPSEC_SA_AUTH_AND_CIPHER_INFORMATION0 IPsec SA(보안 연결)의 인증 및 암호화 알고리즘에 대한 정보를 저장합니다.
IPSEC_SA_AUTH_INFORMATION0 IPsec SA(보안 연결)의 인증 알고리즘에 대한 정보를 저장합니다.
IPSEC_SA_BUNDLE0 IPsec SA(보안 연결) 번들에 대한 정보를 저장하는 데 사용됩니다. (IPSEC_SA_BUNDLE0)
IPSEC_SA_BUNDLE1 IPsec SA(보안 연결) 번들에 대한 정보를 저장하는 데 사용됩니다. (IPSEC_SA_BUNDLE1)
IPSEC_SA_CIPHER_INFORMATION0 IPsec SA(보안 연결)의 암호화 알고리즘에 대한 정보를 저장합니다.
IPSEC_SA_CONTEXT_CHANGE0 IPsec SA(보안 연결) 컨텍스트 변경에 대한 정보를 포함합니다.
IPSEC_SA_CONTEXT_ENUM_TEMPLATE0 SA(보안 연결) 컨텍스트를 열거하는 데 사용되는 열거형 템플릿입니다.
IPSEC_SA_CONTEXT_SUBSCRIPTION0 특정 SA(IPsec 보안 연결) 컨텍스트에 대한 알림을 구독하는 데 사용되는 정보를 저장합니다.
IPSEC_SA_CONTEXT0

인바운드 및 아웃바운드 SA 쌍을 캡슐화합니다.
IPSEC_SA_CONTEXT1 인바운드 및 아웃바운드 SA(보안 연결) 쌍을 캡슐화합니다.
IPSEC_SA_DETAILS0 IPsec SAS(보안 연결)를 열거할 때 반환되는 정보를 저장하는 데 사용됩니다. (IPSEC_SA_DETAILS0)
IPSEC_SA_DETAILS1 IPsec SAS(보안 연결)를 열거할 때 반환되는 정보를 저장하는 데 사용됩니다. (IPSEC_SA_DETAILS1)
IPSEC_SA_ENUM_TEMPLATE0 IPsec SAS(보안 연결)의 열거를 제한하는 데 사용되는 템플릿을 지정합니다.
IPSEC_SA_IDLE_TIMEOUT0 IPsec 정책의 SA(보안 연결) 유휴 시간 제한입니다.
IPSEC_SA_LIFETIME0 IPsec SA(보안 연결)의 수명(초/킬로바이트/패킷)을 저장합니다.
IPSEC_SA_TRANSFORM0 IPsec 빠른 모드 정책에 IPsec SA(보안 연결) 변환을 저장하는 데 사용됩니다.
IPSEC_SA0 IPsec SA(보안 연결)에 대한 정보를 저장하는 데 사용됩니다.
IPSEC_STATISTICS0 IPsec 통계 구조의 최상위 수준입니다. (IPSEC_STATISTICS0)
IPSEC_STATISTICS1 IPsec 통계 구조의 최상위 수준입니다. (IPSEC_STATISTICS1)
IPSEC_TOKEN0 IPsec별 액세스 토큰에 대한 다양한 정보입니다.
IPSEC_TRAFFIC_STATISTICS0 IPsec 트래픽 통계를 저장합니다. (IPSEC_TRAFFIC_STATISTICS0)

IPSEC_TRAFFIC_STATISTICS1 IPsec 트래픽 통계를 저장합니다. (IPSEC_TRAFFIC_STATISTICS1)
IPSEC_TRAFFIC0 IPsec 트래픽을 설명하는 매개 변수를 지정합니다. (IPSEC_TRAFFIC0)
IPSEC_TRAFFIC1 IPsec 트래픽을 설명하는 매개 변수를 지정합니다. (IPSEC_TRAFFIC1)
IPSEC_TRANSPORT_POLICY0 전송 모드 IPsec에 대한 빠른 모드 협상 정책을 저장합니다. (IPSEC_TRANSPORT_POLICY0)
IPSEC_TRANSPORT_POLICY1 전송 모드 IPsec에 대한 빠른 모드 협상 정책을 저장합니다. (IPSEC_TRANSPORT_POLICY1)
IPSEC_TRANSPORT_POLICY2 전송 모드 IPsec에 대한 빠른 모드 협상 정책을 저장합니다. (IPSEC_TRANSPORT_POLICY2)
IPSEC_TUNNEL_ENDPOINT0 터널 모드 SA의 끝점에 대한 주소 정보를 저장하는 데 사용됩니다.
IPSEC_TUNNEL_ENDPOINTS0 터널 모드 SA의 끝점을 저장하는 데 사용됩니다. (IPSEC_TUNNEL_ENDPOINTS0)
IPSEC_TUNNEL_ENDPOINTS1 터널 모드 SA의 끝점을 저장하는 데 사용됩니다. (IPSEC_TUNNEL_ENDPOINTS1)
IPSEC_TUNNEL_ENDPOINTS2 터널 모드 SA의 끝점을 저장하는 데 사용됩니다. (IPSEC_TUNNEL_ENDPOINTS2)
IPSEC_TUNNEL_POLICY0 터널 모드 IPsec에 대한 빠른 모드 협상 정책을 저장합니다. (IPSEC_TUNNEL_POLICY0)
IPSEC_TUNNEL_POLICY1 터널 모드 IPsec에 대한 빠른 모드 협상 정책을 저장합니다. (IPSEC_TUNNEL_POLICY1)
IPSEC_TUNNEL_POLICY2

터널 모드 IPsec에 대한 빠른 모드 협상 정책을 저장합니다. (IPSEC_TUNNEL_POLICY2)
IPSEC_V4_UDP_ENCAPSULATION0 ESP(보안 페이로드) 캡슐화를 캡슐화하기 위한 UDP(사용자 데이터그램 프로토콜) 캡슐화 포트를 저장합니다.
IPSEC_VIRTUAL_IF_TUNNEL_INFO0 가상 인터페이스 터널링과 관련된 정보를 저장하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

Windows 방화벽 기술

아티클2023. 03. 09.

Windows 방화벽 기술 개요입니다.

Windows 방화벽 기술을 개발하려면 다음 헤더가 필요합니다.

- icftypes.h
- natupnp.h
- netcon.h
- netfw.h
- networkisolation.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 방화벽 기술

열거형

FW_DYNAMIC_KEYWORD_ADDRESS_ENUM_FLAGS 열거형 작업에 포함할 개체의 종류를 지정하는 상수를 정의합니다.
FW_DYNAMIC_KEYWORD_ADDRESS_FLAGS IP 주소를 확인하는 방법을 지정하는 상수를 정의합니다.
FW_DYNAMIC_KEYWORD_ORIGIN_TYPE **FW_DYNAMIC_KEYWORD_ADDRESS_DATA0** 동적 키워드 주소 개체의 원본을 지정하는 상수를 정의합니다.
INET_FIREWALL_AC_CHANGE_TYPE INET_FIREWALL_AC_CHANGE_TYPE 열거형은 발생한 앱 컨테이너 변경 유형을 지정합니다. (INET_FIREWALL_AC_CHANGE_TYPE)
INET_FIREWALL_AC_CHANGE_TYPE INET_FIREWALL_AC_CHANGE_TYPE 열거형은 발생한 앱 컨테이너 변경 유형을 지정합니다. (INET_FIREWALL_AC_CHANGE_TYPE)

INET_FIREWALL_AC_CREATION_TYPE INET_FIREWALL_AC_CREATION_TYPE 열거형은 알림이 전달될 앱 컨테이너 만들기 이벤트의 유형을 지정합니다.
INET_FIREWALL_AC_CREATION_TYPE INET_FIREWALL_AC_CREATION_TYPE 열거형은 알림이 전달될 앱 컨테이너 만들기 이벤트의 유형을 지정합니다.
NET_FW_ACTION 규칙 또는 기본 설정에 대한 작업을 지정합니다. (NET_FW_ACTION)
NET_FW_AUTHENTICATE_TYPE 트래픽을 허용하기 위해 발생해야 하는 인증 유형을 지정합니다.
NET_FW_EDGE_TRAVERSAL_TYPE 에지 통과 트래픽이 허용되는 조건입니다.
NET_FW_IP_PROTOCOL 인터넷 프로토콜을 지정합니다.
NET_FW_IP_VERSION 포트의 IP 버전을 지정합니다.
NET_FW_MODIFY_STATE 현재 정책에 대한 수정의 효과를 지정합니다.
NET_FW_POLICY_TYPE NET_FW_POLICY_TYPE 열거형 형식은 정책 유형을 지정합니다.
NET_FW_PROFILE_TYPE 프로필 유형을 지정합니다. (NET_FW_PROFILE_TYPE)
NET_FW_PROFILE_TYPE2 프로필 유형을 지정합니다. (NET_FW_PROFILE_TYPE2)
NET_FW_RULE_CATEGORY 방화벽 규칙 범주입니다.

NET_FW_RULE_DIRECTION 규칙이 적용되는 트래픽의 방향을 지정합니다.
NET_FW_SCOPE 포트에서 수신 대기할 수 있는 주소의 범위를 지정합니다.
NET_FW_SERVICE_TYPE 서비스 유형을 지정합니다.
NETCON_CHARACTERISTIC_FLAGS NETCON_CHARACTERISTIC_FLAGS 열거형 유형은 네트워크 연결에 대해 가능한 특성을 지정합니다.
NETCON_MEDIATYPE NETCON_MEDIATYPE 값은 컴퓨터가 네트워크에 연결할 수 있는 방법을 열거합니다.
NETCON_STATUS NETCON_STATUS 형식은 네트워크 연결에 대해 가능한 상태 조건을 열거합니다.
NETCON_TYPE NETCON_TYPE 형식은 다양한 종류의 네트워크 연결을 열거합니다.
NETISO_ERROR_TYPE NETISO_ERROR_TYPE 열거형은 네트워크 격리 작업과 관련된 오류 유형을 지정합니다.
NETISO_ERROR_TYPE NETISO_ERROR_TYPE 열거형은 네트워크 격리 작업과 관련된 오류 유형을 지정합니다. (NETISO_ERROR_TYPE)
NETISO_FLAG NETISO_FLAG 열거형은 앱 컨테이너에 대해 이진 파일을 반환해야 하는지 여부를 지정합니다. (NETISO_FLAG)
NETISO_FLAG NETISO_FLAG 열거형은 앱 컨테이너에 대해 이진 파일을 반환해야 하는지 여부를 지정합니다. (NETISO_FLAG)

SHARINGCONNECTIONTYPE SHARINGCONNECTIONTYPE 형식의 값은 가능한 유형의 공유 연결을 열거합니다.

함수

추가 Add 메서드는 새 포트 매핑을 만들고 컬렉션에 추가합니다.
추가 Add 메서드는 컬렉션에 새 애플리케이션을 추가합니다.
추가 새 포트를 열고 컬렉션에 추가합니다.
추가 Add 메서드는 컬렉션에 새 규칙을 추가합니다.
AddPortMapping 이 연결에 대한 서비스 포트 매핑을 추가합니다.
복제 Clone 메서드는 이 열거형에서 새 열거형 인터페이스를 만듭니다. (IEnumNetSharingEveryConnection.Clone)
복제 Clone 메서드는 이 열거형에서 새 열거형 인터페이스를 만듭니다. (IEnumNetSharingPortMapping.Clone)
복제 Clone 메서드는 이 열거형에서 새 열거형 인터페이스를 만듭니다. (IEnumNetSharingPrivateConnection.Clone)
복제 Clone 메서드는 이 열거형에서 새 열거형 인터페이스를 만듭니다. (IEnumNetSharingPublicConnection.Clone)

연결 Connect 메서드는 이 네트워크 연결을 연결하거나 설정합니다.
삭제 Delete 메서드는 연결 폴더에서 이 연결을 삭제합니다.
삭제 Delete 메서드는 특정 연결에 대한 포트 매핑 목록에서 포트 매핑을 삭제합니다.
사용 안 함 Disable 메서드는 특정 연결에 대한 포트 매핑을 사용하지 않도록 설정합니다.
DisableInternetFirewall DisableInternetFirewall 메서드는 이 연결에서 인터넷 연결 방화벽을 사용하지 않도록 설정합니다.
DisableSharing DisableSharing 메서드는 이 연결에서 공유를 사용하지 않도록 설정합니다. 또한 이 연결에 대한 모든 매핑을 사용하지 않도록 설정합니다. 인터넷 연결 방화벽 또는 브리지 구성을 사용하지 않도록 설정하지 않습니다.
연결 끊기 Disconnect 메서드는 이 연결의 연결을 끊습니다.
중복 Duplicate 메서드는 연결 폴더에 이 연결의 중복을 만듭니다.
EditDescription EditDescription 메서드는 이 포트 매핑의 설명 속성을 지정된 값으로 설정합니다.
EditInternalClient EditInternalClient 메서드는 이 포트 매핑의 내부 클라이언트 속성을 지정된 값으로 설정합니다.
EditInternalPort EditInternalPort 메서드는 이 포트 매핑에 대한 내부 포트를 설정합니다.
사용 Enable 메서드는 이 포트 매핑을 사용하거나 사용하지 않도록 설정합니다.

사용 Enable 메서드는 특정 연결에 대한 포트 매핑을 사용하도록 설정합니다.
EnableInternetFirewall EnableInternetFirewall 메서드는 이 연결에서 인터넷 연결 방화벽을 사용하도록 설정합니다.
EnableRuleGroup 지정된 방화벽 규칙 그룹을 사용하거나 사용하지 않도록 설정합니다.
EnableSharing EnableSharing 메서드를 사용하면 이 연결에서 공유할 수 있습니다.
get__NewEnum get__NewEnum 메서드는 정적 포트 매핑 컬렉션에 대한 열거자를 검색합니다.
get__NewEnum get__NewEnum 메서드는 연결 컬렉션에 대한 열거자를 검색합니다.
get__NewEnum get__NewEnum 메서드는 포트 매핑 컬렉션에 대한 열거자를 검색합니다.
get__NewEnum get__NewEnum 메서드는 프라이빗 연결 컬렉션에 대한 열거자를 검색합니다.
get__NewEnum get__NewEnum 메서드는 공용 연결 컬렉션에 대한 열거자를 검색합니다.
get__NewEnum 컬렉션의 모든 애플리케이션을 반복하는 데 사용할 수 있는 IEnumVARIANT를 지원하는 개체를 반환합니다.
get__NewEnum 컬렉션의 모든 포트를 반복하는 데 사용할 수 있는 IEnumVARIANT를 지원하는 개체를 반환합니다.
get__NewEnum 컬렉션에 등록된 모든 타사 방화벽 제품을 반복하는 데 사용할 수 있는 IEnumVARIANT를 지원하는 개체를 반환합니다.

get__NewEnum 컬렉션의 모든 규칙을 반복하는 데 사용할 수 있는 IEnumVARIANT를 지원하는 개체를 반환합니다.
get__NewEnum 컬렉션의 모든 서비스를 반복하는 데 사용할 수 있는 IEnumVARIANT를 지원하는 개체를 반환합니다.
get_Action 규칙 또는 기본 설정에 대한 작업을 지정합니다. (INetFwRule.get_Action)
get_AllowInboundEchoRequest 허용되는지 여부를 나타냅니다. (INetFwIcmpSettings.get_AllowInboundEchoRequest)
get_AllowInboundMaskRequest 허용되는지 여부를 나타냅니다. (INetFwIcmpSettings.get_AllowInboundMaskRequest)
get_AllowInboundRouterRequest 허용되는지 여부를 나타냅니다. (INetFwIcmpSettings.get_AllowInboundRouterRequest)
get_AllowInboundTimestampRequest 허용되는지 여부를 나타냅니다. (INetFwIcmpSettings.get_AllowInboundTimestampRequest)
get_AllowOutboundDestinationUnreachable 허용되는지 여부를 나타냅니다. (INetFwIcmpSettings.get_AllowOutboundDestinationUnreachable)
get_AllowOutboundPacketTooBig 허용되는지 여부를 나타냅니다. (INetFwIcmpSettings.get_AllowOutboundPacketTooBig)
get_AllowOutboundParameterProblem 허용되는지 여부를 나타냅니다. (INetFwIcmpSettings.get_AllowOutboundParameterProblem)
get_AllowOutboundSourceQuench 아웃바운드 원본 쿼리가 허용되는지 여부를 나타냅니다. (Get)
get_AllowOutboundTimeExceeded 아웃바운드 시간을 초과할 수 있는지 여부를 나타냅니다. (Get)

get_AllowRedirect 리디렉션이 허용되는지 여부를 나타냅니다. (Get)
get_ApplicationName 이 규칙이 적용되는 애플리케이션의 이름을 지정합니다. (Get)
get_AuthorizedApplications 프로필의 권한 있는 애플리케이션 컬렉션을 검색합니다.
get_BlockAllInboundTraffic 방화벽에서 인바운드 트래픽을 허용하지 않아야 하는지 여부를 나타냅니다. (Get)
get_BuiltIn 포트가 시스템에 의해 정의되는지 여부를 나타냅니다.
get_Characteristics get_Characteristics 메서드는 연결에 대한 미디어 형식을 검색합니다.
get_Count get_Count 메서드는 컬렉션의 포트 매핑 수를 검색합니다.
get_Count get__Count 메서드는 연결 컬렉션의 항목 수를 검색합니다.
get_Count get__Count 메서드는 포트 매핑 컬렉션의 항목 수를 검색합니다.
get_Count get__Count 메서드는 프라이빗 연결 컬렉션의 항목 수를 검색합니다.
get_Count get_Count 메서드는 공용 연결 컬렉션의 항목 수를 검색합니다.
get_Count 컬렉션의 항목 수를 지정합니다.

get_Count 컬렉션의 항목 수를 산출하는 읽기 전용 요소를 검색합니다. (INetFwOpenPorts.get_Count)
get_Count 등록된 타사 방화벽 제품 수를 나타냅니다.
get_Count 컬렉션의 규칙 수를 반환합니다.
get_Count 컬렉션의 항목 수를 산출하는 읽기 전용 요소를 검색합니다. (INetFwServices.get_Count)
get_CurrentProfile 현재 방화벽 프로필을 검색합니다.
get_CurrentProfileType 현재 적용 중인 방화벽 프로필의 유형을 검색합니다.
get_CurrentProfileTypes 현재 활성 방화벽 프로필을 검색합니다.
get_Customized 서비스와 연결된 포트 중 하나 이상이 사용자 지정되었는지 여부를 나타냅니다.
get_DefaultInboundAction 인바운드 트래픽에 대한 기본 작업을 지정합니다. 이러한 설정은 기본적으로 차단입니다. (Get)
get_DefaultOutboundAction 아웃바운드 트래픽에 대한 기본 작업을 지정합니다. 이러한 설정은 기본적으로 허용입니다. (Get)
get_Description get_Description 메서드는 이 포트 매핑과 연결된 설명을 검색합니다.
get_Description 이 규칙에 대한 설명을 지정합니다. (Get)

get_DeviceName get_DeviceName 메서드는 연결과 연결된 디바이스의 이름을 검색합니다.
get_Direction 규칙이 적용되는 트래픽의 방향을 지정합니다. (Get)
get_DisplayName 타사 방화벽 제품 등록의 표시 이름을 나타냅니다. (Get)
get_EdgeTraversal 이 규칙에 대해 에지 순회를 사용할지 여부를 나타냅니다. (Get)
get_EdgeTraversalOptions 이 속성은 NET_FW_EDGE_TRAVERSAL_TYPE 정의된 방화벽 규칙의 에지 속성에 액세스하는 데 사용할 수 있습니다. (Get)
get_Enabled get_Enabled 메서드는 포트 매핑을 사용할 수 있는지 여부를 검색합니다.
get_Enabled get_Enabled 메서드는 이 포트 매핑의 상태를 검색합니다.
get_Enabled 이 애플리케이션에 대한 설정이 현재 사용하도록 설정되어 있는지 여부를 나타냅니다. (Get)
get_Enabled 이 포트에 대한 설정이 현재 사용하도록 설정되어 있는지 여부를 나타냅니다. (Get)
get_Enabled 원격 관리를 사용할 수 있는지 여부를 나타냅니다. (Get)
get_Enabled 규칙을 사용하거나 사용하지 않도록 설정합니다. (Get)
get_Enabled 서비스와 연결된 모든 포트를 사용할 수 있는지 여부를 나타냅니다. (Get)

get_EnumEveryConnection get_EnumEveryConnection 메서드는 연결 폴더의 모든 연결에 대한 열거형 인터페이스를 검색합니다.
get_EnumPortMappings get_EnumPortMappings 메서드는 IEnumNetSharingPortMapping 인터페이스를 검색합니다. 이 인터페이스를 사용하여 이 연결에 대한 포트 매핑을 열거합니다.
get_EnumPrivateConnections get_EnumPrivateConnections 메서드는 비공개 공유 연결에 대한 열거형 인터페이스를 검색합니다.
get_EnumPublicConnections EnumPublicConnections 메서드는 공개적으로 공유된 연결에 대한 열거형 인터페이스를 검색합니다.
get_ExceptionsNotAllowed 방화벽에서 예외를 허용하지 않아야 하는지 여부를 나타냅니다. (Get)
get_ExcludedInterfaces 방화벽 설정이 제외되는 인터페이스 목록을 지정합니다. (Get)
get_ExternalIPAddress get_ExternalIPAddress 메서드는 NAT 컴퓨터에서 이 포트 매핑에 대한 외부 IP 주소를 검색합니다.
get_ExternalPort get_ExternalPort 메서드는 이 포트 매핑을 위해 NAT 컴퓨터의 외부 포트를 검색합니다.
get_ExternalPort get_ExternalPort 메서드는 이 포트 매핑과 연결된 외부 포트를 검색합니다.
get_FirewallEnabled 방화벽이 로컬로 사용하도록 설정되어 있는지 여부를 나타냅니다(그룹 정책 설정으로 인해 유효한 결과가 다를 수 있음). (Get)
get_FirewallEnabled 방화벽을 사용할 수 있는지 여부를 나타냅니다. (Get)

get_GloballyOpenPorts 프로필의 전역적으로 열려 있는 포트의 컬렉션을 검색합니다.
get_GloballyOpenPorts 서비스와 연결된 전역으로 열려 있는 포트의 컬렉션을 검색합니다.
get_Grouping 개별 규칙이 속한 그룹을 지정합니다. (Get)
get_Guid get_Guid 메서드는 연결에 대한 GUID(Globally Unique Identifier)를 검색합니다.
get_IcmpSettings 프로필의 ICMP 설정을 검색합니다.
get_IcmpTypesAndCodes 이 규칙의 ICMP 형식 및 코드 목록을 지정합니다. (Get)
get_INetSharingConfigurationForINetConnection get_INetSharingConfigurationForINetConnection 메서드는 지정된 연결에 대한 INetSharingConfiguration 인터페이스를 검색합니다.
get_Interfaces 규칙이 적용되는 인터페이스 목록을 지정합니다. (Get)
get_InterfaceTypes 규칙이 적용되는 인터페이스 형식의 목록을 지정합니다. (Get)
get_InternalClient get_InternalClient 메서드는 이 포트 매핑에 대한 내부 클라이언트의 이름을 검색합니다.
get_InternalPort get_InternalPort 메서드는 이 포트 매핑을 위해 클라이언트 컴퓨터의 내부 포트를 검색합니다.
get_InternalPort get_InternalPort 메서드는 이 포트 매핑과 연결된 내부 포트를 검색합니다.

get_InternetFirewallEnabled get_InternetFirewallEnabled 메서드는 이 연결에서 인터넷 연결 방화벽을 사용할 수 있는지 여부를 결정합니다.
get_IPProtocol get_IPProtocol 메서드는 이 포트 매핑과 연결된 IP 프로토콜을 검색합니다.
get_IpVersion 이 애플리케이션의 IP 버전 설정을 지정합니다. (Get)
get_IpVersion 이 포트에 대한 IP 버전 설정을 지정합니다. (Get)
get_IpVersion IP 버전을 지정합니다. (INetFwRemoteAdminSettings.get_IpVersion)
get_IpVersion 서비스에 권한이 부여된 방화벽 IP 버전을 지정합니다. (Get)
get_IsRuleGroupCurrentlyEnabled 지정된 방화벽 규칙 그룹을 사용할지 여부를 결정합니다. (INetFwPolicy2.get_IsRuleGroupCurrentlyEnabled)
get_Item get_Item 메서드는 컬렉션에서 지정된 포트 매핑을 검색합니다.
get_LocalAddresses 이 규칙의 로컬 주소 목록을 지정합니다. (Get)
get_LocalAppPackageId Windows 스토어 앱이든 데스크톱 앱이든 관계없이 프로세스의 패키지 식별자 또는 앱 컨테이너 식별자를 지정합니다. (Get)
get_LocalPolicy 로컬 방화벽 정책을 검색합니다.

get_LocalPolicyModifyState 규칙 또는 규칙 그룹을 추가하거나 설정하는 것이 현재 방화벽 프로필에 적용되는지 여부를 결정합니다.
get_LocalPorts 이 규칙의 로컬 포트 목록을 지정합니다. (Get)
get_LocalUserAuthorizedList 앱 컨테이너에 대한 권한 있는 로컬 사용자 목록을 지정합니다. (Get)
get_LocalUserOwner 규칙의 소유자인 사용자의 SID(사용자 보안 식별자)를 지정합니다. (Get)
get_MediaType get_MediaType 메서드는 연결에 대한 미디어 형식을 검색합니다.
get_Name get_Name 메서드는 연결의 이름을 검색합니다.
get_Name get_Name 메서드는 이 포트 매핑의 이름을 검색합니다.
get_Name 이 애플리케이션의 이름을 지정합니다. (Get)
get_Name 이 포트의 이름을 지정합니다. (Get)
get_Name 이 규칙의 이름을 지정합니다. (Get)
get_Name 서비스의 이름을 검색합니다.
get_NATEventManager get_NATEventManager 메서드는 로컬 컴퓨터에서 사용하는 NAT에 대한 INATEventManager 인터페이스를 검색합니다.

get_NetConnectionProps get_NetConnectionProps 메서드는 지정된 연결에 대한 속성 인터페이스를 검색합니다.
get_NotificationsDisabled 대화형 방화벽 알림을 사용하지 않도록 설정할지 여부를 나타냅니다. (INetFwPolicy2.get_NotificationsDisabled)
get_NotificationsDisabled 대화형 방화벽 알림을 사용하지 않도록 설정할지 여부를 나타냅니다. (INetFwProfile.get_NotificationsDisabled)
get_Options get_Options 메서드는 이 포트 매핑과 관련된 옵션을 검색합니다.
get_PathToSignedProductExe 타사 방화벽 제품 등록의 서명된 실행 파일 경로를 나타냅니다.
get_Port 이 포트에 대한 호스트 순서 포트 번호를 지정합니다. (Get)
get_ProcessImageFileName 이 애플리케이션의 프로세스 이미지 파일 이름을 지정합니다. (Get)
get_Profiles 규칙이 속한 프로필을 지정합니다. (Get)
get_Properties get_Properties 메서드는 포트 매핑에 대한 속성을 검색합니다.
get_Protocol get_Protocol 메서드는 이 포트 매핑과 연결된 프로토콜을 검색합니다.
get_Protocol 이 포트에 대한 프로토콜 형식 설정을 지정합니다. (Get)
get_Protocol 이 규칙의 IP 프로토콜을 지정합니다. (Get)

get_RemoteAddresses 애플리케이션이 트래픽을 수신 대기할 수 있는 원격 주소 집합을 지정합니다. (Get)
get_RemoteAddresses 포트에서 트래픽을 수신 대기할 수 있는 원격 주소 집합을 지정합니다. (Get)
get_RemoteAddresses 원격 관리가 허용되는 원격 주소 집합을 지정합니다. (Get)
get_RemoteAddresses 이 규칙의 원격 주소 목록을 지정합니다. (Get)
get_RemoteAddresses 서비스 포트가 트래픽을 수신 대기할 수 있는 원격 주소 집합을 지정합니다. (Get)
get_RemoteAdminSettings 원격 관리를 제어하는 설정을 지정합니다.
get_RemoteMachineAuthorizedList 앱 컨테이너에 액세스할 권한이 있는 원격 컴퓨터 목록을 지정합니다. (Get)
get_RemotePorts 이 규칙에 대한 원격 포트 목록을 지정합니다. (Get)
get_RemoteUserAuthorizedList 앱 컨테이너에 액세스할 수 있는 권한이 있는 원격 사용자 목록을 지정합니다. (Get)
get_RuleCategories 타사 방화벽 제품 등록의 경우 타사 방화벽이 Windows 방화벽에서 소유권을 가져오려는 규칙 범주를 나타냅니다. (Get)
get_Rules 방화벽 규칙의 컬렉션을 검색합니다.
get_Rules Windows 서비스 강화 네트워킹 규칙의 컬렉션을 검색합니다.

get_Scope 포트가 수신 대기할 수 있는 네트워크 범위를 제어합니다. (INetFwAuthorizedApplication.get_Scope)
get_Scope 포트가 수신 대기할 수 있는 네트워크 범위를 제어합니다. (INetFwOpenPort.get_Scope)
get_Scope 원격 관리가 허용되는 네트워크 범위를 제어합니다. (Get)
get_Scope 포트가 수신 대기할 수 있는 네트워크 범위를 제어합니다. (INetFwService.get_Scope)
get_SecureFlags 컬렉션을 허용하도록 IPsec에서 제공하는 보안 수준의 방화벽 확인을 지정합니다. 허용되는 값은 NET_FW_AUTHENTICATE_TYPE 열거형의 값과 일치해야 합니다. (Get)
get_ServiceName 애플리케이션의 서비스 이름 속성을 지정합니다. (Get)
get_ServiceRestriction Windows 서비스 강화 저장소에 액세스하는 데 사용되는 인터페이스를 검색합니다.
get_Services 프로필의 서비스 컬렉션을 검색합니다.
get_SharingConnectionType get_SharingConnectionType 메서드는 이 연결에서 사용하도록 설정된 공유 유형을 결정합니다.
get_SharingEnabled get_SharingEnabled 메서드는 이 연결에서 공유를 사용할 수 있는지 여부를 결정합니다.
get_SharingInstalled 현재 설치된 Windows XP 버전이 연결 공유를 지원하는지 여부를 보고합니다.

get_StaticPortMappingCollection get_StaticPortMappingCollection 메서드는 로컬 컴퓨터에서 사용하는 NAT의 정적 포트 매핑 컬렉션에 대한 인터페이스를 검색합니다.
get_Status get_Status 메서드는 연결 상태를 검색합니다.
get_TargetIPAddress get_TargetIPAddress 메서드는 이 포트 매핑에 대한 대상 컴퓨터의 IP 주소를 검색합니다.
get_TargetName get_TargetName 메서드는 이 포트 매핑에 대한 대상 컴퓨터의 이름을 검색합니다.
get_Type 프로필의 형식을 지정합니다.
get_Type 서비스의 형식을 검색합니다.
get_UnicastResponsesToMulticastBroadcastDisabled 방화벽이 멀티캐스트 및 브로드캐스트 트래픽에 대한 유니캐스트 응답을 허용하지 않아야 하는지 여부를 나타냅니다. (INetFwPolicy2.get_UnicastResponsesToMulticastBroadcastDisabled)
get_UnicastResponsesToMulticastBroadcastDisabled 방화벽이 멀티캐스트 및 브로드캐스트 트래픽에 대한 유니캐스트 응답을 허용하지 않아야 하는지 여부를 나타냅니다. (INetFwProfile.get_UnicastResponsesToMulticastBroadcastDisabled)
GetProfileByType 요청된 형식의 프로필을 검색합니다.
GetProperties GetProperties 메서드는 이 네트워크 연결에 대한 속성을 포함하는 구조를 검색합니다.
GetUiObjectClassId GetUiObjectClassId 메서드는 이 연결에 대한 사용자 인터페이스 개체의 클래스 식별자를 검색합니다.

IsIcmpTypeAllowed 지정된 ICMP 형식이 허용되는지 여부를 확인합니다.
IsPortAllowed 애플리케이션이 지정된 포트에서 인바운드 트래픽을 수신 대기할 수 있는지 여부를 결정합니다.
IsRuleGroupEnabled 지정된 방화벽 규칙 그룹을 사용할지 여부를 결정합니다. (INetFwPolicy2.IsRuleGroupEnabled)
Item Item 메서드는 컬렉션에 있는 경우 지정된 애플리케이션을 반환합니다.
Item 컬렉션에 있는 경우 지정된 포트를 반환합니다.
Item Item 메서드는 컬렉션에 있는 경우 지정된 인덱스가 있는 제품을 반환합니다.
Item Item 메서드는 컬렉션에 있는 경우 지정된 규칙을 반환합니다.
Item 컬렉션에 있는 경우 지정된 서비스를 반환합니다.
NcFreeNetconProperties NcFreeNetconProperties 함수는 NETCON_PROPERTIES 구조체와 연결된 메모리를 해제합니다.
NcIsValidConnectionName NcIsValidConnectionName 함수는 전달된 연결 이름이 유효한지 확인합니다.
NetworkIsolationDiagnoseConnectFailureAndGetInfo NetworkIsolationDiagnoseConnectFailureAndGetInfo 함수는 기능 누락으로 인한 네트워크 격리 연결 실패에 대한 정보를 가져옵니다.
NetworkIsolationDiagnoseConnectFailureAndGetInfo NetworkIsolationDiagnoseConnectFailureAndGetInfo 함수는 기능 누락으로 인한 네트워크 격리 연결 실패에 대한 정보를 가져옵니다.

NetworkIsolationEnumAppContainers NetworkIsolationEnumAppContainers 함수는 시스템에서 만든 모든 앱 컨테이너를 열거합니다.
NetworkIsolationEnumAppContainers NetworkIsolationEnumAppContainers 함수는 시스템에서 만든 모든 앱 컨테이너를 열거합니다. (NetworkIsolationEnumAppContainers)
NetworkIsolationEnumerateAppContainerRules 앱 컨테이너와 관련된 모든 규칙을 열거합니다.
NetworkIsolationFreeAppContainers NetworkIsolationFreeAppContainers 함수는 하나 이상의 앱 컨테이너에 할당된 메모리 리소스를 해제하는 데 사용됩니다.
NetworkIsolationFreeAppContainers NetworkIsolationFreeAppContainers 함수는 하나 이상의 앱 컨테이너에 할당된 메모리 리소스를 해제하는 데 사용됩니다. (NetworkIsolationFreeAppContainers)
NetworkIsolationGetAppContainerConfig NetworkIsolationGetAppContainerConfig 함수는 하나 이상의 앱 컨테이너에 대한 구성 정보를 검색하는 데 사용됩니다.
NetworkIsolationGetAppContainerConfig NetworkIsolationGetAppContainerConfig 함수는 하나 이상의 앱 컨테이너에 대한 구성 정보를 검색하는 데 사용됩니다.
NetworkIsolationGetEnterpriseIdAsync WIP(Windows Information Protection) 또는 WDAG(Windows Defender Application Guard) 시나리오의 컨텍스트에서 네트워크 격리 엔드포인트를 기반으로 하는 엔터프라이즈 ID를 가져옵니다.
NetworkIsolationGetEnterpriseIdClose 이 API는 NetworkIsolationGetEnterpriseIdAsync에서 반환된 핸들을 닫고 작업을 동기화하는 데 사용됩니다.
NetworkIsolationRegisterForAppContainerChanges NetworkIsolationRegisterForAppContainerChanges 함수는 앱 컨테이너의 변경 내용에 대한 알림 배달을 등록하는 데 사용됩니다.

NetworkIsolationRegisterForAppContainerChanges NetworkIsolationRegisterForAppContainerChanges 함수는 앱 컨테이너의 변경 내용에 대한 알림 배달을 등록하는 데 사용됩니다.
NetworkIsolationSetAppContainerConfig NetworkIsolationSetAppContainerConfig 함수는 하나 이상의 앱 컨테이너 구성을 설정하는 데 사용됩니다.
NetworkIsolationSetAppContainerConfig NetworkIsolationSetAppContainerConfig 함수는 하나 이상의 앱 컨테이너 구성을 설정하는 데 사용됩니다. (NetworkIsolationSetAppContainerConfig)
NetworkIsolationSetupAppContainerBinaries NetworkIsolationSetupAppContainerBinaries 함수는 소프트웨어 설치 관리자가 앱 컨테이너에서 실행 중인 애플리케이션의 이미지 경로에 대한 정보를 제공하는 데 사용됩니다.
NetworkIsolationSetupAppContainerBinaries NetworkIsolationSetupAppContainerBinaries 함수는 소프트웨어 설치 관리자가 앱 컨테이너에서 실행 중인 애플리케이션의 이미지 경로에 대한 정보를 제공하는 데 사용됩니다.
NetworkIsolationUnregisterForAppContainerChanges NetworkIsolationUnregisterForAppContainerChanges 함수는 앱 컨테이너 변경 등록을 취소하고 알림 수신을 중지하는 데 사용됩니다.
NetworkIsolationUnregisterForAppContainerChanges NetworkIsolationUnregisterForAppContainerChanges 함수는 앱 컨테이너 변경 등록을 취소하고 알림 수신을 중지하는 데 사용됩니다.
NewExternalIPAddress NAT 컴퓨터의 외부 IP 주소가 변경되면 시스템에서 NewExternalIPAddress 메서드를 호출합니다.
NewNumberOfEntries NAT 포트 매핑의 총 수가 변경되면 시스템에서 NewNumberOfEntries 메서드를 호출합니다.
다음 Next 메서드는 현재 열거형 위치에서 시작하는 Connections 폴더에서 지정된 수의 연결을 검색합니다.

다음 Next 메서드는 현재 열거형 위치에서 시작하는 지정된 수의 포트 매핑을 검색합니다.
다음 Next 메서드는 현재 열거형 위치에서 시작하는 개인 공유 연결의 지정된 수를 검색합니다. (IEnumNetSharingPrivateConnection.Next)
다음 Next 메서드는 현재 열거형 위치에서 시작하는 개인 공유 연결의 지정된 수를 검색합니다. (IEnumNetSharingPublicConnection.Next)
PAC_CHANGES_CALLBACK_FN PAC_CHANGES_CALLBACK_FN 콜백 함수는 앱 컨테이너 변경 알림 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다.
PAC_CHANGES_CALLBACK_FN PAC_CHANGES_CALLBACK_FN 콜백 함수는 앱 컨테이너 변경 알림 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다.
PFN_FWADDDYNAMICKEYWORDADDRESS0 지정된 동적 키워드 주소를 추가하기 위해 호출하는 서비스에서 진입점의 함수 포인터 형식입니다.
PFN_FWDELETEDYNAMICKEYWORDADDRESS0 지정된 ID를 사용하여 동적 키워드 주소를 삭제하기 위해 호출하는 서비스의 진입점의 함수 포인터 형식입니다.
PFN_FWENUMDYNAMICKEYWORDADDRESSBYID0 ID별로 특정 동적 키워드 주소를 열거하기 위해 호출하는 서비스의 진입점의 함수 포인터 형식입니다.
PFN_FWENUMDYNAMICKEYWORDADDRESSESBYTYPE0 동적 키워드 주소를 형식별로 열거하기 위해 호출하는 서비스의 진입점의 함수 포인터 형식입니다. 전달된 열거형 플래그에 따라 개체의 특정 하위 집합을 요청할 수 있습니다.
PFN_FWFREEDYNAMICKEYWORDADDRESSDATA0 서비스에서 할당한 동적 키워드 주소 데이터 구조체를 해제하기 위해 호출하는 서비스의 진입점의 함수 포인터 형식입니다.

PFN_FWUPDATEDYNAMICKEYWORDADDRESS0 동적 키워드 주소를 입력 ID로 업데이트하기 위해 호출하는 서비스의 진입점 함수 포인터 형식입니다.
put_Action 규칙 또는 기본 설정에 대한 작업을 지정합니다. (INetFwRule.put_Action)
put_AllowInboundEchoRequest 허용되는지 여부를 나타냅니다. (INetFwIcmpSettings.put_AllowInboundEchoRequest)
put_AllowInboundMaskRequest 허용되는지 여부를 나타냅니다. (INetFwIcmpSettings.put_AllowInboundMaskRequest)
put_AllowInboundRouterRequest 허용되는지 여부를 나타냅니다. (INetFwIcmpSettings.put_AllowInboundRouterRequest)
put_AllowInboundTimestampRequest 허용되는지 여부를 나타냅니다. (INetFwIcmpSettings.put_AllowInboundTimestampRequest)
put_AllowOutboundDestinationUnreachable 허용되는지 여부를 나타냅니다. (INetFwIcmpSettings.put_AllowOutboundDestinationUnreachable)
put_AllowOutboundPacketTooBig 허용되는지 여부를 나타냅니다. (INetFwIcmpSettings.put_AllowOutboundPacketTooBig)
put_AllowOutboundParameterProblem 허용되는지 여부를 나타냅니다. (INetFwIcmpSettings.put_AllowOutboundParameterProblem)
put_AllowOutboundSourceQuench 아웃바운드 원본 쿼리가 허용되는지 여부를 나타냅니다. (Put)
put_AllowOutboundTimeExceeded 아웃바운드 시간을 초과할 수 있는지 여부를 나타냅니다. (Put)
put_AllowRedirect 리디렉션이 허용되는지 여부를 나타냅니다. (Put)

put_ApplicationName 이 규칙이 적용되는 애플리케이션의 이름을 지정합니다. (Put)
put_BlockAllInboundTraffic 방화벽에서 인바운드 트래픽을 허용하지 않아야 하는지 여부를 나타냅니다. (Put)
put_DefaultInboundAction 인바운드 트래픽에 대한 기본 작업을 지정합니다. 이러한 설정은 기본적으로 차단입니다. (Put)
put_DefaultOutboundAction 아웃바운드 트래픽에 대한 기본 작업을 지정합니다. 이러한 설정은 기본적으로 허용입니다. (Put)
put_Description 이 규칙에 대한 설명을 지정합니다. (Put)
put_Direction 규칙이 적용되는 트래픽의 방향을 지정합니다. (Put)
put_DisplayName 타사 방화벽 제품 등록의 표시 이름을 나타냅니다. (Put)
put_EdgeTraversal 이 규칙에 대해 에지 순회를 사용할지 여부를 나타냅니다. (Put)
put_EdgeTraversalOptions 이 속성은 NET_FW_EDGE_TRAVERSAL_TYPE 정의된 방화벽 규칙의 에지 속성에 액세스하는 데 사용할 수 있습니다. (Put)
put_Enabled 이 애플리케이션에 대한 설정이 현재 사용하도록 설정되어 있는지 여부를 나타냅니다. (Put)
put_Enabled 이 포트에 대한 설정이 현재 사용하도록 설정되어 있는지 여부를 나타냅니다. (Put)
put_Enabled 원격 관리를 사용할 수 있는지 여부를 나타냅니다. (Put)

put_Enabled 규칙을 사용하거나 사용하지 않도록 설정합니다. (Put)
put_Enabled 서비스와 연결된 모든 포트를 사용할 수 있는지 여부를 나타냅니다. (Put)
put_ExceptionsNotAllowed 방화벽에서 예외를 허용하지 않아야 하는지 여부를 나타냅니다. (Put)
put_ExcludedInterfaces 방화벽 설정이 제외되는 인터페이스 목록을 지정합니다. (Put)
put_ExternalIPAddressCallback put_ExternalIPAddressCallback 메서드를 사용하면 UPnP 기술을 사용하는 NAT 애플리케이션이 NAT에 콜백 인터페이스를 등록할 수 있습니다. NAT의 외부 IP 주소가 변경되면 시스템에서 이 콜백 인터페이스의 첫 번째 메서드를 호출합니다.
put_FirewallEnabled 방화벽이 로컬로 사용하도록 설정되어 있는지 여부를 나타냅니다(그룹 정책 설정으로 인해 유효한 결과가 다를 수 있음). (Put)
put_FirewallEnabled 방화벽을 사용할 수 있는지 여부를 나타냅니다. (Put)
put_Grouping 개별 규칙이 속한 그룹을 지정합니다. (Put)
put_IcmpTypesAndCodes 이 규칙의 ICMP 형식 및 코드 목록을 지정합니다. (Put)
put_Interfaces 규칙이 적용되는 인터페이스 목록을 지정합니다. (Put)
put_InterfaceTypes 규칙이 적용되는 인터페이스 형식의 목록을 지정합니다. (Put)

put_IpVersion 이 애플리케이션의 IP 버전 설정을 지정합니다. (Put)
put_IpVersion 이 포트에 대한 IP 버전 설정을 지정합니다. (Put)
put_IpVersion IP 버전을 지정합니다. (INetFwRemoteAdminSettings.put_IpVersion)
put_IpVersion 서비스에 권한이 부여된 방화벽 IP 버전을 지정합니다. (Put)
put_LocalAddresses 이 규칙의 로컬 주소 목록을 지정합니다. (Put)
put_LocalAppPackageId Windows 스토어 앱이든 데스크톱 앱이든 관계없이 프로세스의 패키지 식별자 또는 앱 컨테이너 식별자를 지정합니다. (Put)
put_LocalPorts 이 규칙의 로컬 포트 목록을 지정합니다. (Put)
put_LocalUserAuthorizedList 앱 컨테이너에 대한 권한 있는 로컬 사용자 목록을 지정합니다. (Put)
put_LocalUserOwner 규칙의 소유자인 사용자의 SID(사용자 보안 식별자)를 지정합니다. (Put)
put_Name 이 애플리케이션의 이름을 지정합니다. (Put)
put_Name 이 포트의 이름을 지정합니다. (Put)
put_Name 이 규칙의 이름을 지정합니다. (Put)

put_NotificationsDisabled 대화형 방화벽 알림을 사용하지 않도록 설정할지 여부를 나타냅니다. (INetFwPolicy2.put_NotificationsDisabled)
put_NotificationsDisabled 대화형 방화벽 알림을 사용하지 않도록 설정할지 여부를 나타냅니다. (INetFwProfile.put_NotificationsDisabled)
put_NumberOfEntriesCallback put_NumberOfEntriesCallback 메서드를 사용하면 UPnP 기술을 사용하는 NAT 애플리케이션이 NAT에 콜백 인터페이스를 등록할 수 있습니다. NAT 포트 매핑 수가 변경되면 시스템에서 이 콜백 인터페이스의 첫 번째 메서드를 호출합니다.
put_Port 이 포트의 호스트 순서 포트 번호를 지정합니다. (Put)
put_ProcessImageFileName 이 애플리케이션의 프로세스 이미지 파일 이름을 지정합니다. (Put)
put_Profiles 규칙이 속한 프로필을 지정합니다. (Put)
put_Protocol 이 포트에 대한 프로토콜 유형 설정을 지정합니다. (Put)
put_Protocol 이 규칙의 IP 프로토콜을 지정합니다. (Put)
put_RemoteAddresses 애플리케이션이 트래픽을 수신 대기할 수 있는 원격 주소 집합을 지정합니다. (Put)
put_RemoteAddresses 포트가 트래픽을 수신 대기할 수 있는 원격 주소 집합을 지정합니다. (Put)
put_RemoteAddresses 원격 관리가 허용되는 원격 주소 집합을 지정합니다. (Put)

put_RemoteAddresses 이 규칙의 원격 주소 목록을 지정합니다. (Put)
put_RemoteAddresses 서비스 포트가 트래픽을 수신 대기할 수 있는 원격 주소 집합을 지정합니다. (Put)
put_RemoteMachineAuthorizedList 앱 컨테이너에 액세스할 수 있는 권한이 있는 원격 컴퓨터 목록을 지정합니다. (Put)
put_RemotePorts 이 규칙의 원격 포트 목록을 지정합니다. (Put)
put_RemoteUserAuthorizedList 앱 컨테이너에 액세스할 수 있는 권한이 있는 원격 사용자 목록을 지정합니다. (Put)
put_RuleCategories 타사 방화벽 제품 등록의 경우 타사 방화벽이 Windows 방화벽에서 소유권을 가져오려는 규칙 범주를 나타냅니다. (Put)
put_Scope 포트가 수신 대기할 수 있는 네트워크 범위를 제어합니다. (INetFwAuthorizedApplication.put_Scope)
put_Scope 포트가 수신 대기할 수 있는 네트워크 범위를 제어합니다. (INetFwOpenPort.put_Scope)
put_Scope 원격 관리가 허용되는 네트워크 범위를 제어합니다. (Put)
put_Scope 포트가 수신 대기할 수 있는 네트워크 범위를 제어합니다. (INetFwService.put_Scope)
put_SecureFlags 컬렉션을 허용하도록 IPsec에서 제공하는 보안 수준의 방화벽 확인을 지정합니다. 허용되는 값은 NET_FW_AUTHENTICATE_TYPE 열거형의 값과 일치해야 합니다. (Put)

put_ServiceName 애플리케이션의 서비스 이름 속성을 지정합니다. (Put)
put_UnicastResponsesToMulticastBroadcastDisabled 방화벽이 멀티캐스트 및 브로드캐스트 트래픽에 대한 유니캐스트 응답을 허용하지 않아야 하는지 여부를 나타냅니다. (INetFwPolicy2.put_UnicastResponsesToMulticastBroadcastDisabled)
put_UnicastResponsesToMulticastBroadcastDisabled 방화벽이 멀티캐스트 및 브로드캐스트 트래픽에 대한 유니캐스트 응답을 허용하지 않아야 하는지 여부를 나타냅니다. (INetFwProfile.put_UnicastResponsesToMulticastBroadcastDisabled)
등록 Register 메서드는 타사 방화벽 제품을 등록합니다.
제거 Remove 메서드는 컬렉션에서 지정된 포트 매핑을 제거합니다.
제거 Remove 메서드는 컬렉션에서 애플리케이션을 제거합니다.
제거 포트를 닫고 컬렉션에서 제거합니다.
제거 Remove 메서드는 컬렉션에서 규칙을 제거합니다.
RemovePortMapping RemovePortMapping 메서드는 이 연결에 대한 매핑 목록에서 서비스 포트 매핑을 제거합니다.
이름 바꾸기 Rename 메서드는 이 연결의 이름을 바꿉니다.
재설정 Reset 메서드를 사용하면 열거형의 시작 부분에서 후속 열거형 호출이 작동합니다. (IEnumNetSharingEveryConnection.Reset)

재설정 Reset 메서드를 사용하면 열거형의 시작 부분에서 후속 열거형 호출이 작동합니다. (IEnumNetSharingPortMapping.Reset)
재설정 Reset 메서드를 사용하면 열거형의 시작 부분에서 후속 열거형 호출이 작동합니다. (IEnumNetSharingPrivateConnection.Reset)
재설정 Reset 메서드를 사용하면 열거형의 시작 부분에서 후속 열거형 호출이 작동합니다. (IEnumNetSharingPublicConnection.Reset)
RestoreDefaults 로컬 구성을 기본 설치된 상태로 복원합니다.
RestoreLocalFirewallDefaults 로컬 방화벽 구성을 기본 상태로 복원합니다.
RestrictService RestrictService 메서드는 지정된 서비스에 대한 서비스 제한을 설정하거나 해제합니다.
ServiceRestricted ServiceRestricted 메서드는 방화벽 규칙에 지정된 리소스로 트래픽을 제한하기 위해 서비스 제한 규칙을 사용할 수 있는지 여부를 나타냅니다.
skip Skip 메서드는 이 열거형에 대해 지정된 개인 공유 연결 수를 건너뜁니다. (IEnumNetSharingEveryConnection.Skip)
skip Skip 메서드는 이 열거형에 대해 지정된 수의 포트 매핑을 건너뜁니다.
skip Skip 메서드는 이 열거형에 대해 지정된 개인 공유 연결 수를 건너뜁니다. (IEnumNetSharingPrivateConnection.Skip)
skip Skip 메서드는 이 열거형에 대해 지정된 수의 공개적으로 공유된 연결을 건너뜁니다.

인터페이스

IEnumNetSharingEveryConnection IEnumNetSharingEveryConnection 인터페이스는 Connections 폴더의 모든 연결을 열거하는 메서드를 제공합니다.
IEnumNetSharingPortMapping IEnumNetSharingPortMapping 인터페이스는 특정 연결에 대한 포트 매핑을 열거하는 메서드를 제공합니다.
IEnumNetSharingPrivateConnection IEnumNetSharingPrivateConnection 인터페이스는 현재 구성된 비공개 공유 연결을 열거하는 메서드를 제공합니다.
IEnumNetSharingPublicConnection IEnumNetSharingPublicConnection 인터페이스는 현재 구성된 공개적으로 공유된 연결을 열거하는 메서드를 제공합니다.
INATEventManager INATEventManager 인터페이스는 UPnP 기술을 사용하여 NAT 애플리케이션에 대한 메서드를 제공하여 NAT에 콜백 인터페이스를 등록합니다. NAT 구성이 변경되면 시스템은 이러한 인터페이스의 메서드를 호출합니다.
INATExternalIPAddressCallback INATExternalIPAddressCallback 인터페이스는 UPnP 기술을 사용하여 NAT 애플리케이션에 의해 구현됩니다. NAT 컴퓨터의 외부 IP 주소가 변경되면 시스템에서 호출하는 메서드를 제공합니다.
INATNumberOfEntriesCallback INATNumberOfEntriesCallback 인터페이스는 포트 매핑 수가 변경되면 시스템에서 호출하는 메서드를 제공합니다.
INetConnection INetConnection 인터페이스는 네트워크 연결을 관리하는 메서드를 제공합니다.
INetConnectionProps INetConnectionProps 인터페이스를 사용하여 연결에 대한 속성을 검색합니다.

INetFwAuthorizedApplication INetFwAuthorizedApplication 인터페이스는 권한이 부여된 애플리케이션의 속성에 대한 액세스를 방화벽에 여는 기능을 제공합니다.
INetFwAuthorizedApplications INetFwAuthorizedApplications 인터페이스는 방화벽에서 권한 있는 열린 포트의 애플리케이션 컬렉션에 대한 액세스를 제공합니다.
INetFwIcmpSettings INetFwIcmpSettings 인터페이스는 ICMP 패킷을 제어하는 설정에 대한 액세스를 제공합니다.
INetFwMgr INetFwMgr 인터페이스는 컴퓨터의 방화벽 설정에 대한 액세스를 제공합니다.
INetFwOpenPort INetFwOpenPort 인터페이스는 방화벽에서 열린 포트의 속성에 대한 액세스를 제공합니다.
INetFwOpenPorts INetFwOpenPorts 인터페이스는 표준 Automation 컬렉션 인터페이스입니다.
INetFwPolicy INetFwPolicy 인터페이스는 방화벽 정책에 대한 액세스를 제공합니다.
INetFwPolicy2 방화벽 정책에 액세스하려면
INetFwProduct 타사 방화벽 등록의 속성에 액세스하려면
INetFwProducts Windows 방화벽을 사용하여 타사 방화벽 제품을 등록하고 등록된 제품을 열거하기 위한 메서드 및 속성에 액세스하려면
INetFwProfile INetFwProfile 인터페이스는 방화벽 설정 프로필에 대한 액세스를 제공합니다.

INetFwRemoteAdminSettings INetFwRemoteAdminSettings 인터페이스는 원격 관리를 제어하는 설정에 대한 액세스를 제공합니다.
INetFwRule 규칙의 속성에.
INetFwRule2 애플리케이션 또는 서비스가 INetFwRule의 모든 속성과 NET_FW_EDGE_TRAVERSAL_TYPE 지정된 방화벽 규칙의 네 가지 에지 속성에 액세스할 수 있도록 허용합니다.
INetFwRule3 애플리케이션 또는 서비스가 INetFwRule2의 모든 속성에 액세스하고 앱 컨테이너의 요구 사항에 액세스할 수 있도록 허용합니다.
INetFwRules 방화벽 규칙의 컬렉션입니다.
INetFwService INetFwService 인터페이스는 방화벽을 통해 수신 대기할 수 있는 권한이 부여될 수 있는 서비스의 속성에 대한 액세스를 제공합니다.
INetFwServiceRestriction Windows 서비스 강화 네트워킹 규칙에 액세스합니다.
INetFwServices INetFwServices 인터페이스는 방화벽을 통해 수신할 수 있는 권한이 부여될 수 있는 서비스 컬렉션에 대한 액세스를 제공하는 표준 Automation 인터페이스입니다.
INetSharingConfiguration INetSharingConfiguration 인터페이스는 연결 공유, 포트 매핑 및 인터넷 연결 방화벽을 관리하는 메서드를 제공합니다.
INetSharingEveryConnectionCollection INetSharingEveryConnectionCollection 인터페이스를 사용하면 VBScript 및 JScript와 같은 언어를 스크립팅하여 연결 폴더의 모든 연결을 열거할 수 있습니다.

INetSharingManager INetSharingManager 인터페이스는 Manager 개체의 기본 인터페이스입니다. INetSharingManager는 공유가 설치되어 있는지 확인하고, 포트 매핑을 관리하고, 퍼블릭 및 프라이빗 연결에 대한 열거형 인터페이스를 가져오는 메서드를 제공합니다.
INetSharingPortMapping INetSharingPortMapping 인터페이스는 특정 포트 매핑을 관리하기 위한 메서드를 제공합니다.
INetSharingPortMappingCollection INetSharingPortMappingCollection 인터페이스를 사용하면 VBScript 및 JScript와 같은 언어를 스크립팅하여 포트 매핑을 열거할 수 있습니다.
INetSharingPortMappingProps INetSharingPortMappingProps 인터페이스는 네트워크 연결 포트 매핑의 속성을 검색하고 설정하는 메서드를 제공합니다.
INetSharingPrivateConnectionCollection INetSharingPrivateConnectionCollection 인터페이스를 사용하면 VBScript 및 JScript와 같은 언어를 스크립팅하여 프라이빗 연결을 열거할 수 있습니다.
INetSharingPublicConnectionCollection INetSharingPublicConnectionCollection 인터페이스를 사용하면 VBScript 및 JScript와 같은 언어를 스크립팅하여 공용 연결을 열거할 수 있습니다.
IStaticPortMapping IStaticPortMapping 인터페이스는 특정 포트 매핑에 대한 정보를 검색하고 변경하는 메서드를 제공합니다.
IStaticPortMappingCollection IStaticPortMappingCollection 인터페이스는 정적 포트 매핑의 컬렉션을 관리하는 메서드를 제공합니다.
IUPnPNAT IUPnPNAT 인터페이스는 UPnP를 사용하여 NAT(네트워크 주소 변환)를 관리하기 위한 기본 인터페이스입니다. IUPnPNAT 인터페이스는 UPnP 기술을 사용하여 NAT API의 다른 모든 인터페이스에 직접 또는 간접적으로 액세스를 제공합니다.

구조체

FW_DYNAMIC_KEYWORD_ADDRESS_DATA0 **열거형** API가 호출되면 클라이언트에 반환된 데이터를 보유합니다.
FW_DYNAMIC_KEYWORD_ADDRESS0 클라이언트가 IP 주소 목록을 포함하는 동적 키워드 주소를 만들 수 있습니다.
INET_FIREWALL_AC_BINARIES INET_FIREWALL_AC_BINARIES 구조에는 앱 컨테이너에서 실행되는 애플리케이션에 대한 이진 경로가 포함됩니다.
INET_FIREWALL_AC_BINARIES INET_FIREWALL_AC_BINARIES 구조에는 앱 컨테이너에서 실행되는 애플리케이션에 대한 이진 경로가 포함됩니다. (INET_FIREWALL_AC_BINARIES)
INET_FIREWALL_AC_CAPABILITIES INET_FIREWALL_AC_CAPABILITIES 구조에는 앱 컨테이너의 기능에 대한 정보가 포함되어 있습니다.
INET_FIREWALL_AC_CAPABILITIES INET_FIREWALL_AC_CAPABILITIES 구조에는 앱 컨테이너의 기능에 대한 정보가 포함되어 있습니다. (INET_FIREWALL_AC_CAPABILITIES)
INET_FIREWALL_AC_CHANGE INET_FIREWALL_AC_CHANGE 구조에는 앱 컨테이너의 변경 내용에 대한 정보가 포함되어 있습니다. (INET_FIREWALL_AC_CHANGE)
INET_FIREWALL_AC_CHANGE INET_FIREWALL_AC_CHANGE 구조에는 앱 컨테이너의 변경 내용에 대한 정보가 포함되어 있습니다. (INET_FIREWALL_AC_CHANGE)
INET_FIREWALL_APP_CONTAINER INET_FIREWALL_APP_CONTAINER 구조에는 특정 앱 컨테이너에 대한 정보가 포함됩니다. (INET_FIREWALL_APP_CONTAINER)
INET_FIREWALL_APP_CONTAINER INET_FIREWALL_APP_CONTAINER 구조에는 특정 앱 컨테이너에 대한 정보가 포함됩니다. (INET_FIREWALL_APP_CONTAINER)

NETCON_PROPERTIES NETCON_PROPERTIES 구조체는 네트워크 연결의 속성을 설명하는 값을 저장합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Windows GDI

아티클2023. 03. 16.

Windows GDI 기술 개요입니다.

Windows GDI를 개발하려면 다음 헤더가 필요합니다.

- fontsub.h
- mmsystem.h
- prnasnot.h
- prntvpt.h
- t2embapi.h
- tvout.h
- windef.h
- xpsprint.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows GDI
- 문서 및 인쇄

Functions

AbortPath AbortPath 함수는 지정된 디바이스 컨텍스트의 모든 경로를 닫고 삭제합니다.
AddFontMemResourceEx AddFontMemResourceEx 함수는 메모리 이미지의 글꼴 리소스를 시스템에 추가합니다.
AddFontResourceA AddFontResource 함수는 지정된 파일의 글꼴 리소스를 시스템 글꼴 테이블에 추가합니다. 이후에는 모든 애플리케이션에서 텍스트 출력에 글꼴을 사용할 수 있습니다. (ANSI)
AddFontResourceExA AddFontResourceEx 함수는 지정된 파일의 글꼴 리소스를 시스템에 추가합니다. AddFontResourceEx 함수를 사용하여 추가된 글꼴은 프라이빗으로 표시될 수 있으며 열거할 수 없습니다. (ANSI)

AddFontResourceExW AddFontResourceEx 함수는 지정된 파일의 글꼴 리소스를 시스템에 추가합니다. AddFontResourceEx 함수를 사용하여 추가된 글꼴은 프라이빗으로 표시될 수 있으며 열거할 수 없습니다. (유니코드)
AddFontResourceW AddFontResource 함수는 지정된 파일의 글꼴 리소스를 시스템 글꼴 테이블에 추가합니다. 이후에는 모든 애플리케이션에서 텍스트 출력에 글꼴을 사용할 수 있습니다. (유니코드)
AlphaBlend AlphaBlend 함수는 투명 또는 반투명 픽셀이 있는 비트맵을 표시합니다.
AngleArc AngleArc 함수는 선 세그먼트와 호를 그립니다.
AnimatePalette AnimatePalette 함수는 지정된 논리 팔레트의 항목을 대체합니다.
Arc Arc 함수는 타원형 호를 그립니다.
ArcTo ArcTo 함수는 타원형 호를 그립니다.
BeginPaint BeginPaint 함수는 그리기 위해 지정된 창을 준비하고 그림에 대한 정보로 PAINTSTRUCT 구조를 채웁니다.
BeginPath BeginPath 함수는 지정된 디바이스 컨텍스트에서 경로 대괄호를 엽니다.
Bitblt BitBlt 함수는 지정된 원본 디바이스 컨텍스트에서 대상 디바이스 컨텍스트로 픽셀 사각형에 해당하는 색 데이터의 비트 블록 전송을 수행합니다.
CancelDC CancelDC 함수는 지정된 DC(디바이스 컨텍스트)에서 보류 중인 작업을 취소합니다.

CFP_ALLOCPROC CreateFontPackage 및 MergeFontPackage에서 메모리를 할당하는 데 사용하는 클라이언트 제공 콜백 함수입니다.
CFP_FREEPROC CreateFontPackage 및 MergeFontPackage에서 메모리를 해제하는 데 사용되는 클라이언트 제공 콜백 함수입니다.
CFP_REALLOCPROC 할당된 버퍼의 크기를 변경해야 하는 경우 CreateFontPackage 및 MergeFontPackage에서 메모리를 다시 할당하는 데 사용되는 클라이언트 제공 콜백 함수입니다.
ChangeDisplaySettingsA ChangeDisplaySettings 함수는 기본 디스플레이 디바이스의 설정을 지정된 그래픽 모드로 변경합니다. (ANSI)
ChangeDisplaySettingsExA ChangeDisplaySettingsEx 함수는 지정된 디스플레이 디바이스의 설정을 지정된 그래픽 모드로 변경합니다. (ANSI)
ChangeDisplaySettingsExW ChangeDisplaySettingsEx 함수는 지정된 디스플레이 디바이스의 설정을 지정된 그래픽 모드로 변경합니다. (유니코드)
ChangeDisplaySettingsW ChangeDisplaySettings 함수는 기본 디스플레이 디바이스의 설정을 지정된 그래픽 모드로 변경합니다. (유니코드)
Chord(현) 코드 함수는 코드(타원의 교차점과 secant라고 하는 선 세그먼트로 경계가 지정된 영역)를 그립니다. 코드는 현재 펜을 사용하여 윤곽선으로 표시되고 현재 브러시를 사용하여 채워집니다.
ClientToScreen ClientToScreen 함수는 지정된 지점의 클라이언트 영역 좌표를 화면 좌표로 변환합니다.
CloseEnhMetaFile CloseEnhMetaFile 함수는 향상된 메타파일 디바이스 컨텍스트를 닫고 향상된 형식 메타파일을 식별하는 핸들을 반환합니다.

CloseFigure CloseFigure 함수는 경로의 열린 그림을 닫습니다.
CloseMetaFile CloseMetaFile 함수는 메타파일 디바이스 컨텍스트를 닫고 Windows 형식 메타파일을 식별하는 핸들을 반환합니다.
CombineRgn CombineRgn 함수는 두 영역을 결합하고 결과를 세 번째 지역에 저장합니다. 두 영역은 지정된 모드에 따라 결합됩니다.
CombineTransform CombineTransform 함수는 두 개의 세계 공간을 페이지 공간 변환과 연결합니다.
CopyEnhMetaFileA CopyEnhMetaFile 함수는 향상된 형식 메타파일의 내용을 지정된 파일에 복사합니다. (ANSI)
CopyEnhMetaFileW CopyEnhMetaFile 함수는 향상된 형식 메타파일의 내용을 지정된 파일에 복사합니다. (유니코드)
CopyMetaFileA CopyMetaFile 함수는 Windows 형식 메타파일의 콘텐츠를 지정된 파일에 복사합니다. (ANSI)
CopyMetaFileW CopyMetaFile 함수는 Windows 형식 메타파일의 콘텐츠를 지정된 파일에 복사합니다. (유니코드)
CopyRect CopyRect 함수는 한 사각형의 좌표를 다른 사각형에 복사합니다.
CreateBitmap CreateBitmap 함수는 지정된 너비, 높이 및 색 형식(색 평면 및 픽셀당 비트)을 사용하여 비트맵을 만듭니다.
CreateBitmapIndirect CreateBitmapIndirect 함수는 지정된 너비, 높이 및 색 형식(색 평면 및 픽셀당 비트)을 사용하여 비트맵을 만듭니다.

CreateBrushIndirect CreateBrushIndirect 함수는 지정된 스타일, 색 및 패턴을 가진 논리 브러시를 만듭니다.
CreateCompatibleBitmap CreateCompatibleBitmap 함수는 지정된 디바이스 컨텍스트와 연결된 디바이스와 호환되는 비트맵을 만듭니다.
CreateCompatibleDC CreateCompatibleDC 함수는 지정된 디바이스와 호환되는 DC(메모리 디바이스 컨텍스트)를 만듭니다.
CreateDCA CreateDC 함수는 지정된 이름을 사용하여 디바이스에 대한 DC(디바이스 컨텍스트)를 만듭니다. (ANSI)
CreateDCW CreateDC 함수는 지정된 이름을 사용하여 디바이스에 대한 DC(디바이스 컨텍스트)를 만듭니다. (유니코드)
CreateDIBitmap CreateDIBitmap 함수는 DIB에서 호환되는 DDB(비트맵)를 만들고 필요에 따라 비트맵 비트를 설정합니다.
CreateDIBPatternBrush CreateDIBPatternBrush 함수는 지정된 DIB(디바이스 독립적 비트맵)로 지정된 패턴이 있는 논리 브러시를 만듭니다.
CreateDIBPatternBrushPt CreateDIBPatternBrushPt 함수는 DIB(디바이스 독립적 비트맵)로 지정된 패턴이 있는 논리 브러시를 만듭니다.
CreateDIBSection CreateDIBSection 함수는 애플리케이션이 직접 쓸 수 있는 DIB를 만듭니다.
CreateDiscardableBitmap CreateDiscardableBitmap 함수는 지정된 디바이스와 호환되는 삭제 가능한 비트맵을 만듭니다.

CreateEllipticRgn CreateEllipticRgn 함수는 타원형 영역을 만듭니다.
CreateEllipticRgnIndirect CreateEllipticRgnIndirect 함수는 타원형 영역을 만듭니다.
CreateEnhMetaFileA CreateEnhMetaFile 함수는 향상된 형식 메타파일을 위한 디바이스 컨텍스트를 만듭니다. 이 디바이스 컨텍스트를 사용하여 디바이스 독립적 그림을 저장할 수 있습니다. (ANSI)
CreateEnhMetaFileW CreateEnhMetaFile 함수는 향상된 형식 메타파일을 위한 디바이스 컨텍스트를 만듭니다. 이 디바이스 컨텍스트를 사용하여 디바이스 독립적 그림을 저장할 수 있습니다. (유니코드)
CreateFontA CreateFont 함수는 지정된 특성을 가진 논리 글꼴을 만듭니다. 이후 논리 글꼴을 모든 디바이스의 글꼴로 선택할 수 있습니다. (ANSI)
CreateFontIndirectA CreateFontIndirect 함수는 지정된 특성을 가진 논리 글꼴을 만듭니다. 이후 글꼴을 모든 디바이스 컨텍스트의 현재 글꼴로 선택할 수 있습니다. (ANSI)
CreateFontIndirectExA CreateFontIndirectEx 함수는 지정된 구조체의 특징이 있는 논리 글꼴을 지정합니다. 이후 글꼴을 모든 디바이스 컨텍스트의 현재 글꼴로 선택할 수 있습니다. (ANSI)
CreateFontIndirectExW CreateFontIndirectEx 함수는 지정된 구조체의 특징이 있는 논리 글꼴을 지정합니다. 이후 글꼴을 모든 디바이스 컨텍스트의 현재 글꼴로 선택할 수 있습니다. (유니코드)
CreateFontIndirectW CreateFontIndirect 함수는 지정된 특성을 가진 논리 글꼴을 만듭니다. 이후 글꼴을 모든 디바이스 컨텍스트의 현재 글꼴로 선택할 수 있습니다. (유니코드)
CreateFontPackage CreateFontPackage 함수는 일반적으로 프린터에 전달하기 위해 지정된 TrueType 글꼴의 하위 집합 버전을 만듭니다.

CreateFontW CreateFont 함수는 지정된 특성을 가진 논리 글꼴을 만듭니다. 이후 논리 글꼴을 모든 디바이스의 글꼴로 선택할 수 있습니다. (유니코드)
CreateHalftonePalette CreateHalftonePalette 함수는 지정된 DC(디바이스 컨텍스트)에 대한 하프톤 팔레트를 만듭니다.
CreateHatchBrush CreateHatchBrush 함수는 지정된 해치 패턴과 색이 있는 논리 브러시를 만듭니다.
CreateICA CreateIC 함수는 지정된 디바이스에 대한 정보 컨텍스트를 만듭니다. (ANSI)
CreateICW CreateIC 함수는 지정된 디바이스에 대한 정보 컨텍스트를 만듭니다. (유니코드)
CreateMetaFileA CreateMetaFile 함수는 Windows 형식 메타파일의 디바이스 컨텍스트를 만듭니다. (ANSI)
CreateMetaFileW CreateMetaFile 함수는 Windows 형식 메타파일의 디바이스 컨텍스트를 만듭니다. (유니코드)
CreatePalette CreatePalette 함수는 논리 팔레트를 만듭니다.
CreatePatternBrush CreatePatternBrush 함수는 지정된 비트맵 패턴을 사용하여 논리 브러시를 만듭니다. 비트맵은 CreateDIBSection 함수에서 만든 DIB 섹션 비트맵이거나 디바이스 종속 비트맵일 수 있습니다.
CreatePen CreatePen 함수는 지정된 스타일, 너비 및 색이 있는 논리 펜을 만듭니다. 이후에 펜을 디바이스 컨텍스트로 선택하고 선과 곡선을 그리는 데 사용할 수 있습니다.
CreatePenIndirect CreatePenIndirect 함수는 구조체에 지정된 스타일, 너비 및 색이 있는 논리적 코스메틱 펜을 만듭니다.

CreatePolygonRgn CreatePolygonRgn 함수는 다각형 영역을 만듭니다.
CreatePolyPolygonRgn CreatePolyPolygonRgn 함수는 일련의 다각형으로 구성된 영역을 만듭니다. 다각형은 겹칠 수 있습니다.
CreateRectRgn CreateRectRgn 함수는 사각형 영역을 만듭니다.
CreateRectRgnIndirect CreateRectRgnIndirect 함수는 사각형 영역을 만듭니다.
CreateRoundRectRgn CreateRoundRectRgn 함수는 둥근 모서리가 있는 사각형 영역을 만듭니다.
CreateScalableFontResourceA CreateScalableFontResource 함수는 확장 가능한 글꼴에 대한 글꼴 리소스 파일을 만듭니다. (ANSI)
CreateScalableFontResourceW CreateScalableFontResource 함수는 확장 가능한 글꼴에 대한 글꼴 리소스 파일을 만듭니다. (유니코드)
CreateSolidBrush CreateSolidBrush 함수는 지정된 단색을 가진 논리 브러시를 만듭니다.
DeleteDC DeleteDC 함수는 지정된 DC(디바이스 컨텍스트)를 삭제합니다.
DeleteEnhMetaFile DeleteEnhMetaFile 함수는 고급 형식 메타파일 또는 고급 형식 메타파일 핸들을 삭제합니다.
DeleteFont DeleteFont 매크로는 글꼴 개체를 삭제하여 글꼴 개체와 연결된 모든 시스템 리소스를 해제합니다.
DeleteMetaFile DeleteMetaFile 함수는 Windows 형식 메타파일 또는 Windows 형식 메타파일 핸들을 삭제합니다.

DeleteObject DeleteObject 함수는 논리 펜, 브러시, 글꼴, 비트맵, 지역 또는 팔레트를 삭제하여 개체와 연결된 모든 시스템 리소스를 해제합니다. 개체가 삭제되면 지정된 핸들이 더 이상 유효하지 않습니다.
디빈덱스 DIBINDEX 매크로는 인덱스를 DIB 색 테이블의 항목으로 가져오고 지정된 인덱스와 연결된 색을 지정하는 COLORREF 값을 반환합니다.
DPtoLP DPtoLP 함수는 디바이스 좌표를 논리 좌표로 변환합니다. 변환은 디바이스 컨텍스트의 매핑 모드, 창 및 뷰포트의 원본 및 익스텐트 설정 및 월드 변환에 따라 달라집니다.
DrawAnimatedRects 창의 캡션에 애니메이션을 적용하여 아이콘의 열기 또는 창 최소화 또는 최대화를 나타냅니다.
DrawCaption DrawCaption 함수는 창 캡션을 그립니다.
DrawEdge DrawEdge 함수는 하나 이상의 사각형 가장자리를 그립니다.
그리스케이프 DrawEscape 함수는 GDI(그래픽 디바이스 인터페이스)를 통해 직접 사용할 수 없는 지정된 비디오 디스플레이의 그리기 기능을 제공합니다.
DrawFocusRect DrawFocusRect 함수는 사각형에 포커스가 있음을 나타내는 데 사용되는 스타일로 사각형을 그립니다.
DrawFrameControl DrawFrameControl 함수는 지정된 형식과 스타일의 프레임 컨트롤을 그립니다.
DrawStateA DrawState 함수는 이미지를 표시하고 비활성 상태 또는 기본 상태와 같은 상태를 나타내기 위해 시각적 효과를 적용합니다. (ANSI)

DRAWSTATEPROC DrawStateProc 함수는 DrawState 함수에 대한 복잡한 이미지를 렌더링하는 애플리케이션 정의 콜백 함수입니다.
DrawStateW DrawState 함수는 이미지를 표시하고 비활성 상태 또는 기본 상태와 같은 상태를 나타내기 위해 시각적 효과를 적용합니다. (유니코드)
Drawtext DrawText 함수는 지정된 사각형에 서식이 지정된 텍스트를 그립니다. 지정된 메서드(탭 확장, 문자 양쪽 맞춤, 줄 바꿈 등)에 따라 텍스트의 서식을 지정합니다. (DrawTextW)
DrawTextA DrawText 함수는 지정된 사각형에 서식이 지정된 텍스트를 그립니다. 지정된 메서드(탭 확장, 문자 양쪽 맞춤, 줄 바꿈 등)에 따라 텍스트의 서식을 지정합니다. (DrawTextA)
DrawTextExA DrawTextEx 함수는 지정된 사각형에 서식이 지정된 텍스트를 그립니다. (ANSI)
DrawTextExW DrawTextEx 함수는 지정된 사각형에 서식이 지정된 텍스트를 그립니다. (유니코드)
DrawTextW DrawText 함수는 지정된 사각형에 서식이 지정된 텍스트를 그립니다. 지정된 메서드(탭 확장, 문자 양쪽 맞춤, 줄 바꿈 등)에 따라 텍스트의 서식을 지정합니다. (DrawTextW)
타원 Ellipse 함수는 타원을 그립니다. 타원의 중심은 지정된 경계 사각형의 중심입니다. 타원은 현재 펜을 사용하여 간략하게 설명되며 현재 브러시를 사용하여 채워집니다.
EndPaint EndPaint 함수는 지정된 창에서 그리기 끝을 표시합니다. 이 함수는 BeginPaint 함수를 호출할 때마다 필요하지만 그리기는 완료된 후에만 필요합니다.
EndPath EndPath 함수는 경로 대괄호를 닫고 대괄호로 정의된 경로를 지정된 디바이스 컨텍스트로 선택합니다.

ENHMFENUMPROC EnhMetaFileProc 함수는 EnumEnhMetaFile 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
EnumDisplayDevicesA EnumDisplayDevices 함수를 사용하면 현재 세션의 디스플레이 디바이스에 대한 정보를 가져올 수 있습니다. (ANSI)
EnumDisplayDevicesW EnumDisplayDevices 함수를 사용하면 현재 세션의 디스플레이 디바이스에 대한 정보를 가져올 수 있습니다. (유니코드)
EnumDisplayMonitors EnumDisplayMonitors 함수는 지정된 클리핑 사각형의 교집합과 디바이스 컨텍스트의 표시되는 영역에 의해 형성된 영역을 교차하는 디스플레이 모니터(미러링 드라이버와 연결된 보이지 않는 의사 모니터 포함)를 열거합니다. EnumDisplayMonitors는 열거된 각 모니터에 대해 애플리케이션 정의 MonitorEnumProc 콜백 함수를 한 번 호출합니다. GetSystemMetrics(SM_CMONITORS)는 디스플레이 모니터만 계산합니다.
EnumDisplaySettingsA EnumDisplaySettings 함수는 디스플레이 디바이스의 그래픽 모드 중 하나에 대한 정보를 검색합니다. 디스플레이 디바이스의 모든 그래픽 모드에 대한 정보를 검색하려면 이 함수를 일련의 호출합니다. (ANSI)
EnumDisplaySettingsExA EnumDisplaySettingsEx 함수는 디스플레이 디바이스의 그래픽 모드 중 하나에 대한 정보를 검색합니다. 디스플레이 디바이스의 모든 그래픽 모드에 대한 정보를 검색하려면 이 함수를 일련의 호출합니다. (ANSI)
EnumDisplaySettingsExW EnumDisplaySettingsEx 함수는 디스플레이 디바이스의 그래픽 모드 중 하나에 대한 정보를 검색합니다. 디스플레이 디바이스의 모든 그래픽 모드에 대한 정보를 검색하려면 이 함수를 일련의 호출합니다. (유니코드)
EnumDisplaySettingsW EnumDisplaySettings 함수는 디스플레이 디바이스의 그래픽 모드 중 하나에 대한 정보를 검색합니다. 디스플레이 디바이스의 모든 그래픽 모드에 대한 정보를 검색하려면 이 함수를 일련의 호출합니다. (유니코드)

EnumEnhMetaFile EnumEnhMetaFile 함수는 각 레코드를 검색하고 지정된 콜백 함수에 전달하여 고급 형식 메타파일 내에서 레코드를 열거합니다.
EnumFontFamiliesA EnumFontFamilies 함수는 지정된 디바이스에서 사용할 수 있는 지정된 글꼴 패밀리의 글꼴을 열거합니다. (ANSI)
EnumFontFamiliesExA EnumFontFamiliesEx 함수는 LOGFONT 구조체에 지정된 글꼴 특성과 일치하는 시스템의 고유한 글꼴을 모두 열거합니다. EnumFontFamiliesEx는 서체 이름, 문자 집합 또는 둘 다를 기반으로 글꼴을 열거합니다. (ANSI)
EnumFontFamiliesExW EnumFontFamiliesEx 함수는 LOGFONT 구조체에 지정된 글꼴 특성과 일치하는 시스템의 고유한 글꼴을 모두 열거합니다. EnumFontFamiliesEx는 서체 이름, 문자 집합 또는 둘 다를 기반으로 글꼴을 열거합니다. (유니코드)
EnumFontFamiliesW EnumFontFamilies 함수는 지정된 디바이스에서 사용할 수 있는 지정된 글꼴 패밀리의 글꼴을 열거합니다. (유니코드)
EnumFontsA EnumFonts 함수는 지정된 디바이스에서 사용할 수 있는 글꼴을 열거합니다. (ANSI)
EnumFontsW EnumFonts 함수는 지정된 디바이스에서 사용할 수 있는 글꼴을 열거합니다. (유니코드)
EnumMetaFile EnumMetaFile 함수는 각 레코드를 검색하고 지정된 콜백 함수에 전달하여 Windows 형식 메타파일 내에서 레코드를 열거합니다.
EnumObjects EnumObjects 함수는 지정된 DC(디바이스 컨텍스트)에 사용할 수 있는 펜 또는 브러시를 열거합니다.
EqualRect EqualRect 함수는 지정된 두 사각형이 왼쪽 위와 오른쪽 아래 모서리의 좌표를 비교하여 같은지 여부를 결정합니다.

EqualRgn EqualRgn 함수는 지정된 두 지역을 확인하여 동일한지 여부를 확인합니다. 함수는 크기와 도형이 같은 경우 두 영역이 동일한 것으로 간주합니다.
ExcludeClipRect ExcludeClipRect 함수는 지정된 사각형을 뺀 기존 클리핑 영역으로 구성된 새 클리핑 영역을 만듭니다.
ExcludeUpdateRgn ExcludeUpdateRgn 함수는 창의 업데이트된 영역을 클리핑 영역에서 제외하여 창의 잘못된 영역 내에서 그리는 것을 방지합니다.
ExtCreatePen ExtCreatePen 함수는 지정된 스타일, 너비 및 브러시 특성을 가진 논리적 화장품 또는 기하학적 펜을 만듭니다.
ExtCreateRegion ExtCreateRegion 함수는 지정된 지역 및 변환 데이터에서 지역을 만듭니다.
ExtFloodFill ExtFloodFill 함수는 디스플레이 표면의 영역을 현재 브러시로 채웁니다.
ExtSelectClipRgn ExtSelectClipRgn 함수는 지정된 모드를 사용하여 지정된 영역을 현재 클리핑 영역과 결합합니다.
ExtTextOutA ExtTextOut 함수는 현재 선택한 글꼴, 배경색 및 텍스트 색을 사용하여 텍스트를 그립니다. 필요에 따라 클리핑, 불투명 또는 둘 다에 사용할 차원을 제공할 수 있습니다. (ANSI)
ExtTextOutW ExtTextOut 함수는 현재 선택한 글꼴, 배경색 및 텍스트 색을 사용하여 텍스트를 그립니다. 필요에 따라 클리핑, 불투명 또는 둘 다에 사용할 차원을 제공할 수 있습니다. (유니코드)
FillPath FillPath 함수는 현재 경로의 열린 그림을 닫고 현재 브러시 및 다각형 채우기 모드를 사용하여 경로의 내부를 채웁니다.

FillRect FillRect 함수는 지정된 브러시를 사용하여 사각형을 채웁니다. 이 함수는 왼쪽 및 위쪽 테두리를 포함하지만 사각형의 오른쪽 및 아래쪽 테두리는 제외합니다.
FillRgn FillRgn 함수는 지정된 브러시를 사용하여 영역을 채웁니다.
FlattenPath FlattenPath 함수는 선택한 경로의 모든 곡선을 현재 DC(디바이스 컨텍스트)로 변환하여 각 곡선을 선 시퀀스로 변환합니다.
FloodFill FloodFill 함수는 디스플레이 표면의 영역을 현재 브러시로 채웁니다. 영역은 색 매개 변수에 지정된 대로 바인딩된 것으로 간주됩니다.
FrameRect FrameRect 함수는 지정된 브러시를 사용하여 지정된 사각형 주위에 테두리를 그립니다. 테두리의 너비와 높이는 항상 하나의 논리 단위입니다.
FrameRgn FrameRgn 함수는 지정된 브러시를 사용하여 지정된 영역 주위에 테두리를 그립니다.
GdiAlphaBlend GdiAlphaBlend 함수는 투명 또는 반투명 픽셀이 있는 비트맵을 표시합니다.
GdiComment GdiComment 함수는 버퍼의 주석을 지정된 고급 형식 메타파일로 복사합니다.
GdiFlush GdiFlush 함수는 호출 스레드의 현재 일괄 처리를 플러시합니다.
GdiGetBatchLimit GdiGetBatchLimit 함수는 호출 스레드의 현재 일괄 처리에 누적될 수 있는 최대 함수 호출 수를 반환합니다. 시스템은 이 제한을 초과할 때마다 현재 일괄 처리를 플러시합니다.
GdiGradientFill GdiGradientFill 함수는 사각형 및 삼각형 구조를 채웁니다.

GdiSetBatchLimit GdiSetBatchLimit 함수는 호출 스레드의 현재 일괄 처리에 누적될 수 있는 최대 함수 호출 수를 설정합니다. 시스템은 이 제한을 초과할 때마다 현재 일괄 처리를 플러시합니다.
GdiTransparentBlt GdiTransparentBlt 함수는 지정된 원본 디바이스 컨텍스트에서 대상 디바이스 컨텍스트로 픽셀 사각형에 해당하는 색 데이터의 비트 블록 전송을 수행합니다.
GetArcDirection GetArcDirection 함수는 지정된 디바이스 컨텍스트에 대한 현재 호 방향을 검색합니다. 호 및 사각형 함수는 호 방향을 사용합니다.
GetAspectRatioFilterEx GetAspectRatioFilterEx 함수는 현재 가로 세로 비율 필터에 대한 설정을 검색합니다.
GetBitmapBits GetBitmapBits 함수는 지정된 디바이스 종속 비트맵의 비트맵 비트를 버퍼에 복사합니다.
GetBitmapDimensionEx GetBitmapDimensionEx 함수는 호환되는 비트맵의 차원을 검색합니다. 검색된 차원은 SetBitmapDimensionEx 함수에 의해 설정되어야 합니다.
GetBkColor GetBkColor 함수는 지정된 디바이스 컨텍스트의 현재 배경색을 반환합니다.
GetBkMode GetBkMode 함수는 지정된 디바이스 컨텍스트에 대한 현재 백그라운드 혼합 모드를 반환합니다. 디바이스 컨텍스트의 배경 혼합 모드는 실선이 아닌 텍스트, 부화 브러시 및 펜 스타일에 영향을 줍니다.
GetBoundsRect GetBoundsRect 함수는 지정된 디바이스 컨텍스트에 대해 현재 누적된 경계 사각형을 가져옵니다.
GetBrushOrgEx GetBrushOrgEx 함수는 지정된 디바이스 컨텍스트에 대한 현재 브러시 원본을 검색합니다. 이 함수는 GetBrushOrg 함수를 대체합니다.

GetBValue GetBValue 매크로는 빨강, 녹색, 파랑(RGB) 값의 파란색 구성 요소에 대한 강도 값을 검색합니다.
GetCharABCWidthsA GetCharABCWidths 함수는 현재 TrueType 글꼴에서 지정된 범위의 연속 문자의 너비를 논리적 단위로 검색합니다. 이 함수는 TrueType 글꼴로만 성공합니다. (ANSI)
GetCharABCWidthsFloatA GetCharABCWidthsFloat 함수는 현재 글꼴에서 지정된 범위의 연속 문자 너비를 논리적 단위로 검색합니다. (ANSI)
GetCharABCWidthsFloatW GetCharABCWidthsFloat 함수는 현재 글꼴에서 지정된 범위의 연속 문자 너비를 논리적 단위로 검색합니다. (유니코드)
GetCharABCWidthsI GetCharABCWidthsI 함수는 현재 TrueType 글꼴에서 지정된 범위의 연속 문자 모양 인덱스의 너비를 논리 단위로 검색합니다. 이 함수는 TrueType 글꼴로만 성공합니다.
GetCharABCWidthsW GetCharABCWidths 함수는 현재 TrueType 글꼴에서 지정된 범위의 연속 문자의 너비를 논리적 단위로 검색합니다. 이 함수는 TrueType 글꼴로만 성공합니다. (유니코드)
GetCharacterPlacementA GetCharacterPlacement 함수는 문자 너비, 캐리트 위치 지정, 문자열 내 순서 지정 및 문자 모양 렌더링과 같은 문자열에 대한 정보를 검색합니다. (ANSI)
GetCharacterPlacementW GetCharacterPlacement 함수는 문자 너비, 캐리트 위치 지정, 문자열 내 순서 지정 및 문자 모양 렌더링과 같은 문자열에 대한 정보를 검색합니다. (유니코드)
GetCharWidth32A GetCharWidth32 함수는 현재 글꼴에서 지정된 범위의 연속 문자의 너비를 논리적 좌표로 검색합니다. (ANSI)
GetCharWidth32W GetCharWidth32 함수는 현재 글꼴에서 지정된 범위의 연속 문자의 너비를 논리적 좌표로 검색합니다. (유니코드)

GetCharWidthA GetCharWidth 함수는 현재 글꼴에서 지정된 범위의 연속 문자의 너비를 논리적 좌표로 검색합니다. (ANSI)
GetCharWidthFloatA GetCharWidthFloat 함수는 현재 글꼴에서 지정된 범위에서 연속 문자의 소수 자릿수 너비를 검색합니다. (ANSI)
GetCharWidthFloatW GetCharWidthFloat 함수는 현재 글꼴에서 지정된 범위에서 연속 문자의 소수 자릿수 너비를 검색합니다. (유니코드)
GetCharWidthI GetCharWidthI 함수는 현재 글꼴에서 지정된 범위의 연속 문자 모양 인덱스의 너비를 논리적 좌표로 검색합니다.
GetCharWidthW GetCharWidth 함수는 현재 글꼴에서 지정된 범위의 연속 문자의 너비를 논리적 좌표로 검색합니다. (유니코드)
GetClipBox GetClipBox 함수는 디바이스의 현재 표시 영역 주위에 그릴 수 있는 가장 좁은 경계 사각형의 크기를 검색합니다.
GetClipRgn GetClipRgn 함수는 지정된 디바이스 컨텍스트에 대한 현재 애플리케이션 정의 클리핑 영역을 식별하는 핸들을 검색합니다.
GetColorAdjustment GetColorAdjustment 함수는 지정된 DC(디바이스 컨텍스트)에 대한 색 조정 값을 검색합니다.
GetCurrentObject GetCurrentObject 함수는 지정된 DC(디바이스 컨텍스트)로 선택된 지정된 형식의 개체에 대한 핸들을 검색합니다.
GetCurrentPositionEx GetCurrentPositionEx 함수는 논리 좌표에서 현재 위치를 검색합니다.

Getdc GetDC 함수는 지정된 창의 클라이언트 영역 또는 전체 화면에 대한 DC(디바이스 컨텍스트)에 대한 핸들을 검색합니다.
GetDCBrushColor GetDCBrushColor 함수는 지정된 DC(디바이스 컨텍스트)에 대한 현재 브러시 색을 검색합니다.
GetDCEx GetDCEx 함수는 지정된 창의 클라이언트 영역 또는 전체 화면에 대한 DC(디바이스 컨텍스트)에 대한 핸들을 검색합니다.
GetDCOrgEx GetDCOrgEx 함수는 지정된 DC(디바이스 컨텍스트)에 대한 최종 번역 원본을 검색합니다.
GetDCPenColor GetDCPenColor 함수는 지정된 DC(디바이스 컨텍스트)에 대한 현재 펜 색을 검색합니다.
GetDeviceCaps GetDeviceCaps 함수는 지정된 디바이스에 대한 디바이스별 정보를 검색합니다.
GetDIBColorTable GetDIBColorTable 함수는 현재 지정된 디바이스 컨텍스트로 선택된 DIB 섹션 비트맵의 색 테이블에 있는 항목 범위에서 RGB(빨강, 녹색, 파랑) 색 값을 검색합니다.
GetDIBits GetDIBits 함수는 지정된 호환 가능한 비트맵의 비트를 검색하고 지정된 형식을 사용하여 버퍼에 DIB로 복사합니다.
GetEnhMetaFileA GetEnhMetaFile 함수는 지정된 파일에 저장된 고급 형식 메타파일을 식별하는 핸들을 만듭니다. (ANSI)
GetEnhMetaFileBits GetEnhMetaFileBits 함수는 지정된 고급 형식 메타파일의 내용을 검색하여 버퍼에 복사합니다.
GetEnhMetaFileDescriptionA GetEnhMetaFileDescription 함수는 고급 형식 메타파일에서 선택적 텍스트 설명을 검색하고 지정된 버퍼에 문자열을 복사합니다. (ANSI)

GetEnhMetaFileDescriptionW GetEnhMetaFileDescription 함수는 고급 형식 메타파일에서 선택적 텍스트 설명을 검색하고 지정된 버퍼에 문자열을 복사합니다. (유니코드)
GetEnhMetaFileHeader GetEnhMetaFileHeader 함수는 지정된 고급 형식 메타파일의 헤더가 포함된 레코드를 검색합니다.
GetEnhMetaFilePaletteEntries GetEnhMetaFilePaletteEntries 함수는 지정된 향상된 메타파일에서 선택적 팔레트 항목을 검색합니다.
GetEnhMetaFileW GetEnhMetaFile 함수는 지정된 파일에 저장된 고급 형식 메타파일을 식별하는 핸들을 만듭니다. (유니코드)
GetFontData GetFontData 함수는 TrueType 글꼴에 대한 글꼴 메트릭 데이터를 검색합니다.
GetFontLanguageInfo GetFontLanguageInfo 함수는 지정된 표시 컨텍스트에 대해 현재 선택된 글꼴에 대한 정보를 반환합니다. 애플리케이션은 일반적으로 이 정보와 GetCharacterPlacement 함수를 사용하여 표시할 문자열을 준비합니다.
GetFontUnicodeRanges GetFontUnicodeRanges 함수는 글꼴에서 지원되는 유니코드 문자에 대한 정보를 반환합니다. 정보는 GLYPHSET 구조체로 반환됩니다.
GetGlyphIndicesA GetGlyphIndices 함수는 문자열을 문자 모양 인덱스 배열로 변환합니다. 함수를 사용하여 글꼴에 문자 모양이 있는지 여부를 확인할 수 있습니다. (ANSI)
GetGlyphIndicesW GetGlyphIndices 함수는 문자열을 문자 모양 인덱스 배열로 변환합니다. 함수를 사용하여 글꼴에 문자 모양이 있는지 여부를 확인할 수 있습니다. (유니코드)
GetGlyphOutlineA GetGlyphOutline 함수는 TrueType 글꼴에서 지정된 디바이스 컨텍스트로 선택된 문자의 윤곽선 또는 비트맵을 검색합니다. (ANSI)

GetGlyphOutlineW GetGlyphOutline 함수는 TrueType 글꼴에서 지정된 디바이스 컨텍스트로 선택된 문자의 윤곽선 또는 비트맵을 검색합니다. (유니코드)
GetGraphicsMode GetGraphicsMode 함수는 지정된 디바이스 컨텍스트에 대한 현재 그래픽 모드를 검색합니다.
GetGValue GetGValue 매크로는 빨강, 녹색, 파랑(RGB) 값의 녹색 구성 요소에 대한 강도 값을 검색합니다.
GetKerningPairsA GetKerningPairs 함수는 지정된 디바이스 컨텍스트에 대해 현재 선택된 글꼴의 문자 커닝 쌍을 검색합니다. (ANSI)
GetKerningPairsW GetKerningPairs 함수는 지정된 디바이스 컨텍스트에 대해 현재 선택된 글꼴의 문자 커닝 쌍을 검색합니다. (유니코드)
GetLayout GetLayout 함수는 DC(디바이스 컨텍스트)의 레이아웃을 반환합니다.
GetMapMode GetMapMode 함수는 현재 매핑 모드를 검색합니다.
GetMetaFileA GetMetaFile 함수는 지정된 파일에 저장된 메타파일을 식별하는 핸들을 만듭니다. (ANSI)
GetMetaFileBitsEx GetMetaFileBitsEx 함수는 Windows 형식 메타파일의 내용을 검색하고 지정된 버퍼에 복사합니다.
GetMetaFileW GetMetaFile 함수는 지정된 파일에 저장된 메타파일을 식별하는 핸들을 만듭니다. (유니코드)
GetMetaRgn GetMetaRgn 함수는 지정된 디바이스 컨텍스트에 대한 현재 메타레지온을 검색합니다.

GetMiterLimit GetMiterLimit 함수는 지정된 디바이스 컨텍스트에 대한 miter 제한을 검색합니다.
GetMonitorInfoA GetMonitorInfo 함수는 디스플레이 모니터에 대한 정보를 검색합니다. (ANSI)
GetMonitorInfoW GetMonitorInfo 함수는 디스플레이 모니터에 대한 정보를 검색합니다. (유니코드)
GetNearestColor GetNearestColor 함수는 지정된 색 값을 사용할 때 표시되는 시스템 색상표에서 색을 식별하는 색 값을 검색합니다.
GetNearestPaletteIndex GetNearestPaletteIndex 함수는 지정된 색 값과 가장 일치하는 지정된 논리 팔레트의 항목에 대한 인덱스를 검색합니다.
GetObject GetObject 함수(wingdi.h)는 지정된 그래픽 개체에 대한 정보를 검색합니다.
GetObjectA GetObject 함수는 지정된 그래픽 개체에 대한 정보를 검색합니다. (GetObjectA)
GetObjectType GetObjectType은 지정된 개체의 형식을 검색합니다.
GetObjectW GetObjectW(유니코드) 함수(wingdi.h)는 지정된 그래픽 개체에 대한 정보를 검색합니다.
GetOutlineTextMetricsA GetOutlineTextMetrics 함수는 TrueType 글꼴에 대한 텍스트 메트릭을 검색합니다. (ANSI)
GetOutlineTextMetricsW GetOutlineTextMetrics 함수는 TrueType 글꼴에 대한 텍스트 메트릭을 검색합니다. (유니코드)
GetPaletteEntries GetPaletteEntries 함수는 지정된 논리 팔레트에서 지정된 색상표 항목 범위를 검색합니다.

GetPath GetPath 함수는 지정된 디바이스 컨텍스트로 선택된 경로에 있는 곡선의 제어점 및 선의 엔드포인트를 정의하는 좌표를 검색합니다.
GetPixel GetPixel 함수는 지정된 좌표에서 픽셀의 빨강, 녹색, 파랑(RGB) 색 값을 검색합니다.
GetPolyFillMode GetPolyFillMode 함수는 현재 다각형 채우기 모드를 검색합니다.
GetRandomRgn GetRandomRgn 함수는 지정된 디바이스 컨텍스트의 시스템 클리핑 영역을 특정 지역에 복사합니다.
GetRasterizerCaps GetRasterizerCaps 함수는 TrueType 글꼴이 시스템에 설치되어 있는지 여부를 나타내는 플래그를 반환합니다.
GetRegionData GetRegionData 함수는 지정된 버퍼를 지역을 설명하는 데이터로 채웁니다. 이 데이터에는 영역을 구성하는 사각형의 크기가 포함됩니다.
GetRgnBox GetRgnBox 함수는 지정된 영역의 경계 사각형을 검색합니다.
GetROP2 GetROP2 함수는 지정된 디바이스 컨텍스트의 포그라운드 혼합 모드를 검색합니다. 혼합 모드는 펜 또는 내부 색과 이미 화면에 있는 색을 결합하여 새 색을 생성하는 방법을 지정합니다.
GetRValue GetRValue 매크로는 빨강, 녹색, 파랑(RGB) 값의 빨간색 구성 요소에 대한 강도 값을 검색합니다.
GetStockObject GetStockObject 함수는 스톡 펜, 브러시, 글꼴 또는 팔레트 중 하나에 대한 핸들을 검색합니다.
GetStretchBltMode GetStretchBltMode 함수는 현재 스트레치 모드를 검색합니다. 스트레치 모드는 StretchBlt 함수가 호출될 때 늘어나거나 압축되는 비트맵에서 색 데이터를 추가하거나 제거하는 방법을 정의합니다.

GetSysColorBrush GetSysColorBrush 함수는 지정된 색 인덱스에 해당하는 논리 브러시를 식별하는 핸들을 검색합니다.
GetSystemPaletteEntries GetSystemPaletteEntries 함수는 지정된 DC(디바이스 컨텍스트)와 연결된 시스템 팔레트에서 색상표 항목의 범위를 검색합니다.
GetSystemPaletteUse GetSystemPaletteUse 함수는 지정된 DC(디바이스 컨텍스트)에 대한 시스템(물리적) 팔레트의 현재 상태를 검색합니다.
GetTabbedTextExtentA GetTabbedTextExtent 함수는 문자열의 너비와 높이를 계산합니다. (ANSI)
GetTabbedTextExtentW GetTabbedTextExtent 함수는 문자열의 너비와 높이를 계산합니다. (유니코드)
GetTextAlign GetTextAlign 함수는 지정된 디바이스 컨텍스트에 대한 텍스트 맞춤 설정을 검색합니다.
GetTextCharacterExtra GetTextCharacterExtra 함수는 지정된 디바이스 컨텍스트에 대한 현재 문자 간 간격을 검색합니다.
GetTextColor GetTextColor 함수는 지정된 디바이스 컨텍스트의 현재 텍스트 색을 검색합니다.
GetTextExtentExPointA GetTextExtentExPoint 함수는 지정된 공백 내에 들어갈 지정된 문자열의 문자 수를 검색하고 각 문자에 대한 텍스트 범위로 배열을 채웁니다. (ANSI)
GetTextExtentExPointI GetTextExtentExPointI 함수는 지정된 공백 내에 들어갈 지정된 문자열의 문자 수를 검색하고 각 문자에 대한 텍스트 범위로 배열을 채웁니다.
GetTextExtentExPointW GetTextExtentExPoint 함수는 지정된 공백 내에 들어갈 지정된 문자열의 문자 수를 검색하고 각 문자에 대한 텍스트 범위로 배열을 채웁니다. (유니코드)

GetTextExtentPoint32A GetTextExtentPoint32 함수는 지정된 텍스트 문자열의 너비와 높이를 계산합니다. (ANSI)
GetTextExtentPoint32W GetTextExtentPoint32 함수는 지정된 텍스트 문자열의 너비와 높이를 계산합니다. (유니코드)
GetTextExtentPointA GetTextExtentPoint 함수는 지정된 텍스트 문자열의 너비와 높이를 계산합니다. (ANSI)
GetTextExtentPointI GetTextExtentPointI 함수는 지정된 문자 모양 인덱스 배열의 너비와 높이를 계산합니다.
GetTextExtentPointW GetTextExtentPoint 함수는 지정된 텍스트 문자열의 너비와 높이를 계산합니다. (유니코드)
GetTextFaceA GetTextFace 함수는 지정된 디바이스 컨텍스트에 선택된 글꼴의 서체 이름을 검색합니다. (ANSI)
GetTextFaceW GetTextFace 함수는 지정된 디바이스 컨텍스트에 선택된 글꼴의 서체 이름을 검색합니다. (유니코드)
GetTextMetrics GetTextMetrics 함수(wingdi.h)는 지정된 버퍼를 현재 선택한 글꼴에 대한 메트릭으로 채웁니다.
GetTextMetricsA GetTextMetrics 함수는 지정된 버퍼를 현재 선택한 글꼴에 대한 메트릭으로 채웁니다. (GetTextMetricsA)
GetTextMetricsW GetTextMetricsW(유니코드) 함수(wingdi.h)는 지정된 버퍼를 현재 선택한 글꼴에 대한 메트릭으로 채웁니다.
GetUpdateRect GetUpdateRect 함수는 지정된 창의 업데이트 영역을 완전히 묶는 가장 작은 사각형의 좌표를 검색합니다.

GetUpdateRgn GetUpdateRgn 함수는 지정된 지역에 복사하여 창의 업데이트 영역을 검색합니다. 업데이트 영역의 좌표는 창의 왼쪽 위 모서리(즉, 클라이언트 좌표)를 기준으로 합니다.
GetViewportExtEx GetViewportExtEx 함수는 지정된 디바이스 컨텍스트에 대한 현재 뷰포트의 x 익스텐트 및 y 익스텐트를 검색합니다.
GetViewportOrgEx GetViewportOrgEx 함수는 지정된 디바이스 컨텍스트에 대한 뷰포트 원본의 x 좌표 및 y 좌표를 검색합니다.
GetWindowDC GetWindowDC 함수는 제목 표시줄, 메뉴 및 스크롤 막대를 포함하여 전체 창에 대한 디바이스 컨텍스트(DC)를 검색합니다.
GetWindowExtEx 이 함수는 지정된 디바이스 컨텍스트에 대한 창의 x 익스텐트 및 y 익스텐트 를 검색합니다.
GetWindowOrgEx GetWindowOrgEx 함수는 지정된 디바이스 컨텍스트에 대한 창 원본의 x 좌표 및 y 좌표를 검색합니다.
GetWindowRgn GetWindowRgn 함수는 창의 창 영역 복사본을 가져옵니다.
GetWindowRgnBox GetWindowRgnBox 함수는 창의 창 영역에 대한 가장 엄격한 경계 사각형의 크기를 검색합니다.
GetWinMetaFileBits GetWinMetaFileBits 함수는 메타파일의 향상된 형식 레코드를 Windows 형식 레코드로 변환하고 변환된 레코드를 지정된 버퍼에 저장합니다.
GetWorldTransform GetWorldTransform 함수는 현재 세계 공간에서 페이지 공간으로의 변환을 검색합니다.

GOBJENUMPROC EnumObjectsProc 함수는 EnumObjects 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
GradientFill GradientFill 함수는 사각형 및 삼각형 구조를 채웁니다.
GrayStringA GrayString 함수는 지정된 위치에 회색 텍스트를 그립니다. (ANSI)
GRAYSTRINGPROC OutputProc 함수는 GrayString 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
GrayStringW GrayString 함수는 지정된 위치에 회색 텍스트를 그립니다. (유니코드)
InflateRect InflateRect 함수는 지정된 사각형의 너비와 높이를 늘리거나 줄입니다.
IntersectClipRect IntersectClipRect 함수는 현재 클리핑 영역과 지정된 사각형의 교차점에서 새 클리핑 영역을 만듭니다.
IntersectRect IntersectRect 함수는 두 원본 사각형의 교집합을 계산하고 교차 사각형의 좌표를 대상 사각형에 배치합니다.
InvalidateRect InvalidateRect 함수는 지정된 창의 업데이트 영역에 사각형을 추가합니다. 업데이트 지역은 다시 그려야 하는 창의 클라이언트 영역 부분을 나타냅니다.
InvalidateRgn InvalidateRgn 함수는 창의 현재 업데이트 영역에 추가하여 지정된 지역 내의 클라이언트 영역을 무효화합니다.
InvertRect InvertRect 함수는 사각형 내부의 각 픽셀에 대한 색 값에 대해 논리 NOT 연산을 수행하여 창에서 사각형을 반전합니다.

InvertRgn InvertRgn 함수는 지정된 영역의 색을 반전합니다.
IsRectEmpty IsRectEmpty 함수는 지정된 사각형이 비어 있는지 여부를 결정합니다.
LineDDA LineDDA 함수는 지정된 시작점과 끝점으로 정의된 선에 대해 강조 표시해야 하는 픽셀을 결정합니다.
LINEDDAPROC LineDDAProc 함수는 LineDDA 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
Lineto LineTo 함수는 현재 위치에서 지정된 지점까지 선을 그립니다.
LoadBitmapA LoadBitmap 함수는 모듈의 실행 파일에서 지정된 비트맵 리소스를 로드합니다. (ANSI)
LoadBitmapW LoadBitmap 함수는 모듈의 실행 파일에서 지정된 비트맵 리소스를 로드합니다. (유니코드)
LockWindowUpdate LockWindowUpdate 함수는 지정된 창에서 그리기를 사용하지 않도록 설정하거나 사용하도록 설정합니다. 한 번에 하나의 창만 잠글 수 있습니다.
LPtoDP LPtoDP 함수는 논리 좌표를 디바이스 좌표로 변환합니다. 변환은 디바이스 컨텍스트의 매핑 모드, 창 및 뷰포트의 원본 및 익스텐트 설정 및 월드 변환에 따라 달라집니다.
MAKEPOINTS MAKEPOINTS 매크로는 점의 x 및 y 좌표를 포함하는 값을 POINTS 구조체로 변환합니다.
MAKEROP4 MAKEROP4 매크로는 MaskBlt 함수와 함께 사용할 4차 래스터 작업 코드를 만듭니다.

MapWindowPoints MapWindowPoints 함수는 한 창을 기준으로 한 좌표 공간에서 다른 창을 기준으로 좌표 공간으로 점 집합을 변환(매핑)합니다.
MaskBlt MaskBlt 함수는 지정된 마스크 및 래스터 작업을 사용하여 원본 및 대상 비트맵의 색 데이터를 결합합니다.
MergeFontPackage MergeFontPackage 함수는 CreateFontPackage에서 만든 글꼴을 조작합니다.
MFENUMPROC EnumMetaFileProc 함수는 Windows 형식 메타파일 레코드를 처리하는 애플리케이션 정의 콜백 함수입니다.
ModifyWorldTransform ModifyWorldTransform 함수는 지정된 모드를 사용하여 디바이스 컨텍스트의 월드 변환을 변경합니다.
MONITORENUMPROC MonitorEnumProc 함수는 EnumDisplayMonitors 함수에 의해 호출되는 애플리케이션 정의 콜백 함수입니다.
MonitorFromPoint MonitorFromPoint 함수는 지정된 지점을 포함하는 디스플레이 모니터에 대한 핸들을 검색합니다.
MonitorFromRect MonitorFromRect 함수는 지정된 사각형과 가장 큰 교집합 영역이 있는 디스플레이 모니터에 대한 핸들을 검색합니다.
MonitorFromWindow MonitorFromWindow 함수는 지정된 창의 경계 사각형과 가장 큰 교집합 영역이 있는 디스플레이 모니터에 대한 핸들을 검색합니다.
MoveToEx MoveToEx 함수는 현재 위치를 지정된 지점으로 업데이트하고 필요에 따라 이전 위치를 반환합니다.

OffsetClipRgn OffsetClipRgn 함수는 디바이스 컨텍스트의 클리핑 영역을 지정된 오프셋으로 이동합니다.
OffsetRect OffsetRect 함수는 지정된 사각형을 지정된 오프셋으로 이동합니다.
OffsetRgn OffsetRgn 함수는 지정된 오프셋별로 영역을 이동합니다.
OffsetViewportOrgEx OffsetViewportOrgEx 함수는 지정된 가로 및 세로 오프셋을 사용하여 디바이스 컨텍스트에 대한 뷰포트 원본을 수정합니다.
OffsetWindowOrgEx OffsetWindowOrgEx 함수는 지정된 가로 및 세로 오프셋을 사용하여 디바이스 컨텍스트의 창 원본을 수정합니다.
PaintDesktop PaintDesktop 함수는 지정된 디바이스 컨텍스트의 클리핑 영역을 바탕 화면 패턴 또는 배경 화면으로 채웁니다. 함수는 주로 셸 데스크톱용으로 제공됩니다.
PaintRgn PaintRgn 함수는 현재 디바이스 컨텍스트에 선택된 브러시를 사용하여 지정된 영역을 그립니다.
PALETTEINDEX PALETTEINDEX 매크로는 논리적 색상표 항목에 대한 인덱스를 허용하고 지정된 인덱스와 연결된 색을 지정하는 COLORREF 값으로 구성된 색상표 항목 지정자를 반환합니다.
PALETTERGB PALETTERGB 매크로는 빨강, 녹색 및 파랑의 상대 강도를 나타내는 세 개의 값을 허용하고, 상위 바이트에서 2로 구성된 색상표 상대 빨강, 녹색, 파랑(RGB) 지정자 및 3개의 낮은 순서 바이트에서 RGB 값을 반환합니다. 색 팔레트를 사용하는 애플리케이션은 명시적 RGB 값 대신 이 지정자를 색을 예상하는 함수에 전달할 수 있습니다.
PatBlt PatBlt 함수는 현재 지정된 디바이스 컨텍스트로 선택된 브러시를 사용하여 지정된 사각형을 그립니다. 브러시 색과 표면 색 또는 색은 지정된 래스터 연산을 사용하여 결합됩니다.

PathToRegion PathToRegion 함수는 선택한 경로에서 지정된 디바이스 컨텍스트로 영역을 만듭니다. 결과 지역은 디바이스 좌표를 사용합니다.
Pie 원형 함수는 타원과 두 방사형의 교차점에 의해 경계가 지정된 원형 모양의 웨지를 그립니다. 원형은 현재 펜을 사용하여 윤곽선으로 표시되고 현재 브러시를 사용하여 채워집니다.
PlayEnhMetaFile PlayEnhMetaFile 함수는 지정된 고급 형식 메타파일에 저장된 그림을 표시합니다.
PlayEnhMetaFileRecord PlayEnhMetaFileRecord 함수는 레코드로 식별된 GDI(그래픽 디바이스 인터페이스) 함수를 실행하여 향상된 메타파일 레코드를 재생합니다.
PlayMetaFile PlayMetaFile 함수는 지정된 디바이스의 지정된 Windows 형식 메타파일에 저장된 그림을 표시합니다.
PlayMetaFileRecord PlayMetaFileRecord 함수는 해당 레코드에 포함된 GDI(그래픽 디바이스 인터페이스) 함수를 실행하여 Windows 형식 메타파일 레코드를 재생합니다.
PlgBlt PlgBlt 함수는 원본 디바이스 컨텍스트의 지정된 사각형에서 대상 디바이스 컨텍스트의 지정된 병렬로 색 데이터의 비트를 비트 블록 전송합니다.
POINTSTOPOINT POINTSTOPOINT 매크로는 POINTS 구조체의 내용을 POINT 구조체에 복사합니다.
POINTTOPOINTS POINTTOPOINTS 매크로는 POINT 구조를 POINTS 구조체로 변환합니다.
PolyBezier PolyBezier 함수는 하나 이상의 B 지어 곡선을 그립니다.
PolyBezierTo PolyBezierTo 함수는 하나 이상의 B zier 곡선을 그립니다.

PolyDraw PolyDraw 함수는 선 세그먼트 및 B zier 곡선 집합을 그립니다.
Polygon Polygon 함수는 직선으로 연결된 둘 이상의 꼭짓점으로 구성된 다각형을 그립니다. 다각형은 현재 펜을 사용하여 윤곽선이 지정되고 현재 브러시 및 다각형 채우기 모드를 사용하여 채워집니다.
폴리라인 Polyline 함수는 지정된 배열의 점을 연결하여 일련의 선 세그먼트를 그립니다.
PolylineTo PolylineTo 함수는 하나 이상의 직선을 그립니다.
PolyPolygon PolyPolygon 함수는 일련의 닫힌 다각형을 그립니다. 각 다각형은 현재 펜을 사용하여 윤곽선이 지정되고 현재 브러시 및 다각형 채우기 모드를 사용하여 채워집니다. 이 함수에서 그린 다각형은 겹칠 수 있습니다.
PolyPolyline PolyPolyline 함수는 여러 개의 연결된 선 세그먼트를 그립니다.
PolyTextOutA PolyTextOut 함수는 지정된 디바이스 컨텍스트에서 현재 선택된 글꼴 및 텍스트 색을 사용하여 여러 문자열을 그립니다. (ANSI)
PolyTextOutW PolyTextOut 함수는 지정된 디바이스 컨텍스트에서 현재 선택된 글꼴 및 텍스트 색을 사용하여 여러 문자열을 그립니다. (유니코드)
PtInRect PtInRect 함수는 지정된 점이 지정된 사각형 내에 있는지 여부를 결정합니다.
PtInRegion PtInRegion 함수는 지정된 지점이 지정된 지역 내에 있는지 여부를 결정합니다.
PtVisible PtVisible 함수는 지정된 지점이 디바이스 컨텍스트의 클리핑 영역 내에 있는지 여부를 결정합니다.

RealizePalette RealizePalette 함수는 현재 논리 팔레트의 색상표 항목을 시스템 팔레트에 매핑합니다.
사각형 Rectangle 함수는 사각형을 그립니다. 사각형은 현재 펜을 사용하여 윤곽선으로 표시되고 현재 브러시를 사용하여 채워집니다.
RectInRegion RectInRegion 함수는 지정된 사각형의 일부가 영역 경계 내에 있는지 여부를 결정합니다.
RectVisible RectVisible 함수는 지정된 사각형의 일부가 디바이스 컨텍스트의 클리핑 영역 내에 있는지 여부를 결정합니다.
RedrawWindow RedrawWindow 함수는 창의 클라이언트 영역에서 지정된 사각형 또는 영역을 업데이트합니다.
ReleaseDC ReleaseDC 함수는 디바이스 컨텍스트(DC)를 해제하여 다른 애플리케이션에서 사용할 수 있게 합니다. ReleaseDC 함수의 효과는 DC 유형에 따라 달라집니다. 일반 및 창 DC만 해제합니다. 클래스 또는 개인 DC에는 영향을 주지 않습니다.
RemoveFontMemResourceEx RemoveFontMemResourceEx 함수는 메모리 이미지 파일에서 추가된 글꼴을 제거합니다.
RemoveFontResourceA RemoveFontResource 함수는 시스템 글꼴 테이블에서 지정된 파일의 글꼴을 제거합니다. (ANSI)
RemoveFontResourceExA RemoveFontResourceEx 함수는 시스템 글꼴 테이블에서 지정된 파일의 글꼴을 제거합니다. (ANSI)
RemoveFontResourceExW RemoveFontResourceEx 함수는 시스템 글꼴 테이블에서 지정된 파일의 글꼴을 제거합니다. (유니코드)
RemoveFontResourceW RemoveFontResource 함수는 시스템 글꼴 테이블에서 지정된 파일의 글꼴을 제거합니다. (유니코드)

ResetDCA ResetDC 함수는 지정된 정보를 사용하여 지정된 프린터 또는 플로터 DC(디바이스 컨텍스트)를 업데이트합니다. (ANSI)
ResetDCW ResetDC 함수는 지정된 정보를 사용하여 지정된 프린터 또는 플로터 DC(디바이스 컨텍스트)를 업데이트합니다. (유니코드)
ResizePalette ResizePalette 함수는 지정된 값에 따라 논리 팔레트의 크기를 늘리거나 줄입니다.
RestoreDC RestoreDC 함수는 DC(디바이스 컨텍스트)를 지정된 상태로 복원합니다. DC는 SaveDC 함수에 대한 이전 호출로 만든 스택에서 상태 정보를 표시하여 복원됩니다.
RGB RGB 매크로는 제공된 인수 및 출력 디바이스의 색 기능에 따라 빨강, 녹색, 파랑(RGB) 색을 선택합니다.
RoundRect RoundRect 함수는 둥근 모서리가 있는 사각형을 그립니다. 사각형은 현재 펜을 사용하여 윤곽선으로 표시되고 현재 브러시를 사용하여 채워집니다.
SaveDC SaveDC 함수는 선택한 개체 및 그래픽 모드(예: 비트맵, 브러시, 팔레트, 글꼴, 펜, 영역, 그리기 모드 및 매핑 모드)를 설명하는 데이터를 컨텍스트 스택에 복사하여 지정된 DC(디바이스 컨텍스트)의 현재 상태를 저장합니다.
ScaleViewportExtEx ScaleViewportExtEx 함수는 지정된 곱셈 및 제수에 의해 형성된 비율을 사용하여 디바이스 컨텍스트에 대한 뷰포트를 수정합니다.
ScaleWindowExtEx ScaleWindowExtEx 함수는 지정된 곱셈 및 제수에 의해 형성된 비율을 사용하여 디바이스 컨텍스트의 창을 수정합니다.
ScreenToClient ScreenToClient 함수는 화면에서 지정된 지점의 화면 좌표를 클라이언트 영역 좌표로 변환합니다.

SelectClipPath SelectClipPath 함수는 현재 경로를 디바이스 컨텍스트의 클리핑 영역으로 선택하여 지정된 모드를 사용하여 새 영역을 기존 클리핑 영역과 결합합니다.
SelectClipRgn SelectClipRgn 함수는 지정된 디바이스 컨텍스트에 대한 현재 클리핑 영역으로 영역을 선택합니다.
SelectFont SelectFont 매크로는 지정된 DC(디바이스 컨텍스트)에 글꼴 개체를 선택합니다. 새 글꼴 개체는 이전 글꼴 개체를 대체합니다.
Selectobject SelectObject 함수는 지정된 DC(디바이스 컨텍스트)에 개체를 선택합니다. 새 개체는 동일한 형식의 이전 개체를 대체합니다.
Palette 선택 SelectPalette 함수는 지정된 논리 팔레트를 디바이스 컨텍스트로 선택합니다.
SetArcDirection SetArcDirection은 아크 및 사각형 함수에 사용할 그리기 방향을 설정합니다.
SetBitmapBits SetBitmapBits 함수는 비트맵의 색 데이터 비트를 지정된 값으로 설정합니다.
SetBitmapDimensionEx SetBitmapDimensionEx 함수는 기본 차원을 비트맵에 할당합니다. 이러한 차원은 애플리케이션에서 사용할 수 있습니다. 그러나 시스템에서는 사용되지 않습니다.
SetBkColor SetBkColor 함수는 현재 배경색을 지정된 색 값으로 설정하거나 디바이스가 지정된 색 값을 나타낼 수 없는 경우 가장 가까운 실제 색으로 설정합니다.
SetBkMode SetBkMode 함수는 지정된 디바이스 컨텍스트의 배경 혼합 모드를 설정합니다. 배경 혼합 모드는 실선이 아닌 텍스트, 빗살 무늬 브러시 및 펜 스타일과 함께 사용됩니다.
SetBoundsRect SetBoundsRect 함수는 지정된 디바이스 컨텍스트에 대한 경계 사각형 정보의 누적을 제어합니다.

SetBrushOrgEx SetBrushOrgEx 함수는 GDI가 애플리케이션이 선택한 다음 브러시에 할당하는 브러시 원본을 지정된 디바이스 컨텍스트로 설정합니다.
SetColorAdjustment SetColorAdjustment 함수는 지정된 값을 사용하여 DC(디바이스 컨텍스트)의 색 조정 값을 설정합니다.
SetDCBrushColor SetDCBrushColor 함수는 현재 DC(디바이스 컨텍스트) 브러시 색을 지정된 색 값으로 설정합니다. 디바이스가 지정된 색 값을 나타낼 수 없는 경우 색은 가장 가까운 실제 색으로 설정됩니다.
SetDCPenColor SetDCPenColor 함수는 현재 DC(디바이스 컨텍스트) 펜 색을 지정된 색 값으로 설정합니다. 디바이스가 지정된 색 값을 나타낼 수 없는 경우 색은 가장 가까운 실제 색으로 설정됩니다.
SetDIBColorTable SetDIBColorTable 함수는 현재 지정된 디바이스 컨텍스트로 선택된 DIB의 색 테이블에 있는 항목 범위에서 RGB(빨강, 녹색, 파랑) 색 값을 설정합니다.
SetDIBits SetDIBits 함수는 지정된 DIB에 있는 색 데이터를 사용하여 호환되는 DDB(비트맵)의 픽셀을 설정합니다.
SetDIBitsToDevice SetDIBitsToDevice 함수는 DIB, JPEG 또는 PNG 이미지의 색 데이터를 사용하여 대상 디바이스 컨텍스트와 연결된 디바이스의 지정된 사각형에 있는 픽셀을 설정합니다.
SetEnhMetaFileBits SetEnhMetaFileBits 함수는 지정된 데이터에서 메모리 기반 고급 형식 메타파일을 만듭니다.
SetGraphicsMode SetGraphicsMode 함수는 지정된 디바이스 컨텍스트에 대한 그래픽 모드를 설정합니다.
SetLayout SetLayout 함수는 DC(디바이스 컨텍스트)의 레이아웃을 변경합니다.

SetMapMode SetMapMode 함수는 지정된 디바이스 컨텍스트의 매핑 모드를 설정합니다. 매핑 모드는 페이지 공간 단위를 디바이스 공간 단위로 변환하는 데 사용되는 측정 단위를 정의하고 디바이스의 x축과 y축의 방향도 정의합니다.
SetMapperFlags SetMapperFlags 함수는 논리 글꼴을 실제 글꼴에 매핑할 때 글꼴 매퍼가 사용하는 알고리즘을 변경합니다.
SetMetaFileBitsEx SetMetaFileBitsEx 함수는 제공된 데이터에서 메모리 기반 Windows 형식 메타파일을 만듭니다.
SetMetaRgn SetMetaRgn 함수는 지정된 디바이스 컨텍스트의 현재 클리핑 영역을 현재 메타레지온과 교차하고 결합된 지역을 지정된 디바이스 컨텍스트의 새 메타레지온으로 저장합니다.
SetMiterLimit SetMiterLimit 함수는 지정된 디바이스 컨텍스트에 대한 miter 조인 길이 제한을 설정합니다.
SetPaletteEntries SetPaletteEntries 함수는 논리적 팔레트의 항목 범위에서 RGB(빨강, 녹색, 파랑) 색 값 및 플래그를 설정합니다.
Setpixel SetPixel 함수는 지정된 좌표의 픽셀을 지정된 색으로 설정합니다.
SetPixelV SetPixelV 함수는 지정된 좌표의 픽셀을 지정된 색의 가장 가까운 근사값으로 설정합니다. 지점은 클리핑 영역과 디바이스 표면의 표시되는 부분에 있어야 합니다.
SetPolyFillMode SetPolyFillMode 함수는 다각형을 채우는 함수의 다각형 채우기 모드를 설정합니다.
SetRect SetRect 함수는 지정된 사각형의 좌표를 설정합니다. 이는 RECT 구조체의 적절한 멤버에 왼쪽, 위쪽, 오른쪽 및 아래쪽 인수를 할당하는 것과 같습니다.

SetRectEmpty SetRectEmpty 함수는 모든 좌표가 0으로 설정된 빈 사각형을 만듭니다.
SetRectRgn SetRectRgn 함수는 지정된 좌표를 사용하여 영역을 사각형 영역으로 변환합니다.
SetROP2 SetROP2 함수는 현재 전경 혼합 모드를 설정합니다.
SetStretchBltMode SetStretchBltMode 함수는 지정된 디바이스 컨텍스트에서 비트맵 스트레치 모드를 설정합니다.
SetSystemPaletteUse SetSystemPaletteUse 함수를 사용하면 애플리케이션에서 시스템 팔레트에 2 또는 20개의 정적 색이 포함되어 있는지 여부를 지정할 수 있습니다.
SetTextAlign SetTextAlign 함수는 지정된 디바이스 컨텍스트에 대한 텍스트 맞춤 플래그를 설정합니다.
SetTextCharacterExtra SetTextCharacterExtra 함수는 문자 간 간격을 설정합니다. 시스템에서 텍스트 줄을 작성할 때 구분 문자를 포함하여 각 문자에 문자 간 간격이 추가됩니다.
SetTextColor SetTextColor 함수는 지정된 디바이스 컨텍스트의 텍스트 색을 지정된 색으로 설정합니다.
SetTextJustification SetTextJustification 함수는 시스템에서 텍스트 문자열의 나누기 문자에 추가해야 하는 공간의 양을 지정합니다. 이 공간은 애플리케이션이 TextOut 또는 ExtTextOut 함수를 호출할 때 추가됩니다.
SetViewportExtEx 지정된 값을 사용하여 디바이스 컨텍스트에 대한 뷰포트의 가로 및 세로 익스텐트 를 설정합니다.
SetViewportOrgEx SetViewportOrgEx 함수는 창 원본(0,0)에 매핑되는 디바이스 지점을 지정합니다.

SetWindowExtEx SetWindowExtEx 함수는 지정된 값을 사용하여 디바이스 컨텍스트에 대한 창의 가로 및 세로 익스텐트를 설정합니다.
SetWindowOrgEx SetWindowOrgEx 함수는 뷰포트 원본(0,0)에 매핑되는 창 지점을 지정합니다.
SetWindowRgn SetWindowRgn 함수는 창의 창 영역을 설정합니다.
SetWinMetaFileBits SetWinMetaFileBits 함수는 메타파일을 이전 Windows 형식에서 새로운 향상된 형식으로 변환하고 새 메타파일을 메모리에 저장합니다.
SetWorldTransform SetWorldTransform 함수는 지정된 디바이스 컨텍스트의 월드 공간과 페이지 공간 간에 2차원 선형 변환을 설정합니다. 이 변환은 그래픽 출력의 크기를 조정, 회전, 전단 또는 변환하는 데 사용할 수 있습니다.
Stretchblt StretchBlt 함수는 필요한 경우 원본 사각형에서 대상 사각형의 크기에 맞게 비트맵을 확장하거나 압축하여 비트맵을 대상 사각형으로 복사합니다.
StretchDIBits StretchDIBits 함수는 DIB, JPEG 또는 PNG 이미지의 픽셀 사각형에 대한 색 데이터를 지정된 대상 사각형에 복사합니다.
StrokeAndFillPath StrokeAndFillPath 함수는 경로의 열린 그림을 닫고, 현재 펜을 사용하여 경로의 윤곽선을 스트로크하고, 현재 브러시를 사용하여 내부를 채웁니다.
StrokePath StrokePath 함수는 현재 펜을 사용하여 지정된 경로를 렌더링합니다.
SubtractRect SubtractRect 함수는 한 사각형을 다른 사각형에서 빼서 형성된 사각형의 좌표를 결정합니다.

TabbedTextOutA TabbedTextOut 함수는 지정된 위치에 문자 문자열을 쓰고 탭을 탭 중지 위치 배열에 지정된 값으로 확장합니다. 텍스트는 현재 선택한 글꼴, 배경색 및 텍스트 색으로 작성됩니다. (ANSI)
TabbedTextOutW TabbedTextOut 함수는 지정된 위치에 문자 문자열을 쓰고 탭을 탭 중지 위치 배열에 지정된 값으로 확장합니다. 텍스트는 현재 선택한 글꼴, 배경색 및 텍스트 색으로 작성됩니다. (유니코드)
TextOutA TextOut 함수는 현재 선택한 글꼴, 배경색 및 텍스트 색을 사용하여 지정된 위치에 문자열을 씁니다. (ANSI)
TextOutW TextOut 함수는 현재 선택한 글꼴, 배경색 및 텍스트 색을 사용하여 지정된 위치에 문자열을 씁니다. (유니코드)
TransparentBlt TransparentBlt 함수는 지정된 원본 디바이스 컨텍스트에서 대상 디바이스 컨텍스트로 픽셀 사각형에 해당하는 색 데이터의 비트 블록 전송을 수행합니다.
TTCharToUnicode 8비트 문자 코드 값의 배열을 16비트 유니코드 값으로 변환합니다.
TTDeleteEmbeddedFont 포함된 글꼴 hFontReference에서 사용하는 메모리를 해제합니다.
TTEmbedFont 하위 설정 와이드 문자(16비트) 글꼴을 포함하는 글꼴 구조를 만듭니다. 디바이스 컨텍스트(hDC)의 현재 글꼴은 글꼴 정보를 제공합니다.
TTEmbedFontEx 하위 설정 UCS-4 문자(32비트) 글꼴을 포함하는 글꼴 구조를 만듭니다. 디바이스 컨텍스트(hDC)의 현재 글꼴은 글꼴 정보를 제공합니다.
TTEmbedFontFromFileA 하위 설정 와이드 문자(16비트) 글꼴을 포함하는 글꼴 구조를 만듭니다. 외부 파일은 글꼴 정보를 제공합니다.

TTEnableEmbeddingForFacename 서체 제외 목록에서 얼굴 이름을 추가하거나 제거합니다.
TTGetEmbeddedFontInfo 포함된 글꼴에 대한 정보(예: 포함 권한)를 검색합니다. TTGetEmbeddedFontInfo는 TTLoadEmbeddedFont와 동일한 작업을 수행하지만 포함된 글꼴에 대한 내부 데이터 구조를 할당하지는 않습니다.
TTGetEmbeddingType 글꼴의 포함 권한을 가져옵니다.
TTGetNewFontName TTLoadEmbeddedFont를 통해 로드된 글꼴의 패밀리 이름을 가져옵니다.
TTIsEmbeddingEnabled 서체 제외 목록에 지정된 글꼴이 포함되어 있는지 여부를 확인합니다.
TTIsEmbeddingEnabledForFacename 지정된 글꼴에 포함을 사용할 수 있는지 여부를 결정합니다.
TTLoadEmbeddedFont 문서 스트림에서 포함된 글꼴을 읽고 설치합니다. 또한 클라이언트가 글꼴의 포함 권한을 추가로 제한할 수 있습니다.
TTRunValidationTests 지정된 크기 범위에서 와이드 문자(16비트) 글꼴의 일부 또는 모든 문자 모양 데이터의 유효성을 검사합니다.
TTRunValidationTestsEx 지정된 크기 범위에서 UCS-4 문자(32비트) 글꼴의 일부 또는 모든 문자 모양 데이터의 유효성을 검사합니다.
UnionRect UnionRect 함수는 두 사각형의 공용 구조체를 만듭니다. 공용 구조체는 두 원본 사각형을 모두 포함하는 가장 작은 사각형입니다.
UnrealizeObject UnrealizeObject 함수는 브러시의 원본을 다시 설정하거나 논리 팔레트를 다시 설정합니다.

UpdateColors UpdateColors 함수는 클라이언트 영역의 현재 색을 현재 실현된 논리 팔레트에 다시 매핑하여 지정된 디바이스 컨텍스트의 클라이언트 영역을 업데이트합니다.
UpdateWindow UpdateWindow 함수는 창의 업데이트 영역이 비어 있지 않은 경우 창에 WM_PAINT 메시지를 보내 지정된 창의 클라이언트 영역을 업데이트합니다.
ValidateRect ValidateRect 함수는 지정된 창의 업데이트 영역에서 사각형을 제거하여 사각형 내의 클라이언트 영역의 유효성을 검사합니다.
ValidateRgn ValidateRgn 함수는 지정된 창의 현재 업데이트 영역에서 지역을 제거하여 지역 내의 클라이언트 영역의 유효성을 검사합니다.
WidenPath WidenPath 함수는 현재 선택한 펜을 사용하여 경로가 지정된 디바이스 컨텍스트로 스트로크된 경우 그릴 영역으로 현재 경로를 다시 정의합니다.
WindowFromDC WindowFromDC 함수는 지정된 DC(디스플레이 디바이스 컨텍스트)와 연결된 창에 대한 핸들을 반환합니다. 지정된 디바이스 컨텍스트를 사용하는 출력 함수는 이 창에 그립니다.

구조체

ABC ABC 구조체에는 TrueType 글꼴의 문자 너비가 포함됩니다.
ABCFLOAT ABCFLOAT 구조체에는 글꼴 문자의 A, B 및 C 너비가 포함됩니다.
AXESLISTA AXESLIST 구조체에는 여러 마스터 글꼴의 모든 축에 대한 정보가 포함됩니다. (ANSI)

AXESLISTW AXESLIST 구조체에는 여러 마스터 글꼴의 모든 축에 대한 정보가 포함됩니다. (유니코드)
AXISINFOA AXISINFO 구조체에는 여러 마스터 글꼴의 축에 대한 정보가 포함되어 있습니다. (ANSI)
AXISINFOW AXISINFO 구조체에는 여러 마스터 글꼴의 축에 대한 정보가 포함되어 있습니다. (유니코드)
비트맵 BITMAP 구조체는 비트맵의 형식, 너비, 높이, 색 형식 및 비트 값을 정의합니다.
BITMAPCOREHEADER BITMAPCOREHEADER 구조체에는 DIB의 차원 및 색 형식에 대한 정보가 포함되어 있습니다.
BITMAPCOREINFO BITMAPCOREINFO 구조체는 DIB의 차원 및 색 정보를 정의합니다.
BITMAPFILEHEADER BITMAPFILEHEADER 구조체에는 DIB가 포함된 파일의 형식, 크기 및 레이아웃에 대한 정보가 포함되어 있습니다.
BITMAPINFO BITMAPINFO 구조체는 DIB의 차원 및 색 정보를 정의합니다.
BITMAPINFOHEADER BITMAPINFOHEADER 구조체에는 디바이스 독립적 비트맵(DIB)의 차원 및 색 형식에 대한 정보가 포함되어 있습니다.
BITMAPV4HEADER BITMAPV4HEADER 구조체는 비트맵 정보 헤더 파일입니다. BITMAPINFOHEADER 구조체의 확장 버전입니다. 애플리케이션은 추가 기능에 BITMAPV5HEADER 구조를 사용할 수 있습니다.
BITMAPV5HEADER BITMAPV5HEADER 구조체는 비트맵 정보 헤더 파일입니다. BITMAPINFOHEADER 구조체의 확장 버전입니다.

BLENDFUNCTION BLENDFUNCTION 구조체는 원본 및 대상 비트맵에 대한 혼합 함수를 지정하여 혼합을 제어합니다.
COLORADJUSTMENT COLORADJUSTMENT 구조체는 스트레치 모드가 HALFTONE일 때 StretchBlt 및 StretchDIBits 함수에서 사용하는 색 조정 값을 정의합니다. SetColorAdjustment 함수를 호출하여 색 조정 값을 설정할 수 있습니다.
DESIGNVECTOR DESIGNVECTOR 구조체는 애플리케이션에서 여러 마스터 글꼴의 축 값을 지정하는 데 사용됩니다.
DIBSECTION DIBSECTION 구조에는 CreateDIBSection 함수를 호출하여 만든 DIB에 대한 정보가 포함됩니다.
DISPLAY_DEVICEA DISPLAY_DEVICE 구조체는 EnumDisplayDevices 함수의 iDevNum 매개 변수로 지정된 디스플레이 디바이스에 대한 정보를 받습니다. (ANSI)
DISPLAY_DEVICEW DISPLAY_DEVICE 구조체는 EnumDisplayDevices 함수의 iDevNum 매개 변수로 지정된 디스플레이 디바이스에 대한 정보를 받습니다. (유니코드)
DRAWTEXTPARAMS DRAWTEXTPARAMS 구조체에는 DrawTextEx 함수에 대한 확장 서식 옵션이 포함되어 있습니다.
EMR EMR 구조체는 모든 향상된 메타파일 레코드에 대한 기본 구조를 제공합니다. 향상된 메타파일 레코드에는 향상된 형식 메타파일로 그림의 일부를 만드는 데 사용되는 특정 GDI 함수에 대한 매개 변수가 포함되어 있습니다.
EMRABORTPATH AbortPath, BeginPath, EndPath, CloseFigure, FlattenPath, WidenPath, SetMetaRgn, SaveDC 및 RealizePalette 향상된 메타파일 레코드에 대한 데이터를 포함합니다.
EMRALPHABLEND EMRALPHABLEND 구조체에는 AlphaBlend 고급 메타파일 레코드에 대한 멤버가 포함됩니다.

EMRANGLEARC EMRANGLEARC 구조체에는 AngleArc 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRARC EMRARC, EMRARCTO, EMRCHORD 및 EMRPIE 구조에는 Arc, ArcTo, 코드 및 파이 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRBITBLT EMRBITBLT 구조체에는 BitBlt 향상된 메타파일 레코드에 대한 멤버가 포함됩니다. GDI(그래픽 디바이스 인터페이스)는 메타파일 레코드에 저장하기 전에 디바이스 종속 비트맵을 디바이스 독립적 DIB(비트맵)로 변환합니다.
EMRCOLORCORRECTPALETTE EMRCOLORCORRECTPALETTE 구조체에는 ColorCorrectPalette 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRCOLORMATCHTOTARGET EMRCOLORMATCHTOTARGET 구조체에는 ColorMatchToTarget 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRCREATEBRUSHINDIRECT EMRCREATEBRUSHINDIRECT 구조체에는 CreateBrushIndirect 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRCREATECOLORSPACE EMRCREATECOLORSPACE 구조체에는 CreateColorSpace 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRCREATECOLORSPACEW EMRCREATECOLORSPACEW 구조체에는 CreateColorSpace 향상된 메타파일 레코드에 대한 멤버가 포함됩니다. 유니코드 논리 색 공간이 있고 원시 원본 프로필 데이터를 포함하는 선택적 배열도 있다는 점에서 EMRCREATECOLORSPACE와 다릅니다.
EMRCREATEDIBPATTERNBRUSHPT EMRCREATEDIBPATTERNBRUSHPT 구조체에는 CreateDIBPatternBrushPt 향상된 메타파일 레코드에 대한 멤버가 포함됩니다. BITMAPINFO 구조 뒤에는 압축된 DIB(디바이스 독립적 비트맵)를 형성하는 비트맵 비트가 옵니다.

EMRCREATEMONOBRUSH EMRCREATEMONOBRUSH 구조체에는 CreatePatternBrush(흑백 비트맵을 전달한 경우) 또는 CreateDIBPatternBrush(단색 DIB를 전달한 경우) 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRCREATEPALETTE EMRCREATEPALETTE 구조체에는 CreatePalette 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRCREATEPEN EMRCREATEPEN 구조에는 CreatePen 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRELLIPSE EMRELLIPSE 및 EMRRECTANGLE 구조에는 타원 및 Rectangle 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMREOF EMREOF 구조에는 메타파일의 끝을 나타내는 향상된 메타파일 레코드에 대한 데이터가 포함되어 있습니다.
EMREXCLUDECLIPRECT EMREXCLUDECLIPRECT 및 EMRINTERSECTCLIPRECT 구조에는 ExcludeClipRect 및 IntersectClipRect 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMREXTCREATEFONTINDIRECTW EMREXTCREATEFONTINDIRECTW 구조체에는 CreateFontIndirect 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMREXTCREATEPEN EMREXTCREATEPEN 구조체에는 ExtCreatePen 향상된 메타파일 레코드에 대한 멤버가 포함됩니다. 레코드에 BITMAPINFO 구조체가 포함된 경우 압축된 DIB(디바이스 독립적 비트맵)를 구성하는 비트맵 비트가 뒤따릅니다.
EMREXTFLOODFILL EMREXTFLOODFILL 구조체에는 ExtFloodFill 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMREXTSELECTCLIPRGN EMREXTSELECTCLIPRGN 구조체에는 ExtSelectClipRgn 고급 메타파일 레코드에 대한 멤버가 포함됩니다.

EMREXTTEXTOUTA EMREXTTEXTOUTA 및 EMREXTTEXTOUTW 구조체에는 ExtTextOut, TextOut 또는 DrawText 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRFILLPATH EMRFILLPATH, EMRSTROKEANDFILLPATH 및 EMRSTROKEPATH 구조에는 FillPath, StrokeAndFillPath 및 StrokePath 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRFILLRGN EMRFILLRGN 구조체에는 FillRgn 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRFORMAT EMRFORMAT 구조에는 향상된 메타파일의 그래픽 데이터를 식별하는 정보가 포함되어 있습니다. GDICOMMENT_MULTIFORMATS 향상된 메타파일 공개 주석에는 EMRFORMAT 구조체의 배열이 포함되어 있습니다.
EMRFRAMERGN EMRFRAMERGN 구조체에는 FrameRgn 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRGDICOMMENT EMRGDICOMMENT 구조에는 애플리케이션별 데이터가 포함됩니다.
EMRGLSBOUNDEDRECORD EMRGLSBOUNDEDRECORD 구조체에는 OpenGL 함수에서 생성된 향상된 메타파일 레코드에 대한 멤버가 포함되어 있습니다. 메타파일을 재생할 때 크기를 조정해야 하는 픽셀 단위의 정보가 포함된 OpenGL 함수에 대한 데이터가 포함됩니다.
EMRGLSRECORD EMRGLSRECORD 구조체에는 OpenGL 함수에서 생성된 향상된 메타파일 레코드에 대한 멤버가 포함되어 있습니다. OpenGL 뷰포트로 자동으로 스케일링되는 OpenGL 함수에 대한 데이터가 포함되어 있습니다.
EMRGRADIENTFILL EMRGRADIENTFILL 구조체에는 GradientFill 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRINVERTRGN EMRINVERTRGN 및 EMRPAINTRGN 구조에는 InvertRgn 및 PaintRgn 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.

EMRLINETO EMRLINETO 및 EMRMOVETOEX 구조체에는 LineTo 및 MoveToEx 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRMASKBLT EMRMASKBLT 구조체에는 MaskBlt 향상된 메타파일 레코드에 대한 멤버가 포함됩니다. GDI(그래픽 디바이스 인터페이스)는 메타파일 레코드에 저장하기 전에 디바이스 종속 비트맵을 디바이스 독립적 DIB(비트맵)로 변환합니다.
EMRMODIFYWORLDTRANSFORM EMRMODIFYWORLDTRANSFORM 구조체에는 ModifyWorldTransform 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMROFFSETCLIPRGN EMROFFSETCLIPRGN 구조체에는 OffsetClipRgn 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRPIXELFORMAT EMRPIXELFORMAT 구조체에는 SetPixelFormat 향상된 메타파일 레코드에 대한 멤버가 포함됩니다. ENHMETAHEADER의 픽셀 형식 정보는 이 구조를 나타냅니다.
EMRPLGBLT EMRPLGBLT 구조체에는 PlgBlt 향상된 메타파일 레코드에 대한 멤버가 포함됩니다. GDI(그래픽 디바이스 인터페이스)는 메타파일 레코드에 저장하기 전에 디바이스 종속 비트맵을 디바이스 독립적 DIB(비트맵)로 변환합니다.
EMRPOLYDRAW EMRPOLYDRAW 구조체에는 PolyDraw 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRPOLYDRAW16 EMRPOLYDRAW16 구조체에는 PolyDraw 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRPOLYLINE EMRPOLYLINE, EMRPOLYBEZIER, EMRPOLYGON, EMRPOLYBEZIERTO 및 EMRPOLYLINETO 구조체에는 Polyline, PolyBezier, Polygon, PolyBezierTo 및 PolylineTo 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.

EMRPOLYLINE16 EMRPOLYLINE16, EMRPOLYBEZIER16, EMRPOLYGON16, EMRPOLYBEZIERTO16 및 EMRPOLYLINETO16 구조체에는 Polyline, PolyBezier, Polygon, PolyBezierTo 및 PolylineTo 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRPOLYPOLYLINE EMRPOLYPOLYLINE 및 EMRPOLYPOLYGON 구조체에는 PolyPolyline 및 PolyPolygon 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRPOLYPOLYLINE16 EMRPOLYPOLYLINE16 및 EMRPOLYPOLYGON16 구조체에는 PolyPolyline 및 PolyPolygon 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRPOLYTEXTOUTA EMRPOLYTEXTOUTA 및 EMRPOLYTEXTOUTW 구조체에는 PolyTextOut 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRRESIZEPALETTE EMRRESIZEPALETTE 구조체에는 ResizePalette 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRRESTOREDC EMRRESTOREDC 구조체에는 RestoreDC 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRROUNDRECT EMRROUNDRECT 구조체에는 RoundRect 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSCALEVIEWPORTEXTEX EMRSCALEVIEWPORTEXTEX 및 EMRSCALEWINDOWEXTEX 구조체에는 ScaleViewportExtExtEx 및 ScaleWindowExtEx 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSELECTCLIPPATH SelectClipPath, SetBkMode, SetMapMode, SetPolyFillMode, SetROP2, SetStretchBltMode, SetTextAlign, SetICMMode 및 SetLayout 향상된 메타파일 레코드에 대한 매개 변수를 포함합니다.
EMRSELECTOBJECT EMRSELECTOBJECT 및 EMRDELETEOBJECT 구조체에는 SelectObject 및 DeleteObject 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.

EMRSELECTPALETTE EMRSELECTPALETTE 구조체에는 SelectPalette 향상된 메타파일 레코드에 대한 멤버가 포함됩니다. SelectPalette의 bForceBackground 매개 변수는 항상 TRUE로 기록되므로 색상표가 배경 팔레트로 실현됩니다.
EMRSETARCDIRECTION EMRSETARCDIRECTION 구조체에는 SetArcDirection 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSETBKCOLOR EMRSETBKCOLOR 및 EMRSETTEXTCOLOR 구조체에는 SetBkColor 및 SetTextColor 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSETCOLORADJUSTMENT EMRSETCOLORADJUSTMENT 구조체에는 SetColorAdjustment 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSETCOLORSPACE EMRSETCOLORSPACE, EMRSELECTCOLORSPACE 및 EMRDELETECOLORSPACE 구조체에는 SetColorSpace 및 DeleteColorSpace 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSETDIBITSTODEVICE EMRSETDIBITSTODEVICE 구조체에는 SetDIBitsToDevice 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSETICMPROFILE EMRSETICMPROFILE 구조체에는 SetICMProfile 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSETMAPPERFLAGS EMRSETMAPPERFLAGS 구조체에는 SetMapperFlags 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSETMITERLIMIT EMRSETMITERLIMIT 구조체에는 SetMiterLimit 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSETPALETTEENTRIES EMRSETPALETTEENTRIES 구조체에는 SetPaletteEntries 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.

EMRSETPIXELV EMRSETPIXELV 구조체에는 SetPixelV 향상된 메타파일 레코드에 대한 멤버가 포함됩니다. 향상된 메타파일을 만들면 SetPixel에 대한 호출도 이 레코드에 기록됩니다.
EMRSETVIEWPORTEXTEX EMRSETVIEWPORTEXTEX 및 EMRSETWINDOWEXTEX 구조체에는 SetViewportExtEx 및 SetWindowExtEx 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSETVIEWPORTORGEX EMRSETVIEWPORTORGEX, EMRSETWINDOWORGEX 및 EMRSETBRUSHORGEX 구조체에는 SetViewportOrgEx, SetWindowOrgEx 및 SetBrushOrgEx 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSETWORLDTRANSFORM EMRSETWORLDTRANSFORM 구조체에는 SetWorldTransform 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSTRETCHBLT EMRSTRETCHBLT 구조체에는 StretchBlt 향상된 메타파일 레코드에 대한 멤버가 포함됩니다. GDI(그래픽 디바이스 인터페이스)는 메타파일 레코드에 저장하기 전에 디바이스 종속 비트맵을 DIB(디바이스 독립적 비트맵)로 변환합니다.
EMRSTRETCHDIBITS EMRSTRETCHDIBITS 구조체에는 StretchDIBits 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRTEXT EMRTEXT 구조체에는 텍스트 출력에 대한 멤버가 포함됩니다.
EMRTRANSPARENTBLT EMRTRANSPARENTBLT 구조체에는 TransparentBLT 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
ENHMETAHEADER ENHMETAHEADER 구조에는 향상된 메타파일에 저장된 그림의 크기, 향상된 메타파일의 레코드 수, 그림을 만든 디바이스의 해상도 등과 같은 향상된 메타파일 데이터가 포함됩니다. 이 구조체는 항상 향상된 메타파일의 첫 번째 레코드입니다.

ENHMETARECORD ENHMETARECORD 구조체에는 고급 형식 메타파일로 그림의 일부를 만드는 데 사용되는 GDI(그래픽 디바이스 인터페이스) 함수를 설명하는 데이터가 포함되어 있습니다.
ENUMLOGFONTA ENUMLOGFONT 구조체는 글꼴의 특성, 글꼴의 전체 이름 및 글꼴 스타일을 정의합니다. (ANSI)
ENUMLOGFONTEXA ENUMLOGFONTEX 구조체에는 열거형 글꼴에 대한 정보가 포함되어 있습니다. (ANSI)
ENUMLOGFONTEXDVA ENUMLOGFONTEXDV 구조체에는 글꼴을 만드는 데 사용되는 정보가 포함됩니다. (ANSI)
ENUMLOGFONTEXDVW ENUMLOGFONTEXDV 구조체에는 글꼴을 만드는 데 사용되는 정보가 포함됩니다. (유니코드)
ENUMLOGFONTEXW ENUMLOGFONTEX 구조체에는 열거형 글꼴에 대한 정보가 포함되어 있습니다. (유니코드)
ENUMLOGFONTW ENUMLOGFONT 구조체는 글꼴의 특성, 글꼴의 전체 이름 및 글꼴 스타일을 정의합니다. (유니코드)
ENUMTEXTMETRICA ENUMTEXTMETRIC 구조체에는 실제 글꼴에 대한 정보가 포함됩니다. (ANSI)
ENUMTEXTMETRICW ENUMTEXTMETRIC 구조체에는 실제 글꼴에 대한 정보가 포함됩니다. (유니코드)
EXTLOGFONTA EXTLOGFONT 구조체는 글꼴의 특성을 정의합니다. (ANSI)
EXTLOGFONTW EXTLOGFONT 구조체는 글꼴의 특성을 정의합니다. (유니코드)
EXTLOGPEN EXTLOGPEN 구조체는 확장된 펜의 펜 스타일, 너비 및 브러시 특성을 정의합니다.

FIXED FIXED 구조체에는 고정 소수점 실수의 정수 부분과 소수 부분이 포함됩니다.
GCP_RESULTSA GCP_RESULTS 구조체에는 문자열의 문자에 대한 정보가 포함됩니다. 이 구조체는 GetCharacterPlacement 함수의 결과를 받습니다. 일부 언어의 경우 배열의 첫 번째 요소에는 더 많은 언어 종속 정보가 포함될 수 있습니다. (ANSI)
GCP_RESULTSW GCP_RESULTS 구조체에는 문자열의 문자에 대한 정보가 포함됩니다. 이 구조체는 GetCharacterPlacement 함수의 결과를 받습니다. 일부 언어의 경우 배열의 첫 번째 요소에는 더 많은 언어 종속 정보가 포함될 수 있습니다. (유니코드)
문자 모양 GLYPHMETRICS 구조체에는 문자 셀의 문자 모양 배치 및 방향에 대한 정보가 포함되어 있습니다.
GLYPHSET GLYPHSET 구조체에는 유니코드 코드 포인트 범위에 대한 정보가 포함됩니다.
GRADIENT_RECT GRADIENT_RECT 구조체는 GradientFill 함수의 pVertex 배열에 있는 두 꼭짓점의 인덱스를 지정합니다. 이 두 꼭짓점은 사각형의 왼쪽 위와 오른쪽 아래 경계를 형성합니다.
GRADIENT_TRIANGLE GRADIENT_TRIANGLE 구조체는 GradientFill 함수의 pVertex 배열에 있는 세 꼭짓점의 인덱스를 지정합니다. 이 세 꼭짓점은 하나의 삼각형을 형성합니다.
HANDLETABLE HANDLETABLE 구조체는 각각 GDI(그래픽 디바이스 인터페이스) 개체를 식별하는 핸들 배열입니다.
KERNINGPAIR KERNINGPAIR 구조체는 커닝 쌍을 정의합니다.
LOGBRUSH LOGBRUSH 구조체는 실제 브러시의 스타일, 색 및 패턴을 정의합니다. CreateBrushIndirect 및 ExtCreatePen 함수에서 사용됩니다.

LOGBRUSH32 LOGBRUSH32 구조체는 실제 브러시의 스타일, 색 및 패턴을 정의합니다.
LOGFONTA LOGFONT 구조체는 글꼴의 특성을 정의합니다. (ANSI)
LOGFONTW LOGFONT 구조체는 글꼴의 특성을 정의합니다. (유니코드)
LOGPALETTE LOGPALETTE 구조체는 논리 팔레트를 정의합니다.
LOGPEN LOGPEN 구조체는 펜의 스타일, 너비 및 색을 정의합니다. CreatePenIndirect 함수는 LOGPEN 구조를 사용합니다.
MAT2 MAT2 구조체에는 GetGlyphOutline 함수에서 사용하는 변환 행렬의 값이 포함됩니다.
메타헤더 METAHEADER 구조체에는 Windows 형식 메타파일에 대한 정보가 포함되어 있습니다.
METARECORD METARECORD 구조체에는 Windows 형식 메타파일 레코드가 포함되어 있습니다.
MONITORINFO MONITORINFO 구조에는 디스플레이 모니터에 대한 정보가 포함되어 있습니다. GetMonitorInfo 함수는 MONITORINFO 구조체 또는 MONITORINFOEX 구조체에 정보를 저장합니다. MONITORINFO 구조체는 MONITORINFOEX 구조체의 하위 집합입니다.
MONITORINFOEXA MONITORINFOEX 구조체에는 디스플레이 모니터에 대한 정보가 포함되어 있습니다. GetMonitorInfo 함수는 정보를 MONITORINFOEX 구조체 또는 MONITORINFO 구조체에 저장합니다. MONITORINFOEX 구조체는 MONITORINFO 구조체의 상위 집합입니다. (ANSI)

MONITORINFOEXW MONITORINFOEX 구조체에는 디스플레이 모니터에 대한 정보가 포함되어 있습니다. GetMonitorInfo 함수는 정보를 MONITORINFOEX 구조체 또는 MONITORINFO 구조체에 저장합니다. MONITORINFOEX 구조체는 MONITORINFO 구조체의 상위 집합입니다. (유니코드)
NEWTEXTMETRICA NEWTEXTMETRIC 구조에는 실제 글꼴을 설명하는 데이터가 포함되어 있습니다. (ANSI)
NEWTEXTMETRICEXA NEWTEXTMETRICEX 구조체에는 실제 글꼴에 대한 정보가 포함됩니다. (ANSI)
NEWTEXTMETRICEXW NEWTEXTMETRICEX 구조체에는 실제 글꼴에 대한 정보가 포함됩니다. (유니코드)
NEWTEXTMETRICW NEWTEXTMETRIC 구조에는 실제 글꼴을 설명하는 데이터가 포함되어 있습니다. (유니코드)
OUTLINETEXTMETRICA OUTLINETEXTMETRIC 구조에는 TrueType 글꼴을 설명하는 메트릭이 포함되어 있습니다. (ANSI)
OUTLINETEXTMETRICW OUTLINETEXTMETRIC 구조에는 TrueType 글꼴을 설명하는 메트릭이 포함되어 있습니다. (유니코드)
PAINTSTRUCT PAINTSTRUCT 구조체에는 애플리케이션에 대한 정보가 포함됩니다. 이 정보는 해당 애플리케이션이 소유한 창의 클라이언트 영역을 그리는 데 사용할 수 있습니다.
PANOSE PANOSE 구조체는 TrueType 글꼴에 대한 PANOSE 글꼴 분류 값을 설명합니다. 그런 다음 이러한 특성을 사용하여 글꼴을 비슷한 모양이지만 다른 이름의 다른 글꼴과 연결합니다.
POINTFX POINTFX 구조체에는 TrueType 글꼴의 문자 윤곽선을 설명하는 점의 좌표가 포함되어 있습니다.
POLYTEXTA POLYTEXT 구조체는 PolyTextOut 함수가 텍스트 문자열을 그리는 방법을 설명합니다. (ANSI)

POLYTEXTW POLYTEXT 구조체는 PolyTextOut 함수가 텍스트 문자열을 그리는 방법을 설명합니다. (유니코드)
RASTERIZER_STATUS RASTERIZER_STATUS 구조체에는 TrueType이 설치되어 있는지 여부에 대한 정보가 포함되어 있습니다. 이 구조는 애플리케이션이 GetRasterizerCaps 함수를 호출할 때 채워집니다.
RGBQUAD RGBQUAD 구조체는 빨간색, 녹색 및 파란색의 상대적 강도로 구성된 색을 설명합니다.
RGBTRIPLE RGBTRIPLE 구조체는 빨간색, 녹색 및 파란색의 상대적 강도로 구성된 색을 설명합니다. BITMAPCOREINFO 구조체의 bmciColors 멤버는 RGBTRIPLE 구조체의 배열로 구성됩니다.
RGNDATA RGNDATA 구조체에는 영역을 구성하는 머리글 및 사각형 배열이 포함됩니다. 사각형은 위쪽에서 아래쪽, 왼쪽에서 오른쪽으로 정렬됩니다. 겹치지 않습니다.
RGNDATAHEADER RGNDATAHEADER 구조체는 GetRegionData 함수에서 반환된 데이터를 설명합니다.
TEXTMETRICA TEXTMETRIC 구조에는 실제 글꼴에 대한 기본 정보가 포함되어 있습니다. 모든 크기는 논리 단위로 지정됩니다. 즉, 표시 컨텍스트의 현재 매핑 모드에 따라 달라집니다. (ANSI)
TEXTMETRICW TEXTMETRIC 구조에는 실제 글꼴에 대한 기본 정보가 포함되어 있습니다. 모든 크기는 논리 단위로 지정됩니다. 즉, 표시 컨텍스트의 현재 매핑 모드에 따라 달라집니다. (유니코드)
TRIVERTEX TRIVERTEX 구조체에는 색 정보 및 위치 정보가 포함됩니다.
TTEMBEDINFO TTEMBEDINFO 구조체에는 포함된 글꼴 개체를 합법적으로 참조할 수 있는 URL 목록이 포함되어 있습니다.
TTLOADINFO TTLOADINFO 구조체에는 포함된 글꼴 개체를 가져온 URL이 포함됩니다.

TTPOLYCURVE TTPOLYCURVE 구조체에는 TrueType 문자의 윤곽선에 있는 곡선에 대한 정보가 포함되어 있습니다.
TTPOLYGONHEADER TTPOLYGONHEADER 구조체는 TrueType 문자 윤곽선에서 윤곽선의 시작 위치와 형식을 지정합니다.
TTVALIDATIONTESTSPARAMS TTVALIDATIONTESTSPARAMS 구조체에는 Microsoft OpenType 글꼴을 테스트하기 위한 매개 변수가 포함되어 있습니다.
TTVALIDATIONTESTSPARAMSEX TTVALIDATIONTESTSPARAMSEX 구조체에는 Microsoft OpenType 글꼴을 테스트하기 위한 매개 변수가 포함되어 있습니다.
WCRANGE WCRANGE 구조체는 유니코드 문자의 범위를 지정합니다.
Xform XFORM 구조체는 페이지 공간 변환에 대한 세계 공간을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Windows HTTP 서비스(WinHTTP)

아티클2023. 08. 22.

WinHTTP(Windows HTTP 서비스) 기술 개요입니다.

WinHTTP(Windows HTTP 서비스)를 개발하려면 다음 헤더가 필요합니다.

- winhttp.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows HTTP 서비스(WinHTTP)

구조체

WINHTTP_SECURITY_INFO WINHTTP_SECURITY_INFO 구조에는 HTTP 요청에 대한 다양한 타이밍 정보가 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Windows 이미징 구성 요소

Windows 이미징 구성 요소 기술 개요입니다.

Windows 이미징 구성 요소를 개발하려면 다음 헤더가 필요합니다.

- wincodec.h
- wincodecsdk.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 이미징 구성 요소

Enumerations

테이블 확장

WIC8BIMIptcDigestProperties 8BIM IPTC 다이제스트 메타데이터 블록에 있는 메타데이터 항목의 식별자를 지정합니다.
WIC8BIMIptcProperties 8BIM IPTC 블록에 있는 메타데이터 항목의 식별자를 지정합니다.
WIC8BIMResolutionInfoProperties 8BIMResolutionInfo 블록에 있는 메타데이터 항목의 식별자를 지정합니다.
WICBitmapAlphaChannelOption 원하는 알파 채널 사용을 지정합니다.
WICBitmapChainType 체인 유형을 지정하는 상수를 정의합니다.
WICBitmapCreateCacheOption 원하는 캐시 사용을 지정합니다.
WICBitmapDecoderCapabilities 디코더의 기능을 지정합니다.
WICBitmapDitherType

이미지 형식 간에 변환할 때 적용할 디더 알고리즘의 형식을 지정합니다.
WICBitmapEncoderCacheOption 인코더에 사용할 수 있는 캐시 옵션을 지정합니다.
WICBitmapInterpolationMode 이미지 크기를 조정하는 데 사용할 샘플링 또는 필터링 모드를 지정합니다.
WICBitmapLockFlags IWICBitmap에 대한 액세스를 지정합니다.
WICBitmapPaletteType 인덱싱된 이미지 형식에 사용되는 색상표의 유형을 지정합니다.
WICBitmapTransformOptions 대칭 이동 및 회전 변환을 지정합니다.
WICColorContextType 색 컨텍스트 형식을 지정합니다.
WICComponentEnumerateOptions 구성 요소 열거 옵션을 지정합니다.
WICComponentSigning 구성 요소 서명 상태를 지정합니다.
WICComponentType WIC(Windows 이미징 구성 요소) 구성 요소의 유형을 지정합니다.
WICDdsAlphaMode DDS 이미지에 포함된 픽셀 색 구성 요소 값의 의미를 지정합니다.
WICDdsDimension DDS 이미지에 포함된 데이터의 차원 형식을 지정합니다.
WICDecodeOptions 디코딩 옵션을 지정합니다.

WICGifApplicationExtensionProperties GIF(그래픽 교환 형식) 이미지의 애플리케이션 확장 메타데이터 속성을 지정합니다.
WICGifCommentExtensionProperties GIF(그래픽 교환 형식) 이미지의 주석 확장 메타데이터 속성을 지정합니다.
WICGifGraphicControlExtensionProperties GIF(그래픽 교환 형식) 이미지에 대한 각 프레임 애니메이션 간의 전환을 정의하는 그래픽 컨트롤 확장 메타데이터 속성을 지정합니다.
WICGifImageDescriptorProperties GIF(그래픽 교환 형식) 프레임에 대한 이미지 설명자 메타데이터 속성을 지정합니다.
WICGifLogicalScreenDescriptorProperties GIF(그래픽 교환 형식) 메타데이터의 논리적 화면 설명자 속성을 지정합니다.
WICHeifCompressionOption HEIF(고효율 이미지 형식) 압축 옵션을 지정하는 상수를 정의합니다.
WICHeifHdrProperties HEIF(고효율 이미지 형식) 이미지의 HDR 속성을 지정합니다.
WICHeifProperties HEIF(고효율 이미지 형식) 이미지의 속성을 지정합니다.
WICJpegChrominanceProperties JPEG 크로미낸스 테이블 속성을 지정합니다.
WICJpegCommentProperties JPEG 주석 속성을 지정합니다.
WICJpegIndexingOptions JPEG 이미지를 인덱싱하는 옵션을 지정합니다.
WICJpegLuminanceProperties JPEG 광도 테이블 속성을 지정합니다.

WICJpegScanType JPEG 이미지 검사에서 픽셀 데이터의 메모리 레이아웃을 지정합니다.
WICJpegTransferMatrix Y'Cb'Cr'에서 R'G'B'로의 변환 매트릭스를 지정합니다.
WICJpegYCrCbSubsamplingOption JPEG YCrCB 하위 샘플링 옵션을 지정합니다.
WICMetadataCreationOptions 메타데이터 만들기 옵션을 지정합니다.
WICNamedWhitePoint 원시 이미지의 명명된 흰색 균형을 지정합니다.
WICPersistOptions 스트림을 사용하여 구성 요소를 초기화할 때 사용되는 WIC(Windows 이미징 구성 요소) 옵션을 지정합니다.
WICPixelFormatNumericRepresentation WIC 픽셀 형식의 숫자 표현에 대한 기본 형식을 지정하는 상수를 정의합니다.
WICPlanarOptions IWICPlanarBitmapSourceTransform 구현에 대한 추가 옵션을 지정합니다.
WICPngBkgdProperties PNG(이식 가능한 네트워크 그래픽) 배경(bKGD) 청크 메타데이터 속성을 지정합니다.
WICPngChrmProperties CIE XYZ 색에 대한 PNG(이식 가능한 네트워크 그래픽) cHRM 청크 메타데이터 속성을 지정합니다.
WICPngFilterOption 압축 최적화에 사용할 수 있는 PNG(이식 가능한 네트워크 그래픽) 필터를 지정합니다.
WICPngGamaProperties PNG(이식 가능한 네트워크 그래픽) gAMA 청크 메타데이터 속성을 지정합니다.

WICPngHistProperties PNG(이식 가능한 네트워크 그래픽) hIST 청크 메타데이터 속성을 지정합니다.
WICPngIccpProperties PNG(이식 가능한 네트워크 그래픽) iCCP 청크 메타데이터 속성을 지정합니다.
WICPngItxtProperties PNG(이식 가능한 네트워크 그래픽) iTXT 청크 메타데이터 속성을 지정합니다.
WICPngSrgbProperties PNG(이식 가능한 네트워크 그래픽) sRGB 청크 메타데이터 속성을 지정합니다.
WICPngTimeProperties PNG(이식 가능한 네트워크 그래픽) tIME 청크 메타데이터 속성을 지정합니다.
WICProgressNotification 진행률 알림 콜백을 호출해야 하는 시기를 지정합니다.
WICProgressOperation 알림을 받을 진행률 작업을 지정합니다.
WICRawCapabilities 원시 이미지의 기능 지원을 지정합니다.
WICRawParameterSet 원시 코덱에서 사용하는 매개 변수 집합을 지정합니다.
WICRawRenderMode 다음 CopyPixels 호출의 렌더링 의도를 지정합니다.
WICRawRotationCapabilities 코덱의 회전 기능을 지정합니다.
WICSectionAccessLevel Windows GDI(그래픽 디바이스 인터페이스) 섹션의 액세스 수준을 지정합니다.
WICTiffCompressionOption

TIFF(태그가 지정된 이미지 파일 형식) 압축 옵션을 지정합니다.
WICWebpAnimProperties WebP 이미지의 애니메이션 속성을 지정합니다.
WICWebpAnmfProperties WebP 이미지의 애니메이션 프레임 속성을 지정합니다.

Functions

테이블 확장

AddWriter IWICMetadataWriter를 추가하여 최상위 메타데이터 블록을 추가합니다.
AppendFrameToChain 지정된 형식의 체인에 연결된 프레임을 만듭니다.
CanConvert 원본 픽셀 형식을 대상 픽셀 형식으로 변환할 수 있는지 여부를 결정합니다.
CanConvert 형식 변환기가 한 형식에서 다른 형식으로 변환할 수 있는지 쿼리합니다.
ClearIndexing IWICJpegFrameDecode::SetIndexing을 사용하여 인덱싱된 JPEG에서 인덱싱을 제거합니다.
Clone 현재 IWICEnumMetadataItem의 복사본을 만듭니다.
Commit 이미지에 대한 모든 변경 내용을 커밋하고 스트림을 닫습니다.
Commit 프레임을 이미지에 커밋합니다.

Commit 이미지 스트림에 대한 메타데이터 변경 내용을 완료합니다.
CopyBlocks 기본적으로 DDS 파일 내에 저장되므로 픽셀 데이터를 요청합니다.
CopyPalette 디코더의 IWICPalette를 복사합니다.
CopyPalette 인덱싱된 픽셀 형식의 색 테이블을 검색합니다.
CopyPixels 개체에 픽셀을 생성하도록 지시합니다.
CopyPixels 제공된 입력 매개 변수를 사용하여 픽셀 데이터를 복사합니다.
CopyPixels 픽셀을 대상 평면에 복사합니다. 제공된 입력 매개 변수에 의해 구성됩니다.
CopyScan 압축된 JPEG 검사의 복사본을 WIC 디코더 프레임의 출력 스트림에서 직접 검색합니다.
CreateBitmap IWICBitmap 개체를 만듭니다.
CreateBitmapClipper IWICBitmapClipper 개체의 새 인스턴스를 만듭니다.
CreateBitmapFlipRotator IWICBitmapFlipRotator 개체의 새 인스턴스를 만듭니다.
CreateBitmapFromHBITMAP 비트맵 핸들에서 IWICBitmap을 만듭니다.
CreateBitmapFromHICON

아이콘 핸들에서 IWICBitmap을 만듭니다.
CreateBitmapFromMemory 메모리 블록에서 IWICBitmap을 만듭니다.
CreateBitmapFromSource IWICBitmapSource에서 IWICBitmap을 만듭니다.
CreateBitmapFromSourceRect IWICBitmapSource의 지정된 사각형에서 IWICBitmap을 만듭니다.
CreateBitmapScaler IWICBitmapScaler의 새 인스턴스를 만듭니다.
CreateColorContext IWICColorContext 클래스의 새 인스턴스를 만듭니다.
CreateColorTransformer IWICColorTransform 클래스의 새 인스턴스를 만듭니다.
CreateComponentEnumerator 지정된 구성 요소 형식의 IEnumUnknown 개체를 만듭니다.
CreateComponentInfo 지정된 CLSID(구성 요소 클래스 식별자)에 대한 IWICComponentInfo 클래스의 새 인스턴스를 만듭니다.
CreateDecoder IWICBitmapDecoder의 새 인스턴스를 만듭니다.
CreateDecoderFromFileHandle 지정된 파일 핸들을 기반으로 IWICBitmapDecoder의 새 인스턴스를 만듭니다.
CreateDecoderFromFilename 지정된 파일을 기반으로 IWICBitmapDecoder 클래스의 새 인스턴스를 만듭니다.
CreateDecoderFromStream 지정된 IStream을 기반으로 IWICBitmapDecoder 클래스의 새 인스턴스를 만듭니다.

CreateEncoder IWICBitmapEncoder 클래스의 새 인스턴스를 만듭니다.
CreateEncoderPropertyBag 인코더 속성 모음을 만듭니다.
CreateFastMetadataEncoderFromDecoder 지정된 IWICBitmapDecoder를 기반으로 빠른 메타데이터 인코더의 새 인스턴스를 만듭니다.
CreateFastMetadataEncoderFromFrameDecode 지정된 이미지 프레임을 기반으로 빠른 메타데이터 인코더의 새 인스턴스를 만듭니다.
CreateFormatConverter IWICFormatConverter 클래스의 새 인스턴스를 만듭니다.
CreateImageEncoder 새 이미지 인코더 개체를 만듭니다.
CreateInstance 새 IWICBitmapDecoder 인스턴스를 만듭니다.
CreateInstance 새 IWICBitmapEncoder 인스턴스를 만듭니다.
CreateInstance 새 IWICFormatConverter 인스턴스를 만듭니다.
CreateInstance IWICMetadataReader의 인스턴스를 만듭니다.
CreateInstance IWICMetadataWriter의 인스턴스를 만듭니다.
CreateMetadataReader 지정된 매개 변수를 기반으로 IWICMetadataReader를 만듭니다. (IWICComponentFactory.CreateMetadataReader)

CreateMetadataReaderFromContainer 지정된 매개 변수를 기반으로 IWICMetadataReader를 만듭니다. (IWICComponentFactory.CreateMetadataReaderFromContainer)
CreateMetadataWriter 지정된 매개 변수를 기반으로 IWICMetadataWriter를 만듭니다.
CreateMetadataWriterFromReader 지정된 IWICMetadataReader에서 IWICMetadataWriter를 만듭니다.
CreateNewFrame 새 IWICBitmapFrameEncode 인스턴스를 만듭니다.
CreateNewFrame 인코딩할 새 프레임을 만듭니다.
CreatePalette IWICPalette 클래스의 새 인스턴스를 만듭니다.
CreateQueryReaderFromBlockReader 지정된 IWICMetadataBlockReader에서 IWICMetadataQueryReader를 만듭니다.
CreateQueryWriter 쿼리 작성기의 새 인스턴스를 만듭니다.
CreateQueryWriterFromBlockWriter 지정된 IWICMetadataBlockWriter에서 IWICMetadataQueryWriter를 만듭니다.
CreateQueryWriterFromReader 지정된 쿼리 판독기를 기반으로 쿼리 작성기의 새 인스턴스를 만듭니다. 쿼리 작성기는 쿼리 판독기에서 메타데이터로 미리 채워집니다.
CreateStream IWICStream 클래스의 새 인스턴스를 만듭니다.
DoesRequireFixedSize 메타데이터 처리기에 고정 크기가 필요한지 여부를 확인합니다.

DoesRequireFullStream 처리기에 전체 스트림이 필요한지 여부를 확인합니다.
DoesSupportAnimation 코덱이 애니메이션을 지원하는지 여부를 나타내는 값을 검색합니다.
DoesSupportChainType 지정된 체인 형식이 지원되는지 여부를 확인합니다.
DoesSupportChromakey 코덱이 chromakeys를 지원하는지 여부를 나타내는 값을 검색합니다.
DoesSupportIndexing 이 디코더가 효율적인 임의 액세스를 위해 인덱싱을 지원하는지 여부를 나타내는 값을 검색합니다.
DoesSupportLossless 코덱이 무손실 형식을 지원하는지 여부를 나타내는 값을 검색합니다.
DoesSupportMultiframe 코덱이 다중 프레임 이미지를 지원하는지 여부를 나타내는 값을 검색합니다.
DoesSupportPadding 메타데이터 처리기가 패딩을 지원하는지 여부를 결정합니다.
DoesSupportTransform IWICBitmapSourceTransform 인터페이스의 구현을 통해 특정 변환 옵션이 기본적으로 지원되는지 여부를 결정합니다.
DoesSupportTransform 이 메서드를 사용하여 원하는 평면 출력이 지원되는지 확인하고 호출자가 최적화된 코드 경로(있는 경우)를 선택할 수 있도록 합니다.
GetAcHuffmanTable 지정된 검사 및 테이블에 대한 AC Huffman 테이블의 복사본을 검색합니다. (IWICJpegFrameDecode.GetAcHuffmanTable)
GetAcHuffmanTable

지정된 검사 및 테이블에 대한 AC Huffman 테이블의 복사본을 검색합니다. (IWICJpegFrameEncode.GetAcHuffmanTable)
GetAuthor 구성 요소 작성자의 이름을 검색합니다.
GetBitsPerPixel 픽셀 형식의 BPP(픽셀당 비트)를 가져옵니다.
GetChainedFrame 지정된 형식의 체인에 대한 프레임을 검색합니다.
GetChainedFrameCount 지정된 형식의 체인에 대한 프레임 수를 검색합니다.
GetChannelCount 픽셀 형식에 포함된 채널 수를 가져옵니다.
GetChannelMask 픽셀 형식의 채널 마스크를 가져옵니다.
GetClosestPixelFormat 원하는 형식이 지정된 경우 IWICBitmapSourceTransform 구현에서 픽셀을 기본적으로 복사할 수 있는 가장 가까운 픽셀 형식을 검색합니다.
GetClosestSize 원하는 차원에 따라 구현에서 기본적으로 크기를 조정할 수 있는 가장 가까운 차원을 반환합니다.
GetCLSID 구성 요소의 클래스 식별자(CLSID)를 검색합니다.
GetColorContext 픽셀 형식의 IWICColorContext를 가져옵니다.
GetColorContexts 이미지의 IWICColorContext 개체를 검색합니다.
GetColorContexts

이미지 프레임과 연결된 IWICColorContext를 검색합니다.
GetColorCount 색 테이블의 색 수를 검색합니다.
GetColorManagementVersion 코덱이 지원하는 색 관리 버전 번호를 검색합니다.
GetColors 제공된 색 배열을 내부 색 테이블의 색으로 채웁니다. 색 배열은 GetColorCount의 반환 결과에 따라 크기가 조정되어야 합니다.
GetComponentType 구성 요소의 WICComponentType을 검색합니다.
GetContainerFormat 코덱과 연결된 컨테이너 GUID를 검색합니다.
GetContainerFormat 이미지의 컨테이너 형식을 검색합니다.
GetContainerFormat 인코더의 컨테이너 형식을 검색합니다.
GetContainerFormat 메타데이터 쿼리 판독기 컨테이너 형식을 가져옵니다.
GetContainerFormat 디코더의 컨테이너 형식을 검색합니다.
GetContainerFormats 메타데이터 처리기에서 지원하는 컨테이너 형식을 검색합니다.
GetContrast 원시 이미지의 대비 값을 가져옵니다.
GetCount 최상위 메타데이터 블록 수를 검색합니다.

GetCount 판독기 내의 메타데이터 항목 수를 가져옵니다.
GetCurrentLevel 디코더의 현재 점진적 수준을 가져옵니다.
GetCurrentParameterSet 현재 매개 변수 집합을 가져옵니다.
GetDataPointer 잠긴 사각형의 왼쪽 위 픽셀에 대한 포인터를 가져옵니다.
GetDcHuffmanTable 지정된 검사 및 테이블에 대한 DC Huffman 테이블의 복사본을 검색합니다. (IWICJpegFrameDecode.GetDcHuffmanTable)
GetDcHuffmanTable 지정된 검사 및 테이블에 대한 DC Huffman 테이블의 복사본을 검색합니다. (IWICJpegFrameEncode.GetDcHuffmanTable)
GetDecoderInfo 이미지에 대한 IWICBitmapDecoderInfo를 검색합니다.
GetDeviceManufacturer 코덱과 연결된 디바이스 제조업체의 이름을 검색합니다.
GetDeviceManufacturer 메타데이터 처리기의 디바이스 제조업체를 검색합니다.
GetDeviceModels 코덱과 연결된 디바이스 모델의 쉼표로 구분된 목록을 검색합니다.
GetDeviceModels 메타데이터 처리기를 지원하는 디바이스 모델을 검색합니다.
GetEncoderInfo 인코더에 대한 IWICBitmapEncoderInfo를 검색합니다.

GetEnumerator 메타데이터 계층 내의 현재 상대 위치에 있는 모든 메타데이터 항목의 열거자를 가져옵니다.
GetEnumerator 각 최상위 메타데이터 블록을 나타내는 IWICMetadataReader 개체의 열거형을 검색합니다.
GetEnumerator 모든 메타데이터 항목의 열거자를 가져옵니다.
GetExifColorSpace EXIF(Exchangeable Image File) 색 공간 색 컨텍스트를 검색합니다.
GetExposureCompensation 원시 이미지의 노출 보정 중지 값을 가져옵니다.
GetFileExtensions 코덱과 연결된 파일 이름 확장 프로그램의 쉼표로 구분된 목록을 검색합니다.
GetFormatGUID 픽셀 형식 GUID를 가져옵니다.
GetFormatInfo DDS 이미지가 저장되는 형식에 대한 정보를 가져옵니다.
GetFrame 이미지의 지정된 프레임을 검색합니다.
GetFrame DDS 이미지의 지정된 프레임을 검색합니다.
GetFrameCount 이미지의 총 프레임 수를 검색합니다.
GetFrameHeader 전체 프레임에서 헤더 데이터를 검색합니다.
GetFriendlyName

구성 요소에 대해 사람이 읽을 수 있는 표시 이름인 구성 요소의 친숙한 이름을 검색합니다.
GetGamma 원시 이미지의 현재 감마 설정을 가져옵니다.
GetHeader 메타데이터 기록기에 대한 메타데이터 헤더를 가져옵니다.
GetKelvinRangeInfo 원시 이미지의 현재 켈빈 범위에 대한 정보를 가져옵니다.
GetLevelCount CODEC에서 지원하는 점진적 디코딩 수준 수를 가져옵니다.
GetLocation 루트 메타데이터 블록을 기준으로 현재 경로를 검색합니다.
GetMetadataByName 메타데이터 쿼리 식으로 식별되는 메타데이터 블록 또는 항목을 검색합니다.
GetMetadataFormat 메타데이터 처리기의 메타데이터 형식을 검색합니다.
GetMetadataFormat 판독기와 연결된 메타데이터 형식을 가져옵니다.
GetMetadataHandlerInfo 판독기와 연결된 메타데이터 처리기 정보를 가져옵니다.
GetMetadataQueryReader 디코더에서 메타데이터 쿼리 판독기를 검색합니다.
GetMetadataQueryReader 프레임에 대한 메타데이터 쿼리 판독기를 검색합니다.
GetMetadataQueryWriter 인코더에 대한 메타데이터 쿼리 작성기를 검색합니다.

GetMetadataQueryWriter 인코더 프레임에 대한 메타데이터 쿼리 작성기를 가져옵니다.
GetMetadataQueryWriter 빠른 메타데이터 인코딩을 위해 메타데이터 쿼리 작성기를 검색합니다.
GetMimeTypes 코덱과 연결된 mime 형식의 쉼표로 구분된 시퀀스를 검색합니다.
GetNamedWhitePoint 원시 이미지의 명명된 흰색 점을 가져옵니다.
GetNoiseReduction 원시 이미지의 노이즈 감소 값을 가져옵니다.
GetNumericRepresentation 픽셀 형식의 WICPixelFormatNumericRepresentation을 검색합니다.
GetParameters DDS 관련 데이터를 가져옵니다. (IWICDdsDecoder.GetParameters)
GetParameters DDS 관련 데이터를 가져옵니다. (IWICDdsEncoder.GetParameters)
GetPatterns 디코더에서 지원하는 파일 패턴 서명을 검색합니다.
GetPatterns 메타데이터 판독기와 연결된 메타데이터 패턴을 가져옵니다.
GetPersistOptions 스트림을 사용하여 구성 요소를 초기화할 때 사용되는 지속형 옵션을 가져옵니다.
GetPixelFormat 잠긴 픽셀 영역의 픽셀 형식을 가져옵니다. 잠긴 영역에서 픽셀당 바이트 수를 계산하는 데 사용할 수 있습니다.

GetPixelFormat 비트맵 소스의 픽셀 형식을 검색합니다.
GetPixelFormats 코덱이 지원하는 픽셀 형식을 검색합니다.
GetPixelFormats 변환기가 지원하는 픽셀 형식을 나타내는 GUID 목록을 검색합니다.
GetPreferredVendorGUID 기본 공급 업체 GUID를 가져옵니다.
GetPreview 지원되는 경우 미리 보기 이미지를 검색합니다.
GetProfileBytes 색 컨텍스트 프로필을 검색합니다.
GetQuantizationTable 양자화 테이블의 복사본을 검색합니다. (IWICJpegFrameDecode.GetQuantizationTable)
GetQuantizationTable 양자화 테이블의 복사본을 검색합니다. (IWICJpegFrameEncode.GetQuantizationTable)
GetReaderByIndex 지정된 최상위 메타데이터 블록에 대한 IWICMetadataReader를 검색합니다.
GetRenderMode 현재 WICRawRenderMode를 가져옵니다.
GetResolution 픽셀과 실제 세계 측정 사이의 샘플링 속도를 검색합니다.
GetRotation 현재 회전 각도를 가져옵니다.
GetSaturation

원시 이미지의 채도 값을 가져옵니다.
GetScanHeader 지정된 인덱스를 사용하여 검사에 대한 SOS(검사 시작) 표식에서 매개 변수를 검색합니다.
GetSharpness 원시 이미지의 선명도 값을 가져옵니다.
GetSigningStatus 구성 요소의 서명 상태를 검색합니다.
GetSize 잠긴 사각형의 너비와 높이를 픽셀 단위로 검색합니다.
GetSize 비트맵의 픽셀 너비와 높이를 검색합니다.
GetSizeInBlocks DDS 이미지의 너비와 높이를 블록 단위로 가져옵니다.
GetSpecVersion 구성 요소의 사양 버전을 검색합니다.
GetStream 구성 요소에서 보유하는 스트림을 가져옵니다.
GetStride 메모리의 stride 값에 대한 액세스를 제공합니다.
GetThumbnail 이미지의 비트맵 썸네일(있는 경우)을 검색합니다.
GetThumbnail 코덱에서 지원하는 경우 프레임의 작은 미리 보기를 검색합니다.
GetTint 원시 이미지의 색조 값을 가져옵니다.

GetToneCurve 원시 이미지의 톤 곡선을 가져옵니다.
GetType 색 컨텍스트 형식을 검색합니다. (IWICColorContext.GetType)
GetType 색상표를 설명하는 WICBitmapPaletteType을 검색합니다.
GetValue 메타데이터 항목 값을 가져옵니다.
GetValueByIndex 지정된 인덱스에서 메타데이터 항목을 가져옵니다.
GetVendorGUID 공급업체 GUID를 검색합니다.
GetVersion 구성 요소의 버전을 검색합니다.
GetWhitePointKelvin 원시 이미지의 흰색 점 켈빈 온도를 가져옵니다.
GetWhitePointRGB 흰색 점 RGB 값을 가져옵니다.
GetWriterByIndex 지정된 인덱스에 있는 IWICMetadataWriter를 검색합니다.
HasAlpha 색상표에 불투명하지 않은 항목(즉, 알파가 1보다 작은 항목)이 포함되어 있는지 여부를 나타냅니다.
Initialize 제공된 매개 변수를 사용하여 비트맵 클리퍼를 초기화합니다.
Initialize

제공된 스트림을 사용하여 디코더를 초기화합니다.
Initialize 인코더에 비트를 인코딩할 위치를 알려주는 IStream을 사용하여 인코더를 초기화합니다.
Initialize 제공된 매개 변수를 사용하여 비트맵 대칭 이동 회전자를 초기화합니다.
Initialize 지정된 속성을 사용하여 프레임 인코더를 초기화합니다.
Initialize 제공된 매개 변수를 사용하여 비트맵 눈금자를 초기화합니다.
Initialize IWICBitmapSource를 사용하여 IWICColorTransform을 초기화하고 IWICColorContext에서 다른 IWICColorContext로 변환합니다.
Initialize 형식 변환기를 초기화합니다.
Initialize 평면 소스를 사용하여 형식 변환기를 초기화하고 인터리브 출력 픽셀 형식을 지정합니다.
InitializeCustom 제공된 사용자 지정 색 항목으로 색상표를 초기화합니다.
InitializeFromBitmap 참조 비트맵을 기반으로 계산된 최적화 값을 사용하여 팔레트를 초기화합니다.
InitializeFromBlockReader 지정된 IWICMetadataBlockReader에서 IWICMetadataBlockWriter를 초기화합니다. 이렇게 하면 메타데이터 블록 판독기에서 모든 메타데이터가 메타데이터 블록 기록기를 미리 채우게 됩니다.
InitializeFromExifColorSpace EXIF(Exchangeable Image File) 색 공간을 사용하여 색 컨텍스트를 초기화합니다.
InitializeFromFilename

지정된 파일에서 색 컨텍스트를 초기화합니다.
InitializeFromFilename 특정 파일에서 스트림을 초기화합니다.
InitializeFromIStream 다른 스트림에서 스트림을 초기화합니다. 액세스 권한은 기본 스트림에서 상속됩니다.
InitializeFromIStreamRegion 스트림을 다른 스트림의 하위 스트림으로 초기화합니다.
InitializeFromMemory 메모리 블록에서 색 컨텍스트를 초기화합니다.
InitializeFromMemory 메모리 블록을 스트림으로 처리하도록 스트림을 초기화합니다. 스트림은 버퍼 크기를 초과할 수 없습니다.
InitializeFromPalette 지정된 색상표에 따라 색상표를 초기화합니다.
InitializePredefined 색상표를 WICBitmapPaletteType으로 지정된 미리 정의된 색상표 중 하나로 초기화하고 선택적으로 투명한 색을 추가합니다.
IsBlackWhite 색상표가 흑백인지 여부를 설명하는 값을 검색합니다.
IsGrayscale 색상표가 회색조인지 여부를 설명하는 값을 검색합니다.
LoadEx 지정된 매개 변수를 사용하여 입력 스트림에서 데이터를 로드합니다.
LoadParameterSet 원하는 WICRawParameterSet 옵션을 설정합니다.
Lock 비트맵의 사각형 영역에 대한 액세스를 제공합니다.

MatchesMimeType 지정된 mime 형식이 코덱의 mime 형식과 일치하는지 여부를 나타내는 값을 검색합니다.
MatchesPattern 코덱이 지정된 스트림 내에서 패턴을 인식하는지 여부를 나타내는 값을 검색합니다.
MatchesPattern 스트림에 메타데이터 항목 패턴이 포함되어 있는지 여부를 확인합니다.
Next 열거형의 현재 위치를 고급으로 설정합니다.
Notify 원시 이미지 매개 변수 변경 알림에 사용되는 애플리케이션 정의 콜백 메서드입니다.
Notify 알림 방법은 규정 준수에 대해서만 문서화됩니다. 해당 사용은 권장되지 않으며 나중에 변경되거나 사용할 수 없습니다. 대신 RegisterProgressNotification을 사용합니다.
PFNProgressNotification 코덱 구성 요소 진행률이 발생할 때 호출되는 애플리케이션 정의 콜백 함수입니다.
QueryCapability 지정된 스트림을 기반으로 디코더의 기능을 검색합니다.
QueryRawCapabilitiesInfo 원시 이미지에 대해 지원되는 기능에 대한 정보를 검색합니다.
RefreshStream 유지 중인 스트림의 콘텐츠가 변경되었을 수 있음을 구성 요소에 알릴 수 있습니다. 구성 요소는 스트림에서 캐시된 정보를 더티하여 응답해야 합니다.
RegisterProgressNotification 진행률 알림 콜백 함수를 등록합니다.
RemoveMetadataByName 메타데이터 쿼리 식을 사용하여 특정 위치에서 메타데이터 항목을 제거합니다.

RemoveValue 지정된 매개 변수와 일치하는 메타데이터 항목을 제거합니다.
RemoveValueByIndex 지정된 인덱스에서 메타데이터 항목을 제거합니다.
RemoveWriterByIndex 지정된 인덱스 위치에서 메타데이터 기록기를 제거합니다.
Reset 현재 위치를 열거형의 시작 부분으로 다시 설정합니다.
SaveEx 지정된 매개 변수를 사용하여 지정된 입력 IStream에 IWICPersistStream을 저장합니다.
SetColorContexts 인코더에 대한 IWICColorContext 개체를 설정합니다.
SetColorContexts 지정된 숫자 IWICColorContext 프로필을 프레임으로 설정합니다.
SetContrast 원시 이미지의 대비 값을 설정합니다.
SetCurrentLevel CopyPixels에 대한 다음 호출에서 검색할 수준을 지정합니다.
SetDestinationColorContext 대상 색 컨텍스트를 설정합니다.
SetExposureCompensation 노출 보정 중지 값을 설정합니다.
SetGamma 원하는 감마 값을 설정합니다.
SetIndexing

효율적인 임의 액세스를 위해 JPEG의 인덱싱을 사용하도록 설정합니다.
SetMetadataByName 메타데이터 항목을 특정 위치로 설정합니다.
SetNamedWhitePoint 원시 파일의 명명된 흰색 점을 설정합니다.
SetNoiseReduction 원시 이미지의 노이즈 감소 값을 설정합니다.
SetNotificationCallback 알림 콜백 메서드를 설정합니다.
SetPalette 색상표 수정에 대한 액세스를 제공합니다.
SetPalette 이미지의 전역 색상표를 설정합니다.
SetPalette 인덱싱된 픽셀 형식의 IWICPalette를 설정합니다.
SetParameters DDS 관련 데이터를 설정합니다.
SetPixelFormat 인코더가 지정된 픽셀 형식을 사용하도록 요청합니다.
SetPreview 이미지의 전역 미리 보기를 설정합니다.
SetRenderMode 현재 WICRawRenderMode를 설정합니다.
SetResolution 이미지의 물리적 해상도를 변경합니다.

SetResolution 출력 이미지의 물리적 해상도를 설정합니다.
SetRotation 원하는 회전 각도를 설정합니다.
SetSaturation 원시 이미지의 채도 값을 설정합니다.
SetSharpness 원시 이미지의 선명도 값을 설정합니다.
SetSize 프레임의 출력 이미지 크기를 설정합니다.
SetThumbnail 이미지의 전역 썸네일을 설정합니다.
SetThumbnail 코덱에서 지원하는 경우 프레임 썸네일을 설정합니다.
SetTint 원시 이미지의 색조 값을 설정합니다.
SetToneCurve 원시 이미지의 톤 곡선을 설정합니다.
SetValue 지정된 메타데이터 항목을 설정합니다.
SetValueByIndex 메타데이터 항목을 지정된 인덱스로 설정합니다.
SetWhitePointKelvin 흰색 점 켈빈 값을 설정합니다.
SetWhitePointRGB

흰색 점 RGB 값을 설정합니다.
SetWriterByIndex 지정된 인덱스 위치에 있는 메타데이터 기록기를 바꿉니다.
Skip 지정된 개체 수로 건너뜁니다.
SupportsTransparency 형식이 투명 픽셀을 지원하는지 여부를 반환합니다.
WICConvertBitmapSource 지정된 IWICBitmapSource에서 원하는 픽셀 형식으로 IWICBitmapSource를 가져옵니다.
WICCreateBitmapFromSection GDI(Windows Graphics Device Interface) 섹션 핸들의 픽셀에 의해 지원되는 IWICBitmapSource를 반환합니다. (WICCreateBitmapFromSection)
WICCreateBitmapFromSectionEx GDI(Windows Graphics Device Interface) 섹션 핸들의 픽셀에 의해 지원되는 IWICBitmapSource를 반환합니다. (WICCreateBitmapFromSectionEx)
WICGetMetadataContentSize 지정된 IWICMetadataWriter에 포함된 메타데이터 콘텐츠의 크기를 반환합니다. 반환된 크기는 헤더 및 메타데이터의 길이를 차지합니다.
WICMapGuidToShortName 지정된 GUID와 연결된 짧은 이름을 가져옵니다.
WICMapSchemaToName 지정된 스키마와 연결된 이름을 가져옵니다.
WICMapShortNameToGuid 지정된 짧은 이름과 연결된 GUID를 가져옵니다.
WICMatchMetadataContent 지정된 스트림 내의 콘텐츠와 가장 일치하는 지정된 컨테이너 형식 및 공급업체에 대한 메타데이터 형식 GUID를 가져옵니다.

WICSerializeMetadataContent 지정된 스트림에 메타데이터를 씁니다.
WriteFrame IWICBitmapFrameEncode에서 제공하는 프레임에 이미지를 인코딩합니다.
WriteFrameThumbnail 이미지를 IWICBitmapFrameEncode에서 제공하는 프레임에 썸네일로 인코딩합니다.
WritePixels 호출자가 제공한 버퍼에서 IWICBitmapFrameEncode 개체로 스캔 라인 데이터를 복사합니다.
WritePixels 원본 평면에서 인코딩된 형식으로 줄을 씁니다. (IWICPlanarBitmapFrameEncode.WritePixels)
WriteScan JPEG 프레임에 스캔 데이터를 씁니다.
WriteSource 비트맵 원본을 인코딩합니다.
WriteSource 원본 평면에서 인코딩된 형식으로 줄을 씁니다. (IWICPlanarBitmapFrameEncode.WriteSource)
WriteThumbnail 지정된 이미지를 지정된 WIC 비트맵 인코더에 썸네일로 인코딩합니다.

Interfaces

테이블 확장

IWICBitmap 비트맵의 쓰기 가능성 및 정적 메모리 내 표현의 개념을 IWICBitmapSource에 추가하는 메서드를 정의합니다.

IWICBitmapClipper 지정된 사각형 영역에 대한 입력 비트맵의 잘린 버전을 생성하는 메서드를 노출합니다.
IWICBitmapCodecInfo 특정 코덱에 대한 정보를 제공하는 메서드를 노출합니다.
IWICBitmapCodecProgressNotification 인코더 및 디코더에 대한 진행률 알림에 사용되는 메서드를 노출합니다.
IWICBitmapDecoder 디코더를 나타내는 메서드를 노출합니다.
IWICBitmapDecoderInfo 디코더에 대한 정보를 제공하는 메서드를 노출합니다.
IWICBitmapEncoder 썸네일, 프레임 및 색상표와 같은 인코더의 속성을 설정하는 메서드를 정의합니다.
IWICBitmapEncoderInfo 인코더에 대한 정보를 제공하는 메서드를 노출합니다.
IWICBitmapFlipRotator 가로 또는 세로로 대칭 이동하거나 90도 단위로 회전할 수 있는 비트맵 소스를 생성하는 메서드를 제공합니다. 대칭 이동은 회전 전에 수행됩니다.
IWICBitmapFrameChainReader 다양한 형식의 체인을 통해 현재 프레임에 연결된 프레임에 대한 액세스를 제공합니다.
IWICBitmapFrameChainWriter 프레임을 서로 다른 형식의 체인으로 연결합니다.
IWICBitmapFrameDecode 인코딩된 파일의 개별 이미지 프레임을 디코딩하는 메서드를 정의합니다.
IWICBitmapFrameEncode 인코더의 개별 이미지 프레임을 나타냅니다.

IWICBitmapLock Lock 메서드를 지원하는 메서드를 노출합니다.
IWICBitmapScaler 다시 샘플링 또는 필터링 알고리즘을 사용하여 입력 비트맵의 크기가 조정된 버전을 나타냅니다.
IWICBitmapSource 픽셀이 검색되는 원본을 참조하지만 다시 쓸 수 없는 메서드를 노출합니다.
IWICBitmapSourceTransform 기본 IWICBitmapSource 구현에 특정 작업을 오프로드하는 메서드를 노출합니다.
IWICColorContext 색 관리를 위한 메서드를 노출합니다.
IWICColorTransform 한 색 컨텍스트에서 다른 색 컨텍스트로 IWICBitmapSource를 변환하는 메서드를 노출합니다.
IWICComponentFactory 구성 요소 개발자가 사용하는 구성 요소를 만드는 메서드를 노출합니다. 여기에는 코덱 및 메타데이터 처리기 개발자가 사용할 메타데이터 판독기, 기록기 및 기타 서비스가 포함됩니다.
IWICComponentInfo 구성 요소 정보를 제공하는 메서드를 노출합니다.
IWICDdsDecoder DDS 이미지 형식과 관련된 정보 및 기능을 제공합니다.
IWICDdsEncoder 인코더에 DDS 형식 특정 정보를 쓸 수 있습니다.
IWICDdsFrameDecode 기본 DXGI_FORMAT 형식의 DDS 이미지 데이터의 단일 프레임과 이미지 데이터에 대한 정보를 제공합니다.
IWICDevelopRaw 원시 코덱 형식의 기능에 대한 액세스를 제공하는 메서드를 노출합니다.

IWICDevelopRawNotificationCallback 원본 이미지 변경 알림에 대한 콜백 메서드를 제공합니다.
IWICEnumMetadataItem 개별 메타데이터 항목에 대한 열거형 서비스를 제공하는 메서드를 노출합니다.
IWICFastMetadataEncoder 현재 위치 메타데이터 편집에 사용되는 메서드를 노출합니다. 빠른 메타데이터 인코더를 사용하면 이미지를 완전히 다시 인코딩하지 않고도 이미지에 메타데이터를 추가하고 제거할 수 있습니다.
IWICFormatConverter 이미지 데이터를 한 픽셀 형식에서 다른 픽셀 형식으로 변환하여 디더링 및 하프톤을 인덱싱된 형식, 팔레트 변환 및 알파 임계값으로 처리하는 IWICBitmapSource를 나타냅니다.
IWICFormatConverterInfo 픽셀 형식 변환기 정보를 제공하는 메서드를 노출합니다.
IWICImageEncoder ID2D1Image 인터페이스를 IWICBitmapEncoder로 인코딩합니다.
IWICImagingFactory 디코더, 인코더 및 픽셀 형식 변환기와 같은 WIC(Windows 이미징 구성 요소)에 대한 구성 요소를 만드는 데 사용되는 메서드를 노출합니다.
IWICImagingFactory2 IWICImageEncoder를 만드는 기능을 포함하는 WIC 팩터리 인터페이스의 확장입니다.
IWICJpegFrameDecode JPEG 이미지를 디코딩하는 메서드를 노출합니다. SOF(Start Of Frame) 헤더, SOS(Start of Scan) 헤더, 허프만 및 양자화 테이블, 그리고 압축된 JPEG 데이터에 대한 액세스를 제공합니다. 또한 효율적인 임의 액세스를 위해 인덱싱을 사용하도록 설정합니다.
IWICJpegFrameEncode 압축된 JPEG 검사 데이터를 WIC 인코더의 출력 스트림에 직접 쓰기 위한 메서드를 노출합니다. 또한 Huffman 및 양자화 테이블에 대한 액세스를 제공합니다.
IWICMetadataBlockReader 코덱의 모든 최상위 메타데이터 블록에 대한 액세스를 제공하는 메서드를 노출합니다.

IWICMetadataBlockWriter 메타데이터 인코딩을 사용하도록 설정하는 메서드를 노출합니다. 이 인터페이스는 디코더 및 해당 이미지 프레임에 의해 구현됩니다.
IWICMetadataHandlerInfo 등록된 메타데이터 처리기에 대한 기본 정보를 제공하는 메서드를 노출합니다.
IWICMetadataQueryReader 메타데이터 쿼리 식을 사용하여 디코더 또는 해당 이미지 프레임에서 메타데이터 블록 및 항목을 검색하는 메서드를 노출합니다.
IWICMetadataQueryWriter 메타데이터 쿼리 식을 사용하여 메타데이터 블록 및 항목을 인코더 또는 해당 이미지 프레임에 설정하거나 제거하는 메서드를 노출합니다.
IWICMetadataReader 밑줄 메타데이터 콘텐츠에 대한 액세스를 제공하는 메서드를 노출합니다. 이 인터페이스는 ISV(독립 소프트웨어 공급업체)에 의해 구현되어 새 메타데이터 판독기를 만듭니다.
IWICMetadataReaderInfo 등록된 메타데이터 판독기에 대한 기본 정보를 제공하는 메서드를 노출합니다.
IWICMetadataWriter 메타데이터 콘텐츠 작성에 대한 액세스를 제공하는 메서드를 노출합니다. 이는 ISV(독립 소프트웨어 공급업체)에서 새 메타데이터 작성기를 만들기 위해 구현됩니다.
IWICMetadataWriterInfo 등록된 메타데이터 기록기에 대한 기본 정보를 제공하는 메서드를 노출합니다.
IWICPalette 주로 인덱싱된 픽셀 형식에 대한 색 테이블에 액세스하고 빌드하기 위한 메서드를 노출합니다.
IWICPersistStream 추가 로드를 제공하고 WICPersistOptions를 사용하는 메서드를 저장하는 메서드를 노출합니다.
IWICPixelFormatInfo 픽셀 형식에 대한 정보를 제공하는 메서드를 노출합니다.

IWICPixelFormatInfo2 픽셀 형식에 대한 추가 정보를 제공하여 IWICPixelFormatInfo를 확장합니다.
IWICPlanarBitmapFrameEncode 평면 구성 요소 이미지 픽셀을 인코더에 쓸 수 있습니다.
IWICPlanarBitmapSourceTransform 픽셀 구성 요소가 별도의 구성 요소 평면에 저장되는 평면 Y'CbCr 픽셀 형식에 대한 액세스를 제공합니다.
IWICPlanarFormatConverter 서식 변환기를 평면 원본으로 초기화할 수 있습니다.
IWICProgressCallback IWICProgressCallback 인터페이스는 규정 준수에 대해서만 문서화됩니다. 해당 사용은 권장되지 않으며 나중에 변경되거나 사용할 수 없습니다. 대신 RegisterProgressNotification을 사용합니다.
IWICProgressiveLevelControl 점진적 디코딩에 대한 정보를 가져오고 제어하는 메서드를 노출합니다.
IWICStream 이미징 및 메타데이터 콘텐츠를 참조하기 위한 WIC(Windows 이미징 구성 요소) 스트림을 나타냅니다.
IWICStreamProvider 스트림 공급자에 대한 메서드를 노출합니다.

Structures

테이블 확장

WICBitmapPattern 특정 형식을 식별하는 데 사용할 수 있는 이미지 파일 내의 패턴을 식별하는 멤버를 포함합니다.
WICBitmapPlane 평면 픽셀 형식에 대한 구성 요소 평면의 픽셀 형식, 버퍼, 스트라이드 및 크기를 지정합니다.

WICBitmapPlaneDescription 구성 요소 평면의 픽셀 형식과 크기를 지정합니다.
WICDdsFormatInfo DDS 형식의 DXGI_FORMAT 및 차단 정보를 지정합니다.
WICDdsParameters 포함된 데이터의 DDS 이미지 차원, DXGI_FORMAT 및 알파 모드를 지정합니다.
WICImageParameters 이미지를 인코딩할 때 일반적으로 사용되는 기본 매개 변수를 재정의하는 데 사용할 수 있는 매개 변수를 정의합니다.
WICJpegFrameHeader JPEG 프레임 헤더를 나타냅니다. (WICJpegFrameHeader)
WICJpegScanHeader JPEG 프레임 헤더를 나타냅니다. (WICJpegScanHeader)
WICMetadataHeader 메타데이터 헤더를 나타냅니다.
WICMetadataPattern 메타데이터 패턴을 나타냅니다.
WICRawCapabilitiesInfo 원시 코덱 기능을 정의합니다.
WICRawToneCurve 원시 이미지 톤 곡선을 나타냅니다.
WICRawToneCurvePoint 원시 이미지 톤 곡선 점을 나타냅니다.
WICRect WIC(Windows 이미징 구성 요소) API의 사각형을 나타냅니다.

Last updated on 2025. 07. 16.

WIP(Windows Information Protection)

아티클2023. 08. 22.

WIP(Windows Information Protection) 기술 개요입니다.

WIP(Windows Information Protection)를 개발하려면 다음 헤더가 필요합니다.

- edpwin32.h
- efswrtinterop.h
- srpapi.h

이 기술에 대한 지침은 다음을 참조하세요.

- WIP(Windows Information Protection)
- WIP(Windows Information Protection)에 대한 지침 및 모범 사례 생성
- Microsoft Intune 사용하여 WIP(Windows Information Protection) 정책 만들기

열거형

ENTERPRISE_DATA_POLICIES 앱이 WIP(Windows Information Protection)에 대해 인식되는지 여부와 앱이 정책에 의해 관리되는지 여부를 나타냅니다.

Functions

GetForWindow 현재 앱 창과 연결된 보호 정책 관리자 개체를 반환합니다.
ProtectFileToEnterpriseIdentity 파일의 데이터를 엔터프라이즈 ID로 보호하여 해당 엔터프라이즈 ID와 연결된 사용자만 데이터에 액세스할 수 있도록 합니다. 그런 다음, 애플리케이션은 표준 API를 사용하여 파일에서 읽거나 쓸 수 있습니다.
RequestAccessForAppWithAuditingInfoForWindowAsync

특정 대상 앱에 대한 엔터프라이즈 보호 콘텐츠에 대한 액세스를 요청합니다. (IProtectionPolicyManagerInterop2.RequestAccessForAppWithAuditingInfoForWindowAsync)
RequestAccessForAppWithMessageForWindowAsync 특정 대상 앱에 대한 엔터프라이즈 보호 콘텐츠에 대한 액세스를 요청합니다. (IProtectionPolicyManagerInterop2.RequestAccessForAppWithMessageForWindowAsync)
RequestAccessForAppWithWindowAsync 특정 대상 앱에 대한 엔터프라이즈 보호 콘텐츠에 대한 액세스를 요청합니다. (IProtectionPolicyManagerInterop2.RequestAccessForAppWithWindowAsync)
RequestAccessForWindowAsync ID에 대한 엔터프라이즈 보호 콘텐츠에 대한 액세스를 요청합니다. (IProtectionPolicyManagerInterop.RequestAccessForWindowAsync)
RequestAccessWithAuditingInfoForWindowAsync ID에 대한 엔터프라이즈 보호 콘텐츠에 대한 액세스를 요청합니다. (IProtectionPolicyManagerInterop2.RequestAccessWithAuditingInfoForWindowAsync)
RequestAccessWithMessageForWindowAsync ID에 대한 엔터프라이즈 보호 콘텐츠에 대한 액세스를 요청합니다. (IProtectionPolicyManagerInterop2.RequestAccessWithMessageForWindowAsync)
SrpCloseThreadNetworkContext SrpCreateThreadNetworkContext에서 선택적으로 반환되었을 수 있는 원래 컨텍스트로 스레드를 다시 복원합니다.
SrpCreateThreadNetworkContext 엔터프라이즈 ID를 현재 스레드의 데이터 컨텍스트로 설정합니다. 프로세스 컨텍스트에 동일한 엔터프라이즈 ID가 이미 있는 경우에만 허용됩니다. 필요에 따라 기존 스레드 토큰을 반환합니다.
SrpDisablePermissiveModeFileEncryption 현재 스레드에서 파일 암호화에 대한 허용 모드를 사용하지 않도록 설정합니다.
SrpDoesPolicyAllowAppExecution 소프트웨어 제한 정책에 따라 패키지된 앱을 실행할 수 있는지 여부를 평가합니다.
SrpEnablePermissiveModeFileEncryption

현재 스레드 및 이 스레드가 만들거나 게시하는 모든 스레드에서 파일 암호화에 대한 허용 모드를 사용하도록 설정합니다.
SrpGetEnterpriseIds 지정된 토큰의 엔터프라이즈 식별자 목록을 가져옵니다.
SrpGetEnterprisePolicy 정책은 Windows 10 버전 1607에 적용할 수 있습니다.앱의 엔터프라이즈 정책에 대한 정보를 가져옵니다.
SrpIsTokenService 서비스가 토큰 서비스인지 여부를 식별합니다.
SrpSetTokenEnterpriseId 토큰에 대한 데이터 의도를 설정합니다. 호출자 프로세스는 제공된 엔터프라이즈 ID에 대해 엔터프라이즈가 허용되어야 합니다.

인터페이스

IProtectionPolicyManagerInterop 보호된 콘텐츠에 대한 엔터프라이즈 보호 정책을 관리합니다. (IProtectionPolicyManagerInterop)
IProtectionPolicyManagerInterop2 보호된 콘텐츠에 대한 엔터프라이즈 보호 정책을 관리합니다. (IProtectionPolicyManagerInterop2)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Windows Installer 공급자

아티클2023. 08. 22.

Windows Installer 공급자 기술 개요입니다.

Windows Installer 공급자 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows Installer 공급자

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Windows 인터넷

아티클2023. 03. 16.

Windows 인터넷 기술 개요입니다.

Windows 인터넷을 개발하려면 다음 헤더가 필요합니다.

- proofofpossessioncookieinfo.h
- wininet.h
- winineti.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 인터넷

열거형

INTERNET_SCHEME URL_COMPONENTS 구조체의 nScheme 멤버와 함께 사용되는 플래그를 정의합니다.
InternetCookieState InternetCookieState 열거형은 쿠키의 상태를 정의합니다.

Functions

CommitUrlCacheEntryA 지정된 파일의 데이터를 인터넷 캐시에 저장하고 지정된 URL과 연결합니다. (ANSI)
CommitUrlCacheEntryW 지정된 파일의 데이터를 인터넷 캐시에 저장하고 지정된 URL과 연결합니다. (유니코드)
CreateMD5SSOHash CreateMD5SSOHash 함수(wininet.h)는 Microsoft Passport 암호를 가져오고, 특정 문자열을 사용하여 MD5 해시를 만들고, 결과를 반환합니다.

CreateMD5SSOHash CreateMD5SSOHash 함수(winineti.h)는 지정된 계정 또는 영역에 대한 기본 Microsoft Passport 암호를 가져오고 MD5 해시를 만들고 결과를 반환합니다.
CreateUrlCacheContainerA 지정된 캐시 경로에 캐시 컨테이너를 만들어 지정된 이름, 캐시 접두사 및 컨테이너 형식에 따라 캐시 항목을 저장합니다. (ANSI)
CreateUrlCacheContainerW 지정된 캐시 경로에 캐시 컨테이너를 만들어 지정된 이름, 캐시 접두사 및 컨테이너 형식에 따라 캐시 항목을 저장합니다. (유니코드)
CreateUrlCacheEntryA 지정된 URL 및 파일 이름 확장명을 기반으로 캐시 항목을 저장하기 위한 로컬 파일 이름을 만듭니다. (ANSI)
CreateUrlCacheEntryW 지정된 URL 및 파일 이름 확장명을 기반으로 캐시 항목을 저장하기 위한 로컬 파일 이름을 만듭니다. (유니코드)
CreateUrlCacheGroup 캐시 그룹 ID를 생성합니다.
DeleteUrlCacheContainerA 지정된 이름에 따라 캐시 컨테이너(캐시 항목 포함)를 삭제합니다. (ANSI)
DeleteUrlCacheContainerW 지정된 이름에 따라 캐시 컨테이너(캐시 항목 포함)를 삭제합니다. (유니코드)
DeleteUrlCacheEntry DeleteUrlCacheEntry 함수(wininet.h)는 파일이 있는 경우 캐시에서 원본 이름과 연결된 파일을 제거합니다.
DeleteUrlCacheEntryA 파일이 있는 경우 캐시에서 원본 이름과 연결된 파일을 제거합니다. (DeleteUrlCacheEntryA)

DeleteUrlCacheEntryW DeleteUrlCacheEntryW(유니코드) 함수(wininet.h)는 파일이 있는 경우 캐시에서 원본 이름과 연결된 파일을 제거합니다.
DeleteUrlCacheGroup 캐시 인덱스 파일에서 지정된 GROUPID 및 연결된 상태를 해제합니다.
DetectAutoProxyUrl DetectAutoProxyUrl 함수(wininet.h)는 WPAD 자동 프록시 스크립트의 위치를 확인하려고 시도합니다.
DetectAutoProxyUrl DetectAutoProxyUrl 함수(winineti.h)는 WPAD 자동 프록시 스크립트의 위치를 확인하려고 시도합니다.
FindCloseUrlCache 지정된 캐시 열거형 핸들을 닫습니다.
FindFirstUrlCacheEntryA 인터넷 캐시의 열거형을 시작합니다. (ANSI)
FindFirstUrlCacheEntryExA 인터넷 캐시의 필터링된 열거형을 시작합니다. (ANSI)
FindFirstUrlCacheEntryExW 인터넷 캐시의 필터링된 열거형을 시작합니다. (유니코드)
FindFirstUrlCacheEntryW 인터넷 캐시의 열거형을 시작합니다. (유니코드)
FindFirstUrlCacheGroup 인터넷 캐시에서 캐시 그룹의 열거를 시작합니다.
FindNextUrlCacheEntryA 인터넷 캐시에서 다음 항목을 검색합니다. (ANSI)

FindNextUrlCacheEntryExA FindFirstUrlCacheEntryEx 함수로 시작된 캐시 열거형에서 다음 캐시 항목을 찾습니다. (ANSI)
FindNextUrlCacheEntryExW FindFirstUrlCacheEntryEx 함수로 시작된 캐시 열거형에서 다음 캐시 항목을 찾습니다. (유니코드)
FindNextUrlCacheEntryW 인터넷 캐시에서 다음 항목을 검색합니다. (유니코드)
FindNextUrlCacheGroup FindFirstUrlCacheGroup에서 시작한 캐시 그룹 열거형에서 다음 캐시 그룹을 검색합니다.
FreeProofOfPossessionCookieInfoArray
FreeUrlCacheSpaceA 캐시의 공간을 확보합니다. (ANSI)
FreeUrlCacheSpaceW 캐시의 공간을 확보합니다. (유니코드)
FtpCommandA 명령을 FTP 서버로 직접 보냅니다. (ANSI)
FtpCommandW 명령을 FTP 서버로 직접 보냅니다. (유니코드)
FtpCreateDirectoryA FTP 서버에 새 디렉터리를 만듭니다. (ANSI)
FtpCreateDirectoryW FTP 서버에 새 디렉터리를 만듭니다. (유니코드)
FtpDeleteFileA FTP 서버에 저장된 파일을 삭제합니다. (ANSI)

FtpDeleteFileW FTP 서버에 저장된 파일을 삭제합니다. (유니코드)
FtpFindFirstFileA 지정된 FTP 세션의 지정된 디렉터리를 검색합니다. 파일 및 디렉터리 항목은 WIN32_FIND_DATA 구조의 애플리케이션에 반환됩니다. (ANSI)
FtpFindFirstFileW 지정된 FTP 세션의 지정된 디렉터리를 검색합니다. 파일 및 디렉터리 항목은 WIN32_FIND_DATA 구조의 애플리케이션에 반환됩니다. (유니코드)
FtpGetCurrentDirectoryA 지정된 FTP 세션의 현재 디렉터리를 검색합니다. (ANSI)
FtpGetCurrentDirectoryW 지정된 FTP 세션의 현재 디렉터리를 검색합니다. (유니코드)
FtpGetFileA FTP 서버에서 파일을 검색하고 지정된 파일 이름 아래에 저장하여 프로세스에 새 로컬 파일을 만듭니다. (ANSI)
FtpGetFileSize 요청된 FTP 리소스의 파일 크기를 검색합니다.
FtpGetFileW FTP 서버에서 파일을 검색하고 지정된 파일 이름 아래에 저장하여 프로세스에 새 로컬 파일을 만듭니다. (유니코드)
FtpOpenFileA 읽기 또는 쓰기를 위해 FTP 서버의 원격 파일에 대한 액세스를 시작합니다. (ANSI)
FtpOpenFileW 읽기 또는 쓰기를 위해 FTP 서버의 원격 파일에 대한 액세스를 시작합니다. (유니코드)
FtpPutFileA FTP 서버에 파일을 저장합니다. (ANSI)

FtpPutFileW FTP 서버에 파일을 저장합니다. (유니코드)
FtpRemoveDirectoryA FTP 서버에서 지정된 디렉터리를 제거합니다. (ANSI)
FtpRemoveDirectoryW FTP 서버에서 지정된 디렉터리를 제거합니다. (유니코드)
FtpRenameFileA FTP 서버에 저장된 파일의 이름을 바꿉니다. (ANSI)
FtpRenameFileW FTP 서버에 저장된 파일의 이름을 바꿉니다. (유니코드)
FtpSetCurrentDirectoryA FTP 서버의 다른 작업 디렉터리를 변경합니다. (ANSI)
FtpSetCurrentDirectoryW FTP 서버의 다른 작업 디렉터리를 변경합니다. (유니코드)
GetCookieInfoForUri 소유 증명 쿠키에 사용할 제공된 URI에 대한 쿠키 정보를 가져옵니다.
GetCookieInfoWithUriForAccount 제공된 WebAccount 및 URI에 해당하는 쿠키 정보를 검색합니다.
GetUrlCacheConfigInfoA 캐시 구성에 대한 정보를 검색합니다. (ANSI)
GetUrlCacheConfigInfoW 캐시 구성에 대한 정보를 검색합니다. (유니코드)
GetUrlCacheEntryInfoA 캐시 항목에 대한 정보를 검색합니다. (ANSI)

GetUrlCacheEntryInfoExA HttpSendRequest 함수에 의해 오프라인 모드로 적용되는 리디렉션을 고려하여 지정된 URL과 연결된 캐시 항목에 대한 정보를 검색합니다. (ANSI)
GetUrlCacheEntryInfoExW HttpSendRequest 함수에 의해 오프라인 모드로 적용되는 리디렉션을 고려하여 지정된 URL과 연결된 캐시 항목에 대한 정보를 검색합니다. (유니코드)
GetUrlCacheEntryInfoW 캐시 항목에 대한 정보를 검색합니다. (유니코드)
GetUrlCacheGroupAttributeA 지정된 캐시 그룹의 특성 정보를 검색합니다. (ANSI)
GetUrlCacheGroupAttributeW 지정된 캐시 그룹의 특성 정보를 검색합니다. (유니코드)
GOPHER_ATTRIBUTE_ENUMERATOR Gopher 서버에서 특성 정보를 처리하는 콜백 함수의 프로토타입입니다.
GopherCreateLocatorA 선택기 문자열의 구성 요소 부분에서 Gopher 또는 Gopher+ 로케이터 문자열을 만듭니다. (ANSI)
GopherCreateLocatorW 선택기 문자열의 구성 요소 부분에서 Gopher 또는 Gopher+ 로케이터 문자열을 만듭니다. (유니코드)
GopherFindFirstFileA Gopher 로케이터 및 검색 조건을 사용하여 서버와 세션을 만들고 요청된 문서, 이진 파일, 인덱스 서버 또는 디렉터리 트리를 찾습니다. (ANSI)
GopherFindFirstFileW Gopher 로케이터 및 검색 조건을 사용하여 서버와 세션을 만들고 요청된 문서, 이진 파일, 인덱스 서버 또는 디렉터리 트리를 찾습니다. (유니코드)
GopherGetAttributeA 서버에서 특정 특성 정보를 검색합니다. (ANSI)

GopherGetAttributeW 서버에서 특정 특성 정보를 검색합니다. (유니코드)
GopherGetLocatorTypeA Gopher 로케이터를 구문 분석하고 해당 특성을 결정합니다. (ANSI)
GopherGetLocatorTypeW Gopher 로케이터를 구문 분석하고 해당 특성을 결정합니다. (유니코드)
GopherOpenFileA Gopher 서버에서 Gopher 데이터 파일을 읽기 시작합니다. (ANSI)
GopherOpenFileW Gopher 서버에서 Gopher 데이터 파일을 읽기 시작합니다. (유니코드)
HttpAddRequestHeadersA HTTP 요청 핸들에 하나 이상의 HTTP 요청 헤더를 추가합니다. (HttpAddRequestHeadersA)
HttpAddRequestHeadersW HTTP 요청 핸들에 하나 이상의 HTTP 요청 헤더를 추가합니다. (HttpAddRequestHeadersW)
HttpEndRequestA HttpSendRequestEx에서 시작된 HTTP 요청을 종료합니다. (ANSI)
HttpEndRequestW HttpSendRequestEx에서 시작된 HTTP 요청을 종료합니다. (유니코드)
HttpOpenRequestA HTTP 요청 핸들을 만듭니다. (ANSI)
HttpOpenRequestW HTTP 요청 핸들을 만듭니다. (유니코드)
HttpQueryInfoA HTTP 요청과 연결된 헤더 정보를 검색합니다. (ANSI)

HttpQueryInfoW HTTP 요청과 연결된 헤더 정보를 검색합니다. (유니코드)
HttpSendRequestA 지정된 요청을 HTTP 서버로 보내 호출자가 일반적으로 HttpSendRequestEx에 전달되는 것 이상으로 추가 데이터를 보낼 수 있도록 합니다. (ANSI)
HttpSendRequestExA 지정된 요청을 HTTP 서버로 보냅니다. (HttpSendRequestExA)
HttpSendRequestExW 지정된 요청을 HTTP 서버로 보냅니다. (HttpSendRequestExW)
HttpSendRequestW 지정된 요청을 HTTP 서버로 보내 호출자가 일반적으로 HttpSendRequestEx에 전달되는 것 이상으로 추가 데이터를 보낼 수 있도록 합니다. (유니코드)
INTERNET_STATUS_CALLBACK 이 콜백 함수에 대한 포인터를 정의합니다.
InternetAttemptConnect 인터넷에 연결하려고 시도합니다.
InternetAutodial InternetAutodial 함수(wininet.h)를 사용하면 모뎀이 기본 인터넷 연결에 자동으로 전화를 걸 수 있습니다.
InternetAutodial InternetAutodial 함수(winineti.h)를 사용하면 모뎀이 기본 인터넷 연결에 자동으로 전화를 걸 수 있습니다.
InternetAutodialHangup InternetAutodialHangup 함수(wininet.h)는 자동 전화 접속 연결을 끊습니다.
InternetAutodialHangup InternetAutodialHangup 함수(winineti.h)는 자동 전화 접속 연결을 끊습니다.

InternetCanonicalizeUrlA 안전하지 않은 문자와 공백을 이스케이프 시퀀스로 변환하는 것을 포함하는 URL을 정식화합니다. (ANSI)
InternetCanonicalizeUrlW 안전하지 않은 문자와 공백을 이스케이프 시퀀스로 변환하는 것을 포함하는 URL을 정식화합니다. (유니코드)
InternetCheckConnectionA 인터넷에 연결할 수 있는 경우 애플리케이션이 검사 수 있습니다. (ANSI)
InternetCheckConnectionW 인터넷에 연결할 수 있는 경우 애플리케이션이 검사 수 있습니다. (유니코드)
InternetClearAllPerSiteCookieDecisions 사이트별로 쿠키에 대해 내린 모든 결정을 지웁니다.
InternetCloseHandle 단일 인터넷 핸들을 닫습니다.
InternetCombineUrlA 기본 및 상대 URL을 단일 URL로 결합합니다. 결과 URL은 정식화됩니다(InternetCanonicalizeUrl 참조). (ANSI)
InternetCombineUrlW 기본 및 상대 URL을 단일 URL로 결합합니다. 결과 URL은 정식화됩니다(InternetCanonicalizeUrl 참조). (유니코드)
InternetConfirmZoneCrossing InternetConfirmZoneCrossing 함수(wininet.h)는 보안 URL과 비보안 URL 간의 변경 내용을 확인합니다.
InternetConfirmZoneCrossingA 보안 URL과 안전하지 않은 URL 간의 변경 내용을 확인합니다. 두 URL 간의 보안이 변경되면 항상 사용자에게 알릴 수 있습니다. 일반적으로 애플리케이션은 사용자가 대화 상자와의 상호 작용을 통해 변경 사항을 승인하도록 허용해야 합니다. (InternetConfirmZoneCrossingA)

InternetConfirmZoneCrossingW InternetConfirmZoneCrossingW(유니코드) 함수(wininet.h)는 보안 URL과 안전하지 않은 URL 간의 변경 내용을 확인합니다.
InternetConnectA 지정된 사이트에 대한 FTP(파일 전송 프로토콜) 또는 HTTP 세션을 엽니다. (ANSI)
InternetConnectW 지정된 사이트에 대한 FTP(파일 전송 프로토콜) 또는 HTTP 세션을 엽니다. (유니코드)
InternetCrackUrlA URL을 구성 요소 부분으로 크래킹합니다. (ANSI)
InternetCrackUrlW URL을 구성 요소 부분으로 크래킹합니다. (유니코드)
InternetCreateUrlA 구성 요소 부분에서 URL을 만듭니다. (ANSI)
InternetCreateUrlW 구성 요소 부분에서 URL을 만듭니다. (유니코드)
InternetDial InternetDial 함수(wininet.h)는 모뎀을 사용하여 인터넷에 대한 연결을 시작합니다.
InternetDial InternetDial 함수(winineti.h)는 모뎀을 사용하여 인터넷에 대한 연결을 시작합니다.
InternetDialA ANSI(InternetDialA) 함수(wininet.h)는 모뎀을 사용하여 인터넷에 대한 연결을 시작합니다.
InternetDialA ANSI(InternetDialA) 함수(winineti.h)는 모뎀을 사용하여 인터넷에 대한 연결을 시작합니다.
InternetDialW InternetDialW(유니코드) 함수(wininet.h)는 모뎀을 사용하여 인터넷에 대한 연결을 시작합니다.

InternetDialW InternetDialW(유니코드) 함수(winineti.h)는 모뎀을 사용하여 인터넷에 대한 연결을 시작합니다.
InternetEnumPerSiteCookieDecisionA InternetEnumPerSiteCookieDecisionA(ANSI) 함수(wininet.h)는 설정된 사이트별 쿠키 규정을 사용하여 웹 사이트의 도메인 및 쿠키 설정을 검색합니다.
InternetEnumPerSiteCookieDecisionA InternetEnumPerSiteCookieDecisionA(ANSI) 함수(winineti.h)는 사이트별 쿠키 규정이 설정된 웹 사이트의 도메인 및 쿠키 설정을 검색합니다.
InternetEnumPerSiteCookieDecisionW InternetEnumPerSiteCookieDecisionW(유니코드) 함수(wininet.h)는 설정된 사이트별 쿠키 규정을 사용하여 웹 사이트의 도메인 및 쿠키 설정을 검색합니다.
InternetEnumPerSiteCookieDecisionW InternetEnumPerSiteCookieDecisionW(유니코드) 함수(winineti.h)는 사이트별 쿠키 규정이 설정된 웹 사이트의 도메인 및 쿠키 설정을 검색합니다.
InternetErrorDlg 적절한 대화 상자가 있는 경우 InternetErrorDlg에 전달되는 오류에 대한 대화 상자를 표시합니다.
InternetFindNextFileA FtpFindFirstFile.Windows XP 및 Windows Server 2003 R2 이전 호출의 결과로 시작된 파일 검색을 계속합니다. 또는 GopherFindFirstFile에 대한 이전 호출의 결과로 파일 검색을 계속합니다. (ANSI)
InternetFindNextFileW FtpFindFirstFile.Windows XP 및 Windows Server 2003 R2 이전 호출의 결과로 시작된 파일 검색을 계속합니다. 또는 GopherFindFirstFile에 대한 이전 호출의 결과로 파일 검색을 계속합니다. (유니코드)
InternetFreeCookies INTERNET_COOKIE2 구조체의 배열을 해제합니다.
InternetGetConnectedState InternetGetConnectedState 함수(wininet.h)는 로컬 시스템의 연결된 상태를 검색하며 사용하지 않는 것이 좋습니다.

InternetGetConnectedState InternetGetConnectedState 함수(winineti.h)는 로컬 시스템의 연결된 상태를 검색합니다.
InternetGetConnectedStateEx InternetGetConnectedStateEx 함수(wininet.h)는 지정된 인터넷 연결의 연결된 상태를 검색하며 사용하지 않는 것이 좋습니다.
InternetGetConnectedStateEx InternetGetConnectedStateEx 함수(winineti.h)는 지정된 인터넷 연결의 연결된 상태를 검색합니다.
InternetGetConnectedStateExA INTERNETGetConnectedStateExA(ANSI) 함수(wininet.h)는 지정된 인터넷 연결의 연결된 상태를 검색하며 사용하지 않는 것이 좋습니다.
InternetGetConnectedStateExW InternetGetConnectedStateExW(유니코드) 함수(wininet.h)는 지정된 인터넷 연결의 연결된 상태를 검색하며 사용하지 않는 것이 좋습니다.
InternetGetConnectedStateExW InternetGetConnectedStateExW(유니코드) 함수는 지정된 인터넷 연결의 연결된 상태를 검색합니다. 이 API를 사용하지 않는 것이 좋습니다. 대신 INetworkListManager::GetConnectivity 메서드를 사용합니다.
InternetGetCookieA 지정된 URL에 대한 쿠키를 검색합니다. (ANSI)
InternetGetCookieEx2 지정된 URL과 연결된 하나 이상의 쿠키를 검색합니다.
InternetGetCookieExA 지정된 URL과 연결된 쿠키에 저장된 데이터를 검색합니다. (ANSI)
InternetGetCookieExW 지정된 URL과 연결된 쿠키에 저장된 데이터를 검색합니다. (유니코드)
InternetGetCookieW 지정된 URL에 대한 쿠키를 검색합니다. (유니코드)

InternetGetLastResponseInfoA 이 함수를 호출하는 스레드에서 마지막 오류 설명 또는 서버 응답을 검색합니다. (ANSI)
InternetGetLastResponseInfoW 이 함수를 호출하는 스레드에서 마지막 오류 설명 또는 서버 응답을 검색합니다. (유니코드)
InternetGetPerSiteCookieDecisionA 지정된 도메인의 쿠키에 대한 결정을 검색합니다. (ANSI)
InternetGetPerSiteCookieDecisionW 지정된 도메인의 쿠키에 대한 결정을 검색합니다. (유니코드)
InternetGoOnline InternetGoOnline 함수(wininet.h)는 사용자에게 URL에 대한 연결을 시작할 수 있는 권한을 묻는 메시지를 표시합니다.
InternetGoOnline InternetGoOnline 함수(winineti.h)는 사용자에게 URL에 대한 연결을 시작할 수 있는 권한을 묻는 메시지를 표시합니다.
InternetGoOnlineA ANSI(InternetGoOnlineA) 함수(wininet.h)는 사용자에게 URL에 대한 연결을 시작할 수 있는 권한을 묻는 메시지를 표시합니다.
InternetGoOnlineA ANSI(InternetGoOnlineA) 함수(winineti.h)는 사용자에게 URL에 대한 연결을 시작할 수 있는 권한을 묻는 메시지를 표시합니다.
InternetGoOnlineW InternetGoOnlineW(유니코드) 함수(wininet.h)는 사용자에게 URL에 대한 연결을 시작할 수 있는 권한을 묻는 메시지를 표시합니다.
InternetGoOnlineW InternetGoOnlineW(유니코드) 함수(winineti.h)는 사용자에게 URL에 대한 연결을 시작할 수 있는 권한을 묻는 메시지를 표시합니다.
InternetHangUp InternetHangUp 함수(wininet.h)는 모뎀에 인터넷 연결을 끊도록 지시합니다.

InternetHangUp InternetHangUp 함수(winineti.h)는 모뎀에 인터넷 연결을 끊도록 지시합니다.
InternetInitializeAutoProxyDll InternetInitializeAutoProxyDll 함수(wininet.h)는 InternetInitializeAutoProxyDll이라는 두 개의 WinINet 함수 중 하나입니다.
InternetInitializeAutoProxyDll InternetInitializeAutoProxyDll이라는 두 개의 WinINet 함수가 있습니다. 첫 번째는 프록시 구성의 내부 상태를 새로 고치고 두 번째는 WinINet의 자동 프록시 지원의 일부입니다.
InternetLockRequestFile 사용 중인 파일에 잠금을 배치합니다.
InternetOpenA 애플리케이션의 WinINet 함수 사용을 초기화합니다. (ANSI)
InternetOpenUrlA 전체 FTP 또는 HTTP URL로 지정된 리소스를 엽니다. (ANSI)
InternetOpenUrlW 전체 FTP 또는 HTTP URL로 지정된 리소스를 엽니다. (유니코드)
InternetOpenW 애플리케이션의 WinINet 함수 사용을 초기화합니다. (유니코드)
InternetQueryDataAvailable 서버를 쿼리하여 사용 가능한 데이터의 양을 확인합니다.
InternetQueryOptionA 지정된 핸들에서 인터넷 옵션을 쿼리합니다. (ANSI)
InternetQueryOptionW 지정된 핸들에서 인터넷 옵션을 쿼리합니다. (유니코드)
InternetReadFile InternetOpenUrl, FtpOpenFile 또는 HttpOpenRequest 함수에서 연 핸들에서 데이터를 읽습니다.

InternetReadFileExA InternetOpenUrl 또는 HttpOpenRequest 함수에서 연 핸들에서 데이터를 읽습니다. (ANSI)
InternetReadFileExW InternetOpenUrl 또는 HttpOpenRequest 함수에서 연 핸들에서 데이터를 읽습니다. (유니코드)
InternetSetCookieA 지정된 URL과 연결된 쿠키를 만듭니다. (InternetSetCookieA)
InternetSetCookieEx2 지정된 URL과 연결된 쿠키를 만듭니다. (InternetSetCookieEx2)
InternetSetCookieExA InternetSetCookieEx 함수는 지정된 URL과 연결된 지정된 이름의 쿠키를 만듭니다. 이 함수는 타사 쿠키를 만들 수 있어 InternetSetCookie 함수와 다릅니다. (ANSI)
InternetSetCookieExW InternetSetCookieEx 함수는 지정된 URL과 연결된 지정된 이름의 쿠키를 만듭니다. 이 함수는 타사 쿠키를 만들 수 있어 InternetSetCookie 함수와 다릅니다. (유니코드)
InternetSetCookieW 지정된 URL과 연결된 쿠키를 만듭니다. (InternetSetCookieW)
InternetSetDialState InternetSetDialState 함수(wininet.h)는 지원되지 않으며 사용되지 않으며 사용해서는 안 됩니다.
InternetSetDialState InternetSetDialState 함수(wininet.h)는 사용되지 않으며 지원되지 않으며 사용하지 마세요.
InternetSetDialStateA ANSI(InternetSetDialStateA) 함수(wininet.h)는 지원되지 않으며 사용되지 않으며 사용해서는 안 됩니다.
InternetSetDialStateA ANSI(InternetSetDialStateA) 함수(wininet.h)는 사용되지 않으며 지원되지 않으며 사용하지 마세요.

InternetSetDialStateW InternetSetDialStateW(유니코드) 함수(wininet.h)는 지원되지 않으며 사용되지 않으며 사용해서는 안 됩니다.
InternetSetDialStateW InternetSetDialStateW(유니코드) 함수(wininet.h)는 사용되지 않으며 지원되지 않으며 사용하지 마세요.
InternetSetFilePointer InternetReadFile에 대한 파일 위치를 설정합니다. 동기 호출입니다. 그러나 InternetReadFile에 대한 후속 호출은 캐시에서 데이터를 사용할 수 없으며 서버가 임의 액세스를 지원하지 않는 경우 보류 중인 데이터를 차단하거나 반환할 수 있습니다.
InternetSetOptionA 인터넷 옵션을 설정합니다. (ANSI)
InternetSetOptionExA 지원되지 않습니다. InternetSetOption 함수를 호출하는 스텁으로만 구현됩니다. InternetSetOptionEx에는 자체 기능이 없습니다. 현재 이 함수를 사용하지 마세요. (ANSI)
InternetSetOptionExW 지원되지 않습니다. InternetSetOption 함수를 호출하는 스텁으로만 구현됩니다. InternetSetOptionEx에는 자체 기능이 없습니다. 현재 이 함수를 사용하지 마세요. (유니코드)
InternetSetOptionW 인터넷 옵션을 설정합니다. (유니코드)
InternetSetPerSiteCookieDecisionA 지정된 도메인의 쿠키에 대한 결정을 설정합니다. (ANSI)
InternetSetPerSiteCookieDecisionW 지정된 도메인의 쿠키에 대한 결정을 설정합니다. (유니코드)
InternetSetStatusCallback 작업 중에 진행률이 수행되면 WinINet 함수가 호출할 수 있는 콜백 함수를 설정합니다.

InternetTimeFromSystemTime InternetTimeFromSystemTime 함수(wininet.h)는 HTTP 버전 1.0 사양에 따라 날짜와 시간의 형식을 지정합니다.
InternetTimeFromSystemTimeA HTTP 버전 1.0 사양에 따라 날짜 및 시간 형식을 지정합니다. (InternetTimeFromSystemTimeA)
InternetTimeFromSystemTimeW InternetTimeFromSystemTimeW(유니코드) 함수(wininet.h)는 HTTP 버전 1.0 사양에 따라 날짜와 시간의 형식을 지정합니다.
InternetTimeToSystemTime InternetTimeToSystemTime 함수(wininet.h)는 HTTP 시간/날짜 문자열을 SYSTEMTIME 구조체로 변환합니다.
InternetTimeToSystemTimeA HTTP 시간/날짜 문자열을 SYSTEMTIME 구조체로 변환합니다. (InternetTimeToSystemTimeA)
InternetTimeToSystemTimeW InternetTimeToSystemTimeW(유니코드) 함수(wininet.h)는 HTTP 시간/날짜 문자열을 SYSTEMTIME 구조체로 변환합니다.
InternetUnlockRequestFile InternetLockRequestFile을 사용하여 잠긴 파일의 잠금을 해제합니다.
InternetWriteFile 열려 있는 인터넷 파일에 데이터를 씁니다.
PrivacyGetZonePreferenceW PrivacyGetZonePreferenceW(유니코드) 함수(wininet.h)는 지정된 URLZONE 및 PrivacyType에 대한 개인 정보 설정을 검색합니다.
PrivacyGetZonePreferenceW PrivacyGetZonePreferenceW(유니코드) 함수(winineti.h)는 지정된 URLZONE 및 PrivacyType에 대한 개인 정보 설정을 검색합니다.

PrivacySetZonePreferenceW PrivacySetZonePreferenceW(유니코드) 함수(wininet.h)는 지정된 URLZONE 및 PrivacyType에 대한 개인 정보 설정을 설정합니다.
PrivacySetZonePreferenceW PrivacySetZonePreferenceW 함수(winineti.h)는 지정된 URLZONE 및 PrivacyType에 대한 개인 정보 설정을 설정합니다.
ReadUrlCacheEntryStream RetrieveUrlCacheEntryStream 함수를 사용하여 열린 스트림에서 캐시된 데이터를 읽습니다.
ResumeSuspendedDownload ResumeSuspendedDownload 함수는 사용자 인터페이스 대화 상자에서 일시 중단된 요청을 다시 시작합니다.
RetrieveUrlCacheEntryFileA 지정된 URL과 연결된 캐시 항목 파일을 잠급니다. (ANSI)
RetrieveUrlCacheEntryFileW 지정된 URL과 연결된 캐시 항목 파일을 잠급니다. (유니코드)
RetrieveUrlCacheEntryStreamA 캐시 데이터에 액세스하는 가장 효율적이고 구현 독립적인 방법을 제공합니다. (ANSI)
RetrieveUrlCacheEntryStreamW 캐시 데이터에 액세스하는 가장 효율적이고 구현 독립적인 방법을 제공합니다. (유니코드)
SetUrlCacheEntryGroup SetUrlCacheEntryGroup 함수(wininet.h)는 캐시 그룹에 항목을 추가하거나 캐시 그룹에서 항목을 제거합니다.
SetUrlCacheEntryGroupA 캐시 그룹에 항목을 추가하거나 캐시 그룹에서 항목을 제거합니다. (SetUrlCacheEntryGroupA)
SetUrlCacheEntryGroupW SetUrlCacheEntryGroupW(유니코드) 함수(wininet.h)는 캐시 그룹에 항목을 추가하거나 캐시 그룹에서 항목을 제거합니다.

SetUrlCacheEntryInfoA INTERNET_CACHE_ENTRY_INFO 구조체의 지정된 멤버를 설정합니다. (ANSI)
SetUrlCacheEntryInfoW INTERNET_CACHE_ENTRY_INFO 구조체의 지정된 멤버를 설정합니다. (유니코드)
SetUrlCacheGroupAttributeA 지정된 캐시 그룹의 특성 정보를 설정합니다. (ANSI)
SetUrlCacheGroupAttributeW 지정된 캐시 그룹의 특성 정보를 설정합니다. (유니코드)
UnlockUrlCacheEntryFile UnlockUrlCacheEntryFile 함수(wininet.h)는 캐시에서 사용하기 위해 파일을 검색하는 동안 잠긴 캐시 항목의 잠금을 해제합니다.
UnlockUrlCacheEntryFileA 캐시에서 사용하기 위해 파일을 검색하는 동안 잠긴 캐시 항목의 잠금을 해제합니다. (UnlockUrlCacheEntryFileA)
UnlockUrlCacheEntryFileW UnlockUrlCacheEntryFileW(유니코드) 함수(wininet.h)는 캐시에서 사용하기 위해 파일을 검색하는 동안 잠긴 캐시 항목의 잠금을 해제합니다.
UnlockUrlCacheEntryStream RetrieveUrlCacheEntryStream 함수를 사용하여 검색된 스트림을 닫습니다.

인터페이스

IProofOfPossessionCookieInfoManager 소유 증명 쿠키 생성을 지원합니다.
IProofOfPossessionCookieInfoManager2 WebAccount에 대한 소유 증명 쿠키 만들기를 지원합니다.

구조체

AUTO_PROXY_SCRIPT_BUFFER AUTO_PROXY_SCRIPT_BUFFER 구조체는 InternetInitializeAutoProxyDll이 열리는 파일을 식별하는 대신 버퍼의 autoproxy 스크립트를 InternetInitializeAutoProxyDll 에 전달하는 데 사용됩니다.
AutoProxyHelperFunctions AutoProxyHelperFunctions 구조체는 InternetInitializeAutoProxyDll에 전달할 수 있는 프록시 자동 구성 함수의 v 테이블을 만드는 데 사용됩니다.
AutoProxyHelperVtbl AutoProxyHelperVtbl 구조체는 PAC(프록시 AutoConfig) 도우미 함수에 대한 포인터의 v 테이블을 만듭니다. 프록시 자동 구성 도우미 함수의 양식 및 사용에 대한 사양은 PAC(탐색기 프록시 자동 구성) 파일 형식 설명서를 참조하세요.
GOPHER_ATTRIBUTE_TYPE 개체에 대한 단일 Gopher 특성의 관련 정보를 포함합니다.
GOPHER_FIND_DATAA GopherFindFirstFile 및 InternetFindNextFile 함수에서 검색한 정보를 포함합니다. (ANSI)
GOPHER_FIND_DATAW GopherFindFirstFile 및 InternetFindNextFile 함수에서 검색한 정보를 포함합니다. (유니코드)
HTTP_VERSION_INFO 전역 HTTP 버전을 포함합니다.
INTERNET_ASYNC_RESULT 비동기 함수 호출의 결과를 포함합니다. 이 구조체는 InternetStatusCallback과 함께 사용됩니다.
INTERNET_AUTH_NOTIFY_DATA 인증 요청에 대한 알림 데이터를 포함합니다.
INTERNET_BUFFERSA 데이터 및 헤더 정보를 모두 포함합니다. (ANSI)

INTERNET_BUFFERSW 데이터 및 헤더 정보를 모두 포함합니다. (유니코드)
INTERNET_CACHE_CONFIG_INFOA 인터넷 캐시의 구성에 대한 정보를 포함합니다. (ANSI)
INTERNET_CACHE_CONFIG_INFOW 인터넷 캐시의 구성에 대한 정보를 포함합니다. (유니코드)
INTERNET_CACHE_ENTRY_INFOA 인터넷 캐시의 항목에 대한 정보를 포함합니다. (ANSI)
INTERNET_CACHE_ENTRY_INFOW 인터넷 캐시의 항목에 대한 정보를 포함합니다. (유니코드)
INTERNET_CACHE_GROUP_INFOA 특정 캐시 그룹에 대한 정보를 포함합니다. (ANSI)
INTERNET_CACHE_GROUP_INFOW 특정 캐시 그룹에 대한 정보를 포함합니다. (유니코드)
INTERNET_CACHE_TIMESTAMPS 인터넷 캐시에 저장된 리소스에 대한 LastModified 및 만료 시간을 포함합니다.
INTERNET_CERTIFICATE_INFO 서버에서 반환된 인증서 정보를 포함합니다. 이 구조체는 InternetQueryOption 함수에서 사용됩니다.
INTERNET_CONNECTED_INFO 전역 온라인/오프라인 상태를 설정하는 정보를 포함합니다.
INTERNET_COOKIE2 INTERNET_COOKIE2 구조체에는 쿠키의 구성 요소가 포함됩니다. 이 구조체는 InternetGetCookieEx2 및 InternetSetCookieEx2 함수와 함께 사용됩니다.

INTERNET_DIAGNOSTIC_SOCKET_INFO INTERNET_DIAGNOSTIC_SOCKET_INFO 구조체는 INTERNET_OPTION_DIAGNOSTIC_SOCKET_INFO 플래그가 HTTP 요청에 대한 핸들과 함께 전달될 때 InternetQueryOption 함수에 의해 반환됩니다.
INTERNET_PER_CONN_OPTION_LISTA 특정 인터넷 연결에 대한 옵션 목록을 포함합니다. (ANSI)
INTERNET_PER_CONN_OPTION_LISTW 특정 인터넷 연결에 대한 옵션 목록을 포함합니다. (유니코드)
INTERNET_PER_CONN_OPTIONA 옵션의 값을 포함합니다. (ANSI)
INTERNET_PER_CONN_OPTIONW 옵션의 값을 포함합니다. (유니코드)
INTERNET_PROXY_INFO InternetOpen 함수 호출에서 가져온 핸들에 대한 프록시 정보를 가져오거나 설정하기 위해 INTERNET_OPTION_PROXY 값과 함께 제공되는 정보를 포함합니다.
INTERNET_VERSION_INFO 서버의 HTTP 버전 번호를 포함합니다. 이 구조체는 INTERNET_OPTION_VERSION 플래그를 InternetQueryOption 함수에 전달할 때 사용됩니다.
InternetCookieHistory InternetCookieHistory 구조에는 쿠키 기록이 포함됩니다.
ProofOfPossessionCookieInfo
URL_COMPONENTSA URL의 구성 요소를 포함합니다. 이 구조체는 InternetCrackUrl 및 InternetCreateUrl 함수와 함께 사용됩니다. (ANSI)
URL_COMPONENTSW URL의 구성 요소를 포함합니다. 이 구조체는 InternetCrackUrl 및 InternetCreateUrl 함수와 함께 사용됩니다. (유니코드)

Windows Machine Learning

아티클2023. 03. 16.

Windows Machine Learning 기술 개요입니다.

Windows Machine Learning을 개발하려면 다음 헤더가 필요합니다.

- winml.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows Machine Learning

열거형

WINML_BINDING_TYPE 다양한 유형의 WinML 바인딩을 지정합니다.
WINML_FEATURE_TYPE 다양한 유형의 WinML 기능을 지정합니다.
WINML_RUNTIME_TYPE 다양한 유형의 WinML 런타임을 지정합니다.
WINML_TENSOR_DATA_TYPE WinML 텐서의 다양한 데이터 형식을 지정합니다.

Functions

BindValue 입력/출력을 지정된 모델에 바인딩합니다.
지우기 모델에 대한 바인딩을 지웁니다.

CreateEvaluationContext WinML 평가 컨텍스트 개체를 만듭니다.
CreateRuntime WinML 런타임을 만듭니다.
EnumerateMetadata 모델의 메타데이터를 가져옵니다.
EnumerateModelInputs WinML 모델 입력을 열거합니다.
EnumerateModelOutputs WinML 모델 출력을 열거합니다.
EvaluateModel WinML 모델을 평가합니다.
GetDescription WinML 모델 설명을 검색합니다.
GetValueByName 특정 바인딩 이름에 대한 입력/출력 설명을 반환합니다.
LoadModel WinML 모델을 로드합니다.

인터페이스

IWinMLEvaluationContext WinML 모델에 입력 및 출력을 바인딩하는 컨텍스트를 나타냅니다.

IWinMLModel 해당 메타데이터가 있는 Windows Machine Learning 모델을 나타냅니다. 에는 모델 설명(이름, 작성자, 버전 관리 등)과 예상된 입력 및 출력이 포함됩니다.
IWinMLRuntime WinML 모델을 로드하고 평가할 런타임을 나타냅니다.
IWinMLRuntimeFactory 모델 로드 및 평가를 위해 WinML 런타임을 만드는 팩터리를 나타냅니다.

구조체

WINML_BINDING_DESC WinML 바인딩에 대한 설명을 포함합니다.
WINML_IMAGE_BINDING_DESC 형식 이미지의 바인딩에 대한 속성을 포함합니다.
WINML_IMAGE_VARIABLE_DESC 이미지 변수 설명에 대한 속성을 포함합니다.
WINML_MAP_BINDING_DESC 형식 맵의 바인딩에 대한 속성을 포함합니다.
WINML_MAP_VARIABLE_DESC 지도 변수의 설명 속성을 포함합니다.
WINML_MODEL_DESC 모델의 설명 속성을 포함합니다.
WINML_RESOURCE_BINDING_DESC 리소스 바인딩의 설명 속성을 포함합니다.

WINML_SEQUENCE_BINDING_DESC 시퀀스 바인딩의 설명 속성을 포함합니다.
WINML_SEQUENCE_VARIABLE_DESC 시퀀스 변수의 설명 속성을 포함합니다.
WINML_TENSOR_BINDING_DESC 텐서 바인딩의 설명 속성을 포함합니다.
WINML_TENSOR_VARIABLE_DESC 텐서 변수의 설명 속성을 포함합니다.
WINML_VARIABLE_DESC 변수의 설명 속성을 포함합니다.

MI(Windows Management Infrastructure)

아티클2023. 03. 09.

MI(Windows Management Infrastructure) 기술에 대한 개요입니다.

MI(Windows Management Infrastructure)를 개발하려면 다음 헤더가 필요합니다.

- mi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- MI(Windows Management Infrastructure)

열거형

MI_CallbackMode WriteError 및 PromptUser 함수에 대한 CIM 확장의 콜백 모드를 정의합니다.
MI_CancellationReason 작업이 취소되는 이유를 시스템에 알리기 위해 작업 취소 요청에 전달할 값입니다. 서비스가 종료되는 경우 이러한 값 중 하나를 공급자에게 전달할 수도 있습니다.
MI_DestinationOptions_ImpersonationType DCOM 프로토콜 처리기에서 서버에서 가장을 수행하는 방법을 지정하는 데 사용됩니다.
MI_ErrorCategory 이 열거형은 CIM 확장에 대한 오류 범주를 정의합니다.
MI_LocaleType 로캘을 설정하고 가져오는 경우 로캘 유형이 필요합니다.
MI_OperationCallback_ResponseType MI_CallbackMode MI_CALLBACKMODE_INQUIRE 경우 콜백에서 이러한 값 중 하나를 사용할 수 있습니다.

MI_PromptType CIM 확장에 대한 프롬프트 형식을 정의합니다.
MI_ProviderArchitecture 이 열거형은 서버에서 사용되는 WMI 공급자 아키텍처를 정의합니다.
MI_Result 함수 반환 코드를 정의합니다.
MI_SubscriptionDeliveryType 밀어넣기 또는 끌어오기 구독 배달 유형을 구분합니다. DCOM 프로토콜을 사용할 때는 지원되지 않습니다.
MI_Type 이러한 값은 CIM 데이터 형식에 대한 한정자, 속성, 참조, 매개 변수 및 메서드 반환 값의 데이터 형식을 지정합니다.

Functions

MI_Application_Close MI_Application_Initialize 호출을 통해 초기화된 관리 인프라 클라이언트 API를 초기화합니다.
MI_Application_InitializeV1 MI(관리 인프라) 클라이언트 API 호출을 수행할 수 있도록 애플리케이션을 초기화합니다.
MI_Application_NewClass MI_ClassDecl 구조체에서 MI_Class 만듭니다.
MI_Application_NewDeserializer serialize된 개체를 클래스 또는 인스턴스로 다시 변환하는 데 사용할 수 있는 역직렬 변환기 개체를 만듭니다.
MI_Application_NewDestinationOptions MI_Application_NewSession 함수와 함께 사용할 수 있는 MI_DestinationOptions 개체를 만듭니다.

MI_Application_NewHostedProvider 호스트된 공급자를 로컬 컴퓨터의 WMI 엔진에 등록합니다.
MI_Application_NewInstance 인스턴스가 필요한 다양한 MI 작업 API에 전달할 새 MI_Instance 개체를 만듭니다.
MI_Application_NewInstanceFromClass 클래스 개체를 기반으로 새 MI_Instance 개체를 만듭니다.
MI_Application_NewOperationOptions MI_Session 개체의 연산 함수와 함께 사용할 수 있는 MI_OperationOptions 개체를 만듭니다.
MI_Application_NewParameterSet 새 매개 변수 집합을 만듭니다.
MI_Application_NewSerializer 그런 다음 인스턴스와 클래스를 다양한 형식으로 serialize하는 데 사용할 수 있는 serializer 개체를 검색합니다.
MI_Application_NewSession 단일 대상에 대한 작업 집합에 대한 연결을 공유하는 데 사용되는 세션을 만듭니다.
MI_Application_NewSubscriptionDeliveryOptions 특정 프로토콜에 대한 구독 작업을 수행하는 데 필요한 구성을 나타내는 MI_SubscriptionDeliveryOptions 개체를 만듭니다.
MI_Class_Clone MI_Class 개체를 복제합니다.
MI_Class_Delete MI_Class 개체를 삭제합니다.
MI_Class_GetClassName 지정된 클래스의 클래스 이름을 가져옵니다.
MI_Class_GetClassQualifierSet 지정된 클래스 개체와 연결된 한정자 집합을 가져옵니다.

MI_Class_GetElement 클래스에서 지정된 명명된 요소의 모든 세부 정보를 가져옵니다.
MI_Class_GetElementAt 요소 인덱스 기반 클래스 요소의 세부 정보를 가져옵니다.
MI_Class_GetElementCount 클래스의 요소 수를 가져옵니다.
MI_Class_GetMethod 메서드 이름을 기반으로 메서드의 세부 정보를 가져옵니다.
MI_Class_GetMethodAt 메서드 인덱스 기반 메서드의 세부 정보를 가져옵니다.
MI_Class_GetMethodCount 클래스의 메서드 수를 가져옵니다.
MI_Class_GetNameSpace 지정된 클래스의 네임스페이스 이름을 가져옵니다.
MI_Class_GetParentClass 지정된 클래스의 부모 클래스를 가져옵니다.
MI_Class_GetParentClassName 지정된 클래스의 부모 클래스 이름을 가져옵니다.
MI_Class_GetServerName 지정된 클래스에서 서버의 이름을 가져옵니다.
MI_Context_Canceled 작업이 취소되었는지 여부를 확인합니다. 이 함수는 예약되어 있습니다. 대신 MI_Context_RegisterCancel 함수를 사용합니다.
MI_Context_ConstructInstance 스택 또는 구조체의 멤버로 MI 클래스 인스턴스를 초기화합니다.

MI_Context_ConstructParameters 공급자는 이 함수를 호출하여 매개 변수의 인스턴스를 초기화합니다.
MI_Context_GetCustomOption 클라이언트에서 설정한 옵션을 검색합니다.
MI_Context_GetCustomOptionAt 클라이언트에서 설정한 특정 인덱스에서 옵션을 검색합니다.
MI_Context_GetCustomOptionCount 공급자가 사용할 수 있는 사용자 지정 옵션 수를 가져옵니다.
MI_Context_GetLocale 클라이언트가 작업에 대해 지정한 요청된 로캘 정보를 검색합니다.
MI_Context_GetLocalSession 공급자가 공급자를 호스트하는 로컬 서버에 대해 CIM 작업을 수행할 수 있는 로컬 세션(MI_Session)을 가져옵니다.
MI_Context_GetNumberOption 작업 이름에 따라 클라이언트가 설정하는 숫자 옵션을 가져옵니다.
MI_Context_GetStringOption 작업 이름에 따라 클라이언트가 설정하는 문자열 옵션을 가져옵니다.
MI_Context_NewDynamicInstance 클래스의 새 동적 인스턴스(클래스 선언이 없는 약한 형식의 인스턴스)를 만듭니다.
MI_Context_NewInstance 클래스 선언이 지정된 클래스의 새 인스턴스를 만듭니다.
MI_Context_NewParameters 메서드 선언이 지정된 경우 메서드의 새 인스턴스를 만듭니다.
MI_Context_PostCimError 요청에 대한 응답으로 반환 코드와 오류 메시지(CIM_Error 개체 형식)를 서버에 게시합니다.

MI_Context_PostError 공급자는 요청에 대한 응답으로 클라이언트에 반환 코드를 게시하기 위해 이 함수를 호출합니다.
MI_Context_PostIndication 구독 작업 요청에 대한 응답으로 표시 결과를 서버에 게시합니다.
MI_Context_PostInstance 요청에 대한 응답으로 인스턴스를 클라이언트(서버를 통해)에 다시 게시합니다.
MI_Context_PostResult 요청에 대한 응답으로 최종 종료 결과 코드를 클라이언트(서버를 통해)에 다시 게시합니다.
MI_Context_PromptUser 작업을 계속할지 아니면 취소할지를 쿼리하는 프롬프트 메시지를 클라이언트에 보냅니다.
MI_Context_RefuseUnload 공급자 인프라에 공급자를 언로드하지 말라고 지시합니다.
MI_Context_RegisterCancel 작업이 취소될 때 호출되는 콜백을 등록합니다.
MI_Context_RequestUnload 모듈 또는 공급자를 언로드하도록 요청합니다.
MI_Context_SetStringOption 컨텍스트별 옵션을 설정합니다.
MI_Context_ShouldContinue 클라이언트를 쿼리하여 작업을 계속해야 하는지 확인합니다. (MI_Context_ShouldContinue)
MI_Context_ShouldProcess 클라이언트를 쿼리하여 작업을 계속해야 하는지 확인합니다. (MI_Context_ShouldProcess)
MI_Context_WriteCimError CIM(정보 제공) 오류 인스턴스를 클라이언트에 보냅니다.

MI_Context_WriteDebug 클라이언트에 디버그 메시지를 보냅니다.
MI_Context_WriteError 오류 코드 및 오류 메시지를 클라이언트에 보냅니다.
MI_Context_WriteMessage 클라이언트에 작업 메시지를 보냅니다.
MI_Context_WriteProgress 클라이언트에 진행률 메시지를 보냅니다.
MI_Context_WriteStreamParameter 메서드 호출을 위해 스트리밍된 매개 변수 데이터를 클라이언트에 보냅니다.
MI_Context_WriteVerbose 클라이언트에 자세한 메시지를 씁니다.
MI_Context_WriteWarning 클라이언트에 경고 메시지를 씁니다.
MI_Deserializer_Class_GetClassName serialize된 클래스 버퍼에서 클래스 이름을 가져옵니다.
MI_Deserializer_Class_GetParentClassName serialize된 클래스 버퍼에서 부모 클래스 이름을 가져옵니다.
MI_Deserializer_ClassObjectNeeded 역직렬화하는 동안 요청된 클래스 개체를 제공하는 데 사용됩니다.
MI_Deserializer_Close 역직렬 변환기 개체를 닫고 역직렬 변환기 내에 있는 연결된 메모리를 삭제합니다.
MI_Deserializer_DeserializeClass 직렬화된 버퍼를 MI_Class 개체로 역직렬화합니다.

MI_Deserializer_DeserializeInstance 직렬화된 버퍼를 MI_Instance 개체로 역직렬화합니다.
MI_Deserializer_Instance_GetClassName serialize된 인스턴스와 연결된 클래스 이름을 가져옵니다.
MI_DestinationOptions_AddDestinationCredentials 대상과 통신하기 위한 자격 증명을 설정합니다.
MI_DestinationOptions_AddProxyCredentials 프록시에 대해 인증할 자격 증명을 추가합니다.
MI_DestinationOptions_Clone MI_DestinationOptions 구조체의 복사본을 만듭니다.
MI_DestinationOptions_Delete MI_Application_NewDestinationOptions 또는 MI_DestinationOptions_Clone 함수를 사용하여 만든 대상 옵션 구조를 삭제합니다.
MI_DestinationOptions_GetCertCACheck 서버 인증서 CA 확인 값을 가져옵니다.
MI_DestinationOptions_GetCertCNCheck 서버 인증서 CN 확인 값을 가져옵니다.
MI_DestinationOptions_GetCertRevocationCheck 서버 인증서의 해지 확인 값을 가져옵니다.
MI_DestinationOptions_GetCredentialsAt 지정된 인덱스에서 자격 증명을 가져옵니다.
MI_DestinationOptions_GetCredentialsCount 이전에 추가된 자격 증명의 수를 가져옵니다. (MI_DestinationOptions_GetCredentialsCount)
MI_DestinationOptions_GetCredentialsPasswordAt 지정된 인덱스 기반의 자격 증명 암호를 가져옵니다.

MI_DestinationOptions_GetDataLocale 사용자가 설정한 데이터 로캘(UI 로캘과 반대)을 가져옵니다.
MI_DestinationOptions_GetDestinationPort 전송에 대한 기본 포트를 가져옵니다.
MI_DestinationOptions_GetEncodePortInSPN 포트의 서비스 주체 이름 인코딩 값을 가져옵니다.
MI_DestinationOptions_GetHttpUrlPrefix HTTP URL 접두사 가져오기
MI_DestinationOptions_GetImpersonationType 가장 형식을 가져옵니다.
MI_DestinationOptions_GetMaxEnvelopeSize 서버로 전송되거나 서버에서 클라이언트가 받은 패킷의 최대 크기를 가져옵니다.
MI_DestinationOptions_GetNumber 이전에 추가한 사용자 지정 번호 옵션을 가져옵니다. (MI_DestinationOptions_GetNumber)
MI_DestinationOptions_GetOption 옵션 이름을 기반으로 이전에 추가된 옵션 값을 가져옵니다. (MI_DestinationOptions_GetOption)
MI_DestinationOptions_GetOptionAt 지정된 인덱스 기반의 이전에 추가된 옵션 값을 가져옵니다. (MI_DestinationOptions_GetOptionAt)
MI_DestinationOptions_GetOptionCount 이전에 추가한 옵션 수를 가져옵니다. (MI_DestinationOptions_GetOptionCount)
MI_DestinationOptions_GetPacketEncoding 이전에 설정한 패킷 인코딩 설정을 가져옵니다.
MI_DestinationOptions_GetPacketIntegrity 패킷 무결성 설정을 가져옵니다.

MI_DestinationOptions_GetPacketPrivacy 패킷 개인 정보(암호화) 설정을 가져옵니다.
MI_DestinationOptions_GetProxyType 사용자가 설정한 프록시 유형을 가져옵니다.
MI_DestinationOptions_GetString 이전에 추가한 사용자 지정 문자열 옵션을 가져옵니다.
MI_DestinationOptions_GetTimeout 기본 옵션 시간 제한 값을 가져옵니다.
MI_DestinationOptions_GetTransport 클라이언트가 추가한 전송 설정을 가져옵니다.
MI_DestinationOptions_GetUILocale 사용자가 설정한 사용자 인터페이스 로캘을 가져옵니다.
MI_DestinationOptions_SetCertCACheck SSL 전송에 대한 CA 인증서 검사를 사용하거나 사용하지 않도록 설정합니다.
MI_DestinationOptions_SetCertCNCheck SSL 전송이 사용될 때 인증서 CN 검사를 사용하거나 사용하지 않도록 설정합니다.
MI_DestinationOptions_SetCertRevocationCheck SSL을 통해 통신할 때 인증서 해지를 사용하거나 사용하지 않도록 설정합니다.
MI_DestinationOptions_SetDataLocale 작업에 사용할 기본 데이터 로캘을 설정합니다.
MI_DestinationOptions_SetDestinationPort 대상과 통신하는 데 사용할 포트를 설정합니다.
MI_DestinationOptions_SetEncodePortInSPN 원격 컴퓨터에 대한 연결을 설정할 때 서비스 주체 이름에서 포트 번호의 인코딩을 사용하거나 사용하지 않도록 설정합니다.

MI_DestinationOptions_SetHttpUrlPrefix HTTP 및 HTTPS를 통해 이동하는 전송에 대한 기본 HTTP URL 접두사 설정
MI_DestinationOptions_SetImpersonationType 가장 유형을 설정합니다.
MI_DestinationOptions_SetMaxEnvelopeSize 전송의 최대 패킷 크기를 설정합니다.
MI_DestinationOptions_SetNumber 사용자 지정 숫자 옵션 값을 설정합니다.
MI_DestinationOptions_SetPacketEncoding 특정 프로토콜 핸들에 대한 인코딩 메커니즘을 설정합니다.
MI_DestinationOptions_SetPacketIntegrity 프로토콜 연결의 패킷 무결성(서명)을 사용하거나 사용하지 않도록 설정합니다.
MI_DestinationOptions_SetPacketPrivacy 패킷 개인 정보 보호(암호화)를 사용하거나 사용하지 않도록 설정합니다.
MI_DestinationOptions_SetProxyType 프록시를 통해 대상과 통신할 때 사용할 프록시 설정 유형을 설정합니다.
MI_DestinationOptions_SetString 사용자 지정 문자열 옵션을 설정합니다. (MI_DestinationOptions_SetString)
MI_DestinationOptions_SetTimeout 기본 옵션 시간 제한 값을 설정합니다.
MI_DestinationOptions_SetTransport 대상 컴퓨터와 통신하는 데 사용할 전송을 설정합니다.
MI_DestinationOptions_SetUILocale 작업에 대한 기본 UI 로캘을 설정합니다.

MI_Filter_Evaluate 공급자는 지정된 필터에 대해 인스턴스를 평가하기 위해 이 함수를 호출합니다.
MI_Filter_GetExpression 필터 언어 및 식을 가져옵니다.
MI_HostedProvider_Close MI_Application_NewHostedProvider 반환된 호스트된 공급자 핸들을 닫습니다.
MI_HostedProvider_GetApplication 호스트된 공급자 핸들이 만들어진 최상위 애플리케이션 핸들을 가져옵니다.
MI_Instance_AddElement 동적 인스턴스에 새 속성을 추가합니다(런타임에 스키마를 확장할 수 있는 동적 인스턴스에서만 지원됨).
MI_Instance_ClearElement 명명된 요소(CIM 속성)의 값을 지우고 NULL로 설정합니다.
MI_Instance_ClearElementAt 지정된 인덱스에 있는 요소(CIM 속성)의 값을 지우고 NULL로 설정합니다.
MI_Instance_Clone 힙에 지정된 인스턴스의 복사본을 만듭니다.
MI_Instance_Delete 힙에서 만들거나 다른 인스턴스에서 복제된 인스턴스를 삭제합니다.
MI_Instance_Destruct 스택 또는 구조체의 멤버로 만든 인스턴스를 삭제합니다.
MI_Instance_GetClass 인스턴스와 연결된 MI_Class 가져옵니다.
MI_Instance_GetClassName 지정된 인스턴스의 클래스 이름을 가져옵니다.

MI_Instance_GetElement 명명된 요소(CIM 속성)의 값을 가져옵니다.
MI_Instance_GetElementAt 지정된 인덱스에서 요소(CIM 속성)의 값을 가져옵니다.
MI_Instance_GetElementCount 인스턴스의 요소 수를 가져옵니다.
MI_Instance_GetNameSpace 지정된 인스턴스의 네임스페이스 이름을 가져옵니다.
MI_Instance_GetServerName 지정된 인스턴스에서 서버 이름을 가져옵니다.
MI_Instance_IsA 인스턴스 자체가 classDecl에서 제공하는 클래스의 인스턴스인지 여부를 확인합니다.
MI_Instance_Normalize MI_Instance_ExFT 구조를 구문 분석한 다음 MI_InstanceFT 함수 테이블을 검색합니다.
MI_Instance_SetElement 지정된 인스턴스에서 지정된 이름을 사용하여 요소의 값을 설정합니다.
MI_Instance_SetElementAt 인스턴스의 지정된 인덱스에서 요소 값을 설정합니다.
MI_Instance_SetNameSpace 지정된 인스턴스의 네임스페이스 이름을 설정합니다.
MI_Instance_SetServerName 지정된 인스턴스의 서버 이름을 설정합니다.
MI_Operation_Cancel 실행 중인 작업을 취소합니다.

MI_Operation_Close 작업 핸들을 닫습니다.
MI_Operation_GetClass 클래스 작업에 대한 동기 결과를 가져옵니다.
MI_Operation_GetIndication 구독에서 동기 결과를 가져옵니다.
MI_Operation_GetInstance 인스턴스 작업에 대한 동기 결과를 가져옵니다.
MI_Operation_GetSession 작업과 연결된 세션을 가져옵니다.
MI_OperationOptions_Clone MI_OperationOptions 구조체의 복사본을 만듭니다.
MI_OperationOptions_Delete 옵션 집합 및 관련 메모리를 삭제합니다.
MI_OperationOptions_DisableChannel MI_Context_WriteMessage 사용하여 지정된 채널에 대한 로깅을 사용하지 않도록 설정합니다.
MI_OperationOptions_EnableChannel MI_Context_WriteMessage 사용하여 지정된 채널에 대한 로깅을 사용하도록 설정합니다.
MI_OperationOptions_GetEnabledChannels 이전에 사용하도록 설정된 채널의 목록을 가져옵니다.
MI_OperationOptions_GetNumber 이전에 추가한 사용자 지정 번호 옵션을 가져옵니다. (MI_OperationOptions_GetNumber)
MI_OperationOptions_GetOption 옵션 이름에 따라 이전에 추가된 옵션 값을 가져옵니다. (MI_OperationOptions_GetOption)

MI_OperationOptions_GetOptionAt 지정된 인덱스 기반 이전에 추가된 옵션 값을 가져옵니다. (MI_OperationOptions_GetOptionAt)
MI_OperationOptions_GetOptionCount 이전에 추가한 옵션 수를 가져옵니다. (MI_OperationOptions_GetOptionCount)
MI_OperationOptions_GetPromptUserMode 공급자의 MI_Context_PromptUser 호출에 응답하는 방법을 서버에 알려주는 값을 가져옵니다. (MI_OperationOptions_GetPromptUserMode)
MI_OperationOptions_GetPromptUserRegularMode 공급자의 MI_Context_PromptUser 호출에 응답하는 방법을 서버에 알려주는 값을 가져옵니다. (MI_OperationOptions_GetPromptUserRegularMode)
MI_OperationOptions_GetProviderArchitecture 작업에 대한 공급자 아키텍처를 가져옵니다.
MI_OperationOptions_GetResourceUri 작업에 사용되는 리소스 URI를 가져옵니다.
MI_OperationOptions_GetResourceUriPrefix 작업에 사용되는 리소스 URI 접두사입니다.
MI_OperationOptions_GetString 사용자 지정 문자열 옵션을 가져옵니다.
MI_OperationOptions_GetTimeout 작업 시간 제한 값을 가져옵니다.
MI_OperationOptions_GetUseMachineID 작업 요청에서 컴퓨터 식별 정보를 사용할지 여부를 나타내는 값을 가져옵니다.
MI_OperationOptions_GetWriteErrorMode 오류 보고 모드를 설정합니다. (MI_OperationOptions_GetWriteErrorMode)
MI_OperationOptions_SetCustomOption 작업에 대한 사용자 지정 옵션을 설정합니다.

MI_OperationOptions_SetNumber 사용자 지정 숫자 옵션 값을 설정합니다.
MI_OperationOptions_SetPromptUserMode MI_Context_PromptUser 함수에 대한 공급자의 호출에 응답하는 방법을 서버에 알려주는 값을 설정합니다. (MI_OperationOptions_SetPromptUserMode)
MI_OperationOptions_SetPromptUserRegularMode MI_Context_PromptUser 함수에 대한 공급자의 호출에 응답하는 방법을 서버에 알려주는 값을 설정합니다. (MI_OperationOptions_SetPromptUserRegularMode)
MI_OperationOptions_SetProviderArchitecture 작업에 대한 공급자 아키텍처를 설정합니다.
MI_OperationOptions_SetResourceUri 작업에 사용할 리소스 URI를 설정합니다.
MI_OperationOptions_SetResourceUriPrefix 작업에 사용할 리소스 URI 접두사 설정
MI_OperationOptions_SetString 사용자 지정 문자열 옵션을 설정합니다. (MI_OperationOptions_SetString)
MI_OperationOptions_SetTimeout 특정 작업에 대한 작업 시간 제한을 설정합니다.
MI_OperationOptions_SetUseMachineID 작업 요청에서 컴퓨터 ID 정보 전송을 사용하거나 사용하지 않도록 설정합니다.
MI_OperationOptions_SetWriteErrorMode 오류 보고 모드를 설정합니다. (MI_OperationOptions_SetWriteErrorMode)
MI_ParameterSet_GetMethodReturnType 지정된 매개 변수 집합에 대한 메서드 반환 형식 및 한정자 집합을 가져옵니다.
MI_ParameterSet_GetParameter 매개 변수 이름을 기반으로 메서드의 매개 변수 정보를 가져옵니다.

MI_ParameterSet_GetParameterAt 지정된 인덱스에서 메서드의 매개 변수 정보를 가져옵니다.
MI_ParameterSet_GetParameterCount 메서드의 매개 변수 집합에 있는 매개 변수 수를 가져옵니다.
MI_PropertySet_AddElement 속성 목록에 이름을 추가합니다.
MI_PropertySet_Clear 속성 목록에서 모든 이름을 제거합니다. 그 후 개수는 0입니다. 이렇게 하면 속성 목록을 다시 사용할 수 있습니다(소멸 및 재구성할 필요 없이).
MI_PropertySet_Clone 힙에 지정된 속성 집합의 복사본을 만듭니다.
MI_PropertySet_ContainsElement 속성 목록에 지정된 속성 이름이 포함되어 있는지 여부를 확인합니다.
MI_PropertySet_Delete 힙에서 생성된 지정된 속성 목록을 삭제합니다.
MI_PropertySet_Destruct 스택에서 생성된 지정된 속성 목록을 삭제합니다.
MI_PropertySet_GetElementAt 지정된 인덱스에서 속성 집합의 요소를 가져옵니다.
MI_PropertySet_GetElementCount 지정된 속성 집합의 요소 수를 가져옵니다.
MI_QualifierSet_GetQualifier 지정된 한정자 이름을 기준으로 한정자 정보를 가져옵니다.
MI_QualifierSet_GetQualifierAt 지정된 인덱스에서 한정자를 가져옵니다.

MI_QualifierSet_GetQualifierCount 한정자 집합의 한정자 수를 가져옵니다.
MI_Serializer_Close serializer 개체를 닫고 연결된 내부 메모리를 해제합니다.
MI_Serializer_SerializeClass serializer를 만들 때 지정된 형식으로 버퍼로 MI_Class 직렬화합니다. 클래스와 모든 부모 클래스가 직렬화되는지 아니면 자식 클래스만 제어하도록 플래그에 옵션을 전달할 수 있습니다.
MI_Serializer_SerializeInstance serializer를 만들 때 지정된 형식으로 버퍼에 MI_Instance 직렬화합니다. 클래스가 버퍼와 인스턴스로 직렬화되는지 제어하기 위해 플래그에 옵션을 전달할 수 있습니다.
MI_Server_GetSystemName 서버의 시스템 이름을 가져옵니다.
MI_Server_GetVersion 공급자를 생성할 때 사용되는 MI_VERSION 매크로의 값을 가져옵니다.
MI_Session_AssociatorInstances 특정 키 인스턴스와 연결된 인스턴스를 찾습니다.
MI_Session_Close 세션을 닫고 연결된 모든 메모리를 해제합니다.
MI_Session_CreateInstance 세션이 나타내는 서버에 인스턴스를 만듭니다.
MI_Session_DeleteInstance 세션이 나타내는 서버의 인스턴스를 삭제합니다.
MI_Session_EnumerateClasses 지정된 세션의 클래스를 열거합니다.
MI_Session_EnumerateInstances 클래스와 연결된 모든 인스턴스(세션이 나타내는 서버)를 열거합니다.

MI_Session_GetApplication 지정된 세션을 만드는 데 사용된 애플리케이션 핸들을 가져옵니다.
MI_Session_GetClass 특정 클래스 이름을 기반으로 MI_Class 선언을 가져옵니다.
MI_Session_GetInstance 세션이 나타내는 서버에서 지정된 인스턴스를 가져옵니다.
MI_Session_Invoke 공급자에서 메서드를 호출합니다.
MI_Session_ModifyInstance 세션이 나타내는 서버의 기존 인스턴스를 업데이트.
MI_Session_QueryInstances 쿼리 식을 기반으로 인스턴스 집합을 쿼리합니다.
MI_Session_ReferenceInstances 지정된 키 인스턴스를 참조하는 연결 개체를 찾습니다.
MI_Session_Subscribe 세션이 나타내는 서버의 표시를 구독합니다.
MI_Session_TestConnection 세션이 나타내는 서버와 통신하여 연결을 테스트하여 응답 여부를 확인합니다.
MI_SubscriptionDeliveryOptions_AddDeliveryCredentials 푸시 표시 결과를 제공하기 위해 클라이언트에 다시 연결할 때 사용할 배달 자격 증명에 대한 구독 옵션을 설정합니다.
MI_SubscriptionDeliveryOptions_Clone MI_SubscriptionDeliveryOptions 구조체의 복사본을 만듭니다.
MI_SubscriptionDeliveryOptions_Delete 지정된 구독 배달 옵션 구조를 삭제합니다.

MI_SubscriptionDeliveryOptions_GetBookmark 이전에 설정된 구독 책갈피를 가져옵니다.
MI_SubscriptionDeliveryOptions_GetCredentialsAt 지정된 인덱스 기반 이전에 추가된 자격 증명을 가져옵니다.
MI_SubscriptionDeliveryOptions_GetCredentialsCount 이전에 추가한 자격 증명 수를 가져옵니다. (MI_SubscriptionDeliveryOptions_GetCredentialsCount)
MI_SubscriptionDeliveryOptions_GetCredentialsPasswordAt 지정된 인덱스 기반 이전에 추가된 자격 증명 암호를 가져옵니다.
MI_SubscriptionDeliveryOptions_GetDateTime 이전에 설정된 날짜/시간 옵션을 가져옵니다.
MI_SubscriptionDeliveryOptions_GetDeliveryDestination 이전에 설정된 구독 배달 대상을 가져옵니다.
MI_SubscriptionDeliveryOptions_GetDeliveryPortNumber 이전에 설정된 배달 포트 번호를 가져옵니다.
MI_SubscriptionDeliveryOptions_GetDeliveryRetryAttempts 배달 재시도 횟수를 가져옵니다.
MI_SubscriptionDeliveryOptions_GetDeliveryRetryInterval 배달 재시도 간격을 가져오고 배달을 다시 시도하기 전에 대기할 시간을 가져옵니다.
MI_SubscriptionDeliveryOptions_GetExpirationTime 배달 만료 값(타임스탬프 또는 간격으로 표현할 수 있는)을 가져옵니다.
MI_SubscriptionDeliveryOptions_GetHeartbeatInterval 배달 하트비트 간격을 가져옵니다.
MI_SubscriptionDeliveryOptions_GetInterval 지정된 옵션의 배달 간격을 가져옵니다.

MI_SubscriptionDeliveryOptions_GetMaximumLatency 클라이언트에 전달하기 전에 서버에서 결과를 보유할 최대 시간을 가져옵니다.
MI_SubscriptionDeliveryOptions_GetNumber 명명된 숫자 옵션의 값을 가져옵니다.
MI_SubscriptionDeliveryOptions_GetOption 명명된 옵션의 값을 가져옵니다.
MI_SubscriptionDeliveryOptions_GetOptionAt 지정된 인덱스에서 옵션을 가져옵니다.
MI_SubscriptionDeliveryOptions_GetOptionCount 이전에 설정된 옵션의 수를 가져옵니다.
MI_SubscriptionDeliveryOptions_GetString 명명된 문자열 옵션의 값을 가져옵니다.
MI_SubscriptionDeliveryOptions_SetBookmark 구독 표시 배달에 대한 책갈피를 설정합니다.
MI_SubscriptionDeliveryOptions_SetDateTime 명명된 DateTime 옵션의 값을 설정합니다.
MI_SubscriptionDeliveryOptions_SetDeliveryDestination 표시가 전달될 대상 엔드포인트를 설정합니다.
MI_SubscriptionDeliveryOptions_SetDeliveryPortNumber 구독 배달 포트 번호를 설정합니다.
MI_SubscriptionDeliveryOptions_SetDeliveryRetryAttempts 푸시 배달 구독이 결과를 배달하려고 시도하는 횟수를 설정합니다.
MI_SubscriptionDeliveryOptions_SetDeliveryRetryInterval 푸시 배달을 위한 구독에 대한 배달 재시도 간격을 설정합니다.

MI_SubscriptionDeliveryOptions_SetExpirationTime 구독 만료 시간(구독이 종료되는 경우)을 설정합니다.
MI_SubscriptionDeliveryOptions_SetHeartbeatInterval 하트비트 간격을 설정합니다.
MI_SubscriptionDeliveryOptions_SetInterval 명명된 간격 옵션의 값을 설정합니다.
MI_SubscriptionDeliveryOptions_SetMaximumLatency 클라이언트에 전달하기 전에 서버에서 결과를 보유할 최대 시간을 설정합니다.
MI_SubscriptionDeliveryOptions_SetNumber 전용 함수에서 다루지 않는 명명된 숫자 옵션의 값을 설정합니다.
MI_SubscriptionDeliveryOptions_SetString 전용 함수에서 다루지 않는 명명된 문자열 옵션의 값을 설정합니다.
MI_Utilities_CimErrorFromErrorCode 운영 체제별 오류 코드를 CIM 오류 인스턴스에 매핑합니다.
MI_Utilities_MapErrorToMiErrorCategory 운영 체제별 오류 코드를 오류 범주에 매핑합니다.

구조체

MI_Application 초기화된 인프라를 나타냅니다.
MI_ApplicationFT MI_Application 구조에 사용되는 지원 구조체입니다. "MI_Application_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.

MI_Array 배열을 나타내는 일반화된 형식입니다. 데이터 요소 형식을 제외한 모든 배열의 크기가 같으므로 일반화할 수 있습니다.
MI_ArrayField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ArrayField)
MI_BooleanA MI_Boolean 형식의 배열을 나타냅니다.
MI_BooleanAField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_BooleanAField)
MI_BooleanField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_BooleanField)
MI_Char16A MI_Char16 형식의 배열을 나타냅니다. (MI_Char16A)
MI_Char16AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Char16AField)
MI_Char16Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Char16Field)
MI_Class 인스턴스의 스키마를 나타냅니다.
MI_ClassDecl 이 구조체는 클래스 선언을 간략하게 설명합니다. 클래스 이름 및 계층, 속성, 한정자 및 메서드를 포함합니다.
MI_ClassFT MI_Class 구조에 사용되는 지원 구조체입니다. "MI_Class_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.

MI_ClientFT_V1 클라이언트 함수 테이블.
MI_ConstBooleanA MI_ConstBoolean 형식의 배열을 나타냅니다.
MI_ConstBooleanAField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstBooleanAField)
MI_ConstBooleanField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstBooleanField)
MI_ConstChar16A MI_Char16 형식의 배열을 나타냅니다. (MI_ConstChar16A)
MI_ConstChar16AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstChar16AField)
MI_ConstChar16Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstChar16Field)
MI_ConstDatetimeA MI_Datatime 형식의 배열을 나타냅니다.
MI_ConstDatetimeAField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstDatetimeAField)
MI_ConstDatetimeField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstDatetimeField)
MI_ConstInstanceA MI_Instance 형식의 배열을 나타냅니다. (MI_ConstInstanceA)
MI_ConstInstanceAField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstInstanceAField)

MI_ConstInstanceField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstInstanceField)
MI_ConstReal32A MI_Real32 형식의 배열을 나타냅니다. (MI_ConstReal32A)
MI_ConstReal32AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstReal32AField)
MI_ConstReal32Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstReal32Field)
MI_ConstReal64A MI_Real64 형식의 배열을 나타냅니다. (MI_ConstReal64A)
MI_ConstReal64AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstReal64AField)
MI_ConstReal64Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstReal64Field)
MI_ConstReferenceA MI_Instance 형식의 배열을 나타냅니다. (MI_ConstReferenceA)
MI_ConstReferenceAField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstReferenceAField)
MI_ConstReferenceField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstReferenceField)
MI_ConstSint16A MI_Sint16 형식의 배열을 나타냅니다. (MI_ConstSint16A)
MI_ConstSint16AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstSint16AField)

MI_ConstSint16Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstSint16Field)
MI_ConstSint32A MI_Sint32 형식의 배열을 나타냅니다. (MI_ConstSint32A)
MI_ConstSint32AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstSint32AField)
MI_ConstSint32Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstSint32Field)
MI_ConstSint64A MI_Sint64 형식의 배열을 나타냅니다. (MI_ConstSint64A)
MI_ConstSint64AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstSint64AField)
MI_ConstSint64Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstSint64Field)
MI_ConstSint8A MI_Sint8 형식의 배열을 나타냅니다. (MI_ConstSint8A)
MI_ConstSint8AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstSint8AField)
MI_ConstSint8Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstSint8Field)
MI_ConstStringA MI_Char 형식의 배열을 나타냅니다.
MI_ConstStringAField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstStringAField)

MI_ConstStringField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstStringField)
MI_ConstUint16A MI_Uint16A 형식의 배열을 나타냅니다.
MI_ConstUint16AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstUint16AField)
MI_ConstUint16Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstUint16Field)
MI_ConstUint32A MI_Uint32 형식의 배열을 나타냅니다. (MI_ConstUint32A)
MI_ConstUint32AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstUint32AField)
MI_ConstUint32Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstUint32Field)
MI_ConstUint64A MI_Uint64 형식의 배열을 나타냅니다. (MI_ConstUint64A)
MI_ConstUint64AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstUint64AField)
MI_ConstUint64Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstUint64Field)
MI_ConstUint8A MI_Uint8 형식의 배열을 나타냅니다. (MI_ConstUint8A)
MI_ConstUint8AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstUint8AField)

MI_ConstUint8Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstUint8Field)
MI_Context 공급자가 수행해야 하는 작업에 대한 컨텍스트를 보유합니다.
MI_ContextFT MI_Context 구조에 사용되는 지원 구조체입니다. "MI_Context_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_Datetime MI_Timestamp 및 MI_Interval 통합을 나타냅니다.
MI_DatetimeA MI_Datetime 형식의 배열을 나타냅니다.
MI_DatetimeAField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_DatetimeAField)
MI_DatetimeField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_DatetimeField)
MI_Deserializer MI_Application_NewDeserializer 만든 역직렬화 개체입니다. 개체 자체는 내부적으로 사용되므로 수동으로 사용하거나 변경하면 안 됩니다.
MI_DeserializerFT MI_ClientFT_V1 구조에 사용되는 지원 구조체입니다. "MI_Deserializer_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_DestinationOptions 대상 옵션 집합을 나타냅니다. 대상 옵션은 작업이 서버와 통신하는 방식을 정의하는 구성 집합입니다.
MI_DestinationOptionsFT MI_DestinationOptions 구조에 사용되는 지원 구조체입니다. "MI_DestinationOptions_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.

MI_FeatureDecl MI_PropertyDeclMI_ParameterDecland MI_MethodDecl 구조체에 공통적인 속성을 포함합니다.
MI_Filter 함수 테이블 MI_FilterFT 대한 참조를 포함합니다.
MI_FilterFT MI_Filter 구조에 사용되는 지원 구조체입니다. "MI_Filter_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_HostedProvider 클라이언트 애플리케이션에서 공급자의 호스팅을 나타냅니다.
MI_HostedProviderFT MI_HostedProvider 구조에 사용되는 지원 구조체입니다. "MI_HostedProvider_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_Instance 이 구조체는 CIM 인스턴스를 나타냅니다. 이 개체는 직접 액세스하면 안 됩니다. 대신 MI_Instance_* 함수를 사용해야 합니다.
MI_InstanceA MI_Instance 구조체의 배열을 나타냅니다.
MI_InstanceAField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_InstanceAField)
MI_InstanceExFT MI_InstanceFT 구조를 확장합니다.
MI_InstanceField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_InstanceField)
MI_InstanceFT MI_Instance 구조에 사용되는 지원 구조체입니다. 이러한 구조를 조작하려면 이름 접두사 MI_Instance_ 함수를 사용합니다.

MI_Interval MI_Interval 시간 간격을 나타냅니다.
MI_MethodDecl CIM 메서드를 나타냅니다.
MI_Module 공급자가 생성한 이 개체에는 공급자 관리자가 이 모듈 내에서 공급자를 관리하는 데 필요한 모든 데이터가 포함됩니다.
MI_ObjectDecl MI_ClassDecl 및 MI_PropertyDecl 구조에 공통적인 속성을 포함합니다.
MI_Operation 단일 작업 실행을 나타냅니다. 이 개체는 작업에 대한 작업을 수행하기 위한 내부 함수 테이블을 보유합니다.
MI_OperationCallbacks 작업을 수행하기 위한 모든 콜백 함수 포인터를 보유하는 구조체입니다.
MI_OperationFT MI_Operation 구조체에 사용되는 지원 구조체입니다. "MI_Operation_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_OperationOptions 작업 옵션 집합을 나타냅니다.
MI_OperationOptionsFT MI_OperationOptions 구조에 사용되는 지원 구조체입니다. "MI_OperationOptions_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_ParameterDecl CIM 메서드 매개 변수를 나타냅니다.
MI_ParameterSet 클래스 정의의 메서드 매개 변수를 보유합니다.

MI_ParameterSetFT MI_ParameterSet 구조에 사용되는 지원 구조체입니다. 이러한 구조를 조작하려면 이름 접두사 MI_ParameterSet_ 함수를 사용합니다.
MI_PropertyDecl 클래스 선언의 클래스 속성(요소)을 나타냅니다.
MI_PropertySet 속성 이름 집합을 구현합니다.
MI_PropertySetFT MI_PropertySet 구조체에 사용되는 지원 구조체입니다. "MI_PropertySet_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_ProviderFT MI_ClassDecl 및 MI_Module 구조에 사용되는 지원 구조입니다.
MI_Qualifier CIM 한정자를 나타냅니다.
MI_QualifierDecl CIM 한정자 선언을 나타냅니다.
MI_QualifierSet 개발자가 클래스 정의의 한정자를 볼 수 있습니다.
MI_QualifierSetFT MI_QualifierSet 구조에 사용되는 지원 구조체입니다. "MI_QualifierSet_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_Real32A MI_Real32 형식의 배열을 나타냅니다. (MI_Real32A)
MI_Real32AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Real32AField)

MI_Real32Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Real32Field)
MI_Real64A MI_Real64 형식의 배열을 나타냅니다. (MI_Real64A)
MI_Real64AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Real64AField)
MI_Real64Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Real64Field)
MI_ReferenceA MI_Instance 형식에 대한 포인터 배열을 나타냅니다.
MI_ReferenceAField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ReferenceAField)
MI_ReferenceField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ReferenceField)
MI_SchemaDecl CIM 클래스 및 CIM 한정자 선언을 포함하는 CIM 스키마의 스키마 개체를 나타냅니다.
MI_Serializer 특정 serialization 기술에 연결된 개체입니다.
MI_SerializerFT MI_ClientFT_V1 구조에 사용되는 지원 구조체입니다. "MI_Serializer_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_Server 이 구조체는 _Context, Instance, PropertySet 및 Filter의 모든 형식에 대한 기본 함수 테이블을 정의합니다.

MI_ServerFT MI_Server 구조에 사용되는 지원 구조체입니다. "MI_Server_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_Session 대상과 연결되고 연결된 자격 증명 및 옵션 집합이 있는 개체입니다. .
MI_SessionCallbacks 로깅 및 오류 메시지를 처리하는 콜백 함수 포인터용 컨테이너입니다.
MI_SessionFT 세션 개체의 모든 작업에 대한 함수 테이블입니다.
MI_Sint16A MI_Sint16 형식의 배열을 나타냅니다. (MI_Sint16A)
MI_Sint16AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Sint16AField)
MI_Sint16Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Sint16Field)
MI_Sint32A MI_Sint32 형식의 배열을 나타냅니다. (MI_Sint32A)
MI_Sint32AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Sint32AField)
MI_Sint32Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Sint32Field)
MI_Sint64A MI_Sint64 형식의 배열을 나타냅니다. (MI_Sint64A)
MI_Sint64AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Sint64AField)

MI_Sint64Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Sint64Field)
MI_Sint8A MI_Sint8 형식의 배열을 나타냅니다. (MI_Sint8A)
MI_Sint8AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Sint8AField)
MI_Sint8Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Sint8Field)
MI_StringA null로 끝나는 MI_Char* 문자열에 대한 포인터 배열을 나타냅니다.
MI_StringAField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_StringAField)
MI_StringField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_StringField)
MI_SubscriptionDeliveryOptions 구독 옵션 개체는 구독 작업에 전달하는 데 사용되는 구성 옵션을 저장합니다.
MI_SubscriptionDeliveryOptionsFT MI_SubscriptionDeliveryOptions 구조에 사용되는 지원 구조체입니다. "MI_SubscriptionDeliveryOptions_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_Timestamp MI_Timestamp 타임스탬프 또는 특정 시점을 지정합니다.
MI_Uint16A MI_Uint16 형식의 배열을 나타냅니다.
MI_Uint16AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Uint16AField)

MI_Uint16Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Uint16Field)
MI_Uint32A MI_Uint32 형식의 배열을 나타냅니다. (MI_Uint32A)
MI_Uint32AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Uint32AField)
MI_Uint32Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Uint32Field)
MI_Uint64A MI_Uint64 형식의 배열을 나타냅니다. (MI_Uint64A)
MI_Uint64AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Uint64AField)
MI_Uint64Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Uint64Field)
MI_Uint8A MI_Uint8 형식의 배열을 나타냅니다. (MI_Uint8A)
MI_Uint8AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Uint8AField)
MI_Uint8Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Uint8Field)
MI_UserCredentials 사용자의 자격 증명입니다. 여기에는 인증 유형과 사용자 이름 및 암호 또는 인증서 지문이 포함됩니다.
MI_UsernamePasswordCreds 구독 작업에 사용되는 사용자 이름/암호 조합입니다.

MI_UtilitiesFT MI_ClientFT_V1 구조에 사용되는 지원 구조체입니다. 이름 접두사 "MI_Utilities_"와 함께 함수를 사용하여 이러한 구조를 조작합니다.
MI_Value 모든 CIM 데이터 형식의 통합입니다.

윈도우즈 관리 도구 (Windows Management Instrumentation)

Windows Management Instrumentation 기술에 대한 개요입니다.

Windows Management Instrumentation을 개발하려면 다음 헤더가 필요합니다.

- chptrarr.h
- chstrarr.h
- chstring.h
- frquery.h
- instance.h
- methodco.h
- objpath.h
- refptrco.h
- thrdbase.h
- wbemdisp.h
- wbemglue.h
- wbemprov.h
- wbemtime.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 관리 계측
- WMI 정보

클래스

테이블 확장

CFrameworkQuery CFrameworkQuery 클래스는 쿼리를 구문 분석하고 조작하는 데 사용됩니다. 여기에는 특정 속성 값을 반환하는 메서드와 포함된 특정 속성을 확인하기 위해 쿼리를 테스트하는 메서드가 포함되어 있습니다.
CHPtrArray CHPtrArray 클래스는 WMI 공급자 프레임워크의 일부입니다. CHPtrArray는 WMI 쿼리의 공급자 처리에 사용되는 배열 포인터 관리를 위한 유틸리티 인터페이스입니다.
CHString

다음 표에서는 CHString 메서드를 나열합니다.

CHStringArray

다음 표에서는 CHStringArray 메서드 및 연산자를 나열합니다.

CInstance

CInstance 클래스는 WMI 공급자 프레임워크에서 지원하는 인스턴스에 대해 정의된 속성 값을 검색하고 업데이트하는 데 사용됩니다. 또한 CInstance 클래스는 공급자 프레임워크의 CInstance 인터페이스 구현에 대한 액세스를 제공합니다.

CObjectPathParser

원격 컴퓨터 이름, 네임스페이스 및 클래스를 포함할 수 있는 WMI 경로를 구문 분석합니다. 이 개체는 사용하지 않는 것이 좋습니다. 대신 IWbemPath COM 인터페이스를 사용합니다.

CThreadBase

CThreadBase 클래스는 WMI 공급자 프레임워크에 대한 내부 스레드 안전 메커니즘을 제공하는 기본 클래스입니다. 이 클래스는 내부적으로 호출됩니다.

CWbemGlueFactory

CWbemGlueFactory 클래스는 WMI 공급자 프레임워크의 일부입니다. Provider Framework는 이 인터페이스의 메서드를 내부적으로 구현하여 공급자에 대한 클래스의 새 인스턴스를 만듭니다.

CWbemProviderGlue

CWbemProviderGlue는 WMI(Windows Management Instrumentation) API의 COM(구성 요소 개체 모델) 인터페이스를 공급자 클래스에서 파생된 클래스와 연결하고 공급자가 서로를 쿼리하는 데 사용할 메서드를 제공합니다.

MethodContext

MethodContext 클래스는 공급자가 IWbemClassObject 정보를 얻거나 설정하는 데 사용되는 구조체에 대한 포인터입니다. WMI는 포인터를 기반으로 하는 기능을 구현하지 않습니다.

공급자

Provider 클래스는 프레임워크 공급자가 지원하는 클래스 또는 클래스의 기본 클래스입니다.

TRefPointerCollection

TRefPointerCollection 클래스는 개체에 대한 포인터를 수집하는 컨테이너 클래스입니다. 이러한 포인터는 열거할 수 있습니다.

WBEMTime

WBEMTime 클래스는 다양한 Windows 및 ANSI C 런타임 형식 간의 변환을 용이하게 합니다. 자세한 내용은 WBEMTimeSpan 클래스 메서드도 참조하세요.
WBEMTimeSpan WBEMTimeSpan 클래스는 시간 범위를 나노초 단위로 유지합니다.

열거 목록

테이블 확장

CIMTYPE_ENUMERATION 다른 CIM 데이터 형식을 지정하는 값을 정의합니다.
ObjectParserFlags CObjectPathParser에 대한 생성자에서 사용하는 플래그입니다.
tag_WBEM_GET_KEY_FLAGS 텍스트의 형식을 제어하는 플래그를 포함합니다.
tag_WBEM_GET_TEXT_FLAGS 텍스트가 반환되는 방법을 제어하는 플래그를 포함합니다.
tag_WBEM_PATH_CREATE_FLAG 허용되는 경로의 형식을 지정하는 플래그를 포함합니다.
tag_WBEM_PATH_STATUS_FLAG 경로를 설명하는 데 사용되는 플래그를 포함합니다.
WBEM_BACKUP_RESTORE_FLAGS IWbemBackupRestore::Restore 메서드 및 IWbemBackupRestoreEx::Restore 메서드에 사용되는 플래그를 포함합니다.
WBEM_CHANGE_FLAG_TYPE 메서드 매개 변수 플래그를 포함합니다.

WBEM_COMPARISON_FLAG IWbemClassObject::CompareTo 메서드를 사용할 때 수행할 비교를 정의하는 플래그를 포함합니다.
WBEM_COMPILER_OPTIONS IMofCompiler 메서드에 대한 옵션 플래그를 포함합니다.
WBEM_CONDITION_FLAG_TYPE IWbemClassObject::GetNames 메서드와 함께 사용되는 플래그를 포함합니다.
WBEM_CONNECT_OPTIONS IWbemLocator::ConnectServer 메서드에 대한 플래그를 포함합니다.
WBEM_FLAVOR_TYPE 한정자 버전을 나열합니다.
WBEM_GENERIC_FLAG_TYPE 범용 메서드 매개 변수 플래그를 포함합니다.
WBEM_GENUS_TYPE 클래스와 인스턴스를 구분하는 데 사용되는 상수가 포함되어 있습니다.
WBEM_LIMITS 일부 제한 값을 정의합니다.
WBEM_QUERY_FLAG_TYPE 쿼리 또는 열거자를 정의하는 데 사용되는 플래그를 포함합니다.
WBEM_REFRESHER_FLAGS 새로 고침 메서드의 동작을 수정하는 플래그를 포함합니다.
WBEM_SECURITY_FLAGS 보안 액세스 수준을 설정하는 데 사용되는 플래그를 포함합니다.
WBEM_STATUS_TYPE 비동기 작업의 상태를 설명합니다.
WBEM_TEXT_FLAG_TYPE

IWbemClassObject::GetObjectText 메서드의 실행을 제어하는 플래그를 포함합니다.
WBEM_TIMEOUT_TYPE IEnumWbemClassObject::Next 메서드의 시간 제한을 지정하는 데 사용되는 값을 포함합니다.
WBEM_UNSECAPP_FLAG_TYPE IWbemUnsecuredApartment::CreateSinkStub 메서드를 사용할 때 콜백에 대한 액세스 검사를 제어하는 데 사용됩니다.
WbemAuthenticationLevelEnum 보안 인증 수준을 정의합니다.
WbemChangeFlagEnum 클래스 또는 인스턴스에 대한 쓰기 작업을 수행하는 방법을 정의합니다.
WbemCimtypeEnum 속성 값의 유효한 CIM 형식을 정의합니다.
WbemComparisonFlagEnum 개체 비교에 대한 설정을 정의하고 SWbemObject.CompareTo_ 사용됩니다.
WbemConnectOptionsEnum 원격 컴퓨터에서 WMI에 대한 연결이 실패할 때 SWbemLocator.ConnectServer 메서드 호출에서 매개 변수로 사용되는 보안 플래그를 정의합니다.
WbemErrorEnum WMI 호출에 대한 스크립팅 API에서 반환할 수 있는 오류를 정의합니다.
WbemFlagEnum SWbemServices.ExecQuery, SWbemServices.ExecQueryAsync, SWbemServices.SubclassesOf 및 SWbemServices.InstancesOf에서 사용되는 상수 정의
WbemImpersonationLevelEnum 보안 가장 수준을 정의합니다. 이러한 상수는 SWbemSecurity와 함께 사용됩니다.
WbemObjectTextFormatEnum SWbemObjectEx.GetText_ 사용할 유효한 개체 텍스트 형식을 정의합니다.

WbemPrivilegeEnum 권한을 정의합니다. 이러한 상수는 SWbemSecurity와 함께 사용되어 일부 작업에 필요한 권한을 부여합니다.
WbemQueryFlagEnum 호출에서 반환되는 개체 수를 결정하는 열거형 또는 쿼리의 깊이를 정의합니다.
WBEMSTATUS WMI API의 메서드에서 반환된 오류 및 상태 코드를 포함합니다.
WbemTextFlagEnum 생성된 개체 텍스트의 내용을 정의하고 SWbemObject.GetObjectText_ 사용됩니다.
WbemTimeout 제한 시간 상수 정의 이 상수는 SWbemEventSource.NextEvent에서 사용됩니다.
WMI_OBJ_TEXT SWbemObjectEx.GetText_ 사용할 유효한 개체 텍스트 형식을 정의합니다.
WMIQ_ANALYSIS_TYPE GetAnalysis 메서드를 사용하여 수행할 분석 유형을 지정하는 데 사용되는 상수를 포함합니다.
WMIQ_ASSOCQ_FLAGS 쿼리의 기능을 나타내는 플래그를 포함합니다.
WMIQ_RPN_TOKEN_FLAGS GetAnalysis 메서드에 사용되는 쿼리 토큰을 설명하는 플래그를 포함합니다.

Functions

테이블 확장

AccessCheck

AccessCheck 메서드는 이벤트 공급자에 의해 구현되며 소비자가 wszQuery에 지정된 이벤트를 구독할 때 WMI(Windows Management Instrumentation)에서 호출됩니다.
추가 Add 메서드는 배열의 끝에 새 요소를 추가하여 배열을 하나씩 늘입니다.
추가 Add 메서드는 컬렉션에 대한 참조를 추가합니다.
AddEnum IWbemConfigureRefresher::AddEnum 메서드는 요청된 새로 고침에 열거자를 추가합니다.
AddObjectByPath IWbemConfigureRefresher::AddObjectByPath 메서드는 개체 경로를 지정하여 개체를 새로 고침에 추가합니다.
AddObjectByTemplate IWbemConfigureRefresher::AddObjectByTemplate 메서드를 사용하면 IWbemClassObject 인스턴스 템플릿을 지정하여 새로 고칠 개체를 새로 고칠 개체를 추가할 수 있습니다.
AddObjects IWbemHiPerfEnum::AddObjects 메서드는 제공된 인스턴스 개체를 열거자에 추가합니다.
AddRefresher IWbemConfigureRefresher::AddRefresher 메서드는 새로 고침에 새로 고침을 추가합니다.
AllocSysString AllocSysString 메서드는 Automation과 호환되는 새 BSTR 문자열을 할당합니다. 그런 다음 종료 NULL 문자를 포함하여 CHStringstring의 내용을 복사합니다.
AllPropertiesAreRequired AllPropertiesAreRequired 메서드는 인스턴스에 대한 모든 속성이 요청되었는지 여부를 나타냅니다.
레코드 추가 Append 메서드는 지정된 배열의 끝에 다른 배열의 내용을 추가합니다.
백업 IWbemBackupRestore::Backup 메서드는 정적 리포지토리의 내용을 별도의 파일에 백업합니다.

BeginEnum BeginEnum 메서드는 컬렉션 열거를 시작합니다.
BeginEnumeration 열거형을 열거형의 시작 부분으로 다시 설정합니다.
BeginEnumeration IWbemContext::BeginEnumeration 메서드는 개체에 있는 모든 컨텍스트 값의 열거형을 다시 설정합니다.
BeginEnumeration 개체의 모든 한정자가 열거되기 전에 IWbemQualifierSet::BeginEnumeration 메서드가 다시 설정됩니다.
BeginMethodEnumeration IWbemClassObject::BeginMethodEnumeration 메서드 호출을 사용하여 개체에 사용할 수 있는 메서드의 열거를 시작합니다.
beginRead BeginRead 메서드는 공급자가 WMI 공급자 프레임워크에 빌드된 경우 데이터 읽기 작업의 시작을 표시하여 스레드 보안을 제공합니다. CThreadBase는 내부적으로 호출됩니다.
BeginWrite BeginWrite 메서드는 공급자가 WMI 공급자 프레임워크에 빌드된 경우 데이터 쓰기 작업의 시작을 표시하여 스레드 보안을 제공합니다. CThreadBase는 내부적으로 호출됩니다.
CancelAsyncCall IWbemServices::CancelAsyncCall 메서드는 원래 비동기 메서드에 전달된 IWbemObjectSink 포인터를 기반으로 현재 보류 중인 모든 비동기 호출을 취소합니다.
CancelQuery 논리 이벤트 소비자가 Windows Management를 사용하여 관련 이벤트 쿼리 필터를 취소할 때마다 IWbemEventProviderQuerySink::CancelQuery 메서드를 호출합니다.
CHPtrArray CHPtrArray 메서드 생성자는 CHPtrArray의 새 인스턴스를 초기화합니다.
CHString CHString::CHString 함수는 지정된 데이터를 사용하여 새 CHString 개체를 초기화합니다.

CHString 지정된 데이터를 사용하여 새 CHString 개체를 초기화합니다. (오버로드 2/6)
CHString 지정된 데이터를 사용하여 새 CHString 개체를 초기화합니다. (오버로드 4/6)
CHString 지정된 데이터를 사용하여 새 CHString 개체를 초기화합니다. (오버로드 5/6)
CHString 지정된 데이터를 사용하여 새 CHString 개체를 초기화합니다. (오버로드 1/6)
CHString 지정된 데이터를 사용하여 새 CHString 개체를 초기화합니다. (오버로드 6/6)
CHString 지정된 데이터를 사용하여 새 CHString 개체를 초기화합니다. (오버로드 3/6)
CHStringArray CHStringArray 생성자는 빈 CHStringArray 배열을 생성합니다.
지우기 Clear 메서드는 WBEMTime 개체의 시간을 잘못된 시간으로 설정합니다.
지우기 Clear 메서드는 시간 범위를 INVALID_TIME 설정합니다.
클론 IEnumWbemClassObject::Clone 메서드는 전체 열거자의 논리적 복사본을 만들어 열거형에서 현재 위치를 유지합니다.
클론 IWbemClassObject::Clone 메서드는 현재 개체의 전체 복제본인 새 개체를 반환합니다. 새 개체의 COM 참조 수는 1입니다.
클론

IWbemContext::Clone 메서드는 현재 IWbemContext 개체의 논리적 복사본을 만듭니다. 이 메서드는 IWbemContext 개체가 거의 동일한 많은 호출을 수행해야 하는 경우에 유용할 수 있습니다.
CObjectPathParser 전체 개체 경로가 필요한 CObjectPathParser 개체의 인스턴스를 생성하고 초기화합니다. 이 개체는 사용하지 않는 것이 좋습니다. 대신 IWbemPath COM 인터페이스를 사용합니다.
함께 합치다 Collate 메서드는 wcscoll 함수를 사용하여 CHString 문자열을 다른 문자열과 비교합니다.
커밋 Commit 메서드는 WMI에 인스턴스를 보내는 데 사용됩니다. 이 메서드는 도우미 함수이며 재정의해서는 안 됩니다.
커밋 Commit 메서드는 현재 인스턴스를 WMI로 반환합니다.
비교 Compare 메서드는 wcscmp 함수를 사용하여 이 CHString 문자열을 다른 문자열과 비교합니다.
CompareNoCase CompareNoCase 메서드는 _wcsicmp 함수를 사용하여 CHString 문자열을 다른 문자열과 비교합니다.
CompareTo IWbemClassObject::CompareTo 메서드는 개체를 다른 Windows Management 개체와 비교합니다. 이 비교 프로세스에는 특정 제약 조건이 있습니다.
CompileBuffer IMofCompiler::CompileBuffer 메서드는 이진 MOF 데이터가 포함된 버퍼 또는 ASCII 형식의 텍스트 버퍼를 컴파일합니다.
CompileFile IMofCompiler::CompileFile 메서드는 MOF 파일(이진 MOF 포함)을 컴파일하고 WMI 리포지토리에 정보를 저장합니다.
ConnectServer IWbemLocator::ConnectServer 메서드는 dCOM을 통해 strNetworkResource 매개 변수에 지정된 컴퓨터의 WMI 네임스페이스에 대한 연결을 만듭니다.

복사 Copy 메서드는 지정된 배열의 요소를 다른 배열의 요소로 덮어씁니다.
CreateBMOF IMofCompiler::CreateBMOF 메서드는 이진 MOF 파일을 만듭니다.
CreateClassEnum IWbemServices::CreateClassEnum 메서드는 선택 조건을 충족하는 모든 클래스에 대한 열거자를 반환합니다.
CreateClassEnumAsync IWbemServices::CreateClassEnumAsync 메서드는 클래스 공급자가 지원하는 모든 클래스의 열거형을 반환합니다.
CreateClassPart 경로의 클래스 또는 키 부분을 초기화합니다.
CreateInstanceEnum IWbemServices::CreateInstanceEnum 메서드는 사용자가 지정한 선택 조건에 따라 지정된 클래스의 인스턴스를 반환하는 열거자를 만듭니다.
CreateInstanceEnumAsync IWbemServices::CreateInstanceEnumAsync 메서드는 사용자가 지정한 선택 조건에 따라 지정된 클래스의 인스턴스를 비동기적으로 반환하는 열거자를 만듭니다.
CreateNewInstance CreateNewInstance 메서드는 새 CInstance 개체를 할당하고 포인터를 반환합니다.
CreateObjectStub CreateObjectStub 메서드는 Windows Management에서 비동기 호출을 수신하는 데 도움이 되도록 개체 전달자 싱크를 만듭니다.
CreateRefreshableEnum 새로 고칠 수 있는 새 열거형을 만듭니다.
CreateRefreshableObject 새로 고칠 수 있는 인스턴스 개체를 요청합니다.

CreateRefresher 새로 고침을 만듭니다.
CreateSinkStub CreateSinkStub 메서드는 IUnsecuredApartment::CreateObjectStub와 유사하며 개체 전달자 싱크를 만들고 Windows Management에서 비동기 호출을 수신하기 위한 액세스 검사를 수행합니다.
CThreadBase CThreadBase::CThreadBase 생성자는 CThreadBase의 새 인스턴스를 초기화합니다. CThreadBase는 내부적으로 호출됩니다.
CWbemGlueFactory CWbemGlueFactory::CWbemGlueFactory 생성자는 CWbemGlueFactory의 새 인스턴스를 초기화합니다. 이 클래스는 WMI 공급자 프레임워크의 일부입니다. (?? 0CWbemGlueFactory@@QEAA@XZ)
CWbemGlueFactory CWbemGlueFactory::CWbemGlueFactory 생성자는 CWbemGlueFactory의 새 인스턴스를 초기화합니다. 이 클래스는 WMI 공급자 프레임워크의 일부입니다. (오버로드 1/2)
Delete IWbemClassObject::D elete 메서드는 CIM 클래스 정의 및 모든 해당 한정자에서 지정된 속성을 삭제합니다.
Delete IWbemQualifierSet::D elete 메서드는 지정된 한정자를 이름으로 삭제합니다.
DeleteAll IWbemContext::D eleteAll 메서드는 현재 개체에서 명명된 모든 컨텍스트 값을 제거하여 개체를 비웁니다.
DeleteClass IWbemServices::D eleteClass 메서드는 현재 네임스페이스에서 지정된 클래스를 삭제합니다.
DeleteClassAsync IWbemServices::D eleteClassAsync 메서드는 현재 네임스페이스에서 지정된 클래스를 삭제합니다.
DeleteClassPart IWbemPath::D eleteClassPart 메서드는 경로의 클래스 부분을 삭제합니다.

DeleteInstance 인스턴스를 삭제하기 위해 WMI에서 DeleteInstance 메서드를 호출합니다. (오버로드 2/2)
DeleteInstance 인스턴스를 삭제하기 위해 WMI에서 DeleteInstance 메서드를 호출합니다. (오버로드 1/2)
DeleteInstance IWbemServices::D eleteInstance 메서드는 현재 네임스페이스에서 기존 클래스의 인스턴스를 삭제합니다.
DeleteInstanceAsync IWbemServices::D eleteInstanceAsync 메서드는 현재 네임스페이스에서 기존 클래스의 인스턴스를 비동기적으로 삭제합니다. 작업의 확인 또는 실패는 호출자가 구현한 IWbemObjectSink 인터페이스를 통해 보고됩니다.
DeleteMethod IWbemClassObject::D eleteMethod 메서드를 사용하여 메서드를 삭제합니다. 이 호출은 현재 개체가 CIM 클래스 정의인 경우에만 지원됩니다. CIM 인스턴스를 가리키는 IWbemClassObject 포인터에서는 메서드 조작을 사용할 수 없습니다.
DeleteValue IWbemContext::D eleteValue 메서드는 IWbemContext::SetValue에서 만든 명명된 컨텍스트 값을 삭제합니다.
ElementAt CHStringArray::ElementAt(int)(chstrarr.h) 메서드는 배열 내의 요소 포인터에 대한 임시 참조를 반환합니다.
ElementAt CHStringArray::ElementAt(chstrarr.h) 메서드는 배열 내의 요소 포인터에 대한 임시 참조를 반환합니다.
빈 IWbemQuery::Empty 메서드는 파서가 보유하고 있는 메모리를 해제합니다.
빈 Empty 메서드는 CHString 개체를 빈 문자열로 만들고 메모리를 적절하게 해제합니다.
빈 Empty 메서드는 컬렉션에서 모든 항목을 제거합니다.

EndEnum EndEnum 메서드는 컬렉션의 열거형을 종료합니다.
EndEnumeration IWbemClassObject::EndEnumeration 메서드는 IWbemClassObject::BeginEnumeration으로 시작된 열거형 시퀀스를 종료합니다.
EndEnumeration IWbemContext::EndEnumeration 메서드는 IWbemContext::BeginEnumeration으로 시작하는 열거형 시퀀스를 종료합니다. 이 호출은 필요하지 않지만 열거형과 연결된 시스템 리소스를 가능한 한 빨리 해제합니다.
EndEnumeration IWbemQualifierSet::BeginEnumeration 및 IWbemQualifierSet::Next로 시작된 열거형을 종료하려는 경우 IWbemQualifierSet::EndEnumeration 메서드를 호출합니다.
EndMethodEnumeration IWbemClassObject::EndMethodEnumeration 메서드는 IWbemClassObject::BeginMethodEnumeration으로 시작된 메서드 열거형 시퀀스를 종료하는 데 사용됩니다.
EndRead EndRead 메서드는 공급자가 WMI 공급자 프레임워크에 빌드된 경우 데이터 읽기 작업의 끝을 표시하여 스레드 보안을 제공합니다. CThreadBase는 내부적으로 호출됩니다.
EndWrite EndWrite 메서드는 공급자가 WMI 공급자 프레임워크에 빌드된 경우 데이터 쓰기 작업의 끝을 표시하여 스레드 보안을 제공합니다. CThreadBase는 내부적으로 호출됩니다.
EnumerateInstances EnumerateInstances 메서드는 프레임워크 공급자 클래스의 모든 인스턴스를 검색하기 위해 WMI에서 호출됩니다.
ExecMethod ExecMethod 메서드는 클래스 또는 인스턴스에서 메서드를 호출하기 위해 WMI에 의해 호출됩니다. (오버로드 1/2)
ExecMethod ExecMethod 메서드는 클래스 또는 인스턴스에서 메서드를 호출하기 위해 WMI에 의해 호출됩니다. (오버로드 2/2)

ExecMethod CIM 개체에서 내보낸 메서드를 실행합니다.
ExecMethodAsync CIM 개체에서 내보낸 메서드를 비동기적으로 실행합니다.
ExecNotificationQuery IWbemServices::ExecNotificationQuery 메서드는 이벤트를 수신하는 쿼리를 실행합니다. 호출은 즉시 반환되며, 사용자는 이벤트가 도착할 때 반환된 열거자를 폴링할 수 있습니다. 반환된 열거자를 해제하면 쿼리가 취소됩니다.
ExecNotificationQueryAsync 이벤트 알림을 중지하기 위해 CancelAsyncCall이 호출될 때까지 지정된 응답 처리기에 이벤트가 제공된다는 점을 제외하고 IWbemServices::ExecNotificationQueryAsync 메서드는 IWbemServices::ExecNotificationQuery와 동일한 작업을 수행합니다.
ExecQuery ExecQuery 메서드는 WQL(WMI 쿼리 언어) 쿼리를 처리하기 위해 WMI에서 호출됩니다.
ExecQuery IWbemServices::ExecQuery 메서드는 쿼리를 실행하여 개체를 검색합니다.
ExecQueryAsync IWbemServices::ExecQueryAsync 메서드는 쿼리를 실행하여 개체를 비동기적으로 검색합니다.
찾기 Find 메서드는 문자열에서 부분 문자열의 첫 번째 일치 항목을 검색합니다.
찾기 CHString::Find 메서드는 문자열에서 부분 문자열의 첫 번째 일치 항목을 검색합니다.
FindConsumer FindConsumer 함수는 WMI가 이벤트를 보낼 수 있는 싱크 개체를 찾아 반환합니다.
FindOneOf FindOneOf 메서드는 lpszCharSet에 포함된 문자와 일치하는 첫 번째 문자에 대한 문자열을 검색합니다.

플러시 Flush 메서드는 공급자 프레임워크에서 공급자가 사용 중인 불필요한 메모리를 모두 삭제하기 위해 호출됩니다.
형식 CHString::Format 메서드는 일련의 문자와 값을 CHString에 서식을 지정하고 저장합니다.
형식 CHString::Format 메서드는 일련의 문자와 값을 CHString 문자열에 서식을 지정하고 저장합니다.
FormatMessageW CHString::FormatMessageW(유니코드) 메서드는 메시지 문자열의 형식을 지정합니다.
FormatMessageW CHString::FormatMessageW(유니코드) 메서드는 메시지 문자열의 형식을 지정합니다.
FormatV FormatV 메서드는 형식이 지정된 문자열과 변수 인수 목록을 CHString 문자열에 씁니다.
FrameworkLoginDLL FrameworkLoginDLL 메서드는 공급자 서버를 로드할 수 있는지 여부를 확인하기 위해 DLL_PROCESS_ATTACH 값을 DllMain으로 보낼 때 호출됩니다. (오버로드 1/2)
FrameworkLoginDLL FrameworkLoginDLL 메서드는 공급자 서버를 로드할 수 있는지 여부를 확인하기 위해 DLL_PROCESS_ATTACH 값을 DllMain으로 보낼 때 호출됩니다. (오버로드 2/2)
FrameworkLogoffDLL FrameworkLogoffDLL 메서드는 DllCanUnloadNow에 의해 호출되어 공급자 서버가 사용되고 있지 않고 언로드될 수 있는지 여부를 확인합니다. (오버로드 1/2)
FrameworkLogoffDLL FrameworkLogoffDLL 메서드는 DllCanUnloadNow에 의해 호출되어 공급자 서버가 사용되고 있지 않고 언로드될 수 있는지 여부를 확인합니다. (오버로드 2/2)
무료 구문 분석되지 않은 경로가 포함된 메모리를 해제합니다. 이 개체는 사용하지 않는 것이 좋습니다. 대신 IWbemPath COM 인터페이스를 사용합니다. (오버로드 2/2)

무료 구문 분석되지 않은 경로가 포함된 메모리를 해제합니다. 이 개체는 사용하지 않는 것이 좋습니다. 대신 IWbemPath COM 인터페이스를 사용합니다. (오버로드 1/2)
FreeExtra FreeExtra 메서드는 이전에 문자열에 의해 할당되었지만 더 이상 필요하지 않은 추가 메모리를 해제합니다.
FreeExtra FreeExtra 메서드는 배열이 증가하는 동안 할당된 추가 메모리를 해제합니다. 이 메서드는 배열의 크기 또는 상한에 영향을 주지 않습니다.
FreeMemory IWbemQuery::FreeMemory 메서드는 이전 GetAnalysis 호출에서 파서가 호출자에게 반환하는 메모리를 해제합니다.
Get IWbemClassObject::Get 메서드는 지정된 속성 값(있는 경우)을 검색합니다. 이 메서드는 시스템 속성을 반환할 수도 있습니다.
Get IWbemQualifierSet::Get 메서드는 지정된 명명된 한정자를 가져옵니다(있는 경우).
GetAllDerivedInstances GetAllDerivedInstances 메서드는 기본 클래스의 인스턴스 또는 해당 기본 클래스의 자식 목록을 검색합니다.
GetAllDerivedInstancesAsynch GetAllDerivedInstancesAsynch 메서드는 특정 공급자에서 지원되고 특정 기본 클래스에서 파생된 인스턴스 목록을 검색합니다. 이 메서드를 사용하면 한 번에 하나의 인스턴스를 반환하여 공급자가 비동기적으로 응답할 수 있습니다.
GetAllInstances GetAllInstances 메서드는 특정 클래스에서 반환된 인스턴스 목록을 검색합니다.
GetAllInstancesAsynch GetAllInstancesAsynch 메서드는 특정 클래스에서 반환된 인스턴스 목록을 검색합니다. 이 메서드를 사용하면 한 번에 하나의 인스턴스를 반환하여 공급자가 비동기적으로 응답할 수 있습니다.

GetAllocLength GetAllocLength 메서드는 할당된 문자열 버퍼의 크기를 반환합니다.
GetAnalysis 성공적인 쿼리 구문 분석의 결과를 가져옵니다.
GetAt GetAt 메서드는 CHPtrArray 배열의 요소에 액세스합니다.
GetAt CHString::GetAt(int)(chstring.h) 메서드는 인덱스 번호로 지정된 단일 문자를 반환합니다.
GetAt CHString::GetAt(chstring.h) 메서드는 인덱스 번호로 지정된 단일 문자를 반환합니다.
GetAt 지정된 인덱스에서 배열 요소를 가져옵니다.
Getbool Getbool 메서드는 부울 속성을 검색합니다.
GetBSTR 시간을 CIM 날짜 및 시간 형식의 BSTR 값으로 가져옵니다.
GetBSTR GetBSTR 메서드는 날짜 및 시간 형식의 BSTR로 시간 범위를 가져옵니다.
GetBuffer GetBuffer 메서드는 CHString 개체의 내부 문자 버퍼에 대한 포인터를 반환합니다.
GetBufferSetLength GetBufferSetLength 메서드는 CHString 개체의 내부 문자 버퍼에 대한 포인터를 반환하여 nNewLength에 지정된 길이와 정확히 일치하도록 필요한 경우 길이를 잘라내거나 늘입니다.
GetByte GetByte 메서드는 BYTE 호환 속성을 검색합니다.

GetCallStatus IWbemCallResult::GetCallStatus 메서드는 현재 미해결 반동기 호출의 상태를 사용자에게 반환합니다. 이 호출이 WBEM_S_NO_ERROR 반환하면 IWbemServices 메서드에 대한 원래 호출이 완료됩니다.
GetCHString GetCHString 메서드는 문자열 속성을 검색합니다.
GetClassName IWbemPath::GetClassName 메서드는 경로에서 클래스 이름 부분을 검색합니다.
GetClassObjectInterface GetClassObjectInterface 메서드는 IWbemClassObject 인터페이스 포인터를 반환합니다.
GetCount IWbemPathKeyList::GetCount 메서드는 경로의 키 수를 검색합니다.
GetData CHString 개체에서 데이터를 가져오는 GetData 메서드입니다.
GetData GetData 메서드는 배열의 요소에 직접 액세스합니다.
GetDateTime GetDateTime 메서드는 datetime 속성을 반환합니다.
GetDMTF GetDMTF 메서드는 BSTR 값을 CIM 날짜 및 시간 형식으로 변환합니다.
GetDMTFNonNtfs GetDMTFNonNtfs 메서드는 표준 시간대가 없는 FAT에서 CIM 날짜 및 시간 형식의 DMTF 날짜를 가져옵니다.
GetDOUBLE GetDOUBLE 메서드는 DOUBLE 속성을 검색합니다.
GetDWORD GetDWORD 메서드는 DWORD 속성을 검색합니다.

GetEmbeddedObject GetEmbeddedObject 메서드는 포함된 CInstance 개체에 대한 포인터를 검색합니다. 포인터를 사용하여 포함된 개체의 속성을 가져와서 설정할 수 있습니다.
GetEmptyInstance GetEmptyInstance 메서드는 지정된 클래스의 채워지지 않은 단일 인스턴스를 검색합니다. (오버로드 1/2)
GetEmptyInstance GetEmptyInstance 메서드는 지정된 클래스의 채워지지 않은 단일 인스턴스를 검색합니다. (오버로드 2/2)
GetErrorCodeText 오류 코드와 연결된 텍스트 문자열 설명을 반환합니다.
GetFacilityCodeText IWbemStatusCodeText::GetFacilityCodeText 메서드는 오류가 발생한 하위 시스템의 이름(예: "Windows", "WBEM", "SSPI" 또는 "RPC")을 반환합니다.
GetFILETIME GetFILETIME 메서드는 시간을 MFC FILETIME 구조체로 가져옵니다.
GetInfo IWbemPath::GetInfo 메서드는 파서 개체에 배치된 경로에 대한 세부 정보를 반환합니다.
GetInfo IWbemPathKeyList::GetInfo 메서드는 키의 상태 비트를 검색합니다.
GetInstanceByPath CWbemProviderGlue::GetInstanceByPath(wbemglue.h) 메서드는 특정 개체 경로로 식별된 인스턴스를 검색합니다.
GetInstanceByPath CWbemProviderGlue::GetInstanceByPath(wbemglue.h) 메서드는 공급자 GetObject 메서드를 호출하여 특정 개체 경로로 식별되는 인스턴스를 검색합니다.
GetInstanceKeysByPath GetInstanceKeysByPath 메서드는 키 속성만 채워진 특정 개체 경로로 식별되는 인스턴스를 검색합니다.

GetInstancePropertiesByPath 지정된 속성만 채워진 특정 개체 경로로 식별되는 인스턴스를 검색합니다. 채울 속성의 이름은 CHString 배열에 있습니다.
GetInstancesByQuery CWbemProviderGlue::GetInstancesByQuery(wbemglue.h) 메서드는 특정 쿼리와 일치하는 인스턴스 목록을 검색합니다.
GetInstancesByQuery CWbemProviderGlue::GetInstancesByQuery(wbemglue.h) 메서드는 특정 쿼리와 일치하는 인스턴스 목록을 검색합니다.
GetInstancesByQueryAsynch GetInstancesByQueryAsynch 메서드는 특정 공급자가 지원하는 인스턴스 목록을 검색하고 특정 쿼리와 일치합니다. 이 메서드를 사용하면 한 번에 하나의 인스턴스를 반환하여 공급자가 비동기적으로 응답할 수 있습니다.
GetKey IWbemPathKeyList::GetKey 메서드는 키의 이름 또는 값을 검색합니다. 키의 순서는 중요하지 않지만 키는 0에서 인덱싱됩니다.
GetKey2 IWbemPathKeyList::GetKey2 메서드는 키 이름 또는 값을 검색하고 값을 VARIANT로 반환합니다. 키는 0에서 인덱싱되지만 키 순서는 중요하지 않습니다.
GetKeyList 개별 키에 액세스할 수 있도록 IWbemPathKeyList 포인터를 검색합니다.
GetLength GetLength 메서드는 이 CHString 문자열의 와이드 문자 수를 가져옵니다. 개수에는 NULL 종결자가 포함되지 않습니다.
GetLocalComputerName GetLocalComputerName 메서드는 컴퓨터 이름에 대한 상수 참조를 CHString 형식으로 반환합니다.
GetLocalInstancePath GetLocalInstancePath 메서드는 지정된 인스턴스에 대한 전체 개체 경로를 빌드하려고 시도합니다. 이 메서드는 도우미 함수이며 재정의해서는 안 됩니다.

GetLocalOffsetForDate GetLocalOffsetForDate 메서드는 인수에 제공된 FILETIME에 대해 GMT와 현지 시간 사이의 오프셋을 분 단위(+ 또는 -)로 반환합니다. (오버로드 1/4)
GetLocalOffsetForDate GetLocalOffsetForDate 메서드는 인수에 제공된 FILETIME에 대해 GMT와 현지 시간 사이의 오프셋을 분 단위(+ 또는 -)로 반환합니다. (오버로드 3/4)
GetLocalOffsetForDate GetLocalOffsetForDate 메서드는 인수에 제공된 FILETIME에 대해 GMT와 현지 시간 사이의 오프셋을 분 단위(+ 또는 -)로 반환합니다. (오버로드 4/4)
GetLocalOffsetForDate GetLocalOffsetForDate 메서드는 인수에 제공된 FILETIME에 대해 GMT와 현지 시간 사이의 오프셋을 분 단위(+ 또는 -)로 반환합니다. (오버로드 2/4)
GetMethod 요청된 메서드에 대한 정보를 반환합니다.
GetMethodContext GetMethodContext 메서드는 MethodContext 개체에 대한 포인터를 반환합니다.
GetMethodOrigin IWbemClassObject::GetMethodOrigin 메서드는 메서드가 선언된 클래스를 확인하는 데 사용됩니다.
GetMethodQualifierSet IWbemClassObject::GetMethodQualifierSet는 특정 메서드에 대한 한정자 집합을 검색하는 데 사용됩니다.
GetNames 개체의 속성 이름을 검색합니다.
GetNames IWbemContext::GetNames 메서드는 명명된 컨텍스트 값의 모든 이름의 SAFEARRAY 구조를 반환합니다.
GetNames IWbemQualifierSet::GetNames 메서드는 현재 개체 또는 속성에서 사용할 수 있는 모든 한정자의 이름을 검색합니다. 또는 IFlags의 필터 값에 따라 이 메서드는 특정 한정자의 이름을 검색합니다.

GetNamespace GetNamespace 메서드는 CHString 형식으로 네임스페이스 이름에 대한 상수 참조를 반환합니다. 반환된 이름은 원래 공급자 생성자에 지정된 두 번째 매개 변수입니다.
GetNamespaceAt 인덱스에 따라 네임스페이스를 검색합니다.
GetNamespaceConnection GetNameSpaceConnection 메서드는 네임스페이스 연결을 검색하는 데 사용됩니다. (오버로드 2/2)
GetNamespaceConnection GetNameSpaceConnection 메서드는 네임스페이스 연결을 검색하는 데 사용됩니다. (오버로드 1/2)
GetNamespaceCount IWbemPath::GetNamespaceCount 메서드는 경로의 네임스페이스 수를 반환합니다.
GetNext GetNext 메서드는 컬렉션의 다음 인스턴스에 대한 포인터를 가져옵니다.
GetObject GetObject 메서드는 클래스의 인스턴스를 검색하기 위해 WMI에 의해 호출됩니다. (오버로드 1/3)
GetObject GetObject 메서드는 클래스의 인스턴스를 검색하기 위해 WMI에 의해 호출됩니다. (오버로드 2/3)
GetObject GetObject 메서드는 클래스의 인스턴스를 검색하기 위해 WMI에 의해 호출됩니다. (오버로드 3/3)
GetObject IWbemServices::GetObject 메서드는 클래스 또는 인스턴스를 검색합니다. 이 메서드는 현재 IWbemServices 개체와 연결된 네임스페이스에서만 개체를 검색합니다.
GetObjectAsync IWbemServices::GetObjectAsync 메서드는 경로에 따라 클래스 정의 또는 인스턴스를 검색합니다.
GetObjects IWbemHiPerfEnum::GetObjects 메서드는 현재 열거자에 있는 개체를 검색합니다.

GetObjects 제공된 배열에 개체의 키가 아닌 속성을 삽입합니다.
GetObjectText IWbemClassObject::GetObjectText 메서드는 MOF 구문에 있는 개체의 텍스트 렌더링을 반환합니다.
GetProperty IWbemPropertyProvider::GetProperty 메서드는 Windows Management에서 개별 속성 값을 검색하기 위해 호출됩니다.
GetPropertyHandle GetPropertyHandle 메서드는 속성을 식별하는 고유 핸들을 반환합니다. IWbemObjectAccess 메서드를 사용하여 속성 값을 읽거나 쓸 때 이 핸들을 사용하여 속성을 식별할 수 있습니다.
GetPropertyInfoByHandle GetPropertyInfoByHandle 메서드는 속성 핸들과 연결된 속성의 이름 및 데이터 형식을 반환합니다.
GetPropertyOrigin IWbemClassObject::GetPropertyOrigin 메서드는 특정 속성이 도입된 클래스의 이름을 검색합니다.
GetPropertyQualifierSet IWbemClassObject::GetPropertyQualifierSet 메서드는 클래스 개체의 특정 속성에 대한 한정자 집합을 가져옵니다. 인스턴스 또는 클래스 정의의 멤버인 속성과 함께 이 메서드를 사용할 수 있습니다.
GetProviderName GetProviderName 메서드는 공급자의 생성자에 사용되는 클래스의 이름을 검색합니다.
GetQualifierSet IWbemClassObject::GetQualifierSet 메서드는 개체가 인스턴스인지 클래스 정의인지 여부에 관계없이 전체 클래스 개체의 한정자 집합에 대한 읽기 및 쓰기 작업을 허용하는 인터페이스 포인터를 반환합니다.
GetQuery GetQuery 메서드는 CFrameworkQuery 개체와 연결된 실제 WQL 명령을 검색합니다.
GetQueryClassName GetQueryClassName 메서드는 쿼리에서 클래스 이름을 검색합니다.

GetRequiredProperties GetRequiredProperties 메서드는 쿼리의 SELECT 문에 지정된 모든 속성 목록을 반환합니다. SELECT 절과 WHERE 절 모두에서 속성을 반환합니다.
GetRestrictedSink IWbemEventSink::GetRestrictedSink 메서드는 제한된 이벤트 싱크를 검색합니다. 제한된 이벤트 싱크는 이벤트 공급자 등록에 정의된 이벤트의 하위 집합을 필터링하는 싱크입니다.
GetResultObject IWbemCallResult::GetResultObject 메서드는 IWbemServices::GetObject 또는 IWbemServices::ExecMethod에 대한 이전의 반동기 호출에서 개체를 검색하려고 시도합니다.
GetResultServices IWbemServices 포인터를 검색합니다. 이 포인터는 IWbemServices::OpenNamespace를 사용할 수 있게 되면 반동기 호출로 인해 발생합니다.
GetResultString IWbemCallResult::GetResultString 메서드는 IWbemServices::P utInstance에서 새로 만든 인스턴스의 할당된 개체 경로를 반환합니다.
GetScope 인덱스를 기반으로 범위를 검색합니다.
GetScopeAsText 인덱스를 기반으로 텍스트 형식의 범위를 검색합니다.
GetScopeCount IWbemPath::GetScopeCount 메서드는 경로의 범위 수를 반환합니다.
GetServer IWbemPath::GetServer 메서드는 경로의 서버 부분을 검색합니다.
GetService IWbemDecoupledBasicEventProvider::GetService 메서드는 WMI로 다시 호출하는 데 사용할 IWbemService 개체를 검색합니다. 이 메서드는 완전히 동시 액세스를 제공합니다.
getSink

IWbemDecoupledBasicEventProvider::GetSink 메서드는 WMI로 이벤트 전달을 위해 IWbemObjectSink 개체를 검색합니다. 이 메서드는 완전히 동시 액세스를 제공합니다.
GetSize GetSize 함수는 포인터 배열 크기를 가져옵니다. 인덱스는 0부터 시작하므로 크기가 가장 큰 인덱스보다 큽니다.
GetSize GetSize 메서드는 배열의 크기를 가져옵니다. 인덱스는 0부터 시작하므로 크기가 가장 큰 인덱스보다 큽니다.
GetSize GetSize 메서드는 목록에 있는 항목 수를 반환합니다.
GetStatus GetStatus 메서드는 속성이 존재하는지 여부를 확인하고, 있는 경우 해당 형식을 결정합니다.
GetStatusObject GetStatusObject 메서드는 IWbemClassObject 정보에 대한 내부 포인터를 가져옵니다. WMI는 포인터를 기반으로 하는 기능을 구현하지 않습니다.
GetStringArray GetStringArray 메서드는 문자열 배열을 나타내는 속성을 검색합니다.
GetStructtm GetStructtm 메서드는 ANSI C 런타임 구조체 tm 구조체로 시간을 가져옵니다.
GetSYSTEMTIME GetSYSTEMTIME 메서드는 시간을 MFC SYSTEMTIME 구조체로 가져옵니다.
GetText IWbemPath::GetText 메서드는 이전에 파서 개체에 배치된 경로의 텍스트 표현을 반환합니다.
GetText IWbemPathKeyList::GetText 메서드는 키 목록을 텍스트로 검색합니다.
GetText

IWbemObjectTextSrc::GetText 메서드는 IWbemClassObject 개체의 텍스트 표현을 만듭니다. 예를 들어 XML 표현입니다.
GetTime GetTime 메서드는 시간을 64비트 정수로 반환합니다.
GetTime GetTime 메서드는 시간 범위를 64비트 정수로 반환합니다.
Gettime_t Gettime_t 메서드는 ANSI C 런타임 time_t 변수로 시간을 가져옵니다.
GetTimeSpan GetTimeSpan 메서드는 WMI 시간 범위를 나타내는 속성을 검색합니다.
GetUpperBound GetUpperBound 메서드는 배열의 현재 상한을 가져옵니다. 배열 인덱스는 0부터 시작하므로 이 함수는 GetSize보다 작은 값을 반환합니다.
GetValue IWbemContext::GetValue 메서드는 이름으로 명명된 특정 컨텍스트 값을 검색하는 데 사용됩니다.
GetValuesForProp GetValuesForProp 메서드는 쿼리 내에 표시될 때 해당 속성에 의해 생성되는 특정 속성에 대한 모든 값을 반환합니다. (오버로드 1/2)
GetValuesForProp GetValuesForProp 메서드는 쿼리 내에 표시될 때 해당 속성에 의해 생성되는 특정 속성에 대한 모든 값을 반환합니다. (오버로드 2/2)
GetVariant GetVariant 메서드는 VARIANT 속성을 검색합니다.
GetWBEMINT16 GetWBEMINT16 메서드는 16비트 정수 속성을 검색합니다.
GetWBEMINT64 GetWBEMINT64 메서드는 64비트 정수 속성을 검색합니다. (오버로드 1/3)

GetWBEMINT64 GetWBEMINT64 메서드는 64비트 정수 속성을 검색합니다. (오버로드 3/3)
GetWBEMINT64 GetWBEMINT64 메서드는 64비트 정수 속성을 검색합니다. (오버로드 2/3)
GetWCHAR GetWCHAR 메서드는 WCHAR 문자열 속성을 검색합니다.
GetWORD GetWORD 메서드는 WORD 속성을 검색합니다.
가리키다 알림을 제공하기 위해 소스에서 호출됩니다.
IndicateToConsumer 실제로 소비자에게 이벤트를 전달하기 위해 WMI에 의해 호출됩니다.
InheritsFrom IWbemClassObject::InheritsFrom 메서드는 현재 클래스 또는 인스턴스가 지정된 부모 클래스에서 파생되는지 여부를 결정합니다.
초기화 클라이언트 요청을 받을 공급자를 초기화하기 위해 Windows Management에서 호출됩니다. 모든 유형의 공급자는 이 메서드를 구현해야 합니다.
InsertAt InsertAt 메서드는 nStartIndex에서 지정한 인덱스에 다른 CHStringArray 배열의 모든 요소를 삽입합니다.
InsertAt InsertAt 메서드는 지정된 인덱스에 요소(또는 요소의 여러 복사본) 또는 다른 배열의 모든 요소를 삽입합니다.
IsActive IWbemEventSink::IsActive 메서드는 싱크가 필터링하는 이벤트에 관심이 있는지 확인하기 위해 공급자가 사용합니다.

IsDerivedFrom CWbemProviderGlue::IsDerivedFrom(wbemglue.h) 메서드는 한 클래스가 다른 클래스에서 파생되는지 여부를 결정합니다.
IsDerivedFrom CWbemProviderGlue::IsDerivedFrom(wbemglue.h) 메서드는 한 클래스가 다른 클래스에서 파생되는지 여부를 결정합니다.
IsEmpty IsEmpty 메서드는 빈 조건에 대한 CHString 문자열을 테스트합니다.
IsLocal IWbemPath::IsLocal 메서드는 전달된 컴퓨터 이름이 경로의 컴퓨터 이름과 일치하는지 또는 경로의 서버 이름이 NULL 또는 "."인지 테스트합니다.
IsNull IsNull 메서드는 특정 속성의 값이 NULL인지 여부를 확인합니다.
IsOk IsOk 메서드는 WBEMTime 개체가 유효한 시간을 나타내는지 여부를 나타냅니다.
IsOk IsOk 메서드는 WBEMTimeSpan 개체가 유효한 시간 범위를 나타내는지 여부를 나타냅니다.
IsPropertyRequired IsPropertyRequired 메서드는 쿼리에서 특정 속성을 요청했는지 여부를 결정합니다. SELECT 절과 WHERE 절이 모두 선택됩니다.
IsRelative IWbemPath::IsRelative 메서드는 파서에 이미 설정된 경로가 특정 컴퓨터 및 네임스페이스를 기준으로 하는지 테스트합니다.
IsRelativeOrChild IWbemPath::IsRelativeOrChild 메서드는 파서에 이미 설정된 경로가 특정 컴퓨터 및 네임스페이스의 상대 경로인지 또는 자식인지 테스트합니다.
IsSameClassName

IWbemPath::IsSameClassName 메서드는 전달된 클래스 이름이 경로의 이름과 일치하는지 테스트합니다. 이 메서드는 경로에 실제로 클래스 이름이 있는 경우에만 TRUE를 반환할 수 있습니다.
KeysOnly KeysOnly 메서드는 키 속성만 필요한지 여부를 나타냅니다.
왼쪽 CHString 문자열에서 첫 번째 nCount 문자를 추출하고 추출된 부분 문자열의 복사본을 반환합니다.
LoadStringW LoadStringW 메서드는 Windows 문자열 리소스(nID로 식별됨)를 기존 CHString 개체로 읽습니다. (오버로드 1/2)
LoadStringW LoadStringW 메서드는 Windows 문자열 리소스(nID로 식별됨)를 기존 CHString 개체로 읽습니다. (오버로드 2/2)
잠금 Lock 메서드는 잠금이 해제될 때까지 다른 스레드가 IWbemObjectAccess 개체를 업데이트하지 못하도록 합니다.
LockBuffer LockBuffer 메서드는 버퍼의 문자열을 잠급 수 있습니다.
MakeLocalPath MakeLocalPath 메서드는 상대 경로에서 전체 인스턴스 경로를 빌드합니다.
MakeLower MakeLower 메서드는 CHString 문자열을 소문자 문자열로 변환합니다.
MakeReverse MakeReverse 메서드는 CHString 문자열의 문자 순서를 반대로 바꿉니다.
MakeSingleton IWbemPathKeyList::MakeSingleton 메서드는 키가 싱글톤인지 여부를 제어합니다.
MakeUpper MakeUpper 메서드는 CHString 문자열을 대문자 문자열로 변환합니다.

Mid Mid 메서드는 nFirst 위치(0부터 시작)부터 CHString 문자열에서 길이 nCount 문자의 부분 문자열을 추출합니다. 메서드는 추출된 부분 문자열의 복사본을 반환합니다.
Mid CHString::Mid 메서드는 nFirst 위치(0부터 시작)부터 CHString 문자열에서 길이 nCount 문자의 부분 문자열을 추출합니다.
NewQuery 논리 이벤트 소비자가 Windows Management에 관련 이벤트 쿼리 필터를 등록할 때 IWbemEventProviderQuerySink::NewQuery 메서드를 호출합니다.
다음 IEnumWbemClassObject::Next 메서드를 사용하여 열거형의 현재 위치에서 시작하는 하나 이상의 개체를 가져옵니다.
다음 IWbemClassObject::Next 메서드는 IWbemClassObject::BeginEnumeration으로 시작된 열거형에서 다음 속성을 검색합니다.
다음 IWbemContext::Next 메서드는 IWbemContext::BeginEnumeration으로 시작하는 모든 컨텍스트 값의 열거형에서 다음 값을 검색합니다.
다음 IWbemQualifierSet::Next 메서드는 IWbemQualifierSet::BeginEnumeration으로 시작된 열거형에서 다음 한정자를 검색합니다.
NextAsync 싱크에 대한 개체의 제어된 비동기 검색이 필요한 경우 NextAsync 메서드를 사용합니다.
NextMethod IWbemClassObject::BeginMethodEnumeration에 대한 호출로 시작하는 메서드 열거형 시퀀스에서 다음 메서드를 검색하는 데 사용됩니다.
OnFinalRelease OnFinalRelease 메서드는 참조 수가 0에 도달하면 Release에서 호출하는 가상 함수입니다. CThreadBase는 내부적으로 호출됩니다.

OpenNamespace IWbemServices::OpenNamespace 메서드는 지정된 자식 네임스페이스를 운영 컨텍스트로 사용하는 새 IWbemServices 포인터를 호출자에게 제공합니다.
연산자- WBEMTime 클래스 빼기 연산자(—)의 이 오버로드는 결과 시간이 포함된 새 시간 개체를 생성하기 위해 개체의 시간 범위를 뺍니다. (오버로드 2/2)
연산자- WBEMTime 클래스 빼기 연산자(—)의 이 오버로드는 결과 시간이 포함된 새 시간 개체를 생성하기 위해 개체의 시간 범위를 뺍니다. (오버로드 1/2)
연산자- WBEMTimeSpan 클래스 빼기 연산자(—)는 메서드가 실행되는 개체에서 시간 범위를 뺍니다.
연산자 LPCWSTR 연산자 LPCWSTR 변환 연산자는 CHString 개체에 포함된 NULL로 끝나는 C 문자열에 액세스하는 효율적인 방법을 제공합니다.
operator!= chstring.operator-not-equal-to 함수는 두 문자열을 비교합니다. 이러한 연산자는 대/소문자를 구분하는 Compare 메서드를 편리하게 대체합니다.
operator!= 두 WBEMTime 개체를 비교하기 위해 WBEMTime::operator-not-equal-to(wbemtime.h) 비교 연산자(== != <<= >>=)가 오버로드되었습니다.
operator!= 같지 않은 비교 연산자를 사용하여 두 WBEMTimeSpan 개체를 비교합니다.
operator+ WBEMTime 클래스 추가 연산자(+)가 오버로드되어 시간 범위별로 개체의 시간을 증가시켰습니다.
operator+ WBEMTimeSpan 클래스 add 연산자는 한 번 범위를 다른 시간에 추가하여 메서드에서 반환된 새 WBEMTimeSpan 개체에 합계를 배치합니다.
operator+=

WBEMTime 클래스 추가 및 할당(+=) 연산자가 오버로드되어 시간 범위별로 개체의 시간을 증가시켰습니다.
operator+= WBEMTimeSpan 클래스 추가 및 할당 연산자는 한 번 범위를 다른 시간에 추가합니다. 이 작업은 결과 시간을 포함하는 새 시간 범위를 만듭니다.
연산자< chstring.operator-less-than 함수는 두 문자열을 비교합니다. 이러한 연산자는 대/소문자를 구분하는 Compare 메서드를 편리하게 대체합니다.
연산자< 두 WBEMTime 개체를 비교하기 위해 WBEMTime::operator-less-than(wbemtime.h) 비교 연산자(== != <<= >>=)가 오버로드되었습니다.
연산자< 보다 작은 비교 연산자를 사용하여 두 WBEMTimeSpan 개체를 비교합니다.
operator<= chstring.operator-less-than-equal-to 함수는 두 문자열을 비교합니다. 이러한 연산자는 대/소문자를 구분하는 Compare 메서드를 편리하게 대체합니다.
operator<= 두 WBEMTime 개체를 비교하기 위해 WBEMTime::operator-less-than-equal-to(wbemtime.h) 비교 연산자(== != <<= >>=)가 오버로드되었습니다.
operator<= 보다 작거나 같은 비교 연산자를 사용하여 두 WBEMTimeSpan 개체를 비교합니다.
operator= WBEMTime 클래스 할당 연산자 오버로드 메서드는 CIM 날짜 시간 형식 매개 변수를 사용합니다. (오버로드 4/5)
operator= WBEMTime 클래스 할당 연산자 오버로드 메서드는 CIM 날짜 시간 형식 매개 변수를 사용합니다. (오버로드 5/5)
operator=

WBEMTime 클래스 할당 연산자 오버로드 메서드는 CIM 날짜 시간 형식 매개 변수를 사용합니다. (오버로드 2/5)
operator= WBEMTime 클래스 할당 연산자 오버로드 메서드는 CIM 날짜 시간 형식 매개 변수를 사용합니다. (오버로드 1/5)
operator= WBEMTime 클래스 할당 연산자 오버로드 메서드는 CIM 날짜 시간 형식 매개 변수를 사용합니다. (오버로드 3/5)
operator= BSTR 시간 간격 값을 CIM 날짜 및 시간 형식의 WBEMTimeSpan 개체로 변환합니다. (오버로드 2/3)
operator= BSTR 시간 간격 값을 CIM 날짜 및 시간 형식의 WBEMTimeSpan 개체로 변환합니다. (오버로드 1/3)
operator= BSTR 시간 간격 값을 CIM 날짜 및 시간 형식의 WBEMTimeSpan 개체로 변환합니다. (오버로드 3/3)
operator-= WBEMTime 클래스 subtract-and-assign(—=) 연산자가 오버로드되어 시간 범위별로 개체의 시간을 감소시켰습니다.
operator-= 빼기 및 할당 연산자(—=)를 사용하여 두 WBEMTimeSpan 개체를 비교합니다.
operator== chstring.operator-equal-equal-to 함수는 두 문자열을 비교합니다. 이러한 연산자는 대/소문자를 구분하는 Compare 메서드를 편리하게 대체합니다.
operator== 두 WBEMTime 개체를 비교하기 위해 WBEMTime::operator-equal-equal-to(wbemtime.h) 비교 연산자(== != <<= >>=)가 오버로드되었습니다.
operator== 같음 비교 연산자를 사용하여 두 WBEMTimeSpan 개체를 비교합니다.
연산자>

chstring.operator-greater-than 함수는 두 문자열을 비교합니다. 이러한 연산자는 대/소문자를 구분하는 Compare 메서드를 편리하게 대체합니다.
연산자>   두 WBEMTime 개체를 비교하기 위해 WBEMTime 비교 연산자(== != <<= >>=)가 오버로드되었습니다.
연산자>   보다 큰 비교 연산자를 사용하여 두 WBEMTimeSpan 개체를 비교합니다.
operator>=   chstring.operator-greater-than-equal-to 함수는 두 문자열을 비교합니다. 이러한 연산자는 대/소문자를 구분하는 Compare 메서드를 편리하게 대체합니다.
operator>=   두 WBEMTime 개체를 비교하기 위해 WBEMTime::operator-greater-than-equal-to(wbemtime.h) 비교 연산자(== != <<= >>=)가 오버로드되었습니다.
operator>=   보다 크거나 같은 비교 연산자를 사용하여 두 WBEMTimeSpan 개체를 비교합니다.
구문 분석   쿼리 문자열을 구문 분석합니다.
구문 분석   WMI 경로가 포함된 문자열을 서버, 네임스페이스, 클래스, 인스턴스를 식별하는 키 등의 경로 부분을 포함하는 구조체로 구문 분석합니다. 이 개체는 사용하지 않는 것이 좋습니다. 대신 IWbemPath COM 인터페이스를 사용합니다.
일시 중지   IWbemBackupRestoreEx::P ause 메서드는 WMI(Windows Management Instrumentation) 리포지토리에서 쓰기 작업을 잠그고 읽기 작업이 차단될 수 있습니다.
PromptUser   TBD(IWbemObjectSinkEx.PromptUser)
ProvideEvents   Windows Management는 IWbemEventProvider::P rovideEvents 메서드를 호출하여 이벤트 공급자에게 이벤트 배달을 시작하도록 신호를 표시합니다.

공급자 공급자 메서드는 공급자의 인스턴스를 만듭니다. 이 메서드는 WMI 공급자 프레임워크의 일부입니다.
놓다 명명된 속성을 새 값으로 설정합니다.
놓다 IWbemQualifierSet::P ut 메서드는 명명된 한정자와 값을 씁니다. 새 한정자는 동일한 이름의 이전 값을 덮어씁니다. 한정자는 존재하지 않으면 만들어집니다.
PutClass IWbemServices::P utClass 메서드는 새 클래스를 만들거나 기존 클래스를 업데이트합니다. pObject 매개 변수로 지정된 클래스가 모든 필수 속성 값으로 올바르게 초기화되어 있어야 합니다.
PutClassAsync IWbemServices::P utClassAsync 메서드는 새 클래스를 만들거나 기존 클래스를 업데이트합니다.
PutInstance PutInstance 메서드는 인스턴스를 업데이트합니다. (오버로드 2/2)
PutInstance PutInstance 메서드는 인스턴스를 업데이트합니다. (오버로드 1/2)
PutInstance IWbemServices::P utInstance 메서드는 기존 클래스의 인스턴스를 만들거나 업데이트합니다. 인스턴스는 WMI 리포지토리에 기록됩니다.
PutInstanceAsync IWbemServices::P utInstanceAsync 메서드는 기존 클래스의 인스턴스를 비동기적으로 만들거나 업데이트합니다. 업데이트 확인 또는 오류 보고는 호출자가 구현한 IWbemObjectSink 인터페이스를 통해 제공됩니다.
PutMethod 메서드를 만드는 데 사용됩니다.
PutProperty IWbemPropertyProvider::P utProperty 메서드는 속성 공급자가 지원하는 속성 값을 업데이트하기 위해 Windows Management에서 호출합니다.

QueryInstances 제공된 IWbemObjectSink 인스턴스를 사용하여 지정된 클래스의 인스턴스를 반환합니다.
QueryObjectSink IWbemServices::QueryObjectSink 메서드를 사용하면 호출자가 Windows Management에서 내보낸 알림 처리기를 가져올 수 있습니다.
ReadDWORD ReadDWORD 메서드는 속성 핸들을 사용하여 32비트 속성 데이터를 읽습니다.
ReadPropertyValue ReadPropertyValue 메서드는 속성 핸들과 연결된 속성의 지정된 바이트 수를 반환합니다.
ReadQWORD ReadQWORD 메서드는 속성 핸들로 식별된 64비트 속성 데이터를 읽습니다.
새로 고침 IWbemRefresher::Refresh 메서드는 새로 고칠 수 있는 모든 개체, 열거자 및 중첩된 새로 고침을 업데이트합니다. WMI 새로 고침은 새로 고침에 대한 클라이언트 요청에 대한 응답으로 이 함수를 호출합니다.
등록기 IWbemDecoupledRegistrar::Register 메서드는 WMI에 개체 인터페이스를 등록합니다.
ReleaseBuffer GetBuffer에서 할당한 버퍼 사용을 종료합니다.
Remove IWbemConfigureRefresher::Remove 메서드는 새로 고침에서 개체, 열거자 또는 중첩된 새로 고침을 제거하는 데 사용됩니다.
removeAll IWbemHiPerfEnum::RemoveAll 메서드는 열거자의 모든 개체를 비웁니다.
removeAll RemoveAll 메서드는 이 배열에서 모든 CHPtrArray 멤버를 제거합니다.
removeAll

RemoveAll 메서드는 이 배열에서 모든 CHString 멤버를 제거합니다.
RemoveAllKeys IWbemPathKeyList::RemoveAllKeys 메서드는 모든 키를 제거합니다.
RemoveAllNamespaces IWbemPath::RemoveAllNamespaces 메서드는 경로의 네임스페이스 부분을 제거합니다.
RemoveAllScopes IWbemPath::RemoveAllScopes 메서드는 경로에서 모든 범위를 제거합니다.
RemoveAt RemoveAt 메서드는 배열의 지정된 인덱스에서 시작하는 하나 이상의 요소를 제거합니다.
RemoveKey IWbemPathKeyList::RemoveKey 메서드는 wszName 매개 변수와 일치하는 키를 제거합니다.
RemoveNamespaceAt IWbemPath::RemoveNamespaceAt 메서드는 특정 인덱스의 네임스페이스를 제거합니다. 맨 왼쪽 네임스페이스의 인덱스 값은 0이지만 오른쪽의 네임스페이스는 인덱스 값이 점점 더 높습니다.
RemoveObjects IWbemHiPerfEnum::RemoveObjects 메서드는 새로 고침에서 개체(새로 고침 식별자에 의해 식별됨)를 제거합니다.
RemoveScope IWbemPath::RemoveScope 메서드는 인덱스에 따라 범위를 제거합니다.
재설정 IEnumWbemClassObject::Reset 메서드는 열거형 시퀀스를 다시 시작으로 다시 설정합니다. CIM 개체는 동적이므로 이 메서드를 호출해도 이전에 가져온 것과 동일한 개체 목록이 반드시 반환되는 것은 아닙니다.
복원 IWbemBackupRestore::Restore 메서드는 현재 리포지토리의 내용을 삭제하고 이전에 지정한 백업의 내용으로 복원합니다.
Resume

IWbemBackUpRestoreEx::Resume 메서드는 작업을 계속할 수 있도록 WMI(Windows Management Instrumentation) 리포지토리에 대한 잠금을 해제합니다.
ReverseFind ReverseFind 메서드는 CHString 문자열에서 부분 문자열의 마지막 일치 항목을 검색합니다. 이 메서드는 런타임 함수인 wcsrchr과 유사합니다.
오른쪽 CHString 문자열에서 마지막 nCount 문자를 추출하고 추출된 부분 문자열의 복사본을 반환합니다.
SetAt 인덱스 번호로 지정된 문자 하나를 덮어씁니다.
SetAt CHStringArray::SetAt(int,LPCWSTR)(chstrarr.h) 메서드는 지정된 인덱스에서 배열 요소를 설정합니다.
SetAt CHStringArray::SetAt(chstrarr.h) 메서드는 지정된 인덱스에서 배열 요소를 설정합니다.
SetAtGrow 지정된 인덱스에서 배열 요소를 설정합니다.
SetBatchingParameters IWbemEventSink::SetBatchingParameters 메서드는 최대 이벤트 버퍼 크기 및 관련 처리 대기 시간 값을 설정하는 데 사용됩니다.
Setbool Setbool 메서드는 부울 속성을 설정합니다.
SetByte SetByte 메서드는 BYTE 속성을 설정합니다.
SetCharSplat SetCharSplat(LPCWSTR, DWORD) 메서드는 문자열을 설정합니다.
SetCharSplat SetCharSplat(LPCWSTR, LPCSTR) 메서드는 문자열 속성을 설정합니다.

SetCharSplat SetCharSplat(LPCWSTR, LPCWSTR) 메서드는 문자열 속성을 설정합니다.
SetCHString SetCHString(LPCWSTR, const CHString&) 메서드는 문자열 속성을 설정합니다. (오버로드 2/2)
SetCHString SetCHString(LPCWSTR, LPCSTR) 메서드는 문자열 속성을 설정합니다.
SetCHString SetCHString(LPCWSTR, const CHString&) 메서드는 문자열 속성을 설정합니다. (오버로드 1/2)
SetClassName IWbemPath::SetClassName 메서드는 경로의 클래스 이름 부분을 설정합니다.
SetCreationClassName SetCreationClassName 메서드는 지정된 인스턴스의 CreationClassName 문자열 속성(있는 경우)을 이 공급자의 이름으로 설정합니다.
SetDateTime SetDateTime 메서드는 datetime 속성을 설정합니다.
SetDMTF SetDMTF 메서드는 WBEMTime 개체의 시간을 설정합니다. 날짜 및 시간 형식의 BSTR 매개 변수에 의해 시간이 지정됩니다. 1601년 1월 1일 자정 이전의 날짜 및 시간 인수는 유효하지 않습니다.
SetDOUBLE CInstance::SetDOUBLE 메서드
SetDWORD SetDWORD 메서드는 DWORD 속성을 설정합니다.
SetEmbeddedObject SetEmbeddedObject 메서드는 포함된 CInstance 속성을 설정합니다.
SetKey 키의 이름 또는 값 쌍을 설정합니다.

SetKey2 변형을 사용하여 키의 이름 또는 값 쌍을 설정합니다.
SetNamespaceAt IWbemPath::SetNamespace 메서드는 경로에서 네임스페이스가 배치되는 위치를 지정하기 위해 0부터 시작하는 인덱싱을 사용하여 경로의 네임스페이스를 설정합니다.
SetNull SetNull 메서드는 속성을 NULL로 설정합니다.
SetRegistrationObject IWbemProviderIdentity::SetRegistrationObject 메서드는 이벤트 공급자를 초기화하기 전에 Windows 관리 서비스에서 호출됩니다(공급자가 IWbemProviderIdentity를 구현하는 경우).
SetScope IWbemPath::SetScope 메서드는 인덱스를 기반으로 경로의 범위를 설정합니다. 인덱스는 항상 0이며 범위는 경로의 클래스 또는 키 부분입니다. 이 메서드는 클래스 이름도 설정합니다.
SetServer IWbemPath::SetServer 메서드는 경로의 서버 부분을 설정합니다.
SetSinkSecurity 전달되는 모든 이벤트에 대해 싱크에서 SD(보안 설명자)를 설정하는 데 사용됩니다.
SetSize SetSize 메서드는 빈 배열 또는 기존 배열의 크기를 설정합니다.
setStatus 알림 시퀀스의 끝을 나타내거나 싱크에 다른 상태 코드를 보내기 위해 소스에서 호출됩니다.
setStatus IWbemProviderInitSink::SetStatus 메서드는 공급자가 완전히 또는 부분적으로 초기화되었는지 여부를 Windows Management에 나타냅니다.
SetStatusObject SetStatusObject 메서드는 오류가 발생할 때 자세한 정보를 제공하는 데 사용되는 상태 개체의 매개 변수를 설정합니다. 이 상태 개체는 Win32_PrivilegesStatus 클래스에서 파생됩니다.

SetStatusObject SetStatusObject 메서드는 IWbemClassObject 정보에 대한 내부 포인터를 설정합니다. WMI는 포인터를 기반으로 하는 기능을 구현하지 않습니다.
SetStringArray SetStringArray 메서드는 문자열 배열을 나타내는 속성을 설정합니다.
SetText IWbemPath::SetText 메서드는 경로 파서에서 경로 정보를 반환할 수 있도록 경로를 구문 분석합니다.
SetTimeSpan SetTimeSpan 메서드는 시간 범위를 나타내는 속성을 설정합니다.
SetValue IWbemContext::SetValue 메서드는 명명된 컨텍스트 값을 만들거나 덮어씁니다.
SetVariant SetVariant 메서드는 VARIANT 속성을 설정합니다.
SetWBEMINT16 SetWBEMINT16 메서드는 16비트 정수 속성을 설정합니다.
SetWBEMINT64 SetWBEMINT64(LPCWSTR, const LONGLONG&) 메서드는 64비트 정수 값을 설정합니다. (오버로드 3/3)
SetWBEMINT64 SetWBEMINT64(LPCWSTR, const LONGLONG&) 메서드는 64비트 정수 값을 설정합니다. (오버로드 1/3)
SetWBEMINT64 SetWBEMINT64(LPCWSTR, const LONGLONG&) 메서드는 64비트 정수 값을 설정합니다. (오버로드 2/3)
SetWCHARSplat SetWCHARSplat 메서드는 문자열 속성을 설정합니다.
SetWORD SetWORD 메서드는 WORD 속성을 설정합니다.

종료 IWbemShutdown::Shutdown 메서드는 공급자 서비스가 필요하지 않음을 공급자에게 나타냅니다.
건너뛰기 IEnumWbemClassObject::Skip 메서드를 사용하여 열거형의 현재 위치를 지정된 수의 개체만큼 앞으로 이동할 수 있습니다. 또한 이는 NextAsync에 대한 후속 호출에 영향을 주지만 NextAsync로 시작된 보류 중인 배달에는 영향을 주지 않습니다.
SpanExcluding SpanExcluding 메서드는 lpszCharSet에서 문자가 처음 나타나는 앞에 오는 모든 문자를 추출하고 반환합니다.
SpanIncluding SpanIncluding 메서드는 lpszCharSet으로 식별되는 문자열의 문자를 추출합니다.
SpawnDerivedClass IWbemClassObject::SpawnDerivedClass 메서드를 사용하여 현재 개체에서 새로 파생된 클래스 개체를 만듭니다.
SpawnInstance IWbemClassObject::SpawnInstance 메서드를 사용하여 클래스의 새 인스턴스를 만듭니다.
StopRefreshing 제공된 식별자에 해당하는 개체 또는 열거자 새로 고침을 중지합니다.
TRefPointerCollection 이 생성자는 새 TRefPointerCollection 개체를 초기화합니다. (TRefPointerCollection.TRefPointerCollection)
TRefPointerCollection 이 생성자는 새 TRefPointerCollection 개체를 초기화합니다. (오버로드 1/2)
TrimLeft TrimLeft 메서드는 새 줄, 공백 및 탭 문자를 제거하여 문자열에서 선행 공백 문자를 트리밍합니다.
TrimRight TrimRight 메서드는 후행 줄 바꿈, 공백 및 탭 문자를 제거하여 문자열에서 후행 공백 문자를 트리밍합니다.

잠금 해제 Unlock 메서드를 사용하면 다른 스레드가 IWbemObjectAccess 개체의 속성 값을 업데이트할 수 있습니다.
UnlockBuffer UnlockBuffer 메서드는 LockBuffer를 호출하여 이전에 보호된 버퍼의 잠금을 해제하고 참조 수를 1로 다시 설정합니다.
구문 분석 해제 구문 분석된 경로를 포함하는 구조를 문자열로 변환합니다. 이 개체는 사용하지 않는 것이 좋습니다. 대신 IWbemPath COM 인터페이스를 사용합니다.
등록 취소 IWbemDecoupledRegistrar::UnRegister 메서드는 WMI에서 개체 인터페이스 등록을 제거합니다.
ValidateDeletionFlags ValidateDeletionFlags 메서드는 플래그 집합이 삭제 작업에 유효한지 여부를 결정합니다.
ValidateEnumerationFlags ValidateEnumerationFlags 메서드는 플래그 집합이 열거형에 유효한지 여부를 결정합니다.
ValidateFlags ValidateFlags 메서드는 플래그 집합이 유효한지 여부를 결정합니다.
ValidateGetObjFlags ValidateGetObjFlags 메서드는 플래그 집합이 인스턴스 검색 작업에 유효한지 여부를 결정합니다.
ValidateMethodFlags ValidateMethodFlags 메서드는 플래그 집합이 실행 메서드 작업에 유효한지 여부를 결정합니다.
ValidatePutInstanceFlags ValidatePutInstanceFlags 메서드는 플래그 집합이 인스턴스 업데이트 작업에 유효한지 여부를 결정합니다.
ValidateQueryFlags ValidateQueryFlags 메서드는 플래그 집합이 쿼리 작업에 유효한지 여부를 결정합니다.
WBEMTime WBEMTime 클래스 생성자는 다양한 Windows 및 ANSI C 런타임 형식 간의 변환을 용이하게 합니다.

WBEMTime WBEMTime 클래스 생성자 오버로드 메서드는 BSTR 매개 변수를 사용합니다.
WBEMTime WBEMTime 오버로드 클래스 생성자는 FILETIME 참조 매개 변수를 사용합니다.
WBEMTime WBEMTime 오버로드 클래스 생성자는 SYSTEMTIME 매개 변수를 사용합니다.
WBEMTime WBEMTime 오버로드 클래스 생성자는 ANSI C time_t 구조 매개 변수를 사용합니다.
WBEMTime WBEMTime 오버로드 클래스 생성자는 ANSI C tm 구조 매개 변수를 사용합니다.
WBEMTimeSpan WBEMTimeSpan 클래스 생성자는 시간 범위 개체를 만듭니다. 생성자가 오버로드됩니다. (WBEMTimeSpan::WbemTimeSpan)
WBEMTimeSpan 이 WBEMTimeSpan 클래스 생성자는 간격 형식의 BSTR에서 시간 범위 개체를 만듭니다.
WBEMTimeSpan WBEMTimeSpan 클래스 생성자는 시간 범위 개체를 만듭니다. 생성자가 오버로드됩니다. (오버로드 1/3)
WBEMTimeSpan WBEMTimeSpan 클래스 생성자는 시간 범위 개체를 만듭니다. 생성자가 오버로드됩니다. (오버로드 2/3)
WBEMTimeSpan 이 WBEMTimeSpan 클래스 생성자는 경과된 일 수의 정수에서 시간 범위 개체를 만듭니다.
WriteDWORD WriteDWORD 메서드는 속성 핸들로 식별되는 속성에 32비트 데이터를 씁니다.
WriteError TBD(IWbemObjectSinkEx.WriteError)

WriteMessage TBD(IWbemObjectSinkEx.WriteMessage)
WriteProgress TBD(IWbemObjectSinkEx.WriteProgress)
WritePropertyValue WritePropertyValue 메서드는 속성 핸들로 식별되는 속성에 지정된 바이트 수를 씁니다. 이 메서드를 사용하여 문자열 및 다른 모든 비 DWORD 또는 비 QWORD 데이터를 설정합니다.
WriteQWORD WriteQWORD 메서드는 속성 핸들을 사용하여 속성에 64비트 데이터를 씁니다.
WriteStreamParameter TBD(IWbemObjectSinkEx.WriteStreamParameter)

Interfaces

테이블 확장

IEnumWbemClassObject IEnumWbemClassObject 인터페이스는 CIM(Common Information Model) 개체를 열거하는 데 사용되며 표준 COM 열거자와 유사합니다.
IMofCompiler Mofd.dll구현된 IMofCompiler 인터페이스는 MOF(Managed Object Format) 컴파일러 및 MOF 파일을 컴파일하는 다른 애플리케이션에서 사용하는 COM 인터페이스를 제공합니다.
IUnsecuredApartment IUnsecuredApartment 인터페이스는 클라이언트 프로세스에서 비동기 호출을 만드는 프로세스를 간소화하는 데 사용됩니다.
IWbemBackupRestore IWbemBackupRestore 인터페이스는 WMI 리포지토리의 콘텐츠를 백업하고 복원합니다.

IWbemBackupRestoreEx IWbemBackupRestoreEx 인터페이스는 리포지토리의 콘텐츠를 백업하고 복원합니다.
IWbemCallResult IWbemServices 인터페이스의 반동기 호출에 사용됩니다. 이러한 호출을 수행하면 호출된 IWbemServices 메서드가 IWbemCallResult 개체와 함께 즉시 반환됩니다.
IWbemClassObject 클래스 정의와 클래스 개체 인스턴스를 모두 포함하고 조작합니다.
IWbemConfigureRefresher IWbemConfigureRefresher 인터페이스는 클라이언트 코드에서 열거자, 개체 및 중첩된 새로 고침을 새로 고침에 추가하는 데 사용됩니다.
IWbemContext IWbemContext 인터페이스는 필요에 따라 WMI에 IWbemServices 호출을 제출할 때 공급자에게 추가 컨텍스트 정보를 전달하는 데 사용됩니다. IWbemServices의 모든 기본 호출은 이 형식의 개체를 가리키는 선택적 매개 변수를 사용합니다.
IWbemDecoupledBasicEventProvider IWbemDecoupledBasicEventProvider 인터페이스는 분리된 공급자를 WMI에 등록하는 공동 생성 가능한 인터페이스입니다. 만든 개체는 IWbemDecoupledRegistrar::Register의 pUnknown 인수에 전달되어야 합니다.
IWbemDecoupledRegistrar IWbemDecoupledRegistrar 인터페이스는 분리된 공급자를 WMI와 연결합니다. 이 인터페이스를 사용하면 프로세스 호스팅 공급자가 인터페이스의 상호 운용성 수명을 정의하고 다른 공급자와 공존할 수 있습니다.
IWbemEventConsumerProvider 이벤트 소비자 공급자에 대한 기본 인터페이스를 제공합니다. 이 인터페이스와 FindConsumer 메서드를 통해 이벤트 소비자 공급자는 지정된 이벤트를 수신해야 하는 이벤트 소비자를 나타낼 수 있습니다.
IWbemEventProvider IWbemEventProvider 인터페이스를 사용하여 이벤트 공급자와의 통신을 시작합니다. Windows Management는 소비자가 공급자가 생성한 이벤트를 수신하는 데 관심을 표시한 경우 이 인터페이스의 공급자 구현을 호출합니다.
IWbemEventProviderQuerySink

IWbemEventProviderQuerySink 인터페이스는 성능을 최적화하기 위해 현재 활성 상태인 이벤트 쿼리 필터 종류를 알고자 하는 이벤트 공급자에 의해 선택적으로 구현됩니다.
IWbemEventProviderSecurity IWbemEventProviderSecurity 인터페이스는 필요에 따라 해당 이벤트에 대한 소비자 액세스를 제한하려는 이벤트 공급자에 의해 구현됩니다. 이 인터페이스를 사용하는 시기에 대한 자세한 내용은 WMI 이벤트 보안을 참조하세요.
IWbemEventSink 제한된 쿼리 집합을 사용하여 이벤트 공급자와의 통신을 시작합니다.
IWbemHiPerfEnum 인스턴스 개체의 열거에 대한 신속한 액세스를 제공하기 위해 새로 고침 작업에 사용됩니다.
IWbemHiPerfProvider 공급자가 새로 고칠 수 있는 개체 및 열거자를 제공할 수 있도록 합니다.
IWbemLocator IWbemLocator 인터페이스를 사용하여 특정 호스트 컴퓨터에서 WMI용 IWbemServices 인터페이스에 대한 초기 네임스페이스 포인터를 가져옵니다.
IWbemObjectAccess 개체의 메서드 및 속성에 대한 액세스를 제공합니다.
IWbemObjectSink IWbemObjectSink 인터페이스는 WMI 프로그래밍 모델 내에서 모든 유형의 알림을 받을 수 있는 싱크 인터페이스를 만듭니다.
IWbemObjectSinkEx WMI 프로그래밍 모델 내에서 모든 유형의 알림을 받을 수 있는 싱크 인터페이스를 만듭니다.
IWbemObjectTextSrc IWbemObjectTextSrc 인터페이스는 IWbemClassObject 인스턴스를 다른 텍스트 형식으로 변환하는 데 사용됩니다.
IWbemPath IWbemPath 인터페이스는 개체 경로 파서의 기본 인터페이스이며, 표준 방식으로 프로그램에서 사용할 수 있는 경로를 구문 분석합니다. 이 인터페이스는 경로 정보를 설정하고 검색하기 위한 기본 인터페이스입니다.

IWbemPathKeyList 경로 키의 세부 정보에 액세스하는 데 사용됩니다.
IWbemPropertyProvider WMI 클래스의 인스턴스에서 개별 속성 검색 및 업데이트를 지원합니다.
IWbemProviderIdentity 공급자가 동일한 CLSID 값으로 둘 이상의 이름(__Win32Provider 여러 인스턴스)을 사용하여 자신을 등록하는 경우 IWbemProviderIdentity 인터페이스는 이벤트 공급자에 의해 구현됩니다.
IWbemProviderInit IWbemProviderInit 인터페이스는 Windows Management에서 공급자를 초기화하기 위해 호출됩니다. IWbemProviderInit를 구현하려면 모든 공급자가 필요합니다.
IWbemProviderInitSink IWbemProviderInitSink 인터페이스는 WMI에서 구현되며 공급자가 초기화 상태를 보고하도록 호출합니다.
IWbemQualifierSet 단일 속성 또는 전체 개체(클래스 또는 인스턴스)에 대한 명명된 한정자의 전체 집합에 대한 컨테이너 역할을 합니다.
IWbemQuery WQL(WMI 쿼리 언어) 쿼리를 구문 분석할 수 있는 진입점을 제공합니다.
IWbemRefresher 열거자 또는 새로 고침 개체와 같은 새로 고칠 수 있는 개체를 새로 고칠 수 있는 진입점을 제공합니다.
IWbemServices 클라이언트 및 공급자가 WMI 서비스에 액세스하는 데 사용됩니다. 인터페이스는 WMI 및 WMI 공급자에 의해 구현되며 기본 WMI 인터페이스입니다.
IWbemShutdown IWbemShutdown 인터페이스는 개체의 인스턴스를 삭제할 준비가 되었음을 공급자에게 나타냅니다. 공급자는 이 호출을 사용하여 현재 참조하는 리소스를 해제할 수 있습니다.
IWbemStatusCodeText IWbemStatusCodeText 인터페이스는 오류 코드 또는 오류가 발생한 하위 시스템의 이름에 대한 텍스트 문자열 설명을 추출합니다.

IWbemUnboundObjectSink IWbemUnboundObjectSink 인터페이스는 모든 논리 이벤트 소비자에 의해 구현됩니다. 이벤트 개체 배달을 허용하는 간단한 싱크 인터페이스입니다.
IWbemUnsecuredApartment 클라이언트 애플리케이션에서 Unsecapp.exe 비동기 콜백에 대한 액세스 검사를 수행하는지 여부를 확인할 수 있습니다.

구조

테이블 확장

SWbemAssocQueryInf WMIQ_ANALYSIS_ASSOC_QUERY 분석 유형을 사용하는 경우 IWbemQuery::GetAnalysis 메서드의 정보를 포함합니다.
SWbemQueryQualifiedName SWbemQueryQualifiedName 구조체는 IWbemQuery::GetAnalysis 메서드의 속성 이름을 저장합니다.
SWbemRpnEncodedQuery SWbemRpnEncodedQuery 구조에는 WMIQ_ANALYSIS_RPN_SEQUENCE 분석 유형을 사용할 때 IWbemQuery::GetAnalysis 메서드의 정보가 포함됩니다. 구조체의 모든 필드가 활성으로 사용되는 것은 아닙니다. 일부는 나중에 사용하도록 예약되어 있기 때문입니다.
SWbemRpnQueryToken SWbemRpnQueryToken 구조체는 WMIQ_ANALYSIS_RPN_SEQUENCE 형식 쿼리의 쿼리 토큰을 나타냅니다. 쿼리 토큰의 예는 다음과 같습니다. _j > 4.
WBEM_COMPILE_STATUS_INFO IMofCompiler 인터페이스에 대한 오류를 설명합니다.

Last updated on 2025. 10. 08.

Windows Media 디바이스 관리자 11 SDK

아티클2023. 08. 22.

Windows Media 디바이스 관리자 11 SDK 기술 개요입니다.

Windows Media 디바이스 관리자 11 SDK를 개발하려면 다음 헤더가 필요합니다.

- mswmdm.h
- mtpext.h
- scclient.h
- scserver.h
- wmdmlog.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows Media 디바이스 관리자 11 SDK

Functions

Additem AddItem 메서드는 인터페이스에 메타데이터 속성을 추가합니다.
시작 Begin 메서드는 작업이 시작되었음을 나타냅니다. 가능한 경우 작업 기간의 추정치가 제공됩니다.
Begin3 Begin3 메서드는 Windows Media 디바이스 관리자 작업을 시작하려고 함을 나타내기 위해 호출됩니다.
BeginRead BeginRead 메서드는 "디바이스에서 읽기" 작업이 시작되었음을 나타냅니다. Windows Media 디바이스 관리자 애플리케이션이 IWMDMStorageControl::Read를 호출하고 이 IWMDMOperation 인터페이스를 전달하는 경우에만 이 메서드를 호출합니다.
BeginSession BeginSession 메서드는 전송 세션의 시작을 나타냅니다. 전송 세션당 한 번만 발생해야 하는 작업을 최적화하는 데 사용할 수 있습니다.

BeginSession BeginSession 메서드는 디바이스 세션을 시작합니다.
BeginWrite BeginWrite 메서드는 "디바이스에 쓰기" 작업이 시작되었음을 나타냅니다. Windows Media 디바이스 관리자 애플리케이션이 IWMDMStorageControl/2/3::Insert/2/3을 호출하고 이 인터페이스를 전달하는 경우에만 이 메서드를 호출합니다.
복제 Clone 메서드는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다. (IMDSPEnumDevice.Clone)
복제 Clone 메서드는 현재 열거형 상태와 동일한 열거자를 포함하는 다른 열거자를 만듭니다. (IMDSPEnumStorage.Clone)
복제 Clone 메서드는 IWMDMEnumDevice 인터페이스의 복사본을 반환합니다. 새 열거자는 현재 열거자와 동일한 열거형 상태를 지정합니다.
복제 Clone 메서드는 현재 열거자와 동일한 열거형 상태를 가진 다른 열거자를 만듭니다.
닫기 Close 메서드는 미디어 디바이스의 스토리지 매체에 있는 파일을 닫습니다.
CreateDevice CreateDevice 메서드는 Windows Media 디바이스 관리자 PnP 하위 시스템으로부터 가져온 정식 디바이스에 해당하는 IMDSPDevice 개체를 가져오기 위해 호출됩니다.
CreateEmptyMetadataObject CreateEmptyMetadataObject 메서드는 새 IWMDMMetaData 인터페이스를 만듭니다. 이 인터페이스는 스토리지의 메타데이터 속성을 설정하거나 검색하는 데 사용됩니다.
CreateStorage CreateStorage 메서드는 새 스토리지를 만들고 새로 만든 스토리지의 IMDSPStorage 인터페이스에 대한 포인터를 반환합니다.

CreateStorage2 CreateStorage2 메서드는 지정된 이름의 새 스토리지를 만들고 새로 만든 스토리지의 IMDSPStorage 인터페이스에 대한 포인터를 반환합니다.
CreateStorageWithMetadata CreateStorageWithMetadata 메서드는 지정된 메타데이터를 새 스토리지에 적용하여 새 스토리지를 만들고 새로 만든 스토리지의 IMDSPStorage 인터페이스에 대한 포인터를 반환합니다.
삭제 Delete 메서드는 미디어 디바이스의 스토리지 매체에서 개체 또는 개체를 제거합니다.
삭제 Delete 메서드는 이 스토리지를 영구적으로 삭제합니다.
DeviceIoControl DeviceIoControl 메서드는 디바이스 I/O 컨트롤을 호출합니다.
DeviceIoControl DeviceIoControl 메서드는 디바이스에 IOCTL(디바이스 I/O 컨트롤) 코드를 보냅니다. 통과 메서드입니다. Windows Media 디바이스 관리자 매개 변수의 유효성을 검사한 후 서비스 공급자에게 호출을 전달합니다.
사용 Enable 메서드는 로깅을 사용하거나 사용하지 않도록 설정합니다. 기본적으로 로깅을 사용하도록 설정됩니다.
End End 메서드는 성공 여부에 관계없이 읽기 또는 쓰기 작업이 완료되었음을 나타내며 완료 코드를 반환합니다.
End End 메서드는 작업이 완료되었음을 나타냅니다.
End2 End2 메서드는 완성 상태 표시기를 제공하여 IWMDMProgress::End를 확장합니다.
End3 End3 메서드는 작업이 완료되었음을 나타내기 위해 Windows Media 디바이스 관리자 호출됩니다.

EndSession EndSession 메서드는 전송 세션의 끝을 나타냅니다.
EndSession EndSession 메서드는 디바이스 세션을 종료합니다.
EnumDevices EnumDevices 메서드는 현재 연결되어 있고 서비스 공급자가 알고 있는 설치된 물리적 또는 소프트웨어 디바이스를 열거합니다.
EnumDevices EnumDevices 메서드는 컴퓨터에 연결된 휴대용 디바이스를 열거하는 데 사용할 수 있는 IWMDMEnumDevice 인터페이스에 대한 포인터를 검색합니다.
EnumDevices2 EnumDevices2 메서드는 컴퓨터에 연결된 휴대용 디바이스를 열거하는 데 사용되는 열거형 인터페이스를 검색합니다.
EnumStorage EnumStorage 메서드는 디바이스의 최상위 스토리지를 나타내는 열거자 개체의 IMDSPEnumStorage 인터페이스에 대한 포인터를 검색합니다. 디바이스에 대한 최상위 스토리지는 스토리지 매체의 루트 디렉터리입니다.
EnumStorage EnumStorage 메서드는 IMDSPEnumStorage 인터페이스에 액세스하여 디바이스의 개별 스토리지 미디어를 열거합니다.
EnumStorage EnumStorage 메서드는 IWMDMEnumStorage 인터페이스를 검색하여 디바이스의 스토리지를 열거합니다.
EnumStorage EnumStorage 메서드는 IWMDMEnumStorage 인터페이스를 검색하여 현재 스토리지의 직속 자식 스토리지를 열거합니다.
ExamineData ExamineData 메서드는 Windows Media 디바이스 관리자 이 메서드에 전달하는 데이터를 검사하여 콘텐츠에 대한 권한과 책임을 결정합니다.

FindStorage FindStorage 메서드는 지정된 영구 고유 식별자가 있는 스토리지를 찾습니다. 스토리지의 영구 고유 식별자는 해당 스토리지의 g_wszWMDMPersistentUniqueID 속성에 의해 설명됩니다. (IMDSPDevice3.FindStorage)
FindStorage FindStorage 메서드는 지정된 영구 고유 식별자가 있는 스토리지를 찾습니다. 스토리지의 영구 고유 식별자는 해당 스토리지의 g_wszWMDMPersistentUniqueID 속성에 의해 설명됩니다. (IMDSPStorage4.FindStorage)
FindStorage FindStorage 메서드는 영구 고유 식별자를 통해 스토리지를 찾습니다. 다른 메서드와 달리 이 메서드는 루트 스토리지에서 재귀적으로 검색할 수 있습니다.
FindStorage FindStorage 메서드는 영구 고유 식별자를 기반으로 현재 루트 스토리지의 스토리지를 검색합니다.
GetAttributes GetAttributes 메서드는 이 스토리지 개체의 특성을 검색합니다.
GetAttributes GetAttributes 메서드는 스토리지의 특성을 검색합니다.
GetAttributes2 GetAttributes2 메서드는 파일 또는 스토리지의 특성을 가져옵니다.
GetAttributes2 GetAttributes2 메서드는 스토리지의 확장 특성을 검색합니다.
GetCanonicalName GetCanonicalPName 메서드는 디바이스의 정식 이름을 가져옵니다.
GetCanonicalName GetCanonicalName 메서드는 디바이스의 정식 이름을 검색합니다.
GetCapabilities GetCapabilities 메서드는 이 컨트롤 인터페이스가 연결된 디바이스의 기능 마스크를 검색합니다. 이 기능은 미디어 디바이스에서 지원하는 디바이스 컨트롤의 메서드를 설명합니다.

GetCapabilities GetCapabilities 메서드는 이 인터페이스의 instance 연결된 스토리지 매체의 기능을 검색합니다.
GetCapabilities GetCapabilities 메서드는 디바이스 기능을 검색하여 디바이스가 수행할 수 있는 작업을 결정합니다. 이 기능은 미디어 디바이스에서 지원하는 디바이스 컨트롤의 메서드를 설명합니다.
GetCapabilities GetCapabilities 메서드는 루트 스토리지 매체의 기능을 검색합니다.
GetDataDemands GetDataDemands 메서드는 보안 콘텐츠 공급자가 지정된 콘텐츠 부분에 대한 권한과 책임을 결정하는 데 필요한 데이터를 보고합니다.
Getdate GetDate 메서드는 스토리지 개체(파일 또는 폴더)가 가장 최근에 수정된 날짜를 검색합니다.
Getdate GetDate 메서드는 스토리지가 마지막으로 수정된 날짜를 검색합니다.
GetDCStatus GetDCStatus 메서드는 디바이스의 컨트롤 상태 검색합니다.
GetDevice GetDevice 메서드는 이 인터페이스가 연결된 스토리지 매체가 탑재된 디바이스에 대한 포인터를 검색합니다.
GetDeviceCount GetDeviceCount 메서드는 현재 연결되어 있고 서비스 공급자가 알고 있는 설치된 물리적 또는 소프트웨어 디바이스의 수를 반환합니다.
GetDeviceCount GetDeviceCount 메서드는 현재 컴퓨터에 연결된 휴대용 디바이스의 수를 검색합니다.
GetDeviceFromCanonicalName GetDeviceFromCanonicalName 메서드는 지정된 정식 이름을 가진 디바이스에 대한 IWMDMDevice 인터페이스를 검색합니다. IWMDMDevice2::GetCanonicalName을 호출하여 디바이스의 정식 이름을 검색할 수 있습니다.

GetDeviceIcon GetDeviceIcon 메서드는 디바이스 서비스 공급자가 이 디바이스를 나타내는 데 사용해야 한다고 나타내는 아이콘을 나타내는 HICON을 반환합니다.
GetDeviceIcon GetDeviceIcon 메서드는 디바이스가 연결되면 디바이스 제조업체가 표시하려는 아이콘에 대한 핸들을 검색합니다.
GetFormatCapability GetFormatCapability 메서드는 특정 개체 형식의 각 측면에 대해 디바이스에서 지원하는 값 또는 값 범위에 대한 정보를 디바이스에서 검색합니다.
GetFormatCapability GetFormatCapability 메서드는 지정된 형식의 파일에 대한 디바이스 지원을 검색합니다. 기능은 지원되는 속성 및 허용되는 값으로 표현됩니다.
GetFormatSupport GetFormatSupport 메서드는 디바이스에서 지원하는 모든 형식을 검색합니다. 형식 정보에는 코덱, 파일 형식 및 디지털 권한 관리 체계가 포함됩니다.
GetFormatSupport GetFormatSupport 메서드는 코덱 및 파일 형식을 포함하여 디바이스에서 지원하는 모든 형식을 검색합니다.
GetFormatSupport2 GetFormatSupport2 메서드는 오디오 및 비디오 코덱, MIME 파일 형식을 포함하여 디바이스에서 지원하는 형식을 가져옵니다.
GetFormatSupport2 GetFormatSupport2 메서드는 오디오 및 비디오 코덱, MIME 파일 형식을 포함하여 디바이스에서 지원하는 형식을 검색합니다.
GetItemCount GetItemCount 메서드는 인터페이스에서 보유한 총 속성 수를 검색합니다.
GetLastPlayPosition GetLastPlayPosition 메서드는 개체의 마지막 재생 위치를 검색합니다. 개체는 미디어 디바이스의 음악 파일이어야 합니다.

GetLastPlayPosition GetLastPlayPosition 메서드는 개체의 마지막 재생 위치를 검색합니다. 개체는 미디어 디바이스의 오디오 파일이어야 합니다.
GetLogFileName GetLogFileName 메서드는 현재 로그 파일의 전체 경로를 반환합니다.
GetLongestPlayPosition GetLongestPlayPosition 메서드는 개체의 가장 긴 재생 위치를 검색합니다. 개체는 미디어 디바이스의 음악 파일이어야 합니다.
GetLongestPlayPosition GetLongestPlayPosition 메서드는 파일의 가장 긴 재생 위치를 검색합니다. 파일은 미디어 디바이스의 오디오 파일이어야 합니다.
GetManufacturer GetManufacturer 메서드는 디바이스 제조업체의 이름을 검색합니다. (IMDSPDevice.GetManufacturer)
GetManufacturer GetManufacturer 메서드는 디바이스 제조업체의 이름을 검색합니다. (IWMDMDevice.GetManufacturer)
GetMetadata GetMetadata 메서드는 서비스 공급자에서 메타데이터를 검색합니다.
GetMetadata GetMetadata 메서드는 스토리지와 연결된 메타데이터를 검색합니다.
GetName GetName 메서드는 디바이스의 이름을 검색합니다.
GetName GetName 메서드는 스토리지 개체의 표시 이름을 검색합니다.
GetName GetName 메서드는 사람이 읽을 수 있는 미디어 디바이스 이름을 검색합니다.

GetName GetName 메서드는 스토리지의 표시 이름을 검색합니다.
GetObjectAttributes GetObjectAttributes 메서드를 사용하면 애플리케이션이 디바이스에 기록되는 개체의 특성을 지정할 수 있습니다. Windows Media 디바이스 관리자 파일의 특성을 알아보기 위해 파일이 디바이스에 기록되기 전에 이 메서드를 호출합니다.
GetObjectAttributes2 Windows Media 디바이스 관리자 파일의 특성을 알아보기 위해 파일이 디바이스에 기록되면 GetObjectAttributes를 호출합니다.
GetObjectName Windows Media 디바이스 관리자 디바이스에서 이름을 지정해야 하는 항목을 알기 위해 개체가 디바이스에 기록되기 전에 GetObjectName을 호출합니다.
GetObjectTotalSize Windows Media 디바이스 관리자 개체의 총 크기(바이트)를 검색하기 위해 파일이 디바이스에 기록되기 전에 GetObjectTotalSize를 호출합니다.
GetParent GetParent 메서드는 현재 스토리지의 부모를 검색합니다.
GetParent GetParent 메서드는 스토리지의 부모를 검색합니다.
GetPlayLength GetPlayLength 메서드는 개체와 관련한 단위로 개체의 재생 길이를 검색합니다. 총 길이가 아니라 개체가 재생할 수 있는 나머지 길이입니다.
GetPlayLength GetPlayLength 메서드는 개체의 재생 길이를 형식에 적합한 단위로 검색합니다. 총 길이가 아니라 파일이 재생할 수 있는 나머지 길이입니다.
GetPlayOffset GetPlayOffset 메서드는 개체와 관련한 단위로 개체의 재생 오프셋을 검색합니다. IMDSPDeviceControl::P lay의 다음 호출을 위한 시작점입니다.

GetPlayOffset GetPlayOffset 메서드는 개체의 플레이 오프셋을 형식에 적합한 단위로 검색합니다. 이것은 Play의 다음 호출을 위한 시작점입니다.
GetPowerSource GetPowerSource 메서드는 디바이스가 배터리, 외부 전원 또는 둘 다에서 실행할 수 있는지 여부와 현재 실행 중인 전원 유형을 보고합니다.
GetPowerSource GetPowerSource 메서드는 전원에 대한 정보와 디바이스에 남아 있는 전원의 백분율을 검색합니다.
GetProperty GetProperty 메서드는 특정 디바이스 속성을 검색합니다.
GetProperty GetProperty 메서드는 특정 디바이스 메타데이터 속성을 검색합니다.
GetReferences GetReferences 메서드는 연결 스토리지에 포함된 참조(예: 재생 목록 또는 앨범 개체를 나타내는 참조)로 구성된 IMDSPStorage 개체에 대한 포인터 배열을 반환합니다.
GetReferences GetReferences 메서드는 이 스토리지가 가리키는 IWMDMStorage 개체에 대한 포인터 배열을 검색합니다. 추상 앨범 또는 재생 목록은 일반적으로 MTP 디바이스에서 참조 컬렉션으로 저장됩니다.
GetRevision GetRevision 메서드는 현재 사용 중인 Windows Media 디바이스 관리자 버전 번호를 검색합니다.
GetRevocationURL GetRevocationURL 메서드는 업데이트된 구성 요소를 다운로드할 수 있는 URL을 검색합니다. (IMDSPRevoked.GetRevocationURL)
GetRevocationURL GetRevocationURL 메서드는 업데이트된 구성 요소를 다운로드할 수 있는 URL을 검색합니다. (IWMDMRevoked.GetRevocationURL)
GetRights GetRights 메서드는 개체에 대한 권한 정보를 검색합니다.

GetRights GetRights 메서드는 현재 콘텐츠에 대한 권한 정보를 검색합니다. 권한은 파일별로 다릅니다.
GetRights GetRights 메서드는 사용이 허가된 스토리지에 대한 권한 정보를 검색합니다.
GetRightsOnClearChannel GetRightsOnClearChannel 메서드는 명확한 채널의 현재 콘텐츠에 대한 권한 정보를 검색합니다.
GetRightsWithProgress GetRightsWithProgress 메서드는 스토리지 개체에 대한 권한 정보를 검색하여 진행률을 모니터링하기 위한 콜백 메커니즘을 제공합니다.
GetRootStorage GetRootStorage 메서드는 스토리지 매체의 루트 스토리지에 대한 IMDSPStorage 인터페이스에 대한 포인터를 검색합니다.
GetSCPSession GetSCPSession 메서드는 세션 개체를 나타내는 ISCPSecureQuery 인터페이스에 대한 포인터를 가져오는 데 사용됩니다.
GetSecureQuery GetSecureQuery 메서드는 ISCPSecureQuery 인터페이스에 대한 포인터를 가져오는 데 사용됩니다.
GetSecureQuery GetSecureQuery 메서드는 세션에 대한 보안 쿼리 개체를 가져오는 데 사용됩니다.
GetSerialNumber GetSerialNumber 메서드는 디바이스를 고유하게 식별하는 일련 번호를 검색합니다.
GetSerialNumber GetSerialNumber 메서드는 스토리지 매체를 고유하게 식별하는 일련 번호를 검색합니다. 이 메서드는 보호된 콘텐츠 전송을 위해 구현해야 하지만 그렇지 않으면 선택 사항입니다. 자세한 내용은 필수 및 선택적 인터페이스를 참조하세요.
GetSerialNumber GetSerialNumber 메서드는 디바이스를 고유하게 식별하는 일련 번호를 검색합니다.

GetSerialNumber GetSerialNumber 메서드는 스토리지 매체를 고유하게 식별하는 일련 번호를 검색합니다.
GetSize GetSize 메서드는 스토리지 개체의 크기를 바이트 단위로 검색합니다.
GetSize GetSize 메서드는 스토리지 크기를 바이트 단위로 검색합니다.
GetSizeParams GetSizeParams 메서드는 현재 로그 파일의 현재 크기 매개 변수를 검색합니다.
GetSpecifiedMetadata GetSpecifiedMetadata 메서드는 스토리지에 대해 지정된 메타데이터 개체만 검색합니다.
GetSpecifiedMetadata GetSpecifiedMetadata 메서드는 스토리지에서 하나 이상의 특정 메타데이터 속성을 검색합니다.
GetSpecifyPropertyPages GetSpecifyPropertyPages 메서드는 휴대용 디바이스의 비표준 기능을 설명하는 속성 페이지를 가져옵니다.
GetSpecifyPropertyPages GetSpecifyPropertyPages 메서드는 디바이스의 속성 페이지를 검색합니다. 속성 페이지를 사용하여 디바이스별 속성 및 브랜딩 정보를 보고할 수 있습니다.
GetStatus GetStatus 메서드는 디바이스에서 제공할 수 있는 모든 디바이스 상태 정보를 검색합니다.
GetStatus GetStatus 메서드는 스토리지 매체의 현재 상태 검색합니다.
GetStatus GetStatus 메서드는 디바이스 상태 정보를 검색합니다.
GetStatus GetStatus 메서드는 디바이스의 컨트롤 상태 검색합니다.

GetStatus GetStatus 메서드는 스토리지 매체의 현재 상태 검색합니다.
GetStorage GetStorage 메서드를 사용하면 모든 스토리지를 열거하는 대신 이름을 기반으로 스토리지로 직접 이동하여 찾을 수 있습니다.
GetStorage GetStorage 메서드를 사용하면 모든 스토리지를 열거하는 대신 스토리지 이름에서 스토리지 개체로 직접 이동하여 찾을 수 있습니다.
GetStorage GetStorage 메서드는 루트 스토리지의 직속 자식에서 지정된 이름의 스토리지를 검색합니다.
GetStorage GetStorage 메서드는 모든 자식을 열거하지 않고도 현재 스토리지에서 이름으로 자식 스토리지를 직접 검색합니다.
GetStorageGlobals GetStorageGlobals 메서드는 IMDSPStorageGlobals 인터페이스를 검색하여 스토리지 매체에 대한 전역 정보에 대한 액세스를 제공합니다.
GetStorageGlobals GetStorageGlobals 메서드는 이 스토리지의 루트 스토리지에 대한 IWMDMStorageGlobals 인터페이스를 검색합니다.
GetTotalBad GetTotalBad 메서드는 스토리지 매체에서 사용할 수 없는 공간의 총 크기를 바이트 단위로 검색합니다. (IMDSPStorageGlobals.GetTotalBad)
GetTotalBad GetTotalBad 메서드는 스토리지 매체에서 사용할 수 없는 공간의 총 크기를 바이트 단위로 검색합니다. (IWMDMStorageGlobals.GetTotalBad)
GetTotalFree GetTotalFree 메서드는 스토리지 매체의 사용 가능한 총 공간을 바이트 단위로 검색합니다.
GetTotalFree

GetTotalFree 메서드는 스토리지 매체의 사용 가능한 총 공간을 바이트 단위로 검색합니다.
GetTotalLength GetTotalLength 메서드는 개체와 관련한 단위로 개체의 총 재생 길이를 검색합니다. 반환되는 값은 재생 길이 및 오프셋의 현재 설정에 관계없이 총 길이입니다.
GetTotalLength GetTotalLength 메서드는 개체의 총 재생 길이를 형식에 적합한 단위로 검색합니다. 반환되는 값은 재생 길이 및 오프셋의 현재 설정에 관계없이 총 길이입니다.
GetTotalSize GetTotalSize 메서드는 이 IMDSPStorageGlobals 인터페이스와 연결된 매체의 총 크기(바이트)를 검색합니다.
GetTotalSize GetTotalSize 메서드는 IWMDMStorageGlobals 인터페이스와 연결된 스토리지 매체의 총 크기(바이트)를 검색합니다.
GetType GetType 메서드는 디바이스 유형 정보를 검색합니다.
GetType GetType 메서드는 디바이스에서 지원하는 작업을 검색합니다.
GetVersion GetVersion 메서드는 디바이스의 버전 번호를 검색합니다.
GetVersion GetVersion 메서드는 디바이스의 제조업체 정의 버전 번호를 검색합니다.
초기화 Initialize 메서드는 스토리지 매체의 형식을 지정합니다. (IMDSPStorageGlobals.Initialize)
초기화 Initialize 메서드는 스토리지 매체의 형식을 지정합니다. (IWMDMStorageGlobals.Initialize)
삽입 Insert 메서드는 디바이스의 스토리지에 콘텐츠를 넣습니다.

Insert2 Insert2 메서드는 콘텐츠를 스토리지 옆에 배치합니다. 이 메서드는 애플리케이션이 새 대상 이름을 지정하고 사용자 지정 COM 개체에 대한 포인터를 제공할 수 있도록 하여 IWMDMStorageControl::Insert를 확장합니다.
Insert3 Insert3 메서드는 콘텐츠를 스토리지 옆에 배치합니다. 이 메서드는 애플리케이션이 전송되는 개체의 메타데이터 및 형식을 명시적으로 지정할 수 있도록 하여 IWMDMStorageControl2::Insert2를 확장합니다.
IsEnabled IsEnabled 메서드는 로깅이 사용되는지 여부를 결정합니다.
LogDword LogDword 메서드는 DWORD 값을 현재 로그 파일에 기록합니다. 캐리지 리턴 및 줄 바꿈이 각 로그 항목에 추가됩니다.
LogString LogString 메서드는 문자열을 현재 로그 파일에 기록합니다. 캐리지 리턴 및 줄 바꿈이 각 로그 항목에 추가됩니다.
MakeDecision MakeDecision 메서드는 콘텐츠에 대한 액세스가 허용되는지 여부를 결정합니다. 액세스가 허용되는 경우 이 메서드는 콘텐츠에 액세스하는 데 사용할 인터페이스를 반환합니다.
MakeDecision2 MakeDecision2 메서드는 Windows Media 디바이스 관리자 이 메서드에 전달하는 데이터를 검사하여 보안 콘텐츠 공급자가 콘텐츠에 대한 책임이 있는지 여부를 결정합니다.
MakeDecisionOnClearChannel MakeDecisionOnClearChannel 메서드는 콘텐츠에 대한 액세스가 명확한 채널에서 허용되는지 여부를 결정합니다. 액세스가 허용되는 경우 이 메서드는 콘텐츠에 액세스하는 데 사용되는 인터페이스를 반환합니다.
이동 Move 메서드는 미디어 디바이스에서 파일 또는 폴더를 이동합니다.
이동 Move 메서드는 현재 스토리지를 디바이스의 새 위치로 이동합니다.

다음 Next 메서드는 다음 celtIMDSPDevice 인터페이스에 대한 포인터를 검색합니다.
다음 Next 메서드는 다음 celtIMDSPStorage 인터페이스에 대한 포인터를 반환합니다.
다음 Next 메서드는 IWMDMDevice 인터페이스로 표시되는 다음 디바이스에 대한 포인터를 반환합니다.
다음 Next 메서드는 다음 형제 스토리지에 대한 포인터를 검색합니다.
ObjectData ObjectData 메서드는 개체 데이터 블록을 Windows Media 디바이스 관리자 다시 전송합니다.
열기 Open 메서드는 연결된 개체를 열고 읽기 또는 쓰기 작업을 위해 준비합니다. 이 작업은 스토리지 개체가 파일을 나타내는 경우에만 유효합니다.
일시 중지 Pause 메서드는 콘텐츠 내의 현재 위치에서 현재 재생 또는 레코드 세션을 일시 중지합니다. (IMDSPDeviceControl.Pause)
일시 중지 Pause 메서드는 콘텐츠 내의 현재 위치에서 현재 재생 또는 레코드 세션을 일시 중지합니다. (IWMDMDeviceControl.Pause)
재생 Play 메서드는 현재 검색 위치에서 재생을 시작합니다. Seek 메서드가 호출되지 않은 경우 첫 번째 파일의 시작 부분에서 재생이 시작되고 재생 길이가 정의되지 않습니다.
재생 Play 메서드는 현재 검색 위치에서 재생을 시작합니다. IWMDMDeviceControl::Seek 메서드가 호출되지 않은 경우 첫 번째 파일의 시작 부분에서 재생이 시작되고 재생 길이가 정의되지 않습니다.
진행률 Progress 메서드는 작업이 아직 진행 중임을 나타냅니다.

Progress3 Progress3 메서드는 진행 중인 작업의 상태 나타내기 위해 Windows Media 디바이스 관리자 호출됩니다.
QueryByIndex QueryByIndex 메서드는 인덱스로 지정된 속성의 값을 검색합니다.
QueryByName QueryByName 메서드는 이름으로 지정된 속성의 값을 검색합니다.
읽기 Read 메서드는 현재 위치의 개체에서 데이터를 읽습니다. 이 작업은 스토리지 개체가 파일을 나타내는 경우에만 유효합니다.
읽기 Read 메서드는 현재 스토리지를 컴퓨터에 복사합니다.
ReadOnClearChannel ReadOnClearChannel 메서드는 보안 인증 채널을 사용하지 않고 현재 위치의 개체에서 데이터를 읽습니다.
레코드 Record 메서드는 현재 검색 위치에 있는 디바이스의 외부 레코드 입력에서 녹음/녹화를 시작합니다. Seek 메서드를 먼저 호출해야 합니다.
레코드 Record 메서드는 현재 검색 위치에 있는 디바이스의 외부 레코드 입력에서 녹음/녹화를 시작합니다. IWMDMDeviceControl::Seek 메서드를 먼저 호출해야 합니다.
다시 초기화 Reinitialize 메서드는 Windows Media 디바이스 관리자 모든 Windows Media 디바이스 관리자 디바이스를 다시 검색하도록 강제합니다.
이름 바꾸기 Rename 메서드는 파일 또는 폴더일 수 있는 연결된 개체의 이름을 바꿉니다.
이름 바꾸기 Rename 메서드는 현재 스토리지의 이름을 바꿉니다.

Reset Reset 메서드는 열거형 시퀀스를 시작 부분으로 다시 설정합니다. Next에 대한 후속 호출은 열거형 시퀀스에서 첫 번째 Windows Media 디바이스 관리자 인터페이스를 가져옵니다.
Reset Reset 메서드는 열거형 시퀀스를 시작 부분으로 다시 설정합니다. Next 메서드에 대한 후속 호출은 열거형 시퀀스에서 첫 번째 스토리지 인터페이스를 가져옵니다.
Reset Reset 메서드는 Next가 첫 번째 디바이스에 대한 포인터를 반환하도록 열거형을 다시 설정합니다.
Reset Reset 메서드는 열거형 시퀀스를 다시 시작 부분으로 설정합니다.
Reset Reset 메서드는 현재 로그 파일의 내용을 삭제합니다.
다시 시작 Resume 메서드는 Pause를 호출하는 동안 저장된 파일 위치에서 현재 재생 또는 레코드 작업을 다시 시작합니다.
다시 시작 Resume 메서드는 Pause를 호출하는 동안 저장된 파일 위치에서 현재 재생 또는 레코드 작업을 다시 시작합니다.
SACAuth SACAuth 메서드는 구성 요소 간에 보안 인증 채널을 설정합니다.
SACGetProtocols SACGetProtocols 메서드는 구성 요소에서 다른 구성 요소에서 지원하는 인증 프로토콜을 검색하는 데 사용됩니다.
Seek Seek 메서드는 Play 또는 Record 메서드에서 시작점으로 사용되는 위치를 찾습니다. (IMDSPDeviceControl.Seek)
Seek

Seek 메서드는 개체 내의 현재 위치를 설정합니다. 이 작업은 스토리지 개체가 파일을 나타내는 경우에만 유효합니다.
Seek Seek 메서드는 Play 또는 Record 메서드에서 시작점으로 사용되는 위치를 찾습니다. (IWMDMDeviceControl.Seek)
SendOpaqueCommand SendOpaqueCommand 메서드는 Windows Media 디바이스 관리자 통해 명령을 보냅니다. Windows Media 디바이스 관리자 조치를 취하지 않고 명령을 통해 디바이스에 전달합니다.
SendOpaqueCommand SendOpaqueCommands 메서드는 Windows Media 디바이스 관리자 통해 명령을 보냅니다. Windows Media 디바이스 관리자 조치를 취하지 않고 명령을 통해 디바이스에 전달합니다.
SendOpaqueCommand SendOpaqueCommand 메서드는 Windows Media 디바이스 관리자 통해 디바이스 관련 명령을 디바이스에 보냅니다. Windows Media 디바이스 관리자 명령을 읽으려고 시도하지 않습니다.
SendOpaqueCommand SendOpaqueCommand 메서드는 Windows Media 디바이스 관리자 통해 명령을 처리하지 않고 스토리지에 보냅니다.
SetAttributes SetAttributes 메서드는 스토리지 개체의 특성을 설정합니다.
SetAttributes SetAttributes 메서드는 스토리지의 특성을 설정합니다.
SetAttributes2 SetAttributes2 메서드는 스토리지 개체의 오디오 및 비디오 형식 및 확장 특성을 설정할 수 있도록 하여 IMDSPStorage::SetAttributes를 확장합니다.
SetAttributes2 SetAttributes2 메서드는 스토리지의 확장 특성을 설정합니다.
SetDeviceEnumPreference SetDeviceEnumPreference 메서드는 디바이스 열거형 기본 설정을 지정합니다. (IMDServiceProvider3.SetDeviceEnumPreference)

SetDeviceEnumPreference SetDeviceEnumPreference 메서드는 디바이스 열거형 기본 설정을 지정합니다. (IWMDeviceManager3.SetDeviceEnumPreference)
SetEnumPreference SetEnumPreference 메서드는 스토리지에 대한 기본 보기 모드를 설정합니다.
SetLogFileName SetLogFileName 메서드는 현재 로그 파일의 전체 경로를 설정합니다. 모든 후속 로그 항목이 이 파일에 배치됩니다.
SetMetadata SetMetadata 메서드는 지정된 콘텐츠와 연결된 메타데이터를 제공합니다.
SetMetadata SetMetadata 메서드는 스토리지의 메타데이터를 설정합니다.
SetObjectAttributes SetObjectAttributes 메서드는 파일 특성을 지정합니다. 이 메서드는 현재 Windows Media 디바이스 관리자 호출되지 않습니다.
SetObjectAttributes2 SetObjectAttributes2 메서드는 파일 또는 스토리지의 특성을 설정합니다. 이 메서드는 현재 Windows Media 디바이스 관리자 호출되지 않습니다.
SetObjectName SetObjectName 메서드는 읽거나 쓰는 콘텐츠에 이름을 할당합니다. 이 메서드는 현재 Windows Media 디바이스 관리자 호출되지 않습니다.
SetObjectTotalSize SetObjectTotalSize 메서드는 개체의 총 크기를 바이트 단위로 할당합니다. 이 메서드는 현재 Windows Media 디바이스 관리자 호출되지 않습니다.
SetPlayLength SetPlayLength 메서드는 개체와 관련한 단위로 개체의 재생 길이를 설정합니다. 실제 길이에 관계없이 개체가 재생되는 최대 길이입니다.
SetPlayLength

SetPlayLength 메서드는 개체의 재생 길이를 형식에 적합한 단위로 설정합니다. 실제 길이에 관계 없이 개체가 재생되는 최대 길이입니다.
SetPlayOffset SetPlayOffset 메서드는 개체와 관련한 단위로 개체의 재생 오프셋을 설정합니다. IMDSPDeviceControl::P lay의 다음 호출에 대한 시작점을 지정합니다.
SetPlayOffset SetPlayOffset 메서드는 개체의 재생 오프셋을 형식에 적합한 단위로 설정합니다. 이는 Play의 다음 호출에 대한 시작점을 지정합니다.
SetProperty SetProperty 메서드는 쓰기 가능한 특정 디바이스 속성을 설정합니다.
SetProperty SetProperty 메서드는 쓰기 가능한 경우 특정 디바이스 속성을 설정합니다.
SetReferences SetReferences 메서드는 이 스토리지에 포함된 기존 참조를 덮어쓰는 참조(예: 재생 목록/앨범)가 있는 스토리지에 포함된 참조를 설정합니다.
SetReferences SetReferences 메서드는 스토리지에서 보유한 기존 참조를 덮어쓰는 참조(예: 재생 목록 또는 앨범)가 있는 스토리지에 포함된 참조를 설정합니다.
SetSizeParams SetSizeParams 메서드는 현재 로그 파일에 대한 현재 크기 매개 변수를 설정합니다.
Skip Skip 메서드는 열거형 시퀀스에서 지정된 다음 수의 미디어 디바이스 인터페이스를 건너뜁니다.
Skip Skip 메서드는 열거형 시퀀스에서 지정된 다음 수의 스토리지 인터페이스를 건너뜁니다.
Skip Skip 메서드는 열거형 시퀀스에서 지정된 수의 디바이스를 건너뜁니다.
Skip

Skip 메서드는 열거형 시퀀스에서 지정된 수의 스토리지를 건너뜁니다.
중지 Stop 메서드는 현재 스트림을 중지합니다.
중지 Stop 메서드는 현재 레코드 또는 재생 작업을 중지합니다.
TransferComplete TransferComplete 메서드는 Windows Media 디바이스 관리자 호출되어 데이터 보안 전송이 종료되었음을 알립니다. 이 메서드에서 보안 콘텐츠 공급자는 대상 디바이스에서 콘텐츠를 사용하도록 설정하는 데 필요한 추가 처리를 수행할 수 있습니다.
TransferCompleteForDevice TransferCompleteForDevice 메서드는 Windows Media 디바이스 관리자 특정 디바이스에 대한 데이터 전송의 끝을 알리기 위해 호출됩니다.
TransferContainerData TransferContainerData 메서드는 컨테이너 파일 데이터를 보안 콘텐츠 공급자로 전송합니다. 보안 콘텐츠 공급자는 컨테이너를 내부적으로 분해하고 컨테이너에서 추출할 때 사용할 수 있는 콘텐츠 부분을 보고합니다.
TransferContainerData2 TransferContainerData2 메서드는 컨테이너 파일 데이터를 보안 콘텐츠 공급자로 전송합니다.
TransferContainerDataOnClearChannel TransferContainerDataOnClearChannel 메서드는 지우기 채널을 통해 컨테이너 파일 데이터를 콘텐츠 공급자로 전송합니다.
TransferObjectData TransferObjectData 메서드는 애플리케이션이 컴퓨터와 데이터 블록을 전송할 수 있도록 호출됩니다.
TransferObjectDataOnClearChannel TransferObjectDataOnClearChannel 메서드는 IWMDMOperation::TransferObjectData의 보다 효율적인 구현입니다.
TransferToDevice

TransferToDevice 메서드는 Windows Media 디바이스 관리자 서비스 공급자에게 콘텐츠 전송 콘텐츠를 위임하기 위해 호출됩니다. 원본을 파일 또는 작업 인터페이스로 지정할 수 있습니다.
WMDMMessage WMDMMessage 메서드는 클라이언트에서 구현하고 플러그 앤 플레이 호환 디바이스 또는 스토리지 매체가 연결되거나 제거될 때 Windows Media 디바이스 관리자 호출하는 콜백 방법입니다.
쓰기 Write 메서드는 개체 내의 현재 위치에 있는 개체에 데이터를 씁니다. 이 작업은 스토리지 개체가 파일을 나타내는 경우에만 유효합니다.
WriteOnClearChannel WriteOnClearChannel 메서드는 보안 인증 채널을 사용하지 않고 개체 내의 현재 위치에 데이터를 개체에 씁니다.

인터페이스

IComponentAuthenticate IComponentAuthenticate 인터페이스는 Windows Media 디바이스 관리자 모듈 간에 안전하고 암호화된 통신을 제공합니다.
IMDServiceProvider IMDServiceProvider 인터페이스는 Windows Media 디바이스 관리자 서비스 공급자에 연결하는 데 사용하는 초기 인터페이스입니다.
IMDServiceProvider2 IMDServiceProvider2 인터페이스는 지정된 디바이스 경로 이름에 대한 IMDSPDevice 개체를 가져오는 방법을 제공하여 IMDServiceProvider 인터페이스를 확장합니다. 디바이스 경로 이름은 PnP(플러그 앤 플레이) 하위 시스템에 있습니다.
IMDServiceProvider3 IMDServiceProvider3 인터페이스는 디바이스 열거형 기본 설정을 위한 메서드를 제공하여 IMDServiceProvider2 인터페이스를 확장합니다.
IMDSPDevice IMDSPDevice 인터페이스는 미디어 디바이스와의 instance 기반 연결을 제공합니다.

IMDSPDevice2 IMDSPDevice2 인터페이스는 확장된 형식을 가져오고, PnP(플러그 앤 플레이) 디바이스 이름을 가져오고, 속성 페이지를 사용할 수 있게 하고, 이름에서 스토리지 매체에 대한 포인터를 가져올 수 있도록 하여 IMDSPDevice를 확장합니다.
IMDSPDevice3 IMDSPDevice3 인터페이스는 Windows 미디어 플레이어 동기화할 것으로 예상되는 디바이스에 대해 지원되어야 합니다.
IMDSPDeviceControl IMDSPDeviceControl 인터페이스는 디바이스를 제어하는 메서드를 제공합니다.
IMDSPDirectTransfer IMDSPDirectTransfer 인터페이스를 사용하면 Windows Media 디바이스 관리자 서비스 공급자에게 콘텐츠 전송을 위임할 수 있습니다.
IMDSPEnumDevice IMDSPEnumDevice 인터페이스는 미디어 디바이스를 열거하는 데 사용됩니다.
IMDSPEnumStorage IMDSPEnumStorage 인터페이스는 디바이스의 스토리지 미디어를 열거하는 데 사용됩니다.
IMDSPObject IMDSPObject 인터페이스는 스토리지 미디어와 데이터 전송을 관리합니다. Open, Read, Write 및 Close 메서드는 스토리지 개체가 파일인 경우에만 유효합니다.
IMDSPObject2 Windows Media 디바이스 관리자 IMDSPObject2를 사용하여 보다 효율적인 파일 읽기 및 쓰기를 지원합니다. 참고 서비스 공급자가 값이 1인 UseExtendedWmdm 디바이스 매개 변수를 추가하지 않으면 Windows Media 디바이스 관리자 이 인터페이스를 호출하지 않습니다. 이에 대한 자세한 내용은 디바이스 매개 변수를 참조하세요. .
IMDSPObjectInfo IMDSPObjectInfo 인터페이스는 IMDSPDeviceControl 인터페이스에서 재생 가능한 개체를 참조하거나 액세스하는 방법을 설명하는 매개 변수를 가져오고 설정하는 메서드를 제공합니다.
IMDSPRevoked IMDSPRevoked 인터페이스는 업데이트된 구성 요소를 다운로드할 수 있는 URL을 검색합니다. 이

인터페이스를 구현하는 것은 선택 사항입니다. 자세한 내용은 필수 및 선택적 인터페이스를 참조하세요.
IMDSPStorage IMDSPStorage 인터페이스는 디바이스의 스토리지 매체와 인스턴스 기반 연결을 제공합니다.
IMDSPStorage2 IMDSPStorage2 인터페이스는 확장 특성을 가져오고 설정하는 메서드를 제공하고 이름에서 스토리지 매체에 대한 포인터를 가져올 수 있도록 하여 IMDSPStorage를 확장합니다.
IMDSPStorage3 IMDSPStorage3 인터페이스는 메타데이터를 지원하여 IMDSPStorage2를 확장합니다.
IMDSPStorage4 IMDSPStorage4 인터페이스는 가상 스토리지(예: 재생 목록 및 앨범) 및 메타데이터를 지원하기 위해 IMDSPStorage3을 확장합니다. 참고 서비스 공급자가 값이 1인 UseExtendedWmdm 디바이스 매개 변수를 추가하지 않으면 Windows Media 디바이스 관리자 이 인터페이스를 호출하지 않습니다. 이에 대한 자세한 내용은 디바이스 매개 변수를 참조하세요. .
IMDSPStorageGlobals IMDSPStorage 인터페이스에서 가져온 IMDSPStorageGlobals 인터페이스는 스토리지 매체에 대한 전역 정보를 검색하는 메서드를 제공합니다. 여기에는 사용 가능한 공간의 양, 매체의 일련 번호 등이 포함될 수 있습니다.
ISCPSecureAuthenticate ISCPSecureAuthenticate 인터페이스는 보안 콘텐츠 공급자의 기본 인터페이스로, Windows Media는 보안 콘텐츠 공급자를 인증하고 보안 콘텐츠 공급자에 의해 인증되도록 쿼리를 디바이스 관리자.
ISCPSecureAuthenticate2 ISCPSecureAuthenticate2 인터페이스는 세션 개체를 가져오는 방법을 제공하여 ISCPSecureAuthenticate를 확장합니다.
ISCPSecureExchange ISCPSecureExchange 인터페이스는 콘텐츠와 연결된 보안 콘텐츠 및 권한을 교환하는 데 사용됩니다. 보안 콘텐츠 공급자는 이 인터페이스를 구현하고 보안 Windows Media 디바이스 관리자 구현에서 해당 메서드를 호출합니다.
ISCPSecureExchange2

ISCPSecureExchange2 인터페이스는 새 버전의 TransferContainerData 메서드를 제공하여 ISCPSecureExchange를 확장합니다.
ISCPSecureExchange3 ISCPSecureExchange3 인터페이스는 향상된 데이터 교환 성능과 전송 완료 콜백 메서드를 제공하여 ISCPSecureExchange2를 확장합니다.
ISCPSecureQuery ISCPSecureQuery 인터페이스는 Windows Media 디바이스 관리자 쿼리하여 보안 콘텐츠의 소유권을 확인합니다.
ISCPSecureQuery2 ISCPSecureQuery2 인터페이스는 보안 콘텐츠 공급자가 콘텐츠를 담당하는지 여부를 결정하는 기능을 통해 ISCPSecureQuery를 확장하고, 그렇다면 해지된 구성 요소를 업데이트하고 해지된 구성 요소를 결정하는 URL을 제공합니다.
ISCPSecureQuery3 ISCPSecureQuery3 인터페이스는 권한을 검색하고 명확한 채널에서 의사 결정을 내리기 위한 새로운 메서드 집합을 제공하여 ISCPSecureQuery2를 확장합니다.
ISCPSession ISCPSession 인터페이스는 여러 작업에 대한 효율적인 공통 상태 관리를 제공합니다. SCP(보안 콘텐츠 공급자) 세션은 여러 파일을 전송할 때 유용합니다.
IWMDeviceManager IWMDeviceManager 인터페이스는 애플리케이션에 대한 최상위 Windows Media 디바이스 관리자 인터페이스입니다.
IWMDeviceManager2 IWMDeviceManager2 인터페이스는 IWMDeviceManager 인터페이스를 확장합니다.
IWMDeviceManager3 IWMDeviceManager3 인터페이스는 디바이스 열거형 기본 설정을 설정하는 메서드를 제공하여 IWMDeviceManager2 인터페이스를 확장합니다.
IWMDMDevice IWMDMDevice 인터페이스는 단일 휴대용 디바이스를 검사하고 탐색하는 메서드를 제공합니다. 인터페이스를 사용하여 디바이스에 대한 정보를 얻고 해당 스토리지를 열거할 수 있습니다. IWMDMDevice2는 이 인터페이스의 기능을 확장합니다.

IWMDMDevice2 IWMDMDevice2 인터페이스는 디바이스에서 지원하는 비디오 형식을 얻고, 이름에서 스토리지를 찾고, 속성 페이지를 사용할 수 있도록 하여 IWMDMDevice를 확장합니다.
IWMDMDevice3 IWMDMDevice3 인터페이스는 디바이스에서 속성을 쿼리하고, 디바이스 I/O 제어 코드를 전송하는 메서드를 제공하고, 스토리지를 검색하고 디바이스 형식 기능을 검색하는 업그레이드된 메서드를 제공하여 IWMDMDevice2를 확장합니다.
IWMDMDeviceControl IWMDMDeviceControl 인터페이스는 디바이스에서 재생을 제어하는 메서드를 제공합니다.
IWMDMDeviceSession IWMDMDeviceSession 인터페이스는 여러 작업을 한 세션에 번들로 묶어 디바이스 작업의 효율성을 향상시킵니다.
IWMDMEnumDevice IWMDMEnumDevice 인터페이스는 컴퓨터에 연결된 휴대용 디바이스를 열거합니다. 이 인터페이스를 가져오려면 IWMDeviceManager::EnumDevices를 호출합니다.
IWMDMEnumStorage IWMDMEnumStorage 인터페이스는 디바이스의 스토리지를 열거합니다.
IWMDMLogger IWMDMLogger 인터페이스는 Windows Media 디바이스 관리자 애플리케이션 및 서비스 공급자가 공통 로그 파일에 항목을 기록하는 데 사용됩니다.
IWMDMMetaData IWMDMMetaData 인터페이스는 스토리지의 메타데이터 속성(예: 아티스트, 앨범, 장르 등)을 설정하고 검색합니다.
IWMDMNotification 선택적 애플리케이션 구현 IWMDMNotification 인터페이스를 사용하면 디바이스 또는 메모리 스토리지(예: RAM 카드)가 컴퓨터에서 연결되거나 연결이 끊어지면 애플리케이션 및 서비스 공급자가 알림을 받을 수 있습니다. 참고 이 메서드는 등록된 플러그 앤 플레이 디바이스에 대해서만 호출됩니다. 다른 디바이스 도착 또는 출발 시 이 인터페이스가 호출되지 않습니다. 이 인터페이스 GUID는 mssachlp.lib에 올바르게 정의되지 않았습니다. 따라서 다음을 수행해야 합니다.
IWMDMObjectInfo

IWMDMObjectInfo 인터페이스는 IWMDMDeviceControl 인터페이스에서 디바이스에서 재생 가능한 파일을 처리하는 방법을 제어하는 정보를 가져오고 설정합니다. 이 인터페이스는 재생할 수 없는 파일을 위한 것이 아닙니다.
IWMDMOperation 이 선택적 애플리케이션 구현 IWMDMOperation 인터페이스를 사용하면 애플리케이션에서 파일을 전송하는 동안 컴퓨터에서 데이터를 읽거나 쓰는 방법을 제어할 수 있습니다.
IWMDMOperation2 선택적 애플리케이션 구현 IWMDMOperation2 인터페이스는 확장 특성을 가져와서 설정하는 메서드를 제공하여 IWMDMOperation을 확장합니다.
IWMDMOperation3 선택적 애플리케이션 구현 IWMDMOperation3 인터페이스는 효율성을 높이기 위해 암호화되지 않은 데이터를 전송하는 새로운 방법을 제공하여 IWMDMOperation을 확장합니다.
IWMDMProgress 애플리케이션에서 구현한 선택적 IWMDMProgress를 사용하면 애플리케이션이 미디어 또는 파일 전송 서식 지정과 같은 작업의 진행률을 추적할 수 있습니다.
IWMDMProgress2 선택적 애플리케이션 구현 IWMDMProgress2 인터페이스는 상태 표시기를 제공하여 IWMDMProgress::End 메서드를 확장합니다.
IWMDMProgress3 선택적 애플리케이션 구현 IWMDMProgress3 인터페이스는 모니터링되는 이벤트를 지정하고 컨텍스트별 정보를 허용하는 추가 입력 매개 변수를 제공하여 IWMDMProgress2를 확장합니다. 이 콜백 인터페이스를 구현하는 애플리케이션은 새 메서드 외에도 이전 버전과의 호환성을 위해 IWMDMProgress 및 IWMDMProgress2에 해당하는 메서드에 대한 구현을 제공해야 합니다.
IWMDMRevoked IWMDMRevoked 인터페이스는 해지 오류로 인해 전송이 실패하는 경우 업데이트된 구성 요소를 다운로드할 수 있는 URL을 검색합니다.
IWMDMStorage IWMDMStorage 인터페이스의 instance 디바이스에서 스토리지(파일, 폴더 또는 재생 목록과 같은 데이터 또는 컬렉션 개체의 일반 이름)를 검사하고 탐색하는 메서드를 제공합니다.
IWMDMStorage2 IWMDMStorage2 인터페이스는 이름으로 자식 스토리지를 가져와 확장 특성을 가져와서 설정할

수 있도록 하여 IWMDMStorage를 확장합니다. IWMDMStorage3 인터페이스는 메타데이터를 지원하여 이 인터페이스를 확장합니다.
IWMDMStorage3 IWMDMStorage3 인터페이스는 메타데이터를 노출하여 IWMDMStorage2를 확장합니다.
IWMDMStorage4 IWMDMStorage4 인터페이스는 스토리지에 사용 가능한 메타데이터의 하위 집합을 검색하고 다른 스토리지에 대한 참조 목록을 설정 및 검색하는 메서드를 제공하여 IWMDMStorage3을 확장합니다.
IWMDMStorageControl IWMDMStorageControl 인터페이스는 스토리지, 디바이스 또는 디바이스와 컴퓨터 간에 파일을 삽입, 삭제 또는 이동하는 데 사용됩니다.
IWMDMStorageControl2 IWMDMStorageControl2 인터페이스는 스토리지에 콘텐츠를 삽입할 때 대상 파일의 이름을 설정할 수 있도록 하여 IWMDMStorageControl을 확장합니다.
IWMDMStorageControl3 IWMDMStorageControl3 인터페이스는 IWMDMMetaData 인터페이스 포인터를 허용하는 Insert 메서드를 제공하여 IWMDMStorageControl2를 확장합니다.
IWMDMStorageGlobals IWMDMStorageGlobals 인터페이스는 디바이스의 스토리지 매체(예: 플래시 ROM 카드)에 대한 전역 정보를 검색하는 방법을 제공합니다.

구조체

MTP_COMMAND_DATA_IN MTP_COMMAND_DATA_IN 구조에는 IWMDMDevice3::D eviceIoControl 메서드를 사용하여 디바이스로 전송되는 MTP(미디어 전송 프로토콜) 사용자 지정 명령이 포함되어 있습니다.
MTP_COMMAND_DATA_OUT MTP_COMMAND_DATA_OUT 구조에는 IWMDMDevice3::D eviceIoControl 호출을 종료할 때 디바이스 드라이버가 채우는 MTP(미디어 전송 프로토콜) 응답이 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

MAPI(Windows 메시징)

아티클2023. 08. 22.

MAPI(Windows 메시징) 기술 개요입니다.

MAPI(Windows 메시징)를 개발하려면 다음 헤더가 필요합니다.

- mapi.h
- mapiunicodehelp.h
- wabmem.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- MAPI(Windows 메시징)

Functions

MAPIADDRESS MAPIAddress 함수는 주소 목록 항목 집합을 만들거나 수정합니다.
MAPIDELETEMAIL MAPIDeleteMail 함수는 메시지를 삭제합니다.
MAPIDETAILS MAPIDetails 함수는 선택한 주소 목록 항목의 세부 정보가 포함된 대화 상자를 표시합니다.
MAPIFINDNEXT MAPIFindNext 함수는 지정된 유형의 들어오는 메시지의 다음(또는 첫 번째) 메시지 식별자를 검색합니다.
MAPIFreeBuffer MAPIFreeBuffer 함수(mapi.h)는 메시징 시스템에서 할당한 메모리를 해제합니다.
MAPIFREEBUFFER MAPIFreeBuffer 함수(wabmem.h)는 메시징 시스템에서 할당한 메모리를 해제합니다. (MAPIFreeBuffer)
MAPILOGOFF

MAPILogoff 함수는 메시징 시스템으로 세션을 종료합니다.
MAPILOGON MAPILogon 함수는 간단한 MAPI 세션을 시작하여 기본 메시지 저장소 및 주소록 공급자를 로드합니다.
MAPIREADMAIL MAPIReadMail 함수는 읽기용 메시지를 검색합니다.
MAPIRESOLVENAME MAPIResolveName 함수는 사용자가 입력한 메시지 받는 사람의 이름을 명확한 주소 목록 항목으로 변환합니다.
MAPISAVEMAIL MAPISaveMail 함수는 메시지를 메시지 저장소에 저장합니다.
MAPISENDDOCUMENTS MAPISendDocuments 함수는 하나 이상의 첨부 파일과 표지가 있는 표준 메시지를 보냅니다.
MAPISENDMAIL ANSI 메시지를 보냅니다.
MAPISendMailHelper 유니코드 메시지 정보를 가져와 MAPISendMailW를 사용하여 메시지를 보내거나 필요한 경우 메시지를 ANSI로 변환하고 MAPISendMail을 사용하여 메시지를 보냅니다.
MAPISENDMAILW 유니코드 메시지를 보냅니다. 이 함수는 ANSI 함수 MAPISendMail을 대체합니다.

구조체

MapiFileDesc MapiFileDesc 구조체에는 임시 파일로 저장된 메시지 첨부 파일이 포함된 파일에 대한 정보가 포함되어 있습니다.
MapiFileDescW

MapiFileDescW 구조체에는 임시 파일로 저장된 메시지 첨부 파일이 포함된 파일에 대한 정보가 포함됩니다. 해당 파일에는 정적 OLE 개체, 포함된 OLE 개체, 포함된 메시지 및 기타 형식의 파일이 포함될 수 있습니다.
MapiFileTagExt MapiFileTagExt 구조체는 생성 시 메시지 첨부 파일의 형식과 현재 인코딩 형식을 지정하므로 대상에서 원래 형식으로 복원할 수 있습니다.
MapiMessage MapiMessage 구조체에는 메시지에 대한 정보가 포함됩니다. 유니코드 지원의 경우 MapiMessageW 구조를 사용합니다.
MapiMessageW MapiMessageW 구조체에는 메시지에 대한 정보가 포함됩니다.
MapiRecipDesc MapiRecipDesc 구조에는 메시지 보낸 사람 또는 받는 사람에 대한 정보가 포함됩니다. 유니코드 지원의 경우 MapiRecipDescW 구조를 사용합니다.
MapiRecipDescW MapiRecipDescW 구조에는 메시지 보낸 사람 또는 받는 사람에 대한 정보가 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Windows Messenger

아티클2023. 03. 09.

Windows Messenger 기술 개요입니다.

Windows Messenger를 개발하려면 다음 헤더가 필요합니다.

- comsvcs.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows Messenger

Windows Mixed Reality

아티클2023. 08. 22.

Windows Mixed Reality 기술 개요입니다.

Windows Mixed Reality 개발하려면 다음 헤더가 필요합니다.

- holographicspaceinterop.h
- spatialinteractionmanagerinterop.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows Mixed Reality

Functions

CreateForWindow HolographicSpace 개체를 인스턴스화하고 현재 애플리케이션에 바인딩합니다.
GetForWindow 활성 애플리케이션에 바인딩된 SpatialInteractionManager 개체를 검색합니다.

인터페이스

IHolographicSpaceInterop WinRT HolographicSpace 개체와의 상호 운용성을 사용하도록 설정하고 홀로그램 장면을 나타내기 위해 HolographicSpace 멤버에 대한 액세스를 제공합니다.
ISpatialInteractionManagerInterop WinRT SpatialInteractionManager 개체와의 상호 운용성을 사용하도록 설정하고, 손, 모션 컨트롤러 및 시스템 음성 명령의 사용자 입력에 액세스하기 위해 SpatialInteractionManager 멤버에 대한 액세스를 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Windows 멀티미디어

아티클2023. 08. 22.

Windows 멀티미디어 기술 개요입니다.

Windows 멀티미디어를 개발하려면 다음 헤더가 필요합니다.

- digitalv.h
- joystickapi.h
- mmddk.h
- mmeapi.h
- mmiscapi.h
- mmreg.h
- msacm.h
- timeapi.h
- vfw.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 멀티미디어

Functions

acmDriverAdd acmDriverAdd 함수(msacm.h)는 사용 가능한 ACM 드라이버 목록에 드라이버를 추가합니다. (acmDriverAddW)
acmDriverAddA acmDriverAdd 함수는 사용 가능한 ACM 드라이버 목록에 드라이버를 추가합니다. (acmDriverAddA)
acmDriverAddW acmDriverAddW(유니코드) 함수(msacm.h)는 사용 가능한 ACM 드라이버 목록에 드라이버를 추가합니다. (acmDriverAddW)
acmDriverClose acmDriverClose 함수는 이전에 연 ACM 드라이버 instance 닫습니다. 함수가 성공하면 핸들이 무효화됩니다.

acmDriverDetails acmDriverDetails 함수(msacm.h)는 지정된 ACM 드라이버를 쿼리하여 해당 기능을 확인합니다. (acmDriverDetailsW)
acmDriverDetailsA acmDriverDetails 함수는 지정된 ACM 드라이버를 쿼리하여 해당 기능을 확인합니다. (acmDriverDetailsA)
acmDriverDetailsW acmDriverDetails(유니코드) 함수는 지정된 ACM 드라이버를 쿼리하여 해당 기능을 확인합니다. (acmDriverDetailsW)
acmDriverEnum acmDriverEnum 함수는 사용 가능한 ACM 드라이버를 열거하며, 드라이버가 더 이상 없거나 콜백 함수가 FALSE를 반환할 때까지 계속됩니다.
ACMDRIVERENUMCB acmDriverEnumCallback 함수는 acmDriverEnum 함수와 함께 사용되는 콜백 함수를 지정합니다. acmDriverEnumCallback 이름은 애플리케이션 정의 함수 이름의 자리 표시자입니다.
acmDriverID acmDriverID 함수는 열려 있는 ACM 드라이버 instance 또는 스트림 핸들과 연결된 ACM 드라이버 식별자의 핸들을 반환합니다.
acmDriverMessage acmDriverMessage 함수는 지정된 ACM 드라이버 instance 사용자 정의 메시지를 보냅니다.
acmDriverOpen acmDriverOpen 함수는 지정된 ACM 드라이버를 열고 드라이버와 통신하는 데 사용할 수 있는 드라이버 instance 핸들을 반환합니다.
acmDriverPriority acmDriverPriority 함수는 ACM 드라이버의 우선 순위와 상태를 수정합니다.
ACMDRIVERPROC acmDriverProc 함수는 ACM 드라이버와 함께 사용되는 콜백 함수를 지정합니다.
acmDriverRemove

acmDriverRemove 함수는 사용 가능한 ACM 드라이버 목록에서 ACM 드라이버를 제거합니다. 호출 애플리케이션에 대해서만 드라이버가 제거됩니다. 드라이버가 전역적으로 설치된 경우에도 다른 애플리케이션에서 이 드라이버를 사용할 수 있습니다.
acmFilterChoose acmFilterChoose 함수는 사용자가 파형 오디오 필터를 선택할 수 있도록 하는 ACM 정의 대화 상자를 만듭니다.
acmFilterChooseA acmFilterChoose 함수는 사용자가 파형 오디오 필터를 선택할 수 있도록 하는 ACM 정의 대화 상자를 만듭니다. (acmFilterChooseA)
ACMFILTERCHOOSEHOOKPROC acmFilterChooseHookProc 함수는 acmFilterChoose 대화 상자를 연결하는 사용자 정의 함수를 지정합니다.
acmFilterChooseW acmFilterChoose(유니코드) 함수는 사용자가 파형 오디오 필터를 선택할 수 있도록 하는 ACM 정의 대화 상자를 만듭니다. (acmFilterChooseW)
acmFilterDetails acmFilterDetails 함수는 특정 파형 오디오 필터 태그가 있는 필터에 대한 세부 정보를 ACM에 쿼리합니다.
acmFilterDetailsA acmFilterDetails 함수는 특정 파형 오디오 필터 태그가 있는 필터에 대한 세부 정보를 ACM에 쿼리합니다. (acmFilterDetailsA)
acmFilterDetailsW acmFilterDetails(유니코드) 함수는 특정 파형 오디오 필터 태그가 있는 필터에 대한 세부 정보를 ACM에 쿼리합니다. (acmFilterDetailsW)
acmFilterEnum acmFilterEnum 함수는 ACM 드라이버에서 지정된 필터 태그에 사용할 수 있는 파형 오디오 필터를 열거합니다. 이 함수는 필터 태그에 더 이상 적합한 필터가 없거나 콜백 함수가 FALSE를 반환할 때까지 계속 열거합니다.
acmFilterEnumA acmFilterEnum 함수는 ACM 드라이버에서 지정된 필터 태그에 사용할 수 있는 파형 오디오 필터를

열거합니다. 이 함수는 필터 태그에 더 이상 적합한 필터가 없거나 콜백 함수가 FALSE를 반환할 때까지 계속 열거합니다. (acmFilterEnumA)
ACMFILTERENUMCB ACMFILTERENUMCB 콜백 함수는 acmFilterEnum 함수와 함께 사용되는 콜백 함수를 지정합니다.
ACMFILTERENUMCBA acmFilterEnumCallback 함수는 acmFilterEnum 함수와 함께 사용되는 콜백 함수를 지정합니다. acmFilterEnumCallback 이름은 애플리케이션 정의 함수 이름의 자리 표시자입니다. (ACMFILTERENUMCBA)
ACMFILTERENUMCBW ACMFILTERENUMCBW(유니코드) 콜백 함수는 acmFilterEnum 함수와 함께 사용되는 콜백 함수를 지정합니다.
acmFilterEnumW acmFilterEnum(유니코드) 함수는 ACM 드라이버에서 지정된 필터 태그에 사용할 수 있는 파형 오디오 필터를 열거합니다. 이 함수는 필터 태그에 더 이상 적합한 필터가 없거나 콜백 함수가 FALSE를 반환할 때까지 계속 열거합니다. (acmFilterEnumW)
acmFilterTagDetails acmFilterTagDetails 함수는 특정 파형 오디오 필터 태그에 대한 세부 정보를 ACM에 쿼리합니다.
acmFilterTagDetailsA acmFilterTagDetails 함수는 특정 파형 오디오 필터 태그에 대한 세부 정보를 ACM에 쿼리합니다. (acmFilterTagDetailsA)
acmFilterTagDetailsW acmFilterTagDetails(유니코드) 함수는 특정 파형 오디오 필터 태그에 대한 세부 정보를 ACM에 쿼리합니다. (acmFilterTagDetailsW)
acmFilterTagEnum acmFilterTagEnum 함수는 ACM 드라이버에서 사용할 수 있는 파형 오디오 필터 태그를 열거합니다. 이 함수는 더 이상 적합한 필터 태그가 없거나 콜백 함수가 FALSE를 반환할 때까지 계속 열거합니다.
acmFilterTagEnumA acmFilterTagEnum 함수는 ACM 드라이버에서 사용할 수 있는 파형 오디오 필터 태그를 열거합니다. 이 함수는 더 이상 적합한 필터 태그가 없거나 콜백 함수가 FALSE를 반환할 때까지 계속 열거합니다. (acmFilterTagEnumA)

ACMFILTERTAGENUMCB ACMFILTERTAGENUMCB 콜백 함수는 acmFilterTagEnum 함수와 함께 사용되는 콜백 함수를 지정합니다.
ACMFILTERTAGENUMCBA acmFilterTagEnumCallback 함수는 acmFilterTagEnum 함수와 함께 사용되는 콜백 함수를 지정합니다. acmFilterTagEnumCallback 함수 이름은 애플리케이션 정의 함수 이름의 자리 표시자입니다. (ACMFILTERTAGENUMCBA)
ACMFILTERTAGENUMCBW ACMFILTERTAGENUMCBW(유니코드) 콜백 함수는 acmFilterTagEnum 함수와 함께 사용되는 콜백 함수를 지정합니다.
acmFilterTagEnumW acmFilterTagEnum(유니코드) 함수는 ACM 드라이버에서 사용할 수 있는 파형 오디오 필터 태그를 열거합니다. 이 함수는 더 이상 적합한 필터 태그가 없거나 콜백 함수가 FALSE를 반환할 때까지 계속 열거합니다. (acmFilterTagEnumW)
acmFormatChoose acmFormatChoose 함수는 사용자가 파형 오디오 형식을 선택할 수 있도록 하는 ACM 정의 대화 상자를 만듭니다.
acmFormatChooseA acmFormatChoose 함수는 사용자가 파형 오디오 형식을 선택할 수 있도록 하는 ACM 정의 대화 상자를 만듭니다. (acmFormatChooseA)
ACMFORMATCHOOSEHOOKPROC acmFormatChooseHookProc 함수는 acmFormatChoose 대화 상자를 후크하는 사용자 정의 함수를 지정합니다. acmFormatChooseHookProc 이름은 애플리케이션 정의 이름의 자리 표시자입니다.
acmFormatChooseW acmFormatChoose(유니코드) 함수는 사용자가 파형 오디오 형식을 선택할 수 있도록 하는 ACM 정의 대화 상자를 만듭니다. (acmFormatChooseW)
acmFormatDetails acmFormatDetails 함수는 특정 파형 오디오 형식 태그에 대한 형식 세부 정보를 ACM에 쿼리합니다.
acmFormatDetailsA

acmFormatDetails 함수는 특정 파형 오디오 형식 태그에 대한 형식 세부 정보를 ACM에 쿼리합니다. (acmFormatDetailsA)
acmFormatDetailsW acmFormatDetails(유니코드) 함수는 특정 파형 오디오 형식 태그에 대한 형식 세부 정보를 ACM에 쿼리합니다. (acmFormatDetailsW)
acmFormatEnum acmFormatEnum 함수는 ACM 드라이버에서 지정된 형식 태그에 사용할 수 있는 파형 오디오 형식을 열거합니다. 이 함수는 서식 태그에 더 이상 적합한 형식이 없거나 콜백 함수가 FALSE를 반환할 때까지 계속 열거합니다.
acmFormatEnumA acmFormatEnum 함수는 ACM 드라이버에서 지정된 형식 태그에 사용할 수 있는 파형 오디오 형식을 열거합니다. 이 함수는 서식 태그에 더 이상 적합한 형식이 없거나 콜백 함수가 FALSE를 반환할 때까지 계속 열거합니다. (acmFormatEnumA)
ACMFORMATENUMCB ACMFORMATENUMCB 콜백 함수는 acmFormatEnum 함수와 함께 사용되는 콜백 함수를 지정합니다.
ACMFORMATENUMCBA acmFormatEnumCallback 함수는 acmFormatEnum 함수와 함께 사용되는 콜백 함수를 지정합니다. acmFormatEnumCallback 이름은 애플리케이션 정의 함수 이름의 자리 표시자입니다. (ACMFORMATENUMCBA)
ACMFORMATENUMCBW ACMFORMATENUMCBW(유니코드) 콜백 함수는 acmFormatEnum 함수와 함께 사용되는 콜백 함수를 지정합니다.
acmFormatEnumW acmFormatEnum(유니코드) 함수는 ACM 드라이버에서 지정된 형식 태그에 사용할 수 있는 파형 오디오 형식을 열거합니다. 이 함수는 서식 태그에 더 이상 적합한 형식이 없거나 콜백 함수가 FALSE를 반환할 때까지 계속 열거합니다. (acmFormatEnumW)
acmFormatSuggest acmFormatSuggest 함수는 ACM 또는 지정된 ACM 드라이버를 쿼리하여 제공된 원본 형식에 대한 대상 형식을 제안합니다.
acmFormatTagDetails

acmFormatTagDetails 함수는 특정 파형 오디오 형식 태그에 대한 자세한 내용을 ACM에 쿼리합니다.
acmFormatTagDetailsA acmFormatTagDetails 함수는 특정 파형 오디오 형식 태그에 대한 자세한 내용을 ACM에 쿼리합니다. (acmFormatTagDetailsA)
acmFormatTagDetailsW acmFormatTagDetails(유니코드) 함수는 특정 파형 오디오 형식 태그에 대한 자세한 내용을 ACM에 쿼리합니다. (acmFormatTagDetailsW)
acmFormatTagEnum acmFormatTagEnum 함수는 ACM 드라이버에서 사용할 수 있는 파형 오디오 형식 태그를 열거합니다. 이 함수는 더 이상 적합한 형식 태그가 없거나 콜백 함수가 FALSE를 반환할 때까지 계속 열거합니다.
acmFormatTagEnumA acmFormatTagEnum 함수는 ACM 드라이버에서 사용할 수 있는 파형 오디오 형식 태그를 열거합니다. 이 함수는 더 이상 적합한 형식 태그가 없거나 콜백 함수가 FALSE를 반환할 때까지 계속 열거합니다. (acmFormatTagEnumA)
ACMFORMATTAGENUMCB ACMFORMATTAGENUMCB 콜백 함수는 acmFormatTagEnum 함수와 함께 사용되는 콜백 함수를 지정합니다.
ACMFORMATTAGENUMCBA acmFormatTagEnumCallback 함수는 acmFormatTagEnum 함수와 함께 사용되는 콜백 함수를 지정합니다. acmFormatTagEnumCallback 이름은 애플리케이션 정의 함수 이름의 자리 표시자입니다. (ACMFORMATTAGENUMCBA)
ACMFORMATTAGENUMCBW ACMFORMATTAGENUMCBW(유니코드) 콜백 함수는 acmFormatTagEnum 함수와 함께 사용되는 콜백 함수를 지정합니다.
acmFormatTagEnumW acmFormatTagEnum(유니코드) 함수는 ACM 드라이버에서 사용할 수 있는 파형 오디오 형식 태그를 열거합니다. 이 함수는 더 이상 적합한 형식 태그가 없거나 콜백 함수가 FALSE를 반환할 때까지 계속 열거합니다. (acmFormatTagEnumW)
acmGetVersion

acmGetVersion 함수는 ACM의 버전 번호를 반환합니다.
acmMetrics acmMetrics 함수는 ACM 또는 관련 ACM 개체에 대한 다양한 메트릭을 반환합니다.
acmStreamClose acmStreamClose 함수는 ACM 변환 스트림을 닫습니다. 함수가 성공하면 핸들이 무효화됩니다.
acmStreamConvert acmStreamConvert 함수는 지정된 변환 스트림에서 변환을 수행하도록 ACM에 요청합니다. 변환은 스트림이 열린 방법에 따라 동기 또는 비동기일 수 있습니다.
acmStreamMessage acmStreamMessage 함수는 ACM 드라이버에 드라이버 관련 메시지를 보냅니다.
acmStreamOpen acmStreamOpen 함수는 ACM 변환 스트림을 엽니다. 변환 스트림은 지정된 오디오 형식에서 다른 오디오 형식으로 데이터를 변환하는 데 사용됩니다.
acmStreamPrepareHeader acmStreamPrepareHeader 함수는 ACM 스트림 변환을 위해 ACMSTREAMHEADER 구조를 준비합니다.
acmStreamReset acmStreamReset 함수는 지정된 ACM 스트림에 대한 변환을 중지합니다. 보류 중인 모든 버퍼는 완료된 것으로 표시되고 애플리케이션에 반환됩니다.
acmStreamSize acmStreamSize 함수는 ACM 스트림의 원본 또는 대상 버퍼에 권장되는 크기를 반환합니다.
acmStreamUnprepareHeader acmStreamUnprepareHeader 함수는 ACM 스트림에 대한 acmStreamPrepareHeader 함수에서 수행하는 준비를 정리합니다.
auxGetDevCaps auxGetDevCaps 함수(mmeapi.h)는 지정된 보조 출력 디바이스의 기능을 검색합니다.
auxGetDevCapsA

auxGetDevCaps 함수는 지정된 보조 출력 디바이스의 기능을 검색합니다. (auxGetDevCapsA)

auxGetDevCapsW

auxGetDevCapsW(유니코드) 함수(mmeapi.h)는 지정된 보조 출력 디바이스의 기능을 검색합니다.

auxGetNumDevs

auxGetNumDevs 함수는 시스템에 있는 보조 출력 디바이스의 수를 검색합니다.

auxGetVolume

auxGetVolume 함수는 지정된 보조 출력 디바이스의 현재 볼륨 설정을 검색합니다.

auxOutMessage

auxOutMessage 함수는 지정된 보조 출력 디바이스에 메시지를 보냅니다. 또한 이 함수는 메시지의 일부로 전달된 디바이스 식별자에서 오류 검사를 수행합니다.

auxSetVolume

auxSetVolume 함수는 지정된 보조 출력 디바이스의 볼륨을 설정합니다.

AVIBuildFilterA

AVIBuildFilter 함수는 GetOpenFileName 또는 GetSaveFileName 함수에서 이후에 사용되는 필터 사양을 빌드합니다. (ANSI)

AVIBuildFilterW

AVIBuildFilter 함수는 GetOpenFileName 또는 GetSaveFileName 함수에서 이후에 사용되는 필터 사양을 빌드합니다. (유니코드)

AVIClearClipboard

AVIClearClipboard 함수는 클립보드에서 AVI 파일을 제거합니다.

AVIFileAddRef

AVIFileAddRef 함수는 AVI 파일의 참조 수를 증가합니다.

AVIFileCreateStream

AVIFileCreateStream 함수(vfw.h)는 기존 파일에 새 스트림을 만들고 새 스트림에 대한 인터페이스를 만듭니다.

AVIFileCreateStreamA

AVIFileCreateStream 함수는 기존 파일에 새 스트림을 만들고 새 스트림에 대한 인터페이스를 만듭니다. (AVIFileCreateStreamA)
AVIFileCreateStreamW AVIFileCreateStreamW(유니코드) 함수(vfw.h)는 기존 파일에 새 스트림을 만들고 새 스트림에 대한 인터페이스를 만듭니다.
AVIFileEndRecord AVIFileEndRecord 함수는 비디오의 1:1 인터리브 요소를 오디오 데이터에 사용하는 인터리브 파일을 작성할 때 레코드의 끝을 표시합니다. (비디오의 각 프레임은 동일한 양의 오디오 데이터와 산재합니다.)
AVIFileExit AVIFileExit 함수는 AVIFile 라이브러리를 종료하고 라이브러리에 대한 참조 수를 감소합니다.
AVIFileGetStream AVIFileGetStream 함수는 지정된 AVI 파일과 연결된 스트림 인터페이스의 주소를 반환합니다.
AVIFileInfo AVIFileInfo 함수(vfw.h)는 AVI 파일에 대한 정보를 가져옵니다.
AVIFileInfoA AVIFileInfo 함수는 AVI 파일에 대한 정보를 가져옵니다. (AVIFileInfoA)
AVIFileInfoW AVIFileInfoW 함수(vfw.h)는 AVI 파일에 대한 정보를 가져옵니다.
AVIFileInit AVIFileInit 함수는 AVIFile 라이브러리를 초기화합니다.
AVIFileOpen AVIFileOpen 함수(vfw.h)는 AVI 파일을 열고 액세스하는 데 사용되는 파일 인터페이스의 주소를 반환합니다.
AVIFileOpenA AVIFileOpen 함수는 AVI 파일을 열고 액세스하는 데 사용되는 파일 인터페이스의 주소를 반환합니다. (AVIFileOpenA)
AVIFileOpenW

AVIFileOpenW 함수(vfw.h)는 AVI 파일을 열고 액세스하는 데 사용되는 파일 인터페이스의 주소를 반환합니다.
AVIFileReadData AVIFileReadData 함수는 작성자 또는 저작권 정보와 같은 전체 파일에 적용되는 선택적 헤더 데이터를 읽습니다.
AVIFileRelease AVIFileRelease 함수는 AVI 파일 인터페이스 핸들의 참조 수를 감소시키고 개수가 0에 도달하면 파일을 닫습니다.
AVIFileWriteData AVIFileWriteData 함수는 보조 데이터(일반 헤더, 형식 및 스트림 데이터 제외)를 파일에 씁니다.
AVIGetFromClipboard AVIGetFromClipboard 함수는 클립보드에서 AVI 파일을 복사합니다.
AVIMakeCompressedStream AVIMakeCompressedStream 함수는 압축되지 않은 스트림 및 압축 필터에서 압축 스트림을 만들고 압축된 스트림에 대한 포인터의 주소를 반환합니다. 이 함수는 오디오 및 비디오 압축을 지원합니다.
AVIMakeFileFromStreams AVIMakeFileFromStreams 함수는 별도의 스트림에서 AVIFile 인터페이스 포인터를 생성합니다.
AVIMakeStreamFromClipboard AVIMakeStreamFromClipboard 함수는 클립보드의 스트림 데이터에서 편집 가능한 스트림을 만듭니다.
AVIPutFileOnClipboard AVIPutFileOnClipboard 함수는 AVI 파일을 클립보드에 복사합니다.
AVISaveA AVISave 함수는 다른 파일 또는 메모리의 데이터 스트림을 결합하여 파일을 빌드합니다. (ANSI)
AVISaveOptions AVISaveOptions 함수는 파일에 대한 저장 옵션을 검색하여 버퍼에 반환합니다.
AVISaveOptionsFree

AVISaveOptionsFree 함수는 AVISaveOptions 함수에서 할당한 리소스를 해제합니다.
AVISaveVA AVISaveV 함수는 다른 파일 또는 메모리의 데이터 스트림을 결합하여 파일을 빌드합니다. (ANSI)
AVISaveVW AVISaveV 함수는 다른 파일 또는 메모리의 데이터 스트림을 결합하여 파일을 빌드합니다. (유니코드)
AVISaveW AVISave 함수는 다른 파일 또는 메모리의 데이터 스트림을 결합하여 파일을 빌드합니다. (유니코드)
AVIStreamAddRef AVIStreamAddRef 함수는 AVI 스트림의 참조 수를 증가합니다.
AVIStreamBeginStreaming AVIStreamBeginStreaming 함수는 스트리밍에 사용되는 매개 변수를 지정하고 스트림 처리기가 스트리밍을 준비할 수 있도록 합니다.
AVIStream만들기 AVIStreamCreate 함수는 파일과 연결되지 않은 스트림을 만듭니다.
AVIStreamDataSize AVIStreamDataSize 매크로는 지정된 스트림에 대한 선택적 헤더 데이터를 검색하는 데 필요한 버퍼 크기(바이트)를 결정합니다.
AVIStreamEnd AVIStreamEnd 매크로는 스트림의 끝과 연결된 샘플을 계산합니다.
AVIStreamEndStreaming AVIStreamEndStreaming 함수는 스트리밍을 종료합니다.
AVIStreamEndTime AVIStreamEndTime 매크로는 스트림의 끝을 나타내는 시간을 반환합니다.
AVIStreamFindSample

AVIStreamFindSample 함수는 지정된 위치를 기준으로 샘플의 위치(키 프레임, 없음 프레임 또는 형식 변경이 포함된 프레임)를 반환합니다.
AVIStreamFormatSize AVIStreamFormatSize 매크로는 샘플에 대한 형식 정보를 스트림에 저장하는 데 필요한 버퍼 크기(바이트)를 결정합니다.
AVIStreamGetFrame AVIStreamGetFrame 함수는 압축 해제된 비디오 프레임의 주소를 반환합니다.
AVIStreamGetFrameClose AVIStreamGetFrameClose 함수는 비디오 프레임의 압축을 해제하는 데 사용되는 리소스를 해제합니다.
AVIStreamGetFrameOpen AVIStreamGetFrameOpen 함수는 지정된 비디오 스트림에서 비디오 프레임의 압축을 풀도록 준비합니다.
AVIStreamInfoA AVIStreamInfo 함수는 스트림 헤더 정보를 가져옵니다. (ANSI)
AVIStreamInfoW AVIStreamInfo 함수는 스트림 헤더 정보를 가져옵니다. (유니코드)
AVIStreamIsKeyFrame AVIStreamIsKeyFrame 매크로는 지정된 스트림의 샘플이 키 프레임인지 여부를 나타냅니다.
AVIStreamLength AVIStreamLength 함수는 스트림의 길이를 반환합니다.
AVIStreamLengthTime AVIStreamLengthTime 매크로는 스트림의 길이를 시간 단위로 반환합니다.
AVIStreamNearestKeyFrame AVIStreamNearestKeyFrame 매크로는 스트림에서 지정된 위치 또는 그 이전의 키 프레임을 찾습니다.
AVIStreamNearestKeyFrameTime

AVIStreamNearestKeyFrameTime 매크로는 스트림에서 지정된 시간(또는 이전)에 가장 가까운 키 프레임의 시작에 해당하는 시간을 결정합니다.
AVIStreamNearestSample AVIStreamNearestSample 매크로는 스트림에서 지정된 위치 또는 그 이전의 가장 가까운 비어 있지 않은 샘플을 찾습니다.
AVIStreamNearestSampleTime AVIStreamNearestSampleTime 매크로는 스트림에서 지정된 시간에 가장 가까운 샘플의 시작에 해당하는 시간을 결정합니다.
AVIStreamNextKeyFrame AVIStreamNextKeyFrame 매크로는 스트림에서 지정된 위치에 따라 다음 키 프레임을 찾습니다.
AVIStreamNextKeyFrameTime AVIStreamNextKeyFrameTime 매크로는 지정된 시간부터 스트림의 다음 키 프레임 시간을 반환합니다.
AVIStreamNextSample AVIStreamNextSample 매크로는 스트림의 지정된 위치에서 다음 비어 있지 않은 샘플을 찾습니다.
AVIStreamNextSampleTime AVIStreamNextSampleTime 매크로는 샘플이 스트림의 다음 샘플로 변경되는 시간을 반환합니다. 이 매크로는 스트림에서 다음 흥미로운 시간을 찾습니다.
AVIStreamOpenFromFileA AVIStreamOpenFromFile 함수는 파일에서 단일 스트림을 엽니다. (ANSI)
AVIStreamOpenFromFileW AVIStreamOpenFromFile 함수는 파일에서 단일 스트림을 엽니다. (유니코드)
AVIStreamPrevKeyFrame AVIStreamPrevKeyFrame 매크로는 스트림에서 지정된 위치 앞에 있는 키 프레임을 찾습니다.
AVIStreamPrevKeyFrameTime AVIStreamPrevKeyFrameTime 매크로는 지정된 시간부터 스트림에서 이전 키 프레임의 시간을 반환합니다.
AVIStreamPrevSample

AVIStreamPrevSample 매크로는 스트림에서 지정된 위치 앞에 있는 가장 가까운 없음 샘플을 찾습니다.
AVIStreamPrevSampleTime AVIStreamPrevSampleTime 매크로는 스트림에서 지정된 시간보다 앞에 오는 가장 가까운 없음 샘플의 시간을 결정합니다.
AVIStreamRead AVIStreamRead 함수는 스트림 유형에 따라 스트림에서 오디오, 비디오 또는 기타 데이터를 읽습니다.
AVIStreamReadData AVIStreamReadData 함수는 스트림에서 선택적 헤더 데이터를 읽습니다.
AVIStreamReadFormat AVIStreamReadFormat 함수는 스트림 형식 데이터를 읽습니다.
AVIStreamRelease AVIStreamRelease 함수는 AVI 스트림 인터페이스 핸들의 참조 수를 감소시키고 개수가 0에 도달하면 스트림을 닫습니다.
AVIStreamSampleSize AVIStreamRelease 매크로는 스트림에서 하나의 정보 샘플을 저장하는 데 필요한 버퍼 크기를 결정합니다. 크기는 lPos로 지정된 위치에 있는 샘플에 해당합니다.
AVIStreamSampleToSample AVIStreamSampleToSample 매크로는 두 번째 스트림에서 발생하는 샘플과 동시에 발생하는 스트림의 샘플을 반환합니다.
AVIStreamSampleToTime AVIStreamSampleToTime 함수는 스트림 위치를 샘플에서 밀리초로 변환합니다.
AVIStreamSetFormat AVIStreamSetFormat 함수는 지정된 위치에서 스트림의 형식을 설정합니다.
AVIStreamStart AVIStreamStart 함수는 스트림의 시작 샘플 번호를 반환합니다.
AVIStreamStartTime

AVIStreamStartTime 매크로는 스트림의 첫 번째 샘플 시작 시간을 반환합니다.
AVIStreamTimeToSample AVIStreamTimeToSample 함수는 밀리초에서 샘플로 변환합니다.
AVIStreamWrite AVIStreamWrite 함수는 스트림에 데이터를 씁니다.
AVIStreamWriteData AVIStreamWriteData 함수는 선택적 헤더 정보를 스트림에 씁니다.
시작 Begin 메서드는 스트리밍 작업을 준비합니다. 애플리케이션이 AVIStreamBeginStreaming 함수를 사용할 때 호출됩니다.
시작 Begin 메서드는 스트림에서 프레임 복사본을 추출하고 압축을 풀도록 준비합니다. 애플리케이션이 AVIStreamGetFrameOpen 함수를 사용할 때 호출됩니다.
capCaptureAbort capCaptureAbort 매크로는 캡처 작업을 중지합니다. 이 매크로를 사용하거나 WM_CAP_ABORT 메시지를 명시적으로 보낼 수 있습니다.
capCaptureGetSetup capCaptureGetSetup 매크로는 스트리밍 캡처 매개 변수의 현재 설정을 검색합니다. 이 매크로를 사용하거나 WM_CAP_GET_SEQUENCE_SETUP 메시지를 명시적으로 보낼 수 있습니다.
capCaptureSequence capCaptureSequence 매크로는 파일에 대한 스트리밍 비디오 및 오디오 캡처를 시작합니다. 이 매크로를 사용하거나 WM_CAP_SEQUENCE 메시지를 명시적으로 보낼 수 있습니다.
capCaptureSequenceNoFile capCaptureSequenceNoFile 매크로는 파일에 데이터를 쓰지 않고 스트리밍 비디오 캡처를 시작합니다. 이 매크로를 사용하거나 WM_CAP_SEQUENCE_NOFILE 메시지를 명시적으로 보낼 수 있습니다.
capCaptureSetSetup capCaptureSetSetup 매크로는 스트리밍 캡처에 사용되는 구성 매개 변수를 설정합니다. 이 매크로를 사용하거나 WM_CAP_SET_SEQUENCE_SETUP 메시지를 명시적으로 보낼 수 있습니다.

capCaptureSingleFrame capCaptureSingleFrame 매크로는 capCaptureSingleFrameOpen 매크로를 사용하여 연 캡처 파일에 단일 프레임을 추가합니다. 이 매크로를 사용하거나 WM_CAP_SINGLE_FRAME 메시지를 명시적으로 보낼 수 있습니다.
capCaptureSingleFrameClose capCaptureSingleFrameClose 매크로는 capCaptureSingleFrameOpen 매크로에서 연 캡처 파일을 닫습니다. 이 매크로를 사용하거나 WM_CAP_SINGLE_FRAME_CLOSE 메시지를 명시적으로 보낼 수 있습니다.
capCaptureSingleFrameOpen capCaptureSingleFrameOpen 매크로는 단일 프레임 캡처를 위한 캡처 파일을 엽니다. 캡처 파일의 모든 이전 정보를 덮어씁니다. 이 매크로를 사용하거나 WM_CAP_SINGLE_FRAME_OPEN 메시지를 명시적으로 보낼 수 있습니다.
capCaptureStop capCaptureStop 매크로는 캡처 작업을 중지합니다. 이 매크로를 사용하거나 WM_CAP_STOP 메시지를 명시적으로 보낼 수 있습니다.
CAPCONTROLCALLBACK capControlCallback 함수는 스트리밍 캡처를 시작하고 종료하기 위해 정밀도 제어에 사용되는 콜백 함수입니다. capControlCallback 이름은 애플리케이션에서 제공하는 함수 이름의 자리 표시자입니다.
capCreateCaptureWindowA capCreateCaptureWindow 함수는 캡처 창을 만듭니다. (ANSI)
capCreateCaptureWindowW capCreateCaptureWindow 함수는 캡처 창을 만듭니다. (유니코드)
capDlgVideoCompression capDlgVideoCompression 매크로는 사용자가 캡처 프로세스 중에 사용할 압축기를 선택할 수 있는 대화 상자를 표시합니다.
capDlgVideoDisplay capDlgVideoDisplay 매크로는 사용자가 비디오 출력을 설정하거나 조정할 수 있는 대화 상자를 표시합니다.
capDlgVideoFormat

capDlgVideoFormat 매크로는 사용자가 비디오 형식을 선택할 수 있는 대화 상자를 표시합니다.
capDlgVideoSource capDlgVideoSource 매크로는 사용자가 비디오 원본을 제어할 수 있는 대화 상자를 표시합니다.
capDriverConnect capDriverConnect 매크로는 캡처 창을 캡처 드라이버에 연결합니다. 이 매크로를 사용하거나 WM_CAP_DRIVER_CONNECT 메시지를 명시적으로 보낼 수 있습니다.
capDriverDisconnect capDriverDisconnect 매크로는 캡처 창에서 캡처 드라이버의 연결을 끊습니다. 이 매크로를 사용하거나 WM_CAP_DRIVER_DISCONNECT 메시지를 명시적으로 보낼 수 있습니다.
capDriverGetCaps capDriverGetCaps 매크로는 캡처 창에 현재 연결된 캡처 드라이버의 하드웨어 기능을 반환합니다. 이 매크로를 사용하거나 WM_CAP_DRIVER_GET_CAPS 메시지를 명시적으로 보낼 수 있습니다.
capDriverGetName capDriverGetName 매크로는 캡처 창에 연결된 캡처 드라이버의 이름을 반환합니다. 이 매크로를 사용하거나 WM_CAP_DRIVER_GET_NAME 메시지를 명시적으로 호출할 수 있습니다.
capDriverGetVersion capDriverGetVersion 매크로는 캡처 창에 연결된 캡처 드라이버의 버전 정보를 반환합니다. 이 매크로를 사용하거나 WM_CAP_DRIVER_GET_VERSION 메시지를 명시적으로 보낼 수 있습니다.
capEditCopy capEditCopy 매크로는 비디오 프레임 버퍼 및 연결된 팔레트의 내용을 클립보드에 복사합니다. 이 매크로를 사용하거나 WM_CAP_EDIT_COPY 메시지를 명시적으로 보낼 수 있습니다.
CAPERRORCALLBACKA capErrorCallback 함수는 비디오 캡처에 사용되는 오류 콜백 함수입니다. capErrorCallback 이름은 애플리케이션에서 제공하는 함수 이름의 자리 표시자입니다. (ANSI)
CAPERRORCALLBACKW capErrorCallback 함수는 비디오 캡처에 사용되는 오류 콜백 함수입니다. capErrorCallback 이름은 애플리케이션에서 제공하는 함수 이름의 자리 표시자입니다. (유니코드)
capFileAlloc

capFileAlloc 매크로는 지정된 크기의 캡처 파일을 만듭니다(사전 할당). 이 매크로를 사용하거나 WM_CAP_FILE_ALLOCATE 메시지를 명시적으로 보낼 수 있습니다.
capFileGetCaptureFile capFileGetCaptureFile 매크로는 현재 캡처 파일의 이름을 반환합니다. 이 매크로를 사용하거나 WM_CAP_FILE_GET_CAPTURE_FILE 메시지를 명시적으로 호출할 수 있습니다.
capFileSaveAs capFileSaveAs 매크로는 캡처 파일의 내용을 다른 파일에 복사합니다. 이 매크로를 사용하거나 WM_CAP_FILE_SAVEAS 메시지를 명시적으로 호출할 수 있습니다.
capFileSaveDIB capFileSaveDIB 매크로는 현재 프레임을 DIB 파일에 복사합니다. 이 매크로를 사용하거나 WM_CAP_FILE_SAVEDIB 메시지를 명시적으로 호출할 수 있습니다.
capFileSetCaptureFile capFileSetCaptureFile 매크로는 비디오 캡처에 사용되는 파일의 이름을 지정합니다. 이 매크로를 사용하거나 WM_CAP_FILE_SET_CAPTURE_FILE 메시지를 명시적으로 호출할 수 있습니다.
capFileSetInfoChunk capFileSetInfoChunk 매크로는 정보 청크를 설정하고 지웁니다. 캡처하는 동안 정보 청크를 AVI 파일에 삽입하여 텍스트 문자열 또는 사용자 지정 데이터를 포함할 수 있습니다. 이 매크로를 사용하거나 WM_CAP_FILE_SET_INFOCHUNK 메시지를 명시적으로 호출할 수 있습니다.
capGetAudioFormat capGetAudioFormat 매크로는 오디오 형식을 가져옵니다. 이 매크로를 사용하거나 WM_CAP_GET_AUDIOFORMAT 메시지를 명시적으로 호출할 수 있습니다.
capGetAudioFormatSize capGetAudioFormatSize 매크로는 오디오 형식의 크기를 가져옵니다. 이 매크로를 사용하거나 WM_CAP_GET_AUDIOFORMAT 메시지를 명시적으로 호출할 수 있습니다.
capGetDriverDescriptionA capGetDriverDescription 함수는 캡처 드라이버의 버전 설명을 검색합니다. (ANSI)
capGetDriverDescriptionW capGetDriverDescription 함수는 캡처 드라이버의 버전 설명을 검색합니다. (유니코드)
capGetMCIDeviceName capGetMCIDeviceName 매크로는 capSetMCIDeviceName 매크로를 사용하여 이전에 설정한 MCI

디바이스의 이름을 검색합니다. 이 매크로를 사용하거나 WM_CAP_GET_MCI_DEVICE 메시지를 명시적으로 호출할 수 있습니다.
capGetStatus capGetStatus 매크로는 캡처 창의 상태 검색합니다. 이 매크로를 사용하거나 WM_CAP_GET_STATUS 메시지를 명시적으로 호출할 수 있습니다.
capGetUserData capGetUserData 매크로는 캡처 창과 연결된 LONG_PTR 데이터 값을 검색합니다. 이 매크로를 사용하거나 WM_CAP_GET_USER_DATA 메시지를 명시적으로 호출할 수 있습니다.
capGetVideoFormat capGetVideoFormat 매크로는 사용 중인 비디오 형식의 복사본을 검색합니다. 이 매크로를 사용하거나 WM_CAP_GET_VIDEOFORMAT 메시지를 명시적으로 호출할 수 있습니다.
capGetVideoFormatSize capGetVideoFormatSize 매크로는 비디오 형식에 필요한 크기를 검색합니다. 이 매크로를 사용하거나 WM_CAP_GET_VIDEOFORMAT 메시지를 명시적으로 호출할 수 있습니다.
capGrabFrame capGrabFrame 매크로는 캡처 드라이버에서 단일 프레임을 검색하고 표시합니다. 캡처 후 오버레이 및 미리 보기가 사용하지 않도록 설정됩니다. 이 매크로를 사용하거나 WM_CAP_GRAB_FRAME 메시지를 명시적으로 호출할 수 있습니다.
capGrabFrameNoStop capGrabFrameNoStop 매크로는 캡처 디바이스에서 압축되지 않은 단일 프레임으로 프레임 버퍼를 채우고 표시합니다.
capOverlay capOverlay 매크로는 오버레이 모드를 사용하거나 사용하지 않도록 설정합니다. 오버레이 모드에서는 하드웨어 오버레이를 사용하여 비디오가 표시됩니다. 이 매크로를 사용하거나 WM_CAP_SET_OVERLAY 메시지를 명시적으로 호출할 수 있습니다.
capPaletteAuto capPaletteAuto 매크로는 캡처 드라이버가 비디오 프레임을 샘플링하고 자동으로 새 팔레트를 만들 것을 요청합니다. 이 매크로를 사용하거나 WM_CAP_PAL_AUTOCREATE 메시지를 명시적으로 호출할 수 있습니다.
capPaletteManual capPaletteManual 매크로는 캡처 드라이버가 비디오 프레임을 수동으로 샘플링하고 새 팔레트를

만들 것을 요청합니다. 이 매크로를 사용하거나 WM_CAP_PAL_MANUALCREATE 메시지를 명시적으로 호출할 수 있습니다.
capPaletteOpen capPaletteOpen 매크로는 색상표 파일에서 새 팔레트를 로드하고 캡처 드라이버에 전달합니다.
capPalettePaste capPalettePaste 매크로는 클립보드에서 팔레트를 복사하여 캡처 드라이버에 전달합니다. 이 매크로를 사용하거나 WM_CAP_PAL_PASTE 메시지를 명시적으로 호출할 수 있습니다.
capPaletteSave capPaletteSave 매크로는 현재 색상표를 팔레트 파일에 저장합니다. 색상표 파일은 일반적으로 파일 이름 확장명 를 사용합니다. Pal. 이 매크로를 사용하거나 WM_CAP_PAL_SAVE 메시지를 명시적으로 보낼 수 있습니다.
capPreview capPreview 매크로는 미리 보기 모드를 사용하거나 사용하지 않도록 설정합니다.
capPreviewRate capPreviewRate 매크로는 프레임 표시 속도를 미리 보기 모드로 설정합니다. 이 매크로를 사용하거나 WM_CAP_SET_PREVIEWRATE 메시지를 명시적으로 호출할 수 있습니다.
capPreviewScale capPreviewScale 매크로는 미리 보기 비디오 이미지의 크기 조정을 사용하거나 사용하지 않도록 설정합니다.
capSetAudioFormat capSetAudioFormat 매크로는 스트리밍 또는 단계 캡처를 수행할 때 사용할 오디오 형식을 설정합니다. 이 매크로를 사용하거나 WM_CAP_SET_AUDIOFORMAT 메시지를 명시적으로 호출할 수 있습니다.
capSetCallbackOnCapControl capSetCallbackOnCapControl 매크로는 애플리케이션에서 콜백 함수를 설정하여 정확한 기록 제어를 제공합니다. 이 매크로를 사용하거나 WM_CAP_SET_CALLBACK_CAPCONTROL 메시지를 명시적으로 호출할 수 있습니다.
capSetCallbackOnError capSetCallbackOnError 매크로는 클라이언트 애플리케이션에서 오류 콜백 함수를 설정합니다. 오류가 발생하면 AVICap에서 이 절차를 호출합니다. 이 매크로를 사용하거나 WM_CAP_SET_CALLBACK_ERROR 메시지를 명시적으로 호출할 수 있습니다.

capSetCallbackOnFrame capSetCallbackOnFrame 매크로는 애플리케이션에서 미리 보기 콜백 함수를 설정합니다. AVICap은 캡처 창이 미리 보기 프레임을 캡처할 때 이 절차를 호출합니다. 이 매크로를 사용하거나 WM_CAP_SET_CALLBACK_FRAME 메시지를 명시적으로 호출할 수 있습니다.
capSetCallbackOnStatus capSetCallbackOnStatus 매크로는 애플리케이션에서 상태 콜백 함수를 설정합니다. AVICap은 캡처 창이 변경 상태 때마다 이 절차를 호출합니다. 이 매크로를 사용하거나 WM_CAP_SET_CALLBACK_STATUS 메시지를 명시적으로 호출할 수 있습니다.
capSetCallbackOnVideoStream capSetCallbackOnVideoStream 매크로는 애플리케이션에서 콜백 함수를 설정합니다. AVICap은 비디오 버퍼가 채워지면 스트리밍 캡처 중에 이 절차를 호출합니다. 이 매크로를 사용하거나 WM_CAP_SET_CALLBACK_VIDEOSTREAM 메시지를 명시적으로 호출할 수 있습니다.
capSetCallbackOnWaveStream capSetCallbackOnWaveStream 매크로는 애플리케이션에서 콜백 함수를 설정합니다.
capSetCallbackOnYield capSetCallbackOnYield 매크로는 애플리케이션에서 콜백 함수를 설정합니다. AVICap은 스트리밍 캡처 중에 캡처 창이 생성되면 이 절차를 호출합니다. 이 매크로를 사용하거나 WM_CAP_SET_CALLBACK_YIELD 메시지를 명시적으로 호출할 수 있습니다.
capSetMCIDeviceName capSetMCIDeviceName 매크로는 데이터를 캡처하는 데 사용할 MCI 비디오 디바이스의 이름을 지정합니다. 이 매크로를 사용하거나 WM_CAP_SET_MCI_DEVICE 메시지를 명시적으로 호출할 수 있습니다.
capSetScrollPos capSetScrollPos 매크로는 캡처 창에 표시할 비디오 프레임의 부분을 정의합니다.
capSetUserData capSetUserData 매크로는 LONG_PTR 데이터 값을 캡처 창과 연결합니다. 이 매크로를 사용하거나 WM_CAP_SET_USER_DATA 메시지를 명시적으로 호출할 수 있습니다.
capSetVideoFormat capSetVideoFormat 매크로는 캡처된 비디오 데이터의 형식을 설정합니다. 이 매크로를 사용하거나 WM_CAP_SET_VIDEOFORMAT 메시지를 명시적으로 호출할 수 있습니다.

CAPSTATUSCALLBACKA capStatusCallback 함수는 비디오 캡처에 사용되는 상태 콜백 함수입니다. capStatusCallback 이름은 애플리케이션에서 제공하는 함수 이름의 자리 표시자입니다. (ANSI)
CAPSTATUSCALLBACKW capStatusCallback 함수는 비디오 캡처에 사용되는 상태 콜백 함수입니다. capStatusCallback 이름은 애플리케이션에서 제공하는 함수 이름의 자리 표시자입니다. (유니코드)
CAPVIDEOCALLBACK capVideoStreamCallback 함수는 캡처된 비디오 프레임을 선택적으로 처리하기 위해 스트리밍 캡처와 함께 사용되는 콜백 함수입니다. capVideoStreamCallback 이름은 애플리케이션에서 제공하는 함수 이름의 자리 표시자입니다.
CAPWAVECALLBACK capWaveStreamCallback 함수는 필요에 따라 오디오 데이터의 버퍼를 처리하기 위해 스트리밍 캡처와 함께 사용되는 콜백 함수입니다. capWaveStreamCallback 이름은 애플리케이션에서 제공하는 함수 이름의 자리 표시자입니다.
CAPYIELDCALLBACK capYieldCallback 함수는 비디오 캡처에 사용되는 수익률 콜백 함수입니다. capYieldCallback 이름은 애플리케이션에서 제공하는 함수 이름의 자리 표시자입니다.
복제 Clone 메서드는 스트림을 복제합니다. 애플리케이션에서 EditStreamClone 함수를 사용할 때 호출됩니다.
CloseDriver 설치 가능한 드라이버를 닫습니다.
Copy Copy 메서드는 스트림 또는 그 일부를 임시 스트림에 복사합니다. 애플리케이션에서 EditStreamCopy 함수를 사용할 때 호출됩니다.
만들기 Create 메서드는 파일과 연결되지 않은 스트림 처리기를 초기화합니다. 애플리케이션에서 AVIStreamCreate 함수를 사용할 때 호출됩니다.
CreateEditableStream

CreateEditableStream 함수는 편집 가능한 스트림을 만듭니다. 다른 스트림 편집 함수를 사용하기 전에 이 함수를 사용합니다.
CreateStream CreateStream 메서드는 쓰기용 스트림을 만듭니다. 애플리케이션이 AVIFileCreateStream 함수를 사용할 때 호출됩니다.
잘라내기 Cut 메서드는 스트림의 일부를 제거하고 임시 스트림에 배치합니다. 애플리케이션에서 EditStreamCut 함수를 사용할 때 호출됩니다.
DefDriverProc 설치 가능한 드라이버에서 처리되지 않는 메시지에 대한 기본 처리를 제공합니다. 이 함수는 설치 가능한 드라이버의 DriverProc 함수 내에서만 사용됩니다. (DefDriverProc)
삭제 Delete 메서드는 스트림에서 데이터를 삭제합니다.
DrawDibBegin DrawDib 함수는 DrawDib DC의 매개 변수를 변경하거나 새 DrawDib DC를 초기화합니다.
DrawDibChangePalette DrawDibChangePalette 함수는 DIB 그리기에 사용되는 색상표 항목을 설정합니다.
DrawDibClose DrawDibClose 함수는 DrawDib DC를 닫고 DrawDib가 할당한 리소스를 해제합니다.
DrawDibDraw DrawDibDraw 함수는 화면에 DIB를 그립니다.
DrawDibEnd DrawDibEnd 함수는 DrawDibBegin 또는 DrawDibDraw 함수로 설정된 DrawDib DC의 플래그 및 기타 설정을 지웁니다.
DrawDibGetBuffer DrawDibGetBuffer 함수는 압축 해제를 위해 DrawDib에서 사용하는 버퍼의 위치를 검색합니다.
DrawDibGetPalette

DrawDibGetPalette 함수는 DrawDib DC에서 사용하는 팔레트를 검색합니다.
DrawDibOpen DrawDibOpen 함수는 사용할 DrawDib 라이브러리를 열고 그리기용 DrawDib DC를 만듭니다.
DrawDibProfileDisplay DrawDibProfileDisplay 함수는 DrawDib 함수를 사용할 때 디스플레이 시스템의 설정을 결정합니다.
DrawDibRealize DrawDibRealize 함수는 지정된 DC와 함께 사용할 DrawDib DC의 팔레트를 인식합니다.
DrawDibSetPalette DrawDibSetPalette 함수는 DIB 그리기에 사용되는 팔레트를 설정합니다.
DrawDibStart DrawDibStart 함수는 스트리밍 재생을 위해 DrawDib DC를 준비합니다.
DrawDibStop DrawDibStop 함수는 스트리밍 재생을 위해 DrawDib DC에서 사용하는 리소스를 해제합니다.
DrawDibTime DrawDibTime 함수는 그리기 작업에 대한 타이밍 정보를 검색하고 디버그 작업 중에 사용됩니다.
DrawDibUpdate DrawDibUpdate 매크로는 DrawDib 오프 스크린 버퍼의 마지막 프레임을 그립니다.
DriverCallback 콜백 함수를 호출하거나, 창에 메시지를 보내거나, 스레드 차단을 해제합니다. 작업은 알림 플래그의 값에 따라 달라집니다. 이 함수는 설치 가능한 드라이버의 DriverProc 함수 내에서만 사용됩니다.
DRIVERPROC 설치 가능한 드라이버에 대한 드라이버 메시지를 처리합니다. DriverProc은 드라이버 제공 함수입니다.
DrvDefDriverProc

설치 가능한 드라이버에서 처리되지 않는 메시지에 대한 기본 처리를 제공합니다. 이 함수는 설치 가능한 드라이버의 DriverProc 함수 내에서만 사용됩니다. D
DrvGetModuleHandle 설치 가능한 드라이버가 포함된 모듈의 instance 핸들을 검색합니다. 이 함수는 이전 버전의 Windows와의 호환성을 위해 제공됩니다.
EditStreamClone EditStreamClone 함수는 중복 편집 가능한 스트림을 만듭니다.
EditStreamCopy EditStreamCopy 함수는 편집 가능한 스트림(또는 그 일부)을 임시 스트림에 복사합니다.
EditStreamCut EditStreamCut 함수는 편집 가능한 스트림의 전체 또는 일부를 삭제하고 스트림의 삭제된 부분에서 임시 편집 가능한 스트림을 만듭니다.
EditStreamPaste EditStreamPaste 함수는 한 스트림에서 스트림(또는 그 일부)을 복사하여 지정된 위치에 있는 다른 스트림 내에 붙여넣습니다.
EditStreamSetInfoA EditStreamSetInfo 함수는 편집 가능한 스트림의 특성을 변경합니다. (ANSI)
EditStreamSetInfoW EditStreamSetInfo 함수는 편집 가능한 스트림의 특성을 변경합니다. (유니코드)
EditStreamSetNameA EditStreamSetName 함수는 설명 문자열을 스트림에 할당합니다. (ANSI)
EditStreamSetNameW EditStreamSetName 함수는 설명 문자열을 스트림에 할당합니다. (유니코드)
End End 메서드는 스트리밍 작업을 종료합니다. 애플리케이션이 AVIStreamEndStreaming 함수를 사용할 때 호출됩니다.
End

End 메서드는 프레임 추출 및 압축 해제를 종료합니다. 애플리케이션이 AVIStreamGetFrameClose 함수를 사용할 때 호출됩니다.
EndRecord EndRecord 메서드는 긴밀하게 인터리브된 AVI 파일에 "REC" 청크를 씁니다(비디오로 오디오의 일대일 인터리브 인수 포함). 애플리케이션이 AVIFileEndRecord 함수를 사용할 때 호출됩니다.
FindSample FindSample 메서드는 키 프레임 또는 비어 있지 않은 프레임의 스트림에서 위치를 가져옵니다. 애플리케이션이 AVIStreamFindSample 함수를 사용할 때 호출됩니다.
GetDriverModuleHandle 설치 가능한 드라이버가 포함된 모듈의 instance 핸들을 검색합니다.
GetFrame GetFrame 메서드는 스트림에서 압축 해제된 프레임 복사본을 검색합니다. 애플리케이션이 AVIStreamGetFrame 함수를 사용할 때 호출됩니다.
GetOpenFileNamePreviewA GetOpenFileNamePreview 함수는 열기 대화 상자를 사용하여 파일을 선택합니다. 대화 상자를 사용하면 사용자가 현재 지정된 AVI 파일을 미리 볼 수도 있습니다. 이 함수는 GetOpenFileName 함수에 있는 기능을 보강합니다. (ANSI)
GetOpenFileNamePreviewW GetOpenFileNamePreview 함수는 열기 대화 상자를 사용하여 파일을 선택합니다. 대화 상자를 사용하면 사용자가 현재 지정된 AVI 파일을 미리 볼 수도 있습니다. 이 함수는 GetOpenFileName 함수에 있는 기능을 보강합니다. (유니코드)
GetSaveFileNamePreviewA GetSaveFileNamePreview 함수는 다른 이름으로 저장 대화 상자를 사용하여 파일을 선택합니다. 대화 상자를 사용하면 사용자가 현재 지정된 파일을 미리 볼 수도 있습니다. 이 함수는 GetSaveFileName 함수에 있는 기능을 보강합니다. (ANSI)
GetSaveFileNamePreviewW GetSaveFileNamePreview 함수는 다른 이름으로 저장 대화 상자를 사용하여 파일을 선택합니다. 대화 상자를 사용하면 사용자가 현재 지정된 파일을 미리 볼 수도 있습니다. 이 함수는 GetSaveFileName 함수에 있는 기능을 보강합니다. (유니코드)
GetStream

GetStream 메서드는 파일에서 스트림에 액세스하여 스트림을 엽니다. 애플리케이션이 AVIFileGetStream 함수를 사용할 때 호출됩니다.
ICAbout ICAbout 매크로는 비디오 압축 드라이버에 정보 대화 상자를 표시하도록 알 수 있습니다. 이 매크로를 사용하거나 ICM_ABOUT 메시지를 명시적으로 호출할 수 있습니다.
ICClose ICClose 함수는 압축기 또는 압축 해제기를 닫습니다.
ICCompress ICCompress 함수는 단일 비디오 이미지를 압축합니다.
ICCompressBegin ICCompressBegin 매크로는 비디오 압축 드라이버에 데이터 압축을 준비하도록 알 수 있습니다. 이 매크로를 사용하거나 ICM_COMPRESS_BEGIN 메시지를 명시적으로 호출할 수 있습니다.
ICCompressEnd ICCompressEnd 매크로는 압축을 위해 할당된 압축 및 사용 가능한 리소스를 종료하도록 비디오 압축 드라이버에 알 보냅니다. 이 매크로를 사용하거나 ICM_COMPRESS_END 메시지를 명시적으로 호출할 수 있습니다.
ICCompressGetFormat ICCompressGetFormat 매크로는 비디오 압축 드라이버에서 압축된 데이터의 출력 형식을 요청합니다. 이 매크로를 사용하거나 ICM_COMPRESS_GET_FORMAT 메시지를 명시적으로 호출할 수 있습니다.
ICCompressGetFormatSize ICCompressGetFormatSize 매크로는 비디오 압축 드라이버에서 압축된 데이터의 출력 형식 크기를 요청합니다. 이 매크로를 사용하거나 ICM_COMPRESS_GET_FORMAT 메시지를 명시적으로 호출할 수 있습니다.
ICCompressGetSize ICCompressGetSize 매크로는 비디오 압축 드라이버가 지정된 출력 형식으로 압축될 때 데이터 프레임의 최대 크기를 제공하도록 요청합니다. 이 매크로를 사용하거나 ICM_COMPRESS_GET_SIZE 메시지를 명시적으로 호출할 수 있습니다.
ICCompressorChoose ICCompressorChoose 함수는 사용자가 압축기를 선택할 수 있는 대화 상자를 표시합니다. 이 함수는 등록된 모든 압축기를 표시하거나 특정 형식을 지원하는 압축기만 나열할 수 있습니다.

ICCompressorFree ICCompressorFree 함수는 다른 VCM 함수에서 사용하는 COMPVARS 구조의 리소스를 해제합니다.
ICCompressQuery ICCompressQuery 매크로는 비디오 압축 드라이버를 쿼리하여 특정 입력 형식을 지원하는지 또는 특정 입력 형식을 특정 출력 형식으로 압축할 수 있는지 확인합니다.
ICConfigure ICConfigure 매크로는 비디오 압축 드라이버에 구성 대화 상자를 표시하도록 알 수 있습니다. 이 매크로를 사용하거나 ICM_CONFIGURE 메시지를 명시적으로 보낼 수 있습니다.
ICDecompress ICDecompress 함수는 단일 비디오 프레임의 압축을 풉니다.
ICDecompressBegin ICDecompressBegin 매크로는 비디오 압축 해제 드라이버에 데이터 압축 해제를 준비하도록 알 수 있습니다. 이 매크로를 사용하거나 ICM_DECOMPRESS_BEGIN 메시지를 명시적으로 호출할 수 있습니다.
ICDecompressEnd ICDecompressEnd 매크로는 압축 해제를 위해 할당된 압축 해제 및 사용 가능한 리소스를 종료하도록 비디오 압축 해제 드라이버를 알 수 있습니다. 이 매크로를 사용하거나 ICM_DECOMPRESS_END 메시지를 명시적으로 호출할 수 있습니다.
ICDecompressEx ICDecompressEx 함수는 단일 비디오 프레임의 압축을 풉니다.
ICDecompressExBegin ICDecompressExBegin 함수는 데이터 압축을 풀기 위한 압축 해제기를 준비합니다.
ICDecompressExEnd ICDecompressExEnd 매크로는 압축 해제를 위해 할당된 압축 해제 및 해제 리소스를 종료하도록 비디오 압축 해제 드라이버를 알 수 있습니다. 이 매크로를 사용하거나 ICM_DECOMPRESSEX_END 메시지를 명시적으로 호출할 수 있습니다.
ICDecompressExQuery ICDecompressExQuery 함수는 압축 해제기가 특정 형식으로 데이터를 압축 해제할 수 있는지 여부를 결정합니다.

ICDecompressGetFormat ICDecompressGetFormat 매크로는 비디오 압축 해제 드라이버에서 압축 해제된 데이터의 출력 형식을 요청합니다. 이 매크로를 사용하거나 ICM_DECOMPRESS_GET_FORMAT 메시지를 명시적으로 호출할 수 있습니다.
ICDecompressGetFormatSize ICDecompressGetFormatSize 매크로는 비디오 압축 해제 드라이버에서 압축 해제된 데이터의 출력 형식 크기를 요청합니다. 이 매크로를 사용하거나 ICM_DECOMPRESS_GET_FORMAT 메시지를 명시적으로 호출할 수 있습니다.
ICDecompressGetPalette ICDecompressGetPalette 매크로는 비디오 압축 해제 드라이버가 출력 BITMAPINFOHEADER 구조의 색 테이블을 제공해 줄 것을 요청합니다. 이 매크로를 사용하거나 ICM_DECOMPRESS_GET_PALETTE 메시지를 명시적으로 호출할 수 있습니다.
ICDecompressOpen ICDecompressOpen 매크로는 지정된 형식과 호환되는 압축 해제기를 엽니다.
ICDecompressQuery ICDecompressQuery 매크로는 비디오 압축 해제 드라이버를 쿼리하여 특정 입력 형식을 지원하는지 또는 특정 입력 형식을 특정 출력 형식으로 압축 해제할 수 있는지 확인합니다.
ICDecompressSetPalette ICDecompressSetPalette 매크로는 색상표를 사용하는 형식으로 압축을 풀 때 사용할 비디오 압축 해제 드라이버의 팔레트를 지정합니다. 이 매크로를 사용하거나 ICM_DECOMPRESS_SET_PALETTE 메시지를 명시적으로 호출할 수 있습니다.
ICDraw ICDraw 함수는 그리기 위해 이미지를 압축 해제합니다.
ICDrawBegin ICDrawBegin 함수는 렌더러를 초기화하고 그리기 대상을 준비합니다.
ICDrawChangePalette ICDrawChangePalette 매크로는 렌더링 드라이버에 영화 팔레트가 변경되고 있음을 알립니다. 이 매크로를 사용하거나 ICM_DRAW_CHANGEPALETTE 메시지를 명시적으로 호출할 수 있습니다.
ICDrawEnd ICDrawEnd 매크로는 렌더링 드라이버에 현재 이미지를 화면으로 압축 해제하고 압축 해제 및 그리

기에 할당된 리소스를 해제하도록 알립니다. 이 매크로를 사용하거나 ICM_DRAW_END 메시지를 명시적으로 호출할 수 있습니다.
ICDrawFlush ICDrawFlush 매크로는 렌더링 드라이버에 그리기를 기다리는 모든 이미지 버퍼의 콘텐츠를 렌더링하도록 알립니다. 이 매크로를 사용하거나 ICM_DRAW_FLUSH 메시지를 명시적으로 호출할 수 있습니다.
ICDrawGetTime ICDrawGetTime 매크로는 그리기 프레임의 타이밍을 제어하는 렌더링 드라이버를 요청하여 내부 클록의 현재 값을 반환합니다. 이 매크로를 사용하거나 ICM_DRAW_GETTIME 메시지를 명시적으로 호출할 수 있습니다.
ICDrawOpen ICDrawOpen 매크로는 지정된 형식으로 이미지를 그릴 수 있는 드라이버를 엽니다.
ICDrawQuery ICDrawQuery 매크로는 렌더링 드라이버를 쿼리하여 특정 형식으로 데이터를 렌더링할 수 있는지 확인합니다. 이 매크로를 사용하거나 ICM_DRAW_QUERY 메시지를 명시적으로 호출할 수 있습니다.
ICDrawRealize ICDrawRealize 매크로는 렌더링 드라이버에 그리는 동안 그리기 팔레트를 실현하도록 알립니다. 이 매크로를 사용하거나 ICM_DRAW_REALIZE 메시지를 명시적으로 호출할 수 있습니다.
ICDrawRenderBuffer ICDrawRenderBuffer 매크로는 렌더링 드라이버에 전달된 프레임을 그리도록 알립니다. 이 매크로를 사용하거나 ICM_DRAW_RENDERBUFFER 메시지를 명시적으로 호출할 수 있습니다.
ICDrawSetTime ICDrawSetTime 매크로는 그리기 프레임의 타이밍을 처리하는 렌더링 드라이버에 동기화 정보를 제공합니다.
ICDrawStart ICDrawStart 매크로는 렌더링 드라이버에 그리기 프레임 타이밍에 대한 내부 클록을 시작하도록 알립니다. 이 매크로를 사용하거나 ICM_DRAW_START 메시지를 명시적으로 호출할 수 있습니다.
ICDrawStartPlay ICDrawStartPlay 매크로는 렌더링 드라이버에 재생 작업의 시작 및 종료 시간을 제공합니다. 이 매크로를 사용하거나 ICM_DRAW_START_PLAY 메시지를 명시적으로 호출할 수 있습니다.

ICDrawStop ICDrawStop 매크로는 렌더링 드라이버에 그리기 프레임 타이밍에 대한 내부 클록을 중지하도록 알립니다. 이 매크로를 사용하거나 ICM_DRAW_STOP 메시지를 명시적으로 호출할 수 있습니다.
ICDrawStopPlay ICDrawStopPlay 매크로는 재생 작업이 완료되면 렌더링 드라이버에 알 수 있습니다. 이 매크로를 사용하거나 ICM_DRAW_STOP_PLAY 메시지를 명시적으로 호출할 수 있습니다.
ICDrawSuggestFormat ICDrawSuggestFormat 함수는 그리기 처리기에 입력 데이터 형식을 제안하도록 알립니다.
ICDrawWindow ICDrawWindow 매크로는 렌더링 드라이버에 ICM_DRAW_BEGIN 메시지에 대해 지정된 창을 다시 그려야 한다는 것을 알립니다. 창이 이동되었거나 일시적으로 가려집니다. 이 매크로를 사용하거나 ICM_DRAW_WINDOW 메시지를 명시적으로 호출할 수 있습니다.
ICGetBuffersWanted ICGetBuffersWanted 매크로는 드라이버에 할당할 버퍼 수를 쿼리합니다. 이 매크로를 사용하거나 ICM_GETBUFFERSWANTED 메시지를 명시적으로 호출할 수 있습니다.
ICGetDefaultKeyFrameRate ICGetDefaultKeyFrameRate 매크로는 비디오 압축 드라이버에서 기본 키 프레임 간격(또는 기본 설정)을 쿼리합니다. 이 매크로를 사용하거나 ICM_GETDEFAULTKEYFRAMERATE 메시지를 명시적으로 호출할 수 있습니다.
ICGetDefaultQuality ICGetDefaultQuality 매크로는 비디오 압축 드라이버를 쿼리하여 기본 품질 설정을 제공합니다. 이 매크로를 사용하거나 ICM_GETDEFAULTQUALITY 메시지를 명시적으로 호출할 수 있습니다.
ICGetDisplayFormat ICGetDisplayFormat 함수는 압축된 이미지를 표시하는 데 사용할 수 있는 최상의 형식을 결정합니다. 또한 열린 압축기의 핸들을 지정하지 않은 경우 함수는 압축기를 엽니다.
ICGetInfo ICGetInfo 함수는 압축기 정보를 가져옵니다.
ICGetState ICGetState 매크로는 비디오 압축 드라이버를 쿼리하여 메모리 블록에서 현재 구성을 반환합니다. 이 매크로를 사용하거나 ICM_GETSTATE 메시지를 명시적으로 호출할 수 있습니다.

ICGetStateSize ICGetStateSize 매크로는 비디오 압축 드라이버를 쿼리하여 구성 정보를 검색하는 데 필요한 메모리 양을 확인합니다. 이 매크로를 사용하거나 ICM_GETSTATE 메시지를 명시적으로 호출할 수 있습니다.
ICImageCompress ICImageCompress 함수는 이미지를 지정된 크기로 압축합니다. 이 함수에는 초기화 함수가 필요하지 않습니다.
ICImageDecompress ICImageDecompress 함수는 초기화 함수를 사용하지 않고 이미지를 압축 해제합니다.
ICInfo ICInfo 함수는 설치된 특정 압축기에 대한 정보를 검색하거나 설치된 압축기를 열거합니다.
ICInstall ICInstall 함수는 새 압축기 또는 압축 해제기를 설치합니다.
ICLocate ICLocate 함수는 지정된 형식으로 이미지를 처리할 수 있는 압축기 또는 압축 해제기를 찾거나 지정된 형식의 이미지를 하드웨어로 직접 압축 해제할 수 있는 드라이버를 찾습니다.
ICOpen ICOpen 함수는 압축기 또는 압축 해제기를 엽니다.
ICOpenFunction ICOpenFunction 함수는 함수로 정의된 압축기 또는 압축 해제기를 엽니다.
ICQueryAbout ICQueryAbout 매크로는 비디오 압축 드라이버를 쿼리하여 정보 대화 상자가 있는지 확인합니다. 이 매크로를 사용하거나 ICM_ABOUT 메시지를 명시적으로 호출할 수 있습니다.
ICQueryConfigure ICQueryConfigure 매크로는 비디오 압축 드라이버를 쿼리하여 구성 대화 상자가 있는지 확인합니다. 이 매크로를 사용하거나 ICM_CONFIGURE 메시지를 명시적으로 보낼 수 있습니다.
ICRemove ICRemove 함수는 설치된 압축기를 제거합니다.

ICSendMessage ICSendMessage 함수는 압축기로 메시지를 보냅니다.
ICSeqCompressFrame ICSeqCompressFrame 함수는 프레임 시퀀스에서 한 프레임을 압축합니다.
ICSeqCompressFrameEnd ICSeqCompressFrameEnd 함수는 ICSeqCompressFrameStart 및 ICSeqCompressFrame 함수를 사용하여 시작된 시퀀스 압축을 종료합니다.
ICSeqCompressFrameStart ICSeqCompressFrameStart 함수는 ICSeqCompressFrame 함수를 사용하여 프레임 시퀀스를 압축하기 위한 리소스를 초기화합니다.
ICSetState ICSetState 매크로는 비디오 압축 드라이버에 압축기의 상태를 설정하도록 알 수 있습니다. 이 매크로를 사용하거나 ICM_SETSTATE 메시지를 명시적으로 호출할 수 있습니다.
ICSetStatusProc ICSetStatusProc 함수는 상태 콜백 함수의 주소를 압축기로 보냅니다. 압축기는 긴 작업 중에 이 함수를 호출합니다.
정보 Info 메서드는 AVI 파일에 대한 정보와 함께 를 반환합니다. 애플리케이션이 AVIFileInfo 함수를 사용할 때 호출됩니다.
정보 Info 메서드는 스트림에 대한 정보를 사용하여 AVISTREAMINFO 구조를 채우고 반환합니다. 애플리케이션이 AVIStreamInfo 함수를 사용할 때 호출됩니다.
joyConfigChanged joyConfigChanged 함수는 조이스틱 드라이버에 구성이 변경되었으며 레지스트리에서 다시 로드되어야 한다고 알릴 수 있습니다.
joyGetDevCaps joyGetDevCaps 함수(joystickapi.h)는 조이스틱을 쿼리하여 기능을 확인합니다. uJoyID는 쿼리할 조이스틱의 식별자입니다.

joyGetDevCapsA joyGetDevCaps 함수는 조이스틱을 쿼리하여 기능을 확인합니다. (joyGetDevCapsA)
joyGetDevCapsW joyGetDevCapsW(유니코드) 함수는 조이스틱을 쿼리하여 포괄적인 기능을 확인합니다.
joyGetNumDevs joyGetNumDevs 함수는 조이스틱 드라이버에 지원하는 조이스틱 수를 쿼리합니다.
joyGetPos joyGetPos 함수는 조이스틱의 위치와 단추 상태 쿼리합니다.
joyGetPosEx joyGetPosEx 함수는 조이스틱의 위치와 단추 상태 쿼리합니다.
joyGetThreshold joyGetThreshold 함수는 조이스틱에서 현재 이동 임계값을 쿼리합니다.
joyReleaseCapture joyReleaseCapture 함수는 지정된 캡처된 조이스틱을 해제합니다.
joySetCapture joySetCapture 함수는 지정된 창으로 메시지를 보내 조이스틱을 캡처합니다.
joySetThreshold joySetThreshold 함수는 조이스틱의 이동 임계값을 설정합니다.
MCIWndCanConfig MCIWndCanConfig 매크로는 MCI 디바이스가 구성 대화 상자를 표시할 수 있는지 여부를 결정합니다. 이 매크로를 사용하거나 MCIWNDM_CAN_CONFIG 메시지를 명시적으로 보낼 수 있습니다.
MCIWndCanEject MCIWndCanEject 매크로는 MCI 디바이스가 미디어를 배출할 수 있는지 여부를 결정합니다. 이 매크로를 사용하거나 MCIWNDM_CAN_EJECT 메시지를 명시적으로 보낼 수 있습니다.
MCIWndCanPlay MCIWndCanPlay 매크로는 MCI 디바이스가 다른 종류의 데이터 파일 또는 콘텐츠를 재생할 수 있

는지 여부를 결정합니다. 이 매크로를 사용하거나 MCIWNDM_CAN_PLAY 메시지를 명시적으로 보낼 수 있습니다.
MCIWndCanRecord MCIWndCanRecord 매크로는 MCI 디바이스가 녹음/녹화를 지원하는지 여부를 결정합니다. 이 매크로를 사용하거나 MCIWNDM_CAN_RECORD 메시지를 명시적으로 보낼 수 있습니다.
MCIWndCanSave MCIWndCanSave 매크로는 MCI 디바이스가 데이터를 저장할 수 있는지 여부를 결정합니다. 이 매크로를 사용하거나 MCIWNDM_CAN_SAVE 메시지를 명시적으로 보낼 수 있습니다.
MCIWndCanWindow MCIWndCanWindow 매크로는 MCI 디바이스가 창 지향 MCI 명령을 지원하는지 여부를 결정합니다. 이 매크로를 사용하거나 MCIWNDM_CAN_WINDOW 메시지를 명시적으로 보낼 수 있습니다.
MCIWndChangeStyles MCIWndChangeStyles 매크로는 MCIWnd 창에서 사용하는 스타일을 변경합니다. 이 매크로를 사용하거나 MCIWNDM_CHANGESTYLES 메시지를 명시적으로 보낼 수 있습니다.
MCIWndClose MCIWndClose 매크로는 MCIWnd 창과 연결된 MCI 디바이스 또는 파일을 닫습니다.
MCIWndCreateA MCIWndCreate 함수는 MCIWnd 창 클래스를 등록하고 MCI 서비스를 사용하기 위한 MCIWnd 창을 만듭니다. MCIWndCreate는 MCI 디바이스 또는 파일(예: AVI 파일)을 열고 MCIWnd 창과 연결할 수도 있습니다. (ANSI)
MCIWndCreateW MCIWndCreate 함수는 MCIWnd 창 클래스를 등록하고 MCI 서비스를 사용하기 위한 MCIWnd 창을 만듭니다. MCIWndCreate는 MCI 디바이스 또는 파일(예: AVI 파일)을 열고 MCIWnd 창과 연결할 수도 있습니다. (유니코드)
MCIWndDestroy MCIWndDestroy 매크로는 MCIWnd 창과 연결된 MCI 디바이스 또는 파일을 닫고 창을 삭제합니다. 이 매크로를 사용하거나 WM_CLOSE 메시지를 명시적으로 보낼 수 있습니다.
MCIWndEject MCIWndEject 매크로는 MCI 디바이스에 해당 미디어를 꺼내는 명령을 보냅니다. 이 매크로를 사용하거나 MCIWNDM_EJECT 메시지를 명시적으로 보낼 수 있습니다.

MCIWndEnd MCIWndEnd 매크로는 현재 위치를 콘텐츠의 끝으로 이동합니다. 이 매크로를 사용하거나 MCI_SEEK 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetActiveTimer MCIWndGetActiveTimer 매크로는 MCIWnd 창이 활성 창일 때 사용되는 업데이트 기간을 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETACTIVETIMER 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetAlias MCIWndGetAlias 매크로는 mciSendString 함수를 사용하여 MCI 디바이스 또는 파일을 여는 데 사용되는 별칭을 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETALIAS 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetDest MCIWndGetDest 매크로는 재생 중에 AVI 파일의 이미지를 확대하거나 확대하는 데 사용되는 대상 사각형의 좌표를 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GET_DEST 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetDevice MCIWndGetDevice 매크로는 현재 MCI 디바이스의 이름을 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETDEVICE 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetDeviceID MCIWndGetDeviceID 매크로는 mciSendCommand 함수와 함께 사용할 현재 MCI 디바이스의 식별자를 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETDEVICEID 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetEnd MCIWndGetEnd 매크로는 MCI 디바이스 또는 파일의 콘텐츠 끝 위치를 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETEND 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetError MCIWndGetError 매크로는 마지막으로 발생한 MCI 오류를 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETERROR 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetFileName MCIWndGetFileName 매크로는 MCI 디바이스에서 사용하는 파일 이름을 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETFILENAME 메시지를 명시적으로 보낼 수 있습니다.

MCIWndGetInactiveTimer MCIWndGetInactiveTimer 매크로는 MCIWnd 창이 비활성 창일 때 사용되는 업데이트 기간을 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETINACTIVETIMER 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetLength MCIWndGetLength 매크로는 MCI 디바이스에서 현재 사용되는 콘텐츠 또는 파일의 길이를 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETLENGTH 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetMode MCIWndGetMode 매크로는 MCI 디바이스의 현재 작동 모드를 검색합니다. MCI 디바이스에는 상수로 지정된 여러 작동 모드가 있습니다. 이 매크로를 사용하거나 MCIWNDM_GETMODE 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetPalette MCIWndGetPalette 매크로는 MCI 디바이스에서 사용하는 팔레트의 핸들을 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETPALETTE 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetPosition MCIWndGetPosition 매크로는 MCI 디바이스의 콘텐츠 내에서 현재 위치의 숫자 값을 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETPOSITION 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetPositionString MCIWndGetPositionString 매크로는 MCI 디바이스의 콘텐츠 내에서 현재 위치의 숫자 값을 검색합니다.
MCIWndGetRepeat MCIWndGetRepeat 매크로는 연속 재생이 활성화되었는지 여부를 결정합니다. 이 매크로를 사용하거나 MCIWNDM_GETREPEAT 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetSource MCIWndGetSource 매크로는 재생 중에 AVI 파일의 이미지를 자르는 데 사용되는 원본 사각형의 좌표를 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GET_SOURCE 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetSpeed MCIWndGetSpeed 매크로는 MCI 디바이스의 재생 속도를 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETSPEED 메시지를 명시적으로 보낼 수 있습니다.

MCIWndGetStart MCIWndGetStart 매크로는 MCI 디바이스 또는 파일의 콘텐츠 시작 위치를 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETSTART 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetStyles MCIWndGetStyles 매크로는 창에서 사용하는 현재 MCIWnd 창 스타일을 지정하는 플래그를 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETSTYLES 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetTimeFormat MCIWndGetTimeFormat 매크로는 MCI 디바이스의 현재 시간 형식을 숫자 값과 문자열로 _as 두 가지 형식으로 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETTIMEFORMAT 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetVolume MCIWndGetVolume 매크로는 MCI 디바이스의 현재 볼륨 설정을 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETVOLUME 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetZoom MCIWndGetZoom 매크로는 MCI 디바이스에서 사용하는 현재 확대/축소 값을 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETZOOM 메시지를 명시적으로 보낼 수 있습니다.
MCIWndHome MCIWndHome 매크로는 현재 위치를 콘텐츠의 시작 부분으로 이동합니다. 이 매크로를 사용하거나 MCI_SEEK 명령을 명시적으로 보낼 수 있습니다.
MCIWndNew MCIWndNew 매크로는 현재 MCI 디바이스에 대한 새 파일을 만듭니다. 이 매크로를 사용하거나 MCIWNDM_NEW 메시지를 명시적으로 보낼 수 있습니다.
MCIWndOpen MCIWndOpen 매크로는 MCI 디바이스를 열고 MCIWnd 창에 연결합니다.
MCIWndOpenDialog MCIWndOpenDialog 매크로는 사용자가 지정한 데이터 파일과 해당 유형의 MCI 디바이스를 열고 MCIWnd 창에 연결합니다.
MCIWndOpenInterface MCIWndOpenInterface 매크로는 지정된 인터페이스와 연결된 데이터 스트림 또는 파일을

MCIWnd 창에 연결합니다. 이 매크로를 사용하거나 MCIWNDM_OPENINTERFACE 메시지를 명시적으로 보낼 수 있습니다.
MCIWndPause MCIWndPause 매크로는 MCI 디바이스에 명령을 보내 재생 또는 녹화를 일시 중지합니다.
MCIWndPlay MCIWndPlay 매크로는 MCI 디바이스에 명령을 보내 콘텐츠의 현재 위치에서 재생을 시작합니다. 이 매크로를 사용하거나 MCI_PLAY 명령을 명시적으로 보낼 수 있습니다.
MCIWndPlayFrom MCIWndPlayFrom 매크로는 지정된 위치에서 콘텐츠 끝까지 또는 다른 명령이 재생을 중지할 때까지 MCI 디바이스의 콘텐츠를 재생합니다. 이 매크로를 사용하거나 MCIWNDM_PLAYFROM 메시지를 명시적으로 보낼 수 있습니다.
MCIWndPlayFromTo MCIWndPlayFromTo 매크로는 지정된 시작 위치와 종료 위치 간에 콘텐츠의 일부를 재생합니다.
MCIWndPlayReverse MCIWndPlayReverse 매크로는 현재 위치에서 시작하여 콘텐츠의 시작 부분에서 또는 다른 명령이 재생을 중지할 때까지 현재 콘텐츠를 역방향으로 재생합니다.
MCIWndPlayTo MCIWndPlayTo 매크로는 MCI 디바이스의 콘텐츠를 현재 위치에서 지정된 끝 위치로 재생하거나 다른 명령이 재생을 중지할 때까지 재생합니다.
MCIWndPutDest MCIWndPutDest 매크로는 재생 중에 AVI 파일의 이미지를 확대/축소하거나 확대하는 데 사용되는 대상 사각형의 좌표를 다시 정의합니다. 이 매크로를 사용하거나 MCIWNDM_PUT_DEST 메시지를 명시적으로 보낼 수 있습니다.
MCIWndPutSource MCIWndPutSource 매크로는 재생 중에 AVI 파일의 이미지를 자르는 데 사용되는 원본 사각형의 좌표를 다시 정의합니다. 이 매크로를 사용하거나 MCIWNDM_PUT_SOURCE 메시지를 명시적으로 보낼 수 있습니다.
MCIWndRealize MCIWndRealize 매크로는 전경 또는 백그라운드에서 MCI 창이 실현되는 방식을 제어합니다. 또한 이 매크로는 프로세스에서 MCI 창의 팔레트를 실현합니다. 이 매크로를 사용하거나 MCIWNDM_REALIZE 메시지를 명시적으로 보낼 수 있습니다.

MCIWndRecord MCIWndRecord 매크로는 MCI 디바이스를 사용하여 콘텐츠 기록을 시작합니다. 녹음/녹화 프로세스는 콘텐츠의 현재 위치에서 시작되며 녹음/녹화 기간 동안 기존 데이터를 덮어씁니다.
MCIWndRegisterClass MCIWndRegisterClass 함수는 MCI 창 클래스 MCIWND_WINDOW_CLASS 등록합니다.
MCIWndResume MCIWndResume 매크로는 일시 중지된 모드에서 재생 또는 녹화 콘텐츠를 다시 시작합니다. 이 매크로는 콘텐츠의 현재 위치에서 재생 또는 녹화를 다시 시작합니다. 이 매크로를 사용하거나 MCI_RESUME 명령을 명시적으로 보낼 수 있습니다.
MCIWndReturnString MCIWndReturnString 매크로는 MCI 디바이스로 전송된 최신 MCI 문자열 명령에 대한 응답을 검색합니다. 회신의 정보는 null로 끝나는 문자열로 제공됩니다. 이 매크로를 사용하거나 MCIWNDM_RETURNSTRING 메시지를 명시적으로 보낼 수 있습니다.
MCIWndSave MCIWndSave 매크로는 MCI 디바이스에서 현재 사용하는 콘텐츠를 저장합니다.
MCIWndSaveDialog MCIWndSaveDialog 매크로는 MCI 디바이스에서 현재 사용하는 콘텐츠를 저장합니다. 이 매크로는 사용자가 콘텐츠를 저장할 파일 이름을 선택할 수 있도록 저장 대화 상자를 표시합니다. 이 매크로를 사용하거나 MCI_SAVE 명령을 명시적으로 보낼 수 있습니다.
MCIWndSeek MCIWndSeek 매크로는 재생 위치를 콘텐츠의 지정된 위치로 이동합니다. 이 매크로를 사용하거나 MCI_SEEK 명령을 명시적으로 사용할 수 있습니다.
MCIWndSendString MCIWndSendString 매크로는 MCIWnd 창과 연결된 디바이스에 문자열 형식의 MCI 명령을 보냅니다. 이 매크로를 사용하거나 MCIWNDM_SENDSTRING 메시지를 명시적으로 보낼 수 있습니다.
MCIWndSetActiveTimer MCIWndSetActiveTimer 매크로는 MCIWnd에서 MCIWnd 창에서 트랙바를 업데이트하고, 창 제목 표시줄에 표시된 위치 정보를 업데이트하고, MCIWnd 창이 활성화된 경우 부모 창에 알림 메시지를 보내는 데 사용하는 업데이트 기간을 설정합니다. 이 매크로를 사용하거나 MCIWNDM_SETACTIVETIMER 메시지를 명시적으로 보낼 수 있습니다.

MCIWndSetInactiveTimer MCIWndSetInactiveTimer 매크로는 MCIWnd에서 MCIWnd 창에서 트랙바를 업데이트하고, 창 제목 표시줄에 표시되는 위치 정보를 업데이트하고, MCIWnd 창이 비활성 상태일 때 부모 창에 알림 메시지를 보내는 데 사용하는 업데이트 기간을 설정합니다. 이 매크로를 사용하거나 MCIWNDM_SETINACTIVETIMER 메시지를 명시적으로 보낼 수 있습니다.
MCIWndSetOwner MCIWndSetOwner 매크로는 MCIWnd 창과 연결된 알림 메시지를 받도록 창을 설정합니다. 이 매크로를 사용하거나 MCIWNDM_SETOWNER 메시지를 명시적으로 보낼 수 있습니다.
MCIWndSetPalette MCIWndSetPalette 매크로는 MCIWnd 창과 연결된 MCI 디바이스에 팔레트 핸들을 보냅니다. 이 매크로를 사용하거나 MCIWNDM_SETPALETTE 메시지를 명시적으로 보낼 수 있습니다.
MCIWndSetRepeat MCIWndSetRepeat 매크로는 연속 재생과 연결된 반복 플래그를 설정합니다. 이 매크로를 사용하거나 MCIWNDM_SETREPEAT 메시지를 명시적으로 보낼 수 있습니다.
MCIWndSetSpeed MCIWndSetSpeed 매크로는 MCI 디바이스의 재생 속도를 설정합니다. 이 매크로를 사용하거나 MCIWNDM_SETSPEED 메시지를 명시적으로 보낼 수 있습니다.
MCIWndSetTimeFormat MCIWndSetTimeFormat 매크로는 MCI 디바이스의 시간 형식을 설정합니다. 이 매크로를 사용하거나 MCIWNDM_SETTIMEFORMAT 메시지를 명시적으로 보낼 수 있습니다.
MCIWndSetTimers MCIWndSetTimers 매크로는 MCIWnd에서 MCIWnd 창에서 트랙바를 업데이트하고, 창 제목 표시줄에 표시되는 위치 정보를 업데이트하고, 부모 창에 알림 메시지를 보내는 데 사용하는 업데이트 기간을 설정합니다.
MCIWndSetVolume MCIWndSetVolume 매크로는 MCI 디바이스의 볼륨 수준을 설정합니다. 이 매크로를 사용하거나 MCIWNDM_SETVOLUME 메시지를 명시적으로 보낼 수 있습니다.
MCIWndSetZoom MCIWndSetZoom 매크로는 확대/축소 비율에 따라 비디오 이미지의 크기를 조정합니다. 이 마르코는 일정한 가로 세로 비율을 유지하면서 MCIWnd 창의 크기를 조정합니다. 이 매크로를 사용하거나 MCIWNDM_SETZOOM 메시지를 명시적으로 보낼 수 있습니다.

MCIWndStep MCIWndStep 매크로는 콘텐츠의 현재 위치를 지정된 증분 단위로 앞으로 또는 뒤로 이동합니다. 이 매크로를 사용하거나 MCI_STEP 명령을 명시적으로 보낼 수 있습니다.
MCIWndStop MCIWndStop 매크로는 MCIWnd 창과 연결된 MCI 디바이스의 콘텐츠 재생 또는 기록을 중지합니다. 이 매크로를 사용하거나 MCI_STOP 명령을 명시적으로 보낼 수 있습니다.
MCIWndUseFrames MCIWndUseFrames 매크로는 MCI 디바이스의 시간 형식을 프레임으로 설정합니다. 이 매크로를 사용하거나 MCIWNDM_SETTIMEFORMAT 메시지를 명시적으로 보낼 수 있습니다.
MCIWndUseTime MCIWndUseTime 매크로는 MCI 디바이스의 시간 형식을 밀리초로 설정합니다. 이 매크로를 사용하거나 MCIWNDM_SETTIMEFORMAT 메시지를 명시적으로 보낼 수 있습니다.
MCIWndValidateMedia MCIWndValidateMedia 매크로는 콘텐츠의 시작 및 끝 위치, 콘텐츠의 현재 위치 및 트랙바를 현재 시간 형식에 따라 업데이트합니다.
MEVT_EVENTPARM MEVT_EVENTPARM 매크로는 MIDIEVENT 구조체의 dwEvent 멤버에 지정된 값에서 이벤트 매개 변수 또는 길이를 검색합니다.
MEVT_EVENTTYPE MEVT_EVENTTYPE 매크로는 MIDIEVENT 구조체의 dwEvent 멤버에 지정된 값에서 이벤트 형식을 검색합니다.
midiConnect midiConnect 함수는 MIDI 입력 디바이스를 MIDI 스루 또는 출력 디바이스에 연결하거나 MIDI를 통해 디바이스를 MIDI 출력 디바이스에 연결합니다.
midiDisconnect midiDisconnect 함수는 MIDI를 통해 또는 출력 디바이스에서 MIDI 입력 디바이스의 연결을 끊거나 MIDI 출력 디바이스에서 MIDI를 통해 디바이스의 연결을 끊습니다.
midiInAddBuffer midiInAddBuffer 함수는 지정된 열린 MIDI 입력 디바이스에 입력 버퍼를 보냅니다. 이 함수는 시스템 전용 메시지에 사용됩니다.

midiInClose midiInClose 함수는 지정된 MIDI 입력 디바이스를 닫습니다.
midiInGetDevCaps midiInGetDevCaps 함수(mmeapi.h)는 지정된 MIDI 입력 디바이스의 기능을 결정합니다.
midiInGetDevCapsA midiInGetDevCaps 함수는 지정된 MIDI 입력 디바이스의 기능을 결정합니다. (midiInGetDevCapsA)
midiInGetDevCapsW midiInGetDevCapsW(유니코드) 함수(mmeapi.h)는 지정된 MIDI 입력 디바이스의 기능을 결정합니다.
midiInGetErrorText midiInGetErrorText 함수(mmeapi.h)는 지정된 오류 코드로 식별된 오류에 대한 텍스트 설명을 검색합니다.
midiInGetErrorTextA midiInGetErrorText 함수는 지정된 오류 코드로 식별된 오류에 대한 텍스트 설명을 검색합니다. (midiInGetErrorTextA)
midiInGetErrorTextW midiInGetErrorTextW(유니코드) 함수(mmeapi.h)는 지정된 오류 코드로 식별된 오류에 대한 텍스트 설명을 검색합니다.
midiInGetID midiInGetID 함수는 지정된 MIDI 입력 디바이스에 대한 디바이스 식별자를 가져옵니다.
midiInGetNumDevs midiInGetNumDevs 함수는 시스템의 MIDI 입력 디바이스 수를 검색합니다.
midiInMessage midiInMessage 함수는 MIDI 디바이스 드라이버에 메시지를 보냅니다.
midiInOpen midiInOpen 함수는 지정된 MIDI 입력 디바이스를 엽니다.

midiInPrepareHeader midiInPrepareHeader 함수는 MIDI 입력을 위한 버퍼를 준비합니다.
midiInReset midiInReset 함수는 지정된 MIDI 입력 디바이스에서 입력을 중지합니다.
midiInStart midiInStart 함수는 지정된 MIDI 입력 디바이스에서 MIDI 입력을 시작합니다.
midiInStop midiInStop 함수는 지정된 MIDI 입력 디바이스에서 MIDI 입력을 중지합니다.
midiInUnprepareHeader midiInUnprepareHeader 함수는 midiInPrepareHeader 함수에서 수행하는 준비를 정리합니다.
midiOutCacheDrumPatches midiOutCacheDrumPatches 함수는 내부 MIDI 신시사이저 디바이스가 지정된 키 기반 타악기 패치 집합을 미리 로드하고 캐시하도록 요청합니다.
midiOutCachePatches midiOutCachePatches 함수는 내부 MIDI 신시사이저 디바이스가 지정된 패치 집합을 미리 로드하고 캐시하도록 요청합니다.
midiOutClose midiOutClose 함수는 지정된 MIDI 출력 디바이스를 닫습니다.
midiOutGetDevCaps midiOutGetDevCaps 함수(mmeapi.h)는 지정된 MIDI 출력 디바이스를 쿼리하여 해당 기능을 확인합니다.
midiOutGetDevCapsA midiOutGetDevCaps 함수는 지정된 MIDI 출력 디바이스를 쿼리하여 해당 기능을 확인합니다. (midiOutGetDevCapsA)
midiOutGetDevCapsW midiOutGetDevCapsW(유니코드) 함수(mmeapi.h)는 지정된 MIDI 출력 디바이스를 쿼리하여 해당 기능을 확인합니다.

midiOutGetErrorText midiOutGetErrorText(mmeapi.h) 함수는 지정된 오류 코드로 식별된 오류에 대한 텍스트 설명을 검색합니다.
midiOutGetErrorTextA midiOutGetErrorText 함수는 지정된 오류 코드로 식별된 오류에 대한 텍스트 설명을 검색합니다. (midiOutGetErrorTextA)
midiOutGetErrorTextW midiOutGetErrorTextW(유니코드) 함수(mmeapi.h)는 지정된 오류 코드로 식별된 오류에 대한 텍스트 설명을 검색합니다.
midiOutGetID midiOutGetID 함수는 지정된 MIDI 출력 디바이스에 대한 디바이스 식별자를 검색합니다.
midiOutGetNumDevs midiOutGetNumDevs 함수는 시스템에 있는 MIDI 출력 디바이스 수를 검색합니다.
midiOutGetVolume midiOutGetVolume 함수는 MIDI 출력 디바이스의 현재 볼륨 설정을 검색합니다.
midiOutLongMsg midiOutLongMsg 함수는 지정된 MIDI 출력 디바이스에 시스템 전용 MIDI 메시지를 보냅니다.
midiOutMessage midiOutMessage 함수는 MIDI 디바이스 드라이버에 메시지를 보냅니다. 이 함수는 MIDI API에서 지원되지 않는 드라이버 관련 메시지에만 사용됩니다.
midiOutOpen midiOutOpen 함수는 재생을 위해 MIDI 출력 디바이스를 엽니다.
midiOutPrepareHeader midiOutPrepareHeader 함수는 출력을 위해 MIDI 시스템 전용 또는 스트림 버퍼를 준비합니다.
midiOutReset midiOutReset 함수는 지정된 MIDI 출력 디바이스에 대한 모든 MIDI 채널의 모든 메모를 해제합니다.

midiOutSetVolume midiOutSetVolume 함수는 MIDI 출력 디바이스의 볼륨을 설정합니다.
midiOutShortMsg midiOutShortMsg 함수는 지정된 MIDI 출력 디바이스에 짧은 MIDI 메시지를 보냅니다.
midiOutUnprepareHeader midiOutUnprepareHeader 함수는 midiOutPrepareHeader 함수에서 수행하는 준비를 정리합니다.
midiStreamClose midiStreamClose 함수는 열린 MIDI 스트림을 닫습니다.
midiStreamOpen midiStreamOpen 함수는 출력을 위한 MIDI 스트림을 엽니다. 기본적으로 디바이스는 일시 중지 모드로 열립니다. 이 함수에서 검색한 스트림 핸들은 스트림에 대한 모든 후속 참조에서 사용해야 합니다.
midiStreamOut midiStreamOut 함수는 MIDI 데이터의 스트림(버퍼)을 MIDI 출력 디바이스로 재생하거나 큐에 대기합니다.
midiStreamPause midiStreamPause 함수는 지정된 MIDI 스트림의 재생을 일시 중지합니다.
midiStreamPosition midiStreamPosition 함수는 MIDI 스트림에서 현재 위치를 검색합니다.
midiStreamProperty midiStreamProperty 함수는 MIDI 출력 디바이스와 연결된 MIDI 데이터 스트림의 속성을 설정하거나 검색합니다.
midiStreamRestart midiStreamRestart 함수는 일시 중지된 MIDI 스트림을 다시 시작합니다.
midiStreamStop midiStreamStop 함수는 지정된 MIDI 출력 디바이스에 대한 모든 MIDI 채널의 모든 메모를 해제합니다.

mixerClose mixerClose 함수는 지정된 믹서 디바이스를 닫습니다.
mixerGetControlDetails mixerGetControlDetails 함수(mmeapi.h)는 오디오 라인과 연결된 단일 컨트롤에 대한 세부 정보를 검색합니다.
mixerGetControlDetailsA mixerGetControlDetails 함수는 오디오 라인과 연결된 단일 컨트롤에 대한 세부 정보를 검색합니다. (mixerGetControlDetailsA)
mixerGetControlDetailsW mixerGetControlDetailsW(유니코드) 함수(mmeapi.h)는 오디오 라인과 연결된 단일 컨트롤에 대한 세부 정보를 검색합니다.
mixerGetDevCaps mixerGetDevCaps 함수(mmeapi.h)는 지정된 믹서 디바이스를 쿼리하여 해당 기능을 확인합니다.
mixerGetDevCapsA mixerGetDevCaps 함수는 지정된 믹서 디바이스를 쿼리하여 해당 기능을 확인합니다. (mixerGetDevCapsA)
mixerGetDevCapsW mixerGetDevCapsW(유니코드) 함수(mmeapi.h)는 지정된 믹서 디바이스를 쿼리하여 해당 기능을 확인합니다.
mixerGetID mixerGetID 함수는 지정된 디바이스 핸들과 연결된 믹서 디바이스의 디바이스 식별자를 검색합니다.
mixerGetLineControls mixerGetLineControls 함수(mmeapi.h)는 오디오 라인과 연결된 하나 이상의 컨트롤을 검색합니다.
mixerGetLineControlsA mixerGetLineControls 함수는 오디오 라인과 연결된 하나 이상의 컨트롤을 검색합니다. (mixerGetLineControlsA)
mixerGetLineControlsW

mixerGetLineControlsW(유니코드) 함수는 오디오 라인과 연결된 하나 이상의 컨트롤을 검색합니다. (mixerGetLineControlsW)
mixerGetLineInfo mixerGetLineInfo 함수는 믹서 디바이스의 특정 줄에 대한 정보를 검색하고 함수를 수정하는 매개 변수를 포함합니다.
mixerGetLineInfoA mixerGetLineInfo 함수는 믹서 디바이스의 특정 줄에 대한 정보를 검색합니다. (mixerGetLineInfoA)
mixerGetLineInfoW mixerGetLineInfoW(유니코드) 함수는 믹서 디바이스의 특정 줄에 대한 정보를 검색합니다. (mixerGetLineInfoW)
mixerGetNumDevs mixerGetNumDevs 함수는 시스템에 있는 믹서 디바이스 수를 검색합니다.
mixerMessage mixerMessage 함수는 사용자 지정 믹서 드라이버 메시지를 믹서 드라이버에 직접 보냅니다.
mixerOpen mixerOpen 함수는 지정된 믹서 디바이스를 열고 애플리케이션이 핸들을 닫을 때까지 디바이스가 제거되지 않도록 합니다.
mixerSetControlDetails mixerSetControlDetails 함수는 오디오 라인과 연결된 단일 컨트롤의 속성을 설정합니다.
mmGetCurrentTask mmGetCurrentTask 함수는 더 이상 사용되지 않습니다. 애플리케이션에서 이 함수를 사용하면 안 됩니다.
mmioAdvance mmioAdvance 함수는 mmioGetInfo 함수를 사용하여 직접 I/O 버퍼 액세스를 위해 설정된 파일의 I/O 버퍼를 진행합니다.
mmioAscend mmioAscend 함수는 mmioDescend 함수를 사용하여 내림차순으로 내림차순으로 또는 mmioCreateChunk 함수를 사용하여 만든 RIFF 파일의 청크에서 오름차순으로 상승합니다.

mmioClose mmioClose 함수는 mmioOpen 함수를 사용하여 연 파일을 닫습니다.
mmioCreateChunk mmioCreateChunk 함수는 mmioOpen 함수를 사용하여 연 RIFF 파일에 청크를 만듭니다.
mmioDescend mmioDescend 함수는 mmioOpen 함수를 사용하여 연 RIFF 파일의 청크로 내림차순입니다. 지정된 청크를 검색할 수도 있습니다.
mmioFlush mmioFlush 함수는 버퍼가 기록된 경우 파일의 I/O 버퍼를 디스크에 씁니다.
mmioFOURCC mmioFOURCC 매크로는 4자를 4자 코드로 변환합니다.
mmioGetInfo mmioGetInfo 함수는 mmioOpen 함수를 사용하여 연 파일에 대한 정보를 검색합니다. 이 정보를 사용하면 파일이 버퍼링된 I/O용으로 열려 있는 경우 애플리케이션이 I/O 버퍼에 직접 액세스할 수 있습니다.
mmioInstallIOProc mmioInstallIOProc 함수는 사용자 지정 I/O 프로시저를 설치하거나 제거하고 해당 4자 코드를 사용하여 설치된 I/O 프로시저를 찾습니다.
mmioInstallIOProcA mmioInstallIOProc 함수는 사용자 지정 I/O 프로시저를 설치하거나 제거합니다. 또한 이 함수는 해당 4자 코드를 사용하여 설치된 I/O 프로시저를 찾습니다. (mmioInstallIOProcA)
mmioInstallIOProcW mmioInstallIOProcW(유니코드) 함수는 사용자 지정 I/O 프로시저를 설치하거나 제거하고 해당 코드를 사용하여 설치된 I/O 프로시저를 찾습니다.
mmioOpen mmioOpen 함수는 버퍼링되지 않거나 버퍼링된 I/O에 대한 파일을 엽니다. 는 파일을 만듭니다. 파일을 삭제합니다. 또는 파일이 있는지 확인합니다.
mmioOpenA

mmioOpen 함수는 버퍼링되지 않거나 버퍼링된 I/O에 대한 파일을 엽니다. 는 파일을 만듭니다. 파일을 삭제합니다. 또는 파일이 있는지 확인합니다. (mmioOpenA)
mmioOpenW mmioOpenW(유니코드) 함수는 버퍼링되지 않거나 버퍼링된 I/O에 대한 파일을 엽니다. 는 파일을 만듭니다. 파일을 삭제합니다. 또는 파일이 있는지 확인합니다. (mmioOpenW)
MMIOPROC MMIOProc 함수는 mmioInstallIOProc 함수에 의해 설치된 사용자 지정 입력/출력(I/O) 프로시저입니다.
mmioRead mmioRead 함수는 mmioOpen 함수를 사용하여 연 파일에서 지정된 바이트 수를 읽습니다.
mmioRename mmioRename 함수는 지정된 파일의 이름을 바꾸고 파일 이름을 포함하는 문자열을 수정하는 매개 변수를 포함합니다.
mmioRenameA mmioRename 함수는 지정된 파일의 이름을 바꿉니다. (mmioRenameA)
mmioRenameW mmioRenameW(유니코드) 함수는 지정된 파일의 이름을 바꾸고 파일 이름을 포함하는 문자열을 수정하는 매개 변수를 포함합니다. (mmioRenameW)
mmioSeek mmioSeek 함수는 mmioOpen 함수를 사용하여 연 파일의 현재 파일 위치를 변경합니다.
mmioSendMessage mmioSendMessage 함수는 지정된 파일과 연결된 I/O 프로시저에 메시지를 보냅니다.
mmioSetBuffer mmioSetBuffer 함수는 버퍼링된 I/O를 사용하거나 사용하지 않도록 설정하거나 mmioOpen 함수를 사용하여 연 파일의 버퍼 또는 버퍼 크기를 변경합니다.
mmioSetInfo mmioSetInfo 함수는 mmioOpen 함수를 사용하여 연 파일에 대한 mmioGetInfo 함수에서 검색한 정보를 업데이트합니다. 이 함수를 사용하여 버퍼링된 I/O에 대해 열린 파일의 직접 버퍼 액세스를 종료합니다.

mmioStringToFOURCC mmioStringToFOURCC 함수는 null로 끝나는 문자열을 4자 코드로 변환하고 변환을 수정하는 매개 변수를 포함합니다. (mmioStringToFOURCCW)
mmioStringToFOURCCA mmioStringToFOURCC 함수는 null로 끝나는 문자열을 4자 코드로 변환합니다. (mmioStringToFOURCCA)
mmioStringToFOURCCW mmioStringToFOURCCW(유니코드) 함수는 null로 끝나는 문자열을 4자 코드로 변환합니다. (mmioStringToFOURCCW)
mmioWrite mmioWrite 함수는 mmioOpen 함수를 사용하여 열린 파일에 지정된 바이트 수를 씁니다.
mmTaskBlock mmTaskBlock 함수는 더 이상 사용되지 않습니다. 애플리케이션에서 이 함수를 사용하면 안 됩니다.
mmTaskCreate mmTaskCreate 함수는 더 이상 사용되지 않습니다. 애플리케이션에서 이 함수를 사용하면 안 됩니다.
mmTaskSignal mmTaskSignal 함수는 더 이상 사용되지 않습니다. 애플리케이션에서 이 함수를 사용하면 안 됩니다.
mmTaskYield mmTaskYield 함수는 더 이상 사용되지 않습니다. 애플리케이션에서 이 함수를 사용하면 안 됩니다.
OpenDriver 설치 가능한 드라이버의 instance 열고 드라이버의 기본 설정 또는 드라이버별 값을 사용하여 instance 초기화합니다.
붙여넣기 Paste 메서드는 스트림 또는 스트림의 일부를 다른 스트림에 복사합니다. 애플리케이션이 EditStreamPaste 함수를 사용할 때 호출됩니다.

읽기 Read 메서드는 스트림에서 데이터를 읽고 애플리케이션 정의 버퍼에 복사합니다. 버퍼가 제공되지 않으면 데이터의 다음 버퍼를 검색하는 데 필요한 버퍼 크기를 결정합니다. 애플리케이션이 AVIStreamRead 함수를 사용할 때 호출됩니다.
ReadData ReadData 메서드는 파일 헤더를 읽습니다. 애플리케이션이 AVIFileReadData 함수를 사용할 때 호출됩니다.
ReadData ReadData 메서드는 스트림의 데이터 헤더를 읽습니다. 애플리케이션이 AVIStreamReadData 함수를 사용할 때 호출됩니다.
ReadFormat ReadFormat 메서드는 스트림에서 형식 정보를 가져옵니다.
SendDriverMessage 지정된 메시지를 설치 가능한 드라이버에 보냅니다.
SetFormat SetFormat 메서드는 스트림의 형식 정보를 설정합니다. 애플리케이션이 AVIStreamSetFormat 함수를 사용할 때 호출됩니다.
SetFormat SetFormat 메서드는 추출되는 프레임의 압축 해제된 이미지 형식을 설정하고 압축 해제 작업에 대한 버퍼를 선택적으로 제공합니다.
SetInfo SetInfo 메서드는 스트림의 특성을 변경합니다. 애플리케이션에서 EditStreamSetInfo 함수를 사용할 때 호출됩니다.
sndOpenSound 지정된 사운드 이벤트를 엽니다.
StretchDIB StretchDIB 함수는 디바이스 독립적 비트맵을 한 메모리 위치에서 다른 메모리 위치로 복사하고 대상 사각형에 맞게 이미지 크기를 조정합니다.

timeBeginPeriod timeBeginPeriod 함수는 주기적 타이머에 대한 최소 해상도를 요청합니다.
timeEndPeriod timeEndPeriod 함수는 이전에 설정된 최소 타이머 해상도를 지웁니다.
timeGetDevCaps timeGetDevCaps 함수는 타이머 디바이스를 쿼리하여 해상도를 확인합니다.
timeGetSystemTime timeGetSystemTime 함수는 시스템 시간을 밀리초 단위로 검색합니다.
timeGetTime timeGetTime 함수는 시스템 시간을 밀리초 단위로 검색합니다. 시스템 시간은 Windows가 시작된 이후 경과된 시간입니다.
waveInAddBuffer waveInAddBuffer 함수는 지정된 파형 오디오 입력 디바이스에 입력 버퍼를 보냅니다. 버퍼가 채워지면 애플리케이션에 알림이 표시됩니다.
waveInClose waveInClose 함수는 지정된 파형 오디오 입력 디바이스를 닫습니다.
waveInGetDevCaps waveInGetDevCaps 함수는 지정된 파형 오디오 입력 디바이스의 기능을 검색합니다.
waveInGetErrorText waveInGetErrorText 함수는 지정된 오류 번호로 식별된 오류에 대한 텍스트 설명을 검색합니다.
waveInGetID waveInGetID 함수는 지정된 파형 오디오 입력 디바이스에 대한 디바이스 식별자를 가져옵니다.
waveInGetNumDevs waveInGetNumDevs 함수는 시스템에 있는 파형 오디오 입력 디바이스의 수를 반환합니다.
waveInGetPosition waveInGetPosition은 Windows Vista를 기준으로 더 이상 사용할 수 없습니다.

waveInMessage waveInMessage 함수는 파형 오디오 입력 디바이스 드라이버에 메시지를 보냅니다.
waveInOpen waveInOpen 함수는 녹음을 위해 지정된 파형 오디오 입력 디바이스를 엽니다.
waveInPrepareHeader waveInPrepareHeader 함수는 파형 오디오 입력을 위한 버퍼를 준비합니다.
waveInReset waveInReset 함수는 지정된 파형 오디오 입력 디바이스에서 입력을 중지하고 현재 위치를 0으로 다시 설정합니다. 보류 중인 모든 버퍼는 완료된 것으로 표시되고 애플리케이션에 반환됩니다.
waveInStart waveInStart 함수는 지정된 파형 오디오 입력 디바이스에서 입력을 시작합니다.
waveInStop waveInStop 함수는 파형 오디오 입력을 중지합니다.
waveInUnprepareHeader waveInUnprepareHeader 함수는 waveInPrepareHeader 함수에 의해 수행되는 준비를 정리합니다.
waveOutBreakLoop waveOutBreakLoop 함수는 지정된 파형 오디오 출력 디바이스에서 루프를 중단하고 드라이버 목록의 다음 블록으로 재생을 계속할 수 있습니다.
waveOutClose waveOutClose 함수는 지정된 파형 오디오 출력 디바이스를 닫습니다.
waveOutGetDevCaps waveOutGetDevCaps 함수는 지정된 파형 오디오 출력 디바이스의 기능을 검색합니다.
waveOutGetErrorText waveOutGetErrorText 함수는 지정된 오류 번호로 식별된 오류에 대한 텍스트 설명을 검색합니다.
waveOutGetID waveOutGetID 함수는 지정된 파형 오디오 출력 디바이스에 대한 디바이스 식별자를 검색합니다.

waveOutGetNumDevs waveOutGetNumDevs 함수는 시스템에 있는 파형 오디오 출력 디바이스의 수를 검색합니다.
waveOutGetPitch waveOutGetPitch 함수는 지정된 파형 오디오 출력 디바이스에 대한 현재 피치 설정을 검색합니다.
waveOutGetPlaybackRate waveOutGetPlaybackRate 함수는 지정된 파형 오디오 출력 디바이스에 대한 현재 재생 속도를 검색합니다.
waveOutGetPosition waveOutGetPosition 함수는 지정된 파형 오디오 출력 디바이스의 현재 재생 위치를 검색합니다.
waveOutGetVolume waveOutGetVolume 함수는 지정된 파형 오디오 출력 디바이스의 현재 볼륨 수준을 검색합니다.
waveOutMessage waveOutMessage 함수는 파형 오디오 출력 디바이스 드라이버에 메시지를 보냅니다.
waveOutOpen waveOutOpen 함수는 재생을 위해 지정된 파형 오디오 출력 디바이스를 엽니다.
waveOutPause waveOutPause 함수는 지정된 파형 오디오 출력 디바이스에서 재생을 일시 중지합니다. 현재 위치가 저장됩니다. waveOutRestart 함수를 사용하여 현재 위치에서 재생을 다시 시작합니다.
waveOutPrepareHeader waveOutPrepareHeader 함수는 재생을 위해 파형 오디오 데이터 블록을 준비합니다.
waveOutReset waveOutReset 함수는 지정된 파형 오디오 출력 디바이스에서 재생을 중지하고 현재 위치를 0으로 다시 설정합니다. 보류 중인 모든 재생 버퍼는 완료(WHDR_DONE)로 표시되고 애플리케이션에 반환됩니다.
waveOutRestart waveOutRestart 함수는 일시 중지된 파형 오디오 출력 디바이스에서 재생을 다시 시작합니다.

waveOutSetPitch waveOutSetPitch 함수는 지정된 파형 오디오 출력 디바이스의 피치를 설정합니다.
waveOutSetPlaybackRate waveOutSetPlaybackRate 함수는 지정된 파형 오디오 출력 디바이스의 재생 속도를 설정합니다.
waveOutSetVolume waveOutSetVolume 함수는 지정된 파형 오디오 출력 디바이스의 볼륨 수준을 설정합니다.
waveOutUnprepareHeader waveOutUnprepareHeader 함수는 waveOutPrepareHeader 함수에서 수행하는 준비를 정리합니다. 이 함수는 디바이스 드라이버가 데이터 블록으로 완료된 후에 호출되어야 합니다. 버퍼를 해제하기 전에 이 함수를 호출해야 합니다.
waveOutWrite waveOutWrite 함수는 지정된 파형 오디오 출력 디바이스에 데이터 블록을 보냅니다.
쓰기 Write 메서드는 스트림에 데이터를 씁니다. 애플리케이션이 AVIStreamWrite 함수를 사용할 때 호출됩니다.
WriteData WriteData 메서드는 파일 헤더를 씁니다. 애플리케이션이 AVIFileWriteData 함수를 사용할 때 호출됩니다.
WriteData WriteData 메서드는 스트림에 대한 헤더를 씁니다. 애플리케이션이 AVIStreamWriteData 함수를 사용할 때 호출됩니다.

인터페이스

IAVIEditStream IAVIEditStream 인터페이스는 편집 가능한 스트림 조작 및 수정을 지원합니다. 다음 사용자 지정 메서드 외에도 IUnknown::QueryInterface, IUnknown::AddRef, IUnknown::Release를 사용합니다.

IAVIFile IAVIFile 인터페이스는 파일 및 파일 헤더를 열고 조작하고 스트림 인터페이스를 만들고 가져오는 것을 지원합니다. 다음 사용자 지정 메서드 외에도 IUnknown::QueryInterface, IUnknown::AddRef 및 IUnknown::Release를 사용합니다.
IAVIStream IAVIStream 인터페이스는 파일 내에서 데이터 스트림을 만들고 조작하는 것을 지원합니다. 다음 사용자 지정 메서드 외에도 IUnknown::QueryInterface, IUnknown::AddRef, IUnknown::Release를 사용합니다.
IAVIStreaming IAVIStreaming 인터페이스는 스트리밍 작업에서 재생을 위해 열린 데이터 스트림을 준비할 수 있습니다. 다음 사용자 지정 메서드 외에도 IUnknown::QueryInterface, IUnknown::AddRef, IUnknown::Release를 사용합니다.
IGetFrame IGetFrame 인터페이스는 열린 스트림에서 개별 프레임의 추출, 압축 해제 및 표시를 지원합니다.

구조체

ACMDRIVERDETAILS ACMDRIVERDETAILS 구조체는 ACM 드라이버의 기능을 설명합니다.
ACMFILTERCHOOSE ACMFILTERCHOOSE 구조체에는 ACM이 시스템 정의 파형 오디오 필터 선택 대화 상자를 초기화하는 데 사용하는 정보가 포함되어 있습니다.
ACMFILTERDETAILS ACMFILTERDETAILS 구조체는 ACM 드라이버에 대한 특정 필터 태그에 대한 파형 오디오 필터를 자세히 설명합니다.
ACMFILTERTAGDETAILS ACMFILTERTAGDETAILS 구조체는 ACM 필터 드라이버에 대한 파형 오디오 필터 태그를 자세히 설명합니다.
ACMFORMATCHOOSE

ACMFORMATCHOOSE 구조체에는 ACM이 시스템 정의 파형 오디오 형식 선택 대화 상자를 초기화하는 데 사용하는 정보가 포함되어 있습니다.
ACMFORMATDETAILS ACMFORMATDETAILS 구조체는 ACM 드라이버의 특정 형식 태그에 대한 파형 오디오 형식을 자세히 설명합니다.
ACMFORMATTAGDETAILS ACMFORMATTAGDETAILS 구조체는 ACM 드라이버에 대한 파형 오디오 형식 태그를 자세히 설명합니다.
ACMSTREAMHEADER ACMSTREAMHEADER 구조는 변환 스트림에 대한 ACM 변환 원본 및 대상 버퍼 쌍을 식별하는 데 사용되는 헤더를 정의합니다.
AUXCAPS AUXCAPS 구조는 보조 출력 디바이스의 기능을 설명합니다. (AUXCAPS)
AUXCAPSA AUXCAPS 구조는 보조 출력 디바이스의 기능을 설명합니다. (AUXCAPSA)
AUXCAPSW AUXCAPS 구조는 보조 출력 디바이스의 기능을 설명합니다. (AUXCAPSW)
AVICOMPRESSOPTIONS AVICOMPRESSOPTIONS 구조체에는 스트림에 대한 정보와 스트림이 압축 및 저장되는 방법에 대한 정보가 포함되어 있습니다. 이 구조체는 데이터를 AVIMakeCompressedStream 함수(또는 AVIMakeCompressedStream을 사용하는 AVISave 함수)에 전달합니다.
AVIFILEINFOA AVIFILEINFO 구조체에는 전체 AVI 파일에 대한 전역 정보가 포함되어 있습니다. (ANSI)
AVIFILEINFOW AVIFILEINFO 구조체에는 전체 AVI 파일에 대한 전역 정보가 포함되어 있습니다. (유니코드)
AVISTREAMINFOA AVISTREAMINFO 구조체에는 단일 스트림에 대한 정보가 포함되어 있습니다. (ANSI)
AVISTREAMINFOW

AVISTREAMINFO 구조체에는 단일 스트림에 대한 정보가 포함되어 있습니다. (유니코드)
CAPDRIVERCAPS CAPDRIVERCAPS 구조체는 캡처 드라이버의 기능을 정의합니다. 애플리케이션은 캡처 창을 캡처 드라이버에 연결할 때마다 WM_CAP_DRIVER_GET_CAPS 메시지 또는 capDriverGetCaps 매크로를 사용하여 드라이버 기능의 복사본을 CAPDRIVERCAPS 구조에 배치해야 합니다.
CAPINFOCHUNK CAPINFOCHUNK 구조체에는 AVI 캡처 파일 내에서 정보 청크를 정의하는 데 사용할 수 있는 매개 변수가 포함되어 있습니다. WM_CAP_FILE_SET_INFOCHUNK 메시지 또는 capSetInfoChunk 매크로는 CAPINFOCHUNK 구조를 캡처 창으로 보내는 데 사용됩니다.
CAPSTATUS CAPSTATUS 구조체는 캡처 창의 현재 상태를 정의합니다.
CAPTUREPARMS CAPTUREPARMS 구조에는 스트리밍 비디오 캡처 프로세스를 제어하는 매개 변수가 포함되어 있습니다. 이 구조는 캡처 속도, 캡처하는 동안 사용할 버퍼 수 및 캡처가 종료되는 방법에 영향을 주는 매개 변수를 가져와서 설정하는 데 사용됩니다.
COMPVARS COMPVARS 구조는 ICCompressorChoose, ICSeqCompressFrame 및 ICCompressorFree와 같은 함수에 대한 압축기 설정을 설명합니다.
DRAWDIBTIME DRAWDIBTIME 구조에는 DrawDib 작업 집합을 수행하기 위한 경과된 타이밍 정보가 포함되어 있습니다. DrawDibTime 함수는 호출될 때마다 각 작업에 대한 개수 및 경과된 시간 값을 다시 설정합니다.
DRVCONFIGINFO 설치 가능한 드라이버와 연결된 레지스트리 키 및 값 이름을 포함합니다.
ICCOMPRESS ICCOMPRESS 구조에는 ICM_COMPRESS 메시지와 함께 사용되는 압축 매개 변수가 포함되어 있습니다.
ICCOMPRESSFRAMES ICCOMPRESSFRAMES 구조에는 ICM_COMPRESS_FRAMES_INFO 메시지와 함께 사용되는 압축 매개 변수가 포함되어 있습니다.

ICDECOMPRESS ICDECOMPRESS 구조체에는 ICM_DECOMPRESS 메시지와 함께 사용되는 압축 해제 매개 변수가 포함되어 있습니다.
ICDECOMPRESSEX ICDECOMPRESSEX 구조체에는 ICM_DECOMPRESSEX 메시지와 함께 사용되는 압축 해제 매개 변수가 포함되어 있습니다.
ICDRAW ICDRAW 구조체에는 비디오 데이터를 화면에 그리기 위한 매개 변수가 포함되어 있습니다. 이 구조체는 ICM_DRAW 메시지와 함께 사용됩니다.
ICDRAWBEGIN ICDRAWBEGIN 구조체에는 ICM_DRAW_BEGIN 메시지와 함께 사용되는 압축 해제 매개 변수가 포함되어 있습니다.
ICDRAWSUGGEST ICDRAWSUGGEST 구조체에는 적절한 입력 형식을 제안하기 위해 ICM_DRAW_SUGGESTFORMAT 메시지와 함께 사용되는 압축 매개 변수가 포함되어 있습니다.
ICINFO ICINFO 구조에는 비디오 압축 드라이버에서 제공하는 압축 매개 변수가 포함되어 있습니다. 드라이버는 ICM_GETINFO 메시지를 받을 때 구조를 채우거나 업데이트합니다.
ICOPEN ICOPEN 구조에는 압축 또는 압축 해제되는 데이터 스트림, 드라이버의 버전 번호 및 드라이버 사용 방법에 대한 정보가 포함됩니다.
ICSETSTATUSPROC ICSETSTATUSPROC 구조에는 ICM_SET_STATUS_PROC 메시지와 함께 사용되는 상태 정보가 포함되어 있습니다.
조이캡스 JOYCAPS(joystickapi.h) 구조에는 조이스틱 기능에 대한 중요한 정보가 포함되어 있습니다.
JOYCAPSA JOYCAPS 구조에는 조이스틱 기능에 대한 정보가 포함되어 있습니다. (JOYCAPSA)

JOYCAPSW JOYCAPSW(유니코드)(조이스틱api.h) 구조에는 조이스틱 기능에 대한 정보가 포함되어 있습니다.
JOYINFO JOYINFO 구조체에는 조이스틱 위치 및 단추 상태에 대한 정보가 포함되어 있습니다.
JOYINFOEX JOYINFOEX 구조체에는 조이스틱 위치, 관점 위치 및 단추 상태에 대한 확장된 정보가 포함되어 있습니다.
MCI_DGV_CAPTURE_PARMSA ANSI(MCI_DGV_CAPTURE_PARMSA) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_CAPTURE 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_CAPTURE_PARMSW MCI_DGV_CAPTURE_PARMSW(유니코드) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_CAPTURE 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_COPY_PARMS MCI_DGV_COPY_PARMS 구조에는 디지털 비디오 디바이스에 대한 MCI_COPY 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_CUE_PARMS MCI_DGV_CUE_PARMS 구조에는 디지털 비디오 디바이스에 대한 MCI_CUE 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_CUT_PARMS MCI_DGV_CUT_PARMS 구조에는 디지털 비디오 디바이스에 대한 MCI_CUT 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_DELETE_PARMS MCI_DGV_DELETE_PARMS 구조에는 디지털 비디오 디바이스에 대한 MCI_DELETE 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_INFO_PARMSA ANSI(MCI_DGV_INFO_PARMSA) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_INFO 명령에 대한 매개 변수가 포함되어 있습니다.

MCI_DGV_INFO_PARMSW MCI_DGV_INFO_PARMSW(유니코드) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_INFO 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_LIST_PARMSA ANSI(MCI_DGV_LIST_PARMSA) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_LIST 명령에 대한 정보가 포함되어 있습니다.
MCI_DGV_LIST_PARMSW MCI_DGV_LIST_PARMSW(유니코드) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_LIST 명령에 대한 정보가 포함되어 있습니다.
MCI_DGV_MONITOR_PARMS MCI_DGV_MONITOR_PARMS 구조에는 MCI_MONITOR 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_OPEN_PARMSA ANSI(MCI_DGV_OPEN_PARMSA) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_OPEN 명령에 대한 정보가 포함되어 있습니다.
MCI_DGV_OPEN_PARMSW MCI_DGV_OPEN_PARMSW(유니코드) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_OPEN 명령에 대한 정보가 포함되어 있습니다.
MCI_DGV_PASTE_PARMS MCI_DGV_PASTE_PARMS 구조에는 디지털 비디오 디바이스에 대한 MCI_PASTE 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_QUALITY_PARMSA ANSI(MCI_DGV_QUALITY_PARMSA) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_QUALITY 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_QUALITY_PARMSW MCI_DGV_QUALITY_PARMSW(유니코드) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_QUALITY 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_RECORD_PARMS MCI_DGV_RECORD_PARMS 구조에는 디지털 비디오 디바이스에 대한 MCI_RECORD 명령에 대한 매개 변수가 포함되어 있습니다.

MCI_DGV_RECT_PARMS MCI_DGV_RECT_PARMS 구조에는 디지털 비디오 디바이스에 대한 MCI_FREEZE, MCI_PUT, MCI_UNFREEZE 및 MCI_WHERE 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_RESERVE_PARMSA ANSI(MCI_DGV_RESERVE_PARMSA) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_RESERVE 명령에 대한 정보가 포함되어 있습니다.
MCI_DGV_RESERVE_PARMSW MCI_DGV_RESERVE_PARMSW(유니코드) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_RESERVE 명령에 대한 정보가 포함되어 있습니다.
MCI_DGV_RESTORE_PARMSA ANSI(MCI_DGV_RESTORE_PARMSA) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_RESTORE 명령에 대한 정보가 포함되어 있습니다.
MCI_DGV_RESTORE_PARMSW MCI_DGV_RESTORE_PARMSW(유니코드) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_RESTORE 명령에 대한 정보가 포함되어 있습니다.
MCI_DGV_SAVE_PARMSA ANSI(MCI_DGV_SAVE_PARMSA) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_SAVE 명령에 대한 정보가 포함되어 있습니다.
MCI_DGV_SAVE_PARMSW MCI_DGV_SAVE_PARMSW(유니코드) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_SAVE 명령에 대한 정보가 포함되어 있습니다.
MCI_DGV_SET_PARMS MCI_DGV_SET_PARMS 구조에는 디지털 비디오 디바이스에 대한 MCI_SET 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_SETAUDIO_PARMSA ANSI(MCI_DGV_SETAUDIO_PARMSA) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_SETAUDIO 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_SETAUDIO_PARMSW MCI_DGV_SETAUDIO_PARMSW(유니코드) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_SETAUDIO 명령에 대한 매개 변수가 포함되어 있습니다.

MCI_DGV_SETVIDEO_PARMSA ANSI(MCI_DGV_SETVIDEO_PARMSA) 구조에는 디지털 비디오 디바이스에 대한 MCI_SETVIDEO 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_SETVIDEO_PARMSW MCI_DGV_SETVIDEO_PARMSW(유니코드) 구조에는 디지털 비디오 디바이스에 대한 MCI_SETVIDEO 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_SIGNAL_PARMS MCI_DGV_SIGNAL_PARMS 구조에는 디지털 비디오 디바이스에 대한 MCI_SIGNAL 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_STATUS_PARMSA ANSI(MCI_DGV_STATUS_PARMSA) 구조에는 디지털 비디오 디바이스에 대한 MCI_STATUS 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_STATUS_PARMSW MCI_DGV_STATUS_PARMSW(유니코드) 구조에는 디지털 비디오 디바이스에 대한 MCI_STATUS 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_STEP_PARMS MCI_DGV_STEP_PARMS 구조에는 디지털 비디오 디바이스에 대한 MCI_STEP 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_UPDATE_PARMS MCI_DGV_UPDATE_PARMS 구조에는 MCI_UPDATE 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_WINDOW_PARMSA ANSI(MCI_DGV_WINDOW_PARMSA) 구조에는 디지털 비디오 디바이스에 대한 MCI_WINDOW 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_WINDOW_PARMSW MCI_DGV_WINDOW_PARMSW(유니코드) 구조에는 디지털 비디오 디바이스에 대한 MCI_WINDOW 명령에 대한 매개 변수가 포함되어 있습니다.
MIDIEVENT MIDIEVENT 구조체는 스트림 버퍼의 MIDI 이벤트를 설명합니다.

MIDIHDR MIDIHDR 구조체는 MIDI 시스템 배타적 또는 스트림 버퍼를 식별하는 데 사용되는 헤더를 정의합니다.
MIDIINCAPS MIDIINCAPS 구조는 MIDI 입력 디바이스의 기능을 설명합니다. (MIDIINCAPS)
MIDIINCAPSA MIDIINCAPS 구조는 MIDI 입력 디바이스의 기능을 설명합니다. (MIDIINCAPSA)
MIDIINCAPSW MIDIINCAPS 구조는 MIDI 입력 디바이스의 기능을 설명합니다. (MIDIINCAPSW)
MIDIOUTCAPS MIDIOUTCAPS 구조는 MIDI 출력 디바이스의 기능을 설명합니다. (MIDIOUTCAPS)
MIDIOUTCAPSA MIDIOUTCAPS 구조는 MIDI 출력 디바이스의 기능을 설명합니다. (MIDIOUTCAPSA)
MIDIOUTCAPSW MIDIOUTCAPS 구조는 MIDI 출력 디바이스의 기능을 설명합니다. (MIDIOUTCAPSW)
MIDIPROPTEMPO MIDIPROPTEMPO 구조체에는 스트림에 대한 tempo 속성이 포함됩니다.
MIDIPROPTIMEDIV MIDIPROPTIMEDIV 구조체에는 스트림의 시간 나누기 속성이 포함됩니다.
MIDISTRMBUFFVER MIDISTRMBUFFVER 구조체에는 MEVT_VERSION 형식의 긴 MIDI 이벤트에 대한 버전 정보가 포함되어 있습니다.
MIXERCAPS MIXERCAPS 구조는 믹서 디바이스의 기능을 설명합니다. (MIXERCAPS)
MIXERCAPSA MIXERCAPS 구조는 믹서 디바이스의 기능을 설명합니다. (MIXERCAPSA)

MIXERCAPSW MIXERCAPS 구조는 믹서 디바이스의 기능을 설명합니다. (MIXERCAPSW)
MIXERCONTROL MIXERCONTROL 구조체는 오디오 라인에 대한 단일 컨트롤의 상태 및 메트릭을 설명합니다. (MIXERCONTROL, *PMIXERCONTROL, FAR *PMIXERCONTROL)
MIXERCONTROLA MIXERCONTROL 구조체는 오디오 라인에 대한 단일 컨트롤의 상태 및 메트릭을 설명합니다. (mixercontrola)
MIXERCONTROLDETAILS MIXERCONTROLDETAILS 구조체는 오디오 믹서 컨트롤의 상태 정보를 검색하거나 설정하는 컨트롤 세부 구조체를 나타냅니다. (MIXERCONTROLDETAILS)
MIXERCONTROLDETAILS_BOOLEAN MIXERCONTROLDETAILS_BOOLEAN 구조체는 오디오 믹서 컨트롤에 대한 부울 컨트롤 속성을 검색하고 설정합니다.
MIXERCONTROLDETAILS_LISTTEXT MIXERCONTROLDETAILS_LISTTEXT 구조체는 다중 항목 컨트롤에 대한 목록 텍스트, 레이블 텍스트 및/또는 대역 범위 정보를 검색합니다.
MIXERCONTROLDETAILS_LISTTEXTA MIXERCONTROLDETAILS 구조체는 오디오 믹서 컨트롤의 상태 정보를 검색하거나 설정하는 컨트롤 세부 구조체를 나타냅니다. (mixercontroldetails_listtexta)
MIXERCONTROLDETAILS_LISTTEXTW MIXERCONTROLDETAILS 구조체는 오디오 믹서 컨트롤의 상태 정보를 검색하거나 설정하는 컨트롤 세부 구조체를 나타냅니다. (mixercontroldetails_listtextw)
MIXERCONTROLDETAILS_SIGNED MIXERCONTROLDETAILS_SIGNED 구조체는 오디오 믹서 컨트롤에 대한 서명된 형식 컨트롤 속성을 검색하고 설정합니다.
MIXERCONTROLDETAILS_UNSIGNED MIXERCONTROLDETAILS_UNSIGNED 구조체는 오디오 믹서 컨트롤에 대한 서명되지 않은 형식 컨트롤 속성을 검색하고 설정합니다.

MIXERCONTROLW MIXERCONTROL 구조체는 오디오 라인에 대한 단일 컨트롤의 상태 및 메트릭을 설명합니다. (mixercontrolw)
MIXERLINE MIXERLINE 구조체는 오디오 라인의 상태 및 메트릭을 설명합니다. (MIXERLINE)
MIXERLINEA MIXERLINE 구조체는 오디오 라인의 상태 및 메트릭을 설명합니다. (MIXERLINEA)
MIXERLINECONTROLS MIXERLINECONTROLS 구조체에는 오디오 라인의 컨트롤에 대한 정보가 포함되어 있습니다. (MIXERLINECONTROLS)
MIXERLINECONTROLSA MIXERLINECONTROLS 구조체에는 오디오 라인의 컨트롤에 대한 정보가 포함되어 있습니다. (MIXERLINECONTROLSA)
MIXERLINECONTROLSW MIXERLINECONTROLS 구조체에는 오디오 라인의 컨트롤에 대한 정보가 포함되어 있습니다. (MIXERLINECONTROLSW)
MIXERLINEW MIXERLINE 구조체는 오디오 라인의 상태 및 메트릭을 설명합니다. (MIXERLINEW)
MMCKINFO MMCKINFO 구조체에는 RIFF 파일의 청크에 대한 정보가 포함되어 있습니다.
PCMWAVEFORMAT PCMWAVEFORMAT 구조체는 PCM 파형 오디오 데이터의 데이터 형식을 설명합니다. 이 구조체는 WAVEFORMATEX 구조체로 대체됩니다.
PCMWAVEFORMAT PCMWAVEFORMAT 구조체는 PCM 파형 오디오 데이터의 데이터 형식을 설명합니다. 이 구조체는 WAVEFORMATEX 구조체로 대체되었습니다.
TIMECAPS TIMECAPS 구조에는 타이머의 해상도에 대한 정보가 포함되어 있습니다.

VIDEOHDR VIDEOHDR 구조체는 capVideoStreamCallback 함수에서 사용됩니다.
WAVEFILTER WAVEFILTER 구조체는 파형 오디오 데이터에 대한 필터를 정의합니다.
WAVEFORMAT WAVEFORMAT 구조체는 파형 오디오 데이터의 형식을 설명합니다. 이 구조체는 WAVEFORMATEX 구조체로 대체됩니다.
WAVEFORMAT WAVEFORMAT 구조체는 파형 오디오 데이터의 형식을 설명합니다. 이 구조체는 WAVEFORMATEX 구조체로 대체되었습니다.
WAVEFORMATEX WAVEFORMATEX 구조체는 파형 오디오 데이터의 형식을 정의합니다.
WAVEFORMATEXTENSIBLE WAVEFORMATEXTENSIBLE 구조체는 WAVEFORMATEX에서 허용하는 것보다 두 개 이상의 채널 이상의 샘플 해상도를 갖는 형식에 대해 파형 오디오 데이터의 형식을 정의합니다.
WAVEHDR WAVEHDR 구조체는 파형 오디오 버퍼를 식별하는 데 사용되는 헤더를 정의합니다.
WAVEINCAPS WAVEINCAPS 구조는 파형 오디오 입력 디바이스의 기능을 설명합니다. (WAVEINCAPS)
WAVEINCAPSA WAVEINCAPS 구조는 파형 오디오 입력 디바이스의 기능을 설명합니다. (WAVEINCAPSA)
WAVEINCAPSW WAVEINCAPS 구조는 파형 오디오 입력 디바이스의 기능을 설명합니다. (WAVEINCAPSW)
WAVEOUTCAPS WAVEOUTCAPS 구조는 파형 오디오 출력 디바이스의 기능을 설명합니다. (WAVEOUTCAPS)
WAVEOUTCAPSA

WAVEOUTCAPS 구조체는 파형 오디오 출력 디바이스의 기능을 설명합니다. (WAVEOUTCAPSA)
WAVEOUTCAPSW WAVEOUTCAPS 구조체는 파형 오디오 출력 디바이스의 기능을 설명합니다. (WAVEOUTCAPSW)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Windows 네트워크 가상화

아티클2023. 03. 09.

Windows 네트워크 가상화 기술 개요입니다.

Windows 네트워크 가상화를 개발하려면 다음 헤더가 필요합니다.

- wnvapi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 네트워크 가상화

열거형

WNV_NOTIFICATION_TYPE 지정된 WNV(Windows 네트워크 가상화) 알림의 유형을 지정합니다.
WNV_OBJECT_TYPE WNV 알림 유형이 WnvObjectChangeType인 경우 지정된 WNV(Windows Network Virtualization) 알림의 개체 유형을 지정합니다.

Functions

WnvOpen WNV 알림을 요청하고 수신하는 데 사용할 Windows WNV(네트워크 가상화) 드라이버 개체에 대한 핸들을 제공합니다.
WnvRequestNotification 특정 유형의 이벤트가 발생할 때마다 Windows WNV(네트워크 가상화) 드라이버에서 알림을 요청합니다.

구조체

WNV_IP_ADDRESS IP 주소 개체를 정의합니다.
WNV_NOTIFICATION_PARAM WnvRequestNotification 함수 호출에서 버전, 알림 유형 및 버퍼 위치를 지정합니다.
WNV_OBJECT_CHANGE_PARAM Windows WNV(네트워크 가상화) 드라이버가 WnvObjectChangeType 유형의 알림을 생성하도록 하는 이벤트의 매개 변수를 지정합니다.
WNV_OBJECT_HEADER WnvRequestNotification 함수에 전달되는 WNV_NOTIFICATION_PARAM 구조체의 주 버전, 부 버전 및 버퍼 크기를 지정합니다.
WNV_POLICY_MISMATCH_PARAM Windows WNV(네트워크 가상화) 드라이버가 WnvPolicyMismatchType 알림을 생성하도록 하는 이벤트(일반적으로 들어오는 패킷)의 매개 변수를 지정합니다.
WNV_PROVIDER_ADDRESS_CHANGE_PARAM 공급자 주소의 DAD(중복 주소 검색) 상태 변경을 지정하여 Windows WNV(네트워크 가상화) 드라이버에서 이 구조가 포함된 WnvProviderAddressType 개체 형식을 지정하는 WnvObjectChangeType 알림을 생성합니다.
WNV_REDIRECT_PARAM Windows WNV(네트워크 가상화) 드라이버가 WnvRedirectType 알림을 생성하도록 하는 이벤트의 매개 변수(들어오는 인터넷 제어 메시지 프로토콜 리디렉션 패킷 수신)를 지정합니다.

WNet(Windows 네트워킹)

WNet(Windows 네트워킹) 기술 개요입니다.

WNet(Windows 네트워킹) 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- WNet(Windows 네트워킹)

Functions

테이블 확장

MultinetGetConnectionPerformanceA 네트워크 리소스에 액세스하는 데 사용되는 연결의 예상 성능에 대한 정보를 반환합니다. (ANSI)
MultinetGetConnectionPerformanceW 네트워크 리소스에 액세스하는 데 사용되는 연결의 예상 성능에 대한 정보를 반환합니다. (유니코드)
WNetAddConnection2A WNetAddConnection2 함수는 네트워크 리소스에 연결하고 로컬 디바이스를 네트워크 리소스로 리디렉션할 수 있습니다. (ANSI)
WNetAddConnection2W WNetAddConnection2 함수는 네트워크 리소스에 연결하고 로컬 디바이스를 네트워크 리소스로 리디렉션할 수 있습니다. (유니코드)
WNetAddConnection3A WNetAddConnection3 함수는 네트워크 리소스에 연결합니다. 이 함수는 로컬 디바이스를 네트워크 리소스로 리디렉션할 수 있습니다. (ANSI)
WNetAddConnection3W WNetAddConnection3 함수는 네트워크 리소스에 연결합니다. 이 함수는 로컬 디바이스를 네트워크 리소스로 리디렉션할 수 있습니다. (유니코드)
WNetAddConnection4A WNetAddConnection4 함수는 네트워크 리소스에 연결합니다. 이 함수는 로컬 디바이스를 네트워크 리소스로 리디렉션할 수 있습니다. (ANSI)

WNetAddConnection4W WNetAddConnection4 함수는 네트워크 리소스에 연결합니다. 이 함수는 로컬 디바이스를 네트워크 리소스로 리디렉션할 수 있습니다. (유니코드)
WNetAddConnectionA WNetAddConnection 함수를 사용하면 호출하는 애플리케이션이 로컬 디바이스를 네트워크 리소스에 연결할 수 있습니다. 성공적인 연결은 영구적입니다. 즉, 시스템이 후속 로그온 작업 중에 연결을 자동으로 복원합니다. (ANSI)
WNetAddConnectionW WNetAddConnection 함수를 사용하면 호출하는 애플리케이션이 로컬 디바이스를 네트워크 리소스에 연결할 수 있습니다. 성공적인 연결은 영구적입니다. 즉, 시스템이 후속 로그온 작업 중에 연결을 자동으로 복원합니다. (유니코드)
WNetCancelConnection2A WNetCancelConnection2 함수는 기존 네트워크 연결을 취소합니다. 함수를 호출하여 현재 연결되지 않은 기억된 네트워크 연결을 제거할 수도 있습니다. (ANSI)
WNetCancelConnection2W WNetCancelConnection2 함수는 기존 네트워크 연결을 취소합니다. 함수를 호출하여 현재 연결되지 않은 기억된 네트워크 연결을 제거할 수도 있습니다. (유니코드)
WNetCancelConnectionA WNetCancelConnection 함수는 기존 네트워크 연결을 취소합니다. (ANSI)
WNetCancelConnectionW WNetCancelConnection 함수는 기존 네트워크 연결을 취소합니다. (유니코드)
WNetCloseEnum WNetCloseEnum 함수는 WNetOpenEnum 함수 호출로 시작된 네트워크 리소스 열거형을 종료합니다.
WNetConnectionDialog WNetConnectionDialog 함수는 네트워크 리소스에 연결하기 위한 일반 검색 대화 상자를 시작합니다. 이 함수에는 대화 상자의 소유자 창에 대한 핸들이 필요합니다.
WNetConnectionDialog1A WNetConnectionDialog1 함수는 네트워크 리소스에 연결하기 위한 일반 검색 대화 상자를 표시합니다. 이 함수는 대화 상자 매개 변수를 설정하기 위해 CONNECTDLGSTRUCT가 필요합니다. (ANSI)

WNetConnectionDialog1W WNetConnectionDialog1 함수는 네트워크 리소스에 연결하기 위한 일반 검색 대화 상자를 표시합니다. 이 함수는 대화 상자 매개 변수를 설정하기 위해 CONNECTDLGSTRUCT가 필요합니다. (유니코드)
WNetDisconnectDialog WNetDisconnectDialog 함수는 네트워크 리소스에서 연결을 끊기 위한 일반 검색 대화 상자를 시작합니다. 이 함수에는 대화 상자의 소유자 창에 대한 핸들이 필요합니다.
WNetDisconnectDialog1A WNetDisconnectDialog1 함수는 네트워크 리소스의 연결을 끊으려고 시도합니다. (ANSI)
WNetDisconnectDialog1W WNetDisconnectDialog1 함수는 네트워크 리소스의 연결을 끊으려고 시도합니다. (유니코드)
WNetEnumResourceA WNetEnumResource 함수는 WNetOpenEnum 함수에 대한 호출로 시작된 네트워크 리소스의 열거를 계속합니다. (ANSI)
WNetEnumResourceW WNetEnumResource 함수는 WNetOpenEnum 함수에 대한 호출로 시작된 네트워크 리소스의 열거를 계속합니다. (유니코드)
WNetGetConnectionA WNetGetConnection 함수는 로컬 디바이스와 연결된 네트워크 리소스의 이름을 검색합니다. (ANSI)
WNetGetConnectionW WNetGetConnection 함수는 로컬 디바이스와 연결된 네트워크 리소스의 이름을 검색합니다. (유니코드)
WNetGetLastErrorA WNetGetLastError 함수는 WNet 함수에서 설정한 최신 확장 오류 코드를 검색합니다. 네트워크 공급자가 이 오류 코드를 보고했습니다. 일반적으로 SDK 헤더 파일 WinError.h에 포함된 오류 중 하나가 아닙니다. (ANSI)
WNetGetLastErrorW WNetGetLastError 함수는 WNet 함수에서 설정한 최신 확장 오류 코드를 검색합니다. 네트워크 공급자가 이 오류 코드를 보고했습니다. 일반적으로 SDK 헤더 파일 WinError.h에 포함된 오류 중 하나가 아닙니다. (유니코드)

WNetGetNetworkInformationA WNetGetNetworkInformation 함수는 이전 네트워크 열거형에서 이름이 반환된 특정 네트워크 공급자에 대한 확장 정보를 반환합니다. (ANSI)
WNetGetNetworkInformationW WNetGetNetworkInformation 함수는 이전 네트워크 열거형에서 이름이 반환된 특정 네트워크 공급자에 대한 확장 정보를 반환합니다. (유니코드)
WNetGetProviderNameA WNetGetProviderName 함수는 특정 유형의 네트워크에 대한 공급자 이름을 가져옵니다. (ANSI)
WNetGetProviderNameW WNetGetProviderName 함수는 특정 유형의 네트워크에 대한 공급자 이름을 가져옵니다. (유니코드)
WNetGetResourceInformationA 네트워크 리소스에 대한 원격 경로를 제공하는 경우 WNetGetResourceInformation 함수는 리소스를 소유하는 네트워크 공급자를 식별하고 리소스 유형에 대한 정보를 가져옵니다. (ANSI)
WNetGetResourceInformationW 네트워크 리소스에 대한 원격 경로를 제공하는 경우 WNetGetResourceInformation 함수는 리소스를 소유하는 네트워크 공급자를 식별하고 리소스 유형에 대한 정보를 가져옵니다. (유니코드)
WNetGetResourceParentA WNetGetResourceParent 함수는 네트워크 찾아보기 계층 구조에서 네트워크 리소스의 부모를 반환합니다. 검색은 지정된 네트워크 리소스의 위치에서 시작됩니다. (ANSI)
WNetGetResourceParentW WNetGetResourceParent 함수는 네트워크 찾아보기 계층 구조에서 네트워크 리소스의 부모를 반환합니다. 검색은 지정된 네트워크 리소스의 위치에서 시작됩니다. (유니코드)
WNetGetUniversalNameA WNetGetUniversalName 함수는 네트워크 리소스에 대한 드라이브 기반 경로를 사용하고 보다 보편적인 형식의 이름을 포함하는 정보 구조를 반환합니다. (ANSI)
WNetGetUniversalNameW WNetGetUniversalName 함수는 네트워크 리소스에 대한 드라이브 기반 경로를 사용하고 보다 보편적인 형식의 이름을 포함하는 정보 구조를 반환합니다. (유니코드)

WNetGetUserA WNetGetUser 함수는 현재 기본 사용자 이름 또는 네트워크 연결을 설정하는 데 사용되는 사용자 이름을 검색합니다. (ANSI)
WNetGetUserW WNetGetUser 함수는 현재 기본 사용자 이름 또는 네트워크 연결을 설정하는 데 사용되는 사용자 이름을 검색합니다. (유니코드)
WNetOpenEnumA WNetOpenEnum 함수는 네트워크 리소스 또는 기존 연결의 열거형을 시작합니다. WNetEnumResource 함수를 호출하여 열거형을 계속할 수 있습니다. (ANSI)
WNetOpenEnumW WNetOpenEnum 함수는 네트워크 리소스 또는 기존 연결의 열거형을 시작합니다. WNetEnumResource 함수를 호출하여 열거형을 계속할 수 있습니다. (유니코드)
WNetRestoreConnectionW WNetRestoreConnectionW 함수는 네트워크 리소스에 대한 연결을 복원합니다. 이 함수는 필요한 경우 사용자에게 이름과 암호를 묻는 메시지를 표시합니다.
WNetUseConnectionA WNetUseConnection 함수는 네트워크 리소스에 연결합니다. 이 함수는 로컬 디바이스를 네트워크 리소스로 리디렉션할 수 있습니다. (ANSI)
WNetUseConnectionW WNetUseConnection 함수는 네트워크 리소스에 연결합니다. 이 함수는 로컬 디바이스를 네트워크 리소스로 리디렉션할 수 있습니다. (유니코드)

구조

테이블 확장

CONNECTDLGSTRUCTA WNetConnectionDialog1 함수에서 검색 대화 상자 매개 변수를 설정하는 데 사용됩니다. (ANSI)
CONNECTDLGSTRUCTW

WNetConnectionDialog1 함수에서 검색 대화 상자 매개 변수를 설정하는 데 사용됩니다. (유니코드)
DISCDLGSTRUCTA WNetDisconnectDialog1 함수에 사용됩니다. 구조체에는 연결 끊기 시도에 필요한 정보가 포함되어 있습니다. (ANSI)
DISCDLGSTRUCTW WNetDisconnectDialog1 함수에 사용됩니다. 구조체에는 연결 끊기 시도에 필요한 정보가 포함되어 있습니다. (유니코드)
NETINFOSTRUCT WNetGetNetworkInformation 함수에서 반환된 네트워크 공급자를 설명하는 정보를 포함합니다.

Last updated on 2025. 08. 16.

Windows 속성 시스템

아티클2023. 03. 16.

Windows 속성 시스템 기술 개요입니다.

Windows 속성 시스템을 개발하려면 다음 헤더가 필요합니다.

- propidl.h
- propsys.h
- propvarutil.h
- shellapi.h
- shobjidl_core.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 속성 시스템

열거형

_PROPERTYUI_FLAGS 속성 기능을 지정합니다.
GETPROPERTYSTOREFLAGS IShellItem2::GetPropertyStore 또는 IPropertyStoreFactory::GetPropertyStore와 같이 속성 저장소를 만드는 메서드에 의해 검색되는 속성 저장소 개체를 수정하는 플래그를 나타냅니다.
PDOPSTATUS 작업 상태 플래그를 제공합니다.
PKA_FLAGS 속성 변경 배열 동작에 대해 설명합니다.
PLACEHOLDER_STATES 자리 표시자 파일에 사용할 수 있는 상태를 지정합니다. System.FilePlaceholderStatus(PKEY_FilePlaceholderStatus) 속성을 통해 이 값을 검색합니다.
PROPDESC_AGGREGATION_TYPE 여러 항목을 선택할 때 속성 값이 표시되는 방법을 설명합니다.

PROPDESC_COLUMNINDEX_TYPE 속성을 인덱싱할 수 있는지 여부 또는 방법을 나타냅니다.
PROPDESC_CONDITION_TYPE Windows Vista의 쿼리 작성기 UI에서 속성을 표시할 때 사용할 조건 유형에 대해 설명하지만 Windows 7 이상에서는 사용할 수 없습니다.
PROPDESC_ENUMFILTER 반환되는 속성 설명의 필터링된 목록을 설명합니다.
PROPDESC_FORMAT_FLAGS PSFormatForDisplay와 같은 속성 설명 도우미 함수에서 속성 문자열의 형식을 나타내는 데 사용됩니다.
PROPDESC_RELATIVEDESCRIPTION_TYPE displayInfo 요소의 relativeDescriptionType 특성에 의해 결정되는 속성 설명의 상대 설명 형식에 대해 설명합니다.
PROPDESC_SEARCHINFO_FLAGS Windows Search에서 속성을 인덱싱할지 여부와 방법을 결정합니다.
PROPDESC_TYPE_FLAGS 속성의 .propdesc 파일에서 typeInfo 요소의 특성을 설명합니다.
PROPDESC_VIEW_FLAGS 이러한 플래그는 속성 설명 목록 문자열의 속성을 설명합니다.
PROPVAR_COMPARE_UNIT 이러한 플래그는 특정 PROPVARIANT 구조 비교와 연결됩니다.
PSC_STATE 속성의 상태를 지정합니다. 메모리 내 속성 저장소 캐시를 호스팅하는 코드에 의해 수동으로 설정됩니다.

SYNC_ENGINE_STATE_FLAGS 파일 인덱서에서 속성 저장소의 PKEY_StorageProviderStatus 값에 내부 엔진 상태를 노출하기 위해 동기화 엔진에서 사용하는 값을 지정합니다. 속성을 업데이트하려면 먼저 GPS_EXTRINSICPROPERTIES 플래그를 사용하여 IShellItem2::GetPropertyStore를 호출합니다. 다음으로 반환된 개체의 IPropertyStore::SetValue 메서드를 호출하여 PKEY_StorageProviderStatus 키를 지정하여 이러한 SYNC_ENGINE_STATE_FLAGS 사용하여 속성의 비트 마스크 값을 설정합니다.
SYNC_TRANSFER_STATUS System.SyncTransferStatus 속성에 사용할 수 있는 상태 값을 지정합니다.

함수

Append 배열의 끝에 변경 작업을 삽입합니다.
AppendOrReplace 제공된 변경 내용과 동일한 속성 키에 영향을 주는 변경 내용의 첫 번째 발생을 바꿉니다. 속성 키가 아직 배열에 없는 경우 이 메서드는 배열의 끝에 변경 사항을 추가합니다.
ApplyToPropVariant 속성 값에 변경 사항을 적용합니다.
ClearPropVariantArray 배열에 저장된 PROPVARIANT 구조체 배열에서 사용하는 메모리 및 참조를 해제합니다.
ClearVariantArray 배열에 저장된 VARIANT 구조체의 배열에서 사용하는 메모리 및 참조를 해제합니다.
CoerceToCanonicalValue 속성 설명에 따라 값을 정식 값으로 강제 변환합니다.

EnumeratePropertyDescriptions	시스템에서 속성 설명의 전체 또는 부분 목록을 가져오기 위해 IPropertyDescriptionList를 구현하는 하위 시스템 개체의 인스턴스를 가져옵니다.
FindMatchingIndex	지정된 속성 값을 목록의 열거된 값과 비교하고 일치하는 인덱스를 반환합니다.
FormatForDisplay	속성 값의 형식이 지정된 유니코드 문자열 표현을 가져옵니다. (IPropertyDescription.FormatForDisplay)
FormatForDisplay	속성 값의 형식이 지정된 유니코드 문자열 표현을 가져옵니다. (IPropertySystem.FormatForDisplay)
FormatForDisplay	개발자는 대신 IPropertyDescription을 사용해야 합니다. 속성 값의 형식이 지정된 유니코드 문자열 표현을 가져옵니다.
FormatForDisplayAlloc	할당된 메모리 버퍼에 대한 속성 값의 문자열 표현을 가져옵니다.
FreePropVariantArray	PROPVARIANT 구조의 배열에서 사용되는 메모리 및 참조를 해제합니다.
GetAdditionalSortByAliases	추가 정렬 열 값을 포함하는 IPropertyDescriptionList 인터페이스에 대한 포인터의 주소를 가져옵니다.
GetAggregationType	UI에서 여러 항목을 선택할 때 속성 값이 표시되는 방식을 설명하는 값을 가져옵니다.

Getat	지정된 배열 인덱스에서 변경 작업을 가져옵니다.
Getat	속성 설명 목록의 지정된 인덱스에서 속성 설명을 가져옵니다.
Getat	목록의 지정된 인덱스에서 IPropertyEnumType 개체를 가져옵니다.
GetCanonicalName	지역화된 이름에 관계없이 속성이 시스템에 알려진 대/소문자를 구분하는 이름을 가져옵니다.
GetColumnIndexType	현재 속성의 인덱싱 방법을 결정합니다.
GetColumnState	이 플래그를 사용하는 인터페이스 또는 API에서 속성을 처리하는 방법을 설명하는 열 상태 플래그를 가져옵니다.
GetConditionAt	지원되지 않습니다. 지정된 인덱스에서 조건을 가져옵니다.
GetConditionType	쿼리 작성기 UI에 속성을 표시할 때 사용할 조건 유형 및 기본 조건 작업을 가져옵니다. 이 속성에 대해 표시되는 조건자 조건 목록(예: 같음, 보다 작음 및 포함)에 영향을 줍니다.
GetCount	배열의 변경 작업 수를 가져옵니다.
GetCount	속성 목록에 포함된 속성 수를 가져옵니다.

GetCount 목록의 요소 수를 가져옵니다.
GetDefaultColumnWidth 목록 보기에서 속성의 기본 열 너비를 가져옵니다.
GetDefaultWidth 개발자는 대신 IPropertyDescription을 사용해야 합니다. 속성 설명의 너비를 가져옵니다.
GetDisplayName 모든 UI에 표시된 대로 속성의 표시 이름을 가져옵니다.
GetDisplayName 개발자는 대신 IPropertyDescription을 사용해야 합니다. 사용자에게 표시하기에 적합한 속성의 이름을 지정하는 문자열을 가져옵니다.
GetDisplayText 열거형 정보 구조에서 표시 텍스트를 가져옵니다.
GetDisplayType 속성을 표시하는 데 사용되는 현재 데이터 형식을 가져 옵니다.
GetEditInvitation 다양한 대화 상자에서 호스트되는 편집 컨트롤에 사용 되는 텍스트를 가져옵니다.
GetEnumType 열거형 정보 구조에서 열거형 형식을 가져옵니다.
GetEnumTypeList 속성의 가능한 값을 열거하는 데 사용할 수 있는 IPropertyEnumTypeList의 인스턴스를 가져옵니다.

GetFlags 개발자는 대신 IPropertyDescription을 사용해야 합니다. 지정된 속성에 대한 속성 기능 플래그를 가져옵니다.
GetGroupingRange 뷰가 속성별로 그룹화될 때 사용할 그룹화 메서드를 가 져오고 그룹화 형식을 검색합니다.
GetHelpInfo 개발자는 대신 IPropertyDescription을 사용해야 합니다. (IPropertyUI.GetHelpInfo)
GetImageReference 속성 열거형과 연결된 이미지 참조를 검색합니다.
GetImageReferenceForValue 속성 값과 연결된 이미지 참조를 가져옵니다.
GetMaxSize 속성 스키마의 searchInfo 요소에서 최대 크기 값을 가 져옵니다.
GetProjectionString 항목의 정식 이름을 포함하는 문자열에 대한 포인터를 반환합니다.
GetPropertyDescription 지정된 PROPERTYKEY에 대한 속성 설명을 얻기 위해 IPropertyDescription을 구현하는 하위 시스템 개체의 인스턴스를 가져옵니다.
GetPropertyDescription 개발자는 대신 IPropertyDescription을 사용해야 합니다. 지정된 속성의 속성 설명을 가져옵니다.
GetPropertyDescriptionByName 지정된 정식 이름에 대한 속성 설명을 얻기 위해 IPropertyDescription을 구현하는 하위 시스템 개체의 인스턴스를 가져옵니다.

GetPropertyDescriptionListFromString 제공된 문자열에 따라 정렬된 속성 설명 컬렉션을 가져오기 위해 IPropertyDescriptionList를 구현하는 하위 시스템 개체의 인스턴스를 가져옵니다.
GetPropertyKey 속성의 고유 식별자 역할을 하는 구조를 가져옵니다.
GetPropertyStore 제공된 플래그에 해당하는 IPropertyStore 개체를 가져옵니다.
GetPropertyStoreForKeys 속성 키 집합이 지정된 IPropertyStore 개체를 가져옵니다. 이렇게 하면 IPropertyStoreFactory::GetPropertyStore 호출에 비해 IPropertyStore 개체를 가져오는 대체 방법이 더 빠를 수 있습니다.
GetPropertyType 속성의 변형 형식을 가져옵니다.
GetRangeMinValue 열거형 정보 구조에서 최소값을 가져옵니다.
GetRangeSetValue 열거형 정보 구조에서 설정된 값을 가져옵니다.
GetRelatedProperty 관련 속성을 나타내는 IPropertyDescription 개체를 검색합니다.
GetRelativeDescription 속성 설명에 지정된 방식으로 두 속성 값을 비교합니다. 두 속성이 비교되는 방법을 설명하는 두 개의 표시 문자열을 반환합니다.
GetRelativeDescriptionType 속성 설명에 대한 상대 설명 형식을 가져옵니다.

GetSearchInfoFlags 속성과 연결된 PROPDESC_SEARCHINFO_FLAGS 가져옵니다.
GetSortByAlias 기본 정렬 열을 포함하는 IPropertyDescription 인터페이스에 대한 포인터의 주소를 가져옵니다.
GetSortDescription 정렬 제품의 특정 표현을 나타내는 속성의 현재 정렬 설명 플래그를 가져옵니다.
GetSortDescriptionLabel 현재 정렬 순서를 설명하는 지역화된 표시 문자열을 가져옵니다.
GetState 지정된 속성 키의 상태를 가져옵니다.
GetTypeFlags 속성의 사용 및 기능을 설명하는 플래그 집합을 가져옵니다.
GetValue 열거형 정보 구조에서 값을 가져옵니다.
GetValueAndState 속성 키의 값 및 상태 데이터를 가져옵니다.
GetViewFlags 속성의 뷰를 제어하는 현재 플래그 집합을 가져옵니다.
InitPropVariantFromBoolean 지정된 부울 값을 사용하여 지정된 PROPVARIANT 구조를 VT_BOOL 초기화합니다.

InitPropVariantFromBooleanVector 지정된 부울 벡터에서 PROPVARIANT 구조를 초기화합니다.	
InitPropVariantFromBuffer 버퍼의 내용을 사용하여 PROPVARIANT 구조를 초기화합니다.	
InitPropVariantFromCLSID CLSID(클래스 식별자)를 기반으로 PROPVARIANT 구조를 초기화합니다.	
InitPropVariantFromDouble 지정된 double 값을 기반으로 PROPVARIANT 구조를 초기화합니다.	
InitPropVariantFromDoubleVector 지정된 double 값 벡터를 기반으로 PROPVARIANT 구조를 초기화합니다.	
InitPropVariantFromFileTime FILETIME 구조에 저장된 정보를 기반으로 PROPVARIANT 구조를 초기화합니다.	
InitPropVariantFromFileTimeVector FILETIME 값의 지정된 벡터에서 PROPVARIANT 구조를 초기화합니다.	
InitPropVariantFromGUIDAsBuffer GUID를 기반으로 PROPVARIANT 구조를 초기화합니다. 구조체가 VT_VECTOR	VT_UI1.
InitPropVariantFromGUIDAsString GUID를 기반으로 PROPVARIANT 구조를 초기화합니다. 구조체는 VT_LPWSTR 초기화됩니다.	
InitPropVariantFromInt16 16비트 정수 값을 기반으로 PROPVARIANT 구조를 초기화합니다.	

InitPropVariantFromInt16Vector	지정된 16비트 정수 값의 벡터를 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromInt32	32비트 정수 값을 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromInt32Vector	32비트 정수 값의 벡터를 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromInt64	지정된 Int64 값을 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromInt64Vector	Int64 값의 벡터를 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromPropVariantVectorElem	지정된 PROPVARIANT 벡터 요소를 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromResource	실행 파일에 포함된 문자열 리소스를 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromString	지정된 문자열을 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromStringAsVector	지정된 문자열에서 PROPVARIANT 구조를 초기화합니다. 문자열은 세미콜론으로 구분된 목록으로 구문 분석됩니다(예::_"A; B; C").

InitPropVariantFromStringVector	지정된 문자열 벡터에서 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromStrRet	STRRET 구조에 저장된 문자열을 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromUInt16	부호 없는 16비트 정수 값을 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromUInt16Vector	부호 없는 16비트 정수 값의 벡터를 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromUInt32	부호 없는 32비트 정수 값을 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromUInt32Vector	부호 없는 32비트 정수 값의 벡터를 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromUInt64	부호 없는 64비트 정수 값을 사용하여 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromUInt64Vector	부호 없는 64비트 정수의 벡터를 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantVectorFromPropVariant	다른 PROPVARIANT에 저장된 값을 사용하여 PROPVARIANT 구조에서 벡터 요소를 초기화합니다.
InitVariantFromBoolean	부울 값을 사용하여 VARIANT 구조를 초기화합니다.

InitVariantFromBooleanArray 부울 값 배열에서 VARIANT 구조를 초기화합니다.	
InitVariantFromBuffer 버퍼의 내용을 사용하여 VARIANT 구조를 초기화합니다.	
InitVariantFromDispatch IDispatch 개체의 인스턴스를 기반으로 VARIANT 구조를 초기화합니다.	
InitVariantFromDosDateTime Microsoft MS-DOS에서 사용하는 형식으로 지정된 날짜 및 시간을 사용하여 VARIANT 구조를 초기화합니다. 날짜 및 시간 값은 VARIANT에 날짜 및 시간을 저장하는 데 사용되는 형식으로 변환됩니다.	
InitVariantFromDouble DOUBLE 형식의 값을 사용하여 VARIANT 구조를 초기화합니다.	
InitVariantFromDoubleArray DOUBLE 형식의 값 배열을 사용하여 VARIANT 구조를 초기화합니다.	
InitVariantFromFileTime FILETIME 구조체의 내용을 사용하여 VARIANT 구조를 초기화합니다.	
InitVariantFromFileTimeArray FILETIME 구조의 배열을 사용하여 VARIANT 구조를 초기화합니다.	
InitVariantFromGUIDAsBuffer GUID를 기반으로 VARIANT 구조를 초기화합니다. 구조체가 VT_ARRAY 초기화됩니다.	VT_UI1.

InitVariantFromGUIDAsString GUID를 기반으로 VARIANT 구조를 초기화합니다. 구조체는 VT_BSTR 형식으로 초기화됩니다.
InitVariantFromInt16 16비트 정수 값을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromInt16Array 16비트 정수 값 배열을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromInt32 32비트 정수 값을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromInt32Array 32비트 정수 값 배열을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromInt64 64비트 정수 값을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromInt64Array 64비트 정수 값 배열을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromResource 실행 파일에 포함된 문자열 리소스를 기반으로 VARIANT 구조를 초기화합니다.
InitVariantFromString 문자열을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromStringArray 문자열 배열을 사용하여 VARIANT 구조를 초기화합니다.

InitVariantFromStrRet STRRET 구조에 저장된 문자열을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromUInt16 부호 없는 16비트 정수 값을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromUInt16Array 부호 없는 16비트 정수 값 배열을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromUInt32 부호 없는 32비트 정수 값을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromUInt32Array 부호 없는 32비트 정수 값 배열을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromUInt64 부호 없는 64비트 정수 값을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromUInt64Array 부호 없는 64비트 정수 값 배열을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromVariantArrayElem 다른 VARIANT 구조에 저장된 값을 사용하여 VARIANT 구조를 초기화합니다.
InsertAt 지정된 위치에 있는 배열에 변경 작업을 삽입합니다.
IsKeyInArray 특정 속성 키가 변경 배열에 있는지 여부를 지정합니다.

IsPropertyWritable 속성 처리기에서 사용자가 UI에서 특정 속성을 편집할 수 있는지 여부를 쿼리합니다.
IsPropVariantString 지정된 PROPVARIANT 구조체가 문자열 형식인지 여부를 지정합니다.
IsPropVariantVector PROPVARIANT 구조체에 벡터 형식이 있는지 여부를 지정합니다.
IsValueCanonical 속성 설명의 정의에 따라 속성이 정식인지 여부를 나타내는 값을 가져옵니다.
IsVariantArray 변형이 배열인지 여부를 지정합니다.
IsVariantString 변형이 문자열인지 여부를 지정합니다.
IsVarTypeFloat VARTYPE이 float인지 여부를 반환합니다.
IsVarTypeInteger VARTYPE이 정수인지 여부를 반환합니다.
IsVarTypeNumber VARTYPE이 숫자인지 여부를 지정합니다.
IsVarTypeSignedInteger VARTYPE이 부 서명된 정수인지 여부를 반환합니다.
IsVarTypeUnsignedInteger VARTYPE이 부호 없는 정수인지 여부를 반환합니다.

ParsePropertyName 개발자는 대신 IPropertyDescription을 사용해야 합니다. 지정된 속성 이름의 문자를 읽고 속성의 FMTID 및 PROPID를 식별합니다.
PifMgr_CloseProperties PifMgr_OpenProperties 사용하여 연 애플리케이션 속성을 닫습니다.
PifMgr_GetProperties .pif 파일에서 지정된 데이터 블록을 반환합니다.
PifMgr_OpenProperties Microsoft MS-DOS 애플리케이션과 연결된 .pif 파일을 열고 애플리케이션의 속성에 대한 핸들을 반환합니다.
PifMgr_SetProperties .pif 파일의 데이터 블록에 값을 할당합니다.
PropVariantChangeType PROPVARIANT 구조체로 저장된 값을 다른 변형 형식의 해당 값으로 강제 변환합니다.
PropVariantClear PROPVARIANT 구조를 지웁니다.
PropVariantCompare 기본 비교 단위 및 설정에 따라 두 PROPVARIANT 구조를 비교합니다.
PropVariantCompareEx 호출자가 지정된 비교 단위 및 플래그에 따라 두 PROPVARIANT 구조를 비교할 수 있도록 하여 PropVariantCompare를 확장합니다.
PropVariantCopy PROPVARIANT 구조체의 복사본을 만듭니다.

PropVariantGetBooleanElem VT_VECTOR VT_BOOL 형식의 PROPVARIANT 구조에서 단일 부울 요소를 추출합니다.	VT_BOOL 또는 VT_ARRAY	VT_BOOL.	
PropVariantGetDoubleElem VT_R8 형식의 PROPVARIANT 구조에서 단일 double 요소를 추출합니다VT_VECTOR	VT_R8 또는 VT_ARRAY	VT_R8.	
PropVariantGetElem 다른 PROPVARIANT 구조의 지정된 요소를 기반으로 PROPVARIANT 구조를 초기화합니다.			
PropVariantGetElementCount PROPVARIANT 구조체의 요소 수를 검색합니다.			
PropVariantGetFileTimeElem VT_FILETIME 형식의 PROPVARIANT 구조에서 단일 FILETIME 요소를 추출합니다VT_VECTOR	VT_FILETIME 또는 VT_ARRAY	VT_FILETIME.	
PropVariantGetInt16Elem VT_I2 형식의 PROPVARIANT 구조에서 단일 Int16 요소를 추출합니다VT_VECTOR	VT_I2 또는 VT_ARRAY	VT_I2.	
PropVariantGetInt32Elem VT_VECTOR VT_I4 형식의 PROPVARIANT에서 단일 Int32 요소를 추출합니다.	VT_I4 또는 VT_ARRAY	VT_I4.	
PropVariantGetInt64Elem VT_I8 형식의 PROPVARIANT 구조에서 단일 Int64 요소를 추출 VT_VECTOR합니다.	VT_I8 또는 VT_ARRAY	VT_I8.	
PropVariantGetStringElem VT_LPWSTR, VT_BSTR, VT_VECTOR 형식의 PROPVARIANT 구조에서 단일 유니코드 문자열 요소를 추출합니다.	VT_LPWSTR, VT_VECTOR	VT_BSTR 또는 VT_ARRAY	VT_BSTR.
PropVariantGetUInt16Elem VT_VECTOR VT_U12 형식의 PROPVARIANT 구조에서 부호 없는 단일 Int16 요소를 추출합니다.	VT_U12 또는 VT_ARRAY	VT_U12.	

PropVariantGetUInt32Elem VT_UI4 형식의 PROPVARIANT 구조에서 부호 없는 단일 Int32 요소를 추출합니다VT_VECTOR	VT_UI4 또는 VT_ARRAY	VT_UI4.
PropVariantGetUInt64Elem VT_UI8 형식의 PROPVARIANT 구조에서 부호 없는 단일 Int64 요소를 추출합니다VT_VECTOR	VT_UI8 또는 VT_ARRAY	VT_UI8.
PropVariantToBoolean PROPVARIANT 구조체의 부울 속성 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.		
PropVariantToBooleanVector PROPVARIANT 구조에서 부울 벡터를 추출합니다.		
PropVariantToBooleanVectorAlloc PROPVARIANT 구조에서 새로 할당된 부울 벡터로 데이터를 추출합니다.		
PropVariantToBooleanWithDefault PROPVARIANT 구조체의 부울 속성 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.		
PropVariantToBSTR PROPVARIANT 구조체의 BSTR 속성 값을 추출합니다.		
PropVariantToBuffer VT_VECTOR 형식의 PROPVARIANT 구조에서 버퍼 값을 추출합니다.	VT_UI1 또는 VT_ARRRAY	VT_UI1.
PropVariantToCLSID PROPVARIANT 구조체의 CLSID(클래스 식별자) 속성 값을 추출합니다.		
PropVariantToDouble PROPVARIANT 구조체에서 이중 값을 추출합니다.		

PropVariantToDoubleVector	PROPVARIANT 구조에서 double의 벡터를 추출합니다.
PropVariantToDoubleVectorAlloc	PROPVARIANT 구조에서 새로 할당된 이중 벡터로 데이터를 추출합니다.
PropVariantToDoubleWithDefault	PROPVARIANT 구조체의 이중 속성 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.
PropVariantToFileTime	PROPVARIANT 구조체에서 FILETIME 구조를 추출합니다.
PropVariantToFileTimeVector	PROPVARIANT 구조에서 FILETIME 벡터로 데이터를 추출합니다.
PropVariantToFileTimeVectorAlloc	PROPVARIANT 구조에서 새로 할당된 FILETIME 벡터로 데이터를 추출합니다.
PropVariantToGUID	PROPVARIANT 구조체에서 GUID 값을 추출합니다.
PropVariantToInt16	PROPVARIANT 구조체의 Int16 속성 값을 추출합니다.
PropVariantToInt16Vector	PROPVARIANT 구조에서 Int16 값의 벡터를 추출합니다.
PropVariantToInt16VectorAlloc	PROPVARIANT 구조에서 새로 할당된 Int16 벡터로 데이터를 추출합니다.

PropVariantToInt16WithDefault PROPVARIANT 구조체의 Int16 속성 값을 추출합니다. 현재 값이 없으면 지정된 기본값이 반환됩니다.
PropVariantToInt32 PROPVARIANT 구조체의 Int32 속성 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
PropVariantToInt32Vector PROPVARIANT 구조체에서 long 값의 벡터를 추출합니다.
PropVariantToInt32VectorAlloc PROPVARIANT 구조에서 새로 할당된 Int32 벡터로 데이터를 추출합니다.
PropVariantToInt32WithDefault PROPVARIANT 구조체에서 Int32 값을 추출합니다. 현재 값이 없으면 지정된 기본값이 반환됩니다.
PropVariantToInt64 PROPVARIANT 구조체에서 LONGLONG 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
PropVariantToInt64Vector PROPVARIANT 구조에서 Int64 벡터로 데이터를 추출합니다.
PropVariantToInt64VectorAlloc PROPVARIANT 구조에서 새로 할당된 LONGLONG 벡터로 데이터를 추출합니다.
PropVariantToInt64WithDefault PROPVARIANT 구조체의 Int64 속성 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.
PropVariantToString PROPVARIANT 구조체에서 문자열 값을 추출합니다.

PropVariantToStringAlloc	PROPVARIANT 구조체에서 문자열 속성 값을 추출합니다.
PropVariantToStringVector	PROPVARIANT 구조체에서 문자열 벡터를 추출합니다.
PropVariantToStringVectorAlloc	PROPVARIANT 구조에서 새로 할당된 벡터의 새로 할당된 문자열로 데이터를 추출합니다.
PropVariantToStringWithDefault	PROPVARIANT 구조체의 문자열 속성 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.
PropVariantToStrRet	PROPVARIANT 구조체에서 문자열을 추출하여 STRRET 구조에 배치합니다.
PropVariantToUInt16	PROPVARIANT 구조체에서 부호 없는 짧은 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
PropVariantToUInt16Vector	PROPVARIANT 구조체에서 부호 없는 짧은 벡터로 데이터를 추출합니다.
PropVariantToUInt16VectorAlloc	PROPVARIANT 구조에서 새로 할당된 부호 없는 짧은 벡터로 데이터를 추출합니다.
PropVariantToUInt16WithDefault	PROPVARIANT 구조체에서 부호 없는 짧은 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.
PropVariantToUInt32	PROPVARIANT 구조체에서 ULONG 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.

PropVariantToUInt32Vector PROPVARIANT 구조에서 ULONG 벡터로 데이터를 추출합니다.
PropVariantToUInt32VectorAlloc PROPVARIANT 구조에서 새로 할당된 ULONG 벡터로 데이터를 추출합니다.
PropVariantToUInt32WithDefault PROPVARIANT 구조체에서 ULONG 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.
PropVariantToUInt64 PROPVARIANT 구조체에서 UInt64 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
PropVariantToUInt64Vector PROPVARIANT 구조에서 ULONGLONG 벡터로 데이터를 추출합니다.
PropVariantToUInt64VectorAlloc PROPVARIANT 구조에서 새로 할당된 ULONGLONG 벡터로 데이터를 추출합니다.
PropVariantToUInt64WithDefault PROPVARIANT 구조에서 ULONGLONG 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.
PropVariantToVariant PROPVARIANT 구조체의 내용을 VARIANT 구조체로 변환합니다.
PropVariantToWinRTPropertyValue PROPVARIANT 구조에서 Windows 런타임 속성 값으로 데이터를 추출합니다.
PSCoerceToCanonicalValue 속성 설명에 따라 속성 값을 정식 값으로 변환합니다.

PSCreateAdapterFromPropertyStore IPropertyStore에서 어댑터를 만듭니다.
PSCreateDelayedMultiplexPropertyStore 여러 속성 저장소를 포함하는 읽기 전용 지연 바인딩 속성 저장소를 만듭니다.
PSCreateMemoryPropertyStore 메모리 내 속성 저장소를 만듭니다.
PSCreateMultiplexPropertyStore 여러 속성 저장소가 포함된 읽기 전용 속성 저장소를 만듭니다. 각 저장소는 IPropertyStore 또는 IPropertySetStorage를 지원해야 합니다.
PSCreatePropertyChangeArray IPropertyChange 개체 집합에 대한 컨테이너를 만듭니다. 이 컨테이너는 IFileOperation과 함께 사용하여 파일 집합에 속성 변경 내용 집합을 적용할 수 있습니다.
PSCreatePropertyStoreFromObject IPropertyStore 또는 IPropertySetStorage를 지원하는 개체의 IUnknown 인터페이스를 허용합니다. 개체가 IPropertySetStorage를 지원하는 경우 IPropertyStore를 지원하게 래핑됩니다.
PSCreatePropertyStoreFromPropertySetStorage IPropertyStore 인터페이스에서 IPropertySetStorage 인터페이스를 래핑합니다.
PSCreateSimplePropertyChange 간단한 속성 변경을 만듭니다.
PSEnumeratePropertyDescriptions 스키마 하위 시스템의 IPropertySystem::EnumeratePropertyDescriptions를 호출하는 래퍼 API입니다.

PSFormatForDisplay PROPVARIANT 구조에 저장된 속성 값의 형식이 지정된 유니코드 문자열 표현을 가져옵니다. 호출자는 출력 버퍼를 할당해야 합니다.
PSFormatForDisplayAlloc PROPVARIANT 구조에 저장된 속성 값의 형식이 지정된 유니코드 문자열 표현을 가져옵니다. 이 함수는 출력 문자열에 대한 메모리를 할당합니다.
PSFormatPropertyValue 속성 저장소에 저장된 속성 값의 형식이 지정된 유니코드 문자열 표현을 가져옵니다. 이 함수는 출력 문자열에 대한 메모리를 할당합니다.
PSGetImageReferenceForValue 지정된 속성에 대한 속성 설명 인터페이스의 인스턴스를 가져옵니다.
PSGetItemPropertyHandler 셸 항목에 대한 속성 처리기를 검색합니다. (PSGetItemPropertyHandler)
PSGetItemPropertyHandlerWithCreateObject 셸 항목에 대한 속성 처리기를 검색합니다. (PSGetItemPropertyHandlerWithCreateObject)
PSGetNamedPropertyFromPropertyStorage 속성 이름으로 직렬화된 속성 스토리지의 값을 가져옵니다.
PSGetNameFromPropertyKey PROPERTYKEY가 지정된 경우 속성의 정식 이름을 검색합니다.
PSGetPropertyDescription PROPERTYKEY 구조로 지정된 속성에 대한 속성 설명 인터페이스의 인스턴스를 가져옵니다.

PSGetPropertyDescriptionByName	지정된 속성 이름에 대한 속성 설명 인터페이스의 인스턴스를 가져옵니다.
PSGetPropertyDescriptionListFromString	지정된 속성 목록에 대한 속성 설명 목록 인터페이스의 인스턴스를 가져옵니다.
PSGetPropertyFromPropertyStorage	직렬화된 속성 스토리지에 저장된 속성의 값을 가져옵니다.
PSGetPropertyKeyFromName	정식 속성 이름의 속성 키를 가져옵니다.
PSGetPropertySystem	IPropertySystem을 구현하는 하위 시스템 개체의 인스턴스를 가져옵니다.
PSGetPropertyValue	속성 저장소에서 속성 값을 가져옵니다.
PSLookupPropertyHandlerCLSID	등록된 컴퓨터별 파일 속성 처리기의 CLSID(클래스 식별자)를 가져옵니다.
PSPropertyBag_Delete	속성 모음에서 속성을 삭제합니다.
PSPropertyBag_ReadBOOL	속성 모음에서 속성의 BOOL 데이터 값을 읽습니다.
PSPropertyBag_ReadBSTR	속성 모음의 속성에서 BSTR 데이터 값을 읽습니다.
PSPropertyBag_ReadDWORD	속성 모음의 속성에서 DWORD 데이터 값을 읽습니다.

PSPropertyBag_ReadGUID 속성 모음의 속성에서 GUID 데이터 값을 읽습니다.
PSPropertyBag_ReadInt 속성 모음의 속성에서 int 데이터 값을 읽습니다.
PSPropertyBag_ReadLONG 속성 모음의 속성에서 LONG 데이터 값을 읽습니다.
PSPropertyBag_ReadPOINTL 지정된 속성 모음의 POINTL 구조에 저장된 속성 좌표를 검색합니다.
PSPropertyBag_ReadPOINTS 지정된 속성 모음의 POINTS 구조에 저장된 속성 좌표를 검색합니다.
PSPropertyBag_ReadPropertyKey 지정된 속성 모음에서 속성의 속성 키를 읽습니다.
PSPropertyBag_ReadRECTL 지정된 속성 모음에 포함된 속성에 저장된 사각형의 좌표를 검색합니다.
PSPropertyBag_ReadSHORT 속성 모음에 있는 속성의 SHORT 데이터 값을 읽습니다.
PSPropertyBag_ReadStr 속성 모음에서 속성의 문자열 데이터 값을 읽습니다.
PSPropertyBag_ReadStrAlloc 속성 모음의 속성에서 문자열 데이터 값을 읽고 읽는 문자열에 대한 메모리를 할당합니다.
PSPropertyBag_ReadStream 지정된 속성 모음에 포함된 지정된 속성에 저장된 데이터 스트림을 읽습니다.

PSPropertyBag_ReadType 속성 모음에 저장된 속성의 데이터 값 형식을 읽습니다.
PSPropertyBag_ReadULONGLONG 속성 모음의 속성에서 ULONGLONG 데이터 값을 읽습니다.
PSPropertyBag_ReadUnknown 속성 모음에서 알 수 없는 데이터 값의 지정된 속성을 읽습니다.
PSPropertyBag_WriteBOOL 속성 모음에 있는 속성의 BOOL 값을 설정합니다.
PSPropertyBag_WriteBSTR 속성 모음에 있는 속성의 BSTR 값을 설정합니다.
PSPropertyBag_WriteDWORD 속성 모음에 있는 속성의 DWORD 값을 설정합니다.
PSPropertyBag_WriteGUID 속성 모음에 있는 속성의 GUID 값을 설정합니다.
PSPropertyBag_WriteInt 속성 모음에 있는 속성의 int 값을 설정합니다.
PSPropertyBag_WriteLONG 속성 모음에 있는 속성의 LONG 값을 설정합니다.
PSPropertyBag_WritePOINTL 지정된 속성 모음의 aPOINTL 구조에 속성 좌표를 저장합니다.
PSPropertyBag_WritePOINTS 지정된 속성 모음의 aPOINTS 구조에 속성 좌표를 저장합니다.

PSPropertyBag_WritePropertyKey 속성 모음에 있는 속성의 속성 키 값을 설정합니다.
PSPropertyBag_WriteRECTL 속성 모음의 속성에 사각형의 좌표를 저장합니다.
PSPropertyBag_WriteSHORT 속성 모음에 있는 속성의 SHORT 값을 설정합니다.
PSPropertyBag_WriteStr 속성 모음에 있는 속성의 문자열 값을 설정합니다.
PSPropertyBag_WriteStream 속성 모음의 속성에 데이터 스트림을 씁니다.
PSPropertyBag_WriteULONGLONG 속성 모음에 있는 속성의 ULONGLONG 값을 설정합니다.
PSPropertyBag_WriteUnknown 속성 모음에 알 수 없는 데이터 값의 속성을 씁니다.
PSPropertyKeyFromString 문자열을 PROPERTYKEY 구조체로 변환합니다.
PSRefreshPropertySchema 지원되지 않습니다. 이 함수를 호출하는 것은 유효하지만 함수를 수행하기 위해 구현되지 않으므로 그렇게 할 이유가 없습니다.
PSRegisterPropertySchema 스키마 하위 시스템에 속성 설명 스키마 파일의 추가를 알릴 수 있습니다. (PSRegisterPropertySchema)
PSSetPropertyValue 속성 저장소의 속성 값을 설정합니다.

PSStringFromPropertyKey	해당 속성의 키에서 속성을 식별하는 문자열을 만듭니다.
PSUnregisterPropertySchema	스키마 하위 시스템에 속성 설명 스키마 파일 제거를 알릴 수 있습니다.
RefreshPropertySchema	지원 안 됨 (IPropertySystem.RefreshPropertySchema)
RegisterPropertySchema	스키마 하위 시스템에 속성 설명 스키마 파일의 추가를 알릴 수 있습니다. (IPropertySystem.RegisterPropertySchema)
RemoveAt	지정된 변경 사항을 제거합니다.
SetValueAndState	속성 키에 대한 값 및 상태 데이터를 설정합니다.
SHAddDefaultPropertiesByExt	지정된 파일 확장명용으로 등록된 속성 저장소에 기본 속성을 추가합니다.
SHGetPropertyStoreForWindow	특정 창의 속성 컬렉션을 나타내는 개체를 검색하여 해당 속성을 쿼리하거나 설정할 수 있습니다.
SHGetPropertyStoreFromIDList	포인터에서 PIDL(항목 식별자 목록)에 대한 IPropertyStore 또는 관련 인터페이스를 지원하는 개체를 검색합니다.
SHGetPropertyStoreFromParsingName	경로 또는 구문 분석 이름이 지정된 항목에 대한 속성 저장소를 반환합니다.

SHPropStgCreate 요청된 속성 집합 작업에 대한 코드 페이지 검색 또는 할당의 적절한 처리를 보장합니다.
SHPropStgReadMultiple 사용되지 않는 속성 집합에 대해 ANSI 및 유니코드 변환이 제대로 처리되도록 IPropertyStorage::ReadMultiple 함수를 래핑합니다.
SHPropStgWriteMultiple 사용되지 않는 속성 집합에 대해 ANSI 및 유니코드 변환이 제대로 처리되도록 IPropertyStorage::WriteMultiple 함수를 래핑합니다.
UnregisterPropertySchema 로컬 컴퓨터에서 .propdesc 파일의 파일 경로를 사용하여 속성 설명 스키마(.propdesc) 파일을 제거하는 스키마 하위 시스템에 알릴 수 있습니다.
VariantCompare 기본 비교 규칙에 따라 두 변형 구조를 비교합니다.
VariantGetBooleanElem 변형 구조체에서 단일 부울 요소를 추출합니다.
VariantGetDoubleElem 변형 구조체에서 하나의 double 요소를 추출합니다.
VariantGetElem 지정된 variant 요소에서 VARIANT 구조를 초기화합니다.
VariantGetElementCount 변형 구조체의 요소 수를 검색합니다.
VariantGetInt16Elem 변형 구조체에서 단일 Int16 요소를 추출합니다.

VariantGetInt32Elem 변형 구조체에서 단일 Int32 요소를 추출합니다.
VariantGetInt64Elem 변형 구조체에서 단일 Int64 요소를 추출합니다.
VariantGetStringElem 변형 구조체에서 단일 와이드 문자열 요소를 추출합니다.
VariantGetUInt16Elem 변형 구조체에서 서명되지 않은 단일 Int16 요소를 추출합니다.
VariantGetUInt32Elem 변형 구조체에서 서명되지 않은 단일 Int32 요소를 추출합니다.
VariantGetUInt64Elem 변형 구조체에서 서명되지 않은 단일 Int64 요소를 추출합니다.
VariantToBoolean VARIANT 구조체에서 부울 속성의 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
VariantToBooleanArray VARIANT 구조체에서 부울 값 배열을 추출합니다.
VariantToBooleanArrayAlloc BOOL 값의 배열을 할당한 다음 VARIANT 구조에서 해당 배열로 데이터를 추출합니다.
VariantToBooleanWithDefault VARIANT 구조체에서 BOOL 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.

VariantToBuffer VT_ARRRAY 형식의 VARIANT 구조에 저장된 버퍼의 내용을 추출합니다.	VT_UI1.
VariantToDosDateTime VARIANT 구조에서 Microsoft MS-DOS 형식의 날짜 및 시간 값을 추출합니다.	
VariantToDouble VARIANT 구조체에서 DOUBLE 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.	
VariantToDoubleArray VARIANT 구조에서 DOUBLE 값 배열을 추출합니다.	
VariantToDoubleArrayAlloc DOUBLE 값 배열을 할당한 다음 VARIANT 구조에서 해당 배열로 데이터를 추출합니다.	
VariantToDoubleWithDefault VARIANT 구조체에서 DOUBLE 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.	
VariantToFileTime 변형 구조체에서 FILETIME 구조를 추출합니다.	
VariantToGUID 변형 구조체의 GUID 속성 값을 추출합니다.	
VariantToInt16 변형 구조체의 Int16 속성 값을 추출합니다. 값을 추출할 수 없는 경우 이 함수에서 기본값을 할당합니다.	
VariantToInt16Array 벡터 구조에서 Int16 배열로 데이터를 추출합니다.	

VariantToInt16ArrayAlloc 벡터 구조에서 새로 할당된 Int16 배열로 데이터를 추출합니다.
VariantToInt16WithDefault 변형 구조체의 Int16 속성 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.
VariantToInt32 변형 구조체의 Int32 속성 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
VariantToInt32Array 벡터 구조에서 Int32 배열로 데이터를 추출합니다.
VariantToInt32ArrayAlloc 벡터 구조에서 새로 할당된 Int32 배열로 데이터를 추출합니다.
VariantToInt32WithDefault 변형 구조체의 Int32 속성 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.
VariantToInt64 변형 구조체의 Int64 속성 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
VariantToInt64Array 벡터 구조에서 Int64 배열로 데이터를 추출합니다.
VariantToInt64ArrayAlloc 벡터 구조에서 새로 할당된 Int64 배열로 데이터를 추출합니다.
VariantToInt64WithDefault 변형 구조체의 Int64 속성 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.

VariantToPropVariant	VARIANT 구조체의 내용을 PROPVARIANT 구조체에 복사합니다.
VariantToString	변형 구조체의 변형 값을 문자열로 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
VariantToStringAlloc	변형 구조체의 변형 값을 새로 할당된 문자열로 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
VariantToStringArray	벡터 구조에서 문자열 배열로 데이터를 추출합니다.
VariantToStringArrayAlloc	벡터 구조에서 새로 할당된 문자열 배열로 데이터를 추출합니다.
VariantToStringWithDefault	변형 구조체의 문자열 속성 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.
VariantToStrRet	원본 변형이 VT_BSTR 경우 문자열을 추출하여 STRRET 구조에 배치합니다.
VariantToUInt16	변형 구조체의 부호 없는 Int16 속성 값을 추출합니다. 값을 추출할 수 없는 경우 이 함수에서 기본값을 할당합니다.
VariantToUInt16Array	벡터 구조에서 서명되지 않은 Int16 배열로 데이터를 추출합니다.
VariantToUInt16ArrayAlloc	벡터 구조에서 새로 할당된 부호 없는 Int16 배열로 데이터를 추출합니다.

VariantToUInt16WithDefault	변형 구조체의 부호 없는 Int16 속성 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.
VariantToUInt32	변형 구조체의 부호 없는 Int32 속성 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
VariantToUInt32Array	벡터 구조에서 서명되지 않은 Int32 배열로 데이터를 추출합니다.
VariantToUInt32ArrayAlloc	벡터 구조에서 새로 할당된 부호 없는 Int32 배열로 데이터를 추출합니다.
VariantToUInt32WithDefault	변형 구조체의 부호 없는 Int32 속성 값을 추출합니다. 현재 값이 없으면 지정된 기본값이 반환됩니다.
VariantToUInt64	변형 구조체의 부호 없는 Int64 속성 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
VariantToUInt64Array	벡터 구조에서 서명되지 않은 Int64 배열로 데이터를 추출합니다.
VariantToUInt64ArrayAlloc	벡터 구조에서 새로 할당된 부호 없는 Int64 배열로 데이터를 추출합니다.
VariantToUInt64WithDefault	변형 구조체의 부호 없는 Int64 속성 값을 추출합니다. 현재 값이 없으면 지정된 기본값이 반환됩니다.
WinRTPropertyValueToPropVariant	Windows 런타임 속성 값에서 PROPVARIANT 구조로 콘텐츠를 복사합니다.

인터페이스

IPropertyChange 단일 속성에 대한 변경 사항을 캡슐화하는 메서드를 노출합니다.
IPropertyChangeArray IFileOperation에 전달될 수 있는 여러 변경 작업에 대한 메서드를 노출합니다.
IPropertyDescription 개별 속성 설명 세부 정보를 열거하고 검색하는 메서드를 노출합니다. (IPropertyDescription)
IPropertyDescription2 개별 속성 설명 세부 정보를 열거하고 검색하는 메서드를 노출합니다. (IPropertyDescription2)
IPropertyDescriptionAliasInfo 항목에 대한 "정렬 기준" 열 속성을 가져오는 메서드를 노출합니다. 이 인터페이스는 지정된 속성에 대한 기본 또는 보조 정렬 열을 검색하려는 UI 개체에서 사용됩니다.
IPropertyDescriptionList 목록으로 표시되는 속성 설명 컬렉션에서 정보를 추출하는 메서드를 노출합니다.
IPropertyDescriptionRelatedPropertyInfo IPropertyDescription 인터페이스를 검색하는 메서드를 제공합니다.
IPropertyDescriptionSearchInfo 속성에 대한 검색 관련 정보를 노출합니다.
IPropertyEnumType 열거형 정보에서 데이터를 추출하는 메서드를 노출합니다. IPropertyEnumType은 런타임에 프로그래밍 방식으로 속성 스키마의 enum 및 enumRange 요소에 대한 액세스를 제공합니다.
IPropertyEnumType2 열거형 정보에서 데이터를 추출하는 메서드를 노출합니다. IPropertyEnumType2는 IPropertyEnumType을 확장합니다.

IPropertyEnumTypeList 속성에 대해 가능한 값을 열거하는 메서드를 노출합니다.
IPropertyStoreCache 처리기가 각 속성에 대한 다양한 상태를 관리할 수 있도록 하는 메서드를 노출합니다.
IPropertyStoreCapabilities 사용자가 UI에서 속성을 편집할 수 있는지 여부를 결정하는 메서드를 노출합니다.
IPropertyStoreFactory IPropertyStore 개체를 가져오는 메서드를 노출합니다.
IPropertySystem 속성 설명을 얻고, 속성 스키마를 등록 및 등록 취소하고, 속성 설명을 열거하고, 형식이 엄격한 방식으로 속성 값의 서식을 지정하는 메서드를 노출합니다.
IPropertyUI 개발자는 대신 IPropertyDescription을 사용해야 합니다. (IPropertyUI)

구조체

PROPERTYKEY 속성을 프로그래밍 방식으로 식별하는 FMTID/PID 식별자를 지정합니다. SHCOLUMNID를 대체합니다.
PROPPRG 이 구조체에는 .pif 파일의 정보가 포함됩니다. PifMgr_GetProperties 사용됩니다.

Windows 원격 관리

아티클2023. 08. 22.

Windows 원격 관리 기술 개요입니다.

Windows 원격 관리를 개발하려면 다음 헤더가 필요합니다.

- wsman.h
- wsmandisp.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 원격 관리

열거형

WSManAuthenticationFlags 작업에 대한 인증 방법을 결정합니다.
WSManCallbackFlags 모든 콜백 함수에서 사용하는 플래그 집합을 정의합니다.
WSManDataType WSMAN_DATA 구조체에서 공용 구조체의 현재 데이터 형식을 지정합니다.
WSManProxyAccessType 프록시 액세스 유형을 정의합니다.
WSManProxyAccessTypeFlags 프록시 액세스 유형 플래그를 정의합니다.
WSManProxyAuthenticationFlags 프록시 인증 메커니즘을 결정합니다.
WSManSessionOption 세션에 대한 확장 옵션 집합을 정의합니다. 이러한 옵션은 WSManSetSessionOption 메서드와 함께 사용됩니다.

Functions

AddOption 요청을 처리하는 데 필요한 데이터를 추가합니다. 예를 들어 일부 WMI 공급자에는 공급자별 정보가 포함된 IWbemContext 또는 SWbemNamedValueSet 개체가 필요할 수 있습니다.
AddSelector ResourceLocator 개체에 선택기를 추가합니다. 선택기는 리소스의 특정 instance 지정합니다.
ClearOptions ResourceLocator 개체에서 옵션을 제거합니다.
ClearSelectors ResourceLocator 개체에서 모든 선택기를 제거합니다. Get, Put 또는 Enumerate와 같은 IWSManSession 개체 작업에서 리소스 URI를 지정하는 대신 ResourceLocator 개체를 제공할 수 있습니다.
만들기 리소스의 새 instance 만들고 새 개체의 EPR(엔드포인트 참조)을 반환합니다.
CreateConnectionOptions 세션을 만들 때 사용되는 사용자 이름과 암호를 지정하는 IWSManConnectionOptions 개체를 만듭니다.
CreateResourceLocator IWSManSession.Get, IWSManSession.Put 또는 Session.Enumerate와 같은 세션 개체 작업에서 리소스 URI 대신 사용할 수 있는 ResourceLocator 개체를 만듭니다.
CreateSession 그런 다음 후속 네트워크 작업에 사용할 수 있는 Session 개체를 만듭니다.
삭제 리소스 URI에 지정된 리소스를 삭제합니다.
열거 테이블, 데이터 수집 또는 로그 리소스를 열거합니다.
EnumerationFlagHierarchyDeep

IWSManSession::Enumerate 메서드의 flags 매개 변수에 사용할 열거형 상수 EnumerationFlagHierarchyDeep의 값을 반환합니다.
EnumerationFlagHierarchyDeepBasePropsOnly IWSManSession::Enumerate 메서드의 flags 매개 변수에 사용할 열거형 상수 EnumerationFlagHierarchyDeepBasePropsOnly의 값을 반환합니다.
EnumerationFlagHierarchyShallow IWSManSession::Enumerate 메서드의 flags 매개 변수에 사용할 열거형 상수 EnumerationFlagHierarchyShallow의 값을 반환합니다.
EnumerationFlagNonXmlText IWSManSession::Enumerate 메서드의 flags 매개 변수에 사용할 열거형 상수 WSManFlagNonXmlText의 값을 반환합니다.
EnumerationFlagReturnEPR IWSManSession::Enumerate 메서드의 flags 매개 변수에 사용할 열거형 상수 EnumerationFlagReturnEPR의 값을 반환합니다.
EnumerationFlagReturnObject IWSManSession::Enumerate 메서드의 flags 매개 변수에 사용할 열거형 상수 EnumerationFlagReturnObject의 값을 반환합니다.
EnumerationFlagReturnObjectAndEPR IWSManSession::Enumerate 메서드의 flags 매개 변수에 사용할 열거형 상수 EnumerationFlagReturnObjectAndEPR의 값을 반환합니다.
가져오기 URI에 지정된 리소스를 검색하고 리소스의 현재 instance XML 표현을 반환합니다.
get_AtEndOfStream IWSManEnumerator::ReadItem을 호출하여 IWSManEnumerator 개체의 항목 끝에 도달했음을 나타냅니다.
get_BatchItems 각 열거형 일괄 처리의 항목 수를 설정하고 가져옵니다. (Get)
get_CertificateThumbprint

클라이언트 인증서 인증을 사용하여 인증할 때 사용할 인증서 지문을 설정하거나 가져옵니다. (Get)
get_CommandLine 자동화 구성 요소를 로드하는 프로세스의 명령줄을 가져옵니다.
get_Error Windows 원격 관리 서비스에서 IWSManSession 개체, IWSManConnectionOptions 개체 또는 IWSManResourceLocator 개체를 만들 수 없는 경우 이전 IWSMan 메서드 호출에 대한 추가 오류 정보를 XML 스트림에서 가져옵니다.
get_Error 추가 오류 정보의 XML 표현을 가져옵니다. (IWSManEnumerator.get_Error)
get_Error 추가 오류 정보의 XML 표현을 가져옵니다. (IWSManResourceLocator.get_Error)
get_Error IWSManSession 개체 메서드에 대한 이전 호출에 대한 XML 스트림의 추가 오류 정보를 가져옵니다.
get_FragmentDialect IWSManResourceLocator가 Get, Put 또는 Enumerate와 같은 IWSManSession 개체 메서드에서 사용될 때 리소스 조각 방언에 대한 언어 언어를 가져오거나 설정합니다. (Get)
get_FragmentPath Get, Put 또는 Enumerate와 같은 IWSManSession 개체 메서드에서 ResourceLocator를 사용할 때 리소스 조각 또는 속성의 경로를 가져오거나 설정합니다. (Get)
get_MustUnderstandOptions ResourceLocator 개체의 MustUnderstandOptions 값을 가져오거나 설정합니다. (Get)
get_ResourceURI 요청된 리소스의 리소스 URI입니다. 이 속성은 특정 인스턴스에 대한 쿼리 문자열이 아닌 경로만 포함할 수 있습니다. (Get)
get_Timeout 클라이언트 애플리케이션이 Windows 원격 관리가 작업을 완료할 때까지 대기하는 최대 시간(밀리초)을 설정하고 가져옵니다. (Get)

get_UserName 원격 컴퓨터에서 로컬 또는 도메인 계정의 사용자 이름을 설정하고 가져옵니다. 이 속성은 인증에 대한 사용자 이름을 결정합니다. (Get)
GetErrorMessage 오류 번호의 텍스트가 포함된 서식이 지정된 문자열을 반환합니다.
Identify 원격 컴퓨터를 쿼리하여 WS-Management 프로토콜을 지원하는지 확인합니다.
Invoke 메서드를 호출하고 메서드 호출 결과를 반환합니다.
ProxyAuthenticationUseBasic IWSManConnectionOptionsEx2::SetProxy 메서드의 authenticationMechanism 매개 변수에 사용할 프록시 인증 플래그 WSManFlagProxyAuthenticationUseBasic의 값을 반환합니다.
ProxyAuthenticationUseDigest IWSManConnectionOptionsEx2::SetProxy 메서드의 authenticationMechanism 매개 변수에 사용할 프록시 인증 플래그 WSManFlagProxyAuthenticationUseDigest 값을 반환합니다.
ProxyAuthenticationUseNegotiate IWSManConnectionOptionsEx2::SetProxy 메서드의 authenticationMechanism 매개 변수에 사용할 프록시 인증 플래그 WSManFlagProxyAuthenticationUseNegotiate의 값을 반환합니다.
ProxyAutoDetect IWSManConnectionOptionsEx2::SetProxy 메서드의 accessType 매개 변수에 사용할 프록시 액세스 형식 플래그 WSManProxyAutoDetect의 값을 반환합니다.
ProxyIEConfig IWSManConnectionOptionsEx2::SetProxy 메서드의 accessType 매개 변수에 사용할 프록시 액세스 형식 플래그 WSManProxyIEConfig의 값을 반환합니다.
ProxyNoProxyServer IWSManConnectionOptionsEx2::SetProxy 메서드의 accessType 매개 변수에 사용할 프록시 액세스 형식 플래그 WSManProxyNoProxyServer의 값을 반환합니다.
ProxyWinHttpConfig

IWSManConnectionOptionsEx2::SetProxy 메서드의 accessType 매개 변수에 사용할 프록시 액세스 형식 플래그 WSManProxyWinHttpConfig의 값을 반환합니다.
넣어 리소스를 업데이트합니다.
put_BatchItems 각 열거형 일괄 처리의 항목 수를 설정하고 가져옵니다. (Put)
put_CertificateThumbprint 클라이언트 인증서 인증을 사용하여 인증할 때 사용할 인증서 지문을 설정하거나 가져옵니다. (Put)
put_FragmentDialect IWSManResourceLocator가 Get, Put 또는 Enumerate와 같은 IWSManSession 개체 메서드에서 사용될 때 리소스 조각 방언에 대한 언어 언어를 가져오거나 설정합니다. (Put)
put_FragmentPath Get, Put 또는 Enumerate와 같은 IWSManSession 개체 메서드에서 ResourceLocator를 사용할 때 리소스 조각 또는 속성의 경로를 가져오거나 설정합니다. (Put)
put_MustUnderstandOptions ResourceLocator 개체에 대한 MustUnderstandOptions 값을 가져오거나 설정합니다. (Put)
put_Password 원격 컴퓨터에서 로컬 또는 도메인 계정의 암호를 설정합니다.
put_ResourceURI 요청된 리소스의 리소스 URI입니다. 이 속성은 특정 인스턴스에 대한 쿼리 문자열이 아닌 경로만 포함할 수 있습니다. (Put)
put_Timeout 클라이언트 애플리케이션이 Windows 원격 관리가 작업을 완료할 때까지 대기하는 최대 시간(밀리초)을 설정하고 가져옵니다. (Put)
put_UserName 원격 컴퓨터에서 로컬 또는 도메인 계정의 사용자 이름을 설정하고 가져옵니다. 이 속성은 인증에 대한 사용자 이름을 결정합니다. (Put)

ReadItem 리소스에서 항목을 검색하고 항목의 XML 표현을 반환합니다.
SessionFlagCredUsernamePassword IWSMan::CreateSession의 flags 매개 변수에 사용할 인증 플래그 WSManFlagCredUsernamePassword 값을 반환합니다.
SessionFlagEnableSPNServerPort IWSMan::CreateSession의 flags 매개 변수에 사용할 인증 플래그 WSManFlagEnableSPNServerPort의 값을 반환합니다.
SessionFlagNoEncryption IWSMan::CreateSession의 flags 매개 변수에 사용할 인증 플래그 WSManFlagNoEncryption 값을 반환합니다.
SessionFlagSkipCACheck IWSMan::CreateSession 메서드의 flags 매개 변수에 사용할 WSManFlagSkipCACheck 인증 플래그의 값을 반환합니다.
SessionFlagSkipCNCheck IWSMan::CreateSession의 flags 매개 변수에 사용할 인증 플래그 WSManFlagSkipCNCheck 값을 반환합니다.
SessionFlagUseBasic IWSMan::CreateSession의 flags 매개 변수에 사용할 인증 플래그 WSManFlagUseBasic의 값을 반환합니다.
SessionFlagUseClientCertificate IWSMan::CreateSession의 flags 매개 변수에 사용할 인증 플래그 WSManFlagUseClientCertificate의 값을 반환합니다.
SessionFlagUseCredSsp IWSMan::CreateSession의 flags 매개 변수에 사용할 인증 플래그 WSManFlagUseCredSsp의 값을 반환합니다.
SessionFlagUseDigest IWSMan::CreateSession의 flags 매개 변수에 사용할 인증 플래그 WSManFlagUseDigest 값을 반환합니다.

SessionFlagUseKerberos IWSMan::CreateSession의 flags 매개 변수에 사용할 인증 플래그 WSManFlagUseKerberos의 값을 반환합니다.
SessionFlagUseNegotiate IWSMan::CreateSession의 flags 매개 변수에 사용할 인증 플래그 WSManFlagUseNegotiate의 값을 반환합니다.
SessionFlagUseNoAuthentication IWSMan::CreateSession의 flags 매개 변수에 사용할 인증 플래그 WSManFlagUseNoAuthentication 값을 반환합니다.
SessionFlagUTF8 IWSMan::CreateSession의 flags 매개 변수에 사용할 인증 플래그 WSManFlagUTF8의 값을 반환합니다.
SetProxy 세션에 대한 프록시 정보를 설정합니다.
WSMAN_PLUGIN_AUTHORIZE_OPERATION 특정 작업에 권한을 부여합니다.
WSMAN_PLUGIN_AUTHORIZE_QUERY_QUOTA 연결 권한이 부여된 후 사용자의 할당량 정보를 검색합니다.
WSMAN_PLUGIN_AUTHORIZE_RELEASE_CONTEXT 플러그 인이 WSManPluginAuthzUserComplete 또는 WSManPluginAuthzOperationComplete에서 보고하는 컨텍스트를 해제합니다.
WSMAN_PLUGIN_AUTHORIZE_USER 연결에 권한을 부여합니다.
WSMAN_PLUGIN_COMMAND 플러그 인에 대한 명령 콜백을 정의합니다.
WSMAN_PLUGIN_CONNECT 플러그 인에 대한 연결 콜백을 정의합니다.

WSMAN_PLUGIN_RECEIVE 플러그 인에 대한 수신 콜백을 정의합니다.
WSMAN_PLUGIN_RELEASE_COMMAND_CONTEXT 플러그 인에 대한 릴리스 명령 콜백을 정의합니다.
WSMAN_PLUGIN_RELEASE_SHELL_CONTEXT 플러그 인에 대한 릴리스 셸 콜백을 정의합니다.
WSMAN_PLUGIN_SEND 플러그 인에 대한 송신 콜백을 정의합니다.
WSMAN_PLUGIN_SHELL 플러그 인에 대한 셸 콜백을 정의합니다.
WSMAN_PLUGIN_SHUTDOWN 플러그 인에 대한 종료 콜백을 정의합니다.
WSMAN_PLUGIN_SIGNAL 플러그 인에 대한 신호 콜백을 정의합니다.
WSMAN_PLUGIN_STARTUP 플러그 인에 대한 시작 콜백을 정의합니다.
WSMAN_SHELL_COMPLETION_FUNCTION 셸 작업에 대해 호출되는 콜백 함수로, 원격 요청이 발생합니다.
WSManCloseCommand 명령을 삭제하고 연결된 리소스를 해제합니다.
WSManCloseOperation 비동기 작업을 취소하거나 닫습니다.
WSManCloseSession 세션 개체를 닫습니다.
WSManCloseShell

셸 개체를 삭제하고 셸과 연결된 리소스를 해제합니다.
WSManConnectShell 기존 서버 세션에 연결합니다.
WSManConnectShellCommand 셸에서 실행되는 기존 명령에 연결합니다.
WSManCreateSession 세션 개체를 만듭니다.
WSManCreateShell 셸 개체를 만듭니다.
WSManCreateShellEx 클라이언트 지정 셸 ID를 추가하여 WSManCreateShell 함수와 동일한 기능을 사용하여 셸 개체를 만듭니다.
WSManDeinitialize Windows 원격 관리 클라이언트 스택을 초기화합니다.
WSManDisconnectShell 활성 셸 및 관련 명령의 네트워크 연결을 끊습니다.
WSManGetErrorMessage 특정 오류 및 언어 코드와 관련된 오류 메시지를 검색합니다.
WSManGetSessionOptionAsDword 세션 옵션의 값을 가져옵니다. (WSManGetSessionOptionAsDword)
WSManGetSessionOptionAsString 세션 옵션의 값을 가져옵니다. (WSManGetSessionOptionAsString)
WSManInitialize Windows 원격 관리 클라이언트 API를 초기화합니다.
WSManPluginAuthzOperationComplete 사용자 작업에 대한 성공적인 권한 부여 또는 실패한 권한 부여를 보고합니다.

WSManPluginAuthzQueryQuotaComplete WSManPluginAuthzQueryQuota 플러그 인 진입점에서 호출되며 플러그 인이 요청을 수행할 수 있는지 여부를 호출해야 합니다.
WSManPluginAuthzUserComplete 성공적인 사용자 연결 권한 부여 또는 실패한 사용자 연결 권한 부여를 보고합니다.
WSManPluginFreeRequestDetails 작업 플러그 인 진입점에 전달되는 WSMAN_PLUGIN_REQUEST 구조에 할당된 메모리를 해제합니다.
WSManPluginGetOperationParameters 작업과 연결된 제한 시간 및 데이터 제한과 같은 항목에 대한 운영 정보를 가져옵니다.
WSManPluginOperationComplete WSManPluginStartup 및 WSManPluginShutdown 메서드를 제외한 모든 작업 진입점에 의한 작업 완료를 보고합니다.
WSManPluginReceiveResult WSMAN_PLUGIN_RECEIVE 플러그 인 호출에 대한 결과를 보고하고 결과를 반환하는 대부분의 셸 플러그 인에서 사용됩니다.
WSManPluginReportContext 셸 및/또는 명령에 대해 추가 작업을 수행할 수 있도록 셸 및 명령 컨텍스트를 WinRM(Windows 원격 관리) 인프라에 다시 보고합니다.
WSManReceiveShellOutput 실행 중인 명령 또는 셸에서 출력을 검색합니다.
WSManReconnectShell 이전에 연결이 끊긴 셸 세션을 다시 연결합니다. 셸 세션의 연결된 명령을 다시 연결하려면 WSManReconnectShellCommand를 사용합니다.
WSManReconnectShellCommand 이전에 연결이 끊긴 명령을 다시 연결합니다.
WSManRunShellCommand 기존 셸 내에서 명령 실행을 시작하고 명령이 완료될 때까지 기다리지 않습니다.

WSManRunShellCommandEx 명령 ID 옵션을 추가하여 WSManRunShellCommand 함수와 동일한 기능을 제공합니다.
WSManSendShellInput 입력 스트림을 실행 중인 명령 또는 셸에 Ipes합니다.
WSManSetSessionOption 세션에 대한 확장 옵션 집합을 설정합니다.
WSManSignalShell 기존 명령 또는 셸 자체에 컨트롤 코드를 보냅니다.

인터페이스

IWSMan Session 개체로 표시되는 세션을 만드는 데 사용되는 메서드 및 속성을 제공합니다.
IWSManConnectionOptions IWSManConnectionOptions 개체는 원격 컴퓨터의 로컬 계정과 연결된 사용자 이름과 암호를 제공하기 위해 IWSMan::CreateSession 메서드에 전달됩니다.
IWSManConnectionOptionsEx IWSManConnectionOptionsEx 개체는 인증에 사용되는 클라이언트 인증서의 지문을 제공하기 위해 IWSMan::CreateSession 메서드에 전달됩니다.
IWSManConnectionOptionsEx2 IWSManConnectionOptionsEx2 개체는 IWSMan::CreateSession 메서드에 전달되어 프록시 서버에 연결할 인증 메커니즘, 액세스 유형 및 자격 증명을 제공합니다.
IWSManEnumerator WS-Management 프로토콜 WS-Enumeration:Enumerate 작업과 같은 작업에서 반환된 결과 스트림을 나타냅니다.
IWSManEx

IWSManResourceLocator 개체 만들기, 열거형 및 세션 플래그 값을 반환하는 메서드 및 확장된 오류 정보를 가져오는 메서드를 포함하도록 IWSMan 인터페이스의 메서드와 속성을 확장합니다.
IWSManEx2 클라이언트 인증서를 사용하여 인증과 관련된 세션 플래그 값을 반환하는 메서드를 포함하도록 IWSManEx 인터페이스의 메서드 및 속성을 확장합니다.
IWSManEx3 CredSSP(자격 증명 보안 지원 공급자)를 사용하여 인증과 관련된 세션 플래그 값을 반환하는 메서드를 포함하도록 IWSManEx 인터페이스의 메서드 및 속성을 확장합니다.
IWSManResourceLocator 리소스에 대한 경로를 제공합니다. IWSManSession.Get, IWSManSession.Put 또는 IWSManSession.Enumerate와 같은 IWSManSession 개체 작업의 리소스 URI 대신 IWSManResourceLocator 개체를 사용할 수 있습니다.
IWSManSession 작업 및 세션 설정을 정의합니다.

구조체

WSMAN_AUTHENTICATION_CREDENTIALS 인증 방법 및 서버 또는 프록시 인증에 사용되는 자격 증명을 정의합니다.
WSMAN_AUTHZ_QUOTA 권한 부여 플러그 인에 대한 사용자 단위로 할당량 정보를 보고합니다.
WSMAN_CERTIFICATE_DETAILS 클라이언트 인증서와 함께 전송된 인바운드 요청에 대한 클라이언트 정보를 저장합니다.
WSMAN_COMMAND_ARG_SET 명령줄에 전달되는 인수 집합을 나타냅니다.
WSMAN_DATA WinRM(Windows 원격 관리) API에 사용되는 인바운드 및 아웃바운드 데이터를 포함합니다.

WSMAN_DATA_BINARY 다양한 Windows 원격 관리 함수에 사용할 이진 데이터를 보유합니다.
WSMAN_DATA_TEXT 다양한 Windows 원격 관리 함수와 함께 사용할 텍스트 데이터를 보유합니다.
WSMAN_ENVIRONMENT_VARIABLE 이름 및 값 쌍을 사용하여 개별 환경 변수를 정의합니다.
WSMAN_ENVIRONMENT_VARIABLE_SET 환경 변수의 배열을 정의합니다.
WSMAN_ERROR WinRM(Windows 원격 관리) 클라이언트에서 반환하는 오류 정보를 포함합니다.
WSMAN_FILTER WSMAN_FILTER 나중에 사용하도록 예약되어 있습니다.
WSMAN_FRAGMENT WSMAN_FRAGMENT 나중에 사용할 수 있습니다.
WSMAN_KEY 선택기 집합 내의 키 및 값 쌍을 나타내며 특정 리소스를 식별하는 데 사용됩니다.
WSMAN_OPERATION_INFO 플러그 인이 요청을 수행해야 하는 특정 리소스 엔드포인트를 나타냅니다.
WSMAN_OPTION 특정 옵션 이름 및 값 쌍을 나타냅니다.
WSMAN_OPTION_SET 옵션 집합을 나타냅니다.
WSMAN_PLUGIN_REQUEST 플러그 인 요청에 대한 정보를 지정합니다.
WSMAN_PROXY_INFO

프록시 정보를 지정합니다.
WSMAN_RECEIVE_DATA_RESULT WSManReceiveShellOutput 메서드에서 받은 출력 데이터를 나타냅니다.
WSMAN_RESPONSE_DATA WSMan 작업에서 받은 출력 데이터를 나타냅니다.
WSMAN_SELECTOR_SET 리소스의 ID를 나타내는 키 집합을 정의합니다.
WSMAN_SENDER_DETAILS 모든 인바운드 요청에 대한 클라이언트 세부 정보를 지정합니다.
WSMAN_SHELL_ASYNC 모든 셸 작업에 전달할 비동기 구조를 정의합니다.
WSMAN_SHELL_DISCONNECT_INFO 클라이언트의 연결이 끊긴 후에도 셸이 열린 상태로 유지되는 최대 기간(밀리초)을 지정합니다.
WSMAN_SHELL_STARTUP_INFO_V10 WSMAN_SHELL_STARTUP_INFO_V10(wsman.h) 구조체는 WSManCreateShell 함수와 함께 사용할 셸 시작 매개 변수를 정의합니다.
WSMAN_SHELL_STARTUP_INFO_V11 WSMAN_SHELL_STARTUP_INFO_V11(wsman.h) 구조체는 WSManCreateShell 함수와 함께 사용할 셸 시작 매개 변수를 정의합니다.
WSMAN_STREAM_ID_SET 셸 및 명령에 대한 입력 또는 출력에 사용되는 모든 스트림을 나열합니다.
WSMAN_USERNAME_PASSWORD_CREDS 인증에 사용되는 자격 증명을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Windows 리본 프레임워크

아티클2023. 08. 22.

Windows 리본 프레임워크 기술 개요입니다.

Windows 리본 프레임워크를 개발하려면 다음 헤더가 필요합니다.

- uiribbon.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 리본 프레임워크

열거형

UI_COLLECTIONCHANGE 컬렉션에 적용할 수 있는 변경 형식을 식별하는 값을 지정합니다.
UI_COMMANDTYPE 리본 컨트롤과 연결된 명령의 형식을 식별하는 값을 지정합니다.
UI_CONTEXTAVAILABILITY 상황에 맞는 탭의 가용성을 식별하는 값을 지정합니다.
UI_CONTROLDOCK QAT(빠른 실행 도구 모음)의 도크 상태를 식별하는 값을 지정합니다.
UI_EVENTLOCATION 리본 컨트롤과 연결된 이벤트가 시작될 수 있는 위치를 식별합니다.
UI_EVENTTYPE 리본 메뉴와 연결된 이벤트 유형을 식별합니다.
UI_EXECUTIONVERB 사용자가 명령에서 시작할 수 있는 작업에 매핑되는 실행 ID를 식별하는 값을 지정합니다.
UI_FONTDELTASIZE

강조 표시된 텍스트 실행의 글꼴 크기를 늘리거나 감소시켜야 하는지 여부를 식별하는 값을 지정합니다.
UI_FONTPROPERTIES Strikethrough와 같은 FontControl의 글꼴 속성 상태를 식별하는 값을 지정합니다.
UI_FONTUNDERLINE FontControl의 밑줄 상태를 식별하는 값을 지정합니다.
UI_FONTVERTICALPOSITION FontControl의 세로 맞춤 상태를 식별하는 값을 지정합니다.
UI_INVALIDATIONS 무효화할 명령의 측면을 식별하는 값을 지정합니다.
UI_OWNERSHIP Windows 리본 프레임워크에서 이미지를 만드는 소유권 조건을 식별하는 값을 지정합니다.
UI_SWATCHCOLORMODE 견본에 표준 모드 또는 단색 모드가 있는지 여부를 지정합니다.
UI_SWATCHCOLORTYPE DropDownColorPicker 또는 FontControl 색 선택기(텍스트 색 또는 텍스트 강조 표시)의 색 견본을 채우는 방법을 식별하는 값을 지정합니다. 참고 권장 사항입니다.
UI_VIEWTYPE 리본 프레임워크 뷰를 식별하는 값을 지정합니다.
UI_VIEWVERB 리본 프레임워크 보기에서 완료할 작업 유형을 식별하는 값을 지정합니다.

Functions

추가 IUICollection의 끝에 항목을 추가합니다.

지우기 IUICollection에서 모든 항목을 삭제합니다.
CreateImage 비트맵 이미지에서 IUIImage 개체를 만듭니다.
파괴 Windows 리본 프레임워크의 instance 대한 모든 개체, 후크 및 참조를 종료하고 해제합니다.
실행 명령 처리기에 바인딩된 명령의 실행 이벤트에 응답합니다.
FlushPendingInvalidations 보류 중인 모든 명령 업데이트를 처리합니다.
GetBitmap Windows 리본 프레임워크의 리본 및 컨텍스트 팝업 UI에서 아이콘으로 표시할 비트맵을 검색합니다.
GetCount IUICollection에 포함된 항목 수를 검색합니다.
GetHeight 리본의 높이를 검색합니다.
GetItem 지정된 인덱스의 IUICollection에서 항목을 검색합니다.
GetUICommandProperty 명령 속성, 값 또는 상태를 검색합니다.
GetValue 속성 키로 식별된 값을 검색합니다.
Getview IUIRibbon 또는 IUIContextualUI와 같은 Windows 리본 프레임워크 뷰를 나타내는 인터페이스에 대한 포인터의 주소를 검색합니다.

초기화 호스트 애플리케이션을 Windows 리본 프레임워크에 연결합니다.
삽입 지정된 인덱스의 IUICollection에 항목을 삽입합니다.
InvalidateUICommand Windows 리본 프레임워크 Command 속성, 값 또는 상태를 무효화합니다.
LoadSettingsFromStream 이진 스트림에서 리본 설정을 읽습니다.
LoadUI Windows 리본 프레임워크 UI 리소스 또는 컴파일된 태그 파일을 로드합니다.
OnChanged IUICollection이 변경되면 호출됩니다.
OnCreateUICommand IUICommandHandler에 명령을 바인딩하기 위해 Windows 리본 프레임워크 태그에 지정된 각 명령에 대해 호출됩니다.
OnDestroyUICommand 애플리케이션 창이 제거될 때 Windows 리본 프레임워크 태그에 지정된 각 명령에 대해 호출됩니다.
OnUIEvent 리본 이벤트가 발생했음을 알리는 알림을 받습니다.
OnViewChanged 보기 상태가 변경되면 호출합니다.
RemoveAt 지정된 인덱스의 IUICollection에서 항목을 제거합니다.
바꾸기 IUICollection의 지정된 인덱스에 있는 항목을 다른 항목으로 바꿉니다.

SaveSettingsToStream 리본 설정을 이진 스트림에 씁니다.
SetEventLogger 리본 이벤트에 대한 이벤트 로거를 설정합니다.
SetModes 사용하도록 설정할 애플리케이션 모드를 지정합니다.
SetUICommandProperty 명령 속성, 값 또는 상태를 설정합니다.
ShowAtLocation ContextPopup을 표시합니다.
UpdateProperty Windows 리본 프레임워크의 속성 업데이트 요청에 응답합니다.

인터페이스

IUIApplication IUIApplication 인터페이스는 애플리케이션에서 구현되며 Windows 리본 프레임워크에 대한 콜백 진입점 메서드를 정의합니다.
IUICollection IUICollection 인터페이스는 리본 프레임워크에 의해 구현됩니다.
IUICollectionChangedEvent IUICollectionChangedEvent 인터페이스는 애플리케이션에서 구현되며 런타임에 컬렉션의 변경 내용을 처리하는 데 필요한 메서드를 정의합니다.
IUICommandHandler IUICommandHandler 인터페이스는 애플리케이션에서 구현되며 명령 정보를 수집하고 Windows 리본 프레임워크에서 명령 이벤트를 처리하는 메서드를 정의합니다.

IUIContextualUI IUIContextualUI 인터페이스는 리본 프레임워크에서 구현되며 컨텍스트 팝업 보기의 핵심 기능을 제공합니다.
IUIEventingManager IUIEventingManager 인터페이스는 리본 프레임워크에서 구현되며 리본 이벤트에 등록하는 애플리케이션에 대한 알림 기능을 제공합니다.
IUIEventLogger IUIEventLogger 인터페이스는 애플리케이션에서 구현되며 리본 이벤트 콜백 메서드를 정의합니다.
IUIFramework IUIFramework 인터페이스는 Windows 리본 프레임워크에서 구현되며 프레임워크의 핵심 기능을 제공하는 메서드를 정의합니다.
IUIImage IUIImage 인터페이스는 애플리케이션에서 구현되며 Windows 리본 프레임워크 의 리본 및 컨텍스트 팝업 UI에 표시할 이미지를 검색하는 메서드를 정의합니다.
IUIImageFromBitmap IUIImageFromBitmap은 IUIImage 개체를 만드는 메서드를 정의하는 Windows 리본 프레임워크에서 구현하는 팩터리 인터페이스입니다.
IUIRibbon IUIRibbon 인터페이스는 Windows 리본 프레임워크에서 구현되며 리본 메뉴의 설정 및 속성을 지정하는 기능을 제공합니다.
IUISimplePropertySet IUISimplePropertySet은 속성 키로 식별된 값을 검색하는 메서드를 정의하는 읽기 전용 인터페이스입니다.

구조체

UI_EVENTPARAMS 리본 이벤트에 대한 정보를 포함합니다.

UI_EVENTPARAMS_COMMAND 이벤트와 연결된 명령에 대한 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Windows 런타임 C++ 참조

아티클2025. 01. 23.

이 기술 영역의 헤더는 데스크톱 애플리케이션에서 연결된 Windows 런타임 API를 호출할 수 있도록 제공됩니다.

연결된 Windows 런타임 API와 상호 운용되는 C++ 앱을 개발하려면 다음 헤더가 필요합니다.

- accountssettingspaneinterop.h
- activation.h
- activationregistration.h
- asyncinfo.h
- corewindow.h
- dragdropinterop.h
- eventtoken.h
- hstring.h
- inputpaneinterop.h
- inspectable.h
- memorybuffer.h
- playtomanagerinterop.h
- print3dmanagerinterop.h
- printmanagerinterop.h
- radialcontrollerinterop.h
- remotesystemadditionalinfo.h
- restrictederrorinfo.h
- robuffer.h
- rometadata.h
- rometadataapi.h
- rometadataresolution.h
- roparameterizediid.h
- roregistrationapi.h
- sharewindowcommandsourceinterop.h
- shcore.h
- systemmediatransportcontrolsinterop.h
- uiviewsettingsinterop.h
- useractivityinterop.h
- userconsentverifierinterop.h
- weakreference.h
- webauthenticationcoremanagerinterop.h

- .interop.h windows.data.pdf
- windows.devices.display.core.interop.h
- windows.foundation.h
- windows.graphics.capture.interop.h
- windows.graphics.display.interop.h
- windows.graphics.effects.interop.h
- windows.graphics.imaging.interop.h
- windows.media.core.interop.h
- windows.ui.composition.interop.h
- windows.ui.viewmanagement.core.coreframeworkinputviewinterop.h
- windows.ui.xaml.hosting.desktopwindowxamlsource.h
- windows.ui.xaml.hosting.referencetracker.h
- windows.ui.xaml.media.dxinterop.h
- windowsstoragecom.h
- winstring.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- windows 런타임 C++ 참조

열거형

테이블 확장

__x_ABI_CWindows_CFoundation_CActivationType 활성화 가능한 클래스에 대한 활성화 종류를 지정합니다. A
ActivationType ActivationType(activationregistration.h)은 활성화 가능한 클래스에 대한 활성화 종류를 지정합니다.
ACTIVATIONTYPE ACTIVATIONTYPE 열거형(objidl.h)은 활성화 가능한 클래스에 대한 활성화 종류를 지정합니다.
agileReferenceOptions RoGetAgileReference 함수에 대한 옵션을 지정합니다.
asyncStatus

비동기 작업의 상태를 나타냅니다.
BSOS_OPTIONS COM(구성 요소 개체 모델) IStream을 캡슐화하는 RandomAccessStream의 동작을 지정합니다.
GRAPHICS_EFFECT_PROPERTY_MAPPING 강력한 형식의 효과 속성이 기본 Direct2D 효과 속성에 매핑되는 방법을 나타냅니다.
HANDLE_ACCESS_OPTIONS 파일에 대한 핸들의 액세스 수준을 정의합니다.
HANDLE_CREATION_OPTIONS 존재하거나 존재하지 않는 파일에 대해 수행할 작업을 나타냅니다.
HANDLE_OPTIONS 파일 핸들의 플래그를 정의합니다.
HANDLE_SHARING_OPTIONS 파일 핸들의 요청된 공유 모드를 정의합니다.
IdentityType out-of-process 서버에 대한 정품 인증 종류를 나타냅니다.
instancingType out-of-process 서버에 대한 인스턴스화 동작의 종류를 나타냅니다.
PropertyType IPropertyValue 인터페이스를 구현하는 Windows 런타임 개체에 저장된 형식을 식별합니다.
PropertyType IPropertyValue 인터페이스를 구현하는 Windows 런타임 개체에 저장된 형식을 식별합니다.
RegistrationScope 활성화 가능한 클래스의 배포 범위를 나타냅니다.
RO_ERROR_REPORTING_FLAGS RoOriginateError 및 RoTransformError 함수의 동작을 지정합니다.

RO_INIT_TYPE 이 스레드에서 만든 개체에 대한 들어오는 호출에 사용되는 동시성 모델을 결정합니다.
RoErrorReportingFlags RoOriginateError 및 RoTransformError 함수의 동작을 지정합니다.
ThreadingType In-process 서버를 활성화하는 데 사용할 아파트 스레딩 모델을 나타냅니다.
TrustLevel 활성화 가능한 클래스의 신뢰 수준을 나타냅니다.

함수

테이블 확장

ActivateInstance 현재 정품 인증 팩터리에 연결된 Windows 런타임 클래스의 새 인스턴스를 만듭니다.
AddMemoryPressure 마지막 알림 이후 증가된 메모리 할당을 호스트에 알립니다. CLR은 이를 사용하여 가비지 수집을 실행할 시기를 결정하는 알고리즘을 알릴 수 있습니다.
AddRefFromReferenceTracker 참조 추적기가 FindTrackerTargets에 대한 이전 호출에서 대상 XAML 개체를 반환하고 있음을 나타냅니다. 참조는 IUnknown::AddRef 대신 참조 추적기 개체에 의해 유지됩니다.
AddRefFromTrackerSource 추적기 원본이 참조 추적기에서 IUnknown::AddRef를 호출할 때마다를 나타냅니다. AddRef 호출 후에 호출되었습니다.
attachToWindow 현재 **IDesktopWindowXamlSourceNative** 인스턴스를 창 핸들과 연결된 데스크톱 앱의 부모 UI 요소에 연결합니다.

BeginDraw 표면에 그리기를 시작합니다.
BeginDraw 그리기 위해 제공된 DXGI 표면을 엽니다.
BeginDraw 연결된 SurfaceImageSource 또는 VirtualSurfaceImageSource에 대한 업데이트를 시작합니다.
버퍼 버퍼의 바이트 배열을 가져옵니다.
취소 이미 진행 중인 비동기 작업의 취소를 요청합니다.
CapturePropagationContext 언어 경계를 넘어 스레드 간에 예외의 컨텍스트를 캡처합니다.
CheckCompositionTextureSupport 미정
닫기 비동기 작업 개체를 닫습니다.
닫기 할당된 리소스의 해제, 해제 또는 재설정과 관련된 애플리케이션 정의 작업을 수행합니다.
닫기 할당된 리소스의 해제, 해제 또는 재설정과 관련된 애플리케이션 정의 작업을 수행합니다.
CloseEnum 지정된 열거형 인스턴스에 대한 참조를 해제합니다.
CloseEnum 지정된 핸들로 식별되는 열거자를 닫습니다.
CloseUnbufferedFileHandle

IUnbufferedFileHandleProvider::OpenUnbufferedFileHandle을 호출하여 만든 임의 액세스 바이트 스트림에서 핸들을 닫습니다.
CoDecodeProxy 프록시된 개체에 대한 인터페이스가 제공된 서버 프로세스에서 COM(구성 요소 개체 모델) 인터페이스의 구현을 찾습니다.
ConnectFromTrackerSource 참조 추적기 원본이 참조 추적기 개체에 첫 번째 COM 참조를 만들었음을 나타냅니다.
copySurface 컴퍼지션 그리기 표면(또는 컴퍼지션 가상 그리기 표면)의 내용을 다시 읽습니다.
CountEnum 지정된 열거자가 검색한 열거형의 요소 수를 가져옵니다.
만들기 스토리지 폴더에 있는 파일에 대한 핸들을 만듭니다.
만들기 파일에 대한 핸들을 만듭니다.
createBoolean 지정된 8비트 부울 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
createBoolean 지정된 8비트 부울 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateBooleanArray 지정된 8비트 부울 값 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateBooleanArray 지정된 8비트 부울 값 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateChar16 지정된 유니코드 문자를 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateChar16

지정된 유니코드 문자를 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateChar16Array 유니코드 문자의 지정된 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateChar16Array 유니코드 문자의 지정된 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateCompositionSurfaceForHandle 스왑 체인의 핸들에 사용할 CompositionSurface 인스턴스를 만듭니다. CompositionSurface에서 미디어 스왑 체인을 호스트하려면 IMFMediaEngineEx::GetVideoSwapchainHandle 메서드를 사용합니다.
CreateCompositionSurfaceForSwapChain 스왑 체인에 사용할 CompositionSurface의 인스턴스를 만듭니다.
CreateCompositionTexture 미정
CreateControlInput 호출자의 UI 스레드에서 ICoreInputSourceBase 개체를 만듭니다.
CreateControlInputEx 작업자 스레드 또는 UI 스레드에 ICoreInputSourceBase 개체를 만듭니다.
CreateDateTime 지정된 DateTime 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateDateTime 지정된 DateTime 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateDateTimeArray 지정된 DateTime 값 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateDateTimeArray 지정된 DateTime 값 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateDirect3D11DeviceFromDXGIDevice

IDXGIDevice에서 IDirect3DDevice 인스턴스를 만듭니다.
CreateDirect3D11SurfaceFromDXGISurface IDXGISurface에서 IDirect3DSurface의 인스턴스를 만듭니다.
CreateDirect3DDevice IDXGIDeviceIDirect3DDevice 인스턴스를 만듭니다.
CreateDirect3DSurface IDXGISurfaceIDirect3DSurface 인스턴스를 만듭니다.
CreateDouble 지정된 64비트 부동 소수점 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateDouble 지정된 64비트 부동 소수점 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateDoubleArray 지정된 64비트 부동 소수점 값 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateDoubleArray 지정된 64비트 부동 소수점 값 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
createEmpty 빈 값을 나타내는 새 IPropertyValue 개체를 만듭니다.
createEmpty 빈 값을 나타내는 새 IPropertyValue 개체를 만듭니다.
CreateForMonitor 그래픽 캡처 항목을 만들기 위해 모니터를 대상으로 합니다.
CreateForWindow RadialController 개체를 인스턴스화하고 활성 애플리케이션에 바인딩합니다.
CreateForWindow 그래픽 캡처 항목을 만들기 위한 단일 창을 대상으로 합니다.

createFromMF2DBuffer2 제공된 IMF2DBuffer2에서 ISoftwareBitmapNative를 만듭니다.
CreateFromMFSample 제공된 IMFSample에서 IAudioFrameNative를 만듭니다.
CreateFromMFSample 제공된 IMFSample에서 IVideoFrameNative를 만듭니다.
CreateFromWICBitmap 제공된 IWICBitmap에서 ISoftwareBitmapNative를 만듭니다.
CreateGraphicsDevice 지정된 렌더링 디바이스에서 지원되는 CompositionGraphicsDevice를 만듭니다.
CreateGuid 지정된 GUID 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateGuid 지정된 GUID 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateGuidArray 지정된 Guid 값 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateGuidArray 지정된 Guid 값 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateInspectable 지정된 IInspectable 개체를 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateInspectable 지정된 IInspectable 개체를 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateInspectableArray 지정된 IInspectable 개체 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateInspectableArray

지정된 IInspectable 개체 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateInt32 지정된 부속 32비트 정수 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateInt32 지정된 부속 32비트 정수 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateInt32Array 서명된 32비트 정수 값의 지정된 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateInt32Array 서명된 32비트 정수 값의 지정된 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateInt64 지정된 서명된 64비트 정수 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateInt64 지정된 서명된 64비트 정수 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateInt64Array 서명된 64비트 정수 값의 지정된 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateInt64Array 서명된 64비트 정수 값의 지정된 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreatePoint 지정된 Point 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreatePoint 지정된 Point 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreatePointArray 지정한 점 값 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreatePointArray 지정한 점 값 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.

CreateRandomAccessStreamOnFile 파일에 대한 Windows 런타임 임의 액세스 스트림을 만듭니다.
CreateRandomAccessStreamOverStream IStream 기본 구현을 중심으로 Windows 런타임 임의 액세스 스트림을 만듭니다.
createRect 지정된 Rect 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
createRect 지정된 Rect 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateRectArray 지정된 Rect 값 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateRectArray 지정된 Rect 값 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateSessionForWindow 지정된 창과 연결된 UserActivitySession 만듭니다.
createSharedHandle DisplaySurface 또는 DisplayFence 개체의 경우 Direct3D 또는 기타 그래픽 API와의 interop에 사용할 수 있는 공유 핸들을 만듭니다.
createSingle 지정된 32비트 부동 소수점 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
createSingle 지정된 32비트 부동 소수점 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
createSingleArray 지정된 32비트 부동 소수점 값 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
createSingleArray 지정된 32비트 부동 소수점 값 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.

CreateSize 지정된 Size 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateSize 지정된 Size 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateSizeArray 지정된 Size 값 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateSizeArray 지정된 Size 값 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
createSourcePresentationHandle 이 경로에서 검색에 대한 액세스를 제어하기 위한 NT 핸들을 만듭니다.
CreateStreamOverRandomAccessStream Windows 런타임 IRandomAccessStream 개체 주위에 IStream을 만듭니다.
CreateString 지정된 문자열 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateString 지정된 문자열 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateStringArray 지정된 문자열 값 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateStringArray 지정된 문자열 값 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateTimeSpan 지정된 TimeSpan 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateTimeSpan 지정된 TimeSpan 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateTimeSpanArray

TimeSpan 값의 지정된 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateTimeSpanArray TimeSpan 값의 지정된 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateUInt32 지정된 부호 없는 32비트 정수 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateUInt32 지정된 부호 없는 32비트 정수 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateUInt32Array 부호 없는 32비트 정수 값의 지정된 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateUInt32Array 부호 없는 32비트 정수 값의 지정된 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateUInt64 지정된 부호 없는 64비트 정수 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateUInt64 지정된 부호 없는 64비트 정수 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateUInt64Array 부호 없는 64비트 정수 값의 지정된 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateUInt64Array 부호 없는 64비트 정수 값의 지정된 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateUInt8 지정된 부호 없는 8비트 정수 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateUInt8 지정된 부호 없는 8비트 정수 값을 포함하는 새 IPropertyValue 개체를 만듭니다.
CreateUInt8Array 부호 없는 8비트 정수 값의 지정된 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.

CreateUInt8Array 부호 없는 8비트 정수 값의 지정된 배열을 포함하는 새 IPropertyValue 개체를 만듭니다.
DECLSPEC_XFGVIRT
DefineScope 새 메타데이터를 만들 수 있는 새 영역을 메모리에 만듭니다.
DisconnectFromTrackerSource 참조 추적기 원본이 참조 추적기 추적을 중지했음을 나타냅니다.
DisconnectUnusedReferenceSources 호스트가 가비지 수집을 수행하고 불필요한 참조 원본을 모두 제거하라는 요청입니다.
EndDraw 표면 개체에 그리기의 끝을 표시합니다.
EndDraw ISurfaceImageSourceNative::EndDraw 메서드(windows.ui.xaml.media.dxinterop.h)는 표면 그리기 작업을 닫습니다.
EndDraw ISurfaceImageSourceNativeWithD2D::EndDraw 메서드(windows.ui.xaml.media.dxinterop.h)는 표면 그리기 작업을 닫습니다.
EnumAssemblyRefs 어셈블리 매니페스트에 정의된 mdAssemblyRef 인스턴스를 열거합니다.
EnumCustomAttributes 지정된 형식 또는 멤버와 연결된 사용자 지정 특성 정의 토큰을 열거합니다.
EnumEvents 지정된 TypeDef 토큰에 대한 이벤트 정의 토큰을 열거합니다.
EnumExportedTypes 현재 메타데이터 범위의 어셈블리 매니페스트에서 참조되는 내보낸 형식을 열거합니다.

EnumFields 지정된 TypeDef 토큰에서 참조하는 형식에 대한 FieldDef 토큰을 열거합니다.
EnumFieldsWithName 지정된 이름을 사용하여 지정된 형식의 FieldDef 토큰을 열거합니다.
EnumFiles 현재 어셈블리 매니페스트에서 참조되는 파일을 열거합니다.
EnumGenericParamConstraints 지정된 토큰이 나타내는 제네릭 매개 변수와 연결된 제네릭 매개 변수 제약 조건 배열에 대한 열거자를 가져옵니다.
EnumGenericParams 지정된 TypeDef 또는 MethodDef 토큰과 연결된 제네릭 매개 변수 토큰 배열의 열거자를 가져옵니다.
EnumInterfaceImpls 인터페이스 구현을 나타내는 MethodDef 토큰을 열거합니다.
EnumManifestResources 현재 어셈블리 매니페스트에서 참조되는 리소스의 열거자에 대한 포인터를 가져옵니다.
EnumMemberRefs 지정된 형식의 멤버를 나타내는 MemberRef 토큰을 열거합니다.
EnumMembers 지정된 형식의 멤버를 나타내는 MemberDef 토큰을 열거합니다.
EnumMembersWithName 지정된 이름의 지정된 형식의 멤버를 나타내는 MemberDef 토큰을 열거합니다.
EnumMethodImpls 지정된 형식의 메서드를 나타내는 MethodBody 및 MethodDeclaration 토큰을 열거합니다.
EnumMethods 지정된 형식의 메서드를 나타내는 MethodDef 토큰을 열거합니다.

EnumMethodSemantics 지정된 메서드가 관련된 속성 및 속성 변경 이벤트를 열거합니다.
EnumMethodSpecs 지정된 MethodDef 또는 MemberRef 토큰과 연결된 MethodSpec 토큰 배열에 대한 열거자를 가져옵니다.
EnumMethodsWithName 지정된 이름을 가지며 지정된 TypeDef 토큰에서 참조하는 형식으로 정의된 메서드를 열거합니다.
EnumModuleRefs 가져온 모듈을 나타내는 ModuleRef 토큰을 열거합니다.
EnumParams 지정된 MethodDef 토큰에서 참조하는 메서드의 매개 변수를 나타내는 ParamDef 토큰을 열거합니다.
EnumPermissionSets 지정된 메타데이터 범위의 개체에 대한 사용 권한을 열거합니다.
EnumProperties 지정된 TypeDef 토큰에서 참조하는 형식의 속성을 나타내는 PropertyDef 토큰을 열거합니다.
열거형 현재 범위에서 독립 실행형 서명을 나타내는 서명 토큰을 열거합니다.
EnumTypeDefs 현재 범위 내의 모든 형식을 나타내는 TypeDef 토큰을 열거합니다.
EnumTypeRefs 현재 메타데이터 범위에 정의된 TypeRef 토큰을 열거합니다.
EnumTypeSpecs 현재 메타데이터 범위에 정의된 TypeSpec 토큰을 열거합니다.
EnumUnresolvedMethods 현재 메타데이터 범위에서 해결되지 않은 메서드를 나타내는 MemberDef 토큰을 열거합니다.

EnumUserStrings 현재 메타데이터 범위에서 하드 코딩된 문자열을 나타내는 문자열 토큰을 열거합니다.
FindAssembliesByName CLR(공용 언어 런타임)에서 참조를 확인하기 위해 사용하는 표준 규칙을 사용하여 지정된 이름의 어셈블리 배열을 가져옵니다.
FindAssembly 어셈블리의 이름을 가져옵니다.
findAssemblyModule 어셈블리 모듈의 이름을 찾습니다.
FindExportedTypeByName 이름 및 묶는 형식이 지정된 내보낸 형식에 대한 포인터를 가져옵니다.
FindManifestResourceByName 지정된 이름의 매니페스트 리소스에 대한 포인터를 가져옵니다.
FindMemberRef 지정된 형식으로 묶이고 지정된 이름과 메타데이터 서명이 있는 멤버 참조에 대한 MemberRef 토큰에 대한 포인터를 가져옵니다.
FindTrackerTargets 참조 추적기 원본에서 연결할 수 있는 참조 추적기 대상을 찾습니다. 는 ReferenceTrackingStarted와 FindTrackerTargetsCompleted 호출 간에 가비지 수집기에서 호출해야 합니다.
FindTrackerTargetsCompleted 가비지 수집 시스템이 IReferenceTracker::FindTrackerTargets에 필요한 모든 호출을 완료했음을 나타냅니다. 이때 XAML은 보호하려는 모든 참조 추적기 대상을 페깅했습니다.
FindTypeDefByName 지정된 이름을 가진 Type에 대한 TypeDef 메타데이터 토큰에 대한 포인터를 가져옵니다.
FindTypeRef 지정된 범위에 있고 지정된 이름을 가진 Type 참조에 대한 TypeRef 토큰에 대한 포인터를 가져옵니다.

FlushAllSurfacesWithDevice 지정된 디바이스와 연결된 모든 SurfaceImageSource 또는 VirtualSurfaceImageSource 개체에 대한 모든 현재 GPU 작업을 플러시합니다.
FoundTrackerTarget XAML 개체 참조 추적기 대상이 발견되면 호출됩니다.
get_ActivatableClassId 현재 활성화 가능한 클래스의 클래스 식별자를 가져옵니다.
get_ActivationType 현재 활성화 가능한 클래스에 대한 활성화 종류를 가져옵니다.
get_AppUserModelId 앱의 사용자 모델에 대한 식별자를 가져옵니다.
get_Attributes 현재 활성화 가능한 클래스와 연결된 특성을 가져옵니다.
get_CommandLine Out-of-process 서버를 시작하는 데 사용되는 명령줄을 가져옵니다.
get_DllPath In-process 서버에 대한 정규화된 경로를 가져옵니다.
get_ErrorCode 비동기 작업의 종료 상태를 검색합니다.
get_ExePath Out-of-process 서버에 대한 파일 경로를 가져옵니다.
get_Id 비동기 작업의 식별자를 검색합니다.
get_Identity out-of-process 서버의 ID를 가져옵니다.

get_IdentityType Out-of-process 서버에 대한 활성화 유형을 가져옵니다.
get_Instancing out-of-process 서버에 대한 인스턴스화 동작을 가져옵니다.
get_Permissions out-of-process 서버에 대한 권한을 가져옵니다.
get_RegisteredTrustLevel 현재 활성화 가능한 클래스의 신뢰 수준을 가져옵니다.
get_RegistrationScope 현재 활성화 가능한 클래스의 배포 범위를 가져옵니다.
get_ServerName 현재 out-of-process 서버의 이름을 가져옵니다.
get_ServerRegistration Out-of-process 서버에 대한 등록 세부 정보를 가져옵니다.
get_Status 비동기 작업의 상태를 나타내는 값을 가져옵니다.
get_ThreadingType In-process 서버를 활성화하기 위한 아파트 스레딩 모델을 가져옵니다.
get_Type 현재 IPropertyValue 개체에 저장된 값의 데이터 형식을 가져옵니다.
get_Type 현재 IPropertyValue 개체에 저장된 값의 데이터 형식을 가져옵니다.
get_WindowHandle 앱의 CoreWindow에 대한 핸들(HWND)을 가져옵니다.
get_WindowHandle

현재 IDesktopWindowXamlSourceNative 인스턴스와 연결된 부모 UI 요소의 창 핸들을 가져옵니다.
GetActivationFactory 지정된 런타임 클래스에 대한 활성화 팩터리를 가져옵니다. (WinRTGetActivationFactory)
GetAssemblyFromScope 현재 범위의 어셈블리에 대한 포인터를 가져옵니다.
GetAssemblyProps 지정된 메타데이터 서명이 있는 어셈블리의 속성 집합을 가져옵니다.
GetAssemblyRefProps 지정된 메타데이터 서명을 사용하여 어셈블리 참조의 속성 집합을 가져옵니다.
GetAvailableFence 해당 정보가 알려진 경우 컴퍼지션 텍스처를 사용할 수 있게 되는 시기를 나타내는 Direct3D 동기화 펜스/값 쌍을 검색합니다.
GetBlob 지정된 열 인덱스에서 BLOB(Binary Large Object)에 대한 포인터를 가져옵니다.
GetBlobHeapSize 검색된 이진 데이터에 대한 포인터에 대한 포인터입니다.
GetBoolean 현재 IPropertyValue 개체에 저장된 8비트 부울 값을 가져옵니다.
GetBoolean 현재 IPropertyValue 개체에 저장된 8비트 부울 값을 가져옵니다.
GetBooleanArray 현재 IPropertyValue 개체에 저장된 8비트 부울 값의 배열을 가져옵니다.
GetBooleanArray 현재 IPropertyValue 개체에 저장된 8비트 부울 값의 배열을 가져옵니다.
GetChar16 현재 IPropertyValue 개체에 저장된 유니코드 문자를 가져옵니다.

GetChar16 현재 IPropertyValue 개체에 저장된 유니코드 문자를 가져옵니다.
GetChar16Array 현재 IPropertyValue 개체에 저장된 유니코드 문자의 배열을 가져옵니다.
GetChar16Array 현재 IPropertyValue 개체에 저장된 유니코드 문자의 배열을 가져옵니다.
GetCodedTokenInfo 지정된 행 인덱스와 연결된 토큰 배열에 대한 포인터를 가져옵니다.
GetColumn 지정된 테이블의 지정된 열과 행 셀에 포함된 값에 대한 포인터를 가져옵니다.
GetColumnInfo 지정한 테이블의 지정된 열에 대한 데이터를 가져옵니다.
GetCORSystemDirectory 현재 CLR(공용 언어 런타임)을 보유하는 디렉터리를 가져옵니다. 이 메서드는 Out-of-process 디버거에서만 사용할 수 있습니다. 다른 구성 요소에서 호출되는 경우 E_NOTIMPL 반환합니다.
GetCustomAttributeByName 이름과 소유자가 지정된 사용자 지정 특성을 가져옵니다.
GetCustomAttributeProps 메타데이터 토큰이 지정된 경우 사용자 지정 특성의 값을 가져옵니다.
GetData 이 메서드는 소프트웨어 비트맵 데이터에 대한 액세스를 제공하는 인터페이스를 반환합니다.
GetData 이 메서드는 오디오 데이터에 대한 액세스를 제공하는 인터페이스를 반환합니다.
GetData 이 메서드는 비디오 데이터에 대한 액세스를 제공하는 인터페이스를 반환합니다.

GetDateTime 현재 IPropertyValue 개체에 저장된 DateTime 값을 가져옵니다.
GetDateTime 현재 IPropertyValue 개체에 저장된 DateTime 값을 가져옵니다.
GetDateTimeArray 현재 IPropertyValue 개체에 저장된 DateTime 값의 배열을 가져옵니다.
GetDateTimeArray 현재 IPropertyValue 개체에 저장된 DateTime 값의 배열을 가져옵니다.
GetDevice 이 메서드는 비디오 데이터와 연결된 디바이스를 반환합니다.
GetDouble 현재 IPropertyValue 개체에 저장된 64비트 부동 소수점 값을 가져옵니다.
GetDouble 현재 IPropertyValue 개체에 저장된 64비트 부동 소수점 값을 가져옵니다.
GetDoubleArray 현재 IPropertyValue 개체에 저장된 64비트 부동 소수점 값의 배열을 가져옵니다.
GetDoubleArray 현재 IPropertyValue 개체에 저장된 64비트 부동 소수점 값의 배열을 가져옵니다.
GetDXGIInterface IDirect3DDevice 인스턴스에서 DXGI 인터페이스를 검색합니다.
GetDXGIInterface IDirect3DSurface 인스턴스에서 DXGI 인터페이스를 검색합니다.
GetDXGIInterfaceFromObject 개체에서 DXGI 인터페이스를 검색합니다.
GetEffectId

효과의 ID를 검색합니다.
GetErrorDetails 제한된 오류 설명을 포함하여 오류에 대한 정보를 반환합니다.
GetExportedTypeProps 지정된 메타데이터 서명을 사용하여 내보낸 형식의 속성 집합을 가져옵니다.
getFieldMarshal 지정된 필드 메타데이터 토큰으로 표시되는 네이티브 관리되지 않는 필드 형식에 대한 포인터를 가져옵니다.
GetFieldProps 지정된 FieldDef 토큰에서 참조하는 필드와 연결된 메타데이터를 가져옵니다.
GetFileProps 지정된 메타데이터 서명이 있는 파일의 속성을 가져옵니다.
GetForMonitor 지정된 모니터에 대한 DisplayInformation 개체를 검색합니다.
GetForWindow 지정된 창에 대한 InputPane 개체의 인스턴스를 가져옵니다.
GetForWindow 지정된 창에 대한 PlayToManager 인스턴스를 가져옵니다.
GetForWindow 지정된 창의 PrintManager 인스턴스를 가져옵니다.
GetForWindow 활성 애플리케이션에 바인딩된 RadialControllerConfiguration 개체를 검색합니다.
GetForWindow 창 식별자(창 핸들)에 해당하는 ShareWindowCommandSource 개체를 가져옵니다. (IShareWindowCommandSourceInterop::GetForWindow)
GetForWindow

지정된 창에 대한 DisplayInformation 개체를 검색합니다.
GetForWindow 현재 애플리케이션 창에 대한 AccountsSettingsPane 개체를 가져옵니다.
GetForWindow 현재 애플리케이션의 창에 대한 Print3DManager 개체를 가져옵니다.
GetForWindow 지정된 창에 대한 ISystemMediaTransportControls 인터페이스의 인스턴스를 가져옵니다.
GetForWindow 현재 애플리케이션의 창에 대한 UIViewSettings 개체를 가져옵니다.
GetForWindow 지정된 창과 연결된 UserActivityRequestManager 만듭니다.
GetForWindow 활성 애플리케이션의 창에 대한 CoreDragDropManager 개체를 가져옵니다.
GetGenericParamConstraintProps 지정된 제약 조건 토큰으로 표시되는 제네릭 매개 변수 제약 조건과 연결된 메타데이터를 가져옵니다.
GetGenericParamProps 지정된 토큰이 나타내는 제네릭 매개 변수와 연결된 메타데이터를 가져옵니다.
GetGuid 지정된 인덱스에서 행에서 GUID를 가져옵니다.
GetGuid 현재 IPropertyValue 개체에 저장된 GUID 값을 가져옵니다.
GetGuid 현재 IPropertyValue 개체에 저장된 GUID 값을 가져옵니다.
GetGuidArray 현재 IPropertyValue 개체에 저장된 Guid 값의 배열을 가져옵니다.

GetGuidArray 현재 IPropertyValue 개체에 저장된 Guid 값의 배열을 가져옵니다.
GetGuidHeapSize GUID 힙의 크기(바이트)를 가져옵니다.
getIids 현재 Windows 런타임 클래스에서 구현하는 인터페이스를 가져옵니다.
GetInspectableArray 현재 IPropertyValue 개체에 저장된 IInspectable 개체에 대한 포인터 배열을 가져옵니다.
GetInspectableArray 현재 IPropertyValue 개체에 저장된 IInspectable 개체에 대한 포인터 배열을 가져옵니다.
GetInt32 현재 IPropertyValue 개체에 저장된 서명된 32비트 정수 값을 가져옵니다.
GetInt32 현재 IPropertyValue 개체에 저장된 서명된 32비트 정수 값을 가져옵니다.
GetInt32Array 현재 IPropertyValue 개체에 저장된 서명된 32비트 정수 값의 배열을 가져옵니다.
GetInt32Array 현재 IPropertyValue 개체에 저장된 서명된 32비트 정수 값의 배열을 가져옵니다.
GetInt64 현재 IPropertyValue 개체에 저장된 서명된 64비트 정수 값을 가져옵니다.
GetInt64 현재 IPropertyValue 개체에 저장된 서명된 64비트 정수 값을 가져옵니다.
GetInt64Array 현재 IPropertyValue 개체에 저장된 서명된 64비트 정수 값의 배열을 가져옵니다.
GetInt64Array

현재 IPropertyValue 개체에 저장된 서명된 64비트 정수 값의 배열을 가져옵니다.
GetInterface IDirect3DDxgiInterfaceAccess 개체에 의해 래핑되는 DXGI 인터페이스를 검색합니다.
GetInterfaceImplProps 지정된 메서드를 구현하는 Type 및 해당 메서드를 선언하는 인터페이스에 대한 메타데이터 토큰에 대한 포인터를 가져옵니다.
GetLanguageException 오류 개체에서 저장된 IUnknown 개체를 가져옵니다.
GetManifestResourceProps 지정된 메타데이터 서명을 사용하여 매니페스트 리소스의 속성 집합을 가져옵니다.
GetMemberProps 지정된 메타데이터 토큰에서 참조하는 Type 멤버의 이름, 이진 서명 및 상대 가상 주소를 포함한 메타데이터 정보를 가져옵니다.
GetMemberRefProps 지정된 토큰에서 참조하는 멤버와 연결된 메타데이터를 가져옵니다.
GetMetaDataStorage 지정된 섹션에 저장된 메타데이터의 크기와 내용을 가져옵니다.
GetMetaDataStreamInfo 지정된 인덱스에 있는 메타데이터 스트림의 이름, 크기 및 내용을 가져옵니다.
GetMethodProps 지정된 MethodDef 토큰에서 참조하는 메서드와 연결된 메타데이터를 가져옵니다.
GetMethodSemantics 지정된 MethodDef 토큰에서 참조하는 메서드와 지정된 EventProp 토큰에서 참조하는 쌍을 이루는 속성과 이벤트 간의 관계를 나타내는 플래그를 가져옵니다.
GetMethodSpecProps 지정된 MethodSpec 토큰에서 참조하는 메서드의 메타데이터 서명을 가져옵니다.

GetMode StorageFile.OpenAsync 메서드를 호출하여 임의 액세스 바이트 스트림을 열 때 사용된 파일 액세스 모드를 검색합니다.
GetModuleFromScope 현재 메타데이터 범위에서 참조되는 모듈에 대한 메타데이터 토큰을 가져옵니다.
GetModuleRefProps 지정된 메타데이터 토큰에서 참조하는 모듈의 이름을 가져옵니다.
GetNamedPropertyMapping 효과 속성에 대한 매핑을 검색합니다.
GetNameFromToken 지정된 메타데이터 토큰에서 참조하는 개체의 UTF-8 이름을 가져옵니다. 이 메서드는 사용되지 않습니다.
GetNativeCallConvFromSig 지정된 서명 포인터로 표시되는 메서드에 대한 네이티브 호출 규칙을 가져옵니다.
GetNestedClassProps 지정된 중첩 형식의 부모 형식에 대한 TypeDef 토큰을 가져옵니다.
GetNextBlob 테이블에서 다음 BLOB(Binary Large Object)의 인덱스입니다.
GetNextGuid 현재 테이블 열에 있는 다음 GUID 값의 인덱스입니다.
GetNextString 현재 테이블 열에 있는 다음 문자열의 인덱스입니다.
GetNextUserString 현재 테이블 열에 하드 코딩된 다음 문자열이 포함된 행의 인덱스를 가져옵니다.
GetNumTables 현재 IMetaDataTables 인스턴스의 범위에 있는 테이블 수를 가져옵니다.

GetOption 현재 메타데이터 범위에 대해 지정된 옵션의 값을 가져옵니다. 이 옵션은 현재 메타데이터 범위에 대한 호출을 처리하는 방법을 제어합니다.
GetParamForMethodIndex 지정된 MethodDef 토큰이 나타내는 메서드의 지정된 매개 변수를 나타내는 토큰을 가져옵니다.
GetParamProps 지정된 ParamDef 토큰에서 참조하는 매개 변수에 대한 메타데이터 값을 가져옵니다.
GetPEKind 현재 메타데이터 범위에 정의된 PE(이식 가능한 실행 파일) 파일(일반적으로 DLL 또는 EXE 파일)의 코드 특성을 식별하는 값을 가져옵니다.
GetPermissionSetProps 지정된 사용 권한 토큰으로 표시되는 System.Security.PermissionSet과 연결된 메타데이터를 가져옵니다.
GetPinvokeMap PInvoke 호출의 대상 어셈블리를 나타내는 ModuleRef 토큰을 가져옵니다.
GetPoint 현재 IPropertyValue 개체에 저장된 Point 값을 가져옵니다.
GetPoint 현재 IPropertyValue 개체에 저장된 Point 값을 가져옵니다.
GetPointArray 현재 IPropertyValue 개체에 저장된 Point 값의 배열을 가져옵니다.
GetPointArray 현재 IPropertyValue 개체에 저장된 Point 값의 배열을 가져옵니다.
GetPreviousLanguageExceptionErrorInfo 이전 언어 예외 오류 정보 개체를 검색합니다.
GetPropagationContextHead

전파 컨텍스트 헤드를 검색합니다.
GetProperty 지정된 인덱스에서 효과 속성을 검색합니다.
GetPropertyCount 효과에 대한 속성 수를 검색합니다.
GetRect 현재 IPropertyValue 개체에 저장된 Rect 값을 가져옵니다.
GetRect 현재 IPropertyValue 개체에 저장된 Rect 값을 가져옵니다.
GetRectArray 현재 IPropertyValue 개체에 저장된 Rect 값의 배열을 가져옵니다.
GetRectArray 현재 IPropertyValue 개체에 저장된 Rect 값의 배열을 가져옵니다.
getReference 제한된 오류 정보에 대한 참조를 반환합니다.
GetReferenceTrackerManager XAML 개체에서 IReferenceTrackerManager 인터페이스를 가져옵니다.
getRenderingDevice 렌더링 디바이스를 가져옵니다.
GetRestrictedErrorInfo 현재 논리 스레드에서 SetRestrictedErrorInfo에 대한 이전 호출에서 설정한 제한된 오류 정보 개체를 가져옵니다.
GetRow 지정한 테이블 인덱스 테이블의 지정된 행 인덱스에서 행을 가져옵니다.
GetRuntimeClassName 현재 Windows 런타임 개체의 정규화된 이름을 가져옵니다.

GetRVA 지정된 토큰이 나타내는 메서드 또는 필드의 상대 가상 주소(RVA) 및 구현 플래그를 가져옵니다.
GetScopeProps 현재 메타데이터 범위에서 어셈블리 또는 모듈의 이름과 버전 식별자를 선택적으로 가져옵니다.
GetSigFromToken 지정된 토큰과 연결된 이진 메타데이터 서명을 가져옵니다.
getSingle 현재 IPropertyValue 개체에 저장된 32비트 부동 소수점 값을 가져옵니다.
getSingle 현재 IPropertyValue 개체에 저장된 32비트 부동 소수점 값을 가져옵니다.
GetSingleArray 현재 IPropertyValue 개체에 저장된 32비트 부동 소수점 값의 배열을 가져옵니다.
GetSingleArray 현재 IPropertyValue 개체에 저장된 32비트 부동 소수점 값의 배열을 가져옵니다.
GetSize 현재 IPropertyValue 개체에 저장된 Size 값을 가져옵니다.
GetSize 현재 IPropertyValue 개체에 저장된 Size 값을 가져옵니다.
GetSizeArray 현재 IPropertyValue 개체에 저장된 Size 값의 배열을 가져옵니다.
GetSizeArray 현재 IPropertyValue 개체에 저장된 Size 값의 배열을 가져옵니다.
GetSource 지정된 인덱스에서 효과 원본을 검색합니다.
GetSourceCount

효과의 원본 수를 검색합니다.
GetSourceId TBDI
GetStackBackTrace 백 스택 추적을 검색합니다.
GetString 현재 참조 범위의 테이블 열에서 지정된 인덱스에서 문자열을 가져옵니다.
GetString 현재 IPropertyValue 개체에 저장된 문자열 값을 가져옵니다.
GetString 현재 IPropertyValue 개체에 저장된 문자열 값을 가져옵니다.
GetStringArray 현재 IPropertyValue 개체에 저장된 문자열 값의 배열을 가져옵니다.
GetStringArray 현재 IPropertyValue 개체에 저장된 문자열 값의 배열을 가져옵니다.
getStringHeapSize 문자열 힙의 크기(바이트)를 가져옵니다.
GetTableIndex 지정된 토큰에서 참조하는 테이블의 인덱스입니다.
GetTableInfo 지정된 테이블의 이름, 행 크기, 행 수, 열 수 및 키 열 인덱스를 가져옵니다.
GetTimeSpan 현재 IPropertyValue 개체에 저장된 TimeSpan 값을 가져옵니다.
GetTimeSpan 현재 IPropertyValue 개체에 저장된 TimeSpan 값을 가져옵니다.

GetTimeSpanArray 현재 IPropertyValue 개체에 저장된 TimeSpan 값의 배열을 가져옵니다.
GetTimeSpanArray 현재 IPropertyValue 개체에 저장된 TimeSpan 값의 배열을 가져옵니다.
GetTrackerTarget 참조 추적기 원본을 참조하는 참조 추적기 대상을 제공하도록 호스트에 요청합니다. 그런 다음 추적기 대상은 추적기 원본의 수명을 제어합니다.
GetTransformedRestrictedErrorInfo 변환된 제한된 오류 정보를 검색합니다.
GetTrustLevel 현재 Windows 런타임 개체의 신뢰 수준을 가져옵니다.
GetTypeDefProps 지정된 TypeDef 토큰으로 표시되는 Type에 대한 메타데이터 정보를 반환합니다.
GetTypeRefProps 지정된 TypeRef 토큰에서 참조하는 Type과 연결된 메타데이터를 가져옵니다.
GetTypeSpecFromToken 지정된 토큰이 나타내는 형식 사양의 이진 메타데이터 서명을 가져옵니다.
GetUInt32 현재 IPropertyValue 개체에 저장된 부호 없는 32비트 정수 값을 가져옵니다.
GetUInt32 현재 IPropertyValue 개체에 저장된 부호 없는 32비트 정수 값을 가져옵니다.
GetUInt32Array 현재 IPropertyValue 개체에 저장된 부호 없는 32비트 정수 값의 배열을 가져옵니다.
GetUInt32Array 현재 IPropertyValue 개체에 저장된 부호 없는 32비트 정수 값의 배열을 가져옵니다.

GetUInt64 현재 IPropertyValue 개체에 저장된 부호 없는 64비트 정수 값을 가져옵니다.
GetUInt64 현재 IPropertyValue 개체에 저장된 부호 없는 64비트 정수 값을 가져옵니다.
GetUInt64Array 현재 IPropertyValue 개체에 저장된 부호 없는 64비트 정수 값의 배열을 가져옵니다.
GetUInt64Array 현재 IPropertyValue 개체에 저장된 부호 없는 64비트 정수 값의 배열을 가져옵니다.
GetUInt8 현재 IPropertyValue 개체에 저장된 부호 없는 8비트 정수 값을 가져옵니다.
GetUInt8 현재 IPropertyValue 개체에 저장된 부호 없는 8비트 정수 값을 가져옵니다.
GetUInt8Array 현재 IPropertyValue 개체에 저장된 부호 없는 8비트 정수 값의 배열을 가져옵니다.
GetUInt8Array 현재 IPropertyValue 개체에 저장된 부호 없는 8비트 정수 값의 배열을 가져옵니다.
GetUpdateRectCount 업데이트해야 하는 표면의 총 영역 수를 가져옵니다.
GetUpdateRects 공유 화면에서 업데이트해야 하는 영역 집합을 가져옵니다.
GetUserString 지정된 메타데이터 토큰으로 표시되는 리터럴 문자열을 가져옵니다.
GetUserString 현재 범위의 문자열 열에 있는 지정된 인덱스에서 하드 코딩된 문자열을 가져옵니다.
GetUserStringHeapSize

사용자 문자열 힙의 크기(바이트)를 가져옵니다.
GetVersionString 어셈블리를 빌드하는 데 사용된 런타임의 버전 번호를 가져옵니다.
GetVisibleBounds 공유 표면의 표시되는 영역 경계를 가져옵니다.
GetWeakReference IWeakReferenceSource에서 약한 참조를 검색합니다.
GetWindow 창 식별자(창 핸들)를 가져옵니다.
HSTRING_UserFree HSTRING_UserFree 함수(inspectable.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HSTRING_UserFree HSTRING_UserFree 함수는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다. (HSTRING_UserFree)
HSTRING_UserFree HSTRING_UserFree 함수(winstring.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HSTRING_UserFree64 HSTRING_UserFree64 함수(inspectable.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HSTRING_UserFree64 HSTRING_UserFree64 함수는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다. (HSTRING_UserFree64)
HSTRING_UserMarshal HSTRING_UserMarshal 함수(inspectable.h)는 HSTRING 개체를 RPC 버퍼로 마샬링합니다.
HSTRING_UserMarshal

HSTRING_UserMarshal 함수는 HSTRING 개체를 RPC 버퍼로 마샬링합니다. (HSTRING_UserMarshal)
HSTRING_UserMarshal HSTRING_UserMarshal 함수(winstring.h)는 HSTRING 개체를 RPC 버퍼로 마샬링합니다.
HSTRING_UserMarshal64 HSTRING_UserMarshal64 함수(inspectable.h)는 HSTRING 개체를 RPC 버퍼로 마샬링합니다.
HSTRING_UserMarshal64 HSTRING_UserMarshal64 함수는 HSTRING 개체를 RPC 버퍼로 마샬링합니다. (HSTRING_UserMarshal64)
HSTRING_UserSize HSTRING_UserSize 함수(inspectable.h)는 HSTRING 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
HSTRING_UserSize HSTRING_UserSize 함수는 HSTRING 개체의 와이어 크기를 계산하고 핸들과 데이터를 검색합니다. (HSTRING_UserSize)
HSTRING_UserSize HSTRING_UserSize 함수(winstring.h)는 HSTRING 개체의 와이어 크기를 계산하고 핸들과 데이터를 검색합니다.
HSTRING_UserSize64 HSTRING_UserSize64 함수(inspectable.h)는 HSTRING 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
HSTRING_UserSize64 HSTRING_UserSize64 함수는 HSTRING 개체의 와이어 크기를 계산하고 핸들과 데이터를 검색합니다. (HSTRING_UserSize64)
HSTRING_UserUnmarshal HSTRING_UserUnmarshal 함수(inspectable.h)는 RPC 버퍼에서 HSTRING 개체를 숨깁니다.
HSTRING_UserUnmarshal HSTRING_UserUnmarshal 함수는 RPC 버퍼에서 HSTRING 개체를 숨깁니다. (HSTRING_UserUnmarshal)

HSTRING_UserUnmarshal HSTRING_UserUnmarshal 함수(winstring.h)는 RPC 버퍼에서 HSTRING 개체를 숨깁니다.
HSTRING_UserUnmarshal64 HSTRING_UserUnmarshal64 함수(inspectable.h)는 RPC 버퍼에서 HSTRING 개체를 숨깁니다.
HSTRING_UserUnmarshal64 HSTRING_UserUnmarshal64 함수는 RPC 버퍼에서 HSTRING 개체를 숨깁니다. (HSTRING_UserUnmarshal64)
int64FromLuid LUID에서 64비트 부가 정수로 변환합니다.
무효화 그리기를 위해 공유 표면의 특정 영역을 무효화합니다.
IsErrorPropagationEnabled Windows 런타임 API 이벤트에 대한 콜백 함수로 등록된 대리자가 반환하는 오류에 대해 CoreApplication.UnhandledErrorDetected 이벤트가 발생하는지 또는 비동기 메서드가 완료되었는지 여부를 나타냅니다.
IsGlobal 지정된 메타데이터 토큰으로 표현되는 필드, 메서드 또는 형식에 전역 범위가 있는지 여부를 나타내는 값을 가져옵니다.
IsValidToken 지정된 토큰이 코드 개체에 대한 유효한 참조를 보유하는지 여부를 나타내는 값을 가져옵니다.
LuidFromInt64 64비트 부가 정수에서 LUID로 변환합니다.
MetaDataGetDispenser 디스펜서 클래스를 만듭니다.
NotifyEndOfReferenceTrackingOnThread 호출 스레드에서 참조 추적을 더 이상 사용할 수 없음을 호스트에 알깁니다. XAML은 FrameworkView가 초기화되지 않은 경우 이를 호출합니다.

OnBrokenCallback IUnbufferedFileHandleProvider::OpenUnbufferedFileHandle 메서드를 호출하여 가져오는 핸들에 대한 기회 잠금이 끊어질 때 실행됩니다.
OnUninitialize 등록 된 아파트가 종료 될 때 호출됩니다.
openScope 디스크에서 기존 파일을 열고 읽기 위해 해당 메타데이터를 메모리에 매핑합니다.
OpenScopeOnITypeInfo 지정된 범위 유형을 엽니다.
openScopeOnMemory 기존 메타데이터가 포함된 메모리 영역을 엽니다. 즉, 이 메서드는 기존 데이터가 메타데이터로 처리되는 지정된 메모리 영역을 엽니다.
openSharedHandle 공유 기본 화면, 공유 펜스 및 원본 프레젠테이션 핸들에 대한 핸들을 엽니다.
OpenUnbufferedFileHandle StorageFile.OpenAsync 메서드가 만든 임의 액세스 바이트 스트림에서 핸들을 가져오고 핸들에 대한 기회 잠금이 끊어질 때 실행하려는 콜백 메서드를 등록합니다.
oplockBreaking 이 메서드는 현재 구현되지 않았습니다. (IOplockBreakingHandler.OplockBreaking)
pdfCreateRenderer PDF(이식 가능한 문서 형식) 파일의 단일 페이지를 표시하기 위한 IPdfRendererNative 인터페이스의 인스턴스를 가져옵니다.
pdfRenderParams PDF_RENDER_PARAMS 구조를 채웁니다. PDF_RENDER_PARAMS 구조체는 PDF(이식 가능한 문서 형식) 파일의 단일 페이지를 출력하기 위한 속성 집합을 나타냅니다.
페그 참조 추적기 대상이 XAML 프레임워크에서 사용 중이며 수집해서는 안 됨을 표시합니다.

PegFromTrackerSource 추적기 원본이 참조 추적기 개체를 보호할 수 없음을 나타냅니다.
PINSPECT_HSTRING_CALLBACK WindowsInspectString 함수에서 사용하는 콜백에 대한 함수 포인터를 제공합니다.
PINSPECT_MEMORY_CALLBACK RoInspectCapturedStackBackTrace 함수에서 사용하는 콜백에 대한 함수 포인터를 제공합니다.
preTranslateMessage WinRT XAML 프레임워크가 WinRT XAML 컨트롤을 호스트하는 **DesktopWindowXamlSource** 개체에 대한 Windows 메시지를 처리할 수 있도록 합니다.
put_MessageHandled CoreWindow에 대한 메시지가 처리되었는지 여부를 설정합니다. (ICoreInputInterop.put_MessageHandled)
put_MessageHandled CoreWindow에 대한 메시지가 처리되었는지 여부를 설정합니다. (ICoreWindowInterop.put_MessageHandled)
ReferenceTrackingCompleted 가비지 수집 시스템이 수집 프로세스를 완료했음을 나타냅니다. 이때 XAML은 추적된 참조를 업데이트하려는 스레드의 차단을 해제합니다.
ReferenceTrackingStarted 가비지 수집기가 컬렉션을 수행하고 있음을 나타냅니다. 컬렉션이 완료되면 가비지 수집기에서 FindTrackerTargetsCompleted를 호출합니다.
RegisterForUpdatesNeeded 공유 화면에 대한 업데이트가 요청될 때 그리기를 수행할 콜백을 등록합니다.
releaseDisconnectedReferenceSources 호스트가 참조 원본으로 연결이 끊긴 참조 추적기 개체에서 IUnknown::Release를 호출하도록 요청합니다.
releaseFromReferenceTracker AddRefFromReferenceTracker에 대한 이전 호출에 표시된 XAML 개체 참조를 해제합니다.

ReleaseFromTrackerSource 추적기 원본이 참조 추적기에서 IUnknown::Release를 호출할 때마다를 나타냅니다. 는 릴리스 호출 전에 호출되어야 합니다.
RemoveMemoryPressure 마지막 알림 이후 감소된 메모리 할당을 호스트에 알립니다.
RenderPageToDeviceContext IPdfRendererNative::RenderPageToDeviceContext 메서드는 PDF(이식 가능한 문서 형식) 파일의 단일 페이지를 비트맵 이미지로 출력합니다.
RenderPageToSurface PDF(이식 가능한 문서 형식) 파일의 단일 페이지를 Microsoft DirectX 이미지 데이터 개체에 출력합니다.
RequestTokenForWindowAsync 웹 계정 공급자로부터 토큰을 비동기적으로 요청합니다. 필요한 경우 사용자에게 자격 증명을 입력하라는 메시지가 표시됩니다. (IWebAuthenticationCoreManagerInterop::RequestTokenForWindowAsync)
RequestTokenWithWebAccountForWindowAsync 웹 계정 공급자로부터 토큰을 비동기적으로 요청합니다. 필요한 경우 사용자에게 자격 증명을 입력하라는 메시지가 표시됩니다. (IWebAuthenticationCoreManagerInterop::RequestTokenWithWebAccountForWindowAsync)
RequestVerificationForWindowAsync Microsoft Passport PIN, Windows Hello 또는 지문 판독기 같은 디바이스를 사용하여 확인을 수행합니다.
ResetEnum 지정된 열거자를 지정된 위치로 다시 설정합니다.
크기 조정 표면의 크기를 변경합니다.
크기 조정 표면 크기를 조정합니다.

해결 개체에 대한 Agile 참조의 인터페이스 ID를 가져옵니다.
해결 개체에 대한 강력한 참조를 반환하여 약한 참조를 해결합니다.
ResolveTypeRef 지정된 TypeRef 토큰이 나타내는 형식 참조를 확인합니다.
ResumeDraw Surface 개체에서 그리기를 다시 시작합니다.
ResumeDraw 그리기 작업을 다시 시작합니다.
RoActivateInstance 지정된 Windows 런타임 클래스를 활성화합니다.
RoCaptureErrorContext RoFailFastWithErrorContext 함수에 대한 이후 호출에 사용할 수 있도록 현재 오류 컨텍스트를 저장합니다.
RoClearError TEB(현재 스레드 환경 블록)에서 기존 오류 정보를 제거합니다.
RoCreatePropertySetSerializer IPropertySetSerializer의 새 인스턴스를 만듭니다.
RoFailFastWithErrorContext 현재 프로세스에서 연속할 수 없는 예외를 발생합니다.
RoFreeParameterizedTypeExtra RoGetParameterizedTypeInstanceIID에 의해 할당된 핸들을 해제합니다.
RoGetActivatableClassRegistration 클래스 등록 정보를 검색할 수 있습니다.

RoGetActivationFactory 지정된 런타임 클래스에 대한 활성화 팩터리를 가져옵니다. (RoGetActivationFactory)
RoGetAgileReference 지정된 인터페이스에서 지정한 개체에 대한 Agile 참조를 만듭니다.
RoGetApartmentIdentifier 현재 아파트의 고유 식별자를 가져옵니다.
RoGetBufferMarshaler 마샬링될 때 IBuffer 인터페이스와 연결된 의미 체계를 구현하는 표준 IBuffer 마샬러를 제공합니다.
RoGetErrorReportingFlags Windows 런타임 오류 함수의 현재 보고 동작을 가져옵니다.
RoGetMatchingRestrictedErrorInfo
RoGetMetaDataFile 지정된 형식 이름의 ABI(애플리케이션 이진 인터페이스)를 설명하는 메타데이터 파일을 찾아 검색합니다.
RoGetParameterizedTypeInstanceIID 매개 변수가 있는 인터페이스 또는 대리자가 지정된 형식 인수로 인스턴스화될 때 발생하는 인터페이스 또는 대리자 형식의 IID(인터페이스 식별자)를 계산합니다.
RoGetServerActivatableClasses 호출 프로세스의 패키지 ID에 따라 등록된 지정된 EXE(실행 파일) 서버에 대해 등록된 활성화 가능한 클래스를 검색합니다.
RoInitialize 지정된 동시성 모델을 사용하여 현재 스레드에서 Windows 런타임을 초기화합니다.
RoInspectCapturedStackBackTrace 디버거가 대상 프로세스에서 호출 스택을 검사하는 방법을 제공합니다.
RoInspectThreadErrorInfo 오류가 발생한 지점에서 호출 스택을 나타내는 오류 개체를 가져옵니다.

RoIsApiContractMajorVersionPresent 지정된 이름 및 주 버전 번호가 있는 API 계약이 있는지 여부를 나타내려면 true 또는 false를 반환합니다.
RoIsApiContractPresent 지정된 이름과 주 버전 및 부 버전 번호가 있는 API 계약이 있는지 여부를 나타내려면 true 또는 false를 반환합니다.
RoOriginateError 연결된 디버거에 오류 및 정보 문자열을 보고합니다. (RoOriginateError)
RoOriginateErrorW 연결된 디버거에 오류 및 정보 문자열을 보고합니다. (RoOriginateErrorW)
RoOriginateLanguageException 오류, 정보 문자열 및 오류 개체를 연결된 디버거에 보고합니다.
RoParameterizedTypeExtraGetTypeSignature 지정된 핸들을 사용하여 RoGetParameterizedTypeInstanceIID에 대한 마지막 호출에서 IID를 계산하는 데 사용되는 형식 서명을 가져옵니다.
RoParseTypeName 매개 변수가 있는 형식의 경우 형식 이름과 기존 형식 매개 변수를 구문 분석합니다.
RoRegisterActivationFactories Windows 런타임 exe 서버에 대한 배열 out-of-process 정품 인증 팩터리를 등록합니다.
RoRegisterForApartmentShutdown 현재 아파트가 종료될 때 호출할 IApartmentShutdown 콜백을 등록합니다.
RoReportFailedDelegate 대리자 오류가 발생할 때 전역 오류 처리기를 트리거합니다.
RoReportUnhandledError 처리되지 않은 예외가 발생하면 전역 오류 처리기를 트리거합니다.
RoResolveNamespace

Windows 런타임에서 지원하는 프로그래밍 언어에서 지정된 Windows 런타임 네임스페이스의 직접 자식, 형식 및 하위 네임스페이스를 결정합니다.
RoResolveRestrictedErrorInfoReference 지정된 참조를 기반으로 IRestrictedErrorInfo 인터페이스 포인터를 반환합니다.
RoRevokeActivationFactories Windows 런타임에서 등록된 정품 인증 팩터리 배열을 제거합니다.
RoSetErrorReportingFlags Windows 런타임 오류 함수의 보고 동작을 설정합니다.
RoTransformError 수정된 오류 및 정보 문자열을 연결된 디버거에 보고합니다.
RoTransformErrorW 변환된 오류와 정보 문자열을 연결된 디버거에 보고합니다.
RoUninitialize 현재 스레드에서 Windows 런타임을 닫습니다.
RoUnregisterForApartmentShutdown 이전에 등록된 IApartmentShutdown 인터페이스의 등록을 취소합니다.
스크롤 논리 표면의 사각형 영역을 스크롤합니다.
SetActivitySourceHost 호스트 정보를 UserActivityChannel 또는 UserActivity 개체와 연결합니다.
SetDevice D3D11_CREATE_DEVICE_BGRA_SUPPORT 사용하여 만든 DXGI 디바이스를 설정하여 표면을 그립니다.
SetDevice D3D11_CREATE_DEVICE_BGRA_SUPPORT 사용하여 만든 Microsoft DXGI(DirectX Graphics Infrastructure) 또는 Direct2D 디바이스를 설정하여 표면을 그립니다.

SetInputSource 앱의 CoreIndependentInputSource 또는 CoreComponentInputSource에 대한 입력 소스를 설정합니다.
setOption 지정된 옵션을 현재 메타데이터 범위에 대해 지정된 값으로 설정합니다. 이 옵션은 현재 메타데이터 범위에 대한 호출을 처리하는 방법을 제어합니다.
SetReferenceTrackerHost XAML에 IReferenceTrackerHost 인터페이스를 등록합니다.
setRenderingDevice 렌더링 디바이스를 설정합니다.
SetRestrictedErrorInfo 현재 스레드에 대한 제한된 오류 정보 개체를 설정합니다.
SetSwapChain SwapChainBackgroundPanel에 대한 DirectX 스왑 체인을 설정합니다.
SetSwapChain SwapChainPanel에 대한 DirectX 스왑 체인을 설정합니다.
SetSwapChainHandle 스왑 체인에 대한 핸들을 사용하여 SwapChainPanel에 대한 DirectX 스왑 체인을 설정합니다.
ShowAddAccountForWindowAsync 계정 추가 화면을 표시합니다.
showManageAccountsForWindowAsync 계정 관리 화면을 표시합니다.
ShowPlayToUIForWindow 지정된 창에 대한 재생 UI를 표시합니다.
ShowPrintUIForWindowAsync 지정한 창의 콘텐츠를 인쇄하기 위한 UI를 표시합니다.

ShowPrintUIForWindowAsync 프로그래밍 방식으로 3D 인쇄 사용자 인터페이스를 시작합니다.
SuspendDraw Surface 개체에서 그리기를 일시 중단합니다.
SuspendDraw 그리기 작업을 일시 중단합니다.
ToString 현재 개체를 나타내는 문자열을 가져옵니다.
ToString 현재 개체를 나타내는 문자열을 가져옵니다.
  참조 추적기 대상이 XAML 프레임워크에서 더 이상 사용되지 않으며 수집할 수 있음을 표시합니다.
UpdatesNeeded VirtualSurfaceImageSource에 대한 업데이트가 요청되면 그리기 동작을 수행합니다.
windowsCompareStringOrdinal 지정된 두 HSTRING 개체를 비교하고 정렬 순서로 상대 위치를 나타내는 정수를 반환합니다.
windowsConcatString 지정된 두 문자열을 연결합니다.
WindowsCreateString 지정된 소스 문자열을 기반으로 새 HSTRING을 만듭니다.
WindowsCreateStringReference 지정된 문자열을 기반으로 새 문자열 참조를 만듭니다.
WindowsDeleteString 문자열 버퍼의 참조 수를 줄입니다.
WindowsDeleteStringBuffer

HSTRING으로 승격되지 않은 경우 미리 할당된 문자열 버퍼를 삭제합니다.
windowsDuplicateString 지정된 문자열의 복사본을 만듭니다.
WindowsGetStringLen 유니코드 문자로 지정된 문자열의 길이를 가져옵니다.
WindowsGetStringRawBuffer 지정된 문자열의 백업 버퍼를 검색합니다.
WindowsInspectString 디버거가 다른 주소 공간, 원격 또는 덤프에서 Windows 런타임 HSTRING의 값을 표시하는 방법을 제공합니다. (WindowsInspectString)
windowsInspectString2 디버거가 다른 주소 공간, 원격 또는 덤프에서 Windows 런타임 HSTRING의 값을 표시하는 방법을 제공합니다. (WindowsInspectString2)
WindowsIsStringEmpty 지정된 문자열이 빈 문자열인지 여부를 나타냅니다.
WindowsPreallocateStringBuffer HSTRING 생성에 사용할 변경 가능한 문자 버퍼를 할당합니다.
windowsPromoteStringBuffer 지정된 HSTRING_BUFFER HSTRING을 만듭니다.
windowsReplaceString 지정된 문자열에 있는 문자 집합의 모든 항목을 다른 문자 집합으로 바꿔서 새 문자열을 만듭니다.
WindowsStringHasEmbeddedNull 지정된 문자열에 null 문자가 포함되어 있는지 여부를 나타냅니다.
windowsSubstring 지정된 문자열에서 부분 문자열을 검색합니다. 부분 문자열은 지정된 문자 위치에서 시작됩니다.
WindowsSubstringWithSpecifiedLength

지정된 문자열에서 부분 문자열을 검색합니다. 부분 문자열은 지정된 문자 위치에서 시작하며 지정된 길이를 줍니다.
WindowsTrimStringEnd 원본 문자열에서 지정된 문자 집합의 후행 항목을 모두 제거합니다.
windowsTrimStringStart 원본 문자열에서 지정된 문자 집합의 선행 항목을 모두 제거합니다.

인터페이스

테이블 확장

IAccountsSettingsPaneInterop IAccountsSettingsPaneInterop은 WinRT AccountsSettingsPane 개체와의 상호 운용성을 가능하게 하고 계정 창을 표시하는 멤버에 대한 액세스를 제공합니다.
IActivatableClassRegistration 클래스에 대한 등록 정보를 가져올 수 있습니다.
IActivationFactory Windows 런타임에서 클래스를 활성화할 수 있습니다.
IAgileReference 개체에 대한 Agile 참조를 검색할 수 있습니다.
IApartmentShutdown 아파트 종료 알림 처리기의 등록을 사용하도록 설정합니다.
IAsyncAction 비동기 작업을 나타냅니다.
IAsyncAction 비동기 작업을 나타냅니다.

IAsyncInfo 비동기 작업을 지원합니다.
IAudioFrameNative 오디오 데이터의 프레임을 나타냅니다.
IAudioFrameNativeFactory IAudioFrameNative의 인스턴스를 만듭니다.
IClosable 할당된 리소스를 해제하는 메서드를 정의합니다.
IClosable 할당된 리소스를 해제하는 메서드를 정의합니다.
ICompositionDrawingSurfaceInterop RECT를 사용하여 표면 개체에 그리기하여 그릴 영역을 정의할 수 있는 네이티브 상호 운용 인터페이스입니다. 이 인터페이스는 C++에서만 사용할 수 있습니다.
ICompositionDrawingSurfaceInterop2 컴퍼지션 드로잉 표면(또는 컴퍼지션 가상 그리기 표면)의 내용을 다시 읽을 수 있는 네이티브 상호 운용 인터페이스입니다.
ICompositionGraphicsDeviceInterop 그래픽 디바이스를 가져오고 설정할 수 있는 네이티브 상호 운용 인터페이스입니다. 이 인터페이스는 C++에서만 사용할 수 있습니다.
ICompositionTextureInterop 미정
ICompositorInterop 스왑 체인 표면 및 그래픽 디바이스를 만들 수 있는 네이티브 상호 운용 인터페이스입니다. 이 인터페이스는 C++에서만 사용할 수 있습니다.
ICompositorInterop2
ICoreInputInterop

Windows 애플리케이션의 핵심 입력 개체에서 입력 소스를 사용하도록 설정합니다.
ICoreWindowInterop 앱이 이 인터페이스와 연결된 창(CoreWindow)의 창 핸들을 가져올 수 있도록 합니다.
IDesktopWindowXamlSourceNative WinRT XAML 컨트롤을 호스트하는 **DesktopWindowXamlSource** 개체를 데스크톱 앱의 부모 UI 요소에 연결하는 데 사용할 수 있는 멤버를 제공합니다.
IDesktopWindowXamlSourceNative2 WinRT XAML 프레임워크가 WinRT XAML 컨트롤을 호스트하는 **DesktopWindowXamlSource** 개체에 대한 Windows 메시지를 처리할 수 있도록 하는 메서드를 제공합니다.
IDisplayDeviceInterop Windows 런타임 Windows.Devices.Display.Core 네임스페이스의 Direct3D(및 기타 그래픽 API)와 API 간의 상호 운용 방법을 제공합니다.
IDisplayInformationStaticsInterop DisplayInformation 정품 인증 팩터리에 대한 데스크톱 interop 인터페이스입니다.
IDisplayPathInterop Windows 런타임 클래스 DisplayPath 및 기타 디스플레이 관련 API 간에 상호 운용하는 메서드를 제공합니다.
IDllServerActivatableClassRegistration In-Process 서버에 대한 등록 정보를 가져올 수 있습니다.
IDragDropManagerInterop WinRT CoreDragDropManager 개체와의 상호 운용성을 사용하도록 설정합니다.
IExeServerActivatableClassRegistration Out-of-process 서버에 대한 등록 정보를 가져올 수 있습니다.
IExeServerRegistration 등록된 Out-of-process 서버를 나타냅니다.
IFindReferenceTargetsCallback IReferenceTracker::FindTrackerTargets에서 콜백에 대한 인터페이스를 정의합니다. 이 인터페이스의

구현은 찾은 모든 IReferenceTrackerTarget 인스턴스를 FoundTrackerTarget 메서드에 전달해야 합니다.
IGraphicsCaptureItemInterop GraphicsCaptureItem 정품 인증 팩터리에 대한 데스크톱 interop 인터페이스입니다.
IGraphicsEffectD2D1Interop IGraphicsEffect에 대응하고 메타데이터 쿼리를 허용하는 네이티브 상호 운용 인터페이스입니다. 이 인터페이스는 C++에서만 사용할 수 있습니다.
IInputPaneInterop 데스크톱 앱에서 InputPane 클래스의 멤버에 액세스할 수 있습니다.
IInspectable 모든 Windows 런타임 클래스에 필요한 기능을 제공합니다.
ILanguageExceptionErrorInfo RoOriginateLanguageException 호출을 사용하여 오류 정보에 저장된 IUnknown 포인터를 검색할 수 있습니다.
ILanguageExceptionErrorInfo2 언어 프로젝션이 ILanguageExceptionErrorInfo와 마찬가지로 오류 정보를 제공하고 검색할 수 있도록 하며, 언어 경계를 넘어 작업하는 추가적인 이점이 있습니다.
ILanguageExceptionStackBackTrace 프로젝션에서 해당 예외에 대한 사용자 지정 스택 추적을 제공할 수 있습니다.
ILanguageExceptionTransform 다른 예외를 catch하는 catch 처리기의 컨텍스트에서 throw되는 예외의 모든 컨텍스트 및 모든 컨텍스트를 시스템에서 사용할 수 있도록 언어 프로젝션을 허용합니다.
IMetaDataAssemblyImport 어셈블리 매니페스트의 내용에 액세스하고 검사하는 메서드를 제공합니다.
IMetaDataDispenser 새 메타데이터 범위를 만들거나 기존 메타데이터 범위를 여는 메서드를 제공합니다.
IMetaDataDispenserEx

IMetaDataDispenser 인터페이스를 확장하여 메타데이터 API가 현재 메타데이터 범위에서 작동하는 방식을 제어하는 기능을 제공합니다.
IMetaDataImport PE(이식 가능한 실행 파일) 파일 또는 다른 원본(예: 형식 라이브러리 또는 독립 실행형 런타임 메타데이터 이진)에서 기존 메타데이터를 가져오고 조작하는 메서드를 제공합니다.
IMetaDataImport2 IMetaDataImport 인터페이스를 확장하여 제네릭 형식으로 작업하는 기능을 제공합니다.
IMetaDataTables 테이블의 메타데이터 정보를 저장하고 검색하기 위한 메서드를 제공합니다.
IMetaDataTables2 메타데이터 스트림 작업을 위한 메서드를 포함하도록 IMetaDataTables를 확장합니다.
IOplockBreakingHandler 이 인터페이스는 현재 구현되지 않습니다.
IPdfRendererNative PDF(이식 가능한 문서 형식) 파일의 단일 페이지를 표시하기 위한 고성능 API를 나타냅니다.
IPlayToManagerInterop 여러 창을 관리하는 Windows 스토어 앱에서 PlayToManager 메서드에 액세스할 수 있습니다.
IPrinting3DManagerInterop WinRT Print3DManager 개체와의 상호 운용성을 사용하도록 설정합니다.
IPrintManagerInterop 여러 창을 관리하는 Windows 스토어 앱에서 PrintManager 메서드에 액세스할 수 있습니다.
IPropertyValue Windows 런타임 속성 저장소의 값을 나타냅니다.
IPropertyValue Windows 런타임 속성 저장소의 값을 나타냅니다.
IPropertyValueStatics

속성 저장소에 저장할 수 있는 IPropertyValue 개체를 만듭니다.
IPropertyValueStatics 속성 저장소에 저장할 수 있는 IPropertyValue 개체를 만듭니다.
IRadialControllerConfigurationInterop WinRT RadialControllerConfiguration 개체와의 상호 운용성을 사용하도록 설정하고 RadialController 메뉴를 사용자 지정하기 위해 RadialControllerConfiguration 멤버에 대한 액세스를 제공합니다.
IRadialControllerInterop WinRT RadialController 개체와의 상호 운용성을 사용하도록 설정하고 상호 작용 환경을 사용자 지정하기 위해 RadialController 멤버에 대한 액세스를 제공합니다.
IRandomAccessStreamFileAccessMode StorageFile.OpenAsync 메서드를 호출하여 임의 액세스 바이트 스트림을 열 때 사용된 파일 액세스 모드에 대한 액세스를 제공합니다.
IReferenceTracker XAML 개체 참조를 관리하기 위해 XAML 프레임워크에서 구현하는 인터페이스를 정의합니다.
IReferenceTrackerHost XAML 프레임워크에서 사용하는 GC(가비지 수집) 시스템에서 사용하는 전역 서비스를 제공하는 인터페이스를 정의합니다.
IReferenceTrackerManager XAML 개체 참조 관리자에 대한 인터페이스를 정의합니다. 이 인터페이스를 구현하여 XAML 개체에서 IReferenceTracker의 인스턴스를 관리합니다.
IReferenceTrackerTarget XAML에서 참조되는 가비지 수집기 개체에 의해 구현되는 인터페이스를 정의합니다.
IRestrictedErrorInfo 제한된 오류 정보를 포함하여 오류의 세부 정보를 나타냅니다.
IShareWindowCommandEventArgsInterop 창 식별자(창 핸들)에 해당하는 ShareWindowCommandSource 개체를 가져옵니다. (IShareWindowCommandEventArgsInterop)

IShareWindowCommandSourceInterop ShareWindowCommandSource 개체를 가져오도록 허용하는 네이티브 상호 운용 인터페이스입니다.
ISoftwareBitmapNative 소프트웨어 비트맵을 나타냅니다.
ISoftwareBitmapNativeFactory ISoftwareBitmapNative의 인스턴스를 만듭니다.
IStorageFolderHandleAccess 스토리지 폴더의 운영 체제 핸들에 대한 액세스를 제공합니다.
IStorageItemHandleAccess 스토리지 파일의 운영 체제 핸들에 대한 액세스를 제공합니다.
IStringable 현재 개체를 문자열로 나타내는 방법을 제공합니다.
IStringable 현재 개체를 문자열로 나타내는 방법을 제공합니다.
ISurfaceImageSourceManagerNative 동일한 프로세스에서 만든 모든 SurfaceImageSource 개체에서 대량 작업을 수행할 수 있습니다.
ISurfaceImageSourceNative Direct2D 드로잉에 대한 공유 고정 크기 화면의 구현을 제공합니다.
ISurfaceImageSourceNativeWithD2D SurfaceImageSource 또는 VirtualSurfaceImageSource에 표시되는 공유 Microsoft DirectX 표면의 구현을 제공합니다.
ISwapChainBackgroundPanelNative XAML과 DirectX 스왑 체인 간의 상호 운용성을 제공합니다.
ISwapChainPanelNative

ISwapChainPanelNative 인터페이스(windows.ui.xaml.media.dxinterop.h)는 XAML과 DirectX 스왑 체인 간의 상호 운용을 제공합니다.
ISwapChainPanelNative2 ISwapChainPanelNative2 인터페이스(windows.ui.xaml.media.dxinterop.h)는 XAML과 DirectX 스왑 체인 간의 상호 운용을 제공합니다.
ISystemMediaTransportControlsInterop 앱에서 ISystemMediaTransportControls 인터페이스의 인스턴스를 가져올 수 있습니다.
IUIViewSettingsInterop WinRT UIViewSettings 개체와의 상호 운용성을 사용하도록 설정합니다.
IUnbufferedFileHandleOplockCallback IUnbufferedFileHandleProvider::OpenUnbufferedFileHandle 메서드를 호출하여 가져오는 핸들에 대한 기회 잠금이 끊어질 때 실행할 콜백 메서드를 정의합니다.
IUnbufferedFileHandleProvider StorageFile.OpenAsync 메서드가 만든 임의 액세스 바이트 스트림의 핸들에 대한 액세스를 제공합니다.
IUserActivityInterop WinRT UserActivity 개체와의 상호 운용성을 사용하도록 설정합니다.
IUserActivityRequestManagerInterop WinRT UserActivityRequestManager 개체와의 상호 운용성을 사용하도록 설정합니다.
IUserActivitySourceHostInterop IUserActivitySourceHostInterop 인터페이스를 사용하면 WinRT와 UserActivityChannel 또는 UserActivity 개체와의 상호 운용성을 사용할 수 있습니다.
IUserConsentVerifierInterop WinRT UserConsentVerifier 클래스 개체와의 상호 운용성을 사용하도록 설정하고 UserConsentVerifier 멤버에 대한 액세스를 제공하여 현재 사용자를 확인합니다.
IVideoFrameNative 비디오 데이터의 프레임을 나타냅니다.
IVideoFrameNativeFactory

IVideoFrameNative의 인스턴스를 만듭니다.
IVirtualSurfaceImageSourceNative DirectX 드로잉에 대한 큰(화면 크기보다 큰) 공유 표면의 구현을 제공합니다.
IVirtualSurfaceUpdatesCallbackNative VirtualSurfaceImageSource가 업데이트를 요청할 때 그리기 동작을 구현하기 위한 인터페이스를 제공합니다.
IWeakReference 개체에 대한 약한 참조를 나타냅니다.
IWeakReferenceSource 약한 참조를 검색할 수 있는 원본 개체를 나타냅니다.
IWebAuthenticationCoreManagerInterop 웹 계정 공급자로부터 토큰을 가져오기 위한 핵심 메서드를 포함합니다.

구조

테이블 확장

dateTime 일반적으로 날짜 및 시간으로 표현되는 인스턴트 시간을 나타냅니다.
dateTime 일반적으로 날짜 및 시간으로 표현되는 인스턴트 시간을 나타냅니다.
eventRegistrationToken 이벤트 원본에 등록된 이벤트 처리기를 식별합니다.
HSTRING_HEADER HSTRING의 헤더를 나타냅니다.
IBufferByteAccess

버퍼를 바이트 배열로 나타냅니다.
IDirect3DDxgiInterfaceAccess IDirect3DDxgiInterfaceAccess는 IDirect3DDevice 또는 IDirect3DSurface를 구현하는 모든 항목에 의해 구현되어야 하는 COM 인터페이스입니다.
IRoMetaDataLocator RoGetParameterizedTypeInstanceIID 함수가 런타임 메타데이터에 액세스할 수 있도록 합니다.
IRoSimpleMetaDataBuilder 메타데이터 로케이터에 검색된 메타데이터의 대상을 제공합니다.
LUID 어댑터의 로컬 식별자를 설명합니다.
PDF_RENDER_PARAMS PDF(이식 가능한 문서 형식) 파일의 단일 페이지를 출력하기 위한 속성 집합을 나타냅니다.
지점 2차원 평면에서 점을 정의하는 순서가 지정된 부동 소수점 x 및 y 좌표 쌍을 나타냅니다.
지점 2차원 평면에서 점을 정의하는 순서가 지정된 부동 소수점 x 및 y 좌표 쌍을 나타냅니다.
Rect 사각형의 위치와 크기를 나타내는 4개의 부동 소수점 숫자 집합을 포함합니다.
Rect 사각형의 위치와 크기를 나타내는 4개의 부동 소수점 숫자 집합을 포함합니다.
ServerInformation 서버 프로세스에서 COM(구성 요소 개체 모델) 인터페이스의 구현을 나타냅니다.
크기 높이와 너비를 지정하는 순서가 지정된 부동 소수점 숫자 쌍을 나타냅니다.
크기 높이와 너비를 지정하는 순서가 지정된 부동 소수점 숫자 쌍을 나타냅니다.

TimeSpan 시간 간격을 나타냅니다.
TimeSpan 시간 간격을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

Windows 검색

아티클2023. 03. 16.

Windows Search 기술 개요입니다.

Windows Search를 개발하려면 다음 헤더가 필요합니다.

- filtereg.h
- structuredquery.h
- structuredquerycondition.h
- subsmgr.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 검색

열거형

AUTH_TYPE 콘텐츠 액세스에 대한 인증 유형을 설명합니다.
CASE_REQUIREMENT 쿼리에 대한 키워드의 사례 요구 사항(있는 경우)을 지정합니다.
CatalogPausedReason ISearchCatalogManager::GetCatalogStatus에서 카탈로그가 일시 중지된 이유를 확인하는 데 사용됩니다.
CatalogStatus ISearchCatalogManager::GetCatalogStatus에서 카탈로그의 현재 상태를 확인하는 데 사용됩니다.
CLUSION_REASON 이러한 플래그는 URL이 현재 크롤링 범위에서 포함되거나 제외되는 이유를 열거합니다.
CONDITION_CREATION_OPTIONS 조건 트리 노드:_ICondition, ICondition2, IConditionFactory, IConditionFactory2 및 IConditionGenerator의 형식을 나타내기 위해 다음 인터페이스와 함께 사용할 플래그 집합을 제공합니다.

CONDITION_OPERATION ICondition::GetComparisonInfo에서 작업을 나타내기 위해 다음 메서드와 함께 사용할 플래그 집합을 제공합니다. ICondition2::GetLeafConditionInfo, IConditionFactory::MakeLeaf, IConditionFactory2::CreateBooleanLeaf, IConditionFactory2::CreateIntegerLeaf, IConditionFactory2::MakeLeaf, IConditionFactory2::CreateStringLeaf 및 IConditionGenerator::GenerateForLeaf.
CONDITION_TYPE 조건 트리 노드:_ICondition::GetConditionType, IConditionFactory::MakeAndOr, IConditionFactory2::CreateCompoundFromArray 및 IConditionFactory2::CreateCompoundFromObjectArray의 형식을 나타내기 위해 다음 메서드와 함께 사용할 플래그 집합을 제공합니다.
FOLLOW_FLAGS 크롤링 또는 인덱싱 시 동작을 정의하는 데 사용됩니다. 이러한 플래그는 ISearchCrawlScopeManager::AddDefaultScopeRule 및 ISearchCrawlScopeManager::AddUserScopeRule 메서드에서 사용됩니다.
INTERVAL_LIMIT_KIND 이러한 값은 IInterval::GetLimits에서 쌍으로 반환되어 상한과 하한이 있는 범위를 지정합니다.
NAMED_ENTITY_CERTAINTY 명명된 엔터티의 확실성 수준을 정의합니다.
PRIORITY_LEVEL IRowsetPrioritization 인터페이스에서 쿼리로 지정된 범위에 대한 현재 인덱서 우선 순위 수준을 설정하거나 검색하는 데 사용됩니다.
PROXY_ACCESS ISearchManager에서 프록시 사용을 상태 지정하는 데 사용됩니다.
QUERY_PARSER_MANAGER_OPTION IQueryParserManager::SetOption에서 구문 분석 옵션을 설정하는 데 사용됩니다. 스키마 및 지역화 옵션을 지정하는 데 사용할 수 있습니다.
ROWSETEVENT_ITEMSTATE 행 집합의 검색 조건과 일치하는 항목이 현재 해당 행 집합에 있는지 여부를 설명합니다.

ROWSETEVENT_TYPE 행 집합의 데이터에 대한 변경 형식을 설명합니다.
SEARCH_INDEXING_PHASE 현재 검색 인덱싱 단계의 상태를 지정합니다.
SEARCH_KIND_OF_CHANGE 원본 싱크가 항목이 변경되었음을 클라이언트에 알립니다.
SEARCH_NOTIFICATION_PRIORITY 변경된 항목을 처리하는 우선 순위를 나타냅니다.
SEARCH_QUERY_SYNTAX 쿼리 구문의 형식을 지정합니다. (SEARCH_QUERY_SYNTAX)
SEARCH_TERM_EXPANSION 검색어의 와일드카드 옵션을 나타냅니다. ISearchQueryHelper::get_QueryTermExpansion 및 ISearchQueryHelper::p ut_QueryTermExpansion 메서드에서 사용됩니다.
STRUCTURED_QUERY_MULTIOPTION 개별 옵션을 나타내기 위해 IQueryParser::SetMultiOption에서 사용하는 플래그 집합입니다.
STRUCTURED_QUERY_PARSE_ERROR 구문 분석 오류를 식별하기 위해 IQuerySolution::GetErrors와 함께 사용할 플래그 집합입니다. 각 구문 분석 오류는 쿼리 문자열을 구문 분석할 때 하나 이상의 토큰이 무시되었음을 나타냅니다.
STRUCTURED_QUERY_RESOLVE_OPTION 조건 트리로 데이터를 확인하는 옵션입니다.
STRUCTURED_QUERY_SINGLE_OPTION 개별 옵션을 나타내기 위해 IQueryParser::SetOption 및 IQueryParser::GetOption과 함께 사용할 플래그 집합입니다.
STRUCTURED_QUERY_SYNTAX 쿼리 구문의 형식을 지정합니다. (STRUCTURED_QUERY_SYNTAX)

tagPRIORITIZE_FLAGS PrioritizeMatchingURL에서 인덱서가 이전에 인덱싱하지 못한 항목을 처리하는 방법을 지정하는 데 사용됩니다.

Functions

추가 구문 분석 중인 입력 문자열의 토큰화된 범위에서 식별된 대로 이 INamedEntityCollector 컬렉션에 명명된 단일 엔터티를 추가합니다.
AddDefaultScopeRule URL을 이 규칙의 기본 범위로 추가합니다.
AddHierarchicalScope 검색 엔진에 계층적 범위를 추가합니다.
AddRequestParameter 속성 값 집합을 요청합니다.
AddRoot 검색 엔진에 새 검색 루트를 추가합니다.
AddUserScopeRule 사용자가 새 규칙을 만들거나 인덱싱할 URL을 추가할 때 새 크롤링 범위 규칙을 추가합니다.
Base 이 엔터티의 부모 엔터티를 검색합니다.
BindToFilter 처리 중인 항목을 적절한 IFilterand에 바인딩하고 IFilter에 대한 포인터를 검색합니다.
BindToStream 처리 중인 항목을 IStream 인터페이스 [구조적 스토리지] 데이터 스트림에 바인딩하고 해당 스트림에 대한 포인터를 검색합니다.

BreakText 텍스트를 구문 분석하여 단어와 구를 식별하고 IWordSink 및 IPhraseSink 개체에 결과를 제공합니다.
복제 현재 내용과 상태가 동일한 IEnumSearchRoots 개체의 복사본을 만듭니다.
복제 현재 내용과 상태가 동일한 이 IEnumSearchScopeRules 개체의 복사본을 만듭니다.
복제 이 ICondition 개체의 전체 복사본을 만듭니다.
CloseAccessor 이전에 만든 IUrlAccessor 개체를 닫습니다.
ComposePhrase 지원 안 됨 (IWordBreaker.ComposePhrase)
CreateAccessor IUrlAccessor 개체를 만들고 초기화합니다.
CreateAccessorEx IUrlAccessor 개체를 만들고 초기화합니다. 이 메서드는 ISearchProtocol::CreateAccessor 메서드와 동일한 기본 기능을 사용하지만 프로토콜 처리기에 추가 데이터를 제공하는 추가 pUserData 매개 변수를 포함합니다.
CreateBooleanLeaf TRUE 또는 FALSE인 검색 조건을 만듭니다. (IConditionFactory2.CreateBooleanLeaf)
CreateCatalog Windows Search 인덱서에서 새 사용자 지정 카탈로그를 만들고 해당 카탈로그에 대한 참조를 반환합니다.
CreateCompoundFromArray 조건 노드 배열에서 AND(연결) 또는 OR(분리)인 리프 조건 노드를 만듭니다. 반환된 개체는 ICondition 및 ICondition2를 지원합니다.

CreateCompoundFromObjectArray 하위 조건 컬렉션의 AND(연결) 또는 OR(분리)인 리프 조건 노드를 만듭니다. 반환된 개체는 ICondition 및 ICondition2를 지원합니다.
CreateICommand ICommand를 만듭니다.
CreateIntegerLeaf 정수 값에 대한 리프 조건 노드를 만듭니다. 반환된 개체는 ICondition 및 ICondition2를 지원합니다.
CreateLeaf 모든 값에 대한 리프 조건 노드를 만듭니다. 반환된 개체는 ICondition 및 ICondition2를 지원합니다.
CreateLoadedParser IQueryParser 인터페이스 구현의 새 인스턴스를 만듭니다. 쿼리 파서의 이 인스턴스는 지정된 카탈로그에 대한 스키마와 함께 로드되고 지정된 언어로 지역화됩니다. 다른 모든 설정은 기본 설정으로 초기화됩니다.
CreateNegation 다른 조건(이 노드의 하위 노드)의 논리 부정(NOT)인 조건 노드를 만듭니다. (IConditionFactory2.CreateNegation)
CreateStringLeaf 속성 값과 상수 값의 비교를 나타내는 문자열 값에 대한 리프 조건 노드를 만듭니다. 반환된 개체는 ICondition 및 ICondition2를 지원합니다.
CreateTrueFalse TRUE 또는 FALSE인 검색 조건을 만듭니다. (IConditionFactory2.CreateTrueFalse)
DefaultPhrase 이 메서드는 IConditionGenerator의 이 인스턴스에서 인식할 때 엔터티, 관계 또는 명명된 엔터티의 형식 및 값 쌍을 나타내는 구를 생성하려고 합니다.
DefaultPhrase 다시 작업에서 이 엔터티에 사용할 기본 구를 검색합니다.

DefaultPhrase 이 명명된 엔터티에 사용할 기본 구를 다시 작업에서 검색합니다.
DefaultPhrase 다시 작업에서 이 관계에 사용할 기본 구를 검색합니다.
DeleteCatalog Windows Search 인덱서에서 기존 카탈로그 및 연결된 모든 인덱싱된 데이터를 삭제합니다.
대상 관계의 대상 IEntity 개체를 검색합니다. relationshipo의 대상은 속성의 형식에 해당합니다.
엔터티 로드된 스키마의 각 엔터티에 대해 하나의 항목을 사용하여 IEntity 개체의 열거형을 검색합니다.
EnumerateExcludedExtensions 구현되지 않았습니다. (ISearchCatalogManager.EnumerateExcludedExtensions)
EnumerateRoots ISearchCrawlScopeManager의 이 인스턴스가 인식하는 모든 루트의 열거형을 반환합니다.
EnumerateScopeRules ISearchCrawlScopeManager 인터페이스의 이 인스턴스가 인식하는 모든 범위 규칙의 열거형을 반환합니다.
EnumPropInfo 속성 목록에서 i-th 항목을 가져옵니다.
GenerateForLeaf 리프 쿼리 식이 될 특수 쿼리 식을 생성합니다.
GenerateSQLFromUserQuery AQS(고급 쿼리 구문) 또는 NQS(자연 쿼리 구문)로 표현된 클라이언트 제공 쿼리 문자열을 기반으로 SQL(구조적 쿼리 언어) 쿼리를 생성합니다.
GenerateWordForms 단어에 대한 대체 양식을 생성하고 이러한 양식을 IWordFormSink 개체에 넣습니다.

get_AuthenticationType 이 검색 루트 아래의 URL에 액세스하는 데 필요한 인증 유형을 검색합니다.
get_BypassList 인덱서에서 프록시 바이패스 목록을 가져옵니다. 이 목록은 로컬 항목 또는 URL을 결정하는 데 사용되며 프록시 서버를 통과할 필요가 없습니다. 이 목록은 ISearchManager::SetProxy를 호출하여 설정합니다.
get_ConnectionString 창 검색 인덱스용 OLE DB 연결 문자열을 반환합니다.
get_ConnectTimeout 인덱싱을 위해 저장소에 연결하기 위한 연결 제한 시간 값을 가져옵니다.
get_DataTimeout 인덱서와 검색 필터 호스트 간의 데이터 트랜잭션에 대한 데이터 제한 시간 값(초)을 가져옵니다. 이 값은 TIMEOUT_INFO 구조체에 포함됩니다.
get_DiacriticSensitivity 카탈로그가 발음에 중요한지 여부를 나타내는 값을 가져옵니다. 분음 부호는 특수 윗주 값 또는 발음을 나타내기 위해 문자에 추가된 표시입니다.
get_EnumerationDepth 이 검색 루트의 열거형 깊이를 가져옵니다.
get_FollowDirectories 검색 엔진이 하위 디렉터리 및 계층적 범위를 따르는지 여부를 나타내는 BOOL 값을 가져옵니다.
get_FollowFlags 지원되지 않습니다. 이 메서드는 호출 시 E_InvalidArg 반환합니다.
get_HostDepth 인덱싱할 때 크롤링할 호스트 트리의 정도를 나타내는 값을 가져옵니다.
get_IsDefault 기본 규칙인지 여부를 식별하는 값을 가져옵니다.

get_IsHierarchical 검색이 계층적 트리 구조에 루팅되는지 여부를 나타내는 값을 가져옵니다.
get_IsIncluded 이 규칙이 포함 규칙인지 여부를 식별하는 값을 가져옵니다. 포함 규칙은 크롤링 범위에 포함해야 하는 범위를 식별합니다.
get_LocalBypass 항목 또는 URL을 찾기 위해 프록시 서버를 바이패스해야 하는지 여부를 결정하는 값을 검색합니다.
get_Name 현재 카탈로그의 이름을 가져옵니다.
get_Password 구현되지 않았습니다. (ISearchRoot.get_Password)
get_PatternOrURL 규칙의 패턴 또는 URL을 가져옵니다. 범위 규칙은 포함하거나 제외할 URL 또는 경로를 결정합니다.
get_PortNumber 프록시 서버와 통신하는 데 사용되는 포트 번호를 검색합니다. 이 포트 번호는 인덱서에 저장되며 ISearchManager::SetProxy 메서드에 의해 설정됩니다.
get_ProvidesNotifications 검색 엔진이 검색 루트 아래의 URL 변경 내용에 대해 (프로토콜 처리기 또는 다른 애플리케이션에 의해) 알림을 받는지 여부를 나타내는 값을 가져옵니다.
get_ProxyName 프로토콜 처리기에서 사용할 프록시 이름을 검색합니다.
get_QueryContentLocale 쿼리에 대한 LCID(언어 코드 식별자)를 가져옵니다.
get_QueryContentProperties 검색 용어가 속성을 명시적으로 지정하지 않는 경우 쿼리에 포함된 속성 목록을 가져옵니다.

get_QueryKeywordLocale AQS(고급 쿼리 구문) 키워드를 구문 분석할 때 사용할 로캘의 LCID(언어 코드 식별자)를 가져옵니다.
get_QueryMaxResults 쿼리에서 반환할 최대 결과 수를 가져옵니다.
get_QuerySelectColumns 쿼리의 SELECT 문에서 요청된 열(또는 속성)을 가져옵니다.
get_QuerySorting 쿼리 결과 집합의 정렬 순서를 가져옵니다.
get_QuerySyntax 쿼리의 구문을 가져옵니다.
get_QueryTermExpansion 쿼리 용어를 확장하는 방법을 지정하는 값을 가져옵니다.
get_QueryWhereRestrictions WHERE 절의 쿼리에 추가된 제한을 가져옵니다.
get_RootURL 이 검색 루트의 시작점 URL을 가져옵니다.
get_Schedule 구현되지 않았습니다. (ISearchRoot.get_Schedule)
get_UseNotificationsOnly 이 검색 루트를 알림으로만 인덱싱하고 크롤링하지 않아야 하는지 여부를 나타내는 값을 가져옵니다.
get_UseProxy 사용할 프록시 서버를 검색합니다.
get_User 구현되지 않았습니다. (ISearchRoot.get_User)

get_UserAgent 사용자 에이전트 문자열을 검색합니다.
GetCatalog 이름으로 카탈로그를 검색하고 해당 카탈로그에 대한 새 ISearchCatalogManager 개체를 만듭니다.
GetCatalogStatus 카탈로그의 상태를 가져옵니다.
GetCLSID 처리 중인 URL 항목의 문서 형식에 대한 CLSID를 가져옵니다.
GetCodePage URL 항목의 속성에 대한 코드 페이지를 가져옵니다.
GetColumnMapper 열 매퍼 개체를 검색합니다.
GetComparisonInfo 리프 검색 조건 노드에서 속성 이름, 작업 및 값을 검색합니다. (ICondition.GetComparisonInfo)
GetConditionType 이 검색 조건 노드의 조건 유형을 검색하여 논리적 AND 또는 OR 또는 NOT 또는 리프 노드로 식별합니다.
GetCrawlScopeManager 이 검색 카탈로그에 대한 ISearchCrawlScopeManager 인터페이스를 가져옵니다.
GetData 데이터 청크를 나타내는 PROPVARIANT 및 입력 문자열을 검색합니다.
GetData IEntity, IRelationship 또는 ISchemaProvider 개체의 메타데이터에서 하나의 키/값 쌍을 검색합니다.
GetDefaultCatalog 시스템의 기본 카탈로그를 결정합니다.

GetDiacriticSensitivity 구현된 ISearchLanguageSupport 인터페이스에서 발음에 대한 민감도를 가져옵니다. 분음 부호는 특수 윗주 값 또는 발음을 나타내기 위해 문자에 추가된 악센트 표시입니다.
GetDisplayUrl URL 항목에 대한 사용자에게 친숙한 경로를 가져옵니다.
GetDocFormat MIME(다목적 인터넷 메일 확장) 문자열로 표시되는 문서 형식을 가져옵니다.
GetEntity 로드된 스키마에서 이름으로 엔터티를 검색합니다.
GetErrors IQuerySolution 조건 트리를 생성할 때 파서가 인식하지 못했거나 사용하지 않은 입력 문자열의 부분을 식별합니다.
GetFileName 필터 호스트가 인덱싱에 사용하는 항목의 파일 이름을 검색합니다. 항목이 파일 시스템에 존재하지 않고 IUrlAccessor::BindToStream 메서드가 구현된 경우 이 메서드는 항목에 대한 셸의 System.ParsingPath 속성을 반환합니다.
GetFilter 제공된 매개 변수에 대한 적절한 IFilteraccording을 검색합니다.
GetHost 해당하는 경우 콘텐츠 원본의 호스트 이름을 가져옵니다.
GetImpersonationSidBlobs 지정된 URL에 대한 SID(사용자 보안 식별자) 배열을 검색합니다. 이 메서드를 사용하면 프로토콜 처리기가 파일에 액세스할 수 있는 사용자와 검색 프로토콜 호스트를 지정하여 파일을 인덱싱하기 위해 사용자를 가장할 수 있습니다.
GetIndexerVersion 현재 인덱서의 버전을 두 청크로 검색합니다. _the 주 버전 기호 및 부 버전 기호입니다.
GetIndexerVersionStr 현재 인덱서의 버전을 단일 문자열로 검색합니다.

GetInputTerms 리프 노드의 경우 ICondition::GetInputTerms는 입력 문자열의 속성, 작업 및 검색 조건 노드의 값을 생성한 부분(또는 범위)에 대한 정보를 검색합니다.
GetItemsChangedSink 변경 알림 싱크 인터페이스를 가져옵니다.
GetLastModified URL이 마지막으로 수정된 시기를 식별하는 타임스탬프를 가져옵니다.
GetLeafConditionInfo 리프 검색 조건 노드에서 속성 이름, 작업 및 값을 검색합니다. (ICondition2.GetLeafConditionInfo)
GetLexicalData 쿼리 문자열, 토큰화된 방법 및 구문 분석하는 데 사용된 LCID(언어 코드 식별자) 및 단어 분리기를 보고합니다.
GetLicenseToUse 이 IStemmer 구현에 대한 라이선스 정보를 가져옵니다.
GetLicenseToUse IWordBreaker 인터페이스의 이 구현에 대한 라이선스 정보에 대한 포인터를 가져옵니다.
GetLimits 간격의 하한 및 상한을 지정합니다. 각 제한은 무한 또는 특정 값일 수 있습니다.
GetLocale 리프 검색 조건 노드에서 속성 이름, 작업 및 값을 검색합니다. (ICondition2.GetLocale)
GetNamedEntity 엔터티 이름을 기반으로 INamedEntity 개체를 검색합니다.
GetOplockEventHandle OpLock(기회 잠금)의 이벤트 핸들을 가져옵니다. 이벤트 개체는 OpLock이 끊어지면 신호 상태로 설정되므로 인덱서가 기본 IUrlAccessor 개체에 대한 모든 작업을 중지할 수 있습니다.

GetOption 이 쿼리 파서에 대해 지정된 단순 옵션 값을 검색합니다.
GetParameter 구현되지 않았습니다. (ISearchCatalogManager.GetParameter)
GetParameter 지원되지 않습니다. 이 메서드는 호출할 때 E_INVALIDARG 반환합니다. (ISearchManager.GetParameter)
GetParentScopeVersionId 부모 포함 URL의 버전 ID를 가져옵니다.
GetPersistentItemsChangedSink 클라이언트에 대한 변경 알림 이벤트 싱크 인터페이스를 가져옵니다. 이 메서드는 클라이언트 애플리케이션 및 프로토콜 처리기에서 인덱서에 변경 내용을 알리는 데 사용됩니다.
GetPropInfoFromId DBID에서 속성 정보를 가져옵니다.
GetPropInfoFromName 이름에서 속성 정보를 가져옵니다. 그러면 매개 변수 ppPropId에 DBID 포인터가 반환되며, 호출 수신자(이 클래스)가 아닌 호출자가 해제해야 합니다.
GetQuery 솔루션의 조건 트리 및 의미 체계 형식을 검색합니다.
GetQueryHelper 현재 카탈로그에 대한 ISearchQueryHelper 인터페이스를 가져옵니다.
GetRedirectedURL 현재 항목의 리디렉션된 URL을 가져옵니다.
GetRelationship 이름으로 요청된 대로 이 엔터티에 대한 IRelationship 개체를 검색합니다.

GetSchemaProvider 현재 로드된 스키마를 검색하기 위한 스키마 공급자를 검색합니다.
GetScopePriority 이 쿼리에서 지정한 범위에 대한 현재 인덱서 우선 순위 수준을 검색합니다.
GetScopeStatistics 이 쿼리에서 지정한 범위를 설명하는 정보를 가져옵니다.
GetSecurityDescriptor URL 항목의 보안 설명자를 가져옵니다. 보안은 쿼리 시간에 적용되므로 이 설명자는 읽기 액세스에 대한 보안을 식별합니다.
GetSecurityProvider URL에 대한 보안 공급자를 가져옵니다.
GetSize URL로 지정된 콘텐츠의 크기를 가져옵니다.
GetSubConditions 검색 조건 노드의 하위 조건 컬렉션과 컬렉션을 열거하기 위한 인터페이스의 IID를 검색합니다.
GetToken 개별 토큰의 위치, 길이 및 재정의 문자열을 검색합니다.
GetURLIndexingState 구현되지 않았습니다. (ISearchCatalogManager.GetURLIndexingState)
GetValue 이 명명된 엔터티의 값을 문자열로 검색합니다.
GetValueNormalization 검색 조건 노드의 문자 정규화된 값을 검색합니다.
GetValueType 검색 조건 노드 값의 의미 체계 형식을 검색합니다.

GetVersion 파일 매핑이 호출 프로세스의 주소 공간에 매핑되도록 하고 CSM(크롤링 범위 관리자)의 상태가 변경되었는지 클라이언트에 알립니다.
HasChildScopeRule 지정된 URL에 범위에 자식 규칙이 있는지 여부를 식별합니다.
HasParentScopeRule 지정된 URL의 범위에 부모 규칙이 있는지 여부를 식별합니다.
IncludedInCrawlScope 지정된 URL이 크롤링 범위에 포함되어 있는지 여부를 나타내는 표시기를 검색합니다.
IncludedInCrawlScopeEx 지정된 URL이 크롤링 범위에 포함되는지 여부와 이유에 대한 표시기를 검색합니다.
Init 형태소 분석기를 초기화합니다.
Init IWordBreaker 구현을 초기화하고 구성 요소가 작동하는 모드를 나타냅니다.
Init 프로토콜 처리기를 초기화합니다.
초기화 인터페이스의 모든 상태를 기본값으로 다시 설정하며 스키마에서 필요한 정보를 검색합니다.
InitializeOptions 지정된 쿼리 파서의 NQS(자연 쿼리 구문) 및 자동 와일드카드 문자에 대한 플래그를 설정합니다.
IsDirectory 항목 URL이 디렉터리를 가리키는지 여부를 확인합니다.
IsDocument 항목 URL이 문서인지 디렉터리인지 여부를 확인합니다.

IsMapUpToDate 맵이 최신 상태인지 여부를 확인합니다.
IsOplockBroken 인덱싱되는 항목의 OpLock(기회 잠금) 상태를 확인합니다. (IOpLockStatus.IsOplockBroken)
IsOplockValid 인덱싱되는 항목의 OpLock(기회 잠금) 상태를 확인합니다. (IOpLockStatus.IsOplockValid)
IsPrefixNormalized 쿼리 토큰이 문서 토큰의 접두사인지 여부를 결정하며 대/소문자, 너비 및(선택 사항) 발음을 무시합니다.
Isreal 관계가 실제 관계인지 여부를 보고합니다.
LoadIFilter 셀 데이터 원본에 매핑된 가장 적합한 필터를 검색하고 로드합니다.
LoadIFilterFromStorage do not use:_this 메서드가 구현되지 않았습니다. (ILoadFilter.LoadIFilterFromStorage)
LoadIFilterFromStream do not use:_this 메서드가 구현되지 않았습니다. (ILoadFilter.LoadIFilterFromStream)
LoadStemmer 지정된 LCID(언어 코드 식별자)에 등록된 형태소 분석기 단어에 대한 인터페이스를 검색합니다.
LoadWordBreaker 지정된 LCID(언어 코드 식별자)에 등록된 단어 분리기에 대한 인터페이스를 검색합니다.
지역화 입력 문자열에서 키워드를 지역화합니다.
지역화 지정된 로캘에 대해 현재 로드된 스키마를 지역화합니다.

LookupAuthoredNamedEntity 토큰화된 문자열에서 지정된 형식의 명명된 엔터티를 찾고 엔터티 값과 엔터티 값이 차지하는 토큰 수를 반환합니다.
MakeAndOr 하위 조건 컬렉션의 AND(논리 결합) 또는 OR(분리)인 조건 노드를 만듭니다.
MakeLeaf 속성 값과 상수 값의 비교를 나타내는 리프 조건 노드를 만듭니다.
MakeNot 다른 조건(이 노드의 하위 노드)의 논리 부정(NOT)인 조건 노드를 만듭니다. (IConditionFactory.MakeNot)
MetaData 이 엔터티에 대한 IMetaData 개체의 열거형을 검색합니다.
MetaData 이 관계에 대한 IMetaData 개체의 열거형을 검색합니다.
MetaData 로드된 스키마에 대한 전역 IMetaData 개체의 열거형을 검색합니다.
이름 이 엔터티의 이름을 검색합니다.
이름 관계의 이름을 검색합니다.
NamedEntities 이 형식의 알려진 각 명명된 엔터티에 대해 하나씩 INamedEntity 개체의 열거형을 검색합니다.
다음 지정된 수의 ISearchRoot 요소를 검색합니다.
다음 지정된 수의 ISearchScopeRule 요소를 검색합니다.

NumberOfItems 카탈로그의 항목 수를 가져옵니다.
NumberOfItemsToIndex 카탈로그 내에서 인덱싱할 항목 수를 가져옵니다.
NumberOfTokens 컬렉션의 토큰 수를 검색합니다.
OnCatalogStatusChange 카탈로그 상태가 변경되면 클라이언트에 알리기 위해 검색 서비스에서 호출됩니다.
Onchange 구현되지 않았습니다. (ISearchViewChangedSink.OnChange)
OnChangedItem 항목이 수정되었음을 클라이언트에 알리기 위해 인덱서에서 호출됩니다. 이 항목이 클라이언트 행 집합에 대한 조건의 일부(또는 전체)와 일치했을 수 있습니다.
OnDeletedItem 항목이 삭제되었음을 클라이언트에 알리기 위해 인덱서에서 호출됩니다. 이 항목이 클라이언트 행 집합에 대한 검색 조건의 일부(또는 전체)와 일치했을 수 있습니다.
OnItemIndexedStatusChange 특정 문서 또는 항목의 상태가 변경되면 클라이언트에 알리기 위해 검색 서비스에서 호출됩니다.
OnItemsChanged 변경된 일부 항목을 다시 인덱싱하도록 인덱서에 알리려면 이 메서드를 호출합니다.
OnItemsChanged 인덱서에 변경된 항목을 인덱싱하도록 알 수 있습니다.
OnNewItem 클라이언트 행 집합에 대한 조건의 일부(또는 전체)와 일치할 수 있는 새 항목을 클라이언트에 알리기 위해 인덱서에서 호출됩니다.

OnRowsetEvent 클라이언트 행 집합과 관련된 이벤트를 클라이언트에 알리기 위해 인덱서에서 호출됩니다.
Parse 구조적 쿼리 키워드 및/또는 내용이 포함된 입력 문자열을 구문 분석하여 IQuerySolution 개체를 생성합니다.
ParsePropertyValue 지정된 속성의 조건을 구문 분석합니다.
PrioritizeMatchingURLs 지정된 패턴과 일치하는 URL이 있는 인덱싱 항목에 더 높은 우선 순위를 부여하도록 인덱서에 지시합니다. 그러면 이러한 항목의 우선 순위가 다른 인덱싱 작업보다 높습니다.
put_AuthenticationType 이 검색 루트 아래의 URL에 액세스하는 데 필요한 인증 유형을 설정합니다.
put_ConnectTimeout TIMEOUT_INFO 구조체의 연결 제한 시간 값을 초 단위로 설정합니다.
put_DataTimeout 인덱서와 검색 필터 호스트 간의 데이터 트랜잭션에 대한 제한 시간 값을 설정합니다. 이 정보는 TIMEOUT_INFO 구조에 저장되며 몇 초 단위로 측정됩니다.
put_DiacriticSensitivity 카탈로그가 발음에 중요한지 여부를 결정하는 값을 설정합니다. 분음 부호는 특수한 윗주 값 또는 발음을 나타내기 위해 문자에 추가된 표시입니다.
put_EnumerationDepth 이 검색 루트의 열거형 깊이를 설정합니다.
put_FollowDirectories 검색 엔진이 이 검색 루트에 대한 하위 디렉터리 및 계층적 범위를 따라야 하는지 여부를 나타내는 BOOL 값을 설정합니다.
put_HostDepth 인덱싱할 때 크롤링할 호스트 트리의 정도를 나타내는 값을 설정합니다.

put_IsHierarchical 검색이 계층적 트리 구조에 루팅되는지 여부를 나타내는 값을 설정합니다.
put_Password 구현되지 않았습니다. (ISearchRoot.put_Password)
put_ProvidesNotifications 검색 엔진에 검색 루트 아래의 URL 변경 내용에 대한 알림(프로토콜 처리기 또는 기타 애플리케이션)을 나타내는 값을 설정합니다.
put_QueryContentLocale 쿼리의 LCID(언어 코드 식별자)를 설정합니다.
put_QueryContentProperties 검색 용어가 속성을 명시적으로 지정하지 않는 경우 쿼리에 포함할 속성을 설정합니다.
put_QueryKeywordLocale AQS(고급 쿼리 구문) 키워드를 구문 분석할 때 사용할 로캘의 LCID(언어 코드 식별자)를 설정합니다.
put_QueryMaxResults 쿼리에서 반환할 최대 결과 수를 설정합니다.
put_QuerySelectColumns select 문에서 요청된 열(또는 속성)을 설정합니다.
put_QuerySorting 쿼리 결과 집합의 정렬 순서를 설정합니다.
put_QuerySyntax 쿼리의 구문을 설정합니다.
put_QueryTermExpansion 쿼리 용어를 확장하는 방법을 지정하는 값을 설정합니다.
put_QueryWhereRestrictions WHERE 절의 쿼리에 추가된 제한을 설정합니다.

put_RootURL 현재 검색 루트의 URL을 설정합니다.
put_Schedule 구현되지 않았습니다. (ISearchRoot.put_Schedule)
put_UseNotificationsOnly 이 검색 루트를 알림으로만 인덱싱하고 크롤링하지 않아야 하는지 여부를 나타내는 값을 설정합니다.
put_User 구현되지 않았습니다. (ISearchRoot.put_User)
put_UserAgent 사용자 에이전트가 웹 사이트 및 서비스에 전달하여 자신을 식별하는 사용자 에이전트 문자열을 설정합니다.
PutPhrase IPhraseSink 개체에 쿼리 시간 구를 넣습니다.
PutSmallPhrase WordBreaker용 IPhraseSink 개체에 작은 쿼리 시간 구를 넣습니다.
RecognizeNamedEntities 입력 문자열에서 명명된 엔터티를 식별하고 해당 엔터티를 포함하는 컬렉션을 만듭니다.
RegisterViewForNotification 구현되지 않았습니다. (ISearchCatalogManager.RegisterViewForNotification)
다시 인덱스 카탈로그의 모든 URL을 다시 인덱싱합니다.
ReindexMatching URL 제공된 패턴과 일치하는 모든 항목을 다시 인덱싱합니다. 이 메서드는 Windows 7 이전에 구현되지 않았습니다.

ReindexSearchRoot 지정된 루트에서 모든 URL을 다시 인덱싱합니다.
관계 이 엔터티가 가지고 있는 각 관계에 대해 하나씩 IRelationship 개체의 열거형을 검색합니다.
RemoveDefaultScopeRule 검색 엔진에서 기본 범위 규칙을 제거합니다.
RemoveRoot 검색 엔진에서 검색 루트를 제거합니다.
RemoveScopeRule 검색 엔진에서 범위 규칙을 제거합니다.
재설정 IEnumSearchRoots::Next에 대한 후속 호출이 처음부터 검색되도록 내부 카운터를 목록의 시작 부분으로 이동합니다.
재설정 IEnumSearchScopeRules::Next에 대한 후속 호출이 처음부터 검색되도록 내부 카운터를 목록의 시작 부분으로 이동합니다.
재설정 데이터베이스를 다시 빌드하고 전체 인덱싱을 수행하여 기본 카탈로그를 다시 설정합니다.
해결 조건 트리에서 다음과 같은 다양한 변환을 수행합니다._resolves 상대 날짜/시간 식이 있는 조건과 절대 날짜/시간이 있는 조건(VT_FILETIME). 인식된 다른 명명된 엔터티를 실제 값이 있는 조건 트리로 바꿉니다. 조건 트리를 간소화합니다. 가상 또는 복합 속성을 다른 속성의 OR 트리로 바꿉니다. 는 조건이 적용되지 않은 속성 키워드가 있는 쿼리에서 발생하는 조건 트리를 제거합니다.
ResolveCondition 조건 트리에서 다양한 변환을 수행하여 계산을 위해 확인된 조건을 수행합니다. 반환된 개체는 ICondition 및 ICondition2를 지원합니다.
RestatePropertyValueToString 조건에 대해 지정된 속성을 쿼리 문자열로 다시 지정합니다.

RestateToString 조건을 구조적 쿼리 문자열로 다시 지정합니다. 조건이 원래 쿼리 문자열을 구문 분석한 결과인 경우 해당 쿼리 문자열의 키워드가 매우 많이 사용됩니다. 그렇지 않은 경우 기본 키워드가 사용됩니다.
RevertToDefaultScopes 기본 범위로 되돌려 줍니다.
RootEntity 로드된 스키마의 루트 엔터티를 검색합니다.
SaveAll 검색 엔진에 모든 변경 내용을 커밋합니다.
SaveBinary 로드된 스키마를 지정된 경로에 스키마 이진 파일로 저장합니다.
SetDiacriticSensitivity 구현된 ISearchLanguageSupport 인터페이스가 발음에 중요한지 여부를 나타내는 값을 설정합니다. 분음 부호는 특수 윗주 값 또는 발음을 나타내기 위해 문자에 추가된 악센트 표시입니다.
SetExtensionClusion 구현되지 않았습니다. (ISearchCatalogManager.SetExtensionClusion)
SetMultiOption 입력 문자열을 구문 분석할 때 사용할 복잡한 옵션(예: 지정된 조건 생성기)을 설정합니다.
Setoption 입력 문자열을 구문 분석하기 위해 지정된 단어 분리기 같은 단일 옵션을 설정합니다.
Setoption 이 IQueryParserManager 개체의 단일 옵션을 변경합니다. 예를 들어 이 메서드는 로드할 스키마 이진 파일의 이름 또는 지역화된 스키마 이진 파일의 위치를 변경할 수 있습니다.
SetParameter 카탈로그의 이름/값 매개 변수를 설정합니다.

SetParameter 지원되지 않습니다. 이 메서드는 호출 시 E_INVALIDARG 반환합니다. (ISearchManager.SetParameter)
SetProxy 인덱서의 작동 방식을 결정하는 정보를 인덱서에 저장하고 프록시 서버와 통신합니다.
SetScopePriority 이 쿼리에서 지정한 범위의 현재 인덱서 우선 순위 수준을 설정합니다.
ShouldIndexItemContent 항목의 콘텐츠를 인덱싱해야 하는지 여부를 식별합니다.
ShouldIndexProperty 속성을 인덱싱해야 하는지 여부를 식별합니다.
종료 프로토콜 처리기를 종료합니다.
skip 지정된 요소 수를 건너뜁니다. (IEnumSearchRoots.Skip)
skip 지정된 요소 수를 건너뜁니다. (IEnumSearchScopeRules.Skip)
StartedMonitoringScope 인덱스 관리 알림 원본이 "모니터링되는 범위" 목록에 자신을 추가하도록 허용합니다.
StartedMonitoringScope 지정된 계층 범위 내의 항목에 대한 변경 내용을 모니터링하도록 인덱서에 알리기 위해 알림 공급자가 호출합니다.
StoppedMonitoringScope 구현되지 않았습니다. (ISearchItemsChangedSink.StoppedMonitoringScope)

StoppedMonitoringScope 지정된 계층 범위 내의 항목에 대한 변경 내용 모니터링을 중지하도록 인덱서에 알리기 위해 알림 공급자가 호출합니다.
ThreadIdle 프로토콜 처리기가 빌드되었을 수 있는 캐시를 정리할 수 있도록 필터링 스레드가 유휴 상태임을 프로토콜 처리기에 알릴 수 있습니다.
ThreadInit 프로토콜 처리기와 프로토콜 호스트 간의 통신을 초기화합니다.
ThreadShutdown 프로토콜 처리기에 스레드가 종료되고 있음을 알 수 있습니다.
UnregisterViewForNotification 구현되지 않았습니다. (ISearchCatalogManager.UnregisterViewForNotification)
URLBeingIndexed 현재 인덱싱 중인 URL을 가져옵니다. 현재 처리 중인 인덱싱이 없으면 pszUrl이 NULL로 설정됩니다.
VerifyCatalog 카탈로그 위치의 유효성을 검사합니다.
WriteProperties 구현되지 않았습니다. (ISearchQueryHelper.WriteProperties)

인터페이스

IColumnMapper 파일 기반 쿼리에 대한 속성 정보를 검색합니다.
IColumnMapperCreator 열 매퍼 개체를 검색하는 메서드를 포함합니다.

ICondition 검색 조건에 대한 정보를 검색하는 메서드를 제공합니다.
ICondition2 ICondition 인터페이스의 기능을 확장합니다. ICondition2는 검색 조건에 대한 정보를 검색하는 메서드를 제공합니다.
IConditionFactory 쿼리 문자열을 구문 분석하여 얻은 조건 트리를 만들거나 확인하는 메서드를 제공합니다.
IConditionFactory2 IConditionFactory의 기능을 확장합니다. IConditionFactory2는 쿼리 문자열을 구문 분석하여 얻은 조건 트리를 만들거나 확인하는 메서드를 제공합니다.
IConditionGenerator 명명된 엔터티를 처리하고 특수 조건을 생성하는 메서드를 제공합니다.
IEntity 스키마에서 엔터티 형식에 대한 정보를 검색하는 메서드를 제공합니다.
IEnumSearchRoots 카탈로그의 검색 루트(예: SystemIndex)를 열거하는 메서드를 제공합니다.
IEnumSearchScopeRules 범위 규칙을 열거합니다.
IInterval 간격의 제한을 가져오는 메서드를 제공합니다.
ILoadFilter 필터를 로드하는 메서드를 제공하는 FilterRegistration 개체에 의해 구현되는 메서드 및 속성을 정의합니다.
IMetaData IEntity, IRelationship 또는 ISchemaProvider 개체에서 문자열의 키/값 쌍을 검색하는 메서드를 제공합니다.

INamedEntity 명명된 엔터티 값의 값 또는 기본 구를 가져오는 메서드를 제공합니다.
INamedEntityCollector IConditionGenerator 개체로 식별되는 명명된 엔터티를 누적하는 메서드를 제공합니다.
IOpLockStatus 인덱싱하는 동안 항목에 대해 Microsoft WDS(Windows Desktop Search)에서 사용하는 기회 잠금을 확인하는 메서드를 제공합니다.
IProtocolHandlerSite 프로토콜 처리기의 IUrlAccessor 개체에 대한 메서드를 제공하여 URL 항목에 대한 적절한 필터에 대해 필터 디먼을 쿼리합니다.
IQueryParser 입력 문자열을 IQuerySolution 개체로 구문 분석하는 메서드를 제공합니다.
IQueryParserManager IQueryParser 개체에 대한 옵션을 만들고, 초기화하고, 변경하는 메서드를 제공합니다.
IQuerySolution 구문 분석된 쿼리의 해석에 대한 정보를 검색하는 메서드를 제공합니다.
IRelationship 스키마 속성에 대한 정보를 검색하는 메서드를 제공합니다.
IRichChunk 데이터 청크를 문자열 및 PROPVARIANT 값으로 나타냅니다.
IRowsetEvents 이벤트 알림을 수신하는 메서드를 노출합니다.
IRowsetPrioritization 이 쿼리에서 지정한 범위에 대한 현재 인덱서 우선 순위 수준을 설정하거나 검색합니다.
ISchemaLocalizerSupport 지정된 문자열에서 키워드를 지역화하는 메서드를 제공합니다.

ISchemaProvider 검색할 수 있는 스키마 리포지토리를 제공합니다.
ISearchCatalogManager 다시 인덱싱 또는 시간 제한 설정과 같은 목적으로 검색 카탈로그를 관리하는 메서드를 제공합니다.
ISearchCatalogManager2 ISearchCatalogManager 인터페이스를 확장하여 검색 카탈로그를 관리합니다(예: 다시 인덱싱 또는 시간 제한 설정).
ISearchCrawlScopeManager 크롤링 및/또는 감시할 컨테이너의 검색 엔진과 크롤링 또는 감시 시 포함하거나 제외할 해당 컨테이너 아래의 항목을 알리는 메서드를 제공합니다.
ISearchCrawlScopeManager2 ISearchCrawlScopeManager 인터페이스의 기능을 확장합니다.
ISearchItemsChangedSink 인덱싱된 항목의 변경 내용에 대한 알림을 제공합니다. 또한 변경된 항목에 대해 모니터링되는 계층적 범위에 대한 알림을 제공합니다.
ISearchLanguageSupport 동의어 사전 정보에 액세스하는 메서드를 제공합니다.
ISearchManager Search Service 제어하는 메서드를 제공합니다. 이 인터페이스는 카탈로그 전체에서 검색 엔진에 영향을 주는 설정 및 개체를 관리합니다.
ISearchManager2 애플리케이션이 Windows Search 인덱서에서 사용자 지정 카탈로그를 만들고 삭제할 수 있도록 설정했습니다.
ISearchNotifyInlineSite Search Service 카탈로그 및 인덱스 상태에 대한 업데이트를 알림 공급자에게 보내는 데 사용하는 메서드를 제공합니다.

ISearchPersistentItemsChangedSink 변경 알림을 전달하여 항목을 업데이트해야 하는 인덱서에 경고하는 메서드를 제공합니다.
ISearchProtocol IUrlAccessor 개체를 호출, 초기화 및 관리하는 메서드를 제공합니다. (ISearchProtocol)
ISearchProtocol2 IUrlAccessor 개체를 호출, 초기화 및 관리하는 메서드를 제공합니다. (ISearchProtocol2)
ISearchProtocolThreadContext 이 선택적 인터페이스를 사용하면 프로토콜 처리기가 프로토콜 호스트에서 필터링하는 데 사용되는 스레드에서 작업을 수행할 수 있습니다.
ISearchQueryHelper 사용자 입력에서 쿼리를 빌드하고, 쿼리를 Windows Search SQL로 변환하고, 연결 문자열을 가져와 창 검색 인덱스로의 연결을 초기화하는 메서드를 제공합니다.
ISearchRoot 검색 루트를 조작하는 메서드를 제공합니다. 속성 멤버에 대한 변경 내용은 검색 루트에 속하는 모든 URL에 적용됩니다. URL은 검색 루트 URL과 일치하거나 해당 URL의 계층적 자식인 경우 검색 루트 아래에 있습니다.
ISearchScopeRule 크롤링 및 인덱싱에 대한 범위 규칙을 정의하는 메서드를 제공합니다.
ISearchViewChangedSink 구현되지 않았습니다. (ISearchViewChangedSink)
ISimpleCommandCreator 파일 카탈로그와 상호 작용하는 메서드를 포함합니다.
IStemmer 언어별 형태소 분석기를 만드는 메서드를 제공합니다. 형태소 분석기는 지정된 단어의 굴절된 형태를 생성합니다.
ITokenCollection 단어 분리기를 사용하여 발생하는 토큰을 가져옵니다.

IUrlAccessor 필터 호스트에 대한 Gatherer에서 URL을 제공하는 콘텐츠 원본의 개별 항목을 처리하는 메서드를 제공합니다.
IUrlAccessor2 IUrlAccessor 인터페이스의 기능을 확장합니다.
IUrlAccessor3 IUrlAccessor3::GetImpersonationSidBlobs 메서드를 사용하여 IUrlAccessor2 인터페이스의 기능을 확장하여 지정된 URL에 대한 SID(사용자 보안 식별자)를 식별합니다.
IUrlAccessor4 항목의 콘텐츠를 인덱싱해야 하는지 여부를 식별하는 IUrlAccessor4::ShouldIndexItemContent 메서드를 사용하여 IUrlAccessor3 인터페이스의 기능을 확장합니다.
IWordBreaker 텍스트를 구문 분석하고 개별 단어와 구를 식별합니다. 이 인터페이스는 언어별 언어 리소스 구성 요소입니다. 백그라운드 프로세스에서 사용되며 처리량과 최소한의 리소스 사용 모두에 최적화되어야 합니다.
IWordFormSink 쿼리 시간 동안 형태소 분석기가 생성하는 대체 단어 양식 목록을 처리합니다.

구조체

AUTHENTICATION_INFO 콘텐츠 액세스에 대한 보안 인증 정보를 설명합니다.
FILTERED_DATA_SOURCES 필터가 로드되는 셸 데이터 원본에 대한 매개 변수를 지정합니다.
HITRANGE 쿼리 검색 조건이 인덱싱된 데이터와 일치하는 경우 일치하는 데이터의 범위를 식별합니다.

INCREMENTAL_ACCESS_INFO 마지막 액세스 날짜 및 수정 시간과 같은 증분 크롤링에서 사용하는 액세스 정보를 포함합니다.
ITEM_INFO 현재 항목에 대한 IUrlAccessor 개체에 전달된 정보를 포함합니다. 예를 들어 애플리케이션 이름 및 카탈로그 이름입니다.
ITEMPROP Windows 속성 시스템의 속성에 대한 정보를 저장하고 IItemPropertyBag 인터페이스에서 사용됩니다.
PROXY_INFO 프록시에 대한 정보를 저장합니다. ISearchProtocol에서 사용됩니다.
SEARCH_COLUMN_PROPERTIES 이 구조는 구현되지 않습니다.
SEARCH_ITEM_CHANGE 인덱싱된 항목의 변경 내용을 지정합니다.
SEARCH_ITEM_INDEXING_STATUS 인덱싱할 문서의 상태를 설명합니다.
SEARCH_ITEM_PERSISTENT_CHANGE 인덱싱할 항목에서 발생한 변경의 종류에 대한 정보를 포함합니다. 이 구조는 ISearchPersistentItemsChangedSink::OnItemsChanged 메서드와 함께 사용하여 변경된 내용에 대한 정보를 인덱서에 전달합니다.
TEXT_SOURCE 단어 분리기가 처리할 텍스트에 대한 정보를 포함합니다.
TIMEOUT_INFO 연결 및 데이터에 대한 제한 시간 값을 저장합니다.

Windows Server 백업

아티클2023. 08. 22.

Windows Server Backup 기술 개요입니다.

Windows Server Backup을 개발하려면 다음 헤더가 필요합니다.

- wsbapp.h
- wsbonline.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows Server 백업

열거형

WSB_OB_STATUS_ENTRY_PAIR_TYPE 매개 변수 값의 형식을 나타냅니다.

Functions

중단 불완전한 비동기 작업을 취소합니다.
CheckConsistency 백업할 볼륨에 대해 섀도 복사본을 만든 후 섀도 복사본에서 VSS 기록기 구성 요소의 일관성을 확인합니다.
DeregisterOnlineBackupFromWindowsServerBackup 이미 등록된 클라우드 백업 공급자를 등록 해제합니다.
IsRollForwardSupported 애플리케이션이 롤 포워드 복원을 지원하는지 여부를 보고합니다.
OrderComponents

애플리케이션 구성 요소를 복원할 순서를 지정합니다.
PostRestore 애플리케이션별 PostRestore 작업을 수행합니다.
PreRestore 애플리케이션별 PreRestore 작업을 수행합니다.
QueryStatus 비동기 작업의 상태 쿼리합니다.
RegisterOnlineBackupWithWindowsServerBackup Windows Server Backup에 클라우드 백업 공급자를 등록합니다.
UpdateOBStatusInWindowsServerBackup Windows Server Backup MMC 스냅인에서 클라우드 백업 공급자 상태 업데이트.

인터페이스

IWsbApplicationAsync 비동기 작업의 진행률을 모니터링하고 제어하는 메서드를 정의합니다.
IWsbApplicationBackupSupport 애플리케이션의 VSS 기록기 구성 요소의 일관성을 확인하는 메서드를 정의합니다.
IWsbApplicationRestoreSupport 애플리케이션별 복원 작업을 수행하기 위한 메서드를 정의합니다.

구조체

WSB_OB_REGISTRATION_INFO Windows Server Backup에 클라우드 백업 공급자를 등록하는 정보를 포함합니다.

WSB_OB_STATUS_ENTRY Windows Server Backup MMC 스냅인에 표시할 하나의 항목에 대한 상태 정보를 포함합니다.
WSB_OB_STATUS_ENTRY_VALUE_TYPE_PAIR 값 리소스 문자열을 확장하는 데 사용되는 매개 변수의 값 및 값 형식을 포함합니다.
WSB_OB_STATUS_INFO Windows Server Backup MMC 스냅인에서 클라우드 백업 공급자 상태 업데이트하는 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Windows 소켓 2

아티클2024. 04. 05.

Windows 소켓 2 기술에 대한 개요입니다.

Windows 소켓 2를 개발하려면 다음 헤더가 필요합니다.

- af_irda.h
- in6addr.h
- mstcpip.h
- mswsock.h
- mswsockdef.h
- nsemail.h
- nspapi.h
- socketapi.h
- sporder.h
- transportsettingcommon.h
- winsock.h
- ws2atm.h
- ws2spi.h
- ws2tcpip.h
- wsipv6ok.h
- wsnwlink.h
- wsrm.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 소켓 2

열거형

테이블 확장

CONTROL_CHANNEL_TRIGGER_STATUS Windows 스토어 앱에서 백그라운드 네트워크 알림을 수신하기 위해 ControlChannelTrigger와 함께 사용되는 TCP 소켓에 대한 REAL_TIME_NOTIFICATION_CAPABILITY 전송 설정에 대한 쿼리의 상태 지정합니다.
eWINDOW_ADVANCE_METHOD

eWINDOW_ADVANCE_METHOD 열거형은 신뢰할 수 있는 멀티캐스트에 사용되는 창 사전 모드를 지정합니다.
MULTICAST_MODE_TYPE 멀티캐스트 그룹 주소에 대한 필터 모드를 지정합니다.
NAPI_PROVIDER_LEVEL 지정된 도메인에 대한 NS_EMAIL 네임스페이스 공급자의 공급자 기관 수준을 지정합니다.
NAPI_PROVIDER_TYPE 네임스페이스 공급자에 대해 예상되는 호스팅 유형을 지정합니다.
RIO_NOTIFICATION_COMPLETION_TYPE Winsock 등록 I/O 확장을 사용하여 데이터를 보내거나 받을 때 RIONotify 함수에 사용할 완료 큐 알림 유형을 지정합니다.
SOCKET_SECURITY_PROTOCOL 네트워크 트래픽을 보호하기 위해 소켓에서 사용할 보안 프로토콜 유형을 나타냅니다.
SOCKET_USAGE_TYPE 소켓의 사용 유형을 지정하는 데 사용됩니다.
TCPSTATE TCP(Transmission Control Protocol) 연결의 가능한 상태를 나타냅니다.
WSAECOMPARATOR Windows 소켓 WSAECOMPARATOR 열거형 형식은 Windows 소켓 2의 버전 비교 의미 체계에 사용됩니다.
WSC_PROVIDER_INFO_TYPE 열거형 유형은 Windows 소켓 2에서 LSP(계층화된 서비스 프로토콜)의 정보 클래스를 지정하는 데 사용됩니다.

Functions

테이블 확장

__WSAFDIsSet __WSAFDIsSet 함수(winsock2.h)는 소켓이 소켓 설명자 집합에 포함되는지 여부를 지정합니다.
__WSAFDIsSet __WSAFDIsSet 함수(winsock.h)는 소켓이 소켓 설명자 집합에 포함되는지 여부를 지정합니다.
받아들일 accept 함수는 소켓에서 들어오는 연결 시도를 허용합니다.
AcceptEx AcceptEx 함수(mswsock.h)는 새 연결을 수락하고 로컬 및 원격 주소를 반환하며 클라이언트 애플리케이션에서 보낸 첫 번째 데이터 블록을 받습니다.
AcceptEx AcceptEx 함수(winsock.h)는 새 연결을 수락하고 로컬 및 원격 주소를 반환하며 클라이언트 애플리케이션에서 보낸 첫 번째 데이터 블록을 받습니다.
주장
bind bind 함수는 로컬 주소를 소켓과 연결합니다. (bind 함수(winsock2.h))
bind bind 함수(winsock.h)는 로컬 주소를 소켓과 연결합니다.
closesocket closesocket 함수는 기존 소켓을 닫습니다. (closesocket 함수(winsock2.h))
closesocket closesocket 함수(winsock.h)는 기존 소켓을 닫습니다.
connect connect 함수는 지정된 소켓에 대한 연결을 설정합니다.
EnumProtocolsA EnumProtocols 함수는 로컬 호스트에서 활성 상태인 지정된 네트워크 프로토콜 집합에 대한 정보를 검색합니다. (ANSI)

EnumProtocolsW EnumProtocols 함수는 로컬 호스트에서 활성 상태인 지정된 네트워크 프로토콜 집합에 대한 정보를 검색합니다. (유니코드)
FD_SET FD_SET 매크로(winsock2.h)는 Windows Sockets(Winsock) 함수 및 서비스 공급자가 소켓을 집합에 배치하는 데 사용됩니다.
FD_SET FD_SET 매크로(winsock.h)는 Windows 소켓(Winsock) 함수 및 서비스 공급자가 소켓을 세트에 배치하는 데 사용됩니다.
freeaddrinfo getaddrinfo 함수가 addrinfo 구조에서 동적으로 할당하는 주소 정보를 해제합니다.
FreeAddrInfoEx FreeAddrInfoEx 함수(ws2tcpip.h)는 GetAddrInfoEx 함수가 addrinfoex 구조에서 동적으로 할당하는 주소 정보를 해제합니다.
FreeAddrInfoExW FreeAddrInfoExW(유니코드) 함수(ws2tcpip.h)는 GetAddrInfoEx 함수가 addrinfoex 구조에 동적으로 할당하는 주소 정보를 해제합니다.
FreeAddrInfoW GetAddrInfoW 함수가 addrinfoW 구조에서 동적으로 할당하는 주소 정보를 해제합니다.
gai_strerrorA gai_strerror 함수는 getaddrinfo 함수에서 반환된 EAI_* 오류를 기반으로 오류 메시지를 인쇄하는 데 도움이 됩니다. (ANSI)
gai_strerrorW gai_strerror 함수는 getaddrinfo 함수에서 반환된 EAI_* 오류를 기반으로 오류 메시지를 인쇄하는 데 도움이 됩니다. (유니코드)
GetAcceptExSockaddrs GetAcceptExSockaddrs 함수(mswsock.h)는 AcceptEx 함수 호출에서 얻은 데이터를 구문 분석하고 로컬 및 원격 주소를 sockaddr 구조에 전달합니다.

GetAcceptExSockaddrs GetAcceptExSockaddrs 함수(winsock.h)는 AcceptEx 함수 호출에서 얻은 데이터를 구문 분석하고 로컬 및 원격 주소를 sockaddr 구조에 전달합니다.
GetAddressByNameA Windows 소켓 2를 기준으로 GetAddressByName을 더 이상 사용할 수 없습니다. (ANSI)
GetAddressByNameW Windows 소켓 2를 기준으로 GetAddressByName을 더 이상 사용할 수 없습니다. (유니코드)
getaddrinfo ANSI 호스트 이름에서 주소로 프로토콜 독립적 변환을 제공합니다.
GetAddrInfoExA 요청을 처리해야 하는 네임스페이스 공급자를 한정하기 위한 추가 매개 변수를 사용하여 프로토콜 독립적 이름 확인을 제공합니다. (ANSI)
GetAddrInfoExCancel GetAddrInfoEx 함수에 의해 비동기 작업을 취소합니다.
GetAddrInfoExOverlappedResult GetAddrInfoEx 함수에 대한 비동기 작업에서 사용하는 OVERLAPPED 구조체의 반환 코드를 가져옵니다.
GetAddrInfoExW 요청을 처리해야 하는 네임스페이스 공급자를 한정하기 위한 추가 매개 변수를 사용하여 프로토콜 독립적 이름 확인을 제공합니다. (유니코드)
GetAddrInfoW 유니코드 호스트 이름에서 주소로 프로토콜 독립적 변환을 제공합니다.
gethostbyaddr gethostbyaddr 함수는 네트워크 주소에 해당하는 호스트 정보를 검색합니다. (gethostbyaddr는 더 이상 권장되지 않으며 대신 getnameinfo를 사용합니다.)
gethostbyaddr Windows 소켓 2를 기준으로 gethostbyaddr 함수(winsock.h)는 더 이상 사용하지 않는 것이 좋습니다.

gethostbyaddr gethostbyaddr 매크로 함수(wsipv6ok.h)는 네트워크 주소에 해당하는 호스트 정보를 검색합니다.
gethostbyname gethostbyname 함수(winsock2.h) 함수는 호스트 데이터베이스에서 호스트 이름에 해당하는 호스트 정보를 검색합니다.
gethostbyname gethostbyname 함수(winsock.h)는 호스트 데이터베이스에서 호스트 이름에 해당하는 호스트 정보를 검색합니다.
gethostbyname gethostbyname 매크로 함수(wsipv6ok.h)는 호스트 데이터베이스에서 호스트 이름에 해당하는 호스트 정보를 검색합니다.
gethostname gethostname 함수(winsock2.h)는 로컬 컴퓨터의 표준 호스트 이름을 검색합니다.
gethostname gethostname 함수(winsock.h)는 로컬 컴퓨터의 표준 호스트 이름을 검색합니다.
GetHostNameW GetHostNameW 함수는 로컬 컴퓨터의 표준 호스트 이름을 유니코드 문자열로 검색합니다.
getipv4sourcefilter IPv4 소켓의 멀티캐스트 필터 상태를 검색합니다.
GetNameByTypeA GetNameByType 함수는 지정된 서비스 유형에 대한 네트워크 서비스의 이름을 검색합니다. (ANSI)
GetNameByTypeW GetNameByType 함수는 지정된 서비스 유형에 대한 네트워크 서비스의 이름을 검색합니다. (유니코드)
getnameinfo 주소에서 ANSI 호스트 이름으로, 포트 번호에서 ANSI 서비스 이름으로 프로토콜 독립적 이름 확인을 제공합니다.

GetNameInfoW 주소에서 유니코드 호스트 이름으로, 포트 번호에서 유니코드 서비스 이름으로 프로토콜 독립적 이름 확인을 제공합니다.
getpeername getpeername 함수(winsock2.h)는 소켓이 연결된 피어의 주소를 검색합니다.
getpeername getpeername 함수(winsock.h)는 소켓이 연결된 피어의 주소를 검색합니다.
getprotobyname getprotobyname 함수(winsock2.h)는 프로토콜 이름에 해당하는 프로토콜 정보를 검색합니다.
getprotobyname getprotobyname 함수(winsock.h)는 프로토콜 이름에 해당하는 프로토콜 정보를 검색합니다.
getprotobynumber getprotobynumber 함수(winsock2.h)는 프로토콜 번호에 해당하는 프로토콜 정보를 검색합니다.
getprotobynumber getprotobynumber 함수(winsock.h)는 프로토콜 번호에 해당하는 프로토콜 정보를 검색합니다.
getservbyname getservbyname 함수(winsock2.h)는 서비스 이름 및 프로토콜에 해당하는 서비스 정보를 검색합니다.
getservbyname getservbyname 함수(winsock.h)는 서비스 이름 및 프로토콜에 해당하는 서비스 정보를 검색합니다.
getservbyport getservbyport 함수(winsock2.h)는 포트 및 프로토콜에 해당하는 서비스 정보를 검색합니다.
getservbyport getservbyport 함수(winsock.h)는 포트 및 프로토콜에 해당하는 서비스 정보를 검색합니다.
GetServiceA

GetService 함수는 기본 네임스페이스 집합 또는 지정된 네임스페이스의 컨텍스트에서 네트워크 서비스에 대한 정보를 검색합니다. (ANSI)
GetServiceW GetService 함수는 기본 네임스페이스 집합 또는 지정된 네임스페이스의 컨텍스트에서 네트워크 서비스에 대한 정보를 검색합니다. (유니코드)
getsockname getsockname 함수(winsock2.h)는 지정된 소켓 설명자의 로컬 이름을 검색합니다.
getsockname getsockname 함수(winsock.h)는 소켓의 로컬 이름을 검색합니다.
getsockopt getsockopt 함수는 모든 형식의 소켓과 연결된 소켓 옵션의 현재 값을 모든 상태로 검색하고 결과를 optval에 저장합니다.
getsockopt getsockopt 함수(winsock.h)는 소켓 옵션을 검색합니다.
getsourcefilter IPv4 또는 IPv6 소켓의 멀티캐스트 필터 상태를 검색합니다.
GetTypeByNameA GetTypeByName 함수는 이름으로 지정된 네트워크 서비스에 대한 서비스 유형 GUID를 검색합니다. (ANSI)
GetTypeByNameW GetTypeByName 함수는 이름으로 지정된 네트워크 서비스에 대한 서비스 유형 GUID를 검색합니다. (유니코드)
htond double을 호스트에서 TCP/IP 네트워크 바이트 순서(big-endian)로 변환합니다.
htonf float를 호스트에서 TCP/IP 네트워크 바이트 순서(big-endian)로 변환합니다.
htonl

htonl 함수(winsock2.h)는 u_long 호스트에서 TCP/IP 네트워크 바이트 순서(big-endian)로 변환합니다.
htonl htonl 함수(winsock.h)는 u_long 호스트에서 TCP/IP 네트워크 바이트 순서(big-endian)로 변환합니다.
htonll 서명되지 않은 __int64 호스트에서 TCP/IP 네트워크 바이트 순서(big-endian)로 변환합니다.
htons htons 함수(winsock2.h)는 u_short 호스트에서 TCP/IP 네트워크 바이트 순서(big-endian)로 변환합니다.
htons htons 함수(winsock.h)는 u_short 호스트에서 TCP/IP 네트워크 바이트 순서(big-endian)로 변환합니다.
IN4_ADDR_EQUAL
IN4_CLASSA
IN4_CLASSB
IN4_CLASSC
IN4_CLASSD
IN4_IS_ADDR_6TO4ELIGIBLE
IN4_IS_ADDR_BROADCAST
IN4_IS_ADDR_LINKLOCAL
IN4_IS_ADDR_LOOPBACK

IN4_IS_ADDR_MC_ADMINLOCAL
IN4_IS_ADDR_MC_LINKLOCAL
IN4_IS_ADDR_MC_SITELOCAL
IN4_IS_ADDR_MULTICAST
IN4_IS_ADDR_RFC1918
IN4_IS_ADDR_SITELOCAL
IN4_IS_ADDR_UNSPECIFIED
IN4_IS_UNALIGNED_ADDR_6TO4ELIGIBLE
IN4_IS_UNALIGNED_ADDR_BROADCAST
IN4_IS_UNALIGNED_ADDR_LINKLOCAL
IN4_IS_UNALIGNED_ADDR_LOOPBACK
IN4_IS_UNALIGNED_ADDR_MULTICAST
IN4_IS_UNALIGNED_ADDR_RFC1918
IN4_IS_UNALIGNED_ADDR_UNSPECIFIED
IN4_MULTICAST
IN4_UNALIGNED_ADDR_EQUAL
IN4_UNCANONICALIZE_SCOPE_ID

IN4ADDR_ISANY
IN4ADDR_ISEQUAL
IN4ADDR_ISLOOPBACK
IN4ADDR_ISUNSPECIFIED
IN4ADDR_SCOPE_ID
IN4ADDR_SETANY
IN4ADDR_SETLOOPBACK
IN4ADDR_SETSOCKADDR
IN6_ADDR_EQUAL
IN6_EXTRACT_V4ADDR_FROM_6TO4
IN6_EXTRACT_V4ADDR_FROM_ISATAP
IN6_GET_ADDR_V4COMPAT
IN6_GET_ADDR_V4MAPPED
IN6_IS_ADDR_6TO4
IN6_IS_ADDR_ALLNODESONLINK
IN6_IS_ADDR_ALLNODESONNODE
IN6_IS_ADDR_ALLROUTERSONLINK

IN6_IS_ADDR_ANYCAST
IN6_IS_ADDR_EUI64
IN6_IS_ADDR_GLOBAL
IN6_IS_ADDR_ISATAP
IN6_IS_ADDR_LINKLOCAL
IN6_IS_ADDR_LOOPBACK
IN6_IS_ADDR_MC_GLOBAL
IN6_IS_ADDR_MC_LINKLOCAL
IN6_IS_ADDR_MC_NODELOCAL
IN6_IS_ADDR_MC_ORGLOCAL
IN6_IS_ADDR_MC_SITELOCAL
IN6_IS_ADDR_MULTICAST
IN6_IS_ADDR_SITELOCAL
IN6_IS_ADDR_SOLICITEDNODE
IN6_IS_ADDR_SUBNET_RESERVED_ANYCAST
IN6_IS_ADDR_SUBNET_ROUTER_ANYCAST
IN6_IS_ADDR_TEREDO

IN6_IS_ADDR_UNSPECIFIED
IN6_IS_ADDR_V4COMPAT
IN6_IS_ADDR_V4MAPPED
IN6_IS_ADDR_V4TRANSLATED
IN6_MULTICAST_SCOPE
IN6_PREFIX_EQUAL
IN6_SET_6TO4_PREFIX
IN6_SET_ADDR_LOOPBACK
IN6_SET_ADDR_SOLICITEDNODE
IN6_SET_ADDR_UNSPECIFIED
IN6_SET_ADDR_V4COMPAT
IN6_SET_ADDR_V4MAPPED
IN6_SET_ISATAP_IDENTIFIER
IN6_UNCANONICALIZE_SCOPE_ID
IN6ADDR_IS6TO4
IN6ADDR_ISANY
IN6ADDR_ISEQUAL

IN6ADDR_ISISATAP
IN6ADDR_ISLOOPBACK
IN6ADDR_ISTEREDO
IN6ADDR_ISUNSPECIFIED
IN6ADDR_ISV4MAPPED
IN6ADDR_SETANY
IN6ADDR_SETLOOPBACK
IN6ADDR_SETSOCKADDR
IN6ADDR_SETV4MAPPED
inet_addr inet_addr 함수(winsock2.h)는 IPv4 점선 소수점 주소가 포함된 문자열을 IN_ADDR 구조체에 적합한 주소로 변환합니다.
inet_addr inet_addr 함수(winsock.h)는 IPv4 점선 소수점 주소가 포함된 문자열을 IN_ADDR 구조체에 적합한 주소로 변환합니다.
inet_addr inet_addr 매크로 함수(wsipv6ok.h)는 IPv4 점선 소수점 주소가 포함된 문자열을 IN_ADDR 구조체에 적합한 주소로 변환합니다.
inet_ntoa inet_ntoa 함수(winsock2.h)는 (Ipv4) 인터넷 네트워크 주소를 인터넷 표준 점선 10진수 형식의 ASCII 문자열로 변환합니다.
inet_ntoa

inet_ntoa 함수(winsock.h)는 (Ipv4) 인터넷 네트워크 주소를 인터넷 표준 점선 10진수 형식의 ASCII 문자열로 변환합니다.
inet_ntoa inet_ntoa 매크로 함수(wsipv6ok.h)는 (Ipv4) 인터넷 네트워크 주소를 인터넷 표준 점선 10진수 형식의 ASCII 문자열로 변환합니다.
inet_ntop InetNtop 함수는 IPv4 또는 IPv6 인터넷 네트워크 주소를 인터넷 표준 형식의 문자열로 변환합니다. 이 함수의 ANSI 버전은 inet_ntop. (InetNtopA 또는 inet_ntop)
inet_pton InetPton 함수는 표준 텍스트 프레젠테이션 형식의 IPv4 또는 IPv6 인터넷 네트워크 주소를 숫자 이진 형식으로 변환합니다. 이 함수의 ANSI 버전은 inet_pton. (InetPtonA 또는 inet_pton)
InetNtopW InetNtop 함수는 IPv4 또는 IPv6 인터넷 네트워크 주소를 인터넷 표준 형식의 문자열로 변환합니다. 이 함수의 ANSI 버전은 inet_ntop. (InetNtopW)
InetPtonW InetPton 함수는 표준 텍스트 프레젠테이션 형식의 IPv4 또는 IPv6 인터넷 네트워크 주소를 숫자 이진 형식으로 변환합니다. 이 함수의 ANSI 버전은 inet_pton. (InetPtonW)
ioctlsocket ioctlsocket 함수(winsock2.h)는 소켓의 I/O 모드를 제어하며 모든 상태의 모든 소켓에서 사용할 수 있습니다.
ioctlsocket ioctlsocket 함수(winsock.h)는 소켓의 I/O 모드를 제어합니다.
listen 수신 대기 함수는 들어오는 연결을 수신 대기하는 상태에 소켓을 배치합니다.
LPFN_CONNECTEX ConnectEx 함수는 지정된 소켓에 대한 연결을 설정하고 연결이 설정되면 필요에 따라 데이터를 보냅니다.
LPFN_DISCONNECTEX

소켓의 연결을 닫고 소켓 핸들을 다시 사용할 수 있습니다. 참고 이 함수는 Windows 소켓 사양에 대한 Microsoft 전용 확장입니다.
LPFN_RIOCLOSECOMPLETIONQUEUE Winsock 등록 I/O 확장을 사용하여 요청을 보내고 받아 I/O 완료 알림에 사용되는 기존 완료 큐를 닫습니다.
LPFN_RIOCREATECOMPLETIONQUEUE Winsock 등록 I/O 확장에 사용할 특정 크기의 I/O 완료 큐를 만듭니다.
LPFN_RIOCREATEREQUESTQUEUE Winsock 등록 I/O 확장에 사용할 지정된 소켓 및 I/O 완료 큐를 사용하여 등록된 I/O 소켓 설명자를 만듭니다.
LPFN_RIODEQUEUECOMPLETION Winsock 등록 I/O 확장에 사용할 I/O 완료 큐에서 항목을 제거합니다.
LPFN_RIODEREGISTERBUFFER Winsock 등록 I/O 확장에 사용되는 등록된 버퍼를 등록 취소합니다.
LPFN_RIONOTIFY Winsock 등록 I/O 확장에 사용하기 위해 알림 동작에 사용할 메서드를 I/O 완료 큐에 등록합니다.
LPFN_RIORECEIVE Winsock 등록 I/O 확장에 사용하기 위해 연결된 등록된 I/O TCP 소켓 또는 바인딩된 등록된 I/O UDP 소켓에서 네트워크 데이터를 받습니다.
LPFN_RIORECEIVEEX 연결된 등록된 I/O TCP 소켓 또는 바인딩된 등록된 I/O UDP 소켓에서 Winsock 등록 I/O 확장에 사용할 추가 옵션을 사용하여 네트워크 데이터를 받습니다.
LPFN_RIOREGISTERBUFFER Winsock 등록 I/O 확장에 사용할 지정된 버퍼를 사용하여 등록된 버퍼 설명자인 RIO_BUFFERID 등록합니다.
LPFN_RIORESIZECOMPLETIONQUEUE Winsock 등록 I/O 확장에 사용하기 위해 I/O 완료 큐의 크기를 더 크거나 작게 조정합니다.
LPFN_RIORESIZEREQUESTQUEUE

Winsock 등록 I/O 확장에 사용하기 위해 요청 큐의 크기를 더 크거나 작게 조정합니다.
LPFN_RIOSEND Winsock 등록 I/O 확장에 사용하기 위해 연결된 등록된 I/O TCP 소켓 또는 바인딩된 등록된 I/O UDP 소켓에 네트워크 데이터를 보냅니다.
LPFN_RIOSENDEX 연결된 등록된 I/O TCP 소켓 또는 바인딩된 등록된 I/O UDP 소켓에서 Winsock 등록 I/O 확장에 사용할 추가 옵션을 사용하여 네트워크 데이터를 보냅니다.
LPFN_TRANSMITPACKETS 연결된 소켓을 통해 메모리 내 데이터 또는 파일 데이터를 전송합니다.
LPFN_WSARECVMSG *_LPFN_WSARECVMSG_*은 함수 포인터 형식입니다. 앱에서 일치하는 **WSARecvMsg** 콜백 함수를 구현합니다. 시스템은 콜백 함수를 사용하여 연결된 소켓을 통해 메모리 내 데이터 또는 파일 데이터로 전송합니다.
LPNSPCLEANUP 특정 Windows 소켓 네임스페이스 서비스 공급자의 사용을 종료합니다.
LPNSPGETSERVICECLASSINFO 네임스페이스 공급자와 관련된 모든 관련 클래스 정보(스키마)를 검색합니다.
LPNSPINSTALLSERVICECLASS NSPInstallServiceClass 함수는 네임스페이스 공급자 내에서 서비스 클래스 스키마를 등록합니다.
LPNSPIOCTL IOCTL을 네임스페이스 서비스 공급자에게 보냅니다.
LPNSPLOOKUPSERVICEBEGIN WSAQUERYSET 구조에 포함된 정보로 제한되는 클라이언트 쿼리를 시작합니다.
LPNSPLOOKUPSERVICEEND NSPLookupServiceBegin 및 NSPLookupServiceNext에 대한 이전 호출 후 핸들을 해제하기 위해 호출됩니다.
LPNSPLOOKUPSERVICENEXT

요청된 서비스 정보를 검색하기 위해 NSPLookupServiceBegin에 대한 이전 호출에서 핸들을 가져온 후 호출됩니다.
LPNSPREMOVESERVICECLASS 네임스페이스에서 지정된 서비스 클래스를 영구적으로 제거합니다.
LPNSPSETSERVICE 네임스페이스 내에서 서비스 instance 등록하거나 등록 취소합니다.
LPNSPV2CLEANUP 네임스페이스 서비스 공급자 버전 2(NSPv2) 공급자에게 클라이언트 세션이 종료되었음을 알 수 있습니다.
LPNSPV2CLIENTSESSIONRUNDOWN 네임스페이스 서비스 공급자 버전 2(NSPv2) 공급자에게 클라이언트 세션이 종료되고 있음을 알 수 있습니다.
LPNSPV2LOOKUPSERVICEBEGIN WSAQUERYSET2 구조에 포함된 정보로 제한되는 네임스페이스 버전 2 서비스 공급자의 클라이언트 쿼리를 시작합니다.
LPNSPV2LOOKUPSERVICEEND NSPv2LookupServiceBegin 및 NSPv2LookupServiceNextEx에 대한 이전 호출 후 핸들을 해제하기 위해 호출됩니다.
LPNSPV2LOOKUPSERVICENEXTEX 네임스페이스 버전 2 서비스 공급자에서 요청된 정보를 검색하기 위해 NSPv2LookupServiceBegin에 대한 이전 호출에서 핸들을 가져온 후 호출됩니다.
LPNSPV2SETSERVICEEX NSPv2(네임스페이스 서비스 공급자 버전 2) 공급자의 네임스페이스 내에서 이름 또는 서비스 instance 등록하거나 등록 취소합니다.
LPNSPV2STARTUP 네임스페이스 서비스 공급자 version-2(NSPv2) 공급자에게 새 클라이언트 프로세스가 공급자 사용을 시작하도록 알 수 있습니다.
LPWSAOVERLAPPED_COMPLETION_ROUTINE TBD(LPWSAOVERLAPPED_COMPLETION_ROUTINE)

LPWSPACCEPT LPWSPAccept 함수는 조건 함수의 반환 값에 따라 연결을 조건부로 허용합니다.
LPWSPADDRESSTOSTRING LPWSPAddressToString 함수는 sockaddr 구조체의 모든 구성 요소를 주소의 사람이 읽을 수 있는 숫자 문자열 표현으로 변환합니다. 이는 주로 표시 목적으로 사용됩니다.
LPWSPASYNCSELECT LPWSPAsyncSelect 함수는 소켓에 대한 네트워크 이벤트의 Windows 메시지 기반 이벤트 알림을 요청합니다.
LPWSPBIND LPWSPBind 함수는 로컬 주소(즉, 이름)를 소켓과 연결합니다.
LPWSPCANCELBLOCKINGCALL LPWSPCancelBlockingCall 함수는 현재 진행 중인 차단 호출을 취소합니다.
LPWSPCLEANUP LPWSPCleanup 함수는 Windows 소켓 서비스 공급자의 사용을 종료합니다.
LPWSPCLOSESOCKET LPWSPCloseSocket 함수는 소켓을 닫습니다.
LPWSPCONNECT LPWSPConnect 함수는 피어에 대한 연결을 설정하고, 연결 데이터를 교환하고, 제공된 흐름 사양에 따라 필요한 서비스 품질을 지정합니다.
LPWSPDUPLICATESOCKET LPWSPDuplicateSocket 함수는 공유 소켓에 대한 새 소켓 설명자를 만드는 데 사용할 수 있는 WSAPROTOCOL_INFO 구조를 반환합니다.
LPWSPENUMNETWORKEVENTS LPWSPEnumNetworkEvents 함수는 표시된 소켓에 대한 네트워크 이벤트의 발생을 보고합니다.
LPWSPEVENTSELECT **LPWSPEventSelect** 함수는 제공된 네트워크 이벤트 집합과 연결할 이벤트 개체를 지정합니다.

LPWSPGETOVERLAPPEDRESULT LPWSPGetOverlappedResult 함수는 지정된 소켓에서 겹치는 작업의 결과를 반환합니다.
LPWSPGETPEERNAME LPWSPGetPeerName 함수는 소켓이 연결된 피어의 주소를 가져옵니다.
LPWSPGETQOSBYNAME LPWSPGetQOSByName 함수는 명명된 템플릿을 기반으로 QOS 구조를 초기화하거나 사용 가능한 템플릿 이름의 열거형을 검색합니다.
LPWSPGETSOCKNAME LPWSPGetSockName 함수는 소켓의 로컬 이름을 가져옵니다.
LPWSPGETSOCKOPT LPWSPGetSockOpt 함수는 소켓 옵션을 검색합니다.
LPWSPIOCTL LPWSPIoctl 함수는 소켓의 모드를 제어합니다.
LPWSPJOINLEAF LPWSPJoinLeaf 함수는 리프 노드를 다중 포인트 세션에 조인하고, 연결 데이터를 교환하고, 제공된 흐름 사양에 따라 필요한 서비스 품질을 지정합니다.
LPWSPLISTEN LPWSPListen 함수는 들어오는 연결을 수신 대기하는 소켓을 설정합니다.
LPWSPRECV LPWSPRecv 함수는 소켓에서 데이터를 받습니다.
LPWSPRECVDISCONNECT LPWSPRecvDisconnect 함수는 소켓에서 수신을 종료하고 소켓이 연결 지향인 경우 연결 끊기 데이터를 검색합니다.
LPWSPRECVFROM LPWSPRecvFrom 함수는 데이터그램을 수신하고 원본 주소를 저장합니다.
LPWSPSELECT

LPWSPSelect 함수는 하나 이상의 소켓의 상태 결정합니다.
LPWSPSEND LPWSPSend 함수는 연결된 소켓에 데이터를 보냅니다.
LPWSPSENDDISCONNECT LPWSPSendDisconnect 함수는 소켓에 대한 연결 종료를 시작하고 연결 끊기 데이터를 보냅니다.
LPWSPSENDTO WSPSendTo 함수는 겹치는 I/O를 사용하여 특정 대상으로 데이터를 보냅니다.
LPWSPSETSOCKOPT LPWSPSetSockOpt 함수는 소켓 옵션을 설정합니다.
LPWSPSHUTDOWN LPWSPShutdown 함수는 소켓에서 송신 및/또는 수신을 사용하지 않도록 설정합니다.
LPWSPSOCKET LPWSPSocket 함수는 소켓을 만듭니다.
LPWSPSTRINGTOADDRESS WSPStringToAddress 함수는 사람이 읽을 수 있는 숫자 문자열을 이러한 구조를 사용하는 Windows 소켓 루틴에 전달하는 데 적합한 소켓 주소 구조체(sockaddr)로 변환합니다.
NSPStartup DLL 진입점 목록과 같은 공급자에 대한 동적 정보를 검색합니다.
ntohd 서명되지 않은 __int64 TCP/IP 네트워크 순서에서 호스트 바이트 순서(Intel 프로세서의 little-endian)로 변환하고 double을 반환합니다.
ntohf 서명되지 않은 __int32 TCP/IP 네트워크 순서에서 호스트 바이트 순서(Intel 프로세서의 little-endian)로 변환하고 float를 반환합니다.
ntohl ntohl 함수(winsock2.h)는 u_long TCP/IP 네트워크 순서에서 호스트 바이트 순서(Intel 프로세서의 little-endian)로 변환합니다.

ntohl ntohl 함수(winsock.h)는 u_long TCP/IP 네트워크 순서에서 호스트 바이트 순서(Intel 프로세서의 little-endian)로 변환합니다.
ntohll 서명되지 않은 __int64 TCP/IP 네트워크 순서에서 호스트 바이트 순서(Intel 프로세서의 little-endian)로 변환합니다.
ntohs ntohs 함수(winsock2.h)는 u_short TCP/IP 네트워크 바이트 순서에서 호스트 바이트 순서(Intel 프로세서의 little-endian)로 변환합니다.
ntohs ntohs 함수(winsock.h)는 u_short TCP/IP 네트워크 바이트 순서에서 호스트 바이트 순서(Intel 프로세서의 little-endian)로 변환합니다.
ProcessSocketNotifications 소켓 집합을 완료 포트와 연결하고 해당 포트에서 이미 보류 중인 알림을 검색합니다. 연결되면 완료 포트는 지정된 소켓 상태 알림을 받습니다.
Recv recv 함수(winsock2.h)는 연결된 소켓 또는 바인딩된 연결 없는 소켓에서 데이터를 받습니다.
Recv recv 함수(winsock.h)는 연결된 소켓 또는 바인딩된 연결 없는 소켓에서 데이터를 받습니다.
recvfrom recvfrom 함수는 데이터그램을 수신하고 원본 주소를 저장합니다.
recvfrom recvfrom 함수는 데이터그램을 수신하고 원본 주소를 저장합니다.
선택 select 함수는 필요한 경우 동기 I/O를 수행하기 위해 대기하는 하나 이상의 소켓의 상태 결정합니다.
send 연결된 소켓에 데이터를 보냅니다. (보내기)

Sendto sendto 함수(winsock2.h)는 소켓에 나가는 데이터를 쓰는 데 사용됩니다.
Sendto sendto 함수(winsock.h)는 데이터를 특정 대상으로 보냅니다.
SetAddrInfoExA 이름, 서비스 이름 및 연결된 주소를 특정 네임스페이스 공급자에 등록하거나 등록 취소합니다. (ANSI)
SetAddrInfoExW 이름, 서비스 이름 및 연결된 주소를 특정 네임스페이스 공급자에 등록하거나 등록 취소합니다. (유니코드)
setipv4sourcefilter IPv4 소켓의 멀티캐스트 필터 상태를 설정합니다.
SetServiceA SetService 함수는 하나 이상의 네임스페이스 내의 네트워크 서비스를 레지스트리에서 등록하거나 제거합니다. (ANSI)
SetServiceW SetService 함수는 하나 이상의 네임스페이스 내의 네트워크 서비스를 레지스트리에서 등록하거나 제거합니다. (유니코드)
SetSocketMediaStreamingMode 서비스 품질이 필요한 스트리밍 미디어를 전송하는 데 네트워크를 사용할지 여부를 나타냅니다.
setsockopt setsockopt 함수(winsock2.h)는 모든 형식의 소켓과 연결된 소켓 옵션의 현재 값을 모든 상태로 설정합니다.
setsockopt setsockopt 함수(winsock.h)는 소켓 옵션을 설정합니다.
setsourcefilter IPv4 또는 IPv6 소켓의 멀티캐스트 필터 상태를 설정합니다.

shutdown 종료 함수(winsock2.h)는 수신, 전송 또는 둘 다를 사용하지 않도록 설정하기 위해 모든 유형의 소켓에서 사용됩니다.
shutdown shutdown 함수(winsock.h)는 소켓에서 송신 또는 수신을 사용하지 않도록 설정합니다.
socket 소켓 함수는 특정 전송 서비스 공급자에 바인딩된 소켓을 만듭니다.
SocketNotificationRetrieveEvents 이 인라인 도우미 함수는 OVERLAPPED_ENTRY 이벤트 마스크를 검색하는 편의를 위해 제공됩니다.
전송파일 TransmitFile 함수(mswsock.h)는 연결된 소켓 핸들을 통해 파일 데이터를 전송합니다.
전송파일 TransmitFile 함수(winsock.h)는 연결된 소켓 핸들을 통해 파일 데이터를 전송합니다.
WPUCloseEvent WPUCloseEvent 함수는 열린 이벤트 개체 핸들을 닫습니다.
WPUCloseSocketHandle WPUCloseSocketHandle 함수는 기존 소켓 핸들을 닫습니다.
WPUCloseThread WPUCloseThread 함수는 WPUOpenCurrentThread를 호출하여 연 스레드를 닫습니다.
WPUCompleteOverlappedRequest WPUCompleteOverlappedRequest 함수는 겹치는 I/O 작업에 대해 겹치는 I/O 완료 알림을 수행합니다.
WPUCreateEvent WPUCreateEvent 함수는 새 이벤트 개체를 만듭니다.
WPUCreateSocketHandle WPUCreateSocketHandle 함수는 새 소켓 핸들을 만듭니다.

WPUFDIsSet WPUFDIsSet 함수는 지정된 소켓 핸들의 멤버 자격을 확인합니다.
WPUGetProviderPath WPUGetProviderPath 함수는 지정된 공급자에 대한 DLL 경로를 검색합니다.
WPUModifyIFSHandle WPUModifyIFSHandle 함수는 Ws2_32.dll 수정된 IFS 핸들을 받습니다.
WPUOpenCurrentThread WPUOpenCurrentThread 함수는 계층화된 서비스 공급자에서 겹치는 함수와 함께 사용할 수 있는 현재 스레드에 대한 핸들을 엽니다.
WPUPostMessage WPUPostMessage 함수는 이전 버전의 WSOCK32.dll 이전 버전과의 호환성을 유지하는 방식으로 표준 Windows PostMessage 함수를 수행합니다.
WPUQueryBlockingCallback WPUQueryBlockingCallback 함수는 차단 작업을 서비스하는 동안 서비스 공급자가 주기적으로 호출해야 하는 콜백 함수에 대한 포인터를 반환합니다.
WPUQuerySocketHandleContext WPUQuerySocketHandleContext 함수는 지정된 소켓 핸들과 연결된 컨텍스트 값을 쿼리합니다.
WPUQueueApc WPUQueueApc 함수는 겹치는 I/O 완료 루틴을 쉽게 호출하기 위해 지정된 스레드에 사용자 모드 APC(비동기 프로시저 호출)를 큐에 대기합니다.
WPUResetEvent WPUResetEvent 함수는 지정된 이벤트 개체의 상태를 서명되지 않은 상태로 다시 설정합니다. 이 호출은 인터럽트 컨텍스트 내에서 사용하기에 안전합니다.
WPUSetEvent WPUSetEvent 함수는 지정된 이벤트 개체의 상태를 신호로 설정합니다. 이 호출은 인터럽트 컨텍스트 내에서 사용하기에 안전합니다.
WSAAccept

WSAAccept 함수는 조건 함수의 반환 값에 따라 연결을 조건부로 허용하고, 서비스 흐름 사양의 품질을 제공하며, 연결 데이터 전송을 허용합니다.
WSAAddressToStringA sockaddr 구조체의 모든 구성 요소를 주소의 사람이 읽을 수 있는 문자열 표현으로 변환합니다. (ANSI)
WSAAddressToStringW sockaddr 구조체의 모든 구성 요소를 주소의 사람이 읽을 수 있는 문자열 표현으로 변환합니다. (유니코드)
WSAAdvertiseProvider 적합한 모든 클라이언트에서 특정 네임스페이스 버전 2 공급자를 사용할 수 있게 합니다.
WSAAsyncGetHostByAddr WSAAsyncGetHostByAddr 함수(winsock2.h)는 주소에 해당하는 호스트 정보를 비동기적으로 검색합니다.
WSAAsyncGetHostByAddr WSAAsyncGetHostByAddr 함수(winsock.h)는 주소에 해당하는 호스트 정보를 비동기적으로 검색합니다.
WSAAsyncGetHostByAddr WSAAsyncGetHostByAddr 매크로 함수(wsipv6ok.h)는 주소에 해당하는 호스트 정보를 비동기적으로 검색합니다.
WSAAsyncGetHostByName WSAAsyncGetHostByName 함수(winsock2.h)는 호스트 이름에 해당하는 호스트 정보를 비동기적으로 검색합니다.
WSAAsyncGetHostByName WSAAsyncGetHostByName 함수(winsock.h)는 호스트 이름에 해당하는 호스트 정보를 비동기적으로 검색합니다.
WSAAsyncGetHostByName WSAAsyncGetHostByName 매크로 함수(wsipv6ok.h)는 호스트 이름에 해당하는 호스트 정보를 비동기적으로 검색합니다.
WSAAsyncGetProtoByName

WSAAsyncGetProtoByName 함수(winsock2.h)는 프로토콜 이름에 해당하는 프로토콜 정보를 비동기적으로 검색합니다.
WSAAsyncGetProtoByName WSAAsyncGetProtoByName 함수(winsock.h)는 프로토콜 이름에 해당하는 프로토콜 정보를 비동기적으로 검색합니다.
WSAAsyncGetProtoByNumber WSAAsyncGetProtoByNumber 함수(winsock2.h)는 프로토콜 번호에 해당하는 프로토콜 정보를 비동기적으로 검색합니다.
WSAAsyncGetProtoByNumber WSAAsyncGetProtoByNumber 함수(winsock.h)는 프로토콜 번호에 해당하는 프로토콜 정보를 비동기적으로 검색합니다.
WSAAsyncGetServByName WSAAsyncGetServByName 함수(winsock2.h)는 서비스 이름 및 포트에 해당하는 서비스 정보를 비동기적으로 검색합니다.
WSAAsyncGetServByName WSAAsyncGetServByName 함수(winsock.h)는 서비스 이름 및 포트에 해당하는 서비스 정보를 비동기적으로 검색합니다.
WSAAsyncGetServByPort WSAAsyncGetServByPort 함수(winsock2.h)는 포트 및 프로토콜에 해당하는 서비스 정보를 비동기적으로 검색합니다.
WSAAsyncGetServByPort WSAAsyncGetServByPort 함수(winsock.h)는 포트 및 프로토콜에 해당하는 서비스 정보를 비동기적으로 검색합니다.
WSAAsyncSelect WSAAsyncSelect 함수(winsock2.h)는 소켓에 대한 네트워크 이벤트의 Windows 메시지 기반 알림을 요청합니다.
WSAAsyncSelect WSAAsyncSelect 함수(winsock.h)는 소켓에 대한 네트워크 이벤트의 Windows 메시지 기반 알림을 요청합니다.
WSACancelAsyncRequest

WSACancelAsyncRequest 함수(winsock2.h)는 불완전한 비동기 작업을 취소합니다.
WSACancelAsyncRequest WSACancelAsyncRequest 함수(winsock.h)는 불완전한 비동기 작업을 취소합니다.
WSACancelBlockingCall WSACancelBlockingCall 함수가 Windows Sockets 2 사양, 수정 버전 2.2.0에 따라 제거되었습니다.
WSACleanup WSACleanup 함수(winsock2.h)는 WS2_32.dll 사용을 종료합니다.
WSACleanup WSACleanup 함수(winsock.h)는 WS2_32.dll 사용을 종료합니다.
WSACloseEvent WSACloseEvent 함수는 열린 이벤트 개체 핸들을 닫습니다.
WSAConnect WSAConnect 함수는 다른 소켓 애플리케이션에 대한 연결을 설정하고, 연결 데이터를 교환하고, 지정된 FLOWSPEC 구조에 따라 필요한 서비스 품질을 지정합니다.
WSAConnectByList 대상 주소 집합(호스트 이름 및 포트)으로 표시되는 가능한 엔드포인트 컬렉션 중 하나에 대한 연결을 설정합니다.
WSAConnectByNameA 지정된 호스트 및 포트에 대한 연결을 설정합니다. (ANSI)
WSAConnectByNameW 지정된 호스트 및 포트에 대한 연결을 설정합니다. (유니코드)
WSACreateEvent WSACreateEvent 함수는 새 이벤트 개체를 만듭니다.
WSADeleteSocketPeerTargetName 피어 대상 이름과 소켓의 IP 주소 간의 연결을 제거합니다. 반환이 성공하면 IP 주소와 대상 이름 간에 나중에 연결되지 않습니다.

WSADuplicateSocketA WSADuplicateSocket 함수는 공유 소켓에 대한 새 소켓 설명자를 만드는 데 사용할 수 있는 WSAPROTOCOL_INFO 구조를 반환합니다. WSADuplicateSocket 함수는 QOS 사용 소켓에서 사용할 수 없습니다. (ANSI)
WSADuplicateSocketW WSADuplicateSocket 함수는 공유 소켓에 대한 새 소켓 설명자를 만드는 데 사용할 수 있는 WSAPROTOCOL_INFO 구조를 반환합니다. WSADuplicateSocket 함수는 QOS 사용 소켓에서 사용할 수 없습니다. (유니코드)
WSAEnumNameSpaceProvidersA WSAEnumNameSpaceProviders 함수는 사용 가능한 네임스페이스 공급자에 대한 정보를 검색합니다. (ANSI)
WSAEnumNameSpaceProvidersExA 사용 가능한 네임스페이스 공급자에 대한 정보를 검색합니다. (ANSI)
WSAEnumNameSpaceProvidersExW 사용 가능한 네임스페이스 공급자에 대한 정보를 검색합니다. (유니코드)
WSAEnumNameSpaceProvidersW WSAEnumNameSpaceProviders 함수는 사용 가능한 네임스페이스 공급자에 대한 정보를 검색합니다. (유니코드)
WSAEnumNetworkEvents WSAEnumNetworkEvents 함수는 표시된 소켓에 대한 네트워크 이벤트 발생을 검색하고 내부 네트워크 이벤트 레코드를 지우고 이벤트 개체를 다시 설정합니다(선택 사항).
WSAEnumProtocolsA WSAEnumProtocols 함수는 사용 가능한 전송 프로토콜에 대한 정보를 검색합니다. (ANSI)
WSAEnumProtocolsW WSAEnumProtocols 함수는 사용 가능한 전송 프로토콜에 대한 정보를 검색합니다. (유니코드)
WSAEventSelect WSAEventSelect 함수는 지정된 FD_XXX 네트워크 이벤트 집합과 연결할 이벤트 개체를 지정합니다.

WSAGetFailConnectOnIcmpError **TCP_FAIL_CONNECT_ON_ICMP_ERROR** 소켓 옵션의 상태를 쿼리합니다.
WSAGetIcmpErrorInfo 연결 설정 중에 TCP 소켓에서 수신된 ICMP 오류에 대한 정보를 검색합니다.
WSAGetIPUserMtu 소켓에 대한 사용자 정의 IP 계층 MTU를 검색합니다.
WSAGetLastError WSAGetLastError 함수(winsock2.h)는 실패한 마지막 Windows 소켓 작업에 대한 오류 상태 반환합니다.
WSAGetLastError WSAGetLastError 함수(winsock.h)는 실패한 마지막 Windows 소켓 작업에 대한 오류 상태 반환합니다.
WSAGetOverlappedResult WSAGetOverlappedResult 함수는 지정된 소켓에서 겹치는 작업의 결과를 검색합니다.
WSAGetQOSByName WSAGetQOSByName 함수는 명명된 템플릿을 기반으로 QOS 구조를 초기화하거나 사용 가능한 템플릿 이름의 열거형을 검색하는 버퍼를 제공합니다.
WSAGetServiceClassInfoA WSAGetServiceClassInfo 함수는 지정된 네임스페이스 공급자에서 지정된 서비스 클래스와 관련된 클래스 정보(스키마)를 검색합니다. (ANSI)
WSAGetServiceClassInfoW WSAGetServiceClassInfo 함수는 지정된 네임스페이스 공급자에서 지정된 서비스 클래스와 관련된 클래스 정보(스키마)를 검색합니다. (유니코드)
WSAGetServiceClassNameByClassIdA WSAGetServiceClassNameByClassId 함수는 지정된 형식과 연결된 서비스의 이름을 검색합니다. 이 이름은 FTP 또는 SNA와 같은 일반 서비스 이름이고 해당 서비스의 특정 instance 이름이 아닙니다. (ANSI)
WSAGetServiceClassNameByClassIdW

WSAGetServiceClassNameByClassId 함수는 지정된 형식과 연결된 서비스의 이름을 검색합니다. 이 이름은 FTP 또는 SNA와 같은 일반 서비스 이름이고 해당 서비스의 특정 instance 이름이 아닙니다. (유니코드)
WSAGetUdpRecvMaxCoalescedSize UDP 소켓에 대해 받은 병합된 메시지의 최대 크기를 검색합니다.
WSAGetUdpSendMessageSize UDP 소켓의 구분 메시지 크기를 검색합니다.
WSAHtonl WSAHtonl 함수는 호스트 바이트 순서에서 네트워크 바이트 순서로 u_long 변환합니다.
WSAHtons WSAHtons 함수는 호스트 바이트 순서에서 네트워크 바이트 순서로 u_short 변환합니다.
WSAImpersonateSocketPeer 애플리케이션 수준 권한 부여를 수행하기 위해 소켓 피어에 해당하는 보안 주체를 가장하는 데 사용됩니다.
WSAInstallServiceClassA WSAInstallServiceClass 함수는 네임스페이스 내에 서비스 클래스 스키마를 등록합니다. (ANSI)
WSAInstallServiceClassW WSAInstallServiceClass 함수는 네임스페이스 내에 서비스 클래스 스키마를 등록합니다. (유니코드)
WSAIoctl WSAIoctl 함수는 소켓의 모드를 제어합니다.
WSAIsBlocking 이 함수는 Windows 소켓 2 사양, 수정 버전 2.2.0에 따라 제거되었습니다. (WSAIsBlocking)
WSAJoinLeaf WSAJoinLeaf 함수는 리프 노드를 다중 포인트 세션에 조인하고, 연결 데이터를 교환하고, 지정된 FLOWSPEC 구조에 따라 필요한 서비스 품질을 지정합니다.
WSALookupServiceBeginA

WSALookupServiceBegin 함수는 WSAQUERYSET 구조에 포함된 정보로 제한되는 클라이언트 쿼리를 시작합니다. (ANSI)
WSALookupServiceBeginW WSALookupServiceBegin 함수는 WSAQUERYSET 구조에 포함된 정보로 제한되는 클라이언트 쿼리를 시작합니다. (유니코드)
WSALookupServiceEnd WSALookupServiceEnd 함수는 WSALookupServiceBegin 및 WSALookupServiceNext에 대한 이전 호출 후 핸들을 해제하기 위해 호출됩니다.
WSALookupServiceNextA WSALookupServiceNext 함수는 요청된 서비스 정보를 검색하기 위해 WSALookupServiceBegin에 대한 이전 호출에서 핸들을 가져온 후 호출됩니다. (ANSI)
WSALookupServiceNextW WSALookupServiceNext 함수는 요청된 서비스 정보를 검색하기 위해 WSALookupServiceBegin에 대한 이전 호출에서 핸들을 가져온 후 호출됩니다. (유니코드)
WSANSPIoctl 개발자가 등록된 네임스페이스에 대한 I/O 컨트롤 호출을 수행할 수 있습니다.
WSANtohl WSANtohl 함수는 u_long 네트워크 바이트 순서에서 호스트 바이트 순서로 변환합니다.
WSANtohs WSANtohs 함수는 u_short 네트워크 바이트 순서에서 호스트 바이트 순서로 변환합니다.
WSAPoll WSAPoll 함수는 하나 이상의 소켓 상태 결정합니다.
WSAProviderCompleteAsyncCall 네임스페이스 버전 2 공급자에 대한 비동기 호출이 완료되면 클라이언트에 알린 것입니다.
WSAProviderConfigChange WSAProviderConfigChange 함수는 공급자 구성이 변경되면 애플리케이션에 알 수 있습니다.
WSAQuerySocketSecurity

소켓의 연결에 적용되는 보안에 대한 정보를 쿼리합니다.
WSARecv 연결된 소켓 또는 바인딩된 연결 없는 소켓에서 데이터를 받습니다. (WSARecv)
WSARecvDisconnect WSARecvDisconnect 함수는 소켓에서 수신을 종료하고 소켓이 연결 지향인 경우 연결 끊기 데이터를 검색합니다.
WSARecvEx WSARecvEx 함수(mswsock.h)는 연결된 소켓 또는 바인딩된 연결 없는 소켓에서 데이터를 수신합니다.
WSARecvEx WSARecvEx 함수(winsock.h)는 연결된 소켓 또는 바인딩된 연결 없는 소켓에서 데이터를 받습니다.
WSARecvFrom 데이터그램을 수신하고 원본 주소를 저장합니다.
WSARemoveServiceClass WSARemoveServiceClass 함수는 레지스트리에서 서비스 클래스 스키마를 영구적으로 제거합니다.
WSAResetEvent WSAResetEvent 함수는 지정된 이벤트 개체의 상태를 서명되지 않은 상태로 다시 설정합니다.
WSARevertImpersonation 소켓 피어의 가장을 종료합니다. WSAImpersonateSocketPeer를 호출하고 액세스 검사를 완료한 후에 호출해야 합니다.
WSASend 연결된 소켓에 데이터를 보냅니다. (WSASend)
WSASendDisconnect WSASendDisconnect 함수는 소켓에 대한 연결 종료를 시작하고 연결 끊기 데이터를 보냅니다.
WSASendMsg 연결된 소켓과 연결되지 않은 소켓에서 데이터 및 선택적 제어 정보를 보냅니다. 참고 이 함수는 Windows 소켓 사양에 대한 Microsoft 전용 확장입니다. .

WSASendTo 해당하는 경우 겹치는 I/O를 사용하여 특정 대상으로 데이터를 보냅니다.
WSASetBlockingHook 이 함수는 Windows 소켓 2 사양, 수정 버전 2.2.0에 따라 제거되었습니다. (WSASetBlockingHook)
WSASetEvent WSASetEvent 함수는 지정된 이벤트 개체의 상태를 신호로 설정합니다.
WSASetFailConnectOnIcmpError **TCP_FAIL_CONNECT_ON_ICMP_ERROR** 소켓 옵션의 상태를 설정합니다.
WSASetIPUserMtu 소켓에서 사용자 정의 IP 계층 MTU를 설정합니다.
WSASetLastError WSASetLastError 함수(winsock2.h)는 WSAGetLastError 함수를 통해 검색할 수 있는 오류 코드를 설정합니다.
WSASetLastError WSASetLastError 함수(winsock.h)는 WSAGetLastError 함수를 통해 검색할 수 있는 오류 코드를 설정합니다.
WSASetServiceA WSASetService 함수는 하나 이상의 네임스페이스 내에서 서비스가 instance 레지스트리에서 등록하거나 제거합니다. (ANSI)
WSASetServiceW WSASetService 함수는 하나 이상의 네임스페이스 내에서 서비스가 instance 레지스트리에서 등록하거나 제거합니다. (유니코드)
WSASetSocketPeerTargetName 피어 IP 주소에 해당하는 SPN(피어 대상 이름)을 지정하는 데 사용됩니다. 이 대상 이름은 인증해야 하는 피어를 안전하게 식별하기 위해 클라이언트 애플리케이션에서 지정해야 합니다.
WSASetSocketSecurity 소켓에 대한 보안을 사용하도록 설정하고 적용합니다.

WSASetUdpRecvMaxCoalescedSize UDP 소켓에 설정된 병합된 메시지의 최대 크기를 설정합니다.
WSASetUdpSendMessageSize UDP 소켓의 구분 메시지 크기를 설정합니다.
WSASocketA WSASocket 함수는 특정 전송 서비스 공급자에 바인딩된 소켓을 만듭니다. (ANSI)
WSASocketW WSASocket 함수는 특정 전송 서비스 공급자에 바인딩된 소켓을 만듭니다. (유니코드)
WSAStartup WSAStartup 함수(winsock2.h)는 프로세스에서 Winsock DLL 사용을 시작합니다.
WSAStartup WSAStartup 함수(winsock.h)는 프로세스에서 Winsock DLL 사용을 시작합니다.
WSAStringToAddressA WSAStringToAddress 함수는 표준 텍스트 프레젠테이션 양식의 네트워크 주소를 sockaddr 구조체의 숫자 이진 형식으로 변환하며, 이러한 구조를 사용하는 Windows 소켓 루틴에 전달하는 데 적합합니다. (ANSI)
WSAStringToAddressW WSAStringToAddress 함수는 표준 텍스트 프레젠테이션 양식의 네트워크 주소를 sockaddr 구조체의 숫자 이진 형식으로 변환하며, 이러한 구조를 사용하는 Windows 소켓 루틴에 전달하는 데 적합합니다. (유니코드)
WSAUnadvertiseProvider 클라이언트에서 특정 네임스페이스 버전 2 공급자를 더 이상 사용할 수 없게 만듭니다.
WSAUnhookBlockingHook 이 함수는 Windows 소켓 2 사양인 수정 버전 2.2.0에 따라 제거되었습니다. (WSAUnhookBlockingHook)
WSAWaitForMultipleEvents 지정된 이벤트 개체 중 하나 또는 전부가 신호 상태에 있거나, 제한 시간 간격이 만료되거나, I/O 완료 루틴이 실행된 경우를 반환합니다.

WSCDeinstallProvider 시스템 구성 데이터베이스에서 지정된 전송 공급자를 제거합니다.
WSCDeinstallProvider32 시스템 구성 데이터베이스에서 지정된 32비트 전송 공급자를 제거합니다.
WSCEnableNSProvider 지정된 네임스페이스 공급자의 상태를 변경합니다.
WSCEnableNSProvider32 지정된 32비트 네임스페이스 공급자를 사용하거나 사용하지 않도록 설정합니다.
WSCEnumNameSpaceProviders32 사용 가능한 32비트 네임스페이스 공급자에 대한 정보를 반환합니다. 참고 이 호출은 64비트 플랫폼에서 사용할 WSAEnumNameSpaceProviders의 엄격한 32비트 버전입니다. 64비트 프로세스가 32비트 카탈로그에 액세스할 수 있도록 제공됩니다. .
WSCEnumNameSpaceProvidersEx32 사용 가능한 32비트 네임스페이스 공급자에 대한 정보를 검색합니다.
WSCEnumProtocols WSCEnumProtocols 함수는 사용 가능한 전송 프로토콜에 대한 정보를 검색합니다.
WSCEnumProtocols32 사용 가능한 전송 프로토콜에 대한 정보를 검색합니다. 참고 이 호출은 64비트 플랫폼에서 사용하기 위해 엄격하게 32비트 버전의 WSCEnumProtocols입니다. 64비트 프로세스가 32비트 카탈로그에 액세스할 수 있도록 제공됩니다. .
WSCGetApplicationCategory 애플리케이션과 연결된 LSP(계층화된 서비스 공급자) 범주를 검색합니다.
WSCGetProviderInfo LSP(계층화된 서비스 공급자)의 정보 클래스와 연결된 데이터를 검색합니다.
WSCGetProviderInfo32 32비트 LSP(계층화된 서비스 공급자)에 대한 정보 클래스와 연결된 데이터를 검색합니다. 참고 이 호출은 64비트 플랫폼에서 사용하기 위해 엄격하게 32비트 버전의 WSCGetProviderInfo입니다.

WSCGetProviderPath WSCGetProviderPath 함수는 지정된 공급자에 대한 DLL 경로를 검색합니다.
WSCGetProviderPath32 지정된 32비트 공급자에 대한 DLL 경로를 검색합니다. 참고 이 호출은 64비트 플랫폼에서 사용하기 위해 엄격하게 32비트 버전의 WSCGetProviderPath입니다. 64비트 프로세스가 32비트 카탈로그에 액세스할 수 있도록 제공됩니다. .
WSCInstallNameSpace 네임스페이스 공급자를 설치합니다. (WSCInstallNameSpace)
WSCInstallNameSpace32 지정된 32비트 네임스페이스 공급자를 설치합니다. (WSCInstallNamespace32)
WSCInstallNameSpaceEx 네임스페이스 공급자를 설치합니다. (WSCInstallNameSpaceEx)
WSCInstallNameSpaceEx32 지정된 32비트 네임스페이스 공급자를 설치합니다. (WSCInstallNameSpaceEx32)
WSCInstallProvider 지정된 전송 공급자를 시스템 구성 데이터베이스에 설치합니다.
WSCInstallProvider64_32 지정된 전송 서비스 공급자를 64비트 컴퓨터의 32비트 및 64비트 시스템 구성 데이터베이스에 설치합니다.
WSCInstallProviderAndChains 지정된 32비트 전송 공급자와 특정 프로토콜 체인을 32비트 컴퓨터의 Winsock 2 시스템 구성 데이터베이스에 설치합니다.
WSCInstallProviderAndChains64_32 지정된 전송 공급자와 해당 특정 프로토콜 체인을 64비트 컴퓨터의 32비트 및 64비트 Winsock 2 시스템 구성 데이터베이스에 모두 설치합니다.
WSCInstallQOSTemplate 시스템 구성 데이터베이스에 지정된 QoS 템플릿을 설치합니다.

WSCRemoveQOSTemplate 시스템 구성 데이터베이스에서 지정된 QoS 템플릿을 제거합니다.
WSCSetApplicationCategory 애플리케이션과 연결된 허용된 LSP(계층화된 서비스 공급자) 범주를 설정합니다.
WSCSetProviderInfo LSP(계층화된 서비스 공급자)에 대해 지정된 정보 클래스의 데이터 값을 설정합니다.
WSCSetProviderInfo32 LSP(계층화된 서비스 공급자)에 대해 지정된 정보 클래스의 데이터 값을 설정합니다.
WSCUnInstallNameSpace 표시된 이름 공간 공급자를 제거합니다.
WSCUnInstallNameSpace32 특정 32비트 네임스페이스 공급자를 제거합니다.
WSCUpdateProvider 시스템 구성 데이터베이스에서 지정된 전송 공급자를 수정합니다.
WSCUpdateProvider32 시스템 구성 데이터베이스에서 지정된 32비트 전송 공급자를 수정합니다. 참고 이 호출은 64비트 플랫폼에서 사용하기 위해 엄격하게 32비트 버전의 WSCUpdateProvider입니다. 64비트 프로세스가 32비트 카탈로그에 액세스할 수 있도록 제공됩니다. .
WSCWriteNameSpaceOrder 사용 가능한 Windows 소켓(Winsock) 2 네임스페이스 공급자의 순서를 변경합니다. 네임스페이스 공급자의 순서는 이름 확인을 위해 열거하거나 쿼리할 때 네임스페이스의 우선 순위를 결정합니다.
WSCWriteNameSpaceOrder32 32비트 카탈로그에서 사용 가능한 Windows 소켓(Winsock) 2 네임스페이스 공급자의 순서를 변경합니다.
WSCWriteProviderOrder 사용 가능한 전송 공급자의 순서를 다시 지정하는 데 사용됩니다.

WSCWriteProviderOrder32 사용 가능한 32비트 전송 공급자의 순서를 다시 지정하는 데 사용됩니다.
WSPStartup WSPStartup 함수는 클라이언트에서 WINDOWS 소켓 SPI(서비스 공급자 인터페이스)의 사용을 시작합니다.

구조체

테이블 확장

ADDRINFO_DNS_SERVER Winsock API에 사용되는 DNS(사용자 지정 도메인 이름 시스템) 서버를 나타냅니다.
ADDRINFOA getaddrinfo 함수에서 호스트 주소 정보를 보유하는 데 사용됩니다.
ADDRINFOEX2A 정식 이름과 정규화된 도메인 이름이 모두 요청된 경우 GetAddrInfoEx 함수에서 호스트 주소 정보를 보유하는 데 사용됩니다. (ANSI)
ADDRINFOEX2W 정식 이름과 정규화된 도메인 이름이 모두 요청된 경우 GetAddrInfoEx 함수에서 호스트 주소 정보를 보유하는 데 사용됩니다. (유니코드)
ADDRINFOEX3 특정 네트워크 인터페이스가 요청되었을 때 GetAddrInfoEx 함수에서 호스트 주소 정보를 보유하는 데 사용됩니다. (addrinfoex3W)
ADDRINFOEX4 특정 네트워크 인터페이스가 요청되었을 때 GetAddrInfoEx 함수에서 호스트 주소 정보를 보유하는 데 사용됩니다. (addrinfoex4W)
ADDRINFOEX5

특정 네트워크 인터페이스가 요청되었을 때 GetAddrInfoExW 함수에서 호스트 주소 정보를 보유하는 데 사용됩니다. (ADDRINFOEX5)
ADDRINFOEX6 특정 네트워크 인터페이스가 요청되었을 때 GetAddrInfoExW 함수에서 호스트 주소 정보를 보유하는 데 사용됩니다. (ADDRINFOEX6)
ADDRINFOEXA GetAddrInfoEx 함수에서 호스트 주소 정보를 보유하는 데 사용됩니다. (ANSI)
ADDRINFOEXW GetAddrInfoEx 함수에서 호스트 주소 정보를 보유하는 데 사용됩니다. (유니코드)
ADDRINFOW GetAddrInfoW 함수에서 호스트 주소 정보를 보유하는 데 사용됩니다.
AFPROTOCOLS AFPROTOCOLS 구조는 애플리케이션 프로그래머가 쿼리를 제한할 수 있는 프로토콜 목록을 제공합니다. AFPROTOCOLS 구조체는 쿼리 목적으로만 사용됩니다.
ASSOCIATE_NAMERES_CONTEXT_INPUT 정규화된 도메인 이름에 대한 전송 설정 ID 및 핸들을 포함합니다.
ATM_ADDRESS ATM_ADDRESS 구조체는 ATM 기반 소켓에 대한 ATM 주소 데이터를 보유합니다.
ATM_BHLI ATM_BHLI 구조체는 연결된 ATM 소켓에 대한 B-HLI 정보를 식별하는 데 사용됩니다.
ATM_BLLI ATM_BLLI 구조체는 연결된 ATM 소켓에 대한 B-LLI 정보를 식별하는 데 사용됩니다.
BLOB Binary Large Object에서 파생된 BLOB(wtypesbase.h) 구조체에는 데이터 블록에 대한 정보가 포함됩니다.
BLOB

Binary Large Object에서 파생된 BLOB 구조체(winsock2.h)에는 데이터 블록에 대한 정보가 포함되어 있습니다.
BLOB 이진 큰 개체에서 파생된 BLOB 구조체(nspapi.h)에는 데이터 블록에 대한 정보가 포함됩니다.
CSADDR_INFO CSADDR_INFO(ws2def.h) 구조체에는 소켓, 네트워크 서비스 또는 네임스페이스 공급자에 대한 Windows 소켓 주소 정보가 포함됩니다.
CSADDR_INFO CSADDR_INFO 구조체(nspapi.h)에는 소켓, 네트워크 서비스 또는 네임스페이스 공급자에 대한 Windows 소켓 주소 정보가 포함됩니다.
fd_set Fd_set 구조체(winsock2.h)는 Windows 소켓(Winsock) 함수 및 서비스 공급자가 소켓을 집합에 배치하는 데 사용됩니다.
fd_set fd_set 구조체(winsock.h)는 Windows 소켓(Winsock) 함수 및 서비스 공급자가 소켓을 집합에 배치하는 데 사용됩니다.
GROUP_FILTER 멀티캐스트 IPv6 또는 IPv4 주소에 대한 멀티캐스트 필터링 매개 변수를 제공합니다.
GROUP_REQ IPv6 또는 IPv4 주소에 대한 멀티캐스트 그룹 정보를 제공합니다.
GROUP_SOURCE_REQ 원본 IP 주소를 포함하는 IPv6 또는 IPv4 주소에 대한 멀티캐스트 그룹 정보를 제공합니다.
HOSTENT HOSTENT 구조체(winsock2.h)는 함수에서 호스트 이름, IPv4 주소 등과 같은 지정된 호스트에 대한 정보를 저장하는 데 사용됩니다.
HOSTENT HOSTENT(winsock.h) 구조체는 함수에서 호스트 이름, IPv4 주소 등과 같은 지정된 호스트에 대한 정보를 저장하는 데 사용됩니다.

ICMP_ERROR_INFO 수신된 ICMP 오류 정보를 저장하는 데 사용됩니다.
in_addr in_addr 구조체는 IPv4 인터넷 주소를 나타냅니다.
IN_PKTINFO in_pktinfo 구조는 수신된 패킷 주소 정보를 저장하는 데 사용되며 Windows에서 수신된 패킷에 대한 정보를 반환하는 데 사용되며 패킷을 보내는 데 사용할 로컬 IPv4 주소를 지정할 수도 있습니다.
IN6_PKTINFO in6_pktinfo 구조는 수신된 IPv6 패킷 주소 정보를 저장하는 데 사용되며 Windows에서 수신된 패킷에 대한 정보를 반환하는 데 사용되며 패킷을 보내는 데 사용할 로컬 IPv6 주소를 지정할 수도 있습니다.
INET_PORT_RANGE SIO_ACQUIRE_PORT_RESERVATION IOCTL에서 TCP 또는 UDP 포트 블록에 대한 런타임 예약을 획득하는 데 사용하는 입력 데이터를 제공합니다.
INET_PORT_RESERVATION_INSTANCE TCP 또는 UDP 포트 블록에 대한 포트 예약 및 토큰을 포함합니다.
INET_PORT_RESERVATION_TOKEN TCP 또는 UDP 포트 블록에 대한 포트 예약 토큰을 포함합니다.
INTERFACE_INFO INTERFACE_INFO 구조체는 인터페이스 IP 주소에 대한 정보를 얻기 위해 SIO_GET_INTERFACE_LIST ioctl 명령과 함께 사용됩니다.
INTERFACE_INFO_EX INTERFACE_INFO_EX 구조는 인터페이스 IP 주소에 대한 정보를 얻기 위해 SIO_GET_INTERFACE_LIST IOCTL 명령과 함께 사용됩니다.
IP_MREQ ip_mreq 구조는 IPv4 주소에 대한 멀티캐스트 그룹 정보를 제공합니다.
IP_MREQ_SOURCE ip_mreq_source 구조는 IPv4 주소에 대한 멀티캐스트 그룹 정보를 제공합니다.

IP_MSFILTER ip_msfilter 구조는 IPv4 주소에 대한 멀티캐스트 필터링 매개 변수를 제공합니다.
IPV6_MREQ ipv6_mreq 구조는 IPv6 주소에 대한 멀티캐스트 그룹 정보를 제공합니다.
IPX_ADDRESS_DATA IPX_ADDRESS_DATA 구조체는 IPX가 바인딩된 특정 어댑터에 대한 정보를 제공합니다. optname 매개 변수에서 IPX_ADDRESS 지정하는 getsockopt 함수 호출과 함께 사용됩니다.
IPX_NETNUM_DATA IPX_NETNUM_DATA 구조체는 지정된 IPX 네트워크 번호에 대한 정보를 제공합니다. optname 매개 변수에서 IPX_GETNETINFO 지정하는 getsockopt 함수 호출과 함께 사용됩니다.
IPX_SPXCONNSTATUS_DATA IPX_SPXCONNSTATUS_DATA 구조체는 연결된 SPX 소켓에 대한 정보를 제공합니다.
머무르게 LINGER 구조체(winsock2.h)는 데이터가 큐에 대기할 때 해당 소켓이 작동하는 방식을 지정하는 특정 소켓에 대한 정보를 유지 관리합니다.
머무르게 LINGER(winsock.h) 구조는 데이터를 보낼 큐에 대기할 때 해당 소켓이 작동하는 방식을 지정하는 특정 소켓에 대한 정보를 유지 관리합니다.
NAPI_DOMAIN_DESCRIPTION_BLOB NS_EMAIL 네임스페이스에 대한 네임스페이스 공급자가 처리하는 도메인에 대해 설명합니다.
NAPI_PROVIDER_INSTALLATION_BLOB NS_EMAIL 네임스페이스에 대한 네임스페이스 공급자를 설치하는 데 필요한 정보를 포함합니다.
NS_SERVICE_INFOA 지정된 네임스페이스 또는 기본 네임스페이스 집합의 컨텍스트에서 네트워크 서비스 또는 네트워크 서비스 유형에 대한 정보를 포함합니다. (ANSI)
NS_SERVICE_INFOW 지정된 네임스페이스 또는 기본 네임스페이스 집합의 컨텍스트에서 네트워크 서비스 또는 네트워크 서비스 유형에 대한 정보를 포함합니다. (유니코드)

NSP_ROUTINE NSPv1(네임스페이스 서비스 공급자 버전 1) 공급자가 구현하는 함수에 대한 정보를 포함합니다.
NSPV2_ROUTINE 네임스페이스 서비스 공급자 version-2(NSPv2) 공급자가 구현하는 함수에 대한 정보를 포함합니다.
PROTOCOL_INFOA 프로토콜에 대한 정보를 포함합니다. (ANSI)
PROTOCOL_INFOW 프로토콜에 대한 정보를 포함합니다. (유니코드)
PROTOENT PROTOENT 구조체(winsock2.h)에는 지정된 프로토콜 이름에 해당하는 이름과 프로토콜 번호가 포함됩니다.
PROTOENT PROTOENT(winsock.h) 구조체에는 지정된 프로토콜 이름에 해당하는 이름 및 프로토콜 번호가 포함됩니다.
REAL_TIME_NOTIFICATION_SETTING_INPUT Windows 스토어 앱에서 백그라운드 네트워크 알림을 수신하기 위해 ControlChannelTrigger와 함께 사용되는 TCP 소켓에 대한 REAL_TIME_NOTIFICATION_CAPABILITY 전송 설정에 적용할 입력 설정을 제공합니다.
REAL_TIME_NOTIFICATION_SETTING_OUTPUT Windows 스토어 앱에서 백그라운드 네트워크 알림을 수신하기 위해 ControlChannelTrigger와 함께 사용되는 TCP 소켓에 대한 REAL_TIME_NOTIFICATION_CAPABILITY 전송 설정에 대한 쿼리의 출력 설정을 제공합니다.
RIO_BUF Winsock 등록 I/O 확장을 사용하여 네트워크 데이터를 보내거나 받는 데 사용되는 등록된 버퍼의 일부를 지정합니다.
RIO_EXTENSION_FUNCTION_TABLE Winsock 등록 I/O 확장을 구현하는 함수에 대한 정보를 포함합니다.
RIO_NOTIFICATION_COMPLETION

Winsock 등록 I/O 확장을 사용하여 네트워크 데이터를 보내거나 받기 위해 RIONotify 함수와 함께 사용할 I/O 완성 방법을 지정합니다.
RIORESULT Winsock 등록 I/O 확장에 사용되는 요청 완료 결과를 나타내는 데 사용되는 데이터를 포함합니다.
RM_FEC_INFO RM_FEC_INFO 구조는 신뢰할 수 있는 멀티캐스트에서 FEC(정방향 오류 수정)를 사용하기 위한 설정을 지정합니다. 이 구조체는 RM_USE_FEC 소켓 옵션과 함께 사용됩니다.
RM_RECEIVER_STATS 신뢰할 수 있는 멀티캐스트 수신기 세션에 대한 통계 정보를 제공합니다. 이 구조체는 RM_RECEIVER_STATISTICS 소켓 옵션과 함께 사용됩니다.
RM_SEND_WINDOW RM_SEND_WINDOW 구조체는 신뢰할 수 있는 멀티캐스트 보내기 창을 지정합니다. 이 구조체는 RM_RATE_WINDOW_SIZE 소켓 옵션과 함께 사용됩니다.
RM_SENDER_STATS 신뢰할 수 있는 멀티캐스트 보낸 사람 세션에 대한 통계 정보를 제공합니다. 이 구조체는 RM_SENDER_STATISTICS 소켓 옵션과 함께 사용됩니다.
SERVENT SERVENT 구조체(winsock2.h)는 지정된 서비스 이름의 이름과 서비스 번호를 저장하거나 반환하는 데 사용됩니다.
SERVENT SERVENT(winsock.h) 구조체는 지정된 서비스 이름에 대한 이름 및 서비스 번호를 저장하거나 반환하는 데 사용됩니다.
SERVICE_ADDRESS 서비스에 대한 주소 정보를 포함합니다. 구조체는 RPC(원격 프로시저 호출), 명명된 파이프 및 소켓을 포함하여 다양한 유형의 IPC(Interprocess Communication) 메커니즘 및 해당 주소 양식을 수용할 수 있습니다.
SERVICE_ADDRESSES SERVICE_ADDRESSES 구조에는 SERVICE_ADDRESS 데이터 구조의 배열이 포함되어 있습니다.
SERVICE_INFOA

네트워크 서비스 또는 네트워크 서비스 유형에 대한 정보를 포함합니다. (ANSI)
SERVICE_INFOW 네트워크 서비스 또는 네트워크 서비스 유형에 대한 정보를 포함합니다. (유니코드)
SERVICE_TYPE_INFO_ABSA SERVICE_TYPE_INFO_ABS 구조에는 네트워크 서비스 유형에 대한 정보가 포함됩니다. SERVICE_TYPE_INFO_ABS 사용하여 네임스페이스에 네트워크 서비스 유형을 추가합니다. (ANSI)
SERVICE_TYPE_INFO_ABSW SERVICE_TYPE_INFO_ABS 구조에는 네트워크 서비스 유형에 대한 정보가 포함됩니다. SERVICE_TYPE_INFO_ABS 사용하여 네임스페이스에 네트워크 서비스 유형을 추가합니다. (유니코드)
SERVICE_TYPE_VALUE_ABSA 네트워크 서비스 유형 값에 대한 정보를 포함합니다. 이 정보는 네임스페이스에만 해당될 수 있습니다. (ANSI)
SERVICE_TYPE_VALUE_ABSW 네트워크 서비스 유형 값에 대한 정보를 포함합니다. 이 정보는 네임스페이스에만 해당될 수 있습니다. (유니코드)
SOCK_NOTIFY_REGISTRATION ProcessSocketNotifications 함수에 제공된 정보를 나타냅니다.
SOCKADDR sockaddr 구조는 선택한 프로토콜에 따라 달라집니다. (sockaddr)
sockaddr_atm Windows 소켓 sockaddr_atm 구조체는 ATM 소켓에 대한 소켓 주소 정보를 저장합니다.
sockaddr_gen 제네릭 소켓 주소 정보를 제공하며 INTERFACE_INFO 구조와 함께 사용됩니다.
SOCKADDR_IN SOCKADDR_IN(winsock.h) 구조는 선택한 프로토콜에 따라 달라집니다.
sockaddr_in6_old

sockaddr_in6_old(ws2ipdef.h) 구조는 선택한 프로토콜에 따라 달라집니다.
SOCKADDR_IRDA SOCKADDR_IRDA 구조는 주소 패밀리 AF_IRDA 정의된 IrDA 소켓 작업과 함께 사용됩니다.
SOCKET_ADDRESS SOCKET_ADDRESS 구조는 프로토콜별 주소 정보를 저장합니다.
SOCKET_PEER_TARGET_NAME 피어 대상의 IP 주소와 이름 및 소켓에서 사용할 보안 프로토콜 유형을 포함합니다.
SOCKET_PROCESSOR_AFFINITY 소켓과 RSS 프로세서 코어 및 NUMA 노드 간의 연결을 포함합니다.
SOCKET_SECURITY_QUERY_INFO WSAQuerySocketSecurity 함수에서 반환된 보안 정보를 포함합니다.
SOCKET_SECURITY_QUERY_TEMPLATE WSAQuerySocketSecurity 함수에서 사용하는 보안 템플릿을 포함합니다.
SOCKET_SECURITY_SETTINGS 소켓에 대한 일반 보안 요구 사항을 지정합니다.
SOCKET_SECURITY_SETTINGS_IPSEC IPsec과 관련된 다양한 보안 요구 사항 및 설정을 지정합니다.
TCP_INFO_v0 소켓에 대해 수집된 TCP(Transmission Control Protocol) 통계를 포함합니다.
TCP_INFO_v1 소켓에 대해 수집된 TCP(Transmission Control Protocol) 통계를 포함합니다. (버전 1.0)
TCP_INITIAL_RTO_PARAMETERS 소켓에서 사용할 초기 RTO(재전송 시간 제한) 매개 변수를 구성하기 위해 SIO_TCP_INITIAL_RTO IOCTL에서 사용하는 데이터를 지정합니다.
TIMESTAMPING_CONFIG

SIO_TIMESTAMPING IOCTL에서 데이터그램 소켓에 대한 타임스탬프 수신을 구성하는 데 사용하는 입력 구조에 대해 설명합니다.
TIMEVAL TIMEVAL 구조체(winsock2.h)는 시간 간격을 지정하는 데 사용됩니다. BSD(Berkeley Software Distribution) Time.h 헤더 파일과 연결됩니다.
TIMEVAL TIMEVAL(winsock.h) 구조체는 시간 간격을 지정하는 데 사용됩니다. BSD(Berkeley Software Distribution) Time.h 헤더 파일과 연결됩니다.
TRANSMIT_FILE_BUFFERS TRANSMIT_FILE_BUFFERS 구조체(mswsock.h)는 TransmitFile 함수 파일 전송 작업 중 파일 데이터 전후에 전송할 데이터를 지정합니다.
TRANSMIT_FILE_BUFFERS TRANSMIT_FILE_BUFFERS(winsock.h) 구조는 TransmitFile 함수 파일 전송 작업 중 파일 데이터 전후에 전송할 데이터를 지정합니다.
TRANSMIT_PACKETS_ELEMENT TransmitPackets 함수에서 전송할 단일 데이터 요소를 지정합니다.
TRANSPORT_SETTING_ID TRANSPORT_SETTING_ID 구조체(mstcpip.h)는 특정 IOCTL에서 소켓에 대한 전송 설정을 적용하거나 쿼리하는 데 사용하는 전송 설정 ID를 지정합니다.
TRANSPORT_SETTING_ID TRANSPORT_SETTING_ID(transportsettingcommon.h) 구조체는 SIO_APPLY_TRANSPORT_SETTING 및 SIO_QUERY_TRANSPORT_SETTING IOCTL에서 사용하는 전송 설정 ID를 지정합니다.
WSABUF WSABUF 구조를 사용하면 일부 Winsock 함수에서 사용하는 데이터 버퍼를 만들거나 조작할 수 있습니다.
WSACOMPLETION 등록된 네임스페이스에 대한 I/O 컨트롤 호출에 대한 완료 알림 설정을 지정합니다.
WSADATA WSADATA 구조체(winsock2.h)에는 Windows 소켓 구현에 대한 정보가 포함되어 있습니다.

WSADATA WSADATA(winsock.h) 구조체에는 Windows 소켓 구현에 대한 정보가 포함되어 있습니다.
WSAMSG WSARecvMsg 및 WSASendMsg 함수와 함께 사용하여 연결 및 연결되지 않은 소켓에 대한 주소 및 선택적 제어 정보와 메시지 데이터를 저장하는 데 사용되는 버퍼 배열을 저장합니다.
WSANAMESPACE_INFOA 네임스페이스 공급자에 대한 모든 등록 정보를 포함합니다. (WSANAMESPACE_INFOA)
WSANAMESPACE_INFOEXA 네임스페이스 공급자에 대한 모든 등록 정보를 포함합니다. (WSANAMESPACE_INFOEXA)
WSANAMESPACE_INFOEXW 네임스페이스 공급자에 대한 모든 등록 정보를 포함합니다. (WSANAMESPACE_INFOEXW)
WSANAMESPACE_INFOW 네임스페이스 공급자에 대한 모든 등록 정보를 포함합니다. (WSANAMESPACE_INFOW)
WSANETWORKEVENTS WSANETWORKEVENTS 구조체는 네트워크 이벤트에 대한 소켓의 내부 정보를 저장하는 데 사용됩니다.
WSANSCLASSINFOA WSANSCLASSINFO 구조체는 특정 Windows 소켓 네임스페이스에 대한 개별 매개 변수 정보를 제공합니다. (ANSI)
WSANSCLASSINFOW WSANSCLASSINFO 구조체는 특정 Windows 소켓 네임스페이스에 대한 개별 매개 변수 정보를 제공합니다. (유니코드)
WSAOVERLAPPED 겹치는 I/O 작업의 시작과 후속 완료 사이의 통신 매체를 제공합니다.
WSAPOLLFD WSAPoll 함수에서 사용하는 소켓 정보를 저장합니다.

WSAPROTOCOL_INFOA 지정된 프로토콜에 대한 전체 정보를 저장하거나 검색하는 데 사용됩니다. (ANSI)
WSAPROTOCOL_INFOW 지정된 프로토콜에 대한 전체 정보를 저장하거나 검색하는 데 사용됩니다. (유니코드)
WSAPROTOCOLCHAIN WSAPROTOCOLCHAIN 구조에는 프로토콜 체인을 구성하는 카탈로그 항목 식별자의 계산된 목록이 포함되어 있습니다.
WSAQUERYSET2A 서비스 클래스 ID, 서비스 이름, 해당 네임스페이스 식별자 및 프로토콜 정보뿐만 아니라 서비스가 수신 대기하는 전송 주소 집합을 포함하여 지정된 서비스에 대한 관련 정보를 제공합니다. (ANSI)
WSAQUERYSET2W 서비스 클래스 ID, 서비스 이름, 해당 네임스페이스 식별자 및 프로토콜 정보뿐만 아니라 서비스가 수신 대기하는 전송 주소 집합을 포함하여 지정된 서비스에 대한 관련 정보를 제공합니다. (유니코드)
WSAQUERYSETA 서비스 클래스 ID, 서비스 이름, 해당 네임스페이스 식별자 및 프로토콜 정보뿐만 아니라 서비스가 수신 대기하는 전송 주소 집합을 포함하여 지정된 서비스에 대한 관련 정보를 제공합니다. (ANSI)
WSAQUERYSETW 서비스 클래스 ID, 서비스 이름, 해당 네임스페이스 식별자 및 프로토콜 정보뿐만 아니라 서비스가 수신 대기하는 전송 주소 집합을 포함하여 지정된 서비스에 대한 관련 정보를 제공합니다. (유니코드)
WSASERVICECLASSINFOA WSASERVICECLASSINFO 구조체에는 지정된 서비스 클래스에 대한 정보가 포함됩니다. Windows 소켓 2의 각 서비스 클래스에는 단일 WSASERVICECLASSINFO 구조가 있습니다. (ANSI)
WSASERVICECLASSINFOW WSASERVICECLASSINFO 구조체에는 지정된 서비스 클래스에 대한 정보가 포함됩니다. Windows 소켓 2의 각 서비스 클래스에는 단일 WSASERVICECLASSINFO 구조가 있습니다. (유니코드)
WSATHREADID WSATHREADID 구조를 사용하면 공급자가 WPUQueueApc 함수를 사용하여 APC(비동기 프로시저 호출)를 큐에 대기할 수 있는 스레드를 식별할 수 있습니다.

WSAVERSION WSAVERSION 구조는 Windows 소켓에서 버전 비교를 제공합니다.
WSC_PROVIDER_AUDIT_INFO Windows 소켓 2의 LSP(계층화된 서비스 공급자) 항목에 대한 감사 정보를 포함합니다.
WSPDATA WSPDATA 구조에는 서비스 공급자 정보가 포함됩니다.
WSPPROC_TABLE 서비스 공급자 함수에 대한 포인터 테이블을 포함합니다.
WSPUPCALLTABLE 서비스 공급자 업콜 함수에 대한 포인터 테이블을 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

Linux용 Windows 하위 시스템

아티클2023. 08. 22.

Linux용 Windows 하위 시스템 기술 개요입니다.

Linux용 Windows 하위 시스템 개발하려면 다음 헤더가 필요합니다.

- wslapi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Linux용 Windows 하위 시스템

열거형

WSL_DISTRIBUTION_FLAGS WSL_DISTRIBUTION_FLAGS 열거형은 Linux용 Windows 하위 시스템(WSL)의 분포 동작을 지정합니다.

Functions

WslConfigureDistribution Linux용 Windows 하위 시스템(WSL)에 등록된 배포의 동작을 수정합니다.
WslGetDistributionConfiguration Linux용 Windows 하위 시스템(WSL)에 등록된 배포판의 현재 구성을 검색합니다.
WslIsDistributionRegistered 배포판이 WSL(Linux용 Windows 하위 시스템)에 등록되었는지 확인합니다.
WslLaunch 특정 배포의 컨텍스트에서 WSL(Linux용 Windows 하위 시스템) 프로세스를 시작합니다.
WslLaunchInteractive 특정 배포의 컨텍스트에서 WSL(대화형 Linux용 Windows 하위 시스템) 프로세스를 시작합니다.

WslRegisterDistribution WSL(Linux용 Windows 하위 시스템)에 새 배포를 등록합니다.
WslUnregisterDistribution Linux용 Windows 하위 시스템(WSL)에서 배포를 등록 취소합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Windows 동기화

아티클2023. 08. 22.

Windows 동기화 기술 개요입니다.

Windows 동기화를 개발하려면 다음 헤더가 필요합니다.

- syncregistration.h
- winsync.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 동기화

열거형

CONFLICT_RESOLUTION_POLICY 동기화 세션에 사용할 동시성 충돌 해결 정책의 옵션을 나타냅니다.
FILTERING_TYPE 필터링된 동기화 중에 변경 일괄 처리에 포함된 정보의 유형을 나타냅니다.
KNOWLEDGE_COOKIE_COMPARISON_RESULT ISyncKnowledge2::CompareToKnowledgeCookie를 사용하여 지식 쿠키를 지식 개체와 비교할 때 가능한 결과를 나타냅니다.
SYNC_FULL_ENUMERATION_ACTION ISyncCallback::OnFullEnumerationNeeded에 대한 응답으로 애플리케이션에서 수행할 작업을 나타냅니다.
SYNC_PROGRESS_STAGE 동기화 세션의 단계를 나타냅니다.
SYNC_PROVIDER_ROLE 동기화 세션의 다른 공급자를 기준으로 공급자의 역할을 나타냅니다.
SYNC_REGISTRATION_EVENT 다양한 유형의 동기화 등록 이벤트를 나타냅니다.

SYNC_RESOLVE_ACTION 특정 동시성 충돌을 resolve 위해 수행되는 작업을 나타냅니다.
SYNC_SERIALIZATION_VERSION 특정 구성 요소가 호환되는 Microsoft Sync Framework 버전을 나타냅니다.
SYNC_STATISTICS 구성 요소에 대한 정보를 전달하는 통계 유형을 나타냅니다.

Functions

AddChangeUnitMetadata 항목 변경에 변경 단위 메타데이터를 추가합니다.
AddFilter 파생 클래스에서 구현할 때 원본 공급자의 변경 열거에 사용되는 필터를 설정합니다.
AddItemMetadataToGroup 현재 열려 있는 그룹에 지정된 항목 변경을 추가합니다.
AddLoggedConflict 충돌을 나타내는 메타데이터를 변경 일괄 처리에 추가합니다.
BeginOrderedGroup 변경 일괄 처리에서 정렬된 그룹을 엽니다. 이 그룹은 항목 ID별로 정렬됩니다.
BeginSession 공급자가 동기화 세션에 참가하고 있음을 알 수 있습니다.
BeginUnorderedGroup 변경 일괄 처리에서 순서가 지정되지 않은 그룹을 엽니다. 이 그룹의 항목 변경 내용은 순서에 따라 다를 수 있습니다.
복제

열거자를 복제하고 현재 열거자와 동일한 상태에 있는 새 열거자를 반환합니다. (IEnumChangeUnitExceptions.Clone)
복제 열거자를 복제하고 현재 열거자와 동일한 상태에 있는 새 열거자를 반환합니다. (IEnumClockVector.Clone)
복제 열거자를 복제하고 현재 열거자와 동일한 상태에 있는 새 열거자를 반환합니다. (IEnumFeedClockVector.Clone)
복제 열거자를 복제하고 현재 열거자와 동일한 상태에 있는 새 열거자를 반환합니다. (IEnumRangeExceptions.Clone)
복제 열거자를 복제하고 현재 열거자와 동일한 상태에 있는 새 열거자를 반환합니다. (IEnumSingleItemExceptions.Clone)
복제 이 메서드가 구현되지 않은 경우 (IEnumSyncChanges.Clone)
복제 이 메서드가 구현되지 않은 경우 (IEnumSyncChangeUnits.Clone)
복제 이 개체의 새 instance 만들고 이 개체의 데이터를 새 개체로 복사합니다.
복제 열거자를 복제하고 현재 열거자와 동일한 상태에 있는 새 열거자를 반환합니다. (IEnumSyncProviderConfigUIInfos.Clone)
복제 열거자를 복제하고 현재 열거자와 동일한 상태에 있는 새 열거자를 반환합니다. (IEnumSyncProviderInfos.Clone)
CompareToKnowledgeCookie 지정된 지식 쿠키와 이 지식 개체를 빠르게 비교합니다.

보완 이 개체에 포함되어 있지만 지정된 지식에 포함되지 않은 지식을 반환합니다.
ContainsChange 지정한 항목 변경 내용이 이 지식에 의해 알려져 있는지 여부를 나타냅니다.
ContainsChangeUnit 지정된 변경 단위 변경 내용이 이 지식에 의해 알려져 있는지 여부를 나타냅니다.
ContainsKnowledge 지정된 지식이 이 지식으로 알려져 있는지 여부를 나타냅니다.
ContainsKnowledgeForChangeUnit 지정된 변경 단위에 대한 지정된 지식이 이 지식에 의해 알려져 있는지 여부를 나타냅니다.
ContainsKnowledgeForItem 지정된 항목에 대한 지정된 지식이 이 지식에 의해 알려져 있는지 여부를 나타냅니다.
ConvertFullEnumerationChangeBatchToRegularChangeBatch ISyncFullEnumerationChangeBatch 개체를 ISyncChangeBatch 개체로 변환합니다.
ConvertVersion 다른 복제본(replica) 버전을 이 지식을 소유한 복제본(replica) 호환되는 버전으로 변환합니다.
CreateAndRegisterNewSyncProvider 새 동기화 공급자를 만들고 등록합니다.
CreateSyncProviderConfigUIRegistrationInstance 동기화 공급자 구성 UI의 메모리 내 instance 만듭니다.
CreateSyncProviderRegistrationInstance 동기화 공급자의 메모리 내 instance 만듭니다.
EndOrderedGroup 변경 일괄 처리에서 이전에 연 정렬된 그룹을 닫습니다.
EndSession

참가한 동기화 세션이 완료되었음을 공급자에게 알렸습니다.
EndUnorderedGroup 변경 일괄 처리에서 이전에 열린 순서가 지정되지 않은 그룹을 닫습니다.
EnumerateSyncProviderConfigUIs 지정된 조건에 대해 등록된 모든 ISyncProviderConfigUIInfo 개체를 열거하는 IEnumSyncProviderConfigUIInfos 열거형 인터페이스를 반환합니다.
EnumerateSyncProviders 지정된 조건에 대해 등록된 모든 ISyncProviderInfo 개체를 열거하는 IEnumSyncProviderInfos 열거형 인터페이스를 반환합니다.
ExcludeChangeUnit 지식에서 지정된 변경 단위에 대한 지식을 제거합니다.
ExcludeItem 지식에서 지정된 항목에 대한 지식을 제거합니다.
FindClockVectorForChangeUnit 지정된 변경 단위 ID와 연결된 클록 벡터를 가져옵니다.
FindClockVectorForItem 지정된 항목 ID와 연결된 클록 벡터를 가져옵니다.
FindMinTickCountForReplica 지정된 복제본(replica) 대한 지식에서 최소 틱 수를 찾습니다.
FindOrAddReplica IReplicaKeyMap 개체에 항목을 추가하거나 항목을 찾습니다.
ForgetToVersion 지정된 버전보다 작거나 같은 모든 버전이 잊혀졌을 수 있으며 해당 삭제 표시가 삭제되었을 수 있음을 반영하기 위해 잊어버린 지식을 업데이트.
GetBatchLevelKnowledgeShouldBeApplied 일괄 처리가 대상 복제본(replica) 적용된 후 일괄 처리에 대해 학습된 지식을 저장해야 하는지 여부를 나타내는 값을 가져옵니다.

GetChange 새 등록 이벤트를 나타내는 ISyncRegistrationChange 개체를 가져옵니다.
GetChangeBatch 대상 공급자의 지정된 지식에 포함되지 않은 항목에 대한 항목 메타데이터를 포함하는 변경 일괄 처리를 가져옵니다.
GetChangeEnumerator 이 변경 일괄 처리에서 항목 변경 내용을 열거하는 IEnumSyncChanges 개체를 가져옵니다.
GetChangeUnitChangeTime 지정된 변경 단위가 마지막으로 변경된 날짜와 시간을 가져옵니다.
GetChangeUnitExceptions 지식에 저장된 IChangeUnitException 개체를 열거할 수 있는 개체를 가져옵니다.
GetChangeUnitId 예외와 연결된 변경 단위의 변경 단위 ID를 가져옵니다.
GetChangeUnitId 필터를 정의하는 변경 단위 ID 배열의 지정된 인덱스에 저장된 변경 단위 ID를 가져옵니다.
GetChangeUnitId 이 변경 단위의 ID를 검색합니다.
GetChangeUnitIdCount 필터를 정의하는 변경 단위 ID 수를 가져옵니다.
GetChangeUnits 이 변경 내용에 포함된 변경 단위를 열거할 수 있는 개체를 가져옵니다.
GetChangeUnitVersion 변경 단위 변경의 버전을 가져옵니다.
GetChangeVersion 이 변경 내용과 연결된 버전을 가져옵니다.

GetChangeWithRecoverableError 오류를 발생시킨 항목 변경을 가져옵니다.
GetClockVector 이 예외와 연결된 클록 벡터를 가져옵니다. (IChangeUnitException.GetClockVector)
GetClockVector 이 예외와 연결된 클록 벡터를 가져옵니다. (IRangeException.GetClockVector)
GetClockVector 항목 예외와 연결된 클록 벡터를 가져옵니다.
GetClockVectorElementCount 클록 벡터에 포함된 요소 수를 가져옵니다.
GetClockVectorElements 클록 벡터 요소를 반복하는 열거자를 반환합니다.
GetClosedLowerBoundItemId 대상 버전이 필요한 항목 ID에서 닫힌 하한을 가져옵니다.
GetClosedRangeEnd 제외할 항목 ID 범위의 상한을 가져옵니다.
GetClosedRangeStart 제외할 항목 ID 범위의 하한을 가져옵니다.
GetClosedUpperBoundItemId 대상 버전이 필요한 항목 ID의 닫힌 상한을 가져옵니다.
GetColumnCount 이 기술 자료에 포함된 열 수를 가져옵니다.
GetCreationVersion 변경된 항목의 생성 버전을 가져옵니다.
GetDestinationProviderConflictingChange

대상 공급자에서 변경 메타데이터를 가져옵니다.
GetDestinationProviderConflictingData 대상 복제본(replica) 변경 항목에 대한 항목 데이터를 검색하는 데 사용할 수 있는 개체를 가져옵니다.
GetErrorDescription 오류에 대한 설명을 가져옵니다.
GetEvent 보류 중인 다음 등록 이벤트를 가져옵니다.
GetFilterInfo 변경 일괄 처리를 만들 때 지정한 ISyncFilterInfo를 가져옵니다.
GetFlags 클록 벡터 요소에 대한 추가 정보를 지정하는 플래그를 가져옵니다.
GetFlags 이 변경 내용과 연결된 플래그를 가져옵니다.
GetFlags 필터 정보 개체에 대한 추가 정보를 지정하는 플래그를 가져옵니다.
GetForgottenKnowledgeRecoveryRangeEnd 세션이 잊어버린 지식 복구를 수행할 때 복구 범위의 상한을 가져옵니다.
GetForgottenKnowledgeRecoveryRangeStart 세션이 잊어버린 지식 복구를 수행할 때 복구 범위의 하한을 가져옵니다.
GetFullEnumerationChangeBatch ID가 지정된 하한보다 큰 항목에 대한 항목 메타데이터를 전체 열거형의 일부로 포함하는 변경 일괄 처리를 가져옵니다.
GetIdParameters 공급자의 ID 형식 스키마를 가져옵니다. (IAsynchronousDataRetriever.GetIdParameters)
GetIdParameters

공급자의 ID 형식 스키마를 가져옵니다. (ISynchronousDataRetriever.GetIdParameters)
GetIdParameters 공급자의 ID 형식 스키마를 가져옵니다. (ISyncKnowledge2.GetIdParameters)
GetIdParameters 공급자의 ID 형식 스키마를 가져옵니다. (ISyncProvider.GetIdParameters)
GetInfoForChangeApplication 직렬화된 변경 적용자를 위해 저장된 데이터를 검색합니다.
GetInspector 기술 개체의 내용을 검색하는 데 사용할 수 있는 개체를 반환합니다.
GetInstanceId 동기화 공급자의 instance ID를 반환합니다.
GetInstanceId 이벤트와 연결된 동기화 공급자 또는 동기화 공급자 구성 UI의 instance ID를 가져옵니다.
GetIsLastBatch 이 변경 일괄 처리의 변경 내용이 동기화 세션의 마지막 일괄 처리인지 여부를 나타내는 플래그를 가져옵니다.
GetItemChange 이 변경 단위 변경 내용이 포함된 항목 변경을 가져옵니다.
GetItemChangeTime 지정된 항목이 마지막으로 변경된 날짜와 시간을 가져옵니다.
GetItemDisplayName 오류를 발생시킨 항목의 표시 이름을 가져옵니다.
GetItemId 예외와 연결된 변경 단위가 포함된 항목의 항목 ID를 가져옵니다.
GetItemId 예외에 지정된 항목의 ID를 가져옵니다.

GetKnowledgeCookie 빠른 비교에 사용할 수 있는 이 지식 개체의 간단한 읽기 전용 표현을 가져옵니다.
GetLearnedForgottenKnowledge 대상 공급자가 복구 동기화 중에 이 변경 일괄 처리의 모든 변경 내용을 적용할 때 대상 복제본(replica) 학습한다는 잊어버린 지식을 가져옵니다.
GetLearnedForgottenKnowledge 대상 공급자가 복구 동기화 중에 이 변경 내용을 적용할 때 대상 복제본(replica) 학습한다는 잊어버린 지식을 가져옵니다.
GetLearnedKnowledge 이 변경 내용이 해당 항목 저장소에 적용될 때 복제본(replica) 학습할 지식을 가져옵니다.
GetLearnedKnowledge 대상 공급자가 이 변경 일괄 처리의 모든 변경 내용을 적용할 때 대상 복제본(replica) 학습한다는 지식을 가져옵니다.
GetLearnedKnowledgeAfterRecoveryComplete 전체 열거형의 변경 내용을 적용한 후 대상 복제본(replica) 학습할 지식을 가져옵니다.
GetLearnedKnowledgeAfterRecoveryComplete 복구 동기화의 모든 변경 내용을 적용한 후 대상 복제본(replica) 학습할 지식을 가져옵니다.
GetLearnedKnowledgeWithPrerequisite 대상 공급자가 변경 일괄 처리의 필수 구성 요소 지식을 기반으로 이 변경 일괄 처리의 모든 변경 내용을 적용할 때 대상 복제본(replica) 학습하는 지식을 가져옵니다.
GetLearnedKnowledgeWithPrerequisite 대상 공급자가 변경 내용과 관련된 필수 구성 요소 지식을 기반으로 이 변경 내용을 적용할 때 대상 복제본(replica) 학습하는 지식을 가져옵니다.
GetLowestUncontainedId 이 지식에 포함되지 않고 지정된 지식에 포함된 가장 낮은 항목 ID를 반환합니다.
GetMadeWithKnowledge 이 변경 내용에 대한 정보를 가져옵니다.

GetMinimumSupportedVersion 이 개체와 함께 사용할 수 있는 지원되는 최소 버전의 Microsoft Sync Framework 구성 요소를 가져옵니다.
GetOwnerReplicaId 이 변경을 시작한 복제본(replica) ID를 가져옵니다.
GetOwnerReplicaId 이 지식을 소유한 복제본(replica) ID를 가져옵니다.
GetPrerequisiteKnowledge 대상 공급자가 이 변경 일괄 처리를 처리하는 데 필요한 최소 정보를 가져옵니다.
GetPrerequisiteKnowledge 대상 공급자가 이 변경을 처리하는 데 필요한 최소 정보를 가져옵니다.
GetProvider 항목 변경을 건너뛴 공급자의 역할을 가져옵니다.
GetRangeCount 이 기술 조각에 포함된 범위 수를 가져옵니다.
GetRangeExceptions 지식에 저장된 IRangeException 개체를 열거할 수 있는 개체를 가져옵니다.
GetRecoverableErrorDataForChange 복구 가능한 오류에 대한 추가 데이터를 가져옵니다.
GetRecoverableErrorDataForChangeUnit 지정된 변경 단위의 복구 가능한 오류에 대한 추가 데이터를 가져옵니다.
GetRegisteredProperties 읽기 및 쓰기를 위한 구성 UI 속성을 가져옵니다.
GetRemainingWorkEstimateForSession 세션에 대한 나머지 작업의 예상을 가져옵니다.

GetReplicaKey 이 클록 벡터 요소와 연결된 복제본(replica) 대한 복제본(replica) 키를 가져옵니다.
GetReplicaKeyMap 이 지식과 연결된 IReplicaKeyMap 개체를 가져옵니다.
GetResolveActionForChange 충돌에 대한 충돌 해결 작업을 가져옵니다.
GetResolveActionForChangeUnit 충돌하는 변경 단위 변경에 대한 충돌 해결 작업을 가져옵니다.
GetRootItemId 변경된 항목의 ID를 가져옵니다.
GetScopeVector 지식에 포함된 변경 내용을 정의하는 클록 벡터를 가져옵니다.
GetSessionErrorStatus 동기화 세션이 실패한 이유를 나타내는 오류 값을 가져옵니다.
GetSingleItemExceptions 지식에 저장된 ISingleItemException 개체를 열거할 수 있는 개체를 가져옵니다.
GetSourceForgottenKnowledge 원본 복제본(replica) 대해 잊어버린 지식을 가져옵니다.
GetSourceProviderConflictingChange 원본 공급자에서 변경 메타데이터를 가져옵니다.
GetSourceProviderConflictingData 원본 복제본(replica) 변경 항목에 대한 항목 데이터를 검색하는 데 사용할 수 있는 개체를 가져옵니다.
GetStage 오류가 발생했을 때 동기화 세션의 단계를 가져옵니다.

GetStatistics 이 개체에 포함된 지정된 통계 데이터를 가져옵니다.
GetSyncBatchParameters 변경 일괄 처리에 포함될 요청된 항목 변경 횟수와 동기화 scope 대한 현재 정보를 가져옵니다.
GetSyncChange 항목 저장소에서 변경 데이터를 검색할 변경 항목을 가져옵니다.
GetSyncProvider 동기화 공급자의 instance 만듭니다.
GetSyncProviderConfigUI 동기화 공급자 구성 UI의 instance 만듭니다.
GetSyncProviderConfigUIFromInstanceId 지정된 고유 instance ID에 대해 초기화되고 인스턴스화된 ISyncProviderConfigUI 개체를 반환합니다.
GetSyncProviderConfigUIInfo 지정된 고유 instance ID에 대한 ISyncProviderConfigUIInfo 개체를 반환합니다.
GetSyncProviderConfigUIInfoforProvider 지정된 동기화 공급자 instance ID에 대한 ISyncProviderConfigUIInfo 개체를 반환합니다.
GetSyncProviderFromInstanceId 특정 고유 instance ID에 대해 초기화되고 인스턴스화된 IRegisteredSyncProvider 개체를 반환합니다.
GetSyncProviderInfo 특정 동기화 공급자 instance ID에 대한 ISyncProviderInfo 개체를 반환합니다.
GetSyncProviderState 지정된 동기화 공급자의 상태를 반환합니다.
GetSyncProviderWithError 동기화 실패를 발생시킨 공급자의 ISyncProvider 인터페이스를 가져옵니다.

GetSyncTime 항목의 when 값에 해당하는 SYNC_TIME 값을 가져옵니다.
GetTickCount 이 클록 벡터 요소에 포함된 틱 개수 범위의 상한을 정의하는 틱 수를 가져옵니다.
GetUpdateCount FeedSync 항목에 대한 업데이트 수를 가져옵니다.
GetUpperBoundItemId 변경 일괄 처리의 모든 그룹에 대한 지식으로 표현되는 가장 높은 항목 ID를 가져옵니다.
GetVersion 이 기술 구조의 버전을 가져옵니다.
GetWorkEstimate 이 변경에 대한 작업 예상을 가져옵니다.
GetWorkEstimateForBatch 일괄 처리에 대한 작업 예상을 가져옵니다.
Init 동기화 세션이 준비되기 전에 동기화 공급자를 초기화합니다.
Init 동기화 공급자에 대한 구성 UI를 초기화합니다.
초기화 지정된 변경 단위 ID 배열을 포함하는 IChangeUnitListFilterInfo 클래스의 새 instance 초기화합니다.
초기화 파생 클래스에서 구현되는 경우 IKnowledgeSyncProvider로 변환할 ISyncProvider 개체를 사용하여 IProviderConverter 개체를 초기화합니다.
초기화 오류를 발생시킨 항목의 지정된 표시 이름과 오류에 대한 설명을 사용하여 개체를 초기화합니다.

IntersectsWithKnowledge 지정된 지식이 이 지식과 교차하는지 여부를 나타냅니다.
IsCanceled 동기화 세션이 취소되었는지 여부를 나타냅니다.
IsNoConflictsSpecified FeedSync 항목에 대해 충돌이 유지되는지 여부를 나타내는 값을 가져옵니다.
LoadChangeData 변경에 대한 항목 데이터를 검색합니다. (IAsynchronousDataRetriever.LoadChangeData)
LoadChangeData 변경에 대한 항목 데이터를 검색합니다. (ISynchronousDataRetriever.LoadChangeData)
LoadChangeDataComplete IAsynchronousDataRetriever::LoadChangeData가 성공적으로 완료되었음을 나타냅니다.
LoadChangeDataError IAsynchronousDataRetriever 메서드가 실패했음을 나타냅니다.
LoadInfoFromChangeApplication 직렬화된 변경 적용자를 위한 데이터를 저장합니다.
LookupReplicaId 지정된 복제본(replica) 키에 해당하는 복제본(replica) ID를 가져옵니다.
LookupReplicaKey 지정된 복제본(replica) ID에 해당하는 복제본(replica) 키를 가져옵니다.
MapRemoteToLocal 지식 개체를 다른 복제본(replica) 이 지식을 소유하는 복제본(replica) 호환되는 개체로 변환합니다.
ModifySyncProvider 이 ISyncProviderConfigUI에 의해 구성된 동기화 공급자의 ISyncProviderInfo를 업데이트.
다음

사용 가능한 경우 변경 단위 예외 집합의 다음 요소를 반환합니다. (IEnumChangeUnitExceptions.Next)
다음 사용할 수 있는 경우 클록 벡터의 다음 요소를 반환합니다.
다음 사용 가능한 경우 클록 벡터의 다음 요소를 반환합니다.
다음 사용 가능한 경우 변경 단위 예외 집합의 다음 요소를 반환합니다. (IEnumRangeExceptions.Next)
다음 사용할 수 있는 경우 단일 항목 예외 집합의 다음 요소를 반환합니다.
다음 다음 항목 변경을 반환합니다.
다음 다음 변경 단위를 반환합니다.
다음 다음 ISyncProviderConfigUIInfo 개체를 반환합니다.
다음 다음 ISyncProviderInfo 개체를 반환합니다.
NextColumn 이 기술 조각이 적용되는 변경 단위 ID 집합의 다음 변경 단위 ID를 반환합니다.
NextCoreFragments 사용 가능한 경우 지식의 다음 ICoreFragment 개체를 반환합니다.
NextRange 이 기술 조각에 포함된 다음 범위와 범위의 항목에 대해 알려진 내용을 정의하는 클록 벡터를 반환합니다.
Onchange

변경 내용이 적용되기 전에 발생합니다.
OnChangeApplied 변경 내용이 성공적으로 적용된 후에 발생합니다.
OnChangeFailed 변경 내용이 적용되지 않으면 발생합니다.
OnConflict 동시성 충돌 해결 정책이 CRP_NONE 설정될 때 충돌이 감지될 때 발생합니다.
OnFullEnumerationNeeded 원본 공급자의 잊어버린 지식이 대상 공급자의 현재 지식에 포함되지 않은 경우에 발생합니다.
OnProgress 진행률을 보고하기 위해 동기화 세션 중에 주기적으로 발생합니다.
OnProgress 애플리케이션에 동기화 진행률을 보고합니다.
OnRecoverableError 동기화 공급자가 항목을 로드하거나 저장할 때 복구 가능한 오류를 설정하는 경우에 발생합니다.
ProcessChangeBatch 충돌을 감지하고 항목 저장소에 변경 내용을 적용하여 변경 내용 집합을 처리합니다.
ProcessFullEnumerationChangeBatch 항목 저장소에 변경 내용을 적용하여 전체 열거형에 대한 변경 내용 집합을 처리합니다.
ProjectOntoChangeUnit 지정된 변경 단위에 대한 정보를 가져옵니다.
ProjectOntoColumnSet 이 개체에 포함된 모든 항목에 대해 지정된 변경 단위 집합에 대한 지식을 반환합니다.
ProjectOntoItem 지정된 항목에 대한 정보를 가져옵니다.

ProjectOntoKnowledgeWithPrerequisite 템플릿 지식에 지정된 조각에 대한 필수 구성 요소 지식이 포함된 경우 템플릿 지식에서 지정한 지식 조각에 대한 지식을 반환합니다.
ProjectOntoRange 지정된 항목 ID 범위에 대한 정보를 가져옵니다.
RegisterCallback 비동기 메서드가 처리를 완료할 때 IAsynchronousDataRetriever 개체에서 호출할 콜백 인터페이스를 등록합니다.
RegisterForEvent 등록 저장소를 변경할 때 발생하는 새 등록 이벤트의 도착 알림을 받도록 사용자를 등록합니다.
RequestFilter 대상 공급자가 지정한 필터를 변경 열거 중에 원본 공급자가 사용할 것을 요청합니다.
Reset 열과 범위 열거자를 모두 해당 집합의 시작 부분으로 다시 설정합니다.
Reset 열거자를 지식의 시작 부분으로 다시 설정합니다.
Reset 열거자를 변경 단위 예외 집합의 시작 부분으로 다시 설정합니다.
Reset 열거자를 클록 벡터의 시작 부분으로 다시 설정합니다. (IEnumClockVector.Reset)
Reset 열거자를 클록 벡터의 시작 부분으로 다시 설정합니다. (IEnumFeedClockVector.Reset)
Reset 열거자를 범위 예외 집합의 시작 부분으로 다시 설정합니다.
Reset 열거자를 단일 항목 예외 집합의 시작 부분으로 다시 설정합니다.

Reset 열거자를 목록의 시작 부분으로 다시 설정합니다. (IEnumSyncChanges.Reset)
Reset 열거자를 목록의 시작 부분으로 다시 설정합니다. (IEnumSyncChangeUnits.Reset)
Reset 열거자를 ISyncProviderConfigUIInfo 개체 컬렉션의 시작 부분으로 다시 설정합니다.
Reset 열거자를 ISyncProviderInfo 집합의 시작 부분으로 다시 설정합니다.
Reset 다음 동기화 세션에서 새 복제본(replica) 표시되도록 동기화 공급자를 다시 설정합니다.
RevokeCallback IAsynchronousDataRetriever 개체가 지정된 콜백 인터페이스를 더 이상 사용하지 않아야 하며 이에 대한 참조를 해제해야 했음을 나타냅니다.
RevokeEvent 새 등록 이벤트의 도착 알림에서 사용자를 등록 취소합니다.
직렬화 복제본(replica) 키 맵 데이터를 바이트 배열로 직렬화합니다.
직렬화 변경 일괄 처리를 바이트 배열로 직렬화합니다.
직렬화 필터 데이터를 바이트 배열로 직렬화합니다.
직렬화 지식 개체 데이터를 바이트 배열로 직렬화합니다.
SerializeWithOptions 지정된 버전 및 serialization 옵션에 따라 변경 일괄 처리 개체 데이터를 바이트 배열로 직렬화합니다.

SerializeWithOptions 지정된 버전 및 serialization 옵션에 따라 지식 개체 데이터를 바이트 배열로 직렬화합니다.
SetForgottenKnowledgeRecoveryRange 세션이 잊어버린 지식 복구를 수행할 때 복구 범위를 설정합니다.
SetLastBatch 동기화 세션에서 열거할 변경 내용이 더 이상 없음을 나타내는 플래그를 설정합니다.
SetLocalTickCount 이 지식을 소유한 복제본(replica) 대한 틱 수를 설정합니다.
SetPrerequisiteKnowledge 대상 공급자가 이 변경 일괄 처리를 처리하는 데 필요한 최소 정보를 설정합니다.
SetProviderWithError 동기화가 실패하게 된 공급자를 나타냅니다.
SetRecoverableErrorOnChange 이 변경 내용에 대해 복구 가능한 오류가 발생했음을 나타냅니다.
SetRecoverableErrorOnChangeUnit 지정된 변경 단위의 데이터가 항목 저장소에서 로드되었을 때 복구 가능한 오류가 발생했음을 나타냅니다.
SetRemainingWorkEstimateForSession 세션에 대한 나머지 작업의 예상값을 설정합니다.
SetResolveActionForChange 충돌에 대한 충돌 해결 작업을 설정합니다.
SetResolveActionForChangeUnit 충돌하는 변경 단위 변경에 대한 충돌 해결 작업을 설정합니다.
SetSyncProviderState 지정된 동기화 공급자의 상태를 설정합니다.

SetWorkEstimate 이 변경에 대한 작업 예상을 설정합니다.
SetWorkEstimateForBatch 일괄 처리에 대한 작업 예상을 설정합니다.
Skip 지정된 개수의 변경 단위 예외를 건너뜁니다.
Skip 지정된 수의 클록 벡터 요소를 건너뜁니다. (IEnumClockVector.Skip)
Skip 지정된 수의 클록 벡터 요소를 건너뜁니다. (IEnumFeedClockVector.Skip)
Skip 지정된 범위 예외 수를 건너뜁니다.
Skip 지정된 수의 단일 항목 예외를 건너뜁니다.
Skip 이 메서드가 구현되지 않은 경우 (IEnumSyncChanges.Skip)
Skip 이 메서드가 구현되지 않은 경우 (IEnumSyncChangeUnits.Skip)
Skip 지정된 수의 ISyncProviderConfigUIInfo 개체를 건너뜁니다.
Skip 지정된 수의 ISyncProviderInfo 개체를 건너뜁니다.
SpecifyFilter 파생 클래스에서 구현되는 경우 변경 열거 중에 원본 공급자가 사용하는 필터를 협상합니다.
Union

지정된 지식을 현재 지식과 결합합니다.
UnregisterSyncProvider 등록 저장소에서 지정된 동기화 공급자의 등록을 취소하고 제거합니다.
UnregisterSyncProviderConfigUI 등록 저장소에서 지정된 동기화 공급자 구성 UI의 등록을 취소하고 제거합니다.

인터페이스

IAsynchronousDataRetriever 대상 공급자가 원본 공급자에서 항목 데이터를 비동기적으로 검색하는 메커니즘을 나타냅니다.
IChangeConflict 두 항목 간의 충돌을 나타냅니다.
IChangeUnitException 기술 개체에서 제외할 변경 단위를 나타냅니다.
IChangeUnitListFilterInfo ISyncChangeBatch 개체의 항목에 포함되는 변경 단위를 제어하는 데 사용할 수 있는 필터를 나타냅니다.
IClockVector 기술 구조의 클록 벡터를 나타냅니다.
IClockVectorElement 기술 구조의 클록 벡터 요소를 나타냅니다.
IConstructReplicaKeyMap IReplicaKeyMap 개체에 항목을 추가합니다.
ICoreFragment 특정 변경 단위 집합에 대한 scope 모든 항목에 대한 지식을 나타냅니다.

ICoreFragmentInspector 기술 개체에 포함된 ICoreFragment 개체를 열거합니다.
IDataRetrieverCallback IAsynchronousDataRetriever 메서드에서 처리가 완료되었음을 나타내기 위해 IAsynchronousDataRetriever 개체가 호출할 수 있는 메서드를 나타냅니다.
IEnumChangeUnitExceptions 기술 개체에 저장된 변경 단위 예외를 열거합니다.
IEnumClockVector 클록 벡터에 저장된 클록 벡터 요소를 열거합니다.
IEnumFeedClockVector FeedSync 정보를 포함하는 클록 벡터에 저장된 클록 벡터 요소를 열거합니다.
IEnumRangeExceptions 기술 개체에 저장된 범위 예외를 열거합니다.
IEnumSingleItemExceptions 지식 개체에 저장된 단일 항목 예외를 열거합니다.
IEnumSyncChanges 항목 변경 목록을 열거합니다.
IEnumSyncChangeUnits 변경 단위 목록을 열거합니다.
IEnumSyncProviderConfigUIInfos 동기화 공급자를 빌드하고 등록하는 데 사용되는 구성 UI 정보를 포함하는 ISyncProviderConfigUIInfo 개체를 열거합니다.
IEnumSyncProviderInfos 동기화 공급자의 instance 만드는 데 사용되는 정보가 포함된 ISyncProviderInfo 개체를 열거합니다.
IFeedClockVector FeedSync 정보를 포함하는 클록 벡터를 나타냅니다.

IFeedClockVectorElement FeedSync 정보를 포함하는 클록 벡터 요소를 나타냅니다.
IFilterRequestCallback 중재는 대상 공급자와 원본 공급자 간의 필터 협상을 필터링합니다.
IForgottenKnowledge 삭제 표시 정리로 인해 잊혀진 지식을 나타냅니다.
IKnowledgeSyncProvider 지식을 사용하여 동기화를 수행하는 동기화 공급자를 나타냅니다.
ILoadChangeContext 항목 저장소에서 로드할 변경 내용에 대한 정보를 나타냅니다.
IProviderConverter 파생 클래스에서 구현되는 경우 는 ISyncProvider 개체를 IKnowledgeSyncProvider 개체로 변환할 수 있는 개체를 나타냅니다.
IRangeException 기술 개체에서 제외할 항목 ID 범위를 나타냅니다.
IRecoverableError 항목이 로드되거나 항목이 저장되었을 때 발생한 복구 가능한 오류를 나타냅니다.
IRecoverableErrorData 복구 가능한 오류에 대한 정보를 나타냅니다.
IRegisteredSyncProvider 등록된 동기화 공급자를 나타냅니다. 이 인터페이스는 동기화 공급자의 작성기에 의해 구현됩니다.
IReplicaKeyMap 복제본(replica) 키와 복제본(replica) ID 간의 매핑을 나타냅니다.
IRequestFilteredSync 파생 클래스에서 구현되는 경우 는 변경 열거 중에 원본 공급자가 사용할 필터를 지정할 수 있는 대상 공급자를 나타냅니다.

ISingleItemException 기술 개체에서 제외할 항목을 나타냅니다.
ISupportFilteredSync 파생 클래스에서 구현되는 경우 는 필터링된 변경 열거를 지원하고 사용되는 필터 유형을 협상할 수 있는 원본 공급자를 나타냅니다.
ISupportLastWriteTime 항목 또는 변경 단위가 마지막으로 변경된 날짜와 시간을 보고할 수 있는 동기화 공급자를 나타냅니다. 이 기능은 마지막 기록기-승리 충돌 해결을 구현하는 애플리케이션에 유용합니다.
ISyncCallback 애플리케이션에 동기화 이벤트를 알리는 데 사용되는 애플리케이션 콜백을 나타냅니다.
ISyncCallback2 애플리케이션에 동기화 이벤트를 알리는 데 사용되는 추가 애플리케이션 콜백을 나타냅니다.
ISyncChange 항목에 대한 변경 사항을 나타냅니다.
ISyncChangeBatch 변경 내용 집합에 대한 메타데이터를 나타냅니다. (ISyncChangeBatch)
ISyncChangeBatchAdvanced 변경 내용 집합에 대한 추가 정보를 나타냅니다.
ISyncChangeBatchBase 변경 내용 집합에 대한 메타데이터를 나타냅니다. (ISyncChangeBatchBase)
ISyncChangeBatchBase2 ISyncChangeBatchBase 개체의 추가 기능을 나타냅니다.
ISyncChangeBatchWithPrerequisite 변경 일괄 처리와 관련된 필수 구성 요소 지식을 기반으로 하는 변경 일괄 처리에 대한 메타데이터를 나타냅니다.
ISyncChangeBuilder

항목 변경에 대한 추가 데이터를 제공합니다.
ISyncChangeUnit 항목에 포함된 변경 단위에 대한 변경 사항을 나타냅니다.
ISyncChangeWithPrerequisite 변경과 관련된 필수 구성 요소 지식을 기반으로 하는 변경 내용에 대한 메타데이터를 나타냅니다.
ISyncFilterInfo ISyncChangeBatch 개체에 포함된 데이터를 제어하는 데 사용되는 필터에 대한 정보를 나타냅니다.
ISyncFilterInfo2 ISyncChangeBatch 개체에 포함된 변경 내용을 제어하는 데 사용할 수 있는 필터에 대한 추가 정보를 나타냅니다.
ISyncFullEnumerationChange 복구 동기화 중 ISyncChange 개체에 대한 추가 정보를 나타냅니다.
ISyncFullEnumerationChangeBatch 복구 동기화의 일부로 만들어진 변경 내용 집합에 대한 메타데이터를 나타냅니다.
ISynchronousDataRetriever 대상 공급자가 원본 공급자에서 항목 데이터를 검색하는 메커니즘을 나타냅니다.
ISyncKnowledge 복제본(replica) 항목 저장소에 대해 가지고 있는 지식을 나타냅니다.
ISyncKnowledge2 복제본(replica) 항목 저장소에 대한 지식에 대한 추가 정보를 나타냅니다.
ISyncProvider 동기화 세션에서 데이터를 다른 동기화 공급자와 동기화하는 데 사용할 수 있는 동기화 공급자를 나타냅니다.
ISyncProviderConfigUI 동기화 공급자를 빌드하고 등록하는 데 사용되는 구성 UI 정보를 나타냅니다.
ISyncProviderConfigUIInfo

동기화 공급자 구성 UI의 instance 만드는 데 필요한 정보 및 속성을 나타냅니다.
ISyncProviderInfo 동기화 공급자의 instance 만드는 데 필요한 정보 및 속성을 나타냅니다.
ISyncProviderRegistration 동기화 공급자 등록을 나타냅니다.
ISyncRegistrationChange 동기화 공급자 또는 동기화 공급자 구성 UI 등록에 대한 변경 사항을 나타냅니다. 변경 내용은 등록 이벤트로 보고됩니다.
ISyncSessionExtendedErrorInfo 동기화가 실패한 공급자에 대한 정보를 나타냅니다.
ISyncSessionState 현재 동기화 세션에 대한 정보를 나타냅니다.
ISyncSessionState2 현재 동기화 세션에 대한 추가 정보를 나타냅니다.

구조체

ID_PARAMETER_PAIR 동기화 엔터티 ID의 형식을 나타냅니다.
ID_PARAMETERS 동기화 세션에서 엔터티를 식별하는 데 사용되는 ID 그룹의 형식 스키마를 나타냅니다.
SYNC_RANGE 항목 ID의 범위를 나타냅니다.
SYNC_SESSION_STATISTICS 단일 단방향 동기화 세션에 대한 통계를 나타냅니다.

SYNC_TIME 날짜 및 시간 값을 나타냅니다.
SYNC_VERSION 항목 또는 변경 단위의 버전을 나타냅니다.
SyncProviderConfigUIConfiguration 동기화 공급자 구성 UI에 대한 정보를 나타냅니다.
SyncProviderConfiguration 동기화 공급자 구성에 대한 정보를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Windows 시스템 평가 도구

아티클2023. 08. 22.

Windows 시스템 평가 도구 기술 개요입니다.

Windows 시스템 평가 도구를 개발하려면 다음 헤더가 필요합니다.

- winsatcominterfacei.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 시스템 평가 도구

열거형

WINSAT_ASSESSMENT_STATE 평가의 가능한 상태를 정의합니다.
WINSAT_ASSESSMENT_TYPE 평가의 가능한 하위 구성 요소 정의
WINSAT_BITMAP_SIZE WinSAT 점수를 나타내는 데 사용할 비트맵의 크기를 정의합니다.

Functions

CancelAssessment 현재 실행 중인 평가를 취소합니다.
get_AllXML 지정된 XPath를 사용하여 공식 XML 평가 문서에서 데이터를 검색합니다. 쿼리는 WinSAT 데이터 저장소의 모든 공식 평가에 대해 실행됩니다.
get_AssessmentDateTime 평가가 실행된 날짜와 시간을 검색합니다.

get_AssessmentState 평가 상태를 검색합니다.
get_Bitmap WinSAT 기본 점수에 대한 비트맵을 검색합니다.
get_Description 하위 구성 요소에 대한 설명을 검색합니다.
get_Info 가장 최근의 공식 평가 결과(예: 기본 점수 및 평가 실행 날짜)에 대한 정보를 제공하는 인터페이스를 검색합니다.
get_RatingStateDesc 평가가 유효한지 여부를 나타내기 위해 UI에서 사용할 수 있는 문자열을 검색합니다.
get_Score 하위 구성 요소의 점수를 검색합니다.
get_SystemRating 컴퓨터의 기본 점수를 검색합니다.
get_Title 하위 구성 요소의 제목을 검색합니다.
get_XML 지정된 XPath를 사용하여 XML 평가 문서에서 데이터를 검색합니다. 쿼리는 WinSAT 데이터 저장소에서 가장 최근의 공식 평가에 대해 실행됩니다.
GetAssessmentInfo 평가의 하위 구성 요소에 대한 요약 정보를 검색합니다.
InitiateAssessment 임시 평가를 시작합니다.
InitiateFormalAssessment 공식 평가를 시작합니다.

WinSATComplete 평가가 성공하거나 실패하거나 취소되면 알림을 받습니다.
WinSATUpdate 평가가 진행 중일 때 알림을 받습니다.

인터페이스

IInitiateWinSATAssessment 평가를 시작합니다.
IProvideWinSATAssessmentInfo 평가의 하위 구성 요소(예: 해당 점수)에 대한 요약 정보를 가져옵니다.
IProvideWinSATResultsInfo 평가 결과(예: 기본 점수 및 평가 실행 날짜)에 대한 정보를 가져옵니다.
IProvideWinSATVisuals WinSAT 평가를 그래픽으로 나타내기 위해 사용자 인터페이스에서 사용할 수 있는 요소를 검색합니다.
IQueryAllWinSATAssessments 모든 공식 WinSAT 평가에 대한 세부 정보를 검색합니다.
IQueryRecentWinSATAssessment 가장 최근의 공식 WinSAT 평가 결과에 대한 세부 정보를 검색합니다.
IWinSATInitiateEvents 평가가 완료되거나 진행 중일 때 알림을 받도록 이 인터페이스를 구현합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

Windows 업데이트 에이전트 API

아티클2023. 03. 16.

Windows 업데이트 에이전트 API 기술에 대한 개요입니다.

Windows 업데이트 에이전트 API를 개발하려면 다음 헤더가 필요합니다.

- wuapi.h
- wuapicommon.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 업데이트 에이전트 API
- WUA(Windows 업데이트 에이전트) API 참조
- Windows 업데이트 에이전트 API 사용

열거형

AddServiceFlag IUpdateServiceManager2 인터페이스가 서비스 등록 요청을 처리할 수 있는 가능한 방법을 정의합니다.
AutoDownloadMode 자동 업데이트 자동으로 업데이트를 다운로드할지 여부를 결정하는 데 사용되는 논리 유형을 정의합니다.
AutomaticUpdatesNotificationLevel 관리자 권한 사용자에게 자동 업데이트 이벤트에 대한 알림을 받을 수 있는 방법을 정의합니다.
AutomaticUpdatesPermissionType IAutomaticUpdatesSettings 인터페이스의 NotificationLevel 속성 또는 IAutomaticUpdatesSettings2 인터페이스의 IncludeRecommendedUpdates 속성을 설정할 수 있는 방법을 정의합니다.
AutomaticUpdatesScheduledInstallationDay 자동 업데이트 업데이트를 설치하거나 제거하는 요일을 정의합니다.

AutomaticUpdatesUserType 사용자 유형을 정의합니다.
AutoSelectionMode 사용자가 Windows 업데이트 사용자 인터페이스에서 사용 가능한 업데이트를 볼 때 특정 업데이트를 자동으로 선택할지 여부를 결정하는 데 사용되는 논리 유형을 정의합니다.
DeploymentAction 업데이트가 명시적으로 배포되는 작업을 정의합니다.
DownloadPhase IDownloadProgress 인터페이스의 CurrentUpdateDownloadPhase 속성에서 반환되는 현재 업데이트의 다운로드 진행률을 정의합니다.
DownloadPriority 다운로드 작업의 가능한 우선 순위를 정의합니다.
InstallationImpact 업데이트를 설치하거나 제거하여 발생할 수 있는 가능한 영향 수준을 정의합니다.
InstallationRebootBehavior 업데이트에 대한 가능한 다시 시작 동작을 정의합니다.
OperationResultCode 업데이트에 대한 다운로드, 설치, 제거 또는 확인 작업의 가능한 결과를 정의합니다.
SearchScope search:_per-machine 업데이트, 사용자별 업데이트 또는 둘 다에서 반환해야 하는 다양한 업데이트를 정의합니다.
UpdateExceptionContext IUpdateException 개체를 제공할 수 있는 컨텍스트를 정의합니다.
UpdateLockdownOption WUA(Windows 업데이트 Agent) 개체가 Windows 업데이트 액세스할 수 있는 기능을 정의합니다.

UpdateOperation 업데이트에서 시도할 수 있는 작업을 정의합니다.
UpdateServiceOption 검사 패키지 서비스에 대한 서비스 등록이 제거되는 방식에 영향을 주는 옵션을 정의합니다.
UpdateServiceRegistrationState 업데이트 서비스에 사용할 수 있는 상태를 정의합니다.
UpdateType 드라이버 또는 소프트웨어 업데이트와 같은 업데이트 유형을 정의합니다.

함수

AcceptEula Windows 업데이트 연결된 Microsoft 소프트웨어 사용 조건에 동의합니다.
추가 컬렉션에 항목을 추가합니다. (IStringCollection.Add)
추가 컬렉션에 항목을 추가합니다. (IUpdateCollection.Add)
AddScanPackageService 검사 패키지를 WUA(Windows 업데이트 Agent)에 서비스로 등록한 다음, IUpdateService 인터페이스를 반환합니다.
AddService WUA(Windows 업데이트 Agent)에 서비스를 등록합니다.
AddService2 권한 부여 캐비닛 파일(.cab)을 요구하지 않고 WUA(Windows 업데이트 에이전트)에 서비스를 등록합니다. 또한 이 메서드는 IUpdateServiceRegistration 인터페이스에 대한 포인터를 반환합니다.

BeginDownload 업데이트와 연결된 콘텐츠 파일의 비동기 다운로드를 시작합니다.
BeginInstall 업데이트의 비동기 설치를 시작합니다.
BeginSearch 업데이트에 대한 비동기 검색 실행을 시작합니다. 검색은 현재 구성된 검색 옵션을 사용합니다.
BeginUninstall 업데이트의 비동기 제거를 시작합니다.
CheckPermission 특정 사용자 또는 사용자 유형에 선택한 작업을 수행할 수 있는 권한이 있는지 여부를 결정합니다.
정리 비동기 작업이 완료될 때까지 기다렸다가 모든 콜백을 해제합니다.
정리 비동기 작업이 완료될 때까지 기다린 다음 모든 콜백을 해제합니다.
정리 비동기 작업이 완료될 때까지 기다린 다음 모든 콜백을 해제합니다.
지우기 컬렉션에서 모든 요소를 제거합니다. (IStringCollection.Clear)
지우기 컬렉션에서 모든 요소를 제거합니다. (IUpdateCollection.Clear)
커밋 이전에 스테이징되었거나 설치된 업데이트를 완료합니다.
복사 컬렉션의 전체 읽기/쓰기 복사본을 만듭니다.

복사 컬렉션의 단순 읽기/쓰기 복사본을 만듭니다.
CopyFromCache 업데이트 내용을 지정된 경로에 복사합니다.
CopyToCache 지정된 원본 위치에서 WUA(내부 Windows 업데이트 에이전트) 다운로드 캐시로 업데이트할 파일을 복사합니다.
CopyToCache 외부 업데이트 이진 파일을 업데이트에 복사합니다.
CreateUpdateDownloader 이 세션에 대한 IUpdateDownloader 인터페이스를 반환합니다.
CreateUpdateInstaller 이 세션에 대한 IUpdateInstaller 인터페이스를 반환합니다.
CreateUpdateSearcher 이 세션에 대한 IUpdateSearcher 인터페이스를 반환합니다.
CreateUpdateServiceManager 세션에 대한 IUpdateServiceManager2 인터페이스에 대한 포인터를 반환합니다.
DetectNow 자동 업데이트 사용하도록 설정된 경우 자동 업데이트 검색 작업을 시작합니다. 업데이트가 검색되면 설치 동작은 IAutomaticUpdatesSettings 인터페이스의 NotificationLevel 속성에 의해 결정됩니다.
다운로드 업데이트와 연결된 콘텐츠 파일의 동기 다운로드를 시작합니다.
EnableService 자동 업데이트 필요한 모든 구성 요소를 사용하도록 설정합니다.

EndDownload 비동기 다운로드를 완료합니다.
EndInstall 업데이트의 비동기 설치를 완료합니다.
EndSearch 업데이트에 대한 비동기 검색을 완료합니다.
EndUninstall 업데이트의 비동기 제거를 완료합니다.
EscapeString 문자열을 검색 조건 문자열에서 리터럴 값으로 사용할 수 있는 문자열로 변환합니다.
get__NewEnum 컬렉션을 열거하는 데 사용할 수 있는 IEnumVARIANT 인터페이스를 가져옵니다. (ICategoryCollection.get__NewEnum)
get__NewEnum 컬렉션을 열거하는 데 사용할 수 있는 IEnumVARIANT 인터페이스를 가져옵니다. (IStringCollection.get__NewEnum)
get__NewEnum 컬렉션을 열거하는 데 사용할 수 있는 IEnumVARIANT 인터페이스를 가져옵니다. (IUpdateCollection.get__NewEnum)
get__NewEnum 컬렉션을 열거하는 데 사용되는 IEnumVARIANT 인터페이스를 가져옵니다. (IUpdateDownloadContentCollection.get__NewEnum)
get__NewEnum 컬렉션을 열거하는 데 사용할 수 있는 IEnumVARIANT 인터페이스를 가져옵니다. (IUpdateExceptionCollection.get__NewEnum)
get__NewEnum 컬렉션을 열거하는 데 사용할 수 있는 IEnumVARIANT 인터페이스를 가져옵니다. (IUpdateHistoryEntryCollection.get__NewEnum)

get__NewEnum 컬렉션을 열거하는 데 사용할 수 있는 IEnumVARIANT 인터페이스를 가져옵니다. (IUpdateServiceCollection.get__NewEnum)
get__NewEnum 컬렉션을 열거하는 데 사용되는 IEnumVARIANT 인터페이스를 가져옵니다. (IWindowsDriverUpdateEntryCollection.get__NewEnum)
get_Address 프록시 서버의 주소 및 10진수 포트 번호를 가져오고 설정합니다. (Get)
get_AllowSourcePrompts 업데이트를 설치할 때 사용자에게 원본 프롬프트를 표시할지 여부를 나타내는 부울 값을 가져오고 설정합니다. (Get)
get_AltText 이미지의 대체 텍스트를 가져옵니다.
get_AsyncState IUpdateDownloader.BeginDownload 메서드에 전달되는 호출자별 상태 개체를 가져옵니다.
get_AsyncState IUpdateInstaller.BeginInstall 메서드 또는 IUpdateInstaller.BeginUninstall 메서드에 전달되는 호출자별 상태 개체를 가져옵니다.
get_AsyncState IUpdateSearch.BeginSearch 메서드에 전달되는 호출자별 상태 개체를 가져옵니다.
get_AttemptCloseAppsIfNecessary 업데이트 설치 관리자가 즉시 업데이트 설치를 차단하여 애플리케이션을 닫을지 여부를 나타내는 값을 가져옵니다.
get_AutoDetect IWebProxy가 프록시 설정을 자동으로 검색하는지 여부를 나타내는 부울 값을 가져오고 설정합니다. (Get)
get_AutoDownload 업데이트의 자동 다운로드 모드를 나타내는 값을 가져옵니다.

get_AutoDownload 업데이트의 자동 다운로드 모드를 나타내는 AutoDownloadMode 값을 가져옵니다.
get_AutoSelection Windows 업데이트 제어판 업데이트의 자동 선택 모드를 나타내는 값을 가져옵니다.
get_AutoSelection Windows 업데이트 제어판 업데이트의 자동 선택 모드를 나타내는 AutoSelectionMode 값을 가져옵니다.
get_AutoSelectOnWebSites 업데이트가 Windows 업데이트 자동으로 선택되도록 플래그가 지정되는지 여부를 나타내는 부울 값을 가져옵니다.
get_BrowseOnly 사용 가능한 업데이트를 검색하여 업데이트를 검색할 수 있는지 여부를 나타내는 부울 값을 가져옵니다. (IUpdate3.get_BrowseOnly)
get_BrowseOnly 사용 가능한 업데이트를 검색하여 업데이트를 검색할 수 있는지 여부를 나타내는 부울 값을 가져옵니다. (IWindowsDriverUpdate3.get_BrowseOnly)
get_BundledUpdates 업데이트에 대한 번들 업데이트의 정렬된 목록에 대한 정보가 포함된 인터페이스를 가져옵니다.
get_BypassList 프록시 서버를 사용하지 않는 주소 컬렉션을 가져오고 설정합니다. (Get)
get_BypassProxyOnLocal 로컬 주소가 프록시 서버를 우회하는지 여부를 나타내는 부울 값을 가져오고 설정합니다. (Get)
get_CanAutomaticallyUpgradeService BeginSearch 및 Search 메서드에 대한 이후 호출로 인해 WUA(Windows 업데이트 Agent)로 자동 업그레이드되는지 여부를 나타내는 부울 값을 가져오고 설정합니다. (Get)
get_CanRegisterWithAU 서비스가 자동 업데이트 등록할 수 있는지 여부를 나타내는 부울 값을 가져옵니다.

get_CanRequestUserInput 부울 값 thast를 가져오면 업데이트 설치 또는 제거에서 사용자 입력을 확인할 수 있는지 여부를 나타냅니다.
get_CanRequireSource 업데이트의 원본 미디어가 설치 또는 제거에 필요한지 여부를 나타내는 부울 값을 가져옵니다.
get_Categories 업데이트가 속한 범주의 컬렉션을 포함하는 인터페이스를 가져옵니다.
get_Categories 업데이트가 속한 업데이트 범주의 컬렉션을 가져옵니다.
get_CategoryID 범주의 식별자를 가져옵니다.
get_Children 이 범주의 자식 범주를 포함하는 인터페이스 컬렉션을 가져옵니다.
get_ClientApplicationID 현재 클라이언트 애플리케이션을 가져오고 설정합니다. (IUpdateDownloader.get_ClientApplicationID)
get_ClientApplicationID 업데이트를 처리한 클라이언트 애플리케이션의 식별자를 가져옵니다.
get_ClientApplicationID 현재 클라이언트 애플리케이션을 가져오고 설정합니다. (IUpdateInstaller.get_ClientApplicationID)
get_ClientApplicationID 현재 클라이언트 애플리케이션을 식별합니다. (Get)
get_ClientApplicationID 현재 클라이언트 애플리케이션의 식별자를 가져오고 설정합니다. (Get)
get_ClientApplicationID 현재 클라이언트 애플리케이션을 가져오고 설정합니다. (IUpdateSession.get_ClientApplicationID)

get_ContentValidationCert 서비스의 내용에 서명하는 데 사용되는 인증서의 SHA-1 해시를 가져옵니다.
get_Context 검색 결과의 컨텍스트를 가져옵니다.
get_Count 컬렉션의 요소 수를 가져옵니다. (ICategoryCollection.get_Count)
get_Count 컬렉션의 요소 수를 가져옵니다. (IStringCollection.get_Count)
get_Count 컬렉션의 요소 수를 가져옵니다. (IUpdateCollection.get_Count)
get_Count 컬렉션의 요소 수를 가져옵니다. (IUpdateDownloadContentCollection.get_Count)
get_Count 컬렉션의 요소 수를 가져옵니다. (IUpdateExceptionCollection.get_Count)
get_Count 컬렉션의 요소 수를 가져옵니다. (IUpdateHistoryEntryCollection.get_Count)
get_Count 컬렉션의 요소 수를 가져옵니다. (IUpdateServiceCollection.get_Count)
get_Count 컬렉션에 포함된 요소 수를 가져옵니다.
get_CurrentUpdateBytesDownloaded 다운로드 중인 업데이트 파일 또는 콘텐츠 파일에 대해 전송된 데이터의 양을 바이트 단위로 지정하는 문자열을 가져옵니다.
get_CurrentUpdateBytesToDownload 다운로드 중인 업데이트의 콘텐츠 파일 또는 파일에 대해 전송해야 하는 데이터의 양을 바이트 단위로 예측하는 문자열을 가져옵니다.

get_CurrentUpdateDownloadPhase 현재 진행 중인 다운로드의 단계를 지정하는 DownloadPhase 열거형 값을 가져옵니다.
get_CurrentUpdatePercentComplete 다운로드된 현재 업데이트의 예상 백분율을 가져옵니다.
get_CurrentUpdatePercentComplete 현재 업데이트의 설치 또는 제거 프로세스가 진행된 정도를 백분율로 가져옵니다.
get_CveIDs 업데이트와 연결된 CVE(일반적인 취약성 및 노출) ID의 컬렉션을 가져옵니다.
get_CveIDs 업데이트와 연결된 CVE(Common Vulnerabilities and Exposures) 식별자 컬렉션을 포함합니다.
get_Date 업데이트가 적용된 날짜 및 시간을 가져옵니다.
get_Deadline 업데이트를 설치해야 하는 날짜를 가져옵니다.
get_DeltaCompressedContentAvailable 업데이트에 대해 서버에서 델타 압축 콘텐츠를 사용할 수 있는지 여부를 나타내는 부울 값을 가져옵니다.
get_DeltaCompressedContentPreferred 델타 압축 콘텐츠를 사용할 수 있는 경우 업데이트를 다운로드 및 설치 또는 제거하는 동안 델타 압축 콘텐츠를 선호할지 여부를 나타내는 부울 값을 가져옵니다.
get_DeploymentAction 업데이트가 배포되는 작업을 가져옵니다.
get_Description 범주에 대한 설명을 가져옵니다.
get_Description 업데이트에 대한 지역화된 설명을 가져옵니다.

get_Description 업데이트에 대한 설명을 가져옵니다.
get_DeviceProblemNumber Windows 드라이버 업데이트에 일치하는 디바이스의 문제 번호를 가져옵니다. (IWindowsDriverUpdate.get_DeviceProblemNumber)
get_DeviceProblemNumber Windows 드라이버 업데이트에 일치하는 디바이스의 문제 번호를 가져옵니다. (IWindowsDriverUpdateEntry.get_DeviceProblemNumber)
get_DeviceStatus Windows 드라이버 업데이트에 일치하는 디바이스의 상태를 가져옵니다. (IWindowsDriverUpdate.get_DeviceStatus)
get_DeviceStatus Windows 드라이버 업데이트에 일치하는 디바이스의 상태를 가져옵니다. (IWindowsDriverUpdateEntry.get_DeviceStatus)
get_DownloadContents 업데이트의 다운로드 내용에 대한 파일 정보를 가져옵니다.
get_DownloadPriority 업데이트의 제안된 다운로드 우선 순위를 가져옵니다.
get_DownloadUrl 업데이트를 호스트하는 서버에서 다운로드 콘텐츠의 위치를 가져옵니다.
get_DriverClass Windows 드라이버 업데이트의 클래스를 가져옵니다.
get_DriverClass Windows 드라이버 업데이트의 클래스를 검색합니다.
get_DriverHardwareID Windows 드라이버 업데이트가 설치 가능하도록 일치해야 하는 하드웨어 ID 또는 호환 ID를 가져옵니다.

get_DriverHardwareID Windows 드라이버 업데이트가 설치 가능하도록 일치해야 하는 하드웨어 또는 호환 식별자를 가져옵니다.
get_DriverManufacturer Windows 드라이버 업데이트 제조업체의 언어 고정 이름을 가져옵니다. (IWindowsDriverUpdate.get_DriverManufacturer)
get_DriverManufacturer Windows 드라이버 업데이트 제조업체의 언어 고정 이름을 가져옵니다. (IWindowsDriverUpdateEntry.get_DriverManufacturer)
get_DriverModel Windows 드라이버 업데이트가 의도된 디바이스의 언어 고정 모델 이름을 가져옵니다. (IWindowsDriverUpdate.get_DriverModel)
get_DriverModel Windows 드라이버 업데이트가 의도된 디바이스의 언어 고정 모델 이름을 가져옵니다. (IWindowsDriverUpdateEntry.get_DriverModel)
get_DriverProvider Windows 드라이버 업데이트 공급자의 언어 고정 이름을 가져옵니다. (IWindowsDriverUpdate.get_DriverProvider)
get_DriverProvider Windows 드라이버 업데이트 공급자의 언어 고정 이름을 가져옵니다. (IWindowsDriverUpdateEntry.get_DriverProvider)
get_DriverVerDate Windows 드라이버 업데이트의 드라이버 버전 날짜를 가져옵니다. (IWindowsDriverUpdate.get_DriverVerDate)
get_DriverVerDate Windows 드라이버 업데이트의 드라이버 버전 날짜를 가져옵니다. (IWindowsDriverUpdateEntry.get_DriverVerDate)
get_EulaAccepted 업데이트와 연결된 Microsoft 소프트웨어 사용 조건이 컴퓨터에 허용되는지 여부를 나타내는 부울 값을 가져옵니다.

get_EulaText 업데이트와 연결된 Microsoft 소프트웨어 사용 조건의 전체 지역화된 텍스트를 가져옵니다.
get_ExpirationDate 권한 부여 캐비닛 파일이 만료되는 날짜를 가져옵니다.
get_FeaturedUpdatesEnabled 지원 안 됨 (IAutomaticUpdatesSettings3.get_FeaturedUpdatesEnabled)
get_ForceQuiet Windows Installer가 사용자 상호 작용 없이 업데이트를 강제로 설치해야 하는지 여부를 나타내는 부울 값을 가져오고 설정합니다. (Get)
get_HandlerID 업데이트의 설치 처리기를 가져옵니다.
get_Height 이미지의 높이를 픽셀 단위로 가져옵니다.
get_HResult 다운로드하는 동안 예외 코드 번호가 발생하는 경우 예외 코드 번호를 가져옵니다.
get_HResult 설치 중에 발생하는 예외의 HRESULT(있는 경우)를 가져옵니다.
get_HResult 업데이트 작업 중에 발생하는 예외 HRESULT 값(있는 경우)을 가져옵니다.
get_HResult 검색 결과에 대한 Windows 기반 HRESULT 코드를 가져옵니다.
get_HResult 업데이트 작업에서 반환되는 HRESULT 값을 가져옵니다.
get_HResult 업데이트 작업 중에 발생하는 HRESULT 예외 값을 가져옵니다.

get_Identity 업데이트의 고유 식별자를 포함하는 인터페이스를 가져옵니다.
get_IgnoreDownloadPriority 다운로드 우선 순위를 무시할지 여부를 나타내는 부울 값을 가져오고 설정합니다. (Get)
get_Image 범주와 연결된 이미지에 대한 정보가 들어 있는 인터페이스를 가져옵니다.
get_Image 업데이트와 연결된 이미지에 대한 정보가 포함된 인터페이스를 가져옵니다.
get_Impact 업데이트의 설치 또는 제거가 컴퓨터에 미치는 영향을 나타내는 InstallationImpact 열거형을 가져옵니다.
get_IncludePotentiallySupersededUpdates 검색 결과에 검색 결과의 다른 업데이트로 대체되는 업데이트가 포함되는지 여부를 나타내는 부울 값을 가져오고 설정합니다. (Get)
get_IncludeRecommendedUpdates 업데이트 검색 및 업데이트 설치가 수행될 때 선택 사항 또는 권장 업데이트를 포함할지 여부를 나타내는 부울 값을 가져오고 설정합니다. (Get)
get_InstallationBehavior 업데이트의 설치 옵션이 포함된 인터페이스를 가져옵니다.
get_IsBeta 업데이트가 베타 릴리스인지 여부를 나타내는 부울 값을 가져옵니다.
get_IsBusy 특정 시간에 컴퓨터에서 설치 또는 제거가 진행 중인지 여부를 나타내는 부울 값을 가져옵니다.
get_IsCompleted IUpdateDownloader.BeginDownload에 대한 호출이 완전히 처리되었는지 여부를 나타내는 설정을 가져옵니다.

get_IsCompleted IUpdateInstaller.BeginInstall 또는 IUpdateInstaller.BeginUninstall 메서드에 대한 호출이 완전히 처리되었는지 여부를 나타내는 값을 가져옵니다.
get_IsCompleted IUpdateSearch.BeginSearch 메서드에 대한 호출이 완전히 처리되었는지 여부를 나타내는 부울 값을 가져옵니다.
get_IsDefaultAUService 서비스가 자동 업데이트 등록되었는지 여부와 서비스가 현재 자동 업데이트 기본 서비스로 사용되는지 여부를 나타내는 부울 값을 가져옵니다.
get_IsDeltaCompressedContent 업데이트가 이진 업데이트인지 아니면 전체 파일 업데이트인지를 나타내는 부울 값을 가져옵니다.
get_IsDownloaded 모든 업데이트 콘텐츠가 컴퓨터에 캐시되는지 여부를 나타내는 부울 값을 가져옵니다.
get_IsForced WUA(Windows 업데이트 Agent)가 이미 설치되어 있거나 설치할 수 없는 업데이트를 강제로 다운로드하는지 여부를 나타내는 부울 값을 가져오고 설정합니다. (Get)
get_IsForced 업데이트를 강제로 설치할지 또는 제거할지를 나타내는 부울 값을 가져오거나 설정합니다. (Get)
get_IsHidden 사용자가 업데이트를 숨길지 여부를 나타내는 부울 값을 가져옵니다. (Get)
get_IsInstalled 검색이 수행될 때 업데이트가 컴퓨터에 설치되어 있는지 여부를 나타내는 부울 값을 가져옵니다.
get_IsManaged 서비스가 관리되는 서비스인지 여부를 나타내는 부울 값을 가져옵니다.
get_IsMandatory 업데이트 설치가 필수인지 여부를 나타내는 부울 값을 가져옵니다.

get_IsPendingRegistrationWithAU 서비스를 추가할 때 자동 업데이트 등록할지 여부를 나타내는 부울 값을 가져옵니다.
get_IsPresent 컴퓨터에 업데이트가 있는지 여부를 나타내는 부울 값을 가져옵니다.
get_IsPresent 컴퓨터에 업데이트가 설치되어 있는지 여부를 나타내는 부울 값을 가져옵니다.
get_IsRegisteredWithAU 서비스가 자동 업데이트 등록되었는지 여부를 나타내는 부울 값을 가져옵니다.
get_IsScanPackageService 서비스가 검사 패키지를 기반으로 하는지 여부를 나타내는 부울 값을 가져옵니다.
get_IssueDate 권한 부여 캐비닛 파일이 발급된 날짜를 가져옵니다.
get_IsUninstallable 사용자가 컴퓨터에서 업데이트를 제거할 수 있는지 여부를 나타내는 부울 값을 가져옵니다.
get_Item 컬렉션에서 ICategory 인터페이스를 가져옵니다.
get_Item 컬렉션의 문자열을 가져오거나 설정합니다. (Get)
get_Item 컬렉션의 IUpdate 인터페이스를 가져오거나 설정합니다. (Get)
get_Item IUpdateDownloadContentCollection 인터페이스에서 업데이트에 대한 다운로드 콘텐츠를 가져옵니다.
get_Item 컬렉션에서 IUpdateException 인터페이스를 가져옵니다.

get_Item 컬렉션에서 IUpdateHistoryEntry 인터페이스를 가져옵니다.
get_Item 컬렉션에서 IUpdateService 인터페이스를 가져오고 설정합니다.
get_Item 컬렉션에서 IWindowsDriverUpdateEntry 인터페이스를 가져옵니다.
get_KBArticleIDs 업데이트와 연결된 Microsoft 기술 자료 문서 ID의 컬렉션을 가져옵니다.
get_Languages 업데이트에서 지원하는 언어를 포함하는 인터페이스를 가져옵니다.
get_LastDeploymentChangeTime 업데이트를 배포하는 서버에서 마지막으로 게시된 업데이트 날짜(UTC(협정 세계시) 날짜 및 시간을 가져옵니다.
get_LastInstallationSuccessDate 일부 오류가 발생하더라도 자동 업데이트 업데이트를 성공적으로 설치한 마지막 시간 및 UTC(협정 세계시) 날짜를 가져옵니다.
get_LastSearchSuccessDate AutomaticUpdates가 업데이트를 성공적으로 검색한 마지막 시간 및 UTC(협정 세계시) 날짜를 가져옵니다.
get_MaxDownloadSize 업데이트의 최대 다운로드 크기를 가져옵니다.
get_Message 검색 결과를 설명하는 메시지를 가져옵니다.
get_MinDownloadSize 업데이트의 최소 다운로드 크기를 가져옵니다.

get_MoreInfoUrls 업데이트에 대한 자세한 내용에 대한 하이퍼링크를 지정하는 언어별 문자열 컬렉션을 가져옵니다.
get_MsrcSeverity 업데이트의 Microsoft 보안 대응 센터 심각도 등급을 가져옵니다.
get_Name 범주의 지역화된 이름을 가져옵니다.
get_Name 서비스의 이름을 가져옵니다.
get_NonAdministratorsElevated 관리자가 아닌 사용자가 관리자의 승인 없이 일부 업데이트 관련 작업을 수행할 수 있는지 여부를 나타내는 부울 값을 가져오고 설정합니다. (Get)
get_NotificationLevel 자동 업데이트 이벤트에 대해 사용자에게 알림을 받는 방법을 가져오고 설정합니다. (Get)
get_OemHardwareSupportLink OEM 하드웨어에 대한 기술 지원 정보에 대한 하이퍼링크를 가져옵니다.
get_OffersWindowsUpdates 부울 값을 가져오면 현재 서비스가 Windows 업데이트 업데이트를 제공하는지 여부를 나타냅니다.
get_Online UpdateSearcher가 온라인 상태가 되어 업데이트를 검색하는지 여부를 나타내는 부울 값을 가져오고 설정합니다. (Get)
get_Operation 업데이트 작업을 지정하는 UpdateOperation 값을 가져옵니다.
get_Order 형제 범주 중에서 이 범주의 권장 표시 순서를 가져옵니다.
get_Parent 이 범주의 부모 범주를 설명하는 인터페이스를 가져옵니다.

get_ParentHwnd 대화 상자를 포함할 수 있는 부모 창에 대한 핸들을 가져오고 설정합니다. (Get)
get_ParentWindow 대화 상자를 포함할 수 있는 부모 창을 나타내는 인터페이스를 가져오고 설정합니다. (Get)
get_PercentComplete 다운로드된 모든 업데이트의 예상 백분율을 가져옵니다.
get_PercentComplete 전체 설치 또는 제거 프로세스가 얼마나 진행되었는지 백분율로 가져옵니다.
get_PerUser 사용자별 업데이트인지 여부를 나타내는 부울 값을 가져옵니다.
get_PerUser 업데이트가 사용자별 업데이트인지 여부를 나타내는 부울 값을 가져옵니다.
get_Priority 다운로드의 우선 순위 수준을 가져오고 설정합니다. (Get)
get_Product 제품의 언어 고정 이름을 가져옵니다.
get_Progress 콜백을 만들 때 비동기 다운로드의 진행률을 포함하는 인터페이스를 가져옵니다.
get_Progress 콜백이 수행되었을 때 비동기 설치 또는 제거의 진행률을 포함하는 인터페이스를 가져옵니다.
get_ReadOnly 자동 업데이트 설정이 읽기 전용인지 여부를 나타내는 부울 값을 가져옵니다.
get_ReadOnly 컬렉션이 읽기 전용인지 여부를 나타내는 부울 값을 가져옵니다.

get_ReadOnly 업데이트 컬렉션이 읽기 전용인지 여부를 나타내는 부울 값을 가져옵니다.
get_ReadOnly 세션 개체가 읽기 전용인지 여부를 나타내는 부울 값을 가져옵니다.
get_ReadOnly WebProxy 개체가 읽기 전용인지 여부를 나타내는 부울 값을 가져옵니다.
get_RebootBehavior 업데이트를 설치하거나 제거할 때 발생하는 다시 시작 동작을 지정하는 InstallationRebootBehavior 열거형을 가져옵니다.
get_RebootRequired 업데이트 설치 또는 제거를 완료하기 위해 컴퓨터를 다시 시작해야 하는지 여부를 나타내는 부울 값을 가져옵니다.
get_RebootRequired 하나 이상의 업데이트를 설치하거나 제거하는 데 시스템 다시 시작이 필요한지 여부를 나타내는 부울 값을 가져옵니다.
get_RebootRequired 설치를 완료하기 위해 컴퓨터에서 시스템 다시 시작이 필요한지 또는 업데이트 제거를 완료해야 하는지를 나타내는 부울 값을 가져옵니다.
get_RebootRequired 업데이트 설치를 완료하기 위해 컴퓨터에서 시스템 다시 시작이 필요한지 여부를 나타내는 부울 값을 가져옵니다.
get_RebootRequired 업데이트를 설치하거나 제거한 후 컴퓨터를 다시 시작해야 하는지 여부를 나타내는 부울 값을 가져옵니다.
get_RebootRequiredBeforeInstallation 업데이트를 설치하거나 제거하기 전에 시스템 다시 시작이 필요한지 여부를 나타내는 부울 값을 가져옵니다.

get_RecommendedCpuSpeed 업데이트를 설치하는 데 사용되는 권장 CPU 속도를 메가헤르츠(MHz)로 가져옵니다.
get_RecommendedHardDiskSpace 업데이트를 설치하기 전에 하드 디스크에서 사용할 수 있는 권장 여유 공간을 가져옵니다. 사용 가능한 공간은 MB(메가바이트)로 지정됩니다.
get_RecommendedMemory 업데이트를 설치하기 전에 컴퓨터에서 사용할 수 있는 권장 실제 메모리 크기를 가져옵니다. 실제 메모리 크기는 MB(메가바이트)로 지정됩니다.
get_RedirectUrls 리다이렉터 캐비닛 파일의 URL을 포함합니다.
get_RegistrationState 서비스 등록의 현재 상태를 나타내는 UpdateServiceRegistrationState 값을 가져옵니다.
get_ReleaseNotes 업데이트에 대한 지역화된 릴리스 정보를 가져옵니다.
get_Required 그룹 정책 자동 업데이트 서비스가 필요한지 여부를 나타내는 부울 값을 가져옵니다.
get_RequiresNetworkConnectivity 업데이트를 설치하거나 제거하려면 네트워크 연결이 필요한지 여부를 나타내는 부울 값을 가져옵니다.
get_ResultCode 다운로드 결과를 지정하는 OperationResultCodeenumeration을 가져옵니다.
get_ResultCode 업데이트에 대한 작업의 결과를 지정하는 OperationResultCode 값을 가져옵니다. (IInstallationResult.get_ResultCode)
get_ResultCode 검색 결과를 지정하는 OperationResultCode 열거형을 가져옵니다.

get_ResultCode 업데이트 작업의 결과를 지정하는 OperationResultCode 열거형 값을 가져옵니다.
get_ResultCode 업데이트에 대한 작업의 결과를 지정하는 OperationResultCode 값을 가져옵니다. (IUpdateHistoryEntry.get_ResultCode)
get_ResultCode 업데이트에 대한 작업의 결과를 지정하는 OperationResultCode 값을 가져옵니다. (IUpdateInstallationResult.get_ResultCode)
get_Results IAutomaticUpdatesResults 인터페이스에 대한 포인터를 반환합니다.
get_RevisionNumber 업데이트의 수정 번호를 가져옵니다.
get_RootCategories 컴퓨터에서 현재 사용할 수 있는 루트 범주의 인터페이스 컬렉션을 가져옵니다.
get_ScheduledInstallationDay 자동 업데이트 업데이트를 설치하거나 제거하는 요일을 가져오고 설정합니다. (Get)
get_ScheduledInstallationTime 자동 업데이트 업데이트를 설치하거나 제거하는 시간을 가져오고 설정합니다. (Get)
get_SecurityBulletinIDs 업데이트와 연결된 보안 공지 ID를 포함하는 문자열 값의 컬렉션을 가져옵니다.
get_ServerSelection 업데이트를 제공한 서버를 나타내는 ServerSelection 값을 가져옵니다.
get_ServerSelection 업데이트를 검색할 서버를 나타내는 ServerSelection 값을 가져오고 설정합니다. (Get)
get_Service IUpdateService2 인터페이스에 대한 포인터를 가져옵니다. 이 속성은 기본 속성입니다.

get_ServiceEnabled 자동 업데이트 필요한 모든 구성 요소를 사용할 수 있는지 여부를 나타내는 부울 값을 가져옵니다.
get_ServiceID Windows 업데이트가 아닌 업데이트 서비스의 서비스 식별자를 가져옵니다.
get_ServiceID 검색할 사이트가 Windows 업데이트 사이트가 아닌 경우 검색할 사이트를 가져오고 설정합니다. (Get)
get_ServiceID 서비스의 식별자를 검색하거나 설정합니다.
get_Services WUA에 등록된 서비스의 IUpdateServiceCollection을 가져옵니다.
get_ServiceUrl 서비스의 URL을 검색합니다.
get_Settings 자동 업데이트 대한 구성 설정을 가져옵니다.
get_SetupPrefix 설치 파일의 접두사 식별
get_Source 이미지의 원본 위치를 가져옵니다.
get_SupersededUpdateIDs 업데이트 식별자의 컬렉션을 가져옵니다. 이 식별자 컬렉션은 업데이트로 대체되는 업데이트를 지정합니다.
get_SupportUrl 업데이트에 대한 언어별 지원 정보에 대한 하이퍼링크를 가져옵니다.
get_SupportUrl 업데이트에 대한 언어별 지원 정보에 대한 하이퍼링크를 가져옵니다.

get_Title 업데이트의 지역화된 제목을 가져옵니다.
get_Title 업데이트의 제목을 가져옵니다.
get_TotalBytesDownloaded 다운로드된 데이터의 총 양을 바이트 단위로 지정하는 문자열을 가져옵니다.
get_TotalBytesToDownload 다운로드할 총 데이터 양(바이트)의 추정치를 나타내는 문자열을 가져옵니다.
get_Type 범주의 형식을 가져옵니다.
get_Type 업데이트의 형식을 가져옵니다.
get_UninstallationBehavior 업데이트에 대한 제거 옵션이 포함된 인터페이스를 가져옵니다.
get_UninstallationNotes 업데이트에 대한 제거 정보를 가져옵니다.
get_UninstallationNotes 업데이트의 제거 정보를 가져옵니다.
get_UninstallationSteps 업데이트에 대한 제거 단계가 포함된 인터페이스를 가져옵니다.
get_UninstallationSteps 업데이트에 대한 제거 단계를 포함하는 IStringCollection 인터페이스를 가져옵니다.
get_UnmappedResultCode 업데이트 작업에서 반환되는 매핑되지 않은 결과 코드를 가져옵니다.

get_UpdateID 업데이트의 수정 버전 독립적 식별자를 가져옵니다.
get_UpdateIdentity 업데이트의 ID를 포함하는 IUpdateIdentity 인터페이스를 가져옵니다.
get_Updates 범주에 즉시 속하는 업데이트 컬렉션이 포함된 인터페이스를 가져옵니다.
get_Updates 다운로드에 지정된 업데이트의 읽기 전용 컬렉션을 포함하는 인터페이스를 가져옵니다.
get_Updates 설치 또는 제거에 지정된 업데이트의 읽기 전용 컬렉션을 포함하는 인터페이스를 가져옵니다.
get_Updates 검색에서 발생하는 업데이트의 인터페이스 컬렉션을 가져옵니다.
get_Updates 다운로드를 위해 지정된 업데이트의 읽기 전용 컬렉션을 포함하는 인터페이스를 가져오고 설정합니다. (Get)
get_Updates 설치 또는 제거에 대해 지정된 업데이트의 읽기 전용 컬렉션을 포함하는 인터페이스를 가져오고 설정합니다. (Get)
get_UserLocale 업데이트 정보를 검색할 기본 로캘을 가져오고 설정합니다. (Get)
get_UserName 인증을 위해 프록시 서버에 제출할 사용자 이름을 가져오고 설정합니다. (Get)
get_Warnings 검색으로 인해 발생하는 경고의 컬렉션을 가져옵니다.
get_WebProxy 서버에 액세스하는 데 사용되는 프록시 설정을 가져오고 설정합니다. (Get)

get_Width 이미지의 너비를 픽셀 단위로 가져옵니다.
get_WindowsDriverUpdateEntries 업데이트에 적용할 수 있는 드라이버 업데이트 항목을 가져옵니다.
GetInfo WUA(Windows 업데이트 Agent)에 대한 버전 정보를 검색합니다.
GetProgress 다운로드의 현재 진행 상황을 설명하는 IDownloadProgress 인터페이스를 반환합니다.
GetProgress 설치 또는 제거의 현재 진행 상황을 설명하는 IInstallationProgress 인터페이스를 반환합니다.
GetTotalHistoryCount 컴퓨터의 업데이트 이벤트 수를 반환합니다.
GetUpdateResult 지정된 업데이트 다운로드 결과를 반환합니다.
GetUpdateResult 지정된 업데이트에 대한 다운로드 정보가 포함된 IUpdateDownloadResult 인터페이스를 반환합니다.
GetUpdateResult 지정된 업데이트의 설치 또는 제거 결과를 반환합니다.
GetUpdateResult 지정된 업데이트에 대한 설치 결과를 포함하는 IUpdateInstallationResult 인터페이스를 반환합니다.
삽입 지정된 위치의 컬렉션에 항목을 삽입합니다. (IStringCollection.Insert)
삽입 지정된 위치의 컬렉션에 항목을 삽입합니다. (IUpdateCollection.Insert)

설치 업데이트의 동기 설치를 시작합니다.
Invoke 호출자에게 다운로드가 완료되었다는 것을 알 수 있습니다.
Invoke IUpdateDownloader.BeginDownload 메서드를 호출하여 시작된 비동기 다운로드 진행률의 변경 알림을 처리합니다.
Invoke IUpdateInstaller.BeginInstall 또는 IUpdateInstaller.BeginUninstall에 대한 호출로 시작되는 비동기 설치 또는 제거 완료 알림을 처리합니다.
Invoke IUpdateInstaller.BeginInstall 메서드 또는 IUpdateInstaller.BeginUninstall 메서드를 호출하여 시작된 비동기 설치 또는 제거 진행률의 변경 내용 알림을 처리합니다.
Invoke IUpdateSearcher.BeginSearch 메서드를 호출하여 시작되는 비동기 검색 완료 알림을 처리합니다.
잠금 이 메서드를 구현하는 개체의 메서드 및 속성에 대한 액세스를 제한합니다.
일시 중지 자동 업데이트를 일시 중지합니다.
PromptForCredentials 사용자에게 프록시 인증에 사용할 암호를 묻는 메시지를 표시합니다.
PromptForCredentialsFromHwnd 부모 창의 hWnd 속성을 사용하여 프록시 인증에 사용할 암호를 묻는 메시지를 사용자에게 표시합니다.
put_Address 프록시 서버의 주소 및 10진수 포트 번호를 가져오고 설정합니다. (Put)

put_AllowSourcePrompts 업데이트를 설치할 때 사용자에게 원본 프롬프트를 표시할지 여부를 나타내는 부울 값을 가져오고 설정합니다. (Put)
put_AttemptCloseAppsIfNecessary 업데이트 설치 관리자가 애플리케이션을 닫을지 여부를 나타내는 값을 설정하여 업데이트의 즉각적인 설치를 차단합니다.
put_AutoDetect IWebProxy가 프록시 설정을 자동으로 검색하는지 여부를 나타내는 부울 값을 가져오고 설정합니다. (Put)
put_BypassList 프록시 서버를 사용하지 않는 주소 컬렉션을 가져오고 설정합니다. (Put)
put_BypassProxyOnLocal 로컬 주소가 프록시 서버를 우회하는지 여부를 나타내는 부울 값을 가져오고 설정합니다. (Put)
put_CanAutomaticallyUpgradeService BeginSearch 및 Search 메서드에 대한 이후 호출로 인해 WUA(Windows 업데이트 Agent)로 자동 업그레이드되는지 여부를 나타내는 부울 값을 가져오고 설정합니다. (Put)
put_ClientApplicationID 현재 클라이언트 애플리케이션을 가져오고 설정합니다. (IUpdateDownloader.put_ClientApplicationID)
put_ClientApplicationID 현재 클라이언트 애플리케이션을 가져오고 설정합니다. (IUpdateInstaller.put_ClientApplicationID)
put_ClientApplicationID 현재 클라이언트 애플리케이션을 식별합니다. (Put)
put_ClientApplicationID 현재 클라이언트 애플리케이션의 식별자를 가져오고 설정합니다. (Put)
put_ClientApplicationID 현재 클라이언트 애플리케이션을 가져오고 설정합니다. (IUpdateSession.put_ClientApplicationID)

put_FeaturedUpdatesEnabled 지원 안 됨 (IAutomaticUpdatesSettings3.put_FeaturedUpdatesEnabled)
put_ForceQuiet Windows Installer가 사용자 상호 작용 없이 업데이트를 강제로 설치해야 하는지 여부를 나타내는 부울 값을 가져오고 설정합니다. (Put)
put_IgnoreDownloadPriority 다운로드 우선 순위를 무시할지 여부를 나타내는 부울 값을 가져오고 설정합니다. (Put)
put_IncludePotentiallySupersededUpdates 검색 결과에 검색 결과의 다른 업데이트로 대체되는 업데이트가 포함되는지 여부를 나타내는 부울 값을 가져오고 설정합니다. (Put)
put_IncludeRecommendedUpdates 업데이트 검색 및 업데이트 설치가 수행될 때 선택 사항 또는 권장 업데이트를 포함할지 여부를 나타내는 부울 값을 가져오고 설정합니다. (Put)
put_IsForced WUA(Windows 업데이트 Agent)가 이미 설치되어 있거나 설치할 수 없는 업데이트를 강제로 다운로드하는지 여부를 나타내는 부울 값을 가져오고 설정합니다. (Put)
put_IsForced 업데이트를 강제로 설치할지 또는 제거할지를 나타내는 부울 값을 가져오거나 설정합니다. (Put)
put_IsHidden 사용자가 업데이트를 숨길지 여부를 나타내는 부울 값을 가져옵니다. (Put)
put_Item 컬렉션에서 문자열을 가져오거나 설정합니다. (Put)
put_Item 컬렉션에서 IUpdate 인터페이스를 가져오거나 설정합니다. (Put)
put_NonAdministratorsElevated 관리자가 아닌 사용자가 관리자 승인 없이 일부 업데이트 관련 작업을 수행할 수 있는지 여부를 나타내는 부울 값을 가져오고 설정합니다. (Put)

put_NotificationLevel 자동 업데이트 이벤트에 대해 사용자에게 알림을 받는 방법을 가져오고 설정합니다. (Put)
put_Online UpdateSearcher가 업데이트를 검색하기 위해 온라인 상태가 되는지 여부를 나타내는 부울 값을 가져오고 설정합니다. (Put)
put_ParentHwnd 대화 상자를 포함할 수 있는 부모 창에 대한 핸들을 가져오고 설정합니다. (Put)
put_ParentWindow 대화 상자를 포함할 수 있는 부모 창을 나타내는 인터페이스를 가져오고 설정합니다. (Put)
put_Priority 다운로드의 우선 순위 수준을 가져오고 설정합니다. (Put)
put_ScheduledInstallationDay 자동 업데이트 업데이트를 설치하거나 제거하는 요일을 가져오고 설정합니다. (Put)
put_ScheduledInstallationTime 자동 업데이트 업데이트를 설치하거나 제거하는 시간을 가져오고 설정합니다. (Put)
put_ServerSelection 업데이트를 검색할 서버를 나타내는 ServerSelection 값을 가져오고 설정합니다. (Put)
put_ServiceID 검색할 사이트가 Windows 업데이트 사이트가 아닌 경우 검색할 사이트를 가져오고 설정합니다. (Put)
put_Updates 다운로드를 위해 지정된 업데이트의 읽기 전용 컬렉션을 포함하는 인터페이스를 가져오고 설정합니다. (Put)
put_Updates 설치 또는 제거에 대해 지정된 업데이트의 읽기 전용 컬렉션을 포함하는 인터페이스를 가져오고 설정합니다. (Put)

put_UserLocale 업데이트 정보를 검색할 기본 로캘을 가져오고 설정합니다. (Put)
put_UserName 인증을 위해 프록시 서버에 제출할 사용자 이름을 가져오고 설정합니다. (Put)
put_WebProxy 서버에 액세스하는 데 사용되는 프록시 설정을 가져오고 설정합니다. (Put)
QueryHistory 컴퓨터에서 업데이트 이벤트의 기록을 동기적으로 쿼리합니다.
QueryHistory 컴퓨터에서 업데이트 이벤트의 기록을 동기적으로 쿼리합니다.
QueryServiceRegistration IUpdateServiceRegistration 인터페이스에 대한 포인터를 반환합니다.
RecordInstallationResult 업데이트 결과를 기록합니다. 결과는 IStringCollection 개체에 의해 지정됩니다.
새로 고침 최신 자동 업데이트 설정을 검색합니다.
RegisterServiceWithAU 자동 업데이트 서비스를 등록합니다.
RemoveAt 컬렉션에서 지정된 인덱스의 항목을 제거합니다. (IStringCollection.RemoveAt)
RemoveAt 컬렉션에서 지정된 인덱스의 항목을 제거합니다. (IUpdateCollection.RemoveAt)
RemoveService WUA(Windows 업데이트 에이전트)에서 서비스 등록을 제거합니다.

RequestAbort 비동기 다운로드를 종료하도록 요청합니다.
RequestAbort 설치 또는 제거를 취소하도록 요청합니다.
RequestAbort 비동기 검색을 취소하도록 요청합니다.
재개 자동 업데이트가 일시 중지된 경우 자동 업데이트를 다시 시작합니다.
RunWizard 업데이트를 설치하는 단계를 통해 로컬 사용자를 안내하는 마법사를 시작합니다.
저장 현재 자동 업데이트 설정을 적용합니다.
검색 업데이트를 동기식으로 검색합니다. 검색은 현재 구성된 검색 옵션을 사용합니다.
Setoption 서비스 ID를 지정하는 개체에 대한 옵션을 설정합니다. 또한 SetOption 메서드를 사용하여 자동 업데이트 등록을 변경할 때 경고가 표시되는지 여부를 확인합니다.
SetPassword 인증을 위해 프록시 서버에 제출할 암호를 설정합니다.
ShowSettingsDialog 자동 업데이트 대한 설정이 포함된 대화 상자를 표시합니다.
제거 업데이트의 동기 제거를 시작합니다.
UnregisterServiceWithAU 자동 업데이트 사용하여 서비스 등록을 취소합니다.

인터페이스

IAutomaticUpdates 자동 업데이트 기능을 포함합니다. (IAutomaticUpdates)
IAutomaticUpdates2 자동 업데이트 기능을 포함합니다. (IAutomaticUpdates2)
IAutomaticUpdatesResults 자동 업데이트 설명하는 읽기 전용 속성을 포함합니다.
IAutomaticUpdatesSettings 자동 업데이트 사용할 수 있는 설정을 포함합니다. (IAutomaticUpdatesSettings)
IAutomaticUpdatesSettings2 자동 업데이트 사용할 수 있는 설정을 포함합니다. (IAutomaticUpdatesSettings2)
IAutomaticUpdatesSettings3 자동 업데이트 사용할 수 있는 설정을 포함합니다. (IAutomaticUpdatesSettings3)
ICategory 업데이트가 속한 범주를 나타냅니다.
ICategoryCollection ICategory 인터페이스의 순서가 지정된 읽기 전용 목록을 나타냅니다.
IDownloadCompletedCallback 비동기 다운로드가 완료될 때 사용되는 콜백을 제공합니다.
IDownloadCompletedCallbackArgs 다운로드 완료에 대한 정보를 포함합니다. 이 인터페이스는 IDownloadCompletedCallback 대리자에 대한 매개 변수 역할을 합니다. 업데이트의 다운로드 및 설치는 비동기적입니다.
IDownloadJob 다운로드 작업에 사용할 수 있는 속성 및 메서드를 포함합니다.

IDownloadProgress 비동기 다운로드 작업의 진행률을 나타냅니다.
IDownloadProgressChangedCallback 비동기 다운로드 작업의 진행 상황을 나타내는 알림을 처리합니다.
IDownloadProgressChangedCallbackArgs 비동기 다운로드 작업의 진행 중인 변경 내용에 대한 정보를 포함합니다.
IDownloadResult 다운로드 작업의 결과를 나타냅니다.
IImageInformation 업데이트 또는 범주와 연결된 지역화된 이미지에 대한 정보를 포함합니다.
IInstallationAgent 업데이트 결과를 기록합니다.
IInstallationBehavior 업데이트의 설치 및 제거 옵션을 나타냅니다.
IInstallationCompletedCallback 비동기 설치 또는 제거가 완료되었음을 나타내는 알림을 처리합니다.
IInstallationCompletedCallbackArgs 설치 완료에 대한 정보를 포함하고 IInstallationCompletedCallback 대리자에 대한 매개 변수 역할을 합니다. 업데이트의 다운로드 및 설치는 비동기적입니다.
IInstallationJob 설치 또는 제거 작업에 사용할 수 있는 속성과 메서드를 포함합니다.
IInstallationProgress 비동기 설치 또는 제거의 진행률을 나타냅니다.
IInstallationProgressChangedCallback 비동기 설치 또는 제거의 진행 상황에 대한 알림을 처리하는 Invoke 메서드를 정의합니다.

IInstallationProgressChangedCallbackArgs 콜백이 수행되었을 때 비동기 설치 또는 제거 진행률의 변경 내용에 대한 정보를 포함합니다.
IInstallationResult 설치 또는 제거의 결과를 나타냅니다.
IInvalidProductLicenseException 제품에 대해 잘못된 라이선스가 검색될 때 throw되는 예외를 캡슐화합니다.
ISearchCompletedCallback 비동기 검색 작업의 완료에 대한 알림을 처리하는 메서드를 포함합니다.
ISearchCompletedCallbackArgs 비동기 검색 완료에 대한 정보를 포함합니다. 또한 SearchCompletedCallback 대리자에 대한 매개 변수 역할을 합니다.
ISearchJob 검색 작업에 사용할 수 있는 속성과 메서드를 포함합니다.
ISearchResult 검색 결과를 나타냅니다.
IStringCollection 정렬된 문자열 목록을 나타냅니다.
ISystemInformation 지정된 컴퓨터에 대한 정보를 포함합니다. 이 정보는 WUA(Windows 업데이트 에이전트)와 관련이 있습니다.
IUpdate 업데이트에 사용할 수 있는 속성 및 메서드를 포함합니다. (IUpdate)
IUpdate2 업데이트에 사용할 수 있는 속성 및 메서드를 포함합니다. (IUpdate2)
IUpdate3 업데이트에 사용할 수 있는 속성 및 메서드를 포함합니다. (IUpdate3)

IUpdate4 업데이트에 사용할 수 있는 속성 및 메서드를 포함합니다. (IUpdate4)
IUpdate5 업데이트에 사용할 수 있는 속성 및 메서드를 포함합니다. (IUpdate5)
IUpdateCollection 정렬된 업데이트 목록을 나타냅니다.
IUpdateDownloadContent 업데이트의 다운로드 콘텐츠를 나타냅니다. (IUpdateDownloadContent)
IUpdateDownloadContent2 업데이트의 다운로드 콘텐츠를 나타냅니다. (IUpdateDownloadContent2)
IUpdateDownloadContentCollection 업데이트에 대한 다운로드 콘텐츠의 컬렉션을 나타냅니다.
IUpdateDownloader 서버에서 업데이트를 다운로드합니다.
IUpdateDownloadResult 업데이트에 대한 다운로드 작업의 상태를 나타내는 속성을 포함합니다.
IUpdateException 불완전한 ISearchResult 개체에서 반환된 검색 결과의 측면에 대한 정보를 나타냅니다.
IUpdateExceptionCollection IUpdateException 인터페이스의 순서가 지정된 읽기 전용 목록을 나타냅니다.
IUpdateHistoryEntry 업데이트의 기록된 기록을 나타냅니다. (IUpdateHistoryEntry)
IUpdateHistoryEntry2 업데이트의 기록된 기록을 나타냅니다. (IUpdateHistoryEntry2)

IUpdateHistoryEntryCollection IUpdateHistoryEntry 인터페이스의 순서가 지정된 읽기 전용 목록을 나타냅니다.
IUpdateIdentity 업데이트의 고유 식별자를 나타냅니다.
IUpdateInstallationResult 업데이트 설치 또는 제거의 상태에 사용할 수 있는 속성 및 메서드를 포함합니다.
IUpdateInstaller 컴퓨터에서 또는 컴퓨터에 업데이트를 설치하거나 제거합니다.
IUpdateInstaller2 컴퓨터에 업데이트를 설치하거나 제거합니다. (IUpdateInstaller2)
IUpdateInstaller3 컴퓨터에 업데이트를 설치하거나 제거합니다. (IUpdateInstaller3)
IUpdateInstaller4 이전에 스테이징되거나 설치된 업데이트를 마무리하는 메서드를 제공합니다.
IUpdateLockdown 이 인터페이스의 메서드를 구현하는 개체의 메서드 및 속성에 대한 액세스를 제한합니다.
IUpdateSearcher 서버에서 업데이트를 검색합니다. (IUpdateSearcher)
IUpdateSearcher2 서버에서 업데이트를 검색합니다. (IUpdateSearcher2)
IUpdateSearcher3 서버에서 업데이트를 검색합니다. (IUpdateSearcher3)
IUpdateService WUA(Windows 업데이트 Agent) 또는 자동 업데이트 등록된 서비스에 대한 정보를 포함합니다. (IUpdateService)

IUpdateService2 WUA(Windows 업데이트 Agent) 또는 자동 업데이트 등록된 서비스에 대한 정보를 포함합니다. (IUpdateService2)
IUpdateServiceCollection IUpdateService 인터페이스 목록을 나타냅니다.
IUpdateServiceManager Windows 업데이트 에이전트 또는 자동 업데이트 업데이트 서비스의 등록을 추가하거나 제거합니다. (IUpdateServiceManager)
IUpdateServiceManager2 Windows 업데이트 에이전트 또는 자동 업데이트 업데이트 서비스의 등록을 추가하거나 제거합니다. (IUpdateServiceManager2)
IUpdateServiceRegistration 서비스의 등록 상태에 대한 정보를 포함합니다.
IUpdateSession 호출자가 업데이트를 포함하는 작업을 수행할 수 있는 세션을 나타냅니다. 예를 들어 이 인터페이스는 호출자가 검색, 다운로드, 설치 또는 제거 작업을 수행하는 세션을 나타냅니다. (IUpdateSession)
IUpdateSession2 호출자가 업데이트를 포함하는 작업을 수행할 수 있는 세션을 나타냅니다. 예를 들어 이 인터페이스는 호출자가 검색, 다운로드, 설치 또는 제거 작업을 수행하는 세션을 나타냅니다. (IUpdateSession2)
IUpdateSession3 호출자가 업데이트를 포함하는 작업을 수행할 수 있는 세션을 나타냅니다. 예를 들어 이 인터페이스는 호출자가 검색, 다운로드, 설치 또는 제거 작업을 수행하는 세션을 나타냅니다. (IUpdateSession3)
IWebProxy HTTP 프록시 설정을 포함합니다.
IWindowsDriverUpdate Windows 드라이버 업데이트에서만 사용할 수 있는 속성 및 메서드를 포함합니다.

IWindowsDriverUpdate2 Windows 드라이버 업데이트에서만 사용할 수 있는 속성과 메서드를 포함합니다. (IWindowsDriverUpdate2)
IWindowsDriverUpdate3 Windows 드라이버 업데이트에서만 사용할 수 있는 속성과 메서드를 포함합니다. (IWindowsDriverUpdate3)
IWindowsDriverUpdate4 Windows 드라이버 업데이트에서만 사용할 수 있는 속성과 메서드를 포함합니다. (IWindowsDriverUpdate4)
IWindowsDriverUpdate5 Windows 드라이버 업데이트에서만 사용할 수 있는 속성과 메서드를 포함합니다. (IWindowsDriverUpdate5)
IWindowsDriverUpdateEntry Windows 드라이버 업데이트에서만 사용할 수 있는 속성을 포함합니다.
IWindowsDriverUpdateEntryCollection 드라이버 업데이트와 연결된 드라이버 업데이트 항목의 컬렉션을 포함합니다. 모든 속성에는 표준 컬렉션 의미 체계가 있습니다.
IWindowsUpdateAgentInfo WUA(Windows 업데이트 Agent) 버전에 대한 정보를 검색합니다.

Windows 웹 서비스

아티클2023. 03. 16.

Windows Web Services 기술 개요입니다.

Windows 웹 서비스를 개발하려면 다음 헤더가 필요합니다.

- icontentprefetchertasktrigger.h
- webservices.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- Windows 웹 서비스

열거형

WS_ADDRESSING_VERSION 주소 지정 헤더에 사용되는 사양의 버전을 식별합니다.
WS_BINDING_TEMPLATE_TYPE 지원되는 다양한 보안 바인딩 조합의 열거형입니다.
WS_CALL_PROPERTY_ID 클라이언트 쪽 서비스 작업에서 호출을 구성하기 위한 선택적 매개 변수입니다.
WS_CALLBACK_MODEL 콜백의 스레딩 동작(예: WS_ASYNC_CALLBACK)을 지정합니다.
WS_CERT_CREDENTIAL_TYPE WS_CERT_CREDENTIAL 하위 형식의 선택기로 사용되는 인증서 자격 증명의 형식입니다.
WS_CHANNEL_BINDING 채널에 사용할 프로토콜 스택을 나타냅니다.
WS_CHANNEL_PROPERTY_ID 각 채널 속성은 ID로 식별되며 연결된 값을 가집니다. 채널을 만들 때 속성을 지정하지 않으면 기본값이 사용됩니다.

WS_CHANNEL_STATE 채널이 있을 수 있는 다양한 상태입니다.
WS_CHANNEL_TYPE 세션인지 여부 및 지원되는 통신 방향과 같은 채널의 기본 특성을 나타냅니다.
WS_CHARSET 문서의 문자 집합을 식별합니다.
WS_COOKIE_MODE HTTP 쿠키를 처리하는 방법을 지정하는 데 사용되는 열거형입니다.
WS_DATETIME_FORMAT WS_DATETIME 텍스트 형식을 지정합니다.
WS_ENCODING 다른 인코딩(메시지 형식)입니다.
WS_ENDPOINT_ADDRESS_EXTENSION_TYPE WS_ENDPOINT_ADDRESS 확장 필드 내의 확장 형식을 식별합니다.
WS_ENDPOINT_IDENTITY_TYPE WS_ENDPOINT_IDENTITY 하위 형식의 선택기로 사용되는 엔드포인트 ID의 형식입니다.
WS_ENVELOPE_VERSION 봉투 구조에 사용되는 사양의 버전입니다.
WS_ERROR_PROPERTY_ID 오류와 연결된 속성 값 집합입니다. WsGetErrorProperty 및 WsSetErrorProperty를 사용하여 설정 및 검색됩니다.
WS_EXCEPTION_CODE 이 구성 요소에서 throw하는 구조적 예외 코드입니다. 이러한 예외는 치명적이며 애플리케이션에서 처리해서는 안 됩니다.
WS_EXTENDED_PROTECTION_POLICY 확장된 보호 데이터의 유효성을 검사해야 하는지를 정의합니다.

WS_EXTENDED_PROTECTION_SCENARIO 확장된 보호의 유효성을 검사하는 방법을 정의합니다.
WS_FAULT_DISCLOSURE 오류에 포함된 오류 정보의 양을 제어합니다. 오류 개체는 오류 문자열의 일부로 중요한 데이터를 포함할 수 있으므로 오류 문자열 정보를 모든 오류에 포함하는 것이 항상 적절하지는 않습니다.
WS_FAULT_ERROR_PROPERTY_ID 오류에 대한 정보입니다.
WS_FIELD_MAPPING 구조체의 필드가 XML로 표시되는 방식을 지정합니다. WS_FIELD_DESCRIPTION 내에서 사용됩니다.
WS_HEADER_TYPE 헤더의 형식을 식별합니다.
WS_HEAP_PROPERTY_ID 각 힙 속성은 ID로 식별되며 연결된 값을 가집니다.
WS_HTTP_HEADER_AUTH_TARGET HTTP 헤더 인증 보안 바인딩의 대상을 정의합니다.
WS_HTTP_PROXY_SETTING_MODE 프록시 설정은 바인딩 WS_HTTP_CHANNEL_BINDING 있는 채널에 대한 HTTP 프록시 설정을 나타냅니다. WS_CHANNEL_PROPERTY_HTTP_PROXY_SETTING_MODE 채널 속성의 일부로 지정됩니다.
WS_IP_VERSION IP 버전을 지정합니다.
WS_LISTENER_PROPERTY_ID 각 수신기 속성은 WS_LISTENER_PROPERTY 형식이며 ID로 식별되며 연결된 값을 가집니다. 수신기를 만들 때 속성을 지정하지 않으면 기본값이 사용됩니다.
WS_LISTENER_STATE 수신기가 있을 수 있는 다양한 상태입니다.

WS_MESSAGE_INITIALIZATION WsInitializeMessageshould가 메시지에 추가하는 헤더를 지정합니다.
WS_MESSAGE_PROPERTY_ID 각 메시지 속성은 WS_MESSAGE_PROPERTY 형식이며 ID로 식별되며 연결된 값을 가집니다.
WS_MESSAGE_SECURITY_USAGE 메시지 보안 바인딩이 WS-Security 메커니즘을 사용하여 메시지에 해당하는 보안 토큰을 연결하는 방법을 정의합니다.
WS_MESSAGE_STATE 메시지가 있을 수 있는 다양한 상태입니다.
WS_METADATA_EXCHANGE_TYPE 형식의 메타데이터 교환을 사용하도록 설정하고 사용하지 않도록 설정하는 방법에 대한 정보입니다.
WS_METADATA_PROPERTY_ID 각 메타데이터 속성은 WS_METADATA_PROPERTY 형식이며 ID로 식별되며 연결된 값을 가집니다. 메타데이터를 만들 때 속성을 지정하지 않으면 기본값이 사용됩니다.
WS_METADATA_STATE 메타데이터 개체의 상태입니다.
WS_MOVE_TO 이 열거형은 xml 문서에 대해 이동하는 다양한 방법을 식별합니다.
WS_OPERATION_CONTEXT_PROPERTY_ID 컨텍스트에서 사용할 수 있는 속성입니다. 컨텍스트의 지정된 지점에서 모든 속성을 사용할 수 있는 것은 아닙니다. 모든 컨텍스트 속성은 WsGetOperationContextProperty를 통해 사용할 수 있습니다.
WS_OPERATION_STYLE 다양한 작업 스타일의 열거형입니다.
WS_PARAMETER_TYPE 다른 매개 변수 형식입니다.

WS_POLICY_EXTENSION_TYPE 이 열거형의 값은 WS_POLICY_EXTENSION 하위 형식을 식별하는 데 사용됩니다.
WS_POLICY_PROPERTY_ID 각 정책 속성 및 관련 값을 식별합니다.
WS_POLICY_STATE 정책 개체의 상태입니다.
WS_PROTECTION_LEVEL 보내고 받은 메시지에 필요한 무결성 및 기밀 수준을 정의합니다.
WS_PROXY_PROPERTY_ID 서비스 프록시를 구성하기 위한 선택적 매개 변수입니다. WS_PROXY_PROPERTY_STATE 예외를 제외하고 모든 값은 WS_PROXY_PROPERTY* 매개 변수의 일부로 WsCreateServiceProxy에서만 사용할 수 있습니다.
WS_READ_OPTION 값이 필요한지 여부와 값을 할당하는 방법을 지정합니다.
WS_RECEIVE_OPTION 채널에서 수신할 때 메시지가 필요한지 여부를 지정합니다.
WS_REPEATING_HEADER_OPTION 이 열거형은 헤더가 메시지에 두 번 이상 나타날지 여부를 지정하는 데 사용됩니다.
WS_REQUEST_SECURITY_TOKEN_ACTION WS-Trust를 사용하여 보안 토큰을 협상할 때 사용할 작업 집합을 정의합니다.
WS_REQUEST_SECURITY_TOKEN_PROPERTY_ID 발급자에서 보안 토큰을 요청하기 위한 속성을 식별합니다. WsRequestSecurityToken은 WS_REQUEST_SECURITY_TOKEN_PROPERTY* 매개 변수의 일부로 사용됩니다.
WS_SAML_AUTHENTICATOR_TYPE 들어오는 SAML 토큰의 유효성을 검사하기 위해 서버 쪽(예: 신뢰 당사자)에서 사용되는 SAML 토큰 인증자의 형식 ID입니다.

WS_SECURE_CONVERSATION_VERSION 메시지 보안 및 혼합 모드 보안에 사용할 WS-SecureConversation 사양 버전을 정의합니다.
WS_SECURITY_ALGORITHM_ID WS-Security와 함께 사용할 보안 알고리즘을 정의합니다. 이러한 값은 메시지 보안 바인딩 및 혼합 모드 보안 바인딩과 관련이 있습니다.
WS_SECURITY_ALGORITHM_PROPERTY_ID 보안 알고리즘 노브를 나타내는 속성을 식별합니다.
WS_SECURITY_ALGORITHM_SUITE_NAME 서명 및 암호화와 같은 작업에 사용되는 보안 알고리즘 제품군입니다. 이 열거형의 값은 WS-SecurityPolicy 1.1ection 7.1에 정의된 제품군에 해당합니다.
WS_SECURITY_BINDING_CONSTRAINT_TYPE 이 열거형의 값은 WS_SECURITY_BINDING_CONSTRAINT 하위 형식을 식별하는 데 사용됩니다.
WS_SECURITY_BINDING_PROPERTY_ID 보안 바인딩 설정을 지정하는 데 사용되는 속성을 식별합니다. 보안 바인딩 설정은 보안 설명에서 사용되는 보안 바인딩에 있습니다.
WS_SECURITY_BINDING_TYPE WS_SECURITY_BINDING 하위 형식의 선택기로 사용되는 보안 바인딩의 형식입니다.
WS_SECURITY_CONTEXT_PROPERTY_ID 보안 컨텍스트 개체의 속성을 식별합니다. 이 열거형은 WsGetSecurityContextProperty와 함께 사용됩니다.
WS_SECURITY_HEADER_LAYOUT WS-Security 보안 헤더의 요소에 적용되는 레이아웃 규칙입니다. 이 설정은 메시지 보안 바인딩 및 혼합 모드 보안 바인딩과 관련이 있습니다.
WS_SECURITY_HEADER_VERSION 메시지 보안 및 혼합 모드 보안에 사용할 WS-Security 사양 버전입니다.
WS_SECURITY_KEY_ENTROPY_MODE 메시지 및 혼합 모드 보안으로 수행된 보안 토큰 협상 중에 발급된 키에 임의성을 적용하는 방법을 정의합니다.

WS_SECURITY_KEY_HANDLE_TYPE 보안 키의 유형입니다.
WS_SECURITY_KEY_TYPE 보안 토큰의 키 형식입니다. 키에 대해 보안 토큰을 쿼리할 때 반환 형식으로 사용됩니다. 보안 토큰 서비스에서 보안 토큰을 요청할 때 필요한 키 형식을 지정하는 데도 사용됩니다.
WS_SECURITY_PROPERTY_ID 채널 전체 보안 설정을 나타내는 속성을 식별합니다. 이 열거형은 WS_SECURITY_PROPERTY 구조 내에서 사용되며, WS_SECURITY_DESCRIPTION 구조 내에서 사용됩니다.
WS_SECURITY_TIMESTAMP_USAGE 메시지 보안 및 혼합 모드 보안을 사용하여 WS-Security 헤더에서 타임스탬프 요소를 생성하고 요구해야 하는 시기를 정의합니다.
WS_SECURITY_TOKEN_PROPERTY_ID 보안 토큰에서 추출할 수 있는 필드 및 속성의 키를 정의합니다. 모든 보안 토큰 형식에 대해 모든 속성이 유효하지는 않습니다. WsGetSecurityTokenProperty 함수는 여기에 정의된 값을 키로 사용합니다.
WS_SECURITY_TOKEN_REFERENCE_MODE 메시지 및 혼합 모드 보안 바인딩을 사용하여 서명, 암호화된 항목 및 파생 토큰에서 보안 토큰을 참조하는 데 사용하는 메커니즘입니다.
WS_SERVICE_CANCEL_REASON 취소 이유입니다.
WS_SERVICE_ENDPOINT_PROPERTY_ID 각 속성은 지정된 WS_SERVICE_ENDPOINT 구조를 구성하기 위한 선택적 매개 변수를 나타냅니다. 이 열거형은 WS_SERVICE_ENDPOINT 일부인 WS_SERVICE_ENDPOINT_PROPERTY 구조 내에서 사용됩니다.
WS_SERVICE_HOST_STATE 서비스 호스트가 있을 수 있는 상태입니다.
WS_SERVICE_PROPERTY_ID 서비스 호스트를 구성하기 위한 선택적 매개 변수입니다. 이 열거형은 WsCreateServiceHost를 호출할 때 또는 WsGetServiceHostProperty를 호출할 때 자체에서 WS_SERVICE_PROPERTY 구조 내에서 사용됩니다.

WS_SERVICE_PROXY_STATE 서비스 프록시의 상태입니다.
WS_TRACE_API WS_TRACE_API 열거형입니다.
WS_TRANSFER_MODE 전송되거나 수신된 메시지가 스트리밍되거나 버퍼링되는지 여부입니다.
WS_TRUST_VERSION 메시지 보안 및 혼합 모드 보안에 사용할 WS-Trust 사양 버전을 정의합니다.
WS_TYPE serialization에 지원되는 형식입니다.
WS_TYPE_MAPPING 직렬화되거나 역직렬화될 때 WS_TYPE XML 간 매핑 방법
WS_URL_SCHEME_TYPE WsDecodeUrl, WsEncodeUrl 및 WsCombineUrl과 함께 사용되는 스키마 집합입니다.
WS_USERNAME_CREDENTIAL_TYPE WS_USERNAME_CREDENTIAL 하위 형식의 선택기로 사용되는 사용자 이름/암호 자격 증명의 형식입니다.
WS_VALUE_TYPE 고정 크기 기본 형식의 형식입니다.
WS_WINDOWS_INTEGRATED_AUTH_CREDENTIAL_TYPE WS_WINDOWS_INTEGRATED_AUTH_CREDENTIAL 하위 형식의 선택기로 사용되는 Windows 통합 인증 자격 증명의 형식입니다.
WS_WINDOWS_INTEGRATED_AUTH_PACKAGE Windows 통합 인증에 사용할 특정 SSP 패키지를 정의합니다.

WS_WRITE_OPTION 지정된 스토리지에 값이 포함되어 있는지 또는 값에 대한 포인터를 포함할지, XML 콘텐츠에서 값을 nil로 나타낼 수 있는지 여부를 지정합니다.
WS_XML_BUFFER_PROPERTY_ID 각 XML 버퍼 속성은 ID로 식별되며 연결된 값을 가집니다.
WS_XML_CANONICALIZATION_ALGORITHM XML 정식화 알고리즘의 값입니다.
WS_XML_CANONICALIZATION_PROPERTY_ID 각 XML 정식화 속성과 연결된 값을 식별합니다. 이 열거형은 WsStartReaderCanonicalization 및 WsStartWriterCanonicalization에 대한 매개 변수로 사용되는 WS_XML_CANONICALIZATION_PROPERTY 구조 내에서 사용됩니다.
WS_XML_NODE_TYPE WS_XML_NODE 구조체의 형식입니다.
WS_XML_READER_ENCODING_TYPE WS_XML_READER_ENCODING 구조체의 형식입니다.
WS_XML_READER_INPUT_TYPE WS_XML_READER_INPUT 구조체의 형식입니다.
WS_XML_READER_PROPERTY_ID 각 XML 판독기 속성과 연결된 값을 식별합니다.
WS_XML_SECURITY_TOKEN_PROPERTY_ID XML 보안 토큰을 만들기 위한 속성 모음의 키입니다. 이 열거형은 WsCreateXmlSecurityToken에 대한 매개 변수로 사용되는 WS_XML_SECURITY_TOKEN_PROPERTY 구조 내에서 사용됩니다.
WS_XML_TEXT_TYPE WS_XML_TEXT 구조체의 형식입니다.
WS_XML_WRITER_ENCODING_TYPE WS_XML_WRITER_ENCODING 구조체의 형식입니다.

WS_XML_WRITER_OUTPUT_TYPE WS_XML_WRITER_OUTPUT 구조체의 형식입니다.
WS_XML_WRITER_PROPERTY_ID 각 xml 기록기 속성은 ID로 식별되며 연결된 값을 가집니다.

Functions

IsRegisteredForContentPrefetch 앱 패키지가 콘텐츠 프리페치 백그라운드 작업에 등록되었는지를 나타냅니다.
TriggerContentPrefetcherTask 지정된 앱 패키지에 대한 콘텐츠 프리페치 백그라운드 작업을 트리거합니다.
WS_ABANDON_MESSAGE_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsAbandonMessage 호출을 처리합니다.
WS_ABORT_CHANNEL_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsAbortChannel 호출을 처리합니다.
WS_ABORT_LISTENER_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsAbortListener 호출을 처리합니다.
WS_ACCEPT_CHANNEL_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsAcceptChannel 호출을 처리합니다.
WS_ASYNC_CALLBACK 비동기 모델에 사용되는 콜백 함수 매개 변수입니다.
WS_ASYNC_FUNCTION WsAsyncExecute와 함께 일련의 비동기 작업에서 호출할 다음 함수를 지정하는 데 사용됩니다. (WS_ASYNC_FUNCTION)

WS_CERT_ISSUER_LIST_NOTIFICATION_CALLBACK 서버에 허용되는 인증서 발급자 목록을 클라이언트에 알립니다.
WS_CERTIFICATE_VALIDATION_CALLBACK HTTP 서버에 대한 연결이 설정되고 헤더가 전송될 때 인증서의 유효성을 검사하기 위해 WS_CERTIFICATE_VALIDATION_CALLBACK 콜백이 호출됩니다.
WS_CLOSE_CHANNEL_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsCloseChannel 호출을 처리합니다.
WS_CLOSE_LISTENER_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsCloseListener 호출을 처리합니다.
WS_CREATE_CHANNEL_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsCreateChannel 호출을 처리합니다.
WS_CREATE_CHANNEL_FOR_LISTENER_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsCreateChannelForListener 호출을 처리합니다.
WS_CREATE_DECODER_CALLBACK 디코더 인스턴스 만들기를 처리합니다.
WS_CREATE_ENCODER_CALLBACK 인코더 인스턴스 만들기를 처리합니다.
WS_CREATE_LISTENER_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsCreateListener 호출을 처리합니다.
WS_DECODER_DECODE_CALLBACK 메시지를 디코딩합니다.
WS_DECODER_END_CALLBACK 메시지의 끝을 디코딩합니다.
WS_DECODER_GET_CONTENT_TYPE_CALLBACK 메시지의 콘텐츠 형식을 가져옵니다. (WS_DECODER_GET_CONTENT_TYPE_CALLBACK)

WS_DECODER_START_CALLBACK 메시지 디코딩을 시작합니다.
WS_DURATION_COMPARISON_CALLBACK 두 기간을 비교합니다.
WS_DYNAMIC_STRING_CALLBACK 지정된 문자열을 최적화된 형식으로 작성할 수 있는지 여부를 결정합니다.
WS_ENCODER_ENCODE_CALLBACK 메시지를 인코딩합니다.
WS_ENCODER_END_CALLBACK 메시지의 끝을 인코딩합니다.
WS_ENCODER_GET_CONTENT_TYPE_CALLBACK 메시지의 콘텐츠 형식을 가져옵니다. (WS_ENCODER_GET_CONTENT_TYPE_CALLBACK)
WS_ENCODER_START_CALLBACK 메시지 인코딩을 시작합니다.
WS_FREE_CHANNEL_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsFreeChannel 호출을 처리합니다.
WS_FREE_DECODER_CALLBACK 디코더 인스턴스 해제를 처리합니다.
WS_FREE_ENCODER_CALLBACK 인코더 인스턴스 해제를 처리합니다.
WS_FREE_LISTENER_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsFreeListener 호출을 처리합니다.
WS_GET_CERT_CALLBACK 보안 런타임에 인증서를 제공합니다.

WS_GET_CHANNEL_PROPERTY_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsGetChannelProperty 호출을 처리합니다.
WS_GET_LISTENER_PROPERTY_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsGetListenerProperty 호출을 처리합니다.
WS_HTTP_REDIRECT_CALLBACK RFC2616에 설명된 대로 HTTP 자동 리디렉션 기능을 사용하여 메시지가 자동으로 다른 서비스로 리디렉션될 때 호출됩니다.
WS_IS_DEFAULT_VALUE_CALLBACK 값이 기본값인지 여부를 확인합니다.
WS_MESSAGE_DONE_CALLBACK 메시지가 WsReadEnvelopeStartfunction에 제공된 WS_XML_READER 구조체 또는 WsWriteEnvelopeStart 함수에 제공된 WS_XML_WRITER 구조체의 사용을 완료했음을 호출자에게 알릴 수 있습니다.
WS_OPEN_CHANNEL_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsOpenChannel 호출을 처리합니다.
WS_OPEN_LISTENER_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsOpenListener 호출을 처리합니다.
WS_OPERATION_CANCEL_CALLBACK 서비스 호스트의 중단된 종료로 인해 비동기 서비스 작업 호출의 취소에 대한 알림을 제공합니다.
WS_OPERATION_FREE_STATE_CALLBACK 애플리케이션에서 취소 콜백에 등록된 상태 정보를 정리할 수 있습니다.
WS_PROXY_MESSAGE_CALLBACK 입력 메시지의 헤더를 보내려고 하거나 출력 메시지 헤더가 방금 수신될 때 호출됩니다.
WS_PULL_BYTES_CALLBACK WsPullBytes 함수에서 작성해야 하는 데이터를 요청하는 데 사용됩니다.

WS_PUSH_BYTES_CALLBACK WsPushBytes 함수에서 데이터를 기록하도록 요청하는 데 사용됩니다.
WS_READ_CALLBACK WS_XML_READERto 일부 원본에서 버퍼로 읽는 데 사용됩니다.
WS_READ_MESSAGE_END_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsReadMessageEnd 호출을 처리합니다.
WS_READ_MESSAGE_START_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsReadMessageStart 호출을 처리합니다.
WS_READ_TYPE_CALLBACK WS_TYPEhas 지정되면 값을 읽습니다.
WS_RESET_CHANNEL_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsResetChannel 호출을 처리합니다.
WS_RESET_LISTENER_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsResetListener 호출을 처리합니다.
WS_SERVICE_ACCEPT_CHANNEL_CALLBACK 서비스 호스트가 엔드포인트 수신기에서 채널을 수락할 때 호출됩니다.
WS_SERVICE_CLOSE_CHANNEL_CALLBACK 채널이 엔드포인트에서 닫혀 있거나 중단될 때 호출됩니다.
WS_SERVICE_MESSAGE_RECEIVE_CALLBACK defaultMessageHandlerCallback이 설정된 WS_SERVICE_CONTRACT 구성된 엔드포인트에서 WS_MESSAGE 수신될 때 호출됩니다.
WS_SERVICE_SECURITY_CALLBACK 들어오는 메시지의 헤더가 수신되고 본문이 처리되지 않을 때 호출됩니다.
WS_SERVICE_STUB_CALLBACK 서비스 작업 호출에 위임하기 위해 서비스 모델에서 호출됩니다.

WS_SET_CHANNEL_PROPERTY_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsSetChannelProperty 호출을 처리합니다.
WS_SET_LISTENER_PROPERTY_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsSetListenerProperty 호출을 처리합니다.
WS_SHUTDOWN_SESSION_CHANNEL_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsShutdownSessionChannel 호출을 처리합니다.
WS_STRING_VALUE 상수 문자열이 지정된 WS_STRING 구조를 초기화합니다.
WS_VALIDATE_PASSWORD_CALLBACK 수신기 쪽에서 사용자 이름/암호 쌍의 유효성을 검사합니다.
WS_VALIDATE_SAML_CALLBACK SAML 어설션의 유효성을 검사합니다.
WS_WRITE_CALLBACK WS_XML_WRITER 함수에서 사용자가 결정한 대상에 지정된 버퍼를 쓰는 데 사용됩니다.
WS_WRITE_MESSAGE_END_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsWriteMessageEnd 호출을 처리합니다.
WS_WRITE_MESSAGE_START_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsWriteMessageStart 호출을 처리합니다.
WS_WRITE_TYPE_CALLBACK WS_CUSTOM_TYPEhas 지정될 때 요소를 작성하기 위해 호출됩니다.
WS_XML_STRING_DICTIONARY_VALUE 연결된 사전 ID가 있는 경우 WS_XML_STRING 구조에 대한 이니셜라이저를 제공합니다.
WS_XML_STRING_VALUE 연결된 사전 ID가 없는 경우 WS_XML_STRING 구조에 대한 이니셜라이저를 제공합니다.

WsAbandonCall 지정된 서비스 프록시에서 지정된 호출을 중단합니다.
WsAbandonMessage 지정된 채널에서 지정된 메시지의 나머지를 건너뜁니다.
WsAbortChannel 지정된 채널에 대해 보류 중인 모든 I/O를 취소합니다.
WsAbortListener 지정된 수신기에 대해 보류 중인 I/O를 취소합니다.
WsAbortServiceHost 지정된 서비스 호스트에서 모든 현재 작업을 중단합니다.
WsAbortServiceProxy 서비스 프록시를 중단하고 서비스 프록시에서 보류 중인 I/O를 취소합니다.
WsAcceptChannel 지정된 수신기에서 들어오는 다음 메시지를 수락합니다.
WsAddCustomHeader 지정된 애플리케이션 정의 헤더를 메시지에 추가합니다.
WsAddErrorString 지정된 오류 문자열을 오류 개체에 추가합니다.
WsAddMappedHeader 지정된 매핑된 헤더를 메시지에 추가합니다.
WsAddressMessage 지정된 엔드포인트 주소로 메시지를 주소 지정합니다.
WsAlloc 지정된 힙에서 메모리 세그먼트를 할당합니다.

WsAsyncExecute 비동기 작업을 구현하기 위한 도우미 함수입니다.
WsCall 서비스 프록시에서 내부적으로 지정된 메타데이터에 따라 지정된 인수의 형식을 지정하고 메시지로 보내는 데 사용됩니다. 애플리케이션은 이 함수를 직접 호출해서는 안 됩니다.
WsCheckMustUnderstandHeaders 지정된 헤더가 수신자가 인식했는지 확인합니다. 이 함수는 받은 메시지에 대해 모든 헤더를 읽은 후에 호출해야 합니다. .
WsCloseChannel 지정된 채널을 닫습니다.
WsCloseListener 지정된 수신기가 수신을 중지하도록 합니다.
WsCloseServiceHost 지정된 서비스 호스트와의 통신을 닫습니다.
WsCloseServiceProxy 지정된 서비스 프록시와의 통신을 닫습니다.
WsCombineUrl 지정된 URL 참조(절대 또는 상대 URL) 및 지정된 절대 기본 URL에서 절대 URL을 생성합니다.
WsCopyError 지정된 원본에서 지정된 대상으로 오류 개체를 복사합니다.
WsCopyNode 지정된 XML 판독기에서 지정된 XML 작성기에 현재 노드를 복사합니다.
WsCountOf 지정된 배열의 요소 수를 반환합니다.
WsCreateChannel 엔드포인트를 사용하여 메시지 교환 채널을 만듭니다.

WsCreateChannelForListener 지정된 수신기와 연결된 채널을 만듭니다.
WsCreateError 다양한 오류 정보를 기록하기 위해 함수에 전달할 수 있는 오류 개체를 만듭니다.
WsCreateFaultFromError 지정된 오류 개체에서 WS_FAULT 생성합니다.
WsCreateHeap 힙 개체를 만듭니다.
WsCreateListener 지정된 속성을 사용하여 수신기를 만듭니다.
WsCreateMessage 지정된 속성을 사용하여 메시지 개체를 만듭니다.
WsCreateMessageForChannel 지정된 채널에서 사용할 메시지를 만듭니다.
WsCreateMetadata 메타데이터 문서를 수집하고 처리하는 데 사용되는 메타데이터 개체를 만듭니다.
WsCreateReader 지정된 속성을 사용하여 XML 판독기를 만듭니다.
WsCreateServiceEndpointFromTemplate 정책 템플릿에서 서비스 엔드포인트(WS_SERVICE_ENDPOINT)를 만들기 위한 도우미 루틴입니다.
WsCreateServiceHost 지정된 엔드포인트에 대한 서비스 호스트를 만듭니다.
WsCreateServiceProxy 지정된 속성을 사용하여 서비스 프록시를 만듭니다.

WsCreateServiceProxyFromTemplate 정책 템플릿에서 서비스 프록시를 만들기 위한 도우미 루틴입니다.
WsCreateWriter 는 지정된 속성을 사용하여 XML 기록기를 만듭니다.
WsCreateXmlBuffer XML 데이터를 처리하는 데 사용할 수 있는 XML 버퍼를 만듭니다.
WsCreateXmlSecurityToken 지정된 XML 양식에서 보안 토큰을 만듭니다.
WsDateTimeToFileTime WS_DATETIME 개체를 FILETIME 개체로 변환합니다. FILETIME 개체에 대한 참조는 출력 매개 변수에 의해 반환됩니다.
WsDecodeUrl URL의 구성 요소를 평가하여 "체계"를 결정합니다.
WsEncodeUrl 구성 요소 파트가 지정된 경우 지정된 WS_URL URL 문자열로 인코딩합니다. 값은 필요에 따라 이스케이프되고, 결합되고, 지정된 WS_HEAP 저장되며 결과는 WS_STRING 반환됩니다.
WsEndReaderCanonicalization 이 함수는 이전 WsStartReaderCanonicalization 함수 호출로 시작된 XML 정식화를 중지합니다. 판독기에서 버퍼링된 나머지 정식 바이트는 콜백 함수에 기록됩니다.
WsEndWriterCanonicalization 이 함수는 이전 WsStartWriterCanonicalization 호출에서 시작된 XML 정식화를 중지합니다. 기록기에서 버퍼링된 나머지 정식 바이트는 콜백 함수에 기록됩니다.
WsFileTimeToDateTime FILETIME 개체에 대한 참조를 사용하여 WS_DATETIME 개체로 변환합니다. WS_DATETIME 개체에 대한 참조는 출력 매개 변수에 의해 반환됩니다.
WsFillBody 메시지에서 읽기에 사용할 수 있는 바이트 수가 충분한지 확인합니다.

WsFillReader 판독기에서 후속 판독기 함수에서 사용할 XML 데이터의 최소 바이트 수를 버퍼링했는지 확인합니다.
WsFindAttribute 현재 요소의 특성을 검색하여 지정된 이름과 네임스페이스가 있는 특성을 검색하고 WsReadStartAttribute에 전달될 수 있는 인덱스를 반환합니다.
WsFlushBody 기록된 모든 누적된 메시지 본문 데이터를 플러시합니다.
WsFlushWriter 충분한 데이터가 버퍼링된 경우 WS_XML_WRITER_STREAM_OUTPUT 지정된 콜백을 호출하도록 기록기에 지시합니다.
WsFreeChannel Channel 개체와 연결된 메모리 리소스를 해제합니다.
WsFreeError WsCreateError를 사용하여 만든 Error 개체와 연결된 메모리 리소스를 해제합니다. 그러면 개체 및 해당 구성 요소가 해제됩니다.
WsFreeHeap 이렇게 하면 WsAlloc를 사용하여 힙 개체 및 할당과 연결된 메모리가 해제됩니다.
WsFreeListener 수신기 개체와 연결된 메모리 리소스를 해제합니다.
WsFreeMessage Message 개체와 연결된 메모리 리소스를 해제합니다.
WsFreeMetadata 메타데이터 개체와 연결된 메모리 리소스를 해제합니다.
WsFreeReader XML_Reader 개체와 연결된 메모리 리소스를 해제합니다.

WsFreeSecurityToken 보안 토큰 개체와 연결된 메모리 리소스를 해제합니다.
WsFreeServiceHost 서비스 호스트 개체와 연결된 메모리를 해제합니다.
WsFreeServiceProxy 서비스 프록시 리소스와 연결된 메모리를 해제합니다.
WsFreeWriter XML 기록기 개체와 연결된 메모리 리소스를 해제합니다.
WsGetChannelProperty 채널 매개 변수에서 참조하는 채널의 속성을 검색합니다.
WsGetCustomHeader 메시지의 애플리케이션 정의 헤더를 찾아 역직렬화합니다.
WsGetDictionary XML Dictionary 개체를 검색합니다. 검색된 사전은 사전 참조 매개 변수에 의해 반환됩니다.
WsGetErrorProperty 오류 매개 변수에서 참조하는 WS_ERROR 개체의 속성을 검색합니다.
WsGetErrorString 오류 개체에서 오류 문자열을 검색합니다.
WsGetFaultErrorDetail WS_ERROR 개체에 저장된 오류 세부 정보를 읽습니다.
WsGetFaultErrorProperty 오류 매개 변수에서 참조하는 WS_ERROR 개체의 오류 오류 속성을 검색합니다.
WsGetHeader 메시지에서 특정 표준 헤더를 찾아 역직렬화합니다.

WsGetHeaderAttributes 이 함수는 ULONG 매개 변수를 판독기를 배치할 헤더 요소의 WS_HEADER_ATTRIBUTES 채웁니다. 메시지의 봉투 버전은 반환할 특성을 결정하는 데 사용됩니다.
WsGetHeapProperty 지정된 힙의 특정 속성을 검색합니다.
WsGetListenerProperty 지정된 수신기 개체 속성을 검색합니다. 검색할 속성은 WS_LISTENER_PROPERTY_ID 입력 매개 변수로 식별됩니다.
WsGetMappedHeader 메시지에서 매핑된 헤더를 찾아 역직렬화합니다.
WsGetMessageProperty 지정된 Message 개체 속성을 검색합니다. 검색할 속성은 WS_MESSAGE_PROPERTY_ID 입력 매개 변수로 식별됩니다.
WsGetMetadataEndpoints 메타데이터 개체 문서에 정의된 "엔드포인트"를 반환합니다.
WsGetMetadataProperty 지정된 WS_METADATA 개체 속성을 검색합니다. 검색할 속성은 WS_METADATA_PROPERTY_ID 입력 매개 변수로 식별됩니다.
WsGetMissingMetadataDocumentAddress 이 함수는 메타데이터 개체에서 참조하는 누락된 문서의 주소를 반환합니다.
WsGetNamespaceFromPrefix 이 함수는 바인딩된 접두사에서 네임스페이스를 반환합니다.
WsGetOperationContextProperty 지정된 작업 컨텍스트의 속성을 반환합니다. 이러한 속성의 유효성은 작업 컨텍스트 자체의 수명으로 제한됩니다.
WsGetPolicyAlternativeCount 정책 개체에서 사용할 수 있는 대안 수를 검색합니다. 대체 개수를 사용하여 WsMatchPolicyAlternative를 사용하여 각 대안을 반복할 수 있습니다.

WsGetPolicyProperty 정책 개체의 속성을 검색합니다.
WsGetPrefixFromNamespace 이 함수는 네임스페이스가 바인딩된 접두사 값을 반환합니다. 범위에 두 개 이상의 접두사가 있을 수 있으며 이 함수는 해당 접두사 중 하나를 자유롭게 반환할 수 있습니다.
WsGetReaderNode 이 함수는 XML 판독기의 현재 위치에 있는 XML 노드를 반환합니다.
WsGetReaderPosition 판독기의 현재 위치를 반환합니다. XmlBuffer로 설정된 판독기에서만 사용할 수 있습니다.
WsGetReaderProperty 이 함수는 지정된 XML 판독기의 속성을 반환합니다.
WsGetSecurityContextProperty 지정된 보안 컨텍스트의 속성을 가져옵니다.
WsGetSecurityTokenProperty 보안 토큰에서 필드 또는 속성을 추출합니다.
WsGetServiceHostProperty 지정된 서비스 호스트 속성을 검색합니다. 검색할 속성은 WS_SERVICE_PROPERTY_ID 입력 매개 변수로 식별됩니다.
WsGetServiceProxyProperty 이 함수는 지정된 서비스 프록시 속성을 검색합니다. 검색할 속성은 WS_PROXY_PROPERTY_ID 입력 매개 변수로 식별됩니다.
WsGetWriterPosition 작성기의 현재 위치를 반환합니다. XmlBuffer로 설정된 작성기에서만 사용할 수 있습니다. 버퍼에 쓸 때 위치는 새 데이터가 배치되기 전의 xml 노드를 나타냅니다.
WsGetWriterProperty 지정된 XML 기록기 속성을 검색합니다. 검색할 속성은 WS_XML WRITER_PROPERTY_ID 입력 매개 변수로 식별됩니다.

WsGetXmlAttribute 지정된 localName을 사용하여 범위에서 가장 가까운 xml 특성을 찾아 해당 값을 반환합니다. 반환된 값은 지정된 힙에 배치됩니다.
WsInitializeMessage 이 함수는 처리를 준비하기 위해 메시지의 헤더를 초기화합니다.
WsMarkHeaderAsUnderstood 이 함수는 애플리케이션에서 헤더를 "이해"로 표시합니다.
WsMatchPolicyAlternative 정책 대안이 지정된 정책 제약 조건과 호환되는지 확인합니다.
WsMoveReader moveTo 매개 변수에 지정된 대로 판독기의 현재 위치를 이동합니다. 이 함수는 XmlBuffer로 설정된 판독기에서만 사용할 수 있습니다.
WsMoveWriter moveTo 매개 변수에 지정된 대로 작성기의 현재 위치를 이동합니다.
WsOffsetOf 구조체와 필드의 이름이 지정된 경우 구조체 내 필드의 오프셋(바이트)을 반환합니다.
WsOpenChannel 엔드포인트에 대한 채널을 엽니다.
WsOpenListener 지정된 주소에서 "수신 대기"를 시작합니다. 수신기가 열리면 채널을 수락할 수 있습니다. 열기에 성공하면 수신기 리소스를 해제하기 전에 WsCloseListener 함수를 사용하여 수신기를 닫아야 합니다.
WsOpenServiceHost 통신을 위한 서비스 호스트를 열고 모든 엔드포인트에서 수신기를 시작합니다. WsOpenSerivceHost가 호출될 때까지 클라이언트 애플리케이션은 서비스 엔드포인트에 연결할 수 없습니다.
WsOpenServiceProxy 서비스 엔드포인트에 대한 서비스 프록시를 엽니다.

WsPullBytes 요소 내에 쓸 바이트를 가져오기 위해 호출할 콜백을 설정합니다. 일부 인코딩에서는 데이터 복사본을 제거하여 더 효율적일 수 있습니다.
WsPushBytes 요소 내에서 바이트를 쓰기 위해 호출할 콜백을 설정합니다. 일부 인코딩에서는 데이터 복사본을 제거하여 더 효율적일 수 있습니다.
WsReadArray 판독기에서 일련의 요소를 읽고 지정된 값 형식에 따라 해당 콘텐츠를 해석합니다.
WsReadAttribute 지정된 WS_TYPE 값을 생성하는 특성을 읽습니다.
WsReadBody 메시지의 XML Readerof 값을 역직렬화하는 도우미 함수입니다. WS_MESSAGE_STATE WS_MESSAGE_STATE_READING 설정해야 합니다. 이 함수는 상태 전환을 일으키지 않습니다.
WsReadBytes Reader에서 텍스트를 읽고 base64 사양에 따라 문자를 바이트로 디코딩합니다.
WsReadChars 판독기에서 지정된 수의 텍스트 문자를 읽습니다.
WsReadCharsUtf8 판독기에서 지정된 수의 텍스트 문자를 읽고 UTF-8로 인코딩된 텍스트를 반환합니다.
WsReadElement 지정된 WS_TYPE 값을 생성하는 요소를 읽습니다.
WsReadEndAttribute 판독기를 읽은 특성이 포함된 요소 노드로 다시 이동합니다.
WsReadEndElement 이 함수는 현재 판독기 노드가 End 요소인지 확인하고 판독기를 다음 노드로 진행합니다.

WsReadEndpointAddressExtension WS_ENDPOINT_ADDRESS 확장을 읽습니다.
WsReadEnvelopeEnd 메시지의 닫는 요소를 읽습니다. 이 작업을 통해 채널 이외의 원본에서 메시지를 읽을 수 있습니다. 원본이 채널인 경우 WsReadMessageEnd를 사용합니다.
WsReadEnvelopeStart 메시지의 헤더를 읽고 본문 요소를 읽을 준비를 합니다.
WsReadMessageEnd 채널에서 메시지의 닫는 요소를 읽습니다.
WsReadMessageStart 채널에서 다음 메시지의 헤더를 읽고 본문 요소를 읽을 준비를 합니다.
WsReadMetadata 메타데이터 요소를 읽고 메타데이터 개체의 메타데이터 문서에 추가합니다.
WsReadNode 이 작업은 판독기를 입력 스트림의 다음 노드로 진행합니다.
WsReadQualifiedName 정규화된 이름을 읽고 XML_READER 현재 네임스페이스 범위에 따라 접두사, localName 및 네임스페이스로 구분합니다.
WsReadStartAttribute WsReadValue, WsReadChars 또는 WsReadBytes를 사용하여 콘텐츠를 읽을 수 있도록 판독기를 지정된 특성으로 이동합니다.
WsReadStartElement 이 함수를 호출하면 판독기에서 공백을 건너뛰는 시작 요소를 지나게 됩니다.
WsReadToStartElement 필요한 경우 공백 및 주석을 건너뛰는 다음 시작 요소로 판독기를 진행합니다. 필요에 따라 요소의 localName 및 네임스페이스를 확인할 수도 있습니다.

WsReadType WS_TYPE_MAPPING 따라 XML에서 지정된 WS_TYPE 값을 읽습니다.
WsReadValue 판독기에서 텍스트를 읽고 지정된 값 형식에 따라 구문 분석합니다.
WsReadXmlBuffer 판독기에서 WS_XML_BUFFER 현재 노드를 읽습니다.
WsReadXmlBufferFromBytes 판독기를 사용하여 인코딩된 바이트 집합을 WS_XML_BUFFER 변환합니다.
WsReceiveMessage 메시지를 받고 메시지 본문을 값으로 역직렬화합니다.
WsRegisterOperationForCancel 서비스 작업은 이 함수를 사용하여 취소 알림을 등록할 수 있습니다. 서비스 작업이 실행 중일 때만 이 API를 호출하는 것이 유효합니다. 서비스 작업이 완료된 후 호출하는 동작은 지원되지 않습니다.
WsRemoveCustomHeader 메시지에서 사용자 지정 헤더를 제거합니다. 이 함수는 메시지에 한 번 표시되고 최종 수신자를 대상으로 하는 헤더 유형을 처리하도록 설계되었습니다. 최종 수신기 이외의 역할을 대상으로 하는 헤더는 무시됩니다.
WsRemoveHeader 메시지에서 표준 WS_HEADER_TYPE 개체를 제거합니다.
WsRemoveMappedHeader 메시지에서 매핑된 헤더의 모든 인스턴스를 제거합니다.
WsRemoveNode xml 버퍼에서 지정된 위치에 있는 노드를 제거합니다. 요소에 배치하면 모든 자식 요소와 해당 끝 요소를 포함한 요소가 제거되고, 그렇지 않으면 단일 노드가 제거됩니다.
WsRequestReply 요청 메시지를 보내고 상관 관계가 있는 회신 메시지를 받는 데 사용됩니다.

WsRequestSecurityToken 페더레이션 시나리오에서 토큰 발급자 역할을 하는 STS(보안 토큰 서비스)에서 보안 토큰을 가져옵니다.
WsResetChannel 다시 사용할 수 있도록 채널을 다시 설정합니다.
WsResetError 오류 개체 매개 변수의 내용을 해제하지만 오류 개체 매개 변수에 할당된 리소스는 해제하지 않습니다.
WsResetHeap 모든 힙 할당을 해제합니다. WsAlloc을 사용하여 힙에 할당한 내용은 더 이상 유효하지 않습니다. 힙 개체 자체에 대한 할당은 해제되지 않습니다.
WsResetListener 다시 사용할 수 있도록 수신기 개체를 다시 설정합니다. 이 함수를 사용하려면 수신기 상태를 WS_LISTENER_STATE_CREATED 또는 WS_LISTENER_STATE_CLOSED 설정해야 합니다.
WsResetMessage 메시지 상태를 다시 WS_MESSAGE_STATE_EMPTY 설정합니다. 이 상태에서 Message 개체를 다시 사용할 수 있습니다.
WsResetMetadata 메타데이터 개체 상태를 WS_METADATA_STATE_CREATED 다시 설정합니다. 이 상태에서 메타데이터 개체를 다시 사용할 수 있습니다. 메타데이터 개체를 사용하여 검색된 WS_POLICY 개체가 해제됩니다.
WsResetServiceHost 다시 열 수 있도록 서비스 호스트를 다시 설정합니다.
WsResetServiceProxy 서비스 프록시를 다시 설정합니다.
WsRevokeSecurityContext 보안 컨텍스트를 취소합니다. 서버 쪽에서만 호출할 수 있습니다. 이 보안 컨텍스트를 사용하는 추가 요청이 실패하고 오류가 클라이언트로 전송됩니다.

WsSendFaultMessageForError WS_ERROR 개체가 지정된 경우 오류 메시지를 보냅니다.
WsSendMessage serialization을 사용하여 채널에 메시지를 보내 본문 요소를 작성합니다.
WsSendReplyMessage 받은 메시지에 대한 회신인 메시지를 보냅니다.
WsSetChannelProperty 채널의 속성을 설정합니다.
WsSetErrorProperty WS_ERROR 개체 속성을 설정합니다.
WsSetFaultErrorDetail WS_ERROR 개체에 저장된 오류 세부 정보를 작성합니다.
WsSetFaultErrorProperty WS_ERROR 개체의 Fault 속성을 설정합니다.
WsSetHeader 메시지에서 지정된 표준 헤더를 추가하거나 바꿉니다.
WsSetInput XML 판독기 인코딩 및 입력 소스를 설정합니다.
WsSetInputToBuffer 판독기 입력을 지정된 XML 버퍼로 설정합니다. WsSetInputToBuffer에 지정된 판독기 속성은 WsCreateReader에서 설정한 속성을 재정의합니다.
WsSetListenerProperty Listenerobject 속성을 설정합니다.
WsSetMessageProperty 이 작업은 Messageproperty를 설정합니다.

WsSetOutput 작성기에 대한 인코딩 및 출력 콜백을 설정합니다. 콜백은 작성기에 버퍼를 제공하고 비동기 i/o를 수행하는 데 사용됩니다.
WsSetOutputToBuffer 이 작업은 지정된 버퍼의 끝에 기록기를 배치합니다.
WsSetReaderPosition 판독기의 현재 위치를 설정합니다. WsGetReaderPosition 또는 WsGetWriterPosition 호출을 통해 위치를 가져와야 합니다. 이 함수는 WS_XML_BUFFER 설정된 판독기에서만 사용할 수 있습니다.
WsSetWriterPosition 작성기의 현재 위치를 설정합니다. WsGetReaderPosition 또는 WsGetWriterPosition 호출을 통해 위치를 가져와야 합니다.
WsShutdownSessionChannel 세션 채널에 대한 메시지의 끝을 알리는 데 사용됩니다.
WsSkipNode 입력 스트림에서 판독기를 앞으로 옵니다.
WsStartReaderCanonicalization 이 작업은 지정된 XML 판독기를 표준 또는 "정식" 형식으로 배치하는 프로세스를 시작합니다.
WsStartWriterCanonicalization 지정된 XML 기록기에서 정식화를 시작합니다.
WsTrimXmlWhitespace 문자 시퀀스에서 선행 및 후행 공백을 제거합니다.
WsVerifyXmlNCName 입력 문자열이 유효한 XML NCName인지 확인합니다.
WsWriteArray 이 작업은 XML 기록기에 일련의 요소를 보냅니다.

WsWriteAttribute 형식화된 값을 XML 특성으로 작성합니다.
WsWriteBody 메시지 본문에 값을 씁니다. 이 함수는 메시지의 XML 기록기에 값을 직렬화하는 도우미 함수입니다. 메시지 상태는 WS_MESSAGE_STATE_WRITING 설정해야 합니다. 이 함수는 상태 전환을 일으키지 않습니다.
WsWriteBytes 인코딩에 최적화된 형식으로 작성기에 바이트를 씁니다. 텍스트 인코딩을 작성할 때 base64로 인코딩된 바이트를 내보냅니다. 이진 형식으로 쓸 때 바이트를 직접 내보냅니다.
WsWriteChars 요소 또는 특성에 일련의 문자를 씁니다.
WsWriteCharsUtf8 요소 또는 특성에 UTF-8로 인코딩된 일련의 문자를 씁니다.
WsWriteElement 형식화된 값을 XML 요소로 작성합니다.
WsWriteEndAttribute 이 작업은 현재 요소에 특성 작성을 완료합니다. WsWriteStartAttribute를 호출하는 경우 기록기는 WsWriteEndAttribute가 호출될 때까지 다른 요소 또는 특성을 쓸 수 없습니다.
WsWriteEndCData 기록기에서 CDATA 섹션을 종료합니다.
WsWriteEndElement 작성기에 끝 요소를 씁니다.
WsWriteEndStartElement 작성기가 현재 요소를 커밋하고 추가 특성이 요소에 기록되지 않도록 합니다.
WsWriteEnvelopeEnd 메시지의 닫는 요소를 씁니다.

WsWriteEnvelopeStart 메시지의 현재 헤더 집합을 포함하여 메시지의 시작을 작성하고 본문 요소를 작성할 준비를 합니다.
WsWriteMessageEnd 메시지의 닫는 요소를 채널에 씁니다.
WsWriteMessageStart 메시지의 모든 헤더를 채널에 쓰고 본문 요소를 작성할 준비를 합니다.
WsWriteNode 지정된 노드를 XML 기록기에 씁니다.
WsWriteQualifiedName XML 정규화된 이름을 기록기에 씁니다.
WsWriteStartAttribute 이 작업은 현재 요소에 특성을 쓰기 시작합니다.
WsWriteStartCData 이 작업은 기록기에서 CDATA 섹션을 시작합니다.
WsWriteStartElement 시작 요소를 작성기에 씁니다.
WsWriteText 지정된 텍스트를 XML 작성기에 씁니다.
WsWriteType WS_TYPE_MAPPING 따라 지정된 WS_TYPE 값을 XML에 씁니다.
WsWriteValue 이 작업은 기본 값에 대한 최상의 표현을 기본 인코딩에서 파생하고 파생된 값을 Writer 개체에 전달합니다.
WsWriteXmlBuffer 작성기에 WS_XML_BUFFER 씁니다.

WsWriteXmlBufferToBytes 기록기를 사용하여 WS_XML_BUFFER 인코딩된 바이트 집합으로 변환합니다.
WsWriteXmlnsAttribute Xmlns 특성을 현재 요소에 씁니다.
WsXmlStringEquals 두 WS_XML_STRING 개체를 같은지 비교합니다. 이 작업은 String 개체에 포함된 문자 값의 서수 비교를 수행합니다.

인터페이스

IContentPrefetcherTaskTrigger IContentPrefetcherTaskTrigger 인터페이스는 설치된 앱 패키지가 이 백그라운드 작업에 대해 등록되었는지 확인하고 콘텐츠 프리페치 작업을 수동으로 트리거할 수 있는 메서드를 정의하여 콘텐츠 프리페치 동작 및 성능 테스트를 지원합니다.

구조체

WS_ANY_ATTRIBUTE 이 형식은 필드에 직접 매핑되지 않은 특성을 저장하는 데 사용됩니다.
WS_ANY_ATTRIBUTES 이 형식은 구조체의 필드에 직접 매핑되지 않은 특성 집합을 저장하는 데 사용됩니다.
WS_ASYNC_CONTEXT 비동기 모델과 함께 비동기 콜백 및 비동기 콜백에 전달될 포인터를 지정하는 데 사용됩니다.

WS_ASYNC_OPERATION WsAsyncExecute와 함께 일련의 비동기 작업에서 호출할 다음 함수를 지정하는 데 사용됩니다. (WS_ASYNC_OPERATION)
WS_ASYNC_STATE WsAsyncExecute에서 비동기 작업의 상태를 관리하는 데 사용됩니다.
WS_ATTRIBUTE_DESCRIPTION C 데이터 형식과 XML 특성 간의 매핑을 나타냅니다.
WS_BOOL_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정합니다. (WS_BOOL_DESCRIPTION)
WS_BUFFERS WS_BYTES 연속 배열을 나타내는 데 사용되는 구조체입니다.
WS_BYTE_ARRAY_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정합니다. (WS_BYTE_ARRAY_DESCRIPTION)
WS_BYTES 바이트 배열을 직렬화하고 역직렬화하는 데 사용됩니다.
WS_BYTES_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정합니다. (WS_BYTES_DESCRIPTION)
WS_CALL_PROPERTY 프록시 속성을 지정합니다. (WS_CALL_PROPERTY)
WS_CAPI_ASYMMETRIC_SECURITY_KEY_HANDLE CAPI 1.0 키 핸들로 비대칭 암호화 키를 지정하는 형식입니다.
WS_CERT_CREDENTIAL 모든 인증서 자격 증명 형식에 대한 추상 기본 형식입니다.

WS_CERT_ENDPOINT_IDENTITY 인증서 엔드포인트 ID 유형
WS_CERT_MESSAGE_SECURITY_BINDING_CONSTRAINT WS_XML_TOKEN_MESSAGE_SECURITY_BINDING 함께 사용할 수 있는 보안 바인딩 제약 조건입니다.
WS_CERT_SIGNED_SAML_AUTHENTICATOR 예상되는 발급자 인증서 배열을 기반으로 SAML 토큰 인증자를 지정하는 형식입니다.
WS_CERTIFICATE_VALIDATION_CALLBACK_CONTEXT WS_CERTIFICATE_VALIDATION_CALLBACK_CONTEXT 구조에는 HTTP 연결에 대한 인증서의 유효성을 검사하기 위한 콜백 함수 및 상태가 포함됩니다.
WS_CHANNEL_DECODER 수신된 메시지의 콘텐츠 형식 및 인코딩된 바이트를 변환할 수 있는 콜백 집합을 지정하는 데 사용되는 구조체입니다.
WS_CHANNEL_ENCODER 전송된 메시지의 콘텐츠 형식 및 인코딩된 바이트를 변환할 수 있는 콜백 집합을 지정하는 데 사용되는 구조체입니다.
WS_CHANNEL_PROPERTIES WS_CHANNEL_PROPERTY 구조체 집합을 지정합니다.
WS_CHANNEL_PROPERTY 채널별 설정을 지정합니다.
WS_CHANNEL_PROPERTY_CONSTRAINT 특정 채널 속성에 대한 제약 조건을 지정합니다.
WS_CHAR_ARRAY_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정합니다. (WS_CHAR_ARRAY_DESCRIPTION)
WS_CONTRACT_DESCRIPTION 서비스 모델에 대한 서비스 계약의 메타데이터입니다.

WS_CUSTOM_CERT_CREDENTIAL 애플리케이션에 대한 콜백에서 제공할 인증서 자격 증명을 지정하는 형식입니다.
WS_CUSTOM_CHANNEL_CALLBACKS 사용자 지정 채널의 구현을 형성하는 콜백 집합을 지정하는 데 사용되는 구조체입니다.
WS_CUSTOM_HTTP_PROXY WS_CHANNEL_PROPERTY_CUSTOM_HTTP_PROXY 사용하여 채널에 대한 사용자 지정 프록시를 지정하는 데 사용되는 구조체입니다.
WS_CUSTOM_LISTENER_CALLBACKS 사용자 지정 수신기의 구현을 형성하는 콜백 집합을 지정하는 데 사용되는 구조체입니다.
WS_CUSTOM_TYPE_DESCRIPTION C 데이터 형식과 XML 요소 간의 사용자 지정 매핑을 나타냅니다.
WS_DATETIME 이 구조는 날짜와 시간을 나타내는 데 사용됩니다.
WS_DATETIME_DESCRIPTION 이 형식 설명은 WS_DATETIME_TYPE 사용되며 선택 사항입니다. 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하는 데 사용됩니다.
WS_DECIMAL_DESCRIPTION WS_DECIMAL_TYPE 사용되는 선택적 형식 설명입니다. 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하는 데 사용됩니다.
WS_DEFAULT_VALUE 필드의 기본값을 정의합니다. WS_FIELD_DESCRIPTION 사용됩니다.
WS_DEFAULT_WINDOWS_INTEGRATED_AUTH_CREDENTIAL 현재 Windows ID를 기반으로 하는 Windows 통합 인증 자격 증명을 제공하는 형식입니다.

WS_DISALLOWED_USER_AGENT_SUBSTRINGS 차단된 UserAgent 하위 문자열의 목록을 지정합니다. WS_LISTENER_PROPERTY_DISALLOWED_USER_AGENT 수신기 속성과 함께 사용됩니다.
WS_DNS_ENDPOINT_IDENTITY DNS 이름으로 표시되는 엔드포인트 ID를 지정하기 위한 형식입니다.
WS_DOUBLE_DESCRIPTION WS_DOUBLE_TYPE 사용되는 선택적 형식 설명입니다. 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하는 데 사용됩니다.
WS_DURATION xsd:duration 데이터 형식을 나타냅니다.
WS_DURATION_DESCRIPTION WS_DURATION_TYPE 사용하는 선택적 형식 설명입니다. 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하는 데 사용됩니다.
WS_ELEMENT_DESCRIPTION C 데이터 형식과 XML 요소 간의 매핑을 나타냅니다.
WS_ENDPOINT_ADDRESS 엔드포인트의 네트워크 주소를 나타냅니다.
WS_ENDPOINT_ADDRESS_DESCRIPTION WS_ENDPOINT_ADDRESS XML 요소 간의 매핑에 대한 정보입니다.
WS_ENDPOINT_IDENTITY 모든 엔드포인트 ID의 기본 형식입니다.
WS_ENDPOINT_POLICY_EXTENSION 이 구조는 엔드포인트 정책 확장을 지정하는 데 사용됩니다.
WS_ENUM_DESCRIPTION WS_ENUM_TYPE 함께 사용되며 필요한 형식 설명입니다. 열거형의 값을 직렬화하고 역직렬화하는 데 사용되는 정보를 제공합니다.

WS_ENUM_VALUE 열거형(WS_ENUM_DESCRIPTION)의 일부인 단일 값에 대한 serialization 정보를 제공합니다.
WS_ERROR_PROPERTY 오류 특정 설정을 지정합니다.
WS_FAULT 오류는 처리 실패를 전달하는 메시지 본문에 전달되는 값입니다. 오류는 WS_FAULT 구조를 사용하여 모델링됩니다.
WS_FAULT_CODE 오류 코드를 나타냅니다.
WS_FAULT_DESCRIPTION WS_FAULT XML 요소 간의 매핑에 대한 정보입니다.
WS_FAULT_DETAIL_DESCRIPTION 오류 메시지의 세부 요소에 대한 설명입니다.
WS_FAULT_REASON 오류에 대한 설명을 포함합니다.
WS_FIELD_DESCRIPTION 구조체 내의 필드에 대한 serialization 정보를 나타냅니다.
WS_FLOAT_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하기 위해 WS_FLOAT_TYPE 사용하는 선택적 형식 설명입니다.
WS_GUID_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하기 위해 WS_GUID_TYPE 사용하는 선택적 형식 설명입니다.
WS_HEAP_PROPERTIES WS_HEAP_PROPERTYs 집합을 지정하는 데 사용되는 구조체입니다.

WS_HEAP_PROPERTY 힙 특정 설정을 지정합니다.
WS_HOST_NAMES 호스트 이름 목록을 포함하는 구조체입니다.
WS_HTTP_BINDING_TEMPLATE http 바인딩을 위해 애플리케이션에서 채울 HTTP 템플릿 구조입니다.
WS_HTTP_HEADER_AUTH_BINDING_TEMPLATE 애플리케이션에서 채울 HTTP 헤더 인증 보안 템플릿 정보입니다. WS_HTTP_HEADER_AUTH_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_HTTP_HEADER_AUTH_POLICY_DESCRIPTION http 채널 바인딩을 지정하는 정책을 설명합니다. (WS_HTTP_HEADER_AUTH_POLICY_DESCRIPTION)
WS_HTTP_HEADER_AUTH_SECURITY_BINDING 기본 다이제스트(RFC 2617) 및 SPNEGO(RFC4559) 프로토콜을 기반으로 대상 서비스 또는 HTTP 프록시 서버에 대한 HTTP 헤더 인증 사용을 지정하기 위한 보안 바인딩 하위 유형입니다.
WS_HTTP_HEADER_AUTH_SECURITY_BINDING_CONSTRAINT WS_HTTP_HEADER_AUTH_SECURITY_BINDING 해당하는 보안 바인딩 제약 조건입니다.
WS_HTTP_HEADER_AUTH_SECURITY_BINDING_POLICY_DESCRIPTION 이 형식 설명은 템플릿 API와 함께 입력 정책 설정에 따라 생성된 템플릿을 설명하는 데 사용됩니다. (WS_HTTP_HEADER_AUTH_SECURITY_BINDING_POLICY_DESCRIPTION)
WS_HTTP_HEADER_AUTH_SECURITY_BINDING_TEMPLATE HTP 헤더 인증 프로토콜 기반 보안 사용을 지정하기 위한 보안 바인딩 템플릿입니다.
WS_HTTP_HEADER_MAPPING WS_HTTP_MESSAGE_MAPPING 일부로 매핑되는 개별 헤더를 지정합니다.

WS_HTTP_MESSAGE_MAPPING 메시지 개체에 HTTP 요청 또는 응답을 나타내는 방법입니다.
WS_HTTP_POLICY_DESCRIPTION http 채널 바인딩을 지정하는 정책을 설명합니다. (WS_HTTP_POLICY_DESCRIPTION)
WS_HTTP_REDIRECT_CALLBACK_CONTEXT HTTP 자동 리디렉션 동작을 제어하기 위한 콜백 함수 및 상태를 지정합니다.
WS_HTTP_SSL_BINDING_TEMPLATE 애플리케이션에서 채울 SSL 보안 템플릿 정보입니다. WS_HTTP_SSL_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_HTTP_SSL_HEADER_AUTH_BINDING_TEMPLATE 애플리케이션에서 입력할 사용자 이름/암호 보안 템플릿 정보입니다. WS_HTTP_SSL_HEADER_AUTH_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_HTTP_SSL_HEADER_AUTH_POLICY_DESCRIPTION SSL 전송 보안 및 헤더 인증을 사용하여 http 채널 바인딩을 지정하는 정책을 설명합니다.
WS_HTTP_SSL_KERBEROS_APREQ_BINDING_TEMPLATE 애플리케이션에서 입력할 사용자 이름/암호 보안 템플릿 정보입니다. WS_HTTP_SSL_KERBEROS_APREQ_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_HTTP_SSL_KERBEROS_APREQ_POLICY_DESCRIPTION SSL 전송 보안 및 KERBEROS AP_REQ 메시지 보안을 사용하여 http 채널 바인딩을 지정하는 정책을 설명합니다.
WS_HTTP_SSL_KERBEROS_APREQ_SECURITY_CONTEXT_BINDING_TEMPLATE 애플리케이션에서 채울 보안 템플릿 정보입니다. WS_HTTP_SSL_KERBEROS_APREQ_SECURITY_CONTEXT_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_HTTP_SSL_KERBEROS_APREQ_SECURITY_CONTEXT_POLICY_DESCRIPTION SSL 전송 보안을 사용하여 http 채널 바인딩을 통해 보안 컨텍스트 메시지 바인딩을 지정하는 정책을 설명합니다. 부트스트랩 채널은 SSL 전송 보안 및 KERBEROS AP_REQ 메시지 보안이 포함된 http 채널 바인딩을 사용합니다.

WS_HTTP_SSL_POLICY_DESCRIPTION http 채널 바인딩을 지정하는 정책을 설명합니다. (WS_HTTP_SSL_POLICY_DESCRIPTION)
WS_HTTP_SSL_USERNAME_BINDING_TEMPLATE 애플리케이션에서 입력할 사용자 이름/암호 보안 템플릿 정보입니다. WS_HTTP_SSL_USERNAME_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_HTTP_SSL_USERNAME_POLICY_DESCRIPTION SSL 전송 보안 및 사용자 이름/암호 메시지 보안을 사용하여 http 채널 바인딩을 지정하는 정책을 설명합니다.
WS_HTTP_SSL_USERNAME_SECURITY_CONTEXT_BINDING_TEMPLATE 애플리케이션에서 입력할 보안 템플릿 정보입니다. WS_HTTP_SSL_USERNAME_SECURITY_CONTEXT_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_HTTP_SSL_USERNAME_SECURITY_CONTEXT_POLICY_DESCRIPTION SSL 전송 보안을 사용하여 http 채널 바인딩을 통한 보안 컨텍스트 메시지 바인딩을 지정하는 정책을 설명합니다. 부트스트랩 채널은 SSL 전송 보안 및 사용자 이름/암호 메시지 보안과 함께 http 채널 바인딩을 사용합니다.
WS_HTTP_URL HTTP URL을 지정하기 위한 URL 하위 형식입니다.
WS_HTTPS_URL HTTPS URL을 지정하기 위한 URL 하위 형식입니다.
WS_INT16_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하기 위해 WS_INT16_TYPE 사용되는 선택적 형식 설명입니다.
WS_INT32_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하기 위해 WS_INT32_TYPE 사용하는 선택적 형식 설명입니다.
WS_INT64_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하기 위해 WS_INT64_TYPE 사용되는 선택적 형식 설명입니다.

WS_INT8_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하기 위해 WS_INT8_TYPE 사용되는 선택적 형식 설명입니다.
WS_ISSUED_TOKEN_MESSAGE_SECURITY_BINDING_CONSTRAINT 발급 당사자로부터 발급된 토큰을 가져오는 방법에 대한 정보를 추출하는 데 사용할 수 있는 보안 바인딩 제약 조건입니다.
WS_ITEM_RANGE WS_FIELD_DESCRIPTION 내에서 WS_REPEATING_ELEMENT_FIELD_MAPPING, WS_REPEATING_ELEMENT_CHOICE_FIELD_MAPPING 또는 WS_REPEATING_ANY_ELEMENT_FIELD_MAPPING 사용할 때 나타날 수 있는 최소 및 최대 항목 수를 정의합니다.
WS_KERBEROS_APREQ_MESSAGE_SECURITY_BINDING WS-Security를 사용하여 Kerberos AP_REQ 티켓을 직접(즉, 세션을 설정하지 않고) 보안 토큰으로 사용하도록 지정하기 위한 보안 바인딩 하위 형식입니다.
WS_KERBEROS_APREQ_MESSAGE_SECURITY_BINDING_CONSTRAINT WS_KERBEROS_APREQ_MESSAGE_SECURITY_BINDING 해당하는 보안 바인딩 제약 조건입니다.
WS_KERBEROS_APREQ_MESSAGE_SECURITY_BINDING_POLICY_DESCRIPTION 이 형식 설명은 템플릿 API와 함께 입력 정책 설정에 따라 생성된 템플릿을 설명하는 데 사용됩니다. (WS_KERBEROS_APREQ_MESSAGE_SECURITY_BINDING_POLICY_DESCRIPTION)
WS_KERBEROS_APREQ_MESSAGE_SECURITY_BINDING_TEMPLATE WS-Security를 사용하여 Kerberos AP_REQ 티켓을 직접(즉, 세션을 설정하지 않고) 보안 토큰으로 사용하도록 지정하기 위한 보안 바인딩 템플릿입니다.
WS_LISTENER_PROPERTIES WS_LISTENER_PROPERTY 구조체 집합을 지정합니다.
WS_LISTENER_PROPERTY 수신기 특정 설정을 지정합니다.
WS_MESSAGE_DESCRIPTION 지정된 작업 설명에 대한 입력/출력 WS_MESSAGE 대한 스키마입니다.

WS_MESSAGE_PROPERTIES WS_MESSAGE_PROPERTY 구조체 집합을 지정합니다.
WS_MESSAGE_PROPERTY 메시지별 설정을 지정합니다.
WS_METADATA_ENDPOINT 메타데이터 문서에서 읽은 단일 엔드포인트에 대한 정보입니다.
WS_METADATA_ENDPOINTS 메타데이터 문서에서 읽은 모든 엔드포인트에 대한 정보입니다.
WS_METADATA_PROPERTY 메타데이터 개체 설정을 지정합니다.
WS_NAMEDPIPE_SSPI_TRANSPORT_SECURITY_BINDING 명명된 파이프 전송과 함께 Windows 통합 인증 프로토콜(예: Kerberos, NTLM 또는 SPNEGO)의 사용을 지정하기 위한 보안 바인딩 하위 유형입니다.
WS_NCRYPT_ASYMMETRIC_SECURITY_KEY_HANDLE 비대칭 암호화 키를 CryptoNG NCRYPT_KEY_HANDLE 지정하기 위한 형식입니다.
WS_NETPIPE_URL net.pipe URL을 지정하기 위한 URL 하위 형식입니다.
WS_NETTCP_URL net.tcp URL을 지정하기 위한 URL 하위 형식입니다.
WS_OPAQUE_WINDOWS_INTEGRATED_AUTH_CREDENTIAL SspiPromptForCredentials 및 관련 API 제품군에서 만든 불투명 핸들로 Windows 통합 인증 자격 증명을 제공하는 형식입니다. 이 기능은 Windows 7 이상에서만 사용할 수 있습니다.
WS_OPERATION_DESCRIPTION 서비스 작업에 대한 메타데이터입니다.

WS_PARAMETER_DESCRIPTION 들어오는/나가는 메시지 필드 설명에 있는 매개 변수의 인덱스입니다.
WS_POLICY_CONSTRAINTS 채널에 대한 정책 제약 조건을 지정합니다.
WS_POLICY_EXTENSION 모든 정책 확장 구조에 대한 기본 클래스입니다. 정책 확장은 사용자 지정 어설션과 같은 애플리케이션에서 직접 처리하는 어설션입니다.
WS_POLICY_PROPERTIES WS_POLICY_PROPERTY 구조체 집합을 지정합니다.
WS_POLICY_PROPERTY 정책 개체 설정을 지정합니다.
WS_PROXY_MESSAGE_CALLBACK_CONTEXT 입력 또는 출력 메시지의 헤더를 각각 연결하거나 검사하려는 애플리케이션의 콜백 함수 및 상태를 지정합니다.
WS_PROXY_PROPERTY 프록시 속성을 지정합니다. (WS_PROXY_PROPERTY)
WS_RAW_SYMMETRIC_SECURITY_KEY_HANDLE 대칭 암호화 키를 원시 바이트로 지정하기 위한 형식입니다.
WS_REQUEST_SECURITY_TOKEN_PROPERTY 발급자로부터 보안 토큰을 요청하기 위한 속성을 지정합니다.
WS_REQUEST_SECURITY_TOKEN_PROPERTY_CONSTRAINT 이 구조는 특정 요청 보안 토큰 속성에 대한 제약 조건 집합을 지정하는 데 사용됩니다. 지정되지 않은 속성 제약 조건은 기본 제약 조건을 사용합니다.
WS_RSA_ENDPOINT_IDENTITY RSA 엔드포인트 ID를 입력합니다.

WS_SAML_AUTHENTICATOR 들어오는 SAML 토큰의 유효성을 검사하기 위해 서버 쪽에서 사용되는 모든 SAML 인증자에 대한 추상 기본 형식입니다.
WS_SAML_MESSAGE_SECURITY_BINDING SAML 어설션을 메시지 보안 토큰으로 사용하도록 지정하기 위한 보안 바인딩 하위 형식입니다.
WS_SECURITY_ALGORITHM_PROPERTY 암호화 알고리즘 설정을 지정합니다.
WS_SECURITY_ALGORITHM_SUITE WS-Security와 함께 사용할 보안 알고리즘 및 키 길이를 정의합니다. 이 설정은 메시지 보안 바인딩 및 혼합 모드 보안 바인딩과 관련이 있습니다.
WS_SECURITY_BINDING 모든 보안 바인딩에 대한 추상 기본 형식입니다.
WS_SECURITY_BINDING_CONSTRAINT 모든 보안 바인딩 제약 조건 구조에 대한 기본 클래스입니다.
WS_SECURITY_BINDING_PROPERTIES 보안 바인딩 설정의 배열을 지정합니다.
WS_SECURITY_BINDING_PROPERTY 보안 바인딩 특정 설정을 지정합니다.
WS_SECURITY_BINDING_PROPERTY_CONSTRAINT 이 구조는 특정 보안 바인딩 속성에 대한 제약 조건 집합을 지정하는 데 사용됩니다. 지정되지 않은 속성 제약 조건은 기본 제약 조건을 사용합니다.
WS_SECURITY_CONSTRAINTS 이 구조는 보안 관련 제약 조건을 WS_POLICY_CONSTRAINTS 일부로 지정합니다.
WS_SECURITY_CONTEXT_MESSAGE_SECURITY_BINDING WS-SecureConversation을 사용하여 클라이언트와 서버 간에 협상된 보안 컨텍스트 토큰의 사용을 지정하기 위한 보안 바인딩 하위 형식입니다.

WS_SECURITY_CONTEXT_MESSAGE_SECURITY_BINDING_CONSTRAINT WS_SECURITY_CONTEXT_MESSAGE_SECURITY_BINDING 해당하는 보안 바인딩 제약 조건입니다.
WS_SECURITY_CONTEXT_MESSAGE_SECURITY_BINDING_POLICY_DESCRIPTION 이 형식 설명은 템플릿 API와 함께 입력 정책 설정에 따라 생성된 템플릿을 설명하는 데 사용됩니다. (WS_SECURITY_CONTEXT_MESSAGE_SECURITY_BINDING_POLICY_DESCRIPTION)
WS_SECURITY_CONTEXT_MESSAGE_SECURITY_BINDING_TEMPLATE 애플리케이션에서 제공하는 보안 컨텍스트 보안 바인딩의 사용을 지정하기 위한 보안 바인딩 템플릿입니다. (WS_SECURITY_CONTEXT_MESSAGE_SECURITY_BINDING_TEMPLATE)
WS_SECURITY_CONTEXT_PROPERTY WS_SECURITY_CONTEXT 속성을 정의합니다.
WS_SECURITY_CONTEXT_SECURITY_BINDING_POLICY_DESCRIPTION 이 형식 설명은 입력 정책 설정에 따라 생성된 보안 컨텍스트 관련 템플릿을 설명하기 위해 템플릿 API와 함께 사용됩니다.
WS_SECURITY_CONTEXT_SECURITY_BINDING_TEMPLATE 애플리케이션에서 제공하는 보안 컨텍스트 보안 바인딩의 사용을 지정하기 위한 보안 바인딩 템플릿입니다. (WS_SECURITY_CONTEXT_SECURITY_BINDING_TEMPLATE)
WS_SECURITY_DESCRIPTION 채널(클라이언트 쪽) 또는 수신기(서버 쪽)에 대한 보안 요구 사항을 지정하는 데 사용되는 최상위 구조입니다.
WS_SECURITY_KEY_HANDLE 암호화 키를 지정하는 모든 형식에 대한 추상 기본 형식입니다. 이러한 키는 일반적으로 일반 XML 보안 토큰 또는 사용자 지정 보안 토큰에 대해 지정됩니다.
WS_SECURITY_PROPERTIES 채널 전체 보안 설정의 배열을 지정합니다.
WS_SECURITY_PROPERTY 채널 전체 보안 설정을 지정합니다.

WS_SECURITY_PROPERTY_CONSTRAINT 이 구조는 특정 보안 속성에 대한 제약 조건 집합을 지정하는 데 사용됩니다. 지정되지 않은 속성 제약 조건은 기본 제약 조건을 사용합니다.
WS_SERVICE_CONTRACT 엔드포인트에서 서비스 계약을 지정합니다.
WS_SERVICE_ENDPOINT 서비스 호스트의 개별 엔드포인트를 나타냅니다. 엔드포인트의 속성은 주소, 바인딩 및 계약을 지정하는 데 사용됩니다.
WS_SERVICE_ENDPOINT_METADATA 엔드포인트의 포트 요소를 나타냅니다. 포트 요소는 serviceName에 지정된 대로 서비스 요소에 대해 생성되고 WS_SERVICE_HOST WS_SERVICE_PROPERTY_METADATA 속성에 대한 serviceNs가 생성됩니다.
WS_SERVICE_ENDPOINT_PROPERTY 서비스별 설정을 지정합니다. (WS_SERVICE_ENDPOINT_PROPERTY)
WS_SERVICE_METADATA 서비스 메타데이터 문서 배열을 지정합니다. WS_STRING 배열로 표시되는 WSDL/XSD 문서의 컬렉션일 수 있습니다.
WS_SERVICE_METADATA_DOCUMENT 서비스 메타데이터를 구성하는 개별 문서를 지정합니다.
WS_SERVICE_PROPERTY 서비스별 설정을 지정합니다. (WS_SERVICE_PROPERTY)
WS_SERVICE_PROPERTY_ACCEPT_CALLBACK 채널이 성공적으로 수락되면 호출되는 콜백을 지정합니다.
WS_SERVICE_PROPERTY_CLOSE_CALLBACK 채널을 닫을 때 호출되는 콜백을 지정합니다. 자세한 내용은 WS_SERVICE_CLOSE_CHANNEL_CALLBACK 참조하세요.

WS_SERVICE_SECURITY_IDENTITIES 확장된 보호의 유효성을 검사하는 데 사용되는 SPN(서버 보안 주체 이름) 목록입니다.
WS_SOAPUDP_URL soap.udp URL을 지정하기 위한 URL 하위 형식입니다.
WS_SPN_ENDPOINT_IDENTITY SPN(서비스 사용자 이름)으로 표시되는 엔드포인트 ID를 지정하기 위한 형식입니다.
WS_SSL_TRANSPORT_SECURITY_BINDING SSL/TLS 프로토콜 기반 전송 보안 사용을 지정하기 위한 보안 바인딩 하위 유형입니다.
WS_SSL_TRANSPORT_SECURITY_BINDING_CONSTRAINT WS_SSL_TRANSPORT_SECURITY_BINDING 해당하는 보안 바인딩 제약 조건입니다.
WS_SSL_TRANSPORT_SECURITY_BINDING_POLICY_DESCRIPTION 이 형식 설명은 템플릿 API와 함께 입력 정책 설정에 따라 생성된 템플릿을 설명하는 데 사용됩니다. (WS_SSL_TRANSPORT_SECURITY_BINDING_POLICY_DESCRIPTION)
WS_SSL_TRANSPORT_SECURITY_BINDING_TEMPLATE SSL/TLS 프로토콜 기반 전송 보안 사용을 지정하기 위한 보안 바인딩 템플릿입니다.
WS_SSPI_TRANSPORT_SECURITY_BINDING_POLICY_DESCRIPTION 이 형식 설명은 템플릿 API와 함께 입력 정책 설정에 따라 생성된 템플릿을 설명하는 데 사용됩니다. (WS_SSPI_TRANSPORT_SECURITY_BINDING_POLICY_DESCRIPTION)
WS_STRING 유니코드 문자 및 길이 배열입니다.
WS_STRING_DESCRIPTION 이 형식 설명은 WS_STRING_TYPE 사용되며 선택 사항입니다. 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하는 데 사용됩니다.

WS_STRING_USERNAME_CREDENTIAL 사용자 이름/암호 쌍을 문자열로 제공하는 형식입니다.
WS_STRING_WINDOWS_INTEGRATED_AUTH_CREDENTIAL Windows 자격 증명을 사용자 이름, 암호, 도메인 문자열로 제공하는 형식입니다.
WS_STRUCT_DESCRIPTION C 구조체 형식 및 XML 요소에 매핑하는 방법에 대한 정보입니다. WS_STRUCT_TYPE 사용됩니다.
WS_SUBJECT_NAME_CERT_CREDENTIAL 인증서의 주체 이름, 저장소 위치 및 저장소 이름을 사용하여 인증서 자격 증명을 지정하는 형식입니다. 포함된 채널 또는 수신기가 열릴 때 지정된 자격 증명이 로드됩니다.
WS_TCP_BINDING_TEMPLATE TCP 바인딩을 위해 애플리케이션에서 채울 TCP 템플릿 구조입니다.
WS_TCP_POLICY_DESCRIPTION http 채널 바인딩을 지정하는 정책을 설명합니다. (WS_TCP_POLICY_DESCRIPTION)
WS_TCP_SSPI_BINDING_TEMPLATE 애플리케이션에서 채울 HTTP 헤더 인증 보안 템플릿 정보입니다. WS_TCP_SSPI_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_TCP_SSPI_KERBEROS_APREQ_BINDING_TEMPLATE 애플리케이션에서 입력할 사용자 이름/암호 보안 템플릿 정보입니다. WS_TCP_SSPI_KERBEROS_APREQ_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_TCP_SSPI_KERBEROS_APREQ_POLICY_DESCRIPTION Windows SSPI 전송 보안 및 kerberos 메시지 보안을 사용하여 TCP 채널 바인딩을 지정하는 정책을 설명합니다.
WS_TCP_SSPI_KERBEROS_APREQ_SECURITY_CONTEXT_BINDING_TEMPLATE 애플리케이션에서 채울 보안 템플릿 정보입니다. WS_TCP_SSPI_KERBEROS_APREQ_SECURITY_CONTEXT_BINDING_TEMPLATE_TYPE 연결됩니다.

WS_TCP_SSPI_KERBEROS_APREQ_SECURITY_CONTEXT_POLICY_DESCRIPTION Windows SSPI 전송 보안과 함께 TCP 채널 바인딩을 사용하여 보안 컨텍스트 메시지 바인딩을 지정하는 정책을 설명합니다. 부트스트랩 채널은 Windows SSPI 전송 보안 및 kerberos 메시지 보안이 포함된 TCP 채널 바인딩을 사용합니다.
WS_TCP_SSPI_POLICY_DESCRIPTION Windows SSPI를 사용하여 TCP 채널 바인딩을 지정하는 정책을 설명합니다.
WS_TCP_SSPI_TRANSPORT_SECURITY_BINDING TCP 전송과 함께 Windows 통합 인증 프로토콜(예: Kerberos, NTLM 또는 SPNEGO)의 사용을 지정하기 위한 보안 바인딩 하위 유형입니다.
WS_TCP_SSPI_TRANSPORT_SECURITY_BINDING_CONSTRAINT WS_TCP_SSPI_TRANSPORT_SECURITY_BINDING 해당하는 보안 바인딩 제약 조건입니다.
WS_TCP_SSPI_TRANSPORT_SECURITY_BINDING_TEMPLATE Windows SSPI 프로토콜 기반 전송 보안 사용을 지정하기 위한 보안 바인딩 템플릿입니다.
WS_TCP_SSPI_USERNAME_BINDING_TEMPLATE 애플리케이션에서 입력할 사용자 이름/암호 보안 템플릿 정보입니다. WS_TCP_SSPI_USERNAME_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_TCP_SSPI_USERNAME_POLICY_DESCRIPTION Windows SSPI 전송 보안 및 사용자 이름/암호 메시지 보안을 사용하여 TCP 채널 바인딩을 지정하는 정책을 설명합니다.
WS_TCP_SSPI_USERNAME_SECURITY_CONTEXT_BINDING_TEMPLATE 애플리케이션에서 채울 보안 템플릿 정보입니다. WS_TCP_SSPI_USERNAME_SECURITY_CONTEXT_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_TCP_SSPI_USERNAME_SECURITY_CONTEXT_POLICY_DESCRIPTION Windows SSPI 전송 보안과 함께 TCP 채널 바인딩을 사용하여 보안 컨텍스트 메시지 바인딩을 지정하는 정책을 설명합니다. 부트스트랩 채널은 Windows SSPI 전송 보안 및 사용자 이름/암호 메시지 보안과 함께 TCP 채널 바인딩을 사용합니다.

WS_THUMBPRINT_CERT_CREDENTIAL 인증서의 지문, 저장 위치 및 저장소 이름을 사용하여 인증서 자격 증명을 지정하는 형식입니다. 포함된 채널 또는 수신기가 열릴 때 지정된 자격 증명이 로드됩니다.
WS_TIMESPAN 서명된 64비트 시간 간격을 100나노초 단위로 나타냅니다.
WS_TIMESPAN_DESCRIPTION 이 형식 설명은 WS_TIMESPAN_TYPE 사용되며 선택 사항입니다. 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하는 데 사용됩니다.
WS_UINT16_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하기 위해 WS_UINT16_TYPE 사용하는 선택적 형식 설명입니다.
WS_UINT32_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하기 위해 WS_UINT32_TYPE 사용하는 선택적 형식 설명입니다.
WS_UINT64_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하기 위해 WS_UINT64_TYPE 사용하는 선택적 형식 설명입니다.
WS_UINT8_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하기 위해 WS_UINT8_TYPE 사용하는 선택적 형식 설명입니다.
WS_UNION_DESCRIPTION 공용 구조체 유형 내의 선택 항목에 대한 정보입니다. 이는 WS_UNION_TYPE 함께 사용됩니다.
WS_UNION_FIELD_DESCRIPTION 공용 구조체 내의 필드에 대한 serialization 정보를 나타냅니다. WS_UNION_DESCRIPTION 참조하세요.
WS_UNIQUE_ID 고유 ID URI를 나타냅니다.

WS_UNIQUE_ID_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하기 위해 WS_UNIQUE_ID_TYPE 사용하는 선택적 형식 설명입니다.
WS_UNKNOWN_ENDPOINT_IDENTITY 알 수 없는 엔드포인트 ID를 입력합니다. 이 형식은 역직렬화되었지만 이해되지 않은 엔드포인트 ID 형식을 나타내는 데만 사용됩니다.
WS_UPN_ENDPOINT_IDENTITY UPN(사용자 계정 이름)으로 표시되는 엔드포인트 ID를 지정하기 위한 형식입니다.
WS_URL WsDecodeUrl 및 WsEncodeUrl API와 함께 사용되는 모든 URL 구성표에 대한 추상 기본 형식입니다.
WS_USERNAME_CREDENTIAL 모든 사용자 이름/암호 자격 증명에 대한 추상 기본 형식입니다.
WS_USERNAME_MESSAGE_SECURITY_BINDING 애플리케이션 제공 사용자 이름/암호 쌍을 직접(즉, 원샷) 보안 토큰으로 사용하도록 지정하기 위한 보안 바인딩 하위 유형입니다.
WS_USERNAME_MESSAGE_SECURITY_BINDING_CONSTRAINT WS_USERNAME_MESSAGE_SECURITY_BINDING 해당하는 보안 바인딩 제약 조건입니다.
WS_USERNAME_MESSAGE_SECURITY_BINDING_POLICY_DESCRIPTION 이 형식 설명은 템플릿 API와 함께 입력 정책 설정에 따라 생성된 템플릿을 설명하는 데 사용됩니다. (WS_USERNAME_MESSAGE_SECURITY_BINDING_POLICY_DESCRIPTION)
WS_USERNAME_MESSAGE_SECURITY_BINDING_TEMPLATE 애플리케이션 제공 사용자 이름/암호 쌍을 직접(즉, 원샷) 보안 토큰으로 사용하도록 지정하기 위한 보안 바인딩 템플릿입니다.
WS_UTF8_ARRAY_DESCRIPTION 이 형식 설명은 WS_UTF8_ARRAY_TYPE 사용되며 선택 사항입니다. 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하는 데 사용됩니다.

WS_VOID_DESCRIPTION 직렬화되거나 역직렬화되지 않는 필드에 대한 정보를 지정합니다.
WS_WINDOWS_INTEGRATED_AUTH_CREDENTIAL Windows 통합 인증에 사용되는 모든 자격 증명 형식에 대한 추상 기본 형식입니다.
WS_WSZ_DESCRIPTION 이 형식 설명은 WS_WSZ_TYPE 사용되며 선택 사항입니다. 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하는 데 사용됩니다.
WS_XML_ATTRIBUTE 특성을 나타냅니다(예 <: a:purchaseOrder xmlns:a="http://tempuri.org" id="5">)
WS_XML_BASE64_TEXT base64로 인코딩된 데이터를 나타냅니다.
WS_XML_BOOL_TEXT "true" 또는 "false" 텍스트를 나타내는 부울 값입니다.
WS_XML_BUFFER_PROPERTY XML 버퍼 특정 설정을 지정합니다.
WS_XML_CANONICALIZATION_INCLUSIVE_PREFIXES 배타적 XML 정식화 중에 포괄 접두사로 처리되어야 하는 XML 접두사 배열입니다. 포괄 접두사 처리는 RFC 3741에 정의되어 있습니다.
WS_XML_CANONICALIZATION_PROPERTY XML 정식화가 수행되는 방법을 제어하는 설정을 지정합니다.
WS_XML_COMMENT_NODE 주석을 나타냅니다.
WS_XML_DATETIME_TEXT xsd:dateTime으로 형식이 지정된 날짜/시간을 나타냅니다.

WS_XML_DECIMAL_TEXT 12 바이트 고정 소수점 값을 나타냅니다.
WS_XML_DICTIONARY 고유한 문자열 집합을 나타냅니다. 이 정보는 이진 인코딩에서 더 압축된 xml 문서를 작성하는 데 사용됩니다.
WS_XML_DOUBLE_TEXT 8 바이트 부동 소수점 값을 나타냅니다.
WS_XML_ELEMENT_NODE xml의 시작 요소를 나타냅니다(예:
WS_XML_FLOAT_TEXT 4 바이트 부동 소수점 값을 나타냅니다.
WS_XML_GUID_TEXT "xxxxxxxx-xxxx-xx" 텍스트로 서식이 지정된 GUID를 나타냅니다.
WS_XML_INT32_TEXT 부 서명된 32비트 정수입니다.
WS_XML_INT64_TEXT 부 서명된 64비트 정수입니다.
WS_XML_LIST_TEXT 단일 공백 문자로 구분된 텍스트 값 목록을 나타냅니다.
WS_XML_NODE 모든 종류의 XML 노드에 대한 기본 형식입니다. XML 노드는 XML의 데이터 단위입니다.
WS_XML_NODE_POSITION XML 버퍼 내의 위치를 나타냅니다.

WS_XML_QNAME XML 이름(요소 또는 특성)을 로컬 이름, 네임스페이스 쌍으로 지정하는 데 사용되는 구조체입니다.
WS_XML_QNAME_DESCRIPTION 이 형식 설명은 WS_XML_QNAME_TYPE 사용되며 선택 사항입니다. 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하는 데 사용됩니다.
WS_XML_QNAME_TEXT "prefix:localName" 텍스트로 서식이 지정된 qname을 나타냅니다.
WS_XML_READER_BINARY_ENCODING 판독기에서 읽는 바이트를 이진 xml로 해석해야 함을 나타내는 데 사용됩니다.
WS_XML_READER_BUFFER_INPUT xml 입력의 원본이 버퍼임을 지정합니다.
WS_XML_READER_ENCODING 이 구조체는 다양한 종류의 판독기 인코딩에 대한 기본 형식입니다.
WS_XML_READER_INPUT 판독기에서 xml 문서를 구성하는 바이트를 가져올 위치를 지정합니다.
WS_XML_READER_MTOM_ENCODING 판독기에서 읽는 바이트를 MTOM 형식으로 해석해야 함을 나타내는 데 사용됩니다.
WS_XML_READER_PROPERTIES WS_XML_READER_PROPERTYs 집합을 지정하는 데 사용되는 구조체입니다.
WS_XML_READER_PROPERTY 판독기별 설정을 지정합니다.
WS_XML_READER_RAW_ENCODING 판독기에서 문서의 바이트를 base64로 인코딩된 문자로 표시해야 함을 나타내는 데 사용됩니다.

WS_XML_READER_STREAM_INPUT 콜백에서 xml의 원본을 가져오라고 지정합니다.	
WS_XML_READER_TEXT_ENCODING 판독기에서 읽는 바이트를 텍스트 xml로 해석해야 함을 나타내는 데 사용됩니다.	
WS_XML_SECURITY_TOKEN_PROPERTY XML 보안 토큰의 속성을 지정합니다.	
WS_XML_STRING 선택적으로 연결된 사전 정보가 있는 문자열을 나타냅니다. xml API는 WS_XML_STRINGs 사용하여 접두사, localNames 및 네임스페이스를 식별합니다.	
WS_XML_STRING_DESCRIPTION 이 형식 설명은 WS_XML_STRING_TYPE 사용되며 선택 사항입니다. 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하는 데 사용됩니다.	
WS_XML_TEXT xml 텍스트 콘텐츠의 노드를 나타냅니다.	
WS_XML_TEXT_NODE 요소, 특성 또는 CDATA 콘텐츠를 나타냅니다.	
WS_XML_TIMESPAN_TEXT "[+ 텍스트로 서식이 지정된 시간 범위를 나타냅니다.	-][d?.]HH:mm:ss[.fffffff]" d는 요일을 나타내는 일련의 숫자입니다.
WS_XML_TOKEN_MESSAGE_SECURITY_BINDING XML 형식으로 애플리케이션에 이미 사용할 수 있는 보안 토큰의 사용을 지정하기 위한 보안 바인딩 하위 형식입니다.	
WS_XML_UINT64_TEXT 부호 없는 64비트 정수입니다.	
WS_XML_UNIQUE_ID_TEXT "urn:uuid:xxxxxxxx-xxxx-xxxx-xxxx-xx" 텍스트로 서식이 지정된 GUID를 나타냅니다.	

WS_XML_UTF16_TEXT UTF-16 바이트로 인코딩된 텍스트를 나타냅니다.
WS_XML_UTF8_TEXT UTF-8 바이트로 인코딩된 텍스트를 나타냅니다.
WS_XML_WRITER_BINARY_ENCODING 작성기가 바이트를 이진 xml로 내보내야 함을 나타내는 데 사용됩니다.
WS_XML_WRITER_BUFFER_OUTPUT 생성된 바이트를 버퍼에 배치되도록 지정합니다.
WS_XML_WRITER_ENCODING 이 구조체는 다양한 종류의 기록기 인코딩에 대한 기본 형식입니다.
WS_XML_WRITER_MTOM_ENCODING 판독기에서 MTOM 형식으로 바이트를 내보내야 함을 나타내는 데 사용됩니다. MTOM 형식은 포함된 base64로 인코딩된 텍스트가 아닌 이진 MIME 부분으로 작성된 바이트를 나타냅니다.
WS_XML_WRITER_OUTPUT 작성기에서 xml 문서를 구성하는 바이트를 내보낸 위치를 지정합니다.
WS_XML_WRITER_PROPERTIES WS_XML_WRITER_PROPERTYs 집합을 지정하는 데 사용되는 구조체입니다.
WS_XML_WRITER_PROPERTY 작성기별 설정을 지정합니다.
WS_XML_WRITER_RAW_ENCODING 기록기가 디코딩된 base64 문자에서 바이트를 내보내야 함을 나타내는 데 사용됩니다.
WS_XML_WRITER_STREAM_OUTPUT 생성된 바이트를 콜백으로 보내도록 지정합니다.

WS_XML_WRITER_TEXT_ENCODING 판독기에서 바이트를 텍스트 xml로 내보내야 임을 나타냅니다.

winlocation_com_ref

아티클2023. 03. 09.

Winlocation_Com_Ref 기술에 대한 개요입니다.

Winlocation_Com_Ref 기술은 헤더와 연결되지 않습니다.

winsensors_com_ref

아티클2023. 03. 09.

Winsensors_Com_Ref 기술 개요입니다.

Winsensors_Com_Ref 기술은 헤더와 연결되지 않습니다.

wlbsprov

아티클2023. 03. 09.

네트워크 부하 분산 공급자 기술 개요입니다.

네트워크 부하 분산 공급자 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 네트워크 부하 분산 공급자

NFS용 WMI 공급자

아티클2023. 08. 22.

NFS용 WMI 공급자 기술 개요입니다.

NFS용 WMI 공급자 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- NFS용 WMI 공급자

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

WPD 애플리케이션 프로그래밍 인터페이스

아티클2023. 03. 16.

WPD 애플리케이션 프로그래밍 인터페이스 기술 개요입니다.

WPD 애플리케이션 프로그래밍 인터페이스를 개발하려면 다음 헤더가 필요합니다.

- portabledeviceapi.h
- portabledeviceconnectapi.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- WPD 애플리케이션 프로그래밍 인터페이스

함수

조언 Advise 메서드는 디바이스 이벤트를 수신하는 애플리케이션 정의 콜백을 등록합니다.
조언 서비스 이벤트를 수신하는 애플리케이션 정의 콜백 개체를 등록합니다.
취소 Cancel 메서드는 보류 중인 작업을 취소합니다. (IEnumPortableDeviceObjectIDs.Cancel)
취소 Cancel 메서드는 이 인터페이스에서 보류 중인 작업을 취소합니다.
취소 Cancel 메서드는 이 인터페이스에서 보류 중인 요청을 취소합니다.
취소 Cancel 메서드는 이 인터페이스에서 호출되는 보류 중인 작업을 취소합니다.
취소 Cancel 메서드는 이 인터페이스에서 진행 중인 호출을 취소합니다.

취소 Cancel 메서드는 보류 중인 호출을 취소합니다.
취소 Cancel 메서드는 보류 중인 속성 요청을 취소합니다.
취소 Cancel 메서드는 보류 중인 작업을 취소합니다. (IPortableDeviceResources.Cancel)
취소 이 인터페이스에서 보류 중인 작업을 취소합니다.
취소 보류 중인 작업을 취소합니다.
취소 보류 중인 메서드 호출을 취소합니다.
취소 MTP/Bluetooth 디바이스에 연결하거나 연결을 끊기 위해 보류 중인 요청을 취소합니다.
Capabilities Capabilities 메서드는 휴대용 디바이스의 기능을 쿼리하는 데 사용되는 인터페이스를 검색합니다.
Capabilities 서비스 기능을 검색합니다.
복제 Clone 메서드는 현재 IEnumPortableDeviceObjectIDs 인터페이스를 복제합니다.
닫기 Close 메서드는 디바이스와의 연결을 닫습니다.
닫기 서비스에 대한 연결을 해제합니다.

연결 MTP/Bluetooth 디바이스에 비동기 연결 요청을 보냅니다.
콘텐츠 Content 메서드는 디바이스의 개체에 액세스하는 데 사용할 수 있는 인터페이스를 검색합니다.
콘텐츠 서비스 콘텐츠에 대한 액세스를 검색합니다.
복사 Copy 메서드는 디바이스의 한 위치에서 다른 위치로 개체를 복사합니다.
CreateObjectWithPropertiesAndData CreateObjectWithPropertiesAndData 메서드는 디바이스에서 속성과 데이터를 모두 사용하여 개체를 만듭니다.
CreateObjectWithPropertiesOnly CreateObjectWithPropertiesOnly 메서드는 디바이스의 속성만 있는 개체를 만듭니다.
CreateResource CreateResource 메서드는 리소스를 만듭니다.
삭제 Delete 메서드는 디바이스에서 하나 이상의 개체를 삭제합니다.
삭제 Delete 메서드는 디바이스의 지정된 개체에서 지정된 속성을 삭제합니다.
삭제 Delete 메서드는 pszObjectID 매개 변수로 식별된 개체에서 하나 이상의 리소스를 삭제합니다.
연결 끊기 MTP/Bluetooth 디바이스에 비동기 연결 끊기 요청을 보냅니다.
EnumObjects EnumObjects 메서드는 개체의 직계 자식 개체를 열거하는 데 사용되는 인터페이스를 검색합니다. 특정 속성을 사용하여 개체를 열거할 수 있는 선택적 필터가 있습니다.

GetCommandOptions GetCommandOptions 메서드는 디바이스에서 지정된 명령에 대해 지원되는 모든 옵션을 검색합니다.
GetCommandOptions WPD 명령의 옵션을 검색합니다.
GetDeviceDescription 디바이스에 대한 설명을 검색합니다.
GetDeviceForService 지정된 서비스와 연결된 디바이스를 검색합니다.
GetDeviceFriendlyName 디바이스의 사용자에게 친숙한 이름을 검색합니다.
GetDeviceManufacturer 디바이스 제조업체의 이름을 검색합니다.
GetDeviceProperty 컴퓨터의 디바이스에 의해 저장된 속성 값을 검색합니다. (이러한 속성은 Windows 이식 가능 디바이스에서 정의한 표준 속성이 아닙니다.)
GetDevices 컴퓨터에 연결된 휴대용 디바이스 목록을 검색합니다.
GetDeviceServices 지정된 디바이스와 연결된 서비스 목록을 검색합니다.
GetEventAttributes 이벤트의 특성을 검색합니다.
GetEventOptions GetEventOptions 메서드는 디바이스에서 지정된 이벤트에 대해 지원되는 모든 옵션을 검색합니다.
GetEventParameterAttributes 이벤트 매개 변수의 특성을 검색합니다.

GetFixedPropertyAttributes GetFixedPropertyAttributes 메서드는 지정된 속성 및 형식에 대한 표준 속성 특성을 검색합니다.
GetFormatAttributes 형식의 특성을 검색합니다.
GetFormatPropertyAttributes 형식 속성의 특성을 검색합니다.
GetFormatRenderingProfiles 형식의 렌더링 프로필을 검색합니다.
GetFunctionalCategories GetFunctionalCategories 메서드는 디바이스에서 지원하는 모든 기능 범주를 검색합니다.
GetFunctionalObjects GetFunctionalObjects 메서드는 디바이스에서 지정된 범주와 일치하는 모든 기능 개체를 검색합니다.
GetInheritedServices 지정된 상속 형식이 있는 서비스를 검색합니다.
GetMethodAttributes 지정된 메서드를 설명하는 데 사용되는 특성을 검색합니다.
GetMethodParameterAttributes 지정된 메서드 매개 변수를 설명하는 데 사용되는 특성을 검색합니다.
GetObjectID GetObjectID 메서드는 디바이스에 기록된 리소스의 개체 ID를 검색합니다. 이 메서드는 데이터 스트림에서 IStream::Commit을 호출한 후에만 유효합니다.
GetObjectIDsFromPersistentUniqueIDs GetObjectIDsFromPersistentUniqueIDs 메서드는 PUID(영구 고유 ID)가 지정된 경우 하나 이상의 개체의 현재 개체 ID를 검색합니다.

GetPnPDeviceID GetPnPDeviceID 메서드는 애플리케이션이 디바이스를 여는 데 사용한 PnP(플러그 앤 플레이) 디바이스 식별자를 검색합니다.
GetPnPID 커넥터의 PnP(플러그 앤 플레이) 디바이스 식별자를 검색합니다.
GetPnPServiceID 서비스의 PnP(플러그 앤 플레이) 식별자를 검색합니다.
GetPrivateDevices GetPrivateDevices 메서드는 컴퓨터에 연결된 개인 휴대용 디바이스 목록을 검색합니다. 이러한 개인 디바이스는 이러한 특정 디바이스용으로 설계된 애플리케이션을 통해서만 액세스할 수 있습니다.
Getproperty 지정된 MTP/Bluetooth Bus 열거자 디바이스에 대한 속성을 검색합니다.
GetPropertyAttributes GetPropertyAttributes 메서드는 디바이스에서 지정된 개체 속성의 특성을 검색합니다.
GetResourceAttributes GetResourceAttributes 메서드는 개체의 지정된 리소스에서 모든 특성을 검색합니다.
GetServiceObjectID 서비스의 개체 식별자를 검색합니다. 예를 들어 이 개체 식별자를 사용하여 서비스의 속성에 액세스할 수 있습니다.
GetStream GetStream 메서드는 디바이스의 개체에서 콘텐츠 데이터를 읽거나 쓸 수 있는 IStream 인터페이스를 가져옵니다. 검색된 인터페이스를 사용하면 개체 데이터를 읽거나 쓸 수 있습니다.
GetSupportedCommands GetSupportedCommands 메서드는 이 디바이스에 대해 지원되는 모든 명령 목록을 검색합니다.
GetSupportedCommands 서비스에서 지원하는 명령을 검색합니다.

GetSupportedContentTypes GetSupportedContentTypes 메서드는 디바이스에서 지정된 기능 개체 형식에 대해 지원되는 모든 콘텐츠 형식을 검색합니다.
GetSupportedEvents GetSupportedEvents 메서드는 이 디바이스에 대해 지원되는 이벤트를 검색합니다.
GetSupportedEvents 서비스에서 지원하는 이벤트를 검색합니다.
GetSupportedFormatProperties GetSupportedFormatProperties 메서드는 디바이스에서 지정된 형식의 개체에서 지원하는 속성을 검색합니다.
GetSupportedFormatProperties 지정된 형식에 대해 서비스에서 지원하는 속성을 검색합니다.
GetSupportedFormats GetSupportedFormats 메서드는 디바이스에서 지정된 개체 형식에 대해 지원되는 형식을 검색합니다. 예를 들어 오디오 개체를 지정하면 WPD_OBJECT_FORMAT_WMA, WPD_OBJECT_FORMAT_WAV 및 WPD_OBJECT_FORMAT_MP3 반환될 수 있습니다.
GetSupportedFormats 서비스에서 지원하는 형식을 검색합니다.
GetSupportedMethods 서비스에서 지원하는 메서드를 검색합니다.
GetSupportedMethodsByFormat 지정된 형식에 대해 서비스에서 지원하는 메서드를 검색합니다.
GetSupportedProperties GetSupportedProperties 메서드는 지정된 개체가 지원하는 속성 목록을 검색합니다. 이러한 속성 중 일부가 실제로 값을 가질 수 있는 것은 아닙니다.
GetSupportedResources GetSupportedResources 메서드는 특정 개체에서 지원하는 리소스 목록을 검색합니다.

GetValues GetValues 메서드는 디바이스의 지정된 개체에서 지정된 속성 목록을 검색합니다.
Invoke 메서드를 동기적으로 호출합니다.
InvokeAsync 메서드를 비동기적으로 호출합니다.
메서드 서비스에서 사용자 지정 기능을 호출하는 데 사용되는 IPortableDeviceServiceMethods 인터페이스를 검색합니다.
이동 Move 메서드는 디바이스의 한 위치에서 다른 위치로 하나 이상의 개체를 이동합니다.
다음 Next 메서드는 열거형 시퀀스에서 하나 이상의 개체 ID를 검색합니다.
OnComplete 콜백 메서드가 실행을 완료했음을 나타냅니다.
OnEnd OnEnd 메서드는 IPortableDevicePropertiesBulk::Start에서 시작된 대량 작업이 완료되면 SDK에서 호출됩니다.
Onevent OnEvent 메서드는 비동기 이벤트에 대해 애플리케이션에 알리기 위해 SDK에서 호출됩니다.
OnProgress OnProgress 메서드는 IPortableDevicePropertiesBulk::Start에서 시작된 대량 작업이 디바이스에 데이터를 전송하고 일부 정보를 다시 받은 경우 SDK에서 호출됩니다.
OnStart OnStart 메서드는 IPortableDevicePropertiesBulk::Start가 시작하려고 할 때 SDK에서 호출됩니다.

열기 Open 메서드는 애플리케이션과 디바이스 간의 연결을 엽니다.
열기 서비스에 대한 연결을 엽니다.
속성 Properties 메서드는 디바이스의 개체에서 속성을 얻거나 설정하는 데 필요한 인터페이스를 검색합니다.
QueueGetValuesByObjectFormat QueueGetValuesByObjectFormat 인터페이스는 디바이스에서 특정 형식의 개체 속성에 대한 요청을 큐에 대기합니다.
QueueGetValuesByObjectList QueueGetValuesByObjectList 메서드는 디바이스에 있는 하나 이상의 지정된 개체에서 하나 이상의 지정된 속성에 대한 요청을 큐에 대기합니다.
QueueSetValuesByObjectList QueueSetValuesByObjectList 메서드는 디바이스에서 하나 이상의 지정된 개체에 대해 하나 이상의 지정된 값을 설정하라는 요청을 큐에 대기합니다.
RefreshDeviceList RefreshDeviceList 메서드는 컴퓨터에 연결된 디바이스 목록을 새로 고칩니다.
재설정 Reset 메서드는 열거형 시퀀스를 시작 부분으로 다시 설정합니다. (IEnumPortableDeviceObjectIDs.Reset)
SeekInUnits SeekInUnits 메서드는 대체 단위에 따라 스트림에서 검색을 수행합니다.
SendCommand SendCommand 메서드는 디바이스에 명령을 보내고 결과를 동기적으로 검색합니다.
SendCommand 표준 WPD 명령 및 해당 매개 변수를 서비스에 보냅니다.

SetProperty MTP/Bluetooth Bus 열거자 디바이스에서 지정된 속성을 설정합니다.
SetValues SetValues 메서드는 디바이스의 지정된 개체에서 하나 이상의 속성을 추가하거나 수정합니다.
skip Skip 메서드는 열거형 시퀀스에서 지정된 수의 개체를 건너뜁니다.
시작 Start 메서드는 큐에 대기된 작업을 시작합니다.
전달 Transfer 메서드는 기존 개체 리소스의 콘텐츠 데이터를 읽거나 쓰는 데 사용되는 인터페이스를 검색합니다.
Unadvise Unadvise 메서드는 클라이언트가 콜백 알림을 받지 못하도록 등록을 취소합니다. 이전에 Advise를 호출한 경우 이 메서드를 호출해야 합니다.
Unadvise 서비스 이벤트 콜백 개체의 등록을 취소합니다.
UpdateObjectWithPropertiesAndData 디바이스에 있는 속성 및 데이터를 사용하여 개체를 업데이트.

인터페이스

IEnumPortableDeviceObjectIDs IEnumPortableDeviceObjectIDs 인터페이스는 이식 가능한 디바이스의 개체를 열거합니다. 디바이스에서 IPortableDeviceContent::EnumObjects를 호출하여 처음에 이 인터페이스를 가져옵니다.
IPortableDevice IPortableDevice 인터페이스는 휴대용 디바이스에 대한 액세스를 제공합니다.

IPortableDeviceCapabilities IPortableDeviceCapabilities는 지원되는 형식, 명령 및 기능 개체를 포함하여 다양한 디바이스 기능을 인터페이스합니다. IPortableDevice::Capabilities를 호출하여 디바이스에서 이 인터페이스를 검색할 수 있습니다.
IPortableDeviceConnector 쌍을 이루는 MTP/Bluetooth 디바이스에 대한 연결 관리 및 속성 검색에 사용되는 메서드를 정의합니다.
IPortableDeviceContent IPortableDeviceContent 인터페이스는 디바이스에서 콘텐츠를 만들고, 열거하고, 검사하고, 삭제하는 메서드를 제공합니다. 이 인터페이스를 얻으려면 IPortableDevice::Content를 호출합니다.
IPortableDeviceContent2 디바이스에서 찾은 콘텐츠에 대한 액세스를 제공하는 추가 메서드를 정의합니다.
IPortableDeviceDataStream IPortableDeviceDataStream 인터페이스는 데이터 전송에 사용되는 IStream에 추가 메서드를 노출합니다.
IPortableDeviceEventCallback 애플리케이션이 IPortableDevice::Advise를 호출하여 수신하도록 등록한 경우 비동기 콜백을 수신하도록 애플리케이션에서 구현한 IPortableDeviceEventCallback 인터페이스입니다.
IPortableDeviceManager 컴퓨터에 연결된 디바이스를 열거하고 제조업체, 식별 이름 및 설명을 포함하여 설치 정보를 요청하는 간단한 방법을 제공합니다.
IPortableDeviceProperties IPortableDeviceProperties 인터페이스는 디바이스 또는 디바이스 자체의 개체에서 속성을 검색, 추가 또는 삭제합니다.
IPortableDevicePropertiesBulk IPortableDevicePropertiesBulk 인터페이스는 디바이스의 여러 개체에서 비동기적으로 여러 속성을 쿼리하거나 설정합니다.

IPortableDevicePropertiesBulkCallback IPortableDevicePropertiesBulkCallback 인터페이스는 IPortableDevicePropertiesBulk 인터페이스를 사용하여 시작된 비동기 작업의 진행률을 추적하기 위해 애플리케이션에 의해 구현됩니다. 애플리케이션이 IPortableDevicePropertiesBulk::Start를 호출한 후 Windows 이식 가능 디바이스는 IPortableDevicePropertiesBulkCallback::OnStart를 먼저 호출한 다음 작업이 완료되거나 애플리케이션이 IPortableDevicePropertiesBulk:Cancel을 호출하거나 OnProgress에 대한 오류 값을 반환할 때까지 정보를 사용하여 IPortableDevicePropertiesBulkCallback::OnProgress를 반복적으로 호출합니다. 마지막으로 작업이 성공적으로 완료되었는지 여부에 관계없이 Windows 이식 가능한 디바이스는 IPortableDevicePropertiesBulkCallback::OnEnd를 호출합니다.
IPortableDeviceResources IPortableDeviceResources 인터페이스는 개체의 원시 데이터에 대한 액세스를 제공합니다. 이 인터페이스를 사용하여 개체에서 리소스를 읽거나 씁니다. 이 인터페이스를 얻으려면 IPortableDeviceContent::Transfer를 호출합니다.
IPortableDeviceService 서비스에 대한 액세스를 제공합니다.
IPortableDeviceServiceCapabilities 서비스의 기능을 설명하는 정보를 검색합니다.
IPortableDeviceServiceManager 서비스와 연결된 디바이스 및 디바이스에 있는 서비스 목록을 검색합니다.
IPortableDeviceServiceMethodCallback 애플리케이션이 콜백 메서드의 완료를 추적하는 데 사용하는 메서드를 포함합니다. 서비스 메서드를 비동기적으로 호출하는 애플리케이션은 이 인터페이스를 구현하고 IPortableDeviceServiceMethods::InvokeAsync에 매개 변수로 제공할 수 있습니다.
IPortableDeviceServiceMethods 서비스에서 메서드 호출을 호출하거나 취소합니다.
IPortableDeviceUnitsStream IPortableDeviceUnitsStream 인터페이스는 프레임 또는 밀리초와 같은 대체 단위를 사용하여 스트림에서 작동하거나 검색하는 방법을 제공합니다.

WPD Automation 개체 모델

아티클2023. 08. 22.

WPD Automation 개체 모델 기술 개요입니다.

WPD Automation 개체 모델 기술은 헤더와 연결되지 않습니다.

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- WPD Automation 개체 모델

Functions

GetDeviceDispatch 지정된 WPD 디바이스 식별자에 대한 WPD Automation Device 개체를 인스턴스화합니다. (IPortableDeviceDispatchFactory.GetDeviceDispatch)
GetDeviceFromId 지정된 WPD 디바이스 식별자에 대한 WPD Automation Device 개체를 인스턴스화합니다. (IPortableDeviceWebControl.GetDeviceFromId)
GetDeviceFromIdAsync 지정된 WPD 디바이스 식별자에 대해 WPD Automation Device 개체를 비동기적으로 인스턴스화합니다.

인터페이스

IPortableDeviceDispatchFactory WPD Automation Device 개체를 인스턴스화할 수 있는 팩터리를 나타냅니다.
IPortableDeviceWebControl Windows 스토어 앱에서 WPD Automation Device 개체를 인스턴스화할 수 있는 팩터리를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

WPF 비트맵 효과

아티클2023. 03. 09.

WPF 비트맵 효과 기술 개요입니다.

WPF 비트맵 효과를 개발하려면 다음 헤더가 필요합니다.

- mileffects.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- WPF 비트맵 효과

Functions

_NewEnum 새 열거형을 검색합니다.
추가 IMILBitmapEffectGroup에 효과를 추가합니다.
AddBackLink IMILBitmapEffectOutputConnectorImpl::AddBackLink 메서드
ConnectTo 입력 커넥터를 지정된 출력 커넥터에 연결합니다.
CreateContext IMILBitmapEffectRenderContext 개체를 만듭니다.
CreateEffect IMILBitmapEffect 개체를 만듭니다.
CreateEffectOuter 외부 IMILBitmapEffect 개체를 만듭니다.

Dirtyregion 지정된 IMILBitmapEffectPrimitive의 지정된 영역을 무효화합니다.
get_Count 열거형의 항목 수를 검색합니다.
get_Parent 열거형의 부모 효과 그룹을 검색합니다.
GetAffineMatrix 효과에 대한 아핀 변환 행렬을 검색합니다.
GetBitmapEffect 커넥터와 연결된 IMILBitmapEffect를 가져옵니다.
GetChildren 효과 그룹의 자식을 가져옵니다.
GetConnection 입력 커넥터가 연결된 IMILBitmapEffectOutputConnector를 가져옵니다.
GetConnection 출력 커넥터와 연결된 IMILBitmapEffectInputConnector를 가져옵니다.
GetFinalTransform 최종 MILMatrixF 변환을 가져옵니다.
Getformat 지정된 핀의 픽셀 형식을 검색합니다.
GetIndex 핀에 대한 0부터 시작하는 인덱스 값을 검색합니다.
GetInputBitmapSource 지정된 렌더링 컨텍스트의 효과에 대한 입력 비트맵 소스를 가져옵니다.

GetInputConnector 지정된 핀 인덱스와 연결된 입력 커넥터를 검색합니다.
GetInputConnector 내부 커넥터와 연결된 IMILBitmapEffectInputConnector를 가져옵니다.
GetInputConnectorInfo 지정된 입력 핀과 연결된 IMILBitmapEffectConnectorInfo를 검색합니다.
GetInputSource 입력 IWICBitmapSource 인터페이스를 검색합니다.
GetInputSourceBounds 입력 소스의 범위를 가져옵니다.
GetInteriorInputConnector 지정된 인덱스의 효과에 대한 입력 커넥터를 검색합니다.
GetInteriorOutputConnector 지정된 인덱스의 효과에 대한 출력 커넥터를 검색합니다.
GetNumberChildren 효과 그룹의 자식 수를 검색합니다.
GetNumberConnections 출력 커넥터에 있는 연결 수를 검색합니다.
GetNumberFormats 핀에서 지원하는 픽셀 형식의 수를 검색합니다.
GetNumberInputs 비트맵 효과가 구현하는 입력 핀 수를 검색합니다.
GetNumberOutputs 비트맵 효과가 구현하는 출력 핀 수를 검색합니다.

GetOptimalFormat 핀에 대한 최적의 픽셀 형식을 검색합니다.
GetOutput 효과의 출력을 가져옵니다.
GetOutput 비트맵 효과에 대한 픽셀 처리를 수행합니다.
GetOutputBitmapSource 지정된 렌더링 컨텍스트의 효과에 대한 출력 비트맵 소스를 가져옵니다.
GetOutputBounds 렌더링 컨텍스트의 출력 범위를 가져옵니다.
GetOutputConnector 지정된 핀 인덱스와 연결된 출력 커넥터를 검색합니다.
GetOutputConnector 내부 출력 커넥터와 연결된 IMILBitmapEffectOutputConnector를 가져옵니다.
GetOutputConnectorInfo 지정된 출력 핀과 연결된 IMILBitmapEffectConnectorInfo를 검색합니다.
GetOutputDPI 인치당 출력 점(dpi)을 가져옵니다.
GetOutputPixelFormat 출력 픽셀 형식 GUID를 가져옵니다.
GetParentEffect 효과의 부모를 가져옵니다.
GetTransform 렌더링 컨텍스트의 행렬 변환을 가져옵니다.

GetUseSoftwareRenderer 소프트웨어 렌더링을 사용할지 여부를 나타내는 값을 가져옵니다.
HasAffineTransform 효과에 아핀 변환이 있는지 여부를 확인합니다.
HasInverseTransform 효과에 역 변환이 있는지 여부를 확인합니다.
초기화 지정된 개체를 사용하여 효과를 초기화합니다.
IsConnected 커넥터가 효과에 연결되어 있는지 여부를 확인합니다.
IsDirty 효과를 업데이트해야 하는지 여부를 결정합니다.
IsInPlaceModificationAllowed 효과에서 현재 위치 수정을 허용하는지 여부를 결정합니다.
IsVolatile 현재 효과가 휘발성으로 간주되는지 여부를 결정합니다. 효과가 일시적이면 효과 프레임워크는 효과의 출력을 캐시하려고 시도하지 않습니다.
Item 지정된 인덱스에서 효과를 검색합니다.
전처리 지정된 렌더링 컨텍스트를 미리 처리합니다.
PropertyChange IMILBitmapEffectPrimitive에 속성 변경에 대해 알릴 수 있습니다.
RemoveBackLink IMILBitmapEffectOutputConnectorImpl::RemoveBackLink 메서드

SetInitialTransform 초기 MILMatrixF 변환을 가져옵니다.
SetInputSource 효과 입력 소스를 설정합니다.
SetOutputDPI dpi(인치당 출력 점)를 설정합니다.
SetOutputPixelFormat 출력 픽셀 형식을 설정합니다.
SetParentEffect 효과의 부모를 설정합니다.
SetRegionOfInterest 효과에 대한 관심 영역을 설정합니다.
SetUseSoftwareRenderer 소프트웨어 렌더링을 사용할지 여부를 나타내는 값을 설정합니다.
TransformPoint 지정된 지점을 변환합니다.
TransformRect 지정된 사각형의 출력을 변환합니다.
UpdateOutputBounds 지정된 지역을 사용하여 출력 범위를 업데이트.
UpdateTransform 출력 변환을 새 행렬로 업데이트.

인터페이스

IMILBitmapEffect WPF(Windows Presentation Foundation) 비트맵 효과를 정의하는 메서드를 노출합니다.
IMILBitmapEffectConnections 비트맵 효과에 의해 노출되는 입력 및 출력 커넥터를 검색하는 데 사용되는 메서드를 노출합니다.
IMILBitmapEffectConnectionsInfo 비트맵 효과에 의해 노출되는 입력 및 출력 핀에 대한 정보를 검색하는 메서드를 노출합니다.
IMILBitmapEffectConnector 효과 출력 핀을 정의하는 메서드를 노출합니다.
IMILBitmapEffectConnectorInfo 특정 입력 또는 출력 커넥터 핀에 대한 정보를 검색하는 메서드를 노출합니다.
IMILBitmapEffectEvents 효과 이벤트를 정의하는 메서드를 노출합니다.
IMILBitmapEffectFactory WPF(Windows Presentation Foundation) Microsoft Win32 비트맵 효과 개체를 만드는 데 사용되는 메서드를 노출합니다.
IMILBitmapEffectGroup 효과 그룹에 액세스하는 데 사용되는 메서드를 노출합니다.
IMILBitmapEffectGroupImpl 효과 그룹을 정의하는 메서드를 노출합니다.
IMILBitmapEffectImpl OUT IMILBitmapEffect 개체를 정의하는 메서드를 노출합니다.
IMILBitmapEffectInputConnector 입력 연결을 정의하는 메서드를 노출합니다.
IMILBitmapEffectInteriorInputConnector 내부 입력 커넥터를 정의하는 메서드를 노출합니다.

IMILBitmapEffectInteriorOutputConnector 내부 출력 커넥터를 정의하는 메서드를 노출합니다.
IMILBitmapEffectOutputConnector 출력 커넥터를 정의하는 메서드를 노출합니다. (IMILBitmapEffectOutputConnector)
IMILBitmapEffectOutputConnectorImpl 출력 커넥터를 정의하는 메서드를 노출합니다. (IMILBitmapEffectOutputConnectorImpl)
IMILBitmapEffectPrimitive 비트맵 효과의 출력을 만드는 메서드를 노출합니다. WPF(타사 Windows Presentation Foundation) 비트맵 효과를 만들려면 이 인터페이스를 구현해야 합니다.
IMILBitmapEffectPrimitiveImpl 비트맵 효과의 상태를 보고하는 메서드를 노출합니다. WPF(타사 Windows Presentation Foundation) 비트맵 효과를 만들려면 이 인터페이스를 구현해야 합니다.
IMILBitmapEffectRenderContext IMILBitmapEffectRenderContext 개체를 정의하는 메서드를 노출합니다.
IMILBitmapEffectRenderContextImpl IMILBitmapEffectRenderContext를 정의하는 메서드를 노출합니다.
IMILBitmapEffects 효과의 열거형을 정의하는 메서드를 노출합니다.

구조체

MilMatrix3x2D 3x3 행렬을 나타냅니다.
MILMatrixF 4x4 아핀 변환 행렬을 나타냅니다.

MilPoint2D 2차원 공간의 x 및 y 좌표 쌍을 나타냅니다. M
MilRectD 사각형의 너비, 높이 및 위치를 설명합니다. M

XAML 진단

아티클2023. 08. 22.

XAML 진단 기술 개요입니다.

XAML 진단을 개발하려면 다음 헤더가 필요합니다.

- xamlom.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- XAML 진단

열거형

BaseValueSource 속성의 유효 값이 설정된 위치를 지정하는 상수를 정의합니다.
MetadataBit XAML 진단에서 반환된 PropertyChainValue를 정의하는 데 사용되는 상수를 정의합니다.
RenderTargetBitmapOptions 시각적 트리의 렌더링할 부분을 지정하는 상수를 정의합니다.
ResourceType 리소스 사전에서 리소스의 형식을 지정하는 상수를 정의합니다.
VisualElementState 시각적 트리에서 요소의 상태를 지정하는 상수를 정의합니다.
VisualMutationType 요소가 라이브 시각적 트리에 추가되거나 제거되었는지 여부를 지정하는 상수를 정의합니다.

Functions

Addchild 지정된 인덱스에서 컬렉션에 자식 요소를 추가합니다.
AddDictionaryItem ResourceDictionary에 항목을 추가하고 지정된 키를 사용하여 리소스를 참조하는 트리의 모든 요소를 다시 확인합니다.
AdviseVisualTreeChange 시각적 트리에 대한 변경 내용 수신 대기를 시작합니다.
ClearChildren 부모 컬렉션에서 모든 자식 요소를 지웁니다.
ClearProperty XAML 요소에서 지정된 속성을 지웁니다.
CopyBytesTo 비트맵 데이터의 지정된 오프셋에서 지정된 최대 바이트 수까지 호출자의 버퍼(pvBytes)로 복사하고 복사된 바이트 수를 반환합니다.
CreateInstance XAML 런타임, 열거형 또는 기본 형식의 instance 만듭니다.
GetApplication 애플리케이션의 instance 가져옵니다.
GetBitmapDescription IBitmapData에 저장된 비트맵 데이터를 설명하는 BitmapDescription을 가져옵니다.
GetCollectionCount 컬렉션의 수를 가져옵니다.
GetCollectionElements 컬렉션의 요소를 가져옵니다.
GetDictionaryItem ResourceDictionary에서 항목을 가져옵니다.

GetDispatcher UI 스레드의 요소에 액세스하는 데 사용되는 코어 디스패처를 가져옵니다.
GetEnums XAML 런타임 및 총 개수에 정의된 모든 열거형의 배열을 가져옵니다.
GetHandleFromIInspectable IInspectable의 InstanceHandle 표현을 가져옵니다.
GetIInspectableFromHandle XAML 진단 캐시에서 IInspectable을 가져옵니다.
GetInitializationData XAML 진단에 전달된 초기화 데이터를 가져옵니다.
GetProperty 지정된 종속성 속성의 유효 값을 가져옵니다.
GetPropertyIndex 지정된 속성 이름의 속성 인덱스 를 가져옵니다.
GetPropertyValuesChain 전달된 요소에 설정된 모든 속성의 배열과 속성의 유효 값을 설정하는 데 관련된 모든 스타일의 배열을 가져옵니다.
GetSourceBitmapDescription IBitmapData에 저장된 비트맵 데이터의 원래 형식을 설명하는 BitmapDescription을 가져옵니다.
GetStride 데이터의 보폭을 가져옵니다. 비트맵의 각 행 길이(바이트)입니다.
GetUiLayer 트리의 요소를 강조 표시하기 위해 그리는 데 사용할 수 있는 시각적 개체 진단 루트를 가져옵니다.
Hittest 지정된 사각형 내에 있는 시각적 트리의 모든 요소를 가져옵니다.

InitializeXamlDiagnosticsEx Xaml Diagnostics 세션을 초기화합니다. XAML 진단 API를 사용하는 디버깅 도구의 진입점입니다.
OnElementStateChanged 변경되는 경우 시각적 트리에서 요소의 상태를 전달합니다.
OnVisualTreeChange 변경 시 시각적 트리의 상태를 전달합니다.
RegisterInstance XAML Diagnostics 캐시에 IInspectable을 추가하고 개체에 대해 새로 만든 InstanceHandle을 반환합니다.
RemoveChild 지정된 인덱스에서 자식 요소를 제거합니다.
RemoveDictionaryItem ResourceDictionary에서 항목을 제거하고 지정된 키를 사용하여 리소스를 참조하는 트리의 모든 요소를 다시 확인합니다.
RenderTargetBitmap 핸들에서 설명하는 개체를 나타내는 이미지를 반환하거나 개체에 이러한 이미지가 없거나 제공할 수 없는 경우 오류를 반환합니다.
ReplaceResource 기존 리소스를 동일한 형식의 새 리소스로 바꿉니다.
ResolveResource 트리의 요소에 대한 리소스를 확인하고 지정된 속성 인덱스에서 제공하는 속성에 리소스를 적용합니다.
SetProperty XAML 요소의 속성 값을 설정합니다.
UnadviseVisualTreeChange 시각적 트리의 변경 내용 수신 대기를 중지합니다.

인터페이스

IBitmapData 시각적 트리의 노드와 연결된 이미지를 나타냅니다.
IVisualTreeService XAML 시각적 트리를 관리하는 메서드를 제공합니다.
IVisualTreeService2 IVisualTreeService 개체의 추가 기능을 나타냅니다.
IVisualTreeService3 IVisualTreeService2 개체의 추가 기능을 나타냅니다.
IVisualTreeServiceCallback 시각적 트리의 상태를 전달합니다.
IVisualTreeServiceCallback2 IVisualTreeServiceCallback 개체의 추가 기능을 나타냅니다.
IXamlDiagnostics XAML 진단 세션을 나타냅니다.

구조체

BitmapDescription IBitmapData에 저장된 비트맵에 대한 정보를 나타냅니다.
CollectionElementValue 컬렉션의 요소를 나타냅니다.
EnumType XAML 런타임 열거형을 나타냅니다.

ParentChildRelation 부모 개체를 자식 개체와 연결합니다.
PropertyChainSource 대상 형식의 원본 개체(Style)를 나타냅니다.
PropertyChainValue 요소에 정의된 속성을 나타냅니다.
SourceInfo 개체의 XAML 원본 문서에 대한 정보를 나타냅니다.
VisualElement Microsoft Visual Studio의 라이브 시각적 트리에 있는 XAML 요소를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

XAudio2 API

아티클2023. 03. 11.

XAudio2 API 기술 개요입니다.

XAudio2 API를 개발하려면 다음 헤더가 필요합니다.

- hrtfapoapi.h
- x3daudio.h
- xapo.h
- xapobase.h
- xapofx.h
- xaudio2.h
- xaudio2fx.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- XAudio2 API

클래스

CXAPOBase IXAPO 인터페이스의 기본 구현입니다.
CXAPOParametersBase IXAPOParameters 인터페이스의 기본 구현입니다.

열거형

HrtfDirectivityType 여러 주식 직접성 패턴 중 하나를 나타냅니다.
HrtfDistanceDecayType 소리에 적용되는 거리 기반 감쇠 유형을 나타냅니다.

HrtfEnvironment 여러 주식 환경 유형 중 하나를 나타냅니다.
XAPO_BUFFER_FLAGS 스트림 버퍼의 내용을 설명합니다.
XAUDIO2_FILTER_TYPE 필터 유형을 나타냅니다.

Functions

AddRef XAudio2 개체에 대한 참조를 추가합니다.
BeginProcess 현재 프로세스 매개 변수를 반환합니다.
CalcInputFrames 지정된 출력 프레임 수를 생성하는 데 필요한 입력 프레임 수를 반환합니다.
CalcOutputFrames 지정된 입력 프레임 수에서 생성되는 출력 프레임 수를 반환합니다.
CommitChanges 지정된 식별자를 사용하여 태그가 지정된 작업 집합을 원자성으로 적용합니다.
CreateFX 요청된 XAPOFX 효과의 인스턴스를 만듭니다.
CreateHrtfApo HRTF(헤드 관련 전송 함수) 처리를 위한 IXAPO 인터페이스의 인스턴스를 만듭니다.
CreateMasteringVoice 마스터링 음성을 만들고 구성합니다.

CreateSourceVoice 원본 음성을 만들고 구성합니다.
CreateSubmixVoice 서브믹스 음성을 만들고 구성합니다.
CXAPOBase CXAPOBase 클래스의 인스턴스를 만듭니다.
CXAPOParametersBase CXAPOParametersBase 클래스의 인스턴스를 만듭니다.
DestroyVoice 음성을 삭제합니다. 필요한 경우 음성을 중지하고 XAudio2 그래프에서 제거합니다.
DisableEffect 음성의 효과 체인에서 지정된 위치에서 효과를 사용하지 않도록 설정합니다.
불연속 XAudio2 음성에 현재 큐에 있는 마지막 버퍼 이후 버퍼가 더 이상 오지 않음을 알 수 있습니다.
EnableEffect 음성의 효과 체인에서 지정된 위치에서 효과를 사용하도록 설정합니다.
EndProcess CXAPOParametersBase에 XAPO가 현재 프로세스 매개 변수에 대한 액세스를 완료했음을 알 수 있습니다.
ExitLoop 현재 루프 영역의 끝에 도달하면 음성 반복을 중지합니다.
FlushSourceBuffers 음성 큐에서 보류 중인 모든 오디오 버퍼를 제거합니다.
GetChannelMask 이 음성의 채널 마스크를 반환합니다.

GetChannelVolumes 채널당 음성의 볼륨 수준을 반환합니다.
GetEffectParameters 음성 효과 체인에서 지정된 효과의 현재 효과별 매개 변수를 반환합니다.
GetEffectState 음성의 효과 체인에서 지정된 위치에 있는 효과의 실행 상태를 반환합니다.
GetFilterParameters 음성의 필터 매개 변수를 가져옵니다.
GetFrequencyRatio 음성의 주파수 조정 비율을 반환합니다.
GetOutputFilterParameters 이 음성의 송신 중 하나에서 필터 매개 변수를 반환합니다.
GetOutputMatrix 음성에 대한 최종 출력의 각 채널 볼륨 수준을 가져옵니다. 이러한 채널은 지정된 대상 음성의 입력 채널에 매핑됩니다.
GetParameters 효과별 매개 변수의 현재 값을 가져옵니다.
GetPerformanceData 사용 가능한 메모리 또는 CPU 사용량과 같은 현재 리소스 사용량 세부 정보를 반환합니다.
GetRegistrationProperties XAPO의 등록 속성을 반환합니다.
GetRegistrationPropertiesInternal XAPO가 만들어진 등록 속성을 포함하는 XAPO_REGISTRATION_PROPERTIES 구조체에 대한 포인터를 반환합니다.
GetState 음성의 현재 커서 위치 데이터를 반환합니다.

GetVoiceDetails 음성의 생성 플래그, 입력 채널 및 샘플 속도에 대한 정보를 반환합니다.
GetVolume 음성의 현재 전체 볼륨 수준을 가져옵니다.
초기화 효과별 초기화를 수행합니다.
IsInputFormatSupported 특정 입력 형식이 지정된 출력 형식에 대해 지원되는지 쿼리합니다.
IsLocked XAPO가 잠겨 있는지 여부를 쿼리합니다.
IsOutputFormatSupported 지정된 입력 형식에 대해 특정 출력 형식이 지원되는지 쿼리합니다.
LockForProcess XAudio2에서 호출되어 실시간 스레드에서 프로세스가 호출되기 전에 최종 초기화를 수행할 수 있도록 하는 XAPO의 입력 및 출력 구성을 잠급니다.
OnBufferEnd 음성이 버퍼 처리를 완료할 때 호출됩니다.
OnBufferStart 음성이 새 오디오 버퍼 처리를 시작할 때 호출됩니다.
OnCriticalError XAudio2를 종료하고 다시 시작해야 하는 중요한 시스템 오류가 발생하는 경우 호출됩니다.
OnLoopEnd 음성이 루프의 끝 위치에 도달하면 호출됩니다.
OnProcessingPassEnd 오디오 처리 패스가 종료된 직후 XAudio2에서 호출됩니다.

OnProcessingPassStart 오디오 처리 패스가 시작되기 직전에 XAudio2에서 호출됩니다.
OnSetParameters 사용자 정의 매개 변수 유효성 검사를 허용하기 위해 IXAPOParameters::SetParameters에서 호출됩니다.
OnStreamEnd 음성이 연속 오디오 스트림 재생을 막 마쳤을 때 호출됩니다.
OnVoiceError 음성 처리 중에 심각한 오류가 발생할 때 호출됩니다.
OnVoiceProcessingPassEnd 음성에 대한 처리 패스가 종료된 직후에 호출됩니다.
OnVoiceProcessingPassStart XAudio2가 음성의 버퍼 큐에서 데이터를 읽기 직전에 각 음성에 대한 각 처리 패스 중에 호출됩니다.
ParametersChanged IXAPOParameters::SetParameters가 마지막 처리 통과 이후 호출되었는지를 나타냅니다.
처리 지정된 입력 및 출력 버퍼에서 XAPO의 DSP(디지털 신호 처리) 코드를 실행합니다.
ProcessThru 처리를 통해 XAPO를 사용하지 않도록 설정하면 IXAPO::P rocess 구현에서 호출됩니다.
QueryInterface XAudio2 개체에서 지정된 COM 인터페이스에 대한 쿼리입니다.
RegisterForCallbacks XAudio2 엔진 콜백 목록에 IXAudio2EngineCallback 포인터를 추가합니다.
릴리스 XAudio2 개체에 대한 참조를 해제합니다.

재설정 프레임 기록에 종속된 변수를 다시 설정합니다.
ReverbConvertI3DL2ToNative I3DL2(대화형 3D 오디오 렌더링 지침 수준 2.0) 매개 변수를 네이티브 XAudio2 매개 변수로 변환하는 인라인 함수입니다.
SetChannelVolumes 채널당 음성의 볼륨 수준을 설정합니다.
SetDebugConfiguration XAudio2에 대한 전역 디버그 로깅 옵션을 변경합니다.
SetEffectChain 음성의 효과 체인을 바꿉니다.
SetEffectParameters 음성 효과 체인에서 지정된 효과에 대한 매개 변수를 설정합니다.
SetEnvironment 시뮬레이션할 음향 환경을 선택합니다.
SetFilterParameters 음성의 필터 매개 변수를 설정합니다.
SetFrequencyRatio 음성의 주파수 조정 비율을 설정합니다.
SetOutputFilterParameters 이 음성의 송신 중 하나에 대한 필터 매개 변수를 설정합니다.
SetOutputMatrix 음성에 대한 최종 출력의 각 채널 볼륨 수준을 설정합니다. 이러한 채널은 지정된 대상 음성의 입력 채널에 매핑됩니다.
SetOutputVoices 음성의 출력을 받을 새 서브믹스 또는 마스터링 음성 집합을 지정합니다.

SetParameters 효과별 매개 변수를 설정합니다.
SetSourceGain 현재 원본 위치에 대한 사용자 지정 직접 경로 게인 값을 설정합니다. HrtfDistanceDecayType 사용자 지정 감쇠 유형으로 재생되는 소리에만 유효합니다.
SetSourceOrientation 수신기의 좌표계와 관련하여 원본 방향에 대한 회전 행렬을 설정합니다.
SetSourcePosition 수신기를 기준으로 소리의 위치를 설정합니다.
SetSourceSampleRate 음성을 만들 때 지정된 속도와 다른 샘플 속도로 원본 데이터를 사용하도록 음성을 다시 구성합니다.
SetVolume 음성의 전체 볼륨 수준을 설정합니다.
시작 음성으로 오디오 사용 및 처리를 시작합니다. 연결된 서브믹스 또는 마스터링 음성 또는 출력 디바이스에 결과를 전달합니다.
Startengine 오디오 처리 스레드를 시작합니다.
중지 현재 음성으로 오디오 소비를 중지합니다.
StopEngine 오디오 처리 스레드를 중지합니다.
SubmitSourceBuffer 음성 큐에 새 오디오 버퍼를 추가합니다.

UnlockForProcess LockForProcess 메서드를 사용하여 할당된 변수의 할당을 취소합니다.
UnregisterForCallbacks XAudio2 엔진 콜백 목록에서 IXAudio2EngineCallback 포인터를 제거합니다.
ValidateFormatDefault 오디오 형식이 지원되는 기본 범위 내에 속하는지 확인합니다.
ValidateFormatPair XAPO에서 입력 및 출력 형식 쌍 구성을 지원하는지 확인합니다.
X3DAudioCalculate 3D 매개 변수에 대한 DSP 설정을 계산합니다.
X3DAudioInitialize 모든 전역 3D 오디오 상수 설정
XAPOAlloc 이후에 애플리케이션에 반환되는 임의의 크기 구조를 할당해야 하는 IXAPO 메서드에서 사용하는 메모리 할당 매크로입니다.
XAPOFree XAPOAlloc 매크로를 사용하여 할당된 메모리를 해제하는 데 사용되는 매크로입니다.
XAudio2AmplitudeRatioToDecibels 진폭 비율 값을 데시벨 값으로 변환하는 인라인 함수입니다.
XAudio2Create 새 XAudio2 개체를 만들고 IXAudio2 인터페이스에 대한 포인터를 반환합니다.
XAudio2CreateReverb APO(새 반향 오디오 처리 개체)를 만들고 포인터를 반환합니다.
XAudio2CreateVolumeMeter 새 볼륨 미터 APO(오디오 처리 개체)를 만들고 포인터를 반환합니다.

XAudio2CutoffFrequencyToOnePoleCoefficient 헤르츠로 표현된 필터 차단 주파수에서 XAUDIO2_FILTER_PARAMETERS 구조체의 Frequency 멤버와 함께 사용되는 필터 계수로 변환하는 인라인 함수입니다.
XAudio2CutoffFrequencyToRadians 헤르츠로 표현된 필터 차단 주파수에서 XAUDIO2_FILTER_PARAMETERS 구조체의 Frequency 멤버에 사용되는 라디안 주파수 값으로 변환하는 인라인 함수입니다.
XAudio2DecibelsToAmplitudeRatio 데시벨 값을 진폭 비율 값으로 변환하는 인라인 함수입니다.
XAudio2FrequencyRatioToSemitones 빈도 비율 값을 세미톤 값으로 변환하는 인라인 함수입니다.
XAudio2RadiansToCutoffFrequency XAUDIO2_FILTER_PARAMETERS 사용된 라디안 주파수에서 헤르츠의 절대 주파수로 다시 변환하는 인라인 함수입니다.
XAudio2SemitonesToFrequencyRatio 세미톤 값을 빈도 비율 값으로 변환하는 인라인 함수입니다.

인터페이스

IXAPO XAudio2 효과 체인에 사용되는 오디오 처리 개체의 인터페이스입니다.
IXAPOHrtfParameters HRTF(헤드 관련 전송 함수)가 소리에 적용되는 방식을 제어하는 매개 변수를 설정하는 데 사용되는 인터페이스입니다.
IXAPOParameters XAPO가 효과별 매개 변수를 사용할 수 있도록 하는 선택적 인터페이스입니다.

IXAudio2 IXAudio2는 모든 오디오 엔진 상태, 오디오 처리 스레드, 음성 그래프 등을 관리하는 XAudio2 개체의 인터페이스입니다.
IXAudio2EngineCallback IXAudio2EngineCallback 인터페이스에는 IXAudio2 엔진에서 특정 이벤트가 발생할 때 클라이언트에 알리는 메서드가 포함되어 있습니다.
IXAudio2MasteringVoice 마스터링 음성은 오디오 출력 디바이스를 나타내는 데 사용됩니다.
IXAudio2SourceVoice 원본 음성을 사용하여 XAudio2 처리 파이프라인에 오디오 데이터를 제출합니다.
IXAudio2SubmixVoice 서브믹스 음성은 주로 성능 향상 및 효과 처리에 사용됩니다.
IXAudio2Voice IXAudio2Voice는 IXAudio2SourceVoice, IXAudio2SubmixVoice 및 IXAudio2MasteringVoice가 파생되는 기본 인터페이스를 나타냅니다. 아래에 나열된 메서드는 모든 음성 서브클래스에 공통적으로 적용됩니다.
IXAudio2VoiceCallback IXAudio2VoiceCallback 인터페이스에는 지정된 IXAudio2SourceVoice에서 특정 이벤트가 발생할 때 클라이언트에 알리는 메서드가 포함되어 있습니다.

구조체

FXECHO_INITDATA FXECHO XAPOFX와 함께 사용할 초기화 매개 변수입니다.
FXECHO_PARAMETERS FXECHO XAPOFX와 함께 사용할 매개 변수입니다.

FXEQ_PARAMETERS FXEQ XAPO와 함께 사용할 매개 변수입니다.
FXMASTERINGLIMITER_PARAMETERS FXMasteringLimiter XAPO와 함께 사용할 매개 변수입니다.
FXREVERB_PARAMETERS FXReverb XAPO와 함께 사용할 매개 변수입니다.
HrtfApoInit HRTF 공간 오디오를 초기화하는 데 사용되는 매개 변수를 지정합니다.
HrtfDirectivity 기본 직접성 패턴 설명자입니다. 소리에 적용되는 직접성의 유형을 설명합니다.
HrtfDirectivityCardioid 심장 직접성 패턴을 설명합니다.
HrtfDirectivityCone 원뿔 지향성을 설명합니다.
HrtfDistanceDecay 거리 기반 감쇠 동작을 설명합니다.
HrtfOrientation HRTF 직접성 개체의 방향을 나타냅니다.
HrtfPosition 오른손 좌표계를 사용하여 3D 공간의 위치를 나타냅니다.
X3DAUDIO_CONE 방출기 방향과 관련하여 DSP 동작의 크기를 조정하여 단일 채널 비 LFE 방출기 방향성을 지정합니다.
X3DAUDIO_DISTANCE_CURVE 선형 세그먼트로 구성된 명시적 조각 곡선을 정의하여 정규화된 거리와 관련하여 DSP 동작을 직접 정의합니다.

X3DAUDIO_DISTANCE_CURVE_POINT 지정된 정규화된 거리에서 DSP 설정을 정의합니다.
X3DAUDIO_DSP_SETTINGS X3DAudioCalculate 호출에서 결과를 받습니다.
X3DAUDIO_EMITTER 임의의 수의 사운드 채널과 함께 사용되는 단일 지점 또는 다중 지점 3D 오디오 원본을 정의합니다.
X3DAUDIO_LISTENER 3D 오디오 수신 지점을 정의합니다.
XAPO_LOCKFORPROCESS_PARAMETERS XAPO가 잠겨 있는 동안 일정하게 유지되는 스트림 버퍼 매개 변수를 정의합니다. IXAPO::LockForProcess 메서드와 함께 사용됩니다.
XAPO_PROCESS_BUFFER_PARAMETERS 한 호출에서 다음 호출로 변경할 수 있는 스트림 버퍼 매개 변수를 정의합니다. Process 메서드와 함께 사용됩니다.
XAPO_REGISTRATION_PROPERTIES XAPO의 일반적인 특징을 설명합니다. IXAPO::GetRegistrationProperties, CXAPOParametersBase::CXAPOParametersBase 및 CXAPOBase::CXAPOBase와 함께 사용됩니다.
XAUDIO2_BUFFER IXAudio2SourceVoice::SubmitSourceBuffer와 함께 사용되는 오디오 데이터 버퍼를 나타냅니다.
XAUDIO2_BUFFER_WMA xWMA 데이터를 제출할 때 IXAudio2SourceVoice::SubmitSourceBuffer와 함께 사용됩니다.
XAUDIO2_DEBUG_CONFIGURATION XAudio2에 대한 새 전역 디버그 구성을 포함합니다. SetDebugConfiguration 함수와 함께 사용됩니다.
XAUDIO2_EFFECT_CHAIN 효과 체인을 정의합니다.

XAUDIO2_EFFECT_DESCRIPTOR 효과 체인에서 사용할 XAPO에 대한 정보를 포함합니다.
XAUDIO2_FILTER_PARAMETERS 원본 음성에 대한 필터 매개 변수를 정의합니다.
XAUDIO2_PERFORMANCE_DATA 성능 정보를 포함합니다. (XAUDIO2_PERFORMANCE_DATA)
XAUDIO2_SEND_DESCRIPTOR 다른 음성에서 보내는 대상인 대상 음성을 정의하고 필터를 사용해야 하는지 여부를 지정합니다.
XAUDIO2_VOICE_DETAILS 생성 플래그, 입력 채널 및 음성의 샘플 속도에 대한 정보를 포함합니다.
XAUDIO2_VOICE_SENDS 단일 출력 음성에서 데이터를 받을 음성 집합을 정의합니다.
XAUDIO2_VOICE_STATE 음성의 현재 상태 및 커서 위치 데이터를 반환합니다.
XAUDIO2FX_REVERB_I3DL2_PARAMETERS ReverbConvertI3DL2ToNative 함수에서 사용할 I3DL2(대화형 3D 오디오 렌더링 지침 수준 2.0) 매개 변수에 대해 설명합니다.
XAUDIO2FX_REVERB_PARAMETERS 반향 APO에서 사용하기 위한 매개 변수에 대해 설명합니다.
XAUDIO2FX_VOLUMEMETER_LEVELS 볼륨 미터 APO와 함께 사용하기 위한 매개 변수에 대해 설명합니다.

XblIdpAuthManager

아티클2023. 03. 09.

XblIdpAuthManager 기술 개요입니다.

XblIdpAuthManager를 개발하려면 다음 헤더가 필요합니다.

- xblidpauthmanager.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- XblIdpAuthManager

열거형

XBL_IDP_AUTH_TOKEN_STATUS Microsoft용으로 예약되어 있습니다. (XBL_IDP_AUTH_TOKEN_STATUS)

Functions

GetAgeGroup Microsoft용으로 예약되어 있습니다. (IXblIdpAuthTokenResult.GetAgeGroup)
GetEnforcementBans Microsoft용으로 예약되어 있습니다. (IXblIdpAuthTokenResult.GetEnforcementBans)
GetEnvironment Microsoft용으로 예약되어 있습니다. (IXblIdpAuthManager.GetEnvironment)
GetEnvironment Microsoft용으로 예약되어 있습니다. (IXblIdpAuthTokenResult.GetEnvironment)
GetErrorCode Microsoft용으로 예약되어 있습니다. (IXblIdpAuthTokenResult.GetErrorCode)

GetGamerAccount Microsoft용으로 예약되어 있습니다. (IXblIdpAuthManager.GetGamerAccount)
GetGamertag Microsoft용으로 예약되어 있습니다. (IXblIdpAuthTokenResult.GetGamertag)
GetHelpId Microsoft용으로 예약되어 있습니다. (IXblIdpAuthTokenResult.GetHelpId)
GetMessage Microsoft용으로 예약되어 있습니다. (IXblIdpAuthTokenResult.GetMessage)
GetMsaAccountId Microsoft용으로 예약되어 있습니다. (IXblIdpAuthTokenResult.GetMsaAccountId)
GetMsaAppId Microsoft용으로 예약되어 있습니다. (IXblIdpAuthTokenResult.GetMsaAppId)
GetMsaPolicy Microsoft용으로 예약되어 있습니다. (IXblIdpAuthTokenResult.GetMsaPolicy)
GetMsaTarget Microsoft용으로 예약되어 있습니다. (IXblIdpAuthTokenResult.GetMsaTarget)
GetPrivileges Microsoft용으로 예약되어 있습니다. (IXblIdpAuthTokenResult.GetPrivileges)
GetRedirect Microsoft용으로 예약되어 있습니다. (IXblIdpAuthTokenResult.GetRedirect)
GetRestrictions Microsoft용으로 예약되어 있습니다. (IXblIdpAuthTokenResult.GetRestrictions)
GetSandbox Microsoft용으로 예약되어 있습니다. (IXblIdpAuthManager.GetSandbox)

GetSandbox Microsoft용으로 예약되어 있습니다. (IXblIdpAuthTokenResult.GetSandbox)
GetSignature Microsoft용으로 예약되어 있습니다. (IXblIdpAuthTokenResult.GetSignature)
GetStatus Microsoft용으로 예약되어 있습니다. (IXblIdpAuthTokenResult.GetStatus)
GetTitleRestrictions Microsoft용으로 예약되어 있습니다. (IXblIdpAuthTokenResult.GetTitleRestrictions)
GetToken Microsoft용으로 예약되어 있습니다. (IXblIdpAuthTokenResult.GetToken)
GetTokenAndSignatureWithTokenResult Microsoft용으로 예약되어 있습니다. (IXblIdpAuthManager.GetTokenAndSignatureWithTokenResult)
GetXuid Microsoft용으로 예약되어 있습니다. (IXblIdpAuthTokenResult.GetXuid)
SetAppViewInitialized Microsoft용으로 예약되어 있습니다. (IXblIdpAuthManager.SetAppViewInitialized)
SetGamerAccount Microsoft용으로 예약되어 있습니다. (IXblIdpAuthManager.SetGamerAccount)

인터페이스

IXblIdpAuthManager Microsoft용으로 예약되어 있습니다. (IXblIdpAuthManager)

IXblIdpAuthTokenResult Microsoft용으로 예약되어 있습니다. (IXblIdpAuthTokenResult)

XInput 게임 컨트롤러 API

아티클2024. 03. 05.

XInput 게임 컨트롤러 API 기술 개요입니다.

XInput 게임 컨트롤러 API를 개발하려면 다음 헤더가 필요합니다.

- xinput.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- XInput 게임 컨트롤러 API

XInput 게임 컨트롤러 API와 인터페이스하는 하드웨어 디바이스를 개발하려면 를 참조하세요 https://aka.ms/gipdocs.

Functions

테이블 확장

XInputEnable XInput의 보고 상태를 설정합니다.
XInputGetAudioDeviceIds 지정된 컨트롤러에 연결된 헤드셋과 연결된 사운드 렌더링 및 사운드 캡처 오디오 장치 ID를 검색합니다.
XInputGetBatteryInformation 무선 컨트롤러의 배터리 종류 및 충전 상태 검색합니다.
XInputGetCapabilities 연결된 컨트롤러의 기능과 기능을 검색합니다.
XInputGetDSoundAudioDeviceGuids 지정된 컨트롤러에 연결된 헤드셋과 연결된 사운드 렌더링 및 사운드 캡처 디바이스 GUID를 가져옵니다.
XInputGetKeystroke

게임 패드 입력 이벤트를 검색합니다.
XInputGetState 지정된 컨트롤러의 현재 상태를 검색합니다.
XInputSetState 연결된 컨트롤러에 데이터를 보냅니다. 이 함수는 컨트롤러의 진동 함수를 활성화하는 데 사용됩니다.

구조체

테이블 확장

XINPUT_BATTERY_INFORMATION 배터리 유형 및 충전 상태에 대한 정보를 포함합니다.
XINPUT_CAPABILITIES 연결된 컨트롤러의 기능을 설명합니다. XInputGetCapabilities 함수는 XINPUT_CAPABILITIES 반환합니다.
XINPUT_GAMEPAD 컨트롤러의 현재 상태를 설명합니다.
XINPUT_KEYSTROKE XInputGetKeystroke에서 반환된 키 입력 데이터를 지정합니다.
XINPUT_STATE 컨트롤러의 상태를 나타냅니다.
XINPUT_VIBRATION 컨트롤러의 진동 함수에 대한 모터 속도 수준을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

XML HTTP 확장 요청

아티클2023. 08. 22.

XML HTTP 확장 요청 기술 개요입니다.

XML HTTP 확장 요청을 개발하려면 다음 헤더가 필요합니다.

- msxml6.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- XML HTTP 확장 요청

열거형

XHR_AUTH 프록시에 연결하거나 HTTP 서버에 연결하는 데 인증을 사용할지 여부를 지정합니다.
XHR_CERT_ERROR_FLAG IXMLHTTPRequest3Callback 인터페이스에서 OnServerCertificateReceived 메서드를 처리하여 서버와 SSL 협상 중에 서버 인증서 오류를 나타내는 플래그를 정의합니다.
XHR_CERT_IGNORE_FLAG IXMLHTTPRequest3 인터페이스에서 SetProperty 메서드를 호출하여 특정 인증서 오류를 무시하도록 나가는 HTTP 요청에 할당할 수 있는 플래그를 정의합니다.
XHR_COOKIE_FLAG GetCookie 메서드를 호출하여 SETCookie 메서드 또는 HTTP 쿠키 jar에서 쿼리를 호출하여 HTTP 쿠키 jar의 쿠키에 할당할 수 있는 플래그 집합을 정의합니다.
XHR_COOKIE_STATE 쿠키의 상태를 지정합니다.
XHR_CRED_PROMPT 인증을 위해 사용자에게 자격 증명 프롬프트를 허용할지 여부를 지정합니다.
XHR_PROPERTY SetProperty 메서드를 호출하여 나가는 HTTP 요청에 할당할 수 있는 속성을 정의합니다.

Functions

중단 현재 HTTP 요청을 취소합니다.
GetAllResponseHeaders 모든 HTTP 응답 헤더의 값을 검색합니다.
GetCookie HTTP 쿠키 jar에서 지정된 URL과 연결된 쿠키를 가져옵니다.
GetResponseHeader 응답 헤더에서 HTTP 헤더의 값을 검색합니다.
OnClientCertificateRequested 클라이언트가 서버와 SSL 협상 중에 클라이언트 인증서에 대한 요청을 수신할 때 발생합니다.
OnDataAvailable 클라이언트가 서버에서 HTTP 응답 데이터의 일부를 수신할 때 발생합니다.
OnError 오류가 발생하거나 요청이 중단되었을 때 발생합니다.
OnHeadersAvailable HTTP 요청이 서버로 전송되고 서버가 응답 헤더로 응답한 후에 발생합니다.
OnRedirect 클라이언트가 서버가 새 URL로 리디렉션하는 HTTP 요청을 보낼 때 발생합니다.
OnResponseReceived 클라이언트가 서버로부터 전체 응답을 받았을 때 발생합니다.
OnServerCertificateReceived 클라이언트가 서버와 SSL 협상 중에 인증서 오류 또는 서버 인증서 체인을 수신할 때 발생합니다.
열기

IXMLHTTPRequest2 요청을 초기화하고 요청에 대한 메서드, URL 및 인증 정보를 지정합니다. 이 메서드를 호출한 후 Send 메서드를 호출하여 요청 및 데이터(있는 경우)를 서버로 보내야 합니다.
'보내기' HTTP 요청을 서버에 비동기적으로 보냅니다. 성공하면 앱에서 구현한 IXMLHTTPRequest2Callback 인터페이스의 메서드가 호출되어 응답을 처리합니다.
SetClientCertificate Open 메서드에 지정된 URL에 대해 인증하는 데 사용할 클라이언트 인증서를 설정합니다.
SetCookie HTTP 쿠키 jar에서 지정된 URL과 연결된 쿠키를 설정합니다.
SetCustomResponseStream HTTP 응답을 수신하기 위한 표준 스트림을 대체할 사용자 지정 스트림을 제공합니다.
SetProperty 나가는 HTTP 요청에 대한 속성을 설정합니다.
SetRequestHeader 기본 요청 헤더와 함께 서버로 보낼 HTTP 헤더의 이름을 지정합니다.

인터페이스

IXMLHTTPRequest2 HTTP 요청을 구성 및 보내고 콜백을 사용하여 HTTP 응답 처리 중에 알림을 수신하는 데 필요한 메서드 및 속성을 제공합니다. 참고 이 인터페이스는 Windows Phone 8.1에서 지원됩니다. .
IXMLHTTPRequest2Callback HTTP 요청 및 응답 처리에 영향을 주는 이벤트의 미해결 IXMLHTTPRequest2 요청을 사용하여 애플리케이션에 알리는 콜백을 정의합니다. 참고 이 인터페이스는 Windows Phone 8.1에서 지원됩니다. .
IXMLHTTPRequest3 HTTP 요청을 구성 및 보내고 콜백을 사용하여 HTTP 응답 처리 중에 알림을 수신하는 데 필요한 메서드 및 속성을 제공합니다.

IXMLHTTPRequest3Callback HTTP 요청 및 응답 처리에 영향을 주는 이벤트의 미해결 IXMLHTTPRequest3 요청을 사용하여 애플리케이션에 알리는 콜백을 정의합니다.

구조체

XHR_CERT 인코딩된 인증서를 가리키는 버퍼를 정의합니다.
XHR_COOKIE SetCookie 메서드를 호출하여 HTTP 쿠키 jar에 추가하거나 GetCookie 메서드를 호출하여 HTTP 쿠키 jar에서 검색할 수 있는 쿠키를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

XPS 문서

아티클2023. 03. 16.

XPS 문서 기술 개요입니다.

XPS 문서를 개발하려면 다음 헤더가 필요합니다.

- documenttarget.h
- xpsdigitalsignature.h
- xpsobjectmodel.h
- xpsobjectmodel_1.h
- xpsobjectmodel_2.h

이 기술에 대한 프로그래밍 지침은 다음을 참조하세요.

- 문서 및 인쇄

열거형

EDefaultDevmodeType 인쇄 티켓이 가능한 모든 설정을 지정하지 않는 경우 사용자가 기본값의 원본으로 사용할 DEVMODE를 지정할 수 있습니다.
EPrintTicketScope 인쇄 티켓의 범위를 지정합니다.
PrintDocumentPackageCompletion PrintDocumentPackageCompletion 열거형은 인쇄 작업의 상태를 지정합니다.
XPS_COLOR_INTERPOLATION 색 보간에 사용되는 감마 함수에 대해 설명합니다.
XPS_COLOR_TYPE XPS_COLOR 구조체에서 사용하는 색 형식에 대해 설명합니다.
XPS_DASH_CAP 파선 스트로크에서 대시 캡의 스타일을 지정합니다.

XPS_DOCUMENT_TYPE 문서가 serialize된 형식을 나타냅니다.
XPS_FILL_RULE 지정된 점이 기하 도형의 일부인지 여부를 확인하기 위해 복합 셰이프에서 사용하는 규칙입니다.
XPS_FONT_EMBEDDING 글꼴을 포함하는 옵션에 대해 설명합니다.
XPS_IMAGE_TYPE 이미지 유형을 설명합니다.
XPS_INTERLEAVING XPS OM의 콘텐츠를 파일 또는 스트림에 쓸 때 인터리브할지 여부를 지정합니다.
XPS_LINE_CAP 선 세그먼트 대문자의 모양을 지정합니다.
XPS_LINE_JOIN 두 개의 교차 선 세그먼트에 의해 만들어진 조인트 설명입니다.
XPS_OBJECT_TYPE IXpsOMShareable에서 파생된 개체의 형식을 설명합니다.
XPS_SEGMENT_STROKE_PATTERN 그림의 모든 세그먼트, 일부 또는 없음 세그먼트가 스트로크되는지 여부를 나타냅니다.
XPS_SEGMENT_TYPE 선 세그먼트에 대해 설명합니다.
XPS_SIGN_FLAGS 서명하기 전에 태그 호환성 검색을 실행해야 하는지 여부를 지정합니다.
XPS_SIGN_POLICY XPS 문서의 선택적 부분이 있는 경우 서명된 부분을 나타내는 비트 열거자입니다.

XPS_SIGNATURE_STATUS 문서의 디지털 서명 상태를 설명합니다.
XPS_SPREAD_METHOD 스프레드 영역을 채우는 방법을 설명합니다.
XPS_STYLE_SIMULATION 글꼴 또는 문자 모양의 시뮬레이션 스타일을 설명합니다.
XPS_THUMBNAIL_SIZE 썸네일 이미지의 크기를 설명합니다.
XPS_TILE_MODE 타일 브러시의 바둑판식 배열 동작에 대해 설명합니다.

Functions

AddModelTexture 지정된 텍스처 부분 및 스트림에서 새 3D 모델 텍스처를 만듭니다.
Addpage 패키지에서 현재 열려 있는 FixedDocument 부분에 새 FixedPage 파트를 씁니다.
Addresource 패키지에 새 파트 리소스를 만듭니다.
AddSignatureBlock 새 IXpsSignatureBlock 인터페이스를 만들고 서명 블록 컬렉션에 추가합니다.
Append 컬렉션의 끝에 IXpsOMColorProfileResource 인터페이스 포인터를 추가합니다.
Append 컬렉션의 끝에 XPS_DASH 구조를 추가합니다.

Append 사전의 끝에 해당 키와 함께 IXpsOMShareable 인터페이스를 추가합니다.
Append 컬렉션의 끝에 IXpsOMDocument 인터페이스를 추가합니다.
Append 컬렉션의 끝에 IXpsOMFontResource 인터페이스를 추가합니다.
Append 컬렉션의 끝에 IXpsOMGeometryFigure 인터페이스를 추가합니다.
Append 컬렉션의 끝에 IXpsOMGradientStop 인터페이스를 추가합니다.
Append 컬렉션의 끝에 IXpsOMImageResource 인터페이스를 추가합니다.
Append 컬렉션의 끝에 IXpsOMPageReference 인터페이스를 추가합니다.
Append 컬렉션의 끝에 IOpcPartUri 인터페이스를 추가합니다.
Append 컬렉션의 끝에 IXpsOMRemoteDictionaryResource 인터페이스를 추가합니다.
Append 컬렉션의 끝에 IXpsOMSignatureBlockResource 인터페이스를 추가합니다.
Append 컬렉션의 끝에 IXpsOMVisual 인터페이스를 추가합니다.
ApplyEdits 속성 간 유효성 검사를 수행한 다음, 부모 IXpsOMGlyphs 인터페이스에 변경 내용을 복사합니다.

취소 현재 인쇄 작업을 취소합니다.
복제 인터페이스의 전체 복사본을 만듭니다. (IXpsOMCanvas.Clone)
복제 인터페이스의 전체 복사본을 만듭니다. (IXpsOMCoreProperties.Clone)
복제 인터페이스의 전체 복사본을 만듭니다. (IXpsOMDictionary.Clone)
복제 인터페이스의 전체 복사본을 만듭니다. (IXpsOMDocument.Clone)
복제 인터페이스의 전체 복사본을 만듭니다. (IXpsOMGeometry.Clone)
복제 인터페이스의 전체 복사본을 만듭니다. (IXpsOMGeometryFigure.Clone)
복제 인터페이스의 전체 복사본을 만듭니다. (IXpsOMGlyphs.Clone)
복제 IXpsOMGradientStop 인터페이스의 전체 복사본을 만듭니다.
복제 인터페이스의 전체 복사본을 만듭니다. (IXpsOMImageBrush.Clone)
복제 인터페이스의 전체 복사본을 만듭니다. (IXpsOMLinearGradientBrush.Clone)
복제 인터페이스의 전체 복사본을 만듭니다. (IXpsOMMatrixTransform.Clone)

복제 인터페이스의 전체 복사본을 만듭니다. (IXpsOMPage.Clone)
복제 인터페이스의 전체 복사본을 만듭니다. (IXpsOMPageReference.Clone)
복제 인터페이스의 전체 복사본을 만듭니다. (IXpsOMPath.Clone)
복제 인터페이스의 전체 복사본을 만듭니다. (IXpsOMRadialGradientBrush.Clone)
복제 인터페이스의 전체 복사본을 만듭니다. (IXpsOMSolidColorBrush.Clone)
복제 인터페이스의 전체 복사본을 만듭니다. (IXpsOMVisualBrush.Clone)
닫기 패키지의 열린 부분을 닫은 다음 패키지를 닫습니다.
CollectLinkTargets IsHyperlinkTarget 속성이 TRUE로 설정된 모든 문서 하위 트리 개체의 이름을 포함하는 IXpsOMNameCollection 인터페이스를 가져옵니다.
CollectPartResources 페이지와 연결된 모든 부분 기반 리소스 목록을 만듭니다.
ConvertHDPhotoToJpegXR 이미지 리소스를 HD Photo에서 JpegXR로 변환합니다.
ConvertJpegXRToHDPhoto JpegXR에서 HD Photo로 이미지 리소스를 변환합니다.
CreateCanvas 페이지 요소를 그룹화하는 데 사용되는 IXpsOMCanvas 인터페이스를 만듭니다.

CreateColorProfileResource 색 프로필 리소스 스트림에 액세스하는 데 사용되는 IXpsOMColorProfileResource 인터페이스를 만듭니다.
CreateCoreProperties XPS 문서를 설명하는 메타데이터를 포함하는 IXpsOMCoreProperties 인터페이스를 만듭니다.
CreateDictionary 속성 리소스를 공유할 수 있는 IXpsOMDictionary 인터페이스를 만듭니다.
CreateDocument IXpsOMPageReference 인터페이스 집합을 순서가 지정된 시퀀스로 포함할 수 있는 IXpsOMDocument 인터페이스를 만듭니다.
CreateDocumentPackageTargetForPrintJob IPrintDocumentPackageTarget 개체를 만들기 위한 진입점 역할을 합니다.
CreateDocumentSequence XPS 문서의 IXpsOMDocument 인터페이스를 포함할 수 있는 IXpsOMDocumentSequence 인터페이스를 만듭니다.
CreateDocumentStructureResource 문서 구조 리소스 스트림에 대한 액세스를 제공하는 IXpsOMDocumentStructureResource 인터페이스를 만듭니다.
CreateFontResource 글꼴 리소스에 IStream 인터페이스를 제공하는 IXpsOMFontResource 인터페이스를 만듭니다.
CreateGeometry 경로 또는 클리핑 영역의 모양을 지정하는 IXpsOMGeometry 인터페이스를 만듭니다.
CreateGeometryFigure IXpsOMGeometry 인터페이스로 정의된 개체의 일부를 지정하는 IXpsOMGeometryFigure 인터페이스를 만듭니다.
CreateGlyphs 페이지에 나타나는 텍스트를 지정하는 IXpsOMGlyphs 인터페이스를 만듭니다.

CreateGradientStop 그라데이션 내에서 단일 색 및 위치 정의를 나타내는 IXpsOMGradientStop 인터페이스를 만듭니다.
CreateImageBrush IXpsOMImageBrush 인터페이스를 만듭니다.
CreateImageResource 이미지 리소스 스트림에 액세스하는 데 사용되는 IXpsOMImageResource 인터페이스를 만듭니다.
CreateLinearGradientBrush IXpsOMLinearGradientBrush 인터페이스를 만듭니다.
CreateMatrixTransform 아핀 행렬 변환을 지정하는 IXpsOMMatrixTransform 인터페이스를 만듭니다.
CreatePackage XPS 개체 모델 문서 트리의 루트 노드 역할을 하는 IXpsOMPackage 인터페이스를 만듭니다.
CreatePackageFromFile XPS 패키지 파일을 열고 인스턴스화된 XPS 문서 개체 트리를 반환합니다.
CreatePackageFromFile1 XPS 패키지 파일을 열고 인스턴스화된 XPS 문서 개체 트리를 반환합니다. 이 메서드는 XPS 또는 XPS_DOCUMENT_TYPE_ OPENXPS XPS_DOCUMENT_TYPE_ 형식의 XPS 문서가 포함된 파일을 읽습니다.
CreatePackageFromStream XPS 패키지가 포함된 스트림을 열고 인스턴스화된 XPS 문서 개체 트리를 반환합니다.
CreatePackageFromStream1 XPS 패키지가 포함된 스트림을 열고 인스턴스화된 XPS 문서 개체 트리를 반환합니다.
CreatePackageWriterOnFile XPS OM의 내용을 XPS 패키지에 쓰기 위한 파일을 엽니다.

CreatePackageWriterOnFile1 XPS OM의 내용을 지정된 형식의 XPS 패키지에 쓰기 위한 파일을 엽니다. 이 메서드는 MSXPS 문서 또는 OpenXPS 문서에 대한 패키지 작성기를 생성합니다.
CreatePackageWriterOnStream XPS OM의 내용을 XPS 패키지에 쓰기 위한 스트림을 엽니다.
CreatePackageWriterOnStream1 XPS OM의 내용을 지정된 형식의 XPS 패키지에 쓰기 위한 스트림을 엽니다.
CreatePage 단일 페이지의 내용을 나타내는 개체 트리의 루트 노드를 제공하는 IXpsOMPage 인터페이스를 만듭니다.
CreatePageFromStream 지정된 스트림에서 페이지 태그를 읽어 IXpsOMPage 인터페이스를 만들고 채웁니다.
CreatePageFromStream1 지정된 스트림에서 페이지 태그를 읽어 IXpsOMPage1 인터페이스를 만들고 채웁니다.
CreatePageReference 페이지의 가상화를 가능하게 하는 IXpsOMPageReference 인터페이스를 만듭니다.
CreatePartResources 부분 기반 리소스를 포함할 수 있는 IXpsOMPartResources 인터페이스를 만듭니다.
CreatePartUri 지정된 URI를 사용하는 IOpcPartUri 인터페이스를 만듭니다.
CreatePartUriCollection IOpcPartUri 인터페이스 포인터를 포함할 수 있는 IXpsOMPartUriCollection 인터페이스를 만듭니다.
CreatePath 페이지의 그래픽 경로 요소를 지정하는 IXpsOMPath 인터페이스를 만듭니다.

CreatePrintTicketResource PrintTicket 스트림에 액세스할 수 있도록 하는 IXpsOMPrintTicketResource 인터페이스를 만듭니다.
CreateRadialGradientBrush IXpsOMRadialGradientBrush 인터페이스를 만듭니다.
CreateReadOnlyStreamOnFile 지정된 파일에 읽기 전용 IStream을 만듭니다.
CreateRemoteDictionaryResource 속성 리소스를 공유할 수 있도록 하는 IXpsOMRemoteDictionaryResource 인터페이스를 만듭니다.
CreateRemoteDictionaryResourceFromStream 원격 리소스 사전 태그를rooted IXpsOMRemoteDictionaryResource 인터페이스에 로드합니다.
CreateRemoteDictionaryResourceFromStream1 원격 리소스 사전 태그를rooted IXpsOMRemoteDictionaryResource 인터페이스에 로드합니다. dictionaryMarkupStream 매개 변수에서 참조하는 사전은 OpenXPS 또는 MSXPS 네임스페이스의 태그를 포함할 수 있습니다.
CreateRequest 새 IXpsSignatureRequest 인터페이스를 만들고 서명 블록에 추가합니다.
CreateSignatureBlockResource 하나 이상의 서명 요청을 포함할 수 있는 IXpsOMSignatureBlockResource를 만듭니다.
CreateSigningOptions 새 IXpsSigningOptions 인터페이스를 만듭니다.
CreateSolidColorBrush 단일 단색의 브러시를 지정하는 IXpsOMSolidColorBrush 인터페이스를 만듭니다.
CreateStoryFragmentsResource 페이지의 StoryFragments 부분의 리소스 스트림 콘텐츠에 대한 액세스를 제공하는 IXpsOMStoryFragmentsResource 인터페이스를 만듭니다.

CreateVisualBrush 시각적 개체를 사용하는 IXpsOMTileBrush인 IXpsOMVisualBrush 인터페이스를 만듭니다.
CreateXpsOMPackageWriter StartXpsPrintJob1 함수가 만든 인쇄 작업에 사용할 IXpsOMPackageWriter 인터페이스를 만듭니다.
DiscardPage 메모리에서 페이지를 삭제합니다.
GenerateThumbnail 페이지의 썸네일 이미지를 생성합니다.
GenerateUnusedLookupKey 리소스 사전의 리소스에서 조회 키로 사용할 수 있는 고유한 이름을 생성합니다.
GetAccessibilityLongDescription 개체 내용에 대한 긴(자세한) 텍스트 설명을 가져옵니다. (IXpsOMCanvas.GetAccessibilityLongDescription)
GetAccessibilityLongDescription 개체 내용에 대한 긴(자세한) 텍스트 설명을 가져옵니다. (IXpsOMPath.GetAccessibilityLongDescription)
GetAccessibilityShortDescription 개체 내용에 대한 간단한 텍스트 설명을 가져옵니다.
GetAccessibilityShortDescription 개체 내용에 대한 간단한 텍스트 설명을 가져옵니다.
AdvisoryPageDimensions 가져오기 페이지의 제안된 차원을 가져옵니다.
Getat 컬렉션의 지정된 위치에서 IXpsOMColorProfileResource 인터페이스 포인터를 가져옵니다.
Getat 컬렉션의 지정된 위치에서 XPS_DASH 구조를 가져옵니다.

Getat 사전의 지정된 인덱스에 있는 항목의 IXpsOMShareable 인터페이스 포인터 및 키 이름 문자열을 가져옵니다.
Getat 컬렉션의 지정된 위치에서 IXpsOMDocument 인터페이스 포인터를 가져옵니다.
Getat 컬렉션의 지정된 위치에서 IXpsOMFontResource 인터페이스 포인터를 가져옵니다.
Getat 컬렉션의 지정된 위치에서 IXpsOMGeometryFigure 인터페이스 포인터를 가져옵니다.
Getat 컬렉션의 지정된 위치에서 IXpsOMGradientStop 인터페이스 포인터를 가져옵니다.
Getat 컬렉션의 지정된 위치에서 IXpsOMImageResource 인터페이스 포인터를 가져옵니다.
Getat 컬렉션의 지정된 위치에 저장된 이름 문자열을 가져옵니다.
Getat 컬렉션의 지정된 위치에서 IXpsOMPageReference 인터페이스 포인터를 가져옵니다.
Getat 컬렉션의 지정된 위치에서 IOpcPartUri 인터페이스 포인터를 가져옵니다.
Getat 컬렉션의 지정된 위치에서 IXpsOMRemoteDictionaryResource 인터페이스 포인터를 가져옵니다.
Getat 컬렉션의 지정된 위치에서 IXpsOMSignatureBlockResource 인터페이스 포인터를 가져옵니다.
Getat 컬렉션의 지정된 위치에서 IXpsOMVisual 인터페이스 포인터를 가져옵니다.

Getat 컬렉션의 지정된 위치에서 IXpsSignatureBlock 인터페이스 포인터를 가져옵니다.
Getat 컬렉션의 지정된 위치에서 IXpsSignature 인터페이스 포인터를 가져옵니다.
Getat 컬렉션의 지정된 위치에서 IXpsSignatureRequest 인터페이스 포인터를 가져옵니다.
GetBidiLevel 양방향 텍스트의 수준을 가져옵니다.
GetBidiLevel 부모 IXpsOMGlyphs 인터페이스의 양방향 텍스트 수준을 가져옵니다.
GetBleedBox 페이지의 블리드 상자의 크기를 가져옵니다.
GetByKey 지정된 키를 포함하는 항목의 IXpsOMShareable 인터페이스 포인터를 가져옵니다.
GetByPartName 인터페이스의 파트 이름과 일치하여 컬렉션에서 IXpsOMColorProfileResource 인터페이스 포인터를 가져옵니다.
GetByPartName 인터페이스의 파트 이름과 일치하여 컬렉션에서 IXpsOMFontResource 인터페이스 포인터를 가져옵니다.
GetByPartName 인터페이스의 파트 이름과 일치하여 컬렉션에서 IXpsOMImageResource 인터페이스 포인터를 가져옵니다.
GetByPartName 인터페이스의 파트 이름과 일치하여 컬렉션에서 IXpsOMRemoteDictionaryResource 인터페이스 포인터를 가져옵니다.

GetByPartName 인터페이스의 파트 이름과 일치하여 컬렉션에서 IXpsOMSignatureBlockResource 인터페이스 포인터를 가져옵니다.
GetCategory 범주 속성을 가져옵니다.
GetCenter 방사형 그라데이션 영역 줄임표의 중심점을 가져옵니다.
GetCertificateEnumerator 서명에 연결된 패키지 인증서를 열거하는 IOpcCertificateEnumerator 인터페이스에 대한 포인터를 가져옵니다.
GetCertificateSet 서명에 인증서를 추가하는 데 사용할 수 있는 IOpcCertificateSet 인터페이스를 가져옵니다.
GetClipGeometry 시각적 개체의 클리핑 영역의 확인된 기하 도형을 포함하는 IXpsOMGeometry 인터페이스에 대한 포인터를 가져옵니다.
GetClipGeometryLocal 시각적 개체의 클리핑 영역의 공유되지 않은 로컬 기하 도형을 포함하는 IXpsOMGeometry 인터페이스에 대한 포인터를 가져옵니다.
GetClipGeometryLookup 시각적 개체의 클리핑 영역을 포함하는 리소스 사전에서 IXpsOMGeometry 인터페이스에 대한 조회 키를 가져옵니다.
GetColor 그라데이션 중지점의 색 값 및 색 프로필을 가져옵니다.
GetColor 브러시의 색 값 및 색 프로필을 가져옵니다.
GetColorInterpolationMode 색 보간에 사용할 감마 함수를 가져옵니다.

GetColorProfileResource 이미지와 연결된 색 프로필 리소스를 포함하는 IXpsOMColorProfileResource 인터페이스에 대한 포인터를 가져옵니다.
GetColorProfileResources XPS 문서에서 사용되는 색 프로필을 포함하는 IXpsOMColorProfileResourceCollection 인터페이스를 가져옵니다.
GetContentBox 페이지 콘텐츠 상자의 크기를 가져옵니다.
GetContentStatus contentStatus 속성을 가져옵니다.
GetContentType contentType 속성을 가져옵니다.
GetCoreProperties XPS 패키지의 IXpsOMCoreProperties 인터페이스에 대한 포인터를 가져옵니다.
GetCount 컬렉션의 IXpsOMColorProfileResource 인터페이스 포인터 수를 가져옵니다.
GetCount 컬렉션의 XPS_DASH 구조체 수를 가져옵니다.
GetCount 사전에 있는 엔트리의 수를 가져옵니다.
GetCount 컬렉션의 IXpsOMDocument 인터페이스 포인터 수를 가져옵니다.
GetCount 컬렉션의 IXpsOMFontResource 인터페이스 포인터 수를 가져옵니다.
GetCount 컬렉션의 IXpsOMGeometryFigure 인터페이스 포인터 수를 가져옵니다.

GetCount 컬렉션의 IXpsOMGradientStop 인터페이스 포인터 수를 가져옵니다.
GetCount 컬렉션의 IXpsOMImageResource 인터페이스 포인터 수를 가져옵니다.
GetCount 컬렉션의 이름 문자열 수를 가져옵니다.
GetCount 컬렉션의 IXpsOMPageReference 인터페이스 포인터 수를 가져옵니다.
GetCount 컬렉션의 IOpcPartUri 인터페이스 포인터 수를 가져옵니다.
GetCount 컬렉션의 IXpsOMRemoteDictionaryResource 인터페이스 포인터 수를 가져옵니다.
GetCount 컬렉션의 IXpsOMSignatureBlockResource 인터페이스 포인터 수를 가져옵니다.
GetCount 컬렉션의 IXpsOMVisual 인터페이스 포인터 수를 가져옵니다.
GetCount 컬렉션의 IXpsSignatureBlock 인터페이스 포인터 수를 가져옵니다.
GetCount 컬렉션의 IXpsSignature 인터페이스 포인터 수를 가져옵니다.
GetCount 컬렉션의 IXpsSignatureRequest 인터페이스 포인터 수를 가져옵니다.
GetCreated 만든 속성을 가져옵니다.

GetCreator creator 속성을 가져옵니다.
GetCustomObjectEnumerator 서명의 사용자 지정 개체를 열거하는 IOpcSignatureCustomObjectEnumerator 인터페이스에 대한 포인터를 가져옵니다.
GetCustomObjects 서명 사용자 지정 개체 집합이 포함된 IOpcSignatureCustomObjectSet 인터페이스에 대한 포인터를 가져옵니다.
GetCustomReferenceEnumerator 서명의 사용자 지정 참조를 열거하는 IOpcSignatureReferenceEnumerator 인터페이스에 대한 포인터를 가져옵니다.
GetCustomReferences 서명 사용자 지정 참조 집합을 포함하는 IOpcSignatureReferenceSet 인터페이스에 대한 포인터를 가져옵니다.
GetDescription 설명 속성을 가져옵니다.
GetDeviceFontName 디바이스 글꼴의 이름을 가져옵니다. (IXpsOMGlyphs.GetDeviceFontName)
GetDeviceFontName 디바이스 글꼴의 이름을 가져옵니다. (IXpsOMGlyphsEditor.GetDeviceFontName)
GetDictionary 캔버스와 연결된 사전의 확인된 IXpsOMDictionary 인터페이스에 대한 포인터를 가져옵니다.
GetDictionary 이 페이지와 연결된 확인된 IXpsOMDictionary 인터페이스에 대한 포인터를 가져옵니다.
GetDictionary 이 리소스와 연결된 원격 사전의 IXpsOMDictionary 인터페이스에 대한 포인터를 가져옵니다.

GetDictionaryLocal 공유되지 않은 로컬 사전의 IXpsOMDictionary 인터페이스에 대한 포인터를 가져옵니다.
GetDictionaryLocal 이 페이지와 연결된 공유되지 않은 로컬 사전의 IXpsOMDictionary 인터페이스에 대한 포인터를 가져옵니다.
GetDictionaryResource 원격 사전 리소스의 IXpsOMRemoteDictionaryResource 인터페이스에 대한 포인터를 가져옵니다.
GetDictionaryResource 이 페이지에서 사용되는 공유 사전 리소스의 IXpsOMRemoteDictionaryResource 인터페이스에 대한 포인터를 가져옵니다.
GetDigestMethod 현재 다이제스트 메서드를 가져옵니다.
GetDiscardControlPartName XPS 패키지에 있는 삭제 컨트롤 파트의 이름을 가져옵니다.
GetDocumentName 문서 파트의 URI를 포함하는 IOpcPartUri 인터페이스에 대한 포인터를 가져옵니다.
GetDocuments 문서 시퀀스에 지정된 문서를 포함하는 IXpsOMDocumentCollection 인터페이스에 대한 포인터를 가져옵니다.
GetDocumentSequence XPS 패키지의 문서 시퀀스를 포함하는 IXpsOMDocumentSequence 인터페이스에 대한 포인터를 가져옵니다.
GetDocumentStructureResource 문서에 대한 구조적 정보를 포함하는 리소스의 IXpsOMDocumentStructureResource 인터페이스에 대한 포인터를 가져옵니다.
GetDocumentType 이 패키지를 초기화하는 데 사용된 데이터의 문서 형식을 가져옵니다. 이 메서드는 문서가 XPS 또는 OpenXPS 형식인지 여부를 확인하는 데 사용됩니다. 자세한 내용은 XPS 문서를 참조하세요.

GetDocumentType 이 페이지를 초기화하는 데 사용된 FixedPage 태그 유형을 가져옵니다. 이 메서드는 문서가 XPS 또는 OpenXPS 형식인지 여부를 확인하는 데 사용됩니다. 자세한 내용은 XPS 문서를 참조하세요.
GetDocumentType 리소스의 XPS_DOCUMENT_TYPE 가져옵니다.
GetDocumentTypeFromFile 지정된 파일에 저장된 XPS 문서의 형식을 검색합니다.
GetDocumentTypeFromStream 지정된 스트림에 저장된 XPS 문서의 형식을 검색합니다.
GetEmbeddingOption 리소스가 serialize될 때 적용될 포함 옵션을 가져옵니다.
GetEndPoint 그라데이션의 끝점을 가져옵니다.
GetFigures 이 기하 도형을 구성하는 그림의 컬렉션을 포함하는 기하 도형의 IXpsOMGeometryFigureCollection 인터페이스에 대한 포인터를 가져옵니다.
GetFillBrush 텍스트에 사용할 채우기 브러시의 확인된 IXpsOMBrush 인터페이스에 대한 포인터를 가져옵니다.
GetFillBrush 경로에 대한 채우기 브러시가 포함된 확인된 IXpsOMBrush 인터페이스에 대한 포인터를 가져옵니다.
GetFillBrushLocal 텍스트에 사용할 채우기 브러시의 공유되지 않은 로컬 IXpsOMBrush 인터페이스에 대한 포인터를 가져옵니다.
GetFillBrushLocal 경로에 대한 채우기 브러시가 포함된 공유되지 않은 로컬 IXpsOMBrush 인터페이스에 대한 포인터를 가져옵니다.

GetFillBrushLookup 리소스 사전에 저장되고 채우기 브러시로 사용되는 IXpsOMBrush 인터페이스의 조회 키를 가져옵니다.
GetFillBrushLookup 리소스 사전에 저장되고 경로의 채우기 브러시로 사용되는 브러시의 조회 키를 가져옵니다.
GetFillRule 사용할 채우기 규칙을 설명하는 XPS_FILL_RULE 값을 가져옵니다.
GetFlags 이 서명에 사용할 서명 플래그를 지정하는 XPS_SIGN_FLAGS 값을 가져옵니다.
GetFontRenderingEmSize 글꼴 크기를 가져옵니다. (IXpsOMGlyphs.GetFontRenderingEmSize)
GetFontResource 이 텍스트에 필요한 글꼴 리소스 개체의 IXpsOMFontResource 인터페이스에 대한 포인터를 가져옵니다.
GetFontResources XPS 문서에서 사용되는 글꼴을 포함하는 IXpsOMFontResourceCollection 인터페이스를 가져옵니다.
GetGeometry 이 경로의 확인된 채우기 영역을 설명하는 경로의 IXpsOMGeometry 인터페이스에 대한 포인터를 가져옵니다.
GetGeometryLocal 이 경로에 대해 확인된 채우기 영역의 공유되지 않은 로컬 기하 도형을 가져옵니다.
GetGeometryLookup 리소스 사전에 저장되고 이 경로의 확인된 채우기 영역을 설명하는 공유 기하 도형 개체의 조회 키를 가져옵니다.
GetGlyphIndexCount 문자 모양 인덱스 수를 가져옵니다.

GetGlyphIndexCount 문자 모양 인덱스 수를 가져옵니다.
GetGlyphIndices 글꼴의 특정 문자 모양 인덱스를 설명하는 XPS_GLYPH_INDEX 구조체 배열을 가져옵니다. (IXpsOMGlyphs.GetGlyphIndices)
GetGlyphIndices 글꼴의 특정 문자 모양 인덱스를 설명하는 XPS_GLYPH_INDEX 구조체 배열을 가져옵니다. (IXpsOMGlyphsEditor.GetGlyphIndices)
GetGlyphMappingCount 문자 모양 매핑 수를 가져옵니다. (IXpsOMGlyphs.GetGlyphMappingCount)
GetGlyphMappingCount 문자 모양 매핑 수를 가져옵니다. (IXpsOMGlyphsEditor.GetGlyphMappingCount)
GetGlyphMappings GETGlyphIndices에서 반환되는 XPS_GLYPH_INDEX 구조체 배열의 항목에 UTF-16 스칼라 값을 매핑하는 방법을 설명하는 XPS_GLYPH_MAPPING 구조체의 배열을 가져옵니다. (IXpsOMGlyphs.GetGlyphMappings)
GetGlyphMappings GETGlyphIndices에서 반환되는 XPS_GLYPH_INDEX 구조체 배열의 항목에 UTF-16 스칼라 값을 매핑하는 방법을 설명하는 XPS_GLYPH_MAPPING 구조체의 배열을 가져옵니다. (IXpsOMGlyphsEditor.GetGlyphMappings)
GetGlyphsEditor 개체의 문자 모양을 편집하는 데 사용할 IXpsOMGlyphsEditor 인터페이스에 대한 포인터를 가져옵니다.
GetGradientOrigin 방사형 그라데이션의 원점을 가져옵니다.
GetGradientStops 그라데이션을 정의하는 IXpsOMGradientStop 인터페이스의 컬렉션을 포함하는 IXpsOMGradientStopCollection 인터페이스에 대한 포인터를 가져옵니다.

GetHyperlinkNavigateUri 이 시각적 개체가 연결되는 IUri 인터페이스에 대한 포인터를 가져옵니다.
GetIdentifier 식별자 속성을 가져옵니다.
GetImageResource 브러시의 소스로 사용할 이미지 리소스를 포함하는 IXpsOMImageResource 인터페이스에 대한 포인터를 가져옵니다.
GetImageResources XPS 문서에 사용되는 이미지가 포함된 IXpsOMImageResourceCollection 인터페이스를 가져옵니다.
GetImageType 이미지 리소스의 형식을 가져옵니다.
GetIntent 서명의 의도 또는 의미를 설명하는 문자열을 설정합니다. (IXpsSignatureRequest.GetIntent)
GetIsClosed 그림이 닫혀 있는지 여부를 나타내는 값을 가져옵니다.
GetIsFilled 그림이 채워지는지 여부를 나타내는 값을 가져옵니다.
GetIsHyperlinkTarget 페이지가 하이퍼링크의 대상인지 여부를 나타내는 부울 값을 가져옵니다.
GetIsHyperlinkTarget 시각적 개체가 하이퍼링크의 대상인지 여부를 나타내는 값을 가져옵니다.
GetIsSideways 텍스트를 옆으로 회전된 문자 모양으로 렌더링할지 여부를 나타내는 부울 값을 가져옵니다. (IXpsOMGlyphs.GetIsSideways)

GetIsSideways 텍스트를 옆으로 회전된 문자 모양으로 렌더링할지 여부를 나타내는 부울 값을 가져옵니다. (IXpsOMGlyphsEditor.GetIsSideways)
GetKeywords 키워드 속성을 가져옵니다.
GetLanguage 언어 속성을 가져옵니다.
GetLanguage 페이지의 Language 속성을 가져옵니다.
GetLanguage 시각적 개체 및 해당 콘텐츠의 Language 속성을 가져옵니다.
GetLastModifiedBy lastModifiedBy 속성을 가져옵니다.
GetLastPrinted lastPrinted 속성을 가져옵니다.
GetMatrix 변환 행렬을 지정하는 XPS_MATRIX 구조를 가져옵니다.
GetModified 수정된 속성을 가져옵니다.
GetName 페이지의 Name 속성을 가져옵니다.
GetName 시각적 개체의 Name 속성을 가져옵니다.
GetOffset 그라데이션 중지점의 오프셋 값을 가져옵니다.

GetOpacity 브러시의 불투명도를 가져옵니다.
GetOpacity 이 시각적 개체의 불투명도 값을 가져옵니다.
GetOpacityMaskBrush 시각적 개체의 불투명 마스크 브러시의 IXpsOMBrush 인터페이스에 대한 포인터를 가져옵니다.
GetOpacityMaskBrushLocal 시각적 개체에 대한 공유되지 않은 로컬 불투명 마스크 브러시를 가져옵니다.
GetOpacityMaskBrushLookup 리소스 사전에서 공유 불투명 마스크 브러시의 조회 키 이름을 가져옵니다.
GetOrigin 텍스트의 시작 위치를 가져옵니다.
GetOwner 핵심 속성을 포함하는 IXpsOMPackage 인터페이스에 대한 포인터를 가져옵니다.
GetOwner 사전이 포함된 인터페이스에 대한 포인터를 가져옵니다.
GetOwner 문서를 포함하는 IXpsOMDocumentSequence 인터페이스에 대한 포인터를 가져옵니다.
GetOwner 문서 시퀀스가 포함된 IXpsOMPackage 인터페이스에 대한 포인터를 가져옵니다.
GetOwner 리소스를 포함하는 IXpsOMDocument 인터페이스에 대한 포인터를 가져옵니다. (IXpsOMDocumentStructureResource.GetOwner)
GetOwner 기하 도형을 포함하는 IXpsOMGeometry 인터페이스에 대한 포인터를 가져옵니다.

GetOwner 그라데이션 중지점이 포함된 IXpsOMGradientBrush 인터페이스에 대한 포인터를 가져옵니다.
GetOwner 페이지가 포함된 IXpsOMPageReference 인터페이스에 대한 포인터를 가져옵니다.
GetOwner 페이지 참조가 포함된 IXpsOMDocument 인터페이스에 대한 포인터를 가져옵니다.
GetOwner 부모의 IUnknown 인터페이스를 가져옵니다.
GetOwner 리소스를 포함하는 IXpsOMDocument 인터페이스에 대한 포인터를 가져옵니다. (IXpsOMSignatureBlockResource.GetOwner)
GetOwner 이 리소스를 포함하는 IXpsOMPage 인터페이스에 대한 포인터를 가져옵니다.
GetPackageTarget 클라이언트가 지정된 대상 형식의 문서를 추가할 수 있는 특정 문서 패키지 대상에 대한 포인터를 검색합니다. 클라이언트는 이 메서드를 여러 번 호출할 수 있지만 항상 동일한 대상 ID를 사용해야 합니다.
GetPackageTargetTypes 지원되는 대상 형식을 열거합니다.
GetPage 페이지가 포함된 IXpsOMPage 인터페이스에 대한 포인터를 가져옵니다.
GetPageDimensions 페이지 차원을 가져옵니다.
GetPageReferences 페이지의 가상화된 액세스를 허용하는 문서의 IXpsOMPageReferenceCollection 인터페이스를 가져옵니다.

GetPartName 파트가 serialize될 때 사용할 이름을 가져옵니다.
GetPartName SignatureDefinitions 부분의 URI를 포함하는 IOpcPartUri 인터페이스에 대한 포인터를 가져옵니다.
GetPolicy 서명을 만들 때 사용되는 서명 정책을 나타내는 XPS_SIGN_POLICY 값을 가져옵니다.
GetPolicy 서명 정책을 지정하는 XPS_SIGN_POLICY 값을 가져옵니다.
GetPrintTicketResource 문서 수준 인쇄 티켓의 IXpsOMPrintTicketResource 인터페이스를 가져옵니다.
GetPrintTicketResource 문서 시퀀스에 할당된 작업 수준 인쇄 티켓에 대한 IXpsOMPrintTicketResource 인터페이스를 가져옵니다.
GetPrintTicketResource 페이지와 연결된 페이지 수준 인쇄 티켓 리소스의 IXpsOMPrintTicketResource 인터페이스에 대한 포인터를 가져옵니다.
GetProhibitedCaretStopCount 금지된 케어 중지 횟수를 가져옵니다. (IXpsOMGlyphs.GetProhibitedCaretStopCount)
GetProhibitedCaretStopCount 금지된 케어 중지 횟수를 가져옵니다. (IXpsOMGlyphsEditor.GetProhibitedCaretStopCount)
GetProhibitedCaretStops 금지된 케어 중지 위치의 배열을 가져옵니다. (IXpsOMGlyphs.GetProhibitedCaretStops)
GetProhibitedCaretStops 금지된 케어 중지 위치의 배열을 가져옵니다. (IXpsOMGlyphsEditor.GetProhibitedCaretStops)
GetRadiiSizes 방사형 그라데이션 영역의 타원을 정의하는 반지름의 크기를 가져옵니다.

GetRemoteDictionaryResources XPS 문서에서 사용되는 원격 리소스 사전이 포함된 IXpsOMRemoteDictionaryResourceCollection 인터페이스를 가져옵니다.
GetRequestedSigner 서명했거나 패키지에 서명하도록 요청하는 사용자의 ID를 가져옵니다.
GetRequestId 서명 요청의 고유 식별자를 가져옵니다.
GetRequests 서명 요청 컬렉션이 포함된 IXpsSignatureRequestCollection 인터페이스에 대한 포인터를 가져옵니다.
GetRequestSignByDate 요청된 서명자가 문서의 지정된 부분에 서명해야 하는 날짜와 시간을 가져옵니다.
GetRevision 수정 속성을 가져옵니다.
GetSegmentCount 그림의 세그먼트 수를 가져옵니다.
GetSegmentData 기하 도형의 세그먼트 데이터 요소를 가져옵니다.
GetSegmentDataCount 그림의 세그먼트 데이터 요소 수를 가져옵니다.
GetSegmentStrokePattern 그림의 세그먼트가 스트로크되는지 여부를 나타내는 XPS_SEGMENT_STROKE_PATTERN 값을 가져옵니다.
GetSegmentStrokes 그림의 세그먼트에 대한 스트로크 정의를 가져옵니다.

GetSegmentTypes 그림의 세그먼트 형식을 가져옵니다.
GetSignature 서명 요청과 동일한 고유 식별자를 가진 XPS 디지털 서명을 포함하는 IXpsSignature 인터페이스에 대한 포인터를 가져옵니다.
GetSignatureBlockResources 문서의 디지털 서명 블록 리소스 컬렉션을 참조하는 IXpsOMSignatureBlockResourceCollection 인터페이스에 대한 포인터를 가져옵니다.
GetSignatureBlocks 서명 블록 컬렉션이 포함된 IXpsSignatureBlockCollection 인터페이스에 대한 포인터를 가져옵니다.
GetSignatureId Signature 요소의 Id 특성 값을 가져옵니다. (IXpsSignature.GetSignatureId)
GetSignatureId Signature 요소의 Id 특성 값을 가져옵니다. (IXpsSigningOptions.GetSignatureId)
GetSignatureMethod 서명 메서드를 가져옵니다.
GetSignatureOriginPartName 서명 원본 파트의 파트 이름을 가져옵니다.
GetSignaturePartName 서명 파트의 파트 이름을 가져옵니다.
GetSignaturePartName 문서 서명 파트의 파트 이름을 가져옵니다.
GetSignatures XPS 디지털 서명 컬렉션이 포함된 IXpsSignatureCollection 인터페이스에 대한 포인터를 가져옵니다.

GetSignatureValue 서명의 암호화된 해시 값을 가져옵니다.
GetSignatureXml 디지털 서명의 XML 태그를 가져옵니다.
GetSigningLocale 패키지 서명 위치의 법적 관할권을 가져옵니다.
GetSigningTime 서명 생성 날짜 및 시간을 가져옵니다.
GetSigningTimeFormat 서명 시간의 형식을 가져옵니다.
GetSigningTimeFormat 서명 시간 문자열의 형식을 가져옵니다.
GetSnapsToPixels 경로가 렌더링될 때 경로를 디바이스 픽셀에 스냅할지 여부를 나타내는 부울 값을 가져옵니다.
GetSpotLocation 표시되는 디지털 서명 또는 디지털 서명 요청이 표시되는 페이지의 페이지 및 위치를 가져옵니다.
GetSpreadMethod 그라데이션 영역 외부의 영역을 렌더링하는 방법을 설명하는 XPS_SPREAD_METHOD 값을 가져옵니다.
GetStartPoint 그림의 시작점을 가져옵니다.
GetStartPoint 그라데이션의 시작점을 가져옵니다.
GetStoryFragmentsResource 페이지와 연결된 StoryFragments 파트 리소스의 IXpsOMStoryFragmentsResource 인터페이스에 대한 포인터를 가져옵니다.

GetStream 이 리소스와 연결된 스트림의 읽기 전용인 새 복사본을 가져옵니다. (IXpsOMColorProfileResource.GetStream)
GetStream 이 리소스와 연결된 스트림의 읽기 전용인 새 복사본을 가져옵니다. (IXpsOMDocumentStructureResource.GetStream)
GetStream 이 리소스와 연결된 스트림의 읽기 전용인 새 복사본을 가져옵니다. (IXpsOMFontResource.GetStream)
GetStream 이 리소스와 연결된 스트림의 읽기 전용인 새 복사본을 가져옵니다. (IXpsOMImageResource.GetStream)
GetStream 이 리소스와 연결된 스트림의 읽기 전용인 새 복사본을 가져옵니다. (IXpsOMPrintTicketResource.GetStream)
GetStream 이 리소스와 연결된 스트림의 읽기 전용인 새 복사본을 가져옵니다. (IXpsOMSignatureBlockResource.GetStream)
GetStream 이 리소스와 연결된 스트림의 읽기 전용인 새 복사본을 가져옵니다. (IXpsOMStoryFragmentsResource.GetStream)
GetStrokeBrush 경로에 대해 설정된 스트로크 브러시가 포함된 확인된 IXpsOMBrush 인터페이스에 대한 포인터를 가져옵니다.
GetStrokeBrushLocal 경로에 대한 스트로크 브러시가 포함된 공유되지 않은 로컬 IXpsOMBrush 인터페이스에 대한 포인터를 가져옵니다.
GetStrokeBrushLookup 리소스 사전에 저장되고 경로에 대한 스트로크 브러시로 사용할 브러시의 조회 키를 가져옵니다.

GetStrokeDashCap 스트로크 대시에 사용할 끝 캡의 스타일을 가져옵니다.
GetStrokeDashes 스트로크의 대시 패턴을 정의하는 XPS_DASH 구조를 포함하는 IXpsOMDashCollection 인터페이스에 대한 포인터를 가져옵니다.
GetStrokeDashOffset 스트로크의 원점에서 대시 배열 패턴의 시작 지점까지의 오프셋을 가져옵니다.
GetStrokeEndLineCap 스트로크 선의 끝 캡 스타일을 가져옵니다.
GetStrokeLineJoin 스트로크 선을 조인하는 스타일을 가져옵니다.
GetStrokeMiterLimit 스트로크에 대해 설정된 마이터 제한 값을 가져옵니다.
GetStrokeStartLineCap 스트로크 선의 시작 부분에 있는 선 캡의 스타일을 가져옵니다.
GetStrokeThickness 스트로크 두께를 가져옵니다.
GetStyleSimulations 문자 모양을 렌더링할 때 적용될 스타일 시뮬레이션을 가져옵니다.
GetSubject 주체 속성을 가져옵니다.
GetThumbnailResource XPS 패키지와 연결된 썸네일 리소스의 IXpsOMImageResource 인터페이스에 대한 포인터를 가져옵니다.

GetThumbnailResource 페이지와 연결된 썸네일 이미지 리소스의 IXpsOMImageResource 인터페이스에 대한 포인터를 가져옵니다.
GetTileMode 브러시의 타일 모드를 설명하는 XPS_TILE_MODE 값을 가져옵니다.
GetTitle 제목 속성을 가져옵니다.
GetTransform 기하 도형의 확인된 행렬 변환을 포함하는 기하 도형의 IXpsOMMatrixTransform 인터페이스에 대한 포인터를 가져옵니다.
GetTransform 브러시에 대해 확인된 행렬 변환을 포함하는 IXpsOMMatrixTransform 인터페이스에 대한 포인터를 가져옵니다. (IXpsOMGradientBrush.GetTransform)
GetTransform 브러시에 대해 확인된 행렬 변환을 포함하는 IXpsOMMatrixTransform 인터페이스에 대한 포인터를 가져옵니다. (IXpsOMTileBrush.GetTransform)
GetTransform 시각적 개체의 확인된 행렬 변환을 포함하는 IXpsOMMatrixTransform 인터페이스에 대한 포인터를 가져옵니다.
GetTransformLocal 기하 도형에 대한 공유되지 않은 로컬 행렬 변환을 포함하는 IXpsOMMatrixTransform 인터페이스에 대한 포인터를 가져옵니다.
GetTransformLocal 브러시에 대한 공유되지 않은 로컬 확인 행렬 변환을 포함하는 IXpsOMMatrixTransform 인터페이스에 대한 포인터를 가져옵니다.
GetTransformLocal 브러시에 대한 공유되지 않은 로컬 확인 행렬 변환을 포함하는 IXpsOMMatrixTransform 인터페이스에 대한 포인터를 가져옵니다.

GetTransformLocal 시각적 개체에 대한 공유되지 않은 로컬 확인 행렬 변환을 포함하는 IXpsOMMatrixTransform 인터페이스에 대한 포인터를 가져옵니다.
GetTransformLookup 기하 도형에 대해 확인된 행렬 변환을 포함하는 IXpsOMMatrixTransform 인터페이스에 대한 조회 키를 가져옵니다.
GetTransformLookup 브러시에 사용할 공유 행렬 변환 인터페이스의 조회 키 이름을 가져옵니다.
GetTransformLookup 브러시에 대해 확인된 행렬 변환이 포함된 리소스 사전에서 IXpsOMMatrixTransform 인터페이스를 식별하는 조회 키를 가져옵니다.
GetTransformLookup 시각적 개체에 대해 확인된 행렬 변환이 포함된 리소스 사전에서 IXpsOMMatrixTransform 인터페이스의 조회 키 이름을 가져옵니다.
GetType 인터페이스의 개체 형식을 가져옵니다.
GetUnicodeString 이스케이프되지 않은 UTF-16 스칼라 값의 텍스트를 가져옵니다. (IXpsOMGlyphs.GetUnicodeString)
GetUnicodeString 이스케이프되지 않은 UTF-16 스칼라 값의 텍스트를 가져옵니다. (IXpsOMGlyphsEditor.GetUnicodeString)
GetUseAliasedEdgeMode 별칭이 지정된 에지 모드를 사용하여 캔버스에 있는 개체의 가장자리를 렌더링할지 여부를 결정하는 부울 값을 가져옵니다.
GetVersion 버전 속성을 가져옵니다.

GetViewbox 타일에서 사용할 원본 이미지 부분을 가져옵니다.
GetViewport 단일 타일로 덮인 대상 기하 도형의 부분을 가져옵니다.
GetVisual 브러시의 소스로 사용할 확인된 시각적 개체의 인터페이스에 대한 포인터를 가져옵니다.
GetVisualLocal 브러시의 소스로 사용되는 공유되지 않은 로컬 시각적 개체의 인터페이스에 대한 포인터를 가져옵니다.
GetVisualLookup 리소스 사전에서 시각적 개체의 조회 키 이름을 가져옵니다. 시각적 개체는 브러시의 소스로 사용됩니다.
GetVisuals 캔버스의 시각적 개체 컬렉션을 포함하는 IXpsOMVisualCollection 인터페이스에 대한 포인터를 가져옵니다.
GetVisuals 페이지의 시각적 개체 컬렉션을 포함하는 IXpsOMVisualCollection 인터페이스에 대한 포인터를 가져옵니다.
GetXpsOMFactory 문서 패키지에 대한 IXpsOMObjectFactory 개체를 가져옵니다. (IXpsDocumentPackageTarget.GetXpsOMFactory)
GetXpsOMFactory 문서 패키지에 대한 IXpsOMObjectFactory 개체를 가져옵니다. (IXpsDocumentPackageTarget3D.GetXpsOMFactory)
GetXpsOMPackageWriter 문서 패키지의 IXpsOMPackageWriter 개체를 가져옵니다.
GetXpsOMPackageWriter3D 문서 패키지에 대한 새 IXpsOMPackageWriter3D 개체를 가져옵니다.

GetXpsType 문서 패키지의 XPS_DOCUMENT_TYPE 열거형 값을 가져옵니다.
HasRestrictedFonts 참조된 페이지의 문서 하위 트리에 EmbeddingOption 속성이 XPS_FONT_EMBEDDING_RESTRICTED 설정된 글꼴 리소스가 있는 문자 모양이 포함되어 있는지 여부를 나타내는 부울 값을 가져옵니다.
InsertAt 컬렉션의 지정된 위치에 IXpsOMColorProfileResource 인터페이스 포인터를 삽입합니다.
InsertAt 컬렉션의 지정된 위치에 XPS_DASH 구조를 삽입합니다.
InsertAt 사전의 지정된 위치에 IXpsOMShareable 인터페이스를 삽입하고 인터페이스를 식별하는 키를 설정합니다.
InsertAt 컬렉션의 지정된 위치에 IXpsOMDocument 인터페이스 포인터를 삽입합니다.
InsertAt 컬렉션의 지정된 위치에 IXpsOMFontResource 인터페이스 포인터를 삽입합니다.
InsertAt 컬렉션의 지정된 위치에 IXpsOMGeometryFigure 인터페이스 포인터를 삽입합니다.
InsertAt 컬렉션의 지정된 위치에 IXpsOMGradientStop 인터페이스 포인터를 삽입합니다.
InsertAt 컬렉션의 지정된 위치에 IXpsOMImageResource 인터페이스 포인터를 삽입합니다.
InsertAt 컬렉션의 지정된 위치에 IXpsOMPageReference 인터페이스 포인터를 삽입합니다.

InsertAt 컬렉션의 지정된 위치에 IOpcPartUri 인터페이스 포인터를 삽입합니다.
InsertAt 컬렉션의 지정된 위치에 IXpsOMRemoteDictionaryResource 인터페이스 포인터를 삽입합니다.
InsertAt 컬렉션의 지정된 위치에 IXpsOMSignatureBlockResource 인터페이스 포인터를 삽입합니다.
InsertAt 컬렉션의 지정된 위치에 IXpsOMVisual 인터페이스 포인터를 삽입합니다.
IsClosed IXpsOMPackageWriter 인터페이스의 상태를 가져옵니다.
IsPageLoaded 페이지를 로드할지 여부를 나타내는 참조된 페이지 상태를 가져옵니다.
LoadPackageFile 파일에서 디지털 서명 관리자로 기존 XPS 패키지를 로드합니다.
LoadPackageStream 스트림에서 디지털 서명 관리자로 XPS 패키지를 로드합니다.
PackageStatusUpdated 진행 중인 인쇄 작업이 이벤트를 발생하거나 작업이 완료되는 경우 패키지의 상태를 업데이트.
PTCloseProvider 인쇄 티켓 공급자 핸들을 닫습니다.
PTConvertDevModeToPrintTicket DEVMODE 구조를 IStream 내의 인쇄 티켓으로 변환합니다.
PTConvertPrintTicketToDevMode 인쇄 티켓을 DEVMODE 구조체로 변환합니다.

PTGetPrintCapabilities XML 인쇄 스키마에 따라 서식이 지정된 프린터의 기능을 검색합니다.
PTGetPrintDeviceCapabilities XML 인쇄 스키마에 따라 서식이 지정된 디바이스 프린터의 기능을 검색합니다.
PTGetPrintDeviceResources XML 인쇄 스키마에 따라 서식이 지정된 프린터의 인쇄 장치 리소스를 검색합니다.
PTMergeAndValidatePrintTicket 두 인쇄 티켓을 병합하고 유효한 실행 가능한 인쇄 티켓을 반환합니다.
PTOpenProvider 인쇄 티켓 공급자의 인스턴스를 엽니다. (PTOpenProvider)
PTOpenProviderEx 인쇄 티켓 공급자의 인스턴스를 엽니다. (PTOpenProviderEx)
PTQuerySchemaVersionSupport 지정한 프린터에서 지원하는 인쇄 스키마의 최고(최신) 버전을 검색합니다.
PTReleaseMemory 인쇄 티켓 및 인쇄 기능과 관련된 버퍼를 해제합니다.
RemoveAt 컬렉션의 지정된 위치에서 IXpsOMColorProfileResource 인터페이스 포인터를 제거하고 해제합니다.
RemoveAt 컬렉션의 지정된 위치에서 XPS_DASH 구조를 제거하고 해제합니다.
RemoveAt 사전의 지정된 위치에서 항목을 제거하고 해제합니다.
RemoveAt 컬렉션의 지정된 위치에서 IXpsOMDocument 인터페이스 포인터를 제거하고 해제합니다.

RemoveAt 컬렉션의 지정된 위치에서 IXpsOMFontResource 인터페이스 포인터를 제거하고 해제합니다.
RemoveAt 컬렉션의 지정된 위치에서 IXpsOMGeometryFigure 인터페이스 포인터를 제거하고 해제합니다.
RemoveAt 컬렉션의 지정된 위치에서 IXpsOMGradientStop 인터페이스 포인터를 제거하고 해제합니다.
RemoveAt 컬렉션의 지정된 위치에서 IXpsOMImageResource 인터페이스 포인터를 제거하고 해제합니다.
RemoveAt 컬렉션의 지정된 위치에서 IXpsOMPageReference 인터페이스 포인터를 제거하고 해제합니다.
RemoveAt 컬렉션의 지정된 위치에서 IOpcPartUri 인터페이스 포인터를 제거하고 해제합니다.
RemoveAt 컬렉션의 지정된 위치에서 IXpsOMRemoteDictionaryResource 인터페이스 포인터를 제거하고 해제합니다.
RemoveAt 컬렉션의 지정된 위치에서 IXpsOMSignatureBlockResource 인터페이스 포인터를 제거하고 해제합니다.
RemoveAt 컬렉션의 지정된 위치에서 IXpsOMVisual 인터페이스 포인터를 제거하고 해제합니다.
RemoveAt 컬렉션의 지정된 위치에서 IXpsSignatureBlock 인터페이스 포인터를 제거하고 해제합니다.
RemoveAt 컬렉션의 지정된 위치에서 IXpsSignature 인터페이스 포인터를 제거하고 해제합니다.
RemoveAt 컬렉션의 지정된 위치에서 IXpsSignatureRequest 인터페이스 포인터를 제거하고 해제합니다.

SavePackageToFile XPS 패키지를 파일에 저장합니다.
SavePackageToStream XPS 패키지를 스트림에 기록하여 저장합니다.
SetAccessibilityLongDescription 개체 내용에 대한 긴(자세한) 텍스트 설명을 설정합니다. (IXpsOMCanvas.SetAccessibilityLongDescription)
SetAccessibilityLongDescription 개체 내용에 대한 긴(자세한) 텍스트 설명을 설정합니다. (IXpsOMPath.SetAccessibilityLongDescription)
SetAccessibilityShortDescription 개체 내용에 대한 간단한 텍스트 설명을 설정합니다. (IXpsOMCanvas.SetAccessibilityShortDescription)
SetAccessibilityShortDescription 개체 내용에 대한 간단한 텍스트 설명을 설정합니다. (IXpsOMPath.SetAccessibilityShortDescription)
AdvisoryPageDimensions 설정 페이지의 제안된 차원을 설정합니다.
SetAt 컬렉션의 지정된 위치에 있는 IXpsOMColorProfileResource 인터페이스 포인터를 바꿉니다.
SetAt 컬렉션의 지정된 위치에 있는 XPS_DASH 구조를 바꿉니다.
SetAt 사전의 지정된 위치에 있는 항목을 바꿉니다.
SetAt 컬렉션의 지정된 위치에 있는 IXpsOMDocument 인터페이스 포인터를 바꿉니다.

SetAt 컬렉션의 지정된 위치에 있는 IXpsOMFontResource 인터페이스 포인터를 바꿉니다.
SetAt 컬렉션의 지정된 위치에 있는 IXpsOMGeometryFigure 인터페이스 포인터를 바꿉니다.
SetAt 컬렉션의 지정된 위치에 있는 IXpsOMGradientStop 인터페이스 포인터를 바꿉니다.
SetAt 컬렉션의 지정된 위치에 있는 IXpsOMImageResource 인터페이스 포인터를 바꿉니다.
SetAt 컬렉션의 지정된 위치에 있는 IXpsOMPageReference 인터페이스 포인터를 바꿉니다.
SetAt 컬렉션의 지정된 위치에 있는 IOpcPartUri 인터페이스 포인터를 바꿉니다.
SetAt 컬렉션의 지정된 위치에 있는 IXpsOMRemoteDictionaryResource 인터페이스 포인터를 바꿉니다.
SetAt 컬렉션의 지정된 위치에 있는 IXpsOMSignatureBlockResource 인터페이스 포인터를 바꿉니다.
SetAt 컬렉션의 지정된 위치에 있는 IXpsOMVisual 인터페이스 포인터를 바꿉니다.
SetBidiLevel 양방향 텍스트의 수준을 설정합니다.
SetBleedBox 페이지 블리드 상자의 크기를 설정합니다.
SetCategory 범주 속성을 설정합니다.

SetCenter 방사형 그라데이션 영역 타원의 중심점을 설정합니다.
SetClipGeometryLocal 시각적 개체에 대한 공유되지 않은 로컬 클리핑 영역을 설정합니다.
SetClipGeometryLookup 리소스 사전에서 공유 클립 기하 도형의 조회 키 이름을 설정합니다.
SetColor 그라데이션 중지점의 색 값 및 색 프로필을 설정합니다.
SetColor 브러시의 색 값 및 색 프로필을 설정합니다.
SetColorInterpolationMode 색 보간에 사용할 감마 함수를 설명하는 XPS_COLOR_INTERPOLATION 값을 설정합니다.
SetColorProfileResource 이미지와 연결된 색 프로필 리소스를 포함하는 IXpsOMColorProfileResource 인터페이스에 대한 포인터를 설정합니다.
SetContent 이 리소스와 연결할 읽기 전용 스트림을 설정합니다. (IXpsOMColorProfileResource.SetContent)
SetContent 이 리소스와 연결할 읽기 전용 스트림을 설정합니다. (IXpsOMDocumentStructureResource.SetContent)
SetContent 이 리소스와 연결할 읽기 전용 스트림을 설정합니다. (IXpsOMFontResource.SetContent)
SetContent 이 리소스와 연결할 읽기 전용 스트림을 설정합니다. (IXpsOMImageResource.SetContent)
SetContent 이 리소스와 연결할 읽기 전용 스트림을 설정합니다. (IXpsOMPrintTicketResource.SetContent)

SetContent 이 리소스와 연결할 읽기 전용 스트림을 설정합니다. (IXpsOMSignatureBlockResource.SetContent)
SetContent 이 리소스와 연결할 읽기 전용 스트림을 설정합니다. (IXpsOMStoryFragmentsResource.SetContent)
SetContentBox 페이지 콘텐츠 상자의 크기를 설정합니다.
SetContentStatus contentStatus 속성을 설정합니다.
SetContentType contentType 속성을 설정합니다.
SetCoreProperties XPS 패키지의 IXpsOMCoreProperties 인터페이스를 설정합니다.
SetCreated 만든 속성을 설정합니다.
SetCreator 작성자 속성을 설정합니다.
SetDescription description 속성을 설정합니다.
SetDeviceFontName 디바이스 글꼴의 이름을 설정합니다.
SetDictionary 이 리소스와 연결할 원격 사전의 IXpsOMDictionary 인터페이스에 대한 포인터를 설정합니다.
SetDictionaryLocal 공유되지 않은 로컬 사전의 IXpsOMDictionary 인터페이스 포인터를 설정합니다.

SetDictionaryLocal 페이지의 로컬 사전 리소스에 대한 IXpsOMDictionary 인터페이스 포인터를 설정합니다.
SetDictionaryResource 원격 사전 리소스의 IXpsOMRemoteDictionaryResource 인터페이스 포인터를 설정합니다.
SetDictionaryResource 페이지의 원격 사전 리소스에 대한 IXpsOMRemoteDictionaryResource 인터페이스 포인터를 설정합니다.
SetDigestMethod 다이제스트 메서드의 URI를 설정합니다.
SetDiscardControlPartName XPS 패키지에 있는 무시 컨트롤 파트의 이름을 설정합니다.
SetDocumentSequence XPS 패키지의 IXpsOMDocumentSequence 인터페이스를 설정합니다.
SetDocumentStructureResource 문서의 IXpsOMDocumentStructureResource 인터페이스를 설정합니다.
SetEndPoint 그라데이션의 끝점을 설정합니다.
SetFillBrushLocal IXpsOMBrush 인터페이스 포인터를 공유되지 않은 로컬 채우기 브러시로 설정합니다.
SetFillBrushLocal 채우기 브러시로 사용할 로컬 공유되지 않은 IXpsOMBrush 인터페이스에 대한 포인터를 설정합니다.
SetFillBrushLookup 공유 채우기 브러시의 조회 키 이름을 설정합니다.
SetFillBrushLookup 채우기 브러시로 사용할 리소스 사전에서 공유 브러시의 조회 키 이름을 설정합니다.

SetFillRule 사용할 채우기 규칙을 설명하는 XPS_FILL_RULE 값을 설정합니다.
SetFlags 이 서명에 사용할 서명 플래그를 지정하는 XPS_SIGN_FLAGS 값을 설정합니다.
SetFontRenderingEmSize 텍스트의 글꼴 크기를 설정합니다.
SetFontResource 이 텍스트에 필요한 글꼴 리소스 개체의 IXpsOMFontResource 인터페이스에 대한 포인터를 설정합니다.
SetGeometryLocal 이 경로에 대해 설정할 확인된 채우기 영역의 기하 도형을 포함하는 공유되지 않는 로컬 IXpsOMGeometry 인터페이스에 대한 포인터를 설정합니다.
SetGeometryLookup 리소스 사전에서 공유 기하 도형의 조회 키 이름을 설정합니다.
SetGlyphIndices 글꼴에서 사용할 문자 모양 인덱스를 설명하는 XPS_GLYPH_INDEX 구조체 배열을 설정합니다.
SetGlyphMappings UnicodeString 속성의 UTF-16 스칼라 값을 XPS_GLYPH_INDEX 구조체 배열의 항목에 매핑하는 방법을 설명하는 XPS_GLYPH_MAPPING 구조체 배열을 설정합니다.
SetGradientOrigin 방사형 그라데이션의 원점을 설정합니다.
SetHyperlinkNavigateUri 시각적 개체 하이퍼링크의 대상 URI를 설정합니다.
SetIdentifier 식별자 속성을 설정합니다.

SetImageResource 브러시의 원본으로 사용할 이미지 리소스가 포함된 IXpsOMImageResource 인터페이스에 대한 포인터를 설정합니다.
SetIntent 서명의 의도 또는 의미를 설명하는 문자열을 설정합니다. (IXpsSignatureRequest.SetIntent)
SetIsClosed 그림이 닫혀 있는지 여부를 나타내는 값을 설정합니다.
SetIsFilled 그림이 채워지는지 여부를 나타내는 값을 설정합니다.
SetIsHyperlinkTarget 페이지가 하이퍼링크의 대상인지 여부를 지정합니다.
SetIsHyperlinkTarget 시각적 개체가 하이퍼링크의 대상인지 여부를 지정합니다.
SetIsSideways 텍스트를 옆으로 회전된 문자 모양으로 렌더링할지 여부를 나타내는 값을 설정합니다.
SetKeywords 키워드 속성을 설정합니다.
SetLanguage 언어 속성을 설정합니다.
SetLanguage 페이지의 Language 속성을 설정합니다.
SetLanguage 시각적 개체의 Language 속성을 설정합니다.
SetLastModifiedBy lastModifiedBy 속성을 설정합니다.

SetLastPrinted lastPrinted 속성을 설정합니다.
SetMatrix 변환 행렬을 지정하는 XPS_MATRIX 구조를 설정합니다.
SetModelPrintTicket 지정한 부분을 사용하여 인쇄 티켓을 만듭니다.
SetModified 수정된 속성을 설정합니다.
SetName 이 페이지의 Name 속성을 설정합니다.
SetName 시각적 개체의 Name 속성을 설정합니다.
SetOffset 그라데이션 중지점의 오프셋 위치를 설정합니다.
SetOpacity 브러시의 불투명도를 설정합니다.
SetOpacity 시각적 개체의 불투명도 값을 설정합니다.
SetOpacityMaskBrushLocal IXpsOMBrush 인터페이스 포인터를 공유되지 않는 로컬 불투명 마스크 브러시로 설정합니다.
SetOpacityMaskBrushLookup 리소스 사전에서 공유 불투명 마스크 브러시의 조회 키 이름을 설정합니다.
SetOrigin 텍스트의 시작 위치를 설정합니다.

SetPage 페이지 참조의 IXpsOMPage 인터페이스를 설정합니다.
SetPageDimensions 페이지의 크기를 설정합니다.
SetPartName 파트가 serialize될 때 사용할 이름을 설정합니다.
SetPolicy 서명 정책을 나타내는 XPS_SIGN_POLICY 값을 설정합니다.
SetPrintTicketResource 문서 수준 인쇄 티켓의 IXpsOMPrintTicketResource 인터페이스를 설정합니다.
SetPrintTicketResource 문서 시퀀스에 대한 작업 수준 인쇄 티켓 리소스를 설정합니다.
SetPrintTicketResource 페이지에 할당할 페이지 수준 인쇄 티켓 리소스의 IXpsOMPrintTicketResource 인터페이스 포인터를 설정합니다.
SetProhibitedCaretStops 금지된 케어 중지 위치의 배열을 설정합니다.
SetRadiiSizes 방사형 그라데이션 영역의 줄임표를 정의하는 radii의 크기를 설정합니다.
SetRequestedSigner 패키지에 서명하거나 서명하도록 요청된 사람의 ID를 설정합니다.
SetRequestSignByDate 요청된 서명자가 문서의 지정된 부분에 서명해야 하는 날짜 및 시간을 설정합니다.
SetRevision 수정 속성을 설정합니다.

SetSegments 그림의 세그먼트에 대한 세그먼트 정보 및 데이터 요소를 설정합니다.
SetSignatureId Signature 요소의 Id 특성 값을 설정합니다. (IXpsSigningOptions.SetSignatureId)
SetSignatureMethod 서명 메서드를 설정합니다.
SetSignatureOriginPartName 서명 원본 파트의 파트 이름을 설정합니다.
SetSignaturePartName 문서 서명 파트의 파트 이름을 설정합니다.
SetSignatureXml 디지털 서명의 XML 태그를 설정합니다.
SetSigningLocale 패키지 서명 위치의 법적 관할권을 설정합니다.
SetSigningTimeFormat 서명 시간 문자열의 형식을 설정합니다.
SetSnapsToPixels 해당 경로가 렌더링될 때 경로가 디바이스 픽셀에 스냅되는지 여부를 나타내는 부울 값을 설정합니다.
SetSpotLocation 표시되는 디지털 서명 또는 디지털 서명 요청이 표시되는 페이지의 페이지와 위치를 지정합니다.
SetSpreadMethod 그라데이션 영역 외부의 영역을 렌더링하는 방법을 설명하는 XPS_SPREAD_METHOD 값을 설정합니다.
SetStartPoint 그림의 시작점을 설정합니다.

SetStartPoint 그라데이션의 시작점을 설정합니다.
SetStoryFragmentsResource 페이지에 할당할 StoryFragments 리소스의 IXpsOMStoryFragmentsResource 인터페이스 포인터를 설정합니다.
SetStrokeBrushLocal 스트로크 브러시로 사용할 로컬 공유되지 않은 IXpsOMBrush 인터페이스에 대한 포인터를 설정합니다.
SetStrokeBrushLookup 스트로크 브러시로 사용할 공유 브러시의 조회 키 이름을 설정합니다.
SetStrokeDashCap 스트로크 대시 캡의 스타일을 설정합니다.
SetStrokeDashOffset 스트로크의 원점에서 대시 배열 패턴의 시작점으로 오프셋을 설정합니다.
SetStrokeEndLineCap 스트로크 선의 끝 캡 스타일을 설정합니다.
SetStrokeLineJoin 스트로크 선 조인 스타일을 설정합니다.
SetStrokeMiterLimit 경로의 마이터 제한을 설정합니다.
SetStrokeStartLineCap 스트로크 선의 시작 부분에 있는 스트로크 선 캡의 스타일을 설정합니다.
SetStrokeThickness 스트로크 두께를 설정합니다.
SetStyleSimulations 문자 모양이 렌더링될 때 적용될 스타일 시뮬레이션을 설정합니다.

SetSubject 주체 속성을 설정합니다.
SetThumbnailResource XPS 문서의 썸네일 이미지를 설정합니다.
SetThumbnailResource 페이지에 할당할 썸네일 이미지 리소스의 IXpsOMImageResource 인터페이스에 대한 포인터를 설정합니다.
SetTileMode 브러시의 바둑판식 배열 모드를 설명하는 XPS_TILE_MODE 값을 설정합니다.
SetTitle 제목 속성을 설정합니다.
SetTransformLocal 공유되지 않은 로컬 행렬 변환을 설정합니다. (IXpsOMGeometry.SetTransformLocal)
SetTransformLocal IXpsOMMatrixTransform 인터페이스 포인터를 브러시에 사용할 로컬 공유되지 않은 행렬 변환으로 설정합니다.
SetTransformLocal IXpsOMMatrixTransform 인터페이스 포인터를 공유되지 않은 로컬 행렬 변환으로 설정합니다.
SetTransformLocal 공유되지 않은 로컬 행렬 변환을 설정합니다. (IXpsOMVisual.SetTransformLocal)
SetTransformLookup 리소스 사전에서 공유 행렬 변환의 조회 키 이름을 설정합니다. (IXpsOMGeometry.SetTransformLookup)
SetTransformLookup 브러시에 사용할 공유 행렬 변환의 조회 키 이름을 설정합니다.

SetTransformLookup 이 브러시의 변환으로 사용할 공유 행렬 변환의 조회 키 이름을 설정합니다.
SetTransformLookup 리소스 사전에서 공유 행렬 변환의 조회 키 이름을 설정합니다. (IXpsOMVisual.SetTransformLookup)
SetUnicodeString 이스케이프되지 않은 UTF-16 스칼라 값으로 텍스트를 설정합니다.
SetUseAliasedEdgeMode 별칭이 지정된 에지 모드를 사용하여 이 캔버스에 있는 개체의 가장자리를 렌더링할지 여부를 결정하는 값을 설정합니다.
SetVersion 버전 속성을 설정합니다.
SetViewbox 타일 이미지로 사용할 원본 콘텐츠의 부분을 설정합니다.
SetViewport 단일 타일로 덮인 대상 기하 도형의 부분을 설정합니다.
SetVisualLocal 브러시의 소스로 사용되는 공유되지 않은 로컬 시각적 개체의 인터페이스 포인터를 설정합니다.
SetVisualLookup 브러시의 원본으로 사용할 리소스 사전에 저장된 공유 시각적 개체의 조회 키 이름을 설정합니다.
서명 서명 옵션에 지정된 대로 XPS 패키지의 내용에 서명하고 결과 디지털 서명을 반환합니다.
StartNewDocument 패키지의 FixedDocumentSequence에서 새 FixedDocument를 열고 초기화합니다.
Verify 지정된 X.509 인증서에 대해 서명을 확인합니다.

쓰기 지정된 스트림에 페이지를 씁니다.
Write1 원격 사전 리소스를 스트림으로 직렬화합니다.
WriteToFile XPS 패키지를 지정된 파일에 씁니다.
WriteToFile1 XPS OM을 지정된 형식의 XPS 패키지로 파일에 씁니다.
WriteToStream XPS 패키지를 지정된 스트림에 씁니다.
WriteToStream1 XPS OM을 지정된 형식의 XPS 패키지로 스트림에 씁니다.

인터페이스

IPrintDocumentPackageStatusEvent 인쇄 작업의 진행률을 나타냅니다.
IPrintDocumentPackageTarget 사용자가 지원되는 패키지 대상 형식을 열거하고 지정된 형식 ID를 사용하여 만들 수 있습니다. IPrintDocumentPackageTarget은 패키지 인쇄 진행률 및 취소 추적도 지원합니다.
IPrintDocumentPackageTargetFactory 인쇄 작업을 시작하기 위해 IPrintDocumentPackageTarget과 함께 사용됩니다.
IXpsDocumentPackageTarget IXpsDocumentPackageTarget 인터페이스에는 문서 인쇄 모델에서 XPS 콘텐츠를 인쇄하는 데 필요한 요소가 포함되어 있습니다.

IXpsDocumentPackageTarget3D 인쇄를 위해 XPS에 3D 콘텐츠를 보내는 메서드를 제공합니다.
IXpsOMBrush 그래픽 개체를 그리는 데 사용되는 개체를 정의합니다. IXpsOMBrush에서 파생되는 클래스는 영역이 그려지는 방법을 설명합니다.
IXpsOMCanvas 시각적 요소 및 관련 속성의 그룹입니다.
IXpsOMColorProfileResource 색 프로필 리소스에 IStream 인터페이스를 제공합니다.
IXpsOMColorProfileResourceCollection IXpsOMColorProfileResource 인터페이스 포인터의 컬렉션입니다.
IXpsOMCoreProperties 이 인터페이스는 XPS 문서의 핵심 속성 부분에 저장된 메타데이터에 대한 액세스를 제공합니다.
IXpsOMDashCollection XPS_DASH 구조체의 컬렉션입니다.
IXpsOMDictionary 사전은 XPS 패키지에서 리소스를 공유하는 데 사용됩니다.
IXpsOMDocument 문서를 구성하는 고정 페이지 및 문서 수준 리소스의 순서가 지정된 시퀀스입니다.
IXpsOMDocumentCollection IXpsOMDocument 인터페이스 포인터의 컬렉션입니다.
IXpsOMDocumentSequence XPS 문서 콘텐츠가 있는 루트 개체입니다.
IXpsOMDocumentStructureResource DocumentStructure 파트의 리소스 스트림에 대한 XML 콘텐츠에 대한 액세스를 제공합니다.

IXpsOMFontResource 글꼴 리소스에 IStream 인터페이스를 제공합니다.
IXpsOMFontResourceCollection IXpsOMFontResource 인터페이스 포인터의 컬렉션입니다.
IXpsOMGeometry 경로 또는 클리핑 영역의 모양을 설명합니다.
IXpsOMGeometryFigure IXpsOMGeometry 인터페이스에 의해 지정된 경로 또는 클리핑 영역의 한 부분을 설명합니다.
IXpsOMGeometryFigureCollection IXpsOMGeometryFigure 인터페이스 포인터의 컬렉션입니다.
IXpsOMGlyphs 페이지에 표시되는 텍스트를 설명합니다.
IXpsOMGlyphsEditor IXpsOMGlyphs 인터페이스의 텍스트 콘텐츠에 영향을 주는 속성을 일괄 수정할 수 있습니다.
IXpsOMGradientBrush 이 인터페이스는 그라데이션 중지점으로 구성된 그라데이션을 설명합니다. IXpsOMGradientBrush에서 상속되는 클래스는 그라데이션 중지를 해석하는 다양한 방법을 지정합니다.
IXpsOMGradientStop 그라데이션 내의 단일 색과 위치를 나타냅니다.
IXpsOMGradientStopCollection IXpsOMGradientStop 인터페이스 포인터의 컬렉션입니다.
IXpsOMImageBrush 래스터 이미지를 소스로 사용하는 브러시입니다.
IXpsOMImageResource 이미지 리소스에 IStream 인터페이스를 제공합니다.

IXpsOMImageResourceCollection IXpsOMImageResource 인터페이스 포인터의 컬렉션입니다.
IXpsOMLinearGradientBrush 선형 그라데이션을 지정합니다. 이 그라데이션은 벡터를 따라 색 그라데이션입니다.
IXpsOMMatrixTransform 개체 모델의 다른 개체에 적용할 수 있는 아핀 행렬 변환을 지정합니다.
IXpsOMNameCollection 이름 문자열의 컬렉션입니다.
IXpsOMObjectFactory XPS 문서 개체 모델에 개체를 만듭니다.
IXpsOMObjectFactory1 IXpsOMObjectFactory에서 상속됩니다.
IXpsOMPackage XPS 개체 모델 트리에 대한 최상위 항목을 제공합니다.
IXpsOMPackage1 IXpsOMPackage에서 상속됩니다.
IXpsOMPackageTarget StartXpsPrintJob1 함수에서 만든 인쇄 작업에서 사용할 수 있는 IXpsOMPackageWriter를 만드는 메서드를 제공합니다.
IXpsOMPackageWriter XPS 문서의 일부를 패키지 파일에 증분 방식으로 씁니다.
IXpsOMPackageWriter3D 모델 텍스처 및 인쇄 티켓을 지원하는 메서드를 포함합니다.
IXpsOMPage 단일 페이지의 내용을 포함하는 개체 트리의 루트 노드를 제공합니다.

IXpsOMPage1 IXpsOMPage에서 상속됩니다.
IXpsOMPageReference XPS 문서에서 페이지의 가상화를 사용하도록 설정합니다.
IXpsOMPageReferenceCollection IXpsOMPageReference 인터페이스 포인터의 컬렉션입니다.
IXpsOMPart 모든 XPS 문서 파트 인터페이스에 대한 기본 인터페이스입니다.
IXpsOMPartResources XPS 문서의 모든 공유 부분 기반 리소스에 대한 액세스를 제공합니다.
IXpsOMPartUriCollection IOpcPartUri 인터페이스 포인터의 컬렉션입니다.
IXpsOMPath 텍스트가 아닌 시각적 개체 항목을 설명합니다.
IXpsOMPrintTicketResource PrintTicket 리소스에 IStream 인터페이스를 제공합니다.
IXpsOMRadialGradientBrush 방사형 그라데이션을 지정합니다.
IXpsOMRemoteDictionaryResource XPS 패키지의 페이지가 리소스를 공유할 수 있도록 하는 인터페이스를 제공합니다.
IXpsOMRemoteDictionaryResource1 IXpsOMRemoteDictionaryResource를 확장하여 OpenXPS 문서를 지원하는 메서드를 제공합니다.
IXpsOMRemoteDictionaryResourceCollection IXpsOMRemoteDictionaryResource 인터페이스 포인터의 컬렉션입니다.

IXpsOMResource XPS 개체 모델의 리소스 인터페이스에 대한 기본 인터페이스로 사용됩니다.
IXpsOMShareable 공유 가능한 인터페이스의 기본 인터페이스입니다.
IXpsOMSignatureBlockResource 서명 블록 리소스에 IStream 인터페이스를 제공합니다.
IXpsOMSignatureBlockResourceCollection IXpsOMSignatureBlockResource 인터페이스 포인터의 컬렉션입니다.
IXpsOMSolidColorBrush 단일 색 브러시입니다.
IXpsOMStoryFragmentsResource 페이지의 StoryFragments 부분의 리소스 스트림 콘텐츠에 대한 액세스를 제공합니다.
IXpsOMThumbnailGenerator 썸네일 이미지 리소스를 생성합니다.
IXpsOMTileBrush 타일 브러시는 시각적 이미지를 사용하여 이미지를 반복하여 영역을 그립니다.
IXpsOMVisual 경로, 캔버스 및 문자 모양 인터페이스에 대한 기본 인터페이스입니다.
IXpsOMVisualBrush 시각적 요소를 원본으로 사용하는 브러시입니다.
IXpsOMVisualCollection IXpsOMVisual 인터페이스 포인터의 컬렉션입니다.
IXpsSignature 단일 디지털 서명을 나타냅니다.

IXpsSignatureBlock SignatureDefinitions 부분에 저장된 서명 요청 블록을 나타냅니다.
IXpsSignatureBlockCollection IXpsSignatureBlock 인터페이스의 컬렉션입니다.
IXpsSignatureCollection IXpsSignature 인터페이스 포인터의 컬렉션입니다.
IXpsSignatureManager XPS 문서의 디지털 서명 및 디지털 서명 요청을 관리합니다.
IXpsSignatureRequest 서명 요청의 구성 요소에 액세스합니다.
IXpsSignatureRequestCollection IXpsSignatureRequest 인터페이스의 컬렉션입니다.
IXpsSigningOptions 새 서명에 사용되는 개별 서명 옵션에 대한 액세스를 제공합니다.

구조체

XPS_COLOR colorType이 XPS_COLOR_TYPE_CONTEXT 경우 XPS_COLOR 구조체의 내용입니다.
XPS_DASH 이 구조체는 경로의 대시 요소를 설명합니다.
XPS_GLYPH_MAPPING 문자 모양-인덱스 매핑에 대해 설명합니다.
XPS_MATRIX 3-by-3 행렬의 왼쪽 두 열을 설명합니다.

XPS_POINT 2차원 공간에서 x 및 y 좌표 쌍을 나타냅니다. (XPS_POINT)
XPS_RECT 사각형의 너비, 높이 및 위치를 설명합니다. (XPS_RECT)
XPS_SIZE 개체의 크기를 설명합니다.

accctrl.h 헤더

아티클2023. 08. 22.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)
- 보안 및 ID

accctrl.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

ACTRL_ACCESS_ENTRY_LISTA 액세스 항목 목록을 포함합니다. (ANSI)
ACTRL_ACCESS_ENTRY_LISTW 액세스 항목 목록을 포함합니다. (유니코드)
ACTRL_ACCESS_ENTRYA 지정된 트러스티에 대한 액세스 제어 정보를 포함합니다. 이 구조체는 ACE에 저장된 액세스 제어 정보와 동등한 정보를 저장합니다. (ANSI)
ACTRL_ACCESS_ENTRYW 지정된 트러스티에 대한 액세스 제어 정보를 포함합니다. 이 구조체는 ACE에 저장된 액세스 제어 정보와 동등한 정보를 저장합니다. (유니코드)
ACTRL_ACCESSA 개체 및 해당 속성에 대한 액세스 제어 목록의 배열을 포함합니다. (ANSI)
ACTRL_ACCESSW 개체 및 해당 속성에 대한 액세스 제어 목록의 배열을 포함합니다. (유니코드)
ACTRL_PROPERTY_ENTRYA 개체 또는 개체의 지정된 속성에 대한 액세스 제어 항목 목록을 포함합니다. (ANSI)
ACTRL_PROPERTY_ENTRYW 개체 또는 개체의 지정된 속성에 대한 액세스 제어 항목 목록을 포함합니다. (유니코드)

EXPLICIT_ACCESS_A 지정된 수탁자의 액세스 제어 정보를 정의합니다. (ANSI)
EXPLICIT_ACCESS_W 지정된 수탁자의 액세스 제어 정보를 정의합니다. (유니코드)
INHERITED_FROMA 개체의 상속된 ACE(액세스 제어 항목)에 대한 정보를 제공합니다. (ANSI)
INHERITED_FROMW 개체의 상속된 ACE(액세스 제어 항목)에 대한 정보를 제공합니다. (유니코드)
OBJECTS_AND_NAME_A 이름으로 트러스티를 식별하는 문자열과 ACE(개체별 액세스 제어 항목)의 개체 형식을 식별하는 추가 문자열을 포함합니다. (ANSI)
OBJECTS_AND_NAME_W 이름으로 트러스티를 식별하는 문자열과 ACE(개체별 액세스 제어 항목)의 개체 형식을 식별하는 추가 문자열을 포함합니다. (유니코드)
OBJECTS_AND_SID ACE(개체별 액세스 제어 항목)의 개체 형식을 식별하는 수탁자와 GUID를 식별하는 SID(보안 식별자)를 포함합니다.
TRUSTEE_A ACE(액세스 제어 항목)가 적용되는 사용자 계정, 그룹 계정 또는 로그온 세션을 식별합니다. (ANSI)
TRUSTEE_W ACE(액세스 제어 항목)가 적용되는 사용자 계정, 그룹 계정 또는 로그온 세션을 식별합니다. (유니코드)

열거형

ACCESS_MODE EXPLICIT_ACCESS 구조의 액세스 권한이 트러스티에 적용되는 방식을 나타내는 값을 포함합니다.

MULTIPLE_TRUSTEE_OPERATION TRUSTEE 구조가 가장 트러스티인지 여부를 나타내는 값을 포함합니다.
PROG_INVOKE_SETTING TreeSetNamedSecurityInfo 또는 TreeResetNamedSecurityInfo 함수에 대한 호출의 진행률을 추적하는 데 사용되는 함수의 초기 설정을 나타냅니다.
SE_OBJECT_TYPE 보안을 지원하는 Windows 개체의 형식에 해당하는 값을 포함합니다.
TRUSTEE_FORM TRUSTEE 구조체의 ptstrName 멤버가 가리키는 데이터 형식을 나타내는 값입니다.
TRUSTEE_TYPE TRUSTEE 구조로 식별되는 트러스티 유형을 나타내는 값입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

accountssettingspaneinterop.h 헤더

아티클2023. 08. 22.

이 헤더는 WinRT 상호 운용성에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- WinRT 상호 운용성
- AccountSettingsPane 클래스

accountssettingspaneinterop.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAccountsSettingsPaneInterop IAccountsSettingsPaneInterop은 WinRT AccountsSettingsPane 개체와의 상호 운용성을 가능하게 하고 멤버에 대한 액세스를 제공하여 계정 창을 표시합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

aclapi.h 헤더

아티클2023. 08. 22.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

aclapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

BuildExplicitAccessWithNameA 호출자가 지정한 데이터를 사용하여 EXPLICIT_ACCESS 구조를 초기화합니다. 트러스티는 이름 문자열로 식별됩니다. (ANSI)
BuildExplicitAccessWithNameW 호출자가 지정한 데이터를 사용하여 EXPLICIT_ACCESS 구조를 초기화합니다. 트러스티는 이름 문자열로 식별됩니다. (유니코드)
BuildSecurityDescriptorA 새 보안 설명자를 할당하고 초기화합니다. (ANSI)
BuildSecurityDescriptorW 새 보안 설명자를 할당하고 초기화합니다. (유니코드)
BuildTrusteeWithNameA TRUSTEE 구조를 초기화합니다. 호출자는 트러스티 이름을 지정합니다. 함수는 구조체의 다른 멤버를 기본값으로 설정합니다. (ANSI)
BuildTrusteeWithNameW TRUSTEE 구조를 초기화합니다. 호출자는 트러스티 이름을 지정합니다. 함수는 구조체의 다른 멤버를 기본값으로 설정합니다. (유니코드)
BuildTrusteeWithObjectsAndNameA ACE(개체별 액세스 제어 항목) 정보를 사용하여 TRUSTEE 구조를 초기화하고 구조체의 나머지 멤버를 기본값으로 초기화합니다. 호출자는 또한 트러스티의 이름을 지정합니다. (ANSI)

BuildTrusteeWithObjectsAndNameW ACE(개체별 액세스 제어 항목) 정보를 사용하여 TRUSTEE 구조를 초기화하고 구조체의 나머지 멤버를 기본값으로 초기화합니다. 호출자는 또한 트러스티의 이름을 지정합니다. (유니코드)
BuildTrusteeWithObjectsAndSidA ACE(개체별 액세스 제어 항목) 정보를 사용하여 TRUSTEE 구조를 초기화하고 구조체의 나머지 멤버를 기본값으로 초기화합니다. (ANSI)
BuildTrusteeWithObjectsAndSidW ACE(개체별 액세스 제어 항목) 정보를 사용하여 TRUSTEE 구조를 초기화하고 구조체의 나머지 멤버를 기본값으로 초기화합니다. (유니코드)
BuildTrusteeWithSidA TRUSTEE 구조를 초기화합니다. 호출자는 수탁자의 SID(보안 식별자)를 지정합니다. 함수는 구조체의 다른 멤버를 기본값으로 설정하고 SID와 연결된 이름을 조회하지 않습니다. (ANSI)
BuildTrusteeWithSidW TRUSTEE 구조를 초기화합니다. 호출자는 수탁자의 SID(보안 식별자)를 지정합니다. 함수는 구조체의 다른 멤버를 기본값으로 설정하고 SID와 연결된 이름을 조회하지 않습니다. (유니코드)
FreeInheritedFromArray GetInheritanceSource 함수에서 할당한 메모리를 해제합니다.
GetAuditedPermissionsFromAclA 지정된 트러스티에 대한 감사된 액세스 권한을 검색합니다. (ANSI)
GetAuditedPermissionsFromAclW 지정된 트러스티에 대한 감사된 액세스 권한을 검색합니다. (유니코드)
GetEffectiveRightsFromAclA ACL 구조가 지정된 수탁자에게 부여하는 유효 액세스 권한을 검색합니다. 트러스티의 유효 액세스 권한은 ACL이 트러스티 또는 트러스티가 구성원인 그룹에 부여하는 액세스 권한입니다. (ANSI)
GetEffectiveRightsFromAclW ACL 구조가 지정된 수탁자에게 부여하는 유효 액세스 권한을 검색합니다. 트러스티의 유효 액세스 권한은 ACL이 트러스티 또는 트러스티가 구성원인 그룹에 부여하는 액세스 권한입니다. (유니코드)
GetExplicitEntriesFromAclA

ACL(액세스 제어 목록)에서 ACL(액세스 제어 항목)을 설명하는 구조체 배열을 검색합니다. (ANSI)
GetExplicitEntriesFromAclW ACL(액세스 제어 목록)에서 ACL(액세스 제어 항목)을 설명하는 구조체 배열을 검색합니다. (유니코드)
GetInheritanceSourceA ACL(액세스 제어 목록)에서 상속된 ACE(액세스 제어 항목)의 원본에 대한 정보를 반환합니다. (ANSI)
GetInheritanceSourceW ACL(액세스 제어 목록)에서 상속된 ACE(액세스 제어 항목)의 원본에 대한 정보를 반환합니다. (유니코드)
GetNamedSecurityInfoA 이름으로 지정된 개체에 대한 보안 설명자의 복사본을 검색합니다. (ANSI)
GetNamedSecurityInfoW 이름으로 지정된 개체에 대한 보안 설명자의 복사본을 검색합니다. (유니코드)
GetSecurityInfo 핸들로 지정된 개체에 대한 보안 설명자의 복사본을 검색합니다.
GetTrusteeFormA 지정된 TRUSTEE 구조에서 트러스티 이름을 검색합니다. 이 값은 구조체가 이름 문자열 또는 SID(보안 식별자)를 사용하여 트러스티를 식별하는지 여부를 나타냅니다. (ANSI)
GetTrusteeFormW 지정된 TRUSTEE 구조에서 트러스티 이름을 검색합니다. 이 값은 구조체가 이름 문자열 또는 SID(보안 식별자)를 사용하여 트러스티를 식별하는지 여부를 나타냅니다. (유니코드)
GetTrusteeNameA 지정된 TRUSTEE 구조에서 트러스티 이름을 검색합니다. (ANSI)
GetTrusteeNameW 지정된 TRUSTEE 구조에서 트러스티 이름을 검색합니다. (유니코드)
GetTrusteeTypeA

지정된 TRUSTEE 구조에서 트러스티 형식을 검색합니다. 이 값은 트러스티가 사용자, 그룹 또는 트러스티 형식을 알 수 없는지 여부를 나타냅니다. (ANSI)
GetTrusteeTypeW 지정된 TRUSTEE 구조에서 트러스티 형식을 검색합니다. 이 값은 트러스티가 사용자, 그룹 또는 트러스티 형식을 알 수 없는지 여부를 나타냅니다. (유니코드)
LookupSecurityDescriptorPartsA 자체 상대 보안 설명자에서 보안 정보를 검색합니다. (ANSI)
LookupSecurityDescriptorPartsW 자체 상대 보안 설명자에서 보안 정보를 검색합니다. (유니코드)
SetEntriesInAclA 새 액세스 제어 또는 감사 제어 정보를 기존 ACL 구조에 병합하여 새 ACL(액세스 제어 목록)을 만듭니다. (ANSI)
SetEntriesInAclW 새 액세스 제어 또는 감사 제어 정보를 기존 ACL 구조에 병합하여 새 ACL(액세스 제어 목록)을 만듭니다. (유니코드)
SetNamedSecurityInfoA 지정된 개체의 보안 설명자에 지정된 보안 정보를 설정합니다. (ANSI)
SetNamedSecurityInfoW 지정된 개체의 보안 설명자에 지정된 보안 정보를 설정합니다. (유니코드)
SetSecurityInfo 지정된 개체의 보안 설명자에 지정된 보안 정보를 설정합니다. 호출자는 핸들로 개체를 식별합니다.
TreeResetNamedSecurityInfoA 지정된 개체 트리의 보안 설명자에서 지정된 보안 정보를 다시 설정합니다. (ANSI)
TreeResetNamedSecurityInfoW 지정된 개체 트리의 보안 설명자에서 지정된 보안 정보를 다시 설정합니다. (유니코드)
TreeSetNamedSecurityInfoA

지정된 개체 트리의 보안 설명자에 지정된 보안 정보를 설정합니다. (ANSI)	
TreeSetNamedSecurityInfoW 지정된 개체 트리의 보안 설명자에 지정된 보안 정보를 설정합니다. (유니코드)	

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

aclui.h 헤더

아티클2023. 03. 16.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- CIMWin32 WMI 공급자
- 보안 및 ID

aclui.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEffectivePermission 개체에 대한 보안 주체에 대한 유효 권한을 결정하는 방법을 제공합니다.
IEffectivePermission2 개체에 대한 보안 주체에 대한 유효 권한을 확인하는 방법을 제공합니다.
ISecurityInformation 액세스 제어 편집기가 CreateSecurityPage 및 EditSecurity 함수의 호출자와 통신할 수 있도록 합니다.
ISecurityInformation2 액세스 제어 편집기가 ISecurityInformation 인터페이스에서 제공하지 않는 정보를 클라이언트에서 가져올 수 있도록 합니다.
ISecurityInformation3 사용자가 편집 단추에 방패 이미지를 표시하는 액세스 제어 편집기 페이지에서 편집 단추를 클릭할 때 관리자 권한 액세스 제어 편집기를 표시하는 데 필요한 메서드를 제공합니다.
ISecurityInformation4 ACE(액세스 제어 편집기)가 공유의 보안 설명자를 가져와 공유 페이지를 초기화할 수 있도록 합니다.
ISecurityObjectTypeInfo 임의 액세스 제어 목록(DCL) 및 시스템 액세스 제어 목록(SACL)에서 상속된 ACE(액세스 제어 항목)의 원본을 결정하는 방법을 제공합니다.

함수

CreateSecurityPage 사용자가 개체의 DACL(임의 액세스 제어 목록)에서 ACE(액세스 제어 항목)에서 허용되거나 거부된 액세스 권한을 보고 편집할 수 있도록 하는 기본 보안 속성 페이지를 만듭니다.
EditSecurity 기본 보안 속성 페이지가 포함된 속성 시트를 표시합니다. 이 속성 페이지를 사용하면 사용자가 개체의 DACL에서 ACE가 허용하거나 거부하는 액세스 권한을 보고 편집할 수 있습니다.
EditSecurityAdvanced 기본 보안 속성 페이지가 포함된 속성 시트를 표시할 때 보안 페이지 형식을 포함하도록 EditSecurity 함수를 확장합니다.

구조체

EFFPERM_RESULT_LIST 유효 사용 권한을 나열합니다.
SECURITY_OBJECT 보안 개체 정보를 포함합니다.
SI_ACCESS 보안 개체에 대한 액세스 권한 또는 기본 액세스 마스크에 대한 정보를 포함합니다.
SI_INHERIT_TYPE 자식 개체에서 ACE(액세스 제어 항목)를 상속하는 방법에 대한 정보를 포함합니다.
SI_OBJECT_INFO 액세스 제어 편집기를 초기화하는 데 사용됩니다.
SID_INFO ISecurityInformation2::LookupSids에서 반환된 SID 구조에 해당하는 일반 이름 목록을 포함합니다.

SID_INFO_LIST SID_INFO 구조의 목록을 포함합니다.

열거형

SI_PAGE_TYPE 액세스 제어 편집기 속성 시트의 속성 페이지 유형을 나타내는 값을 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

activation.h 헤더

아티클2023. 03. 09.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

activation.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IActivationFactory Windows 런타임을 통해 클래스를 활성화할 수 있습니다.

activationregistration.h 헤더

아티클2023. 08. 22.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

activationregistration.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IActivatableClassRegistration 클래스에 대한 등록 정보를 얻을 수 있습니다.
IDllServerActivatableClassRegistration In Process 서버에 대한 등록 정보를 얻을 수 있습니다.
IExeServerActivatableClassRegistration Out-of-process 서버에 대한 등록 정보를 얻을 수 있습니다.
IExeServerRegistration 등록된 Out-of-process 서버를 나타냅니다.

열거형

__x_ABI_CWindows_CFoundation_CActivationType 활성화 가능한 클래스에 대한 활성화 종류를 지정합니다. A
activationType? ActivationType(activationregistration.h)은 활성화 가능한 클래스에 대한 활성화 종류를 지정합니다.
IdentityType

Out-of-process 서버에 대한 정품 인증의 종류를 나타냅니다.
InstancingType out-of-process 서버에 대한 인스턴스화 동작의 종류를 나타냅니다.
RegistrationScope 활성화 가능한 클래스의 배포 scope 나타냅니다.
ThreadingType In Process 서버를 활성화하는 데 사용할 아파트 스레딩 모델을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

activitycoordinator.h 헤더

아티클2024. 06. 12.

이 헤더는 Activity Coordinator C/C++ API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 활동 코디네이터

activitycoordinator.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

CreateActivityCoordinatorPolicy 새 정책을 할당하고 제공된 정책 템플릿 열거형에 따라 필드를 미리 채웁니다.
DestroyActivityCoordinatorPolicy CreateActivityCoordinatorPolicy를 통해 얻은 정책을 삭제합니다.
GetActivityCoordinatorPolicyResourceCondition 정책에 대한 리소스의 원하는 조건을 가져옵니다.
IsActivityCoordinatorResourceSupported 이 함수를 사용하면 앱이 런타임에 지원되는 리소스를 확인할 수 있습니다.
SetActivityCoordinatorPolicyResourceCondition 정책에 대한 리소스의 원하는 조건을 설정합니다.
SubscribeActivityCoordinatorPolicy 정책의 열기 및 닫기 알림을 받는 구독을 만듭니다.
UnsubscribeActivityCoordinatorPolicy 구독을 삭제합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

activitycoordinatortypes.h 헤더

아티클2023. 08. 22.

이 헤더는 Activity Coordinator C/C++ API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 활동 코디네이터

activitycoordinatortypes.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

콜백 함수

ACTIVITY_COORDINATOR_CALLBACK 활동에 대한 모든 알림에 대해 호출할 콜백입니다.

열거형

ACTIVITY_COORDINATOR_CONDITION 지원되는 조건 집합의 열거형입니다.
ACTIVITY_COORDINATOR_NOTIFICATION 구독 콜백에 대해 가능한 알림 유형의 열거형입니다.
ACTIVITY_COORDINATOR_POLICY_TEMPLATE 지원되는 템플릿 정책 집합의 열거형입니다.
ACTIVITY_COORDINATOR_RESOURCE 지원되는 리소스 집합의 열거형입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

adhoc.h 헤더

아티클2023. 08. 22.

이 헤더는 네이티브 와이파이에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네이티브 와이파이

adhoc.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDot11AdHocInterface NIC(무선 네트워크 인터페이스 카드)를 나타냅니다.
IDot11AdHocInterfaceNotificationSink IDot11AdHocInterface에서 지원하는 알림을 정의합니다.
IDot11AdHocManager 802.11 임시 네트워크를 만들고 관리합니다.
IDot11AdHocManagerNotificationSink IDot11AdHocManager 인터페이스에서 지원하는 알림을 정의합니다.
IDot11AdHocNetwork 연결 범위 내에서 사용 가능한 임시 네트워크 대상을 나타냅니다.
IDot11AdHocNetworkNotificationSink IDot11AdHocNetwork 인터페이스에서 지원하는 알림을 정의합니다.
IDot11AdHocSecuritySettings 무선 임시 네트워크에 대한 보안 설정을 지정합니다.
IEnumDot11AdHocInterfaces 현재 표시되는 802.11 임시 네트워크 인터페이스의 컬렉션을 나타냅니다.
IEnumDot11AdHocNetworks

현재 표시되는 802.11 임시 네트워크의 컬렉션을 나타냅니다.
IEnumDot11AdHocSecuritySettings 표시되는 각 무선 임시 네트워크와 연결된 보안 설정의 컬렉션을 나타냅니다.

열거형

DOT11_ADHOC_AUTH_ALGORITHM 임시 네트워크에서 사용자 또는 컴퓨터 인증에 대한 인증 알고리즘을 지정합니다.
DOT11_ADHOC_CIPHER_ALGORITHM 임시 네트워크에서 정보를 암호화하고 암호를 해독하는 데 사용되는 암호 알고리즘을 지정합니다.
DOT11_ADHOC_CONNECT_FAIL_REASON 연결 시도가 실패한 이유를 지정합니다.
DOT11_ADHOC_NETWORK_CONNECTION_STATUS 임시 네트워크의 연결 상태를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

adshlp.h 헤더

아티클2023. 08. 22.

이 헤더는 Active Directory 서비스 인터페이스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Active Directory 서비스 인터페이스

adshlp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

ADsBuildEnumerator ADsBuildEnumerator 함수는 지정된 ADSI 컨테이너 개체에 대한 열거자 개체를 만듭니다.
ADsBuildVarArrayInt ADsBuildVarArrayInt 함수는 DWORD 값 배열에서 정수의 변형 배열을 빌드합니다.
ADsBuildVarArrayStr ADsBuildVarArrayStr 함수는 유니코드 문자열 배열에서 변형 배열을 빌드합니다.
ADsEncodeBinaryData ADsEncodeBinaryData 함수는 BLOB(Binary Large Object)을 검색 필터에 포함하기에 적합한 유니코드 형식으로 변환합니다.
ADsEnumerateNext ADsEnumerateNext 함수는 열거자의 현재 커서 위치에서 지정된 수의 요소를 열거합니다.
ADsFreeEnumerator ADsBuildEnumerator 함수를 사용하여 만든 열거자 개체를 해제합니다.
ADsGetLastError ADsGetLastError 함수는 호출 스레드의 마지막 오류 코드 값을 검색합니다.
ADsGetObject 경로 및 지정된 인터페이스 식별자가 지정된 경우 개체에 바인딩합니다.

ADsOpenObject 명시적 사용자 이름 및 암호 자격 증명을 사용하여 ADSI 개체에 바인딩합니다.
ADsSetLastError ADsSetLastError는 호출 스레드에 대한 마지막 오류 코드 값을 설정합니다.
AllocADsMem 지정된 크기의 메모리 블록을 할당합니다.
AllocADsStr 에 메모리를 할당하고 지정된 문자열을 복사합니다.
BinarySDToSecurityDescriptor 이진 보안 설명자를 IADsSecurityDescriptor 개체로 변환합니다.
FreeADsMem AllocADsMem 또는 ReallocADsMem에서 할당한 메모리를 해제합니다.
FreeADsStr AllocADsStr 또는 ReallocADsStr에서 할당한 문자열의 메모리를 해제합니다.
ReallocADsMem 기존 메모리 블록을 다시 할당하고 복사합니다.
ReallocADsStr 유니코드 문자열의 복사본을 만듭니다.
SecurityDescriptorToBinarySD IADsSecurityDescriptor 개체를 이진 보안 설명자 형식으로 변환합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

adsprop.h 헤더

아티클2023. 08. 22.

이 헤더는 Active Directory Domain Services 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Active Directory Domain Services

adsprop.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

ADsPropCheckIfWritable ADsPropCheckIfWritable 함수는 특성을 쓸 수 있는지 여부를 결정합니다.
ADsPropCreateNotifyObj ADsPropCreateNotifyObj 함수는 Active Directory Domain Services 속성 시트 확장에서 사용할 알림 개체를 만들거나 가져오는 데 사용됩니다.
ADsPropGetInitInfo Active Directory Domain Services 속성 시트 확장이 적용되는 디렉터리 개체 데이터를 가져오는 데 사용됩니다.
ADsPropSendErrorMessage ADsPropSendErrorMessage 함수는 ADsPropShowErrorDialog 함수를 호출하여 표시되는 오류 메시지 목록에 오류 메시지를 추가합니다.
ADsPropSetHwnd 속성 페이지 창 핸들의 알림 개체에 알리는 데 사용됩니다. (ADsPropSetHwnd)
ADsPropSetHwndWithTitle 속성 페이지 창 핸들의 알림 개체에 알리는 데 사용됩니다. (ADsPropSetHwndWithTitle)
ADsPropShowErrorDialog ADsPropShowErrorDialog 함수는 ADsPropSendErrorMessage 함수 또는 WM_ADSPROP_NOTIFY_ERROR 대한 호출을 통해 누적된 오류 메시지가 포함된 대화 상자를 표시합니다.

구조체

ADSPROPERROR ADSPROPERROR 구조는 ADsPropSendErrorMessage 함수 또는 WM_ADSPROP_NOTIFY_ERROR 메시지를 사용하여 알림 개체에 오류 데이터를 전달하는 데 사용됩니다.
ADSPROPINITPARAMS ADsPropGetInitInfo 함수와 함께 사용하여 표시 지정자가 적용되는 개체 데이터를 가져옵니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

adtgen.h 헤더

아티클2023. 08. 22.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

adtgen.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

열거형

AUDIT_PARAM_TYPE 사용할 수 있는 감사 매개 변수의 유형을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

advpub.h 헤더

아티클2023. 08. 22.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

advpub.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

RegInstallA 제공된 테이블의 문자열 레지스트리 값을 업데이트. (ANSI)
RegInstallW 제공된 테이블의 문자열 레지스트리 값을 업데이트. (유니코드)

구조체

STRENTRYA 레지스트리 문자열 대체를 나타냅니다. (ANSI)
STRENTRYW 레지스트리 문자열 대체를 나타냅니다. (유니코드)
STRTABLEA 레지스트리 문자열 대체 테이블을 나타냅니다. (ANSI)
STRTABLEW 레지스트리 문자열 대체 테이블을 나타냅니다. (유니코드)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

af_irda.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows 소켓 2에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 소켓 2

af_irda.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

SOCKADDR_IRDA SOCKADDR_IRDA 구조는 주소 패밀리 AF_IRDA 정의된 IrDA 소켓 작업과 함께 사용됩니다.

amsi.h 헤더

아티클2023. 08. 22.

이 헤더는 맬웨어 방지 검사 인터페이스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 맬웨어 방지 검사 인터페이스

amsi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAmsiStream 검사할 스트림을 나타냅니다.
IAntimalware 맬웨어 방지 제품을 나타냅니다. (IAntimalware)
IAntimalware2 맬웨어 방지 제품을 나타냅니다. (IAntimalware2)
IAntimalwareProvider 맬웨어 방지 제품의 공급자를 나타냅니다. (IAntimalwareProvider)
IAntimalwareProvider2 맬웨어 방지 제품의 공급자를 나타냅니다. (IAntimalwareProvider2)

Functions

AmsiCloseSession AmsiOpenSession에서 연 세션을 닫습니다.
AmsiInitialize AMSI API를 초기화합니다.

AmsiNotifyOperation 맬웨어 방지 공급자에게 임의 작업에 대한 알림을 보냅니다. (AmsiNotifyOperation)
AmsiOpenSession 여러 검사 요청의 상관 관계를 지정할 수 있는 세션을 엽니다.
AmsiResultIsMalware 검사 결과가 콘텐츠를 차단해야 했음을 나타내는지 여부를 확인합니다.
AmsiScanBuffer 버퍼가 가득한 콘텐츠에서 맬웨어를 검사합니다.
AmsiScanString 문자열에서 맬웨어를 검색합니다.
AmsiUninitialize AmsiInitialize에서 원래 연 AMSI API의 instance 제거합니다.

열거형

AMSI_ATTRIBUTE IAmsiStream::GetAttribute에서 요청할 수 있는 특성 유형을 지정합니다.
AMSI_RESULT 검사에서 반환되는 결과의 유형을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

apiquery2.h 헤더

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 노트

apiquery2.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

GetApiSetModuleBaseName 지정된 API 집합을 구현하는 모듈 또는 이진 파일의 파일 이름을 검색합니다.
IsApiSetImplemented IsApiSetImplemented 함수는 지정된 API 집합이 컴퓨터에 있는지 테스트합니다.

Last updated on 2025. 12. 18.

appcompatapi.h 헤더

아티클2023. 08. 22.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

appcompatapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

ApphelpCheckShellObject 애플리케이션에서 잘못된 확장 개체를 검색하고 실행하지 못하도록 차단하거나 수정할 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

appmgmt.h 헤더

아티클2023. 08. 22.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 그룹 정책 API
- Windows 셸

appmgmt.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

GetLocalManagedApplications 대상 컴퓨터에서 GetLocalManagedApplications 함수를 실행하여 해당 컴퓨터에서 관리되는 애플리케이션 목록을 가져올 수 있습니다.
GetManagedApplicationCategories GetManagedApplicationCategories 함수는 도메인에 대한 애플리케이션 범주 목록을 가져옵니다. 목록은 도메인의 모든 사용자에 대해 동일합니다.
GetManagedApplications GetManagedApplications 함수는 지정된 사용자 컨텍스트에 대한 ARP(프로그램 추가/제거)의 추가 창에 표시되는 애플리케이션 목록을 가져옵니다.
InstallApplication InstallApplication 함수는 도메인에 속한 대상 사용자에게 배포된 애플리케이션을 설치할 수 있습니다.
UninstallApplication UninstallApplication 함수는 Windows Installer .msi 파일을 사용하여 설치 및 설치를 처리하는 그룹 정책 애플리케이션을 제거합니다.

구조체

APPCATEGORYINFO

제어판 프로그램 추가/제거에 애플리케이션 범주 정보를 제공합니다. APPCATEGORYINFOLIST 구조체는 애플리케이션 게시자에 대한 범주의 전체 목록을 만드는 데 사용됩니다.
APPCATEGORYINFOLIST 애플리케이션 게시자에서 제어판 프로그램 추가/제거에 지원되는 애플리케이션 범주 목록을 제공합니다.
INSTALLDATA INSTALLDATA 구조는 InstallApplication에서 설치할 그룹 정책 애플리케이션을 지정합니다.
INSTALLSPEC INSTALLSPEC 구조는 사용자에게 친숙한 이름 및 그룹 정책 GUID 또는 파일 이름 확장명을 사용하여 그룹 정책 애플리케이션을 지정합니다. INSTALLDATA 구조체의 Spec 멤버는 이 정보를 InstallApplication 함수에 제공합니다.
LOCALMANAGEDAPPLICATION LOCALMANAGEDAPPLICATION 구조는 사용자 또는 컴퓨터에 대해 설치된 관리되는 애플리케이션을 설명합니다. GetLocalManagedApplications 함수에서 반환됩니다.
MANAGEDAPPLICATION MANAGEDAPPLICATION 구조에는 애플리케이션에 대한 정보가 포함됩니다. GetManagedApplications 함수는 MANAGEDAPPLICATION 구조체의 배열을 반환합니다.

열거형

INSTALLSPECTYPE INSTALLSPECTYPE 열거형 값은 그룹 정책 애플리케이션을 InstallApplication 함수에 지정할 수 있는 방법을 정의합니다. 값은 INSTALLDATA의 Type 멤버에 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

appmodel.h 헤더

아티클2025. 01. 27.

이 헤더는 Windows 스토어 앱의 패키징, 배포 및 쿼리에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 스토어 앱 패키징, 배포 및 쿼리

appmodel.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

테이블 확장

AddPackageDependency 이전에 정의된 PackageDependency를 특정 패키지에 확인하고 호출 프로세스의 패키지 그래프에 추가합니다. 종속성이 추가된 후 다른 코드 로드 메서드(예: LoadLibrary 및 CoCreateInstance)는 확인된 패키지에서 이진 파일을 찾을 수 있습니다.
AppPolicyGetClrCompat 프라이빗 리플렉션을 사용하도록 설정하고 관리되는 개체를 민첩하게 만들지 여부를 결정할 수 있도록 프로세스의 애플리케이션 유형을 나타내는 값을 검색합니다.
appPolicyGetCreateFileAccess 프로세스에 IO 디바이스(파일, 파일 스트림, 디렉터리, 실제 디스크, 볼륨, 콘솔 버퍼, 테이프 드라이브, 통신 리소스, mailslot 및 파이프)에 대한 전체 또는 제한된 액세스 권한이 있는지 여부를 나타내는 값을 검색합니다.
AppPolicyGetLifecycleManagement PLM(프로세스 수명 주기 관리자)에서 프로세스를 일시 중단/재개할 수 있는지 여부를 나타내는 값을 검색합니다.
AppPolicyGetMediaFoundationCodecLoading 프로세스의 정책에서 비 Windows(타사) 플러그 인을 로드할 수 있는지 여부를 나타내는 값을 검색합니다. 이 값을 사용하여 비 Windows(타사) 플러그 인을 허용할지 여부를 결정할 수 있습니다.
AppPolicyGetProcessTerminationMethod 프로세스를 종료하는 데 사용되는 메서드를 검색합니다.

AppPolicyGetShowDeveloperDiagnostic 어설션과 같은 개발자 정보를 사용자에게 노출하는 프로세스에 사용되는 메서드를 검색합니다.
appPolicyGetThreadInitializationType beginthread[ex]가 스레드를 만들 때 프로세스에 대해 자동으로 수행되어야 하는 초기화 종류를 검색합니다.
appPolicyGetWindowingModel 프로세스에서 CoreWindow 기반 또는 HWND 기반 창 모델을 사용하는지 여부를 나타내는 값을 검색합니다. 값을 사용하여 창 상태 변경 알림(크기 변경, 표시 유형 변경 등)에 등록하는 방법을 결정할 수 있습니다.
ClosePackageInfo 지정된 패키지 정보에 대한 참조를 닫습니다.
DeletePackageDependency TryCreatePackageDependency 메서드를 사용하여 이전에 만든 프레임워크 패키지 종속성에 대한 설치 시간 참조를 삭제합니다. 이 메서드는 다른 앱에 종속성이 없는 경우 프레임워크 패키지를 제거하는 것이 안전하다는 것을 OS에 알릴 수 있습니다.
FindPackagesByPackageFamily 현재 사용자의 지정된 패밀리 이름을 가진 패키지를 찾습니다.
FormatApplicationUserModelId 패키지 패밀리 이름 및 PRAID(패키지 상대 애플리케이션 ID)에서 애플리케이션 사용자 모델 ID를 생성합니다.
GetApplicationUserModelId 지정된 프로세스에 대한 애플리케이션 사용자 모델 ID를 가져옵니다.
GetApplicationUserModelIdFromToken 지정된 토큰의 애플리케이션 사용자 모델 ID를 가져옵니다.
GetCurrentApplicationUserModelId 현재 프로세스에 대한 애플리케이션 사용자 모델 ID를 가져옵니다.
GetCurrentPackageFamilyName 호출 프로세스의 패키지 패밀리 이름을 가져옵니다.

GetCurrentPackageFullName 호출 프로세스의 패키지 전체 이름을 가져옵니다.
GetCurrentPackageId 호출 프로세스의 패키지 식별자(ID)를 가져옵니다.
GetCurrentPackageInfo 호출 프로세스에 대한 패키지 정보를 가져옵니다. (GetCurrentPackageInfo)
GetCurrentPackageInfo2 호출 프로세스에 대한 패키지 정보를 가져옵니다. (GetCurrentPackageInfo2)
GetCurrentPackagePath 호출 프로세스의 패키지 경로를 가져옵니다. (GetCurrentPackagePath)
GetCurrentPackagePath2 호출 프로세스의 패키지 경로를 가져옵니다. (GetCurrentPackagePath2)
GetIdForPackageDependencyContext 지정된 컨텍스트 핸들에 대한 패키지 종속성을 반환합니다.
GetPackageApplicationIds 지정된 패키지에 있는 앱의 ID를 가져옵니다.
GetPackageFamilyName 지정된 프로세스의 패키지 패밀리 이름을 가져옵니다.
GetPackageFamilyNameFromToken 지정된 토큰의 패키지 패밀리 이름을 가져옵니다.
GetPackageFullName 지정된 프로세스의 패키지 전체 이름을 가져옵니다.
GetPackageFullNameFromToken 지정된 토큰의 패키지 전체 이름을 가져옵니다.
GetPackageId

지정된 프로세스의 패키지 식별자(ID)를 가져옵니다.
GetPackageInfo 지정된 패키지에 대한 패키지 정보를 가져옵니다. (GetPackageInfo)
GetPackageInfo2 지정된 패키지에 대한 패키지 정보를 가져옵니다. (GetPackageInfo2)
GetPackagePath 지정된 패키지의 경로를 가져옵니다.
GetPackagePathByFullName 지정된 패키지의 경로를 가져옵니다. (GetPackagePathByFullName)
GetPackagePathByFullName2 지정된 패키지의 경로를 가져옵니다. (GetPackagePathByFullName2)
GetPackagesByPackageFamily 현재 사용자의 지정된 패밀리 이름을 가진 패키지를 가져옵니다.
getResolvedPackageFullNameForPackageDependency 패키지 종속성을 확인할 경우 사용할 패키지 전체 이름을 반환합니다. 이 함수는 프로세스 그래프에 패키지를 추가하지 않습니다.
GetStagedPackageOrigin 지정된 패키지의 원본을 가져옵니다.
GetStagedPackagePathByFullName 지정된 스테이징된 패키지의 경로를 가져옵니다. (GetStagedPackagePathByFullName)
GetStagedPackagePathByFullName2 지정된 스테이징된 패키지의 경로를 가져옵니다. (GetStagedPackagePathByFullName2)
OpenPackageInfoByFullName 지정된 패키지의 패키지 정보를 엽니다.
packageFamilyNameFromFullName 지정된 패키지 전체 이름의 패키지 패밀리 이름을 가져옵니다.

packageFamilyNameFromId 지정된 패키지 식별자의 패키지 패밀리 이름을 가져옵니다.
PackageFullNameFromId 지정된 패키지 식별자(ID)의 패키지 전체 이름을 가져옵니다.
PackageIdFromFullName 지정된 패키지 전체 이름의 패키지 식별자(ID)를 가져옵니다.
PackageNameAndPublisherIdFromFamilyName 지정된 패키지 패밀리 이름의 패키지 이름 및 게시자 ID(ID)를 가져옵니다.
ParseApplicationUserModelId 애플리케이션 사용자 모델 ID를 패키지 패밀리 이름 및 PRAID(패키지 상대 애플리케이션 ID)로 분해합니다.
RemovePackageDependency 현재 프로세스의 패키지 그래프(즉, AddPackageDependency 함수를 사용하여 추가된 프레임워크 패키지 종속성에 대한 런타임 참조)에서 확인된 패키지 종속성을 제거합니다.
TryCreatePackageDependency 지정된 패키지 패밀리 이름, 최소 버전 및 추가 조건을 사용하여 현재 앱에 대한 프레임워크 패키지 종속성에 대한 설치 시간 참조를 만듭니다.
VerifyApplicationUserModelId 애플리케이션 사용자 모델 ID가 구문적으로 유효한지 여부를 확인합니다.
VerifyPackageFamilyName 패키지 패밀리 이름이 구문적으로 유효한지 여부를 확인합니다.
VerifyPackageFullName 패키지 전체 이름이 구문적으로 유효한지 여부를 확인합니다.
VerifyPackageId 패키지 ID가 구문적으로 유효한지 여부를 확인합니다.
verifyPackageRelativeApplicationId

패키지 상대 애플리케이션 ID가 구문적으로 유효한지 여부를 확인합니다.

구조

테이블 확장

PACKAGE_ID 이름, 버전 및 게시자와 같은 패키지 식별 정보를 나타냅니다.
PACKAGE_INFO 패키지 식별자, 전체 이름 및 설치 위치를 포함하는 패키지 식별 정보를 나타냅니다.
PACKAGE_VERSION 패키지 버전 정보를 나타냅니다.

열거형

테이블 확장

AddPackageDependencyOptions 패키지 종속성을 추가할 때 적용할 수 있는 옵션을 정의합니다.
AppPolicyClrCompat AppPolicyClrCompat 열거형은 프라이빗 리플렉션을 사용하도록 설정할지 또는 관리되는 개체를 민첩하게 설정할지 결정할 수 있도록 프로세스의 애플리케이션 형식을 나타냅니다.
AppPolicyCreateFileAccess AppPolicyCreateFileAccess 열거형은 프로세스에 IO 디바이스(파일, 파일 스트림, 디렉터리, 실제 디스크, 볼륨, 콘솔 버퍼, 테이프 드라이브, 통신 리소스, mailslot 및 파이프)에 대한 전체 또는 제한된 액세스 권한이 있는지 여부를 나타냅니다.
AppPolicyLifecycleManagement AppPolicyLifecycleManagement 열거형은 프로세스가 수명 주기 관리인지 여부를 나타냅니다.

appPolicyMediaFoundationCodecLoading AppPolicyMediaFoundationCodecLoading 열거형은 프로세스의 정책에서 비 Windows(타사) 플러그 인을 로드할 수 있는지 여부를 나타냅니다.
AppPolicyProcessTerminationMethod AppPolicyProcessTerminationMethod 열거형은 프로세스를 종료하는 데 사용되는 메서드를 나타냅니다.
AppPolicyShowDeveloperDiagnostic AppPolicyShowDeveloperDiagnostic 열거형은 어설션과 같은 개발자 정보를 사용자에게 표시하는 프로세스에 사용되는 메서드를 나타냅니다.
AppPolicyThreadInitializationType AppPolicyThreadInitializationType 열거형은 beginthread[ex]가 스레드를 만들 때 프로세스에 대해 자동으로 수행되어야 하는 초기화의 종류를 나타냅니다.
AppPolicyWindowingModel AppPolicyWindowingModel 열거형은 프로세스에서 CoreWindow 기반 또는 HWND 기반 창 모델을 사용하는지 여부를 나타냅니다.
CreatePackageDependencyOptions TryCreatePackageDependency 함수를 사용하여 패키지 종속성을 만들 때 적용할 수 있는 옵션을 정의합니다.
packageDependencyLifetimeKind 패키지 종속성의 수명을 정의하는 데 사용되는 아티팩트 형식을 나타내는 값을 지정합니다.
packageDependencyProcessorArchitectures TryCreatePackageDependency 함수를 사용하여 만드는 프레임워크 패키지 종속성에 대한 프로세서 아키텍처를 정의합니다.
PackageOrigin 패키지의 원본을 지정합니다.
PackagePathType 검색할 패키지 폴더의 유형을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

appnotify.h 헤더

아티클2023. 08. 22.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

appnotify.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

RegisterAppStateChangeNotification 앱이 콜백 함수를 등록할 수 있도록 하며, 이를 통해 라이브러리가 일시 중단된 상태로 전환되거나 중단되었다는 알림을 받을 수 있습니다.
UnregisterAppStateChangeNotification RegisterAppStateChangeNotification을 통해 등록된 변경 알림을 취소합니다.

콜백 함수

PAPPSTATE_CHANGE_ROUTINE 앱이 일시 중단된 상태로 들어가거나 떠날 때 앱에 알리는 앱 정의 콜백 함수를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

appxpackaging.h 헤더

이 헤더는 Windows 스토어 앱의 패키징, 배포 및 쿼리에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 스토어 앱 패키징, 배포 및 쿼리

appxpackaging.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Interfaces

테이블 확장

IAppxAppInstallerReader 앱 설치 관리자 파일에 대한 XML DOM에 대한 액세스를 제공합니다.
IAppxBlockMapBlock IAppxBlockMapBlock 인터페이스는 앱 패키지에 대한 블록 맵 파일(AppxBlockMap.xml)에 포함된 파일 내의 개별 블록을 나타내는 읽기 전용 개체를 제공합니다.
IAppxBlockMapBlocksEnumerator 블록 맵의 블록을 단일 파일로 열거합니다.
IAppxBlockMapFile 블록 맵의 파일을 나타냅니다.
IAppxBlockMapFilesEnumerator 블록 맵에서 파일을 열거합니다.
IAppxBlockMapReader 파일 특성 및 블록 해시에 대한 액세스를 제공하는 블록 맵의 읽기 전용 개체 모델을 나타냅니다.
IAppxBundleFactory 번들 패키지를 읽고 쓰기 위한 개체를 만듭니다.
IAppxBundleFactory2 번들 파일의 예상 다이제스트를 지정하는 기능을 사용하여 번들 패키지를 읽고 쓰는 개체를 만듭니다.

IAppxBundleFactory3 패키지 내용에 대한 원격 URI 지정 지원을 추가하여 번들 패키지를 읽고 쓰는 개체를 만듭니다.
IAppxBundleManifestOptionalBundleInfo 번들 패키지 매니페스트의 <OptionalBundle> 요소에 대한 읽기 전용 개체 모델을 제공합니다.
IAppxBundleManifestOptionalBundleInfoEnumerator 번들의 선택적 번들 정보를 열거합니다.
IAppxBundleManifestPackageInfo 번들 패키지 매니페스트의 <Package> 요소에 대한 읽기 전용 개체 모델을 제공합니다. (IAppxBundleManifestPackageInfo)
IAppxBundleManifestPackageInfo2 번들 패키지 매니페스트의 <Package> 요소에 대한 읽기 전용 개체 모델을 제공합니다. (IAppxBundleManifestPackageInfo2)
IAppxBundleManifestPackageInfoEnumerator 번들 패키지 매니페스트에 설명된 페이로드 패키지 목록에 대한 읽기 전용 개체 모델을 제공합니다.
IAppxBundleManifestReader 번들 패키지의 매니페스트에 대한 읽기 전용 개체 모델을 제공합니다. (IAppxBundleManifestReader)
IAppxBundleManifestReader2 번들 패키지의 매니페스트에 대한 읽기 전용 개체 모델을 제공합니다. (IAppxBundleManifestReader2)
IAppxBundleReader 번들 패키지에 대한 읽기 전용 개체 모델을 제공합니다.
IAppxBundleReader2 번들 패키지를 읽기 위해 IAppxPackageReader의 인스턴스를 검색하는 기능을 제공합니다.
IAppxBundleWriter 번들 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxBundleWriter)
IAppxBundleWriter2 번들 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxBundleWriter2)

IAppxBundleWriter3 번들 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxBundleWriter3)
IAppxBundleWriter4 번들 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxBundleWriter4)
IAppxContentGroup 콘텐츠 그룹에 대한 정보를 검색합니다.
IAppxContentGroupFilesEnumerator 콘텐츠 그룹 맵에서 콘텐츠 그룹의 파일을 열거합니다.
IAppxContentGroupMapReader 콘텐츠 그룹 맵에 대한 정보를 가져옵니다.
IAppxContentGroupMapWriter 콘텐츠 그룹 맵에 대한 쓰기 전용 개체 모델을 제공합니다.
IAppxContentGroupsEnumerator 콘텐츠 그룹 맵에서 콘텐츠 그룹을 열거합니다.
IAppxDigestProvider 앱 패키징 개체의 다이제스트 문자열 표현을 검색하기 위한 API를 제공합니다.
IAppxEncryptedBundleWriter 암호화된 번들 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxEncryptedBundleWriter)
IAppxEncryptedBundleWriter2 암호화된 번들 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxEncryptedBundleWriter2)
IAppxEncryptedBundleWriter3 암호화된 번들 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxEncryptedBundleWriter3)
IAppxEncryptedPackageWriter 암호화된 앱 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxEncryptedPackageWriter)
IAppxEncryptedPackageWriter2

암호화된 앱 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxEncryptedPackageWriter2)
IAppxEncryptionFactory 패키지와 번들을 암호화, 암호 해독, 읽기 및 쓰기 위한 개체를 만듭니다.
IAppxEncryptionFactory2 Windows 앱 패키지 및 번들을 암호화, 암호 해독, 읽기 및 쓰기 위한 개체를 만듭니다. (IAppxEncryptionFactory2)
IAppxEncryptionFactory3 Windows 앱 패키지 및 번들을 암호화, 암호 해독, 읽기 및 쓰기 위한 개체를 만듭니다. (IAppxEncryptionFactory3)
IAppxEncryptionFactory4 Windows 앱 패키지 및 번들을 암호화하기 위한 개체를 만듭니다.
IAppxEncryptionFactory5 암호화된 Windows 앱 패키지 및 번들을 읽기 위한 개체를 만듭니다.
IAppxFactory 앱 패키지를 읽고 쓰기 위한 개체를 만듭니다. (IAppxFactory)
IAppxFactory2 앱 패키지를 읽고 쓰기 위한 개체를 만듭니다. (IAppxFactory2)
IAppxFactory3 앱 패키지, 매니페스트 및 앱 설치 관리자 파일을 읽고 쓰기 위한 개체를 만듭니다.
IAppxFactory4
IAppxFile 패키지의 페이로드 또는 공간 파일에 대한 정보를 검색합니다.
IAppxFilesEnumerator 패키지의 페이로드 파일을 열거합니다.
IAppxManifestApplication 애플리케이션의 특성 값에 대한 액세스를 제공합니다.

IAppxManifestApplicationsEnumerator 패키지 매니페스트에 정의된 애플리케이션을 열거합니다.
IAppxManifestDeviceCapabilitiesEnumerator 패키지 매니페스트에 정의된 디바이스 기능을 열거합니다.
IAppxManifestMainPackageDependenciesEnumerator 앱 매니페스트의 <MainPackageDependency> 요소를 열거합니다.
IAppxManifestMainPackageDependency 기본 패키지 종속성의 특성 값에 대한 액세스를 제공합니다.
IAppxManifestOptionalPackageInfo 선택적 패키지 정보의 특성 값에 대한 액세스를 제공합니다.
IAppxManifestPackageDependenciesEnumerator 패키지 매니페스트에 정의된 패키지 종속성을 열거합니다.
IAppxManifestPackageDependency 다른 패키지에서 한 패키지의 종속성을 설명합니다. (IAppxManifestPackageDependency)
IAppxManifestPackageDependency2 다른 패키지에서 한 패키지의 종속성을 설명합니다. (IAppxManifestPackageDependency2)
IAppxManifestPackageId 패키지 ID에 대한 액세스를 제공합니다.
IAppxManifestPackageId2 앱 패키지 ID에 대한 액세스를 제공합니다.
IAppxManifestProperties 패키지 매니페스트의 속성 섹션에 대한 읽기 전용 액세스를 제공합니다.
IAppxManifestReader 매니페스트 요소 및 특성에 액세스하는 메서드를 제공하는 패키지 매니페스트의 개체 모델을 나타냅니다. (IAppxManifestReader)

IAppxManifestReader2 매니페스트 요소 및 특성에 액세스하는 메서드를 제공하는 패키지 매니페스트의 개체 모델을 나타냅니다. (IAppxManifestReader2)
IAppxManifestReader5 매니페스트 요소 및 특성에 액세스하는 메서드를 제공하는 패키지 매니페스트의 개체 모델을 나타냅니다. (IAppxManifestReader5)
IAppxManifestReader6 매니페스트 요소 및 특성에 액세스하는 메서드를 제공하는 패키지 매니페스트의 개체 모델을 나타냅니다. (IAppxManifestReader6)
IAppxManifestResourcesEnumerator 패키지 매니페스트에 정의된 리소스를 열거합니다.
IAppxManifestTargetDeviceFamily AppxManifest.xml대상 디바이스 패밀리에 대한 정보를 검색합니다.
IAppxPackageEditor 앱 패키지를 편집하는 기능을 제공합니다.
IAppxPackageReader 앱 패키지에 대한 읽기 전용 개체 모델을 제공합니다.
IAppxPackageWriter 앱 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxPackageWriter)
IAppxPackageWriter2 앱 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxPackageWriter2)
IAppxPackageWriter3 앱 패키지에 대한 쓰기 전용 개체 모델을 제공합니다. (IAppxPackageWriter3)
IAppxSourceContentGroupMapReader 원본 콘텐츠 그룹 맵에 대한 정보를 가져옵니다.

구조

테이블 확장

APPX_ENCRYPTED_EXEMPTIONS Windows 앱 패키지 암호화에서 제외된 파일입니다.
APPX_ENCRYPTED_PACKAGE_SETTINGS 암호화된 Windows 앱 패키지에 대한 설정입니다.
APPX_ENCRYPTED_PACKAGE_SETTINGS2 암호화된 Windows 앱 패키지 설정입니다.
APPX_KEY_INFO Windows 앱 패키지 키 정보입니다.
APPX_PACKAGE_SETTINGS 패키지를 만드는 데 사용되는 패키지 설정을 나타냅니다.
APPX_PACKAGE_WRITER_PAYLOAD_STREAM 앱 패키지에 쓸 파일의 데이터 및 메타데이터를 포함합니다.

열거 목록

테이블 확장

APPX_BUNDLE_FOOTPRINT_FILE_TYPE 번들에 있는 공간 파일의 형식을 지정합니다.
APPX_BUNDLE_PAYLOAD_PACKAGE_TYPE IAppxBundleManifestPackageInfo 개체에 대한 패키지 유형을 지정합니다.
APPX_CAPABILITIES 패키지에서 요청한 기능 또는 권한을 지정합니다.

APPX_COMPRESSION_OPTION 패키지에 파일을 저장하는 데 사용되는 압축 수준을 지정합니다.
APPX_ENCRYPTED_PACKAGE_OPTIONS 암호화된 앱 패키지 옵션입니다.
APPX_FOOTPRINT_FILE_TYPE 패키지의 공간 파일 형식을 지정합니다.
APPX_PACKAGE_ARCHITECTURE 패키지에서 지원하는 프로세서 아키텍처를 지정합니다. (APPX_PACKAGE_ARCHITECTURE)
APPX_PACKAGE_ARCHITECTURE2 패키지에서 지원하는 프로세서 아키텍처를 지정합니다. (APPX_PACKAGE_ARCHITECTURE2)
APPX_PACKAGE_EDITOR_UPDATE_PACKAGE_MANIFEST_OPTIONS 매니페스트를 업데이트할 때 앱 매니페스트 유효성 검사 옵션입니다.
APPX_PACKAGE_EDITOR_UPDATE_PACKAGE_OPTION 앱 패키지를 업데이트할 때 사용할 옵션입니다.

Last updated on 2026. 01. 27.

asyncinfo.h 헤더

아티클2023. 03. 09.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

asyncinfo.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAsyncInfo 비동기 작업을 지원합니다.

열거형

AsyncStatus 비동기 작업의 상태를 나타냅니다.

atacct.h 헤더

아티클2023. 03. 16.

이 헤더는 네트워크 관리에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 관리

atacct.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

GetNetScheduleAccountInformation GetNetScheduleAccountInformation 함수는 AT 서비스 계정 이름을 검색합니다.
SetNetScheduleAccountInformation SetNetScheduleAccountInformation 함수는 AT 서비스 계정 이름 및 암호를 설정합니다. AT 서비스 계정 이름 및 암호는 NetScheduleJobAdd를 사용하여 만든 예약된 작업의 자격 증명으로 사용됩니다.

atlthunk.h 헤더

아티클2023. 08. 22.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

atlthunk.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AtlThunk_AllocateData ATL unk에 대한 메모리 공간을 할당합니다.
AtlThunk_DataToCode AtlThunkData_t 매개 변수에 해당하는 실행 가능한 함수를 반환합니다.
AtlThunk_FreeData ATL unk와 연결된 메모리를 해제합니다.
AtlThunk_InitData ATL unk를 초기화합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

audioapotypes.h 헤더

아티클2023. 03. 09.

이 헤더는 원격 데스크톱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 데스크톱 서비스

audioapotypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

APO_CONNECTION_PROPERTY 동적으로 변경되는 연결 속성을 포함합니다.
APO_CONNECTION_PROPERTY_V2 동적으로 변경되는 연결 속성을 포함합니다. 이 구조의 버전 2에는 IApoAuxiliaryInputConfiguration으로 초기화된 보조 참조 스트림을 동기화하는 데 사용할 수 있는 타임스탬프가 도입되었습니다.

열거형

APO_BUFFER_FLAGS 각 APO 연결과 연결된 APO_CONNECTION_PROPERTY 구조에 대한 버퍼 유효성 검사 플래그를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

audioclient.h 헤더

아티클2023. 03. 22.

이 헤더는 핵심 오디오 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 핵심 오디오 API

audioclient.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAcousticEchoCancellationControl 오디오 캡처 엔드포인트가 AEC(음향 에코 취소)를 지원하는지 여부를 결정하는 메커니즘을 제공하며, 이 경우 클라이언트가 참조 스트림으로 사용해야 하는 오디오 렌더링 엔드포인트를 설정할 수 있도록 합니다.
IAudioCaptureClient IAudioCaptureClient 인터페이스를 사용하면 클라이언트가 캡처 엔드포인트 버퍼에서 입력 데이터를 읽을 수 있습니다.
IAudioClient IAudioClient 인터페이스를 사용하면 클라이언트가 오디오 애플리케이션과 오디오 엔진(공유 모드 스트림의 경우) 또는 오디오 엔드포인트 디바이스의 하드웨어 버퍼(단독 모드 스트림용) 간에 오디오 스트림을 만들고 초기화할 수 있습니다.
IAudioClient2 IAudioClient2 인터페이스는 IAudioClient 인터페이스에서 파생되며, WASAPI(Windows Audio Session API) 오디오 클라이언트가 다음을 수행할 수 있도록 하는 추가 메서드 집합이 있습니다._opt 오프로드, 쿼리 스트림 속성 및 오프로드를 처리하는 하드웨어에서 정보를 가져옵니다. 기본 엔드포인트가 하드웨어 오디오 엔진을 지원하고, 엔드포인트가 오디오 시스템에서 열거되고 검색되었으며, 엔드포인트에서 사용할 수 있는 오프로드 핀 인스턴스가 여전히 있는 경우 오디오 클라이언트가 오프로드된 스트림을 만드는 데 성공할 수 있습니다.
IAudioClient3 IAudioClient3 인터페이스는 IAudioClient2 인터페이스에서 파생되며, WASAPI(Windows Audio Session API) 오디오 클라이언트가 오디오 엔진의 지원되는 주기 및 현재 주기를 쿼리하고 지정된 주기로 공유 오디오 스트림의 초기화를 요청할 수 있는 추가 메서드 집합이 있습니다.

IAudioClientDuckingControl 앱의 오디오 렌더링 스트림이 활성 상태일 때 시스템에서 다른 스트림의 오디오를 더킹하지 않도록 앱이 지정할 수 있는 SetDuckingOptionsForCurrentStream 메서드를 제공합니다.
IAudioClock IAudioClock 인터페이스를 사용하면 클라이언트가 스트림의 데이터 속도와 스트림의 현재 위치를 모니터링할 수 있습니다.
IAudioClock2 IAudioClock2 인터페이스는 현재 디바이스 위치를 가져오는 데 사용됩니다.
IAudioClockAdjustment IAudioClockAdjustment 인터페이스는 스트림의 샘플 속도를 조정하는 데 사용됩니다.
IAudioEffectsChangedNotificationClient 콜백 인터페이스를 사용하면 오디오 효과 목록이 변경되거나 효과 변경을 사용하도록 설정하는 데 필요한 리소스가 변경될 때 애플리케이션에서 알림을 받을 수 있습니다.
IAudioEffectsManager 오디오 효과 파이프라인에 대한 관리 기능을 제공합니다.
IAudioRenderClient IAudioRenderClient 인터페이스를 사용하면 클라이언트가 렌더링 엔드포인트 버퍼에 출력 데이터를 쓸 수 있습니다.
IAudioStreamVolume IAudioStreamVolume 인터페이스를 사용하면 클라이언트가 오디오 스트림의 모든 채널에 대한 볼륨 수준을 제어하고 모니터링할 수 있습니다.
IAudioViewManagerService HWND를 오디오 스트림과 연결하기 위한 API를 제공합니다.
IChannelAudioVolume IChannelAudioVolume 인터페이스를 사용하면 클라이언트가 스트림이 속한 오디오 세션의 모든 채널에 대한 볼륨 수준을 제어하고 모니터링할 수 있습니다.

ISimpleAudioVolume ISimpleAudioVolume 인터페이스를 사용하면 클라이언트가 오디오 세션의 마스터 볼륨 수준을 제어할 수 있습니다.

구조체

AUDIO_EFFECT 오디오 효과를 나타냅니다.
AudioClientProperties AudioClientProperties 구조체(audioclient.h)는 클라이언트 오디오 스트림의 속성을 설명하는 매개 변수를 설정하는 데 사용됩니다.

열거형

_AUDCLNT_BUFFERFLAGS _AUDCLNT_BUFFERFLAGS 열거형은 오디오 엔드포인트 버퍼의 상태를 나타내는 플래그를 정의합니다.
AUDCLNT_STREAMOPTIONS 오디오 스트림의 특성을 설명하는 값을 정의합니다.
AUDIO_DUCKING_OPTIONS 오디오 더킹 옵션을 지정합니다. IAudioClientDuckingControl::SetDuckingOptionsForCurrentStream을 호출할 때 이 열거형의 값을 사용합니다.
AUDIO_EFFECT_STATE 오디오 효과의 상태를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

audioclientactivationparams.h 헤더

아티클2023. 08. 22.

이 헤더는 핵심 오디오 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 핵심 오디오 API

audioclientactivationparams.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

AUDIOCLIENT_ACTIVATION_PARAMS ActivateAudioInterfaceAsync 호출에 대한 활성화 매개 변수를 지정합니다.
AUDIOCLIENT_PROCESS_LOOPBACK_PARAMS 루프백 활성화가 요청되는 ActivateAudioInterfaceAsync 호출에 대한 매개 변수를 지정합니다.

열거형

AUDIOCLIENT_ACTIVATION_TYPE ActivateAudioInterfaceAsync 호출에 전달된 AUDIOCLIENT_ACTIVATION_PARAMS 구조체의 활성화 유형을 지정합니다.
PROCESS_LOOPBACK_MODE ActivateAudioInterfaceAsync 호출에 전달된 AUDIOCLIENT_ACTIVATION_PARAMS 구조체에 대한 루프백 모드를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

audioendpoints.h 헤더

아티클2023. 03. 16.

이 헤더는 핵심 오디오 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 핵심 오디오 API

audioendpoints.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAudioEndpointFormatControl 현재 오디오 엔드포인트 디바이스 형식을 다시 설정하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

audioenginebaseapo.h 헤더

아티클2023. 08. 11.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 오디오 디바이스 DDI 참조
- 핵심 오디오 API

audioenginebaseapo.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IApoAcousticEchoCancellation 이 인터페이스는 AEC(음향 에코 취소) 시나리오를 사용하도록 AMO에 의해 구현됩니다.
IApoAcousticEchoCancellation2 API가 참조 스트림의 원하는 속성을 지정할 수 있도록 IAcousticEchoCancellation을 확장합니다.
IApoAuxiliaryInputConfiguration 오디오 엔진이 보조 입력 스트림을 추가 및 제거할 수 있도록 API가 구현할 수 있는 메서드를 제공합니다.
IApoAuxiliaryInputRT APO의 보조 입력을 구동하는 데 사용되는 실시간 안전 인터페이스입니다.
IAudioDeviceModulesClient API(오디오 처리 개체)는 이 인터페이스를 구현하여 **IAudioDeviceModulesManager** instance 대한 참조를 가져옵니다.
IAudioProcessingObject 시스템 효과 SAPO(오디오 처리 개체)는 일반적으로 실시간 프로세스 스레드에서 사용되거나 호출됩니다.
IAudioProcessingObjectConfiguration IAudioProcessingObjectConfiguration 인터페이스는 APO를 구성하는 데 사용됩니다. 이 인터페이스는 메서드를 사용하여 처리를 위해 APO를 잠그고 잠금을 해제합니다.

IAudioProcessingObjectRT 이 인터페이스는 실시간 모드에서 작동할 수 있으며 해당 메서드를 실시간 처리 스레드 형식이라고 할 수 있습니다.
IAudioSystemEffects IAudioSystemEffects 인터페이스는 IUnknown에서 상속되는 기본 메서드를 사용하며 Initialize 메서드를 구현해야 합니다.
IAudioSystemEffects2 IAudioSystemEffects2 인터페이스는 지정된 모드에서 처리 개체에 대한 정보를 검색하기 위한 Windows 8.1 함께 도입되었습니다.
IAudioSystemEffectsCustomFormats IAudioSystemEffectsCustomFormats 인터페이스는 Windows Vista 이상 버전의 Windows에서 지원됩니다.

구조체

APO_REG_PROPERTIES APO_REG_PROPERTIES 구조체는 IAudioProcessingObject::GetRegistrationProperties에서 APO(오디오 처리 개체)의 등록 속성을 반환하는 데 사용됩니다.
APOInitBaseStruct APOInitBaseStruct 구조체는 IAudioProcessingObject::Initialize의 다른 초기화 데이터 앞에 있어야 하는 기본 초기화 헤더입니다.
APOInitSystemEffects APOInitSystemEffects 구조체는 초기화를 위해 시스템 효과 APO에 전달됩니다.
APOInitSystemEffects2 APOInitSystemEffects2 구조체는 초기화를 위해 APO(오디오 처리 개체)에 추가 초기화 컨텍스트를 제공할 수 있도록 Windows 8.1 도입되었습니다.

열거형

APO_FLAG APO_FLAG 열거형은 APO(오디오 처리 개체)에서 플래그로 사용되는 상수를 정의합니다.
APO_REFERENCE_STREAM_PROPERTIES IApoAcousticEchoCancellation2::GetDesiredReferenceStreamProperties 콜백 메서드에 대한 루프백 스트림 속성을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

audioengineendpoint.h 헤더

아티클2023. 08. 22.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 핵심 오디오 API
- 원격 데스크톱 서비스

audioengineendpoint.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAudioDeviceEndpoint 디바이스 엔드포인트 개체를 초기화하고 디바이스가 나타내는 디바이스의 기능을 가져옵니다.
IAudioEndpoint 오디오 엔드포인트에 대한 정보를 오디오 엔진에 제공합니다. 이 인터페이스는 오디오 엔드포인트에 의해 구현됩니다.
IAudioEndpointControl 엔드포인트의 스트림 상태를 제어합니다.
IAudioEndpointLastBufferControl 오프로드 스트림 클라이언트가 마지막 버퍼가 부분적으로만 전송되었음을 엔드포인트에 알릴 수 있도록 하는 기능을 제공합니다.
IAudioEndpointOffloadStreamMeter IAudioEndpointOffloadStreamMeter 인터페이스는 오프로드된 오디오 스트림의 오디오 채널에 대한 일반 정보를 검색합니다.
IAudioEndpointOffloadStreamMute IAudioEndpointOffloadStreamMute 인터페이스를 사용하면 클라이언트가 오프로드된 오디오 스트림의 음소거 상태 조작할 수 있습니다.
IAudioEndpointOffloadStreamVolume IAudioEndpointOffloadStreamVolume 인터페이스를 사용하면 클라이언트 애플리케이션이 오프로드된 오디오 스트림의 볼륨 수준을 조작할 수 있습니다.

IAudioEndpointRT 엔드포인트 버퍼의 현재 읽기 위치와 쓰기 위치의 차이를 가져옵니다.
IAudioInputEndpointRT 각 처리 패스에 대한 입력 버퍼를 가져옵니다.
IAudioLfxControl IAudioLfxControl 인터페이스를 사용하면 클라이언트가 오프로드된 오디오 스트림에서 로컬 효과를 적용하거나 제거할 수 있습니다.
IAudioOutputEndpointRT 각 처리 패스에 대한 출력 버퍼를 가져옵니다.
IHardwareAudioEngineBase IHardwareAudioEngineBase 인터페이스는 하드웨어 오디오 엔진에 대한 정보를 구성하고 검색하는 데 사용할 오디오 스택의 오디오 엔드포인트에 의해 구현됩니다.

구조체

AE_CURRENT_POSITION 디바이스에서 클라이언트로 현재 프레임 위치를 보고합니다.

열거형

AE_POSITION_FLAGS AE_CURRENT_POSITION 구조체에 대한 상수를 정의합니다. 이러한 상수는 현재 위치의 유효성 정도를 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

audioengineextensionapo.h 헤더

아티클2023. 08. 29.

이 헤더는 오디오 디바이스 DDI 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 오디오 디바이스 DDI 참조

오디오 드라이버와 함께 제공 될 수 있는 API (오디오 처리 개체)에 대 한 Windows 11 API 에 대 한 자세한 내용은 참조 Windows 11 오디오 처리 개체에 대 한 API입니다.

audioengineextensionapo.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAudioProcessingObjectLoggingService API에 대한 로깅 서비스를 나타냅니다.
IAudioProcessingObjectNotifications APO 엔드포인트 및 시스템 효과 알림에 대한 일반적인 오디오 관련 알림을 등록하고 수신하기 위해 클라이언트에서 구현합니다.
IAudioProcessingObjectNotifications2 APO 엔드포인트 및 시스템 효과 알림에 대한 일반적인 오디오 관련 알림을 등록하고 수신하기 위해 클라이언트에서 구현합니다. 이 인터페이스는 현재 디바이스에서 실행 중인 Windows 버전에서 에서 지원되는 알림 유형을 확인하는 기능을 추가합니다.
IAudioProcessingObjectPreferredFormatSupport 이 인터페이스는 API에서 기본 입력 또는 출력 형식을 지정할 수 있도록 구현됩니다.
IAudioProcessingObjectRTQueueService API에 대한 실시간 작업 큐 서비스를 나타냅니다.
IAudioSystemEffects3 APOInitSystemEffects3 구조체를 IAudioProcessingObject::Initialize 메서드에 전달해야 하는 클라이언트에 의해 구현됩니다.

구조체

AcousticEchoCanceller_Reference_Input 현재 AEC(Acoustic Echo Cancellation) 구성과 관련된 확장된 정보를 포함합니다.
APO_NOTIFICATION APO 엔드포인트 또는 시스템 효과에 대한 변경에 대한 알림을 나타냅니다.
APO_NOTIFICATION_DESCRIPTOR 요청된 APO 알림을 지정합니다.
APOInitSystemEffects3 APO 초기화 매개 변수를 제공하여 APOInitSystemEffects2를 확장하여 로깅을 위한 서비스 공급자를 지정하는 기능을 추가합니다.
AUDIO_ENDPOINT_PROPERTY_CHANGE_APO_NOTIFICATION_DESCRIPTOR 엔드포인트 속성 변경 APO 알림을 지정합니다.
AUDIO_ENDPOINT_PROPERTY_CHANGE_NOTIFICATION 속성 변경 APO 알림을 나타냅니다.
AUDIO_ENDPOINT_VOLUME_APO_NOTIFICATION_DESCRIPTOR 엔드포인트 볼륨 APO 알림을 지정합니다.
AUDIO_ENDPOINT_VOLUME_CHANGE_NOTIFICATION 오디오 엔드포인트 볼륨 변경 APO 알림을 나타냅니다.
AUDIO_ENDPOINT_VOLUME_CHANGE_NOTIFICATION2 오디오 엔드포인트 볼륨 변경 APO 알림을 나타냅니다. 볼륨 변경 이벤트에 대한 추가 정보를 제공하는 업데이트된 AUDIO_ENDPOINT_VOLUME_CHANGE_NOTIFICATION 버전입니다.
AUDIO_ENVIRONMENT_STATE_CHANGE_NOTIFICATION 오디오 환경 변경 APO 알림을 나타냅니다.
AUDIO_MICROPHONE_BOOST_APO_NOTIFICATION_DESCRIPTOR 엔드포인트 마이크 부스트 APO 알림을 지정합니다.

AUDIO_MICROPHONE_BOOST_NOTIFICATION 오디오 마이크 부스트 APO 알림을 나타냅니다.
AUDIO_SYSTEMEFFECT SAPO(시스템 효과 오디오 처리 개체) 오디오 효과를 나타냅니다.
AUDIO_SYSTEMEFFECTS_PROPERTY_CHANGE_APO_NOTIFICATION_DESCRIPTOR AUDIO_SYSTEMEFFECTS_PROPERTY_CHANGE_APO_NOTIFICATION_DESCRIPTOR 구조체 (audioengineextensionapo.h)는 시스템 효과 속성 변경 APO 알림을 지정합니다.
AUDIO_SYSTEMEFFECTS_PROPERTY_CHANGE_NOTIFICATION 시스템 오디오 효과 APO 알림을 나타냅니다.
AUDIO_VOLUME_NOTIFICATION_DATA2 볼륨 변경 알림 이벤트에 대한 정보를 나타냅니다. 이 구조체는 AUDIO_ENDPOINT_VOLUME_CHANGE_NOTIFICATION2 구조체에서 사용됩니다.

열거형

APO_LOG_LEVEL IAudioProcessingObjectLoggingService::ApoLog로 기록된 APO 이벤트의 수준을 지정합니다.
APO_NOTIFICATION_TYPE APO_NOTIFICATION 형식을 지정합니다.
AUDIO_SYSTEMEFFECT_STATE SAPO(시스템 효과 오디오 처리 개체) 오디오 효과의 상태를 지정합니다.
DEVICE_ORIENTATION_TYPE 형식 APO_NOTIFICATION_TYPE_DEVICE_ORIENTATION 알림에 대한 디바이스 방향 값을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

audiomediatype.h 헤더

아티클2023. 03. 09.

이 헤더는 오디오 디바이스 DDI 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 오디오 디바이스 DDI 참조

audiomediatype.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAudioMediaType IAudioMediaType 인터페이스는 SAPO가 적절한 오디오 데이터 형식에 대해 오디오 엔진과 협상하는 데 사용되는 정보를 가져올 수 있도록 하는 메서드를 노출합니다.

Functions

CreateAudioMediaType CreateAudioMediaType 함수는 호출자가 지정한 형식을 사용하여 오디오 형식을 설명하는 미디어 형식 개체를 만듭니다.
CreateAudioMediaTypeFromUncompressedAudioFormat CreateAudioMediaTypeFromUncompressedAudioFormat 함수는 UNCOMPRESSEDAUDIOFORMAT 구조에 제공된 정보를 사용하여 오디오 형식을 설명하는 미디어 형식 개체를 만듭니다.

구조체

UNCOMPRESSEDAUDIOFORMAT UNCOMPRESSEDAUDIOFORMAT 구조체는 프레임 속도, 채널 마스크 및 압축되지 않은 오디오 데이터 형식의 기타 특성을 지정합니다.

audiopolicy.h 헤더

아티클2023. 03. 16.

이 헤더는 핵심 오디오 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 핵심 오디오 API

audiopolicy.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAudioSessionControl IAudioSessionControl 인터페이스를 사용하면 클라이언트가 오디오 세션에 대한 컨트롤 매개 변수를 구성하고 세션의 이벤트를 모니터링할 수 있습니다.
IAudioSessionControl2 IAudioSessionControl2 인터페이스는 클라이언트에서 오디오 세션에 대한 정보를 가져오는 데 사용할 수 있습니다.
IAudioSessionEnumerator IAudioSessionEnumerator 인터페이스는 오디오 디바이스의 오디오 세션을 열거합니다.
IAudioSessionEvents IAudioSessionEvents 인터페이스는 볼륨 수준 변경, 표시 이름 및 세션 상태와 같은 세션 관련 이벤트에 대한 알림을 제공합니다.
IAudioSessionManager IAudioSessionManager 인터페이스를 사용하면 클라이언트가 프로세스 간 및 프로세스별 오디오 세션 모두에 대한 세션 컨트롤 및 볼륨 컨트롤에 액세스할 수 있습니다.
IAudioSessionManager2 IAudioSessionManager2 인터페이스를 사용하면 애플리케이션이 오디오 디바이스에 대한 서브믹스를 관리할 수 있습니다.
IAudioSessionNotification IAudioSessionNotification 인터페이스는 오디오 세션을 만들 때 알림을 제공합니다.

IAudioVolumeDuckNotification IAudioVolumeDuckNotification 인터페이스는 시스템에서 스트림 감쇠 변경 내용에 대한 알림을 보내는 데 사용됩니다. 스트림 감쇠 또는 더킹은 Windows 7에서 도입된 기능으로, 새 통신 스트림이 열릴 때 시스템이 비 통신 스트림의 볼륨을 조정합니다. 이 기능에 대한 자세한 내용은 기본 더킹 환경을 참조하세요.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

audiosessiontypes.h 헤더

아티클2023. 08. 22.

이 헤더는 핵심 오디오 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 핵심 오디오 API

audiosessiontypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

열거형

AUDCLNT_SHAREMODE AUDCLNT_SHAREMODE 열거형은 오디오 스트림이 공유 모드에서 실행되는지 아니면 배타적 모드에서 실행되는지를 나타내는 상수를 정의합니다.
AUDIO_STREAM_CATEGORY 오디오 스트림의 범주를 지정합니다.
AudioSessionState AudioSessionState 열거형은 오디오 세션의 현재 상태를 나타내는 상수를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

audiostatemonitorapi.h 헤더

아티클2023. 06. 23.

이 헤더는 핵심 오디오 기능의 일부로 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 핵심 오디오 API

audiostatemonitorapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAudioStateMonitor 오디오 스트림의 소리 수준을 쿼리하고 소리 수준이 변경되면 알림을 수신하기 위한 API를 제공합니다.

Functions

CreateCaptureAudioStateMonitor 캡처 스트림에 대한 IAudioStateMonitor의 새 instance 만듭니다.
CreateCaptureAudioStateMonitorForCategory 지정된 오디오 범주를 사용하여 캡처 스트림에 대한 IAudioStateMonitor의 새 instance 만듭니다.
CreateCaptureAudioStateMonitorForCategoryAndDeviceId 지정된 오디오 범주 및 오디오 디바이스 ID를 사용하여 캡처 스트림에 대한 IAudioStateMonitor의 새 instance 만듭니다.
CreateCaptureAudioStateMonitorForCategoryAndDeviceRole 지정된 오디오 범주 및 오디오 디바이스 역할을 사용하여 캡처 스트림에 대한 IAudioStateMonitor의 새 instance 만듭니다.
CreateRenderAudioStateMonitor 렌더링 스트림에 대한 IAudioStateMonitor의 새 instance 만듭니다.

CreateRenderAudioStateMonitorForCategory 지정된 오디오 범주를 사용하여 렌더링 스트림에 대한 IAudioStateMonitor의 새 instance 만듭니다.
CreateRenderAudioStateMonitorForCategoryAndDeviceId 지정된 오디오 범주 및 오디오 디바이스 ID를 사용하여 렌더링 스트림에 대한 IAudioStateMonitor의 새 instance 만듭니다.
CreateRenderAudioStateMonitorForCategoryAndDeviceRole 지정된 오디오 범주 및 오디오 디바이스 역할을 사용하여 렌더링 스트림에 대한 IAudioStateMonitor의 새 instance 만듭니다.

콜백 함수

AudioStateMonitorCallback 시스템이 IAudioStreamStateMonitor에서 모니터링하는 오디오 스트림의 사운드 수준을 변경할 때 발생합니다.

열거형

AudioStateMonitorSoundLevel

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

authif.h 헤더

아티클2023. 08. 22.

이 헤더는 네트워크 정책 서버에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 정책 서버

authif.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

콜백 함수

PRADIUS_EXTENSION_FREE_ATTRIBUTES RadiusExtensionFreeAttributes 함수는 애플리케이션 정의 함수이며, RadiusExtensionProcessEx에서 반환된 특성에서 차지하는 메모리를 해제하기 위해 NPS에서 호출됩니다.
PRADIUS_EXTENSION_INIT RadiusExtensionInit 함수는 애플리케이션 정의 함수이며 서비스가 시작되는 동안 NPS에서 호출됩니다. RadiusExtensionInit를 사용하여 확장 DLL에 대한 초기화 작업을 수행합니다.
PRADIUS_EXTENSION_PROCESS RadiusExtensionProcess 함수는 애플리케이션 정의 함수이며 NPS가 NAS(네트워크 액세스 서버)에서 수신하는 각 인증 또는 회계 패킷에 대해 NPS에서 호출됩니다.
PRADIUS_EXTENSION_PROCESS_2 애플리케이션 정의 함수이며 NPS가 수신하는 각 인증 또는 회계 패킷에 대해 NPS에서 호출합니다.
PRADIUS_EXTENSION_PROCESS_EX RadiusExtensionProcessEx 함수는 애플리케이션 정의 함수이며 NPS가 NAS(네트워크 액세스 서버)에서 수신하는 각 인증 또는 회계 패킷에 대해 NPS에서 호출됩니다.
PRADIUS_EXTENSION_TERM RadiusExtensionTerm 함수는 애플리케이션 정의 함수이며 확장 DLL을 언로드하기 전에 NPS에서 호출됩니다. RadiusExtensionTerm을 사용하여 확장 DLL에 대한 클린 작업을 수행합니다.

구조체

RADIUS_ATTRIBUTE RADIUS_ATTRIBUTE 구조체는 RADIUS 특성 또는 확장 특성을 나타냅니다.
RADIUS_ATTRIBUTE_ARRAY RADIUS_ATTRIBUTE_ARRAY 구조체는 특성 배열을 나타냅니다.
RADIUS_EXTENSION_CONTROL_BLOCK RADIUS_EXTENSION_CONTROL_BLOCK 구조는 현재 RADIUS 요청에 대한 정보를 제공합니다. 또한 요청과 연결된 특성을 가져오고 요청의 처리 설정을 위한 함수를 제공합니다.
RADIUS_VSA_FORMAT RADIUS_VSA_FORMAT 구조체는 RADIUS 공급업체별 특성의 문자열 부분 형식을 나타냅니다.

열거형

RADIUS_ACTION RADIUS_ACTION 형식은 Access-Request에 대한 응답으로 NPS 확장 DLL이 생성할 수 있는 응답을 열거합니다.
RADIUS_ATTRIBUTE_TYPE RADIUS_ATTRIBUTE_TYPE 형식은 RADIUS 특성에 대해 가능한 형식을 열거합니다.
RADIUS_AUTHENTICATION_PROVIDER RADIUS_AUTHENTICATION_PROVIDER 형식은 NPS에서 사용할 수 있는 가능한 인증 공급자를 열거합니다.
RADIUS_CODE RADIUS_CODE 열거형 형식은 가능한 RADIUS 패킷 코드를 열거합니다.
RADIUS_DATA_TYPE RADIUS_DATA_TYPE 형식은 RADIUS 특성 또는 확장 특성에 대해 가능한 데이터 형식을 열거합니다.
RADIUS_EXTENSION_POINT

RADIUS_EXTENSION_POINT 열거형 형식은 RadiusExtensionProcess2 함수를 호출할 수 있는 경우 RADIUS 요청 프로세스에서 가능한 지점을 열거합니다.
RADIUS_REJECT_REASON_CODE RADIUS_REJECT_REASON_CODE 열거형은 가능한 RADIUS 패킷 거부 코드를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

authz.h 헤더

아티클2023. 03. 16.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

authz.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

AuthzAccessCheck 지정된 보안 설명자 집합에 대해 클라이언트에 부여할 수 있는 액세스 비트를 결정합니다.
AuthzAddSidsToContext 기존 컨텍스트의 복사본을 만들고 지정된 SID(보안 식별자) 및 제한된 SID 집합을 추가합니다.
AuthzCachedAccessCheck 이전 AuthzAccessCheck 호출에서 정적 부여 비트를 포함하는 캐시된 핸들에 따라 빠른 액세스 검사를 수행합니다.
AuthzEnumerateSecurityEventSources 기본적으로 설치되지 않은 등록된 보안 이벤트 원본을 검색합니다.
AuthzFreeAuditEvent AuthzInitializeObjectAccessAuditEvent 함수에 의해 할당된 구조를 해제합니다.
AuthzFreeCentralAccessPolicyCache CAP 캐시의 할당을 취소할 수 있도록 CAP 캐시 참조 수를 1씩 줄입니다.
AuthzFreeContext 클라이언트 컨텍스트와 연결된 모든 구조체 및 메모리를 해제합니다. 클라이언트에 대한 핸들 목록은 이 호출에서 해제됩니다.
AuthzFreeHandle 핸들 목록에서 핸들을 찾아 삭제합니다.

AuthzFreeResourceManager Resource Manager 개체를 해제합니다.
AuthzGetInformationFromContext Authz 컨텍스트에 대한 정보를 반환합니다.
AuthzInitializeCompoundContext 지정된 사용자 및 디바이스 보안 컨텍스트에서 사용자 모드 컨텍스트를 만듭니다.
AuthzInitializeContextFromAuthzContext 기존 클라이언트 컨텍스트를 기반으로 새 클라이언트 컨텍스트를 만듭니다.
AuthzInitializeContextFromSid SID(사용자 보안 식별자)에서 사용자 모드 클라이언트 컨텍스트를 만듭니다.
AuthzInitializeContextFromToken 커널 토큰에서 클라이언트 권한 부여 컨텍스트를 초기화합니다. 커널 토큰은 TOKEN_QUERY 위해 열려 있어야 합니다.
AuthzInitializeObjectAccessAuditEvent 개체에 대한 감사를 초기화합니다.
AuthzInitializeObjectAccessAuditEvent2 AuthzAccessCheck 함수에 사용할 AUTHZ_AUDIT_EVENT_HANDLE 핸들을 할당하고 초기화합니다.
AuthzInitializeRemoteResourceManager 원격 리소스 관리자를 할당하고 초기화합니다. 호출자는 결과 핸들을 사용하여 서버에 구성된 리소스 관리자의 원격 인스턴스에 대한 RPC 호출을 수행할 수 있습니다.
AuthzInitializeResourceManager Authz를 사용하여 클라이언트가 다양한 리소스에 액세스할 수 있는지 확인합니다.
AuthzInitializeResourceManagerEx 리소스 관리자 구조를 할당하고 초기화합니다.
AuthzInstallSecurityEventSource 지정된 소스를 보안 이벤트 원본으로 설치합니다.

AuthzModifyClaims Authz 클라이언트 컨텍스트에서 사용자 및 디바이스 클레임을 추가, 삭제 또는 수정합니다.
AuthzModifySecurityAttributes 지정된 클라이언트 컨텍스트에서 보안 특성 정보를 수정합니다.
AuthzModifySids Authz 클라이언트 컨텍스트에서 사용자 및 디바이스 그룹을 추가, 삭제 또는 수정합니다.
AuthzOpenObjectAudit 지정된 보안 설명자의 SACL(시스템 액세스 제어 목록)을 읽고 해당 SACL에서 지정한 적절한 감사를 생성합니다.
AuthzRegisterCapChangeNotification CAP 업데이트 알림 콜백을 등록합니다.
AuthzRegisterSecurityEventSource LSA(로컬 보안 기관)에 보안 이벤트 원본을 등록합니다.
AuthzReportSecurityEvent 등록된 보안 이벤트 원본에 대한 보안 감사를 생성합니다.
AuthzReportSecurityEventFromParams 지정된 감사 매개 변수 배열을 사용하여 등록된 보안 이벤트 원본에 대한 보안 감사를 생성합니다.
AuthzSetAppContainerInformation 현재 Authz 컨텍스트에서 앱 컨테이너 및 기능 정보를 설정합니다.
AuthzUninstallSecurityEventSource 유효한 보안 이벤트 원본 목록에서 지정된 소스를 제거합니다.
AuthzUnregisterCapChangeNotification 이전에 등록된 CAP 업데이트 알림 콜백을 제거합니다.
AuthzUnregisterSecurityEventSource LSA(로컬 보안 기관)를 사용하여 보안 이벤트 원본을 등록 취소합니다.

구조체

AUTHZ_ACCESS_REPLY 액세스 확인 회신을 정의합니다.
AUTHZ_ACCESS_REQUEST 액세스 확인 요청을 정의합니다.
AUTHZ_INIT_INFO 리소스 관리자에 대한 초기화 정보를 정의합니다.
AUTHZ_REGISTRATION_OBJECT_TYPE_NAME_OFFSET 등록 개체 형식 이름의 오프셋을 지정합니다.
AUTHZ_RPC_INIT_INFO_CLIENT 클라이언트에 대한 원격 리소스 관리자를 초기화합니다.
AUTHZ_SECURITY_ATTRIBUTE_FQBN_VALUE 보안 특성과 연결된 정규화된 이진 이름 값을 지정합니다.
AUTHZ_SECURITY_ATTRIBUTE_OCTET_STRING_VALUE 보안 특성에 대한 8진수 문자열 값을 지정합니다.
AUTHZ_SECURITY_ATTRIBUTE_V1 권한 부여 컨텍스트와 연결할 수 있는 보안 특성을 정의합니다.
AUTHZ_SECURITY_ATTRIBUTES_INFORMATION 하나 이상의 보안 특성 및 값을 지정합니다.
AUTHZ_SOURCE_SCHEMA_REGISTRATION 원본 스키마 등록에 대한 정보를 지정합니다.

열거형

AUTHZ_CONTEXT_INFORMATION_CLASS 기존 AuthzClientContext에서 검색할 정보의 형식을 지정합니다. 이 열거형은 AuthzGetInformationFromContext 함수에서 사용됩니다.
AUTHZ_SECURITY_ATTRIBUTE_OPERATION AuthzModifySecurityAttributes 함수를 호출하여 보안 특성에 대한 수정 유형을 나타냅니다.
AUTHZ_SID_OPERATION AuthzModifySids 함수를 호출하여 수행할 수 있는 SID 작업의 형식을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

aux_ulib.h 헤더

아티클2023. 03. 09.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

aux_ulib.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

AuxUlibInitialize Aux_ulib 라이브러리를 초기화합니다.
AuxUlibIsDLLSynchronizationHeld 호출자가 동기화 기본 형식을 보유하고 있는지 여부를 확인합니다.
AuxUlibSetSystemFileCacheSize 현재 파일 시스템 캐시 크기를 설정합니다.

avrfsdk.h 헤더

아티클2023. 08. 22.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

avrfsdk.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

VerifierEnumerateResource 디버깅 및 지원 도구에서 사용할 운영 체제 리소스를 열거합니다.
VerifierIsPerUserSettingsEnabled 애플리케이션 검증 도구가 사용자별 설정을 사용할 수 있는지 여부를 결정합니다.

콜백 함수

AVRF_HANDLEOPERATION_ENUMERATE_CALLBACK 핸들 추적의 열거와 관련된 정보를 받습니다.
AVRF_HEAPALLOCATION_ENUMERATE_CALLBACK 힙 할당과 관련된 정보를 받습니다.
AVRF_RESOURCE_ENUMERATE_CALLBACK 힙 할당 또는 핸들 추적 정보를 열거하기 위한 특수 콜백 함수 중 하나에 대한 액세스를 제공합니다.

구조체

AVRF_BACKTRACE_INFORMATION 수행된 백트레이스에 대한 정보를 포함합니다.
AVRF_HANDLE_OPERATION 핸들 추적 정보를 수집하는 데 필요한 정보를 포함합니다.
AVRF_HEAP_ALLOCATION 힙 할당에 대한 메타데이터 정보를 저장합니다.

열거형

eAvrfResourceTypes VerifierEnumerateResource 함수를 사용하여 열거할 수 있는 리소스 유형을 지정합니다.
eHANDLE_TRACE_OPERATIONS 발생한 핸들 작업의 유형을 식별합니다.
eHeapAllocationState 현재 힙 할당 상태를 지정합니다.
eHeapEnumerationLevel 열거형 작업을 계속할지 중지할지 여부를 결정합니다.
eUserAllocationState 애플리케이션의 현재 힙 할당 상태를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

avrt.h 헤더

아티클2023. 08. 22.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

avrt.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AvQuerySystemResponsiveness 멀티미디어 클래스 스케줄러 서비스에서 사용하는 시스템 응답성 설정을 검색합니다.
AvRevertMmThreadCharacteristics 스레드가 지정된 작업과 연결된 작업을 더 이상 수행하지 않음을 나타냅니다.
AvRtCreateThreadOrderingGroup 스레드 순서 그룹을 만듭니다.
AvRtCreateThreadOrderingGroupExA 스레드 순서 그룹을 만들고 서버 스레드를 작업에 연결합니다. (ANSI)
AvRtCreateThreadOrderingGroupExW 스레드 순서 그룹을 만들고 서버 스레드를 작업에 연결합니다. (유니코드)
AvRtDeleteThreadOrderingGroup 호출자가 만든 지정된 스레드 순서 그룹을 삭제합니다. 컨텍스트 정보를 포함하여 스레드 순서 그룹에 대한 리소스를 정리하고 를 반환합니다.
AvRtJoinThreadOrderingGroup 클라이언트 스레드를 스레드 순서 그룹에 조인합니다.
AvRtLeaveThreadOrderingGroup 클라이언트 스레드가 스레드 순서 그룹을 떠날 수 있도록 합니다.
AvRtWaitOnThreadOrderingGroup

스레드 순서 그룹의 클라이언트 스레드가 실행될 때까지 기다릴 수 있도록 합니다.
AvSetMmMaxThreadCharacteristicsA 호출 스레드를 지정된 태스크와 연결합니다. (ANSI)
AvSetMmMaxThreadCharacteristicsW 호출 스레드를 지정된 태스크와 연결합니다. (유니코드)
AvSetMmThreadCharacteristicsA 호출 스레드를 지정된 태스크와 연결합니다. (ANSI)
AvSetMmThreadCharacteristicsW 호출 스레드를 지정된 태스크와 연결합니다. (유니코드)
AvSetMmThreadPriority 동일한 작업을 수행하는 다른 스레드를 기준으로 호출 스레드의 스레드 우선 순위를 조정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

azroles.h 헤더

아티클2023. 08. 22.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

azroles.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAzApplication 애플리케이션의 설치된 instance 정의합니다. 애플리케이션이 설치될 때 IAzApplication 개체가 만들어집니다.
IAzApplication2 IAzApplication 인터페이스에서 상속되며 IAzClientContext2 개체를 초기화하는 추가 메서드를 구현합니다.
IAzApplication3 IAzRoleAssignment, IAzRoleDefinition 및 IAzScope2 개체를 관리하는 메서드를 제공합니다.
IAzApplicationGroup 보안 주체 컬렉션을 정의합니다.
IAzApplicationGroup2 BizRule 그룹 형식에 대한 지원을 추가하여 IAzApplicationGroup 인터페이스를 확장합니다.
IAzApplicationGroups IAzApplicationGroup 개체의 컬렉션을 나타냅니다.
IAzApplications IAzApplication 개체의 컬렉션을 나타냅니다.
IAzAuthorizationStore 권한 부여 정책 저장소의 루트인 컨테이너를 정의합니다.

IAzAuthorizationStore2 AzAuthorizationStore 개체에서 상속되며 IAzApplication2 개체를 만들고 여는 메서드를 구현합니다.
IAzAuthorizationStore3 비즈니스 규칙(BizRule) 지원 및 캐싱을 관리하는 메서드를 사용하여 IAzAuthorizationStore2 인터페이스를 확장합니다.
IAzBizRuleContext BizRule(비즈니스 규칙) 작업에 대한 정보를 포함합니다.
IAzBizRuleInterfaces 비즈니스 규칙(BizRule) 스크립트에서 호출할 수 있는 IDispatch 인터페이스 목록을 관리하는 데 사용되는 메서드 및 속성을 제공합니다.
IAzBizRuleParameters 비즈니스 규칙(BizRule) 스크립트에 전달할 수 있는 매개 변수 목록을 관리하는 데 사용되는 메서드 및 속성을 제공합니다.
IAzClientContext 특정 클라이언트를 설명하는 상태를 유지 관리합니다.
IAzClientContext2 IAzClientContext 인터페이스에서 상속되며 클라이언트 컨텍스트를 조작하는 새 메서드를 구현합니다.
IAzClientContext3 IAzClientContext2 인터페이스를 확장합니다.
IAzNameResolver SID(보안 식별자)를 보안 표시 이름으로 변환합니다.
IAzObjectPicker 사용자가 목록에서 하나 이상의 보안 주체를 선택할 수 있는 대화 상자를 표시합니다.
IAzOperation 애플리케이션에서 지원하는 하위 수준 작업을 정의합니다.

IAzOperation2 작업과 연결된 역할 할당을 반환하는 메서드를 사용하여 IAzOperation을 확장합니다.
IAzOperations IAzOperation 개체의 컬렉션을 나타냅니다.
IAzPrincipalLocator 권한 부여 관리자에서 ADAM 보안 주체를 찾아 선택합니다.
IAzRole scope 내의 사용자 집합에서 수행할 수 있는 작업 집합을 정의합니다.
IAzRoleAssignment 사용자 및 그룹을 할당할 수 있는 역할을 나타냅니다.
IAzRoleAssignments IAzRoleAssignment 개체의 컬렉션을 나타냅니다.
IAzRoleDefinition 작업 집합을 지정하는 하나 이상의 IAzRoleDefinition, IAzTask 및 IAzOperation 개체를 나타냅니다.
IAzRoleDefinitions IAzRoleDefinition 개체의 컬렉션을 나타냅니다.
IAzRoles IAzRole 개체의 컬렉션을 나타냅니다.
IAzScope 애플리케이션이 액세스를 관리하는 리소스의 논리적 컨테이너를 정의합니다.
IAzScope2 IAzScope 인터페이스를 확장하여 IAzRoleAssignment 및 IAzRoleDefinition 개체를 관리합니다.
IAzScopes IAzScope 개체의 컬렉션을 나타냅니다.
IAzTask

작업 집합에 대해 설명합니다.
IAzTask2 작업과 연결된 역할 할당을 반환하는 메서드를 사용하여 IAzTask 인터페이스를 확장합니다.
IAzTasks IAzTask 개체의 컬렉션을 나타냅니다.

열거형

AZ_PROP_CONSTANTS Authorization Manager에서 사용하는 상수를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

baseaudioprocessingobject.h 헤더

아티클2023. 08. 22.

이 헤더는 오디오 디바이스 DDI 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 오디오 디바이스 DDI 참조

baseaudioprocessingobject.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AERT_Allocate AERT_Allocate 유틸리티 함수는 오디오 처리 개체에서 사용할 메모리 세그먼트를 할당하고 잠가줍니다.
AERT_Free AERT_Free 유틸리티 함수는 오디오 처리 개체가 오디오 데이터를 처리하는 데 사용하기 위해 AERT_Allocate 함수에 의해 잠긴 메모리를 해제합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

batclass.h 헤더

아티클2024. 11. 18.

이 헤더는 배터리 장치 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 배터리 장치 참조

batclass.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

테이블 확장

BatteryClassInitializeDevice BatteryClassInitializeDevice 루틴은 클래스 드라이버에 대한 새 배터리 디바이스를 초기화합니다.
BatteryClassIoctl BatteryClassIoctl은 시스템 정의 배터리 IOCTL을 처리합니다.
BatteryClassQueryWmiDataBlock BatteryClassQueryWmiDataBlock 루틴은 배터리 클래스 드라이버가 드라이버를 대신하여 처리하는 WMI 데이터 블록 쿼리 요청을 처리할 수 있도록 DpWmiQueryDataBlock 루틴 내의 배터리 미니클래스 드라이버에서 사용됩니다.
BatteryClassStatusNotify BatteryClassStatusNotify는 배터리 클래스 드라이버에 배터리 상태 변경을 알립니다.
BatteryClassSystemControl BatteryClassSystemControl 루틴은 배터리 미니클래스 드라이버를 대신하여 WMI IRP를 처리합니다.
BatteryClassUnload BatteryClassUnload는 더 이상 사용되지 않는 배터리 디바이스에 대한 리소스를 해제합니다.
BatteryClassUpdateData BatteryClassUpdateData는 배터리 데이터가 변경될 때 클래스 드라이버에 알립니다. 이 메커니즘은 이벤트/상태를 보고하는 미니포트용입니다.

콜백 함수

테이블 확장

BCLASS_DISABLE_STATUS_NOTIFY_CALLBACK BatteryMiniDisableStatusNotify는 배터리 디바이스에 대한 상태 알림을 사용하지 않도록 설정합니다.
BCLASS_QUERY_INFORMATION_CALLBACK BatteryMiniQueryInformation은 지정된 배터리 디바이스에 대한 정보를 반환합니다.
BCLASS_QUERY_STATUS_CALLBACK BatteryMiniQueryStatus는 지정된 배터리 디바이스에 대한 상태 정보를 반환합니다.
BCLASS_QUERY_TAG_CALLBACK BatteryMiniQueryTag는 현재 배터리 태그를 반환합니다.
BCLASS_SET_INFORMATION_CALLBACK BatteryMiniSetInformation은 배터리가 충전 또는 방전 상태로 들어가거나 배터리에 대한 중요한 바이어스 값을 설정하라는 요청을 합니다.
BCLASS_SET_STATUS_NOTIFY_CALLBACK BatteryMiniSetStatusNotify는 클래스 드라이버에 알림이 필요한 배터리 용량 및 전원 상태 수준을 설정합니다.

구조

테이블 확장

BATTERY_MINIPORT_INFO 배터리 미니 클래스 드라이버는 배터리 클래스 드라이버의 BatteryClassInitializeDevice 루틴을 호출하기 전에이 구조를 채웁니다.
BATTERY_MINIPORT_INFO_V1_1 배터리 미니 클래스 드라이버는 배터리 클래스 드라이버의 BatteryClassInitializeDevice 루틴을 호

출하기 전에 BATTERY_MINIPORT_INFO_V1_1 구조를 채웁니다. BATTERY_MINIPORT_INFO_V1_1 이전 구조 BATTERY_MINIPORT_INFO 업데이트된 버전입니다.
BATTERY_NOTIFY 배터리 미니 클래스 드라이버는 BatteryMiniSetStatusNotify 루틴이 호출될 때 BATTERY_NOTIFY 구조를 받습니다.
BATTERY_TAG_CHANGE 이 구조는 시스템 사용을 위해 예약되어 있습니다. (BATTERY_TAG_CHANGE)
BATTERY_WMI_CYCLE_COUNT BatteryClassQueryWmiDataBlock 함수에 사용할 배터리의 충전 주기 수에 대한 정보를 정의합니다.
BATTERY_WMI_FULL_CHARGED_CAPACITY BatteryClassQueryWmiDataBlock과 함께 사용할 배터리 용량에 대한 정보를 정의합니다.)
BATTERY_WMI_RUNTIME BatteryClassQueryWmiDataBlock 함수에 사용할 배터리의 예상 런타임에 대한 정보를 정의합니다.
BATTERY_WMI_STATIC_DATA 배터리에 대한 정적 데이터를 정의합니다.
BATTERY_WMI_STATUS 배터리 상태 정보를 정의합니다.
BATTERY_WMI_STATUS_CHANGE 이 구조는 시스템 사용을 위해 예약되어 있습니다. (BATTERY_WMI_STATUS_CHANGE)
BATTERY_WMI_TEMPERATURE BatteryClassQueryWmiDataBlock 함수와 함께 사용할 배터리 온도에 대한 정보를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

bcp47mrm.h 헤더

아티클2023. 08. 22.

BCP-47 언어 태그를 사용하기 위한 함수를 제공합니다.

이 헤더는 메뉴 및 기타 리소스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 메뉴 및 기타 리소스

bcp47mrm.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

GetDistanceOfClosestLanguageInList 지정된 언어 코드와 언어 목록에서 가장 가까운 일치 항목 사이의 거리를 결정합니다.
IsWellFormedTag BCP-47 언어 태그가 올바른 형식인지 여부를 확인합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

bcrypt.h 헤더

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

bcrypt.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

BCRYPT_INIT_AUTH_MODE_INFO BCryptEncrypt 및 BCryptDecrypt 함수 호출에 사용할 BCRYPT_AUTHENTICATED_CIPHER_MODE_INFO 구조를 초기화합니다.
BCryptAddContextFunction 기존 CNG 컨텍스트에서 지원하는 함수 목록에 암호화 함수를 추가합니다.
BCryptCloseAlgorithmProvider 알고리즘 공급자를 닫습니다.
BCryptConfigureContext 기존 CNG 컨텍스트에 대한 구성 정보를 설정합니다.
BCryptConfigureContextFunction 기존 CNG 컨텍스트의 암호화 함수에 대한 구성 정보를 설정합니다.
BCryptCreateContext 새 CNG 구성 컨텍스트를 만듭니다.
BCryptCreateHash 해시 또는 MAC(메시지 인증 코드) 개체를 만들기 위해 호출됩니다.
BCryptCreateMultiHash BCryptCreateMultiHash 함수는 여러 해시 작업의 병렬 계산을 허용하는 다중 해시 상태를 만듭니다.
BCryptDecrypt

데이터 블록을 해독합니다.
BCryptDeleteContext 기존 CNG 구성 컨텍스트를 삭제합니다.
BCryptDeriveKey 비밀 핸들에서 키를 파생합니다. (BCryptDeriveKey)
BCryptDeriveKeyCapi 해시 값에서 키를 파생합니다.
BCryptDeriveKeyPBKDF2 RFC 2898에 정의된 대로 PBKDF2 키 파생 알고리즘을 사용하여 해시 값에서 키를 파생합니다.
BCryptDestroyHash 해시 또는 MAC(메시지 인증 코드) 개체를 제거합니다.
BCryptDestroyKey 키를 삭제합니다.
BCryptDestroySecret BCryptSecretAgreement 함수를 사용하여 만든 비밀 계약 핸들을 삭제합니다.
BCryptDuplicateHash 기존 해시 또는 MAC(메시지 인증 코드) 개체를 복제합니다.
BCryptDuplicateKey 대칭 키의 중복을 만듭니다.
BCryptEncrypt 데이터 블록을 암호화합니다. (BCryptEncrypt)
BCryptEnumAlgorithms 등록된 알고리즘 식별자 목록을 가져옵니다.
BCryptEnumContextFunctionProviders 지정된 구성 테이블의 컨텍스트에 대한 암호화 함수에 대한 공급자를 가져옵니다.

BCryptEnumContextFunctions 지정된 구성 테이블의 컨텍스트에 대한 암호화 함수를 가져옵니다.
BCryptEnumContexts 지정된 구성 테이블에 있는 컨텍스트의 식별자를 가져옵니다.
BCryptEnumProviders 지정된 알고리즘을 지원하는 모든 CNG 공급자를 가져옵니다.
BCryptEnumRegisteredProviders 등록된 공급자에 대한 정보를 검색합니다.
BCryptExportKey 나중에 사용할 수 있도록 유지할 수 있는 메모리 BLOB에 키를 내보냅니다.
BCryptFinalizeKeyPair 퍼블릭/프라이빗 키 쌍을 완료합니다.
BCryptFinishHash BCryptHashData에 대한 이전 호출에서 누적된 데이터에 대한 해시 또는 MAC(메시지 인증 코드) 값을 검색합니다.
BCryptFreeBuffer CNG 함수 중 하나에서 할당한 메모리를 해제하는 데 사용됩니다.
BCryptGenerateKeyPair 빈 공개/프라이빗 키 쌍을 만듭니다.
BCryptGenerateSymmetricKey 제공된 키에서 대칭 키 암호화 알고리즘과 함께 사용할 키 개체를 만듭니다.
BCryptGenRandom 난수를 생성합니다.
BCryptGetFipsAlgorithmmode FIPS(Federal Information Processing Standard) 규정 준수를 사용할 수 있는지 여부를 결정합니다.

BCryptGetProperty CNG 개체의 명명된 속성 값을 검색합니다.
BCryptHash 단일 해시 또는 MAC 계산을 수행합니다. BCryptCreateHash, BCryptHashData, BCryptFinishHash 및 BCryptDestroyHash에 대한 호출을 래핑하는 편리한 함수입니다.
BCryptHashData 데이터 버퍼에서 단방향 해시 또는 MAC(메시지 인증 코드)를 수행합니다.
BCryptImportKey 키 BLOB에서 대칭 키를 가져옵니다.
BCryptImportKeyPair 키 BLOB에서 퍼블릭/프라이빗 키 쌍을 가져옵니다.
BCryptKeyDerivation 비밀 계약을 요구하지 않고 키를 파생합니다.
BCryptOpenAlgorithmProvider CNG 공급자를 로드하고 초기화합니다.
BCryptProcessMultiOperations BCryptProcessMultiOperations 함수는 다중 개체 상태에서 연산 시퀀스를 처리합니다.
BCryptQueryContextConfiguration 지정된 CNG 컨텍스트에 대한 현재 구성을 검색합니다.
BCryptQueryContextFunctionConfiguration 기존 CNG 컨텍스트에 대한 암호화 함수 구성 정보를 가져옵니다.
BCryptQueryContextFunctionProperty 기존 CNG 컨텍스트에서 암호화 함수의 명명된 속성 값을 가져옵니다.
BCryptQueryProviderRegistration CNG 공급자에 대한 정보를 검색합니다.

BCryptRegisterConfigChangeNotify 사용자 모드 CNG 구성 변경 이벤트 처리기를 만듭니다.
BCryptRegisterConfigChangeNotify BCryptRegisterConfigChangeNotify(PRKEVENT) 함수가 커널 모드 CNG 구성 변경 이벤트 처리기를 만드는 방법을 설명합니다.
BCryptRemoveContextFunction 기존 CNG 컨텍스트에서 지원되는 함수 목록에서 암호화 함수를 제거합니다.
BCryptResolveProviders 지정된 조건을 충족하는 모든 공급자의 컬렉션을 가져옵니다.
BCryptSecretAgreement 프라이빗 및 공개 키에서 비밀 계약 값을 만듭니다. (BCryptSecretAgreement)
BCryptSetContextFunctionProperty 기존 CNG 컨텍스트에서 암호화 함수의 명명된 속성 값을 설정합니다.
BCryptSetProperty CNG 개체의 명명된 속성 값을 설정합니다.
BCryptSignHash 해시 값의 서명을 만듭니다. (BCryptSignHash)
BCryptUnregisterConfigChangeNotify BCryptRegisterConfigChangeNotify(HANDLE*) 함수를 사용하여 만든 사용자 모드 CNG 구성 변경 이벤트 처리기를 제거합니다.
BCryptUnregisterConfigChangeNotify BCryptRegisterConfigChangeNotify(HANDLE*) 함수를 사용하여 만든 사용자 모드 CNG 구성 변경 이벤트 처리기를 제거합니다.
BCryptVerifySignature 지정된 서명이 지정된 해시와 일치하는지 확인합니다. (BCryptVerifySignature)

구조

테이블 확장

BCRYPT_ALGORITHM_IDENTIFIER 암호화 알고리즘 식별자를 포함하기 위해 BCryptEnumAlgorithms 함수와 함께 사용됩니다.
BCRYPT_AUTHENTICATED_CIPHER_MODE_INFO 인증된 암호 모드와 관련된 추가 정보를 포함하기 위해 BCryptEncrypt 및 BCryptDecrypt 함수와 함께 사용됩니다.
BCRYPT_DH_KEY_BLOB 메모리에서 Diffie-Hellman 공개 키 또는 프라이빗 키 BLOB의 헤더로 사용됩니다.
BCRYPT_DH_PARAMETER_HEADER Diffie-Hellman 키에 대한 매개 변수 헤더 정보를 포함하는 데 사용됩니다.
BCRYPT_DSA_KEY_BLOB DSA(디지털 서명 알고리즘) 공개 키 또는 메모리의 프라이빗 키 BLOB에 대한 헤더로 사용됩니다. (BCRYPT_DSA_KEY_BLOB)
BCRYPT_DSA_KEY_BLOB_V2 DSA(디지털 서명 알고리즘) 공개 키 또는 메모리의 프라이빗 키 BLOB에 대한 헤더로 사용됩니다. (BCRYPT_DSA_KEY_BLOB_V2)
BCRYPT_DSA_PARAMETER_HEADER DSA(디지털 서명 알고리즘) 키에 대한 매개 변수 헤더 정보를 포함하는 데 사용됩니다.
BCRYPT_DSA_PARAMETER_HEADER_V2 DSA(디지털 서명 알고리즘) 키에 대한 매개 변수 헤더 정보를 포함합니다.
BCRYPT_ECCKEY_BLOB 메모리의 타원 곡선 공개 키 또는 프라이빗 키 BLOB에 대한 헤더로 사용됩니다.
BCRYPT_INTERFACE_VERSION CNG 공급자의 프로그래밍 인터페이스에 대한 버전 정보를 포함합니다.

BCRYPT_KEY_BLOB 모든 CNG 키 BLOB의 기본 구조입니다.
BCRYPT_KEY_DATA_BLOB_HEADER 키 데이터 BLOB에 대한 정보를 포함하는 데 사용됩니다.
BCRYPT_KEY_LENGTHS_STRUCT 공급자가 지원하는 키 크기의 범위를 정의합니다.
BCRYPT_MULTI_HASH_OPERATION BCRYPT_MULTI_HASH_OPERATION 구조체는 다중 해시 작업에서 단일 작업을 정의합니다.
BCRYPT_MULTI_OBJECT_LENGTH_STRUCT BCRYPT_MULTI_OBJECT_LENGTH_STRUCT 구조체에는 BCryptCreateMultiHash 함수에 대한 pbHashObject 버퍼의 크기를 결정하는 정보가 포함되어 있습니다.
BCRYPT_OAEP_PADDING_INFO OAEP(최적 비대칭 암호화 패딩) 체계에 대한 옵션을 제공하는 데 사용됩니다.
BCRYPT_OID DER로 인코딩된 OID(개체 식별자)에 대한 정보를 포함합니다.
BCRYPT_OID_LIST BCRYPT_OID 구조체의 컬렉션을 포함하는 데 사용됩니다. 이 구조체를 BCRYPT_HASH_OID_LIST 속성과 함께 사용하여 DER(고유 인코딩 규칙) 인코딩을 사용하여 인코딩된 해시 개체 식별자(OID) 목록을 검색합니다.
BCRYPT_PKCS1_PADDING_INFO PKCS에 대한 옵션을 제공하는 데 사용됩니다.
BCRYPT_PROVIDER_NAME CNG 공급자의 이름을 포함합니다.
BCRYPT_PSS_PADDING_INFO PSS(Probabilistic Signature Scheme) 패딩 스키마에 대한 옵션을 제공하는 데 사용됩니다.
BCRYPT_RSAKEY_BLOB

메모리의 RSA 공개 키 또는 프라이빗 키 BLOB에 대한 헤더로 사용됩니다.
BCryptBuffer BCryptBuffer 구조가 제네릭 암호화 API: CNG(차세대) 버퍼를 나타내는 방법을 설명합니다.
BCryptBufferDesc BCryptBufferDesc 구조에 제네릭 암호화 API: CNG(차세대) 버퍼 집합이 포함된 방법을 설명합니다.
CRYPT_CONTEXT_CONFIG CNG 컨텍스트에 대한 구성 정보를 포함합니다.
CRYPT_CONTEXT_FUNCTION_CONFIG CNG 컨텍스트의 암호화 함수에 대한 구성 정보를 포함합니다.
CRYPT_CONTEXT_FUNCTION_PROVIDERS CNG 구성 컨텍스트에 대한 암호화 함수 공급자 집합을 포함합니다.
CRYPT_CONTEXT_FUNCTIONS CNG 구성 컨텍스트에 대한 암호화 함수 집합을 포함합니다.
CRYPT_CONTEXTS CNG 구성 컨텍스트 식별자 집합을 포함합니다.
CRYPT_IMAGE_REF CNG 공급자 모듈에 대한 정보를 포함합니다.
CRYPT_IMAGE_REG CNG 공급자에 대한 이미지 등록 정보를 포함합니다.
CRYPT_INTERFACE_REG CNG 공급자가 지원하는 인터페이스 형식에 대한 정보를 포함하는 데 사용됩니다.
CRYPT_PROPERTY_REF CNG 컨텍스트 속성에 대한 정보를 포함합니다.
CRYPT_PROVIDER_REF 공급자가 지원하는 암호화 인터페이스에 대한 정보를 포함합니다.

CRYPT_PROVIDER_REFS 공급자 참조의 컬렉션을 포함합니다.
CRYPT_PROVIDER_REG CNG 공급자에 대한 등록 정보를 포함하는 데 사용됩니다.
CRYPT_PROVIDERS 등록된 CNG 공급자에 대한 정보를 포함합니다.

열거형

테이블 확장

BCRYPT_HASH_OPERATION_TYPE BCRYPT_HASH_OPERATION_TYPE 열거형은 해시 작업 유형을 지정합니다.
BCRYPT_MULTI_OPERATION_TYPE BCRYPT_MULTI_OPERATION_TYPE 열거형은 BCryptProcessMultiOperations 함수에 전달되는 다중 작업의 형식을 지정합니다.
DSAFIPSVERSION_ENUM FIPS 버전 정보를 포함합니다.
HASHALGORITHM_ENUM 서명 및 해시 알고리즘을 지정합니다.

Last updated on 2025. 07. 22.

bindlink.h 헤더

아티클2024. 02. 28.

이 API를 사용하면 관리자 사용자가 바인딩 필터(미니 필터 bindflt.sys)를 통해 파일 시스템 네임스페이스를 로컬 "가상 경로"에 바인딩할 수 있습니다. 바인딩 링크는 로컬 "가상 경로"에서 로컬 또는 원격 "지원 경로"로 파일 시스템 리디렉션을 제공합니다. 주로 두 가지 종류의 시나리오를 사용하도록 설정할 수 있습니다. 첫째, 네트워크 공유를 통해 원격 파일을 로컬로 표시하여 앱 호환성을 향상시키고, 둘째, 애플리케이션이 파일을 복사하지 않고도 다른 위치의 파일이 새 위치에 표시되도록 하는 시나리오를 사용하도록 설정할 수 있습니다. 바인딩 링크는 애플리케이션에 투명하며 모든 기존 API는 이 리디렉션에 대한 지식 없이 작동합니다. 가상 경로에 대한 실제 파일 또는 디렉터리가 만들어지지 않으며 바인딩 링크는 가상 경로에 대한 지원 경로에 있는 파일 및 디렉터리의 보안 설명자와 권한을 확장합니다.

bindlink.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

CreateBindLink 이 API를 사용하면 관리자가 가상 경로와 지원 경로 간에 바인딩 링크를 만들 수 있습니다.
RemoveBindLink 이 API를 사용하면 사용자가 CreateBindLink를 호출하여 이전에 만든 링크를 제거할 수 있습니다.

열거형

테이블 확장

CREATE_BIND_LINK_FLAGS 이러한 플래그를 CreateBindLink에 전달하여 사용자의 요구에 맞게 기본 바인딩 링크 동작을 변경할 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

bits.h 헤더

아티클2023. 03. 16.

이 헤더는 Background Intelligent Transfer Service에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- BITS(Background Intelligent Transfer Service)

bits.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IBackgroundCopyCallback IBackgroundCopyCallback 인터페이스를 구현하여 작업이 완료되었거나 수정되었거나 오류가 발생했음을 알 수 있습니다. 클라이언트는 작업 상태에 대해 폴링하는 대신 이 인터페이스를 사용합니다.
IBackgroundCopyError IBackgroundCopyError 인터페이스를 사용하여 오류의 원인과 전송 프로세스를 진행할 수 있는지 여부를 확인합니다.
IBackgroundCopyFile IBackgroundCopyFile에는 작업의 일부인 파일에 대한 정보가 포함되어 있습니다. 예를 들어 IBackgroundCopyFile 메서드를 사용하여 파일의 로컬 및 원격 이름을 검색하고 진행률 정보를 전송할 수 있습니다.
IBackgroundCopyJob IBackgroundCopyJob 인터페이스를 사용하여 작업에 파일을 추가하고, 작업의 우선 순위 수준을 설정하고, 작업의 상태를 확인하고, 작업을 시작하고 중지합니다.
IBackgroundCopyManager 전송 작업을 만들고, 큐의 작업이 포함된 열거자 개체를 검색하고, 큐에서 개별 작업을 검색합니다.
IEnumBackgroundCopyFiles IEnumBackgroundCopyFiles 인터페이스를 사용하여 작업에 포함된 파일을 열거합니다. IEnumBackgroundCopyFiles 인터페이스 포인터를 얻으려면 IBackgroundCopyJob::EnumFiles 메서드를 호출합니다.

IEnumBackgroundCopyJobs IEnumBackgroundCopyJobs 인터페이스를 사용하여 전송 큐의 작업 목록을 열거합니다. IEnumBackgroundCopyJobs 인터페이스 포인터를 얻으려면 IBackgroundCopyManager::EnumJobs 메서드를 호출합니다.

구조체

BG_FILE_INFO 전송할 파일의 로컬 및 원격 이름을 제공합니다.
BG_FILE_PROGRESS 전송된 바이트 수와 같은 파일 관련 진행률 정보를 제공합니다.
BG_JOB_PROGRESS 전송된 바이트 및 파일 수와 같은 작업 관련 진행률 정보를 제공합니다.
BG_JOB_TIMES 작업 관련 타임스탬프를 제공합니다.

열거형

BG_ERROR_CONTEXT 오류가 발생한 컨텍스트를 지정하는 상수를 정의합니다.
BG_JOB_PRIORITY 작업의 우선 순위 수준을 지정하는 상수를 정의합니다.
BG_JOB_PROXY_USAGE 파일 전송에 사용할 프록시를 지정하는 상수를 정의합니다. 각 작업에 대해 서로 다른 프록시 설정을 정의할 수 있습니다.

BG_JOB_STATE 작업의 다양한 상태를 지정하는 상수를 정의합니다.
BG_JOB_TYPE 다운로드와 같은 전송 작업의 유형을 지정하는 상수를 정의합니다.

bits1_5.h 헤더

아티클2023. 03. 16.

이 헤더는 Background Intelligent Transfer Service에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- BITS(Background Intelligent Transfer Service)

bits1_5.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IBackgroundCopyJob2 업로드-회신 작업에서 회신 데이터를 검색하고, 클라이언트로 회신 데이터 전송의 진행률을 결정하고, 명령줄 실행을 요청하고, 프록시 및 원격 서버 인증 요청에 대한 자격 증명을 제공합니다.

구조체

BG_AUTH_CREDENTIALS 사용자 인증 요청에 사용할 대상(프록시 또는 서버), 인증 체계 및 사용자의 자격 증명을 식별합니다. 구조체는 IBackgroundCopyJob2::SetCredentials 메서드에 전달됩니다.
BG_AUTH_CREDENTIALS_UNION BG_AUTH_CREDENTIALS 구조에 지정된 인증 체계에 사용할 자격 증명을 식별합니다.
BG_BASIC_CREDENTIALS BG_BASIC_CREDENTIALS 구조는 인증할 사용자 이름과 암호를 식별합니다.
BG_JOB_REPLY_PROGRESS 업로드-회신 작업의 회신 부분과 관련된 진행률 정보를 제공합니다.

열거형

BG_AUTH_SCHEME 프록시 또는 서버가 사용자 인증을 요청할 때 사용할 인증 체계를 지정하는 상수를 정의합니다.
BG_AUTH_TARGET 자격 증명이 프록시 또는 서버 사용자 인증 요청에 사용되는지 여부를 지정하는 상수를 정의합니다.

bits10_1.h 헤더

아티클2023. 08. 22.

이 헤더는 Background Intelligent Transfer Service에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Background Intelligent Transfer Service

bits10_1.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IBackgroundCopyCallback3 클라이언트는 IBackgroundCopyCallback3 인터페이스를 구현하여 파일 범위의 다운로드가 완료되었다는 알림을 받습니다.
IBackgroundCopyFile6 이 인터페이스를 사용하여 주문형 다운로드 작업에 대한 파일 범위를 요청합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

bits10_2.h 헤더

아티클2023. 08. 22.

이 헤더는 Background Intelligent Transfer Service에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Background Intelligent Transfer Service

bits10_2.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IBackgroundCopyJobHttpOptions2 이 인터페이스를 사용하여 및/또는 를 검색하여 BITS 전송에 사용되는 HTTP 메서드를 재정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

bits10_3.h 헤더

아티클2023. 08. 22.

이 헤더는 Background Intelligent Transfer Service에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Background Intelligent Transfer Service

bits10_3.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IBackgroundCopyJobHttpOptions3 이 인터페이스를 사용하여 HTTP 고객 헤더를 쓰기 전용으로 설정하거나 구현한 서버 인증서 유효성 검사 콜백 메서드를 설정합니다.
IBackgroundCopyServerCertificateValidationCallback 서버 인증서는 HTTPS 연결이 열릴 때 전송됩니다. 이 메서드를 사용하여 호출할 콜백을 구현하여 해당 서버 인증서의 유효성을 검사합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

bits2_0.h 헤더

아티클2023. 08. 22.

이 헤더는 Background Intelligent Transfer Service에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Background Intelligent Transfer Service

bits2_0.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IBackgroundCopyFile2 IBackgroundCopyFile2 인터페이스를 사용하여 파일에 대한 새 원격 이름을 지정하고 다운로드할 범위 목록을 검색합니다.
IBackgroundCopyJob3 IBackgroundCopyJob3 인터페이스를 사용하여 파일 범위를 다운로드하고 원격 파일 이름의 접두사를 변경합니다.

구조체

BG_FILE_RANGE 파일에서 다운로드할 바이트 범위를 식별합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

bits2_5.h 헤더

아티클2023. 08. 22.

이 헤더는 Background Intelligent Transfer Service에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Background Intelligent Transfer Service

bits2_5.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IBackgroundCopyJobHttpOptions 이 인터페이스를 사용하여 인증서 기반 클라이언트 인증에 대한 클라이언트 인증서 및 HTTP 요청에 대한 사용자 지정 헤더를 지정할 수 있습니다.

열거형

BG_CERT_STORE_LOCATION 인증서 저장소의 위치를 지정하는 상수를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

bits3_0.h 헤더

아티클2023. 03. 16.

이 헤더는 Background Intelligent Transfer Service에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- BITS(Background Intelligent Transfer Service)

bits3_0.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IBackgroundCopyCallback2 파일 다운로드가 완료되었다는 알림을 받으려면 이 인터페이스를 구현합니다.
IBackgroundCopyFile3 이 인터페이스를 사용하여 다운로드한 콘텐츠가 포함된 임시 파일의 이름을 검색하고 피어가 콘텐츠를 요청할 수 있도록 파일의 유효성을 검사합니다.
IBackgroundCopyJob4 이 인터페이스를 사용하여 피어 캐싱을 사용하도록 설정하고, 다운로드 시간을 제한하고, 사용자 토큰 특성을 검사합니다.
IBitsPeer IBitsPeer를 사용하여 이웃의 피어에 대한 정보를 가져옵니다.
IBitsPeerCacheAdministration IBitsPeerCacheAdministration을 사용하여 콘텐츠를 다운로드할 수 있는 피어 풀을 관리합니다.
IBitsPeerCacheRecord IBitsPeerCacheRecord를 사용하여 캐시의 파일에 대한 정보를 가져옵니다.
IEnumBitsPeerCacheRecords IEnumBitsPeerCacheRecords를 사용하여 캐시의 레코드를 열거합니다.
IEnumBitsPeers IEnumBitsPeers를 사용하여 BITS에서 검색한 피어 목록을 열거합니다.

bits4_0.h 헤더

아티클2023. 03. 16.

이 헤더는 Background Intelligent Transfer Service에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- BITS(Background Intelligent Transfer Service)

bits4_0.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IBackgroundCopyFile4 이 인터페이스를 사용하여 피어 및 원본 서버에 대한 다운로드 통계를 검색합니다.
IBitsTokenOptions IBitsTokenOptions를 사용하여 BITS(Background Intelligent Transfer Service) 전송 작업에 대한 보안 토큰 쌍을 연결하고 관리합니다.

bits5_0.h 헤더

아티클2023. 08. 22.

이 헤더는 Background Intelligent Transfer Service에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Background Intelligent Transfer Service

bits5_0.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IBackgroundCopyFile5 이 인터페이스를 사용하여 BITS 파일 전송의 제네릭 속성을 얻거나 설정합니다.
IBackgroundCopyJob5 이 인터페이스를 사용하여 작업의 여러 선택적 동작을 쿼리하거나 설정합니다.

구조체

BITS_FILE_PROPERTY_VALUE BITS 파일의 속성 값을 제공합니다.
BITS_JOB_PROPERTY_VALUE BITS_JOB_PROPERTY_ID 열거 값을 기반으로 BITS 작업의 속성 값을 제공합니다.

열거형

BITS_FILE_PROPERTY_ID 백그라운드 복사 파일 속성에 해당하는 ID 값을 지정하는 상수를 정의합니다.

BITS_JOB_PROPERTY_ID BITS 작업의 속성 ID를 지정하는 상수를 정의합니다.
BITS_JOB_TRANSFER_POLICY BITS 속성에 해당하는 ID 값을 지정하는 상수를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

bitscfg.h 헤더

아티클2023. 08. 22.

이 헤더는 Background Intelligent Transfer Service에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Background Intelligent Transfer Service

bitscfg.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IBITSExtensionSetup IBITSExtensionSetup 인터페이스를 사용하여 가상 디렉터리에 대한 BITS 업로드를 사용하거나 사용하지 않도록 설정합니다.
IBITSExtensionSetupFactory IBITSExtensionSetupFactory 인터페이스를 사용하여 IBITSExtensionSetup 인터페이스에 대한 포인터를 가져옵니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

bluetoothapis.h 헤더

아티클2023. 03. 16.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Bluetooth
- Bluetooth 디바이스 참조

bluetoothapis.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

BluetoothAuthenticateDevice 원격 Bluetooth 디바이스에 인증 요청을 보냅니다.
BluetoothAuthenticateDeviceEx BluetoothAuthenticateDeviceEx 함수는 원격 Bluetooth 디바이스에 인증 요청을 보냅니다.
BluetoothAuthenticateMultipleDevices 호출자가 Bluetooth 연결 마법사의 단일 인스턴스에서 여러 디바이스를 인증하라는 메시지를 표시할 수 있도록 합니다.
BluetoothDisplayDeviceProperties 디바이스 정보 속성 시트를 제어판 시작합니다.
BluetoothEnableDiscovery BluetoothEnableDiscovery 함수는 로컬 Bluetooth 라디오 또는 라디오의 검색 상태를 변경합니다.
BluetoothEnableIncomingConnections BluetoothEnableIncomingConnections 함수는 로컬 Bluetooth 라디오가 들어오는 연결을 허용하는지 여부를 수정합니다.
BluetoothEnumerateInstalledServices BluetoothEnumerateInstalledServices 함수는 Bluetooth 디바이스에서 사용하도록 설정된 서비스 GUID(Globally Unique Identifiers)를 열거합니다.

BluetoothFindDeviceClose BluetoothFindDeviceClose 함수는 디바이스 쿼리와 연결된 열거형 핸들을 닫습니다.
BluetoothFindFirstDevice BluetoothFindFirstDevice 함수는 Bluetooth 디바이스 열거를 시작합니다.
BluetoothFindFirstRadio BluetoothFindFirstRadio 함수는 로컬 Bluetooth 라디오의 열거를 시작합니다.
BluetoothFindNextDevice BluetoothFindNextDevice 함수는 다음 Bluetooth 디바이스를 찾습니다.
BluetoothFindNextRadio BluetoothFindNextRadio 함수는 다음 Bluetooth 라디오를 찾습니다.
BluetoothFindRadioClose BluetoothFindRadioClose 함수는 Bluetooth 라디오 찾기와 관련된 열거형 핸들을 닫습니다.
BluetoothGetDeviceInfo 원격 Bluetooth 디바이스에 대한 정보를 검색합니다.
BluetoothGetRadioInfo Bluetooth 라디오에 대한 정보를 가져옵니다.
BluetoothIsConnectable BluetoothIsConnectable 함수는 Bluetooth 라디오 또는 라디오를 연결할 수 있는지 여부를 결정합니다.
BluetoothIsDiscoverable BluetoothIsDiscoverable 함수는 Bluetooth 라디오 또는 라디오를 검색할 수 있는지 여부를 결정합니다.
BluetoothIsVersionAvailable BluetoothIsVersionAvailable 함수는 설치된 Bluetooth 이진 집합이 요청된 버전을 지원하는지 나타냅니다.

BluetoothRegisterForAuthentication BluetoothRegisterForAuthentication 함수는 특정 Bluetooth 디바이스에서 인증을 요청할 때 호출되는 콜백 함수를 등록합니다.
BluetoothRegisterForAuthenticationEx BluetoothRegisterForAuthenticationEx 함수는 핀 요청, 숫자 비교 및 콜백 함수에 대한 애플리케이션을 등록합니다.
BluetoothRemoveDevice Bluetooth 디바이스와 컴퓨터 간의 인증을 제거하고 디바이스에 대한 캐시된 서비스 정보를 지웁니다.
BluetoothSdpEnumAttributes BluetoothSdpEnumAttributes 함수는 SDP 레코드 스트림을 통해 열거되고 레코드의 각 특성에 대한 콜백 함수를 호출합니다.
BluetoothSdpGetAttributeValue BluetoothSdpGetAttributeValue 함수는 특성 식별자에 대한 특성 값을 검색합니다.
BluetoothSdpGetContainerElementData 컨테이너 스트림을 반복하고 컨테이너 요소 내에 포함된 각 요소를 반환합니다.
BluetoothSdpGetElementData SDP 스트림에서 단일 요소를 검색하고 구문 분석합니다.
BluetoothSdpGetString SDP 레코드에 포함된 원시 문자열을 유니코드 문자열로 변환합니다.
BluetoothSelectDevices Bluetooth 디바이스 선택을 사용하도록 설정합니다.
BluetoothSelectDevicesFree BluetoothSelectDevices에 대한 이전 호출과 연결된 리소스를 해제합니다.
BluetoothSendAuthenticationResponse BluetoothSendAuthenticationResponse 함수는 암호 응답을 보내는 인증 요청이 수신될 때 호출됩니다.

BluetoothSendAuthenticationResponseEx BluetoothSendAuthenticationResponseEx 함수는 암호 전송에 대한 인증 요청 또는 숫자 비교 응답이 수행되면 호출됩니다.
BluetoothSetLocalServiceInfo 특정 Bluetooth 라디오에 대한 로컬 서비스 정보를 설정합니다.
BluetoothSetServiceState Bluetooth 디바이스에 대한 서비스를 사용하거나 사용하지 않도록 설정합니다.
BluetoothUnregisterAuthentication BluetoothUnregisterAuthentication 함수는 이전에 BluetoothRegisterForAuthentication 함수에 대한 호출로 등록된 콜백 루틴에 대한 등록을 제거합니다.
BluetoothUpdateDeviceRecord Bluetooth 디바이스에 대한 로컬 컴퓨터 캐시를 업데이트.

콜백 함수

PFN_AUTHENTICATION_CALLBACK BluetoothRegisterForAuthentication 함수와 함께 사용됩니다.
PFN_AUTHENTICATION_CALLBACK_EX PFN_AUTHENTICATION_CALLBACK_EX 함수는 BluetoothRegisterForAuthenticationEx 함수와 함께 사용되는 콜백 함수 프로토타입입니다.
PFN_BLUETOOTH_ENUM_ATTRIBUTES_CALLBACK BluetoothSdpEnumAttributes 함수 호출에 전달된 pSDPStream 매개 변수에 있는 각 특성에 대해 한 번 호출되는 콜백 함수 프로토타입입니다.
PFN_DEVICE_CALLBACK Bluetooth 디바이스 선택과 관련하여 사용되는 콜백 프로토타입입니다.

구조체

BLUETOOTH_ADDRESS_STRUCT BLUETOOTH_ADDRESS 구조는 Bluetooth 디바이스의 주소를 제공합니다.
BLUETOOTH_AUTHENTICATE_RESPONSE BLUETOOTH_AUTHENTICATE_RESPONSE 구조에는 BTH_REMOTE_AUTHENTICATE_REQUEST 이벤트에 대한 응답으로 전달된 정보가 포함됩니다.
BLUETOOTH_AUTHENTICATION_CALLBACK_PARAMS BLUETOOTH_AUTHENTICATION_CALLBACK_PARAMS 구조에는 인증 요청에 응답하는 Bluetooth 디바이스에 대한 특정 구성 정보가 포함됩니다.
BLUETOOTH_COD_PAIRS BLUETOOTH_COD_PAIRS 구조는 COD(Bluetooth 디바이스 클래스) 정보의 사양 및 검색을 제공합니다.
BLUETOOTH_DEVICE_INFO_STRUCT BLUETOOTH_DEVICE_INFO 구조는 Bluetooth 디바이스에 대한 정보를 제공합니다.
BLUETOOTH_DEVICE_SEARCH_PARAMS Bluetooth 디바이스 검색에 대한 검색 조건을 지정합니다.
BLUETOOTH_FIND_RADIO_PARAMS BLUETOOTH_FIND_RADIO_PARAMS 구조는 설치된 Bluetooth 라디오를 열거하는 데 도움이 됩니다.
BLUETOOTH_LOCAL_SERVICE_INFO_STRUCT Bluetooth 디바이스에 대한 로컬 서비스 정보를 포함합니다.
BLUETOOTH_NUMERIC_COMPARISON_INFO BLUETOOTH_NUMERIC_COMPARISON_INFO 구조체에는 숫자 비교를 통해 인증에 사용되는 숫자 값이 포함됩니다.
BLUETOOTH_OOB_DATA_INFO BLUETOOTH_OOB_DATA_INFO 구조에는 대역 외 디바이스 페어링을 설정하기 전에 인증하는 데 사용되는 데이터가 포함됩니다.

BLUETOOTH_PASSKEY_INFO BLUETOOTH_PASSKEY_INFO 구조에는 인증에 사용되는 암호 값이 포함되어 있습니다. 암호는 암호와 비슷하지만 암호 값은 인증에 한 번만 사용됩니다.
BLUETOOTH_PIN_INFO BLUETOOTH_PIN_INFO 구조에는 PIN을 통한 인증에 사용되는 정보가 포함되어 있습니다.
BLUETOOTH_RADIO_INFO Bluetooth 라디오에 대한 정보를 포함합니다.
BLUETOOTH_SELECT_DEVICE_PARAMS Bluetooth 디바이스 및 서비스의 표시 유형, 인증 및 선택을 용이하게 하고 관리합니다.
SDP_ELEMENT_DATA SDP_ELEMENT_DATA 구조체는 SDP 요소 데이터를 저장합니다.
SDP_STRING_TYPE_DATA SDP_STRING_TYPE_DATA 구조체는 SDP 문자열 형식에 대한 정보를 저장합니다.

열거형

BLUETOOTH_AUTHENTICATION_METHOD BLUETOOTH_AUTHENTICATION_METHOD 열거형은 디바이스 페어링 중에 지원되는 인증 유형을 정의합니다.
BLUETOOTH_AUTHENTICATION_REQUIREMENTS BLUETOOTH_AUTHENTICATION_REQUIREMENTS 열거형은 인증에 필요한 '중간의 남자' 보호를 지정합니다.
BLUETOOTH_IO_CAPABILITY BLUETOOTH_IO_CAPABILITY 열거형은 Bluetooth 디바이스의 입력/출력 기능을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

bluetoothleapis.h 헤더

아티클2023. 03. 16.

이 헤더는 Bluetooth 디바이스 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Bluetooth 디바이스 참조

bluetoothleapis.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

BluetoothGATTAbortReliableWrite 신뢰할 수 있는 쓰기 프로시저의 끝을 지정하고 쓰기를 중단해야 합니다.
BluetoothGATTBeginReliableWrite BluetoothGATTBeginReliableWrite 함수는 신뢰할 수 있는 쓰기가 시작되도록 지정합니다.
BluetoothGATTEndReliableWrite 신뢰할 수 있는 쓰기의 끝을 지정하고 쓰기를 커밋해야 합니다.
BluetoothGATTGetCharacteristics 지정된 서비스에 사용할 수 있는 모든 특성을 가져옵니다.
BluetoothGATTGetCharacteristicValue 지정된 특성의 값을 가져옵니다.
BluetoothGATTGetDescriptors 지정된 특성에 사용할 수 있는 모든 설명자를 가져옵니다.
BluetoothGATTGetDescriptorValue 지정된 설명자의 값을 가져옵니다.
BluetoothGATTGetIncludedServices 지정된 서비스에 사용할 수 있는 모든 포함된 서비스를 가져옵니다.

BluetoothGATTGetServices BluetoothGATTGetServices 함수는 서버에 사용할 수 있는 모든 기본 서비스를 가져옵니다.
BluetoothGATTRegisterEvent 특성 핸들로 식별된 지정된 특성에 대한 특성 값 변경 이벤트 중에 다시 호출할 루틴을 등록합니다.
BluetoothGATTSetCharacteristicValue 지정된 특성 값을 Bluetooth 디바이스에 씁니다.
BluetoothGATTSetDescriptorValue 지정된 설명자 값을 Bluetooth 디바이스에 씁니다.
BluetoothGATTUnregisterEvent 지정된 특성 값 변경 이벤트를 등록 취소합니다.

bthdef.h 헤더

아티클2023. 03. 16.

이 헤더는 Bluetooth에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Bluetooth

bthdef.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

BTH_DEVICE_INFO Bluetooth 디바이스에 대한 정보를 저장합니다.
BTH_HCI_EVENT_INFO Bluetooth에 대한 WM_DEVICECHANGE 메시지 가져오기와 관련하여 사용됩니다.
BTH_L2CAP_EVENT_INFO L2CAP 채널과 연결된 이벤트에 대한 데이터를 포함합니다.
BTH_RADIO_IN_RANGE Bluetooth 디바이스에 대한 데이터를 통신 범위 내에 저장합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

bthledef.h 헤더

아티클2023. 08. 22.

이 헤더는 Bluetooth 디바이스 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Bluetooth 디바이스 참조

bthledef.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

IsBthLEUuidMatch 두 UUID가 서로 일치하는지 여부를 결정합니다.

콜백 함수

PFNBLUETOOTH_GATT_EVENT_CALLBACK 프로필 드라이버는 특정 특성의 값이 변경될 때마다 호출할 Bluetooth GATT 이벤트 콜백을 구현합니다.

구조체

BLUETOOTH_GATT_VALUE_CHANGED_EVENT BLUETOOTH_GATT_VALUE_CHANGED_EVENT 구조체는 변경된 특성 값을 설명합니다.
BLUETOOTH_GATT_VALUE_CHANGED_EVENT_REGISTRATION BLUETOOTH_GATT_VALUE_CHANGED_EVENT_REGISTRATION 구조체는 변경된 하나 이상의 특성을 설명합니다.
BTH_LE_GATT_CHARACTERISTIC BTH_LE_GATT_CHARACTERISTIC 구조는 BLUEtooth LE(저에너지) GATT(제네릭 특성) 프로필 특성을 설명합니다.

BTH_LE_GATT_CHARACTERISTIC_VALUE BTH_LE_GATT_CHARACTERISTIC_VALUE 구조는 BLUEtooth LE(저에너지) GATT(제네릭 특성) 프로필 특성 값을 설명합니다.
BTH_LE_GATT_DESCRIPTOR BTH_LE_GATT_DESCRIPTOR 구조는 BLUEtooth LE(저에너지) GATT(제네릭 특성) 프로필 설명자를 설명합니다.
BTH_LE_GATT_DESCRIPTOR_VALUE BTH_LE_GATT_DESCRIPTOR_VALUE 구조체는 부모 특성을 설명합니다.
BTH_LE_GATT_SERVICE BTH_LE_GATT_SERVICE 구조는 BLUEtooth LE(저에너지) GATT(제네릭 특성) 프로필 서비스를 설명합니다.
BTH_LE_UUID BTH_LE_UUID 구조에는 LE(Bluetooth Low Energy) UUID(유니버설 고유 식별자)에 대한 정보가 포함되어 있습니다.

열거형

BTH_LE_GATT_DESCRIPTOR_TYPE BTH_LE_GATT_DESCRIPTOR_TYPE 열거형은 다양한 유형의 Bluetooth LE 일반 특성(GATT)을 설명합니다.
BTH_LE_GATT_EVENT_TYPE BTH_LE_GATT_EVENT_TYPE 열거형은 다양한 유형의 LE(Bluetooth Low Energy) 제네릭 특성(GATT) 프로필 이벤트를 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

bthsdpdef.h 헤더

아티클2023. 08. 22.

이 헤더는 Bluetooth에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Bluetooth

bthsdpdef.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

SdpAttributeRange SdpAttributeRange 구조체는 Bluetooth 쿼리에서 쿼리에서 반환할 특성 집합을 제한하는 데 사용됩니다.
SdpQueryUuid SdpQueryUuid 구조는 UUID를 쉽게 검색할 수 있도록 합니다.
SdpQueryUuidUnion SdpQueryUuidUnion 공용 구조체에는 SDP 쿼리를 수행할 UUID가 포함되어 있습니다. SdpQueryUuid 구조체와 함께 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

callobj.h 헤더

아티클2023. 08. 22.

이 헤더는 COM(구성 요소 개체 모델)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)

callobj.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICallFrame 스택 프레임과 같은 호출 프레임을 조작할 수 있습니다.
ICallFrameEvents 메서드 호출 알림을 전달합니다.
ICallFrameWalker 흥미로운 값을 찾는 스택 프레임을 안내합니다.
ICallIndirect 기존의 직접 호출이 아닌 호출 인수에 대한 간접 참조를 사용하여 개체를 호출합니다.
ICallInterceptor 인터페이스에서 직접 호출을 알리려는 이벤트 싱크의 등록 및 등록 취소를 지원합니다.
ICallUnmarshal 원격 호출의 서버(수신) 쪽에서 사용됩니다.

Functions

CoGetInterceptor

지정된 인터페이스를 가로챌 적절한 인터셉터를 인스턴스화하고 새로 만든 인터셉터를 반환합니다.

구조체

CALLFRAME_MARSHALCONTEXT 마샬링을 수행해야 하는 컨텍스트에 대한 정보를 제공합니다.
CALLFRAMEINFO 호출 프레임의 메서드와 같은 호출 프레임에 대한 정보를 제공합니다(매개 변수가 in, out 또는 in/out 매개 변수, [in], [out] 또는 [in, out] 인터페이스의 수, 인터페이스 ID, 인터페이스의 메서드 수 및 이 메서드의 매개 변수 수인 경우).
CALLFRAMEPARAMINFO 스택의 매개 변수에 대한 정보를 제공합니다.

열거형

CALLFRAME_COPY 부모 프레임에 대한 수명 종속성을 확인하여 복사한 호출 프레임 데이터를 부모 프레임의 데이터와 공유할 수 있는지 여부를 결정합니다.
CALLFRAME_FREE 해제할 매개 변수 형식을 결정합니다. (CALLFRAME_FREE)
CALLFRAME_NULL 해제할 매개 변수 형식을 결정합니다. (CALLFRAME_NULL)
CALLFRAME_WALK 연습할 매개 변수 형식을 결정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

camerauicontrol.h 헤더

아티클2023. 03. 16.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보
- Directshow
- Microsoft Media Foundation
- Windows Media Format 11 SDK

camerauicontrol.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICameraUIControl 카메라 디바이스에 대한 사용자 인터페이스 컨트롤을 사용하도록 설정합니다.
ICameraUIControlEventCallback 카메라 사용자 인터페이스 컨트롤에서 이벤트를 수신하기 위한 콜백 인터페이스입니다.

capi.h 헤더

아티클2023. 03. 08.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

capi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

JAVA_TRUST 신뢰 정보를 포함합니다.

casetup.h 헤더

아티클2023. 03. 16.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

casetup.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICertificateEnrollmentPolicyServerSetup ICertificateEnrollmentPolicyServerSetup 인터페이스는 ADCS(Active Directory Certificate Services) 내의 CEP(인증서 등록 정책) 웹 서비스를 나타냅니다.
ICertificateEnrollmentServerSetup ICertificateEnrollmentServerSetup 인터페이스는 ADCS(Active Directory Certificate Services) 내의 CES(인증서 등록 웹 서비스)를 나타냅니다.
ICertSrvSetup 인증서 서비스 컴퓨터에 CA(인증 기관) 및 인증 기관 웹 등록 역할을 설치하고 제거하는 기능을 정의합니다.
ICertSrvSetupKeyInformation CA(인증 기관) 또는 MICROSOFT SCEP(단순 인증서 등록 프로토콜) 역할 설정에 사용되는 프라이빗 키 속성 집합을 정의합니다.
ICertSrvSetupKeyInformationCollection ICertSrvSetupKeyInformation 개체의 컬렉션을 채우고 열거하는 기능을 정의합니다.
IMSCEPSetup 인증서 서비스 컴퓨터에 NDES(네트워크 디바이스 등록 서비스) 역할을 설치하고 제거하는 기능을 정의합니다.

열거형

CASetupProperty ICertSrvSetup 인터페이스를 사용할 때 CA(인증 기관) 역할의 설정 및 구성을 위한 속성 형식을 지정합니다.
CEPSetupProperty ICertificateEnrollmentPolicyServerSetup 인터페이스의 GetProperty 및 SetProperty 메서드에서 검색하거나 설정할 속성 정보의 형식을 지정하는 데 사용됩니다.
CESSetupProperty ICertificateEnrollmentServerSetup 인터페이스의 GetProperty 및 SetProperty 메서드에서 검색하거나 설정할 속성 정보의 형식을 지정하는 데 사용됩니다.
MSCEPSetupProperty IMSCEPSetup을 사용하여 MICROSOFT SCEP(단순 인증서 등록 프로토콜) 역할의 설정 및 구성을 위한 속성 유형을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ccgplugins.h 헤더

아티클2023. 03. 16.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

ccgplugins.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICcgDomainAuthCredentials 개발자가 런타임에 자신의 자격 증명을 동적으로 제공하여 Active Directory를 사용하여 도메인에 가입되지 않은 컨테이너를 인증할 수 있도록 하는 클라이언트 구현 인터페이스입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

cchannel.h 헤더

아티클2023. 03. 08.

이 헤더는 원격 데스크톱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 데스크톱 서비스

cchannel.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

콜백 함수

CHANNEL_INIT_EVENT_FN 원격 데스크톱 서비스가 클라이언트 DLL에 가상 채널 이벤트를 알리기 위해 호출하는 애플리케이션 정의 콜백 함수입니다.
CHANNEL_OPEN_EVENT_FN 원격 데스크톱 서비스가 클라이언트 DLL에 특정 가상 채널에 대한 이벤트를 알리기 위해 호출하는 애플리케이션 정의 콜백 함수입니다.
VIRTUALCHANNELCLOSE 가상 채널의 클라이언트 끝을 닫습니다.
VIRTUALCHANNELENTRY 원격 데스크톱 서비스 가상 채널을 사용하는 애플리케이션의 클라이언트 쪽 DLL에 대한 애플리케이션 정의 진입점입니다.
VIRTUALCHANNELINIT 원격 데스크톱 서비스 가상 채널에 대한 클라이언트 DLL의 액세스를 초기화합니다.
VIRTUALCHANNELOPEN 가상 채널의 클라이언트 끝을 엽니다.
VIRTUALCHANNELWRITE 가상 채널의 클라이언트 끝에서 서버 끝의 파트너 애플리케이션으로 데이터를 보냅니다.

구조체

CHANNEL_ENTRY_POINTS 가상 채널에 액세스하기 위해 클라이언트 쪽 DLL에서 호출하는 함수에 대한 포인터를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

celib.h 헤더

아티클2023. 03. 16.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

celib.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

열거형

ENUM_PERIOD 시간 범위의 단위를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

certadm.h 헤더

아티클2023. 08. 22.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

certadm.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICertAdmin 제대로 권한이 부여된 클라이언트에 대한 관리 기능을 제공합니다.
ICertAdmin2 제대로 권한이 부여된 클라이언트에 대한 관리 기능을 제공합니다.
IOCSPAdmin OCSP(온라인 인증서 상태 프로토콜) 응답자 서버를 관리하는 기능을 제공합니다.
IOCSPCA구성 OCSP(온라인 인증서 상태 프로토콜) 서비스가 특정 CA(인증 기관)에 대한 인증서 상태 요청에 응답할 수 있도록 하는 정의 집합을 나타냅니다.
IOCSPCAConfigurationCollection OCSP(온라인 인증서 상태 프로토콜) 서비스가 인증서 상태 응답을 제공하도록 구성된 인증서 집합을 나타냅니다.
IOCSPProperty OCSPServiceProperties 또는 ProviderProperties의 이름-값 쌍을 나타냅니다.
IOCSPPropertyCollection OCSP(온라인 인증서 상태 프로토콜) 서비스에 대한 구성 가능한 특성 속성(이름-값 쌍) 집합을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

certbcli.h 헤더

아티클2023. 08. 22.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

certbcli.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CertSrvBackupClose CertSrvBackupOpenFile 함수에서 연 파일을 닫습니다.
CertSrvBackupEnd 인증서 서비스 백업 세션을 종료합니다.
CertSrvBackupFree 특정 인증서 서비스 백업 API에서 할당된 메모리를 해제하는 데 사용됩니다.
CertSrvBackupGetBackupLogsW 지정된 백업 컨텍스트에 대해 백업해야 하는 인증서 서비스 로그 파일 이름 목록을 검색합니다.
CertSrvBackupGetDatabaseNamesW 지정된 백업 컨텍스트에 대해 백업해야 하는 Certificate Services 데이터베이스 파일 이름 목록을 검색합니다.
CertSrvBackupGetDynamicFileListW 지정된 백업 컨텍스트에 대해 백업해야 하는 Certificate Services 동적 파일 이름 목록을 검색합니다.
CertSrvBackupOpenFileW 백업할 파일을 엽니다.
CertSrvBackupPrepareW 백업 작업을 위해 인증서 서비스 서버를 준비하는 데 사용됩니다.

CertSrvBackupRead Certificate Services 파일에서 바이트를 읽습니다.
CertSrvBackupTruncateLogs 중복 레코드를 제거하고 로그 파일에 사용되는 디스크 스토리지 공간을 줄입니다.
CertSrvIsServerOnlineW 인증서 서비스 서버가 온라인 상태인지 확인합니다. 인증서 서비스 서버가 온라인 상태가 아니면 백업 작업이 성공하지 않습니다.
CertSrvRestoreEnd 인증서 서비스 복원 세션을 종료합니다.
CertSrvRestoreGetDatabaseLocationsW 백업 및 복원 시나리오에서 모두 사용되며 백업 또는 복원되는 모든 파일에 대한 Certificate Services 데이터베이스 위치 이름 목록을 검색합니다.
CertSrvRestorePrepareW 복원 작업을 위해 인증서 서비스 instance 준비합니다.
CertSrvRestoreRegisterComplete 등록된 인증서 서비스 복원 작업을 완료합니다.
CertSrvRestoreRegisterThroughFile 인증서 서비스 복원을 등록합니다. (CertSrvRestoreRegisterThroughFile)
CertSrvRestoreRegisterW 인증서 서비스 복원을 등록합니다. (CertSrvRestoreRegisterW)
CertSrvServerControlW 프로그래밍 방식으로 인증서 서비스를 중지하는 서비스 제어 명령을 실행합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

certcli.h 헤더

아티클2023. 08. 22.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

certcli.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICertConfig ICertConfig 인터페이스는 인증서 서비스 서버에 대한 공용 구성 데이터(클라이언트 설정 중에 지정됨)를 검색하는 기능을 제공합니다.
ICertConfig2 인증서 서비스 서버에 대한 공용 구성 데이터(클라이언트 설정 중에 지정됨)를 검색하는 기능을 제공합니다.
ICertGetConfig 인증서 서비스 서버에 대한 공용 구성 데이터(클라이언트 설정 중에 지정됨)를 검색하는 기능을 제공합니다.
ICertRequest 클라이언트 또는 중간 애플리케이션과 인증서 서비스 간의 통신을 제공합니다.
ICertRequest2 클라이언트 또는 중간 애플리케이션과 Certificate Services 간의 통신을 제공합니다. (ICertRequest2)
ICertRequest3 클라이언트 또는 중간 애플리케이션과 Certificate Services 간의 통신을 제공합니다. (ICertRequest3)

열거형

X509EnrollmentAuthFlags 인증 유형을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

certenc.h 헤더

아티클2023. 03. 16.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

certenc.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICertEncodeAltName 인증서 확장에 사용되는 대체 이름을 처리하는 메서드를 제공합니다.
ICertEncodeBitString 인증서 확장에 사용되는 비트 문자열을 처리하는 메서드를 제공합니다.
ICertEncodeCRLDistInfo 인증서 확장에 사용되는 CRL(인증서 해지 목록) 배포 정보 배열을 처리하는 메서드를 제공합니다.
ICertEncodeDateArray 인증서 확장에 사용되는 날짜 배열을 처리하는 메서드를 제공합니다.
ICertEncodeLongArray 인증서 확장에 사용되는 Long 배열을 처리하는 메서드를 제공합니다.
ICertEncodeStringArray 인증서 확장에 사용되는 문자열 배열을 처리하는 메서드를 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

certenroll.h 헤더

아티클2023. 03. 16.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 모바일 장치 관리 설정 공급자
- 보안 및 ID

certenroll.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAlternativeName IX509ExtensionAlternativeNames 개체에서 AlternativeNames 확장의 인스턴스를 나타내는 데 사용됩니다.
IAlternativeNames IAlternativeName 개체의 컬렉션을 관리할 수 있는 메서드와 속성을 포함합니다.
IBinaryConverter 바이트 배열에서 유니코드로 인코딩된 문자열을 만들고, 유니코드로 인코딩된 문자열에서 바이트 배열을 만들고, 문자열에 적용된 유니코드 인코딩 형식을 수정할 수 있는 일반적인 메서드가 포함되어 있습니다.
ICertificateAttestationChallenge 애플리케이션에서 서버에서 받은 키 증명 챌린지의 암호를 해독할 수 있습니다.
ICertificatePolicies ICertificatePolicy 개체의 컬렉션을 관리할 수 있는 메서드와 속성을 포함합니다.
ICertificatePolicy 인증서를 사용할 수 있는 목적을 식별하는 인증서 정책을 지정하는 데 사용할 수 있습니다.
ICertificationAuthorities ICertificationAuthorities 인터페이스는 ICertificationAuthority 개체의 컬렉션을 관리하는 다음 메서드 및 속성을 정의합니다.

ICertificationAuthority ICertificationAuthority 인터페이스는 단일 인증 기관을 나타냅니다. 인증 기관의 컬렉션은 ICertificationAuthorities 인터페이스로 표시됩니다.
ICertProperties 인증서 속성 컬렉션을 관리할 수 있는 메서드 및 속성을 포함합니다.
ICertProperty 외부 속성을 인증서와 연결하는 데 사용할 수 있습니다.
ICertPropertyArchived 인증서가 보관되었는지 여부를 식별하는 인증서 속성을 나타냅니다.
ICertPropertyArchivedKeyHash 보관을 위해 인증 기관에 제출된 암호화된 프라이빗 키의 SHA-1 해시를 나타냅니다.
ICertPropertyAutoEnroll 인증서 자동 등록을 사용하도록 구성된 템플릿을 식별하는 인증서 속성을 나타냅니다.
ICertPropertyBackedUp 인증서가 백업되었는지 여부와 인증서가 저장된 날짜 및 시간을 식별하는 외부 인증서 속성을 나타냅니다.
ICertPropertyDescription 인증서에 대한 설명 정보가 포함된 문자열을 지정하고 검색할 수 있습니다.
ICertPropertyEnrollment 클라이언트가 IX509Enrollment 인터페이스에서 등록 메서드를 호출할 때 생성된 인증서 및 CA(인증 기관) 정보를 포함하는 인증서 속성을 나타냅니다.
ICertPropertyEnrollmentPolicyServer CEP(인증서 등록 정책) 서버 및 CES(인증서 등록 서버)에 대한 정보를 포함하는 외부 인증서 속성을 나타냅니다.
ICertPropertyFriendlyName 인증서의 표시 이름을 포함하는 문자열을 지정하고 검색할 수 있습니다.

ICertPropertyKeyProvInfo 프라이빗 키에 대한 정보를 포함하는 인증서 속성을 나타냅니다.
ICertPropertyRenewal 기존 인증서를 갱신할 때 만든 새 인증서의 SHA-1 해시를 포함하는 인증서 속성을 나타냅니다.
ICertPropertyRequestOriginator 요청이 만들어진 컴퓨터의 DNS(Domain Naming System) 이름을 포함하는 인증서 속성을 나타냅니다.
ICertPropertySHA1Hash 인증서의 SHA-1 해시를 포함하는 인증서 속성을 나타냅니다.
ICryptAttribute ICryptAttribute 인터페이스는 인증서 요청의 암호화 특성을 나타냅니다. 이러한 특성의 컬렉션은 PKCS의 CertificateRequestInfo 구조에 포함됩니다.
ICryptAttributes ICryptAttributes 인터페이스에는 ICryptAttribute 개체의 컬렉션을 관리할 수 있는 메서드 및 속성이 포함되어 있습니다.
ICspAlgorithm 암호화 공급자가 구현하는 알고리즘을 나타냅니다.
ICspAlgorithms ICspAlgorithms 인터페이스는 ICspAlgorithm 개체의 컬렉션을 관리하는 다음 메서드 및 속성을 정의합니다.
ICspInformation 암호화 공급자에 대한 일반 정보에 대한 액세스를 제공합니다.
ICspInformations ICspInformations 인터페이스는 다음 메서드 및 속성을 정의하여 ICspInformation 개체의 컬렉션을 관리합니다.
ICspStatus 암호화 공급자/알고리즘 쌍에 대한 정보를 포함합니다. (ICspStatus)

ICspStatuses 암호화 공급자/알고리즘 쌍에 대한 정보를 포함합니다. (ICspStatuses)
IObjectId OID(개체 식별자)를 나타냅니다.
IObjectIds IObjectIds 인터페이스는 IObjectId 개체의 컬렉션을 관리할 수 있는 메서드 및 속성을 정의합니다.
IPolicyQualifier 인증서 정책과 연결할 수 있는 한정자를 나타냅니다.
IPolicyQualifiers IPolicyQualifier 개체의 컬렉션을 관리할 수 있는 메서드 및 속성을 정의합니다.
ISignerCertificate 인증서 요청에 서명할 수 있는 서명 인증서를 나타냅니다.
ISignerCertificates ISignerCertificates 인터페이스는 다음 메서드 및 속성을 정의하여 ISignerCertificate 개체의 컬렉션을 관리합니다.
ISmimeCapabilities 다음 메서드 및 속성을 정의하여 ISmimeCapability 개체의 컬렉션을 관리합니다.
ISmimeCapability 전자 메일 받는 사람의 암호 해독 기능을 식별하는 SMIMECapabilities 확장을 나타냅니다.
IX500DistinguishedName X.500 DN(고유 이름)을 나타냅니다.
IX509Attribute PKCS의 특성을 나타내는 데 사용할 수 있습니다.
IX509AttributeArchiveKey 인증 기관에서 보관할 암호화된 프라이빗 키가 포함된 특성을 나타냅니다.

IX509AttributeArchiveKeyHash 인증 기관에서 보관할 암호화된 프라이빗 키의 SHA-1 해시를 포함하는 특성을 나타냅니다.
IX509AttributeClientId 인증서 요청을 생성한 클라이언트를 식별하는 데 사용할 수 있는 특성을 나타냅니다.
IX509AttributeCspProvider 인증서를 요청하는 엔터티에서 사용하는 암호화 공급자를 식별하는 특성을 나타냅니다.
IX509AttributeExtensions 인증서 요청에서 인증서 확장을 초기화하고 검색하는 메서드 및 속성을 정의합니다.
IX509AttributeOSVersion 인증서 요청이 생성된 클라이언트 운영 체제에 대한 버전 정보를 포함하는 특성을 나타냅니다.
IX509AttributeRenewalCertificate 갱신되는 인증서가 포함된 특성을 나타냅니다. 이 특성은 PKCS에 자동으로 배치됩니다.
IX509Attributes IX509Attributes 인터페이스는 IX509Attribute 개체의 컬렉션을 관리할 수 있는 다음 메서드와 속성을 정의합니다.
IX509CertificateRequest IX509CertificateRequest 인터페이스는 인증서 등록 API에서 구현하는 각 요청 개체에 공통되고 상속되는 메서드 및 속성을 식별하는 추상 기본 인증서 요청을 나타냅니다.
IX509CertificateRequestCertificate IX509CertificateRequestCertificate 인터페이스는 자체 생성된 인증서에 대한 요청 개체를 나타내며, 등록 또는 인증 기관을 거치지 않고 직접 인증서를 만들 수 있습니다.
IX509CertificateRequestCertificate2 IX509CertificateRequestCertificate2 인터페이스는 자체 생성 인증서에 대한 요청 개체를 나타내며, 등록 또는 인증 기관을 거치지 않고 직접 인증서를 만들 수 있습니다.
IX509CertificateRequestCmc CMC(CMS를 통해 인증서 관리 메시지) 인증서 요청을 나타냅니다.

IX509CertificateRequestCmc2 IX509CertificateRequestCmc2 인터페이스는 CMC(CMS를 통해 인증서 관리 메시지) 인증서 요청을 나타냅니다.
IX509CertificateRequestPkcs10 IX509CertificateRequestPkcs10 인터페이스는 PKCS를 나타냅니다.
IX509CertificateRequestPkcs10V2 IX509CertificateRequestPkcs10V2 인터페이스는 PKCS를 나타냅니다.
IX509CertificateRequestPkcs10V3 IX509CertificateRequestPkcs10V3 인터페이스는 PKCS를 나타냅니다.
IX509CertificateRequestPkcs7 IX509CertificateRequestPkcs7 인터페이스는 PKCS를 나타냅니다.
IX509CertificateRequestPkcs7V2 IX509CertificateRequestPkcs7V2 인터페이스는 PKCS를 나타냅니다.
IX509CertificateTemplate IX509CertificateTemplate 인터페이스는 인증서 요청 템플릿을 나타냅니다. IX509CertificateTemplateWritable 인터페이스를 초기화하는 데 사용할 수 있습니다.
IX509CertificateTemplates IX509CertificateTemplates 인터페이스는 IX509CertificateTemplate 개체의 컬렉션을 관리하는 다음 메서드 및 속성을 정의합니다.
IX509CertificateTemplateWritable IX509CertificateTemplateWritable 인터페이스를 사용하면 템플릿 저장소에 템플릿을 추가하거나 삭제할 수 있습니다. 현재 Active Directory는 사용 가능한 유일한 저장소입니다.
IX509EndorsementKey X.509 인증 키 인터페이스
IX509Enrollment 최상위 개체를 나타내며 인증서 계층에 등록하고 인증서 응답을 설치할 수 있습니다.

IX509Enrollment2 IX509Enrollment2 인터페이스를 사용하면 인증서 계층 구조에 등록하고 인증서 응답을 설치할 수 있습니다.
IX509EnrollmentHelper IX509EnrollmentHelper 인터페이스는 웹 애플리케이션이 인증서를 등록하고, 자격 증명 캐시에 정책 서버 자격 증명을 저장하고, 정책 서버 및 등록 서버를 등록할 수 있도록 하는 메서드를 정의합니다.
IX509EnrollmentPolicyServer IX509EnrollmentPolicyServer 인터페이스는 CEP(인증서 등록 정책) 서버를 나타냅니다.
IX509EnrollmentStatus IX509EnrollmentStatus 인터페이스를 사용하여 인증서 등록 트랜잭션에 대한 자세한 오류 정보를 지정하거나 검색할 수 있습니다.
IX509EnrollmentWebClassFactory 웹 페이지에서 다음 개체를 만드는 데 사용할 수 있습니다.
IX509Extension 인증서 요청에 대한 확장을 정의하는 데 사용할 수 있습니다.
IX509ExtensionAlternativeNames 인증서의 주체에 대해 하나 이상의 대체 이름 양식을 지정할 수 있습니다. 인증 기관은 인증된 공개 키에 이름을 바인딩하여 확장을 처리합니다.
IX509ExtensionAuthorityKeyIdentifier AuthorityKeyIdentifier 확장을 지정할 수 있습니다.
IX509ExtensionBasicConstraints 인증서 주체가 인증 기관인지 여부를 지정할 수 있으며, 이 경우 이 확장 ID가 정의된 인증 기관 아래에 존재할 수 있는 하위 인증 기관 체인의 깊이를 지정할 수 있습니다.
IX509ExtensionCertificatePolicies 각각 OID(개체 식별자) 및 선택적 정책 한정자로 구성된 정책 정보 용어 컬렉션을 지정할 수 있습니다. 단일 정책 용어는 ICertificatePolicy 개체에 의해 정의됩니다.

IX509ExtensionEnhancedKeyUsage 인증서에 포함된 공개 키의 의도된 용도를 식별하는 OID(개체 식별자) 컬렉션을 정의하는 데 사용할 수 있습니다.
IX509ExtensionKeyUsage 인증서에 포함된 공개 키로 수행할 수 있는 작업에 대한 제한을 정의하는 데 사용할 수 있습니다.
IX509ExtensionMSApplicationPolicies 애플리케이션에서 인증서를 사용하는 방법을 나타내는 OID(개체 식별자) 컬렉션을 지정할 수 있습니다.
IX509Extensions IX509Extensions 인터페이스는 다음 메서드와 속성을 정의하여 IX509Extension 개체의 컬렉션을 관리합니다.
IX509ExtensionSmimeCapabilities 보낸 사람이 두 당사자가 지원하는 가장 안전한 알고리즘을 선택할 수 있도록 전자 메일 받는 사람의 암호 해독 기능을 전자 메일 보낸 사람에게 보고하는 데 사용할 수 있습니다.
IX509ExtensionSubjectKeyIdentifier SubjectKeyIdentifier 확장을 지정할 수 있습니다.
IX509ExtensionTemplate CertificateTemplate 확장을 초기화하거나 검색하는 데 사용할 수 있는 메서드 및 속성을 정의합니다.
IX509ExtensionTemplateName 템플릿 이름 확장을 초기화하거나 검색하는 데 사용할 수 있는 메서드 및 속성을 정의합니다.
IX509MachineEnrollmentFactory 웹 페이지에서 IX509EnrollmentHelper 개체를 만드는 데 사용할 수 있습니다.
IX509NameValuePair 제네릭 이름-값 쌍을 나타냅니다.
IX509NameValuePairs IX509NameValuePairs 인터페이스는 IX509NameValuePair 개체의 컬렉션을 관리하는 다음 메서드와 속성을 정의합니다.

IX509PolicyServerListManager IX509PolicyServerListManager 인터페이스는 IX509PolicyServerUrl 개체의 컬렉션을 관리할 수 있는 다음 메서드와 속성을 정의합니다.
IX509PolicyServerUrl IX509PolicyServerUrl 인터페이스를 사용하여 CEP(인증서 등록 정책) 서버와 연결된 속성 값을 설정하거나 검색하고 연결된 레지스트리 값을 업데이트할 수 있습니다.
IX509PrivateKey 암호화, 서명 및 키 계약에 사용할 수 있는 비대칭 프라이빗 키를 나타냅니다.
IX509PublicKey 퍼블릭/프라이빗 키 쌍의 공개 키를 나타냅니다.
IX509SCEPEnrollment X.509 간단한 컴퓨터 등록 프로토콜 인터페이스
IX509SignatureInformation 인증서 요청에 서명하는 데 사용되는 정보를 나타냅니다.

콜백 함수

ImportPFXToProvider PFX 인증서를 가져옵니다.
ImportPFXToProviderFreeData PFX 인증서 컨텍스트를 해제합니다.

열거형

AlgorithmFlags 암호화 알고리즘에 대한 검색을 구체화하는 데 사용할 수 있는 플래그를 포함합니다.
AlgorithmOperationFlags 알고리즘이 수행할 수 있는 작업을 지정합니다.
AlgorithmType 암호화 공급자가 지원하는 암호화 알고리즘의 용도를 지정합니다.
AlternativeNameType IAlternativeName 개체를 초기화할 때 지정할 수 있는 대체 이름 형식을 지정합니다.
CERTENROLL_OBJECTID 인증서 등록 API에서 지원하는 미리 정의된 OID(개체 식별자)를 포함합니다.
CERTENROLL_PROPERTYID 인증서 저장소의 인증서와 연결할 수 있는 외부 속성에 대한 미리 정의된 개체 식별자를 포함합니다.
CommitTemplateFlags 템플릿 저장 및 삭제 옵션을 지정합니다.
EncodingType 표시를 위해 바이트 배열에 적용되는 인코딩 유형을 지정합니다.
EnrollmentCAProperty 인증 기관 속성 값을 지정합니다.
EnrollmentDisplayStatus 사용자 인터페이스에 등록 상태 정보를 표시할지 여부를 지정합니다.
EnrollmentEnrollStatus 인증서 요청의 등록 상태를 지정합니다.
EnrollmentPolicyFlags 그룹 정책 플래그를 지정합니다.

EnrollmentPolicyServerPropertyFlags 기본 정책 서버를 지정합니다.
EnrollmentSelectionStatus 등록 프로세스 중에 개체의 등록 상태를 모니터링할지 여부를 지정합니다.
EnrollmentTemplateProperty 지정된 템플릿에 대한 속성 값을 포함합니다.
ImportPFXFlags PFX 인증서를 가져올 때 사용할 플래그입니다.
InnerRequestLevel PKCS 내에서 인증서 요청의 포함 수준을 지정합니다.
InstallResponseRestrictionFlags 인증서 체인의 로컬 설치에 적용되는 제한을 식별하는 플래그를 포함합니다.
KeyIdentifierHashAlgorithm 인증서 요청에서 공개 키를 해시하는 데 사용되는 알고리즘을 지정합니다.
ObjectIdGroupId OID(개체 식별자)가 속한 범주 또는 그룹을 지정합니다.
ObjectIdPublicKeyFlags 공개 키 알고리즘을 서명 또는 암호화에 사용할지 여부를 지정합니다.
PFXExportOptions PFX(개인 정보 교환) 메시지를 만들 때 포함되는 인증서 체인의 양을 지정합니다.
Pkcs10AllowedSignatureTypes 인증서 요청에 서명할 때 허용되는 서명 유형을 지정합니다.
PolicyQualifierType 인증서 정책에 적용되는 한정자 유형을 지정합니다.

PolicyServerUrlFlags CEP(인증서 등록 정책) 서버 플래그를 포함합니다.
PolicyServerUrlPropertyID GetStringProperty 메서드에서 반환하거나 IX509PolicyServerUrl 인터페이스의 SetStringProperty 메서드에서 설정할 속성 값의 형식을 지정하는 값을 포함합니다.
RequestClientInfoClientId 인증서 요청을 만든 애플리케이션의 유형을 지정합니다.
WebEnrollmentFlags 웹 등록 동작을 지정합니다.
WebSecurityLevel 웹 사용 메서드 또는 속성이 스크립팅에 안전한지 여부를 지정합니다.
X500NameFlags 고유 이름 또는 RDN(상대 고유 이름)의 표시 및 인코딩 특성을 지정합니다.
X509CertificateEnrollmentContext 인증서가 의도된 최종 엔터티의 특성을 지정합니다.
X509CertificateTemplateEnrollmentFlag 등록하는 동안 서버 및 클라이언트 작업을 지정하는 값을 포함합니다.
X509CertificateTemplateGeneralFlag 템플릿 및 관련 인증서에 대한 사용 및 수정 정보를 포함합니다.
X509CertificateTemplatePrivateKeyFlag 프라이빗 키에 대한 클라이언트 작업을 지정하는 값을 포함합니다.
X509CertificateTemplateSubjectNameFlag 주체 이름과 관련된 서버 및 클라이언트 작업을 지정하는 값을 포함합니다.
X509EnrollmentPolicyExportFlags IX509EnrollmentPolicyServer 인터페이스의 Export 메서드에서 정책 서버에서 내보낼 항목을 지정하는 데 사용됩니다.

X509EnrollmentPolicyLoadOption IX509EnrollmentPolicyServer 인터페이스의 LoadPolicy 메서드에서 정책 서버에서 정책을 검색하는 방법을 지정하는 데 사용됩니다.
X509KeySpec 레거시 CSP(암호화 서비스 공급자)에 대한 키의 용도를 지정합니다.
X509KeyUsageFlags 인증서에 포함된 키의 용도를 지정합니다.
X509PrivateKeyExportFlags 프라이빗 키에 대한 내보내기 정책을 지정합니다.
X509PrivateKeyProtection 암호화 공급자가 지원하는 프라이빗 키 보호 수준을 지정합니다.
X509PrivateKeyUsageFlags 프라이빗 키의 허용된 사용을 지정합니다.
X509PrivateKeyVerify 프라이빗 키 확인 중에 사용자 인터페이스가 표시되는지 여부와 암호화 공급자가 스마트 카드 공급자인 경우 확인을 진행할 수 있는지 여부를 지정합니다.
X509ProviderType 암호화 공급자의 유형을 지정합니다.
X509RequestInheritOptions 기존 인증서에서 새 요청을 만들 때 키, 확장 값 및 외부 속성을 상속하는 방법을 지정합니다.
X509RequestType 인증서 요청 유형을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

certexit.h 헤더

아티클2023. 08. 22.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

certexit.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICertExit 인증서 서비스 서버와 종료 모듈 간의 통신을 제공합니다.
ICertExit2 인증서 서비스 서버와 종료 모듈 간의 통신을 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

certif.h 헤더

아티클2023. 08. 22.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

certif.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICertServerExit 서버 엔진에서 내보내고 종료 모듈에서 호출됩니다.
ICertServerPolicy 정책 모듈이 인증서 서비스와 통신할 수 있도록 허용합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

certmod.h 헤더

아티클2023. 03. 16.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

certmod.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICertManageModule 인증서 서비스 정책 또는 종료 모듈에 대한 정보를 검색하기 위해 제공됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

certpol.h 헤더

아티클2023. 08. 22.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

certpol.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICertPolicy 인증서 서비스 서버 엔진과 정책 모듈 간의 통신을 제공합니다.
ICertPolicy2 인증서 서비스 서버 엔진과 정책 모듈 간의 통신을 제공합니다.
INDESPolicy NDES 정책 모듈 인터페이스입니다. 엔터프라이즈 CA에 대해 설치되면 NDES는 사용자가 구성된 NDES 템플릿(사용자 및 컴퓨터 템플릿 모두)에 대한 등록 권한이 있는지 확인한 후 암호를 생성합니다.

열거형

X509SCEPDisposition 응답 메시지를 처리하기 위한 요청의 결과 처리에 대해 설명합니다.
X509SCEPFailInfo SCEP 인증서 등록 실패의 특성을 설명합니다.

피드백

이 페이지가 도움이 되었나요?

Yes No

Microsoft Q&A에서 도움말 보기

certpoleng.h 헤더

아티클2023. 08. 22.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

certpoleng.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

PstAcquirePrivateKey 호출자의 프라이빗 키를 지정된 인증서와 연결합니다.
PstGetCertificates 지정된 서버에서 사용자를 인증하는 데 사용할 수 있는 인증서를 지정하는 인증서 체인을 검색합니다.
PstGetTrustAnchors 지정된 서버에서 신뢰할 수 있는 CA(인증 기관) 목록을 검색합니다.
PstGetUserNameForCertificate 지정된 인증서와 연결된 사용자 이름을 검색합니다.
PstMapCertificate 지정된 인증서와 연결된 사용자 토큰을 만드는 데 사용할 수 있는 정보를 지정하는 구조를 검색합니다.
PstValidate 지정된 인증서의 유효성을 검사합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

certsrv.h 헤더

아티클2023. 03. 16.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

certsrv.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

열거형

ENUM_CATYPES CA(인증 기관) 유형을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

certview.h 헤더

아티클2023. 08. 22.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

certview.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICertView 제대로 권한이 부여된 클라이언트가 인증서 서비스 데이터베이스의 사용자 지정 또는 전체 보기를 만들 수 있습니다.
ICertView2 제대로 권한이 부여된 클라이언트가 인증서 서비스 데이터베이스의 사용자 지정 또는 전체 보기를 만들 수 있도록 허용합니다.
IEnumCERTVIEWATTRIBUTE 행 열거형 시퀀스의 현재 행에 대한 인증서 특성을 포함하는 특성 열거형 시퀀스를 나타냅니다.
IEnumCERTVIEWCOLUMN 열거형 시퀀스의 현재 행에 대한 열 데이터를 포함하는 열거형 시퀀스를 나타냅니다.
IEnumCERTVIEWEXTENSION 행 열거형 시퀀스의 현재 행에 대한 인증서 확장 데이터를 포함하는 확장 열거형 시퀀스를 나타냅니다.
IEnumCERTVIEWROW 인증서 서비스 보기의 행에 있는 데이터를 포함하는 행 열거형 시퀀스를 나타내며, 각 행과 연결된 열, 특성 및 확장에 대한 추가 액세스를 허용합니다.

피드백

이 페이지가 도움이 되었나요?

Yes No

Microsoft Q&A에서 도움말 보기

cfapi.h 헤더

아티클2023. 03. 16.

이 헤더는 클라우드 필터 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 클라우드 필터 API

cfapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CfCloseHandle CfOpenFileWithOplock에서 반환된 파일 또는 디렉터리 핸들을 닫습니다. CfApi.h 내에서 사용되는 핸들에서만 표준 Win32 파일 핸들과 함께 사용하면 안 됩니다.
CfConnectSyncRoot 동기화 공급자와 동기화 필터 API 간의 양방향 통신을 시작합니다.
CfConvertToPlaceholder 일반 파일/디렉터리를 자리 표시자 파일/디렉터리로 변환합니다.
CfCreatePlaceholders 동기화 루트 트리 아래에 하나 이상의 새 자리 표시자 파일 또는 디렉터리를 만듭니다.
CfDisconnectSyncRoot CfConnectSyncRoot에서 만든 통신 채널의 연결을 끊습니다.
CfExecute 모든 연결 키 기반 자리 표시자 작업의 기본 진입점입니다. 동기화 공급자가 플랫폼의 다양한 콜백에 응답하는 데 사용됩니다.
CfGetCorrelationVector 동기화 공급자가 지정된 자리 표시자 파일에 대한 현재 상관 관계 벡터를 쿼리할 수 있습니다.
CfGetPlaceholderInfo 자리 표시자 파일 또는 폴더의 다양한 특성을 가져옵니다.

CfGetPlaceholderRangeInfo 자리 표시자 파일 또는 폴더에 대한 범위 정보를 가져옵니다.
CfGetPlaceholderRangeInfoForHydration ConnectionKey, TransferKey 및 FileId를 식별자로 사용하여 자리 표시자 파일 또는 폴더에 대한 범위 정보를 가져옵니다.
CfGetPlaceholderStateFromAttributeTag 파일의 FileAttributes 및 ReparseTag 값을 기반으로 자리 표시자 상태 집합을 가져옵니다.
CfGetPlaceholderStateFromFileInfo 파일의 다양한 정보를 기반으로 자리 표시자 상태 집합을 가져옵니다.
CfGetPlaceholderStateFromFindData WIN32_FIND_DATA 구조체에 따라 자리 표시자 상태 집합을 가져옵니다.
CfGetPlatformInfo 플랫폼 버전 정보를 가져옵니다.
CfGetSyncRootInfoByHandle 파일 핸들에 지정된 지정된 파일을 포함하는 동기화 루트의 다양한 특성을 가져옵니다.
CfGetSyncRootInfoByPath 동기화 루트 아래의 파일에 지정된 다양한 동기화 루트 정보를 가져옵니다.
CfGetTransferKey 자리 표시자 파일 또는 폴더로 데이터 전송을 시작합니다.
CfGetWin32HandleFromProtectedHandle 모든 핸들 기반 Win32 API와 함께 사용할 수 있도록 보호된 핸들을 Win32 핸들로 변환합니다.
CfHydratePlaceholder 지정된 바이트 범위가 자리 표시자에 디스크에 있는지 확인하여 자리 표시자 파일을 수화합니다. 파일에만 유효합니다.

CfOpenFileWithOplock 파일 또는 디렉터리(일반 및 자리 표시자 파일 모두)에 대한 비동기 불투명 핸들을 열고 열린 플래그에 따라 적절한 oplock을 설정합니다.
CfQuerySyncProviderStatus 동기화 공급자를 쿼리하여 공급자의 상태를 가져옵니다.
CfReferenceProtectedHandle 호출자가 보호된 핸들을 비 CfApi Win32 API와 함께 사용할 수 있는 Win32 핸들에 참조할 수 있도록 허용합니다.
CfRegisterSyncRoot 일회성 동기화 루트 등록을 수행합니다.
CfReleaseProtectedHandle CfReferenceProtectedHandle에서 참조하는 보호된 핸들을 해제합니다.
CfReleaseTransferKey CfGetTransferKey에서 가져온 전송 키를 해제합니다.
CfReportProviderProgress 동기화 공급자가 대역 외 진행 상황을 보고할 수 있습니다.
CfReportProviderProgress2 동기화 공급자가 대역 외 진행 상황을 보고할 수 있습니다. 추가 매개 변수를 사용하여 CfReportProviderProgress를 확장합니다.
CfReportSyncStatus 먼저 CfConnectSyncRoot에 대한 호출과 연결하지 않고도 동기화 공급자가 지정된 동기화 루트에서 해당 상태를 플랫폼에 알릴 수 있습니다.
CfRevertPlaceholder 자리 표시자를 다시 일반 파일로 되돌리고 재분석 태그, 파일 ID 등과 같은 모든 특수 특성을 제거합니다.
CfSetCorrelationVector 동기화 공급자가 플랫폼에 자리 표시자 파일에서 원격 분석 목적으로 특정 상관 관계 벡터를 사용하도록 지시할 수 있습니다. 선택 사항입니다.

CfSetInSyncState 자리 표시자 파일 또는 폴더의 동기화 중 상태를 설정합니다.
CfSetPinState 이렇게 하면 사용자의 의도를 나타내는 데 사용되는 자리 표시자의 핀 상태가 설정됩니다. 동기화 공급자뿐만 아니라 모든 애플리케이션에서 이 함수를 호출할 수 있습니다.
CfUnregisterSyncRoot 이전에 등록된 동기화 루트의 등록을 취소합니다.
CfUpdatePlaceholder 자리 표시자 파일 또는 디렉터리의 특성을 업데이트.
CfUpdateSyncProviderStatus 동기화 공급자의 현재 상태를 업데이트.

구조체

CF_CALLBACK_INFO 일반적인 콜백 정보를 포함합니다.
CF_CALLBACK_PARAMETERS 파일 오프셋, 길이, 플래그 등과 같은 콜백 특정 매개 변수를 포함합니다.
CF_CALLBACK_REGISTRATION 동기화 공급자가 등록할 콜백입니다.
CF_FILE_RANGE 자리 표시자 파일의 데이터 범위를 지정합니다.
CF_FS_METADATA 자리 표시자 파일 또는 디렉터리 메타데이터입니다.

CF_HYDRATION_POLICY 기본 하이드레이션 정책 및 해당 한정자를 지정합니다.
CF_OPERATION_INFO 자리 표시자 파일 또는 폴더의 작업에 대한 정보입니다.
CF_OPERATION_PARAMETERS 자리 표시자 파일 또는 폴더에 대한 작업의 매개 변수입니다.
CF_PLACEHOLDER_BASIC_INFO 기본 자리 표시자 정보입니다.
CF_PLACEHOLDER_CREATE_INFO 새 자리 표시자 파일 또는 디렉터리를 만들기 위한 자리 표시자 정보를 포함합니다.
CF_PLACEHOLDER_STANDARD_INFO 표준 자리 표시자 정보입니다.
CF_PLATFORM_INFO 클라우드 파일 플랫폼에 대한 정보를 반환합니다. 이는 여러 버전의 Windows에서 실행되는 동기화 공급자를 위한 것입니다.
CF_POPULATION_POLICY 기본 채우기 정책 및 해당 한정자를 지정합니다.
CF_PROCESS_INFO 사용자 프로세스에 대한 정보를 포함합니다.
CF_SYNC_POLICIES 동기화 루트에서 사용하는 동기화 정책을 정의합니다.
CF_SYNC_REGISTRATION 등록할 동기화 공급자 및 동기화 루트의 세부 정보입니다.
CF_SYNC_ROOT_BASIC_INFO 기본 동기화 루트 정보입니다.

CF_SYNC_ROOT_PROVIDER_INFO 루트 공급자 정보를 동기화합니다.
CF_SYNC_ROOT_STANDARD_INFO 표준 동기화 루트 정보입니다.
CF_SYNC_STATUS CF_OPERATION_INFO 구조에서 지정된 동기화 루트의 상태를 설명하는 데 사용됩니다.

열거형

CF_CALLBACK_CANCEL_FLAGS 자리 표시자 파일 또는 폴더에 대한 데이터 가져오기를 취소하기 위한 콜백 플래그입니다.
CF_CALLBACK_CLOSE_COMPLETION_FLAGS 이전에 읽기/쓰기/삭제 액세스를 위해 연 동기화 루트 중 하나에 있는 자리 표시자가 이제 닫혀 있음을 동기화 공급자에 알리기 위한 콜백 플래그입니다.
CF_CALLBACK_DEHYDRATE_COMPLETION_FLAGS 동기화 루트 중 하나에 있는 자리 표시자가 성공적으로 탈수되었음을 동기화 공급자에게 알리는 콜백 플래그입니다.
CF_CALLBACK_DEHYDRATE_FLAGS 동기화 공급자에게 동기화 루트 중 하나의 자리 표시자가 탈수될 것임을 알리는 콜백 플래그입니다.
CF_CALLBACK_DEHYDRATION_REASON 클라우드 파일이 탈수된 이유를 지정합니다.
CF_CALLBACK_DELETE_COMPLETION_FLAGS 자리 표시자가 성공적으로 삭제되었음을 동기화 공급자에 알리기 위한 콜백 플래그입니다.
CF_CALLBACK_DELETE_FLAGS 이 콜백은 동기화 공급자에게 동기화 루트 중 하나에 있는 자리 표시자 파일 또는 디렉터리가 삭제되거나 삭제되지 않을 것임을 알리는 데 사용됩니다.

CF_CALLBACK_FETCH_DATA_FLAGS 자리 표시자 파일 또는 폴더에 대한 데이터를 가져오기 위한 콜백 플래그입니다.
CF_CALLBACK_FETCH_PLACEHOLDERS_FLAGS 자리 표시자 파일 또는 디렉터리의 콘텐츠에 대한 정보를 가져오기 위한 플래그입니다.
CF_CALLBACK_OPEN_COMPLETION_FLAGS 읽기/쓰기/삭제 액세스를 위해 자리 표시자가 성공적으로 열렸다는 것을 동기화 공급자에 알리기 위한 콜백 플래그입니다.
CF_CALLBACK_RENAME_COMPLETION_FLAGS 동기화 루트 중 하나에 있는 자리 표시자의 이름이 성공적으로 변경되었음을 동기화 공급자에 알리는 콜백 플래그입니다.
CF_CALLBACK_RENAME_FLAGS 플래그를 다시 호출하여 동기화 루트 중 하나에 있는 자리 표시자의 이름을 바꾸거나 이동하려고 한다고 동기화 공급자에게 알릴 수 있습니다.
CF_CALLBACK_TYPE 자리 표시자 파일 또는 폴더에 사용되는 다양한 유형의 콜백을 포함합니다.
CF_CALLBACK_VALIDATE_DATA_FLAGS 자리 표시자 파일 또는 디렉터리의 데이터의 유효성을 검사하는 플래그입니다.
CF_CONNECT_FLAGS 콜백이 호출될 때 동기화 공급자가 요청할 수 있는 추가 정보입니다.
CF_CONVERT_FLAGS 자리 표시자 파일/디렉터리 변환 플래그에 대한 일반 파일/디렉터리입니다.
CF_CREATE_FLAGS 자리 표시자 파일 또는 디렉터리를 만들기 위한 플래그입니다.
CF_DEHYDRATE_FLAGS 자리 표시자 탈수 플래그입니다.

CF_HARDLINK_POLICY 자리 표시자 파일에서 하드 링크가 허용되는지 여부를 지정합니다.
CF_HYDRATE_FLAGS 자리 표시자 데이터 하이드레이션 플래그입니다.
CF_HYDRATION_POLICY_MODIFIER 동기화 공급자가 플랫폼에서 자리 표시자 파일을 하이드레이션하는 방법을 제어할 수 있습니다. 기본 정책:_CF_HYDRATION_POLICY_PRIMARY 사용할 수 있는 한정자입니다.
CF_HYDRATION_POLICY_PRIMARY 동기화 공급자가 플랫폼에서 자리 표시자 파일을 하이드레이션하는 방법을 제어할 수 있습니다. 이것이 기본 정책입니다.
CF_IN_SYNC_STATE 자리 표시자 파일 및 폴더의 동기화 중 상태를 지정합니다.
CF_INSYNC_POLICY 플랫폼이 자리 표시자 파일 또는 디렉터리에서 동기화 중 상태를 지워야 하는 시기를 동기화 공급자가 제어할 수 있도록 하는 정책입니다.
CF_OPEN_FILE_FLAGS 파일 열기에 대한 다양한 권한을 요청하는 플래그입니다.
CF_OPERATION_ACK_DATA_FLAGS 자리 표시자 파일 또는 폴더에 대한 데이터를 확인하고 승인하는 플래그입니다.
CF_OPERATION_ACK_DEHYDRATE_FLAGS 자리 표시자 파일 또는 디렉터리의 탈수를 승인하는 플래그입니다.
CF_OPERATION_ACK_DELETE_FLAGS 자리 표시자 파일 또는 디렉터리의 삭제를 승인하는 플래그입니다.
CF_OPERATION_ACK_RENAME_FLAGS 자리 표시자 파일 또는 디렉터리 이름 바꾸기를 승인하기 위한 플래그입니다.

CF_OPERATION_RESTART_HYDRATION_FLAGS 자리 표시자 파일 또는 폴더에서 데이터 하이드레이션을 다시 시작하는 플래그입니다.
CF_OPERATION_RETRIEVE_DATA_FLAGS 자리 표시자 파일 또는 폴더에 대한 데이터를 검색하는 플래그입니다.
CF_OPERATION_TRANSFER_DATA_FLAGS 자리 표시자 파일 또는 폴더를 수화하기 위해 데이터를 전송하는 플래그입니다.
CF_OPERATION_TRANSFER_PLACEHOLDERS_FLAGS 자리 표시자 파일 또는 디렉터리를 전송할 때 동작을 지정하는 플래그입니다.
CF_OPERATION_TYPE 자리 표시자 파일 및 디렉터리에서 수행할 수 있는 작업 유형입니다.
CF_PIN_STATE 자리 표시자 파일 또는 디렉터리의 상태를 고정합니다.
CF_PLACEHOLDER_CREATE_FLAGS 자리 표시자별로 자리 표시자를 만들기 위한 플래그입니다.
CF_PLACEHOLDER_INFO_CLASS 자리 표시자 정보에 대한 정보 클래스입니다.
CF_PLACEHOLDER_MANAGEMENT_POLICY CF_SYNC_POLICIES 구조체에 대한 자리 표시자 관리 정책을 지정합니다.
CF_PLACEHOLDER_RANGE_INFO_CLASS 자리 표시자 파일 데이터 범위의 형식입니다.
CF_PLACEHOLDER_STATE 자리 표시자 파일 또는 폴더의 상태입니다.
CF_POPULATION_POLICY_MODIFIER 모집단 정책 한정자를 정의합니다. 기본 정책:_CF_POPULATION_POLICY_PRIMARY 사용할 수 있는 한정자입니다.

CF_POPULATION_POLICY_PRIMARY 동기화 공급자가 플랫폼에서 자리 표시자 디렉터리 및 파일을 만드는 방법을 제어할 수 있습니다. 기본 정책입니다.
CF_REGISTER_FLAGS 동기화 루트를 등록하고 업데이트하기 위한 플래그입니다.
CF_REVERT_FLAGS 자리 표시자 파일을 일반 파일로 되돌리기 위한 플래그입니다.
CF_SET_IN_SYNC_FLAGS 자리 표시자 파일 및 폴더에 대한 동기화 상태 플래그입니다.
CF_SET_PIN_FLAGS 자리 표시자 핀 플래그입니다.
CF_SYNC_PROVIDER_STATUS 동기화 공급자의 현재 상태입니다.
CF_SYNC_ROOT_INFO_CLASS 동기화 루트 정보의 유형입니다.
CF_UPDATE_FLAGS 자리 표시자 파일 또는 디렉터리를 업데이트하기 위한 플래그입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

cfg.h 헤더

아티클2023. 03. 08.

이 헤더는 디바이스 및 드라이버 설치 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 디바이스 및 드라이버 설치 참조

cfg.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

열거형

PNP_VETO_TYPE PnP 관리자가 작업 수행 요청을 거부하는 경우 PNP_VETO_TYPE 열거형을 사용하여 거부 이유를 식별합니다.

cfgmgr32.h 헤더

이 헤더는 디바이스 및 드라이버 설치 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 디바이스 및 드라이버 설치 참조

cfgmgr32.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

CM_Add_Empty_Log_Conf CM_Add_Empty_Log_Conf 함수는 로컬 컴퓨터에서 지정된 구성 유형 및 지정된 디바이스 인스턴스에 대해 빈 논리 구성을 만듭니다.
CM_Add_Empty_Log_Conf_Ex CM_Add_Empty_Log_Conf_Ex 함수는 로컬 또는 원격 컴퓨터에서 지정된 구성 유형 및 지정된 디바이스 인스턴스에 대해 빈 논리 구성을 만듭니다.
CM_Add_ID_ExA CM_Add_ID_Ex 함수는 디바이스 ID(아직 없는 경우)를 로컬 또는 원격 컴퓨터의 디바이스 인스턴스의 하드웨어 ID 목록 또는 호환 ID 목록에 추가합니다. (ANSI)
CM_Add_ID_ExW CM_Add_ID_Ex 함수는 디바이스 ID(아직 없는 경우)를 로컬 또는 원격 컴퓨터의 디바이스 인스턴스의 하드웨어 ID 목록 또는 호환 ID 목록에 추가합니다. (유니코드)
CM_Add_IDA CM_Add_ID 함수는 지정된 디바이스 ID(아직 없는 경우)를 디바이스 인스턴스의 하드웨어 ID 목록 또는 호환 ID 목록에 추가합니다.
CM_Add_IDW CM_Add_ID 함수는 지정된 디바이스 ID(아직 없는 경우)를 디바이스 인스턴스의 하드웨어 ID 목록 또는 호환 ID 목록에 추가합니다.
CM_Add_Range
CM_Add_Res_Des

CM_Add_Res_Des 함수는 논리 구성에 리소스 설명자를 추가합니다.
CM_Add_Res_Des_Ex CM_Add_Res_Des_Ex 함수는 논리 구성에 리소스 설명자를 추가합니다. 논리 구성은 로컬 또는 원격 컴퓨터에 있을 수 있습니다.
CM_Connect_MachineA CM_Connect_Machine 함수는 원격 컴퓨터에 대한 연결을 만듭니다. (ANSI)
CM_Connect_MachineW CM_Connect_Machine 함수는 원격 컴퓨터에 대한 연결을 만듭니다. (유니코드)
CM_Create_DevNode_ExA
CM_Create_DevNode_ExW
CM_Create_DevNodeA
CM_Create_DevNodeW
CM_Create_Range_List
CM_Delete_Class_Key CM_Delete_Class_Key 함수는 시스템에서 지정된 설치된 디바이스 클래스를 제거합니다.
CM_Delete_Class_Key_Ex
CM_Delete_Device_Interface_Key_ExA CM_Delete_Device_Interface_Key_ExA 함수는 애플리케이션 및 드라이버가 인터페이스 관련 정보를 저장하는 데 사용하는 레지스트리 하위 키를 삭제합니다.
CM_Delete_Device_Interface_Key_ExW CM_Delete_Device_Interface_Key_ExW 함수는 애플리케이션 및 드라이버가 인터페이스 관련 정보를 저장하는 데 사용하는 레지스트리 하위 키를 삭제합니다.
CM_Delete_Device_Interface_KeyA

CM_Delete_Device_Interface_Key 함수는 애플리케이션 및 드라이버가 인터페이스 관련 정보를 저장하는 데 사용하는 레지스트리 하위 키를 삭제합니다. (ANSI)
CM_Delete_Device_Interface_KeyW CM_Delete_Device_Interface_Key 함수는 애플리케이션 및 드라이버가 인터페이스 관련 정보를 저장하는 데 사용하는 레지스트리 하위 키를 삭제합니다. (유니코드)
CM_Delete_DevNode_Key CM_Delete_DevNode_Key 함수는 디바이스와 연결된 지정된 사용자 액세스 가능 레지스트리 키를 삭제합니다.
CM_Delete_DevNode_Key_Ex
CM_Delete_Range
CM_Detect_Resource_Conflict
CM_Detect_Resource_Conflict_Ex
CM_Disable_DevNode CM_Disable_DevNode 함수는 디바이스를 사용하지 않도록 설정합니다.
CM_Disable_DevNode_Ex
CM_Disconnect_Machine CM_Disconnect_Machine 함수는 원격 컴퓨터에 대한 연결을 제거합니다.
CM_Dup_Range_List
CM_Enable_DevNode CM_Enable_DevNode 함수는 디바이스를 사용하도록 설정합니다.
CM_Enable_DevNode_Ex
CM_Enumerate_Classes CM_Enumerate_Classes 함수는 반복적으로 호출될 때 각 클래스의 GUID를 제공하여 로컬 머신의 설치된 디바이스 클래스를 열거합니다.

CM_Enumerate_Classes_Ex CM_Enumerate_Classes_Ex 함수는 반복적으로 호출될 때 각 클래스의 GUID를 제공하여 로컬 또는 원격 머신의 설치된 디바이스 클래스를 열거합니다.
CM_Enumerate_Enumerators_ExA CM_Enumerate_Enumerators_Ex 함수는 각 열거자의 이름을 제공하여 로컬 또는 원격 컴퓨터의 디바이스 열거자를 열거합니다. (ANSI)
CM_Enumerate_Enumerators_ExW CM_Enumerate_Enumerators_Ex 함수는 각 열거자의 이름을 제공하여 로컬 또는 원격 컴퓨터의 디바이스 열거자를 열거합니다. (유니코드)
CM_Enumerate_EnumeratorsA CM_Enumerate_Enumerators 함수는 각 열거자의 이름을 제공하여 로컬 컴퓨터의 디바이스 열거자를 열거합니다. (ANSI)
CM_Enumerate_EnumeratorsW CM_Enumerate_Enumerators 함수는 각 열거자의 이름을 제공하여 로컬 컴퓨터의 디바이스 열거자를 열거합니다. (유니코드)
CM_Find_Range
CM_First_Range
CM_Free_Log_Conf CM_Free_Log_Conf 함수는 로컬 컴퓨터에서 논리 구성 및 연결된 모든 리소스 설명자를 제거합니다.
CM_Free_Log_Conf_Ex CM_Free_Log_Conf_Ex 함수는 로컬 또는 원격 컴퓨터에서 논리 구성 및 연결된 모든 리소스 설명자를 제거합니다.
CM_Free_Log_Conf_Handle CM_Free_Log_Conf_Handle 함수는 논리 구성 핸들을 무효화하고 연결된 메모리 할당을 해제합니다.
CM_Free_Range_List
CM_Free_Res_Des

CM_Free_Res_Des 함수는 로컬 컴퓨터의 논리적 구성에서 리소스 설명자를 제거합니다.
CM_Free_Res_Des_Ex CM_Free_Res_Des_Ex 함수는 로컬 또는 원격 컴퓨터의 논리 구성에서 리소스 설명자를 제거합니다.
CM_Free_Res_Des_Handle CM_Free_Res_Des_Handle 함수는 리소스 설명 핸들을 무효화하고 연결된 메모리 할당을 해제합니다.
CM_Free_Resource_Conflict_Handle CM_Free_Resource_Conflict_Handle 함수는 리소스 충돌 목록에 대한 핸들을 무효화하고 핸들의 연결된 메모리 할당을 해제합니다.
CM_Get_Child CM_Get_Child 함수는 로컬 컴퓨터의 디바이스 트리에서 지정된 디바이스 노드(devnode)의 첫 번째 자식 노드에 대한 디바이스 인스턴스 핸들을 검색하는 데 사용됩니다.
CM_Get_Child_Ex CM_Get_Child_Ex 함수는 로컬 또는 원격 머신의 디바이스 트리에서 지정된 디바이스 노드(devnode)의 첫 번째 자식 노드에 대한 디바이스 인스턴스 핸들을 검색하는 데 사용됩니다.
CM_Get_Class_Key_Name_ExA
CM_Get_Class_Key_Name_ExW
CM_Get_Class_Key_NameA
CM_Get_Class_Key_NameW
CM_Get_Class_Name_ExA
CM_Get_Class_Name_ExW
CM_Get_Class_NameA
CM_Get_Class_NameW

CM_Get_Class_Property_ExW CM_Get_Class_Property_ExW 함수는 디바이스 인터페이스 클래스 또는 디바이스 설정 클래스에 대해 설정된 디바이스 속성을 검색합니다.
CM_Get_Class_Property_Keys CM_Get_Class_Property_Keys 함수는 디바이스 인터페이스 클래스 또는 디바이스 설정 클래스에 대해 설정된 디바이스 속성을 나타내는 디바이스 속성 키의 배열을 검색합니다.
CM_Get_Class_Property_Keys_Ex CM_Get_Class_Property_Keys_Ex 함수는 디바이스 인터페이스 클래스 또는 디바이스 설정 클래스에 대해 설정된 디바이스 속성을 나타내는 디바이스 속성 키의 배열을 검색합니다.
CM_Get_Class_PropertyW CM_Get_Class_Property 함수는 디바이스 인터페이스 클래스 또는 디바이스 설정 클래스에 대해 설정된 디바이스 속성을 검색합니다.
CM_Get_Class_Registry_PropertyA CM_Get_Class_Registry_Property 함수는 디바이스 설정 클래스 속성을 검색합니다. (ANSI)
CM_Get_Class_Registry_PropertyW CM_Get_Class_Registry_Property 함수는 디바이스 설정 클래스 속성을 검색합니다. (유니코드)
CM_Get_Depth CM_Get_Depth 함수는 로컬 컴퓨터의 디바이스 트리 내에서 지정된 디바이스 노드(devnode)의 깊이를 가져오는 데 사용됩니다.
CM_Get_Depth_Ex CM_Get_Depth_Ex 함수는 로컬 또는 원격 머신의 디바이스 트리 내에서 지정된 디바이스 노드(devnode)의 깊이를 가져오는 데 사용됩니다.
CM_Get_Device_ID_ExA CM_Get_Device_ID_Ex 함수는 로컬 또는 원격 컴퓨터에서 지정된 디바이스 인스턴스에 대한 디바이스 인스턴스 ID를 검색합니다. (ANSI)
CM_Get_Device_ID_ExW CM_Get_Device_ID_Ex 함수는 로컬 또는 원격 컴퓨터에서 지정된 디바이스 인스턴스에 대한 디바이스 인스턴스 ID를 검색합니다. (유니코드)

CM_Get_Device_ID_List_ExA CM_Get_Device_ID_List_Ex 함수는 로컬 또는 원격 컴퓨터의 디바이스 인스턴스에 대한 디바이스 인스턴스 ID 목록을 검색합니다. (ANSI)
CM_Get_Device_ID_List_ExW CM_Get_Device_ID_List_Ex 함수는 로컬 또는 원격 컴퓨터의 디바이스 인스턴스에 대한 디바이스 인스턴스 ID 목록을 검색합니다. (유니코드)
CM_Get_Device_ID_List_Size_ExA CM_Get_Device_ID_List_Size_Ex 함수는 로컬 또는 원격 컴퓨터의 디바이스 인스턴스에 대한 디바이스 인스턴스 ID 목록을 보유하는 데 필요한 버퍼 크기를 검색합니다. (ANSI)
CM_Get_Device_ID_List_Size_ExW CM_Get_Device_ID_List_Size_Ex 함수는 로컬 또는 원격 컴퓨터의 디바이스 인스턴스에 대한 디바이스 인스턴스 ID 목록을 보유하는 데 필요한 버퍼 크기를 검색합니다. (유니코드)
CM_Get_Device_ID_List_SizeA CM_Get_Device_ID_List_Size 함수는 로컬 컴퓨터의 디바이스 인스턴스에 대한 디바이스 인스턴스 ID 목록을 보유하는 데 필요한 버퍼 크기를 검색합니다. (ANSI)
CM_Get_Device_ID_List_SizeW CM_Get_Device_ID_List_Size 함수는 로컬 컴퓨터의 디바이스 인스턴스에 대한 디바이스 인스턴스 ID 목록을 보유하는 데 필요한 버퍼 크기를 검색합니다. (유니코드)
CM_Get_Device_ID_ListA CM_Get_Device_ID_List 함수는 로컬 컴퓨터의 디바이스 인스턴스에 대한 디바이스 인스턴스 ID 목록을 검색합니다. (ANSI)
CM_Get_Device_ID_ListW CM_Get_Device_ID_List 함수는 로컬 컴퓨터의 디바이스 인스턴스에 대한 디바이스 인스턴스 ID 목록을 검색합니다. (유니코드)
CM_Get_Device_ID_Size CM_Get_Device_ID_Size 함수는 로컬 컴퓨터의 디바이스 인스턴스에 대한 디바이스 인스턴스 ID를 보유하는 데 필요한 버퍼 크기를 검색합니다.
CM_Get_Device_ID_Size_Ex CM_Get_Device_ID_Size_Ex 함수는 로컬 또는 원격 컴퓨터에서 디바이스 인스턴스에 대한 디바이스 인스턴스 ID를 보유하는 데 필요한 버퍼 크기를 검색합니다.

CM_Get_Device_IDA CM_Get_Device_ID 함수는 로컬 컴퓨터에서 지정된 디바이스 인스턴스에 대한 디바이스 인스턴스 ID를 검색합니다. (ANSI)
CM_Get_Device_IDW CM_Get_Device_ID 함수는 로컬 컴퓨터에서 지정된 디바이스 인스턴스에 대한 디바이스 인스턴스 ID를 검색합니다. (유니코드)
CM_Get_Device_Interface_Alias_ExA
CM_Get_Device_Interface_Alias_ExW
CM_Get_Device_Interface_AliasA CM_Get_Device_Interface_Alias 함수는 별칭이 있는 경우 지정된 디바이스 인터페이스 인스턴스의 별칭을 반환합니다. (ANSI)
CM_Get_Device_Interface_AliasW CM_Get_Device_Interface_Alias 함수는 별칭이 있는 경우 지정된 디바이스 인터페이스 인스턴스의 별칭을 반환합니다. (유니코드)
CM_Get_Device_Interface_List_ExA
CM_Get_Device_Interface_List_ExW
CM_Get_Device_Interface_List_Size_ExA
CM_Get_Device_Interface_List_Size_ExW
CM_Get_Device_Interface_List_SizeA CM_Get_Device_Interface_List_Size 함수는 CM_Get_Device_Interface_List 함수에 전달해야 하는 버퍼 크기를 검색합니다. (ANSI)
CM_Get_Device_Interface_List_SizeW CM_Get_Device_Interface_List_Size 함수는 CM_Get_Device_Interface_List 함수에 전달해야 하는 버퍼 크기를 검색합니다. (유니코드)
CM_Get_Device_Interface_ListA

CM_Get_Device_Interface_List 함수는 지정된 디바이스 인터페이스 클래스에 속하는 디바이스 인터페이스 인스턴스 목록을 검색합니다. (ANSI)
CM_Get_Device_Interface_ListW CM_Get_Device_Interface_List 함수는 지정된 디바이스 인터페이스 클래스에 속하는 디바이스 인터페이스 인스턴스 목록을 검색합니다. (유니코드)
CM_Get_Device_Interface_Property_ExW CM_Get_Device_Interface_Property_ExW 함수는 디바이스 인터페이스에 대해 설정된 디바이스 속성을 검색합니다.
CM_Get_Device_Interface_Property_Keys_ExW CM_Get_Device_Interface_Property_Keys_ExW 함수는 디바이스 인터페이스에 대해 설정된 디바이스 속성을 나타내는 디바이스 속성 키 배열을 검색합니다.
CM_Get_Device_Interface_Property_KeysW CM_Get_Device_Interface_Property_Keys 함수는 디바이스 인터페이스에 대해 설정된 디바이스 속성을 나타내는 디바이스 속성 키 배열을 검색합니다.
CM_Get_Device_Interface_PropertyW CM_Get_Device_Interface_Property 함수는 디바이스 인터페이스에 대해 설정된 디바이스 속성을 검색합니다.
CM_Get_DevNode_Custom_Property_ExA
CM_Get_DevNode_Custom_Property_ExW
CM_Get_DevNode_Custom_PropertyA
CM_Get_DevNode_Custom_PropertyW
CM_Get_DevNode_Property_ExW CM_Get_DevNode_Property_ExW 함수는 디바이스 인스턴스 속성을 검색합니다.
CM_Get_DevNode_Property_Keys CM_Get_DevNode_Property_Keys 함수는 디바이스 인스턴스에 대해 설정된 디바이스 속성을 나타내는 디바이스 속성 키의 배열을 검색합니다.

CM_Get_DevNode_Property_Keys_Ex CM_Get_DevNode_Property_Keys_Ex 함수는 디바이스 인스턴스에 대해 설정된 디바이스 속성을 나타내는 디바이스 속성 키의 배열을 검색합니다.
CM_Get_DevNode_PropertyW CM_Get_DevNode_Property 함수는 디바이스 인스턴스 속성을 검색합니다.
CM_Get_DevNode_Registry_Property_ExA
CM_Get_DevNode_Registry_Property_ExW
CM_Get_DevNode_Registry_PropertyA CM_Get_DevNode_Registry_Property 함수는 레지스트리에서 지정된 디바이스 속성을 검색합니다. (ANSI)
CM_Get_DevNode_Registry_PropertyW CM_Get_DevNode_Registry_Property 함수는 레지스트리에서 지정된 디바이스 속성을 검색합니다. (유니코드)
CM_Get_DevNode_Status CM_Get_DevNode_Status 함수는 로컬 컴퓨터의 디바이스 트리에 있는 디바이스 노드(devnode)에서 디바이스 인스턴스의 상태를 가져옵니다.
CM_Get_DevNode_Status_Ex CM_Get_DevNode_Status_Ex 함수는 로컬 또는 원격 머신의 디바이스 트리에 있는 디바이스 노드(devnode)에서 디바이스 인스턴스의 상태를 가져옵니다.
CM_Get_First_Log_Conf CM_Get_First_Log_Conf 함수는 로컬 컴퓨터의 지정된 디바이스 인스턴스와 연결된 지정된 구성 유형의 첫 번째 논리 구성을 가져옵니다.
CM_Get_First_Log_Conf_Ex CM_Get_First_Log_Conf_Ex 함수는 로컬 또는 원격 컴퓨터에서 지정된 디바이스 인스턴스와 연결된 첫 번째 논리 구성을 가져옵니다.
CM_Get_Global_State
CM_Get_Global_State_Ex

CM_Get_Hardware_Profile_Info_ExA
CM_Get_Hardware_Profile_Info_ExW
CM_Get_Hardware_Profile_InfoA
CM_Get_Hardware_Profile_InfoW
CM_Get_HW_Prof_Flags_ExA CM_Get_HW_Prof_Flags_Ex 함수는 원격 컴퓨터 또는 로컬 컴퓨터의 디바이스 인스턴스에 대한 하드웨어 프로필별 구성 플래그를 검색합니다. (ANSI)
CM_Get_HW_Prof_Flags_ExW CM_Get_HW_Prof_Flags_Ex 함수는 원격 컴퓨터 또는 로컬 컴퓨터의 디바이스 인스턴스에 대한 하드웨어 프로필별 구성 플래그를 검색합니다. (유니코드)
CM_Get_HW_Prof_FlagsA CM_Get_HW_Prof_Flags 함수는 로컬 컴퓨터의 디바이스 인스턴스에 대한 하드웨어 프로필별 구성 플래그를 검색합니다. (ANSI)
CM_Get_HW_Prof_FlagsW CM_Get_HW_Prof_Flags 함수는 로컬 컴퓨터의 디바이스 인스턴스에 대한 하드웨어 프로필별 구성 플래그를 검색합니다. (유니코드)
CM_Get_Log_Conf_Priority CM_Get_Log_Conf_Priority 함수는 로컬 컴퓨터에서 지정된 논리 구성의 구성 우선 순위를 가져옵니다.
CM_Get_Log_Conf_Priority_Ex CM_Get_Log_Conf_Priority_Ex 함수는 로컬 또는 원격 컴퓨터에서 지정된 논리 구성의 구성 우선 순위를 가져옵니다.
CM_Get_Next_Log_Conf CM_Get_Next_Log_Conf 함수는 로컬 컴퓨터의 특정 디바이스 인스턴스와 연결된 다음 논리 구성을 가져옵니다.
CM_Get_Next_Log_Conf_Ex

CM_Get_Next_Log_Conf_Ex 함수는 로컬 또는 원격 컴퓨터의 특정 디바이스 인스턴스와 연결된 다음 논리 구성을 가져옵니다.
CM_Get_Next_Res_Des CM_Get_Next_Res_Des 함수는 로컬 컴퓨터의 논리적 구성에 대해 지정된 리소스 유형의 다음 리소스 설명자에 대한 핸들을 가져옵니다.
CM_Get_Next_Res_Des_Ex CM_Get_Next_Res_Des_Ex 함수는 로컬 또는 원격 컴퓨터의 논리적 구성에 대해 지정된 리소스 유형의 다음 리소스 설명자에 대한 핸들을 가져옵니다.
CM_Get_Parent CM_Get_Parent 함수는 로컬 컴퓨터의 디바이스 트리에서 지정된 디바이스 노드(devnode)의 부모 노드에 대한 디바이스 인스턴스 핸들을 가져옵니다.
CM_Get_Parent_Ex CM_Get_Parent_Ex 함수는 로컬 또는 원격 머신의 디바이스 트리에서 지정된 디바이스 노드(devnode)의 부모 노드에 대한 디바이스 인스턴스 핸들을 가져옵니다.
CM_Get_Res_Des_Data CM_Get_Res_Des_Data 함수는 로컬 컴퓨터의 리소스 설명자에 저장된 정보를 검색합니다.
CM_Get_Res_Des_Data_Ex CM_Get_Res_Des_Data_Ex 함수는 로컬 또는 원격 컴퓨터의 리소스 설명자에 저장된 정보를 검색합니다.
CM_Get_Res_Des_Data_Size CM_Get_Res_Des_Data_Size 함수는 로컬 컴퓨터의 지정된 리소스 설명자에 포함된 정보를 보관하는 데 필요한 버퍼 크기를 가져옵니다.
CM_Get_Res_Des_Data_Size_Ex CM_Get_Res_Des_Data_Size_Ex 함수는 로컬 또는 원격 컴퓨터의 지정된 리소스 설명자에 포함된 정보를 보관하는 데 필요한 버퍼 크기를 가져옵니다.
CM_Get_Resource_Conflict_Count CM_Get_Resource_Conflict_Count 함수는 지정된 리소스 충돌 목록에 포함된 충돌 횟수를 가져옵니다.
CM_Get_Resource_Conflict_DetailsA CM_Get_Resource_Conflict_Details 함수는 충돌 목록의 리소스 충돌 중 하나에 대한 세부 정보를 가져옵니다. (ANSI)

CM_Get_Resource_Conflict_DetailsW CM_Get_Resource_Conflict_Details 함수는 충돌 목록의 리소스 충돌 중 하나에 대한 세부 정보를 가져옵니다. (유니코드)
CM_Get_Sibling CM_Get_Sibling 함수는 로컬 컴퓨터의 디바이스 트리에서 지정된 디바이스 노드(devnode)의 다음 형제 노드에 대한 디바이스 인스턴스 핸들을 가져옵니다.
CM_Get_Sibling_Ex CM_Get_Sibling_Ex 함수는 로컬 또는 원격 컴퓨터의 디바이스 트리에서 지정된 디바이스 노드의 다음 형제 노드에 대한 디바이스 인스턴스 핸들을 가져옵니다.
CM_Get_Version CM_Get_Version 함수는 로컬 컴퓨터에 대한 PnP(플러그 앤 플레이) Configuration Manager DLL(Cfgmgr32.dll)의 버전 4.0을 반환합니다.
CM_Get_Version_Ex CM_Get_Version_Ex 함수는 로컬 또는 원격 컴퓨터에 대한 PnP(플러그 앤 플레이) Configuration Manager DLL(Cfgmgr32.dll)의 버전 4.0을 반환합니다.
CM_Intersect_Range_List
CM_Invert_Range_List
CM_Is_Dock_Station_Present CM_Is_Dock_Station_Present 함수는 도킹 스테이션이 로컬 컴퓨터에 있는지 여부를 식별합니다.
CM_Is_Dock_Station_Present_Ex CM_Is_Dock_Station_Present_Ex 함수는 도킹 스테이션이 로컬 또는 원격 컴퓨터에 있는지 여부를 식별합니다.
CM_Is_Version_Available CM_Is_Version_Available 함수는 지정된 버전의 PnP(플러그 앤 플레이) Configuration Manager DLL(Cfgmgr32.dll)이 로컬 컴퓨터에서 지원되는지 여부를 나타냅니다.
CM_Is_Version_Available_Ex CM_Is_Version_Available_Ex 함수는 지정된 버전의 PNP(플러그 앤 플레이) Configuration Manager DLL(Cfgmgr32.dll)이 로컬 또는 원격 컴퓨터에서 지원되는지 여부를 나타냅니다.

CM_Locate_DevNode_ExA CM_Locate_DevNode_Ex 함수는 로컬 컴퓨터 또는 원격 컴퓨터에서 지정된 디바이스 인스턴스 ID와 연결된 디바이스 노드에 대한 디바이스 인스턴스 핸들을 가져옵니다. (ANSI)
CM_Locate_DevNode_ExW CM_Locate_DevNode_Ex 함수는 로컬 컴퓨터 또는 원격 컴퓨터에서 지정된 디바이스 인스턴스 ID와 연결된 디바이스 노드에 대한 디바이스 인스턴스 핸들을 가져옵니다. (유니코드)
CM_Locate_DevNodeA CM_Locate_DevNode 함수는 로컬 컴퓨터에서 지정된 디바이스 인스턴스 ID와 연결된 디바이스 노드에 대한 디바이스 인스턴스 핸들을 가져옵니다. (ANSI)
CM_Locate_DevNodeW CM_Locate_DevNode 함수는 로컬 컴퓨터에서 지정된 디바이스 인스턴스 ID와 연결된 디바이스 노드에 대한 디바이스 인스턴스 핸들을 가져옵니다. (유니코드)
CM_MapCrToWin32Err 지정된 CONFIGRET 코드를 해당하는 시스템 오류 코드로 변환합니다.
CM_Merge_Range_List
CM_Modify_Res_Des CM_Modify_Res_Des 함수는 로컬 컴퓨터에서 지정된 리소스 설명자를 수정합니다.
CM_Modify_Res_Des_Ex CM_Modify_Res_Des_Ex 함수는 로컬 또는 원격 컴퓨터에서 지정된 리소스 설명자를 수정합니다.
CM_Move_DevNode
CM_Move_DevNode_Ex
CM_Next_Range
CM_Open_Class_Key_ExA
CM_Open_Class_Key_ExW

CM_Open_Class_KeyA CM_Open_Class_Key 함수는 디바이스 설정 클래스 레지스트리 키, 디바이스 인터페이스 클래스 레지스트리 키 또는 클래스의 특정 하위 키를 엽니다. (ANSI)
CM_Open_Class_KeyW CM_Open_Class_Key 함수는 디바이스 설정 클래스 레지스트리 키, 디바이스 인터페이스 클래스 레지스트리 키 또는 클래스의 특정 하위 키를 엽니다. (유니코드)
CM_Open_Device_Interface_Key_ExA CM_Open_Device_Interface_Key_ExA 함수는 애플리케이션 및 드라이버가 디바이스 인터페이스와 관련된 정보를 저장하는 데 사용하는 레지스트리 하위 키를 엽니다.
CM_Open_Device_Interface_Key_ExW CM_Open_Device_Interface_Key_ExW 함수는 애플리케이션 및 드라이버가 디바이스 인터페이스와 관련된 정보를 저장하는 데 사용하는 레지스트리 하위 키를 엽니다.
CM_Open_Device_Interface_KeyA CM_Open_Device_Interface_Key 함수는 애플리케이션 및 드라이버가 디바이스 인터페이스와 관련된 정보를 저장하는 데 사용하는 레지스트리 하위 키를 엽니다. (ANSI)
CM_Open_Device_Interface_KeyW CM_Open_Device_Interface_Key 함수는 애플리케이션 및 드라이버가 디바이스 인터페이스와 관련된 정보를 저장하는 데 사용하는 레지스트리 하위 키를 엽니다. (유니코드)
CM_Open_DevNode_Key CM_Open_DevNode_Key 함수는 디바이스별 구성 정보에 대한 레지스트리 키를 엽니다.
CM_Open_DevNode_Key_Ex
CM_Query_And_Remove_SubTree_ExA CM_Query_And_Remove_SubTree_Ex 함수는 디바이스 인스턴스와 해당 자식을 제거할 수 있는지 여부를 확인하고, 제거하면 제거합니다. (ANSI)
CM_Query_And_Remove_SubTree_ExW CM_Query_And_Remove_SubTree_Ex 함수는 디바이스 인스턴스와 해당 자식을 제거할 수 있는지 여부를 확인하고, 제거하면 제거합니다. (유니코드)
CM_Query_And_Remove_SubTreeA

CM_Query_And_Remove_SubTree 함수는 디바이스 인스턴스와 해당 자식을 제거할 수 있는지 여부를 확인하고, 제거하면 제거합니다. (ANSI)
CM_Query_And_Remove_SubTreeW CM_Query_And_Remove_SubTree 함수는 디바이스 인스턴스와 해당 자식을 제거할 수 있는지 여부를 확인하고, 제거하면 제거합니다. (유니코드)
CM_Query_Arbitrator_Free_Data
CM_Query_Arbitrator_Free_Data_Ex
CM_Query_Arbitrator_Free_Size
CM_Query_Arbitrator_Free_Size_Ex
CM_Query_Remove_SubTree
CM_Query_Remove_SubTree_Ex
CM_Query_Resource_Conflict_List CM_Query_Resource_Conflict_List 함수는 지정된 디바이스 인스턴스의 리소스 설명과 충돌하는 리소스 요구 사항이 있는 디바이스 인스턴스를 식별합니다.
CM_Reenumerate_DevNode CM_Reenumerate_DevNode 함수는 지정된 디바이스 노드 및 모든 자식으로 식별되는 디바이스를 열거합니다.
CM_Reenumerate_DevNode_Ex CM_Reenumerate_DevNode_Ex 함수는 지정된 디바이스 노드 및 모든 자식으로 식별되는 디바이스를 열거합니다.
CM_Register_Device_Driver
CM_Register_Device_Driver_Ex
CM_Register_Device_Interface_ExA

CM_Register_Device_Interface_ExW
CM_Register_Device_InterfaceA
CM_Register_Device_InterfaceW
CM_Register_Notification CM_Register_Notification 함수는 지정된 형식의 PnP 이벤트가 발생할 때 호출할 애플리케이션 콜백 루틴을 등록합니다.
CM_Remove_SubTree
CM_Remove_SubTree_Ex
CM_Request_Device_Eject_ExA CM_Request_Device_Eject_Ex 함수는 디바이스가 이동식인 경우 안전한 제거를 위해 로컬 또는 원격 디바이스 인스턴스를 준비합니다. 디바이스를 물리적으로 배출할 수 있으면 해당 디바이스가 됩니다. (ANSI)
CM_Request_Device_Eject_ExW CM_Request_Device_Eject_Ex 함수는 디바이스가 이동식인 경우 안전한 제거를 위해 로컬 또는 원격 디바이스 인스턴스를 준비합니다. 디바이스를 물리적으로 배출할 수 있으면 해당 디바이스가 됩니다. (유니코드)
CM_Request_Device_EjectA CM_Request_Device_Eject 함수는 디바이스가 이동식인 경우 안전한 제거를 위해 로컬 디바이스 인스턴스를 준비합니다. 디바이스를 물리적으로 배출할 수 있으면 해당 디바이스가 됩니다. (ANSI)
CM_Request_Device_EjectW CM_Request_Device_Eject 함수는 디바이스가 이동식인 경우 안전한 제거를 위해 로컬 디바이스 인스턴스를 준비합니다. 디바이스를 물리적으로 배출할 수 있으면 해당 디바이스가 됩니다. (유니코드)
CM_Request_Eject_PC CM_Request_Eject_PC 함수는 로컬 도킹 스테이션에 삽입된 휴대용 PC를 배출할 것을 요청합니다.
CM_Request_Eject_PC_Ex CM_Request_Eject_PC_Ex 함수는 로컬 또는 원격 도킹 스테이션에 삽입된 휴대용 PC를 배출할 것을 요청합니다.

CM_Run_Detection
CM_Run_Detection_Ex
CM_Set_Class_Property_ExW CM_Set_Class_Property_ExW 함수는 디바이스 설정 클래스 또는 디바이스 인터페이스 클래스에 대한 클래스 속성을 설정합니다.
CM_Set_Class_PropertyW CM_Set_Class_Property 함수는 디바이스 설정 클래스 또는 디바이스 인터페이스 클래스에 대한 클래스 속성을 설정합니다.
CM_Set_Class_Registry_PropertyA CM_Set_Class_Registry_Property 함수는 디바이스 설정 클래스의 속성을 설정하거나 삭제합니다. (ANSI)
CM_Set_Class_Registry_PropertyW CM_Set_Class_Registry_Property 함수는 디바이스 설정 클래스의 속성을 설정하거나 삭제합니다. (유니코드)
CM_Set_Device_Interface_Property_ExW CM_Set_Device_Interface_Property_ExW 함수는 디바이스 인터페이스의 디바이스 속성을 설정합니다.
CM_Set_Device_Interface_PropertyW CM_Set_Device_Interface_Property 함수는 디바이스 인터페이스의 디바이스 속성을 설정합니다.
CM_Set_DevNode_Problem CM_Set_DevNode_Problem 함수는 로컬 컴퓨터에 설치된 디바이스에 대한 문제 코드를 설정합니다.
CM_Set_DevNode_Problem_Ex CM_Set_DevNode_Problem_Ex 함수는 로컬 또는 원격 컴퓨터에 설치된 디바이스에 대한 문제 코드를 설정합니다.
CM_Set_DevNode_Property_ExW CM_Set_DevNode_Property_ExW 함수는 디바이스 인스턴스 속성을 설정합니다.
CM_Set_DevNode_PropertyW CM_Set_DevNode_Property 함수는 디바이스 인스턴스 속성을 설정합니다.

CM_Set_DevNode_Registry_Property_ExA
CM_Set_DevNode_Registry_Property_ExW
CM_Set_DevNode_Registry_PropertyA CM_Set_DevNode_Registry_Property 함수는 레지스트리에서 지정된 디바이스 속성을 설정합니다. (ANSI)
CM_Set_DevNode_Registry_PropertyW CM_Set_DevNode_Registry_Property 함수는 레지스트리에서 지정된 디바이스 속성을 설정합니다. (유니코드)
CM_Set_HW_Prof
CM_Set_HW_Prof_Ex
CM_Set_HW_Prof_Flags_ExA
CM_Set_HW_Prof_Flags_ExW
CM_Set_HW_Prof_FlagsA
CM_Set_HW_Prof_FlagsW
CM_Setup_DevNode CM_Setup_DevNode 함수는 디바이스 구성에 문제가 있으므로 실행되지 않는 디바이스 인스턴스를 다시 시작합니다.
CM_Setup_DevNode_Ex
CM_Test_Range_Available
CM_Uninstall_DevNode CM_Uninstall_DevNode 함수는 디바이스 인스턴스와 연결된 모든 영구 상태를 제거합니다.
CM_Uninstall_DevNode_Ex

CM_Unregister_Device_Interface_ExA
CM_Unregister_Device_Interface_ExW
CM_Unregister_Device_InterfaceA
CM_Unregister_Device_InterfaceW
CM_Unregister_Notification 코드가 Windows 7 또는 이전 버전의 Windows를 대상으로 하는 경우 CM_Unregister_Notification 대신 UnregisterDeviceNotification을 사용합니다.
CM_WaitNoPendingInstallEvents CMP_WaitNoPendingInstallEvents(CM_WaitNoPendingInstallEvents) 함수는 PnP 관리자가 수행할 보류 중인 디바이스 설치 작업이 없을 때까지 기다립니다.

구조

테이블 확장

BUSNUMBER_DES BUSNUMBER_DES 구조는 디바이스 인스턴스의 버스 번호 사용량을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다.
BUSNUMBER_RANGE BUSNUMBER_RANGE 구조는 디바이스 인스턴스의 버스 번호 사용량을 설명하는 리소스 요구 사항 목록을 지정합니다. 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.
BUSNUMBER_RESOURCE BUSNUMBER_RESOURCE 구조는 디바이스 인스턴스의 버스 번호 사용량을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정합니다. 리소스 목록 및 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.
CM_NOTIFY_EVENT_DATA

디바이스 알림 이벤트 데이터 구조입니다.
CM_NOTIFY_FILTER 디바이스 알림 필터 구조입니다.
CONFLICT_DETAILS_A CONFLICT_DETAILS 구조체는 CM_Get_Resource_Conflict_Details 함수에 대한 매개 변수로 사용됩니다. (ANSI)
CONFLICT_DETAILS_W CONFLICT_DETAILS 구조체는 CM_Get_Resource_Conflict_Details 함수에 대한 매개 변수로 사용됩니다. (유니코드)
CS_DES CS_DES 구조는 디바이스 인스턴스에 대한 디바이스 클래스별 리소스 사용을 설명하는 리소스 목록을 지정하는 데 사용됩니다. 리소스 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.
CS_RESOURCE CS_RESOURCE 구조는 디바이스 인스턴스에 대한 디바이스 클래스별 리소스 사용을 설명하는 리소스 목록을 지정하는 데 사용됩니다. 리소스 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.
DMA_DES DMA_DES 구조는 디바이스 인스턴스에 대한 DMA(직접 메모리 액세스) 채널 사용을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다.
DMA_RANGE DMA_RANGE 구조는 디바이스 인스턴스에 대한 DMA 채널 사용을 설명하는 리소스 요구 사항 목록을 지정합니다. 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.
DMA_RESOURCE DMA_RESOURCE 구조는 디바이스 인스턴스에 대한 DMA 채널 사용량을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다.
IO_DES IO_DES 구조는 디바이스 인스턴스의 I/O 포트 사용량을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다. 리소스 목록 및 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.
IO_RANGE

IO_RANGE 구조는 디바이스 인스턴스의 I/O 포트 사용을 설명하는 리소스 요구 사항 목록을 지정합니다. 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.

IO_RESOURCE

IO_RESOURCE 구조는 디바이스 인스턴스의 I/O 포트 사용량을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다.

IRQ_DES_32

IRQ_DES 구조는 디바이스 인스턴스에 대한 IRQ 줄 사용을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다. 리소스 목록 및 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요. (32비트)

IRQ_DES_64

IRQ_DES 구조는 디바이스 인스턴스에 대한 IRQ 줄 사용을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다. 리소스 목록 및 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요. (64비트)

IRQ_RANGE

IRQ_RANGE 구조는 디바이스 인스턴스에 대한 IRQ 줄 사용을 설명하는 리소스 요구 사항 목록을 지정합니다. 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.

IRQ_RESOURCE_32

IRQ_RESOURCE 구조는 디바이스 인스턴스의 IRQ 줄 사용을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다. (32비트)

IRQ_RESOURCE_64

IRQ_RESOURCE 구조는 디바이스 인스턴스의 IRQ 줄 사용을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다. (64비트)

MEM_DES

MEM_DES 구조는 디바이스 인스턴스의 메모리 사용량을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다. 리소스 목록 및 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.

MEM_RANGE

MEM_RANGE 구조는 디바이스 인스턴스의 메모리 사용량을 설명하는 리소스 요구 사항 목록을 지정합니다. 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.

MEM_RESOURCE

MEM_RESOURCE 구조는 디바이스 인스턴스의 메모리 사용량을 설명하는 리소스 목록 또는 리소스 요구

사항 목록을 지정하는 데 사용됩니다. 리소스 목록 및 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.
MFCARD_DES MFCARD_DES 구조는 다기능 디바이스 인스턴스에서 제공하는 하드웨어 함수 중 하나에서 리소스 사용량을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다.
MFCARD_RESOURCE MFCARD_RESOURCE 구조는 다기능 디바이스 인스턴스에서 제공하는 하드웨어 함수 중 하나에서 리소스 사용량을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다.
PCCARD_DES PCCARD_DES 구조는 PC 카드 인스턴스의 리소스 사용량을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다. 리소스 목록 및 리소스 요구 사항 목록에 대한 자세한 내용은 하드웨어 리소스를 참조하세요.
PCCARD_RESOURCE PCCARD_RESOURCE 구조는 PC 카드 인스턴스의 리소스 사용량을 설명하는 리소스 목록 또는 리소스 요구 사항 목록을 지정하는 데 사용됩니다.

열거 목록

테이블 확장

CM_NOTIFY_ACTION 이 열거형은 플러그 앤 플레이 디바이스 이벤트 유형을 식별합니다.

Last updated on 2025. 10. 08.

chptrarr.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows Management Instrumentation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Management Instrumentation

chptrarr.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

CHPtrArray CHPtrArray 클래스는 WMI 공급자 프레임워크의 일부입니다. CHPtrArray는 WMI 쿼리의 공급자 처리에 사용되는 배열 포인터 관리를 위한 유틸리티 인터페이스입니다.

chstrarr.h 헤더

아티클2023. 08. 22.

이 헤더는 Windows Management Instrumentation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Management Instrumentation

chstrarr.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

CHStringArray 다음 표에는 CHStringArray 메서드 및 연산자가 나열되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

chstring.h 헤더

아티클2023. 12. 15.

이 헤더는 Windows Management Instrumentation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Management Instrumentation

chstring.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

테이블 확장

CHString 다음 표에는 CHString 메서드가 나와 있습니다.

Functions

테이블 확장

NA:chstring::operator!= chstring.operator-not-equal-to 함수는 두 문자열을 비교합니다. 이러한 연산자는 대/소문자를 구분하는 Compare 메서드를 편리하게 대체합니다.
NA:chstring::operator< chstring.operator-less-than 함수는 두 문자열을 비교합니다. 이러한 연산자는 대/소문자를 구분하는 Compare 메서드를 편리하게 대체합니다.
NA:chstring::operator<= chstring.operator-less-than-equal-to 함수는 두 문자열을 비교합니다. 이러한 연산자는 대/소문자를 구분하는 Compare 메서드를 편리하게 대체합니다.
NA:chstring::operator==

chstring.operator-equal-equal-to 함수는 두 문자열을 비교합니다. 이러한 연산자는 대/소문자를 구분하는 Compare 메서드를 편리하게 대체합니다.
NA:chstring::operator> chstring.operator-greater-than 함수는 두 문자열을 비교합니다. 이러한 연산자는 대/소문자를 구분하는 Compare 메서드를 편리하게 대체합니다.
NA:chstring::operator>= chstring.operator-greater-than-equal-to 함수는 두 문자열을 비교합니다. 이러한 연산자는 대/소문자를 구분하는 Compare 메서드를 편리하게 대체합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

cimfs.h 헤더

아티클2023. 08. 22.

CIM(복합 이미지)을 생성하고 탑재하는 기능을 제공합니다.

cimfs.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CimCloseImage CimCloseImage 함수는 이미지 핸들과 연결된 리소스를 해제합니다.
CimCloseStream CimCloseStream 함수는 스트림 핸들과 연결된 리소스를 해제합니다.
CimCommitImage CimCommitImage 함수는 이미지 핸들이 나타내는 이미지를 커밋합니다.
CimCreateAlternateStream CimCreateAlternateStream 함수는 이미지 핸들이 나타내는 이미지를 기준으로 지정된 크기의 대체 스트림을 경로에 추가합니다.
CimCreateFile CimCreateFile 함수는 이미지 핸들이 나타내는 이미지를 기준으로 경로에 지정된 메타데이터가 있는 새 파일 또는 디렉터리를 추가합니다.
CimCreateHardLink CimCreateHardLink 함수는 이미지 핸들이 나타내는 이미지를 기준으로 기존 경로에 하드 링크를 추가합니다.
CimCreateImage CimCreateImage 함수는 지정된 위치에 새 이미지를 나타내는 핸들을 만듭니다. 선택적으로 해당 위치의 기존 이미지를 기반으로 합니다.
CimDeletePath CimDeletePath 함수는 이미지 핸들이 나타내는 이미지를 기준으로 경로에서 파일, 스트림, 디렉터리 또는 하드링크를 제거합니다.

CimDismountImage CimDismountImage 함수는 volumeId로 탑재된 이미지를 볼륨 GUID로 분리합니다.
CimMountImage CimMountImage 함수는 cimPath에 지정된 위치에서 명명된 이미지를 volumeId로 지정된 볼륨 GUID가 있는 볼륨으로 탑재합니다.
CimWriteStream CimWriteStream 함수는 지정된 버퍼의 데이터를 스트림 핸들이 나타내는 스트림에 씁니다.

구조체

CIMFS_FILE_METADATA CIMFS_FILE_METADATA 구조체는 CimCreateFile에서 추가할 파일에 대한 파일 메타데이터를 지정합니다.

열거형

CIM_MOUNT_IMAGE_FLAGS CIM_MOUNT_IMAGE_FLAGS 열거형에는 CimMountImage의 동작을 수정하는 데 사용할 수 있는 플래그가 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

clfs.h 헤더

아티클2023. 08. 22.

이 헤더는 데이터 액세스 및 스토리지에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지

clfs.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

ClfsLsnEqual 동일한 스트림의 두 LSN이 같은지 여부를 결정합니다.
ClfsLsnGreater 한 LSN이 다른 LSN보다 큰지 여부를 확인합니다. 두 LSN은 동일한 스트림에서 온 것이어야 합니다.
ClfsLsnLess 한 LSN이 다른 LSN보다 작은지 여부를 확인합니다. 두 LSN은 동일한 스트림에서 온 것이어야 합니다.
ClfsLsnNull 지정된 LSN이 CLFS_LSN_NULL 가능한 가장 작은 LSN과 같은지 여부를 결정합니다.

구조체

CLFS_NODE_ID 노드 식별자를 나타냅니다.
CLS_ARCHIVE_DESCRIPTOR GetNextLogArchiveExtent 함수에서 로그 보관 익스텐트에 대한 정보를 반환하는 데 사용됩니다.
CLS_CONTAINER_INFORMATION

컨테이너에 대한 일반적인 정보를 설명합니다.
CLS_INFORMATION 로그에 대한 일반적인 정보를 설명합니다.
CLS_IO_STATISTICS GetLogIoStatistics에서 보고하는 통계를 정의합니다.
CLS_IO_STATISTICS_HEADER 로그의 I/O 성능 카운터를 정의하는 GetLogIoStatistics 함수에서 검색한 정보의 헤더입니다.
CLS_LSN 유효한 로그 주소를 나타냅니다.
CLS_SCAN_CONTEXT CLS_SCAN_CONTEXT 구조에는 ScanLogContainers에서 검사하는 컨테이너에 대한 정보가 포함되어 있습니다.
CLS_WRITE_ENTRY 로그 레코드의 일부가 될 사용자 버퍼와 해당 길이를 포함합니다.

열거형

CLFS_CONTEXT_MODE 클라이언트가 로그를 검사하는 데 사용하는 방향 및 액세스 방법을 나타내는 컨텍스트 모드 형식을 지정합니다.
CLFS_IOSTATS_CLASS CLFS에서 보고한 I/O 통계 유형을 정의하며 클라이언트가 GetLogIoStatistics를 호출할 때 사용됩니다.
CLFS_LOG_ARCHIVE_MODE 로그가 임시인지 여부를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

clfsmgmt.h 헤더

아티클2023. 08. 22.

이 헤더는 데이터 액세스 및 스토리지에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지

clfsmgmt.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

CLFS_MGMT_NOTIFICATION CLFS_MGMT_NOTIFICATION 구조는 클라이언트가 수신하는 알림에 대한 정보를 지정합니다.
CLFS_MGMT_POLICY CLFS_MGMT_POLICY 구조는 CLFS(공용 로그 파일 시스템) 관리 정책을 지정합니다. PolicyType 멤버는 정책에 사용되는 멤버를 지정합니다.

열거형

CLFS_MGMT_POLICY_TYPE CLFS_MGMT_POLICY_TYPE 열거형에는 유효한 정책 유형이 나열됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

clfsmgmtw32.h 헤더

아티클2023. 08. 22.

이 헤더는 데이터 액세스 및 스토리지에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지

clfsmgmtw32.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DeregisterManageableLogClient 로그 관리자를 사용하여 클라이언트를 등록 취소합니다.
HandleLogFull 로그를 예약하거나 로그에 추가하려는 시도가 로그 전체 오류 메시지와 함께 실패할 때 관리되는 로그 클라이언트에서 호출됩니다. 로그 관리자는 클라이언트에 대한 로그 전체 조건을 resolve 시도하고 결과가 알려지면 클라이언트에 알 수 있습니다.
InstallLogPolicy 로그에 대한 정책을 설치(설정)합니다.
LogTailAdvanceFailure LogTailAdvanceFailure 함수는 로그 클라이언트가 비상을 진행하기 위해 로그 관리의 요청을 준수할 수 없음을 나타내기 위해 호출됩니다.
QueryLogPolicy QueryLogPolicy 함수를 사용하면 지정된 로그에 대해 설치된 정책을 가져올 수 있습니다.
ReadLogNotification 로그 관리자에서 알림을 검색합니다. 알림을 사용할 수 있는 경우 로그 관리자에서 대기 중인 알림을 즉시 검색합니다. 그렇지 않으면 알림이 생성될 때까지 요청이 보류 중인 상태로 유지됩니다.
RegisterForLogWriteNotification RegisterForLogWriteNotification 함수는 로그 쓰기 알림을 사용하거나 사용하지 않도록 설정하기 위해 관리되는 로그 클라이언트에 의해 호출됩니다.

RegisterManageableLogClient 로그 관리자에 클라이언트를 등록합니다. 클라이언트는 콜백을 사용하여 알림을 받을지 또는 ReadLogNotification을 사용하여 검색을 위해 대기하는 알림을 포함할지를 지정할 수 있습니다.
RemoveLogPolicy 지정된 정책을 기본 동작으로 다시 설정합니다.
SetLogFileSizeWithPolicy 설치된 정책의 상태에 따라 로그에서 컨테이너를 추가하거나 삭제합니다.

콜백 함수

PLOG_FULL_HANDLER_CALLBACK LOG_FULL_HANDLER_CALLBACK 함수는 HandleLogFull에 대한 호출이 완료되었다는 알림을 받는 애플리케이션 정의 콜백 함수입니다.
PLOG_TAIL_ADVANCE_CALLBACK LOG_TAIL_ADVANCE_CALLBACK 함수는 로그 테일을 진행하는 애플리케이션 정의 콜백 함수입니다. 콜백은 로그 관리를 위해 등록하는 스레드에서 APC(비동기 프로시저 호출)의 컨텍스트에서 호출됩니다.
PLOG_UNPINNED_CALLBACK LOG_UNPINNED_CALLBACK 함수는 로그가 고정 해제되었다는 알림을 받는 애플리케이션 정의 콜백 함수입니다.

구조체

LOG_MANAGEMENT_CALLBACKS LOG_MANAGEMENT_CALLBACKS 구조는 클라이언트 프로그램에 정보가 필요한 콜백에 대해 CLFS(Common Log File System)에 등록하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

clfsw32.h 헤더

아티클2023. 08. 22.

이 헤더는 데이터 액세스 및 스토리지에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지

clfsw32.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AddLogContainer 호출 프로세스에 .blf 파일에 대한 쓰기 액세스 권한이 있고 컨테이너의 대상 디렉터리에 파일을 만드는 기능이 있는 경우 로그 핸들과 연결된 실제 로그에 컨테이너를 추가합니다.
AddLogContainerSet 호출 프로세스에서 로그 핸들에 액세스할 수 있는 경우 로그 핸들과 연결된 실제 로그에 여러 로그 컨테이너를 추가합니다.
AdvanceLogBase 로그 스트림의 LSN(기본 로그 시퀀스 번호)을 지정된 LSN으로 이동합니다.
AlignReservedLog 예약 레코드 집합에 대한 섹터 정렬 예약 크기를 계산합니다.
AllocReservedLog 예약된 레코드 집합에 대한 섹터 맞춤 공간을 할당합니다.
CloseAndResetLogFile 로그 파일을 다시 설정한 다음 로그를 종료합니다.
CreateLogContainerScanContext ScanLogContainers와 함께 사용할 검사 컨텍스트를 만들어 로그와 연결된 모든 로그 컨테이너를 열거하고 첫 번째 검사를 수행합니다.
CreateLogFile

로그를 만들거나 엽니다.
CreateLogMarshallingArea 로그에 대한 마샬링 영역을 만들고 성공하면 마샬링 컨텍스트를 반환합니다. 마샬링 영역을 만들기 전에 로그에 컨테이너가 하나 이상 있어야 합니다.
DeleteLogByHandle 삭제할 지정된 로그를 표시합니다. 로그에 대한 모든 핸들, 마샬링 영역 및 읽기 컨텍스트가 닫혀 있으면 로그가 실제로 삭제됩니다. 로그가 실제 로그인 경우 해당 기본 컨테이너가 삭제됩니다.
DeleteLogFile 삭제할 로그를 표시합니다. 로그에 대한 모든 핸들, 마샬링 영역 및 읽기 컨텍스트가 닫혀 있으면 로그가 실제로 삭제됩니다. 로그가 실제 로그인 경우 해당 기본 컨테이너가 삭제됩니다.
DeleteLogMarshallingArea CreateLogMarshallingArea를 성공적으로 호출하여 만든 마샬링 영역을 삭제합니다.
DumpLogRecords 지정된 로그를 검사합니다. 레코드 형식에 따라 로그 레코드를 필터링합니다. 및 는 호출자가 여는 출력 파일 스트림에 레코드를 배치합니다.
FlushLogBuffers 이 마샬링 영역에 추가된 모든 레코드를 강제로 디스크로 플러시합니다.
FlushLogToLsn 지정된 LSN(로그 시퀀스 번호)을 사용하여 이 마샬링 영역에 추가된 모든 레코드를 디스크에 강제로 플러시합니다. 이 작업 중에 지정된 것보다 많은 레코드가 플러시될 수 있습니다.
FreeReservedLog ReserveAndAppendLog, ReserveAndAppendLogAligned 또는 AllocReservedLog를 호출하여 만든 마샬링 영역에서 예약된 로그 레코드 수를 줄입니다.
GetLogContainerName 지정된 컨테이너의 전체 경로 이름을 검색합니다.
GetLogFileInformation 지정된 로그 및 해당 현재 상태에 대한 메타데이터가 포함된 버퍼를 반환하며, 이는 CLFS_INFORMATION 구조체에 의해 정의됩니다.

GetLogIoStatistics 지정된 핸들과 연결된 전용 또는 멀티플렉싱 로그에 대한 로그 I/O 통계를 검색합니다.
GetNextLogArchiveExtent 로그 보관 컨텍스트에서 다음 보관 익스텐트 집합을 검색합니다.
LsnBlockOffset 지정된 LSN에 포함된 섹터 맞춤 블록 오프셋을 반환합니다.
LsnContainer 지정된 LSN에 포함된 논리 컨테이너 ID를 검색합니다.
LsnCreate 컨테이너 ID, 블록 오프셋 및 레코드 시퀀스 번호가 지정된 LSN(로그 시퀀스 번호)을 만듭니다.
LsnRecordSequence 지정된 LSN에 포함된 레코드 시퀀스 번호를 검색합니다.
PrepareLogArchive 보관을 위한 물리적 로그를 준비합니다.
ReadLogArchiveMetadata 메타데이터의 보관 보기 범위를 지정된 버퍼에 복사합니다.
ReadLogRecord 세 가지 모드 중 하나에서 지정된 LSN(로그 시퀀스 번호)에서 읽기 시퀀스를 시작하고 지정된 로그 레코드의 첫 번째 및 읽기 컨텍스트를 반환합니다.
ReadLogRestartArea WriteLogRestartArea의 마샬링 영역과 연결된 로그에 성공적으로 기록된 마지막 다시 시작 영역을 반환합니다.
ReadNextLogRecord ReadLogRecord 또는 ReadLogRestartArea를 호출하여 시작되는 시퀀스에서 다음 레코드를 읽습니다.
ReadPreviousLogRestartArea

읽기 컨텍스트 pvReadContext에 지정된 현재 다시 시작 레코드를 기준으로 하는 이전 로그 다시 시작 영역을 읽습니다. 이 읽기 컨텍스트는 ReadLogRestartArea를 호출하여 이전에 만든 컨텍스트입니다.
RemoveLogContainer 전용 또는 멀티플렉싱된 로그 핸들과 연결된 로그에서 하나의 컨테이너를 제거합니다.
RemoveLogContainerSet 전용 또는 멀티플렉싱된 로그 핸들과 연결된 로그에서 여러 컨테이너를 제거합니다.
ReserveAndAppendLog 로그 버퍼의 공간을 예약하거나 로그 레코드를 로그에 추가하거나 둘 다 수행합니다. 함수는 atomic입니다.
ReserveAndAppendLogAligned 로그 버퍼의 공간을 예약하거나 로그 레코드 또는 둘 다에 로그 레코드를 추가합니다.
ScanLogContainers 로그 컨테이너를 열거합니다. 이 함수를 반복적으로 호출하여 모든 로그 컨테이너를 반복합니다.
SetEndOfLog 이 함수는 더 이상 사용되지 않습니다. 대신 TruncateLog를 사용합니다.
SetLogArchiveMode 지정된 로그에 대한 로그 보관 지원을 사용하거나 사용하지 않도록 설정합니다.
SetLogArchiveTail 보관 가능한 로그의 마지막 LSN(보관 로그 시퀀스 번호) 또는 보관 꼬리를 설정합니다.
TerminateLogArchive PrepareLogArchive에서 로그 보관 컨텍스트에 대해 원래 할당된 시스템 리소스의 할당을 취소합니다.
TerminateReadLog 읽기 컨텍스트를 종료합니다. 이 함수는 지정된 읽기 컨텍스트와 연결된 시스템 할당 리소스를 해제합니다. 이 함수를 호출한 후 로그 레코드를 읽으려고 시도하지 마세요. 확정되지 않은 결과를 받게 됩니다.
TruncateLog

로그를 자립니다. 함수는 로그의 끝을 지정된 값으로 설정합니다.
ValidateLog 로그 보관 전과 로그 복원 후 로그 메타데이터 및 데이터의 일관성을 확인합니다.
WriteLogRestartArea 새 클라이언트 다시 시작 영역을 로그에 추가하고 필요에 따라 로그의 LSN(기본 로그 시퀀스 번호)을 앞으로 이동합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

cloneviewhelper.h 헤더

아티클2023. 08. 22.

이 헤더는 디스플레이 디바이스 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 디바이스 참조 표시

cloneviewhelper.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

어댑터 어댑터 구조는 그래픽 어댑터를 설명합니다.
어댑터 어댑터 구조에는 그래픽 어댑터 목록이 포함되어 있습니다.
DisplayMode DisplayMode 구조는 디스플레이 디바이스를 설명합니다.
DisplayModes DisplayModes 구조체에는 디스플레이 모드 목록이 포함되어 있습니다.
Sources 원본 구조에는 VidPN(Video Present Network) 토폴로지를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

cluadmex.h 헤더

아티클2023. 03. 08.

이 헤더는 Windows 클러스터링에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 클러스터링

cluadmex.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IGetClusterDataInfo IGetClusterDataInfo 인터페이스는 장애 조치(failover) 클러스터 관리자 확장에 의해 호출되어 클러스터에 대한 정보를 검색합니다.
IGetClusterGroupInfo IGetClusterGroupInfo 인터페이스는 장애 조치(failover) 클러스터 관리자 확장에 의해 호출되어 그룹에 대한 정보를 검색합니다.
IGetClusterNetInterfaceInfo IGetClusterNetInterfaceInfo 인터페이스는 장애 조치(failover) 클러스터 관리자 확장에 의해 호출되어 네트워크 인터페이스에 대한 정보를 검색합니다.
IGetClusterNetworkInfo 장애 조치(failover) 클러스터 관리자 확장에 의해 호출되어 네트워크에 대한 정보를 검색합니다.
IGetClusterNodeInfo IGetClusterNodeInfo 인터페이스는 장애 조치(failover) 클러스터 관리자 확장에 의해 호출되어 노드에 대한 정보를 검색합니다.
IGetClusterObjectInfo 클러스터 개체에 대한 정보를 검색하기 위해 장애 조치(failover) 클러스터 관리자 확장에 의해 호출됩니다.
IGetClusterResourceInfo 리소스에 대한 정보를 검색하기 위해 장애 조치(failover) 클러스터 관리자 확장에 의해 호출됩니다.

IGetClusterUIInfo 장애 조치(failover) 클러스터 관리자의 사용자 인터페이스에 대한 정보를 검색하기 위해 장애 조치(failover) 클러스터 관리자 확장에 의해 호출됩니다.
IWCContextMenuCallback IWCContextMenuCallback 인터페이스는 장애 조치(failover) 클러스터 관리자 확장에 의해 호출되어 장애 조치(failover) 클러스터 관리자 상황에 맞는 메뉴에 항목을 추가합니다.
IWCPropertySheetCallback IWCPropertySheetCallback 인터페이스는 장애 조치(failover) 클러스터 관리자 확장에 의해 호출되어 장애 조치(failover) 클러스터 관리자 속성 시트에 속성 페이지를 추가합니다.
IWCWizard97Callback IWCWizard97Callback 인터페이스는 장애 조치(failover) 클러스터 관리자 확장에 의해 호출되어 Wizard97 속성 페이지를 Wizard97 마법사(예: 클러스터 애플리케이션 마법사)에 추가합니다.
IWCWizardCallback IWCWizardCallback 인터페이스는 장애 조치(failover) 클러스터 관리자 확장에 의해 호출되어 장애 조치(failover) 클러스터 관리자 마법사에 속성 페이지를 추가하고 탐색을 관리합니다.
IWEExtendContextMenu IWEExtendContextMenu 인터페이스를 구현하여 클러스터 개체에 대한 장애 조치(failover) 클러스터 관리자 상황에 맞는 메뉴를 확장합니다.
IWEExtendPropertySheet IWEExtendPropertySheet 인터페이스를 구현하여 클러스터 개체에 대한 속성 시트 페이지를 만들고 장애 조치(failover) 클러스터 관리자 속성 시트에 추가합니다.
IWEExtendWizard IWEExtendWizard 인터페이스를 구현하여 장애 조치(failover) 클러스터 관리자의 새 리소스 마법사 또는 클러스터 애플리케이션 마법사에 마법사 페이지를 추가합니다.
IWEExtendWizard97 IWEExtendWizard97 인터페이스를 구현하여 마법사97 스타일 마법사 페이지를 장애 조치(failover) 클러스터 관리자 마법사에 추가합니다.
IWEInvokeCommand 장애 조치(failover) 클러스터 관리자는 사용자가 IWEExtendContextMenu 인터페이스를 사용하여 만든 상황에 맞는 메뉴 항목을 선택할 때 IWEInvokeCommand 인터페이스의 구현을 호출합니다.

clusapi.h 헤더

아티클2023. 03. 08.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 장애 조치(failover) 클러스터 WMI 공급자
- Windows 클러스터링

clusapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AddClusterGroupDependency 두 클러스터 그룹 간에 종속성을 추가합니다.
AddClusterGroupSetDependency 두 클러스터 그룹 집합 간의 종속성을 추가합니다.
AddClusterGroupToGroupSetDependency 클러스터 그룹과 클러스터 그룹 집합 간의 종속성을 추가합니다.
AddClusterNode 기존 클러스터에 노드를 추가합니다.
AddClusterResourceDependency 두 리소스 간의 종속성 관계를 만듭니다.
AddClusterResourceNode 리소스를 실행할 수 있는 가능한 노드 목록에 노드를 추가합니다.
AddResourceToClusterSharedVolumes 클러스터 공유 볼륨에 스토리지를 추가합니다.
ALIGN_CLUSPROP 값 목록 내에서 구조체를 올바르게 맞춥니다.

BackupClusterDatabase 클러스터 데이터베이스 및 모든 레지스트리 검사점의 백업을 만듭니다.
CancelClusterGroupOperation 클라이언트가 그룹에 대해 보류 중인 MoveClusterGroup 또는 MoveClusterGroupEx 작업을 취소할 수 있도록 합니다. 그런 다음 그룹은 영구 상태로 돌아갑니다.
CanResourceBeDependent 한 리소스가 다른 리소스에 종속될 수 있는지 여부를 결정합니다.
ChangeClusterResourceGroup 한 그룹에서 다른 그룹으로 리소스를 이동합니다.
CloseCluster 클러스터 핸들을 닫습니다.
CloseClusterGroup 그룹 핸들을 닫습니다.
CloseClusterGroupSet OpenClusterGroupSet에서 반환된 그룹 집합 핸들을 닫습니다.
CloseClusterNetInterface 네트워크 인터페이스 핸들을 닫습니다.
CloseClusterNetwork 네트워크 핸들을 닫습니다.
CloseClusterNode 노드 핸들을 닫습니다.
CloseClusterNotifyPort CreateClusterNotifyPort를 통해 설정된 알림 포트를 닫습니다.
CloseClusterResource 리소스 핸들을 닫습니다.

CLUSCTL_GET_ACCESS_MODE 제어 코드에서 액세스 코드를 추출합니다. 제어 코드를 구성하는 다양한 코드에 대한 자세한 내용은 제어 코드 아키텍처를 참조하세요.
CLUSCTL_GET_CONTROL_FUNCTION 제어 코드에서 작업 코드를 추출합니다. 제어 코드를 구성하는 다양한 코드에 대한 자세한 내용은 제어 코드 아키텍처를 참조하세요.
CLUSCTL_GET_CONTROL_OBJECT 컨트롤 코드에서 개체 코드를 추출합니다. 제어 코드를 구성하는 다양한 코드에 대한 자세한 내용은 제어 코드 아키텍처를 참조하세요.
CLUSCTL_GET_USER 컨트롤 코드에서 사용자 비트를 검색합니다. 자세한 내용은 제어 코드 아키텍처를 참조하세요.
CLUSCTL_USER_CODE 올바른 형식의 사용자 정의 컨트롤 코드를 생성합니다. 컨트롤 코드의 비트 레이아웃에 대한 자세한 내용은 제어 코드 아키텍처를 참조하세요.
CLUSPROP_BINARY_DECLARE rgb 멤버가 호출자가 결정한 크기로 설정된 CLUSPROP_BINARY 구조를 만듭니다.
CLUSPROP_PROPERTY_NAME_DECLARE sz 멤버가 호출자에 의해 결정된 크기로 설정된 CLUSPROP_PROPERTY_NAME 구조를 만듭니다.
CLUSPROP_SZ_DECLARE sz 멤버가 호출자에 의해 결정된 크기로 설정된 CLUSPROP_SZ 구조를 만듭니다.
CLUSTER_GET_MAJOR_VERSION 클러스터 서비스 버전 번호의 주 버전 부분을 추출합니다.
CLUSTER_GET_MINOR_VERSION 클러스터 서비스 버전 번호의 부 버전 부분을 추출합니다.
CLUSTER_GET_UPGRADE_VERSION 클러스터 서비스 버전 번호의 업그레이드 버전 부분을 추출합니다.

CLUSTER_MAKE_VERSION 클러스터 서비스 버전 번호를 만듭니다.
ClusterAddGroupToGroupSet 클러스터의 그룹 집합에 지정된 그룹을 추가합니다.
ClusterCloseEnum ClusterOpenEnum에서 원래 연 클러스터 열거형 핸들을 닫습니다.
ClusterCloseEnumEx ClusterOpenEnumEx 함수에서 연 열거형에 대한 핸들을 닫습니다.
ClusterControl 클러스터에 영향을 주는 작업을 시작합니다.
ClusterEnum 클러스터의 클러스터 개체를 열거하고 호출할 때마다 한 개체의 이름을 반환합니다.
ClusterEnumEx 클러스터의 개체를 열거한 다음 클러스터 개체의 이름과 속성을 가져옵니다.
ClusterGetEnumCount 클러스터 열거형 핸들과 연결된 클러스터 개체의 수를 반환합니다.
ClusterGetEnumCountEx 클러스터 열거형 핸들과 연결된 클러스터 개체의 수를 반환합니다.
ClusterGroupCloseEnum 그룹 열거형 핸들을 닫습니다.
ClusterGroupCloseEnumEx 열거형을 닫고 hGroupEnumEx 핸들이 보유한 모든 메모리를 해제합니다.
ClusterGroupControl 그룹에 영향을 주는 작업을 시작합니다. 수행되는 작업은 dwControlCode 매개 변수에 전달된 제어 코드에 따라 달라집니다.

ClusterGroupEnum 그룹의 기본 소유자인 그룹 또는 노드의 리소스를 열거하고 각 호출과 함께 리소스 또는 노드의 이름을 반환합니다.
ClusterGroupEnumEx 열거형의 항목을 검색합니다.
ClusterGroupGetEnumCount 그룹 열거형 핸들과 연결된 클러스터 개체의 수를 반환합니다.
ClusterGroupGetEnumCountEx 열거형의 요소 수를 반환합니다.
ClusterGroupOpenEnum 그룹의 리소스 및/또는 기본 설정 소유자 목록에 포함된 노드를 반복하기 위한 열거자를 엽니다.
ClusterGroupOpenEnumEx 그룹 열거형에 대한 핸들을 엽니다.
ClusterGroupSetCloseEnum 그룹 집합에 대한 열린 열거형을 닫습니다.
ClusterGroupSetControl 그룹 집합에 영향을 주는 작업을 시작합니다.
ClusterGroupSetEnum 다음 열거 가능 개체를 반환합니다.
ClusterGroupSetGetEnumCount 열거자의 컬렉션에 포함된 항목 수를 가져옵니다.
ClusterGroupSetOpenEnum 클러스터에 대한 그룹 집합의 열거형을 시작합니다.
ClusterNetInterfaceCloseEnum 네트워크 인터페이스 열거형 핸들을 닫습니다.

ClusterNetInterfaceControl 네트워크 인터페이스에 영향을 주는 작업을 시작합니다. 수행되는 작업은 dwControlCode 매개 변수에 전달된 제어 코드에 따라 달라집니다.
ClusterNetInterfaceEnum 클러스터에 설치된 네트워크 인터페이스를 열거하고 각 호출과 함께 하나의 이름을 반환합니다.
ClusterNetInterfaceOpenEnum 설치된 네트워크 인터페이스를 반복하기 위한 열거자를 엽니다.
ClusterNetworkCloseEnum 네트워크 열거형 핸들을 닫습니다.
ClusterNetworkControl 네트워크에서 작업을 시작합니다. 수행되는 작업은 dwControlCode 매개 변수에 전달된 제어 코드에 따라 달라집니다.
ClusterNetworkEnum 네트워크에서 클러스터 개체를 열거하고 호출할 때마다 한 개체의 이름을 반환합니다.
ClusterNetworkGetEnumCount 네트워크 열거형 핸들과 연결된 클러스터 개체의 수를 반환합니다.
ClusterNetworkOpenEnum 네트워크의 개체를 반복하기 위한 열거자를 엽니다.
ClusterNodeCloseEnum 노드 열거형 핸들을 닫습니다. (ClusterNodeCloseEnum)
ClusterNodeCloseEnumEx 노드 열거형 핸들을 닫습니다. (ClusterNodeCloseEnumEx)
ClusterNodeControl 노드에 영향을 주는 작업을 시작합니다. 수행되는 작업은 dwControlCode 매개 변수에 전달된 제어 코드에 따라 달라집니다.

ClusterNodeEnum 노드에 설치된 네트워크 인터페이스 또는 그룹을 열거하고 각 호출과 함께 각각의 이름을 반환합니다.
ClusterNodeEnumEx CLUSTER_ENUM_ITEM 열거형에서 지정된 클러스터 노드를 검색합니다.
ClusterNodeGetEnumCount 노드 열거형 핸들과 연결된 클러스터 개체의 수를 반환합니다.
ClusterNodeGetEnumCountEx 노드 열거형 핸들과 연결된 클러스터 개체의 수를 반환합니다.
ClusterNodeOpenEnum 노드에 설치된 네트워크 인터페이스 또는 그룹을 반복하기 위한 열거자를 엽니다.
ClusterNodeOpenEnumEx 노드에 설치된 네트워크 인터페이스 또는 그룹을 반복할 수 있는 열거자를 엽니다.
ClusterOpenEnum 클러스터의 클러스터 개체를 반복하기 위한 열거자를 엽니다.
ClusterOpenEnumEx 개체를 반복하기 위해 클러스터에 대한 핸들을 엽니다.
ClusterRegBatchAddCommand 클러스터 레지스트리 키에서 실행될 일괄 처리에 명령을 추가합니다.
ClusterRegBatchCloseNotification 일괄 처리 알림과 연결된 메모리를 해제합니다.
ClusterRegBatchReadCommand 일괄 처리 알림에서 명령을 읽습니다.
ClusterRegCloseBatch ClusterRegCreateBatch 함수에서 만든 일괄 처리를 실행하거나 무시합니다. (ClusterRegCloseBatch)

ClusterRegCloseBatchEx ClusterRegCreateBatch 함수에서 만든 일괄 처리를 실행하거나 무시합니다. (ClusterRegCloseBatchEx)
ClusterRegCloseBatchNotifyPort ClusterRegCreateBatchNotifyPort 함수에서 만든 일괄 처리 알림 포트에 대한 구독을 닫습니다.
ClusterRegCloseKey 클러스터 데이터베이스 키의 핸들을 해제합니다.
ClusterRegCloseReadBatch 읽기 일괄 처리를 실행하고 읽기 일괄 처리 실행의 결과를 반환합니다. (ClusterRegCloseReadBatch)
ClusterRegCloseReadBatchEx 읽기 일괄 처리를 실행하고 읽기 일괄 처리 실행의 결과를 반환합니다. (ClusterRegCloseReadBatchEx)
ClusterRegCloseReadBatchReply 읽기 일괄 처리 결과 핸들을 닫고 연결된 메모리를 해제합니다.
ClusterRegCreateBatch 클러스터 레지스트리 키에서 명령을 실행할 일괄 처리를 만듭니다.
ClusterRegCreateBatchNotifyPort 일괄 처리 알림 포트에 대한 구독을 만듭니다.
ClusterRegCreateKey 지정된 클러스터 데이터베이스 키를 만듭니다. 키가 데이터베이스에 이미 있는 경우 ClusterRegCreateKey는 변경하지 않고 엽니다.
ClusterRegCreateReadBatch 클러스터 레지스트리 키에서 읽기 명령을 실행하는 읽기 일괄 처리에 대한 핸들을 만듭니다.
ClusterRegDeleteKey 클러스터 데이터베이스 키를 삭제합니다.

ClusterRegDeleteValue 클러스터 데이터베이스 키에서 명명된 값을 제거합니다.
ClusterRegEnumKey 열린 클러스터 데이터베이스 키의 하위 키를 열거합니다.
ClusterRegEnumValue 열린 클러스터 데이터베이스 키의 값을 열거합니다.
ClusterRegGetBatchNotification 일괄 처리 알림을 가져옵니다.
ClusterRegGetKeySecurity 지정된 클러스터 데이터베이스 키를 보호하는 보안 설명자의 복사본을 반환합니다.
ClusterRegOpenKey 기존 클러스터 데이터베이스 키를 엽니다.
ClusterRegQueryInfoKey 클러스터 데이터베이스 키에 대한 정보를 반환합니다.
ClusterRegQueryValue 열린 클러스터 데이터베이스 키의 값과 연결된 이름, 형식 및 데이터 구성 요소를 반환합니다.
ClusterRegReadBatchAddCommand 클러스터 레지스트리 키에서 실행되는 일괄 처리에 읽기 명령을 추가합니다.
ClusterRegReadBatchReplyNextCommand 읽기 일괄 처리 결과에서 다음 명령을 읽습니다.
ClusterRegSetKeySecurity 클러스터 데이터베이스 키에 대한 보안 특성을 설정합니다.
ClusterRegSetValue 클러스터 데이터베이스 키의 값을 설정합니다.

ClusterRegSyncDatabase 클러스터 데이터베이스를 클러스터와 동기화합니다.
ClusterRemoveGroupFromGroupSet 현재 멤버인 그룹 집합에서 지정된 그룹을 제거합니다.
ClusterResourceCloseEnum 리소스 열거형 핸들을 닫습니다.
ClusterResourceCloseEnumEx 리소스 열거형에 대한 핸들을 닫습니다.
ClusterResourceControl 리소스에 영향을 주는 작업을 시작합니다. 수행되는 작업은 dwControlCode 매개 변수에 전달된 제어 코드에 따라 달라집니다.
ClusterResourceControlAsUser 리소스에 영향을 주는 작업을 시작합니다.
ClusterResourceEnum 리소스의 종속 리소스, 노드 또는 둘 다를 열거합니다.
ClusterResourceEnumEx 리소스를 열거한 다음 현재 종속 리소스 또는 노드에 대한 포인터를 반환합니다.
ClusterResourceGetEnumCount 리소스 열거형 핸들과 연결된 클러스터 개체의 수를 반환합니다.
ClusterResourceGetEnumCountEx 리소스 열거형 핸들과 연결된 클러스터 개체의 수를 반환합니다.
ClusterResourceOpenEnum 리소스의 종속성 및 노드를 반복하기 위한 열거자를 엽니다.
ClusterResourceOpenEnumEx 리소스의 종속성 및 노드를 반복할 수 있는 리소스 열거형에 대한 핸들을 엽니다.

ClusterResourceTypeCloseEnum 리소스 종류 열거형 핸들을 닫습니다.
ClusterResourceTypeControl 리소스 종류에 영향을 주는 작업을 시작합니다. 수행되는 작업은 dwControlCode 매개 변수에 전달된 제어 코드에 따라 달라집니다.
ClusterResourceTypeControlAsUser 리소스 종류에 영향을 주는 작업을 시작합니다.
ClusterResourceTypeEnum 리소스 형식의 가능한 소유자 노드 또는 리소스를 열거합니다.
ClusterResourceTypeGetEnumCount resource_type 열거형 핸들과 연결된 클러스터 개체의 수를 반환합니다.
ClusterResourceTypeOpenEnum 리소스 형식의 가능한 소유자 노드 또는 리소스를 반복하기 위한 열거자를 엽니다.
ClusterSetAccountAccess 클러스터에 대한 ACL(계정 액세스 목록)을 업데이트.
ClusterSharedVolumeSetSnapshotState 클러스터 공유 볼륨의 스냅샷 상태를 업데이트.
ClusterUpgradeFunctionalLevel 클러스터에서 운영 체제의 롤링 업그레이드를 시작합니다. PCLUSAPI_CLUSTER_UPGRADE 이 함수에 대한 포인터를 정의합니다.
CreateCluster 클러스터를 만들고 시작합니다.
CreateClusterCNOless 클러스터 이름 및 IP 주소 리소스 없이 클러스터를 만듭니다.
CreateClusterGroup 클러스터에 그룹을 추가하고 새로 추가된 그룹에 핸들을 반환합니다.

CreateClusterGroupEx 단일 작업에서 CLUSTER_CREATE_GROUP_INFO 구조에 지정된 옵션을 사용하여 새 클러스터 그룹을 만듭니다.
CreateClusterGroupSet 클러스터에 그룹 집합을 추가하고 새로 추가된 그룹 집합에 핸들을 반환합니다.
CreateClusterNameAccount 클러스터 이름 리소스를 만든 다음 클러스터를 호스트하는 컴퓨터가 도메인의 멤버가 아닌 경우에도 클러스터를 도메인에 추가합니다.
CreateClusterNotifyPort 알림 포트를 만들거나 수정합니다. 알림 포트에 대한 자세한 내용은 클러스터 이벤트 수신을 참조하세요.
CreateClusterNotifyPortV2 알림 포트를 만들거나 수정합니다. 알림 포트에 대한 자세한 내용은 클러스터 이벤트 수신을 참조하세요.
CreateClusterResource 클러스터에 리소스를 만듭니다.
CreateClusterResourceType 클러스터에 리소스 종류를 만듭니다.
DeleteClusterGroup 클러스터에서 오프라인 및 빈 그룹을 제거합니다.
DeleteClusterGroupSet 클러스터에서 지정된 그룹 집합을 삭제합니다.
DeleteClusterResource 클러스터에서 오프라인 리소스를 제거합니다.
DeleteClusterResourceType 클러스터에서 리소스 유형을 제거합니다.

DestroyCluster 클러스터를 제거합니다.
DestroyClusterGroup 클러스터에서 지정된 그룹을 삭제합니다.
EvictClusterNode 클러스터 데이터베이스에서 노드를 삭제합니다.
EvictClusterNodeEx 클러스터에서 노드를 제거하고 노드에서 정리 작업을 시작합니다.
FailClusterResource 리소스 오류를 시작합니다.
GetClusterFromGroup 그룹과 연결된 클러스터에 대한 핸들을 반환합니다.
GetClusterFromGroupSet GetClusterFromGroupSet 함수(clusapi.h)는 그룹 집합과 연결된 클러스터에 대한 핸들을 반환합니다.
GetClusterFromNetInterface 네트워크 인터페이스와 연결된 클러스터에 대한 핸들을 반환합니다.
GetClusterFromNetwork 네트워크와 연결된 클러스터에 대한 핸들을 반환합니다.
GetClusterFromNode 노드와 연결된 클러스터에 대한 핸들을 반환합니다.
GetClusterFromResource 리소스와 연결된 클러스터에 대한 핸들을 반환합니다.
GetClusterGroupKey 그룹에 대한 클러스터 데이터베이스 하위 트리의 루트를 엽니다.

GetClusterGroupState 그룹의 현재 상태를 반환합니다.
GetClusterInformation 클러스터의 이름과 버전을 검색합니다.
GetClusterKey 클러스터에 대한 클러스터 데이터베이스 하위 트리의 루트를 엽니다.
GetClusterNetInterface 클러스터의 네트워크에 노드 인터페이스의 이름을 반환합니다.
GetClusterNetInterfaceKey 네트워크 인터페이스 개체에 대한 클러스터 데이터베이스 하위 트리의 루트를 엽니다.
GetClusterNetInterfaceState 네트워크 인터페이스의 현재 상태를 반환합니다.
GetClusterNetworkId 네트워크의 식별자를 반환합니다.
GetClusterNetworkKey 네트워크에 대한 클러스터 데이터베이스 하위 트리의 루트를 엽니다.
GetClusterNetworkState 네트워크의 현재 상태입니다.
GetClusterNodeId 클러스터 노드의 고유 식별자를 반환합니다.
GetClusterNodeKey 노드에 대한 클러스터 데이터베이스 하위 트리의 루트를 엽니다.
GetClusterNodeState 노드의 현재 상태를 반환합니다.

GetClusterNotify 알림 포트에 대해 저장된 다음 알림 이벤트와 관련된 정보입니다.
GetClusterNotifyV2 알림 포트에 대한 다음 알림 이벤트에 대한 정보를 검색합니다.
GetClusterQuorumResource 클러스터의 쿼럼 리소스 이름을 반환합니다.
GetClusterResourceDependencyExpression 지정된 리소스와 연결된 종속성 식을 검색합니다.
GetClusterResourceKey 리소스에 대한 클러스터 데이터베이스 하위 트리의 루트를 엽니다.
GetClusterResourceNetworkName 리소스가 종속된 네트워크 이름 리소스의 Name 프라이빗 속성을 검색합니다.
GetClusterResourceState 리소스의 현재 상태를 반환합니다.
GetClusterResourceTypeKey 리소스 종류에 대한 클러스터 데이터베이스 하위 트리의 루트를 엽니다.
GetCurrentClusterNodeId 현재 클러스터 노드의 고유 식별자를 반환합니다.
GetNodeClusterState 클러스터 서비스가 노드에서 설치 및 실행 중인지 여부를 결정합니다.
GetNotifyEventHandle 알림 이벤트에 대한 핸들을 검색합니다.
IsFileOnClusterSharedVolume 파일이 클러스터 공유 볼륨에 있는지 여부를 지정합니다.

MoveClusterGroup 그룹 및 모든 해당 리소스를 한 노드에서 다른 노드로 이동합니다.
MoveClusterGroupEx 플래그 및 버퍼를 추가하여 기존 MoveClusterGroup 메서드를 확장합니다.
OfflineClusterGroup 그룹을 오프라인으로 전환합니다.
OfflineClusterGroupEx OfflineClusterGroup 메서드를 확장합니다.
OfflineClusterResource 리소스를 오프라인으로 전환합니다.
OfflineClusterResourceEx OfflineClusterResource 메서드를 확장합니다.
OnlineClusterGroup 그룹을 온라인으로 가져옵니다. (OnlineClusterGroup)
OnlineClusterGroupEx 그룹을 온라인으로 가져옵니다. (OnlineClusterGroupEx)
OnlineClusterResource 오프라인 또는 실패한 리소스를 온라인으로 가져옵니다. (OnlineClusterResource)
OnlineClusterResourceEx 오프라인 또는 실패한 리소스를 온라인으로 가져옵니다. (OnlineClusterResourceEx)
OpenCluster 클러스터에 대한 연결을 열고 클러스터에 대한 핸들을 반환합니다. (OpenCluster)
OpenClusterEx 클러스터에 대한 연결을 열고 클러스터에 대한 핸들을 반환합니다. (OpenClusterEx)

OpenClusterGroup 장애 조치(failover) 클러스터 그룹을 열고 해당 그룹에 대한 핸들을 반환합니다. (OpenClusterGroup)
OpenClusterGroupEx 장애 조치(failover) 클러스터 그룹을 열고 해당 그룹에 대한 핸들을 반환합니다. (OpenClusterGroupEx)
OpenClusterGroupSet 지정된 그룹 집합에 대한 핸들을 엽니다.
OpenClusterNetInterface 네트워크 인터페이스에 대한 핸들을 엽니다. (OpenClusterNetInterface)
OpenClusterNetInterfaceEx 네트워크 인터페이스에 대한 핸들을 엽니다. (OpenClusterNetInterfaceEx)
OpenClusterNetwork 네트워크에 대한 연결을 열고 네트워크에 대한 핸들을 반환합니다. (OpenClusterNetwork)
OpenClusterNetworkEx 네트워크에 대한 연결을 열고 네트워크에 대한 핸들을 반환합니다. (OpenClusterNetworkEx)
OpenClusterNode 노드를 열고 노드에 대한 핸들을 반환합니다. (OpenClusterNode)
OpenClusterNodeEx 노드를 열고 노드에 대한 핸들을 반환합니다. (OpenClusterNodeEx)
OpenClusterResource 리소스를 열고 해당 리소스에 대한 핸들을 반환합니다. (OpenClusterResource)
OpenClusterResourceEx 리소스를 열고 해당 리소스에 대한 핸들을 반환합니다. (OpenClusterResourceEx)

PauseClusterNode 노드가 클러스터 활동을 일시적으로 일시 중단할 것을 요청합니다. PCLUSAPI_PAUSE_CLUSTER_NODE 형식은 이 함수에 대한 포인터를 정의합니다.
PauseClusterNodeEx 노드가 클러스터 활동을 일시적으로 일시 중단하는 것을 요청합니다.
RegisterClusterNotify 알림 포트에 대해 저장된 이벤트 목록에 이벤트 유형을 추가합니다.
RegisterClusterNotifyV2 이벤트 유형에 알림 키를 추가하여 알림 포트에 이벤트 유형을 등록합니다.
RegisterClusterResourceTypeNotifyV2 클러스터 알림 포트에 알림 유형을 추가합니다.
RemoveClusterGroupDependency 두 클러스터 그룹 간의 종속성을 제거합니다.
RemoveClusterGroupSetDependency 그룹 집합의 종속성 식에서 그룹 집합을 제거합니다.
RemoveClusterGroupToGroupSetDependency 그룹의 종속성 식에서 그룹 집합을 제거합니다.
RemoveClusterResourceDependency 두 리소스 간의 종속성 관계를 제거합니다.
RemoveClusterResourceNode 리소스를 호스트할 수 있는 노드 목록에서 노드를 제거합니다.
RemoveResourceFromClusterSharedVolumes 클러스터 공유 볼륨에서 스토리지를 제거합니다.
RestartClusterResource 클러스터 리소스를 다시 시작합니다.

RestoreClusterDatabase 클러스터 데이터베이스를 복원하고 함수가 호출되는 노드에서 클러스터 서비스를 다시 시작합니다. 이 노드를 복원 노드라고합니다.
ResumeClusterNode 일시 중지된 노드가 클러스터 활동을 다시 시작하라는 요청입니다. PCLUSAPI_RESUME_CLUSTER_NODE 형식은 이 함수에 대한 포인터를 정의합니다.
ResumeClusterNodeEx 지정된 장애 복구(failback) 작업을 시작한 다음 일시 중지된 노드가 클러스터 작업을 다시 시작해 주라고 요청합니다.
SetClusterGroupName 그룹의 이름을 설정합니다.
SetClusterGroupNodeList 그룹에 대한 기본 설정 노드 목록을 설정합니다.
SetClusterGroupSetDependencyExpression 클러스터 그룹 집합에 대한 종속성 식을 설정합니다.
SetClusterName 클러스터의 이름을 설정합니다.
SetClusterNetworkName 네트워크의 이름을 설정합니다.
SetClusterNetworkPriorityOrder 클러스터 노드 간의 내부 통신에 사용되는 네트워크 집합의 우선 순위 순서를 설정합니다.
SetClusterQuorumResource 클러스터에 대한 쿼럼 리소스로 리소스를 설정합니다.
SetClusterResourceDependencyExpression hResource에서 참조하는 리소스와 연결할 종속성 식을 지정합니다. 리소스에 대한 기존 종속성 관계를 덮어씁니다.

SetClusterResourceName 리소스의 이름을 설정합니다.
SetClusterServiceAccountPassword 사용 가능한 모든 클러스터 노드에서 클러스터 서비스 사용자 계정의 암호를 변경합니다.
SetGroupDependencyExpression 클러스터 그룹에 대한 종속성 식을 설정합니다.

콜백 함수

PCLUSTER_SETUP_PROGRESS_CALLBACK 클러스터 설정 진행에 대한 정기적인 업데이트를 수신하는 콜백 함수입니다.
PCLUSTER_UPGRADE_PROGRESS_CALLBACK 클러스터에서 운영 체제의 롤링 업그레이드에 대한 상태 정보를 검색합니다. PCLUSTER_UPGRADE_PROGRESS_CALLBACK 형식은 이 함수에 대한 포인터를 정의합니다.

구조체

CLUS_CHKDSK_INFO Chkdsk 작업에 대한 정보를 나타냅니다.
CLUS_CSV_MAINTENANCE_MODE_INFO CSV(클러스터 공유 볼륨)에서 유지 관리 모드를 사용하거나 사용하지 않도록 설정합니다.
CLUS_CSV_VOLUME_INFO CSV(클러스터 공유 볼륨)에 대한 정보를 나타냅니다.
CLUS_CSV_VOLUME_NAME CSV(클러스터 공유 볼륨)의 이름을 나타냅니다.

CLUS_DISK_NUMBER_INFO 실제 디스크의 디스크 번호에 대한 정보를 나타냅니다.
CLUS_DNN_LEADER_STATUS Scale-Out 파일 서버에 대한 DNN(분산 네트워크 이름) 리소스의 상태를 나타냅니다.
CLUS_DNN_SODAFS_CLONE_STATUS Scale-Out 파일 서버 복제본의 상태를 나타냅니다.
CLUS_FORCE_QUORUM_INFO 대부분의 노드 클러스터에서 쿼럼을 설정하기에 충분한 노드 목록에 대한 정보를 지정합니다.
CLUS_FTSET_INFO FT(내결함성) 집합에 대한 정보를 포함합니다. 이 구조체는 CLUSPROP_FTSET_INFO 구조체에서 값 목록에 항목을 만드는 데 사용됩니다.
CLUS_MAINTENANCE_MODE_INFO 클러스터 노드에서 유지 관리 모드를 사용하거나 사용하지 않도록 설정합니다.
CLUS_MAINTENANCE_MODE_INFOEX 스토리지 클래스 리소스에 대한 확장된 유지 관리 모드 설정을 나타냅니다.
CLUS_NETNAME_IP_INFO_ENTRY NetName 리소스의 IP 정보를 나타냅니다.
CLUS_NETNAME_IP_INFO_FOR_MULTICHANNEL 다중 채널이 사용하도록 설정된 NetName 리소스의 IP 정보를 나타냅니다.
CLUS_NETNAME_PWD_INFO 컴퓨터 이름과 연결된 보안 주체를 다시 설정하기 위한 정보를 제공합니다.
CLUS_NETNAME_VS_TOKEN_INFO 토큰을 요청하는 데 필요한 데이터를 포함합니다. CLUSCTL_RESOURCE_NETNAME_GET_VIRTUAL_SERVER_TOKEN 제어 코드의 입력 매개 변수로 사용됩니다.

CLUS_PARTITION_INFO 스토리지 클래스 리소스 볼륨 및 파일 시스템을 설명하는 데이터를 포함합니다. CLUSPROP_PARTITION_INFO 구조체의 데이터 멤버로 사용되며 일부 제어 코드 작업의 반환 값으로 사용됩니다.
CLUS_PARTITION_INFO_EX 스토리지 클래스 리소스 볼륨 및 파일 시스템에 대해 설명합니다.
CLUS_PARTITION_INFO_EX2 스토리지 클래스 리소스의 디스크 파티션 정보를 설명합니다.
CLUS_PROVIDER_STATE_CHANGE_INFO 공급자 리소스의 상태에 대한 데이터를 포함합니다.
CLUS_RESOURCE_CLASS_INFO 리소스 클래스 데이터를 포함합니다. CLUSPROP_RESOURCE_CLASS_INFO 구조체의 데이터 멤버로 사용되며 일부 제어 코드 작업의 반환 값으로 사용됩니다.
CLUS_SCSI_ADDRESS SCSI 주소 데이터를 포함합니다. CLUSPROP_SCSI_ADDRESS 구조체의 데이터 멤버로 사용되며 일부 제어 코드 작업의 반환 값으로 사용됩니다.
CLUS_SHARED_VOLUME_BACKUP_MODE CSV의 백업 모드를 설명합니다.
CLUS_STARTING_PARAMS 노드의 클러스터 서비스 시작 시도가 클러스터를 구성하거나 조인하려는 시도를 나타내는지 여부와 노드가 이전에 이 버전의 클러스터 서비스를 시작하려고 시도했는지 여부를 나타냅니다.
CLUS_STORAGE_GET_AVAILABLE_DRIVELETTERS 노드에서 사용할 수 있는 드라이버 문자의 비트 마스크를 포함합니다. CLUSCTL_RESOURCE_TYPE_STORAGE_GET_DRIVELETTERS 컨트롤 코드의 반환 값으로 사용됩니다.
CLUS_STORAGE_REMAP_DRIVELETTER 노드의 디스크 드라이브에 대한 기존 및 대상 드라이브 문자를 식별합니다.
CLUS_STORAGE_SET_DRIVELETTER 스토리지 클래스 리소스와 연결된 디스크 파티션에 대한 드라이브 문자 정보를 제공합니다.

CLUSCTL_GROUP_GET_LAST_MOVE_TIME_OUTPUT 그룹을 다른 노드로 마지막으로 이동한 시간에 대한 정보를 지정합니다.
CLUSCTL_RESOURCE_STATE_CHANGE_REASON_STRUCT 리소스 상태 변경 이유를 제공하기 위해 CLUSCTL_RESOURCE_STATE_CHANGE_REASON 제어 코드와 함께 전송됩니다.
CLUSCTL_RESOURCE_TYPE_STORAGE_GET_AVAILABLE_DISKS_EX2_INPUT CLUSCTL_RESOURCE_TYPE_STORAGE_GET_AVAILABLE_DISKS_EX2_INT 제어 코드의 입력 버퍼를 나타냅니다.
CLUSPROP_BINARY 이진 데이터 값을 설명합니다.
CLUSPROP_BUFFER_HELPER 속성 목록 또는 값 목록을 작성하거나 구문 분석하는 데 사용됩니다.
CLUSPROP_DWORD 디스크의 실제 드라이브를 식별하는 숫자 값에 대해 설명합니다.
CLUSPROP_FILETIME 파일의 날짜 및 타임스탬프를 설명합니다.
CLUSPROP_FTSET_INFO FT(내결함성) 집합에 대한 정보를 포함합니다. 값 목록의 항목으로 사용되며 CLUSPROP_VALUE 및 CLUS_FTSET_INFO 구조로 구성됩니다.
CLUSPROP_LARGE_INTEGER 부 서명된 큰 정수에 대해 설명합니다.
CLUSPROP_LIST 속성 목록의 시작 부분에 액세스합니다.
CLUSPROP_LONG 서명된 LONG 데이터에 대해 설명합니다.

CLUSPROP_PARTITION_INFO 스토리지 클래스 리소스와 관련된 정보를 포함합니다.
CLUSPROP_PARTITION_INFO_EX CLUSPROP_PARTITION_INFO_EX 구조에는 스토리지 클래스 리소스와 관련된 정보가 포함됩니다.
CLUSPROP_PARTITION_INFO_EX2 스토리지 클래스 리소스에 대한 파티션 정보를 포함하는 값 목록 항목입니다. 이 구조체는 입력이며 CLUSCTL_RESOURCE_STORAGE_GET_DISK_INFO_EX2 제어 코드의 반환 값입니다.
CLUSPROP_REQUIRED_DEPENDENCY 다른 리소스의 필수 종속성인 리소스에 대해 설명합니다. 이 공용 구조체는 CLUSCTL_RESOURCE_GET_REQUIRED_DEPENDENCIES 또는 CLUSCTL_RESOURCE_TYPE_GET_REQUIRED_DEPENDENCIES 제어 코드 작업에서 반환된 값 목록의 값으로 사용됩니다.
CLUSPROP_RESOURCE_CLASS 리소스 클래스에 대해 설명합니다.
CLUSPROP_RESOURCE_CLASS_INFO 리소스 클래스와 관련된 정보를 설명합니다.
CLUSPROP_SCSI_ADDRESS SCSI 디바이스의 주소를 설명합니다.
CLUSPROP_SECURITY_DESCRIPTOR 보안 설명자를 설명합니다.
CLUSPROP_SYNTAX 데이터 값의 형식과 형식을 설명합니다. CLUSPROP_VALUE 구조체의 구문 멤버로 사용됩니다.
CLUSPROP_SZ 여러 NULL 종료 유니코드 문자열에 대해 설명합니다.
CLUSPROP_ULARGE_INTEGER 부호 없는 큰 정수에 대해 설명합니다.

CLUSPROP_VALUE 값 목록에 사용되는 데이터 값의 구문과 길이를 설명합니다. CLUSPROP_VALUE 구조체는 CLUSPROP_BINARY 및 CLUSPROP_SZ 같은 특정 형식의 데이터를 설명하는 모든 구조체에서 제네릭 헤더로 사용됩니다.
CLUSPROP_WORD 숫자 데이터를 설명합니다.
CLUSTER_BATCH_COMMAND 현재 일괄 처리 명령 데이터가 ClusterRegBatchReadCommand 함수로 전송되는 순서를 나타냅니다.
CLUSTER_CREATE_GROUP_INFO CLUSTER_CREATE_GROUP_INFO 호출자는 새 그룹을 만들 때 추가 속성을 제공할 수 있습니다.
CLUSTER_ENUM_ITEM CLUSTER_ENUM_ITEM 클러스터 개체의 속성을 포함하며 ClusterEnumEx 및 ClusterNodeEnumEx 함수에서 클러스터를 열거하는 데 사용됩니다.
CLUSTER_GROUP_ENUM_ITEM CLUSTER_GROUP_ENUM_ITEM(clusapi.h)는 클러스터 그룹의 속성을 포함하며 ClusterGroupEnumEx 함수에서 클러스터 그룹을 열거하는 데 사용됩니다.
CLUSTER_IP_ENTRY 클러스터의 IP 주소를 설명합니다.
CLUSTER_MEMBERSHIP_INFO CLUSTER_MEMBERSHIP_INFO(clusapi.h) 구조체는 클러스터의 멤버 자격 정보를 나타냅니다.
CLUSTER_READ_BATCH_COMMAND 읽기 일괄 처리의 단일 명령에 대한 결과를 나타냅니다.
CLUSTER_RESOURCE_ENUM_ITEM CLUSTER_RESOURCE_ENUM_ITEM 클러스터 리소스의 속성을 나타내며 ClusterResourceEnumEx 함수에서 클러스터 리소스를 열거하는 데 사용됩니다.

CLUSTER_SET_PASSWORD_STATUS SetClusterServiceAccountPassword 함수에서 각 클러스터 노드에 대한 클러스터 서비스 사용자 계정 암호 변경 결과를 반환하는 데 사용됩니다.
CLUSTER_SHARED_VOLUME_STATE_INFO CSV(클러스터 공유 볼륨)의 상태에 대한 정보를 나타냅니다. (CLUSTER_SHARED_VOLUME_STATE_INFO)
CLUSTER_SHARED_VOLUME_STATE_INFO_EX CSV(클러스터 공유 볼륨)의 상태에 대한 정보를 나타냅니다. (CLUSTER_SHARED_VOLUME_STATE_INFO_EX)
CLUSTER_VALIDATE_CSV_FILENAME 유효성 검사 작업 중 CSV(클러스터 공유 볼륨)를 나타냅니다.
CLUSTER_VALIDATE_DIRECTORY CLUSTER_VALIDATE_DIRECTORY(clusapi.h) 구조체는 유효성을 검사하기 위해 디렉터리에 전달되며, 자세한 내용은 확인할 수 있습니다.
CLUSTER_VALIDATE_NETNAME CLUSTER_VALIDATE_NETNAME(clusapi.h) 구조체는 유효성을 검사하기 위해 네트워크 이름을 전달하며, 더 많은 정보를 확인할 수 있습니다.
CLUSTER_VALIDATE_PATH CLUSTER_VALIDATE_PATH(clusapi.h) 구조체는 유효성을 검사하기 위해 네트워크 이름을 전달하며 더 많은 정보를 확인할 수 있습니다.
CLUSTERVERSIONINFO 노드에 로컬로 설치된 클러스터 서비스 버전에 대한 정보를 설명합니다.
CLUSTERVERSIONINFO_NT4 CLUSTERVERSIONINFO_NT4(clusapi.h) 구조체는 여러 필드 형식과 관련이 있으며 자세한 내용은 결정해야 합니다.
CREATE_CLUSTER_CONFIG 초기 클러스터 구성을 정의합니다.

CREATE_CLUSTER_NAME_ACCOUNT CreateClusterNameAccount 함수에서 클러스터를 도메인에 추가하는 데 사용하는 클러스터 이름 리소스 및 도메인 자격 증명에 대해 설명합니다. PCREATE_CLUSTER_NAME_ACCOUNT 이 구조체에 대한 포인터를 정의합니다.
FILESHARE_CHANGE 이벤트 알림 목록의 항목 형식에 대해 설명합니다.
FILESHARE_CHANGE_LIST 파일 서버 리소스에서 관리하는 파일 공유에 대한 이벤트 알림 목록을 설명합니다.
GROUP_FAILURE_INFO GROUP_FAILURE_INFO(clusapi.h) 구조체는 그룹 오류에 대한 장애 조치(failover) 시도에 대한 정보를 나타냅니다.
GROUP_FAILURE_INFO_BUFFER GROUP_FAILURE_INFO_BUFFER(clusapi.h) 구조체는 GROUP_FAILURE_INFO 구조체에 대한 버퍼를 나타냅니다.
NOTIFY_FILTER_AND_TYPE NOTIFY_FILTER_AND_TYPE(clusapi.h) 구조체는 CreateClusterNotifyPortV2 함수에서 만든 알림 포트에 대한 필터를 나타냅니다.
RESOURCE_FAILURE_INFO RESOURCE_FAILURE_INFO(clusapi.h) 구조체는 리소스에 대한 장애 조치(failover) 시도에 대한 정보를 나타냅니다.
RESOURCE_FAILURE_INFO_BUFFER RESOURCE_FAILURE_INFO_BUFFER(clusapi.h) 구조체는 리소스 오류에 대한 버퍼를 나타냅니다.
RESOURCE_TERMINAL_FAILURE_INFO_BUFFER RESOURCE_TERMINAL_FAILURE_INFO_BUFFER(clusapi.h) 구조체는 리소스의 터미널 오류에 대한 버퍼를 나타냅니다.
SR_RESOURCE_TYPE_DISK_INFO 디스크가 복제에 적합한지 여부를 나타내는 정보 집합에 대해 설명합니다.

SR_RESOURCE_TYPE_ELIGIBLE_DISKS_RESULT 스토리지 복제를 위한 리소스 유형 제어 코드 작업에서 검색한 데이터 디스크 집합에 대해 설명합니다.
SR_RESOURCE_TYPE_QUERY_ELIGIBLE_LOGDISKS 지정된 데이터 디스크의 로그 디스크로 사용할 수 있는 검색된 디스크 집합에 대해 설명합니다.
SR_RESOURCE_TYPE_QUERY_ELIGIBLE_SOURCE_DATADISKS 복제를 위한 원본 사이트로 사용할 수 있는 검색된 데이터 디스크 집합에 대해 설명합니다.
SR_RESOURCE_TYPE_QUERY_ELIGIBLE_TARGET_DATADISKS 복제를 위한 대상 사이트로 사용할 수 있는 검색된 데이터 디스크 집합에 대해 설명합니다.
SR_RESOURCE_TYPE_REPLICATED_DISK 복제된 디스크를 나타냅니다.
SR_RESOURCE_TYPE_REPLICATED_DISKS_RESULT 복제된 디스크의 검색된 집합에 대해 설명합니다.
SR_RESOURCE_TYPE_REPLICATED_PARTITION_ARRAY 디스크에 복제된 모든 파티션을 나열합니다.
SR_RESOURCE_TYPE_REPLICATED_PARTITION_INFO 복제된 파티션에 대해 설명합니다.

열거형

CLCTL_CODES CLCTL_CODES(clusapi.h)는 제어 코드가 수행할 수 있는 작업을 열거합니다.
CLUS_CHARACTERISTICS CLUS_CHARACTERISTICS(clusapi.h)는 리소스 종류 및 리소스의 특성을 열거합니다.

CLUS_FLAGS CLUS_FLAGS(clusapi.h)는 리소스 또는 그룹을 핵심 리소스로 식별합니다.
CLUS_GROUP_START_SETTING CLUS_GROUP_START_SETTING(clusapi.h)는 클러스터 그룹에 대한 시작 설정을 열거합니다.
CLUS_RESSUBCLASS CLUS_RESSUBCLASS(clusapi.h)는 공유 리소스를 관리하는 리소스 하위 클래스를 식별합니다.
CLUS_RESSUBCLASS_NETWORK CLUS_RESSUBCLASS_NETWORK(clusapi.h)는 IP 주소 공급자를 관리하는 리소스 하위 클래스를 식별합니다.
CLUS_RESSUBCLASS_STORAGE CLUS_RESSUBCLASS_STORAGE(clusapi.h)는 공유 버스를 관리하는 리소스 하위 클래스를 식별합니다.
CLUSCTL_CLUSTER_CODES CLUSCTL_CLUSTER_CODES(clusapi.h)는 ClusterControl 함수에서 사용하는 클러스터 제어 코드를 열거합니다.
CLUSCTL_GROUP_CODES CLUSCTL_GROUP_CODES(clusapi.h)는 groupcontrol 코드를 열거합니다.
CLUSCTL_GROUPSET_CODES CLUSCTL_GROUPSET_CODES(clusapi.h)는 ClusterGroupSetControl 함수에서 사용하는 groupset 제어 코드를 열거합니다.
CLUSCTL_NETINTERFACE_CODES CLUSCTL_NETINTERFACE_CODES(clusapi.h)는 네트워크 인터페이스 제어 코드를 열거합니다.
CLUSCTL_NETWORK_CODES CLUSCTL_NETWORK_CODES(clusapi.h)는 네트워크 제어 코드를 열거합니다.
CLUSCTL_NODE_CODES CLUSCTL_NODE_CODES(clusapi.h)는 노드 제어 코드를 열거합니다.

CLUSCTL_RESOURCE_CODES CLUSCTL_RESOURCE_CODES(clusapi.h)는 리소스 제어 코드를 열거합니다.
CLUSCTL_RESOURCE_TYPE_CODES CLUSCTL_RESOURCE_TYPE_CODES(clusapi.h)는 리소스 형식 제어 코드를 열거합니다.
CLUSGROUP_TYPE CLUSGROUP_TYPE(clusapi.h)는 만들 클러스터 그룹의 유형을 지정합니다.
CLUSPROP_IPADDR_ENABLENETBIOS CLUSPROP_DWORD 구조와 함께 사용하는 경우 IP 주소 리소스의 EnableNetBIOS 속성 기능을 사용하거나 사용하지 않도록 설정합니다.
CLUSPROP_PIFLAGS CLUSPROP_PIFLAGS(clusapi.h)는 디스크 파티션 정보를 나타냅니다. 열거형 플래그는 스토리지 클래스 리소스인 디스크 파티션의 특정 속성을 식별합니다.
CLUSTER_CHANGE CLUSTER_CHANGE(clusapi.h)는 반환된 알림의 유형을 설명합니다.
CLUSTER_CHANGE_CLUSTER_V2 CLUSTER_CHANGE_CLUSTER_V2(clusapi.h)는 클러스터에 대해 생성된 알림 목록을 정의합니다.
CLUSTER_CHANGE_GROUP_V2 CLUSTER_CHANGE_GROUP_V2(clusapi.h)는 그룹에 대해 생성된 알림 목록을 정의합니다.
CLUSTER_CHANGE_GROUPSET_V2 CLUSTER_CHANGE_GROUPSET_V2(clusapi.h)는 그룹 집합에 대해 생성된 알림 목록을 정의합니다.
CLUSTER_CHANGE_NETINTERFACE_V2 CLUSTER_CHANGE_NETINTERFACE_V2(clusapi.h)는 클러스터 네트워크 인터페이스에 대해 생성된 알림 집합을 정의합니다.
CLUSTER_CHANGE_NETWORK_V2 CLUSTER_CHANGE_NETWORK_V2(clusapi.h)는 클러스터 네트워크에 대해 생성된 알림을 정의합니다.

CLUSTER_CHANGE_NODE_UPGRADE_PHASE_V2 CLUSTER_CHANGE_NODE_UPGRADE_PHASE_V2(clusapi.h)는 클러스터 노드 업그레이드를 위해 생성된 알림을 정의합니다.
CLUSTER_CHANGE_NODE_V2 CLUSTER_CHANGE_NODE_V2(clusapi.h)는 클러스터 노드에 대해 생성된 알림을 정의합니다.
CLUSTER_CHANGE_QUORUM_V2 CLUSTER_CHANGE_QUORUM_V2(clusapi.h)는 쿼럼 관련 정보에 대해 생성된 알림을 정의합니다.
CLUSTER_CHANGE_REGISTRY_V2 CLUSTER_CHANGE_REGISTRY_V2(clusapi.h)는 레지스트리 키에 대해 생성된 알림을 정의합니다.
CLUSTER_CHANGE_RESOURCE_TYPE_V2 CLUSTER_CHANGE_RESOURCE_TYPE_V2(clusapi.h)는 리소스 종류에 대해 생성된 알림 집합을 정의합니다.
CLUSTER_CHANGE_RESOURCE_V2 CLUSTER_CHANGE_RESOURCE_V2(clusapi.h)는 리소스에 대해 생성된 알림 목록을 정의합니다.
CLUSTER_CHANGE_SHARED_VOLUME_V2 CLUSTER_CHANGE_SHARED_VOLUME_V2(clusapi.h)는 클러스터 공유 볼륨에 대해 생성된 알림을 정의합니다.
CLUSTER_CHANGE_SPACEPORT_V2 CLUSTER_CHANGE_SPACEPORT_V2(clusapi.h)는 열거형입니다.
CLUSTER_CONTROL_OBJECT CLUSTER_CONTROL_OBJECT(clusapi.h)는 컨트롤 코드가 적용되는 클러스터 개체의 형식을 나타내는 컨트롤 코드의 8비트 개체 구성 요소입니다. 자세한 내용은 제어 코드 아키텍처를 참조하세요.
CLUSTER_CSV_VOLUME_FAULT_STATE CSV(클러스터 공유 볼륨)에 대한 다양한 오류 상태를 정의합니다.
CLUSTER_ENUM CLUSTER_ENUM(clusapi.h)는 열거되는 클러스터 개체의 형식을 설명합니다.

CLUSTER_GROUP_AUTOFAILBACK_TYPE CLUSTER_GROUP_AUTOFAILBACK_TYPE(clusapi.h)는 AutoFailbackType 그룹 공통 속성에서 해당 노드가 장애 조치(failover) 후 다시 온라인 상태가 될 때 기본 소유자로 식별된 노드로 장애 복구(failback)해야 하는지 여부를 지정하는 데 사용됩니다.
CLUSTER_GROUP_ENUM CLUSTER_GROUP_ENUM(clusapi.h)는 ClusterGroupEnum 및 ClusterGroupOpenEnum 함수에 의해 열거되는 클러스터 개체의 형식을 설명합니다.
CLUSTER_GROUP_PRIORITY CLUSTER_GROUP_PRIORITY(clusapi.h)는 그룹의 우선 순위 수준을 지정합니다.
CLUSTER_GROUP_STATE CLUSTER_GROUP_STATE(clusapi.h)는 그룹의 가능한 상태를 열거합니다.
CLUSTER_MGMT_POINT_TYPE CLUSTER_MGMT_POINT_TYPE(clusapi.h)는 클러스터에 대한 관리 지점의 유형을 지정합니다.
CLUSTER_NETINTERFACE_STATE CLUSTER_NETINTERFACE_STATE(clusapi.h)는 네트워크 인터페이스 상태의 가능한 값을 열거합니다.
CLUSTER_NETWORK_ENUM CLUSTER_NETWORK_ENUM(clusapi.h)는 ClusterNetworkEnum 및 ClusterNetworkOpenEnum 함수에 의해 열거되는 클러스터 개체의 형식을 설명합니다.
CLUSTER_NETWORK_ROLE CLUSTER_NETWORK_ROLE(clusapi.h)는 클러스터에서 네트워크가 수행하는 역할을 설명합니다.
CLUSTER_NETWORK_STATE CLUSTER_NETWORK_STATE(clusapi.h)는 네트워크 상태의 가능한 값을 열거합니다.
CLUSTER_NODE_DRAIN_STATUS CLUSTER_NODE_DRAIN_STATUS(clusapi.h)는 노드 드레이닝 상태의 가능한 값을 열거합니다.
CLUSTER_NODE_ENUM CLUSTER_NODE_ENUM(clusapi.h)는 ClusterNodeEnum 및 ClusterNodeOpenEnum 함수에 의해 열거되는 클러스터 개체의 형식을 설명합니다.

CLUSTER_NODE_RESUME_FAILBACK_TYPE ResumeClusterNodeEx 함수에 의해 일시 중지된 상태의 클러스터 노드가 다시 시작될 때 사용할 장애 복구 유형을 지정합니다.
CLUSTER_NODE_STATE CLUSTER_NODE_STATE(clusapi.h)는 클러스터 노드의 상태를 설명합니다.
CLUSTER_NODE_STATUS CLUSTER_NODE_STATUS(clusapi.h)는 클러스터 노드의 상태를 설명합니다.
CLUSTER_NOTIFICATIONS_VERSION CLUSTER_NOTIFICATIONS_VERSION(clusapi.h)는 다양한 버전의 클러스터 알림 열거형을 정의합니다.
CLUSTER_OBJECT_TYPE CLUSTER_OBJECT_TYPE(clusapi.h)는 알림이 요청되거나 생성되는 개체의 형식을 정의합니다.
CLUSTER_PROPERTY_FORMAT CLUSTER_PROPERTY_FORMAT(clusapi.h)는 속성 목록에서 속성 값의 데이터 형식을 지정합니다.
CLUSTER_PROPERTY_SYNTAX CLUSTER_PROPERTY_SYNTAX(clusapi.h)는 속성 목록의 구문 구조에 대해 가능한 값을 제공합니다.
CLUSTER_PROPERTY_TYPE CLUSTER_PROPERTY_TYPE(clusapi.h)는 클러스터 속성 목록에서 지원하는 속성 형식을 정의합니다.
CLUSTER_QUORUM_TYPE CLUSTER_QUORUM_TYPE(clusapi.h)는 만들 쿼럼 리소스의 유형입니다.
CLUSTER_QUORUM_VALUE CLUSTER_QUORUM_VALUE(clusapi.h)는 CLUSCTL_CLUSTER_CHECK_VOTER_DOWN 또는 CLUSCTL_CLUSTER_CHECK_VOTER_EVICT 제어 코드를 사용하여 ClusterControl 함수에서 반환한 값을 열거합니다.
CLUSTER_REG_COMMAND CLUSTER_REG_COMMAND(clusapi.h)는 클러스터 조인을 시도할 때 로컬 노드가 수행할 수 있는 클러스터 레지스트리 명령을 열거합니다.

CLUSTER_RESOURCE_CLASS CLUSTER_RESOURCE_CLASS(clusapi.h)는 리소스의 클래스를 정의합니다.
CLUSTER_RESOURCE_CREATE_FLAGS CLUSTER_RESOURCE_CREATE_FLAGS(clusapi.h)는 지정된 리소스를 할당할 리소스 모니터를 결정합니다.
CLUSTER_RESOURCE_EMBEDDED_FAILURE_ACTION CLUSTER_RESOURCE_EMBEDDED_FAILURE_ACTION(clusapi.h)는 리소스에 포함된 오류가 있을 때 수행할 수 있는 다양한 작업을 지정합니다.
CLUSTER_RESOURCE_ENUM CLUSTER_RESOURCE_ENUM(clusapi.h)는 ClusterResourceEnum 또는 ClusterResourceOpenEnum 함수에 의해 열거되는 클러스터 개체의 형식을 설명합니다.
CLUSTER_RESOURCE_RESTART_ACTION CLUSTER_RESOURCE_RESTART_ACTION(clusapi.h)는 RestartAction 리소스 공용 속성에서 리소스가 실패할 경우 클러스터 서비스에서 수행할 작업을 지정하는 데 사용됩니다.
CLUSTER_RESOURCE_STATE CLUSTER_RESOURCE_STATE(clusapi.h) 열거형은 리소스의 작동 상태를 설명합니다.
CLUSTER_RESOURCE_STATE_CHANGE_REASON CLUSTER_RESOURCE_STATE_CHANGE_REASON(clusapi.h)는 제어 코드에서 리소스 상태 변경 이유를 설명하는 데 사용됩니다.
CLUSTER_RESOURCE_TYPE_ENUM CLUSTER_RESOURCE_TYPE_ENUM ClusterResourceTypeEnum 및 ClusterResourceTypeOpenEnum 함수에 의해 열거되는 클러스터 개체의 형식을 설명합니다.
CLUSTER_SETUP_PHASE ClusterSetupProgressCallback 함수에서 클러스터 설치 프로세스의 현재 단계를 식별하는 데 사용됩니다.
CLUSTER_SETUP_PHASE_SEVERITY 클러스터 설치 프로세스의 현재 단계의 심각도를 설명합니다.

CLUSTER_SETUP_PHASE_TYPE 클러스터 설치 프로세스의 진행 상황을 설명합니다.
CLUSTER_SHARED_VOLUME_BACKUP_STATE CSV 백업 상태를 설명합니다.
CLUSTER_SHARED_VOLUME_SNAPSHOT_STATE CLUSTER_SHARED_VOLUME_SNAPSHOT_STATE(clusapi.h)는 공유 볼륨에 대한 다양한 스냅샷 상태를 지정합니다.
CLUSTER_SHARED_VOLUME_STATE 클러스터 공유 볼륨의 상태를 정의합니다.
CLUSTER_UPGRADE_PHASE 클러스터에서 운영 체제의 롤링 업그레이드 상태를 설명합니다. 이 열거형은 ClusterUpgradeProgressCallback 콜백 함수에서 사용됩니다.
FILESHARE_CHANGE_ENUM 파일 공유 이벤트 알림 목록의 항목을 설명하기 위해 FILESHARE_CHANGE 구조에서 사용할 수 있는 변경 이벤트를 포함합니다.
MAINTENANCE_MODE_TYPE_ENUM 유지 관리를 위해 표시될 때 스토리지 클래스 리소스를 배치할 수 있는 가능한 상태를 정의합니다.
NODE_CLUSTER_STATE NODE_CLUSTER_STATE(clusapi.h)는 클러스터의 상태를 나타냅니다. GetNodeClusterState 함수는 이 열거형을 사용합니다.
PLACEMENT_OPTIONS 클러스터 배치 옵션을 정의합니다.
SR_DISK_REPLICATION_ELIGIBLE 클러스터 노드의 디스크가 복제에 적합하거나 부적격할 수 있는 다양한 이유를 지정합니다.
SR_REPLICATED_DISK_TYPE SR_RESOURCE_TYPE_REPLICATED_DISK 구조체에 대해 복제된 디스크 유형을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

cmdtree.h 헤더

아티클2023. 03. 09.

이 헤더는 인덱싱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 인덱싱 서비스

cmdtree.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICommandTree ICommandTree 인터페이스는 명령을 지원하는 공급자에 대한 선택 사항입니다. 쿼리 트리를 조작하는 메서드가 포함되어 있습니다. 명령 트리를 지원하는 공급자는 ICommandText 인터페이스를 통해 동일한 기능을 지정하는 것도 지원해야 합니다.
IQuery IQuery 인터페이스는 나중에 사용하도록 예약되어 있습니다.

구조체

DBBYGUID DBBYGUID 구조체는 노드에 대한 추가 정보를 제공합니다.
DBCOMMANDTREE DBCOMMANDTREE 구조는 이 참조의 데이터 조작 연산자 및 데이터 정의 연산자 섹션에 설명된 대로 OLE DB 명령 트리의 노드를 나타내는 데 사용되는 기본 데이터 구조입니다.
DBCONTENT DBCONTENT 구조체는 DBOP_content 연산자에 필요한 특정 정보를 나타냅니다.
DBCONTENTPROXIMITY DBCONTENTPROXIMITY 구조체는 DBOP_content_proximity 연산자에 필요한 특정 정보를 나타냅니다.

DBCONTENTSCOPE DBCONTENTSCOPE 구조체는 명령 트리에서 범위 인수를 전달하는 데 사용됩니다.
DBCONTENTTABLE DBCONTENTTABLE 구조체는 명령 트리의 컴퓨터 및 카탈로그 이름을 나타냅니다.
DBCONTENTVECTOR DBCONTENTVECTOR 구조체는 DBOP_content_vector_or 연산자에 필요한 특정 정보를 나타냅니다. (DBCONTENTVECTOR)
DBGROUPINFO DBCONTENTVECTOR 구조체는 DBOP_content_vector_or 연산자에 필요한 특정 정보를 나타냅니다. (DBGROUPINFO)
DBLIKE DBLIKE 구조는 DBOP_like 연산자에 필요한 특정 정보를 나타냅니다.
DBPARAMETER DBPARAMETER 구조체는 스칼라 매개 변수에 대한 값을 정의하는 데 사용됩니다.
DBPROBABILISTIC DBPROBABILISTIC 구조체는 나중에 사용할 수 있습니다.
DBRELEVANTDOCUMENT DBRELEVANTDOCUMENT 구조체는 나중에 사용할 수 있습니다.
DBSETFUNC DBSETFUNC 구조체는 선택 작업에 사용할 집계 함수를 지정합니다.
DBSORTINFO DBSORTINFO 구조체는 열이 정렬되는 순서(즉, 오름차순 또는 내림차순)를 저장합니다. 이 정보는 DBOP_sort_list_element 노드 내에 저장됩니다.
DBTEXT DBTEXT 구조는 DBOP_text_command 노드에서 사용됩니다.

열거형

DBCOMMANDOPENUM DBCOMMANDOPENUM 열거형 형식에는 명령 트리의 노드에 대해 가능한 명령 연산자 목록이 포함되어 있습니다.
DBCOMMANDREUSEENUM DBCOMMANDREUSEENUM 열거형 형식은 이전 명령의 상태를 유지할지 여부를 지정합니다.
DBVALUEKINDENUM DBVALUEKINDENUM 열거형 형식은 DBCOMMANDTREE 구조체 내의 공용 구조체 멤버의 형식을 나타내는 데 사용됩니다.

cmnquery.h 헤더

아티클2023. 08. 22.

이 헤더는 Active Directory Domain Services 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Active Directory Domain Services

cmnquery.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICommonQuery 시스템 제공 디렉터리 서비스 쿼리 대화 상자를 프로그래밍 방식으로 표시하는 데 사용됩니다.
IPersistQuery 영구 스토리지와 쿼리 매개 변수를 저장하고 검색하는 데 사용됩니다.
IQueryForm 양식 개체가 시스템에서 제공하는 디렉터리 서비스 쿼리 대화 상자에 양식과 페이지를 추가할 수 있도록 쿼리 폼 확장 개체에 의해 구현됩니다.

콜백 함수

LPCQADDFORMSPROC 쿼리 폼 확장에 의해 호출되어 쿼리 대화 상자에 폼을 추가합니다.
LPCQADDPAGESPROC 쿼리 폼 확장에 의해 호출되어 쿼리 대화 상자의 쿼리 폼에 페이지를 추가합니다.
LPCQPAGEPROC 쿼리 페이지에서 발생하는 이벤트의 쿼리 양식 확장에 알리기 위해 쿼리 대화 상자에서 호출됩니다.

구조체

CQFORM CQAddFormsProc 콜백 함수를 사용하여 쿼리 대화 상자에 추가된 쿼리 양식을 정의하는 데 사용됩니다.
CQPAGE CQAddPagesProc 콜백 함수를 사용하여 쿼리 대화 상자의 폼에 추가된 쿼리 페이지를 정의하는 데 사용됩니다.
OPENQUERYWINDOW ICommonQuery::OpenQueryWindow 메서드와 함께 디렉터리 서비스 쿼리 대화 상자를 초기화하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

codecapi.h 헤더

아티클2023. 03. 16.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Directshow
- Microsoft Media Foundation

codecapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

열거형

eAVAudioChannelConfig 오디오 비트 스트림의 오디오 채널에 대한 스피커 구성을 지정합니다. 이 열거형은 AVAudioChannelConfig 속성과 함께 사용됩니다.
eAVDDSurroundMode 오디오가 Dolby 서라운드로 인코딩되는지 여부를 지정합니다. 이 열거형은 AVDDSurroundMode 속성과 함께 사용됩니다.
eAVDecAACDownmixMode AAC 디코더가 표준 MPEG-2/MPEG-4 스테레오 다운믹스 수식을 사용하는지 여부를 지정합니다.
eAVDecAudioDualMono 입력 오디오 스트림이 스테레오인지 이중 모노인지를 지정합니다. 이 열거형은 AVDecAudioDualMono 속성과 함께 사용됩니다.
eAVDecAudioDualMonoReproMode 디코더가 이중 모노 오디오를 재현하는 방법을 지정합니다. 이 열거형은 AVDecAudioDualMonoReproMode 속성과 함께 사용됩니다.
eAVDecDDOperationalMode Dolby AC-3 오디오 스트림의 압축 제어 모드를 지정합니다. 이 열거형은 AVDecDDOperationalMode 속성과 함께 사용됩니다.
eAVDecDDStereoDownMixMode Dolby Digital 오디오 디코더의 스테레오 다운믹스 모드를 지정합니다.

eAVDecHEAACDynamicRangeControl AAC 디코더가 동적 범위 제어를 수행하는지 여부를 지정합니다.
eAVDecVideoInputScanType 디코딩된 비디오 스트림이 인터레이스되는 방법을 지정합니다. 이 열거형은 AVDecVideoInputScanType 속성과 함께 사용됩니다.
eAVDecVideoSoftwareDeinterlaceMode 비디오 디코더의 소프트웨어 디인터레이스 모드를 지정합니다. 이 열거형은 AVDecVideoSoftwareDeinterlaceMode 속성과 함께 사용됩니다.
eAVDecVideoSWPowerLevel 비디오 디코더의 절전 수준을 지정합니다.
eAVDSPLoudnessEqualization 오디오 디코더 또는 DSP(디지털 신호 프로세서)에서 음음 균등화를 사용할 수 있는지 여부를 지정합니다.
eAVDSPSpeakerFill 오디오 디코더 또는 DSP(디지털 신호 프로세서)에서 스피커 채우기를 사용할 수 있는지 여부를 지정합니다.
eAVEncAdaptiveMode 비디오 인코더에 대한 적응형 인코딩 모드를 지정합니다.
eAVEncAudioDualMono 2 채널 오디오가 스테레오 또는 듀얼 모노로 인코딩되는지 여부를 지정합니다. 이 열거형은 AVEncAudioDualMono 속성과 함께 사용됩니다.
eAVEncAudioInputContent 오디오 콘텐츠에 음악 또는 음성이 포함되어 있는지 여부를 지정합니다. 이 열거형은 AVEncAudioInputContent 속성과 함께 사용됩니다.
eAVEncCommonRateControlMode 인코더의 속도 제어 모드를 지정합니다. 이 열거형은 AVEncCommonRateControlMode 코덱 속성과 함께 사용됩니다.

eAVEncCommonStreamEndHandling 인코더가 스트림의 끝에 있는 그림의 부분 그룹(GOP)을 삭제할지 여부를 지정합니다. 이 열거형은 AVEncCommonStreamEndHandling 코덱 속성과 함께 사용됩니다.
eAVEncDDAtoDConverterType Dolby Digital 오디오 스트림에 대한 아날로그-디지털(A/D) 변환 유형을 지정합니다. 이 열거형은 AVEncDDAtoDConverterType 속성과 함께 사용됩니다.
eAVEncDDDynamicRangeCompressionControl Dolby Digital 오디오 스트림의 동적 범위 제어 프로필을 지정합니다. 이 열거형은 AVEncDDDynamicRangeCompressionControl 속성과 함께 사용됩니다.
eAVEncDDHeadphoneMode Dolby Digital 오디오 스트림의 헤드폰 모드를 지정합니다. 이 열거형은 AVEncDDHeadphoneMode 속성과 함께 사용됩니다.
eAVEncDDPreferredStereoDownMixMode Dolby Digital 오디오 스트림의 기본 스테레오 다운믹스 모드를 지정합니다. 이 열거형은 AVEncDDPreferredStereoDownMixMode 속성과 함께 사용됩니다.
eAVEncDDProductionRoomType Dolby Digital 오디오 스트림의 회의실 유형을 지정합니다. 이 열거형은 AVEncDDProductionRoomType 속성과 함께 사용됩니다.
eAVEncDDService Dolby Digital 오디오 스트림에 포함된 오디오 서비스를 지정합니다. 이 열거형은 AVEncDDService 속성과 함께 사용됩니다.
eAVEncDDSurroundExMode Dolby Digital Surround EX에서 Dolby Digital 오디오 스트림을 인코딩할지 여부를 지정합니다. 이 열거형은 AVEncDDSurroundExMode 속성과 함께 사용됩니다.
eAVEncH264PictureType 비디오 인코더에서 출력되는 그림의 형식을 지정합니다.
eAVEncH264VLevel H.264 인코딩 수준을 지정합니다.

eAVEncH264VProfile H.264 비디오 프로필을 지정합니다.
eAVEncInputVideoSystem 비디오 원본의 명목 범위를 지정합니다. 이 열거형은 AVEncInputVideoSystem 속성과 함께 사용됩니다.
eAVEncMPACodingMode MPEG 오디오 인코딩 모드를 지정합니다. 이 열거형은 AVEncMPACodingMode 속성과 함께 사용됩니다.
eAVEncMPAEmphasisType 디코딩할 때 사용해야 하는 강조 해제 필터의 형식을 지정합니다. 이 열거형은 AVEncMPAEmphasisType 속성과 함께 사용됩니다.
eAVEncMPALayer MPEG 오디오 계층을 지정합니다. 이 열거형은 AVEncMPALayer 속성과 함께 사용됩니다.
eAVEncMPVFrameFieldMode 인코더가 인코딩된 필드 또는 인코딩된 프레임을 생성하는지 여부를 지정합니다. 이 열거형은 AVEncMPVFrameFieldMode 속성과 함께 사용됩니다.
eAVEncMPVIntraVLCTable 엔트로피 코딩에 사용할 VLC(가변 길이 코딩) 테이블을 지정합니다. 이 열거형은 AVEncMPVIntraVLCTable 속성과 함께 사용됩니다.
eAVEncMPVLevel MPEG-2 프로필을 지정합니다. 이 열거형은 AVEncMPVLevel 속성과 함께 사용됩니다.
eAVEncMPVProfile MPEG-2 프로필을 지정합니다. 이 열거형은 AVEncMPVProfile 속성과 함께 사용됩니다.
eAVEncMPVQScaleType 분위수 배율 선형 또는 비선형인지 여부를 지정합니다. 이 열거형은 AVEncMPVQScaleType 속성과 함께 사용됩니다.
eAVEncMPVScanPattern macroblock 검사 패턴을 지정합니다. 이 열거형은 AVEncMPVScanPattern 속성과 함께 사용됩니다.

eAVEncMPVSceneDetection 인코더가 새 장면을 감지할 때 동작하는 방식을 지정합니다. 이 열거형은 AVEncMPVSceneDetection 속성과 함께 사용됩니다.
eAVEncMuxOutput 멀티플렉서에서 생성되는 출력 스트림의 형식을 지정합니다. 이 열거형은 AVEncMuxOutputStreamType 속성과 함께 사용됩니다.
eAVEncVideoChromaResolution 크로마 해상도를 지정합니다. 이 열거형은 AVEncVideoInputChromaResolution 및 AVEncVideoOutputChromaResolution 속성과 함께 사용됩니다.
eAVEncVideoChromaSubsampling chroma siting을 지정합니다. 크로마 앉는 것은 요로 샘플에 상대적인 크로마 견본의 위치를 정의합니다. 이 열거형은 AVEncVideoInputChromaSubsampling 및 AVEncVideoOutputChromaSubsampling 속성과 함께 사용됩니다.
eAVEncVideoColorLighting 비디오 원본을 보기 위한 조명 조건을 지정합니다. 이 열거형은 AVEncVideoInputColorLighting 및 AVEncVideoOutputColorLighting 속성과 함께 사용됩니다.
eAVEncVideoColorNominalRange 비디오 원본의 명목 범위를 지정합니다. 이 열거형은 AVEncVideoInputChromaSubsampling 및 AVEncVideoOutputChromaSubsampling 속성과 함께 사용됩니다.
eAVEncVideoColorPrimaries 비디오의 색 기본값을 지정합니다. 이 열거형은 AVEncVideoInputColorPrimaries 및 AVEncVideoOutputColorPrimaries 속성과 함께 사용됩니다.
eAVEncVideoColorTransferFunction R'G'B'에서 RGB로 변환 함수를 지정합니다. 이 열거형은 AVEncVideoInputColorTransferFunction 및 AVEncVideoOutputColorTransferFunction 속성과 함께 사용됩니다.
eAVEncVideoColorTransferMatrix Y'Cb'Cr' 색 공간에서 R'G'B' 색 공간으로의 변환 매트릭스를 지정합니다. 이 열거형은 AVEncVideoInputColorTransferMatrix 및 AVEncVideoOutputColorTransferMatrix 속성과 함께 사용됩니다.

eAVEncVideoContentType 비디오 콘텐츠의 형식을 지정합니다.
eAVEncVideoFilmContent 입력 비디오의 원본이 필름인지 비디오인지를 지정합니다. 이 열거형은 AVEncVideoSourceFilmContent 속성과 함께 사용됩니다.
eAVEncVideoOutputFrameRateConversion 출력 프레임 속도가 입력 프레임 속도와 일치하지 않는 경우 인코더가 프레임 속도를 변환할지 여부를 지정합니다. 이 열거형은 AVEncVideoOutputFrameRateConversion 속성과 함께 사용됩니다.
eAVEncVideoOutputScanType 인코더가 출력 비디오를 인터레이스하는 방법을 지정합니다. 이 열거형은 AVEncVideoOutputScanType 속성과 함께 사용됩니다.
eAVEncVideoSourceScanType 인코더의 입력 프레임이 프로그레시브 또는 인터레이스되는지 여부를 지정합니다. 이 열거형은 AVEncVideoForceSourceScanType 속성과 함께 사용됩니다.
eAVFastDecodeMode 비디오 디코딩 속도를 지정합니다. 이 열거형은 AVDecVideoFastDecodeMode 속성과 함께 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

comadmin.h 헤더

아티클2023. 08. 22.

이 헤더는 COM+(Component Services)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM+(구성 요소 서비스)

comadmin.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICatalogCollection COM+ 카탈로그의 모든 컬렉션을 나타냅니다. ICatalogCollection을 사용하면 컬렉션의 항목을 열거, 추가, 제거 및 검색하고 관련 컬렉션에 액세스할 수 있습니다.
ICatalogObject COM+ 카탈로그의 컬렉션에 있는 항목을 나타냅니다. ICatalogObject를 사용하면 카탈로그의 개체에 의해 노출되는 속성을 가져와서 배치할 수 있습니다.
ICOMAdminCatalog 프로그래밍 방식 COM+ 관리를 수행하고, 카탈로그의 컬렉션에 액세스하고, COM+ 애플리케이션 및 구성 요소를 설치하고, 서비스를 시작 및 중지하고, 원격 서버에 연결하는 세션을 시작합니다.
ICOMAdminCatalog2 ICOMAdminCatalog 인터페이스의 확장입니다.

열거형

COMAdminTxIsolationLevelOptions 트랜잭션에 사용할 격리 수준을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

combaseapi.h 헤더

아티클2023. 03. 16.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM+(구성 요소 서비스)
- COM(구성 요소 개체 모델)
- 구조적 저장소
- Windows 셸
- Windows 멀티미디어
- Windows 런타임 C++ 참조

combaseapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CLSIDFromProgID ProgID가 지정된 경우 레지스트리에서 CLSID를 조회합니다.
CLSIDFromProgIDEx COMClassStore 정책을 사용하도록 설정하면 CLSIDFromProgIDEx 함수(combaseapi.h)가 자동 설치를 트리거합니다.
CLSIDFromString StringFromCLSID 함수에서 생성된 문자열을 원래 CLSID로 다시 변환합니다.
CoAddRefServerProcess 프로세스별 전역 참조 수를 증분합니다.
CoAllowUnmarshalerCLSID 호출 프로세스에 대해서만 허용된 목록에 unmarshaler CLSID를 추가합니다.
CoCancelCall 지정된 스레드에서 보류 중인 아웃바운드 DCOM 메서드 호출의 취소를 요청합니다.
CoCopyProxy 지정된 프록시의 프라이빗 복사본을 만듭니다.

CoCreateFreeThreadedMarshaler 컨텍스트 종속 마샬링이 가능한 집계 가능한 개체를 만듭니다.
CoCreateGuid CLSID 및 인터페이스 식별자에 사용되는 고유한 128비트 정수인 GUID를 만듭니다.
CoCreateInstance 지정된 CLSID와 연결된 클래스의 초기화되지 않은 단일 개체를 만듭니다.
CoCreateInstanceEx 특정 컴퓨터에 특정 클래스의 인스턴스를 만듭니다.
CoCreateInstanceFromApp 앱 컨테이너 내에서 특정 컴퓨터에 특정 클래스의 인스턴스를 만듭니다.
CoDecodeProxy 프록시된 개체에 대한 인터페이스가 지정된 서버 프로세스에서 COM(구성 요소 개체 모델) 인터페이스의 구현을 찾습니다.
CoDecrementMTAUsage CoIncrementMTAUsage 함수에 대한 이전 호출에서 수행한 증분을 해제합니다.
CoDisableCallCancellation CoEnableCallCancellation에 대한 호출 작업을 실행 취소합니다. CoEnableCallCancellation에 대한 모든 호출이 CoDisableCallCancellation 호출에 의해 균형을 이루면 호출 스레드에서 동기 호출의 취소를 사용하지 않도록 설정합니다.
CoDisconnectContext 현재 컨텍스트의 개체를 가리키는 모든 인터페이스 포인터를 대신하여 유지 관리되는 모든 프록시 연결의 연결을 끊습니다.
CoDisconnectObject 지정된 개체를 가리키는 모든 인터페이스 포인터를 대신하여 유지 관리되는 모든 원격 프로세스 연결을 끊습니다.
CoEnableCallCancellation 호출 스레드에서 동기 호출을 취소할 수 있습니다.

CoFileTimeNow CoFileTimeNow 함수(combaseapi.h)는 현재 시간을 FILETIME 구조체로 반환합니다.
CoFreeUnusedLibraries DLL에 더 이상 인스턴스화된 COM 개체가 없기 때문에 더 이상 사용되지 않는 모든 DLL을 언로드합니다.
CoFreeUnusedLibrariesEx 더 이상 사용되지 않고 언로드 지연이 만료된 DLL을 언로드합니다.
CoGetApartmentType 현재 아파트 유형 및 형식 한정자를 반환합니다.
CoGetCallContext 현재 스레드에서 현재 호출의 컨텍스트를 검색합니다.
CoGetCallerTID 호출자 스레드의 아파트 ID를 포함하는 DWORD에 대한 포인터를 반환합니다.
CoGetCancelObject 동일한 또는 다른 클라이언트 스레드에서 보류 중인 아웃바운드 COM 메서드 호출에 해당하는 cancel 개체에서 호출 제어 인터페이스(일반적으로 ICancelMethodCalls)에 대한 포인터를 가져옵니다.
CoGetClassObject 지정된 CLSID와 연결된 클래스 개체의 인터페이스에 대한 포인터를 제공합니다.
CoGetContextToken 현재 컨텍스트에 대한 IObjContext 구현에 대한 포인터를 반환합니다.
CoGetCurrentLogicalThreadId 현재 실제 스레드의 논리적 스레드 식별자를 반환합니다.
CoGetCurrentProcess 현재 스레드에 고유한 값을 반환합니다. CoGetCurrentProcess를 사용하여 스레드 ID 재사용 문제를 방지할 수 있습니다.

CoGetDefaultContext 지정된 아파트의 기본 컨텍스트에 대한 참조를 검색합니다.
CoGetInterfaceAndReleaseStream 인터페이스 포인터가 포함된 버퍼를 분리하고 인터페이스 포인터가 다른 스레드에서 호출 스레드로 마샬링되면 스트림을 해제합니다.
CoGetMalloc 애플리케이션이 해당 메서드를 호출하여 메모리를 관리할 수 있도록 기본 OLE 작업 메모리 할당자(IMalloc 인터페이스의 시스템 구현을 지원)에 대한 포인터를 검색합니다.
CoGetMarshalSizeMax 지정된 인터페이스 포인터를 지정된 개체로 마샬링하는 데 필요한 바이트 수의 상한을 반환합니다.
CoGetObjectContext 현재 개체의 컨텍스트를 반환합니다.
CoGetPSClsid 지정된 인터페이스에 대한 프록시 및 스텁을 구현하는 DLL의 CLSID를 반환합니다.
CoGetStandardMarshal 호출자에 따라 클라이언트 프로세스 또는 서버 프로세스에서 기본 또는 표준 마샬링 개체를 만들고 해당 개체의 IMarshal 구현에 대한 포인터를 반환합니다.
CoGetStdMarshalEx 경량 클라이언트 쪽 처리기에 사용할 집계된 표준 마샬러를 만듭니다.
CoGetTreatAsClass 지정된 개체를 에뮬레이트할 수 있는 개체의 CLSID를 반환합니다.
CoImpersonateClient 서버가 호출 기간 동안 현재 호출의 클라이언트를 가장할 수 있도록 합니다.
CoIncrementMTAUsage MTA 스레드가 실행되고 있지 않은 경우 MTA 지원을 활성으로 유지합니다.

CoInitializeEx 호출 스레드에서 사용할 COM 라이브러리를 초기화하고, 스레드의 동시성 모델을 설정하고, 필요한 경우 스레드에 대한 새 아파트를 만듭니다.
CoInitializeSecurity 보안을 등록하고 프로세스의 기본 보안 값을 설정합니다.
CoInvalidateRemoteMachineBindings 서비스 제어 관리자에게 지정된 컴퓨터에 대해 캐시된 RPC 바인딩 핸들을 플러시하도록 지시합니다.
CoIsHandlerConnected 원격 개체가 해당 in-process 개체에 연결되어 있는지 여부를 확인합니다.
CoLockObjectExternal 개체가 메모리에 유지되도록 개체를 잠그거나 이러한 잠금을 해제하기 위해 호출됩니다.
CoMarshalHresult COUnmarshalHresult 함수를 사용하여 HRESULT를 지정된 스트림으로 마샬링합니다.
CoMarshalInterface 일부 클라이언트 프로세스에서 프록시 개체를 초기화하는 데 필요한 데이터를 스트림에 씁니다.
CoMarshalInterThreadInterfaceInStream 동일한 프로세스에서 한 스레드에서 다른 스레드로 인터페이스 포인터를 마샬링합니다.
CoQueryAuthenticationServices CoInitializeSecurity라는 프로세스가 있을 때 등록된 인증 서비스 목록을 검색합니다.
CoQueryClientBlanket 서버에서 호출하여 현재 스레드에서 실행되는 메서드를 호출한 클라이언트에 대해 알아봅니다.
CoQueryProxyBlanket 클라이언트가 지정된 프록시에서 호출하는 데 사용하는 인증 정보를 검색합니다.
CoRegisterActivationFilter 프로세스 전체 필터를 등록하여 활성화 요청을 처리합니다.

CoRegisterClassObject EXE 클래스 개체를 OLE에 등록하여 다른 애플리케이션이 연결할 수 있도록 합니다.
CoRegisterDeviceCatalog 마샬링 코드가 해당 인터페이스를 마샬링할 수 있도록 다운로드한 DLL이 실행 중인 프로세스 내에 디바이스 카탈로그 인터페이스를 등록할 수 있도록 합니다.
CoRegisterPSClsid 마샬링 코드가 해당 인터페이스를 마샬링할 수 있도록 다운로드한 DLL이 실행 중인 프로세스 내에 사용자 지정 인터페이스를 등록할 수 있도록 합니다.
CoRegisterSurrogate 해당 ISurrogate 인터페이스 포인터를 통해 서로게이트 프로세스를 등록합니다.
CoReleaseMarshalData 이전에 마샬링된 데이터 패킷을 제거합니다.
CoReleaseServerProcess 전역 프로세스별 참조 수를 감소합니다.
CoResumeClassObjects 등록된 모든 클래스에 대해 SCM에 알리기 위해 여러 클래스 개체를 등록하고 해당 클래스 개체에 대한 활성화 요청을 허용하는 서버에서 호출됩니다.
CoRevertToSelf 실행 스레드에 대한 인증 정보를 복원합니다.
CoRevokeClassObject 이전에 CoRegisterClassObject 함수에 등록된 클래스 개체를 더 이상 사용할 수 없음을 OLE에 알릴 수 있습니다.
CoRevokeDeviceCatalog CoRegisterDeviceCatalog에 대한 이전 호출에서 수행한 디바이스 카탈로그 등록을 취소합니다.
CoSetCancelObject 현재 스레드에서 후속 취소 작업 중에 사용할 취소 개체를 설정(등록) 또는 다시 설정(등록 취소)합니다.

CoSetProxyBlanket 지정된 프록시에서 호출하는 데 사용할 인증 정보를 설정합니다.
CoSuspendClassObjects 프로세스 내에 등록된 모든 클래스 개체에서 SCM의 새 활성화 요청을 방지합니다.
CoSwitchCallContext CoGetCallContext에서 사용하는 호출 컨텍스트 개체를 전환합니다.
CoTaskMemAlloc IMalloc::Alloc와 동일한 방식으로 작업 메모리 블록을 할당합니다.
CoTaskMemFree CoTaskMemAlloc 또는 CoTaskMemRealloc 함수에 대한 호출을 통해 이전에 할당된 작업 메모리 블록을 해제합니다.
CoTaskMemRealloc 이전에 할당된 작업 메모리 블록의 크기를 변경합니다.
CoTestCancel 서버에서 실행 중인 호출이 클라이언트에 의해 취소되었는지 여부를 확인합니다.
CoUninitialize 현재 스레드에서 COM 라이브러리를 닫고, 스레드에 의해 로드된 모든 DLL을 언로드하고, 스레드가 유지 관리하는 다른 모든 리소스를 해제하고, 스레드의 모든 RPC 연결을 강제로 닫습니다.
CoUnmarshalHresult 지정된 스트림에서 HRESULT 형식을 언마샬링합니다.
CoUnmarshalInterface CoMarshalInterface 함수에 대한 이전 호출에 의해 스트림에 기록된 데이터를 사용하여 새로 만든 프록시를 초기화하고 해당 프록시에 대한 인터페이스 포인터를 반환합니다.
CoWaitForMultipleHandles 지정된 핸들이 신호를 받을 때까지 또는 지정된 시간 제한 기간이 경과될 때까지 기다립니다.

CoWaitForMultipleObjects CoWaitForMultipleHandles를 대체합니다. 이 대체 API는 ASTA에서 지원되지 않는 CoWaitForMultipleHandles에 대한 옵션을 숨깁니다.
CreateStreamOnHGlobal HGLOBAL 메모리 핸들을 사용하여 스트림 콘텐츠를 저장하는 스트림 개체를 만듭니다.
DllCanUnloadNow 이 함수를 구현하는 DLL이 사용 중인지 여부를 확인합니다. 그렇지 않은 경우 호출자는 메모리에서 DLL을 언로드할 수 있습니다.
DllGetClassObject DLL 개체 처리기 또는 개체 애플리케이션에서 클래스 개체를 검색합니다.
FreePropVariantArray FreePropVariantArray 함수는 rgvars 배열의 각 PROPVARIANT 구조체에서 PropVariantClear를 호출하여 배열의 각 멤버에 대해 값을 0으로 만듭니다.
GetHGlobalFromStream GetHGlobalFromStream 함수는 CreateStreamOnHGlobal 함수를 호출하여 만든 스트림에 대한 전역 메모리 핸들을 검색합니다.
IID_PPV_ARGS 인터페이스 포인터를 검색하는 데 사용되며, 사용된 인터페이스 포인터의 형식에 따라 요청된 인터페이스의 IID 값을 자동으로 입력합니다. 이렇게 하면 컴파일 시간에 전달된 값의 형식을 확인하여 일반적인 코딩 오류가 발생하지 않습니다.
IIDFromString StringFromIID 함수에서 생성된 문자열을 원래 IID(인터페이스 식별자)로 다시 변환합니다.
ProgIDFromCLSID 지정된 CLSID에 대한 ProgID를 검색합니다.
PropVariantClear 지정된 PROPVARIANT 구조에서 해제할 수 있는 모든 요소를 해제합니다.
PropVariantCopy PropVariantCopy 함수는 한 PROPVARIANT 구조체의 내용을 다른 구조체에 복사합니다.

RoGetAgileReference 지정된 인터페이스에서 지정한 개체에 대한 Agile 참조를 만듭니다.
StringFromCLSID CLSID를 인쇄 가능한 문자 문자열로 변환합니다. CLSID는 항상 다른 문자열로 변환됩니다.
StringFromGUID2 GUID(Globally Unique Identifier)를 인쇄 가능한 문자 문자열로 변환합니다.
StringFromIID 인터페이스 식별자를 인쇄 가능한 문자 문자열로 변환합니다.

구조체

ServerInformation 서버 프로세스에서 COM(구성 요소 개체 모델) 인터페이스의 구현을 나타냅니다.

열거형

AgileReferenceOptions RoGetAgileReference 함수에 대한 옵션을 지정합니다.
COWAIT_FLAGS CoWaitForMultipleHandles 함수의 동작을 지정합니다.
CWMO_FLAGS CoWaitForMultipleObjects 함수에 대한 플래그를 제공합니다.
REGCLS 클래스 개체에 대한 연결 유형을 제어합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

comcat.h 헤더

아티클2023. 08. 22.

이 헤더는 COM(구성 요소 개체 모델)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)

comcat.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICatInformation 특정 클래스에서 구현되거나 필요한 범주에 대한 정보와 지정된 컴퓨터에 등록된 범주에 대한 정보를 가져옵니다.
ICatRegister 레지스트리에서 구성 요소 범주 정보를 등록 및 등록 취소하는 메서드를 제공합니다. 여기에는 사람이 읽을 수 있는 범주 이름과 지정된 구성 요소 또는 클래스에서 구현/필요한 범주가 모두 포함됩니다.
IEnumCATEGORYINFO 시스템에 등록된 구성 요소 범주를 열거합니다.
IEnumGUID 클라이언트가 COM 클래스에 대한 클래스 ID 컬렉션을 열거할 수 있도록 합니다.

구조체

CATEGORYINFO 구성 요소 범주를 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

coml2api.h 헤더

아티클2023. 08. 22.

이 헤더는 Structured Storage에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 구조적 저장소

coml2api.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CreateILockBytesOnHGlobal HGLOBAL 메모리 핸들을 사용하여 복합 파일의 메모리 내 스토리지용 바이트를 저장하는 바이트 배열 개체를 만듭니다.
FmtIdToPropStgName FMTID(속성 집합 형식 식별자)를 스토리지 또는 스트림 이름으로 변환합니다.
GetConvertStg GetConvertStg 함수는 지정된 스토리지 개체에 대한 convert 비트의 현재 값을 반환합니다.
GetHGlobalFromILockBytes GetHGlobalFromILockBytes 함수는 CreateILockBytesOnHGlobal 함수를 사용하여 만든 바이트 배열 개체에 대한 전역 메모리 핸들을 검색합니다.
PropStgNameToFmtId 속성 집합 스토리지 또는 스트림 이름을 해당 형식 식별자로 변환합니다.
ReadClassStg ReadClassStg 함수는 WriteClassStg 함수를 사용하여 스토리지 개체에 이전에 작성된 CLSID를 읽습니다.
ReadClassStm WriteClassStm 함수를 사용하여 스트림 개체에 이전에 작성된 CLSID를 읽습니다.
StgCreateDocfile

IStorage 인터페이스에 대한 COM 제공 복합 파일 구현을 사용하여 새 복합 파일 스토리지 개체를 만듭니다.
StgCreateDocfileOnILockBytes 호출자가 제공하는 바이트 배열 개체 위에 새 복합 파일 스토리지 개체를 만들고 엽니다.
StgCreatePropSetStg 지정된 스토리지 개체에서 속성 집합 스토리지 개체를 만듭니다.
StgCreatePropStg 지정된 스토리지 또는 스트림 개체에서 속성 집합을 만들고 엽니다.
StgCreateStorageEx IStorage 또는 IPropertySetStorage 인터페이스에 대해 제공된 구현을 사용하여 새 스토리지 개체를 만듭니다.
StgIsStorageFile StgIsStorageFile 함수는 특정 디스크 파일에 스토리지 개체가 포함되어 있는지 여부를 나타냅니다.
StgIsStorageILockBytes StgIsStorageILockBytes 함수는 지정된 바이트 배열에 스토리지 개체가 포함되어 있는지 여부를 나타냅니다.
StgOpenPropStg 지정된 스토리지 또는 스트림 개체에서 지정된 속성 집합을 엽니다.
StgOpenStorage 파일 시스템에서 기존 루트 스토리지 개체를 엽니다.
StgOpenStorageEx 파일 시스템에서 기존 루트 스토리지 개체를 엽니다. 이 함수를 사용하여 복합 파일 및 일반 파일을 엽니다.
StgOpenStorageOnILockBytes StgOpenStorageOnILockBytes 함수는 디스크 파일에 존재하지 않고 대신 호출자가 제공하는 기본 바이트 배열이 있는 기존 스토리지 개체를 엽니다.
StgSetTimes

StgSetTimes 함수는 기본 파일 시스템에서 지원하는 경우 표시된 파일의 생성, 액세스 및 수정 시간을 설정합니다.
WriteClassStg WriteClassStg 함수는 지정된 CLSID(클래스 식별자)를 스토리지 개체에 저장합니다.
WriteClassStm WriteClassStm 함수는 지정된 CLSID를 스트림에 저장합니다.

구조체

STGOPTIONS StgCreateStorageEx 및 StgOpenStorageEx 함수에서 섹터 크기와 같은 스토리지 개체의 기능을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

commctrl.h 헤더

아티클2023. 06. 08.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 키보드 및 마우스 입력
- Windows 셸
- Windows 컨트롤

commctrl.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

_TrackMouseEvent 마우스 포인터가 창을 벗어나거나 지정된 시간 동안 창 위에 있을 때 메시지를 게시합니다. 이 함수는 TrackMouseEvent가 있으면 호출하고 그렇지 않으면 에뮬레이트합니다.
Animate_Close AVI 클립을 닫습니다. 이 매크로를 사용하거나 NULL 매개 변수를 전달하여 ACM_OPEN 메시지를 명시적으로 보낼 수 있습니다.
Animate_Create 애니메이션 컨트롤을 만듭니다. Animate_Create는 CreateWindow 함수를 호출하여 애니메이션 컨트롤을 만듭니다.
Animate_IsPlaying Audio-Video 인터리브(AVI) 클립이 재생되고 있는지 확인합니다. 이 매크로를 사용하거나 ACM_ISPLAYING 메시지를 보낼 수 있습니다.
Animate_Open AVI 클립을 열고 애니메이션 컨트롤에 첫 번째 프레임을 표시합니다. 이 매크로를 사용하거나 ACM_OPEN 메시지를 명시적으로 보낼 수 있습니다.
Animate_OpenEx 지정된 모듈의 리소스에서 AVI 클립을 열고 애니메이션 컨트롤에 첫 번째 프레임을 표시합니다. 이 매크로를 사용하거나 ACM_OPEN 메시지를 명시적으로 보낼 수 있습니다.

Animate_Play 애니메이션 컨트롤에서 AVI 클립을 재생합니다. 스레드가 계속 실행되는 동안 컨트롤은 백그라운드에서 클립을 재생합니다. 이 매크로를 사용하거나 ACM_PLAY 메시지를 명시적으로 보낼 수 있습니다.
Animate_Seek 애니메이션 컨트롤을 지시하여 AVI 클립의 특정 프레임을 표시합니다. 컨트롤은 스레드가 계속 실행되는 동안 배경에 클립을 표시합니다. 이 매크로를 사용하거나 ACM_PLAY 메시지를 명시적으로 보낼 수 있습니다.
Animate_Stop 애니메이션 컨트롤에서 AVI 클립 재생을 중지합니다. 이 매크로를 사용하거나 ACM_STOP 메시지를 명시적으로 보낼 수 있습니다.
Button_GetIdealSize 이미지 목록이 있는 경우 텍스트와 이미지에 가장 잘 맞는 단추 크기를 가져옵니다. 이 매크로를 사용하거나 BCM_GETIDEALSIZE 메시지를 명시적으로 보낼 수 있습니다.
Button_GetImageList 단추 컨트롤에 설정된 이미지 목록을 설명하는 BUTTON_IMAGELIST 구조체를 가져옵니다. 이 매크로를 사용하거나 BCM_GETIMAGELIST 메시지를 명시적으로 보낼 수 있습니다.
Button_GetNote 명령 링크 단추와 연결된 메모의 텍스트를 가져옵니다. 이 매크로를 사용하거나 BCM_GETNOTE 메시지를 명시적으로 보낼 수 있습니다.
Button_GetNoteLength 명령 링크에 대한 설명에 표시될 수 있는 메모 텍스트의 길이를 가져옵니다. 이 매크로를 사용하거나 BCM_GETNOTELENGTH 메시지를 명시적으로 보냅니다.
Button_GetSplitInfo 지정된 분할 단추 컨트롤에 대한 정보를 가져옵니다. 이 매크로를 사용하거나 BCM_GETSPLITINFO 메시지를 명시적으로 보냅니다.
Button_GetTextMargin 단추 컨트롤에서 텍스트를 그리는 데 사용되는 여백을 가져옵니다. 이 매크로를 사용하거나 BCM_GETTEXTMARGIN 메시지를 명시적으로 보낼 수 있습니다.

Button_SetDropDownState BS_SPLITBUTTON 스타일로 지정된 단추의 드롭다운 상태를 설정합니다. 이 매크로를 사용하거나 BCM_SETDROPDOWNSTATE 메시지를 명시적으로 보냅니다.
Button_SetElevationRequiredState 관리자 권한의 아이콘을 표시하기 위해 지정된 단추 또는 명령 링크에 대한 상승 필수 상태를 설정합니다. 이 매크로를 사용하거나 BCM_SETSHIELD 메시지를 명시적으로 보냅니다.
Button_SetImageList 단추 컨트롤에 이미지 목록을 할당합니다. 이 매크로를 사용하거나 BCM_SETIMAGELIST 메시지를 명시적으로 보낼 수 있습니다.
Button_SetNote 지정된 명령 링크 단추와 관련된 메모의 텍스트를 설정합니다. 이 매크로를 사용하거나 BCM_SETNOTE 메시지를 명시적으로 보낼 수 있습니다.
Button_SetSplitInfo 지정된 분할 단추 컨트롤에 대한 정보를 설정합니다. 이 매크로를 사용하거나 BCM_SETSPLITINFO 메시지를 명시적으로 보냅니다.
Button_SetTextMargin 단추 컨트롤에서 텍스트를 그리기 위한 여백을 설정합니다. 이 매크로를 사용하거나 BCM_SETTEXTMARGIN 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_GetCueBannerText 콤보 상자의 편집 컨트롤에 표시되는 큐 배너 텍스트를 가져옵니다. 이 매크로를 사용하거나 CB_GETCUEBANNER 메시지를 명시적으로 보냅니다.
ComboBox_GetMinVisible 콤보 상자의 드롭다운 목록에 표시되는 항목의 최소 수를 가져옵니다.
ComboBox_SetCueBannerText 콤보 상자의 편집 컨트롤에 대해 표시되는 큐 배너 텍스트를 설정합니다.
ComboBox_SetMinVisible 콤보 상자의 드롭다운 목록에 표시되는 항목의 최소 수를 설정합니다.

CreateMappedBitmap 도구 모음에서 사용할 비트맵을 만듭니다.
CreateStatusWindowA 일반적으로 애플리케이션의 상태를 표시하는 데 사용되는 상태 창을 만듭니다. (ANSI)
CreateStatusWindowW 일반적으로 애플리케이션의 상태를 표시하는 데 사용되는 상태 창을 만듭니다. (유니코드)
CreateToolbarEx 도구 모음 창을 만들고 지정된 단추를 도구 모음에 추가합니다.
CreateUpDownControl 업다운 컨트롤을 만듭니다. 참고:_This 함수는 사용되지 않습니다. 16비트 함수이며 범위와 위치에 대해 32비트 값을 처리할 수 없습니다.
DateTime_CloseMonthCal DTP(날짜 및 시간 선택기) 컨트롤을 닫습니다. 이 매크로를 사용하거나 DTM_CLOSEMONTHCAL 메시지를 명시적으로 보냅니다.
DateTime_GetDateTimePickerInfo 지정된 DTP(날짜 및 시간 선택기) 컨트롤에 대한 정보를 가져옵니다.
DateTime_GetIdealSize 클리핑 없이 컨트롤을 표시하는 데 필요한 크기를 가져옵니다. 이 매크로를 사용하거나 DTM_GETIDEALSIZE 메시지를 명시적으로 보냅니다.
DateTime_GetMonthCal DTP(날짜 및 시간 선택기) 자식 월 달력 컨트롤에 대한 핸들을 가져옵니다. 이 매크로를 사용하거나 DTM_GETMONTHCAL 메시지를 명시적으로 보낼 수 있습니다.
DateTime_GetMonthCalColor DTP(날짜 및 시간 선택기) 컨트롤 내에서 월 달력의 지정된 부분에 대한 색을 가져옵니다. 이 매크로를 사용하거나 DTM_GETMCCOLOR 메시지를 명시적으로 보낼 수 있습니다.
DateTime_GetMonthCalFont DTP(날짜 및 시간 선택기) 컨트롤의 자식 월 달력 컨트롤이 현재 사용하는 글꼴을 가져옵니다. 이 매크로를 사용하거나 DTM_GETMCFONT 메시지를 명시적으로 보낼 수 있습니다.

DateTime_GetMonthCalStyle 지정된 DTP(날짜 및 시간 선택기) 컨트롤의 스타일을 가져옵니다. 이 매크로를 사용하거나 DTM_GETMCSTYLE 메시지를 명시적으로 보냅니다.
DateTime_GetRange DTP(날짜 및 시간 선택기) 컨트롤에 허용되는 현재 최소 및 최대 시스템 시간을 가져옵니다. 이 매크로를 사용하거나 DTM_GETRANGE 메시지를 명시적으로 보낼 수 있습니다.
DateTime_GetSystemtime DTP(날짜 및 시간 선택기) 컨트롤에서 현재 선택된 시간을 가져와 지정된 SYSTEMTIME 구조체에 배치합니다. 이 매크로를 사용하거나 DTM_GETSYSTEMTIME 메시지를 명시적으로 보낼 수 있습니다.
DateTime_SetFormat 지정된 형식 문자열을 기반으로 DTP(날짜 및 시간 선택기) 컨트롤의 표시를 설정합니다. 이 매크로를 사용하거나 DTM_SETFORMAT 메시지를 명시적으로 보낼 수 있습니다.
DateTime_SetMonthCalColor DTP(날짜 및 시간 선택기) 컨트롤 내에서 월 달력의 지정된 부분에 대한 색을 설정합니다. 이 매크로를 사용하거나 DTM_SETMCCOLOR 메시지를 명시적으로 보낼 수 있습니다.
DateTime_SetMonthCalFont DTP(날짜 및 시간 선택기) 컨트롤의 자식 월 달력 컨트롤에서 사용할 글꼴을 설정합니다. 이 매크로를 사용하거나 DTM_SETMCFONT 메시지를 명시적으로 보낼 수 있습니다.
DateTime_SetMonthCalStyle 지정된 DTP(날짜 및 시간 선택기) 컨트롤의 스타일을 설정합니다. 이 매크로를 사용하거나 DTM_SETMCSTYLE 메시지를 명시적으로 보냅니다.
DateTime_SetRange DTP(날짜 및 시간 선택기) 컨트롤에 허용되는 최소 및 최대 시스템 시간을 설정합니다. 이 매크로를 사용하거나 DTM_SETRANGE 메시지를 명시적으로 보낼 수 있습니다.
DateTime_SetSystemtime DTP(날짜 및 시간 선택기) 컨트롤을 지정된 날짜 및 시간으로 설정합니다. 이 매크로를 사용하거나 DTM_SETSYSTEMTIME 메시지를 명시적으로 보낼 수 있습니다.

DefSubclassProc 창의 하위 클래스 체인에서 다음 처리기를 호출합니다. 하위 클래스 체인의 마지막 처리기는 창에 대한 원래 창 프로시저를 호출합니다.
DrawInsert 지정한 끌기 목록 상자의 부모 창에 삽입 아이콘을 그립니다.
DrawShadowText 그림자가 있는 텍스트를 그립니다.
DrawStatusTextA DrawStatusText 함수는 테두리가 있는 상태 창 스타일로 지정된 텍스트를 그립니다. (ANSI)
DrawStatusTextW DrawStatusText 함수는 테두리가 있는 상태 창 스타일로 지정된 텍스트를 그립니다. (유니코드)
Edit_EnableSearchWeb "Bing을 사용하여 검색..."을 사용하거나 사용하지 않도록 설정합니다. 편집 컨트롤의 상황에 맞는 메뉴 항목입니다. 이 매크로를 사용하거나 EM_ENABLESEARCHWEB 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetCaretIndex 지정된 편집 컨트롤에 대한 caret 위치의 문자 인덱스 를 가져옵니다. 이 매크로를 사용하거나 EM_GETCARETINDEX 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetCueBannerText 편집 컨트롤에서 텍스트 큐 또는 팁으로 표시되는 텍스트를 가져옵니다. 이 매크로를 사용하거나 EM_GETCUEBANNER 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetEndOfLine 편집 컨트롤의 내용에 사용되는 줄 문자의 끝을 가져옵니다. 이 매크로를 사용하거나 EM_GETENDOFLINE 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetExtendedStyle 지정된 편집 컨트롤에 현재 사용 중인 확장 스타일을 가져옵니다. 이 매크로를 사용하거나 EM_GETEXTENDEDSTYLE 메시지를 명시적으로 보낼 수 있습니다.

Edit_GetFileLine 지정된 파일(또는 논리적) 줄의 텍스트를 가져옵니다(텍스트 줄 바꿈 구분 기호는 무시됨). 이 매크로를 사용하거나 EM_GETFILELINE 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetFileLineCount 파일(또는 논리적) 줄 수를 가져옵니다(텍스트 줄 바꿈 구분 기호는 무시됨). 이 매크로를 사용하거나 EM_GETFILELINECOUNT 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetFileLineFromChar 지정된 문자 인덱스가 포함된 파일(또는 논리적) 텍스트 줄의 인덱스(텍스트 줄 바꿈 구분 기호는 무시됨)를 가져옵니다. 이 매크로를 사용하거나 EM_FILELINEFROMCHAR 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetFileLineIndex 지정된 표시줄에 따라 파일(또는 논리적) 텍스트 줄의 인덱스 를 가져옵니다. 이 매크로를 사용하거나 EM_FILELINEINDEX 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetFileLineLength 지정된 문자 인덱스에서 파일(또는 논리적) 텍스트 줄의 길이를 가져옵니다(텍스트 줄 바꿈 구분 기호는 무시됨). 이 매크로를 사용하거나 EM_FILELINELENGTH 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetHilite 이 매크로는 구현되지 않습니다. (Edit_GetHilite)
Edit_GetZoom 편집 컨트롤의 현재 확대/축소 비율을 가져옵니다(확대/축소 비율은 항상 1/64에서 64 사이임). 이 매크로를 사용하거나 EM_GETZOOM 메시지를 명시적으로 보낼 수 있습니다.
Edit_HideBalloonTip 편집 컨트롤과 연결된 풍선 팁을 숨깁니다. 이 매크로를 사용하거나 EM_HIDEBALLOONTIP 메시지를 명시적으로 보낼 수 있습니다.
Edit_NoSetFocus 한 줄 편집 컨트롤이 키보드 포커스를 받지 않도록 합니다. 이 매크로를 사용하거나 EM_NOSETFOCUS 메시지를 명시적으로 보낼 수 있습니다.

Edit_SearchWeb "Bing을 사용하여 검색..."을 호출합니다. 편집 컨트롤의 상황에 맞는 메뉴 항목입니다. 이 매크로를 사용하거나 EM_SEARCHWEB 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetCaretIndex caret를 찾을 문자 인덱스 를 설정합니다. 이 매크로를 사용하거나 EM_SETCARETINDEX 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetCueBannerText 편집 컨트롤에 대해 텍스트 큐 또는 팁으로 표시되는 텍스트를 설정합니다. 이 매크로를 사용하거나 EM_SETCUEBANNER 메시지를 명시적으로 보낼 수 있습니다. (Edit_SetCueBannerText)
Edit_SetCueBannerTextFocused 편집 컨트롤에 대해 텍스트 큐 또는 팁으로 표시되는 텍스트를 설정합니다. 이 매크로를 사용하거나 EM_SETCUEBANNER 메시지를 명시적으로 보낼 수 있습니다. (Edit_SetCueBannerTextFocused)
Edit_SetEndOfLine 편집 컨트롤의 내용에 사용되는 줄 문자의 끝을 설정합니다. 이 매크로를 사용하거나 EM_SETENDOFLINE 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetExtendedStyle 스타일 마스크를 사용하여 편집 컨트롤의 확장 스타일을 설정합니다. 이 매크로를 사용하거나 EM_SETEXTENDEDSTYLE 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetHilite 이 매크로는 구현되지 않습니다. (Edit_SetHilite)
Edit_SetZoom 편집 컨트롤의 현재 확대/축소 비율을 설정합니다(확대/축소 비율은 항상 1/64에서 64 사이임). 이 매크로를 사용하거나 EM_SETZOOM 메시지를 명시적으로 보낼 수 있습니다.
Edit_ShowBalloonTip 편집 컨트롤과 연결된 풍선 팁을 표시합니다. 이 매크로를 사용하거나 EM_SHOWBALLOONTIP 메시지를 명시적으로 보낼 수 있습니다.
Edit_TakeFocus 한 줄 편집 컨트롤을 강제 적용하여 키보드 포커스를 받습니다. 이 매크로를 사용하거나 EM_TAKEFOCUS 메시지를 명시적으로 보낼 수 있습니다.

FIRST_IPADDRESS IPM_GETADDRESS 메시지와 함께 검색된 압축된 IP 주소에서 필드 0 값을 추출합니다.
FlatSB_EnableScrollBar 플랫 스크롤 막대 방향 단추 중 하나 또는 둘 모두를 사용하거나 사용하지 않도록 설정합니다. 창에 대해 플랫 스크롤 막대가 초기화되지 않은 경우 이 함수는 표준 EnableScrollBar 함수를 호출합니다.
FlatSB_GetScrollInfo 플랫 스크롤 막대에 대한 정보를 가져옵니다. 창에 대해 플랫 스크롤 막대가 초기화되지 않은 경우 이 함수는 표준 GetScrollInfo 함수를 호출합니다.
FlatSB_GetScrollPos 플랫 스크롤 막대에서 엄지 위치를 가져옵니다. 창에 대해 플랫 스크롤 막대가 초기화되지 않은 경우 이 함수는 표준 GetScrollPos 함수를 호출합니다.
FlatSB_GetScrollProp 플랫 스크롤 막대의 속성을 가져옵니다. 이 함수는 이 창에 대해 InitializeFlatSB가 호출되었는지 확인하는 데에도 사용할 수 있습니다.
FlatSB_GetScrollPropPtr 플랫 스크롤 막대의 속성을 가져옵니다.
FlatSB_GetScrollRange 플랫 스크롤 막대의 스크롤 범위를 가져옵니다. 창에 대해 플랫 스크롤 막대가 초기화되지 않은 경우 이 함수는 표준 GetScrollRange 함수를 호출합니다.
FlatSB_SetScrollInfo 플랫 스크롤 막대에 대한 정보를 설정합니다. 창에 대해 플랫 스크롤 막대가 초기화되지 않은 경우 이 함수는 표준 SetScrollInfo 함수를 호출합니다.
FlatSB_SetScrollPos 플랫 스크롤 막대에서 엄지의 현재 위치를 설정합니다. 창에 대해 플랫 스크롤 막대가 초기화되지 않은 경우 이 함수는 표준 SetScrollPos 함수를 호출합니다.
FlatSB_SetScrollProp 플랫 스크롤 막대의 속성을 설정합니다.

FlatSB_SetScrollRange 플랫 스크롤 막대의 스크롤 범위를 설정합니다. 창에 대해 플랫 스크롤 막대가 초기화되지 않은 경우 이 함수는 표준 SetScrollRange 함수를 호출합니다.
FlatSB_ShowScrollBar 플랫 스크롤 막대를 표시하거나 숨깁니다. 창에 대해 플랫 스크롤 막대가 초기화되지 않은 경우 이 함수는 표준 ShowScrollBar 함수를 호출합니다.
FORWARD_WM_NOTIFY WM_NOTIFY 메시지를 보내거나 게시합니다.
FOURTH_IPADDRESS IPM_GETADDRESS 메시지와 함께 검색된 압축된 IP 주소에서 필드 3 값을 추출합니다.
GetEffectiveClientRect 지정된 모든 컨트롤을 포함하는 클라이언트 영역의 사각형 크기를 계산합니다.
GetMUILanguage 특정 프로세스에 대한 공용 컨트롤에서 현재 사용 중인 언어를 가져옵니다.
GetWindowSubclass 지정된 창 하위 클래스 콜백에 대한 참조 데이터를 검색합니다.
HANDLE_WM_NOTIFY WM_NOTIFY 메시지를 처리하는 함수를 호출합니다.
Header_ClearAllFilters 지정된 헤더 컨트롤에 대한 모든 필터를 지웁니다. 이 매크로를 사용하거나 HDM_CLEARFILTER 메시지를 명시적으로 보낼 수 있습니다.
Header_ClearFilter 지정된 헤더 컨트롤에 대한 필터를 지웁니다. 이 매크로를 사용하거나 HDM_CLEARFILTER 메시지를 명시적으로 보낼 수 있습니다.
Header_CreateDragImage 기존 헤더 컨트롤 내에서 항목 이미지의 투명 버전을 만듭니다. 이 매크로를 사용하거나 HDM_CREATEDRAGIMAGE 메시지를 명시적으로 보낼 수 있습니다.

Header_DeleteItem 헤더 컨트롤에서 항목을 삭제합니다. 이 매크로를 사용하거나 HDM_DELETEITEM 메시지를 명시적으로 보낼 수 있습니다.
Header_EditFilter 필터 단추에 포커스가 있는 경우 입력 포커스를 편집 상자로 이동합니다.
Header_GetBitmapMargin 기존 헤더 컨트롤에서 비트맵의 여백 너비(픽셀)를 가져옵니다. 이 매크로를 사용하거나 HDM_GETBITMAPMARGIN 메시지를 명시적으로 보낼 수 있습니다.
Header_GetFocusedItem 포커스가 있는 헤더 컨트롤의 항목을 가져옵니다. 이 매크로를 사용하거나 HDM_GETFOCUSEDITEM 메시지를 명시적으로 보냅니다.
Header_GetImageList 기존 헤더 컨트롤에 대해 설정된 이미지 목록에 대한 핸들을 가져옵니다. 이 매크로를 사용하거나 HDM_GETIMAGELIST 메시지를 명시적으로 보낼 수 있습니다.
Header_GetItem 헤더 컨트롤의 항목에 대한 정보를 가져옵니다. 이 매크로를 사용하거나 HDM_GETITEM 메시지를 명시적으로 보낼 수 있습니다.
Header_GetItemCount 헤더 컨트롤의 항목 수를 가져옵니다. 이 매크로를 사용하거나 HDM_GETITEMCOUNT 메시지를 명시적으로 보낼 수 있습니다.
Header_GetItemDropDownRect 헤더 컨트롤의 지정된 항목에 대한 드롭다운 단추의 좌표를 가져옵니다. 헤더 컨트롤은 HDF_SPLITBUTTON 형식이어야 합니다. 이 매크로를 사용하거나 HDM_GETITEMDROPDOWNRECT 메시지를 명시적으로 보냅니다.
Header_GetItemRect 헤더 컨트롤의 지정된 항목에 대한 경계 사각형을 가져옵니다. 이 매크로를 사용하거나 HDM_GETIMRECT 메시지를 명시적으로 보낼 수 있습니다.
Header_GetOrderArray 헤더 컨트롤에 있는 항목의 현재 왼쪽에서 오른쪽 순서를 가져옵니다. 이 매크로를 사용하거나 HDM_GETORDERARRAY 메시지를 명시적으로 보낼 수 있습니다.

Header_GetOverflowRect 지정된 헤더 컨트롤에 대한 드롭다운 오버플로 영역의 좌표를 가져옵니다. 헤더 컨트롤은 HDF_SPLITBUTTON 형식이어야 합니다. 이 매크로를 사용하거나 HDM_GETOVERFLOWRECT 메시지를 명시적으로 보냅니다.
Header_GetStateImageList 기존 헤더 컨트롤 상태에 대해 설정된 이미지 목록에 대한 핸들을 가져옵니다.
Header_GetUnicodeFormat 컨트롤의 유니코드 문자 형식 플래그를 가져옵니다. 이 매크로를 사용하거나 HDM_GETUNICODEFORMAT 메시지를 명시적으로 보낼 수 있습니다.
Header_InsertItem 헤더 컨트롤에 새 항목을 삽입합니다. 이 매크로를 사용하거나 HDM_INSERTITEM 메시지를 명시적으로 보낼 수 있습니다.
Header_Layout 부모 창 내에서 헤더 컨트롤의 올바른 크기와 위치를 검색합니다. 이 매크로를 사용하거나 HDM_LAYOUT 메시지를 명시적으로 보낼 수 있습니다.
Header_OrderToIndex 헤더 컨트롤의 순서에 따라 항목의 인덱스 값을 검색합니다. 이 매크로를 사용하거나 HDM_ORDERTOINDEX 메시지를 명시적으로 보낼 수 있습니다.
Header_SetBitmapMargin 기존 헤더 컨트롤에서 비트맵의 여백 너비를 설정합니다. 이 매크로를 사용하거나 HDM_SETBITMAPMARGIN 메시지를 명시적으로 보낼 수 있습니다.
Header_SetFilterChangeTimeout 필터 특성에서 변경이 발생하는 시간과 HDN_FILTERCHANGE 알림 게시 사이의 시간 제한 간격을 설정합니다. 이 매크로를 사용하거나 HDM_SETFILTERCHANGETIMEOUT 메시지를 명시적으로 보낼 수 있습니다.
Header_SetFocusedItem 헤더 컨트롤의 지정된 항목에 포커스를 설정합니다. 이 매크로를 사용하거나 HDM_SETFOCUSEDITEM 메시지를 명시적으로 보냅니다.

Header_SetHotDivider 외부 끌어서 놓기 작업의 대상을 나타내기 위해 헤더 항목 사이의 구분선 색을 변경합니다. 이 매크로를 사용하거나 HDM_SETHOTDIVIDER 메시지를 명시적으로 보낼 수 있습니다.
Header_SetImageList 기존 헤더 컨트롤에 이미지 목록을 할당합니다. 이 매크로를 사용하거나 HDM_SETIMAGELIST 메시지를 명시적으로 보낼 수 있습니다.
Header_SetItem 헤더 컨트롤에서 지정된 항목의 특성을 설정합니다. 이 매크로를 사용하거나 HDM_SETITEM 메시지를 명시적으로 보낼 수 있습니다.
Header_SetOrderArray 헤더 항목의 왼쪽에서 오른쪽 순서를 설정합니다. 이 매크로를 사용하거나 HDM_SETORDERARRAY 메시지를 명시적으로 보낼 수 있습니다.
Header_SetStateImageList 기존 헤더 컨트롤 상태에 이미지 목록을 할당합니다.
Header_SetUnicodeFormat 컨트롤의 유니코드 문자 형식 플래그를 설정합니다.
HIMAGELIST_QueryInterface 이미지 목록의 HIMAGELIST 핸들에 해당하는 IImageList 또는 IImageList2 개체에 대한 포인터를 검색합니다.
ImageList_Add 이미지 목록에 하나의 이미지 또는 여러 이미지를 추가합니다. (ImageList_Add)
ImageList_AddIcon 이미지 목록에 아이콘 또는 커서를 추가합니다. ImageList_AddIcon 함수는 ImageList_ReplaceIcon 함수를 호출합니다.
ImageList_AddMasked 이미지 목록에 하나의 이미지 또는 여러 이미지를 추가하여 지정된 비트맵에서 마스크를 생성합니다. (ImageList_AddMasked)

ImageList_BeginDrag 이미지 끌기를 시작합니다. (ImageList_BeginDrag)
ImageList_Copy 지정된 이미지 목록 내의 이미지를 복사합니다.
ImageList_Create 새 이미지 목록을 만듭니다.
ImageList_Destroy 이미지 목록을 제거합니다.
ImageList_DragEnter 창 내의 지정된 위치에 끌기 이미지를 표시합니다.
ImageList_DragLeave 지정된 창의 잠금을 해제하고 끌기 이미지를 숨겨 창을 업데이트할 수 있습니다.
ImageList_DragMove 끌어서 놓기 작업 중에 끌기 이미지를 이동합니다. 이 함수는 일반적으로 WM_MOUSEMOVE 메시지에 대한 응답으로 호출됩니다. (ImageList_DragMove)
ImageList_DragShowNolock 끌기 이미지를 표시하거나 숨깁니다. (ImageList_DragShowNolock)
ImageList_Draw 지정된 디바이스 컨텍스트에서 이미지 목록 항목을 그립니다. (ImageList_Draw)
ImageList_DrawEx 지정된 디바이스 컨텍스트에서 이미지 목록 항목을 그립니다. 이 함수는 지정된 그리기 스타일을 사용하고 이미지를 지정된 색과 혼합합니다.
ImageList_DrawIndirect IMAGELISTDRAWPARAMS 구조체를 기반으로 이미지 목록 이미지를 그립니다.
ImageList_Duplicate 기존 이미지 목록의 복제본을 만듭니다.

ImageList_EndDrag 끌기 작업을 종료합니다. (ImageList_EndDrag)
ImageList_ExtractIcon ImageList_GetIcon 함수를 호출하여 이미지 목록의 이미지 및 마스크를 기반으로 아이콘 또는 커서를 만듭니다.
ImageList_GetBkColor 이미지 목록의 현재 배경색을 검색합니다.
ImageList_GetDragImage 끌기 이미지에 사용되는 임시 이미지 목록을 검색합니다. 또한 이 함수는 현재 끌기 위치와 끌기 위치를 기준으로 끌어온 이미지의 오프셋을 검색합니다.
ImageList_GetIcon 이미지에서 아이콘을 만들고 이미지 목록의 마스크를 만듭니다.
ImageList_GetIconSize 이미지 목록에서 이미지 차원을 검색합니다. 이미지 목록의 모든 이미지는 차원이 동일합니다.
ImageList_GetImageCount 이미지 목록의 이미지 수를 검색합니다.
ImageList_GetImageInfo 이미지에 대 한 정보를 검색합니다.
ImageList_LoadBitmap ImageList_LoadImage 함수를 호출하여 지정된 비트맵 리소스에서 이미지 목록을 만듭니다.
ImageList_LoadImageA 지정된 비트맵에서 이미지 목록을 만듭니다. (ANSI)
ImageList_LoadImageW 지정된 비트맵에서 이미지 목록을 만듭니다. (유니코드)

ImageList_Merge 기존의 두 이미지를 결합하여 새 이미지를 만듭니다. 또한 이 함수는 이미지를 저장할 새 이미지 목록을 만듭니다.
ImageList_Read 스트림에서 이미지 목록을 읽습니다.
ImageList_ReadEx 스트림에서 이미지 목록을 읽고 이미지 목록에 IImageList 인터페이스를 반환합니다.
ImageList_Remove 이미지 목록에서 이미지를 제거합니다. (ImageList_Remove)
ImageList_RemoveAll ImageList_Remove 함수를 호출하여 이미지 목록에서 모든 이미지를 제거합니다.
ImageList_Replace 이미지 목록에 있는 이미지를 새 이미지로 바꿉니다. (ImageList_Replace)
ImageList_ReplaceIcon 이미지를 아이콘 또는 커서로 바꿉니다. (ImageList_ReplaceIcon)
ImageList_SetBkColor 이미지 목록의 배경색을 설정합니다. 이 함수는 아이콘을 추가하거나 흑백 비트맵이 있는 ImageList_AddMasked를 사용하는 경우에만 적용됩니다. 마스크가 없으면 전체 이미지가 그려지므로 배경색이 표시되지 않습니다.
ImageList_SetDragCursorImage 지정된 이미지(일반적으로 마우스 커서 이미지)를 현재 끌기 이미지와 결합하여 새 끌기 이미지를 만듭니다.
ImageList_SetIconSize 이미지 목록의 이미지 차원을 설정하고 목록에서 모든 이미지를 제거합니다. (ImageList_SetIconSize)
ImageList_SetImageCount 기존 이미지 목록의 크기를 조정합니다. (ImageList_SetImageCount)

ImageList_SetOverlayImage 오버레이 마스크로 사용할 이미지 목록에 지정된 이미지를 추가합니다. 이미지 목록에는 버전 4.70 이하에서 최대 4개, 버전 4.71에서 최대 15개의 오버레이 마스크를 포함할 수 있습니다. 이 함수는 지정된 이미지에 오버레이 마스크 인덱스를 할당합니다.
ImageList_Write 스트림에 이미지 목록을 씁니다. (ImageList_Write)
ImageList_WriteEx 스트림에 이미지 목록을 씁니다. (ImageList_WriteEx)
INDEXTOOVERLAYMASK ImageList_Draw 함수에서 사용할 수 있도록 오버레이 마스크의 인덱스를 준비합니다.
INDEXTOSTATEIMAGEMASK 트리 뷰 컨트롤 또는 목록 뷰 컨트롤이 인덱스를 사용하여 항목의 상태 이미지를 검색할 수 있도록 상태 이미지의 인덱스를 준비합니다.
InitCommonControls 특정 공용 컨트롤 창 클래스를 등록하고 초기화합니다. 이 함수는 사용되지 않습니다. 새 애플리케이션은 InitCommonControlsEx 함수를 사용해야 합니다.
InitCommonControlsEx 공용 컨트롤 DLL(Comctl32.dll)이 로드되었는지 확인하고 DLL에서 특정 공통 컨트롤 클래스를 등록합니다. 애플리케이션은 공용 컨트롤을 만들기 전에 이 함수를 호출해야 합니다.
InitializeFlatSB 특정 창에 대한 플랫 스크롤 막대를 초기화합니다.
InitMUILanguage 애플리케이션이 시스템 언어와 다른 공용 컨트롤과 함께 사용할 언어를 지정할 수 있도록 합니다.
LBItemFromPt 목록 상자에 지정된 지점에서 항목의 인덱스를 검색합니다.
ListView_ApproximateViewRect 지정된 항목 수를 표시하는 데 필요한 대략적인 너비와 높이를 계산합니다. 이 매크로를 사용하거나 LVM_APPROXIMATEVIEWRECT 메시지를 명시적으로 보낼 수 있습니다.

ListView_Arrange 아이콘 보기에서 항목을 정렬합니다. 이 매크로를 사용하거나 LVM_ARRANGE 메시지를 명시적으로 보낼 수 있습니다.
ListView_CancelEditLabel 항목 텍스트 편집 작업을 취소합니다. 이 매크로를 사용하거나 LVM_CANCELEDITLABEL 메시지를 명시적으로 보낼 수 있습니다.
ListView_CreateDragImage 지정된 항목에 대한 끌기 이미지 목록을 만듭니다. 이 매크로를 사용하거나 LVM_CREATEDRAGIMAGE 메시지를 명시적으로 보낼 수 있습니다.
ListView_DeleteAllItems 목록 뷰 컨트롤에서 모든 항목을 제거합니다. 이 매크로를 사용하거나 LVM_DELETEALLITEMS 메시지를 명시적으로 보낼 수 있습니다.
ListView_DeleteColumn 목록 뷰 컨트롤에서 열을 제거합니다. 이 매크로를 사용하거나 LVM_DELETECOLUMN 메시지를 명시적으로 보낼 수 있습니다.
ListView_DeleteItem 목록 뷰 컨트롤에서 항목을 제거합니다. 이 매크로를 사용하거나 LVM_DELETEITEM 메시지를 명시적으로 보낼 수 있습니다.
ListView_EditLabel 지정된 목록 뷰 항목의 텍스트를 바로 편집하기 시작합니다. 메시지는 지정된 항목을 암시적으로 선택하고 포커스를 지정합니다. 이 매크로를 사용하거나 LVM_EDITLABEL 메시지를 명시적으로 보낼 수 있습니다.
ListView_EnableGroupView 목록 뷰 컨트롤의 항목이 그룹으로 표시되는지 여부를 사용 또는 사용하지 않도록 설정합니다. 이 매크로를 사용하거나 LVM_ENABLEGROUPVIEW 메시지를 명시적으로 보낼 수 있습니다.
ListView_EnsureVisible 필요한 경우 목록 뷰 컨트롤을 스크롤하여 목록 뷰 항목이 전체적으로 또는 부분적으로 표시되도록 합니다. 이 매크로를 사용하거나 LVM_ENSUREVISIBLE 메시지를 명시적으로 보낼 수 있습니다.

ListView_FindItem 지정된 특성을 가진 목록 뷰 항목을 검색합니다. 이 매크로를 사용하거나 LVM_FINDITEM 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetBkColor 목록 뷰 컨트롤에서 배경색을 가져옵니다. 이 매크로를 사용하거나 LVM_GETBKCOLOR 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetBkImage 목록 뷰 컨트롤에서 배경 이미지를 가져옵니다. 이 매크로를 사용하거나 LVM_GETBKIMAGE 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetCallbackMask 목록 뷰 컨트롤의 콜백 마스크를 가져옵니다. 이 매크로를 사용하거나 LVM_GETCALLBACKMASK 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetCheckState 목록 뷰 컨트롤의 항목이 선택되어 있는지 여부를 확인합니다. 이것은 LVS_EX_CHECKBOXES 스타일이 있는 목록 뷰 컨트롤에만 사용해야 합니다.
ListView_GetColumn 목록 뷰 컨트롤 열의 특성을 가져옵니다. 이 매크로를 사용하거나 LVM_GETCOLUMN 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetColumnOrderArray 목록 뷰 컨트롤에서 열의 현재 왼쪽에서 오른쪽 순서를 가져옵니다. 이 매크로를 사용하거나 LVM_GETCOLUMNORDERARRAY 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetColumnWidth 보고서 또는 목록 뷰에서 열의 너비를 가져옵니다. 이 매크로를 사용하거나 LVM_GETCOLUMNWIDTH 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetCountPerPage 목록 또는 보고서 뷰에 있을 때 목록 뷰 컨트롤의 표시 영역에 수직으로 들어갈 수 있는 항목 수를 계산합니다. 완전히 표시되는 항목만 계산됩니다. 이 매크로를 사용하거나 LVM_GETCOUNTPERPAGE 메시지를 명시적으로 보낼 수 있습니다.

ListView_GetEditControl 목록 보기 항목의 텍스트를 편집하는 데 사용되는 편집 컨트롤에 대한 핸들을 가져옵니다. 이 매크로를 사용하거나 LVM_GETEDITCONTROL 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetEmptyText 목록 뷰 컨트롤이 비어 있을 때 표시할 텍스트를 가져옵니다. 이 매크로를 사용하거나 LVM_GETEMPTYTEXT 메시지를 명시적으로 보냅니다.
ListView_GetExtendedListViewStyle 지정된 목록 뷰 컨트롤에 현재 사용 중인 확장 스타일을 가져옵니다. 이 매크로를 사용하거나 LVM_GETEXTENDEDLISTVIEWSTYLE 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetFocusedGroup 포커스가 있는 그룹을 가져옵니다. 이 매크로를 사용하거나 LVM_GETFOCUSEDGROUP 메시지를 명시적으로 보냅니다.
ListView_GetFooterInfo 지정된 목록 뷰 컨트롤의 바닥글에 대한 정보를 가져옵니다. 이 매크로를 사용하거나 LVM_GETFOOTERINFO 메시지를 명시적으로 보냅니다.
ListView_GetFooterItem 지정된 목록 뷰 컨트롤의 바닥글 항목에 대한 정보를 가져옵니다. 이 매크로를 사용하거나 LVM_GETFOOTERITEM 메시지를 명시적으로 보냅니다.
ListView_GetFooterItemRect 목록 뷰 컨트롤에서 지정된 항목의 바닥글 좌표를 가져옵니다. 이 매크로를 사용하거나 LVM_GETFOOTERITEMRECT 메시지를 명시적으로 보냅니다.
ListView_GetFooterRect 지정된 목록 뷰 컨트롤의 바닥글 좌표를 가져옵니다. 이 매크로를 사용하거나 LVM_GETFOOTERRECT 메시지를 명시적으로 보냅니다.
ListView_GetGroupCount 그룹의 개수를 가져옵니다. 이 매크로를 사용하거나 LVM_GETGROUPCOUNT 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetGroupHeaderImageList 기존 목록 뷰 컨트롤에 대해 설정된 그룹 헤더 이미지 목록을 가져옵니다.

ListView_GetGroupInfo 그룹 정보를 가져옵니다. 이 매크로를 사용하거나 LVM_GETGROUPINFO 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetGroupInfoByIndex 지정된 그룹에 대한 정보를 가져옵니다. 이 매크로를 사용하거나 LVM_GETGROUPINFOBYINDEX 메시지를 명시적으로 보냅니다.
ListView_GetGroupMetrics 그룹 표시에 대한 정보를 가져옵니다. 이 매크로를 사용하거나 LVM_GETGROUPMETRICS 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetGroupRect 지정된 그룹의 사각형을 가져옵니다. 이 매크로를 사용하거나 LVM_GETGROUPRECT 메시지를 명시적으로 보냅니다.
ListView_GetGroupState 지정된 그룹의 상태를 가져옵니다. 이 매크로를 사용하거나 LVM_GETGROUPSTATE 메시지를 명시적으로 보냅니다.
ListView_GetHeader 목록 뷰 컨트롤에서 사용하는 헤더 컨트롤에 대한 핸들을 가져옵니다. 이 매크로를 사용하거나 LVM_GETHEADER 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetHotCursor 핫 추적을 사용하도록 설정한 상태에서 포인터가 항목 위에 있을 때 사용되는 HCURSOR 값을 검색합니다. 이 매크로를 사용하거나 LVM_GETHOTCURSOR 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetHotItem 핫 항목의 인덱스를 가져옵니다. 이 매크로를 사용하거나 LVM_GETHOTITEM 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetHoverTime 항목을 선택하기 전에 항목 위로 마우스 커서를 이동해야 하는 시간을 가져옵니다. 이 매크로를 사용하거나 LVM_GETHOVERTIME 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetImageList 목록 뷰 항목을 그리는 데 사용되는 이미지 목록에 대한 핸들을 가져옵니다. 이 매크로를 사용하거나 LVM_GETIMAGELIST 메시지를 명시적으로 보낼 수 있습니다.

ListView_GetInsertMark 삽입 지점의 위치를 가져옵니다. 이 매크로를 사용하거나 LVM_GETINSERTMARK 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetInsertMarkColor 삽입 지점의 색을 가져옵니다. 이 매크로를 사용하거나 LVM_GETINSERTMARKCOLOR 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetInsertMarkRect 삽입 지점을 경계하는 사각형을 가져옵니다. 이 매크로를 사용하거나 LVM_GETINSERTMARKRECT 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetISearchString 목록 뷰 컨트롤의 증분 검색 문자열을 가져옵니다. 이 매크로를 사용하거나 LVM_GETISEARCHSTRING 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetItem 목록 뷰 항목의 특성 중 일부 또는 전부를 가져옵니다. 이 매크로를 사용하거나 LVM_GETITEM 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetItemCount 목록 뷰 컨트롤의 항목 수를 가져옵니다. 이 매크로를 사용하거나 LVM_GETITEMCOUNT 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetItemIndexRect 지정된 목록 뷰 컨트롤의 현재 뷰에서 하위 항목의 전체 또는 일부에 대한 경계 사각형을 가져옵니다. 이 매크로를 사용하거나 LVM_GETITEMINDEXRECT 메시지를 명시적으로 보냅니다.
ListView_GetItemPosition 목록 뷰 항목의 위치를 가져옵니다. 이 매크로를 사용하거나 LVM_GETITEMPOSITION 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetItemRect 현재 뷰에서 항목의 전체 또는 일부에 대한 경계 사각형을 가져옵니다. 이 매크로를 사용하거나 LVM_GETITEMRECT 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetItemSpacing 목록 뷰 컨트롤의 항목 간 간격을 결정합니다. 이 매크로를 사용하거나 LVM_GETITEMSPACING 메시지를 명시적으로 보낼 수 있습니다.

ListView_GetItemState 목록 뷰 항목의 상태를 가져옵니다. 이 매크로를 사용하거나 LVM_GETITEMSTATE 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetItemText 목록 뷰 항목 또는 하위 항목의 텍스트를 가져옵니다. 이 매크로를 사용하거나 LVM_GETITEMTEXT 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetNextItem 지정된 속성이 있고 지정된 항목에 대한 지정된 관계를 포함하는 목록 뷰 항목을 검색합니다. 이 매크로를 사용하거나 LVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetNextItemIndex 지정된 속성과 다른 특정 항목과의 관계가 있는 특정 목록 뷰 컨트롤의 항목 인덱스를 가져옵니다. 이 매크로를 사용하거나 LVM_GETNEXTITEMINDEX 메시지를 명시적으로 보냅니다.
ListView_GetNumberOfWorkAreas 목록 뷰 컨트롤의 작업 영역 수를 가져옵니다. 이 매크로를 사용하거나 LVM_GETNUMBEROFWORKAREAS 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetOrigin 목록 뷰 컨트롤의 현재 뷰 원점을 가져옵니다. 이 매크로를 사용하거나 LVM_GETORIGIN 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetOutlineColor LVS_EX_BORDERSELECT 확장 창 스타일이 설정된 경우 목록 뷰 컨트롤의 테두리 색을 가져옵니다. 이 매크로를 사용하거나 LVM_GETOUTLINECOLOR 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetSelectedColumn 선택한 열을 지정하는 정수 를 가져옵니다. 이 매크로를 사용하거나 LVM_GETSELECTEDCOLUMN 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetSelectedCount 목록 뷰 컨트롤에서 선택한 항목의 수를 결정합니다. 이 매크로를 사용하거나 LVM_GETSELECTEDCOUNT 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetSelectionMark 목록 뷰 컨트롤에서 선택 표시를 가져옵니다. 이 매크로를 사용하거나 LVM_GETSELECTIONMARK 메시지를 명시적으로 보낼 수 있습니다.

ListView_GetStringWidth 지정된 목록 뷰 컨트롤의 현재 글꼴을 사용하여 지정된 문자열의 너비를 결정합니다. 이 매크로를 사용하거나 LVM_GETSTRINGWIDTH 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetSubItemRect 목록 뷰 컨트롤의 하위 사이트를 둘러싸는 사각형에 대한 정보를 가져옵니다.
ListView_GetTextBkColor 목록 뷰 컨트롤의 텍스트 배경색을 가져옵니다. 이 매크로를 사용하거나 LVM_GETTEXTBKCOLOR 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetTextColor 목록 뷰 컨트롤의 텍스트 색을 가져옵니다. 이 매크로를 사용하거나 LVM_GETTEXTCOLOR 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetTileInfo 목록 뷰 컨트롤의 타일에 대한 정보를 가져옵니다. 이 매크로를 사용하거나 LVM_GETTILEINFO 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetTileViewInfo 타일 뷰에서 목록 뷰 컨트롤에 대한 정보를 가져옵니다. 이 매크로를 사용하거나 LVM_GETTILEVIEWINFO 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetToolTips 목록 뷰 컨트롤이 도구 설명을 표시하는 데 사용하는 도구 설명 컨트롤을 가져옵니다. 이 매크로를 사용하거나 LVM_GETTOOLTIPS 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetTopIndex 목록 또는 보고서 뷰에 있을 때 가장 많이 표시되는 항목의 인덱스를 가져옵니다. 이 매크로를 사용하거나 LVM_GETTOPINDEX 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetUnicodeFormat 컨트롤의 유니코드 문자 형식 플래그를 가져옵니다. 이 매크로를 사용하거나 LVM_GETUNICODEFORMAT 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetView 목록 뷰 컨트롤의 현재 뷰를 가져옵니다. 이 매크로를 사용하거나 LVM_GETVIEW 메시지를 명시적으로 보낼 수 있습니다.

ListView_GetViewRect 목록 뷰 컨트롤에 있는 모든 항목의 경계 사각형을 가져옵니다. 목록 뷰는 아이콘 또는 작은 아이콘 뷰에 있어야 합니다. 이 매크로를 사용하거나 LVM_GETVIEWRECT 메시지를 명시적으로 보낼 수 있습니다.
ListView_GetWorkAreas 목록 뷰 컨트롤에서 작업 영역을 가져옵니다. 이 매크로를 사용하거나 LVM_GETWORKAREAS 메시지를 명시적으로 보낼 수 있습니다.
ListView_HasGroup 목록 뷰 컨트롤에 지정된 그룹이 있는지 여부를 확인합니다. 이 매크로를 사용하거나 LVM_HASGROUP 메시지를 명시적으로 보낼 수 있습니다.
ListView_HitTest 있는 경우 지정된 위치에 있는 목록 뷰 항목을 결정합니다. 이 매크로를 사용하거나 LVM_HITTEST 메시지를 명시적으로 보낼 수 있습니다. (ListView_HitTest)
ListView_HitTestEx 있는 경우 지정된 위치에 있는 목록 뷰 항목을 결정합니다. 이 매크로를 사용하거나 LVM_HITTEST 메시지를 명시적으로 보낼 수 있습니다. (ListView_HitTestEx)
ListView_InsertColumn 목록 뷰 컨트롤에 새 열을 삽입합니다. 이 매크로를 사용하거나 LVM_INSERTCOLUMN 메시지를 명시적으로 보낼 수 있습니다.
ListView_InsertGroup 그룹을 목록 뷰 컨트롤에 삽입합니다. 이 매크로를 사용하거나 LVM_INSERTGROUP 메시지를 명시적으로 보낼 수 있습니다.
ListView_InsertGroupSorted 순서가 지정된 그룹 목록에 그룹을 삽입합니다. 이 매크로를 사용하거나 LVM_INSERTGROUPSORTED 메시지를 명시적으로 보낼 수 있습니다.
ListView_InsertItem 목록 뷰 컨트롤에 새 항목을 삽입합니다. 이 매크로를 사용하거나 LVM_INSERTITEM 메시지를 명시적으로 보낼 수 있습니다.

ListView_InsertMarkHitTest 지정된 지점에 가장 가까운 삽입 지점을 검색합니다. 이 매크로를 사용하거나 LVM_INSERTMARKHITTEST 메시지를 명시적으로 보낼 수 있습니다.
ListView_IsGroupViewEnabled 목록 뷰 컨트롤에 그룹 보기가 사용하도록 설정되어 있는지 확인합니다. 이 매크로를 사용하거나 LVM_ISGROUPVIEWENABLED 메시지를 명시적으로 보낼 수 있습니다.
ListView_IsItemVisible 목록 뷰 컨트롤의 항목이 표시되는지 여부를 나타냅니다. 이 매크로를 사용하거나 LVM_ISITEMVISIBLE 메시지를 명시적으로 보냅니다.
ListView_MapIDToIndex 항목의 ID를 인덱스로 매핑합니다. 이 매크로를 사용하거나 LVM_MAPIDTOINDEX 메시지를 명시적으로 보낼 수 있습니다.
ListView_MapIndexToID 항목의 인덱스가 고유 ID에 매핑됩니다. 이 매크로를 사용하거나 LVM_MAPINDEXTOID 메시지를 명시적으로 보낼 수 있습니다.
ListView_MoveGroup 이 매크로는 구현되지 않습니다. (ListView_MoveGroup)
ListView_MoveItemToGroup 이 매크로는 구현되지 않습니다. (ListView_MoveItemToGroup)
ListView_RedrawItems 목록 뷰 컨트롤이 항목 범위를 다시 그리도록 합니다. 이 매크로를 사용하거나 LVM_REDRAWITEMS 메시지를 명시적으로 보낼 수 있습니다.
ListView_RemoveAllGroups 목록 뷰 컨트롤에서 모든 그룹을 제거합니다. 이 매크로를 사용하거나 LVM_REMOVEALLGROUPS 메시지를 명시적으로 보낼 수 있습니다.
ListView_RemoveGroup 목록 뷰 컨트롤에서 그룹을 제거합니다. 이 매크로를 사용하거나 LVM_REMOVEGROUP 메시지를 명시적으로 보낼 수 있습니다.

ListView_Scroll 목록 뷰 컨트롤의 콘텐츠를 스크롤합니다. 이 매크로를 사용하거나 LVM_SCROLL 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetBkColor 목록 뷰 컨트롤에서 배경색을 설정합니다. 이 매크로를 사용하거나 LVM_SETBKCOLOR 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetBkImage 목록 뷰 컨트롤에서 배경 이미지를 설정합니다. 이 매크로를 사용하거나 LVM_SETBKIMAGE 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetCallbackMask 목록 뷰 컨트롤의 콜백 마스크를 변경합니다. 이 매크로를 사용하거나 LVM_SETCALLBACKMASK 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetCheckState 목록 뷰 컨트롤에서 항목을 선택하거나 선택 취소합니다. 이 매크로를 사용하거나 LVM_SETITEMSTATE 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetColumn 목록 뷰 열의 특성을 설정합니다. 이 매크로를 사용하거나 LVM_SETCOLUMN 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetColumnOrderArray 목록 뷰 컨트롤에서 열의 왼쪽에서 오른쪽 순서를 설정합니다. 이 매크로를 사용하거나 LVM_SETCOLUMNORDERARRAY 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetColumnWidth 보고서 뷰에서 열의 너비 또는 목록 뷰 모드에서 모든 열의 너비를 변경하는 데 사용됩니다. 이 매크로를 사용하거나 LVM_SETCOLUMNWIDTH 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetExtendedListViewStyle 목록 뷰 컨트롤을 위한 확장 스타일을 설정합니다. 이 매크로를 사용하거나 LVM_SETEXTENDEDLISTVIEWSTYLE 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetExtendedListViewStyleEx 스타일 마스크를 사용하여 목록 뷰 컨트롤을 위한 확장 스타일을 설정합니다. 이 매크로를 사용하거나 LVM_SETEXTENDEDLISTVIEWSTYLE 메시지를 명시적으로 보낼 수 있습니다.

ListView_SetGroupHeaderImageList 목록 뷰 컨트롤의 그룹 머리글에 이미지 목록을 할당합니다.
ListView_SetGroupInfo 그룹 정보를 설정합니다. 이 매크로를 사용하거나 LVM_SETGROUPINFO 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetGroupMetrics 그룹 표시에 대한 정보를 설정합니다. 이 매크로를 사용하거나 LVM_SETGROUPMETRICS 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetGroupState 지정된 그룹의 상태를 설정합니다.
ListView_SetHotCursor 핫 추적이 사용하도록 설정된 상태에서 포인터가 항목 위에 있을 때 목록 뷰 컨트롤이 사용하는 HCURSOR를 설정합니다. 이 매크로를 사용하거나 LVM_SETHOTCURSOR 메시지를 명시적으로 보낼 수 있습니다. 핫 추적이 사용하도록 설정되었는지 확인하려면 SystemParametersInfo를 호출합니다.
ListView_SetHotItem 목록 뷰 컨트롤에서 핫 항목을 설정합니다. 이 매크로를 사용하거나 LVM_SETHOTITEM 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetHoverTime 항목을 선택하기 전에 항목 위로 마우스 커서를 이동해야 하는 시간을 설정합니다. 이 매크로를 사용하거나 LVM_SETHOVERTIME 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetIconSpacing 목록 뷰 컨트롤의 아이콘 사이의 간격을 LVS_ICON 스타일로 설정합니다. 이 매크로를 사용하거나 LVM_SETICONSPACING 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetImageList 목록 뷰 컨트롤에 이미지 목록을 할당합니다. 이 매크로를 사용하거나 LVM_SETIMAGELIST 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetInfoTip 도구 설명 텍스트를 설정합니다. 이 매크로를 사용하거나 LVM_SETINFOTIP 메시지를 명시적으로 보낼 수 있습니다.

ListView_SetInsertMark 삽입 지점을 정의된 위치로 설정합니다. 이 매크로를 사용하거나 LVM_SETINSERTMARK 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetInsertMarkColor 삽입 지점의 색을 설정합니다. 이 매크로를 사용하거나 LVM_SETINSERTMARKCOLOR 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetItem 목록 뷰 항목의 특성 중 일부 또는 전부를 설정합니다. ListView_SetItem을 사용하여 하위 항목의 텍스트를 설정할 수도 있습니다. 이 매크로를 사용하거나 LVM_SETITEM 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetItemCount 목록 뷰 컨트롤이 지정된 수의 항목에 대한 메모리를 할당하도록 합니다. 이 매크로를 사용하거나 LVM_SETITEMCOUNT 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetItemCountEx 가상 목록 뷰에 있는 항목의 가상 번호를 설정합니다. 이 매크로를 사용하거나 LVM_SETITEMCOUNT 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetItemIndexState 지정된 목록 뷰 항목의 상태를 설정합니다. 이 매크로를 사용하거나 LVM_SETITEMINDEXSTATE 메시지를 명시적으로 보냅니다.
ListView_SetItemPosition 항목을 목록 뷰 컨트롤(아이콘 또는 작은 아이콘 뷰)의 지정된 위치로 이동합니다. 이 매크로를 사용하거나 LVM_SETITEMPOSITION 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetItemPosition32 항목을 목록 뷰 컨트롤(아이콘 또는 작은 아이콘 뷰)의 지정된 위치로 이동합니다.
ListView_SetItemState 목록 뷰 컨트롤에서 항목의 상태를 변경합니다. 이 매크로를 사용하거나 LVM_SETITEMSTATE 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetItemText 목록 뷰 항목 또는 하위 항목의 텍스트를 변경합니다. 이 매크로를 사용하거나 LVM_SETITEMTEXT 메시지를 명시적으로 보낼 수 있습니다.

ListView_SetOutlineColor LVS_EX_BORDERSELECT 확장 창 스타일이 설정된 경우 목록 보기 컨트롤의 테두리 색을 설정합니다. 이 매크로를 사용하거나 LVM_SETOUTLINECOLOR 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetSelectedColumn 선택한 열의 인덱스를 설정합니다. 이 매크로를 사용하거나 LVM_SETSELECTEDCOLUMN 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetSelectionMark 목록 뷰 컨트롤에서 선택 표시를 설정합니다. 이 매크로를 사용하거나 LVM_SETSELECTIONMARK 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetTextBkColor list-view 컨트롤에서 텍스트의 배경색을 설정합니다. 이 매크로를 사용하거나 LVM_SETTEXTBKCOLOR 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetTextColor 목록 뷰 컨트롤의 텍스트 색을 설정합니다. 이 매크로를 사용하거나 LVM_SETTEXTCOLOR 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetTileInfo 목록 뷰 컨트롤의 기존 타일에 대한 정보를 설정합니다. 이 매크로를 사용하거나 LVM_SETTILEINFO 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetTileViewInfo 목록 뷰 컨트롤이 타일 보기에서 사용하는 정보를 설정합니다. 이 매크로를 사용하거나 LVM_SETTILEVIEWINFO 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetToolTips 목록 뷰 컨트롤이 도구 설명을 표시하는 데 사용할 도구 설명 컨트롤을 설정합니다. 이 매크로를 사용하거나 LVM_SETTOOLTIPS 메시지를 명시적으로 보낼 수 있습니다.
ListView_SetUnicodeFormat 컨트롤의 유니코드 문자 형식 플래그를 설정합니다. (ListView_SetUnicodeFormat)
ListView_SetView 목록 뷰 컨트롤의 뷰를 설정합니다. 이 매크로를 사용하거나 LVM_SETVIEW 메시지를 명시적으로 보낼 수 있습니다.

ListView_SetWorkAreas 목록 뷰 컨트롤 내에서 작업 영역을 설정합니다. 이 매크로를 사용하거나 LVM_SETWORKAREAS 메시지를 명시적으로 보낼 수 있습니다.
ListView_SortGroups 애플리케이션 정의 비교 함수를 사용하여 목록 뷰 컨트롤 내에서 ID별로 그룹을 정렬합니다. 이 매크로를 사용하거나 LVM_SORTGROUPS 메시지를 명시적으로 보낼 수 있습니다.
ListView_SortItems 애플리케이션 정의 비교 함수를 사용하여 목록 뷰 컨트롤의 항목을 정렬합니다. 각 항목의 인덱스는 새 시퀀스를 반영하도록 변경됩니다. 이 매크로를 사용하거나 LVM_SORTITEMS 메시지를 명시적으로 보낼 수 있습니다.
ListView_SortItemsEx 애플리케이션 정의 비교 함수를 사용하여 목록 뷰 컨트롤의 항목을 정렬합니다. 각 항목의 인덱스는 새 시퀀스를 반영하도록 변경됩니다. 이 매크로를 사용하거나 LVM_SORTITEMSEX 메시지를 명시적으로 보낼 수 있습니다.
ListView_SubItemHitTest 지정된 위치에 있는 목록 뷰 항목 또는 하위 항목을 결정합니다. 이 매크로를 사용하거나 LVM_SUBITEMHITTEST 메시지를 명시적으로 보낼 수 있습니다. (ListView_SubItemHitTest)
ListView_SubItemHitTestEx 지정된 위치에 있는 목록 뷰 항목 또는 하위 항목을 결정합니다. 이 매크로를 사용하거나 LVM_SUBITEMHITTEST 메시지를 명시적으로 보낼 수 있습니다. (ListView_SubItemHitTestEx)
ListView_Update 목록 뷰 항목을 업데이트합니다. list-view 컨트롤에 LVS_AUTOARRANGE 스타일이 있는 경우 이 매크로를 사용하면 목록 보기 컨트롤이 정렬됩니다. 이 매크로를 사용하거나 LVM_UPDATE 메시지를 명시적으로 보낼 수 있습니다.
LoadIconMetric 클라이언트 지정 시스템 메트릭을 사용하여 지정된 아이콘 리소스를 로드합니다.
LoadIconWithScaleDown 아이콘을 로드합니다. 아이콘이 표준 크기가 아닌 경우 이 함수는 작은 이미지를 확대하는 대신 큰 이미지의 크기를 조정합니다.

MakeDragList 지정된 단일 선택 목록 상자를 끌기 목록 상자로 변경합니다.
MAKEIPADDRESS 4바이트 값을 IPM_SETADDRESS 메시지에 사용하기에 적합한 단일 LPARAM으로 압축합니다.
MAKEIPRANGE 2바이트 값을 IPM_SETRANGE 메시지에 사용하기에 적합한 단일 LPARAM으로 압축합니다.
MenuHelp WM_MENUSELECT 및 WM_COMMAND 메시지를 처리하고 지정된 상태 창에 현재 메뉴에 대한 도움말 텍스트를 표시합니다.
MonthCal_GetCalendarBorder 월 달력 컨트롤의 테두리 크기(픽셀)를 가져옵니다. 이 매크로를 사용하거나 MCM_GETCALENDARBORDER 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetCalendarCount 달력 컨트롤에 현재 표시된 달력 수를 가져옵니다. 이 매크로를 사용하거나 MCM_GETCALENDARCOUNT 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetCalendarGridInfo 일정 눈금에 대한 정보를 가져옵니다.
MonthCal_GetCALID 지정된 달력 컨트롤의 현재 달력 ID를 가져옵니다. 이 매크로를 사용하거나 MCM_GETCALID 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetColor 월 달력 컨트롤의 지정된 부분에 대한 색을 검색합니다. 이 매크로를 사용하거나 MCM_GETCOLOR 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetCurrentView 월 달력 컨트롤에 대한 뷰를 가져옵니다. 이 매크로를 사용하거나 MCM_GETCURRENTVIEW 메시지를 명시적으로 보낼 수 있습니다.

MonthCal_GetCurSel 현재 선택된 날짜를 검색합니다. 이 매크로를 사용하거나 MCM_GETCURSEL 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetFirstDayOfWeek 월 달력 컨트롤의 첫 번째 요일을 검색합니다. 이 매크로를 사용하거나 MCM_GETFIRSTDAYOFWEEK 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetMaxSelCount 월 달력 컨트롤에서 선택할 수 있는 최대 날짜 범위를 검색합니다. 이 매크로를 사용하거나 MCM_GETMAXSELCOUNT 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetMaxTodayWidth 월 달력 컨트롤에서 "today" 문자열의 최대 너비를 검색합니다. 여기에는 레이블 텍스트와 날짜 텍스트가 포함됩니다. 이 매크로를 사용하거나 MCM_GETMAXTODAYWIDTH 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetMinReqRect 월 달력 컨트롤에서 전체 월을 표시하는 데 필요한 최소 크기를 검색합니다. 크기 정보는 RECT 구조체의 형태로 제공됩니다. 이 매크로를 사용하거나 MCM_GETMINREQRECT 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetMonthDelta 월 달력 컨트롤의 스크롤 속도를 검색합니다. 스크롤 속도는 사용자가 스크롤 단추를 클릭할 때 컨트롤이 디스플레이를 이동하는 개월 수입니다. 이 매크로를 사용하거나 MCM_GETMONTHDELTA 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetMonthRange 월 달력 컨트롤 표시의 상한 및 하한을 나타내는 날짜 정보(SYSTEMTIME 구조체 사용)를 검색합니다. 이 매크로를 사용하거나 MCM_GETMONTHRANGE 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetRange 월 달력 컨트롤에 설정된 최소 및 최대 허용 날짜를 검색합니다. 이 매크로를 사용하거나 MCM_GETRANGE 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetSelRange 현재 사용자가 선택한 날짜 범위의 상한과 하한을 나타내는 날짜 정보를 가져옵니다. 이 매크로를 사용하거나 MCM_GETSELRANGE 메시지를 명시적으로 보낼 수 있습니다.

MonthCal_GetToday 월 달력 컨트롤에 대해 "today"로 지정된 날짜의 날짜 정보를 검색합니다. 이 매크로를 사용하거나 MCM_GETTODAY 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_GetUnicodeFormat 컨트롤의 유니코드 문자 형식 플래그를 검색합니다. 이 매크로를 사용하거나 MCM_GETUNICODEFORMAT 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_HitTest 화면의 지정된 지점에 있는 월 달력 컨트롤의 일부를 결정합니다. 이 매크로를 사용하거나 MCM_HITTEST 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetCalendarBorder 월 달력 컨트롤의 테두리 크기(픽셀)를 설정합니다. 이 매크로를 사용하거나 MCM_SETCALENDARBORDER 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetCALID 지정된 달력 컨트롤의 달력 ID를 설정합니다. 이 매크로를 사용하거나 MCM_SETCALID 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetColor 월 달력 컨트롤의 지정된 부분에 대한 색을 설정합니다. 이 매크로를 사용하거나 MCM_SETCOLOR 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetCurrentView 월 달력 컨트롤에 대한 뷰를 설정합니다. 이 매크로를 사용하거나 MCM_SETCURRENTVIEW 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetCurSel 월 달력 컨트롤에 대해 현재 선택한 날짜를 설정합니다. 지정된 날짜가 표시되지 않으면 컨트롤이 디스플레이를 업데이트하여 표시합니다. 이 매크로를 사용하거나 MCM_SETCURSEL 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetDayState 달력 컨트롤 내에서 현재 표시되는 모든 달의 날짜 상태를 설정합니다. 이 매크로를 사용하거나 MCM_SETDAYSTATE 메시지를 명시적으로 보낼 수 있습니다.

MonthCal_SetFirstDayOfWeek 월 달력 컨트롤의 첫 번째 요일을 설정합니다. 이 매크로를 사용하거나 MCM_SETFIRSTDAYOFWEEK 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetMaxSelCount 월 달력 컨트롤에서 선택할 수 있는 최대 일수를 설정합니다. 이 매크로를 사용하거나 MCM_SETMAXSELCOUNT 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetMonthDelta 월 달력 컨트롤에 대한 스크롤 비율을 설정합니다. 스크롤 속도는 사용자가 스크롤 단추를 클릭할 때 컨트롤이 디스플레이를 이동하는 개월 수입니다. 이 매크로를 사용하거나 MCM_SETMONTHDELTA 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetRange 월 달력 컨트롤에 허용되는 최소 및 최대 날짜를 설정합니다. 이 매크로를 사용하거나 MCM_SETRANGE 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetSelRange 월 달력 컨트롤에 대한 선택을 지정된 날짜 범위로 설정합니다. 이 매크로를 사용하거나 MCM_SETSELRANGE 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetToday 월 달력 컨트롤에 대한 "today" 선택을 설정합니다. 이 매크로를 사용하거나 MCM_SETTODAY 메시지를 명시적으로 보낼 수 있습니다.
MonthCal_SetUnicodeFormat 컨트롤의 유니코드 문자 형식 플래그를 설정합니다. (MonthCal_SetUnicodeFormat)
MonthCal_SizeRectToMin 지정된 사각형에 맞는 달력 수를 계산한 다음, 해당 달력 수에 맞게 사각형이 필요한 최소 크기를 반환합니다. 이 매크로를 사용하거나 MCM_SIZERECTTOMIN 메시지를 명시적으로 보낼 수 있습니다.
Pager_ForwardMouse Pager 컨트롤에 대한 마우스 전달을 사용하거나 사용하지 않도록 설정합니다. 마우스 전달이 사용하도록 설정되면 Pager 컨트롤이 포함된 창에 WM_MOUSEMOVE 메시지를 전달합니다. 이 매크로를 사용하거나 PGM_FORWARDMOUSE 메시지를 명시적으로 보낼 수 있습니다.

Pager_GetBkColor Pager 컨트롤의 현재 배경색을 검색합니다. 이 매크로를 사용하거나 PGM_GETBKCOLOR 메시지를 명시적으로 보낼 수 있습니다.
Pager_GetBorder Pager 컨트롤의 현재 테두리 크기를 검색합니다. 이 매크로를 사용하거나 PGM_GETBORDER 메시지를 명시적으로 보낼 수 있습니다.
Pager_GetButtonSize Pager 컨트롤의 현재 단추 크기를 검색합니다. 이 매크로를 사용하거나 PGM_GETBUTTONSIZE 메시지를 명시적으로 보낼 수 있습니다.
Pager_GetButtonState Pager 컨트롤에서 지정된 단추의 상태를 검색합니다. 이 매크로를 사용하거나 PGM_GETBUTTONSTATE 메시지를 명시적으로 보낼 수 있습니다.
Pager_GetDropTarget Pager 컨트롤의 IDropTarget 인터페이스 포인터를 검색합니다. 이 매크로를 사용하거나 PGM_GETDROPTARGET 메시지를 명시적으로 보낼 수 있습니다.
Pager_GetPos Pager 컨트롤의 현재 스크롤 위치를 검색합니다. 이 매크로를 사용하거나 PGM_GETPOS 메시지를 명시적으로 보낼 수 있습니다.
Pager_RecalcSize Pager 컨트롤이 포함된 창의 크기를 다시 계산하도록 합니다. 이 매크로를 사용하면 PGN_CALCSIZE 알림이 전송됩니다. 이 매크로를 사용하거나 PGM_RECALCSIZE 메시지를 명시적으로 보낼 수 있습니다.
Pager_SetBkColor Pager 컨트롤의 현재 배경색을 설정합니다. 이 매크로를 사용하거나 PGM_SETBKCOLOR 메시지를 명시적으로 보낼 수 있습니다.
Pager_SetBorder Pager 컨트롤의 현재 테두리 크기를 설정합니다. 이 매크로를 사용하거나 PGM_SETBORDER 메시지를 명시적으로 보낼 수 있습니다.

Pager_SetButtonSize Pager 컨트롤의 현재 단추 크기를 설정합니다. 이 매크로를 사용하거나 PGM_SETBUTTONSIZE 메시지를 명시적으로 보낼 수 있습니다.
Pager_SetChild Pager 컨트롤에 포함된 창을 설정합니다.
Pager_SetPos Pager 컨트롤의 스크롤 위치를 설정합니다. 이 매크로를 사용하거나 PGM_SETPOS 메시지를 명시적으로 보낼 수 있습니다.
Pager_SetScrollInfo 시간 제한 값, 시간 제한당 라인 및 라인당 픽셀을 포함하여 Pager 컨트롤의 스크롤 매개 변수를 설정합니다. 이 매크로를 사용하거나 PGM_SETSETSCROLLINFO 메시지를 명시적으로 보낼 수 있습니다.
RemoveWindowSubclass 창에서 하위 클래스 콜백을 제거합니다.
SECOND_IPADDRESS IPM_GETADDRESS 메시지와 함께 검색된 압축된 IP 주소에서 필드 1 값을 추출합니다.
SetWindowSubclass 창 서브클래스 콜백을 설치하거나 업데이트합니다.
ShowHideMenuCtl 지정된 메뉴 항목의 확인 표시 특성을 설정하거나 제거하고 해당 컨트롤을 표시하거나 숨깁니다.
TabCtrl_AdjustRect 창 사각형이 지정된 탭 컨트롤의 표시 영역을 계산하거나 지정된 표시 영역에 해당하는 창 사각형을 계산합니다. 이 매크로를 사용하거나 TCM_ADJUSTRECT 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_DeleteAllItems 탭 컨트롤에서 모든 항목을 제거합니다. 이 매크로를 사용하거나 TCM_DELETEALLITEMS 메시지를 명시적으로 보낼 수 있습니다.

TabCtrl_DeleteItem 탭 컨트롤에서 항목을 제거합니다. 이 매크로를 사용하거나 TCM_DELETEITEM 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_DeselectAll 탭 컨트롤의 항목을 초기화하여 TCIS_BUTTONPRESSED 상태로 설정된 모든 항목을 지웁니다. 이 매크로를 사용하거나 TCM_DESELECTALL 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_GetCurFocus 탭 컨트롤에 포커스가 있는 항목의 인덱스를 반환합니다. 이 매크로를 사용하거나 TCM_GETCURFOCUS 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_GetCurSel 탭 컨트롤에서 현재 선택된 탭을 결정합니다. 이 매크로를 사용하거나 TCM_GETCURSEL 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_GetExtendedStyle 현재 탭 컨트롤에 사용 중인 확장 스타일을 검색합니다. 이 매크로를 사용하거나 TCM_GETEXTENDEDSTYLE 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_GetImageList 탭 컨트롤과 연결된 이미지 목록을 검색합니다. 이 매크로를 사용하거나 TCM_GETIMAGELIST 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_GetItem 탭 컨트롤의 탭에 대한 정보를 검색합니다. 이 매크로를 사용하거나 TCM_GETITEM 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_GetItemCount 탭 컨트롤에서 탭 수를 검색합니다. 이 매크로를 사용하거나 TCM_GETITEMCOUNT 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_GetItemRect 탭 컨트롤의 탭에 대한 경계 사각형을 검색합니다. 이 매크로를 사용하거나 TCM_GETIMRECT 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_GetRowCount 탭 컨트롤에서 탭의 현재 행 수를 검색합니다. 이 매크로를 사용하거나 TCM_GETROWCOUNT 메시지를 명시적으로 보낼 수 있습니다.

TabCtrl_GetToolTips 탭 컨트롤과 연결된 도구 설명 컨트롤에 대한 핸들을 검색합니다. 이 매크로를 사용하거나 TCM_GETTOOLTIPS 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_GetUnicodeFormat 컨트롤의 유니코드 문자 형식 플래그를 검색합니다. 이 매크로를 사용하거나 TCM_GETUNICODEFORMAT 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_HighlightItem 탭 항목의 강조 표시 상태를 설정합니다. 이 매크로를 사용하거나 TCM_HIGHLIGHTITEM 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_HitTest 지정된 화면 위치에 있는 탭(있는 경우)을 결정합니다. 이 매크로를 사용하거나 TCM_HITTEST 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_InsertItem 탭 컨트롤에 새 탭을 삽입합니다. 이 매크로를 사용하거나 TCM_INSERTITEM 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_RemoveImage 탭 컨트롤의 이미지 목록에서 이미지를 제거합니다. 이 매크로를 사용하거나 TCM_REMOVEIMAGE 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_SetCurFocus 탭 컨트롤의 지정된 탭에 포커스를 설정합니다. 이 매크로를 사용하거나 TCM_SETCURFOCUS 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_SetCurSel 탭 컨트롤에서 탭을 선택합니다. 이 매크로를 사용하거나 TCM_SETCURSEL 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_SetExtendedStyle 탭 컨트롤이 사용할 확장 스타일을 설정합니다. 이 매크로를 사용하거나 TCM_SETEXTENDEDSTYLE 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_SetImageList 탭 컨트롤에 이미지 목록을 할당합니다. 이 매크로를 사용하거나 TCM_SETIMAGELIST 메시지를 명시적으로 보낼 수 있습니다.

TabCtrl_SetItem 탭 특성의 일부 또는 전부를 설정합니다. 이 매크로를 사용하거나 TCM_SETITEM 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_SetItemExtra 탭 컨트롤에서 애플리케이션 정의 데이터용으로 예약된 탭당 바이트 수를 설정합니다. 이 매크로를 사용하거나 TCM_SETITEMEXTRA 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_SetItemSize 고정 너비 또는 소유자가 그린 탭 컨트롤에서 탭의 너비와 높이를 설정합니다. 이 매크로를 사용하거나 TCM_SETITEMSIZE 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_SetMinTabWidth 탭 컨트롤에서 항목의 최소 너비를 설정합니다. 이 매크로를 사용하거나 TCM_SETMINTABWIDTH 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_SetPadding 탭 컨트롤에서 각 탭의 아이콘과 레이블 주변의 공간(패딩)을 설정합니다. 이 매크로를 사용하거나 TCM_SETPADDING 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_SetToolTips 탭 컨트롤에 도구 설명 컨트롤을 할당합니다. 이 매크로를 사용하거나 TCM_SETTOOLTIPS 메시지를 명시적으로 보낼 수 있습니다.
TabCtrl_SetUnicodeFormat 컨트롤의 유니코드 문자 형식 플래그를 설정합니다. (TabCtrl_SetUnicodeFormat)
TaskDialog TaskDialog 함수는 작업 대화 상자를 만들고, 표시하고, 작동합니다.
TaskDialogIndirect TaskDialogIndirect 함수는 작업 대화 상자를 만들고, 표시하고, 작동합니다.
THIRD_IPADDRESS IPM_GETADDRESS 메시지와 함께 검색된 압축된 IP 주소에서 필드 2 값을 추출합니다.
TreeView_CreateDragImage 트리 뷰 컨트롤에서 지정된 항목에 대한 끌기 비트맵을 만듭니다.

TreeView_DeleteAllItems 트리 뷰 컨트롤에서 모든 항목을 삭제합니다.
TreeView_DeleteItem 트리 뷰 컨트롤에서 항목과 모든 자식을 제거합니다. TVM_DELETEITEM 메시지를 명시적으로 보낼 수도 있습니다.
TreeView_EditLabel 지정된 항목의 텍스트에 대한 현재 위치 편집을 시작하여 항목의 텍스트를 텍스트가 포함된 한 줄 편집 컨트롤로 대체합니다.
TreeView_EndEditLabelNow 트리 뷰 항목의 레이블 편집을 종료합니다. 이 매크로를 사용하거나 TVM_ENDEDITLABELNOW 메시지를 명시적으로 보낼 수 있습니다.
TreeView_EnsureVisible 필요한 경우 트리 뷰 항목이 표시되고 부모 항목이 확장되거나 트리 뷰 컨트롤이 스크롤되도록 합니다. 이 매크로를 사용하거나 TVM_ENSUREVISIBLE 메시지를 명시적으로 보낼 수 있습니다.
TreeView_Expand TreeView_Expand 매크로는 지정된 부모 항목(있는 경우)과 연결된 자식 항목 목록을 확장하거나 축소합니다. 이 매크로를 사용하거나 TVM_EXPAND 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetBkColor 컨트롤의 현재 배경색을 검색합니다. 이 매크로를 사용하거나 TVM_GETBKCOLOR 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetCheckState 지정된 항목의 확인 상태를 가져옵니다. TVM_GETITEMSTATE 메시지를 직접 사용할 수도 있습니다.
TreeView_GetChild 지정된 트리 뷰 항목의 첫 번째 자식 항목을 검색합니다. 이 매크로를 사용하거나 TVGN_CHILD 플래그와 함께 TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetCount 트리 뷰 컨트롤에서 항목 수를 검색합니다. 이 매크로를 사용하거나 TVM_GETCOUNT 메시지를 명시적으로 보낼 수 있습니다.

TreeView_GetDropHilight 끌어서 놓기 작업의 대상인 트리 뷰 항목을 검색합니다. 이 매크로를 사용하거나 TVGN_DROPHILITE 플래그와 함께 TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetEditControl 트리 뷰 항목의 텍스트를 편집하는 데 사용되는 편집 컨트롤에 대한 핸들을 검색합니다. 이 매크로를 사용하거나 TVM_GETEDITCONTROL 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetExtendedStyle 지정된 트리 뷰 컨트롤에 대한 확장 스타일을 검색합니다. 이 매크로를 사용하거나 TVM_GETEXTENDEDSTYLE 메시지를 명시적으로 보냅니다.
TreeView_GetFirstVisible 트리 뷰 컨트롤 창에서 처음 표시되는 항목을 검색합니다. 이 매크로를 사용하거나 TVGN_FIRSTVISIBLE 플래그와 함께 TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetImageList 트리 뷰 컨트롤과 연결된 일반 또는 상태 이미지 목록에 대한 핸들을 검색합니다. 이 매크로를 사용하거나 TVM_GETIMAGELIST 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetIndent 자식 항목이 부모 항목에 비해 들여쓰기가 적용된 양을 픽셀 단위로 검색합니다. 이 매크로를 사용하거나 TVM_GETINDENT 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetInsertMarkColor 트리 뷰의 삽입 마크를 그리는 데 사용되는 색을 검색합니다. 이 매크로를 사용하거나 TVM_GETINSERTMARKCOLOR 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetISearchString 트리 뷰 컨트롤에 대한 증분 검색 문자열을 검색합니다. 트리 뷰 컨트롤은 증분 검색 문자열을 사용하여 사용자가 입력한 문자를 기반으로 항목을 선택합니다. 이 매크로를 사용하거나 TVM_GETISEARCHSTRING 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetItem 트리 뷰 항목의 특성 중 일부 또는 전부를 검색합니다. 이 매크로를 사용하거나 TVM_GETITEM 메시지를 명시적으로 보낼 수 있습니다.

TreeView_GetItemHeight 트리 뷰 항목의 현재 높이를 검색합니다. 이 매크로를 사용하거나 TVM_GETITEMHEIGHT 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetItemPartRect 항목의 지정된 부분에 대한 "적중 영역"을 구성하는 가능한 가장 큰 경계 사각형을 검색합니다. 이 매크로를 사용하거나 TVM_GETITEMPARTRECT 메시지를 명시적으로 보냅니다.
TreeView_GetItemRect 트리 뷰 항목의 경계 사각형을 검색하고 항목이 표시되는지 여부를 나타냅니다. 이 매크로를 사용하거나 TVM_GETITEMRECT 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetItemState 트리 뷰 항목의 상태 특성 중 일부 또는 전부를 검색합니다. 이 매크로를 사용하거나 TVM_GETITEMSTATE 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetLastVisible 트리 뷰 컨트롤에서 마지막으로 확장된 항목을 검색합니다. 트리 뷰 창에 표시되는 마지막 항목은 검색하지 않습니다. 이 매크로를 사용하거나 TVGN_LASTVISIBLE 플래그와 함께 TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetLineColor 현재 선 색을 가져옵니다. TVM_GETLINECOLOR 메시지를 직접 사용할 수도 있습니다.
TreeView_GetNextItem 지정된 항목과 지정된 관계를 포함하는 트리 보기 항목을 검색합니다. 이 매크로를 사용하거나, 아래에 설명된 TreeView_Get 매크로 중 하나를 사용하거나, TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetNextSelected 지정된 트리 항목에 대한 TVGN_NEXTSELECTED 관계를 포함하는 트리 뷰 항목을 검색합니다.
TreeView_GetNextSibling 트리 뷰 컨트롤에서 지정된 항목의 다음 형제 항목을 검색합니다. 이 매크로를 사용하거나 TVGN_NEXT 플래그와 함께 TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetNextVisible 트리 뷰 컨트롤에서 지정된 항목 다음에 표시되는 다음 항목을 검색합니다. 이 매크로를 사용하거나 TVGN_NEXTVISIBLE 플래그와 함께 TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.

TreeView_GetParent 지정된 트리 뷰 항목의 부모 항목을 검색합니다. 이 매크로를 사용하거나 TVGN_PARENT 플래그와 함께 TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetPrevSibling 트리 뷰 컨트롤에서 지정된 항목의 이전 형제 항목을 검색합니다. 이 매크로를 사용하거나 TVGN_PREVIOUS 플래그와 함께 TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetPrevVisible 트리 뷰 컨트롤에서 지정된 항목 앞에 표시되는 첫 번째 항목을 검색합니다. 이 매크로를 사용하거나 TVGN_PREVIOUSVISIBLE 플래그와 함께 TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetRoot 트리 뷰 컨트롤의 맨 위 또는 첫 번째 항목을 검색합니다. 이 매크로를 사용하거나 TVGN_ROOT 플래그와 함께 TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetScrollTime 트리 뷰 컨트롤의 최대 스크롤 시간을 검색합니다. 이 매크로를 사용하거나 TVM_GETSCROLLTIME 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetSelectedCount TreeView_GetSelectedCount 매크로
TreeView_GetSelection 트리 뷰 컨트롤에서 현재 선택된 항목을 검색합니다. 이 매크로를 사용하거나 TVGN_CARET 플래그와 함께 TVM_GETNEXTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetTextColor 컨트롤의 현재 텍스트 색을 검색합니다. 이 매크로를 사용하거나 TVM_GETTEXTCOLOR 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetToolTips 트리 뷰 컨트롤에서 사용하는 자식 도구 설명 컨트롤에 대한 핸들을 검색합니다. 이 매크로를 사용하거나 TVM_GETTOOLTIPS 메시지를 명시적으로 보낼 수 있습니다.
TreeView_GetUnicodeFormat 컨트롤의 유니코드 문자 형식 플래그를 검색합니다. 이 매크로를 사용하거나 TVM_GETUNICODEFORMAT 메시지를 명시적으로 보낼 수 있습니다.

TreeView_GetVisibleCount 트리 뷰 컨트롤의 클라이언트 창에서 완전히 볼 수 있는 항목 수를 가져옵니다. 이 매크로를 사용하거나 TVM_GETVISIBLECOUNT 메시지를 명시적으로 보낼 수 있습니다.
TreeView_HitTest 트리 뷰 컨트롤의 클라이언트 영역을 기준으로 지정된 지점의 위치를 결정합니다. 이 매크로를 사용하거나 TVM_HITTEST 메시지를 명시적으로 보낼 수 있습니다.
TreeView_InsertItem 트리 뷰 컨트롤에 새 항목을 삽입합니다. 이 매크로를 사용하거나 TVM_INSERTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_MapAccIDToHTREEITEM 접근성 ID를 HTREEITEM에 매핑합니다. 이 매크로를 사용하거나 TVM_MAPACCIDTOHTREEITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_MapHTREEITEMToAccID HTREEITEM을 접근성 ID에 매핑합니다. 이 매크로를 사용하거나 TVM_MAPHTREEITEMTOACCID 메시지를 명시적으로 보낼 수 있습니다.
TreeView_Select 지정된 트리 뷰 항목을 선택하거나 항목을 보기로 스크롤하거나 끌어서 놓기 작업의 대상을 나타내는 데 사용되는 스타일로 항목을 다시 그립니다.
TreeView_SelectDropTarget 끌어서 놓기 작업의 대상을 나타내는 데 사용되는 스타일에서 지정된 트리 뷰 컨트롤 항목을 다시 그립니다. 이 매크로 또는 TreeView_Select 매크로를 사용하거나 TVM_SELECTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SelectItem 지정된 트리 뷰 항목을 선택합니다. 이 매크로 또는 TreeView_Select 매크로를 사용하거나 TVM_SELECTITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SelectSetFirstVisible 트리 뷰 컨트롤을 세로로 스크롤하여 지정된 항목이 표시되는지 확인합니다.
TreeView_SetAutoScrollInfo 자동 스크롤 특성을 확인하는 데 사용되는 정보를 설정합니다. 이 매크로를 사용하거나 TVM_SETAUTOSCROLLINFO 메시지를 명시적으로 보냅니다.

TreeView_SetBkColor 컨트롤의 배경색을 설정합니다. 이 매크로를 사용하거나 TVM_SETBKCOLOR 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetBorder 트리 뷰 컨트롤의 항목에 대한 테두리 크기를 설정합니다. 이 매크로를 사용하거나 TVM_SETBORDER 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetCheckState 항목의 상태 이미지를 "checked" 또는 "unchecked"로 설정합니다. TVM_SETITEM 메시지를 직접 사용할 수도 있습니다.
TreeView_SetExtendedStyle 지정된 TreeView 컨트롤의 확장 스타일을 설정합니다. 이 매크로를 사용하거나 TVM_SETEXTENDEDSTYLE 메시지를 명시적으로 보냅니다.
TreeView_SetHot 트리 뷰 컨트롤에 대한 핫 항목을 설정합니다. 이 매크로를 사용하거나 TVM_SETHOT 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetImageList 트리 뷰 컨트롤에 대한 일반 또는 상태 이미지 목록을 설정하고 새 이미지를 사용하여 컨트롤을 다시 그립니다. 이 매크로를 사용하거나 TVM_SETIMAGELIST 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetIndent 트리 뷰 컨트롤의 들여쓰기 너비를 설정하고 새 너비를 반영하도록 컨트롤을 다시 그립니다. 이 매크로를 사용하거나 TVM_SETINDENT 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetInsertMark 트리 뷰 컨트롤에서 삽입 표시를 설정합니다. 이 매크로를 사용하거나 TVM_SETINSERTMARK 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetInsertMarkColor 트리 뷰의 삽입 표시를 그리는 데 사용되는 색을 설정합니다. 이 매크로를 사용하거나 TVM_SETINSERTMARKCOLOR 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetItem TreeView_SetItem 매크로는 트리 뷰 항목 특성의 일부 또는 전부를 설정합니다. 이 매크로를 사용하거나 TVM_SETITEM 메시지를 명시적으로 보낼 수 있습니다.

TreeView_SetItemHeight 트리 뷰 항목의 높이를 설정합니다. 이 매크로를 사용하거나 TVM_SETITEMHEIGHT 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetItemState 트리 뷰 항목의 상태 특성을 설정합니다. 이 매크로를 사용하거나 TVM_SETITEM 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetLineColor 현재 선 색을 설정합니다. TVM_SETLINECOLOR 메시지를 직접 사용할 수도 있습니다.
TreeView_SetScrollTime 트리 뷰 컨트롤의 최대 스크롤 시간을 설정합니다. 이 매크로를 사용하거나 TVM_SETSCROLLTIME 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetTextColor 컨트롤의 텍스트 색을 설정합니다. 이 매크로를 사용하거나 TVM_SETTEXTCOLOR 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetToolTips 트리 뷰 컨트롤의 자식 도구 설명 컨트롤을 설정합니다. 이 매크로를 사용하거나 TVM_SETTOOLTIPS 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SetUnicodeFormat 컨트롤의 유니코드 문자 형식 플래그를 설정합니다. (TreeView_SetUnicodeFormat)
TreeView_ShowInfoTip 트리 뷰 컨트롤에서 지정된 항목에 대한 infotip을 표시합니다. 이 매크로를 사용하거나 TVM_SHOWINFOTIP 메시지를 명시적으로 보냅니다.
TreeView_SortChildren 트리 뷰 컨트롤에서 지정된 부모 항목의 자식 항목을 정렬합니다. 이 매크로를 사용하거나 TVM_SORTCHILDREN 메시지를 명시적으로 보낼 수 있습니다.
TreeView_SortChildrenCB 항목을 비교하는 애플리케이션 정의 콜백 함수를 사용하여 트리 뷰 항목을 정렬합니다. 이 매크로를 사용하거나 TVM_SORTCHILDRENCB 메시지를 명시적으로 보낼 수 있습니다.

UninitializeFlatSB 특정 창에 대한 플랫 스크롤 막대의 초기화를 해제합니다. 지정된 창은 표준 스크롤 막대로 되돌아갑니다.

콜백 함수

PFNLVGROUPCOMPARE LVGroupCompare 함수는 LVM_INSERTGROUPSORTED 및 LVM_SORTGROUPS 메시지와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PFTASKDIALOGCALLBACK TaskDialogCallbackProc 함수는 TaskDialogIndirect 함수와 함께 사용되는 애플리케이션 정의 함수입니다.
SUBCLASSPROC RemoveWindowSubclass 및 SetWindowSubclass에서 사용하는 콜백 함수의 프로토타입을 정의합니다.

구조체

BUTTON_IMAGELIST 단추 컨트롤과 함께 사용되는 이미지 목록에 대한 정보를 포함합니다.
BUTTON_SPLITINFO 분할 단추(BS_SPLITBUTTON 및 BS_DEFSPLITBUTTON 스타일)을 정의하는 정보를 포함합니다. BCM_GETSPLITINFO 및 BCM_SETSPLITINFO 메시지와 함께 사용됩니다.
COLORMAP 비트맵의 색을 매핑하기 위해 CreateMappedBitmap 함수에서 사용하는 정보를 포함합니다.
COLORSCHEME 도구 모음 또는 Rebar의 단추 그리기에 대한 정보를 포함합니다.

COMBOBOXEXITEMA ComboBoxEx 컨트롤의 항목에 대한 정보를 포함합니다. (ANSI)
COMBOBOXEXITEMW ComboBoxEx 컨트롤의 항목에 대한 정보를 포함합니다. (유니코드)
DATETIMEPICKERINFO DTP(날짜 및 시간 선택기) 컨트롤에 대한 정보를 포함합니다.
DRAGLISTINFO 끌기 이벤트에 대한 정보를 포함합니다. DRAGLISTINFO를 가리키는 포인터는 끌기 목록 메시지의 lParam 매개 변수로 전달됩니다.
EDITBALLOONTIP 단추 컨트롤과 연결된 풍선 팁에 대한 정보를 포함합니다.
HD_TEXTFILTERA 헤더 컨트롤 텍스트 필터에 대한 정보를 포함합니다. (ANSI)
HD_TEXTFILTERW 헤더 컨트롤 텍스트 필터에 대한 정보를 포함합니다. (유니코드)
HDHITTESTINFO 적중 테스트에 대한 정보를 포함합니다. 이 구조체는 HDM_HITTEST 메시지와 함께 사용되며 HD_HITTESTINFO 구조체를 대체합니다.
HDITEMA 헤더 컨트롤의 항목에 대한 정보를 포함합니다. 이 구조체는 HD_ITEM 구조를 대체합니다. (ANSI)
HDITEMW 헤더 컨트롤의 항목에 대한 정보를 포함합니다. 이 구조체는 HD_ITEM 구조를 대체합니다. (유니코드)
HDLAYOUT 헤더 컨트롤의 크기와 위치를 설정하는 데 사용되는 정보를 포함합니다. HDLAYOUT은 HDM_LAYOUT 메시지와 함께 사용됩니다. 이 구조체는 HD_LAYOUT 구조를 대체합니다.

IMAGEINFO IMAGEINFO 구조체에는 이미지 목록의 이미지에 대한 정보가 포함됩니다. 이 구조체는 IImageList::GetImageInfo 함수와 함께 사용됩니다.
IMAGELISTDRAWPARAMS IMAGELISTDRAWPARAMS 구조체는 이미지 목록 그리기 작업에 대한 정보를 포함하며 IImageList::D raw 함수와 함께 사용됩니다.
INITCOMMONCONTROLSEX DLL(동적 연결 라이브러리)에서 공용 컨트롤 클래스를 로드하는 데 사용되는 정보를 전달합니다. 이 구조체는 InitCommonControlsEx 함수와 함께 사용됩니다.
LHITTESTINFO 지정된 위치에 해당하는 링크 관련 정보를 가져오는 데 사용됩니다.
LITEM 링크 항목에 대한 정보를 설정하고 검색하는 데 사용됩니다.
LVBKIMAGEA 목록 뷰 컨트롤의 배경 이미지에 대한 정보를 포함합니다. 이 구조체는 배경 이미지 정보를 설정하고 검색하는 데 사용됩니다. (ANSI)
LVBKIMAGEW 목록 뷰 컨트롤의 배경 이미지에 대한 정보를 포함합니다. 이 구조체는 배경 이미지 정보를 설정하고 검색하는 데 사용됩니다. (유니코드)
LVCOLUMNA 보고서 뷰의 열에 대한 정보를 포함합니다. 이 구조체는 열을 만들고 조작하는 데 모두 사용됩니다. 이 구조체는 LV_COLUMN 구조체를 대체합니다. (ANSI)
LVCOLUMNW 보고서 뷰의 열에 대한 정보를 포함합니다. 이 구조체는 열을 만들고 조작하는 데 모두 사용됩니다. 이 구조체는 LV_COLUMN 구조체를 대체합니다. (유니코드)
LVFINDINFOA 목록 뷰 항목을 검색할 때 사용되는 정보를 포함합니다. 이 구조체는 LV_FINDINFO와 동일하지만 표준 명명 규칙에 맞게 이름이 변경되었습니다. (ANSI)

LVFINDINFOW 목록 뷰 항목을 검색할 때 사용되는 정보를 포함합니다. 이 구조체는 LV_FINDINFO와 동일하지만 표준 명명 규칙에 맞게 이름이 변경되었습니다. (유니코드)
LVFOOTERINFO 목록 뷰 컨트롤의 바닥글에 대한 정보를 포함합니다.
LVFOOTERITEM 바닥글 항목에 대한 정보를 포함합니다.
LVGROUP 그룹을 설정하고 검색하는 데 사용됩니다.
LVGROUPMETRICS 목록 뷰 컨트롤의 그룹 표시에 대한 정보를 포함합니다.
LVHITTESTINFO 적중 테스트에 대한 정보를 포함합니다.
LVINSERTGROUPSORTED 그룹을 정렬하는 데 사용됩니다. LVM_INSERTGROUPSORTED 와함께 사용됩니다.
LVINSERTMARK 삽입 지점을 설명하는 데 사용됩니다.
LVITEMA 목록 뷰 항목의 특성을 지정하거나 받습니다. 이 구조체는 항목 들여쓰기를 사용하도록 설정하는 새 마스크 값(LVIF_INDENT)을 지원하도록 업데이트되었습니다. 이 구조체는 LV_ITEM 구조체를 대체합니다. (ANSI)
LVITEMINDEX 목록 뷰 항목에 대한 인덱스 정보를 포함합니다.
LVITEMW 목록 뷰 항목의 특성을 지정하거나 받습니다. 이 구조체는 항목 들여쓰기를 사용하도록 설정하는 새 마스크 값(LVIF_INDENT)을 지원하도록 업데이트되었습니다. 이 구조체는 LV_ITEM 구조체를 대체합니다. (유니코드)

LVSETINFOTIP 설정할 도구 설명 텍스트에 대한 정보를 제공합니다.
LVTILEINFO 타일 뷰에 표시될 때 목록 뷰 컨트롤의 항목에 대한 정보를 제공합니다.
LVTILEVIEWINFO 타일 뷰에 표시될 때 목록 뷰 컨트롤에 대한 정보를 제공합니다.
MCGRIDINFO 달력 컨트롤의 일부에 대한 정보를 포함합니다.
MCHITTESTINFO 월 달력 컨트롤에 대한 적중 횟수 테스트 지점과 관련된 정보를 전달합니다. 이 구조체는 MCM_HITTEST 메시지 및 해당 MonthCal_HitTest 매크로와 함께 사용됩니다.
NMBCDROPDOWN BCN_DROPDOWN 알림에 대한 정보를 포함합니다.
NMBCHOTITEM 단추 컨트롤 위의 마우스 놓기에 대한 정보를 포함합니다.
NMCBEDRAGBEGINA CBEN_DRAGBEGIN 알림 코드와 함께 사용되는 정보를 포함합니다. (ANSI)
NMCBEDRAGBEGINW CBEN_DRAGBEGIN 알림 코드와 함께 사용되는 정보를 포함합니다. (유니코드)
NMCBEENDEDITA ComboBoxEx 컨트롤 내에서 편집 작업의 결론에 대한 정보를 포함합니다. 이 구조체는 CBEN_ENDEDIT 알림 코드와 함께 사용됩니다. (ANSI)
NMCBEENDEDITW ComboBoxEx 컨트롤 내에서 편집 작업의 결론에 대한 정보를 포함합니다. 이 구조체는 CBEN_ENDEDIT 알림 코드와 함께 사용됩니다. (유니코드)

NMCHAR 문자 알림 메시지에 사용되는 정보를 포함합니다.
NMCOMBOBOXEXA 알림 코드와 함께 사용할 ComboBoxEx 항목에 특정한 정보를 포함합니다. (ANSI)
NMCOMBOBOXEXW 알림 코드와 함께 사용할 ComboBoxEx 항목에 특정한 정보를 포함합니다. (유니코드)
NMCUSTOMDRAW NM_CUSTOMDRAW 알림 코드에 특정한 정보를 포함합니다.
NMCUSTOMSPLITRECTINFO 분할 단추의 두 사각형에 대한 정보를 포함합니다. NM_GETCUSTOMSPLITRECT 알림과 함께 전송됩니다.
NMCUSTOMTEXT 사용자 지정 텍스트 알림에 사용되는 정보를 포함합니다.
NMDATETIMECHANGE DTP(날짜 및 시간 선택기) 컨트롤에서 발생한 변경 내용에 대한 정보가 포함되어 있습니다. 이 구조체는 DTN_DATETIMECHANGE 알림 코드와 함께 사용됩니다.
NMDATETIMEFORMATA DTP(날짜 및 시간 선택기) 컨트롤 내에서 콜백 필드를 정의하는 형식 문자열의 일부에 대한 정보가 포함되어 있습니다. (ANSI)
NMDATETIMEFORMATQUERYA DTP(날짜 및 시간 선택기) 컨트롤 콜백 필드에 대한 정보가 포함되어 있습니다. (ANSI)
NMDATETIMEFORMATQUERYW DTP(날짜 및 시간 선택기) 컨트롤 콜백 필드에 대한 정보가 포함되어 있습니다. (유니코드)
NMDATETIMEFORMATW DTP(날짜 및 시간 선택기) 컨트롤 내에서 콜백 필드를 정의하는 형식 문자열의 일부에 대한 정보가 포함되어 있습니다. (유니코드)

NMDATETIMESTRINGA DTP(날짜 및 시간 선택기) 컨트롤에서 수행된 편집 작업과 관련된 정보가 포함되어 있습니다. 이 메시지는 DTN_USERSTRING 알림 코드와 함께 사용됩니다. (ANSI)
NMDATETIMESTRINGW DTP(날짜 및 시간 선택기) 컨트롤에서 수행된 편집 작업과 관련된 정보가 포함되어 있습니다. 이 메시지는 DTN_USERSTRING 알림 코드와 함께 사용됩니다. (유니코드)
NMDATETIMEWMKEYDOWNA DTN_WMKEYDOWN 알림 코드를 설명하고 처리하는 데 사용되는 정보를 전달합니다. (ANSI)
NMDATETIMEWMKEYDOWNW DTN_WMKEYDOWN 알림 코드를 설명하고 처리하는 데 사용되는 정보를 전달합니다. (유니코드)
NMDAYSTATE MCN_GETDAYSTATE 알림 코드를 처리하는 데 필요한 정보를 전달합니다. 이 구조체의 모든 멤버는 MCN_GETDAYSTATE를 처리할 때 수신 애플리케이션이 설정해야 하는 prgDayState를 제외하고 입력용입니다.
NMHDDISPINFOA HDN_GETDISPINFO 알림 코드를 처리하는 데 사용되는 정보를 포함합니다. (ANSI)
NMHDDISPINFOW HDN_GETDISPINFO 알림 코드를 처리하는 데 사용되는 정보를 포함합니다. (유니코드)
NMHDFILTERBTNCLICK 필터 단추 클릭의 특성을 지정하거나 받습니다.
NMHEADERA 헤더 컨트롤 알림 메시지에 대한 정보를 포함합니다. 이 구조체는 HD_NOTIFY 구조를 대체합니다. (ANSI)
NMHEADERW 헤더 컨트롤 알림 메시지에 대한 정보를 포함합니다. 이 구조체는 HD_NOTIFY 구조를 대체합니다. (유니코드)
NMIPADDRESS IPN_FIELDCHANGED 알림 코드의 정보를 포함합니다.

NMITEMACTIVATE LVN_ITEMACTIVATE 알림 코드에 대한 정보를 포함합니다.
NMKEY 주요 알림 메시지와 함께 사용되는 정보를 포함합니다.
NMLINK NMLINK에는 알림 정보가 포함됩니다. NM_CLICK 또는 NM_RETURN 메시지와 함께 이 구조체를 보냅니다.
NMLISTVIEW 목록 보기 알림 메시지에 대한 정보를 포함합니다. 이 구조체는 NM_LISTVIEW 구조체와 동일하지만 표준 명명 규칙에 맞게 이름이 변경되었습니다.
NMLVCACHEHINT 가상 목록 뷰와 함께 사용할 캐시된 항목 정보를 업데이트하는 데 사용되는 정보를 포함합니다.
NMLVCUSTOMDRAW 목록 뷰 컨트롤에서 보낸 NM_CUSTOMDRAW(목록 뷰) 알림 코드와 관련된 정보를 포함합니다.
NMLVDISPINFOA LVN_GETDISPINFO 또는 LVN_SETDISPINFO 알림 코드에 대한 정보를 포함합니다. 이 구조체는 LV_DISPINFO 구조체와 동일하지만 표준 명명 규칙에 맞게 이름이 변경되었습니다. (ANSI)
NMLVDISPINFOW LVN_GETDISPINFO 또는 LVN_SETDISPINFO 알림 코드에 대한 정보를 포함합니다. 이 구조체는 LV_DISPINFO 구조체와 동일하지만 표준 명명 규칙에 맞게 이름이 변경되었습니다. (유니코드)
NMLVEMPTYMARKUP LVN_GETEMPTYMARKUP 알림 코드와 함께 사용되는 정보를 포함합니다.
NMLVFINDITEMA 소유자가 가상 목록 보기 컨트롤에서 요청한 항목을 찾는 데 필요한 정보를 포함합니다. 이 구조체는 LVN_ODFINDITEM 알림 코드와 함께 사용됩니다. (ANSI)
NMLVFINDITEMW 소유자가 가상 목록 보기 컨트롤에서 요청한 항목을 찾는 데 필요한 정보를 포함합니다. 이 구조체는 LVN_ODFINDITEM 알림 코드와 함께 사용됩니다. (유니코드)

NMLVGETINFOTIPA 항목에 대한 도구 설명을 표시하는 데 필요한 목록 뷰 항목 정보를 포함하고 받습니다. 이 구조체는 LVN_GETINFOTIP 알림 코드와 함께 사용됩니다. (ANSI)
NMLVGETINFOTIPW 항목에 대한 도구 설명을 표시하는 데 필요한 목록 뷰 항목 정보를 포함하고 받습니다. 이 구조체는 LVN_GETINFOTIP 알림 코드와 함께 사용됩니다. (유니코드)
NMLVKEYDOWN LVN_KEYDOWN 알림 코드를 처리하는 데 사용되는 정보를 포함합니다. 이 구조체는 NMLVKEYDOWN 구조체와 동일하지만 표준 명명 규칙에 맞게 이름이 변경되었습니다.
NMLVLINK LVN_LINKCLICK 알림 코드에 대한 정보를 포함합니다.
NMLVODSTATECHANGE LVN_ODSTATECHANGED 알림 코드를 처리하는 데 사용할 정보를 포함하는 구조체입니다.
NMLVSCROLL 스크롤 작업에 대한 정보를 제공합니다.
NMMOUSE 마우스 알림 메시지에 사용되는 정보를 포함합니다.
NMOBJECTNOTIFY TBN_GETOBJECT, TCN_GETOBJECT 및 PSN_GETOBJECT 알림 코드와 함께 사용되는 정보를 포함합니다.
NMPGCALCSIZE Pager 컨트롤이 포함된 창의 스크롤 가능 영역을 계산하는 데 사용하는 정보를 포함하고 받습니다. PGN_CALCSIZE 알림과 함께 사용됩니다.
NMPGHOTITEM PGN_HOTITEMCHANGE 알림 코드와 함께 사용되는 정보를 포함합니다.
NMPGSCROLL 포함된 창을 스크롤할 때 Pager 컨트롤이 사용하는 정보를 포함하고 받습니다. PGN_SCROLL 알림과 함께 사용됩니다.

NMRBAUTOSIZE RBN_AUTOSIZE 알림 코드를 처리하는 데 사용되는 정보를 포함합니다.
NMREBAR 다양한 철근 알림을 처리하는 데 사용되는 정보를 포함합니다.
NMREBARAUTOBREAK RBN_AUTOBREAK 알림 코드와 함께 사용되는 정보를 포함합니다.
NMREBARCHEVRON RBN_CHEVRONPUSHED 알림 코드를 처리하는 데 사용되는 정보를 포함합니다.
NMREBARCHILDSIZE RBN_CHILDSIZE 알림 코드를 처리하는 데 사용되는 정보를 포함합니다.
NMREBARSPLITTER RBN_SPLITTERDRAG 알림 코드를 처리하는 데 사용되는 정보를 포함합니다.
NMSEARCHWEB EN_SEARCHWEB 알림 코드를 처리하는 데 사용되는 정보를 포함합니다.
NMSELCHANGE MCN_SELCHANGE 알림 코드를 처리하는 데 필요한 정보를 전달합니다.
NMTBCUSTOMDRAW 도구 모음 컨트롤에서 보낸 NM_CUSTOMDRAW 알림 코드와 관련된 정보를 포함합니다.
NMTBDISPINFOA 도구 모음 항목의 표시 정보를 포함하고 받습니다. 이 구조체는 TBN_GETDISPINFO 알림 코드와 함께 사용됩니다. (ANSI)
NMTBDISPINFOW 도구 모음 항목의 표시 정보를 포함하고 받습니다. 이 구조체는 TBN_GETDISPINFO 알림 코드와 함께 사용됩니다. (유니코드)

NMTBGETINFOTIPA 도구 모음 항목의 정보 팁 정보를 포함하고 받습니다. 이 구조체는 TBN_GETINFOTIP 알림 코드와 함께 사용됩니다. (ANSI)
NMTBGETINFOTIPW 도구 모음 항목의 정보 팁 정보를 포함하고 받습니다. 이 구조체는 TBN_GETINFOTIP 알림 코드와 함께 사용됩니다. (유니코드)
NMTBHOTITEM TBN_HOTITEMCHANGE 알림 코드와 함께 사용되는 정보를 포함합니다.
NMTBRESTORE 도구 모음 상태가 저장되었을 때 애플리케이션이 NMTBSAVE에 배치된 정보를 추출할 수 있습니다. TBN_RESTORE 알림 코드를 받으면 이 구조체가 애플리케이션에 전달됩니다.
NMTBSAVE TBN_SAVE 알림 코드를 받으면 이 구조체가 애플리케이션에 전달됩니다. 여기에는 현재 저장 중인 단추에 대한 정보가 있습니다. 애플리케이션은 멤버 값을 수정하여 추가 정보를 저장할 수 있습니다.
NMTCKEYDOWN 탭 컨트롤의 키 누름에 대한 정보를 포함합니다. TCN_KEYDOWN 알림 코드와 함께 사용됩니다. 이 구조체는 TC_KEYDOWN 구조체를 대체합니다.
NMTOOLBARA 도구 모음 알림 코드를 처리하는 데 사용되는 정보를 포함합니다. 이 구조체는 TBNOTIFY 구조체를 대체합니다. (ANSI)
NMTOOLBARW 도구 모음 알림 코드를 처리하는 데 사용되는 정보를 포함합니다. 이 구조체는 TBNOTIFY 구조체를 대체합니다. (유니코드)
NMTOOLTIPSCREATED NM_TOOLTIPSCREATED 알림 코드와 함께 사용되는 정보를 포함합니다.
NMTRBTHUMBPOSCHANGING 트랙바 변경 알림에 대한 정보를 포함합니다. 이 메시지는 TRBN_THUMBPOSCHANGING 알림과 함께 전송됩니다.

NMTREEVIEWA 트리 뷰 알림 메시지에 대한 정보를 포함합니다. 이 구조체는 NM_TREEVIEW 구조체와 동일하지만 현재 명명 규칙을 따르도록 이름이 변경되었습니다. (ANSI)
NMTREEVIEWW 트리 뷰 알림 메시지에 대한 정보를 포함합니다. 이 구조체는 NM_TREEVIEW 구조체와 동일하지만 현재 명명 규칙을 따르도록 이름이 변경되었습니다. (유니코드)
NMTTCUSTOMDRAW 도구 설명 컨트롤에서 보낸 NM_CUSTOMDRAW 알림 코드에 특정한 정보를 포함합니다.
NMTTDISPINFOA TTN_GETDISPINFO 알림 코드를 처리하는 데 사용되는 정보를 포함합니다. 이 구조체는 TOOLTIPTEXT 구조체를 대체합니다. (ANSI)
NMTTDISPINFOW TTN_GETDISPINFO 알림 코드를 처리하는 데 사용되는 정보를 포함합니다. 이 구조체는 TOOLTIPTEXT 구조체를 대체합니다. (유니코드)
NMTVASYNCDRAW 아이콘 또는 오버레이 트리 항목 그리기 실패 이유에 대한 설명이 포함되어 있습니다.
NMTVCUSTOMDRAW 트리 뷰 컨트롤에서 보낸 NM_CUSTOMDRAW(트리 뷰) 알림 코드와 관련된 정보를 포함합니다.
NMTVDISPINFOA 트리 뷰 항목에 대한 표시 정보를 포함하고 받습니다. 이 구조체는 TV_DISPINFO 구조체와 동일하지만 현재 명명 규칙을 따르도록 이름이 변경되었습니다. (ANSI)
NMTVDISPINFOEXA 확장된 TreeView 알림 정보와 관련된 정보를 포함합니다. (ANSI)
NMTVDISPINFOEXW 확장된 TreeView 알림 정보와 관련된 정보를 포함합니다. (유니코드)
NMTVDISPINFOW 트리 뷰 항목에 대한 표시 정보를 포함하고 받습니다. 이 구조체는 TV_DISPINFO 구조체와 동일하지만 현재 명명 규칙을 따르도록 이름이 변경되었습니다. (유니코드)

NMTVGETINFOTIPA 항목에 대한 도구 설명을 표시하는 데 필요한 트리 뷰 항목 정보를 포함하고 받습니다. 이 구조체는 TVN_GETINFOTIP 알림 코드와 함께 사용됩니다. (ANSI)
NMTVGETINFOTIPW 항목에 대한 도구 설명을 표시하는 데 필요한 트리 뷰 항목 정보를 포함하고 받습니다. 이 구조체는 TVN_GETINFOTIP 알림 코드와 함께 사용됩니다. (유니코드)
NMTVITEMCHANGE 트리 뷰 항목 변경에 대한 정보를 포함합니다. 이 구조체는 TVN_ITEMCHANGED 및 TVN_ITEMCHANGING 알림과 함께 전송됩니다.
NMTVKEYDOWN 트리 뷰 컨트롤의 키보드 이벤트에 대한 정보를 포함합니다. 이 구조체는 TVN_KEYDOWN 알림 코드와 함께 사용됩니다. 구조체는 TV_KEYDOWN 구조체와 동일하지만 현재 명명 규칙을 따르도록 이름이 변경되었습니다.
NMTVSTATEIMAGECHANGING NM_TVSTATEIMAGECHANGING 알림 코드에 대한 정보를 포함합니다.
NMUPDOWN 업다운 제어 알림 메시지와 관련된 정보를 포함합니다. NM_UPDOWN 구조체와 동일하며 이를 대체합니다.
NMVIEWCHANGE MCN_VIEWCHANGE 알림 코드를 처리하는 데 필요한 정보를 저장합니다.
PBRANGE 진행률 표시줄 컨트롤의 상한 및 하한에 대한 정보를 포함합니다. 이 구조체는 PBM_GETRANGE 메시지와 함께 사용됩니다.
RBHITTESTINFO 적중 테스트 작업과 관련된 정보를 포함합니다. 이 구조체는 RB_HITTEST 메시지와 함께 사용됩니다.
REBARBANDINFOA rebar 컨트롤에서 밴드를 정의하는 정보를 포함합니다. (ANSI)

REBARBANDINFOW rebar 컨트롤에서 밴드를 정의하는 정보를 포함합니다. (유니코드)
REBARINFO rebar 컨트롤 특성을 설명하는 정보를 포함합니다.
TASKDIALOG_BUTTON TASKDIALOG_BUTTON 구조에는 작업 대화 상자에 단추를 표시하는 데 사용되는 정보가 포함되어 있습니다. TASKDIALOGCONFIG 구조체에서 이 구조체를 사용합니다.
TASKDIALOGCONFIG TASKDIALOGCONFIG 구조에는 작업 대화 상자를 표시하는 데 사용되는 정보가 포함되어 있습니다. TaskDialogIndirect 함수에서 이 구조체를 사용합니다.
TBADDBITMAP 단추 이미지가 포함된 비트맵을 도구 모음에 추가합니다.
TBBUTTON 도구 모음의 단추에 대한 정보를 포함합니다.
TBBUTTONINFOA 도구 모음의 특정 단추에 대한 정보를 포함하거나 받습니다. (ANSI)
TBBUTTONINFOW 도구 모음의 특정 단추에 대한 정보를 포함하거나 받습니다. (유니코드)
TBINSERTMARK 도구 모음 컨트롤의 삽입 표시에 대한 정보를 포함합니다.
TBMETRICS 도구 모음 항목을 축소하거나 확장하는 데 사용되는 도구 모음의 메트릭을 정의합니다.
TBREPLACEBITMAP TB_REPLACEBITMAP 메시지와 함께 사용되어 도구 모음 비트맵을 다른 도구 모음 비트맵으로 바꿉니다.

TBSAVEPARAMSA TB_SAVERESTORE 메시지가 저장되는 레지스트리 위치를 지정하고 도구 모음 상태에 대한 정보를 검색합니다. (ANSI)
TBSAVEPARAMSW TB_SAVERESTORE 메시지가 저장되는 레지스트리 위치를 지정하고 도구 모음 상태에 대한 정보를 검색합니다. (유니코드)
TCHITTESTINFO 적중 테스트에 대한 정보를 포함합니다. 이 구조체는 TC_HITTESTINFO 구조체를 대체합니다.
TCITEMA 탭 항목의 특성을 지정하거나 받습니다. TCM_INSERTITEM, TCM_GETITEM 및 TCM_SETITEM 메시지와 함께 사용됩니다. 이 구조체는 TC_ITEM 구조체를 대체합니다. (ANSI)
TCITEMHEADERA 탭의 특성을 지정하거나 수신합니다. TCM_INSERTITEM, TCM_GETITEM 및 TCM_SETITEM 메시지와 함께 사용됩니다. 이 구조체는 TC_ITEMHEADER 구조체를 대체합니다. (ANSI)
TCITEMHEADERW 탭의 특성을 지정하거나 수신합니다. TCM_INSERTITEM, TCM_GETITEM 및 TCM_SETITEM 메시지와 함께 사용됩니다. 이 구조체는 TC_ITEMHEADER 구조체를 대체합니다. (유니코드)
TCITEMW 탭 항목의 특성을 지정하거나 받습니다. TCM_INSERTITEM, TCM_GETITEM 및 TCM_SETITEM 메시지와 함께 사용됩니다. 이 구조체는 TC_ITEM 구조체를 대체합니다. (유니코드)
TTGETTITLE 도구 설명 컨트롤의 제목에 대한 정보를 제공합니다.
TTHITTESTINFOA 도구 설명 컨트롤에서 점이 지정된 도구의 경계 사각형에 있는지 여부를 확인하는 데 사용하는 정보를 포함합니다. 점이 사각형에 있으면 구조체가 도구에 대한 정보를 받습니다. (ANSI)
TTHITTESTINFOW 도구 설명 컨트롤에서 점이 지정된 도구의 경계 사각형에 있는지 여부를 확인하는 데 사용하는 정보를 포함합니다. 점이 사각형에 있으면 구조체가 도구에 대한 정보를 받습니다. (유니코드)

TTTOOLINFOA TOOLINFO 구조체는 도구 설명 컨트롤의 도구에 대한 정보를 포함합니다. (ANSI)
TTTOOLINFOW TOOLINFO 구조체는 도구 설명 컨트롤의 도구에 대한 정보를 포함합니다. (유니코드)
TVGETITEMPARTRECTINFO 트리 항목의 지정된 부분에 대한 "적중 영역"을 식별하기 위한 정보를 포함합니다. 구조체는 TVM_GETITEMPARTRECT 메시지 및 TreeView_GetItemPartRect 매크로와 함께 사용됩니다.
TVHITTESTINFO 트리 뷰 컨트롤을 기준으로 지점의 위치를 결정하는 데 사용되는 정보를 포함합니다.
TVINSERTSTRUCTA 트리 뷰 컨트롤에 새 항목을 추가하는 데 사용되는 정보를 포함합니다. 이 구조체는 TVM_INSERTITEM 메시지와 함께 사용됩니다. 구조체는 TV_INSERTSTRUCT 구조체와 동일하지만 현재 명명 규칙을 따르도록 이름이 변경되었습니다. (ANSI)
TVINSERTSTRUCTW 트리 뷰 컨트롤에 새 항목을 추가하는 데 사용되는 정보를 포함합니다. 이 구조체는 TVM_INSERTITEM 메시지와 함께 사용됩니다. 구조체는 TV_INSERTSTRUCT 구조체와 동일하지만 현재 명명 규칙을 따르도록 이름이 변경되었습니다. (유니코드)
TVITEMA 트리 뷰 항목의 특성을 지정하거나 받습니다. 이 구조체는 TV_ITEM 구조체와 동일하지만 현재 명명 규칙을 따르도록 이름이 변경되었습니다. 새 애플리케이션은 이 구조체를 사용해야 합니다. (ANSI)
TVITEMEXA 트리 뷰 항목의 특성을 지정하거나 받습니다. 이 구조체는 TVITEM 구조체의 향상된 기능입니다. 새 애플리케이션은 적절한 경우 이 구조체를 사용해야 합니다. (ANSI)
TVITEMEXW 트리 뷰 항목의 특성을 지정하거나 받습니다. 이 구조체는 TVITEM 구조체의 향상된 기능입니다. 새 애플리케이션은 적절한 경우 이 구조체를 사용해야 합니다. (유니코드)
TVITEMW 트리 뷰 항목의 특성을 지정하거나 받습니다. 이 구조체는 TV_ITEM 구조체와 동일하지만 현재 명명 규칙을 따르도록 이름이 변경되었습니다. 새 애플리케이션은 이 구조체를 사용해야 합니다. (유니코드)

TVSORTCB 트리 뷰 컨트롤에서 자식 항목을 정렬하는 데 사용되는 정보를 포함합니다. 이 구조체는 TVM_SORTCHILDRENCB 메시지와 함께 사용됩니다. 이 구조체는 TV_SORTCB 구조체와 동일하지만 현재 명명 규칙을 따르도록 이름이 변경되었습니다.
UDACCEL 업다운 컨트롤에 대한 가속 정보가 포함됩니다.

열거형

EC_ENDOFLINE 편집 컨트롤에서 사용하는 줄 문자의 끝을 나타냅니다.
EC_SEARCHWEB_ENTRYPOINT 웹 검색의 진입점을 나타내는 상수를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

commdlg.h 헤더

아티클2023. 03. 14.

이 헤더는 대화 상자에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 대화 상자

commdlg.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IPrintDialogCallback Print 속성 시트가 표시되는 동안 애플리케이션이 PrintDlgEx 함수에서 알림 및 메시지를 받을 수 있도록 하는 메서드를 제공합니다.
IPrintDialogServices PrintDlgEx 함수를 사용하여 애플리케이션에서 현재 선택된 프린터에 대한 정보를 검색할 수 있도록 하는 메서드를 제공합니다.

Functions

CDSIZEOF_STRUCT 지정된 멤버까지 구조체의 크기를 가져옵니다.
CommDlg_OpenSave_GetFilePathA ASCII 버전의 CommDlg_OpenSave_GetFilePath
CommDlg_OpenSave_GetFilePathW CommDlg_OpenSave_GetFilePath 와이드 문자열 버전
CommDlg_OpenSave_GetFolderIDList 탐색기 스타일 열기 또는 다른 이름으로 저장 대화 상자가 현재 열려 있는 폴더에 해당하는 항목 식별자 목록의 주소를 검색합니다.

CommDlg_OpenSave_GetFolderPathA ASCII 버전의 CommDlg_OpenSave_GetFolderPath
CommDlg_OpenSave_GetFolderPathW CommDlg_OpenSave_GetFolderPath 와이드 문자열 버전
CommDlg_OpenSave_GetSpecA ASCII 버전의 CommDlg_OpenSave_GetSpec
CommDlg_OpenSave_GetSpecW CommDlg_OpenSave_GetSpec 와이드 문자열 버전
CommDlg_OpenSave_HideControl 탐색기 스타일 열기 또는 다른 이름으로 저장 대화 상자에서 지정된 컨트롤을 숨깁니다.
CommDlg_OpenSave_SetControlText 탐색기 스타일 열기 또는 다른 이름으로 저장 대화 상자에서 지정된 컨트롤의 텍스트를 설정합니다.
CommDlg_OpenSave_SetDefExt 탐색기 스타일 열기 또는 다른 이름으로 저장 대화 상자의 기본 파일 이름 확장명을 설정합니다.
CommDlgExtendedError CommDlgExtendedError 함수(commdlg.h)는 일반적인 대화 상자 오류 코드를 반환합니다.
FindTextA 사용자가 검색할 문자열과 문서에서 텍스트를 검색할 때 사용할 옵션을 지정할 수 있는 시스템 정의 모덜리스 찾기 대화 상자를 만듭니다. (ANSI)
FindTextW 사용자가 검색할 문자열과 문서에서 텍스트를 검색할 때 사용할 옵션을 지정할 수 있는 시스템 정의 모덜리스 찾기 대화 상자를 만듭니다. (유니코드)
GetFileTitleA 지정된 파일의 이름을 검색합니다. (ANSI)

GetFileTitleW 지정된 파일의 이름을 검색합니다. (유니코드)
GetOpenFileNameA 사용자가 열 파일 또는 파일 집합의 드라이브, 디렉터리 및 이름을 지정할 수 있는 열기 대화 상자를 만듭니다. (ANSI)
GetOpenFileNameW 사용자가 열 파일 또는 파일 집합의 드라이브, 디렉터리 및 이름을 지정할 수 있는 열기 대화 상자를 만듭니다. (유니코드)
GetSaveFileNameA 사용자가 저장할 파일의 드라이브, 디렉터리 및 이름을 지정할 수 있는 저장 대화 상자를 만듭니다. (ANSI)
GetSaveFileNameW 사용자가 저장할 파일의 드라이브, 디렉터리 및 이름을 지정할 수 있는 저장 대화 상자를 만듭니다. (유니코드)
ReplaceTextA 사용자가 검색할 문자열과 대체 문자열을 지정할 수 있는 시스템 정의 모덜리스 대화 상자와 찾기 및 바꾸기 작업을 제어하는 옵션을 만듭니다. (ANSI)
ReplaceTextW 사용자가 검색할 문자열과 대체 문자열을 지정할 수 있는 시스템 정의 모덜리스 대화 상자와 찾기 및 바꾸기 작업을 제어하는 옵션을 만듭니다. (유니코드)

콜백 함수

ColorW 선택 ChooseColor의 와이드 문자열 버전
선택FontA ChooseFont의 ASCII 버전

ChooseFontW ChooseFont의 와이드 문자열 버전
LPCCHOOKPROC 색 대화 상자의 기본 대화 상자 프로시저에 대한 메시지 또는 알림을 받습니다. ChooseColor 함수와 함께 사용되는 애플리케이션 정의 또는 라이브러리 정의 콜백 함수입니다.
LPCFHOOKPROC 글꼴 대화 상자의 기본 대화 상자 프로시저에 대한 메시지 또는 알림을 받습니다. ChooseFont 함수와 함께 사용되는 애플리케이션 정의 또는 라이브러리 정의 콜백 프로시저입니다.
LPFRHOOKPROC 찾기 또는 바꾸기 대화 상자의 기본 대화 상자 프로시저에 대한 메시지 또는 알림을 받습니다.
LPOFNHOOKPROC 대화 상자에서 보낸 알림 메시지를 받습니다.
LPPAGEPAINTHOOK 페이지 설정 대화 상자에서 샘플 페이지의 그리기를 사용자 지정할 수 있는 메시지를 받습니다. PagePaintHook 후크 프로시저는 PageSetupDlg 함수와 함께 사용되는 애플리케이션 정의 또는 라이브러리 정의 콜백 함수입니다.
LPPAGESETUPHOOK 페이지 설정 대화 상자의 기본 대화 상자 프로시저에 대한 메시지 또는 알림을 받습니다. PageSetupHook 후크 프로시저는 PageSetupDlg 함수와 함께 사용되는 애플리케이션 정의 또는 라이브러리 정의 콜백 함수입니다.
LPPRINTHOOKPROC 인쇄 대화 상자의 기본 대화 상자 프로시저에 대한 메시지 또는 알림을 받습니다. PrintDlg 함수와 함께 사용되는 애플리케이션 정의 또는 라이브러리 정의 콜백 함수입니다.
LPSETUPHOOKPROC PrintDlg 함수와 함께 사용되는 애플리케이션 정의 또는 라이브러리 정의 콜백 함수입니다. 후크 프로시저는 설치 프로그램 인쇄 대화 상자의 기본 대화 상자 프로시저에 대한 메시지 또는 알림을 받습니다.
PageSetupDlgA PageSetupDlg의 ASCII 버전

PageSetupDlgW PageSetupDlg의 와이드 문자열 버전
PrintDlgA PrintDlg의 ASCII 버전
PrintDlgExA PrintDlgEx의 ASCII 버전
PrintDlgExW PrintDlgEx의 와이드 문자열 버전
PrintDlgW PrintDlg의 와이드 문자열 버전

구조체

CHOOSECOLORA CHOOSECOLORA(ANSI) 구조체(commdlg.h)에는 ChooseColor 함수가 색 대화 상자를 초기화하는 데 사용하는 정보가 포함되어 있습니다.
CHOOSECOLORA CHOOSECOLORA(ANSI) 구조체 r1(commdlg.h)에는 ChooseColor 함수가 색 대화 상자를 초기화하는 데 사용하는 정보가 포함되어 있습니다.
CHOOSECOLORW CHOOSECOLORW(유니코드) 구조체(commdlg.h)에는 ChooseColor 함수가 색 대화 상자를 초기화하는 데 사용하는 정보가 포함되어 있습니다.
CHOOSECOLORW CHOOSECOLORW(유니코드) 구조체 r1(commdlg.h)에는 ChooseColor 함수가 색 대화 상자를 초기화하는 데 사용하는 정보가 포함되어 있습니다.

CHOOSEFONTA ChooseFont 함수가 글꼴 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. 사용자가 대화 상자를 닫은 후 시스템은 이 구조에서 사용자의 선택 항목에 대한 정보를 반환합니다. (ANSI)
CHOOSEFONTW ChooseFont 함수가 글꼴 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. 사용자가 대화 상자를 닫은 후 시스템은 이 구조에서 사용자의 선택 항목에 대한 정보를 반환합니다. (유니코드)
DEVNAMES 프린터의 드라이버, 디바이스 및 출력 포트 이름을 식별하는 문자열을 포함합니다.
FINDREPLACEA FindText 및 ReplaceText 함수가 찾기 및 바꾸기 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. (ANSI)
FINDREPLACEW FindText 및 ReplaceText 함수가 찾기 및 바꾸기 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. (유니코드)
OFNOTIFYA 열기 또는 다른 이름으로 저장 대화 상자의 OFNHookProc 후크 프로시저로 전송된 WM_NOTIFY 메시지에 대한 정보를 포함합니다. WM_NOTIFY 메시지의 lParam 매개 변수는 OFNOTIFY 구조체에 대한 포인터입니다. (ANSI)
OFNOTIFYEXA CDN_INCLUDEITEM 알림 메시지에 대한 정보를 포함합니다. (ANSI)
OFNOTIFYEXW CDN_INCLUDEITEM 알림 메시지에 대한 정보를 포함합니다. (유니코드)
OFNOTIFYW 열기 또는 다른 이름으로 저장 대화 상자의 OFNHookProc 후크 프로시저로 전송된 WM_NOTIFY 메시지에 대한 정보를 포함합니다. WM_NOTIFY 메시지의 lParam 매개 변수는 OFNOTIFY 구조체에 대한 포인터입니다. (유니코드)
OPENFILENAME_NT4A OPENFILENAME_NT4 구조는 _WIN32_WINNT 0x0400 설정된 OPENFILENAME과 동일합니다. (ANSI)

OPENFILENAME_NT4W OPENFILENAME_NT4 구조는 _WIN32_WINNT 0x0400 설정된 OPENFILENAME과 동일합니다. (유니코드)
OPENFILENAMEA GetOpenFileName 및 GetSaveFileName 함수가 열기 또는 다른 이름으로 저장 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. 사용자가 대화 상자를 닫은 후 시스템은 이 구조에서 사용자의 선택 항목에 대한 정보를 반환합니다. (ANSI)
OPENFILENAMEW GetOpenFileName 및 GetSaveFileName 함수가 열기 또는 다른 이름으로 저장 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. 사용자가 대화 상자를 닫은 후 시스템은 이 구조에서 사용자의 선택 항목에 대한 정보를 반환합니다. (유니코드)
PAGESETUPDLGA PageSetupDlg 함수가 페이지 설정 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. 사용자가 대화 상자를 닫은 후 시스템은 이 구조의 사용자 정의 페이지 매개 변수에 대한 정보를 반환합니다. (ANSI)
PAGESETUPDLGW PageSetupDlg 함수가 페이지 설정 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. 사용자가 대화 상자를 닫은 후 시스템은 이 구조의 사용자 정의 페이지 매개 변수에 대한 정보를 반환합니다. (유니코드)
PRINTDLGA PrintDlg 함수가 인쇄 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. 사용자가 대화 상자를 닫은 후 시스템은 이 구조를 사용하여 사용자의 선택 항목에 대한 정보를 반환합니다. (ANSI)
PRINTDLGEXA PrintDlgEx 함수가 Print 속성 시트를 초기화하는 데 사용하는 정보를 포함합니다. 사용자가 속성 시트를 닫은 후 시스템은 이 구조를 사용하여 사용자의 선택 항목에 대한 정보를 반환합니다. (ANSI)
PRINTDLGEXW PrintDlgEx 함수가 Print 속성 시트를 초기화하는 데 사용하는 정보를 포함합니다. 사용자가 속성 시트를 닫은 후 시스템은 이 구조를 사용하여 사용자의 선택 항목에 대한 정보를 반환합니다. (유니코드)

PRINTDLGW PrintDlg 함수가 인쇄 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. 사용자가 대화 상자를 닫은 후 시스템은 이 구조를 사용하여 사용자의 선택 항목에 대한 정보를 반환합니다. (유니코드)
PRINTPAGERANGE 인쇄 작업의 페이지 범위를 나타냅니다. 인쇄 작업에는 둘 이상의 페이지 범위가 있을 수 있습니다. 이 정보는 PrintDlgEx 함수를 호출할 때 PRINTDLGEX 구조체에 제공됩니다.

commoncontrols.h 헤더

아티클2023. 06. 08.

이 헤더는 Windows 컨트롤에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 컨트롤

commoncontrols.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IImageList 이미지 목록으로 조작하고 상호 작용하는 메서드를 공개합니다.
IImageList2 이미지 목록을 조작하고 상호 작용하기 위한 추가 메서드를 제공하여 IImageList를 확장합니다.

Functions

ImageList_CoCreateInstance 이미지 목록의 단일 인스턴스를 만들고 해당 인스턴스에 대한 인터페이스 포인터를 반환합니다.

구조체

IMAGEINFO IMAGEINFO 구조체는 이미지 목록의 이미지에 대한 정보를 포함하며 IImageList::GetImageInfo 함수와 함께 사용됩니다.
IMAGELISTDRAWPARAMS IMAGELISTDRAWPARAMS 구조체에는 이미지 목록 그리기 작업에 대한 정보가 포함되어 있습니다. 이 구조체는 IImageList::D raw 함수와 함께 사용됩니다.

IMAGELISTSTATS 이미지 목록 통계를 포함합니다. GetStatistics에서 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

comppkgsup.h 헤더

아티클2024. 02. 24.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

comppkgsup.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

AreDvdCodecsEnabled 현재 장치에서 DVD 코덱을 사용할 수 있는지 여부를 나타내는 값을 반환합니다.
GetDefaultContentDecryptionModuleFactory Windows에 기본 제공되는 지정된 키 시스템에 대한 IMFContentDecryptionModuleFactory 구현을 반환합니다.
GetMediaComponentPackageInfo 지정된 요구 사항을 충족하는 시스템에 설치된 모든 미디어 코덱의 속성 목록을 반환합니다.
GetMediaExtensionCommunicationFactory 미디어 확장을 등록하기 위한 통신 팩터리를 만듭니다.
GetServerForPMP RegisterServerForPMP에 대한 이전 호출과 함께 PMP(보호된 미디어 프로세스) 사용에 대해 등록된 COM 서버를 가져옵니다.
InstantiateComponentFromPackage 애플리케이션 패키지에 클래스의 instance 만듭니다.
IsMediaBehaviorEnabled 지정된 GUID와 연결된 미디어 동작을 사용할 수 있는지 여부를 나타내는 값을 가져옵니다.
RegisterMediaExtensionPackage

미디어 확장을 현재 사용자의 지정된 PFN(패키지 패밀리 이름)에 등록합니다.
RegisterServerForPMP COM Server CLSID 및 PMP(Protected Media Process) 사용을 위한 클래스 팩터리를 등록합니다.
RequireNetworkDuringMediaTaskCompletion 미디어 작업 완료에 필요한 네트워크 연결 수를 증가 또는 감소합니다.
UnregisterServerForPMP 이전에 PMP(보호된 미디어 프로세스) 사용에 대해 등록된 COM 서버 CLSID 및 클래스 팩터리를 등록합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

compressapi.h 헤더

아티클2023. 08. 22.

이 헤더는 압축 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 압축 API

compressapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CloseCompressor 를 호출하여 열린 COMPRESSOR_HANDLE 닫습니다.
CloseDecompressor 를 호출하여 열린 DECOMPRESSOR_HANDLE 닫습니다.
압축 정보 블록을 가져와서 압축합니다.
CreateCompressor 새 COMPRESSOR_HANDLE 생성합니다.
CreateDecompressor 새 DECOMPRESSOR_HANDLE 생성합니다.
압축 압축된 정보 블록을 가져와 압축을 풉니다.
QueryCompressorInformation 압축기에서 특정 압축 알고리즘에 대한 정보를 쿼리합니다.
QueryDecompressorInformation 이 함수를 사용하여 특정 압축 알고리즘에 대한 정보를 쿼리합니다.
ResetCompressor

새 스트림의 압축을 위해 압축기를 준비합니다.
ResetDecompressor 새 스트림의 압축 해제를 위해 압축 해제기를 준비합니다.
SetCompressorInformation 특정 압축 알고리즘에 대한 압축기 정보를 설정합니다.
SetDecompressorInformation 특정 압축 알고리즘에 대한 압축 해제기에서 정보를 설정합니다.

구조체

COMPRESS_ALLOCATION_ROUTINES 선택적 메모리 할당 및 할당 취소 루틴을 포함하는 구조체입니다.

열거형

COMPRESS_INFORMATION_CLASS 이 열거형의 값은 설정되거나 검색되는 정보 클래스의 유형을 식별합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

comsvcs.h 헤더

아티클2023. 08. 22.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM+(구성 요소 서비스)
- Windows Messenger

comsvcs.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ContextInfo 현재 컨텍스트 개체에 대한 트랜잭션, 작업 및 컨텍스트 정보를 검색합니다. 이 인터페이스의 메서드를 사용하여 개체 컨텍스트 내에 포함된 관련 정보를 검색할 수 있습니다.
ContextInfo2 ContextInfo 인터페이스를 통해 사용할 수 있는 정보를 보완하여 개체의 컨텍스트에 대한 추가 정보를 제공합니다.
IAppDomainHelper 애플리케이션이 실행되는 격리된 환경인 애플리케이션 도메인에 관리되는 개체를 바인딩합니다.
IAssemblyLocator .NET Framework 공용 언어 런타임에서 관리 코드를 사용할 때 어셈블리에 대한 정보를 검색합니다.
IAsyncErrorNotify CoCreateActivity에서 만든 작업을 통해 제출되는 비동기 일괄 처리 작업에서 오류 트래핑을 구현하는 데 사용됩니다.
ICheckSxsConfig 현재 병렬 어셈블리의 구성을 검사 데 사용됩니다.
IComActivityEvents 활동이 생성, 제거 또는 시간 초과된 경우 구독자에게 알립니다.

IComApp2Events COM+ 서버 애플리케이션이 로드, 종료 또는 일시 중지된 경우 구독자에게 알합니다.
IComAppEvents COM+ 서버 애플리케이션이 시작, 종료 또는 강제로 종료될 경우 구독자에게 알합니다.
IComCRMEvents 구성 요소 서비스의 CRM(보상 Resource Manager) 기능의 활동에 대해 구독자에게 알합니다.
IComExceptionEvents 사용자 코드에서 처리되지 않은 예외가 발생할 때 구독자에게 알린 것입니다.
IComIdentityEvents IIS(인터넷 정보 서비스) ASP(활성 서버 페이지) 페이지의 일부인 활동에 대해 구독자에게 알깁니다. 예를 들어 ASP 페이지에서 COM+ 개체가 호출되면 사용자에게 이 작업에 대한 알림이 표시됩니다.
IComInstance2Events 클라이언트에서 개체를 만들거나 해제하는 경우 구독자에게 알 수 있습니다.
IComInstanceEvents 개체의 생성 또는 릴리스를 구독자에게 알릴 수 있습니다.
ICOMLBArguments COM+ 구성 요소 부하 분산 서비스를 활성화하는 데 사용됩니다.
IComLTxEvents COM+ 트랜잭션과 관련된 이벤트를 구독자에게 알 수 있습니다.
IComMethod2Events 개체의 메서드가 호출, 반환 또는 예외를 생성했는지 구독자에게 알립니다. (IComMethod2Events)
IComMethodEvents 개체의 메서드가 호출, 반환 또는 예외를 생성했는지 구독자에게 알립니다. (IComMethodEvents)
IComObjectConstruction2Events 생성된 개체가 만들어지면 구독자에게 알 수 있습니다.

IComObjectConstructionEvents 생성된 개체가 개체 풀에 만들어지는 경우 구독자에게 알 수 있습니다.
IComObjectEvents JIT(Just-In-Time) 활성화 개체의 instance 만들거나 해제했는지 구독자에게 알합니다.
IComObjectPool2Events 개체 풀에서 트랜잭션 또는 비 트랜잭션 개체가 추가되거나 가져올 경우 구독자에게 알 수 있습니다.
IComObjectPoolEvents 새 개체가 풀에 추가되면 구독자에게 알 수 있습니다.
IComObjectPoolEvents2 풀에 대해 새 개체를 만들거나 풀에서 제거할 때 구독자에게 알 수 있습니다.
IComQCEvents 대기 중인 메시지가 만들어지거나, 큐에서 해제되거나, 재시도 또는 배달 못 한 편지 큐로 이동되었는지 구독자에게 알 수 있습니다.
IComResourceEvents 리소스가 생성, 할당, 추적 또는 제거되었는지 구독자에게 알립니다.
IComSecurityEvents 메서드 호출의 인증이 성공했거나 실패한 경우 구독자에게 알 수 있습니다.
IComThreadEvents STA(단일 스레드 아파트)가 만들어지거나 종료되고 아파트 스레드가 할당될 때 구독자에게 알 수 있습니다.
IComTrackingInfoCollection 추적 정보 수집에 대한 정보를 검색합니다.
IComTrackingInfoEvents 컬렉션에 대한 추적 정보가 변경되면 구독자에게 알 수 있습니다.
IComTrackingInfoObject

추적 정보 개체의 속성을 검색합니다.
IComTrackingInfoProperties 추적 정보 개체 및 해당 이름과 연결된 총 속성 수를 검색합니다.
IComTransaction2Events Microsoft DTC(Distributed Transaction Coordinator) 트랜잭션이 시작, 커밋 또는 중단되면 구독자에게 알합니다. 트랜잭션이 2단계 커밋 프로토콜의 준비 단계에 있을 때 구독자에게도 알림이 표시됩니다.
IComTransactionEvents Microsoft DTC(Distributed Transaction Coordinator) 트랜잭션이 시작, 커밋 또는 중단되면 구독자에게 알합니다.
IComUserEvent 지정된 사용자 정의 메트릭의 구독자에게 알 수 있습니다.
IContextProperties 컨텍스트 개체 속성에 대한 액세스를 제공합니다.
IContextState 컨텍스트 상태 플래그를 조작하여 개체 비활성화 및 트랜잭션 투표를 제어합니다.
ICreateWithLocalTransaction 지정된 로컬 트랜잭션의 scope 내에서 실행되는 COM+ 개체를 만듭니다. (ICreateWithLocalTransaction)
ICreateWithTipTransactionEx TIP(Transaction Internet Protocol)를 사용하는 수동 트랜잭션 내에 참여하는 개체를 만듭니다.
ICreateWithTransactionEx 수동 트랜잭션 내에 참여하는 개체를 만듭니다.
ICrmCompensator Microsoft Visual C++ 사용할 때 구조화되지 않은 로그 레코드를 CRM Compensator에 전달합니다.
ICrmCompensatorVariants Microsoft Visual Basic을 사용할 때 CRM Compensator에 구조화된 로그 레코드를 전달합니다.

ICrmFormatLogRecords 로그 레코드를 일반 모니터링 도구를 사용하여 표시할 수 있도록 볼 수 있는 형식으로 변환합니다.
ICrmLogControl CRM 작업자 및 CRM Compensator가 로그에 레코드를 쓰고 지속성으로 만드는 수단입니다.
ICrmMonitor CRM의 현재 상태 스냅샷 캡처하고 특정 CRM 클럭을 보유합니다.
ICrmMonitorClerks 서기의 상태에 대한 정보를 검색합니다.
ICrmMonitorLogRecords 특정 CRM 서기가 지정된 트랜잭션에 대해 유지 관리하는 개별 로그 레코드를 모니터링합니다.
IDispenserDriver COM+ 리소스 디스펜서의 소유자에 의해 호출되어 리소스를 만들고, 참여시키고, 평가하고, 준비하고, 삭제합니다.
IDispenserManager 디스펜서 관리자에 연결합니다.
IEnumNames 이름을 열거합니다.
IGetAppTrackerData 관리 애플리케이션이 COM+ 애플리케이션 실행에 대한 통계 정보를 검색할 수 있도록 합니다.
IGetContextProperties 호출자가 현재 개체의 컨텍스트와 연결된 속성을 가져올 수 있도록 합니다.
IGetSecurityCallContext 현재 호출과 연결된 SecurityCallContext 클래스에서 만든 개체에 대한 참조를 검색합니다. (IGetSecurityCallContext)
IHolder 설치된 리소스 디스펜서에 대한 리소스를 할당하거나 해제합니다.

IManagedActivationEvents 현재 COM+ 컨텍스트 내에서 관리되는 개체에 대한 스텁을 만들고 삭제하는 데 사용됩니다.
IManagedObjectInfo 관리되는 개체의 스텁에 대해 설명합니다.
IManagedPoolAction COM+ 개체 풀에서 개체를 해제하기 전에 알림을 받을 수 있도록 합니다.
IManagedPooledObj COM+ 개체 풀에서 관리되는 개체를 사용하는 방법을 설명합니다.
IMessageMover 메시지를 한 큐에서 다른 큐로 이동합니다.
IMTSActivity MTSCreateActivity 함수에서 만든 작업을 통해 일괄 처리 작업을 제출합니다.
IMTSCall MTSCreateActivity 함수에서 만든 작업을 통해 제출되는 일괄 처리 작업을 구현합니다.
IMtsEventInfo 사용자 정의 이벤트를 설명합니다.
IMtsEvents 실행 중인 패키지에 대한 정보를 가져오고 이벤트 싱크를 설정하는 메서드를 제공합니다.
IMtsGrp 실행 중인 패키지를 열거하는 메서드를 제공합니다.
IMTSLocator 현재 프로세스에 대한 이벤트 디스패처의 IMtsEvents 인터페이스에 대한 액세스를 제공하는 단일 이벤트에 대해 설명합니다.
IObjectConstruct 다른 메서드 또는 개체의 매개 변수를 전달하여 개체 생성 프로세스를 제어합니다.

IObjectConstructString 생성자 문자열에 대한 액세스를 제공합니다. 개체를 생성하는 동안 매개 변수를 지정하려는 경우 사용합니다.
IObjectContext 현재 개체의 컨텍스트에 대한 액세스를 제공합니다. 개체의 컨텍스트는 트랜잭션을 사용하거나 개체의 보안을 처리할 때 주로 사용됩니다. (IObjectContext)
IObjectContextActivity 현재 개체 컨텍스트와 연결된 활동 식별자를 검색합니다.
IObjectContextInfo 현재 컨텍스트 개체에 대한 트랜잭션, 활동 및 컨텍스트 정보를 검색합니다.
IObjectContextInfo2 개체의 컨텍스트에 대한 추가 정보를 제공합니다. 이 인터페이스는 IObjectContextInfo 인터페이스를 확장합니다.
IObjectContextTip TIP(트랜잭션 인터넷 프로토콜) 트랜잭션 컨텍스트를 설명하는 속성을 검색합니다.
IObjectControl COM+ 개체에 대한 컨텍스트별 초기화 및 정리 프로시저를 정의하고 개체를 재활용할 수 있는지 여부를 지정합니다.
IObjPool 트랜잭션 리소스 풀의 각 개체에 대한 키를 나타냅니다.
IPlaybackControl 메시지 큐 배달 메커니즘의 서버 쪽 재생 오류 및 클라이언트 쪽 오류의 비정상적인 처리에 참여할 수 있습니다.
IPoolManager 호출자가 개체 풀을 제어할 수 있도록 합니다.
IProcessInitializer Dllhost.exe 시작하거나 종료할 때마다 호출할 수 있는 메서드를 제공합니다.

ISecurityCallContext 현재 호출의 보안 호출 컨텍스트에 대한 보안 방법 및 정보에 대한 액세스를 제공합니다.
ISecurityCallersColl 호출자 컬렉션의 개별 호출자에 대한 정보에 대한 액세스를 제공합니다.
ISecurityIdentityColl 호출자의 ID를 나타내는 보안 정보 컬렉션에 대한 액세스를 제공합니다. 이 컬렉션에서 사용할 수 있는 항목은 SID, 계정 이름, 인증 서비스, 인증 수준 및 가장 수준입니다.
ISecurityProperty 현재 개체의 원래 호출자 또는 직접 호출자의 보안 식별자를 결정합니다. 그러나 개체의 호출자에 대한 정보를 가져오는 기본 방법은 ISecurityCallContext 인터페이스를 사용하는 것입니다.
ISelectCOMLBServer COM+ 구성 요소 부하 분산 서비스를 활성화합니다.
ISendMethodEvents 구현하는 개체의 메서드가 호출되거나 호출에서 반환할 때마다 구독자에게 알립니다.
IServiceActivity CoCreateActivity에서 만든 작업을 통해 제출된 일괄 처리 작업을 호출하는 데 사용됩니다.
IServiceCall CoCreateActivity에서 만든 작업을 통해 제출되는 일괄 처리 작업을 구현하는 데 사용됩니다.
IServiceComTIIntrinsicsConfig CoCreateActivity 또는 CoEnterServiceDomain 함수를 호출할 때 수행되는 작업에 대한 COMTI(COM Transaction Integrator) 내장 함수를 구성합니다.
IServiceIISIntrinsicsConfig CoCreateActivity 또는 CoEnterServiceDomain 함수를 호출할 때 수행되는 작업에 대한 IIS 내장 함수를 구성합니다.
IServiceInheritanceConfig 현재 컨텍스트를 기반으로 새 컨텍스트를 생성할지 또는 CServiceConfig의 정보만을 기반으로 새 컨텍스트를 만들 것인지 결정합니다.

IServicePartitionConfig CoCreateActivity 또는 CoEnterServiceDomain을 호출할 때 수행되는 작업에 파티션을 사용하는 방법을 구성합니다.
IServicePool COM+ 개체 풀을 관리하는 데 사용됩니다.
IServicePoolConfig 개체 풀을 구성하는 데 사용됩니다.
IServiceSxsConfig CoCreateActivity 또는 CoEnterServiceDomain을 호출할 때 수행되는 작업에 대해 병렬 어셈블리를 구성합니다.
IServiceSynchronizationConfig CoCreateActivity 또는 CoEnterServiceDomain을 호출할 때 수행되는 작업에 대한 동기화를 구성합니다.
IServiceSysTxnConfig 트랜잭션 프록시를 사용하여 지정한 기존 트랜잭션의 scope 코드 집합을 실행할 수 있습니다.
IServiceThreadPoolConfig CoCreateActivity를 호출하여 반환되는 활동 개체의 스레드 풀을 구성합니다.
IServiceTrackerConfig CoCreateActivity 또는 CoEnterServiceDomain을 호출할 때 수행되는 작업에 대한 추적기 속성을 구성합니다.
IServiceTransactionConfig CoCreateActivity 또는 CoEnterServiceDomain을 호출할 때 수행되는 작업에 대한 트랜잭션 서비스를 구성합니다. (IServiceTransactionConfig)
IServiceTransactionConfigBase CoCreateActivity 또는 CoEnterServiceDomain을 호출할 때 수행되는 작업에 대한 트랜잭션 서비스를 구성합니다. (IServiceTransactionConfigBase)
ISharedProperty 공유 속성의 값을 설정하거나 검색하는 데 사용할 수 있는 속성 메서드를 노출합니다.

ISharedPropertyGroup 공유 속성 그룹의 공유 속성을 만들고 액세스하는 데 사용됩니다.
ISharedPropertyGroupManager 공유 속성 그룹을 만들고 기존 공유 속성 그룹에 대한 액세스를 가져오는 데 사용됩니다.
ISystemAppEventData COM+ 애플리케이션 instance 만들거나 다시 구성할 때 구독자에게 알 수 있습니다.
IThreadPoolKnobs 스레드 풀의 동작을 제어하는 데 사용됩니다.
ITransactionContext 단일 트랜잭션에서 여러 COM+ 개체의 작업을 작성하고 명시적으로 트랜잭션을 커밋하거나 중단할 수 있습니다.
ITransactionContextEx 트랜잭션을 시작하는 제네릭 트랜잭션 개체에 대한 기본 메서드를 제공합니다. 이 인터페이스의 메서드를 호출하면 단일 트랜잭션에서 여러 COM+ 개체의 작업을 작성하고 트랜잭션을 명시적으로 커밋하거나 중단할 수 있습니다.
ITransactionProperty 트랜잭션 리소스 풀을 가져오는 데 사용됩니다.
ITransactionProxy COM+ 트랜잭션 컨텍스트가 비 DTC 트랜잭션에서 작동하는 방법을 제공합니다.
ITransactionResourcePool 트랜잭션이 완료될 때까지 사용되는 IObjPool에서 키로 지정한 풀된 개체 목록을 유지 관리합니다.
ITransactionStatus CServiceConfig가 CoEnterServiceDomain 호출에서 트랜잭션을 사용하도록 구성된 경우 CoLeaveServiceDomain 호출로 완료된 트랜잭션의 상태 검색하는 데 사용됩니다.
Objectcontext 현재 개체의 컨텍스트에 대한 액세스를 제공합니다. 개체의 컨텍스트는 트랜잭션을 사용하거나 개체의 보안을 처리할 때 주로 사용됩니다. (ObjectContext)

ObjectControl 구성 요소에서 이 인터페이스를 구현하는 경우 COM+ 런타임 환경은 적절한 시간에 개체에서 해당 메서드를 자동으로 호출합니다.
SecurityProperty 현재 개체의 원래 호출자 및 직접 호출자에 대한 정보를 검색합니다.

Functions

CoCreateActivity COM+ 구성 요소를 만들지 않고 COM+ 서비스를 사용할 수 있는 동기 또는 비동기 일괄 작업을 수행하기 위한 작업을 만듭니다.
CoEnterServiceDomain COM+ 서비스를 사용할 수 있는 코드를 입력하는 데 사용됩니다.
CoLeaveServiceDomain COM+ 서비스를 사용하는 코드를 남기는 데 사용됩니다.
GetManagedExtensions 설치된 COM+버전이 서비스된 구성 요소(관리되는 개체)를 관리하기 위해 제공되는 특수 기능을 지원하는지 여부를 결정합니다.
GetObjectContext 현재 COM+ 개체와 연결된 컨텍스트에 대한 참조를 검색합니다.
MTSCreateActivity 동기 또는 비동기 일괄 처리를 수행하는 단일 스레드 아파트에 활동을 만듭니다.
RecycleSurrogate 호출 프로세스를 재활용합니다.
SafeRef SafeRef 함수

구조체

ApplicationProcessRecycleInfo COM+ 애플리케이션을 호스팅하는 프로세스의 재활용에 대한 세부 정보를 나타냅니다.
ApplicationProcessStatistics COM+ 애플리케이션을 호스팅하는 프로세스에 대한 통계 정보를 나타냅니다.
ApplicationProcessSummary COM+ 애플리케이션을 호스팅하는 프로세스에 대한 요약 정보를 나타냅니다.
ApplicationSummary 특정 프로세스에서 호스트되는 COM+ 애플리케이션을 나타냅니다. 프로세스의 모든 SWC(구성 요소 없는 서비스) 컨텍스트에 대한 의사 애플리케이션 항목을 나타낼 수도 있습니다.
ComponentHangMonitorInfo COM+ 구성 요소에 대한 중단 모니터링 구성을 나타냅니다.
ComponentStatistics 특정 프로세스에서 호스트되는 COM+ 구성 요소에 대한 통계 정보를 나타냅니다.
ComponentSummary 특정 프로세스에서 호스트되는 COM+ 구성 요소에 대한 요약 정보를 나타냅니다. SWC(구성 요소 없는 서비스) 컨텍스트를 나타낼 수도 있습니다.
COMSVCSEVENTINFO 생성된 시간 및 생성한 프로세스 서버 및 COM+ 애플리케이션과 같은 이벤트에 대한 컨텍스트 정보를 나타냅니다.
CrmLogRecordRead CRM(보상 Resource Manager)에 대한 구조화되지 않은 로그 레코드를 포함합니다.

열거형

COMPLUS_APPTYPE 추적기 서버에서 추적하는 애플리케이션 유형을 나타냅니다.
CRMFLAGS CRM 보상자에 대한 특정 로그 레코드가 작성된 시기에 대한 정보를 제공합니다.
CRMREGFLAGS CRM 보상자가 수신해야 하는 트랜잭션 완료 단계와 복구를 시도한 후에도 의심스러운 트랜잭션이 남아 있는 경우 복구가 실패해야 하는지 여부를 제어합니다.
CrmTransactionState 트랜잭션의 현재 트랜잭션 상태를 나타냅니다.
CSC_Binding CoCreateActivity에서 반환된 작업을 통해 제출된 모든 작업을 STA(단일 스레드 아파트)에만 바인딩해야 하는지 여부를 나타냅니다. 이 열거형은 MTA(다중 스레드 아파트)에는 영향을 주지 않습니다.
CSC_COMTIIntrinsicsConfig 현재 COMTI(COM Transaction Integrator) 내장 함수가 새 컨텍스트로 전파되는지 여부를 나타냅니다.
CSC_IISIntrinsicsConfig 현재 IIS 내장 함수가 새 컨텍스트로 전파되는지 여부를 나타냅니다.
CSC_InheritanceConfig 현재 컨텍스트를 기반으로 새 컨텍스트를 만들거나 CServiceConfig의 정보를 기반으로 새 컨텍스트를 만들 것인지를 나타냅니다.
CSC_PartitionConfig 묶은 컨텍스트가 실행되는 COM+ 파티션을 나타냅니다.
CSC_SxsConfig CServiceConfig에 대해 병렬 어셈블리를 구성하는 방법을 나타냅니다.
CSC_SynchronizationConfig CServiceConfig에 대해 동기화를 구성하는 방법을 나타냅니다.

CSC_ThreadPool CoCreateActivity에서 반환된 작업을 통해 제출되는 작업이 실행되는 스레드 풀을 나타냅니다.
CSC_TrackerConfig 묶은 코드가 실행되는 컨텍스트에 추적기 속성이 추가되는지 여부를 나타냅니다.
CSC_TransactionConfig 트랜잭션이 CServiceConfig에 대해 구성된 방식을 나타냅니다.
GetAppTrackerDataFlags IGetAppTrackerData 인터페이스에 대한 호출에서 반환되는 데이터를 제어합니다.
TRACKING_COLL_TYPE 추적 정보 컬렉션에 있는 개체의 형식을 나타냅니다.
TransactionVote 현재 트랜잭션을 커밋하거나 중단할 개체의 준비 상태를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

control.h 헤더

아티클2023. 10. 05.

이 헤더는 DirectShow에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Directshow

control.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAMStats IAMStats 인터페이스는 필터 그래프 관리자에서 성능 데이터를 검색합니다.
IBasicAudio IBasicAudio 인터페이스는 오디오 스트림의 볼륨과 균형을 제어합니다. 이 인터페이스는 오디오 렌더러(WaveOut) 필터 및 DirectSound 렌더러 필터에서 구현되지만 필터 그래프 관리자를 통해 애플리케이션에 노출됩니다.
IBasicVideo IBasicVideo 인터페이스는 대상 및 원본 사각형과 같은 비디오 속성을 설정합니다.
IBasicVideo2 IBasicVideo2 인터페이스는 IBasicVideo 인터페이스를 확장합니다.
IDeferredCommand IDeferredCommand 인터페이스는 IQueueCommand 인터페이스를 사용하여 큐에 대기된 그래프 제어 명령을 취소하거나 수정합니다. 애플리케이션이 Filter Graph Manager에서 IQueueCommand 메서드를 호출하면 IDeferredCommand 인터페이스에 대한 포인터를 받습니다. 애플리케이션은 인터페이스를 사용하여 명령을 취소하거나 연기하거나 명령에서 반환 값을 검색할 수 있습니다.
IMediaControl IMediaControl 인터페이스는 필터 그래프를 통해 데이터 흐름을 제어하는 메서드를 제공합니다.
IMediaEvent IMediaEvent 인터페이스에는 이벤트 알림을 검색하고 Filter Graph Manager의 이벤트 기본 처리를 재정의하는 메서드가 포함되어 있습니다.

IMediaEventEx IMediaEventEx 인터페이스는 이벤트 알림을 검색하고 필터 그래프의 기본 이벤트 처리를 재정의하는 메서드를 포함하는 IMediaEvent 인터페이스를 상속합니다.
IMediaPosition IMediaPosition 인터페이스에는 스트림 내의 위치를 찾는 메서드가 포함되어 있습니다.
IQueueCommand IQueueCommand 인터페이스는 지정된 시간에 처리하기 위한 명령을 큐에 대기합니다.
IVideoWindow IVideoWindow 인터페이스는 비디오 창의 속성을 설정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

corewindow.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

corewindow.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICoreInputInterop Windows 애플리케이션의 핵심 입력 개체에서 입력 원본을 사용하도록 설정합니다.
ICoreInputInterop2
ICoreWindowAdapterInterop
ICoreWindowComponentInterop
ICoreWindowInterop 앱이 이 인터페이스와 연결된 창(CoreWindow)의 창 핸들을 가져올 수 있도록 합니다.

Functions

CreateControlInput 호출자의 UI 스레드에 ICoreInputSourceBase 개체를 만듭니다.
CreateControlInputEx 작업자 스레드 또는 UI 스레드에 ICoreInputSourceBase 개체를 만듭니다.
HWND_UserFree

HWND_UserFree64
HWND_UserMarshal
HWND_UserMarshal64
HWND_UserSize
HWND_UserSize64
HWND_UserUnmarshal
HWND_UserUnmarshal64

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

cpl.h 헤더

아티클2023. 08. 22.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

cpl.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

콜백 함수

APPLET_PROC 제어판 애플리케이션의 진입점 역할을 합니다. 라이브러리 정의 콜백 함수입니다.

구조체

CPLINFO 제어판 애플리케이션에서 지원하는 대화 상자에 대한 리소스 정보 및 애플리케이션 정의 값을 포함합니다. (CPLINFO)
NEWCPLINFOA 제어판 애플리케이션에서 지원하는 대화 상자에 대한 리소스 정보 및 애플리케이션 정의 값을 포함합니다. (NEWCPLINFOA)
NEWCPLINFOW 제어판 애플리케이션에서 지원하는 대화 상자에 대한 리소스 정보 및 애플리케이션 정의 값을 포함합니다. (NEWCPLINFOW)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

credentialprovider.h 헤더

아티클2023. 08. 22.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

credentialprovider.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IConnectableCredentialProviderCredential IConnectableCredentialProviderCredential 개체를 연결하고 연결을 끊는 메서드를 노출합니다.
ICredentialProvider 자격 증명 공급자의 설정 및 조작에 사용되는 메서드를 노출합니다. 모든 자격 증명 공급자는 이 인터페이스를 구현해야 합니다.
ICredentialProviderCredential 자격 증명을 처리할 수 있는 메서드를 노출합니다.
ICredentialProviderCredential2 사용자의 SID(보안 식별자)를 검색하는 메서드를 추가하여 ICredentialProviderCredential 인터페이스를 확장합니다. 자격 증명은 해당 사용자와 연결되며 사용자의 타일 아래에 그룹화할 수 있습니다.
ICredentialProviderCredentialEvents 로그온 UI 또는 자격 증명 UI에서 상태 또는 텍스트 변경 이벤트를 알리기 위해 자격 증명에서 사용하는 비동기 콜백 메커니즘을 제공합니다.
ICredentialProviderCredentialEvents2 Logon UI 또는 자격 증명 UI에서 필드를 일괄 업데이트할 수 있는 메서드를 추가하여 ICredentialProviderCredentialEvents 인터페이스를 확장합니다.
ICredentialProviderCredentialWithFieldOptions 자격 증명 공급자 프레임워크가 로그온 또는 자격 증명 UI에서 필드의 옵션을 사용자 지정했는지 여부를 확인할 수 있도록 하는 메서드를 제공합니다.

ICredentialProviderEvents 자격 증명 공급자가 자격 증명 목록 또는 해당 필드의 변경 내용을 알리기 위해 사용하는 비동기 콜백 메커니즘을 제공합니다.
ICredentialProviderFilter 런타임에 사용할 수 있는 정보를 기반으로 자격 증명 공급자를 동적으로 필터링하는 데 사용됩니다.
ICredentialProviderSetUserArray 자격 증명 공급자가 로그온 또는 자격 증명 UI에 표시될 사용자 집합을 수신할 수 있도록 하는 메서드를 제공합니다.
ICredentialProviderUser 로그온 또는 자격 증명 UI에 포함된 개별 사용자의 특정 속성을 검색하는 데 사용되는 메서드를 제공합니다.
ICredentialProviderUserArray 로그온 또는 자격 증명 UI에 표시되는 사용자 집합을 나타냅니다. 이 정보를 사용하면 자격 증명 공급자가 집합을 열거하여 각 사용자에 대한 속성 정보를 검색하여 필드를 채움하거나 집합을 필터링할 수 있습니다.
IQueryContinueWithStatus 네트워크에 연결하여 이러한 시도를 계속해야 하는지 확인하는 동안 자격 증명 공급자가 QueryContinue를 호출할 수 있는 표준 메커니즘을 제공하는 메서드를 노출합니다.

구조체

CREDENTIAL_PROVIDER_CREDENTIAL_SERIALIZATION 자격 증명에 대한 세부 정보를 포함합니다.
CREDENTIAL_PROVIDER_FIELD_DESCRIPTOR 자격 증명의 단일 필드를 설명합니다. 예를 들어 문자열 또는 사용자 이미지입니다.

열거형

CREDENTIAL_PROVIDER_ACCOUNT_OPTIONS 자격 증명 공급자가 "다른 사용자" 타일과 연결하기 위해 반환해야 하는 자격 증명 유형을 나타냅니다. ICredentialProviderUserArray_GetAccountOptions 사용합니다.
CREDENTIAL_PROVIDER_CREDENTIAL_FIELD_OPTIONS 로그온 또는 자격 증명 UI의 단일 필드에 대한 사용자 지정 옵션을 제공합니다.
CREDENTIAL_PROVIDER_FIELD_INTERACTIVE_STATE 필드의 상태 및 사용자가 필드와 상호 작용할 수 있는 방법을 설명합니다. 자격 증명 공급자는 다양한 대화형 상태로 필드를 표시할 수 있습니다.
CREDENTIAL_PROVIDER_FIELD_STATE 자격 증명 UI에서 단일 필드의 상태를 지정합니다.
CREDENTIAL_PROVIDER_FIELD_TYPE 자격 증명 필드의 형식을 지정합니다. CREDENTIAL_PROVIDER_FIELD_DESCRIPTOR 사용합니다.
CREDENTIAL_PROVIDER_GET_SERIALIZATION_RESPONSE 자격 증명 공급자가 자격 증명을 직렬화하려고 할 때의 응답을 설명합니다.
CREDENTIAL_PROVIDER_STATUS_ICON 표시할 상태 아이콘을 나타냅니다.
CREDENTIAL_PROVIDER_USAGE_SCENARIO 자격 증명 공급자가 지원되는 시나리오를 선언합니다. CPUS(자격 증명 공급자 사용 시나리오)를 사용하면 자격 증명 공급자가 시나리오에서 고유한 열거 동작 및 UI 필드 설정을 제공할 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

credssp.h 헤더

아티클2023. 08. 22.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

credssp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

CREDSSP_CRED Schannel 및 Negotiate 보안 패키지에 대한 인증 데이터를 지정합니다.
SecPkgContext_ClientCreds QueryContextAttributes(CredSSP) 함수를 호출할 때 클라이언트 자격 증명을 지정합니다.

열거형

CREDSPP_SUBMIT_TYPE CREDSSP_CRED 구조에서 지정한 자격 증명의 유형을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

cryptdlg.h 헤더

아티클2023. 08. 22.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

cryptdlg.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CertModifyCertificatesToTrust 지정된 목적을 위해 CTL(인증서 신뢰 목록)의 인증서 집합을 수정합니다.
CertSelectCertificateA 사용자가 지정된 조건과 일치하는 인증서 집합에서 인증서를 선택할 수 있는 대화 상자를 제공합니다. (ANSI)
CertSelectCertificateW 사용자가 지정된 조건과 일치하는 인증서 집합에서 인증서를 선택할 수 있는 대화 상자를 제공합니다. (유니코드)
CertViewPropertiesA CertViewProperties 함수는 UI(사용자 인터페이스) 대화 상자에서 인증서의 속성을 표시합니다. 이 함수에는 연결된 가져오기 라이브러리가 없습니다. CryptDlg.dll 동적으로 연결하려면 LoadLibrary 및 GetProcAddress 함수를 사용해야 합니다. (ANSI)
CertViewPropertiesW CertViewProperties 함수는 UI(사용자 인터페이스) 대화 상자에서 인증서의 속성을 표시합니다. 이 함수에는 연결된 가져오기 라이브러리가 없습니다. CryptDlg.dll 동적으로 연결하려면 LoadLibrary 및 GetProcAddress 함수를 사용해야 합니다. (유니코드)
GetFriendlyNameOfCertA 인증서의 표시 이름을 검색합니다. (ANSI)
GetFriendlyNameOfCertW 인증서의 표시 이름을 검색합니다. (유니코드)

콜백 함수

PFNCMFILTERPROC 각 인증서를 필터링하여 CertSelectCertificate 함수에 의해 표시되는 인증서 선택 대화 상자에 나타날지 여부를 확인합니다.
PFNCMHOOKPROC CertSelectCertificate 함수에서 생성된 인증서 선택 대화 상자에서 메시지를 처리하기 전에 호출됩니다.

구조체

CERT_SELECT_STRUCT_A 인증서 선택 대화 상자에 표시되는 인증서를 선택할 조건을 포함합니다. 이 구조체는 CertSelectCertificate 함수에 사용됩니다. (ANSI)
CERT_SELECT_STRUCT_W 인증서 선택 대화 상자에 표시되는 인증서를 선택할 조건을 포함합니다. 이 구조체는 CertSelectCertificate 함수에 사용됩니다. (유니코드)
CERT_VIEWPROPERTIES_STRUCT_A CERT_VIEWPROPERTIES_STRUCT 구조는 인증서의 속성을 표시하기 위해 CertViewProperties 함수를 호출할 때 사용되는 정보를 정의합니다. (ANSI)
CERT_VIEWPROPERTIES_STRUCT_W CERT_VIEWPROPERTIES_STRUCT 구조는 인증서의 속성을 표시하기 위해 CertViewProperties 함수를 호출할 때 사용되는 정보를 정의합니다. (유니코드)
CTL_MODIFY_REQUEST CTL(인증서 신뢰 목록)을 수정하는 요청을 포함합니다. 이 구조체는 CertModifyCertificatesToTrust 함수에 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

cryptuiapi.h 헤더

아티클2023. 08. 22.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

cryptuiapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CertSelectionGetSerializedBlob CERT_SELECTUI_INPUT 구조에서 직렬화된 인증서 BLOB을 검색하는 데 사용되는 도우미 함수입니다.
CryptUIDlgCertMgr 사용자가 인증서를 관리할 수 있는 대화 상자를 표시합니다.
CryptUIDlgSelectCertificateFromStore 지정된 저장소에서 인증서를 선택할 수 있는 대화 상자를 표시합니다.
CryptUIDlgViewCertificateA 지정된 인증서를 표시하는 대화 상자를 표시합니다. (ANSI)
CryptUIDlgViewCertificateW 지정된 인증서를 표시하는 대화 상자를 표시합니다. (유니코드)
CryptUIDlgViewContext 인증서, CTL 또는 CRL 컨텍스트를 표시합니다.
CryptUIWizDigitalSign 문서 또는 BLOB에 디지털 서명합니다.
CryptUIWizExport 인증서, CTL(인증서 신뢰 목록), CRL(인증서 해지 목록) 또는 인증서 저장소를 파일로 내보냅니다.
CryptUIWizFreeDigitalSignContext

CryptUIWizDigitalSign 함수에 의해 할당된 CRYPTUI_WIZ_DIGITAL_SIGN_CONTEXT 구조를 해제합니다.
CryptUIWizImport 인증서, CTL(인증서 신뢰 목록), CRL(인증서 해지 목록) 또는 인증서 저장소를 인증서 저장소로 가져옵니다.

콜백 함수

PFNCFILTERPROC CryptUIWizDigitalSign 함수에 의해 표시되는 디지털 서명 마법사에 표시되는 인증서를 필터링하는 애플리케이션 정의 콜백 함수입니다.

구조체

CERT_SELECTUI_INPUT CertSelectionGetSerializedBlob 함수에서 저장소 또는 인증서 체인 배열에 포함된 인증서를 직렬화하는 데 사용됩니다. 반환된 직렬화된 BLOB을 CredUIPromptForWindowsCredentials 함수에 전달할 수 있습니다.
CRYPTUI_CERT_MGR_STRUCT 인증서 관리자 대화 상자에 대한 정보를 포함합니다.
CRYPTUI_INITDIALOG_STRUCT CRYPTUI_VIEWCERTIFICATE_STRUCT 구조를 지원합니다.
CRYPTUI_VIEWCERTIFICATE_STRUCTA 볼 인증서에 대한 정보를 포함합니다. 이 구조체는 CryptUIDlgViewCertificate 함수에 사용됩니다. (ANSI)
CRYPTUI_VIEWCERTIFICATE_STRUCTW 볼 인증서에 대한 정보를 포함합니다. 이 구조체는 CryptUIDlgViewCertificate 함수에 사용됩니다. (유니코드)

CRYPTUI_WIZ_DIGITAL_SIGN_BLOB_INFO CryptUIWizDigitalSign 함수에서 사용하는 공개 키 BLOB에 대한 정보를 포함합니다.
CRYPTUI_WIZ_DIGITAL_SIGN_CERT_PVK_INFO CryptUIWizDigitalSign 함수에서 사용하는 인증서를 포함하는 PVK 파일에 대한 정보를 포함합니다.
CRYPTUI_WIZ_DIGITAL_SIGN_CONTEXT BLOB에 대한 정보를 포함하기 위해 CryptUIWizDigitalSign 함수와 함께 사용됩니다.
CRYPTUI_WIZ_DIGITAL_SIGN_EXTENDED_INFO 서명에 대한 확장 정보를 포함하기 위해 CRYPTUI_WIZ_DIGITAL_SIGN_INFO 구조체와 함께 사용됩니다.
CRYPTUI_WIZ_DIGITAL_SIGN_INFO 디지털 서명에 대한 정보를 포함합니다.
CRYPTUI_WIZ_DIGITAL_SIGN_PVK_FILE_INFO 디지털 서명 마법사에서 사용하는 PVK 파일에 대한 정보를 포함하기 위해 CRYPTUI_WIZ_DIGITAL_SIGN_INFO 구조와 함께 사용됩니다.
CRYPTUI_WIZ_DIGITAL_SIGN_STORE_INFO 디지털 서명 마법사에서 사용하는 인증서 저장소에 대한 정보를 포함합니다.
CRYPTUI_WIZ_EXPORT_CERTCONTEXT_INFO 인증서가 내보내는 개체일 때 CryptUIWizExport 함수의 작업을 제어하는 정보를 포함합니다.
CRYPTUI_WIZ_EXPORT_INFO CryptUIWizExport 함수의 작업을 제어하는 정보를 포함합니다.
CRYPTUI_WIZ_IMPORT_SRC_INFO CryptUIWizImport 함수로 가져올 주체를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

cryptxml.h 헤더

아티클2023. 08. 22.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

cryptxml.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CryptXmlAddObject 인코딩을 위해 열린 문서 컨텍스트의 Signature에 Object 요소를 추가합니다.
CryptXmlClose 암호화 XML 개체 핸들을 닫습니다.
CryptXmlCreateReference XML 서명에 대한 참조를 만듭니다.
CryptXmlDigestReference 애플리케이션에서 확인된 참조를 다이제스트하는 데 사용됩니다. 이 함수는 다이제스트를 업데이트하기 전에 변환을 적용합니다.
CryptXmlEncode 제공된 XML 기록기 콜백 함수를 사용하여 서명 데이터를 인코딩합니다.
CryptXmlGetAlgorithmInfo CRYPT_XML_ALGORITHM 구조를 디코딩하고 알고리즘에 대한 정보를 반환합니다.
CryptXmlGetDocContext 제공된 핸들에서 지정한 문서 컨텍스트를 반환합니다.
CryptXmlGetReference 제공된 핸들에서 지정한 Reference 요소를 반환합니다.
CryptXmlGetSignature

XML Signature 요소를 반환합니다.
CryptXmlGetStatus 제공된 핸들에서 지정한 개체에 대한 상태 정보를 포함하는 CRYPT_XML_STATUS 구조를 반환합니다.
CryptXmlGetTransforms 기본 변환 체인 엔진에 대한 정보를 반환합니다.
CryptXmlImportPublicKey 제공된 핸들에서 지정한 공개 키를 가져옵니다.
CryptXmlOpenToDecode XML 디지털 서명을 열어 디코딩하고 CRYPT_XML_SIGNATURE 구조를 캡슐화하는 문서 컨텍스트의 핸들을 반환합니다. 문서 컨텍스트에는 하나 이상의 서명 요소가 포함될 수 있습니다.
CryptXmlOpenToEncode XML 디지털 서명을 열어 인코딩하고 열린 Signature 요소의 핸들을 반환합니다. 핸들은 단일 CRYPT_XML_SIGNATURE 구조로 문서 컨텍스트를 캡슐화하고 CryptXmlClose 함수가 호출될 때까지 열린 상태로 유지됩니다.
CryptXmlSetHMACSecret CryptXmlSign 또는 CryptXmlVerify 함수를 호출하기 전에 핸들에서 HMAC 비밀을 설정합니다.
CryptXmlSign SignedInfo 요소의 암호화 서명을 만듭니다.
CryptXmlVerifySignature SignedInfo 요소의 암호화 서명 유효성 검사를 수행합니다.

콜백 함수

CryptXmlDllCloseDigest CryptXmlDllCreateDigest 함수에서 할당한 CRYPT_XML_DIGEST 해제합니다.
CryptXmlDllCreateDigest

지정된 메서드에 대한 다이제스트 개체를 만듭니다.
CryptXmlDllCreateKey KeyValue 요소를 구문 분석하고 CNG(Cryptography API:_Next Generation) BCrypt 키 핸들을 만들어 서명을 확인합니다.
CryptXmlDllDigestData 다이제스트에 데이터를 넣습니다.
CryptXmlDllEncodeAlgorithm Agile 알고리즘에 대한 SignatureMethod 또는 DigestMethod 요소를 기본 매개 변수로 인코딩합니다.
CryptXmlDllEncodeKeyValue KeyValue 요소를 인코딩합니다.
CryptXmlDllFinalizeDigest 다이제스트 값을 검색합니다.
CryptXmlDllGetAlgorithmInfo XML 알고리즘을 디코딩하고 알고리즘에 대한 정보를 반환합니다.
CryptXmlDllGetInterface 지정된 알고리즘에 대한 암호화 확장 함수에 대한 포인터를 검색합니다.
CryptXmlDllSignData 데이터에 서명합니다. (CryptXmlDllSignData)
CryptXmlDllVerifySignature 서명을 확인합니다.
PFN_CRYPT_XML_CREATE_TRANSFORM 지정된 데이터 공급자에 대한 변환을 만듭니다.
PFN_CRYPT_XML_DATA_PROVIDER_CLOSE 데이터 공급자를 해제합니다.
PFN_CRYPT_XML_DATA_PROVIDER_READ

XML 데이터를 읽습니다.
PFN_CRYPT_XML_ENUM_ALG_INFO 미리 정의되고 등록된 CRYPT_XML_ALGORITHM_INFO 항목을 열거합니다.
PFN_CRYPT_XML_WRITE_CALLBACK XML 데이터를 씁니다.

구조체

CRYPT_XML_ALGORITHM 메시지에 서명하거나 변환하는 데 사용되는 알고리즘을 지정합니다.
CRYPT_XML_ALGORITHM_INFO 알고리즘 정보를 포함합니다.
CRYPT_XML_BLOB 임의의 바이트 배열을 포함합니다.
CRYPT_XML_CRYPTOGRAPHIC_INTERFACE 구현된 CryptXML 함수를 노출합니다.
CRYPT_XML_DATA_BLOB XML로 인코딩된 데이터를 포함합니다.
CRYPT_XML_DATA_PROVIDER XML 데이터 공급자에 대한 인터페이스를 지정합니다.
CRYPT_XML_DOC_CTXT 문서 컨텍스트 정보를 정의합니다.
CRYPT_XML_ISSUER_SERIAL X.509에서 발급된 고유 이름 일련 번호 쌍을 포함합니다.
CRYPT_XML_KEY_DSA_KEY_VALUE

DSA(디지털 서명 알고리즘) 키 값을 정의합니다. CRYPT_XML_KEY_DSA_KEY_VALUE 구조체는 CRYPT_XML_KEY_VALUE 구조체에서 키 값 공용 구조체의 요소로 사용됩니다.
CRYPT_XML_KEY_ECDSA_KEY_VALUE ECDSA(타원 곡선 디지털 서명 알고리즘) 키 값을 정의합니다. CRYPT_XML_KEY_ECDSA_KEY_VALUE 구조체는 CRYPT_XML_KEY_VALUE 구조체에서 키 값 공용 구조체의 요소로 사용됩니다.
CRYPT_XML_KEY_INFO 주요 정보 데이터를 캡슐화합니다.
CRYPT_XML_KEY_INFO_ITEM KeyInfo 요소에 해당하는 키 정보 데이터를 캡슐화합니다. KeyInfo 요소를 사용하면 받는 사람이 서명의 유효성을 검사하는 데 필요한 키를 가져올 수 있습니다.
CRYPT_XML_KEY_RSA_KEY_VALUE RSA 키 값을 정의합니다. CRYPT_XML_KEY_RSA_KEY_VALUE 구조체는 CRYPT_XML_KEY_VALUE 구조체에서 키 값 공용 구조체의 요소로 사용됩니다.
CRYPT_XML_KEY_VALUE 서명의 유효성을 검사하는 데 유용할 수 있는 단일 공개 키를 포함합니다.
CRYPT_XML_KEYINFO_PARAM CryptXmlSign 함수에서 인코딩할 KeyInfo 요소의 멤버를 지정하는 데 사용됩니다.
CRYPT_XML_OBJECT 서명의 Object 요소에 대해 설명합니다.
CRYPT_XML_PROPERTY CryptXML 속성에 대한 정보를 포함합니다.
CRYPT_XML_REFERENCE Reference 요소를 채우는 데 사용되는 정보를 포함합니다.
CRYPT_XML_REFERENCES CRYPT_XML_REFERENCE 구조체의 배열을 정의합니다.
CRYPT_XML_SIGNATURE Signature 요소를 채우는 데 사용되는 정보를 포함합니다.

CRYPT_XML_SIGNED_INFO XML로 인코딩된 SignedInfo 요소에 대해 설명합니다.
CRYPT_XML_STATUS 서명 유효성 검사 상태 대한 정보, SignedInfo 요소에 대한 요약 상태 정보 또는 Reference 요소 배열에 대한 요약 상태 정보를 반환합니다.
CRYPT_XML_TRANSFORM_CHAIN_CONFIG XML 디지털 서명에 사용할 수 있는 애플리케이션 정의 변환을 포함합니다.
CRYPT_XML_TRANSFORM_INFO 데이터 변환을 적용할 때 사용되는 정보를 포함합니다.
CRYPT_XML_X509DATA X509Data 요소의 선택 시퀀스를 나타냅니다.
CRYPT_XML_X509DATA_ITEM X509Data 명명된 요소에서 인코딩할 X.509 데이터를 나타냅니다.

열거형

CRYPT_XML_CHARSET XML에 사용되는 문자 집합을 지정하는 데 사용됩니다.
CRYPT_XML_KEYINFO_SPEC CryptXmlSign 함수에서 dwKeyInfoSpec 매개 변수의 값을 지정합니다.
CRYPT_XML_PROPERTY_ID XML 속성의 형식과 사용을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

cscapi.h 헤더

아티클2023. 08. 22.

이 헤더는 오프라인 파일에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 오프라인 파일

cscapi.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

OfflineFilesEnable 오프라인 파일 기능을 사용하거나 사용하지 않도록 설정합니다.
OfflineFilesQueryStatus 오프라인 파일 기능이 활성화되어 있는지 여부와 활성 상태인지 여부를 확인합니다.
OfflineFilesQueryStatusEx 오프라인 파일 기능을 사용할 수 있는지 여부와 활성화되어 있고 사용 가능한지 여부를 결정합니다. 이 함수는 추가 출력 매개 변수가 있다는 점을 제외하고 OfflineFilesQueryStatus 함수와 동일합니다.
OfflineFilesStart 오프라인 파일 서비스를 시작합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

cscobj.h 헤더

아티클2023. 08. 22.

이 헤더는 오프라인 파일에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 오프라인 파일

cscobj.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumOfflineFilesItems IOfflineFilesItem 인터페이스 포인터의 컬렉션을 나타냅니다.
IEnumOfflineFilesSettings 오프라인 파일 서비스와 연결된 설정 개체를 열거합니다.
IOfflineFilesCache 오프라인 파일 캐시를 관리하는 데 사용됩니다.
IOfflineFilesCache2 RenameItemEx 메서드를 구현합니다.
IOfflineFilesChangeInfo 오프라인으로 작업하는 동안 항목에 대한 로컬 변경 내용과 관련된 정보를 나타냅니다.
IOfflineFilesConnectionInfo 오프라인 파일의 온라인 오프라인 전환 동작과 관련된 쿼리 및 작업 기능을 제공합니다.
IOfflineFilesDirectoryItem 오프라인 파일 캐시의 디렉터리 항목을 나타냅니다.
IOfflineFilesDirtyInfo 오프라인 파일 캐시의 비동기("더티") 파일에 대한 정보를 나타냅니다.
IOfflineFilesErrorInfo

오류와 관련된 텍스트 설명 및 원시 데이터 블록을 제공합니다.
IOfflineFilesEvents 오프라인 파일과 관련된 중요한 이벤트를 보고하는 데 사용됩니다.
IOfflineFilesEvents2 오프라인 파일과 연결된 추가 이벤트를 보고하는 데 사용됩니다.
IOfflineFilesEvents3 투명하게 캐시된 항목과 연결된 이벤트를 보고하는 데 사용됩니다.
IOfflineFilesEventsFilter 게시된 이벤트의 수신자가 수신하는 이벤트 인스턴스 수를 제한하는 메커니즘을 제공합니다.
IOfflineFilesFileItem 오프라인 파일 캐시의 파일 항목을 나타냅니다.
IOfflineFilesFileSysInfo 오프라인 파일 캐시의 파일 시스템 항목과 연결된 표준 정보를 나타냅니다.
IOfflineFilesGhostInfo 오프라인 파일 캐시에 있는 항목의 고스팅 상태 나타냅니다.
IOfflineFilesItem 오프라인 파일 캐시의 단일 항목을 나타냅니다.
IOfflineFilesItemContainer 오프라인 파일 캐시의 항목 열거형 기능에 액세스하는 데 사용됩니다.
IOfflineFilesItemFilter 열거형에 적용할 필터의 instance 나타냅니다.
IOfflineFilesPinInfo 오프라인 파일 캐시에 있는 항목의 고정된 상태 나타냅니다.
IOfflineFilesPinInfo2 오프라인 파일 캐시의 항목이 부분적으로 고정되어 있는지 여부를 확인하는 메서드를 정의합니다.

IOfflineFilesProgress 긴 오프라인 파일 작업의 호출자에게 진행률을 다시 보고하는 데 사용됩니다. (IOfflineFilesProgress)
IOfflineFilesServerItem 오프라인 파일 캐시의 서버 항목을 나타냅니다.
IOfflineFilesSetting 오프라인 파일 서비스의 동작을 제어하는 설정을 나타냅니다.
IOfflineFilesShareInfo 캐시된 항목에 대한 공유 관련 정보를 제공합니다.
IOfflineFilesShareItem 오프라인 파일 캐시의 공유 항목을 나타냅니다.
IOfflineFilesSimpleProgress 긴 오프라인 파일 작업의 호출자에게 진행률을 다시 보고하는 데 사용됩니다. (IOfflineFilesSimpleProgress)
IOfflineFilesSuspend 오프라인 파일 캐시에서 공유 루트 또는 디렉터리 트리를 일시 중단하거나 해제합니다.
IOfflineFilesSuspendInfo 항목이 일시 중단되었는지 여부와 일시 중단된 루트인지 여부를 결정합니다.
IOfflineFilesSyncConflictHandler IOfflineFilesCache::Synchronize 메서드를 호출하는 클라이언트에서 동기화 충돌이 감지될 때 충돌 해결 전략을 규정하는 데 사용됩니다.
IOfflineFilesSyncErrorInfo 동기화 오류가 발생한 항목에 대한 세부 정보를 전달하기 위해 IOfflineFilesSyncProgress::SyncItemResult 메서드와 함께 제공됩니다.
IOfflineFilesSyncErrorItemInfo 동기화 오류와 관련된 항목에 대한 파일 특성, 시간 정보 및 파일 크기를 제공합니다.

IOfflineFilesSyncProgress 동기화 및 동기화 관련 작업 중에 호출자에게 진행률을 다시 보고하는 데 사용됩니다.
IOfflineFilesTransparentCacheInfo 투명하게 캐시된 항목과 연결된 정보를 나타냅니다.

열거형

OFFLINEFILES_CACHING_MODE IOfflineFilesCache::IsPathCacheable 및 IOfflineFilesShareInfo::GetShareCachingMode와 같은 메서드에 사용되는 캐싱 모드에 대해 설명합니다.
OFFLINEFILES_COMPARE IOfflineFilesItemFilter::GetTimeFilter 메서드에서 수행할 비교 유형을 지정합니다.
OFFLINEFILES_CONNECT_STATE 오프라인 파일 캐시에 있는 항목의 연결 상태를 설명합니다.
OFFLINEFILES_EVENTS 이벤트 싱크에서 수신하거나 제외할 이벤트를 설명하는 이벤트 식별자 코드입니다.
OFFLINEFILES_ITEM_COPY 항목의 로컬, 원격 또는 원본 복사본을 쿼리할지 여부를 지정합니다.
OFFLINEFILES_ITEM_TIME 사용할 캐시 항목과 연결된 시간 값을 지정합니다.
OFFLINEFILES_ITEM_TYPE 오프라인 파일 캐시의 항목 유형을 식별합니다.
OFFLINEFILES_OFFLINE_REASON 항목이 오프라인인 이유를 나타냅니다.
OFFLINEFILES_OP_RESPONSE

항목 처리를 계속, 다시 시도 또는 중지할지 여부를 지정합니다.
OFFLINEFILES_PATHFILTER_MATCH 이벤트가 필터와 얼마나 가깝게 일치해야 하는지 지정합니다.
OFFLINEFILES_SETTING_VALUE_TYPE IOfflineFilesSetting::GetValueType 메서드에서 반환된 데이터 형식을 식별합니다.
OFFLINEFILES_SYNC_CONFLICT_RESOLVE IOfflineFilesSyncConflictHandler::ResolveConflict 메서드에서 반환된 충돌 해결 코드를 식별합니다.
OFFLINEFILES_SYNC_OPERATION 동기화 오류가 발생했을 때 수행된 동기화 작업의 유형을 나타냅니다.
OFFLINEFILES_SYNC_STATE 오프라인 파일 항목의 동기화 상태를 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ctffunc.h 헤더

아티클2023. 08. 22.

이 헤더는 Text Services Framework에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 텍스트 서비스 프레임워크 비교

ctffunc.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumTfCandidates IEnumTfCandidates 인터페이스는 텍스트 서비스에서 구현되며 TSF 관리자가 후보 문자열 개체의 열거형을 제공하는 데 사용됩니다.
IEnumTfLatticeElements IEnumTfLatticeElements 인터페이스는 TSF 관리자에 의해 구현되어 격자 요소의 열거형을 제공합니다.
ITfCandidateList ITfCandidateList 인터페이스는 텍스트 서비스에서 구현되며 TSF 관리자 또는 클라이언트(애플리케이션 또는 기타 텍스트 서비스)에서 후보 문자열 개체를 가져오고 조작하는 데 사용됩니다.
ITfCandidateString ITfCandidateString 인터페이스는 텍스트 서비스에서 구현되며 TSF 관리자 또는 클라이언트에서 후보 문자열 개체에 대한 정보를 가져오는 데 사용됩니다.
ITfFnAdviseText ITfFnAdviseText 인터페이스는 텍스트 서비스에서 구현되며 TSF 관리자가 컨텍스트의 텍스트 또는 격자 요소가 변경될 때 알림을 제공하는 데 사용됩니다.
ITfFnBalloon ITfFnBalloon 인터페이스는 텍스트 서비스에서 구현되며 애플리케이션 또는 다른 텍스트 서비스에서 텍스트 서비스가 언어 모음에 추가하는 풍선 항목을 업데이트하는 데 사용됩니다.
ITfFnConfigure ITfFnConfigure 인터페이스는 텍스트 서비스가 구성 대화 상자를 표시할 수 있도록 Text Services 제어판 애플리케이션을 사용하도록 텍스트 서비스에 의해 구현됩니다.

ITfFnConfigureRegisterEudc ITfFnConfigureRegisterEudc 인터페이스는 지정된 EUDC에 대한 키 시퀀스를 등록하는 UI를 제공하기 위해 텍스트 서비스에 의해 구현됩니다.
ITfFnConfigureRegisterWord ITfFnConfigureRegisterWord 인터페이스는 텍스트 서비스에서 구현되어 IME(활성 입력 방법 편집기)가 텍스트 서비스에 단어 등록 대화 상자를 표시하도록 합니다.
ITfFnGetLinguisticAlternates ITfFnGetLinguisticAlternates 인터페이스는 텍스트 서비스 및/또는 TSF 관리자에 의해 구현되어 매개 변수로 전달된 지정된 범위 내의 텍스트에 대한 언어 대체를 제공합니다.
ITfFnGetPreferredTouchKeyboardLayout ITfFnGetPreferredTouchKeyboardLayout 인터페이스는 텍스트 서비스에서 구현되어 받은 편지함 Windows 8 터치 키보드에서 지원하는 특정 키보드 레이아웃의 사용을 지정합니다.
ITfFnGetSAPIObject ITfFnGetSAPIObject 인터페이스는 SAPI(Speech API) 텍스트 서비스에 의해 구현됩니다. 이 인터페이스는 TSF 관리자 또는 클라이언트(애플리케이션 또는 기타 텍스트 서비스)에서 다양한 SAPI 개체를 가져오는 데 사용됩니다.
ITfFnLangProfileUtil ITfFnLangProfileUtil 인터페이스는 음성 텍스트 서비스에서 구현되며 음성 텍스트 서비스에 대한 유틸리티 메서드를 제공하는 데 사용됩니다.
ITfFnLMInternal ITfFnLMInternal 인터페이스는 사용되지 않습니다.
ITfFnLMProcessor ITfFnLMProcessor 인터페이스는 언어 모델 텍스트 서비스에서 구현되며 애플리케이션 또는 텍스트 서비스에서 대체 언어 모델 처리를 사용하도록 설정하는 데 사용됩니다.
ITfFnPlayBack ITfFnPlayBack 인터페이스는 SAPI(Speech API) 텍스트 서비스에 의해 구현됩니다. 이 인터페이스는 TSF 관리자 또는 클라이언트(애플리케이션 또는 기타 텍스트 서비스)에서 음성 입력 텍스트에 대한 오디오 데이터를 제어하는 데 사용됩니다.
ITfFnPropertyUIStatus

ITfFnPropertyUIStatus 인터페이스는 텍스트 서비스에서 구현되며 애플리케이션 또는 텍스트 서비스에서 텍스트 서비스 속성 UI의 상태 가져오고 설정하는 데 사용됩니다.
ITfFnReconversion ITfFnReconversion 인터페이스는 텍스트 서비스에 의해 구현되며 TSF 관리자 또는 클라이언트에서 텍스트 서비스에서 제공하는 텍스트의 재구성을 지원하는 데 사용됩니다.
ITfFnSearchCandidateProvider IME(입력 방법 편집기)에서 통합 검색 환경을 사용하도록 설정합니다.
ITfFnShowHelp ITfFnShowHelp 인터페이스는 언어 모음이 언어 모음 도움말 메뉴에 텍스트 서비스에 대한 도움말 명령을 배치할 수 있도록 텍스트 서비스에 의해 구현됩니다.
ITfIntegratableCandidateListUIElement 텍스트 서비스 및 IME(입력 메서드 편집기)가 UI 없는 모드 동작을 조정할 수 있도록 합니다.
ITfLMLattice ITfLMLattice 인터페이스는 음성 텍스트 서비스에서 lattice 요소 속성에 대한 정보를 제공하기 위해 구현되며 클라이언트(애플리케이션 또는 기타 텍스트 서비스)에서 사용됩니다.
IUIManagerEventSink IUIManagerEventSink 인터페이스는 IME UI 통합을 지원하는 앱에서 구현되어 IME UI 모양에 대한 알림을 받습니다.

구조체

TF_LMLATTELEMENT TF_LMLATTELEMENT 구조체에는 격자 요소에 대한 정보가 포함됩니다. 격자 요소는 음성 인식에 사용됩니다. 이 구조체는 IEnumTfLatticeElements::Next 메서드와 함께 사용됩니다.

열거형

TfCandidateResult TfCandidateResult 열거형의 요소는 ITfCandidateList::SetResult 메서드와 함께 사용하여 지정된 후보 문자열에서 수행된 재구성 작업의 결과를 지정합니다.
TfIntegratableCandidateListSelectionStyle TfIntegratableCandidateListSelectionStyle 열거형의 요소는 통합 가능한 후보 목록 선택 스타일을 지정합니다.
TfSapiObject TfSapiObject 열거형의 요소는 ITfFnGetSAPIObject::Get 메서드와 함께 사용하여 특정 유형의 SAPI(Speech API) 개체를 지정합니다.
TKBLayoutType TKBLayoutType 열거형의 요소는 ITfFnGetPreferredTouchKeyboardLayout::GetLayout 호출에서 IME에 의해 전달되어 레이아웃 유형을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ctfspui.h 헤더

아티클2023. 08. 22.

이 헤더는 Text Services Framework에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 텍스트 서비스 프레임워크 비교

ctfspui.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ITfSpeechUIServer ITfSpeechUIServer 인터페이스는 TSF 언어 모음에서 음성 관련 사용자 인터페이스를 관리합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ctfutb.h 헤더

아티클2023. 08. 22.

이 헤더는 Text Services Framework에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 텍스트 서비스 프레임워크 비교

ctfutb.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumTfLangBarItems IEnumTfLangBarItems 인터페이스는 TSF 관리자에 의해 구현되어 언어 모음 항목 개체의 열거형을 제공합니다.
ITfLangBarEventSink ITfLangBarEventSink 인터페이스는 애플리케이션 또는 텍스트 서비스에서 구현되며 언어 모음에서 언어 모음에서 발생하는 특정 이벤트에 대한 알림을 제공하는 데 사용됩니다.
ITfLangBarItem ITfLangBarItem 인터페이스는 언어 모음 항목 공급자에 의해 구현되며 언어 모음 관리자가 언어 모음 항목에 대한 자세한 정보를 가져오는 데 사용됩니다.
ITfLangBarItemBalloon ITfLangBarItemBalloon 인터페이스는 애플리케이션 또는 텍스트 서비스에서 구현되며 언어 모음 관리자가 언어 모음의 풍선 항목과 관련된 정보를 가져오는 데 사용됩니다.
ITfLangBarItemBitmap ITfLangBarItemBitmap 인터페이스는 애플리케이션 또는 텍스트 서비스에서 구현되며 언어 모음 관리자가 언어 모음의 비트맵 항목과 관련된 정보를 가져오는 데 사용됩니다.
ITfLangBarItemBitmapButton ITfLangBarItemBitmapButton 인터페이스는 언어 모음 비트맵 단추 공급자에 의해 구현되며 언어 모음 관리자가 언어 모음의 비트맵 단추 항목과 관련된 정보를 가져오는 데 사용됩니다.
ITfLangBarItemButton ITfLangBarItemButton 인터페이스는 언어 모음 단추 공급자에 의해 구현되며 언어 모음 관리자가 언어 모음의 단추 항목에 대한 정보를 가져오는 데 사용됩니다.

ITfLangBarItemMgr ITfLangBarItemMgr 인터페이스는 언어 모음에서 구현되며 텍스트 서비스에서 언어 모음의 항목을 관리하는 데 사용됩니다.
ITfLangBarItemSink ITfLangBarItemSink 인터페이스는 언어 모음에 의해 구현되며 언어 모음 항목 공급자가 언어 모음 항목의 변경 내용을 언어 모음에 알리는 데 사용됩니다.
ITfLangBarMgr ITfLangBarMgr 인터페이스는 TSF 관리자에 의해 구현되며 텍스트 서비스에서 이벤트 싱크 알림을 관리하고 부동 언어 모음 표시줄 표시 설정을 구성하는 데 사용됩니다. 인터페이스 ID가 IID_ITfLangBarMgr.
ITfMenu ITfMenu 인터페이스는 언어 모음에서 구현되며 언어 모음 단추 공급자가 언어 모음에서 단추에 대해 표시할 메뉴에 항목을 추가하는 데 사용됩니다.
ITfSystemDeviceTypeLangBarItem ITfSystemDeviceTypeLangBarItem 인터페이스는 시스템 언어 모음 항목에 의해 구현되며 애플리케이션 또는 텍스트 서비스에서 시스템 항목이 아이콘을 표시하는 방법을 제어하는 데 사용됩니다.
ITfSystemLangBarItem ITfSystemLangBarItem 인터페이스는 시스템 언어 모음 메뉴에서 구현되며 시스템 언어 모음 확장에서 메뉴에 표시되는 아이콘 및/또는 도구 설명 문자열을 수정하는 데 사용됩니다.
ITfSystemLangBarItemSink ITfSystemLangBarItemSink 인터페이스는 시스템 언어 모음 메뉴 확장에 의해 구현되며 기존 시스템 언어 모음 메뉴에 메뉴 항목을 추가할 수 있도록 시스템 언어 모음 메뉴(호스트)에서 사용됩니다.
ITfSystemLangBarItemText ITfSystemLangBarItemText 인터페이스는 시스템 언어 모음에서 구현되며 시스템 언어 모음 확장에서 메뉴에 표시되는 설명을 수정하는 데 사용됩니다.

구조체

TF_LANGBARITEMINFO TF_LANGBARITEMINFO 구조체는 언어 모음 항목에 대한 정보를 보관하는 데 사용됩니다.
TF_LBBALLOONINFO TF_LBBALLOONINFO 구조에는 언어 모음 풍선 항목에 대한 정보가 포함됩니다.

열거형

TfLBBalloonStyle TfLBBalloonStyle 열거형의 요소는 언어 막대 풍선 스타일을 지정하는 데 사용됩니다.
TfLBIClick TfLBIClick 열거형의 요소는 도구 모음 항목을 클릭하는 데 사용된 마우스 단추를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ctxtcall.h 헤더

아티클2023. 08. 22.

이 헤더는 COM(구성 요소 개체 모델)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)

ctxtcall.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IContextCallback 특정 COM+ 개체 컨텍스트 내에서 함수를 실행하는 메커니즘을 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d2d1.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct2D에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct2D

d2d1.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID2D1Bitmap ID2D1RenderTarget에 바인딩된 비트맵을 나타냅니다.
ID2D1BitmapBrush 비트맵으로 영역을 그립니다. (ID2D1BitmapBrush)
ID2D1BitmapRenderTarget CreateCompatibleRenderTarget 메서드에서 만든 중간 텍스처로 렌더링합니다.
ID2D1Brush 영역을 그리는 개체를 정의합니다. ID2D1Brush에서 파생되는 인터페이스는 영역이 그려지는 방식을 설명합니다.
ID2D1DCRenderTarget GDI 디바이스 컨텍스트에 명령을 그리는 문제를 해결합니다.
ID2D1DrawingStateBlock 렌더링 대상:_the 앤티앨리어싱 모드, 변환, 태그 및 텍스트 렌더링 옵션의 그리기 상태를 나타냅니다.
ID2D1EllipseGeometry 타원을 나타냅니다.
ID2D1Factory Direct2D 리소스를 만듭니다. (ID2D1Factory)

ID2D1GdiInteropRenderTarget GDI 그리기 명령을 수락할 수 있는 디바이스 컨텍스트에 대한 액세스를 제공합니다.
ID2D1Geometry 기하 도형을 조작하고 측정하기 위한 도우미 메서드 집합을 정의하고 기하 도형 리소스를 나타냅니다. ID2D1Geometry에서 상속되는 인터페이스는 특정 셰이프를 정의합니다.
ID2D1GeometryGroup 다른 ID2D1Geometry 개체로 구성된 복합 기하 도형을 나타냅니다.
ID2D1GeometrySink 선, 호, 입방형 베지어 곡선 및 4차원 베지어 곡선을 포함할 수 있는 기하학적 경로를 설명합니다.
ID2D1GradientStopCollection 선형 및 방사형 그라데이션 브러시에 대한 D2D1_GRADIENT_STOP 개체의 컬렉션을 나타냅니다.
ID2D1HwndRenderTarget 그리기 지침을 창에 렌더링합니다.
ID2D1Image 임의의 2D 평면을 채울 수 있는 픽셀의 생산자를 나타냅니다. (ID2D1Image)
ID2D1Layer 레이어를 렌더링하는 데 필요한 백업 저장소를 나타냅니다.
ID2D1LinearGradientBrush 선형 그라데이션으로 영역을 그립니다.
ID2D1Mesh 삼각형 목록을 형성하는 꼭짓점 집합을 나타냅니다.
ID2D1PathGeometry 호, 곡선 및 선으로 구성될 수 있는 복잡한 도형을 나타냅니다.
ID2D1RadialGradientBrush 방사형 그라데이션으로 영역을 그립니다.

ID2D1RectangleGeometry 2차원 사각형을 설명합니다.
ID2D1RenderTarget 그리기 명령을 받을 수 있는 개체를 나타냅니다. ID2D1RenderTarget에서 상속되는 인터페이스는 다양한 방식으로 수신하는 그리기 명령을 렌더링합니다.
ID2D1Resource Direct2D 그리기 리소스를 나타냅니다.
ID2D1RoundedRectangleGeometry 둥근 사각형을 설명합니다.
ID2D1SimplifiedGeometrySink 이차 베지어 곡선 또는 호를 포함하지 않는 기하학적 경로를 설명합니다.
ID2D1SolidColorBrush 단색으로 영역을 그립니다.
ID2D1StrokeStyle 스트로크에 대한 대문자, 마이터 제한, 선 조인 및 대시 정보를 설명합니다. (ID2D1StrokeStyle)
ID2D1TessellationSink ID2D1Mesh 개체를 삼각형으로 채웁니다.
ID2D1TransformedGeometry 변환된 기하 도형을 나타냅니다.

Functions

D2D1CreateFactory D2D1CreateFactory(D2D1_FACTORY_TYPE,REFIID,D2D1_FACTORY_OPTIONS*,void**)는 Direct2D 리소스를 만드는 데 사용할 수 있는 팩터리 개체를 만듭니다.

D2D1CreateFactory D2D1CreateFactory(D2D1_FACTORY_TYPE,REFIID,void**) 함수는 Direct2D 리소스를 만드는 데 사용할 수 있는 팩터리 개체를 만듭니다.
D2D1InvertMatrix 지정된 행렬을 반전하려고 시도합니다.
D2D1IsMatrixInvertible 지정된 행렬을 반전할 수 있는지 여부를 나타냅니다.
D2D1MakeRotateMatrix 지정된 지점에 대해 지정된 각도로 회전하는 회전 변환을 만듭니다.
D2D1MakeSkewMatrix 지정된 x축 각도, y축 각도 및 중심점을 갖는 기울이기 변환을 만듭니다.

구조체

D2D1_ARC_SEGMENT 두 점 사이의 타원형 호를 설명합니다.
D2D1_BEZIER_SEGMENT 두 점 사이에 그려진 입방형 베지어 세그먼트를 나타냅니다.
D2D1_BITMAP_BRUSH_PROPERTIES ID2D1BitmapBrush의 확장 모드 및 보간 모드에 대해 설명합니다. (D2D1_BITMAP_BRUSH_PROPERTIES)
D2D1_BITMAP_PROPERTIES 비트맵의 픽셀 형식 및 dpi를 설명합니다.
D2D1_BRUSH_PROPERTIES 브러시의 불투명도 및 변환에 대해 설명합니다.

D2D1_DRAWING_STATE_DESCRIPTION 렌더링 대상의 그리기 상태를 설명합니다.
D2D1_ELLIPSE 타원의 중심점, x-반지름 및 y 반경을 포함합니다.
D2D1_FACTORY_OPTIONS ID2D1Factory 개체의 디버깅 수준을 포함합니다.
D2D1_GRADIENT_STOP 그라데이션 중지점의 위치와 색을 포함합니다.
D2D1_HWND_RENDER_TARGET_PROPERTIES ID2D1HwndRenderTarget에 대한 HWND, 픽셀 크기 및 프레젠테이션 옵션을 포함합니다.
D2D1_LAYER_PARAMETERS 콘텐츠 범위, 마스크 정보, 불투명도 설정 및 레이어 리소스에 대한 기타 옵션을 포함합니다. (D2D1_LAYER_PARAMETERS)
D2D1_LINEAR_GRADIENT_BRUSH_PROPERTIES ID2D1LinearGradientBrush에 대한 그라데이션 축의 시작점과 엔드포인트를 포함합니다.
D2D1_QUADRATIC_BEZIER_SEGMENT 4차원 베지어 세그먼트의 제어점 및 끝점을 포함합니다.
D2D1_RADIAL_GRADIENT_BRUSH_PROPERTIES ID2D1RadialGradientBrush에 대한 그라데이션 타원의 그라데이션 원점 오프셋과 그라데이션 타원의 크기와 위치를 포함합니다.
D2D1_RENDER_TARGET_PROPERTIES 렌더링 옵션(하드웨어 또는 소프트웨어), 픽셀 형식, DPI 정보, 원격 옵션 및 렌더링 대상에 대한 Direct3D 지원 요구 사항이 포함되어 있습니다.
D2D1_ROUNDED_RECT 둥근 사각형의 차원과 모서리 반경을 포함합니다.
D2D1_STROKE_STYLE_PROPERTIES

셰이프를 윤곽선으로 표시하는 스트로크를 설명합니다. (D2D1_STROKE_STYLE_PROPERTIES)
D2D1_TRIANGLE 삼각형을 설명하는 세 가지 꼭짓점을 포함합니다.

열거형

D2D1_ANTIALIAS_MODE 텍스트가 아닌 기본 형식의 가장자리를 렌더링하는 방법을 지정합니다.
D2D1_ARC_SIZE 호가 180도보다 커야 하는지 여부를 지정합니다.
D2D1_BITMAP_INTERPOLATION_MODE 이미지 크기를 조정하거나 회전할 때 사용되는 알고리즘을 지정합니다.
D2D1_CAP_STYLE 선 또는 세그먼트 끝의 셰이프를 설명합니다.
D2D1_COMBINE_MODE 두 기하 도형을 결합하는 데 사용할 수 있는 여러 메서드를 지정합니다.
D2D1_COMPATIBLE_RENDER_TARGET_OPTIONS 호환되는 렌더링 대상을 만들 때 지원되는 추가 기능을 지정합니다. 이 열거형은 멤버 값의 비트 조합을 허용합니다.
D2D1_DASH_STYLE 스트로크의 대시 및 간격 시퀀스를 설명합니다.
D2D1_DC_INITIALIZE_MODE 디바이스 컨텍스트가 렌더링 대상에서 검색될 때 GDI 렌더링을 위해 초기화하는 방법을 지정합니다.
D2D1_DEBUG_LEVEL Direct2D 디버그 계층에서 제공하는 정보 유형을 나타냅니다.

D2D1_DRAW_TEXT_OPTIONS 텍스트 맞춤을 표시하지 않거나 레이아웃 사각형에 클리핑을 사용할 수 있는지 여부를 지정합니다. 이 열거형은 멤버 값의 비트 조합을 허용합니다.
D2D1_EXTEND_MODE 브러시가 일반 콘텐츠 영역 외부의 영역을 그리는 방법을 지정합니다.
D2D1_FACTORY_TYPE Direct2D가 여러 스레드에서 안전하게 액세스할 수 있도록 ID2D1Factory 및 만든 리소스에 대한 동기화를 제공하는지 여부를 지정합니다.
D2D1_FEATURE_LEVEL 렌더링 대상의 하드웨어 렌더링에 필요한 최소 DirectX 지원에 대해 설명합니다.
D2D1_FIGURE_BEGIN 특정 ID2D1SimplifiedGeometrySink 그림이 채워져 있는지 빈지 여부를 나타냅니다.
D2D1_FIGURE_END 특정 ID2D1SimplifiedGeometrySink 그림이 열려 있는지 닫혀 있는지 여부를 나타냅니다.
D2D1_FILL_MODE 기하 도형 또는 그림의 교차 영역을 결합하여 복합 기하 도형의 영역을 형성하는 방법을 지정합니다.
D2D1_GAMMA 보간에 사용되는 감마를 지정합니다.
D2D1_GEOMETRY_RELATION 한 기하 도형 개체가 다른 기하 도형 개체와 공간적으로 어떻게 관련되어 있는지 설명합니다.
D2D1_GEOMETRY_SIMPLIFICATION_OPTION 기하 도형을 ID2D1SimplifiedGeometrySink로 단순화하는 방법을 지정합니다.
D2D1_LAYER_OPTIONS 레이어 리소스를 적용하여 레이어를 만들 때 적용할 수 있는 옵션을 지정합니다.
D2D1_LINE_JOIN

두 선 또는 세그먼트를 연결하는 모양을 설명합니다.
D2D1_OPACITY_MASK_CONTENT 불투명 마스크에 그래픽 또는 텍스트가 포함되어 있는지 여부를 설명합니다. Direct2D는 이 정보를 사용하여 불투명 마스크를 혼합할 때 사용할 감마 공간을 결정합니다.
D2D1_PATH_SEGMENT 세그먼트를 스트로크해야 하는지 여부와 이 세그먼트와 이전 세그먼트 간의 조인이 원활해야 하는지 여부를 나타냅니다. 이 열거형은 멤버 값의 비트 조합을 허용합니다.
D2D1_PRESENT_OPTIONS 렌더링 대상이 콘텐츠를 표시할 때 동작하는 방식을 설명합니다. 이 열거형은 멤버 값의 비트 조합을 허용합니다.
D2D1_RENDER_TARGET_TYPE 렌더링 대상이 하드웨어 또는 소프트웨어 렌더링을 사용하는지 또는 Direct2D가 렌더링 모드를 선택해야 하는지 여부를 설명합니다.
D2D1_RENDER_TARGET_USAGE 렌더링 대상을 원격으로 사용하는 방법과 GDI와 호환되어야 하는지 여부를 설명합니다. 이 열거형은 멤버 값의 비트 조합을 허용합니다.
D2D1_SWEEP_DIRECTION 타원형 호가 그려지는 방향을 정의합니다.
D2D1_TEXT_ANTIALIAS_MODE 텍스트 그리기에 사용되는 앤티앨리어싱 모드를 설명합니다.
D2D1_WINDOW_STATE 창이 가려지는지 여부를 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d2d1_1.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct2D에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct2D

d2d1_1.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID2D1Bitmap1 ID2D1DeviceContext의 표면으로 사용하거나 시스템 메모리에 매핑할 수 있고 추가 색 컨텍스트 정보를 포함할 수 있는 비트맵을 나타냅니다.
ID2D1BitmapBrush1 비트맵으로 영역을 그립니다. (ID2D1BitmapBrush1)
ID2D1ColorContext ID2D1Bitmap1 개체와 함께 사용할 수 있는 색 컨텍스트를 나타냅니다.
ID2D1CommandList 기록 및 재생할 수 있는 명령 시퀀스를 나타냅니다.
ID2D1CommandSink 명령 싱크는 명령 목록에 기록된 명령의 재생을 받으려는 경우 애플리케이션에 대해 구현됩니다.
ID2D1Device 개체와 디바이스 컨텍스트를 함께 사용할 수 있는 리소스 도메인을 나타냅니다. (ID2D1Device)
ID2D1DeviceContext 대상에 렌더링하는 데 사용되는 상태 및 명령 버퍼 집합을 나타냅니다.
ID2D1DrawingStateBlock1 기존 앤티앨리어스 모드, 변환, 태그 및 텍스트 렌더링 모드 외에도 기본 혼합의 기능을 추가하는 그리기 상태 블록의 구현입니다.

ID2D1Effect Direct2D의 기본 이미지 처리 구문을 나타냅니다.
ID2D1Factory1 Direct2D 리소스를 만듭니다. (ID2D1Factory1)
ID2D1GdiMetafile WMF, EMF 또는 EMF+ 메타파일을 래핑하는 Direct2D 리소스입니다.
ID2D1GdiMetafileSink 메타파일을 재생할 수 있는 개발자 구현 인터페이스입니다.
ID2D1GradientStopCollection1 선형 및 방사형 그라데이션 브러시에 대한 D2D1_GRADIENT_STOP 개체의 컬렉션을 나타냅니다. 그라데이션 중지 컬렉션에 추가된 모든 새 매개 변수에 대한 get 메서드를 제공합니다.
ID2D1ImageBrush ID2D1Image를 기반으로 하는 브러시를 나타냅니다.
ID2D1Multithread Direct2D가 여러 스레드를 사용하는 앱에서 전용 리소스 액세스를 제어하는 데 사용하는 Direct2D 팩터리의 잠금 메커니즘입니다.
ID2D1PathGeometry1 ID2D1PathGeometry1 인터페이스는 ID2D1PathGeometry에 기능을 추가합니다. 특히 기하 도형별 ComputePointAndSegmentAtLength 메서드 경로를 제공합니다.
ID2D1PrintControl ID2D1CommandList에 저장된 Direct2D 기본 형식을 고정 페이지 표현으로 변환합니다. 그런 다음 인쇄 하위 시스템에서 기본 형식을 사용합니다.
ID2D1속성 데이터 기반 애플리케이션이 Direct2D 효과의 상태를 수정할 수 있도록 하는 런타임 바인딩 가능 및 검색 가능한 속성 집합을 나타냅니다.
ID2D1StrokeStyle1 스트로크에 대한 대문자, 마이터 제한, 선 조인 및 대시 정보를 설명합니다. (ID2D1StrokeStyle1)

Functions

D2D1ConvertColorSpace 지정된 색을 한 색 영역에서 다른 색으로 변환합니다.
D2D1CreateDevice 제공된 DXGI 디바이스와 연결된 새 Direct2D 디바이스를 만듭니다.
D2D1CreateDeviceContext DXGI 표면과 연결된 새 Direct2D 디바이스 컨텍스트를 만듭니다.
D2D1SinCos 각도의 사인과 코사인을 반환합니다.
D2D1Tan 각도의 탄젠트를 반환합니다.
D2D1Vec3Length 3차원 벡터의 길이를 반환합니다.

콜백 함수

PD2D1_EFFECT_FACTORY 효과의 구현에 대해 설명합니다.

구조체

D2D1_BITMAP_BRUSH_PROPERTIES1 ID2D1BitmapBrush의 확장 모드 및 보간 모드에 대해 설명합니다. (D2D1_BITMAP_BRUSH_PROPERTIES1)

D2D1_BITMAP_PROPERTIES1 이 구조를 사용하면 사용 가능한 비트맵 옵션 및 색 컨텍스트 정보를 사용하여 ID2D1Bitmap1을 만들 수 있습니다.
D2D1_CREATION_PROPERTIES Direct2D 디바이스, 팩터리 및 디바이스 컨텍스트가 만들어지는 옵션을 지정합니다.
D2D1_DRAWING_STATE_DESCRIPTION1 디바이스 컨텍스트의 그리기 상태를 설명합니다.
D2D1_EFFECT_INPUT_DESCRIPTION 효과의 기능을 설명합니다.
D2D1_IMAGE_BRUSH_PROPERTIES 이미지 브러시 기능에 대해 설명합니다.
D2D1_LAYER_PARAMETERS1 콘텐츠 범위, 마스크 정보, 불투명도 설정 및 계층 리소스에 대한 기타 옵션을 포함합니다. (D2D1_LAYER_PARAMETERS1)
D2D1_MAPPED_RECT ID2D1Bitmap1::Map API의 매핑된 메모리에 대해 설명합니다.
D2D1_POINT_DESCRIPTION 경로 기하 도형의 점을 설명합니다.
D2D1_PRINT_CONTROL_PROPERTIES ID2D1PrintControl 개체에 대한 만들기 속성입니다.
D2D1_RENDERING_CONTROLS 이미징 효과 렌더러에 적용할 제한 사항에 대해 설명합니다.
D2D1_STROKE_STYLE_PROPERTIES1 도형을 윤곽선으로 표시하는 스트로크를 설명합니다. (D2D1_STROKE_STYLE_PROPERTIES1)

열거형

D2D1_BITMAP_OPTIONS 비트맵을 사용하는 방법을 지정합니다.
D2D1_BUFFER_PRECISION Direct2D에서 이미징 파이프라인의 비트 깊이를 나타냅니다.
D2D1_COLOR_INTERPOLATION_MODE 색을 보간하는 방법을 정의합니다.
D2D1_COLOR_SPACE 색 공간에 적용해야 하는 옵션을 정의합니다.
D2D1_COMPOSITE_MODE 모든 Direct2D 혼합 작업에 대한 혼합 모드를 지정하는 데 사용됩니다.
D2D1_DEVICE_CONTEXT_OPTIONS 수명 동안 디바이스 컨텍스트에 적용되는 옵션을 지정합니다.
D2D1_INTERPOLATION_MODE ID2D1DeviceContext::D rawImage 및 2D 아핀 변환 효과를 사용하여 이미지 크기 조정 품질을 지정하는 데 사용됩니다.
D2D1_LAYER_OPTIONS1 레이어 콘텐츠를 준비하는 방법을 지정합니다.
D2D1_MAP_OPTIONS 해당 ID2D1Bitmap1에서 매핑할 메모리를 처리하는 방법을 지정합니다.
D2D1_PRIMITIVE_BLEND 모든 Direct2D 기본 형식에 대한 기하학적 혼합 모드를 지정하는 데 사용됩니다.
D2D1_PRINT_FONT_SUBSET_MODE 인쇄하는 동안 글꼴 리소스를 하위 집합으로 설정해야 하는 시기를 정의합니다.
D2D1_PROPERTY ID2D1Effect에 대한 ID2D1Properties 인터페이스에 있는 시스템 속성의 인덱스를 지정합니다.

D2D1_PROPERTY_TYPE Direct2D 속성 인터페이스에서 지원하는 속성 유형을 지정합니다.
D2D1_STROKE_TRANSFORM_TYPE 월드 변환, 인치당 점(dpi) 및 스트로크 너비가 기본 형식을 스트로크하는 데 사용되는 펜의 모양에 미치는 영향을 정의합니다.
D2D1_SUBPROPERTY 모든 속성에 있을 수 있는 시스템 하위 속성의 인덱스를 지정합니다.
D2D1_THREADING_MODE 디바이스, 팩터리 및 디바이스 컨텍스트를 동시에 만드는 동안 사용되는 스레딩 모드를 지정합니다.
D2D1_UNIT_MODE Direct2D의 단위를 해석하는 방법을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d2d1_1helper.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct2D에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct2D

d2d1_1helper.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

Matrix4x4F Matrix4x4F 클래스는 4x4 행렬을 나타내며 행렬을 만들기 위한 편리한 메서드를 제공합니다.

Functions

BitmapBrushProperties1 D2D1_BITMAP_BRUSH_PROPERTIES1 구조를 만듭니다.
BitmapProperties1 D2D1_BITMAP_PROPERTIES1 구조를 만듭니다.
ConvertColorSpace 한 색 공간에서 다른 색 공간으로 D2D1_COLOR_F 변환합니다.
CreationProperties 루트 수준 만들기 세부 정보를 설명하는 D2D1_CREATION_PROPERTIES 반환합니다.
DrawingStateDescription1 D2D1_DRAWING_STATE_DESCRIPTION1 구조를 만듭니다.
EffectInputDescription D2D1_EFFECT_INPUT_DESCRIPTION 구조를 만듭니다.

ImageBrushProperties D2D1_IMAGE_BRUSH_PROPERTIES 구조를 만듭니다.
LayerParameters1 콘텐츠 범위, 마스크 정보, 불투명도 설정 및 레이어 리소스에 대한 기타 옵션을 포함하는 D2D1_LAYER_PARAMETERS1 구조체를 반환합니다.
Point2L 긴 정수의 점 구조체를 반환합니다.
PrintControlProperties 채워진 D2D1_PRINT_CONTROL_PROPERTIES 구조를 반환합니다.
RectL 채워진 D2D1_RECT_L 구조를 반환합니다.
RenderingControls 채워진 D2D1_RENDERING_CONTROLS 구조를 반환합니다.
SetDpiCompensatedEffectInput 디바이스 컨텍스트의 DPI가 변경되면 시각적 모양을 유지하기 위해 DPI 보정 효과를 삽입하는 동안 비트맵을 효과 입력으로 설정합니다.
StrokeStyleProperties1 채워진 D2D1_STROKE_STYLE_PROPERTIES1 구조를 반환합니다.
Vector2F 채워진 D2D1_VECTOR_2F 구조를 반환합니다.
Vector3F 채워진 D2D1_VECTOR_3F 구조를 반환합니다.
Vector4F 채워진 D2D1_VECTOR_4F 구조를 반환합니다.

구조체

TypeTraits TypeTraits(d2d1_1helper.h) 구조체에는 지정된 형식을 사용하여 데이터를 저장하는 Point, Size 및 Rect의 구현이 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d2d1_2.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct2D에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct2D

d2d1_2.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID2D1CommandSink1 이 인터페이스는 기존 ID2D1CommandSink 인터페이스와 동일한 모든 함수를 수행합니다. 또한 SetPrimitiveBlend1 메서드를 통해 새로운 기본 혼합 모드인 MIN 및 ADD에 액세스할 수 있습니다.
ID2D1Device1 개체와 디바이스 컨텍스트를 함께 사용할 수 있는 리소스 도메인을 나타냅니다. (ID2D1Device1)
ID2D1DeviceContext1 기하 도형 실현 개체를 만들고 그릴 수 있습니다.
ID2D1Factory2 Direct2D 리소스를 만듭니다. (ID2D1Factory2)
ID2D1GeometryRealization 채워지거나 스트로크된 기하 도형의 디바이스 및 변환 종속 표현을 캡슐화합니다.

Functions

D2D1ComputeMaximumScaleFactor 지정된 변환이 벡터를 확장할 수 있는 최대 요소를 계산합니다.

열거형

D2D1_RENDERING_PRIORITY 렌더링 우선 순위는 Direct2D가 렌더링 워크로드를 제한하는 정도에 영향을 줍니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d2d1_3.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct2D에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct2D

d2d1_3.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID2D1ColorContext1 색 관리 효과와 함께 사용할 색 컨텍스트를 나타냅니다.
ID2D1CommandSink2 이 인터페이스는 기존 ID2D1CommandSink1 인터페이스와 동일한 모든 함수를 수행합니다. 또한 잉크 렌더링 및 그라데이션 메시 렌더링에 액세스할 수 있습니다.
ID2D1CommandSink3 이 인터페이스는 기존 ID2D1CommandSink2 인터페이스와 동일한 모든 함수를 수행합니다. 또한 스프라이트 일괄 처리 렌더링에 액세스할 수 있습니다.
ID2D1CommandSink4 이 인터페이스는 기존 ID2D1CommandSink3 인터페이스와 동일한 모든 함수를 수행합니다. 또한 SetPrimitiveBlend2 메서드를 통해 새로운 기본 혼합 모드 MAX에 액세스할 수 있습니다.
ID2D1CommandSink5 이 인터페이스는 기존 ID2D1CommandSink4 인터페이스와 동일한 모든 함수를 수행하고 BlendImage 메서드에 액세스할 수 있도록 합니다.
ID2D1Device2 개체와 디바이스 컨텍스트를 함께 사용할 수 있는 리소스 도메인을 나타냅니다. 이 인터페이스는 기존 ID2D1Device1 인터페이스와 동일한 모든 함수를 수행합니다. 또한 ID2D1DeviceContext2 개체를 만들 수 있습니다.
ID2D1Device3 개체와 디바이스 컨텍스트를 함께 사용할 수 있는 리소스 도메인을 나타냅니다. 이 인터페이스는

ID2D1Device2 인터페이스와 동일한 모든 함수를 수행합니다. 또한 ID2D1DeviceContext3 개체를 만들 수 있습니다.
ID2D1Device4 개체와 디바이스 컨텍스트를 함께 사용할 수 있는 리소스 도메인을 나타냅니다. 이 인터페이스는 ID2D1Device3 인터페이스와 동일한 모든 함수를 수행합니다. 또한 ID2D1DeviceContext4 개체를 만들 수 있습니다.
ID2D1Device5 개체와 디바이스 컨텍스트를 함께 사용할 수 있는 리소스 도메인을 나타냅니다. 이 인터페이스는 ID2D1Device4 인터페이스와 동일한 모든 함수를 수행합니다. 또한 ID2D1DeviceContext5 개체를 만들 수 있습니다.
ID2D1Device6 개체와 디바이스 컨텍스트를 함께 사용할 수 있는 리소스 도메인을 나타냅니다. 이 인터페이스는 ID2D1Device5 인터페이스와 동일한 모든 함수를 수행하고 ID2D1DeviceContext6 개체를 만들 수 있습니다.
ID2D1DeviceContext2 이 인터페이스는 ID2D1DeviceContext1 인터페이스와 동일한 모든 기능을 수행하며 잉크 렌더링, 그라데이션 메시 렌더링 및 향상된 이미지 로드와 같은 기능을 사용할 수 있습니다.
ID2D1DeviceContext3 이 인터페이스는 ID2D1DeviceContext2 인터페이스와 동일한 모든 함수를 수행하며 스프라이트 일괄 처리를 만들고 그리는 기능을 사용할 수 있습니다.
ID2D1DeviceContext4 이 인터페이스는 ID2D1DeviceContext3 인터페이스와 동일한 모든 함수를 수행하며 새로운 유형의 색 글꼴 문자 모양을 처리하는 기능을 사용할 수 있습니다.
ID2D1DeviceContext5 이 인터페이스는 ID2D1DeviceContext4 인터페이스와 동일한 모든 함수를 수행하며 색 컨텍스트 및 Svg 문서를 만들 수 있습니다.
ID2D1DeviceContext6 이 인터페이스는 기존 ID2D1DeviceContext5 인터페이스와 동일한 모든 함수를 수행하고 BlendImage 메서드에 액세스할 수 있도록 합니다.
ID2D1Factory3

Direct2D 리소스를 만듭니다. 또한 이 인터페이스를 사용하면 ID2D1Device2 개체를 만들 수 있습니다.
ID2D1Factory4 Direct2D 리소스를 만듭니다. 또한 이 인터페이스를 사용하면 ID2D1Device3 개체를 만들 수 있습니다.
ID2D1Factory5 Direct2D 리소스를 만듭니다. 또한 이 인터페이스를 사용하면 ID2D1Device4 개체를 만들 수 있습니다.
ID2D1Factory6 Direct2D 리소스를 만듭니다. 또한 이 인터페이스를 사용하면 ID2D1Device5 개체를 만들 수 있습니다.
ID2D1Factory7 Direct2D 리소스를 만듭니다. 이 인터페이스는 ID2D1Factory6 인터페이스와 동일한 모든 함수를 수행하고 ID2D1Device6 개체를 만들 수 있습니다.
ID2D1GdiMetafile1 이 인터페이스는 기존 ID2D1GdiMetafile 인터페이스와 동일한 모든 함수를 수행합니다. 또한 메타파일 DPI 및 경계에 액세스할 수 있습니다.
ID2D1GdiMetafileSink1 이 인터페이스는 기존 ID2D1GdiMetafileSink 인터페이스와 동일한 모든 함수를 수행합니다. 또한 메타파일 레코드에 액세스할 수 있습니다.
ID2D1GradientMesh 패치로 구성된 그라데이션 메시의 디바이스 종속 표현을 나타냅니다. ID2D1DeviceContext2::CreateGradientMesh 메서드를 사용하여 ID2D1GradientMesh의 instance 만듭니다.
ID2D1ImageSource 임의의 2D 평면을 채울 수 있는 픽셀의 생산자를 나타냅니다. (ID2D1ImageSource)
ID2D1ImageSourceFromWic WIC에서 공급된 2D 픽셀 데이터를 생성합니다.
ID2D1Ink

일련의 베지어 세그먼트 및 너비로 정의된 가변 너비 잉크의 단일 연속 스트로크를 나타냅니다.
ID2D1InkStyle 잉크를 렌더링할 때 ID2D1DeviceContext2::D rawInk와 같은 메서드에서 사용할 스타일 속성의 컬렉션을 나타냅니다. 잉크 스타일은 펜 팁(펜 팁) 셰이프 및 변환을 정의합니다.
ID2D1LookupTable3D LookupTable3D 효과에 전달할 수 있는 3D 조회 테이블 데이터에 대한 컨테이너입니다.
ID2D1SpriteBatch 연결된 그리기 속성이 있는 스프라이트의 단일 그룹을 나타냅니다.
ID2D1SvgGlyphStyle 이 개체는 SVG 문자 모양을 렌더링할 때 사용되는 컨텍스트 채우기, 컨텍스트 스트로크 및 컨텍스트 값에 대한 값을 제공합니다.
ID2D1TransformedImageSource 원본 이미지 원본과 리소스를 공유하는 이미지 원본을 나타냅니다.

Functions

D2D1GetGradientMeshInteriorPointsFromCoonsPatch Coons 패치를 정의하는 점을 기반으로 그라데이션 메시 패치의 내부 점을 반환합니다.

구조체

D2D1_GRADIENT_MESH_PATCH 16개의 제어점, 4개의 모서리 색 및 경계 플래그가 있는 텐서 패치를 나타냅니다. ID2D1GradientMesh는 1개 이상의 그라데이션 메시 패치로 구성됩니다. GradientMeshPatch 함수 또는 GradientMeshPatchFromCoonsPatch 함수를 사용하여 만듭니다.
D2D1_INK_BEZIER_SEGMENT

ID2D1Ink 개체를 만드는 데 사용할 Bezier 세그먼트를 나타냅니다. 이 구조는 x 및 y 좌표 외에 반경을 포함하는 D2D1_INK_POINTs 구성된다는 점에서 D2D1_BEZIER_SEGMENT 다릅니다.
D2D1_INK_POINT D2D1_INK_BEZIER_SEGMENT 일부를 구성하는 점 반경 쌍을 나타냅니다.
D2D1_INK_STYLE_PROPERTIES ID2D1InkStyle 개체에 사용되는 일반 펜 팁 셰이프와 변환을 정의합니다.
D2D1_SIMPLE_COLOR_PROFILE 색 공간에 대한 간단한 설명입니다.
D2D1_TRANSFORMED_IMAGE_SOURCE_PROPERTIES 변환된 이미지 원본의 속성입니다.

열거형

D2D1_COLOR_BITMAP_GLYPH_SNAP_OPTION 색 비트맵 문자 모양을 렌더링할 때 픽셀 맞춤 정책을 지정합니다.
D2D1_COLOR_CONTEXT_TYPE 색 프로필이 정의되는 방법을 지정합니다.
D2D1_GAMMA1 보간 및 혼합에 사용되는 감마를 결정합니다.
D2D1_IMAGE_SOURCE_FROM_DXGI_OPTIONS CreateImageSourceFromDxgi(있는 경우)에서 수행하는 기본 변환을 제어하는 옵션 플래그입니다.
D2D1_IMAGE_SOURCE_LOADING_OPTIONS 새 ID2D1ImageSource를 만들 때 옵션 플래그를 제어합니다.
D2D1_INK_NIB_SHAPE 잉크 닙(펜 팁)의 모양을 D2D1_INK_STYLE_PROPERTIES 구조체의 일부로 지정합니다.

D2D1_ORIENTATION 이미지가 표시되는 대칭 이동 및 회전을 지정합니다.
D2D1_PATCH_EDGE_MODE 그라데이션 메시 가장자리를 렌더링하는 방법을 지정합니다.
D2D1_SPRITE_OPTIONS ID2D1DeviceContext3::D rawSpriteBatch 호출의 일부로 스프라이트 일괄 처리를 그리는 방법에 대한 추가 측면을 지정합니다.
D2D1_TRANSFORMED_IMAGE_SOURCE_OPTIONS 변환된 이미지 원본에 대한 옵션 플래그입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d2d1_3helper.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct2D에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct2D

d2d1_3helper.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

GradientMeshPatch 지정된 제어점, 색 및 경계 플래그를 포함하는 D2D1_GRADIENT_MESH_PATCH 구조를 만듭니다.
GradientMeshPatchFromCoonsPatch 지정된 Coons 패치 설명에서 D2D1_GRADIENT_MESH_PATCH 만듭니다.
InkBezierSegment D2D1_INK_BEZIER_SEGMENT 구조를 만듭니다.
InkPoint D2D1_INK_POINT 구조를 만듭니다.
InkStyleProperties D2D1_INK_STYLE_PROPERTIES 구조를 만듭니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d2d1effectauthor.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct2D에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct2D

d2d1effectauthor.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID2D1AnalysisTransform 분석 효과에 데이터를 제공합니다.
ID2D1BlendTransform 혼합 연산을 변환 그래프에 삽입할 수 있도록 하는 메서드를 제공합니다.
ID2D1BorderTransform 지정된 확장 모드를 사용하여 입력 사각형을 무한대로 확장합니다.
ID2D1BoundsAdjustmentTransform 이전 효과 또는 비트맵의 출력 사각형을 수정하는 효과에 대한 지원 변환입니다.
ID2D1ComputeInfo 컴퓨팅 셰이더 렌더링 패스에 대한 정보 사양을 사용하도록 설정합니다.
ID2D1ComputeTransform 컴퓨팅 셰이더를 사용하는 변환을 정의합니다.
ID2D1ConcreteTransform 고정 기능을 제공하는 효과 렌더링 시스템에서 구현하는 변환 집합을 나타냅니다.
ID2D1DrawInfo 이 인터페이스는 꼭짓점 또는 픽셀 셰이더의 GPU 렌더링 패스를 설명하는 데 사용됩니다. ID2D1DrawTransform에 전달됩니다.
ID2D1DrawTransform

GPU에서 구현된 변환에 대한 Shantzis 계산의 특수 구현입니다.
ID2D1EffectContext 효과 및 변환 작성자용 팩터리 메서드 및 기타 상태 관리를 제공합니다. (ID2D1EffectContext)
ID2D1EffectImpl 사용자 지정 효과의 인터페이스 및 동작을 효과 작성자가 지정할 수 있습니다.
ID2D1OffsetTransform 렌더링 패스를 삽입하지 않고 입력 비트맵을 오프셋하도록 효과 렌더링 시스템에 지시합니다.
ID2D1RenderInfo 모든 다양한 변환 구현에 공통적인 렌더링 정보를 설명합니다.
ID2D1ResourceTexture 변환에서 만든 리소스 텍스처를 추적합니다.
ID2D1SourceTransform 변환 파이프라인 그래프의 CPU 기반 래스터화 단계를 나타냅니다.
ID2D1Transform 변환 작성자가 구현한 모든 변환의 기본 인터페이스를 나타냅니다.
ID2D1TransformGraph 변환 노드의 그래프를 나타냅니다.
ID2D1TransformNode 변환 토폴로지의 노드에 대해 설명합니다.
ID2D1VertexBuffer mappable 1차원 꼭짓점 버퍼를 정의합니다.

콜백 함수

PD2D1_PROPERTY_GET_FUNCTION

효과에서 속성을 가져옵니다.
PD2D1_PROPERTY_SET_FUNCTION 효과에 대한 속성을 설정합니다.

구조체

D2D1_BLEND_DESCRIPTION 특정 혼합 변환에 사용할 혼합 설명을 정의합니다.
D2D1_CUSTOM_VERTEX_BUFFER_PROPERTIES 꼭짓점 셰이더 및 입력 요소 설명을 정의하여 입력 레이아웃을 정의합니다.
D2D1_FEATURE_DATA_D3D10_X_HARDWARE_OPTIONS D3D10 기능 수준의 옵션인 컴퓨팅 셰이더 지원에 대해 설명합니다.
D2D1_FEATURE_DATA_DOUBLES 셰이더에서 doubles에 대한 지원을 설명합니다.
D2D1_INPUT_DESCRIPTION 변환이 입력 텍스처에 설정할 수 있는 옵션에 대해 설명합니다.
D2D1_INPUT_ELEMENT_DESC 꼭짓점 레이아웃에 대한 단일 요소에 대한 설명입니다.
D2D1_PROPERTY_BINDING 해당 속성을 가져와서 설정하는 함수 쌍에 대한 속성 바인딩을 정의합니다.
D2D1_RESOURCE_TEXTURE_PROPERTIES 원래 리소스 텍스처를 만들 때 리소스 텍스처를 정의합니다.
D2D1_VERTEX_BUFFER_PROPERTIES 모든 꼭짓점 셰이더 정의에 대한 표준인 꼭짓점 버퍼의 속성을 정의합니다.
D2D1_VERTEX_RANGE

꼭짓점 버퍼의 전체 콘텐츠보다 작게 렌더링할 때 사용되는 꼭짓점 범위를 정의합니다.

열거형

D2D1_BLEND 색 원본 중 하나를 파생하는 방법을 지정하고 선택적으로 색 원본에 대한 사전 블렌드 작업을 지정합니다.
D2D1_BLEND_OPERATION 두 색 원본에 대한 혼합 작업을 지정합니다.
D2D1_CHANGE_TYPE 렌더러가 사용자 지정 꼭짓점 셰이더와 상호 작용하는 방식에 영향을 주는 플래그를 설명합니다. (D2D1_CHANGE_TYPE)
D2D1_CHANNEL_DEPTH 호출자가 렌더링 파이프라인에서 단계의 채널 깊이를 제어할 수 있습니다.
D2D1_FEATURE ID2D1EffectContext::CheckFeatureSupport를 사용하여 쿼리할 수 있는 기본 Direct3D 디바이스의 기능을 정의합니다.
D2D1_FILTER 변환이 입력 텍스처에서 사용하도록 선택할 수 있는 필터링 모드를 나타냅니다.
D2D1_PIXEL_OPTIONS 픽셀 셰이더 샘플링을 제한하는 방법을 나타냅니다.
D2D1_VERTEX_OPTIONS 렌더러가 사용자 지정 꼭짓점 셰이더와 상호 작용하는 방식에 영향을 주는 플래그를 설명합니다. (D2D1_VERTEX_OPTIONS)
D2D1_VERTEX_USAGE 꼭짓점 버퍼가 자주 또는 자주 변경되는지 여부를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d2d1effectauthor_1.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct2D에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct2D

d2d1effectauthor_1.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID2D1EffectContext1 효과 및 변환 작성자용 팩터리 메서드 및 기타 상태 관리를 제공합니다. (ID2D1EffectContext1)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d2d1effecthelpers.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct2D에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct2D

d2d1effecthelpers.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

StringGetter 문자열 형식 속성에 대한 member-function 속성 getter 콜백을 호출합니다.
StringSetter 문자열 형식 속성에 대한 member-function 속성 setter 콜백을 호출합니다.
ValueGetter value-type 속성에 대한 member-function 속성 setter 콜백을 호출합니다. (ValueGetter)
ValueSetter value-type 속성에 대한 member-function 속성 setter 콜백을 호출합니다. (ValueSetter)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d2d1effects.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct2D에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct2D

d2d1effects.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

열거형

D2D1_2DAFFINETRANSFORM_INTERPOLATION_MODE 이미지 크기를 조정하기 위해 2D 아핀 변환 효과와 함께 사용할 보간 모드입니다. 품질과 속도의 범위는 6가지 배율 모드입니다.
D2D1_2DAFFINETRANSFORM_PROP 2D 아핀 변환 효과의 속성에 대한 식별자입니다.
D2D1_3DPERSPECTIVETRANSFORM_INTERPOLATION_MODE 3D 원근 변환 효과가 이미지에 사용하는 보간 모드입니다. 품질과 속도의 범위는 5가지 배율 모드입니다.
D2D1_3DPERSPECTIVETRANSFORM_PROP 3D 원근 변환 효과의 속성에 대한 식별자입니다.
D2D1_3DTRANSFORM_INTERPOLATION_MODE 3D 변환 효과가 이미지에 사용하는 보간 모드입니다. 품질과 속도의 범위는 5가지 배율 모드입니다.
D2D1_3DTRANSFORM_PROP 3D 변환 효과의 속성에 대한 식별자입니다.
D2D1_ARITHMETICCOMPOSITE_PROP 산술 복합 효과의 속성에 대한 식별자입니다.
D2D1_ATLAS_PROP Atlas 효과의 속성에 대한 식별자입니다.

D2D1_BITMAPSOURCE_ALPHA_MODE 비트맵 소스 효과 출력의 알파 모드를 지정합니다.
D2D1_BITMAPSOURCE_INTERPOLATION_MODE 비트맵 소스 효과에서 이미지 크기를 조정하는 데 사용되는 보간 모드입니다.
D2D1_BITMAPSOURCE_ORIENTATION 비트맵 소스 효과로 대칭 이동 및/또는 회전 작업을 수행할지 여부를 지정합니다.
D2D1_BITMAPSOURCE_PROP 비트맵 소스 효과의 속성에 대한 식별자입니다.
D2D1_BLEND_MODE 혼합 효과에 사용되는 혼합 모드입니다.
D2D1_BLEND_PROP Blend 효과의 속성에 대한 식별자입니다.
D2D1_BORDER_EDGE_MODE 테두리 효과에 대한 에지 모드입니다.
D2D1_BORDER_MODE 자르기 효과가 소수 픽셀 좌표에 떨어지는 자르기 사각형을 처리하는 방법을 지정합니다.
D2D1_BORDER_PROP 테두리 효과의 속성에 대한 식별자입니다.
D2D1_BRIGHTNESS_PROP 밝기 효과의 속성에 대한 식별자입니다.
D2D1_CHANNEL_SELECTOR 변위 맵이 효과를 내는 색 채널을 지정하여 강도를 확장하고 이를 사용하여 이미지를 X 또는 Y 방향으로 공간적으로 대체합니다.
D2D1_COLORMANAGEMENT_ALPHA_MODE 색 관리 효과가 입력 이미지에 포함된 알파 데이터를 해석하는 방법을 나타냅니다.

D2D1_COLORMANAGEMENT_PROP 색 관리 효과의 속성에 대한 식별자입니다.
D2D1_COLORMANAGEMENT_QUALITY 색 관리 효과에 대한 변환의 품질 수준입니다.
D2D1_COLORMANAGEMENT_RENDERING_INTENT 색 관리 효과가 사용해야 하는 ICC 렌더링 의도를 지정합니다.
D2D1_COLORMATRIX_ALPHA_MODE 색 행렬 효과 출력의 알파 모드입니다.
D2D1_COLORMATRIX_PROP 색 행렬 효과의 속성에 대한 식별자입니다.
D2D1_COMPOSITE_PROP 복합 효과의 속성에 대한 식별자입니다.
D2D1_CONVOLVEMATRIX_PROP Convolve 행렬 효과의 속성에 대한 식별자입니다.
D2D1_CONVOLVEMATRIX_SCALE_MODE Convolve 행렬 효과가 이미지를 해당 커널 단위 길이로 크기 조정하는 데 사용하는 보간 모드입니다. 품질과 속도에 따라 6가지 크기 조정 모드가 있습니다.
D2D1_CROP_PROP 자르기 효과의 속성에 대한 식별자입니다.
D2D1_DIRECTIONALBLUR_OPTIMIZATION 방향 흐림 효과에 대한 최적화 모드를 지정합니다.
D2D1_DIRECTIONALBLUR_PROP 방향 흐림 효과의 속성에 대한 식별자입니다.
D2D1_DISCRETETRANSFER_PROP 불연속 전송 효과의 속성에 대한 식별자입니다.

D2D1_DISPLACEMENTMAP_PROP 변위 맵 효과의 속성에 대한 식별자입니다.
D2D1_DISTANTDIFFUSE_PROP 원거리 확산 조명 효과의 속성에 대한 식별자입니다.
D2D1_DISTANTDIFFUSE_SCALE_MODE 효과가 이미지를 해당 커널 단위 길이로 크기 조정하는 데 사용하는 보간 모드입니다. 품질과 속도에 따라 6가지 크기 조정 모드가 있습니다.
D2D1_DISTANTSPECULAR_PROP 원거리 반사 조명 효과의 속성에 대한 식별자입니다.
D2D1_DISTANTSPECULAR_SCALE_MODE 원거리 반사 조명 효과의 보간 모드는 이미지를 해당 커널 단위 길이로 크기 조정하는 데 사용합니다. 품질과 속도에 따라 6가지 크기 조정 모드가 있습니다.
D2D1_DPICOMPENSATION_INTERPOLATION_MODE DPI 보정 효과가 이미지 크기를 조정하는 데 사용하는 보간 모드입니다.
D2D1_DPICOMPENSATION_PROP DPI 보정 효과의 속성에 대한 식별자입니다.
D2D1_FLOOD_PROP 홍수 효과의 속성에 대한 식별자입니다.
D2D1_GAMMATRANSFER_PROP 감마 전송 효과의 속성에 대한 식별자입니다.
D2D1_GAUSSIANBLUR_OPTIMIZATION 가우스 흐림 효과에 대한 최적화 모드입니다.
D2D1_GAUSSIANBLUR_PROP 가우스 흐림 효과의 속성에 대한 식별자입니다.
D2D1_HISTOGRAM_PROP 히스토그램 효과의 속성에 대한 식별자입니다.

D2D1_HUEROTATION_PROP Hue 회전 효과의 속성에 대한 식별자입니다.
D2D1_LINEARTRANSFER_PROP 선형 전송 효과의 속성에 대한 식별자입니다.
D2D1_MORPHOLOGY_MODE 형태 효과의 모드입니다.
D2D1_MORPHOLOGY_PROP 형태 효과의 속성에 대한 식별자입니다.
D2D1_OPACITYMETADATA_PROP 불투명도 메타데이터 효과의 속성에 대한 식별자입니다.
D2D1_POINTDIFFUSE_PROP 점 확산 조명 효과의 속성에 대한 식별자입니다.
D2D1_POINTDIFFUSE_SCALE_MODE 점 확산 조명 효과가 이미지를 해당 커널 단위 길이로 크기 조정하는 데 사용하는 보간 모드입니다. 품질과 속도에 따라 6가지 크기 조정 모드가 있습니다.
D2D1_POINTSPECULAR_PROP 점 반사 조명 효과의 속성에 대한 식별자입니다.
D2D1_POINTSPECULAR_SCALE_MODE 점 반사 조명 효과가 이미지를 해당 커널 단위 길이로 크기 조정하는 데 사용하는 보간 모드입니다. 품질과 속도에 따라 6가지 크기 조정 모드가 있습니다.
D2D1_SATURATION_PROP 채도 효과의 속성에 대한 식별자입니다.
D2D1_SCALE_INTERPOLATION_MODE 크기 조정 효과가 이미지 크기를 조정하는 데 사용하는 보간 모드입니다. 품질과 속도의 범위는 6가지 배율 모드입니다.
D2D1_SCALE_PROP

눈금 효과의 속성에 대한 식별자입니다.
D2D1_SHADOW_OPTIMIZATION 그림자 효과에 대한 성능 최적화 수준입니다.
D2D1_SHADOW_PROP 그림자 효과의 속성에 대한 식별자입니다.
D2D1_SPOTDIFFUSE_PROP 스폿 확산 조명 효과의 속성에 대한 식별자입니다.
D2D1_SPOTDIFFUSE_SCALE_MODE 스폿 확산 조명 효과가 이미지를 해당 커널 단위 길이로 크기 조정하는 데 사용하는 보간 모드입니다. 품질과 속도에 따라 6가지 크기 조정 모드가 있습니다.
D2D1_SPOTSPECULAR_PROP 스폿 반사 조명 효과의 속성에 대한 식별자입니다.
D2D1_SPOTSPECULAR_SCALE_MODE 스폿 반사 조명 효과가 이미지를 해당 커널 단위 길이로 크기 조정하는 데 사용하는 보간 모드입니다. 품질과 속도에 따라 6가지 크기 조정 모드가 있습니다.
D2D1_TABLETRANSFER_PROP 테이블 전송 효과의 속성에 대한 식별자입니다.
D2D1_TILE_PROP 타일 효과의 속성에 대한 식별자입니다.
D2D1_TURBULENCE_NOISE 난기류 효과에 대한 난기류 노이즈 모드입니다. 프랙탈 노이즈 또는 난기류 함수를 기반으로 비트맵을 생성할지 여부를 나타냅니다.
D2D1_TURBULENCE_PROP 난기류 효과의 속성에 대한 식별자입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d2d1effects_1.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct2D에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct2D

d2d1effects_1.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

열거형

D2D1_YCBCR_CHROMA_SUBSAMPLING YCbCr 효과에 사용되는 입력 크로마 이미지의 크로마 하위 샘플링을 지정합니다.
D2D1_YCBCR_INTERPOLATION_MODE YCbCr 효과의 보간 모드를 지정합니다.
D2D1_YCBCR_PROP YCbCr 효과의 속성에 대한 식별자입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d2d1effects_2.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct2D에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct2D

d2d1effects_2.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

열거형

D2D1_CHROMAKEY_PROP Chroma-key 효과의 속성에 대한 식별자입니다.
D2D1_CONTRAST_PROP 대비 효과의 속성에 대한 식별자입니다.
D2D1_EDGEDETECTION_MODE 에지 검색 효과의 D2D1_EDGEDETECTION_PROP_MODE 속성 값입니다.
D2D1_EDGEDETECTION_PROP 에지 검색 효과의 속성에 대한 식별자입니다.
D2D1_EMBOSS_PROP Emboss 효과의 속성에 대한 식별자입니다.
D2D1_EXPOSURE_PROP 노출 효과의 속성에 대한 식별자입니다.
D2D1_HDRTONEMAP_DISPLAY_MODE HDR 톤 맵 효과의 D2D1_HDRTONEMAP_PROP_DISPLAY_MODE 속성 값을 지정하는 상수를 정의합니다.
D2D1_HDRTONEMAP_PROP HDR 톤 맵 효과의 최상위 속성을 식별하는 상수를 정의합니다.
D2D1_HIGHLIGHTSANDSHADOWS_INPUT_GAMMA

강조 표시 및 그림자 효과의 D2D1_HIGHLIGHTSANDSHADOWS_PROP_INPUT_GAMMA 속성 값입니다.
D2D1_HIGHLIGHTSANDSHADOWS_PROP 강조 표시 및 그림자 효과의 속성에 대한 식별자입니다.
D2D1_HUETORGB_INPUT_COLOR_SPACE Hue to RGB 효과의 D2D1_HUETORGB_PROP_INPUT_COLOR_SPACE 속성 값입니다.
D2D1_HUETORGB_PROP Hue에서 RGB로의 속성에 대한 식별자입니다.
D2D1_LOOKUPTABLE3D_PROP 3D 조회 테이블 효과의 속성에 대한 식별자입니다.
D2D1_POSTERIZE_PROP Posterize 효과의 속성에 대한 식별자입니다.
D2D1_RGBTOHUE_OUTPUT_COLOR_SPACE RGB to Hue 효과의 D2D1_RGBTOHUE_PROP_OUTPUT_COLOR_SPACE 속성 값입니다.
D2D1_RGBTOHUE_PROP RGB to Hue 효과의 속성에 대한 식별자입니다.
D2D1_SEPIA_PROP 세피아 효과의 속성에 대한 식별자입니다.
D2D1_SHARPEN_PROP 선명 효과의 속성에 대한 식별자입니다.
D2D1_STRAIGHTEN_PROP Straighten 효과의 속성에 대한 식별자입니다.
D2D1_STRAIGHTEN_SCALE_MODE Straighten 효과의 D2D1_STRAIGHTEN_PROP_SCALE_MODE 속성 값입니다.
D2D1_TEMPERATUREANDTINT_PROP 온도 및 색조 효과의 속성에 대한 식별자입니다.

D2D1_VIGNETTE_PROP 비네팅 효과의 속성에 대한 식별자입니다.
D2D1_WHITELEVELADJUSTMENT_PROP 흰색 수준 조정 효과의 최상위 속성을 식별하는 상수를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d2d1helper.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct2D에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct2D

d2d1helper.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

ColorF 알려진 색을 나타내는 상수 집합을 정의하고 다양한 색을 인스턴스화하는 편리한 메서드를 제공합니다.
Matrix3x2F Matrix3x2F 클래스는 3x2 행렬을 나타내며 행렬을 만들기 위한 편리한 메서드를 제공합니다.

Functions

ArcSegment D2D1_ARC_SEGMENT 구조를 만듭니다.
BezierSegment D2D1_BEZIER_SEGMENT 구조를 만듭니다.
BitmapBrushProperties D2D1_BITMAP_BRUSH_PROPERTIES 구조를 만듭니다.
BitmapProperties D2D1_BITMAP_PROPERTIES 구조를 만듭니다.
BrushProperties D2D1_BRUSH_PROPERTIES 구조를 만듭니다.

DrawingStateDescription D2D1_DRAWING_STATE_DESCRIPTION 구조를 만듭니다.
타원 D2D1_ELLIPSE 구조를 만듭니다.
FloatMax 최대 부동 소수점 값을 반환합니다.
Gradientstop D2D1_GRADIENT_STOP 구조를 만듭니다.
HwndRenderTargetProperties D2D1_HWND_RENDER_TARGET_PROPERTIES 구조를 만듭니다.
IdentityMatrix ID 행렬을 만듭니다. (IdentityMatrix)
InfiniteRect 왼쪽 위 모서리가 (음의 무한대, 음의 무한대)로 설정되고 오른쪽 아래 모서리가 (무한대, 무한대)로 설정된 사각형을 만듭니다.
LayerParameters D2D1_LAYER_PARAMETERS 구조를 만듭니다.
LinearGradientBrushProperties D2D1_LINEAR_GRADIENT_BRUSH_PROPERTIES 구조를 만듭니다.
PixelFormat D2D1_PIXEL_FORMAT 구조를 만듭니다.
Point2F 지정된 x 좌표와 y 좌표를 포함하는 D2D1_POINT_2F 구조를 만듭니다.
Point2U 지정된 x 좌표와 y 좌표를 포함하는 D2D1_POINT_2U 구조를 만듭니다.

QuadraticBezierSegment D2D1_QUADRATIC_BEZIER_SEGMENT 구조를 만듭니다.
RadialGradientBrushProperties D2D1_RADIAL_GRADIENT_BRUSH_PROPERTIES 구조를 만듭니다.
RectF 지정된 차원을 포함하는 D2D1_RECT_F 구조를 만듭니다.
RectU 지정된 차원을 포함하는 D2D1_RECT_U 구조를 만듭니다.
RenderTargetProperties D2D1_RENDER_TARGET_PROPERTIES 구조를 만듭니다.
RoundedRect D2D1_ROUNDED_RECT 구조를 만듭니다.
Sizef 지정된 너비와 높이를 포함하는 D2D1_SIZE_F 구조를 만듭니다.
SizeU 지정된 너비와 높이를 포함하는 D2D1_SIZE_U 구조를 만듭니다.
StrokeStyleProperties D2D1_STROKE_STYLE_PROPERTIES 구조를 만듭니다.

구조체

TypeTraits TypeTraits(d2d1helper.h) 구조체에는 지정된 형식을 사용하여 데이터를 저장하는 Point, Size 및 Rect의 구현이 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d2d1svg.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct2D에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct2D

d2d1svg.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID2D1SvgAttribute SVG 특성을 설명하는 인터페이스입니다.
ID2D1SvgDocument SVG 문서를 나타냅니다.
ID2D1SvgElement 모든 SVG 요소에 대한 인터페이스입니다.
ID2D1SvgPaint SVG 채우기 또는 스트로크 값을 설명하는 인터페이스입니다.
ID2D1SvgPathData SVG 경로 데이터를 설명하는 인터페이스입니다. 경로 데이터는 'path' 요소에서 'd' 특성으로 설정할 수 있습니다.
ID2D1SvgPointCollection 다각형 또는 다각형 요소의 SVG 포인트 값을 설명하는 인터페이스입니다.
ID2D1SvgStrokeDashArray SVG stroke-dasharray 값을 설명하는 인터페이스입니다.

구조체

D2D1_SVG_LENGTH SVG 길이를 나타냅니다.
D2D1_SVG_PRESERVE_ASPECT_RATIO 모든 SVG preserveAspectRatio 설정을 나타냅니다.
D2D1_SVG_VIEWBOX SVG viewBox를 나타냅니다.

열거형

D2D1_SVG_ASPECT_ALIGN SVG preserveAspectRatio 특성의 맞춤 부분입니다.
D2D1_SVG_ASPECT_SCALING SVG preserveAspectRatio 특성의 meetOrSlice 부분입니다.
D2D1_SVG_ATTRIBUTE_POD_TYPE 설정하거나 가져올 SVG POD 특성의 형식을 정의합니다.
D2D1_SVG_ATTRIBUTE_STRING_TYPE 설정하거나 가져올 SVG 문자열 특성의 형식을 정의합니다.
D2D1_SVG_DISPLAY SVG 표시 속성의 값을 지정합니다.
D2D1_SVG_LENGTH_UNITS SVG 길이의 단위를 지정합니다.
D2D1_SVG_LINE_CAP SVG stroke-linecap 속성의 값을 지정합니다.
D2D1_SVG_LINE_JOIN SVG stroke-linejoin 속성의 값을 지정합니다.

D2D1_SVG_OVERFLOW SVG 오버플로 속성의 값을 지정합니다.
D2D1_SVG_PAINT_TYPE SVG 채우기 또는 스트로크에 대한 페인트 유형을 지정합니다.
D2D1_SVG_PATH_COMMAND 명령된 경로를 나타냅니다. 각 명령은 세그먼트 데이터의 부동 소수를 참조할 수 있습니다. _ABSOLUTE 끝나는 명령은 데이터를 절대 좌표로 해석합니다. _RELATIVE 끝나는 명령은 데이터를 이전 지점을 기준으로 해석합니다.
D2D1_SVG_UNIT_TYPE SVG 그라데이션 또는 clipPath 요소에 사용되는 좌표계를 정의합니다.
D2D1_SVG_VISIBILITY SVG 표시 유형 속성의 값을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d3d10.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct3D 10 그래픽에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 10 그래픽

d3d10.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID3D10Asynchronous 이 인터페이스는 GPU에서 데이터를 비동기적으로 검색하는 메서드를 캡슐화합니다. (ID3D10Asynchronous)
ID3D10BlendState 이 혼합 상태 인터페이스는 출력 병합기 단계에 대한 Direct3D 10.0 디바이스의 혼합 상태에 액세스합니다.
ID3D10Buffer 버퍼 인터페이스는 비정형 메모리인 버퍼 리소스에 액세스합니다. 버퍼는 일반적으로 꼭짓점 또는 인덱스 데이터를 저장합니다. (ID3D10Buffer)
ID3D10Counter 이 인터페이스는 GPU 성능을 측정하기 위한 메서드를 캡슐화합니다. (ID3D10Counter)
ID3D10DepthStencilState 깊이 스텐실 상태 인터페이스는 출력 병합 단계에 대한 깊이 스텐실 테스트를 설정하는 깊이 스텐실 상태에 액세스합니다.
ID3D10DepthStencilView 깊이 스텐실 보기 인터페이스는 깊이 스텐실 테스트 중에 텍스처 리소스에 액세스합니다. (ID3D10DepthStencilView)
ID3D10Device 디바이스 인터페이스는 Direct3D 10.0에 대한 가상 어댑터를 나타냅니다. 렌더링을 수행하고 Direct3D 리소스를 만드는 데 사용됩니다.

ID3D10DeviceChild 디바이스-자식 인터페이스는 디바이스에서 사용하는 데이터에 액세스합니다. (ID3D10DeviceChild)
ID3D10GeometryShader geometry-shader 인터페이스는 geometry 셰이더 단계를 제어하는 실행 프로그램(기하 도형 셰이더)을 관리합니다. (ID3D10GeometryShader)
ID3D10InputLayout 입력 레이아웃 인터페이스는 입력 어셈블러 단계의 입력 데이터에 액세스합니다.
ID3D10Multithread 다중 스레드 인터페이스는 다중 스레드 설정에 액세스하며 스레드로부터 안전한 계층이 켜져 있는 경우에만 사용할 수 있습니다.
ID3D10PixelShader 픽셀 셰이더 인터페이스는 픽셀 셰이더 단계를 제어하는 실행 프로그램(픽셀 셰이더)을 관리합니다. (ID3D10PixelShader)
ID3D10Predicate 조건자 인터페이스는 이전 그리기 호출의 결과에 따라 기하 도형을 처리해야 하는지 여부를 결정합니다. (ID3D10Predicate)
ID3D10Query 쿼리 인터페이스는 GPU의 정보를 쿼리합니다. (ID3D10Query)
ID3D10RasterizerState 래스터라이저 상태 인터페이스는 래스터라이저 단계에 대한 래스터라이저 상태에 액세스합니다.
ID3D10RenderTargetView render-target-view 인터페이스는 렌더링 중에 액세스할 수 있는 렌더링 대상 하위 리소스를 식별합니다. (ID3D10RenderTargetView)
ID3D10Resource 리소스 인터페이스는 모든 리소스에 대한 일반적인 작업을 제공합니다. (ID3D10Resource)
ID3D10SamplerState 샘플러 상태 인터페이스는 텍스처에 대한 샘플러 상태에 액세스합니다.

ID3D10ShaderResourceView 셰이더-resource-view 인터페이스는 렌더링 중에 셰이더가 액세스할 수 있는 하위 리소스를 지정합니다. 셰이더 리소스의 예로는 상수 버퍼, 텍스처 버퍼, 텍스처 또는 샘플러가 있습니다. (ID3D10ShaderResourceView)
ID3D10Texture1D 1D 텍스처 인터페이스는 구조화된 메모리인 텍셀 데이터에 액세스합니다. (ID3D10Texture1D)
ID3D10Texture2D 2D 텍스처 인터페이스는 구조화된 메모리인 텍셀 데이터를 관리합니다. (ID3D10Texture2D)
ID3D10Texture3D 3D 텍스처 인터페이스는 구조화된 메모리인 텍셀 데이터에 액세스합니다. (ID3D10Texture3D)
ID3D10VertexShader 꼭짓점 셰이더 인터페이스는 꼭짓점 셰이더 단계를 제어하는 실행 프로그램(꼭짓점 셰이더)을 관리합니다. (ID3D10VertexShader)
ID3D10View 뷰 인터페이스는 렌더링하는 동안 파이프라인이 액세스할 수 있는 리소스 부분을 지정합니다(보기 참조).

Functions

D3D10CalcSubresource 텍스처에 대한 하위 리소스 인덱스 계산

구조체

CD3D10_BUFFER_DESC 버퍼 리소스에 대해 설명합니다. D

CD3D10_TEXTURE1D_DESC 1D 텍스처에 대해 설명합니다. D
CD3D10_TEXTURE2D_DESC 2D 텍스처를 설명합니다. D
CD3D10_TEXTURE3D_DESC CD3D10_TEXTURE3D_DESC(d3d10.h) 구조체는 3D 텍스처를 설명합니다.
D3D10_BLEND_DESC 혼합 상태를 설명합니다. (D3D10_BLEND_DESC)
D3D10_BOX 3D 상자를 정의합니다. (D3D10_BOX)
D3D10_BUFFER_DESC 버퍼 리소스에 대해 설명합니다. (D3D10_BUFFER_DESC)
D3D10_BUFFER_RTV 렌더링 대상 뷰에서 사용할 버퍼 리소스의 요소를 지정합니다.
D3D10_BUFFER_SRV 셰이더 리소스 뷰에서 사용할 버퍼 리소스의 요소를 지정합니다. (D3D10_BUFFER_SRV)
D3D10_COUNTER_DESC 카운터에 대해 설명합니다. (D3D10_COUNTER_DESC)
D3D10_COUNTER_INFO 비디오 카드 성능 카운터 기능에 대한 정보입니다. (D3D10_COUNTER_INFO)
D3D10_DEPTH_STENCIL_DESC 깊이 스텐실 상태를 설명합니다. (D3D10_DEPTH_STENCIL_DESC)
D3D10_DEPTH_STENCIL_VIEW_DESC 깊이 스텐실 보기를 사용하여 액세스할 수 있는 텍스처의 하위 리소스를 지정합니다.
D3D10_DEPTH_STENCILOP_DESC

스텐실 테스트 결과에 따라 수행할 수 있는 스텐실 작업에 대해 설명합니다.
D3D10_INPUT_ELEMENT_DESC 입력 어셈블러 단계의 단일 요소에 대한 설명입니다. (D3D10_INPUT_ELEMENT_DESC)
D3D10_MAPPED_TEXTURE2D 2D 텍스처의 하위 리소스 데이터에 대한 액세스를 제공합니다.
D3D10_MAPPED_TEXTURE3D 3D 텍스처의 하위 리소스 데이터에 대한 액세스를 제공합니다.
D3D10_QUERY_DATA_PIPELINE_STATISTICS ID3D10Asynchronous::Begin 및 ID3D10Asynchronous::End 호출 간의 그래픽 파이프라인 활동에 대한 정보를 쿼리합니다.
D3D10_QUERY_DATA_SO_STATISTICS ID3D10Asynchronous::Begin 및 ID3D10Asynchronous::End 사이에 스트림 출력 버퍼로 스트리밍되는 데이터의 양에 대한 정보를 쿼리합니다.
D3D10_QUERY_DATA_TIMESTAMP_DISJOINT 타임스탬프 쿼리의 안정성에 대한 쿼리 정보입니다. (D3D10_QUERY_DATA_TIMESTAMP_DISJOINT)
D3D10_QUERY_DESC 쿼리를 설명합니다. (D3D10_QUERY_DESC)
D3D10_RASTERIZER_DESC 래스터라이저 상태를 설명합니다.
D3D10_RENDER_TARGET_VIEW_DESC 렌더링 대상 보기를 사용하여 액세스할 수 있는 리소스의 하위 리소스를 지정합니다.
D3D10_SAMPLER_DESC 샘플러 상태를 설명합니다. (D3D10_SAMPLER_DESC)
D3D10_SHADER_RESOURCE_VIEW_DESC 셰이더 리소스 뷰에 대해 설명합니다. (D3D10_SHADER_RESOURCE_VIEW_DESC)
D3D10_SO_DECLARATION_ENTRY

출력 슬롯의 꼭짓점 버퍼에 있는 꼭짓점 요소에 대한 설명입니다. (D3D10_SO_DECLARATION_ENTRY)
D3D10_SUBRESOURCE_DATA 하위 리소스를 초기화하기 위한 데이터를 지정합니다. (D3D10_SUBRESOURCE_DATA)
D3D10_TEX1D_ARRAY_DSV 깊이 스텐실 보기에서 사용할 1D 텍스처 배열의 하위 리소스를 지정합니다.
D3D10_TEX1D_ARRAY_RTV 렌더링 대상 보기에서 사용할 1D 텍스처 배열의 하위 리소스를 지정합니다.
D3D10_TEX1D_ARRAY_SRV 셰이더 리소스 뷰에서 사용할 1D 텍스처 배열의 하위 리소스를 지정합니다.
D3D10_TEX1D_DSV 깊이 스텐실 보기에 액세스할 수 있는 1D 텍스처의 하위 리소스를 지정합니다. (D3D10_TEX1D_DSV)
D3D10_TEX1D_RTV 렌더링 대상 보기에서 사용할 1D 텍스처의 하위 리소스를 지정합니다. (D3D10_TEX1D_RTV)
D3D10_TEX1D_SRV 셰이더 리소스 뷰에서 사용할 1D 텍스처의 하위 리소스를 지정합니다. (D3D10_TEX1D_SRV)
D3D10_TEX2D_ARRAY_DSV 깊이 스텐실 보기에 액세스할 수 있는 배열 2D 텍스처의 하위 리소스를 지정합니다.
D3D10_TEX2D_ARRAY_RTV 렌더링 대상 보기에서 사용할 2D 텍스처 배열의 하위 리소스를 지정합니다.
D3D10_TEX2D_ARRAY_SRV 셰이더 리소스 뷰에서 사용할 2D 텍스처 배열의 하위 리소스를 지정합니다.
D3D10_TEX2D_DSV 깊이 스텐실 보기에 액세스할 수 있는 2D 텍스처의 하위 리소스를 지정합니다. (D3D10_TEX2D_DSV)

D3D10_TEX2D_RTV 렌더링 대상 보기에서 사용할 2D 텍스처의 하위 리소스를 지정합니다. (D3D10_TEX2D_RTV)
D3D10_TEX2D_SRV 셰이더 리소스 뷰에서 사용할 2D 텍스처의 하위 리소스를 지정합니다. (D3D10_TEX2D_SRV)
D3D10_TEX2DMS_ARRAY_DSV 깊이 스텐실 보기에 대한 다중 샘플링된 2D 텍스처 배열의 하위 리소스를 지정합니다.
D3D10_TEX2DMS_ARRAY_RTV 렌더링 대상 보기에서 사용할 다중 샘플링된 2D 텍스처 배열의 하위 리소스를 지정합니다.
D3D10_TEX2DMS_ARRAY_SRV 셰이더 리소스 뷰에서 사용할 다중 샘플링된 2D 텍스처 배열의 하위 리소스를 지정합니다.
D3D10_TEX2DMS_DSV 깊이 스텐실 보기에 액세스할 수 있는 다중 샘플링된 2D 텍스처의 하위 리소스를 지정합니다. (D3D10_TEX2DMS_DSV)
D3D10_TEX2DMS_RTV 렌더링 대상 보기에서 사용할 다중 샘플링된 2D 텍스처의 하위 리소스를 지정합니다. (D3D10_TEX2DMS_RTV)
D3D10_TEX2DMS_SRV 셰이더 리소스 뷰에서 사용할 다중 샘플링된 2D 텍스처의 하위 리소스를 지정합니다.
D3D10_TEX3D_RTV 렌더링 대상 보기에서 사용할 3D 텍스처의 하위 리소스를 지정합니다.
D3D10_TEX3D_SRV 셰이더 리소스 뷰에서 사용할 3D 텍스처의 하위 리소스를 지정합니다.
D3D10_TEXCUBE_SRV 셰이더 리소스 뷰에서 사용할 큐브 텍스처의 하위 리소스를 지정합니다. (D3D10_TEXCUBE_SRV)
D3D10_TEXTURE1D_DESC 1D 텍스처에 대해 설명합니다. (D3D10_TEXTURE1D_DESC)

D3D10_TEXTURE2D_DESC 2D 텍스처를 설명합니다. (D3D10_TEXTURE2D_DESC)
D3D10_TEXTURE3D_DESC 3D 텍스처를 설명합니다. (D3D10_TEXTURE3D_DESC)
D3D10_VIEWPORT 뷰포트의 차원을 정의합니다. (D3D10_VIEWPORT)

열거형

D3D10_ASYNC_GETDATA_FLAG ID3D10Asynchronous::GetData의 동작을 제어하는 선택적 플래그입니다.
D3D10_BIND_FLAG 파이프라인에 리소스를 바인딩하는 방법을 식별합니다. (D3D10_BIND_FLAG)
D3D10_BLEND 혼합 옵션. 혼합 옵션은 데이터 원본과 선택적 사전 혼합 작업을 식별합니다.
D3D10_BLEND_OP RGB 또는 알파 혼합 작업. (D3D10_BLEND_OP)
D3D10_CLEAR_FLAG 지울 깊이 스텐실의 부분을 지정합니다. 일반적으로 ID3D10Device::ClearDepthStencilView와 함께 사용됩니다.
D3D10_COLOR_WRITE_ENABLE 혼합하는 동안 쓰기 가능한 렌더링 대상의 각 픽셀 구성 요소를 식별합니다. (D3D10_COLOR_WRITE_ENABLE)
D3D10_COMPARISON_FUNC 비교 옵션. (D3D10_COMPARISON_FUNC)

D3D10_COUNTER 성능 카운터 유형입니다.
D3D10_COUNTER_TYPE 성능 카운터의 데이터 형식입니다. (D3D10_COUNTER_TYPE)
D3D10_CPU_ACCESS_FLAG 리소스에 허용되는 CPU 액세스 유형을 지정합니다. (D3D10_CPU_ACCESS_FLAG)
D3D10_CREATE_DEVICE_FLAG 디바이스 만들기 플래그.
D3D10_CULL_MODE 특정 방향을 향한 삼각형이 그려지지 않음을 나타냅니다. (D3D10_CULL_MODE)
D3D10_DEPTH_WRITE_MASK 깊이 데이터를 작성하기 위한 깊이 스텐실 버퍼의 부분을 식별합니다. (D3D10_DEPTH_WRITE_MASK)
D3D10_DSV_DIMENSION 깊이 스텐실 보기에서 사용되는 리소스에 액세스하는 방법을 지정합니다. (D3D10_DSV_DIMENSION)
D3D10_FILL_MODE 삼각형을 렌더링할 때 사용할 채우기 모드를 결정합니다. (D3D10_FILL_MODE)
D3D10_FILTER 텍스처 샘플링 중 필터링 옵션입니다. (D3D10_FILTER)
D3D10_FILTER_TYPE 배율 또는 축소 샘플러 필터의 유형입니다. (D3D10_FILTER_TYPE)
D3D10_FORMAT_SUPPORT 지정된 형식 및 지정된 디바이스에 대해 지원되는 리소스(ID3D10Device::CheckFormatSupport 참조).
D3D10_INPUT_CLASSIFICATION

입력 슬롯에 포함된 데이터 형식입니다. (D3D10_INPUT_CLASSIFICATION)
D3D10_MAP CPU에서 읽고 쓰기 위해 액세스할 리소스를 식별합니다. 애플리케이션은 이러한 플래그 중 하나 이상을 결합할 수 있습니다. (D3D10_MAP)
D3D10_MAP_FLAG GPU에서 사용하는 리소스에 대해 Map이 호출될 때 CPU가 응답하는 방법을 지정합니다.
D3D10_QUERY 쿼리 형식입니다. (D3D10_QUERY)
D3D10_QUERY_MISC_FLAG 기타 쿼리 동작을 설명하는 플래그입니다. (D3D10_QUERY_MISC_FLAG)
D3D10_RAISE_FLAG 연속 불가능한 예외로 오류를 발생시키는 옵션입니다. (D3D10_RAISE_FLAG)
D3D10_RESOURCE_DIMENSION 사용 중인 리소스의 유형을 식별합니다. (D3D10_RESOURCE_DIMENSION)
D3D10_RESOURCE_MISC_FLAG 리소스에 대한 덜 일반적인 다른 옵션을 식별합니다.
D3D10_RTV_DIMENSION 렌더링 대상 보기에서 사용되는 리소스에 액세스하는 방법을 지정합니다.
D3D10_STENCIL_OP 깊이 스텐실 테스트 중에 수행할 수 있는 스텐실 작업입니다. (D3D10_STENCIL_OP)
D3D10_TEXTURE_ADDRESS_MODE 텍스처 경계 밖에 있는 텍스처 좌표를 확인하는 기술을 식별합니다. (D3D10_TEXTURE_ADDRESS_MODE)
D3D10_TEXTURECUBE_FACE 큐브 텍스처의 다른 얼굴입니다. (D3D10_TEXTURECUBE_FACE)
D3D10_USAGE

렌더링하는 동안 예상되는 리소스 사용을 식별합니다. 사용량은 CPU 및/또는 GPU에서 리소스에 액세스할 수 있는지 여부를 직접 반영합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d3d10_1.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct3D 10 그래픽에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 10 그래픽

d3d10_1.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID3D10BlendState1 이 혼합 상태 인터페이스는 출력 병합기 단계에 대한 Direct3D 10.1 디바이스의 혼합 상태에 액세스합니다.
ID3D10Device1 디바이스 인터페이스는 Direct3D 10.1에 대한 가상 어댑터를 나타냅니다. 렌더링을 수행하고 Direct3D 리소스를 만드는 데 사용됩니다.
ID3D10ShaderResourceView1 셰이더-resource-view 인터페이스는 렌더링 중에 셰이더가 액세스할 수 있는 하위 리소스를 지정합니다. 셰이더 리소스의 예로는 상수 버퍼, 텍스처 버퍼, 텍스처 또는 샘플러가 있습니다. (ID3D10ShaderResourceView1)

Functions

D3D10CreateDevice1 디스플레이 어댑터를 나타내는 Direct3D 10.1 디바이스를 만듭니다.
D3D10CreateDeviceAndSwapChain1 Direct3D 10.1 디바이스 및 스왑 체인을 만듭니다.

구조체

D3D10_BLEND_DESC1 Direct3D 10.1 디바이스의 혼합 상태를 설명합니다.
D3D10_RENDER_TARGET_BLEND_DESC1 Direct3D 10.1 디바이스의 렌더링 대상에 대한 혼합 상태를 설명합니다.
D3D10_SHADER_RESOURCE_VIEW_DESC1 셰이더 리소스 뷰에 대해 설명합니다. (D3D10_SHADER_RESOURCE_VIEW_DESC1)
D3D10_TEXCUBE_ARRAY_SRV1 셰이더 리소스 뷰에서 사용할 큐브 텍스처 배열의 하위 리소스를 지정합니다.

열거형

D3D10_FEATURE_LEVEL1 요청된 하드웨어 가속 버전입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d3d10_1shader.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct3D 10 그래픽에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 10 그래픽

d3d10_1shader.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID3D10ShaderReflection1 셰이더 리플렉션 인터페이스는 셰이더 정보에 액세스합니다. (ID3D10ShaderReflection1)

구조체

D3D10_SHADER_DEBUG_FILE_INFO 셰이더에 포함된 파일에 대해 설명합니다.
D3D10_SHADER_DEBUG_INFO D3D10GetShaderDebugInfo에서 반환된 ID3D10Blob 인터페이스의 형식을 설명합니다.
D3D10_SHADER_DEBUG_INPUT_INFO 셰이더 입력에 대해 설명합니다.
D3D10_SHADER_DEBUG_INST_INFO 명령 데이터를 포함합니다.
D3D10_SHADER_DEBUG_OUTPUTREG_INFO 셰이더 출력 레지스터에 대해 설명합니다.
D3D10_SHADER_DEBUG_OUTPUTVAR 셰이더 출력 변수에 대해 설명합니다.

D3D10_SHADER_DEBUG_SCOPE_INFO 변수 이름을 디버그 변수에 매핑하는 scope 데이터를 포함합니다.
D3D10_SHADER_DEBUG_SCOPEVAR_INFO 셰이더 scope 변수에 대해 설명합니다.
D3D10_SHADER_DEBUG_TOKEN_INFO 셰이더 요소의 원본 위치를 제공합니다.
D3D10_SHADER_DEBUG_VAR_INFO 셰이더 소스 변수에 대한 정보를 나타냅니다.

열거형

D3D10_SHADER_DEBUG_REGTYPE 셰이더 레지스터 형식입니다.
D3D10_SHADER_DEBUG_SCOPETYPE 범위 유형입니다.
D3D10_SHADER_DEBUG_VARTYPE scope 함수와 변수를 구분합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d3d10effect.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct3D 10 그래픽에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 10 그래픽

d3d10effect.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID3D10Effect ID3D10Effect 인터페이스는 렌더링 효과를 구현하기 위한 상태 개체, 리소스 및 셰이더 집합을 관리합니다.
ID3D10EffectBlendVariable blend-variable 인터페이스는 혼합 상태에 액세스합니다.
ID3D10EffectConstantBuffer 상수 버퍼 인터페이스는 상수 버퍼 또는 텍스처 버퍼에 액세스합니다.
ID3D10EffectDepthStencilVariable 깊이 스텐실 변수 인터페이스는 깊이 스텐실 상태에 액세스합니다.
ID3D10EffectDepthStencilViewVariable depth-stencil-view-variable 인터페이스는 깊이 스텐실 뷰에 액세스합니다.
ID3D10EffectMatrixVariable 행렬 변수 인터페이스는 행렬에 액세스합니다.
ID3D10EffectPass 통과 인터페이스는 기술 내에서 상태 할당을 캡슐화합니다.
ID3D10EffectPool 풀 인터페이스는 효과 간에 변수를 공유하기 위한 공통 메모리 공간(또는 풀)을 나타냅니다.
ID3D10EffectRasterizerVariable

래스터라이저 변수 인터페이스는 래스터라이저 상태에 액세스합니다.
ID3D10EffectRenderTargetViewVariable render-target-view 인터페이스는 렌더링 대상에 액세스합니다.
ID3D10EffectSamplerVariable 샘플러 인터페이스는 샘플러 상태에 액세스합니다.
ID3D10EffectScalarVariable 효과 스칼라 변수 인터페이스는 스칼라 값에 액세스합니다.
ID3D10EffectShaderResourceVariable 셰이더 리소스 인터페이스는 셰이더 리소스에 액세스합니다.
ID3D10EffectShaderVariable 셰이더 변수 인터페이스는 셰이더 변수에 액세스합니다.
ID3D10EffectStringVariable 문자열 변수 인터페이스는 문자열 변수에 액세스합니다.
ID3D10EffectTechnique ID3D10EffectTechnique 인터페이스는 패스의 컬렉션입니다.
ID3D10EffectType ID3D10EffectType 인터페이스는 형식별로 효과 변수에 액세스합니다.
ID3D10EffectVariable ID3D10EffectVariable 인터페이스는 모든 효과 변수의 기본 클래스입니다.
ID3D10EffectVectorVariable 벡터 변수 인터페이스는 4개 구성 요소 벡터에 액세스합니다.
ID3D10StateBlock 상태 블록 인터페이스는 렌더링 상태를 캡슐화합니다.

Functions

D3D10CompileEffectFromMemory 효과를 컴파일합니다.
D3D10CreateEffectFromMemory 컴파일된 효과를 포함하는 버퍼에서 ID3D10Effect를 만듭니다.
D3D10CreateEffectPoolFromMemory 효과 풀(또는 공유 메모리 위치)을 만들어 효과 간에 변수를 공유할 수 있도록 합니다.
D3D10CreateStateBlock 상태 블록을 만듭니다.
D3D10DisassembleEffect 컴파일된 효과를 어셈블리 명령 및 등록 할당이 포함된 텍스트 문자열로 디스어셈블하는 이 함수는 더 이상 사용되지 않습니다. 대신 D3DDisassemble10Effect를 사용합니다.
D3D10StateBlockMaskDifference 두 개의 상태 블록 마스크를 비트 XOR과 결합합니다.
D3D10StateBlockMaskDisableAll 상태 블록 마스크를 사용하여 모든 상태 캡처를 사용하지 않도록 설정합니다.
D3D10StateBlockMaskDisableCapture 상태 블록 마스크를 사용하여 상태 캡처를 사용하지 않도록 설정합니다.
D3D10StateBlockMaskEnableAll 상태 블록 마스크를 사용하도록 설정하여 모든 상태 변수를 캡처하고 적용합니다.
D3D10StateBlockMaskEnableCapture 상태 블록 마스크에서 상태 값 범위를 사용하도록 설정합니다.
D3D10StateBlockMaskGetSetting 상태 블록 마스크의 요소를 가져옵니다. 는 요소를 캡처하고 적용하기 위해 마스크에서 허용되는지 여부를 결정합니다.
D3D10StateBlockMaskIntersect 두 상태 블록 마스크를 비트 AND와 결합합니다.

D3D10StateBlockMaskUnion 두 상태 블록 마스크를 비트 OR과 결합합니다.

구조체

D3D10_EFFECT_DESC 효과를 설명합니다.
D3D10_EFFECT_SHADER_DESC 효과 셰이더에 대해 설명합니다.
D3D10_EFFECT_TYPE_DESC 효과 변수 형식에 대해 설명합니다.
D3D10_EFFECT_VARIABLE_DESC 효과 변수를 설명합니다.
D3D10_PASS_DESC 파이프라인 상태를 포함하는 효과 패스에 대해 설명합니다.
D3D10_PASS_SHADER_DESC 셰이더를 포함하는 효과 변수에 대해 설명합니다.
D3D10_STATE_BLOCK_MASK 디바이스 상태를 나타냅니다.
D3D10_TECHNIQUE_DESC 효과 기술을 설명합니다.

열거형

D3D10_DEVICE_STATE_TYPES 효과 디바이스 상태 유형입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d3d10misc.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct3D 10 그래픽에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 10 그래픽

d3d10misc.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

D3D10CreateBlob 버퍼를 만듭니다. 참고 이 함수를 사용하는 대신 D3DCreateBlob API를 사용하는 것이 좋습니다.
D3D10CreateDevice 디스플레이 어댑터를 나타내는 Direct3D 10.0 디바이스를 만듭니다.
D3D10CreateDeviceAndSwapChain Direct3D 10.0 디바이스 및 스왑 체인을 만듭니다.

열거형

D3D10_DRIVER_TYPE 디바이스 드라이버 유형입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d3d10sdklayers.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct3D 10 그래픽에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 10 그래픽

d3d10sdklayers.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID3D10Debug 디버그 인터페이스는 디버그 설정을 제어하고 파이프라인 상태의 유효성을 검사하며 디버그 계층이 켜져 있는 경우에만 사용할 수 있습니다. (ID3D10Debug)
ID3D10InfoQueue 정보 큐 인터페이스는 디버그 메시지를 저장, 검색 및 필터링합니다. 큐는 메시지 큐, 선택적 스토리지 필터 스택 및 선택적 검색 필터 스택으로 구성됩니다. (ID3D10InfoQueue)
ID3D10SwitchToRef 참조로 전환 인터페이스(참조로 전환 계층 참조)를 사용하면 애플리케이션이 하드웨어와 소프트웨어 디바이스 간에 전환할 수 있습니다.

구조체

D3D10_INFO_QUEUE_FILTER 디버그 메시지 필터; 에는 허용하거나 거부할 메시지 유형의 목록이 포함되어 있습니다. (D3D10_INFO_QUEUE_FILTER)
D3D10_INFO_QUEUE_FILTER_DESC 특정 유형의 메시지가 필터를 통과하도록 허용하거나 거부합니다. (D3D10_INFO_QUEUE_FILTER_DESC)
D3D10_MESSAGE 정보 큐의 디버그 메시지입니다. (D3D10_MESSAGE)

열거형

D3D10_MESSAGE_CATEGORY 디버그 메시지의 범주입니다. (D3D10_MESSAGE_CATEGORY)
D3D10_MESSAGE_ID 정보 큐 필터를 설정하기 위한 메시지 디버그(D3D10_INFO_QUEUE_FILTER 참조) 이러한 메시지를 사용하여 스토리지 및 검색 필터를 통과하도록 메시지 범주를 허용하거나 거부합니다.
D3D10_MESSAGE_SEVERITY 정보 큐에 대한 메시지 심각도 수준을 디버그합니다. (D3D10_MESSAGE_SEVERITY)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d3d10shader.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct3D 10 그래픽에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 10 그래픽

d3d10shader.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID3D10ShaderReflection 셰이더 리플렉션 인터페이스는 셰이더 정보에 액세스합니다. (ID3D10ShaderReflection)
ID3D10ShaderReflectionConstantBuffer 이 셰이더 리플렉션 인터페이스는 상수 버퍼에 대한 액세스를 제공합니다. (ID3D10ShaderReflectionConstantBuffer)
ID3D10ShaderReflectionType 이 셰이더 리플렉션 인터페이스는 변수 형식에 대한 액세스를 제공합니다. (ID3D10ShaderReflectionType)
ID3D10ShaderReflectionVariable 이 셰이더 리플렉션 인터페이스는 변수에 대한 액세스를 제공합니다. (ID3D10ShaderReflectionVariable)

Functions

D3D10CompileShader HLSL 셰이더를 컴파일합니다.
D3D10DisassembleShader 컴파일된 셰이더를 어셈블리 지침 및 등록 할당이 포함된 텍스트 문자열로 디스어셈블하는 이 함수는 더 이상 사용되지 않습니다. 대신 D3DDisassemble을 사용합니다.

D3D10GetGeometryShaderProfile 지정된 디바이스에 가장 적합한 기하 도형 셰이더 프로필을 가져옵니다.
D3D10GetInputAndOutputSignatureBlob 셰이더 서명이 포함된 버퍼를 가져옵니다.
D3D10GetInputSignatureBlob 셰이더 입력 서명이 포함된 버퍼를 가져옵니다.
D3D10GetOutputSignatureBlob 셰이더 출력 서명이 포함된 버퍼를 가져옵니다.
D3D10GetPixelShaderProfile 지정된 디바이스에 가장 적합한 픽셀 셰이더 프로필을 가져옵니다.
D3D10GetShaderDebugInfo 셰이더 디버그 정보를 가져옵니다. 디버그 정보는 D3D10CompileShader에서 생성되며 셰이더 본문에 포함됩니다.
D3D10GetVertexShaderProfile 지정된 디바이스에 가장 적합한 꼭짓점 셰이더 프로필을 가져옵니다.
D3D10PreprocessShader 컴파일된 셰이더에서 찾을 셰이더 토큰이 포함된 셰이더 텍스트 문자열을 생성합니다.
D3D10ReflectShader 컴파일된 셰이더에 대한 정보를 검색하기 위해 셰이더 리플렉션 개체를 만드는 이 함수는 더 이상 사용되지 않습니다. 대신 D3DReflect를 사용합니다.

구조체

D3D10_SHADER_BUFFER_DESC 셰이더 상수 버퍼에 대해 설명합니다. (D3D10_SHADER_BUFFER_DESC)

D3D10_SHADER_DESC 셰이더에 대해 설명합니다. (D3D10_SHADER_DESC)
D3D10_SHADER_INPUT_BIND_DESC 셰이더 리소스가 셰이더 입력에 바인딩되는 방법을 설명합니다. (D3D10_SHADER_INPUT_BIND_DESC)
D3D10_SHADER_TYPE_DESC 셰이더 변수 형식에 대해 설명합니다. (D3D10_SHADER_TYPE_DESC)
D3D10_SHADER_VARIABLE_DESC 셰이더 변수에 대해 설명합니다. (D3D10_SHADER_VARIABLE_DESC)
D3D10_SIGNATURE_PARAMETER_DESC 셰이더 서명을 설명합니다. (D3D10_SIGNATURE_PARAMETER_DESC)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d3d11.h 헤더

아티클2023. 08. 22.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 11 그래픽
- Microsoft Media Foundation

d3d11.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID3D11Asynchronous 이 인터페이스는 GPU에서 데이터를 비동기적으로 검색하는 메서드를 캡슐화합니다. (ID3D11Asynchronous)
ID3D11AuthenticatedChannel 그래픽 드라이버 또는 Microsoft Direct3D 런타임과 통신 채널을 제공합니다.
ID3D11BlendState 혼합 상태 인터페이스에는 출력 병합기 단계에 바인딩할 수 있는 혼합 상태에 대한 설명이 있습니다.
ID3D11Buffer 버퍼 인터페이스는 비정형 메모리인 버퍼 리소스에 액세스합니다. 버퍼는 일반적으로 꼭짓점 또는 인덱스 데이터를 저장합니다. (ID3D11Buffer)
ID3D11ClassInstance 이 인터페이스는 HLSL 클래스를 캡슐화합니다.
ID3D11ClassLinkage 이 인터페이스는 HLSL 동적 연결을 캡슐화합니다.
ID3D11CommandList ID3D11CommandList 인터페이스는 재생을 위한 그래픽 명령 목록을 캡슐화합니다.
ID3D11ComputeShader

컴퓨팅 셰이더 인터페이스는 컴퓨팅 셰이더 단계를 제어하는 실행 프로그램(컴퓨팅 셰이더)을 관리합니다.
ID3D11Counter 이 인터페이스는 GPU 성능을 측정하기 위한 메서드를 캡슐화합니다. (ID3D11Counter)
ID3D11CryptoSession 암호화 세션을 나타냅니다.
ID3D11DepthStencilState 깊이 스텐실 상태 인터페이스에는 출력 병합기에 바인딩할 수 있는 깊이 스텐실 상태에 대한 설명이 있습니다.
ID3D11DepthStencilView 깊이 스텐실 보기 인터페이스는 깊이 스텐실 테스트 중에 텍스처 리소스에 액세스합니다. (ID3D11DepthStencilView)
ID3D11Device 디바이스 인터페이스는 가상 어댑터를 나타냅니다. 리소스를 만드는 데 사용됩니다.
ID3D11DeviceChild 디바이스-자식 인터페이스는 디바이스에서 사용하는 데이터에 액세스합니다. (ID3D11DeviceChild)
ID3D11DeviceContext ID3D11DeviceContext 인터페이스는 렌더링 명령을 생성하는 디바이스 컨텍스트를 나타냅니다.
ID3D11DomainShader 도메인 셰이더 인터페이스는 도메인 셰이더 단계를 제어하는 실행 프로그램(도메인 셰이더)을 관리합니다.
ID3D11GeometryShader geometry-shader 인터페이스는 geometry 셰이더 단계를 제어하는 실행 프로그램(기하 도형 셰이더)을 관리합니다. (ID3D11GeometryShader)
ID3D11HullShader 헐 셰이더 인터페이스는 헐 셰이더 단계를 제어하는 실행 프로그램(헐 셰이더)을 관리합니다.
ID3D11InputLayout

입력 레이아웃 인터페이스에는 메모리에 배치된 꼭짓점 데이터를 그래픽 파이프라인의 입력 어셈블러 단계에 공급하는 방법에 대한 정의가 있습니다.
ID3D11PixelShader 픽셀 셰이더 인터페이스는 픽셀 셰이더 단계를 제어하는 실행 프로그램(픽셀 셰이더)을 관리합니다. (ID3D11PixelShader)
ID3D11Predicate 조건자 인터페이스는 이전 그리기 호출의 결과에 따라 기하 도형을 처리해야 하는지 여부를 결정합니다. (ID3D11Predicate)
ID3D11Query 쿼리 인터페이스는 GPU의 정보를 쿼리합니다. (ID3D11Query)
ID3D11RasterizerState 래스터라이저 상태 인터페이스에는 래스터라이저 단계에 바인딩할 수 있는 래스터라이저 상태에 대한 설명이 있습니다.
ID3D11RenderTargetView render-target-view 인터페이스는 렌더링 중에 액세스할 수 있는 렌더링 대상 하위 리소스를 식별합니다. (ID3D11RenderTargetView)
ID3D11Resource 리소스 인터페이스는 모든 리소스에 대한 일반적인 작업을 제공합니다. (ID3D11Resource)
ID3D11SamplerState 샘플러 상태 인터페이스에는 텍스처 샘플 작업에서 참조하기 위해 파이프라인의 셰이더 단계에 바인딩할 수 있는 샘플러 상태에 대한 설명이 있습니다.
ID3D11ShaderResourceView 셰이더-resource-view 인터페이스는 렌더링 중에 셰이더가 액세스할 수 있는 하위 리소스를 지정합니다. 셰이더 리소스의 예로는 상수 버퍼, 텍스처 버퍼 및 텍스처가 있습니다.
ID3D11Texture1D 1D 텍스처 인터페이스는 구조화된 메모리인 텍셀 데이터에 액세스합니다. (ID3D11Texture1D)
ID3D11Texture2D 2D 텍스처 인터페이스는 구조화된 메모리인 텍셀 데이터를 관리합니다. (ID3D11Texture2D)

ID3D11Texture3D 3D 텍스처 인터페이스는 구조화된 메모리인 텍셀 데이터에 액세스합니다. (ID3D11Texture3D)
ID3D11UnorderedAccessView 뷰 인터페이스는 렌더링 중에 파이프라인이 액세스할 수 있는 리소스 부분을 지정합니다. (ID3D11UnorderedAccessView)
ID3D11VertexShader 꼭짓점 셰이더 인터페이스는 꼭짓점 셰이더 단계를 제어하는 실행 프로그램(꼭짓점 셰이더)을 관리합니다. (ID3D11VertexShader)
ID3D11VideoContext Microsoft Direct3D 11 디바이스의 비디오 기능을 제공합니다. (ID3D11VideoContext)
ID3D11VideoDecoder Microsoft Direct3D 11용 하드웨어 가속 비디오 디코더를 나타냅니다.
ID3D11VideoDecoderOutputView 비디오 디코딩 중에 액세스할 수 있는 출력 화면을 식별합니다.
ID3D11VideoDevice Microsoft Direct3D 11 디바이스의 비디오 디코딩 및 비디오 처리 기능을 제공합니다. (ID3D11VideoDevice)
ID3D11VideoProcessor Microsoft Direct3D 11용 비디오 프로세서를 나타냅니다.
ID3D11VideoProcessorEnumerator Microsoft Direct3D 11 디바이스의 비디오 프로세서 기능을 열거합니다. (ID3D11VideoProcessorEnumerator)
ID3D11VideoProcessorInputView 비디오 처리 중에 액세스할 수 있는 입력 화면을 식별합니다.
ID3D11VideoProcessorOutputView 비디오 처리 중에 액세스할 수 있는 출력 표면을 식별합니다.

ID3D11View 뷰 인터페이스는 렌더링 중에 파이프라인이 액세스할 수 있는 리소스 부분을 지정합니다. (ID3D11View)

Functions

D3D11CalcSubresource 텍스처에 대한 하위 리소스 인덱스를 계산합니다.
D3D11CreateDevice 디스플레이 어댑터를 나타내는 디바이스를 만듭니다. (D3D11CreateDevice)
D3D11CreateDeviceAndSwapChain 디스플레이 어댑터와 렌더링에 사용되는 스왑 체인을 나타내는 디바이스를 만듭니다.

구조체

CD3D11_BLEND_DESC 혼합 상태 구조를 나타내며 혼합 상태 구조를 만들기 위한 편리한 메서드를 제공합니다.
CD3D11_BOX 상자를 나타내고 상자를 만들기 위한 편리한 메서드를 제공합니다.
CD3D11_BUFFER_DESC 버퍼를 나타내며 버퍼를 만들기 위한 편리한 메서드를 제공합니다.
CD3D11_COUNTER_DESC 카운터를 나타내며 카운터를 만들기 위한 편리한 메서드를 제공합니다.
CD3D11_DEPTH_STENCIL_DESC

깊이 스텐실 상태 구조를 나타내며 깊이 스텐실 상태 구조를 만들기 위한 편리한 메서드를 제공합니다.
CD3D11_DEPTH_STENCIL_VIEW_DESC 깊이 스텐실 보기를 나타내며 깊이 스텐실 뷰를 만들기 위한 편리한 방법을 제공합니다.
CD3D11_QUERY_DESC 쿼리를 나타내며 쿼리를 만들기 위한 편리한 메서드를 제공합니다.
CD3D11_RASTERIZER_DESC 래스터라이저 상태 구조를 나타내며 래스터라이저 상태 구조를 만들기 위한 편리한 메서드를 제공합니다.
CD3D11_RECT 사각형을 나타내며 사각형을 만들기 위한 편리한 메서드를 제공합니다.
CD3D11_RENDER_TARGET_VIEW_DESC 렌더링 대상 뷰를 나타내며 렌더링 대상 뷰를 만들기 위한 편리한 메서드를 제공합니다.
CD3D11_SAMPLER_DESC 샘플러 상태를 나타내며 샘플러 상태를 만들기 위한 편리한 메서드를 제공합니다.
CD3D11_SHADER_RESOURCE_VIEW_DESC 셰이더 리소스 뷰를 나타내며 셰이더 리소스 뷰를 만들기 위한 편리한 메서드를 제공합니다.
CD3D11_TEXTURE1D_DESC 1D 텍스처를 나타내며 1D 텍스처를 만들기 위한 편리한 방법을 제공합니다.
CD3D11_TEXTURE2D_DESC 2D 텍스처를 나타내며 2D 텍스처를 만들기 위한 편리한 방법을 제공합니다.
CD3D11_TEXTURE3D_DESC 3D 텍스처를 나타내며 3D 텍스처를 만들기 위한 편리한 방법을 제공합니다.
CD3D11_UNORDERED_ACCESS_VIEW_DESC 순서가 지정되지 않은 액세스 보기를 나타내며 순서가 지정되지 않은 액세스 뷰를 만들기 위한 편리한 메서드를 제공합니다.

CD3D11_VIEWPORT 뷰포트를 나타내며 뷰포트를 만들기 위한 편리한 메서드를 제공합니다.
D3D11_AES_CTR_IV 128비트 AES-CTR(Advanced Encryption Standard CTR 모드) 블록 암호 암호화에 대한 IV(초기화 벡터)를 포함합니다. (D3D11_AES_CTR_IV)
D3D11_AUTHENTICATED_CONFIGURE_ACCESSIBLE_ENCRYPTION_INPUT D3D11_AUTHENTICATED_CONFIGURE_ENCRYPTION_WHEN_ACCESSIBLE 명령에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_CONFIGURE_CRYPTO_SESSION_INPUT D3D11_AUTHENTICATED_CONFIGURE_CRYPTO_SESSION 명령에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_CONFIGURE_INITIALIZE_INPUT D3D11_AUTHENTICATED_CONFIGURE_INITIALIZE 명령에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_CONFIGURE_INPUT ID3D11VideoContext::ConfigureAuthenticatedChannel 메서드에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_CONFIGURE_OUTPUT ID3D11VideoContext::ConfigureAuthenticatedChannel 메서드의 응답을 포함합니다.
D3D11_AUTHENTICATED_CONFIGURE_PROTECTION_INPUT D3D11_AUTHENTICATED_CONFIGURE_PROTECTION 명령에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_CONFIGURE_SHARED_RESOURCE_INPUT D3D11_AUTHENTICATED_CONFIGURE_SHARED_RESOURCE 명령에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_PROTECTION_FLAGS 비디오 콘텐츠에 대한 보호 수준을 지정합니다.
D3D11_AUTHENTICATED_QUERY_ACCESSIBILITY_ENCRYPTION_GUID_COUNT_OUTPUT D3D11_AUTHENTICATED_QUERY_ENCRYPTION_WHEN_ACCESSIBLE_GUID_COUNT 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_ACCESSIBILITY_ENCRYPTION_GUID_INPUT

D3D11_AUTHENTICATED_QUERY_ENCRYPTION_WHEN_ACCESSIBLE_GUID 쿼리에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_QUERY_ACCESSIBILITY_ENCRYPTION_GUID_OUTPUT D3D11_AUTHENTICATED_QUERY_ENCRYPTION_WHEN_ACCESSIBLE_GUID 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_ACCESSIBILITY_OUTPUT D3D11_AUTHENTICATED_QUERY_ACCESSIBILITY_ATTRIBUTES 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_CHANNEL_TYPE_OUTPUT D3D11_AUTHENTICATED_QUERY_CHANNEL_TYPE 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_CRYPTO_SESSION_INPUT D3D11_AUTHENTICATED_QUERY_CRYPTO_SESSION 쿼리에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_QUERY_CRYPTO_SESSION_OUTPUT D3D11_AUTHENTICATED_QUERY_CRYPTO_SESSION 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_CURRENT_ACCESSIBILITY_ENCRYPTION_OUTPUT D3D11_AUTHENTICATED_QUERY_CURRENT_ENCRYPTION_WHEN_ACCESSIBLE 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_DEVICE_HANDLE_OUTPUT D3D11_AUTHENTICATED_QUERY_DEVICE_HANDLE 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_INPUT ID3D11VideoContext::QueryAuthenticatedChannel 메서드에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_QUERY_OUTPUT ID3D11VideoContext::QueryAuthenticatedChannel 메서드의 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_OUTPUT_ID_COUNT_INPUT D3D11_AUTHENTICATED_QUERY_OUTPUT_ID_COUNT 쿼리에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_QUERY_OUTPUT_ID_COUNT_OUTPUT D3D11_AUTHENTICATED_QUERY_OUTPUT_ID_COUNT 쿼리에 대한 응답을 포함합니다.

D3D11_AUTHENTICATED_QUERY_OUTPUT_ID_INPUT D3D11_AUTHENTICATED_QUERY_OUTPUT_ID 쿼리에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_QUERY_OUTPUT_ID_OUTPUT D3D11_AUTHENTICATED_QUERY_OUTPUT_ID 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_PROTECTION_OUTPUT D3D11_AUTHENTICATED_QUERY_PROTECTION 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_RESTRICTED_SHARED_RESOURCE_PROCESS_COUNT_OUTPUT D3D11_AUTHENTICATED_QUERY_RESTRICTED_SHARED_RESOURCE_PROCESS_COUNT 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_RESTRICTED_SHARED_RESOURCE_PROCESS_INPUT D3D11_AUTHENTICATED_QUERY_RESTRICTED_SHARED_RESOURCE_PROCESS 쿼리에 대한 입력 데이터를 포함합니다.
D3D11_AUTHENTICATED_QUERY_RESTRICTED_SHARED_RESOURCE_PROCESS_OUTPUT D3D11_AUTHENTICATED_QUERY_RESTRICTED_SHARED_RESOURCE_PROCESS 쿼리에 대한 응답을 포함합니다.
D3D11_AUTHENTICATED_QUERY_UNRESTRICTED_PROTECTED_SHARED_RESOURCE_COUNT_OUTPUT D3D11_AUTHENTICATED_QUERY_UNRESTRICTED_PROTECTED_SHARED_RESOURCE_COUNT 쿼리에 대한 응답을 포함합니다.
D3D11_BLEND_DESC ID3D11Device::CreateBlendState를 호출하여 혼합 상태 개체를 만드는 데 사용하는 혼합 상태를 설명합니다.
D3D11_BOX 3D 상자를 정의합니다. (D3D11_BOX)
D3D11_BUFFER_DESC 버퍼 리소스에 대해 설명합니다. (D3D11_BUFFER_DESC)
D3D11_BUFFER_RTV 렌더링 대상 뷰에서 사용할 버퍼 리소스의 요소를 지정합니다.

D3D11_BUFFER_SRV 셰이더 리소스 뷰에서 사용할 버퍼 리소스의 요소를 지정합니다. (D3D11_BUFFER_SRV)
D3D11_BUFFER_UAV 순서가 지정되지 않은 액세스 뷰에서 사용할 버퍼의 요소에 대해 설명합니다. (D3D11_BUFFER_UAV)
D3D11_BUFFEREX_SRV 셰이더 리소스 뷰에서 사용할 원시 버퍼 리소스의 요소에 대해 설명합니다.
D3D11_CLASS_INSTANCE_DESC HLSL 클래스 instance 설명합니다.
D3D11_COUNTER_DESC 카운터에 대해 설명합니다. (D3D11_COUNTER_DESC)
D3D11_COUNTER_INFO 비디오 카드 성능 카운터 기능에 대한 정보입니다. (D3D11_COUNTER_INFO)
D3D11_DEPTH_STENCIL_DESC 깊이 스텐실 상태를 설명합니다. (D3D11_DEPTH_STENCIL_DESC)
D3D11_DEPTH_STENCIL_VIEW_DESC 깊이 스텐실 보기에서 액세스할 수 있는 텍스처의 하위 리소스를 지정합니다.
D3D11_DEPTH_STENCILOP_DESC 스텐실 테스트 결과에 따라 수행할 수 있는 스텐실 작업입니다.
D3D11_DRAW_INDEXED_INSTANCED_INDIRECT_ARGS 인덱싱된 인스턴스화된 간접 그리기 인수입니다.
D3D11_DRAW_INSTANCED_INDIRECT_ARGS 인스턴스화된 간접 그리기 인수입니다.
D3D11_ENCRYPTED_BLOCK_INFO 비디오 화면에서 암호화되는 바이트를 지정합니다.
D3D11_FEATURE_DATA_ARCHITECTURE_INFO

Direct3D 11.1 어댑터 아키텍처에 대한 정보를 설명합니다.
D3D11_FEATURE_DATA_D3D10_X_HARDWARE_OPTIONS 현재 그래픽 드라이버의 컴퓨팅 셰이더 및 원시 및 구조화된 버퍼 지원에 대해 설명합니다.
D3D11_FEATURE_DATA_D3D11_OPTIONS 현재 그래픽 드라이버의 Direct3D 11.1 기능 옵션에 대해 설명합니다.
D3D11_FEATURE_DATA_D3D11_OPTIONS1 현재 그래픽 드라이버의 Direct3D 11.2 기능 옵션에 대해 설명합니다.
D3D11_FEATURE_DATA_D3D11_OPTIONS2 현재 그래픽 드라이버의 Direct3D 11.3 기능 옵션에 대해 설명합니다. (D3D11_FEATURE_DATA_D3D11_OPTIONS2)
D3D11_FEATURE_DATA_D3D11_OPTIONS3 현재 그래픽 드라이버의 Direct3D 11.3 기능 옵션에 대해 설명합니다. (D3D11_FEATURE_DATA_D3D11_OPTIONS3)
D3D11_FEATURE_DATA_D3D11_OPTIONS5 현재 그래픽 드라이버에서 공유 리소스에 대한 지원 수준을 설명합니다.
D3D11_FEATURE_DATA_D3D9_OPTIONS 현재 그래픽 드라이버의 Direct3D 9 기능 옵션에 대해 설명합니다. (D3D11_FEATURE_DATA_D3D9_OPTIONS)
D3D11_FEATURE_DATA_D3D9_OPTIONS1 현재 그래픽 드라이버의 Direct3D 9 기능 옵션에 대해 설명합니다. (D3D11_FEATURE_DATA_D3D9_OPTIONS1)
D3D11_FEATURE_DATA_D3D9_SHADOW_SUPPORT 현재 그래픽 드라이버의 Direct3D 9 섀도 지원에 대해 설명합니다.
D3D11_FEATURE_DATA_D3D9_SIMPLE_INSTANCING_SUPPORT 간단한 인스턴스화가 지원되는지 여부를 설명합니다.
D3D11_FEATURE_DATA_DISPLAYABLE 현재 그래픽 드라이버에서 지원되는 표시 가능한 표면의 수준을 설명합니다.

D3D11_FEATURE_DATA_DOUBLES 현재 그래픽 드라이버의 이중 데이터 형식 지원에 대해 설명합니다.
D3D11_FEATURE_DATA_FORMAT_SUPPORT 지정된 형식에 대해 현재 그래픽 드라이버에서 지원하는 리소스에 대해 설명합니다. (D3D11_FEATURE_DATA_FORMAT_SUPPORT)
D3D11_FEATURE_DATA_FORMAT_SUPPORT2 지정된 형식에 대해 현재 그래픽 드라이버에서 지원하는 순서가 지정되지 않은 리소스 옵션을 설명합니다.
D3D11_FEATURE_DATA_GPU_VIRTUAL_ADDRESS_SUPPORT 리소스당 최대 주소 비트 및 프로세스당 최대 주소 비트를 포함하여 기능 데이터 GPU 가상 주소 지원에 대해 설명합니다.
D3D11_FEATURE_DATA_MARKER_SUPPORT GPU 프로파일링 기술이 지원되는지 여부를 설명합니다.
D3D11_FEATURE_DATA_SHADER_CACHE 현재 그래픽 드라이버에서 지원되는 셰이더 캐싱 수준을 설명합니다. (D3D11_FEATURE_DATA_SHADER_CACHE)
D3D11_FEATURE_DATA_SHADER_MIN_PRECISION_SUPPORT 현재 그래픽 드라이버의 셰이더에 대한 정밀도 지원 옵션을 설명합니다.
D3D11_FEATURE_DATA_THREADING 현재 그래픽 드라이버에서 지원하는 다중 스레딩 기능에 대해 설명합니다.
D3D11_INPUT_ELEMENT_DESC 입력 어셈블러 단계의 단일 요소에 대한 설명입니다. (D3D11_INPUT_ELEMENT_DESC)
D3D11_MAPPED_SUBRESOURCE 하위 리소스 데이터에 대한 액세스를 제공합니다.
D3D11_OMAC MAC(메시지 인증 코드)를 포함합니다.

D3D11_QUERY_DATA_PIPELINE_STATISTICS ID3D11DeviceContext::Begin 및 ID3D11DeviceContext::End 호출 간의 그래픽 파이프라인 활동에 대한 정보를 쿼리합니다.
D3D11_QUERY_DATA_SO_STATISTICS ID3D11DeviceContext::Begin 및 ID3D11DeviceContext::End 사이에 스트림 출력 버퍼로 스트리밍되는 데이터의 양에 대한 정보를 쿼리합니다.
D3D11_QUERY_DATA_TIMESTAMP_DISJOINT 타임스탬프 쿼리의 안정성에 대한 쿼리 정보입니다. (D3D11_QUERY_DATA_TIMESTAMP_DISJOINT)
D3D11_QUERY_DESC 쿼리를 설명합니다. (D3D11_QUERY_DESC)
D3D11_RASTERIZER_DESC 래스터라이저 상태를 설명합니다. (D3D11_RASTERIZER_DESC)
D3D11_RENDER_TARGET_BLEND_DESC 렌더링 대상의 혼합 상태를 설명합니다. (D3D11_RENDER_TARGET_BLEND_DESC)
D3D11_RENDER_TARGET_VIEW_DESC 렌더링 대상 뷰를 사용하여 액세스할 수 있는 리소스의 하위 리소스를 지정합니다.
D3D11_SAMPLER_DESC 샘플러 상태를 설명합니다. (D3D11_SAMPLER_DESC)
D3D11_SHADER_RESOURCE_VIEW_DESC 셰이더 리소스 뷰를 설명합니다. (D3D11_SHADER_RESOURCE_VIEW_DESC)
D3D11_SO_DECLARATION_ENTRY 출력 슬롯의 꼭짓점 버퍼에 있는 꼭짓점 요소에 대한 설명입니다. (D3D11_SO_DECLARATION_ENTRY)
D3D11_SUBRESOURCE_DATA 하위 리소스를 초기화하기 위한 데이터를 지정합니다. (D3D11_SUBRESOURCE_DATA)
D3D11_TEX1D_ARRAY_DSV 깊이 스텐실 보기에서 사용할 1D 텍스처 배열의 하위 리소스를 지정합니다.

D3D11_TEX1D_ARRAY_RTV 렌더링 대상 보기에서 사용할 1D 텍스처 배열의 하위 리소스를 지정합니다.
D3D11_TEX1D_ARRAY_SRV 셰이더 리소스 뷰에서 사용할 1D 텍스처 배열의 하위 리소스를 지정합니다.
D3D11_TEX1D_ARRAY_UAV 순서가 지정되지 않은 액세스 1D 텍스처 리소스의 배열에 대해 설명합니다. (D3D11_TEX1D_ARRAY_UAV)
D3D11_TEX1D_DSV 깊이 스텐실 보기에 액세스할 수 있는 1D 텍스처의 하위 리소스를 지정합니다. (D3D11_TEX1D_DSV)
D3D11_TEX1D_RTV 렌더링 대상 보기에서 사용할 1D 텍스처의 하위 리소스를 지정합니다. (D3D11_TEX1D_RTV)
D3D11_TEX1D_SRV 셰이더 리소스 뷰에서 사용할 1D 텍스처의 하위 리소스를 지정합니다. (D3D11_TEX1D_SRV)
D3D11_TEX1D_UAV 순서가 지정되지 않은 액세스 1D 텍스처 리소스에 대해 설명합니다. (D3D11_TEX1D_UAV)
D3D11_TEX2D_ARRAY_DSV 깊이 스텐실 보기에 액세스할 수 있는 배열 2D 텍스처의 하위 리소스를 지정합니다.
D3D11_TEX2D_ARRAY_RTV 렌더링 대상 보기에서 사용할 2D 텍스처 배열의 하위 리소스를 지정합니다.
D3D11_TEX2D_ARRAY_SRV 셰이더 리소스 뷰에서 사용할 2D 텍스처 배열의 하위 리소스를 지정합니다.
D3D11_TEX2D_ARRAY_UAV 순서가 지정되지 않은 액세스 2D 텍스처 리소스의 배열에 대해 설명합니다. (D3D11_TEX2D_ARRAY_UAV)
D3D11_TEX2D_ARRAY_VPOV 비디오 프로세서 출력 보기에 대한 텍스처 리소스를 식별합니다. (D3D11_TEX2D_ARRAY_VPOV)

D3D11_TEX2D_DSV 깊이 스텐실 보기에 액세스할 수 있는 2D 텍스처의 하위 리소스를 지정합니다. (D3D11_TEX2D_DSV)
D3D11_TEX2D_RTV 렌더링 대상 보기에서 사용할 2D 텍스처의 하위 리소스를 지정합니다. (D3D11_TEX2D_RTV)
D3D11_TEX2D_SRV 셰이더 리소스 뷰에서 사용할 2D 텍스처의 하위 리소스를 지정합니다. (D3D11_TEX2D_SRV)
D3D11_TEX2D_UAV 순서가 지정되지 않은 액세스 2D 텍스처 리소스에 대해 설명합니다. (D3D11_TEX2D_UAV)
D3D11_TEX2D_VDOV 비디오 디코더 출력 뷰의 텍스처 리소스를 식별합니다.
D3D11_TEX2D_VPIV 비디오 프로세서 입력 보기의 텍스처 리소스를 식별합니다.
D3D11_TEX2D_VPOV 비디오 프로세서 출력 보기에 대한 텍스처 리소스를 식별합니다. (D3D11_TEX2D_VPOV)
D3D11_TEX2DMS_ARRAY_DSV 깊이 스텐실 보기에 대해 다중 샘플링된 2D 텍스처 배열의 하위 리소스를 지정합니다.
D3D11_TEX2DMS_ARRAY_RTV 렌더링 대상 보기에서 사용할 다중 샘플링된 2D 텍스처 배열의 하위 리소스를 지정합니다.
D3D11_TEX2DMS_ARRAY_SRV 셰이더 리소스 뷰에서 사용할 다중 샘플링된 2D 텍스처 배열의 하위 리소스를 지정합니다.
D3D11_TEX2DMS_DSV 깊이 스텐실 보기에 액세스할 수 있는 다중 샘플링된 2D 텍스처의 하위 리소스를 지정합니다. (D3D11_TEX2DMS_DSV)
D3D11_TEX2DMS_RTV 렌더링 대상 보기에서 사용할 다중 샘플링된 2D 텍스처의 하위 리소스를 지정합니다. (D3D11_TEX2DMS_RTV)

D3D11_TEX2DMS_SRV 셰이더 리소스 뷰에서 사용할 다중 샘플링된 2D 텍스처의 하위 리소스를 지정합니다.
D3D11_TEX3D_RTV 렌더링 대상 보기에서 사용할 3D 텍스처의 하위 리소스를 지정합니다.
D3D11_TEX3D_SRV 셰이더 리소스 뷰에서 사용할 3D 텍스처의 하위 리소스를 지정합니다.
D3D11_TEX3D_UAV 순서가 지정되지 않은 액세스 3D 텍스처 리소스에 대해 설명합니다. (D3D11_TEX3D_UAV)
D3D11_TEXCUBE_ARRAY_SRV 셰이더 리소스 뷰에서 사용할 큐브 텍스처 배열의 하위 리소스를 지정합니다.
D3D11_TEXCUBE_SRV 셰이더 리소스 뷰에서 사용할 큐브 텍스처의 하위 리소스를 지정합니다. (D3D11_TEXCUBE_SRV)
D3D11_TEXTURE1D_DESC 1D 텍스처에 대해 설명합니다. (D3D11_TEXTURE1D_DESC)
D3D11_TEXTURE2D_DESC 2D 텍스처를 설명합니다. (D3D11_TEXTURE2D_DESC)
D3D11_TEXTURE3D_DESC 3D 텍스처를 설명합니다. (D3D11_TEXTURE3D_DESC)
D3D11_UNORDERED_ACCESS_VIEW_DESC 순서가 지정되지 않은 액세스 보기를 사용하여 액세스할 수 있는 리소스의 하위 리소스를 지정합니다.
D3D11_VIDEO_COLOR Microsoft Direct3D 11 비디오의 색 값을 정의합니다.
D3D11_VIDEO_COLOR_RGBA RGB 색 값을 지정합니다. (D3D11_VIDEO_COLOR_RGBA)

D3D11_VIDEO_COLOR_YCbCrA YCbCr 색 값을 지정합니다. (D3D11_VIDEO_COLOR_YCbCrA)
D3D11_VIDEO_CONTENT_PROTECTION_CAPS 그래픽 드라이버의 콘텐츠 보호 기능을 설명합니다.
D3D11_VIDEO_DECODER_BUFFER_DESC 디코딩을 위해 압축된 버퍼에 대해 설명합니다. (D3D11_VIDEO_DECODER_BUFFER_DESC)
D3D11_VIDEO_DECODER_CONFIG DXVA(DirectX Video Acceleration)용 Microsoft Direct3D 11 디코더 디바이스의 구성에 대해 설명합니다.
D3D11_VIDEO_DECODER_DESC Microsoft Direct3D 11 비디오 디코더 또는 비디오 프로세서의 비디오 스트림에 대해 설명합니다.
D3D11_VIDEO_DECODER_EXTENSION ID3D11VideoContext::D ecoderExtension 메서드에 대한 드라이버 관련 데이터를 포함합니다.
D3D11_VIDEO_DECODER_OUTPUT_VIEW_DESC 비디오 디코더 출력 보기를 설명합니다.
D3D11_VIDEO_PROCESSOR_CAPS Microsoft Direct3D 11 비디오 프로세서의 기능에 대해 설명합니다.
D3D11_VIDEO_PROCESSOR_COLOR_SPACE 비디오 처리를 위한 색 공간을 지정합니다.
D3D11_VIDEO_PROCESSOR_CONTENT_DESC 비디오 프로세서의 비디오 스트림에 대해 설명합니다.
D3D11_VIDEO_PROCESSOR_CUSTOM_RATE 프레임 속도 변환 또는 IVTC(역 텔레신)에 대한 사용자 지정 속도를 지정합니다. (D3D11_VIDEO_PROCESSOR_CUSTOM_RATE)
D3D11_VIDEO_PROCESSOR_FILTER_RANGE 이미지 필터에 대해 지원되는 값의 범위를 정의합니다. (D3D11_VIDEO_PROCESSOR_FILTER_RANGE)

D3D11_VIDEO_PROCESSOR_INPUT_VIEW_DESC 비디오 프로세서 입력 보기를 설명합니다.
D3D11_VIDEO_PROCESSOR_OUTPUT_VIEW_DESC 비디오 프로세서 출력 보기를 설명합니다.
D3D11_VIDEO_PROCESSOR_RATE_CONVERSION_CAPS 디인터레이싱 및 역 텔레신을 포함하여 프레임 속도 변환과 관련된 비디오 프로세서 기능 그룹을 정의합니다.
D3D11_VIDEO_PROCESSOR_STREAM ID3D11VideoContext::VideoProcessorBlt 메서드에 대한 스트림 수준 데이터를 포함합니다.
D3D11_VIEWPORT 뷰포트의 차원을 정의합니다. (D3D11_VIEWPORT)

열거형

D3D11_ASYNC_GETDATA_FLAG ID3D11DeviceContext::GetData의 동작을 제어하는 선택적 플래그입니다.
D3D11_AUTHENTICATED_CHANNEL_TYPE Microsoft Direct3D 인증 채널의 유형을 지정합니다.
D3D11_AUTHENTICATED_PROCESS_IDENTIFIER_TYPE D3D11_AUTHENTICATED_QUERY_RESTRICTED_SHARED_RESOURCE_PROCESS_OUTPUT 구조에서 식별되는 프로세스 유형을 지정합니다.
D3D11_BIND_FLAG 파이프라인에 리소스를 바인딩하는 방법을 식별합니다. (D3D11_BIND_FLAG)
D3D11_BLEND 픽셀 셰이더 및 렌더링 대상에 대한 값을 변조하는 혼합 요소입니다.

D3D11_BLEND_OP RGB 또는 알파 혼합 작업. (D3D11_BLEND_OP)
D3D11_BUFFER_UAV_FLAG 버퍼 리소스에 대한 순서가 지정되지 않은 액세스 보기 옵션을 식별합니다. (D3D11_BUFFER_UAV_FLAG)
D3D11_BUFFEREX_SRV_FLAG 버퍼 리소스를 보는 방법을 식별합니다. (D3D11_BUFFEREX_SRV_FLAG)
D3D11_BUS_TYPE 그래픽 어댑터에서 사용되는 I/O 버스의 유형을 지정합니다.
D3D11_CLEAR_FLAG 지울 깊이 스텐실의 부분을 지정합니다.
D3D11_COLOR_WRITE_ENABLE 혼합하는 동안 쓰기 가능한 렌더링 대상의 각 픽셀 구성 요소를 식별합니다. (D3D11_COLOR_WRITE_ENABLE)
D3D11_COMPARISON_FUNC 비교 옵션. (D3D11_COMPARISON_FUNC)
D3D11_CONSERVATIVE_RASTERIZATION_TIER 하드웨어 및 드라이버가 보수적인 래스터화를 지원하는지와 계층 수준에서 지원하는지 지정합니다.
D3D11_CONTENT_PROTECTION_CAPS 콘텐츠 보호 기능을 설명하는 플래그를 포함합니다.
D3D11_COUNTER 성능 카운터에 대한 옵션입니다.
D3D11_COUNTER_TYPE 성능 카운터의 데이터 형식입니다. (D3D11_COUNTER_TYPE)
D3D11_CPU_ACCESS_FLAG

리소스에 허용되는 CPU 액세스 유형을 지정합니다. (D3D11_CPU_ACCESS_FLAG)
D3D11_CREATE_DEVICE_FLAG 디바이스를 만드는 데 사용되는 매개 변수에 대해 설명합니다.
D3D11_CULL_MODE 특정 방향을 향한 삼각형이 그려지지 않음을 나타냅니다. (D3D11_CULL_MODE)
D3D11_DEPTH_WRITE_MASK 깊이 데이터를 작성하기 위한 깊이 스텐실 버퍼의 부분을 식별합니다. (D3D11_DEPTH_WRITE_MASK)
D3D11_DEVICE_CONTEXT_TYPE 디바이스 컨텍스트 옵션.
D3D11_DSV_DIMENSION 깊이 스텐실 보기에서 사용되는 리소스에 액세스하는 방법을 지정합니다. (D3D11_DSV_DIMENSION)
D3D11_DSV_FLAG 깊이 스텐실 보기 옵션입니다.
D3D11_FEATURE Direct3D 11 기능 옵션.
D3D11_FILL_MODE 삼각형을 렌더링할 때 사용할 채우기 모드를 결정합니다. (D3D11_FILL_MODE)
D3D11_FILTER 텍스처 샘플링 중 필터링 옵션입니다. (D3D11_FILTER)
D3D11_FILTER_REDUCTION_TYPE 샘플러 필터 감소 유형을 지정합니다.
D3D11_FILTER_TYPE 배율 또는 축소 샘플러 필터의 유형입니다. (D3D11_FILTER_TYPE)
D3D11_FORMAT_SUPPORT

지정된 형식 및 지정된 디바이스에 대해 지원되는 리소스(ID3D11Device::CheckFormatSupport 및 ID3D11Device::CheckFeatureSupport 참조).
D3D11_FORMAT_SUPPORT2 컴퓨팅 셰이더 리소스에 대한 순서가 지정되지 않은 리소스 지원 옵션입니다 (ID3D11Device::CheckFeatureSupport 참조).
D3D11_INPUT_CLASSIFICATION 입력 슬롯에 포함된 데이터 형식입니다. (D3D11_INPUT_CLASSIFICATION)
D3D11_MAP CPU에서 읽고 쓰기 위해 액세스할 리소스를 식별합니다. 애플리케이션은 이러한 플래그 중 하나 이상을 결합할 수 있습니다. (D3D11_MAP)
D3D11_MAP_FLAG 애플리케이션이 GPU에서 사용 중인 리소스에서 ID3D11DeviceContext::Map 메서드를 호출할 때 CPU가 응답하는 방법을 지정합니다.
D3D11_QUERY 쿼리 형식입니다. (D3D11_QUERY)
D3D11_QUERY_MISC_FLAG 기타 쿼리 동작을 설명하는 플래그입니다. (D3D11_QUERY_MISC_FLAG)
D3D11_RAISE_FLAG 연속 불가능한 예외로 오류를 발생시키는 옵션입니다. (D3D11_RAISE_FLAG)
D3D11_RESOURCE_DIMENSION 사용 중인 리소스의 유형을 식별합니다. (D3D11_RESOURCE_DIMENSION)
D3D11_RESOURCE_MISC_FLAG 리소스에 대한 옵션을 식별합니다.
D3D11_RTV_DIMENSION 이러한 플래그는 렌더링 대상으로 볼 리소스 유형을 식별합니다.
D3D11_SHADER_CACHE_SUPPORT_FLAGS

현재 그래픽 드라이버의 셰이더 캐싱 지원 수준을 설명합니다. (D3D11_SHADER_CACHE_SUPPORT_FLAGS)
D3D11_SHADER_MIN_PRECISION_SUPPORT 셰이더 단계에서 최소 정밀도 수준을 지정하는 값입니다.
D3D11_SHARED_RESOURCE_TIER 현재 그래픽 드라이버에서 공유 리소스에 대한 지원 수준을 지정하는 상수를 정의합니다.
D3D11_STANDARD_MULTISAMPLE_QUALITY_LEVELS 다중 샘플 패턴 형식을 지정합니다.
D3D11_STENCIL_OP 깊이 스텐실 테스트 중에 수행할 수 있는 스텐실 작업입니다. (D3D11_STENCIL_OP)
D3D11_TEXTURE_ADDRESS_MODE 텍스처 경계 밖에 있는 텍스처 좌표를 확인하는 기술을 식별합니다. (D3D11_TEXTURE_ADDRESS_MODE)
D3D11_TEXTURECUBE_FACE 큐브 텍스처의 다른 얼굴입니다. (D3D11_TEXTURECUBE_FACE)
D3D11_TILED_RESOURCES_TIER 타일형 리소스가 지원되는 계층 수준을 나타냅니다.
D3D11_UAV_DIMENSION 순서가 지정되지 않은 액세스 보기 옵션입니다.
D3D11_USAGE 렌더링하는 동안 예상되는 리소스 사용을 식별합니다. 사용량은 CPU 및/또는 GPU(그래픽 처리 장치)에서 리소스에 액세스할 수 있는지 여부를 직접 반영합니다.
D3D11_VDOV_DIMENSION 비디오 디코딩 출력 보기에서 사용되는 리소스에 액세스하는 방법을 지정합니다.
D3D11_VIDEO_DECODER_BUFFER_TYPE 디코딩을 위해 압축된 버퍼의 형식을 지정합니다.

D3D11_VIDEO_FRAME_FORMAT 비디오 스트림이 인터레이스되는 방법을 설명합니다. (D3D11_VIDEO_FRAME_FORMAT)
D3D11_VIDEO_PROCESSOR_ALPHA_FILL_MODE 비디오 처리를 위한 알파 채우기 모드를 지정합니다. (D3D11_VIDEO_PROCESSOR_ALPHA_FILL_MODE)
D3D11_VIDEO_PROCESSOR_AUTO_STREAM_CAPS 비디오 프로세서의 자동 이미지 처리 기능을 지정합니다.
D3D11_VIDEO_PROCESSOR_DEVICE_CAPS Microsoft Direct3D 11 비디오 프로세서에 대한 비디오 처리 기능을 정의합니다.
D3D11_VIDEO_PROCESSOR_FEATURE_CAPS Microsoft Direct3D 11 비디오 프로세서가 지원할 수 있는 기능을 정의합니다.
D3D11_VIDEO_PROCESSOR_FILTER 비디오 프로세서 필터를 식별합니다.
D3D11_VIDEO_PROCESSOR_FILTER_CAPS Microsoft Direct3D 11 비디오 프로세서에 대한 이미지 필터 기능을 정의합니다.
D3D11_VIDEO_PROCESSOR_FORMAT_CAPS Microsoft Direct3D 11 비디오 프로세서의 입력 형식과 관련된 기능을 정의합니다.
D3D11_VIDEO_PROCESSOR_FORMAT_SUPPORT 비디오 처리에 비디오 형식을 사용할 수 있는 방법을 지정합니다.
D3D11_VIDEO_PROCESSOR_ITELECINE_CAPS 비디오 프로세서의 역 텔레신(IVTC) 기능을 지정합니다.
D3D11_VIDEO_PROCESSOR_NOMINAL_RANGE YUV 데이터의 광도 범위에 대한 값을 지정합니다.
D3D11_VIDEO_PROCESSOR_OUTPUT_RATE 비디오 프로세서가 입력 스트림에서 출력 프레임을 생성하는 속도를 지정합니다.

D3D11_VIDEO_PROCESSOR_PROCESSOR_CAPS 디인터레이싱, IVTC(역방향 텔레신) 및 프레임 속도 변환과 관련된 비디오 처리 기능을 지정합니다.
D3D11_VIDEO_PROCESSOR_ROTATION 비디오 회전 상태를 지정합니다.
D3D11_VIDEO_PROCESSOR_STEREO_CAPS Microsoft Direct3D 11 비디오 프로세서에 대한 스테레오 3D 기능을 정의합니다.
D3D11_VIDEO_PROCESSOR_STEREO_FLIP_MODE 스테레오 3D 비디오의 경우 프레임 0 또는 프레임 1의 데이터가 가로 또는 세로로 대칭 이동되는지 여부를 지정합니다.
D3D11_VIDEO_PROCESSOR_STEREO_FORMAT 스테레오 3D 비디오 프레임의 메모리 레이아웃을 지정합니다.
D3D11_VIDEO_USAGE 비디오 프로세서의 용도를 지정합니다.
D3D11_VPIV_DIMENSION 비디오 프로세서 입력 보기에서 사용되는 리소스에 액세스하는 방법을 지정합니다.
D3D11_VPOV_DIMENSION 비디오 프로세서 출력 뷰에서 사용되는 리소스에 액세스하는 방법을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d3d11_1.h 헤더

아티클2023. 08. 22.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 11 그래픽
- Microsoft Media Foundation

d3d11_1.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID3D11BlendState1 혼합 상태 인터페이스에는 출력 병합기 단계에 바인딩할 수 있는 혼합 상태에 대한 설명이 있습니다. 이 혼합 상태 인터페이스는 논리 작업과 혼합 작업을 지원합니다.
ID3D11Device1 디바이스 인터페이스는 가상 어댑터를 나타냅니다. 리소스를 만드는 데 사용됩니다. ID3D11Device1은 ID3D11Device의 메서드에 새 메서드를 추가합니다.
ID3D11DeviceContext1 디바이스 컨텍스트 인터페이스는 디바이스 컨텍스트를 나타냅니다. 명령을 렌더링하는 데 사용됩니다. ID3D11DeviceContext1은 ID3D11DeviceContext의 메서드에 새 메서드를 추가합니다.
ID3D11RasterizerState1 래스터라이저 상태 인터페이스에는 래스터라이저 단계에 바인딩할 수 있는 래스터라이저 상태에 대한 설명이 있습니다. 이 래스터라이저 상태 인터페이스는 강제 샘플 수를 지원합니다.
ID3D11VideoContext1 Microsoft Direct3D 11 디바이스의 비디오 기능을 제공합니다. (ID3D11VideoContext1)
ID3D11VideoDevice1 Microsoft Direct3D 11 디바이스의 비디오 디코딩 및 비디오 처리 기능을 제공합니다. (ID3D11VideoDevice1)
ID3D11VideoProcessorEnumerator1 Microsoft Direct3D 11 디바이스의 비디오 프로세서 기능을 열거합니다. (ID3D11VideoProcessorEnumerator1)

ID3DDeviceContextState ID3DDeviceContextState 인터페이스는 Microsoft Direct3D 디바이스에 대한 상태 및 동작 정보를 포함하는 컨텍스트 상태 개체를 나타냅니다.
ID3DUserDefinedAnnotation ID3DUserDefinedAnnotation 인터페이스를 사용하면 애플리케이션에서 애플리케이션의 코드 흐름 내에서 개념 섹션 및 마커를 설명할 수 있습니다.

구조체

CD3D11_BLEND_DESC1 ID3D11Device1::CreateBlendState1을 호출하여 혼합 상태 개체를 만드는 데 사용하는 혼합 상태를 설명합니다. D
CD3D11_RASTERIZER_DESC1 CD3D11_RASTERIZER_DESC1(d3d11_1.h) 구조체는 래스터라이저 상태를 설명합니다.
D3D11_BLEND_DESC1 ID3D11Device1::CreateBlendState1을 호출하여 혼합 상태 개체를 만드는 데 사용하는 혼합 상태를 설명합니다. (D3D11_BLEND_DESC1)
D3D11_KEY_EXCHANGE_HW_PROTECTION_DATA 하드웨어 콘텐츠 보호를 위한 주요 교환 데이터를 나타냅니다.
D3D11_KEY_EXCHANGE_HW_PROTECTION_INPUT_DATA 하드웨어 콘텐츠 보호를 위한 키 교환 입력 데이터를 나타냅니다.
D3D11_KEY_EXCHANGE_HW_PROTECTION_OUTPUT_DATA 하드웨어 콘텐츠 보호를 위한 주요 교환 출력 데이터를 나타냅니다.
D3D11_RASTERIZER_DESC1 래스터라이저 상태를 설명합니다. (D3D11_RASTERIZER_DESC1)
D3D11_RENDER_TARGET_BLEND_DESC1 렌더링 대상의 혼합 상태를 설명합니다. (D3D11_RENDER_TARGET_BLEND_DESC1)

D3D11_VIDEO_DECODER_BEGIN_FRAME_CRYPTO_SESSION ID3D11VideoContext::D ecoderBeginFrame 메서드에 데이터를 제공합니다.
D3D11_VIDEO_DECODER_BUFFER_DESC1 디코딩을 위해 압축된 버퍼에 대해 설명합니다. (D3D11_VIDEO_DECODER_BUFFER_DESC1)
D3D11_VIDEO_DECODER_SUB_SAMPLE_MAPPING_BLOCK 하위 샘플 매핑 블록에 대해 설명합니다.
D3D11_VIDEO_PROCESSOR_STREAM_BEHAVIOR_HINT ID3DVideoContext1::VideoProcessorGetBehaviorHints 메서드에 전달된 입력 스트림에 대한 정보를 제공합니다.
D3D11_VIDEO_SAMPLE_DESC 비디오 샘플을 설명합니다.

열거형

D3D11_1_CREATE_DEVICE_CONTEXT_STATE_FLAG ID3D11Device1::CreateDeviceContextState 메서드를 사용하여 디바이스 컨텍스트 상태 개체(ID3DDeviceContextState)를 만드는 데 사용되는 플래그에 대해 설명합니다.
D3D11_COPY_FLAGS 해당 리소스 내 지역의 복사 또는 업데이트 작업 중에 리소스의 기존 콘텐츠를 처리하는 방법을 지정합니다.
D3D11_CRYPTO_SESSION_STATUS ID3D11CryptoSession 인터페이스의 상태 나타냅니다.
D3D11_LOGIC_OP 렌더링 대상에 대해 구성할 논리 작업을 지정합니다. (D3D11_LOGIC_OP)
D3D11_VIDEO_DECODER_CAPS 비디오 디코더의 기능을 지정합니다.

D3D11_VIDEO_PROCESSOR_BEHAVIOR_HINTS 비디오 처리 작업을 수행하는 가장 효율적인 방법을 나타내는 플래그를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d3d11_2.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct3D 11 그래픽에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 11 그래픽

d3d11_2.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID3D11Device2 디바이스 인터페이스는 가상 어댑터를 나타냅니다. 리소스를 만드는 데 사용됩니다. ID3D11Device2는 ID3D11Device1의 메서드에 새 메서드를 추가합니다.
ID3D11DeviceContext2 디바이스 컨텍스트 인터페이스는 디바이스 컨텍스트를 나타냅니다. 명령을 렌더링하는 데 사용됩니다. ID3D11DeviceContext2는 ID3D11DeviceContext1의 메서드에 새 메서드를 추가합니다.

구조체

D3D11_PACKED_MIP_DESC 밉맵이 있는 타일식 리소스의 타일 구조에 대해 설명합니다. (D3D11_PACKED_MIP_DESC)
D3D11_SUBRESOURCE_TILING 타일형 하위 리소스 볼륨을 설명합니다. (D3D11_SUBRESOURCE_TILING)
D3D11_TILE_REGION_SIZE 타일형 영역의 크기를 설명합니다. (D3D11_TILE_REGION_SIZE)
D3D11_TILE_SHAPE 크기를 지정하여 타일의 모양을 설명합니다. (D3D11_TILE_SHAPE)
D3D11_TILED_RESOURCE_COORDINATE

타일형 리소스의 좌표를 설명합니다. (D3D11_TILED_RESOURCE_COORDINATE)

열거형

D3D11_CHECK_MULTISAMPLE_QUALITY_LEVELS_FLAG 다중 샘플 품질 수준을 검사 방법을 식별합니다.
D3D11_TILE_COPY_FLAG 타일을 복사하는 방법을 식별합니다.
D3D11_TILE_MAPPING_FLAG 타일 매핑 작업을 수행하는 방법을 식별합니다.
D3D11_TILE_RANGE_FLAG ID3D11DeviceContext2::UpdateTiles와 함께 사용할 타일 매핑 범위를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d3d11_3.h 헤더

아티클2023. 08. 22.

이 헤더는 Direct3D 11 그래픽에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 11 그래픽

d3d11_3.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID3D11Device3 디바이스 인터페이스는 가상 어댑터를 나타냅니다. 리소스를 만드는 데 사용됩니다. ID3D11Device3은 ID3D11Device2의 메서드에 새 메서드를 추가합니다.
ID3D11DeviceContext3 디바이스 컨텍스트 인터페이스는 디바이스 컨텍스트를 나타냅니다. 명령을 렌더링하는 데 사용됩니다. ID3D11DeviceContext3은 ID3D11DeviceContext2의 메서드에 새 메서드를 추가합니다.
ID3D11DeviceContext4 디바이스 컨텍스트 인터페이스는 디바이스 컨텍스트를 나타냅니다. 명령을 렌더링하는 데 사용됩니다. ID3D11DeviceContext4는 ID3D11DeviceContext3의 메서드에 새 메서드를 추가합니다.
ID3D11Fence 펜스, CPU 동기화에 사용되는 개체 및 하나 이상의 GPU를 나타냅니다. (ID3D11Fence)
ID3D11Query1 GPU(그래픽 처리 장치)에서 정보를 쿼리하기 위한 쿼리 개체를 나타냅니다.
ID3D11RasterizerState2 래스터라이저 상태 인터페이스는 래스터라이저 단계에 바인딩할 수 있는 래스터라이저 상태에 대한 설명을 포함합니다. 이 래스터라이저 상태 인터페이스는 강제 샘플 수 및 보수적 래스터화 모드를 지원합니다.
ID3D11RenderTargetView1 render-target-view 인터페이스는 렌더링 중에 액세스할 수 있는 렌더링 대상 하위 리소스를 나타냅니다.

ID3D11ShaderResourceView1 셰이더 리소스 뷰 인터페이스는 렌더링 중에 셰이더가 액세스할 수 있는 하위 리소스를 나타냅니다. 셰이더 리소스의 예로는 상수 버퍼, 텍스처 버퍼 및 텍스처가 있습니다.
ID3D11Texture2D1 2D 텍스처 인터페이스는 구조화된 메모리인 텍셀 데이터를 나타냅니다.
ID3D11Texture3D1 3D 텍스처 인터페이스는 구조화된 메모리인 텍셀 데이터를 나타냅니다.
ID3D11UnorderedAccessView1 순서가 지정되지 않은 액세스 뷰 인터페이스는 렌더링 중에 파이프라인이 액세스할 수 있는 리소스 부분을 나타냅니다.

구조체

CD3D11_QUERY_DESC1 쿼리를 설명합니다. D
CD3D11_RASTERIZER_DESC2 CD3D11_RASTERIZER_DESC2(d3d11_3.h) 구조체는 래스터라이저 상태를 설명합니다.
CD3D11_RENDER_TARGET_VIEW_DESC1 렌더링 대상 뷰를 사용하여 액세스할 수 있는 리소스의 하위 리소스에 대해 설명합니다. D
CD3D11_SHADER_RESOURCE_VIEW_DESC1 셰이더 리소스 뷰를 설명합니다. D
CD3D11_TEXTURE2D_DESC1 CD3D11_TEXTURE2D_DESC1(d3d11_3.h) 구조체는 2D 텍스처를 설명합니다.
CD3D11_TEXTURE3D_DESC1 CD3D11_TEXTURE3D_DESC1(d3d11_3.h) 구조체는 3D 텍스처를 설명합니다.

CD3D11_UNORDERED_ACCESS_VIEW_DESC1 순서가 지정되지 않은 액세스 보기를 사용하여 액세스할 수 있는 리소스의 하위 리소스에 대해 설명합니다. D
D3D11_QUERY_DESC1 쿼리를 설명합니다. (D3D11_QUERY_DESC1)
D3D11_RASTERIZER_DESC2 래스터라이저 상태를 설명합니다. (D3D11_RASTERIZER_DESC2)
D3D11_RENDER_TARGET_VIEW_DESC1 렌더링 대상 뷰를 사용하여 액세스할 수 있는 리소스의 하위 리소스에 대해 설명합니다. (D3D11_RENDER_TARGET_VIEW_DESC1)
D3D11_SHADER_RESOURCE_VIEW_DESC1 셰이더 리소스 뷰를 설명합니다. (D3D11_SHADER_RESOURCE_VIEW_DESC1)
D3D11_TEX2D_ARRAY_RTV1 렌더링 대상 보기에서 사용할 2D 텍스처 배열의 하위 리소스에 대해 설명합니다. (D3D11_TEX2D_ARRAY_RTV1)
D3D11_TEX2D_ARRAY_SRV1 셰이더 리소스 뷰에서 사용할 2D 텍스처 배열의 하위 리소스에 대해 설명합니다. (D3D11_TEX2D_ARRAY_SRV1)
D3D11_TEX2D_ARRAY_UAV1 순서가 지정되지 않은 액세스 2D 텍스처 리소스의 배열에 대해 설명합니다. (D3D11_TEX2D_ARRAY_UAV1)
D3D11_TEX2D_RTV1 렌더링 대상 보기에서 사용할 2D 텍스처의 하위 리소스에 대해 설명합니다. (D3D11_TEX2D_RTV1)
D3D11_TEX2D_SRV1 셰이더 리소스 뷰에서 사용할 2D 텍스처의 하위 리소스에 대해 설명합니다. (D3D11_TEX2D_SRV1)
D3D11_TEX2D_UAV1 순서가 지정되지 않은 액세스 2D 텍스처 리소스에 대해 설명합니다. (D3D11_TEX2D_UAV1)

D3D11_TEXTURE2D_DESC1 2D 텍스처를 설명합니다. (D3D11_TEXTURE2D_DESC1)
D3D11_TEXTURE3D_DESC1 3D 텍스처를 설명합니다. (D3D11_TEXTURE3D_DESC1)
D3D11_UNORDERED_ACCESS_VIEW_DESC1 순서가 지정되지 않은 액세스 보기를 사용하여 액세스할 수 있는 리소스의 하위 리소스에 대해 설명합니다. (D3D11_UNORDERED_ACCESS_VIEW_DESC1)

열거형

D3D11_CONSERVATIVE_RASTERIZATION_MODE 보수적인 래스터화가 켜지거나 꺼져 있는지 여부를 식별합니다. (D3D11_CONSERVATIVE_RASTERIZATION_MODE)
D3D11_CONTEXT_TYPE 쿼리가 발생하는 컨텍스트를 지정합니다.
D3D11_FENCE_FLAG 펜스 옵션을 지정합니다. (D3D11_FENCE_FLAG)
D3D11_TEXTURE_LAYOUT 텍스처 레이아웃 옵션을 지정합니다. (D3D11_TEXTURE_LAYOUT)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d3d11_4.h 헤더

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 11의 그래픽
- Microsoft Media Foundation

d3d11_4.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

테이블 확장

ID3D11Device4 디바이스 인터페이스는 가상 어댑터를 나타냅니다. 리소스를 만드는 데 사용됩니다. ID3D11Device4는 RegisterDeviceRemovedEvent 및 UnregisterDeviceRemoved와 같은 ID3D11Device3의 메서드에 새 메서드를 추가합니다.
ID3D11Device5 디바이스 인터페이스는 가상 어댑터를 나타냅니다. 리소스를 만드는 데 사용됩니다. ID3D11Device5는 ID3D11Device4에 새 메서드를 추가합니다.
ID3D11Multithread 다중 스레드 애플리케이션의 중요한 섹션에 대한 스레딩 보호를 제공합니다.
ID3D11VideoContext2 Microsoft Direct3D 11 디바이스의 비디오 기능을 제공합니다. (ID3D11VideoContext2)
id3D11VideoContext3 Microsoft Direct3D 11 디바이스의 비디오 기능을 제공합니다. (ID3D11VideoContext3)
ID3D11VideoDevice2 Microsoft Direct3D 11 디바이스의 비디오 디코딩 및 비디오 처리 기능을 제공합니다.

구조

테이블 확장

D3D11_FEATURE_DATA_D3D11_OPTIONS4 현재 그래픽 드라이버의 Direct3D 11.4 기능 옵션에 대해 설명합니다.
D3D11_FEATURE_DATA_VIDEO_DECODER_HISTOGRAM 지정된 기능이 D3D11_FEATURE_VIDEO_DECODER_HISTOGRAM 경우 ID3D11VideoDevice2::CheckFeatureSupport 호출에 대한 데이터를 제공합니다.
D3D11_VIDEO_DECODER_BUFFER_DESC2

열거 목록

테이블 확장

D3D11_CRYPTO_SESSION_KEY_EXCHANGE_FLAGS 암호화 세션 키 교환 플래그를 지정하는 상수를 정의합니다.
D3D11_FEATURE_VIDEO 쿼리할 Direct3D 11 비디오 기능 또는 기능 집합을 지정합니다.
D3D11_VIDEO_DECODER_HISTOGRAM_COMPONENT 구성 요소별 히스토그램 정보의 배열에 대한 인덱스를 지정합니다. (D3D11_VIDEO_DECODER_HISTOGRAM_COMPONENT)
D3D11_VIDEO_DECODER_HISTOGRAM_COMPONENT_FLAGS 비디오 디코딩 히스토그램과 함께 사용되는 구성 요소의 하위 집합을 나타내는 플래그입니다. (D3D11_VIDEO_DECODER_HISTOGRAM_COMPONENT_FLAGS)

Last updated on 2025. 06. 18.

d3d11on12.h 헤더

아티클2023. 08. 23.

이 헤더는 Direct3D 12 그래픽에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 12 그래픽

d3d11on12.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID3D11On12Device Direct3D11on12에 대한 D3D11 리소스 만들기, 래핑 및 릴리스를 처리합니다.
ID3D11On12Device1 Direct3D 11 디바이스를 전달하지만 대신 Direct3D 12를 활용하려는 구성 요소와의 상호 운용성을 향상할 수 있습니다.
ID3D11On12Device2 Direct3D 11 API를 통해 만든 리소스를 가져와 Direct3D 12에서 사용할 수 있습니다.

Functions

D3D11On12CreateDevice Direct3D 12에서 Direct3D 11 기능을 사용하는 디바이스를 만들고 Direct3D 11 interop에 사용할 기존 Direct3D 12 디바이스를 지정합니다.

구조체

D3D11_RESOURCE_FLAGS ID3D11On12Device::CreateWrappedResource와 함께 사용되어 바인딩 플래그, 기타 플래그 및 CPU 액세스 플래그를 포함하여 리소스 속성 또는 힙 속성에서 유추되는 플래그를 재정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d3d11sdklayers.h 헤더

아티클2023. 08. 23.

이 헤더는 Direct3D 11 그래픽에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 11 그래픽

d3d11sdklayers.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID3D11Debug 디버그 인터페이스는 디버그 설정을 제어하고 파이프라인 상태의 유효성을 검사하며 디버그 계층이 켜져 있는 경우에만 사용할 수 있습니다. (ID3D11Debug)
ID3D11InfoQueue 정보 큐 인터페이스는 디버그 메시지를 저장, 검색 및 필터링합니다. 큐는 메시지 큐, 선택적 스토리지 필터 스택 및 선택적 검색 필터 스택으로 구성됩니다. (ID3D11InfoQueue)
ID3D11RefDefaultTrackingOptions 기본 추적 인터페이스 집합은 참조 기본 추적 옵션을 설정합니다.
ID3D11RefTrackingOptions 추적 인터페이스는 참조 추적 옵션을 설정합니다.
ID3D11SwitchToRef ID3D11SwitchToRef 인터페이스
ID3D11TracingDevice 추적 디바이스 인터페이스는 셰이더 추적 정보를 설정하여 셰이더 실행의 정확한 로깅 및 재생을 가능하게 합니다.

구조체

D3D11_INFO_QUEUE_FILTER 디버그 메시지 필터; 에는 허용하거나 거부할 메시지 유형의 목록이 포함되어 있습니다. (D3D11_INFO_QUEUE_FILTER)
D3D11_INFO_QUEUE_FILTER_DESC 특정 유형의 메시지가 필터를 통과하도록 허용하거나 거부합니다. (D3D11_INFO_QUEUE_FILTER_DESC)
D3D11_MESSAGE 정보 큐의 디버그 메시지입니다. (D3D11_MESSAGE)

열거형

D3D11_MESSAGE_CATEGORY 디버그 메시지의 범주입니다. (D3D11_MESSAGE_CATEGORY)
D3D11_MESSAGE_ID 정보 큐 필터를 설정하기 위한 메시지 디버그(D3D11_INFO_QUEUE_FILTER 참조) 이러한 메시지를 사용하여 스토리지 및 검색 필터를 통과하도록 메시지 범주를 허용하거나 거부합니다.
D3D11_MESSAGE_SEVERITY 정보 큐에 대한 메시지 심각도 수준을 디버그합니다. (D3D11_MESSAGE_SEVERITY)
D3D11_RLDO_FLAGS 디바이스 개체의 수명에 대해 보고할 정보의 양에 대한 옵션입니다.
D3D11_SHADER_TRACKING_OPTIONS 셰이더 디버그 추적을 수행하는 방법을 지정하는 옵션입니다.
D3D11_SHADER_TRACKING_RESOURCE_TYPE 추적할 리소스 종류를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d3d11shader.h 헤더

아티클2023. 08. 23.

이 헤더는 Direct3D 11 그래픽에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 11 그래픽

d3d11shader.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID3D11FunctionLinkingGraph 함수 연결-그래프 인터페이스는 서로 값을 전달하는 미리 컴파일된 함수 호출 시퀀스로 구성된 셰이더를 생성하는 데 사용됩니다.
ID3D11FunctionParameterReflection function-parameter-reflection 인터페이스는 function-parameter 정보에 액세스합니다. (ID3D11FunctionParameterReflection)
ID3D11FunctionReflection 함수-리플렉션 인터페이스는 함수 정보에 액세스합니다. (ID3D11FunctionReflection)
ID3D11LibraryReflection 라이브러리 리플렉션 인터페이스는 라이브러리 정보에 액세스합니다. (ID3D11LibraryReflection)
ID3D11Linker 링커 인터페이스는 셰이더 모듈을 연결하는 데 사용됩니다.
ID3D11LinkingNode 연결 노드 인터페이스는 셰이더 연결에 사용됩니다.
ID3D11모듈 모듈 인터페이스는 리소스 다시 바인딩에 사용되는 모듈의 instance 만듭니다.
ID3D11ModuleInstance 모듈 instance 인터페이스는 리소스 다시 바인딩에 사용됩니다.

ID3D11ShaderReflection 셰이더 리플렉션 인터페이스는 셰이더 정보에 액세스합니다. (ID3D11ShaderReflection)
ID3D11ShaderReflectionConstantBuffer 이 셰이더 리플렉션 인터페이스는 상수 버퍼에 대한 액세스를 제공합니다. (ID3D11ShaderReflectionConstantBuffer)
ID3D11ShaderReflectionType 이 셰이더 리플렉션 인터페이스는 변수 형식에 대한 액세스를 제공합니다. (ID3D11ShaderReflectionType)
ID3D11ShaderReflectionVariable 이 셰이더 리플렉션 인터페이스는 변수에 대한 액세스를 제공합니다. (ID3D11ShaderReflectionVariable)

구조체

D3D11_FUNCTION_DESC 함수를 설명합니다. (D3D11_FUNCTION_DESC)
D3D11_LIBRARY_DESC 라이브러리에 대해 설명합니다. (D3D11_LIBRARY_DESC)
D3D11_PARAMETER_DESC 함수 매개 변수를 설명합니다. (D3D11_PARAMETER_DESC)
D3D11_SHADER_BUFFER_DESC 셰이더 상수 버퍼를 설명합니다. (D3D11_SHADER_BUFFER_DESC)
D3D11_SHADER_DESC 셰이더에 대해 설명합니다. (D3D11_SHADER_DESC)
D3D11_SHADER_INPUT_BIND_DESC 셰이더 리소스가 셰이더 입력에 바인딩되는 방법을 설명합니다. (D3D11_SHADER_INPUT_BIND_DESC)

D3D11_SHADER_TYPE_DESC 셰이더 변수 형식을 설명합니다. (D3D11_SHADER_TYPE_DESC)
D3D11_SHADER_VARIABLE_DESC 셰이더 변수에 대해 설명합니다. (D3D11_SHADER_VARIABLE_DESC)
D3D11_SIGNATURE_PARAMETER_DESC 셰이더 서명을 설명합니다. (D3D11_SIGNATURE_PARAMETER_DESC)

열거형

D3D11_SHADER_VERSION_TYPE 셰이더 유형을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d3d11shadertracing.h 헤더

아티클2023. 08. 23.

이 헤더는 Direct3D 11 그래픽에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 11 그래픽

d3d11shadertracing.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID3D11ShaderTrace ID3D11ShaderTrace 인터페이스는 셰이더 실행 추적을 가져오는 메서드를 구현합니다.
ID3D11ShaderTraceFactory ID3D11ShaderTraceFactory 인터페이스는 셰이더 추적 정보 개체를 생성하는 메서드를 구현합니다.

Functions

D3DDisassemble11Trace 셰이더 추적 단계에서 지정한 컴파일된 Microsoft HLSL(High Level Shader Language) 코드의 섹션을 디스어셈블합니다.

구조체

D3D11_COMPUTE_SHADER_TRACE_DESC 추적할 컴퓨팅 셰이더의 instance 설명합니다.
D3D11_DOMAIN_SHADER_TRACE_DESC 추적할 도메인 셰이더의 instance 설명합니다.

D3D11_GEOMETRY_SHADER_TRACE_DESC 추적할 기하 도형 셰이더의 instance 설명합니다.
D3D11_HULL_SHADER_TRACE_DESC 추적할 헐 셰이더의 instance 설명합니다.
D3D11_PIXEL_SHADER_TRACE_DESC 추적할 픽셀 셰이더의 instance 설명합니다.
D3D11_SHADER_TRACE_DESC 셰이더 추적 개체를 설명합니다.
D3D11_TRACE_REGISTER 추적 레지스터에 대해 설명합니다.
D3D11_TRACE_STATS 추적에 대한 통계를 지정합니다.
D3D11_TRACE_STEP 명령인 추적 단계를 설명합니다.
D3D11_TRACE_VALUE 추적 값을 설명합니다.
D3D11_VERTEX_SHADER_TRACE_DESC 추적할 꼭짓점 셰이더의 instance 설명합니다.

열거형

D3D11_SHADER_TYPE 추적을 위한 셰이더 유형을 식별합니다.
D3D11_TRACE_GS_INPUT_PRIMITIVE 기하 도형 셰이더 입력 기본 형식을 식별합니다.

D3D11_TRACE_REGISTER_TYPE 추적 레지스터의 유형을 식별합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d3d12.h 헤더

아티클2023. 03. 14.

이 헤더는 Direct3D 12 그래픽에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 12 그래픽

d3d12.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID3D12CommandAllocator GPU(그래픽 처리 장치) 명령에 대한 스토리지 할당을 나타냅니다.
ID3D12CommandList ID3D12GraphicsCommandList가 상속하는 인터페이스입니다. GPU가 실행하는 순서가 지정된 명령 집합을 나타내며 확장이 그래픽(예: 컴퓨팅 및 복사)에 대한 명령 목록보다 다른 명령 목록을 지원할 수 있도록 허용합니다.
ID3D12CommandQueue 명령 목록을 제출하고, 명령 목록 실행을 동기화하고, 명령 큐를 계측하고, 리소스 타일 매핑을 업데이트하는 메서드를 제공합니다.
ID3D12CommandSignature 명령 서명 개체를 사용하면 앱에서 사용할 버퍼 형식, 명령 유형 및 리소스 바인딩을 포함하여 간접 그리기를 지정할 수 있습니다.
ID3D12DescriptorHeap 설명자 힙은 설명자의 연속 할당 컬렉션으로, 모든 설명자에 대한 하나의 할당입니다.
ID3D12Device 가상 어댑터를 나타냅니다. 명령 할당자, 명령 목록, 명령 큐, 펜스, 리소스, 파이프라인 상태 개체, 힙, 루트 서명, 샘플러 및 여러 리소스 뷰를 만드는 데 사용됩니다.
ID3D12Device1 가상 어댑터를 나타내고 ID3D12Device에서 제공하는 메서드 범위를 확장합니다.

ID3D12Device10 TBD
ID3D12Device2 가상 어댑터를 나타냅니다. 이 인터페이스는 ID3D12Device1을 확장하여 파이프라인 상태 스트림 설명에서 파이프라인 상태 개체를 만듭니다.
ID3D12Device3 가상 어댑터를 나타냅니다. 이 인터페이스는 ID3D12Device2를 확장하여 GPU 오류 또는 디바이스 제거 시나리오의 경우에도 지속되는 시스템 메모리에 특수한 용도의 진단 힙을 만들 수 있도록 지원합니다.
ID3D12Device4 가상 어댑터를 나타냅니다. 이 인터페이스는 ID3D12Device3을 확장합니다.
ID3D12Device5 가상 어댑터를 나타냅니다. 이 인터페이스는 ID3D12Device4를 확장합니다.
ID3D12Device6 가상 어댑터를 나타냅니다. 이 인터페이스는 ID3D12Device5를 확장합니다.
ID3D12Device7 가상 어댑터를 나타냅니다. 이 인터페이스는 ID3D12Device6을 확장합니다.
ID3D12Device8 가상 어댑터를 나타냅니다. 이 인터페이스는 ID3D12Device7을 확장합니다.
ID3D12Device9 가상 어댑터를 나타냅니다. 이 인터페이스는 ID3D12Device8 을 확장하여 셰이더 캐시를 관리하는 메서드를 추가합니다.
ID3D12DeviceChild ID3D12PipelineLibrary, ID3D12CommandList, ID3D12Pageable 및 ID3D12RootSignature를 포함하여 다른 핵심 인터페이스가 상속하는 인터페이스입니다. 생성된 디바이스 개체로 돌아가는 메서드를 제공합니다.
ID3D12DeviceRemovedExtendedData 디바이스 DRED(확장 데이터 제거) 데이터에 대한 런타임 액세스를 제공합니다.

ID3D12DeviceRemovedExtendedDataSettings 이 인터페이스는 DRED(디바이스 제거된 확장 데이터) 설정을 제어합니다.
ID3D12Fence 펜스, CPU 동기화에 사용되는 개체 및 하나 이상의 GPU를 나타냅니다. (ID3D12Fence)
ID3D12Fence1 펜스를 나타냅니다. 이 인터페이스는 ID3D12Fence를 확장하고 원래 펜스를 만드는 데 사용되는 플래그의 검색을 지원합니다.
ID3D12GraphicsCommandList 렌더링을 위한 그래픽 명령 목록을 캡슐화합니다. 명령 목록 실행을 계측하고 파이프라인 상태를 설정 및 지우기 위한 API를 포함합니다.
ID3D12GraphicsCommandList1 렌더링을 위한 그래픽 명령 목록을 캡슐화하고, 인터페이스를 확장하여 프로그래밍 가능한 샘플 위치, 지연 래치 기술 구현을 위한 원자성 복사본 및 선택적 깊이 경계 테스트를 지원합니다.
ID3D12GraphicsCommandList2 렌더링을 위한 그래픽 명령 목록을 캡슐화하여 버퍼에 직접 직접 값을 쓸 수 있도록 인터페이스를 확장합니다.
ID3D12GraphicsCommandList3 렌더링을 위한 그래픽 명령 목록을 캡슐화합니다.
ID3D12GraphicsCommandList4 렌더링을 위한 그래픽 명령 목록을 캡슐화하고, 광선 추적 및 렌더링 패스를 지원하도록 인터페이스를 확장합니다.
ID3D12GraphicsCommandList5 렌더링을 위한 그래픽 명령 목록을 캡슐화하여 VRS(가변 속도 음영)를 지원하도록 인터페이스를 확장합니다.
ID3D12GraphicsCommandList7 TBD

ID3D12Heap 힙은 실제 메모리를 관리하는 데 사용되는 연속 메모리 할당의 추상화입니다. 이 힙은 ID3D12Resource 개체와 함께 사용하여 배치된 리소스 또는 예약된 리소스를 지원할 수 있습니다.
ID3D12LifetimeOwner 개체의 수명 변경에 대한 알림을 받는 데 사용되는 애플리케이션 정의 콜백을 나타냅니다.
ID3D12LifetimeTracker 수명 추적 개체의 수명을 제어하기 위한 기능을 나타냅니다.
ID3D12MetaCommand 메타 명령을 나타냅니다. 메타 명령은 IHV(독립 하드웨어 공급업체)에 의해 가속화되는 알고리즘을 나타내는 Direct3D 12 개체입니다. 드라이버에서 구현하는 명령 생성기에 대한 불투명 참조입니다.
ID3D12Object ID3D12Device 및 ID3D12DeviceChild가 상속하는 인터페이스입니다. 프라이빗 데이터를 연결하고 개체 이름에 주석을 추가하는 메서드를 제공합니다.
ID3D12Pageable 다른 많은 핵심 인터페이스가 상속하는 인터페이스입니다. 개체 형식이 GPU에 액세스할 수 있는 메모리의 양을 캡슐화한다는 것을 나타냅니다. 하지만 애플리케이션이 개체의 상주를 조작할 수 있는지 여부를 강력하게 나타내지는 않습니다.
ID3D12PipelineLibrary 파이프라인 라이브러리, 특히 개별 PSO 로드 및 검색을 관리합니다.
ID3D12PipelineLibrary1 파이프라인 라이브러리를 관리합니다. 이 인터페이스는 ID3D12PipelineLibrary를 확장하여 파이프라인 상태 스트림 설명에서 PSO를 로드합니다.
ID3D12PipelineState 현재 설정된 모든 셰이더의 상태와 특정 고정 함수 상태 개체를 나타냅니다.
ID3D12ProtectedResourceSession 보호된 리소스 세션의 유효성을 모니터링합니다. (ID3D12ProtectedResourceSession)
ID3D12ProtectedResourceSession1 보호된 리소스 세션의 유효성을 모니터링합니다. (ID3D12ProtectedResourceSession1)

ID3D12ProtectedSession 다양한 유형의 세션에서 세션의 유효성을 일관되게 모니터링할 수 있는 기본 기능을 제공합니다.
ID3D12QueryHeap 쿼리 힙을 관리합니다. 쿼리 힙은 인덱스에서 참조하는 쿼리 배열을 보유합니다.
ID3D12Resource CPU 및 GPU가 실제 메모리 또는 힙에 읽고 쓰는 일반화된 기능을 캡슐화합니다. 셰이더 샘플링에 최적화된 다차원 데이터뿐만 아니라 간단한 데이터 배열을 구성하고 조작하기 위한 추상화도 포함되어 있습니다.
ID3D12RootSignature 루트 서명은 그래픽 파이프라인에 바인딩되는 리소스를 정의합니다. 루트 서명은 앱에서 구성되고 명령 목록을 셰이더에 필요한 리소스에 연결합니다. 현재 앱당 하나의 그래픽과 하나의 컴퓨팅 루트 서명이 있습니다.
ID3D12RootSignatureDeserializer 직렬화된 루트 서명 버전 1.0의 역직렬화된 D3D12_ROOT_SIGNATURE_DESC 데이터 구조를 반환하는 메서드를 포함합니다.
ID3D12SDKConfiguration SDK 구성 메서드를 제공합니다.
ID3D12ShaderCacheSession 셰이더 캐시 세션을 나타냅니다.
ID3D12StateObject 애플리케이션이 단일 단위로 관리하고 컴파일 또는 최적화와 같이 처리하도록 드라이버에 원자성으로 제공되는 셰이더를 포함한 구성 상태의 가변 양을 나타냅니다.
ID3D12StateObjectProperties ID3D12StateObject의 속성을 가져오고 설정하는 메서드를 제공합니다.
ID3D12Tools 이 인터페이스는 PIX와 같은 도구에 대한 런타임을 구성하는 데 사용됩니다. 다른 시나리오에 대해서는 의도되거나 지원되지 않습니다.

ID3D12VersionedRootSignatureDeserializer 직렬화된 루트 서명의 모든 버전의 역직렬화된 D3D12_ROOT_SIGNATURE_DESC1 데이터 구조를 반환하는 메서드를 포함합니다.
ID3D12VirtualizationGuestDevice TBD

Functions

D3D12CreateDevice 디스플레이 어댑터를 나타내는 디바이스를 만듭니다. (D3D12CreateDevice)
D3D12CreateRootSignatureDeserializer 레이아웃 정의(D3D12_ROOT_SIGNATURE_DESC)를 확인할 수 있도록 루트 서명을 역직렬화합니다.
D3D12CreateVersionedRootSignatureDeserializer GetUnconvertedRootSignatureDesc를 통해 역직렬화된 데이터 구조를 반환할 수 있는 인터페이스를 생성합니다.
D3D12EnableExperimentalFeatures 실험적 기능 목록을 사용하도록 설정합니다.
D3D12GetDebugInterface 디버그 인터페이스를 가져옵니다.
D3D12GetInterface 시스템이 Windows 개발자 모드에 있을 때 런타임에 SDK 버전을 선택합니다.
D3D12SerializeRootSignature ID3D12Device::CreateRootSignature에 전달할 수 있는 루트 서명 버전 1.0을 직렬화합니다.
D3D12SerializeVersionedRootSignature ID3D12Device::CreateRootSignature에 전달할 수 있는 모든 버전의 루트 서명을 직렬화합니다.

구조체

D3D12_AUTO_BREADCRUMB_NODE 연결된 목록의 노드로 디바이스 DRED(제거된 확장 데이터) 자동 이동 경로 데이터를 나타냅니다.
D3D12_BARRIER_GROUP 지정된 형식의 장벽 그룹을 설명합니다.
D3D12_BARRIER_SUBRESOURCE_RANGE 논리적으로 인접한 하위 리소스 범위를 전환할 수 있습니다.
D3D12_BLEND_DESC 혼합 상태를 설명합니다. (D3D12_BLEND_DESC)
D3D12_BOX 3D 상자를 설명합니다.
D3D12_BUFFER_BARRIER 버퍼 메모리 액세스 장벽에 대해 설명합니다. 버퍼 장벽에서 리소스 메모리를 특정 액세스 유형에 대해 표시해야 하는 시기를 나타내는 데 사용됩니다.
D3D12_BUFFER_RTV 렌더링 대상 뷰에서 사용할 버퍼 리소스의 요소에 대해 설명합니다.
D3D12_BUFFER_SRV 셰이더 리소스 뷰에서 사용할 버퍼 리소스의 요소에 대해 설명합니다.
D3D12_BUFFER_UAV 순서가 지정되지 않은 액세스 뷰에서 사용할 버퍼의 요소에 대해 설명합니다. (D3D12_BUFFER_UAV)
D3D12_BUILD_RAYTRACING_ACCELERATION_STRUCTURE_DESC 광선 추적 가속 구조를 설명합니다. 이 구조를 ID3D12GraphicsCommandList4::BuildRaytracingAccelerationStructure에 전달하여 빌드할 가속 구조를 설명합니다.

D3D12_BUILD_RAYTRACING_ACCELERATION_STRUCTURE_INPUTS 광선 추적 가속 구조 빌드 작업에 대한 입력을 정의합니다. 이 구조체는 ID3D12GraphicsCommandList4::BuildRaytracingAccelerationStructure 및 ID3D12Device5::GetRaytracingAccelerationStructurePrebuildInfo에서 사용됩니다.
D3D12_BUILD_RAYTRACING_ACCELERATION_STRUCTURE_TOOLS_VISUALIZATION_HEADER 가속 구조 시각화의 GPU 메모리 레이아웃에 대해 설명합니다.
D3D12_CACHED_PIPELINE_STATE 파이프라인 상태를 저장합니다.
D3D12_CLEAR_VALUE 특정 리소스에 대한 명확한 작업을 최적화하는 데 사용되는 값을 설명합니다.
D3D12_COMMAND_QUEUE_DESC 명령 큐에 대해 설명합니다.
D3D12_COMMAND_SIGNATURE_DESC 명령 서명의 인수(매개 변수)를 설명합니다.
D3D12_COMPUTE_PIPELINE_STATE_DESC 컴퓨팅 파이프라인 상태 개체에 대해 설명합니다.
D3D12_CONSTANT_BUFFER_VIEW_DESC 볼 상수 버퍼에 대해 설명합니다.
D3D12_CPU_DESCRIPTOR_HANDLE CPU 설명자 핸들을 설명합니다.
D3D12_DEPTH_STENCIL_DESC 깊이 스텐실 상태를 설명합니다. (D3D12_DEPTH_STENCIL_DESC)
D3D12_DEPTH_STENCIL_DESC1 깊이 스텐실 상태를 설명합니다. (D3D12_DEPTH_STENCIL_DESC1)
D3D12_DEPTH_STENCIL_VALUE 깊이 및 스텐실 값을 지정합니다.

D3D12_DEPTH_STENCIL_VIEW_DESC 깊이 스텐실 보기에서 액세스할 수 있는 텍스처의 하위 리소스에 대해 설명합니다.
D3D12_DEPTH_STENCILOP_DESC 스텐실 테스트 결과에 따라 수행할 수 있는 스텐실 작업에 대해 설명합니다.
D3D12_DESCRIPTOR_HEAP_DESC 설명자 힙에 대해 설명합니다.
D3D12_DESCRIPTOR_RANGE 설명자 범위를 설명합니다.
D3D12_DESCRIPTOR_RANGE1 플래그가 있는 설명자 범위를 설명하여 변동성을 확인합니다.
D3D12_DEVICE_REMOVED_EXTENDED_DATA DRED(디바이스 제거 확장 데이터) 버전 1.0 데이터를 나타냅니다.
D3D12_DEVICE_REMOVED_EXTENDED_DATA1 디바이스 DRED(제거된 확장 데이터) 버전 1.1 데이터를 나타냅니다.
D3D12_DISCARD_REGION 무시 리소스 작업에 대한 세부 정보를 설명합니다.
D3D12_DISPATCH_ARGUMENTS 컴퓨팅 셰이더에서 사용할 디스패치 매개 변수에 대해 설명합니다.
D3D12_DISPATCH_RAYS_DESC ID3D12GraphicsCommandList4::D ispatchRays 호출로 시작된 광선 디스패치 작업의 속성을 설명합니다.
D3D12_DRAW_ARGUMENTS 그리기 인스턴스에 대한 매개 변수를 설명합니다.
D3D12_DRAW_INDEXED_ARGUMENTS 인덱싱된 인스턴스를 그리기 위한 매개 변수를 설명합니다.

D3D12_DRED_ALLOCATION_NODE 연결된 목록의 노드로서 DRED(디바이스 제거 확장 데이터)에서 추적한 할당에 대한 데이터를 설명합니다.
D3D12_DRED_AUTO_BREADCRUMBS_OUTPUT 연결된 D3D12_AUTO_BREADCRUMB_NODE 개체 목록의 헤드에 대한 포인터를 포함합니다.
D3D12_DRED_PAGE_FAULT_OUTPUT 지정된 VA(가상 주소)의 GPU 페이지 오류와 관련된 할당 데이터에 대해 설명합니다.
D3D12_DXIL_LIBRARY_DESC 상태 개체에 포함할 수 있는 DXIL 라이브러리 상태 하위 개체에 대해 설명합니다.
D3D12_DXIL_SUBOBJECT_TO_EXPORTS_ASSOCIATION 이 하위 개체는 현재 릴리스에서 지원되지 않습니다.
D3D12_EXISTING_COLLECTION_DESC 상태 개체에 포함할 수 있는 기존 컬렉션을 설명하는 상태 하위 개체입니다.
D3D12_EXPORT_DESC DXIL 라이브러리 또는 컬렉션 상태 개체와 같은 상태 하위 개체에서 내보내기를 설명합니다.
D3D12_FEATURE_DATA_ARCHITECTURE 애플리케이션이 특정 어댑터 속성에 더 잘 최적화할 수 있도록 어댑터 아키텍처에 대한 세부 정보를 제공합니다.
D3D12_FEATURE_DATA_ARCHITECTURE1 애플리케이션이 특정 어댑터 속성에 더 잘 최적화할 수 있도록 각 어댑터의 아키텍처 세부 정보에 대한 세부 정보를 제공합니다.
D3D12_FEATURE_DATA_COMMAND_QUEUE_PRIORITY 다양한 명령 큐 유형의 우선 순위 지정에 대한 어댑터의 지원을 자세히 설명합니다.
D3D12_FEATURE_DATA_CROSS_NODE 여러 어댑터 간의 리소스 공유에 대한 지원 수준을 나타냅니다(예: 여러 GPU).

D3D12_FEATURE_DATA_D3D12_OPTIONS 현재 그래픽 드라이버의 Direct3D 12 기능 옵션에 대해 설명합니다.
D3D12_FEATURE_DATA_D3D12_OPTIONS1 HLSL 6.0 웨이브 작업에 대한 지원 수준을 설명합니다.
D3D12_FEATURE_DATA_D3D12_OPTIONS10 SUM 결합자를 사용할 수 있는지 여부와 메시 셰이더에서 *SV_ShadingRate* 설정할 수 있는지 여부를 나타냅니다.
D3D12_FEATURE_DATA_D3D12_OPTIONS11 설명자 힙의 리소스에서 64비트 정수 원자성 지원 여부를 나타냅니다.
D3D12_FEATURE_DATA_D3D12_OPTIONS12 향상된 장벽이 지원되는지 여부를 나타냅니다.
D3D12_FEATURE_DATA_D3D12_OPTIONS13 TBD
D3D12_FEATURE_DATA_D3D12_OPTIONS2 어댑터가 깊이 경계 테스트 및 프로그래밍 가능한 샘플 위치에 대해 제공하는 지원 수준을 나타냅니다.
D3D12_FEATURE_DATA_D3D12_OPTIONS3 어댑터가 타임스탬프 쿼리, 형식 캐스팅, 즉시 쓰기, 뷰 인스턴스화 및 barycentrics에 대해 제공하는 지원 수준을 나타냅니다.
D3D12_FEATURE_DATA_D3D12_OPTIONS4 64KB 정렬 MSAA 텍스처, 교차 API 공유 및 네이티브 16비트 셰이더 작업에 대한 지원 수준을 나타냅니다.
D3D12_FEATURE_DATA_D3D12_OPTIONS5 어댑터가 렌더링 패스, 광선 추적 및 셰이더 리소스 뷰 계층 3 타일식 리소스에 대해 제공하는 지원 수준을 나타냅니다.
D3D12_FEATURE_DATA_D3D12_OPTIONS6 어댑터가 VRS(가변 속도 음영)에 대해 제공하는 지원 수준을 나타내고 백그라운드 처리가 지원되는지 여부를 나타냅니다.

D3D12_FEATURE_DATA_D3D12_OPTIONS7 어댑터가 메시 및 증폭 셰이더 및 샘플러 피드백에 제공하는 지원 수준을 나타냅니다.
D3D12_FEATURE_DATA_D3D12_OPTIONS8 정렬되지 않은 블록 압축 텍스처가 지원되는지 여부를 나타냅니다.
D3D12_FEATURE_DATA_D3D12_OPTIONS9 메시 셰이더, 8개 이상의 *SV_RenderTargetArrayIndex* 값, 형식화된 리소스 64비트 정수 원자, 파생 및 파생 종속 텍스처 샘플 작업 및 WaveMMA(wave_matrix) 작업에 대한 지원 수준이 있는지 여부를 나타냅니다.
D3D12_FEATURE_DATA_DISPLAYABLE 이 기능은 현재 미리 보기로 제공됩니다.
D3D12_FEATURE_DATA_EXISTING_HEAPS 어댑터가 기존 시스템 메모리에서 힙 만들기를 지원하는지 여부에 대한 세부 정보를 제공합니다.
D3D12_FEATURE_DATA_FEATURE_LEVELS 현재 그래픽 드라이버에서 지원하는 기능 수준에 대한 정보를 설명합니다.
D3D12_FEATURE_DATA_FORMAT_INFO DXGI 데이터 형식 및 평면 수를 설명합니다.
D3D12_FEATURE_DATA_FORMAT_SUPPORT 지정된 형식에 대해 현재 그래픽 드라이버에서 지원하는 리소스를 설명합니다. (D3D12_FEATURE_DATA_FORMAT_SUPPORT)
D3D12_FEATURE_DATA_GPU_VIRTUAL_ADDRESS_SUPPORT 리소스당 최대 주소 비트 및 프로세스당 최대 주소 비트를 포함하여 어댑터의 GPU 가상 주소 공간 제한을 자세히 설명합니다.
D3D12_FEATURE_DATA_MULTISAMPLE_QUALITY_LEVELS 지정된 형식 및 샘플 수에 대한 다중 샘플링 이미지 품질 수준을 설명합니다.
D3D12_FEATURE_DATA_PROTECTED_RESOURCE_SESSION_SUPPORT 보호된 리소스 세션에 대한 지원 수준을 나타냅니다.

D3D12_FEATURE_DATA_PROTECTED_RESOURCE_SESSION_TYPE_COUNT 보호된 리소스 세션 유형의 수를 나타냅니다.
D3D12_FEATURE_DATA_PROTECTED_RESOURCE_SESSION_TYPES 보호된 리소스 세션 유형의 목록을 나타냅니다.
D3D12_FEATURE_DATA_QUERY_META_COMMAND 어댑터가 메타 명령에 대해 제공하는 지원 수준을 나타냅니다.
D3D12_FEATURE_DATA_ROOT_SIGNATURE 루트 서명 버전 지원을 나타냅니다.
D3D12_FEATURE_DATA_SERIALIZATION 힙 serialization에 대한 지원 수준을 나타냅니다.
D3D12_FEATURE_DATA_SHADER_CACHE 현재 그래픽 드라이버에서 지원되는 셰이더 캐싱 수준을 설명합니다. (D3D12_FEATURE_DATA_SHADER_CACHE)
D3D12_FEATURE_DATA_SHADER_MODEL 지원되는 셰이더 모델을 포함합니다.
D3D12_GLOBAL_BARRIER 리소스 메모리 액세스 장벽에 대해 설명합니다. 전역, 텍스처 및 버퍼 장벽에서 리소스 메모리를 특정 액세스 유형에 대해 표시해야 하는 시기를 나타내는 데 사용됩니다.
D3D12_GLOBAL_ROOT_SIGNATURE 연결된 셰이더와 함께 사용할 전역 루트 서명 상태 하위 개체를 정의합니다.
D3D12_GPU_DESCRIPTOR_HANDLE GPU 설명자 핸들을 설명합니다.
D3D12_GPU_VIRTUAL_ADDRESS_AND_STRIDE GPU 가상 주소 및 인덱싱 진행을 나타냅니다.
D3D12_GPU_VIRTUAL_ADDRESS_RANGE GPU 가상 주소 범위를 나타냅니다.

D3D12_GPU_VIRTUAL_ADDRESS_RANGE_AND_STRIDE GPU 가상 주소 범위 및 보폭을 나타냅니다.
D3D12_GRAPHICS_PIPELINE_STATE_DESC 그래픽 파이프라인 상태 개체를 설명합니다.
D3D12_HEAP_DESC 힙에 대해 설명합니다.
D3D12_HEAP_PROPERTIES 힙 속성에 대해 설명합니다.
D3D12_HIT_GROUP_DESC 상태 개체에 포함할 수 있는 광선 추적 적중 그룹 상태 하위 개체에 대해 설명합니다.
D3D12_INDEX_BUFFER_VIEW 볼 인덱스 버퍼에 대해 설명합니다.
D3D12_INDIRECT_ARGUMENT_DESC 명령 서명과 함께 사용할 간접 인수(간접 매개 변수)를 설명합니다.
D3D12_INPUT_ELEMENT_DESC 그래픽 파이프라인의 입력 어셈블러 단계에 대한 단일 요소를 설명합니다.
D3D12_INPUT_LAYOUT_DESC 입력 어셈블러 단계의 입력 버퍼 데이터를 설명합니다.
D3D12_LOCAL_ROOT_SIGNATURE 연결된 셰이더와 함께 사용할 로컬 루트 서명 상태 하위 개체를 정의합니다.
D3D12_MEMCPY_DEST 메모리 복사 작업의 대상을 설명합니다.
D3D12_META_COMMAND_DESC 메타 명령에 대해 설명합니다.

D3D12_META_COMMAND_PARAMETER_DESC 메타 명령에 대한 매개 변수를 설명합니다.
D3D12_MIP_REGION 밉 영역의 크기를 설명합니다.
D3D12_NODE_MASK 상태 개체가 적용되는 GPU 노드를 식별하는 상태 하위 개체입니다.
D3D12_PACKED_MIP_INFO 밉맵이 있는 타일식 리소스의 타일 구조에 대해 설명합니다. (D3D12_PACKED_MIP_INFO)
D3D12_PIPELINE_STATE_STREAM_DESC 파이프라인 상태 스트림에 대해 설명합니다.
D3D12_PLACED_SUBRESOURCE_FOOTPRINT 오프셋 및 D3D12_SUBRESOURCE_FOOTPRINT 포함하여 배치된 하위 리소스의 공간을 설명합니다.
D3D12_PROTECTED_RESOURCE_SESSION_DESC 어댑터당 보호된 리소스 세션에 대한 플래그를 설명합니다.
D3D12_PROTECTED_RESOURCE_SESSION_DESC1 어댑터당 보호된 리소스 세션에 대한 플래그 및 보호 유형을 설명합니다.
D3D12_QUERY_DATA_PIPELINE_STATISTICS BeginQuery 및 EndQuery 호출 간의 그래픽 파이프라인 활동에 대한 정보를 쿼리합니다.
D3D12_QUERY_DATA_SO_STATISTICS 스트림 출력에 대한 쿼리 데이터를 설명합니다.
D3D12_QUERY_HEAP_DESC 쿼리 힙의 용도를 설명합니다. 쿼리 힙에는 개별 쿼리의 배열이 포함됩니다.
D3D12_RANGE 메모리 범위를 설명합니다.

D3D12_RANGE_UINT64 64비트 주소 공간의 메모리 범위를 설명합니다.
D3D12_RASTERIZER_DESC 래스터라이저 상태를 설명합니다. (D3D12_RASTERIZER_DESC)
D3D12_RAYTRACING_AABB 광선 추적 기하 도형으로 사용되는 AABB(축 맞춤 경계 상자)를 나타냅니다.
D3D12_RAYTRACING_ACCELERATION_STRUCTURE_POSTBUILD_INFO_COMPACTED_SIZE_DESC 압축 후 가속 구조에 대한 공간 요구 사항을 설명합니다.
D3D12_RAYTRACING_ACCELERATION_STRUCTURE_POSTBUILD_INFO_CURRENT_SIZE_DESC 현재 가속 구조에서 사용되는 공간을 설명합니다.
D3D12_RAYTRACING_ACCELERATION_STRUCTURE_POSTBUILD_INFO_DESC 가속 구조에서 생성할 빌드 후 정보에 대한 설명입니다. EmitRaytracingAccelerationStructurePostbuildInfo 및 BuildRaytracingAccelerationStructure 호출에서 이 구조를 사용합니다.
D3D12_RAYTRACING_ACCELERATION_STRUCTURE_POSTBUILD_INFO_SERIALIZATION_DESC 직렬화된 가속 구조 및 헤더의 크기와 레이아웃을 설명합니다.
D3D12_RAYTRACING_ACCELERATION_STRUCTURE_POSTBUILD_INFO_TOOLS_VISUALIZATION_DESC 가속 구조를 도구로 시각화할 수 있는 폼으로 디코딩하기 위한 공간 요구 사항을 설명합니다.
D3D12_RAYTRACING_ACCELERATION_STRUCTURE_PREBUILD_INFO 광선 추적 가속 구조에 대한 사전 빌드 정보를 나타냅니다. GetRaytracingAccelerationStructurePrebuildInfo를 호출하여 이 구조체의 인스턴스를 가져옵니다.
D3D12_RAYTRACING_ACCELERATION_STRUCTURE_SRV 광선 추적 가속 구조를 저장하기 위한 SRV(셰이더 리소스 뷰) 구조체입니다.
D3D12_RAYTRACING_GEOMETRY_AABBS_DESC D3D12_BUILD_RAYTRACING_ACCELERATION_STRUCTURE_INPUTS 구조체에서 광선 추적 가속 구조 빌드 작업에 입력 데이터를 제공하는 데 사용되는 축 맞춤 경계 상자 집합을 설명합니다.

D3D12_RAYTRACING_GEOMETRY_DESC 광선 추적 가속 구조 빌드 작업에 입력 데이터를 제공하기 위해 D3D12_BUILD_RAYTRACING_ACCELERATION_STRUCTURE_INPUTS 구조에 사용되는 기하 도형 집합을 설명합니다.
D3D12_RAYTRACING_GEOMETRY_TRIANGLES_DESC 광선 추적 기하 도형으로 사용되는 삼각형 집합을 설명합니다. 이 구조체가 가리키는 기하 도형은 항상 삼각형 목록 형식으로 인덱싱되거나 인덱싱되지 않습니다. 삼각형 스트립은 지원되지 않습니다.
D3D12_RAYTRACING_INSTANCE_DESC 가속 구조 빌드 프로세스 중에 GPU 메모리에 사용되는 광선 추적 가속 구조의 인스턴스를 설명합니다.
D3D12_RAYTRACING_PIPELINE_CONFIG 광선 추적 파이프라인 구성을 나타내는 상태 하위 개체입니다.
D3D12_RAYTRACING_PIPELINE_CONFIG1 플래그가 있는 광선 추적 파이프라인 구성을 나타내는 상태 하위 개체입니다.
D3D12_RAYTRACING_SHADER_CONFIG 셰이더 구성을 나타내는 상태 하위 개체입니다.
D3D12_RENDER_PASS_BEGINNING_ACCESS 렌더링 패스로 전환할 때 애플리케이션에서 요청한 리소스에 대한 액세스를 설명합니다.
D3D12_RENDER_PASS_BEGINNING_ACCESS_CLEAR_PARAMETERS 렌더링 패스의 시작 부분에서 리소스를 지워야 하는 명확한 값을 설명합니다.
D3D12_RENDER_PASS_DEPTH_STENCIL_DESC DSV(깊이 스텐실 뷰)에 대한 바인딩(렌더링 패스 기간 동안 고정됨)과 시작 및 끝 액세스 특성에 대해 설명합니다.
D3D12_RENDER_PASS_ENDING_ACCESS 렌더링 패스에서 전환할 때 애플리케이션에서 요청한 리소스에 대한 액세스를 설명합니다.
D3D12_RENDER_PASS_ENDING_ACCESS_RESOLVE_PARAMETERS 렌더링 패스가 끝날 때 확인할 리소스에 대해 설명합니다.

D3D12_RENDER_PASS_ENDING_ACCESS_RESOLVE_SUBRESOURCE_PARAMETERS 렌더링 패스가 끝날 때 해결에 관련된 하위 리소스에 대해 설명합니다.
D3D12_RENDER_PASS_RENDER_TARGET_DESC 하나 이상의 RTV(렌더링 대상 뷰)에 대한 바인딩(렌더링 패스 기간 동안 고정됨)과 시작 및 끝 액세스 특성에 대해 설명합니다.
D3D12_RENDER_TARGET_BLEND_DESC 렌더링 대상의 혼합 상태를 설명합니다. (D3D12_RENDER_TARGET_BLEND_DESC)
D3D12_RENDER_TARGET_VIEW_DESC 렌더링 대상 뷰를 사용하여 액세스할 수 있는 리소스의 하위 리소스에 대해 설명합니다.
D3D12_RESOURCE_ALIASING_BARRIER 동일한 힙에 매핑된 두 개의 서로 다른 리소스 사용 간의 전환에 대해 설명합니다.
D3D12_RESOURCE_ALLOCATION_INFO 리소스를 할당하는 데 필요한 매개 변수에 대해 설명합니다.
D3D12_RESOURCE_ALLOCATION_INFO1 오프셋을 포함하여 리소스를 할당하는 데 필요한 매개 변수에 대해 설명합니다.
D3D12_RESOURCE_BARRIER 리소스 장벽(리소스 사용 전환)을 설명합니다.
D3D12_RESOURCE_DESC 텍스처와 같은 리소스를 설명합니다. 이 구조체는 광범위하게 사용됩니다.
D3D12_RESOURCE_DESC1 밉 영역을 포함하여 텍스처와 같은 리소스에 대해 설명합니다. 이 구조체는 여러 메서드에서 사용됩니다.
D3D12_RESOURCE_TRANSITION_BARRIER 서로 다른 사용량 간의 하위 리소스 전환에 대해 설명합니다.
D3D12_RESOURCE_UAV_BARRIER 향후 UAV 액세스를 시작하기 전에 모든 UAV 액세스가 완료되어야 하는 리소스를 나타냅니다.

D3D12_ROOT_CONSTANTS 셰이더에 하나의 상수 버퍼로 표시되는 루트 서명의 상수 인라인을 설명합니다.
D3D12_ROOT_DESCRIPTOR 셰이더에 표시되는 루트 서명 버전 1.0의 설명자를 인라인으로 설명합니다.
D3D12_ROOT_DESCRIPTOR_TABLE 설명자 테이블의 루트 서명 1.0 레이아웃을 단일 기본 설명자 핸들을 기준으로 하는 설명자 범위의 컬렉션으로 설명합니다.
D3D12_ROOT_DESCRIPTOR_TABLE1 설명자 테이블의 루트 서명 1.1 레이아웃을 단일 기본 설명자 핸들을 기준으로 하는 설명자 범위의 컬렉션으로 설명합니다.
D3D12_ROOT_DESCRIPTOR1 셰이더에 표시되는 루트 서명 버전 1.1의 설명자를 인라인으로 설명합니다.
D3D12_ROOT_PARAMETER 루트 서명 버전 1.0의 슬롯에 대해 설명합니다.
D3D12_ROOT_PARAMETER1 루트 서명 버전 1.1의 슬롯에 대해 설명합니다.
D3D12_ROOT_SIGNATURE_DESC 루트 서명 버전 1.0의 레이아웃을 설명합니다.
D3D12_ROOT_SIGNATURE_DESC1 루트 서명 버전 1.1의 레이아웃을 설명합니다.
D3D12_RT_FORMAT_ARRAY 렌더링 대상 형식의 배열을 래핑합니다.
D3D12_SAMPLE_POSITION 프로그래밍 가능한 샘플 위치와 함께 사용할 하위 픽셀 샘플 위치를 설명합니다.
D3D12_SAMPLER_DESC 샘플러 상태를 설명합니다. (D3D12_SAMPLER_DESC)

D3D12_SERIALIZED_DATA_DRIVER_MATCHING_IDENTIFIER 직렬화된 가속 구조에 대한 드라이버 버전 관리를 설명하는 불투명 데이터 구조입니다.
D3D12_SERIALIZED_RAYTRACING_ACCELERATION_STRUCTURE_HEADER 직렬화된 광선 추적 가속 구조체의 헤더를 정의합니다.
D3D12_SHADER_BYTECODE 셰이더 데이터를 설명합니다. (D3D12_SHADER_BYTECODE)
D3D12_SHADER_CACHE_SESSION_DESC 셰이더 캐시 세션에 대해 설명합니다.
D3D12_SHADER_RESOURCE_VIEW_DESC 셰이더 리소스 뷰에 대해 설명합니다. (D3D12_SHADER_RESOURCE_VIEW_DESC)
D3D12_SO_DECLARATION_ENTRY 출력 슬롯의 꼭짓점 버퍼에 있는 꼭짓점 요소에 대해 설명합니다.
D3D12_STATE_OBJECT_CONFIG 상태 개체의 일반 속성을 정의합니다.
D3D12_STATE_OBJECT_DESC 상태 개체에 대한 설명입니다. 이 구조를 ID3D12Device::CreateStateObject에 전달합니다.
D3D12_STATE_SUBOBJECT 상태 개체 설명 내의 하위 개체를 나타냅니다. D3D12_STATE_OBJECT_DESC 함께 사용합니다.
D3D12_STATIC_SAMPLER_DESC 정적 샘플러에 대해 설명합니다.
D3D12_STREAM_OUTPUT_BUFFER_VIEW 스트림 출력 버퍼에 대해 설명합니다.
D3D12_STREAM_OUTPUT_DESC 스트리밍 출력 버퍼에 대해 설명합니다.

D3D12_SUBOBJECT_TO_EXPORTS_ASSOCIATION 상태 개체에 직접 정의된 하위 개체를 셰이더 내보내기와 연결합니다.
D3D12_SUBRESOURCE_DATA 하위 리소스 데이터를 설명합니다. (D3D12_SUBRESOURCE_DATA)
D3D12_SUBRESOURCE_FOOTPRINT 부모 리소스에 대한 하위 리소스의 형식, 너비, 높이, 깊이 및 행 피치를 설명합니다.
D3D12_SUBRESOURCE_INFO 하위 리소스 데이터를 설명합니다. (D3D12_SUBRESOURCE_INFO)
D3D12_SUBRESOURCE_RANGE_UINT64 하위 리소스 메모리 범위를 설명합니다.
D3D12_SUBRESOURCE_TILING 타일형 하위 리소스 볼륨을 설명합니다. (D3D12_SUBRESOURCE_TILING)
D3D12_TEX1D_ARRAY_DSV 깊이 스텐실 보기에서 사용할 1D 텍스처 배열의 하위 리소스에 대해 설명합니다.
D3D12_TEX1D_ARRAY_RTV 렌더링 대상 보기에서 사용할 1D 텍스처 배열의 하위 리소스에 대해 설명합니다.
D3D12_TEX1D_ARRAY_SRV 셰이더 리소스 뷰에서 사용할 1D 텍스처 배열의 하위 리소스에 대해 설명합니다.
D3D12_TEX1D_ARRAY_UAV 순서가 지정되지 않은 액세스 1D 텍스처 리소스의 배열을 설명합니다. (D3D12_TEX1D_ARRAY_UAV)
D3D12_TEX1D_DSV 깊이 스텐실 보기에 액세스할 수 있는 1D 텍스처의 하위 리소스에 대해 설명합니다.
D3D12_TEX1D_RTV 렌더링 대상 보기에서 사용할 1D 텍스처의 하위 리소스에 대해 설명합니다.

D3D12_TEX1D_SRV 셰이더 리소스 뷰에서 사용할 1D 텍스처의 하위 리소스를 지정합니다. (D3D12_TEX1D_SRV)
D3D12_TEX1D_UAV 순서가 지정되지 않은 액세스 1D 텍스처 리소스에 대해 설명합니다. (D3D12_TEX1D_UAV)
D3D12_TEX2D_ARRAY_DSV 깊이 스텐실 보기에 액세스할 수 있는 2D 텍스처 배열의 하위 리소스에 대해 설명합니다.
D3D12_TEX2D_ARRAY_RTV 렌더링 대상 보기에서 사용할 2D 텍스처 배열의 하위 리소스에 대해 설명합니다. (D3D12_TEX2D_ARRAY_RTV)
D3D12_TEX2D_ARRAY_SRV 셰이더 리소스 뷰에서 사용할 2D 텍스처 배열의 하위 리소스에 대해 설명합니다. (D3D12_TEX2D_ARRAY_SRV)
D3D12_TEX2D_ARRAY_UAV 순서가 지정되지 않은 액세스 2D 텍스처 리소스의 배열을 설명합니다. (D3D12_TEX2D_ARRAY_UAV)
D3D12_TEX2D_DSV 깊이 스텐실 보기에 액세스할 수 있는 2D 텍스처의 하위 리소스를 설명합니다.
D3D12_TEX2D_RTV 렌더링 대상 보기에서 사용할 2D 텍스처의 하위 리소스에 대해 설명합니다. (D3D12_TEX2D_RTV)
D3D12_TEX2D_SRV 셰이더 리소스 뷰에서 사용할 2D 텍스처의 하위 리소스에 대해 설명합니다. (D3D12_TEX2D_SRV)
D3D12_TEX2D_UAV 순서가 지정되지 않은 액세스 2D 텍스처 리소스에 대해 설명합니다. (D3D12_TEX2D_UAV)
D3D12_TEX2DMS_ARRAY_DSV 깊이 스텐실 보기에 대한 다중 샘플링된 2D 텍스처 배열의 하위 리소스에 대해 설명합니다.
D3D12_TEX2DMS_ARRAY_RTV 렌더링 대상 보기에서 사용할 다중 샘플링된 2D 텍스처 배열의 하위 리소스에 대해 설명합니다.

D3D12_TEX2DMS_ARRAY_SRV 셰이더 리소스 뷰에서 사용할 다중 샘플링된 2D 텍스처 배열의 하위 리소스에 대해 설명합니다.
D3D12_TEX2DMS_DSV 깊이 스텐실 보기에 액세스할 수 있는 다중 샘플링된 2D 텍스처의 하위 리소스에 대해 설명합니다.
D3D12_TEX2DMS_RTV 렌더링 대상 보기에서 사용할 다중 샘플링된 2D 텍스처의 하위 리소스를 설명합니다.
D3D12_TEX2DMS_SRV 셰이더 리소스 뷰에서 사용할 다중 샘플링된 2D 텍스처의 하위 리소스에 대해 설명합니다.
D3D12_TEX3D_RTV 렌더링 대상 보기에서 사용할 3D 텍스처의 하위 리소스에 대해 설명합니다.
D3D12_TEX3D_SRV 셰이더 리소스 뷰에서 사용할 3D 텍스처의 하위 리소스에 대해 설명합니다.
D3D12_TEX3D_UAV 순서가 지정되지 않은 액세스 3D 텍스처 리소스에 대해 설명합니다. (D3D12_TEX3D_UAV)
D3D12_TEXCUBE_ARRAY_SRV 셰이더 리소스 뷰에서 사용할 큐브 텍스처 배열의 하위 리소스에 대해 설명합니다.
D3D12_TEXCUBE_SRV 셰이더 리소스 뷰에서 사용할 큐브 텍스처의 하위 리소스에 대해 설명합니다.
D3D12_TEXTURE_BARRIER 텍스처에 대한 액세스 전환을 표현합니다.
D3D12_TEXTURE_COPY_LOCATION 텍스처 복사본을 위해 텍스처의 일부를 설명합니다.
D3D12_TILE_REGION_SIZE 타일형 영역의 크기를 설명합니다. (D3D12_TILE_REGION_SIZE)

D3D12_TILE_SHAPE 크기를 지정하여 타일의 모양을 설명합니다. (D3D12_TILE_SHAPE)
D3D12_TILED_RESOURCE_COORDINATE 타일형 리소스의 좌표를 설명합니다. (D3D12_TILED_RESOURCE_COORDINATE)
D3D12_UNORDERED_ACCESS_VIEW_DESC 순서가 지정되지 않은 액세스 보기를 사용하여 액세스할 수 있는 리소스의 하위 리소스에 대해 설명합니다.
D3D12_VERSIONED_DEVICE_REMOVED_EXTENDED_DATA 버전이 지정된 DRED(디바이스 제거 확장 데이터) 데이터를 나타냅니다.
D3D12_VERSIONED_ROOT_SIGNATURE_DESC 루트 서명 설명의 모든 버전을 보유하며 serialization/deserialization 함수와 함께 사용하도록 설계되었습니다.
D3D12_VERTEX_BUFFER_VIEW 꼭짓점 버퍼 뷰에 대해 설명합니다.
D3D12_VIEW_INSTANCE_LOCATION 뷰 인스턴스와 연결된 뷰포트/스텐실 및 렌더링 대상을 지정합니다.
D3D12_VIEW_INSTANCING_DESC 뷰 인스턴스화 구성 중에 사용되는 매개 변수를 지정합니다.
D3D12_VIEWPORT 뷰포트의 차원에 대해 설명합니다.
D3D12_WRITEBUFFERIMMEDIATE_PARAMETER ID3D12CommandList2::WriteBufferImmediate를 사용하여 작성된 즉각적인 값 및 대상 주소를 지정합니다.

열거형

D3D_ROOT_SIGNATURE_VERSION 루트 서명 레이아웃의 버전을 지정합니다.
D3D_SHADER_MODEL 셰이더 모델을 지정합니다.
D3D12_AUTO_BREADCRUMB_OP 렌더링/컴퓨팅 GPU 작업을 지정하는 상수를 정의합니다. (D3D12_AUTO_BREADCRUMB_OP)
D3D12_AXIS_SHADING_RATE 가로 또는 세로 축을 따라 음영 속도(가변 속도 음영 또는 VRS)를 지정하는 상수를 정의합니다.
D3D12_BACKGROUND_PROCESSING_MODE 이후에 제출되는 GPU 작업에 적용할 동적 최적화 수준을 지정하는 상수를 정의합니다.
D3D12_BLEND 픽셀 셰이더 및 렌더링 대상에 대한 값을 변조하는 혼합 요소를 지정합니다.
D3D12_BLEND_OP RGB 또는 알파 혼합 작업을 지정합니다.
D3D12_BUFFER_SRV_FLAGS 버퍼 리소스를 보는 방법을 식별합니다. (D3D12_BUFFER_SRV_FLAGS)
D3D12_BUFFER_UAV_FLAGS 버퍼 리소스에 대한 순서가 지정되지 않은 액세스 보기 옵션을 식별합니다. (D3D12_BUFFER_UAV_FLAGS)
D3D12_CLEAR_FLAGS 깊이 스텐실 보기에서 지울 내용을 지정합니다.
D3D12_COLOR_WRITE_ENABLE 혼합하는 동안 쓰기 가능한 렌더링 대상의 각 픽셀의 구성 요소를 식별합니다.
D3D12_COMMAND_LIST_FLAGS D3D12_COMMAND_LIST_FLAGS 열거형은 명령 목록을 만들 때 사용할 플래그를 지정합니다.

D3D12_COMMAND_LIST_SUPPORT_FLAGS 다양한 작업을 지원할 수 있는 명령 목록의 종류를 결정하는 데 사용됩니다.
D3D12_COMMAND_LIST_TYPE 명령 목록의 형식을 지정합니다.
D3D12_COMMAND_QUEUE_FLAGS 명령 큐를 만들 때 사용할 플래그를 지정합니다.
D3D12_COMMAND_QUEUE_PRIORITY 명령 큐의 우선 순위 수준을 정의합니다.
D3D12_COMPARISON_FUNC 비교 옵션을 지정합니다.
D3D12_CONSERVATIVE_RASTERIZATION_MODE 보수적 래스터화가 켜지거나 꺼져 있는지 여부를 식별합니다. (D3D12_CONSERVATIVE_RASTERIZATION_MODE)
D3D12_CONSERVATIVE_RASTERIZATION_TIER 보수적 래스터화의 계층 수준을 식별합니다.
D3D12_CPU_PAGE_PROPERTY 힙의 CPU 페이지 속성을 지정합니다.
D3D12_CROSS_NODE_SHARING_TIER 계층 1 에뮬레이트, 계층 1 또는 계층 2와 같은 어댑터의 노드 간에 공유 수준을 지정합니다.
D3D12_CULL_MODE 특정 방향을 향한 삼각형이 그려지지 않도록 지정합니다.
D3D12_DEPTH_WRITE_MASK 깊이 데이터를 작성하기 위한 깊이 스텐실 버퍼의 부분을 식별합니다.
D3D12_DESCRIPTOR_HEAP_FLAGS 힙에 대한 옵션을 지정합니다.

D3D12_DESCRIPTOR_HEAP_TYPE 설명자 힙의 형식을 지정합니다.
D3D12_DESCRIPTOR_RANGE_FLAGS 설명자와 루트 서명 1.1 설명에서 참조하는 데이터의 변동성을 지정하여 일부 드라이버 최적화를 사용하도록 설정할 수 있습니다.
D3D12_DESCRIPTOR_RANGE_TYPE 예를 들어 설명자 테이블의 일부에 100개의 SRV(셰이더 리소스 뷰)가 있는 경우 해당 범위를 100이 아닌 하나의 항목에서 선언할 수 있도록 범위를 지정합니다.
D3D12_DRED_ALLOCATION_TYPE 3D12DDI_HANDLETYPE 열거형 값과 일치하고 숫자와 동일합니다.
D3D12_DRED_ENABLEMENT 렌더링/컴퓨팅 GPU 작업을 지정하는 상수를 정의합니다. (D3D12_DRED_ENABLEMENT)
D3D12_DRED_FLAGS D3D12_DEVICE_REMOVED_EXTENDED_DATA 구조에서 Direct3D 런타임에 대한 컨트롤 플래그를 지정하는 데 사용되는 상수를 정의합니다.
D3D12_DRED_VERSION D3D12_VERSIONED_DEVICE_REMOVED_EXTENDED_DATA 구조에서 사용하는 DRED(Device Removed Extended Data)의 버전을 지정하는 상수를 정의합니다.
D3D12_DRIVER_MATCHING_IDENTIFIER_STATUS 직렬화된 데이터가 현재 디바이스 및 드라이버 버전과 호환되는지 여부를 쿼리하는 ID3D12Device5::CheckDriverMatchingIdentifier 호출의 결과를 지정합니다.
D3D12_DSV_DIMENSION 깊이 스텐실 보기에서 사용되는 리소스에 액세스하는 방법을 지정합니다. (D3D12_DSV_DIMENSION)
D3D12_DSV_FLAGS 깊이 스텐실 보기 옵션을 지정합니다.
D3D12_ELEMENTS_LAYOUT 요소의 위치를 식별하는 방법을 설명합니다.

D3D12_EXPORT_FLAGS 상태 하위 개체에서 기호를 내보낼 때 적용할 플래그입니다.
D3D12_FEATURE 쿼리할 Direct3D 12 기능 또는 기능 집합을 지정하는 상수를 정의합니다.
D3D12_FENCE_FLAGS 펜스 옵션을 지정합니다. (D3D12_FENCE_FLAGS)
D3D12_FILL_MODE 삼각형을 렌더링할 때 사용할 채우기 모드를 지정합니다.
D3D12_FILTER 텍스처 샘플링 중 필터링 옵션을 지정합니다.
D3D12_FILTER_REDUCTION_TYPE 필터 감소 유형을 지정합니다.
D3D12_FILTER_TYPE 배율 또는 축소 샘플러 필터의 유형을 지정합니다.
D3D12_FORMAT_SUPPORT1 제공된 형식에 대해 지원되는 리소스를 지정합니다.
D3D12_FORMAT_SUPPORT2 제공된 형식에 대해 지원되는 순서가 지정되지 않은 리소스 옵션을 지정합니다.
D3D12_GRAPHICS_STATES 그래픽 명령 목록과 관련된 상태를 지정하는 플래그를 정의합니다. 값은 비트 OR를 함께 사용할 수 있습니다.
D3D12_HEAP_FLAGS 힙에 텍스처를 포함할 수 있는지 여부 및 리소스가 어댑터 간에 공유되는지 여부와 같은 힙 옵션을 지정합니다.
D3D12_HEAP_SERIALIZATION_TIER 힙 serialization 지원을 지정하는 상수를 정의합니다.

D3D12_HEAP_TYPE 힙의 형식을 지정합니다. 상주하는 경우 힙은 특정 CPU 캐시 속성이 있는 특정 실제 메모리 풀에 상주합니다.
D3D12_HIT_GROUP_TYPE 광선 추적 적중 그룹 상태 하위 개체의 유형을 지정합니다. 이 열거형의 값을 D3D12_HIT_GROUP_DESC 구조체와 함께 사용합니다.
D3D12_INDEX_BUFFER_STRIP_CUT_VALUE 삼각형 스트립 기본 토폴로지를 사용하는 경우 꼭짓점 위치는 연속 삼각형 "스트립"의 꼭짓점으로 해석됩니다.
D3D12_INDIRECT_ARGUMENT_TYPE 간접 매개 변수의 형식을 지정합니다.
D3D12_INPUT_CLASSIFICATION 입력 슬롯에 포함된 데이터의 형식을 식별합니다.
D3D12_LIFETIME_STATE 수명 추적 개체의 수명 상태를 지정하는 상수를 정의합니다.
D3D12_LOGIC_OP 렌더링 대상에 대해 구성할 논리 작업을 지정합니다. (D3D12_LOGIC_OP)
D3D12_MEASUREMENTS_ACTION 이전 워크로드 계측의 결과로 수행해야 하는 작업을 지정하는 상수를 정의합니다.
D3D12_MEMORY_POOL 힙의 메모리 풀을 지정합니다.
D3D12_MESH_SHADER_TIER 메시 및 증폭 셰이더 지원을 지정하는 상수를 정의합니다.
D3D12_META_COMMAND_PARAMETER_FLAGS 메타 명령에 대한 매개 변수의 플래그를 지정하는 상수를 정의합니다. 값은 비트 OR를 함께 사용할 수 있습니다.

D3D12_META_COMMAND_PARAMETER_STAGE 메타 명령에 대한 매개 변수의 단계를 지정하는 상수를 정의합니다.
D3D12_META_COMMAND_PARAMETER_TYPE 메타 명령에 대한 매개 변수의 데이터 형식을 지정하는 상수를 정의합니다.
D3D12_MULTIPLE_FENCE_WAIT_FLAGS 여러 펜스에 대한 여러 대기 플래그를 지정합니다.
D3D12_MULTISAMPLE_QUALITY_LEVEL_FLAGS 품질 수준을 결정하는 옵션을 지정합니다.
D3D12_PIPELINE_STATE_FLAGS 파이프라인 상태를 제어하는 플래그입니다.
D3D12_PIPELINE_STATE_SUBOBJECT_TYPE 파이프라인 상태 스트림 설명에서 하위 개체의 형식을 지정합니다.
D3D12_PREDICATION_OP 적용할 조건자 작업을 지정합니다.
D3D12_PRIMITIVE_TOPOLOGY_TYPE 파이프라인이 기하 도형 또는 헐 셰이더 입력 기본 형식을 해석하는 방법을 지정합니다.
D3D12_PROGRAMMABLE_SAMPLE_POSITIONS_TIER 어댑터에서 제공하는 프로그래밍 가능한 샘플 위치에 대한 지원 수준을 지정합니다.
D3D12_PROTECTED_RESOURCE_SESSION_FLAGS 보호된 리소스 세션 플래그를 지정하는 상수를 정의합니다.
D3D12_PROTECTED_RESOURCE_SESSION_SUPPORT_FLAGS 보호된 리소스 세션 지원을 지정하는 상수를 정의합니다.
D3D12_PROTECTED_SESSION_STATUS 보호된 세션 상태를 지정하는 상수를 정의합니다.

D3D12_QUERY_HEAP_TYPE 만들 쿼리 힙의 형식을 지정합니다.
D3D12_QUERY_TYPE 쿼리 유형을 지정합니다.
D3D12_RAY_FLAGS 투명도, 컬링 및 조기 실행 동작을 재정의하기 위해 TraceRay 함수에 전달된 플래그입니다.
D3D12_RAYTRACING_ACCELERATION_STRUCTURE_BUILD_FLAGS 광선 추적 가속 구조의 빌드에 대한 플래그를 지정합니다. 가속 구조 빌드 작업에 대한 입력을 제공하는 D3D12_BUILD_RAYTRACING_ACCELERATION_STRUCTURE_INPUTS 구조와 함께 이 열거형의 값을 사용합니다.
D3D12_RAYTRACING_ACCELERATION_STRUCTURE_COPY_MODE CopyRaytracingAccelerationStructure를 호출할 때 수행되는 복사 작업의 유형을 지정합니다.
D3D12_RAYTRACING_ACCELERATION_STRUCTURE_POSTBUILD_INFO_TYPE EmitRaytracingAccelerationStructurePostbuildInfo 및 BuildRaytracingAccelerationStructure를 호출하여 검색할 수 있는 가속 구조 사후 빌드 정보의 유형을 지정합니다.
D3D12_RAYTRACING_ACCELERATION_STRUCTURE_TYPE 광선 추적 가속 구조체의 형식을 지정합니다.
D3D12_RAYTRACING_GEOMETRY_FLAGS D3D12_RAYTRACING_GEOMETRY_DESC 구조체에서 광선 추적 기하 도형에 대한 플래그를 지정합니다.
D3D12_RAYTRACING_GEOMETRY_TYPE 광선 추적에 사용되는 기하 도형의 유형을 지정합니다. 이 열거형의 값을 사용하여 D3D12_RAYTRACING_GEOMETRY_DESC 기하 도형 형식을 지정합니다.
D3D12_RAYTRACING_INSTANCE_FLAGS 광선 추적 가속 구조 인스턴스에 대한 플래그입니다. 이러한 플래그는 개별 인스턴스에 대한 D3D12_RAYTRACING_GEOMETRY_FLAGS 재정의하는 데 사용할 수 있습니다.
D3D12_RAYTRACING_PIPELINE_FLAGS 광선 추적 파이프라인에 대한 구성 플래그를 지정하는 상수를 정의합니다.

D3D12_RAYTRACING_TIER 그래픽 디바이스에서 광선 추적 지원 수준을 지정합니다.
D3D12_RENDER_PASS_BEGINNING_ACCESS_TYPE 렌더링 패스로 전환할 때 애플리케이션이 지정된 리소스에 부여되는 액세스 유형을 지정합니다.
D3D12_RENDER_PASS_ENDING_ACCESS_TYPE 렌더링 패스에서 전환할 때 애플리케이션이 지정된 리소스에 부여되는 액세스 유형을 지정합니다.
D3D12_RENDER_PASS_FLAGS 렌더링 패스의 특성을 지정합니다. 예를 들어 일시 중단 또는 다시 시작할 렌더링 패스인지 여부입니다.
D3D12_RENDER_PASS_TIER 그래픽 디바이스에서 렌더링 패스에 대한 지원 수준을 지정합니다.
D3D12_RESIDENCY_FLAGS EnqueuMakeResident 함수와 함께 사용하여 메모리 예산을 초과할 때 상주 작업이 진행되는 방식을 선택합니다.
D3D12_RESIDENCY_PRIORITY 애플리케이션 우선 순위 체계를 신속하게 설정하는 데 유용한 광범위한 상주 우선 순위 버킷을 지정합니다.
D3D12_RESOLVE_MODE 확인 작업을 지정합니다.
D3D12_RESOURCE_BARRIER_FLAGS 분할 리소스 장벽을 설정하기 위한 플래그입니다.
D3D12_RESOURCE_BARRIER_TYPE 리소스 장벽 유형(리소스 사용의 전환) 설명을 지정합니다.
D3D12_RESOURCE_BINDING_TIER 사용 중인 리소스 바인딩의 계층을 식별합니다.

D3D12_RESOURCE_DIMENSION 사용 중인 리소스의 유형을 식별합니다. (D3D12_RESOURCE_DIMENSION)
D3D12_RESOURCE_FLAGS 리소스 작업에 대한 옵션을 지정합니다.
D3D12_RESOURCE_HEAP_TIER 하드웨어 및 드라이버에서 지원하는 리소스 힙 계층을 지정합니다.
D3D12_RESOURCE_STATES 리소스 사용 방법에 대한 리소스의 상태를 지정하는 상수를 정의합니다.
D3D12_ROOT_DESCRIPTOR_FLAGS 일부 드라이버 최적화를 사용하도록 설정할 수 있는 루트 서명 1.1 설명에서 설명자가 참조하는 데이터의 변동성을 지정합니다.
D3D12_ROOT_PARAMETER_TYPE 루트 서명 슬롯의 형식을 지정합니다.
D3D12_ROOT_SIGNATURE_FLAGS 루트 서명 레이아웃에 대한 옵션을 지정합니다.
D3D12_RTV_DIMENSION 렌더링 대상으로 볼 리소스의 유형을 식별합니다.
D3D12_SAMPLER_FEEDBACK_TIER 샘플러 피드백 지원을 지정하는 상수를 정의합니다.
D3D12_SERIALIZED_DATA_TYPE 직렬화된 데이터의 형식을 지정합니다. ID3D12Device5::CheckDriverMatchingIdentifier를 호출할 때 이 열거형의 값을 사용합니다.
D3D12_SHADER_CACHE_CONTROL_FLAGS 셰이더 캐시 제어 옵션을 지정하는 상수를 정의합니다.
D3D12_SHADER_CACHE_FLAGS 셰이더 캐시 플래그를 지정하는 상수를 정의합니다.

D3D12_SHADER_CACHE_KIND_FLAGS 셰이더 캐시의 종류를 지정하는 상수를 정의합니다.
D3D12_SHADER_CACHE_MODE 셰이더 캐시 모드를 지정하는 상수를 정의합니다.
D3D12_SHADER_CACHE_SUPPORT_FLAGS 현재 그래픽 드라이버의 셰이더 캐싱 지원 수준을 설명합니다. (D3D12_SHADER_CACHE_SUPPORT_FLAGS)
D3D12_SHADER_COMPONENT_MAPPING 메모리가 SRV(셰이더 리소스 뷰)에 의해 라우팅되는 방법을 지정합니다.
D3D12_SHADER_MIN_PRECISION_SUPPORT 현재 그래픽 드라이버의 셰이더에 대한 최소 정밀도 지원 옵션을 설명합니다.
D3D12_SHADER_VISIBILITY 지정된 루트 서명 슬롯의 내용에 액세스할 수 있는 셰이더를 지정합니다.
D3D12_SHADING_RATE 음영 속도를 지정하는 상수를 정의합니다(가변 속도 음영 또는 VRS의 경우).
D3D12_SHADING_RATE_COMBINER 음영 속도 결합자를 지정하는 상수를 정의합니다(가변 속도 음영 또는 VRS용).
D3D12_SHARED_RESOURCE_COMPATIBILITY_TIER API 간 공유 지원 계층을 지정하는 상수를 정의합니다.
D3D12_SRV_DIMENSION 셰이더 리소스로 볼 리소스의 유형을 식별합니다.
D3D12_STATE_OBJECT_FLAGS 상태 개체에 대한 제약 조건을 지정합니다. D3D12_STATE_OBJECT_CONFIG 구조체에서 이 열거형의 값을 사용합니다.
D3D12_STATE_OBJECT_TYPE 상태 개체의 형식을 지정합니다. D3D12_STATE_OBJECT_DESC 함께 사용합니다.

D3D12_STATE_SUBOBJECT_TYPE 상태 하위 개체의 형식입니다. D3D12_STATE_SUBOBJECT 함께 사용합니다.
D3D12_STATIC_BORDER_COLOR 정적 샘플러의 테두리 색을 지정합니다.
D3D12_STENCIL_OP 깊이 스텐실 테스트 중에 수행할 수 있는 스텐실 작업을 식별합니다.
D3D12_TEXTURE_ADDRESS_MODE 텍스처 경계 밖에 있는 텍스처 좌표를 확인하는 기술을 식별합니다.
D3D12_TEXTURE_COPY_TYPE 수행되는 텍스처 복사본 유형을 지정합니다.
D3D12_TEXTURE_LAYOUT 텍스처 레이아웃 옵션을 지정합니다. (D3D12_TEXTURE_LAYOUT)
D3D12_TILE_COPY_FLAGS 타일을 복사하는 방법을 지정합니다.
D3D12_TILE_MAPPING_FLAGS 타일 매핑 작업을 수행하는 방법을 지정합니다.
D3D12_TILE_RANGE_FLAGS 타일 매핑 범위를 지정합니다.
D3D12_TILED_RESOURCES_TIER 타일식 리소스가 지원되는 계층 수준을 식별합니다.
D3D12_TRI_STATE TBD
D3D12_UAV_DIMENSION 순서가 지정되지 않은 액세스 보기 옵션을 식별합니다.

D3D12_VARIABLE_SHADING_RATE_TIER 음영 속도 계층(가변 속도 음영 또는 VRS의 경우)을 지정하는 상수를 정의합니다.
D3D12_VIEW_INSTANCING_FLAGS 뷰 인스턴스화 옵션을 지정합니다.
D3D12_VIEW_INSTANCING_TIER 뷰 인스턴스가 지원되는 계층 수준을 나타냅니다.
D3D12_WAVE_MMA_TIER WaveMMA(wave_matrix) 작업에 대한 지원 수준을 지정하는 상수를 정의합니다.
D3D12_WRITEBUFFERIMMEDIATE_MODE WriteBufferImmediate 작업에서 사용하는 모드를 지정합니다.

d3d12sdklayers.h 헤더

아티클2023. 03. 16.

이 헤더는 Direct3D 12 그래픽에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 12 그래픽

d3d12sdklayers.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID3D12Debug 디버그 계층을 켜는 데 사용되는 인터페이스입니다.
ID3D12Debug1 GPU-Based 유효성 검사 및 종속 명령 큐 동기화를 디버그 계층에 추가합니다.
ID3D12Debug2 구성 가능한 수준의 GPU 기반 유효성 검사를 디버그 계층에 추가합니다. (ID3D12Debug2)
ID3D12Debug3 구성 가능한 수준의 GPU 기반 유효성 검사를 디버그 계층에 추가합니다. (ID3D12Debug3)
ID3D12Debug4 디버그 계층을 사용하지 않도록 설정하는 기능을 추가합니다.
ID3D12Debug5 개체의 자동 명명을 구성하는 기능을 디버그 계층에 추가합니다.
ID3D12Debug6 디버그 인터페이스는 디버그 설정을 제어합니다.
ID3D12DebugCommandList 명령 목록을 모니터링하고 디버그하는 메서드를 제공합니다.

ID3D12DebugCommandList1 이 인터페이스를 사용하면 추가 명령 목록 디버그 계층 설정을 수정할 수 있습니다.
ID3D12DebugCommandQueue 명령 큐를 모니터링하고 디버그하는 메서드를 제공합니다.
ID3D12DebugDevice 이 인터페이스는 디버깅을 위한 그래픽 디바이스를 나타냅니다.
ID3D12DebugDevice1 디바이스 전체 디버그 계층 설정을 지정합니다.
ID3D12InfoQueue 정보 큐 인터페이스는 디버그 메시지를 저장, 검색 및 필터링합니다. 큐는 메시지 큐, 선택적 스토리지 필터 스택 및 선택적 검색 필터 스택으로 구성됩니다. (ID3D12InfoQueue)
ID3D12SharingContract D3D11On12 진단 계층과 그래픽 진단 도구 간의 계약의 일부입니다.

구조체

D3D12_DEBUG_COMMAND_LIST_GPU_BASED_VALIDATION_SETTINGS GPU-Based 유효성 검사에서 사용하는 명령 목록별 설정에 대해 설명합니다.
D3D12_DEBUG_DEVICE_GPU_BASED_VALIDATION_SETTINGS GPU-Based 유효성 검사에서 사용하는 설정에 대해 설명합니다.
D3D12_DEBUG_DEVICE_GPU_SLOWDOWN_PERFORMANCE_FACTOR 성능이 낮은 그래픽 어댑터를 시뮬레이션하기 위해 디버그 디바이스에서 삽입한 인공 속도 저하의 양을 설명합니다.
D3D12_INFO_QUEUE_FILTER 디버그 메시지 필터; 에는 허용하거나 거부할 메시지 형식 목록이 포함되어 있습니다. (D3D12_INFO_QUEUE_FILTER)

D3D12_INFO_QUEUE_FILTER_DESC 필터를 통과하는 특정 유형의 메시지를 허용하거나 거부합니다. (D3D12_INFO_QUEUE_FILTER_DESC)
D3D12_MESSAGE 정보 큐의 디버그 메시지입니다. (D3D12_MESSAGE)

열거형

D3D12_DEBUG_COMMAND_LIST_PARAMETER_TYPE ID3D12DebugCommandList1::SetDebugParameter 및 ID3D12DebugCommandList1::GetDebugParameter에서 사용하는 디버그 매개 변수 형식을 나타냅니다.
D3D12_DEBUG_DEVICE_PARAMETER_TYPE ID3D12DebugDevice1::SetDebugParameter 및 ID3D12DebugDevice1::GetDebugParameter의 pData 매개 변수가 가리키는 메모리의 데이터 형식을 지정합니다.
D3D12_DEBUG_FEATURE 선택적 D3D12 디버그 계층 기능에 대한 플래그입니다.
D3D12_GPU_BASED_VALIDATION_FLAGS 런타임에 수행할 GPU 기반 유효성 검사 수준을 설명합니다.
D3D12_GPU_BASED_VALIDATION_PIPELINE_STATE_CREATE_FLAGS GPU-Based ID3D12Device::CreateGraphicsPipelineState 및 ID3D12Device::CreateComputePipelineState 중에 유효성 검사에서 패치된 파이프라인 상태를 처리하는 방법을 지정합니다.
D3D12_GPU_BASED_VALIDATION_SHADER_PATCH_MODE 디바이스 또는 명령 목록 수준에서 유효성 검사 GPU-Based 사용하는 셰이더 패치 유형을 지정합니다.
D3D12_MESSAGE_CATEGORY 디버그 메시지의 범주를 지정합니다.

D3D12_MESSAGE_ID 정보 큐 필터를 설정하기 위한 디버그 메시지 ID를 지정합니다(D3D12_INFO_QUEUE_FILTER 참조). 이러한 메시지를 사용하여 메시지 범주가 스토리지 및 검색 필터를 통과하도록 허용하거나 거부합니다.
D3D12_MESSAGE_SEVERITY 정보 큐에 대한 메시지 심각도 수준을 디버그합니다. (D3D12_MESSAGE_SEVERITY)
D3D12_RLDO_FLAGS 라이브 디바이스 개체의 수명에 대해 보고할 정보의 양에 대한 옵션을 지정합니다.

d3d12shader.h 헤더

아티클2023. 08. 23.

이 헤더는 Direct3D 12 그래픽에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 12 그래픽

d3d12shader.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID3D12FunctionParameterReflection function-parameter-reflection 인터페이스는 function-parameter 정보에 액세스합니다. (ID3D12FunctionParameterReflection)
ID3D12FunctionReflection 함수-리플렉션 인터페이스는 함수 정보에 액세스합니다. (ID3D12FunctionReflection)
ID3D12LibraryReflection 라이브러리 리플렉션 인터페이스는 라이브러리 정보에 액세스합니다. (ID3D12LibraryReflection)
ID3D12ShaderReflection 셰이더 리플렉션 인터페이스는 셰이더 정보에 액세스합니다. (ID3D12ShaderReflection)
ID3D12ShaderReflectionConstantBuffer 이 셰이더 리플렉션 인터페이스는 상수 버퍼에 대한 액세스를 제공합니다. (ID3D12ShaderReflectionConstantBuffer)
ID3D12ShaderReflectionType 이 셰이더 리플렉션 인터페이스는 변수 형식에 대한 액세스를 제공합니다. (ID3D12ShaderReflectionType)
ID3D12ShaderReflectionVariable 이 셰이더 리플렉션 인터페이스는 변수에 대한 액세스를 제공합니다. (ID3D12ShaderReflectionVariable)

구조체

D3D12_FUNCTION_DESC 함수를 설명합니다. (D3D12_FUNCTION_DESC)
D3D12_LIBRARY_DESC 라이브러리에 대해 설명합니다. (D3D12_LIBRARY_DESC)
D3D12_PARAMETER_DESC 함수 매개 변수를 설명합니다. (D3D12_PARAMETER_DESC)
D3D12_SHADER_BUFFER_DESC 셰이더 상수 버퍼에 대해 설명합니다. (D3D12_SHADER_BUFFER_DESC)
D3D12_SHADER_DESC 셰이더에 대해 설명합니다. (D3D12_SHADER_DESC)
D3D12_SHADER_INPUT_BIND_DESC 셰이더 리소스가 셰이더 입력에 바인딩되는 방법을 설명합니다. (D3D12_SHADER_INPUT_BIND_DESC)
D3D12_SHADER_TYPE_DESC 셰이더 변수 형식에 대해 설명합니다. (D3D12_SHADER_TYPE_DESC)
D3D12_SHADER_VARIABLE_DESC 셰이더 변수에 대해 설명합니다. (D3D12_SHADER_VARIABLE_DESC)
D3D12_SIGNATURE_PARAMETER_DESC 셰이더 서명을 설명합니다. (D3D12_SIGNATURE_PARAMETER_DESC)

열거형

D3D12_SHADER_VERSION_TYPE Direct3D에서 인식하는 셰이더 유형을 열거합니다. D3D12_SHADER_DESC 구조체의 Version 멤버를 인코딩하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d3d12video.h 헤더

아티클2023. 03. 09.

이 헤더는 Microsoft Media Foundation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft Media Foundation

d3d12video.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID3D12VideoDecodeCommandList 비디오 디코딩을 위한 그래픽 명령 목록을 캡슐화합니다.
ID3D12VideoDecodeCommandList1 비디오 디코딩을 위한 그래픽 명령 목록을 캡슐화합니다. 이 인터페이스는 ID3D12VideoDecodeCommandList에서 상속되며 비디오 디코딩 히스토그램에 대한 지원을 추가합니다.
ID3D12VideoDecodeCommandList2 비디오 확장 명령에 대한 지원을 추가합니다. (ID3D12VideoDecodeCommandList2)
ID3D12VideoDecodeCommandList3 비디오 디코딩을 위한 그래픽 명령 목록을 캡슐화합니다.
ID3D12VideoDecoder Direct3D 12 비디오 디코더를 나타냅니다.
ID3D12VideoDecoder1 디코딩 작업을 수행하기 위한 해상도 독립적 리소스 및 상태를 포함하는 Direct3D 12 비디오 디코더를 나타냅니다. 보호된 리소스에 대한 지원을 추가합니다.
ID3D12VideoDecoderHeap Direct3D 12 비디오 디코더 힙을 나타냅니다.
ID3D12VideoDecoderHeap1 ID3D12VideoDecoderHeap1 인터페이스(d3d12video.h)는 ID3D12VideoDecoderHeap에서 상속되며 보호된 리소스에 대한 지원을 도입합니다.
ID3D12VideoDevice 비디오 기능을 쿼리하고 비디오 디코더 및 프로세서를 인스턴스화하는 기능을 포함하여 Microsoft Direct3D 12 디바이스의 비디오 디코딩 및 처리 기능을 제공합니다.

ID3D12VideoDevice1 동작 추정에 대한 지원을 추가합니다.
ID3D12VideoDevice2 보호된 리소스에 대한 지원을 추가합니다.
ID3D12VideoDevice3 ID3D12VideoDevice 인터페이스를 확장하여 지원 비디오 인코딩 기능을 추가합니다.
ID3D12VideoEncodeCommandList 동작 추정을 포함하여 비디오 인코딩을 위한 그래픽 명령 목록을 캡슐화합니다.
ID3D12VideoEncodeCommandList1 이 인터페이스는 ID3D12VideoEncodeCommandList에서 상속되며 비디오 확장 명령에 대한 지원을 추가합니다.
ID3D12VideoEncodeCommandList2 이 인터페이스는 ID3D12VideoEncodeCommandList1에서 상속되며 비디오를 인코딩하고 인코딩 작업 메타데이터를 확인하는 메서드를 추가합니다.
ID3D12VideoEncodeCommandList3 비디오 인코딩을 위한 그래픽 명령 목록을 캡슐화합니다.
ID3D12VideoEncoder Direct3D 12 비디오 인코더를 나타냅니다.
ID3D12VideoEncoderHeap Direct3D 12 비디오 인코더 힙을 나타냅니다.
ID3D12VideoExtensionCommand 비디오 확장 명령을 나타내는 참조 개수 개체입니다.
ID3D12VideoMotionEstimator 이 인터페이스는 비디오 동작 추정 작업에 대한 컨텍스트를 유지 관리합니다.
ID3D12VideoMotionVectorHeap 예상 동작 벡터가 저장되는 힙을 나타냅니다.
ID3D12VideoProcessCommandList 비디오 처리를 위한 그래픽 명령 목록을 캡슐화합니다. (ID3D12VideoProcessCommandList)
ID3D12VideoProcessCommandList1 비디오 처리를 위한 그래픽 명령 목록을 캡슐화합니다. (ID3D12VideoProcessCommandList1)

ID3D12VideoProcessCommandList2 비디오 확장 명령에 대한 지원을 추가합니다. (ID3D12VideoProcessCommandList2)
ID3D12VideoProcessCommandList3 비디오 처리를 위한 그래픽 명령 목록을 캡슐화합니다.
ID3D12VideoProcessor 비디오 프로세서를 만든 ID3D12VideoDevice::CreateVideoProcessor 호출에 대한 매개 변수에 대한 정보를 가져오는 메서드를 제공합니다.
ID3D12VideoProcessor1 ID3D12VideoProcessor에서 상속되며 보호된 리소스에 대한 지원을 제공합니다.

구조체

D3D12_FEATURE_DATA_VIDEO_DECODE_CONVERSION_SUPPORT 지원되는 프로필 목록을 검색합니다. (D3D12_FEATURE_DATA_VIDEO_DECODE_CONVERSION_SUPPORT)
D3D12_FEATURE_DATA_VIDEO_DECODE_FORMATS 지원되는 형식 목록을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_DECODE_HISTOGRAM 지정된 기능이 D3D12_FEATURE_VIDEO_DECODE_HISTOGRAM 경우 ID3D12VideoDevice::CheckFeatureSupport 호출에 대한 데이터를 제공합니다.
D3D12_FEATURE_DATA_VIDEO_DECODE_PROFILES 지원되는 프로필 목록을 검색합니다. (D3D12_FEATURE_DATA_VIDEO_DECODE_PROFILES)
D3D12_FEATURE_DATA_VIDEO_DECODE_SUPPORT 비디오 디코딩에 대한 지원 정보를 검색합니다.
D3D12_FEATURE_DATA_VIDEO_DECODER_HEAP_SIZE 비디오 디코더 힙의 할당 크기를 설명합니다. (D3D12_FEATURE_DATA_VIDEO_DECODER_HEAP_SIZE)
D3D12_FEATURE_DATA_VIDEO_ENCODER_CODEC 지정된 코덱이 비디오 인코딩에 지원되는지 여부를 나타내는 값을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_CODEC_CONFIGURATION_SUPPORT 지정된 코덱 구성 지원 매개 변수가 제공된 HEVC 인코딩 구성에 대해 지원되는지 여부를 나타내는 값을 검색하거나 H.264 인코딩에 지원되는 구성을 검색합니다.

D3D12_FEATURE_DATA_VIDEO_ENCODER_CODEC_PICTURE_CONTROL_SUPPORT 지정된 코덱 및 프로필에 대한 그림 컨트롤 지원을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_FRAME_SUBREGION_LAYOUT_MODE 지정된 코드, 프로필 및 수준에 대해 지정된 프레임 하위Region 레이아웃 모드가 지원되는지 여부를 나타내는 값을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_HEAP_SIZE 힙 개체의 L0 및 L1 크기뿐만 아니라 비디오 인코딩에 대해 지정된 코덱이 지원되는지 여부를 나타내는 값을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_INPUT_FORMAT 지정된 코덱, 프로필 및 형식이 비디오 인코딩에 지원되는지 여부를 나타내는 값을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_INTRA_REFRESH_MODE 지정된 코덱, 프로필 및 수준에 대해 지정된 인트라 새로 고침 모드가 지원되는지 여부를 나타내는 값을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_OUTPUT_RESOLUTION 지정된 코덱에 대해 지원되는 해결 방법 목록을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_OUTPUT_RESOLUTION_RATIOS_COUNT D3D12_FEATURE_DATA_VIDEO_ENCODER_OUTPUT_RESOLUTION_RATIOS_COUNT 구조체(d3d12video.h)는 지정된 코덱에 대해 지원되는 해상도 비율 수를 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_PROFILE_LEVEL 지정된 프로필이 비디오 인코딩에 지원되는지 여부를 나타내는 값을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_RATE_CONTROL_MODE 지정된 코덱을 사용하여 비디오 인코딩에 대해 지정된 속도 제어 모드가 지원되는지 여부를 나타내는 값을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_RESOLUTION_SUPPORT_LIMITS D3D12_FEATURE_DATA_VIDEO_ENCODER_SUPPORT 구조체에 대한 비디오 인코더 해상도 지원 제한을 나타냅니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_RESOURCE_REQUIREMENTS 지정된 인코딩 구성을 사용하여 비디오 인코딩에 대한 리소스 요구 사항을 나타내는 값을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_ENCODER_SUPPORT 지정된 비디오 인코딩 기능 및 구성 값에 대한 지원을 나타내는 값을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_EXTENSION_COMMAND_COUNT 비디오 확장 명령 수를 검색합니다.

D3D12_FEATURE_DATA_VIDEO_EXTENSION_COMMAND_PARAMETER_COUNT 지정된 매개 변수 단계에 대해 지원되는 매개 변수 수를 검색합니다.
D3D12_FEATURE_DATA_VIDEO_EXTENSION_COMMAND_PARAMETERS 지정된 매개 변수 단계에 대한 비디오 확장 명령 매개 변수 목록을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_EXTENSION_COMMAND_SIZE 비디오 확장 명령의 할당 크기를 확인합니다.
D3D12_FEATURE_DATA_VIDEO_EXTENSION_COMMAND_SUPPORT 명령 정의 입력 및 출력 구조를 사용하여 비디오 확장 명령 지원을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_EXTENSION_COMMANDS 드라이버에서 비디오 확장 명령 목록을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_MOTION_ESTIMATOR 지정된 기능이 D3D12_FEATURE_VIDEO_MOTION_ESTIMATOR 경우 ID3D12VideoDevice::CheckFeatureSupport 호출에 대한 데이터를 제공합니다. 비디오 인코더의 동작 예측 기능을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_MOTION_ESTIMATOR_PROTECTED_RESOURCES 지정된 기능이 D3D12_FEATURE_VIDEO_MOTION_ESTIMATOR_PROTECTED_RESOURCES 경우 ID3D12VideoDevice::CheckFeatureSupport 호출에 대한 데이터를 제공합니다. 비디오 동작 추정에 대한 보호된 리소스 지원을 검색합니다.
D3D12_FEATURE_DATA_VIDEO_MOTION_ESTIMATOR_SIZE 비디오 동작 추정기 힙의 할당 크기를 설명합니다.
D3D12_FEATURE_DATA_VIDEO_PROCESS_MAX_INPUT_STREAMS 비디오 프로세서에서 지원하는 사용 가능한 입력 스트림의 최대 수를 검색합니다.
D3D12_FEATURE_DATA_VIDEO_PROCESS_REFERENCE_INFO 지정된 디인터레이스 모드, 필터, 속도 변환 또는 자동 처리 기능에 필요한 과거 및 이후 참조 프레임 수를 검색합니다.
D3D12_FEATURE_DATA_VIDEO_PROCESS_SUPPORT 지정된 기능이 D3D12_FEATURE_VIDEO_PROCESS_SUPPORT 경우 ID3D12VideoDevice::CheckFeatureSupport 호출에 대한 데이터를 제공합니다.
D3D12_FEATURE_DATA_VIDEO_PROCESSOR_SIZE 비디오 디코더 힙의 할당 크기를 설명합니다. (D3D12_FEATURE_DATA_VIDEO_PROCESSOR_SIZE)
D3D12_QUERY_DATA_VIDEO_DECODE_STATISTICS ID3D12VideoDecodeCommandList::EndQuery를 호출하여 호출된 비디오 디코딩 통계 쿼리의 데이터를 나타냅니다.

D3D12_RESOLVE_VIDEO_MOTION_VECTOR_HEAP_INPUT ID3D12VideoEncodeCommandList::ResolveMotionVectorHeap 호출에 대한 입력 데이터를 제공합니다.
D3D12_RESOLVE_VIDEO_MOTION_VECTOR_HEAP_OUTPUT ID3D12VideoEncodeCommandList::ResolveMotionVectorHeap에 대한 호출에서 출력 데이터를 받습니다.
D3D12_RESOURCE_COORDINATE 리소스의 좌표를 설명합니다.
D3D12_VIDEO_DECODE_COMPRESSED_BITSTREAM 비디오가 디코딩되는 압축된 비트스트림을 나타냅니다.
D3D12_VIDEO_DECODE_CONFIGURATION 비디오 디코더의 구성에 대해 설명합니다.
D3D12_VIDEO_DECODE_CONVERSION_ARGUMENTS 출력 변환 디코딩을 위한 매개 변수를 지정합니다. (D3D12_VIDEO_DECODE_CONVERSION_ARGUMENTS)
D3D12_VIDEO_DECODE_CONVERSION_ARGUMENTS1 출력 변환 디코딩을 위한 매개 변수를 지정합니다. (D3D12_VIDEO_DECODE_CONVERSION_ARGUMENTS1)
D3D12_VIDEO_DECODE_FRAME_ARGUMENT 프레임의 디코딩 매개 변수를 나타냅니다.
D3D12_VIDEO_DECODE_INPUT_STREAM_ARGUMENTS 비디오 디코딩 작업의 입력 스트림에 대한 매개 변수를 지정합니다.
D3D12_VIDEO_DECODE_OUTPUT_HISTOGRAM 단일 구성 요소에 대한 히스토그램 출력 버퍼를 나타냅니다.
D3D12_VIDEO_DECODE_OUTPUT_STREAM_ARGUMENTS 비디오 디코딩 작업의 출력 스트림에 대한 매개 변수를 지정합니다. (D3D12_VIDEO_DECODE_OUTPUT_STREAM_ARGUMENTS)
D3D12_VIDEO_DECODE_OUTPUT_STREAM_ARGUMENTS1 비디오 디코딩 작업의 출력 스트림에 대한 매개 변수를 지정합니다. (D3D12_VIDEO_DECODE_OUTPUT_STREAM_ARGUMENTS1)
D3D12_VIDEO_DECODE_REFERENCE_FRAMES 현재 디코딩 작업에 대한 참조 프레임 목록을 포함합니다.

D3D12_VIDEO_DECODER_DESC ID3D12VideoDecoder에 대해 설명합니다.
D3D12_VIDEO_DECODER_HEAP_DESC ID3D12VideoDecoderHeap에 대해 설명합니다.
D3D12_VIDEO_ENCODE_REFERENCE_FRAMES 인코딩 작업에 대한 재구성된 참조 이미지를 나타냅니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION 비디오 인코딩을 위한 코덱 구성 구조를 나타냅니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_H264 H.264 인코딩에 대한 코덱 구성을 나타냅니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_HEVC HEVC 인코딩에 대한 코덱 구성을 나타냅니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_SUPPORT 비디오 인코딩을 위한 코덱 구성 지원 구조를 나타냅니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_SUPPORT_H264 H.264 인코딩에 대한 인코더 코덱 구성 지원을 나타냅니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_SUPPORT_HEVC HEVC 인코딩에 대한 인코더 코덱 구성 지원을 나타냅니다.
D3D12_VIDEO_ENCODER_CODEC_PICTURE_CONTROL_SUPPORT 여러 코덱에 대한 그림 컨트롤 지원 구조를 나타냅니다.
D3D12_VIDEO_ENCODER_CODEC_PICTURE_CONTROL_SUPPORT_H264 H.264 비디오 인코딩에 대한 그림 컨트롤 지원 설정을 나타냅니다.
D3D12_VIDEO_ENCODER_CODEC_PICTURE_CONTROL_SUPPORT_HEVC HEVC 비디오 인코딩에 대한 그림 컨트롤 지원 설정을 나타냅니다.
D3D12_VIDEO_ENCODER_COMPRESSED_BITSTREAM 인코딩 작업에 대해 압축된 비트 스트림 출력을 캡슐화합니다.
D3D12_VIDEO_ENCODER_DESC ID3D12VideoEncoder에 대해 설명합니다.

D3D12_VIDEO_ENCODER_ENCODE_OPERATION_METADATA_BUFFER ID3D12VideoEncodeCommandList2::EncodeFrame 작업에 대한 메타데이터를 포함하는 버퍼를 나타냅니다.
D3D12_VIDEO_ENCODER_ENCODEFRAME_INPUT_ARGUMENTS ID3D12VideoEncodeCommandList2::EncodeFrame에 대한 입력 인수를 나타냅니다.
D3D12_VIDEO_ENCODER_ENCODEFRAME_OUTPUT_ARGUMENTS ID3D12VideoEncodeCommandList2::EncodeFrame에 대한 출력 인수를 나타냅니다.
D3D12_VIDEO_ENCODER_FRAME_SUBREGION_METADATA 비디오 인코더 프레임 하위 리소스 메타데이터를 나타냅니다.
D3D12_VIDEO_ENCODER_HEAP_DESC ID3D12VideoEncoderHeap에 대해 설명합니다.
D3D12_VIDEO_ENCODER_INTRA_REFRESH 비디오 인코딩에 대한 새로 고침 내부 설정을 나타냅니다.
D3D12_VIDEO_ENCODER_LEVEL_SETTING 비디오 인코더 수준 설정을 나타냅니다.
D3D12_VIDEO_ENCODER_LEVEL_TIER_CONSTRAINTS_HEVC HEVC(고효율 비디오 코딩) 수준 설정 구성을 위한 수준 및 계층을 연결합니다.
D3D12_VIDEO_ENCODER_OUTPUT_METADATA ID3D12VideoEncodeCommandList2::EncodeFrame 작업에 대한 메타데이터를 나타냅니다.
D3D12_VIDEO_ENCODER_OUTPUT_METADATA_STATISTICS ID3D12VideoEncodeCommandList2::EncodeFrame 작업에 대한 인코딩 통계를 나타냅니다.
D3D12_VIDEO_ENCODER_PICTURE_CONTROL_CODEC_DATA 여러 코덱에 대해 연결된 EncodeFrame 명령에 대한 그림 수준 컨트롤 요소를 나타냅니다.
D3D12_VIDEO_ENCODER_PICTURE_CONTROL_CODEC_DATA_H264 H.264 인코딩에 연결된 EncodeFrame 명령에 대한 그림 수준 컨트롤 요소를 나타냅니다.
D3D12_VIDEO_ENCODER_PICTURE_CONTROL_CODEC_DATA_H264_REFERENCE_PICTURE_LIST_MODIFICATION_OPERATION H264 비디오 인코딩에 대한 그림 목록 수정 작업을 나타냅니다.
D3D12_VIDEO_ENCODER_PICTURE_CONTROL_CODEC_DATA_H264_REFERENCE_PICTURE_MARKING_OPERATION 참조 그림의 변경 내용을 작업에 필요한 작업 식별자 및 관련 매개 변수의 튜플로 메모리 작업으로 설명합니다.

D3D12_VIDEO_ENCODER_PICTURE_CONTROL_CODEC_DATA_HEVC HEVC 인코딩에 연결된 EncodeFrame 명령에 대한 그림 수준 컨트롤 요소를 나타냅니다.
D3D12_VIDEO_ENCODER_PICTURE_CONTROL_DESC 06/30/2021
D3D12_VIDEO_ENCODER_PICTURE_CONTROL_SUBREGIONS_LAYOUT_DATA 그림 컨트롤 하위 영역을 여러 코덱에 대한 조각으로 정의합니다.
D3D12_VIDEO_ENCODER_PICTURE_CONTROL_SUBREGIONS_LAYOUT_DATA_SLICES 하위 리소스를 이 분할 모드를 지원하는 코덱의 조각으로 정의합니다.
D3D12_VIDEO_ENCODER_PICTURE_RESOLUTION_DESC 비디오 인코더 그림 해상도를 정의합니다.
D3D12_VIDEO_ENCODER_PICTURE_RESOLUTION_RATIO_DESC 해상도 비율을 돌이킬 수 없는 분수로 정의합니다.
D3D12_VIDEO_ENCODER_PROFILE_DESC 인코더 프로필을 설명합니다.
D3D12_VIDEO_ENCODER_RATE_CONTROL 비디오 인코더 속도 제어 구성을 나타냅니다.
D3D12_VIDEO_ENCODER_RATE_CONTROL_CBR 상수 비트 전송률 모드에 대한 속도 제어 구조 정의를 나타냅니다.
D3D12_VIDEO_ENCODER_RATE_CONTROL_CONFIGURATION_PARAMS D3D12_VIDEO_ENCODER_RATE_CONTROL 구조체에 대한 비디오 인코더 속도 제어 구조 정의를 나타냅니다.
D3D12_VIDEO_ENCODER_RATE_CONTROL_CQP 상수 양자화 매개 변수 모드에 대한 속도 제어 구조 정의를 나타냅니다.
D3D12_VIDEO_ENCODER_RATE_CONTROL_QVBR 비트 전송률이 제한된 상수 품질 대상에 대한 속도 제어 구조 정의를 나타냅니다.
D3D12_VIDEO_ENCODER_RATE_CONTROL_VBR 가변 비트 전송률 모드에 대한 속도 제어 구조 정의를 나타냅니다.
D3D12_VIDEO_ENCODER_RECONSTRUCTED_PICTURE 인코딩 작업에 전달된 입력 프레임에서 생성된 재구성된 그림을 나타냅니다.

D3D12_VIDEO_ENCODER_REFERENCE_PICTURE_DESCRIPTOR_H264 H.264 비디오 인코딩에 대한 참조 그림 설명자를 나타냅니다.
D3D12_VIDEO_ENCODER_REFERENCE_PICTURE_DESCRIPTOR_HEVC HEVC 비디오 인코딩에 대한 참조 그림 설명자를 나타냅니다.
D3D12_VIDEO_ENCODER_RESOLVE_METADATA_INPUT_ARGUMENTS ID3D12VideoEncodeCommandList2::ResolveEncoderOutputMetadata 호출에 대한 입력 인수를 나타냅니다.
D3D12_VIDEO_ENCODER_RESOLVE_METADATA_OUTPUT_ARGUMENTS ID3D12VideoEncodeCommandList2::ResolveEncoderOutputMetadata 호출에 대한 출력 인수를 나타냅니다.
D3D12_VIDEO_ENCODER_SEQUENCE_CONTROL_DESC D3D12_VIDEO_ENCODER_SEQUENCE_CONTROL_DESC 시퀀스 수준에서 비디오 인코딩 작업의 입력에 대한 매개 변수를 나타냅니다.
D3D12_VIDEO_ENCODER_SEQUENCE_GOP_STRUCTURE 여러 비디오 코덱의 GOP 구조를 나타냅니다.
D3D12_VIDEO_ENCODER_SEQUENCE_GOP_STRUCTURE_H264 H.264 비디오 인코딩에 대한 GOP 구조를 나타냅니다.
D3D12_VIDEO_ENCODER_SEQUENCE_GOP_STRUCTURE_HEVC HEVC 비디오 인코딩에 대한 GOP 구조를 나타냅니다.
D3D12_VIDEO_EXTENSION_COMMAND_DESC 비디오 확장 명령에 대해 설명합니다. (D3D12_VIDEO_EXTENSION_COMMAND_DESC)
D3D12_VIDEO_EXTENSION_COMMAND_INFO 비디오 확장 명령에 대해 설명합니다. (D3D12_VIDEO_EXTENSION_COMMAND_INFO)
D3D12_VIDEO_EXTENSION_COMMAND_PARAMETER_INFO 비디오 확장 명령 매개 변수를 설명합니다.
D3D12_VIDEO_FORMAT 리소스 콘텐츠 설명에 대한 픽셀 형식과 색 공간의 조합을 정의합니다.
D3D12_VIDEO_MOTION_ESTIMATOR_DESC ID3D12VideoMotionEstimator를 설명합니다. ID3D12VideoDevice1::CreateVideoMotionEstimator에 이 구조를 전달하여 ID3D12VideoMotionEstimator 인스턴스를 만듭니다.

D3D12_VIDEO_MOTION_ESTIMATOR_INPUT ID3D12VideoEncodeCommandList::EstimateMotion 호출에 대한 입력 매개 변수를 지정합니다.
D3D12_VIDEO_MOTION_ESTIMATOR_OUTPUT ID3D12VideoEncodeCommandList::EstimateMotion 호출에 대한 출력 매개 변수를 지정합니다.
D3D12_VIDEO_MOTION_VECTOR_HEAP_DESC ID3D12VideoMotionEstimatorHeap에 대해 설명합니다. ID3D12VideoDevice1::CreateVideoMotionEstimatorHeap에 이 구조를 전달하여 ID3D12VideoMotionEstimatorHeap의 인스턴스를 만듭니다.
D3D12_VIDEO_PROCESS_ALPHA_BLENDING 비디오 처리를 위한 알파 혼합 매개 변수를 지정합니다.
D3D12_VIDEO_PROCESS_FILTER_RANGE 이미지 필터에 대해 지원되는 값의 범위를 정의합니다. (D3D12_VIDEO_PROCESS_FILTER_RANGE)
D3D12_VIDEO_PROCESS_INPUT_STREAM 비디오 프로세서 혼합 기능에 대한 입력 정보를 포함합니다.
D3D12_VIDEO_PROCESS_INPUT_STREAM_ARGUMENTS ID3D12VideoCommandList::P rocessFrames에 전달된 입력 스트림에 대한 입력 스트림 인수를 지정합니다.
D3D12_VIDEO_PROCESS_INPUT_STREAM_ARGUMENTS1 각 호출에 대한 필드 형식 변경을 지원하는 ID3D12VideoProcessCommandList1::P rocessFrames1에 전달된 입력 스트림에 대한 입력 스트림 인수를 지정합니다.
D3D12_VIDEO_PROCESS_INPUT_STREAM_DESC 비디오 프로세스 작업의 입력 스트림에 대한 매개 변수를 지정합니다.
D3D12_VIDEO_PROCESS_INPUT_STREAM_RATE 스트림 속도에 대한 정보를 제공합니다.
D3D12_VIDEO_PROCESS_LUMA_KEY 루마 키 지정에 사용되는 설정을 지정합니다.
D3D12_VIDEO_PROCESS_OUTPUT_STREAM 비디오 처리 명령의 출력 스트림을 나타냅니다.
D3D12_VIDEO_PROCESS_OUTPUT_STREAM_ARGUMENTS ID3D12VideoCommandList::P rocessFrames에 전달된 출력에 대한 출력 스트림 인수를 지정합니다.

D3D12_VIDEO_PROCESS_OUTPUT_STREAM_DESC ID3D12VideoProcessCommandList::P rocessFrames에 전달된 출력에 대한 출력 스트림 인수를 지정합니다.
D3D12_VIDEO_PROCESS_REFERENCE_SET 비디오 처리를 수행하는 데 필요한 참조 프레임을 포함합니다.
D3D12_VIDEO_PROCESS_TRANSFORM 비디오 처리를 위한 변환 매개 변수를 지정합니다.
D3D12_VIDEO_SAMPLE 그림 버퍼의 너비, 높이, 서식 및 색 공간에 대해 설명합니다.
D3D12_VIDEO_SCALE_SUPPORT 비디오 스케일러에 대해 지원되는 출력 크기의 크기 조정 범위에 대해 설명합니다.
D3D12_VIDEO_SIZE_RANGE 비디오 스케일러에 대해 지원되는 크기 범위에 대해 설명합니다.

열거형

D3D12_BITSTREAM_ENCRYPTION_TYPE 비트 스트림 암호화 유형을 지정합니다.
D3D12_FEATURE_VIDEO 쿼리할 Direct3D 12 비디오 기능 또는 기능 집합을 지정합니다.
D3D12_VIDEO_DECODE_ARGUMENT_TYPE D3D12_VIDEO_DECODE_FRAME_ARGUMENT 인수 형식을 지정합니다.
D3D12_VIDEO_DECODE_CONFIGURATION_FLAGS 비디오 디코딩에 대한 구성을 지정합니다.
D3D12_VIDEO_DECODE_CONVERSION_SUPPORT_FLAGS 비디오 디코딩 변환 작업이 지원되는지 여부를 지정합니다.
D3D12_VIDEO_DECODE_HISTOGRAM_COMPONENT 구성 요소별 히스토그램 정보의 배열에 대한 인덱스를 지정합니다. (D3D12_VIDEO_DECODE_HISTOGRAM_COMPONENT)

D3D12_VIDEO_DECODE_HISTOGRAM_COMPONENT_FLAGS 비디오 디코딩 히스토그램과 함께 사용되는 구성 요소의 하위 집합을 나타내는 플래그입니다. (D3D12_VIDEO_DECODE_HISTOGRAM_COMPONENT_FLAGS)
D3D12_VIDEO_DECODE_STATUS 비디오 디코딩 작업의 상태를 지정합니다.
D3D12_VIDEO_DECODE_SUPPORT_FLAGS 비디오 디코딩 작업이 지원되는지 여부를 지정합니다.
D3D12_VIDEO_DECODE_TIER 애플리케이션 정의 텍스처 및 버퍼의 필수 형식을 결정하는 하드웨어 비디오 디코더의 디코딩 계층을 지정합니다.
D3D12_VIDEO_ENCODER_CODEC Direct3D 12 비디오 인코딩을 위한 코덱을 지정합니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_H264_DIRECT_MODES H.264 비디오 인코딩에 대한 직접 모드를 지정합니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_H264_FLAGS H.264 비디오 인코딩에 대한 구성 플래그를 지정합니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_H264_SLICES_DEBLOCKING_MODE_FLAGS D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_H264_SLICES_DEBLOCKING_MODES 값의 비트 OR 조합을 허용하는 플래그 열거형입니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_H264_SLICES_DEBLOCKING_MODES H.264 사양의 *disable_deblocking_filter_idc* 구문에 정의된 조각 디블로킹 모드를 지정합니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_HEVC_CUSIZE HEVC에 대한 루마 코딩 블록 크기에 사용할 수 있는 값을 지정합니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_HEVC_FLAGS HEVC 비디오 인코딩에 대한 구성 플래그를 지정합니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_HEVC_TUSIZE HEVC에 대한 루마 변환 블록 크기에 사용할 수 있는 값을 지정합니다.
D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_SUPPORT_H264_FLAGS H.264 비디오 인코딩에 대한 구성 지원 플래그를 지정합니다.

D3D12_VIDEO_ENCODER_CODEC_CONFIGURATION_SUPPORT_HEVC_FLAGS HEVC 비디오 인코딩에 대한 구성 지원 플래그를 지정합니다.
D3D12_VIDEO_ENCODER_ENCODE_ERROR_FLAGS ID3D12VideoEncodeCommandList2::EncodeFrame 작업 중에 발생한 오류를 지정합니다.
D3D12_VIDEO_ENCODER_FLAGS 비디오 인코더 만들기에 대한 플래그를 지정합니다.
D3D12_VIDEO_ENCODER_FRAME_SUBREGION_LAYOUT_MODE 비디오 인코더 프레임 하위 구성 레이아웃 모드를 지정합니다.
D3D12_VIDEO_ENCODER_FRAME_TYPE_H264 H.264 비디오 프레임의 형식을 지정합니다.
D3D12_VIDEO_ENCODER_FRAME_TYPE_HEVC HEVC 비디오 프레임의 형식을 지정합니다.
D3D12_VIDEO_ENCODER_HEAP_FLAGS 비디오 인코딩에 대한 힙 옵션을 지정합니다.
D3D12_VIDEO_ENCODER_INTRA_REFRESH_MODE 비디오 인코더 인트라 새로 고침 모드를 지정합니다.
D3D12_VIDEO_ENCODER_LEVELS_H264 H.264 인코딩의 인코더 수준을 지정합니다.
D3D12_VIDEO_ENCODER_LEVELS_HEVC HEVC(고효율 비디오 코딩) 인코딩에 대한 인코더 수준을 지정합니다.
D3D12_VIDEO_ENCODER_MOTION_ESTIMATION_PRECISION_MODE 비디오 인코딩에 대한 동작 추정 정밀도 모드를 지정합니다.
D3D12_VIDEO_ENCODER_PICTURE_CONTROL_CODEC_DATA_H264_FLAGS H.264 관련 그림 컨트롤 속성에 대한 플래그를 지정합니다.
D3D12_VIDEO_ENCODER_PICTURE_CONTROL_CODEC_DATA_HEVC_FLAGS HEVC 관련 그림 컨트롤 속성에 대한 플래그를 지정합니다.
D3D12_VIDEO_ENCODER_PICTURE_CONTROL_FLAGS 비디오 인코더 그림 컨트롤 플래그를 지정합니다.

D3D12_VIDEO_ENCODER_PROFILE_H264 H.264 인코딩에 대한 인코더 프로필을 지정합니다.
D3D12_VIDEO_ENCODER_PROFILE_HEVC HEVC(고효율 비디오 코딩) 인코딩에 대한 인코더 프로필을 지정합니다.
D3D12_VIDEO_ENCODER_RATE_CONTROL_FLAGS 3D12_VIDEO_ENCODER_RATE_CONTROL 구조체에 대한 플래그를 지정합니다.
D3D12_VIDEO_ENCODER_RATE_CONTROL_MODE 비디오 인코더 속도 제어 모드를 지정합니다.
D3D12_VIDEO_ENCODER_SEQUENCE_CONTROL_FLAGS 비디오 인코더 시퀀스 컨트롤 속성에 대한 플래그를 지정합니다.
D3D12_VIDEO_ENCODER_SUPPORT_FLAGS 비디오 인코더 기능에 대한 플래그를 지정합니다.
D3D12_VIDEO_ENCODER_TIER_HEVC HEVC(고효율 비디오 코딩) 인코딩에 대한 인코더 계층을 지정합니다.
D3D12_VIDEO_ENCODER_VALIDATION_FLAGS 지원되지 않는 인코더 기능을 지정하는 플래그입니다.
D3D12_VIDEO_EXTENSION_COMMAND_PARAMETER_FLAGS 연결된 비디오 확장 명령 매개 변수의 사용을 지정합니다.
D3D12_VIDEO_EXTENSION_COMMAND_PARAMETER_STAGE 비디오 확장 명령에 대한 매개 변수 단계를 지정합니다.
D3D12_VIDEO_EXTENSION_COMMAND_PARAMETER_TYPE 비디오 확장 명령에 대한 매개 변수 유형을 지정합니다.
D3D12_VIDEO_FIELD_TYPE 비디오 프레임이 인터레이스되는 방법을 지정합니다.
D3D12_VIDEO_FRAME_CODED_INTERLACE_TYPE 코딩된 비디오 프레임의 인터레이스 유형을 지정합니다.
D3D12_VIDEO_FRAME_STEREO_FORMAT 스테레오 3D 비디오 프레임의 메모리 레이아웃을 정의합니다.

D3D12_VIDEO_MOTION_ESTIMATOR_SEARCH_BLOCK_SIZE 비디오 동작 추정에 지원되는 검색 블록 크기를 정의합니다.
D3D12_VIDEO_MOTION_ESTIMATOR_SEARCH_BLOCK_SIZE_FLAGS 비디오 인코더가 지원할 수 있는 동작 예측 검색 블록 크기를 지정합니다.
D3D12_VIDEO_MOTION_ESTIMATOR_VECTOR_PRECISION 비디오 동작 추정에 대한 벡터 정밀도 값을 정의합니다.
D3D12_VIDEO_MOTION_ESTIMATOR_VECTOR_PRECISION_FLAGS 비디오 인코더가 지원하는 동작 예측 벡터 정밀도를 지정합니다.
D3D12_VIDEO_PROCESS_ALPHA_FILL_MODE 비디오 처리를 위한 알파 채우기 모드를 지정합니다. (D3D12_VIDEO_PROCESS_ALPHA_FILL_MODE)
D3D12_VIDEO_PROCESS_AUTO_PROCESSING_FLAGS 비디오 프로세서에서 지원할 수 있는 자동 처리 기능을 지정합니다.
D3D12_VIDEO_PROCESS_DEINTERLACE_FLAGS 디인터레이싱 비디오 프로세서 기능을 지정합니다.
D3D12_VIDEO_PROCESS_FEATURE_FLAGS 비디오 프로세서에서 지원할 수 있는 기능을 지정합니다.
D3D12_VIDEO_PROCESS_FILTER_FLAGS D3D12_VIDEO_PROCESS_FILTER 열거형에서 정의한 이미지 필터에 대한 지원을 지정합니다.
D3D12_VIDEO_PROCESS_INPUT_STREAM_FLAGS 비디오 처리 입력 스트림에 대한 플래그를 지정합니다.
D3D12_VIDEO_PROCESS_ORIENTATION 비디오 프로세서에서 수행할 방향 작업을 지정합니다.
D3D12_VIDEO_PROCESS_SUPPORT_FLAGS 비디오 형식 및 색 영역 변환 작업이 지원되는지 여부를 지정합니다.
D3D12_VIDEO_PROTECTED_RESOURCE_SUPPORT_FLAGS 비디오 작업에서 보호된 리소스에 대한 지원을 지정합니다.
D3D12_VIDEO_SCALE_SUPPORT_FLAGS 비디오 스케일링기의 크기 조정 기능을 지정합니다.

d3d9.h 헤더

아티클2023. 08. 23.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 9 그래픽
- Microsoft Media Foundation

d3d9.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDirect3D9 IDirect3D9(d3d9.h) 인터페이스 애플리케이션은 IDirect3D9 인터페이스의 메서드를 사용하여 Microsoft Direct3D 개체를 만들고 환경을 설정합니다.
IDirect3D9Ex 애플리케이션은 IDirect3D9Ex 인터페이스(IDirect3D9에서 상속됨)의 메서드를 사용하여 Microsoft Direct3D 9Ex 개체를 만들고 환경을 설정합니다.
IDirect3D9ExOverlayExtension Direct3D 디바이스의 오버레이 하드웨어 기능을 쿼리합니다. (IDirect3D9ExOverlayExtension)
IDirect3DAuthenticatedChannel9 그래픽 드라이버 또는 Direct3D 런타임과 통신 채널을 제공합니다. 이 인터페이스에 대한 포인터를 얻으려면 IDirect3DDevice9Video::CreateAuthenticatedChannel을 호출합니다.
IDirect3DBaseTexture9 IDirect3DBaseTexture9(d3d9.h) 인터페이스 애플리케이션은 IDirect3DBaseTexture9 인터페이스의 메서드를 사용하여 텍스처 리소스를 조작합니다.
IDirect3DCryptoSession9 암호화 세션을 나타냅니다. 이 인터페이스에 대한 포인터를 얻으려면 IDirect3DDevice9Video::CreateCryptoSession을 호출합니다.
IDirect3DCubeTexture9 IDirect3DCubeTexture9(d3d9.h) 인터페이스 애플리케이션은 IDirect3DCubeTexture9 인터페이스의 메서드를 사용하여 큐브 텍스처 리소스를 조작합니다.

IDirect3DDevice9 IDirect3DDevice9(d3d9.h) 애플리케이션은 IDirect3DDevice9 인터페이스의 메서드를 사용하여 DrawPrimitive 기반 렌더링을 수행하고 리소스를 만듭니다.
IDirect3DDevice9Ex 애플리케이션은 IDirect3DDevice9Ex 인터페이스의 메서드를 사용하여 기본 형식을 렌더링하고, 리소스를 만들고, 시스템 수준 변수로 작업하고, 감마 램프 수준을 조정하고, 색상표를 사용하고, 셰이더를 만듭니다.
IDirect3DDevice9Video 애플리케이션이 그래픽 드라이버에서 구현한 콘텐츠 보호 및 암호화 서비스를 사용할 수 있도록 합니다. 이 인터페이스에 대한 포인터를 얻으려면 D3D9Ex 디바이스에서 QueryInterface를 호출합니다.
IDirect3DIndexBuffer9 IDirect3DIndexBuffer9(d3d9.h) 인터페이스 애플리케이션은 IDirect3DIndexBuffer9 인터페이스의 메서드를 사용하여 인덱스 버퍼 리소스를 조작합니다.
IDirect3DPixelShader9 IDirect3DPixelShader9(d3d9.h) 인터페이스 애플리케이션은 IDirect3DPixelShader9 인터페이스의 메서드를 사용하여 픽셀 셰이더의 기능을 캡슐화합니다.
IDirect3DQuery9 IDirect3DQuery9(d3d9.h) 인터페이스 애플리케이션은 IDirect3DQuery9 인터페이스의 메서드를 사용하여 드라이버에서 비동기 쿼리를 수행합니다.
IDirect3DResource9 IDirect3DResource9(d3d9.h) 인터페이스는 애플리케이션에서 리소스를 쿼리하고 준비하는 데 사용됩니다.
IDirect3DStateBlock9 IDirect3DStateBlock9(d3d9.h) 인터페이스는 애플리케이션에서 렌더링 상태를 캡슐화하는 데 사용됩니다.
IDirect3DSurface9 IDirect3DSurface9(d3d9.h) 인터페이스는 애플리케이션에서 표면을 쿼리하고 준비하는 데 사용됩니다.
IDirect3DSwapChain9

IDirect3DSwapChain9(d3d9.h) 인터페이스는 애플리케이션에서 스왑 체인을 조작하는 데 사용됩니다.
IDirect3DSwapChain9Ex 애플리케이션은 IDirect3DSwapChain9Ex 인터페이스의 메서드를 사용하여 스왑 체인을 조작합니다.
IDirect3DTexture9 IDirect3DTexture9(d3d9.h) 인터페이스는 애플리케이션에서 텍스처 리소스를 조작하는 데 사용됩니다.
IDirect3DVertexBuffer9 IDirect3DVertexBuffer9(d3d9.h) 인터페이스는 애플리케이션에서 꼭짓점 버퍼 리소스를 조작하는 데 사용됩니다.
IDirect3DVertexDeclaration9 IDirect3DVertexDeclaration9(d3d9.h) 인터페이스는 애플리케이션에서 꼭짓점 셰이더 선언을 캡슐화하는 데 사용됩니다.
IDirect3DVertexShader9 IDirect3DVertexShader9(d3d9.h) 인터페이스는 애플리케이션에서 꼭짓점 셰이더의 기능을 캡슐화하는 데 사용됩니다.
IDirect3DVolume9 IDirect3DVolume9(d3d9.h) 인터페이스는 애플리케이션에서 볼륨 리소스를 조작하는 데 사용됩니다.
IDirect3DVolumeTexture9 IDirect3DVolumeTexture9(d3d9.h) 인터페이스는 애플리케이션에서 볼륨 텍스처 리소스를 조작하는 데 사용됩니다.

Functions

Direct3DCreate9 Direct3DCreate9Ex 함수(d3d9.h)는 IDirect3D9 개체를 만들고 인터페이스를 반환합니다.

Direct3DCreate9Ex IDirect3D9Ex 개체를 만들고 해당 개체에 대한 인터페이스를 반환합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d3d9caps.h 헤더

아티클2023. 08. 23.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 9 그래픽
- Microsoft Media Foundation

d3d9caps.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

D3DCAPS9 Direct3D 개체를 통해 노출되는 하드웨어의 기능을 나타냅니다.
D3DCONTENTPROTECTIONCAPS 디스플레이 드라이버의 콘텐츠 보호 기능에 대해 설명합니다.
D3DOVERLAYCAPS Direct3D 디바이스에 대한 하드웨어 오버레이 기능을 지정합니다.
D3DPSHADERCAPS2_0 픽셀 셰이더 드라이버 대문자입니다.
D3DVSHADERCAPS2_0 꼭짓점 셰이더 대문자.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d3d9helper.h 헤더

아티클2023. 08. 23.

이 헤더는 Direct3D 9 그래픽에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 9 그래픽

d3d9helper.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDirect3D9 IDirect3D9 인터페이스(d3d9helper.h)는 Microsoft Direct3D 개체를 만들고 환경을 설정하는 메서드를 제공합니다.
IDirect3DBaseTexture9 IDirect3DBaseTexture9 인터페이스(d3d9helper.h)는 큐브 및 볼륨 텍스처를 포함하여 텍스처 리소스를 조작하는 메서드를 제공합니다.
IDirect3DCubeTexture9 IDirect3DCubeTexture9 인터페이스(d3d9helper.h)는 큐브 텍스처 리소스를 조작하는 메서드를 제공합니다. (IDirect3DCubeTexture9)
IDirect3DDevice9 IDirect3DDevice9 인터페이스(d3d9helper.h)는 DrawPrimitive 기반 렌더링, 리소스 및 셰이더 만들기 등을 수행하는 메서드를 제공합니다.
IDirect3DIndexBuffer9 IDirect3DIndexBuffer9 인터페이스(d3d9helper.h)는 인덱스 버퍼 리소스를 조작하는 메서드를 제공합니다.
IDirect3DPixelShader9 IDirect3DPixelShader9 인터페이스(d3d9helper.h)는 픽셀 셰이더의 기능을 캡슐화하는 메서드를 제공합니다.
IDirect3DQuery9 IDirect3DQuery9 인터페이스(d3d9helper.h)는 드라이버에서 비동기 쿼리를 수행하는 메서드를 제공합니다.

IDirect3DResource9 IDirect3DResource9 인터페이스(d3d9helper.h)는 리소스를 쿼리하고 준비하는 메서드를 제공합니다.
IDirect3DStateBlock9 IDirect3DStateBlock9 인터페이스(d3d9helper.h)는 렌더링 상태를 캡슐화하는 메서드를 제공합니다.
IDirect3DSurface9 IDirect3DSurface9 인터페이스(d3d9helper.h)는 표면을 쿼리하고 준비하는 메서드를 제공합니다.
IDirect3DSwapChain9 IDirect3DSwapChain9 인터페이스(d3d9helper.h)는 스왑 체인을 조작하는 메서드를 제공합니다.
IDirect3DTexture9 IDirect3DTexture9 인터페이스(d3d9helper.h)는 텍스처 리소스를 조작하는 메서드를 제공합니다.
IDirect3DVertexBuffer9 IDirect3DVertexBuffer9 인터페이스(d3d9helper.h)는 꼭짓점 버퍼 리소스를 조작하는 메서드를 제공합니다.
IDirect3DVertexDeclaration9 IDirect3DVertexDeclaration9 인터페이스(d3d9helper.h)는 꼭짓점 셰이더 선언을 캡슐화하는 메서드를 제공합니다.
IDirect3DVertexShader9 IDirect3DVertexShader9 인터페이스(d3d9helper.h)는 꼭짓점 셰이더의 기능을 캡슐화하는 메서드를 제공합니다.
IDirect3DVolume9 IDirect3DVolume9 인터페이스(d3d9helper.h)는 볼륨 리소스를 조작하는 메서드를 제공합니다.
IDirect3DVolumeTexture9 IDirect3DVolumeTexture9 인터페이스(d3d9helper.h)는 볼륨 텍스처 리소스를 조작하는 메서드를 제공합니다.

Functions

Direct3DCreate9 Direct3DCreate9 함수(d3d9helper.h)는 IDirect3D9 개체를 만들고 인터페이스를 반환합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d3dcommon.h 헤더

아티클2023. 08. 23.

이 헤더는 Direct3D 11 그래픽에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 11 그래픽

d3dcommon.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID3D10Blob 이 인터페이스는 임의 길이 데이터를 반환하는 데 사용됩니다.
ID3DDestructionNotifier **ID3DDestructionNotifier** 는 Direct3D nano-COM 개체가 제거될 때 콜백을 등록하는 데 사용할 수 있는 인터페이스입니다.
ID3DInclude ID3DInclude는 애플리케이션이 셰이더를 열고 닫기 위해 사용자 재정의 가능한 메서드를 호출할 수 있도록 사용자가 구현하는 포함 인터페이스입니다.

구조체

D3D_SHADER_MACRO 셰이더 매크로를 정의합니다.

열거형

D3D_CBUFFER_TYPE 상수 버퍼 데이터의 의도된 사용을 식별하는 값입니다.

D3D_DRIVER_TYPE 드라이버 유형 옵션입니다.
D3D_FEATURE_LEVEL Direct3D 디바이스를 대상으로 하는 기능 집합에 대해 설명합니다.
D3D_INCLUDE_TYPE 셰이더의 위치를 나타내는 값
D3D_INTERPOLATION_MODE 래스터화 중에 값을 계산하는 방법에 영향을 미치는 보간 모드를 지정합니다.
D3D_MIN_PRECISION 원하는 최소 보간 정밀도를 나타내는 값입니다.
D3D_NAME 시스템 값 의미 체계를 사용하는 셰이더 매개 변수를 식별하는 값입니다.
D3D_PARAMETER_FLAGS 함수 매개 변수에 대한 의미 체계 플래그를 나타냅니다.
D3D_PRIMITIVE 파이프라인이 기하 도형 또는 헐 셰이더 입력 기본 형식을 해석하는 방법을 나타냅니다.
D3D_PRIMITIVE_TOPOLOGY 파이프라인이 입력 어셈블러 단계에 바인딩된 꼭짓점 데이터를 해석하는 방법을 나타내는 값입니다. 이러한 기본 토폴로지 값은 꼭짓점 데이터가 화면에 렌더링되는 방법을 결정합니다.
D3D_REGISTER_COMPONENT_TYPE 레지스터에 저장할 수 있는 데이터 형식을 식별하는 값입니다.
D3D_RESOURCE_RETURN_TYPE 반환 값 형식을 나타냅니다.
D3D_SHADER_CBUFFER_FLAGS 상수 데이터 버퍼의 의도된 사용을 식별하는 값입니다.

D3D_SHADER_INPUT_FLAGS 셰이더 입력 옵션을 식별하는 값입니다.
D3D_SHADER_INPUT_TYPE 셰이더에 바인딩할 수 있고 셰이더에 대한 리소스 설명의 일부로 반영되는 리소스 종류를 식별하는 값입니다.
D3D_SHADER_VARIABLE_CLASS 셰이더 변수의 클래스를 식별하는 값입니다.
D3D_SHADER_VARIABLE_FLAGS 셰이더 변수에 대한 정보를 식별하는 값입니다.
D3D_SHADER_VARIABLE_TYPE 셰이더 변수에 할당할 수 있는 다양한 데이터, 텍스처 및 버퍼 형식을 식별하는 값입니다.
D3D_SRV_DIMENSION 셰이더 리소스로 볼 리소스의 유형을 식별하는 값입니다.
D3D_TESSELLATOR_DOMAIN 테셀레이터 데이터에 대한 도메인 옵션입니다.
D3D_TESSELLATOR_OUTPUT_PRIMITIVE 출력 기본 형식입니다.
D3D_TESSELLATOR_PARTITIONING 분할 옵션.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d3dcompiler.h 헤더

아티클2023. 08. 23.

이 헤더는 HLSL에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- HLSL

d3dcompiler.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

D3DCompile HLSL 코드 또는 효과 파일을 지정된 대상에 대한 바이트코드로 컴파일합니다.
D3DCompile2 Microsoft HLSL(High Level Shader Language) 코드를 지정된 대상에 대한 바이트코드로 컴파일합니다. (D3DCompile2)
D3DCompileFromFile Microsoft HLSL(High Level Shader Language) 코드를 지정된 대상에 대한 바이트코드로 컴파일합니다. (D3DCompileFromFile)
D3DCompressShaders 셰이더 집합을 보다 압축된 형태로 압축합니다.
D3DCreateBlob 버퍼를 만듭니다.
D3DCreateFunctionLinkingGraph 함수 연결 그래프 인터페이스를 만듭니다.
D3DCreateLinker 링커 인터페이스를 만듭니다. 참고 이 함수는 모든 Direct3D 11 플랫폼에서 미리 컴파일된 HLSL 함수를 만들고 라이브러리에 패키지하고 런타임에 전체 셰이더에 연결하는 데 사용할 수 있는 HLSL 셰이더 연결 기술의 일부입니다. .
D3DDecompressShaders

압축 집합에서 하나 이상의 셰이더를 압축 해제합니다.
D3DDisassemble 컴파일된 HLSL 코드를 분해합니다.
D3DDisassemble10Effect Direct3D10 효과에서 컴파일된 HLSL 코드를 분해합니다.
D3DDisassembleRegion 컴파일된 Microsoft HLSL(High Level Shader Language) 코드의 특정 지역을 디스어셈블합니다.
D3DGetBlobPart 컴파일 결과에서 특정 부분을 검색합니다.
D3DGetDebugInfo 참고 이 API를 사용하여 Windows 스토어 앱을 개발할 수 있지만 Windows 스토어에 제출하는 앱에서는 사용할 수 없습니다. 셰이더 디버그 정보를 가져옵니다.
D3DGetInputAndOutputSignatureBlob 참고 D3DGetInputAndOutputSignatureBlob은 Windows 8.1 후 릴리스에서 변경되거나 사용할 수 없습니다. 대신 D3D_BLOB_INPUT_AND_OUTPUT_SIGNATURE_BLOB 값과 함께 D3DGetBlobPart를 사용합니다. 컴파일 결과에서 입출력 서명을 가져옵니다.
D3DGetInputSignatureBlob 참고 D3DGetInputSignatureBlob은 Windows 8.1 후 릴리스에서 변경되거나 사용할 수 없습니다. 대신 D3D_BLOB_INPUT_SIGNATURE_BLOB 값과 함께 D3DGetBlobPart를 사용합니다. 컴파일 결과에서 입력 서명을 가져옵니다.
D3DGetOutputSignatureBlob 참고 D3DGetOutputSignatureBlob은 Windows 8.1 후 릴리스에서 변경되거나 사용할 수 없습니다. 대신 D3D_BLOB_OUTPUT_SIGNATURE_BLOB 값과 함께 D3DGetBlobPart를 사용합니다. 컴파일 결과에서 출력 서명을 가져옵니다.
D3DGetTraceInstructionOffsets 셰이더 코드 섹션 내 명령에 대한 바이트 오프셋을 검색합니다.
D3DLoadModule 셰이더 모듈의 원본 데이터에서 셰이더 모듈 인터페이스를 만듭니다.

D3DPreprocess 컴파일되지 않은 HLSL 코드를 전처리합니다.
D3DReadFileToBlob 디스크에 있는 파일을 메모리로 읽습니다.
D3DReflection 리플렉션 인터페이스에 대한 포인터를 가져옵니다.
D3DReflectLibrary 함수의 HLSL 라이브러리를 포함하는 원본 데이터에서 라이브러리 반사 인터페이스를 만듭니다.
D3DSetBlobPart 컴파일 결과에 정보를 설정합니다.
D3DStripShader 컴파일 결과에서 원하지 않는 Blob을 제거합니다.
D3DWriteBlobToFile 메모리 Blob을 디스크의 파일에 씁니다.

구조체

D3D_SHADER_DATA 셰이더 데이터를 설명합니다. (D3D_SHADER_DATA)

열거형

D3D_BLOB_PART 임의 길이 데이터 버퍼의 콘텐츠 부분을 식별하는 값입니다.

D3DCOMPILER_STRIP_FLAGS 스트립 플래그 옵션입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

d3dcsx.h 헤더

아티클2023. 08. 23.

이 헤더는 Direct3D 11 그래픽에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 11 그래픽

d3dcsx.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ID3DX11FFT 앞으로 및 역FFT를 캡슐화합니다.
ID3DX11Scan 검사 컨텍스트입니다.
ID3DX11SegmentedScan 분할된 검사 컨텍스트입니다.

Functions

D3DX11CreateFFT ID3DX11FFT COM 인터페이스 개체를 만듭니다. (D3DX11CreateFFT)
D3DX11CreateFFT1DComplex ID3DX11FFT COM 인터페이스 개체를 만듭니다. (D3DX11CreateFFT1DComplex)
D3DX11CreateFFT1DReal ID3DX11FFT COM 인터페이스 개체를 만듭니다. (D3DX11CreateFFT1DReal)
D3DX11CreateFFT2DComplex ID3DX11FFT COM 인터페이스 개체를 만듭니다. (D3DX11CreateFFT2DComplex)

D3DX11CreateFFT2DReal ID3DX11FFT COM 인터페이스 개체를 만듭니다. (D3DX11CreateFFT2DReal)
D3DX11CreateFFT3DComplex ID3DX11FFT COM 인터페이스 개체를 만듭니다. (D3DX11CreateFFT3DComplex)
D3DX11CreateFFT3DReal ID3DX11FFT COM 인터페이스 개체를 만듭니다. (D3DX11CreateFFT3DReal)
D3DX11CreateScan 검사 컨텍스트를 만듭니다.
D3DX11CreateSegmentedScan 분할된 검사 컨텍스트를 만듭니다.

구조체

D3DX11_FFT_BUFFER_INFO FFT에 대한 버퍼 요구 사항을 설명합니다.
D3DX11_FFT_DESC FFT에 대해 설명합니다.

열거형

D3DX11_FFT_CREATE_FLAG FFT 생성 플래그입니다.
D3DX11_FFT_DATA_TYPE FFT 데이터 형식.

D3DX11_FFT_DIM_MASK FFT 데이터의 차원 수입니다.
D3DX11_SCAN_DATA_TYPE 스캔 데이터를 입력합니다.
D3DX11_SCAN_DIRECTION 검색을 수행할 방향입니다.
D3DX11_SCAN_OPCODE opcode를 검사합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

datetimeapi.h 헤더

아티클2023. 08. 23.

이 헤더는 Windows 애플리케이션에 대한 국제화에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 애플리케이션에 대한 국제화

datetimeapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

GetDateFormatA 로캘 식별자가 지정한 로캘의 날짜 문자열로 날짜 서식을 지정합니다. (ANSI)
GetDateFormatEx 날짜 형식을 이름으로 지정된 로캘의 날짜 문자열로 지정합니다.
GetDateFormatW 로캘 식별자가 지정한 로캘의 날짜 문자열로 날짜 서식을 지정합니다. (유니코드)
GetTimeFormatA 시간을 식별자로 지정된 로캘의 시간 문자열로 서식을 지정합니다. 함수는 지정된 시간 또는 로컬 시스템 시간의 형식을 지정합니다. (ANSI)
GetTimeFormatEx 이름으로 지정된 로캘에 대한 시간 문자열로 시간을 형식 지정합니다.
GetTimeFormatW 시간을 식별자로 지정된 로캘의 시간 문자열로 서식을 지정합니다. 함수는 지정된 시간 또는 로컬 시스템 시간의 형식을 지정합니다. (유니코드)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

davclnt.h 헤더

아티클2023. 08. 23.

이 헤더는 WebDAV에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Webdav

davclnt.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DavAddConnection WebDAV 서버 또는 WebDAV 서버의 원격 파일 또는 디렉터리에 대한 보안 연결을 만듭니다.
DavCancelConnectionsToServer WebDAV 서버 또는 WebDAV 서버의 원격 파일 또는 디렉터리에 대한 모든 연결을 닫습니다.
DavDeleteConnection DavAddConnection 함수를 사용하여 만든 연결을 닫습니다.
DavFlushFile 원격 파일의 로컬 버전에서 WebDAV 서버로 데이터를 플러시합니다.
DavGetExtendedError WebDAV 서버가 이전에 실패한 I/O 작업에 대해 반환한 확장 오류 코드 정보를 검색합니다.
DavGetHTTPFromUNCPath 지정된 UNC 경로를 동등한 HTTP 경로로 변환합니다.
DavGetTheLockOwnerOfTheFile WebDAV 서버에 잠긴 파일의 파일 잠금 소유자를 반환합니다.
DavGetUNCFromHTTPPath 지정된 HTTP 경로를 동등한 UNC 경로로 변환합니다.
DavInvalidateCache

WebDAV 서버의 원격 파일에 대한 로컬 캐시의 콘텐츠를 무효화합니다.
DavRegisterAuthCallback WebDAV 클라이언트가 사용자에게 자격 증명을 묻는 메시지를 표시하는 데 사용할 수 있는 애플리케이션 정의 콜백 함수를 등록합니다.
DavUnregisterAuthCallback WebDAV 클라이언트가 사용자에게 자격 증명을 묻는 메시지를 표시하는 데 사용하는 등록된 콜백 함수를 등록 취소합니다.

콜백 함수

PFNDAVAUTHCALLBACK WebDAV 클라이언트는 애플리케이션 정의 DavAuthCallback 콜백 함수를 호출하여 사용자에게 자격 증명을 묻는 메시지를 표시합니다.
PFNDAVAUTHCALLBACK_FREECRED WebDAV 클라이언트는 애플리케이션 정의 DavFreeCredCallback 콜백 함수를 호출하여 DavAuthCallback 콜백 함수에서 검색한 자격 증명 정보를 해제합니다.

구조체

DAV_CALLBACK_AUTH_BLOB DavAuthCallback 콜백 함수에서 검색한 인증 BLOB을 저장합니다.
DAV_CALLBACK_AUTH_UNP DavAuthCallback 콜백 함수에서 검색한 사용자 이름 및 암호 정보를 저장합니다.
DAV_CALLBACK_CRED DavAuthCallback 콜백 함수에서 검색한 사용자 자격 증명 정보를 저장합니다.

열거형

AUTHNEXTSTEP DavAuthCallback 콜백 함수를 성공적으로 호출한 후 WebDAV 클라이언트가 수행해야 하는 다음 작업을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dbghelp.h 헤더

아티클2023. 03. 11.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보
- 시스템 서비스

dbghelp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

EnumDirTree EnumDirTree 함수(dbghelp.h)는 지정된 디렉터리 트리에서 지정된 파일의 모든 항목을 열거합니다.
EnumDirTreeW EnumDirTreeW(유니코드) 함수는 지정된 디렉터리 트리에서 지정된 파일의 모든 항목을 열거합니다.
EnumerateLoadedModules 지정된 프로세스에 대해 로드된 모듈을 열거합니다. (EnumerateLoadedModules)
EnumerateLoadedModules64 지정된 프로세스에 대해 로드된 모듈을 열거합니다. (EnumerateLoadedModules64)
EnumerateLoadedModulesEx EnumerateLoadedModulesEx 함수(dbghelp.h)는 지정된 프로세스에 대해 로드된 모듈을 열거합니다.
EnumerateLoadedModulesExW EnumerateLoadedModulesExW(유니코드) 함수는 지정된 프로세스에 대해 로드된 모듈을 열거합니다.
EnumerateLoadedModulesW64 지정된 프로세스에 대해 로드된 모듈을 열거합니다. (EnumerateLoadedModulesW64)

FindDebugInfoFile 디버그(.dbg) 파일을 찾습니다.
FindDebugInfoFileEx FindDebugInfoFileEx 함수(dbghelp.h)는 지정된 디버그(.dbg) 파일을 찾습니다.
FindDebugInfoFileExW FindDebugInfoFileExW(유니코드) 함수는 지정된 디버그(.dbg) 파일을 찾습니다.
FindExecutableImage 실행 파일을 찾습니다.
FindExecutableImageEx FindExecutableImageEx 함수(dbghelp.h)는 지정된 실행 파일을 찾습니다.
FindExecutableImageExW FindExecutableImageExW(유니코드) 함수는 지정된 실행 파일을 찾습니다.
GetSymLoadError 마지막 기호 로드 오류를 가져옵니다.
GetTimestampForLoadedLibrary 로드된 이미지의 타임스탬프를 검색합니다.
ImageDirectoryEntryToData 이미지 관련 데이터에 대한 액세스 권한을 얻습니다.
ImageDirectoryEntryToDataEx 이미지 헤더 내에서 디렉터리 항목을 찾고 디렉터리 항목에 대한 데이터의 주소를 반환합니다. 이 함수는 있는 데이터의 섹션 헤더(있는 경우)를 반환합니다.
ImagehlpApiVersion 시스템에 설치된 DbgHelp 라이브러리의 버전 정보를 검색합니다.
ImagehlpApiVersionEx 애플리케이션에서 사용하는 라이브러리의 버전 정보를 수정합니다.

ImageNtHeader PE 이미지에서 IMAGE_NT_HEADERS 구조를 찾고 데이터에 대한 포인터를 반환합니다.
ImageRvaToSection 파일로 매핑되는 파일의 이미지 헤더 내에서 상대 RVA(가상 주소)를 찾고 해당 RVA에 대한 섹션 테이블 항목에 대한 포인터를 반환합니다.
ImageRvaToVa 파일로 매핑되는 파일의 이미지 헤더 내에서 상대 RVA(가상 주소)를 찾고 파일에서 해당 바이트의 가상 주소를 반환합니다.
MakeSureDirectoryPathExists 루트부터 시작하여 지정된 경로에 있는 모든 디렉터리를 만듭니다.
MapDebugInformation 이미지의 디버깅 정보에 대한 액세스 권한을 얻습니다.
SearchTreeForFile SearchTreeForFile 함수(dbghelp.h)는 디렉터리 트리에서 지정된 파일을 검색합니다.
SearchTreeForFileW SearchTreeForFileW(유니코드) 함수는 디렉터리 트리에서 지정된 파일을 검색합니다.
SetSymLoadError 기호 로드 오류를 설정합니다.
StackWalk 스택 추적을 가져옵니다. (StackWalk)
StackWalk64 스택 추적을 가져옵니다. (StackWalk64)
StackWalkEx 스택 추적을 가져옵니다. (StackWalkEx)
SymAddrIncludeInlineTrace 지정된 주소가 인라인 프레임 내에 있는지 여부를 나타냅니다.

SymAddSourceStream SymAddSourceStream 함수(dbghelp.h)는 원본 서버에서 사용할 지정된 모듈에 스트림을 추가합니다.
SymAddSourceStreamA 원본 서버에서 사용할 지정된 모듈에 스트림을 추가합니다. (SymAddSourceStreamA)
SymAddSourceStreamW SymAddSourceStreamW(유니코드) 함수는 원본 서버에서 사용할 지정된 모듈에 스트림을 추가합니다.
SymAddSymbol SymAddSymbol 함수(dbghelp.h)는 지정된 모듈에 가상 기호를 추가합니다.
SymAddSymbolW SymAddSymbolW(유니코드) 함수는 지정된 모듈에 가상 기호를 추가합니다.
SymCleanup 프로세스 핸들과 연결된 모든 리소스의 할당을 취소합니다.
SymCompareInlineTrace 두 인라인 추적을 비교합니다.
SymDeleteSymbol SymDeleteSymbol 함수(dbghelp.h)는 지정된 모듈에서 가상 기호를 삭제합니다.
SymDeleteSymbolW SymDeleteSymbolW(유니코드) 함수는 지정된 모듈에서 가상 기호를 삭제합니다.
SymEnumerateModules SymLoadModule64 또는 SymLoadModuleEx 함수에 의해 프로세스에 대해 로드된 모든 모듈을 열거합니다. (SymEnumerateModules)
SymEnumerateModules64 SymLoadModule64 또는 SymLoadModuleEx 함수에 의해 프로세스에 대해 로드된 모든 모듈을 열거합니다. (SymEnumerateModules64)

SymEnumerateModulesW64 SymLoadModule64 또는 SymLoadModuleEx 함수에 의해 프로세스에 대해 로드된 모든 모듈을 열거합니다. (SymEnumerateModulesW64)
SymEnumerateSymbols SymEnumerateSymbols 함수(dbghelp.h)는 지정된 모듈에 대한 모든 기호를 열거합니다.
SymEnumerateSymbols64 지정된 모듈의 모든 기호를 열거합니다. (SymEnumerateSymbols64)
SymEnumerateSymbolsW SymEnumerateSymbolsW(유니코드) 함수는 지정된 모듈에 대한 모든 기호를 열거합니다.
SymEnumerateSymbolsW64 지정된 모듈의 모든 기호를 열거합니다. (SymEnumerateSymbolsW64)
SymEnumLines SymEnumLines 함수(dbghelp.h)는 지정된 모듈의 모든 줄을 열거합니다.
SymEnumLinesW SymEnumLinesW 함수는 지정된 모듈의 모든 줄을 열거합니다.
SymEnumProcesses SymInitialize 함수를 호출한 각 프로세스를 열거합니다.
SymEnumSourceFiles SymEnumSourceFiles 함수(dbghelp.h)는 프로세스의 모든 원본 파일을 열거합니다.
SymEnumSourceFilesW SymEnumSourceFilesW(유니코드) 함수는 프로세스의 모든 원본 파일을 열거합니다.
SymEnumSourceFileTokens 사용 가능한 경우 모듈의 원본 서버 데이터에 있는 모든 개별 항목을 열거합니다.
SymEnumSourceLines SymEnumSourceLines 함수(dbghelp.h)는 모듈의 모든 소스 줄을 열거합니다.

SymEnumSourceLinesW SymEnumSourceLinesW(유니코드) 함수는 모듈의 모든 소스 줄을 열거합니다.
SymEnumSymbols SymEnumSymbols 함수(dbghelp.h)는 프로세스의 모든 기호를 열거합니다.
SymEnumSymbolsEx SymEnumSymbolsEx 함수(dbghelp.h)는 프로세스의 모든 기호를 열거합니다.
SymEnumSymbolsExW SymEnumSymbolsExW(유니코드) 함수는 프로세스의 모든 기호를 열거합니다.
SymEnumSymbolsForAddr SymEnumSymbolsForAddr 함수(dbghelp.h)는 지정된 주소의 기호를 열거합니다.
SymEnumSymbolsForAddrW SymEnumSymbolsForAddrW(유니코드) 함수는 지정된 주소의 기호를 열거합니다.
SymEnumSymbolsW SymEnumSymbolsW(유니코드) 함수는 프로세스의 모든 기호를 열거합니다.
SymEnumTypes SymEnumTypes 함수(dbghelp.h)는 모든 사용자 정의 형식을 열거합니다.
SymEnumTypesByName SymEnumTypesByName 함수(dbghelp.h)는 모든 사용자 정의 형식을 열거합니다.
SymEnumTypesByNameW SymEnumTypesByNameW(유니코드) 함수는 모든 사용자 정의 형식을 열거합니다.
SymEnumTypesW SymEnumTypesW(유니코드) 함수는 모든 사용자 정의 형식을 열거합니다.
SymFindDebugInfoFile SymFindDebugInfoFile 함수(dbghelp.h)는 프로세스 검색 경로에서 .dbg 파일을 찾습니다.

SymFindDebugInfoFileW SymFindDebugInfoFileW(유니코드) 함수는 프로세스 검색 경로에서 .dbg 파일을 찾습니다.
SymFindExecutableImage SymFindExecutableImage 함수(dbghelp.h)는 프로세스 검색 경로에서 실행 파일을 찾습니다.
SymFindExecutableImageW SymFindExecutableImageW(유니코드) 함수는 프로세스 검색 경로에서 실행 파일을 찾습니다.
SymFindFileInPath SymFindFileInPath 함수(dbghelp.h)는 기호 파일 또는 실행 파일 이미지를 찾습니다.
SymFindFileInPathW SymFindFileInPathW(유니코드) 함수는 기호 파일 또는 실행 파일 이미지를 찾습니다.
SymFromAddr SymFromAddr 함수(dbghelp.h)는 지정된 주소에 대한 기호 정보를 검색합니다.
SymFromAddrW SymFromAddrW(유니코드) 함수는 지정된 주소에 대한 기호 정보를 검색합니다.
SymFromIndex SymFromIndex 함수(dbghelp.h)는 지정된 인덱스에 대한 기호 정보를 검색합니다.
SymFromIndexW SymFromIndexW(유니코드) 함수는 지정된 인덱스에 대한 기호 정보를 검색합니다.
SymFromInlineContext SymFromInlineContext 함수(dbghelp.h)는 지정된 주소 및 인라인 컨텍스트에 대한 기호 정보를 검색합니다.
SymFromInlineContextW SymFromInlineContextW(유니코드) 함수는 지정된 주소 및 인라인 컨텍스트에 대한 기호 정보를 검색합니다.
SymFromName SymFromName 함수(dbghelp.h)는 지정된 이름의 기호 정보를 검색합니다.

SymFromNameW SymFromNameW(유니코드) 함수는 지정된 이름의 기호 정보를 검색합니다.
SymFromToken SymFromToken 함수(dbghelp.h)는 지정된 관리 코드 토큰에 대한 기호 정보를 검색합니다.
SymFromTokenW SymFromTokenW(유니코드) 함수는 지정된 관리 코드 토큰에 대한 기호 정보를 검색합니다.
SymFunctionTableAccess 지정된 주소에 대한 함수 테이블 항목을 검색합니다. (SymFunctionTableAccess)
SymFunctionTableAccess64 지정된 주소에 대한 함수 테이블 항목을 검색합니다. (SymFunctionTableAccess64)
SymFunctionTableAccess64AccessRoutines 주소에 대한 함수 테이블 항목 또는 FPO(프레임 포인터 누락) 레코드를 찾습니다.
SymGetExtendedOption 지정된 확장 기호 옵션을 설정 또는 해제할지 여부를 가져옵니다.
SymGetFileLineOffsets64 지정된 모듈 및 파일 이름에 대한 줄 정보를 찾습니다.
SymGetHomeDirectory SymGetHomeDirectory 함수(dbghelp.h)는 Dbghelp에서 사용하는 홈 디렉터리를 검색합니다.
SymGetHomeDirectoryW SymGetHomeDirectoryW(유니코드) 함수는 Dbghelp에서 사용하는 홈 디렉터리를 검색합니다.
SymGetLineFromAddr 지정된 주소의 원본 줄을 찾습니다. (SymGetLineFromAddr)
SymGetLineFromAddr64 지정된 주소의 원본 줄을 찾습니다. (SymGetLineFromAddr64)

SymGetLineFromAddrW64 지정된 주소의 원본 줄을 찾습니다. (SymGetLineFromAddrW64)
SymGetLineFromInlineContext SymGetLineFromInlineContext 함수(dbghelp.h)는 지정된 인라인 컨텍스트에 대한 소스 줄을 찾습니다.
SymGetLineFromInlineContextW SymGetLineFromInlineContextW(유니코드) 함수는 지정된 인라인 컨텍스트에 대한 소스 줄을 찾습니다.
SymGetLineFromName 지정된 모듈, 파일 이름 및 줄 번호의 원본 줄을 찾습니다. (SymGetLineFromName)
SymGetLineFromName64 지정된 모듈, 파일 이름 및 줄 번호의 원본 줄을 찾습니다. (SymGetLineFromName64)
SymGetLineFromNameW64 지정된 모듈, 파일 이름 및 줄 번호의 원본 줄을 찾습니다. (SymGetLineFromNameW64)
SymGetLineNext 다음 원본 줄에 대한 줄 정보를 검색합니다. (SymGetLineNext)
SymGetLineNext64 다음 원본 줄에 대한 줄 정보를 검색합니다. (SymGetLineNext64)
SymGetLineNextW64 다음 원본 줄에 대한 줄 정보를 검색합니다. (SymGetLineNextW64)
SymGetLinePrev 이전 원본 줄의 줄 정보를 검색합니다. (SymGetLinePrev)
SymGetLinePrev64 이전 원본 줄의 줄 정보를 검색합니다. (SymGetLinePrev64)
SymGetLinePrevW64 이전 원본 줄의 줄 정보를 검색합니다. (SymGetLinePrevW64)

SymGetModuleBase 지정된 주소를 포함하는 모듈의 기본 주소를 검색합니다. (SymGetModuleBase)
SymGetModuleBase64 지정된 주소를 포함하는 모듈의 기본 주소를 검색합니다. (SymGetModuleBase64)
SymGetModuleInfo SymGetModuleInfo 함수(dbghelp.h)는 지정된 모듈의 모듈 정보를 검색합니다.
SymGetModuleInfo64 지정된 모듈의 모듈 정보를 검색합니다. (SymGetModuleInfo64)
SymGetModuleInfoW 지정된 모듈의 모듈 정보를 검색합니다. (SymGetModuleInfoW)
SymGetModuleInfoW64 지정된 모듈의 모듈 정보를 검색합니다. (SymGetModuleInfoW64)
SymGetOmaps 로드된 모듈 내에서 omap 테이블을 검색합니다.
SymGetOptions 현재 옵션 마스크를 검색합니다.
SymGetScope SymGetScope 함수(dbghelp.h)는 지정된 인덱스의 범위를 검색합니다.
SymGetScopeW SymGetScopeW(유니코드) 함수는 지정된 인덱스의 범위를 검색합니다.
SymGetSearchPath SymGetSearchPath 함수(dbghelp.h)는 지정된 프로세스에 대한 기호 검색 경로를 검색합니다.
SymGetSearchPathW SymGetSearchPathW(유니코드) 함수는 지정된 프로세스에 대한 기호 검색 경로를 검색합니다.

SymGetSourceFile SymGetSourceFile 함수(dbghelp.h)는 원본 서버에서 지정된 소스 파일을 검색합니다.
SymGetSourceFileChecksum SymGetSourceFileChecksum 함수(dbghelp.h)는 원본 서버에서 지정된 소스 파일 체크섬을 검색합니다.
SymGetSourceFileChecksumW SymGetSourceFileChecksumW(유니코드) 함수는 원본 서버에서 지정된 원본 파일 체크섬을 검색합니다.
SymGetSourceFileFromToken SymGetSourceFileFromToken 함수(dbghelp.h)는 원본 서버에서 지정된 토큰과 연결된 원본 파일을 검색합니다.
SymGetSourceFileFromTokenW SymGetSourceFileFromTokenW(유니코드) 함수는 원본 서버에서 지정된 토큰과 연결된 원본 파일을 검색합니다.
SymGetSourceFileToken SymGetSourceFileToken 함수(dbghelp.h)는 원본 서버에서 지정된 원본 파일에 대한 토큰을 검색합니다.
SymGetSourceFileTokenW SymGetSourceFileTokenW(유니코드) 함수는 원본 서버에서 지정된 원본 파일에 대한 토큰을 검색합니다.
SymGetSourceFileW SymGetSourceFileW(유니코드) 함수는 원본 서버에서 지정된 소스 파일을 검색합니다.
SymGetSourceVarFromToken SymGetSourceVarFromToken 함수(dbghelp.h)는 원본 서버 토큰에서 지정된 변수 이름과 연결된 값을 검색합니다.
SymGetSourceVarFromTokenW SymGetSourceVarFromTokenW(유니코드) 함수는 원본 서버 토큰에서 지정된 변수 이름과 연결된 값을 검색합니다.

SymGetSymbolFile SymGetSymbolFile 함수(dbghelp.h)는 지정된 기호 경로에서 기호 파일을 찾습니다.
SymGetSymbolFileW SymGetSymbolFileW(유니코드) 함수는 지정된 기호 경로에서 기호 파일을 찾습니다.
SymGetSymFromAddr 지정된 주소의 기호를 찾습니다. (SymGetSymFromAddr)
SymGetSymFromAddr64 지정된 주소의 기호를 찾습니다. (SymGetSymFromAddr64)
SymGetSymFromName 지정된 이름의 기호를 찾습니다. (SymGetSymFromName)
SymGetSymFromName64 지정된 이름의 기호를 찾습니다. (SymGetSymFromName64)
SymGetSymNext 다음 기호에 대한 기호 정보를 검색합니다. (SymGetSymNext)
SymGetSymNext64 다음 기호에 대한 기호 정보를 검색합니다. (SymGetSymNext64)
SymGetSymPrev 이전 기호에 대한 기호 정보를 검색합니다. (SymGetSymPrev)
SymGetSymPrev64 이전 기호에 대한 기호 정보를 검색합니다. (SymGetSymPrev64)
SymGetTypeFromName SymGetTypeFromName 함수(dbghelp.h)는 지정된 형식 이름에 대한 형식 인덱스를 검색합니다.
SymGetTypeFromNameW SymGetTypeFromNameW(유니코드) 함수는 지정된 형식 이름의 형식 인덱스를 검색합니다.

SymGetTypeInfo 지정된 형식 인덱스에 대한 형식 정보를 검색합니다.
SymGetTypeInfoEx 여러 형식 정보를 검색합니다.
SymInitialize SymInitialize 함수(dbghelp.h)는 프로세스에 대한 기호 처리기를 초기화합니다.
SymInitializeW SymInitializeW(유니코드) 함수는 프로세스에 대한 기호 처리기를 초기화합니다.
SymLoadModule 기호 테이블을 로드합니다. (SymLoadModule)
SymLoadModule64 기호 테이블을 로드합니다. (SymLoadModule64)
SymLoadModuleEx SymLoadModuleEx 함수(dbghelp.h)는 지정된 모듈에 대한 기호 테이블을 로드합니다.
SymLoadModuleExW SymLoadModuleExW(유니코드) 함수는 지정된 모듈에 대한 기호 테이블을 로드합니다.
SymMatchFileName SymMatchFileName 함수(dbghelp.h)는 문자열을 파일 이름 및 경로와 비교합니다.
SymMatchFileNameW SymMatchFileNameW(유니코드) 함수는 문자열을 파일 이름 및 경로와 비교합니다.
SymMatchString SymMatchString 함수(dbghelp.h)는 지정된 문자열을 지정된 와일드카드 식과 비교합니다.
SymMatchStringA 지정된 문자열을 지정된 와일드카드 식과 비교합니다. (SymMatchStringA)

SymMatchStringW SymMatchStringW(유니코드) 함수는 지정된 문자열을 지정된 와일드카드 식과 비교합니다.
SymNext SymNext 함수(dbghelp.h)는 다음 기호에 대한 기호 정보를 검색합니다.
SymNextW SymNextW(유니코드) 함수는 다음 기호에 대한 기호 정보를 검색합니다.
SymPrev SymPrev 함수(dbghelp.h)는 이전 기호에 대한 기호 정보를 검색합니다.
SymPrevW SymPrevW(유니코드) 함수(dbghelp.h)는 이전 기호에 대한 기호 정보를 검색합니다.
SymQueryInlineTrace 인라인 추적을 쿼리합니다.
SymRefreshModuleList 프로세스에 대한 모듈 목록을 새로 고칩니다.
SymRegisterCallback 기호 처리기에서 사용할 콜백 함수를 등록합니다. (SymRegisterCallback)
SymRegisterCallback64 기호 처리기에서 사용할 콜백 함수를 등록합니다. (SymRegisterCallback64)
SymRegisterCallbackW64 기호 처리기에서 사용할 콜백 함수를 등록합니다. (SymRegisterCallbackW64)
SymRegisterFunctionEntryCallback 알파 컴퓨터의 스택 보행 프로시저에서 사용할 콜백 함수를 등록합니다. (SymRegisterFunctionEntryCallback)
SymRegisterFunctionEntryCallback64 알파 컴퓨터의 스택 보행 프로시저에서 사용할 콜백 함수를 등록합니다. (SymRegisterFunctionEntryCallback64)

SymSearch SymSearch 함수(dbghelp.h)는 지정된 조건을 충족하는 PDB 기호를 검색합니다.
SymSearchW SymSearchW(유니코드) 함수(dbghelp.h)는 지정된 조건을 충족하는 PDB 기호를 검색합니다.
SymSetContext SymEnumSymbols 함수에서 사용하는 컨텍스트 정보를 설정합니다. 이 함수는 PDB 기호에서만 작동합니다.
SymSetExtendedOption 지정된 확장 기호 옵션을 설정하거나 해제합니다.
SymSetHomeDirectory SymSetHomeDirectory 함수(dbghelp.h)는 Dbghelp에서 사용하는 홈 디렉터리를 설정합니다.
SymSetHomeDirectoryW SymSetHomeDirectoryW(유니코드) 함수(dbghelp.h)는 Dbghelp에서 사용하는 홈 디렉터리를 설정합니다.
SymSetOptions 옵션 마스크를 설정합니다.
SymSetParentWindow 호출자가 사용자 인터페이스를 표시하는 데 사용할 창을 설정합니다.
SymSetScopeFromAddr 로컬 범위를 지정된 주소와 일치하는 기호로 설정합니다.
SymSetScopeFromIndex 로컬 범위를 지정된 인덱스와 일치하는 기호로 설정합니다.
SymSetScopeFromInlineContext 로컬 범위를 지정된 주소 및 인라인 컨텍스트와 일치하는 기호로 설정합니다.
SymSetSearchPath SymSetSearchPath 함수(dbghelp.h)는 지정된 프로세스에 대한 검색 경로를 설정합니다.

SymSetSearchPathW SymSetSearchPathW(유니코드) 함수(dbghelp.h)는 지정된 프로세스에 대한 검색 경로를 설정합니다.
SymSrvDeltaName SymSrvDeltaName 함수(dbghelp.h)는 동일한 기호/이미지의 두 버전 간의 관계를 설명하는 파일의 이름을 생성합니다.
SymSrvDeltaNameW SymSrvDeltaNameW(유니코드) 함수(dbghelp.h)는 동일한 기호/이미지의 두 버전 간의 관계를 설명하는 파일의 이름을 생성합니다.
SymSrvGetFileIndexes SymSrvGetFileIndexes 함수(dbghelp.h)는 파일을 저장하는 데 사용할 지정된 .pdb, .dbg 또는 이미지 파일의 인덱스를 검색합니다.
SymSrvGetFileIndexesW SymSrvGetFileIndexesW(유니코드) 함수(dbghelp.h)는 파일을 저장하는 데 사용할 지정된 .pdb, .dbg 또는 이미지 파일의 인덱스를 검색합니다.
SymSrvGetFileIndexInfo SymSrvGetFileIndexInfo 함수(dbghelp.h)는 지정된 .pdb, .dbg 또는 이미지 파일에 대한 인덱스 정보를 검색합니다.
SymSrvGetFileIndexInfoW SymSrvGetFileIndexInfoW(유니코드) 함수(dbghelp.h)는 지정된 .pdb, .dbg 또는 이미지 파일에 대한 인덱스 정보를 검색합니다.
SymSrvGetFileIndexString SymSrvGetFileIndexString 함수(dbghelp.h)는 지정된 .pdb, .dbg 또는 이미지 파일에 대한 인덱스 문자열을 검색합니다.
SymSrvGetFileIndexStringW SymSrvGetFileIndexStringW(유니코드) 함수(dbghelp.h)는 지정된 .pdb, .dbg 또는 이미지 파일에 대한 인덱스 문자열을 검색합니다.
SymSrvGetSupplement SymSrvGetSupplement 함수(dbghelp.h)는 기호 저장소에 대한 보충에서 지정된 파일을 검색합니다.

SymSrvGetSupplementW SymSrvGetSupplementW(유니코드) 함수(dbghelp.h)는 기호 저장소에 대한 보충에서 지정된 파일을 검색합니다.
SymSrvIsStore SymSrvIsStore 함수(dbghelp.h)는 지정된 경로가 기호 저장소를 가리키는지 여부를 결정합니다.
SymSrvIsStoreW SymSrvIsStoreW(유니코드) 함수(dbghelp.h)는 지정된 경로가 기호 저장소를 가리키는지 여부를 결정합니다.
SymSrvStoreFile SymSrvStoreFile 함수(dbghelp.h)는 지정된 기호 저장소에 파일을 저장합니다.
SymSrvStoreFileW SymSrvStoreFileW(유니코드) 함수(dbghelp.h)는 지정된 기호 저장소에 파일을 저장합니다.
SymSrvStoreSupplement SymSrvStoreSupplement 함수(dbghelp.h)는 지정된 보충 파일의 파일을 기호 저장소에 저장합니다.
SymSrvStoreSupplementW SymSrvStoreSupplementW(유니코드) 함수(dbghelp.h)는 지정된 보충에 있는 파일을 기호 저장소에 저장합니다.
SymUnDName 데코레이트된 C++ 기호 이름을 해제합니다. (SymUnDName)
SymUnDName64 데코레이트된 C++ 기호 이름을 해제합니다. (SymUnDName64)
SymUnloadModule 기호 테이블을 언로드합니다. (SymUnloadModule)
SymUnloadModule64 기호 테이블을 언로드합니다. (SymUnloadModule64)

UnDecorateSymbolName UnDecorateSymbolName 함수(dbghelp.h)는 지정된 데코레이트된 C++ 기호 이름을 지정 취소합니다.
UnDecorateSymbolNameW UnDecorateSymbolNameW(유니코드) 함수(dbghelp.h)는 지정된 데코레이트된 C++ 기호 이름을 해제합니다.
UnmapDebugInformation MapDebugInformation 함수 호출에 의해 할당된 메모리 및 리소스의 할당을 취소합니다.

콜백 함수

PENUMDIRTREE_CALLBACK PENUMDIRTREE_CALLBACK(dbghelp.h)는 EnumDirTree 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. 일치 항목을 찾을 때마다 호출됩니다.
PENUMDIRTREE_CALLBACKW PENUMDIRTREE_CALLBACKW(유니코드)는 EnumDirTree 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. 일치 항목을 찾을 때마다 호출됩니다.
PENUMLOADED_MODULES_CALLBACK PENUMLOADED_MODULES_CALLBACK(dbghelp.h)는 EnumerateLoadedModules64 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PENUMLOADED_MODULES_CALLBACK64 PENUMLOADED_MODULES_CALLBACK64(dbghelp.h)는 EnumerateLoadedModules64 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PENUMLOADED_MODULES_CALLBACKW64 PENUMLOADED_MODULES_CALLBACKW64(유니코드)는 EnumerateLoadedModules64 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.

PENUMSOURCEFILETOKENSCALLBACK 모듈에 대해 PDB에 저장된 원본 서버 버전 제어 정보를 열거하는 SymEnumSourceFileTokens 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PFIND_DEBUG_FILE_CALLBACK PFIND_DEBUG_FILE_CALLBACK(dbghelp.h)는 FindDebugInfoFileEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PFIND_DEBUG_FILE_CALLBACKW PFIND_DEBUG_FILE_CALLBACKW(유니코드)는 FindDebugInfoFileEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PFIND_EXE_FILE_CALLBACK PFIND_EXE_FILE_CALLBACK(dbghelp.h)는 FindExecutableImageEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PFIND_EXE_FILE_CALLBACKW PFIND_EXE_FILE_CALLBACKW(유니코드)는 FindExecutableImageEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. FindExecutableImageEx에서 찾은 실행 파일이 올바른 실행 파일인지 여부를 확인합니다.
PFINDFILEINPATHCALLBACK PFINDFILEINPATHCALLBACK(dbghelp.h)은 SymFindFileInPath 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PFINDFILEINPATHCALLBACKW PFINDFILEINPATHCALLBACKW(유니코드)는 SymFindFileInPath 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PFUNCTION_TABLE_ACCESS_ROUTINE PFUNCTION_TABLE_ACCESS_ROUTINE(dbghelp.h)는 StackWalk64 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PFUNCTION_TABLE_ACCESS_ROUTINE64 PFUNCTION_TABLE_ACCESS_ROUTINE64(dbghelp.h)는 StackWalk64 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.

PGET_MODULE_BASE_ROUTINE PGET_MODULE_BASE_ROUTINE(dbghelp.h)는 StackWalk64 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PGET_MODULE_BASE_ROUTINE64 PGET_MODULE_BASE_ROUTINE64(dbghelp.h)는 StackWalk64 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PREAD_PROCESS_MEMORY_ROUTINE PREAD_PROCESS_MEMORY_ROUTINE(dbghelp.h)는 StackWalk64 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PREAD_PROCESS_MEMORY_ROUTINE64 PREAD_PROCESS_MEMORY_ROUTINE64(dbghelp.h)는 StackWalk64 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PSYM_ENUMERATESYMBOLS_CALLBACK PSYM_ENUMERATESYMBOLS_CALLBACK SymEnumSymbols, SymEnumTypes 및 SymEnumTypesByName 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PSYM_ENUMERATESYMBOLS_CALLBACKW PSYM_ENUMERATESYMBOLS_CALLBACKW(유니코드)는 SymEnumSymbols, SymEnumTypes 및 SymEnumTypesByName 함수와 함께 사용되는 콜백 함수입니다.
PSYM_ENUMLINES_CALLBACK PSYM_ENUMLINES_CALLBACK(dbghelp.h)는 SymEnumLines 및 SymEnumSourceLines 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PSYM_ENUMLINES_CALLBACKW PSYM_ENUMLINES_CALLBACKW(유니코드)는 SymEnumLines 및 SymEnumSourceLines 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PSYM_ENUMMODULES_CALLBACK PSYM_ENUMMODULES_CALLBACK(dbghelp.h)는 SymEnumerateModules64 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PSYM_ENUMMODULES_CALLBACK64 PSYM_ENUMMODULES_CALLBACK64(dbghelp.h)는 SymEnumerateModules64 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.

PSYM_ENUMMODULES_CALLBACKW64 PSYM_ENUMMODULES_CALLBACKW64(유니코드)는 SymEnumerateModules64 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PSYM_ENUMPROCESSES_CALLBACK SymEnumProcesses 함수와 함께 사용되는 애플리케이션 정의 함수입니다.
PSYM_ENUMSOURCEFILES_CALLBACK PSYM_ENUMSOURCEFILES_CALLBACK(dbghelp.h)는 SymEnumSourceFiles 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PSYM_ENUMSOURCEFILES_CALLBACKW PSYM_ENUMSOURCEFILES_CALLBACKW(유니코드)는 SymEnumSourceFiles 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PSYM_ENUMSYMBOLS_CALLBACK PSYM_ENUMSYMBOLS_CALLBACK(dbghelp.h)는 SymEnumerateSymbols64 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PSYM_ENUMSYMBOLS_CALLBACK64 PSYM_ENUMSYMBOLS_CALLBACK64(dbghelp.h)는 SymEnumerateSymbols64 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PSYM_ENUMSYMBOLS_CALLBACK64W PSYM_ENUMSYMBOLS_CALLBACK64W(유니코드)는 SymEnumerateSymbols64 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PSYMBOL_FUNCENTRY_CALLBACK PSYMBOL_FUNCENTRY_CALLBACK(dbghelp.h)는 SymRegisterFunctionEntryCallback64 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PSYMBOL_FUNCENTRY_CALLBACK64 PSYMBOL_FUNCENTRY_CALLBACK64(dbghelp.h)는 SymRegisterFunctionEntryCallback64 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PSYMBOL_REGISTERED_CALLBACK PSYMBOL_REGISTERED_CALLBACK(dbghelp.h)는 SymRegisterCallback64 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.

PSYMBOL_REGISTERED_CALLBACK64 PSYMBOL_REGISTERED_CALLBACK64(dbghelp.h)는 SymRegisterCallback64 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PSYMBOLSERVERCALLBACKPROC 기호 서버 DLL에 대한 진입점입니다.
PTRANSLATE_ADDRESS_ROUTINE PTRANSLATE_ADDRESS_ROUTINE(dbghelp.h)는 StackWalk64 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PTRANSLATE_ADDRESS_ROUTINE64 PTRANSLATE_ADDRESS_ROUTINE64(dbghelp.h)는 StackWalk64 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.

구조체

주소 주소를 나타냅니다. STACKFRAME64 구조체에 사용됩니다. (ADDRESS)
ADDRESS64 주소를 나타냅니다. STACKFRAME64 구조체에 사용됩니다. (ADDRESS64)
API_VERSION 라이브러리 버전을 포함합니다.
IMAGE_DEBUG_INFORMATION 디버깅 정보를 포함합니다.
IMAGEHLP_CBA_EVENT IMAGEHLP_CBA_EVENT 구조체(dbghelp.h)에는 디버깅 이벤트에 대한 정보가 포함됩니다.
IMAGEHLP_CBA_EVENTW IMAGEHLP_CBA_EVENTW(유니코드) 구조체(dbghelp.h)에는 디버깅 이벤트에 대한 정보가 포함됩니다.

IMAGEHLP_CBA_READ_MEMORY 메모리 읽기 작업에 대한 정보를 포함합니다.
IMAGEHLP_DEFERRED_SYMBOL_LOAD 지연된 기호 로드에 대한 정보를 포함합니다. (IMAGEHLP_DEFERRED_SYMBOL_LOAD)
IMAGEHLP_DEFERRED_SYMBOL_LOAD64 지연된 기호 로드에 대한 정보를 포함합니다. (IMAGEHLP_DEFERRED_SYMBOL_LOAD64)
IMAGEHLP_DEFERRED_SYMBOL_LOADW64 지연된 기호 로드에 대한 정보를 포함합니다. (IMAGEHLP_DEFERRED_SYMBOL_LOADW64)
IMAGEHLP_DUPLICATE_SYMBOL 중복 기호 정보를 포함합니다. (IMAGEHLP_DUPLICATE_SYMBOL)
IMAGEHLP_DUPLICATE_SYMBOL64 중복 기호 정보를 포함합니다. (IMAGEHLP_DUPLICATE_SYMBOL64)
IMAGEHLP_GET_TYPE_INFO_PARAMS 모듈에 대한 형식 정보를 포함합니다.
IMAGEHLP_LINE 원본 파일 줄을 나타냅니다. (IMAGEHLP_LINE)
IMAGEHLP_LINE64 원본 파일 줄을 나타냅니다. (IMAGEHLP_LINE64)
IMAGEHLP_LINEW64 원본 파일 줄을 나타냅니다. (IMAGEHLP_LINEW64)
IMAGEHLP_MODULE 모듈 정보를 포함합니다. (IMAGEHLP_MODULE)
IMAGEHLP_MODULE64 모듈 정보를 포함합니다. (IMAGEHLP_MODULE64)

IMAGEHLP_MODULEW64 모듈 정보를 포함합니다. (IMAGEHLP_MODULEW64)
IMAGEHLP_STACK_FRAME 스택 프레임 정보를 포함합니다.
IMAGEHLP_SYMBOL 기호 정보를 포함합니다. (IMAGEHLP_SYMBOL)
IMAGEHLP_SYMBOL64 기호 정보를 포함합니다. (IMAGEHLP_SYMBOL64)
IMAGEHLP_SYMBOLW64 기호 정보를 포함합니다. (IMAGEHLP_SYMBOLW64)
KDHELP 커널 디버거가 스레드의 커널 스택에서 사용자 모드 콜백을 추적하는 데 사용하는 정보입니다. (KDHELP)
KDHELP64 커널 디버거가 스레드의 커널 스택에서 사용자 모드 콜백을 추적하는 데 사용하는 정보입니다. (KDHELP64)
LOADED_IMAGE 로드된 이미지에 대한 정보를 포함합니다.
MODLOAD_CVMISC CodeView 및 Misc 레코드를 포함합니다.
MODLOAD_DATA 모듈 데이터를 포함합니다.
Omap 주소 맵의 항목을 설명합니다.
SOURCEFILE SOURCEFILE 구조체(dbghelp.h)에는 원본 파일 정보가 포함됩니다.

SOURCEFILEW SOURCEFILEW(유니코드) 구조체(dbghelp.h)에는 원본 파일 정보가 포함됩니다.
SRCCODEINFO SRCCODEINFO 구조체(dbghelp.h)에는 줄 정보가 포함됩니다.
SRCCODEINFOW SRCCODEINFOW(유니코드) 구조체(dbghelp.h)에는 줄 정보가 포함됩니다.
Stackframe 스택 프레임을 나타냅니다. (STACKFRAME)
STACKFRAME_EX 확장 스택 프레임을 나타냅니다.
STACKFRAME64 스택 프레임을 나타냅니다. (STACKFRAME64)
SYMBOL_INFO SYMBOL_INFO 구조체(dbghelp.h)에는 기호 정보가 포함됩니다.
SYMBOL_INFOW SYMBOL_INFOW(유니코드) 구조체(dbghelp.h)에는 기호 정보가 포함됩니다.
SYMSRV_INDEX_INFO SYMSRV_INDEX_INFO 구조체(dbghelp.h)에는 기호 서버 인덱스 정보가 포함됩니다.
SYMSRV_INDEX_INFOW SYMSRV_INDEX_INFOW(유니코드) 구조체(dbghelp.h)에는 기호 서버 인덱스 정보가 포함됩니다.
TI_FINDCHILDREN_PARAMS 형식 인덱스 정보를 포함합니다. SymGetTypeInfo 함수에서 사용됩니다.

열거형

IMAGEHLP_EXTENDED_OPTIONS SymGetExtendedOption 및 SymSetExtendedOption 함수를 사용하여 가져와서 설정할 수 있는 확장 기호 옵션을 나열합니다.
IMAGEHLP_SYMBOL_TYPE_INFO 검색할 기호 정보의 형식을 식별합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dbt.h 헤더

아티클2023. 08. 23.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

dbt.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

_DEV_BROADCAST_USERDEFINED DBT_USERDEFINED 디바이스 이벤트와 연결된 사용자 정의 이벤트 및 선택적 데이터를 포함합니다.
DEV_BROADCAST_DEVICEINTERFACE_A 디바이스 클래스에 대한 정보를 포함합니다. (ANSI)
DEV_BROADCAST_DEVICEINTERFACE_W 디바이스 클래스에 대한 정보를 포함합니다. (유니코드)
DEV_BROADCAST_HANDLE 파일 시스템 핸들에 대한 정보를 포함합니다.
DEV_BROADCAST_HDR WM_DEVICECHANGE 메시지를 통해 보고된 디바이스 이벤트와 관련된 정보에 대한 표준 헤더 역할을 합니다.
DEV_BROADCAST_OEM OEM 정의 디바이스 유형에 대한 정보를 포함합니다.
DEV_BROADCAST_PORT_A 모뎀, 직렬 또는 병렬 포트에 대한 정보를 포함합니다. (ANSI)
DEV_BROADCAST_PORT_W 모뎀, 직렬 또는 병렬 포트에 대한 정보를 포함합니다. (유니코드)

DEV_BROADCAST_VOLUME 논리 볼륨에 대한 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dciman.h 헤더

아티클2023. 08. 23.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

dciman.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DCIBeginAccess 지정된 사각형을 기반으로 프레임 버퍼를 표시하는 액세스 포인터를 가져옵니다.
DCICloseProvider 디스플레이의 디바이스 컨텍스트를 닫습니다.
DCICreatePrimary 기본 표면을 만들고 표면 정보를 가져옵니다.
DCIDestroy 디스플레이 디바이스의 기본 표면을 제거합니다.
DCIEndAccess 표시 프레임 버퍼에 대한 액세스를 해제합니다.
DCIOpenProvider 디스플레이의 디바이스 컨텍스트 핸들을 가져옵니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dcommon.h 헤더

아티클2023. 08. 23.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct2D
- DirectWrite

dcommon.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

D2D_MATRIX_3X2_F 3-by-2 행렬을 나타냅니다.
D2D_MATRIX_4X3_F 4-by-3 부동 소수점 행렬을 설명합니다.
D2D_MATRIX_4X4_F 4-by-4 부동 소수점 행렬을 설명합니다.
D2D_MATRIX_5X4_F 5-by-4 부동 소수점 행렬을 설명합니다.
D2D_POINT_2F 2차원 공간에서 부동 소수점 값으로 표현되는 x 좌표 및 y 좌표 쌍을 나타냅니다.
D2D_POINT_2U 2차원 공간에서 부호 없는 32비트 정수 값으로 표현되는 x 좌표 및 y 좌표 쌍을 나타냅니다.
D2D_RECT_F 왼쪽 위 모서리(왼쪽, 위쪽)의 좌표와 오른쪽 아래 모서리(오른쪽, 아래쪽)의 좌표로 정의된 사각형을 나타냅니다.
D2D_RECT_U 좌표의 왼쪽 위 모퉁이 쌍(왼쪽, 위쪽) 및 오른쪽 아래 모퉁이 좌표 쌍(오른쪽, 아래쪽)으로 정의된 사각형을 나타냅니다. 이러한 좌표는 32비트 정수 값으로 표현됩니다.

D2D_SIZE_F 정렬된 부동 소수점 값 쌍(일반적으로 사각형의 너비 및 높이)을 저장합니다.
D2D_SIZE_U 보통 사각형의 너비와 높이의 순서로 정렬된 정수 쌍을 저장합니다.
D2D_VECTOR_2F FLOAT 값 2개(x, y)의 벡터입니다.
D2D_VECTOR_3F FLOAT 값 3개(x, y, z)의 벡터입니다.
D2D_VECTOR_4F FLOAT 값 4개(x, y, z, w)의 벡터입니다.
D2D1_PIXEL_FORMAT 비트맵 또는 렌더링 대상에 대한 데이터 형식 및 알파 모드를 포함합니다.

열거형

D2D1_ALPHA_MODE 비트맵 또는 렌더링 대상의 알파 값을 처리하는 방법을 지정합니다.
DWRITE_GLYPH_IMAGE_FORMATS 글꼴 전체 수준 또는 문자 모양별로 글꼴에서 지원되는 서식을 지정합니다.
DWRITE_MEASURING_MODE 텍스트 레이아웃에 사용되는 측정 방법을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dcomp.h 헤더

아티클2023. 06. 13.

이 헤더는 DirectComposition에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DirectComposition

dcomp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDCompositionAffineTransform2DEffect 산술 복합 효과는 입력 이미지에서 픽셀의 가중 합계를 사용하여 2개의 이미지를 결합하는 데 사용됩니다. (IDCompositionAffineTransform2DEffect)
IDCompositionArithmeticCompositeEffect 산술 복합 효과는 입력 이미지에서 픽셀의 가중 합계를 사용하여 2개의 이미지를 결합하는 데 사용됩니다. (IDCompositionArithmeticCompositeEffect)
IDCompositionBlendEffect 혼합 효과는 2개 이미지를 결합하는 데 사용됩니다.
IDCompositionBrightnessEffect 밝기 효과는 이미지의 밝기를 제어합니다.
IDCompositionClip 시각적 하위 트리의 렌더링을 사각형 영역으로 제한하는 데 사용되는 클립 개체를 나타냅니다.
IDCompositionColorMatrixEffect 색 행렬 효과는 비트맵의 RGBA 값을 변경합니다.
IDCompositionCompositeEffect 복합 효과는 2개 이상의 이미지를 결합하는 데 사용됩니다.
IDCompositionDelegatedInkTrail

IDCompositionDesktopDevice Win32 데스크톱 애플리케이션에서 DirectComposition을 사용하려면 애플리케이션에서 IDCompositionDesktopDevice 인터페이스를 사용해야 합니다.
IDCompositionDevice 다른 모든 Microsoft DirectComposition 개체에 대한 팩터리 역할을 하며 트랜잭션 컴퍼지션을 제어하는 메서드를 제공합니다. (IDCompositionDevice)
IDCompositionDevice2 다른 모든 Microsoft DirectComposition 개체에 대한 팩터리 역할을 하며 트랜잭션 컴퍼지션을 제어하는 메서드를 제공합니다. (IDCompositionDevice2)
IDCompositionDevice3 다른 모든 Microsoft DirectComposition 개체에 대한 팩터리 역할을 하며 트랜잭션 컴퍼지션을 제어하는 메서드를 제공합니다. (IDCompositionDevice3)
IDCompositionDevice4 컴퍼지션 텍스처의 루트 팩터리 역할을 합니다.
IDCompositionDeviceDebug 애플리케이션 디버깅 및 성능 튜닝에 도움이 되는 렌더링 기능에 대한 액세스를 제공합니다. 이 인터페이스는 DirectComposition 디바이스 인터페이스에서 쿼리할 수 있습니다.
IDCompositionEffect 시각적 개체 하위 트리의 래스터화를 수정하는 비트맵 효과를 나타냅니다.
IDCompositionEffectGroup 시각적 개체 하위 트리의 래스터화를 수정하기 위해 함께 적용되는 비트맵 효과 그룹을 나타냅니다.
IDCompositionFilterEffect 필터 효과를 나타냅니다.
IDCompositionGaussianBlurEffect 가우스 흐림 효과는 일반적으로 이미지 노이즈를 줄이고 세부 정보를 줄이기 위해 가우스 함수에 의해 이미지를 흐리게 하는 데 사용됩니다.

IDCompositionHueRotationEffect 색조 회전 효과는 회전 각도에 따라 색 행렬을 적용하여 이미지의 색조를 변경합니다.
IDCompositionInkTrailDevice
IDCompositionLinearTransferEffect 선형 전송 효과는 각 채널에 대해 제공하는 값 목록에서 만든 선형 함수를 사용하여 이미지의 색 강도를 매핑하는 데 사용됩니다.
IDCompositionMatrixTransform 3-by-2 행렬에 의해 정의된 임의의 아핀 2D 변환을 나타냅니다. (IDCompositionMatrixTransform)
IDCompositionMatrixTransform3D 4-by-4 행렬에 의해 정의된 임의의 3D 변환을 나타냅니다.
IDCompositionRectangleClip 시각적 하위 트리의 렌더링을 지정된 사각형 영역으로 제한하는 클립 개체를 나타냅니다. 필요에 따라 클립 개체에 둥근 모서리가 지정되어 있을 수 있습니다.
IDCompositionRotateTransform z축을 중심으로 시각적 개체의 회전에 영향을 주는 2D 변환을 나타냅니다. 좌표계는 지정된 중심점을 중심으로 회전됩니다.
IDCompositionRotateTransform3D 3D 공간에서 임의의 축을 따라 시각적 개체의 회전에 영향을 주는 3D 변환을 나타냅니다. 좌표계는 지정된 중심점을 중심으로 회전됩니다.
IDCompositionSaturationEffect 이 효과는 이미지의 채도를 변경하는 데 사용됩니다. 채도 효과는 색 행렬 효과의 특수화입니다.
IDCompositionScaleTransform x축 및 y축을 따라 시각적 개체의 배율에 영향을 주는 2D 변환을 나타냅니다. 좌표계는 지정된 중심점에서 크기가 조정됩니다.
IDCompositionScaleTransform3D x축, y축 및 z축을 따라 시각적 개체의 배율에 영향을 주는 3D 변환 효과를 나타냅니다. 좌표계는 지정된 중심점에서 크기가 조정됩니다.

IDCompositionShadowEffect 그림자 효과는 이미지의 알파 채널에서 그림자를 생성하는 데 사용됩니다. 그림자는 알파 값이 높을수록 불투명하고 알파 값이 낮을수록 투명합니다. 흐림 효과의 양과 그림자의 색을 설정할 수 있습니다.
IDCompositionSkewTransform x축 및 y축을 따라 시각적 개체의 기울이기 에 영향을 주는 2D 변환을 나타냅니다. 좌표계가 지정된 중심점 주위로 기울어져 있습니다.
IDCompositionSurface 시각적 트리에서 컴퍼지션을 위해 시각적 개체와 연결할 수 있는 실제 비트맵을 나타냅니다. 이 인터페이스를 사용하여 비트맵 콘텐츠를 업데이트할 수도 있습니다.
IDCompositionSurfaceFactory 애플리케이션에서 제공하는 렌더링 디바이스와 연결된 표면 및 가상 표면 개체를 만듭니다.
IDCompositionTableTransferEffect 테이블 전송 효과는 제공하는 값 목록을 보간하여 만든 전송 함수를 사용하여 이미지의 색 강도를 매핑하는 데 사용됩니다.
IDCompositionTarget Microsoft DirectComposition 시각적 트리와 시각적 트리를 구성해야 하는 대상 간의 바인딩을 나타냅니다.
IDCompositionTexture DComp 시각적 개체에 콘텐츠로 바인딩할 수 있는 원시 Direct3D 텍스처를 나타내는 개체에 대한 인터페이스입니다.
IDCompositionTransform 시각적 하위 트리의 좌표 공간을 수정하는 데 사용할 수 있는 2D 변환을 나타냅니다.
IDCompositionTransform3D 시각적 하위 트리의 래스터화를 수정하는 데 사용할 수 있는 3D 변환 효과를 나타냅니다.
IDCompositionTranslateTransform x축 및 y축을 따라 시각적 개체의 오프셋에만 영향을 주는 2D 변환을 나타냅니다.

IDCompositionTranslateTransform3D x축, y축 및 z축을 따라 시각적 개체의 오프셋에 영향을 주는 3D 변환을 나타냅니다.
IDCompositionTurbulenceEffect 난기류 효과는 Perlin 노이즈 함수를 기반으로 비트맵을 생성하는 데 사용됩니다. 난기류 효과에는 입력 이미지가 없습니다.
IDCompositionVirtualSurface 시각적 트리에서 컴퍼지션에 대한 시각적 개체와 연결할 수 있는 드물게 할당된 비트맵을 나타냅니다.
IDCompositionVisual Microsoft DirectComposition 시각적 개체를 나타냅니다.
IDCompositionVisual2 시각적 트리의 DirectComposition 시각적 개체 하나를 나타냅니다. (IDCompositionVisual2)
IDCompositionVisual3 시각적 트리의 DirectComposition 시각적 개체 하나를 나타냅니다. (IDCompositionVisual3)
IDCompositionVisualDebug 디버그 시각적 개체를 나타냅니다.

Functions

DCompositionAttachMouseDragToHwnd Interaction/InputSink를 만들어 마우스 단추를 아래로 라우팅하고 이후의 이동 및 위로 이벤트를 지정된 HWND로 라우팅합니다.
DCompositionAttachMouseWheelToHwnd Interaction/InputSink를 만들어 마우스 휠 메시지를 지정된 HWND로 라우팅합니다.
DCompositionBoostCompositorClock 시스템에서 대기 시간에 민감한 콘텐츠를 향상시키기 위해 더 높은 새로 고침 속도로 동적으로 전환하도록 요청합니다.

DCompositionCreateDevice 다른 Microsoft DirectComposition 개체를 만드는 데 사용할 수 있는 새 디바이스 개체를 만듭니다. (DCompositionCreateDevice)
DCompositionCreateDevice2 다른 Microsoft DirectComposition 개체를 만드는 데 사용할 수 있는 새 디바이스 개체를 만듭니다. (DCompositionCreateDevice2)
DCompositionCreateDevice3 다른 DirectComposition 개체를 만드는 데 사용할 수 있는 새 DirectComposition 디바이스 개체를 만듭니다.
DCompositionCreateSurfaceHandle Microsoft DirectX 스왑 체인 또는 스왑 버퍼에 바인딩되고 시각적 개체와 연결할 수 있는 새 컴퍼지션 표면 개체를 만듭니다.
DCompositionGetFrameId 지정된 형식의 가장 최근 작성자 프레임의 식별자를 가져옵니다.
DCompositionGetStatistics 컴퍼지션 프레임에 대한 기본 정보와 프레임의 일부인 렌더링 대상 ID 목록을 가져옵니다.
DCompositionGetTargetStatistics
DCompositionWaitForCompositorClock 작성자 클록의 다음 신호가 발생할 때까지 스레드를 중지합니다.

구조체

DCompositionInkTrailPoint

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dcompanimation.h 헤더

아티클2023. 08. 23.

이 헤더는 DirectComposition에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DirectComposition

dcompanimation.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDCompositionAnimation 하나 이상의 Microsoft DirectComposition 개체 속성에 애니메이션 효과를 주는 함수를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dcomptypes.h 헤더

아티클2023. 08. 23.

이 헤더는 DirectComposition에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DirectComposition

dcomptypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

COMPOSITION_FRAME_STATS 작성기 프레임에 대한 타이밍 및 컴퍼지션 통계를 설명합니다.
COMPOSITION_STATS 타이밍 및 컴퍼지션 정보를 설명합니다.
COMPOSITION_TARGET_ID 컴퍼지션 렌더링 대상에 대한 정보를 포함합니다.
COMPOSITION_TARGET_STATS 컴퍼지션 프레임 및 렌더링 대상에 대한 대상별 정보를 포함합니다.
DCOMPOSITION_FRAME_STATISTICS 프레임의 타이밍 및 컴퍼지션 통계를 설명합니다.

열거형

COMPOSITION_FRAME_ID_TYPE 작성기 프레임의 상태 지정하는 상수를 정의합니다.
DCOMPOSITION_BACKFACE_VISIBILITY 시각적 개체에 적용할 백페이스 표시 유형을 지정합니다.

DCOMPOSITION_BITMAP_INTERPOLATION_MODE 비트맵의 픽셀이 화면의 픽셀과 정확히 일대일로 정렬되지 않는 변환으로 비트맵이 구성될 때 사용할 보간 모드를 지정합니다.
DCOMPOSITION_BORDER_MODE 비트맵 또는 클립의 가장자리가 정수 좌표와 축으로 정렬되지 않도록 비트맵을 작성하거나 변환으로 클립을 적용할 때 사용할 테두리 모드를 지정합니다.
DCOMPOSITION_COMPOSITE_MODE 시각적 개체의 비트맵 콘텐츠를 렌더링 대상과 혼합하는 데 사용할 모드입니다.
DCOMPOSITION_OPACITY_MODE 시각적 개체의 유효 불투명도 값을 해당 시각적 개체의 콘텐츠 및 자식에 적용하는 방법을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dde.h 헤더

아티클2023. 08. 23.

이 헤더는 Data Exchange에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 교환

dde.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DdeSetQualityOfService 향후 DDE 대화에 필요한 원시 DDE(동적 데이터 교환) 애플리케이션의 QOS(서비스 품질)를 지정합니다.
FreeDDElParam 게시된 DDE(동적 데이터 교환) 메시지의 lParam 매개 변수로 지정된 메모리를 해제합니다. 게시된 DDE 메시지를 수신하는 애플리케이션은 UnpackDDElParam 함수를 사용하여 lParam 값을 풀고 나면 이 함수를 호출해야 합니다.
ImpersonateDdeClientWindow DDE(동적 데이터 교환) 서버 애플리케이션이 DDE 클라이언트 애플리케이션의 보안 컨텍스트를 가장할 수 있도록 합니다. 이렇게 하면 권한이 없는 DDE 클라이언트로부터 보안 서버 데이터가 보호됩니다.
PackDDElParam DDE(동적 데이터 교환) lParam 값을 프로세스 간에 DDE 데이터를 공유하는 데 사용되는 내부 구조로 압축합니다.
ReuseDDElParam 애플리케이션이 압축된 새 lParam을 할당하는 대신 압축된 DDE(동적 데이터 교환) lParam 매개 변수를 다시 사용할 수 있도록 합니다. 이 함수를 사용하면 압축된 DDE 메시지를 전달하는 애플리케이션의 재할당이 줄어듭니다.
UnpackDDElParam 게시된 DDE 메시지에서 받은 DDE(동적 데이터 교환) lParam 값의 압축을 풉니다.

구조체

DDEACK DDE 애플리케이션이 WM_DDE_ACK 메시지의 일부로 파트너에게 전달하는 상태 플래그를 포함합니다.
DDEADVISE DDE 서버 애플리케이션이 조언 루프 중에 클라이언트 애플리케이션에 데이터를 보내는 방법을 지정하는 플래그를 포함합니다. 클라이언트는 WM_DDE_ADVISE 메시지의 일부로 DDEADVISE 구조체에 대한 핸들을 서버에 전달합니다.
DDEDATA WM_DDE_DATA 메시지의 일부로 전송된 데이터 및 데이터에 대한 정보를 포함합니다.
DDEPOKE WM_DDE_POKE 메시지의 일부로 전송된 데이터 및 데이터에 대한 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ddeml.h 헤더

아티클2023. 08. 23.

이 헤더는 Data Exchange에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 교환

ddeml.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DdeAbandonTransaction 지정된 비동기 트랜잭션을 포기하고 트랜잭션과 연결된 모든 리소스를 해제합니다.
DdeAccessData 지정된 DDE(동적 데이터 교환) 개체의 데이터에 대한 액세스를 제공합니다. 애플리케이션이 개체의 데이터에 대한 액세스를 완료한 경우 DdeUnaccessData 함수를 호출해야 합니다.
DdeAddData 지정된 DDE(동적 데이터 교환) 개체에 데이터를 추가합니다.
DdeClientTransaction 클라이언트와 서버 간의 데이터 트랜잭션을 시작합니다. DDE(동적 데이터 교환) 클라이언트 애플리케이션만 이 함수를 호출할 수 있으며 애플리케이션은 서버와의 대화를 설정한 후에만 사용할 수 있습니다.
DdeCmpStringHandles 두 문자열 핸들의 값을 비교합니다. 문자열 핸들의 값은 연결된 문자열의 경우와 관련이 없습니다.
DdeConnect 지정된 서비스 이름 및 토픽 이름 쌍을 지원하는 서버 애플리케이션과의 대화를 설정합니다. 이러한 서버가 두 개 이상 있는 경우 시스템은 하나만 선택합니다.
DdeConnectList 지정된 서비스 이름 및 토픽 이름 쌍을 지원하는 모든 서버 애플리케이션과 대화를 설정합니다.
DdeCreateDataHandle

DDE(동적 데이터 교환) 개체를 만들고 지정된 버퍼의 데이터로 개체를 채웁니다. DDE 애플리케이션은 파트너 애플리케이션에 데이터를 전달하는 트랜잭션 중에 이 함수를 사용합니다.
DdeCreateStringHandleA 지정된 문자열을 식별하는 핸들을 만듭니다. DDE(동적 데이터 교환) 클라이언트 또는 서버 애플리케이션은 문자열 핸들을 매개 변수로 다른 DDEML(동적 데이터 교환 관리 라이브러리) 함수에 전달할 수 있습니다. (ANSI)
DdeCreateStringHandleW 지정된 문자열을 식별하는 핸들을 만듭니다. DDE(동적 데이터 교환) 클라이언트 또는 서버 애플리케이션은 문자열 핸들을 매개 변수로 다른 DDEML(동적 데이터 교환 관리 라이브러리) 함수에 전달할 수 있습니다. (유니코드)
DdeDisconnect DdeConnect 또는 DdeConnectList 함수에서 시작한 대화를 종료하고 지정된 대화 핸들을 무효화합니다.
DdeDisconnectList 지정된 대화 목록을 삭제하고 목록과 연결된 모든 대화를 종료합니다.
DdeEnableCallback 호출 애플리케이션에서 현재 설정한 특정 대화 또는 모든 대화에 대해 트랜잭션을 사용하거나 사용하지 않도록 설정합니다.
DdeFreeDataHandle DDE(동적 데이터 교환) 개체를 해제하고 개체와 연결된 데이터 핸들을 삭제합니다.
DdeFreeStringHandle 호출 애플리케이션에서 문자열 핸들을 해제합니다.
DdeGetData 지정된 DDE(동적 데이터 교환) 개체의 데이터를 지정된 로컬 버퍼에 복사합니다.
DdeGetLastError DDEML(동적 데이터 교환 관리 라이브러리) 함수의 실패로 설정된 가장 최근의 오류 코드를 검색하고 오류 코드를 DMLERR_NO_ERROR 다시 설정합니다.
DdeImpersonateClient DDE 클라이언트 대화에서 DDE(동적 데이터 교환) 클라이언트 애플리케이션을 가장합니다.

DdeInitializeA DDEML(동적 데이터 교환 관리 라이브러리)에 애플리케이션을 등록합니다. 애플리케이션은 다른 DDEML(동적 데이터 교환 관리 라이브러리) 함수를 호출하기 전에 이 함수를 호출해야 합니다. (ANSI)
DdeInitializeW DDEML(동적 데이터 교환 관리 라이브러리)에 애플리케이션을 등록합니다. 애플리케이션은 다른 DDEML(동적 데이터 교환 관리 라이브러리) 함수를 호출하기 전에 이 함수를 호출해야 합니다. (유니코드)
DdeKeepStringHandle 지정된 핸들과 연결된 사용 횟수를 증분합니다.
DdeNameService DDE(동적 데이터 교환) 서버에서 지원하는 서비스 이름을 등록하거나 등록 취소합니다.
DdePostAdvise 시스템이 지정된 토픽 및 항목에 대한 활성 조언 루프를 사용하여 각 클라이언트에 대한 호출(서버) 애플리케이션의 DDE(동적 데이터 교환) 콜백 함수에 XTYP_ADVREQ 트랜잭션을 보내도록 합니다.
DdeQueryConvInfo DDE(동적 데이터 교환) 트랜잭션 및 트랜잭션이 발생하는 대화에 대한 정보를 검색합니다.
DdeQueryNextServer 지정된 대화 목록에서 다음 대화 핸들을 검색합니다.
DdeQueryStringA 문자열 핸들과 연결된 텍스트를 버퍼에 복사합니다. (ANSI)
DdeQueryStringW 문자열 핸들과 연결된 텍스트를 버퍼에 복사합니다. (유니코드)
DdeReconnect 클라이언트 DDEML(동적 데이터 교환 관리 라이브러리) 애플리케이션이 클라이언트와의 대화를 종료한 서비스와의 대화를 다시 설정하려고 시도할 수 있도록 합니다.
DdeSetUserHandle 애플리케이션 정의 값을 대화 핸들 또는 트랜잭션 식별자와 연결합니다. 이는 비동기 트랜잭션 처

리를 간소화하는 데 유용합니다. 애플리케이션은 DdeQueryConvInfo 함수를 사용하여 이 값을 검색할 수 있습니다.
DdeUnaccessData DDE(동적 데이터 교환) 개체에 액세스하지 않습니다. 애플리케이션이 개체에 대한 액세스를 완료한 후 이 함수를 호출해야 합니다.
DdeUninitialize 호출 애플리케이션과 연결된 모든 DDEML(동적 데이터 교환 관리 라이브러리) 리소스를 해제합니다.

콜백 함수

PFNCALLBACK DDEML(동적 데이터 교환 관리 라이브러리) 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.

구조체

CONVCONTEXT DDE(동적 데이터 교환) 클라이언트 애플리케이션에서 제공하는 정보를 포함합니다. 이 정보는 특수 언어 또는 언어 간 DDE 대화에 유용합니다.
CONVINFO DDE(동적 데이터 교환) 대화에 대한 정보를 포함합니다.
DDEML_MSG_HOOK_DATA DDE(동적 데이터 교환) 메시지에 대한 정보를 포함하고 메시지에서 참조하는 데이터에 대한 읽기 권한을 제공합니다. 이 구조는 DDEML(동적 데이터 교환 관리 라이브러리) 모니터링 애플리케이션에서 사용됩니다.
HSZPAIR DDE 서비스 이름 및 토픽 이름을 포함합니다. DDE 서버 애플리케이션은 XTYP_WILDCONNECT 트랜잭션 중에 이 구조를 사용하여 지원하는 서비스 토픽 쌍을 열거할 수 있습니다.

MONCBSTRUCT 현재 DDE(동적 데이터 교환) 트랜잭션에 대한 정보를 포함합니다. DDE 디버깅 애플리케이션은 시스템이 다른 애플리케이션의 DDE 콜백 함수에 전달하는 트랜잭션을 모니터링할 때 이 구조를 사용할 수 있습니다.
MONCONVSTRUCT DDE(동적 데이터 교환) 대화에 대한 정보를 포함합니다. DDE 모니터링 애플리케이션은 이 구조를 사용하여 설정되었거나 종료된 대화에 대한 정보를 가져올 수 있습니다.
MONERRSTRUCT 현재 DDE(동적 데이터 교환) 오류에 대한 정보를 포함합니다. DDE 모니터링 애플리케이션은 이 구조를 사용하여 DDE 관리 라이브러리 함수에서 반환된 오류를 모니터링할 수 있습니다.
MONHSZSTRUCTA DDE(동적 데이터 교환) 문자열 핸들에 대한 정보를 포함합니다. DDE 모니터링 애플리케이션은 DDE 관리 라이브러리의 문자열 관리자 구성 요소의 작업을 모니터링할 때 이 구조를 사용할 수 있습니다. (ANSI)
MONHSZSTRUCTW DDE(동적 데이터 교환) 문자열 핸들에 대한 정보를 포함합니다. DDE 모니터링 애플리케이션은 DDE 관리 라이브러리의 문자열 관리자 구성 요소의 작업을 모니터링할 때 이 구조를 사용할 수 있습니다. (유니코드)
MONLINKSTRUCT DDE(동적 데이터 교환) advise 루프에 대한 정보를 포함합니다. DDE 모니터링 애플리케이션은 이 구조를 사용하여 시작되거나 종료된 advise 루프에 대한 정보를 얻을 수 있습니다.
MONMSGSTRUCT DDE(동적 데이터 교환) 메시지에 대한 정보를 포함합니다. DDE 모니터링 애플리케이션은 이 구조를 사용하여 전송되거나 게시된 DDE 메시지에 대한 정보를 가져올 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ddkernel.h 헤더

아티클2023. 08. 23.

이 헤더는 디스플레이 디바이스 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 디바이스 참조 표시

ddkernel.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDirectDrawKernel IDirectDrawKernel 인터페이스
IDirectDrawSurfaceKernel IDirectDrawSurfaceKernel 인터페이스

구조체

DDKERNELCAPS DDKERNELCAPS 구조체는 커널 모드 비디오 전송을 위한 미니포트 드라이버에 어떤 지원이 있는지 클라이언트에 알릴 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ddkmapi.h 헤더

아티클2023. 08. 23.

이 헤더는 디스플레이 디바이스 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 디바이스 참조 표시

ddkmapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

콜백 함수

LPDD_NOTIFYCALLBACK NotifyCallback 콜백 함수는 발생한 이벤트와 관련된 작업을 수행합니다.

구조체

DDADDVPCAPTUREBUFF DDADDVPCAPTUREBUFF 구조체에는 내부 캡처 큐에 새 버퍼를 추가하는 데 필요한 정보가 포함되어 있습니다.
DDCAPBUFFINFO DDCAPBUFFINFO 구조체에는 캡처 정보가 포함됩니다.
DDCLOSEHANDLE DDCLOSEHANDLE 구조체에는 Microsoft DirectDraw 개체, 표면, VPE(비디오 포트 확장) 개체 또는 VPE 캡처 핸들이 포함됩니다.
DDFLIPOVERLAY DDFLIPOVERLAY 구조체에는 대칭 이동에 필요한 표면 정보가 포함됩니다.
DDFLIPVIDEOPORT DDFLIPVIDEOPORT 구조체에는 하드웨어 비디오 포트를 대칭 이동하는 데 필요한 정보가 포함되어 있습니다.

DDGETAUTOFLIPIN DDGETAUTOFLIPIN 구조체에는 핸들 정보가 포함됩니다.
DDGETAUTOFLIPOUT DDGETAUTOFLIPOUT 구조체에는 DxApi 함수의 DD_DXAPI_GET_CURRENT_VP_AUTOFLIP_SURFACE 및 DD_DXAPI_GET_LAST_VP_AUTOFLIP_SURFACE 함수 식별자에서 반환된 핸들 및 극성 정보가 포함됩니다.
DDGETFIELDNUMIN DDGETFIELDNUMIN 구조에는 Microsoft DirectDraw 및 VPE(비디오 포트 확장) 개체 핸들 정보가 포함됩니다.
DDGETFIELDNUMOUT DDGETFIELDNUMOUT 구조체에는 하드웨어 비디오 포트의 필드 번호가 포함됩니다.
DDGETKERNELCAPSOUT DDGETKERNELCAPSOUT 구조체에는 Microsoft DirectDraw 개체의 기능이 포함되어 있습니다.
DDGETPOLARITYIN DDGETPOLARITYIN 구조체에는 Microsoft DirectDraw 및 VPE(비디오 포트 확장) 개체 핸들이 포함됩니다.
DDGETPOLARITYOUT DDGETPOLARITYOUT 구조체에는 요청된 극성 정보가 포함됩니다.
DDGETSURFACESTATEIN DDGETSURFACESTATEIN 구조체에는 Microsoft DirectDraw 및 DirectDraw 표면 핸들 정보가 포함됩니다.
DDGETSURFACESTATEOUT DDGETSURFACESTATEOUT 구조체에는 지정된 표면의 기능과 상태 포함됩니다.
DDGETVERSIONNUMBER DDGETVERSIONNUMBER 구조체에는 비디오 미니포트 드라이버의 DxApi 인터페이스에서 지원하는 Microsoft DirectDraw의 커널 모드 비디오 전송 구성 요소의 버전 번호가 포함되어 있습니다.
DDLOCKIN DDLOCKIN 구조체에는 Microsoft DirectDraw 개체 및 DirectDraw 표면 핸들 정보가 포함됩니다.

DDLOCKOUT DDLOCKOUT 구조체에는 표면에 대한 설명이 포함되어 있습니다.
DDOPENDIRECTDRAWIN DDOPENDIRECTDRAWIN 구조체에는 Microsoft DirectDraw 개체 정보가 포함됩니다.
DDOPENDIRECTDRAWOUT DDOPENDIRECTDRAWOUT의 ddRVal 멤버가 DD_OK 설정된 경우 DxApi 함수의 DD_DXAPI_OPENDIRECTDRAW 함수 식별자에 대한 새 Microsoft DirectDraw 핸들이 DDOPENDIRECTDRAWOUT 구조체에 포함됩니다.
DDOPENSURFACEIN DDOPENSURFACEIN 구조체에는 DirectDrawSurface 개체 정보가 포함됩니다.
DDOPENSURFACEOUT DDOPENSURFACEOUT의 ddRVal 멤버가 DD_OK 설정된 경우 DDOPENSURFACEOUT 구조체에는 새 DirectDrawSurface 핸들이 포함됩니다. 이 새 핸들은 DirectDrawSurface 핸들이 필요한 모든 후속 호출에서 사용해야 합니다.
DDOPENVIDEOPORTIN DDOPENVIDEOPORTIN 구조체에는 VPE(비디오 포트 확장) 개체 정보가 포함됩니다.
DDOPENVIDEOPORTOUT ddRVal이 DD_OK 설정된 경우 DDOPENVIDEOPORTOUT 구조체에는 Microsoft DirectDraw 반환 코드와 새 표면 핸들이 포함됩니다. 이 새 핸들은 VPE(비디오 포트 확장) 개체 핸들이 필요한 모든 후속 호출에서 사용해야 합니다.
DDOPENVPCAPTUREDEVICEIN DDOPENVPCAPTUREDEVICEIN 구조에는 VPE(비디오 포트 확장) 캡처 정보가 포함됩니다.
DDOPENVPCAPTUREDEVICEOUT DDOPENVPCAPTUREDEVICEOUT 구조체에는 VPE(비디오 포트 확장) 캡처 핸들이 포함되어 있습니다.
DDREGISTERCALLBACK DDREGISTERCALLBACK 구조체에는 레지스터 콜백 정보가 포함됩니다. 이 구조체는 DxApi 함수의 DD_DXAPI_REGISTER_CALLBACK 및 DD_DXAPI_UNREGISTER_CALLBACK 함수 식별자 모두에서 사용됩니다.

DDSETFIELDNUM DDSETFIELDNUM 구조체에는 핸들과 필드 번호가 포함됩니다.
DDSETSKIPFIELD DDSETSKIPFIELD 구조체에는 시작 필드 정보가 포함됩니다.
DDSETSURFACESTATE DDSETSURFACESTATE 구조체에는 표면 상태 정보가 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ddpbackup.h 헤더

아티클2023. 08. 23.

이 헤더는 데이터 중복 제거 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 중복 제거 API

ddpbackup.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDedupBackupSupport 데이터 중복 제거 재분석 지점, 메타데이터 및 컨테이너 파일의 복사본을 포함하는 백업 저장소에서 파일을 복원하는 방법을 제공합니다.
IDedupReadFileCallback 백업 애플리케이션에서 구현되는 콜백 인터페이스로, 데이터 중복 제거가 백업 저장소에 있는 메타데이터 및 컨테이너 파일에서 콘텐츠를 읽고 필요에 따라 복원 효율성을 향상시킬 수 있습니다.

구조체

DDP_FILE_EXTENT DDP_FILE_EXTENT ReadBackupFile에 대한 보류 중인 호출에서 읽을 파일의 데이터 범위를 나타냅니다.
DEDUP_CONTAINER_EXTENT 논리 컨테이너 파일은 백업 저장소의 단일 세그먼트 또는 여러 세그먼트에 저장될 수 있습니다.

열거형

DEDUP_BACKUP_SUPPORT_PARAM_TYPE

데이터 중복 제거가 최적화되지 않은 복원 또는 최적화된 복원을 수행해야 하는지 여부를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ddraw.h 헤더

아티클2023. 03. 16.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Directdraw
- Directshow

ddraw.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDirectDraw7 애플리케이션은 IDirectDraw7 인터페이스의 메서드를 사용하여 DirectDraw 개체를 만들고 시스템 수준 변수를 사용합니다. 이 섹션은 IDirectDraw7 인터페이스의 메서드에 대한 참조입니다.
IDirectDrawClipper 애플리케이션은 IDirectDrawClipper 인터페이스의 메서드를 사용하여 클립 목록을 관리합니다. 이 섹션은 이 인터페이스의 메서드에 대한 참조입니다.
IDirectDrawColorControl 애플리케이션은 IDirectDrawColorControl 인터페이스의 메서드를 사용하여 색 컨트롤을 가져와서 설정합니다.
IDirectDrawGammaControl 애플리케이션은 IDirectDrawGammaControl 인터페이스의 메서드를 사용하여 기본 표면의 빨간색, 녹색 및 파란색 감마 램프 수준을 조정합니다. 이 섹션은 이 인터페이스의 메서드에 대한 참조입니다.
IDirectDrawPalette 애플리케이션은 IDirectDrawPalette 인터페이스의 메서드를 사용하여 DirectDrawPalette 개체를 만들고 시스템 수준 변수를 사용합니다. 이 섹션은 이 인터페이스의 메서드에 대한 참조입니다.
IDirectDrawSurface7 애플리케이션은 IDirectDrawSurface7 인터페이스의 메서드를 사용하여 DirectDrawSurface 개체를 만들고 시스템 수준 변수를 사용합니다. 이 섹션은 이 인터페이스의 메서드에 대한 참조입니다.

함수

DirectDrawCreate DirectDraw 개체의 인스턴스를 만듭니다.
DirectDrawCreateClipper DirectDraw 개체와 연결되지 않은 DirectDrawClipper 개체의 인스턴스를 만듭니다.
DirectDrawCreateEx DirectX 7.0에서 Direct3D 인터페이스 집합을 지원하는 DirectDraw 개체의 인스턴스를 만듭니다. DirectX 7.0에서 Direct3D의 기능을 사용하려면 이 함수를 사용하여 DirectDraw 개체를 만듭니다.
DirectDrawEnumerateA 이 함수는 DirectDrawEnumerateEx 함수로 대체됩니다. (ANSI)
DirectDrawEnumerateExA 컴퓨터에 설치된 모든 DirectDraw 디바이스를 열거합니다. NULL 항목은 항상 GDI와 공유되는 기본 디스플레이 디바이스를 식별합니다. (ANSI)
DirectDrawEnumerateExW 컴퓨터에 설치된 모든 DirectDraw 디바이스를 열거합니다. NULL 항목은 항상 GDI와 공유되는 기본 디스플레이 디바이스를 식별합니다. (유니코드)
DirectDrawEnumerateW 이 함수는 DirectDrawEnumerateEx 함수로 대체됩니다. (유니코드)

콜백 함수

LPDDENUMCALLBACKA DDEnumCallback 함수는 DirectDrawEnumerate 함수에 대한 애플리케이션 정의 콜백 함수입니다. (ANSI)
LPDDENUMCALLBACKEXA DDEnumCallbackEx 함수는 DirectDrawEnumerateEx 함수에 대한 애플리케이션 정의 콜백 함수입니다. (ANSI)

LPDDENUMCALLBACKEXW DDEnumCallbackEx 함수는 DirectDrawEnumerateEx 함수에 대한 애플리케이션 정의 콜백 함수입니다. (유니코드)
LPDDENUMCALLBACKW DDEnumCallback 함수는 DirectDrawEnumerate 함수에 대한 애플리케이션 정의 콜백 함수입니다. (유니코드)
LPDDENUMMODESCALLBACK 사용하지 마십시오. 이 콜백 함수는 IDirectDraw7::EnumDisplayModes 메서드와 함께 사용되는 EnumModesCallback2 함수로 대체됩니다.
LPDDENUMMODESCALLBACK2 EnumModesCallback2 함수는 IDirectDraw7::EnumDisplayModes 메서드에 대한 애플리케이션 정의 콜백 함수입니다.
LPDDENUMSURFACESCALLBACK LPDDENUMSURFACESCALLBACK 함수(ddraw.h)는 EnumSurfacesCallback7 함수로 대체됩니다.
LPDDENUMSURFACESCALLBACK2 LPDDENUMSURFACESCALLBACK2 함수(ddraw.h)는 EnumSurfacesCallback7 함수로 대체됩니다.
LPDDENUMSURFACESCALLBACK7 EnumSurfacesCallback7 함수는 IDirectDrawSurface7::EnumAttachedSurfaces 및 IDirectDrawSurface7::EnumOverlayZOrders 메서드에 대한 애플리케이션 정의 콜백 함수입니다.

구조체

DDCOLORCONTROL DDCOLORCONTROL 구조체는 Microsoft *DirectDraw VPE(비디오 포트 확장)* 개체와 연결된 색 컨트롤을 정의합니다.
DDSURFACEDESC DDSURFACEDESC 구조체에는 드라이버에서 만들 서피스에 대한 설명이 포함되어 있습니다.

DDSURFACEDESC2 DDSURFACEDESC2 구조체에는 드라이버에서 만들 서피스에 대한 설명이 포함되어 있습니다.
DDBLTBATCH DDBLTBATCH 구조체는 비트 블록 전송(bitblt) 작업을 IDirectDrawSurface7::BltBatch 메서드에 전달합니다.
DDBLTFX DDBLTFX 구조체는 래스터 작업(ROP), 효과 및 재정의 정보를 IDirectDrawSurface7::Blt 메서드에 전달합니다. 이 구조체는 IDirectDrawSurface7::BltBatch 메서드와 함께 사용되는 DDBLTBATCH 구조체의 일부이기도 합니다.
DDCAPS_DX3 DDCAPS_DX3 구조체(ddraw.h)는 DirectDraw 개체를 통해 노출되는 하드웨어의 기능을 나타냅니다.
DDCAPS_DX5 DDCAPS_DX5 구조체(ddraw.h)는 DirectDraw 개체를 통해 노출되는 하드웨어의 기능을 나타냅니다.
DDCAPS_DX6 DDCAPS_DX6 구조체(ddraw.h)는 DirectDraw 개체를 통해 노출되는 하드웨어의 기능을 나타냅니다.
DDCAPS_DX7 DDCAPS_DX7 구조체(ddraw.h)는 DirectDraw 개체를 통해 노출되는 하드웨어의 기능을 나타냅니다.
DDCOLORKEY DDCOLORKEY 구조는 원본 색 키, 대상 색 키 또는 색 공간을 설명합니다.
DDDEVICEIDENTIFIER2 DDDEVICEIDENTIFIER2 구조체는 IDirectDraw7::GetDeviceIdentifier 메서드에 전달되어 디바이스에 대한 정보를 가져옵니다.
DDGAMMARAMP DDGAMMARAMP 구조체에는 IDirectDrawGammaControl::GetGammaRamp 및 IDirectDrawGammaControl::SetGammaRamp 메서드에 대한 빨간색, 녹색 및 파란색 램프 데이터가 포함됩니다.

DDOVERLAYFX DDOVERLAYFX 구조체는 오버레이 정보를 IDirectDrawSurface7::UpdateOverlay 메서드에 전달합니다.
DDPIXELFORMAT DDPIXELFORMAT 구조체는 IDirectDrawSurface7::GetPixelFormat 메서드에 대한 DirectDrawSurface 개체의 픽셀 형식을 설명합니다.
DDSCAPS DDSCAPS 구조체는 Microsoft DirectDraw 표면 개체의 기능을 정의합니다.
DDSCAPS2 DDSCAPS2 구조체는 Microsoft DirectDraw 표면 개체의 추가 기능을 정의합니다.

ddrawgdi.h 헤더

아티클2023. 08. 23.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

ddrawgdi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DdAttachSurface DdAttachSurface 함수는 두 개의 커널 모드 표면 표현을 연결합니다. GdiEntry11은 이 함수의 별칭으로 정의됩니다.
DdCreateDIBSection 해당 색 테이블을 디바이스와 공유하는 DIBSECTION 구조를 만듭니다. GdiEntry9는 이 함수의 별칭으로 정의됩니다.
DdCreateDirectDrawObject NtGdiDdCreateDirectDrawObject 함수에 대한 래퍼이며 Microsoft DirectDraw 개체의 커널 쪽 표현을 만듭니다.
DdCreateSurfaceObject NtGdiDdCreateSurfaceObject 함수에 대한 래퍼이며 커널 모드 표면 개체를 만듭니다. GdiEntry4는 이 함수의 별칭으로 정의됩니다.
DdDeleteDirectDrawObject NtGdiDdDeleteDirectDrawObject 함수에 대한 래퍼이며 이전에 DdCreateDirectDrawObject를 사용하여 만든 커널 모드 Microsoft DirectDraw 개체를 삭제합니다. GdiEntry3은 이 함수의 별칭으로 정의됩니다.
DdDeleteSurfaceObject NtGdiDdDeleteSurfaceObject 함수에 대한 래퍼이며 이전에 NtGdiDdCreateSurfaceObject에서 만든 커널 모드 표면 개체를 삭제합니다. GdiEntry5는 이 함수의 별칭으로 정의됩니다.
DdGetDC

NtGdiDdGetDC 함수에 대한 래퍼이며 표시된 Microsoft DirectDraw 표면을 나타내는 Windows GDI(GDI) 디바이스 컨텍스트(DC)를 반환합니다. GdiEntry7은 이 함수의 별칭으로 정의됩니다.
DdGetDxHandle DirectX API 메커니즘을 제어하는 커널 모드 진입점에 대한 후속 호출에 사용할 커널 모드 Microsoft DirectX API 핸들을 반환합니다.
DdQueryDirectDrawObject NtGdiDdQueryDirectDrawObject 함수에 대한 래퍼이며 이전에 만든 커널 모드 표현에서 기능을 쿼리합니다. GdiEntry2는 이 함수의 별칭으로 정의됩니다.
DdQueryDisplaySettingsUniqueness 데스크톱 스위치, 빠른 사용자 스위치 또는 전체 화면 Microsoft MS-DOS 상자가 있는 경우와 같이 모드 전환이 발생할 때마다 증가되는 정수의 현재 값을 반환합니다.
DdReenableDirectDrawObject NtGdiDdReenableDirectDrawObject 함수에 대한 래퍼입니다.
DdReleaseDC NtGdiDdReleaseDC 함수에 대한 래퍼이며 이전에 DdGetDC 또는 GdiEntry7을 통해 가져온 DC(디바이스 컨텍스트)를 해제합니다. GdiEntry8은 이 함수의 별칭으로 정의됩니다.
DdResetVisrgn NtGdiDdResetVisrgn 함수에 대한 래퍼이며 데스크톱의 창에 대한 클리핑 영역에서 적시에 사용자 모드 정보를 사용하도록 설정합니다.
DdSetGammaRamp DdSetGammaRamp 함수는 디바이스의 감마 램프를 설정합니다.
DdSwapTextureHandles Microsoft DirectDraw 7.0 이전의 디바이스 DDI(드라이버 인터페이스)를 위해 개발되었으며 Microsoft Windows NT 시스템에서는 아무 작업도 수행하지 않습니다. 모든 매개 변수는 무시됩니다. GdiEntry16은 이 함수의 별칭으로 정의됩니다.
DdUnattachSurface DdUnattachSurface 함수는 두 커널 모드 표면 개체 간에 DdAttachSurface로 만든 첨부 파일을 제거합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ddrawi.h 헤더

아티클2023. 08. 23.

이 헤더는 디스플레이 디바이스 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 디스플레이 디바이스 참조

ddrawi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

DDHAL_DESTROYDDLOCALDATA DDHAL_DESTROYDDLOCALDATA 드라이버가 지정된 로컬 DirectDraw 개체에 연결된 표면 집합을 삭제하는 데 필요한 정보를 포함합니다.
DDHAL_WAITFORVERTICALBLANKDATA DD_WAITFORVERTICALBLANKDATA 구조에는 모니터의 세로 빈 정보를 가져오는 데 필요한 정보가 포함되어 있습니다. (DDHAL_WAITFORVERTICALBLANKDATA)
DDCORECAPS DDCORECAPS 구조는 DirectDraw 개체를 통해 애플리케이션에 노출되는 Microsoft DirectDraw 드라이버 및 해당 디바이스의 핵심 기능을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ddrawint.h 헤더

아티클2023. 08. 23.

이 헤더는 디스플레이 디바이스 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 디바이스 참조 표시

ddrawint.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

콜백 함수

PDD_CANCREATESURFACE CanCreateD3DBuffer 콜백 함수는 드라이버가 지정된 설명의 드라이버 수준 명령 또는 꼭짓점 버퍼를 만들 수 있는지 여부를 결정합니다.
PDD_COLORCB_COLORCONTROL DdControlColor 콜백 함수는 오버레이 표면의 광도 및 밝기 컨트롤을 제어합니다.
PDD_CREATEPALETTE DdCreatePalette 콜백 함수는 지정된 DirectDraw 개체에 대한 DirectDrawPalette 개체를 만듭니다.
PDD_CREATESURFACE CreateD3DBuffer 콜백 함수는 지정된 설명의 드라이버 수준 명령 또는 꼭짓점 버퍼를 만드는 데 사용됩니다.
PDD_CREATESURFACEEX D3dCreateSurfaceEx 함수는 Direct3D 렌더링을 위해 표면을 설정할 수 있도록 Microsoft DirectDraw 표면과 Microsoft Direct3D 핸들 값의 연결에 대해 알립니다.
PDD_DESTROYDDLOCAL D3dDestroyDDLocal 함수는 동일한 지정된 로컬 Microsoft DirectDraw 개체에 속하는 D3dCreateSurfaceEx 함수에서 이전에 만든 모든 Microsoft Direct3D 표면을 삭제합니다.
PDD_FLIPTOGDISURFACE DdFlipToGDISurface 콜백 함수는 DirectDraw가 GDI 표면으로 또는 GDI 표면에서 대칭 이동 중일 때 드라이버에 알립니다.

PDD_FREEDRIVERMEMORY DdFreeDriverMemory 콜백 함수는 오프스크린 또는 비로컬 디스플레이 메모리를 해제하여 새 할당 요청을 충족합니다.
PDD_GETAVAILDRIVERMEMORY DdGetAvailDriverMemory 콜백 함수는 드라이버 관리 메모리 힙의 사용 가능한 메모리 양을 쿼리합니다.
PDD_GETDRIVERINFO DdGetDriverInfo 함수는 드라이버에서 지원하는 추가 DirectDraw 및 Direct3D 기능을 드라이버에 쿼리합니다.
PDD_GETDRIVERSTATE D3dGetDriverState 함수는 Microsoft DirectDraw 및 Microsoft Direct3D 런타임 모두에서 드라이버에서 현재 상태에 대한 정보를 가져오는 데 사용됩니다.
PDD_GETSCANLINE DdGetScanLine 콜백 함수는 현재 실제 검사 줄의 수를 반환합니다.
PDD_KERNELCB_SYNCSURFACE DdSyncSurfaceData 콜백 함수는 비디오 미니포트 드라이버에 전달되기 전에 표면 데이터를 설정하고 수정합니다.
PDD_KERNELCB_SYNCVIDEOPORT DdSyncVideoPortData 콜백 함수는 비디오 미니포트 드라이버에 전달되기 전에 VPE 개체 데이터를 설정하고 수정합니다.
PDD_MAPMEMORY DdMapMemory 콜백 함수는 프레임 버퍼의 애플리케이션 수정 가능 부분을 지정된 프로세스의 사용자 모드 주소 공간에 매핑하거나 메모리 매핑을 해제합니다.
PDD_MOCOMPCB_BEGINFRAME DdMoCompBeginFrame 콜백 함수는 새 프레임 디코딩을 시작합니다.
PDD_MOCOMPCB_CREATE DdMoCompCreate 콜백 함수는 드라이버에 소프트웨어 디코더가 지정된 GUID로 동작 보정을 사용하기 시작하도록 알 수 있습니다.

PDD_MOCOMPCB_DESTROY DdMoCompDestroy 콜백 함수는 드라이버에 이 동작 보정 개체가 더 이상 사용되지 않음을 알 수 있습니다. 이제 드라이버는 필요한 정리를 수행해야 합니다.
PDD_MOCOMPCB_ENDFRAME DdMoCompEndFrame 콜백 함수는 디코딩된 프레임을 완료합니다.
PDD_MOCOMPCB_GETCOMPBUFFINFO DDMoCompGetBuffInfo 콜백 함수를 사용하면 드라이버가 지정된 GUID를 지원하는 데 필요한 중간 표면 수와 이러한 각 표면의 크기, 위치 및 형식을 지정할 수 있습니다.
PDD_MOCOMPCB_GETFORMATS DdMoCompGetFormats 콜백 함수는 하드웨어가 데이터를 디코딩할 수 있는 압축되지 않은 형식을 나타냅니다.
PDD_MOCOMPCB_GETGUIDS DdMoCompGetGuids 콜백 함수는 드라이버가 지원하는 GUID 수를 검색합니다.
PDD_MOCOMPCB_GETINTERNALINFO DdMoCompGetInternalInfo 콜백 함수를 사용하면 드라이버가 내부적으로 디스플레이 메모리를 할당하여 동작 보정을 수행한다고 보고할 수 있습니다.
PDD_MOCOMPCB_QUERYSTATUS DdMoCompQueryStatus 콜백 함수는 가장 최근의 렌더링 작업의 상태 지정된 화면에 쿼리합니다.
PDD_MOCOMPCB_RENDER DdMoCompRender 콜백 함수는 매크로 블록이 포함된 표면, 매크로 블록이 있는 각 표면의 오프셋 및 렌더링할 macroblock 데이터의 크기를 지정하여 렌더링할 macroblock을 드라이버에 알려줍니다.
PDD_PALCB_DESTROYPALETTE DdDestroyPalette 콜백 함수는 지정된 팔레트를 삭제합니다.
PDD_PALCB_SETENTRIES DdSetEntries 콜백 함수는 지정된 색상표의 색상표 항목을 업데이트합니다.
PDD_SETEXCLUSIVEMODE

DdSetExclusiveMode 콜백 함수는 DirectDraw 애플리케이션이 배타적 모드로 전환할 때 드라이버에 알 수 있습니다.
PDD_SURFCB_ADDATTACHEDSURFACE DdAddAttachedSurface 콜백 함수는 표면을 다른 표면에 연결합니다.
PDD_SURFCB_BLT DdBlt 콜백 함수는 비트 블록 전송을 수행합니다.
PDD_SURFCB_DESTROYSURFACE DdDestroySurface 콜백 함수는 DirectDraw 표면을 삭제합니다.
PDD_SURFCB_FLIP DdFlip 콜백 함수는 대상 표면과 연결된 표면 메모리가 기본 표면이 되고 현재 표면이 기본 표면이 되지 않습니다.
PDD_SURFCB_GETBLTSTATUS DdGetBltStatus 콜백 함수는 지정된 표면의 blit 상태 쿼리합니다.
PDD_SURFCB_GETFLIPSTATUS DdGetFlipStatus 콜백 함수는 표면에서 가장 최근에 요청된 대칭 이동이 발생했는지 여부를 결정합니다.
PDD_SURFCB_LOCK DdLock 콜백 함수는 지정된 표면 메모리 영역을 잠그고 표면과 연결된 메모리 블록에 대한 유효한 포인터를 제공합니다.
PDD_SURFCB_SETCOLORKEY DdSetColorKey 콜백 함수는 지정된 표면의 색 키 값을 설정합니다.
PDD_SURFCB_SETOVERLAYPOSITION DdSetOverlayPosition 콜백 함수는 오버레이의 위치를 설정합니다.
PDD_SURFCB_SETPALETTE DdSetPalette 콜백 함수는 지정된 표면에 팔레트를 연결합니다.
PDD_SURFCB_UNLOCK DdUnLock 콜백 함수는 지정된 표면에 보관된 잠금을 해제합니다.

PDD_SURFCB_UPDATEOVERLAY DdUpdateOverlay 콜백 함수는 오버레이 표면의 시각적 특성을 변경하거나 수정합니다.
PDD_VPORTCB_CANCREATEVIDEOPORT DdVideoPortCanCreate 콜백 함수는 드라이버가 지정된 설명의 DirectDraw VPE 개체를 지원할 수 있는지 여부를 결정합니다.
PDD_VPORTCB_COLORCONTROL DdVideoPortColorControl 콜백 함수는 VPE 개체 색 컨트롤을 가져오거나 설정합니다.
PDD_VPORTCB_CREATEVIDEOPORT DdVideoPortCreate 콜백 함수는 DirectDraw가 VPE 개체를 만들었다는 사실을 드라이버에 알립니다.
PDD_VPORTCB_DESTROYVPORT DdVideoPortDestroy 콜백 함수는 DirectDraw가 지정된 VPE 개체를 제거했음을 드라이버에 알립니다.
PDD_VPORTCB_FLIP DdVideoPortFlip 콜백 함수는 물리적 대칭 이동을 수행하여 VPE 개체가 새 표면에 데이터를 쓰기 시작합니다.
PDD_VPORTCB_GETBANDWIDTH DdVideoPortGetBandwidth 콜백 함수는 지정된 VPE 개체 출력 형식을 기반으로 디바이스의 프레임 버퍼 메모리의 대역폭 제한을 보고합니다.
PDD_VPORTCB_GETFIELD DdVideoPortGetField 콜백 함수는 인터레이스된 신호의 현재 필드가 짝수인지 홀수인지를 결정합니다.
PDD_VPORTCB_GETFLIPSTATUS DdVideoPortGetFlipStatus 콜백 함수는 표면에서 가장 최근에 요청한 대칭 이동이 발생했는지 여부를 결정합니다.
PDD_VPORTCB_GETINPUTFORMATS DdVideoPortGetInputFormats 콜백 함수는 DirectDraw VPE 개체가 허용할 수 있는 입력 형식을 결정합니다.

PDD_VPORTCB_GETLINE DdVideoPortGetLine 콜백 함수는 하드웨어 비디오 포트의 현재 줄 번호를 반환합니다.
PDD_VPORTCB_GETOUTPUTFORMATS DdVideoPortGetOutputFormats 콜백 함수는 VPE 개체가 지원하는 출력 형식을 결정합니다.
PDD_VPORTCB_GETSIGNALSTATUS DdVideoPortGetSignalStatus 콜백 함수는 하드웨어 비디오 포트에 현재 표시되는 비디오 신호의 상태 검색합니다.
PDD_VPORTCB_GETVPORTCONNECT DdVideoPortGetConnectInfo 콜백 함수는 지정된 VPE 개체에서 지원하는 연결을 반환합니다.
PDD_VPORTCB_UPDATE DdVideoPortUpdate 콜백 함수는 VPE 개체를 시작 및 중지하고 VPE 개체 데이터 스트림을 수정합니다.
PDD_VPORTCB_WAITFORSYNC DdVideoPortWaitForSync 콜백 함수는 다음 세로 동기화가 발생할 때까지 기다립니다.
PDD_WAITFORVERTICALBLANK DdWaitForVerticalBlank 콜백 함수는 디바이스의 세로 빈 상태 반환합니다.

구조체

DD_ADDATTACHEDSURFACEDATA DD_ADDATTACHEDSURFACEDATA 구조체에는 표면을 다른 표면에 연결하는 데 필요한 정보가 포함되어 있습니다.
DD_ATTACHLIST DD_ATTACHLIST 구조체는 Microsoft DirectDraw에 연결된 표면 목록을 유지 관리합니다.
DD_BEGINMOCOMPFRAMEDATA DDHAL_BEGINMOCOMPFRAMEDATA 구조체에는 디코딩을 시작하는 데 필요한 프레임 정보가 포함됩니다.

DD_BLTDATA DD_BLTDATA 구조에는 비트 블록 전송을 수행하기 위한 드라이버와 관련된 정보가 포함됩니다.
DD_CANCREATESURFACEDATA DD_CANCREATESURFACEDATA 구조체에는 버퍼인 CanCreateD3DBuffer의 경우 표면을 만들 수 있는지 여부를 나타내는 데 필요한 정보가 포함되어 있습니다.
DD_CANCREATEVPORTDATA DD_CANCREATEVPORTDATA 구조에는 드라이버가 VPE(비디오 포트 확장) 개체를 만들 수 있는지 여부를 확인하는 데 필요한 정보가 포함되어 있습니다.
DD_CLIPPER_GLOBAL DD_CLIPPER_GLOBAL 구조에는 개체 인스턴스 간에 공유할 수 있는 전역 DirectDrawClipper 데이터가 포함됩니다.
DD_CLIPPER_LOCAL DD_CLIPPER_LOCAL 구조에는 각 개별 DirectDrawClipper 개체에 대한 로컬 데이터가 포함됩니다.
DD_COLORCONTROLDATA DD_COLORCONTROLDATA 구조체에는 지정된 오버레이에 대한 색 제어 정보가 포함됩니다.
DD_CREATEMOCOMPDATA DD_CREATEMOCOMPDATA 구조에는 동작 보정 사용을 시작하는 데 필요한 데이터가 포함됩니다.
DD_CREATEPALETTEDATA DD_CREATEPALETTEDATA 구조에는 이 Microsoft DirectDraw 개체에 대한 DirectDrawPalette 개체를 만드는 데 필요한 정보가 포함되어 있습니다.
DD_CREATESURFACEDATA DD_CREATESURFACEDATA 구조체에는 명령 또는 꼭짓점 버퍼인 CreateD3DBuffer의 경우 표면을 만드는 데 필요한 정보가 포함되어 있습니다.
DD_CREATESURFACEEXDATA DD_CREATESURFACEEXDATA 구조에는 드라이버가 표면을 만들고 제공된 텍스처 핸들과 연결하는 데 필요한 정보가 포함되어 있습니다.
DD_CREATEVPORTDATA

DD_CREATEVPORTDATA 구조에는 생성되는 VPE(비디오 포트 확장) 개체를 설명하는 데 필요한 정보가 포함되어 있습니다.
DD_DESTROYMOCOMPDATA DD_DESTROYMOCOMPDATA 구조에는 동작 보정 수행을 완료하는 데 필요한 정보가 포함되어 있습니다.
DD_DESTROYPALETTEDATA DD_DESTROYPALETTEDATA 구조체에는 지정된 팔레트를 삭제하는 데 필요한 정보가 포함되어 있습니다.
DD_DESTROYSURFACEDATA DD_DESTROYSURFACEDATA 구조체에는 명령 또는 꼭짓점 버퍼인 DestroyD3DBuffer의 경우 지정된 표면을 삭제하는 데 필요한 정보가 포함됩니다.
DD_DESTROYVPORTDATA DD_DESTROYVPORTDATA 구조에는 드라이버가 클린 데 필요한 정보가 포함되어 있습니다.
DD_DIRECTDRAW_GLOBAL DD_DIRECTDRAW_GLOBAL 구조에는 드라이버의 디바이스를 설명하는 드라이버 정보가 포함되어 있습니다.
DD_DIRECTDRAW_LOCAL DD_DIRECTDRAW_LOCAL 구조에는 현재 DirectDraw 프로세스와 관련된 드라이버 정보만 포함됩니다.
DD_ENDMOCOMPFRAMEDATA DD_ENDMOCOMPFRAMEDATA 구조에는 디코딩된 프레임을 완료하는 데 필요한 정보가 포함되어 있습니다.
DD_FLIPDATA DD_FLIPDATA 구조에는 대칭 이동에 필요한 정보가 포함되어 있습니다.
DD_FLIPTOGDISURFACEDATA DD_FLIPTOGDISURFACEDATA 구조에는 GDI 표면 알림 정보가 포함됩니다.
DD_FLIPVPORTDATA DD_FLIPVPORTDATA 구조에는 VPE(비디오 포트 확장) 개체가 대칭 이동을 수행하는 데 필요한 정보가 포함되어 있습니다.

DD_FREEDRIVERMEMORYDATA DD_FREEDRIVERMEMORYDATA 구조에는 무료 요청의 세부 정보가 포함됩니다.
DD_GETAVAILDRIVERMEMORYDATA DD_GETAVAILDRIVERMEMORYDATA 구조에는 드라이버가 사용 가능한 메모리 양을 쿼리하고 반환하는 데 필요한 정보가 포함되어 있습니다.
DD_GETBLTSTATUSDATA DD_GETBLTSTATUSDATA 구조체는 blit 상태 정보를 반환합니다.
DD_GETDRIVERINFODATA DD_GETDRIVERINFODATA 구조는 DdGetDriverInfo 콜백 루틴에 데이터를 전달하는 데 사용됩니다.
DD_GETDRIVERSTATEDATA DD_GETDRIVERSTATEDATA 구조체는 드라이버의 상태를 설명합니다.
DD_GETFLIPSTATUSDATA DD_GETFLIPSTATUSDATA 구조체는 대칭 이동 상태 정보를 반환합니다.
DD_GETINTERNALMOCOMPDATA DD_GETINTERNALMOCOMPDATA 구조에는 내부 메모리 요구 사항이 포함됩니다.
DD_GETMOCOMPCOMPBUFFDATA DD_GETMOCOMPCOMPBUFFDATA 구조에는 압축된 버퍼 정보가 포함됩니다.
DD_GETMOCOMPFORMATSDATA DD_GETMOCOMPFORMATSDATA 구조체에는 압축되지 않은 형식 정보가 포함됩니다.
DD_GETMOCOMPGUIDSDATA DD_GETMOCOMPGUIDSDATA 구조에는 동작 보정 GUID 정보가 포함됩니다.
DD_GETSCANLINEDATA DD_GETSCANLINEDATA 구조에는 현재 검사 줄의 수를 쿼리하고 반환하는 데 필요한 멤버가 포함됩니다.
DD_GETVPORTBANDWIDTHDATA DD_GETVPORTBANDWIDTHDATA 구조체에는 지정된 형식에 대한 대역폭 정보가 포함됩니다.

DD_GETVPORTCONNECTDATA DD_GETVPORTCONNECTDATA 구조에는 지정된 VPE(비디오 포트 확장) 개체에서 지원하는 연결 조합이 포함되어 있습니다.
DD_GETVPORTFIELDDATA DD_GETVPORTFIELDDATA 구조체에는 드라이버가 인터레이스된 신호의 현재 필드가 짝수인지 홀수인지 확인하는 데 필요한 정보가 포함되어 있습니다.
DD_GETVPORTFLIPSTATUSDATA DD_GETVPORTFLIPSTATUSDATA 구조체에는 지정된 표면에 대한 대칭 이동 상태 정보가 포함됩니다.
DD_GETVPORTINPUTFORMATDATA DD_GETVPORTINPUTFORMATDATA 구조체에는 드라이버가 VPE(비디오 포트 확장) 개체가 허용할 수 있는 입력 형식을 반환하는 데 필요한 정보가 포함되어 있습니다.
DD_GETVPORTLINEDATA DD_GETVPORTLINEDATA 구조체에는 하드웨어 비디오 포트의 현재 줄 번호가 포함됩니다.
DD_GETVPORTOUTPUTFORMATDATA DD_GETVPORTOUTPUTFORMATDATA 구조체에는 드라이버가 VPE(비디오 포트 확장) 개체가 지정된 입력 형식에 대해 지원하는 모든 출력 형식을 반환하는 데 필요한 정보가 포함되어 있습니다.
DD_GETVPORTSIGNALDATA DD_GETVPORTSIGNALDATA 구조에는 하드웨어 비디오 포트의 신호 상태 포함됩니다.
DD_LOCKDATA DD_LOCKDATA 구조에는 Microsoft DirectDraw 매개 변수 구조에 정의된 잠금을 수행하는 데 필요한 정보가 포함되어 있습니다.
DD_MAPMEMORYDATA DD_MAPMEMORYDATA 구조에는 프레임 버퍼를 사용자 모드 메모리에 매핑하거나 매핑 해제하는 데 필요한 정보가 포함되어 있습니다.
DD_MORESURFACECAPS DD_MORESURFACECAPS 구조는 DDCORECAPS에 설명된 기능 외에도 더 많은 드라이버 표면 기능을 정의합니다.

DD_MOTIONCOMP_LOCAL DD_MOTIONCOMP_LOCAL 구조에는 각 개별 Microsoft DirectDraw 동작 보정 개체에 대한 로컬 데이터가 포함됩니다.
DD_PALETTE_GLOBAL DD_PALETTE_GLOBAL 구조체에는 개체 인스턴스 간에 공유할 수 있는 전역 DirectDrawPalette 데이터가 포함됩니다.
DD_PALETTE_LOCAL DD_PALETTE_LOCAL 구조체에는 개별 색상표 개체에 고유한 색상표 관련 데이터가 포함되어 있습니다.
DD_QUERYMOCOMPSTATUSDATA DD_QUERYMOCOMPSTATUSDATA 구조에는 이전 프레임의 상태 쿼리하는 데 필요한 정보가 포함되어 있습니다.
DD_RENDERMOCOMPDATA DD_RENDERMOCOMPDATA 구조체에는 프레임을 렌더링하는 데 필요한 정보가 포함됩니다.
DD_SETCOLORKEYDATA DD_SETCOLORKEYDATA 구조체에는 지정된 표면의 색 키 값을 설정하는 데 필요한 정보가 포함되어 있습니다.
DD_SETENTRIESDATA DD_SETENTRIESDATA 구조에는 팔레트 항목을 설정하는 데 필요한 정보가 포함되어 있습니다.
DD_SETEXCLUSIVEMODEDATA DD_SETEXCLUSIVEMODEDATA 구조에는 전용 모드 알림 정보가 포함됩니다.
DD_SETOVERLAYPOSITIONDATA DD_SETOVERLAYPOSITIONDATA 구조체에는 오버레이 표면의 표시 좌표를 변경하는 데 필요한 정보가 포함되어 있습니다.
DD_SETPALETTEDATA DD_SETPALETTEDATA 구조체에는 특정 표면에 대한 팔레트를 설정하는 데 필요한 정보가 포함되어 있습니다.
DD_STEREOMODE

DD_STEREOMODE 구조는 런타임에서 DdGetDriverInfo 호출에서 GUID_DDStereoMode 사용하여 드라이버가 지정된 비디오 디스플레이 모드에 대해 스테레오를 지원하는지 여부를 쿼리하는 데 사용됩니다.
DD_SURFACE_GLOBAL DD_SURFACE_GLOBAL 구조에는 여러 표면 간에 공유할 수 있는 전역 표면 관련 데이터가 포함되어 있습니다.
DD_SURFACE_INT DD_SURFACE_INT 구조체에는 DirectDrawSurface 개체의 인터페이스 정보가 포함됩니다.
DD_SURFACE_LOCAL DD_SURFACE_LOCAL 구조체에는 개별 표면 개체에 고유한 표면 관련 데이터가 포함되어 있습니다.
DD_SURFACE_MORE DD_SURFACE_MORE 구조에는 각 개별 DirectDrawSurface 개체에 대한 추가 로컬 데이터가 포함됩니다.
DD_SYNCSURFACEDATA DD_SYNCSURFACEDATA 구조체에는 표면 정보가 포함됩니다.
DD_SYNCVIDEOPORTDATA DD_SYNCVIDEOPORTDATA 구조에는 VPE(비디오 포트 확장) 개체 정보가 포함됩니다.
DD_UNLOCKDATA DD_UNLOCKDATA 구조에는 Microsoft DirectDraw 매개 변수 구조에 정의된 잠금 해제를 수행하는 데 필요한 정보가 포함되어 있습니다.
DD_UPDATENONLOCALHEAPDATA DD_UPDATENONLOCALHEAPDATA 구조에는 필요한 힙 정보가 포함됩니다.
DD_UPDATEOVERLAYDATA DD_UPDATEOVERLAYDATA 구조체에는 오버레이 표면을 업데이트하는 데 필요한 정보가 포함되어 있습니다.
DD_UPDATEVPORTDATA DD_UPDATEVPORTDATA 구조에는 VPE(비디오 포트 확장) 개체를 시작, 중지 및 변경하는 데 필요한 정보가 포함되어 있습니다.

DD_VIDEOPORT_LOCAL DD_VIDEOPORT_LOCAL 구조에는 개별 Microsoft DirectDraw VPE 개체에 고유한 VPE(비디오 포트 확장) 관련 데이터가 포함되어 있습니다.
DD_VPORTCOLORDATA DD_VPORTCOLORDATA 구조에는 VPE(비디오 포트 확장) 개체 색 제어 정보가 포함됩니다.
DD_WAITFORVERTICALBLANKDATA DD_WAITFORVERTICALBLANKDATA 구조에는 모니터의 세로 빈 정보를 가져오는 데 필요한 정보가 포함되어 있습니다. D
DD_WAITFORVPORTSYNCDATA DD_WAITFORVPORTSYNCDATA 구조에는 드라이버가 VPE(비디오 포트 확장) 개체를 동기화하는 데 필요한 정보가 포함되어 있습니다.
DD_CALLBACKS DD_CALLBACKS 구조에는 디바이스 드라이버가 지원하는 콜백 함수에 대한 진입 포인터가 포함되어 있습니다.
DD_COLORCONTROLCALLBACKS DD_COLORCONTROLCALLBACKS 구조에는 디바이스 드라이버가 지원하는 Microsoft DirectDraw 색 제어 콜백에 대한 항목 포인터가 포함되어 있습니다.
DD_D3DBUFCALLBACKS DD_D3DBUFCALLBACKS 구조는 명령 및 꼭짓점 버퍼의 드라이버 수준 할당을 구현하는 드라이버에서만 사용됩니다.
DD_HALINFO DD_HALINFO 구조는 하드웨어 및 드라이버의 기능을 설명합니다.
DD_KERNELCALLBACKS DD_KERNELCALLBACKS 구조체에는 드라이버가 지원하는 DirectDraw 커널 모드 콜백 함수에 대한 진입 포인터가 포함되어 있습니다.
DD_MISCELLANEOUS2CALLBACKS DD_MISCELLANEOUS2CALLBACKS 구조체는 기타 콜백 루틴의 주소를 반환하는 데 사용됩니다.
DD_MISCELLANEOUSCALLBACKS

DD_MISCELLANEOUSCALLBACKS 구조에는 디바이스 드라이버가 지원하는 메모리 쿼리 콜백에 대한 항목 포인터가 포함되어 있습니다.
DD_MOTIONCOMPCALLBACKS DD_MOTIONCOMPCALLBACKS 구조에는 디바이스 드라이버가 지원하는 동작 보정 콜백 함수에 대한 진입 포인터가 포함되어 있습니다.
DD_NONLOCALVIDMEMCAPS DD_NONLOCALVIDMEMCAPS 구조에는 비로컬 표시 메모리에 대한 기능이 포함되어 있습니다.
DD_NTCALLBACKS DD_NTCALLBACKS 구조에는 디바이스 드라이버에서 지원하는 Microsoft Windows 2000 이상 Microsoft DirectDraw 콜백 함수에 대한 항목 포인터가 포함되어 있습니다.
DD_NTPRIVATEDRIVERCAPS DD_NTPRIVATEDRIVERCAPS 구조를 사용하면 DirectDraw에서 표면을 만들 때 드라이버가 Microsoft DirectDraw의 동작을 변경할 수 있습니다.
DD_PALETTECALLBACKS DD_PALETTECALLBACKS 구조에는 디바이스 드라이버가 지원하는 DirectDraw 팔레트 콜백 함수에 대한 항목 포인터가 포함되어 있습니다.
DD_SURFACECALLBACKS DD_SURFACECALLBACKS 구조에는 디바이스 드라이버가 지원하는 Microsoft DirectDraw 표면 콜백 함수에 대한 진입 포인터가 포함되어 있습니다.
DD_VIDEOPORTCALLBACKS DD_VIDEOPORTCALLBACKS 구조에는 디바이스 드라이버에서 지원하는 Microsoft DirectDraw VPE(비디오 포트 확장) 콜백 함수에 대한 진입 포인터가 포함되어 있습니다.
DDCOMPBUFFERINFO DDCOMPBUFFERINFO 구조체에는 압축 버퍼에 대한 드라이버 제공 정보가 포함되어 있습니다.
DDMOCOMPBUFFERINFO DDMOCOMPBUFFERINFO 구조체는 프레임을 렌더링하는 데 필요한 매크로 블록 정보를 포함하고 이 정보를 DD_RENDERMOCOMPDATA 구조체에 전달합니다.
VIDEOMEMORY VIDEOMEMORY 구조를 사용하면 드라이버가 디스플레이 메모리를 힙으로 관리할 수 있습니다.

VIDEOMEMORYINFO VIDEOMEMORYINFO 구조체는 디스플레이 메모리의 일반적인 형식을 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

debugapi.h 헤더

아티클2023. 08. 23.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

debugapi.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CheckRemoteDebuggerPresent 지정된 프로세스가 디버깅되고 있는지 여부를 결정합니다.
ContinueDebugEvent 디버거가 이전에 디버깅 이벤트를 보고한 스레드를 계속할 수 있도록 합니다.
DebugActiveProcess 디버거가 활성 프로세스에 연결하고 디버그할 수 있도록 합니다.
DebugActiveProcessStop 디버거가 지정된 프로세스를 디버깅하지 못하도록 중지합니다.
DebugBreak 현재 프로세스에서 중단점 예외가 발생하도록 합니다. 이렇게 하면 호출 스레드가 디버거가 예외를 처리하도록 신호를 보냅니다.
IsDebuggerPresent 사용자 모드 디버거에서 호출 프로세스를 디버깅하는지 여부를 결정합니다.
OutputDebugStringA 표시할 문자열을 디버거에 보냅니다. (ANSI)
OutputDebugStringW 표시할 문자열을 디버거에 보냅니다. (유니코드)
WaitForDebugEvent

디버깅 중인 프로세스에서 디버깅 이벤트가 발생하기를 기다립니다. (WaitForDebugEvent)
WaitForDebugEventEx 디버깅 중인 프로세스에서 디버깅 이벤트가 발생하기를 기다립니다. (WaitForDebugEventEx)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

deliveryoptimization.h 헤더

아티클2024. 08. 08.

이 헤더는 배달 최적화에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 배달 최적화

deliveryoptimization.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

테이블 확장

IDO다운로드 다운로드를 시작하고 관리하는 데 사용됩니다.
IDODownloadStatusCallback 다운로드에 대한 알림을 받는 데 사용됩니다.
IDOManager 새 다운로드를 만들고 기존 다운로드를 열거하는 데 사용됩니다.

구조체

테이블 확장

DO_DOWNLOAD_ENUM_CATEGORY **IDOManager::EnumDownloads**에서 특정 속성 값으로 다운로드 열거형을 필터링하는 데 사용됩니다.
DO_DOWNLOAD_RANGE 파일에서 다운로드할 단일 바이트 범위를 식별합니다.
DO_DOWNLOAD_RANGES_INFO

파일에서 다운로드할 바이트 범위의 배열을 식별합니다.
DO_DOWNLOAD_STATUS 특정 다운로드의 상태 가져오는 데 사용됩니다.

열거형

테이블 확장

DODownloadCostPolicy **DODownloadProperty_CostPolicy** 속성과 연결된 비용 정책 옵션의 ID를 지정합니다.
DODownloadProperty 배달 최적화 다운로드 작업의 속성 ID를 지정합니다.
DODownloadState **DO_DOWNLOAD_STATUS** 구조의 일부인 현재 다운로드 상태의 ID를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

devfiltertypes.h 헤더

아티클2024. 11. 15.

이 헤더는 디바이스 및 드라이버 설치 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 디바이스 및 드라이버 설치 참조

devfiltertypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조

테이블 확장

DEVPROP_FILTER_EXPRESSION 필터 결과를 쿼리하는 데 사용되는 필터 식을 설명합니다.

열거형

테이블 확장

DEVPROP_OPERATOR DEVPROP_FILTER_EXPRESSION 사용할 수 있는 작업을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

deviceaccess.h 헤더

아티클2023. 08. 23.

이 헤더는 디바이스 액세스 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 디바이스 액세스 API

deviceaccess.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICreateDeviceAccessAsync ICreateDeviceAccessAsync 인터페이스는 CreateDeviceAccessInstance 호출에서 반환됩니다.
IDeviceIoControl 디바이스 드라이버에 제어 코드를 보냅니다. 이 작업을 수행하면 디바이스가 해당 작업을 수행합니다.
IDeviceRequestCompletionCallback DeviceIoControlAsyncmethod에 대한 호출 완료를 처리하는 메서드를 제공합니다.

Functions

CreateDeviceAccessInstance 디바이스에 액세스하는 데 사용되는 개체를 만듭니다. 인스턴스화된 개체는 IDeviceIoControl 및 ICreateDeviceAccessAsync 인터페이스를 구현합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

devicetopology.h 헤더

아티클2023. 08. 23.

이 헤더는 핵심 오디오 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 핵심 오디오 API

devicetopology.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAudioAutoGainControl IAudioAutoGainControl 인터페이스는 하드웨어 AGC(자동 게인 제어)에 대한 액세스를 제공합니다.
IAudioBass IAudioBass 인터페이스는 하드웨어 베이스 수준 컨트롤에 대한 액세스를 제공합니다.
IAudioChannelConfig IAudioChannelConfig 인터페이스는 하드웨어 채널 구성 컨트롤에 대한 액세스를 제공합니다.
IAudioInputSelector IAudioInputSelector 인터페이스는 하드웨어 멀티플렉서 컨트롤(입력 선택기)에 대한 액세스를 제공합니다.
IAudioLoudness IAudioLoudness 인터페이스는 "음음" 보정 컨트롤에 대한 액세스를 제공합니다.
IAudioMidrange IAudioMidrange 인터페이스는 하드웨어 미드레인지 수준 컨트롤에 대한 액세스를 제공합니다.
IAudioMute IAudioMute 인터페이스는 하드웨어 음소거 컨트롤에 대한 액세스를 제공합니다.
IAudioOutputSelector IAudioOutputSelector 인터페이스는 하드웨어 demultiplexer 컨트롤(출력 선택기)에 대한 액세스를 제공합니다.

IAudioPeakMeter IAudioPeakMeter 인터페이스는 하드웨어 피크 미터 컨트롤에 대한 액세스를 제공합니다.
IAudioTreble IAudioTreble 인터페이스는 하드웨어 고음 수준 컨트롤에 대한 액세스를 제공합니다.
IAudioVolumeLevel IAudioVolumeLevel 인터페이스는 하드웨어 볼륨 제어에 대한 액세스를 제공합니다.
IConnector IConnector 인터페이스는 구성 요소 간의 연결 지점을 나타냅니다.
IControlChangeNotify IControlChangeNotify 인터페이스는 파트(커넥터 또는 하위 단위)의 상태 변경될 때 알림을 제공합니다.
IControlInterface IControlInterface 인터페이스는 디바이스 토폴로지의 파트(커넥터 또는 하위 단위)의 제어 인터페이스를 나타냅니다. 클라이언트는 IPart::GetControlInterface 메서드를 호출하여 파트의 IControlInterface 인터페이스에 대한 참조를 가져옵니다.
IDeviceSpecificProperty IDeviceSpecificProperty 인터페이스는 디바이스별 하드웨어 컨트롤의 제어 값에 대한 액세스를 제공합니다.
IDeviceTopology IDeviceTopology 인터페이스는 오디오 디바이스의 토폴로지 액세스 권한을 제공합니다.
IKsFormatSupport IKsFormatSupport 인터페이스는 오디오 어댑터 디바이스와 시스템 메모리 간의 소프트웨어 구성 I/O 연결(일반적으로 DMA 채널)에서 지원하는 오디오 데이터 형식에 대한 정보를 제공합니다.
IKsJackDescription IKsJackDescription 인터페이스는 오디오 어댑터의 디바이스와 외부 또는 내부 엔드포인트 디바이스(예: 마이크 또는 CD 플레이어) 간의 물리적 연결을 제공하는 잭 또는 내부 커넥터에 대한 정보를 제공합니다.
IKsJackDescription2

IKsJackDescription2 인터페이스는 오디오 어댑터의 디바이스와 외부 또는 내부 엔드포인트 디바이스(예: 마이크 또는 CD 플레이어) 간의 물리적 연결을 제공하는 잭 또는 내부 커넥터에 대한 정보를 제공합니다.
IKsJackSinkInformation IKsJackSinkInformation 인터페이스는 잭이 하드웨어에서 지원되는 경우 잭 싱크 정보에 대한 액세스를 제공합니다.
IPart IPart 인터페이스는 디바이스 토폴로지의 파트(커넥터 또는 하위 단위)를 나타냅니다.
IPartsList IPartsList 인터페이스는 각각 커넥터 또는 하위 단위를 나타내는 IPart 인터페이스가 있는 개체인 파트 목록을 나타냅니다.
IPerChannelDbLevel IPerChannelDbLevel 인터페이스는 오디오 스트림 또는 오디오 스트림의 주파수 대역의 볼륨 수준(데시벨)에 대한 채널별 제어를 제공하는 제네릭 하위 단위 컨트롤 인터페이스를 나타냅니다.
ISubunit ISubunit 인터페이스는 클라이언트와 오디오 엔드포인트 디바이스 간의 데이터 경로에 있는 하드웨어 하위 단위(예: 볼륨 컨트롤)를 나타냅니다.

구조체

KSJACK_DESCRIPTION KSJACK_DESCRIPTION 구조는 오디오 잭을 설명합니다.
KSJACK_DESCRIPTION2 KSJACK_DESCRIPTION2 구조는 오디오 잭을 설명합니다. 커넥터의 오디오 잭에 대한 설명을 얻으려면 IKsJackDescription2::GetJackDescription2를 호출합니다.
KSJACK_SINK_INFORMATION KSJACK_SINK_INFORMATION 구조체는 오디오 잭 싱크에 대한 정보를 저장합니다.
Luid

LUID 구조체는 비디오 포트 식별자를 저장합니다. 이 구조체는 KSJACK_SINK_INFORMATION 구조체의 PortId 멤버에 저장됩니다.

열거형

ConnectorType ConnectorType 열거형은 커넥터가 속한 연결 유형을 나타냅니다.
Dataflow DataFlow 열거형은 커넥터를 통해 오디오 스트림의 데이터 흐름 방향을 나타냅니다.
KSJACK_SINK_CONNECTIONTYPE KSJACK_SINK_CONNECTIONTYPE 열거형은 연결 유형을 지정하는 상수를 정의합니다. 이러한 값은 오디오 잭 싱크에 대한 정보를 저장하는 KSJACK_SINK_INFORMATION 구조에 사용됩니다.
PartType PartType 열거형은 디바이스 토폴로지의 부분이 커넥터인지 하위 단위인지를 나타내는 상수를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

devquery.h 헤더

아티클2024. 11. 15.

이 헤더는 디바이스 및 드라이버 설치 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 디바이스 및 드라이버 설치 참조

devquery.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

테이블 확장

DevCloseObjectQuery 디바이스 쿼리 핸들을 닫습니다.
DevCreateObjectQuery 지정된 쿼리 매개 변수를 기반으로 속성을 검색하는 디바이스 쿼리를 만듭니다.
DevCreateObjectQueryEx 지정된 쿼리 매개 변수 및 확장 매개 변수를 기반으로 속성을 검색하는 디바이스 쿼리를 만듭니다.
devCreateObjectQueryFromId 지정된 쿼리 매개 변수 및 개체 ID를 기반으로 속성을 검색하는 디바이스 쿼리를 만듭니다.
DevCreateObjectQueryFromIdEx 지정된 쿼리 매개 변수, 확장 매개 변수 및 개체 ID를 기반으로 속성을 검색하는 디바이스 쿼리를 만듭니다.
devCreateObjectQueryFromIds 지정된 쿼리 매개 변수 및 개체 ID 목록을 기반으로 속성을 검색하는 디바이스 쿼리를 만듭니다.
DevCreateObjectQueryFromIdsEx 지정된 쿼리 매개 변수, 확장 매개 변수 및 개체 ID 목록을 기반으로 속성을 검색하는 디바이스 쿼리를 만듭니다.

devFindProperty DEVPROPERTY 구조체 배열 내의 특정 속성에 해당하는 DEVPROPERTY를 찾습니다.
DevFreeObjectProperties DevGetObjectProperties 또는 DevGetObjectPropertiesEx 호출에 의해 할당된 DEVPROPERTY 구조를 해제합니다.
devFreeObjects DevGetObjects 또는 DevGetObjectsEx를 호출하여 할당된 DEV_OBJECT 구조를 해제합니다.
DevGetObjectProperties 지정된 개체에 대한 속성 집합을 동기적으로 검색합니다.
DevGetObjectPropertiesEx 지정된 확장 매개 변수에 따라 필요에 따라 지정된 개체에 대한 속성 집합을 동기적으로 검색합니다.
devGetObjects 제공된 요청된 속성 및 필터 조건에 따라 DEV_OBJECT 구조 집합을 동기적으로 검색합니다.
DevGetObjectsEx 제공된 요청된 속성, 확장 매개 변수 및 필터 조건에 따라 DEV_OBJECT 구조 집합을 동기적으로 검색합니다.

콜백 함수

테이블 확장

PDEV_QUERY_RESULT_CALLBACK DevQuery 쿼리 결과를 수신하는 콜백 함수에 필요한 함수 프로토타입입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

devquerydef.h 헤더

아티클2024. 11. 15.

이 헤더는 디바이스 및 드라이버 설치 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 디바이스 및 드라이버 설치 참조

devquerydef.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조

테이블 확장

DEV_OBJECT 디바이스 개체를 나타내는 정보를 포함합니다.
DEV_QUERY_RESULT_ACTION_DATA PDEV_QUERY_RESULT_CALLBACK 콜백 함수에 정보를 제공합니다.

열거형

테이블 확장

DEV_OBJECT_TYPE DEV_OBJECT 형식을 지정합니다.
DEV_QUERY_FLAGS 디바이스 쿼리 동작을 변경하는 플래그를 지정합니다.
DEV_QUERY_RESULT_ACTION DEV_QUERY_RESULT_ACTION_DATA 구조체와 연결된 작업의 형식을 지정합니다.
DEV_QUERY_STATE

DEV_QUERY_RESULT_ACTION_DATA 구조와 연결된 쿼리의 상태를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

dhcpcsdk.h 헤더

아티클2023. 08. 23.

이 헤더는 DHCP(동적 호스트 구성 프로토콜)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DHCP(Dynamic Host Configuration Protocol)

dhcpcsdk.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DhcpCApiCleanup DhcpCApiCleanup 함수를 사용하면 DHCP 함수 호출을 통해 할당된 리소스를 올바르게 클린 수 있습니다. DhcpCApiCleanup 함수는 DhcpCApiInitialize에 대한 이전 호출이 성공적으로 실행된 경우에만 호출되어야 합니다.
DhcpCApiInitialize DhcpCApiInitialize 함수는 DHCP 사용자가 수행한 첫 번째 함수 호출이어야 합니다. 다른 모든 DHCP 함수 호출에 대한 시스템을 준비합니다. 다른 DHCP 함수는 DhcpCApiInitialize 함수가 성공적으로 실행되는 경우에만 호출해야 합니다.
DhcpDeRegisterParamChange DhcpDeRegisterParamChange 함수는 이전에 등록된 이벤트 알림과 연결된 리소스를 해제하고 연결된 이벤트 핸들을 닫습니다.
DhcpRegisterParamChange DhcpRegisterParamChange 함수를 사용하면 클라이언트가 DHCP 구성 매개 변수의 변경 내용 알림을 등록할 수 있습니다.
DhcpRemoveDNSRegistrations DhcpRemoveDNSRegistrations 함수는 클라이언트에 대한 모든 DHCP 시작 DNS 등록을 제거합니다.
DhcpRequestParams DhcpRequestParams 함수를 사용하면 호출자가 DHCP 서버에서 DHCP 데이터를 동기적으로 또는 동기적으로 지속적으로 가져올 수 있습니다.

DhcpUndoRequestParams DhcpUndoRequestParams 함수는 이전에 DhcpRequestParams 함수 호출을 사용하여 수행한 영구 요청을 제거합니다.

구조체

DHCPAPI_PARAMS DHCPAPI_PARAMS 구조체는 DHCP 매개 변수를 요청하는 데 사용됩니다.
DHCPCAPI_CLASSID DHCPCAPI_CLASSID 구조체는 클라이언트 클래스 ID를 정의합니다.
DHCPCAPI_PARAMS_ARRAY DHCPCAPI_PARAMS_ARRAY 구조체는 DHCP 매개 변수를 쿼리하는 데 사용되는 DHCPAPI_PARAMS 구조의 배열을 저장합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dhcpsapi.h 헤더

아티클2023. 08. 23.

이 헤더는 DHCP(동적 호스트 구성 프로토콜)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DHCP(Dynamic Host Configuration Protocol)

dhcpsapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DhcpAddFilterV4 허용/거부 목록에 링크 계층 주소 또는 주소 패턴을 추가합니다.
DhcpAddServer DhcpAddServer 함수는 도메인 디렉터리 서비스에서 유지 관리되는 기존 DHCP 서버 목록에 새 서버를 추가하려고 시도합니다. 지정된 DHCP 서버가 디렉터리 서비스에 이미 있는 경우 오류가 반환됩니다.
DhcpAddSubnetElement 서브넷의 기능 또는 측면을 설명하는 요소를 DHCP 데이터베이스의 서브넷 항목에 추가합니다.
DhcpAddSubnetElementV4 서브넷의 기능 또는 측면을 설명하는 요소를 DHCP 데이터베이스의 서브넷 항목에 추가합니다. 이 함수는 클라이언트 형식을 고려하는 서브넷 요소를 통합하여 DhcpAddSubnetElement를 확장합니다.
DhcpAddSubnetElementV5 DhcpAddSubnetElementV5 함수는 서브넷의 기능 또는 측면을 설명하는 요소를 DHCP 데이터베이스의 서브넷 항목에 추가합니다. Windows 2000 이하: 이 함수를 사용할 수 없습니다.
DhcpAddSubnetElementV6 DhcpAddSubnetElementV6 함수는 서브넷의 기능 또는 측면을 설명하는 요소를 DHCP 데이터베이스의 서브넷 항목에 추가합니다.
DhcpAuditLogGetParams DHCP 서버에서 감사 로그 구성 설정을 반환합니다.

DhcpAuditLogSetParams DHCP 서버에서 감사 로그 생성에 대한 매개 변수를 설정합니다.
DhcpCreateClass 사용자 지정 옵션 클래스를 만듭니다.
DhcpCreateClassV6 사용자 지정 DHCPv6 옵션 클래스를 만듭니다.
DhcpCreateClientInfo DhcpCreateClientInfo 함수는 DHCP 서버에 클라이언트 정보 레코드를 만듭니다.
DhcpCreateClientInfoV4 DHCP 서버에 클라이언트 정보 레코드를 만들어 레코드에 클라이언트 유형(DHCP 또는 BOOTP)을 포함하여 DhcpCreateClientInfo의 기능을 확장합니다.
DhcpCreateClientInfoVQ 제공된 DHCP 클라이언트 임대 레코드를 DHCP 서버 데이터베이스에 만듭니다.
DhcpCreateOption 기본 옵션 수준에서 기본 사용자 및 공급업체 클래스에 대한 옵션 정의를 만듭니다.
DhcpCreateOptionV5 DHCP 옵션을 만듭니다.
DhcpCreateOptionV6 DhcpCreateOptionV6 함수는 DHCP 옵션을 만듭니다.
DhcpCreateSubnet DhcpCreateSubnet 함수는 DHCP 서버에 새 서브넷을 만듭니다.
DhcpCreateSubnetV6 DhcpCreateSubnetV6 함수는 DHCP 서버에 새 서브넷을 만듭니다.
DhcpCreateSubnetVQ DhcpCreateSubnetVQ 함수는 DHCP 서버에 새 IPv4 서브넷 및 관련 NAP 상태 정보를 만듭니다.

DhcpDeleteClass DHCP 서버에서 DHCP 클래스를 삭제합니다.
DhcpDeleteClassV6 DHCPv6 서버에서 DHCP 클래스를 삭제합니다.
DhcpDeleteClientInfo DhcpDeleteClientInfo 함수는 DHCP 서버에서 클라이언트 정보 레코드를 삭제합니다.
DhcpDeleteClientInfoV6 DHCPv6 서버 데이터베이스에서 지정된 DHCPv6 클라이언트 주소 릴리스 레코드를 삭제합니다.
DhcpDeleteFilterV4 DHCP 서버의 허용/거부 목록에서 링크 계층 주소 또는 주소 패턴을 삭제합니다.
DhcpDeleteServer DhcpDeleteServer 함수는 디렉터리 서비스에서 DHCP 서버 및 관련 개체(예: 서브넷 정보 및 IP 예약)를 삭제하려고 시도합니다.
DhcpDeleteSubnet DhcpDeleteSubnet 함수는 DHCP 서버에서 서브넷을 삭제합니다.
DhcpDeleteSubnetV6 DhcpDeleteSubnetV6 함수는 DHCP 서버에서 서브넷을 삭제합니다.
DhcpDeleteSuperScopeV4 DHCP 서버에서 초경을 삭제합니다.
DhcpDsCleanup DhcpDsCleanup 함수는 DhcpDsInit에서 DHCP 서비스에 할당된 디렉터리 서비스 리소스를 해제합니다. 이 함수는 해당하는 각 DHCP 서비스 프로세스에 대해 정확히 한 번 호출되어야 하며 프로세스가 종료된 경우에만 호출해야 합니다.
DhcpDsInit DhcpDsInit 함수는 새 DHCP 서버 프로세스에 대한 디렉터리 서비스 내의 메모리를 초기화합니다.
DhcpEnumClasses

DHCP 서버에 대해 구성된 사용자 또는 공급업체 클래스를 열거합니다.
DhcpEnumClassesV6 DHCPv6 서버에 대해 구성된 사용자 또는 공급업체 클래스를 열거합니다.
DhcpEnumFilterV4 DHCP 서버의 허용 또는 거부 목록에서 모든 필터 레코드를 열거합니다.
DhcpEnumOptions DHCPv4 서버에 저장된 열거형 옵션 집합을 반환합니다.
DhcpEnumOptionsV5 DhcpEnumOptionsV5 함수는 지정된 사용자 또는 공급업체 클래스에 대한 DHCP 옵션의 열거된 목록을 반환합니다.
DhcpEnumOptionsV6 DhcpEnumOptionsV6 함수는 지정된 클래스 및/또는 공급업체에 대한 DHCP 옵션의 열거된 목록을 반환합니다.
DhcpEnumOptionValues 지정된 scope 대해 열거된 옵션 값 목록(옵션 데이터 및 연결된 ID 번호만)을 반환합니다.
DhcpEnumOptionValuesV5 DhcpEnumOptionValuesV5 함수는 지정된 사용자 또는 공급업체 클래스 내의 특정 scope 대해 열거된 옵션 값 목록(옵션 데이터 및 연결된 ID 번호만)을 반환합니다.
DhcpEnumOptionValuesV6 DhcpEnumOptionValuesV6 함수는 지정된 클래스 내의 특정 scope 대해 열거된 옵션 값 목록(옵션 데이터 및 연결된 ID 번호)을 반환합니다.
DhcpEnumServers DhcpEnumServers 함수는 디렉터리 서비스에 있는 DHCP 서버의 열거된 목록을 반환합니다.
DhcpEnumSubnetClients DhcpEnumSubnetClients 함수는 지정된 서브넷에 제공된 IP 주소가 있는 열거된 클라이언트 목록을 반환합니다.
DhcpEnumSubnetClientsFilterStatusInfo

지정된 서브넷에서 서비스되는 모든 DHCP 클라이언트를 열거하고 각 서브넷에 대한 링크 계층 필터 상태 포함합니다.
DhcpEnumSubnetClientsV4 지정된 서브넷에 제공된 IP 주소가 있는 클라이언트 임대 레코드의 열거된 목록을 반환합니다.
DhcpEnumSubnetClientsV5 지정된 서브넷에 제공된 IP 주소가 있는 열거형 클라이언트 목록을 반환합니다.
DhcpEnumSubnetClientsV6 DhcpEnumSubnetClientsV6 함수는 지정된 서브넷에 제공된 IP 주소가 있는 열거형 클라이언트 목록을 반환합니다.
DhcpEnumSubnetClientsVQ 지정된 IPv4 서브넷에서 서비스되는 모든 DHCP 클라이언트를 검색합니다.
DhcpEnumSubnetElements 특정 DHCP 서브넷에 대해 열거된 요소 목록을 반환합니다.
DhcpEnumSubnetElementsV4 특정 DHCP 서브넷에 대해 열거된 요소 목록을 반환합니다. 이 함수는 클라이언트 유형에 따라 IP 예약을 포함할 수 있는 DHCP_SUBNET_ELEMENT_DATA_V4 구조 목록을 반환하여 DhcpEnumSubnetElements를 확장합니다.
DhcpEnumSubnetElementsV5 DhcpEnumSubnetElementsV5 함수는 특정 DHCP 서브넷에 대해 열거된 요소 목록을 반환합니다.
DhcpEnumSubnetElementsV6 DhcpEnumSubnetElementsV6 함수는 특정 DHCP 서브넷에 대해 열거된 요소 목록을 반환합니다.
DhcpEnumSubnets DhcpEnumSubnets 함수는 DHCP 서버에 정의된 서브넷의 열거된 목록을 반환합니다.
DhcpEnumSubnetsV6 DhcpEnumSubnetsV6 함수는 DHCP 서버에 정의된 서브넷의 열거된 목록을 반환합니다.
DhcpGetAllOptions DHCP 서버에 정의된 모든 옵션을 포함하는 배열을 반환합니다.

DhcpGetAllOptionsV6 DhcpGetAllOptionsV6 함수는 DHCP 서버에 정의된 모든 옵션을 포함하는 배열을 반환합니다.
DhcpGetAllOptionValues DHCP 서버의 특정 scope 대해 정의된 모든 옵션 값이 들어 있는 배열을 반환합니다.
DhcpGetAllOptionValuesV6 DhcpGetAllOptionValuesV6 함수는 DHCP 서버의 특정 scope 대해 정의된 모든 옵션 값이 포함된 배열을 반환합니다.
DhcpGetClassInfo DhcpGetClassInfo 함수는 특정 DHCP 서버에 구성된 사용자 또는 공급업체 클래스 정보를 반환합니다.
DhcpGetClientInfo DhcpGetClientInfo 함수는 특정 DHCP 클라이언트에 대한 정보를 반환합니다.
DhcpGetClientInfoV4 특정 DHCP 클라이언트에 대한 정보를 반환합니다. 이 함수는 클라이언트 형식 정보를 포함하는 DHCP_CLIENT_INFO_V4 구조를 반환하여 DhcpGetClientInfo를 확장합니다.
DhcpGetClientInfoV6 DHCPv6 서버에서 특정 IPv6 클라이언트 예약에 대한 IPv6 주소 임대 정보를 검색합니다.
DhcpGetClientInfoVQ DHCP 서버 데이터베이스에서 DHCP 클라이언트 임대 레코드 정보를 검색합니다. (DhcpGetClientInfoVQ)
DhcpGetClientOptions 사용되거나 지원되지 않으므로 ERROR_NOT_IMPLEMENTED만 반환합니다.
DhcpGetFilterV4 DHCPv4 서버의 허용/거부 목록에 대한 사용/사용 안 함 설정을 검색합니다.
DhcpGetMibInfoV5 지정된 DHCP 서버에 대한 현재 통계를 포함하는 MIB 데이터 구조를 가져옵니다.

DhcpGetMibInfoV6 DHCP 서버의 IPv6 카운터 값을 검색합니다.
DhcpGetOptionInfo 기본 사용자 및 공급업체 클래스에 대한 특정 DHCP 옵션에 대한 정보를 반환합니다.
DhcpGetOptionInfoV5 DhcpGetOptionInfoV5 함수는 특정 DHCP 옵션에 대한 정보를 반환합니다.
DhcpGetOptionInfoV6 DhcpGetOptionInfoV6 함수는 특정 DHCP 옵션에 대한 정보를 반환합니다.
DhcpGetOptionValue DhcpGetOptionValue 함수는 특정 scope 대한 DHCP 옵션 값(옵션 코드 및 관련 데이터)을 검색합니다.
DhcpGetOptionValueV5 DhcpGetOptionValueV5 함수는 특정 scope 대한 DHCP 옵션 값(옵션 코드 및 관련 데이터)을 검색합니다.
DhcpGetOptionValueV6 특정 사용자 또는 공급업체 클래스에 대해 DHCPv6 서버에 정의된 특정 옵션의 옵션 값을 검색합니다.
DhcpGetServerBindingInfo DhcpGetServerBindingInfo 함수는 DHCP 서버에 설정된 엔드포인트 바인딩을 반환합니다.
DhcpGetServerBindingInfoV6 DHCPv6 서버와 관련된 IPv6 인터페이스 바인딩 정보의 배열을 검색합니다.
DhcpGetServerSpecificStrings 기본 공급업체 클래스 및 사용자 클래스의 이름을 검색합니다.
DhcpGetSubnetDelayOffer DISCOVER 메시지를 받은 후 DHCP OFFER 메시지의 지연 기간을 가져옵니다.
DhcpGetSubnetInfo

DhcpGetSubnetInfo 함수는 특정 서브넷에 대한 정보를 반환합니다.
DhcpGetSubnetInfoV6 DhcpGetSubnetInfoV6 함수는 특정 서브넷에 대한 정보를 반환합니다.
DhcpGetSubnetInfoVQ DHCP 서버에 정의된 특정 IPv4 서브넷에 대한 정보를 검색합니다.
DhcpGetSuperScopeInfoV4 DHCP 서버의 대범위 정보를 반환합니다.
DhcpGetThreadOptions DhcpGetThreadOptions 함수는 DhcpSetThreadOptions에서 설정한 현재 스레드 옵션을 검색합니다.
DhcpGetVersion DhcpGetVersion 함수는 DHCP 서버의 주 버전 및 부 버전 번호를 반환합니다.
DhcpHlprAddV4PolicyCondition DHCP 서버 정책 조건을 할당, 초기화 및 DHCP 서버 정책에 추가합니다.
DhcpHlprAddV4PolicyExpr DHCP 서버 정책 식을 할당, 초기화 및 DHCP 서버 정책에 추가합니다.
DhcpHlprAddV4PolicyRange DHCP 서버 정책에 DHCP IPv4 범위를 추가합니다.
DhcpHlprCreateV4Policy DHCP 서버 정책 구조를 할당하고 초기화합니다.
DhcpHlprFreeV4Policy DHCP 서버 정책 구조 내의 모든 데이터 구조의 메모리를 해제합니다.
DhcpHlprIsV4PolicySingleUC DHCP 서버 정책이 단일 사용자 클래스를 기반으로 하는지 확인합니다.
DhcpHlprIsV4PolicyValid DHCP 서버 정책을 확인합니다.

DhcpHlprIsV4PolicyWellFormed DHCP 서버 정책 구조가 올바른 형식인지 확인합니다.
DhcpHlprModifyV4PolicyExpr DHCP 서버 정책 구조에서 DHCP 서버 정책 식을 수정합니다.
DhcpHlprResetV4PolicyExpr DHCP 서버 정책 구조에서 DHCP 서버 정책 식을 다시 설정합니다.
DhcpModifyClass 서버에 정의된 DHCP 클래스를 수정합니다.
DhcpModifyClassV6 서버에 정의된 DHCPv6 사용자 또는 공급업체 클래스를 수정합니다.
DhcpRemoveOption DHCP 서버의 기본 옵션 수준에서 기본 사용자 클래스 및 공급업체 클래스에 대한 특정 옵션의 정의를 제거합니다.
DhcpRemoveOptionV5 DHCP 서버의 기본 옵션 수준에서 특정 사용자 클래스 및 공급업체 클래스에 대한 특정 옵션의 정의를 제거합니다. 이렇게 하면 특정 클래스 및 공급업체 이름을 지원하는 DhcpRemoveOption의 기능이 확장됩니다.
DhcpRemoveOptionV6 DhcpRemoveOptionV6 함수는 DHCP 서버에 정의된 옵션을 제거합니다.
DhcpRemoveOptionValue 지정된 scope 대한 기본 사용자 클래스 및 공급업체 클래스에 대한 DHCP4 서버의 특정 옵션에 대한 옵션 값을 제거합니다.
DhcpRemoveOptionValueV5 DhcpRemoveOptionValueV5 함수는 DHCP 서버에 정의된 scope 옵션 값을 제거합니다.
DhcpRemoveOptionValueV6 DhcpRemoveOptionValueV6 함수는 DHCP 서버에 정의된 scope 옵션 값을 제거합니다.

DhcpRemoveSubnetElement DHCPv4 서버에 정의된 IPv4 서브넷에서 IPv4 서브넷 요소를 제거합니다.
DhcpRemoveSubnetElementV4 DHCPv4 서버에 정의된 IPv4 서브넷에서 IPv4 서브넷 요소를 제거합니다. 함수는 클라이언트 유형(DHCP 또는 BOOTP) 정보를 포함하는 서브넷의 사양을 허용하여 DhcpRemoveSubnetElement에서 제공하는 기능을 확장합니다.
DhcpRemoveSubnetElementV5 DhcpRemoveSubnetElementV5 함수는 DHCP 서버에 정의된 서브넷에서 요소를 제거합니다.
DhcpRemoveSubnetElementV6 DhcpRemoveSubnetElementV6 함수는 DHCP 서버에 정의된 서브넷에서 요소를 제거합니다.
DhcpRpcFreeMemory DhcpRpcFreeMemory 함수는 매개 변수로 반환된 버퍼 공간 블록을 해제합니다.
DhcpScanDatabase 메모리 내 캐시와 서버 데이터베이스 간에 동기화되지 않은 임대된 DHCPv4 클라이언트 IPv4 주소를 열거합니다.
DhcpServerBackupDatabase DHCP 서버 데이터베이스 구성, 설정 및 DHCP 클라이언트 임대 레코드를 지정된 파일 위치에 백업합니다.
DhcpServerGetConfig DHCP 서버의 특정 구성 설정을 반환합니다. (DhcpServerGetConfig)
DhcpServerGetConfigV4 DHCP 서버의 특정 구성 설정을 반환합니다. (DhcpServerGetConfigV4)
DhcpServerGetConfigV6 DHCPv6 서버에 대한 구성 정보를 검색합니다.
DhcpServerGetConfigVQ 현재 DHCP 서버 구성 설정을 검색합니다.

DhcpServerQueryAttribute DHCP 서버에서 특정 특성 정보를 반환합니다.
DhcpServerQueryAttributes DHCP 서버에 설정된 특성 배열을 반환합니다.
DhcpServerQueryDnsRegCredentials 클라이언트 동적 DNS 등록을 위해 DHCP 서버에서 사용하는 현재 DNS(도메인 이름 시스템) 자격 증명을 검색합니다.
DhcpServerRedoAuthorization DhcpServerRedoAuthorization 함수는 DHCP 서버의 권한이 부여되었는지 여부를 확인하고 그렇지 않은 경우 임대 작업을 복원합니다.
DhcpServerRestoreDatabase 특정 백업 위치(경로)에서 클라이언트 임대 데이터베이스에 대한 설정, 구성 및 레코드를 복원합니다.
DhcpServerSetConfig 서브넷 및 클라이언트 임대 정보를 저장하는 데 사용되는 JET 데이터베이스에 대한 정보 및 지원되는 프로토콜을 포함하여 특정 설정을 사용하여 DHCPv4 서버를 구성합니다.
DhcpServerSetConfigV4 DhcpServerSetConfigV4 함수는 서브넷 및 클라이언트 임대 정보를 저장하는 데 사용되는 JET 데이터베이스에 대한 정보 및 지원되는 프로토콜을 포함하여 특정 설정으로 DHCP 서버를 구성합니다.
DhcpServerSetConfigV6 scope 또는 서버 수준에서 DHCPv6 서버 구성 데이터를 설정합니다.
DhcpServerSetConfigVQ DHCP 서버 설정을 설정하거나 업데이트합니다.
DhcpServerSetDnsRegCredentialsV5 DHCP 서버에서 DHCP 클라이언트 임대 레코드에 대한 DNS(도메인 이름 시스템) 등록을 만드는 데 사용하는 자격 증명을 설정합니다.
DhcpSetClientInfo

DhcpSetClientInfo 함수는 IP 주소 임대가 DHCP 서버에서 관리되는 클라이언트에 대한 정보를 설정합니다.
DhcpSetClientInfoV4 IP 주소 임대가 DHCP 서버에서 관리되는 클라이언트에 대한 정보를 설정합니다. 이 함수는 호출자가 클라이언트 유형(DHCP 또는 BOOTP)을 지정할 수 있도록 하여 DhcpSetClientInfo에서 제공하는 기능을 확장합니다.
DhcpSetClientInfoV6 DHCPv6 서버 데이터베이스에서 예약된 DHCPv6 클라이언트 임대 레코드를 설정하거나 수정합니다.
DhcpSetClientInfoVQ DHCP 서버 레코드 데이터베이스에서 기존 DHCP 클라이언트 임대 레코드를 설정하거나 수정합니다.
DhcpSetFilterV4 DHCP 서버에서 허용 및 거부 목록을 사용하거나 사용하지 않도록 설정합니다.
DhcpSetOptionInfo 기본 옵션 수준에서 기본 사용자 클래스 및 공급업체 클래스에 대해 지정된 옵션의 옵션 정의를 수정합니다.
DhcpSetOptionInfoV5 특정 DHCP 옵션에 대한 정보를 설정합니다.
DhcpSetOptionInfoV6 DhcpSetOptionInfoV6 함수는 특정 DHCP 옵션에 대한 정보를 설정합니다.
DhcpSetOptionValue DHCP 서버의 특정 옵션 값에 대한 정보를 설정합니다.
DhcpSetOptionValues DHCP 서버에 정의된 특정 scope 대한 옵션 코드 및 관련 데이터 값을 설정합니다.
DhcpSetOptionValuesV5 DHCP 서버에 정의된 특정 scope 대한 옵션 코드 및 관련 데이터 값을 설정합니다. 이 함수는 호출자가 옵션에 대한 클래스 및/또는 공급업체를 지정할 수 있도록 하여 DhcpSetOptionValues에서 제공하는 기능을 확장합니다.

DhcpSetOptionValueV5 DhcpSetOptionValueV5 함수는 DHCP 서버의 특정 옵션 값에 대한 정보를 설정합니다. 이 함수는 호출자가 옵션에 대한 클래스 및/또는 공급업체를 지정할 수 있도록 하여 DhcpSetOptionValue에서 제공하는 기능을 확장합니다.
DhcpSetOptionValueV6 DhcpSetOptionValueV6 함수는 DHCP 서버의 특정 옵션 값에 대한 정보를 설정합니다.
DhcpSetServerBindingInfo DhcpSetServerBindingInfo 함수는 DHCP 서버에 대한 엔드포인트 바인딩을 설정합니다.
DhcpSetServerBindingInfoV6 DHCPv6 서버에 대한 IPv6 인터페이스 바인딩을 설정하거나 수정합니다.
DhcpSetSubnetDelayOffer 특정 DHCP scope 대해 DISCOVER 메시지를 받은 후 DHCP OFFER 메시지의 지연 기간을 설정합니다.
DhcpSetSubnetInfo DhcpSetSubnetInfo 함수는 DHCP 서버에 정의된 서브넷에 대한 정보를 설정합니다.
DhcpSetSubnetInfoV6 DHCPv6 서버에 정의된 IPv6 서브넷에 대한 정보를 설정하거나 업데이트합니다.
DhcpSetSubnetInfoVQ DHCP 서버에 정의된 서브넷에 대한 정보를 설정합니다.
DhcpSetSuperScopeV4 서브넷을 DHCP 서버의 대범위로 설정합니다.
DhcpSetThreadOptions DhcpSetThreadOptions 함수는 현재 실행 중인 DHCP 스레드에 대한 옵션을 설정합니다.
DhcpV4AddPolicyRange 정책에 IP 주소 범위를 추가합니다.
DhcpV4CreateClientInfo

DHCP 서버 데이터베이스에 DHCPv4 클라이언트 임대 레코드를 만듭니다.
DhcpV4CreatePolicy DHCP 서버에 새 정책을 만듭니다.
DhcpV4DeletePolicy DHCP 서버에서 기존 정책을 삭제합니다.
DhcpV4EnumPolicies DHCP 서버에 구성된 정책을 열거합니다.
DhcpV4EnumSubnetClients 지정된 IPv4 서브넷에서 서비스된 모든 DHCP 클라이언트 레코드를 열거합니다.
DhcpV4EnumSubnetReservations 특정 DHCP IPv4 서브넷에 대한 예약을 열거합니다.
DhcpV4FailoverAddScopeToRelationship 지정된 장애 조치(failover) 관계에 DHCPv4 scope 추가합니다.
DhcpV4FailoverCreateRelationship 두 서버 간에 새 DHCPv4 장애 조치(failover) 관계를 만듭니다.
DhcpV4FailoverDeleteRelationship 두 서버 간의 DHCPv4 장애 조치(failover) 관계를 삭제합니다.
DhcpV4FailoverDeleteScopeFromRelationship 지정된 장애 조치(failover) 관계에서 DHCPv4 scope 삭제합니다.
DhcpV4FailoverEnumRelationship 서버에 있는 모든 장애 조치(failover) 관계를 열거합니다.
DhcpV4FailoverGetAddressStatus IPv4 주소의 상태 반환합니다.
DhcpV4FailoverGetClientInfo DHCPv4 클라이언트 임대 정보를 검색합니다.

DhcpV4FailoverGetRelationship 특정 관계 이름에 대한 관계 세부 정보를 검색합니다.
DhcpV4FailoverGetScopeRelationship 지정된 DHCPv4 scope 구성된 장애 조치(failover) 관계를 검색합니다.
DhcpV4FailoverGetScopeStatistics 장애 조치(failover) 관계의 일부인 특정 scope 주소 사용 통계를 검색합니다.
DhcpV4FailoverGetSystemTime DHCP 서버의 현재 시간을 반환합니다.
DhcpV4FailoverSetRelationship DHCPv4 서버 장애 조치(failover) 관계의 매개 변수를 설정하거나 수정합니다.
DhcpV4FailoverTriggerAddrAllocation 장애 조치(failover) 관계의 일부인 주 서버와 보조 서버 간에 무료 주소를 재배포합니다.
DhcpV4GetAllOptionValues 특정 scope 대한 DHCP 옵션 값(옵션 코드 및 관련 데이터) 배열을 검색합니다.
DhcpV4GetClientInfo DHCP 서버 데이터베이스에서 DHCP 클라이언트 임대 레코드 정보를 검색합니다. (DhcpV4GetClientInfo)
DhcpV4GetFreeIPAddress 클라이언트에 임대할 수 있는 사용 가능한 IPv4 주소 목록을 검색합니다.
DhcpV4GetOptionValue 특정 scope 대한 DHCP 옵션 값(옵션 코드 및 관련 데이터)을 검색합니다. 이 함수는 호출자가 옵션에 대한 정책을 지정할 수 있도록 하여 DhcpGetOptionValueV5에서 제공하는 기능을 확장합니다.
DhcpV4GetPolicy DHCP 서버에서 정책을 검색합니다.
DhcpV4QueryPolicyEnforcement DHCP 서버에서 서버 또는 지정된 IPv4 서브넷의 정책 적용 상태를 검색합니다.

DhcpV4RemoveOptionValue DHCP 서버에 정의된 scope 옵션 값을 제거합니다. 이 함수는 호출자가 옵션에 대한 정책을 지정할 수 있도록 하여 DhcpRemoveOptionValueV5에서 제공하는 기능을 확장합니다.
DhcpV4RemovePolicyRange 정책의 IP 주소 범위 목록에서 지정된 IP 주소 범위를 제거합니다.
DhcpV4SetOptionValue DHCP 서버의 특정 옵션 값에 대한 정보를 설정합니다. 이 함수는 호출자가 옵션에 대한 정책을 지정할 수 있도록 하여 DhcpSetOptionValueV5에서 제공하는 기능을 확장합니다.
DhcpV4SetOptionValues DHCP 서버에 정의된 특정 scope 대한 옵션 코드 및 관련 데이터 값을 설정합니다. 이 함수는 호출자가 옵션에 대한 정책을 지정할 수 있도록 하여 DhcpSetOptionValuesV5에서 제공하는 기능을 확장합니다.
DhcpV4SetPolicy 기존 정책의 매개 변수를 하나 이상 업데이트.
DhcpV4SetPolicyEnforcement DHCP 서버에서 서버 또는 지정된 IPv4 서브넷의 정책 적용 상태를 설정합니다.
DhcpV6CreateClientInfo DHCP 서버 데이터베이스에 DHCPv6 클라이언트 임대 레코드를 만듭니다.
DhcpV6GetFreeIPAddress 클라이언트에 임대할 수 있는 사용 가능한 IPv6 주소 목록을 검색합니다.
DhcpV6GetStatelessStatistics 상태 비저장 서버 IPv6 서브넷 통계를 검색합니다.
DhcpV6GetStatelessStoreParams 서버 또는 scope 수준에서 현재 DHCPv6 상태 비정상 클라이언트 인벤토리 구성 설정을 검색합니다.
DhcpV6SetStatelessStoreParams 서버 또는 scope 수준에서 DHCPv6 상태 비정상 클라이언트 인벤토리 구성 설정을 설정합니다.

구조체

DATE_TIME DATE_TIME 구조체는 12:00 자정, 1월 1일, 1C.E. 이후 틱 수(100나노초 증분)로 표현되는 날짜/시간을 포함하는 64비트 정수 값을 정의합니다.
DHCP_ADDR_PATTERN 링크 계층 주소/패턴에 대한 정보를 포함합니다.
DHCP_ALL_OPTION_VALUES 클래스/공급업체 페어링에 따라 구성된 DHCP 서버에 정의된 모든 옵션 값의 집합을 정의합니다.
DHCP_ALL_OPTION_VALUES_PB DHCP_ALL_OPTION_VALUES_PB 구조체는 scope 내의 DHCP 서버에 대한 모든 옵션 값 집합을 정의합니다.
DHCP_ALL_OPTIONS DHCP 서버에서 사용할 수 있는 모든 옵션 집합을 정의합니다.
DHCP_ATTRIB DHCP 서버에서 설정된 특성을 정의합니다.
DHCP_ATTRIB_ARRAY DHCP 서버 특성 집합을 정의합니다.
DHCP_BINARY_DATA DHCP_BINARY_DATA 구조체는 이진 데이터의 불투명 Blob을 정의합니다.
DHCP_BIND_ELEMENT DHCP 서버에 대한 개별 네트워크 바인딩을 정의합니다. 단일 DHCP 서버는 여러 바인딩을 포함하고 여러 네트워크를 제공할 수 있습니다.
DHCP_BIND_ELEMENT_ARRAY DHCP_BIND_ELEMENT_ARRAY 구조체는 DHCP 서버에서 사용하는 네트워크 바인딩 요소의 배열을 정의합니다.
DHCP_BOOTP_IP_RANGE

DHCP_BOOTP_IP_RANGE 구조는 BOOTP 관련 클라이언트에 임대하기 위한 IP 제품군을 정의합니다.
DHCP_CLASS_INFO DHCP 옵션 클래스를 정의합니다.
DHCP_CLASS_INFO_ARRAY DHCP 클래스 정보를 포함하는 요소의 배열을 정의합니다.
DHCP_CLASS_INFO_ARRAY_V6 DHCP_CLASS_INFO_ARRAY_V6 구조에는 사용자 클래스 또는 공급업체 클래스에 대한 정보 목록이 포함되어 있습니다.
DHCP_CLASS_INFO_V6 특정 DHCPv6 사용자 클래스 또는 공급업체 클래스에 대한 정보를 포함합니다.
DHCP_CLIENT_FILTER_STATUS_INFO 필터 상태 정보를 포함하여 DHCPv4 클라이언트에 대한 정보를 정의합니다.
DHCP_CLIENT_FILTER_STATUS_INFO_ARRAY DHCPv4 클라이언트에 대한 정보 요소의 배열을 포함합니다.
DHCP_CLIENT_INFO DHCP_CLIENT_INFO 구조는 DHCP 서버에서 사용하는 클라이언트 정보 레코드를 정의합니다.
DHCP_CLIENT_INFO_ARRAY DHCP_CLIENT_INFO_ARRAY 구조체는 열거형 함수에 사용할 DHCP_CLIENT_INFO 구조체의 배열을 정의합니다.
DHCP_CLIENT_INFO_ARRAY_V4 열거형 함수에 사용할 DHCP_CLIENT_INFO_V4 구조체의 배열을 정의합니다.
DHCP_CLIENT_INFO_ARRAY_V5 열거형 함수에 사용할 DHCP_CLIENT_INFO_V5 구조체의 배열을 정의합니다.
DHCP_CLIENT_INFO_ARRAY_V6 DHCPv6 클라이언트 열거형 함수에 사용할 DHCP_CLIENT_INFO_V6 구조의 배열을 정의합니다.

DHCP_CLIENT_INFO_ARRAY_VQ DHCP_CLIENT_INFO_VQ 구조체의 배열을 지정합니다.
DHCP_CLIENT_INFO_PB DHCP_CLIENT_INFO_PB 구조는 필터 상태 정보 및 IPv4 주소 할당을 초래한 모든 정책을 포함하여 DHCPv4 클라이언트에 대한 정보를 정의합니다.
DHCP_CLIENT_INFO_PB_ARRAY DHCP_CLIENT_INFO_PB_ARRAY 구조는 DHCPv4 클라이언트 정보 요소의 배열을 정의합니다.
DHCP_CLIENT_INFO_V4 DHCP 서버에서 사용하는 클라이언트 정보 레코드를 정의하여 클라이언트 형식 정보를 포함하여 DHCP_CLIENT_INFO 제공된 정의를 확장합니다.
DHCP_CLIENT_INFO_V5 DHCP 서버에서 사용하는 클라이언트 정보 레코드를 정의하여 클라이언트 유형 및 주소 상태 정보를 포함하여 DHCP_CLIENT_INFO 제공된 정의를 확장합니다.
DHCP_CLIENT_INFO_V6 DHCP_CLIENT_INFO_V6 구조에는 DHCPv6 클라이언트에 대한 정보가 포함되어 있습니다.
DHCP_CLIENT_INFO_VQ DHCPv4 클라이언트에 대한 정보를 정의합니다.
DHCP_FAILOVER_RELATIONSHIP DHCP_FAILOVER_RELATIONSHIP 구조는 DHCPv4 서버 장애 조치(failover) 관계에 대한 정보를 정의합니다.
DHCP_FAILOVER_RELATIONSHIP_ARRAY DHCP_FAILOVER_RELATIONSHIP_ARRAY 구조는 파트너 서버 간의 DHCPv4 장애 조치(failover) 관계 배열을 정의합니다.
DHCP_FAILOVER_STATISTICS DHCP_FAILOVER_STATISTICS 구조는 장애 조치(failover) 관계의 일부인 DHCP 서버 scope 통계를 정의합니다.
DHCP_FILTER_ADD_INFO 허용 및 거부 필터 목록에 추가할 링크 계층 필터에 대한 정보를 포함합니다.

DHCP_FILTER_ENUM_INFO 링크 계층 필터 레코드 수에 대한 정보를 포함합니다.
DHCP_FILTER_GLOBAL_INFO 허용 및 거부 필터 목록을 사용하도록 설정하고 사용하지 않도록 설정하는 방법에 대한 정보를 포함합니다.
DHCP_FILTER_RECORD 특정 링크 계층 필터에 대한 정보를 포함합니다.
DHCP_HOST_INFO DHCP_HOST_INFO 구조체는 DHCP 서버(호스트)에 대한 정보를 정의합니다.
DHCP_HOST_INFO_V6 해당 IPv6 주소 및 이름과 같은 DHCPv6 서버(호스트)에 대한 네트워크 정보를 포함합니다.
DHCP_IP_ARRAY DHCP_IP_ARRAY 구조체는 IP 주소 배열을 정의합니다.
DHCP_IP_CLUSTER DHCP_IP_CLUSTER 구조는 네트워크 클러스터의 주소와 마스트를 정의합니다.
DHCP_IP_RANGE DHCP_IP_RANGE 구조체는 IP 주소 범위를 정의합니다.
DHCP_IP_RANGE_ARRAY DHCP_IP_RANGE_ARRAY 구조체는 DHCP IPv4 범위의 배열을 정의합니다.
DHCP_IP_RANGE_V6 DHCPv6 서버에서 사용할 IPv6 주소 범위를 지정합니다.
DHCP_IP_RESERVATION DHCP_IP_RESERVATION 구조는 클라이언트 IP 예약을 정의합니다.
DHCP_IP_RESERVATION_INFO DHCP_IP_RESERVATION_INFO 구조는 DHCPv4 클라이언트에 대한 IPv4 예약을 정의합니다.

DHCP_IP_RESERVATION_V4 DHCP_IP_RESERVATION_V4 구조는 클라이언트 IP 예약을 정의합니다. 이 구조는 예약을 보유하는 클라이언트 유형(DHCP 또는 BOOTP)을 포함하여 IP 예약을 확장합니다.
DHCP_IP_RESERVATION_V6 특정 IPv6 접두사에서 DHCPv6 클라이언트에 대한 IPv6 예약을 정의합니다.
DHCP_IPV6_ADDRESS DHCP_IPV6_ADDRESS 구조체에는 IPv6 주소가 포함되어 있습니다.
DHCP_MIB_INFO 현재 DHCP 서비스에 대한 DHCP 관련 MIB(SNMP 관리 정보 블록)에서 반환된 정보를 정의합니다.
DHCP_MIB_INFO_V5 DHCP 서버에 대한 통계 정보를 포함합니다.
DHCP_MIB_INFO_V6 DHCPv6 서버에 대한 통계를 포함합니다.
DHCP_OPTION DHCP_OPTION 구조체는 단일 DHCP 옵션과 연결된 모든 데이터 요소를 정의합니다.
DHCP_OPTION_ARRAY DHCP_OPTION_ARRAY 구조체는 DHCP 서버 옵션의 배열을 정의합니다.
DHCP_OPTION_DATA DHCP_OPTION_DATA 구조는 DHCP 옵션과 연결된 하나 이상의 데이터 요소에 대한 데이터 컨테이너를 정의합니다.
DHCP_OPTION_DATA_ELEMENT DHCP_OPTION_DATA_ELEMENT 구조체는 DHCP_OPTION_DATA 구조 내에 있는 데이터 요소(배열의 멤버 또는 배열의 멤버)를 정의합니다.
DHCP_OPTION_LIST DHCP 옵션 값 목록(연결된 ID 태그가 있는 옵션 데이터만)을 정의합니다.
DHCP_OPTION_SCOPE_INFO

DHCP_OPTION_SCOPE_INFO 구조체는 특정 DHCP scope 제공된 옵션에 대한 정보를 정의합니다.
DHCP_OPTION_SCOPE_INFO6 DHCP 옵션 scope 연결된 데이터를 정의합니다.
DHCP_OPTION_VALUE DHCP_OPTION_VALUE 구조체는 DHCP 옵션 값(연결된 ID 태그가 있는 옵션 데이터만)을 정의합니다.
DHCP_OPTION_VALUE_ARRAY DHCP_OPTION_VALUE_ARRAY 구조체는 DHCP 옵션 값 목록(연결된 ID 태그가 있는 옵션 데이터만)을 정의합니다.
DHCP_POL_COND DHCP_POL_COND 구조는 DHCP 서버 정책 조건을 정의합니다.
DHCP_POL_COND_ARRAY DHCP_POL_COND_ARRAY 구조는 DHCP 서버 정책 조건의 배열을 정의합니다.
DHCP_POL_EXPR DHCP_POL_EXP 구조는 DHCP 서버 정책 식을 정의합니다.
DHCP_POL_EXPR_ARRAY DHCP_POL_EXPR_ARRAY 구조체는 DHCP 서버 정책 식의 배열을 정의합니다.
DHCP_POLICY DHCP_POLICY 구조는 DHCP 서버 정책을 정의합니다.
DHCP_POLICY_ARRAY DHCP_POLICY_ARRAY 구조는 DHCP 서버 정책의 배열을 정의합니다.
DHCP_RESERVATION_INFO_ARRAY DHCP_RESERVATION_INFO_ARRAY 구조는 DHCPv4 클라이언트에 대한 IPv4 예약 배열을 정의합니다.
DHCP_RESERVED_SCOPE DHCP_RESERVED_SCOPE 구조체는 예약된 DHCP scope 정의합니다.

DHCP_SCAN_ITEM DHCP_SCAN_ITEM 구조는 DHCPv4 서버에 저장된 비동기화된 클라이언트 임대 주소와 고정해야 하는 위치(메모리 내 캐시 또는 데이터베이스)를 정의합니다.
DHCP_SCAN_LIST 고정해야 하는 DHCPv4 서버의 모든 비동기 클라이언트 임대 IP 주소 목록을 정의합니다.
DHCP_SEARCH_INFO DHCP_SEARCH_INFO 구조는 특정 서버 작업을 검색하는 데 사용되는 DHCP 클라이언트 레코드 데이터를 정의합니다.
DHCP_SEARCH_INFO_V6 DHCPv6 서버 데이터베이스를 검색할 용어 또는 값을 포함합니다.
DHCP_SERVER_CONFIG_INFO DHCP_SERVER_CONFIG_INFO 구조는 DHCP 서버를 구성하는 데 사용되는 데이터를 정의합니다.
DHCP_SERVER_CONFIG_INFO_V4 DHCP 서버를 구성하는 데 사용되는 데이터를 정의합니다.
DHCP_SERVER_CONFIG_INFO_V6 DHCPv6 서버에 대한 설정을 포함합니다.
DHCP_SERVER_CONFIG_INFO_VQ DHCP 서버에 대한 설정을 정의합니다.
DHCP_SERVER_SPECIFIC_STRINGS 사용자 및 공급업체 클래스 이름에 대한 기본 문자열 값을 포함합니다.
DHCP_SUBNET_ELEMENT_DATA 서브넷의 기능 또는 제한을 설명하는 요소를 정의합니다. (DHCP_SUBNET_ELEMENT_DATA)
DHCP_SUBNET_ELEMENT_DATA_V4 서브넷의 기능 또는 제한을 설명하는 요소를 정의합니다. (DHCP_SUBNET_ELEMENT_DATA_V4)
DHCP_SUBNET_ELEMENT_DATA_V5

DHCP_SUBNET_ELEMENT_DATA_V5 구조체는 서브넷의 기능 또는 제한을 설명하는 요소를 정의합니다.
DHCP_SUBNET_ELEMENT_DATA_V6 IPv6 예약, IPv6 제외 범위 및 IPv6 범위와 같은 IPv6 접두사 요소에 대한 정의를 포함합니다.
DHCP_SUBNET_ELEMENT_INFO_ARRAY 서브넷 요소 데이터의 배열을 정의합니다.
DHCP_SUBNET_ELEMENT_INFO_ARRAY_V4 서브넷 요소 데이터의 배열을 정의합니다. V4 구조의 요소 데이터에는 클라이언트 형식 정보가 포함됩니다.
DHCP_SUBNET_ELEMENT_INFO_ARRAY_V5 DHCP_SUBNET_ELEMENT_INFO_ARRAY_V5 구조체는 서브넷 요소 데이터의 배열을 정의합니다. V5 구조의 요소 데이터는 BOOTP에 따라 다릅니다.
DHCP_SUBNET_ELEMENT_INFO_ARRAY_V6 DHCP_SUBNET_ELEMENT_DATA_V6 IPv6 접두사 요소의 배열을 정의하는 데이터를 포함합니다.
DHCP_SUBNET_INFO DHCP_SUBNET_INFO 구조체는 서브넷을 설명하는 정보를 정의합니다.
DHCP_SUBNET_INFO_V6 IPv6 서브넷에 대한 정보를 포함합니다.
DHCP_SUBNET_INFO_VQ 서브넷을 설명하는 정보를 정의합니다.
DHCP_SUPER_SCOPE_TABLE DHCP 서버의 대범위 정의
DHCP_SUPER_SCOPE_TABLE_ENTRY 대범위 테이블 내에서 서브넷 항목을 정의합니다.
DHCPDS_SERVER DHCPDS_SERVER 구조는 디렉터리 서비스의 컨텍스트에서 DHCP 서버에 대한 정보를 정의합니다.

DHCPDS_SERVERS DHCPDS_SERVERS 구조는 디렉터리 서비스의 컨텍스트에서 DHCP 서버 목록을 정의합니다.
DHCPV4_FAILOVER_CLIENT_INFO DHCPV4_FAILOVER_CLIENT_INFO 구조는 장애 조치(failover) 관계의 일부인 DHCP 서버 scope 통계를 정의합니다.
DHCPV4_FAILOVER_CLIENT_INFO_ARRAY DHCPV4_FAILOVER_CLIENT_INFO_ARRAY 구조는 장애 조치(failover) 관계의 일부인 DHCP 서버 scope 통계의 배열을 정의합니다.
DHCPV6_BIND_ELEMENT DHCPv6 패킷을 수신하는 DHCP 서버에 대한 IPv6 인터페이스 바인딩을 정의합니다.
DHCPV6_BIND_ELEMENT_ARRAY DHCPv6 인터페이스 바인딩을 포함하는 DHCPV6_BIND_ELEMENT 구조체의 배열을 지정합니다.
DHCPV6_IP_ARRAY DHCPV6_IP_ARRAY 구조체에는 DHCP IPv6 주소 구조의 배열이 포함되어 있습니다.
DHCPV6_STATELESS_PARAMS DHCPV6_STATELESS_PARAMS 구조는 서버 및 scope 수준에서 DHCPv6 상태 비정상 클라이언트 인벤토리 구성 설정을 정의합니다.
DHCPV6_STATELESS_SCOPE_STATS DHCPV6_STATELESS_SCOPE_STATS 구조체는 특정 IPv6 상태 비저장 서브넷에 대한 주소 카운터를 정의합니다. 상태 비정상 클라이언트 인벤토리에서 추가 및 제거된 상태 비 상태 비 상태 IPv6 클라이언트의 수는 이 구조에 저장됩니다.
DHCPV6_STATELESS_STATS DHCPV6_STATELESS_STATS 구조체는 상태 비저장 IPv6 서브넷 통계의 배열을 정의합니다.
DWORD_DWORD DWORD_DWORD 구조체는 64비트 정수 값을 정의합니다.
SCOPE_MIB_INFO 반환된 DHCP 관련 MIB(SNMP 관리 정보 블록) 데이터 내에서 사용할 수 있는 scope 대한 정보를 정의합니다.

SCOPE_MIB_INFO_V5 특정 DHCP scope 대한 정보를 포함합니다.

열거형

DHCP_FAILOVER_MODE DHCP_FAILOVER_MODE 열거형은 장애 조치(failover) 관계에서 DHCPv4 서버 모드 작업을 정의합니다.
DHCP_FAILOVER_SERVER DHCP_FAILOVER_SERVER 열거형은 DHCP 서버가 DHCPv4 장애 조치(failover) 관계의 주 서버인지 보조 서버인지를 정의합니다.
DHCP_FILTER_LIST_TYPE DHCP 서버에서 사용할 수 있는 필터 목록의 유형을 지정합니다.
DHCP_FORCE_FLAG DHCP_FORCE_FLAG 열거형은 DHCP 서브넷 요소 삭제 작업의 힘 수준을 설명하는 플래그 집합을 정의합니다.
DHCP_OPTION_DATA_TYPE DHCP_OPTION_DATA_TYPE 열거형은 DHCP 옵션 데이터를 나타내는 형식 집합을 정의합니다.
DHCP_OPTION_SCOPE_TYPE DHCP_OPTION_SCOPE_TYPE 열거형은 가능한 DHCP 옵션 범위 집합을 정의합니다.
DHCP_OPTION_SCOPE_TYPE6 DHCP 옵션에 대해 가능한 범위 집합을 정의합니다.
DHCP_OPTION_TYPE DHCP_OPTION_TYPE 열거형은 가능한 DHCP 옵션 형식 집합을 정의합니다.
DHCP_POL_ATTR_TYPE DHCP_POL_ATTR_TYPE 열거형은 DHCP 서버 정책의 조건에 대한 특성 형식을 정의합니다.

DHCP_POL_COMPARATOR DHCP_POL_COMPARATOR 열거형은 DHCP 서버 정책을 빌드할 때 조건에 대한 비교 연산자를 정의합니다.
DHCP_POL_LOGIC_OPER DHCP_POL_LOGIC_OPER 열거형은 DHCP 서버 정책에서 식의 구성 조건 및 하위 식을 그룹화하는 방법을 정의합니다.
DHCP_POLICY_FIELDS_TO_UPDATE DHCP_POLICY_FIELDS_TO_UPDATE 열거형은 업데이트해야 하는 DHCP 서버 정책의 속성을 정의합니다.
DHCP_SCAN_FLAG DHCP_SCAN_FLAG 열거형은 데이터베이스 검사 작업 중에 동기화할 수 있는 대상 집합을 정의합니다.
DHCP_SEARCH_INFO_TYPE DHCP 클라이언트 정보 레코드를 검색하는 데 사용되는 가능한 특성 집합을 정의합니다.
DHCP_SEARCH_INFO_TYPE_V6 DHCPv6 클라이언트 정보 레코드를 검색하는 데 사용되는 가능한 특성 집합을 정의합니다.
DHCP_SUBNET_ELEMENT_TYPE DHCP_SUBNET_ELEMENT_TYPE 열거형은 가능한 서브넷 요소 형식 집합을 정의합니다.
DHCP_SUBNET_STATE DHCP_SUBNET_STATE 열거형은 서브넷에 대해 가능한 상태 집합을 정의합니다.
DHCPV6_STATELESS_PARAM_TYPE DHCPV6_STATELESS_PARAM_TYPE 열거형은 DHCPv6 상태 비저장 클라이언트 인벤토리 구성 매개 변수 형식을 정의합니다.
FSM_STATE FSM_STATE 열거형은 DHCPv4 서버에서 가능한 장애 조치(failover) 관계 상태 집합을 정의합니다.
QuarantineStatus QuarantineStatus 열거형은 NAP 서버에서 유효성을 검사한 대로 DHCPv4 클라이언트에 대한 가능한 상태 상태 값을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dhcpssdk.h 헤더

아티클2023. 08. 23.

이 헤더는 DHCP(동적 호스트 구성 프로토콜)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DHCP(Dynamic Host Configuration Protocol)

dhcpssdk.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

콜백 함수

LPDHCP_CONTROL DhcpControlHook 함수는 DHCP 서버 서비스가 시작, 중지, 일시 중지 또는 계속될 때 Microsoft DHCP 서버에서 호출됩니다.
LPDHCP_DELETE_CLIENT DhcpDeleteClientHook 함수는 클라이언트 임대가 활성 임대 데이터베이스에서 삭제되기 직전에 Microsoft DHCP 서버에서 호출됩니다.
LPDHCP_DROP_SEND LPDHCP_DROP_SEND 콜백 함수
LPDHCP_ENTRY_POINT_FUNC DhcpServerCalloutEntry 함수는 타사 DLL을 초기화하고 타사 DLL에서 알림을 원하는 이벤트를 검색하기 위해 Microsoft DHCP Server에서 호출합니다. DhcpServerCalloutEntry 함수는 타사 DLL에 의해 구현됩니다.
LPDHCP_GIVE_ADDRESS DhcpAddressOfferHook 함수는 Microsoft DHCP 서버가 ACK(승인)를 DHCP REQUEST 메시지로 보내기 직전에 Microsoft DHCP 서버에서 호출합니다.
LPDHCP_HANDLE_OPTIONS DhcpHandleOptionsHook 함수를 사용하면 타사 DLL이 DHCP 패킷에서 일반적으로 사용되는 옵션을 가져올 수 있으므로 전체 DHCP 패킷을 처리할 필요가 없습니다. DhcpHandleOptionsHook 함수는 차단해서는 안 됩니다.
LPDHCP_NEWPKT

DhcpNewPktHook 함수는 처리 예정 DHCP 패킷을 받은 직후 Microsoft DHCP 서버에서 호출됩니다.
LPDHCP_PROB DhcpAddressDelHook 함수는 다음 네 개의 정의된 이벤트 중 하나가 발생할 때 Microsoft DHCP 서버에서 호출됩니다.

구조체

DHCP_CALLOUT_TABLE DHCP_CALLOUT_TABLE 구조는 Microsoft DHCP 서버 및 타사 DLL에서 DHCP 서버 이벤트에 대한 알림 요청을 보내는 데 사용됩니다.
DHCP_SERVER_OPTIONS DHCP_SERVER_OPTIONS 구조는 요청된 DHCP 서버 옵션을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dhcpv6csdk.h 헤더

아티클2023. 08. 23.

이 헤더는 DHCP(동적 호스트 구성 프로토콜)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DHCP(Dynamic Host Configuration Protocol)

dhcpv6csdk.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

Dhcpv6CApiCleanup Dhcpv6CApiCleanup 함수를 사용하면 DHCPv6 함수 호출을 사용하는 동안 할당된 리소스를 제대로 클린 수 있습니다. Dhcpv6CApiCleanup 함수는 Dhcpv6CApiInitialize에 대한 이전 호출이 성공적으로 실행된 경우에만 호출되어야 합니다.
Dhcpv6CApiInitialize Dhcpv6CApiInitialize 함수는 DHCPv6 사용자가 수행한 첫 번째 함수 호출이어야 합니다.
Dhcpv6ReleasePrefix 접두사를 해제합니다.
Dhcpv6RenewPrefix 접두사를 갱신합니다.
Dhcpv6RequestParams DHCPv6 클라이언트 캐시 또는 DHCPv6 서버에서 직접 옵션을 요청합니다.
Dhcpv6RequestPrefix 특정 접두사를 요청합니다.

구조체

DHCPV6CAPI_CLASSID IPv6 클라이언트 클래스 ID를 정의합니다.
DHCPV6CAPI_PARAMS 요청된 매개 변수를 포함합니다.
DHCPV6CAPI_PARAMS_ARRAY 요청된 매개 변수의 배열을 포함합니다.
DHCPV6Prefix DHCPV6Prefix에는 IPv6 접두사를 포함합니다.
DHCPV6PrefixLeaseInformation 접두사 임대에 대한 정보입니다.

열거형

StatusCode StatusCode 열거형에는 IPv6 작업에 대한 상태 코드가 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

diagnosticdataquery.h 헤더

아티클2023. 08. 23.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

diagnosticdataquery.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DdqCancelDiagnosticRecordOperation 이 세션에 대한 모든 미해결 진단 데이터 쿼리 API 내부 쿼리 작업을 취소합니다. 이는 장기 실행 쿼리 API를 중단하기 위해 다른 스레드에서 호출할 수 있습니다.
DdqCloseSession 진단 데이터 쿼리 세션 핸들을 닫습니다.
DdqCreateSession 진단 데이터 쿼리 세션을 고유하게 식별하는 데 사용할 진단 데이터 쿼리 API 세션 핸들을 만듭니다.
DdqExtractDiagnosticReport Windows 오류 보고 보고서를 검색하는 데 사용되는 이 API는 지정된 대상 경로로 cab을 추출합니다. 오류 보고서에 cabs가 없으면 작업이 수행되지 않습니다.
DdqFreeDiagnosticRecordLocaleTags HDIAGNOSTIC_EVENT_TAG_DESCRIPTION 핸들에서 참조하는 태그 정보에 할당된 메모리를 해제합니다.
DdqFreeDiagnosticRecordPage HDIAGNOSTIC_RECORD 핸들에서 참조하는 진단 레코드 페이지에 할당된 메모리를 해제합니다.
DdqFreeDiagnosticRecordProducerCategories 범주 집합에 할당된 메모리와 HDIAGNOSTIC_EVENT_CATEGORY_DESCRIPTION 핸들에서 참조하는 범주의 텍스트 표현을 해제합니다.

DdqFreeDiagnosticRecordProducers HDIAGNOSTIC_EVENT_PRODUCER_DESCRIPTION 핸들에서 참조하는 생산자 집합에 할당된 메모리를 해제합니다.
DdqFreeDiagnosticReport HDIAGNOSTIC_REPORT_DATA 핸들에서 참조하는 오류 보고서에 할당된 메모리를 해제합니다.
DdqGetDiagnosticDataAccessLevelAllowed API 호출자에 대해 사용 가능한 가장 높은 데이터 액세스 수준을 반환합니다. NoData, CurrentUserData 또는 AllUserData일 수 있습니다.
DdqGetDiagnosticRecordAtIndex HDIAGNOSTIC_DATA_RECORD 핸들이 가리키는 리소스의 지정된 인덱스에서 진단 데이터 레코드 정보를 가져옵니다.
DdqGetDiagnosticRecordBinaryDistribution 총 예상 업로드 크기에 따라 상위 N개의 시끄러운 이진 파일에 대한 진단 데이터 이벤트 볼륨의 이진 이름 및 연결된 예상 총 업로드를 바이트 단위로 가져옵니다. 여기서 N은 topNBinaries에 전달된 값입니다.
DdqGetDiagnosticRecordCategoryAtIndex HDIAGNOSTIC_DATA_EVENT_CATEGORY_DESCRIPTION 핸들이 가리키는 리소스의 지정된 인덱스에서 진단 레코드 범주를 가져옵니다.
DdqGetDiagnosticRecordCategoryCount HDIAGNOSTIC_EVENT_CATEGORY_DESCRIPTION 핸들이 가리키는 리소스에서 진단 레코드 범주의 수(크기)를 가져옵니다.
DdqGetDiagnosticRecordCount HDIAGNOSTIC_DATA_RECORD 핸들이 가리키는 리소스의 요소 수(크기)를 가져옵니다.
DdqGetDiagnosticRecordLocaleTagAtIndex HDIAGNOSTIC_DATA_EVENT_TAG_DESCRIPTION 핸들이 가리키는 리소스의 지정된 인덱스에서 태그 설명을 가져옵니다.
DdqGetDiagnosticRecordLocaleTagCount HDIAGNOSTIC_DATA_EVENT_TAG_DESCRIPTION 핸들에서 가리키는 리소스의 태그 수(크기)를 가져옵니다.

DdqGetDiagnosticRecordLocaleTags 지정된 로캘 아래의 알려진 모든 태그에 대한 정보를 가져오고 데이터에 대한 핸들(HDIAGNOSTIC_EVENT_TAG_DESCRIPTION)을 제공합니다. 예제 로캘은 "en-US"입니다. 반환 값의 예는 다음 데이터를 포함하는 DIAGNOSTIC_EVENT_TAG_DESCRIPTION 리소스입니다. 태그: 11, 이름: "디바이스 연결 및 구성" 및 설명: "디바이스 식별자(예: IP 주소) 구성, 설정 및 성능을 포함하여 서비스와 네트워크에 연결된 디바이스의 연결 및 구성을 설명하는 데이터".
DdqGetDiagnosticRecordPage 필터링된 레코드의 페이지(일괄 처리)를 가져옵니다. 반환된 레코드에 대한 필터링은 searchCriteria, pageRecordCount, offset 및 baseRowId를 DIAGNOSTIC_DATA_SEARCH_CRITERIA 입력 매개 변수를 사용하여 내부적으로 수행됩니다.
DdqGetDiagnosticRecordPayload rowId로 지정된 이벤트 레코드에 대한 페이로드 텍스트를 가져옵니다.
DdqGetDiagnosticRecordProducerAtIndex HDIAGNOSTIC_EVENT_PRODUCER_DESCRIPTION 핸들이 가리키는 리소스의 지정된 인덱스에서 생산자에 대한 설명을 가져옵니다.
DdqGetDiagnosticRecordProducerCategories 생산자와 범주에는 계층적 관계가 있습니다. 즉, 범주는 생산자에 속합니다. 이 함수는 지정된 진단 생산자 이름에 사용할 수 있는 범주 ID 및 범주의 텍스트 표현을 가져옵니다.
DdqGetDiagnosticRecordProducerCount HDIAGNOSTIC_EVENT_PRODUCER_DESCRIPTION 가리키는 리소스에서 생산자의 수(크기)를 가져옵니다.
DdqGetDiagnosticRecordProducers 진단 데이터 쿼리 세션에 사용할 수 있는 진단 데이터 생산자를 가져옵니다.
DdqGetDiagnosticRecordStats 필터링된 이벤트 기록 진단 데이터 레코드 통계를 가져옵니다. 반환된 통계에 대한 필터링은 입력 매개 변수 DIAGNOSTIC_DATA_SEARCH_CRITERIA 필터를 사용하여 수행됩니다. 레코드 상태는 사용 가능한 검색 조건과 일치하는 레코드 수를 설명하고 데이터의 추가 쿼리에 사용되는 매개 변수를 반환합니다. 이 API의 사용 중 하나는 데이터를 마지막으로 쿼리한 이후 변경 내용이 있는지 검사 것입니다. 출력 매개 변수의 변경은 이벤트 기록 레코드 상태의 상태 변경을 나타냅니다.
DdqGetDiagnosticRecordSummary 생산자가 필터링할 수 있는 진단 데이터 레코드에 대한 일반 통계를 가져옵니다.

DdqGetDiagnosticRecordTagDistribution 지정된 생산자 이름에 따라 개인 정보 태그 이벤트 배포 통계당 진단 데이터 이벤트를 가져옵니다.
DdqGetDiagnosticReport HDIAGNOSTIC_REPORT_DATA 핸들을 통해 이 PC에서 업로드하기 위해 업로드되거나 큐에 추가된 오류 보고서를 가져옵니다.
DdqGetDiagnosticReportAtIndex HDIAGNOSTIC_REPORT_DATA 핸들이 가리키는 리소스의 지정된 인덱스에서 오류 보고서 및 해당 정보를 가져옵니다.
DdqGetDiagnosticReportCount HDIAGNOSTIC_REPORT_DATA 핸들이 가리키는 리소스에서 오류 보고서의 수(크기)를 가져옵니다.
DdqGetDiagnosticReportStoreReportCount 요청된 저장소에 저장된 보고서의 수(크기)를 가져옵니다.
DdqGetSessionAccessLevel 현재 진단 데이터 쿼리 세션의 데이터 액세스 수준을 반환합니다.
DdqGetTranscriptConfiguration 최대 스토리지 크기 및 데이터 기록 시간과 같은 이벤트 기록 구성을 가져옵니다.
DdqIsDiagnosticRecordSampledIn 이벤트에 대한 디바이스의 샘플링된 상태를 가져옵니다.
DdqSetTranscriptConfiguration 최대 스토리지 크기 및 데이터 기록 시간과 같은 이벤트 기록 구성을 설정합니다. 사용자가 상승하지 않으면 구성 설정이 실패합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

diagnosticdataquerytypes.h 헤더

아티클2023. 08. 23.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

diagnosticdataquerytypes.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

DIAGNOSTIC_DATA_EVENT_BINARY_STATS 이 이진 파일 및 전송된 진단 데이터의 양을 설명하는 리소스입니다.
DIAGNOSTIC_DATA_EVENT_CATEGORY_DESCRIPTION 식별자와 이름으로 정의된 범주를 나타내는 리소스입니다. 범주는 지정된 생산자에 대한 레코드를 분류하는 조직 구문입니다. 예를 들어 "검색 데이터"는 생산자 "Microsoft Edge"의 범주일 수 있습니다.
DIAGNOSTIC_DATA_EVENT_PRODUCER_DESCRIPTION 생산자를 나타내는 리소스입니다. 생산자는 이벤트를 내보내는 OS 구성 요소, 애플리케이션 또는 서비스입니다. 예를 들어 "Microsoft Edge"는 Microsoft Edge 브라우저의 생산자 ID입니다.
DIAGNOSTIC_DATA_EVENT_TAG_DESCRIPTION 태그의 이름과 설명으로 정의된 태그를 설명하는 리소스입니다.
DIAGNOSTIC_DATA_EVENT_TAG_STATS 개인 정보 태그 및 이 개인 정보 태그가 있는 이벤트 수를 포함하는 리소스입니다.
DIAGNOSTIC_DATA_EVENT_TRANSCRIPT_CONFIGURATION 최대 스토리지 크기 및 데이터 기록 시간과 같은 이벤트 기록 구성 세부 정보입니다.
DIAGNOSTIC_DATA_GENERAL_STATS 이 리소스에는 진단 데이터 레코드 집합에 대한 일반 통계가 포함되어 있습니다.
DIAGNOSTIC_DATA_RECORD 이 리소스는 개별 진단 데이터 레코드(이벤트)를 설명합니다.

DIAGNOSTIC_DATA_SEARCH_CRITERIA 이 리소스에는 진단 데이터 레코드를 가져올 때 검색 조건의 세부 정보가 포함됩니다.
DIAGNOSTIC_REPORT_DATA 이 리소스에는 진단 보고서에 대한 정보가 포함되어 있습니다.
DIAGNOSTIC_REPORT_PARAMETER 오류 보고서에 대한 매개 변수를 설명하는 리소스입니다.
DIAGNOSTIC_REPORT_SIGNATURE 이 리소스는 진단 보고서의 서명을 설명합니다.

열거형

DdqAccessLevel 이 리소스는 진단 데이터 쿼리 세션에 대한 권한 수준을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

digitalv.h 헤더

아티클2023. 08. 23.

이 헤더는 Windows 멀티미디어에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 멀티미디어

digitalv.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

MCI_DGV_CAPTURE_PARMSA ANSI(MCI_DGV_CAPTURE_PARMSA) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_CAPTURE 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_CAPTURE_PARMSW MCI_DGV_CAPTURE_PARMSW(유니코드) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_CAPTURE 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_COPY_PARMS MCI_DGV_COPY_PARMS 구조에는 디지털 비디오 디바이스에 대한 MCI_COPY 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_CUE_PARMS MCI_DGV_CUE_PARMS 구조에는 디지털 비디오 디바이스에 대한 MCI_CUE 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_CUT_PARMS MCI_DGV_CUT_PARMS 구조에는 디지털 비디오 디바이스에 대한 MCI_CUT 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_DELETE_PARMS MCI_DGV_DELETE_PARMS 구조에는 디지털 비디오 디바이스에 대한 MCI_DELETE 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_INFO_PARMSA ANSI(MCI_DGV_INFO_PARMSA) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_INFO 명령에 대한 매개 변수가 포함되어 있습니다.

MCI_DGV_INFO_PARMSW MCI_DGV_INFO_PARMSW(유니코드) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_INFO 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_LIST_PARMSA ANSI(MCI_DGV_LIST_PARMSA) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_LIST 명령에 대한 정보가 포함되어 있습니다.
MCI_DGV_LIST_PARMSW MCI_DGV_LIST_PARMSW(유니코드) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_LIST 명령에 대한 정보가 포함되어 있습니다.
MCI_DGV_MONITOR_PARMS MCI_DGV_MONITOR_PARMS 구조에는 MCI_MONITOR 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_OPEN_PARMSA ANSI(MCI_DGV_OPEN_PARMSA) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_OPEN 명령에 대한 정보가 포함되어 있습니다.
MCI_DGV_OPEN_PARMSW MCI_DGV_OPEN_PARMSW(유니코드) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_OPEN 명령에 대한 정보가 포함되어 있습니다.
MCI_DGV_PASTE_PARMS MCI_DGV_PASTE_PARMS 구조에는 디지털 비디오 디바이스에 대한 MCI_PASTE 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_QUALITY_PARMSA ANSI(MCI_DGV_QUALITY_PARMSA) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_QUALITY 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_QUALITY_PARMSW MCI_DGV_QUALITY_PARMSW(유니코드) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_QUALITY 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_RECORD_PARMS MCI_DGV_RECORD_PARMS 구조에는 디지털 비디오 디바이스에 대한 MCI_RECORD 명령에 대한 매개 변수가 포함되어 있습니다.

MCI_DGV_RECT_PARMS MCI_DGV_RECT_PARMS 구조에는 디지털 비디오 디바이스에 대한 MCI_FREEZE, MCI_PUT, MCI_UNFREEZE 및 MCI_WHERE 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_RESERVE_PARMSA ANSI(MCI_DGV_RESERVE_PARMSA) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_RESERVE 명령에 대한 정보가 포함되어 있습니다.
MCI_DGV_RESERVE_PARMSW MCI_DGV_RESERVE_PARMSW(유니코드) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_RESERVE 명령에 대한 정보가 포함되어 있습니다.
MCI_DGV_RESTORE_PARMSA ANSI(MCI_DGV_RESTORE_PARMSA) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_RESTORE 명령에 대한 정보가 포함되어 있습니다.
MCI_DGV_RESTORE_PARMSW MCI_DGV_RESTORE_PARMSW(유니코드) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_RESTORE 명령에 대한 정보가 포함되어 있습니다.
MCI_DGV_SAVE_PARMSA ANSI(MCI_DGV_SAVE_PARMSA) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_SAVE 명령에 대한 정보가 포함되어 있습니다.
MCI_DGV_SAVE_PARMSW MCI_DGV_SAVE_PARMSW(유니코드) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_SAVE 명령에 대한 정보가 포함되어 있습니다.
MCI_DGV_SET_PARMS MCI_DGV_SET_PARMS 구조에는 디지털 비디오 디바이스에 대한 MCI_SET 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_SETAUDIO_PARMSA ANSI(MCI_DGV_SETAUDIO_PARMSA) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_SETAUDIO 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_SETAUDIO_PARMSW MCI_DGV_SETAUDIO_PARMSW(유니코드) 구조체(digitalv.h)에는 디지털 비디오 디바이스에 대한 MCI_SETAUDIO 명령에 대한 매개 변수가 포함되어 있습니다.

MCI_DGV_SETVIDEO_PARMSA ANSI(MCI_DGV_SETVIDEO_PARMSA) 구조에는 디지털 비디오 디바이스에 대한 MCI_SETVIDEO 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_SETVIDEO_PARMSW MCI_DGV_SETVIDEO_PARMSW(유니코드) 구조에는 디지털 비디오 디바이스에 대한 MCI_SETVIDEO 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_SIGNAL_PARMS MCI_DGV_SIGNAL_PARMS 구조에는 디지털 비디오 디바이스에 대한 MCI_SIGNAL 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_STATUS_PARMSA ANSI(MCI_DGV_STATUS_PARMSA) 구조에는 디지털 비디오 디바이스에 대한 MCI_STATUS 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_STATUS_PARMSW MCI_DGV_STATUS_PARMSW(유니코드) 구조에는 디지털 비디오 디바이스에 대한 MCI_STATUS 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_STEP_PARMS MCI_DGV_STEP_PARMS 구조에는 디지털 비디오 디바이스에 대한 MCI_STEP 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_UPDATE_PARMS MCI_DGV_UPDATE_PARMS 구조에는 MCI_UPDATE 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_WINDOW_PARMSA ANSI(MCI_DGV_WINDOW_PARMSA) 구조에는 디지털 비디오 디바이스에 대한 MCI_WINDOW 명령에 대한 매개 변수가 포함되어 있습니다.
MCI_DGV_WINDOW_PARMSW MCI_DGV_WINDOW_PARMSW(유니코드) 구조에는 디지털 비디오 디바이스에 대한 MCI_WINDOW 명령에 대한 매개 변수가 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dimm.h 헤더

아티클2023. 06. 08.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

dimm.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

LOGFONTA ANSI(LOGFONTA) 구조체는 글꼴의 문자 셀 또는 문자 특성을 정의합니다. (LOGFONTA)
LOGFONTW LOGFONTW(유니코드) 구조체는 글꼴의 문자 셀 또는 문자 특성을 정의합니다. (LOGFONTW)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dinput.h 헤더

아티클2023. 08. 23.

DirectInput 포함 파일입니다.

자세한 내용은 다음을 참조하세요.

- 휴먼 인터페이스 디바이스 참조

dinput.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

DIEFFESCAPE DIEFFESCAPE 구조는 하드웨어 관련 데이터를 디바이스 드라이버에 직접 전달합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dinputd.h 헤더

아티클2023. 08. 23.

DirectInput에는 디바이스 드라이버 구현자에 대한 파일이 포함되어 있습니다.

자세한 내용은 다음을 참조하세요.

- 휴먼 인터페이스 디바이스 참조

dinputd.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDirectInputEffectDriver 이러한 세 가지 메서드를 사용하면 원래 인터페이스의 기능에 영향을 주지 않고 DirectInputEffectDriver 개체에 추가 인터페이스를 추가할 수 있습니다.
IDirectInputJoyConfig8 IDirectInputJoyConfig8 인터페이스에는 속성 시트를 작성하는 하드웨어 개발자가 레지스트리에 정보를 쓰고 읽을 수 있는 메서드가 포함되어 있습니다.

구조체

DIDEVICESTATE DIDEVICESTATE 구조체는 힘 피드백 디바이스의 상태에 대한 정보를 반환합니다.
디드리버버전 DIDRIVERVERSIONS 구조체는 DirectInput 효과 드라이버에서 버전 정보를 DirectInput에 다시 보고하는 데 사용됩니다.
DIEFFECTATTRIBUTES DIEFFECTATTRIBUTES 구조체는 힘 피드백 디바이스에서 지원하는 각 효과에 대한 레지스트리 키의 "특성" 값에 포함된 정보를 설명합니다.
DIFFDEVICEATTRIBUTES

DIFFDEVICEATTRIBUTES 구조체는 OEMForceFeedback 레지스트리 키의 "특성" 값에 포함된 정보를 설명합니다.
DIFFOBJECTATTRIBUTES DIFFOBJECTATTRIBUTES 구조체는 힘 피드백 디바이스의 각 "개체"에 대한 레지스트리 키의 "FFAttributes" 값에 포함된 정보를 설명합니다.
디히드피니틴포 DIHIDFFINITINFO 구조체는 DirectInput에서 제어하도록 요청하는 디바이스에 대한 HID 힘 피드백 드라이버에 정보를 제공하는 데 사용됩니다.
DIJOYCONFIG DIJOYCONFIG 구조체에는 조이스틱의 구성에 대한 정보가 포함되어 있습니다.
DIJOYTYPEINFO DIJOYTYPEINFO 구조체에는 조이스틱 형식에 대한 정보가 포함되어 있습니다.
DIJOYUSERVALUES DIJOYUSERVALUES 구조체에는 사용자의 조이스틱 설정에 대한 정보가 포함되어 있습니다.
DIOBJECTATTRIBUTES DIOBJECTATTRIBUTES 구조체는 디바이스의 각 "개체"에 대한 레지스트리 키의 "특성" 값에 포함된 정보를 설명합니다. "특성" 값이 없으면 기본 특성이 사용됩니다.
DIOBJECTCALIBRATION DIOBJECTCALIBRATION 구조체는 디바이스의 각 축에 대한 레지스트리 키의 "보정" 값에 포함된 정보를 설명합니다.
JOYREGHWVALUES JOYREGHWVALUES(dinputd.h) 구조체에는 하드웨어에서 반환하는 값 범위(보정으로 채워짐)가 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

directmanipulation.h 헤더

아티클2023. 03. 11.

이 헤더는 직접 조작에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 직접 조작

directmanipulation.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDirectManipulationAutoScrollBehavior 지정된 축 또는 축의 경계에 접근할 때 콘텐츠의 자동 스크롤 애니메이션 동작을 나타냅니다.
IDirectManipulationCompositor 조작된 콘텐츠를 그리기 화면(예: 캔버스(JavaScript를 사용하는 Windows 앱) 또는 Canvas(C++, C# 또는 Visual Basic을 사용하는 Windows 스토어 앱)와 연결하는 작성자 개체를 나타냅니다.
IDirectManipulationCompositor2 조작된 콘텐츠를 여러 프로세스의 그리기 표면과 연결하는 작성자 개체를 나타냅니다.
IDirectManipulationContent 뷰포트 내의 콘텐츠를 캡슐화합니다. 여기서 콘텐츠는 뷰포트 내부에서 잘린 시각적 표면을 나타냅니다.
IDirectManipulationDeferContactService 연락처와 뷰포트 간의 연결을 관리하기 위한 서비스를 나타냅니다.
IDirectManipulationDragDropBehavior 교차 슬라이드 또는 길게 누르기 제스처에 의해 트리거되는 끌어서 놓기 상호 작용에 대한 동작을 나타냅니다.
IDirectManipulationDragDropEventHandler 끌어서 놓기 동작 이벤트를 처리하는 메서드를 정의합니다.

IDirectManipulationFrameInfoProvider 애플리케이션에서 사용하는 컴퍼지션 인프라의 대기 시간을 측정하고 이 데이터를 직접 조작에 제공하는 시간 유지 개체를 나타냅니다.
IDirectManipulationInteractionEventHandler 검색될 때 상호 작용을 처리하는 메서드를 정의합니다.
IDirectManipulationManager 클라이언트 애플리케이션에서 사용할 수 있는 모든 직접 조작 기능 및 API에 대한 액세스를 제공합니다.
IDirectManipulationManager2 클라이언트 애플리케이션에서 사용할 수 있는 모든 직접 조작 기능 및 API에 대한 액세스를 제공하는 IDirectManipulationManager 인터페이스를 확장합니다.
IDirectManipulationManager3 클라이언트 애플리케이션에서 사용할 수 있는 모든 직접 조작 기능 및 API에 대한 액세스를 제공하는 IDirectManipulationManager2 인터페이스를 확장합니다.
IDirectManipulationPrimaryContent 뷰포트 내의 기본 콘텐츠를 캡슐화합니다.
IDirectManipulationUpdateHandler 조작 업데이트 이벤트를 처리하는 메서드를 정의합니다.
IDirectManipulationUpdateManager 작성기 업데이트를 직접 조작으로 보내는 방법을 관리합니다.
IDirectManipulationViewport 사용자 상호 작용에서 입력을 수신하고 처리할 수 있는 창(뷰포트라고 함) 내의 지역을 정의합니다.
IDirectManipulationViewport2 뷰포트에서 동작 관리를 제공합니다. 동작은 직접 조작 워크플로의 특정 부분 기능에 영향을 줍니다.
IDirectManipulationViewportEventHandler 뷰포트의 상태 및 업데이트 이벤트를 처리하는 메서드를 정의합니다.

열거형

DIRECTMANIPULATION_AUTOSCROLL_CONFIGURATION 적용할 자동 스크롤 애니메이션의 유형과 방향을 결정합니다.
DIRECTMANIPULATION_CONFIGURATION 직접 조작에서 사용할 수 있는 상호 작용 구성 상태를 정의합니다.
DIRECTMANIPULATION_DRAG_DROP_CONFIGURATION 끌어서 놓기 상호 작용에 대한 동작을 정의합니다.
DIRECTMANIPULATION_DRAG_DROP_STATUS 뷰포트의 끌어서 놓기 상호 작용 상태를 정의합니다.
DIRECTMANIPULATION_GESTURE_CONFIGURATION SetManualGesture에 전달할 수 있는 제스처를 정의합니다.
DIRECTMANIPULATION_HITTEST_TYPE RegisterHitTestTarget을 통해 등록된 전용 적중 테스트 스레드를 사용할 때 직접 조작에서 적중 테스트를 처리하는 방법을 정의합니다.
DIRECTMANIPULATION_HORIZONTALALIGNMENT 뷰포트 내의 콘텐츠에 대한 가로 맞춤 옵션을 정의합니다.
DIRECTMANIPULATION_INPUT_MODE SetInputMode 또는 SetUpdateMode에 대한 스레딩 동작을 정의합니다. 각 상수의 정확한 의미는 호출된 메서드에 따라 달라집니다.
DIRECTMANIPULATION_INTERACTION_TYPE 직접 조작으로 인식되는 제스처를 정의합니다.
DIRECTMANIPULATION_MOTION_TYPES 직접 조작 동작 유형을 정의합니다.
DIRECTMANIPULATION_SNAPPOINT_COORDINATE 맞춤 지점 컬렉션의 좌표계를 정의합니다.

DIRECTMANIPULATION_SNAPPOINT_TYPE 최종 관성 끝 위치를 계산하는 방법을 수정합니다.
DIRECTMANIPULATION_STATUS 직접 조작의 가능한 상태를 정의합니다.
DIRECTMANIPULATION_VERTICALALIGNMENT 뷰포트 내의 콘텐츠에 대한 세로 맞춤 설정을 정의합니다.
DIRECTMANIPULATION_VIEWPORT_OPTIONS 뷰포트에 대한 입력 동작 옵션을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

directml.h 헤더

아티클2024. 11. 13.

이 헤더는 DirectML에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DirectML

directml.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

테이블 확장

IDMLBindingTable 애플리케이션 관리 설명자 힙의 범위를 래핑하고 DirectML에서 리소스에 대한 바인딩을 만드는 데 사용됩니다. 이 개체를 만들려면 IDMLDevice::CreateBindingTable을 호출합니다.
IDMLCommandRecorder DirectML의 디스패치를 Direct3D 12 명령 목록으로 기록합니다.
IDMLCompiledOperator GPU에서 실행하기에 적합한 컴파일된 효율적인 연산자 형식을 나타냅니다. 이 개체를 만들려면 IDMLDevice::CompileOperator를 호출합니다.
IDMLDebugDevice DirectML 디버그 계층을 제어합니다.
IDMLDevice 연산자, 바인딩 테이블, 명령 레코더 및 기타 개체를 만드는 데 사용되는 DirectML 디바이스를 나타냅니다. (IDMLDevice)
IDMLDevice1 연산자, 바인딩 테이블, 명령 레코더 및 기타 개체를 만드는 데 사용되는 DirectML 디바이스를 나타냅니다. (IDMLDevice1)
IDMLDeviceChild DirectML 디바이스에서 만든 모든 개체에 의해 구현되는 인터페이스입니다.

IDMLDispatchable IDMLCommandRecorder::RecordDispatch를 사용하여 GPU에서 디스패치를 위해 명령 목록에 기록할 수 있는 개체에 의해 구현됩니다.
IDMLObject IDMLDevice 및 IDMLDeviceChild가 직접 상속하는 인터페이스(및 다른 모든 인터페이스는 간접적으로).
IDMLOperator DirectML 연산자를 나타냅니다.
IDMLOperatorInitializer 컴파일된 연산자를 초기화하는 특수한 개체를 나타냅니다. 이 개체의 인스턴스를 만들려면 IDMLDevice::CreateOperatorInitializer를 호출합니다.
IDMLPageable GPU 메모리에서 제거할 수 있으므로 IDMLDevice::Evict 및 IDMLDevice::MakeResident에 제공할 수 있는 개체에 의해 구현됩니다.

함수

테이블 확장

DMLCreateDevice 지정된 Direct3D 12 디바이스에 대한 DirectML 디바이스를 만듭니다. (DMLCreateDevice)
DMLCreateDevice1 지정된 Direct3D 12 디바이스에 대한 DirectML 디바이스를 만듭니다. (DMLCreateDevice1)

구조

테이블 확장

DML_ACTIVATION_CELU_OPERATOR_DESC *InputTensor*모든 요소에 대해 지속적으로 다른 CELU(지수 선형 단위) 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_ELU_OPERATOR_DESC *InputTensor*모든 요소에 대해 ELU(지수 선형 단위) 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_HARD_SIGMOID_OPERATOR_DESC *InputTensor*모든 요소에 대해 하드 시그모이드 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_HARDMAX_OPERATOR_DESC *InputTensor*각 요소에 대해 하드맥스 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_IDENTITY_OPERATOR_DESC id 활성화를 수행하여 *InputTensor* 모든 요소를 *OutputTensor*해당 요소로 효과적으로 복사합니다.
DML_ACTIVATION_LEAKY_RELU_OPERATOR_DESC *InputTensor*모든 요소에 대해 새는 ReLU(정정 선형 단위) 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_LINEAR_OPERATOR_DESC *InputTensor*모든 요소에 대해 선형 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_LOG_SOFTMAX_OPERATOR_DESC *InputTensor*각 요소에 대해 (자연) 소프트맥스 로그 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_PARAMETERIZED_RELU_OPERATOR_DESC *InputTensor*모든 요소에 대해 매개 변수가 있는 ReLU(정정 선형 단위) 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_PARAMETRIC_SOFTPLUS_OPERATOR_DESC *InputTensor*모든 요소에 대해 매개 변수 소프트플러스 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다. (DML_ACTIVATION_PARAMETRIC_SOFTPLUS_OPERATOR_DESC)

DML_ACTIVATION_RELU_GRAD_OPERATOR_DESC ReLU(정정 선형 단위)에 대한 백프로포지션 그라데이션을 계산합니다.
DML_ACTIVATION_RELU_OPERATOR_DESC *InputTensor*모든 요소에 대해 ReLU(정정 선형 단위) 활성화 함수를 수행하여 결과를 *OutputTensor* 해당 요소에 배치합니다.
DML_ACTIVATION_SCALED_ELU_OPERATOR_DESC *InputTensor*모든 요소에 대해 확장된 ELU(지수 선형 단위) 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_SCALED_TANH_OPERATOR_DESC *InputTensor*모든 요소에 대해 확장된 하이퍼볼릭 탄젠트 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_SHRINK_OPERATOR_DESC *InputTensor*모든 요소에 대해 축소 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_SIGMOID_OPERATOR_DESC *InputTensor*모든 요소에 대해 시그모이드 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_SOFTMAX_OPERATOR_DESC *InputTensor*대해 softmax 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_SOFTPLUS_OPERATOR_DESC *InputTensor*모든 요소에 대해 매개 변수 소프트플러스 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다. (DML_ACTIVATION_SOFTPLUS_OPERATOR_DESC)
DML_ACTIVATION_SOFTSIGN_OPERATOR_DESC *InputTensor*모든 요소에 대해 softsign 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ACTIVATION_TANH_OPERATOR_DESC *InputTensor*모든 요소에 대해 하이퍼볼릭 탄젠트 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.

DML_ACTIVATION_THRESHOLDED_RELU_OPERATOR_DESC *InputTensor*모든 요소에 대해 임계값이 지정된 ReLU(정정 선형 단위) 활성화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ADAM_OPTIMIZER_OPERATOR_DESC Adam(**ADA**ptive **M**오멘트 추정) 알고리즘을 기반으로 제공된 그라데이션을 사용하여 업데이트된 가중치(매개 변수)를 계산합니다. 이 연산자는 최적화 프로그램이며 일반적으로 학습 루프의 가중치 업데이트 단계에서 그라데이션 하강을 수행하는 데 사용됩니다.
DML_ARGMAX_OPERATOR_DESC 입력 텐서의 하나 이상의 차원 내에서 최대값 요소의 인덱스를 출력합니다.
DML_ARGMIN_OPERATOR_DESC 입력 텐서의 하나 이상의 차원 내에서 최소값 요소의 인덱스를 출력합니다.
DML_AVERAGE_POOLING_GRAD_OPERATOR_DESC 평균 풀링에 대한 백프로포지션 그라데이션을 계산합니다 (DML_AVERAGE_POOLING_OPERATOR_DESC참조).
DML_AVERAGE_POOLING_OPERATOR_DESC 입력 텐서의 슬라이딩 윈도우 내 요소에 대한 값의 평균을 계산합니다.
DML_BATCH_NORMALIZATION_GRAD_OPERATOR_DESC 일괄 정규화대한 백프로포지션 그라데이션을 계산합니다.
DML_BATCH_NORMALIZATION_OPERATOR_DESC DML_BATCH_NORMALIZATION_OPERATOR_DESC 구조체(directml.h)는 입력에 대한 일괄 처리 정규화를 수행합니다.
DML_BATCH_NORMALIZATION_TRAINING_GRAD_OPERATOR_DESC 일괄 처리 정규화 학습대한 백프로포지션 그라데이션을 계산합니다.
DML_BATCH_NORMALIZATION_TRAINING_OPERATOR_DESC DML_BATCH_NORMALIZATION_TRAINING_OPERATOR_DESC 구조체(directml.h)는 입력에 대한 일괄 처리 정규화를 수행합니다.
DML_BINDING_DESC

IDMLBindingTable 메서드 중 하나에 대한 호출을 통해 바인딩 테이블에 추가할 수 있도록 바인딩에 대한 설명을 포함합니다.
DML_BINDING_PROPERTIES 컴파일된 특정 연산자 또는 연산자 이니셜라이저의 바인딩 요구 사항에 대한 정보를 포함합니다. 이 구조체는 IDMLDispatchable::GetBindingProperties에서 검색됩니다.
DML_BINDING_TABLE_DESC IDMLDevice::CreateBindingTable 및 IDMLBindingTable::Reset에 대한 매개 변수를 지정합니다.
DML_BUFFER_ARRAY_BINDING 개별 버퍼 바인딩의 배열인 리소스 바인딩을 지정합니다.
DML_BUFFER_BINDING ID3D12Resource에 대한 오프셋 및 크기로 표현되는 Direct3D 12 버퍼의 바이트 범위로 설명된 리소스 바인딩을 지정합니다.
DML_BUFFER_TENSOR_DESC Direct3D 12 버퍼 리소스에 저장될 텐서에 대해 설명합니다.
DML_CAST_OPERATOR_DESC 입력의 각 요소를 출력 텐서의 데이터 형식으로 캐스팅하고 결과를 출력의 해당 요소에 저장합니다.
DML_CONVOLUTION_INTEGER_OPERATOR_DESC *InputTensor*사용하여 *FilterTensor* 구성을 수행합니다. 이 연산자는 정수 데이터에 대해 정방향 연결을 수행합니다.
DML_CONVOLUTION_OPERATOR_DESC *InputTensor*사용하여 *FilterTensor* 구성을 수행합니다. 이 연산자는 다양한 표준 나선형 구성을 지원합니다.
DML_CUMULATIVE_PRODUCT_OPERATOR_DESC 축을 따라 텐서의 요소를 곱하여 제품의 실행 중인 집계를 출력 텐서에 기록합니다.
DML_CUMULATIVE_SUMMATION_OPERATOR_DESC 축을 따라 텐서의 요소를 합산하여 합계의 실행 집계를 출력 텐서에 기록합니다.
DML_DEPTH_TO_SPACE_OPERATOR_DESC

깊이에서 공간 데이터 블록으로 데이터를 다시 정렬(permutes)합니다. 연산자는 깊이 차원의 값이 공간 블록에서 높이 및 너비 차원으로 이동되는 입력 텐서의 복사본을 출력합니다. (DML_DEPTH_TO_SPACE_OPERATOR_DESC)
DML_DEPTH_TO_SPACE1_OPERATOR_DESC 깊이에서 공간 데이터 블록으로 데이터를 다시 정렬(permutes)합니다. 연산자는 깊이 차원의 값이 공간 블록에서 높이 및 너비 차원으로 이동되는 입력 텐서의 복사본을 출력합니다. (DML_DEPTH_TO_SPACE1_OPERATOR_DESC)
DML_DIAGONAL_MATRIX_OPERATOR_DESC 주 대각선에 있는 ID와 같은 행렬(또는 기타 명시적 값)을 생성하고 다른 곳에서는 0을 생성합니다.
DML_DYNAMIC_QUANTIZE_LINEAR_OPERATOR_DESC InputTensor정량화하는 데 필요한 수량화 배율과 0포인트 값을 계산한 다음, 해당 양자화를 적용하여 결과를 outputTensor.
DML_ELEMENT_WISE_ABS_OPERATOR_DESC *InputTensor*각 요소에 대한 절대값을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_ACOS_OPERATOR_DESC *InputTensor*각 요소에 대한 아크코사인을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_ACOSH_OPERATOR_DESC *InputTensor*각 요소에 대한 하이퍼볼릭 아크코사인을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_ADD_OPERATOR_DESC *ATensor* 모든 요소를 *BTensor*해당 요소에 추가하여 결과를 *OutputTensor*해당 요소에 배치합니다. (DML_ELEMENT_WISE_ADD_OPERATOR_DESC)
DML_ELEMENT_WISE_ADD1_OPERATOR_DESC *ATensor* 모든 요소를 *BTensor* 해당 요소에 추가하고 결과를 *OutputTensor*해당 요소에 배치하고 융합 활성화 옵션을 사용합니다.
DML_ELEMENT_WISE_ASIN_OPERATOR_DESC *InputTensor*각 요소에 대한 아크사인을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_ASINH_OPERATOR_DESC

*InputTensor*각 요소에 대한 하이퍼볼릭 아크사인을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_ATAN_OPERATOR_DESC *InputTensor*각 요소에 대한 아크탄젠트를 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_ATAN_YX_OPERATOR_DESC *ATensor* 및 *BTensor*각 요소에 대해 2 인수 아크탄젠트를 계산합니다. 여기서 *ATensorY*축 및 *BTensorX*축결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_ATANH_OPERATOR_DESC *InputTensor*각 요소에 대한 하이퍼볼릭 아크탄젠트를 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_BIT_AND_OPERATOR_DESC 입력 텐서의 각 해당 요소 간에 비트 AND를 계산하고 결과를 출력 텐서에 씁니다.
DML_ELEMENT_WISE_BIT_COUNT_OPERATOR_DESC 입력 텐서의 각 요소에 대한 비트 모집단 수(1로 설정된 비트 수)를 계산하고 결과를 출력 텐서에 씁니다.
DML_ELEMENT_WISE_BIT_NOT_OPERATOR_DESC 입력 텐서의 각 요소에 대해 비트 NOT을 계산하고 결과를 출력 텐서에 씁니다.
DML_ELEMENT_WISE_BIT_OR_OPERATOR_DESC 입력 텐서의 각 해당 요소 간에 비트 OR를 계산하고 결과를 출력 텐서에 씁니다.
DML_ELEMENT_WISE_BIT_SHIFT_LEFT_OPERATOR_DESC *BTensor*해당 요소에서 제공하는 비트 수만큼 *ATensor* 각 요소의 논리적 왼쪽 시프트를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_BIT_SHIFT_RIGHT_OPERATOR_DESC *BTensor*해당 요소에서 제공하는 비트 수만큼 *ATensor* 각 요소의 논리적 오른쪽 이동을 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_BIT_XOR_OPERATOR_DESC 입력 텐서의 각 해당 요소 간에 비트 XOR(eXclusive OR)를 계산하고 결과를 출력 텐서에 씁니다.
DML_ELEMENT_WISE_CEIL_OPERATOR_DESC

*InputTensor*각 요소의 최대값을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다. x의 최대값은 x보다 크거나 같은 가장 작은 정수입니다.
DML_ELEMENT_WISE_CLIP_GRAD_OPERATOR_DESC 요소별 클립백프로포지션 그라데이션을 계산합니다.
DML_ELEMENT_WISE_CLIP_GRAD1_OPERATOR_DESC 요소별 클립백프로포지션 그라데이션을 계산합니다.
DML_ELEMENT_WISE_CLIP_OPERATOR_DESC *InputTensor*각 요소에 대해 다음 작업을 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다. 이 연산자는 닫힌 간격 [최소, 최대] 내에서 입력의 모든 요소를 클램프(또는 제한)합니다.
DML_ELEMENT_WISE_CLIP1_OPERATOR_DESC *InputTensor*각 요소에 대해 클램핑(또는 제한) 작업을 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_CONSTANT_POW_OPERATOR_DESC *InputTensor* 각 요소를 *지수OutputTensor*해당 요소에 넣습니다.
DML_ELEMENT_WISE_COS_OPERATOR_DESC *InputTensor*각 요소의 삼각 코사인을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_COSH_OPERATOR_DESC *InputTensor*각 요소의 하이퍼볼릭 코사인을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_DEQUANTIZE_LINEAR_OPERATOR_DESC *ScaleTensor* 및 `ZeroPointTensor`해당 요소와 관련하여 *InputTensor* 모든 요소에 대해 다음 선형 정수화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_DIFFERENCE_SQUARE_OPERATOR_DESC *ATensor*해당 요소에서 *BTensor* 각 요소를 빼고 결과 자체를 곱하고 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_DIVIDE_OPERATOR_DESC *BTensor*해당 요소에 대해 *ATensor* 각 요소의 몫을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_ERF_OPERATOR_DESC

*InputTensor*각 요소에 대해 erf(Gaussian 오류 함수)를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_EXP_OPERATOR_DESC *InputTensor*각 요소에 자연 지수 함수를 적용하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_FLOOR_OPERATOR_DESC *InputTensor*각 요소의 바닥을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_IDENTITY_OPERATOR_DESC *InputTensor*각 요소에 대한 ID를 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_IF_OPERATOR_DESC *ConditionTensor*해당 요소의 값에 따라 *ATensor* 또는 *BTensor*요소를 선택합니다. *ConditionTensor의 0이 아닌 요소는 ATensor선택할* 있지만 값이 0인 요소는 *BTensor*선택합니다.
DML_ELEMENT_WISE_IS_INFINITY_OPERATOR_DESC 지정된 *InfinityMode*따라 IEEE-754 -inf, inf 또는 둘 다에 대해 *InputTensor* 각 요소를 확인하고 결과(true의 경우 1, false의 경우 0)를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_IS_NAN_OPERATOR_DESC 입력 텐서의 각 요소에 대해 입력이 NaN(IEEE-754에 정의된 대로)이면 1을 반환하고, 그렇지 않으면 0을 반환합니다. 결과는 출력 텐서의 해당 요소에 배치됩니다.
DML_ELEMENT_WISE_LOG_OPERATOR_DESC *InputTensor*각 요소의 기본 e(자연) 로그를 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_LOGICAL_AND_OPERATOR_DESC 입력 텐서의 해당 요소 쌍마다 논리 AND를 수행하여 결과(true의 경우 1, false의 경우 0)를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_LOGICAL_EQUALS_OPERATOR_DESC 입력 텐서의 해당 요소 쌍마다 논리 수행하여 결과(true의 경우 1, false의 경우 0)를 OutputTensor해당 요소에 배치합니다.
DML_ELEMENT_WISE_LOGICAL_GREATER_THAN_OPERATOR_DESC 입력 텐서의 해당 요소 쌍마다보다 큰 논리 수행하여 결과(true의 경우 1, false의 경우 0)를 OutputTensor해당 요소에 배치합니다.

DML_ELEMENT_WISE_LOGICAL_GREATER_THAN_OR_EQUAL_OPERATOR_DESC 입력 텐서의 해당 요소 쌍에서 보다 크거나 같은 논리 수행하여 결과(true의 경우 1, false의 경우 0)를 OutputTensor해당 요소에 배치합니다.
DML_ELEMENT_WISE_LOGICAL_LESS_THAN_OPERATOR_DESC 입력 텐서의 해당 요소 쌍마다 미만의 논리 수행하여 결과(true의 경우 1, false의 경우 0)를 OutputTensor해당 요소에 배치합니다.
DML_ELEMENT_WISE_LOGICAL_LESS_THAN_OR_EQUAL_OPERATOR_DESC 입력 텐서의 해당 요소 쌍마다 보다 작거나 같은 논리 수행하여 결과(true의 경우 1, false의 경우 0)를 OutputTensor해당 요소에 배치합니다.
DML_ELEMENT_WISE_LOGICAL_NOT_OPERATOR_DESC *InputTensor*각 요소에 대해 논리 NOT을 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_LOGICAL_OR_OPERATOR_DESC 입력 텐서의 해당 요소 쌍마다 논리 OR을 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_LOGICAL_XOR_OPERATOR_DESC 입력 텐서의 해당 요소 쌍마다 논리 XOR(배타적 또는)를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_MAX_OPERATOR_DESC 입력 텐서에서 두 개의 해당 요소 중 더 큰 요소를 사용하여 결과를 출력 텐서의 해당 요소에 배치합니다.
DML_ELEMENT_WISE_MEAN_OPERATOR_DESC 입력 텐서의 해당 요소 쌍의 각 쌍을 평균하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_MIN_OPERATOR_DESC 입력 텐서에서 해당하는 두 요소 중 더 작은 요소를 가져와 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_MODULUS_FLOOR_OPERATOR_DESC 입력 텐서에서 해당하는 각 요소 쌍에 대해 Python 모듈러스와 동일한 결과를 사용하여 모듈러스를 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_MODULUS_TRUNCATE_OPERATOR_DESC

입력 텐서의 해당 요소 쌍마다 C 모듈러스 연산자를 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_MULTIPLY_OPERATOR_DESC 입력 텐서의 해당 요소 쌍의 각 쌍의 곱을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_NEGATE_OPERATOR_DESC *InputTensor*각 요소를 부정하고 결과를 *OutputTensor*해당 요소에 저장합니다.
DML_ELEMENT_WISE_POW_OPERATOR_DESC ExponentTensor해당 요소의 성능으로 InputTensor의 각 요소를 계산하여 결과를 OutputTensor해당 요소에 배치합니다.
DML_ELEMENT_WISE_QUANTIZE_LINEAR_OPERATOR_DESC *ScaleTensor* 및 `ZeroPointTensor`해당 요소와 관련하여 *InputTensor* 모든 요소에 대해 다음 선형 양자화 함수를 수행하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_QUANTIZED_LINEAR_ADD_OPERATOR_DESC *ATensor* 모든 요소를 *BTensor*해당 요소에 추가하여 결과를 *OutputTensor*해당 요소에 배치합니다. (DML_ELEMENT_WISE_QUANTIZED_LINEAR_ADD_OPERATOR_DESC)
DML_ELEMENT_WISE_RECIP_OPERATOR_DESC 입력 텐서의 각 요소에 대한 상호 계산을 수행하여 결과를 출력 텐서의 해당 요소에 배치합니다.
DML_ELEMENT_WISE_ROUND_OPERATOR_DESC *InputTensor* 각 요소를 정수 값으로 반올림하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_SIGN_OPERATOR_DESC *InputTensor*각 요소의 기호를 나타내는 값을 반환하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_SIN_OPERATOR_DESC *InputTensor*각 요소의 삼각 사인을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_SINH_OPERATOR_DESC *InputTensor*각 요소의 하이퍼볼릭 사인을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_SQRT_OPERATOR_DESC

*InputTensor*각 요소의 제곱근을 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_SUBTRACT_OPERATOR_DESC *ATensor*해당 요소에서 *BTensor* 각 요소를 뺍니다. 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_TAN_OPERATOR_DESC *InputTensor*각 요소의 삼각 탄젠트를 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_TANH_OPERATOR_DESC *InputTensor*요소의 하이퍼볼릭 탄젠트를 계산하여 결과를 *OutputTensor*해당 요소에 배치합니다.
DML_ELEMENT_WISE_THRESHOLD_OPERATOR_DESC 지정된 임계값 *Min*아래의 *InputTensor* 모든 요소를 *Min*바꿉니다. 결과는 *OutputTensor*해당 요소에 배치됩니다.
DML_FEATURE_DATA_FEATURE_LEVELS DirectML 디바이스에서 지원하는 기능 수준에 대한 세부 정보를 제공합니다.
DML_FEATURE_DATA_TENSOR_DATA_TYPE_SUPPORT DirectML 디바이스가 텐서 내의 특정 데이터 형식을 지원하는지 여부에 대한 세부 정보를 제공합니다.
DML_FEATURE_QUERY_FEATURE_LEVELS 하나 이상의 기능 수준에 대한 지원을 위해 DirectML 디바이스를 쿼리하는 데 사용됩니다.
DML_FEATURE_QUERY_TENSOR_DATA_TYPE_SUPPORT 텐서 내의 특정 데이터 형식에 대한 지원을 위해 DirectML 디바이스를 쿼리하는 데 사용됩니다.
DML_FILL_VALUE_CONSTANT_OPERATOR_DESC 지정된 상수 *값*텐서를 채웁니다.
DML_FILL_VALUE_SEQUENCE_OPERATOR_DESC 텐서를 시퀀스로 채웁니다.
DML_GATHER_ELEMENTS_OPERATOR_DESC 인덱스 텐서를 사용하여 지정된 축을 따라 입력 텐서에서 요소를 수집하여 입력에 다시 매핑합니다.

DML_GATHER_ND_OPERATOR_DESC 인덱스 텐서를 사용하여 입력 텐서에서 요소를 수집하여 인덱스를 입력의 전체 하위 블록에 다시 매핑합니다. (DML_GATHER_ND_OPERATOR_DESC)
DML_GATHER_ND1_OPERATOR_DESC 인덱스 텐서를 사용하여 입력 텐서에서 요소를 수집하여 인덱스를 입력의 전체 하위 블록에 다시 매핑합니다. (DML_GATHER_ND1_OPERATOR_DESC)
DML_GATHER_OPERATOR_DESC **IndicesTensor** 사용하여 **Axis**따라 입력 텐서에서 요소를 수집하여 인덱스를 다시 매핑합니다.
DML_GEMM_OPERATOR_DESC `Output = FusedActivation(Alpha * TransA(A) x TransB(B) + Beta * C)`양식의 일반 행렬 곱하기 함수를 수행합니다. 여기서 `x` 행렬 곱셈을 표시하고 `*` 스칼라를 곱한 값을 나타냄
DML_GRAPH_DESC 최적화된 결합 연산자를 컴파일하는 데 사용되는 DirectML 연산자의 그래프를 설명합니다.
DML_GRAPH_EDGE_DESC DML_GRAPH_DESC 정의하고 IDMLDevice1::CompileGraph전달된 DirectML 연산자의 그래프 내 연결에 대한 일반 컨테이너입니다.
DML_GRAPH_NODE_DESC DML_GRAPH_DESC 정의하고 IDMLDevice1::CompileGraph전달된 DirectML 연산자의 그래프 내 노드에 대한 제네릭 컨테이너입니다.
DML_GRU_OPERATOR_DESC 입력에서 (표준 계층) 1층 제어 GRU(되풀이 단위) 함수를 수행합니다. 이 연산자는 여러 게이트를 사용하여 이 계층을 수행합니다. 이러한 게이트는 시퀀스 길이 차원과 *SequenceLengthsTensor*따라 루프에서 여러 번 수행됩니다.
DML_INPUT_GRAPH_EDGE_DESC DML_GRAPH_DESC 정의하고 IDMLDevice1::CompileGraph전달된 DirectML 연산자의 그래프 내에서 연결을 설명합니다. 이 구조체는 그래프 입력에서 내부 노드의 입력으로의 연결을 정의하는 데 사용됩니다.
DML_INTERMEDIATE_GRAPH_EDGE_DESC

DML_GRAPH_DESC 정의하고 IDMLDevice1::CompileGraph전달된 DirectML 연산자의 그래프 내에서 연결을 설명합니다. 이 구조체는 내부 노드 간의 연결을 정의하는 데 사용됩니다.
DML_JOIN_OPERATOR_DESC 지정된 축을 따라 입력 텐서 배열을 연결합니다.
DML_LOCAL_RESPONSE_NORMALIZATION_GRAD_OPERATOR_DESC 로컬 응답 정규화대한 백프로포지션 그라데이션을 계산합니다.
DML_LOCAL_RESPONSE_NORMALIZATION_OPERATOR_DESC 입력에서 LRN(로컬 응답 정규화) 함수를 수행합니다.
DML_LP_NORMALIZATION_OPERATOR_DESC 입력 텐서의 지정된 축을 따라 Lp 정규화 함수를 수행합니다.
DML_LP_POOLING_OPERATOR_DESC 입력 텐서를 통해 슬라이딩 윈도우 내의 요소에서 Lp 정규화된 값을 계산합니다.
DML_LSTM_OPERATOR_DESC 입력에서 LSTM(1층 장기 메모리) 함수를 수행합니다. 이 연산자는 여러 게이트를 사용하여 이 계층을 수행합니다. 이러한 게이트는 시퀀스 길이 차원과 *SequenceLengthsTensor*따라 루프에서 여러 번 수행됩니다.
DML_MATRIX_MULTIPLY_INTEGER_OPERATOR_DESC 정수 데이터에 대해 행렬 곱하기 함수를 수행합니다.
DML_MAX_POOLING_GRAD_OPERATOR_DESC 최대 풀링에 대한 백프로포지션 그라데이션을 계산합니다 (DML_MAX_POOLING2_OPERATOR_DESC참조).
DML_MAX_POOLING_OPERATOR_DESC 입력 텐서를 통해 슬라이딩 윈도우 내의 요소에서 최대값을 계산합니다.
DML_MAX_POOLING1_OPERATOR_DESC 입력 텐서를 통해 슬라이딩 윈도우 내 요소의 최대값을 계산하고 선택적으로 선택한 최대값의 인덱스를 반환합니다. (DML_MAX_POOLING1_OPERATOR_DESC)
DML_MAX_POOLING2_OPERATOR_DESC

입력 텐서를 통해 슬라이딩 윈도우 내 요소의 최대값을 계산하고 선택적으로 선택한 최대값의 인덱스를 반환합니다. (DML_MAX_POOLING2_OPERATOR_DESC)
DML_MAX_UNPOOLING_OPERATOR_DESC *IndicesTensor*제공된 인덱스 값에 따라 출력 텐서 *OutputTensor* 입력 텐서 *InputTensor*값을 입력 풀링 작업에서 얻은 값으로 채워서 최대 풀링 작업(자세한 내용은 DML_MAX_POOLING_OPERATOR1_DESC 참조)을 반전합니다. 이 프로세스의 영향을 받지 않은 출력 텐서의 요소는 0 값으로 남습니다.
DML_MEAN_VARIANCE_NORMALIZATION_OPERATOR_DESC 입력 텐서에서 평균 분산 정규화 함수를 수행합니다. 이 연산자는 정규화를 수행하기 위해 입력 텐서의 평균 및 분산을 계산합니다. (DML_MEAN_VARIANCE_NORMALIZATION_OPERATOR_DESC)
DML_MEAN_VARIANCE_NORMALIZATION1_OPERATOR_DESC 입력 텐서에서 평균 분산 정규화 함수를 수행합니다. 이 연산자는 정규화를 수행하기 위해 입력 텐서의 평균 및 분산을 계산합니다. (DML_MEAN_VARIANCE_NORMALIZATION1_OPERATOR_DESC)
DML_NONZERO_COORDINATES_OPERATOR_DESC 입력 텐서의 0이 아닌 모든 요소의 N차원 좌표를 계산합니다.
DML_ONE_HOT_OPERATOR_DESC *원 핫 인코딩된* 값으로 채워진 텐서를 생성합니다. 이 연산자는 선택한 축의 모든 시퀀스에 대해 해당 시퀀스의 한 요소를 제외한 모든 요소가 OffValue설정되고 나머지 단일 요소가 OnValue설정되는 출력 텐서를 생성합니다.
DML_OPERATOR_DESC 연산자 설명에 대한 제네릭 컨테이너입니다. 이 구조체에 지정된 매개 변수를 사용하여 DirectML 연산자를 생성합니다. 자세한 내용은 IDMLDevice::CreateOperator를 참조하세요.
DML_OPERATOR_GRAPH_NODE_DESC DML_GRAPH_DESC 정의하고 IDMLDevice1::CompileGraph전달된 DirectML 연산자의 그래프 내에서 노드를 디크리브합니다.
DML_OUTPUT_GRAPH_EDGE_DESC DML_GRAPH_DESC 정의하고 IDMLDevice1::CompileGraph전달된 DirectML 연산자의 그래프 내에서 연결을 설명합니다. 이 구조체는 내부 노드의 출력에서 그래프 출력으로의 연결을 정의하는 데 사용됩니다.
DML_PADDING_OPERATOR_DESC

DML_PADDING_OPERATOR_DESC 구조체(directml.h)는 입력 텐서를 가장자리의 상수 또는 미러 값으로 팽창시키고 결과를 출력에 씁니다.
DML_PADDING1_OPERATOR_DESC DML_PADDING1_OPERATOR_DESC 구조체(directml.h)는 입력 텐서를 가장자리에 상수 또는 미러 값으로 팽창시키고 결과를 출력에 씁니다.
DML_QUANTIZED_LINEAR_CONVOLUTION_OPERATOR_DESC *InputTensor*사용하여 *FilterTensor* 구성을 수행합니다. 이 연산자는 정량화된 데이터에 대해 정방향 연결을 수행합니다. 이 연산자는 입력을 정량화하고 출력을 계산한 다음 정량화하는 것과 수학적으로 동일합니다.
DML_QUANTIZED_LINEAR_MATRIX_MULTIPLY_OPERATOR_DESC 정량화된 데이터에 대해 행렬 곱하기 함수를 수행합니다. 이 연산자는 입력을 기준으로 한 다음 행렬을 곱한 다음 출력을 정량화하는 것과 수학적으로 동일합니다.
DML_RANDOM_GENERATOR_OPERATOR_DESC 출력 텐서를 결정적으로 생성된 의사 임의 균일하게 분산된 비트로 채웁니다. 이 연산자는 필요에 따라 업데이트된 내부 생성기 상태를 출력할 수도 있습니다. 이 상태는 연산자의 후속 실행 중에 사용할 수 있습니다.
DML_REDUCE_OPERATOR_DESC 입력 텐서의 하나 이상의 차원 내에서 요소의 감소(합계, 제품, 최소값 등)를 출력합니다.
DML_RESAMPLE_GRAD_OPERATOR_DESC Resample에 대한 백프로포지션 그라데이션을 계산합니다(DML_RESAMPLE1_OPERATOR_DESC참조).
DML_RESAMPLE_OPERATOR_DESC 배율 인수를 사용하여 원본에서 대상 텐서로 요소를 다시 샘플링하여 대상 텐서 크기를 계산합니다. 선형 또는 가장 가까운 인접 보간 모드를 사용할 수 있습니다. (DML_RESAMPLE_OPERATOR_DESC)
DML_RESAMPLE1_OPERATOR_DESC 배율 인수를 사용하여 원본에서 대상 텐서로 요소를 다시 샘플링하여 대상 텐서 크기를 계산합니다. 선형 또는 가장 가까운 인접 보간 모드를 사용할 수 있습니다. (DML_RESAMPLE1_OPERATOR_DESC)
DML_REVERSE_SUBSEQUENCES_OPERATOR_DESC

텐서의 하나 이상의 하위 시퀀스의 요소를 반대로 바꿉니다. 역방향으로 사용할 하위 시퀀스 집합은 제공된 축 및 시퀀스 길이에 따라 선택됩니다.
DML_RNN_OPERATOR_DESC 입력에서 1계층 단순 RNN(되풀이 신경망) 함수를 수행합니다. 이 함수를 입력 게이트라고도 합니다. 이 연산자는 시퀀스 길이 차원과 *SequenceLengthsTensor*따라 루프에서 이 함수를 여러 번 수행합니다.
DML_ROI_ALIGN_GRAD_OPERATOR_DESC ROI_ALIGN 및 ROI_ALIGN1대한 백프로포지션 그라데이션을 계산합니다.
DML_ROI_ALIGN_OPERATOR_DESC 마스크 R-CNN 용지에 설명된 대로 ROI 맞춤 작업을 수행합니다. 요약하자면, 이 작업은 입력 이미지 텐서에서 작물을 추출하고 지정된 *InterpolationMode*사용하여 *OutputTensor* 마지막 2차원으로 지정된 일반 출력 크기로 크기를 조정합니다.
DML_ROI_ALIGN1_OPERATOR_DESC 마스크 R-CNN 용지에 설명된 대로 ROI 맞춤 작업을 수행합니다. 요약하자면, 이 작업은 입력 이미지 텐서에서 잘린 창을 추출하고 지정된 *InterpolationMode*사용하여 *OutputTensor* 마지막 2차원으로 지정된 공통 출력 크기로 크기를 조정합니다.
DML_ROI_POOLING_OPERATOR_DESC 입력 텐서에서 MaxPool 함수를 수행합니다(관심 영역 또는 UI에 따라).
DML_SCALAR_UNION 스칼라 형식의 통합입니다.
DML_SCALE_BIAS DirectML 연산자에 제공된 배율 및 바이어스 용어의 값을 포함합니다. 배율 및 바이어스는 g(x) = x * 배율 + 바이어스 함수를 적용하는 효과가 있습니다.
DML_SCATTER_ND_OPERATOR_DESC 전체 입력 텐서를 출력에 복사한 다음, 선택한 인덱스를 업데이트 텐서의 해당 값으로 덮어씁니다. (DML_SCATTER_ND_OPERATOR_DESC)
DML_SCATTER_OPERATOR_DESC 전체 입력 텐서를 출력에 복사한 다음, 선택한 인덱스를 업데이트 텐서의 해당 값으로 덮어씁니다. (DML_SCATTER_OPERATOR_DESC)
DML_SIZE_2D

텐서 내 요소의 2차원 평면이나 2차원 배율 또는 2차원 너비/높이 값의 크기(DirectML 연산자에 제공됨)를 나타낼 수 있는 값을 포함합니다.
DML_SLICE_GRAD_OPERATOR_DESC Slice에 대한 백프로포지션 그라데이션을 계산합니다(DML_SLICE1_OPERATOR_DESC참조).
DML_SLICE_OPERATOR_DESC 입력 텐서의 단일 하위 영역("조각")을 추출합니다. (DML_SLICE_OPERATOR_DESC)
DML_SLICE1_OPERATOR_DESC 입력 텐서의 단일 하위 영역("조각")을 추출합니다. (DML_SLICE1_OPERATOR_DESC)
DML_SPACE_TO_DEPTH_OPERATOR_DESC 공간 데이터 블록을 깊이로 다시 정렬합니다. 연산자는 높이 및 너비 차원의 값이 깊이 차원으로 이동되는 입력 텐서의 복사본을 출력합니다. (DML_SPACE_TO_DEPTH_OPERATOR_DESC)
DML_SPACE_TO_DEPTH1_OPERATOR_DESC 공간 데이터 블록을 깊이로 다시 정렬합니다. 연산자는 높이 및 너비 차원의 값이 깊이 차원으로 이동되는 입력 텐서의 복사본을 출력합니다. (DML_SPACE_TO_DEPTH1_OPERATOR_DESC)
DML_SPLIT_OPERATOR_DESC 축을 따라 입력 텐서를 여러 출력 텐서로 분할합니다.
DML_TENSOR_DESC DirectML 텐서 설명에 대한 제네릭 컨테이너입니다.
DML_TILE_OPERATOR_DESC 입력 텐서를 바둑판식으로 배열하여 출력 텐서를 생성합니다. 입력 텐서의 각 차원에 있는 요소는 *반복* 배열의 배수로 반복됩니다.
DML_TOP_K_OPERATOR_DESC *InputTensor*축을 따라 각 시퀀스에서 가장 큰 *K* 요소를 선택하고 *OutputValueTensor* 및 *OutputIndexTensor*해당 요소의 값과 인덱스를 각각 반환합니다.
DML_TOP_K1_OPERATOR_DESC *InputTensor*축을 따라 각 시퀀스에서 가장 크거나 작은 *K* 요소를 선택하고 *OutputValueTensor* 및 *OutputIndexTensor*해당 요소의 값과 인덱스를 각각 반환합니다.
DML_UPSAMPLE_2D_OPERATOR_DESC

입력 이미지를 업샘플링하여 결과를 출력 텐서에 기록합니다. 차원의 순서는 NCHW(BatchSize, ChannelCount, Height, Width) 또는 NCDHW(BatchSize, ChannelCount, Depth, Height, Width)여야 하지만 데이터가 다른 형식으로 저장되는 경우 보폭을 사용할 수 있습니다.
DML_VALUE_SCALE_2D_OPERATOR_DESC 요소별 배율 및 바이어스 함수 `Output = Scale * Input + Bias`수행합니다.

열거형

테이블 확장

DML_AXIS_DIRECTION 연산자에 대해 지정된 축을 따라 연산의 방향을 지정하는 상수를 정의합니다(예: 합계, top-k 요소 선택, 최소 요소 선택).
DML_BINDING_TYPE 바인딩 설명(DML_BINDING_DESC 구조)에서 참조하는 리소스의 특성을 지정하는 상수를 정의합니다.
DML_CONVOLUTION_DIRECTION DirectML 컨볼루션 연산자의 방향을 지정하는 상수를 정의합니다 (DML_CONVOLUTION_OPERATOR_DESC 구조에 설명된 대로).
DML_CONVOLUTION_MODE DirectML 구성 연산자에 대한 모드를 지정하는 상수를 정의합니다 (DML_CONVOLUTION_OPERATOR_DESC 구조에 설명된 대로).
DML_CREATE_DEVICE_FLAGS DMLCreateDevice에 추가 디바이스 만들기 옵션을 제공합니다. 값은 비트 OR를 함께 사용할 수 있습니다.
DML_DEPTH_SPACE_ORDER DirectML 연산자 DML_OPERATOR_DEPTH_TO_SPACE1 및 **DML_OPERATOR_SPACE_TO_DEPTH1**적용된 변환을 제어하는 상수 정의
DML_EXECUTION_FLAGS

DirectML에 연산자 실행을 제어하는 옵션을 제공합니다. 이러한 플래그는 한 번에 여러 플래그를 지정하기 위해 함께 비트 OR일 수 있습니다.
DML_FEATURE DirectML 디바이스에서 쿼리할 수 있는 선택적 기능 집합을 정의합니다.
DML_FEATURE_LEVEL DirectML *기능 수준*지정하는 상수를 정의합니다. 기능 수준은 DirectML에서 지원하는 광범위한 기능을 정의합니다.
DML_GRAPH_EDGE_TYPE 그래프 가장자리의 형식을 지정하는 상수를 정의합니다. 이 열거형의 사용은 DML_GRAPH_EDGE_DESC 참조하세요.
DML_GRAPH_NODE_TYPE 그래프 노드의 형식을 지정하는 상수를 정의합니다. 이 열거형의 사용은 DML_GRAPH_NODE_DESC 참조하세요.
DML_INTERPOLATION_MODE DirectML upsample 2-D 연산자의 모드를 지정하는 상수를 정의합니다 (DML_UPSAMPLE_2D_OPERATOR_DESC 구조에 설명된 대로).
DML_IS_INFINITY_MODE 무한대 기호를 지정하는 상수를 정의합니다.
DML_MATRIX_TRANSFORM DirectML 텐서에 적용할 행렬 변환을 지정하는 상수를 정의합니다.
DML_OPERATOR_TYPE 연산자 설명의 형식을 정의합니다.
DML_PADDING_MODE DirectML 패드 연산자의 모드를 지정하는 상수를 정의합니다(DML_PADDING_OPERATOR_DESC 구조에 설명된 대로).
DML_RANDOM_GENERATOR_TYPE 난수 생성기의 형식을 지정하는 상수를 정의합니다.
DML_RECURRENT_NETWORK_DIRECTION

되풀이 DirectML 연산자에 대한 방향을 지정하는 상수를 정의합니다.
DML_REDUCE_FUNCTION DirectML reduce 연산자에 사용할 특정 감소 알고리즘을 지정하는 상수를 정의합니다 (DML_REDUCE_OPERATOR_DESC 구조에 설명된 대로).
DML_ROUNDING_MODE 반올림 모드를 지정하는 상수를 정의합니다.
DML_TENSOR_DATA_TYPE 텐서에 있는 값의 데이터 형식을 지정합니다. DirectML 연산자는 모든 데이터 형식을 지원하지 않을 수 있습니다. 지원하는 데이터 형식을 찾으려면 각 특정 연산자에 대한 설명서를 참조하세요.
DML_TENSOR_FLAGS 텐서 설명에 추가 옵션을 지정합니다. 값은 비트 OR를 함께 사용할 수 있습니다.
DML_TENSOR_TYPE 텐서 설명의 형식을 식별합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

directxcollision.h 헤더

아티클2023. 08. 23.

이 헤더는 DirectXMath에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DirectXMath

directxcollision.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

ContainedBy 삼각형이 6개의 평면(일반적으로 frustum) 내에 포함되어 있는지 여부를 테스트합니다.
교차 삼각형이 광선과 교차하는지 테스트합니다.
교차 두 삼각형이 교차하는지 테스트합니다.
교차 삼각형과 평면이 교차하는지 여부를 테스트합니다.

구조체

BoundingBox 경계 축 맞춤 개체입니다.
BoundingFrustum 경계 frustum 개체입니다.
BoundingOrientedBox 지향 경계 상자 개체입니다.

BoundingSphere 경계 구 개체입니다.

열거형

ContainmentType 개체에 다른 개체가 포함되어 있는지 여부를 나타냅니다.
PlaneIntersectionType 개체가 평면과 교차하는지 여부를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

directxmath.h 헤더

아티클2023. 08. 23.

이 헤더는 DirectXMath에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DirectXMath

directxmath.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

NA:directxmath::operator- XMVECTOR instance 부정을 계산합니다.
NA:directxmath::operator- 두 번째 instance XMVECTOR의 한 instance 빼고 결과를 XMVECTOR의 새 instance 반환합니다.
NA:directxmath::operator* XMVECTOR의 한 instance 두 번째 instance 곱하여 결과를 세 번째 instance 반환합니다.
NA:directxmath::operator* XMVECTOR의 instance 부동 소수점 값으로 곱하여 결과를 XMVECTOR의 새 instance 반환합니다.
NA:directxmath::operator* 부동 소수점 값을 XMVECTOR의 instance 곱하여 결과 XMVECTOR의 새 instance 반환합니다.
NA:directxmath::operator*= XMVECTOR instance 두 번째 instance 곱하여 업데이트된 초기 instance 대한 참조를 반환합니다.
NA:directxmath::operator*= XMVECTOR instance 부동 소수점 값으로 곱하고 업데이트된 instance 대한 참조를 반환합니다.
NA:directxmath::operator/ XMVECTOR의 한 instance 두 번째 instance 나누어 결과를 세 번째 instance.o로 반환합니다.
NA:directxmath::operator/

XMVECTOR instance 부동 소수점 값으로 나누어 결과를 XMVECTOR의 새 instance 반환합니다.
NA:directxmath::operator/= 업데이트된 초기 instance 대한 참조를 반환하여 XMVECTOR instance 두 번째 instance 나눕니다.
NA:directxmath::operator/= XMVECTOR instance 부동 소수점 값으로 나누고 업데이트된 instance 대한 참조를 반환합니다.
NA:directxmath::operator+ XMVECTOR instance ID 작업을 수행합니다.
NA:directxmath::operator+ **XMVECTOR**의 두 인스턴스를 추가하여 결과를 새 instance 반환합니다.
NA:directxmath::operator+= XMVECTOR instance 부동 소수점 값을 추가하고 업데이트된 instance 대한 참조를 반환합니다.
NA:directxmath::operator-= XMVECTOR의 현재 instance 부동 소수점 값을 빼고 업데이트된 현재 instance 결과를 반환합니다.
XMColorAdjustContrast 색의 대비 값을 조정합니다.
XMColorAdjustSaturation 색의 채도 값을 조정합니다.
XMColorEqual 두 색의 같음을 테스트합니다.
XMColorGreater 첫 번째 색의 모든 구성 요소가 두 번째 색의 해당 구성 요소보다 큰지 테스트합니다.
XMColorGreaterOrEqual 첫 번째 색의 모든 구성 요소가 두 번째 색의 해당 구성 요소보다 크거나 같은지 테스트합니다.
XMColorHSLToRGB HSL 색 값을 RGB 색 값으로 변환합니다.

XMColorHSVToRGB HSV 색 값을 RGB 색 값으로 변환합니다.
XMColorIsInfinite 색의 구성 요소가 양수 또는 음의 무한대인지 여부를 테스트합니다.
XMColorIsNaN 색의 구성 요소가 숫자(NaN)가 아닌지 여부를 테스트합니다.
XMColorLess 첫 번째 색의 모든 구성 요소가 두 번째 색의 해당 구성 요소보다 작은지 테스트합니다.
XMColorLessOrEqual 첫 번째 색의 모든 구성 요소가 두 번째 색의 해당 구성 요소보다 작거나 같은지 테스트합니다.
XMColorModulate 해당 구성 요소를 함께 곱하여 두 색을 혼합합니다.
XMColorNegative 색의 음수 RGB 색 값을 결정합니다.
XMColorNotEqual 두 색이 같지 않은지 여부를 테스트합니다.
XMColorrGBToHSL RGB 색 값을 HSL 색 값으로 변환합니다.
XMColorrGBToHSV RGB 색 값을 HSV 색 값으로 변환합니다.
XMColorRGBToSRGB RGB 색 벡터를 sRGB로 변환합니다.
XMColorrGBToXYZ RGB 색 값을 XYZ 색 값으로 변환합니다.
XMColorrGBToYUV

RGB 색 값을 YUV 색 값으로 변환합니다.
XMColorRGBToYUV_HD RGB 색 값을 YUV HD 색 값으로 변환합니다.
XMColorRGBToYUV_UHD RGB 색 값을 YUV UHD 색 값으로 변환합니다.
XMColorSRGBToRGB sRGB 색 벡터를 RGB로 변환합니다.
XMColorSRGBToXYZ SRGB 색 값을 XYZ 색 값으로 변환합니다.
XMColorXYZToRGB XYZ 색 값을 RGB 색 값으로 변환합니다.
XMColorXYZToSRGB XYZ 색 값을 SRGB 색 값으로 변환합니다.
XMColorYUVToRGB YUV 색 값을 RGB 색 값으로 변환합니다.
XMColorYUVToRGB_HD YUV 색 값을 RGB HD 색 값으로 변환합니다.
XMColorYUVToRGB_UHD YUV 색 값을 RGB UHD 색 값으로 변환합니다.
XMComparisonAllFalse 비교 값을 테스트하여 비교된 모든 구성 요소가 false인지 확인합니다.
XMComparisonAllInBounds 비교 값을 테스트하여 비교된 모든 구성 요소가 설정된 범위 내에 있는지 확인합니다.
XMComparisonAllTrue 비교 값을 테스트하여 비교된 모든 구성 요소가 true인지 확인합니다.

XMComparisonAnyFalse 비교 값을 테스트하여 비교된 구성 요소가 false인지 확인합니다.
XMComparisonAnyOutOfBounds 비교 값을 테스트하여 비교된 구성 요소가 설정된 범위를 벗어났는지 확인합니다.
XMComparisonAnyTrue 비교 값을 테스트하여 비교된 구성 요소가 true인지 확인합니다.
XMComparisonMixed 비교 값을 테스트하여 비교된 구성 요소에 혼합된 결과가 있는지 확인합니다(일부 true 및 일부 false).
XMConvertToDegrees 라디안 단위로 측정된 각도를 도 단위로 측정된 각도로 변환합니다.
XMConvertToRadians 도 단위로 측정된 각도를 라디안 단위로 측정된 각도로 변환합니다.
XMConvertVectorFloatToInt float 구성 요소가 있는 XMVECTOR를 int32_t 구성 요소가 있는 XMVECTOR로 변환하고 균일한 바이어스를 적용합니다.
XMConvertVectorFloatToUInt float 구성 요소가 있는 XMVECTOR를 uint32_t 구성 요소가 있는 XMVECTOR로 변환하고 균일한 바이어스를 적용합니다.
XMConvertVectorIntToFloat int32_t 구성 요소가 있는 XMVECTOR를 float 구성 요소가 있는 XMVECTOR로 변환하고 균일한 바이어스를 적용합니다.
XMConvertVectorUIntToFloat uint32_t 구성 요소가 있는 XMVECTOR를 float 구성 요소가 있는 XMVECTOR로 변환하고 균일한 바이어스를 적용합니다.
XMFresnelTerm 극명화되지 않은 광원에 대한 프레넬 용어를 계산합니다.

XMLoadFloat 부동 소수점 스칼라 값을 XMVECTOR에 로드합니다.
XMLoadFloat2 XMFLOAT2를 XMVECTOR에 로드합니다.
XMLoadFloat2A XMFLOAT2A를 XMVECTOR에 로드합니다.
XMLoadFloat3 XMFLOAT3을 XMVECTOR에 로드합니다.
XMLoadFloat3A XMFLOAT3A를 XMVECTOR에 로드합니다.
XMLoadFloat3x3 XMFLOAT3X3을 XMMATRIX에 로드합니다.
XMLoadFloat3x4 **XMFLOAT3X4**를 **XMMATRIX**에 로드합니다.
XMLoadFloat3x4A **XMFLOAT3X4A**를 **XMMATRIX**에 로드합니다.
XMLoadFloat4 XMFLOAT4를 XMVECTOR에 로드합니다.
XMLoadFloat4A XMFLOAT4A를 XMVECTOR에 로드합니다.
XMLoadFloat4x3 XMFLOAT4X3을 XMMATRIX에 로드합니다.
XMLoadFloat4x3A XMFLOAT4X3A를 XMVECTOR에 로드합니다.
XMLoadFloat4x4

XMFLOAT4X4를 XMMATRIX에 로드합니다.
XMLoadFloat4x4A XMFLOAT4X4A를 XMVECTOR에 로드합니다.
XMLoadInt 스칼라 값을 XMVECTOR에 로드합니다.
XMLoadInt2 XMVECTOR의 x 및 y 구성 요소에 데이터를 로드합니다.
XMLoadInt2A XMVECTOR의 x 및 y 구성 요소에 16 바이트 정렬 데이터를 로드합니다.
XMLoadInt3 형식 검사 없이 XMVECTOR의 x, y 및 z 구성 요소에 데이터를 로드합니다.
XMLoadInt3A 형식 검사 없이 XMVECTOR의 x, y 및 z 구성 요소에 16바이트의 정렬된 데이터를 로드합니다.
XMLoadInt4 형식 검사 없이 XMVECTOR에 데이터를 로드합니다.
XMLoadInt4A 형식 검사 없이 16비트 정렬 데이터를 XMVECTOR에 로드합니다.
XMLoadSInt2 서명된 정수 데이터를 XMVECTOR의 x 및 y 구성 요소에 로드합니다.
XMLoadSInt3 XMVECTOR의 x, y 및 z 구성 요소에 부호 있는 정수 데이터를 로드합니다.
XMLoadSInt4 서명된 정수 데이터를 XMVECTOR에 로드합니다.
XMLoadUInt2 부호 없는 정수 데이터를 XMVECTOR의 x 및 y 구성 요소에 로드합니다.

XMLoadUInt3 형식 검사 없이 부호 없는 정수 데이터를 XMVECTOR의 x, y 및 z 구성 요소에 로드합니다.
XMLoadUInt4 부호 없는 정수 데이터를 XMVECTOR에 로드합니다.
XMMatrixAffineTransformation 아핀 변환 매트릭스를 빌드합니다.
XMMatrixAffineTransformation2D xy 평면에서 2D 아핀 변환 매트릭스를 빌드합니다.
XMMatrixDecompose 일반 3D 변환 매트릭스를 스칼라, 회전 및 변환 구성 요소로 구분합니다.
XMMatrixDeterminant 행렬의 결정자를 계산합니다.
XMMatrixIdentity ID 매트릭스를 빌드합니다.
XMMatrixInverse 행렬의 역방향을 계산합니다.
XMMatrixIsIdentity 행렬이 ID 행렬인지 여부를 테스트합니다.
XMMatrixIsInfinite 행렬의 요소가 양수 또는 음의 무한대인지 테스트합니다.
XMMatrixIsNaN 행렬의 요소가 NaN인지 여부를 테스트합니다.
XMMatrixLookAtLH 카메라 위치, 위쪽 방향 및 초점을 사용하여 왼손 좌표계용 보기 행렬을 작성합니다.
XMMatrixLookAtRH

카메라 위치, 위쪽 방향 및 초점을 사용하여 오른손 좌표계용 보기 행렬을 작성합니다.
XMMatrixLookToLH 카메라 위치, 위쪽 방향 및 카메라 방향을 사용하여 왼손 좌표계용 보기 행렬을 작성합니다.
XMMatrixLookToRH 카메라 위치, 위쪽 방향 및 카메라 방향을 사용하여 오른손 좌표계용 보기 행렬을 작성합니다.
XMMatrixMultiply 두 행렬의 곱을 계산합니다.
XMMatrixMultiplyTranspose 두 행렬의 곱의 전이를 계산합니다.
XMMatrixOrthographicLH 왼손 좌표계용 직교 투영 행렬을 작성합니다.
XMMatrixOrthographicOffCenterLH 왼손 좌표계용 사용자 지정 직교 투영 행렬을 작성합니다.
XMMatrixOrthographicOffCenterRH 오른손 좌표계용 사용자 지정 직교 투영 행렬을 작성합니다.
XMMatrixOrthographicRH 오른손 좌표계용 직교 투영 행렬을 작성합니다.
XMMatrixPerspectiveFovLH 보기의 필드를 기준으로 왼손 원근 투영 행렬을 작성합니다.
XMMatrixPerspectiveFovRH 보기의 필드를 기준으로 오른손 원근 투영 행렬을 작성합니다.
XMMatrixPerspectiveLH 왼손 원근 투영 행렬을 작성합니다.
XMMatrixPerspectiveOffCenterLH 왼손 원근 투영 행렬의 사용자 지정 버전을 작성합니다.

XMMatrixPerspectiveOffCenterRH 오른손 원근 투영 행렬의 사용자 지정 버전을 작성합니다.
XMMatrixPerspectiveRH 오른손 원근 투영 행렬을 작성합니다.
XMMatrixReflect 지정된 평면을 통해 벡터를 반영하도록 설계된 변환 매트릭스를 빌드합니다.
XMMatrixRotationAxis 임의의 축을 중심으로 회전하는 행렬을 빌드합니다.
XMMatrixRotationNormal 임의의 일반 벡터를 중심으로 회전하는 행렬을 빌드합니다.
XMMatrixRotationQuaternion 쿼터니언에서 회전 행렬을 작성합니다.
XMMatrixRotationRollPitchYaw 지정된 피치, 요, 롤(오일러 각도)을 기반으로 회전 행렬을 작성합니다.
XMMatrixRotationRollPitchYawFromVector Euler 각도(피치, 요 및 롤)를 포함하는 벡터를 기반으로 회전 행렬을 빌드합니다.
XMMatrixRotationX x축을 중심으로 회전하는 행렬을 빌드합니다.
XMMatrixRotationY y축을 중심으로 회전하는 행렬을 빌드합니다.
XMMatrixRotationZ z축을 중심으로 회전하는 행렬을 빌드합니다.
XMMatrixScaling x축, y축 및 z축을 따라 크기가 조정되는 행렬을 빌드합니다.
XMMatrixScalingFromVector

3D 벡터에서 크기 조정 매트릭스를 빌드합니다.
XMMatrixSet float 값을 사용하여 행렬을 만듭니다.
XMMatrixShadow 기하 도형을 평면으로 평면화하는 변환 행렬을 빌드합니다.
XMMatrixTransformation 변환 매트릭스를 빌드합니다.
XMMatrixTransformation2D xy 평면에서 2D 변환 매트릭스를 빌드합니다.
XMMatrixTranslation 지정된 오프셋에서 변환 행렬을 작성합니다.
XMMatrixTranslationFromVector 벡터에서 변환 행렬을 빌드합니다.
XMMatrixTranspose 행렬의 전이를 계산합니다.
XMMatrixVectorTensorProduct 두 벡터에서 행렬을 빌드합니다.
XMPlaneDot 입력 평면과 4D 벡터 사이의 점 곱을 계산합니다.
XMPlaneDotCoord 입력 평면과 3D 벡터 사이의 점 곱을 계산합니다.
XMPlaneDotNormal 평면의 일반 벡터와 3D 벡터 사이의 점 곱을 계산합니다.
XMPlaneEqual 두 평면이 같은지 여부를 확인합니다.

XMPlaneFromPointNormal 평면의 지점과 일반 벡터에서 생성된 평면의 수식을 계산합니다.
XMPlaneFromPoints 평면의 세 지점에서 생성된 평면의 수식을 계산합니다.
XMPlaneIntersectLine 평면과 선 간의 교차를 찾습니다.
XMPlaneIntersectPlane 두 평면의 교집합을 찾습니다.
XMPlaneIsInfinite 평면의 계수가 양수인지 음의 무한대인지 테스트합니다.
XMPlaneIsNaN 평면 계수가 NaN인지 여부를 테스트합니다.
XMPlaneNearEqual 두 평면이 거의 같은지 여부를 결정합니다.
XMPlaneNormalize x, y 및 z 계수가 단위 일반 벡터를 형성하도록 평면의 계수를 정규화합니다.
XMPlaneNormalizeEst x, y 및 z 계수가 단위 일반 벡터를 형성할 수 있도록 평면의 계수를 추정합니다.
XMPlaneNotEqual 두 평면이 같지 않은지 확인합니다.
XMPlaneTransform 지정된 행렬로 평면을 변환합니다.
XMPlaneTransformStream 지정된 행렬로 평면 스트림을 변환합니다.
XMQuaternionBaryCentric

지정된 쿼터니언을 사용하여 바리센트릭 좌표의 점을 반환합니다. (XMQuaternionBaryCentric)
XMQuaternionBaryCentricV 지정된 쿼터니언을 사용하여 바리센트릭 좌표의 점을 반환합니다. (XMQuaternionBaryCentricV)
XMQuaternionConjugate 쿼터니언의 결합을 계산합니다.
XMQuaternionDot 두 개의 쿼터니언의 점 곱을 계산합니다.
XMQuaternionEqual 두 쿼터니언이 같은지 테스트합니다.
XMQuaternionExp 지정된 순수 쿼터니언의 지수를 계산합니다.
XMQuaternionIdentity ID 쿼터니언을 반환합니다.
XMQuaternionInverse 쿼터니언의 역을 계산합니다.
XMQuaternionIsIdentity 특정 쿼터니언이 ID 쿼터니언인지 테스트합니다.
XMQuaternionIsInfinite 쿼터니언의 구성 요소가 양수 또는 음의 무한대인지 테스트합니다.
XMQuaternionIsNaN 쿼터니언의 구성 요소가 NaN인지 테스트합니다.
XMQuaternionLength 쿼터니언의 크기를 계산합니다.
XMQuaternionLengthSq 쿼터니언 크기의 제곱을 계산합니다.

XMQuaternionLn 지정된 단위 쿼터니언의 자연 로그를 계산합니다.
XMQuaternionMultiply 두 개의 쿼터니언의 곱을 계산합니다.
XMQuaternionNormalize 쿼터니언을 정규화합니다.
XMQuaternionNormalizeEst 정규화된 쿼터니언 버전을 예측합니다.
XMQuaternionNotEqual 두 쿼터니언이 같지 않은지 테스트합니다.
XMQuaternionReciprocalLength 쿼터니언 크기의 상호를 계산합니다.
XMQuaternionRotationAxis 축에 대한 회전 쿼터니언을 계산합니다.
XMQuaternionRotationMatrix 회전 행렬에서 회전 쿼터니언을 계산합니다.
XMQuaternionRotationNormal 일반 벡터에 대한 회전 쿼터니언을 계산합니다.
XMQuaternionRotationRollPitchYaw 피치, 요, 롤(오일러 각도)에 따라 회전 쿼터니언을 계산합니다.
XMQuaternionRotationRollPitchYawFromVector Euler 각도(피치, 요 및 롤)를 포함하는 벡터를 기반으로 회전 쿼터니언을 계산합니다.
XMQuaternionSlerp 구형 선형 보간을 사용하여 두 단위 쿼터니언 사이를 보간합니다. (XMQuaternionSlerp)
XMQuaternionSlerpV

구형 선형 보간을 사용하여 두 단위 쿼터니언 사이를 보간합니다. (XMQuaternionSlerpV)
XMQuaternionSquad 구형 사분면 보간을 사용하여 4개의 단위 쿼터니언 사이를 보간합니다. (XMQuaternionSquad)
XMQuaternionSquadSetup 구형 사분면 보간을 위한 설정 제어 지점의 주소를 제공합니다.
XMQuaternionSquadV 구형 사분면 보간을 사용하여 4개의 단위 쿼터니언 사이를 보간합니다. (XMQuaternionSquadV)
XMQuaternionToAxisAngle 지정된 쿼터니언에 대해 해당 축에 대한 축 및 회전 각도를 계산합니다.
XMScalarACos 부동 소수점 숫자의 아크코사인을 계산합니다.
XMScalarACosEst 부동 소수점 숫자의 아크코사인을 추정합니다.
XMScalarASin 부동 소수점 숫자의 아크사인을 계산합니다.
XMScalarASinEst 부동 소수점 숫자의 아크사인을 추정합니다.
XMScalarCos 라디안 각도의 코사인을 계산합니다.
XMScalarCosEst 라디안 각도의 코사인을 추정합니다.
XMScalarModAngle -XM_PI 및 XM_PI 사이의 각도를 계산합니다.
XMScalarNearEqual 두 부동 소수점 값이 거의 같은지 확인합니다.

XMScalarSin 라디안 각도의 사인을 계산합니다.
XMScalarSinCos 라디안 각도의 사인과 코사인을 모두 계산합니다.
XMScalarSinCosEst 라디안 각도의 사인과 코사인을 모두 추정합니다.
XMScalarSinEst 라디안 각도의 사인을 추정합니다.
XMStoreFloat XMVECTOR를 float에 저장합니다.
XMStoreFloat2 XMFLOAT2에 XMVECTOR를 저장합니다.
XMStoreFloat2A XMFLOAT2A에 XMVECTOR를 저장합니다.
XMStoreFloat3 XMFLOAT3에 XMVECTOR를 저장합니다.
XMStoreFloat3A XMFLOAT3A에 XMVECTOR를 저장합니다.
XMStoreFloat3x3 XMFLOAT3X3에 XMMATRIX를 저장합니다.
XMStoreFloat3x4 **XMFLOAT3X4**에 **XMMATRIX**를 저장합니다.
XMStoreFloat3x4A **XMFLOAT3X4A에 XMMATRIX**를 저장합니다.
XMStoreFloat4

XMFLOAT4에 XMVECTOR를 저장합니다.
XMStoreFloat4A XMFLOAT4A에 XMVECTOR를 저장합니다.
XMStoreFloat4x3 XMFLOAT4X3에 XMMATRIX를 저장합니다.
XMStoreFloat4x3A XMFLOAT4X3A에 XMVECTOR를 저장합니다.
XMStoreFloat4x4 XMFLOAT4X4에 XMMATRIX를 저장합니다.
XMStoreFloat4x4A XMFLOAT4X4A에 XMVECTOR를 저장합니다.
XMStoreInt XMVECTOR를 uint32_t 저장합니다.
XMStoreInt2 XMVECTOR를 2개 요소 uint32_t 배열에 저장합니다.
XMStoreInt2A XMVECTOR를 16 바이트 정렬된 2개 요소 uint32_t 배열에 저장합니다.
XMStoreInt3 XMVECTOR를 3개 요소 uint32_t 배열에 저장합니다.
XMStoreInt3A XMVECTOR를 16 바이트 정렬된 3개 요소 uint32_t 배열에 저장합니다.
XMStoreInt4 XMVECTOR를 4개 요소 uint32_t 배열에 저장합니다.
XMStoreInt4A XMVECTOR를 16 바이트 정렬된 4개 요소 uint32_t 배열에 저장합니다.

XMStoreSInt2 XMVECTOR의 부록 정수 데이터를 XMINT2 구조체에 저장합니다.
XMStoreSInt3 XMVECTOR의 부록 정수 데이터를 XMINT3 구조체에 저장합니다.
XMStoreSInt4 XMVECTOR의 부록된 정수 데이터를 XMINT4 구조체에 저장합니다.
XMStoreUInt2 XMVECTOR의 부호 없는 정수 데이터를 XMUINT2 구조체에 저장합니다.
XMStoreUInt3 XMVECTOR의 부호 없는 정수 데이터를 XMUINT3 구조체에 저장합니다.
XMStoreUInt4 XMVECTOR의 부호 없는 정수 데이터를 XMUINT4 구조체에 저장합니다.
XMVector2AngleBetweenNormals 정규화된 두 2D 벡터 사이의 라디안 각도를 계산합니다.
XMVector2AngleBetweenNormalsEst 정규화된 두 2D 벡터 사이의 라디안 각도를 예측합니다.
XMVector2AngleBetweenVectors 두 2D 벡터 사이의 라디안 각도를 계산합니다.
XMVector2ClampLength 2D 벡터의 길이를 지정된 범위로 고정합니다. (XMVector2ClampLength)
XMVector2ClampLengthV 2D 벡터의 길이를 지정된 범위로 고정합니다. (XMVector2ClampLengthV)
XMVector2Cross 2D 교차 제품을 계산합니다.
XMVector2Dot

2D 벡터 간에 점 제품을 계산합니다.
XMVector2Equal 두 2D 벡터가 같은지 테스트합니다.
XMVector2EqualInt 두 2D 벡터가 같은지 테스트하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVector2EqualIntR 두 2D 벡터가 같은지 테스트하여 각 구성 요소를 부호 없는 정수로 처리합니다. 또한 이 함수는 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector2EqualR 두 2D 벡터가 같은지 테스트합니다. 또한 이 함수는 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector2Greater 하나의 2D 벡터가 다른 2D 벡터보다 큰지 테스트합니다.
XMVector2GreaterOrEqual 한 2D 벡터가 다른 2D 벡터보다 크거나 같은지 테스트합니다.
XMVector2GreaterOrEqualR 한 2D 벡터가 다른 2D 벡터보다 크거나 같은지 테스트하고 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector2GreaterR 한 2D 벡터가 다른 2D 벡터보다 큰지 테스트하고 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector2InBounds 2D 벡터의 구성 요소가 설정된 범위 내에 있는지 여부를 테스트합니다.
XMVector2IntersectLine 두 줄의 교집합을 찾습니다.
XMVector2IsInfinite 2D 벡터의 구성 요소가 양수인지 음의 무한대인지 테스트합니다.

XMVector2IsNaN 2D 벡터의 구성 요소가 NaN인지 테스트합니다.
XMVector2Length 2D 벡터의 길이를 계산합니다.
XMVector2LengthEst 2D 벡터의 길이를 예측합니다.
XMVector2LengthSq 2D 벡터 길이의 제곱을 계산합니다.
XMVector2Less 하나의 2D 벡터가 다른 2D 벡터보다 작은지 테스트합니다.
XMVector2LessOrEqual 한 2D 벡터가 다른 2D 벡터보다 작거나 같은지 테스트합니다.
XMVector2LinePointDistance 선과 점 사이의 최소 거리를 계산합니다. (XMVector2LinePointDistance)
XMVector2NearEqual 하나의 2D 벡터가 다른 2D 벡터 근처에 있는지 테스트합니다.
XMVector2Normalize 정규화된 2D 벡터 버전을 반환합니다.
XMVector2NormalizeEst 2D 벡터의 정규화된 버전을 예측합니다.
XMVector2NotEqual 두 2D 벡터가 같지 않은지 테스트합니다.
XMVector2NotEqualInt 두 벡터가 같지 않은지 테스트하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVector2Orthogonal

2D 벡터에 수직으로 벡터를 계산합니다.
XMVector2ReciprocalLength 2D 벡터 길이의 상호 계산
XMVector2ReciprocalLengthEst 2D 벡터 길이의 상호를 예측합니다.
XMVector2Reflect 2D 일반 벡터에서 인시던트 2D 벡터를 반영합니다.
XMVector2Refract 2D 일반 벡터에서 인시던트 2D 벡터를 굴절합니다. (XMVector2Refract)
XMVector2RefractV 2D 일반 벡터에서 인시던트 2D 벡터를 굴절합니다. (XMVector2RefractV)
XMVector2Transform 행렬로 2D 벡터를 변환합니다.
XMVector2TransformCoord 지정된 행렬에 의해 2D 벡터를 변환하여 결과를 w = 1로 다시 프로젝션합니다.
XMVector2TransformCoordStream 지정된 행렬에 따라 2D 벡터의 스트림을 변환하여 w 좌표가 1.0과 같도록 결과 벡터를 프로젝션합니다.
XMVector2TransformNormal 지정된 행렬에 의해 2D 벡터 정규를 변환합니다.
XMVector2TransformNormalStream 지정된 행렬을 통해 2D 일반 벡터의 스트림을 변환합니다.
XMVector2TransformStream 지정된 행렬을 통해 2D 벡터의 스트림을 변환합니다.
XMVector3AngleBetweenNormals 정규화된 두 3D 벡터 사이의 방사형 각도를 계산합니다.

XMVector3AngleBetweenNormalsEst 정규화된 두 3D 벡터 사이의 방사형 각도를 예측합니다.
XMVector3AngleBetweenVectors 두 3D 벡터 사이의 라디안 각도를 계산합니다.
XMVector3ClampLength 3D 벡터의 길이를 지정된 범위로 고정합니다. (XMVector3ClampLength)
XMVector3ClampLengthV 3D 벡터의 길이를 지정된 범위로 고정합니다. (XMVector3ClampLengthV)
XMVector3ComponentsFromNormal 참조 일반 벡터를 사용하여 3D 벡터를 일반과 병렬 및 수직인 구성 요소로 분할합니다.
XMVector3Cross 두 3D 벡터 간의 교차곱을 계산합니다.
XMVector3Dot 3D 벡터 간의 점 제품을 계산합니다.
XMVector3Equal 두 3D 벡터가 같은지 테스트합니다.
XMVector3EqualInt 두 개의 3D 벡터가 같은지 테스트하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVector3EqualIntR 두 개의 3D 벡터가 같은지 테스트하여 각 구성 요소를 부호 없는 정수로 처리합니다. 또한 이 함수는 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector3EqualR 두 3D 벡터가 같은지 테스트합니다. 또한 이 함수는 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector3Greater 한 3D 벡터가 다른 3D 벡터보다 큰지 테스트합니다.

XMVector3GreaterOrEqual 한 3D 벡터가 다른 3D 벡터보다 크거나 같은지 테스트합니다.
XMVector3GreaterOrEqualR 한 3D 벡터가 다른 3D 벡터보다 크거나 같은지 테스트하고 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector3GreaterR 한 3D 벡터가 다른 3D 벡터보다 큰지 테스트하고 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector3InBounds 3D 벡터의 구성 요소가 설정된 범위 내에 있는지 여부를 테스트합니다.
XMVector3InverseRotate 쿼터니언의 역방향을 사용하여 3D 벡터를 회전합니다.
XMVector3IsInfinite 3D 벡터의 구성 요소가 양수 또는 음수 무한대인지 테스트합니다.
XMVector3IsNaN 3D 벡터의 구성 요소가 NaN인지 여부를 테스트합니다.
XMVector3Length 3D 벡터의 길이를 계산합니다.
XMVector3LengthEst 3D 벡터의 길이를 예측합니다.
XMVector3LengthSq 3D 벡터 길이의 제곱을 계산합니다.
XMVector3Less 한 3D 벡터가 다른 3D 벡터보다 작은지 테스트합니다.
XMVector3LessOrEqual 한 3D 벡터가 다른 3D 벡터보다 작거나 같은지 테스트합니다.

XMVector3LinePointDistance 선과 점 사이의 최소 거리를 계산합니다. (XMVector3LinePointDistance)
XMVector3NearEqual 한 3D 벡터가 다른 3D 벡터 근처에 있는지 테스트합니다.
XMVector3Normalize 3D 벡터의 정규화된 버전을 반환합니다.
XMVector3NormalizeEst 3D 벡터의 정규화된 버전을 예측합니다.
XMVector3NotEqual 두 3D 벡터가 같지 않은지 테스트합니다.
XMVector3NotEqualInt 두 개의 3D 벡터가 같지 않은지 테스트하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVector3Orthogonal 3D 벡터에 수직으로 벡터를 계산합니다.
XMVector3Project 개체 공간에서 화면 공간으로 3D 벡터를 프로젝터합니다.
XMVector3ProjectStream 개체 공간에서 화면 공간으로 3D 벡터 스트림을 프로젝터합니다.
XMVector3ReciprocalLength 3D 벡터 길이의 상호 계산
XMVector3ReciprocalLengthEst 3D 벡터 길이의 상호를 추정합니다.
XMVector3Reflect 3D 일반 벡터에서 인시던트 3D 벡터를 반영합니다.
XMVector3Refract

3D 일반 벡터에서 인시던트 3D 벡터를 굴절합니다. (XMVector3Refract)
XMVector3RefractV 3D 일반 벡터에서 인시던트 3D 벡터를 굴절합니다. (XMVector3RefractV)
XMVector3Rotate 쿼터니언을 사용하여 3D 벡터를 회전합니다.
XMVector3Transform 행렬별로 3D 벡터를 변환합니다.
XMVector3TransformCoord 지정된 행렬에 의해 3D 벡터를 변환하여 결과를 w = 1로 다시 프로젝션합니다.
XMVector3TransformCoordStream 지정된 행렬에 따라 3D 벡터 스트림을 변환하여 w 좌표가 1.0과 같도록 결과 벡터를 프로젝션합니다.
XMVector3TransformNormal 지정된 행렬에 의해 3D 벡터 정규를 변환합니다.
XMVector3TransformNormalStream 지정된 행렬을 통해 3D 일반 벡터의 스트림을 변환합니다.
XMVector3TransformStream 지정된 행렬을 통해 3D 벡터 스트림을 변환합니다.
XMVector3Unproject 화면 공간에서 개체 공간으로 3D 벡터를 프로젝터합니다.
XMVector3UnprojectStream 3D 벡터 스트림을 화면 공간에서 개체 공간으로 변환합니다.
XMVector4AngleBetweenNormals 정규화된 두 4D 벡터 사이의 방사형 각도를 계산합니다.
XMVector4AngleBetweenNormalsEst 정규화된 두 4D 벡터 사이의 라디안 각도를 예측합니다.

XMVector4AngleBetweenVectors 두 4D 벡터 사이의 방사형 각도를 계산합니다.
XMVector4ClampLength 4D 벡터의 길이를 지정된 범위로 고정합니다. (XMVector4ClampLength)
XMVector4ClampLengthV 4D 벡터의 길이를 지정된 범위로 고정합니다. (XMVector4ClampLengthV)
XMVector4Cross 4D 교차 제품을 계산합니다.
XMVector4Dot 4D 벡터 간에 점 제품을 계산합니다.
XMVector4Equal 두 4D 벡터가 같은지 테스트합니다.
XMVector4EqualInt 두 개의 4D 벡터가 같은지 테스트하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVector4EqualIntR 두 개의 4D 벡터가 같은지 테스트하여 각 구성 요소를 부호 없는 정수로 처리합니다. 또한 이 함수는 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector4EqualR 두 4D 벡터가 같은지 테스트합니다. 또한 이 함수는 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector4Greater 하나의 4D 벡터가 다른 4D 벡터보다 큰지 테스트합니다.
XMVector4GreaterOrEqual 한 4D 벡터가 다른 4D 벡터보다 크거나 같은지 테스트합니다.
XMVector4GreaterOrEqualR

한 4D 벡터가 다른 4D 벡터보다 크거나 같은지 테스트하고 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector4GreaterR 한 4D 벡터가 다른 4D 벡터보다 큰지 테스트하고 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 반환합니다.
XMVector4InBounds 4D 벡터의 구성 요소가 설정된 범위 내에 있는지 테스트합니다.
XMVector4IsInfinite 4D 벡터의 구성 요소가 양수인지 음의 무한대인지 테스트합니다.
XMVector4IsNaN 4D 벡터의 구성 요소가 NaN인지 테스트합니다.
XMVector4Length 4D 벡터의 길이를 계산합니다.
XMVector4LengthEst 4D 벡터의 길이를 예측합니다.
XMVector4LengthSq 4D 벡터 길이의 제곱을 계산합니다.
XMVector4Less 하나의 4D 벡터가 다른 4D 벡터보다 작은지 테스트합니다.
XMVector4LessOrEqual 한 4D 벡터가 다른 4D 벡터보다 작거나 같은지 테스트합니다.
XMVector4NearEqual 하나의 4D 벡터가 다른 4D 벡터 근처에 있는지 테스트합니다.
XMVector4Normalize 정규화된 4D 벡터 버전을 반환합니다.
XMVector4NormalizeEst

4D 벡터의 정규화된 버전을 예측합니다.
XMVector4NotEqual 두 4D 벡터가 같지 않은지 테스트합니다.
XMVector4NotEqualInt 두 개의 4D 벡터가 같지 않은지 테스트하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVector4Orthogonal 4D 벡터에 수직인 벡터를 계산합니다.
XMVector4ReciprocalLength 4D 벡터 길이의 상호를 계산합니다.
XMVector4ReciprocalLengthEst 4D 벡터 길이의 상호를 추정합니다.
XMVector4Reflect 4D 일반 벡터에서 인시던트 4D 벡터를 반영합니다.
XMVector4Refract 4D 일반 벡터에서 인시던트 4D 벡터를 굴절합니다. (XMVector4Refract)
XMVector4RefractV 4D 일반 벡터에서 인시던트 4D 벡터를 굴절합니다. (XMVector4RefractV)
XMVector4Transform 행렬별로 4D 벡터를 변환합니다.
XMVector4TransformStream 지정된 행렬을 통해 4D 벡터 스트림을 변환합니다.
XMVectorAbs XMVECTOR의 각 구성 요소에 대한 절대값을 계산합니다.
XMVectorACos XMVECTOR의 각 구성 요소의 아크코사인을 계산합니다.

XMVectorACosEst XMVECTOR의 각 구성 요소의 아크코사인을 추정합니다.
XMVectorAdd 두 벡터의 합계를 계산합니다.
XMVectorAddAngles 각도를 나타내는 두 개의 벡터를 추가합니다.
XMVectorAndCInt 두 번째 벡터의 부정을 사용하여 한 벡터의 논리적 AND를 계산하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVectorAndInt 두 벡터의 논리적 AND를 계산하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVectorASin XMVECTOR의 각 구성 요소의 아크사인을 계산합니다.
XMVectorASinEst XMVECTOR의 각 구성 요소의 아크사인을 추정합니다.
XMVectorATan XMVECTOR의 각 구성 요소의 아크탄젠트를 계산합니다.
XMVectorATan2 Y/X의 아크탄젠트를 계산합니다.
XMVectorATan2Est Y/X의 아크탄젠트를 추정합니다.
XMVectorATanEst XMVECTOR의 각 구성 요소의 아크탄젠트를 예측합니다.
XMVectorBaryCentric 지정된 위치 벡터를 사용하여 Barycentric 좌표의 점을 반환합니다. (XMVectorBaryCentric)

XMVectorBaryCentricV 지정된 위치 벡터를 사용하여 Barycentric 좌표의 점을 반환합니다. (XMVectorBaryCentricV)
XMVectorCatmullRom 지정된 위치 벡터를 사용하여 Catmull-Rom 보간을 수행합니다. (XMVectorCatmullRom)
XMVectorCatmullRomV 지정된 위치 벡터를 사용하여 Catmull-Rom 보간을 수행합니다. (XMVectorCatmullRomV)
XMVectorCeiling XMVECTOR의 각 구성 요소의 최대값을 계산합니다.
XMVectorClamp 벡터의 구성 요소를 지정된 최소 및 최대 범위로 고정합니다.
XMVectorCos XMVECTOR의 각 구성 요소의 코사인을 계산합니다.
XMVectorCosEst XMVECTOR의 각 구성 요소의 코사인을 추정합니다.
XMVectorCosH XMVECTOR의 각 구성 요소에 대한 쌍곡선 코사인을 계산합니다.
XMVectorDivide XMVECTOR의 한 instance 두 번째 instance 나누어 결과를 세 번째 instance 반환합니다. (XMVectorDivide)
XMVectorEqual 두 벡터의 같음을 위해 구성 요소별 테스트를 수행합니다.
XMVectorEqualInt 두 벡터의 같음을 위해 구성 요소별 테스트를 수행하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVectorEqualIntR 두 벡터의 같음을 위해 구성 요소별 테스트를 수행하여 각 구성 요소를 부호 없는 정수로 처리합니

다. 또한 이 함수는 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 설정합니다.
XMVectorEqualR 두 벡터의 같음에 대한 구성 요소별 테스트를 수행하고 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 설정합니다.
XMVectorExp 각 구성 요소의 전원에 대해 발생된 두 개를 계산합니다. 참고 이 함수는 기존 Windows 8 코드의 XMVectorExp2에 대한 호환성 별칭입니다. 이 함수는 Windows 8.1 더 이상 사용되지 않습니다. 사용하지 말고 대신 XMVectorExp2 또는 XMVectorExpE를 사용합니다. .
XMVectorExp10 각 구성 요소의 전원에 대해 발생된 10개를 계산합니다.
XMVectorExp2 각 구성 요소의 전원에 대해 발생된 두 개를 계산합니다.
XMVectorExpE 각 구성 요소의 전원으로 발생된 e(~2.71828)를 계산합니다.
XMVectorFalseInt 0(false) 벡터를 반환합니다.
XMVectorFloor XMVECTOR의 각 구성 요소의 바닥을 계산합니다.
XMVectorGetByIndex 인덱스별 부동 소수점 데이터를 포함하는 XMVECTOR 데이터 형식의 네 가지 구성 요소 중 하나의 값을 검색합니다.
XMVectorGetByIndexPtr 포인터에서 참조하는 부동 소수점의 instance 인덱스에서 참조하는 부동 소수점 데이터를 포함하는 XMVECTOR 데이터 형식의 4개 구성 요소 중 하나의 값을 검색합니다.
XMVectorGetIntByIndex 인덱스별 정수 데이터를 포함하는 XMVECTOR 데이터 형식의 네 가지 구성 요소 중 하나의 값을 검색합니다.

XMVectorGetIntByIndexPtr 포인터에서 참조하는 정수의 instance 인덱스별 정수 데이터를 포함하는 XMVECTOR 데이터 형식의 4개 구성 요소 중 하나의 값을 검색합니다.
XMVectorGetIntW XMVECTOR 데이터 형식의 w 구성 요소를 검색합니다. (XMVectorGetIntW)
XMVectorGetIntWPtr 정수 데이터가 포함된 XMVECTOR 데이터 형식의 w 구성 요소를 검색하고 포인터에서 참조하는 uint32_t instance 해당 구성 요소의 값을 저장합니다.
XMVectorGetIntX XMVECTOR 데이터 형식의 x 구성 요소를 검색합니다. (XMVectorGetIntX)
XMVectorGetIntXPtr 정수 데이터가 포함된 XMVECTOR 데이터 형식의 x 구성 요소를 검색하고 포인터에서 참조하는 uint32_t instance 해당 구성 요소의 값을 저장합니다.
XMVectorGetIntY XMVECTOR 데이터 형식의 y 구성 요소를 검색합니다. (XMVectorGetIntY)
XMVectorGetIntYPtr 정수 데이터가 포함된 XMVECTOR 데이터 형식의 y 구성 요소를 검색하고 포인터에서 참조하는 uint32_t instance 해당 구성 요소의 값을 저장합니다.
XMVectorGetIntZ XMVECTOR 데이터 형식의 z 구성 요소를 검색합니다. (XMVectorGetIntZ)
XMVectorGetIntZPtr 정수 데이터가 포함된 XMVECTOR 데이터 형식의 z 구성 요소를 검색하고 포인터에서 참조하는 uint32_t instance 해당 구성 요소의 값을 저장합니다.
XMVectorGetW XMVECTOR 데이터 형식의 w 구성 요소를 검색합니다. (XMVectorGetW)
XMVectorGetWPtr 부동 소수점 데이터가 포함된 XMVECTOR 데이터 형식의 w 구성 요소를 검색하고 포인터가 참조하는 부동 소수점 instance 해당 구성 요소의 값을 저장합니다.

XMVectorGetX XMVECTOR 데이터 형식의 x 구성 요소를 검색합니다. (XMVectorGetX)
XMVectorGetXPtr 부동 소수점 데이터를 포함하는 XMVECTOR 데이터 형식의 x 구성 요소를 검색하고 포인터에서 참조하는 부동 소수점 instance 해당 구성 요소의 값을 저장합니다.
XMVectorGetY XMVECTOR 데이터 형식의 y 구성 요소를 검색합니다. (XMVectorGetY)
XMVectorGetYPtr 부동 소수점 데이터를 포함하는 XMVECTOR 데이터 형식의 y 구성 요소를 검색하고 포인터에서 참조하는 부동 소수점 instance 해당 구성 요소의 값을 저장합니다.
XMVectorGetZ XMVECTOR 데이터 형식의 z 구성 요소를 검색합니다. (XMVectorGetZ)
XMVectorGetZPtr 부동 소수점 데이터를 포함하는 XMVECTOR 데이터 형식의 z 구성 요소를 검색하고 포인터에서 참조하는 부동 소수점 instance 해당 구성 요소의 값을 저장합니다.
XMVectorGreater 두 벡터 사이의 보다 큰 구성 요소에 대한 구성 요소별 테스트를 수행합니다.
XMVectorGreaterOrEqual 두 벡터 간에 크거나 같은 구성 요소별 테스트를 수행합니다.
XMVectorGreaterOrEqualR 두 벡터 간에 크거나 같은 구성 요소별 테스트를 수행하고 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 설정합니다.
XMVectorGreaterR 두 벡터 사이의 보다 큰 구성 요소별 테스트를 수행하고 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 설정합니다.
XMVectorHermite 지정된 벡터를 사용하여 Hermite 스플라인 보간을 수행합니다. (XMVectorHermite)

XMVectorHermiteV 지정된 벡터를 사용하여 Hermite 스플라인 보간을 수행합니다. (XMVectorHermiteV)
XMVectorInBounds 지정된 벡터의 구성 요소가 설정된 범위 내에 있는지 여부를 테스트합니다.
XMVectorInBoundsR 지정된 벡터의 구성 요소가 특정 범위 내에 있는지 테스트하고 XMComparisonAllTrue와 같은 함수를 사용하여 검사할 수 있는 비교 값을 설정합니다.
XMVectorInsert 지정된 수의 32비트 구성 요소에서 왼쪽으로 벡터를 회전하고 해당 결과의 선택한 요소를 다른 벡터에 삽입합니다.
XMVectorIsInfinite 벡터에서 +/- 무한대에 대한 구성 요소별 테스트를 수행합니다.
XMVectorIsNaN 벡터에서 구성 요소별 NaN 테스트를 수행합니다.
XMVectorLerp 두 벡터 간의 선형 보간을 수행합니다. (XMVectorLerp)
XMVectorLerpV 두 벡터 간의 선형 보간을 수행합니다. (XMVectorLerpV)
XMVectorLess 두 벡터 사이의 미만에 대해 구성 요소별 테스트를 수행합니다.
XMVectorLessOrEqual 두 벡터 간의 작거나 같음인 구성 요소별 테스트를 수행합니다.
XMVectorLog 벡터의 각 구성 요소에 대한 기본 두 로그를 계산합니다. 참고 이 함수는 기존 Windows 8 코드의 XMVectorLog2에 대한 호환성 별칭입니다.
XMVectorLog10

벡터의 각 구성 요소에 대한 기본 10개의 로그를 계산합니다.
XMVectorLog2 벡터의 각 구성 요소에 대한 기본 두 로그를 계산합니다.
XMVectorLoge 벡터의 각 구성 요소에 대한 기본 e 로그를 계산합니다.
XMVectorMax 두 벡터를 구성 요소별로 비교하고 가장 큰 구성 요소를 포함하는 벡터를 반환합니다.
XMVectorMergeXY 두 벡터의 x 및 y 구성 요소를 결합하여 새 벡터를 만듭니다.
XMVectorMergeZW 두 벡터의 z 및 w 구성 요소를 결합하여 새 벡터를 만듭니다.
XMVectorMin 두 벡터를 구성 요소별로 비교하고 가장 작은 구성 요소를 포함하는 벡터를 반환합니다.
XMVectorMod 두 벡터의 몫의 구성 요소별 부동 소수점 나머지를 계산합니다.
XMVectorModAngles 구성 요소별 각도 모듈로 2PI를 계산합니다.
XMVectorMultiply 두 벡터의 구성 요소별 제품을 계산합니다.
XMVectorMultiplyAdd 세 번째 벡터에 추가된 처음 두 벡터의 곱을 계산합니다.
XMVectorNearEqual 지정된 임계값 내에서 두 벡터의 같음을 위해 구성 요소별 테스트를 수행합니다.
XMVectorNegate 벡터의 부정을 계산합니다.

XMVectorNegativeMultiplySubtract 세 번째 벡터와 처음 두 벡터의 곱의 차이를 계산합니다.
XMVectorNorInt 두 벡터의 논리적 NOR을 계산하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVectorNotEqual 두 벡터의 같지 않음에 대한 구성 요소별 테스트를 수행합니다.
XMVectorNotEqualInt 두 벡터의 같지 않음에 대한 구성 요소별 테스트를 수행하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVectorOrInt 두 벡터의 논리적 OR을 계산하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVectorPermute 두 벡터의 구성 요소를 음소거하여 새 벡터를 만듭니다.
XMVectorPow V2의 성능에 맞게 V1을 계산합니다.
XMVectorReciprocal 벡터의 구성 요소별 상호를 계산합니다.
XMVectorReciprocalEst 벡터의 구성 요소별 상호를 예측합니다.
XMVectorReciprocalSqrt 벡터의 구성 요소별 상호 제곱근을 계산합니다.
XMVectorReciprocalSqrtEst 벡터의 구성 요소별 상호 제곱근을 예측합니다.
XMVectorReplicate 부동 소수점 값을 벡터의 네 구성 요소 모두에 복제합니다.

XMVectorReplicateInt 정수 값을 벡터의 네 가지 구성 요소 모두에 복제합니다.
XMVectorReplicateIntPtr 포인터에서 참조하는 정수 값을 벡터의 네 구성 요소 모두에 복제합니다.
XMVectorReplicatePtr 포인터에서 참조하는 부동 소수점 값을 벡터의 네 구성 요소 모두에 복제합니다.
XMVectorRotateLeft 지정된 수의 32비트 요소로 왼쪽으로 벡터를 회전합니다.
XMVectorRotateRight 지정된 수의 32비트 요소로 벡터를 오른쪽으로 회전합니다.
XMVectorRound 벡터의 각 구성 요소를 가장 가까운 정수("은행 반올림"라고 함)로 반올림합니다.
XMVectorSaturate 벡터의 각 구성 요소를 0.0f에서 1.0f 범위로 포화합니다.
XMVectorScale 스칼라는 벡터에 부동 소수점 값을 곱합니다.
XMVectorSelect 두 입력 벡터 간에 구성 요소별 선택을 수행하고 결과 벡터를 반환합니다.
XMVectorSelectControl XMVectorSelect에서 사용할 컨트롤 벡터를 정의합니다.
XMVectorSet 4개의 부동 소수점 값을 사용하여 벡터를 만듭니다.
XMVectorSetBinaryConstant 각 구성 요소가 0.0f 또는 1.0f인 벡터를 만듭니다.
XMVectorSetByIndex

부동 소수점 개체를 사용하여 인덱스에서 참조하는 정수 데이터를 포함하는 XMVECTOR 데이터 형식의 네 가지 구성 요소 중 하나의 값을 설정합니다.
XMVectorSetByIndexPtr 부동 소수점 instance 대한 포인터를 사용하여 인덱스에서 참조하는 부동 소수점 데이터를 포함하는 XMVECTOR 데이터 형식의 네 가지 구성 요소 중 하나의 값을 설정합니다.
XMVectorSetInt 부호 없는 정수 구성 요소가 있는 벡터를 만듭니다.
XMVectorSetIntByIndex 정수 instance 사용하여 인덱스에서 참조하는 정수 데이터를 포함하는 XMVECTOR 데이터 형식의 네 가지 구성 요소 중 하나의 값을 설정합니다.
XMVectorSetIntByIndexPtr 정수 instance 대한 포인터를 사용하여 인덱스에서 참조하는 정수 데이터를 포함하는 XMVECTOR 데이터 형식의 네 가지 구성 요소 중 하나의 값을 설정합니다.
XMVectorSetIntW XMVECTOR 데이터 형식의 w 구성 요소 값을 설정합니다. (XMVectorSetIntW)
XMVectorSetIntWPtr 포인터에서 참조하는 uint32_t instance 포함된 값을 사용하여 정수 데이터를 포함하는 XMVECTOR의 w 구성 요소를 설정합니다.
XMVectorSetIntX XMVECTOR 데이터 형식의 x 구성 요소 값을 설정합니다. (XMVectorSetIntX)
XMVectorSetIntXPtr 포인터에서 참조하는 uint32_t instance 포함된 값을 사용하여 정수 데이터를 포함하는 XMVECTOR의 x 구성 요소를 설정합니다.
XMVectorSetIntY XMVECTOR 데이터 형식의 y 구성 요소 값을 설정합니다. (XMVectorSetIntY)
XMVectorSetIntYPtr 포인터에서 참조하는 uint32_t instance 포함된 값을 사용하여 정수 데이터를 포함하는 XMVECTOR의 y 구성 요소를 설정합니다.

XMVectorSetIntZ XMVECTOR 데이터 형식의 z 구성 요소 값을 설정합니다. (XMVectorSetIntZ)
XMVectorSetIntZPtr 포인터에서 참조하는 uint32_t instance 포함된 값을 사용하여 정수 데이터를 포함하는 XMVECTOR의 z 구성 요소를 설정합니다.
XMVectorSetW XMVECTOR 데이터 형식의 w 구성 요소 값을 설정합니다. (XMVectorSetW)
XMVectorSetWPtr 부동 소수점 데이터를 포함하는 XMVECTOR의 w 구성 요소를 포인터가 참조하는 부동 소수점 instance 포함된 값으로 설정합니다.
XMVectorSetX XMVECTOR 데이터 형식의 x 구성 요소 값을 설정합니다. (XMVectorSetX)
XMVectorSetXPtr 부동 소수점 데이터를 포함하는 XMVECTOR의 x 구성 요소를 포인터에서 참조하는 부동 소수점 instance 포함된 값으로 설정합니다.
XMVectorSetY XMVECTOR 데이터 형식의 y 구성 요소 값을 설정합니다. (XMVectorSetY)
XMVectorSetYPtr 부동 소수점 데이터를 포함하는 XMVECTOR의 y 구성 요소를 포인터가 참조하는 부동 소수점 instance 포함된 값으로 설정합니다.
XMVectorSetZ XMVECTOR 데이터 형식의 z 구성 요소 값을 설정합니다. (XMVectorSetZ)
XMVectorSetZPtr 부동 소수점 데이터를 포함하는 XMVECTOR의 z 구성 요소를 포인터에서 참조하는 부동 소수점 instance 포함된 값으로 설정합니다.
XMVectorShiftLeft 지정된 수의 32비트 요소에서 왼쪽으로 벡터를 이동하여 비어 있는 요소를 두 번째 벡터의 요소로 채웁니다.

XMVectorSin XMVECTOR의 각 구성 요소의 사인을 계산합니다.
XMVectorSinCos XMVECTOR의 각 구성 요소에 대한 사인과 코사인을 계산합니다.
XMVectorSinCosEst XMVECTOR의 각 구성 요소에 대한 사인 및 코사인을 추정합니다.
XMVectorSinEst XMVECTOR의 각 구성 요소의 사인을 추정합니다.
XMVectorSinH XMVECTOR의 각 구성 요소에 대한 쌍곡선 사인을 계산합니다.
XMVectorSplatConstant 동일한 부동 소수점 구성 요소를 사용하여 벡터를 만듭니다. 각 구성 요소는 정수 지수로 2로 나눈 상수입니다.
XMVectorSplatConstantInt 동일한 정수 구성 요소를 사용하여 벡터를 만듭니다.
XMVectorSplatEpsilon 각 구성 요소가 epsilon인 벡터를 반환합니다(1.192092896e-7).
XMVectorSplatInfinity 각 구성 요소가 무한대(0x7F800000)인 벡터를 반환합니다.
XMVectorSplatOne 각 구성 요소가 하나인 벡터를 반환합니다.
XMVectorSplatQNaN 각 구성 요소가 QNaN(0x7CF00000)인 벡터를 반환합니다.
XMVectorSplatSignMask 각 구성 요소가 기호 마스크(0x80000000)인 벡터를 반환합니다.

XMVectorSplatW 벡터의 w 구성 요소를 모든 구성 요소에 복제합니다.
XMVectorSplatX 벡터의 x 구성 요소를 모든 구성 요소에 복제합니다.
XMVectorSplatY 벡터의 y 구성 요소를 모든 구성 요소에 복제합니다.
XMVectorSplatZ 벡터의 z 구성 요소를 모든 구성 요소에 복제합니다.
XMVectorSqrt 벡터의 구성 요소별 제곱근을 계산합니다.
XMVectorSqrtEst 벡터의 구성 요소별 제곱근을 추정합니다.
XMVectorSubtract 두 벡터의 차이를 계산합니다.
XMVectorSubtractAngles 각도를 나타내는 두 개의 벡터를 뺍니다.
XMVectorSum XMVECTOR 구성 요소의 가로 합계를 계산합니다. 가로 합계는 벡터의 각 구성 요소를 함께 추가한 결과입니다.
XMVectorSwizzle 벡터를 스위즐합니다.
XMVectorTan XMVECTOR의 각 구성 요소의 탄젠트를 계산합니다.
XMVectorTanEst XMVECTOR의 각 구성 요소의 탄젠트를 추정합니다.

XMVectorTanH XMVECTOR의 각 구성 요소에 대한 쌍곡선 탄젠트를 계산합니다.
XMVectorTrueInt 각 구성 요소가 true(0xFFFFFFFF)를 나타내는 벡터를 반환합니다.
XMVectorTruncate 벡터의 각 구성 요소를 0 방향으로 가장 가까운 정수 값으로 반올림합니다.
XMVectorXorInt 두 벡터의 논리 XOR을 계산하여 각 구성 요소를 부호 없는 정수로 처리합니다.
XMVectorZero 0 벡터를 만듭니다.
XMVerifyCPUSupport DirectXMath 라이브러리가 현재 플랫폼을 지원하는지 나타냅니다.

구조체

XMFLOAT2 두 개의 단정밀도 부동 소수점 값으로 구성된 2D 벡터입니다.
XMFLOAT2A 16 바이트 경계에 맞춰진 XMFLOAT2 구조체에 대해 설명합니다.
XMFLOAT3 3개의 단정밀도 부동 소수점 값으로 구성된 3D 벡터에 대해 설명합니다.
XMFLOAT3A 16비트 경계에 맞춰진 XMFLOAT3 구조체에 대해 설명합니다.
XMFLOAT3X3 3x3 부동 소수점 행렬입니다.

XMFLOAT3X4 32비트 부동 소수점 구성 요소를 포함하는 3x4 열 주 행렬입니다.
XMFLOAT3X4A 16비트 경계에 맞춰진 32비트 부동 소수점 구성 요소를 포함하는 3x4 열 주 행렬입니다.
XMFLOAT4 4개의 단정밀도 부동 소수점 값으로 구성된 4D 벡터에 대해 설명합니다.
XMFLOAT4A 16 바이트 경계에 맞춰진 XMFLOAT4 구조체에 대해 설명합니다.
XMFLOAT4X3 4*3 부동 소수점 행렬입니다.
XMFLOAT4X3A 16비트 경계에 맞춰진 XMFLOAT4X3 구조체에 대해 설명합니다.
XMFLOAT4X4 4*4 부동 소수점 행렬입니다.
XMFLOAT4X4A 16 바이트 경계에 맞춰진 XMFLOAT4X4 구조체에 대해 설명합니다.
XMINT2 각 구성 요소가 부록 정수인 2D 벡터입니다.
XMINT3 각 구성 요소가 부록 정수인 3D 벡터입니다.
XMINT4 각 구성 요소가 부록 정수인 4D 벡터입니다.
XMMATRIX 4개의 하드웨어 벡터 레지스터에 매핑되는 16비트 경계에 맞춰진 4*4 행렬을 설명합니다.
XMUINT2

각 구성 요소가 부호 없는 정수인 2D 벡터입니다.
XMUINT3 각 구성 요소가 부호 없는 정수인 3D 벡터입니다.
XMUINT4 각 구성 요소가 부호 없는 정수인 4D 벡터입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

directxpackedvector.h 헤더

아티클2023. 08. 23.

이 헤더는 DirectXMath에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DirectXMath

directxpackedvector.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

XMConvertFloatToHalf 단정밀도 부동 소수점 값을 반정밀도 부동 소수점 값으로 변환합니다.
XMConvertFloatToHalfStream 단정밀도 부동 소수점 값의 스트림을 반정밀도 부동 소수점 값의 스트림으로 변환합니다.
XMConvertHalfToFloat 반정밀도 부동 소수점 값을 단정밀도 부동 소수점 값으로 변환합니다.
XMConvertHalfToFloatStream 반정밀도 부동 소수점 값의 스트림을 단정밀도 부동 소수점 값의 스트림으로 변환합니다.
XMLoadByte2 XMBYTE2를 XMVECTOR에 로드합니다.
XMLoadByte4 XMBYTE4를 XMVECTOR에 로드합니다.
XMLoadByteN2 XMBYTEN2를 XMVECTOR에 로드합니다.
XMLoadByteN4 XMVECTOR에 XMBYTEN4를 로드합니다.
XMLoadColor

XMCOLOR를 XMVECTOR에 로드합니다.
XMLoadDec4 XMDEC4를 XMVECTOR에 로드합니다.
XMLoadDecN4 XMDECN4를 XMVECTOR에 로드합니다.
XMLoadFloat3PK XMFLOAT3PK를 XMVECTOR에 로드합니다.
XMLoadFloat3SE XMFLOAT3SE를 XMVECTOR에 로드합니다.
XMLoadHalf2 XMHALF2를 XMVECTOR에 로드합니다.
XMLoadHalf4 XMHALF4를 XMVECTOR에 로드합니다.
XMLoadShort2 XMSHORT2를 XMVECTOR에 로드합니다.
XMLoadShort4 XMSHORT4를 XMVECTOR에 로드합니다.
XMLoadShortN2 XMSHORTN2를 XMVECTOR에 로드합니다.
XMLoadShortN4 XMSHORTN4를 XMVECTOR에 로드합니다.
XMLoadU555 XMU555를 XMVECTOR에 로드합니다.
XMLoadU565 XMU565를 XMVECTOR에 로드합니다.

XMLoadUByte2 XMVECTOR에 XMUBYTE2를 로드합니다.
XMLoadUByte4 XMVECTOR에 XMUBYTE4를 로드합니다.
XMLoadUByteN2 XMVECTOR에 XMUBYTEN2를 로드합니다.
XMLoadUByteN4 XMVECTOR에 XMUBYTEN4를 로드합니다.
XMLoadUDec4 XMUDEC4를 XMVECTOR에 로드합니다.
XMLoadUDecN4 XMUDECN4를 XMVECTOR에 로드합니다.
XMLoadUDecN4_XR 확장된 범위 XMUDECN4를 XMVECTOR에 로드합니다.
XMLoadUNibble4 XMVECTOR에 XMUNIBBLE4를 로드합니다.
XMLoadUShort2 XMUSHORT2를 XMVECTOR에 로드합니다.
XMLoadUShort4 XMUSHORT4를 XMVECTOR에 로드합니다.
XMLoadUShortN2 XMUSHORTN2를 XMVECTOR에 로드합니다.
XMLoadUShortN4 XMVECTOR에 XMUSHORTN4를 로드합니다.
XMLoadXDec4

XMXDEC4를 XMVECTOR에 로드합니다.
XMLoadXDecN4 XMXDECN4를 XMVECTOR에 로드합니다.
XMStoreByte2 XMBYTE2에 XMVECTOR를 저장합니다.
XMStoreByte4 XMBYTE4에 XMVECTOR를 저장합니다.
XMStoreByteN2 XMBYTEN2에 XMVECTOR를 저장합니다.
XMStoreByteN4 XMBYTEN4에 XMVECTOR를 저장합니다.
XMStoreColor XMCOLOR에 XMVECTOR를 저장합니다.
XMStoreDec4 XMDEC4에 XMVECTOR를 저장합니다.
XMStoreDecN4 XMDECN4에 XMVECTOR를 저장합니다.
XMStoreFloat3PK XMFLOAT3PK에 XMVECTOR를 저장합니다.
XMStoreFloat3SE XMFLOAT3SE에 XMVECTOR를 저장합니다.
XMStoreHalf2 XMHALF2에 XMVECTOR를 저장합니다.
XMStoreHalf4 XMHALF4에 XMVECTOR를 저장합니다.

XMStoreShort2 XMSHORT2에 XMVECTOR를 저장합니다.
XMStoreShort4 XMSHORT4에 XMVECTOR를 저장합니다.
XMStoreShortN2 XMSHORTN2에 XMVECTOR를 저장합니다.
XMStoreShortN4 XMSHORTN4에 XMVECTOR를 저장합니다.
XMStoreU555 XMU555에 XMVECTOR를 저장합니다.
XMStoreU565 XMU565에 XMVECTOR를 저장합니다.
XMStoreUByte2 XMVECTOR를 XMUBYTE2에 저장합니다.
XMStoreUByte4 XMUBYTE4에 XMVECTOR를 저장합니다.
XMStoreUByteN2 XMUBYTEN2에 XMVECTOR를 저장합니다.
XMStoreUByteN4 XMUBYTEN4에 XMVECTOR를 저장합니다.
XMStoreUDec4 XMUDEC4에 XMVECTOR를 저장합니다.
XMStoreUDecN4 XMUDECN4에 XMVECTOR를 저장합니다.
XMStoreUDecN4_XR

확장된 범위 XMUDECN4를 XMVECTOR에 저장합니다.
XMStoreUNibble4 XMVECTOR를 XMUNIBBLE4에 저장합니다.
XMStoreUShort2 XMUSHORT2에 XMVECTOR를 저장합니다.
XMStoreUShort4 XMUSHORT4에 XMVECTOR를 저장합니다.
XMStoreUShortN2 XMUSHORTN2에 XMVECTOR를 저장합니다.
XMStoreUShortN4 XMUSHORTN4에 XMVECTOR를 저장합니다.
XMStoreXDec4 XMXDEC4에 XMVECTOR를 저장합니다.
XMStoreXDecN4 XMXDECN4에 XMVECTOR를 저장합니다.

구조체

XMBYTE2 각 구성 요소가 부호 있는 정수인 8비트(1 바이트)인 2D 벡터입니다.
XMBYTE4 각 구성 요소가 부호 있는 정수인 8비트(1 바이트)인 4D 벡터입니다.
XMBYTEN2 부호 있는 정규화된 값을 부호 있는 8비트(1 바이트) 정수로 저장하기 위한 2D 벡터입니다.
XMBYTEN4

부호 있는 정규화된 값을 부호 있는 8비트(1 바이트) 정수로 저장하기 위한 3D 벡터입니다.
XMCOLOR 각 색 채널이 부호 없는 8비트 정수로 지정된 32비트 ARGB(알파 레드 그린 블루) 색 벡터입니다.
XMDEC4 x,y-및 z-구성 요소가 10비트 부호 있는 정수 값으로 표현되고 w-component가 2비트 부호 있는 정수 값으로 표시되는 4D 벡터입니다.
XMDECN4 부호 있는 정규화된 값을 10비트 부호 있는 x-,y-및 z- 구성 요소와 2비트 부호 있는 w-component로 저장하기 위한 4D 벡터입니다.
XMFLOAT3PK X 및 Y 구성 요소가 11비트 부동 소수점 숫자로 저장되고 Z 구성 요소가 10비트 부동 소수점 값으로 저장된 3D 벡터에 대해 설명합니다.
XMFLOAT3SE 각각 동일한 5비트 지수를 공유하는 9비트 매니타사가 있는 3개의 부동 소수점 구성 요소의 3D 벡터를 설명합니다.
XMHALF2 2개의 반정밀도(16비트) 부동 소수점 값으로 구성된 2D 벡터입니다.
XMHALF4 4개의 반정밀도(16비트) 부동 소수점 값으로 구성된 4D 벡터에 대해 설명합니다.
XMSHORT2 부록 및 정규화된 16비트 정수 구성 요소로 구성된 2D 벡터에 대해 설명합니다.
XMSHORT4 16비트 부록 정수 구성 요소로 구성된 4D 벡터입니다.
XMSHORTN2 부호 있는 정규화된 값을 부호 있는 16비트 정수(int16_t 형식)로 저장하기 위한 2D 벡터입니다.
XMSHORTN4 부호 있는 정규화된 값을 부호 있는 16비트 정수(int16_t 형식)로 저장하기 위한 4D 벡터입니다.

XMU555 x,y-및 z-구성 요소가 5비트 부호 없는 정수 값으로 표현되고 w-component가 1비트 정수 값으로 표시되는 4D 벡터입니다.
XMU565 x-및 z-구성 요소가 5비트 부호 없는 정수 값으로 표현되고 y- 구성 요소가 6비트 부호 없는 정수 값으로 표시되는 3D 벡터입니다.
XMUBYTE2 각 구성 요소가 부호 없는 정수인 8비트(1 바이트)인 2D 벡터를 설명합니다.
XMUBYTE4 각 구성 요소가 부호 없는 정수인 8비트(1 바이트)인 4D 벡터를 설명합니다.
XMUBYTEN2 부호 없는 정규화된 값을 부호 있는 8비트(1 바이트) 정수로 저장하기 위한 2D 벡터입니다.
XMUBYTEN4 부호 없는 정규화된 값을 부호 있는 8비트(1 바이트) 정수로 저장하기 위한 3D 벡터입니다.
XMUDEC4 x,y-및 z-구성 요소가 10비트 부호 없는 정수 값으로 표현되고 w-component가 2비트 부호 없는 정수 값으로 표시되는 4D 벡터입니다.
XMUDECN4 부호 없는 정규화된 정수 값을 10비트 부호 없는 x-, y 및 z 구성 요소와 2비트 부호 없는 w-component로 저장하기 위한 4D 벡터입니다.
XMUNIBBLE4 부호 없는 4비트 정수 구성 요소 4개가 있는 4D 벡터입니다.
XMUSHORT2 16비트 부호 없는 정수 구성 요소로 구성된 2D 벡터에 대해 설명합니다.
XMUSHORT4 16비트 부호 없는 정수 구성 요소로 구성된 4D 벡터입니다.

XMUSHORTN2 부호 없는 정규화된 값을 부호 없는 16비트 정수(uint16_t 형식)로 저장하기 위한 2D 벡터입니다.
XMUSHORTN4 부호 있는 16비트 정수(uint16_t 형식)로 부호 없는 정규화된 값을 저장하기 위한 4D 벡터입니다.
XMXDEC4 x,y-및 z-구성 요소가 10비트 부호 있는 정수 값으로 표현되고 w-component가 2비트 부호 없는 정수 값으로 표시되는 4D 벡터입니다.
XMXDECN4 부호 있는 정규화된 값을 10비트 부호 있는 x-,y-및 z- 구성 요소로 저장하고 부호 없는 정규화된 값을 2비트 부호 없는 w-component로 저장하기 위한 4D 벡터입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dispatcherqueue.h 헤더

아티클2023. 10. 28.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

dispatcherqueue.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CreateDispatcherQueueController DispatcherQueueController를 만듭니다. 생성된 DispatcherQueueController를 사용하여 DispatcherQueue의 수명을 만들고 관리하여 디스패처 큐의 스레드에서 대기 중인 작업을 우선 순위순으로 실행합니다.

구조체

DispatcherQueueOptions 새 DispatcherQueueController에 대한 COM 아파트의 스레딩 선호도 및 유형에 대한 옵션을 나타냅니다.

열거형

DISPATCHERQUEUE_THREAD_APARTMENTTYPE 새 DispatcherQueueController에 대한 COM 아파트 유형에 대한 옵션을 지정하는 상수를 정의합니다.
DISPATCHERQUEUE_THREAD_TYPE 새 DispatcherQueueController에 대한 스레딩 선호도에 대한 옵션을 지정하는 상수를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

dmemmgr.h 헤더

아티클2023. 08. 23.

이 헤더는 디스플레이 디바이스 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 디스플레이 디바이스 참조

dmemmgr.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

HeapVidMemAllocAligned HeapVidMemAllocAligned 함수는 DirectDraw 비디오 메모리 힙 관리자를 사용하여 디스플레이 드라이버에 대한 off_screen_memory 할당합니다.
VidMemFree VidMemFree 함수는 HeapVidMemAllocAligned에서 디스플레이 드라이버에 할당된 오프스크림 메모리를 해제합니다.

구조체

DD_GETHEAPALIGNMENTDATA DD_GETHEAPALIGNMENTDATA 구조에는 특정 힙의 필수 맞춤에 대한 데이터가 포함됩니다.
HEAPALIGNMENT HEAPALIGNMENT 구조에는 지정된 디스플레이 메모리 힙에 대한 맞춤 요구 사항을 지정하는 데이터가 포함되어 있습니다.
SURFACEALIGNMENT SURFACEALIGNMENT 구조체는 디스플레이 드라이버가 HeapVidMemAllocAligned에서 할당하는 표면에 대한 맞춤 제한을 설명하는 데 사용됩니다.
VMEMHEAP VMEMHEAP 구조체에는 힙에 대한 정보가 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dmerror.h 헤더

아티클2023. 08. 23.

이 헤더는 COM(구성 요소 개체 모델)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)

dmerror.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

MAKE_HRESULT MAKE_HRESULT 매크로(dmerror.h)는 구성 요소 조각에서 HRESULT 값을 만듭니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

docobj.h 헤더

아티클2023. 08. 23.

이 헤더는 COM(구성 요소 개체 모델)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)

docobj.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IContinueCallback 개체에 계속할지 여부를 주기적으로 요청해야 하는 인터럽트 가능한 프로세스에 대한 일반 콜백 메커니즘을 제공합니다.
IEnumOleDocumentViews 문서 개체에서 지원하는 뷰를 열거합니다.
IOleCommandTarget 개체와 해당 컨테이너가 서로 명령을 디스패치할 수 있도록 합니다. 예를 들어 개체의 도구 모음에는 인쇄, 인쇄 미리 보기, 저장, 새로 만들기 및 확대/축소와 같은 명령에 대한 단추가 포함될 수 있습니다.
IOleDocument 문서 개체가 컨테이너와 통신하여 해당 데이터의 뷰를 만들 수 있도록 합니다.
IOleDocumentSite 문서 개체로 구현된 문서가 현재 위치 활성 개체에 대한 일반 활성화 시퀀스를 무시하고 클라이언트 사이트에 문서 개체로 활성화하도록 직접 지시할 수 있습니다.
IOleDocumentView IOleDocumentView 인터페이스를 사용하면 컨테이너가 문서 개체에서 지원하는 각 보기와 통신할 수 있습니다.
IPrint 특히 일반 및 활성 문서에서 복합 문서를 사용하여 프로그래밍 방식 인쇄를 지원할 수 있습니다.

구조체

OLECMD IOleCommandTarget::QueryStatus 호출을 통해 OLECMDF 열거형의 명령 플래그를 명령 식별자와 연결합니다.
OLECMDTEXT 단일 명령 식별자에 대한 텍스트 이름 또는 상태 문자열을 지정합니다.
Pagerange 페이지 범위를 지정합니다.
PAGESET 인쇄할 하나 이상의 페이지 범위를 식별하고 필요에 따라 짝수 또는 홀수 페이지만 페이지 세트의 일부로 식별합니다.

열거형

DOCMISC 문서 개체에 대한 기타 속성 정보를 제공합니다.
OLECMDEXECOPT 명령 실행 옵션을 지정합니다.
OLECMDF OLECMD 구조체에 지정된 명령에 대해 개체에서 제공하는 지원 유형을 지정합니다.
OLECMDID 실행할 표준 명령을 지정합니다. 이 열거형의 단일 값은 IOleCommandTarget::Exec의 nCmdID 인수에 전달됩니다.
OLECMDID_WINDOWSTATE_FLAG 창 상태를 지정합니다.

OLECMDTEXTF IOleCommandTarget::QueryStatus에 전달된 OLECMDTEXT 구조체에 개체가 저장해야 하는 정보의 형식을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

documenttarget.h 헤더

아티클2023. 12. 16.

이 헤더는 XPS 문서에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- XPS 문서

documenttarget.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

테이블 확장

IPrintDocumentPackageStatusEvent 인쇄 작업의 진행률을 나타냅니다.
IPrintDocumentPackageTarget 사용자가 지원되는 패키지 대상 형식을 열거하고 지정된 형식 ID를 사용하여 만들 수 있습니다. IPrintDocumentPackageTarget은 패키지 인쇄 진행률 및 취소 추적도 지원합니다.
IPrintDocumentPackageTargetFactory 인쇄 작업을 시작하기 위해 IPrintDocumentPackageTarget과 함께 사용됩니다.

구조체

테이블 확장

PrintDocumentPackageStatus PackageStatusUpdated 메서드에서 사용할 페이로드를 정의합니다. 이 구조체는 XPS_JOB_STATUS 제네릭 버전입니다.

열거형

테이블 확장

PrintDocumentPackageCompletion PrintDocumentPackageCompletion 열거형은 인쇄 작업의 상태 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

dot1x.h 헤더

아티클2023. 08. 23.

이 헤더는 네이티브 와이파이에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네이티브 와이파이

dot1x.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

ONEX_AUTH_PARAMS 802.1X 인증에 사용되는 802.1X 인증 매개 변수를 포함합니다.
ONEX_EAP_ERROR 802.1X 인증으로 오류가 발생할 경우 802.1X EAP 오류가 포함됩니다.
ONEX_RESULT_UPDATE_DATA 802.1X 인증에 대한 상태 변경 내용에 대한 정보를 포함합니다.
ONEX_STATUS 현재 802.1X 인증 상태 포함합니다.
ONEX_VARIABLE_BLOB 가변 크기 멤버를 포함하기 위해 다른 802.1X 인증 구조의 멤버로 사용됩니다.

열거형

ONEX_AUTH_IDENTITY 802.1X 인증 상태 사용되는 ID의 가능한 값을 지정합니다.
ONEX_AUTH_RESTART_REASON 802.1X 인증이 다시 시작되는 가능한 이유를 지정합니다.

ONEX_AUTH_STATUS 802.1X 인증 상태 가능한 값을 지정합니다.
ONEX_EAP_METHOD_BACKEND_SUPPORT 802.1X 인증을 위해 지원되는 에 구성된 EAP 메서드가 인증 서버에서 지원되는지 여부에 대한 가능한 값을 지정합니다.
ONEX_NOTIFICATION_TYPE 802.1X 모듈 알림에 대한 WLAN_NOTIFICATION_DATA 구조체의 NotificationCode 멤버의 가능한 값을 지정합니다.
ONEX_REASON_CODE 802.1X 인증이 실패한 이유를 나타내는 가능한 값을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dpa_dsa.h 헤더

아티클2023. 06. 08.

이 헤더는 Windows 컨트롤에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 컨트롤

dpa_dsa.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DPA_AppendPtr DPA(동적 포인터 배열) 끝에 새 항목을 삽입합니다.
DPA_Clone DPA(동적 포인터 배열)를 복제합니다.
DPA_Create DPA(동적 포인터 배열)를 만듭니다.
DPA_CreateEx 지정된 크기 및 힙 위치를 사용하여 DPA(동적 포인터 배열)를 만듭니다.
DPA_DeleteAllPtrs DPA(동적 포인터 배열)에서 모든 항목을 제거하고 그에 따라 DPA를 축소합니다.
DPA_DeletePtr DPA(동적 포인터 배열)에서 항목을 제거합니다. 제거된 항목을 수용하기 위해 필요한 경우 DPA가 축소됩니다.
DPA_Destroy DPA(동적 포인터 배열)를 해제합니다.
DPA_DestroyCallback DPA(동적 포인터 배열)의 각 요소에서 pfnCB를 호출한 다음 DPA를 해제합니다.

DPA_EnumCallback DPA(동적 포인터 배열)를 반복하고 각 항목에서 pfnCB를 호출합니다.
DPA_FastDeleteLastPtr DPA(동적 포인터 배열)에서 마지막 포인터를 삭제합니다.
DPA_FastGetPtr DPA(동적 포인터 배열)에서 지정된 포인터의 값을 가져옵니다.
DPA_GetPtr DPA(동적 포인터 배열)에서 항목을 가져옵니다.
DPA_GetPtrCount DPA(동적 포인터 배열)의 포인터 수를 가져옵니다.
DPA_GetPtrIndex DPA(동적 포인터 배열)에 있는 일치하는 항목의 인덱스 를 가져옵니다.
DPA_GetPtrPtr DPA(동적 포인터 배열)의 내부 포인터 배열에 대한 포인터를 가져옵니다.
DPA_GetSize DPA(동적 포인터 배열)의 크기를 가져옵니다.
DPA_Grow DPA(동적 포인터 배열)의 포인터 수를 변경합니다.
DPA_InsertPtr DPA(동적 포인터 배열)의 지정된 위치에 새 항목을 삽입합니다. 필요한 경우 DPA는 새 항목을 수용하도록 확장됩니다.
DPA_LoadStream 지정된 콜백 함수를 호출하여 각 요소를 읽어 스트림에서 DPA(동적 포인터 배열)를 로드합니다.
DPA_Merge 두 DBA(동적 포인터 배열)의 내용을 결합합니다.

DPA_SaveStream 헤더를 작성한 다음 지정된 콜백 함수를 호출하여 각 요소를 작성하여 DPA(동적 포인터 배열)를 스트림에 저장합니다.
DPA_Search DPA(동적 포인터 배열)에서 항목을 찾습니다.
DPA_SetPtr DPA(동적 포인터 배열)의 항목에 값을 할당합니다.
DPA_SetPtrCount DPA(동적 포인터 배열)의 포인터 수를 설정합니다.
DPA_Sort DPA(동적 포인터 배열)의 항목을 정렬합니다.
DPA_SortedInsertPtr 지정된 기존 항목 앞이나 뒤에 새 항목을 삽입합니다.
DSA_AppendItem DSA(동적 구조 배열)의 끝에 새 항목을 추가합니다.
DSA_Clone DSA(동적 구조체 배열)를 복제합니다.
DSA_Create DSA(동적 구조 배열)를 만듭니다.
DSA_DeleteAllItems DSA(동적 구조 배열)에서 모든 항목을 삭제합니다.
DSA_DeleteItem DSA(동적 구조 배열)에서 항목을 삭제합니다.
DSA_Destroy DSA(동적 구조 배열)를 해제합니다.

DSA_DestroyCallback 각 항목에서 지정된 콜백 함수를 호출하여 DSA(동적 구조 배열)를 반복합니다. 배열 끝에 도달하면 DSA가 해제됩니다.
DSA_EnumCallback DSA(동적 구조 배열)를 반복하고 각 항목에서 pfnCB를 호출합니다.
DSA_GetItem DSA(동적 구조 배열)에서 요소를 가져옵니다.
DSA_GetItemCount DSA(동적 구조 배열)의 항목 수를 가져옵니다.
DSA_GetItemPtr DSA(동적 구조 배열)에서 요소에 대한 포인터를 가져옵니다.
DSA_GetSize DSA(동적 구조 배열)의 크기를 가져옵니다.
DSA_InsertItem DSA(동적 구조 배열)에 새 항목을 삽입합니다. 필요한 경우 DSA는 새 항목을 수용하도록 확장됩니다.
DSA_SetItem DSA(동적 구조 배열)에서 요소의 내용을 설정합니다.
DSA_Sort DSA(동적 구조 배열)의 항목을 정렬합니다.
DSA_Sort DSA_Sort 함수는 DSA(동적 구조 배열)의 항목을 정렬합니다.
Str_SetPtrW ppszCurrent를 pszNew의 복사본으로 설정하고 필요한 경우 이전 값을 해제합니다.

콜백 함수

PFNDACOMPARE DSA_Sort에서 사용하는 비교 함수의 프로토타입을 정의합니다.
PFNDACOMPARECONST 비교 중인 항목이 상수 개체일 때 DSA_Sort에서 사용하는 비교 함수의 프로토타입을 정의합니다.
PFNDAENUMCALLBACK DSA(동적 구조 배열) 및 DPA(동적 포인터 배열) 함수에서 사용하는 콜백 함수의 프로토타입을 정의합니다.
PFNDAENUMCALLBACKCONST 관련된 항목이 상수 데이터에 대한 포인터일 때 DSA(동적 구조 배열) 및 DPA(동적 포인터 배열) 함수에서 사용하는 콜백 함수의 프로토타입을 정의합니다.
PFNDPAMERGE DPA_Merge에서 사용하는 병합 함수의 프로토타입을 정의합니다.
PFNDPAMERGECONST 상수 값을 사용하여 DPA_Merge에서 사용하는 병합 함수의 프로토타입을 정의합니다.
PFNDPASTREAM DPA_LoadStream 및 DPA_SaveStream에서 사용하는 콜백 함수의 프로토타입을 정의합니다.

구조체

DPASTREAMINFO PFNDPASTREAM 콜백 함수에서 사용하는 스트림 항목을 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dpapi.h 헤더

아티클2023. 08. 23.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

dpapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CryptProtectData DATA_BLOB 구조의 데이터에 대한 암호화를 수행합니다.
CryptProtectMemory 는 다른 사용자가 프로세스에서 중요한 정보를 볼 수 없도록 메모리를 암호화합니다.
CryptUnprotectData DATA_BLOB 구조에서 데이터의 무결성 검사 해독하고 수행합니다.
CryptUnprotectMemory CryptProtectMemory 함수를 사용하여 암호화된 메모리의 암호를 해독합니다.
CryptUpdateProtectedState 사용자의 SID(보안 식별자)가 변경된 후 현재 사용자의 master 키를 마이그레이션합니다.

구조체

CRYPT_INTEGER_BLOB CryptoAPI CRYPT_INTEGER_BLOB(dpapi.h) 구조체는 임의의 바이트 배열에 사용됩니다. Wincrypt.h에서 선언되며 다양한 데이터 형식을 포함할 수 있는 개체에 유연성을 제공합니다. (CRYPT_INTEGER_BLOB)
CRYPTPROTECT_PROMPTSTRUCT

CryptProtectData 및 CryptUnprotectData 함수를 사용할 때 프롬프트가 표시되는 시기와 위치에 대한 정보와 프롬프트의 텍스트를 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dragdropinterop.h 헤더

아티클2023. 08. 23.

이 헤더는 WinRT 상호 운용성에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- WinRT 상호 운용성
- CoreDragDropManager 클래스

dragdropinterop.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDragDropManagerInterop WinRT CoreDragDropManager 개체와의 상호 운용성을 사용하도록 설정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

drt.h 헤더

아티클2023. 08. 23.

이 헤더는 피어 투 피어에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 피어 투 피어

drt.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DrtClose DrtClose 함수는 DRT의 로컬 instance 닫습니다.
DrtContinueSearch DrtContinueSearch 함수는 키에 대한 반복 검색을 계속합니다.
DrtCreateDerivedKey DrtCreateDerivedKey 함수는 DRT가 파생 키 보안 공급자를 사용할 때 DrtRegisterKey에서 활용할 수 있는 키를 만듭니다.
DrtCreateDerivedKeySecurityProvider DrtCreateDerivedKeySecurityProvider 함수는 분산 라우팅 테이블에 대한 파생 키 보안 공급자를 만듭니다.
DrtCreateDnsBootstrapResolver DrtCreateDnsBootstrapResolver 함수는 GetAddrInfo 시스템 함수를 사용하여 DRT 메시에 이미 있는 알려진 노드의 호스트 이름을 resolve 부트스트랩 확인자를 만듭니다.
DrtCreateIpv6UdpTransport DrtCreateIpv6UdpTransport 함수는 IPv6 UDP 프로토콜을 기반으로 전송을 만듭니다.
DrtCreateNullSecurityProvider DrtCreateNullSecurityProvider 함수는 null 보안 공급자를 만듭니다. 이 보안 공급자는 노드가 키를 인증할 필요가 없습니다.
DrtCreatePnrpBootstrapResolver

DrtCreatePnrpBootstrapResolver.
DrtDeleteDerivedKeySecurityProvider DrtDeleteDerivedKeySecurityProvider 함수는 분산 라우팅 테이블에 대한 파생 키 보안 공급자를 삭제합니다.
DrtDeleteDnsBootstrapResolver DrtDeleteDnsBootstrapResolver 함수는 DNS 부트스트랩 공급자 instance 삭제합니다.
DrtDeleteIpv6UdpTransport DrtDeleteIpv6UdpTransport 함수는 IPv6 UDP 프로토콜을 기반으로 전송을 삭제합니다.
DrtDeleteNullSecurityProvider DrtDeleteNullSecurityProvider 함수는 분산 라우팅 테이블에 대한 null 보안 공급자를 삭제합니다.
DrtDeletePnrpBootstrapResolver DrtDeletePnrpBootstrapResolver 함수는 PNRP(피어 이름 확인 프로토콜)를 기반으로 부트스트랩 확인자를 삭제합니다.
DrtEndSearch DrtEndSearch 함수는 DRT에서 키 검색을 취소합니다.
DrtGetEventData DrtGetEventData 함수는 신호가 있는 이벤트와 연결된 이벤트 데이터를 검색합니다.
DrtGetEventDataSize DrtGetEventDataSize 함수는 신호 이벤트와 연결된 DRT_EVENT_DATA 구조체의 크기를 반환합니다.
DrtGetInstanceName DrtGetInstanceName 함수는 지정된 DRT 핸들에 해당하는 분산 라우팅 테이블 instance 전체 이름을 검색합니다.
DrtGetInstanceNameSize DrtGetInstanceNameSize 함수는 분산 라우팅 테이블 instance 이름의 크기를 반환합니다.
DrtGetSearchPath DrtGetSearchPath 함수는 검색 작업 중에 연락한 노드 목록을 반환합니다.

DrtGetSearchPathSize DrtGetSearchPathSize 함수는 검색 작업에 사용된 노드 수를 나타내는 검색 경로의 크기를 반환합니다.
DrtGetSearchResult DrtGetSearchResult 함수입니다.
DrtGetSearchResultSize DrtGetSearchResultSize 함수는 사용 가능한 다음 검색 결과의 크기를 반환합니다.
DrtOpen DrtOpen 함수는 DRT_SETTINGS 구조에 지정된 조건에 대해 로컬 분산 라우팅 테이블 instance 만듭니다.
DrtRegisterKey DrtRegisterKey 함수는 DRT에 키를 등록합니다.
DrtStartSearch DrtStartSearch 함수는 DRT_SEARCH_INFO 구조에 지정된 조건을 사용하여 DRT에서 키를 검색합니다.
DrtUnregisterKey DrtUnregisterKey 함수는 DRT에서 키를 등록 취소합니다.
DrtUpdateKey DrtUpdateKey 함수는 등록된 키와 연결된 애플리케이션 데이터를 업데이트합니다.

구조체

DRT_ADDRESS DRT_ADDRESS 구조에는 검색에 참여한 DRT 노드에 대한 엔드포인트 정보가 포함됩니다. 이 정보는 연결 문제를 디버깅하는 데 사용하기 위한 것입니다.
DRT_ADDRESS_LIST

DRT_ADDRESS_LIST 구조체에는 키를 검색하는 동안 접촉한 노드를 나타내는 DRT_ADDRESS 구조 집합이 포함되어 있습니다.
DRT_BOOTSTRAP_PROVIDER DRT_BOOTSTRAP_PROVIDER 구조체는 부트스트랩 공급자가 구현해야 하는 DRT 인터페이스를 정의합니다.
DRT_DATA DRT_DATA 구조에는 데이터 Blob이 포함됩니다. 이 구조체는 여러 DRT 함수에서 사용됩니다.
DRT_EVENT_DATA DRT_EVENT_DATA.
DRT_REGISTRATION DRT_REGISTRATION 구조에는 등록을 구성하는 키 및 애플리케이션 데이터가 포함됩니다.
DRT_SEARCH_INFO DRT_SEARCH_INFO 구조체는 DrtStartSearch에서 발급된 검색 쿼리를 나타냅니다.
DRT_SEARCH_RESULT DRT_SEARCH_RESULT.
DRT_SECURITY_PROVIDER DRT_SECURITY_PROVIDER 구조는 보안 공급자가 구현해야 하는 DRT 인터페이스를 정의합니다.
DRT_SETTINGS DRT_SETTINGS 구조에는 로컬 분산 라우팅 테이블에서 사용하는 설정이 포함됩니다.

열거형

DRT_ADDRESS_FLAGS DRT_ADDRESS_FLAGS 열거형입니다.
DRT_EVENT_TYPE DRT_EVENT_TYPE 열거형은 분산 라우팅 테이블에서 발생할 수 있는 이벤트 집합을 정의합니다.

DRT_LEAFSET_KEY_CHANGE_TYPE DRT_LEAFSET_KEY_CHANGE_TYPE 열거형은 로컬로 등록된 키의 리프 집합에 있는 노드에서 발생할 수 있는 변경 내용 집합을 정의합니다.
DRT_MATCH_TYPE DRT_MATCH_TYPE 열거형은 DrtStartSearch API를 사용하여 검색을 시작한 후 DrtGetSearchResult에서 반환된 검색 결과의 정확도를 정의합니다.
DRT_REGISTRATION_STATE DRT_REGISTRATION_STATE 열거형은 등록된 키에 대한 법적 상태 집합을 정의합니다.
DRT_SCOPE DRT_SCOPE 열거형은 DrtCreateIpv6UdpTransport에서 만든 IPv6 UDP 전송을 사용하는 동안 DRT가 작동하는 IPv6 범위 집합을 정의합니다.
DRT_SECURITY_MODE DRT_SECURITY_MODE 열거형은 DRT에 대한 가능한 보안 모드를 정의합니다. 보안 모드는 DRT_SETTINGS 구조체의 필드에 의해 지정됩니다.
DRT_STATUS DRT_STATUS 열거형은 로컬 DRT instance 상태 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dsadmin.h 헤더

아티클2023. 08. 23.

이 헤더는 Active Directory Domain Services 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Active Directory Domain Services

dsadmin.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDsAdminCreateObj 애플리케이션 또는 구성 요소에서 지정된 개체 클래스에 대한 만들기 마법사를 프로그래밍 방식으로 시작하는 데 사용됩니다.
IDsAdminNewObj IDsAdminNewObj 인터페이스는 기본 또는 보조 개체 만들기 마법사 확장에서 페이지 수 데이터를 가져오고 마법사에서 명령 단추를 제어하는 데 사용됩니다.
IDsAdminNewObjExt IDsAdminNewObjExt 인터페이스는 개체 만들기 마법사 확장에 의해 구현됩니다.
IDsAdminNewObjPrimarySite IDsAdminNewObjPrimarySite 인터페이스는 시스템에서 구현되며 기본 개체 만들기 확장 프로그램에서 Active Directory Domain Services 새 임시 개체를 만든 다음 개체를 영구 메모리에 커밋하는 데 사용됩니다.
IDsAdminNotifyHandler IDsAdminNotifyHandler 인터페이스는 Active Directory 관리 알림 처리기에 의해 구현됩니다.

구조체

DSA_NEWOBJ_DISPINFO

IDsAdminNewObjExt::Initialize 메서드와 함께 Active Directory Domain Services 개체 만들기 마법사에 대한 추가 데이터를 제공하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dsclient.h 헤더

아티클2023. 08. 23.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Active Directory Domain Services
- Microsoft 관리 콘솔 2.0

dsclient.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDsBrowseDomainTree IDsBrowseDomainTree 인터페이스는 애플리케이션에서 도메인 브라우저 대화 상자를 표시하거나 지정된 컴퓨터와 관련된 신뢰 도메인 목록을 가져오는 데 사용됩니다.
IDsDisplaySpecifier displaySpecifier 클래스의 Active Directory 도메인 Service 개체에 대한 액세스를 제공합니다.

Functions

DsBrowseForContainerA Active Directory Domain Services 컨테이너 개체를 검색하는 데 사용되는 대화 상자를 표시합니다. (ANSI)
DsBrowseForContainerW Active Directory Domain Services 컨테이너 개체를 검색하는 데 사용되는 대화 상자를 표시합니다. (유니코드)
DsGetFriendlyClassName 개체 클래스의 지역화된 이름을 검색합니다.
DsGetIcon 지정된 개체 클래스의 아이콘을 가져옵니다.

콜백 함수

LPDSENUMATTRIBUTES DSEnumAttributesCallback 함수는 IDsDisplaySpecifier::EnumClassAttributes 메서드에 의해 열거된 각 특성에 대해 한 번 호출되는 애플리케이션 정의 콜백 함수입니다.

구조체

DOMAIN_DESC IDsBrowseDomainTree::GetDomains 메서드를 사용하여 가져온 도메인 트리의 요소에 대한 데이터를 포함합니다.
DOMAIN_TREE DOMAINTREE 구조에는 IDsBrowseDomainTree::GetDomains 메서드를 사용하여 얻은 도메인 트리의 노드에 대한 데이터가 포함됩니다. 트리 노드의 각 도메인은 DOMAINDESC 구조체로 표시됩니다.
DSBITEMA Active Directory 컨테이너 브라우저 대화 상자의 항목에 대한 데이터를 포함합니다. (ANSI)
DSBITEMW Active Directory 컨테이너 브라우저 대화 상자의 항목에 대한 데이터를 포함합니다. (유니코드)
DSBROWSEINFOA DSBROWSEINFO 구조체는 DsBrowseForContainer 함수와 함께 사용하여 Active Directory 컨테이너 브라우저 대화 상자에 대한 데이터를 제공하고 반환합니다. (ANSI)
DSBROWSEINFOW DSBROWSEINFO 구조체는 DsBrowseForContainer 함수와 함께 사용하여 Active Directory 컨테이너 브라우저 대화 상자에 대한 데이터를 제공하고 반환합니다. (유니코드)
DSCLASSCREATIONINFO IDsDisplaySpecifier::GetClassCreationInfo 메서드와 함께 개체 클래스에 대한 클래스 만들기 마법사 개체에 대한 데이터를 보관하는 데 사용됩니다.

DSDISPLAYSPECOPTIONS 사용되는 표시 지정자에 대한 상황에 맞는 메뉴 또는 속성 페이지 확장에 데이터를 제공하는 데 사용됩니다.
DSOBJECT 디렉터리 개체 데이터를 포함합니다.
DSOBJECTNAMES DSOBJECTNAMES 구조체는 Active Directory 속성 시트 또는 상황에 맞는 메뉴 확장에서 사용할 디렉터리 개체 데이터를 포함하는 데 사용됩니다.
DSPROPERTYPAGEINFO DSPROPERTYPAGEINFO 구조체는 Active Directory 속성 시트 확장에서 확장에 대한 정적 등록 데이터를 가져오는 데 사용됩니다. 이 구조체는 CFSTR_DSPROPERTYPAGEINFO 클립보드 형식으로 제공됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dsgetdc.h 헤더

아티클2023. 08. 23.

이 헤더는 Active Directory Domain Services 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Active Directory Domain Services

dsgetdc.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DsAddressToSiteNamesA 지정된 주소에 해당하는 사이트 이름을 가져옵니다. (ANSI)
DsAddressToSiteNamesExA 지정된 주소에 해당하는 사이트 및 서브넷 이름을 가져옵니다. (ANSI)
DsAddressToSiteNamesExW 지정된 주소에 해당하는 사이트 및 서브넷 이름을 가져옵니다. (유니코드)
DsAddressToSiteNamesW 지정된 주소에 해당하는 사이트 이름을 가져옵니다. (유니코드)
DsDeregisterDnsHostRecordsA DsDeregisterDnsHostRecords 함수는 도메인 컨트롤러에 의해 등록된 A 형식 레코드를 제외하고 DNS 항목을 삭제합니다. 관리자, 계정 운영자 또는 서버 운영자만 이 함수를 호출할 수 있습니다. (ANSI)
DsDeregisterDnsHostRecordsW DsDeregisterDnsHostRecords 함수는 도메인 컨트롤러에 의해 등록된 A 형식 레코드를 제외하고 DNS 항목을 삭제합니다. 관리자, 계정 운영자 또는 서버 운영자만 이 함수를 호출할 수 있습니다. (유니코드)
DsEnumerateDomainTrustsA 지정된 도메인에 대한 도메인 신뢰 데이터를 가져옵니다. (ANSI)

DsEnumerateDomainTrustsW 지정된 도메인에 대한 도메인 신뢰 데이터를 가져옵니다. (유니코드)
DsGetDcCloseW 도메인 컨트롤러 열거형 작업을 닫습니다.
DsGetDcNameA 지정된 도메인에 있는 도메인 컨트롤러의 이름을 반환합니다. (ANSI)
DsGetDcNameW 지정된 도메인에 있는 도메인 컨트롤러의 이름을 반환합니다. (유니코드)
DsGetDcNextA 도메인 컨트롤러 열거 작업에서 다음 도메인 컨트롤러를 검색합니다. (ANSI)
DsGetDcNextW 도메인 컨트롤러 열거 작업에서 다음 도메인 컨트롤러를 검색합니다. (유니코드)
DsGetDcOpenA 새 도메인 컨트롤러 열거형 작업을 엽니다. (ANSI)
DsGetDcOpenW 새 도메인 컨트롤러 열거형 작업을 엽니다. (유니코드)
DsGetDcSiteCoverageA DsGetDcSiteCoverage 함수는 도메인 컨트롤러가 적용되는 모든 사이트의 사이트 이름을 반환합니다. (ANSI)
DsGetDcSiteCoverageW DsGetDcSiteCoverage 함수는 도메인 컨트롤러가 적용되는 모든 사이트의 사이트 이름을 반환합니다. (유니코드)
DsGetForestTrustInformationW 지정된 도메인에 대한 포리스트 트러스트 데이터를 가져옵니다.
DsGetSiteNameA DsGetSiteName 함수는 컴퓨터가 있는 사이트의 이름을 반환합니다. (ANSI)

DsGetSiteNameW DsGetSiteName 함수는 컴퓨터가 있는 사이트의 이름을 반환합니다. (유니코드)
DsMergeForestTrustInformationW 새 포리스트 트러스트 데이터 구조의 변경 내용을 이전 포리스트 트러스트 데이터 구조와 병합합니다.
DsValidateSubnetNameA DsValidateSubnetName 함수는 xxx.xxx.xxx.xxx/YY 양식의 서브넷 이름의 유효성을 검사합니다. (ANSI)
DsValidateSubnetNameW DsValidateSubnetName 함수는 xxx.xxx.xxx.xxx/YY 양식의 서브넷 이름의 유효성을 검사합니다. (유니코드)

구조체

DOMAIN_CONTROLLER_INFOA DsGetDcName 함수와 함께 도메인 컨트롤러에 대한 데이터를 수신하는 데 사용됩니다. (ANSI)
DOMAIN_CONTROLLER_INFOW DsGetDcName 함수와 함께 도메인 컨트롤러에 대한 데이터를 수신하는 데 사용됩니다. (유니코드)
DS_DOMAIN_TRUSTSA DsEnumerateDomainTrusts 함수와 함께 도메인에 대한 신뢰 데이터를 포함하는 데 사용됩니다. (ANSI)
DS_DOMAIN_TRUSTSW DsEnumerateDomainTrusts 함수와 함께 도메인에 대한 신뢰 데이터를 포함하는 데 사용됩니다. (유니코드)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dskquota.h 헤더

아티클2023. 08. 23.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지
- Windows 셸

dskquota.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDiskQuotaControl 단일 NTFS 파일 시스템 볼륨의 디스크 할당량 기능을 제어합니다.
IDiskQuotaEvents 할당량 관련 이벤트 알림을 받습니다.
IDiskQuotaUser 볼륨 할당량 정보 파일의 단일 사용자 할당량 항목을 나타냅니다.
IDiskQuotaUserBatch 컨테이너에 여러 할당량 사용자 개체를 추가한 다음, 단일 호출에서 업데이트를 위해 제출됩니다.
IEnumDiskQuotaUsers 볼륨의 사용자 할당량 항목을 열거합니다.

구조체

DISKQUOTA_USER_INFORMATION 사용자별 할당량 정보를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dsparse.h 헤더

아티클2023. 08. 23.

이 헤더는 Active Directory Domain Services 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Active Directory Domain Services

dsparse.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DsCrackSpnA SPN(서비스 사용자 이름)을 해당 구성 요소 문자열로 구문 분석합니다. (ANSI)
DsCrackSpnW SPN(서비스 사용자 이름)을 해당 구성 요소 문자열로 구문 분석합니다. (유니코드)
DsCrackUnquotedMangledRdnA DsCrackUnquotedMangledRdn 함수는 지정된 상대 고유 이름을 unmangles(unencodes)하고 디코딩된 GUID와 사용된 mangling 형식을 모두 반환합니다. (ANSI)
DsCrackUnquotedMangledRdnW DsCrackUnquotedMangledRdn 함수는 지정된 상대 고유 이름을 unmangles(unencodes)하고 디코딩된 GUID와 사용된 mangling 형식을 모두 반환합니다. (유니코드)
DsGetRdnW 첫 번째 상대 고유 이름의 키와 값을 검색하고 고유 이름 문자열에서 다음 상대 고유 이름에 대한 포인터를 검색합니다.
DsIsMangledDnA DsIsMangledDn 함수는 DN(고유 이름)의 첫 번째 RDN(상대 고유 이름)이 지정된 형식의 잘못된 이름인지 확인합니다. (ANSI)
DsIsMangledDnW DsIsMangledDn 함수는 DN(고유 이름)의 첫 번째 RDN(상대 고유 이름)이 지정된 형식의 잘못된 이름인지 확인합니다. (유니코드)

DsIsMangledRdnValueA 지정된 상대 고유 이름 값이 지정된 형식의 잘못된 이름인지 확인합니다. (ANSI)
DsIsMangledRdnValueW 지정된 상대 고유 이름 값이 지정된 형식의 잘못된 이름인지 확인합니다. (유니코드)
DsMakeSpnA 서비스 instance 식별하는 SPN(서비스 사용자 이름)을 생성합니다. (ANSI)
DsMakeSpnW 서비스 instance 식별하는 SPN(서비스 사용자 이름)을 생성합니다. (유니코드)
DsQuoteRdnValueA RDN 값에 따옴표가 필요한 문자가 포함된 경우 RDN을 따옴표가 붙은 RDN 값으로 변환합니다. (ANSI)
DsQuoteRdnValueW RDN 값에 따옴표가 필요한 문자가 포함된 경우 RDN을 따옴표가 붙은 RDN 값으로 변환합니다. (유니코드)
DsUnquoteRdnValueA DsUnquoteRdnValue 함수는 따옴표가 붙은 RDN 값을 따옴표가 지정되지 않은 RDN 값으로 다시 변환하는 클라이언트 호출입니다. (ANSI)
DsUnquoteRdnValueW DsUnquoteRdnValue 함수는 따옴표가 붙은 RDN 값을 따옴표가 지정되지 않은 RDN 값으로 다시 변환하는 클라이언트 호출입니다. (유니코드)

열거형

DS_MANGLE_FOR DS_MANGLE_FOR 열거형은 상대 고유 이름이 손상되었는지(인코딩) 및 맹글링이 발생하는 형태를 정의하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dsquery.h 헤더

아티클2023. 08. 23.

이 헤더는 Active Directory Domain Services 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Active Directory Domain Services

dsquery.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

DSCOLUMN DSCOLUMN 구조체는 디렉터리 서비스 쿼리 대화 상자의 열을 나타냅니다. 이 구조체의 배열은 DSQUERYPARAMS 구조체에 포함되어 있습니다.
DSQUERYCLASSLIST DSQUERYCLASSLIST 구조는 디렉터리 서비스 쿼리가 만들어지는 클래스 목록을 설명합니다.
DSQUERYINITPARAMS 디렉터리 서비스 쿼리에서 찾아보기 대화 상자를 초기화하는 데 사용되는 데이터에 대해 설명합니다.
DSQUERYPARAMS DSQUERYPARAMS 구조에는 디렉터리 서비스를 검색할 때 디렉터리 서비스 쿼리에서 사용하는 쿼리 데이터가 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dsrole.h 헤더

아티클2023. 08. 23.

이 헤더는 Active Directory Domain Services 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Active Directory Domain Services

dsrole.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DsRoleFreeMemory DsRoleGetPrimaryDomainInformation 함수에서 할당한 메모리를 해제합니다.
DsRoleGetPrimaryDomainInformation 컴퓨터의 상태 데이터를 검색합니다.

구조체

DSROLE_OPERATION_STATE_INFO 컴퓨터에 대한 작동 상태 데이터를 포함하기 위해 DsRoleGetPrimaryDomainInformation 함수와 함께 사용됩니다.
DSROLE_PRIMARY_DOMAIN_INFO_BASIC 도메인 데이터를 포함하기 위해 DsRoleGetPrimaryDomainInformation 함수와 함께 사용됩니다.
DSROLE_UPGRADE_STATUS_INFO DsRoleGetPrimaryDomainInformation 함수와 함께 도메인 업그레이드 상태 데이터를 포함하는 데 사용됩니다.

열거형

DSROLE_MACHINE_ROLE 컴퓨터 역할을 지정하기 위해 DSROLE_PRIMARY_DOMAIN_INFO_BASIC 구조체의 MachineRole 멤버와 함께 사용됩니다.
DSROLE_OPERATION_STATE 컴퓨터의 작동 상태를 나타내기 위해 DSROLE_OPERATION_STATE_INFO 구조와 함께 사용됩니다.
DSROLE_PRIMARY_DOMAIN_INFO_LEVEL DsRoleGetPrimaryDomainInformation 함수와 함께 검색할 데이터 형식을 지정하는 데 사용됩니다.
DSROLE_SERVER_STATE 서버의 역할을 나타내기 위해 DSROLE_UPGRADE_STATUS_INFO 구조와 함께 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dssec.h 헤더

아티클2023. 08. 23.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

dssec.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DSCreateISecurityInfoObject 지정된 DS(디렉터리 서비스) 개체와 연결된 ISecurityInformation 인터페이스의 instance 만듭니다.
DSCreateISecurityInfoObjectEx 지정된 서버에서 지정된 DS(디렉터리 서비스) 개체와 연결된 ISecurityInformation 인터페이스의 instance 만듭니다.
DSCreateSecurityPage Active Directory 개체에 대한 보안 속성 페이지를 만듭니다.
DSEditSecurity DS(Directory Services) 개체의 보안을 편집하기 위한 모달 대화 상자를 표시합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dvp.h 헤더

아티클2023. 08. 23.

이 헤더는 디스플레이 디바이스 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 디스플레이 디바이스 참조

dvp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

DDVIDEOPORTBANDWIDTH DDVIDEOPORTBANDWIDTH 구조체는 특정 VPE(비디오 포트 확장) 개체/픽셀 형식 구성과 함께 사용할 때 오버레이의 대역폭 특성을 설명합니다.
DDVIDEOPORTCAPS DDVIDEOPORTCAPS 구조체는 하드웨어 비디오 포트의 기능 및 맞춤 제한을 설명합니다.
DDVIDEOPORTDESC DDVIDEOPORTDESC 구조체는 생성되는 VPE(비디오 포트 확장) 개체에 대해 설명합니다.
DDVIDEOPORTINFO DDVIDEOPORTINFO 구조는 드라이버가 비디오 데이터를 표면(또는 표면)으로 전송하는 방법을 설명합니다. DDVIDEOPORTINFO는 DD_VIDEOPORT_LOCAL 구조체의 멤버입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dwmapi.h 헤더

아티클2023. 08. 23.

이 헤더는 DWM(데스크톱 창 관리자)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DWM(데스크톱 창 관리자)

dwmapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DwmAttachMilContent 이 함수는 구현되지 않습니다. (DwmAttachMilContent)
DwmDefWindowProc 클라이언트가 아닌 영역 내에서 DWM(데스크톱 창 관리자) 적중 테스트에 대한 기본 창 프로시저입니다.
DwmDetachMilContent 이 함수는 구현되지 않습니다. (DwmDetachMilContent)
DwmEnableBlurBehindWindow 지정된 창에서 흐림 효과를 사용하도록 설정합니다.
DwmEnableComposition DWM(데스크톱 창 관리자) 컴퍼지션을 사용하거나 사용하지 않도록 설정합니다.
DwmEnableMMCSS 호출 프로세스가 활성 상태인 동안 DWM(데스크톱 창 관리자)에 MMCSS(멀티미디어 클래스 일정 서비스) 예약을 옵트인하거나 옵트아웃하도록 알립니다.
DwmExtendFrameIntoClientArea 창 프레임을 클라이언트 영역으로 확장합니다.
DwmFlush 현재 미해결 상태인 모든 Microsoft DirectX 표면 업데이트가 수행될 때까지 호출자를 차단하는 플

러시 호출을 실행합니다. 이렇게 하면 우선 순위가 매우 낮은 매우 복잡한 장면 또는 호출 프로세스를 보정합니다.
DwmGetColorizationColor DWM(데스크톱 창 관리자) 유리 컴퍼지션에 사용되는 현재 색을 검색합니다.
DwmGetCompositionTimingInfo 지정된 창에 대한 현재 컴퍼지션 타이밍 정보를 검색합니다.
DwmGetGraphicsStreamClient 이 함수는 구현되지 않습니다. (DwmGetGraphicsStreamClient)
DwmGetGraphicsStreamTransformHint 이 함수는 구현되지 않습니다. (DwmGetGraphicsStreamTransformHint)
DwmGetTransportAttributes 전송 특성을 검색합니다.
DwmGetUnmetTabRequirements 이 함수는 Windows에서 구현되지 않은 실험적 기능의 일부였습니다. 영향을 주지 않으며 사용하면 안 됩니다.
DwmGetWindowAttribute 창에 적용된 지정된 DWM(데스크톱 창 관리자) 특성의 현재 값을 검색합니다.
DwmInvalidateIconicBitmaps 미리 보기와 피킹 표현 모두 창에서 이전에 제공된 모든 아이콘 비트맵을 새로 고쳐야 함을 나타내기 위해 애플리케이션에서 호출됩니다.
DwmIsCompositionEnabled DWM(데스크톱 창 관리자) 컴퍼지션이 사용되는지 여부를 나타내는 값을 가져옵니다. Windows 7 이하를 실행하는 컴퓨터의 애플리케이션은 WM_DWMCOMPOSITIONCHANGED 알림을 처리하여 컴퍼지션 상태 변경을 수신 대기할 수 있습니다.
DwmModifyPreviousDxFrameDuration 이전 프레임이 표시될 모니터 새로 고침 수를 변경합니다. DwmModifyPreviousDxFrameDuration은 더 이상 지원되지 않습니다. Windows 8.1 시작해서 DwmModifyPreviousDxFrameDuration에 대한 호출은 항상 E_NOTIMPL 반환합니다.

DwmQueryThumbnailSourceSize DWM(데스크톱 창 관리자) 미리 보기의 원본 크기를 검색합니다.
DwmRegisterThumbnail 대상 창과 원본 창 간에 DWM(데스크톱 창 관리자) 썸네일 관계를 만듭니다.
DwmRenderGesture 터치 접촉이 제스처로 인식되었으며 DWM이 해당 제스처에 대한 피드백을 그려야 한다는 것을 DWM(데스크톱 창 관리자)에 알립니다.
DwmSetDxFrameDuration 표시된 프레임을 표시할 모니터 새로 고침 수를 설정합니다. DwmSetDxFrameDuration은 더 이상 지원되지 않습니다. Windows 8.1 DwmSetDxFrameDuration에 대한 호출은 항상 E_NOTIMPL 반환합니다.
DwmSetIconicLivePreviewBitmap 창 또는 탭의 라이브 미리 보기(미리 보기 미리 보기라고도 함)를 표시하도록 정적 아이콘 비트맵을 설정합니다. 작업 표시줄은 이 비트맵을 사용하여 창 또는 탭의 전체 크기 미리 보기를 표시할 수 있습니다.
DwmSetIconicThumbnail 창 또는 탭에서 축소판 그림 표현으로 사용할 정적 아이콘 비트맵을 설정합니다. 작업 표시줄은 이 비트맵을 창 또는 탭의 축소판 그림 스위치 대상으로 사용할 수 있습니다.
DwmSetPresentParameters 프레임 컴퍼지션에 대한 현재 매개 변수를 설정합니다. DwmSetPresentParameters는 더 이상 지원되지 않습니다. Windows 8.1 DwmSetPresentParameters에 대한 호출은 항상 E_NOTIMPL 반환합니다.
DwmSetWindowAttribute 창에 대한 DWM(데스크톱 창 관리자) 비 클라이언트 렌더링 특성의 값을 설정합니다.
DwmShowContact 특정 터치 또는 펜 접촉에 대한 응답으로 그릴 시각적 피드백 유형을 지정하기 위해 앱 또는 프레임워크에서 호출됩니다.
DwmTetherContact 터치 및 끌기 상호 작용의 그래픽 피드백을 사용자에게 사용하도록 설정합니다.

DwmTransitionOwnedWindow DWM(데스크톱 창 관리자)을 사용하여 도구 창의 애니메이션을 조정합니다.
DwmUnregisterThumbnail DwmRegisterThumbnail 함수에서 만든 DWM(데스크톱 창 관리자) 썸네일 관계를 제거합니다.
DwmUpdateThumbnailProperties DWM(데스크톱 창 관리자) 썸네일의 속성을 업데이트.

구조체

DWM_BLURBEHIND DWM(데스크톱 창 관리자) 흐림 효과 속성을 지정합니다. DwmEnableBlurBehindWindow 함수에서 사용됩니다.
DWM_PRESENT_PARAMETERS 프레임 컴퍼지션에 대한 DWM(데스크톱 창 관리자) 비디오 프레임 매개 변수를 지정합니다. DwmSetPresentParameters 함수에서 사용됩니다.
DWM_THUMBNAIL_PROPERTIES DWM(데스크톱 창 관리자) 썸네일 속성을 지정합니다. DwmUpdateThumbnailProperties 함수에서 사용됩니다.
DWM_TIMING_INFO DWM(데스크톱 창 관리자) 컴퍼지션 타이밍 정보를 지정합니다. DwmGetCompositionTimingInfo 함수에서 사용됩니다.
MilMatrix3x2D 변환을 설명하는 3x2 행렬을 지정합니다.
UNSIGNED_RATIO DWM(데스크톱 창 관리자) API에서 사용하는 데이터 형식을 정의합니다. 제네릭 비율을 나타내며 단일 API 내에서도 다양한 용도 및 단위로 사용됩니다.

열거형

DWM_SHOWCONTACT 시각적 피드백을 제공해야 하는 입력 작업을 지정합니다.
DWM_SOURCE_FRAME_SAMPLING DwmSetPresentParameters 함수에서 프레임 샘플링 유형을 지정하는 데 사용하는 플래그입니다.
DWM_SYSTEMBACKDROP_TYPE 클라이언트가 아닌 영역 뒤를 포함하여 창의 시스템 그리기 배경 자료를 지정하기 위한 플래그입니다.
DWM_TAB_WINDOW_REQUIREMENTS 이 열거형은 Windows에서 구현되지 않은 실험적 기능을 지원했습니다. 영향을 주지 않으며 사용하면 안 됩니다.
DWM_WINDOW_CORNER_PREFERENCE DWM_WINDOW_CORNER_PREFERENCE 열거형(dwmapi.h)은 창의 둥근 모서리 기본 설정을 지정합니다.
DWMFLIP3DWINDOWPOLICY DwmSetWindowAttribute 함수에서 Flip3D 창 정책을 지정하는 데 사용하는 플래그입니다.
DWMNCRENDERINGPOLICY DwmSetWindowAttribute 함수에서 비 클라이언트 영역 렌더링 정책을 지정하는 데 사용하는 플래그입니다.
DWMTRANSITION_OWNEDWINDOW_TARGET 대상을 식별합니다.
DWMWINDOWATTRIBUTE DwmGetWindowAttribute 및 DwmSetWindowAttribute 함수에서 사용하는 플래그입니다.
GESTURE_TYPE DwmRenderGesture에 지정된 제스처 유형을 식별합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dwrite.h 헤더

아티클2023. 08. 23.

이 헤더는 DirectWrite 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DirectWrite

dwrite.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDWriteBitmapRenderTarget 문자 모양을 렌더링하는 데 사용할 수 있는 32비트 디바이스 독립적 비트맵 및 디바이스 컨텍스트를 캡슐화합니다.
IDWriteFactory 모든 후속 DirectWrite 개체를 만드는 데 사용됩니다. 이 인터페이스는 모든 DirectWrite 개체에 대한 루트 팩터리 인터페이스입니다.
IDWriteFont 글꼴 컬렉션의 실제 글꼴을 나타냅니다. 이 인터페이스는 실제 글꼴에서 글꼴 얼굴을 만들거나 기존 글꼴 얼굴에서 글꼴 얼굴 메트릭 또는 얼굴 이름과 같은 정보를 검색하는 데 사용됩니다.
IDWriteFontCollection 시스템에 설치된 글꼴 집합 또는 특정 디렉터리의 글꼴 집합과 같은 글꼴 집합을 캡슐화하는 개체입니다. (IDWriteFontCollection)
IDWriteFontCollectionLoader 특정 유형의 키가 지정된 글꼴 컬렉션을 생성하는 데 사용됩니다.
IDWriteFontFace 이 인터페이스는 메트릭, 이름 및 문자 모양 윤곽선과 같은 다양한 글꼴 데이터를 노출합니다. 글꼴 얼굴 형식, 적절한 파일 참조 및 얼굴 식별 데이터가 포함되어 있습니다.
IDWriteFontFamily 관련 글꼴 패밀리를 나타냅니다. (IDWriteFontFamily)

IDWriteFontFile 글꼴 파일을 나타냅니다. 글꼴 관리자 또는 글꼴 뷰어와 같은 애플리케이션은 IDWriteFontFile::Analyze를 호출하여 특정 파일이 글꼴 파일인지 여부와 글꼴 시스템에서 지원하는 글꼴 형식인지 여부를 확인할 수 있습니다.
IDWriteFontFileEnumerator 글꼴 파일 컬렉션을 캡슐화합니다. 글꼴 시스템은 글꼴 컬렉션을 빌드할 때 이 인터페이스를 사용하여 글꼴 파일을 열거합니다.
IDWriteFontFileLoader 글꼴 파일 참조 키에서 글꼴 파일 스트림 개체로 특정 형식의 글꼴 파일 리소스 로드를 처리합니다.
IDWriteFontFileStream 사용자 지정 글꼴 파일 로더에서 글꼴 파일 데이터를 로드합니다.
IDWriteFontList 글꼴 목록을 나타냅니다. (IDWriteFontList)
IDWriteGdiInterop 글꼴 얼굴을 LOGFONT 구조로 변환하거나 GDI 글꼴 설명을 글꼴 면으로 변환하는 메서드와 같은 GDI와의 상호 운용성을 제공합니다. 비트맵 렌더링 대상 개체를 만드는 데도 사용됩니다. (IDWriteGdiInterop)
IDWriteGlyphRunAnalysis 문자 모양 실행을 렌더링하는 데 사용되는 하위 수준 정보를 포함합니다.
IDWriteInlineObject 애플리케이션 정의 인라인 그래픽을 래핑하여 DWrite가 그래픽이 텍스트와 인라인 문자 모양인 것처럼 메트릭을 쿼리할 수 있도록 합니다.
IDWriteLocalFontFileLoader 로컬 글꼴 파일에서 작동하고 글꼴 파일 참조 키에서 로컬 글꼴 파일 정보를 노출하는 IDWriteFontFileLoader 인터페이스의 기본 제공 구현입니다.
IDWriteLocalizedStrings 로캘 이름으로 인덱싱된 문자열의 컬렉션을 나타냅니다.
IDWriteNumberSubstitution

지정된 로캘에 대한 적절한 숫자와 숫자 문장 부호를 보유합니다.
IDWritePixelSnapping DIP당 픽셀(디바이스 독립적 픽셀) 및 텍스트 렌더러의 현재 변환 매트릭스와 같은 픽셀 스냅 속성을 정의합니다.
IDWriteRenderingParams 문자 모양 래스터화 및 필터링을 위한 ClearType 수준, 향상된 대비 및 감마 수정과 같은 텍스트 렌더링 설정을 나타냅니다.
IDWriteTextAnalysisSink 이 인터페이스는 지정된 텍스트 분석의 출력을 수신하기 위해 텍스트 분석기의 클라이언트에 의해 구현됩니다.
IDWriteTextAnalysisSource 분석기에서 텍스트를 제공하기 위해 텍스트 분석기 클라이언트에 의해 구현됩니다.
IDWriteTextAnalyzer 아랍어, 줄 바꿈 기회 결정, 문자 모양 배치 및 숫자 대체와 같은 언어에 대한 양방향(bidi) 지원과 같은 복잡한 스크립트 처리를 위한 다양한 텍스트 속성을 분석합니다.
IDWriteTextFormat IDWriteTextFormat 인터페이스는 텍스트 서식을 지정하는 데 사용되는 글꼴 및 단락 속성을 설명하고 로캘 정보를 설명합니다.
IDWriteTextLayout IDWriteTextLayout 인터페이스는 완전히 분석되고 서식이 지정된 후 텍스트 블록을 나타냅니다.
IDWriteTextRenderer 텍스트, 인라인 개체 및 밑줄과 같은 장식의 렌더링을 수행하는 애플리케이션 정의 콜백 집합을 나타냅니다. (IDWriteTextRenderer)
IDWriteTypography 글꼴 입력 체계 설정을 나타냅니다.

Functions

DWRITE_MAKE_OPENTYPE_TAG OpenType 태그를 32비트 정수로 만들어 태그의 첫 번째 문자가 글꼴 파일의 태그와 비교하는 데 사용할 수 있는 가장 낮은 바이트(작은 엔디안 아키텍처에서 가장 중요하지 않음)입니다.
DWriteCreateFactory 개별 DirectWrite 개체의 후속 생성에 사용되는 DirectWrite 팩터리 개체를 만듭니다.

구조체

DWRITE_CLUSTER_METRICS 문자 모양 클러스터에 대한 정보를 포함합니다.
DWRITE_FONT_FEATURE 현재 글꼴 면에서 입력 체계 기능을 식별하고 실행하는 데 사용되는 속성을 지정합니다.
DWRITE_FONT_METRICS DWRITE_FONT_METRICS 구조체는 글꼴 면 내의 모든 문자 모양에 적용할 수 있는 메트릭을 지정합니다.
DWRITE_GLYPH_METRICS 개별 문자 모양의 메트릭을 지정합니다.
DWRITE_GLYPH_OFFSET 문자 모양 위치에 대한 선택적 조정입니다.
DWRITE_GLYPH_RUN 렌더러가 문자 모양 실행을 그리는 데 필요한 정보를 포함합니다.
DWRITE_GLYPH_RUN_DESCRIPTION DWRITE_GLYPH_RUN 속성과 관련된 추가 속성을 포함합니다.
DWRITE_HIT_TEST_METRICS 적중 테스트에서 얻은 지역을 설명합니다.

DWRITE_INLINE_OBJECT_METRICS 애플리케이션 정의 인라인 개체의 기하학적 측정을 설명하는 속성을 포함합니다.
DWRITE_LINE_BREAKPOINT 문자의 줄 중단점 특성입니다.
DWRITE_LINE_METRICS 서식이 지정된 텍스트 줄에 대한 정보를 포함합니다. (DWRITE_LINE_METRICS)
DWRITE_MATRIX DWRITE_MATRIX 구조체는 렌더링된 문자 모양에 적용할 그래픽 변환을 지정합니다.
DWRITE_OVERHANG_METRICS 표시되는 DIP(디바이스 독립적 픽셀)가 레이아웃 또는 인라인 개체의 각 측면을 오버슈팅하는 정도를 나타냅니다.
DWRITE_SCRIPT_ANALYSIS 텍스트와 해당 쓰기 시스템 스크립트의 연결과 일부 표시 특성을 저장합니다.
DWRITE_SHAPING_GLYPH_PROPERTIES 출력 문자 모양에 대한 셰이핑 출력 속성을 포함합니다.
DWRITE_SHAPING_TEXT_PROPERTIES 출력 문자 모양에 대한 출력 속성 셰이핑
DWRITE_STRIKETHROUGH 취소선의 크기 및 배치에 대한 정보를 포함합니다.
DWRITE_TEXT_METRICS 레이아웃 후 텍스트와 연결된 메트릭을 포함합니다. (DWRITE_TEXT_METRICS)
DWRITE_TEXT_RANGE IDWriteTextLayout 개체가 나타내는 텍스트에 서식이 적용되는 텍스트 위치의 범위를 지정합니다.
DWRITE_TRIMMING 레이아웃 상자를 오버플로하는 텍스트에 대한 트리밍 옵션을 지정합니다.

DWRITE_TYPOGRAPHIC_FEATURES 텍스트 셰이핑 중에 적용할 입력 기능 집합을 포함합니다.
DWRITE_UNDERLINE 밑줄의 너비, 두께, 오프셋, 실행 높이, 읽기 방향 및 흐름 방향에 대한 정보를 포함합니다.

열거형

DWRITE_BREAK_CONDITION 줄 바꿈 동작을 결정하는 데 사용되는 인라인 개체 또는 텍스트 가장자리의 조건을 나타냅니다.
DWRITE_FACTORY_TYPE DirectWrite 팩터리 개체의 형식을 지정합니다.
DWRITE_FLOW_DIRECTION 텍스트 줄이 서로 상대적으로 배치되는 방식의 방향을 나타냅니다.
DWRITE_FONT_FACE_TYPE 전체 글꼴 면의 파일 형식을 나타냅니다.
DWRITE_FONT_FEATURE_TAG 글꼴에서 제공하는 텍스트의 입력 기능을 나타내는 값입니다.
DWRITE_FONT_FILE_TYPE 단일 글꼴 파일로 표시되는 글꼴의 형식입니다. 여러 파일(예: 형식 1)로 구성된 글꼴 형식입니다. PFM 및 . PFB에는 각 파일 형식에 대해 별도의 열거형 값이 있습니다.
DWRITE_FONT_SIMULATIONS 글꼴 면에 적용할 알고리즘 스타일 시뮬레이션을 지정합니다. 굵게 및 비스듬한 시뮬레이션은 비트 OR 연산을 통해 결합할 수 있습니다.
DWRITE_FONT_STRETCH 글꼴의 일반 가로 세로 비율에 비해 글꼴이 확장된 정도를 나타냅니다.

DWRITE_FONT_STYLE 글꼴 얼굴의 스타일을 보통, 기울임꼴 또는 경사로 나타냅니다.
DWRITE_FONT_WEIGHT 스트로크의 가벼움 또는 무거움 측면에서 서체의 밀도를 나타냅니다.
DWRITE_INFORMATIONAL_STRING_ID 글꼴 파일에 포함된 문자열을 식별하는 정보 문자열 열거형입니다.
DWRITE_LINE_SPACING_METHOD 텍스트 레이아웃의 줄 간격에 사용되는 메서드입니다.
DWRITE_NUMBER_SUBSTITUTION_METHOD 숫자 및 관련 문장 부호에 숫자 대체를 적용하는 방법을 지정합니다.
DWRITE_PARAGRAPH_ALIGNMENT 흐름 레이아웃 상자의 위쪽과 아래쪽을 기준으로 흐름 방향 축을 따라 단락 텍스트의 맞춤을 지정합니다.
DWRITE_PIXEL_GEOMETRY 텍스트 렌더링을 위해 가정되는 디바이스 픽셀(즉, 빨간색, 녹색 및 파란색 구성 요소의 물리적 배열)의 내부 구조를 나타냅니다.
DWRITE_READING_DIRECTION 읽기가 진행되는 방향을 지정합니다.
DWRITE_RENDERING_MODE 문자 모양을 렌더링하는 메서드를 나타냅니다.
DWRITE_SCRIPT_SHAPES 텍스트에 대한 추가 셰이핑 요구 사항을 나타냅니다.
DWRITE_TEXT_ALIGNMENT 레이아웃 상자의 선행 및 후행 가장자리를 기준으로 읽기 방향 축을 따라 단락 텍스트의 맞춤을 지정합니다.
DWRITE_TEXTURE_TYPE

알파 텍스처의 형식을 식별합니다.
DWRITE_TRIMMING_GRANULARITY 레이아웃 상자를 오버플로하는 텍스트를 자르는 데 사용되는 텍스트 세분성을 지정합니다.
DWRITE_WORD_WRAPPING 특정 여러 줄 단락에서 사용할 단어 줄 바꿈을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dwrite_1.h 헤더

아티클2023. 08. 23.

이 헤더는 DirectWrite 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DirectWrite

dwrite_1.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDWriteBitmapRenderTarget1 문자 모양을 렌더링하는 데 사용할 수 있는 32비트 디바이스 독립적 비트맵 및 디바이스 컨텍스트를 캡슐화합니다.
IDWriteFactory1 모든 DirectWrite 개체에 대한 루트 팩터리 인터페이스입니다. (IDWriteFactory1)
IDWriteFont1 글꼴 컬렉션의 실제 글꼴을 나타냅니다. (IDWriteFont1)
IDWriteFontFace1 글꼴 얼굴 유형, 적절한 파일 참조 및 얼굴 식별 데이터를 포함합니다. (IDWriteFontFace1)
IDWriteRenderingParams1 문자 모양 래스터화 및 필터링에 대한 텍스트 렌더링 설정을 나타냅니다. (IDWriteRenderingParams1)
IDWriteTextAnalysisSink1 텍스트 분석기의 출력을 수신하기 위해 구현하는 인터페이스입니다.
IDWriteTextAnalysisSource1 텍스트 및 연결된 텍스트 속성과 같은 텍스트 분석기에서 필요한 정보를 제공하기 위해 구현하는 인터페이스입니다.
IDWriteTextAnalyzer1 복잡한 스크립트 처리를 위해 다양한 텍스트 속성을 분석합니다. (IDWriteTextAnalyzer1)

IDWriteTextLayout1 완전히 분석되고 서식이 지정된 후의 텍스트 블록을 나타냅니다. (IDWriteTextLayout1)

구조체

DWRITE_CARET_METRICS DWRITE_CARET_METRICS 구조체는 글꼴에서 caret 배치에 대한 메트릭을 지정합니다.
DWRITE_FONT_METRICS1 DWRITE_FONT_METRICS1 구조체는 글꼴 면 내의 모든 문자 모양에 적용할 수 있는 메트릭을 지정합니다.
DWRITE_JUSTIFICATION_OPPORTUNITY DWRITE_JUSTIFICATION_OPPORTUNITY 구조체는 문자 모양당 근거 정보를 지정합니다.
DWRITE_PANOSE DWRITE_PANOSE 공용 구조체는 IDWriteFont1::GetPanose에서 글꼴을 선택하고 일치시키는 데 사용하는 서체 분류 값을 설명합니다.
DWRITE_SCRIPT_PROPERTIES DWRITE_SCRIPT_PROPERTIES 구조는 캐리트 탐색 및 근거에 대한 스크립트 속성을 지정합니다.
DWRITE_UNICODE_RANGE DWRITE_UNICODE_RANGE 구조체는 유니코드 코드 포인트의 범위를 지정합니다.

열거형

DWRITE_BASELINE DWRITE_BASELINE 열거형에는 텍스트 맞춤 기준을 지정하는 값이 포함됩니다.
DWRITE_GLYPH_ORIENTATION_ANGLE

DWRITE_GLYPH_ORIENTATION_ANGLE 열거형에는 문자 모양이 x축을 지향하는 방법을 지정하는 값이 포함됩니다.
DWRITE_OUTLINE_THRESHOLD DWRITE_OUTLINE_THRESHOLD 열거형에는 개요 모드에서 문자 모양을 렌더링할지 여부를 결정하기 위해 IDWriteFontFace1::GetRecommendedRenderingMode 메서드에서 사용하는 정책을 지정하는 값이 포함되어 있습니다.
DWRITE_PANOSE_ARM_STYLE DWRITE_PANOSE_ARM_STYLE 열거형에는 텍스트에 대한 줄기 및 둥근 문자 형식의 종료 스타일을 지정하는 값이 포함되어 있습니다.
DWRITE_PANOSE_ASPECT DWRITE_PANOSE_ASPECT 열거형에는 문자 면의 너비와 높이 사이의 비율을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_ASPECT_RATIO DWRITE_PANOSE_ASPECT_RATIO 열거형에는 문자 면의 너비와 높이 사이의 비율에 대한 정보를 지정하는 값이 포함됩니다.
DWRITE_PANOSE_CHARACTER_RANGES DWRITE_PANOSE_CHARACTER_RANGES 열거형에는 글꼴에서 사용할 수 있는 문자 유형을 지정하는 값이 포함되어 있습니다.
DWRITE_PANOSE_CONTRAST DWRITE_PANOSE_CONTRAST 열거형에는 대문자 'O'와 같은 문자에 대해 스트로크의 가장 두꺼운 점과 가장 얇은 점 사이의 비율을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_DECORATIVE_CLASS DWRITE_PANOSE_DECORATIVE_CLASS 열거형에는 문자 얼굴의 일반적인 모양을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_DECORATIVE_TOPOLOGY DWRITE_PANOSE_DECORATIVE_TOPOLOGY 열거형에는 글꼴의 전체 도형 특성을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_FAMILY DWRITE_PANOSE_FAMILY 열거형에는 서체 분류의 종류를 지정하는 값이 포함됩니다.
DWRITE_PANOSE_FILL

DWRITE_PANOSE_FILL 열거형에는 채우기 및 선 처리 유형을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_FINIALS DWRITE_PANOSE_FINIALS 열거형에는 문자 끝과 소수 오름차순 처리 방법을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_LETTERFORM DWRITE_PANOSE_LETTERFORM 열거형에는 텍스트에 대한 문자 폼의 원형을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_LINING DWRITE_PANOSE_LINING 열거형에는 장식 서체에 대한 윤곽선 처리를 지정하는 값이 포함되어 있습니다.
DWRITE_PANOSE_MIDLINE DWRITE_PANOSE_MIDLINE 열거형에는 대문자 간에 중간선 배치 및 대각선 줄기 정점 처리에 대한 정보를 지정하는 값이 포함되어 있습니다.
DWRITE_PANOSE_PROPORTION DWRITE_PANOSE_PROPORTION 열거형에는 표준 문자에 대한 추가 세부 정보를 고려하여 문자 모양 비율을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_SCRIPT_FORM DWRITE_PANOSE_SCRIPT_FORM 열거형에는 기울기와 꼬리를 고려하여 문자 얼굴의 일반적인 모양을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_SCRIPT_TOPOLOGY DWRITE_PANOSE_SCRIPT_TOPOLOGY 열거형에는 문자 폼의 토폴로지를 지정하는 값이 포함됩니다.
DWRITE_PANOSE_SERIF_STYLE DWRITE_PANOSE_SERIF_STYLE 열거형에는 serif 텍스트의 모양을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_SPACING DWRITE_PANOSE_SPACING 열거형에는 문자 간격(모노스페이스 대 비례)을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_STROKE_VARIATION

DWRITE_PANOSE_STROKE_VARIATION 열거형에는 텍스트 문자의 얇고 굵은 줄기 사이의 관계를 지정하는 값이 포함됩니다.
DWRITE_PANOSE_SYMBOL_ASPECT_RATIO DWRITE_PANOSE_SYMBOL_ASPECT_RATIO 열거형에는 기호 문자의 가로 세로 비율을 지정하는 값이 포함됩니다.
DWRITE_PANOSE_SYMBOL_KIND DWRITE_PANOSE_SYMBOL_KIND 열거형에는 기호 집합의 종류를 지정하는 값이 포함됩니다.
DWRITE_PANOSE_TOOL_KIND DWRITE_PANOSE_TOOL_KIND 열거형에는 문자 양식을 만드는 데 사용되는 도구의 종류를 지정하는 값이 포함되어 있습니다.
DWRITE_PANOSE_WEIGHT DWRITE_PANOSE_WEIGHT 열거형에는 문자의 가중치를 지정하는 값이 포함됩니다.
DWRITE_PANOSE_XASCENT DWRITE_PANOSE_XASCENT 열거형에는 소문자의 상대 크기를 지정하는 값이 포함됩니다.
DWRITE_PANOSE_XHEIGHT DWRITE_PANOSE_XHEIGHT 열거형에는 소문자의 상대적 크기와 분음 부호(xheight)의 처리에 대한 정보를 지정하는 값이 포함되어 있습니다.
DWRITE_TEXT_ANTIALIAS_MODE DWRITE_TEXT_ANTIALIAS_MODE 열거형에는 렌더링 모드에서 앤티앨리어싱을 호출할 때 텍스트에 사용할 앤티앨리어싱 유형을 지정하는 값이 포함됩니다.
DWRITE_VERTICAL_GLYPH_ORIENTATION DWRITE_VERTICAL_GLYPH_ORIENTATION 열거형에는 텍스트에 원하는 종류의 문자 모양 방향을 지정하는 값이 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dwrite_2.h 헤더

아티클2023. 08. 23.

이 헤더는 DirectWrite 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DirectWrite

dwrite_2.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDWriteColorGlyphRunEnumerator 이 인터페이스를 사용하면 애플리케이션이 색 문자 모양 실행을 통해 열거할 수 있습니다.
IDWriteFactory2 모든 DirectWrite 개체에 대한 루트 팩터리 인터페이스입니다. 나
IDWriteFont2 글꼴 컬렉션의 실제 글꼴을 나타냅니다. (IDWriteFont2)
IDWriteFontFace2 글꼴 얼굴 유형, 적절한 파일 참조 및 얼굴 식별 데이터를 포함합니다. (IDWriteFontFace2)
IDWriteFontFallback 글꼴 목록에서 대체 글꼴에 액세스할 수 있습니다.
IDWriteFontFallbackBuilder 유니코드 글꼴 대체 매핑을 만들고 해당 매핑에서 글꼴 대체 개체를 만들 수 있습니다.
IDWriteRenderingParams2 문자 모양 래스터화 및 필터링에 대한 텍스트 렌더링 설정을 나타냅니다. (IDWriteRenderingParams2)
IDWriteTextAnalyzer2 복잡한 스크립트 처리를 위해 다양한 텍스트 속성을 분석합니다. (IDWriteTextAnalyzer2)
IDWriteTextFormat1

텍스트 서식을 지정하는 데 사용되는 글꼴 및 단락 속성을 설명하고 로캘 정보를 설명합니다. (IDWriteTextFormat1)
IDWriteTextLayout2 완전히 분석되고 서식이 지정된 후의 텍스트 블록을 나타냅니다. (IDWriteTextLayout2)
IDWriteTextRenderer1 텍스트, 인라인 개체 및 밑줄과 같은 장식의 렌더링을 수행하는 애플리케이션 정의 콜백 집합을 나타냅니다. (IDWriteTextRenderer1)

구조체

DWRITE_COLOR_GLYPH_RUN 문자 모양 색 정보를 사용하여 문자 모양 실행을 그리는 데 렌더러가 필요한 정보를 포함합니다.
DWRITE_TEXT_METRICS1 레이아웃 후 텍스트와 연결된 메트릭을 포함합니다. (DWRITE_TEXT_METRICS1)

열거형

DWRITE_GRID_FIT_MODE 문자 모양 윤곽선의 그리드 맞춤을 사용할지 여부를 지정합니다(힌트라고도 함).
DWRITE_OPTICAL_ALIGNMENT 광학 여백 맞춤 모드입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dwrite_3.h 헤더

이 헤더는 DirectWrite 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DirectWrite

dwrite_3.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

테이블 확장

IDWriteAsyncResult 비동기 작업의 결과를 나타냅니다. 클라이언트는 인터페이스를 사용하여 작업이 완료되기를 기다렸다가 결과를 가져올 수 있습니다.
IDWriteColorGlyphRunEnumerator1 정렬된 색 문자 모양 실행 컬렉션에 대한 열거자입니다.
IDWriteFactory3 모든 DirectWrite 개체에 대한 루트 팩터리 인터페이스입니다. (IDWriteFactory3)
IDWriteFactory4 모든 DirectWrite 개체에 대한 루트 팩터리 인터페이스입니다. (IDWriteFactory4)
IDWriteFactory5 모든 DirectWrite 개체에 대한 루트 팩터리 인터페이스입니다. (IDWriteFactory5)
IDWriteFactory6 이 인터페이스는 모든 DirectWrite 개체가 만들어지는 팩터리 개체를 나타냅니다. **IDWriteFactory6** 은 글꼴 및 글꼴 리소스 작업을 위한 새로운 기능을 추가합니다.
IDWriteFactory7 이 인터페이스는 모든 DirectWrite 개체가 만들어지는 팩터리 개체를 나타냅니다. **IDWriteFactory7** 은 시스템 글꼴 작업을 위한 새로운 기능을 추가합니다.
IDWriteFont3 글꼴 컬렉션의 글꼴을 나타냅니다.

IDWriteFontCollection1 시스템에 설치된 글꼴 집합 또는 특정 디렉터리의 글꼴 집합과 같은 글꼴 집합을 캡슐화하는 개체입니다. (IDWriteFontCollection1)
IDWriteFontCollection2 이 인터페이스는 시스템에 설치된 글꼴 집합 또는 특정 디렉터리의 글꼴 집합과 같은 글꼴 집합을 캡슐화합니다. 글꼴 컬렉션 API를 사용하여 사용할 수 있는 글꼴 패밀리 및 글꼴을 검색하고 글꼴에 대한 일부 메타데이터를 가져올 수 있습니다. (IDWriteFontCollection2)
IDWriteFontCollection3 이 인터페이스는 시스템에 설치된 글꼴 집합 또는 특정 디렉터리의 글꼴 집합과 같은 글꼴 집합을 캡슐화합니다. 글꼴 컬렉션 API를 사용하여 사용할 수 있는 글꼴 패밀리 및 글꼴을 검색하고 글꼴에 대한 일부 메타데이터를 가져올 수 있습니다. (IDWriteFontCollection3)
IDWriteFontDownloadListener 글꼴 다운로드 큐에서 알림을 받는 애플리케이션 정의 콜백 인터페이스(IDWriteFontDownloadQueue 인터페이스).
IDWriteFontDownloadQueue 원격 글꼴, 문자, 문자 모양 및 글꼴 조각에 대한 다운로드 요청을 큐에 넣은 인터페이스입니다.
IDWriteFontFace3 글꼴 얼굴 유형, 적절한 파일 참조 및 얼굴 식별 데이터를 포함합니다. (IDWriteFontFace3)
IDWriteFontFace4 글꼴 얼굴 유형, 적절한 파일 참조 및 얼굴 식별 데이터를 포함합니다. (IDWriteFontFace4)
IDWriteFontFace5 글꼴 얼굴 유형, 적절한 파일 참조 및 얼굴 식별 데이터를 포함합니다. (IDWriteFontFace5)
IDWriteFontFaceReference 글꼴 면에 대한 참조를 나타냅니다.
IDWriteFontFaceReference1 글꼴 면에 대한 참조를 나타냅니다. 글꼴에 대한 고유 식별 참조로, 글꼴 얼굴을 만들어 글꼴 메트릭을 쿼리하고 렌더링에 사용할 수 있습니다.
IDWriteFontFallback1

글꼴 목록에서 대체 글꼴에 액세스할 수 있습니다.
IDWriteFontFamily1 관련 글꼴 패밀리를 나타냅니다. (IDWriteFontFamily1)
IDWriteFontFamily2 관련 글꼴 패밀리를 나타냅니다. **IDWriteFontFamily2** 는 글꼴 축 값으로 글꼴 검색을 비롯한 새로운 기능을 추가합니다.
IDWriteFontList1 글꼴 목록을 나타냅니다. (IDWriteFontList1)
IDWriteFontList2 글꼴 목록을 나타냅니다. **IDWriteFontList2** 는 목록에서 사용하는 기본 글꼴 집합 검색을 포함하여 새 기능을 추가합니다.
IDWriteFontResource 글꼴 리소스에 대한 축 정보를 제공하며 특정 글꼴 얼굴 인스턴스를 만드는 데 사용됩니다.
IDWriteFontSet 글꼴 집합을 나타냅니다. (IDWriteFontSet)
IDWriteFontSet1 글꼴 집합을 나타냅니다. (IDWriteFontSet1)
IDWriteFontSet2 글꼴 집합을 나타냅니다. (IDWriteFontSet2)
IDWriteFontSet3 글꼴 집합을 나타냅니다. (IDWriteFontSet3)
IDWriteFontSet4 글꼴 집합을 나타냅니다. (IDWriteFontSet4)
IDWriteFontSetBuilder 글꼴 집합을 빌드하는 메서드를 포함합니다. (IDWriteFontSetBuilder)
IDWriteFontSetBuilder1

글꼴 집합을 빌드하는 메서드를 포함합니다. (IDWriteFontSetBuilder1)
IDWriteFontSetBuilder2 글꼴 집합을 빌드하는 메서드를 포함합니다. (IDWriteFontSetBuilder2)
IDWriteGdiInterop1 글꼴 얼굴을 LOGFONT 구조로 변환하거나 GDI 글꼴 설명을 글꼴 면으로 변환하는 메서드와 같은 GDI와의 상호 운용성을 제공합니다. 비트맵 렌더링 대상 개체를 만드는 데도 사용됩니다. (IDWriteGdiInterop1)
IDWriteInMemoryFontFileLoader 메모리 내 글꼴에 액세스할 수 있는 글꼴 파일 로더를 나타냅니다.
IDWriteRemoteFontFileLoader 원격(즉, 다운로드 가능한) 글꼴에 액세스할 수 있는 글꼴 파일 로더를 나타냅니다.
IDWriteRemoteFontFileStream 글꼴 파일 스트림을 나타내며, 그 중 일부는 로컬이 아닐 수 있습니다.
IDWriteRenderingParams3 문자 모양 래스터화 및 필터링에 대한 텍스트 렌더링 설정을 나타냅니다. (IDWriteRenderingParams3)
IDWriteStringList 숫자로 인덱싱된 문자열의 컬렉션을 나타냅니다.
IDWriteTextFormat2 텍스트 서식을 지정하는 데 사용되는 글꼴 및 단락 속성을 설명하고 로캘 정보를 설명합니다. (IDWriteTextFormat2)
IDWriteTextFormat3 텍스트 서식을 지정하는 데 사용되는 글꼴 및 단락 속성을 설명하고 로캘 정보를 설명합니다. (IDWriteTextFormat3)
IDWriteTextLayout3 완전히 분석되고 서식이 지정된 후의 텍스트 블록을 나타냅니다. (IDWriteTextLayout3)
IDWriteTextLayout4 완전히 분석되고 서식이 지정된 후의 텍스트 블록을 나타냅니다.

Functions

테이블 확장

DWRITE_MAKE_FONT_AXIS_TAG 글꼴 축에 대한 OpenType 태그를 만듭니다.

구조체

테이블 확장

DWRITE_COLOR_GLYPH_RUN1 색 문자 모양 실행을 나타냅니다. IDWriteFactory4::TranslateColorGlyphRun 메서드는 글꼴이 지원하는 항목에 따라 다양한 형식의 순서가 지정된 색 문자 모양 실행 컬렉션을 반환합니다.
DWRITE_FILE_FRAGMENT 글꼴 파일의 바이트 범위를 나타냅니다.
DWRITE_FONT_AXIS_RANGE 글꼴 축에 대해 가능한 값의 최소 및 최대 범위를 나타냅니다.
DWRITE_FONT_AXIS_VALUE 글꼴 축의 값을 나타냅니다. 글꼴 인스턴스를 쿼리하고 만들 때 사용됩니다.
DWRITE_FONT_PROPERTY 글꼴 집합을 필터링하고 명시적 속성을 사용하여 글꼴 집합을 빌드하는 데 사용되는 Font 속성입니다.
DWRITE_GLYPH_IMAGE_DATA GetGlyphImageData의 단일 문자 모양에 대한 데이터입니다.
DWRITE_LINE_METRICS1 서식이 지정된 텍스트 줄에 대한 정보를 포함합니다. (DWRITE_LINE_METRICS1)
DWRITE_LINE_SPACING

텍스트 줄 사이의 세로 간격을 설정합니다.

열거형

테이블 확장

DWRITE_AUTOMATIC_FONT_AXES 글꼴을 선택하는 동안 레이아웃에 자동으로 적용할 수 있는 특정 축을 지정하는 상수를 정의합니다.
DWRITE_CONTAINER_TYPE 글꼴 리소스의 컨테이너 형식을 지정합니다. 컨테이너는 기본 글꼴 파일이 패키지되는 컨테이너를 설명하므로 컨테이너 형식은 글꼴 파일 형식(DWRITE_FONT_FILE_TYPE)과 다릅니다.
DWRITE_FONT_AXIS_ATTRIBUTES 글꼴 축의 특성을 지정하는 상수를 정의합니다.
DWRITE_FONT_AXIS_TAG 글꼴 축에 4자 식별자를 지정하는 상수를 정의합니다.
DWRITE_FONT_FAMILY_MODEL 글꼴 패밀리를 함께 그룹화하는 방법을 지정하는 상수를 정의합니다.
DWRITE_FONT_LINE_GAP_USAGE DWRITE_FONT_METRICS::lineGap 값이 줄 메트릭의 일부여야 하는지 여부를 지정합니다.
DWRITE_FONT_PROPERTY_ID 글꼴에서 문자열을 식별합니다.
DWRITE_FONT_SOURCE_TYPE 글꼴 집합에 글꼴이 포함되는 메커니즘을 지정하는 상수를 정의합니다.
DWRITE_LOCALITY 리소스의 위치를 지정합니다.
DWRITE_RENDERING_MODE1

문자 모양을 렌더링하는 방법을 지정합니다.

Last updated on 2025. 09. 10.

dxcapi.h 헤더

아티클2024. 01. 10.

이 헤더는 HLSL에 대한 참조에서 사용됩니다. 자세한 내용은 다음의 정보를 참조하세요.

- HLSL에 대한 참조

dxcapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

DxcCreateInstance 지정된 CLSID와 연결된 클래스의 초기화되지 않은 단일 개체를 만듭니다.
DxcCreateInstance2 지정된 CLSID와 연결된 클래스의 초기화되지 않은 단일 개체를 만듭니다(사용자 지정 메모리 할당자를 사용하여 컴파일러의 instance 만드는 데 사용할 수 있습니다).

콜백 함수

테이블 확장

DxcCreateInstance2Proc **DxcCreateInstance2Proc** typedef는 **GetProcAddress**를 사용하여 DxcCreateInstance2를 호출하는 데 더 편리하게 사용할 수 있도록 제공됩니다.
DxcCreateInstanceProc **DxcCreateInstanceProc** typedef는 **GetProcAddress**를 사용하여 DxcCreateInstance를 호출하는 데 더 편리하게 사용할 수 있도록 제공됩니다.

구조체

테이블 확장

DxcArgPair TBD
DxcDefine Dxc API에 대한 정의를 제공하기 위한 구조체입니다.
DxcShaderHash ShaderHash의 해시 다이제스트 형식입니다.
DxcText Dxc API에 바이트 또는 텍스트 입력을 제공하는 구조입니다.
IDxcAssembler DxcAssembler를 나타내는 인터페이스입니다.
IDxcBlob DXC API로 전달 및 외부로 전달할 수 있는 크기의 버퍼입니다. **IDxcBlob** 은 **ID3D10Blob** 및 **ID3DBlob**의 별칭입니다.
IDxcBlobEncoding 알려진 인코딩이 있을 수 있는 Blob입니다.
IDxcBlobUtf8 TBD
IDxcCompiler IDxcCompiler는 더 이상 사용되지 않습니다. 대신 IDxcCompiler3 을 사용합니다.
IDxcCompiler2 IDxcCompiler2는 더 이상 사용되지 않습니다. 대신 IDxcCompiler3 을 사용합니다.
IDxcCompiler3 DirectX 셰이더 컴파일러를 나타내는 인터페이스입니다.
IDxcCompilerArgs

DXC에 전달된 인수를 관리하기 위한 인터페이스입니다.
IDxcContainerBuilder DXC 컨테이너 작성기를 나타내는 인터페이스입니다.
IDxcContainerReflection DxcContainerReflection을 나타내는 인터페이스입니다.
IDxcExtraOutputs DXC 작업의 추가 출력을 나타냅니다.
IDxcIncludeHandler 포함 지시문을 처리하기 위한 인터페이스입니다. include 지시문의 처리를 사용자 지정하려면 이 인터페이스의 구현을 제공할 수 있습니다.
IDxcLibrary IDxcUtils는 IDxcLibrary를 대체합니다. 대신 IDxcUtils를 사용합니다.
IDxcLinker DXC 링커 인터페이스입니다.
IDxcOperationResult DXC 작업의 결과입니다. IDxcResult는 IDxcOperationResult를 대체하며 가능한 경우 사용해야 합니다.
IDxcOptimizer DxcOptimizer를 나타내는 인터페이스입니다.
IDxcOptimizerPass 최적화 프로그램 패스를 나타내는 인터페이스입니다.
IDxcPdbUtils PDB 버전 정보를 나타냅니다.
IDxcResult DXC 작업의 결과를 나타냅니다. DXC 작업에는 셰이더 개체 및 오류와 같은 여러 출력이 있을 수 있습니다. 이 인터페이스는 출력에 대한 액세스를 제공합니다.

IDxcUtils DXC에 대한 다양한 유틸리티 함수입니다.
IDxcValidator DXC 셰이더 유효성 검사기를 나타내는 인터페이스입니다.
IDxcValidator2 DXC 셰이더 유효성 검사기를 나타내는 인터페이스입니다.
IDxcVersionInfo PDB 버전 정보를 나타냅니다.
IDxcVersionInfo2 PDB 버전 정보를 나타냅니다.
IDxcVersionInfo3 PDB 버전 정보를 나타냅니다.

열거형

테이블 확장

DXC_OUT_KIND IDxcResult에서 검색할 출력의 종류를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

dxcore.h 헤더

아티클2023. 08. 23.

이 헤더는 DXCore에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DXCore

dxcore.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DXCoreCreateAdapterFactory 추가 DXCore 개체를 생성하는 데 사용할 수 있는 DXCore 어댑터 팩터리를 만듭니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dxcore_interface.h 헤더

아티클2024. 11. 08.

이 헤더는 DXCore에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DXCore

dxcore_interface.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

콜백 함수

테이블 확장

PFN_DXCORE_NOTIFICATION_CALLBACK 알림 이벤트에 대해 DXCore 개체에서 호출하는 콜백 함수(애플리케이션에서 구현)입니다.

구조

테이블 확장

DXCoreAdapterEngineIndex
DXCoreAdapterMemoryBudget 어댑터의 메모리 예산을 설명합니다.
DXCoreAdapterMemoryBudgetNodeSegmentGroup 어댑터의 메모리 세그먼트 그룹에 대해 설명합니다.
DXCoreAdapterProcessSetQueryInput
DXCoreAdapterProcessSetQueryOutput
DXCoreEngineNamePropertyInput

DXCoreEngineNamePropertyOutput
DXCoreEngineQueryInput
DXCoreEngineQueryOutput
DXCoreFrequencyQueryOutput
DXCoreHardwareID 어댑터의 PnP 하드웨어 ID 부분을 나타냅니다.
DXCoreHardwareIDParts
DXCoreMemoryQueryInput
DXCoreMemoryUsage
DXCoreProcessMemoryQueryInput
DXCoreProcessMemoryQueryOutput

열거형

테이블 확장

DXCoreAdapterPreference 목록 정렬 조건으로 사용할 DXCore 어댑터 기본 설정을 지정하는 상수를 정의합니다.
DXCoreAdapterProperty DXCore 어댑터 속성을 지정하는 상수를 정의합니다.
DXCoreAdapterState

DXCore 어댑터 상태의 종류를 지정하는 상수를 정의합니다.
DXCoreHardwareTypeFilterFlags 어댑터를 열거할 때 지정된 하드웨어 유형을 지정하는 상수를 정의합니다.
DXCoreMemoryType
DXCoreNotificationType IDXCoreAdapter 또는 IDXCoreAdapterList 개체에서 발생하는 알림 유형을 지정하는 상수를 정의합니다.
DXCoreRuntimeFilterFlags
DXCoreSegmentGroup 어댑터의 메모리 세그먼트 그룹화를 지정하는 상수를 정의합니다.
DXCoreWorkload

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

dxgi.h 헤더

아티클2023. 08. 23.

이 헤더는 DXGI에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DXGI

dxgi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDXGIAdapter IDXGIAdapter 인터페이스는 디스플레이 하위 시스템(하나 이상의 GPU, DAC 및 비디오 메모리 포함)을 나타냅니다.
IDXGIAdapter1 IDXGIAdapter1 인터페이스는 디스플레이 하위 시스템(하나 이상의 GPU, DAC 및 비디오 메모리 포함)을 나타냅니다.
IDXGIDevice IDXGIDevice 인터페이스는 이미지 데이터를 생성하는 DXGI 개체에 대한 파생 클래스를 구현합니다.
IDXGIDevice1 IDXGIDevice1 인터페이스는 이미지 데이터를 생성하는 DXGI 개체에 대한 파생 클래스를 구현합니다.
IDXGIDeviceSubObject 디바이스에 연결된 개체에서 상속되어 포인터를 검색할 수 있습니다.
IDXGIFactory IDXGIFactory 인터페이스는 전체 화면 전환을 처리하는 DXGI 개체를 생성하는 메서드를 구현합니다.
IDXGIFactory1 IDXGIFactory1 인터페이스는 DXGI 개체를 생성하기 위한 메서드를 구현합니다.

IDXGIKeyedMutex 여러 디바이스에서 사용하는 공유 리소스에 대한 단독 액세스를 허용하는 키 지정된 뮤텍스를 나타냅니다.
IDXGIObject IDXGIObject 인터페이스는 모든 DXGI 개체에 대한 기본 인터페이스입니다. IDXGIObject는 호출자 정의(프라이빗 데이터)를 개체와 연결하고 인터페이스를 부모 개체에 검색할 수 있도록 지원합니다.
IDXGIOutput IDXGIOutput 인터페이스는 어댑터 출력(예: 모니터)을 나타냅니다.
IDXGIResource IDXGIResource 인터페이스는 리소스 공유를 허용하고 리소스가 상주하는 메모리를 식별합니다.
IDXGISurface IDXGISurface 인터페이스는 이미지 데이터 개체에 대한 메서드를 구현합니다.
IDXGISurface1 IDXGISurface1 인터페이스는 Windows GDI(그래픽 디바이스 인터페이스)를 사용하여 Microsoft DXGI(DirectX Graphics Infrastructure) 화면에 렌더링하는 지원을 추가하여 IDXGISurface를 확장합니다.
IDXGISwapChain IDXGISwapChain 인터페이스는 렌더링된 데이터를 출력에 표시하기 전에 저장하기 위해 하나 이상의 표면을 구현합니다.

Functions

CreateDXGIFactory 다른 DXGI 개체를 생성하는 데 사용할 수 있는 DXGI 1.0 팩터리를 만듭니다.
CreateDXGIFactory1 다른 DXGI 개체를 생성하는 데 사용할 수 있는 DXGI 1.1 팩터리를 만듭니다.

구조체

DXGI_ADAPTER_DESC DXGI 1.0을 사용하여 어댑터(또는 비디오 카드)를 설명합니다.
DXGI_ADAPTER_DESC1 DXGI 1.1을 사용하는 어댑터(또는 비디오 카드)에 대해 설명합니다.
DXGI_DISPLAY_COLOR_SPACE 이 구조를 사용하지 마세요. 지원되지 않으며 향후 릴리스에서 헤더에서 제거될 예정입니다.
DXGI_FRAME_STATISTICS 프레임의 타이밍 및 프레젠테이션 통계를 설명합니다.
DXGI_MAPPED_RECT 표면에 액세스하는 데 사용되는 매핑된 사각형에 대해 설명합니다.
DXGI_OUTPUT_DESC 어댑터(비디오 카드)와 디바이스 간의 출력 또는 물리적 연결을 설명합니다.
DXGI_SHARED_RESOURCE 공유 리소스에 대한 핸들을 나타냅니다.
DXGI_SURFACE_DESC 표면을 설명합니다.
DXGI_SWAP_CHAIN_DESC 스왑 체인에 대해 설명합니다. (DXGI_SWAP_CHAIN_DESC)

열거형

DXGI_ADAPTER_FLAG DXGI 어댑터의 유형을 식별합니다. (DXGI_ADAPTER_FLAG)

DXGI_RESIDENCY 리소스의 메모리 위치를 나타내는 플래그입니다.
DXGI_SWAP_CHAIN_FLAG 스왑 체인 동작에 대한 옵션입니다.
DXGI_SWAP_EFFECT IDXGISwapChain1::P resent1을 호출한 후 디스플레이 화면에서 픽셀을 처리하는 옵션입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dxgi1_2.h 헤더

아티클2023. 08. 23.

이 헤더는 DXGI에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DXGI

dxgi1_2.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDXGIAdapter2 IDXGIAdapter2 인터페이스는 하나 이상의 GPU, DAC 및 비디오 메모리를 포함하는 디스플레이 하위 시스템을 나타냅니다.
IDXGIDevice2 IDXGIDevice2 인터페이스는 이미지 데이터를 생성하는 DXGI 개체에 대한 파생 클래스를 구현합니다. 인터페이스는 GPU가 처리를 완료할 때까지 CPU 처리를 차단하고 운영 체제에 리소스를 제공하는 메서드를 노출합니다.
IDXGIDisplayControl IDXGIDisplayControl 인터페이스는 운영 체제의 입체 3D 디스플레이 동작에 대한 사용자 기본 설정을 나타내고 스테레오스코픽 3D 디스플레이 상태 설정하여 사용하거나 사용하지 않도록 설정하는 메서드를 노출합니다.
IDXGIFactory2 IDXGIFactory2 인터페이스에는 IDXGISwapChain보다 더 많은 기능을 사용하여 최신 버전 스왑 체인을 만들고 입체 3D 기능을 모니터링하는 메서드가 포함되어 있습니다.
IDXGIOutput1 IDXGIOutput1 인터페이스는 어댑터 출력(예: 모니터)을 나타냅니다.
IDXGIOutputDuplication IDXGIOutputDuplication 인터페이스는 중복된 데스크톱 이미지에 액세스하고 조작합니다.
IDXGIResource1 IDXGIResource1 인터페이스는 하위 리소스 표면 개체를 만들고 공유 리소스에 대한 핸들을 만들기 위한 지원을 추가하여 IDXGIResource 인터페이스를 확장합니다.

IDXGISurface2 IDXGISurface2 인터페이스는 하위 리소스 표면에 대한 지원을 추가하고 공유 리소스에 대한 핸들을 얻어 IDXGISurface1 인터페이스를 확장합니다.
IDXGISwapChain1 IDXGISwapChain에서 향상된 프레젠테이션 기능을 제공합니다. 이러한 프레젠테이션 기능은 프레젠테이션을 최적화하기 위해 더티 사각형 및 스크롤 사각형을 지정하는 것으로 구성됩니다.

구조체

DXGI_ADAPTER_DESC2 Microsoft DXGI(DirectX Graphics Infrastructure) 1.2를 사용하는 어댑터(또는 비디오 카드)에 대해 설명합니다.
DXGI_MODE_DESC1 디스플레이 모드 및 디스플레이 모드가 스테레오를 지원하는지 여부를 설명합니다.
DXGI_OUTDUPL_DESC DXGI_OUTDUPL_DESC 구조체는 출력의 차원과 바탕 화면 이미지를 포함하는 표면을 설명합니다. 데스크톱 이미지의 형식은 항상 DXGI_FORMAT_B8G8R8A8_UNORM.
DXGI_OUTDUPL_FRAME_INFO DXGI_OUTDUPL_FRAME_INFO 구조는 현재 데스크톱 이미지를 설명합니다.
DXGI_OUTDUPL_MOVE_RECT DXGI_OUTDUPL_MOVE_RECT 구조체는 사각형의 움직임을 설명합니다.
DXGI_OUTDUPL_POINTER_POSITION DXGI_OUTDUPL_POINTER_POSITION 구조체는 하드웨어 커서의 위치를 설명합니다.
DXGI_OUTDUPL_POINTER_SHAPE_INFO DXGI_OUTDUPL_POINTER_SHAPE_INFO 구조체는 커서 셰이프에 대한 정보를 설명합니다.
DXGI_PRESENT_PARAMETERS 운영 체제에서 프레젠테이션을 최적화하는 데 도움이 되는 현재에 대한 정보를 설명합니다.

DXGI_SWAP_CHAIN_DESC1 스왑 체인에 대해 설명합니다. (DXGI_SWAP_CHAIN_DESC1)
DXGI_SWAP_CHAIN_FULLSCREEN_DESC 스왑 체인의 전체 화면 모드에 대해 설명합니다.

열거형

DXGI_ALPHA_MODE 표면의 알파 값, 투명도 동작을 식별합니다.
DXGI_COMPUTE_PREEMPTION_GRANULARITY GPU(그래픽 처리 장치)가 현재 컴퓨팅 작업을 수행할 때 선점될 수 있는 세분성을 식별합니다.
DXGI_GRAPHICS_PREEMPTION_GRANULARITY 현재 그래픽 렌더링 작업을 수행할 때 GPU(그래픽 처리 장치)를 선점할 수 있는 세분성을 식별합니다.
DXGI_OFFER_RESOURCE_PRIORITY IDXGIDevice2::OfferResources 메서드를 호출하여 리소스를 제공할 때 리소스 콘텐츠의 중요도를 식별합니다.
DXGI_OUTDUPL_POINTER_SHAPE_TYPE 포인터 셰이프의 유형을 식별합니다.
DXGI_SCALING 백 버퍼 크기가 대상 출력의 크기와 일치하지 않는 경우 크기 조정 동작을 식별합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dxgi1_3.h 헤더

아티클2023. 08. 23.

이 헤더는 DXGI에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DXGI

dxgi1_3.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDXGIDecodeSwapChain 데스크톱 미디어 앱에서 비디오 데이터를 디코딩하고 DirectComposition 화면에 표시하는 데 사용되는 스왑 체인을 나타냅니다.
IDXGIDevice3 IDXGIDevice3 인터페이스는 이미지 데이터를 생성하는 DXGI 개체에 대한 파생 클래스를 구현합니다. 인터페이스는 DXGI 디바이스에서 그래픽 메모리 사용량을 자르는 메서드를 노출합니다.
IDXGIFactory3 Microsoft DXGI(DirectX Graphics Infrastructure) 개체를 만들 수 있습니다. (IDXGIFactory3)
IDXGIFactoryMedia DirectComposition 화면을 사용하여 비디오를 디코딩하고 표시하는 데스크톱 미디어 앱용 스왑 체인을 만듭니다.
IDXGIOutput2 어댑터 출력(예: 모니터)을 나타냅니다. IDXGIOutput2 인터페이스는 기본 출력 어댑터에서 다중 평면 오버레이 지원을 위해 검사 메서드를 노출합니다.
IDXGIOutput3 어댑터 출력(예: 모니터)을 나타냅니다. IDXGIOutput3 인터페이스는 오버레이 지원을 위해 검사 메서드를 노출합니다.
IDXGISwapChain2 스왑 백 버퍼 크기 조정 및 짧은 대기 시간 스왑 체인을 지원하는 메서드를 사용하여 IDXGISwapChain1을 확장합니다.

IDXGISwapChainMedia 이 스왑 체인 인터페이스를 사용하면 데스크톱 미디어 애플리케이션이 특정 새로 고침 속도에 대한 원활한 변경을 요청할 수 있습니다.

Functions

CreateDXGIFactory2 다른 DXGI 개체를 생성하는 데 사용할 수 있는 DXGI 1.3 팩터리를 만듭니다.
DXGIGetDebugInterface1 Windows 스토어 앱이 Microsoft DXGI(DirectX Graphics Infrastructure)를 디버깅하는 데 사용하는 인터페이스를 검색합니다.

구조체

DXGI_DECODE_SWAP_CHAIN_DESC IDXGIFactoryMedia::CreateDecodeSwapChainForCompositionSurfaceHandle과 함께 디코딩 스왑 체인을 설명하는 데 사용됩니다.
DXGI_FRAME_STATISTICS_MEDIA 앱의 사용자 지정 현재 기간(사용자 지정 새로 고침 속도)에 대한 시스템 승인을 확인하는 데 사용됩니다.
DXGI_MATRIX_3X2_F 3x2 행렬을 나타냅니다. GetMatrixTransform 및 SetMatrixTransform과 함께 사용하여 SwapChainPanel 스왑 체인의 크기 조정 및 변환 변환을 나타냅니다.

열거형

DXGI_FRAME_PRESENTATION_MODE

스왑 체인에 프레임을 표시하는 옵션을 나타냅니다.
DXGI_MULTIPLANE_OVERLAY_YCbCr_FLAGS 스왑 체인 색 공간에 대한 옵션입니다.
DXGI_OVERLAY_SUPPORT_FLAG IDXGIOutput3::CheckOverlaySupport 호출에서 에 대한 검사 오버레이 지원을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dxgi1_4.h 헤더

아티클2023. 08. 23.

이 헤더는 DXGI에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DXGI

dxgi1_4.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDXGIAdapter3 이 인터페이스는 실제 메모리의 예산을 책정하고 예약하기 위한 몇 가지 메모리 상주 메서드를 추가합니다.
IDXGIFactory4 Microsoft DXGI(DirectX Graphics Infrastructure) 개체를 만들 수 있습니다. (IDXGIFactory4)
IDXGIOutput4 어댑터 출력(예: 모니터)을 나타냅니다. IDXGIOutput4 인터페이스는 오버레이 색 공간 지원을 위해 검사 메서드를 노출합니다.
IDXGISwapChain3 스왑 체인의 현재 백 버퍼 인덱스 가져오기 및 색 공간 지원을 지원하는 메서드를 사용하여 IDXGISwapChain2를 확장합니다.

구조체

DXGI_QUERY_VIDEO_MEMORY_INFO 현재 비디오 메모리 예산 매개 변수에 대해 설명합니다.

열거형

DXGI_MEMORY_SEGMENT_GROUP 사용할 메모리 세그먼트 그룹을 지정합니다.
DXGI_OVERLAY_COLOR_SPACE_SUPPORT_FLAG 오버레이 색 공간에 대한 지원을 지정합니다.
DXGI_SWAP_CHAIN_COLOR_SPACE_SUPPORT_FLAG 스왑 체인에 대한 색 공간 지원을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dxgi1_5.h 헤더

아티클2023. 08. 23.

이 헤더는 DXGI에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DXGI

dxgi1_5.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDXGIDevice4 이 인터페이스는 리소스를 제공하고 회수하는 업데이트된 메서드를 제공합니다.
IDXGIFactory5 이 인터페이스를 사용하면 단일 메서드가 가변 새로 고침 빈도 표시를 지원할 수 있습니다.
IDXGIOutput5 어댑터 출력(예: 모니터)을 나타냅니다. IDXGIOutput5 인터페이스는 전체 화면 화면에 지원되는 형식 목록을 지정하는 단일 메서드를 노출합니다.
IDXGISwapChain4 이 인터페이스는 비디오 메타데이터를 설정하기 위한 단일 메서드를 노출합니다.

구조체

DXGI_HDR_METADATA_HDR10 HEVC(고효율 비디오 코딩)를 사용하여 비디오를 압축할 때 사용되는 HDR10에 대한 메타데이터를 설명합니다.

열거형

DXGI_FEATURE 기능 지원을 확인할 때 사용할 다양한 하드웨어 기능을 지정합니다.
DXGI_HDR_METADATA_TYPE 헤더 메타데이터 형식을 지정합니다.
DXGI_OFFER_RESOURCE_FLAGS OfferResources1 메서드에 대한 플래그를 지정합니다.
DXGI_RECLAIM_RESOURCE_RESULTS ReclaimResources1 메서드의 결과 플래그를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dxgi1_6.h 헤더

아티클2023. 08. 23.

이 헤더는 DXGI에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DXGI

dxgi1_6.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDXGIAdapter4 이 인터페이스는 디스플레이 하위 시스템을 나타내며 이 인터페이스 제품군을 확장하여 어댑터의 ACG(임의 코드 가드)와의 호환성을 위해 검사 메서드를 노출합니다.
IDXGIFactory6 이 인터페이스를 사용하면 지정된 GPU 기본 설정에 따라 그래픽 어댑터를 열거하는 단일 메서드를 사용할 수 있습니다.
IDXGIFactory7 이 인터페이스를 사용하면 알림을 등록하여 어댑터 열거 상태 변경을 검색할 수 있습니다.
IDXGIOutput6 어댑터 출력(예: 모니터)을 나타냅니다. IDXGIOutput6 인터페이스는 특정 모니터 기능을 제공하는 메서드를 노출합니다.

Functions

DXGIDeclareAdapterRemovalSupport 프로세스에서 제거되는 그래픽 디바이스에 복원력이 있음을 나타낼 수 있습니다.
DXGIDisableVBlankVirtualization 프로세스에 대해 v-blank 가상화를 사용하지 않도록 설정합니다. 이 가상화는 IDXGIOutput::WaitForVBlank에서 안정적인 가상화된 현재 속도 및 v-blank 주기를 유지하기 위해

기본적으로 모든 스왑 체인에 대해 DRR(동적 새로 고침 속도) 기능에 사용됩니다. 가상화를 사용하지 않도록 설정하면 이러한 API에 변경된 새로 고침 빈도가 표시됩니다.

구조체

DXGI_ADAPTER_DESC3 Microsoft DXGI(DirectX Graphics Infrastructure) 1.6을 사용하는 어댑터(또는 비디오 카드)에 대해 설명합니다.
DXGI_OUTPUT_DESC1 색 기능 및 연결 유형에 대한 추가 정보를 포함하여 어댑터(비디오 카드)와 디바이스 간의 출력 또는 물리적 연결을 설명합니다.

열거형

DXGI_ADAPTER_FLAG3 DXGI 어댑터의 유형을 식별합니다. (DXGI_ADAPTER_FLAG3)
DXGI_GPU_PREFERENCE 실행할 앱에 대한 GPU의 기본 설정입니다.
DXGI_HARDWARE_COMPOSITION_SUPPORT_FLAGS 지원되는 하드웨어 구성 수준을 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dxgicommon.h 헤더

아티클2023. 08. 23.

이 헤더는 DXGI에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DXGI

dxgicommon.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

DXGI_RATIONAL 합리적인 숫자를 나타냅니다.
DXGI_SAMPLE_DESC 리소스에 대한 다중 샘플링 매개 변수에 대해 설명합니다.

열거형

DXGI_COLOR_SPACE_TYPE 색 공간 유형을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dxgidebug.h 헤더

아티클2023. 08. 23.

이 헤더는 DXGI에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DXGI

dxgidebug.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDXGIDebug 이 인터페이스는 디버그 설정을 제어하며 디버그 계층이 켜져 있는 경우에만 사용할 수 있습니다.
IDXGIDebug1 Microsoft DXGI(DirectX Graphics Infrastructure)에 대한 디버그 설정을 제어합니다. Windows 스토어 앱에서 IDXGIDebug1 인터페이스를 사용할 수 있습니다.
IDXGIInfoQueue 이 인터페이스는 디버그 정보 큐를 제어하며 디버그 계층이 켜져 있는 경우에만 사용할 수 있습니다.

Functions

DXGIGetDebugInterface 디버깅 인터페이스를 검색합니다.

구조체

DXGI_INFO_QUEUE_FILTER 허용 및 거부할 메시지 유형 목록이 포함된 디버그 메시지 필터에 대해 설명합니다.

DXGI_INFO_QUEUE_FILTER_DESC 필터를 통과하도록 허용하거나 거부할 메시지 유형에 대해 설명합니다.
DXGI_INFO_QUEUE_MESSAGE 정보 큐의 디버그 메시지를 설명합니다.

열거형

DXGI_DEBUG_RLO_FLAGS ReportLiveObjects와 함께 개체의 수명에 대해 보고할 정보의 양을 지정하는 데 사용되는 플래그입니다.
DXGI_INFO_QUEUE_MESSAGE_CATEGORY 디버그 메시지의 범주를 지정하는 값입니다.
DXGI_INFO_QUEUE_MESSAGE_SEVERITY 정보 큐에 대한 디버그 메시지 심각도 수준을 지정하는 값입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dxgiformat.h 헤더

아티클2023. 08. 23.

이 헤더는 DXGI에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DXGI

dxgiformat.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

열거형

DXGI_FORMAT 완전 형식 및 무형식 형식을 포함한 리소스 데이터 형식입니다. 페이지 맨 아래에 있는 한정자 목록은 각 형식 형식을 보다 완벽하게 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dxmini.h 헤더

아티클2023. 08. 23.

이 헤더는 디스플레이 디바이스 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 디바이스 참조 표시

dxmini.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

콜백 함수

PDX_BOBNEXTFIELD DxBobNextField 콜백 함수는 인터리브된 데이터의 다음 필드를 표시합니다.
PDX_ENABLEIRQ DxEnableIRQ 콜백 함수는 비디오 미니포트 드라이버에 IRQ를 사용하거나 사용하지 않도록 설정해야 함을 나타냅니다.
PDX_FLIPOVERLAY DxFlipOverlay 콜백 함수는 비디오 미니포트 드라이버의 클라이언트가 오버레이를 대칭 이동하려고 하거나 자동 매핑을 사용하도록 설정할 때 호출됩니다.
PDX_FLIPVIDEOPORT DxFlipVideoPort 콜백 함수는 비디오 미니포트 드라이버의 클라이언트가 VPE(비디오 포트 확장) 개체를 대칭 이동하려고 하거나 자동 플리핑을 사용하도록 설정할 때 호출됩니다.
PDX_GETCURRENTAUTOFLIP DxGetCurrentAutoflip 콜백 함수는 디바이스가 하드웨어 자동 매핑이고 비디오 미니포트 드라이버의 클라이언트가 캡처를 위해 비디오 데이터의 현재 필드를 수신하는 화면을 알고자 할 때 호출됩니다.
PDX_GETIRQINFO DxGetIRQInfo 콜백 함수는 드라이버가 인터럽트 요청을 관리한다는 것을 나타냅니다.
PDX_GETPOLARITY DxGetPolarity 콜백 함수는 VPE(비디오 포트 확장) 개체에 의해 작성되는 현재 필드의 극성(짝수 또는 홀수)을 반환합니다.

PDX_GETPREVIOUSAUTOFLIP DxGetPreviousAutoflip 콜백 함수는 디바이스가 하드웨어 자동 매핑이고 비디오 미니포트 드라이버의 클라이언트가 캡처를 위해 비디오 데이터의 이전 필드를 수신한 화면을 알고자 할 때 호출됩니다.
PDX_GETTRANSFERSTATUS DxGetTransferStatus 콜백 함수는 DirectDraw에서 어떤 하드웨어 버스 master 완료했는지 확인하는 데 사용됩니다.
PDX_IRQCALLBACK IRQCallback 함수는 발생한 IRQ와 관련된 작업을 수행합니다.
PDX_LOCK DxLock 콜백 함수는 비디오 미니포트 드라이버의 클라이언트가 프레임 버퍼에 액세스하려고 할 때 호출됩니다.
PDX_SETSTATE DxSetState 콜백 함수는 비디오 미니포트 드라이버의 클라이언트가 bob 모드에서 직조 모드로 전환하려고 하며 그 반대의 경우도 마찬가지입니다.
PDX_SKIPNEXTFIELD 다음 필드를 건너뛰거나 다시 활성화해야 하는 경우 DxSkipNextField 콜백 함수가 호출됩니다.
PDX_TRANSFER DxTransfer 콜백 함수는 드라이버에 표면의 master 데이터를 MDL(메모리 설명자 목록)에 지정된 버퍼로 버스에 알립니다.

구조체

DDBOBNEXTFIELDINFO DDBOBNEXTFIELDINFO 구조체에는 표면에 대한 bob 정보가 포함됩니다.
DDENABLEIRQINFO DDENABLEIRQINFO 구조체에는 인터럽트 사용 설정에 필요한 정보가 포함되어 있습니다.

DDFLIPOVERLAYINFO DDFLIPOVERLAYINFO 구조체에는 표면에 대한 대칭 이동 정보가 포함됩니다.
DDFLIPVIDEOPORTINFO DDFLIPVIDEOPORTINFO 구조체에는 VPE(비디오 포트 확장) 개체 및 표면 정보가 포함됩니다.
DDGETCURRENTAUTOFLIPININFO DDGETCURRENTAUTOFLIPININFO 구조체에는 VPE(비디오 포트 확장) 개체 정보가 포함됩니다.
DDGETCURRENTAUTOFLIPOUTINFO DDGETCURRENTAUTOFLIPOUTINFO 구조체는 표면 정보를 제공합니다.
DDGETIRQINFO DDGETIRQINFO 구조체에는 비디오 미니포트 드라이버에 대한 인터럽트 정보가 포함됩니다.
DDGETPOLARITYININFO DDGETPOLARITYININFO 구조체에는 VPE(비디오 포트 확장) 개체 정보가 포함됩니다.
DDGETPOLARITYOUTINFO DDGETPOLARITYOUTINFO 구조체에는 VPE(비디오 포트 확장) 개체의 극성 정보가 포함됩니다.
DDGETPREVIOUSAUTOFLIPININFO DDGETPREVIOUSAUTOFLIPININFO 구조체는 VPE(비디오 포트 확장) 개체 정보를 제공합니다.
DDGETPREVIOUSAUTOFLIPOUTINFO DDGETPREVIOUSAUTOFLIPOUTINFO 구조체는 표면 데이터를 제공합니다.
DDGETTRANSFERSTATUSOUTINFO DDGETTRANSFERSTATUSOUTINFO 구조체에는 전송 상태 정보가 포함됩니다.
DDLOCKININFO DDLOCKININFO 구조체에는 표면 정보가 포함됩니다.
DDLOCKOUTINFO DDLOCKOUTINFO 구조체에는 DxLock 함수의 표면 정보 출력이 포함됩니다.
DDSETSTATEININFO

DDSETSTATEININFO 구조체에는 표면 및 VPE(비디오 포트 확장) 개체 정보가 포함됩니다.
DDSETSTATEOUTINFO DDSETSTATEOUTINFO 구조체에는 VPE(비디오 포트 확장) 개체에 대한 상태 정보가 포함됩니다.
DDSKIPNEXTFIELDINFO DDSKIPNEXTFIELDINFO 구조체에는 VPE(비디오 포트 확장) 개체에 대한 건너뛰기 정보가 포함됩니다.
DDSURFACEDATA DDSURFACEDATA 구조체는 DirectDraw에서 커널 모드 미니포트 드라이버에 표면을 나타내는 데 사용됩니다.
DDTRANSFERININFO DDTRANSFERININFO 구조체에는 표면에 대한 전송 정보가 포함됩니다.
DDTRANSFEROUTINFO DDTRANSFEROUTINFO 구조체는 캡처되는 필드의 극성을 반환합니다.
DDVIDEOPORTDATA DDVIDEOPORTDATA 구조체는 DirectDraw에서 커널 모드 비디오 미니포트 드라이버에 대한 VPE(비디오 포트 확장) 개체를 나타내는 데 사용됩니다.
DX_IRQDATA DX_IRQDATA 구조체에는 드라이버에서 제공하는 IRQ 정보가 포함됩니다.
DXAPI_INTERFACE DXAPI_INTERFACE 구조에는 비디오 미니포트 드라이버가 Kernel-Mode 비디오 전송을 지원하기 위해 구현하는 인터페이스 콜백 함수가 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dxva2api.h 헤더

아티클2023. 08. 23.

이 헤더는 Microsoft Media Foundation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft Media Foundation

dxva2api.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDirect3DDeviceManager9 두 스레드가 동일한 Direct3D 9 디바이스를 공유할 수 있도록 하고 디바이스의 DXVA(DirectX Video Acceleration) 기능에 대한 액세스를 제공합니다.
IDirectXVideoAccelerationService Direct3D 디바이스에서 DXVA(DirectX Video Acceleration) 서비스를 제공합니다.
IDirectXVideoDecoder DXVA(DirectX 비디오 가속) 비디오 디코더 디바이스를 나타냅니다.
IDirectXVideoDecoderService DXVA(DirectX Video Acceleration) 디코더 서비스에 대한 액세스를 제공합니다.
IDirectXVideoMemoryConfiguration 압축되지 않은 비디오 화면의 비디오 메모리 유형을 설정합니다.
IDirectXVideoProcessor DXVA(DirectX Video Acceleration) 비디오 프로세서 디바이스를 나타냅니다.
IDirectXVideoProcessorService DXVA(DirectX Video Acceleration) 비디오 처리 서비스에 대한 액세스를 제공합니다.

Functions

DXVA2_Fixed32OpaqueAlpha 불투명 알파 값을 포함하는 DXVA2_Fixed32 구조를 반환합니다.
DXVA2_Fixed32TransparentAlpha 투명한 알파 값을 포함하는 DXVA2_Fixed32 구조를 반환합니다.
DXVA2CreateDirect3DDeviceManager9 Direct3D 디바이스 관리자 instance 만듭니다.
DXVA2CreateVideoService DXVA(DirectX Video Acceleration) 서비스 개체를 만듭니다.
DXVA2FixedToFloat DXVA2_Fixed32 값을 부동 소수점 숫자로 변환합니다.
DXVA2FloatToFixed 부동 소수점 숫자를 DXVA2_Fixed32 값으로 변환합니다.

구조체

DXVA2_AES_CTR_IV 128비트 AES-CTR(Advanced Encryption Standard CTR 모드) 블록 암호 암호화에 대한 IV(초기화 벡터)를 포함합니다. (DXVA2_AES_CTR_IV)
DXVA2_AYUVSample16 16비트 AYUV 픽셀 값을 정의합니다.
DXVA2_AYUVSample8 8비트 AYUV 픽셀 값을 정의합니다.
DXVA2_ConfigPictureDecode DXVA 디코더 디바이스의 구성에 대해 설명합니다.
DXVA2_DecodeBufferDesc

디코더에서 DXVA(DirectX Video Acceleration) 디바이스로 전송되는 버퍼에 대해 설명합니다.
DXVA2_DecodeExecuteParams IDirectXVideoDecoder::Execute 메서드에 대한 매개 변수를 포함합니다.
DXVA2_DecodeExtensionData IDirectXVideoDecoder::Execute 메서드에 대한 프라이빗 데이터를 포함합니다.
DXVA2_ExtendedFormat 비디오 스트림의 형식을 설명합니다.
DXVA2_FilterValues DXVA(DirectX Video Acceleration) 이미지 필터에 대한 매개 변수를 포함합니다.
DXVA2_Fixed32 32비트 고정 소수점 숫자를 정의합니다.
DXVA2_Frequency 비디오 빈도를 정의합니다.
DXVA2_ProcAmpValues DXVA(DirectX Video Acceleration) 비디오 처리 작업에 대한 값을 포함합니다.
DXVA2_ValueRange DXVA(DirectX Video Acceleration) 작업에 지원되는 값의 범위를 정의합니다.
DXVA2_VideoDesc DXVA 디코더 디바이스 또는 비디오 프로세서 디바이스에 대한 비디오 스트림을 설명합니다.
DXVA2_VideoProcessBltParams IDirectXVideoProcessor::VideoProcessBlt 메서드에 대한 매개 변수를 포함합니다.
DXVA2_VideoProcessorCaps DVXA(DirectX 비디오 가속) 비디오 프로세서 모드의 기능을 설명합니다.
DXVA2_VideoSample IDirectXVideoProcessor::VideoProcessBlt 메서드에 대한 입력 샘플을 지정합니다.

열거형

DXVA2_NominalRange 색 데이터를 정규화된 [0...1] 범위에 매핑하는 방법을 설명합니다.
DXVA2_SampleFormat 비디오 샘플의 내용을 설명합니다. 이러한 플래그는 DXVA2_ExtendedFormat 구조에서 사용됩니다.
DXVA2_SurfaceType DXVA(DirectX Video Acceleration)의 DirectX 표면 형식에 대해 설명합니다.
DXVA2_VideoChromaSubSampling YUV 비디오 프레임에서 루마 샘플을 기준으로 크로마 값을 배치하는 방법을 설명합니다.
DXVA2_VideoLighting 비디오 콘텐츠를 보기 위한 의도된 조명 조건에 대해 설명합니다.
DXVA2_VideoPrimaries 비디오 원본의 색 기본값을 지정합니다. (DXVA2_VideoPrimaries)
DXVA2_VideoTransferFunction 선형 RGB에서 비선형 RGB(R'G'B')로 변환 함수를 지정합니다. (DXVA2_VideoTransferFunction)
DXVA2_VideoTransferMatrix Y'PbPr(구성 요소 비디오)과 스튜디오 R'G'B' 간의 변환 매트릭스에 대해 설명합니다. (DXVA2_VideoTransferMatrix)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dxva9typ.h 헤더

아티클2023. 08. 23.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Directshow
- Microsoft Media Foundation

dxva9typ.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

DXVA_COPPSetProtectionLevelCmdData COPP(Certified Output Protection Protocol)의 보호 수준 설정 명령에 대한 데이터를 포함합니다.
DXVA_COPPSetSignalingCmdData COPP(Certified Output Protection Protocol)의 신호 설정 명령에 대한 정보를 포함합니다.
DXVA_COPPStatusData COPP(Certified Output Protection Protocol) 상태 요청의 결과를 포함합니다.
DXVA_COPPStatusDisplayData COPP(Certified Output Protection Protocol)의 표시 데이터 쿼리 결과를 포함합니다.
DXVA_COPPStatusHDCPKeyData COPP(Certified Output Protection Protocol)의 HDCP 키 데이터 쿼리 결과를 포함합니다. 이 쿼리는 디바이스의 KSV(HDCP 키 선택 벡터)를 반환합니다.
DXVA_COPPStatusSignalingCmdData COPP(Certified Output Protection Protocol)의 신호 쿼리 결과를 포함합니다.
DXVABufferInfo IDirect3DDXVADevice9::Execute 메서드에 대한 버퍼를 지정합니다.
DXVACompBufferInfo DXVA(DirectX Video Acceleration)의 압축된 표면에 대한 요구 사항을 지정합니다.

DXVAUncompDataInfo DXVA(DirectX Video Acceleration) 비디오 디코딩을 위해 압축되지 않은 표면의 크기 및 픽셀 형식을 지정합니다.

열거형

COPP_ACP_Protection_Level ACP 보호 수준을 지정합니다.
COPP_BusType 그래픽 어댑터에서 사용하는 I/O 버스의 유형을 지정합니다.
COPP_CGMSA_Protection_Level CGMS-A 보호 수준을 지정합니다.
COPP_ConnectorType 물리적 커넥터의 유형을 지정합니다.
COPP_HDCP_Protection_Level HDCP 보호 수준을 지정합니다.
COPP_ImageAspectRatio_EN300294 EN 300 294의 가로 세로 비율을 지정합니다.
COPP_StatusFlags COPP(Certified Output Protection Protocol) 세션의 상태 지정합니다.
COPP_StatusHDCPFlags HDCP 상태 플래그를 포함합니다. 이 열거형은 DXVA_COPPStatusHDCPKeyData 구조체에 사용됩니다.
COPP_TVProtectionStandard COPP(Certified Output Protection Protocol)에 대한 TV 보호 표준 목록을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

dxvahd.h 헤더

아티클2023. 08. 23.

이 헤더는 Microsoft Media Foundation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft Media Foundation

dxvahd.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDXVAHD_Device Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스를 나타냅니다.
IDXVAHD_VideoProcessor Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서를 나타냅니다.

Functions

DXVAHD_CreateDevice Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스를 만듭니다.

콜백 함수

PDXVAHDSW_CreateDevice 소프트웨어 플러그 인 Microsoft DXVA-HD(DirectX Video Acceleration High Definition) 디바이스의 instance 만듭니다.
PDXVAHDSW_CreateVideoProcessor

Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서 플러그 인 소프트웨어를 만듭니다.
PDXVAHDSW_DestroyDevice 소프트웨어 플러그 인 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스의 instance 제거합니다.
PDXVAHDSW_DestroyVideoProcessor 소프트웨어 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서를 제거합니다.
PDXVAHDSW_GetVideoProcessBltStatePrivate 소프트웨어 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서에서 프라이빗 블릿 상태를 가져옵니다.
PDXVAHDSW_GetVideoProcessorCaps 하나 이상의 소프트웨어 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서의 기능을 가져옵니다.
PDXVAHDSW_GetVideoProcessorCustomRates 소프트웨어 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서에서 지원하는 사용자 지정 속도를 가져옵니다.
PDXVAHDSW_GetVideoProcessorDeviceCaps 소프트웨어 플러그 인 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스의 기능을 가져옵니다.
PDXVAHDSW_GetVideoProcessorFilterRange 소프트웨어 플러그 인 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스에서 지원되는 이미지 필터 값 범위를 가져옵니다.
PDXVAHDSW_GetVideoProcessorInputFormats 소프트웨어 플러그 인 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스에서 지원하는 입력 형식을 가져옵니다.
PDXVAHDSW_GetVideoProcessorOutputFormats 소프트웨어 플러그 인 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스에서 지원하는 출력 형식을 가져옵니다.
PDXVAHDSW_GetVideoProcessStreamStatePrivate

소프트웨어 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서에서 프라이빗 스트림 상태를 가져옵니다.
PDXVAHDSW_Plugin Microsoft DirectX 비디오 가속 고화질(DXVA-HD)용 소프트웨어 플러그 인 디바이스를 초기화하는 함수에 대한 포인터입니다.
PDXVAHDSW_ProposeVideoPrivateFormat 소프트웨어 플러그 인 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스에서 프라이빗 표면 형식을 가져옵니다.
PDXVAHDSW_SetVideoProcessBltState 소프트웨어 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서에서 blit 작업에 대한 상태 매개 변수를 설정합니다.
PDXVAHDSW_SetVideoProcessStreamState 소프트웨어 Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서의 입력 스트림에 대한 상태 매개 변수를 설정합니다.
PDXVAHDSW_VideoProcessBltHD 비디오 처리 블릿을 수행합니다.

구조체

DXVAHD_BLT_STATE_ALPHA_FILL_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용할 때 출력 알파 값이 blit 작업에 대해 계산되는 방법을 지정합니다.
DXVAHD_BLT_STATE_BACKGROUND_COLOR_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용하는 경우 블릿 작업의 배경색을 지정합니다.
DXVAHD_BLT_STATE_CONSTRICTION_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용할 때 출력이 블릿 작업에서 다운샘플링되는지 여부를 지정합니다.

DXVAHD_BLT_STATE_OUTPUT_COLOR_SPACE_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용하는 경우 blit 작업의 출력 색 공간을 지정합니다.
DXVAHD_BLT_STATE_PRIVATE_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)의 프라이빗 블릿 상태에 대한 데이터를 포함합니다.
DXVAHD_BLT_STATE_TARGET_RECT_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용하는 경우 블리팅을 위한 대상 사각형을 지정합니다.
DXVAHD_COLOR DXVA-HD의 색 값을 정의합니다.
DXVAHD_COLOR_RGBA RGB 색 값을 지정합니다. (DXVAHD_COLOR_RGBA)
DXVAHD_COLOR_YCbCrA YCbCr 색 값을 지정합니다. (DXVAHD_COLOR_YCbCrA)
DXVAHD_CONTENT_DESC Microsoft DXVA-HD(DirectX Video Acceleration High Definition) 비디오 프로세서의 비디오 스트림에 대해 설명합니다.
DXVAHD_CUSTOM_RATE_DATA 프레임 속도 변환 또는 IVTC(역 텔레신)에 대한 사용자 지정 속도를 지정합니다. (DXVAHD_CUSTOM_RATE_DATA)
DXVAHD_FILTER_RANGE_DATA 이미지 필터에 대해 지원되는 값의 범위를 정의합니다. (DXVAHD_FILTER_RANGE_DATA)
DXVAHD_RATIONAL 합리적인 숫자(비율)를 포함합니다.
DXVAHD_STREAM_DATA IDXVAHD_VideoProcessor::VideoProcessBltHD 메서드에 대한 스트림별 데이터를 포함합니다.

DXVAHD_STREAM_STATE_ALPHA_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용하는 경우 입력 스트림의 평면 알파 값을 지정합니다.
DXVAHD_STREAM_STATE_ASPECT_RATIO_DATA 원본 및 대상 사각형의 PAR(픽셀 가로 세로 비율)을 지정합니다.
DXVAHD_STREAM_STATE_D3DFORMAT_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용할 때 입력 스트림의 형식을 지정합니다.
DXVAHD_STREAM_STATE_DESTINATION_RECT_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용하는 경우 입력 스트림의 대상 사각형을 지정합니다.
DXVAHD_STREAM_STATE_FILTER_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 입력 스트림에서 필터링 작업의 수준을 지정합니다.
DXVAHD_STREAM_STATE_FRAME_FORMAT_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 입력 스트림이 인터레이스되는 방법을 지정합니다.
DXVAHD_STREAM_STATE_INPUT_COLOR_SPACE_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 입력 스트림의 색 공간을 지정합니다.
DXVAHD_STREAM_STATE_LUMA_KEY_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용하는 경우 입력 스트림에 대한 루마 키를 지정합니다.
DXVAHD_STREAM_STATE_OUTPUT_RATE_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용할 때 입력 스트림의 출력 프레임 속도를 지정합니다.
DXVAHD_STREAM_STATE_PALETTE_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용하는 경우 입력 스트림에 대한 색상표 항목을 포함합니다.
DXVAHD_STREAM_STATE_PRIVATE_DATA

Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 입력 스트림의 프라이빗 스트림 상태에 대한 데이터를 포함합니다.
DXVAHD_STREAM_STATE_PRIVATE_IVTC_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스의 역 텔레사인(IVTC) 통계를 포함합니다.
DXVAHD_STREAM_STATE_SOURCE_RECT_DATA Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용할 때 입력 스트림의 원본 사각형을 지정합니다.
DXVAHD_VPCAPS Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서의 기능을 지정합니다.
DXVAHD_VPDEVCAPS Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스의 기능을 지정합니다.
DXVAHDSW_CALLBACKS Microsoft DirectX 비디오 가속 고화질(DXVA-HD)용 소프트웨어 플러그 인에서 구현하는 함수에 대한 포인터를 포함합니다.

열거형

DXVAHD_ALPHA_FILL_MODE Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 블릿 작업에 대해 출력 알파 값을 계산하는 방법을 지정합니다.
DXVAHD_BLT_STATE Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용할 때 blit 작업에 대한 상태 매개 변수를 지정합니다.
DXVAHD_DEVICE_CAPS Microsoft DXVA-HD(DirectX Video Acceleration High Definition) 디바이스에 대한 비디오 처리 기능을 정의합니다.
DXVAHD_DEVICE_TYPE

Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스 유형을 지정합니다.
DXVAHD_DEVICE_USAGE Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스에 사용할 용도를 지정합니다.
DXVAHD_FEATURE_CAPS Microsoft DXVA-HD(DirectX Video Acceleration High Definition) 디바이스에서 지원할 수 있는 기능을 정의합니다.
DXVAHD_FILTER Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 처리를 위한 이미지 필터를 지정합니다.
DXVAHD_FILTER_CAPS Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스에 대한 이미지 조정 및 필터링과 관련된 기능을 정의합니다.
DXVAHD_FRAME_FORMAT 비디오 스트림이 인터레이스되는 방법을 설명합니다. (DXVAHD_FRAME_FORMAT)
DXVAHD_INPUT_FORMAT_CAPS Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 디바이스의 입력 형식과 관련된 기능을 정의합니다.
DXVAHD_ITELECINE_CAPS Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서의 역 텔레사인(IVTC) 기능을 지정합니다.
DXVAHD_OUTPUT_RATE Microsoft DirectX 비디오 가속 고화질(DXVA-HD)을 사용하는 경우 입력 스트림의 출력 프레임 속도를 지정합니다.
DXVAHD_PROCESSOR_CAPS Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서의 처리 기능을 지정합니다.
DXVAHD_STREAM_STATE Microsoft DirectX 비디오 가속 고화질(DXVA-HD) 비디오 프로세서에 대한 입력 스트림의 상태 매개 변수를 지정합니다.
DXVAHD_SURFACE_TYPE

Microsoft DXVA-HD(DirectX Video Acceleration High Definition) 디바이스에서 만든 비디오 화면의 유형을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

eapauthenticatoractiondefine.h 헤더

아티클2023. 08. 23.

이 헤더는 확장 가능한 인증 프로토콜 호스트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 확장 가능한 인증 프로토콜 호스트

eapauthenticatoractiondefine.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

EAP_METHOD_AUTHENTICATOR_RESULT EAP 인증자 메서드에서 반환된 인증 결과를 포함합니다.
EapPeerMethodOutput EAP 피어 메서드에서 반환된 작업 정보를 포함합니다.

열거형

EAP_METHOD_AUTHENTICATOR_RESPONSE_ACTION 인증자가 지원자 또는 EAP 피어 메서드로 보낸 응답 명령 집합을 정의합니다.
EapPeerMethodResponseAction 인증 중에 EAP 인증자가 지원자 또는 EAP 피어 메서드에 나타낼 수 있는 작업 집합을 정의합니다.
EapPeerMethodResultReason EAP 인증자 메서드에서 EAP 피어 메서드로 반환된 EAP 인증 세션의 결과 집합을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

eapauthenticatortypes.h 헤더

아티클2023. 08. 23.

이 헤더는 확장 가능한 인증 프로토콜 호스트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 확장 가능한 인증 프로토콜 호스트

eapauthenticatortypes.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

열거형

EAP_AUTHENTICATOR_SEND_TIMEOUT 패킷이 전송된 후 사용자 입력을 기다리는 시간을 인증자 메서드에 나타냅니다. 제한 시간 값은 없음으로 설정할 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

eaphostpeerconfigapis.h 헤더

아티클2023. 08. 23.

이 헤더는 확장 가능한 인증 프로토콜 호스트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 확장 가능한 인증 프로토콜 호스트

eaphostpeerconfigapis.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

EapHostPeerConfigBlob2Xml 구성 BLOB을 XML로 변환합니다. (EapHostPeerConfigBlob2Xml)
EapHostPeerConfigXml2Blob XML을 구성 BLOB으로 변환합니다. (EapHostPeerConfigXml2Blob)
EapHostPeerCredentialsXml2Blob 자격 증명 BLOB을 생성합니다.
EapHostPeerFreeErrorMemory EAP_ERROR 구조에 할당된 메모리를 해제합니다.
EapHostPeerFreeMemory 구성 API에서 반환된 메모리를 해제합니다.
EapHostPeerGetMethodProperties 연결 및 사용자 데이터가 지정된 경우 EAP 메서드의 속성을 검색하는 데 사용됩니다.
EapHostPeerGetMethods 레거시 EAP 메서드를 포함하여 설치되고 사용할 수 있는 모든 EAP 메서드를 열거합니다.
EapHostPeerInvokeConfigUI 지정된 EAP 메서드의 구성 사용자 인터페이스를 시작합니다.

EapHostPeerInvokeIdentityUI 이 함수는 터널 메서드에 의해 호출되어 내부 메서드의 ID UI를 호출합니다. 이 함수는 인증을 시작하기 위해 사용할 자격 증명뿐만 아니라 ID도 반환합니다.
EapHostPeerInvokeInteractiveUI 사용자로부터 자격 증명을 가져오는 데 사용되는 대화형 사용자 인터페이스를 발생합니다.
EapHostPeerQueryCredentialInputFields 사용자가 SSO(Single Sign-On) 시나리오에서 인증을 수행하기 위해 메서드에 필요한 자격 증명 종류를 확인할 수 있습니다.
EapHostPeerQueryInteractiveUIInputFields 지원자에서 발생할 대화형 UI 구성 요소에 대한 입력 필드를 가져옵니다.
EapHostPeerQueryUIBlobFromInteractiveUIInputFields 사용자 정보를 EAPHost 런타임 함수에서 사용할 수 있는 사용자 BLOB으로 변환합니다. (EapHostPeerQueryUIBlobFromInteractiveUIInputFields)
EapHostPeerQueryUserBlobFromCredentialInputFields SSO(Single Sign-On) UI에서 받은 사용자 입력에서 인증을 시작하는 데 사용할 수 있는 자격 증명 BLOB을 가져옵니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

eaphostpeertypes.h 헤더

아티클2023. 08. 23.

이 헤더는 확장 가능한 인증 프로토콜 호스트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 확장 가능한 인증 프로토콜 호스트

eaphostpeertypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

EAPHOST_AUTH_INFO EAP 인증 프로세스의 여러 단계에서 현재 인증 정보를 설명합니다.
EapHostPeerMethodResult EAP 메서드에 전달되는 인증 세션 중에 EAPHost에서 생성된 결과 데이터를 포함합니다.

열거형

EAPHOST_AUTH_STATUS 인증 프로세스 중에 가능한 EAP 인증 세션 상태 값 집합을 정의합니다.
EapHostPeerAuthParams 가능한 인증 매개 변수 값 집합을 정의합니다.
EapHostPeerMethodResultReason EAP 메서드가 지원자에게 반환한 결과를 설명하는 가능한 이유 집합을 정의합니다.
EapHostPeerResponseAction EAP 인증자 또는 피어 메서드가 인증 중에 지원자에 나타낼 수 있는 작업 집합을 정의합니다.
ISOLATION_STATE 컴퓨터의 가능한 격리 상태 값 집합을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

eapmethodauthenticatorapis.h 헤더

아티클2023. 08. 23.

이 헤더는 확장 가능한 인증 프로토콜 호스트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 확장 가능한 인증 프로토콜 호스트

eapmethodauthenticatorapis.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

EapMethodAuthenticatorBeginSession 서버 EAPHost에 새 EAP 인증 세션을 만듭니다.
EapMethodAuthenticatorEndSession 서버 EAPHost에서 EAP 인증 세션을 닫습니다.
EapMethodAuthenticatorFreeErrorMemory EAP 인증자 메서드에서 할당한 오류 관련 메모리를 해제합니다.
EapMethodAuthenticatorFreeMemory 불투명 사용자 인터페이스 컨텍스트 데이터 버퍼와 연결된 모든 메모리를 해제합니다. (EapMethodAuthenticatorFreeMemory)
EapMethodAuthenticatorGetAttributes EAP 인증자 메서드에서 EAP 인증 특성의 배열을 가져옵니다.
EapMethodAuthenticatorGetInfo 로드된 EAP 인증자 메서드의 구현을 위한 함수 포인터 집합을 가져옵니다. EapMethodAuthenticatorGetInfo는 함수 프로토타입입니다.
EapMethodAuthenticatorGetResult EAP 인증자 메서드에서 인증 결과를 가져옵니다.
EapMethodAuthenticatorInitialize 서버 EAPHost에 대한 EAP 인증자 메서드를 초기화합니다.

EapMethodAuthenticatorInvokeConfigUI 클라이언트에서 EAP 메서드의 연결 구성 사용자 인터페이스 대화 상자를 발생시키는 함수를 정의합니다.
EapMethodAuthenticatorReceivePacket 서버 EAPHost에서 받은 EAP 인증 패킷을 처리하고 응답 작업을 반환합니다.
EapMethodAuthenticatorSendPacket EAP 인증자 메서드에서 인증 패킷을 가져와서 지원자에게 보냅니다.
EapMethodAuthenticatorSetAttributes EAP 인증자 메서드에서 설정할 업데이트된 EAP 인증 특성을 제공합니다.
EapMethodAuthenticatorShutdown EAP 인증자 메서드를 종료하고 서버 EAPHost에서 언로드할 준비를 합니다.
EapMethodAuthenticatorUpdateInnerMethodParams EAPHost 서버에서 EapMethodAuthenticatorBeginSession을 호출하여 이전에 설정한 EAP 인증 세션 설정을 업데이트.

구조체

EAP_AUTHENTICATOR_METHOD_ROUTINES EAPHost Authenticator 메서드 API에 대한 함수 포인터 집합을 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

eapmethodpeerapis.h 헤더

아티클2023. 03. 16.

이 헤더는 Extensible 인증 프로토콜 호스트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 확장 가능한 인증 프로토콜 호스트

eapmethodpeerapis.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

EapPeerBeginSession EAP 메서드를 사용하여 피어 EAPHost에서 EAP 인증 세션을 시작합니다.
EapPeerConfigBlob2Xml 구성 BLOB을 XML로 변환합니다. (EapPeerConfigBlob2Xml)
EapPeerConfigXml2Blob XML을 구성 BLOB으로 변환합니다. (EapPeerConfigXml2Blob)
EapPeerCredentialsXml2Blob XML을 구성 BLOB으로 변환합니다. (EapPeerCredentialsXml2Blob)
EapPeerEndSession EAP 메서드에 대한 EAP 인증 세션을 종료합니다.
EapPeerFreeErrorMemory EAP 피어 메서드에 의해 할당된 오류별 메모리를 해제합니다.
EapPeerFreeMemory 불투명 사용자 인터페이스 컨텍스트 데이터 버퍼와 연결된 모든 메모리를 해제합니다. (EapPeerFreeMemory)
EapPeerGetConfigBlobAndUserBlob EAP 메서드 개발자가 메서드에서 지원하는 다양한 연결 속성 및 사용자 속성을 제공할 수 있습니다. EAPHost는 이 함수를 호출하여 EAP 메서드의 연결 속성 및 사용자 속성을 만듭니다.

EapPeerGetIdentity EAPHost에서 호출된 후 사용자 데이터 및 사용자 ID를 반환합니다.
EapPeerGetInfo EAPHost 서비스에 현재 로드된 EAP 피어 메서드 EapPeerGetInfo 구현을 위한 함수 포인터 집합을 가져옵니다.
EapPeerGetMethodProperties 연결 및 사용자 데이터가 지정된 경우 EAP 메서드의 속성을 검색하는 EAP 메서드 관련 함수입니다.
EapPeerGetResponseAttributes EAP 메서드에서 EAP 응답 특성의 배열을 가져옵니다.
EapPeerGetResponsePacket EAP 메서드에서 응답 패킷을 가져옵니다.
EapPeerGetResult EAP 메서드에서 인증 세션의 결과를 가져옵니다.
EapPeerGetUIContext EAP 메서드에서 사용자 인터페이스 컨텍스트를 가져옵니다.
EapPeerInitialize EAPHost에 대한 EAP 피어 메서드를 초기화합니다.
EapPeerInvokeConfigUI 클라이언트에서 EAP 메서드의 특정 연결 구성 사용자 인터페이스 대화 상자를 발생합니다.
EapPeerInvokeIdentityUI 클라이언트에서 EAP 메서드에 대한 사용자 ID 정보를 가져오기 위해 사용자 지정 대화형 사용자 인터페이스 대화 상자를 발생합니다.
EapPeerInvokeInteractiveUI 클라이언트에서 EAP 메서드에 대한 사용자 지정 대화형 사용자 인터페이스 대화 상자를 발생합니다.

EapPeerProcessRequestPacket EAPHost가 지원자로부터 받은 패킷을 처리합니다.
EapPeerQueryCredentialInputFields EAP 메서드에 대한 EAP SSO(Single Sign-On) 자격 증명 입력 필드를 가져오는 EAP 메서드 관련 함수의 구현을 정의합니다.
EapPeerQueryInteractiveUIInputFields 대화형 UI 구성 요소에 대한 입력 필드를 제공하는 EAP 메서드 API의 구현을 정의합니다.
EapPeerQueryUIBlobFromInteractiveUIInputFields 사용자 정보를 EAPHost 런타임 함수에서 사용할 수 있는 사용자 BLOB으로 변환합니다. (EapPeerQueryUIBlobFromInteractiveUIInputFields)
EapPeerQueryUserBlobFromCredentialInputFields 지원자에서 발생한 대화형 SSO(Single Sign-On) UI에 제공된 사용자 BLOB 데이터를 가져오는 EAP 메서드 함수의 구현을 정의합니다.
EapPeerSetCredentials EAP 메서드에 새 인증 또는 업데이트된 인증 자격 증명을 제공합니다.
EapPeerSetResponseAttributes EAP 메서드에 대한 EAP 응답 특성의 업데이트된 배열을 제공합니다.
EapPeerSetUIContext EAP 메서드에 사용자 인터페이스 컨텍스트를 제공합니다. 이 함수는 EapPeerGetUIContext 함수를 통해 UI가 발생한 후에 호출됩니다.
EapPeerShutdown EAP 메서드를 종료하고 해당 DLL을 언로드할 준비를 합니다.

구조체

EAP_PEER_METHOD_ROUTINES EAPHost 피어 메서드 API에 대한 함수 포인터 집합을 포함합니다.

EapPeerMethodResult 인증 중에 EAP 메서드에서 생성된 결과 데이터를 포함합니다.

eapmethodtypes.h 헤더

아티클2023. 08. 23.

이 헤더는 확장 가능한 인증 프로토콜 호스트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 확장 가능한 인증 프로토콜 호스트

eapmethodtypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

EapPacket EAP 인증 세션 중에 전송된 불투명 데이터의 패킷을 포함합니다.

열거형

EapCode EAP 패킷 유형의 집합을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

eappapis.h 헤더

아티클2023. 08. 23.

이 헤더는 확장 가능한 인증 프로토콜 호스트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 확장 가능한 인증 프로토콜 호스트

eappapis.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

EapHostPeerBeginSession EAP 인증 세션을 시작합니다.
EapHostPeerClearConnection 인증 세션 연결을 지웁니다.
EapHostPeerEndSession EAPHost와 호출 지원자 간의 현재 EAP 인증 세션을 종료하고 세션에 대해 저장된 데이터를 지웁니다.
EapHostPeerFreeEapError EAPHost 런타임 API에서 반환된 EAP_ERROR 구조를 해제합니다.
EapHostPeerFreeRuntimeMemory 런타임 중에 사용되는 메모리 공간을 해제합니다.
EapHostPeerGetAuthStatus EAPHost에서 지원자의 현재 EAP 인증 상태 가져옵니다.
EapHostPeerGetDataToUnplumbCredentials EAPHost에서 SSO에 대한 자격 증명을 저장하는 데 사용하는 연결 ID, 사용자 가장 토큰 및 Eaphost 프로세스 ID를 반환합니다. 이 데이터는 이전에 배관된 자격 증명의 삭제를 해제하는 데 필요합니다.
EapHostPeerGetIdentity

이 함수는 내부 메서드에서 ID 정보를 요청하기 위해 터널 메서드에 의해 호출됩니다. 이 함수는 ID 및 사용자 자격 증명 정보를 반환합니다.
EapHostPeerGetResponseAttributes EAPHost에서 EAP 인증 특성의 배열을 가져옵니다.
EapHostPeerGetResult 지정된 EAP 인증 세션에 대한 인증 결과를 가져옵니다.
EapHostPeerGetSendPacket 지원자가 인증자에게 보내기 위해 EAPHost에서 패킷을 가져와야 하는 경우 지원자가 호출합니다.
EapHostPeerGetUIContext UI를 발생시킬 경우 EAPHost에서 지원자의 사용자 인터페이스 컨텍스트를 가져옵니다.
EapHostPeerInitialize EAPHost 인증 세션을 초기화합니다.
EapHostPeerProcessReceivedPacket 지원자가 EAPHost에서 처리해야 하는 패킷을 받을 때마다 지원자가 호출합니다.
EapHostPeerSetResponseAttributes EAPHost에 업데이트된 EAP 인증 특성을 제공합니다.
EapHostPeerSetUIContext UI가 발생한 후 EAPHost에 로드된 EAP 피어 메서드에 새 사용자 인터페이스 또는 업데이트된 사용자 인터페이스 컨텍스트를 제공합니다.
EapHostPeerUninitialize 모든 EAPHost 인증 세션을 초기화하지 않습니다.

콜백 함수

NotificationHandler

SoH(Health 문)가 변경되고 NAP(네트워크 액세스 보호) 시스템 연결의 재인증이 필요하다는 것을 지원인에게 알릴 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

eaptypes.h 헤더

아티클2023. 08. 23.

이 헤더는 확장 가능한 인증 프로토콜 호스트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 확장 가능한 인증 프로토콜 호스트

eaptypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

EAP_ATTRIBUTE EAP 특성을 포함합니다.
EAP_ATTRIBUTES EAP 특성의 배열을 포함합니다.
EAP_CONFIG_INPUT_FIELD_ARRAY 사용자로부터 가져온 사용자 입력 필드 데이터를 집합적으로 포함하는 EAP_CONFIG_INPUT_FIELD_DATA 구조 집합을 포함합니다.
EAP_CONFIG_INPUT_FIELD_DATA 단일 입력 필드와 연결된 데이터를 포함합니다.
EAP_CRED_EXPIRY_REQ 자격 증명 만료 작업에 대한 이전 및 새 EAP 자격 증명을 모두 포함합니다.
EAP_ERROR EAPHost 작업 중에 발생한 오류에 대한 정보를 포함합니다.
EAP_INTERACTIVE_UI_DATA EAP 지원자에서 발생하는 대화형 UI 구성 요소에 대한 구성 정보를 포함합니다.
EAP_METHOD_INFO EAP 메서드에 대한 정보를 포함합니다. (EAP_METHOD_INFO)

EAP_METHOD_INFO_ARRAY 클라이언트 컴퓨터에 설치된 EAP 메서드에 대한 정보를 포함합니다.
EAP_METHOD_INFO_ARRAY_EX 클라이언트 컴퓨터에 설치된 모든 EAP 메서드에 대한 정보를 포함합니다.
EAP_METHOD_INFO_EX EAP 메서드에 대한 정보를 포함합니다. (EAP_METHOD_INFO_EX)
EAP_METHOD_PROPERTY EAP 메서드 속성을 포함합니다.
EAP_METHOD_PROPERTY_ARRAY EAP 메서드 속성의 배열을 포함합니다.
EAP_METHOD_PROPERTY_VALUE EAP 메서드 속성의 값을 포함합니다.
EAP_METHOD_PROPERTY_VALUE_BOOL EAP 메서드 속성의 부울 값을 포함합니다.
EAP_METHOD_PROPERTY_VALUE_DWORD EAP 메서드 속성의 DWORD 값을 포함합니다.
EAP_METHOD_PROPERTY_VALUE_STRING EAP 메서드 속성의 문자열 값을 포함합니다.
EAP_METHOD_TYPE EAP 메서드에 대한 형식, 식별 및 작성자 정보를 포함합니다.
EAP_TYPE EAP 메서드에 대한 형식 및 공급업체 식별 정보를 포함합니다.
EAP_UI_DATA_FORMAT EAP_UI_DATA_FORMAT 공용 구조체는 EAP_INTERACTIVE_UI_DATA 구조체의 pbUiData 멤버에 저장된 특성의 값을 지정합니다.

EapCertificateCredential EAP 메서드가 인증에 사용하는 인증서에 대한 정보를 포함합니다.
EapCredential 자격 증명 유형 및 적절한 자격 증명에 대한 정보를 포함합니다. EapPeerGetConfigBlobAndUserBlob API에 대한 입력으로 전달됩니다.
EapSimCredential 인증을 위해 EAP 메서드에서 사용하는 SIM에 대한 정보를 포함합니다.
EapUsernamePasswordCredential 사용자를 인증하기 위해 EAP 메서드에서 사용하는 사용자 이름 및 암호를 포함합니다.

열거형

EAP_ATTRIBUTE_TYPE 인증 엔터티에서 사용할 수 있는 가능한 EAP 특성 형식 집합을 정의합니다.
EAP_CONFIG_INPUT_FIELD_TYPE 사용자 자격 증명을 쿼리할 때 사용할 수 있는 가능한 입력 필드 형식 집합을 정의합니다.
EAP_INTERACTIVE_UI_DATA_TYPE 특정 지원 API 호출에 제공되는 대화형 UI 컨텍스트 데이터 형식 집합을 지정합니다.
EAP_METHOD_PROPERTY_TYPE 가능한 EAP 메서드 속성 집합을 지정합니다.
EAP_METHOD_PROPERTY_VALUE_TYPE EAP 메서드 속성 값에 대해 가능한 데이터 형식 집합을 정의합니다.
EapCredentialType EapPeerGetConfigBlobAndUserBlob 함수에 전달할 수 있는 가능한 EAP 자격 증명 집합을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

editionupgradehelper.h 헤더

아티클2023. 08. 23.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

editionupgradehelper.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEditionUpgradeHelper Windows 스토어에서 사용자가 구매한 Windows 제품을 설치하거나, 사용자가 현재 설치한 Windows 버전으로 업그레이드를 수행하거나, 정품이 아닌 Windows 복사본을 Windows 정품 복사본으로 바꿀 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

edpwin32.h 헤더

아티클2023. 08. 23.

이 헤더는 WIP(Windows Information Protection)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- WIP(Windows Information Protection)

edpwin32.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

ProtectFileToEnterpriseIdentity 파일의 데이터를 엔터프라이즈 ID로 보호하여 해당 엔터프라이즈 ID와 연결된 사용자만 데이터에 액세스할 수 있도록 합니다. 그런 다음, 애플리케이션은 표준 API를 사용하여 파일에서 읽거나 쓸 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

efswrtinterop.h 헤더

아티클2023. 03. 09.

이 헤더는 WIP(Windows Information Protection)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- WIP(Windows Information Protection)

efswrtinterop.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IProtectionPolicyManagerInterop 보호된 콘텐츠에 대한 엔터프라이즈 보호 정책을 관리합니다. (IProtectionPolicyManagerInterop)
IProtectionPolicyManagerInterop2 보호된 콘텐츠에 대한 엔터프라이즈 보호 정책을 관리합니다. (IProtectionPolicyManagerInterop2)

ehstorapi.h 헤더

아티클2023. 08. 23.

이 헤더는 고급 스토리지에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 강화된 스토리지

ehstorapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnhancedStorageACT 이 인터페이스는 정보를 가져오고 1667 ACT(주소 지정 가능한 연락처 대상)에 대한 작업을 수행합니다.
IEnhancedStorageACT2 IEnhancedStorageACT2 인터페이스는 1667 ACT(주소 지정 가능한 연락처 대상)에 대한 정보를 가져오는 데 사용됩니다.
IEnhancedStorageSilo IEnhancedStorageSilo 인터페이스는 IEEE 1667 사일로에 대한 액세스 지점이며 사일로 수준에서 정보를 얻고 작업을 수행하는 데 사용됩니다.
IEnhancedStorageSiloAction 이 인터페이스를 IEEE 1667 사일로와 관련된 작업에 대한 액세스 지점으로 사용합니다.
IEnumEnhancedStorageACT 이 인터페이스를 모든 IEEE 1667 ACT(주소 지정 가능한 연락처 대상)의 최상위 열거자로 사용합니다.

구조체

ACT_AUTHORIZATION_STATE ACT_AUTHORIZATION_STATE 구조에는 ACT(Addressable Command Target)의 현재 권한 부여 상태를 설명하는 데이터가 포함되어 있습니다.

SILO_INFO SILO_INFO 구조에는 사일로를 식별하고 설명하는 정보가 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ehstorextensions.h 헤더

아티클2023. 08. 23.

이 헤더는 고급 스토리지에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 강화된 스토리지

ehstorextensions.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

ENHANCED_STORAGE_PASSWORD_SILO_INFORMATION ENHANCED_STORAGE_PASSWORD_SILO_INFORMATION 구조에는 암호 사일로의 기능 및 요구 사항을 정의하는 데이터가 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

elscore.h 헤더

아티클2023. 08. 23.

이 헤더는 Windows 애플리케이션에 대한 Internationalization에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 애플리케이션에 대한 국제화

elscore.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

MappingDoAction 텍스트 인식이 발생한 후 ELS 서비스가 작업을 수행하도록 합니다. 예를 들어 전화 걸기 서비스는 먼저 전화 번호를 인식한 다음 전화 걸기의 "작업"을 수행할 수 있습니다.
MappingFreePropertyBag ELS 텍스트 인식 작업 중에 할당된 메모리 및 리소스를 해제합니다.
MappingFreeServices 애플리케이션이 하나 이상의 ELS 서비스와 상호 작용할 수 있도록 할당된 메모리 및 리소스를 해제합니다. 메모리 및 리소스는 MappingGetServices에 대한 애플리케이션 호출에서 할당됩니다.
MappingGetServices 애플리케이션에서 지정한 기준에 따라 관련 정보와 함께 사용 가능한 ELS 플랫폼 지원 서비스 목록을 검색합니다.
MappingRecognizeText ELS 서비스를 호출하여 텍스트를 인식합니다. 예를 들어 Microsoft 언어 감지 서비스는 입력 텍스트가 작성된 언어를 인식하려고 시도합니다.

콜백 함수

PFN_MAPPINGCALLBACKPROC

MappingRecognizeText 함수에서 생성된 데이터를 비동기적으로 처리하는 애플리케이션 정의 콜백 함수입니다.

구조체

MAPPING_DATA_RANGE 인식된 텍스트 하위 범위의 텍스트 인식 결과를 포함합니다. 이 형식의 구조체 배열은 MAPPING_PROPERTY_BAG 구조체의 ELS(확장 언어 서비스) 서비스에 의해 검색됩니다.
MAPPING_ENUM_OPTIONS MappingGetServices 함수에서 ELS 서비스를 열거하는 데 사용하는 옵션을 포함합니다.
MAPPING_OPTIONS 텍스트 인식에 대한 옵션을 포함합니다. 이 구조에 저장된 값은 MappingRecognizeText의 동작 및 결과에 영향을 줍니다.
MAPPING_PROPERTY_BAG MappingRecognizeText에서 검색한 텍스트 인식 데이터 속성을 포함합니다.
MAPPING_SERVICE_INFO ELS 서비스에 대한 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

emi.h 헤더

아티클2023. 08. 23.

이 헤더는 파워 계량 및 예산 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 파워 계량 및 예산 참조

emi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

IOCTLs

IOCTL_EMI_GET_MEASUREMENT IOCTL_EMI_GET_MEASUREMENT 제어 코드는 현재 에너지 측정 및 측정이 수행된 시간을 검색합니다.
IOCTL_EMI_GET_METADATA IOCTL_EMI_GET_METADATA 제어 코드는 디바이스에서 EMI 메타데이터를 검색합니다.
IOCTL_EMI_GET_METADATA_SIZE IOCTL_EMI_GET_METADATA_SIZE 제어 코드는 IOCTL_EMI_GET_METADATA 요청을 실행하여 디바이스에서 가져올 수 있는 EMI 메타데이터 개체의 크기를 검색합니다.
IOCTL_EMI_GET_VERSION IOCTL_EMI_GET_VERSION 제어 코드는 디바이스에서 지원하는 EMI 인터페이스의 현재 버전을 검색합니다.

구조체

EMI_CHANNEL_MEASUREMENT_DATA EMI_MEASUREMENT_DATA 구조는 현재 에너지 측정 및 측정이 수행된 시간에 대한 데이터를 제공합니다.
EMI_CHANNEL_V2 EMI_CHANNEL_V2 구조는 EMI V2 채널에 대한 데이터를 제공합니다.

EMI_MEASUREMENT_DATA_V2 EMI_MEASUREMENT_DATA_V2 구조는 EMI_VERSION_V2 디바이스의 모든 채널에 대한 현재 에너지 측정 데이터에 대한 데이터를 제공합니다.
EMI_METADATA_SIZE EMI_METADATA_SIZE 구조는 IOCTL_EMI_GET_METADATA 요청을 실행하여 디바이스에서 가져올 수 있는 EMI(에너지 계량 인터페이스) 메타데이터 개체의 크기를 지정합니다.
EMI_METADATA_V1 EMI_METADATA 구조는 하드웨어 모델 및 하드웨어 수정 버전과 같은 EMI(에너지 계량 인터페이스) 인터페이스를 지원하는 디바이스에 대한 메타데이터를 제공합니다.
EMI_METADATA_V2 EMI_METADATA_V2 구조는 EMI_VERSION_V2 인터페이스를 지원하는 디바이스에 대한 메타데이터를 제공합니다. 이 메타데이터에는 하드웨어 디바이스 및 이 디바이스에서 노출하는 EMI 채널에 대한 정보가 포함됩니다.
EMI_VERSION EMI_VERSION 구조는 디바이스에서 지원하는 EMI(에너지 계량 인터페이스)의 버전을 설명합니다.

열거형

EMI_MEASUREMENT_UNIT EMI_MEASUREMENT_UNIT 열거형은 IOCTL_EMI_GET_MEASUREMENT 사용하여 디바이스에서 검색할 수 있는 사용 가능한 에너지 측정 단위를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

emptyvc.h 헤더

아티클2023. 08. 23.

이 헤더는 레거시 Windows 환경 기능에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 레거시 Windows 환경 기능

emptyvc.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEmptyVolumeCache 디스크 정리 관리자가 디스크 정리 처리기와 통신하는 데 사용됩니다. 관리자가 처리기에서 정보를 요청하고 검사 또는 제거 시작과 같은 이벤트를 알릴 수 있도록 하는 메서드를 노출합니다.
IEmptyVolumeCache2 IEmptyVolumeCache를 확장합니다. 이 인터페이스는 IEmptyVolumeCache::Initialize보다 더 나은 지역화 지원을 제공하는 하나의 추가 메서드 InitializeEx를 정의합니다.
IEmptyVolumeCacheCallBack 디스크 정리 처리기에서 디스크 정리 관리자와 통신하는 데 사용하는 메서드를 노출합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

enclaveapi.h 헤더

아티클2024. 02. 03.

이 헤더는 System Services 및 Secure Enclave에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스
- 보안 Enclave(신뢰할 수 있는 실행)

enclaveapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

CallEnclave enclave 내에서 함수를 호출합니다.
CreateEnclave 초기화되지 않은 새 enclave를 만듭니다. enclave는 애플리케이션의 주소 공간 내에서 코드 및 데이터의 격리된 영역입니다. enclave 내에서 실행되는 코드만 동일한 Enclave 내의 데이터에 액세스할 수 있습니다.
DeleteEnclave 지정된 enclave를 삭제합니다.
InitializeEnclave 데이터를 사용하여 만들고 로드한 enclave를 초기화합니다.
IsEnclaveTypeSupported 지정된 유형의 enclave가 지원되는지 여부를 검색합니다.
LoadEnclaveData CreateEnclave를 호출하여 만든 초기화되지 않은 enclave에 데이터를 로드합니다.
LoadEnclaveImageA 이미지와 모든 가져오기를 enclave로 로드합니다. (ANSI)

LoadEnclaveImageW 이미지와 모든 가져오기를 enclave로 로드합니다. (유니코드)
TerminateEnclave enclave 내에서 실행되는 스레드의 실행을 종료합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

endpointvolume.h 헤더

아티클2023. 08. 23.

이 헤더는 핵심 오디오 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 핵심 오디오 API

endpointvolume.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAudioEndpointVolume IAudioEndpointVolume 인터페이스는 오디오 엔드포인트 디바이스를 오가는 오디오 스트림의 볼륨 컨트롤을 나타냅니다.
IAudioEndpointVolumeCallback IAudioEndpointVolumeCallback 인터페이스는 오디오 엔드포인트 디바이스의 볼륨 수준 및 음소거 상태 변경에 대한 알림을 제공합니다.
IAudioEndpointVolumeEx IAudioEndpointVolumeEx 인터페이스는 오디오 스트림에서 디바이스 엔드포인트 간 볼륨 컨트롤을 제공합니다.
IAudioMeterInformation IAudioMeterInformation 인터페이스는 오디오 엔드포인트 디바이스를 오가는 오디오 스트림의 최대 측정기를 나타냅니다.

구조체

AUDIO_VOLUME_NOTIFICATION_DATA AUDIO_VOLUME_NOTIFICATION_DATA 구조는 오디오 엔드포인트 디바이스의 볼륨 수준 또는 음소거 상태의 변화를 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

errhandlingapi.h 헤더

아티클2024. 06. 27.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- System Services

errhandlingapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

테이블 확장

AddVectoredContinueHandler 벡터화된 계속 처리기를 등록합니다.
AddVectoredExceptionHandler 벡터화된 예외 처리기를 등록합니다.
FatalAppExitA 메시지 상자를 표시하고 메시지 상자를 닫으면 애플리케이션을 종료합니다. (ANSI)
FatalAppExitW 메시지 상자를 표시하고 메시지 상자를 닫으면 애플리케이션을 종료합니다. (유니코드)
GetErrorMode 현재 프로세스의 오류 모드를 검색합니다.
GetLastError 호출 스레드의 마지막 오류 코드 값을 검색합니다.
GetThreadErrorMode 호출 스레드에 대한 오류 모드를 검색합니다.
raiseException 호출 스레드에서 예외를 발생합니다.

raiseFailFastException 모든 예외 처리기(프레임 또는 벡터 기반)를 우회하는 예외를 발생합니다.
RemoveVectoredContinueHandler 벡터된 계속 처리기를 등록 취소합니다.
RemoveVectoredExceptionHandler 벡터화된 예외 처리기를 등록 취소합니다.
restoreLastError 호출 스레드에 대한 마지막 오류 코드를 복원합니다.
SetErrorMode 시스템 또는 프로세스가 지정된 심각한 오류 유형을 처리하는지 여부를 제어합니다.
setLastError 호출 스레드에 대한 마지막 오류 코드를 설정합니다.
SetThreadErrorMode 시스템에서 지정된 유형의 심각한 오류를 처리할지 또는 호출 스레드에서 처리할지 여부를 제어합니다.
SetUnhandledExceptionFilter 애플리케이션이 프로세스의 각 스레드에 대한 최상위 예외 처리기를 대체하도록 설정합니다.
UnhandledExceptionFilter 프로세스가 디버그되는 경우 처리되지 않은 예외를 디버거에 전달하는 애플리케이션 정의 함수입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

errorrep.h 헤더

아티클2023. 08. 23.

이 헤더는 Windows 오류 보고 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 오류 보고

errorrep.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AddERExcludedApplicationA 지정된 애플리케이션을 오류 보고에서 제외합니다. (ANSI)
AddERExcludedApplicationW 지정된 애플리케이션을 오류 보고에서 제외합니다. (유니코드)
ReportFault 자체 예외 처리를 수행하는 애플리케이션이 Microsoft에 오류를 보고할 수 있도록 합니다.
WerReportHang 지정된 창에서 "응답 없음" 보고를 시작합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

errors.h 헤더

아티클2023. 08. 23.

이 헤더는 DirectShow에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Directshow

errors.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AMGetErrorTextA AMGetErrorText 함수는 현재 언어 설정을 사용하여 지정된 반환 코드에 대한 오류 메시지를 검색합니다. (ANSI)
AMGetErrorTextW AMGetErrorText 함수는 현재 언어 설정을 사용하여 지정된 반환 코드에 대한 오류 메시지를 검색합니다. (유니코드)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

evalcom2.h 헤더

아티클2023. 08. 23.

이 헤더는 애플리케이션 설치 및 서비스에 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 애플리케이션 설치 및 서비스

evalcom2.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IValidate IValidate 인터페이스를 사용하면 작성 도구가 내부 일관성 평가기 집합에 대해 Windows Installer 패키지의 유효성을 검사할 수 있습니다.

콜백 함수

LPDISPLAYVAL LPDISPLAYVAL 함수 사양은 콜백 함수 프로토타입을 정의합니다. IValidate::SetDisplay 메서드를 사용하면 작성 도구가 등록된 콜백 함수를 통해 ICE 상태 메시지를 받을 수 있습니다.
LPEVALCOMCALLBACK LPEVALCOMCALLBACK 사양은 콜백 함수 프로토타입을 정의합니다. IValidate::SetStatus 메서드를 사용하면 작성 도구가 등록된 콜백 함수를 통해 유효성 검사 진행률에 대한 정보를 받을 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

evcoll.h 헤더

아티클2023. 08. 23.

이 헤더는 Windows 이벤트 수집기에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 이벤트 수집기

evcoll.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

EcClose 다른 이벤트 수집기 함수에서 받은 핸들을 닫습니다.
EcDeleteSubscription 기존 구독을 삭제합니다.
EcEnumNextSubscription 로컬 컴퓨터에 등록된 구독의 열거를 계속합니다.
EcGetObjectArrayProperty 구독의 이벤트 원본에 대한 속성 값을 검색합니다.
EcGetObjectArraySize 구독의 이벤트 원본에 대한 속성 값 배열의 인덱스 수를 검색합니다.
EcGetSubscriptionProperty 구독 개체에서 속성 값을 검색합니다.
EcGetSubscriptionRunTimeStatus 구독 또는 구독 자체의 이벤트 원본에 대한 런타임 상태 정보를 검색합니다.
EcInsertObjectArrayElement 구독의 이벤트 원본에 대한 속성 값 배열에 빈 개체를 삽입합니다.
EcOpenSubscription

기존 구독을 열거나 새 구독을 만듭니다.
EcOpenSubscriptionEnum 로컬 컴퓨터에서 등록된 모든 구독을 열거하는 구독 열거자를 만듭니다.
EcRemoveObjectArrayElement 구독의 이벤트 원본에 대한 속성 값을 포함하는 개체의 배열에서 요소를 제거합니다.
EcRetrySubscription 연결되지 않은 구독의 이벤트 원본에 연결을 다시 시도합니다.
EcSaveSubscription 구독 구성 정보를 저장합니다.
EcSetObjectArrayProperty 구독의 이벤트 원본에 대한 속성 값 배열의 속성 값을 설정합니다.
EcSetSubscriptionProperty 새 값을 설정하거나 구독의 기존 값을 업데이트합니다.

구조체

EC_VARIANT 이벤트 수집기 데이터(구독 데이터) 또는 속성 값을 포함합니다.

열거형

EC_SUBSCRIPTION_CONFIGURATION_MODE 구독의 기본 설정을 변경하는 다양한 구성 모드를 지정합니다.
EC_SUBSCRIPTION_CONTENT_FORMAT

이벤트가 이벤트 수집기 컴퓨터로 전송되기 전에 이벤트를 보내는 컴퓨터에서 이벤트를 렌더링하는 방법을 지정합니다.
EC_SUBSCRIPTION_CREDENTIALS_TYPE 이벤트 원본과 통신할 때 사용할 자격 증명 유형을 지정합니다.
EC_SUBSCRIPTION_DELIVERY_MODE 구독에서 이벤트가 전달되는 방법을 나타내는 값을 정의합니다.
EC_SUBSCRIPTION_PROPERTY_ID 구독 구성에 사용되는 이벤트 구독 속성을 식별하는 값을 정의합니다.
EC_SUBSCRIPTION_RUNTIME_STATUS_ACTIVE_STATUS 구독과 관련된 구독 또는 이벤트 원본의 상태 지정합니다.
EC_SUBSCRIPTION_RUNTIME_STATUS_INFO_ID EC_SUBSCRIPTION_RUNTIME_STATUS_INFO_ID 열거형은 구독과 관련된 특정 이벤트 원본의 상태 또는 특정 이벤트 원본의 상태 가져오는 데 사용되는 값을 지정합니다.
EC_SUBSCRIPTION_TYPE 사용할 구독 유형(원본 시작 또는 수집기 시작 구독)을 지정합니다.
EC_VARIANT_TYPE EC_VARIANT_TYPE 열거형은 Windows 이벤트 수집기 함수에 사용되는 데이터 형식을 지정하는 값을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

eventsys.h 헤더

아티클2023. 08. 23.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM+(구성 요소 서비스)
- COM(구성 요소 개체 모델)

eventsys.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumEventObject COM+ 이벤트 저장소에 등록된 이벤트 개체를 열거합니다.
IEventClass 이벤트 개체의 클래스를 해당 개체가 구현하는 이벤트 인터페이스와 연결합니다.
IEventClass2 이벤트 클래스 개체에 대한 데이터를 설정하고 가져오는 데 사용됩니다. 이 인터페이스는 IEventClass 인터페이스를 확장합니다.
IEventControl 이벤트를 구독자에게 발생시키는 개체인 이벤트 개체의 동작을 제어합니다. (IEventControl)
IEventObjectChange 구독자에게 이벤트 저장소에 대한 변경 내용을 알 수 있습니다.
IEventObjectChange2 파티션 및 애플리케이션 ID 정보를 포함하면서 구독자에게 이벤트 저장소에 대한 변경 내용을 알 수 있습니다.
IEventObjectCollection 이벤트 개체 컬렉션의 개체를 관리합니다.
IEventProperty 명명된 이벤트 속성을 해당 값과 연결합니다.

IEventPublisher 이벤트 게시자에 대한 정보를 등록, 수정, 제거 및 제공합니다.
IEventSubscription 이벤트 구독자와 구독 중인 이벤트 간의 관계에 대한 정보를 지정합니다. 게시자 필터에서 사용됩니다.
IEventSystem 이벤트 데이터 저장소에 대한 액세스를 제공합니다.
IFiringControl 단일 구독에 이벤트를 발생합니다.
IMultiInterfaceEventControl 이벤트를 구독자에게 발생시키는 개체인 이벤트 개체의 동작을 제어합니다. (IMultiInterfaceEventControl)
IMultiInterfacePublisherFilter 이벤트 메서드에 대해 필터링된 구독 캐시를 관리합니다.
IPublisherFilter 이벤트 게시자가 이벤트 알림을 받는 구독자 또는 구독자에게 알림을 받는 순서를 제어할 수 있도록 콜백 인터페이스 역할을 합니다.

구조체

COMEVENTSYSCHANGEINFO 이벤트가 시작된 파티션 및 애플리케이션 ID를 포함하는 시스템 이벤트 구조를 나타냅니다.

열거형

EOC_ChangeType

이벤트 개체에 대한 변경의 특성을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

eventtoken.h 헤더

아티클2023. 08. 15.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

eventtoken.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DECLSPEC_XFGVIRT

구조체

EventRegistrationToken 이벤트 원본에 등록된 이벤트 처리기를 식별합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

evntcons.h 헤더

아티클2023. 03. 11.

이 헤더는 이벤트 추적에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 이벤트 추적

evntcons.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

EtwGetTraitFromProviderTraits
EventAccessControl 지정된 공급자 또는 세션의 권한을 추가하거나 수정합니다.
EventAccessQuery 지정된 컨트롤러 또는 공급자에 대한 권한을 검색합니다.
EventAccessRemove 지정된 공급자 또는 세션에 대한 레지스트리에 정의된 권한을 제거합니다.
GetEventProcessorIndex

구조체

EVENT_EXTENDED_ITEM_EVENT_KEY
EVENT_EXTENDED_ITEM_INSTANCE TraceEventInstance를 사용하여 관련 이벤트를 기록하는 경우 이벤트 간의 관계를 정의합니다.
EVENT_EXTENDED_ITEM_PEBS_INDEX

EVENT_EXTENDED_ITEM_PMC_COUNTERS
EVENT_EXTENDED_ITEM_PROCESS_START_KEY
EVENT_EXTENDED_ITEM_RELATED_ACTIVITYID 이 이벤트의 부모 이벤트를 정의합니다.
EVENT_EXTENDED_ITEM_STACK_KEY32
EVENT_EXTENDED_ITEM_STACK_KEY64
EVENT_EXTENDED_ITEM_STACK_TRACE32 32비트 컴퓨터에서 호출 스택을 정의합니다.
EVENT_EXTENDED_ITEM_STACK_TRACE64 64비트 컴퓨터에서 호출 스택을 정의합니다.
EVENT_EXTENDED_ITEM_TS_ID 이벤트를 기록한 터미널 세션을 정의합니다.
EVENT_HEADER EVENT_HEADER 구조체(evntcons.h)는 이벤트에 대한 정보를 정의합니다.
EVENT_HEADER_EXTENDED_DATA_ITEM EVENT_HEADER_EXTENDED_DATA_ITEM 구조체(evntcons.h)는 ETW가 이벤트 데이터의 일부로 수집하는 확장 데이터를 정의합니다.
EVENT_RECORD EVENT_RECORD 구조체(evntcons.h)는 ETW가 제공하는 이벤트의 레이아웃을 정의합니다.

열거형

ETW_PROVIDER_TRAIT_TYPE ETW(Windows용 이벤트 추적)에서 지원하는 공급자 특성 유형을 지정합니다.
EVENTSECURITYOPERATION EventAccessControl 함수가 수정하는 보안 설명자의 구성 요소를 정의합니다.

evntprov.h 헤더

아티클2023. 08. 23.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 드라이버 개발 도구 참조
- 이벤트 추적

evntprov.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

EventActivityIdControl ETW 이벤트에서 사용할 활동 식별자를 만들고, 쿼리하고, 설정합니다.
EventDataDescCreate EVENT_DATA_DESCRIPTOR 값을 설정합니다.
EventDescCreate 이벤트 설명자의 값을 설정합니다.
EventDescGetChannel 이벤트 설명자에서 채널을 검색합니다.
EventDescGetId 이벤트 설명자에서 이벤트 식별자를 검색합니다.
EventDescGetKeyword 이벤트 설명자에서 키워드(keyword) 검색합니다.
EventDescGetLevel 이벤트 설명자에서 심각도 수준을 검색합니다.
EventDescGetOpcode 이벤트 설명자에서 작업 코드를 검색합니다.

EventDescGetTask 이벤트 설명자에서 작업을 검색합니다.
EventDescGetVersion 이벤트 설명자에서 버전을 검색합니다.
EventDescOrKeyword 이벤트 설명자에 다른 키워드(keyword) 추가합니다.
EventDescSetChannel 이벤트 설명자의 채널 멤버를 설정합니다.
EventDescSetId 이벤트 설명자의 ID 멤버를 설정합니다.
EventDescSetKeyword 이벤트 설명자의 Keyword 멤버를 설정합니다.
EventDescSetLevel 이벤트 설명자의 Level 멤버를 설정합니다.
EventDescSetOpcode 이벤트 설명자의 Opcode 멤버를 설정합니다.
EventDescSetTask 이벤트 설명자의 Task 멤버를 설정합니다.
EventDescSetVersion 이벤트 설명자의 Version 멤버를 설정합니다.
EventDescZero 이벤트 설명자를 0으로 초기화합니다.
EventEnabled 이벤트 공급자가 이벤트의 EVENT_DESCRIPTOR 따라 특정 이벤트를 생성해야 하는지 여부를 결정합니다.

EventProviderEnabled 이벤트 공급자가 이벤트의 수준 및 키워드를 기반으로 특정 이벤트를 생성해야 하는지 여부를 결정합니다.
EventRegister ETW 이벤트 공급자를 등록하여 ETW 이벤트를 작성하는 데 사용할 수 있는 핸들을 만듭니다.
EventSetInformation ETW 이벤트 공급자를 구성합니다.
EventUnregister ETW 이벤트 공급자의 등록을 취소합니다.
EventWrite 현재 스레드의 작업 ID를 사용하는 ETW 이벤트를 씁니다.
EventWriteEx 활동 ID, 선택적 관련 활동 ID, 세션 필터 및 특수 옵션을 사용하여 ETW 이벤트를 작성합니다.
EventWriteString 문자열을 데이터로 포함하는 ETW 이벤트를 씁니다. 이 함수를 사용하면 안 됩니다.
EventWriteTransfer 활동 ID 및 선택적 관련 활동 ID를 사용하여 ETW 이벤트를 씁니다.

콜백 함수

PENABLECALLBACK ETW 이벤트 공급자는 필요에 따라 구성 변경 알림을 수신하도록 EnableCallback 함수를 정의합니다. PENABLECALLBACK 형식은 이 콜백 함수에 대한 포인터를 정의합니다. EnableCallback은 애플리케이션 정의 함수 이름의 자리 표시자입니다.

구조체

EVENT_DATA_DESCRIPTOR EVENT_DATA_DESCRIPTOR 구조체는 ETW 이벤트에서 사용할 데이터 블록을 정의합니다.
EVENT_DESCRIPTOR EVENT_DESCRIPTOR 구조에는 ETW 이벤트에 대한 정보(메타데이터)가 포함됩니다.
EVENT_FILTER_DESCRIPTOR 세션이 공급자의 콜백 사용 함수에 전달하는 필터 데이터를 정의합니다.
EVENT_FILTER_EVENT_ID 이벤트 ID 또는 스택 워크 필터의 EVENT_FILTER_DESCRIPTOR 구조에 사용되는 이벤트 ID를 정의합니다.
EVENT_FILTER_EVENT_NAME 이벤트 이름 또는 줄기 워크 이름 필터의 EVENT_FILTER_DESCRIPTOR 구조에 사용되는 이벤트 ID를 정의합니다.
EVENT_FILTER_HEADER 계측 매니페스트에 정의된 필터 데이터 앞에 와야 하는 헤더 데이터를 정의합니다.
EVENT_FILTER_LEVEL_KW 스택 워크 수준 키워드(keyword) 필터의 EVENT_FILTER_DESCRIPTOR 구조에 사용되는 이벤트 ID를 정의합니다.

열거형

EVENT_INFO_CLASS EVENT_INFO_CLASS 열거형 형식은 EventSetInformation 함수와 함께 ETW 이벤트 공급자 등록에서 수행할 구성 작업을 지정하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

evntrace.h 헤더

아티클2023. 03. 11.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 이벤트 추적
- 커널 모드 드라이버 참조

evntrace.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CloseTrace CloseTrace 함수는 OpenTrace를 사용하여 만든 추적 처리 세션을 닫습니다.
ControlTraceA CONTROLTraceA(ANSI) 함수(evntrace.h)는 지정된 이벤트 추적 세션을 플러시, 쿼리, 업데이트 또는 중지합니다.
ControlTraceW ControlTraceW(유니코드) 함수(evntrace.h)는 지정된 이벤트 추적 세션을 플러시, 쿼리, 업데이트 또는 중지합니다.
CreateTraceInstanceId RegisterTraceGuids 기반("클래식") 이벤트 공급자는 CreateTraceInstanceId 함수를 사용하여 고유한 트랜잭션 식별자를 만들고 등록 핸들에 매핑합니다. 그런 다음, TraceEventInstance 함수를 호출할 때 공급자가 트랜잭션 식별자를 사용할 수 있습니다.
EnableTrace 추적 세션 컨트롤러는 EnableTrace를 호출하여 ETW 이벤트 공급자가 이벤트를 추적 세션에 기록하는 방법을 구성합니다. EnableTraceEx2 함수는 이 함수를 대체합니다.
EnableTraceEx 추적 세션 컨트롤러는 EnableTraceEx를 호출하여 ETW 이벤트 공급자가 이벤트를 추적 세션에 기록하는 방법을 구성합니다. EnableTraceEx2 함수는 이 함수를 대체합니다.

EnableTraceEx2 추적 세션 컨트롤러는 EnableTraceEx2를 호출하여 ETW 이벤트 공급자가 이벤트를 추적 세션에 기록하는 방법을 구성합니다.
EnumerateTraceGuids 현재 컴퓨터에서 실행 중인 이벤트 추적 공급자에 대한 정보를 검색합니다. EnumerateTraceGuidsEx 함수는 이 함수를 대체합니다.
EnumerateTraceGuidsEx 현재 컴퓨터에서 실행 중인 이벤트 추적 공급자에 대한 정보를 검색합니다.
FlushTraceA FLUSHTraceA(ANSI) 함수(evntrace.h)를 사용하면 이벤트 추적 세션이 지정된 세션에 대해 버퍼링된 이벤트를 즉시 전달합니다.
FlushTraceW FlushTraceW(유니코드) 함수(evntrace.h)를 사용하면 이벤트 추적 세션이 지정된 세션에 대해 버퍼링된 이벤트를 즉시 전달합니다.
GetTraceEnableFlags RegisterTraceGuids 기반("클래식") 이벤트 공급자는 GetTraceEnableFlags 함수를 사용하여 추적 컨트롤러에서 지정한 사용 플래그를 검색하여 추적할 이벤트의 범주를 나타냅니다. 공급자는 ControlCallback 함수에서 이 함수를 호출합니다.
GetTraceEnableLevel RegisterTraceGuids 기반("클래식") 이벤트 공급자는 GetTraceEnableLevel 함수를 사용하여 추적 컨트롤러에서 지정한 사용 수준을 검색하여 추적할 이벤트의 수준을 나타냅니다. 공급자는 ControlCallback 함수에서 이 함수를 호출합니다.
GetTraceLoggerHandle RegisterTraceGuids 기반("클래식") 이벤트 공급자는 GetTraceLoggerHandle 함수를 사용하여 이벤트를 작성해야 하는 이벤트 추적 세션의 핸들을 검색합니다. 공급자는 ControlCallback 함수에서 이 함수를 호출합니다.
OpenTraceA ANSI(OpenTraceA) 함수(evntrace.h)는 ETW 실시간 추적 세션 또는 ETW 로그 파일에서 이벤트를 소비하기 위한 ETW 추적 처리 핸들을 엽니다.

OpenTraceFromBufferStream 파일 또는 활성 세션에 직접 연결되지 않은 추적 처리 세션을 만듭니다.
OpenTraceFromFile Tracelog .etl 파일을 처리하는 추적 처리 세션을 만듭니다.
OpenTraceFromRealTimeLogger ETW 실시간 추적 세션 또는 ETW 로그 파일에서 이벤트를 소비하기 위한 ETW 추적 처리 핸들을 엽니다.
OpenTraceFromRealTimeLoggerWithAllocationOptions 활성 실시간 ETW 세션에 연결된 추적 처리 세션을 만듭니다.
OpenTraceW OpenTraceW(유니코드) 함수(evntrace.h)는 ETW 실시간 추적 세션 또는 ETW 로그 파일에서 이벤트를 소비하기 위한 ETW 추적 처리 핸들을 엽니다.
ProcessTrace 하나 이상의 추적 처리 세션에서 소비자에게 이벤트를 전달합니다.
ProcessTraceAddBufferToBufferStream OpenTraceFromBufferStream에서 만든 처리 세션에 ETW 추적 버퍼를 제공합니다.
ProcessTraceBufferDecrementReference ProcessTraceBufferIncrementReference에서 추가한 버퍼에 대한 참조를 해제합니다.
ProcessTraceBufferIncrementReference 제공된 버퍼에서 BufferCallback 중에 호출되어 호출자가 이를 수행할 때까지 해제되지 않도록 합니다.
QueryAllTracesA ANSI(QueryAllTracesA) 함수(evntrace.h) 함수는 호출자가 쿼리할 수 있는 모든 이벤트 추적 세션에 대한 속성과 통계를 검색합니다.
QueryAllTracesW QueryAllTracesW(유니코드) 함수(evntrace.h) 함수는 호출자가 쿼리할 수 있는 모든 이벤트 추적 세션에 대한 속성과 통계를 검색합니다.

QueryTraceA ANSI(QueryTraceA) 함수(evntrace.h)는 지정된 이벤트 추적 세션에 대한 속성 설정 및 세션 통계를 검색합니다.
QueryTraceProcessingHandle OpenTrace에서 연 ETW 추적 처리 세션에 대한 정보를 검색합니다.
QueryTraceW QueryTraceW(유니코드) 함수(evntrace.h)는 지정된 이벤트 추적 세션에 대한 속성 설정 및 세션 통계를 검색합니다.
RegisterTraceGuidsA ANSI(RegisterTraceGuidsA) 함수(evntrace.h)는 사용되지 않는 함수이며 새 코드는 제공된 대안을 사용해야 합니다.
RegisterTraceGuidsW RegisterTraceGuidsW(유니코드) 함수(evntrace.h)는 사용되지 않는 함수이며 새 코드는 제공된 대안을 사용해야 합니다.
RemoveTraceCallback RemoveTraceCallback 함수는 EventCallback 함수가 이벤트 추적 클래스에 대한 이벤트를 수신하는 것을 중지합니다. 이 함수는 사용되지 않습니다.
SetTraceCallback SetTraceCallback 함수는 지정된 이벤트 추적 클래스에 대한 이벤트를 처리하는 EventCallback 함수를 지정합니다. 이 함수는 사용되지 않습니다.
StartTraceA StartTrace 함수는 이벤트 추적 세션을 시작합니다. (ANSI)
StartTraceW StartTrace 함수는 이벤트 추적 세션을 시작합니다. (유니코드)
StopTraceA STOPTraceA(ANSI) 함수(evntrace.h)는 지정된 이벤트 추적 세션을 중지합니다. ControlTrace 함수는 이 함수를 대체합니다.

StopTraceW StopTraceW(유니코드) 함수(evntrace.h)는 지정된 이벤트 추적 세션을 중지합니다. ControlTrace 함수는 이 함수를 대체합니다.
Traceevent RegisterTraceGuids 기반("클래식") 이벤트 공급자는 TraceEvent 함수를 사용하여 구조적 이벤트를 이벤트 추적 세션으로 보냅니다.
TraceEventInstance RegisterTraceGuids 기반("클래식") 이벤트 공급자는 TraceEventInstance 함수를 사용하여 인스턴스 식별자가 있는 이벤트 추적 세션에 구조적 이벤트를 보냅니다.
TraceMessage RegisterTraceGuids 기반("클래식") 이벤트 공급자는 TraceMessage 함수를 사용하여 TMF 기반 WPP(메시지 기반) 이벤트를 이벤트 추적 세션으로 보냅니다.
TraceMessageVa RegisterTraceGuids 기반("클래식") 이벤트 공급자는 TraceMessageVa 함수를 사용하여 메시지 기반(TMF 기반 WPP) 이벤트를 va_list 매개 변수를 사용하여 이벤트 추적 세션으로 보냅니다.
TraceQueryInformation 이벤트 추적 세션에 대한 정보를 제공합니다.
TraceSetInformation 이벤트 추적 세션 설정을 구성합니다.
UnregisterTraceGuids RegisterTraceGuids를 사용하여 등록된 "클래식"(Windows 2000 스타일) ETW 이벤트 추적 공급자를 등록 취소합니다.
UpdateTraceA ANSI(UpdateTraceA) 함수(evntrace.h)는 지정된 이벤트 추적 세션의 속성 설정을 업데이트합니다.
UpdateTraceW UpdateTraceW(유니코드) 함수(evntrace.h)는 지정된 이벤트 추적 세션의 속성 설정을 업데이트합니다.

콜백 함수

PETW_BUFFER_CALLBACK ProcessTrace에서 호출할 BufferCallback에 대한 함수 정의입니다.
PETW_BUFFER_COMPLETION_CALLBACK ProcessTraceAddBufferToBufferStream이 버퍼로 완료될 때 발생할 콜백에 대한 함수 정의입니다. 이 콜백은 일반적으로 버퍼를 적절하게 해제해야 합니다.
PEVENT_CALLBACK ETW 이벤트 소비자는 추적 처리 세션에서 이벤트를 수신하기 위해 이 콜백을 구현합니다. EventRecordCallback 콜백은 이 콜백을 대체합니다.
PEVENT_RECORD_CALLBACK ETW 이벤트 소비자는 추적 처리 세션에서 이벤트를 수신하기 위해 이 콜백을 구현합니다. PEVENT_RECORD_CALLBACK 형식은 이 콜백 함수에 대한 포인터를 정의합니다. EventRecordCallback은 애플리케이션 정의 함수 이름의 자리 표시자입니다.
PEVENT_TRACE_BUFFER_CALLBACKA ANSI(PEVENT_TRACE_BUFFER_CALLBACKA)(evntrace.h) 함수는 추적 처리 세션 중에 ETW가 보내는 이벤트의 각 버퍼에 대한 통계를 가져옵니다.
PEVENT_TRACE_BUFFER_CALLBACKW PEVENT_TRACE_BUFFER_CALLBACKW(유니코드)(evntrace.h) 함수는 추적 처리 세션 중에 ETW가 보내는 이벤트의 각 버퍼에 대한 통계를 가져옵니다.
WMIDPREQUEST RegisterTraceGuids 기반("클래식") 이벤트 공급자는 컨트롤러로부터 알림을 수신하도록 이 함수를 구현합니다. WMIDPREQUEST 형식은 이 콜백 함수에 대한 포인터를 정의합니다. ControlCallback은 애플리케이션 정의 함수 이름의 자리 표시자입니다.

구조체

CLASSIC_EVENT_ID 호출 스택 추적을 사용하도록 설정할 커널 이벤트를 식별합니다.

ENABLE_TRACE_PARAMETERS EnableTraceEx2를 통해 공급자를 사용하도록 설정하는 데 사용되는 정보를 포함합니다.
ENABLE_TRACE_PARAMETERS_V1 EnableTraceEx2를 통해 공급자를 사용하도록 설정하는 데 사용되는 정보를 포함합니다. 이 구조체는 사용되지 않습니다.
ETW_BUFFER_CALLBACK_INFORMATION BufferCallback에 *ConsumerInfo* 매개 변수로 제공되며 현재 처리 세션에 대한 세부 정보를 제공합니다.
ETW_BUFFER_CONTEXT 이벤트에 대한 컨텍스트 정보를 제공합니다.
ETW_BUFFER_HEADER ETW 버퍼의 헤더 구조입니다.
ETW_OPEN_TRACE_OPTIONS OpenTraceFromBufferStream, OpenTraceFromFile, OpenTraceFromRealTimeLogger, OpenTraceFromRealTimeLoggerWithAllocationOptions 함수에 대한 구성 매개 변수를 제공합니다.
ETW_TRACE_PARTITION_INFORMATION ETW 추적에서 가져온 파티션 정보를 포함합니다.
EVENT_INSTANCE_HEADER EVENT_INSTANCE_HEADER 구조에는 TraceEventInstance에서 작성한 모든 이벤트에 공통적인 표준 이벤트 추적 정보가 포함되어 있습니다.
EVENT_INSTANCE_INFO EVENT_INSTANCE_INFO 구조체는 TraceEventInstance에 대해 등록된 이벤트 추적 클래스에 고유한 트랜잭션 식별자를 매핑합니다.
EVENT_TRACE EVENT_TRACE 구조는 이벤트 추적 소비자에게 이벤트 정보를 전달하는 데 사용됩니다.
EVENT_TRACE_HEADER EVENT_TRACE_HEADER 구조에는 TraceEvent에서 작성한 모든 이벤트에 공통적인 표준 이벤트 추적 정보가 포함되어 있습니다.

EVENT_TRACE_LOGFILEA ANSI(EVENT_TRACE_LOGFILEA) 구조체(evntrace.h)는 추적 데이터 원본에 대한 정보를 저장합니다.
EVENT_TRACE_LOGFILEW EVENT_TRACE_LOGFILEW(유니코드) 구조체(evntrace.h)는 추적 데이터 원본에 대한 정보를 저장합니다.
EVENT_TRACE_PROPERTIES EVENT_TRACE_PROPERTIES 구조는 이벤트 추적 세션에 대한 정보를 포함하며 StartTrace 및 ControlTrace와 같은 API와 함께 사용됩니다.
EVENT_TRACE_PROPERTIES_V2 EVENT_TRACE_PROPERTIES_V2 구조는 이벤트 추적 세션에 대한 정보를 포함하며 StartTrace 및 ControlTrace와 같은 API와 함께 사용됩니다.
MOF_FIELD MOF_FIELD 구조를 사용하여 이벤트 데이터를 EVENT_TRACE_HEADER 또는 EVENT_INSTANCE_HEADER 구조에 추가할 수 있습니다.
TRACE_ENABLE_INFO 세션과 세션이 공급자를 사용하도록 설정하는 데 사용한 정보를 정의합니다.
TRACE_GUID_INFO EnumerateTraceGuidsEx에서 반환됩니다. 공급자를 사용하도록 설정한 세션 목록에 대한 헤더를 정의합니다.
TRACE_GUID_PROPERTIES EnumerateTraceGuids에서 반환됩니다. 이벤트 추적 공급자에 대한 정보를 포함합니다.
TRACE_GUID_REGISTRATION RegisterTraceGuids와 함께 이벤트 추적 클래스를 등록하는 데 사용됩니다.
TRACE_LOGFILE_HEADER TRACE_LOGFILE_HEADER 구조에는 이벤트 추적 세션 및 해당 이벤트에 대한 정보가 포함됩니다.
TRACE_PERIODIC_CAPTURE_STATE_INFO TraceQueryInformation 및 TraceSetInformation과 함께 사용하여 주기적인 캡처 상태와 관련된 정보를 얻거나 설정합니다.

TRACE_PROVIDER_INSTANCE_INFO 공급자 GUID의 인스턴스를 정의합니다.
TRACE_VERSION_INFO TraceLogging 세션의 버전 정보를 결정합니다.

열거형

ETW_PROCESS_HANDLE_INFO_TYPE 추적 처리 세션에서 수행할 작업을 지정합니다.
ETW_PROCESS_TRACE_MODES 지원되는 프로세스 추적 모드를 지정합니다.
TRACE_QUERY_INFO_CLASS EnumerateTraceGuidsEx 및 TraceSetInformation과 함께 추적 정보 유형을 지정하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

exdisp.h 헤더

아티클2023. 08. 23.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보
- Windows 셸

exdisp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IShellWindows 열린 Shell 창의 컬렉션에 대한 액세스를 제공합니다.
IWebBrowser2 WebBrowser 컨트롤(Microsoft ActiveX 컨트롤)에 의해 구현되거나 InternetExplorer 애플리케이션(OLE Automation)의 instance 의해 구현되는 메서드를 노출합니다.

열거형

ShellWindowFindWindowOptions 셸 창 컬렉션에서 창을 찾는 옵션을 지정합니다.
ShellWindowTypeConstants Shell 창의 형식을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

expandedresources.h 헤더

아티클2023. 08. 23.

이 헤더는 게임 모드에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 게임 모드

expandedresources.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

GetExpandedResourceExclusiveCpuCount 게임 모드에서 앱에서 사용할 수 있는 전용 CPU 집합의 예상 수를 가져옵니다.
HasExpandedResources 현재 리소스 상태(즉, 앱이 게임 모드 또는 공유 모드에서 실행 중인지 여부)를 가져옵니다.
ReleaseExclusiveCpuSets CPU 독점성을 옵트아웃하여 앱이 모든 코어에 액세스할 수 있도록 하지만 다른 프로세스와 공유해야 하는 비용이 듭니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

exposeenums2managed.h 헤더

아티클2023. 08. 23.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

exposeenums2managed.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

태그 shim 데이터베이스의 항목을 식별합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

faxcom.h 헤더

아티클2023. 08. 23.

이 헤더는 팩스 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 팩스 서비스

faxcom.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IFaxDoc IFaxDoc 이중 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 문서 및 표지를 전송하는 데 사용됩니다.
IFaxJob IFaxJob 이중 인터페이스는 팩스 클라이언트 애플리케이션에서 연결된 팩스 서버의 팩스 작업에 대한 정보에 액세스하는 데 사용됩니다. FaxJobs 개체는 FaxJob 개체의 컬렉션입니다.
IFaxJobs IFaxJobs 이중 인터페이스는 팩스 클라이언트 애플리케이션에서 FaxServer 개체에서 파생된 FaxJob 개체에 액세스하는 데 사용됩니다. 인터페이스는 연결된 팩스 서버와 연결된 팩스 작업을 열거합니다.
IFaxPort IFaxPort 이중 인터페이스는 팩스 클라이언트 애플리케이션에서 연결된 팩스 서버의 팩스 포트에 대한 구성 정보에 액세스하는 데 사용됩니다.
IFaxPorts IFaxPorts 이중 인터페이스는 팩스 클라이언트 애플리케이션에서 FaxServer 개체에서 파생된 FaxPort 개체에 액세스하는 데 사용됩니다. 인터페이스는 활성 팩스 서버에 대한 연결에 대한 포트 구성 정보를 열거합니다.
IFaxRoutingMethod IFaxRoutingMethod 이중 인터페이스는 팩스 클라이언트 애플리케이션에서 연결된 팩스 서버의 팩스 포트에 대한 팩스 라우팅 구성 정보를 검색하는 데 사용됩니다.
IFaxRoutingMethods

IFaxRoutingMethods 이중 인터페이스는 팩스 클라이언트 애플리케이션에서 FaxPort 개체에서 파생된 FaxRoutingMethod 개체에 액세스하는 데 사용됩니다.
IFaxServer IFaxServer 이중 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 서비스에 대한 연결을 관리하는 데 사용됩니다.
IFaxStatus FaxStatus 이중 인터페이스는 팩스 클라이언트 애플리케이션에서 연결된 팩스 서버의 특정 포트에 대한 상태 정보를 검색하는 데 사용됩니다.
IFaxTiff IFaxTiff 이중 인터페이스는 팩스 클라이언트 애플리케이션에서 FaxTiff 개체에 대한 정보를 검색하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

faxcomex.h 헤더

아티클2023. 08. 23.

이 헤더는 팩스 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 팩스 서비스

faxcomex.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

_IFaxAccountNotify 팩스 서비스에서 호출하여 특정 팩스 계정에 대한 이벤트 알림을 보냅니다. 이 속성은 이벤트 알림을 보냅니다. 이벤트에는 들어오는 작업 큐 및 나가는 작업 큐에 대한 변경 내용 및 수신 및 나가는 보관 파일의 변경 내용이 포함됩니다.
_IFaxServerNotify2 IFaxServerNotify2 인터페이스는 팩스 알림에 사용됩니다.
IFaxAccount 팩스 서버의 팩스 계정을 나타냅니다.
IFaxAccountFolders 들어오고 나가는 팩스 큐 및 팩스 보관 파일에 대한 액세스를 제공합니다.
IFaxAccountIncomingArchive 팩스 클라이언트 애플리케이션에서 수신된 인바운드 팩스 메시지의 특정 팩스 계정 보관 파일에 액세스하는 데 사용됩니다. 이 인터페이스를 사용하여 메시지를 검색하고 보관 파일의 크기를 가져옵니다.
IFaxAccountIncomingQueue 팩스 클라이언트 애플리케이션에서 특정 팩스 계정에 대한 작업 큐에서 인바운드 팩스 작업(FaxIncomingJobs 개체)을 검색하는 데 사용됩니다.
IFaxAccountNotify 팩스 서비스에서 호출하여 특정 팩스 계정에 대한 이벤트 알림을 보냅니다. 이 속성은 이벤트 알림을 보냅니다. 이벤트에는 들어오는 작업 큐 및 나가는 작업 큐에 대한 변경 내용 및 수신 및 나가는 보관 파일의 변경 내용이 포함됩니다. (IIFaxAccountNotify)

IFaxAccountOutgoingArchive 팩스 클라이언트 애플리케이션에서 성공적으로 전송된 아웃바운드 팩스 메시지의 지정된 팩스 계정 보관 파일에 액세스하는 데 사용됩니다. 이 인터페이스를 사용하여 메시지를 검색하고 보관 파일의 크기를 가져옵니다.
IFaxAccountOutgoingQueue 팩스 클라이언트 애플리케이션에서 특정 팩스 계정에 대한 작업 큐에서 아웃바운드 팩스 작업(FaxOutgoingJobs 개체)을 검색하는 데 사용됩니다.
IFaxAccounts 팩스 서버의 팩스 계정 컬렉션을 나타냅니다. 계정을 열거하고, 특정 계정을 검색하고, 총 계정 수를 보고하기 위한 메서드와 속성을 제공합니다.
IFaxAccountSet 팩스 계정 추가, 제거 및 검색을 포함하여 팩스 계정 관리를 위한 메서드를 제공합니다.
IFaxActivity IFaxActivity 인터페이스는 읽기 전용 구성 개체를 정의합니다.
IFaxActivityLogging IFaxActivityLogging 인터페이스는 팩스 클라이언트 애플리케이션에서 활동 로깅 옵션을 검색하고 설정하는 데 사용하는 구성 개체를 정의합니다.
IFaxConfiguration 팩스 서비스에 대한 구성 옵션을 제공하는 다양한 메서드를 정의합니다.
IFaxDevice IFaxDevice 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 디바이스 정보를 검색 및 설정하고 팩스 디바이스와 연결된 팩스 라우팅 메서드를 추가 및 제거하는 데 사용하는 구성 개체를 정의합니다.
IFaxDeviceIds IFaxDeviceIds 인터페이스는 팩스 클라이언트 애플리케이션에서 FaxOutboundRoutingGroup 개체와 연결된 정렬된 팩스 디바이스 ID를 열거하는 데 사용하는 구성 컬렉션을 정의합니다.
IFaxDeviceProvider IFaxDeviceProvider 인터페이스는 팩스 서비스에 등록된 FSP(팩스 서비스 공급자)에 대한 정보를 검색하기 위해 팩스 클라이언트 애플리케이션에서 사용하는 구성 개체를 정의합니다.

IFaxDeviceProviders IFaxDeviceProviders 인터페이스는 연결된 팩스 서버의 팩스 디바이스 공급자를 포함하는 구성 컬렉션을 정의합니다.
IFaxDevices IFaxDevices 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 디바이스를 관리하는 데 사용하는 컬렉션을 정의합니다. 여기서 각 디바이스는 FaxDevice 개체로 표시됩니다.
IFaxDocument IFaxDocument 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 문서를 작성하고 처리를 위해 팩스 서비스에 제출하는 데 사용하는 메시징 개체를 정의합니다.
IFaxDocument2 팩스 클라이언트 애플리케이션에서 팩스 문서를 작성하고 처리를 위해 팩스 서비스에 제출하는 데 사용하는 메시징 개체를 정의합니다.
IFaxEventLogging IFaxEventLogging 인터페이스는 팩스 서비스에서 사용하는 이벤트 로깅 범주를 구성하기 위해 팩스 클라이언트 애플리케이션에서 사용하는 구성 개체를 정의합니다.
IFaxFolders IFaxFolders 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 서버의 폴더, 큐에 대기 중인 작업 및 보관된 메시지에 액세스하는 데 사용하는 구성 개체를 정의합니다.
IFaxInboundRouting IFaxInboundRouting 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 서비스에 등록된 인바운드 라우팅 확장에 액세스하는 데 사용되는 구성 개체를 정의하며, FaxInboundRoutingExtensions 개체로 표현되며 FaxInboundRoutingMethods 개체로 표현되는 확장이 노출하는 라우팅 메서드를 정의합니다.
IFaxInboundRoutingExtension IFaxInboundRoutingExtension 인터페이스는 팩스 서비스에 등록된 팩스 라우팅 확장에 대한 정보를 검색하기 위해 팩스 클라이언트 애플리케이션에서 사용하는 구성 개체를 정의합니다.
IFaxInboundRoutingExtensions IFaxInboundRoutingExtensions 인터페이스는 팩스 서비스에 등록된 인바운드 팩스 라우팅 확장을 관리하기 위해 팩스 클라이언트 애플리케이션에서 사용하는 구성 컬렉션을 정의합니다.
IFaxInboundRoutingMethod

IFaxInboundRoutingMethod 인터페이스는 팩스 클라이언트 애플리케이션이 연결된 팩스 서버에서 개별 팩스 인바운드 라우팅 방법에 대한 정보를 검색하는 데 사용하는 구성 개체를 정의합니다.
IFaxInboundRoutingMethods IFaxInboundRoutingMethods 인터페이스는 팩스 클라이언트 애플리케이션에서 정렬된 인바운드 팩스 라우팅 메서드를 관리하는 데 사용하는 구성 컬렉션을 정의합니다.
IFaxIncomingArchive IFaxIncomingArchive 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 서비스에서 성공적으로 수신한 인바운드 팩스 메시지의 보관 파일에 액세스하고 구성하는 데 사용됩니다.
IFaxIncomingJob IFaxIncomingJob 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 서버의 큐에서 들어오는 팩스 작업에 대한 정보를 검색하는 데 사용됩니다.
IFaxIncomingJobs IFaxIncomingJobs 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 서버의 작업 큐에서 인바운드 팩스 작업을 관리하는 데 사용됩니다. 들어오는 각 작업은 FaxIncomingJob 개체로 표시됩니다.
IFaxIncomingMessage 팩스 클라이언트 애플리케이션에서 인바운드 팩스 보관에서 받은 팩스 메시지에 대한 정보를 검색하는 데 사용됩니다. (IFaxIncomingMessage)
IFaxIncomingMessage2 팩스 클라이언트 애플리케이션에서 인바운드 팩스 보관에서 받은 팩스 메시지에 대한 정보를 검색하는 데 사용됩니다. (IFaxIncomingMessage2)
IFaxIncomingMessageIterator IFaxIncomingMessageIterator 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 서비스가 성공적으로 수신한 인바운드 팩스 메시지의 보관을 통해 이동하는 데 사용됩니다.
IFaxIncomingQueue IFaxIncomingQueue 인터페이스는 팩스 클라이언트 애플리케이션에서 작업 큐의 인바운드 팩스 작업(FaxIncomingJobs 개체)을 관리하는 데 사용됩니다. 개체에는 팩스 작업 큐에서 인바운드 팩스를 차단하는 메서드도 포함됩니다.
IFaxJobStatus IFaxJobStatus 인터페이스는 알림 및 작업의 동적 정보를 유지하는 데 사용됩니다.

IFaxLoggingOptions IFaxLoggingOptions 인터페이스는 팩스 클라이언트 애플리케이션에서 이벤트 로깅 범주 및 팩스 서비스에서 사용하는 활동 로깅 옵션에 액세스하고 구성하는 데 사용됩니다.
IFaxOutboundRouting IFaxOutboundRouting 인터페이스는 아웃바운드 라우팅 그룹(IFaxOutboundRoutingGroups 인터페이스) 및 아웃바운드 라우팅 규칙(IFaxOutboundRoutingRules 인터페이스)을 구성하기 위해 팩스 클라이언트 애플리케이션에서 사용하는 구성 개체를 정의합니다.
IFaxOutboundRoutingGroup IFaxOutboundRoutingGroup 인터페이스는 팩스 클라이언트 애플리케이션에서 개별 팩스 아웃바운드 라우팅 그룹에 대한 정보를 검색하는 데 사용되는 구성 개체를 설명합니다.
IFaxOutboundRoutingGroups IFaxOutboundRoutingGroups 인터페이스는 팩스 클라이언트 애플리케이션에서 IFaxOutboundRoutingGroup 인터페이스로 표시되는 팩스 아웃바운드 라우팅 그룹을 관리하는 데 사용하는 구성 컬렉션을 설명합니다.
IFaxOutboundRoutingRule IFaxOutboundRoutingRule 인터페이스는 팩스 클라이언트 애플리케이션에서 개별 팩스 아웃바운드 라우팅 규칙에 대한 정보를 설정하고 검색하는 데 사용하는 구성 개체를 설명합니다.
IFaxOutboundRoutingRules IFaxOutboundRoutingRules 인터페이스는 팩스 아웃바운드 라우팅 규칙을 관리하기 위해 팩스 클라이언트 애플리케이션에서 사용하는 구성 컬렉션을 설명합니다.
IFaxOutgoingArchive IFaxOutgoingArchive 인터페이스는 팩스 서비스에서 성공적으로 전송된 아웃바운드 팩스 메시지의 보관 파일에 액세스하고 구성하기 위해 팩스 클라이언트 애플리케이션에서 사용하는 구성 개체를 설명합니다.
IFaxOutgoingJob IFaxOutgoingJob 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 서버 큐에서 보내는 팩스 작업에 대한 정보를 검색하는 데 사용되는 개체를 설명합니다.
IFaxOutgoingJob2 팩스 클라이언트 애플리케이션에서 팩스 서버 큐에서 보내는 팩스 작업에 대한 정보를 검색하는 데 사용되는 개체를 설명합니다.

IFaxOutgoingJobs IFaxOutgoingJobs 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 서버의 작업 큐에서 아웃바운드 팩스 작업을 관리하는 데 사용되는 메시징 컬렉션을 설명합니다. 각 아웃바운드 작업은 IFaxOutgoingJob 인터페이스로 표시됩니다.
IFaxOutgoingMessage IFaxOutgoingMessage 인터페이스는 아웃바운드 팩스 보관에서 팩스 메시지에 대한 정보를 검색하기 위해 팩스 클라이언트 애플리케이션에서 사용하는 개체를 설명합니다.
IFaxOutgoingMessage2 팩스 클라이언트 애플리케이션에서 아웃바운드 팩스 보관에서 보낸 팩스 메시지에 대한 정보를 검색하는 데 사용됩니다.
IFaxOutgoingMessageIterator IFaxOutgoingMessageIterator 인터페이스는 팩스 서비스가 성공적으로 전송한 팩스 메시지의 보관을 이동하기 위해 팩스 클라이언트 애플리케이션에서 FaxOutgoingMessage 개체로 나타내는 개체를 설명합니다.
IFaxOutgoingQueue IFaxOutgoingQueue 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 서버의 아웃바운드 팩스 큐에서 구성 매개 변수를 설정하고 검색하는 데 사용하는 FaxOutgoingQueue 구성 개체를 정의합니다.
IFaxReceiptOptions IFaxReceiptOptions 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 전송에 대한 배달 영수증을 보내는 데 사용하는 영수증 구성을 설정하고 검색하는 데 사용하는 FaxReceiptOptions 구성 개체를 정의합니다.
IFaxRecipient IFaxRecipient 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 수신자의 개인 정보를 검색하고 설정하는 데 사용되는 FaxRecipient 메시징 개체를 정의합니다.
IFaxRecipients IFaxRecipients 인터페이스는 팩스 클라이언트 애플리케이션에서 단일 팩스 문서의 받는 사람을 나타내는 팩스 수신자 개체(FaxRecipient)를 관리하는 데 사용되는 FaxRecipients 메시징 컬렉션을 정의합니다.
IFaxSecurity

IFaxSecurity 구성 개체는 팩스 클라이언트 애플리케이션에서 팩스 서버의 보안을 구성하는 데 사용되며 호출 애플리케이션이 팩스 서버에 대한 보안 설명자를 설정하고 검색하도록 허용합니다.
IFaxSecurity2 팩스 클라이언트 애플리케이션에서 팩스 서버의 보안을 구성하는 데 사용됩니다. 또한 호출 애플리케이션이 팩스 서버에 대한 보안 설명자를 설정하고 검색할 수 있도록 허용합니다.
IFaxSender IFaxSender 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 보낸 사람 정보를 검색하고 설정하는 데 사용하는 메시징 개체를 정의합니다. 또한 개체에는 보낸 사람 데이터를 저장하고 로컬 레지스트리에서 보낸 사람 데이터를 검색하는 메서드도 포함됩니다.
IFaxServer IFaxServer 인터페이스는 팩스 클라이언트 애플리케이션에서 팩스 서비스에 대한 연결을 관리하는 데 사용하는 메시징 컬렉션을 설명합니다.
IFaxServer2 IFaxServer 인터페이스의 모든 기능을 상속하고 서버 구성, 계정 관리, 보안 개체 및 현재 계정에 대한 읽기 전용 속성을 추가합니다.
IFaxServerNotify2 IFaxServerNotify2 인터페이스는 팩스 알림에 사용됩니다. (IIFaxServerNotify2)

열거형

FAX_ACCESS_RIGHTS_ENUM FAX_ACCESS_RIGHTS_ENUM 열거형은 팩스 서버에 대한 액세스 권한을 정의합니다.
FAX_ACCESS_RIGHTS_ENUM_2 팩스 서버에 대한 액세스 권한을 정의합니다.
FAX_ACCOUNT_EVENTS_TYPE_ENUM 서버가 수신 대기 클라이언트에 보내는 특정 계정의 이벤트 알림 유형을 지정합니다.
FAX_COVERPAGE_TYPE_ENUM

FAX_COVERPAGE_TYPE_ENUM 열거형은 표지 템플릿 파일이 로컬 컴퓨터 표지인지 아니면 서버 기반 표지인지를 정의합니다. 또한 파일이 사용되지 않음을 지정할 수도 있습니다.
FAX_DEVICE_RECEIVE_MODE_ENUM FAX_DEVICE_RECEIVE_MODE_ENUM 열거형은 디바이스가 들어오는 호출에 응답하는 방법을 정의합니다.
FAX_GROUP_STATUS_ENUM FAX_GROUP_STATUS_ENUM 열거형은 아웃바운드 라우팅 그룹에 대한 상태 형식을 정의합니다.
FAX_JOB_EXTENDED_STATUS_ENUM FAX_JOB_EXTENDED_STATUS_ENUM 열거형은 팩스 작업에 대한 확장된 상태 값을 정의합니다.
FAX_JOB_OPERATIONS_ENUM FAX_JOB_OPERATIONS_ENUM 열거형은 팩스 작업에서 수행할 수 있는 작업을 정의합니다. 이 열거형의 멤버는 비트 값이며 조합하여 사용할 수 있습니다.
FAX_JOB_STATUS_ENUM FAX_JOB_STATUS_ENUM 열거형은 팩스 작업의 상태 값을 정의합니다. 참고 fjsPAUSED 및 fjsNOLINE 멤버는 한정자입니다. 이 열거형의 다른 멤버와 함께 사용할 수 있습니다.
FAX_JOB_TYPE_ENUM FAX_JOB_TYPE_ENUM 열거형은 팩스 작업 유형을 정의합니다.
FAX_LOG_LEVEL_ENUM FAX_LOG_LEVEL_ENUM 열거형은 로깅 범주에 대한 이벤트 로깅 수준을 정의합니다.
FAX_PRIORITY_TYPE_ENUM FAX_PRIORITY_TYPE_ENUM 열거형은 아웃바운드 팩스의 우선 순위 유형을 정의합니다.
FAX_PROVIDER_STATUS_ENUM FAX_PROVIDER_STATUS_ENUM 열거형은 팩스 확장(FSP(팩스 서비스 공급자) 또는 팩스 인바운드 라우팅 확장)에 대한 상태 값을 정의합니다.
FAX_RECEIPT_TYPE_ENUM FAX_RECEIPT_TYPE_ENUM 열거형은 아웃바운드 팩스에 대한 배달 보고서 유형(배달 확인 형식)을 정의합니다. 이 열거형의 멤버는 비트 값이며 조합하여 사용할 수 있습니다.
FAX_ROUTING_RULE_CODE_ENUM

FAX_ROUTING_RULE_CODE_ENUM 열거형은 아웃바운드 라우팅에 대한 규칙을 정의합니다.
FAX_RULE_STATUS_ENUM FAX_RULE_STATUS_ENUM 열거형은 아웃바운드 라우팅 규칙에 대한 상태 형식을 정의합니다.
FAX_SCHEDULE_TYPE_ENUM FAX_SCHEDULE_TYPE_ENUM 열거형은 아웃바운드 팩스 예약 유형을 정의합니다.
FAX_SERVER_APIVERSION_ENUM FAX_SERVER_APIVERSION_ENUM 열거형은 팩스 API의 버전을 정의합니다. 아래 값은 지정한 것보다 이전 버전의 팩스 서비스에서 지원되지 않습니다.
FAX_SERVER_EVENTS_TYPE_ENUM FAX_SERVER_EVENTS_TYPE_ENUM 열거형은 팩스 서비스가 이벤트를 수신 대기하는 클라이언트 애플리케이션에 보내는 이벤트 유형을 정의합니다. 이 열거형의 멤버는 비트 값이며 조합하여 사용할 수 있습니다.
FAX_SMTP_AUTHENTICATION_TYPE_ENUM FAX_SMTP_AUTHENTICATION_TYPE_ENUM 열거형은 이메일을 통해 전송된 배달 영수증에 대한 구성 옵션을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

faxdev.h 헤더

아티클2023. 08. 23.

이 헤더는 팩스 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 팩스 서비스

faxdev.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

FaxDevAbortOperation 팩스 서비스는 FaxDevAbortOperation 함수를 호출하여 팩스 서비스 공급자(FSP)가 FaxHandle 매개 변수로 지정된 팩스 작업에 대한 활성 팩스 작업을 종료하도록 요청합니다. 각 FSP는 FaxDevAbortOperation 함수를 내보내야 합니다.
FaxDevEndJob 팩스 서비스는 팩스 작업의 마지막 팩스 작업 후에 FaxDevEndJob 함수를 호출합니다. 각 팩스 서비스 공급자(FSP)는 FaxDevEndJob 함수를 내보내야 합니다.
FaxDevInitialize 팩스 서비스는 FSP(팩스 서비스 공급자) DLL을 로드한 후 서비스가 시작될 때마다 FaxDevInitialize 함수를 호출합니다. 각 FSP는 FaxDevInitialize 함수를 내보내야 합니다.
FaxDevReceive 팩스 서비스는 FaxDevReceive 함수를 호출하여 들어오는 팩스 전송을 팩스 서비스 공급자(FSP)에 신호를 보냅니다. 각 FSP는 FaxDevReceive 함수를 내보내야 합니다.
FaxDevReportStatus 팩스 서비스는 FaxDevReportStatus 함수를 호출하여 FSP(팩스 서비스 공급자)에게 개별 활성 팩스 작업에 대한 상태 정보 또는 팩스 작업 실패 후 상태 정보를 쿼리합니다.
FaxDevSend 팩스 서비스는 FaxDevSend 함수를 호출하여 팩스 서비스 공급자(FSP)에게 송신 팩스 전송을 시작해야 한다는 신호를 보냅니다. 각 FSP는 FaxDevSend 함수를 내보내야 합니다.
FaxDevShutdown 팩스 서비스는 FaxDevShutdown 함수를 호출하여 FSP(팩스 서비스 공급자)에게 서비스가 FSP의

DLL을 언로드하려고 했음을 알립니다. FaxDevShutdown은 FaxDevInitialize 함수에 의해 할당된 전역 리소스를 해제합니다.
FaxDevStartJob 팩스 서비스는 FaxDevStartJob 함수를 호출하여 새 팩스 작업을 초기화합니다. 팩스 서비스는 또한 FaxDevStartJob을 호출하여 각 팩스 작업의 시작을 팩스 서비스 공급자(FSP)에 신호를 보냅니다. 각 FSP는 FaxDevStartJob 함수를 내보내야 합니다.
FaxDevVirtualDeviceCreation 팩스 서비스는 초기화 중에 FaxDevVirtualDeviceCreation 함수를 호출하여 FSP(팩스 서비스 공급자)가 가상 팩스 디바이스를 표시할 수 있도록 합니다.

콜백 함수

PFAX_LINECALLBACK FaxLineCallback 함수는 팩스 서비스가 전화 통신 애플리케이션 프로그래밍 인터페이스(TAPI) 이벤트를 팩스 서비스 공급자(FSP)에 전달하기 위해 호출하는 애플리케이션 정의 또는 라이브러리 정의 콜백 함수입니다.
PFAX_SEND_CALLBACK FaxSendCallback 함수는 FSP(팩스 서비스 공급자)가 팩스 서비스에 발신 팩스 호출이 진행 중임을 알리기 위해 호출하는 애플리케이션 정의 또는 라이브러리 정의 콜백 함수입니다.

구조체

FAX_DEV_STATUS FAX_DEV_STATUS 구조에는 개별 활성 팩스 작업에 대한 상태 및 식별 정보가 포함됩니다.
FAX_RECEIVE FAX_RECEIVE 구조에는 인바운드 팩스 문서에 대한 정보가 포함되어 있습니다. 이 정보에는 팩스 데이터 스트림을 받을 파일의 이름과 수신 디바이스의 이름과 전화 번호가 포함됩니다.
FAX_SEND FAX_SEND 구조에는 아웃바운드 팩스 문서에 대한 정보가 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

faxext.h 헤더

아티클2023. 08. 23.

이 헤더는 팩스 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 팩스 서비스

faxext.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

FaxExtConfigChange FaxExtConfigChange 콜백 함수는 팩스 확장 DLL로 정의된 함수 이름의 자리 표시자입니다. 팩스 확장 DLL은 이 함수를 노출해서는 안 됩니다.
FaxExtFreeBuffer FaxExtFreeBuffer 콜백 함수는 FaxExtGetData 함수를 성공적으로 호출하여 이전에 할당한 메모리의 할당을 취소합니다.
FaxExtGetData FaxExtGetData 콜백 함수는 특정 디바이스 및 GUID에 대한 구성 데이터를 검색합니다.
FaxExtInitializeConfig 팩스 서비스는 FaxExtInitializeConfig 함수를 호출하여 팩스 확장 DLL을 초기화합니다. 서비스는 다른 팩스 확장 초기화 함수를 호출하기 전에 이 함수를 호출합니다.
FaxExtRegisterForEvents FaxExtRegisterForEvents 콜백 함수는 특정 디바이스 및 GUID와 관련된 구성 데이터 변경에 대한 알림을 위해 팩스 확장 DLL을 등록합니다.
FaxExtSetData FaxExtSetData 함수는 특정 디바이스 및 GUID에 대한 구성 데이터를 설정합니다.
FaxExtUnregisterForEvents FaxExtUnregisterForEvents 콜백 함수는 특정 디바이스 및 GUID와 관련된 구성 데이터 변경 내용에 대한 알림을 위해 팩스 확장 DLL의 등록을 취소합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

faxroute.h 헤더

아티클2023. 08. 23.

이 헤더는 팩스 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 팩스 서비스

faxroute.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

FaxRouteDeviceChangeNotification 팩스 서비스는 FaxRouteDeviceChangeNotification 함수를 호출하여 팩스 라우팅 확장 DLL에 팩스 디바이스가 팩스 서버에서 제거되었거나 새 팩스 장치가 설치되었음을 알릴 수 있습니다.
FaxRouteDeviceEnable FaxRouteDeviceEnable 함수를 사용하면 팩스 라우팅 확장 DLL이 특정 팩스 디바이스에 대한 팩스 라우팅 방법을 쿼리, 사용 또는 사용하지 않도록 설정할 수 있습니다. 각 팩스 라우팅 확장 프로그램은 FaxRouteDeviceEnable 함수를 내보내야 합니다.
FaxRouteGetRoutingInfo FaxRouteGetRoutingInfo 함수는 팩스 라우팅 확장 DLL을 쿼리하여 특정 팩스 디바이스에 대한 라우팅 구성 데이터를 가져옵니다. 각 팩스 라우팅 확장 프로그램 DLL은 FaxRouteGetRoutingInfo 함수를 내보내야 합니다.
FaxRouteInitialize 팩스 서비스는 서비스가 시작될 때마다 FaxRouteInitialize 함수를 한 번 호출하여 팩스 라우팅 확장 DLL을 초기화합니다. 각 팩스 라우팅 확장 프로그램 DLL은 FaxRouteInitialize 함수를 내보내야 합니다.
FaxRouteSetRoutingInfo FaxRouteSetRoutingInfo 함수는 특정 팩스 디바이스에 대한 라우팅 구성 데이터를 수정합니다. 각 팩스 라우팅 확장 프로그램 DLL은 FaxRouteSetRoutingInfo 함수를 내보내야 합니다.

콜백 함수

PFAXROUTEADDFILE 팩스 라우팅 메서드는 FaxRouteAddFile 콜백 함수를 호출하여 받은 팩스 문서와 연결된 팩스 파일 목록에 파일을 추가합니다.
PFAXROUTEDELETEFILE 팩스 라우팅 메서드는 FaxRouteDeleteFile 콜백 함수를 호출하여 받은 팩스 문서와 연결된 팩스 파일 목록에서 파일을 삭제합니다.
PFAXROUTEENUMFILE FaxRouteEnumFile 콜백 함수는 받은 팩스 문서와 연결된 팩스 파일 목록에서 파일 이름을 받습니다.
PFAXROUTEENUMFILES 팩스 라우팅 메서드는 FaxRouteEnumFiles 콜백 함수를 호출하여 받은 팩스 문서와 연결된 팩스 파일 목록의 파일을 열거합니다.
PFAXROUTEGETFILE 팩스 라우팅 메서드는 FaxRouteGetFile 콜백 함수를 호출하여 받은 팩스 문서와 연결된 팩스 파일 목록에서 특정 파일의 이름을 검색합니다.
PFAXROUTEMETHOD FaxRouteMethod 함수는 팩스 라우팅 확장 DLL로 정의된 함수 이름의 자리 표시자입니다. 이 함수는 정의된 팩스 라우팅 프로시저를 실행합니다.
PFAXROUTEMODIFYROUTINGDATA 팩스 라우팅 메서드는 FaxRouteModifyRoutingData 콜백 함수를 호출하여 후속 팩스 라우팅 메서드에 대한 라우팅 데이터를 수정합니다.

구조체

FAX_ROUTE FAX_ROUTE 구조에는 받은 팩스 문서에 대한 정보가 포함됩니다. 팩스 서비스는 FaxRouteMethod 함수에 대한 호출에서 구조를 팩스 라우팅 메서드로 전달합니다.
FAX_ROUTE_CALLBACKROUTINES

FAX_ROUTE_CALLBACKROUTINES 구조에는 팩스 서비스에서 제공하는 콜백 함수에 대한 포인터가 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

fci.h 헤더

아티클2023. 08. 23.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

fci.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

FCIAddFile FCIAddFile은 생성 중인 캐비닛에 파일을 추가합니다.
FCI만들기 FCICreate 함수는 FCI 컨텍스트를 만듭니다.
FCIDestroy FCIDestroy 함수는 열려 있는 FCI 컨텍스트를 삭제하여 컨텍스트와 연결된 메모리 및 임시 파일을 해제합니다.
FCIFlushCabinet FCIFlushCabinet 함수는 현재 캐비닛을 완료합니다.
FCIFlushFolder FCIFlushFolder 함수는 생성 중인 현재 폴더를 즉시 완료하도록 강제합니다.
FNFCIALLOC FNFCIALLOC는 애플리케이션 정의 콜백 함수가 FCI 컨텍스트 내에서 메모리를 할당하는 선언을 제공합니다.
FNFCICLOSE FNFCICLOSE 매크로는 애플리케이션 정의 콜백 함수가 FCI 컨텍스트에서 파일을 닫는 선언을 제공합니다.
FNFCIDELETE

FNFCIDELETE 매크로는 애플리케이션 정의 콜백 함수가 FCI 컨텍스트에서 파일을 삭제하는 선언을 제공합니다.
FNFCIFILEPLACED FNFCIFILEPLACED 매크로는 파일이 캐비닛에 배치될 때 알릴 애플리케이션 정의 콜백 함수에 대한 선언을 제공합니다.
FNFCIFREE FNFCIFREE 매크로는 애플리케이션 정의 콜백 함수에 대한 선언을 제공하여 이전에 할당된 메모리를 FCI 컨텍스트에서 해제합니다.
FNFCIGETNEXTCABINET FNFCIGETNEXTCABINET 매크로는 애플리케이션 정의 콜백 함수에 대한 선언을 제공하여 다음 캐비닛에 대한 정보를 요청합니다.
FNFCIGETOPENINFO FNFCIGETOPENINFO 매크로는 애플리케이션 정의 콜백 함수가 파일을 열고 파일 날짜, 시간 및 특성을 검색하는 선언을 제공합니다.
FNFCIGETTEMPFILE FNFCIGETTEMPFILE 매크로는 임시 파일 이름을 얻기 위해 애플리케이션 정의 콜백 함수에 대한 선언을 제공합니다.
FNFCIOPEN FNFCIOPEN 매크로는 애플리케이션 정의 콜백 함수가 FCI 컨텍스트에서 파일을 열도록 선언을 제공합니다.
FNFCIREAD FNFCIREAD 매크로는 애플리케이션 정의 콜백 함수가 FCI 컨텍스트의 파일에서 데이터를 읽도록 선언을 제공합니다.
FNFCISEEK FNFCISEEK 매크로는 FCI 컨텍스트에서 파일 포인터를 지정된 위치로 이동하는 애플리케이션 정의 콜백 함수에 대한 선언을 제공합니다.
FNFCISTATUS FNFCISTATUS 매크로는 애플리케이션 정의 콜백 함수가 사용자를 업데이트하는 선언을 제공합니다.
FNFCIWRITE

FNFCIWRITE 매크로는 애플리케이션 정의 콜백 함수가 FCI 컨텍스트의 파일에 데이터를 쓰는 선언을 제공합니다.

구조체

CCAB CCAB 구조체에는 캐비닛 정보가 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

fdi.h 헤더

아티클2023. 03. 09.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

fdi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

FDICopy FDICopy 함수는 캐비닛에서 파일을 추출합니다.
FDICreate FDICreate 함수는 FDI 컨텍스트를 만듭니다.
FDIDestroy FDIDestroy 함수는 열려 있는 FDI 컨텍스트를 삭제합니다.
FDIIsCabinet FDIIsCabinet 함수는 파일이 캐비닛인지 여부를 확인하고, 파일인 경우 해당 파일에 대한 정보를 반환합니다.
FDITruncateCabinet FDITruncateCabinet 함수는 지정된 폴더 번호에서 시작하는 캐비닛 파일을 자릅니다.
FNALLOC FNALLOC는 FDI 컨텍스트에서 메모리를 할당하기 위해 애플리케이션 정의 콜백 함수에 대한 선언을 제공합니다.
FNCLOSE FNCLOSE 매크로는 FDI 컨텍스트에서 파일을 닫기 위해 애플리케이션 정의 콜백 함수에 대한 선언을 제공합니다.

FNFDINOTIFY FNFDINOTIFY 매크로는 애플리케이션 정의 콜백 알림 함수에 대한 선언을 제공하여 디코더의 상태에 따라 애플리케이션을 업데이트합니다.
FNFREE FNFREE 매크로는 FDI 컨텍스트에서 이전에 할당된 메모리를 해제하기 위해 애플리케이션 정의 콜백 함수에 대한 선언을 제공합니다.
FNOPEN FNOPEN 매크로는 FDI 컨텍스트에서 파일을 열기 위해 애플리케이션 정의 콜백 함수에 대한 선언을 제공합니다.
FNREAD FNREAD 매크로는 애플리케이션 정의 콜백 함수가 FDI 컨텍스트의 파일에서 데이터를 읽는 선언을 제공합니다.
FNSEEK FNSEEK 매크로는 FDI 컨텍스트에서 지정된 위치로 파일 포인터를 이동하는 애플리케이션 정의 콜백 함수에 대한 선언을 제공합니다.
FNWRITE FNWRITE 매크로는 애플리케이션 정의 콜백 함수가 FDI 컨텍스트의 파일에 데이터를 쓰는 선언을 제공합니다.

구조체

FDICABINETINFO FDICABINETINFO 구조체에는 특정 캐비닛 파일에 대한 세부 정보가 포함되어 있습니다.
FDINOTIFICATION FNFDINOTIFY에 정보를 제공하는 FDINOTIFICATION 구조체입니다.

fdi_fci_types.h 헤더

아티클2023. 08. 23.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

fdi_fci_types.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

TCOMPfromLZXWindow TCOMPfromLZXWindow 매크로는 창 크기를 FCIAddFile에 대한 LXZTCOMP 값으로 변환합니다.

구조체

Erf ERF 구조에는 FCI/FDI의 오류 정보가 포함되어 있습니다. 호출자는 이 구조를 수정해서는 안 됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

featurestagingapi.h 헤더

아티클2023. 03. 09.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

featurestagingapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

GetFeatureEnabledState 이 함수는 인프라 전용입니다. (GetFeatureEnabledState)
GetFeatureVariant 이 함수는 인프라 전용입니다. (GetFeatureVariant)
RecordFeatureError 이 함수는 인프라 전용입니다. (RecordFeatureError)
RecordFeatureUsage 이 함수는 인프라 전용입니다. (RecordFeatureUsage)
SubscribeFeatureStateChangeNotification 이 함수는 인프라 전용입니다. (SubscribeFeatureStateChangeNotification)
UnsubscribeFeatureStateChangeNotification 이 함수는 인프라 전용입니다. (UnsubscribeFeatureStateChangeNotification)

구조체

FEATURE_ERROR 이 구조는 인프라 전용입니다.

열거형

FEATURE_CHANGE_TIME 이 열거형은 인프라 전용입니다. (FEATURE_CHANGE_TIME)
FEATURE_ENABLED_STATE 이 열거형은 인프라 전용입니다. (FEATURE_ENABLED_STATE)

fhcfg.h 헤더

아티클2023. 08. 23.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

fhcfg.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IFhConfigMgr IFhConfigMgr 인터페이스를 사용하면 클라이언트 애플리케이션이 이 인터페이스의 메서드가 호출되는 사용자 계정에 대한 파일 기록 구성을 읽고 수정할 수 있습니다.
IFhReassociation 이 인터페이스를 사용하면 클라이언트 애플리케이션이 파일 기록 대상에서 파일 기록 구성을 현재 사용자와 다시 연결할 수 있습니다.
IFhScopeIterator IFhScopeIterator 인터페이스를 사용하면 클라이언트 애플리케이션이 포함 또는 제외 목록에서 개별 항목을 열거할 수 있습니다. 포함 및 제외 목록을 검색하려면 IFhConfigMgr::GetIncludeExcludeRules 메서드를 호출합니다.
IFhTarget IFhTarget 인터페이스를 사용하면 클라이언트 애플리케이션이 파일 기록 백업 대상의 숫자 및 문자열 속성을 읽을 수 있습니다.

열거형

FH_BACKUP_STATUS 파일 기록 백업을 사용할 수 있는지 여부를 지정합니다.
FH_DEVICE_VALIDATION_RESULT

스토리지 디바이스 또는 네트워크 공유를 파일 기록 백업 대상으로 사용할 수 있는지 여부를 나타냅니다.
FH_LOCAL_POLICY_TYPE 파일 기록 기능에 대한 로컬 정책의 형식을 지정합니다. 각 로컬 정책에는 연결된 숫자 매개 변수가 있습니다.
FH_PROTECTED_ITEM_CATEGORY 포함 또는 제외 목록의 형식을 지정합니다.
FH_RETENTION_TYPES 백업 대상에서 이전 버전의 파일 및 폴더를 삭제할 수 있는 조건에 따라 지정합니다.
FH_TARGET_DRIVE_TYPES 파일 기록 백업 대상의 형식을 지정합니다.
FH_TARGET_PROPERTY_TYPE 백업 대상의 속성 유형을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

fhsvcctl.h 헤더

아티클2023. 08. 23.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

fhsvcctl.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

FhServiceBlockBackup 이 함수는 현재 사용자의 백업을 일시적으로 차단합니다.
FhServiceClosePipe FhServiceOpenPipe를 사용하여 열린 파일 기록 서비스에 대한 통신 채널을 닫습니다.
FhServiceOpenPipe 파일 기록 서비스에 대한 통신 채널을 엽니다.
FhServiceReloadConfiguration 이 함수를 사용하면 파일 기록 서비스가 현재 사용자의 파일 기록 구성 파일을 다시 로드합니다.
FhServiceStartBackup 이 함수는 현재 사용자에 대한 즉각적인 백업을 시작합니다.
FhServiceStopBackup 이 함수는 현재 사용자에 대한 지속적인 백업 주기를 중지합니다.
FhServiceUnblockBackup 이 함수는 FhServiceBlockBackup을 통해 차단된 백업의 차단을 해제합니다.

피드백

이 페이지가 도움이 되었나요?

Yes No

Microsoft Q&A에서 도움말 보기

fibersapi.h 헤더

아티클2023. 08. 23.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

fibersapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

FlsAlloc FLS(파이버 로컬 스토리지) 인덱스를 할당합니다.
FlsFree FLS(파이버 로컬 스토리지) 인덱스를 해제하여 재사용할 수 있도록 합니다.
FlsGetValue 지정된 FLS 인덱스에 대한 호출 파이버의 FLS(파이버 로컬 스토리지) 슬롯에서 값을 검색합니다. 각 파이버에는 각 FLS 인덱스에 대한 자체 슬롯이 있습니다.
FlsSetValue 지정된 FLS 인덱스에 대한 호출 파이버의 FLS(파이버 로컬 스토리지) 슬롯에 값을 저장합니다. 각 파이버에는 각 FLS 인덱스에 대한 자체 슬롯이 있습니다.
IsThreadAFiber 현재 스레드가 파이버인지 여부를 확인합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

fileapi.h 헤더

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 접근 및 저장
- System Services

fileapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

기능

테이블 확장

AreFileApisANSI 파일 I/O 함수가 ANSI 또는 OEM 문자 집합 코드 페이지를 사용하는지 여부를 확인합니다.
AreShortNamesEnabled AreShortNamesEnabled 함수는 지정된 볼륨에 짧은 이름을 사용할 수 있는지 여부를 결정합니다.
CompareFileTime 두 개의 파일을 비교합니다.
CreateDirectory2A 새 디렉터리를 만듭니다. (ANSI)
CreateDirectory2W 새 디렉터리를 만듭니다. (유니코드)
CreateDirectoryA 새 디렉터리를 만듭니다. (CreateDirectoryA)
CreateDirectoryW 새 디렉터리를 만듭니다. (CreateDirectoryW)
CreateFile2 파일 또는 I/O 디바이스를 만들거나 엽니다.
CreateFile3

파일 또는 I/O 디바이스를 만들거나 엽니다.
CreateFileA 파일 또는 I/O 디바이스를 만들거나 엽니다. 가장 일반적으로 사용되는 I/O 디바이스는 다음과 같습니다. _file, 파일 스트림, 디렉터리, 실제 디스크, 볼륨, 콘솔 버퍼, 테이프 드라이브, 통신 리소스, mailslot 및 파이프. (ANSI)
CreateFileW 파일 또는 I/O 디바이스를 만들거나 엽니다. 가장 일반적으로 사용되는 I/O 디바이스는 다음과 같습니다. _file, 파일 스트림, 디렉터리, 실제 디스크, 볼륨, 콘솔 버퍼, 테이프 드라이브, 통신 리소스, mailslot 및 파이프. (유니코드)
DefineDosDeviceW MS-DOS 디바이스 이름을 정의, 재정의 또는 삭제합니다. (DefineDosDeviceW)
DeleteFile2A 기존 파일을 삭제합니다. (ANSI)
DeleteFile2W 기존 파일을 삭제합니다. (유니코드)
DeleteFileA 기존 파일을 삭제합니다. (DeleteFileA)
DeleteFileW 기존 파일을 삭제합니다. (DeleteFileW)
DeleteVolumeMountPointW 드라이브 문자 또는 탑재된 폴더를 삭제합니다. (DeleteVolumeMountPointW)
FileTimeToLocalFileTime 파일 시간을 로컬 파일 시간으로 변환합니다.
findClose FindFirstFile, FindFirstFileEx, FindFirstFileNameW, FindFirstFileNameTransactedW, FindFirstFileTransactedW, FindFirstStreamTransactedW 또는 FindFirstStreamW 함수에서 연 파일 검색 핸들을 닫습니다.
findCloseChangeNotification

변경 알림 핸들 모니터링을 중지합니다.
FindFirstChangeNotificationA 변경 알림 핸들을 만들고 초기 변경 알림 필터 조건을 설정합니다. (ANSI)
FindFirstChangeNotificationW 변경 알림 핸들을 만들고 초기 변경 알림 필터 조건을 설정합니다. (유니코드)
FindFirstFileA 디렉터리에서 특정 이름과 일치하는 이름으로 파일 또는 하위 디렉터리를 검색합니다(또는 와일드카드를 사용하는 경우 부분 이름). (ANSI)
FindFirstFileExA 디렉터리에서 지정된 이름과 특성과 일치하는 파일 또는 하위 디렉터리를 검색합니다. (FindFirstFileExA)
FindFirstFileExW 디렉터리에서 지정된 이름과 특성과 일치하는 파일 또는 하위 디렉터리를 검색합니다. (FindFirstFileExW)
FindFirstFileNameW 지정된 파일에 대한 모든 하드 링크의 열거형을 만듭니다. FindFirstFileNameW 함수는 FindNextFileNameW 함수에 대한 후속 호출에 사용할 수 있는 열거형에 대한 핸들을 반환합니다.
FindFirstFileW 디렉터리에서 특정 이름과 일치하는 이름으로 파일 또는 하위 디렉터리를 검색합니다(또는 와일드카드를 사용하는 경우 부분 이름). (유니코드)
FindFirstStreamW 지정된 파일 또는 디렉터리에 ::$DATA 스트림 형식으로 첫 번째 스트림을 열거합니다.
FindFirstVolumeW 컴퓨터에서 볼륨의 이름을 검색합니다. (FindFirstVolumeW)
FindNextChangeNotification 운영 체제가 다음에 적절한 변경을 감지할 때 변경 알림 핸들에 신호를 보내라고 요청합니다.
FindNextFileA FindFirstFile, FindFirstFileEx 또는 FindFirstFileTransacted 함수에 대한 이전 호출에서 파일 검색을 계속합니다. (ANSI)

FindNextFileNameW FindFirstFileNameW 함수를 성공적으로 호출하여 반환된 핸들을 사용하여 파일에 대한 하드 링크를 계속 열거합니다.
FindNextFileW FindFirstFile, FindFirstFileEx 또는 FindFirstFileTransacted 함수에 대한 이전 호출에서 파일 검색을 계속합니다. (유니코드)
FindNextStreamW FindFirstStreamW 함수에 대한 이전 호출로 시작된 스트림 검색을 계속합니다.
FindNextVolumeW FindFirstVolume 함수를 호출하여 시작된 볼륨 검색을 계속합니다. (FindNextVolumeW)
findVolumeClose 지정된 볼륨 검색 핸들을 닫습니다.
FlushFileBuffers 지정된 파일의 버퍼를 플러시하고 버퍼링된 모든 데이터를 파일에 쓰게 합니다.
GetCompressedFileSizeA 지정된 파일을 저장하는 데 사용되는 디스크 스토리지의 실제 바이트 수를 검색합니다. (ANSI)
GetCompressedFileSizeW 지정된 파일을 저장하는 데 사용되는 디스크 스토리지의 실제 바이트 수를 검색합니다. (유니코드)
GetDiskFreeSpaceA 디스크의 사용 가능한 공간을 포함하여 지정된 디스크에 대한 정보를 검색합니다. (ANSI)
GetDiskFreeSpaceExA 디스크 볼륨에서 사용할 수 있는 공간의 양(총 공간 양, 사용 가능한 공간의 총 크기 및 호출 스레드와 연결된 사용자가 사용할 수 있는 총 사용 가능한 공간 양)에 대한 정보를 검색합니다. (ANSI)
GetDiskFreeSpaceExW 디스크 볼륨에서 사용할 수 있는 공간의 양(총 공간 양, 사용 가능한 공간의 총 크기 및 호출 스레드와 연결된 사용자가 사용할 수 있는 총 사용 가능한 공간 양)에 대한 정보를 검색합니다. (유니코드)

GetDiskFreeSpaceW 디스크의 사용 가능한 공간을 포함하여 지정된 디스크에 대한 정보를 검색합니다. (유니코드)
GetDiskSpaceInformationA 지정된 루트 경로의 볼륨에 대한 디스크 공간 정보를 가져옵니다.
GetDiskSpaceInformationW 지정된 루트 경로의 볼륨에 대한 디스크 공간 정보를 가져옵니다.
GetDriveTypeA 디스크 드라이브가 이동식, 고정, CD-ROM, RAM 디스크 또는 네트워크 드라이브인지 여부를 확인합니다. (ANSI)
GetDriveTypeW 디스크 드라이브가 이동식, 고정, CD-ROM, RAM 디스크 또는 네트워크 드라이브인지 여부를 확인합니다. (유니코드)
GetFileAttributesA 지정된 파일 또는 디렉터리에 대한 파일 시스템 특성을 검색합니다. (ANSI)
GetFileAttributesExA 지정된 파일 또는 디렉터리에 대한 특성을 검색합니다. (ANSI)
GetFileAttributesExW 지정된 파일 또는 디렉터리에 대한 특성을 검색합니다. (유니코드)
GetFileAttributesW 지정된 파일 또는 디렉터리에 대한 파일 시스템 특성을 검색합니다. (유니코드)
GetFileInformationByHandle 지정된 파일에 대한 파일 정보를 검색합니다. (GetFileInformationByHandle)
GetFileSize 지정된 파일의 크기(바이트)를 검색합니다.
GetFileSizeEx 지정된 파일의 크기를 검색합니다.

GetFileTime 파일 또는 디렉터리가 생성되고, 마지막으로 액세스하고, 마지막으로 수정한 날짜와 시간을 검색합니다.
GetFileType 지정된 파일의 파일 형식을 검색합니다.
GetFinalPathNameByHandleA 지정된 파일의 최종 경로를 검색합니다. (ANSI)
GetFinalPathNameByHandleW 지정된 파일의 최종 경로를 검색합니다. (유니코드)
GetFullPathNameA 지정된 파일의 전체 경로 및 파일 이름을 검색합니다. (ANSI)
GetFullPathNameW 지정된 파일의 전체 경로 및 파일 이름을 검색합니다. (유니코드)
GetLogicalDrives 현재 사용 가능한 디스크 드라이브를 나타내는 비트 마스크를 검색합니다.
GetLogicalDriveStringsW 시스템에서 유효한 드라이브를 지정하는 문자열로 버퍼를 채웁니다. (GetLogicalDriveStringsW)
GetLongPathNameA 지정된 경로를 긴 형식으로 변환합니다. (ANSI)
GetLongPathNameW 지정된 경로를 긴 형식으로 변환합니다. (유니코드)
GetShortPathNameW 지정된 경로의 짧은 경로 형식을 검색합니다. (GetShortPathNameW)
GetTempFileNameA 임시 파일의 이름을 만듭니다. 고유한 파일 이름이 생성되면 빈 파일이 생성되고 해당 파일의 핸들이 해제됩니다. 그렇지 않으면 파일 이름만 생성됩니다. (GetTempFileNameA)

GetTempFileNameW 임시 파일의 이름을 만듭니다. 고유한 파일 이름이 생성되면 빈 파일이 생성되고 해당 파일의 핸들이 해제됩니다. 그렇지 않으면 파일 이름만 생성됩니다. (GetTempFileNameW)
GetTempPath2A 호출 프로세스의 권한에 따라 임시 파일에 대해 지정된 디렉터리의 경로를 검색합니다. (ANSI)
GetTempPath2W 호출 프로세스의 권한에 따라 임시 파일에 대해 지정된 디렉터리의 경로를 검색합니다. (유니코드)
GetTempPathA 임시 파일에 대해 지정된 디렉터리의 경로를 검색합니다. (ANSI)
GetTempPathW 임시 파일에 대해 지정된 디렉터리의 경로를 검색합니다. (유니코드)
GetVolumeInformationA 지정된 루트 디렉터리와 연결된 파일 시스템 및 볼륨에 대한 정보를 검색합니다. (ANSI)
GetVolumeInformationByHandleW 지정된 파일과 연결된 파일 시스템 및 볼륨에 대한 정보를 검색합니다.
GetVolumeInformationW 지정된 루트 디렉터리와 연결된 파일 시스템 및 볼륨에 대한 정보를 검색합니다. (유니코드)
getVolumeNameForVolumeMountPointW 지정된 볼륨 탑재 지점(드라이브 문자, 볼륨 GUID 경로 또는 탑재된 폴더)과 연결된 볼륨의 볼륨 GUID 경로를 검색합니다. (GetVolumeNameForVolumeMountPointW)
getVolumePathNamesForVolumeNameW 지정된 볼륨의 드라이브 문자 및 탑재된 폴더 경로 목록을 검색합니다. (GetVolumePathNamesForVolumeNameW)
GetVolumePathNameW 지정된 경로가 탑재되는 볼륨 탑재 지점을 검색합니다. (GetVolumePathNameW)
LocalFileTimeToFileTime

UTC(협정 세계시)에 따라 로컬 파일 시간을 파일 시간으로 변환합니다.
LockFile 호출 프로세스에서 배타적 액세스를 위해 지정된 파일을 잠급 수 있습니다.
LockFileEx 호출 프로세스에서 배타적 액세스를 위해 지정된 파일을 잠급 수 있습니다. 이 함수는 동기적으로 또는 비동기적으로 작동할 수 있으며 배타적 잠금 또는 공유 잠금을 요청할 수 있습니다.
QueryDosDeviceW MS-DOS 디바이스 이름에 대한 정보를 검색합니다. (QueryDosDeviceW)
ReadFile 지정된 파일 또는 I/O(입출력) 디바이스에서 데이터를 읽습니다. 읽기는 디바이스에서 지원하는 경우 파일 포인터로 지정된 위치에서 발생합니다.
ReadFileEx 지정된 파일 또는 I/O(입출력) 디바이스에서 데이터를 읽습니다. 읽기가 완료되거나 취소되고 호출 스레드가 경고 대기 상태에 있을 때 지정된 완료 루틴을 호출하여 완료 상태를 비동기적으로 보고합니다.
ReadFileScatter 파일에서 데이터를 읽고 버퍼 배열에 저장합니다.
RemoveDirectory2A 기존 빈 디렉터리를 삭제합니다. (ANSI)
RemoveDirectory2W 기존 빈 디렉터리를 삭제합니다. (유니코드)
RemoveDirectoryA 기존 빈 디렉터리를 삭제합니다. (ANSI)
RemoveDirectoryW 기존 빈 디렉터리를 삭제합니다. (유니코드)
SetEndOfFile 지정된 파일의 실제 파일 크기를 파일 포인터의 현재 위치로 설정합니다.

SetFileApisToANSI 파일 I/O 함수가 현재 프로세스에 대한 ANSI 문자 집합 코드 페이지를 사용하도록 합니다.
SetFileApisToOEM 프로세스에 대한 파일 I/O 함수가 OEM 문자 집합 코드 페이지를 사용하도록 합니다.
setFileAttributesA 파일 또는 디렉터리의 특성을 설정합니다. (ANSI)
SetFileAttributesW 파일 또는 디렉터리의 특성을 설정합니다. (유니코드)
SetFileInformationByHandle 지정된 파일의 파일 정보를 설정합니다.
SetFileIoOverlappedRange 가상 주소 범위를 지정된 파일 핸들과 연결합니다.
SetFilePointer 지정된 파일의 파일 포인터를 이동합니다. (SetFilePointer)
SetFilePointerEx 지정된 파일의 파일 포인터를 이동합니다. (SetFilePointerEx)
SetFileTime 지정된 파일 또는 디렉터리를 만들거나 마지막으로 액세스했거나 마지막으로 수정한 날짜와 시간을 설정합니다.
SetFileValidData 지정된 파일의 유효한 데이터 길이를 설정합니다. 이 함수는 매우 제한된 시나리오에서 유용합니다. 자세한 내용은 설명 섹션을 참조하세요.
UnlockFile 열려 있는 파일에서 영역의 잠금을 해제합니다.
UnlockFileEx 지정된 파일의 영역 잠금을 해제합니다. 이 함수는 동기적으로 또는 비동기적으로 작동할 수 있습니다.

WriteFile 지정된 파일 또는 I/O(입출력) 디바이스에 데이터를 씁니다.
WriteFileEx 지정된 파일 또는 I/O(입출력) 디바이스에 데이터를 씁니다. 쓰기가 완료되거나 취소되고 호출 스레드가 경고 대기 상태일 때 지정된 완료 루틴을 호출하여 완료 상태를 비동기적으로 보고합니다.
WriteFileGather 버퍼 배열에서 데이터를 검색하고 파일에 데이터를 씁니다.

구조

테이블 확장

BY_HANDLE_FILE_INFORMATION GetFileInformationByHandle 함수가 검색하는 정보를 포함합니다.
CREATEFILE2_EXTENDED_PARAMETERS CreateFile2에 대한 선택적 확장 매개 변수를 포함합니다.
CREATEFILE3_EXTENDED_PARAMETERS
DISK_SPACE_INFORMATION DISK_SPACE_INFORMATION 구조에는 특정 볼륨의 디스크 공간에 대한 정보가 포함됩니다.
WIN32_FILE_ATTRIBUTE_DATA 파일 또는 디렉터리에 대한 특성 정보를 포함합니다.
WIN32_FIND_STREAM_DATA FindFirstStreamW 또는 FindNextStreamW 함수에서 찾은 스트림에 대한 정보를 포함합니다.

열거 목록

테이블 확장

DIRECTORY_FLAGS CreateDirectory2 함수와 함께 디렉터리를 만드는 방법을 지정하는 데 사용할 수 있는 플래그를 정의합니다.
STREAM_INFO_LEVELS FindFirstStreamW 함수와 함께 반환된 데이터의 정보 수준을 지정하는 데 사용되는 값을 정의합니다.

Last updated on 2025. 04. 11.

fileapifromapp.h 헤더

아티클2023. 08. 23.

이 헤더는 데이터 액세스 및 스토리지에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지

fileapifromapp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CopyFileFromAppW 새 파일에 기존 파일을 복사합니다. 이 함수는 유니버설 Windows 플랫폼 앱 보안 모델을 준수한다는 점을 제외하고 이 함수의 동작은 CopyFile과 동일합니다.
CreateDirectoryFromAppW 새 디렉터리를 만듭니다. 이 함수는 유니버설 Windows 플랫폼 앱 보안 모델을 준수한다는 점을 제외하고 이 함수의 동작은 CreateDirector와 동일합니다.
CreateFile2FromAppW 파일 또는 I/O 디바이스를 만들거나 엽니다. 이 함수는 유니버설 Windows 플랫폼 앱 보안 모델을 준수한다는 점을 제외하고 이 함수의 동작은 CreateFile2와 동일합니다.
CreateFileFromAppW 파일 또는 I/O 디바이스를 만들거나 엽니다. 이 함수는 유니버설 Windows 플랫폼 앱 보안 모델을 준수한다는 점을 제외하고 이 함수의 동작은 CreateFile과 동일합니다.
DeleteFileFromAppW 기존 파일을 삭제합니다. 이 함수는 유니버설 Windows 플랫폼 앱 보안 모델을 준수한다는 점을 제외하고 이 함수의 동작은 DeleteFile과 동일합니다.
FindFirstFileExFromAppW 디렉터리를 검색하여 지정된 이름 및 특성과 일치하는 파일 또는 하위 디렉터리를 찾습니다.
GetFileAttributesExFromAppW 지정된 파일 또는 디렉터리의 특성을 검색합니다. 이 함수는 유니버설 Windows 플랫폼 앱 보안 모델을 준수한다는 점을 제외하고 이 함수의 동작은 GetFileAttributesEx와 동일합니다.

MoveFileFromAppW 자식을 포함한 기존 파일 또는 디렉터리를 이동합니다. 이 함수는 유니버설 Windows 플랫폼 앱 보안 모델을 준수한다는 점을 제외하고 이 함수의 동작은 MoveFile과 동일합니다.
RemoveDirectoryFromAppW 기존 빈 디렉터리를 삭제합니다. 이 함수는 유니버설 Windows 플랫폼 앱 보안 모델을 준수한다는 점을 제외하고 이 함수의 동작은 RemoveDirectory와 동일합니다.
ReplaceFileFromAppW 한 파일을 다른 파일로 바꾸며, 원본 파일의 백업 복사본을 만드는 옵션을 제공합니다. 이 함수는 유니버설 Windows 플랫폼 앱 보안 모델을 준수한다는 점을 제외하고 이 함수의 동작은 ReplaceFile과 동일합니다.
SetFileAttributesFromAppW 파일 또는 디렉터리의 특성을 설정합니다. 이 함수는 유니버설 Windows 플랫폼 앱 보안 모델을 준수한다는 점을 제외하고 이 함수의 동작은 SetFileAttributes와 동일합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

filehc.h 헤더

아티클2023. 03. 09.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

filehc.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AssociateContextWithName 이름 캐시에 이름을 삽입하여 지정된 FIO_CONTEXT 구조를 찾습니다.
AssociateFileEx 파일을 비시도 컨텍스트와 연결합니다.
CacheCreateFile 캐시에 파일을 만들거나 기존 파일을 찾습니다.
CacheRichCreateFile 캐시에 파일을 만들거나 기존 파일을 찾아서 캐시에 속성을 추가할 수 있습니다.
FindOrCreateNameCache 이름 캐시를 찾거나 만듭니다.
FindSyncContextFromName 지정된 사용자 이름과 연결된 FIO_CONTEXT 구조를 검색합니다.
GetDotStuffState 점 스터핑 메커니즘이 켜져 있을 때 점이 파일에 추가되는지 여부를 결정합니다.
GetFileSizeFromContext 핸들을 사용하여 캐시된 파일 크기를 보고합니다.

InitializeCache 파일 핸들 캐시의 초기화 값을 제공합니다.
InsertFile 캐시에 파일을 삽입합니다.
InvalidateName 사용자가 이름 캐시에서 단일 이름 및 연결된 모든 데이터를 제거할 수 있도록 합니다.
ProduceDotStuffedContext 요청된 상태의 FIO_CONTEXT 구조를 검색합니다.
ReleaseNameCache 이름 캐시를 해제합니다.
SetDotStuffingOnWrites 메시지의 파일 핸들 캐시의 쓰기 경로에서 점 스터핑 속성을 사용하도록 설정합니다.
SetDotStuffState 점 스터핑을 FIO_CONTEXT 구조에서 설정할 수 있습니다.
TerminateCache 캐시를 제거하고 DLL을 사용하여 비동기 I/O를 종료합니다.

콜백 함수

CACHE_DESTROY_CALLBACK 이름 캐시의 항목이 제거 될 때마다 호출 되는 함수입니다.
CACHE_READ_CALLBACK 캐시 내의 항목을 검사하는 데 도움이 되도록 캐시에 제공되는 콜백입니다.
FCACHE_CREATE_CALLBACK 캐시에 항목을 만드는 데 사용되는 콜백 함수입니다. (FCACHE_CREATE_CALLBACK)

FCACHE_RICHCREATE_CALLBACK 캐시에 항목을 만드는 데 사용되는 콜백 함수입니다. (FCACHE_RICHCREATE_CALLBACK)

구조체

NAME_CACHE_CONTEXT 이름 캐시를 나타냅니다.

filter.h 헤더

아티클2023. 08. 23.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 인덱싱 서비스
- Windows 검색

filter.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IFilter 문서에서 텍스트 및 속성(특성이라고도 함)을 검색합니다.

구조체

FILTERREGION 개체 내에서 지정된 텍스트 부분의 위치와 범위를 설명합니다.
FULLPROPSPEC 속성 집합 및 속성 집합 내의 속성을 지정합니다.
STAT_CHUNK 청크의 특성을 설명합니다.

열거형

CHUNK_BREAKTYPE 현재 청크를 이전 청크와 구분하는 중단 유형을 설명합니다.

CHUNKSTATE 현재 청크가 텍스트 형식 속성인지 아니면 값 형식 속성인지를 지정합니다.
IFILTER_FLAGS 호출자가 IPropertySetStorage 및 IPropertyStorage 인터페이스를 사용하여 추가 속성을 찾아야 하는지 여부를 나타냅니다.
IFILTER_INIT 필터링 프로세스를 제어하는 플래그입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

filtereg.h 헤더

아티클2023. 08. 23.

이 헤더는 Windows Search에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 검색

filtereg.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ILoadFilter 필터를 로드하는 메서드를 제공하는 FilterRegistration 개체에 의해 구현되는 메서드 및 속성을 정의합니다.

구조체

FILTERED_DATA_SOURCES 필터가 로드되는 셸 데이터 원본에 대한 매개 변수를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

fltuser.h 헤더

아티클2023. 08. 23.

이 헤더는 설치 가능한 파일 시스템 DDI 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 설치 가능한 파일 시스템 DDI 참조

fltuser.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

FilterAttach FilterAttach 함수는 지정된 볼륨에 새 미니필터 instance 연결합니다.
FilterAttachAtAltitude FilterAttachAtAltitude 함수는 지정된 고도의 볼륨에 새 미니필터 instance 연결하여 미니필터의 INF(설정 정보) 파일의 모든 설정을 재정의하는 디버깅 지원 함수입니다.
FilterClose FilterClose 함수는 열린 미니필터 핸들을 닫습니다.
FilterConnectCommunicationPort FilterConnectCommunicationPort는 파일 시스템 미니필터에서 만든 통신 서버 포트에 대한 새 연결을 엽니다.
필터만들기 FilterCreate 함수는 지정된 미니필터에 대한 핸들을 만듭니다.
FilterDetach FilterDetach 함수는 지정된 미니 필터 instance 지정된 볼륨에서 분리합니다.
FilterFindClose FilterFindClose 함수는 지정된 미니필터 검색 핸들을 닫습니다. FilterFindFirst 및 FilterFindNext 함수는 이 검색 핸들을 사용하여 미니필터를 찾습니다.
FilterFindFirst

FilterFindFirst 함수는 필터 드라이버(미니필터 드라이버 instance 또는 레거시 필터 드라이버)에 대한 정보를 반환하고 등록된 필터의 전역 목록에서 필터 검사를 시작하는 데 사용됩니다.
FilterFindNext FilterFindNext 함수는 FilterFindFirst 호출로 시작된 필터 검색을 계속합니다.
FilterGetDosName FilterGetDosName 함수는 지정된 볼륨 이름에 해당하는 MS-DOS 디바이스 이름을 반환합니다.
FilterGetInformation FilterGetInformation 함수는 미니필터에 대한 다양한 종류의 정보를 반환합니다.
FilterGetMessage FilterGetMessage 함수는 커널 모드 미니필터에서 메시지를 가져옵니다.
FilterInstanceClose FilterInstanceClose 함수는 FilterInstanceCreate에서 연 미니필터 instance 핸들을 닫습니다.
FilterInstanceCreate FilterInstanceCreate 함수는 지정된 미니필터 instance 통신하는 데 사용할 수 있는 핸들을 만듭니다.
FilterInstanceFindClose FilterInstanceFindClose 함수는 지정된 미니필터 instance 검색 핸들을 닫습니다. FilterInstanceFindFirst 및 FilterInstanceFindNext 함수는 이 검색 핸들을 사용하여 미니필터의 인스턴스를 찾습니다.
FilterInstanceFindFirst FilterInstanceFindFirst 함수는 instance 미니필터 드라이버에 대한 정보를 반환하며 미니필터 인스턴스를 검사하기 위한 시작점으로 사용됩니다.
FilterInstanceFindNext FilterInstanceFindNext 함수는 FilterInstanceFindFirst 호출로 시작된 검색 instance 미니필터 드라이버를 계속합니다.
FilterInstanceGetInformation FilterInstanceGetInformation 함수는 미니필터 instance 대한 다양한 종류의 정보를 반환합니다.
FilterLoad

FilterLoad 함수는 미니필터 드라이버를 시스템에 동적으로 로드합니다.
FilterReplyMessage FilterReplyMessage 함수는 커널 모드 미니필터의 메시지에 회신합니다.
FilterSendMessage FilterSendMessage 함수는 커널 모드 미니필터에 메시지를 보냅니다.
FilterUnload FilterLoad를 호출하여 지원 미니필터를 로드한 애플리케이션은 FilterUnload 함수를 호출하여 미니필터를 언로드할 수 있습니다.
FilterVolumeFindClose FilterVolumeFindClose 함수는 지정된 볼륨 검색 핸들을 닫습니다. FilterVolumeFindFirst 및 FilterVolumeFindNext는 이 검색 핸들을 사용하여 볼륨을 찾습니다.
FilterVolumeFindFirst FilterVolumeFindFirst 함수는 볼륨에 대한 정보를 반환합니다.
FilterVolumeFindNext FilterVolumeFindNext 함수는 FilterVolumeFindFirst 호출로 시작된 볼륨 검색을 계속합니다.
FilterVolumeInstanceFindClose FilterVolumeInstanceFindClose 함수는 지정된 볼륨 instance 검색 핸들을 닫습니다. FilterVolumeInstanceFindFirst 및 FilterVolumeInstanceFindNext는 이 검색 핸들을 사용하여 볼륨에서 인스턴스를 찾습니다.
FilterVolumeInstanceFindFirst FilterVolumeInstanceFindFirst 함수는 미니필터 드라이버 instance 또는 레거시 필터 드라이버에 대한 정보를 반환하며 볼륨에 연결된 필터 드라이버 검사를 시작하는 데 사용됩니다.
FilterVolumeInstanceFindNext FilterVolumeInstanceFindNext 함수는 FilterVolumeInstanceFindFirst 호출로 시작된 미니필터 드라이버 instance 또는 레거시 필터 드라이버 검색을 계속합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

fontsub.h 헤더

아티클2023. 08. 23.

이 헤더는 Windows GDI에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows GDI

fontsub.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CreateFontPackage CreateFontPackage 함수는 일반적으로 프린터에 전달하기 위해 지정된 TrueType 글꼴의 하위 집합 버전을 만듭니다.
MergeFontPackage MergeFontPackage 함수는 CreateFontPackage에서 만든 글꼴을 조작합니다.

콜백 함수

CFP_ALLOCPROC CreateFontPackage 및 MergeFontPackage에서 메모리를 할당하는 데 사용하는 클라이언트 제공 콜백 함수입니다.
CFP_FREEPROC CreateFontPackage 및 MergeFontPackage에서 메모리를 해제하는 데 사용되는 클라이언트 제공 콜백 함수입니다.
CFP_REALLOCPROC 할당된 버퍼의 크기를 변경해야 하는 경우 CreateFontPackage 및 MergeFontPackage에서 메모리를 다시 할당하는 데 사용하는 클라이언트 제공 콜백 함수입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

frquery.h 헤더

아티클2023. 08. 23.

이 헤더는 Windows Management Instrumentation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Management Instrumentation

frquery.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

CFrameworkQuery CFrameworkQuery 클래스는 쿼리를 구문 분석하고 조작하는 데 사용됩니다. 여기에는 특정 속성 값을 반환하는 메서드와 포함된 특정 속성을 확인하기 위해 쿼리를 테스트하는 메서드가 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

fsrm.h 헤더

아티클2023. 08. 23.

이 헤더는 파일 서버 Resource Manager 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 파일 서버 리소스 관리자

fsrm.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IFsrmAccessDeniedRemediationClient ADR(액세스 거부 수정) 클라이언트 사용자 인터페이스를 표시하는 데 사용됩니다.
IFsrmAction 모든 FSRM 작업 인터페이스에 대한 기본 클래스입니다.
IFsrmActionCommand 할당량, 파일 화면 또는 파일 관리 작업 이벤트에 대한 응답으로 명령 또는 스크립트를 실행하는 데 사용됩니다.
IFsrmActionEmail 할당량 또는 파일 화면 이벤트에 대한 응답으로 전자 메일 메시지를 보내는 데 사용됩니다.
IFsrmActionEmail2 이메일 알림에 나열된 만료된 파일의 수를 제한하는 데 사용됩니다.
IFsrmActionEventLog 할당량, 파일 화면 또는 파일 관리 작업 이벤트에 대한 응답으로 Windows 애플리케이션 이벤트 로그에 이벤트를 기록하는 데 사용됩니다.
IFsrmActionReport 할당량 또는 파일 화면 이벤트에 대한 응답으로 보고서를 생성하는 데 사용됩니다.
IFsrmCollection FSRM 개체의 컬렉션을 정의합니다.

IFsrmCommittableCollection 컬렉션에 추가하거나 컬렉션에서 제거할 수 있는 동일한 형식의 개체를 가질 수 있는 FSRM 개체의 컬렉션을 정의합니다. 컬렉션의 모든 개체는 단일 일괄 처리 작업에서도 커밋할 수 있습니다.
IFsrmDerivedObjectsResult 원본 템플릿이 CommitAndUpdateDerived 메서드를 호출할 때 결과에 액세스하는 데 사용됩니다.
IFsrmExportImport FSRM 개체를 내보내고 가져오는 데 사용됩니다.
IFsrmMutableCollection 컬렉션에서 개체를 추가하거나 제거할 수 있는 FSRM 개체의 컬렉션을 관리하는 데 사용됩니다.
IFsrmObject 모든 FSRM 개체에 대한 기본 클래스입니다.
IFsrmPathMapper 로컬 경로에 매핑되는 네트워크 공유 경로를 검색하는 데 사용됩니다.
IFsrmSetting FSRM을 구성하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

fsrmenums.h 헤더

아티클2023. 08. 23.

이 헤더는 파일 서버 Resource Manager 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 파일 서버 리소스 관리자

fsrmenums.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

열거형

AdrClientDisplayFlags 가능한 ADR(액세스 거부 수정) 클라이언트 표시 플래그 유형에 대해 설명합니다.
AdrClientErrorType 가능한 액세스 거부 수정(ADR) 클라이언트 오류 유형에 대해 설명합니다.
AdrClientFlags ADR(액세스 거부 수정) 클라이언트 작업을 수행할 수 없는 이유를 나타내는 플래그를 열거합니다.
AdrEmailFlags ADR(액세스 거부 수정) 전자 메일에 대한 옵션을 설명합니다.
FsrmAccountType 명령 작업(FsrmActionType 참조)을 실행할 수 있는 컴퓨터 계정 유형을 정의합니다.
FsrmActionType 할당량 또는 파일 화면 이벤트에 대한 응답으로 트리거할 수 있는 작업을 정의합니다(예: 할당량이 초과되거나 파일이 파일 화면을 위반하는 경우).
FsrmClassificationLoggingFlags 분류를 실행하는 동안 정보를 로깅하는 다양한 옵션을 정의합니다.
FsrmCollectionState 컬렉션 개체의 가능한 상태를 정의합니다.
FsrmCommitOptions

개체 컬렉션을 커밋하는 옵션을 정의합니다.
FsrmEnumOptions 개체 컬렉션을 열거하는 옵션을 정의합니다.
FsrmEventType 이벤트 로깅 작업(FsrmActionType 참조)이 기록할 수 있는 이벤트 유형을 정의합니다.
FsrmExecutionOption 파일에 규칙을 적용하는 방법에 대한 옵션을 정의합니다.
FsrmFileManagementLoggingFlags 파일 관리 작업을 실행할 때 로깅 옵션을 정의합니다.
FsrmFileManagementType 파일 관리 작업 유형을 정의합니다.
FsrmFileScreenFlags 파일 화면을 위반하는 IO 작업 실패 옵션을 정의합니다.
FsrmFileStreamingInterfaceType 가능한 스트리밍 인터페이스 유형을 정의합니다.
FsrmFileStreamingMode 파일 스트림에 사용할 스트리밍 모드를 정의합니다.
FsrmFileSystemPropertyId 파일 시스템 속성 ID의 가능한 형식을 정의합니다.
FsrmGetFilePropertyOptions 파일과 연결된 분류 속성을 검색하는 방법을 정의하는 플래그입니다.
FsrmPipelineModuleType 정의할 수 있는 모듈 유형을 정의합니다.
FsrmPropertyBagField 속성 모음의 유형을 설명합니다.

FsrmPropertyBagFlags 속성 모음에 대한 추가 정보를 제공하는 플래그 값을 정의합니다.
FsrmPropertyConditionType 파일의 속성 값이 특정 조건을 충족하는지 여부를 확인하는 데 사용할 수 있는 가능한 비교 작업을 정의합니다.
FsrmPropertyDefinitionAppliesTo 적용할 수 있는 FSRM 분류 속성을 나타내는 플래그입니다.
FsrmPropertyDefinitionFlags 분류 속성 유형에 플래그를 지정합니다.
FsrmPropertyDefinitionType 정의할 수 있는 파일 분류 속성의 형식을 정의합니다.
FsrmPropertyFlags 분류 속성에 대한 추가 정보를 제공하는 플래그 값을 정의합니다.
FsrmPropertyValueType 속성 조건에서 FSRM 속성에 할당되는 값의 형식을 열거합니다.
FsrmQuotaFlags 할당량을 위반하는 IO 작업 실패, 할당량 추적 사용 또는 사용 안 함 및 할당량 검사 작업의 상태 제공하는 옵션을 정의합니다.
FsrmReportFilter 보고서에 포함된 파일을 제한하는 데 사용할 수 있는 필터를 정의합니다.
FsrmReportFormat 보고서를 생성할 때 사용할 수 있는 파일 형식을 정의합니다.
FsrmReportGenerationContext 보고서가 시작되는 컨텍스트를 정의합니다.
FsrmReportLimit 보고서에 포함된 파일을 제한하는 데 사용되는 제한을 정의합니다.

FsrmReportRunningStatus 보고서 작업에 대한 실행 상태를 정의합니다.
FsrmReportType 생성할 수 있는 보고서 유형을 정의합니다.
FsrmRuleFlags 규칙의 가능한 상태를 정의합니다.
FsrmRuleType 정의할 수 있는 규칙 유형을 정의합니다.
FsrmStorageModuleCaps 스토리지 모듈의 기능을 정의하는 플래그입니다.
FsrmStorageModuleType 가능한 스토리지 모듈 유형을 정의합니다.
FsrmTemplateApplyOptions 파생 개체에 템플릿 변경 내용을 적용하는 옵션을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

fsrmpipeline.h 헤더

아티클2023. 08. 23.

이 헤더는 파일 서버 Resource Manager 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 파일 서버 리소스 관리자

fsrmpipeline.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IFsrmClassificationManager 파일 분류를 관리합니다. 이 인터페이스를 사용하여 분류에 사용할 속성을 정의하고, 파일을 분류하기 위한 분류 규칙을 추가하고, 분류 및 스토리지 모듈을 정의하고, 분류 보고를 사용하도록 설정합니다. (IFsrmClassificationManager)
IFsrmClassificationManager2 파일 분류를 관리합니다. 이 인터페이스를 사용하여 분류에 사용할 속성을 정의하고, 파일을 분류하기 위한 분류 규칙을 추가하고, 분류 및 스토리지 모듈을 정의하고, 분류 보고를 사용하도록 설정합니다. (IFsrmClassificationManager2)
IFsrmClassificationRule 분류 규칙을 정의합니다.
IFsrmClassifierModuleDefinition 분류자 모듈을 정의합니다.
IFsrmClassifierModuleImplementation 분류자 모듈은 이 인터페이스를 구현합니다. FSRM은 분류를 실행할 때 모듈의 구현을 호출합니다.
IFsrmPipelineModuleConnector FSRM과 파이프라인 모듈 구현 간의 통신 채널을 만듭니다.
IFsrmPipelineModuleDefinition 파일을 분류하거나 파일에서 속성을 저장하고 검색하는 데 사용되는 모듈을 정의합니다.
IFsrmPipelineModuleImplementation

IFsrmClassifierModuleImplementation 및 IFsrmStorageModuleImplementation에 대한 추상 인터페이스입니다.
IFsrmProperty 속성의 instance 정의합니다.
IFsrmPropertyBag 파일에 대한 분류 속성을 포함합니다.
IFsrmPropertyBag2 IFsrmPropertyBag2 인터페이스
IFsrmPropertyDefinition 파일을 분류하는 데 사용할 속성을 정의합니다. (IFsrmPropertyDefinition)
IFsrmPropertyDefinition2 파일을 분류하는 데 사용할 속성을 정의합니다. (IFsrmPropertyDefinition2)
IFsrmPropertyDefinitionValue 분류 속성 정의 값을 설명하는 속성을 포함합니다.
IFsrmRule 규칙을 정의합니다.
IFsrmStorageModuleDefinition 속성 값을 읽고 쓰는 데 사용되는 로컬 스토리지 모듈을 정의합니다.
IFsrmStorageModuleImplementation 스토리지 모듈은 이 인터페이스를 구현합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

fsrmquota.h 헤더

아티클2023. 03. 09.

이 헤더는 파일 서버 Resource Manager 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 파일 서버 리소스 관리자

fsrmquota.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IFsrmAutoApplyQuota 자동 할당량이 적용되는 디렉터리의 새 하위 디렉터리와 기존 하위 디렉터리에 할당량을 자동으로 추가하는 데 사용됩니다.
IFsrmQuota 지정된 디렉터리에 대한 할당량을 정의하고 사용 통계를 검색하는 데 사용됩니다.
IFsrmQuotaBase 모든 할당량 인터페이스에 대한 기본 인터페이스입니다.
IFsrmQuotaManager 할당량을 관리하는 데 사용됩니다.
IFsrmQuotaManagerEx 할당량, 확장 버전을 관리하는 데 사용됩니다.
IFsrmQuotaObject 할당량 및 자동 할당량 인터페이스의 기본 클래스입니다.
IFsrmQuotaTemplate 새 할당량 개체를 파생할 수 있는 템플릿을 구성하는 데 사용됩니다.
IFsrmQuotaTemplateImported 가져온 할당량 템플릿을 수정하고 저장하는 데 사용됩니다.

IFsrmQuotaTemplateManager 할당량 템플릿을 관리하는 데 사용됩니다.

fsrmreports.h 헤더

아티클2023. 08. 23.

이 헤더는 파일 서버 Resource Manager 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 파일 서버 리소스 관리자

fsrmreports.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IFsrmFileConditionProperty 파일 조건 속성을 정의합니다.
IFsrmFileManagementJob 파일 관리 작업을 정의합니다.
IFsrmFileManagementJobManager 파일 관리 작업을 관리하는 데 사용됩니다.
IFsrmPropertyCondition 파일 관리 작업에서 파일이 만료되었는지 확인하는 데 사용하는 속성 조건을 정의합니다.
IFsrmReport 단일 보고서에 대한 설명 및 필터를 구성하는 데 사용됩니다.
IFsrmReportJob 보고서 작업을 구성하는 데 사용됩니다.
IFsrmReportManager 보고서 작업을 관리하는 데 사용됩니다.
IFsrmReportScheduler 보고서 작업 및 파일 관리 작업에 대한 예약된 작업을 관리하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

fsrmscreen.h 헤더

아티클2023. 08. 23.

이 헤더는 파일 서버 Resource Manager 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 파일 서버 리소스 관리자

fsrmscreen.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IFsrmFileGroup 하나 이상의 파일 이름 패턴을 기반으로 파일 그룹을 정의하는 데 사용됩니다.
IFsrmFileGroupImported 가져온 파일 그룹 개체를 구성하는 데 사용됩니다.
IFsrmFileGroupManager 파일 그룹 개체를 관리하는 데 사용됩니다.
IFsrmFileScreen 파일 그룹이 지정된 디렉터리에 저장되지 않도록 차단하는 파일 화면을 구성하는 데 사용됩니다.
IFsrmFileScreenBase 모든 파일 화면 인터페이스에 대한 기본 클래스입니다.
IFsrmFileScreenException 파일 차단 프로세스에서 지정된 파일을 제외하는 예외를 구성하는 데 사용됩니다.
IFsrmFileScreenManager 파일 화면 개체를 관리하는 데 사용됩니다.
IFsrmFileScreenTemplate 새 파일 화면을 파생할 수 있는 템플릿을 구성하는 데 사용됩니다.
IFsrmFileScreenTemplateImported

가져온 파일 화면 템플릿을 저장하는 데 사용됩니다.
IFsrmFileScreenTemplateManager 파일 화면 템플릿을 관리하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

fsrmtlb.h 헤더

아티클2023. 08. 23.

이 헤더는 파일 서버 Resource Manager 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 파일 서버 리소스 관리자

fsrmtlb.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

DIFsrmClassificationEvents ClassifyFiles 호출을 처리하는 동안 수신되는 이벤트를 처리합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

functiondiscoveryapi.h 헤더

아티클2023. 08. 23.

이 헤더는 다른 디바이스 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 기타 디바이스 기술

functiondiscoveryapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IFunctionDiscovery 이 인터페이스는 클라이언트 프로그램에서 함수 인스턴스를 검색하고, 범주에 대한 기본 함수 instance 가져와서 함수 검색 기본값을 등록할 수 있는 고급 함수 검색 쿼리 개체를 만드는 데 사용됩니다.
IFunctionDiscoveryNotification 이 인터페이스는 비동기 쿼리를 지원하기 위해 클라이언트 프로그램에 의해 구현되며, 쿼리 매개 변수를 충족하는 함수 instance 추가되거나 제거된 경우 함수 검색에서 클라이언트 프로그램에 알리기 위해 호출됩니다.
IFunctionInstance instance 함수는 IFunctionDiscovery 메서드 중 하나를 호출한 결과로 만들어집니다. 클라이언트 프로그램은 이러한 개체를 직접 만들지 않습니다.
IFunctionInstanceCollection 쿼리 또는 instance 요청 가져오기의 결과로 반환된 IFunctionInstance 개체 그룹을 나타냅니다.
IFunctionInstanceCollectionQuery 범주 및 하위 범주를 기반으로 하는 함수 인스턴스 컬렉션에 대한 비동기 쿼리를 구현합니다.
IFunctionInstanceQuery 범주 및 하위 범주를 기반으로 instance 함수에 대한 비동기 쿼리를 구현합니다.

열거형

QueryUpdateAction 함수 검색이 지정된 함수 instance 수행하는 작업의 유형을 나타냅니다. 이 정보는 클라이언트 프로그램의 변경 알림 처리기에서 사용됩니다.
SystemVisibilityFlags 함수 instance 데이터의 표시 여부를 결정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

functiondiscoveryconstraints.h 헤더

아티클2023. 08. 23.

이 헤더는 다른 디바이스 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 기타 디바이스 기술

functiondiscoveryconstraints.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

열거형

PropertyConstraint 함수 인스턴스를 검색하는 데 사용되는 필터 조건을 한정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

functiondiscoveryprovider.h 헤더

아티클2023. 08. 23.

이 헤더는 다른 디바이스 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 기타 디바이스 기술

functiondiscoveryprovider.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IFunctionDiscoveryProvider 검색 공급자가 구현한 기본 인터페이스입니다. 함수 검색 인프라가 공급자 및 해당 리소스와 통신하는 데 사용하는 기본 인터페이스입니다.
IFunctionDiscoveryProviderFactory 함수 검색 개체를 만드는 팩터리 메서드를 제공합니다.
IFunctionDiscoveryProviderQuery 이 인터페이스는 모든 IFunctionDiscoveryProvider::Query 메서드 호출에 전달되며 쿼리 정의 정보를 포함합니다. 공급자는 이를 사용하여 수신하는 각 쿼리 요청에 대한 제약 조건을 결정해야 합니다.
IFunctionDiscoveryServiceProvider 이 인터페이스는 개체를 만들고 초기화하여 함수 instance 나타내는 리소스에 지정된 액세스 인터페이스를 제공하기 위해 구현됩니다. 개체를 만든 후에는 개체를 초기화하기 위해 Initialize 메서드가 호출됩니다.
IProviderProperties 선택적으로 검색 공급자가 자체 속성 저장소를 직접 만들고 관리하기 위해 구현됩니다.
IProviderPropertyConstraintCollection 이 인터페이스는 IFunctionDiscoveryProviderQuery::GetPropertyConstraints를 통해 공급자가 액세스할 수 있습니다.
IProviderPublishing

클라이언트 프로그램이 함수 인스턴스를 추가하고 제거할 수 있도록 검색 공급자에 의해 구현됩니다.
IProviderQueryConstraintCollection 이 인터페이스는 IFunctionDiscoveryProviderQuery::GetQueryConstraints 메서드를 통해 공급자가 액세스할 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

fwpmtypes.h 헤더

아티클2024. 04. 30.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 드라이버, Windows Vista 이상
- Windows 필터링 플랫폼

fwpmtypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

테이블 확장

FWPM_ACTION0 모든 필터 조건이 true인 경우 수행되는 작업을 지정합니다.
FWPM_CALLOUT_CHANGE0 구독자에게 디스패치된 변경 알림을 지정합니다.
FWPM_CALLOUT_ENUM_TEMPLATE0 설명선 열거형을 제한하는 데 사용됩니다.
FWPM_CALLOUT_SUBSCRIPTION0 변경 알림을 구독하는 데 사용됩니다. (FWPM_CALLOUT_SUBSCRIPTION0)
FWPM_CALLOUT0 콜아웃과 연결된 상태를 저장합니다.
FWPM_CLASSIFY_OPTION0 FWPM_CLASSIFY_OPTION0 구조체입니다.
FWPM_CLASSIFY_OPTIONS0 FWPM_CLASSIFY_OPTIONS0 구조체는 FWPM_CLASSIFY_OPTION0 구조를 저장하는 데 사용됩니다.
FWPM_CONNECTION_ENUM_TEMPLATE0

연결 개체 열거형을 제한하는 데 사용됩니다.
FWPM_CONNECTION_SUBSCRIPTION0 연결 개체에 대한 알림을 구독하는 데 사용되는 정보를 저장합니다.
FWPM_CONNECTION0 연결 개체와 연결된 상태를 저장합니다.
FWPM_FIELD0 필드에 대한 스키마 정보를 지정합니다.
FWPM_FILTER_CHANGE0 구독자에게 디스패치된 변경 알림을 저장합니다.
FWPM_FILTER_CONDITION0 수행할 작업에 대해 true여야 하는 필터 조건을 표현합니다.
FWPM_FILTER_ENUM_TEMPLATE0 필터를 열거하는 데 사용됩니다.
FWPM_FILTER_SUBSCRIPTION0 변경 알림을 구독하는 데 사용됩니다.
FWPM_FILTER0 필터와 연결된 상태를 저장합니다.
FWPM_LAYER_ENUM_TEMPLATE0 레이어를 열거하는 데 사용됩니다.
FWPM_LAYER_STATISTICS0 계층과 관련된 통계를 저장합니다.
FWPM_LAYER0 레이어에 대한 스키마 정보입니다.
FWPM_NET_EVENT_CAPABILITY_ALLOW0 앱 컨테이너 네트워크 기능과 관련하여 허용되는 네트워크 트래픽에 대한 정보를 포함합니다.

FWPM_NET_EVENT_CAPABILITY_DROP0 앱 컨테이너 네트워크 기능과 관련하여 삭제된 네트워크 트래픽에 대한 정보를 포함합니다.
FWPM_NET_EVENT_CLASSIFY_ALLOW0 WFP 분류 엔진에 의해 적용되는 허용된 트래픽을 설명하는 정보를 포함합니다.
FWPM_NET_EVENT_CLASSIFY_DROP_MAC0 MAC 계층 삭제 실패를 설명하는 정보를 포함합니다.
FWPM_NET_EVENT_CLASSIFY_DROP0 계층 삭제 실패를 설명하는 정보를 포함합니다. (FWPM_NET_EVENT_CLASSIFY_DROP0)
FWPM_NET_EVENT_CLASSIFY_DROP1 계층 삭제 실패를 설명하는 정보를 포함합니다. (FWPM_NET_EVENT_CLASSIFY_DROP1)
FWPM_NET_EVENT_CLASSIFY_DROP2 계층 삭제 실패를 설명하는 정보를 포함합니다. (FWPM_NET_EVENT_CLASSIFY_DROP2)
FWPM_NET_EVENT_ENUM_TEMPLATE0 net 이벤트를 열거하는 데 사용됩니다.
FWPM_NET_EVENT_HEADER0 모든 이벤트에 공통적인 정보입니다.
FWPM_NET_EVENT_HEADER1 모든 이벤트에 공통적인 정보입니다. 예약되어 있습니다.
FWPM_NET_EVENT_HEADER2 모든 이벤트에 공통적인 정보를 포함합니다. (FWPM_NET_EVENT_HEADER2)
FWPM_NET_EVENT_HEADER3 모든 이벤트에 공통적인 정보를 포함합니다. (FWPM_NET_EVENT_HEADER3)
FWPM_NET_EVENT_IKEEXT_EM_FAILURE0 FWPM_NET_EVENT_IKEEXT_EM_FAILURE0 구조에는 IKE EM(확장 모드) 오류를 설명하는 정보가 포함되어 있습니다. 참고 FWPM_NET_EVENT_IKEEXT_EM_FAILURE0 Windows Vista에서 사용되는 FWPM_NET_EVENT_IKEEXT_EM_FAILURE 특정 구현입니다.

FWPM_NET_EVENT_IKEEXT_EM_FAILURE1 FWPM_NET_EVENT_IKEEXT_EM_FAILURE1 구조에는 IKE EM(확장 모드) 오류를 설명하는 정보가 포함되어 있습니다. 참고 FWPM_NET_EVENT_IKEEXT_EM_FAILURE1 Windows 7 이상에서 사용되는 FWPM_NET_EVENT_IKEEXT_EM_FAILURE 특정 구현입니다.
FWPM_NET_EVENT_IKEEXT_MM_FAILURE0 IKE/AuthIP MM(주 모드) 오류를 설명하는 정보를 포함합니다. (FWPM_NET_EVENT_IKEEXT_MM_FAILURE0)
FWPM_NET_EVENT_IKEEXT_MM_FAILURE1 IKE/AuthIP MM(주 모드) 오류를 설명하는 정보를 포함합니다. (FWPM_NET_EVENT_IKEEXT_MM_FAILURE1)
FWPM_NET_EVENT_IKEEXT_QM_FAILURE0 IKE/AuthIP QM(빠른 모드) 실패를 설명하는 정보를 포함합니다.
FWPM_NET_EVENT_IKEEXT_QM_FAILURE1 IKE/AuthIP QM(빠른 모드) 실패를 설명하는 정보를 포함합니다. FWPM_NET_EVENT_IKEEXT_QM_FAILURE0 사용할 수도 있습니다.
FWPM_NET_EVENT_IPSEC_DOSP_DROP0 IPsec DoS Protection 드롭 이벤트를 설명하는 정보를 포함합니다.
FWPM_NET_EVENT_IPSEC_KERNEL_DROP0 IPsec 커널 드롭 이벤트를 설명하는 정보를 포함합니다.
FWPM_NET_EVENT_LPM_PACKET_ARRIVAL0 IPSec SPI(보안 매개 변수 인덱스) 값을 설명하는 정보를 포함합니다.
FWPM_NET_EVENT_SUBSCRIPTION0 네트워크 이벤트에 대한 알림을 구독하는 데 사용되는 정보를 저장합니다.
FWPM_NET_EVENT0 모든 이벤트 유형에 대한 정보를 포함합니다. (FWPM_NET_EVENT0)
FWPM_NET_EVENT1 모든 이벤트 유형에 대한 정보를 포함합니다. (FWPM_NET_EVENT1)

FWPM_NET_EVENT2 모든 이벤트 유형에 대한 정보를 포함합니다. (FWPM_NET_EVENT2)
FWPM_NET_EVENT3 모든 이벤트 유형에 대한 정보를 포함합니다. (FWPM_NET_EVENT3)
FWPM_NET_EVENT4 모든 이벤트 유형에 대한 정보를 포함합니다. FWPM_NET_EVENT3 및 FWPM_NET_EVENT2 사용할 수 있습니다. Windows 7의 경우 FWPM_NET_EVENT1 사용할 수 있습니다. Windows Vista의 경우 FWPM_NET_EVENT0 사용할 수 있습니다.
FWPM_NET_EVENT5 모든 이벤트 유형에 대한 정보를 포함합니다. FWPM_NET_EVENT4, FWPM_NET_EVENT3 및 FWPM_NET_EVENT2 사용할 수 있습니다. Windows 7의 경우 FWPM_NET_EVENT1 사용할 수 있습니다. Windows Vista의 경우 FWPM_NET_EVENT0 사용할 수 있습니다.
FWPM_NETWORK_CONNECTION_POLICY_SETTING0 연결 정책 설정에 대한 형식 및 값 쌍을 저장합니다.
FWPM_NETWORK_CONNECTION_POLICY_SETTINGS0 해당 배열의 요소 수와 함께 FWPM_NETWORK_CONNECTION_POLICY_SETTING0 값의 배열을 저장합니다.
FWPM_PROVIDER_CHANGE0 구독자에게 디스패치된 알림을 변경합니다. (FWPM_PROVIDER_CHANGE0)
FWPM_PROVIDER_CONTEXT_CHANGE0 구독자에게 디스패치된 알림을 변경합니다. (FWPM_PROVIDER_CONTEXT_CHANGE0)
FWPM_PROVIDER_CONTEXT_ENUM_TEMPLATE0 공급자 컨텍스트를 열거하는 데 사용됩니다.
FWPM_PROVIDER_CONTEXT_SUBSCRIPTION0 변경 알림을 구독하는 데 사용됩니다. (FWPM_PROVIDER_CONTEXT_SUBSCRIPTION0)
FWPM_PROVIDER_CONTEXT0 공급자 컨텍스트와 연결된 상태를 저장합니다. (FWPM_PROVIDER_CONTEXT0)

FWPM_PROVIDER_CONTEXT1 공급자 컨텍스트와 연결된 상태를 저장합니다. (FWPM_PROVIDER_CONTEXT1)
FWPM_PROVIDER_CONTEXT2 공급자 컨텍스트와 연결된 상태를 저장합니다. (FWPM_PROVIDER_CONTEXT2)
FWPM_PROVIDER_CONTEXT3 공급자 컨텍스트와 연결된 상태를 저장합니다. FWPM_PROVIDER_CONTEXT0, FWPM_PROVIDER_CONTEXT1 및 FWPM_PROVIDER_CONTEXT2 사용할 수 있습니다.
FWPM_PROVIDER_ENUM_TEMPLATE0 공급자를 열거하는 데 사용됩니다.
FWPM_PROVIDER_SUBSCRIPTION0 변경 알림을 구독하는 데 사용됩니다. (FWPM_PROVIDER_SUBSCRIPTION0)
FWPM_PROVIDER0 정책 공급자와 연결된 상태를 저장합니다.
FWPM_SESSION_ENUM_TEMPLATE0 세션을 열거하는 데 사용됩니다.
FWPM_SESSION0 클라이언트 세션과 연결된 상태를 저장합니다.
FWPM_STATISTICS0 특정 계층에서 연결과 관련된 통계를 저장합니다.
FWPM_SUBLAYER_CHANGE0 구독자에게 디스패치된 알림을 변경합니다. (FWPM_SUBLAYER_CHANGE0)
FWPM_SUBLAYER_ENUM_TEMPLATE0 하위 계층을 열거하는 데 사용됩니다.
FWPM_SUBLAYER_SUBSCRIPTION0 변경 알림을 구독하는 데 사용됩니다. (FWPM_SUBLAYER_SUBSCRIPTION0)

FWPM_SUBLAYER0 하위 계층과 연결된 상태를 저장합니다.
FWPM_SYSTEM_PORTS_BY_TYPE0 FWPM_SYSTEM_PORTS_BY_TYPE0 구조체입니다.
FWPM_SYSTEM_PORTS0 FWPM_SYSTEM_PORTS0 구조체입니다.
FWPM_VSWITCH_EVENT_SUBSCRIPTION0 vSwitch 이벤트에 대한 알림을 구독하는 데 사용되는 정보를 저장합니다.
FWPM_VSWITCH_EVENT0 vSwitch 이벤트에 대한 정보를 포함합니다.

열거형

테이블 확장

DL_ADDRESS_TYPE DL_ADDRESS_TYPE 열거형(fwpmtypes.h)은 데이터 링크 계층 주소의 형식을 지정합니다.
FWPM_APPC_NETWORK_CAPABILITY_TYPE 해당 개체 또는 트래픽과 연결된 앱 컨테이너 네트워크 기능의 유형을 지정합니다.
FWPM_CHANGE_TYPE 구독자에게 변경 알림을 디스패치할 때 사용됩니다.
FWPM_CONNECTION_EVENT_TYPE 연결 개체 변경 이벤트의 유형을 지정합니다.
FWPM_ENGINE_OPTION 필터 엔진에 대한 구성 가능한 옵션입니다.

FWPM_FIELD_TYPE 필드의 데이터를 해석하는 방법에 대한 추가 정보를 제공합니다.
FWPM_NET_EVENT_TYPE net 이벤트의 유형입니다.
FWPM_PROVIDER_CONTEXT_TYPE BFE(기본 필터링 엔진)에 저장할 수 있는 공급자 컨텍스트 유형입니다.
FWPM_SERVICE_STATE 필터 엔진의 현재 상태를 지정합니다.
FWPM_SYSTEM_PORT_TYPE FWPM_SYSTEM_PORT_TYPE 열거형 형식입니다.
FWPM_VSWITCH_EVENT_TYPE vSwitch 이벤트의 형식을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

fwpmu.h 헤더

아티클2023. 08. 23.

이 헤더는 Windows 필터링 플랫폼에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 필터링 플랫폼

fwpmu.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

FwpmCalloutAdd0 시스템에 새 콜아웃 개체를 추가합니다.
FwpmCalloutCreateEnumHandle0 일련의 콜아웃 개체를 열거하는 데 사용되는 핸들을 만듭니다.
FwpmCalloutDeleteById0 시스템에서 설명선 개체를 제거합니다. (FwpmCalloutDeleteById0)
FwpmCalloutDeleteByKey0 시스템에서 설명선 개체를 제거합니다. (FwpmCalloutDeleteByKey0)
FwpmCalloutDestroyEnumHandle0 FwpmCalloutCreateEnumHandle0에서 반환된 핸들을 해제합니다.
FwpmCalloutEnum0 설명선 열거자에서 결과의 다음 페이지를 반환합니다.
FwpmCalloutGetById0 설명선 개체를 검색합니다. (FwpmCalloutGetById0)
FwpmCalloutGetByKey0 설명선 개체를 검색합니다. (FwpmCalloutGetByKey0)
FwpmCalloutGetSecurityInfoByKey0

설명자 개체에 대한 보안 설명자의 복사본을 검색합니다.
FwpmCalloutSetSecurityInfoByKey0 설명선 개체의 보안 설명자에 지정된 보안 정보를 설정합니다.
FwpmCalloutSubscribeChanges0 특정 설명선의 변경 내용에 대한 알림 배달을 요청하는 데 사용됩니다.
FwpmCalloutSubscriptionsGet0 현재 모든 설명선 변경 알림 구독의 배열을 검색합니다.
FwpmCalloutUnsubscribeChanges0 설명선 변경 구독을 취소하고 변경 알림 수신을 중지하는 데 사용됩니다.
FwpmConnectionCreateEnumHandle0 연결 개체 집합을 열거하는 데 사용되는 핸들을 만듭니다.
FwpmConnectionDestroyEnumHandle0 FwpmConnectionCreateEnumHandle0에서 반환된 핸들을 해제합니다.
FwpmConnectionEnum0 연결 개체 열거자에서 결과의 다음 페이지를 반환합니다.
FwpmConnectionGetById0 연결 개체를 검색합니다.
FwpmConnectionGetSecurityInfo0 연결 개체 변경 이벤트에 대한 보안 설명자의 복사본을 검색합니다.
FwpmConnectionSetSecurityInfo0 연결 개체 변경 이벤트에 대한 보안 설명자에 지정된 보안 정보를 설정합니다.
FwpmConnectionSubscribe0 연결 개체의 변경 내용에 대한 알림 배달을 요청하는 데 사용됩니다.
FwpmConnectionSubscriptionsGet0 현재 모든 연결 개체 변경 알림 구독의 배열을 검색합니다.

FwpmConnectionUnsubscribe0 연결 개체 변경 이벤트 구독을 취소하고 알림 수신을 중지하는 데 사용됩니다.
FwpmDynamicKeywordSubscribe0 특정 동적 키워드(keyword) 주소(FW_DYNAMIC_KEYWORD_ADDRESS0) 개체의 변경 내용에 대한 알림 배달을 요청합니다.
FwpmDynamicKeywordUnsubscribe0 특정 동적 키워드(keyword) 주소(FW_DYNAMIC_KEYWORD_ADDRESS0) 개체의 변경 내용에 대한 알림 배달을 취소합니다.
FwpmEngineClose0 필터 엔진에 대한 세션을 닫습니다.
FwpmEngineGetOption0 필터 엔진 옵션을 검색합니다.
FwpmEngineGetSecurityInfo0 필터 엔진에 대한 보안 설명자의 복사본을 검색합니다.
FwpmEngineOpen0 필터 엔진에 대한 세션을 엽니다.
FwpmEngineSetOption0 필터 엔진 설정을 변경합니다.
FwpmEngineSetSecurityInfo0 필터 엔진의 보안 설명자에 지정된 보안 정보를 설정합니다.
FwpmFilterAdd0 시스템에 새 필터 개체를 추가합니다.
FwpmFilterCreateEnumHandle0 필터 개체 집합을 열거하는 데 사용되는 핸들을 만듭니다.
FwpmFilterDeleteById0 시스템에서 필터 개체를 제거합니다. (FwpmFilterDeleteById0)

FwpmFilterDeleteByKey0 시스템에서 필터 개체를 제거합니다. (FwpmFilterDeleteByKey0)
FwpmFilterDestroyEnumHandle0 FwpmFilterCreateEnumHandle0에서 반환된 핸들을 해제합니다. (FwpmFilterDestroyEnumHandle0)
FwpmFilterEnum0 필터 열거자에서 결과의 다음 페이지를 반환합니다.
FwpmFilterGetById0 필터 개체를 검색합니다. (FwpmFilterGetById0)
FwpmFilterGetByKey0 필터 개체를 검색합니다. (FwpmFilterGetByKey0)
FwpmFilterGetSecurityInfoByKey0 필터 개체에 대한 보안 설명자의 복사본을 검색합니다.
FwpmFilterSetSecurityInfoByKey0 필터 개체의 보안 설명자에 지정된 보안 정보를 설정합니다.
FwpmFilterSubscribeChanges0 특정 필터의 변경 내용에 대한 알림 배달을 요청하는 데 사용됩니다.
FwpmFilterSubscriptionsGet0 모든 현재 필터 변경 알림 구독의 배열을 검색합니다.
FwpmFilterUnsubscribeChanges0 필터 변경 구독을 취소하고 변경 알림 수신을 중지하는 데 사용됩니다.
FwpmFreeMemory0 WFP(Windows 필터링 플랫폼) 함수에서 할당한 메모리 리소스를 해제하는 데 사용됩니다.
FwpmGetAppIdFromFileName0 파일 이름에서 애플리케이션 식별자를 검색합니다.
FwpmIPsecTunnelAdd0

시스템에 새 IPsec(인터넷 프로토콜 보안) 터널 모드 정책을 추가합니다. (FwpmIpsecTunnelAdd0)
FwpmIPsecTunnelAdd1 시스템에 새 IPsec(인터넷 프로토콜 보안) 터널 모드 정책을 추가합니다. (FwpmIpsecTunnelAdd1)
FwpmIPsecTunnelAdd2 시스템에 새 IPsec(인터넷 프로토콜 보안) 터널 모드 정책을 추가합니다. (FwpmIPsecTunnelAdd2)
FwpmIPsecTunnelDeleteByKey0 시스템에서 IPsec(인터넷 프로토콜 보안) 터널 모드 정책을 제거합니다.
FwpmLayerCreateEnumHandle0 레이어 개체 집합을 열거하는 데 사용되는 핸들을 만듭니다.
FwpmLayerDestroyEnumHandle0 FwpmFilterCreateEnumHandle0에서 반환된 핸들을 해제합니다. (FwpmLayerDestroyEnumHandle0)
FwpmLayerEnum0 레이어 열거자에서 결과의 다음 페이지를 반환합니다.
FwpmLayerGetById0 레이어 개체를 검색합니다. (FwpmLayerGetById0)
FwpmLayerGetByKey0 레이어 개체를 검색합니다. (FwpmLayerGetByKey0)
FwpmLayerGetSecurityInfoByKey0 계층 개체에 대한 보안 설명자의 복사본을 검색합니다.
FwpmLayerSetSecurityInfoByKey0 계층 개체의 보안 설명자에 지정된 보안 정보를 설정합니다.
FwpmNetEventCreateEnumHandle0 네트워크 이벤트 집합을 열거하는 데 사용되는 핸들을 만듭니다.
FwpmNetEventDestroyEnumHandle0 FwpmNetEventCreateEnumHandle0에서 반환된 핸들을 해제합니다.

FwpmNetEventEnum0 네트워크 이벤트 열거자에서 결과의 다음 페이지를 반환합니다. (FwpmNetEventEnum0)
FwpmNetEventEnum1 네트워크 이벤트 열거자에서 결과의 다음 페이지를 반환합니다. (FwpmNetEventEnum1)
FwpmNetEventEnum2 네트워크 이벤트 열거자에서 결과의 다음 페이지를 반환합니다. (FwpmNetEventEnum2)
FwpmNetEventEnum3 네트워크 이벤트 열거자에서 결과의 다음 페이지를 반환합니다. (FwpmNetEventEnum3)
FwpmNetEventsGetSecurityInfo0 FwpmNetEventsGetSecurityInfo0 함수는 네트워크 이벤트 개체에 대한 보안 설명자의 복사본을 검색합니다.
FwpmNetEventsSetSecurityInfo0 네트워크 이벤트 개체의 보안 설명자에 지정된 보안 정보를 설정합니다.
FwpmNetEventSubscribe0 특정 net 이벤트에 대한 알림 배달을 요청하는 데 사용됩니다. (FwpmNetEventSubscribe0)
FwpmNetEventSubscribe1 특정 net 이벤트에 대한 알림 배달을 요청하는 데 사용됩니다. (FwpmNetEventSubscribe1)
FwpmNetEventSubscribe2 특정 net 이벤트에 대한 알림 배달을 요청하는 데 사용됩니다. (FwpmNetEventSubscribe2)
FwpmNetEventSubscriptionsGet0 현재 모든 순 이벤트 알림 구독의 배열을 검색합니다.
FwpmNetEventUnsubscribe0 순 이벤트 구독을 취소하고 알림 수신을 중지하는 데 사용됩니다.
FwpmProviderAdd0 시스템에 새 공급자를 추가합니다.

FwpmProviderContextAdd0 시스템에 새 공급자 컨텍스트를 추가합니다. (FwpmProviderContextAdd0)
FwpmProviderContextAdd1 시스템에 새 공급자 컨텍스트를 추가합니다. (FwpmProviderContextAdd1)
FwpmProviderContextAdd2 시스템에 새 공급자 컨텍스트를 추가합니다. (FwpmProviderContextAdd2)
FwpmProviderContextCreateEnumHandle0 공급자 컨텍스트 집합을 열거하는 데 사용되는 핸들을 만듭니다.
FwpmProviderContextDeleteById0 시스템 에서 공급자 컨텍스트를 제거합니다.
FwpmProviderContextDeleteByKey0 시스템에서 공급자 컨텍스트를 제거합니다.
FwpmProviderContextDestroyEnumHandle0 FwpmProviderContextCreateEnumHandle0에서 반환된 핸들을 해제합니다.
FwpmProviderContextEnum0 공급자 컨텍스트 열거자에서 결과의 다음 페이지를 반환합니다. (FwpmProviderContextEnum0)
FwpmProviderContextEnum1 공급자 컨텍스트 열거자에서 결과의 다음 페이지를 반환합니다. (FwpmProviderContextEnum1)
FwpmProviderContextEnum2 공급자 컨텍스트 열거자에서 결과의 다음 페이지를 반환합니다. (FwpmProviderContextEnum2)
FwpmProviderContextGetById0 공급자 컨텍스트를 검색합니다. (FwpmProviderContextGetById0)
FwpmProviderContextGetById1 공급자 컨텍스트를 검색합니다. (FwpmProviderContextGetById1)
FwpmProviderContextGetById2

공급자 컨텍스트를 검색합니다. (FwpmProviderContextGetById2)
FwpmProviderContextGetByKey0 공급자 컨텍스트를 검색합니다. (FwpmProviderContextGetByKey0)
FwpmProviderContextGetByKey1 공급자 컨텍스트를 검색합니다. (FwpmProviderContextGetByKey1)
FwpmProviderContextGetByKey2 공급자 컨텍스트를 검색합니다. (FwpmProviderContextGetByKey2)
FwpmProviderContextGetSecurityInfoByKey0 공급자 컨텍스트 개체에 대한 보안 설명자의 복사본을 검색합니다.
FwpmProviderContextSetSecurityInfoByKey0 공급자 컨텍스트 개체의 보안 설명자에 지정된 보안 정보를 설정합니다.
FwpmProviderContextSubscribeChanges0 특정 공급자 컨텍스트의 변경 내용에 대한 알림 배달을 요청하는 데 사용됩니다.
FwpmProviderContextSubscriptionsGet0 모든 현재 공급자 컨텍스트 변경 알림 구독의 배열을 검색합니다.
FwpmProviderContextUnsubscribeChanges0 공급자 컨텍스트 변경 구독을 취소하고 변경 알림 수신을 중지하는 데 사용됩니다.
FwpmProviderCreateEnumHandle0 공급자 집합을 열거하는 데 사용되는 핸들을 만듭니다.
FwpmProviderDeleteByKey0 시스템에서 공급자를 제거합니다.
FwpmProviderDestroyEnumHandle0 FwpmProviderCreateEnumHandle0에서 반환된 핸들을 해제합니다.
FwpmProviderEnum0 공급자 열거자에서 결과의 다음 페이지를 반환합니다.

FwpmProviderGetByKey0 공급자를 검색합니다.
FwpmProviderGetSecurityInfoByKey0 공급자 개체에 대한 보안 설명자의 복사본을 검색합니다.
FwpmProviderSetSecurityInfoByKey0 공급자 개체의 보안 설명자에 지정된 보안 정보를 설정합니다.
FwpmProviderSubscribeChanges0 특정 공급자의 변경 내용에 대한 알림 배달을 요청하는 데 사용됩니다.
FwpmProviderSubscriptionsGet0 모든 현재 공급자 변경 알림 구독의 배열을 검색합니다.
FwpmProviderUnsubscribeChanges0 공급자 변경 구독을 취소하고 변경 알림 수신을 중지하는 데 사용됩니다.
FwpmSessionCreateEnumHandle0 세션 개체 집합을 열거하는 데 사용되는 핸들을 만듭니다.
FwpmSessionDestroyEnumHandle0 FwpmSessionCreateEnumHandle0에서 반환된 핸들을 해제합니다.
FwpmSessionEnum0 세션 열거자에서 결과의 다음 페이지를 반환합니다.
FwpmSubLayerAdd0 시스템에 새 하위 계층을 추가합니다.
FwpmSubLayerCreateEnumHandle0 하위 계층 집합을 열거하는 데 사용되는 핸들을 만듭니다.
FwpmSubLayerDeleteByKey0 키로 시스템에서 하위 계층을 삭제합니다.
FwpmSubLayerDestroyEnumHandle0

FwpmSubLayerCreateEnumHandle0에서 반환된 핸들을 해제합니다.
FwpmSubLayerEnum0 하위 계층 열거자에서 결과의 다음 페이지를 반환합니다.
FwpmSubLayerGetByKey0 해당 키로 하위 계층을 검색합니다.
FwpmSubLayerGetSecurityInfoByKey0 하위 계층에 대한 보안 설명자의 복사본을 검색합니다.
FwpmSubLayerSetSecurityInfoByKey0 하위 계층의 보안 설명자에 지정된 보안 정보를 설정합니다.
FwpmSubLayerSubscribeChanges0 특정 하위 계층의 변경 내용에 대한 알림 배달을 요청하는 데 사용됩니다.
FwpmSubLayerSubscriptionsGet0 모든 현재 하위 계층 변경 알림 구독의 배열을 검색합니다.
FwpmSubLayerUnsubscribeChanges0 하위 계층 변경 구독을 취소하고 변경 알림 수신을 중지하는 데 사용됩니다.
FwpmSystemPortsGet0 모든 시스템 포트 형식의 배열을 검색합니다.
FwpmSystemPortsSubscribe0 특정 시스템 포트에 대한 알림 배달을 요청하는 데 사용됩니다.
FwpmSystemPortsUnsubscribe0 시스템 포트 구독을 취소하고 알림 수신을 중지하는 데 사용됩니다.
FwpmTransactionAbort0 현재 세션 내의 현재 트랜잭션이 중단되고 롤백되도록 합니다.
FwpmTransactionBegin0 현재 세션 내에서 명시적 트랜잭션을 시작합니다.

FwpmTransactionCommit0 현재 세션 내에서 현재 트랜잭션을 커밋합니다.
FwpmvSwitchEventsGetSecurityInfo0 vSwitch 이벤트에 대한 보안 설명자의 복사본을 검색합니다.
FwpmvSwitchEventsSetSecurityInfo0 vSwitch 이벤트에 대한 보안 설명자에 지정된 보안 정보를 설정합니다.
FwpmvSwitchEventSubscribe0 특정 vSwitch 이벤트에 대한 알림 배달을 요청하는 데 사용됩니다.
FwpmvSwitchEventUnsubscribe0 vSwitch 이벤트 구독을 취소하고 알림 수신을 중지하는 데 사용됩니다.
IkeextGetStatistics0 IKE(인터넷 키 교환) 및 인증된 인터넷 프로토콜(AuthIP) 통계를 검색합니다. (IkeextGetStatistics0)
IkeextGetStatistics1 IKE(인터넷 키 교환) 및 인증된 인터넷 프로토콜(AuthIP) 통계를 검색합니다. (IkeextGetStatistics1)
IkeextSaCreateEnumHandle0 IKE(인터넷 키 교환) 및 인증된 인터넷 프로토콜(AuthIP) SA(보안 연결) 개체 집합을 열거하는 데 사용되는 핸들을 만듭니다.
IkeextSaDbGetSecurityInfo0 IkeextSaDbGetSecurityInfo0 함수는 SA(보안 연결) 데이터베이스에 대한 보안 설명자의 복사본을 검색합니다.
IkeextSaDbSetSecurityInfo0 IkeextSaDbSetSecurityInfo0 함수는 IKE/AuthIP 보안 연결 데이터베이스의 보안 설명자에 지정된 보안 정보를 설정합니다.
IkeextSaDeleteById0 IkeextSaDeleteById0 함수는 데이터베이스에서 SA(보안 연결)를 제거합니다.
IkeextSaDestroyEnumHandle0

IkeextSaCreateEnumHandle0에서 반환된 핸들을 해제합니다.
IkeextSaEnum0 IKE/AuthIP SA(보안 연결) 열거자에서 결과의 다음 페이지를 반환합니다. (IkeextSaEnum0)
IkeextSaEnum1 IKE/AuthIP SA(보안 연결) 열거자에서 결과의 다음 페이지를 반환합니다. (IkeextSaEnum1)
IkeextSaEnum2 IKE/AuthIP SA(보안 연결) 열거자에서 결과의 다음 페이지를 반환합니다. (IkeextSaEnum2)
IkeextSaGetById0 데이터베이스에서 IKE/AuthIP SA(보안 연결)를 검색합니다. (IkeextSaGetById0)
IkeextSaGetById1 데이터베이스에서 IKE/AuthIP SA(보안 연결)를 검색합니다. (IkeextSaGetById1)
IkeextSaGetById2 데이터베이스에서 IKE/AuthIP SA(보안 연결)를 검색합니다. (IkeextSaGetById2)
IPsecDospGetSecurityInfo0 IPsec DoS Protection 데이터베이스에 대한 보안 설명자의 복사본을 검색합니다.
IPsecDospGetStatistics0 IPsecDospGetStatistics0 함수는 IPsec(인터넷 프로토콜 보안) DoS Protection 통계를 검색합니다.
IPsecDospSetSecurityInfo0 IPsecDospSetSecurityInfo0 함수는 IPsec DoS Protection 데이터베이스의 보안 설명자에 지정된 보안 정보를 설정합니다.
IPsecDospStateCreateEnumHandle0 IPsec DoS Protection 개체 집합을 열거하는 데 사용되는 핸들을 만듭니다.
IPsecDospStateDestroyEnumHandle0 IPsecDospStateCreateEnumHandle0에서 반환된 핸들을 해제합니다.
IPsecDospStateEnum0

IPsec DoS Protection 상태 열거자에서 결과의 다음 페이지를 반환합니다. 각 IPsec DoS Protection 상태 항목은 IPsec DoS Protection 인증 검사를 성공적으로 통과한 흐름에 해당합니다.
IPsecGetStatistics0 IPsec(인터넷 프로토콜 보안) 통계를 검색합니다. (IPsecGetStatistics0)
IPsecGetStatistics1 IPsec(인터넷 프로토콜 보안) 통계를 검색합니다. (IPsecGetStatistics1)
IPsecKeyManagerAddAndRegister0 IPsec에 TIA(신뢰할 수 있는 중간 에이전트)를 등록합니다.
IPsecKeyManagerGetSecurityInfoByKey0 키 관리자에 대한 액세스를 제어하는 보안 설명자의 복사본을 검색합니다.
IPsecKeyManagerSetSecurityInfoByKey0 키 관리자에 대한 액세스를 제어하는 보안 설명자에서 지정된 보안 정보를 설정합니다.
IPsecKeyManagersGet0 현재 신뢰할 수 있는 IA(중간 에이전트) 목록을 반환합니다.
IPsecKeyManagerUnregisterAndDelete0 이전에 IPsec에 등록된 TIA(신뢰할 수 있는 중간 에이전트)의 등록을 취소합니다.
IPsecSaContextAddInbound0 IPsecSaContextAddInbound0 함수는 기존 SA 컨텍스트에 인바운드 SA(IPsec 보안 연결) 번들을 추가합니다. 참고 IPsecSaContextAddInbound0은 Windows Vista에서 사용되는 IPsecSaContextAddInbound의 특정 구현입니다.
IPsecSaContextAddInbound1 IPsecSaContextAddInbound1 함수는 기존 SA 컨텍스트에 인바운드 IPsec SA(보안 연결) 번들을 추가합니다. 참고 IPsecSaContextAddInbound1은 Windows 7 이상에서 사용되는 IPsecSaContextAddInbound의 특정 구현입니다.
IPsecSaContextAddOutbound0 IPsecSaContextAddOutbound0 함수는 기존 SA 컨텍스트에 아웃바운드 SA(IPsec 보안 연결) 번들을 추가합니다. 참고 IPsecSaContextAddOutbound0은 Windows Vista에서 사용되는 IPsecSaContextAddOutbound의 특정 구현입니다.

IPsecSaContextAddOutbound1 IPsecSaContextAddOutbound1 함수는 기존 SA 컨텍스트에 아웃바운드 SA(IPsec 보안 연결) 번들을 추가합니다. 참고 IPsecSaContextAddOutbound1은 Windows 7 이상에서 사용되는 IPsecSaContextAddOutbound의 특정 구현입니다.
IPsecSaContextCreate0 IPsec SA(보안 연결) 컨텍스트를 만듭니다. (IPsecSaContextCreate0)
IPsecSaContextCreate1 IPsec SA(보안 연결) 컨텍스트를 만듭니다. (IPsecSaContextCreate1)
IPsecSaContextCreateEnumHandle0 SA(IPsec 보안 연결) 컨텍스트 개체 집합을 열거하는 데 사용되는 핸들을 만듭니다.
IPsecSaContextDeleteById0 IPsec SA(보안 연결) 컨텍스트를 삭제합니다.
IPsecSaContextDestroyEnumHandle0 IPsecSaContextCreateEnumHandle0에서 반환된 핸들을 해제합니다.
IPsecSaContextEnum0 IPsec SA(보안 연결) 컨텍스트 열거자에서 결과의 다음 페이지를 반환합니다. (IPsecSaContextEnum0)
IPsecSaContextEnum1 IPsec SA(보안 연결) 컨텍스트 열거자에서 결과의 다음 페이지를 반환합니다. (IPsecSaContextEnum1)
IPsecSaContextExpire0 IPsec SA(보안 연결) 컨텍스트가 만료되어야 했음을 나타냅니다.
IPsecSaContextGetById0 IPsec SA(보안 연결) 컨텍스트를 검색합니다. (IPsecSaContextGetById0)
IPsecSaContextGetById1 IPsec SA(보안 연결) 컨텍스트를 검색합니다. (IPsecSaContextGetById1)

IPsecSaContextGetSpi0 SA(보안 연결) 컨텍스트에 대한 SPI(보안 매개 변수 인덱스)를 검색합니다. (IPsecSaContextGetSpi0)
IPsecSaContextGetSpi1 SA(보안 연결) 컨텍스트에 대한 SPI(보안 매개 변수 인덱스)를 검색합니다. (IPsecSaContextGetSpi1)
IPsecSaContextSetSpi0 SA(보안 연결) 컨텍스트에 대한 SPI(보안 매개 변수 인덱스)를 설정합니다.
IPsecSaContextSubscribe0 특정 IPsec SA(보안 연결) 컨텍스트와 관련된 알림 배달을 요청하는 데 사용됩니다.
IPsecSaContextSubscriptionsGet0 현재 모든 SA(IPsec 보안 연결) 변경 알림 구독의 배열을 검색합니다.
IPsecSaContextUnsubscribe0 IPsec SA(보안 연결) 변경 구독을 취소하고 변경 알림 수신을 중지하는 데 사용됩니다.
IPsecSaContextUpdate0 IPsec SA(보안 연결) 컨텍스트를 업데이트.
IPsecSaCreateEnumHandle0 IPsec(인터넷 프로토콜 보안) SA(보안 연결) 개체 집합을 열거하는 데 사용되는 핸들을 만듭니다.
IPsecSaDbGetSecurityInfo0 IPsec SA(보안 연결) 데이터베이스에 대한 보안 설명자의 복사본을 검색합니다.
IPsecSaDbSetSecurityInfo0 IPsec 보안 연결 데이터베이스의 보안 설명자에 지정된 보안 정보를 설정합니다.
IPsecSaDestroyEnumHandle0 IPsecSaCreateEnumHandle0에서 반환된 핸들을 해제합니다.
IPsecSaEnum0 IPsec SA(보안 연결) 열거자에서 결과의 다음 페이지를 반환합니다. (IPsecSaEnum0)
IPsecSaEnum1

IPsec SA(보안 연결) 열거자에서 결과의 다음 페이지를 반환합니다. (IPsecSaEnum1)

콜백 함수

FWPM_CALLOUT_CHANGE_CALLBACK0 설명선 변경 알림 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다.
FWPM_CONNECTION_CALLBACK0 연결 개체 구독 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다.
FWPM_DYNAMIC_KEYWORD_CALLBACK0 구현하는 콜백 함수는 동적 키워드(keyword) 주소(FW_DYNAMIC_KEYWORD_ADDRESS0) 개체의 변경 내용에 대한 알림과 함께 호출됩니다.
FWPM_FILTER_CHANGE_CALLBACK0 필터 변경 알림 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다.
FWPM_NET_EVENT_CALLBACK0 net 이벤트 구독 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다. (FWPM_NET_EVENT_CALLBACK0)
FWPM_NET_EVENT_CALLBACK1 net 이벤트 구독 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다. (FWPM_NET_EVENT_CALLBACK1)
FWPM_NET_EVENT_CALLBACK2 net 이벤트 구독 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다. (FWPM_NET_EVENT_CALLBACK2)
FWPM_PROVIDER_CHANGE_CALLBACK0 공급자 변경 알림 프로세스에 사용자 지정 동작을 추가합니다.
FWPM_PROVIDER_CONTEXT_CHANGE_CALLBACK0 공급자 컨텍스트 변경 알림 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다.
FWPM_SUBLAYER_CHANGE_CALLBACK0

하위 계층 변경 알림 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다.
FWPM_SYSTEM_PORTS_CALLBACK0 시스템 포트 구독 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다.
FWPM_VSWITCH_EVENT_CALLBACK0 vSwitch 이벤트 구독 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다.
IPSEC_KEY_MANAGER_DICTATE_KEY0 TIA(신뢰할 수 있는 중개자 에이전트)에서 협상 중인 SA에 대한 키를 지시하는 데 사용됩니다.
IPSEC_KEY_MANAGER_KEY_DICTATION_CHECK0 TIA(신뢰할 수 있는 중개자 에이전트)가 협상 중인 SA의 키를 지시할지 여부를 나타냅니다.
IPSEC_KEY_MANAGER_NOTIFY_KEY0 협상 중인 SA에 대한 키를 IA(신뢰할 수 있는 중간 에이전트)에 알리는 데 사용됩니다.
IPSEC_SA_CONTEXT_CALLBACK0 IPsec SA(보안 연결) 컨텍스트 구독 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다.

구조체

IPSEC_KEY_MANAGER_CALLBACKS0 SA 협상의 다양한 단계에서 IPsec에서 호출해야 하는 콜백 집합을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

fwpstypes.h 헤더

아티클2024. 05. 07.

이 헤더는 네트워크 드라이버, Windows Vista 및 이후 버전에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 드라이버, Windows Vista 이상

fwpstypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

테이블 확장

FWPS_ACTION0 FWPS_ACTION0 구조는 필터의 모든 필터링 조건이 true인 경우 필터 엔진이 수행하는 런타임 작업을 지정합니다. FWPS_ACTION0 특정 버전의 FWPS_ACTION.
FWPS_ALE_ENDPOINT_ENUM_TEMPLATE0 FWPS_ALE_ENDPOINT_ENUM_TEMPLATE0 구조는 ALE(애플리케이션 계층 적용) 엔드포인트를 열거할 템플릿을 지정합니다. FWPS_ALE_ENDPOINT_ENUM_TEMPLATE0 특정 버전의 FWPS_ALE_ENDPOINT_ENUM_TEMPLATE.
FWPS_ALE_ENDPOINT_PROPERTIES0 FWPS_ALE_ENDPOINT_PROPERTIES0 구조는 ALE(애플리케이션 계층 적용) 엔드포인트의 속성을 지정합니다. FWPS_ALE_ENDPOINT_PROPERTIES0 특정 버전의 FWPS_ALE_ENDPOINT_PROPERTIES.
FWPS_CLASSIFY_OUT0 FWPS_CLASSIFY_OUT0 구조체는 설명선의 classifyFn 설명선 함수의 호출자에게 반환되는 데이터를 정의합니다. FWPS_CLASSIFY_OUT0 특정 버전의 FWPS_CLASSIFY_OUT.
FWPS_DISCARD_METADATA0 FWPS_DISCARD_METADATA0 구조는 필터 엔진, 네트워크 계층 또는 전송 계층에서 삭제된 데이터를 설명합니다. FWPS_DISCARD_METADATA0 특정 버전의 FWPS_DISCARD_METADATA.
FWPS_FILTER_CONDITION0 FWPS_FILTER_CONDITION0 구조체는 필터에 대한 런타임 필터링 조건을 정의합니다. FWPS_FILTER_CONDITION0 특정 버전의 FWPS_FILTER_CONDITION.

FWPS_FILTER0 FWPS_FILTER0 구조체는 필터 엔진에서 런타임 필터를 정의합니다. 참고 FWPS_FILTER0 Windows Vista 이상에서 사용되는 특정 버전의 FWPS_FILTER.
FWPS_FILTER1 FWPS_FILTER1 구조체는 필터 엔진에서 런타임 필터를 정의합니다. FWPS_FILTER1 Windows 7 이상에서 사용되는 특정 버전의 FWPS_FILTER.
FWPS_FILTER2 FWPS_FILTER2 구조체는 필터 엔진에서 런타임 필터를 정의합니다. FWPS_FILTER2 Windows 8 이상에서 사용되는 특정 버전의 FWPS_FILTER.
FWPS_FILTER3 필터 엔진에서 런타임 필터를 정의합니다.
FWPS_INBOUND_FRAGMENT_METADATA0 FWPS_INBOUND_FRAGMENT_METADATA0 구조는 수신된 패킷 조각에 대한 조각 데이터를 설명합니다. FWPS_INBOUND_FRAGMENT_METADATA0 특정 버전의 FWPS_INBOUND_FRAGMENT_METADATA.
FWPS_INCOMING_VALUE0 FWPS_INCOMING_VALUE0 구조체는 개별 데이터 값을 정의합니다. FWPS_INCOMING_VALUE0 특정 버전의 FWPS_INCOMING_VALUE.
FWPS_INCOMING_VALUES0 FWPS_INCOMING_VALUES0 구조체는 필터 엔진이 설명선의 classifyFn 설명선 함수에 전달하는 데이터 값을 정의합니다. FWPS_INCOMING_VALUES0 특정 버전의 FWPS_INCOMING_VALUES.

열거형

테이블 확장

FWPS_DISCARD_MODULE0 FWPS_DISCARD_MODULE0 열거형 형식은 데이터를 삭제한 모듈의 형식을 지정합니다. FWPS_DISCARD_MODULE0 특정 버전의 FWPS_DISCARD_MODULE.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

fwpsu.h 헤더

아티클2024. 06. 12.

이 헤더는 Windows 필터링 플랫폼에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 필터링 플랫폼

fwpsu.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

FWPS_IS_L2_METADATA_FIELD_PRESENT TBD
FWPS_IS_METADATA_FIELD_PRESENT TBD
FwpsAleEndpointCreateEnumHandle0 다른 ALE(애플리케이션 계층 적용) 엔드포인트 함수와 함께 엔드포인트 데이터를 열거하는 데 사용할 수 있는 핸들을 만듭니다.
FwpsAleEndpointDestroyEnumHandle0 FwpsAleEndpointCreateEnumHandle0을 호출하여 만든 엔드포인트 열거형 핸들을 삭제합니다.
FwpsAleEndpointEnum0 ALE(애플리케이션 계층 적용) 엔드포인트를 열거합니다.
FwpsAleEndpointGetById0 ALE(애플리케이션 계층 적용) 엔드포인트에 대한 정보를 검색합니다.
FwpsAleEndpointGetSecurityInfo0 ALE(애플리케이션 계층 적용) 엔드포인트 열거 세션에 대한 보안 정보를 검색합니다.
FwpsAleEndpointSetSecurityInfo0 ALE(애플리케이션 계층 적용) 엔드포인트 열거 세션에 대한 보안 정보를 설정합니다.

FwpsOpenToken0 액세스 토큰을 엽니다.

콜백 함수

테이블 확장

FWPS_CALLOUT_BOOTTIME_CALLOUT_DELETE_NOTIFY_FN0 TBD
FWPS_CALLOUT_CLASSIFY_FN0 필터 엔진은 콜아웃에서 처리할 데이터가 있을 때마다 콜아웃의 classifyFn0 콜아웃 함수를 호출합니다.
FWPS_CALLOUT_CLASSIFY_FN1 필터 엔진은 콜아웃에서 처리할 데이터가 있을 때마다 콜아웃의 classifyFn1 콜아웃 함수를 호출합니다.
FWPS_CALLOUT_CLASSIFY_FN2 필터 엔진은 콜아웃에서 처리할 데이터가 있을 때마다 콜아웃의 classifyFn2 콜아웃 함수를 호출합니다.
FWPS_CALLOUT_CLASSIFY_FN3 TBD
FWPS_CALLOUT_FLOW_DELETE_NOTIFY_FN0 필터 엔진은 콜아웃의 flowDeleteFn 콜아웃 함수를 호출하여 콜아웃에서 처리 중인 데이터 흐름이 종료되고 있음을 콜아웃에 알립니다.
FWPS_CALLOUT_NOTIFY_FN0 필터 엔진은 콜아웃의 notifyFn0 콜아웃 함수를 호출하여 콜아웃 드라이버에 콜아웃과 연결된 이벤트에 대해 알립니다.
FWPS_CALLOUT_NOTIFY_FN1

필터 엔진은 콜아웃의 **notifyFn1** 콜아웃 함수를 호출하여 콜아웃 드라이버에 콜아웃과 연결된 이벤트에 대해 알립니다.
FWPS_CALLOUT_NOTIFY_FN2 필터 엔진은 콜아웃의 notifyFn2 콜아웃 함수를 호출하여 콜아웃 드라이버에 콜아웃과 연결된 이벤트에 대해 알립니다.
FWPS_CALLOUT_NOTIFY_FN3 TBD

구조체

테이블 확장

FWPS_INCOMING_METADATA_VALUES0 필터 엔진이 설명선의 classifyFn 설명선 함수에 전달하는 메타데이터 값을 정의합니다.

열거형

테이블 확장

FWPS_BUILTIN_LAYERS 기본 제공 런타임 필터링 계층 식별자를 지정하는 상수를 정의합니다. 각각은 크기가 64비트인 LUID(로컬 고유 식별자)로 표시됩니다.
FWPS_FIELDS_ALE_AUTH_CONNECT_V4 FWPS_LAYER_ALE_AUTH_CONNECT_V4 및 FWPS_LAYER_ALE_AUTH_CONNECT_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_AUTH_CONNECT_V6 FWPS_LAYER_ALE_AUTH_CONNECT_V6 및 FWPS_LAYER_ALE_AUTH_CONNECT_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_AUTH_LISTEN_V4

FWPS_LAYER_ALE_AUTH_LISTEN_V4 및 FWPS_LAYER_ALE_AUTH_LISTEN_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_AUTH_LISTEN_V6 FWPS_LAYER_ALE_AUTH_LISTEN_V6 및 FWPS_LAYER_ALE_AUTH_LISTEN_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_AUTH_RECV_ACCEPT_V4 FWPS_LAYER_ALE_AUTH_RECV_ACCEPT_V4 및 FWPS_LAYER_ALE_AUTH_RECV_ACCEPT_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_AUTH_RECV_ACCEPT_V6 FWPS_LAYER_ALE_AUTH_RECV_ACCEPT_V6 및 FWPS_LAYER_ALE_AUTH_RECV_ACCEPT_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_BIND_REDIRECT_V4 FWPS_LAYER_ALE_BIND_REDIRECT_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_BIND_REDIRECT_V6 FWPS_LAYER_ALE_BIND_REDIRECT_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_CONNECT_REDIRECT_V4 FWPS_LAYER_ALE_CONNECT_REDIRECT_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_CONNECT_REDIRECT_V6 FWPS_LAYER_ALE_CONNECT_REDIRECT_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_ENDPOINT_CLOSURE_V4 FWPS_LAYER_ALE_ENDPOINT_CLOSURE_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_ENDPOINT_CLOSURE_V6 FWPS_LAYER_ALE_ENDPOINT_CLOSURE_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_FLOW_ESTABLISHED_V4

FWPS_LAYER_ALE_FLOW_ESTABLISHED_V4 및 FWPS_LAYER_ALE_FLOW_ESTABLISHED_V4_DISCARD 런타임 필터링 계층에 대한 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_FLOW_ESTABLISHED_V6 FWPS_LAYER_ALE_FLOW_ESTABLISHED_V6 및 FWPS_LAYER_ALE_FLOW_ESTABLISHED_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_RESOURCE_ASSIGNMENT_V4 FWPS_LAYER_ALE_RESOURCE_ASSIGNMENT_V4 및 FWPS_LAYER_ALE_RESOURCE_ASSIGNMENT_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_RESOURCE_ASSIGNMENT_V6 FWPS_LAYER_ALE_RESOURCE_ASSIGNMENT_V6 및 FWPS_LAYER_ALE_RESOURCE_ASSIGNMENT_V6_DISCARD 런타임 필터링 계층에 대한 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_RESOURCE_RELEASE_V4 FWPS_LAYER_ALE_RESOURCE_RELEASE_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_ALE_RESOURCE_RELEASE_V6 FWPS_LAYER_ALE_RESOURCE_RELEASE_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_DATAGRAM_DATA_V4 FWPS_LAYER_ALE_RESOURCE_ASSIGNMENT_V4 및 FWPS_LAYER_ALE_RESOURCE_ASSIGNMENT_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_DATAGRAM_DATA_V6 FWPS_LAYER_ALE_RESOURCE_ASSIGNMENT_V6 및 FWPS_LAYER_ALE_RESOURCE_ASSIGNMENT_V6_DISCARD 런타임 필터링 계층에 대한 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_EGRESS_VSWITCH_ETHERNET FWPS_LAYER_EGRESS_VSWITCH_ETHERNET 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_EGRESS_VSWITCH_TRANSPORT_V4

FWPS_LAYER_EGRESS_VSWITCH_TRANSPORT_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_EGRESS_VSWITCH_TRANSPORT_V6 FWPS_LAYER_EGRESS_VSWITCH_TRANSPORT_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_IKEEXT_V4 FWPS_LAYER_IKEEXT_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_IKEEXT_V6 FWPS_LAYER_IKEEXT_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_INBOUND_ICMP_ERROR_V4 FWPS_LAYER_INBOUND_ICMP_ERROR_V4 및 FWPS_LAYER_INBOUND_ICMP_ERROR_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_INBOUND_ICMP_ERROR_V6 FWPS_LAYER_INBOUND_ICMP_ERROR_V6 및 FWPS_LAYER_INBOUND_ICMP_ERROR_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_INBOUND_IPPACKET_V4 FWPS_LAYER_INBOUND_IPPACKET_V4 및 FWPS_LAYER_INBOUND_IPPACKET_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_INBOUND_IPPACKET_V6 FWPS_LAYER_INBOUND_IPPACKET_V6 및 FWPS_LAYER_INBOUND_IPPACKET_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_INBOUND_MAC_FRAME_ETHERNET FWPS_LAYER_INBOUND_MAC_FRAME_ETHERNET 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_INBOUND_MAC_FRAME_NATIVE FWPS_LAYER_INBOUND_MAC_FRAME_ETHERNET 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_INBOUND_MAC_FRAME_NATIVE_FAST TBD

FWPS_FIELDS_INBOUND_RESERVED2 값이 내부용으로 예약된 열거형입니다.
FWPS_FIELDS_INBOUND_TRANSPORT_FAST TBD
FWPS_FIELDS_INBOUND_TRANSPORT_V4 FWPS_LAYER_INBOUND_TRANSPORT_V4 및 FWPS_LAYER_INBOUND_TRANSPORT_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_INBOUND_TRANSPORT_V6 FWPS_LAYER_INBOUND_TRANSPORT_V6 및 FWPS_LAYER_INBOUND_TRANSPORT_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_INGRESS_VSWITCH_ETHERNET FWPS_LAYER_INGRESS_VSWITCH_ETHERNET 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_INGRESS_VSWITCH_TRANSPORT_V4 FWPS_LAYER_INGRESS_VSWITCH_TRANSPORT_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_INGRESS_VSWITCH_TRANSPORT_V6 FWPS_LAYER_INGRESS_VSWITCH_TRANSPORT_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_IPFORWARD_V4 FWPS_LAYER_IPFORWARD_V4 및 FWPS_LAYER_IPFORWARD_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_IPFORWARD_V6 FWPS_LAYER_IPFORWARD_V6 및 FWPS_LAYER_IPFORWARD_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_IPSEC_KM_DEMUX_V4 FWPS_LAYER_IPSEC_KM_DEMUX_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_IPSEC_KM_DEMUX_V6

FWPS_LAYER_IPSEC_KM_DEMUX_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_IPSEC_V4 FWPS_LAYER_IPSEC_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_IPSEC_V6 FWPS_LAYER_IPSEC_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_KM_AUTHORIZATION FWPS_LAYER_KM_AUTHORIZATION 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_NAME_RESOLUTION_CACHE_V4 FWPS_LAYER_NAME_RESOLUTION_CACHE_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_NAME_RESOLUTION_CACHE_V6 FWPS_LAYER_NAME_RESOLUTION_CACHE_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_OUTBOUND_ICMP_ERROR_V4 FWPS_LAYER_OUTBOUND_ICMP_ERROR_V4 및 FWPS_LAYER_OUTBOUND_ICMP_ERROR_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_OUTBOUND_ICMP_ERROR_V6 FWPS_LAYER_OUTBOUND_ICMP_ERROR_V6 및 FWPS_LAYER_OUTBOUND_ICMP_ERROR_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_OUTBOUND_IPPACKET_V4 FWPS_LAYER_OUTBOUND_IPPACKET_V4 및 FWPS_LAYER_OUTBOUND_IPPACKET_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_OUTBOUND_IPPACKET_V6 FWPS_LAYER_OUTBOUND_IPPACKET_V6 및 FWPS_LAYER_OUTBOUND_IPPACKET_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_OUTBOUND_MAC_FRAME_ETHERNET

FWPS_LAYER_OUTBOUND_MAC_FRAME_ETHERNET 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_OUTBOUND_MAC_FRAME_NATIVE FWPS_LAYER_OUTBOUND_MAC_FRAME_NATIVE 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_OUTBOUND_MAC_FRAME_NATIVE_FAST TBD
FWPS_FIELDS_OUTBOUND_NETWORK_POLICY_V4 FWPS_LAYER_OUTBOUND_NETWORK_CONNECTION_POLICY_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_OUTBOUND_NETWORK_POLICY_V6 FWPS_LAYER_OUTBOUND_NETWORK_CONNECTION_POLICY_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_OUTBOUND_TRANSPORT_FAST TBD
FWPS_FIELDS_OUTBOUND_TRANSPORT_V4 FWPS_LAYER_OUTBOUND_TRANSPORT_V4 및 FWPS_LAYER_OUTBOUND_TRANSPORT_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_OUTBOUND_TRANSPORT_V6 FWPS_LAYER_OUTBOUND_TRANSPORT_V6 및 FWPS_LAYER_OUTBOUND_TRANSPORT_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_RPC_EP_ADD FWPS_LAYER_RPC_EP_ADD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_RPC_EPMAP FWPS_LAYER_RPC_EPMAP 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_RPC_PROXY_CONN FWPS_LAYER_RPC_PROXY_CONN 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.

FWPS_FIELDS_RPC_PROXY_IF_IF FWPS_LAYER_RPC_PROXY_IF 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_RPC_UM FWPS_LAYER_RPC_UM 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_STREAM_PACKET_V4 FWPS_LAYER_STREAM_PACKET_V4 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_STREAM_PACKET_V6 FWPS_LAYER_STREAM_PACKET_V6 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_STREAM_V4 FWPS_LAYER_STREAM_V4 및 FWPS_LAYER_STREAM_V4_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
FWPS_FIELDS_STREAM_V6 FWPS_LAYER_STREAM_V6 및 FWPS_LAYER_STREAM_V6_DISCARD 런타임 필터링 계층의 데이터 필드 식별자를 지정합니다.
INET_DISCARD_REASON 전송 계층 중 하나에서 데이터를 삭제할 수 있는 이유를 정의합니다.
IP_DISCARD_REASON 네트워크 계층 중 하나에서 데이터를 삭제할 수 있는 이유를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

fwptypes.h 헤더

아티클2023. 08. 23.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 드라이버, Windows Vista 이상
- Windows 필터링 플랫폼

fwptypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

FWP_BYTE_ARRAY16 정확히 16바이트 배열을 저장합니다.
FWP_BYTE_ARRAY6 정확히 6바이트 배열을 저장합니다. 예약되어 있습니다.
FWP_BYTE_BLOB 가변적인 바이트 수를 포함하는 배열을 저장합니다.
FWP_CONDITION_VALUE0 일치하는 필터를 테스트할 때 필터 조건에서 사용되는 값을 포함합니다.
FWP_RANGE0 값의 범위를 지정합니다.
FWP_TOKEN_INFORMATION FWP_TOKEN_INFORMATION 구조는 사용자 모드 분류에 사용되는 보안 식별자 집합을 정의합니다.
FWP_V4_ADDR_AND_MASK 호스트 순서로 IPv4 주소 및 마스크를 지정합니다.
FWP_V6_ADDR_AND_MASK IPv6 주소 및 마스크를 지정합니다.

FWP_VALUE0 다양한 데이터 형식 중 하나일 수 있는 데이터 값을 정의합니다.
FWPM_DISPLAY_DATA0 개체에 대한 선택적 이름 및 선택적 설명을 저장합니다.
IPSEC_VIRTUAL_IF_TUNNEL_INFO0 가상 인터페이스 터널링과 관련된 정보를 저장하는 데 사용됩니다.

열거형

FWP_AF FWP_AF 열거형 형식입니다.
FWP_CLASSIFY_OPTION_TYPE FWP_CLASSIFY_OPTION_TYPE 열거형 형식은 런타임 분류 중에 설명선 및 shim에서 사용됩니다. FWP_CLASSIFY_OPTION_TYPE 유니캐스트, 멀티캐스트 및 느슨한 원본 매핑 상태에 대한 시간 제한 옵션을 지정하고 아웃바운드 멀티캐스트 및 브로드캐스트 트래픽에 대한 상태 생성 차단 또는 권한을 사용하도록 설정합니다.
FWP_DATA_TYPE FWP_VALUE0 또는 FWP_CONDITION_VALUE0structure 저장할 수 있는 데이터 형식입니다.
FWP_DIRECTION 네트워크 트래픽의 방향입니다.
FWP_ETHER_ENCAP_METHOD 이더넷 II 및 SNAP 트래픽을 캡슐화하는 방법을 지정합니다. 예약되어 있습니다.
FWP_FILTER_ENUM_TYPE 필터 열거형 조건을 해석하는 방법을 지정합니다.
FWP_IP_VERSION IP 버전을 지정합니다. (FWP_IP_VERSION)

FWP_MATCH_TYPE 필터 조건에서 허용되는 다양한 일치 형식입니다.
FWP_VSWITCH_NETWORK_TYPE vSwitch의 네트워크 유형을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

fxsutility.h 헤더

아티클2023. 08. 23.

이 헤더는 팩스 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 팩스 서비스

fxsutility.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CanSendToFaxRecipient Windows Vista 함수 SendToFaxRecipient를 호출하는 메뉴 항목 또는 기타 UI를 사용할 수 있도록 할지 여부를 결정하기 위해 애플리케이션에서 호출합니다.
SendToFaxRecipient 애플리케이션에서 파일을 팩스로 보내도록 호출합니다.

열거형

SendToMode 애플리케이션 내에서 파일을 팩스로 전송하는 방법을 정의합니다. Windows Vista에는 하나의 가능한 값만 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

gamingdeviceinformation.h 헤더

아티클2023. 08. 23.

이 헤더는 게임 장치 정보에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 게임 디바이스 정보

gamingdeviceinformation.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

GetGamingDeviceModelInformation 게임이 실행 중인 디바이스에 대한 정보를 가져옵니다.

구조체

GAMING_DEVICE_MODEL_INFORMATION 게임이 실행 중인 디바이스에 대한 정보를 포함합니다.

열거형

GAMING_DEVICE_DEVICE_ID 게임이 실행 중인 디바이스의 유형을 나타냅니다.
GAMING_DEVICE_VENDOR_ID 게임이 실행 중인 콘솔의 공급업체를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

gamingtcui.h 헤더

아티클2023. 04. 27.

이 헤더는 타이틀 호출 가능 UI에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 타이틀 호출 가능 UI

gamingtcui.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CheckGamingPrivilegeSilently 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (CheckGamingPrivilegeSilently)
CheckGamingPrivilegeWithUI 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (CheckGamingPrivilegeWithUI)
ProcessPendingGameUI 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (ProcessPendingGameUI)
ShowChangeFriendRelationshipUI 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (ShowChangeFriendRelationshipUI)
ShowChangeFriendRelationshipUIForUser 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (ShowChangeFriendRelationshipUIForUser)
ShowGameInviteUI 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (ShowGameInviteUI)
ShowPlayerPickerUI 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (ShowPlayerPickerUI)
ShowProfileCardUI 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (ShowProfileCardUI)

ShowProfileCardUIForUser 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (ShowProfileCardUIForUser)
ShowTitleAchievementsUI 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (ShowTitleAchievementsUI)
TryCancelPendingGameUI 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (TryCancelPendingGameUI)

콜백 함수

GameUICompletionRoutine 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (GameUICompletionRoutine)
PlayerPickerUICompletionRoutine 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (PlayerPickerUICompletionRoutine)

열거형

KnownGamingPrivileges 사용하지 마십시오. 이 API는 Xbox 개발자만 지원됩니다. (KnownGamingPrivileges)

gb18030.h 헤더

아티클2023. 08. 23.

이 헤더는 Windows 애플리케이션에 대한 Internationalization에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 애플리케이션에 대한 국제화

gb18030.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

NlsDllCodePageTranslation 플래그 설정에 따라 코드 페이지 정보를 얻거나 변환을 수행하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

gdiplusbase.h 헤더

아티클2023. 08. 23.

이 헤더는 GDI+에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- GDI+

gdiplusbase.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

GdiplusBase GdiplusBase 클래스는 GDI+ 개체에 대한 메모리 할당 및 할당 취소를 제공합니다. GdiplusBase는 다른 모든 GDI+ 클래스의 기본 클래스 역할을 하므로 GdiplusBase의 instance 만들 필요가 없습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

gdiplusbrush.h 헤더

아티클2023. 08. 23.

이 헤더는 GDI+에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- GDI+

gdiplusbrush.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

Brush Brush 클래스는 Brush 개체를 정의하는 추상 기본 클래스입니다. Brush 개체는 사각형, 타원, 원형, 다각형 및 경로와 같은 그래픽 셰이프의 내부를 그리는 데 사용됩니다.
HatchBrush 이 HatchBrush 클래스는 해치 스타일, 전경색 및 배경색을 사용하여 사각형 브러시를 정의합니다.
LinearGradientBrush LinearGradientBrush 클래스는 선형 그라데이션 브러시의 시작 경계선에서 선형 그라데이션 브러시의 끝 경계선까지 색이 균등하게 변하는 색 그라데이션을 그리는 브러시를 정의합니다.
SolidBrush SolidBrush 클래스는 단색 Brush 개체를 정의합니다. Brush 개체는 페인트 브러시가 도형 내부를 칠할 수 있는 방식과 유사한 도형을 채우는 데 사용됩니다. 이 클래스는 Brush 추상 기본 클래스에서 상속됩니다.
TextureBrush TextureBrush 클래스는 채우기에 사용되는 Image 개체를 포함하는 Brush 개체를 정의합니다. Fill 이미지는 Brush 개체에 포함된 로컬 Matrix 개체를 사용하여 변환할 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

gdipluscolor.h 헤더

아티클2023. 08. 24.

이 헤더는 GDI+에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- GDI+

gdipluscolor.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

색상 Color 개체는 색을 나타내는 32비트 값을 저장합니다.

열거형

ColorChannelFlags ColorChannelFlags 열거형은 CMYK(cyan, magenta, yellow, black) 색 공간에서 개별 채널을 지정합니다. 이 열거형은 ImageAttributes::SetOutputChannel 메서드에서 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

gdipluscolormatrix.h 헤더

이 헤더는 GDI+에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- GDI+

gdipluscolormatrix.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조

테이블 확장

ColorMap ColorMap 구조체에는 두 개의 Color 개체가 포함됩니다. ImageAttributes 클래스의 여러 메서드는 ColorMap 구조의 배열인 색 다시 매핑 테이블을 사용하여 이미지 색을 조정합니다.
ColorMatrix ColorMatrix 구조체에는 실제 숫자의 5×5 행렬이 포함됩니다. ImageAttributes 클래스의 여러 메서드는 색 행렬을 사용하여 이미지 색을 조정합니다.

열거 목록

테이블 확장

ColorAdjustType ColorAdjustType 열거형은 색 조정 정보를 사용하는 GDI+ 개체를 지정합니다.
ColorMatrixFlags ColorMatrixFlags 열거형은 ImageAttributes 개체의 색 및 회색조 조정 설정의 영향을 받는 이미지 및 색 유형을 지정합니다.
HistogramFormat HistogramFormat 열거형은 비트맵의 색 채널을 나타내는 히스토그램의 수와 형식을 지정합니다. 이 열거형은 Bitmap::GetHistogram 메서드와 함께 사용됩니다.

Last updated on 2025. 10. 08.

gdipluseffects.h 헤더

아티클2023. 08. 24.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Directshow
- GDI+

gdipluseffects.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

흐리게 Blur 클래스를 사용하면 가우스 흐림 효과를 비트맵에 적용하고 흐림 효과의 특성을 지정할 수 있습니다.
BrightnessContrast BrightnessContrast 클래스를 사용하면 비트맵의 밝기와 대비를 변경할 수 있습니다.
ColorBalance ColorBalance 클래스를 사용하면 비트맵의 색 균형(빨강, 녹색 및 파랑의 상대적 크기)을 변경할 수 있습니다.
ColorCurve ColorCurve 클래스는 _exposure, 밀도, 대비, 강조 표시, 그림자, 중간, 흰색 채도 및 검은색 채도의 8가지 개별 조정을 포함합니다.
ColorLUT ColorLUTParams 구조체에는 각각 특정 색 채널(_alpha, 빨간색, 녹색 또는 파랑)에 대한 조회 테이블인 4개의 멤버가 있습니다.
ColorMatrixEffect ColorMatrixEffect 클래스를 사용하면 비트맵에 아핀 변환을 적용할 수 있습니다.
효과 Effect 클래스는 비트맵에 효과 및 조정을 적용하는 데 사용할 수 있는 11개의 클래스에 대한 기본 클래스로 사용됩니다. 다음 클래스는 효과에서 내림차순입니다.

HueSaturationLightness HueSaturationLightness 클래스를 사용하면 비트맵의 색조, 채도 및 가벼움을 변경할 수 있습니다.
수준 Levels 클래스는 세 가지 비트맵 조정(_highlight, 중간 및 그림자)을 포함합니다.
RedEyeCorrection RedEyeCorrection 클래스를 사용하면 플래시 사진에서 발생하는 빨간색 눈을 수정할 수 있습니다.
선명하게 Sharpen 클래스를 사용하면 비트맵의 선명도를 조정할 수 있습니다.
색조 Tint 클래스를 사용하면 비트맵에 색조를 적용할 수 있습니다.

구조체

BlurParams BlurParams 구조체에는 가우스 흐림의 특성을 지정하는 멤버가 포함됩니다.
BrightnessContrastParams BrightnessContrastParams 구조체에는 밝기 또는 대비 조정의 특성을 지정하는 멤버가 포함됩니다.
ColorBalanceParams ColorBalanceParams 구조체에는 색 균형 조정의 특성을 지정하는 멤버가 포함됩니다.
ColorCurveParams ColorCurveParams 구조체에는 비트맵의 색에 대한 조정을 지정하는 멤버가 포함됩니다.
ColorLUTParams ColorLUTParams 구조체에는 비트맵에 대한 색 조정을 지정하는 멤버(색 조회 테이블)가 포함됩니다.
HueSaturationLightnessParams

HueSaturationLightnessParams 구조체에는 비트맵에 대한 색조, 채도 및 가벼움 조정을 지정하는 멤버가 포함되어 있습니다.
LevelParams LevelsParams 구조체에는 비트맵의 밝게, 중간 또는 어두운 영역에 대한 조정을 지정하는 멤버가 포함됩니다.
RedEyeCorrectionParams RedEyeCorrectionParams 구조체에는 적목 보정이 적용되는 비트맵 영역을 지정하는 멤버가 포함됩니다.
SharpenParams SharpenParams 구조체에는 비트맵에 대한 선명도 조정의 특성을 지정하는 멤버가 포함됩니다.
TintParams TintParams 구조체에는 비트맵에 대한 색조 조정의 특성을 지정하는 멤버가 포함됩니다.

열거형

CurveAdjustments ColorCurve 클래스는 CurveAdjustments 열거형에 나열된 8개의 비트맵 조정을 포함합니다.
CurveChannel CurveChannel 열거형은 ColorCurve 비트맵 조정의 영향을 받는 색상 채널을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

gdiplusenums.h 헤더

아티클2023. 08. 24.

이 헤더는 GDI+에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- GDI+

gdiplusenums.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

ObjectTypeIsValid ObjectTypeIsValid 함수는 ObjectType 열거형의 요소가 유효한 개체 형식을 나타내는지 여부를 결정합니다.

열거형

BrushType BrushType 열거형은 브러시 유형을 나타냅니다. 브러시에는 5가지 유형이 있습니다.
CombineMode CombineMode 열거형은 새 지역을 기존 지역과 결합하는 방법을 지정합니다.
CompositingMode CompositingMode 열거형은 렌더링된 색을 배경색과 결합하는 방법을 지정합니다. 이 열거형은 Graphics 클래스의 Graphics::GetCompositingMode 및 Graphics::SetCompositingMode 메서드에서 사용됩니다.
CompositingQuality CompositingQuality 열거형은 색이 배경색과 혼합될 때 감마 보정이 적용되는지 여부를 지정합니다.
CoordinateSpace CoordinateSpace 열거형은 좌표 공간을 지정합니다.

DashCap DashCap 열거형은 파선에서 각 대시의 양쪽 끝에 사용할 그래픽 셰이프의 형식을 지정합니다.
DashStyle DashStyle 열거형은 Windows GDI+ 펜으로 그린 선의 선 스타일을 지정합니다. 선은 미리 정의된 여러 스타일 또는 사용자 지정 스타일 중 하나를 사용하여 그릴 수 있습니다.
DriverStringOptions DriverStringOptions 열거형은 드라이버 문자열에 대한 렌더링의 간격, 방향 및 품질을 지정합니다.
EmfPlusRecordType EmfPlusRecordType 열거형은 WMF(Windows 메타파일 형식), EMF(Enhanced Metafile) 및 EMF+ 파일에 사용되는 메타파일 레코드 형식을 식별합니다. EmfPlusRecordType 열거형의 요소는 세 개의 그룹으로 제공됩니다.
EmfToWmfBitsFlags EMF(Enhanced Metafile) 메타파일을 WMF(Windows 메타파일 형식) 메타파일로 변환하는 Metafile::EmfToWmfBits 메서드에 대한 옵션을 지정합니다.
EmfType EmfType 열거형은 EMF(Enhanced Metafile) 파일에 배치되는 레코드의 특성을 지정합니다. 이 열거형은 Metafile 클래스의 여러 생성자에서 사용됩니다.
EncoderParameterValueType EncoderParameterValueType 열거형은 이미지 코덱(인코더/디코더) 매개 변수에 대한 데이터 형식을 지정합니다. 이 열거형의 요소는 EncoderParameter 개체의 Type 데이터 멤버에 할당됩니다.
EncoderValue EncoderValue 열거형은 이미지 인코더에 인수로 전달할 수 있는 값을 지정합니다. 이미지 인코더에 대한 자세한 내용은 이미지 인코더 및 디코더 사용을 참조하세요.
FillMode FillMode 열거형은 경로 또는 곡선이 자체적으로 교차할 때 형성되는 영역을 채우는 방법을 지정합니다.
FlushIntention FlushIntention 열거형은 그래픽 작업의 큐를 플러시할 시기를 지정합니다.

FontStyle FontStyle 열거형은 글꼴 서체의 스타일을 지정합니다. 스타일을 결합할 수 있습니다.
HatchStyle HatchStyle 열거형은 HatchBrush 형식의 브러시에서 사용하는 해치 패턴을 지정합니다. 빗살 무늬 배경색 및 배경 위에 그리는 선의 구성 됩니다.
HotkeyPrefix HotkeyPrefix 열거형은 핫 키를 표시하는 방법을 지정합니다. 세 가지 옵션이 있습니다. 아무것도 _do 핫 키 밑줄을 표시하고 핫 키 밑줄을 숨깁니다.
ImageType ImageType 열거형은 이미지가 비트맵인지 메타파일인지를 나타냅니다. Image::GetType 메서드는 이 열거형의 요소를 반환합니다.
보간Mode InterpolationMode 열거형은 이미지 크기 조정 또는 회전 시 사용되는 알고리즘을 지정합니다. 이 열거형은 Graphics 클래스의 Graphics::GetInterpolationMode 및 Graphics::SetInterpolationMode 메서드에서 사용됩니다.
LinearGradientMode LinearGradientMode 열거형은 선형 그라데이션 브러시의 색 변경이 발생하는 방향을 지정합니다.
LineCap LineCap 열거형은 Windows GDI+ 펜으로 그린 선의 끝에 사용할 그래픽 셰이프의 유형을 지정합니다.
LineJoin LineJoin 열거형은 동일한 펜으로 그려지고 끝이 만나는 두 줄을 조인하는 방법을 지정합니다. 두 줄 끝의 교집합에서 선 조인은 조인을 더 연속적으로 보이게 합니다.
MatrixOrder MatrixOrder 열거형은 새 행렬에 기존 행렬을 곱할 때 곱하기 순서를 지정합니다.
MetafileFrameUnit MetafileFrameUnit 열거형은 메타파일 프레임 사각형에 대한 측정 단위를 지정합니다.
MetafileType

MetafileType 열거형은 메타파일의 형식을 지정합니다. MetafileHeader::GetType 메서드는 이 열거형의 요소를 반환합니다.
ObjectType ObjectType 열거형은 EMF+ 레코드의 개체 형식 값을 나타냅니다.
PathPointType PathPointType 열거형은 경로의 데이터 요소에 대한 점 형식 및 플래그를 나타냅니다.
PenAlignment PenAlignment 열거형은 그리는 스트로크를 기준으로 펜의 맞춤을 지정합니다.
PenType PenType 열거형은 펜이 그리는 패턴, 질감 또는 그라데이션의 형식을 나타냅니다.
PixelOffsetMode PixelOffsetMode 열거형은 Graphics 개체의 픽셀 오프셋 모드를 지정합니다. 이 열거형은 Graphics 클래스의 Graphics::GetPixelOffsetMode 및 Graphics::SetPixelOffsetMode 메서드에서 사용됩니다.
SmoothingMode SmoothingMode 열거형은 선과 곡선에 적용되는 스무딩(앤티앨리어싱)의 유형을 지정합니다. 이 열거형은 Graphics 클래스의 Graphics::GetSmoothingMode 및 Graphics::SetSmoothingMode 메서드에서 사용됩니다.
StringAlignment StringAlignment 열거형은 경계 사각형을 참조하여 문자열을 정렬하는 방법을 지정합니다. 경계 사각형은 텍스트가 표시되는 영역을 정의하는 데 사용됩니다.
StringDigitSubstitute StringDigitSubstitute 열거형은 사용자의 로캘 또는 언어에 따라 문자열의 숫자를 대체하는 방법을 지정합니다.
StringFormatFlags StringFormatFlags 열거형은 텍스트 레이아웃 정보(예: 방향 및 클리핑) 및 표시 조작(예: 줄임표 삽입, 숫자 대체 및 글꼴에서 지원되지 않는 문자 표현)을 지정합니다.
StringTrimming StringTrimming 열거형은 문자열이 레이아웃 사각형에 맞도록 문자열에서 문자를 자르는 방법을 지정합니다. 레이아웃 사각형은 표시 문자열의 위치를 지정하고 크기를 조정하는 데 사용됩니다.

TextRenderingHint TextRenderingHint 열거형은 텍스트를 렌더링하는 데 사용되는 프로세스를 지정합니다. 이 프로세스는 텍스트의 품질에 영향을 줍니다.
단위 단위 열거형은 지정된 데이터 형식에 대한 측정 단위를 지정합니다.
WarpMode WarpMode 열거형은 이미지를 변환하는 데 사용할 수 있는 워프 모드를 지정합니다.
Wrapmode WrapMode 열거형은 이미지의 반복 복사본을 사용하여 영역을 타일로 지정하는 방법을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

gdiplusgraphics.h 헤더

아티클2023. 08. 24.

이 헤더는 GDI+에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- GDI+

gdiplusgraphics.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

그래픽	Graphics 클래스는 선, 곡선, 그림, 이미지 및 텍스트를 그리는 메서드를 제공합니다. Graphics 개체는 표시 디바이스의 특성과 그릴 항목의 특성을 저장합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

gdiplusheaders.h 헤더

아티클2023. 08. 24.

이 헤더는 GDI+에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- GDI+

gdiplusheaders.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

Bitmap Bitmap 클래스는 Image 클래스에서 상속됩니다.
CachedBitmap CachedBitmap 개체는 특정 디바이스에 표시하기 위해 최적화된 형식으로 비트맵을 저장합니다. 캐시된 비트맵을 표시하려면 Graphics::D rawCachedBitmap 메서드를 호출합니다.
CustomLineCap CustomLineCap 클래스는 사용자 지정 줄 한도를 캡슐화합니다.
글꼴 Font 클래스는 특정 글꼴의 패밀리, 높이, 크기 및 스타일(또는 스타일 조합)과 같은 특성을 캡슐화합니다. Font 개체는 문자열을 그릴 때 사용됩니다.
FontCollection FontCollection 클래스는 추상 기본 클래스입니다. 글꼴 컬렉션에서 글꼴 패밀리를 열거하는 메서드가 포함되어 있습니다. 이 클래스에서 빌드된 개체에는 InstalledFontCollection 및 PrivateFontCollection 클래스가 포함됩니다.
FontFamily 이 FontFamily 클래스는 글꼴 패밀리를 구성하는 글꼴 집합을 캡슐화합니다. 글꼴 패밀리는 서체가 같지만 스타일이 다른 글꼴 그룹입니다.
이미지 Image 클래스는 래스터 이미지(비트맵) 및 벡터 이미지(메타파일)를 로드하고 저장하는 메서드를 제공합니다.

InstalledFontCollection InstalledFontCollection 클래스는 시스템에 설치된 글꼴을 나타내는 클래스를 정의합니다.
Metafile Metafile 클래스는 그래픽 메타파일을 정의합니다. 메타파일은 그래픽 API 호출 시퀀스를 설명하는 레코드를 포함합니다. 메타파일을 기록(생성)하고 재생(표시)할 수 있습니다.
PrivateFontCollection PrivateFontCollection은 글꼴에 대한 컬렉션 클래스입니다. 이 클래스는 애플리케이션에 대한 글꼴 컬렉션을 유지합니다. 컬렉션의 글꼴에는 설치된 글꼴과 시스템에 설치되지 않은 글꼴이 포함될 수 있습니다.
지역 Region 클래스는 디스플레이 화면의 영역을 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

gdiplusimageattributes.h 헤더

아티클2023. 08. 24.

이 헤더는 GDI+에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- GDI+

gdiplusimageattributes.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

ImageAttributes ImageAttributes 개체에는 렌더링 중에 비트맵 및 메타파일 색을 조작하는 방법에 대한 정보가 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

gdiplusimagecodec.h 헤더

아티클2023. 08. 24.

이 헤더는 GDI+에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- GDI+

gdiplusimagecodec.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

GetImageDecoders GetImageDecoders 함수는 사용 가능한 이미지 디코더에 대한 정보를 포함하는 ImageCodecInfo 개체의 배열을 가져옵니다.
GetImageDecodersSize GetImageDecodersSize 함수는 사용 가능한 이미지 디코더 수와 GetImageDecoders 함수에서 반환되는 ImageCodecInfo 개체 배열의 총 크기를 가져옵니다.
GetImageEncoders GetImageEncoders 함수는 사용 가능한 이미지 인코더에 대한 정보를 포함하는 ImageCodecInfo 개체의 배열을 가져옵니다.
GetImageEncodersSize GetImageEncodersSize 함수는 사용 가능한 이미지 인코더 수와 GetImageEncoders 함수에서 반환되는 ImageCodecInfo 개체 배열의 총 크기를 가져옵니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

gdiplusimaging.h 헤더

아티클2023. 08. 24.

이 헤더는 GDI+에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- GDI+

gdiplusimaging.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

Bitmapdata BitmapData 클래스는 Bitmap 클래스의 Bitmap::LockBits 및 Bitmap::UnlockBits 메서드에서 사용됩니다. BitmapData 개체는 비트맵의 특성을 저장합니다.
EncoderParameter EncoderParameter 개체는 이미지 인코더에 전달할 수 있는 매개 변수를 보유합니다. EncoderParameter 개체를 사용하여 특정 이미지 인코더의 특정 매개 변수에서 지원하는 가능한 값 목록을 받을 수도 있습니다.
EncoderParameters EncoderParameters 개체는 배열의 EncoderParameter 개체 수를 지정하는 데이터 멤버와 함께 EncoderParameter 개체의 배열입니다.
ImageCodecInfo ImageCodecInfo 개체는 이미지 코덱(인코더/디코더)에 대한 정보를 저장합니다.
ImageItemData ImageItemData 클래스는 사용자 지정 이미지 메타데이터를 저장하고 검색하는 데 사용됩니다. Windows GDI+는 JPEG, PNG 및 GIF 이미지 파일에 대한 사용자 지정 메타데이터를 지원합니다.
PropertyItem PropertyItem 클래스는 Image 및 Bitmap 클래스의 도우미 클래스입니다. PropertyItem 개체는 이미지 메타데이터의 한 부분을 보유합니다.

열거형

ImageCodecFlags ImageCodecFlags 열거형은 이미지 코덱의 특성을 나타냅니다.
ImageFlags ImageFlags 열거형은 Image 개체에 포함된 픽셀 데이터의 특성을 지정합니다. Image::GetFlags 메서드는 이 열거형의 요소를 반환합니다.
ImageLockMode ImageLockMode 열거형은 Bitmap::LockBits 메서드의 flags 매개 변수에 전달되는 플래그를 지정합니다. Bitmap::LockBits 메서드는 픽셀 데이터를 읽거나 쓸 수 있도록 이미지의 일부를 잠급니다.
ItemDataPosition ItemDataPosition 열거형은 이미지 파일에서 사용자 지정 메타데이터의 위치를 지정하는 데 사용됩니다.
RotateFlipType RotateFlipType 열거형은 이미지 회전 방향과 이미지를 대칭 이동하는 데 사용되는 축을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

gdiplusinit.h 헤더

이 헤더는 GDI+에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- GDI+

gdiplusinit.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

GdiplusShutdown GdiplusShutdown 함수는 Windows GDI+에서 사용하는 리소스를 정리합니다. GdiplusStartup에 대한 각 호출은 GdiplusShutdown 호출과 쌍을 이루어야 합니다.
GdiplusStartup GdiplusStartup 함수는 Windows GDI+를 초기화합니다. 다른 GDI+ 호출을 하기 전에 GdiplusStartup을 호출하고 GDI+를 사용한 후 GdiplusShutdown을 호출합니다.

콜백 함수

테이블 확장

DebugEventProc * *DebugEventProc*는 애플리케이션에서 구현하고 **GdiplusStartupInput**의 생성자에 전달하는 콜백 함수의 서명입니다.
NotificationHookProc * *NotificationHookProc*는 **GdiplusStartup** 이 **GdiplusStartupOutput** 구조에서 반환하는 알림 후크 함수의 서명입니다.
NotificationUnhookProc * *NotificationUnhookProc*는 **GdiplusStartup** 이 **GdiplusStartupOutput** 구조에서 반환하는 알림 해제 함수의 서명입니다.

구조체

테이블 확장

GdiplusStartupInput **GdiplusStartupInput** 구조체는 GdiplusStartup 함수에 필요한 인수 블록을 보유합니다.
GdiplusStartupInputEx **GdiplusStartupInputEx** 구조체에는 GdiplusStartup 함수에 필요한 인수 블록이 있습니다.
GdiplusStartupOutput GdiplusStartup 함수는 GdiplusStartupOutput 구조를 사용하여 후크 함수에 대한 포인터와 unhook 함수에 대한 포인터를 반환합니다(출력 매개 변수).

열거형

테이블 확장

DebugEventLevel 디버그 이벤트 수준을 지정하는 상수를 정의합니다.
GdiplusStartupParams 시작에 대한 매개 변수를 지정하는 상수를 정의합니다.

Last updated on 2025. 10. 01.

gdipluslinecaps.h 헤더

아티클2023. 08. 24.

이 헤더는 GDI+에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- GDI+

gdipluslinecaps.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

AdjustableArrowCap AdjustableArrowCap 클래스는 CustomLineCap의 하위 클래스입니다. 이 클래스는 화살표처럼 보이는 줄 뚜껑을 작성합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

gdiplusmatrix.h 헤더

이 헤더는 GDI+에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- GDI+

gdiplusmatrix.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

수업

테이블 확장

행렬 Matrix 개체는 3 × 3 행렬을 나타내며, 이 행렬은 차례로 아핀 변환을 나타냅니다.	

Last updated on 2025. 10. 08.

gdiplusmetaheader.h 헤더

아티클2023. 08. 24.

이 헤더는 GDI+에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- GDI+

gdiplusmetaheader.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

MetafileHeader MetafileHeader 개체는 연결된 메타파일의 속성을 저장합니다.

구조체

ENHMETAHEADER3 ENHMETAHEADER3 구조체에는 메타파일 이미지의 차원, 메타파일의 레코드 수 및 메타파일을 만든 디바이스의 해상도를 포함하여 향상된 메타파일 데이터가 포함됩니다.
PWMFRect16 PWMFRect16 구조체는 배치 가능한 메타파일을 경계하는 사각형을 정의합니다. 사각형은 메타파일의 크기와 위치를 정의합니다.
WmfPlaceableFileHeader WmfPlaceableFileHeader 구조체는 배치 가능한 메타파일 헤더의 필드를 정의합니다. 표시 장치에서 메타파일을 매핑하고 스케일링하는 방법을 지정하는 방법으로 배치 가능한 메타파일을 만들었습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

gdipluspath.h 헤더

아티클2023. 08. 24.

이 헤더는 GDI+에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- GDI+

gdipluspath.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

Graphicspath GraphicsPath 개체는 선, 곡선 및 셰이프 시퀀스를 저장합니다.
GraphicsPathIterator 이 GraphicsPathIterator 클래스는 GraphicsPath 개체에 저장된 경로의 선택한 하위 집합을 격리하는 메서드를 제공합니다.
PathGradientBrush PathGradientBrush 개체는 경로의 내부를 점차 변화하는 색으로 채우는 데 사용할 수 있는 색 그라데이션의 특성을 저장합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

gdipluspen.h 헤더

아티클2023. 08. 24.

이 헤더는 GDI+에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- GDI+

gdipluspen.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

펜 펜 개체는 선과 곡선을 그리는 데 사용되는 Windows GDI+ 개체입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

gdipluspixelformats.h 헤더

아티클2023. 08. 24.

이 헤더는 GDI+에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- GDI+

gdipluspixelformats.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

GetPixelFormatSize GetPixelFormatSize 메서드는 지정된 픽셀 형식으로 사용되는 픽셀당 비트 수를 반환합니다.
IsAlphaPixelFormat IsAlphaPixelFormat 메서드는 지정된 픽셀 형식에 알파 구성 요소가 있는지 여부를 결정합니다.
IsCanonicalPixelFormat IsCanonicalPixelFormat 메서드는 지정된 픽셀 형식이 정식 형식(_PixelFormat32bppARGB 또는 PixelFormat64bppARGB 중 하나인지 여부를 결정합니다.
IsExtendedPixelFormat IsExtendedPixelFormat 메서드는 지정된 픽셀 형식이 색 채널당 16비트를 사용하는지 여부를 결정합니다.
IsIndexedPixelFormat IsIndexedPixelFormat 메서드는 지정된 픽셀 형식이 인덱싱된 형식인지 여부를 결정합니다.

구조체

ColorPalette ColorPalette 구조체는 색상표를 구성하는 색 배열을 정의합니다. 해당 색들은 32비트 ARGB 색입니다.

열거형

DitherType DitherType 열거형은 비트맵을 변환할 때 디더링에 사용할 수 있는 알고리즘을 식별합니다.
PaletteFlags PaletteFlags 열거형은 색상표에 있는 색 데이터의 특성을 나타냅니다.
PaletteType PaletteType 열거형은 Bitmap 클래스의 Bitmap::InitializePalette 및 Bitmap::ConvertFormat 메서드에서 사용됩니다. 열거형의 멤버는 여러 표준 색상표 형식을 식별합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

gdiplusstringformat.h 헤더

아티클2023. 08. 24.

이 헤더는 GDI+에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- GDI+

gdiplusstringformat.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

StringFormat StringFormat 클래스는 텍스트 레이아웃 정보(예: 맞춤, 방향, 탭 정지 및 클리핑)와 표시 조작(예: 트리밍, 요청된 글꼴에서 지원되지 않는 문자의 글꼴 대체 및 서유럽 숫자를 사용하지 않는 언어에 대한 숫자 대체)을 캡슐화합니다. StringFormat 개체를 DrawString Methods 메서드에 전달하여 문자열 형식을 지정할 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

gdiplustypes.h 헤더

아티클2023. 08. 24.

이 헤더는 GDI+에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- GDI+

gdiplustypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

CharacterRange CharacterRange 개체는 문자열 내의 문자 위치 범위를 지정합니다.
PathData PathData 클래스는 GraphicsPath 및 GraphicsPathIterator 클래스에 대한 도우미 클래스입니다.
Point Point 클래스는 2차원 좌표계의 점을 캡슐화합니다.
Pointf PointF 클래스는 2차원 좌표계의 점을 캡슐화합니다.
Rect Rect 개체는 사각형의 왼쪽 위 모서리, 너비 및 높이를 저장합니다.
RectF RectF 개체는 사각형의 왼쪽 위 모서리, 너비 및 높이를 저장합니다.
크기 Size 클래스는 2차원 좌표계의 Width 및 Height 차원을 캡슐화합니다.
Sizef SizeF 클래스는 2차원 좌표계의 Width 및 Height 차원을 캡슐화합니다. SizeF 클래스는 부동 소수점 좌표를 사용합니다.

콜백 함수

EnumerateMetafileProc *EnumerateMetafileProc*는 **Graphics::EnumerateMetafile** 메서드(및 오버로드)에 대해 애플리케이션에서 구현하는 콜백 함수의 서명입니다.
ImageAbort *ImageAbort*는 애플리케이션에서 구현하는 콜백 함수의 서명입니다. 썸네일 이미지를 만들거나 검색하거나 이미지를 그리는 과정에서 GDI+는 이 함수를 호출하여 프로세스를 중단할 수 있는 기회를 제공합니다.

구조체

GdiplusAbort GdiplusAbort 구조체는 Windows GDI+가 시간이 많이 걸리는 렌더링 작업 중에 애플리케이션 정의 Abort 메서드를 주기적으로 호출할 수 있는 메커니즘을 제공합니다.

열거형

상태 상태 열거형은 Windows GDI+ 메서드 호출의 결과를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

genericusbfnioctl.h 헤더

아티클2023. 08. 24.

이 헤더는 USB 드라이버 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- USB 드라이버 참조

genericusbfnioctl.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

IOCTLs

IOCTL_GENERICUSBFN_ACTIVATE_USB_BUS 이 I/O 제어 코드(IOCTL)는 사용자 모드 서비스 또는 애플리케이션에서 전송되어 GenericUSBFn.sys USB(유니버설 직렬 버스)를 활성화하도록 알립니다. 활성화되면 버스는 버스 이벤트를 처리하고 트래픽을 처리할 준비가 됩니다.
IOCTL_GENERICUSBFN_BUS_EVENT_NOTIFICATION 이 I/O 제어 코드(IOCTL)는 USB(유니버설 직렬 버스) 이벤트에 등록하기 위해 사용자 모드 서비스 또는 애플리케이션에서 전송됩니다.
IOCTL_GENERICUSBFN_CONTROL_STATUS_HANDSHAKE_IN 이 I/O 제어 코드(IOCTL)는 사용자 모드 서비스 또는 애플리케이션에서 엔드포인트 0에서 IN 방향으로 핸드셰이크를 상태 길이가 0인 제어를 요청하기 위해 전송됩니다.
IOCTL_GENERICUSBFN_CONTROL_STATUS_HANDSHAKE_OUT 이 I/O 제어 코드(IOCTL)는 OUT 방향으로 엔드포인트 0에서 핸드셰이크를 상태 길이가 0인 제어를 완료하기 위해 사용자 모드 서비스 또는 애플리케이션에서 전송됩니다.
IOCTL_GENERICUSBFN_DEACTIVATE_USB_BUS 이 IOCTL 코드는 nevtot가 지원됩니다.
IOCTL_GENERICUSBFN_GET_CLASS_INFO 이 I/O 제어 코드(IOCTL)는 레지스트리에 구성된 디바이스의 사용 가능한 파이프에 대한 정보를 검색하기 위해 사용자 모드 서비스 또는 애플리케이션에서 전송됩니다.
IOCTL_GENERICUSBFN_GET_CLASS_INFO_EX 이 I/O 제어 코드(IOCTL)는 레지스트리에 구성된 디바이스의 사용 가능한 파이프에 대한 정보를 검색하기 위해 사용자 모드 서비스 또는 애플리케이션에서 전송됩니다.

IOCTL_GENERICUSBFN_GET_INTERFACE_DESCRIPTOR_SET 이 I/O 제어 코드(IOCTL)는 사용자 모드 서비스 또는 애플리케이션에서 전송되어 디바이스의 함수에 대해 설정된 전체 인터페이스 설명자를 가져옵니다. 이 IOCTL 요청은 전체 디바이스에 대해 설정된 인터페이스 설명자를 검색하지 않습니다. 디바이스의 함수에 대해 설정된 USB(유니버설 직렬 버스) 인터페이스 설명자입니다.
IOCTL_GENERICUSBFN_GET_PIPE_STATE 이 I/O 제어 코드(IOCTL)는 지정된 USB(유니버설 직렬 버스) 파이프의 상태를 가져오기 위해 사용자 모드 서비스 또는 애플리케이션에서 전송됩니다.
IOCTL_GENERICUSBFN_REGISTER_USB_STRING 이 I/O 제어 코드(IOCTL)는 문자열 설명자를 등록하기 위해 사용자 모드 서비스 또는 애플리케이션에서 전송됩니다. USB(유니버설 직렬 버스) 문자열 설명자입니다.
IOCTL_GENERICUSBFN_SET_PIPE_STATE 이 I/O 제어 코드(IOCTL)는 지정된 USB(유니버설 직렬 버스) 파이프의 상태를 설정하기 위해 사용자 모드 서비스 또는 애플리케이션에서 전송됩니다.
IOCTL_GENERICUSBFN_TRANSFER_IN 이 I/O 제어 코드(IOCTL)는 사용자 모드 서비스 또는 애플리케이션에서 전송되어 입력 버퍼의 지정된 파이프 ID에 해당하는 엔드포인트에서 IN 방향 전송을 실행합니다. (IOCTL_GENERICUSBFN_TRANSFER_IN)
IOCTL_GENERICUSBFN_TRANSFER_IN_APPEND_ZERO_PKT 이 I/O 제어 코드(IOCTL)는 사용자 모드 서비스 또는 애플리케이션에서 전송되어 입력 버퍼의 지정된 파이프 ID에 해당하는 엔드포인트에서 IN 방향 전송을 실행합니다. (IOCTL_GENERICUSBFN_TRANSFER_IN_APPEND_ZERO_PKT)
IOCTL_GENERICUSBFN_TRANSFER_OUT 이 I/O 제어 코드(IOCTL)는 입력 버퍼의 지정된 파이프 ID에 해당하는 엔드포인트에서 OUT 방향 전송을 실행하기 위해 사용자 모드 서비스 또는 애플리케이션에서 전송됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

gpedit.h 헤더

아티클2023. 08. 24.

이 헤더는 그룹 정책 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 그룹 정책 API

gpedit.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IGPEInformation IGPEInformation 인터페이스는 MMC(Microsoft Management Console) 확장 스냅인이 그룹 정책 개체 편집기와 통신하는 메서드를 제공합니다. MMC에 대한 자세한 내용은 Microsoft 관리 콘솔을 참조하세요.
IGroupPolicyObject IGroupPolicyObject 인터페이스는 그룹 정책 개체 편집기를 사용하지 않고 GPO를 직접 만들고 수정하는 메서드를 제공합니다.
IRSOPInformation IRSOPInformation 인터페이스는 MMC(Microsoft Management Console) 확장 스냅인이 기본 RSoP(결과 정책 집합) 스냅인과 통신하는 메서드를 제공합니다. MMC에 대한 자세한 내용은 Microsoft 관리 콘솔을 참조하세요.

Functions

BrowseForGPO BrowseForGPO 함수는 사용자가 GPO를 열거나 만들 수 있는 GPO 브라우저 대화 상자를 만듭니다.
CreateGPOLink CreateGPOLink 함수는 지정된 GPO와 지정된 사이트, 도메인 또는 조직 구성 단위 간에 링크를 만듭니다.
DeleteAllGPOLinks

DeleteAllGPOLinks 함수는 지정된 사이트, 도메인 또는 조직 구성 단위에 대한 모든 GPO 링크를 삭제합니다.
DeleteGPOLink DeleteGPOLink 함수는 지정된 GPO와 지정된 사이트, 도메인 또는 조직 구성 단위 간의 링크를 삭제합니다.
ExportRSoPData ExportRSoPData 함수는 RSoP 정보가 포함된 WMI 네임스페이스를 데이터 파일로 내보냅니다. 함수는 ImportRSoPData 함수를 호출하여 WMI 네임스페이스로 가져올 수 있는 데이터 파일에 정보를 씁니다.
ImportRSoPData ImportRSoPData 함수는 RSoP 데이터가 포함된 데이터 파일을 WMI 네임스페이스로 가져옵니다. 파일은 ExportRSoPData 함수를 호출하여 생성해야 합니다.

구조체

GPOBROWSEINFO GPOBROWSEINFO 구조체에는 BrowseForGPO 함수가 GPO 브라우저 대화 상자를 초기화하는 데 사용하는 정보가 포함되어 있습니다. 사용자가 대화 상자를 닫은 후 시스템은 이 구조에서 사용자의 작업에 대한 정보를 반환합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

gpmgmt.h 헤더

아티클2023. 08. 24.

이 헤더는 그룹 정책 관리 콘솔에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 그룹 정책 관리 콘솔

gpmgmt.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IGPM IGPM 인터페이스는 GPMC(그룹 정책 관리 콘솔)의 다른 인터페이스에 액세스하는 메서드와 다양한 검색 작업을 수행할 수 있는 다른 개체를 만드는 메서드를 제공합니다.
IGPM2 IGPM2 인터페이스는 GPMC(그룹 정책 관리 콘솔)의 IGPM 인터페이스의 GPMBackupDir 및 InitializeReporting 메서드를 확장합니다.
IGPMAsyncCancel IGPMAsyncCancel 인터페이스에 대한 포인터는 클라이언트가 비동기적으로 호출하는 GPMC(그룹 정책 Management Console) 메서드에 의해 클라이언트에 반환됩니다.
IGPMAsyncProgress IGPMAsyncProgress 인터페이스는 클라이언트에서 구현하고 비동기적으로 실행할 수 있는 GPMC(그룹 정책 Management Console) 메서드에 입력 매개 변수로 전달할 수 있습니다.
IGPMBackup IGPMBackup 인터페이스는 GPMBackup 개체를 삭제하고 GPMBackup 개체의 다양한 속성을 검색할 수 있는 메서드를 지원합니다.
IGPMBackupCollection IGPMBackupCollection 인터페이스에는 GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 애플리케이션이 GPMBackup 개체 컬렉션에 액세스할 수 있도록 하는 메서드가 포함되어 있습니다.
IGPMBackupDir IGPMBackupDir 인터페이스는 GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 GPMBackup 및 GPMBackupCollection 개체를 쿼리할 수 있는 메서드를 지원합니다.

IGPMBackupDirEx IGPMBackupDirEx 인터페이스는 GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 GPMBackup, GPMBackupCollection, GPMStarterGPOBackup 및 GPMStarterGPOBackupCollection 개체를 쿼리할 수 있는 메서드를 지원합니다.
IGPMClientSideExtension IGPMClientSideExtension 인터페이스는 GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 클라이언트 쪽 확장 속성을 쿼리할 수 있는 메서드를 지원합니다.
IGPMConstants IGPMConstants 인터페이스는 여러 GPMC(그룹 정책 Management Console) 상수의 값을 검색하는 메서드를 지원합니다. GPMConstants 개체를 만들려면 IGPM::GetConstants 메서드를 호출합니다.
IGPMConstants2 IGPMConstants2 인터페이스는 여러 GPMC(그룹 정책 관리 콘솔) 상수의 값을 검색하는 메서드를 지원합니다.
IGPMCSECollection IGPMCSECollection 인터페이스에는 GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 애플리케이션이 CSE(클라이언트 쪽 확장) 컬렉션을 쿼리할 수 있도록 하는 메서드가 포함되어 있습니다.
IGPMDomain 지정된 도메인을 나타내며, GPMC(그룹 정책 Management Console) 인터페이스를 사용할 때 SOM(관리) 개체의 scope 쿼리하고, GPO를 만들고, 복원하고, 쿼리하고, WMI 필터를 만들고 쿼리할 수 있는 메서드를 지원합니다.
IGPMDomain2 지정된 도메인을 나타내며, 그룹 정책 관리 콘솔(GPMC) 인터페이스를 사용할 때 SOM(관리) 개체의 scope 쿼리하고, Starter GPO를 만들고, 복원 및 쿼리하고, WMI 필터를 만들고 쿼리할 수 있는 메서드를 지원합니다.
IGPMGPO IGPMGPO 인터페이스는 디렉터리 서비스에서 GPO(그룹 정책 Objects)를 관리할 수 있는 메서드를 지원합니다.
IGPMGPO2 IGPMGPO2 인터페이스는 디렉터리 서비스에서 GPO(그룹 정책 개체) 및 Starter 그룹 정책 개체를 관리할 수 있는 메서드를 지원합니다.

IGPMGPOCollection IGPMGPOCollection 인터페이스에는 GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 애플리케이션이 GPO(그룹 정책 Objects) 컬렉션에 액세스할 수 있는 메서드가 포함되어 있습니다.
IGPMGPOLink IGPMGPOLink 인터페이스는 SOM(관리 scope)에서 GPO 링크를 제거하고 링크 사용 및 적용을 포함하여 GPO 링크의 다양한 속성을 설정하고 검색할 수 있는 메서드를 지원합니다.
IGPMGPOLinksCollection IGPMGPOLinksCollection 인터페이스에는 애플리케이션이 GPMC(그룹 정책 Management) 인터페이스를 사용할 때 GPO 링크 컬렉션에 액세스할 수 있도록 하는 메서드가 포함되어 있습니다.
IGPMMapEntry IGPMMapEntry 인터페이스는 맵 항목에 대한 액세스를 제공합니다.
IGPMMapEntryCollection IGPMMapEntryCollection 인터페이스를 사용하면 애플리케이션이 맵 항목 개체에 액세스할 수 있습니다.
IGPMMigrationTable IGPMMigrationTable 인터페이스는 마이그레이션 테이블에 대한 인터페이스를 제공합니다.
IGPMPermission IGPMPermission 인터페이스에는 GPMC를 사용할 때 권한 관련 속성을 검색하는 메서드가 포함되어 있습니다.
IGPMResult IGPMResult 인터페이스에는 복원, 가져오기, 복사 및 백업과 같은 다양한 유형의 GPO 처리 작업을 수행하는 동안 상태 메시지 정보를 검색하는 메서드가 포함되어 있습니다.
IGPMRSOP IGPMRSOP 인터페이스는 로깅 및 계획 모드 모두에서 RSoP(결과 정책 집합) 쿼리 만들기를 지원하는 메서드를 제공합니다.
IGPMSearchCriteria IGPMSearchCriteria 인터페이스를 사용하면 GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 검색 작업에 사용할 조건을 정의할 수 있습니다. GPMSearchCriteria 개체를 만들려면 IGPM::CreateSearchCriteria 메서드를 호출합니다.

IGPMSecurityInfo IGPMSecurityInfo 인터페이스는 GPMSecurityInfo 컬렉션의 메서드를 정의합니다. 이 컬렉션은 SOM(관리 scope), GPO 또는 WMI 필터와 같은 특정 개체에 설정할 수 있는 정책 관련 권한 집합을 나타냅니다.
IGPMSitesContainer IGPMSitesContainer 인터페이스는 포리스트의 사이트를 나타내는 SOM(관리) 개체의 scope 액세스하는 데 필요한 메서드를 제공합니다.
IGPMSOM IGPMSOM 인터페이스에는 SOM(scope 관리)에 대한 GPO 링크를 만들고 검색하고 SOM에 대한 보안 특성 및 다양한 속성을 설정하고 검색할 수 있는 메서드가 포함되어 있습니다. SOM은 사이트, 도메인 또는 OU일 수 있습니다.
IGPMSOMCollection IGPMSOMCollection 인터페이스는 GPMSOM 개체의 컬렉션을 나타냅니다.
IGPMStarterGPO IGPMStarterGPO 인터페이스는 디렉터리 서비스에서 스타터 그룹 정책 개체(GPO)를 관리할 수 있는 메서드를 지원합니다.
IGPMStarterGPOBackup IGPMStarterGPOBackup 인터페이스는 GPMStarterGPOBackup 개체를 삭제하고 GPMStarterGPOBackup 개체의 다양한 속성을 검색할 수 있는 메서드를 지원합니다.
IGPMStarterGPOBackupCollection IGPMStarterGPOBackupCollection 인터페이스에는 GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 애플리케이션이 GPMStarterGPOBackup 개체 컬렉션에 액세스할 수 있도록 하는 메서드가 포함되어 있습니다.
IGPMStarterGPOCollection IGPMStarterGPOCollection 인터페이스에는 GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 애플리케이션이 GPO(그룹 정책 Objects) 컬렉션에 액세스할 수 있도록 하는 메서드가 포함되어 있습니다.
IGPMStatusMessage IGPMStatusMessage 인터페이스에는 GPO 작업과 관련된 상태 메시지의 다양한 속성을 검색하는 속성 메서드가 포함되어 있습니다.

IGPMStatusMsgCollection IGPMStatusMsgCollection 인터페이스에는 애플리케이션이 GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 상태 메시지 컬렉션에 액세스할 수 있는 메서드가 포함되어 있습니다.
IGPMTrustee IGPMTrustee 인터페이스에는 GPMC(그룹 정책 관리 콘솔)를 사용할 때 지정된 트러스티에 대한 정보를 검색하는 메서드가 포함되어 있습니다.
IGPMWMIFilter IGPMWMIFilter 인터페이스에는 WMI 필터에 대한 보안 특성 및 다양한 속성을 설정하고 검색할 수 있는 메서드가 포함되어 있습니다. WMI 필터 쿼리는 WQL(WMI 쿼리 언어)을 사용하여 지정됩니다.
IGPMWMIFilterCollection IGPMWMIFilterCollection 인터페이스에는 GPMC(그룹 정책 관리 콘솔) 인터페이스를 사용할 때 애플리케이션이 WMI 필터 컬렉션에 액세스할 수 있도록 하는 메서드가 포함되어 있습니다.

열거형

GPMBackupType 만든 백업 유형입니다.
GPMPermissionType 범주, 범주에 포함된 권한 및 적용할 수 있는 개체입니다.
GPMReportingOptions 그룹 정책 관리 콘솔 보고서에 대한 옵션입니다.
GPMSearchProperty 검색 조건의 속성입니다.
GPMStarterGPOType Starter 그룹 정책 개체는 시스템 Starter 그룹 정책 개체 또는 사용자 지정 Starter 그룹 정책 개체입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

guiddef.h 헤더

아티클2023. 08. 24.

이 헤더는 COM(구성 요소 개체 모델)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)

guiddef.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

IsEqualCLSID 두 CLSID가 같은지 여부를 나타내는 부울 값으로 계산됩니다.
IsEqualGUID 두 GUID가 같은지 여부를 나타내는 부울 값으로 계산됩니다.
IsEqualIID 두 IID가 같은지 여부를 나타내는 부울 값으로 평가됩니다.

구조체

GUID GUID 구조체(guiddef.h)는 COM 인터페이스, COM 클래스 개체 또는 EPV(관리자 진입점 벡터)와 같은 개체를 식별합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

handleapi.h 헤더

아티클2023. 08. 24.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

handleapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CloseHandle 열린 개체 핸들을 닫습니다.
CompareObjectHandles 두 개체 핸들을 비교하여 동일한 기본 커널 개체를 참조하는지 확인합니다.
DuplicateHandle 개체 핸들을 복제합니다.
GetHandleInformation 개체 핸들의 특정 속성을 검색합니다.
SetHandleInformation 개체 핸들의 특정 속성을 설정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

hbaapi.h 헤더

아티클2023. 08. 24.

이 헤더는 HBA(호스트 버스 어댑터)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- HBA(호스트 버스 어댑터)

hbaapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

HBA_CloseAdapter
HBA_FreeLibrary
HBA_GetAdapterAttributes
HBA_GetAdapterName
HBA_GetAdapterPortAttributes
HBA_GetBindingCapability
HBA_GetBindingSupport
HBA_GetDiscoveredPortAttributes
HBA_GetEventBuffer
HBA_GetFC4Statistics
HBA_GetFcpPersistentBinding
HBA_GetFCPStatistics

HBA_GetFcpTargetMapping
HBA_GetFcpTargetMappingV2
HBA_GetNumberOfAdapters
HBA_GetPersistentBindingV2
HBA_GetPortAttributesByWWN
HBA_GetPortStatistics
HBA_GetRNIDMgmtInfo
HBA_GetSBStatistics
HBA_GetSBTargetMapping
HBA_GetVendorLibraryAttributes
HBA_GetVersion
HBA_GetWrapperLibraryAttributes
HBA_LoadLibrary
HBA_OpenAdapter HBA(호스트 버스 어댑터)를 열고 핸들을 반환합니다.
HBA_OpenAdapterByWWN
HBA_RefreshAdapterConfiguration

HBA_RefreshInformation
HBA_RegisterForAdapterAddEvents 새 어댑터가 시스템에 추가될 때 호출할 표시된 사용자 콜백 루틴을 등록합니다.
HBA_RegisterForAdapterEvents 어댑터 이벤트에 대해 알 수 있는 사용자 루틴을 선언합니다.
HBA_RegisterForAdapterPortEvents 포트 이벤트가 발생할 때 호출할 표시된 사용자 콜백 루틴을 등록합니다.
HBA_RegisterForAdapterPortStatEvents 포트 통계 이벤트가 발생할 때 호출할 표시된 사용자 콜백 루틴을 등록합니다.
HBA_RegisterForLinkEvents
HBA_RegisterForTargetEvents 지정된 대상 또는 어댑터와 연결된 모든 대상을 사용하여 대상 이벤트를 등록합니다.
HBA_RegisterLibrary
HBA_RegisterLibraryV2
HBA_RemoveAllPersistentBindings
HBA_RemoveCallback 콜백 루틴을 등록 해제합니다.
HBA_RemovePersistentBinding
HBA_ResetStatistics
HBA_SBDskGetCapacity
HBA_ScsiInquiryV2

HBA_ScsiReadCapacityV2
HBA_ScsiReportLUNsV2
HBA_SendCTPassThru
HBA_SendCTPassThruV2
HBA_SendLIRR
HBA_SendReadCapacity
HBA_SendReportLUNs
HBA_SendRLS
HBA_SendRNID
HBA_SendRNIDV2
HBA_SendRPL
HBA_SendRPS
HBA_SendScsiInquiry
HBA_SendSRL
HBA_SetBindingSupport
HBA_SetPersistentBindingV2

HBA_SetRNIDMgmtInfo
SMHBA_GetAdapterAttributes
SMHBA_GetAdapterPortAttributes
SMHBA_GetBindingCapability
SMHBA_GetBindingSupport
SMHBA_GetDiscoveredPortAttributes
SMHBA_GetFCPhyAttributes
SMHBA_GetLUNStatistics
SMHBA_GetNumberOfPorts
SMHBA_GetPersistentBinding
SMHBA_GetPhyStatistics
SMHBA_GetPortAttributesByWWN
SMHBA_GetPortType
SMHBA_GetProtocolStatistics
SMHBA_GetSASPhyAttributes
SMHBA_GetTargetMapping
SMHBA_GetVendorLibraryAttributes

SMHBA_GetVersion
SMHBA_GetWrapperLibraryAttributes
SMHBA_RegisterForAdapterAddEvents
SMHBA_RegisterForAdapterEvents
SMHBA_RegisterForAdapterPhyStatEvents
SMHBA_RegisterForAdapterPortEvents
SMHBA_RegisterForAdapterPortStatEvents
SMHBA_RegisterForTargetEvents
SMHBA_RegisterLibrary
SMHBA_RemoveAllPersistentBindings
SMHBA_RemovePersistentBinding
SMHBA_ScsiInquiry
SMHBA_ScsiReadCapacity
SMHBA_ScsiReportLuns
SMHBA_SendECHO
SMHBA_SendSMPPassThru
SMHBA_SendTEST

SMHBA_SetBindingSupport
SMHBA_SetPersistentBinding

구조체

HBA_ADAPTERATTRIBUTES
HBA_DEVICESELFDESC
HBA_EVENTINFO
HBA_FC4STATISTICS
HBA_FC4TYPES
HBA_FCPBINDING
HBA_FCPBINDING2
HBA_FCPBINDINGENTRY
HBA_FCPBINDINGENTRY2
HBA_FCPID
HBA_FCPSCSIENTRY
HBA_FCPSCSIENTRYV2
HBA_FCPTARGETMAPPING

HBA_FCPTARGETMAPPINGV2
HBA_IPADDRESS
HBA_LIBRARYATTRIBUTES
HBA_LINK_EVENTINFO
HBA_LUID
HBA_MGMTINFO
HBA_NED
HBA_PORTATTRIBUTES
HBA_PORTSTATISTICS
HBA_PTY_EVENTINFO
HBA_RSCN_EVENTINFO
HBA_SBDEVID
HBA_SBDSKCAPACITY
HBA_SCSIID
HBA_WWN
SMHBA_ADAPTERATTRIBUTES
SMHBA_BINDING

SMHBA_BINDINGENTRY
SMHBA_FC_PHY
SMHBA_FC_PORT
SMHBA_LIBRARYATTRIBUTES
SMHBA_LUID
SMHBA_PHYSTATISTICS
SMHBA_PORT
SMHBA_PORTATTRIBUTES
SMHBA_PORTLUN
SMHBA_PROTOCOLSTATISTICS
SMHBA_SAS_PHY
SMHBA_SAS_PORT
SMHBA_SASPHYSTATISTICS
SMHBA_SCSIENTRY
SMHBA_SCSIID
SMHBA_SCSILUN
SMHBA_TARGETMAPPING

열거형

HBA_FCPBINDINGTYPE
HBA_WWNTYPE

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

heapapi.h 헤더

아티클2023. 08. 24.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

heapapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

GetProcessHeap 호출 프로세스의 기본 힙에 대한 핸들을 검색합니다.
GetProcessHeaps 활성 힙 수를 반환하고 호출 프로세스에 대한 모든 활성 힙에 대한 핸들을 검색합니다.
HeapAlloc 힙에서 메모리 블록을 할당합니다. 할당된 메모리는 움직일 수 없습니다.
HeapCompact 지정된 힙에서 커밋된 가장 큰 자유 블록의 크기를 반환합니다. 사용 가능한 전역 플래그에서 힙 병합 사용 안 함이 설정된 경우 이 함수는 힙의 인접한 사용 가능한 메모리 블록도 병합합니다.
HeapCreate 호출 프로세스에서 사용할 수 있는 프라이빗 힙 개체를 만듭니다. 함수는 프로세스의 가상 주소 공간에 공간을 예약하고 이 블록의 지정된 초기 부분에 대한 물리적 스토리지를 할당합니다.
HeapDestroy 지정된 힙 개체를 제거합니다. 프라이빗 힙 개체의 모든 페이지를 커밋 해제하고 해제하고 힙에 대한 핸들을 무효화합니다.
HeapFree HeapAlloc 또는 HeapReAlloc 함수에 의해 힙에서 할당된 메모리 블록을 해제합니다.
HeapLock 지정된 힙과 연결된 중요한 섹션 개체 또는 잠금을 획득하려고 시도합니다.

HeapQueryInformation 지정된 힙에 대한 정보를 검색합니다.
HeapReAlloc 힙에서 메모리 블록을 다시 할당합니다. 이 함수를 사용하면 메모리 블록의 크기를 조정하고 다른 메모리 블록 속성을 변경할 수 있습니다.
HeapSetInformation 지정된 힙에 대한 기능을 사용하도록 설정합니다.
HeapSize HeapAlloc 또는 HeapReAlloc 함수에 의해 힙에서 할당된 메모리 블록의 크기를 검색합니다.
HeapSummary HeapSummary 함수(heapapi.h)는 지정된 힙을 요약합니다.
HeapUnlock 지정된 힙과 연결된 중요한 섹션 개체 또는 잠금의 소유권을 해제합니다.
HeapValidate 지정된 힙의 유효성을 검사합니다. 함수는 힙의 모든 메모리 블록을 검사하고 힙 관리자가 유지 관리하는 힙 제어 구조가 일관된 상태임을 확인합니다.
HeapWalk 지정된 힙의 메모리 블록을 열거합니다.

구조체

HEAP_SUMMARY HEAP_SUMMARY 구조체(heapapi.h)는 HeapSummary 함수를 호출하여 검색된 힙 요약을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

highlevelmonitorconfigurationapi.h 헤더

아티클2023. 08. 24.

이 헤더는 모니터 구성에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 구성 모니터링

highlevelmonitorconfigurationapi.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DegaussMonitor 모니터를 분해합니다.
GetMonitorBrightness 모니터의 최소, 최대 및 현재 밝기 설정을 검색합니다.
GetMonitorCapabilities 모니터의 구성 기능을 검색합니다. 이 함수를 호출하여 모니터에서 지원하는 상위 수준 모니터 구성 함수를 확인합니다.
GetMonitorColorTemperature 모니터의 현재 색 온도를 검색합니다.
GetMonitorContrast 모니터의 최소, 최대 및 현재 대비 설정을 검색합니다.
GetMonitorDisplayAreaPosition 모니터의 최소, 최대 및 현재 가로 또는 세로 위치를 검색합니다.
GetMonitorDisplayAreaSize 모니터의 최소, 최대 및 현재 너비 또는 높이를 검색합니다.
GetMonitorRedGreenOrBlueDrive 모니터의 빨간색, 녹색 또는 파란색 드라이브 값을 검색합니다.

GetMonitorRedGreenOrBlueGain 모니터의 빨간색, 녹색 또는 파란색 게인 값을 검색합니다.
GetMonitorTechnologyType 모니터에서 사용하는 기술 유형을 검색합니다.
RestoreMonitorFactoryColorDefaults 모니터의 색 설정을 공장 기본값으로 복원합니다.
RestoreMonitorFactoryDefaults 모니터의 설정을 공장 기본값으로 복원합니다.
SaveCurrentMonitorSettings 현재 모니터 설정을 디스플레이의 비휘발성 스토리지에 저장합니다. (SaveCurrentMonitorSettings)
SetMonitorBrightness 모니터의 밝기 값을 설정합니다.
SetMonitorColorTemperature 모니터의 색 온도를 설정합니다.
SetMonitorContrast 모니터의 대비 값을 설정합니다.
SetMonitorDisplayAreaPosition 모니터 표시 영역의 가로 또는 세로 위치를 설정합니다.
SetMonitorDisplayAreaSize 모니터 표시 영역의 너비 또는 높이를 설정합니다.
SetMonitorRedGreenOrBlueDrive 모니터의 빨간색, 녹색 또는 파란색 드라이브 값을 설정합니다.
SetMonitorRedGreenOrBlueGain 모니터의 빨간색, 녹색 또는 파란색 게인 값을 설정합니다.

열거형

MC_COLOR_TEMPERATURE 모니터의 색 온도를 설명합니다.
MC_DISPLAY_TECHNOLOGY_TYPE 모니터 디스플레이 기술을 식별합니다.
MC_DRIVE_TYPE 모니터의 빨간색, 녹색 또는 파란색 드라이브를 설정하거나 가져올지 여부를 지정합니다.
MC_GAIN_TYPE 모니터의 빨간색, 녹색 또는 파랑 게인을 가져올지 또는 설정할지를 지정합니다.
MC_POSITION_TYPE 모니터 표시 영역의 세로 또는 가로 위치를 가져올지 또는 설정할지를 지정합니다.
MC_SIZE_TYPE 모니터 표시 영역의 너비 또는 높이를 가져올지 또는 설정할지를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

holographicspaceinterop.h 헤더

아티클2023. 08. 24.

이 헤더는 Windows Mixed Reality 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Mixed Reality

holographicspaceinterop.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IHolographicSpaceInterop WinRT HolographicSpace 개체와의 상호 운용성을 사용하도록 설정하고 홀로그램 장면을 나타내는 HolographicSpace 멤버에 대한 액세스를 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

hrtfapoapi.h 헤더

아티클2023. 08. 24.

이 헤더는 XAudio2 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- XAudio2 API

hrtfapoapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IXAPOHrtfParameters HRTF(헤드 관련 전송 함수)가 소리에 적용되는 방식을 제어하는 매개 변수를 설정하는 데 사용되는 인터페이스입니다.

Functions

CreateHrtfApo HRTF(헤드 관련 전송 함수) 처리를 위한 IXAPO 인터페이스의 instance 만듭니다.

구조체

HrtfApoInit HRTF 공간 오디오를 초기화하는 데 사용되는 매개 변수를 지정합니다.
HrtfDirectivity 기본 직접성 패턴 설명자입니다. 소리에 적용되는 직접성의 유형을 설명합니다.
HrtfDirectivityCardioid 심장 지향성 패턴을 설명합니다.

HrtfDirectivityCone 원뿔 방향성을 설명합니다.
HrtfDistanceDecay 거리 기반 감쇠 동작을 설명합니다.
HrtfOrientation HRTF 직접성 개체의 방향을 나타냅니다.
HrtfPosition 오른손 좌표계를 사용하여 3D 공간의 위치를 나타냅니다.

열거형

HrtfDirectivityType 여러 주식 지향성 패턴 중 하나를 나타냅니다.
HrtfDistanceDecayType 소리에 적용된 거리 기반 감쇠 유형을 나타냅니다.
Hrtf환경 여러 주식 환경 유형 중 하나를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

hstring.h 헤더

아티클2023. 08. 24.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

hstring.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

HSTRING_HEADER HSTRING의 헤더를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

htmlhelp.h 헤더

아티클2023. 08. 24.

이 헤더는 Microsoft HTML 도움말 1.4에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft HTML 도움말 1.4

htmlhelp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

HtmlHelpA 도움말 창을 표시합니다. (ANSI)
HtmlHelpW 도움말 창을 표시합니다. (유니코드)

구조체

HH_AKLINK 검색하려는 하나 이상의 ALink 이름 또는 KLink 키워드를 지정하려면 이 구조를 사용합니다.
HH_FTS_QUERY 전체 텍스트 검색에 이 구조를 사용합니다.
HH_POPUP 팝업 창의 특성을 지정하거나 수정하려면 이 구조를 사용합니다.
HH_WINTYPE 창 형식의 특성을 지정하거나 수정하려면 이 구조를 사용합니다.
HHN_NOTIFY

이 구조를 사용하여 탐색된 항목의 파일 이름을 반환하거나 생성된 도움말 창의 창 형식 이름을 반환합니다.
HHNTRACK 이 구조체는 현재 항목의 파일 이름과 도구 모음에서 숨기기 단추를 클릭하여 탐색 창을 숨기는 등 발생할 사용자 작업을 지정하는 상수를 반환합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

http.h 헤더

아티클2023. 03. 16.

이 헤더는 HTTP Server API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- HTTP Server API(HTTP 서버 API)

http.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

HTTP_EQUAL_VERSION HTTP_VERSION 구조체가 지정된 주/부 버전 조합과 같거나 0이 아니면 0이 아닌 값을 반환합니다.
HTTP_GREATER_EQUAL_VERSION HTTP_GREATER_EQUAL_VERSION 매크로는 HTTP_VERSION 구조체가 지정된 주/부 버전 조합보다 크거나 같거나 0이 아니면 0이 아닌 값을 반환합니다.
HTTP_GREATER_VERSION HTTP_VERSION 구조체가 지정된 주/부 버전 조합보다 크거나 그렇지 않으면 0이 아닌 값을 반환합니다.
HTTP_IS_NULL_ID HTTP_OPAQUE_ID NULL인지 확인합니다.
HTTP_LESS_EQUAL_VERSION HTTP_VERSION 구조체가 지정된 주/부 버전 조합보다 작거나 같거나 0이 아니면 0이 아닌 값을 반환합니다.
HTTP_LESS_VERSION HTTP_VERSION 구조체가 지정된 주/부 버전 조합보다 작거나 그렇지 않으면 0이 아닌 값을 반환합니다.
HTTP_NOT_EQUAL_VERSION HTTP_VERSION 구조체가 지정된 주/부 버전 조합보다 크거나 작거나 같은 경우 0이 아닌 값을 반환합니다.

HTTP_SET_NULL_ID HTTP_SET_NULL_ID 매크로는 HTTP_OPAQUE_ID NULL로 설정합니다.
HTTP_SET_VERSION 지정된 HTTP_VERSION 구조를 지정된 주/부 버전 조합으로 설정합니다.
HttpAddFragmentToCache HttpAddFragmentToCache 함수는 검색할 수 있는 지정된 이름의 데이터 조각을 캐시하거나 지정된 이름으로 캐시된 데이터를 업데이트합니다.
HttpAddUrl 지정된 URL과 일치하는 요청이 지정된 HTTP 서버 API 요청 큐로 라우팅되도록 지정된 URL을 등록합니다.
HttpAddUrlToUrlGroup URL 그룹 ID로 식별된 URL 그룹에 지정된 URL을 추가합니다.
HTTPAPI_EQUAL_VERSION HTTPAPI_VERSION 구조체가 지정된 주/부 버전 조합과 정확히 같거나 0이 아니면 0이 아닌 값을 반환합니다.
HTTPAPI_GREATER_VERSION HTTPAPI_VERSION 구조체가 지정된 주/부 버전 조합보다 크거나 그렇지 않으면 0이 아닌 값을 반환합니다.
HTTPAPI_LESS_VERSION HTTPAPI_VERSION 구조체가 지정된 주/부 버전 조합보다 작거나 그렇지 않으면 0이 아닌 값을 반환합니다.
HTTPAPI_VERSION_GREATER_OR_EQUAL HTTPAPI_VERSION_GREATER_OR_EQUAL HTTPAPI_VERSION 구조체가 지정된 주/부 버전 조합보다 크거나 같거나 0이 아니면 0이 아닌 값을 반환합니다.
HttpCancelHttpRequest HttpCancelHttpRequest 함수는 지정된 reqest를 취소합니다.
HttpCloseRequestQueue HttpCreateRequestQueue에서 만든 지정된 요청 큐에 대한 핸들을 닫습니다.

HttpCloseServerSession 서버 세션 ID로 식별되는 서버 세션을 삭제합니다.
HttpCloseUrlGroup URL 그룹 ID로 식별된 URL 그룹을 닫습니다.
HttpCreateHttpHandle 호출 애플리케이션에 대한 HTTP 요청 큐를 만들고 해당 애플리케이션에 대한 핸들을 반환합니다.
HttpCreateRequestQueue 새 요청 큐를 만들거나 기존 요청 큐를 엽니다.
HttpCreateServerSession 지정된 버전에 대한 서버 세션을 만듭니다.
HttpCreateUrlGroup 지정된 서버 세션 아래에 URL 그룹을 만듭니다.
HttpDeclarePush HTTP 서버 푸시에 사용할 리소스-하위 리소스 관계를 선언합니다. 그런 다음 HTTP.sys 기본 프로토콜, 연결, 클라이언트 및 정책에서 푸시 작업을 허용하는 경우 지정된 리소스에 대해 HTTP 2.0 서버 푸시를 수행합니다.
HttpDelegateRequestEx 원본 요청 큐에서 대상 요청 큐로 요청을 위임합니다.
HttpDeleteServiceConfiguration HTTP 서버 API 구성 저장소에서 IP 주소 또는 SSL 인증서와 같은 지정된 데이터를 한 번에 하나의 레코드로 삭제합니다.
HttpFindUrlGroupId URL 및 요청 큐에 대한 URL 그룹 ID를 검색합니다.
HttpFlushResponseCache 지정된 요청 큐와 연결된 HTTP Server API 캐시에서 사이트 부분이 지정된 UrlPrefix와 일치하는 이름을 가진 모든 응답 조각을 제거합니다.

HttpInitialize HttpInitialize 함수는 HTTP 서버 API 드라이버를 초기화하고, 아직 시작하지 않은 경우 시작하고, 응답 큐 만들기 및 기타 작업을 지원하기 위해 호출 애플리케이션에 대한 데이터 구조를 할당합니다.
HttpIsFeatureSupported 특정 기능이 지원되는지 여부를 확인합니다.
HttpPrepareUrl 다른 HTTP 함수에서 안전하고 사용할 수 있도록 정규화되지 않은 유니코드 또는 punycode URL을 구문 분석, 분석 및 정규화합니다.
HttpQueryRequestQueueProperty 지정된 핸들로 식별된 요청 큐의 속성을 쿼리합니다.
HttpQueryServerSessionProperty 지정된 서버 세션에서 서버 속성을 쿼리합니다.
HttpQueryServiceConfiguration 하나 이상의 HTTP 서버 API 구성 레코드를 검색합니다.
HttpQueryUrlGroupProperty 지정된 URL 그룹의 속성을 쿼리합니다.
HttpReadFragmentFromCache HttpReadFragmentFromCache 함수는 HTTP 서버 API 캐시에서 지정된 이름을 가진 응답 조각을 검색합니다.
HttpReceiveClientCertificate HttpReceiveClientCertificate 함수는 서버 애플리케이션에서 클라이언트 SSL 인증서 또는 CBT(채널 바인딩 토큰)를 검색하는 데 사용됩니다.
HttpReceiveHttpRequest 지정된 요청 큐에서 동기적으로 또는 비동기적으로 사용 가능한 다음 HTTP 요청을 검색합니다.
HttpReceiveRequestEntityBody 지정된 HTTP 요청에 대한 추가 엔터티 본문 데이터를 받습니다.

HttpRemoveUrl 시스템에서 지정된 UrlPrefix 문자열과 일치하는 요청을 지정된 요청 큐에 라우팅하지 않도록 합니다.
HttpRemoveUrlFromUrlGroup URL 그룹 ID로 식별된 그룹에서 지정된 URL을 제거합니다.
HttpSendHttpResponse 지정된 HTTP 요청에 HTTP 응답을 보냅니다.
HttpSendResponseEntityBody HTTP 응답과 연결된 엔터티 본문 데이터를 보냅니다.
HttpSetRequestProperty 새 속성을 설정하거나 지정된 요청에서 기존 속성을 수정합니다.
HttpSetRequestQueueProperty 새 속성을 설정하거나 지정된 핸들로 식별된 요청 큐에서 기존 속성을 수정합니다.
HttpSetServerSessionProperty 새 서버 세션 속성을 설정하거나 지정된 서버 세션에서 기존 속성을 수정합니다.
HttpSetServiceConfiguration HTTP Server API 구성 저장소에 대한 구성 레코드를 만들고 설정합니다.
HttpSetUrlGroupProperty 새 속성을 설정하거나 지정된 URL 그룹의 기존 속성을 수정합니다.
HttpShutdownRequestQueue 지정된 요청 큐 프로세스에 대한 큐 요청을 중지합니다.
HttpTerminate HTTP Server API에서 애플리케이션의 호출을 처리하는 데 사용하는 리소스를 정리합니다.
HttpUpdateServiceConfiguration 업데이트 HTTP Server API 구성 저장소 내의 구성 레코드에서 TLS(전송 계층 보안) 인증서를 지정하는 서비스 구성 매개 변수입니다.

HttpWaitForDemandStart 새 요청 큐 프로세스에서 처리할 수 있는 새 요청이 도착할 때까지 기다립니다.
HttpWaitForDisconnect 어떤 이유로든 HTTP 클라이언트에 대한 연결이 끊어진 경우 애플리케이션에 알합니다.
HttpWaitForDisconnectEx 이 함수는 HttpWaitForDisconnect에 대한 확장입니다.

구조체

HTTP_BANDWIDTH_LIMIT_INFO HTTP_BANDWIDTH_LIMIT_INFO 구조체는 대역폭 제한 제한을 설정하거나 쿼리하는 데 사용됩니다. 이 구조는 URL 그룹 또는 서버 세션에서 HttpServerBandwidthProperty를 설정하거나 쿼리할 때 사용해야 합니다.
HTTP_BINDING_INFO URL 그룹을 요청 큐와 연결하는 데 사용됩니다.
HTTP_BYTE_RANGE HTTP_BYTE_RANGE 구조는 캐시된 응답 조각, 파일 또는 기타 데이터 블록 내에서 바이트 범위를 지정하는 데 사용됩니다.
HTTP_CACHE_POLICY 캐시된 응답 조각과 연결된 캐시 정책을 정의하는 데 사용됩니다.
HTTP_CHANNEL_BIND_INFO HTTP_CHANNEL_BIND_INFO.
HTTP_CONNECTION_LIMIT_INFO URL 그룹에 대한 최대 미해결 연결 수 제한을 설정하거나 쿼리하는 데 사용됩니다.
HTTP_COOKED_URL 유효성이 검사된 정식 UTF-16 유니코드로 인코딩된 URL 요청 문자열과 포인터 및 요소 길이를 포함합니다.

HTTP_DATA_CHUNK 메모리, 파일 또는 HTTP Server API 응답 조각 캐시의 개별 데이터 블록을 나타냅니다.
HTTP_DELEGATE_REQUEST_PROPERTY_INFO 요청을 위임할 때 추가 속성 정보를 설명합니다.
HTTP_FLOWRATE_INFO 응답의 전송 속도입니다.
HTTP_KNOWN_HEADER HTTP 요청 또는 HTTP 응답의 알려진 헤더에 대한 헤더 값을 포함합니다.
HTTP_LISTEN_ENDPOINT_INFO IP 기반 URL이 특정 IP 주소 또는 와일드카드에서 수신 대기해야 하는지 여부를 제어합니다.
HTTP_LOG_DATA 로그 데이터의 형식을 지정하는 값을 포함합니다.
HTTP_LOG_FIELDS_DATA WC3 로깅을 사용할 때 HTTP 응답에 대해 기록되는 필드를 전달하는 데 사용됩니다.
HTTP_LOGGING_INFO URL 그룹 또는 서버 세션에서 서버 쪽 로깅을 사용하도록 설정하는 데 사용됩니다.
HTTP_MULTIPLE_KNOWN_HEADERS 둘 이상의 헤더가 필요한 경우 HTTP 응답에 포함된 헤더를 지정합니다.
HTTP_PROPERTY_FLAGS 속성 구성을 설정할 때 구성 개체에서 속성을 사용하거나 사용하지 않도록 설정하기 위해 속성 구성 구조에서 사용됩니다.
HTTP_QOS_SETTING_INFO QOS 설정에 대한 정보를 포함합니다.
HTTP_REQUEST_AUTH_INFO 수신 프로세스가 인증된 클라이언트를 가장하는 데 사용할 수 있는 클라이언트 토큰에 대한 핸들이 있는 요청의 인증 상태를 포함합니다.

HTTP_REQUEST_CHANNEL_BIND_STATUS HTTP_REQUEST_CHANNEL_BIND_STATUS.
HTTP_REQUEST_HEADERS HTTP 요청과 함께 전송된 헤더를 포함합니다.
HTTP_REQUEST_INFO 요청에 대한 추가 정보를 사용하여 HTTP_REQUEST 구조를 확장합니다.
HTTP_REQUEST_PROPERTY_STREAM_ERROR HTTP_REQUEST_PROPERTY_STREAM_ERROR 구조체는 HTTP/2 또는 HTTP/3 스트림 오류 코드를 나타냅니다.
HTTP_REQUEST_V1 HTTP_REQUEST 구조를 사용하여 특정 요청과 연결된 데이터를 반환합니다.
HTTP_REQUEST_V2 요청에 대한 자세한 정보를 사용하여 HTTP_REQUEST_V1 요청 구조를 확장합니다.
HTTP_RESPONSE_HEADERS HTTP 응답과 함께 전송된 헤더를 포함합니다.
HTTP_RESPONSE_INFO 응답에 대한 추가 정보를 사용하여 HTTP_RESPONSE 구조를 확장합니다.
HTTP_RESPONSE_V1 HTTP 응답과 연결된 데이터를 포함합니다.
HTTP_RESPONSE_V2 응답에 대한 자세한 정보를 사용하여 HTTP 버전 1.0 응답 구조를 확장합니다.
HTTP_SERVER_AUTHENTICATION_BASIC_PARAMS HTTP_SERVER_AUTHENTICATION_BASIC_PARAMS 구조에는 URL 그룹의 기본 인증에 대한 정보가 포함되어 있습니다. 이 구조체는 HTTP_SERVER_AUTHENTICATION_INFO 구조에 포함되어 있습니다.

HTTP_SERVER_AUTHENTICATION_DIGEST_PARAMS URL 그룹의 다이제스트 인증에 대한 정보를 포함합니다.
HTTP_SERVER_AUTHENTICATION_INFO URL 그룹 또는 서버 세션에서 서버 쪽 인증을 사용하도록 설정하는 데 사용됩니다.
HTTP_SERVICE_BINDING_A HTTP_SERVICE_BINDING_A.
HTTP_SERVICE_BINDING_BASE HTTP_SERVICE_BINDING_BASE.
HTTP_SERVICE_BINDING_W HTTP_SERVICE_BINDING_W.
HTTP_SERVICE_CONFIG_CACHE_SET HttpSetServiceConfiguration 함수의 pConfigInformation 매개 변수에 사용됩니다.
HTTP_SERVICE_CONFIG_IP_LISTEN_PARAM HTTP 서비스가 바인딩하는 IP 주소 목록에서 추가하거나 삭제할 IP 주소를 지정하는 데 사용됩니다.
HTTP_SERVICE_CONFIG_IP_LISTEN_QUERY HttpQueryServiceConfiguration에서 HTTP 서비스가 바인딩하는 IP(인터넷 프로토콜) 주소 목록을 반환하는 데 사용됩니다.
HTTP_SERVICE_CONFIG_SSL_CCS_KEY 포트가 TLS(전송 계층 보안) 핸드셰이크를 수신하는 경우 인증서를 찾기 위해 CCS(중앙 집중식 인증서 저장소) 저장소를 참조하도록 Http.sys 지정하는 SSL 인증서 레코드를 식별하는 키 역할을 합니다.
HTTP_SERVICE_CONFIG_SSL_CCS_QUERY HttpQueryServiceConfiguration 함수를 호출할 때 포트에서 SSL CCS(Centralized Certificate Store) 레코드를 쿼리할 SSL(Secure Sockets Layer) 구성을 지정합니다.
HTTP_SERVICE_CONFIG_SSL_CCS_SET 포트가 TLS(전송 계층 보안) 핸드셰이크를 수신하는 경우 인증서를 찾으려면 Http.sys CCS(중앙 인증서 저장소) 저장소를 참조하도록 지정하는 SSL 인증서 레코드를 나타냅니다.

HTTP_SERVICE_CONFIG_SSL_KEY 지정된 SSL(Secure Sockets Layer) 인증서 레코드가 식별되는 키 역할을 합니다.
HTTP_SERVICE_CONFIG_SSL_PARAM SSL 구성 저장소에서 레코드를 정의합니다.
HTTP_SERVICE_CONFIG_SSL_QUERY SSL 구성 저장소에서 쿼리할 특정 레코드를 지정하는 데 사용됩니다.
HTTP_SERVICE_CONFIG_SSL_SET SSL 저장소에 새 레코드를 추가하거나 기존 레코드를 검색하는 데 사용됩니다.
HTTP_SERVICE_CONFIG_SSL_SNI_KEY SSL SNI 저장소에서 지정된 SSL(Secure Sockets Layer) SNI(서버 이름 표시) 인증서 레코드가 식별되는 키 역할을 합니다.
HTTP_SERVICE_CONFIG_SSL_SNI_QUERY SSL SNI 저장소에서 쿼리할 특정 SSL(Secure Sockets Layer) SNI(서버 이름 표시) 인증서 레코드를 지정하는 데 사용됩니다.
HTTP_SERVICE_CONFIG_SSL_SNI_SET 새 SSL(Secure Sockets Layer) SNI(서버 이름 표시) 인증서 레코드를 SSL SNI 저장소에 추가하거나 기존 레코드를 검색하는 데 사용됩니다.
HTTP_SERVICE_CONFIG_TIMEOUT_SET HTTP Server API 전체 시간 제한 값을 설정하는 데 사용됩니다.
HTTP_SERVICE_CONFIG_URLACL_KEY URL 네임스페이스 예약 저장소에서 특정 예약 레코드를 지정하는 데 사용됩니다.
HTTP_SERVICE_CONFIG_URLACL_PARAM URL 네임스페이스 예약 저장소의 특정 레코드와 연결된 사용 권한을 지정하는 데 사용됩니다.
HTTP_SERVICE_CONFIG_URLACL_QUERY URL 네임스페이스 예약 저장소에서 쿼리할 특정 예약 레코드를 지정하는 데 사용됩니다.

HTTP_SERVICE_CONFIG_URLACL_SET URL 예약 저장소에 새 레코드를 추가하거나 기존 레코드를 검색하는 데 사용됩니다.
HTTP_SSL_CLIENT_CERT_INFO 인증서가 유효한지 여부를 확인하는 데 사용할 수 있는 SSL(Secure Sockets Layer) 클라이언트 인증서에 대한 데이터를 포함합니다.
HTTP_SSL_INFO SSL 핸드셰이크를 통해 가져온 SSL(Secure Sockets Layer)을 사용하는 연결에 대한 데이터를 포함합니다.
HTTP_STATE_INFO 서버 세션 또는 URL 그룹을 사용하거나 사용하지 않도록 설정하는 데 사용됩니다.
HTTP_TIMEOUT_LIMIT_INFO 애플리케이션별 연결 제한 시간을 정의합니다.
HTTP_TRANSPORT_ADDRESS 특정 HTTP 연결에 사용되는 주소(로컬 및 원격)를 지정합니다.
HTTP_UNKNOWN_HEADER 열거형에 이름이 표시되지 않는 HTTP 요청 또는 응답에서 헤더의 이름과 값을 포함합니다.
HTTP_VERSION 요청에 필요한 HTTP 프로토콜의 버전 또는 응답이 제공하는 버전을 정의합니다.
HTTPAPI_VERSION HTTP 서버 API의 버전을 정의합니다.

열거형

HTTP_503_RESPONSE_VERBOSITY HTTP_503_RESPONSE_VERBOSITY 열거형은 503, 서비스를 사용할 수 없음, 오류 응답에 대한 세부 정보 수준을 정의합니다. 이 구조는 요청 큐에서 HttpServer503ResponseProperty를 설정하거나 쿼리할 때 사용해야 합니다.

HTTP_AUTH_STATUS 요청의 인증 상태를 정의합니다.
HTTP_AUTHENTICATION_HARDENING_LEVELS 서버 강화 수준입니다.
HTTP_CACHE_POLICY_TYPE HTTP_CACHE_POLICY_TYPE 열거형 유형은 사용 가능한 캐시 정책을 정의합니다.
HTTP_DATA_CHUNK_TYPE 데이터 청크의 데이터 원본을 정의합니다.
HTTP_DELEGATE_REQUEST_PROPERTY_ID 대리자 요청에 대한 속성 정보를 설명합니다.
HTTP_ENABLED_STATE 요청 큐, 서버 세션 또는 URL 그룹의 상태를 정의합니다.
HTTP_FEATURE_ID HTTP 기능에 대한 식별자를 지정하는 상수를 정의합니다.
HTTP_HEADER_ID HTTP_HEADER_ID 열거형 형식은 HTTP 요청 및 응답에 대해 알려진 헤더를 나열하고 배열 인덱스를 각 헤더와 연결합니다.
HTTP_LOG_DATA_TYPE 로그 데이터의 형식을 식별합니다.
HTTP_LOGGING_ROLLOVER_TYPE 로그 파일 롤오버 형식을 정의합니다.
HTTP_LOGGING_TYPE 수행되는 로깅 유형을 정의합니다.
HTTP_QOS_SETTING_TYPE HTTP_QOS_SETTING_INFO 구조에 포함된 QOS 설정의 형식을 식별합니다.

HTTP_REQUEST_AUTH_TYPE HTTP_REQUEST_AUTH_TYPE 열거형은 HTTP Server API에서 지원하는 인증 유형을 정의합니다. 이 열거형은 HTTP_REQUEST_AUTH_INFO 구조체에서 사용됩니다.
HTTP_REQUEST_INFO_TYPE HTTP_REQUEST_INFO_TYPE 열거형은 HTTP_REQUEST_INFO 구조에 포함된 정보의 형식을 정의합니다. 이 열거형은 HTTP_REQUEST_INFO 구조체에서 사용됩니다.
HTTP_REQUEST_PROPERTY 요청에 대해 HTTP Server API에 의해 구성된 속성을 정의합니다.
HTTP_RESPONSE_INFO_TYPE HTTP_RESPONSE_INFO_TYPE 열거형은 HTTP_RESPONSE_INFO 구조에 포함된 정보의 형식을 정의합니다. 이 열거형은 HTTP_RESPONSE_INFO 구조에 사용됩니다.
HTTP_SERVER_PROPERTY URL 그룹, 서버 세션 또는 요청 큐에서 HTTP Server API에 의해 구성된 속성을 정의합니다.
HTTP_SERVICE_BINDING_TYPE HTTP_SERVICE_BINDING_TYPE 열거형 형식은 서비스 이름에 대한 문자열 형식을 지정합니다.
HTTP_SERVICE_CONFIG_CACHE_KEY HttpSetServiceConfiguration 및 HttpQueryServiceConfiguration 함수에 사용됩니다.
HTTP_SERVICE_CONFIG_ID 서비스 구성 옵션을 정의합니다.
HTTP_SERVICE_CONFIG_QUERY_TYPE HTTP_SERVICE_CONFIG_QUERY_TYPE 열거형 형식은 만들 쿼리의 다양한 형식을 정의합니다. HTTP_SERVICE_CONFIG_SSL_QUERY, HTTP_SERVICE_CONFIG_SSL_CCS_QUERY 및 HTTP_SERVICE_CONFIG_URLACL_QUERY 구조에 사용됩니다.
HTTP_SERVICE_CONFIG_TIMEOUT_KEY HTTP_SERVICE_CONFIG_TIMEOUT_KEY 열거형은 HTTP_SERVICE_CONFIG_TIMEOUT_SET 구조를 통해 쿼리되거나 구성된 타이머 유형을 정의합니다.

HTTP_VERB

HTTP_VERB 열거형 형식은 HTTP_REQUEST 구조체에서 알려진 표준 HTTP 동사를 지정하는 데 사용되는 값을 정의합니다. 이러한 알려진 동사의 대부분은 아래 표시된 대로 RFC 2616 및 RFC 2518에 문서화되어 있습니다.

hwreqchkapi.h 헤더

아티클2023. 06. 02.

> **ⓘ 중요**
>
> 일부 정보는 상업적으로 출시되기 전에 실질적으로 수정될 수 있는 시험판 제품과 관련이 있습니다. Microsoft는 여기에 제공된 정보에 대해 어떠한 명시적이거나 묵시적인 보증도 하지 않습니다.

HWREQCHK(하드웨어 요구 사항 평가기) 라이브러리는 호출자가 하드웨어 디바이스에 대한 정보를 얻고 컴퓨터가 특정 버전의 Windows 11(및 사용 가능한 새 버전의 OS)를 실행할 수 있는지 확인할 수 있는 API 집합입니다. API는 다양한 하드웨어 속성을 쿼리하고 규칙을 평가하여 자격을 확인합니다.

hwreqchkapi.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

EvaluateHardwareRequirement 특정 요구 사항을 평가하고 디바이스가 하드웨어 요구 사항을 충족하는지 여부를 호출자에게 알리는 통과 또는 실패 결과를 반환합니다.
GetHardwareRequirements 모든 제품 유형에 대해 정의된 하드웨어 요구 사항 컬렉션을 반환합니다.
GetHardwareRequirementSystemInfo 지정된 하드웨어 요구 사항을 평가하고 비교하는 데 사용되는 실제 하드웨어 디바이스 시스템 정보를 제공합니다.
GetLatestHardwareRequirement 지정된 제품 유형에 대해 정의된 최신 요구 사항을 반환합니다.

구조체

HWREQCHK_DEVICE_HARDWARE_EVALUATION 단일 하드웨어 요구 사항 검사 결과를 포함합니다.
HWREQCHK_DEVICE_HARDWARE_REQUIREMENT 단일 하드웨어 요구 사항 검사 결과를 포함합니다.
HWREQCHK_DEVICE_HARDWARE_SYSINFO 디바이스 하드웨어에 대한 정보를 제공합니다.

열거형

HWREQCHK_CPU_VENDOR CPU 공급업체를 나타냅니다.
HWREQCHK_PRODUCT_TYPE 제품 유형을 나타냅니다.
HWREQCHK_TARGET_RELEASE 대상 Windows 릴리스를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

iaccess.h 헤더

아티클2023. 08. 24.

이 헤더는 COM(구성 요소 개체 모델)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)

iaccess.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAccessControl 개체의 개체 및 속성에 대한 액세스를 관리할 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

iads.h 헤더

아티클2023. 08. 24.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Active Directory 서비스 인터페이스
- 보안 및 ID

iads.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAD IADs 인터페이스는 ADSI 개체의 기본 개체 기능, 즉 속성 및 메서드를 정의합니다.
IADsAccessControlEntry IADsAccessControlEntry 인터페이스는 디렉터리 클라이언트가 소유 개체의 개별 ACE(액세스 제어 항목)에 액세스하고 조작할 수 있도록 하는 이중 인터페이스입니다.
IADsAccessControlList IADsAccessControlList 인터페이스는 개별 ACE(액세스 제어 항목)를 관리하는 이중 인터페이스입니다.
IADsAcl IADsAcl 인터페이스는 ADSI 클라이언트가 ACL 또는 상속된 ACL 특성 값에 액세스하고 조작할 수 있는 메서드를 제공합니다. 이 인터페이스는 특성을 조작합니다.
IADsADSystemInfo IADsADSystemInfo 인터페이스는 Windows 도메인에서 Windows 운영 체제를 실행하는 경우 로컬 컴퓨터에 대한 데이터를 검색합니다. 예를 들어 로컬 컴퓨터의 도메인, 사이트 및 고유 이름을 가져올 수 있습니다.
IADsBackLink IADsBackLink 인터페이스는 ADSI 클라이언트가 Back Link 특성에 액세스하는 메서드를 제공합니다. 이 인터페이스의 속성 메서드를 호출하여 특성을 가져오고 수정할 수 있습니다.
IADsCaseIgnoreList

IADsCaseIgnoreList 인터페이스는 ADSI 클라이언트가 사례 무시 목록 특성에 액세스하는 메서드를 제공합니다. 이 인터페이스의 속성 메서드를 호출하여 특성을 가져오고 수정할 수 있습니다.
IADsClass IADsClass 인터페이스는 ADSI 개체에 대한 클래스 정의를 제공하는 스키마 클래스 개체를 관리하도록 설계되었습니다. 다른 스키마 관리 인터페이스에는 특성 정의에 대한 IADsProperty 및 특성 구문에 대한 IADsSyntax가 포함됩니다.
IADsCollection IADsCollection 인터페이스는 호스팅 ADSI 개체가 디렉터리 서비스에 대해 임의의 명명된 데이터 요소 집합을 정의하고 관리할 수 있게 해주는 이중 인터페이스입니다.
IADsComputer IADsComputer 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다.
IADsComputerOperations IADsComputerOperations 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다.
IADsContainer IADsContainer 인터페이스를 사용하면 ADSI 컨테이너 개체가 포함된 ADSI 개체를 만들고 삭제하고 관리할 수 있습니다. 컨테이너 개체는 파일 시스템의 와 같은 계층적 디렉터리 트리를 나타내고 디렉터리 계층 구조를 구성합니다.
IADsDeleteOps IADsDeleteOps 인터페이스는 개체가 기본 디렉터리에서 자신을 삭제하는 데 사용할 수 있는 메서드를 지정합니다. 컨테이너 개체의 경우 메서드는 자식 및 전체 하위 트리를 삭제합니다.
IADsDNWithBinary IADsDNWithBinary 인터페이스는 ADSI 클라이언트가 DN(고유 이름)을 개체의 GUID와 연결하는 메서드를 제공합니다.
IADsDNWithString IADsDNWithString 인터페이스는 ADSI 클라이언트가 DN(고유 이름)을 문자열 값에 연결하는 메서드를 제공합니다.
IADsDomain IADsDomain 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다.
IADsEmail

IADsEmail 인터페이스는 ADSI 클라이언트가 Email Address 특성에 액세스하는 메서드를 제공합니다.
IADsExtension IADsExtension 인터페이스는 ADSI 애플리케이션 확장 모델의 기초를 형성합니다.
IADsFaxNumber IADsFaxNumber 인터페이스는 ADSI 클라이언트가 Facsimile 전화 번호 특성에 액세스하는 메서드를 제공합니다.
IADsFileService IADsFileService 인터페이스는 IADsService에서 상속되는 이중 인터페이스입니다.
IADsFileServiceOperations IADsFileServiceOperations 인터페이스는 IADsServiceOperations에서 상속되는 이중 인터페이스입니다.
IADsFileShare IADsFileShare 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다. 네트워크를 통해 게시된 파일 공유를 나타내기 위해 설계되었습니다. IADsFileShare에서 메서드를 호출하여 파일 공유 지점에 대한 데이터에 액세스하거나 게시합니다.
IADsGroup 디렉터리 서비스에서 그룹 멤버 자격 데이터를 관리합니다.
IADsHold IADsHold 인터페이스는 ADSI 클라이언트가 Hold 특성에 액세스하는 메서드를 제공합니다.
IADsLargeInteger LargeInteger 형식의 64비트 정수 조작에 사용됩니다.
IADsLocality IADsLocality 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다.
IADsMembers IADsMembers 인터페이스는 이중 인터페이스입니다.
IADsNamespaces

IADsNamespaces 인터페이스는 ADs 공급자에 의해 구현되며 네임스페이스 개체를 관리하는 데 사용됩니다.
IADsNameTranslate IADsNameTranslateinterface는 ADS_NAME_TYPE_ENUM 열거형에 정의된 다양한 형식 간에 DN(고유 이름)을 변환합니다. 이 기능은 Active Directory의 개체에서 사용할 수 있습니다.
IADsNetAddress IADsNetAddress 인터페이스는 ADSI 클라이언트가 Net Address 특성에 액세스하는 메서드를 제공합니다.
IADSO IADSO 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다.
IADsObjectOptions ADSI 개체를 조작하기 위한 공급자별 옵션을 지정하고 가져오는 직접 메커니즘을 제공합니다.
IADsOctetList IADsOctetList 인터페이스는 ADSI 클라이언트가 Octet List 특성에 액세스하는 메서드를 제공합니다.
IADsOpenDSObject IADsOpenDSObject 인터페이스는 기본 디렉터리 저장소의 개체에 바인딩하기 위한 보안 컨텍스트를 제공하도록 설계되었습니다.
IADsOU organizationalUnit 개체를 관리하는 데 사용됩니다.
IADsPath IADsPath 인터페이스는 ADSI 클라이언트가 Path 특성에 액세스하는 메서드를 제공합니다.
IADsPathname ADSI에서 X.500 및 Windows 경로를 구문 분석합니다.
IADsPostalAddress IADsPostalAddress 인터페이스는 ADSI 클라이언트가 우편 주소 특성에 액세스하는 메서드를 제공합니다.
IADsPrintJob

IADsPrintJob 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다.
IADsPrintJobOperations IADsPrintJobOperations 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다.
IADsPrintQueue IADsPrintQueue 인터페이스는 네트워크의 프린터를 나타냅니다.
IADsPrintQueueOperations 네트워크를 통해 프린터를 제어하는 데 사용됩니다.
IADsProperty IADsProperty 인터페이스는 스키마 클래스 개체에 대한 단일 특성 정의를 관리하도록 설계되었습니다.
IADsPropertyEntry IADsPropertyEntry 인터페이스는 속성 캐시의 속성 항목을 관리하는 데 사용됩니다.
IADsPropertyList IADsPropertyList 인터페이스는 개체의 속성 캐시에 있는 속성 항목 목록을 수정, 읽기 및 업데이트하는 데 사용됩니다.
IADsPropertyValue 미리 정의된 데이터 형식에서 IADsPropertyEntry 개체의 값을 나타내는 데 사용됩니다.
IADsPropertyValue2 모든 데이터 형식으로 IADsPropertyEntry 개체의 값을 나타내는 데 사용됩니다.
IADsReplicaPointer IADsReplicaPointer 인터페이스는 ADSI 클라이언트가 복제본 포인터 특성에 액세스하는 메서드를 제공합니다.
IADsResource IADsResource 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다. 네트워크를 통해 파일 서비스에 대한 열린 리소스를 관리하도록 설계되었습니다.
IADsSecurityDescriptor ADSI 보안 설명자 개체의 속성에 대한 액세스를 제공합니다.

IADsSecurityUtility IADsSecurityUtility 인터페이스는 파일, 파일 공유 또는 레지스트리 키에서 보안 설명자를 가져오기, 설정 또는 검색하는 데 사용됩니다.
IADsService IADsService 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다.
IADsServiceOperations IADsServiceOperations 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다.
IADsSession IADsSession 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다. 네트워크를 통해 파일 서비스에 대한 활성 세션을 나타내도록 설계되었습니다.
IADsSyntax IADsSyntax 인터페이스는 해당 데이터를 나타내는 데 사용되는 사용 가능한 Automation 데이터 형식을 식별하고 수정하는 메서드를 지정합니다.
IADsTimestamp IADsTimestamp 인터페이스는 ADSI 클라이언트가 타임스탬프 특성에 액세스하는 메서드를 제공합니다.
IADsTypedName IADsTypedName 인터페이스는 ADSI 클라이언트가 Typed Name 특성에 액세스할 수 있는 메서드를 제공합니다.
IADsUser IADsUser 인터페이스는 IAD에서 상속되는 이중 인터페이스입니다.
IADsWinNTSystemInfo IADsWinNTSystemInfo 인터페이스는 컴퓨터에 대한 WinNT 시스템 정보를 검색합니다. 이러한 시스템 정보에는 사용자 계정 이름, 사용자 도메인, 호스트 이름 및 호스트 컴퓨터의 기본 도메인 컨트롤러가 포함됩니다.
IDirectoryObject IDirectoryObject 인터페이스는 클라이언트에 디렉터리 서비스 개체에 대한 직접 액세스를 제공하는 비 Automation COM 인터페이스입니다.

IDirectorySchemaMgmt 현재 구현되지 않으며 사용해서는 안 됩니다.
IDirectorySearch IDirectorySearch 인터페이스는 자동화가 아닌 클라이언트가 기본 디렉터리에서 쿼리를 수행하는 데 사용할 수 있는 낮은 오버헤드 메서드를 제공하는 순수 COM 인터페이스입니다.

구조체

ADS_ATTR_DEF ADS_ATTR_DEF 구조체는 사용되지 않는 인터페이스인 IDirectorySchemaMgmt의 일부로만 사용됩니다.
ADS_ATTR_INFO IDirectoryObject::CreateDSObject, IDirectoryObject::GetObjectAttributes 또는 IDirectoryObject::SetObjectAttributes 메서드와 함께 사용할 하나 이상의 특성 값을 포함하는 데 사용됩니다.
ADS_BACKLINK ADS_BACKLINK 구조체는 Back Link 특성 구문의 ADSI 표현입니다.
ADS_CASEIGNORE_LIST ADS_CASEIGNORE_LIST 구조는 사례 무시 목록 특성 구문의 ADSI 표현입니다.
ADS_CLASS_DEF ADS_CLASS_DEF 구조체는 사용되지 않는 인터페이스인 IDirectorySchemaMgmt의 일부로만 사용됩니다. 다음 정보는 레거시 목적으로만 제공됩니다. ADS_CLASS_DEF 구조체는 개체 클래스의 정의를 보유합니다.
ADS_DN_WITH_BINARY ADSVALUE 구조와 함께 이진 데이터를 포함하는 고유 이름 특성 값을 포함하는 데 사용됩니다.
ADS_DN_WITH_STRING ADSVALUE 구조체와 함께 문자열 데이터를 포함하는 고유 이름 특성 값을 포함하는 데 사용됩니다.

ADS_EMAIL ADS_EMAIL 구조체는 EMail Address 특성 구문의 ADSI 표현입니다.
ADS_FAXNUMBER ADS_FAXNUMBER 구조는 Facsimile 전화 번호 특성 구문의 ADSI 표현입니다.
ADS_HOLD ADS_HOLD 구조체는 Hold 특성 구문의 ADSI 표현입니다.
ADS_NETADDRESS ADS_NETADDRESS 구조체는 Net Address 특성 구문의 ADSI 표현입니다.
ADS_NT_SECURITY_DESCRIPTOR ADS_NT_SECURITY_DESCRIPTOR 구조는 Windows용 보안 설명자의 데이터 형식을 정의합니다.
ADS_OBJECT_INFO ADS_OBJECT_INFO 구조체는 디렉터리 서비스 개체의 ID 및 위치를 포함한 데이터를 지정합니다.
ADS_OCTET_LIST ADS_OCTET_LIST 구조체는 정렬된 단일 바이트 문자열 시퀀스의 ADSI 표현입니다.
ADS_OCTET_STRING ADS_OCTET_STRING 구조체는 Active Directory에서 사용되는 Octet String 특성 구문의 ADSI 표현입니다.
ADS_PATH ADS_PATH 구조체는 Path 특성 구문의 ADSI 표현입니다.
ADS_POSTALADDRESS ADS_POSTALADDRESS 구조는 우편 주소 특성의 ADSI 표현입니다.
ADS_PROV_SPECIFIC ADS_PROV_SPECIFIC 구조에는 BLOB(Binary Large Object)로 표시되는 공급자별 데이터가 포함됩니다.
ADS_REPLICAPOINTER 복제본 포인터 특성 구문의 ADSI 표현을 나타냅니다.

ADS_SEARCH_COLUMN ADS_SEARCH_COLUMN 구조는 디렉터리 서비스 데이터베이스에서 반환된 쿼리에서 검색 열의 내용을 지정합니다.
ADS_SEARCHPREF_INFO ADS_SEARCHPREF_INFO 구조는 쿼리 기본 설정을 지정합니다.
ADS_SORTKEY ADS_SORTKEY 구조체는 쿼리를 정렬하는 방법을 지정합니다.
ADS_TIMESTAMP ADS_TIMESTAMP 구조체는 Timestamp 특성 구문의 ADSI 표현입니다.
ADS_TYPEDNAME Typed Name 특성 구문의 ADSI 표현을 나타냅니다.
ADS_VLV VLV(가상 목록 보기) 검색을 수행하는 데 사용되는 메타데이터를 포함합니다.
ADSVALUE ADSI 데이터 형식으로 지정된 값을 포함합니다.

열거형

ADS_ACEFLAG_ENUM ADS_ACEFLAG_ENUM 열거형은 Active Directory 개체에 대한 ACE(Access Control Entry)의 동작을 지정하는 데 사용됩니다.
ADS_ACETYPE_ENUM Active Directory 개체에 대한 액세스 제어 항목의 형식을 지정하는 데 사용됩니다.
ADS_AUTHENTICATION_ENUM 디렉터리 서비스 개체에 바인딩하기 위해 ADSI에서 사용되는 인증 옵션을 지정합니다.

ADS_CHASE_REFERRALS_ENUM ADS_CHASE_REFERRALS_ENUM 열거형은 조회 추적이 발생하는지, 어떻게 하는지를 지정합니다.
ADS_DEREFENUM ADS_DEREFENUM 열거형은 별칭이 역참조되는 프로세스를 지정합니다.
ADS_DISPLAY_ENUM ADS_DISPLAY_ENUM 열거형은 경로를 표시하는 방법을 지정합니다.
ADS_ESCAPE_MODE_ENUM 디렉터리 경로에 이스케이프 문자가 표시되는 방법을 지정합니다.
ADS_FLAGTYPE_ENUM ADS_FLAGTYPE_ENUM 열거형은 ACE(액세스 제어 항목)에 ObjectType 또는 InheritedObjectType 필드가 있음을 나타내는 데 사용할 수 있는 값을 지정합니다.
ADS_FORMAT_ENUM IADsPathname::Retrieve 메서드에서 사용하는 사용 가능한 경로 값 형식을 지정합니다.
ADS_GROUP_TYPE_ENUM ADSI에서 그룹 개체의 형식을 지정합니다.
ADS_NAME_INITTYPE_ENUM ADS_NAME_INITTYPE_ENUM 열거형은 NameTranslate 개체에서 수행할 초기화 유형을 지정합니다. IADsNameTranslate 인터페이스에서 사용됩니다.
ADS_NAME_TYPE_ENUM 고유 이름을 나타내는 데 사용되는 형식을 지정합니다.
ADS_OPTION_ENUM IADsObjectOptions.GetOption 및 IADsObjectOptions.SetOption 메서드를 사용하여 검색하거나 설정할 수 있는 옵션을 나타내는 값을 포함합니다.
ADS_PASSWORD_ENCODING_ENUM IADsObjectOptions::GetOption 및 IADsObjectOptions::SetOption 메서드의 ADS_OPTION_PASSWORD_METHOD 옵션에 사용되는 암호 인코딩 유형을 식별합니다.

ADS_PATHTYPE_ENUM ADS_PATHTYPE_ENUM 열거형은 IADsSecurityUtility 인터페이스가 보안 설명자를 추가하거나 수정할 개체의 형식을 지정합니다.
ADS_PREFERENCES_ENUM ADS_PREFERENCES_ENUM 열거형은 ADSI용 OLE DB 공급자의 쿼리 기본 설정을 지정합니다.
ADS_PROPERTY_OPERATION_ENUM 캐시에서 명명된 속성을 업데이트하는 방법을 지정합니다.
ADS_RIGHTS_ENUM Active Directory 개체에 할당된 액세스 권한을 지정합니다.
ADS_SCOPEENUM 디렉터리 검색의 scope 지정합니다.
ADS_SD_CONTROL_ENUM ADS_SD_CONTROL_ENUM 열거형은 보안 설명자에 대한 컨트롤 플래그를 지정합니다.
ADS_SD_FORMAT_ENUM ADS_SD_FORMAT_ENUM 열거형은 개체의 보안 설명자가 IADsSecurityUtility 인터페이스에 의해 변환될 형식을 지정합니다.
ADS_SD_REVISION_ENUM Active Directory에 대한 ACE(액세스 제어 항목) 또는 ACL(액세스 제어 목록)의 수정 번호를 지정합니다.
ADS_SEARCHPREF_ENUM IDirectorySearch 개체에 대한 기본 설정을 지정합니다.
ADS_SECURITY_INFO_ENUM 개체의 보안 데이터를 검사하는 데 사용할 수 있는 옵션을 지정합니다.
ADS_SETTYPE_ENUM ADS_SETTYPE_ENUM 열거형은 IADsPathname::Set 메서드에서 사용하는 사용 가능한 경로 이름 형식을 지정합니다.

ADS_STATUSENUM IDirectorySearch::SetSearchPreference 메서드를 사용하여 검색 기본 설정 집합의 상태 지정합니다.
ADS_SYSTEMFLAG_ENUM ADS_SYSTEMFLAG_ENUM 열거형은 systemFlags 특성에 할당할 수 있는 일부 값을 정의합니다. 열거형의 일부 값은 attributeSchema 개체와 관련이 있습니다. 다른 값은 모든 클래스의 개체에 설정할 수 있습니다.
ADS_USER_FLAG_ENUM 디렉터리에서 사용자 속성을 설정하는 데 사용되는 플래그를 정의합니다.
ADSI_DIALECT_ENUM ADSI_DIALECT_ENUM 열거형은 ADSI용 OLE DB 공급자에 사용되는 쿼리 언어를 지정합니다.
ADSTYPEENUM ADSI 속성 값의 데이터 형식을 식별하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

icftypes.h 헤더

아티클2023. 08. 24.

이 헤더는 Windows 방화벽 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 방화벽 기술

icftypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

열거형

NET_FW_ACTION 규칙 또는 기본 설정에 대한 작업을 지정합니다. (NET_FW_ACTION)
NET_FW_AUTHENTICATE_TYPE 트래픽을 허용하기 위해 발생해야 하는 인증 유형을 지정합니다.
NET_FW_EDGE_TRAVERSAL_TYPE 에지 순회 트래픽이 허용되는 조건입니다.
NET_FW_IP_PROTOCOL 인터넷 프로토콜을 지정합니다.
NET_FW_IP_VERSION 포트의 IP 버전을 지정합니다.
NET_FW_MODIFY_STATE 현재 정책에 대한 수정의 효과를 지정합니다.
NET_FW_POLICY_TYPE NET_FW_POLICY_TYPE 열거형 형식은 정책 유형을 지정합니다.
NET_FW_PROFILE_TYPE 프로필 유형을 지정합니다. (NET_FW_PROFILE_TYPE)
NET_FW_PROFILE_TYPE2

프로필 유형을 지정합니다. (NET_FW_PROFILE_TYPE2)
NET_FW_RULE_CATEGORY 방화벽 규칙 범주입니다.
NET_FW_RULE_DIRECTION 규칙이 적용되는 트래픽의 방향을 지정합니다.
NET_FW_SCOPE 포트가 수신 대기할 수 있는 주소의 scope 지정합니다.
NET_FW_SERVICE_TYPE 서비스 유형을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

icm.h 헤더

아티클2023. 03. 09.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DDI 참조 인쇄
- Windows 색 시스템

icm.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AssociateColorProfileWithDeviceA 지정된 색 프로필을 지정된 디바이스와 연결합니다. (ANSI)
AssociateColorProfileWithDeviceW 지정된 색 프로필을 지정된 디바이스와 연결합니다. (유니코드)
CheckBitmapBits 지정된 비트맵의 픽셀이 지정된 변환의 출력 영역에 있는지 여부를 확인합니다.
CheckColors 배열의 색이 지정된 변환의 출력 영역에 있는지 여부를 결정합니다.
CloseColorProfile 열린 프로필 핸들을 닫습니다.
CMCheckColors 지정된 색이 지정된 변환의 출력 영역에 있는지 여부를 결정합니다.
CMCheckColorsInGamut 지정된 RGB 트리플이 지정된 변환의 출력 영역에서 있는지 여부를 결정합니다.
CMCheckRGB 비트맵 색을 출력 영역을 검사합니다.

CMConvertColorNameToIndex 명명된 색 공간의 색 이름을 색 프로필의 인덱스 번호로 변환합니다.
CMConvertIndexToColorName 색 공간의 인덱스를 명명된 색 공간의 이름 배열로 변환합니다. (CMConvertIndexToColorName)
CMCreateDeviceLinkProfile ICC 프로필 형식 사양에서 국제 색 컨소시엄에서 지정한 형식으로 디바이스 링크 프로필을 만듭니다.
CMCreateMultiProfileTransform 프로필 배열 또는 단일 디바이스 링크 프로필을 수락하고 색 변환을 만듭니다. 이 변환은 첫 번째 프로필에 지정된 색 공간에서 두 번째 프로필의 색 공간으로의 매핑입니다.
CMCreateProfile **LOGCOLORSPACEA** 구조체에서 디스플레이 색 프로필을 만듭니다.
CMCreateProfileW **LOGCOLORSPACEW** 구조체에서 디스플레이 색 프로필을 만듭니다.
CMCreateTransform 더 이상 사용되지 않습니다. 이 API는 더 이상 사용되지 않으므로 대체 API가 없습니다. 대체 CMM 모듈의 개발자는 이를 구현할 필요가 없습니다. (CMCreateTransform)
CMCreateTransformExt 변환을 만드는 방법을 정의하는 플래그 집합을 사용하여 입력 **LOGCOLORSPACEA** 에서 선택적 대상 공간으로 매핑한 다음 출력 디바이스로 매핑하는 색 변환을 만듭니다.
CMCreateTransformExtW 변환을 만드는 방법을 정의하는 플래그 집합을 사용하여 입력 **LOGCOLORSPACEW** 에서 선택적 대상 공간으로 매핑한 다음 출력 디바이스로 매핑하는 색 변환을 만듭니다.
CMCreateTransformW 더 이상 사용되지 않습니다. 이 API는 더 이상 사용되지 않으므로 대체 API가 없습니다. 대체 CMM 모듈의 개발자는 이를 구현할 필요가 없습니다. (CMCreateTransformW)
CMDeleteTransform 지정된 색 변환을 삭제하고 연결된 모든 메모리를 해제합니다.

CMGetInfo CMM(색 관리 모듈)에 대한 다양한 정보를 검색합니다.
CMGetNamedProfileInfo 지정된 명명된 색 프로필에 대한 정보를 검색합니다.
CMGetPS2ColorRenderingDictionary CMGetPS2ColorRenderingDictionary 및 이와 연결된 추가 매개 변수를 결정해야 합니다.
CMGetPS2ColorRenderingIntent 프로필에서 PostScript 수준 2 색 렌더링 의도를 검색합니다 .
CMGetPS2ColorSpaceArray CMGetPS2ColorSpaceArray 및 관련 매개 변수, 반환 및 설명이 결정되어야 합니다.
CMIsProfileValid 지정된 프로필이 색 관리에 사용할 수 있는 유효한 ICC 프로필인지 여부를 보고합니다.
CMTranslateColors 색 변환을 사용하여 원본 색 공간에서 대상 색 공간 으로 색 배열을 변환합니다.
CMTranslateRGB 애플리케이션에서 제공하는 RGBQuad를 디바이스 색 공간 으로 변환합니다.
CMTranslateRGB 색 변환을 사용하여 한 색 공간에서 다른 색 공간 으로 비트맵을 변환합니다.
CMTranslateRGBsExt 정의된 한 형식의 비트맵을 다른 정의된 형식으로 변환하고 콜백 함수를 주기적으로 호출하여 지정된 경우 진행률을 보고하고 호출 애플리케이션이 번역을 종료하도록 허용합니다.
ColorProfileAddDisplayAssociation ColorProfileAddDisplayAssociation은 설치된 색 프로필을 지정된 범위의 지정된 디스플레이와 연결합니다.

ColorProfileGetDisplayDefault ColorProfileGetDisplayDefault는 지정된 범위의 지정된 디스플레이에 대한 기본 색 프로필을 가져옵니다.
ColorProfileGetDisplayList ColorProfileGetDisplayList는 지정된 범위에서 지정된 디스플레이와 연결된 프로필 목록을 검색합니다.
ColorProfileGetDisplayUserScope ColorProfileGetDisplayUserScope는 제공된 디스플레이의 현재 선택된 색 프로필 범위(사용자 또는 시스템)를 가져옵니다.
ColorProfileRemoveDisplayAssociation ColorProfileRemoveDisplayAssociation은 지정된 범위의 지정된 디스플레이에서 설치된 색 프로필을 분리합니다.
ColorProfileSetDisplayDefaultAssociation ColorProfileSetDisplayDefaultAssociation은 지정된 범위에서 지정된 디스플레이의 기본 프로필로 설치된 색 프로필을 설정합니다.
ConvertColorNameToIndex 명명된 색 공간의 색 이름을 ICC(International Color Consortium) 색 프로필의 인덱스 번호로 변환합니다.
ConvertIndexToColorName 색 공간의 인덱스를 명명된 색 공간의 이름 배열로 변환합니다. (ConvertIndexToColorName)
CreateColorTransformA 애플리케이션에서 색 관리를 수행하는 데 사용할 수 있는 색 변환을 만듭니다. (ANSI)
CreateColorTransformW 애플리케이션에서 색 관리를 수행하는 데 사용할 수 있는 색 변환을 만듭니다. (유니코드)
CreateDeviceLinkProfile 지정된 의도를 사용하여 색상 프로필 집합에서 ICC(International Color Consortium) *디바이스 링크 프로필을* 만듭니다.

CreateMultiProfileTransform 프로필 배열 또는 단일 디바이스 링크 프로필을 수락하고 애플리케이션이 색 매핑을 수행하는 데 사용할 수 있는 색 변환을 만듭니다.
CreateProfileFromLogColorSpaceA 논리 색 공간을디바이스 프로필로 변환합니다. (ANSI)
CreateProfileFromLogColorSpaceW 논리 색 공간을디바이스 프로필로 변환합니다. (유니코드)
DeleteColorTransform 지정된 색 변환을 삭제합니다.
DisassociateColorProfileFromDeviceA 지정된 색 프로필을 지정된 컴퓨터의 지정된 디바이스와 연결 해제합니다. (ANSI)
DisassociateColorProfileFromDeviceW 지정된 색 프로필을 지정된 컴퓨터의 지정된 디바이스와 연결 해제합니다. (유니코드)
EnumColorProfilesA 지정된 열거형 조건을 충족하는 모든 프로필을 열거합니다. (ANSI)
EnumColorProfilesW 지정된 열거형 조건을 충족하는 모든 프로필을 열거합니다. (유니코드)
GetCMMInfo 지정된 색 변환을 만든 CMM(색 관리 모듈)에 대한 다양한 정보를 검색합니다.
GetColorDirectoryA 지정된 컴퓨터에서 Windows COLOR 디렉터리의 경로를 검색합니다. (ANSI)
GetColorDirectoryW 지정된 컴퓨터에서 Windows COLOR 디렉터리의 경로를 검색합니다. (유니코드)
GetColorProfileElement 지정된 색 프로필의 지정된 태그가 지정된 프로필 요소에서 버퍼로 데이터를 복사합니다.

GetColorProfileElementTag 지정된 ICC(International Color Consortium) 색 프로필의 태그 테이블에서 *dwIndex* 로 지정된 태그 이름을 검색합니다. 여기서 *dwIndex* 는 해당 테이블에 대한 하나의 기반 인덱스입니다.
GetColorProfileFromHandle 열려 있는 색 프로필에 대한 핸들이 제공되면 **GetColorProfileFromHandle** 함수는 프로필의 내용을 애플리케이션에서 제공하는 버퍼에 복사합니다. 핸들이 WCS(Windows Color System) 핸들인 경우 DMP가 반환되고 HPROFILE과 연결된 CAMP 및 GMMP가 무시됩니다.
GetColorProfileHeader ICC 색 프로필 또는 WCS XML 프로필에서 ICC 헤더 구조를 검색하거나 파생합니다. 드라이버와 애플리케이션은 **TRUE** 를 반환한다고 가정해야 하며 올바르게 구조화된 헤더가 반환됨을 나타냅니다. 각 태그는 레거시 ICM2 API 또는 XML 스키마 API를 사용하여 독립적으로 유효성을 검사해야 합니다.
GetCountColorProfileElements 지정된 색 프로필에서 태그가 지정된 요소의 수를 검색합니다.
GetNamedProfileInfo 첫 번째 매개 변수에 지정된 ICC(International Color Consortium) 명명된 색 프로필에 대한 정보를 검색합니다.
GetPS2ColorRenderingDictionary 지정된 ICC 색 프로필에서 PostScript 수준 2 색 렌더링 사전을 검색합니다.
GetPS2ColorRenderingIntent ICC 색 프로필에서 PostScript 수준 2 색 렌더링 의도 를 검색합니다.
GetPS2ColorSpaceArray ICC 색 프로필에서 PostScript 수준 2 색 공간 배열을 검색합니다.
GetStandardColorSpaceProfileA 지정된 표준 색 공간에 대해 등록된 색 프로필을 검색 합니다. (ANSI)
GetStandardColorSpaceProfileW 지정된 표준 색 공간에 대해 등록된 색 프로필을 검색 합니다. (유니코드)

InstallColorProfileA 지정된 컴퓨터에 사용할 지정된 프로필을 설치합니다. 프로필도 COLOR 디렉터리에 복사됩니다. (ANSI)
InstallColorProfileW 지정된 컴퓨터에 사용할 지정된 프로필을 설치합니다. 프로필도 COLOR 디렉터리에 복사됩니다. (유니코드)
IsColorProfileTagPresent 지정된 ICC(International Color Consortium) 태그가 지정된 색 프로필에 있는지 여부를 보고합니다.
IsColorProfileValid 지정된 프로필이 유효한 ICC(International Color Consortium) 프로필인지 아니면 색 관리에 사용할 수 있는 유효한 WCS(Windows Color System) 프로필 핸들인지 확인할 수 있습니다.
OpenColorProfileA 지정된 색 프로필에 대한 핸들을 만듭니다. 그런 다음 핸들을 다른 프로필 관리 함수에서 사용할 수 있습니다. (ANSI)
OpenColorProfileW 지정된 색 프로필에 대한 핸들을 만듭니다. 그런 다음 핸들을 다른 프로필 관리 함수에서 사용할 수 있습니다. (유니코드)
RegisterCMMA 지정된 식별 값을 지정된 색 관리 모듈 CMM DLL(동적 연결 라이브러리)과 연결합니다. 이 ID가 색 프로필에 표시되면 Windows에서 해당 CMM을 찾아 변환을 만들 수 있습니다. (ANSI)
RegisterCMMW 지정된 식별 값을 지정된 색 관리 모듈 CMM DLL(동적 연결 라이브러리)과 연결합니다. 이 ID가 색 프로필에 표시되면 Windows에서 해당 CMM을 찾아 변환을 만들 수 있습니다. (유니코드)
SelectCMM 사용할 기본 설정 CMM(색 관리 모듈)을 선택할 수 있습니다.
SetColorProfileElement ICC 색 프로필에서 태그가 지정된 프로필 요소의 요소 데이터를 설정합니다.

SetColorProfileElementReference 지정된 ICC 색 프로필에서 기존 태그와 동일한 데이터를 참조하는 새 태그를 만듭니다.
SetColorProfileElementSize ICC 색 프로필에서 태그가 지정된 요소의 크기를 설정합니다.
SetColorProfileHeader 지정된 ICC 색 프로필에서 헤더 데이터를 설정합니다.
SetStandardColorSpaceProfileA 지정된 표준 색 공간에 대해 지정된 프로필을 등록합니다. GetStandardColorSpaceProfileW를 사용하여 프로필을 쿼리할 수 있습니다. (ANSI)
SetStandardColorSpaceProfileW 지정된 표준 색 공간에 대해 지정된 프로필을 등록합니다. GetStandardColorSpaceProfileW를 사용하여 프로필을 쿼리할 수 있습니다. (유니코드)
SetupColorMatchingA 사용자가 색 관리를 사용하도록 설정할지 여부를 선택할 수 있는 색 관리 대화 상자를 만들고, 이 경우 사용된 색 프로필과 렌더링 의도를 제어할 수 있습니다. (ANSI)
SetupColorMatchingW 사용자가 색 관리를 사용하도록 설정할지 여부를 선택할 수 있는 색 관리 대화 상자를 만들고, 이 경우 사용된 색 프로필과 렌더링 의도를 제어할 수 있습니다. (유니코드)
TranslateBitmapBits 요청된 형식으로 다른 비트맵을 생성하도록 정의된 형식을 갖는 비트맵의 색을 변환합니다.
TranslateColors 색 변환에 정의된 대로 원본 색 공간에서 대상 색 공간 으로 색 배열을 변환합니다.
UninstallColorProfileA 지정된 컴퓨터에서 지정된 색 프로필을 제거합니다. 연결된 파일은 시스템에서 선택적으로 삭제됩니다. (ANSI)
UninstallColorProfileW 지정된 컴퓨터에서 지정된 색 프로필을 제거합니다. 연결된 파일은 시스템에서 선택적으로 삭제됩니다. (유니코드)

UNREgisterCMMA 지정된 색 관리 모듈 CMM DLL(동적 연결 라이브러리)에서 지정된 ID 값을 분리합니다. (ANSI)
UnregisterCMMW 지정된 색 관리 모듈 CMM DLL(동적 연결 라이브러리)에서 지정된 ID 값을 분리합니다. (유니코드)
WcsAssociateColorProfileWithDevice WcsAssociateColorProfileWithDevice는 지정된 WCS 색 프로필을 지정된 디바이스와 연결합니다.
WcsCheckColors 배열의 색이 지정된 WCS 색 변환의 출력 범위 내에 있는지 여부를 결정합니다.
WcsCreateIccProfile WCS 프로필을 ICC(International Color Consortium) 프로필로 변환합니다.
WcsDisassociateColorProfileFromDevice 컴퓨터의 지정된 디바이스에서 지정된 WCS 색 프로필을 분리합니다.
WcsEnumColorProfiles 지정된 프로필 관리 범위의 열거 조건을 충족하는 모든 색 프로필을 열거합니다.
WcsEnumColorProfilesSize **WcsEnumColorProfiles** 함수가 색 프로필을 열거하는 데 필요한 버퍼의 크기(바이트)를 반환합니다.
WcsGetCalibrationManagementState 디스플레이 보정 상태의 시스템 관리를 사용할 수 있는지 여부를 결정합니다.
WcsGetDefaultColorProfile 디바이스에 대한 기본 색 프로필을 검색하거나 디바이스가 지정되지 않은 경우 디바이스 독립적 기본값을 검색합니다.
WcsGetDefaultColorProfileSize 디바이스의 기본 색 프로필 이름(**NULL** 종결자 포함)의 크기를 바이트 단위로 반환합니다.
WcsGetDefaultRenderingIntent 지정된 프로필 관리 범위에서 기본 렌더링 의도를 검색합니다.

WcsGetUsePerUserProfiles 사용자가 지정된 디바이스에 대해 사용자별 프로필 연결 목록을 사용하도록 선택했는지 여부를 결정합니다.
WcsOpenColorProfileA 지정된 색 프로필에 대한 핸들을 만듭니다. (ANSI)
WcsOpenColorProfileW 지정된 색 프로필에 대한 핸들을 만듭니다. (유니코드)
WcsSetCalibrationManagementState 디스플레이 보정 상태의 시스템 관리를 사용하거나 사용하지 않도록 설정합니다.
WcsSetDefaultColorProfile 지정된 프로필 관리 범위에서 지정된 프로필 형식의 기본 색 프로필 이름을 설정합니다.
WcsSetDefaultRenderingIntent 지정된 프로필 관리 범위에서 기본 렌더링 의도를 설정합니다.
WcsSetUsePerUserProfiles 사용자가 지정된 디바이스에 대해 사용자별 프로필 연결 목록을 사용할지 여부를 지정할 수 있습니다.
WcsTranslateColors 색 변환에 정의된 대로 원본 색 공간에서 대상 색 공간으로 색 배열을 변환합니다.

콜백 함수

PBMCALLBACKFN TBD(PBMCALLBACKFN)
PCMSCALLBACKA *PCMSCALLBACKA*(또는 **ApplyCallbackFunction**)는 SetupColorMatchingW 함수가 표시하는 대화 상자가 실행되는 동안 WCS 구성 데이터를 업데이트하는 콜백 함수입니다.

PCMSCALLBACKW *PCMSCALLBACKW*(또는 **ApplyCallbackFunction**)는 SetupColorMatchingW 함수가 표시하는 대화 상자가 실행되는 동안 WCS 구성 데이터를 업데이트하는 콜백 함수입니다.

구조체

CMYKCOLOR CMYKCOLOR 구조체에 대한 설명입니다.
색 COLOR 공용 구조체에 대한 설명입니다.
COLORMATCHSETUPA **COLORMATCHSETUP** 구조체에는 SetupColorMatchingW 함수가 **ColorManagement** 대화 상자를 초기화하는 데 사용하는 정보가 포함되어 있습니다. (ANSI)
COLORMATCHSETUPW **COLORMATCHSETUP** 구조체에는 SetupColorMatchingW 함수가 **ColorManagement** 대화 상자를 초기화하는 데 사용하는 정보가 포함되어 있습니다. (유니코드)
ENUMTYPEA 프로필 열거형 제약 조건을 정의하는 정보를 포함합니다. (ANSI)
ENUMTYPEW 프로필 열거형 제약 조건을 정의하는 정보를 포함합니다. (유니코드)
GENERIC3CHANNEL TBD(GENERIC3CHANNEL)
GRAYCOLOR GRAYCOLOR 구조체에 대한 설명입니다.
HiFiCOLOR HiFiCOLOR 구조체에 대한 설명입니다.

LabCOLOR TBD(LabCOLOR)
NAMED_PROFILE_INFO **NAMED_PROFILE_INFO** 구조는 명명된 색 프로필에 대한 정보를 저장하는 데 사용됩니다.
NAMEDCOLOR TBD(NAMEDCOLOR)
프로필 색 프로필을 정의하는 정보를 포함합니다.
PROFILEHEADER 디바이스 프로필 파일의 내용을 설명하는 정보를 포함합니다. 이 헤더는 디바이스 프로필 파일의 시작 부분에서 발생합니다.
RGBCOLOR TBD(RGBCOLOR)
XYZCOLOR TBD(XYZCOLOR)
YxyCOLOR TBD(YxyCOLOR)

열거형

BMFORMAT **BMFORMAT** 열거형 형식의 값은 여러 WCS 함수에서 특정 비트맵이 있는 형식을 나타내는 데 사용됩니다.
COLORDATATYPE WCS 함수에서 벡터 콘텐츠의 데이터 형식을 나타내는 데 사용됩니다.

COLORPROFILESUBTYPE 색 프로필의 하위 형식을 지정합니다.
COLORPROFILETYPE 색 프로필의 유형을 지정합니다.
COLORTYPE **COLORTYPE** 열거형의 값은 여러 WCS 함수에서 사용됩니다. **COLOR** 형식의 변수는 **COLORTYPE** 열거형으로 열거된 색 공간에 정의됩니다.
WCS_PROFILE_MANAGEMENT_SCOPE 프로필 관리 작업의 범위를 지정합니다(예: 디바이스와 프로필 연결).

icmpapi.h 헤더

아티클2023. 03. 26.

이 헤더는 IP 도우미에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- IP 도우미

icmpapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

Icmp6CreateFile Icmp6CreateFile 함수는 IPv6 ICMP 에코 요청을 실행할 수 있는 핸들을 엽니다.
Icmp6ParseReplies Icmp6ParseReplies 함수는 제공된 회신 버퍼를 구문 분석하고 IPv6 ICMPv6 에코 응답 응답을 반환합니다(있는 경우).
Icmp6SendEcho2 Icmp6SendEcho2 함수는 IPv6 ICMPv6 에코 요청을 보내고 즉시(Event 또는 ApcRoutine이 NULL이 아닌 경우) 반환하거나 지정된 시간 제한 후에 를 반환합니다. ReplyBuffer에는 IPv6 ICMPv6 에코 응답(있는 경우)이 포함됩니다.
IcmpCloseHandle IcmpCloseHandle 함수는 IcmpCreateFile 또는 Icmp6CreateFile 함수를 호출하여 연 핸들을 닫습니다.
IcmpCreateFile IcmpCreateFile 함수는 IPv4 ICMP 에코 요청을 실행할 수 있는 핸들을 엽니다.
IcmpParseReplies 제공된 회신 버퍼를 구문 분석하고 찾은 ICMP 에코 요청 응답 수를 반환합니다.
IcmpSendEcho IcmpSendEcho 함수는 IPv4 ICMP 에코 요청을 보내고 모든 에코 응답 응답을 반환합니다. 제한 시간이 만료되었거나 회신 버퍼가 채워지면 호출이 반환됩니다.

IcmpSendEcho2 **IcmpSendEcho2** 함수는 IPv4 ICMP 에코 요청을 보내고 즉시(*Event* 또는 *ApcRoutine*이 **NULL**이 아닌 경우) 반환하거나 지정된 시간 제한 후에 를 반환합니다. *ReplyBuffer*에는 ICMP 에코 응답(있는 경우)이 포함됩니다.
IcmpSendEcho2Ex IPv4 ICMP 에코 요청을 보내고 즉시(Event 또는 ApcRoutine이 NULL이 아닌 경우) 반환하거나 지정된 제한 시간 후에 를 반환합니다. ReplyBuffer에는 ICMP 응답(있는 경우)이 포함됩니다.

icodecapi.h 헤더

아티클2023. 08. 24.

이 헤더는 Microsoft Media Foundation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft Media Foundation

icodecapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICodecAPI ICodecAPI 인터페이스는 인코더 또는 디코더 필터에서 설정을 설정 및 검색하고 코덱에서 속성을 설정하기 위한 제네릭 메커니즘을 정의합니다.

구조체

CodecAPIEventData CodecAPIEventData 구조체는 EC_CODECAPI_EVENT 이벤트에 대한 이벤트 데이터를 포함하며 ICodecAPI 인터페이스를 지원하는 코덱에 의해 전송됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

icontact.h 헤더

아티클2023. 08. 24.

이 헤더는 Windows 연락처에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 연락처

icontact.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

Icontact 사용하지 마십시오. 단일 연락처에 대한 속성을 읽고 쓰는 메서드를 정의합니다.
IContactCollection 사용하지 마십시오. IContactManager에서 알려진 연락처를 열거합니다.
IContactManager 사용하지 마십시오. 연락처 ID 문자열을 기반으로 연락처를 검색하는 데 사용됩니다.
IContactProperties 사용하지 마십시오. IContact에서 속성을 검색, 설정, 만들기 및 제거하는 데 사용됩니다. 속성 이름 및 확장 메커니즘은 icontactproperties.h에 설명되어 있습니다.
IContactPropertyCollection 사용하지 마십시오. 레이블 또는 속성 집합을 기반으로 연락처 데이터를 필터링하는 데 사용됩니다. IContactProperties 개체로 노출된 연락처 속성을 열거합니다. 각 속성에 대해 이름, 형식, 버전 및 수정 날짜를 검색할 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

icontentprefetchertasktrigger.h 헤더

아티클2023. 08. 24.

이 헤더는 Windows Web Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Web Services

icontentprefetchertasktrigger.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IContentPrefetcherTaskTrigger IContentPrefetcherTaskTrigger 인터페이스는 설치된 앱 패키지가 이 백그라운드 작업에 대해 등록되었는지 확인하고 콘텐츠 프리페치 작업을 수동으로 트리거할 수 있는 메서드를 정의하여 콘텐츠 프리페치 동작 및 성능 테스트를 지원합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

icwcfg.h 헤더

아티클2023. 08. 24.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

icwcfg.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CheckConnectionWizard CheckConnectionWizard 함수는 ICW(인터넷 연결 마법사)가 설치되어 있고 이전에 실행되지 않은지 확인합니다.
SetShellNext ShellNext 레지스트리 키를 지정된 값으로 설정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

identitycommon.h 헤더

아티클2023. 08. 24.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

identitycommon.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

열거형

IDENTITY_TYPE 열거할 ID의 형식을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

identityprovider.h 헤더

아티클2023. 08. 24.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

identityprovider.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAssociatedIdentityProvider ID 공급자가 ID를 로컬 사용자 계정과 연결할 수 있도록 허용합니다.
IConnectedIdentityProvider 연결된 ID 공급자와의 상호 작용 메서드를 제공합니다.
IIdentityAdvise ID가 업데이트되면 ID 공급자가 호출 애플리케이션에 알릴 수 있습니다.
IIdentityProvider ID 공급자를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

identitystore.h 헤더

아티클2023. 03. 16.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

identitystore.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IIdentityStore ID 및 ID 공급자를 열거하고 관리하는 메서드를 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ifdef.h 헤더

아티클2023. 03. 16.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- IP 도우미
- 네트워크 드라이버, Windows Vista 이상

ifdef.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

IF_COUNTED_STRING_LH IF_COUNTED_STRING 구조체는 NDIS 인터페이스에 대해 계산된 문자열을 지정합니다.
NDIS_INTERFACE_INFORMATION NDIS_INTERFACE_INFORMATION 구조는 OID_GEN_INTERFACE_INFO OID에 대한 네트워크 인터페이스에 대한 정보를 제공합니다.
NET_LUID_LH 네트워크 인터페이스에 대한 LUID(로컬 고유 식별자)입니다.
NET_PHYSICAL_LOCATION_LH NET_PHYSICAL_LOCATION 구조는 등록된 네트워크 인터페이스의 물리적 위치에 대한 정보를 NDIS에 제공합니다.

열거형

IF_OPER_STATUS IF_OPER_STATUS 열거형은 인터페이스의 작동 상태를 지정합니다.
NET_IF_ACCESS_TYPE NET_IF_ACCESS_TYPE 열거형 형식은 NDIS 네트워크 인터페이스 액세스 형식을 지정합니다.

NET_IF_ADMIN_STATUS NET_IF_ADMIN_STATUS 열거형 유형은 RFC 2863에 설명된 대로 NDIS 네트워크 인터페이스 관리 상태를 지정합니다.
NET_IF_CONNECTION_TYPE NET_IF_CONNECTION_TYPE 열거형 형식은 NDIS 네트워크 인터페이스 연결 형식을 지정합니다.
NET_IF_DIRECTION_TYPE NET_IF_ACCESS_TYPE 열거형 형식은 NDIS 네트워크 인터페이스 방향 형식을 지정합니다.
NET_IF_MEDIA_CONNECT_STATE NET_IF_MEDIA_CONNECT_STATE 열거형 형식은 NDIS 네트워크 인터페이스 연결 상태를 지정합니다.
NET_IF_MEDIA_DUPLEX_STATE NET_IF_MEDIA_DUPLEX_STATE 열거형 형식은 NDIS 네트워크 인터페이스 이중 상태를 지정합니다.
NET_IF_OPER_STATUS NET_IF_OPER_STATUS 열거형 유형은 현재 NDIS 네트워크 인터페이스 작동 상태를 정의합니다.
TUNNEL_TYPE TUNNEL_TYPE 열거형 형식은 IANA(Internet Assigned Names Authority)에서 설명한 대로 터널에서 사용하는 캡슐화 방법을 정의합니다.

ifmib.h 헤더

아티클2023. 08. 24.

이 헤더는 관리 정보 베이스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 관리 정보 기반

ifmib.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

MIB_IFNUMBER 특정 컴퓨터의 인터페이스 수를 저장합니다.
MIB_IFROW 특정 인터페이스에 대한 정보를 저장합니다. (MIB_IFROW)
MIB_IFTABLE 인터페이스 항목의 테이블을 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

iketypes.h 헤더

아티클2023. 08. 24.

이 헤더는 Windows 필터링 플랫폼에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 필터링 플랫폼

iketypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

IKEEXT_AUTHENTICATION_METHOD0 IKE/AuthIP 인증에 대한 다양한 매개 변수를 지정합니다.
IKEEXT_AUTHENTICATION_METHOD1 IKE/Authip 인증에 대한 다양한 매개 변수를 지정합니다. (IKEEXT_AUTHENTICATION_METHOD1)
IKEEXT_AUTHENTICATION_METHOD2 IKE/Authip 인증에 대한 다양한 매개 변수를 지정합니다. (IKEEXT_AUTHENTICATION_METHOD2)
IKEEXT_CERT_EKUS0 인증서의 EKU(확장 키 사용) 속성에 대한 정보를 포함합니다.
IKEEXT_CERT_NAME0 인증 방법에 대한 인증서 선택 "주체" 조건을 지정합니다.
IKEEXT_CERT_ROOT_CONFIG0 IKE, AuthIP 또는 IKEv2 인증서 루트 구성을 저장합니다.
IKEEXT_CERTIFICATE_AUTHENTICATION0 인증서를 사용하여 인증을 위한 다양한 매개 변수를 지정하는 데 사용됩니다. (IKEEXT_CERTIFICATE_AUTHENTICATION0)
IKEEXT_CERTIFICATE_AUTHENTICATION1 인증서를 사용하여 인증을 위한 다양한 매개 변수를 지정하는 데 사용됩니다. (IKEEXT_CERTIFICATE_AUTHENTICATION1)

IKEEXT_CERTIFICATE_AUTHENTICATION2 인증서를 사용하여 인증을 위한 다양한 매개 변수를 지정하는 데 사용됩니다. (IKEEXT_CERTIFICATE_AUTHENTICATION2)
IKEEXT_CERTIFICATE_CREDENTIAL0 인증서 인증과 관련된 자격 증명 정보를 저장하는 데 사용됩니다. (IKEEXT_CERTIFICATE_CREDENTIAL0)
IKEEXT_CERTIFICATE_CREDENTIAL1 인증서 인증과 관련된 자격 증명 정보를 저장하는 데 사용됩니다. (IKEEXT_CERTIFICATE_CREDENTIAL1)
IKEEXT_CERTIFICATE_CRITERIA0 인증 방법에 적용할 조건 집합을 포함합니다.
IKEEXT_CIPHER_ALGORITHM0 IKE/AuthIP 암호화 알고리즘에 대한 정보를 저장합니다.
IKEEXT_COMMON_STATISTICS0 IKE 및 인증에 공통된 다양한 통계입니다. (IKEEXT_COMMON_STATISTICS0)
IKEEXT_COMMON_STATISTICS1 IKE, Authip 및 IKEv2에 공통된 다양한 통계입니다.
IKEEXT_COOKIE_PAIR0 IKE/Authip 쿠키 쌍을 저장하는 데 사용됩니다.
IKEEXT_CREDENTIAL_PAIR0 인증에 사용되는 자격 증명 정보를 저장하는 데 사용됩니다. (IKEEXT_CREDENTIAL_PAIR0)
IKEEXT_CREDENTIAL_PAIR1 인증에 사용되는 자격 증명 정보를 저장하는 데 사용됩니다. (IKEEXT_CREDENTIAL_PAIR1)
IKEEXT_CREDENTIAL_PAIR2 인증에 사용되는 자격 증명 정보를 저장하는 데 사용됩니다. (IKEEXT_CREDENTIAL_PAIR2)
IKEEXT_CREDENTIAL0

인증에 사용되는 자격 증명 정보를 저장하는 데 사용됩니다. (IKEEXT_CREDENTIAL0)
IKEEXT_CREDENTIAL1 인증에 사용되는 자격 증명 정보를 저장하는 데 사용됩니다. (IKEEXT_CREDENTIAL1)
IKEEXT_CREDENTIAL2 인증에 사용되는 자격 증명 정보를 저장하는 데 사용됩니다. (IKEEXT_CREDENTIAL2)
IKEEXT_CREDENTIALS0 여러 자격 증명 쌍을 저장하는 데 사용됩니다. (IKEEXT_CREDENTIALS0)
IKEEXT_CREDENTIALS1 여러 자격 증명 쌍을 저장하는 데 사용됩니다. (IKEEXT_CREDENTIALS1)
IKEEXT_CREDENTIALS2 여러 자격 증명 쌍을 저장하는 데 사용됩니다. (IKEEXT_CREDENTIALS2)
IKEEXT_EAP_AUTHENTICATION0 EAP(확장 가능한 인증 프로토콜) 인증에 필요한 정보를 저장합니다.
IKEEXT_EM_POLICY0 AuthIP의 확장 모드 협상 정책을 저장하는 데 사용됩니다. (IKEEXT_EM_POLICY0)
IKEEXT_EM_POLICY1 AuthIP의 확장 모드 협상 정책을 저장하는 데 사용됩니다. (IKEEXT_EM_POLICY1)
IKEEXT_EM_POLICY2 AuthIP의 확장 모드 협상 정책을 저장하는 데 사용됩니다. (IKEEXT_EM_POLICY2)
IKEEXT_INTEGRITY_ALGORITHM0 IKE/AuthIP 해시 알고리즘을 저장합니다.
IKEEXT_IP_VERSION_SPECIFIC_COMMON_STATISTICS0 IKE 및 인증에 공통된 다양한 통계입니다. (IKEEXT_IP_VERSION_SPECIFIC_COMMON_STATISTICS0)
IKEEXT_IP_VERSION_SPECIFIC_COMMON_STATISTICS1 키 지정 모듈(IKE, Authip 및 IKEv2)에 공통된 다양한 통계입니다.

IKEEXT_IP_VERSION_SPECIFIC_KEYMODULE_STATISTICS0 키 지정 모듈 및 IP 버전과 관련된 다양한 통계입니다.
IKEEXT_IP_VERSION_SPECIFIC_KEYMODULE_STATISTICS1 키 지정 모듈(IKE, Authip 및 IKEv2) 및 IP 버전과 관련된 다양한 통계입니다.
IKEEXT_IPV6_CGA_AUTHENTICATION0 IPV6 CGA(암호화로 생성된 주소) 인증에 대한 다양한 매개 변수를 지정하는 데 사용됩니다.
IKEEXT_KERBEROS_AUTHENTICATION0 미리 공유된 키 인증에 필요한 정보를 포함합니다. (IKEEXT_KERBEROS_AUTHENTICATION0)
IKEEXT_KERBEROS_AUTHENTICATION1 미리 공유된 키 인증에 필요한 정보를 포함합니다. (IKEEXT_KERBEROS_AUTHENTICATION1)
IKEEXT_KEYMODULE_STATISTICS0 키 지정 모듈과 관련된 다양한 통계를 포함합니다. (IKEEXT_KEYMODULE_STATISTICS0)
IKEEXT_KEYMODULE_STATISTICS1 키 지정 모듈과 관련된 다양한 통계를 포함합니다. (IKEEXT_KEYMODULE_STATISTICS1)
IKEEXT_NAME_CREDENTIAL0 자격 증명 이름 정보를 저장하는 데 사용됩니다.
IKEEXT_NTLM_V2_AUTHENTICATION0 Microsoft Windows NTLM(NT LAN Manager) V2 인증에 필요한 정보를 포함합니다.
IKEEXT_POLICY0 IKE/AuthIP 기본 모드 협상 정책을 저장하는 데 사용됩니다. (IKEEXT_POLICY0)
IKEEXT_POLICY1 IKE/AuthIP 기본 모드 협상 정책을 저장하는 데 사용됩니다. (IKEEXT_POLICY1)
IKEEXT_POLICY2 IKE/AuthIP 기본 모드 협상 정책을 저장하는 데 사용됩니다. (IKEEXT_POLICY2)
IKEEXT_PRESHARED_KEY_AUTHENTICATION0

미리 공유된 키 인증에 필요한 정보를 저장합니다. (IKEEXT_PRESHARED_KEY_AUTHENTICATION0)
IKEEXT_PRESHARED_KEY_AUTHENTICATION1 미리 공유된 키 인증에 필요한 정보를 저장합니다. (IKEEXT_PRESHARED_KEY_AUTHENTICATION1)
IKEEXT_PROPOSAL0 IKE/AuthIP 기본 모드 제안을 저장하는 데 사용됩니다.
IKEEXT_RESERVED_AUTHENTICATION0 내부용으로 예약된 속성입니다.
IKEEXT_SA_DETAILS0 IKE, AuthIP 또는 IKEv2 SAS(보안 연결)를 열거할 때 반환되는 정보를 저장하는 데 사용됩니다.
IKEEXT_SA_DETAILS1 IKE, AuthIP 및 IKEv2 SAS(보안 연결)를 열거할 때 반환되는 정보를 저장하는 데 사용됩니다. (IKEEXT_SA_DETAILS1)
IKEEXT_SA_DETAILS2 IKE, AuthIP 및 IKEv2 SAS(보안 연결)를 열거할 때 반환되는 정보를 저장하는 데 사용됩니다. (IKEEXT_SA_DETAILS2)
IKEEXT_SA_ENUM_TEMPLATE0 IKE/AuthIP SA(보안 연결)를 열거하는 데 사용되는 열거형 템플릿입니다.
IKEEXT_STATISTICS0 다양한 IKE/AuthIP 통계를 저장합니다.
IKEEXT_STATISTICS1 다양한 IKE, AuthIP 및 IKEv2 통계를 저장합니다.
IKEEXT_TRAFFIC0 IKE/Authip 트래픽을 지정합니다.

열거형

IKEEXT_AUTHENTICATION_IMPERSONATION_TYPE 인증에 인증된 인터넷 프로토콜(AuthIP)이 사용될 때 수행할 가장 유형을 지정합니다.
IKEEXT_AUTHENTICATION_METHOD_TYPE IKE(인터넷 키 교환), 인증된 인터넷 프로토콜(AuthIP) 또는 IKEv2에서 사용하는 인증 방법 유형을 지정합니다.
IKEEXT_CERT_CONFIG_TYPE 인증서 구성 유형을 나타냅니다.
IKEEXT_CERT_CRITERIA_NAME_TYPE 인증서 선택 "주체" 조건에 대해 가능한 NAME 필드의 형식을 지정합니다.
IKEEXT_CIPHER_TYPE IKE(인터넷 키 교환) 및 인증된 인터넷 프로토콜(AuthIP) 메시지를 암호화하는 데 사용되는 암호화 알고리즘 유형을 지정합니다.
IKEEXT_DH_GROUP IKE(인터넷 키 교환) 및 인증된 인터넷 프로토콜(AuthIP) 키 생성에 사용되는 Diffie Hellman 그룹의 유형을 지정합니다.
IKEEXT_EM_SA_STATE 인증된 인터넷 프로토콜(AuthIP) 프로토콜의 일부인 EM(확장 모드) 협상 교환에 대한 상태입니다.
IKEEXT_INTEGRITY_TYPE IKE(인터넷 키 교환) 및 인증된 인터넷 프로토콜(AuthIP) 메시지의 무결성 보호에 사용되는 해시 알고리즘 유형을 지정합니다.
IKEEXT_KEY_MODULE_TYPE 키 지정 모듈의 유형을 지정합니다.
IKEEXT_MM_SA_STATE 인증된 인터넷 프로토콜(AuthIP) 및 IKE(인터넷 키 교환) 프로토콜의 일부인 MM(기본 모드) 협상 교환의 상태입니다.
IKEEXT_QM_SA_STATE 인증된 인터넷 프로토콜(AuthIP) 및 IKE(인터넷 키 교환) 프로토콜의 일부인 QM(빠른 모드) 협상 교환에 대한 상태입니다.

IKEEXT_SA_ROLE IKE(인터넷 키 교환) 및 인증된 인터넷 프로토콜(AuthIP) 기본 모드 또는 빠른 모드 협상에 대한 SA(보안 연결) 역할입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

imagehlp.h 헤더

아티클2023. 08. 24.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

imagehlp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

BindImage 가져온 각 함수의 가상 주소를 계산합니다.
BindImageEx 가져온 각 함수의 가상 주소를 계산합니다.
CheckSumMappedFile 지정된 이미지 파일의 체크섬을 계산합니다.
GetImageConfigInformation 이미지의 부하 구성 데이터를 찾아 반환합니다.
GetImageUnusedHeaderBytes 현재 사용되지 않는 PE 헤더 부분의 오프셋 및 크기를 검색합니다.
ImageAddCertificate 지정된 파일에 인증서를 추가합니다.
ImageEnumerateCertificates 이미지 파일에 현재 포함된 인증서에 대한 정보를 검색합니다.
ImageGetCertificateData 파일에서 전체 인증서를 검색합니다.
ImageGetCertificateHeader

섹션 오프셋 배열까지 지정된 인증서의 헤더를 검색합니다.
ImageGetDigestStream 지정된 이미지 파일에서 요청된 데이터를 검색합니다.
ImageLoad 로드된 DLL 목록을 유지 관리합니다.
ImageRemoveCertificate 지정된 파일에서 지정된 인증서를 제거합니다.
ImageUnload ImageLoad 함수에 대한 이전 호출에서 리소스 할당을 취소합니다.
MapAndLoad 이미지를 매핑하고 매핑된 파일에서 데이터를 미리 로드합니다.
MapFileAndCheckSumA 지정된 파일의 체크섬을 계산합니다. (ANSI)
MapFileAndCheckSumW 지정된 파일의 체크섬을 계산합니다. (유니코드)
ReBaseImage 지정된 이미지의 부하 주소를 변경하여 DLL에 필요한 로드 시간을 줄입니다. (ReBaseImage)
ReBaseImage64 지정된 이미지의 부하 주소를 변경하여 DLL에 필요한 로드 시간을 줄입니다. (ReBaseImage64)
SetImageConfigInformation 이미지의 부하 구성 데이터를 찾아 변경합니다.
SplitSymbols 지정된 이미지에서 기호를 제거합니다.
TouchFileTimes 지정된 파일이 마지막으로 수정된 날짜와 시간을 업데이트.

UnMapAndLoad MapAndLoad 함수에 대한 이전 호출에 의해 할당된 모든 리소스의 할당을 취소합니다.
UpdateDebugInfoFile 지정된 정보를 사용하여 기호 파일의 해당 필드를 업데이트합니다.
UpdateDebugInfoFileEx 지정된 확장 정보를 사용하여 기호 파일의 해당 필드를 업데이트합니다.

콜백 함수

DIGEST_FUNCTION ImageGetDigestStream 함수에서 데이터를 처리하는 데 사용하는 애플리케이션 정의 콜백 함수입니다.
PIMAGEHLP_STATUS_ROUTINE BindImageEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. 상태 루틴은 이미지 바인딩 프로세스 중에 호출됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

imagetranscode.h 헤더

아티클2023. 08. 24.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

imagetranscode.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ITranscodeImage Windows에서 지원하는 모든 이미지 형식에서 JPEG 또는 BMP(비트맵) 이미지 형식으로 변환할 수 있는 메서드를 노출합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

imapi.h 헤더

아티클2023. 08. 24.

이 헤더는 이미지 마스터링 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 이미지 마스터링 API

imapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDiscMaster IDiscMaster 인터페이스를 사용하면 애플리케이션에서 이미지 마스터 API를 예약하고, 이미지 마스터 개체에서 지원하는 디스크 마스터 형식 및 디스크 레코더를 열거하고, 디스크의 시뮬레이션되거나 실제 화상을 시작할 수 있습니다.
IDiscMasterProgressEvents IDiscMasterProgressEvents 인터페이스는 IMAPI에서 애플리케이션으로 만들 수 있는 모든 콜백에 대한 단일 인터페이스를 제공합니다.
IDiscRecorder IDiscRecorder 인터페이스를 사용하면 활성 디스크 레코더로 레이블이 지정된 단일 디스크 레코더 디바이스에 액세스할 수 있습니다. MSDiscMasterObj와 같은 IMAPI 개체는 활성 디스크 레코더를 유지 관리합니다.
IJolietDiscMaster IJolietDiscMaster 인터페이스를 사용하면 CD 데이터 디스크를 스테이징할 수 있습니다.
IRedbookDiscMaster IRedbookDiscMaster 인터페이스를 사용하면 오디오 CD 이미지를 스테이징할 수 있습니다. MSDiscMasterObj에서 지원하는 형식 중 하나를 나타내며 한 번에 트랙 모드(고정 크기 오디오 간격)에서 다중 트랙 오디오 디스크를 만들 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

imapi2.h 헤더

아티클2023. 08. 24.

이 헤더는 이미지 마스터링 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 이미지 마스터링 API

imapi2.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

DDiscFormat2DataEvents 이 인터페이스를 구현하여 현재 쓰기 작업에 대한 알림을 받습니다. (DDiscFormat2DataEvents)
DDiscFormat2EraseEvents 이 인터페이스를 구현하여 현재 지우기 작업의 알림을 받습니다.
DDiscFormat2RawCDEvents 이 인터페이스를 구현하여 현재 원시 이미지 쓰기 작업에 대한 알림을 받습니다.
DDiscFormat2TrackAtOnceEvents 이 인터페이스를 구현하여 현재 트랙 쓰기 작업에 대한 알림을 받습니다.
DDiscMaster2Events 컴퓨터에서 CD 또는 DVD 디바이스가 추가되거나 제거될 때 알림을 받도록 이 인터페이스를 구현합니다.
DWriteEngine2Events 이 인터페이스를 구현하여 현재 쓰기 작업에 대한 알림을 받습니다. (DWriteEngine2Events)
IBlockRange 이 인터페이스를 사용하여 미디어에서 단일 연속 범위의 섹터에 대한 정보를 검색합니다. 이 인터페이스는 일반적으로 섹터 범위의 컬렉션을 설명하기 위해 IBlockRangeList 인터페이스와 함께 사용됩니다.
IBlockRangeList 이 인터페이스를 사용하여 미디어에서 연속 섹터 범위 목록을 검색합니다. 이 인터페이스는 새 논

리 세션이 기록될 때 다시 작성 가능한 디스크에서 업데이트해야 하는 섹터를 설명하는 데 사용됩니다.
IBurnVerification IDiscFormat2Data 또는 IDiscFormat2TrackAtOnce와 함께 이 인터페이스를 사용하여 쓰기 작업 후 불타는 미디어가 무결성을 확인하는 방법을 지시하는 Burn Verification Level 속성을 얻거나 설정합니다.
IDiscFormat2 기본 인터페이스입니다. 이 인터페이스 IDiscFormat2Data, IDiscFormat2Erase, IDiscFormat2TrackAtOnce, IDiscFormat2RawCD를 상속하는 다음 인터페이스를 사용합니다.
IDiscFormat2Data 이 인터페이스를 사용하여 디스크에 데이터 스트림을 작성합니다.
IDiscFormat2DataEventArgs 이 인터페이스를 사용하여 현재 쓰기 작업에 대한 정보를 검색합니다. (IDiscFormat2DataEventArgs)
IDiscFormat2Erase 디스크에서 데이터를 지우려면 이 인터페이스를 사용합니다.
IDiscFormat2RawCD 이 인터페이스를 사용하여 DAO(Disc At Once) 모드(중단 없는 녹화라고도 함)를 사용하여 디스크 디바이스에 원시 이미지를 작성합니다.
IDiscFormat2RawCDEventArgs 이 인터페이스를 사용하여 현재 쓰기 작업에 대한 정보를 검색합니다. (IDiscFormat2RawCDEventArgs)
IDiscFormat2TrackAtOnce 이 인터페이스를 사용하여 한 번에 추적 모드에서 빈 CD-R 또는 CD-RW 미디어에 오디오를 작성합니다.
IDiscFormat2TrackAtOnceEventArgs 이 인터페이스를 사용하여 현재 쓰기 작업에 대한 정보를 검색합니다. (IDiscFormat2TrackAtOnceEventArgs)
IDiscMaster2

이 인터페이스를 사용하여 컴퓨터에 설치된 CD 및 DVD 디바이스를 열거합니다.
IDiscRecorder2 이 인터페이스는 물리적 디바이스를 나타냅니다. 이 인터페이스를 사용하여 컴퓨터에 설치된 CD 및 DVD 장치에 대한 정보를 검색하고 트레이를 닫거나 미디어를 꺼내는 등의 작업을 수행합니다.
IDiscRecorder2Ex 이 인터페이스는 물리적 디바이스를 나타냅니다.
IMultisession 파생 다중 인스턴스 인터페이스에 공통적인 속성을 포함하는 기본 인터페이스입니다.
IMultisessionRandomWrite 이 인터페이스를 사용하여 임의 쓰기를 허용하고 실제 세션의 개념을 지원하지 않는 미디어의 현재 상태에 대한 정보를 검색합니다.
IMultisessionSequential 미디어에 이전 세션이 포함된 경우 이 인터페이스를 사용하여 순차적으로 녹화된 미디어에서 이전 가져오기 세션에 대한 정보를 검색합니다.
IMultisessionSequential2 이 인터페이스를 사용하여 순차적으로 녹화된 미디어에서 쓰기 가능한 단위의 크기에 대한 정보를 검색합니다.
IRawCDImageCreator 이 인터페이스를 사용하여 DAO(Disc-at-Once) 모드에서 CD 미디어에 쓰는 데 사용할 RAW CD 이미지를 만듭니다. 이 인터페이스로 만든 이미지는 IDiscFormat2RawCD 인터페이스를 사용하여 CD 미디어에 쓸 수 있습니다.
IRawCDImageTrackInfo 이 인터페이스를 사용하여 CD 미디어에 적용되는 트랙별 속성을 추적합니다.
IStreamConcatenate 이 인터페이스를 사용하여 여러 데이터 스트림을 단일 스트림으로 결합합니다.
IStreamInterleave 이 인터페이스를 사용하여 각 부분의 일부를 번갈아 산재하여 여러 데이터 스트림을 단일 스트림으로 결합합니다.

IStreamPseudoRandomBased 이 인터페이스를 사용하여 의사 임의 데이터로 데이터가 초기화되는 읽기 전용 데이터 스트림을 생성합니다(암호화로 안전하지 않음). SetSize 메서드를 호출하여 요청된 스트림 크기를 설정해야 합니다.
IWriteEngine2 이 인터페이스를 사용하여 디바이스에 데이터 스트림을 작성합니다.
IWriteEngine2EventArgs 이 인터페이스를 사용하여 현재 쓰기 작업에 대한 정보를 검색합니다. 이 인터페이스는 구현하는 DWriteEngine2Events::Update 메서드에 전달됩니다.
IWriteSpeedDescriptor 이 인터페이스를 사용하여 디스크 레코더 및 현재 미디어에서 지원하는 자세한 쓰기 구성(예: 미디어 유형, 쓰기 속도, 회전 속도 제어 유형)을 검색합니다.

열거형

IMAPI_BURN_VERIFICATION_LEVEL IBurnVerification 인터페이스에서 구현하는 번 확인 값을 정의합니다.
IMAPI_CD_SECTOR_TYPE CD 미디어에 쓸 수 있는 섹터 유형을 정의합니다.
IMAPI_CD_TRACK_DIGITAL_COPY_SETTING 지정된 트랙에 사용할 수 있는 디지털 복사 설정 값을 정의합니다.
IMAPI_FEATURE_PAGE_TYPE 논리 단위(CD 및 DVD 장치)에서 지원하는 기능에 대한 값을 정의합니다.
IMAPI_FORMAT2_DATA_MEDIA_STATE 가능한 미디어 상태에 대한 값을 정의합니다.
IMAPI_FORMAT2_DATA_WRITE_ACTION

IDiscFormat2DataEventArgs 인터페이스를 사용할 때 쓰기 작업의 현재 상태를 나타내는 값을 정의합니다.
IMAPI_FORMAT2_RAW_CD_DATA_SECTOR_TYPE 하위 채널 데이터의 형식을 나타내는 값을 정의합니다.
IMAPI_FORMAT2_RAW_CD_WRITE_ACTION IDiscFormat2RawCDEventArgs 인터페이스를 사용할 때 쓰기 작업의 현재 상태를 나타내는 값을 정의합니다.
IMAPI_FORMAT2_TAO_WRITE_ACTION IDiscFormat2TrackAtOnceEventArgs 인터페이스를 사용할 때 쓰기 작업의 현재 상태를 나타내는 값을 정의합니다.
IMAPI_MEDIA_PHYSICAL_TYPE IMAPI에서 지원하는 현재 알려진 미디어 형식에 대한 값을 정의합니다.
IMAPI_MEDIA_WRITE_PROTECT_STATE 미디어 쓰기 보호 상태 나타내는 값을 정의합니다. 지정된 드라이브에서 하나 이상의 쓰기 보호 값을 설정할 수 있습니다.
IMAPI_MODE_PAGE_REQUEST_TYPE MODE_SENSE10 MMC 명령을 사용하여 디바이스로 전송된 요청을 나타내는 값을 정의합니다.
IMAPI_MODE_PAGE_TYPE CD 및 DVD 디바이스에서 지원하는 모드 페이지의 값을 정의합니다.
IMAPI_PROFILE_TYPE CD 및 DVD 디바이스의 가능한 프로필에 대한 값을 정의합니다. 프로필은 디바이스에서 지원하는 미디어 및 기능의 유형을 정의합니다.
IMAPI_READ_TRACK_ADDRESS_TYPE 임의로 쓰기 가능한 하드웨어 결함 관리 미디어 유형의 현재 디스크 프로필에 대한 추적 주소를 해석하는 방법을 나타내는 값을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

imapi2fs.h 헤더

아티클2023. 08. 24.

이 헤더는 이미지 마스터링 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 이미지 마스터링 API

imapi2fs.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

DFileSystemImageEvents 이 인터페이스를 구현하여 현재 쓰기 작업에 대한 알림을 받습니다. (DFileSystemImageEvents)
DFileSystemImageImportEvents 이 인터페이스를 사용하여 현재 파일 시스템 가져오기 작업에 대한 알림을 받습니다.
IBootOptions 이 인터페이스를 사용하여 광학 디스크에 추가할 부팅 이미지를 지정합니다. 부팅 이미지에는 컴퓨터를 시작하는 데 사용되는 하나 이상의 코드 섹터가 포함되어 있습니다.
IEnumFsiItems FsiDirectoryItem 개체에 대한 자식 디렉터리 및 파일 항목을 열거하려면 이 인터페이스를 사용합니다.
IEnumProgressItems 이 인터페이스를 사용하여 진행률 항목 컬렉션을 열거합니다.
IFileSystemImage 이 인터페이스를 사용하여 파일 시스템 이미지를 빌드하고, 세션 매개 변수를 설정하고, 이미지를 가져오거나 내보냅니다.
IFileSystemImage2 이 인터페이스를 사용하여 EFI/UEFI 지원에 필요한 여러 부팅 항목 또는 부팅 이미지를 작성합니다. 예를 들어 Windows XP 및 Windows Vista 모두에 대한 부팅 스트랩이 있는 부팅 미디어입니다.
IFileSystemImage3

이 인터페이스를 사용하여 UDF 파일 시스템(rev 2.50 이상)의 메타데이터 및 메타데이터 미러 파일을 설정하거나 검사 중복성을 확인합니다.
IFileSystemImageResult 이 인터페이스를 사용하여 굽기 이미지, 이미지 데이터 스트림 및 진행률 정보에 대한 정보를 가져옵니다.
IFileSystemImageResult2 IFileSystemImageResult2 인터페이스를 사용하면 데이터 레코더 개체가 다시 쓰기 가능한 디스크에 대해 만든 이미지에서 수정된 블록에 대한 정보를 검색할 수 있습니다.
IFsiDirectoryItem 이 인터페이스를 사용하여 파일 시스템 이미지에서 항목을 추가하거나 항목을 제거합니다.
IFsiDirectoryItem2 이 인터페이스를 사용하여 모든 하위 디렉터리, 파일 및 연결된 명명된 스트림을 포함하는 디렉터리 트리를 파일 시스템 이미지에 추가합니다.
IFsiFileItem 이 인터페이스를 사용하여 파일 콘텐츠의 파일 크기 및 데이터 스트림을 식별합니다.
IFsiFileItem2 이 인터페이스를 사용하여 파일과 연결된 명명된 스트림을 추가, 제거 및 열거할 수 있습니다. 또한 이 인터페이스는 파일의 '실시간' 특성에 대한 액세스 권한을 제공합니다.
IFsiItem 파일 및 디렉터리 항목 모두에 공통적인 속성을 포함하는 기본 인터페이스입니다.
IFsiNamedStreams 이 인터페이스를 사용하여 파일 시스템 이미지의 파일과 연결된 명명된 스트림을 열거합니다.
IIsoImageManager 이 인터페이스를 사용하여 기존 .iso 파일에 굽기 위한 유효한 파일 시스템이 포함되어 있는지 확인합니다.
IProgressItem 이 인터페이스를 사용하여 결과 파일 이미지의 한 세그먼트에 대한 블록 정보를 검색합니다.
IProgressItems

이 인터페이스를 사용하여 결과 이미지의 진행률 항목을 열거합니다.

열거형

EmulationType 부팅 이미지가 에뮬레이트할 미디어 형식에 대한 값을 정의합니다.
FsiFileSystems 인식된 파일 시스템의 값을 정의합니다.
FsiItemType IFileSystemImage::Exists 메서드를 사용하여 찾은 파일 시스템 항목의 값을 정의합니다.
PlatformId 부팅 이미지가 지원하는 운영 체제 아키텍처의 값을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ime.h 헤더

아티클2023. 08. 24.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

ime.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

SendIMEMessageExA 지정된 하위 함수를 통해 IME(입력 메서드 편집기)에 대한 작업 또는 처리를 지정합니다. (ANSI)
SendIMEMessageExW 지정된 하위 함수를 통해 IME(입력 메서드 편집기)에 대한 작업 또는 처리를 지정합니다. (유니코드)

구조체

IMESTRUCT SendIMEMessageEx에서 IME(입력 메서드 편집기) 메시지 및 해당 매개 변수에서 실행할 하위 기능을 지정하는 데 사용됩니다. 이 구조체는 해당 하위 함수에서 반환 값을 수신하는 데도 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

imepad.h 헤더

아티클2023. 03. 11.

이 헤더는 Windows 애플리케이션에 대한 Internationalization에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 애플리케이션에 대한 국제화

imepad.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IImePad IImePad 인터페이스는 IImePadApplet 인터페이스를 구현하는 IMEPadApplet의 앱에 텍스트를 삽입합니다.
IImePadApplet IImePadApplet 인터페이스는 IImePad 인터페이스를 통해 문자열을 앱에 입력합니다.
IImeSpecifyApplets IImeSpecifyApplets 인터페이스는 IImePadApplet 인터페이스를 에뮬레이트하기 위해 IImePad 인터페이스 개체에서 호출된 메서드를 지정합니다.

구조체

APPLETIDLIST IImePadApplet IID 목록을 지정합니다.
IMEAPPLETCFG IImePad에서 애플릿 구성을 지정하고 설정하는 데 사용됩니다.
IMEAPPLETUI IImePadApplet::CreateUI에서 애플릿 창 스타일을 지정하는 데 사용됩니다.

IMECOMPOSITIONSTRINGINFO 앱에서 IME의 컴퍼지션 문자열에 대한 정보를 포함합니다.

imessagedispatcher.h 헤더

아티클2023. 08. 24.

이 헤더는 COM(구성 요소 개체 모델)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)

imessagedispatcher.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IMessageDispatcher ASTA 스레드에서 창 메시지를 특수하게 처리해야 하는 구성 요소에 의해 구현된 콜백 인터페이스입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

imm.h 헤더

아티클2023. 08. 24.

이 헤더는 Windows 애플리케이션에 대한 국제화에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 애플리케이션에 대한 국제화

imm.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

ImmAssociateContext ImmAssociateContext 함수(imm.h)는 지정된 입력 컨텍스트를 지정된 창에 연결합니다.
ImmAssociateContextEx ImmAssociateContextEx 함수(imm.h)는 입력 메서드 컨텍스트와 지정된 창 또는 해당 자식 간의 연결을 변경합니다.
ImmConfigureIMEA ImmConfigureIMEA(ANSI) 함수(imm.h)는 지정된 입력 로캘 식별자의 IME에 대한 구성 대화 상자를 표시합니다.
ImmConfigureIMEW ImmConfigureIMEW(유니코드) 함수(imm.h)는 지정된 입력 로캘 식별자의 IME에 대한 구성 대화 상자를 표시합니다.
ImmCreateContext ImmCreateContext 함수(imm.h)는 컨텍스트에 대한 메모리를 할당하고 초기화하는 새 입력 컨텍스트를 만듭니다.
ImmDestroyContext ImmDestroyContext 함수(imm.h)는 입력 컨텍스트를 해제하고 연결된 메모리를 해제합니다.
ImmDisableIME ImmDisableIME 함수(imm.h)는 스레드 또는 프로세스의 모든 스레드에 대해 IME를 사용하지 않도록 설정합니다.

ImmDisableLegacyIME ImmDisableLegacyIME 함수(imm.h)는 이 스레드가 Windows 스토어 앱 UI 스레드임을 나타냅니다.
ImmDisableTextFrameService Windows Vista를 기준으로 ImmDisableTextFrameService 함수(imm.h)를 더 이상 사용할 수 없습니다.
ImmEnumInputContext ImmEnumInputContext 함수(imm.h)는 지정된 스레드에 대한 입력 컨텍스트를 검색합니다.
ImmEnumRegisterWordA ANSI(ImmEnumRegisterWordA) 함수(imm.h)는 지정된 읽기 문자열, 스타일 및 레지스터 문자열을 갖는 레지스터 문자열을 열거합니다.
ImmEnumRegisterWordW ImmEnumRegisterWordW(유니코드) 함수(imm.h)는 지정된 읽기 문자열, 스타일 및 레지스터 문자열을 갖는 레지스터 문자열을 열거합니다.
ImmEscapeA ANSI(ImmEscapeA) 함수(imm.h)는 다른 IME API 함수를 통해 사용할 수 없는 특정 IME의 기능에 액세스합니다.
ImmEscapeW ImmEscapeW(유니코드) 함수(imm.h)는 다른 IME API 함수를 통해 사용할 수 없는 특정 IME의 기능에 액세스합니다.
ImmGetCandidateListA ImmGetCandidateListA(ANSI) 함수(imm.h)는 후보 목록을 검색합니다.
ImmGetCandidateListCountA ImmGetCandidateListCountA(ANSI) 함수(imm.h)는 후보 목록의 크기를 검색합니다.
ImmGetCandidateListCountW ImmGetCandidateListCountW(유니코드) 함수(imm.h)는 후보 목록의 크기를 검색합니다.
ImmGetCandidateListW ImmGetCandidateListW(유니코드) 함수(imm.h)는 후보 목록을 검색합니다.

ImmGetCandidateWindow ImmGetCandidateWindow 함수(imm.h)는 후보 창에 대한 정보를 검색합니다.
ImmGetCompositionFontA ImmGetCompositionFontA(ANSI) 함수(imm.h)는 컴퍼지션 창에 문자를 표시하는 데 사용되는 논리 글꼴에 대한 정보를 검색합니다.
ImmGetCompositionFontW ImmGetCompositionFontW(유니코드) 함수(imm.h)는 컴퍼지션 창에 문자를 표시하는 데 사용되는 논리적 글꼴에 대한 정보를 검색합니다.
ImmGetCompositionStringA ImmGetCompositionStringA(ANSI) 함수(imm.h)는 컴퍼지션 문자열에 대한 정보를 검색합니다.
ImmGetCompositionStringW ImmGetCompositionStringW(유니코드) 함수(imm.h)는 컴퍼지션 문자열에 대한 정보를 검색합니다.
ImmGetCompositionWindow ImmGetCompositionWindow 함수(imm.h)는 컴퍼지션 창에 대한 정보를 검색합니다.
ImmGetContext ImmGetContext 함수(imm.h)는 지정된 창과 연결된 입력 컨텍스트를 반환합니다.
ImmGetConversionListA ANSI(ImmGetConversionListA) 함수(imm.h)는 IME 관련 메시지를 생성하지 않고 문자 또는 단어의 변환 결과 목록을 검색합니다.
ImmGetConversionListW ImmGetConversionListW(유니코드) 함수(imm.h)는 IME 관련 메시지를 생성하지 않고 문자 또는 단어의 변환 결과 목록을 검색합니다.
ImmGetConversionStatus ImmGetConversionStatus 함수(imm.h)는 현재 변환 상태 검색합니다.
ImmGetDefaultIMEWnd ImmGetDefaultIMEWnd 함수(imm.h)는 IME 클래스에 대한 기본 창 핸들을 검색합니다.

ImmGetDescriptionA ImmGetDescriptionA(ANSI) 함수(imm.h)는 IME에 대한 설명을 지정된 버퍼에 복사합니다.
ImmGetDescriptionW ImmGetDescriptionW(유니코드) 함수(imm.h)는 IME에 대한 설명을 지정된 버퍼에 복사합니다.
ImmGetGuideLineA ImmGetGuideLineA(ANSI) 함수(imm.h)는 오류에 대한 정보를 검색합니다. 애플리케이션은 사용자 알림에 정보를 사용합니다.
ImmGetGuideLineW ImmGetGuideLineW(유니코드) 함수(imm.h)는 오류에 대한 정보를 검색합니다. 애플리케이션은 사용자 알림에 정보를 사용합니다.
ImmGetIMEFileNameA ANSI(ImmGetIMEFileNameA) 함수(imm.h)는 지정된 입력 로캘과 연결된 IME의 파일 이름을 검색합니다.
ImmGetIMEFileNameW ImmGetIMEFileNameW(유니코드) 함수(imm.h)는 지정된 입력 로캘과 연결된 IME의 파일 이름을 검색합니다.
ImmGetImeMenuItemsA ANSI(ImmGetImeMenuItemsA) 함수(imm.h)는 지정된 입력 컨텍스트의 IME 메뉴에 등록된 메뉴 항목을 검색합니다.
ImmGetImeMenuItemsW ImmGetImeMenuItemsW(유니코드) 함수(imm.h)는 지정된 입력 컨텍스트의 IME 메뉴에 등록된 메뉴 항목을 검색합니다.
ImmGetOpenStatus ImmGetOpenStatus 함수(imm.h)는 IME가 열려 있는지 닫혔는지 여부를 결정합니다.
ImmGetProperty ImmGetProperty 함수(imm.h)는 지정된 입력 로캘과 연결된 IME의 속성 및 기능을 검색합니다.
ImmGetRegisterWordStyleA

ImmGetRegisterWordStyleA(ANSI) 함수(imm.h)는 지정된 입력 로캘과 연결된 IME에서 지원하는 스타일 목록을 검색합니다.
ImmGetRegisterWordStyleW ImmGetRegisterWordStyleW(유니코드) 함수(imm.h)는 지정된 입력 로캘과 연결된 IME에서 지원하는 스타일 목록을 검색합니다.
ImmGetStatusWindowPos ImmGetStatusWindowPos 함수(imm.h)는 상태 창의 위치를 검색합니다.
ImmGetVirtualKey ImmGetVirtualKey 함수(imm.h)는 IME가 이미 처리한 키 입력 메시지와 연결된 원래 가상 키 값을 검색합니다.
ImmInstallIMEA ImmInstallIMEA(ANSI) 함수(imm.h)는 IME를 설치합니다.
ImmInstallIMEW ImmInstallIMEW(유니코드) 함수(imm.h)는 IME를 설치합니다.
ImmIsIME ImmIsIME 함수(imm.h)는 지정된 입력 로캘에 IME가 있는지 여부를 결정합니다.
ImmIsUIMessageA ImmIsUIMessageA(ANSI) 함수(imm.h)는 IME 창에 대한 메시지를 확인하고 해당 메시지를 창으로 보냅니다.
ImmIsUIMessageW ImmIsUIMessageW(유니코드) 함수(imm.h)는 IME 창에 대한 메시지를 확인하고 해당 메시지를 창으로 보냅니다.
ImmNotifyIME ImmNotifyIME 함수(imm.h)는 IME에 입력 컨텍스트의 상태 변경 내용을 알 수 있습니다.
ImmRegisterWordA ImmRegisterWordA(ANSI) 함수(imm.h)는 지정된 입력 로캘과 연결된 IME 사전으로 문자열을 등록합니다.
ImmRegisterWordW

ImmRegisterWordW(유니코드) 함수(imm.h)는 지정된 입력 로캘과 연결된 IME 사전으로 문자열을 등록합니다.
ImmReleaseContext ImmReleaseContext 함수(imm.h)는 입력 컨텍스트를 해제하고 입력 컨텍스트에 연결된 메모리의 잠금을 해제합니다. 애플리케이션은 ImmGetContext 함수를 호출할 때마다 이 함수를 호출해야 합니다.
ImmSetCandidateWindow ImmSetCandidateWindow 함수(imm.h)는 후보 창에 대한 정보를 설정합니다.
ImmSetCompositionFontA ImmSetCompositionFontA(ANSI) 함수(imm.h)는 컴퍼지션 창에 문자를 표시하는 데 사용할 논리 글꼴을 설정합니다.
ImmSetCompositionFontW ImmSetCompositionFontW(유니코드) 함수(imm.h)는 컴퍼지션 창에 문자를 표시하는 데 사용할 논리 글꼴을 설정합니다.
ImmSetCompositionStringA ANSI(ImmSetCompositionStringA) 함수(imm.h)는 컴퍼지션 및 읽기 문자열의 문자, 특성 및 절을 설정합니다.
ImmSetCompositionStringW ImmSetCompositionStringW(유니코드) 함수(imm.h)는 컴퍼지션 및 읽기 문자열의 문자, 특성 및 절을 설정합니다.
ImmSetCompositionWindow ImmSetCompositionWindow 함수(imm.h)는 컴퍼지션 창의 위치를 설정합니다.
ImmSetConversionStatus ImmSetConversionStatus 함수(imm.h)는 현재 변환 상태 설정합니다.
ImmSetOpenStatus ImmSetOpenStatus 함수(imm.h)가 IME를 열거나 닫습니다.
ImmSetStatusWindowPos ImmSetStatusWindowPos 함수(imm.h)는 상태 창의 위치를 설정합니다.

ImmSimulateHotKey ImmSimulateHotKey 함수(imm.h)는 지정된 IME 핫 키를 시뮬레이트하여 사용자가 지정된 창에서 핫 키를 누를 때와 동일한 응답을 발생합니다.
ImmUnregisterWordA ANSI(ImmUnregisterWordA) 함수(imm.h)는 지정된 입력 로캘과 연결된 IME 사전에서 레지스터 문자열을 제거합니다.
ImmUnregisterWordW ImmUnregisterWordW(유니코드) 함수(imm.h)는 지정된 입력 로캘과 연결된 IME 사전에서 레지스터 문자열을 제거합니다.

콜백 함수

IMCENUMPROC IMCENUMPROC(imm.h)는 ImmEnumInputContext 함수에서 제공하는 입력 컨텍스트를 처리하는 애플리케이션 정의 콜백 함수입니다.
REGISTERWORDENUMPROCA ANSI(REGISTERWORDENUMPROCA)(imm.h)는 ImmEnumRegisterWord 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
REGISTERWORDENUMPROCW REGISTERWORDENUMPROCW(유니코드)(imm.h)는 ImmEnumRegisterWord 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.

구조체

CANDIDATEFORM CANDIDATEFORM(imm.h) 구조체에는 후보 창에 대한 위치 정보가 포함됩니다.
후보 목록

CANDIDATELIST(imm.h) 구조에는 후보 목록에 대한 정보가 포함됩니다.
COMPOSITIONFORM COMPOSITIONFORM(imm.h) 구조체에는 컴퍼지션 창에 대한 스타일 및 위치 정보가 포함됩니다.
IMECHARPOSITION IMECHARPOSITION(imm.h) 구조체에는 컴퍼지션 창의 문자 위치에 대한 정보가 포함됩니다.
IMEMENUITEMINFOA IMEMENUITEMINFOA(ANSI) 구조체(imm.h)에는 IME 메뉴 항목에 대한 정보가 포함되어 있습니다.
IMEMENUITEMINFOW IMEMENUITEMINFOW(유니코드) 구조체(imm.h)에는 IME 메뉴 항목에 대한 정보가 포함되어 있습니다.
RECONVERTSTRING RECONVERTSTRING(imm.h) 구조체는 IME 재구성을 위한 문자열을 정의합니다. 다시 변환할 문자열을 포함하는 메모리 블록의 첫 번째 항목입니다.
REGISTERWORDA ANSI(REGISTERWORDA) 구조체(imm.h)에는 읽기 정보 또는 등록할 단어가 포함됩니다.
REGISTERWORDW REGISTERWORDW(유니코드) 구조체(imm.h)에는 읽기 정보 또는 등록할 단어가 포함됩니다.
STYLEBUFA ANSI(STYLEBUFA) 구조체(imm.h)에는 스타일의 식별자와 이름이 포함됩니다.
STYLEBUFW STYLEBUFW(유니코드) 구조체(imm.h)에는 스타일의 식별자와 이름이 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

immdev.h 헤더

아티클2023. 08. 24.

이 헤더는 Windows 애플리케이션에 대한 국제화에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 애플리케이션에 대한 국제화

immdev.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

ImmAssociateContext ImmAssociateContext 함수(immdev.h)는 지정된 입력 컨텍스트를 지정된 창에 연결합니다.
ImmAssociateContextEx ImmAssociateContextEx 함수(immdev.h)는 입력 메서드 컨텍스트와 지정된 창 또는 해당 자식 간의 연결을 변경합니다.
ImmConfigureIMEA ImmConfigureIMEA(ANSI) 함수(immdev.h)는 지정된 입력 로캘 식별자의 IME에 대한 구성 대화 상자를 표시합니다.
ImmConfigureIMEW ImmConfigureIMEW(유니코드) 함수(immdev.h)는 지정된 입력 로캘 식별자의 IME에 대한 구성 대화 상자를 표시합니다.
ImmCreateContext ImmCreateContext 함수(immdev.h)는 컨텍스트에 대한 메모리를 할당하고 초기화하는 새 입력 컨텍스트를 만듭니다.
ImmDestroyContext ImmDestroyContext 함수(immdev.h)는 입력 컨텍스트를 해제하고 연결된 메모리를 해제합니다.
ImmDisableIME ImmDisableIME 함수(immdev.h)는 스레드 또는 프로세스의 모든 스레드에 대해 IME를 사용하지 않도록 설정합니다.

ImmDisableLegacyIME ImmDisableLegacyIME 함수(immdev.h)는 이 스레드가 Windows 스토어 앱 UI 스레드임을 나타냅니다.
ImmDisableTextFrameService ImmDisableTextFrameService 함수(immdev.h)는 Windows Vista를 기준으로 더 이상 사용할 수 없습니다.
ImmEnumInputContext ImmEnumInputContext 함수(immdev.h)는 지정된 스레드에 대한 입력 컨텍스트를 검색합니다.
ImmEnumRegisterWordA ImmEnumRegisterWordA(ANSI) 함수(immdev.h)는 지정된 읽기 문자열, 스타일 및 레지스터 문자열을 갖는 레지스터 문자열을 열거합니다.
ImmEnumRegisterWordW ImmEnumRegisterWordW(유니코드) 함수(immdev.h)는 지정된 읽기 문자열, 스타일 및 레지스터 문자열을 갖는 레지스터 문자열을 열거합니다.
ImmEscapeA ANSI(ImmEscapeA) 함수(immdev.h)는 다른 IME API 함수를 통해 사용할 수 없는 특정 IME의 기능에 액세스합니다.
ImmEscapeW ImmEscapeW(유니코드) 함수(immdev.h)는 다른 IME API 함수를 통해 사용할 수 없는 특정 IME의 기능에 액세스합니다.
ImmGetCandidateListA ImmGetCandidateListA(ANSI) 함수(immdev.h)는 후보 목록을 검색합니다.
ImmGetCandidateListCountA ImmGetCandidateListCountA(ANSI) 함수(immdev.h)는 후보 목록의 크기를 검색합니다.
ImmGetCandidateListCountW ImmGetCandidateListCountW(유니코드) 함수(immdev.h)는 후보 목록의 크기를 검색합니다.
ImmGetCandidateListW ImmGetCandidateListW(유니코드) 함수(immdev.h)는 후보 목록을 검색합니다.

ImmGetCandidateWindow ImmGetCandidateWindow 함수(immdev.h)는 후보 창에 대한 정보를 검색합니다.
ImmGetCompositionFontA ImmGetCompositionFontA(ANSI) 함수(immdev.h)는 컴퍼지션 창에 문자를 표시하는 데 현재 사용되는 논리 글꼴에 대한 정보를 검색합니다.
ImmGetCompositionFontW ImmGetCompositionFontW(유니코드) 함수(immdev.h)는 컴퍼지션 창에 문자를 표시하는 데 현재 사용되는 논리 글꼴에 대한 정보를 검색합니다.
ImmGetCompositionStringA ImmGetCompositionStringA(ANSI) 함수(immdev.h)는 컴퍼지션 문자열에 대한 정보를 검색합니다.
ImmGetCompositionStringW ImmGetCompositionStringW(유니코드) 함수(immdev.h)는 컴퍼지션 문자열에 대한 정보를 검색합니다.
ImmGetCompositionWindow ImmGetCompositionWindow 함수(immdev.h)는 컴퍼지션 창에 대한 정보를 검색합니다.
ImmGetContext ImmGetContext 함수(immdev.h)는 지정된 창과 연결된 입력 컨텍스트를 반환합니다.
ImmGetConversionListA ANSI(ImmGetConversionListA) 함수(immdev.h)는 IME 관련 메시지를 생성하지 않고 문자 또는 단어의 변환 결과 목록을 검색합니다.
ImmGetConversionListW ImmGetConversionListW(유니코드) 함수(immdev.h)는 IME 관련 메시지를 생성하지 않고 문자 또는 단어의 변환 결과 목록을 검색합니다.
ImmGetConversionStatus ImmGetConversionStatus 함수(immdev.h)는 현재 변환 상태 검색합니다.
ImmGetDefaultIMEWnd ImmGetDefaultIMEWnd 함수(immdev.h)는 IME 클래스에 대한 기본 창 핸들을 검색합니다.

ImmGetDescriptionA ImmGetDescriptionA(ANSI) 함수(immdev.h)는 IME에 대한 설명을 지정된 버퍼에 복사합니다.
ImmGetDescriptionW ImmGetDescriptionW(유니코드) 함수(immdev.h)는 IME에 대한 설명을 지정된 버퍼에 복사합니다.
ImmGetGuideLineA ImmGetGuideLineA(ANSI) 함수(immdev.h)는 오류에 대한 정보를 검색합니다. 애플리케이션은 사용자 알림에 정보를 사용합니다.
ImmGetGuideLineW ImmGetGuideLineW(유니코드) 함수(immdev.h)는 오류에 대한 정보를 검색합니다. 애플리케이션은 사용자 알림에 정보를 사용합니다.
ImmGetIMEFileNameA ANSI(ImmGetIMEFileNameA) 함수(immdev.h)는 지정된 입력 로캘과 연결된 IME의 파일 이름을 검색합니다.
ImmGetIMEFileNameW ImmGetIMEFileNameW(유니코드) 함수(immdev.h)는 지정된 입력 로캘과 연결된 IME의 파일 이름을 검색합니다.
ImmGetImeMenuItemsA ANSI(ImmGetImeMenuItemsA) 함수(immdev.h)는 지정된 입력 컨텍스트의 IME 메뉴에 등록된 메뉴 항목을 검색합니다.
ImmGetImeMenuItemsW ImmGetImeMenuItemsW(유니코드) 함수(immdev.h)는 지정된 입력 컨텍스트의 IME 메뉴에 등록된 메뉴 항목을 검색합니다.
ImmGetOpenStatus ImmGetOpenStatus 함수(immdev.h)는 IME가 열려 있는지 닫혔는지 여부를 결정합니다.
ImmGetProperty ImmGetProperty 함수(immdev.h)는 지정된 입력 로캘과 연결된 IME의 속성과 기능을 검색합니다.
ImmGetRegisterWordStyleA

ImmGetRegisterWordStyleA(ANSI) 함수(immdev.h)는 지정된 입력 로캘과 연결된 IME에서 지원하는 스타일 목록을 검색합니다.
ImmGetRegisterWordStyleW ImmGetRegisterWordStyleW(유니코드) 함수(immdev.h)는 지정된 입력 로캘과 연결된 IME에서 지원하는 스타일 목록을 검색합니다.
ImmGetStatusWindowPos ImmGetStatusWindowPos 함수(immdev.h)는 상태 창의 위치를 검색합니다.
ImmGetVirtualKey ImmGetVirtualKey 함수(immdev.h)는 IME가 이미 처리한 키 입력 메시지와 연결된 원래 가상 키 값을 검색합니다.
ImmInstallIMEA ImmInstallIMEA(ANSI) 함수(immdev.h)는 IME를 설치합니다.
ImmInstallIMEW ImmInstallIMEW(유니코드) 함수(immdev.h)는 IME를 설치합니다.
ImmIsIME ImmIsIME 함수(immdev.h)는 지정된 입력 로캘에 IME가 있는지 여부를 결정합니다.
ImmIsUIMessageA ImmIsUIMessageA(ANSI) 함수(immdev.h)는 IME 창에 대한 메시지를 확인하고 해당 메시지를 창으로 보냅니다.
ImmIsUIMessageW ImmIsUIMessageW(유니코드) 함수(immdev.h)는 IME 창에 대한 메시지를 확인하고 해당 메시지를 창으로 보냅니다.
ImmNotifyIME ImmNotifyIME 함수(immdev.h)는 IME에 입력 컨텍스트의 상태 변경 내용을 알 수 있습니다.
ImmRegisterWordA ANSI(ImmRegisterWordA) 함수(immdev.h)는 지정된 입력 로캘과 연결된 IME 사전을 사용하여 문자열을 등록합니다.
ImmRegisterWordW

ImmRegisterWordW(유니코드) 함수(immdev.h)는 지정된 입력 로캘과 연결된 IME 사전을 사용하여 문자열을 등록합니다.
ImmReleaseContext ImmReleaseContext 함수(immdev.h)는 입력 컨텍스트를 해제하고 입력 컨텍스트에 연결된 메모리의 잠금을 해제합니다.
ImmRequestMessageA WM_IME_REQUEST 메시지를 생성합니다. (ANSI)
ImmRequestMessageW WM_IME_REQUEST 메시지를 생성합니다. (유니코드)
ImmSetCandidateWindow ImmSetCandidateWindow 함수(immdev.h)는 후보 창에 대한 정보를 설정합니다.
ImmSetCompositionFontA ANSI(ImmSetCompositionFontA) 함수(immdev.h)는 컴퍼지션 창에 문자를 표시하는 데 사용할 논리 글꼴을 설정합니다.
ImmSetCompositionFontW ImmSetCompositionFontW(유니코드) 함수(immdev.h)는 컴퍼지션 창에 문자를 표시하는 데 사용할 논리 글꼴을 설정합니다.
ImmSetCompositionStringA ANSI(ImmSetCompositionStringA) 함수(immdev.h)는 컴퍼지션 및 읽기 문자열의 문자, 특성 및 절을 설정합니다.
ImmSetCompositionStringW ImmSetCompositionStringW(유니코드) 함수(immdev.h)는 컴퍼지션 및 읽기 문자열의 문자, 특성 및 절을 설정합니다.
ImmSetCompositionWindow ImmSetCompositionWindow 함수(immdev.h)는 컴퍼지션 창의 위치를 설정합니다.
ImmSetConversionStatus ImmSetConversionStatus 함수(immdev.h)는 현재 변환 상태 설정합니다.
ImmSetOpenStatus

ImmSetOpenStatus 함수(immdev.h)가 IME를 열거나 닫습니다.
ImmSetStatusWindowPos ImmSetStatusWindowPos 함수(immdev.h)는 상태 창의 위치를 설정합니다.
ImmSimulateHotKey ImmSimulateHotKey 함수(immdev.h)는 지정된 IME 핫 키를 시뮬레이트하여 사용자가 지정된 창에서 핫 키를 누를 때와 동일한 응답을 발생합니다.
ImmUnregisterWordA ANSI(ImmUnregisterWordA) 함수(immdev.h)는 지정된 입력 로캘과 연결된 IME 사전에서 레지스터 문자열을 제거합니다.
ImmUnregisterWordW ImmUnregisterWordW(유니코드) 함수(immdev.h)는 지정된 입력 로캘과 연결된 IME 사전에서 레지스터 문자열을 제거합니다.

콜백 함수

IMCENUMPROC IMCENUMPROC(immdev.h)는 ImmEnumInputContext 함수에서 제공하는 입력 컨텍스트를 처리하는 애플리케이션 정의 콜백 함수입니다.
REGISTERWORDENUMPROCA ANSI(REGISTERWORDENUMPROCA)는 ImmEnumRegisterWord 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
REGISTERWORDENUMPROCW REGISTERWORDENUMPROCW(유니코드)는 ImmEnumRegisterWord 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (registerwordenumprocw)

구조체

CANDIDATEFORM CANDIDATEFORM 구조체(immdev.h)에는 후보 창에 대한 위치 정보가 포함됩니다.
CANDIDATELIST CANDIDATELIST 구조(immdev.h)에는 후보 목록에 대한 정보가 포함됩니다.
COMPOSITIONFORM COMPOSITIONFORM 구조체(immdev.h)에는 컴퍼지션 창에 대한 스타일 및 위치 정보가 포함되어 있습니다.
IMECHARPOSITION IMECHARPOSITION 구조체(immdev.h)에는 컴퍼지션 창의 문자 위치에 대한 정보가 포함됩니다.
IMEMENUITEMINFOA IMEMENUITEMINFOA(ANSI) 구조체(immdev.h)에는 IME 메뉴 항목에 대한 정보가 포함되어 있습니다.
IMEMENUITEMINFOW IMEMENUITEMINFOW(유니코드) 구조체(immdev.h)에는 IME 메뉴 항목에 대한 정보가 포함되어 있습니다.
RECONVERTSTRING RECONVERTSTRING 구조체(immdev.h)는 IME 재구성을 위한 문자열을 정의합니다.
REGISTERWORDA REGISTERWORDA(ANSI) 구조체(immdev.h)에는 등록할 정보 또는 단어를 읽습니다.
REGISTERWORDW REGISTERWORDW(유니코드) 구조체(immdev.h)에는 등록할 정보 또는 단어를 읽습니다.
STYLEBUFA ANSI(STYLEBUFA) 구조체(immdev.h)에는 스타일의 식별자와 이름이 포함됩니다.
STYLEBUFW STYLEBUFW(유니코드) 구조체(immdev.h)에는 스타일의 식별자와 이름이 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

in6addr.h 헤더

아티클2023. 08. 24.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Vista 이전의 네트워크 드라이버
- 네트워크 드라이버, Windows Vista 이상
- Windows 소켓 2

in6addr.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

IN6_ADDR IN6_ADDR 구조체는 IPv6 전송 주소를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

inaddr.h 헤더

아티클2023. 03. 16.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- IP 도우미
- 네트워크 드라이버, Windows Vista 이상
- Windows 소켓 2

inaddr.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

IN_ADDR in_addr 구조체는 IPv4 주소를 나타냅니다.

indexsrv.h 헤더

아티클2023. 08. 24.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 인덱싱 서비스
- Windows 검색

indexsrv.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IColumnMapper 파일 기반 쿼리에 대한 속성 정보를 검색합니다.
IColumnMapperCreator 열 매퍼 개체를 검색하는 메서드를 포함합니다.
IPhraseSink 쿼리 시간 동안 단어 분리기가 쿼리 텍스트에서 구문 분석하는 구를 처리합니다.
ISimpleCommandCreator 파일 카탈로그와 상호 작용하기 위한 메서드를 포함합니다.
IStemmer 언어별 형태소 분석기를 만드는 메서드를 제공합니다. 형태소 분석기는 지정된 단어의 굴절된 형태를 생성합니다.
IWordBreaker 텍스트를 구문 분석하고 개별 단어와 구를 식별합니다. 이 인터페이스는 언어별 언어 리소스 구성 요소입니다. 백그라운드 프로세스에서 사용되며 처리량과 최소한의 리소스 사용 모두에 최적화되어야 합니다.
IWordFormSink 쿼리 시간 동안 형태소 분석기가 생성하는 대체 단어 양식 목록을 처리합니다.

구조체

TEXT_SOURCE 단어 분리기가 처리할 텍스트에 대한 정보를 포함합니다.

열거형

WORDREP_BREAK_TYPE 현재 단어를 이전 단어와 구분하는 중단 유형을 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

infocard.h 헤더

아티클2023. 03. 09.

이 헤더는 QOS(서비스 품질)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 서비스 품질(QOS)

infocard.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

POLICY_ELEMENT POLICY_ELEMENT(infocard.h) 구조체에는 RSVP 정책 요소가 포함되어 있습니다. (POLICY_ELEMENT)

infotech.h 헤더

아티클2023. 08. 24.

이 헤더는 Microsoft HTML 도움말 1.4에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft HTML 도움말 1.4

infotech.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IITDatabase 데이터베이스 개체를 열고 닫고 데이터베이스에 저장된 개체를 인스턴스화하는 데 이 인터페이스를 사용합니다.
IITPropList 이 인터페이스를 사용하여 단어 휠 및 인덱스와 같은 빌드 개체에 대한 속성을 설정합니다. 문서 빌드 프로세스에서 이러한 메서드를 호출하여 모든 빌드 개체에 대한 속성을 정의합니다.
IITResultSet 런타임 애플리케이션에서 이 인터페이스를 사용하여 결과 집합에 대한 정보를 초기화, 빌드 및 가져옵니다.
IITWordWheel 이 인터페이스를 사용하여 단어 휠 키워드(keyword) 조회를 수행합니다. Lookup 메서드는 검색을 수행하는 여러 가지 방법을 제공합니다._it 지정된 접두사를 기반으로 정확한 일치 또는 가장 가까운 근사값을 반환합니다.
IStemmerConfig 이 인터페이스를 사용하여 형태소 분석을 제어하는 구성 정보를 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

inked.h 헤더

아티클2023. 03. 23.

이 헤더는 태블릿 PC에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Tablet PC

inked.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

_IInkEditEvents
IInkEdit . (IInkEdit)

Functions

_IInkEditEvents_AddRef IInkEditEvents 인터페이스 개체의 참조 수를 1씩 늘입니다. 이 메서드는 IInkEditEvents에서 상속된 **IUnknown** 인터페이스의 일부입니다.
_IInkEditEvents_GetIDsOfNames 이름 집합을 해당 DISPID 집합에 매핑합니다.
_IInkEditEvents_GetTypeInfo 개체에 대한 형식 정보를 검색한 다음 인터페이스에 대한 형식 정보를 가져오는 데 사용할 수 있습니다.
_IInkEditEvents_GetTypeInfoCount 개체가 제공하는 형식 정보 인터페이스의 수를 검색합니다.
_IInkEditEvents_Invoke 개체에 의해 노출되는 속성 및 메서드에 대한 액세스를 제공합니다.

_IInkEditEvents_QueryInterface IInkEditEvents 개체의 지정된 COM 인터페이스에 대한 쿼리입니다.
_IInkEditEvents_Release IInkEditEvents 개체에 대한 참조를 해제합니다.

구조체

IEC_GESTUREINFO 특정 제스처에 대한 정보를 포함합니다.
IEC_RECOGNITIONRESULTINFO IInkRecognitionResult 인터페이스 개체에 대한 정보를 포함합니다.
IEC_STROKEINFO Stroke 이벤트에 대한 정보를 포함합니다.

열거형

AppearanceConstants InkEdit 컨트롤이 화면에 표시되는 방식을 지정합니다.
BorderStyleConstants InkEdit 컨트롤의 테두리가 화면에 표시되는 방식을 지정합니다.
DISPID_InkEdit 다양한 잉크 편집기 상태를 지정합니다.
DISPID_InkEditEvents 다양한 잉크 편집기 이벤트를 지정합니다.

InkDisplayMode 컨트롤에 선택 영역이 표시되는 방식을 지정합니다.
InkEditStatus InkEdit 컨트롤이 유휴 상태인지, 잉크를 수집하거나, 잉크를 인식하는지를 지정합니다.
InkInsertMode InkEdit 컨트롤에 잉크를 삽입하는 방법을 지정합니다.
InkMode 잉크 수집을 사용하지 않도록 설정하거나, 잉크를 수집하거나, 잉크 및 제스처를 수집하는지 여부에 관계없이 잉크를 그리는 컬렉션 모드를 지정합니다.
MouseButton 누른 마우스 단추를 지정합니다. (MouseButton)
ScrollBarsConstants InkEdit 컨트롤의 스크롤 막대가 화면에 표시되는 방식을 지정합니다.
SelAlignmentConstants InkEdit 컨트롤의 여백을 기준으로 단락의 맞춤을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

inkpresenterdesktop.h 헤더

아티클2023. 08. 24.

이 헤더는 잉크 입력에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 잉크 입력

inkpresenterdesktop.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IInkCommitRequestHandler IInkCommitRequestHandler 개체를 사용하면 앱(IInkPresenterDesktop 개체 대신)이 보류 중인 모든 Microsoft DirectComposition 명령을 앱의 DirectComposition 시각적 트리에 커밋할 수 있습니다.
IInkDesktopHost IInkDesktopHost 개체를 사용하면 앱 스레드 만들기를 통해 잉크 입력, 처리 및 렌더링을 통해 IInkPresenterDesktop 개체를 호스트하고 앱의 DirectComposition 시각적 트리에 삽입할 수 있습니다.
IInkHostWorkItem IInkHostWorkItem 개체는 IInkDesktopHost 개체 스레드에서 실행할 잉크 작업을 나타냅니다.
IInkPresenterDesktop IInkPresenterDesktop 개체는 클래식 Windows 앱의 DirectComposition 시각적 트리에 구성하고 삽입할 수 있는 InkPresenter를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

inkrenderer.h 헤더

아티클2023. 03. 23.

이 헤더는 잉크 입력에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 잉크 입력

inkrenderer.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IInkD2DRenderer IInkD2DRenderer 개체를 사용하면 기본 InkCanvas 컨트롤 대신 유니버설 Windows 앱의 지정된 Direct2D 디바이스 컨텍스트에 잉크 스트로크를 렌더링할 수 있습니다.
IInkD2DRenderer2 IInkD2DRenderer2 개체를 사용하면 기본 InkCanvas 컨트롤 대신 유니버설 Windows 앱의 지정된 Direct2D 디바이스 컨텍스트에 잉크 스트로크를 렌더링할 수 있습니다.

열거형

INK_HIGH_CONTRAST_ADJUSTMENT 시스템이 대비 테마 모드에 있을 때 IInkD2DRenderer2 개체가 잉크(표준 및 수정)를 그리는 방법을 지정합니다.

inputpaneinterop.h 헤더

아티클2023. 08. 24.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

inputpaneinterop.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IInputPaneInterop 데스크톱 앱에서 InputPane 클래스의 멤버에 액세스할 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

inputpanelconfiguration.h 헤더

아티클2023. 08. 24.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

inputpanelconfiguration.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IInputPanelConfiguration 데스크톱 앱이 Windows 스토어 앱에서 사용되는 포커스 추적 메커니즘을 옵트인하는 기능을 제공합니다.
IInputPanelInvocationConfiguration Windows 스토어 앱에서 자동 호출 동작을 옵트아웃할 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

inputscope.h 헤더

아티클2023. 08. 24.

이 헤더는 Text Services Framework에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 텍스트 서비스 프레임워크 비교

inputscope.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ITfInputScope ITfInputScope 인터페이스는 텍스트 입력 프로세서에서 창과 연결된 문서 컨텍스트를 나타내는 InputScope 값을 가져오는 데 사용됩니다.
ITfInputScope2 ITfInputScope2 인터페이스는 텍스트 입력 프로세서에서 IEnumString 인터페이스 포인터를 가져오는 데 사용되며 이 IEnumString 인터페이스는 애플리케이션이 이 컨텍스트에 대해 지정한 단어 목록을 열거합니다.

Functions

SetInputScope 지정된 창에 대한 입력 scope 설정합니다.
SetInputScopes 지정된 창에 대한 하나의 입력 scope, 여러 입력 범위, 하나의 구 목록, 정규식 및/또는 SRGS(Speech Recognition Grammar Specification) 규칙의 조합을 설정합니다.
SetInputScopes2 인터페이스의 참조를 지우려면 창이 제거되기 전에 애플리케이션이 IS_DEFAULT SetInputScope를 호출해야 합니다.
SetInputScopeXML 사용하지 마십시오. (SetInputScopeXML)

열거형

InputScope InputScope 열거형에는 지정된 필드에 적용되는 입력 범위를 지정하는 값이 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

inspectable.h 헤더

아티클2023. 03. 09.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

inspectable.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IInspectable 모든 Windows 런타임 클래스에 필요한 기능을 제공합니다.

Functions

HSTRING_UserFree HSTRING_UserFree 함수(inspectable.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HSTRING_UserFree64 HSTRING_UserFree64 함수(inspectable.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HSTRING_UserMarshal HSTRING_UserMarshal 함수(inspectable.h)는 HSTRING 개체를 RPC 버퍼로 마샬링합니다.
HSTRING_UserMarshal64 HSTRING_UserMarshal64 함수(inspectable.h)는 HSTRING 개체를 RPC 버퍼로 마샬링합니다.
HSTRING_UserSize HSTRING_UserSize 함수(inspectable.h)는 HSTRING 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.

HSTRING_UserSize64 HSTRING_UserSize64 함수(inspectable.h)는 HSTRING 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
HSTRING_UserUnmarshal HSTRING_UserUnmarshal 함수(inspectable.h)는 RPC 버퍼에서 HSTRING 개체를 숨기지 않습니다.
HSTRING_UserUnmarshal64 HSTRING_UserUnmarshal64 함수(inspectable.h)는 RPC 버퍼에서 HSTRING 개체를 숨기지 않습니다.

열거형

TrustLevel 활성화 가능한 클래스의 신뢰 수준을 나타냅니다.

instance.h 헤더

아티클2023. 08. 24.

이 헤더는 Windows Management Instrumentation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Management Instrumentation

instance.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

CInstance CInstance 클래스는 WMI 공급자 프레임워크에서 지원하는 인스턴스에 대해 정의된 속성 값을 검색하고 업데이트하는 데 사용됩니다. 또한 CInstance 클래스는 공급자 프레임워크의 CInstance 인터페이스 구현에 대한 액세스 권한을 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

interactioncontext.h 헤더

아티클2023. 08. 24.

이 헤더는 상호 작용 컨텍스트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 상호 작용 컨텍스트

interactioncontext.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AddPointerInteractionContext Interaction Context 개체에서 처리되는 포인터 집합에 지정된 포인터를 포함합니다.
BufferPointerPacketsInteractionContext Interaction Context 개체의 버퍼에 단일 입력 포인터의 기록을 추가합니다.
CreateInteractionContext Interaction Context 개체를 만들고 초기화합니다.
DestroyInteractionContext 지정된 Interaction Context 개체를 삭제합니다.
GetCrossSlideParameterInteractionContext 슬라이드 간 상호 작용 동작을 가져옵니다.
GetInertiaParameterInteractionContext 조작의 관성 동작(변환, 회전, 크기 조정)을 가져옵니다.
GetInteractionConfigurationInteractionContext Interaction Context 개체에 대한 상호 작용 구성 상태를 가져옵니다.
GetMouseWheelParameterInteractionContext Interaction Context 개체의 마우스 휠 상태를 가져옵니다.
GetPropertyInteractionContext

상호 작용 컨텍스트 개체 속성을 가져옵니다.
GetStateInteractionContext 현재 상호 작용 컨텍스트 상태 및 컨텍스트가 유휴 상태로 돌아갈 시간을 가져옵니다.
ProcessBufferedPacketsInteractionContext 포인터 입력 프레임의 끝에서 버퍼링된 패킷을 처리합니다.
ProcessInertiaInteractionContext 관성 처리를 위해 상호 작용 컨텍스트 개체에 타이머 입력을 보냅니다.
ProcessPointerFramesInteractionContext 포인터 입력 프레임 집합을 처리합니다.
RegisterOutputCallbackInteractionContext 상호 작용 컨텍스트 개체에서 상호 작용 이벤트를 수신하도록 콜백을 등록합니다.
RemovePointerInteractionContext Interaction Context 개체에서 처리되는 포인터 집합에서 지정된 포인터를 제거합니다.
ResetInteractionContext 상호 작용 상태, 상호 작용 구성 설정 및 모든 매개 변수를 초기 상태로 다시 설정합니다. 현재 상호 작용은 알림 없이 취소됩니다. 다음에 사용하기 전에 상호 작용 컨텍스트를 다시 구성해야 합니다.
SetCrossSlideParametersInteractionContext 슬라이드 간 상호 작용을 구성합니다.
SetInertiaParameterInteractionContext 접촉이 해제된 후 조작의 관성 동작(변환, 회전, 크기 조정)을 구성합니다.
SetInteractionConfigurationInteractionContext 지정된 조작을 처리하도록 Interaction Context 개체를 구성합니다.
SetMouseWheelParameterInteractionContext 마우스 휠 입력에 대한 매개 변수 값을 설정합니다.
SetPivotInteractionContext 단일 입력 포인터를 사용하여 회전 조작을 위해 중심점 및 중심점의 피벗 반경을 설정합니다.

SetPropertyInteractionContext 상호 작용 컨텍스트 개체 속성을 설정합니다.
StopInteractionContext 상호 작용 상태를 INTERACTION_STATE_IDLE 설정하고 모든 상호 작용 구성 설정 및 매개 변수를 그대로 둡니다.

콜백 함수

INTERACTION_CONTEXT_OUTPUT_CALLBACK Interaction Context 개체에서 이벤트를 수신하는 콜백입니다.

구조체

CROSS_SLIDE_PARAMETER 교차 슬라이드 임계값 및 해당 거리 임계값을 정의합니다.
INTERACTION_ARGUMENTS_CROSS_SLIDE 슬라이드 간 상호 작용의 상태를 정의합니다.
INTERACTION_ARGUMENTS_MANIPULATION 조작의 상태를 정의합니다.
INTERACTION_ARGUMENTS_TAP 탭 상호 작용의 상태를 정의합니다.
INTERACTION_CONTEXT_CONFIGURATION 상호 작용 동작을 사용하거나 사용하지 않도록 설정하거나 수정하는 Interaction Context 개체의 구성을 정의합니다.
INTERACTION_CONTEXT_OUTPUT

Interaction Context 개체의 출력을 정의합니다.
MANIPULATION_TRANSFORM 조작에 대한 변환 데이터를 정의합니다.
MANIPULATION_VELOCITY 조작의 속도 데이터를 정의합니다.

열거형

CROSS_SLIDE_FLAGS 슬라이드 간 상호 작용의 상태를 지정합니다.
CROSS_SLIDE_THRESHOLD 슬라이드 간 동작 임계값을 지정합니다.
INERTIA_PARAMETER 조작에 대한 관성 값(변환, 회전, 크기 조정)을 지정합니다.
INTERACTION_CONFIGURATION_FLAGS Interaction Context 개체를 구성할 때 사용할 상호 작용을 지정합니다.
INTERACTION_CONTEXT_PROPERTY Interaction Context 개체의 속성을 지정합니다.
INTERACTION_FLAGS 상호 작용의 상태를 지정합니다.
INTERACTION_ID Interaction Context 개체를 구성하는 데 사용되는 상호 작용 상태를 지정합니다.
INTERACTION_STATE Interaction Context 개체의 상태를 지정합니다.
MANIPULATION_RAILS_STATE

상호 작용에 대한 레일 상태를 지정합니다.
MOUSE_WHEEL_PARAMETER 마우스 휠 회전에 매핑할 수 있는 조작을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

interlockedapi.h 헤더

아티클2023. 08. 24.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

interlockedapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

InitializeSListHead Singly 연결된 목록의 헤드를 초기화합니다. (InitializeSListHead)
InterlockedFlushSList 암시적으로 연결된 목록에서 모든 항목을 제거합니다. 목록에 대한 액세스는 다중 프로세서 시스템에서 동기화됩니다. (InterlockedFlushSList)
InterlockedPopEntrySList 자동으로 연결된 목록의 맨 앞에서 항목을 제거합니다. 목록에 대한 액세스는 다중 프로세서 시스템에서 동기화됩니다. (InterlockedPopEntrySList)
InterlockedPushEntrySList 자동으로 연결된 목록의 맨 앞에 항목을 삽입합니다. 목록에 대한 액세스는 다중 프로세서 시스템에서 동기화됩니다. (InterlockedPushEntrySList)
InterlockedPushListSListEx 다른 Singly 연결된 목록의 맨 앞에 Singly 연결된 목록을 삽입합니다. 목록에 대한 액세스는 다중 프로세서 시스템에서 동기화됩니다. 이 버전의 메서드는 __fastcall 호출 규칙을 사용하지 않습니다.
QueryDepthSList 지정된 연결 목록의 항목 수를 검색합니다. (QueryDepthSList)

피드백

이 페이지가 도움이 되었나요?

Yes No

Microsoft Q&A에서 도움말 보기

intsafe.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

intsafe.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

ByteToChar BYTE 형식의 값을 CHAR 형식의 값으로 변환합니다.
ByteToInt8 BYTE 형식의 값을 INT8 형식의 값으로 변환합니다.
DWordPtrAdd DWORD_PTR 형식의 두 값을 추가합니다.
DWordPtrMult DWORD_PTR 형식의 한 값을 다른 값으로 곱합니다.
DWordPtrSub DWORD_PTR 형식의 한 값을 다른 값에서 뺍니다.
Int8Add INT8 형식의 두 값을 추가합니다.
Int8Mult INT8 형식의 두 값을 곱합니다.
Int8Sub INT8 형식의 한 값을 다른 값에서 뺍니다.

Int8ToUChar INT8 형식의 값을 UCHAR 형식의 값으로 변환합니다.
Int8ToUInt INT8 형식의 값을 UINT 형식의 값으로 변환합니다.
Int8ToUInt8 INT8 형식의 값을 UINT8 형식의 값으로 변환합니다.
Int8ToUIntPtr INT8 형식의 값을 UINT_PTR 형식의 값으로 변환합니다.
Int8ToULong INT8 형식의 값을 ULONG 형식의 값으로 변환합니다.
Int8ToULongLong INT8 형식의 값을 ULONGLONG 형식의 값으로 변환합니다.
Int8ToULongPtr INT8 형식의 값을 ULONG_PTR 형식 값으로 변환합니다.
Int8ToUShort INT8 형식의 값을 USHORT 형식의 값으로 변환합니다.
IntAdd INT 형식의 두 값을 추가합니다.
IntMult INT 형식의 두 값을 곱합니다.
IntPtrAdd INT_PTR 형식의 두 값을 추가합니다.
IntPtrMult INT_PTR 형식의 두 값을 곱합니다.

IntPtrSub INT_PTR 형식의 한 값을 다른 값에서 뺍니다.
IntPtrToChar 형식 INT_PTR 값을 CHAR 형식의 값으로 변환합니다.
IntPtrToInt 형식 INT_PTR 값을 INT 형식의 값으로 변환합니다.
IntPtrToInt8 INT_PTR 형식의 값을 INT8 형식의 값으로 변환합니다.
IntPtrToLong INT_PTR 형식의 값을 LONG 형식의 값으로 변환합니다.
IntPtrToLongPtr 형식 INT_PTR 값을 LONG_PTR 형식 값으로 변환합니다.
IntPtrToShort INT_PTR 형식의 값을 SHORT 형식의 값으로 변환합니다.
IntPtrToUChar 형식 INT_PTR 값을 UCHAR 형식의 값으로 변환합니다.
IntPtrToUInt INT_PTR 형식의 값을 UINT 형식의 값으로 변환합니다.
IntPtrToUInt8 형식 INT_PTR 값을 UINT8 형식의 값으로 변환합니다.
IntPtrToUIntPtr 형식 INT_PTR 값을 UINT_PTR 형식 값으로 변환합니다.
IntPtrToULong 형식 INT_PTR 값을 ULONG 형식의 값으로 변환합니다.

IntPtrToULongLong INT_PTR 형식의 값을 ULONGLONG 형식의 값으로 변환합니다.
IntPtrToULongPtr 형식 INT_PTR 값을 ULONG_PTR 형식의 값으로 변환합니다.
IntPtrToUShort 형식 INT_PTR 값을 USHORT 형식의 값으로 변환합니다.
IntSub INT 형식의 한 값을 다른 값에서 뺍니다.
IntToChar INT 형식의 값을 CHAR 형식의 값으로 변환합니다.
IntToInt8 INT 형식의 값을 INT8 형식의 값으로 변환합니다.
IntToShort INT 형식의 값을 SHORT 형식의 값으로 변환합니다.
IntToUChar INT 형식의 값을 UCHAR 형식의 값으로 변환합니다.
IntToUInt INT 형식의 값을 UINT 형식의 값으로 변환합니다.
IntToUInt8 INT 형식의 값을 UINT8 형식의 값으로 변환합니다.
IntToULong INT 형식의 값을 ULONG 형식의 값으로 변환합니다.
IntToULongLong INT 형식의 값을 UINT_PTR 형식 값으로 변환합니다.

IntToUShort INT 형식의 값을 USHORT 형식의 값으로 변환합니다.
LongAdd LONG 형식의 두 값을 추가합니다.
LongLongAdd LONGLONG 형식의 두 값을 추가합니다.
LongLongMult LONGLONG 형식의 두 값을 곱합니다.
LongLongSub LONGLONG 형식의 한 값을 다른 값에서 뺍니다.
LongLongToChar LONGLONG 형식의 값을 CHAR 형식의 값으로 변환합니다.
LongLongToInt LONGLONG 형식의 값을 INT 형식의 값으로 변환합니다.
LongLongToInt8 LONGLONG 형식의 값을 INT8 형식의 값으로 변환합니다.
LongLongToIntPtr LONGLONG 형식의 값을 INT_PTR 형식의 값으로 변환합니다.
LongLongToLong LONGLONG 형식의 값을 LONG 형식의 값으로 변환합니다.
LongLongToLongPtr LONGLONG 형식의 값을 LONG_PTR 형식의 값으로 변환합니다.
LongLongToShort LONGLONG 형식의 값을 SHORT 형식의 값으로 변환합니다.

LongLongToUChar LONGLONG 형식의 값을 UCHAR 형식의 값으로 변환합니다.
LongLongToUInt LONGLONG 형식의 값을 UINT 형식의 값으로 변환합니다.
LongLongToUInt8 LONGLONG 형식의 값을 UINT8 형식의 값으로 변환합니다.
LongLongToULong LONGLONG 형식의 값을 ULONG 형식의 값으로 변환합니다.
LongLongToULongLong LONGLONG 형식의 값을 ULONGLONG 형식의 값으로 변환합니다.
LongLongToUShort LONGLONG 형식의 값을 USHORT 형식의 값으로 변환합니다.
LongMult LONG 형식의 두 값을 곱합니다.
LongPtrAdd LONG_PTR 형식의 두 값을 추가합니다.
LongPtrMult LONG_PTR 형식의 두 값을 곱합니다.
LongPtrSub LONG_PTR 형식의 한 값을 다른 값에서 뺍니다.
LongPtrToChar LONG_PTR 형식의 값을 CHAR 형식의 값으로 변환합니다.
LongPtrToInt LONG_PTR 형식의 값을 INT 형식의 값으로 변환합니다.

LongPtrToInt8 LONG_PTR 형식의 값을 INT8 형식의 값으로 변환합니다.
LongPtrToIntPtr LONG_PTR 형식의 값을 INT_PTR 형식의 값으로 변환합니다.
LongPtrToLong LONG_PTR 형식의 값을 LONG 형식의 값으로 변환합니다.
LongPtrToShort LONG_PTR 형식의 값을 SHORT 형식의 값으로 변환합니다.
LongPtrToUChar LONG_PTR 형식의 값을 UCHAR 형식의 값으로 변환합니다.
LongPtrToUInt LONG_PTR 형식의 값을 UINT 형식의 값으로 변환합니다.
LongPtrToUInt8 LONG_PTR 형식의 값을 UINT8 형식의 값으로 변환합니다.
LongPtrToUIntPtr LONG_PTR 형식의 값을 UINT_PTR 형식의 값으로 변환합니다.
LongPtrToULong LONG_PTR 형식의 값을 ULONG 형식의 값으로 변환합니다.
LongPtrToULongLong LONG_PTR 형식의 값을 ULONGLONG 형식의 값으로 변환합니다.
LongPtrToULongPtr 형식 LONG_PTR 값을 ULONG_PTR 형식의 값으로 변환합니다.
LongPtrToUShort LONG_PTR 형식의 값을 USHORT 형식의 값으로 변환합니다.

LongSub LONG 형식의 한 값을 다른 값에서 뺍니다.
LongToChar LONG 형식의 값을 CHAR 형식의 값으로 변환합니다.
LongToInt LONG 형식의 값을 INT 형식의 값으로 변환합니다.
LongToInt8 LONG 형식의 값을 INT8 형식의 값으로 변환합니다.
LongToIntPtr LONG 형식의 값을 INT_PTR 형식의 값으로 변환합니다.
LongToShort LONG 형식의 값을 SHORT 형식의 값으로 변환합니다.
LongToUChar LONG 형식의 값을 UCHAR 형식의 값으로 변환합니다.
LongToUInt LONG 형식의 값을 UINT 형식의 값으로 변환합니다.
LongToUInt8 LONG 형식의 값을 UINT8 형식의 값으로 변환합니다.
LongToUIntPtr LONG 형식의 값을 UINT_PTR 형식의 값으로 변환합니다.
LongToULong LONG 형식의 값을 ULONG 형식의 값으로 변환합니다.
LongToULongLong LONG 형식의 값을 ULONGLONG 형식의 값으로 변환합니다.

LongToULongPtr LONG 형식의 값을 ULONG_PTR 형식의 값으로 변환합니다.
LongToUShort LONG 형식의 값을 USHORT 형식의 값으로 변환합니다.
PtrdiffTAdd ptrdiff_t 형식의 두 값을 추가합니다.
PtrdiffTMult ptrdiff_t 형식의 두 값을 곱합니다.
PtrdiffTSub ptrdiff_t 형식의 한 값을 다른 값에서 뺍니다.
ShortAdd SHORT 형식의 두 값을 추가합니다.
ShortMult SHORT 형식의 두 값을 곱합니다.
ShortSub SHORT 형식의 한 값을 다른 값에서 뺍니다.
ShortToChar SHORT 형식의 값을 CHAR 값으로 변환합니다.
ShortToDWordPtr SHORT 형식의 값을 DWORD_PTR 형식 값으로 변환합니다.
ShortToInt8 SHORT 형식의 값을 INT8 형식의 값으로 변환합니다.
ShortToUChar SHORT 형식의 값을 UCHAR 값으로 변환합니다.

ShortToUInt SHORT 형식의 값을 UINT 형식의 값으로 변환합니다.
ShortToUInt8 SHORT 형식의 값을 UINT8 형식의 값으로 변환합니다.
ShortToUIntPtr SHORT 형식의 값을 UINT_PTR 형식의 값으로 변환합니다.
ShortToULong SHORT 형식의 값을 ULONG 형식의 값으로 변환합니다.
ShortToULongLong SHORT 형식의 값을 ULONGLONG 형식의 값으로 변환합니다.
ShortToULongPtr SHORT 형식의 값을 ULONG_PTR 형식 값으로 변환합니다.
ShortToUShort SHORT 형식의 값을 USHORT 형식의 값으로 변환합니다.
SizeTAdd size_t 형식의 두 값을 추가합니다.
SizeTMult size_t 형식의 한 값을 다른 값으로 곱합니다. (SizeTMult)
SizeTSub size_t 형식의 한 값을 다른 값에서 뺍니다.
SSIZETAdd 두 개의 SSIZE_T 값을 함께 추가합니다.
SSIZETMult 한 SSIZE_T 값을 다른 값으로 곱합니다.

SSIZETSub 한 SSIZE_T 값을 다른 값에서 뺍니다.
UInt8Add UINT8 형식의 두 값을 추가합니다.
UInt8Mult UINT8 형식의 두 값을 곱합니다.
UInt8Sub UINT8 형식의 한 값을 다른 값에서 뺍니다.
UInt8ToChar UINT8 형식의 값을 CHAR 형식의 값으로 변환합니다.
UInt8ToInt8 UINT8 형식의 값을 INT8 형식의 값으로 변환합니다.
UIntAdd UINT 형식의 두 값을 추가합니다.
UIntMult UINT 형식의 한 값을 다른 값으로 곱합니다.
UIntPtrAdd UINT_PTR 형식의 두 값을 추가합니다.
UIntPtrMult UINT_PTR 형식의 한 값을 다른 값으로 곱합니다.
UIntPtrSub UINT_PTR 형식의 한 값을 다른 값에서 뺍니다.
UIntPtrToChar 형식 UINT_PTR 값을 CHAR 형식의 값으로 변환합니다.

UIntPtrToInt 형식 SIZE_T 값을 INT 형식의 값으로 변환합니다.
UIntPtrToInt16 UINT_PTR 형식의 값을 INT16 형식의 값으로 변환합니다.
UIntPtrToInt8 UINT_PTR 형식의 값을 INT8 형식의 값으로 변환합니다.
UIntPtrToIntPtr 형식 UINT_PTR 값을 INT_PTR 형식 값으로 변환합니다.
UIntPtrToLong size_t 형식의 값을 LONG 형식의 값으로 변환합니다.
UIntPtrToLongLong UINT_PTR 형식의 값을 LONGLONG 형식의 값으로 변환합니다.
UIntPtrToLongPtr 형식 UINT_PTR 값을 LONG_PTR 형식의 값으로 변환합니다.
UIntPtrToShort UINT_PTR 형식의 값을 SHORT 형식의 값으로 변환합니다.
UIntPtrToUChar 형식 UINT_PTR 값을 UCHAR 형식의 값으로 변환합니다.
UIntPtrToUInt 형식 UINT_PTR 값을 UINT 형식의 값으로 변환합니다.
UIntPtrToUInt16 UINT_PTR 형식의 값을 UINT16 형식의 값으로 변환합니다.
UIntPtrToUInt8 형식 UINT_PTR 값을 UINT8 형식의 값으로 변환합니다.

UIntPtrToULong 형식 UINT_PTR 값을 ULONG 형식의 값으로 변환합니다.
UIntPtrToUShort 형식 UINT_PTR 값을 USHORT 형식의 값으로 변환합니다.
UIntSub UINT 형식의 한 값을 다른 값에서 뺍니다.
UIntToChar UINT 형식의 값을 CHAR 형식의 값으로 변환합니다.
UIntToInt UINT 형식의 값을 INT 형식의 값으로 변환합니다.
UIntToInt8 UINT 형식의 값을 INT8 형식의 값으로 변환합니다.
UIntToIntPtr UINT 형식의 값을 INT_PTR 형식의 값으로 변환합니다.
UIntToLong UINT 형식의 값을 LONG 형식의 값으로 변환합니다.
UIntToLongPtr UINT 형식의 값을 LONG_PTR 형식의 값으로 변환합니다.
UIntToShort UINT 형식의 값을 SHORT 형식의 값으로 변환합니다.
UIntToUChar UINT 형식의 값을 UCHAR 형식의 값으로 변환합니다.
UIntToUInt8 UINT 형식의 값을 UINT8 형식의 값으로 변환합니다.

UIntToUShort UINT 형식의 값을 USHORT 형식의 값으로 변환합니다.
ULongAdd ULONG 형식의 두 값을 추가합니다.
ULongLongAdd SIZE_T 형식의 두 값을 추가합니다.
ULongLongMult size_t 형식의 한 값을 다른 값으로 곱합니다. S
ULongLongSub SIZE_T 형식의 한 값을 다른 값에서 뺍니다.
ULongLongToChar ULONGLONG 형식의 값을 CHAR 형식 값으로 변환합니다.
ULongLongToInt ULONGLONG 형식의 값을 INT 형식의 값으로 변환합니다.
ULongLongToInt8 ULONGLONG 형식의 값을 INT8 형식의 값으로 변환합니다.
ULongLongToLong ULONGLONG 형식의 값을 LONG 형식의 값으로 변환합니다.
ULongLongToLongLong ULONGLONG 형식의 값을 INT_PTR 형식의 값으로 변환합니다.
ULongLongToLongPtr ULONGLONG 형식의 값을 LONG_PTR 형식의 값으로 변환합니다.
ULongLongToShort ULONGLONG 형식의 값을 SHORT 형식의 값으로 변환합니다.

ULongLongToUChar ULONGLONG 형식의 값을 UCHAR 형식의 값으로 변환합니다.
ULongLongToUInt ULONGLONG 형식의 값을 UINT 형식의 값으로 변환합니다.
ULongLongToUInt8 ULONGLONG 형식의 값을 UINT8 형식의 값으로 변환합니다.
ULongLongToUIntPtr ULONGLONG 형식의 값을 UINT_PTR 형식의 값으로 변환합니다.
ULongLongToULong ULONGLONG 형식의 값을 ULONG 형식의 값으로 변환합니다.
ULongLongToULongPtr ULONGLONG 형식의 값을 ULONG_PTR 형식의 값으로 변환합니다.
ULongLongToUShort ULONGLONG 형식의 값을 USHORT 형식의 값으로 변환합니다.
ULongMult ULONG 형식의 한 값을 다른 값으로 곱합니다.
ULongPtrAdd ULONG_PTR 형식의 두 값을 추가합니다.
ULongPtrMult ULONG_PTR 형식의 한 값을 곱합니다.
ULongPtrSub ULONG_PTR 형식의 한 값을 다른 값에서 뺍니다.
ULongPtrToChar ULONG_PTR 형식의 값을 CHAR 형식의 값으로 변환합니다.

ULongPtrToInt size_t 형식의 값을 INT 형식의 값으로 변환합니다.
ULongPtrToInt8 ULONG_PTR 형식의 값을 INT8 형식의 값으로 변환합니다.
ULongPtrToIntPtr ULONG_PTR 형식의 값을 INT_PTR 형식의 값으로 변환합니다.
ULongPtrToLong ULONG_PTR 형식의 값을 LONG 형식의 값으로 변환합니다.
ULongPtrToLongLong SIZE_T 형식의 값을 INT64 형식의 값으로 변환합니다.
ULongPtrToLongPtr ULONG_PTR 형식의 값을 LONG_PTR 형식의 값으로 변환합니다.
ULongPtrToShort ULONG_PTR 형식의 값을 SHORT 형식의 값으로 변환합니다.
ULongPtrToUChar ULONG_PTR 형식의 값을 UCHAR 형식의 값으로 변환합니다.
ULongPtrToUInt ULONG_PTR 형식의 값을 UINT 형식의 값으로 변환합니다.
ULongPtrToUInt8 ULONG_PTR 형식의 값을 UINT8 형식의 값으로 변환합니다.
ULongPtrToUIntPtr ULONG_PTR 형식의 값을 UINT_PTR 형식의 값으로 변환합니다.
ULongPtrToULong ULONG_PTR 형식의 값을 ULONG 형식의 값으로 변환합니다.

ULongPtrToUShort ULONG_PTR 형식의 값을 USHORT 형식의 값으로 변환합니다.
ULongSub ULONG 형식의 한 값을 다른 값에서 뺍니다.
ULongToChar ULONG 형식의 값을 CHAR 형식의 값으로 변환합니다.
ULongToInt ULONG 형식의 값을 INT 형식의 값으로 변환합니다.
ULongToInt8 ULONG 형식의 값을 INT8 형식의 값으로 변환합니다.
ULongToIntPtr ULONG 형식의 값을 INT_PTR 형식의 값으로 변환합니다.
ULongToLong ULONG 형식의 값을 LONG 형식의 값으로 변환합니다.
ULongToLongPtr ULONG 형식의 값을 LONG_PTR 형식의 값으로 변환합니다.
ULongToShort ULONG 형식의 값을 SHORT 형식의 값으로 변환합니다.
ULongToUChar ULONG 형식의 값을 UCHAR 형식의 값으로 변환합니다.
ULongToUInt ULONG 형식의 값을 UINT 형식의 값으로 변환합니다.
ULongToUInt8 ULONG 형식의 값을 UINT8 형식의 값으로 변환합니다.

ULongToUIntPtr ULONG 형식의 값을 UINT_PTR 형식 값으로 변환합니다.
ULongToUShort ULONG 형식의 값을 USHORT 형식의 값으로 변환합니다.
UShortAdd USHORT 형식의 두 값을 추가합니다.
UShortMult USHORT 형식의 한 값을 다른 값으로 곱합니다.
UShortSub USHORT 형식의 한 값을 다른 값에서 뺍니다.
UShortToChar USHORT 형식의 값을 CHAR 형식의 값으로 변환합니다.
UShortToInt8 USHORT 형식의 값을 INT8 형식의 값으로 변환합니다.
UShortToShort USHORT 형식의 값을 SHORT 형식의 값으로 변환합니다.
UShortToUChar USHORT 형식의 값을 UCHAR 형식의 값으로 변환합니다.
UShortToUInt8 USHORT 형식의 값을 UINT8 형식의 값으로 변환합니다.

intshcut.h 헤더

아티클2023. 08. 24.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

intshcut.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

InetIsOffline 시스템이 인터넷에 연결되어 있는지 여부를 확인합니다.
MIMEAssociationDialogA 등록되지 않은 MIME 콘텐츠 형식 대화 상자를 실행합니다. 참고 Windows XP SP2(서비스 팩 2) 이상:_This 함수는 더 이상 지원되지 않습니다. (ANSI)
MIMEAssociationDialogW 등록되지 않은 MIME 콘텐츠 형식 대화 상자를 실행합니다. 참고 Windows XP SP2(서비스 팩 2) 이상:_This 함수는 더 이상 지원되지 않습니다. (유니코드)
TranslateURLA 지정된 URL 문자열에 일반 번역을 적용하여 새 URL 문자열을 만듭니다. (ANSI)
TranslateURLW 지정된 URL 문자열에 일반 번역을 적용하여 새 URL 문자열을 만듭니다. (유니코드)
URLAssociationDialogA 등록되지 않은 URL 프로토콜 대화 상자를 호출합니다. (ANSI)
URLAssociationDialogW 등록되지 않은 URL 프로토콜 대화 상자를 호출합니다. (유니코드)

열거형

MIMEASSOCIATIONDIALOG_IN_FLAGS MIMEAssociationDialog 함수와 함께 사용하여 실행 방법을 결정합니다.
TRANSLATEURL_IN_FLAGS TRANSLATEURL_IN_FLAGS 열거형 값은 TranslateURL 함수와 함께 사용하여 실행 방법을 결정합니다.
URLASSOCIATIONDIALOG_IN_FLAGS URLASSOCIATIONDIALOG_IN_FLAGS 열거된 값은 URLAssociationDialog와 함께 사용하여 실행 방법을 결정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ioapiset.h 헤더

아티클2023. 08. 24.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지
- 시스템 서비스

ioapiset.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CancelIo 지정된 파일에 대해 호출 스레드에서 발급한 보류 중인 모든 I/O(입력 및 출력) 작업을 취소합니다.
CancelIoEx 지정된 파일 핸들에 대해 처리 중인 I/O 작업을 표시합니다. 이 함수는 I/O 작업을 만든 스레드에 관계없이 현재 프로세스의 I/O 작업만 취소합니다.
CancelSynchronousIo 지정된 스레드에서 실행한 보류 중인 동기 I/O 작업을 취소됨으로 표시합니다.
CreateIoCompletionPort I/O(입출력) 완료 포트를 만들고 지정된 파일 핸들에 연결하거나, 아직 파일 핸들에 연결되지 않은 I/O 완료 포트를 만들어 나중에 연결할 수 있습니다.
DeviceIoControl 지정된 디바이스 드라이버에 직접 제어 코드를 보내 해당 디바이스가 해당 작업을 수행하도록 합니다.
GetOverlappedResult 지정된 파일, 명명된 파이프 또는 통신 디바이스에서 겹치는 작업의 결과를 검색합니다.
GetOverlappedResultEx 지정된 시간 제한 간격 내에 지정된 파일, 명명된 파이프 또는 통신 디바이스에서 겹치는 작업의 결과를 검색합니다. 호출 스레드는 경고 대기를 수행할 수 있습니다.

GetQueuedCompletionStatus I/O 완료 패킷을 지정된 I/O 완료 포트의 큐에서 제거하려고 시도합니다.
GetQueuedCompletionStatusEx 여러 완료 포트 항목을 동시에 검색합니다.
PostQueuedCompletionStatus I/O 완료 패킷을 I/O 완료 포트에 게시합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ioringapi.h 헤더

아티클2023. 08. 24.

I/O 링을 만들고 관리하기 위한 API를 제공합니다. I/O 링은 비동기적으로 수행할 I/O 작업의 제출 큐와 완료된 작업의 결과를 포함하는 완료 큐를 유지 관리합니다.

이 헤더는 데이터 액세스 및 스토리지에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지

ioringapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

BuildIoRingCancelRequest 이전에 제출된 I/O 링 작업을 취소하려고 시도합니다.
BuildIoRingReadFile I/O 링을 사용하여 파일에서 비동기 읽기를 수행합니다.
BuildIoRingRegisterBuffers 향후 I/O 링 작업을 위해 버퍼 배열을 시스템에 등록합니다.
BuildIoRingRegisterFileHandles 이후 I/O 링 작업을 위해 시스템에 파일 핸들 배열을 등록합니다.
CloseIoRing CreateIoRing 호출을 사용하여 이전에 연 **HIORING** 핸들을 닫습니다.
CreateIoRing I/O 링 제출/완료 큐 쌍의 새 instance 만들고 I/O 링을 참조하기 위한 핸들을 반환합니다.
GetIoRingInfo I/O 링의 API 버전 및 큐 크기에 대한 정보를 가져옵니다.
IoRingBufferRefFromIndexAndOffset

제공된 버퍼 인덱스 및 오프셋을 사용하여 IORING_BUFFER_REF 구조체의 instance 만듭니다.
IoRingBufferRefFromPointer 제공된 포인터에서 IORING_BUFFER_REF 구조체의 instance 만듭니다.
IoRingHandleRefFromHandle 제공된 파일 핸들에서 IORING_HANDLE_REF 구조체의 instance 만듭니다.
IoRingHandleRefFromIndex 제공된 인덱스에서 IORING_HANDLE_REF 구조체의 instance 만듭니다.
IsIoRingOpSupported 지정된 I/O 링에 대해 지정된 작업의 지원을 쿼리합니다.
PopIoRingCompletion 사용할 수 있는 경우 완료 큐에서 단일 항목을 팝합니다.
QueryIoRingCapabilities OS에서 IORING에 지원되는 기능을 쿼리합니다.
SetIoRingCompletionEvent 완료 큐 이벤트를 IORING에 등록합니다.
SubmitIoRing 생성되었지만 아직 제출되지 않은 항목을 커널의 큐에 제출하고 필요에 따라 작업 집합이 완료될 때까지 기다립니다.

구조체

IORING_BUFFER_REF IORING_BUFFER_REF I/O 링 작업에 사용되는 버퍼에 대한 참조를 나타냅니다.
IORING_CAPABILITIES IORING API 기능을 나타냅니다.

IORING_CQE 완료된 I/O 링 큐 항목을 나타냅니다.
IORING_CREATE_FLAGS CreateIoRing을 호출하여 I/O 링을 만들기 위한 플래그를 지정합니다.
IORING_HANDLE_REF I/O 링 작업에 사용되는 파일 핸들에 대한 참조를 나타냅니다.
IORING_INFO 지정된 I/O 링의 셰이프 및 버전 정보를 나타냅니다.

열거형

IORING_CREATE_ADVISORY_FLAGS CreateIoRing을 호출하여 I/O 링을 만들기 위한 권고 플래그를 지정합니다.
IORING_CREATE_REQUIRED_FLAGS CreateIoRing을 호출하여 I/O 링을 만드는 데 필요한 플래그를 지정합니다.
IORING_REF_KIND IORING_HANDLE_REF 구조체의 형식을 지정합니다.
IORING_SQE_FLAGS I/O 링 제출 큐 항목에 대한 커널 동작 옵션을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ip2string.h 헤더

아티클2023. 08. 24.

이 헤더는 IP 도우미에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- IP 도우미

ip2string.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

RtlEthernetAddressToStringA 이진 이더넷 주소를 이더넷 MAC 주소의 문자열 표현으로 변환합니다. (ANSI)
RtlEthernetAddressToStringW 이진 이더넷 주소를 이더넷 MAC 주소의 문자열 표현으로 변환합니다. (유니코드)
RtlEthernetStringToAddressA 이더넷 MAC 주소의 문자열 표현을 이더넷 주소의 이진 형식으로 변환합니다. (ANSI)
RtlEthernetStringToAddressW 이더넷 MAC 주소의 문자열 표현을 이더넷 주소의 이진 형식으로 변환합니다. (유니코드)
RtlIpv4AddressToStringA IPv4 주소를 인터넷 표준 점선 소수점 형식의 문자열로 변환합니다. (ANSI)
RtlIpv4AddressToStringExA IPv4 주소 및 포트 번호를 인터넷 표준 형식의 문자열로 변환합니다. (ANSI)
RtlIpv4AddressToStringExW IPv4 주소 및 포트 번호를 인터넷 표준 형식의 문자열로 변환합니다. (유니코드)
RtlIpv4AddressToStringW IPv4 주소를 인터넷 표준 점선 소수점 형식의 문자열로 변환합니다. (유니코드)
RtlIpv4StringToAddressA

IPv4 주소의 문자열 표현을 이진 IPv4 주소로 변환합니다. (ANSI)
RtlIpv4StringToAddressExA IPv4 주소 및 포트 번호의 문자열 표현을 이진 IPv4 주소 및 포트로 변환합니다. (ANSI)
RtlIpv4StringToAddressExW IPv4 주소 및 포트 번호의 문자열 표현을 이진 IPv4 주소 및 포트로 변환합니다. (유니코드)
RtlIpv4StringToAddressW IPv4 주소의 문자열 표현을 이진 IPv4 주소로 변환합니다. (유니코드)
RtlIpv6AddressToStringA IPv6 주소를 인터넷 표준 형식의 문자열로 변환합니다. (ANSI)
RtlIpv6AddressToStringExA IPv6 주소, scope ID 및 포트 번호를 문자열로 변환합니다. (ANSI)
RtlIpv6AddressToStringExW IPv6 주소, scope ID 및 포트 번호를 문자열로 변환합니다. (유니코드)
RtlIpv6AddressToStringW IPv6 주소를 인터넷 표준 형식의 문자열로 변환합니다. (유니코드)
RtlIpv6StringToAddressA IPv6 주소의 문자열 표현을 이진 IPv6 주소로 변환합니다. (ANSI)
RtlIpv6StringToAddressExA IPv6 주소, scope ID 및 포트 번호의 문자열 표현을 이진 IPv6 주소, scope ID 및 포트로 변환합니다. (ANSI)
RtlIpv6StringToAddressExW IPv6 주소, scope ID 및 포트 번호의 문자열 표현을 이진 IPv6 주소, scope ID 및 포트로 변환합니다. (유니코드)
RtlIpv6StringToAddressW IPv6 주소의 문자열 표현을 이진 IPv6 주소로 변환합니다. (유니코드)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ipexport.h 헤더

아티클2023. 08. 24.

이 헤더는 IP 도우미에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- IP 도우미

ipexport.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

ARP_SEND_REPLY ARP_SEND_REPLY 구조는 ARP(주소 확인 프로토콜) 회신 메시지에 대한 정보를 저장합니다.
ICMP_ECHO_REPLY IPv4 에코 요청에 대한 응답으로 반환된 데이터를 설명합니다.
ICMP_ECHO_REPLY32 64비트 플랫폼의 IPv4 에코 요청에 대한 응답으로 반환된 데이터를 설명합니다.
ICMPV6_ECHO_REPLY_LH IPv6 에코 요청에 대한 응답으로 반환된 데이터를 설명합니다.
IP_ADAPTER_INDEX_MAP IP_ADAPTER_INDEX_MAP 구조체는 IPv4를 사용하도록 설정된 네트워크 어댑터와 연결된 인터페이스 인덱스와 네트워크 어댑터의 이름을 함께 저장합니다.
IP_ADAPTER_ORDER_MAP IP_ADAPTER_ORDER_MAP 구조체는 어댑터에 대한 정보 배열과 해당 상대 우선 순위를 로컬 컴퓨터에 저장합니다.
IP_INTERFACE_INFO IP_INTERFACE_INFO 구조에는 로컬 시스템에서 IPv4를 사용하도록 설정된 네트워크 인터페이스 어댑터 목록이 포함되어 있습니다.
IP_MCAST_COUNTER_INFO IP_MCAST_COUNTER_INFO 구조는 멀티캐스트 트래픽에 대한 통계 정보를 저장합니다.

IP_OPTION_INFORMATION IP 패킷의 헤더에 포함할 옵션에 대해 설명합니다.
IP_OPTION_INFORMATION32 64비트 플랫폼의 IP 패킷 헤더에 포함할 옵션에 대해 설명합니다.
IP_UNIDIRECTIONAL_ADAPTER_ADDRESS IP_UNIDIRECTIONAL_ADAPTER_ADDRESS 구조체는 단방향 어댑터와 연결된 IPv4 주소를 저장합니다.
IPV6_ADDRESS_EX IPv6 주소를 저장합니다.
TCP_RESERVE_PORT_RANGE TCP_RESERVE_PORT_RANGE 구조체는 예약할 TCP 포트 범위를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

iphlpapi.h 헤더

아티클2023. 08. 24.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- IP 도우미
- Windows 셸

iphlpapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AddIPAddress AddIPAddress 함수는 지정된 IPv4 주소를 지정된 어댑터에 추가합니다.
AllocateAndGetTcpExTableFromStack TCP 연결 테이블을 검색하고 로컬 힙에서 메모리를 할당하여 테이블을 저장합니다.
AllocateAndGetUdpExTableFromStack UDP 연결 테이블을 검색하고 로컬 힙에서 메모리를 할당하여 테이블을 저장합니다.
CancelIfTimestampConfigChange 이 함수는 시스템 사용을 위해 예약되어 있으며 코드에서 호출하면 안 됩니다. (CancelIfTimestampConfigChange)
CancelIPChangeNotify NotifyAddrChange 또는 NotifyRouteChange 함수를 성공적으로 호출하여 이전에 요청된 IPv4 주소 및 경로 변경 내용에 대한 알림을 취소합니다.
CaptureInterfaceHardwareCrossTimestamp 네트워크 어댑터에 대한 크로스 타임스탬프 정보를 검색합니다.
CreateIpForwardEntry CreateIpForwardEntry 함수는 로컬 컴퓨터의 IPv4 라우팅 테이블에 경로를 만듭니다.
CreateIpNetEntry CreateIpNetEntry 함수는 로컬 컴퓨터의 ARP 테이블에 ARP(주소 확인 프로토콜) 항목을 만듭니다.

CreatePersistentTcpPortReservation 로컬 컴퓨터에서 TCP 포트의 연속 블록에 대한 영구 TCP 포트 예약을 만듭니다.
CreatePersistentUdpPortReservation 로컬 컴퓨터에서 연속된 UDP 포트 블록에 대한 영구 UDP 포트 예약을 만듭니다.
CreateProxyArpEntry CreateProxyArpEnry 함수는 지정된 IPv4 주소에 대한 로컬 컴퓨터에 PARP(프록시 주소 확인 프로토콜) 항목을 만듭니다.
DeleteIPAddress DeleteIPAddress 함수는 AddIPAddress를 사용하여 이전에 추가한 IP 주소를 삭제합니다.
DeleteIpForwardEntry 로컬 컴퓨터의 IPv4 라우팅 테이블에서 기존 경로를 삭제합니다.
DeleteIpNetEntry DeleteIpNetEntry 함수는 로컬 컴퓨터의 ARP 테이블에서 ARP 항목을 삭제합니다.
DeletePersistentTcpPortReservation 로컬 컴퓨터에서 연속된 TCP 포트 블록에 대한 영구 TCP 포트 예약을 삭제합니다. (DeletePersistentTcpPortReservation)
DeletePersistentUdpPortReservation 로컬 컴퓨터에서 연속된 TCP 포트 블록에 대한 영구 TCP 포트 예약을 삭제합니다. (DeletePersistentUdpPortReservation)
DeleteProxyArpEntry DeleteProxyArpEntry 함수는 dwAddress 및 dwIfIndex 매개 변수로 지정된 로컬 컴퓨터에서 PARP 항목을 삭제합니다.
DisableMediaSense DisableMediaSense 함수는 로컬 컴퓨터에서 TCP/IP 스택의 미디어 감지 기능을 사용하지 않도록 설정합니다.
EnableRouter EnableRouter 함수는 로컬 컴퓨터에서 IPv4 전달을 켭니다. 또한 EnableRouter는 IPv4 전달을 사용하도록 설정하는 요청 수를 추적하는 참조 횟수를 증가합니다.

FlushIpNetTable FlushIpNetTable 함수는 로컬 컴퓨터의 ARP 테이블에서 지정된 인터페이스에 대한 모든 ARP 항목을 삭제합니다.
GetAdapterIndex GetAdapterIndex 함수는 이름이 지정된 어댑터의 인덱스를 가져옵니다.
GetAdapterOrderMap GetAdapterOrderMap 함수는 로컬 컴퓨터의 인터페이스에 대한 우선 순위를 나타내는 어댑터 순서 맵을 가져옵니다.
GetAdaptersAddresses 로컬 컴퓨터의 어댑터와 연결된 주소를 검색합니다.
GetAdaptersInfo GetAdaptersInfo 함수는 로컬 컴퓨터에 대한 어댑터 정보를 검색합니다.
GetBestInterface GetBestInterface 함수는 지정된 IPv4 주소에 가장 적합한 경로가 있는 인터페이스의 인덱스를 검색합니다.
GetBestInterfaceEx GetBestInterfaceEx 함수는 지정된 IPv4 또는 IPv6 주소에 가장 적합한 경로가 있는 인터페이스의 인덱스를 검색합니다.
GetBestRoute GetBestRoute 함수는 지정된 대상 IP 주소에 대한 최상의 경로를 검색합니다.
GetExtendedTcpTable 애플리케이션에서 사용할 수 있는 TCP 엔드포인트 목록이 포함된 테이블을 검색합니다.
GetExtendedUdpTable 애플리케이션에서 사용할 수 있는 UDP 엔드포인트 목록이 포함된 테이블을 검색합니다.
GetFriendlyIfIndex 인터페이스 인덱스를 사용하고 이전 버전과 호환되는 인터페이스 인덱스, 즉 하위 24비트만 사용하는 인덱스를 반환합니다.

GetIcmpStatistics GetIcmpStatistics 함수는 로컬 컴퓨터에 대한 IPv4 통계에 대한 ICMP(인터넷 제어 메시지 프로토콜)를 검색합니다.
GetIcmpStatisticsEx GetIcmpStatisticsEx 함수는 로컬 컴퓨터에 대한 ICMP(인터넷 제어 메시지 프로토콜) 통계를 검색합니다. GetIcmpStatisticsEx 함수는 IPv6 ICMP 통계를 검색할 수 있습니다.
GetIfEntry GetIfEntry 함수는 로컬 컴퓨터에서 지정된 인터페이스에 대한 정보를 검색합니다.
GetIfTable GetIfTable 함수는 MIB-II 인터페이스 테이블을 검색합니다.
GetInterfaceActiveTimestampCapabilities 네트워크 어댑터의 현재 사용 가능한 타임스탬프 기능을 검색합니다.
GetInterfaceCurrentTimestampCapabilities 이 함수는 시스템 사용을 위해 예약되어 있으며 코드에서 호출하면 안 됩니다. (GetInterfaceCurrentTimestampCapabilities)
GetInterfaceHardwareTimestampCapabilities 이 함수는 시스템 사용을 위해 예약되어 있으며 코드에서 호출하면 안 됩니다. (GetInterfaceHardwareTimestampCapabilities)
GetInterfaceInfo GetInterfaceInfo 함수는 로컬 시스템에서 IPv4를 사용하도록 설정된 네트워크 인터페이스 어댑터 목록을 가져옵니다.
GetInterfaceSupportedTimestampCapabilities 네트워크 어댑터의 지원되는 타임스탬프 기능을 검색합니다.
GetIpAddrTable GetIpAddrTable 함수는 IPv4 주소 매핑 테이블에 대한 인터페이스를 검색합니다.
GetIpErrorString GetIpErrorString 함수는 IP 도우미 오류 문자열을 검색합니다.

GetIpForwardTable GetIpForwardTable 함수는 IPv4 라우팅 테이블을 검색합니다.
GetIpNetTable GetIpNetTable 함수는 IPv4에서 실제 주소 매핑 테이블을 검색합니다.
GetIpStatistics GetIpStatistics 함수는 현재 컴퓨터의 IP 통계를 검색합니다.
GetIpStatisticsEx GetIpStatisticsEx 함수는 현재 컴퓨터에 대한 IP(인터넷 프로토콜) 통계를 검색합니다.
GetNetworkParams GetNetworkParams 함수는 로컬 컴퓨터에 대한 네트워크 매개 변수를 검색합니다.
GetNumberOfInterfaces GetNumberOfInterfaces 함수는 로컬 컴퓨터의 인터페이스 수를 검색합니다.
GetOwnerModuleFromTcp6Entry MIB 테이블 행에서 특정 IPv6 TCP 엔드포인트에 대한 컨텍스트 바인딩을 실행한 모듈에 대한 데이터를 검색합니다.
GetOwnerModuleFromTcpEntry MIB 테이블 행에서 특정 IPv4 TCP 엔드포인트에 대한 컨텍스트 바인딩을 실행한 모듈에 대한 데이터를 검색합니다.
GetOwnerModuleFromUdp6Entry MIB 테이블 행에서 특정 IPv6 UDP 엔드포인트에 대한 컨텍스트 바인딩을 실행한 모듈에 대한 데이터를 검색합니다.
GetOwnerModuleFromUdpEntry MIB 테이블 행에서 특정 IPv4 UDP 엔드포인트에 대한 컨텍스트 바인딩을 실행한 모듈에 대한 데이터를 검색합니다.
GetPerAdapterInfo GetPerAdapterInfo 함수는 지정된 인터페이스에 해당하는 어댑터에 대한 정보를 검색합니다.

GetPerTcp6ConnectionEStats IPv6 TCP 연결에 대한 확장 통계를 검색합니다.
GetPerTcpConnectionEStats IPv4 TCP 연결에 대한 확장 통계를 검색합니다.
GetRTTAndHopCount GetRTTAndHopCount 함수는 RTT(왕복 시간) 및 지정된 대상에 대한 홉 수를 결정합니다.
GetTcp6Table IPv6에 대한 TCP 연결 테이블을 검색합니다. (GetTcp6Table)
GetTcp6Table2 IPv6에 대한 TCP 연결 테이블을 검색합니다. (GetTcp6Table2)
GetTcpStatistics GetTcpStatistics 함수는 로컬 컴퓨터에 대한 TCP 통계를 검색합니다.
GetTcpStatisticsEx GetTcpStatisticsEx 함수는 현재 컴퓨터에 대한 TCP(Transmission Control Protocol) 통계를 검색합니다.
GetTcpStatisticsEx2 GetTcpStatisticsEx2 함수는 현재 컴퓨터에 대한 TCP(Transmission Control Protocol) 통계를 검색합니다.
GetTcpTable IPv4 TCP 연결 테이블을 검색합니다. (GetTcpTable)
GetTcpTable2 IPv4 TCP 연결 테이블을 검색합니다. (GetTcpTable2)
GetUdp6Table IPv6 UDP(사용자 데이터그램 프로토콜) 수신기 테이블을 검색합니다.
GetUdpStatistics

GetUdpStatistics 함수는 로컬 컴퓨터에 대한 UDP(사용자 데이터그램 프로토콜) 통계를 검색합니다.
GetUdpStatisticsEx GetUdpStatisticsEx 함수는 현재 컴퓨터에 대한 UDP(사용자 데이터그램 프로토콜) 통계를 검색합니다.
GetUdpStatisticsEx2 GetUdpStatisticsEx2 함수는 현재 컴퓨터에 대한 UDP(사용자 데이터그램 프로토콜) 통계를 검색합니다.
GetUdpTable IPv4 UDP(사용자 데이터그램 프로토콜) 수신기 테이블을 검색합니다.
GetUniDirectionalAdapterInfo GetUniDirectionalAdapterInfo 함수는 로컬 컴퓨터에 설치된 단방향 어댑터에 대한 정보를 검색합니다. 단방향 어댑터는 데이터그램을 수신할 수 있지만 전송할 수 없는 어댑터입니다.
IpReleaseAddress IpReleaseAddress 함수는 DHCP(동적 호스트 구성 프로토콜)를 통해 이전에 가져온 IPv4 주소를 해제합니다.
IpRenewAddress IpRenewAddressfunction은 DHCP(동적 호스트 구성 프로토콜)를 통해 이전에 가져온 IPv4 주소에 대한 임대를 갱신합니다.
LookupPersistentTcpPortReservation 로컬 컴퓨터에서 연속된 TCP 포트 블록에 대한 영구 TCP 포트 예약에 대한 토큰을 찾습니다.
LookupPersistentUdpPortReservation 로컬 컴퓨터에서 TCP 포트의 연속 블록에 대한 영구 UDP 포트 예약에 대한 토큰을 찾습니다.
NhpAllocateAndGetInterfaceInfoFromStack NhpAllocateAndGetInterfaceInfoFromStack 함수는 로컬 컴퓨터에 대한 어댑터 정보를 가져옵니다.
NotifyAddrChange NotifyAddrChange 함수는 IPv4 주소를 인터페이스에 매핑하는 테이블에서 변경이 발생할 때마다 호출자에게 알림을 보냅니다.

NotifyIfTimestampConfigChange 이 함수는 시스템 사용을 위해 예약되어 있으므로 코드에서 호출하면 안 됩니다. (NotifyIfTimestampConfigChange)
NotifyRouteChange NotifyRouteChange 함수는 IPv4 라우팅 테이블에서 변경이 발생할 때마다 호출자에게 알림을 보냅니다.
ParseNetworkString 입력 네트워크 문자열을 구문 분석하고 지정된 IP 네트워크 문자열 형식의 법적 표현인지 확인합니다. 문자열이 형식 및 해당 사양과 일치하는 경우 함수는 선택적으로 구문 분석된 결과를 반환할 수 있습니다.
RegisterInterfaceTimestampConfigChange 시스템에서 타임스탬프 기능 변경을 알리기 위해 호출하는 사용자 구현 콜백 함수를 등록합니다.
ResolveNeighbor 로컬 컴퓨터에서 인접 IP 주소 항목의 실제 주소를 확인합니다. (ResolveNeighbor)
RestoreMediaSense RestoreMediaSense 함수는 DisableMediaSense 함수가 이전에 호출된 로컬 컴퓨터에서 TCP/IP 스택의 미디어 감지 기능을 복원합니다.
SendARP SendARP 함수는 지정된 대상 IPv4 주소에 해당하는 실제 주소를 가져오기 위해 ARP(주소 확인 프로토콜) 요청을 보냅니다.
SetIfEntry SetIfEntry 함수는 인터페이스의 관리 상태 설정합니다.
SetIpForwardEntry SetIpForwardEntry 함수는 로컬 컴퓨터의 IPv4 라우팅 테이블에서 기존 경로를 수정합니다.
SetIpNetEntry SetIpNetEntry 함수는 로컬 컴퓨터의 ARP 테이블에 있는 기존 ARP 항목을 수정합니다.
SetIpStatistics

SetIpStatistics 함수는 IP 전달을 설정하거나 해제하고 로컬 컴퓨터에 대한 기본 TTL(TTL) 값을 설정합니다.
SetIpStatisticsEx IP 전달을 설정하거나 해제하고 로컬 컴퓨터의 기본 TTL(TTL) 값을 설정합니다.
SetIpTTL SetIpTTL 함수는 로컬 컴퓨터의 기본 TTL(Time to Live) 값을 설정합니다.
SetPerTcp6ConnectionEStats IPv6 TCP 연결에 대한 읽기/쓰기 정보의 값을 설정합니다. 이 함수는 IPv6 TCP 연결에 대한 확장 통계를 사용하거나 사용하지 않도록 설정하는 데 사용됩니다.
SetPerTcpConnectionEStats IPv4 TCP 연결에 대한 읽기/쓰기 정보의 값을 설정합니다. 이 함수는 IPv4 TCP 연결에 대한 확장 통계를 사용하거나 사용하지 않도록 설정하는 데 사용됩니다.
SetTcpEntry SetTcpEntry 함수는 TCP 연결의 상태를 설정합니다.
UnenableRouter UnenableRouter 함수는 IPv4 전달을 사용하도록 설정하는 요청 수를 추적하는 참조 수를 감소합니다. 이 참조 수가 0에 도달하면 UnenableRouter는 로컬 컴퓨터에서 IPv4 전달을 해제합니다.
UnregisterInterfaceTimestampConfigChange **RegisterInterfaceTimestampConfigChange** 호출에서 등록한 콜백 함수를 등록 취소하여 타임스탬프 기능 변경에 대한 알림을 취소합니다.

콜백 함수

INTERFACE_TIMESTAMP_CONFIG_CHANGE_CALLBACK 네트워크 어댑터의 타임스탬프 기능에 대한 변경 내용을 알리기 위해 앱에서 구현하는 콜백 함수입니다.

구조체

INTERFACE_HARDWARE_CROSSTIMESTAMP 네트워크 어댑터에서 검색된 크로스 타임스탬프에 대해 설명합니다.
INTERFACE_HARDWARE_TIMESTAMP_CAPABILITIES 네트워크 인터페이스 카드(NIC) 하드웨어의 타임스탬프 기능을 설명합니다.
INTERFACE_SOFTWARE_TIMESTAMP_CAPABILITIES NIC 미니포트 드라이버의 소프트웨어 타임스탬프 기능에 대해 설명합니다.
INTERFACE_TIMESTAMP_CAPABILITIES 네트워크 어댑터가 지원하는 정확한 타임스탬프 기능을 설명합니다.
INTERFACE_TIMESTAMP_CAPABILITY_FLAGS 이 구조체는 시스템 사용을 위해 예약되어 있으며 코드에서 사용하지 않아야 합니다.
NET_ADDRESS_INFO ParseNetworkString 함수에서 반환된 IP 주소 정보를 포함합니다.

열거형

NET_ADDRESS_FORMAT NET_ADDRESS_FORMAT 열거형은 ParseNetworkString 함수에서 반환하는 네트워크 주소의 형식을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ipmib.h 헤더

아티클2023. 08. 24.

이 헤더는 관리 정보 베이스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 관리 정보 기반

ipmib.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

MIB_ICMP 특정 컴퓨터에 대한 ICMP(인터넷 제어 메시지 프로토콜) 통계를 포함합니다.
MIB_ICMP_EX_XPSP1 특정 컴퓨터에 대한 확장된 ICMP(인터넷 제어 메시지 프로토콜) 통계를 포함합니다.
MIB_IPADDRROW_W2K MIB_IPADDRROW_W2K(ipmib.h)는 MIB_IPADDRTABLE 구조의 특정 IPv4 주소에 대한 정보를 지정합니다.
MIB_IPADDRROW_XP MIB_IPADDRROW_XP(ipmib.h)는 MIB_IPADDRTABLE 구조의 특정 IPv4 주소에 대한 정보를 지정합니다.
MIB_IPADDRTABLE IPv4 주소 항목의 테이블을 포함합니다.
MIB_IPFORWARDNUMBER 특정 IP 라우팅 테이블에 경로 수를 저장합니다.
MIB_IPFORWARDROW IPv4 네트워크 경로를 설명하는 정보를 포함합니다.
MIB_IPFORWARDTABLE IPv4 경로 항목의 테이블을 포함합니다.

MIB_IPMCAST_GLOBAL IP 멀티캐스트에 대한 전역 정보를 특정 컴퓨터에 저장합니다.
MIB_IPMCAST_IF_ENTRY IP 멀티캐스트 인터페이스에 대한 정보를 저장합니다.
MIB_IPMCAST_IF_TABLE IP 멀티캐스트 인터페이스 항목의 테이블을 포함합니다.
MIB_IPMCAST_MFE IP(인터넷 프로토콜) MFE(멀티캐스트 전달 항목)에 대한 정보를 저장합니다.
MIB_IPMCAST_MFE_STATS MFE(멀티캐스트 전달 항목)와 연결된 통계를 저장합니다.
MIB_IPMCAST_MFE_STATS_EX_XP MFE(멀티캐스트 전달 항목)와 연결된 확장된 통계를 저장합니다.
MIB_IPMCAST_OIF_STATS_LH MIB_IPMCAST_OIF_STATS_LH(ipmib.h)는 나가는 멀티캐스트 인터페이스와 연결된 통계를 저장합니다.
MIB_IPMCAST_OIF_STATS_W2K MIB_IPMCAST_OIF_STATS_W2K(ipmib.h)은 나가는 멀티캐스트 인터페이스와 연결된 통계를 저장합니다.
MIB_IPMCAST_OIF_W2K MIB_IPMCAST_OIF_W2K(ipmib.h)는 나가는 IP 멀티캐스트 패킷을 보내는 데 필요한 정보를 저장합니다.
MIB_IPMCAST_OIF_XP MIB_IPMCAST_OIF_XP(ipmib.h)는 나가는 IP 멀티캐스트 패킷을 보내는 데 필요한 정보를 저장합니다.
MIB_IPNETROW_LH MIB_IPNETROW_LH(ipmib.h)에는 IPv4 주소에 대한 ARP(주소 확인 프로토콜) 테이블 항목에 대한 정보가 포함되어 있습니다.

MIB_IPNETROW_W2K MIB_IPNETROW_W2K(ipmib.h)에는 IPv4 주소에 대한 ARP(주소 확인 프로토콜) 테이블 항목에 대한 정보가 포함되어 있습니다.
MIB_IPNETTABLE IPv4 주소에 대한 ARP(주소 확인 프로토콜) 항목의 테이블을 포함합니다.
MIB_IPSTATS_LH MIB_IPSTATS_LH(ipmib.h)는 특정 컴퓨터에서 실행되는 IP 프로토콜에 대한 정보를 저장합니다.
MIB_IPSTATS_W2K MIB_IPSTATS_W2K(ipmib.h)는 특정 컴퓨터에서 실행되는 IP 프로토콜에 대한 정보를 저장합니다.
MIB_MFE_STATS_TABLE MFE(멀티캐스트 전달 항목) 그룹에 대한 통계를 저장합니다.
MIB_MFE_STATS_TABLE_EX_XP MFE(멀티캐스트 전달 항목)에 대한 확장 통계 테이블을 포함합니다.
MIB_MFE_TABLE MFE(멀티캐스트 전달 항목)의 테이블을 포함합니다.
MIBICMPINFO 특정 컴퓨터에 대한 ICMP(인터넷 제어 메시지 프로토콜) 통계를 포함합니다.
MIBICMPSTATS 특정 컴퓨터에서 들어오거나 나가는 ICMP(인터넷 제어 메시지 프로토콜) 메시지에 대한 통계를 포함합니다.
MIBICMPSTATS_EX_XPSP1 특정 컴퓨터에서 들어오거나 나가는 ICMP(인터넷 제어 메시지 프로토콜) 메시지에 대한 확장 통계를 포함합니다.

열거형

ICMP4_TYPE IP 버전 4.0(IPv4) 메시지 유형에 대한 ICMP(인터넷 제어 메시지 프로토콜) 집합을 정의합니다.
ICMP6_TYPE IP 버전 6.0(ICMPv6) 메시지 유형에 대한 인터넷 제어 메시지 프로토콜 집합을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

iprtrmib.h 헤더

아티클2023. 08. 24.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- IP 도우미
- 관리 정보 기반

iprtrmib.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

MIB_BEST_IF 특정 대상 IPv4 주소에 가장 적합한 경로가 있는 인터페이스의 인덱스가 저장됩니다.
MIB_BOUNDARYROW 멀티캐스트 경계에 대한 IPv4 그룹 주소 값과 마스크를 포함합니다.
MIB_IFSTATUS 특정 인터페이스에 대한 상태 정보를 저장합니다.
MIB_IPMCAST_BOUNDARY 라우터의 범위가 지정된 IPv4 멀티캐스트 주소 경계를 나열하는 MIB_IPMCAST_BOUNDARY_TABLE 구조의 행을 포함합니다.
MIB_IPMCAST_BOUNDARY_TABLE 라우터의 범위가 지정된 IPv4 멀티캐스트 주소 경계 목록을 포함합니다.
MIB_IPMCAST_SCOPE scope 정의하는 멀티캐스트 scope 이름과 연결된 IPv4 멀티캐스트 그룹 주소 및 마스크를 포함합니다.
MIB_MCAST_LIMIT_ROW MIB_MCAST_LIMIT_ROW 구조체에는 해당 MIB_IPMCAST_IF_ENTRY 구조체의 구성 가능한 제한 정보가 포함됩니다.
MIB_OPAQUE_INFO

MIB 불투명 쿼리에서 반환된 정보를 포함합니다.
MIB_OPAQUE_QUERY MIB 불투명 쿼리에 대한 정보를 포함합니다.
MIB_PROXYARP PARP(프록시 주소 확인 프로토콜) 항목에 대한 정보를 저장합니다.
TCPIP_OWNER_MODULE_BASIC_INFO TCP 연결과 연결된 모듈 이름 및 모듈 경로 값에 대한 포인터를 포함합니다. TCPIP_OWNER_MODULE_BASIC_INFO 구조체는 GetOwnerModuleFromTcpEntry 및 GetOwnerModuleFromTcp6Entry 함수에 의해 반환됩니다.

열거형

TCP_TABLE_CLASS GetExtendedTcpTable 호출에서 반환되는 테이블 유형을 나타내는 데 사용되는 값 집합을 정의합니다.
TCPIP_OWNER_MODULE_INFO_CLASS GetOwnerModuleFromXXXEntry 제품군의 호출에 전달되는 모듈 정보 구조의 형식을 정의합니다.
UDP_TABLE_CLASS GetExtendedUdpTable 호출에서 반환되는 테이블 유형을 나타내는 데 사용되는 값 집합을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ipsectypes.h 헤더

아티클2023. 08. 24.

이 헤더는 Windows 필터링 플랫폼에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 필터링 플랫폼

ipsectypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

IPSEC_ADDRESS_INFO0 모바일 추가 주소 정보를 저장하는 데 사용됩니다.
IPSEC_AGGREGATE_DROP_PACKET_STATISTICS0 집계 IPsec 커널 패킷 삭제 통계를 저장합니다. 참고 IPSEC_AGGREGATE_DROP_PACKET_STATISTICS0 Windows Vista에서 사용되는 IPSEC_AGGREGATE_DROP_PACKET_STATISTICS 특정 구현입니다.
IPSEC_AGGREGATE_DROP_PACKET_STATISTICS1 집계 IPsec 커널 패킷 삭제 통계를 저장합니다.
IPSEC_AGGREGATE_SA_STATISTICS0 집계 IPsec SA(커널 보안 연결) 통계를 저장합니다.
IPSEC_AH_DROP_PACKET_STATISTICS0 IPsec AH 드롭 패킷 통계를 저장합니다.
IPSEC_AUTH_AND_CIPHER_TRANSFORM0 IPsec 빠른 모드 정책에서 SA 변환에 대한 해시 및 암호화 특정 정보를 함께 저장하는 데 사용됩니다.
IPSEC_AUTH_TRANSFORM_ID0 IPsec SA(보안 연결)에서 사용되는 해시 알고리즘을 고유하게 식별하는 데 사용됩니다.
IPSEC_AUTH_TRANSFORM0 SA 변환에 대한 해시 관련 정보를 지정합니다.

IPSEC_CIPHER_TRANSFORM_ID0 IPsec SA에서 사용되는 암호화 알고리즘을 고유하게 식별하는 데 사용되는 정보를 지정합니다.
IPSEC_CIPHER_TRANSFORM0 IPsec 빠른 모드 정책에서 SA 변환에 대한 암호화 특정 정보를 저장하는 데 사용됩니다.
IPSEC_DOSP_OPTIONS0 IPsec DoS Protection에 대한 구성 매개 변수를 저장하는 데 사용됩니다.
IPSEC_DOSP_STATE_ENUM_TEMPLATE0 IPSEC_DOSP_STATE_ENUM_TEMPLATE0 구조체입니다.
IPSEC_DOSP_STATE0 IPsec DoS Protection에 대한 상태 정보를 저장하는 데 사용됩니다.
IPSEC_DOSP_STATISTICS0 IPSEC_DOSP_STATISTICS0 구조체입니다.
IPSEC_ESP_DROP_PACKET_STATISTICS0 ESP 드롭 패킷 통계를 저장합니다.
IPSEC_GETSPI0 IPSEC_GETSPI0 구조에는 IPsec 드라이버에서 SPI(보안 매개 변수 인덱스)를 요청할 때 제공해야 하는 정보가 포함되어 있습니다. 참고 IPSEC_GETSPI0 Windows Vista에서 사용되는 IPSEC_GETSPI 특정 구현입니다.
IPSEC_GETSPI1 IPSEC_GETSPI1 구조에는 IPsec 드라이버에서 SPI(보안 매개 변수 인덱스)를 요청할 때 제공해야 하는 정보가 포함되어 있습니다. 참고 IPSEC_GETSPI1 Windows 7 이상에서 사용되는 IPSEC_GETSPI 특정 구현입니다.
IPSEC_ID0 IPsec에서 인증하는 ID에 해당하는 정보를 포함합니다.
IPSEC_KEY_MANAGER0 IPsec에 키 관리 콜백을 등록하는 데 사용됩니다.

IPSEC_KEYING_POLICY0 IPsec에 대해 시도할 순서가 지정되지 않은 키 지정 모듈 집합을 정의합니다. (IPSEC_KEYING_POLICY0)
IPSEC_KEYING_POLICY1 IPsec에 대해 시도할 순서가 지정되지 않은 키 지정 모듈 집합을 정의합니다. (IPSEC_KEYING_POLICY1)
IPSEC_KEYMODULE_STATE0 IPsec(인터넷 프로토콜 보안) 키 지정 모듈 관련 정보를 저장합니다.
IPSEC_PROPOSAL0 IPsec 빠른 모드 제안을 저장하는 데 사용됩니다.
IPSEC_SA_AUTH_AND_CIPHER_INFORMATION0 IPsec SA(보안 연결)의 인증 및 암호화 알고리즘에 대한 정보를 저장합니다.
IPSEC_SA_AUTH_INFORMATION0 IPsec SA(보안 연결)의 인증 알고리즘에 대한 정보를 저장합니다.
IPSEC_SA_BUNDLE0 IPsec SA(보안 연결) 번들에 대한 정보를 저장하는 데 사용됩니다. (IPSEC_SA_BUNDLE0)
IPSEC_SA_BUNDLE1 IPsec SA(보안 연결) 번들에 대한 정보를 저장하는 데 사용됩니다. (IPSEC_SA_BUNDLE1)
IPSEC_SA_CIPHER_INFORMATION0 IPsec SA(보안 연결)의 암호화 알고리즘에 대한 정보를 저장합니다.
IPSEC_SA_CONTEXT_CHANGE0 IPsec SA(보안 연결) 컨텍스트 변경에 대한 정보를 포함합니다.
IPSEC_SA_CONTEXT_ENUM_TEMPLATE0 SA(보안 연결) 컨텍스트를 열거하는 데 사용되는 열거형 템플릿입니다.
IPSEC_SA_CONTEXT_SUBSCRIPTION0 특정 IPsec SA(보안 연결) 컨텍스트에 대한 알림을 구독하는 데 사용되는 정보를 저장합니다.

IPSEC_SA_CONTEXT0 인바운드 및 아웃바운드 SA 쌍을 캡슐화합니다.
IPSEC_SA_CONTEXT1 인바운드 및 아웃바운드 SA(보안 연결) 쌍을 캡슐화합니다.
IPSEC_SA_DETAILS0 IPsec SA(보안 연결)를 열거할 때 반환되는 정보를 저장하는 데 사용됩니다. (IPSEC_SA_DETAILS0)
IPSEC_SA_DETAILS1 IPsec SA(보안 연결)를 열거할 때 반환되는 정보를 저장하는 데 사용됩니다. (IPSEC_SA_DETAILS1)
IPSEC_SA_ENUM_TEMPLATE0 IPsec SA(보안 연결)의 열거를 제한하는 데 사용되는 템플릿을 지정합니다.
IPSEC_SA_IDLE_TIMEOUT0 IPsec 정책의 SA(보안 연결) 유휴 시간 제한입니다.
IPSEC_SA_LIFETIME0 IPsec SA(보안 연결)의 수명을 초/킬로바이트/패킷으로 저장합니다.
IPSEC_SA_TRANSFORM0 IPsec 빠른 모드 정책에 IPsec SA(보안 연결) 변환을 저장하는 데 사용됩니다.
IPSEC_SA0 IPsec SA(보안 연결)에 대한 정보를 저장하는 데 사용됩니다.
IPSEC_STATISTICS0 IPsec 통계 구조의 최상위 수준입니다. (IPSEC_STATISTICS0)
IPSEC_STATISTICS1 IPsec 통계 구조의 최상위 수준입니다. (IPSEC_STATISTICS1)
IPSEC_TOKEN0 IPsec 관련 액세스 토큰에 대한 다양한 정보입니다.
IPSEC_TRAFFIC_STATISTICS0

IPsec 트래픽 통계를 저장합니다. (IPSEC_TRAFFIC_STATISTICS0)
IPSEC_TRAFFIC_STATISTICS1 IPsec 트래픽 통계를 저장합니다. (IPSEC_TRAFFIC_STATISTICS1)
IPSEC_TRAFFIC0 IPsec 트래픽을 설명하는 매개 변수를 지정합니다. (IPSEC_TRAFFIC0)
IPSEC_TRAFFIC1 IPsec 트래픽을 설명하는 매개 변수를 지정합니다. (IPSEC_TRAFFIC1)
IPSEC_TRANSPORT_POLICY0 전송 모드 IPsec에 대한 빠른 모드 협상 정책을 저장합니다. (IPSEC_TRANSPORT_POLICY0)
IPSEC_TRANSPORT_POLICY1 전송 모드 IPsec에 대한 빠른 모드 협상 정책을 저장합니다. (IPSEC_TRANSPORT_POLICY1)
IPSEC_TRANSPORT_POLICY2 전송 모드 IPsec에 대한 빠른 모드 협상 정책을 저장합니다. (IPSEC_TRANSPORT_POLICY2)
IPSEC_TUNNEL_ENDPOINT0 터널 모드 SA의 끝점에 대한 주소 정보를 저장하는 데 사용됩니다.
IPSEC_TUNNEL_ENDPOINTS0 터널 모드 SA의 끝점을 저장하는 데 사용됩니다. (IPSEC_TUNNEL_ENDPOINTS0)
IPSEC_TUNNEL_ENDPOINTS1 터널 모드 SA의 끝점을 저장하는 데 사용됩니다. (IPSEC_TUNNEL_ENDPOINTS1)
IPSEC_TUNNEL_ENDPOINTS2 터널 모드 SA의 끝점을 저장하는 데 사용됩니다. (IPSEC_TUNNEL_ENDPOINTS2)
IPSEC_TUNNEL_POLICY0 터널 모드 IPsec에 대한 빠른 모드 협상 정책을 저장합니다. (IPSEC_TUNNEL_POLICY0)
IPSEC_TUNNEL_POLICY1 터널 모드 IPsec에 대한 빠른 모드 협상 정책을 저장합니다. (IPSEC_TUNNEL_POLICY1)

IPSEC_TUNNEL_POLICY2 터널 모드 IPsec에 대한 빠른 모드 협상 정책을 저장합니다. (IPSEC_TUNNEL_POLICY2)
IPSEC_V4_UDP_ENCAPSULATION0 ESP(보안 페이로드) 캡슐화에 대한 UDP(사용자 데이터그램 프로토콜) 캡슐화 포트를 저장합니다.

열거형

IPSEC_AUTH_TYPE 데이터 원본 인증 및 무결성 보호를 위해 IPsec SA에 사용되는 해시 알고리즘의 유형을 나타냅니다.
IPSEC_CIPHER_TYPE 데이터 개인 정보 보호를 위해 IPsec SA에 사용되는 암호화 알고리즘의 유형을 나타냅니다.
IPSEC_FAILURE_POINT IPsec이 실패한 시점입니다.
IPSEC_PFS_GROUP 빠른 모드 PFS(완벽한 전달 비밀)에 사용해야 하는 Diffie Hellman 알고리즘을 지정합니다.
IPSEC_SA_CONTEXT_EVENT_TYPE0 SA(IPsec 보안 연결) 컨텍스트 변경 이벤트의 유형을 지정합니다.
IPSEC_TOKEN_MODE 토큰을 가져올 수 있는 다양한 IPsec 모드입니다.
IPSEC_TOKEN_PRINCIPAL 액세스 토큰 보안 주체입니다.
IPSEC_TOKEN_TYPE IPsec 토큰 형식입니다.
IPSEC_TRAFFIC_TYPE 설명되는 IPsec 트래픽의 유형을 지정합니다.

IPSEC_TRANSFORM_TYPE IPsec SA(보안 연결) 변환의 형식을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

iptypes.h 헤더

아티클2023. 08. 24.

이 헤더는 IP 도우미에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- IP 도우미

iptypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

FIXED_INFO_W2KSP1 FIXED_INFO 구조에는 컴퓨터의 모든 인터페이스에서 동일한 정보가 포함됩니다.
IP_ADAPTER_ADDRESSES_LH IP_ADAPTER_ADDRESSES_LH 구조체(iptypes.h)는 특정 어댑터의 연결된 주소 목록에 대한 헤더 노드입니다.
IP_ADAPTER_ADDRESSES_XP IP_ADAPTER_ADDRESSES_XP 구조체(iptypes.h)는 특정 어댑터의 연결된 주소 목록에 대한 헤더 노드입니다.
IP_ADAPTER_ANYCAST_ADDRESS_XP 단일 애니캐스트 IP 주소를 특정 어댑터의 연결된 주소 목록에 저장합니다.
IP_ADAPTER_DNS_SERVER_ADDRESS_XP IP_ADAPTER_DNS_SERVER_ADDRESS 구조는 특정 어댑터에 대한 DNS 서버 주소의 연결된 목록에 단일 DNS 서버 주소를 저장합니다.
IP_ADAPTER_DNS_SUFFIX IP_ADAPTER_DNS_SUFFIX 구조체는 DNS 접미사를 특정 어댑터의 연결된 DNS 접미사 목록에 저장합니다.
IP_ADAPTER_GATEWAY_ADDRESS_LH 특정 어댑터의 연결된 게이트웨이 주소 목록에 단일 게이트웨이 주소를 저장합니다.
IP_ADAPTER_INFO

로컬 컴퓨터의 특정 네트워크 어댑터에 대한 정보를 포함합니다.
IP_ADAPTER_MULTICAST_ADDRESS_XP IP_ADAPTER_MULTICAST_ADDRESS 구조체는 특정 어댑터의 연결된 주소 목록에 단일 멀티캐스트 주소를 저장합니다.
IP_ADAPTER_PREFIX_XP IP 주소 접두사를 저장합니다. 나
IP_ADAPTER_UNICAST_ADDRESS_LH IP_ADAPTER_UNICAST_ADDRESS_LH 구조체(iptypes.h)는 단일 유니캐스트 IP 주소를 특정 어댑터의 연결된 IP 주소 목록에 저장합니다.
IP_ADAPTER_UNICAST_ADDRESS_XP IP_ADAPTER_UNICAST_ADDRESS_XP 구조체(iptypes.h)는 단일 유니캐스트 IP 주소를 특정 어댑터의 연결된 IP 주소 목록에 저장합니다.
IP_ADAPTER_WINS_SERVER_ADDRESS_LH 특정 어댑터에 대한 WINS 서버 주소의 연결된 목록에 단일 WINS(Windows Internet Name Service) 서버 주소를 저장합니다.
IP_ADDR_STRING IPv4 주소의 연결된 목록에 있는 노드를 나타냅니다.
IP_ADDRESS_STRING IPv4 주소를 점선 10진수 표기법으로 저장합니다.
IP_INTERFACE_NAME_INFO_W2KSP1 로컬 컴퓨터의 IPv4 인터페이스에 대한 정보를 포함합니다.
IP_PER_ADAPTER_INFO_W2KSP1 IP_PER_ADAPTER_INFO 구조체에는 특정 어댑터와 관련된 정보가 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ipxrtdef.h 헤더

아티클2023. 08. 24.

이 헤더는 라우팅 및 원격 액세스 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 라우팅 및 원격 액세스 서비스

ipxrtdef.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

IPX_IF_INFO IPX_IF_INFO 구조체는 IPX 인터페이스에 대한 정보를 저장합니다.
IPXWAN_IF_INFO IPXWAN_IF_INFO 구조체는 IPX WAN 인터페이스에 대한 관리 상태를 저장합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

iscsidsc.h 헤더

아티클2023. 08. 24.

이 헤더는 iSCSI 검색 라이브러리 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- iSCSI 검색 라이브러리 API

iscsidsc.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AddIScsiConnectionA AddIscsiConnection 함수는 기존 세션에 새 iSCSI 연결을 추가합니다. (ANSI)
AddIScsiConnectionW AddIscsiConnection 함수는 기존 세션에 새 iSCSI 연결을 추가합니다. (유니코드)
AddIScsiSendTargetPortalA AddIscsiSendTargetPortal 함수는 iSCSI 초기자 서비스가 SendTargets 요청을 전송하는 대상 포털 목록에 정적 대상 포털을 추가합니다. (ANSI)
AddIScsiSendTargetPortalW AddIscsiSendTargetPortal 함수는 iSCSI 초기자 서비스가 SendTargets 요청을 전송하는 대상 포털 목록에 정적 대상 포털을 추가합니다. (유니코드)
AddIScsiStaticTargetA AddIscsiStaticTarget 함수는 iSCSI 초기자에서 사용할 수 있는 정적 대상 목록에 대상을 추가합니다. (ANSI)
AddIScsiStaticTargetW AddIscsiStaticTarget 함수는 iSCSI 초기자에서 사용할 수 있는 정적 대상 목록에 대상을 추가합니다. (유니코드)
AddISNSServerA AddIsnsServer 함수는 iSCSI 초기자 서비스가 대상을 검색하는 데 사용하는 iSNS(Internet Storage Name Service) 서버 목록에 새 서버를 추가합니다. (ANSI)

AddISNSServerW AddIsnsServer 함수는 iSCSI 초기자 서비스가 대상을 검색하는 데 사용하는 iSNS(Internet Storage Name Service) 서버 목록에 새 서버를 추가합니다. (유니코드)
AddPersistentIScsiDeviceA AddPersistentIscsiDevice 함수는 볼륨 디바이스 이름, 드라이브 문자 또는 탑재 지점 기호 링크를 iSCSI 영구 바인딩 볼륨 및 디바이스 목록에 추가합니다. (ANSI)
AddPersistentIScsiDeviceW AddPersistentIscsiDevice 함수는 볼륨 디바이스 이름, 드라이브 문자 또는 탑재 지점 기호 링크를 iSCSI 영구 바인딩 볼륨 및 디바이스 목록에 추가합니다. (유니코드)
AddRadiusServerA AddRadiusServer. (ANSI)
AddRadiusServerW AddRadiusServer. (유니코드)
ClearPersistentIScsiDevices ClearPersistentIscsiDevices 함수는 영구적으로 바인딩된 iSCSI 볼륨 목록에서 모든 볼륨 및 디바이스를 제거합니다.
GetDevicesForIScsiSessionA GetDevicesForIscsiSession 함수는 현재 세션과 연결된 디바이스에 대한 정보를 검색합니다. (ANSI)
GetDevicesForIScsiSessionW GetDevicesForIscsiSession 함수는 현재 세션과 연결된 디바이스에 대한 정보를 검색합니다. (유니코드)
GetIScsiIKEInfoA GetIscsiIKEInfo 함수는 IPsec 정책 및 초기자 Host-Bus 어댑터(HBA)와 연결된 설정된 미리 공유된 키 값을 검색합니다. (ANSI)
GetIScsiIKEInfoW GetIscsiIKEInfo 함수는 IPsec 정책 및 초기자 Host-Bus 어댑터(HBA)와 연결된 설정된 미리 공유된 키 값을 검색합니다. (유니코드)
GetIScsiInitiatorNodeNameA

GetIscsiInitiatorNodeName 함수는 로컬 컴퓨터에서 세션을 설정할 때 사용되는 공통 초기자 노드 이름을 검색합니다. (ANSI)
GetIScsiInitiatorNodeNameW GetIscsiInitiatorNodeName 함수는 로컬 컴퓨터에서 세션을 설정할 때 사용되는 공통 초기자 노드 이름을 검색합니다. (유니코드)
GetIScsiSessionListA GetIscsiSessionList 함수는 활성 iSCSI 세션 목록을 검색합니다. (ANSI)
GetIScsiSessionListW GetIscsiSessionList 함수는 활성 iSCSI 세션 목록을 검색합니다. (유니코드)
GetIScsiTargetInformationA GetIscsiTargetInformation 함수는 지정된 대상에 대한 정보를 검색합니다. (ANSI)
GetIScsiTargetInformationW GetIscsiTargetInformation 함수는 지정된 대상에 대한 정보를 검색합니다. (유니코드)
GetIScsiVersionInformation GetIscsiVersionInformation 함수는 초기자 버전에 대한 정보를 검색합니다.
LoginIScsiTargetA LoginIscsiTarget 함수는 표시된 대상과 함께 전체 추천 로그인 세션을 설정합니다. (ANSI)
LoginIScsiTargetW LoginIscsiTarget 함수는 표시된 대상과 함께 전체 추천 로그인 세션을 설정합니다. (유니코드)
LogoutIScsiTarget LogoutIscsiTarget 루틴은 지정된 로그인 세션을 닫습니다.
RefreshIScsiSendTargetPortalA RefreshIscsiSendTargetPortal 함수는 iSCSI 초기자 서비스에 지정된 대상 포털을 사용하여 검색 세션을 설정하고 SendTargets 요청을 전송하여 iSCSI 초기자 서비스에 대해 검색된 대상 목록을 새로 고치도록 지시합니다. (ANSI)
RefreshIScsiSendTargetPortalW RefreshIscsiSendTargetPortal 함수는 iSCSI 초기자 서비스에 지정된 대상 포털을 사용하여 검색 세

션을 설정하고 SendTargets 요청을 전송하여 iSCSI 초기자 서비스에 대해 검색된 대상 목록을 새로 고치도록 지시합니다. (유니코드)
RefreshISNSServerA RefreshIsnsServer 함수는 iSCSI 초기자 서비스에 표시된 iSNS(Internet Storage Name Service) 서버를 쿼리하여 iSCSI 초기자 서비스에 대해 검색된 대상 목록을 새로 고치도록 지시합니다. (ANSI)
RefreshISNSServerW RefreshIsnsServer 함수는 iSCSI 초기자 서비스에 표시된 iSNS(Internet Storage Name Service) 서버를 쿼리하여 iSCSI 초기자 서비스에 대해 검색된 대상 목록을 새로 고치도록 지시합니다. (유니코드)
RemoveIScsiConnection RemoveIscsiConnection 함수는 활성 세션에서 연결을 제거합니다.
RemoveIScsiPersistentTargetA RemoveIscsiPersistentTarget 함수는 지정된 하드웨어 초기자 HBA(호스트 버스 어댑터), 초기자 포트 및 대상 포털에 대한 영구 로그인을 제거합니다. (ANSI)
RemoveIScsiPersistentTargetW RemoveIscsiPersistentTarget 함수는 지정된 하드웨어 초기자 HBA(호스트 버스 어댑터), 초기자 포트 및 대상 포털에 대한 영구 로그인을 제거합니다. (유니코드)
RemoveIScsiSendTargetPortalA RemoveIscsiSendTargetPortal 함수는 iSCSI 초기자 서비스가 대상 검색을 위해 SendTargets 요청을 보내는 포털 목록에서 포털을 제거합니다. (ANSI)
RemoveIScsiSendTargetPortalW RemoveIscsiSendTargetPortal 함수는 iSCSI 초기자 서비스가 대상 검색을 위해 SendTargets 요청을 보내는 포털 목록에서 포털을 제거합니다. (유니코드)
RemoveIScsiStaticTargetA RemoveIscsiStaticTarget 함수는 컴퓨터에서 사용할 수 있는 정적 대상 목록에서 대상을 제거합니다. (ANSI)
RemoveIScsiStaticTargetW RemoveIscsiStaticTarget 함수는 컴퓨터에서 사용할 수 있는 정적 대상 목록에서 대상을 제거합니다. (유니코드)
RemoveISNSServerA

RemoveIsnsServer 함수는 iSCSI 초기자 서비스에서 대상을 검색하는 데 사용하는 iSNS(Internet Storage Name Service) 서버 목록에서 서버를 제거합니다. (ANSI)
RemoveISNSServerW RemoveIsnsServer 함수는 iSCSI 초기자 서비스에서 대상을 검색하는 데 사용하는 iSNS(Internet Storage Name Service) 서버 목록에서 서버를 제거합니다. (유니코드)
RemovePersistentIScsiDeviceA RemovePersistentIscsiDevice 함수는 영구적으로 바인딩된 iSCSI 볼륨 목록에서 디바이스 또는 볼륨을 제거합니다. (ANSI)
RemovePersistentIScsiDeviceW RemovePersistentIscsiDevice 함수는 영구적으로 바인딩된 iSCSI 볼륨 목록에서 디바이스 또는 볼륨을 제거합니다. (유니코드)
RemoveRadiusServerA RemoveRadiusServer 함수는 iSCSI 초기자가 구성된 RADIUS 서버 목록에서 RADIUS(원격 인증 전화 접속 사용자 서비스) 서버 항목을 제거합니다. (ANSI)
RemoveRadiusServerW RemoveRadiusServer 함수는 iSCSI 초기자가 구성된 RADIUS 서버 목록에서 RADIUS(원격 인증 전화 접속 사용자 서비스) 서버 항목을 제거합니다. (유니코드)
ReportActiveIScsiTargetMappingsA ReportActiveIscsiTargetMappings 함수는 컴퓨터의 모든 초기자에 대해 현재 활성화된 대상 매핑을 검색합니다. (ANSI)
ReportActiveIScsiTargetMappingsW ReportActiveIscsiTargetMappings 함수는 컴퓨터의 모든 초기자에 대해 현재 활성화된 대상 매핑을 검색합니다. (유니코드)
ReportIScsiInitiatorListA ReportIscsiInitiatorList 함수는 컴퓨터에서 실행 중인 초기자 호스트 버스 어댑터 목록을 검색합니다. (ANSI)
ReportIScsiInitiatorListW ReportIscsiInitiatorList 함수는 컴퓨터에서 실행 중인 초기자 호스트 버스 어댑터 목록을 검색합니다. (유니코드)
ReportIScsiPersistentLoginsA

ReportIscsiPersistentLogins 함수는 영구 로그인 대상 목록을 검색합니다. (ANSI)
ReportIScsiPersistentLoginsW ReportIscsiPersistentLogins 함수는 영구 로그인 대상 목록을 검색합니다. (유니코드)
ReportIScsiSendTargetPortalsA ReportIscsiSendTargetPortals 함수는 iSCSI 초기자 서비스가 SendTarget 요청을 사용하여 자동 검색을 수행하는 데 사용하는 대상 포털 목록을 검색합니다. (ANSI)
ReportIScsiSendTargetPortalsExA ReportIscsiSendTargetPortalsEx 함수는 iSCSI 초기자 서비스가 SendTarget 요청을 사용하여 자동 검색을 수행하는 데 사용하는 정적 대상 포털 목록을 검색합니다. (ANSI)
ReportIScsiSendTargetPortalsExW ReportIscsiSendTargetPortalsEx 함수는 iSCSI 초기자 서비스가 SendTarget 요청을 사용하여 자동 검색을 수행하는 데 사용하는 정적 대상 포털 목록을 검색합니다. (유니코드)
ReportIScsiSendTargetPortalsW ReportIscsiSendTargetPortals 함수는 iSCSI 초기자 서비스가 SendTarget 요청을 사용하여 자동 검색을 수행하는 데 사용하는 대상 포털 목록을 검색합니다. (유니코드)
ReportIScsiTargetPortalsA ReportIscsiTargetPortals 함수는 iSCSI 초기자 서비스에서 검색한 대상 포털 정보를 검색합니다. (ANSI)
ReportIScsiTargetPortalsW ReportIscsiTargetPortals 함수는 iSCSI 초기자 서비스에서 검색한 대상 포털 정보를 검색합니다. (유니코드)
ReportIScsiTargetsA ReportIscsiTargets 함수는 iSCSI 초기자 서비스에서 검색한 대상 목록을 검색하고 iSCSI 초기자 서비스에 목록을 새로 고치도록 지시할 수도 있습니다. (ANSI)
ReportIScsiTargetsW ReportIscsiTargets 함수는 iSCSI 초기자 서비스에서 검색한 대상 목록을 검색하고 iSCSI 초기자 서비스에 목록을 새로 고치도록 지시할 수도 있습니다. (유니코드)
ReportISNSServerListA

ReportIsnsServerList 함수는 iSCSI 초기자 서비스가 검색된 대상에 대해 쿼리하는 iSNS(Internet Storage Name Service) 서버 목록을 검색합니다. (ANSI)
ReportISNSServerListW ReportIsnsServerList 함수는 iSCSI 초기자 서비스가 검색된 대상에 대해 쿼리하는 iSNS(Internet Storage Name Service) 서버 목록을 검색합니다. (유니코드)
ReportPersistentIScsiDevicesA ReportPersistentIscsiDevices 함수는 영구적으로 바인딩된 볼륨 및 디바이스 목록을 검색합니다. (ANSI)
ReportPersistentIScsiDevicesW ReportPersistentIscsiDevices 함수는 영구적으로 바인딩된 볼륨 및 디바이스 목록을 검색합니다. (유니코드)
ReportRadiusServerListA ReportRadiusServerList 함수는 iSCSI 초기자 서비스가 인증 중에 사용하는 RADIUS(원격 인증 전화 접속 서비스) 서버 목록을 검색합니다. (ANSI)
ReportRadiusServerListW ReportRadiusServerList 함수는 iSCSI 초기자 서비스가 인증 중에 사용하는 RADIUS(원격 인증 전화 접속 서비스) 서버 목록을 검색합니다. (유니코드)
SendScsiInquiry SendScsiInquiry 함수는 SCSI INQUIRY 명령을 지정된 대상에 보냅니다.
SendScsiReadCapacity SendScsiReadCapacity 함수는 SCSI READ CAPACITY 명령을 표시된 대상에 보냅니다.
SendScsiReportLuns SendScsiReportLuns 함수는 SCSI REPORT LUNS 명령을 지정된 대상으로 보냅니다.
SetIScsiGroupPresharedKey SetIscsiGroupPresharedKey 함수는 컴퓨터의 모든 초기자에 대해 기본 그룹 사전 공유 키를 설정합니다.
SetIScsiIKEInfoA SetIscsiIKEInfo 함수는 iSCSI 연결을 수행할 때 사용할 지정된 초기자에 대한 IPsec 정책 및 미리 공유된 키를 설정합니다. (ANSI)

SetIScsiIKEInfoW SetIscsiIKEInfo 함수는 iSCSI 연결을 수행할 때 사용할 지정된 초기자에 대한 IPsec 정책 및 미리 공유된 키를 설정합니다. (유니코드)
SetIScsiInitiatorCHAPSharedSecret SetIscsiInitiatorCHAPSharedSecret 함수는 컴퓨터의 모든 초기자에 대한 기본 CHAP(Challenge Handshake Authentication Protocol) 공유 비밀을 설정합니다.
SetIScsiInitiatorNodeNameA SetIscsiInitiatorNodeName 함수는 컴퓨터의 초기자 노드 이름을 설정합니다. 이 이름은 다른 노드와 통신하는 컴퓨터의 모든 초기자 노드에서 활용됩니다. (ANSI)
SetIScsiInitiatorNodeNameW SetIscsiInitiatorNodeName 함수는 컴퓨터의 초기자 노드 이름을 설정합니다. 이 이름은 다른 노드와 통신하는 컴퓨터의 모든 초기자 노드에서 활용됩니다. (유니코드)
SetIScsiInitiatorRADIUSSharedSecret SetIscsiInitiatorRADIUSSharedSecret 함수는 RADIUS(원격 인증 전화 접속 사용자 서비스) 공유 비밀을 설정합니다.
SetIScsiTunnelModeOuterAddressA SetIscsiTunnelModeOuterAddress 함수는 지정된 초기자 HBA(호스트 버스 어댑터)가 지정된 포트를 통해 IPsec 터널 모드로 통신할 때 사용하는 터널 모드 외부 주소를 설정합니다. (ANSI)
SetIScsiTunnelModeOuterAddressW SetIscsiTunnelModeOuterAddress 함수는 지정된 초기자 HBA(호스트 버스 어댑터)가 지정된 포트를 통해 IPsec 터널 모드로 통신할 때 사용하는 터널 모드 외부 주소를 설정합니다. (유니코드)
SetupPersistentIScsiDevices SetupPersistentIscsiDevices 함수는 컴퓨터에 연결된 iSCSI 대상에 할당된 디바이스 및 볼륨 목록을 빌드하고 iSCSI 초기자 서비스의 비휘발성 캐시에 이 목록을 저장합니다.

구조체

IKE_AUTHENTICATION_INFORMATION

IKE_AUTHENTICATION_INFORMATION 구조에는 두 키 관리 디먼 간에 보안 채널을 설정하는 데 사용되는 IKE(인터넷 키 교환) 인증 정보가 포함되어 있습니다.
IKE_AUTHENTICATION_PRESHARED_KEY IKE_AUTHENTICATION_PRESHARED_KEY 구조에는 IKE(인터넷 키 교환) 프로토콜에 사용되는 미리 공유된 키에 대한 정보가 포함되어 있습니다.
ISCSI_CONNECTION_INFOA ISCSI_CONNECTION_INFO 구조에는 연결에 대한 정보가 포함됩니다. (ANSI)
ISCSI_CONNECTION_INFOW ISCSI_CONNECTION_INFO 구조에는 연결에 대한 정보가 포함됩니다. (유니코드)
ISCSI_DEVICE_ON_SESSIONA ISCSI_DEVICE_ON_SESSION 구조는 iSCSI 로그인 세션과 연결된 디바이스를 식별하는 여러 메서드를 지정합니다. (ANSI)
ISCSI_DEVICE_ON_SESSIONW ISCSI_DEVICE_ON_SESSION 구조는 iSCSI 로그인 세션과 연결된 디바이스를 식별하는 여러 메서드를 지정합니다. (유니코드)
ISCSI_LOGIN_OPTIONS ISCSI_LOGIN_OPTIONS 구조체는 초기자에서 로그인 세션의 특성을 지정하는 데 사용됩니다.
ISCSI_SESSION_INFOA ISCSI_SESSION_INFO. (ANSI)
ISCSI_SESSION_INFOW ISCSI_SESSION_INFO. (유니코드)
ISCSI_TARGET_MAPPINGA ISCSI_TARGET_MAPPING. (ANSI)
ISCSI_TARGET_MAPPINGW ISCSI_TARGET_MAPPING. (유니코드)
ISCSI_TARGET_PORTAL_GROUPA ISCSI_TARGET_PORTAL_GROUP. (ANSI)

ISCSI_TARGET_PORTAL_GROUPW ISCSI_TARGET_PORTAL_GROUP. (유니코드)
ISCSI_TARGET_PORTAL_INFO_EXA ISCSI_TARGET_PORTAL_INFO_EX 구조에는 대상 포털에 대한 로그인 자격 증명에 대한 정보가 포함됩니다. (ANSI)
ISCSI_TARGET_PORTAL_INFO_EXW ISCSI_TARGET_PORTAL_INFO_EX 구조에는 대상 포털에 대한 로그인 자격 증명에 대한 정보가 포함됩니다. (유니코드)
ISCSI_TARGET_PORTAL_INFOA ISCSI_TARGET_PORTAL_INFO 구조에는 대상 포털에 대한 정보가 포함됩니다. (ANSI)
ISCSI_TARGET_PORTAL_INFOW ISCSI_TARGET_PORTAL_INFO 구조에는 대상 포털에 대한 정보가 포함됩니다. (유니코드)
ISCSI_TARGET_PORTALA ISCSI_TARGET_PORTAL 구조에는 포털에 대한 정보가 포함됩니다. (ANSI)
ISCSI_TARGET_PORTALW ISCSI_TARGET_PORTAL 구조에는 포털에 대한 정보가 포함됩니다. (유니코드)
ISCSI_UNIQUE_SESSION_ID ISCSI_UNIQUE_SESSION_ID 구조체는 세션을 고유하게 식별하는 데이터를 포함하는 불투명 엔터티입니다.
ISCSI_VERSION_INFO ISCSI_VERSION_INFO 구조에는 iSCSI 소프트웨어 초기자 및 초기자 서비스의 버전 및 빌드 번호가 포함됩니다.
PERSISTENT_ISCSI_LOGIN_INFOA PERSISTENT_ISCSI_LOGIN_INFO 구조에는 컴퓨터가 부팅된 후 Microsoft iSCSI 초기자 서비스에서 설정한 로그인 세션을 설명하는 정보가 포함되어 있습니다. (ANSI)
PERSISTENT_ISCSI_LOGIN_INFOW PERSISTENT_ISCSI_LOGIN_INFO 구조에는 컴퓨터가 부팅된 후 Microsoft iSCSI 초기자 서비스에서 설정한 로그인 세션을 설명하는 정보가 포함되어 있습니다. (유니코드)

SCSI_LUN_LIST SCSI_LUN_LIST 구조체는 대상 디바이스와 연결된 LUN(논리 단위 번호) 목록을 생성하는 데 사용됩니다.

열거형

IKE_AUTHENTICATION_METHOD IKE_AUTHENTICATION_METHOD 열거형은 IKE(인터넷 키 교환) 인증 방법의 유형을 나타냅니다.
ISCSI_AUTH_TYPES ISCSI_AUTH_TYPES 열거형은 사용된 인증 방법의 유형을 나타냅니다.
ISCSI_DIGEST_TYPES ISCSI_DIGEST_TYPES 열거형은 다이제스트 유형을 나타냅니다.
TARGET_INFORMATION_CLASS TARGET_INFORMATION_CLASS 열거형은 GetIScsiTargetInformation 함수가 검색하는 지정된 대상 디바이스에 대한 정보를 지정합니다.
TARGETPROTOCOLTYPE TARGETPROTOCOLTYPE 열거형은 초기자가 대상과 통신하는 데 사용해야 하는 프로토콜 유형을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

isolatedapplauncher.h 헤더

아티클2023. 12. 19.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

> ⚠ **경고**
>
> 더 이상 사용되지 않는 API입니다.

> ⓘ **참고**
>
> 이제 WDAG(Windows Defender Application Guard)가 MDAG(Microsoft Defender Application Guard)입니다. WDAG 이름은 더 이상 사용되지 않지만 일부 API에서는 계속 사용됩니다.

isolatedapplauncher.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

테이블 확장

IIsolatedAppLauncher 앱이 MICROSOFT DEFENDER APPLICATION GUARD(MDAG) 컨테이너(VM 컨테이너 환경)에서 실행 중인지 여부를 확인할 수 있습니다.
IIsolatedProcessLauncher 앱에서 프로세스가 MICROSOFT DEFENDER APPLICATION GUARD(MDAG) 컨테이너(VM 컨테이너 환경)에서 실행 중인지 여부를 확인할 수 있습니다.
IIsolatedProcessLauncher2 앱에서 프로세스가 MICROSOFT DEFENDER APPLICATION GUARD(MDAG) 컨테이너(VM 컨테이너 환경)에서 실행 중인지 여부를 확인할 수 있습니다.

Functions

테이블 확장

IsProcessInIsolatedContainer 프로세스가 격리된 컨테이너에서 실행되고 있는지 확인합니다.
IsProcessInWDAGContainer 프로세스가 MDAG(Microsoft Defender Application Guard) 컨테이너에서 실행되고 있는지 확인합니다.

구조체

테이블 확장

IsolatedAppLauncherTelemetryParameters 컨테이너 내에서 앱을 시작할 때 사용할 원격 분석 매개 변수를 제공하는 구조체입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

isolatedwindowsenvironmentutils.h 헤더

아티클2023. 10. 04.

이러한 API는 호스트 및 격리된 환경의 특성을 쿼리하는 데 사용됩니다. 애플리케이션에서 실행 중인 실행 환경(호스트 또는 격리된 환경)을 찾는 신뢰할 수 있는 메커니즘이 필요합니다. 이를 통해 애플리케이션은 격리된 환경 내에서 사용할 수 있는 시스템/사용자/앱 리소스 및 앱 기능을 기반으로 사용자 환경을 오케스트레이션할 수 있습니다.

isolatedwindowsenvironmentutils.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

IsCrossIsolatedEnvironmentClipboardContent IsCrossIsolatedEnvironmentClipboardContent는 붙여넣는 콘텐츠가 MDAG(Microsoft Defender Application Guard) 경계의 다른 쪽에서 제공되었는지 확인하기 위해 붙여넣기 실패를 감지한 후 호출됩니다.
IsProcessInIsolatedWindowsEnvironment 애플리케이션이 실행 중인 실행 환경을 결정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

isysmon.h 헤더

아티클2023. 08. 24.

이 헤더는 시스템 모니터에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- System Monitor(시스템 모니터)

isysmon.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

열거형

DataSourceTypeConstants 성능 카운터 데이터의 원본을 결정합니다.
DisplayTypeConstants 성능 카운터 데이터가 그래프로 표시되는 방식(예: 선 그래프 또는 히스토그램)을 결정합니다.
ReportValueTypeConstants 히스토그램 및 보고서 뷰 그래프가 샘플링된 마지막 값인지 아니면 샘플링 기간의 값(예: 평균 또는 최소값)을 사용하여 계산된 값을 그래프로 표시할지 확인합니다.
SysmonBatchReason 시스템 모니터를 잠그는 이유를 정의합니다.
SysmonDataType 그래프의 지정된 데이터 요소에서 반환할 데이터 형식을 결정합니다.
SysmonFileType 카운터 데이터가 파일에 저장되는 형식을 결정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

iwscapi.h 헤더

아티클2023. 08. 24.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

iwscapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWscProduct 개별 공급자가 Windows 보안 Center와 상호 작용할 수 있도록 제품 정보를 가져오는 방법을 제공합니다.
IWSCProductList 컴퓨터에 설치된 선택한 공급자 유형에 대한 제품 정보를 수집하는 메서드를 제공합니다.

열거형

WSC_SECURITY_PRODUCT_STATE Windows 보안 센터에서 사용할 수 있는 보안 제품의 현재 상태를 정의합니다.
WSC_SECURITY_SIGNATURE_STATUS 보안 제품의 현재 버전 상태 Windows 보안 Center에 보고합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

jobapi.h 헤더

아티클2023. 08. 24.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

jobapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

IsProcessInJob 프로세스가 지정된 작업에서 실행 중인지 여부를 확인합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

jobapi2.h 헤더

아티클2023. 08. 24.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

jobapi2.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AssignProcessToJobObject 기존 작업 개체에 프로세스를 할당합니다.
CreateJobObjectW 작업 개체를 만들거나 엽니다. (CreateJobObjectW)
FreeMemoryJobObject 작업 개체와 관련된 함수가 할당한 메모리를 해제합니다. 메모리를 할당하는 작업 개체와 관련된 함수에는 QueryIoRateControlInformationJobObject가 포함됩니다.
OpenJobObjectW 기존 작업 개체를 엽니다. (OpenJobObjectW)
QueryInformationJobObject 작업 개체에서 제한 및 작업 상태 정보를 검색합니다.
QueryIoRateControlInformationJobObject 작업 개체의 I/O 속도 컨트롤에 대한 정보를 가져옵니다.
SetInformationJobObject 작업 개체에 대한 제한을 설정합니다.
SetIoRateControlInformationJobObject 작업 개체에 대한 I/O 제한을 설정합니다.
TerminateJobObject

현재 작업과 연결된 모든 프로세스를 종료합니다.

구조체

JOBOBJECT_IO_RATE_CONTROL_INFORMATION 작업에 대한 I/O 속도를 제어하는 데 사용되는 정보를 포함합니다. 이 구조체는 SetIoRateControlInformationJobObject 및 QueryIoRateControlInformationJobObject 함수에서 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

joystickapi.h 헤더

아티클2023. 08. 24.

이 헤더는 Windows 멀티미디어에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 멀티미디어

joystickapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

joyConfigChanged joyConfigChanged 함수는 조이스틱 드라이버에 구성이 변경되었으며 레지스트리에서 다시 로드되어야 한다고 알릴 수 있습니다.
joyGetDevCaps joyGetDevCaps 함수(joystickapi.h)는 조이스틱을 쿼리하여 기능을 확인합니다. uJoyID는 쿼리할 조이스틱의 식별자입니다.
joyGetDevCapsA joyGetDevCaps 함수는 조이스틱을 쿼리하여 기능을 확인합니다. (joyGetDevCapsA)
joyGetDevCapsW joyGetDevCapsW(유니코드) 함수는 조이스틱을 쿼리하여 포괄적인 기능을 확인합니다.
joyGetNumDevs joyGetNumDevs 함수는 조이스틱 드라이버에 지원하는 조이스틱 수를 쿼리합니다.
joyGetPos joyGetPos 함수는 조이스틱의 위치와 단추 상태 쿼리합니다.
joyGetPosEx joyGetPosEx 함수는 조이스틱의 위치와 단추 상태 쿼리합니다.
joyGetThreshold joyGetThreshold 함수는 조이스틱에서 현재 이동 임계값을 쿼리합니다.

joyReleaseCapture joyReleaseCapture 함수는 지정된 캡처된 조이스틱을 해제합니다.
joySetCapture joySetCapture 함수는 지정된 창으로 메시지를 보내 조이스틱을 캡처합니다.
joySetThreshold joySetThreshold 함수는 조이스틱의 이동 임계값을 설정합니다.

구조체

조이캡스 JOYCAPS(joystickapi.h) 구조에는 조이스틱 기능에 대한 중요한 정보가 포함되어 있습니다.
JOYCAPSA JOYCAPS 구조에는 조이스틱 기능에 대한 정보가 포함되어 있습니다. (JOYCAPSA)
JOYCAPSW JOYCAPSW(유니코드)(조이스틱api.h) 구조에는 조이스틱 기능에 대한 정보가 포함되어 있습니다.
JOYINFO JOYINFO 구조체에는 조이스틱 위치 및 단추 상태에 대한 정보가 포함되어 있습니다.
JOYINFOEX JOYINFOEX 구조체에는 조이스틱 위치, 관점 위치 및 단추 상태에 대한 확장된 정보가 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

keycredmgr.h 헤더

아티클2023. 08. 24.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

keycredmgr.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

KeyCredentialManagerFreeInformation KeyCredentialManagerGetInformation 호출에서 KeyCredentialManagerInfo 포인터 변수를 해제하는 API입니다.
KeyCredentialManagerGetInformation 사용자 등록의 고유 식별자를 가져오는 API입니다.
KeyCredentialManagerGetOperationErrorStates 를 호출하여 이전에 작업이 성공할지 여부를 확인하기 위한 필수 조건 API입니다.
KeyCredentialManagerShowUIOperation 요청된 WHFB 작업을 수행하는 API입니다.

구조체

KeyCredentialManagerInfo KeyCredentialManagerGetInformation에서 반환된 데이터 구조입니다.

열거형

KeyCredentialManagerOperationErrorStates KeyCredentialManagerGetOperationErrorStates 함수에서 플래그로 반환된 오류 상태의 열거형입니다.
KeyCredentialManagerOperationType KeyCredentialManagerShowUIOperation에 전달되는 작업 열거형 값입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ksopmapi.h 헤더

아티클2023. 08. 24.

이 헤더는 Microsoft Media Foundation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft Media Foundation

ksopmapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

OPM_ENCRYPTED_INITIALIZATION_PARAMETERS OPM_ENCRYPTED_INITIALIZATION_PARAMETERS(ksopmapi.h) 구조에는 OPM(출력 보호 관리자) 세션에 대한 초기화 매개 변수가 포함되어 있습니다.
OPM_GET_CODEC_INFO_INFORMATION OPM_GET_CODEC_INFO_INFORMATION(ksopmapi.h) 구조체에는 OPM_GET_CODEC_INFO 쿼리의 결과가 포함됩니다.
OPM_GET_CODEC_INFO_PARAMETERS OPM_GET_CODEC_INFO_PARAMETERS(ksopmapi.h) 구조체에는 OPM_GET_CODEC_INFO 명령에 대한 정보가 포함되어 있습니다.
OPM_GET_INFO_PARAMETERS OPM_GET_INFO_PARAMETERS(ksopmapi.h) 구조체에는 IOPMVideoOutput::GetInformation 메서드에 대한 매개 변수가 포함되어 있습니다.
OPM_OMAC OPM_OMAC(ksopmapi.h) 구조에는 OPM(출력 보호 관리자) 메시지에 대한 MAC(메시지 인증 코드)가 포함되어 있습니다.
OPM_RANDOM_NUMBER OPM_RANDOM_NUMBER(ksopmapi.h) 구조체에는 OPM(출력 보호 관리자)에 사용할 128비트 난수가 포함되어 있습니다.
OPM_REQUESTED_INFORMATION

OPM_REQUESTED_INFORMATION(ksopmapi.h) 구조에는 OPM(출력 보호 관리자) 상태 요청의 결과가 포함됩니다.
OPM_STANDARD_INFORMATION OPM_STANDARD_INFORMATION(ksopmapi.h) 구조에는 OPM(출력 보호 관리자) 상태 요청의 결과가 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ktmtypes.h 헤더

아티클2023. 08. 24.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지
- 커널 모드 드라이버 참조

ktmtypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

TRANSACTION_NOTIFICATION 트랜잭션 알림과 연결된 데이터를 포함합니다.
TRANSACTION_NOTIFICATION_RECOVERY_ARGUMENT 복구할 트랜잭션을 나타냅니다. 이 구조는 복구 알림과 함께 전송됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ktmw32.h 헤더

아티클2023. 08. 24.

이 헤더는 데이터 액세스 및 스토리지에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지

ktmw32.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CommitComplete RM(리소스 관리자)이 TM(트랜잭션 관리자)에서 요청한 트랜잭션 커밋을 완료했음을 나타냅니다.
CommitEnlistment 이 인리스트먼트 핸들과 연결된 트랜잭션을 커밋합니다. 이 함수는 통신 리소스 관리자(뛰어난 트랜잭션 관리자라고도 함)에서 사용됩니다.
CommitTransaction 지정된 트랜잭션을 커밋할 것을 요청합니다. (CommitTransaction)
CommitTransactionAsync 지정된 트랜잭션을 커밋할 것을 요청합니다. (CommitTransactionAsync)
CreateEnlistment 인리스트먼트를 만들고, 초기 상태를 설정하고, 지정된 액세스를 사용하여 인리스트먼트에 대한 핸들을 엽니다.
CreateResourceManager 새 RM(리소스 관리자) 개체를 만들고 RM을 TM(트랜잭션 관리자)과 연결합니다.
CreateTransaction 새 트랜잭션 개체를 만듭니다.
CreateTransactionManager 새 TM(트랜잭션 관리자) 개체를 만들고 지정된 액세스 권한이 있는 핸들을 반환합니다.

GetCurrentClockTransactionManager 트랜잭션 관리자에서 가상 클록 값을 가져옵니다.
GetEnlistmentId 지정된 인리스트먼트의 식별자(ID)를 가져옵니다.
GetEnlistmentRecoveryInformation KTM에서 불투명한 복구 데이터 구조를 검색합니다.
GetNotificationResourceManager RM(리소스 관리자)에 대한 알림을 요청하고 받습니다. 이 함수는 트랜잭션 상태가 변경될 때 RM 레지스터에서 알림을 받는 데 사용됩니다.
GetNotificationResourceManagerAsync RM(리소스 관리자)에 대한 비동기 알림을 요청하고 받습니다. 이 함수는 트랜잭션 상태가 변경될 때 RM 레지스터에서 알림을 받는 데 사용됩니다.
GetTransactionId 지정된 트랜잭션의 식별자(ID)를 가져옵니다.
GetTransactionInformation 지정된 트랜잭션에 대해 요청된 정보를 반환합니다.
GetTransactionManagerId 지정된 트랜잭션 관리자의 식별자를 가져옵니다.
OpenEnlistment 기존 인리스트먼트 개체를 열고 인리스트먼트에 대한 핸들을 반환합니다.
OpenResourceManager 기존 RM(리소스 관리자)을 엽니다.
OpenTransaction 기존 트랜잭션을 엽니다.
OpenTransactionManager 기존 트랜잭션 관리자를 엽니다. (OpenTransactionManager)

OpenTransactionManagerById 기존 트랜잭션 관리자를 엽니다. (OpenTransactionManagerById)
PrepareComplete RM(리소스 관리자)이 지정된 트랜잭션에 대해 커밋 또는 중단 작업이 성공하도록 보장하는 데 필요한 모든 처리를 완료했음을 나타냅니다.
PrepareEnlistment 이 인리스트먼트 핸들과 연결된 트랜잭션을 준비합니다. 이 함수는 통신 리소스 관리자(뛰어난 트랜잭션 관리자라고도 함)에서 사용됩니다.
PrePrepareComplete 이제 다른 리소스 관리자가 준비 작업을 시작할 수 있도록 이 리소스 관리자가 준비 작업을 완료했음을 알리는 신호입니다.
PrePrepareEnlistment 이 인리스트먼트 핸들과 연결된 트랜잭션을 미리 준비합니다. 이 함수는 통신 리소스 관리자(뛰어난 트랜잭션 관리자라고도 함)에서 사용됩니다.
ReadOnlyEnlistment 지정된 인리스트먼트를 읽기 전용 인리스트먼트로 변환하도록 요청합니다. 읽기 전용 인리스트먼트는 트랜잭션 결과에 참여할 수 없으며 복구를 위해 지속적으로 기록되지 않습니다.
RecoverEnlistment 인리스트먼트의 상태를 복구합니다.
RecoverResourceManager 로그 파일에서 리소스 관리자의 상태를 복구합니다.
RecoverTransactionManager 로그 파일에서 트랜잭션 관리자의 상태를 복구합니다.
RenameTransactionManager TM(트랜잭션 관리자) 개체의 이름을 바꿉니다. 이 함수는 명명된 TM 핸들에서만 사용할 수 있습니다.
RollbackComplete RM(리소스 관리자)이 트랜잭션 롤백을 성공적으로 완료했음을 나타냅니다.

RollbackEnlistment 인리스트먼트와 연결된 지정된 트랜잭션을 롤백합니다. 이 함수는 읽기 전용 인리스트먼트에 대해 호출할 수 없습니다.
RollbackTransaction 지정된 트랜잭션을 롤백할 것을 요청합니다.
RollbackTransactionAsync 지정된 트랜잭션을 롤백할 것을 요청합니다. 이 함수는 비동기적으로 반환됩니다.
RollforwardTransactionManager 지정된 가상 클록 값으로만 정보를 복구합니다.
SetEnlistmentRecoveryInformation KTM에서 복구 데이터의 불투명한 사용자 정의 구조를 설정합니다.
SetResourceManagerCompletionPort 지정된 I/O 완료 포트를 지정된 RM(리소스 관리자)과 연결합니다. 이 포트는 RM에 대한 모든 알림을 받습니다.
SetTransactionInformation 지정된 트랜잭션에 대한 트랜잭션 정보를 설정합니다.
SinglePhaseReject RM(리소스 관리자)이 단일 단계 요청을 거부했음을 나타냅니다. TM(트랜잭션 관리자)이 이 호출을 받으면 2단계 커밋을 시작하고 등록된 모든 RM에 준비 요청을 보냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

l2cmn.h 헤더

아티클2023. 08. 24.

이 헤더는 네트워크 드라이버, Windows Vista 이상에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 드라이버, Windows Vista 이상

l2cmn.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

L2_NOTIFICATION_DATA 중요 네이티브 802.11 무선 LAN 인터페이스는 Windows 10 이상에서 더 이상 사용되지 않습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

libloaderapi.h 헤더

아티클2023. 05. 18.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 애플리케이션에 대한 국제화
- 메뉴 및 기타 리소스
- 시스템 서비스

libloaderapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AddDllDirectory 프로세스 DLL 검색 경로에 디렉터리를 추가합니다.
DisableThreadLibraryCalls 지정된 DLL(동적 연결 라이브러리)에 대한 DLL_THREAD_ATTACH 및 DLL_THREAD_DETACH 알림을 사용하지 않도록 설정합니다.
EnumResourceLanguagesExA 지정된 이진 모듈과 연결된 지정된 형식 및 이름의 언어별 리소스를 열거합니다. 열거형에 대한 제어를 더 많이 허용하여 EnumResourceLanguages를 확장합니다. (ANSI)
EnumResourceLanguagesExW 지정된 이진 모듈과 연결된 지정된 형식 및 이름의 언어별 리소스를 열거합니다. 열거형에 대한 제어를 더 많이 허용하여 EnumResourceLanguages를 확장합니다. (유니코드)
EnumResourceNamesA 이진 모듈 내에서 지정된 형식의 리소스를 열거합니다. (ANSI)
EnumResourceNamesExA 지정된 이진 모듈과 연결된 지정된 형식의 리소스를 열거합니다. 검색에는 LN 파일과 연결된 .mui 파일이 모두 포함되거나 여러 가지 방법으로 제한될 수 있습니다. (ANSI)

EnumResourceNamesExW 지정된 이진 모듈과 연결된 지정된 형식의 리소스를 열거합니다. 검색에는 LN 파일과 연결된 .mui 파일이 모두 포함되거나 여러 가지 방법으로 제한될 수 있습니다. (유니코드)
EnumResourceNamesW 이진 모듈 내에서 지정된 형식의 리소스를 열거합니다. (유니코드)
EnumResourceTypesExA 지정된 이진 모듈과 연결된 리소스 종류를 열거합니다. (ANSI)
EnumResourceTypesExW 지정된 이진 모듈과 연결된 리소스 종류를 열거합니다. (유니코드)
FindResourceExW 지정된 모듈에서 지정된 형식, 이름 및 언어를 사용하여 리소스의 위치를 결정합니다. (FindResourceExW)
FindResourceW 지정된 모듈에서 지정된 형식과 이름을 가진 리소스의 위치를 결정합니다. (FindResourceW)
FindStringOrdinal 비언어적 비교를 위해 다른 유니코드 문자열에서 유니코드 문자열(와이드 문자)을 찾습니다.
FreeLibrary 로드된 DLL(동적 연결 라이브러리) 모듈을 해제하고 필요한 경우 참조 수를 줄입니다.
FreeLibraryAndExitThread 로드된 DLL(동적 연결 라이브러리)의 참조 수를 1씩 줄인 다음 ExitThread를 호출하여 호출 스레드를 종료합니다.
FreeResource 로드된 리소스의 참조 수를 1씩 줄입니다. 참조 수가 0에 도달하면 리소스가 차지하는 메모리가 해제됩니다.
GetModuleFileNameA 지정된 모듈을 포함하는 파일의 정규화된 경로를 검색합니다. 모듈은 현재 프로세스에 의해 로드되어 있어야 합니다. (ANSI)

GetModuleFileNameW 지정된 모듈을 포함하는 파일의 정규화된 경로를 검색합니다. 모듈은 현재 프로세스에 의해 로드되어 있어야 합니다. (유니코드)
GetModuleHandleA 지정된 모듈에 대한 모듈 핸들을 검색합니다. 호출 프로세스에서 모듈을 로드해야 합니다. (ANSI)
GetModuleHandleExA 지정된 모듈에 대한 모듈 핸들을 검색하고 GET_MODULE_HANDLE_EX_FLAG_UNCHANGED_REFCOUNT 지정하지 않는 한 모듈의 참조 횟수를 증분합니다. 호출 프로세스에서 모듈을 로드해야 합니다. (ANSI)
GetModuleHandleExW 지정된 모듈에 대한 모듈 핸들을 검색하고 GET_MODULE_HANDLE_EX_FLAG_UNCHANGED_REFCOUNT 지정하지 않는 한 모듈의 참조 횟수를 증분합니다. 호출 프로세스에서 모듈을 로드해야 합니다. (유니코드)
GetModuleHandleW 지정된 모듈에 대한 모듈 핸들을 검색합니다. 호출 프로세스에서 모듈을 로드해야 합니다. (유니코드)
GetProcAddress 지정된 DLL(동적 연결 라이브러리)에서 내보낸 함수 또는 변수의 주소를 검색합니다.
LoadLibraryA 지정된 모듈을 호출 프로세스의 주소 공간에 로드합니다. (LoadLibraryA)
LoadLibraryExA 지정된 모듈을 호출 프로세스의 주소 공간에 로드합니다. (LoadLibraryExA)
LoadLibraryExW 지정된 모듈을 호출 프로세스의 주소 공간에 로드합니다. (LoadLibraryExW)
LoadLibraryW 지정된 모듈을 호출 프로세스의 주소 공간에 로드합니다. (LoadLibraryW)

LoadResource 메모리에서 지정된 리소스의 첫 번째 바이트에 대한 포인터를 가져오는 데 사용할 수 있는 핸들을 검색합니다.
LockResource 메모리에서 지정된 리소스에 대한 포인터를 검색합니다.
RemoveDllDirectory AddDllDirectory를 사용하여 프로세스 DLL 검색 경로에 추가된 디렉터리를 제거합니다.
SetDefaultDllDirectories 호출 프로세스가 DLL을 로드할 때 검색할 기본 디렉터리 집합을 지정합니다. 이 검색 경로는 LOAD_LIBRARY_SEARCH 플래그 없이 LoadLibraryEx를 호출할 때 사용됩니다.
SizeofResource 지정된 리소스의 크기(바이트)를 검색합니다.

콜백 함수

ENUMRESLANGPROCA EnumResourceLanguages 및 EnumResourceLanguagesEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (ANSI)
ENUMRESLANGPROCW EnumResourceLanguages 및 EnumResourceLanguagesEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (ANSI)
ENUMRESNAMEPROCA EnumResourceNames 및 EnumResourceNamesEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (ANSI)
ENUMRESNAMEPROCW EnumResourceNames 및 EnumResourceNamesEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (유니코드)

ENUMRESTYPEPROCA EnumResourceTypes 및 EnumResourceTypesEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (ANSI)
ENUMRESTYPEPROCW EnumResourceTypes 및 EnumResourceTypesEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (유니코드)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

libloaderapi2.h 헤더

아티클2023. 08. 24.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

libloaderapi2.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

QueryOptionalDelayLoadedAPI 지연 로드된 DLL의 지정된 함수를 시스템에서 사용할 수 있는지 여부를 결정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

lmaccess.h 헤더

아티클2023. 03. 09.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보
- 네트워크 관리
- 보안 및 ID

lmaccess.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

I_NetLogonControl2 Netlogon 서비스의 다양한 측면을 제어합니다.
NetAccessAdd 지원 안 됨 (NetAccessAdd)
NetAccessDel 지원 안 됨 (NetAccessDel)
NetAccessEnum 지원 안 됨 (NetAccessEnum)
NetAccessGetInfo 지원 안 됨 (NetAccessGetInfo)
NetAccessGetUserPerms 지원 안 됨 (NetAccessGetUserPerms)
NetAccessSetInfo 지원 안 됨 (NetAccessSetInfo)
NetAddServiceAccount 독립 실행형 관리 서비스 계정(sMSA)을 만들거나 gMSA(그룹 관리 서비스 계정)에 대한 자격 증명을 검색하고 계정 정보를 로컬 컴퓨터에 저장합니다.

NetEnumerateServiceAccounts 지정된 서버에서 독립 실행형 sMSA(관리 서비스 계정)를 열거합니다.
NetGetAnyDCName NetGetAnyDCName 함수는 지정된 서버에서 직접 신뢰하는 도메인의 DC(도메인 컨트롤러)의 이름을 반환합니다.
NetGetDCName NetGetDCName 함수는 PDC(기본 도메인 컨트롤러)의 이름을 반환합니다. 지정된 도메인에 대한 백업 도메인 컨트롤러(BDC)의 이름을 반환하지 않습니다. 또한 이 함수를 PDC가 아닌 서버로 원격으로 연결할 수 없습니다.
NetGetDisplayInformationIndex NetGetDisplayInformationIndex 함수는 이름이 지정된 문자열로 시작하거나 이름이 사전순으로 문자열을 따르는 첫 번째 표시 정보 항목의 인덱스를 반환합니다.
NetGroupAdd NetGroupAdd 함수는 보안 데이터베이스(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory인 전역 그룹을 만듭니다.
NetGroupAddUser NetGroupAddUser 함수는 보안 데이터베이스의 기존 전역 그룹(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory의 기존 사용자 계정 멤버 자격을 제공합니다.
NetGroupDel NetGroupDel 함수는 SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory인 보안 데이터베이스에서 전역 그룹을 삭제합니다.
NetGroupDelUser NetGroupDelUser 함수는 SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory인 보안 데이터베이스의 특정 글로벌 그룹에서 사용자를 제거합니다.
NetGroupEnum NetGroupEnum 함수는 보안 데이터베이스의 각 전역 그룹(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory에 대한 정보를 검색합니다.
NetGroupGetInfo NetGroupGetInfo 함수는 보안 데이터베이스의 특정 전역 그룹(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory에 대한 정보를 검색합니다.

NetGroupGetUsers NetGroupGetUsers 함수는 보안 데이터베이스의 특정 전역 그룹(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory의 멤버 목록을 검색합니다.
NetGroupSetInfo NetGroupSetInfo 함수는 보안 데이터베이스(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory인 전역 그룹의 매개 변수를 설정합니다.
NetGroupSetUsers NetGroupSetUsers 함수는 지정된 전역 그룹에 대한 멤버 자격을 설정합니다.
NetIsServiceAccount 지정된 SMSA(독립 실행형 관리 서비스 계정) 또는 gMSA(그룹 관리 서비스 계정)가 지정된 서버의 Netlogon 저장소에 있는지 테스트합니다.
NetLocalGroupAdd NetLocalGroupAdd 함수는 보안 데이터베이스(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory인 로컬 그룹을 만듭니다.
NetLocalGroupAddMember NetLocalGroupAddMember 함수는 사용되지 않습니다. 대신 NetLocalGroupAddMembers 함수를 사용해야 합니다.
NetLocalGroupAddMembers NetLocalGroupAddMembers 함수는 하나 이상의 기존 사용자 계정 또는 글로벌 그룹 계정의 멤버 자격을 기존 로컬 그룹에 추가합니다.
NetLocalGroupDel NetLocalGroupDel 함수는 SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory인 보안 데이터베이스에서 로컬 그룹 계정 및 모든 멤버를 삭제합니다.
NetLocalGroupDelMember NetLocalGroupDelMember 함수는 사용되지 않습니다. 대신 NetLocalGroupDelMembers 함수를 사용해야 합니다.
NetLocalGroupDelMembers NetLocalGroupDelMembers 함수는 기존 로컬 그룹에서 하나 이상의 멤버를 제거합니다. 로컬 그룹 구성원은 사용자 또는 전역 그룹일 수 있습니다.

NetLocalGroupEnum NetLocalGroupEnum 함수는 지정된 서버의 각 로컬 그룹 계정에 대한 정보를 반환합니다.
NetLocalGroupGetInfo NetLocalGroupGetInfo 함수는 서버의 특정 로컬 그룹 계정에 대한 정보를 검색합니다.
NetLocalGroupGetMembers NetLocalGroupGetMembers 함수는 보안 데이터베이스에서 특정 로컬 그룹의 멤버 목록(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory를 검색합니다.
NetLocalGroupSetInfo NetLocalGroupSetInfo 함수는 기존 로컬 그룹의 이름을 변경합니다. 또한 함수는 주석을 로컬 그룹과 연결합니다.
NetLocalGroupSetMembers NetLocalGroupSetMembers 함수는 지정된 로컬 그룹의 멤버 자격을 설정합니다.
NetQueryDisplayInformation NetQueryDisplayInformation 함수는 사용자 계정, 컴퓨터 또는 그룹 계정 정보를 반환합니다. 이 함수를 호출하여 사용자 인터페이스에 표시할 계정 정보를 신속하게 열거합니다.
NetQueryServiceAccount 지정된 관리 서비스 계정에 대한 정보를 가져옵니다.
NetRemoveServiceAccount 계정이 SMSA(독립 실행형 관리 서비스 계정)인 경우 Active Directory 데이터베이스에서 지정된 서비스 계정을 삭제합니다.
NetUserAdd NetUserAdd 함수는 사용자 계정을 추가하고 암호 및 권한 수준을 할당합니다.
NetUserChangePassword NetUserChangePassword 함수는 지정된 네트워크 서버 또는 도메인에 대한 사용자의 암호를 변경합니다.
NetUserDel NetUserDel 함수는 서버에서 사용자 계정을 삭제합니다.

NetUserEnum NetUserEnum 함수는 서버의 모든 사용자 계정에 대한 정보를 검색합니다.
NetUserGetGroups NetUserGetGroups 함수는 지정된 사용자가 속한 전역 그룹 목록을 검색합니다.
NetUserGetInfo NetUserGetInfo 함수는 서버의 특정 사용자 계정에 대한 정보를 검색합니다.
NetUserGetLocalGroups NetUserGetLocalGroups 함수는 지정된 사용자가 속한 로컬 그룹 목록을 검색합니다.
NetUserModalsGet NetUserModalsGet 함수는 보안 데이터베이스의 모든 사용자 및 전역 그룹(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory에 대한 전역 정보를 검색합니다.
NetUserModalsSet NetUserModalsSet 함수는 보안 데이터베이스의 모든 사용자 및 전역 그룹(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory에 대한 전역 정보를 설정합니다.
NetUserSetGroups NetUserSetGroups 함수는 지정된 사용자 계정에 대한 전역 그룹 멤버 자격을 설정합니다.
NetUserSetInfo NetUserSetInfo 함수는 사용자 계정의 매개 변수를 설정합니다.
NetValidatePasswordPolicy NetValidatePasswordPolicy 함수를 사용하면 애플리케이션이 애플리케이션에서 제공하는 계정 데이터베이스에 대한 암호 준수를 확인하고 암호가 암호 정책의 복잡성, 노후화, 최소 길이 및 기록 재사용 요구 사항을 충족하는지 확인할 수 있습니다.
NetValidatePasswordPolicyFree NetValidatePasswordPolicyFree 함수는 NET_VALIDATE_OUTPUT_ARG 구조체인 OutputArg 매개 변수에 대해 NetValidatePasswordPolicy 함수가 할당하는 메모리를 해제합니다.

구조체

GROUP_INFO_0 GROUP_INFO_0 구조에는 보안 데이터베이스의 전역 그룹 이름(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory의 이름이 포함됩니다.
GROUP_INFO_1 GROUP_INFO_1 구조에는 전역 그룹 이름과 그룹과 연결할 주석이 포함됩니다.
GROUP_INFO_1002 GROUP_INFO_1002 구조에는 전역 그룹과 연결할 주석이 포함되어 있습니다.
GROUP_INFO_1005 GROUP_INFO_1005 구조에는 전역 그룹과 연결된 리소스 특성이 포함됩니다.
GROUP_INFO_2 GROUP_INFO_2 구조에는 이름, 식별자 및 리소스 특성을 포함하여 전역 그룹에 대한 정보가 포함됩니다.
GROUP_INFO_3 GROUP_INFO_3 구조에는 이름, SID(보안 식별자) 및 리소스 특성을 포함하여 전역 그룹에 대한 정보가 포함됩니다.
GROUP_USERS_INFO_0 GROUP_USERS_INFO_0 구조에는 전역 그룹 멤버 정보가 포함됩니다.
GROUP_USERS_INFO_1 GROUP_USERS_INFO_1 구조에는 전역 그룹 멤버 정보가 포함됩니다.
LOCALGROUP_INFO_0 LOCALGROUP_INFO_0 구조체에는 로컬 그룹 이름이 포함됩니다.
LOCALGROUP_INFO_1 LOCALGROUP_INFO_1 구조에는 로컬 그룹 이름과 로컬 그룹을 설명하는 주석이 포함되어 있습니다.
LOCALGROUP_INFO_1002 LOCALGROUP_INFO_1002 구조에는 로컬 그룹을 설명하는 주석이 포함되어 있습니다.

LOCALGROUP_MEMBERS_INFO_0 LOCALGROUP_MEMBERS_INFO_0 구조에는 로컬 그룹 멤버와 연결된 SID(보안 식별자)가 포함됩니다. 멤버는 사용자 계정 또는 전역 그룹 계정일 수 있습니다.
LOCALGROUP_MEMBERS_INFO_1 LOCALGROUP_MEMBERS_INFO_1 구조에는 로컬 그룹의 구성원과 연결된 SID(보안 식별자) 및 계정 정보가 포함됩니다.
LOCALGROUP_MEMBERS_INFO_2 LOCALGROUP_MEMBERS_INFO_2 구조에는 로컬 그룹 구성원과 연결된 SID(보안 식별자) 및 계정 정보가 포함됩니다.
LOCALGROUP_MEMBERS_INFO_3 LOCALGROUP_MEMBERS_INFO_3 구조에는 로컬 그룹 구성원과 연결된 계정 이름 및 도메인 이름이 포함됩니다.
LOCALGROUP_USERS_INFO_0 LOCALGROUP_USERS_INFO_0 구조에는 로컬 그룹 멤버 정보가 포함됩니다.
MSA_INFO_0 관리되는 서비스 계정에 대한 정보를 지정합니다.
NET_DISPLAY_GROUP NET_DISPLAY_GROUP 구조에는 계정 관리자가 그룹 계정에 대한 정보를 확인하기 위해 액세스할 수 있는 정보가 포함되어 있습니다.
NET_DISPLAY_MACHINE NET_DISPLAY_MACHINE 구조에는 계정 관리자가 컴퓨터 및 해당 특성에 대한 정보를 확인하기 위해 액세스할 수 있는 정보가 포함되어 있습니다.
NET_DISPLAY_USER NET_DISPLAY_USER 구조에는 계정 관리자가 사용자 계정에 대한 정보를 확인하기 위해 액세스할 수 있는 정보가 포함되어 있습니다.
NET_VALIDATE_AUTHENTICATION_INPUT_ARG 클라이언트 애플리케이션은 애플리케이션이 인증 유효성 검사를 요청할 때 NET_VALIDATE_AUTHENTICATION_INPUT_ARG 구조를 NetValidatePasswordPolicy 함수에 전달합니다.

NET_VALIDATE_OUTPUT_ARG NET_VALIDATE_OUTPUT_ARG 구조에는 사용자의 마지막 로그온 이후 변경된 영구 암호 관련 데이터와 함수의 암호 유효성 검사 결과에 대한 정보가 포함됩니다.
NET_VALIDATE_PASSWORD_CHANGE_INPUT_ARG 클라이언트 애플리케이션은 애플리케이션이 암호 변경 유효성 검사를 요청할 때 NET_VALIDATE_PASSWORD_CHANGE_INPUT_ARG 구조를 NetValidatePasswordPolicy 함수에 전달합니다.
NET_VALIDATE_PASSWORD_HASH NET_VALIDATE_PASSWORD_HASH 구조에는 암호 해시가 포함되어 있습니다.
NET_VALIDATE_PASSWORD_RESET_INPUT_ARG 클라이언트 애플리케이션은 애플리케이션이 암호 재설정 유효성 검사를 요청할 때 NET_VALIDATE_PASSWORD_RESET_INPUT_ARG 구조를 NetValidatePasswordPolicy 함수에 전달합니다.
NET_VALIDATE_PERSISTED_FIELDS NET_VALIDATE_PERSISTED_FIELDS 구조에는 사용자의 암호 속성에 대한 정보가 포함됩니다.
NETLOGON_INFO_1 도메인 컨트롤러에서 수준 1 제어 쿼리 응답을 정의합니다.
NETLOGON_INFO_2 도메인 컨트롤러에서 수준 2 컨트롤 쿼리 응답을 정의합니다.
NETLOGON_INFO_3 도메인 컨트롤러에서 수준 3 제어 쿼리 응답을 정의합니다.
NETLOGON_INFO_4 도메인 컨트롤러에서 수준 4 컨트롤 쿼리 응답을 정의합니다.
USER_INFO_0 USER_INFO_0 구조에는 사용자 계정 이름이 포함됩니다.
USER_INFO_1 USER_INFO_1 구조에는 계정 이름, 암호 데이터, 권한 수준 및 사용자의 홈 디렉터리에 대한 경로를 포함하여 사용자 계정에 대한 정보가 포함됩니다.

USER_INFO_10 USER_INFO_10 구조에는 계정 이름, 계정과 연결된 주석 및 사용자의 전체 이름을 포함하여 사용자 계정에 대한 정보가 포함됩니다.
USER_INFO_1003 USER_INFO_1003 구조에는 사용자 암호가 포함되어 있습니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1005 USER_INFO_1005 구조에는 사용자 네트워크 계정에 할당할 권한 수준이 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1006 USER_INFO_1006 구조에는 사용자의 홈 디렉터리 경로가 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1007 USER_INFO_1007 구조에는 사용자 네트워크 계정과 연결된 주석이 포함되어 있습니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1008 USER_INFO_1008 구조에는 여러 사용자 네트워크 계정 매개 변수를 정의하는 비트 플래그 집합이 포함되어 있습니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1009 USER_INFO_1009 구조에는 사용자의 로그온 스크립트 파일에 대한 경로가 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1010 USER_INFO_1010 구조에는 사용자 네트워크 계정에 할당된 연산자 권한을 정의하는 비트 플래그 집합이 포함되어 있습니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1011 USER_INFO_1011 구조에는 네트워크 사용자의 전체 이름이 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1012 USER_INFO_1012 구조에는 사용자 주석이 포함되어 있습니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.

USER_INFO_1013 USER_INFO_1013 구조에는 네트워크 계정에 대한 예약된 정보가 포함되어 있습니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1014 USER_INFO_1014 구조에는 사용자가 로그온할 수 있는 워크스테이션의 이름이 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1017 USER_INFO_1017 구조에는 네트워크 사용자 계정에 대한 만료 정보가 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1018 USER_INFO_1018 구조에는 네트워크 사용자 계정에 사용할 수 있는 최대 디스크 공간이 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1020 USER_INFO_1020 구조에는 사용자가 네트워크에 로그온할 수 있는 시간이 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1023 USER_INFO_1023 구조에는 네트워크 로그온 요청을 보내야 하는 서버의 이름이 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1024 USER_INFO_1024 구조에는 네트워크 사용자가 선택한 언어에 대한 국가/지역 코드가 포함되어 있습니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1025 USER_INFO_1025 구조에는 네트워크 사용자가 선택한 언어에 대한 코드 페이지가 포함되어 있습니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1051 USER_INFO_1051 구조에는 사용자 계정과 연결된 RID(상대 ID)가 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_1052 USER_INFO_1052 구조에는 네트워크 사용자 프로필에 대한 경로가 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.

USER_INFO_1053 USER_INFO_1053 구조에는 네트워크 계정에 대한 사용자 정보가 포함됩니다. 이 정보 수준은 NetUserSetInfo 함수를 호출하는 경우에만 유효합니다.
USER_INFO_11 USER_INFO_11 구조에는 계정 이름, 권한 수준, 사용자의 홈 디렉터리 경로 및 기타 사용자 관련 네트워크 통계를 포함하여 사용자 계정에 대한 정보가 포함됩니다.
USER_INFO_2 USER_INFO_2 구조에는 계정 이름, 암호 데이터, 권한 수준, 사용자의 홈 디렉터리에 대한 경로 및 기타 사용자 관련 네트워크 통계를 포함하여 사용자 계정에 대한 정보가 포함됩니다.
USER_INFO_20 계정 이름, 사용자의 전체 이름, 계정과 연결된 주석 및 RID(사용자의 상대 ID)를 포함하여 사용자 계정에 대한 정보를 포함합니다.
USER_INFO_21 USER_INFO_21 구조에는 단방향 암호화된 LAN Manager 2.x 호환 암호가 포함되어 있습니다.
USER_INFO_22 USER_INFO_22 구조에는 계정 이름, 권한 수준, 사용자의 홈 디렉터리에 대한 경로, 단방향 암호화된 LAN Manager 2.x 호환 암호 및 기타 사용자 관련 네트워크 통계를 포함하여 사용자 계정에 대한 정보가 포함됩니다.
USER_INFO_23 계정 이름, 사용자의 전체 이름, 계정과 연결된 주석 및 사용자의 SID(보안 식별자)를 포함하여 사용자 계정에 대한 정보를 포함합니다.
USER_INFO_24 인터넷 ID에 연결된 계정에 대한 사용자 계정 정보를 포함합니다. 이 정보에는 사용자의 인터넷 공급자 이름, 사용자의 인터넷 이름 및 사용자의 SID(보안 식별자)가 포함됩니다.
USER_INFO_3 USER_INFO_3 구조에는 계정 이름, 암호 데이터, 권한 수준, 사용자의 홈 디렉터리 경로, RID(상대 식별자) 및 기타 사용자 관련 네트워크 통계를 포함하여 사용자 계정에 대한 정보가 포함됩니다.
USER_INFO_4 USER_INFO_4 구조에는 계정 이름, 암호 데이터, 권한 수준, 사용자의 홈 디렉터리 경로, SID(보안 식별자) 및 기타 사용자 관련 네트워크 통계를 포함하여 사용자 계정에 대한 정보가 포함됩니다.

USER_MODALS_INFO_0 USER_MODALS_INFO_0 구조에는 보안 데이터베이스의 사용자 및 전역 그룹에 대한 전역 암호 정보(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory가 포함됩니다.
USER_MODALS_INFO_1 USER_MODALS_INFO_1 구조에는 로그온 서버 및 도메인 컨트롤러 정보가 포함됩니다.
USER_MODALS_INFO_1001 USER_MODALS_INFO_1001 구조에는 보안 데이터베이스의 암호에 대한 최소 길이(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory가 포함됩니다.
USER_MODALS_INFO_1002 USER_MODALS_INFO_1002 구조에는 보안 데이터베이스의 암호에 대한 최대 기간(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory가 포함됩니다.
USER_MODALS_INFO_1003 USER_MODALS_INFO_1003 구조에는 보안 데이터베이스의 암호에 대한 최소 기간(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory가 포함됩니다.
USER_MODALS_INFO_1004 USER_MODALS_INFO_1004 구조에는 보안 데이터베이스의 사용자 및 전역 그룹(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory에 대한 강제 로그오프 정보가 포함됩니다.
USER_MODALS_INFO_1005 USER_MODALS_INFO_1005 구조에는 보안 데이터베이스의 사용자 및 전역 그룹에 대한 암호 기록 정보(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory가 포함됩니다.
USER_MODALS_INFO_1006 USER_MODALS_INFO_1006 구조에는 로그온 서버 정보가 포함됩니다.
USER_MODALS_INFO_1007 USER_MODALS_INFO_1007 구조에는 도메인 컨트롤러 정보가 포함됩니다.
USER_MODALS_INFO_2 USER_MODALS_INFO_2 구조에는 SAM(보안 계정 관리자) 도메인 이름 및 식별자가 포함됩니다.

USER_MODALS_INFO_3 USER_MODALS_INFO_3 구조에는 보안 데이터베이스의 사용자 및 전역 그룹(SAM(보안 계정 관리자) 데이터베이스 또는 도메인 컨트롤러의 경우 Active Directory에 대한 잠금 정보가 포함됩니다.

열거형

MSA_INFO_LEVEL 관리 서비스 계정의 수준을 나타냅니다.
MSA_INFO_STATE 관리 서비스 계정의 상태를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

lmalert.h 헤더

아티클2023. 08. 24.

이 헤더는 네트워크 관리에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 관리

lmalert.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

ALERT_OTHER_INFO ALERT_OTHER_INFO 매크로는 경고 메시지의 경고 관련 데이터에 대한 포인터를 반환합니다. 데이터는 STD_ALERT 구조를 따르며 ADMIN_OTHER_INFO, PRINT_OTHER_INFO 또는 USER_OTHER_INFO 구조일 수 있습니다.
ALERT_VAR_DATA ALERT_VAR_DATA 매크로는 경고 메시지의 가변 길이 부분에 대한 포인터를 반환합니다. 가변 길이 데이터는 ADMIN_OTHER_INFO, PRINT_OTHER_INFO 또는 USER_OTHER_INFO 구조를 따를 수 있습니다.
NetAlertRaise NetAlertRaise 함수는 특정 이벤트가 발생할 때 등록된 모든 클라이언트에 알깁니다.
NetAlertRaiseEx NetAlertRaiseEx 함수는 특정 이벤트가 발생할 때 등록된 모든 클라이언트에 알깁니다. NetAlertRaiseEx에서 STD_ALERT 구조를 지정할 필요가 없으므로 이 확장 함수를 호출하여 경고 메시지 전송을 간소화할 수 있습니다.

구조체

ADMIN_OTHER_INFO ADMIN_OTHER_INFO 구조에는 오류 메시지 정보가 포함됩니다. NetAlertRaise 및 NetAlertRaiseEx 함수는 ADMIN_OTHER_INFO 구조를 사용하여 관리자의 인터럽트 메시지를 발생시키는 경우 정보를 지정합니다.

ERRLOG_OTHER_INFO ERRLOG_OTHER_INFO 구조에는 오류 로그 정보가 포함됩니다. NetAlertRaise 및 NetAlertRaiseEx 함수는 ERRLOG_OTHER_INFO 구조를 사용하여 오류 로그에 새 항목을 추가할 때 정보를 지정합니다.
PRINT_OTHER_INFO 인쇄 작업에 대한 정보를 포함합니다.
STD_ALERT STD_ALERT 구조에는 중요한 이벤트가 발생한 시간과 날짜가 포함됩니다.
USER_OTHER_INFO USER_OTHER_INFO 구조에는 사용자 오류 코드 정보가 포함됩니다. NetAlertRaise 및 NetAlertRaiseEx 함수는 USER_OTHER_INFO 구조를 사용하여 사용자에게 이벤트 또는 관심 조건에 대한 정보를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

lmapibuf.h 헤더

아티클2023. 08. 24.

이 헤더는 네트워크 관리에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 관리

lmapibuf.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

NetApiBufferAllocate NetApiBufferAllocate 함수는 힙에서 메모리를 할당합니다. NetApiBufferFree 함수와의 호환성이 필요한 경우에만 이 함수를 사용합니다. 그렇지 않으면 메모리 관리 함수를 사용합니다.
NetApiBufferFree NetApiBufferFree 함수는 NetApiBufferAllocate 함수가 할당하는 메모리를 해제합니다. 또한 애플리케이션은 NetApiBufferFree를 호출하여 다른 네트워크 관리 함수가 내부적으로 정보를 반환하는 데 사용하는 메모리를 해제해야 합니다.
NetApiBufferReallocate NetApiBufferReallocate 함수는 NetApiBufferAllocate 함수에 대한 이전 호출에 의해 할당된 버퍼의 크기를 변경합니다.
NetApiBufferSize NetApiBufferSize 함수는 NetApiBufferAllocate 함수 호출에 의해 할당된 버퍼의 크기(바이트)를 반환합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

lmat.h 헤더

아티클2023. 08. 24.

이 헤더는 네트워크 관리에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 관리

lmat.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

NetScheduleJobAdd NetScheduleJobAdd 함수는 지정된 미래 시간과 날짜에 실행할 작업을 제출합니다. 이 함수를 사용하려면 작업이 제출된 컴퓨터에서 일정 서비스를 시작해야 합니다.
NetScheduleJobDel NetScheduleJobDel 함수는 컴퓨터에서 실행하기 위해 대기하는 작업 범위를 삭제합니다. 이 함수를 사용하려면 작업 삭제 요청이 전송되는 컴퓨터에서 일정 서비스를 시작해야 합니다.
NetScheduleJobEnum NetScheduleJobEnum 함수는 지정된 컴퓨터에서 큐에 대기 중인 작업을 나열합니다. 이 함수를 사용하려면 일정 서비스를 시작해야 합니다.
NetScheduleJobGetInfo NetScheduleJobGetInfo 함수는 지정된 컴퓨터에서 대기 중인 특정 작업에 대한 정보를 검색합니다. 이 함수를 사용하려면 일정 서비스를 시작해야 합니다.

구조체

AT_ENUM AT_ENUM 구조에는 제출된 작업에 대한 정보가 포함됩니다. NetScheduleJobEnum 함수는 이 구조를 사용하여 제출된 작업의 전체 큐에 대한 정보를 열거하고 반환합니다.
AT_INFO AT_INFO 구조에는 작업에 대한 정보가 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

lmaudit.h 헤더

아티클2023. 03. 16.

이 헤더는 네트워크 관리에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 관리

lmaudit.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

NetAuditClear NetAuditClear 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함됩니다. 다른 애플리케이션은 이벤트 로깅을 사용해야 합니다.
NetAuditRead NetAuditRead 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함됩니다. 다른 애플리케이션은 이벤트 로깅을 사용해야 합니다.
NetAuditWrite NetAuditWrite 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함됩니다. 다른 애플리케이션은 이벤트 로깅을 사용해야 합니다.

lmconfig.h 헤더

아티클2023. 08. 24.

이 헤더는 네트워크 관리에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 관리

lmconfig.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

NetConfigGet NetConfigGet 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함됩니다. 다른 애플리케이션은 레지스트리를 사용해야 합니다.
NetConfigGetAll NetConfigGetAll 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함됩니다. 다른 애플리케이션은 레지스트리를 사용해야 합니다.
NetConfigSet NetConfigSet 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함됩니다. 다른 애플리케이션은 레지스트리를 사용해야 합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

lmdfs.h 헤더

아티클2023. 03. 11.

이 헤더는 분산 파일 시스템에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 분산 파일 시스템

lmdfs.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

NetDfsAdd 새 DFS(분산 파일 시스템) 링크를 만들거나 DFS 네임스페이스의 기존 링크에 대상을 추가합니다.
NetDfsAddFtRoot 새 도메인 기반 DFS(분산 파일 시스템) 네임스페이스를 만듭니다. 네임스페이스가 이미 있는 경우 함수는 지정된 루트 대상을 추가합니다.
NetDfsAddRootTarget 도메인 기반 또는 독립 실행형 DFS 네임스페이스를 만들거나 기존 도메인 기반 네임스페이스에 새 루트 대상을 추가합니다.
NetDfsAddStdRoot 새 독립 실행형 DFS(분산 파일 시스템) 네임스페이스를 만듭니다.
NetDfsEnum 서버에서 호스트하는 DFS(Distributed File System) 네임스페이스 또는 서버에서 호스트하는 네임스페이스의 DFS 링크를 열거합니다.
NetDfsGetClientInfo DFS 클라이언트에서 유지 관리하는 캐시에서 DFS(분산 파일 시스템) 루트 또는 링크에 대한 정보를 검색합니다.
NetDfsGetFtContainerSecurity 지정된 Active Directory 도메인의 도메인 기반 DFS 네임스페이스에 대한 컨테이너 개체의 보안 설명자를 검색합니다.

NetDfsGetInfo DFS 네임스페이스에서 지정된 DFS(분산 파일 시스템) 루트 또는 링크에 대한 정보를 검색합니다.
NetDfsGetSecurity 지정된 DFS 네임스페이스의 루트 개체에 대한 보안 설명자를 검색합니다.
NetDfsGetStdContainerSecurity 지정된 독립 실행형 DFS 네임스페이스의 컨테이너 개체에 대한 보안 설명자를 검색합니다.
NetDfsGetSupportedNamespaceVersion 지원되는 메타데이터 버전 번호를 결정합니다.
NetDfsMove DFS 링크의 이름을 바꾸거나 이동합니다.
NetDfsRemove DFS(분산 파일 시스템) 링크 또는 DFS 네임스페이스에서 DFS 링크의 특정 링크 대상을 제거합니다. 특정 링크 대상을 제거할 때 링크의 마지막 링크 대상이 제거되면 링크 자체가 제거됩니다.
NetDfsRemoveFtRoot 도메인 기반 DFS(분산 파일 시스템) 네임스페이스에서 지정된 루트 대상을 제거합니다.
NetDfsRemoveFtRootForced 루트 대상 서버가 오프라인인 경우에도 도메인 기반 DFS(분산 파일 시스템) 네임스페이스에서 지정된 루트 대상을 제거합니다.
NetDfsRemoveRootTarget 도메인 기반 DFS 네임스페이스에서 DFS 루트 대상을 제거합니다. 루트 대상이 DFS 네임스페이스의 마지막 루트 대상인 경우 이 함수는 DFS 네임스페이스를 제거합니다. 이 함수를 사용하여 독립 실행형 DFS 네임스페이스를 제거할 수도 있습니다.
NetDfsRemoveStdRoot 독립 실행형 DFS(분산 파일 시스템) 네임스페이스를 삭제합니다.
NetDfsSetClientInfo DFS 클라이언트에서 유지 관리하는 캐시의 DFS(분산 파일 시스템) 루트 또는 링크에 대한 정보를 수정합니다.

NetDfsSetFtContainerSecurity 지정된 Active Directory 도메인의 도메인 기반 DFS 네임스페이스에 대한 컨테이너 개체의 보안 설명자를 설정합니다.
NetDfsSetInfo 특정 DFS(분산 파일 시스템) 루트, 루트 대상, 링크 또는 링크 대상에 대한 정보를 설정하거나 수정합니다.
NetDfsSetSecurity 지정된 DFS 네임스페이스의 루트 개체에 대한 보안 설명자를 설정합니다.
NetDfsSetStdContainerSecurity 지정된 독립 실행형 DFS 네임스페이스의 컨테이너 개체에 대한 보안 설명자를 설정합니다.

구조체

DFS_GET_PKT_ENTRY_STATE_ARG FSCTL_DFS_GET_PKT_ENTRY_STATE 제어 코드와 함께 사용되는 입력 버퍼입니다.
DFS_INFO_1 DFS(분산 파일 시스템) 루트 또는 링크의 이름을 포함합니다.
DFS_INFO_100 DFS(분산 파일 시스템) 루트 또는 링크와 연결된 주석을 포함합니다.
DFS_INFO_101 DFS 루트, 링크, 루트 대상 또는 링크 대상의 스토리지 상태를 설명합니다.
DFS_INFO_102 DFS(Distributed File System) 루트 또는 명명된 DFS 루트의 링크와 연결할 제한 시간 값을 포함합니다.
DFS_INFO_103 DFS 루트 또는 링크에 대한 특정 동작을 설정하는 속성을 포함합니다.

DFS_INFO_104 DFS 루트 대상 또는 링크 대상의 우선 순위를 포함합니다.
DFS_INFO_105 속성 플래그로 지정된 주석, 상태, 제한 시간 및 DFS 동작을 포함하여 DFS 루트 또는 링크에 대한 정보를 포함합니다.
DFS_INFO_106 DFS 루트 대상 또는 링크 대상에 대한 스토리지 상태 및 우선 순위를 포함합니다. 이 구조는 NetDfsSetInfo 함수에만 사용됩니다.
DFS_INFO_107 주석, 상태, 제한 시간, 속성 플래그 및 링크 재 분석 지점 보안 설명자를 포함하여 DFS 루트 또는 링크에 대한 정보를 포함합니다.
DFS_INFO_150 DFS 링크의 재문 분석 지점에 대한 보안 설명자를 포함합니다.
DFS_INFO_2 DFS(분산 파일 시스템) 루트 또는 링크에 대한 정보를 포함합니다. 이 구조체에는 루트 또는 링크의 이름, 상태 및 DFS 대상 수가 포함됩니다.
DFS_INFO_200 도메인 기반 DFS(분산 파일 시스템) 네임스페이스의 이름을 포함합니다.
DFS_INFO_3 DFS(분산 파일 시스템) 루트 또는 링크에 대한 정보를 포함합니다. 이 구조에는 이름, 상태, DFS 대상 수 및 루트 또는 링크의 각 대상에 대한 정보가 포함됩니다.
DFS_INFO_300 DFS 네임스페이스의 이름 및 형식(도메인 기반 또는 독립 실행형)을 포함합니다.
DFS_INFO_4 DFS(분산 파일 시스템) 루트 또는 링크에 대한 정보를 포함합니다. 이 구조에는 이름, 상태, GUID, 제한 시간, 대상 수 및 루트 또는 링크의 각 대상에 대한 정보가 포함됩니다.

DFS_INFO_5 DFS(분산 파일 시스템) 루트 또는 링크에 대한 정보를 포함합니다. 이 구조에는 이름, 상태, GUID, 제한 시간, 네임스페이스/루트/링크 속성, 메타데이터 크기 및 루트 또는 링크의 대상 수가 포함됩니다.
DFS_INFO_50 기존 DFS 네임스페이스의 DFS 메타데이터 버전 및 기능을 포함합니다.
DFS_INFO_6 DFS(분산 파일 시스템) 루트 또는 링크에 대한 정보를 포함합니다. 이 구조에는 이름, 상태, GUID, 제한 시간, 네임스페이스/루트/링크 속성, 메타데이터 크기, 대상 수 및 루트 또는 링크의 각 대상에 대한 정보가 포함됩니다.
DFS_INFO_7 DFS 네임스페이스에 대한 정보를 포함합니다. 이 구조에는 네임스페이스에 대한 메타데이터에 대한 버전 GUID가 포함됩니다.
DFS_INFO_8 이름, 상태, GUID, 제한 시간, 속성 플래그, 메타데이터 크기, DFS 대상 정보 및 루트 또는 링크에 대한 링크 재분석 지점 보안 설명자를 포함합니다.
DFS_INFO_9 이름, 상태, GUID, 제한 시간, 속성 플래그, 메타데이터 크기, DFS 대상 정보, 링크 재분석 지점 보안 설명자 및 루트 또는 링크에 대한 DFS 대상 목록을 포함합니다.
DFS_STORAGE_INFO DFS 네임스페이스 또는 DFS 클라이언트에서 유지 관리하는 캐시의 DFS 루트 또는 링크 대상에 대한 정보를 포함합니다.
DFS_STORAGE_INFO_1 DFS 대상 서버 이름 및 공유 이름과 대상의 상태 및 우선 순위를 포함하여 DFS 대상에 대한 정보를 포함합니다.
DFS_SUPPORTED_NAMESPACE_VERSION_INFO DFS 네임스페이스에 대한 버전 정보를 포함합니다.
DFS_TARGET_PRIORITY 특정 DFS 대상의 우선 순위 클래스 및 순위를 포함합니다.

열거형

DFS_NAMESPACE_VERSION_ORIGIN DFS 네임스페이스 버전 정보의 원본을 식별합니다.
DFS_TARGET_PRIORITY_CLASS DFS_TARGET_PRIORITY_CLASS 열거형 r1(lmdfs.h)은 가능한 DFS 대상 우선 순위 클래스 설정 집합을 정의합니다.

lmerrlog.h 헤더

아티클2023. 08. 24.

이 헤더는 네트워크 관리에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 관리

lmerrlog.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

NetErrorLogClear NetErrorLogClear 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함됩니다. 다른 애플리케이션은 이벤트 로깅을 사용해야 합니다.
NetErrorLogRead NetErrorLogRead 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함됩니다. 다른 애플리케이션은 이벤트 로깅을 사용해야 합니다.
NetErrorLogWrite NetErrorLogWrite 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함됩니다. 다른 애플리케이션은 이벤트 로깅을 사용해야 합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

lmjoin.h 헤더

아티클2023. 03. 16.

이 헤더는 네트워크 관리에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 관리

lmjoin.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

NetAddAlternateComputerName 지정된 컴퓨터의 대체 이름을 추가합니다.
NetCreateProvisioningPackage 오프라인 도메인 가입 작업에서 나중에 사용할 컴퓨터 계정을 프로비전하는 프로비저닝 패키지를 만듭니다. 패키지에는 프로비전 중에 컴퓨터에 추가할 인증서 및 정책에 대한 정보가 포함될 수도 있습니다.
NetEnumerateComputerNames 지정된 컴퓨터의 이름을 열거합니다.
NetFreeAadJoinInformation 테넌트에 대한 조인 정보와 NetGetAadJoinInformation 함수를 호출하여 검색한 지정된 DSREG_JOIN_INFO 구조에 할당된 메모리를 해제합니다.
NetGetAadJoinInformation 지정된 테넌트에 대한 조인 정보를 검색합니다. 이 함수는 현재 사용자가 추가한 Microsoft Azure Active Directory 및 회사 계정에 대한 조인 정보를 검사합니다.
NetGetJoinableOU NetGetJoinableOUs 함수는 컴퓨터 계정을 만들 수 있는 OU(조직 구성 단위) 목록을 검색합니다.
NetGetJoinInformation NetGetJoinInformation 함수는 지정된 컴퓨터에 대한 조인 상태 정보를 검색합니다.

NetJoinDomain NetJoinDomain 함수는 컴퓨터를 작업 그룹 또는 도메인에 조인합니다.
NetProvisionComputerAccount 오프라인 도메인 가입 작업에서 나중에 사용할 컴퓨터 계정을 프로비전합니다.
NetRemoveAlternateComputerName 지정된 컴퓨터의 대체 이름을 제거합니다.
NetRenameMachineInDomain NetRenameMachineInDomain 함수는 도메인에 있는 컴퓨터의 이름을 변경합니다.
NetRequestOfflineDomainJoin 컴퓨터에서 로컬로 실행하여 볼륨에 탑재된 Windows 운영 체제 이미지를 수정합니다. (NetRequestOfflineDomainJoin)
NetRequestProvisioningPackageInstall 컴퓨터에서 로컬로 실행하여 볼륨에 탑재된 Windows 운영 체제 이미지를 수정합니다. (NetRequestProvisioningPackageInstall)
NetSetPrimaryComputerName 지정된 컴퓨터의 기본 컴퓨터 이름을 설정합니다.
NetUnjoinDomain NetUnjoinDomain 함수는 작업 그룹 또는 도메인에서 컴퓨터를 연결 해제합니다.
NetValidateName NetValidateName 함수는 이름이 지정된 이름 유형(컴퓨터 이름, 작업 그룹 이름, 도메인 이름 또는 DNS 컴퓨터 이름)에 대해 유효한지 확인합니다.

구조체

DSREG_JOIN_INFO 디바이스가 Microsoft Azure Active Directory 조인되는 방법에 대한 정보를 포함합니다.

DSREG_USER_INFO 디바이스를 Microsoft Azure Active Directory 조인하는 데 사용되는 사용자 계정에 대한 정보를 포함합니다.
NETSETUP_PROVISIONING_PARAMS NETSETUP_PROVISIONING_PARAMS 구조에는 NetCreateProvisionPackage 함수를 사용하여 프로비저닝 패키지를 만들 때 사용되는 정보가 포함되어 있습니다.

열거형

DSREG_JOIN_TYPE 디바이스를 Microsoft Azure Active Directory 조인할 수 있는 가능한 방법을 지정합니다.

lmmsg.h 헤더

아티클2023. 08. 24.

이 헤더는 네트워크 관리에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 관리

lmmsg.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

NetMessageBufferSend NetMessageBufferSend 함수는 등록된 메시지 별칭에 정보 버퍼를 보냅니다.
NetMessageNameAdd NetMessageNameAdd 함수는 메시지 이름 테이블에 메시지 별칭을 등록합니다. 함수를 사용하려면 메신저 서비스를 시작해야 합니다.
NetMessageNameDel NetMessageNameDel 함수는 메시지 이름 테이블에서 메시지 별칭을 삭제합니다. 함수를 사용하려면 메신저 서비스를 시작해야 합니다.
NetMessageNameEnum NetMessageNameEnum 함수는 지정된 컴퓨터에서 메시지를 받는 메시지 별칭을 나열합니다. 함수를 사용하려면 메신저 서비스를 시작해야 합니다.
NetMessageNameGetInfo NetMessageNameGetInfo 함수는 메시지 이름 테이블의 특정 메시지 별칭에 대한 정보를 검색합니다. 함수를 사용하려면 메신저 서비스를 시작해야 합니다.

구조체

MSG_INFO_0 MSG_INFO_0 구조체는 메시지 별칭을 지정합니다.

MSG_INFO_1 MSG_INFO_1 구조체는 메시지 별칭을 지정합니다. 이 구조체는 호환성을 위해서만 존재합니다. 메시지 전달은 지원되지 않습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

lmremutl.h 헤더

아티클2023. 08. 24.

이 헤더는 네트워크 관리에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 관리

lmremutl.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

NetRemoteComputerSupports NetRemoteComputerSupports 함수는 리트리버를 쿼리하여 원격 시스템에서 지원하는 선택적 기능을 검색합니다.
NetRemoteTOD NetRemoteTOD 함수는 지정된 서버에서 하루 중 시간 정보를 반환합니다.

구조체

TIME_OF_DAY_INFO TIME_OF_DAY_INFO 구조에는 원격 서버의 하루 중 시간에 대한 정보가 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

lmserver.h 헤더

아티클2023. 03. 11.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 관리
- 시스템 서비스

lmserver.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

NetServerComputerNameAdd NetServerComputerNameAdd 함수는 지정된 서버가 활성 상태인 전송을 열거하고 에뮬레이트된 서버 이름을 각 전송에 바인딩합니다.
NetServerComputerNameDel NetServerComputerNameDel 함수를 사용하면 지정된 서버가 NetServerComputerNameAdd 함수에 대한 이전 호출로 설정된 에뮬레이트된 서버 이름 지원을 중단합니다. 이 함수는 에뮬레이트된 이름에서 네트워크 전송을 바인딩 해제하여 이 작업을 수행합니다.
NetServerDiskEnum NetServerDiskEnum 함수는 서버의 디스크 드라이브 목록을 검색합니다. 이 함수는 세 문자 문자열(드라이브 문자, 콜론 및 종료 null 문자)의 배열을 반환합니다.
NetServerEnum NetServerEnum 함수는 도메인에 표시되는 지정된 형식의 모든 서버를 나열합니다.
NetServerGetInfo NetServerGetInfo 함수는 지정된 서버에 대한 현재 구성 정보를 검색합니다.
NetServerSetInfo NetServerSetInfo 함수는 서버의 운영 매개 변수를 설정합니다. 개별적으로 또는 집합적으로 설정할 수 있습니다. 이 정보는 시스템을 다시 초기화한 후에도 계속 적용할 수 있는 방식으로 저장됩니다.
NetServerTransportAdd NetServerTransportAdd 함수는 서버를 전송 프로토콜에 바인딩합니다.

NetServerTransportAddEx NetServerTransportAddEx 함수는 지정된 서버를 전송 프로토콜에 바인딩합니다.
NetServerTransportDel NetServerTransportDel 함수는 서버에서 전송 프로토콜을 바인딩 해제(또는 연결 끊기)합니다. 실제로 서버는 지정된 전송 프로토콜(예: TCP 또는 XNS)을 사용하여 클라이언트와 더 이상 통신할 수 없습니다.
NetServerTransportEnum NetServerTransportEnum 함수는 서버에서 관리하는 전송 프로토콜에 대한 정보를 제공합니다.
SetServiceBits 서비스 제어 관리자 및 서버 서비스에 서비스 유형을 등록합니다.

구조체

SERVER_INFO_100 SERVER_INFO_100 구조에는 이름 및 플랫폼을 포함하여 지정된 서버에 대한 정보가 포함됩니다.
SERVER_INFO_1005 SERVER_INFO_1005 구조에는 지정된 서버를 설명하는 주석이 포함되어 있습니다.
SERVER_INFO_101 SERVER_INFO_101 구조에는 이름, 플랫폼, 서버 유형 및 관련 소프트웨어를 포함하여 지정된 서버에 대한 정보가 포함됩니다.
SERVER_INFO_1010 SERVER_INFO_1010 구조에는 지정된 서버와 연결된 자동 연결 끊기 시간이 포함됩니다.
SERVER_INFO_1016 SERVER_INFO_1016 구조에는 서버가 동일한 네트워크 도메인의 다른 컴퓨터에 표시되는지 여부에 대한 정보가 포함됩니다.
SERVER_INFO_1017 SERVER_INFO_1017 구조에는 지정된 서버와 연결된 네트워크 공지 속도가 포함됩니다.

SERVER_INFO_1018 SERVER_INFO_1018 구조에는 지정된 서버에 대해 공지 속도가 얼마나 달라질 수 있는지에 대한 정보가 포함되어 있습니다.
SERVER_INFO_102 이름, 플랫폼, 서버 유형, 특성 및 관련 소프트웨어를 포함하여 지정된 서버에 대한 정보를 포함합니다.
SERVER_INFO_1107 SERVER_INFO_1107 구조는 지정된 서버에 동시에 로그온할 수 있는 사용자 수를 지정합니다.
SERVER_INFO_1501 SERVER_INFO_1501 구조체는 지정된 서버의 한 세션에서 열 수 있는 파일 수를 지정합니다.
SERVER_INFO_1502 SERVER_INFO_1502 구조는 지정된 서버에 대한 클라이언트당 최대 가상 회로 수를 지정합니다.
SERVER_INFO_1503 SERVER_INFO_1503 구조는 동시에 수행할 수 있는 검색 작업의 수를 지정합니다.
SERVER_INFO_1506 SERVER_INFO_1506 구조에는 지정된 서버에서 할당할 수 있는 최대 작업 항목 수에 대한 정보가 포함됩니다.
SERVER_INFO_1509 SERVER_INFO_1509 구조체는 최대 원시 모드 버퍼 크기를 지정합니다.
SERVER_INFO_1510 SERVER_INFO_1510 구조는 단일 가상 회로를 사용하여 지정된 서버에 로그온할 수 있는 최대 사용자 수를 지정합니다.
SERVER_INFO_1511 SERVER_INFO_1511 구조는 사용자가 단일 가상 회로로 만들 수 있는 최대 트리 연결 수를 지정합니다.
SERVER_INFO_1512 SERVER_INFO_1512 구조에는 지정된 서버가 특정 시간에 할당할 수 있는 페이지가 지정되지 않은 메모리의 최대 크기가 포함됩니다.

SERVER_INFO_1513 SERVER_INFO_1513 구조에는 지정된 서버가 특정 시간에 할당할 수 있는 최대 페이지 가능 메모리 크기가 포함됩니다.
SERVER_INFO_1515 SERVER_INFO_1515 구조는 클라이언트의 로그온 시간이 만료되면 서버에서 클라이언트의 연결을 강제로 끊어야 하는지 여부를 지정합니다.
SERVER_INFO_1516 SERVER_INFO_1516 구조는 서버가 신뢰할 수 있는 시간 원본인지 여부를 지정합니다.
SERVER_INFO_1518 SERVER_INFO_1518 구조는 서버가 LAN Manager 2.x 클라이언트에 표시되는지 여부를 지정합니다.
SERVER_INFO_1523 SERVER_INFO_1523 구조는 서버가 불완전한 검색 작업에 대한 정보를 유지하는 기간을 지정합니다.
SERVER_INFO_1528 SERVER_INFO_1528 구조는 서비스 요청을 깨우기 전에 청소기가 유휴 상태로 유지되는 기간을 지정합니다.
SERVER_INFO_1529 SERVER_INFO_1529 구조는 서버가 더 많은 항목을 할당하기 전에 필요한 최소 수신 작업 항목 수를 지정합니다.
SERVER_INFO_1530 SERVER_INFO_1530 구조는 서버에서 서버 메시지 블록 처리를 시작하는 데 필요한 사용 가능한 수신 작업 항목의 최소 수를 지정합니다.
SERVER_INFO_1533 SERVER_INFO_1533 구조는 클라이언트가 서버에 보낼 수 있는 최대 미해결 요청 수를 지정합니다.
SERVER_INFO_1536 SERVER_INFO_1536 구조는 서버에서 클라이언트가 파일에서 기회 잠금(oplock)을 사용할 수 있도록 허용하는지 여부를 지정합니다.

SERVER_INFO_1538 SERVER_INFO_1538 구조는 여러 MS-DOS FCB(파일 제어 블록)가 단일 위치에 배치되는지 여부를 지정합니다.
SERVER_INFO_1539 SERVER_INFO_1539 구조는 서버가 원시 SMB(서버 메시지 블록)를 처리하는지 여부를 지정합니다.
SERVER_INFO_1540 SERVER_INFO_1540 구조는 서버에서 리디렉션된 서버 드라이브를 공유할 수 있는지 여부를 지정합니다.
SERVER_INFO_1541 SERVER_INFO_1541 구조는 서버에 연결하기 위한 클라이언트의 요청 버스트를 처리하기 위해 서버가 따로 설정한 최소 사용 가능한 연결 블록 수를 지정합니다.
SERVER_INFO_1542 SERVER_INFO_1542 구조는 클라이언트가 서버에 연결하기 위한 요청의 버스트를 처리하기 위해 서버가 따로 설정한 최대 사용 가능한 연결 블록 수를 지정합니다.
SERVER_INFO_1544 SERVER_INFO_1544 구조는 연결 테이블에 할당할 초기 트리 연결 수를 지정합니다.
SERVER_INFO_1550 SERVER_INFO_1550 구조는 경고 메시지를 보내기 전에 남은 사용 가능한 디스크 공간의 백분율을 지정합니다.
SERVER_INFO_1552 SERVER_INFO_1552 구조체는 링크 지연에 허용되는 최대 시간을 지정합니다.
SERVER_INFO_402 지정된 서버에 대한 정보를 포함합니다.
SERVER_INFO_403 SERVER_INFO_403 구조에는 지정된 서버에 대한 정보가 포함됩니다.
SERVER_INFO_502 SERVER_INFO_502 구조체는 사용되지 않습니다. 구조체에는 지정된 서버에 대한 정보가 포함됩니다.

SERVER_INFO_503 SERVER_INFO_503 구조체는 사용되지 않습니다. 구조체에는 지정된 서버에 대한 정보가 포함됩니다.
SERVER_TRANSPORT_INFO_0 SERVER_TRANSPORT_INFO_0 구조에는 네트워크의 이름, 주소 및 위치를 포함하여 지정된 전송 프로토콜에 대한 정보가 포함됩니다.
SERVER_TRANSPORT_INFO_1 SERVER_TRANSPORT_INFO_1 구조에는 이름 및 주소를 포함하여 지정된 전송 프로토콜에 대한 정보가 포함됩니다. 이 정보 수준은 NetServerTransportAddEx 함수에만 유효합니다.
SERVER_TRANSPORT_INFO_2 SERVER_TRANSPORT_INFO_2 구조에는 전송 이름 및 주소를 포함하여 지정된 전송 프로토콜에 대한 정보가 포함됩니다. 이 정보 수준은 NetServerTransportAddEx 함수에만 유효합니다.
SERVER_TRANSPORT_INFO_3 SERVER_TRANSPORT_INFO_3 구조에는 이름, 주소 및 암호(자격 증명)를 포함하여 지정된 전송 프로토콜에 대한 정보가 포함됩니다. 이 정보 수준은 NetServerTransportAddEx 함수에만 유효합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

lmshare.h 헤더

아티클2023. 08. 24.

이 헤더는 데이터 액세스 및 스토리지에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지

lmshare.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

NetConnectionEnum 서버의 공유 리소스에 대한 모든 연결 또는 특정 컴퓨터에서 설정된 모든 연결을 나열합니다.
NetFileClose 리소스를 강제로 닫습니다. 이 함수는 오류가 다른 수단으로 닫을 수 없는 경우에 사용할 수 있습니다. NetFileClose는 파일을 닫기 전에 클라이언트 시스템에 캐시된 데이터를 파일에 쓰지 않으므로 주의해서 사용해야 합니다.
NetFileEnum 지정된 매개 변수에 따라 서버의 열려 있는 일부 또는 모든 파일에 대한 정보를 반환합니다.
NetFileGetInfo 서버 리소스의 특정 열기에 대한 정보를 검색합니다.
NetSessionDel 서버와 워크스테이션 간의 네트워크 세션을 종료합니다.
NetSessionEnum 서버에 설정된 세션에 대한 정보를 제공합니다.
NetSessionGetInfo 특정 서버와 워크스테이션 간에 설정된 세션에 대한 정보를 검색합니다.
NetShareAdd 서버 리소스를 공유합니다.

NetShareCheck 서버가 디바이스를 공유하고 있는지 여부를 확인합니다.
NetShareDel 공유 리소스에 대한 모든 연결이 끊어지도록 서버의 공유 리소스 목록에서 공유 이름을 삭제합니다.
NetShareDelEx 서버의 공유 리소스 목록에서 공유 이름을 삭제하여 해당 공유에 대한 모든 연결을 끊습니다.
NetShareEnum 서버의 각 공유 리소스에 대한 정보를 검색합니다.
NetShareGetInfo 서버의 특정 공유 리소스에 대한 정보를 검색합니다.
NetShareSetInfo 공유 리소스의 매개 변수를 설정합니다.

구조체

CONNECTION_INFO_0 연결의 ID 번호를 포함합니다.
CONNECTION_INFO_1 연결의 ID 번호, 열려 있는 파일 수, 연결 시간, 연결의 사용자 수 및 연결 유형을 포함합니다.
FILE_INFO_2 파일, 디바이스 또는 파이프의 ID 번호를 포함합니다.
FILE_INFO_3 파일, 디바이스 및 파이프에 대한 식별 번호 및 기타 관련 정보를 포함합니다.
SESSION_INFO_0

세션을 설정한 컴퓨터의 이름을 포함합니다.
SESSION_INFO_1 컴퓨터의 이름을 포함하여 세션에 대한 정보를 포함합니다. 사용자의 이름입니다. 컴퓨터에서 파일, 파이프 및 디바이스를 엽니다.
SESSION_INFO_10 컴퓨터의 이름을 포함하여 세션에 대한 정보를 포함합니다. 사용자의 이름입니다. 세션에 대한 활성 및 유휴 시간입니다.
SESSION_INFO_2 컴퓨터의 이름을 포함하여 세션에 대한 정보를 포함합니다. 사용자의 이름입니다. 컴퓨터에서 파일, 파이프 및 디바이스 열기 및 세션을 설정한 클라이언트의 유형입니다.
SESSION_INFO_502 컴퓨터의 이름을 포함하여 세션에 대한 정보를 포함합니다. 사용자의 이름입니다. 컴퓨터에서 파일, 파이프 및 디바이스 열기 및 클라이언트가 사용하는 전송의 이름입니다.
SHARE_INFO_0 공유 리소스의 이름을 포함합니다.
SHARE_INFO_1 리소스의 이름 및 유형 및 리소스와 연결된 주석을 포함하여 공유 리소스에 대한 정보를 포함합니다.
SHARE_INFO_1004 공유 리소스와 연결된 주석을 포함합니다.
SHARE_INFO_1005 공유 리소스에 대한 정보를 포함합니다.
SHARE_INFO_1006 공유 리소스가 수용할 수 있는 최대 동시 연결 수를 지정합니다.
SHARE_INFO_1501 지정된 공유와 연결된 보안 설명자를 포함합니다. 자세한 내용은 보안 설명자를 참조하세요.
SHARE_INFO_2

리소스 이름, 유형 및 권한, 현재 연결 수를 포함하여 공유 리소스에 대한 정보를 포함합니다.
SHARE_INFO_501 리소스의 이름 및 유형 및 리소스와 연결된 주석을 포함하여 공유 리소스에 대한 정보를 포함합니다.
SHARE_INFO_502 리소스 이름, 유형 및 권한, 연결 수 및 기타 관련 정보를 포함하여 공유 리소스에 대한 정보를 포함합니다.
SHARE_INFO_503 공유 리소스에 대한 정보를 포함합니다. 서버 이름도 포함되어 있다는 점을 제외하고 SHARE_INFO_502 구조와 동일합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

lmstats.h 헤더

아티클2023. 08. 24.

이 헤더는 데이터 액세스 및 스토리지에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지

lmstats.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

NetStatisticsGet 서비스에 대한 운영 통계를 검색합니다. 현재 워크스테이션 및 서버 서비스만 지원됩니다.

구조체

STAT_SERVER_0 서버에 대한 통계 정보를 포함합니다.
STAT_WORKSTATION_0 STAT_WORKSTATION_0 구조체(lmstats.h)에는 지정된 워크스테이션에 대한 통계 정보가 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

lmsvc.h 헤더

아티클2023. 08. 24.

이 헤더는 네트워크 관리에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 관리

lmsvc.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

NetServiceControl NetServiceControl 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함됩니다. 다른 애플리케이션은 서비스 함수를 사용해야 합니다.
NetServiceEnum NetServiceEnum 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함됩니다. 다른 애플리케이션은 서비스 함수를 사용해야 합니다.
NetServiceGetInfo NetServiceGetInfo 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함됩니다. 다른 애플리케이션은 서비스 함수를 사용해야 합니다.
NetServiceInstall NetServiceInstall 함수는 사용되지 않습니다. 16비트 버전의 Windows와의 호환성을 위해 포함됩니다. 다른 애플리케이션은 서비스 함수를 사용해야 합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

lmuse.h 헤더

아티클2023. 08. 24.

이 헤더는 네트워크 관리에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 관리

lmuse.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

NetUseAdd NetUseAdd 함수는 로컬 컴퓨터와 원격 서버 간의 연결을 설정합니다.
NetUseDel NetUseDel 함수는 공유 리소스에 대한 연결을 종료합니다.
NetUseEnum NetUseEnum 함수는 로컬 컴퓨터와 원격 서버의 리소스 간의 모든 현재 연결을 나열합니다.
NetUseGetInfo NetUseGetInfo 함수는 공유 리소스에 대한 연결에 대한 정보를 검색합니다.

구조체

USE_INFO_0 USE_INFO_0 구조에는 공유 리소스의 이름과 해당 리소스로 리디렉션된 로컬 디바이스가 포함됩니다.
USE_INFO_1 로컬 디바이스와 공유 리소스 간의 연결에 대한 정보를 포함합니다.
USE_INFO_2

USE_INFO_2 구조에는 연결 유형, 연결 상태, 사용자 이름 및 도메인 이름을 포함하여 로컬 컴퓨터와 공유 리소스 간의 연결에 대한 정보가 포함됩니다.
USE_INFO_3 USE_INFO_3 구조에는 연결 유형, 연결 상태, 사용자 이름, 도메인 이름 및 연결 동작을 설명하는 특정 플래그를 포함하여 로컬 컴퓨터와 공유 리소스 간의 연결에 대한 정보가 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

lmwksta.h 헤더

아티클2023. 08. 24.

이 헤더는 네트워크 관리에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 관리

lmwksta.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

NetWkstaGetInfo NetWkstaGetInfo 함수는 워크스테이션의 구성에 대한 정보를 반환합니다.
NetWkstaSetInfo NetWkstaSetInfo 함수는 시스템이 다시 초기화된 후에도 계속 적용되는 정보로 워크스테이션을 구성합니다.
NetWkstaTransport추가 지원되지 않습니다. (NetWkstaTransportAdd)
NetWkstaTransportDel 지원되지 않습니다. (NetWkstaTransportDel)
NetWkstaTransportEnum NetWkstaTransportEnum 함수는 서버 컴퓨터에 파일 요청을 생성하는 클라이언트 컴퓨터의 소프트웨어인 리렉터에서 관리하는 전송 프로토콜에 대한 정보를 제공합니다.
NetWkstaUserEnum NetWkstaUserEnum 함수는 현재 워크스테이션에 로그온한 모든 사용자에 대한 정보를 나열합니다. 이 목록에는 대화형, 서비스 및 일괄 로그온이 포함됩니다.
NetWkstaUserGetInfo NetWkstaUserGetInfo 함수는 현재 로그온한 사용자에 대한 정보를 반환합니다. 이 함수는 로그온한 사용자의 컨텍스트에서 호출되어야 합니다.
NetWkstaUserSetInfo

NetWkstaUserSetInfo 함수는 워크스테이션의 구성 요소에 대한 사용자별 정보를 설정합니다.

구조체

WKSTA_INFO_100 플랫폼별 정보, 도메인 및 로컬 컴퓨터의 이름, 운영 체제 관련 정보를 포함하여 워크스테이션 환경에 대한 정보를 포함합니다.
WKSTA_INFO_101 플랫폼별 정보, 도메인 및 로컬 컴퓨터의 이름, 운영 체제 관련 정보를 포함하여 워크스테이션 환경에 대한 정보를 포함합니다. (WKSTA_INFO_101)
WKSTA_INFO_102 플랫폼별 정보, 도메인 및 로컬 컴퓨터의 이름, 운영 체제 관련 정보를 포함하여 워크스테이션 환경에 대한 정보를 포함합니다. (WKSTA_INFO_102)
WKSTA_INFO_502 WKSTA_INFO_502 구조체는 사용되지 않습니다. 구조체에는 워크스테이션 환경에 대한 정보가 포함됩니다.
WKSTA_TRANSPORT_INFO_0 WKSTA_TRANSPORT_INFO_0 구조에는 WAN(광역 네트워크) 또는 NetBIOS와 같은 워크스테이션 전송 프로토콜에 대한 정보가 포함됩니다.
WKSTA_USER_INFO_0 WKSTA_USER_INFO_0 구조체에는 지정된 워크스테이션의 사용자 이름이 포함됩니다.
WKSTA_USER_INFO_1 WKSTA_USER_INFO_1 구조에는 특정 워크스테이션과 관련된 사용자 정보가 포함됩니다. 정보에는 현재 사용자의 이름과 워크스테이션에서 액세스하는 도메인이 포함됩니다.
WKSTA_USER_INFO_1101 WKSTA_USER_INFO_1101 구조에는 워크스테이션에서 액세스하는 도메인에 대한 정보가 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

loadperf.h 헤더

아티클2023. 08. 24.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보
- 성능 카운터

loadperf.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

InstallPerfDllA 입력 .ini 파일에 정의된 대로 성능 카운터 문자열을 시스템 레지스트리에 설치합니다. (ANSI)
InstallPerfDllW 입력 .ini 파일에 정의된 대로 성능 카운터 문자열을 시스템 레지스트리에 설치합니다. (유니코드)
LoadPerfCounterTextStringsA 지정된 초기화 파일에 정의된 성능 개체 및 카운터를 컴퓨터에 로드합니다. (ANSI)
LoadPerfCounterTextStringsW 지정된 초기화 파일에 정의된 성능 개체 및 카운터를 컴퓨터에 로드합니다. (유니코드)
UnloadPerfCounterTextStringsA 지정된 애플리케이션에 대한 컴퓨터에서 성능 개체 및 카운터를 언로드합니다. (ANSI)
UnloadPerfCounterTextStringsW 지정된 애플리케이션에 대한 컴퓨터에서 성능 개체 및 카운터를 언로드합니다. (유니코드)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

locationapi.h 헤더

아티클2023. 08. 24.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 위치 API
- winlocation_com_ref

locationapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICivicAddressReport ICivicAddressReport는 주소 형식의 정보를 포함하는 위치 보고서를 나타냅니다.
IDefaultLocation IDefaultLocation은 기본 위치를 지정하거나 검색하는 데 사용되는 메서드를 제공합니다.
ILatLongReport ILatLongReport는 위도 및 경도 형식의 정보를 포함하는 위치 보고서를 나타냅니다.
ILocation 위치 보고서, 이벤트 등록 및 센서 권한을 관리하는 데 사용되는 메서드를 제공합니다.
ILocationEvents ILocationEvents는 이벤트 알림을 수신하려는 경우 구현해야 하는 콜백 메서드를 제공합니다.
ILocationPower Windows 8의 Windows 스토어 앱 브라우저에서 앱이 일시 중단(연결 끊기) 및 복원(연결)되었음을 위치 플랫폼에 알리는 데 사용됩니다. (ILocationPower)
ILocationReport 위치 보고서의 부모 인터페이스입니다.

열거형

LOCATION_REPORT_STATUS 특정 보고서 유형의 새 보고서에 대해 가능한 상태 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

lowlevelmonitorconfigurationapi.h 헤더

아티클2023. 03. 16.

이 헤더는 모니터 구성에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 구성 모니터링

lowlevelmonitorconfigurationapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CapabilitiesRequestAndCapabilitiesReply 모니터의 기능을 설명하는 문자열을 검색합니다.
GetCapabilitiesStringLength 모니터의 기능 문자열 길이를 검색합니다.
GetTimingReport 모니터의 가로 및 세로 동기화 빈도를 검색합니다.
GetVCPFeatureAndVCPFeatureReply 모니터에 대한 VCP(가상 제어판) 코드의 현재 값, 최대값 및 코드 형식을 검색합니다.
SaveCurrentSettings 현재 모니터 설정을 디스플레이의 비휘발성 스토리지에 저장합니다. (SaveCurrentSettings)
SetVCPFeature 모니터에 대한 VCP(가상 제어판) 코드의 값을 설정합니다.

구조체

MC_TIMING_REPORT 모니터의 타이밍 보고서의 정보를 포함합니다.

열거형

MC_VCP_CODE_TYPE VCP(가상 제어판) 코드 형식에 대해 설명합니다.

lpmapi.h 헤더

이 헤더는 QOS(서비스 품질)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 서비스 품질(QOS)

lpmapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

LPM_AdmitRsvpMsg LPM_AdmitRsvpMsg 함수는 정책 기반 허용 제어 결정을 위해 RSVP 메시지를 LPM에 전달하기 위해 PCM에 의해 호출됩니다.
LPM_CommitResv LPM_CommitResv 함수는 LPM에서 예약 약정 결정을 얻기 위해 PCM에 의해 호출됩니다.
LPM_Deinitialize LPM_Deinitialize 함수를 사용하면 시스템 종료 또는 DSBM(지정된 서브넷 대역폭 관리자) 상태 변경으로 인해 PCM에서 초기화 해제를 지시할 수 있습니다.
LPM_DeleteState LPM_DeleteState 함수는 PCM에서 LPM의 RSVP 상태 정보를 삭제하기 위해 호출됩니다.
LPM_GetRsvpObjects LPM_GetRsvpObjects 함수를 사용하면 PCM에서 정책 데이터에 대한 LM을 쿼리할 수 있습니다.
LPM_Initialize LPM_Initialize 함수는 LPM(로컬 정책 모듈)을 초기화합니다.
LPM_IpAddressTable LPM_IpAddressTable 함수는 PCM에서 LPM이 초기화되는 Windows 2000 서버에 할당된 IP 주소 목록을 전달하는 데 사용됩니다.

콜백 함수

테이블 확장

CBADMITRESULT cbAdmitResult 함수는 LPM에서 LPM_AdmitRsvpMsg 요청에 대한 결과를 반환하는 데 사용됩니다.
CBGETRSVPOBJECTS cbGetRsvpObjects 함수는 LM이 LPM_GetRsvpObjects 요청에 대한 결과를 비동기적으로 반환하는 콜백 함수입니다.
PALLOCMEM PALLOCMEM 함수는 PCM에서 제공하는 메모리 할당 함수로, 정책 정보를 PCM에 반환할 때 메모리를 할당하는 데 사용됩니다.
PFREEMEM PFREEMEM 함수는 PCM에서 제공하는 메모리 해제 함수입니다.

구조

테이블 확장

ADSPEC ADSPEC 구조에는 RSVP에 대한 Adspec 메시지 정보가 포함됩니다.
CtrlLoadFlowspec CtrlLoadFlowspec 구조체에는 제어된 Load FLOWSPEC가 포함됩니다.
ERROR_SPEC ERROR_SPEC 구조에는 RSVP 오류 메시지가 포함됩니다.
Error_Spec_IPv4 Error_Spec_IPv4 구조체는 RSVP 전송에 대한 오류 코드 정보를 저장합니다.
FILTER_SPEC FILTER_SPEC 구조체는 RSVP FILTERSPEC에 대한 정보를 저장합니다.

Filter_Spec_IPv4 Filter_Spec_IPv4 구조체에는 IPv4 FILTERSPEC에 대한 정보가 포함됩니다.
Filter_Spec_IPv4GPI Filter_Spec_IPv4GPI 구조에는 IPv4 FILTERSPEC에 대한 일반화된 포트 ID 정보가 포함됩니다.
FLOW_DESC FLOW_DESC 구조에는 RSVP에 대한 흐름 설명자 정보가 포함되어 있습니다.
Gads_parms_t Gads_parms_t 구조는 보장된 서비스 Adspec 매개 변수를 저장합니다.
GenAdspecParams GenAdspecParams 구조체에는 일반 경로 특성화 매개 변수가 포함됩니다.
GenTspec GenTspec 구조체는 제네릭 Tspec 정보를 저장합니다.
GenTspecParms GenTspecParms 구조체는 제네릭 Tspec 매개 변수를 저장합니다.
GuarFlowSpec GuarFlowSpec 구조체에는 보장된 flowspec 정보가 포함되어 있습니다.
GuarRspec GuarRspec 구조체에는 보장된 Rspec 정보가 포함됩니다.
ID_ERROR_OBJECT ID_ERROR_OBJECT 구조에는 RSVP에 대한 ID 정책 요소에 대한 오류 메시지 정보가 포함되어 있습니다.
IntServFlowSpec IntServFlowSpec 구조에는 통합 서비스 흐름 지정에 대한 정보가 포함되어 있습니다.
IntServMainHdr IntServMainHdr 구조체는 통합 서비스 RSVP 개체의 헤더입니다.
IntServParmHdr

IntServParmHdr 구조체는 통합 서비스 매개 변수의 헤더입니다.
IntServiceHdr IntServiceHdr 구조체는 Integrated Services 서비스 개체의 헤더입니다.
IntServTspecBody IntServTspecBody 구조에는 RSVP Tspec에 대한 정보가 포함되어 있습니다.
IS_ADSPEC_BODY IS_ADSPEC_BODY 구조에는 통합 서비스 Adspec 정보가 포함됩니다.
IS_FLOWSPEC IS_FLOWSPEC 구조체는 통합 서비스 FLOWSPEC 개체를 저장합니다.
LPM_INIT_INFO LPM_INIT_INFO 구조에는 로컬 정책 모듈 초기화 정보가 포함됩니다.
LPMIPTABLE LPMIPTABLE 구조체에는 각 인터페이스에 대한 SNMP 인덱스, IP 주소 및 서브넷 마스크를 포함한 IP 정보가 포함됩니다. LPMIPTABLE 구조체는 Lpm_IpAddressTable 함수에 대한 인수로 제공됩니다.
POLICY_DATA POLICY_DATA 구조에는 RSVP 메시지에 대한 정책 데이터가 포함됩니다.
POLICY_DECISION POLICY_DECISION 구조에는 RSVP 정책 결정 정보가 포함됩니다.
POLICY_ELEMENT POLICY_ELEMENT(lpmapi.h) 구조체에는 RSVP 정책 요소가 포함됩니다.
QualAppFlowSpec QualAppFlowSpec 구조에는 정성적 애플리케이션에 대한 FLOWSPEC 정보가 포함되어 있습니다.
QualTspec QualTspec 구조에는 정성적 Tspec 정보가 포함됩니다.
QualTspecParms QualTspecParms 구조체에는 정성적 Tspec 매개 변수가 포함됩니다.

RESV_STYLE RESV_STYLE 구조체에는 RSVP RESV 스타일에 대한 정보가 포함되어 있습니다.
RSVP_HOP RSVP_HOP 구조에는 RSVP 사용 홉에 대한 정보가 포함됩니다.
Rsvp_Hop_IPv4 Rsvp_Hop_IPv4 구조체는 RSVP 사용 IPv4 홉에 대한 정보를 저장합니다.
RSVP_MSG_OBJS RSVP_MSG_OBJS 구조에는 RSVP 메시지 개체가 포함됩니다.
RSVP_SCOPE RSVP_SCOPE 구조는 RSVP 범위 정보를 제공합니다.
RSVP_SESSION RSVP_SESSION 구조체는 RSVP SESSION 메시지에 대한 정보를 저장합니다.
RsvpObjHdr RsvpObjHdr 구조체는 RSVP 메시지에 대한 개체 헤더를 제공합니다.
Scope_list_ipv4 Scope_list_ipv4 구조체에는 RSVP SCOPE 개체 정보가 포함됩니다.
SENDER_TSPEC SENDER_TSPEC 구조에는 RSVP 보낸 사람 Tspec에 대한 정보가 포함됩니다.
Session_IPv4 Session_IPv4 구조체는 IPv4 RSVP SESSION 개체에 대한 정보를 저장합니다.

Last updated on 2025. 10. 08.

lsalookup.h 헤더

아티클2023. 08. 24.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

lsalookup.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

LSA_OBJECT_ATTRIBUTES LSA_OBJECT_ATTRIBUTES 구조체는 LsaOpenPolicy 함수와 함께 사용하여 Policy 개체에 대한 연결의 특성을 지정합니다.
LSA_REFERENCED_DOMAIN_LIST LSA_REFERENCED_DOMAIN_LIST 구조에는 조회 작업에서 참조되는 도메인에 대한 정보가 포함됩니다.
LSA_STRING LSA(로컬 보안 기관) 함수에서 ANSI 문자열을 지정하는 데 사용됩니다.
LSA_TRANSLATED_NAME SID로 식별된 계정에 대한 정보를 반환하기 위해 LsaLookupSids 함수와 함께 사용됩니다.
LSA_TRANSLATED_SID2 계정 이름에 따라 검색되는 SID를 포함합니다.
LSA_TRUST_INFORMATION 도메인을 식별합니다.
LSA_UNICODE_STRING LSA_UNICODE_STRING 구조는 다양한 LSA(로컬 보안 기관) 함수에서 유니코드 문자열을 지정하는 데 사용됩니다.
POLICY_ACCOUNT_DOMAIN_INFO 시스템 계정 도메인의 이름 및 SID를 설정하고 쿼리하는 데 사용됩니다.

POLICY_DNS_DOMAIN_INFO POLICY_DNS_DOMAIN_INFO 구조는 Policy 개체와 연결된 주 도메인에 대한 DNS(도메인 이름 시스템) 정보를 설정하고 쿼리하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

lzexpand.h 헤더

아티클2023. 08. 24.

이 헤더는 데이터 액세스 및 스토리지에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지

lzexpand.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

GetExpandedNameA 파일이 Lempel-Ziv 알고리즘을 통해 압축된 경우 압축된 파일의 원래 이름을 검색합니다. (ANSI)
GetExpandedNameW 파일이 Lempel-Ziv 알고리즘을 통해 압축된 경우 압축된 파일의 원래 이름을 검색합니다. (유니코드)
LZClose LZOpenFile 함수를 사용하여 연 파일을 닫습니다.
LZCopy 원본 파일을 대상 파일에 복사합니다.
LZInit 파일 압축을 해제하는 데 필요한 내부 데이터 구조에 사용할 메모리를 할당한 다음, 파일을 만들고 초기화합니다.
LZOpenFileA 지정된 파일을 만들거나, 열거나, 다시 열거나, 삭제합니다. (ANSI)
LZOpenFileW 지정된 파일을 만들거나, 열거나, 다시 열거나, 삭제합니다. (유니코드)
LZRead 파일에서 지정된 바이트 수(최대)를 읽고 버퍼에 복사합니다.

LZSeek 파일 포인터를 시작 위치에서 지정된 바이트 수만큼 이동합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

madcapcl.h 헤더

아티클2023. 08. 24.

이 헤더는 MADCAP(멀티캐스트 주소 동적 클라이언트 할당 프로토콜)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 멀티캐스트 주소 MADCAP(동적 클라이언트 할당 프로토콜)

madcapcl.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

McastApiCleanup McastApiCleanup 함수는 McastApiStartup으로 할당된 리소스의 할당을 취소합니다. McastApiCleanup 함수는 McastApiStartup을 성공적으로 호출한 후에만 호출해야 합니다.
McastApiStartup McastApiStartup 함수는 요청 클라이언트와 시스템에 구현된 MADCAP 버전 간의 MADCAP 버전 협상을 용이하게 합니다.
McastEnumerateScopes McastEnumerateScopes 함수는 네트워크에서 사용할 수 있는 멀티캐스트 범위를 열거합니다.
McastGenUID McastGenUID 함수는 고유 식별자를 생성하며, 이후에 클라이언트가 주소를 요청하고 갱신하는 데 사용합니다.
McastReleaseAddress McastReleaseAddress 함수는 MCAST 서버에서 임대된 멀티캐스트 주소를 해제합니다.
McastRenewAddress McastRenewAddress 함수는 MADCAP 서버에서 하나 이상의 멀티캐스트 주소를 갱신합니다.
McastRequestAddress McastRequestAddress 함수는 MADCAP 서버에서 하나 이상의 멀티캐스트 주소를 요청합니다.

구조체

IPNG_ADDRESS IPNG_ADDRESS 공용 구조체는 IPv4(인터넷 프로토콜 버전 4) 및 IPv6(인터넷 프로토콜 버전 6) 주소를 제공합니다.
MCAST_CLIENT_UID MCAST_CLIENT_UID 구조는 각 멀티캐스트 요청에 대한 고유 클라이언트 식별자를 설명합니다.
MCAST_LEASE_REQUEST MCAST_LEASE_REQUEST 구조는 지정된 멀티캐스트 scope 대한 요청, 갱신 또는 릴리스 매개 변수를 정의합니다. MCAST_API_VERSION_1 구현에서는 한 번에 하나의 IP 주소만 할당할 수 있습니다.
MCAST_LEASE_RESPONSE MCAST_LEASE_RESPONSE 구조는 멀티캐스트 임대 요청에 응답하는 데 사용됩니다.
MCAST_SCOPE_CTX MCAST_SCOPE_CTX 구조는 멀티캐스트 주소와의 프로그래밍 방식 상호 작용을 위한 scope 컨텍스트를 정의합니다. MCAST_SCOPE_CTX 구조체는 MADCAP 주소를 할당, 갱신 또는 해제하기 위한 핸들로 다양한 MADCAP 함수에서 사용됩니다.
MCAST_SCOPE_ENTRY MCAST_SCOPE_ENTRY 구조체는 지정된 멀티캐스트 scope 대한 전체 정보 집합을 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

magnification.h 헤더

아티클2023. 08. 24.

이 헤더는 Magnification API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 배율 API

magnification.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

MagGetColorEffect 돋보기 컨트롤의 색 변환 행렬을 가져옵니다.
MagGetFullscreenColorEffect 전체 화면 돋보기와 연결된 색 변환 행렬을 검색합니다.
MagGetFullscreenTransform 전체 화면 돋보기에 대한 배율 설정을 검색합니다.
MagGetImageScalingCallback 이미지 크기 조정을 위한 사용자 지정 변환을 구현하는 등록된 콜백 함수를 검색합니다.
MagGetInputTransform 원본 사각형 및 대상 사각형으로 표시되는 펜 및 터치 입력에 대한 현재 입력 변환을 검색합니다.
MagGetWindowFilterList 확대 또는 배율에서 제외된 창 목록을 검색합니다.
MagGetWindowSource 확대되는 영역의 사각형을 가져옵니다.
MagGetWindowTransform 돋보기 컨트롤과 연결된 변환 행렬을 검색합니다.
MagInitialize

돋보기 런타임 개체를 만들고 초기화합니다.
MagSetColorEffect 돋보기 컨트롤의 색 변환 매트릭스를 설정합니다.
MagSetFullscreenColorEffect 전체 화면 돋보기와 연결된 색 변환 매트릭스를 변경합니다.
MagSetFullscreenTransform 전체 화면 돋보기에 대한 돋보기 설정을 변경합니다.
MagSetImageScalingCallback 외부 이미지 필터링 및 크기 조정을 위한 콜백 함수를 설정합니다.
MagSetInputTransform 원본 사각형 및 대상 사각형으로 표시되는 펜 및 터치 입력에 대한 현재 활성 입력 변환을 설정합니다.
MagSetWindowFilterList 확대할 창 목록 또는 확대에서 제외할 창 목록을 설정합니다.
MagSetWindowSource 배율 창의 원본 사각형을 설정합니다.
MagSetWindowTransform 돋보기 컨트롤의 변환 매트릭스를 설정합니다.
MagShowSystemCursor 시스템 커서를 표시하거나 숨깁니다.
MagUninitialize 돋보기 런타임 개체를 제거합니다.

콜백 함수

MagImageScalingCallback 이미지 크기 조정을 위한 사용자 지정 변환을 구현하는 콜백 함수의 프로토타입입니다.

구조체

MAGCOLOREFFECT 돋보기 컨트롤이 돋보기 화면 콘텐츠에 색 효과를 적용하는 데 사용하는 색 변환 매트릭스에 대해 설명합니다.
MAGIMAGEHEADER 이미지 형식을 설명합니다.
MAGTRANSFORM 돋보기 컨트롤이 화면 콘텐츠를 확대하는 데 사용하는 변환 매트릭스에 대해 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

manipulations.h 헤더

아티클2023. 03. 11.

이 헤더는 터치 입력에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 터치 입력

manipulations.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

_IManipulationEvents 조작 및 관성 이벤트를 처리합니다.
IInertiaProcessor IInertiaProcessor 인터페이스는 Windows Touch의 개체 동작에 대한 계산을 처리합니다.
IManipulationProcessor IManipulationProcessor는 멀티터치 입력을 모니터링하고 응답하는 기능을 제공합니다.

열거형

MANIPULATION_PROCESSOR_MANIPULATIONS MANIPULATION_PROCESSOR_MANIPULATIONS 열거형은 대상 개체에 적용할 수 있는 다양한 종류의 조작입니다.

mapi.h 헤더

아티클2023. 08. 24.

이 헤더는 MAPI(Windows Messaging)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- MAPI(Windows 메시징)

mapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

MAPIFreeBuffer MAPIFreeBuffer 함수(mapi.h)는 메시징 시스템에서 할당한 메모리를 해제합니다.

콜백 함수

MAPIADDRESS MAPIAddress 함수는 주소 목록 항목 집합을 만들거나 수정합니다.
MAPIDELETEMAIL MAPIDeleteMail 함수는 메시지를 삭제합니다.
MAPIDETAILS MAPIDetails 함수는 선택한 주소 목록 항목의 세부 정보가 포함된 대화 상자를 표시합니다.
MAPIFINDNEXT MAPIFindNext 함수는 지정된 유형의 들어오는 메시지의 다음(또는 첫 번째) 메시지 식별자를 검색합니다.
MAPILOGOFF MAPILogoff 함수는 메시징 시스템으로 세션을 종료합니다.
MAPILOGON

MAPILogon 함수는 간단한 MAPI 세션을 시작하여 기본 메시지 저장소 및 주소록 공급자를 로드합니다.
MAPIREADMAIL MAPIReadMail 함수는 읽기용 메시지를 검색합니다.
MAPIRESOLVENAME MAPIResolveName 함수는 사용자가 입력한 메시지 받는 사람의 이름을 명확한 주소 목록 항목으로 변환합니다.
MAPISAVEMAIL MAPISaveMail 함수는 메시지를 메시지 저장소에 저장합니다.
MAPISENDDOCUMENTS MAPISendDocuments 함수는 하나 이상의 첨부 파일과 표지가 있는 표준 메시지를 보냅니다.
MAPISENDMAIL ANSI 메시지를 보냅니다.
MAPISENDMAILW 유니코드 메시지를 보냅니다. 이 함수는 ANSI 함수 MAPISendMail을 대체합니다.

구조체

MapiFileDesc MapiFileDesc 구조체에는 임시 파일로 저장된 메시지 첨부 파일이 포함된 파일에 대한 정보가 포함되어 있습니다.
MapiFileDescW MapiFileDescW 구조체에는 임시 파일로 저장된 메시지 첨부 파일이 포함된 파일에 대한 정보가 포함됩니다. 해당 파일에는 정적 OLE 개체, 포함된 OLE 개체, 포함된 메시지 및 기타 형식의 파일이 포함될 수 있습니다.
MapiFileTagExt MapiFileTagExt 구조체는 생성 시 메시지 첨부 파일의 형식과 현재 인코딩 형식을 지정하므로 대상에서 원래 형식으로 복원할 수 있습니다.

MapiMessage MapiMessage 구조체에는 메시지에 대한 정보가 포함됩니다. 유니코드 지원의 경우 MapiMessageW 구조를 사용합니다.
MapiMessageW MapiMessageW 구조체에는 메시지에 대한 정보가 포함됩니다.
MapiRecipDesc MapiRecipDesc 구조에는 메시지 보낸 사람 또는 받는 사람에 대한 정보가 포함됩니다. 유니코드 지원의 경우 MapiRecipDescW 구조를 사용합니다.
MapiRecipDescW MapiRecipDescW 구조에는 메시지 보낸 사람 또는 받는 사람에 대한 정보가 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mapiunicodehelp.h 헤더

아티클2023. 08. 24.

이 헤더는 MAPI(Windows 메시징)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- MAPI(Windows 메시징)

mapiunicodehelp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

MAPISendMailHelper 유니코드 메시지 정보를 가져와 MAPISendMailW를 사용하여 메시지를 보내거나 필요한 경우 메시지를 ANSI로 변환하고 MAPISendMail을 사용하여 메시지를 보냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mbnapi.h 헤더

아티클2023. 03. 09.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- CIMWin32 WMI 공급자
- 모바일 광대역

mbnapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IMbnConnection 디바이스의 네트워크 연결을 나타냅니다.
IMbnConnectionContext 연결 컨텍스트를 관리합니다.
IMbnConnectionContextEvents 이 알림 인터페이스는 비동기 프로비전된 컨텍스트 이벤트를 처리하는 데 사용됩니다.
IMbnConnectionEvents 이 알림 인터페이스는 비동기 연결 요청의 변경 및 완료 상태에 대해 애플리케이션에 신호를 표시합니다.
IMbnConnectionManager IMbnConnection 개체 및 연결 알림에 대한 액세스를 제공합니다.
IMbnConnectionManagerEvents 이 알림 인터페이스는 시스템에서 IMbnConnection 인터페이스의 도착 및 제거에 대해 애플리케이션에 신호를 전송합니다.
IMbnConnectionProfile 이 인터페이스는 모바일 광대역 프로필에 저장된 연결 매개 변수 및 기본 설정에 액세스합니다.

IMbnConnectionProfileEvents 이 알림 인터페이스는 IMbnConnectionProfile 메서드가 모바일 광대역 프로필 상태를 변경할 때 애플리케이션에 신호를 전송합니다.
IMbnConnectionProfileManager 연결 프로필 및 연결 알림에 대한 액세스를 제공합니다.
IMbnConnectionProfileManagerEvents 이 알림 인터페이스는 시스템에서 IMbnConnectionProfile 인터페이스의 도착 및 제거에 대해 애플리케이션에 신호를 전송합니다.
IMbnDeviceService 특정 모바일 광대역 디바이스에서 디바이스 서비스와 통신할 수 있습니다.
IMbnDeviceServicesContext 시스템에서 모바일 광대역 디바이스 개체를 열거하고 검색할 수 있습니다.
IMbnDeviceServicesEvents 시스템의 모바일 광대역 디바이스 서비스와 관련된 알림 이벤트에 대해 애플리케이션에 신호를 전송합니다.
IMbnDeviceServicesManager IMbnDeviceServicesContext 개체 및 모바일 광대역 디바이스 서비스 알림에 대한 액세스를 제공합니다.
IMbnInterface 모바일 광대역 디바이스를 나타냅니다.
IMbnInterfaceEvents 이 인터페이스는 비동기 IMbnInterface 메서드 호출 및 디바이스 상태 변경을 처리하는 데 사용되는 알림 인터페이스입니다.
IMbnInterfaceManager IMbnInterface 개체 및 알림에 대한 액세스를 제공합니다.
IMbnInterfaceManagerEvents 이 알림 인터페이스는 시스템에서 디바이스의 도착 및 제거에 대해 애플리케이션에 신호를 전송합니다.

IMbnMultiCarrier 이 인터페이스는 지원되는 모바일 광대역 디바이스의 다중 통신 사업자 기능을 노출합니다.
IMbnMultiCarrierEvents 이 인터페이스는 비동기 IMbnMultiCarrier 메서드 호출을 처리하는 데 사용되는 알림 인터페이스입니다.
IMbnPin 디바이스 PIN을 나타냅니다.
IMbnPinEvents 이 인터페이스는 비동기 PIN 요청이 완료된 시기를 나타내는 데 사용되는 알림 인터페이스입니다.
IMbnPinManager 디바이스 PIN에 대한 중요한 세부 정보를 제공합니다.
IMbnPinManagerEvents PIN Manager 이벤트가 발생한 시기를 나타내는 데 사용되는 알림 인터페이스입니다.
IMbnRadio IMbnRadio 인터페이스는 모바일 광대역 디바이스의 라디오 상태를 쿼리하고 업데이트하는 데 사용됩니다.
IMbnRadioEvents 라디오 상태의 변경과 상태의 프로그래밍 방식 변경 완료를 나타내는 데 사용되는 알림 인터페이스입니다.
IMbnRegistration 네트워크 등록 데이터에 대한 액세스를 제공합니다.
IMbnRegistrationEvents 등록 이벤트가 발생한 시기를 나타내는 데 사용되는 알림 인터페이스입니다.
IMbnServiceActivation 셀룰러 서비스 활성화를 위한 통과 메커니즘입니다.

IMbnServiceActivationEvents 이 알림 인터페이스는 서비스 활성화 요청의 완료에 대해 애플리케이션에 알릴 수 있습니다.
IMbnSignal 모바일 광대역 연결의 무선 신호 품질을 가져옵니다.
IMbnSignalEvents 신호 이벤트가 발생했음을 나타내는 데 사용되는 알림 인터페이스입니다.
IMbnSms 메시지 송수신 및 메시지 구성 제어를 위한 SMS 인터페이스입니다.
IMbnSmsConfiguration 디바이스의 SMS 구성에 대한 액세스를 제공합니다.
IMbnSmsEvents 이 알림 인터페이스는 SMS 작업의 완료 상태와 디바이스 SMS 상태의 변경 내용으로 애플리케이션에 신호를 전송합니다.
IMbnSmsReadMsgPdu 디바이스 메모리에서 읽은 SMS 메시지를 나타내는 속성의 컬렉션입니다.
IMbnSmsReadMsgTextCdma 디바이스 메모리에서 읽은 CDMA 형식 SMS 메시지를 나타내는 속성의 컬렉션입니다.
IMbnSubscriberInformation 구독자 정보에 대한 액세스를 제공합니다.
IMbnVendorSpecificEvents 이 알림 인터페이스는 공급업체별 작업의 완료 상태 및 디바이스 상태의 기타 공급업체별 변경 내용을 애플리케이션에 알릴 수 있습니다.
IMbnVendorSpecificOperation 애플리케이션에서 기본 모바일 광대역 미니포트 드라이버로 요청을 전달하는 인터페이스입니다.

구조체

MBN_CONTEXT MBN_CONTEXT 구조는 연결 컨텍스트에 대한 정보를 저장합니다.
MBN_DEVICE_SERVICE MBN_DEVICE_SERVICE 구조는 모바일 광대역 디바이스 서비스에 대한 정보를 제공합니다.
MBN_INTERFACE_CAPS MBN_INTERFACE_CAPS 구조체는 인터페이스 기능을 나타냅니다.
MBN_PIN_INFO MBN_PIN_INFO 구조체는 디바이스의 현재 PIN 상태를 나타냅니다.
MBN_PROVIDER MBN_PROVIDER 구조는 네트워크 서비스 공급자를 나타냅니다.
MBN_PROVIDER2 MBN_PROVIDER2 구조는 네트워크 서비스 공급자를 나타냅니다. IMbnMultiCarrier 인터페이스의 여러 공급자별 메서드에서 사용되며 다중 캐리어를 지원하기 위해 MBN_PROVIDER 확장을 제공합니다.
MBN_SMS_FILTER MBN_SMS_FILTER 구조에는 SMS 메시지 집합을 설명하는 값이 포함되어 있습니다.
MBN_SMS_STATUS_INFO MBN_SMS_STATUS_INFO 구조에는 디바이스의 SMS 메시지 저장소 상태가 포함됩니다.

열거형

MBN_ACTIVATION_STATE MBN_ACTIVATION_STATE 열거형 형식은 현재 데이터 연결 상태를 나타냅니다.
MBN_AUTH_PROTOCOL MBN_AUTH_PROTOCOL 열거형 형식은 PDP(패킷 데이터 프로토콜) 활성화에 사용되는 인증 프로토콜을 지정합니다.

MBN_BAND_CLASS MBN_BAND_CLASS 열거형 형식은 빈도 대역 클래스를 정의합니다.
MBN_CELLULAR_CLASS MBN_CELLULAR_CLASS 열거형 형식은 셀룰러 디바이스의 유형을 정의합니다.
MBN_COMPRESSION MBN_COMPRESSION 열거형 형식은 헤더 및 데이터에 대한 데이터 링크에서 압축을 사용할지 여부를 지정합니다.
MBN_CONNECTION_MODE MBN_CONNECTION_MODE 열거형 형식은 요청된 연결 모드를 지정합니다.
MBN_CONTEXT_CONSTANTS MBN_CONTEXT_CONSTANTS 열거형 형식은 MBN_CONTEXT 구조체의 멤버가 지원하는 최대 문자열 길이를 지정합니다.
MBN_CONTEXT_TYPE MBN_CONTEXT_TYPE 열거형 형식은 표시된 컨텍스트 형식을 지정합니다.
MBN_CTRL_CAPS MBN_CTRL_CAPS 열거형 형식은 모든 모바일 광대역 디바이스 제어 기능을 비트 필드로 나타냅니다.
MBN_DATA_CLASS MBN_DATA_CLASS 열거형 형식은 공급자가 지원하는 데이터 클래스를 지정합니다.
MBN_DEVICE_SERVICES_INTERFACE_STATE . (MBN_DEVICE_SERVICES_INTERFACE_STATE)
MBN_INTERFACE_CAPS_CONSTANTS MBN_INTERFACE_CAPS_CONSTANTS 열거형 형식은 MBN_INTERFACE_CAPS 구조체의 다양한 요소에서 사용하는 문자열 값의 최대 길이를 정의합니다.
MBN_MSG_STATUS MBN_MSG_STATUS 열거형 형식은 처리 중인 메시지의 형식을 정의합니다.

MBN_PIN_CONSTANTS MBN_PIN_CONSTANTS 열거형 형식은 MBN_PIN_INFO 구조체에서 사용하는 상수 값을 정의합니다.
MBN_PIN_FORMAT MBN_PIN_FORMAT 열거형 형식은 PIN이 숫자인지 영숫자인지를 나타냅니다.
MBN_PIN_MODE MBN_PIN_MODE 열거형 형식은 PIN 형식을 사용할 수 있는지를 나타냅니다.
MBN_PIN_STATE MBN_PIN_STATE 열거형 형식은 모바일 광대역 디바이스의 현재 PIN 상태를 나타냅니다.
MBN_PIN_TYPE MBN_PIN_TYPE 열거형 형식은 인터페이스에 저장된 정보의 잠금을 해제하는 데 필요한 암호 유형을 나타냅니다.
MBN_PROVIDER_CONSTANTS MBN_PROVIDER_CONSTANTS 열거형 형식에는 MBN_PROVIDER 멤버의 버퍼 길이를 정의하는 값이 포함되어 있습니다.
MBN_PROVIDER_STATE MBN_PROVIDER_STATE 열거형 형식은 공급자 항목에 태그를 지정할 수 있는 다양한 상태를 지정합니다.
MBN_RADIO MBN_RADIO 열거형 형식은 디바이스 라디오가 켜지거나 꺼져 있는지 여부를 나타냅니다.
MBN_READY_STATE MBN_READY_STATE 열거형 형식에는 모바일 광대역 디바이스가 셀룰러 네트워크 트래픽 작업에 참여할 준비가 되었음을 나타내는 값이 포함되어 있습니다.
MBN_REGISTER_MODE MBN_REGISTER_MODE 열거형 형식은 디바이스의 네트워크 선택 모드를 나타냅니다.
MBN_REGISTER_STATE MBN_REGISTER_STATE 열거형 유형은 모바일 광대역 디바이스의 네트워크 등록 상태를 나타냅니다.

MBN_REGISTRATION_CONSTANTS MBN_REGISTRATION_CONSTANTS 열거형 형식에는 IMbnRegistration 인터페이스 작업에서 사용하는 특정 값이 포함됩니다.
MBN_SIGNAL_CONSTANTS MBN_SIGNAL_CONSTANTS 열거형 형식에는 IMbnSignal 인터페이스 작업에서 사용하는 특정 값이 포함됩니다.
MBN_SMS_CAPS MBN_SMS_CAPS 열거형 형식에는 SMS 기능을 지정하는 비트 필드 값이 포함되어 있습니다.
MBN_SMS_CDMA_ENCODING MBN_SMS_CDMA_ENCODING 열거형 형식은 CDMA에 대한 문자 인코딩 형식을 지정합니다.
MBN_SMS_CDMA_LANG MBN_SMS_CDMA_LANG 열거형 형식은 CDMA 메시지에서 사용할 수 있는 다양한 언어를 나타냅니다.
MBN_SMS_FLAG MBN_SMS_FLAG 열거형 형식은 SMS 메시지 클래스를 지정합니다.
MBN_SMS_FORMAT SMS 메시지의 형식입니다.
MBN_SMS_STATUS_FLAG MBN_SMS_STATUS_FLAG 열거형 유형은 디바이스의 SMS 메시지 저장소 상태를 나타냅니다.
MBN_VOICE_CALL_STATE MBN_VOICE_CALL_STATE 열거형 유형은 디바이스의 현재 음성 통화 상태를 지정합니다.
MBN_VOICE_CLASS MBN_VOICE_CLASS 열거형 형식은 디바이스의 음성 기능 및 데이터 서비스와 상호 작용하는 방법을 지정합니다.
WWAEXT_SMS_CONSTANTS MBN_SMS_CONSTANTS 열거형 형식에는 SMS 상수 값이 포함됩니다.

mcx.h 헤더

아티클2023. 03. 11.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

mcx.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

MODEMDEVCAPS 모뎀의 기능에 대한 정보를 포함합니다.
MODEMSETTINGS 모뎀의 구성에 대한 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mdhcp.h 헤더

아티클2023. 08. 24.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 장애 조치(failover) 클러스터 WMI 공급자
- 전화 통신 애플리케이션 프로그래밍 인터페이스

mdhcp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumMcastScope IEnumMcastScope 인터페이스는 IMcastScope 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다. IMcastAddressAllocation::EnumerateScopes 메서드는 IEnumMcastScope에 대한 포인터를 반환합니다.
IMcastAddressAllocation IMcastAddressAllocation은 멀티캐스트 주소 할당을 위한 기본 인터페이스입니다. 애플리케이션은 이 인터페이스에서 COM CoCreateInstance 함수를 호출하여 멀티캐스트 클라이언트 인터페이스 개체를 만듭니다.
IMcastLeaseInfo IMcastLeaseInfo 인터페이스는 멀티캐스트 주소 할당과 관련된 정보를 얻거나 설정할 수 있는 메서드를 노출합니다. IMcastAddressAllocation::CreateLeaseInfo를 호출하여 IMcastLease 개체를 만듭니다.
IMcastScope IMcastScope 인터페이스는 IMcastAddressAllocation::EnumerateScopes 또는 IMcastAddressAllocation::get_Scopes 호출하여 가져옵니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mdmregistration.h 헤더

아티클2023. 08. 24.

이 헤더는 모바일 장치 관리 등록에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 모바일 장치 관리 등록

mdmregistration.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DiscoverManagementService MDM 서비스를 검색합니다.
DiscoverManagementServiceEx 후보 서버를 사용하여 MDM 서비스를 검색합니다.
GetDeviceManagementConfigInfo 공급자 ID와 연결된 구성 정보를 가져옵니다.
GetDeviceRegistrationInfo 디바이스 등록 정보를 검색합니다.
GetManagementAppHyperlink MDM 서비스와 연결된 관리 앱 하이퍼링크를 검색합니다.
IsDeviceRegisteredWithManagement 디바이스가 MDM 서비스에 등록되어 있는지 확인합니다.
IsManagementRegistrationAllowed 로컬 정책에서 MDM 등록이 허용되는지 여부를 확인합니다.
RegisterDeviceWithManagement [MS-MDE]:_Mobile 디바이스 등록 프로토콜을 사용하여 MDM 서비스에 디바이스를 등록합니다.
RegisterDeviceWithManagementUsingAADCredentials

AAD(Azure Active Directory) 자격 증명을 사용하여 MDM 서비스에 디바이스를 등록합니다.
RegisterDeviceWithManagementUsingAADDeviceCredentials AAD(Azure Active Directory) 디바이스 자격 증명을 사용하여 MDM 서비스에 디바이스를 등록합니다.
SetDeviceManagementConfigInfo 공급자 ID와 연결된 구성 정보를 설정합니다.
SetManagedExternally 디바이스가 외부에서 관리되고 MDM 서비스에 등록되지 않음을 MDM 에이전트에 나타냅니다.
UnregisterDeviceWithManagement MDM 서비스를 사용하여 디바이스 등록을 취소합니다.

구조체

MANAGEMENT_SERVICE_INFO 관리 서비스에 대한 엔드포인트 및 정보를 포함합니다.

열거형

REGISTRATION_INFORMATION_CLASS 디바이스 등록에 대한 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

memoryapi.h 헤더

아티클2023. 03. 11.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

memoryapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

AllocateUserPhysicalPages 지정된 프로세스의 AWE(주소 창 확장) 영역 내에서 매핑 및 매핑 해제할 실제 메모리 페이지를 할당합니다.
AllocateUserPhysicalPagesNuma 지정된 프로세스의 AWE(주소 창 확장) 영역 내에서 매핑 및 매핑 해제할 실제 메모리 페이지를 할당하고 실제 메모리에 대한 NUMA 노드를 지정합니다.
CreateFileMapping2 지정된 파일에 대한 명명되거나 명명되지 않은 파일 매핑 개체를 만들거나 엽니다. 물리적 메모리에 대한 기본 NUMA 노드를 확장 매개 변수로 지정할 수 있습니다. *ExtendedParameters* 매개 변수를 참조하세요.
CreateFileMappingFromApp Windows 스토어 앱에서 지정된 파일에 대한 명명되거나 명명되지 않은 파일 매핑 개체를 만들거나 엽니다.
CreateFileMappingNumaW 지정된 파일에 대한 명명되거나 명명되지 않은 파일 매핑 개체를 만들거나 열고 실제 메모리에 대한 NUMA 노드를 지정합니다. (CreateFileMappingNumaW)
CreateFileMappingW 지정된 파일에 대한 명명되거나 명명되지 않은 파일 매핑 개체를 만들거나 엽니다. (CreateFileMappingW)
CreateMemoryResourceNotification 메모리 리소스 알림 개체를 만듭니다.

DiscardVirtualMemory 메모리를 커밋 해제하지 않고 메모리 페이지 범위의 메모리 내용을 삭제합니다. 삭제된 메모리의 내용은 정의되지 않았으며 애플리케이션에서 다시 작성해야 합니다.
FlushViewOfFile 파일의 매핑된 보기 내에서 디스크에 바이트 범위를 씁니다.
FreeUserPhysicalPages AllocateUserPhysicalPages 또는 AllocateUserPhysicalPagesNuma를 사용하여 이전에 할당된 실제 메모리 페이지를 해제합니다.
GetLargePageMinimum 큰 페이지의 최소 크기를 검색합니다.
GetMemoryErrorHandlingCapabilities 시스템의 메모리 오류 처리 기능을 가져옵니다.
GetProcessWorkingSetSize 지정된 프로세스의 최소 및 최대 작업 집합 크기를 검색합니다. (GetProcessWorkingSetSize)
GetProcessWorkingSetSizeEx 지정된 프로세스의 최소 및 최대 작업 집합 크기를 검색합니다. (GetProcessWorkingSetSizeEx)
GetSystemFileCacheSize 시스템 캐시의 작업 집합에 대한 현재 크기 제한을 검색합니다.
GetWriteWatch 가상 메모리 영역에 기록된 페이지의 주소를 검색합니다.
MapUserPhysicalPages 이전에 할당된 실제 메모리 페이지를 AWE(주소 창 확장) 영역의 지정된 주소에 매핑합니다. (MapUserPhysicalPages)
MapViewOfFile 파일 매핑 보기를 호출 프로세스의 주소 공간에 매핑합니다.

MapViewOfFile2 파일 또는 페이지 파일 지원 섹션의 보기를 지정된 프로세스의 주소 공간에 매핑합니다. (MapViewOfFile2)
MapViewOfFile3 파일 또는 페이지 파일 지원 섹션의 보기를 지정된 프로세스의 주소 공간에 매핑합니다. (MapViewOfFile3)
MapViewOfFile3FromApp 파일 매핑 보기를 호출 Windows 스토어 앱의 주소 공간에 매핑합니다. (MapViewOfFile3FromApp)
MapViewOfFileEx 파일 매핑 보기를 호출 프로세스의 주소 공간에 매핑합니다. 호출자는 필요에 따라 보기에 대해 제안된 기본 메모리 주소를 지정할 수 있습니다.
MapViewOfFileFromApp 파일 매핑 보기를 호출 Windows 스토어 앱의 주소 공간에 매핑합니다. (MapViewOfFileFromApp)
MapViewOfFileNuma2 파일 또는 페이지 파일 지원 섹션의 보기를 지정된 프로세스의 주소 공간에 매핑합니다. (MapViewOfFileNuma2)
OfferVirtualMemory 메모리 페이지 범위에 포함된 데이터가 애플리케이션에서 더 이상 필요하지 않으며 필요한 경우 시스템에서 삭제할 수 있음을 나타냅니다.
OpenFileMappingFromApp 명명된 파일 매핑 개체를 엽니다. (OpenFileMappingFromApp)
OpenFileMappingW 명명된 파일 매핑 개체를 엽니다. (OpenFileMappingW)
PrefetchVirtualMemory 프로세스 주소 공간에서 잠재적으로 불협화음이 발생할 수 있는 가상 주소 범위를 메모리로 가져오는 효율적인 메커니즘을 제공합니다.
QueryMemoryResourceNotification 지정된 메모리 리소스 개체의 상태를 검색합니다.

QueryVirtualMemoryInformation QueryVirtualMemoryInformation 함수는 지정된 프로세스의 가상 주소 공간 내에 있는 페이지 또는 페이지 집합에 대한 정보를 반환합니다.
ReadProcessMemory 지정된 프로세스의 메모리 영역에서 데이터를 읽습니다. 읽을 전체 영역에 액세스할 수 있어야 합니다. 그렇지 않으면 작업이 실패합니다.
ReclaimVirtualMemory OfferVirtualMemory를 사용하여 시스템에 제공된 다양한 메모리 페이지를 회수합니다.
RegisterBadMemoryNotification 하나 이상의 잘못된 메모리 페이지가 검색될 때 호출되는 잘못된 메모리 알림을 등록합니다.
ResetWriteWatch 가상 메모리 영역에 대한 쓰기 추적 상태를 다시 설정합니다. GetWriteWatch 함수에 대한 후속 호출은 다시 설정 작업 이후에 작성된 보고서 페이지만 보고합니다.
SetProcessValidCallTargets CFG(Control Flow Guard)에 유효한 간접 호출 대상 목록을 제공하고 유효한 것으로 표시할지 여부를 지정합니다.
SetProcessWorkingSetSize 지정된 프로세스의 최소 및 최대 작업 집합 크기를 설정합니다. (SetProcessWorkingSetSize)
SetProcessWorkingSetSizeEx 지정된 프로세스의 최소 및 최대 작업 집합 크기를 설정합니다. (SetProcessWorkingSetSizeEx)
SetSystemFileCacheSize 파일 시스템 캐시에 대한 작업 집합의 크기를 제한합니다.
UnmapViewOfFile 호출 프로세스의 주소 공간에서 파일의 매핑된 뷰의 매핑을 해제합니다.
UnmapViewOfFile2 파일 또는 페이지 파일 지원 섹션의 이전에 매핑된 뷰의 매핑을 해제합니다.

UnmapViewOfFileEx 추가 플래그 매개 변수를 사용하는 UnmapViewOfFile의 확장 버전입니다.
UnregisterBadMemoryNotification 지정된 잘못된 메모리 알림 핸들을 닫습니다.
VirtualAlloc 호출 프로세스의 가상 주소 공간에서 페이지 영역의 상태를 예약, 커밋 또는 변경합니다. (VirtualAlloc)
VirtualAlloc2 지정된 프로세스의 가상 주소 공간 내에서 메모리 영역의 상태를 예약, 커밋 또는 변경합니다. 이 함수는 할당한 메모리를 0으로 초기화합니다. (VirtualAlloc2)
VirtualAlloc2FromApp 호출 프로세스의 가상 주소 공간에서 페이지 영역의 상태를 예약, 커밋 또는 변경합니다. (VirtualAlloc2FromApp)
VirtualAllocEx 지정된 프로세스의 가상 주소 공간 내에서 메모리 영역의 상태를 예약, 커밋 또는 변경합니다. 이 함수는 할당한 메모리를 0으로 초기화합니다. (VirtualAllocEx)
VirtualAllocExNuma 지정된 프로세스의 가상 주소 공간 내에서 메모리 영역의 상태를 예약, 커밋 또는 변경하고 실제 메모리에 대한 NUMA 노드를 지정합니다.
VirtualAllocFromApp 호출 프로세스의 가상 주소 공간에서 페이지 영역의 상태를 예약, 커밋 또는 변경합니다. (VirtualAllocFromApp)
VirtualFree 호출 프로세스의 가상 주소 공간 내에서 페이지 영역을 해제, 커밋 해제 또는 해제합니다.
VirtualFreeEx 지정된 프로세스의 가상 주소 공간 내에서 메모리 영역을 해제, 커밋 해제 또는 해제합니다.

VirtualLock 프로세스의 가상 주소 공간의 지정된 영역을 실제 메모리에 잠가서 해당 지역에 대한 후속 액세스에서 페이지 오류가 발생하지 않도록 합니다.
VirtualProtect 호출 프로세스의 가상 주소 공간에서 커밋된 페이지의 영역에 대한 보호를 변경합니다. (VirtualProtect)
VirtualProtectEx 지정된 프로세스의 가상 주소 공간에서 커밋된 페이지의 영역에 대한 보호를 변경합니다.
VirtualProtectFromApp 호출 프로세스의 가상 주소 공간에서 커밋된 페이지의 영역에 대한 보호를 변경합니다. (VirtualProtectFromApp)
Virtualquery 호출 프로세스의 가상 주소 공간에서 페이지 범위에 대한 정보를 검색합니다.
VirtualQueryEx 지정된 프로세스의 가상 주소 공간 내의 페이지 범위에 대한 정보를 검색합니다.
VirtualUnlock 프로세스의 가상 주소 공간에서 지정된 범위의 페이지를 잠금 해제하여 필요한 경우 시스템에서 페이지를 페이징 파일로 교환할 수 있도록 합니다.
WriteProcessMemory 지정된 프로세스의 메모리 영역에 데이터를 씁니다. 기록할 전체 영역에 액세스할 수 있어야 합니다. 그렇지 않으면 작업이 실패합니다.

구조체

WIN32_MEMORY_RANGE_ENTRY 메모리 범위를 지정합니다.

WIN32_MEMORY_REGION_INFORMATION 메모리 영역에 대한 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

memorybuffer.h 헤더

아티클2023. 08. 24.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mergemod.h 헤더

아티클2023. 08. 24.

이 헤더는 애플리케이션 설치 및 서비스에 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 애플리케이션 설치 및 서비스

mergemod.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IMsmConfigurableItem IMsmConfigurableItem 인터페이스는 ModuleConfiguration 테이블에서 단일 행을 관리합니다. 이는 모듈에서 구성 가능한 단일 "특성"입니다. 인터페이스는 ModuleConfiguration 테이블의 각 열에 대해 하나씩 읽기 전용 속성으로 구성됩니다.
IMsmConfigureModule IMsmConfigureModule 인터페이스는 콜백 인터페이스입니다. 이렇게 하면 클라이언트가 병합 프로세스 중에 병합 구성 정보를 제공할 수 있습니다.
IMsmDependency IMsmDependency 인터페이스는 단일 모듈 종속성에 대한 세부 정보를 검색합니다.
IMsmError IMsmError 인터페이스는 단일 병합 오류에 대한 세부 정보를 검색합니다.
IMsmGetFiles IMsmGetFiles 인터페이스를 사용하면 클라이언트가 모듈의 특정 언어로 필요한 파일을 검색할 수 있습니다.
IMsmMerge IMsmMerge 인터페이스 및 IMsmMerge2 인터페이스는 Merge 개체에 대한 인터페이스를 제공합니다.
IMsmMerge2 IMsmMerge 인터페이스 및 IMsmMerge2 인터페이스는 Merge 개체에 대한 인터페이스를 제공합

니다. IMsmMerge2 인터페이스는 클라이언트 병합 도구가 새로운 구성 가능한 모듈 기능을 활용하는 방법을 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

messagedispatcherapi.h 헤더

아티클2023. 08. 24.

이 헤더는 COM(구성 요소 개체 모델)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)

messagedispatcherapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CoHandlePriorityEventsFromMessagePump WINDOWS 메시지를 디스패치한 후 ASTA 스레드에서 메시지 디스패처가 호출하여 짧은 실행 인프라 COM 호출 및 메시지 간에 디스패치할 기타 높은 우선 순위 또는 짧은 실행 COM 작업을 위한 기회를 제공합니다.
CoSetMessageDispatcher ASTA 스레드의 COM 대기 API 내에서 디스패치할 수 있는 창 메시지가 있을 때 호출할 스레드별 메시지 디스패처를 등록하거나 등록 취소합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

methodco.h 헤더

아티클2023. 08. 24.

이 헤더는 Windows Management Instrumentation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Management Instrumentation

methodco.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

MethodContext MethodContext 클래스는 공급자가 IWbemClassObject 정보를 얻거나 설정하는 데 사용되는 구조체에 대한 포인터입니다. WMI는 포인터를 기반으로 하는 기능을 구현하지 않습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mfapi.h 헤더

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft Media Foundation
- 스트리밍 미디어 DDI 참조

mfapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

기능

테이블 확장

DEFINE_MEDIATYPE_GUID FOURCC 코드, D3DFORMAT 값 또는 오디오 형식 형식에서 미디어 하위 형식 GUID를 정의합니다.
MFAddPeriodicCallback 고정 간격으로 호출할 콜백 함수를 설정합니다. (MFAddPeriodicCallback)
MFAllocateSerialWorkQueue 작업 항목을 직렬화하도록 보장되는 작업 큐를 만듭니다.
MFAllocateWorkQueue 새 작업 큐를 만듭니다. (MFAllocateWorkQueue)
MFAllocateWorkQueueEx 새 작업 큐를 만듭니다. (MFAllocateWorkQueueEx)
MFAverageTimePerFrameToFrameRate 비디오 프레임의 평균 기간에서 프레임 속도를 초당 프레임 단위로 계산합니다.
MFBeginCreateFile 파일에서 바이트 스트림을 만드는 비동기 요청을 시작합니다.
MFBeginRegisterWorkQueueWithMMCSS 작업 큐를 MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업에 연결합니다. (MFBeginRegisterWorkQueueWithMMCSS)

MFBeginRegisterWorkQueueWithMMCSSEx 작업 큐를 MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업에 연결합니다. (MFBeginRegisterWorkQueueWithMMCSSEx)
MFBeginUnregisterWorkQueueWithMMCSS MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업에서 작업 큐의 등록을 취소합니다. (MFBeginUnregisterWorkQueueWithMMCSS)
MFCalculateBitmapImageSize 비디오 형식의 이미지 크기를 검색합니다.
MFCalculateImageSize 압축되지 않은 비디오 형식의 이미지 크기(바이트)를 검색합니다. (MFCalculateImageSize)
MFCancelCreateFile 파일에서 바이트 스트림을 만드는 비동기 요청을 취소합니다.
MFCancelWorkItem MFScheduleWorkItem 또는 MFScheduleWorkItemEx로 예약된 비동기 작업을 취소하려고 시도합니다.
MFCombineSamples 결합된 기간이 지정된 기간을 초과하지 않는 경우 미디어 샘플을 다른 샘플에 연결합니다.
MFCompareFullToPartialMediaType 전체 미디어 형식과 부분 미디어 형식을 비교합니다.
MFConvertColorInfoFromDXVA MFVIDEOFORMAT 구조체의 확장 색 정보를 설정합니다.
MFConvertColorInfoToDXVA 확장 색 정보를 MFVIDEOFORMAT에서 해당하는 DXVA(DirectX Video Acceleration) 색 정보로 변환합니다.
MFConvertFromFP16Array 16비트 부동 소수점 숫자의 배열을 32비트 부동 소수점 숫자의 배열로 변환합니다.
MFConvertToFP16Array 32비트 부동 소수점 숫자의 배열을 16비트 부동 소수점 숫자의 배열로 변환합니다.

MFCopyImage 한 버퍼에서 다른 버퍼로 이미지 또는 이미지 평면을 복사합니다.
MFCreate2DMediaBuffer 2D 이미지 데이터를 저장할 시스템 메모리 버퍼 개체를 만듭니다.
MFCreateAlignedMemoryBuffer 지정된 바이트 맞춤을 사용하여 시스템 메모리를 할당하고 메모리를 관리하는 미디어 버퍼를 만듭니다.
MFCreateAMMediaTypeFromMFMediaType Media Foundation 미디어 형식에서 DirectShow AM_MEDIA_TYPE 구조를 만듭니다.
MFCreateAsyncResult 비동기 결과 개체를 만듭니다. 비동기 메서드를 구현하는 경우 이 함수를 사용합니다. (MFCreateAsyncResult)
MFCreateAttributes 빈 특성 저장소를 만듭니다.
MFCreateAudioMediaType WAVEFORMATEX 구조체에서 오디오 미디어 형식을 만듭니다.
MFCreateCollection 빈 컬렉션 개체를 만듭니다.
MFCreateDXGIDeviceManager Microsoft DXGI(DirectX Graphics Infrastructure) 장치 관리자 인스턴스를 만듭니다.
MFCreateDXGISurfaceBuffer Microsoft DXGI(DirectX Graphics Infrastructure) 화면을 관리하는 미디어 버퍼를 만듭니다.
MFCreateDXSurfaceBuffer Direct3D 9 화면을 관리하는 미디어 버퍼 개체를 만듭니다.
MFCreateEventQueue 이벤트 큐를 만듭니다.

MFCreateFile 파일에서 바이트 스트림을 만듭니다.
MFCreateLegacyMediaBufferOnMFMediaBuffer Media Foundation 미디어 버퍼를 DMO(DirectX Media Objects)와 호환되는 버퍼로 변환합니다.
MFCreateMediaBufferFromMediaType 지정된 미디어 형식에 가장 적합한 시스템 메모리 버퍼를 할당합니다.
MFCreateMediaBufferWrapper 기존 미디어 버퍼를 래핑하는 미디어 버퍼를 만듭니다.
MFCreateMediaEvent 미디어 이벤트 개체를 만듭니다.
MFCreateMediaExtensionActivate Windows 런타임 클래스에 대한 활성화 개체를 만듭니다.
MFCreateMediaType 빈 미디어 형식을 만듭니다.
MFCreateMediaTypeFromRepresentation 다른 형식 표현에서 Media Foundation 미디어 형식을 만듭니다.
MFCreateMemoryBuffer 시스템 메모리를 할당하고 이를 관리하는 미디어 버퍼를 만듭니다.
MFCreateMFByteStreamWrapper 바이트 스트림에 대한 래퍼를 만듭니다.
MFCreateMFVideoFormatFromMFMediaType 비디오 미디어 형식에서 MFVIDEOFORMAT 구조를 만듭니다.
MFCreateMuxStreamAttributes 멀티플렉싱된 하위 스트림의 콘텐츠를 설명하는 IMFAttributes를 만듭니다.
MFCreateMuxStreamMediaType

멀티플렉싱된 하위 스트림의 미디어 형식을 설명하는 IMFMediaType을 만듭니다.
MFCreateMuxStreamSample 멀티플렉싱된 하위 스트림의 샘플을 포함하는 IMFSample을 만듭니다.
MFCreateSample 빈 미디어 샘플을 만듭니다.
MFCreateTempFile 임시 로컬 파일에서 지원되는 바이트 스트림을 만듭니다.
MFCreateVideoMediaType MFVIDEOFORMAT 구조체에서 비디오 미디어 형식을 만듭니다.
MFCreateVideoMediaTypeFromBitMapInfoHeader 이 함수는 구현되지 않습니다. (MFCreateVideoMediaTypeFromBitMapInfoHeader)
MFCreateVideoMediaTypeFromBitMapInfoHeaderEx BITMAPINFOHEADER 구조체에서 비디오 미디어 형식을 만듭니다.
MFCreateVideoMediaTypeFromSubtype 지정된 하위 형식을 사용하여 부분 비디오 미디어 형식을 만듭니다.
MFCreateVideoMediaTypeFromVideoInfoHeader KS_VIDEOINFOHEADER 구조체에서 미디어 형식을 만듭니다.
MFCreateVideoMediaTypeFromVideoInfoHeader2 KS_VIDEOINFOHEADER2 구조체에서 미디어 형식을 만듭니다.
MFCreateVideoSampleAllocatorEx Microsoft DXGI(DirectX Graphics Infrastructure)와 호환되는 비디오 샘플을 할당하는 개체를 만듭니다.
MFCreateWaveFormatExFromMFMediaType Media Foundation 오디오 미디어 형식을 WAVEFORMATEX 구조체로 변환합니다.
MFCreateWICBitmapBuffer WIC(Windows 이미징 구성 요소)를 관리하는 미디어 버퍼 개체를 만듭니다.

MFEndCreateFile 파일에서 바이트 스트림을 만드는 비동기 요청을 완료합니다.
MFEndRegisterWorkQueueWithMMCSS 작업 큐를 MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업과 연결하는 비동기 요청을 완료합니다. (MFEndRegisterWorkQueueWithMMCSS)
MFEndUnregisterWorkQueueWithMMCSS MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업에서 작업 큐의 등록을 취소하는 비동기 요청을 완료합니다.
MFFrameRateToAverageTimePerFrame 비디오 프레임 속도를 프레임 기간으로 변환합니다.
MFGetAttribute2UINT32asUINT64 값이 UINT64로 압축된 두 UINT32 값인 특성을 가져옵니다.
MFGetAttributeDouble 특성 저장소에서 double 값을 반환하거나 특성이 없는 경우 기본값을 반환합니다.
MFGetAttributeRatio 값이 비율인 특성을 검색합니다.
MFGetAttributesAsBlob 특성 저장소의 내용을 바이트 배열로 변환합니다.
MFGetAttributesAsBlobSize MFGetAttributesAsBlob 함수에 필요한 버퍼의 크기를 검색합니다.
MFGetAttributeSize 값이 너비 및 높이로 표현된 크기인 특성을 검색합니다.
MFGetAttributeString 특성 저장소에서 문자열 값을 가져옵니다.
MFGetAttributeUINT32 특성 저장소에서 UINT32 값을 반환하거나 특성이 없는 경우 기본값을 반환합니다.

MFGetAttributeUINT64 특성 저장소에서 UINT64 값을 반환하거나 특성이 없는 경우 기본값을 반환합니다.
MFGetContentProtectionSystemCLSID 콘텐츠 보호 시스템의 클래스 식별자를 가져옵니다.
MFGetMFTMerit 하드웨어 코덱의 장점 값을 가져옵니다.
MFGetPlaneSize 압축되지 않은 비디오 형식의 이미지 크기(바이트)를 검색합니다. (MFGetPlaneSize)
MFGetPluginControl Microsoft Media Foundation 플러그 인 관리자에 대한 포인터를 가져옵니다.
MFGetStrideForBitmapInfoHeader 비디오 형식의 최소 표면 보폭을 계산합니다.
MFGetTimerPeriodicity MFAddPeriodicCallback 함수에 대한 타이머 간격을 검색합니다.
MFGetUncompressedVideoFormat 압축되지 않은 비디오 형식의 FOURCC 또는 D3DFORMAT 값을 반환합니다.
MFGetWorkQueueMMCSSClass 현재 이 작업 큐와 연결된 MMCSS(멀티미디어 클래스 Scheduler Service) 클래스를 검색합니다. (MFGetWorkQueueMMCSSClass)
MFGetWorkQueueMMCSSPriority 작업 큐의 상대 스레드 우선 순위를 가져옵니다. (MFGetWorkQueueMMCSSPriority)
MFGetWorkQueueMMCSSTaskId 현재 이 작업 큐와 연결된 MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업 식별자를 검색합니다. (MFGetWorkQueueMMCSSTaskId)
MFHeapAlloc 메모리 블록을 할당합니다. (MFHeapAlloc)

MFHeapFree MFHeapAlloc 함수를 호출하여 할당된 메모리 블록을 해제합니다.
MFInitAMMediaTypeFromMFMediaType Media Foundation 미디어 형식에서 DirectShow AM_MEDIA_TYPE 구조를 초기화합니다.
MFInitAttributesFromBlob 바이트 배열에서 특성 저장소의 콘텐츠를 초기화합니다.
MFInitMediaTypeFromAMMediaType DirectShow AM_MEDIA_TYPE 구조체에서 미디어 형식을 초기화합니다.
MFInitMediaTypeFromMFVideoFormat MFVIDEOFORMAT 구조체에서 미디어 형식을 초기화합니다.
MFInitMediaTypeFromMPEG1VideoInfo DirectShow MPEG1VIDEOINFO 구조체에서 미디어 형식을 초기화합니다.
MFInitMediaTypeFromMPEG2VideoInfo DirectShow MPEG2VIDEOINFO 구조체에서 미디어 형식을 초기화합니다.
MFInitMediaTypeFromVideoInfoHeader DirectShow VIDEOINFOHEADER 구조체에서 미디어 형식을 초기화합니다.
MFInitMediaTypeFromVideoInfoHeader2 DirectShow VIDEOINFOHEADER2 구조체에서 미디어 형식을 초기화합니다.
MFInitMediaTypeFromWaveFormatEx WAVEFORMATEX 구조체에서 미디어 형식을 초기화합니다.
MFInitVideoFormat DVD, 아날로그 텔레비전 또는 ATSC 디지털 텔레비전과 같은 표준 비디오 형식의 MFVIDEOFORMAT 구조를 초기화합니다.
MFInitVideoFormat_RGB 압축되지 않은 RGB 비디오 형식에 대한 MFVIDEOFORMAT 구조를 초기화합니다.

MFInvokeCallback 콜백 메서드를 호출하여 비동기 작업을 완료합니다. (MFInvokeCallback)
MFIsFormatYUV FOURCC 코드 또는 D3DFORMAT 값이 YUV 형식인지 여부를 쿼리합니다.
MFllMulDiv 각 용어가 64비트 부호 있는 값인 (a * b) + d) / c를 계산합니다.
MFLockDXGIDeviceManager 공유 Microsoft DXGI(DirectX Graphics Infrastructure) 디바이스 관리자를 잠깁니다.
MFLockPlatform MFShutdown 함수를 차단합니다.
MFLockSharedWorkQueue 공유 작업 큐를 가져오고 잠깁니다. (MFLockSharedWorkQueue)
MFLockWorkQueue 작업 큐를 잠깁니다. (MFLockWorkQueue)
MFMapDX9FormatToDXGIFormat Microsoft Direct3D 9 형식 식별자를 Microsoft DXGI(DirectX Graphics Infrastructure) 형식 식별자로 변환합니다.
MFMapDXGIFormatToDX9Format Microsoft DXGI(DirectX Graphics Infrastructure) 형식 식별자를 Microsoft Direct3D 9 형식 식별자로 변환합니다.
MFPutWaitingWorkItem 이벤트가 신호를 받을 때까지 기다리는 작업 항목을 큐에 대기합니다. (MFPutWaitingWorkItem)
MFPutWorkItem 작업 큐에 비동기 작업을 배치합니다. (MFPutWorkItem)
MFPutWorkItem2 지정된 우선 순위를 사용하여 작업 큐에 비동기 작업을 배치합니다. (MFPutWorkItem2)

MFPutWorkItemEx 작업 큐에 비동기 작업을 배치합니다. (MFPutWorkItemEx)
MFPutWorkItemEx2 지정된 우선 순위를 사용하여 작업 큐에 비동기 작업을 배치합니다. (MFPutWorkItemEx2)
MFRegisterLocalByteStreamHandler 호출자의 프로세스에 바이트 스트림 처리기를 등록합니다.
MFRegisterLocalSchemeHandler 호출자의 프로세스에 스키마 처리기를 등록합니다.
MFRegisterPlatformWithMMCSS 표준 Microsoft Media Foundation 플랫폼 작업 큐를 MMCSS(멀티미디어 클래스 스케줄러 서비스)에 등록합니다.
MFRemovePeriodicCallback MFAddPeriodicCallback 함수에 의해 설정된 콜백 함수를 취소합니다.
MFScheduleWorkItem 지정된 간격 후에 완료될 비동기 작업을 예약합니다. (MFScheduleWorkItem)
MFScheduleWorkItemEx 지정된 간격 후에 완료될 비동기 작업을 예약합니다. (MFScheduleWorkItemEx)
MFSetAttribute2UINT32asUINT64 두 UINT32 값을 UINT64 특성 값으로 압축합니다.
MFSetAttributeRatio 비율을 64비트 특성 값으로 설정합니다.
MFSetAttributeSize 너비와 높이를 단일 64비트 특성 값으로 설정합니다.
MFShutdown Microsoft Media Foundation 플랫폼을 종료합니다.

MFSplitSample 결합된 미디어 샘플을 개별 샘플로 다시 분할합니다.
MFStartup Microsoft Media Foundation을 초기화합니다.
MFTEnum 레지스트리에서 MFT(Media Foundation 변환)를 열거합니다.
MFTEnum2 지정된 검색 조건과 일치하는 MFT(Microsoft Media Foundation 변환) 목록을 가져옵니다.
MFTEnumEx 지정된 검색 조건과 일치하는 MFT(Microsoft Media Foundation 변환) 목록을 가져옵니다. 이 함수는 MFTEnum 함수를 확장합니다.
MFTGetInfo 레지스트리에서 MFT(Media Foundation 변환)에 대한 정보를 가져옵니다.
MFTRegister MFT(Media Foundation 변환)에 대한 정보를 레지스트리에 추가합니다.
MFTRegisterLocal 호출자 프로세스에 MFT(Media Foundation 변환)를 등록합니다. (MFTRegisterLocal)
MFTRegisterLocalByCLSID 호출자 프로세스에 MFT(Media Foundation 변환)를 등록합니다. (MFTRegisterLocalByCLSID)
MFTUnregister MFT(Media Foundation 변환)를 등록 취소합니다.
MFTUnregisterLocal 호출자의 프로세스에서 하나 이상의 MFT(Media Foundation 변환)를 등록 취소합니다.
MFTUnregisterLocalByCLSID 호출자의 프로세스에서 MFT(Media Foundation 변환)를 등록 취소합니다.

MFUnlockDXGIDeviceManager 공유 Microsoft DXGI(DirectX Graphics Infrastructure) 디바이스 관리자의 잠금을 해제합니다.
MFUnlockPlatform MFLockPlatform 함수를 호출하여 잠긴 후 Media Foundation 플랫폼의 잠금을 해제합니다.
MFUnlockWorkQueue 작업 큐의 잠금을 해제합니다. (MFUnlockWorkQueue)
MFUnregisterPlatformFromMMCSS MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업에서 Microsoft Media Foundation 플랫폼 작업 큐의 등록을 취소합니다.
MFUnwrapMediaType MFWrapMediaType 함수에 의해 다른 미디어 형식으로 래핑된 미디어 형식을 검색합니다.
MFValidateMediaTypeSize 비디오 형식 블록에 대한 버퍼의 크기를 확인합니다.
MFWrapMediaType 다른 미디어 형식을 래핑하는 미디어 형식을 만듭니다.
Pack2UINT32AsUINT64 두 UINT32 값을 UINT64 값으로 압축합니다.
PackRatio 비율을 나타내는 두 개의 UINT32 값을 UINT64 값으로 압축합니다.
PackSize UINT32 너비 값과 UINT32 높이 값을 크기를 나타내는 UINT64 값으로 압축합니다.
Unpack2UINT32AsUINT64 UINT64 값에서 하위 순서 및 상위 UINT32 값을 가져옵니다.
UnpackRatio 비율을 나타내는 UINT64 값에서 낮은 순서 및 상위 UINT32 값을 가져옵니다.

unpackSize 크기를 나타내는 UINT64 값에서 하위 순서 및 상위 UINT32 값을 가져옵니다.

콜백 함수

테이블 확장

MFPERIODICCALLBACK MFAddPeriodicCallback 함수에 대한 콜백 함수입니다.

구조

테이블 확장

CapturedMetadataExposureCompensation 이 구조에는 캡처된 사진에 대한 EV 보상 피드백에 대한 Blob 정보가 포함됩니다.
CapturedMetadataISOGains CapturedMetadataISOGains 구조는 MF_CAPTURE_METADATA_ISO_GAINS 대한 Blob 형식을 설명합니다.
CapturedMetadataWhiteBalanceGains 이 구조체는 MF_CAPTURE_METADATA_WHITEBALANCE_GAINS 특성에 대한 Blob 형식을 설명합니다.
DigitalWindowSetting 비디오 캡처를 위한 디지털 창의 경계 설정을 나타냅니다.
FaceCharacterization FaceCharacterization 구조는 MF_CAPTURE_METADATA_FACEROICHARACTERIZATIONS 특성에 대한 Blob 형식을 설명합니다.
FaceCharacterizationBlobHeader

FaceCharacterizationBlobHeader 구조체는 MF_CAPTURE_METADATA_FACEROICHARACTERIZATIONS 특성에 대한 Blob 형식의 크기 및 개수 정보를 설명합니다.
FaceRectInfo FaceRectInfo 구조체는 MF_CAPTURE_METADATA_FACEROIS 특성에 대한 Blob 형식을 설명합니다.
FaceRectInfoBlobHeader FaceRectInfoBlobHeader 구조는 MF_CAPTURE_METADATA_FACEROIS 특성에 대한 Blob 형식의 크기 및 개수 정보를 설명합니다.
HistogramBlobHeader HistogramBlobHeader 구조는 blob 크기 및 MF_CAPTURE_METADATA_HISTOGRAM 특성에 대한 Blob의 히스토그램 수를 설명합니다.
HistogramDataHeader HistogramDataHeader 구조체는 MF_CAPTURE_METADATA_HISTOGRAM 특성에 대한 Blob 형식을 설명합니다.
HistogramGrid HistogramGrid 구조체는 MF_CAPTURE_METADATA_HISTOGRAM 대한 Blob 형식을 설명합니다.
히스토그램 헤더 HistogramHeader 구조체는 MF_CAPTURE_METADATA_HISTOGRAM 대한 Blob 형식을 설명합니다.
MACROBLOCK_DATA 비디오 디코딩 중에 macroblock에 대한 데이터를 제공합니다.
MetadataTimeStamps MetadataTimeStamps 구조체는 MF_CAPTURE_METADATA_FACEROITIMESTAMPS 특성에 대한 Blob 형식을 설명합니다.
MF_FLOAT2 두 구성 요소가 있는 벡터입니다.
MF_FLOAT3 세 가지 구성 요소가 있는 벡터입니다.
MF_QUATERNION

회전을 나타내는 데 사용되는 4차원 벡터입니다.
MFASYNCRESULT IMFAsyncResult 인터페이스를 구현하는 데 필요한 데이터를 포함합니다.
MFCameraExtrinsic_CalibratedTransform 다른 카메라 또는 설정된 외부 참조를 기준으로 카메라의 위치를 설명하는 변환입니다.
MFCameraExtrinsics 다른 카메라 또는 설정된 외부 참조를 기준으로 카메라의 위치를 설명합니다.
MFCameraIntrinsic_DistortionModel 다항식 렌즈 왜곡 모델을 나타냅니다.
MFCameraIntrinsic_PinholeCameraModel 핀홀 카메라 모델을 나타냅니다.
MFFOLDDOWN_MATRIX 다중 채널 오디오를 더 적은 수의 오디오 채널로 변환하는 데 사용되는 계수를 포함합니다. 이 프로세스를 접는 프로세스라고 합니다.
MFPinholeCameraIntrinsic_IntrinsicModel 지정된 해상도에 대한 핀홀 카메라 내장 모델을 나타냅니다.
MFPinholeCameraIntrinsics 실제 세계의 3D 지점을 카메라의 2D 이미지 프레임에 투사하는 방법을 설명하는 0개 또는 1개의 핀홀 카메라 내장 모델이 포함되어 있습니다.
MFSampleExtensionPsnrYuv 인코딩된 비디오 프레임의 Y, U 및 V 평면에 대한 PSNR(최고 신호 대 노이즈 비율) 값을 저장합니다.
MT_ARBITRARY_HEADER ASF(고급 스트리밍 형식) 파일의 이진 스트림에 대한 형식 데이터를 포함합니다.
MT_CUSTOM_VIDEO_PRIMARIES 비디오 원본에 대한 사용자 지정 색 주체를 정의합니다. 색 주체는 RGB 색 공간에서 CIE XYZ 색 공간으로 색을 변환하는 방법을 정의합니다.

ROI_AREA 관심 있는 지역을 정의합니다.

열거 목록

테이블 확장

_MFT_ENUM_FLAG MFT(Media Foundation 변환)를 등록하고 열거하기 위한 플래그를 포함합니다.
MF_TOPOSTATUS 재생 중에 토폴로지의 상태를 지정합니다.
MFASYNC_WORKQUEUE_TYPE 만들 MFAllocateWorkQueueEx 함수의 작업 큐 유형을 지정합니다.
MFDepthMeasurement 비디오 프레임의 깊이 값에 대한 측정 시스템을 지정합니다.
MFFrameSourceTypes 프레임 원본에서 제공하는 데이터 형식을 설명합니다.
MFSampleEncryptionProtectionScheme 암호화된 샘플에 대해 지원되는 보호 체계를 지정합니다.
MFVideo3DFormat 3D 비디오 프레임이 메모리에 저장되는 방법을 지정합니다.
MFVideo3DSampleFormat 3D 비디오 프레임이 미디어 샘플에 저장되는 방법을 지정합니다.
MFVideoDRMFlags 비디오 스트림에 필요한 복사 보호 유형을 지정합니다.

MFVideoPadFlags 비디오 이미지가 지정된 가로 세로 비율에 맞게 채워질지 여부를 지정합니다.
MFVideoRotationFormat 시계 반대 방향으로 비디오 이미지의 회전을 설명합니다.
MFVideoSrcContentHintFlags 비디오 스트림의 의도된 가로 세로 비율을 설명합니다.
MFWaveFormatExConvertFlags 오디오 미디어 형식을 변환하는 방법을 지정하는 플래그를 포함합니다.

Last updated on 2025. 06. 18.

mfcaptureengine.h 헤더

이 헤더는 Microsoft Media Foundation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft Media Foundation

mfcaptureengine.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Interfaces

테이블 확장

IMFCaptureEngine 하나 이상의 캡처 디바이스를 제어합니다.
IMFCaptureEngineClassFactory 캡처 엔진의 인스턴스를 만듭니다. (IMFCaptureEngineClassFactory)
IMFCaptureEngineOnEventCallback 캡처 엔진에서 이벤트를 수신하기 위한 콜백 인터페이스입니다.
IMFCaptureEngineOnSampleCallback 캡처 엔진에서 데이터를 수신하는 콜백 인터페이스입니다.
IMFCaptureEngineOnSampleCallback2 캡처 엔진에서 데이터를 수신하는 데 사용되는 IMFCaptureEngineOnSampleCallback 콜백 인터페이스에 대한 확장입니다.
IMFCapturePhotoSink 사진 싱크를 제어합니다.
IMFCapturePreviewSink 미리 보기 싱크를 제어합니다.
IMFCaptureRecordSink 기록 싱크를 제어합니다.
IMFCaptureSink

캡처 디바이스에서 하나 이상의 스트림을 수신하는 개체인 캡처 싱크를 제어합니다.
IMFCaptureSink2 IMFCaptureSink 인터페이스를 확장하여 레코드 싱크 또는 미리 보기 싱크의 출력 미디어 형식을 동적으로 설정하는 기능을 제공합니다.
IMFCaptureSource 캡처 원본 개체를 제어합니다. 캡처 원본은 오디오 및 비디오 캡처 디바이스를 관리합니다.

구조

테이블 확장

MFVideoNormalizedRect MFVideoNormalizedRect(mfcaptureengine.h) 구조체는 비디오 사각형에서 하위 사각형을 지정하는 데 사용되는 정규화된 사각형을 정의합니다.

열거 목록

테이블 확장

MF_CAPTURE_ENGINE_DEVICE_TYPE 캡처 디바이스의 유형을 지정합니다.
MF_CAPTURE_ENGINE_SINK_TYPE 캡처 싱크의 형식을 지정합니다.
MF_CAPTURE_ENGINE_STREAM_CATEGORY 원본 스트림 범주의 값을 정의합니다.

Last updated on 2026. 01. 29.

mfcontentdecryptionmodule.h 헤더

아티클2023. 08. 24.

이 헤더는 Microsoft Media Foundation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft Media Foundation

mfcontentdecryptionmodule.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IMFContentDecryptionModule DRM 키 시스템의 CDM(콘텐츠 암호 해독 모듈)을 나타냅니다.
IMFContentDecryptionModuleAccess 미디어 키 시스템에 대한 액세스를 제공합니다.
IMFContentDecryptionModuleFactory IMFContentDecryptionModuleAccess 개체를 만들기 위한 팩터리 인터페이스입니다.
IMFContentDecryptionModuleSession 암호화된 미디어 확장 지원을 위해 CDM(콘텐츠 암호 해독 모듈)에 대한 액세스를 제공합니다.
IMFContentDecryptionModuleSessionCallbacks IMFContentDecryptionModuleSession에서 키 상태 변경 업데이트를 수신하기 위한 콜백 메커니즘을 제공합니다.

Functions

MFCreateEncryptedMediaExtensionsStoreActivate MFCreateEncryptedMediaExtensionsStoreActivate 함수는 IMFActivate를 구현하는 개체를 만듭니다. 이 개체의 ActivateObject 구현은 지정된 IMFPMPHostApp 및 클래스 ID를 기반으로 합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mfd3d12.h 헤더

아티클2023. 08. 24.

Media Foundation 생산자와 소비자 간에 Direct3D 리소스에 대한 액세스를 동기화하기 위한 API를 제공합니다.

이 헤더는 Microsoft Media Foundation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft Media Foundation

mfd3d12.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IMFD3D12SynchronizationObject D3D12 리소스 할당자가 D3D12 리소스의 수명을 관리할 수 있도록 허용합니다.
IMFD3D12SynchronizationObjectCommands 리소스를 사용할 준비가 되거나 해제될 수 있는 경우 D3D12 리소스의 생산자 또는 소비자가 다운스트림 구성 요소에 신호를 보낼 수 있도록 하는 명령을 제공합니다.

Functions

MFCreateD3D12SynchronizationObject Media Foundation 개체에 저장된 D3D12 리소스에 대한 액세스를 동기화하는 데 사용되는 Media Foundation D3D12 동기화 기본 형식을 인스턴스화합니다.

열거형

MF_MT_D3D_RESOURCE_VERSION_ENUM 미디어 형식과 연결된 스트림에 사용되는 리소스의 Direct3D 버전을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mfidl.h 헤더

아티클2023. 03. 09.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 미디어 스트리밍 API
- Microsoft Media Foundation
- 스트리밍 미디어 DDI 참조

mfidl.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IMFAudioPolicy 스트리밍 오디오 렌더러(SAR)와 연결된 오디오 세션을 구성합니다.
IMFAudioStreamVolume 개별 오디오 채널의 볼륨 수준을 제어합니다.
IMFByteStreamBuffering 바이트 스트림이 네트워크에서 데이터를 버퍼링하는 방법을 제어합니다.
IMFByteStreamCacheControl 네트워크 바이트 스트림이 로컬 캐시로 데이터를 전송하는 방법을 제어합니다. (IMFByteStreamCacheControl)
IMFByteStreamCacheControl2 네트워크 바이트 스트림이 로컬 캐시로 데이터를 전송하는 방법을 제어합니다. (IMFByteStreamCacheControl2)
IMFByteStreamHandler 바이트 스트림에서 미디어 원본을 만듭니다.
IMFByteStreamTimeSeek 바이트 스트림 기준 시간 위치를 찾습니다.

IMFCameraConfigurationManager IMFCameraConfigurationManager 인터페이스는 COM 함수 CoCreateInstance를 호출하고 CLSID_CAMERACONFIGURATIONMANAGER CLSID 매개 변수로 전달하여 만들 수 있습니다.
IMFCameraControlDefaults 카메라 컨트롤 기본값을 지정하는 메서드를 제공합니다.
IMFCameraControlDefaultsCollection 카메라 컨트롤의 컬렉션을 나타냅니다.
IMFCameraControlMonitor 카메라 컨트롤의 상태가 변경되면 알림을 구독 및 구독 취소하는 데 사용되는 카메라 컨트롤 모니터를 나타냅니다.
IMFCameraControlNotify 카메라 컨트롤의 변경 내용에 대한 알림 콜백을 나타냅니다.
IMFCameraOcclusionStateMonitor 카메라 디바이스의 폐색 상태를 모니터링합니다.
IMFCameraOcclusionStateReport IMFCameraOcclusionStateMonitor에서 감지한 상태 변경과 관련된 카메라 폐색 상태를 제공합니다.
IMFCameraOcclusionStateReportCallback 카메라 폐색 상태 변경 알림을 수신하기 위해 애플리케이션에 의해 구현됩니다.
IMFClock Microsoft Media Foundation의 시계에서 타이밍 정보를 제공합니다.
IMFClockConsumer IMFPresentationClock에 액세스하기 위해 앱에서 구현합니다.
IMFClockStateSink 프레젠테이션 시계에서 상태 변경 알림을 받습니다.

IMFContentDecryptorContext 암호 해독기가 하드웨어 키를 관리하고 하드웨어 샘플의 암호를 해독할 수 있습니다.
IMFContentEnabler 사용자가 미디어 콘텐츠에 액세스하기 위해 수행해야 하는 한 단계를 구현합니다.
IMFContentProtectionDevice 암호 해독기가 보호 시스템에 대한 하드웨어 암호 해독을 구현하는 보안 프로세서와 통신할 수 있습니다.
IMFContentProtectionManager 애플리케이션에 콘텐츠 사용 프로그램 개체에 대한 포인터를 제공하여 보호된 콘텐츠를 재생할 수 있도록 합니다.
IMFDXGIDeviceManagerSource Microsoft Media Foundation 비디오 렌더링 싱크에서 IMFDXGIDeviceManager를 가져오는 기능을 제공합니다.
IMFExtendedCameraControl 이 인터페이스는 캡처 디바이스의 확장 속성을 구성하는 데 사용됩니다.
IMFExtendedCameraController IMFExtendedCameraController 인터페이스를 사용하면 앱이 캡처 디바이스의 확장 속성을 구성하는 데 사용되는 IMFExtendedCameraControl 인스턴스를 검색할 수 있습니다.
IMFFieldOfUseMFTUnlock 애플리케이션이 사용에 제한이 있는 MFT(Media Foundation 변환)를 사용할 수 있도록 합니다.
IMFFinalizableMediaSink 필요에 따라 종료하기 전에 필요한 작업을 수행하기 위해 미디어 싱크에서 지원됩니다.
IMFGetService 지정된 서비스 인터페이스에 대한 개체를 쿼리합니다. (IMFGetService)
IMFHttpDownloadRequest 애플리케이션은 이 인터페이스를 구현하여 Microsoft Media Foundation에서 사용하는 HTTP 및 HTTPS 프로토콜의 기본 구현을 재정의합니다. (IMFHttpDownloadRequest)

IMFHttpDownloadSession 애플리케이션은 이 인터페이스를 구현하여 Microsoft Media Foundation에서 사용하는 HTTP 및 HTTPS 프로토콜의 기본 구현을 재정의합니다. (IMFHttpDownloadSession)
IMFHttpDownloadSessionProvider 애플리케이션은 사용자 지정 HTTP 또는 HTTPS 다운로드 구현을 사용자 지정하기 위해 이 인터페이스를 구현합니다.
IMFInputTrustAuthority PMP(보호된 미디어 경로)의 다른 구성 요소가 ITA(입력 신뢰 기관)에서 제공하는 입력 보호 시스템을 사용할 수 있도록 합니다.
IMFLocalMFTRegistration 호출자 프로세스에서 MFT(Media Foundation 변환)를 등록합니다.
IMFMediaSession 보호된 콘텐츠와 보호되지 않는 콘텐츠에 대한 재생 컨트롤을 제공합니다.
IMFMediaSink 미디어 싱크 개체에 의해 구현됩니다.
IMFMediaSinkPreroll 프레젠테이션 시계가 시작되기 전에 미디어 싱크에서 샘플을 받을 수 있도록 합니다.
IMFMediaSource 미디어 원본 개체에 의해 구현됩니다.
IMFMediaSourceEx IMFMediaSource 인터페이스를 확장하여 미디어 원본에 대한 추가 기능을 제공합니다.
IMFMediaSourcePresentationProvider sequencer 원본에 알림을 제공합니다.
IMFMediaSourceTopologyProvider 애플리케이션이 sequencer 원본에서 토폴로지를 가져올 수 있도록 합니다.

IMFMediaStream 미디어 원본에서 하나의 스트림을 나타냅니다.
IMFMediaStreamSourceSampleRequest MediaStreamSource의 샘플에 대한 요청을 나타냅니다.
IMFMediaTypeHandler 미디어 원본 또는 미디어 싱크와 같은 개체의 미디어 형식을 가져오고 설정합니다.
IMFMetadata 개체에 대한 메타데이터를 관리합니다.
IMFMetadataProvider 미디어 원본 또는 기타 개체에서 메타데이터를 가져옵니다.
IMFNetCredential 인증을 위해 사용자 이름 및 암호 정보를 설정하고 검색합니다.
IMFNetCredentialCache 자격 증명 캐시에서 자격 증명을 가져옵니다.
IMFNetCredentialManager 네트워크 원본에 대한 사용자 자격 증명을 제공하기 위해 애플리케이션에서 구현합니다.
IMFNetCrossOriginSupport HTML5 미디어 다운로드에 대한 원본 간 정책을 적용하려는 클라이언트에서 구현합니다.
IMFNetProxyLocator 서버에 연결할 때 사용할 프록시를 결정합니다.
IMFNetProxyLocatorFactory 사용할 프록시를 결정하는 프록시 로케이터 개체를 만듭니다.
IMFNetResourceFilter 바이트 스트림이 URL을 요청할 때 애플리케이션에 알리고 애플리케이션이 URL 리디렉션을 차단할 수 있도록 합니다.

IMFNetSchemeHandlerConfig 네트워크 구성표 플러그 인을 구성합니다.
IMFObjectReferenceStream 스트림에 대한 인터페이스 포인터를 마샬링합니다. IStream을 지원하는 스트림 개체는 인터페이스 포인터에 대한 사용자 지정 마샬링을 제공하기 위해 이 인터페이스를 노출할 수 있습니다.
IMFOutputPolicy ITA(입력 신뢰 기관)에서 사용 정책을 캡슐화합니다.
IMFOutputSchema 출력 보호 시스템 및 해당 구성 데이터에 대한 정보를 캡슐화합니다.
IMFOutputTrustAuthority 신뢰할 수 있는 출력이 지원하는 하나 이상의 출력 보호 시스템의 기능을 캡슐화합니다.
IMFPMPClient 미디어 원본이 IMFPMPHost 인터페이스에 대한 포인터를 받을 수 있도록 합니다.
IMFPMPClientApp 미디어 원본이 Windows 스토어 앱에서 콘텐츠 보호 기능을 구현하는 메커니즘을 제공합니다.
IMFPMPHost 애플리케이션 프로세스의 미디어 원본이 PMP(보호된 미디어 경로) 프로세스에서 개체를 만들 수 있도록 합니다.
IMFPMPHostApp 미디어 원본이 PMP(보호된 미디어 경로) 프로세스에서 Windows 런타임 개체를 만들 수 있습니다.
IMFPMPServer 미디어 세션의 두 인스턴스가 동일한 PMP(보호된 미디어 경로) 프로세스를 공유할 수 있도록 합니다.
IMFPresentationClock 샘플이 렌더링될 때 예약하고 여러 스트림을 동기화하는 데 사용되는 프레젠테이션 시계를 나타냅니다.

IMFPresentationDescriptor 프레젠테이션의 세부 정보를 설명합니다. 프레젠테이션은 일반적인 프레젠테이션 시간을 공유하는 일련의 관련 미디어 스트림입니다.
IMFPresentationTimeSource 프레젠테이션 시계의 시계 시간을 제공합니다.
IMFProtectedEnvironmentAccess 콘텐츠 보호 시스템이 보호된 환경과 핸드셰이크를 수행할 수 있도록 하는 메서드를 제공합니다. CreateFile 및 DeviceIoControl API는 Windows 스토어 앱에서 사용할 수 없기 때문에 필요합니다.
IMFQualityAdvise 품질 관리자가 파이프라인에서 구성 요소의 오디오 또는 비디오 품질을 조정할 수 있도록 합니다.
IMFQualityAdvise2 파이프라인 개체가 품질 메시지에 대한 응답으로 자체 오디오 또는 비디오 품질을 조정할 수 있도록 합니다.
IMFQualityAdviseLimits 개체에서 지원하는 품질 모드 수를 쿼리합니다.
IMFQualityManager 재생 품질을 조정합니다. 이 인터페이스는 품질 관리자에 의해 노출됩니다.
IMFRateControl 재생 속도를 가져오거나 설정합니다.
IMFRateSupport 역방향 재생을 포함하여 지원되는 재생 속도 범위를 쿼리합니다.
IMFRealTimeClient 파이프라인 개체에 MMCSS(멀티미디어 클래스 스케줄러 서비스)에 등록하도록 알 수 있습니다. (IMFRealTimeClient)
IMFRealTimeClientEx 파이프라인 개체에 MMCSS(멀티미디어 클래스 스케줄러 서비스)에 등록하도록 알 수 있습니다. (IMFRealTimeClientEx)

IMFRelativePanelReport 이 인터페이스는 표시 영역에 대한 현재 상대 패널 위치를 검색하는 데 사용됩니다.
IMFRelativePanelWatcher 디스플레이와 연결된 패널을 모니터링하여 패널의 상대 위치가 변경될 때 앱이 알림을 받도록 합니다.
IMFRemoteDesktopPlugin 터미널 서비스 환경에서 사용할 토폴로지 수정 (IMFRemoteDesktopPlugin)
IMFRemoteProxy 원격 개체의 프록시 역할을 하는 개체에 의해 노출됩니다.
IMFSAMIStyle SAMI 미디어 원본에서 SAMI(Synchronized Accessible Media Interchange) 스타일을 설정하고 검색합니다.
IMFSampleAllocatorControl 비디오 캡처 원본 및 변환에 의해 구현됩니다. 시스템에서 컨테이너 내에서 액세스할 수 있는 메모리를 사용하여 샘플을 할당하는 샘플 할당자를 구성 요소에 제공할 수 있습니다.
IMFSampleGrabberSinkCallback 샘플 그래버 싱크에서 미디어 데이터를 가져오는 콜백 인터페이스입니다.
IMFSampleGrabberSinkCallback2 IMFSampleGrabberSinkCallback 인터페이스를 확장합니다.
IMFSampleProtection PMP(보호된 미디어 경로) 내의 미디어 데이터에 대한 암호화를 제공합니다.
IMFSaveJob 원본 바이트 스트림에서 애플리케이션 제공 바이트 스트림으로 미디어 데이터를 유지합니다.
IMFSchemeHandler URL에서 미디어 원본 또는 바이트 스트림을 만듭니다.

IMFSecureChannel 두 개체 간에 단방향 보안 채널을 설정합니다.
IMFSeekInfo 특정 검색 위치의 경우 가장 가까운 두 키 프레임을 가져옵니다. (IMFSeekInfo)
IMFSensorActivitiesReport 센서의 현재 활동을 설명하는 IMFSensorActivityReport 개체에 대한 액세스를 제공합니다.
IMFSensorActivitiesReportCallback 센서 활동 보고서를 사용할 수 있을 때 콜백을 수신하기 위해 클라이언트에서 구현하는 인터페이스입니다.
IMFSensorActivityMonitor 센서 활동 모니터를 제어하는 메서드를 제공합니다.
IMFSensorActivityReport 센서에 대한 활동 보고서를 나타냅니다.
IMFSensorDevice IMFSensorGroup 인터페이스로 표시되는 센서 그룹에 속할 수 있는 센서 디바이스를 나타냅니다. 이 컨텍스트에서 "디바이스"라는 용어는 물리적 디바이스, 사용자 지정 미디어 원본 또는 프레임 공급자를 참조할 수 있습니다.
IMFSensorGroup IMFMediaSource를 만들 수 있는 센서 디바이스 그룹을 나타냅니다.
IMFSensorProcessActivity 센서와 연결된 프로세스의 활동을 나타냅니다.
IMFSensorProfile 미디어 파운데이션 센서 프로필을 설명합니다.
IMFSensorProfileCollection 미디어 파운데이션 센서 프로필 개체의 컬렉션을 포함합니다.

IMFSensorStream . (IMFSensorStream)
IMFSensorTransformFactory 미디어 파이프라인이 센서 변환의 요구 사항을 쿼리하고 센서 변환의 런타임 인스턴스를 만들 수 있도록 센서 변환에 의해 구현된 인터페이스입니다.
IMFSequencerSource Sequencer 원본에 의해 구현됩니다.
IMFShutdown 명시적으로 종료해야 하는 일부 Media Foundation 개체에 의해 노출됩니다.
IMFSignedLibrary 콘텐츠 보호 시스템이 서명된 라이브러리에서 함수의 프로시저 주소를 가져올 수 있도록 하는 메서드를 제공합니다. 이 메서드는 Windows 스토어 앱에서 사용할 수 없는 GetProcAddress와 동일한 기능을 제공합니다.
IMFSimpleAudioVolume SAR(스트리밍 오디오 렌더러) 및 오디오 캡처 원본과 연결된 오디오 세션의 마스터 볼륨 수준을 제어합니다.
IMFSourceOpenMonitor 비동기 열기 작업의 진행률에 대한 네트워크 원본으로부터 알림을 수신하는 콜백 인터페이스입니다.
IMFSourceResolver URL 또는 바이트 스트림에서 미디어 원본을 만듭니다.
IMFSSLCertificateManager 클라이언트에서 구현하고 Microsoft Media Foundation에서 호출하여 서버에서 요청한 클라이언트 SSL(Secure Sockets Layer) 인증서를 가져옵니다.
IMFStreamDescriptor 미디어 원본의 한 스트림에 대한 정보를 가져옵니다.
IMFStreamingSinkConfig 콘텐츠를 스트리밍하는 데 사용되는 미디어 싱크에 구성 정보를 전달합니다.

IMFStreamSink 미디어 싱크 개체의 스트림을 나타냅니다.
IMFSystemId 시스템 ID 데이터를 검색하는 메서드를 제공합니다.
IMFTimecodeTranslate 영화 협회와 텔레비전 엔지니어 협회(SMPTE) 시간 코드와 100나노초 시간 단위를 변환합니다.
IMFTimer 지정된 시간에 콜백을 호출하는 타이머를 제공합니다.
IMFTopoLoader 부분 토폴로지에서 전체 토폴로지로 변환합니다.
IMFTopology 토폴로지 나타냅니다. 토폴로지는 특정 순서로 연결된 미디어 원본, 싱크 및 변환의 컬렉션을 설명합니다.
IMFTopologyNode 토폴로지의 노드를 나타냅니다.
IMFTopologyNodeAttributeEditor 미디어 세션의 현재 토폴로지에서 하나 이상의 노드 특성을 업데이트.
IMFTrackedSample 비디오 미디어 샘플의 참조 횟수를 추적합니다.
IMFTranscodeProfile 코드 변환 프로필 개체에 의해 구현됩니다.
IMFTranscodeSinkInfoProvider 트랜스코드 싱크 활성화 개체에 의해 구현됩니다.
IMFTrustedInput ITA(입력 신뢰 기관)를 제공하는 구성 요소에 의해 구현됩니다. 이 인터페이스는 각 구성 요소의 스트림에 대한 ITA를 가져오는 데 사용됩니다.

IMFTrustedOutput OTA(출력 신뢰 기관)를 제공하는 구성 요소에 의해 구현됩니다.
IMFVideoCaptureSampleAllocator 비디오 캡처 디바이스에 대한 특수 기능을 사용하여 비디오 미디어 싱크에 대한 비디오 샘플을 할당합니다.
IMFVideoProcessorControl 비디오 프로세서 MFT를 구성합니다. (IMFVideoProcessorControl)
IMFVideoProcessorControl2 비디오 프로세서 MFT를 구성합니다. (IMFVideoProcessorControl2)
IMFVideoSampleAllocator 비디오 미디어 싱크에 대한 비디오 샘플을 할당합니다.
IMFVideoSampleAllocatorCallback 애플리케이션이 향상된 비디오 렌더러(EVR)에 의해 할당된 비디오 샘플을 추적할 수 있도록 합니다.
IMFVideoSampleAllocatorEx Microsoft Direct3D 11 텍스처 표면을 포함하는 비디오 샘플을 할당합니다.
IMFVideoSampleAllocatorNotify IMFVideoSampleAllocatorCallback 인터페이스에 대한 콜백입니다. (IMFVideoSampleAllocatorNotify)
IMFVideoSampleAllocatorNotifyEx IMFVideoSampleAllocatorCallback 인터페이스에 대한 콜백입니다. (IMFVideoSampleAllocatorNotifyEx)
IMFWorkQueueServices 미디어 세션에서 만든 작업 큐를 제어합니다.
IMFWorkQueueServicesEx IMFWorkQueueServices 인터페이스를 확장합니다.

함수

CreateNamedPropertyStore 이름/값 쌍을 저장할 빈 속성 저장소를 만듭니다.
CreatePropertyStore 빈 속성 저장소 개체를 만듭니다.
MFCreate3GPMediaSink 3GP 파일을 작성하기 위한 미디어 싱크를 만듭니다.
MFCreateAC3MediaSink AC-3 미디어 싱크의 인스턴스를 만듭니다.
MFCreateADTSMediaSink ADTS(오디오 데이터 전송 스트림) 미디어 싱크의 인스턴스를 만듭니다.
MFCreateAggregateSource 미디어 원본 컬렉션을 집계하는 미디어 원본을 만듭니다.
MFCreateAudioRenderer 스트리밍 오디오 렌더러를 만듭니다.
MFCreateAudioRendererActivate 스트리밍 오디오 렌더러에 대한 활성화 개체를 만듭니다.
MFCreateAVIMediaSink Audio-Video AVI(인터리브) 싱크를 만듭니다.
MFCreateCameraControlMonitor IMFCameraControlMonitor의 인스턴스를 만듭니다.
MFCreateCameraOcclusionStateMonitor 카메라 폐색 상태가 변경될 때 애플리케이션이 알림을 받을 수 있도록 하는 IMFCameraOcclusionStateMonitor의 새 인스턴스를 만듭니다.

MFCreateContentDecryptorContext 지정된 미디어 보호 시스템에 대한 IMFContentDecryptorContext 인터페이스를 만듭니다.
MFCreateContentProtectionDevice 지정된 미디어 보호 시스템에 대한 IMFContentProtectionDevice 인터페이스를 만듭니다.
MFCreateCredentialCache 자격 증명 캐시 개체를 만듭니다. 애플리케이션은 이 개체를 사용하여 사용자 지정 자격 증명 관리자를 구현할 수 있습니다.
MFCreateDeviceSource 하드웨어 캡처 디바이스에 대한 미디어 원본을 만듭니다.
MFCreateDeviceSourceActivate 하드웨어 캡처 디바이스를 나타내는 활성화 개체를 만듭니다.
MFCreateFMPEG4MediaSink 조각화된 MP4 파일을 작성하기 위한 미디어 싱크를 만듭니다.
MFCreateMediaSession 애플리케이션 프로세스에서 미디어 세션을 만듭니다.
MFCreateMediaTypeFromProperties 속성에서 IMFMediaType을 만듭니다.
MFCreateMFByteStreamOnStream IStream 포인터를 래핑하는 Microsoft Media Foundation 바이트 스트림을 만듭니다.
MFCreateMFByteStreamOnStreamEx IRandomAccessStream 개체를 래핑하는 Microsoft Media Foundation 바이트 스트림을 만듭니다.
MFCreateMP3MediaSink MP3 미디어 싱크를 만듭니다.
MFCreateMPEG4MediaSink MP4 파일을 작성하기 위한 미디어 싱크를 만듭니다.

MFCreateMuxSink 멀티플렉서 MFT(Microsoft Media Foundation 변환)를 래핑하는 제네릭 미디어 싱크를 만듭니다.
MFCreateNetSchemePlugin 네트워크 원본에 대한 체계 처리기를 만듭니다.
MFCreatePMPMediaSession PMP(보호된 미디어 경로) 프로세스 내에 미디어 세션의 인스턴스를 만듭니다.
MFCreatePMPServer PMP(보호된 미디어 경로) 서버 개체를 만듭니다.
MFCreatePresentationClock 프레젠테이션 시계를 만듭니다.
MFCreatePresentationDescriptor 프레젠테이션 설명자를 만듭니다.
MFCreatePropertiesFromMediaType IMFMediaType에서 속성을 만듭니다.
MFCreateProtectedEnvironmentAccess 콘텐츠 보호 시스템이 보호된 환경과 핸드셰이크를 수행할 수 있도록 하는 IMFProtectedEnvironmentAccess 개체를 만듭니다.
MFCreateProxyLocator 기본 프록시 로케이터를 만듭니다.
MFCreateRelativePanelWatcher **IMFRelativePanelWatcher** 인터페이스의 새 인스턴스를 만듭니다.
MFCreateRemoteDesktopPlugin 원격 데스크톱 플러그 인 개체를 만듭니다. 애플리케이션이 터미널 서비스 클라이언트 세션에서 실행 중인 경우 이 개체를 사용합니다.
MFCreateSampleCopierMFT 샘플 복사기 변환의 인스턴스를 만듭니다.

MFCreateSampleGrabberSinkActivate 샘플 그래버 미디어 싱크에 대한 활성화 개체를 만듭니다.
MFCreateSensorActivityMonitor IMFSensorActivityMonitor 인터페이스의 새 인스턴스를 초기화합니다.
MFCreateSensorGroup 제공된 기호 링크 이름을 기반으로 IMFSensorGroup 인터페이스의 인스턴스를 만듭니다.
MFCreateSensorProfile 지정된 형식, 인덱스 및 선택적 제약 조건에 따라 센서 프로필을 만듭니다.
MFCreateSensorProfileCollection 센서 프로필 컬렉션을 만듭니다.
MFCreateSensorStream IMFSensorStream 인터페이스의 인스턴스를 만듭니다.
MFCreateSequencerSegmentOffset 시퀀서 원본 프레젠테이션 내에서 검색하는 데 사용할 수 있는 PROPVARIANT를 만듭니다.
MFCreateSequencerSource sequencer 원본을 만듭니다.
MFCreateSimpleTypeHandler 한 번에 단일 미디어 형식을 지원하는 미디어 형식 처리기를 만듭니다.
MFCreateSourceResolver URL 또는 바이트 스트림에서 미디어 원본을 만드는 데 사용되는 원본 확인자를 만듭니다.
MFCreateStandardQualityManager 품질 관리자의 기본 구현을 만듭니다.
MFCreateStreamDescriptor 스트림 설명자를 만듭니다.

MFCreateStreamOnMFByteStream Microsoft Media Foundation 바이트 스트림을 래핑하는 IStream 포인터를 반환합니다.
MFCreateStreamOnMFByteStreamEx Microsoft Media Foundation 바이트 스트림을 래핑하는 IRandomAccessStream 개체를 만듭니다.
MFCreateSystemTimeSource 시스템 시간을 기반으로 하는 프레젠테이션 시간 원본을 만듭니다.
MFCreateTopoLoader 토폴로지 로더의 새 인스턴스를 만듭니다.
MFCreateTopology 토폴로지 개체를 만듭니다.
MFCreateTopologyNode 토폴로지 노드를 만듭니다.
MFCreateTrackedSample 비디오 미디어 샘플에서 참조 횟수를 추적하는 IMFTrackedSample 개체를 만듭니다.
MFCreateTranscodeProfile 빈 코드 변환 프로필 개체를 만듭니다.
MFCreateTranscodeSinkActivate 트랜스코드 싱크 활성화 개체를 만듭니다.
MFCreateTranscodeTopology 부분 트랜스코드 토폴로지 만들기
MFCreateTranscodeTopologyFromByteStream 바이트 스트림으로 트랜스코딩하기 위한 토폴로지 만들기
MFCreateVideoRendererActivate 향상된 EVR(비디오 렌더러) 미디어 싱크에 대한 활성화 개체를 만듭니다.

MFCreateWAVEMediaSink WAVE 보관 싱크를 만듭니다. WAVE 보관 싱크는 오디오를 가져와서 .wav 파일에 씁니다.
MFDeserializePresentationDescriptor 바이트 배열에서 프레젠테이션 설명자를 역직렬화합니다.
MFEnumDeviceSources 오디오 또는 비디오 캡처 디바이스 목록을 열거합니다.
MFGetLocalId 로컬 시스템 ID를 가져옵니다.
MFGetService 지정된 서비스 인터페이스에 대한 개체를 쿼리합니다. (MFGetService)
MFGetSupportedMimeTypes 원본 확인자용으로 등록된 MIME 형식을 검색합니다.
MFGetSupportedSchemes 원본 확인자용으로 등록된 URL 구성표를 검색합니다.
MFGetSystemId 시스템 ID 데이터를 검색하기 위한 IMFSystemId 개체를 반환합니다.
MFGetSystemTime 시스템 시간을 반환합니다.
MFGetTopoNodeCurrentType 토폴로지 노드와 연결된 스트림의 미디어 형식을 가져옵니다.
MFIsContentProtectionDeviceSupported 지정된 미디어 보호 시스템에 하드웨어 보안 프로세서가 지원되는지 여부를 확인합니다.
MFLoadSignedLibrary 보호된 환경에 서명된 동적 링크 라이브러리를 로드합니다.

MFRequireProtectedEnvironment 미디어 프레젠테이션에 PMP(보호된 미디어 경로)가 필요한지 여부를 쿼리합니다.
MFSerializePresentationDescriptor 프레젠테이션 설명자를 바이트 배열로 직렬화합니다.
MFShutdownObject Media Foundation 개체를 종료하고 개체와 연결된 모든 리소스를 해제합니다. (MFShutdownObject)
MFTranscodeGetAudioOutputAvailableTypes 오디오 인코더에서 출력 형식 목록을 가져옵니다.

구조체

ASF_FLAT_PICTURE 미디어 원본에 대한 메타데이터로 저장된 이미지를 포함합니다. 이 구조는 WM/Picture 메타데이터 특성의 데이터 항목으로 사용됩니다.
ASF_FLAT_SYNCHRONISED_LYRICS 미디어 원본에 대한 메타데이터로 저장된 동기화된 가사를 포함합니다. 이 구조는 WM/Lyrics_Synchronised 메타데이터 특성의 데이터 항목으로 사용됩니다.
MF_BYTE_STREAM_CACHE_RANGE 바이트 범위를 지정합니다.
MF_CAMERA_CONTROL_RANGE_INFO MF_CAMERA_CONTROL_RANGE_INFO 구조체는 카메라 컨트롤에 허용되는 범위, 단계 값 및 기본 값을 나타냅니다.
MF_LEAKY_BUCKET_PAIR 파일의 버퍼링 요구 사항을 지정합니다.
MF_TRANSCODE_SINK_INFO 트랜스코드 싱크 활성화 개체의 오디오 및 비디오 스트림에 대한 정보를 포함합니다.

MFBYTESTREAM_BUFFERING_PARAMS 네트워크 바이트 스트림에 대한 버퍼링 매개 변수를 지정합니다.
MFCLOCK_PROPERTIES 클록의 속성을 정의합니다.
MFCONTENTPROTECTIONDEVICE_INPUT_DATA 보호 시스템 함수에 대한 입력으로 제공하려는 데이터에 대한 정보를 포함합니다.
MFCONTENTPROTECTIONDEVICE_OUTPUT_DATA 보호 시스템 함수에서 출력으로 받은 데이터에 대한 정보를 포함합니다.
MFCONTENTPROTECTIONDEVICE_REALTIMECLIENT_DATA 실시간 작업을 예상 우선 순위에 따라 예약할 수 있도록 MMCSS(멀티미디어 클래스 스케줄러 서비스) 매개 변수의 보안 프로세서를 권장합니다.
MFINPUTTRUSTAUTHORITY_ACCESS_ACTION OTA(출력 신뢰 기관)에서 요청한 작업에 대해 설명합니다. 요청은 ITA(입력 신뢰 기관)로 전송됩니다.
MFINPUTTRUSTAUTHORITY_ACCESS_PARAMS IMFInputTrustAuthority::BindAccess 또는 IMFInputTrustAuthority::UpdateAccess 메서드에 대한 매개 변수를 포함합니다.
MFMediaKeyStatus CDM(콘텐츠 암호 해독 모듈) 세션 키의 상태를 나타냅니다.
MFNetCredentialManagerGetParam 자격 증명 관리자에 대한 인증 정보를 포함합니다.
MFRR_COMPONENT_HASH_INFO 해지된 구성 요소에 대한 정보를 포함합니다.
MFRR_COMPONENTS 하나 이상의 해지된 구성 요소에 대한 정보를 포함합니다.

MFT_REGISTRATION_INFO IMFLocalMFTRegistration::RegisterMFTs 메서드에 대한 매개 변수를 포함합니다.
MFTOPONODE_ATTRIBUTE_UPDATE 토폴로지 노드에 대한 새 특성 값을 지정합니다.
SENSORPROFILEID 센서 프로필 ID를 설명합니다.

열거형

MF_CAMERA_CONTROL_CONFIGURATION_TYPE 카메라 컨트롤의 구성 유형을 지정합니다.
MF_CONNECT_METHOD 토폴로지 로더가 토폴로지 노드를 연결하는 방법을 지정합니다.
MF_CROSS_ORIGIN_POLICY HTML5 미디어 요소에서 사용하는 W3C CORS(원본 간 설정) 특성에 매핑합니다.
MF_MEDIAKEY_STATUS CDM(콘텐츠 암호 해독 모듈) 세션 키의 상태를 지정합니다.
MF_MEDIAKEYSESSION_MESSAGETYPE CDM(콘텐츠 암호 해독 모듈) 메시지의 형식을 지정합니다.
MF_MEDIAKEYSESSION_TYPE IMFContentDecryptionModuleSession 개체로 표현되는 CDM(Content Decryption Module) 세션의 형식을 지정합니다.
MF_MEDIASOURCE_STATUS_INFO . (MF_MEDIASOURCE_STATUS_INFO)

MF_OBJECT_TYPE 원본 확인자에서 만든 개체 형식을 정의합니다.
MF_OPM_ACP_PROTECTION_LEVEL MFPROTECTION_ACP 대한 보호 수준을 정의합니다.
MF_OPM_CGMSA_PROTECTION_LEVEL MFPROTECTION_CGMSA 대한 보호 수준을 정의합니다.
MF_QUALITY_ADVISE_FLAGS IMFQualityAdvise2::NotifyQualityEvent 메서드에 대한 플래그를 포함합니다.
MF_QUALITY_DROP_MODE 파이프라인 구성 요소가 샘플을 삭제하는 방법을 적극적으로 지정합니다.
MF_QUALITY_LEVEL 파이프라인 구성 요소의 품질 수준을 지정합니다.
MF_TOPOLOGY_RESOLUTION_STATUS_FLAGS MF_TOPOLOGY_RESOLUTION_STATUS 특성의 상태 플래그를 정의합니다.
MF_TOPOLOGY_TYPE 토폴로지 노드의 형식을 정의합니다.
MF_TOPONODE_DRAIN_MODE 토폴로지의 변환이 드레이닝되는 시간을 정의합니다.
MF_TOPONODE_FLUSH_MODE 토폴로지의 변환이 플러시되는 시기를 정의합니다.
MF_TRANSCODE_ADJUST_PROFILE_FLAGS MF_TRANSCODE_ADJUST_PROFILE 특성에 설정된 프로필 플래그를 정의합니다.
MF_TRANSCODE_TOPOLOGYMODE_FLAGS MF_TRANSCODE_TOPOLOGYMODE 특성에 대한 플래그를 정의합니다.

MF_TRANSFER_VIDEO_FRAME_FLAGS . (MF_TRANSFER_VIDEO_FRAME_FLAGS)
MF_URL_TRUST_STATUS URL이 신뢰할 수 있는 원본에서 온 것인지 여부를 나타냅니다.
MF_VIDEO_PROCESSOR_ALGORITHM_TYPE MF_VIDEO_PROCESSOR_ALGORITHM 사용하는 비디오 프로세서에 대한 알고리즘을 정의합니다.
MF_VIDEO_PROCESSOR_MIRROR 비디오 이미지를 대칭 이동하는 방법을 지정합니다.
MF_VIDEO_PROCESSOR_ROTATION 비디오 이미지를 회전하는 방법을 지정합니다.
MFAudioConstriction 오디오 수축 값을 지정합니다.
MFCameraOcclusionState 카메라의 폐색 상태를 지정합니다.
MFCLOCK_CHARACTERISTICS_FLAGS 시계의 특성을 설명하는 플래그를 포함합니다.
MFCLOCK_RELATIONAL_FLAGS 클록의 속성을 정의합니다.
MFCLOCK_STATE 클록의 상태를 정의합니다.
MFMEDIASOURCE_CHARACTERISTICS 미디어 원본의 특성을 정의합니다.
MFNET_PROXYSETTINGS 기본 프록시 로케이터가 프록시 서버에 대한 연결 설정을 지정하는 방법을 지정합니다.

MFNetAuthenticationFlags 사용자의 자격 증명을 사용하는 방법을 지정합니다.
MFNetCredentialOptions 캐싱 네트워크 자격 증명에 대한 옵션을 설명합니다.
MFNetCredentialRequirements 자격 증명 관리자가 사용자 자격 증명을 가져오는 방법을 지정합니다.
MFNETSOURCE_CACHE_STATE 미디어 파일 또는 항목에 대한 캐시 상태를 정의합니다.
MFNETSOURCE_PROTOCOL_TYPE 스트리밍 또는 다운로드에 사용되는 제어 프로토콜의 유형을 나타냅니다.
MFNETSOURCE_STATISTICS_IDS 네트워크 원본에서 수집한 통계를 정의합니다.
MFNETSOURCE_TRANSPORT_TYPE 스트리밍 또는 다운로드 데이터(TCP 또는 UDP)에 사용되는 전송 유형에 대해 설명합니다.
MFPMPSESSION_CREATION_FLAGS MFCreatePMPMediaSession 함수의 동작을 정의하는 플래그를 포함합니다.
MFPOLICYMANAGER_ACTION 스트림에서 수행할 수 있는 작업을 정의합니다.
MFRATE_DIRECTION 재생 방향(정방향 또는 역방향)을 지정합니다.
MFSampleAllocatorUsage MFSampleAllocatorUsage 열거형은 IMFSampleAllocatorControl 인터페이스를 구현하는 구성 요소의 할당자 사용을 지정합니다.
MFSensorDeviceMode IMFSensorDevice의 공유 모드를 지정합니다.

MFSensorDeviceType 센서 디바이스의 유형을 지정합니다. 이 열거형의 값은 IMFSensorDevice::GetDeviceType에서 반환됩니다.
MFSensorStreamType IMFSensorDevice와 연결된 스트림이 입력 또는 출력 스트림인지 여부를 지정합니다.
MFSequencerTopologyFlags 시퀀서 원본에 토폴로지를 추가하거나 큐에 이미 있는 토폴로지를 업데이트하기 위한 플래그를 포함합니다.
MFSESSION_GETFULLTOPOLOGY_FLAGS IMFMediaSession::GetFullTopology 메서드에 대한 플래그를 정의합니다.
MFSESSION_SETTOPOLOGY_FLAGS IMFMediaSession::SetTopology 메서드의 동작을 정의합니다.
MFSHUTDOWN_STATUS IMFShutdown::Shutdown 메서드에 대한 호출의 현재 상태를 설명합니다.
MFSTREAMSINK_MARKER_TYPE IMFStreamSink::P laceMarker 메서드에 대한 스트림 마커 정보를 정의합니다.
MFTIMER_FLAGS IMFTimer::SetTimer 메서드에 대한 플래그를 포함합니다.
MFTOPOLOGY_DXVA_MODE 토폴로지 로더가 토폴로지에서 Microsoft DirectX Video Acceleration(DXVA)을 사용할 수 있는지 여부를 지정합니다.
MFTOPOLOGY_HARDWARE_MODE 토폴로지 로더가 하드웨어 기반 MFT(Media Foundation 변환)를 토폴로지에 삽입할지 여부를 지정합니다.
SAMPLE_PROTECTION_VERSION 샘플 보호를 위한 버전 번호를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mfmediacapture.h 헤더

아티클2023. 08. 24.

이 헤더는 Microsoft Media Foundation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft Media Foundation

mfmediacapture.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAdvancedMediaCapture 고급 미디어 캡처를 사용하도록 설정합니다.
IAdvancedMediaCaptureInitializationSettings 고급 미디어 캡처에 대한 초기화 설정을 제공합니다.
IAdvancedMediaCaptureSettings 고급 미디어 캡처에 대한 설정을 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mfmediaengine.h 헤더

이 헤더는 Microsoft Media Foundation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft Media Foundation

mfmediaengine.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Interfaces

테이블 확장

IMFBufferListNotify IMFSourceBufferList 개체가 클라이언트에 중요한 상태 변경을 알릴 수 있도록 합니다.
IMFCdmSuspendNotify 일시 중단 전에 전역 리소스를 일관된 상태로 전환해야 하는 경우 클라이언트가 CDM(콘텐츠 암호 해독 모듈)에 알릴 수 있도록 하는 데 사용됩니다.
IMFExtendedDRMTypeSupport 지정된 키 시스템에 대해 지정된 콘텐츠 형식이 지원되는지 여부를 쿼리하는 메서드를 제공합니다.
IMFMediaEngine 애플리케이션에서 오디오 또는 비디오 파일을 재생할 수 있도록 합니다.
IMFMediaEngineClassFactory 미디어 엔진의 인스턴스를 만듭니다.
IMFMediaEngineClassFactory2 IMFMediaKeys 개체의 인스턴스를 만듭니다.
IMFMediaEngineClassFactory4 지정된 키 시스템에 대한 CDM(Content Decryption Module) 개체의 클래스 팩터리인 IMFContentDecryptionModuleFactory 인스턴스를 만들기 위한 CreateContentDecryptionModuleFactory 메서드를 제공합니다.
IMFMediaEngineClassFactoryEx IMFMediaEngineClassFactory 인터페이스에 대한 확장입니다.

IMFMediaEngineEME 암호화된 미디어 확장 메서드를 추가하기 위해 미디어 엔진에 의해 구현됩니다.
IMFMediaEngineEx IMFMediaEngine 인터페이스를 확장합니다.
IMFMediaEngineExtension 애플리케이션이 미디어 엔진에서 미디어 리소스를 로드할 수 있도록 합니다.
IMFMediaEngineNeedKeyNotify 키 요청 데이터를 알리기 위해 미디어 엔진에 대한 콜백을 나타냅니다.
IMFMediaEngineNotify IMFMediaEngine 인터페이스에 대한 콜백 인터페이스입니다.
IMFMediaEngineOPMInfo OPM(출력 보호 관리자)에 대한 정보를 가져오는 메서드를 제공합니다.
IMFMediaEngineProtectedContent 미디어 엔진이 보호된 비디오 콘텐츠를 재생할 수 있도록 합니다.
IMFMediaEngineSrcElements 미디어 엔진에 미디어 리소스 목록을 제공합니다.
IMFMediaEngineSrcElementsEx IMFMediaEngineSrcElements 인터페이스를 확장하여 추가 기능을 제공합니다.
IMFMediaEngineSupportsSourceTransfer 미디어 엔진과 Play To의 공유 엔진 간에 미디어 원본을 전송할 수 있습니다.
IMFMediaEngineWebSupport 웹 오디오 재생을 사용하도록 설정합니다.
IMFMediaError 미디어 엔진에 대한 현재 오류 상태를 제공합니다.
IMFMediaKeys

DRM(디지털 권한 관리) 키 시스템을 사용하여 미디어 데이터의 암호를 해독하는 데 사용되는 미디어 키를 나타냅니다.
IMFMediaKeySession DRM(디지털 권한 관리) 키 시스템을 사용하는 세션을 나타냅니다.
IMFMediaKeySessionNotify 미디어 키 세션에 대한 정보를 앱에 알리는 메커니즘을 제공합니다.
IMFMediaSourceExtension MSE(미디어 원본 확장)에 대한 기능을 제공합니다.
IMFMediaSourceExtensionNotify IMFMediaSourceExtension과 관련된 이벤트를 발생시키는 기능을 제공합니다.
IMFMediaTimeRange 각 범위가 시작 및 종료 시간으로 정의되는 시간 범위 목록을 나타냅니다.
IMFSourceBuffer IMFMediaSourceExtension에 대한 미디어 데이터가 포함된 버퍼를 나타냅니다.
IMFSourceBufferList IMFSourceBuffer 개체의 컬렉션을 나타냅니다.
IMFSourceBufferNotify IMFSourceBuffer와 관련된 이벤트를 발생시키는 기능을 제공합니다.
IMFTimedText 시간 제한 텍스트 개체는 시간 제한 텍스트의 구성 요소를 나타냅니다.
IMFTimedTextBinary 시간 제한 텍스트 개체의 데이터 콘텐츠를 나타냅니다.
IMFTimedTextCue 시간 제한 텍스트 큐 개체를 나타냅니다.
IMFTimedTextFormattedText 형식이 지정된 시간 제한 텍스트 블록을 나타냅니다.

IMFTimedTextNotify Microsoft Media Foundation 시간 제한 텍스트 알림에 대한 콜백을 정의하는 인터페이스입니다.
IMFTimedTextRegion 시간 제한 텍스트 개체의 표시 영역을 나타냅니다.
IMFTimedTextStyle 시간 제한 텍스트의 스타일을 나타냅니다.
IMFTimedTextTrack 시간 제한 텍스트의 트랙을 나타냅니다.
IMFTimedTextTrackList 시간 제한 텍스트 트랙 목록을 나타냅니다.

구조

테이블 확장

MFVideoNormalizedRect MFVideoNormalizedRect(mfmediaengine.h) 구조체는 비디오 사각형에서 하위 사각형을 지정하는 데 사용되는 정규화된 사각형을 정의합니다.

열거 목록

테이블 확장

MF_MEDIA_ENGINE_CANPLAY 미디어 엔진이 지정된 형식의 미디어 리소스를 재생할 수 있는 가능성을 지정합니다.
MF_MEDIA_ENGINE_CREATEFLAGS IMFMediaEngineClassFactory::CreateInstance 메서드에 대한 플래그를 포함합니다.

MF_MEDIA_ENGINE_ERR 미디어 엔진에 대한 오류 상태 코드를 정의합니다.
MF_MEDIA_ENGINE_EVENT 미디어 엔진에 대한 이벤트 코드를 정의합니다.
MF_MEDIA_ENGINE_EXTENSION_TYPE 미디어 엔진 확장 유형을 지정합니다.
MF_MEDIA_ENGINE_FRAME_PROTECTION_FLAGS 비디오 프레임에 대한 콘텐츠 보호 요구 사항을 지정합니다.
MF_MEDIA_ENGINE_KEYERR 미디어 엔진에 대한 미디어 키 오류 코드를 정의합니다.
MF_MEDIA_ENGINE_NETWORK 미디어 엔진에 대한 네트워크 상태 코드를 정의합니다.
MF_MEDIA_ENGINE_OPM_STATUS OPM(출력 보호 관리자)의 상태를 정의합니다.
MF_MEDIA_ENGINE_PRELOAD 미디어 엔진에 대한 미리 로드 힌트를 정의합니다.
MF_MEDIA_ENGINE_PROTECTION_FLAGS 미디어 엔진이 보호된 콘텐츠를 재생할지 여부와 미디어 엔진이 PMP(보호된 미디어 경로)를 사용할지 여부를 지정하는 플래그를 포함합니다.
MF_MEDIA_ENGINE_READY 미디어 엔진에 대한 준비 상태 값을 정의합니다.
MF_MEDIA_ENGINE_S3D_PACKING_MODE 압축된 3D 비디오 프레임의 레이아웃을 지정합니다.
MF_MEDIA_ENGINE_SEEK_MODE 미디어 엔진 검색 모드에 대한 값을 정의합니다.

MF_MEDIA_ENGINE_STATISTIC 미디어 엔진이 재생 중에 추적하는 통계를 식별합니다.
MF_MEDIA_ENGINE_STREAMTYPE_FAILED 실패한 미디어 스트림의 종류를 식별합니다.
MF_MSE_ERROR 미디어 원본 확장의 다양한 오류 상태를 정의합니다.
MF_MSE_READY 미디어 원본 확장의 다양한 준비 상태를 정의합니다.
MF_TIMED_TEXT_ALIGNMENT 부모 블록 요소에서 텍스트가 정렬되는 방식을 지정합니다.
MF_TIMED_TEXT_CUE_EVENT 시간 제한 텍스트 큐 이벤트의 형식을 지정합니다.
MF_TIMED_TEXT_DECORATION 텍스트가 데코레이팅되는 방식(밑줄 등)을 지정합니다.
MF_TIMED_TEXT_DISPLAY_ALIGNMENT 텍스트가 디스플레이에 맞춰지는 방식을 지정합니다.
MF_TIMED_TEXT_ERROR_CODE 시간 제한 텍스트 트랙에서 발생한 종류 오류를 지정합니다.
MF_TIMED_TEXT_FONT_STYLE 시간 제한 텍스트의 글꼴 스타일을 지정합니다.
MF_TIMED_TEXT_SCROLL_MODE 부모 요소를 스크롤할 때 텍스트가 표시되는 방식을 지정합니다.
MF_TIMED_TEXT_TRACK_KIND 시간 제한 텍스트 트랙의 종류를 지정합니다.
MF_TIMED_TEXT_UNIT_TYPE

시간 제한 텍스트를 측정할 단위를 지정합니다.
MF_TIMED_TEXT_WRITING_MODE 부모 요소에 텍스트를 쓰는 시퀀스를 지정합니다.

Last updated on 2026. 01. 26.

mfmp2dlna.h 헤더

아티클2023. 03. 16.

이 헤더는 Microsoft Media Foundation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft Media Foundation

mfmp2dlna.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IMFDLNASinkInit DLNA(Digital Living Network Alliance) 미디어 싱크를 초기화합니다. (IMFDLNASinkInit)

구조체

MFMPEG2DLNASINKSTATS DLNA(Digital Living Network Alliance) 미디어 싱크의 인코딩 통계를 포함합니다.

mfobjects.h 헤더

아티클2024. 03. 05.

이 헤더는 Microsoft Media Foundation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft Media Foundation

mfobjects.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

테이블 확장

IMF2DBuffer 비디오 프레임과 같은 2차원 표면을 포함하는 버퍼를 나타냅니다. (IMF2DBuffer)
IMF2DBuffer2 비디오 프레임과 같은 2차원 표면을 포함하는 버퍼를 나타냅니다. (IMF2DBuffer2)
IMFActivate 애플리케이션에서 개체 만들기를 연기할 수 있도록 합니다.
IMFAsyncCallback 비동기 메서드가 완료되면 애플리케이션에 알리는 콜백 인터페이스입니다. (IMFAsyncCallback)
IMFAsyncCallbackLogging 비동기 콜백이 연결된 부모 개체에 대한 로깅 정보를 제공합니다.
IMFAsyncResult 비동기 작업의 결과에 대한 정보를 제공합니다. (IMFAsyncResult)
IMFAttributes 개체에 키/값 쌍을 저장하는 일반적인 방법을 제공합니다.
IMFAudioMediaType IMFAudioMediaType은 Windows 7을 기준으로 더 이상 사용할 수 없습니다.

IMFByteStream 로컬 파일, 네트워크 파일 또는 다른 원본일 수 있는 일부 데이터 원본의 바이트 스트림을 나타냅니다.
IMFByteStreamProxyClassFactory 바이트 스트림에 대한 프록시를 만듭니다. (IMFByteStreamProxyClassFactory)
IMFCollection IUnknown 포인터의 제네릭 컬렉션을 나타냅니다.
IMFDXGIBuffer Microsoft DXGI(DirectX Graphics Infrastructure) 표면이 포함된 버퍼를 나타냅니다.
IMFDXGIDeviceManager 두 스레드가 동일한 Microsoft Direct3D 11 디바이스를 공유할 수 있도록 합니다.
IMFMediaBuffer 미디어 데이터를 포함하는 메모리 블록을 나타냅니다.
IMFMediaEvent Media Foundation 개체에서 생성된 이벤트를 나타냅니다. 이 인터페이스를 사용하여 이벤트에 대한 정보를 가져옵니다.
IMFMediaEventGenerator 이벤트를 생성하는 Media Foundation 개체에서 이벤트를 검색합니다.
IMFMediaEventQueue IMFMediaEventGenerator 인터페이스를 구현해야 하는 애플리케이션에 대한 이벤트 큐를 제공합니다.
IMFMediaType 미디어 형식에 대한 설명을 나타냅니다.
IMFMuxStreamAttributesManager 멀티플렉싱된 미디어 원본의 하위 스트림에 대한 IMFAttributes에 대한 액세스를 제공합니다.
IMFMuxStreamMediaTypeManager

멀티플렉싱된 미디어 원본에 대한 스트림 구성 관리를 사용하도록 설정합니다. 스트림 구성은 멀티플렉싱된 출력을 포함할 수 있는 하위 스트림 집합을 정의합니다.
IMFMuxStreamSampleManager 멀티플렉싱된 미디어 원본의 출력 내에서 개별 하위 스트림에 대한 IMFSample 개체를 검색하는 기능을 제공합니다.
IMFPluginControl Microsoft Media Foundation에서 미디어 원본 및 변환을 열거하는 방법을 제어합니다. (IMFPluginControl)
IMFPluginControl2 Microsoft Media Foundation에서 미디어 원본 및 변환을 열거하는 방법을 제어합니다. (IMFPluginControl2)
IMFRemoteAsyncCallback Microsoft Media Foundation 프록시/스텁 DLL에서 프로세스 경계를 넘어 특정 비동기 메서드 호출을 마샬링하는 데 사용됩니다. 애플리케이션은 이 인터페이스를 사용하거나 구현하지 않습니다.
IMFSample 미디어 데이터의 컨테이너 개체인 미디어 샘플을 나타냅니다.
IMFSampleOutputStream 미디어 샘플을 바이트 스트림에 씁니다.
IMFVideoMediaType 비디오 형식에 대한 설명을 나타냅니다.

Functions

테이블 확장

MFDeserializeAttributesFromStream 스트림에서 특성 저장소로 특성을 로드합니다.
MFSerializeAttributesToStream

특성 저장소의 내용을 스트림에 씁니다.

구조체

테이블 확장

MFARGB ARGB 색 값을 설명합니다.
MFAYUVSample 4:4:4:4 Y'Cb'Cr' 샘플을 설명합니다.
MFOffset 오프셋을 고정 소수점 실수로 지정합니다.
MFPaletteEntry 색 테이블에 하나의 색상표 항목을 포함합니다.
MFRatio 비율을 나타냅니다.
MFVideoArea 비디오 프레임 내의 사각형 영역을 지정합니다.
MFVideoCompressedInfo 비디오 압축 형식에 대한 정보를 포함합니다. 이 구조체는 MFVIDEOFORMAT 구조체에서 사용됩니다.
MFVIDEOFORMAT 비디오 형식을 설명합니다.
MFVideoInfo 압축된 형식과 압축되지 않은 형식 모두에 적용되는 비디오 형식 정보를 포함합니다. 이 구조체는 MFVIDEOFORMAT 구조체에서 사용됩니다.

MFVideoSurfaceInfo 압축되지 않은 비디오 형식에 대한 정보를 포함합니다. 이 구조체는 MFVIDEOFORMAT 구조체에서 사용됩니다.

열거형

테이블 확장

MF_ATTRIBUTE_SERIALIZE_OPTIONS 특성 저장소를 직렬화 및 역직렬화하기 위한 플래그를 정의합니다.
MF_ATTRIBUTE_TYPE 키/값 쌍의 데이터 형식을 정의합니다.
MF_ATTRIBUTES_MATCH_TYPE 두 개체의 특성을 비교하는 방법을 지정합니다.
MF_PLUGIN_CONTROL_POLICY IMFPluginControl2::SetPolicy 메서드에 대한 정책 설정을 정의합니다.
MF_Plugin_Type IMFPluginControl 인터페이스의 개체 형식을 지정합니다.
MF2DBuffer_LockFlags IMF2DBuffer2::Lock2DSize 메서드에 대한 플래그를 포함합니다.
MFBYTESTREAM_SEEK_ORIGIN 검색 요청의 원본을 지정합니다.
MFNominalRange 색 데이터에 헤드룸 및 발가락이 포함되는지 여부를 지정합니다.
MFStandardVideoFormat 일반적인 비디오 형식을 지정하는 값을 포함합니다.

MFVideoChromaSubsampling Y'Cb'Cr' 데이터에 대한 크로마 인코딩 체계를 정의하는 플래그를 포함합니다.
MFVideoFlags 비디오 스트림을 설명하는 플래그를 포함합니다.
MFVideoInterlaceMode 비디오 스트림이 인터레이스되는 방법을 지정합니다.
MFVideoLighting 특정 비디오 콘텐츠 집합을 보기 위한 최적의 조명에 대해 설명합니다.
MFVideoPrimaries 비디오 원본의 색 기본값을 지정합니다. (MFVideoPrimaries)
MFVideoTransferFunction 선형 RGB에서 비선형 RGB(R'G'B')로 변환 함수를 지정합니다. (MFVideoTransferFunction)
MFVideoTransferMatrix Y'PbPr(구성 요소 비디오)과 스튜디오 R'G'B' 간의 변환 매트릭스에 대해 설명합니다. (MFVideoTransferMatrix)

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

mfreadwrite.h 헤더

아티클2023. 08. 24.

이 헤더는 Microsoft Media Foundation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft Media Foundation

mfreadwrite.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IMFReadWriteClassFactory 싱크 작성기 또는 원본 판독기의 instance 만듭니다.
IMFSinkWriter Microsoft Media Foundation 싱크 작성기 개체에 의해 구현됩니다.
IMFSinkWriterCallback Microsoft Media Foundation 싱크 작성기에 대한 콜백 인터페이스입니다.
IMFSinkWriterCallback2 IMFSinkWriterCallback 인터페이스를 확장합니다.
IMFSinkWriterEncoderConfig 미디어 유형 및 인코더 구성을 동적으로 변경하기 위한 싱크 작성기에 추가 기능을 제공합니다.
IMFSinkWriterEx IMFSinkWriter 인터페이스를 확장합니다.
IMFSourceReader Microsoft Media Foundation 원본 판독기 개체에 의해 구현됩니다.
IMFSourceReaderCallback Microsoft Media Foundation 원본 판독기용 콜백 인터페이스입니다.

IMFSourceReaderCallback2 IMFSourceReaderCallback 인터페이스를 확장합니다.
IMFSourceReaderEx IMFSourceReader 인터페이스를 확장합니다.

Functions

MFCreateSinkWriterFromMediaSink 미디어 싱크에서 싱크 작성기를 만듭니다.
MFCreateSinkWriterFromURL URL 또는 바이트 스트림에서 싱크 작성기를 만듭니다.
MFCreateSourceReaderFromByteStream 바이트 스트림에서 원본 판독기를 만듭니다.
MFCreateSourceReaderFromMediaSource 미디어 원본에서 원본 판독기를 만듭니다.
MFCreateSourceReaderFromURL URL에서 원본 판독기를 만듭니다.

구조체

MF_SINK_WRITER_STATISTICS 싱크 작성기의 성능에 대한 통계를 포함합니다.

열거형

MF_SOURCE_READER_CONTROL_FLAG IMFSourceReader::ReadSample 메서드에 대한 플래그를 포함합니다.
MF_SOURCE_READER_FLAG IMFSourceReader::ReadSample 메서드의 상태 나타내는 플래그를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mfsharingengine.h 헤더

아티클2023. 08. 24.

이 헤더는 Microsoft Media Foundation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft Media Foundation

mfsharingengine.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IMFImageSharingEngine 이미지 공유를 사용하도록 설정합니다.
IMFImageSharingEngineClassFactory IMFImageSharingEngine의 instance 만듭니다.
IMFMediaSharingEngine 미디어 공유를 사용하도록 설정합니다.
IMFMediaSharingEngineClassFactory IMFMediaSharingEngine의 instance 만듭니다. (IMFMediaSharingEngineClassFactory)
IMFSharingEngineClassFactory 미디어 공유 엔진의 instance 만듭니다. (IMFSharingEngineClassFactory)
IPlayToControl PlayToConnection 개체가 미디어 요소에 연결할 수 있도록 합니다.
IPlayToControlWithCapabilities IPlayToSource에 대한 기능을 제공하여 콘텐츠의 기능을 확인합니다.
IPlayToSourceClassFactory PlayToSource 개체의 instance 만듭니다.

구조체

DEVICE_INFO 미디어 공유 디바이스에 대한 정보를 포함합니다.

열거형

MF_SHARING_ENGINE_EVENT 공유 엔진에서 보내는 이벤트를 정의합니다.
PLAYTO_SOURCE_CREATEFLAGS IPlayToSourceClassFactory::CreateInstance 메서드에 대한 플래그를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mfspatialaudio.h 헤더

아티클2023. 03. 16.

이 헤더는 Microsoft Media Foundation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft Media Foundation

mfspatialaudio.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IMFSpatialAudioObjectBuffer 연결된 위치 및 렌더링 메타데이터가 있는 오디오 데이터의 섹션을 나타냅니다. 공간 오디오 개체는 IMFSpatialAudioSample 인스턴스에 저장되며 Media Foundation 구성 요소 간에 공간 오디오 정보를 전달할 수 있습니다.
IMFSpatialAudioSample 공간 사운드 정보가 있는 멀티미디어 샘플을 나타냅니다. 모든 IMFSpatialAudioSample에는 하나 이상의 IMFSpatialAudioObjectBuffer 개체가 포함됩니다.

mftransform.h 헤더

아티클2023. 08. 24.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft Media Foundation
- 스트리밍 미디어 DDI 참조

mftransform.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IMFDeviceTransform 이 섹션에는 IMFDeviceTransform 인터페이스에 대한 참조 정보가 포함되어 있습니다.
IMFDeviceTransformCallback 시스템 할당 프레임 버퍼가 디바이스 드라이버로 전송될 때 알림을 수신하도록 이 콜백을 구현합니다.
IMFTransform 모든 MFT(Media Foundation Transforms)에서 구현됩니다.

Functions

MFCreateTransformActivate MFT(Media Foundation 변환)에 대한 제네릭 활성화 개체를 만듭니다.

구조체

MFT_INPUT_STREAM_INFO

MFT(Media Foundation 변환)의 입력 스트림에 대한 정보를 포함합니다. 이러한 값을 얻으려면 IMFTransform::GetInputStreamInfo를 호출합니다.
MFT_OUTPUT_DATA_BUFFER Media Foundation 변환에 대한 출력 버퍼에 대한 정보를 포함합니다. 이 구조체는 IMFTransform::ProcessOutput 메서드에 사용됩니다.
MFT_OUTPUT_STREAM_INFO MFT(Media Foundation 변환)의 출력 스트림에 대한 정보를 포함합니다. 이러한 값을 얻으려면 IMFTransform::GetOutputStreamInfo를 호출합니다.
STREAM_MEDIUM 애플리케이션용이 아닙니다.

열거형

_MFT_DRAIN_TYPE 지원되지 않습니다. (_MFT_DRAIN_TYPE)
_MFT_INPUT_DATA_BUFFER_FLAGS IMFTransform::ProcessInput 메서드에 대한 플래그를 정의합니다. 현재는 플래그가 정의되어 있지 않습니다.
_MFT_INPUT_STATUS_FLAGS MFT(Media Foundation 변환)에서 입력 스트림의 상태 나타냅니다.
_MFT_INPUT_STREAM_INFO_FLAGS MFT(Media Foundation 변환)의 입력 스트림에 대해 설명합니다.
_MFT_OUTPUT_DATA_BUFFER_FLAGS IMFTransform::ProcessOutput 메서드에 대한 플래그를 정의합니다.
_MFT_OUTPUT_STATUS_FLAGS MFT(Media Foundation 변환)가 출력 데이터를 생성할 수 있는지 여부를 나타냅니다.
_MFT_OUTPUT_STREAM_INFO_FLAGS

MFT(Media Foundation 변환)의 출력 스트림에 대해 설명합니다.
_MFT_PROCESS_OUTPUT_FLAGS MFT(Media Foundation 변환)에서 출력 샘플을 처리하기 위한 플래그를 정의합니다.
_MFT_PROCESS_OUTPUT_STATUS IMFTransform::P rocessOutput에 대한 호출의 상태 나타냅니다.
_MFT_SET_TYPE_FLAGS MFT(Media Foundation 변환)에서 미디어 형식 설정 또는 테스트에 대한 플래그를 정의합니다.
MF3DVideoOutputType 3D 입체 비디오 스트림을 출력하는 방법을 지정합니다.
MFT_MESSAGE_TYPE MFT(Media Foundation 변환)에 대한 메시지를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mfvirtualcamera.h 헤더

아티클2023. 08. 24.

이 헤더는 Microsoft Media Foundation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft Media Foundation

mfvirtualcamera.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IMFCameraSyncObject 가상 카메라를 만들고 관리하는 앱과 가상 카메라 원본 간의 동기화 메커니즘을 제공합니다.
IMFVirtualCamera Media Foundation 프레임 서버 파이프라인에 연결할 수 있는 가상 카메라를 나타냅니다.

Functions

MFCreateVirtualCamera 호출자가 시스템에서 가상 카메라를 등록, 등록 취소 또는 제거하는 데 사용할 수 있는 가상 카메라 개체를 만듭니다.
MFIsVirtualCameraTypeSupported 플랫폼에서 특정 가상 카메라가 지원되는지 여부를 나타내는 값을 반환합니다.

열거형

MFVirtualCameraAccess 가상 카메라에 대한 액세스 제한을 지정합니다.

MFVirtualCameraLifetime 가상 카메라의 수명을 지정합니다.
MFVirtualCameraType 가상 카메라의 유형을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mgm.h 헤더

아티클2023. 08. 24.

이 헤더는 라우팅 및 원격 액세스 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 라우팅 및 원격 액세스 서비스

mgm.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

MgmAddGroupMembershipEntry MgmAddGroupMembershipEntry 함수는 멀티캐스트 그룹 관리자에게 지정된 인터페이스에 지정된 그룹에 대한 새 수신기가 있음을 알릴 수 있습니다.
MgmDeleteGroupMembershipEntry MgmDeleteGroupMembershipEntry 함수는 멀티캐스트 그룹 관리자에게 지정된 인터페이스에 지정된 그룹에 대한 수신기가 더 이상 없음을 알릴 수 있습니다.
MgmDeRegisterMProtocol MgmDeRegisterMProtocol 함수는 MgmRegisterMProtocol 호출에서 얻은 클라이언트 핸들을 등록 취소합니다.
MgmGetFirstMfe MgmGetFirstMfe 함수는 MFE 목록의 시작 부분에서 시작하는 MFE를 검색합니다.
MgmGetFirstMfeStats MgmGetFirstMfeStats 함수는 MFE 목록의 시작 부분에서 시작하는 MFE 통계를 검색합니다.
MgmGetMfe MgmGetMfe 함수는 특정 MFE를 검색합니다.
MgmGetMfeStats MgmGetMfeStats 함수는 특정 MFE에 대한 통계를 검색합니다. 반환되는 통계에는 수신된 패킷, 받은 바이트 및 나가는 각 인터페이스에서 전달된 패킷이 포함됩니다.

MgmGetNextMfe MgmGetNextMfe 함수는 하나 이상의 MFE를 검색합니다.
MgmGetNextMfeStats MgmGetNextMfeStats 함수는 하나 이상의 MFE 통계 집합을 검색합니다.
MgmGetProtocolOnInterface MgmGetProtocolOnInterface 함수는 지정된 인터페이스를 소유하는 멀티캐스트 라우팅 프로토콜의 프로토콜 ID를 검색합니다.
MgmGroupEnumerationEnd MgmGroupEnumerationEnd 함수는 MgmGroupEnumerationStart에 대한 이전 호출에서 가져온 지정된 열거형 핸들을 해제합니다.
MgmGroupEnumerationGetNext MgmGroupEnumerationGetNext 함수는 다음 그룹 항목 집합을 검색합니다. 이 함수에서 반환되는 정보는 조인된 그룹 목록과 요청된 원본(있는 경우)입니다.
MgmGroupEnumerationStart MgmGroupEnumerationStart 함수는 나중에 조인된 그룹 목록을 가져오는 데 사용되는 열거형 핸들을 가져옵니다. 클라이언트가 핸들을 가져온 후에는 MgmGroupEnumerationGetNext 함수를 사용하여 그룹을 열거해야 합니다.
MgmRegisterMProtocol MgmRegisterMProtocol 함수는 클라이언트에서 멀티캐스트 그룹 관리자에 등록하는 데 사용됩니다.
MgmReleaseInterfaceOwnership MgmReleaseInterfaceOwnership 함수는 클라이언트에서 인터페이스의 소유권을 포기하는 데 사용됩니다. 이 함수가 호출되면 클라이언트를 대신하여 멀티캐스트 그룹 관리자가 유지 관리하는 모든 MFE와 지정된 인터페이스에 대해 유지 관리되는 모든 MFE가 삭제됩니다.
MgmSetMfe MgmSetMfe 함수는 MFE에 대한 업스트림 인접 항목을 변경합니다. MFE에는 수신하는 인터페이스와 멀티캐스트 데이터를 전달하는 인터페이스에 대한 정보가 포함되어 있습니다.
MgmTakeInterfaceOwnership MgmTakeInterfaceOwnership 함수는 인터페이스에서 사용하도록 설정된 경우 클라이언트(예: 라우팅 프로토콜)에 의해 호출됩니다.

콜백 함수

PMGM_CREATION_ALERT_CALLBACK PMGM_CREATION_ALERT_CALLBACK 콜백은 라우팅 프로토콜에 대한 호출입니다. 이 호출은 새 원본의 멀티캐스트 패킷을 전달해야 하는 라우팅 프로토콜이 소유한 인터페이스의 하위 집합을 결정합니다.
PMGM_DISABLE_IGMP_CALLBACK PMGM_DISABLE_IGMP_CALLBACK 콜백은 라우팅 프로토콜이 IGMP를 사용하도록 설정된 인터페이스의 소유권을 가져오거나 해제하고 있음을 알리기 위해 IGMP를 호출하는 것입니다.
PMGM_ENABLE_IGMP_CALLBACK PMGM_ENABLE_IGMP_CALLBACK 콜백은 IGMP를 호출하여 라우팅 프로토콜이 인터페이스 소유권을 가져오거나 해제했음을 알립니다.
PMGM_JOIN_ALERT_CALLBACK PMGM_JOIN_ALERT_CALLBACK 콜백은 다른 라우팅 프로토콜이 소유한 인터페이스에서 하나 이상의 그룹에 대해 새 수신기가 있다는 것을 프로토콜에 알리기 위해 라우팅 프로토콜을 호출하는 것입니다.
PMGM_LOCAL_JOIN_CALLBACK PMGM_LOCAL_JOIN_CALLBACK 콜백은 라우팅 프로토콜을 호출하여 IGMP가 현재 라우팅 프로토콜이 소유한 인터페이스에서 그룹에 대한 새 수신기를 감지했음을 프로토콜에 알립니다.
PMGM_LOCAL_LEAVE_CALLBACK PMGM_LOCAL_LEAVE_CALLBACK 콜백은 라우팅 프로토콜을 호출하여 IGMP가 라우팅 프로토콜이 현재 소유하고 있는 인터페이스의 그룹에 대한 수신기가 더 이상 없음을 감지했음을 라우팅 프로토콜에 알립니다.
PMGM_PRUNE_ALERT_CALLBACK PMGM_PRUNE_ALERT_CALLBACK 콜백은 다른 라우팅 프로토콜이 소유한 인터페이스에 수신기가 더 이상 존재하지 않는다는 것을 프로토콜에 알리기 위해 라우팅 프로토콜을 호출하는 것입니다.
PMGM_RPF_CALLBACK PMGM_RPF_CALLBACK 콜백은 지정된 패킷이 올바른 인터페이스에서 수신되었는지 확인하기 위해 라우팅 프로토콜을 호출하는 것입니다.
PMGM_WRONG_IF_CALLBACK

PMGM_WRONG_IF_CALLBACK 지정된 원본 및 잘못된 인터페이스의 지정된 그룹에 대해 패킷이 수신되었음을 프로토콜에 알리기 위해 라우팅 프로토콜을 호출하는 것입니다.

구조체

MGM_IF_ENTRY MGM_IF_ENTRY 구조체는 라우터 인터페이스를 설명합니다.
ROUTING_PROTOCOL_CONFIG ROUTING_PROTOCOL_CONFIG 구조는 프로토콜이 멀티캐스트 그룹 관리자에 등록될 때 멀티캐스트 그룹 관리자에게 전달되는 라우팅 프로토콜 구성 정보를 설명합니다.
SOURCE_GROUP_ENTRY SOURCE_GROUP_ENTRY 구조체는 그룹 열거형 함수 MgmGroupEnumerationGetNext에서 반환된 항목을 설명합니다.

열거형

MGM_ENUM_TYPES MGM_ENUM_TYPES 열거형에는 멀티캐스트 그룹 관리자가 사용하는 그룹 열거형 유형이 나열됩니다. 이 구조체는 MgmGroupEnumerationStart 함수에서 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mgmtapi.h 헤더

아티클2023. 08. 24.

이 헤더는 단순 네트워크 관리 프로토콜에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Simple Network Management Protocol

mgmtapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

SnmpMgrClose SnmpMgrClose 함수는 지정된 세션과 연결된 통신 소켓 및 데이터 구조를 닫습니다. 이 함수는 SNMP Management API의 요소입니다.
SnmpMgrCtl SnmpMgrCtl 함수는 SNMP 세션과 연결된 운영 매개 변수를 설정합니다. 이 함수는 SNMP Management API의 요소입니다.
SnmpMgrGetTrap SnmpMgrGetTrap 함수는 트랩 수신이 사용하도록 설정된 경우 호출자가 받지 못한 미해결 트랩 데이터를 반환합니다. 이 함수는 SNMP Management API의 요소입니다.
SnmpMgrGetTrapEx SnmpMgrGetTrapEx 함수는 트랩 수신이 사용하도록 설정된 경우 호출자가 받지 못한 미해결 트랩 데이터를 반환합니다.
SnmpMgrOidToStr SnmpMgrOidToStr 함수는 내부 개체 식별자 구조를 문자열 표현으로 변환합니다. 이 함수는 SNMP Management API의 요소입니다.
SnmpMgrOpen SnmpMgrOpen 함수는 통신 소켓 및 데이터 구조를 초기화하여 지정된 SNMP 에이전트와의 통신을 허용합니다. 이 함수는 SNMP Management API의 요소입니다.
SnmpMgrRequest

SnmpMgrRequest 함수는 지정된 에이전트를 사용하여 지정된 작업을 수행할 것을 요청합니다. 이 함수는 SNMP Management API의 요소입니다.
SnmpMgrStrToOid SnmpMgrStrToOid 함수는 개체 식별자의 문자열 형식을 내부 개체 식별자 구조로 변환합니다. 이 함수는 SNMP Management API의 요소입니다.
SnmpMgrTrapListen SnmpMgrTrapListen 함수는 SNMP 트랩 서비스에서 SNMP 트랩을 수신하는 SNMP 관리자 애플리케이션의 기능을 등록합니다. 이 함수는 SNMP Management API의 요소입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mi.h 헤더

아티클2023. 03. 09.

이 헤더는 MI(Windows Management Infrastructure)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- MI(Windows Management Infrastructure)

mi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

MI_Application_Close MI_Application_Initialize 호출을 통해 초기화된 관리 인프라 클라이언트 API를 초기화합니다.
MI_Application_InitializeV1 MI(관리 인프라) 클라이언트 API 호출을 수행할 수 있도록 애플리케이션을 초기화합니다.
MI_Application_NewClass MI_ClassDecl 구조체에서 MI_Class 만듭니다.
MI_Application_NewDeserializer serialize된 개체를 클래스 또는 인스턴스로 다시 변환하는 데 사용할 수 있는 역직렬 변환기 개체를 만듭니다.
MI_Application_NewDestinationOptions MI_Application_NewSession 함수와 함께 사용할 수 있는 MI_DestinationOptions 개체를 만듭니다.
MI_Application_NewHostedProvider 호스트된 공급자를 로컬 컴퓨터의 WMI 엔진에 등록합니다.
MI_Application_NewInstance 인스턴스가 필요한 다양한 MI 작업 API에 전달할 새 MI_Instance 개체를 만듭니다.
MI_Application_NewInstanceFromClass 클래스 개체를 기반으로 새 MI_Instance 개체를 만듭니다.

MI_Application_NewOperationOptions MI_Session 개체의 연산 함수와 함께 사용할 수 있는 MI_OperationOptions 개체를 만듭니다.
MI_Application_NewParameterSet 새 매개 변수 집합을 만듭니다.
MI_Application_NewSerializer 그런 다음 인스턴스와 클래스를 다양한 형식으로 serialize하는 데 사용할 수 있는 serializer 개체를 검색합니다.
MI_Application_NewSession 단일 대상에 대한 작업 집합에 대한 연결을 공유하는 데 사용되는 세션을 만듭니다.
MI_Application_NewSubscriptionDeliveryOptions 특정 프로토콜에 대한 구독 작업을 수행하는 데 필요한 구성을 나타내는 MI_SubscriptionDeliveryOptions 개체를 만듭니다.
MI_Class_Clone MI_Class 개체를 복제합니다.
MI_Class_Delete MI_Class 개체를 삭제합니다.
MI_Class_GetClassName 지정된 클래스의 클래스 이름을 가져옵니다.
MI_Class_GetClassQualifierSet 지정된 클래스 개체와 연결된 한정자 집합을 가져옵니다.
MI_Class_GetElement 클래스에서 지정된 명명된 요소의 모든 세부 정보를 가져옵니다.
MI_Class_GetElementAt 요소 인덱스 기반 클래스 요소의 세부 정보를 가져옵니다.
MI_Class_GetElementCount 클래스의 요소 수를 가져옵니다.

MI_Class_GetMethod 메서드 이름을 기반으로 메서드의 세부 정보를 가져옵니다.
MI_Class_GetMethodAt 메서드 인덱스 기반 메서드의 세부 정보를 가져옵니다.
MI_Class_GetMethodCount 클래스의 메서드 수를 가져옵니다.
MI_Class_GetNameSpace 지정된 클래스의 네임스페이스 이름을 가져옵니다.
MI_Class_GetParentClass 지정된 클래스의 부모 클래스를 가져옵니다.
MI_Class_GetParentClassName 지정된 클래스의 부모 클래스 이름을 가져옵니다.
MI_Class_GetServerName 지정된 클래스에서 서버의 이름을 가져옵니다.
MI_Context_Canceled 작업이 취소되었는지 여부를 확인합니다. 이 함수는 예약되어 있습니다. 대신 MI_Context_RegisterCancel 함수를 사용합니다.
MI_Context_ConstructInstance 스택 또는 구조체의 멤버로 MI 클래스 인스턴스를 초기화합니다.
MI_Context_ConstructParameters 공급자는 이 함수를 호출하여 매개 변수의 인스턴스를 초기화합니다.
MI_Context_GetCustomOption 클라이언트에서 설정한 옵션을 검색합니다.
MI_Context_GetCustomOptionAt 클라이언트에서 설정한 특정 인덱스에서 옵션을 검색합니다.

MI_Context_GetCustomOptionCount 공급자가 사용할 수 있는 사용자 지정 옵션 수를 가져옵니다.
MI_Context_GetLocale 클라이언트가 작업에 대해 지정한 요청된 로캘 정보를 검색합니다.
MI_Context_GetLocalSession 공급자가 공급자를 호스트하는 로컬 서버에 대해 CIM 작업을 수행할 수 있는 로컬 세션(MI_Session)을 가져옵니다.
MI_Context_GetNumberOption 작업 이름에 따라 클라이언트가 설정하는 숫자 옵션을 가져옵니다.
MI_Context_GetStringOption 작업 이름에 따라 클라이언트가 설정하는 문자열 옵션을 가져옵니다.
MI_Context_NewDynamicInstance 클래스의 새 동적 인스턴스(클래스 선언이 없는 약한 형식의 인스턴스)를 만듭니다.
MI_Context_NewInstance 클래스 선언이 지정된 클래스의 새 인스턴스를 만듭니다.
MI_Context_NewParameters 메서드 선언이 지정된 경우 메서드의 새 인스턴스를 만듭니다.
MI_Context_PostCimError 요청에 대한 응답으로 반환 코드와 오류 메시지(CIM_Error 개체 형식)를 서버에 게시합니다.
MI_Context_PostError 공급자는 요청에 대한 응답으로 클라이언트에 반환 코드를 게시하기 위해 이 함수를 호출합니다.
MI_Context_PostIndication 구독 작업 요청에 대한 응답으로 표시 결과를 서버에 게시합니다.
MI_Context_PostInstance 요청에 대한 응답으로 인스턴스를 클라이언트(서버를 통해)에 다시 게시합니다.

MI_Context_PostResult 요청에 대한 응답으로 최종 종료 결과 코드를 클라이언트(서버를 통해)에 다시 게시합니다.
MI_Context_PromptUser 작업을 계속할지 아니면 취소할지를 쿼리하는 프롬프트 메시지를 클라이언트에 보냅니다.
MI_Context_RefuseUnload 공급자 인프라에 공급자를 언로드하지 말라고 지시합니다.
MI_Context_RegisterCancel 작업이 취소될 때 호출되는 콜백을 등록합니다.
MI_Context_RequestUnload 모듈 또는 공급자를 언로드하도록 요청합니다.
MI_Context_SetStringOption 컨텍스트별 옵션을 설정합니다.
MI_Context_ShouldContinue 클라이언트를 쿼리하여 작업을 계속해야 하는지 확인합니다. (MI_Context_ShouldContinue)
MI_Context_ShouldProcess 클라이언트를 쿼리하여 작업을 계속해야 하는지 확인합니다. (MI_Context_ShouldProcess)
MI_Context_WriteCimError CIM(정보 제공) 오류 인스턴스를 클라이언트에 보냅니다.
MI_Context_WriteDebug 클라이언트에 디버그 메시지를 보냅니다.
MI_Context_WriteError 오류 코드 및 오류 메시지를 클라이언트에 보냅니다.
MI_Context_WriteMessage 클라이언트에 작업 메시지를 보냅니다.

MI_Context_WriteProgress 클라이언트에 진행률 메시지를 보냅니다.
MI_Context_WriteStreamParameter 메서드 호출을 위해 스트리밍된 매개 변수 데이터를 클라이언트에 보냅니다.
MI_Context_WriteVerbose 클라이언트에 자세한 메시지를 씁니다.
MI_Context_WriteWarning 클라이언트에 경고 메시지를 씁니다.
MI_Deserializer_Class_GetClassName serialize된 클래스 버퍼에서 클래스 이름을 가져옵니다.
MI_Deserializer_Class_GetParentClassName serialize된 클래스 버퍼에서 부모 클래스 이름을 가져옵니다.
MI_Deserializer_Close 역직렬 변환기 개체를 닫고 역직렬 변환기 내에 있는 연결된 메모리를 삭제합니다.
MI_Deserializer_DeserializeClass 직렬화된 버퍼를 MI_Class 개체로 역직렬화합니다.
MI_Deserializer_DeserializeInstance 직렬화된 버퍼를 MI_Instance 개체로 역직렬화합니다.
MI_Deserializer_Instance_GetClassName serialize된 인스턴스와 연결된 클래스 이름을 가져옵니다.
MI_DestinationOptions_AddDestinationCredentials 대상과 통신하기 위한 자격 증명을 설정합니다.
MI_DestinationOptions_AddProxyCredentials 프록시에 대해 인증할 자격 증명을 추가합니다.

MI_DestinationOptions_Clone MI_DestinationOptions 구조체의 복사본을 만듭니다.
MI_DestinationOptions_Delete MI_Application_NewDestinationOptions 또는 MI_DestinationOptions_Clone 함수를 사용하여 만든 대상 옵션 구조를 삭제합니다.
MI_DestinationOptions_GetCertCACheck 서버 인증서 CA 확인 값을 가져옵니다.
MI_DestinationOptions_GetCertCNCheck 서버 인증서 CN 확인 값을 가져옵니다.
MI_DestinationOptions_GetCertRevocationCheck 서버 인증서의 해지 확인 값을 가져옵니다.
MI_DestinationOptions_GetCredentialsAt 지정된 인덱스에서 자격 증명을 가져옵니다.
MI_DestinationOptions_GetCredentialsCount 이전에 추가된 자격 증명의 수를 가져옵니다. (MI_DestinationOptions_GetCredentialsCount)
MI_DestinationOptions_GetCredentialsPasswordAt 지정된 인덱스 기반의 자격 증명 암호를 가져옵니다.
MI_DestinationOptions_GetDataLocale 사용자가 설정한 데이터 로캘(UI 로캘과 반대)을 가져옵니다.
MI_DestinationOptions_GetDestinationPort 전송에 대한 기본 포트를 가져옵니다.
MI_DestinationOptions_GetEncodePortInSPN 포트의 서비스 주체 이름 인코딩 값을 가져옵니다.
MI_DestinationOptions_GetHttpUrlPrefix HTTP URL 접두사 가져오기

MI_DestinationOptions_GetImpersonationType 가장 형식을 가져옵니다.
MI_DestinationOptions_GetMaxEnvelopeSize 서버로 전송되거나 서버에서 클라이언트가 받은 패킷의 최대 크기를 가져옵니다.
MI_DestinationOptions_GetNumber 이전에 추가한 사용자 지정 번호 옵션을 가져옵니다. (MI_DestinationOptions_GetNumber)
MI_DestinationOptions_GetOption 옵션 이름을 기반으로 이전에 추가된 옵션 값을 가져옵니다. (MI_DestinationOptions_GetOption)
MI_DestinationOptions_GetOptionAt 지정된 인덱스 기반의 이전에 추가된 옵션 값을 가져옵니다. (MI_DestinationOptions_GetOptionAt)
MI_DestinationOptions_GetOptionCount 이전에 추가한 옵션 수를 가져옵니다. (MI_DestinationOptions_GetOptionCount)
MI_DestinationOptions_GetPacketEncoding 이전에 설정한 패킷 인코딩 설정을 가져옵니다.
MI_DestinationOptions_GetPacketIntegrity 패킷 무결성 설정을 가져옵니다.
MI_DestinationOptions_GetPacketPrivacy 패킷 개인 정보(암호화) 설정을 가져옵니다.
MI_DestinationOptions_GetProxyType 사용자가 설정한 프록시 유형을 가져옵니다.
MI_DestinationOptions_GetString 이전에 추가한 사용자 지정 문자열 옵션을 가져옵니다.
MI_DestinationOptions_GetTimeout 기본 옵션 시간 제한 값을 가져옵니다.

MI_DestinationOptions_GetTransport 클라이언트가 추가한 전송 설정을 가져옵니다.
MI_DestinationOptions_GetUILocale 사용자가 설정한 사용자 인터페이스 로캘을 가져옵니다.
MI_DestinationOptions_SetCertCACheck SSL 전송에 대한 CA 인증서 검사를 사용하거나 사용하지 않도록 설정합니다.
MI_DestinationOptions_SetCertCNCheck SSL 전송이 사용될 때 인증서 CN 검사를 사용하거나 사용하지 않도록 설정합니다.
MI_DestinationOptions_SetCertRevocationCheck SSL을 통해 통신할 때 인증서 해지를 사용하거나 사용하지 않도록 설정합니다.
MI_DestinationOptions_SetDataLocale 작업에 사용할 기본 데이터 로캘을 설정합니다.
MI_DestinationOptions_SetDestinationPort 대상과 통신하는 데 사용할 포트를 설정합니다.
MI_DestinationOptions_SetEncodePortInSPN 원격 컴퓨터에 대한 연결을 설정할 때 서비스 주체 이름에서 포트 번호의 인코딩을 사용하거나 사용하지 않도록 설정합니다.
MI_DestinationOptions_SetHttpUrlPrefix HTTP 및 HTTPS를 통해 이동하는 전송에 대한 기본 HTTP URL 접두사 설정
MI_DestinationOptions_SetImpersonationType 가장 유형을 설정합니다.
MI_DestinationOptions_SetMaxEnvelopeSize 전송의 최대 패킷 크기를 설정합니다.
MI_DestinationOptions_SetNumber 사용자 지정 숫자 옵션 값을 설정합니다.

MI_DestinationOptions_SetPacketEncoding 특정 프로토콜 핸들에 대한 인코딩 메커니즘을 설정합니다.
MI_DestinationOptions_SetPacketIntegrity 프로토콜 연결의 패킷 무결성(서명)을 사용하거나 사용하지 않도록 설정합니다.
MI_DestinationOptions_SetPacketPrivacy 패킷 개인 정보 보호(암호화)를 사용하거나 사용하지 않도록 설정합니다.
MI_DestinationOptions_SetProxyType 프록시를 통해 대상과 통신할 때 사용할 프록시 설정 유형을 설정합니다.
MI_DestinationOptions_SetString 사용자 지정 문자열 옵션을 설정합니다. (MI_DestinationOptions_SetString)
MI_DestinationOptions_SetTimeout 기본 옵션 시간 제한 값을 설정합니다.
MI_DestinationOptions_SetTransport 대상 컴퓨터와 통신하는 데 사용할 전송을 설정합니다.
MI_DestinationOptions_SetUILocale 작업에 대한 기본 UI 로캘을 설정합니다.
MI_Filter_Evaluate 공급자는 지정된 필터에 대해 인스턴스를 평가하기 위해 이 함수를 호출합니다.
MI_Filter_GetExpression 필터 언어 및 식을 가져옵니다.
MI_HostedProvider_Close MI_Application_NewHostedProvider 반환된 호스트된 공급자 핸들을 닫습니다.
MI_HostedProvider_GetApplication 호스트된 공급자 핸들이 만들어진 최상위 애플리케이션 핸들을 가져옵니다.

MI_Instance_AddElement 동적 인스턴스에 새 속성을 추가합니다(런타임에 스키마를 확장할 수 있는 동적 인스턴스에서만 지원됨).
MI_Instance_ClearElement 명명된 요소(CIM 속성)의 값을 지우고 NULL로 설정합니다.
MI_Instance_ClearElementAt 지정된 인덱스에 있는 요소(CIM 속성)의 값을 지우고 NULL로 설정합니다.
MI_Instance_Clone 힙에 지정된 인스턴스의 복사본을 만듭니다.
MI_Instance_Delete 힙에서 만들거나 다른 인스턴스에서 복제된 인스턴스를 삭제합니다.
MI_Instance_Destruct 스택 또는 구조체의 멤버로 만든 인스턴스를 삭제합니다.
MI_Instance_GetClass 인스턴스와 연결된 MI_Class 가져옵니다.
MI_Instance_GetClassName 지정된 인스턴스의 클래스 이름을 가져옵니다.
MI_Instance_GetElement 명명된 요소(CIM 속성)의 값을 가져옵니다.
MI_Instance_GetElementAt 지정된 인덱스에서 요소(CIM 속성)의 값을 가져옵니다.
MI_Instance_GetElementCount 인스턴스의 요소 수를 가져옵니다.
MI_Instance_GetNameSpace 지정된 인스턴스의 네임스페이스 이름을 가져옵니다.

MI_Instance_GetServerName 지정된 인스턴스에서 서버 이름을 가져옵니다.
MI_Instance_IsA 인스턴스 자체가 classDecl에서 제공하는 클래스의 인스턴스인지 여부를 확인합니다.
MI_Instance_Normalize MI_Instance_ExFT 구조를 구문 분석한 다음 MI_InstanceFT 함수 테이블을 검색합니다.
MI_Instance_SetElement 지정된 인스턴스에서 지정된 이름을 가진 요소의 값을 설정합니다.
MI_Instance_SetElementAt 인스턴스의 지정된 인덱스에서 요소의 값을 설정합니다.
MI_Instance_SetNameSpace 지정된 인스턴스의 네임스페이스 이름을 설정합니다.
MI_Instance_SetServerName 지정된 인스턴스의 서버 이름을 설정합니다.
MI_Operation_Cancel 실행 중인 작업을 취소합니다.
MI_Operation_Close 작업 핸들을 닫습니다.
MI_Operation_GetClass 클래스 작업에 대한 동기 결과를 가져옵니다.
MI_Operation_GetIndication 구독에서 동기 결과를 가져옵니다.
MI_Operation_GetInstance 인스턴스 작업에 대한 동기 결과를 가져옵니다.

MI_Operation_GetSession 작업과 연결된 세션을 가져옵니다.
MI_OperationOptions_Clone MI_OperationOptions 구조체의 복사본을 만듭니다.
MI_OperationOptions_Delete 옵션 집합 및 관련 메모리를 삭제합니다.
MI_OperationOptions_DisableChannel MI_Context_WriteMessage 사용하여 지정된 채널에 대한 로깅을 사용하지 않도록 설정합니다.
MI_OperationOptions_EnableChannel MI_Context_WriteMessage 사용하여 지정된 채널에 대한 로깅을 사용하도록 설정합니다.
MI_OperationOptions_GetEnabledChannels 이전에 사용하도록 설정된 채널의 목록을 가져옵니다.
MI_OperationOptions_GetNumber 이전에 추가한 사용자 지정 번호 옵션을 가져옵니다. (MI_OperationOptions_GetNumber)
MI_OperationOptions_GetOption 옵션 이름에 따라 이전에 추가된 옵션 값을 가져옵니다. (MI_OperationOptions_GetOption)
MI_OperationOptions_GetOptionAt 지정된 인덱스 기반 이전에 추가된 옵션 값을 가져옵니다. (MI_OperationOptions_GetOptionAt)
MI_OperationOptions_GetOptionCount 이전에 추가한 옵션 수를 가져옵니다. (MI_OperationOptions_GetOptionCount)
MI_OperationOptions_GetPromptUserMode 공급자의 MI_Context_PromptUser 호출에 응답하는 방법을 서버에 알려주는 값을 가져옵니다. (MI_OperationOptions_GetPromptUserMode)
MI_OperationOptions_GetPromptUserRegularMode 공급자의 MI_Context_PromptUser 호출에 응답하는 방법을 서버에 알려주는 값을 가져옵니다. (MI_OperationOptions_GetPromptUserRegularMode)

MI_OperationOptions_GetProviderArchitecture 작업에 대한 공급자 아키텍처를 가져옵니다.
MI_OperationOptions_GetResourceUri 작업에 사용되는 리소스 URI를 가져옵니다.
MI_OperationOptions_GetResourceUriPrefix 작업에 사용되는 리소스 URI 접두사입니다.
MI_OperationOptions_GetString 사용자 지정 문자열 옵션을 가져옵니다.
MI_OperationOptions_GetTimeout 작업 시간 제한 값을 가져옵니다.
MI_OperationOptions_GetUseMachineID 작업 요청에서 컴퓨터 식별 정보를 사용할지 여부를 나타내는 값을 가져옵니다.
MI_OperationOptions_GetWriteErrorMode 오류 보고 모드를 설정합니다. (MI_OperationOptions_GetWriteErrorMode)
MI_OperationOptions_SetCustomOption 작업에 대한 사용자 지정 옵션을 설정합니다.
MI_OperationOptions_SetNumber 사용자 지정 숫자 옵션 값을 설정합니다.
MI_OperationOptions_SetPromptUserMode MI_Context_PromptUser 함수에 대한 공급자의 호출에 응답하는 방법을 서버에 알려주는 값을 설정합니다. (MI_OperationOptions_SetPromptUserMode)
MI_OperationOptions_SetPromptUserRegularMode MI_Context_PromptUser 함수에 대한 공급자의 호출에 응답하는 방법을 서버에 알려주는 값을 설정합니다. (MI_OperationOptions_SetPromptUserRegularMode)
MI_OperationOptions_SetProviderArchitecture 작업에 대한 공급자 아키텍처를 설정합니다.

MI_OperationOptions_SetResourceUri 작업에 사용할 리소스 URI를 설정합니다.
MI_OperationOptions_SetResourceUriPrefix 작업에 사용할 리소스 URI 접두사 설정
MI_OperationOptions_SetString 사용자 지정 문자열 옵션을 설정합니다. (MI_OperationOptions_SetString)
MI_OperationOptions_SetTimeout 특정 작업에 대한 작업 시간 제한을 설정합니다.
MI_OperationOptions_SetUseMachineID 작업 요청에서 컴퓨터 ID 정보 전송을 사용하거나 사용하지 않도록 설정합니다.
MI_OperationOptions_SetWriteErrorMode 오류 보고 모드를 설정합니다. (MI_OperationOptions_SetWriteErrorMode)
MI_ParameterSet_GetMethodReturnType 지정된 매개 변수 집합에 대한 메서드 반환 형식 및 한정자 집합을 가져옵니다.
MI_ParameterSet_GetParameter 매개 변수 이름을 기반으로 메서드의 매개 변수 정보를 가져옵니다.
MI_ParameterSet_GetParameterAt 지정된 인덱스에서 메서드의 매개 변수 정보를 가져옵니다.
MI_ParameterSet_GetParameterCount 메서드의 매개 변수 집합에 있는 매개 변수 수를 가져옵니다.
MI_PropertySet_AddElement 속성 목록에 이름을 추가합니다.
MI_PropertySet_Clear 속성 목록에서 모든 이름을 제거합니다. 그 후 개수는 0입니다. 이렇게 하면 속성 목록을 다시 사용할 수 있습니다(소멸 및 재구성할 필요 없이).

MI_PropertySet_Clone 힙에 지정된 속성 집합의 복사본을 만듭니다.
MI_PropertySet_ContainsElement 속성 목록에 지정된 속성 이름이 포함되어 있는지 여부를 확인합니다.
MI_PropertySet_Delete 힙에서 생성된 지정된 속성 목록을 삭제합니다.
MI_PropertySet_Destruct 스택에서 생성된 지정된 속성 목록을 삭제합니다.
MI_PropertySet_GetElementAt 지정된 인덱스에서 속성 집합의 요소를 가져옵니다.
MI_PropertySet_GetElementCount 지정된 속성 집합의 요소 수를 가져옵니다.
MI_QualifierSet_GetQualifier 지정된 한정자 이름을 기준으로 한정자 정보를 가져옵니다.
MI_QualifierSet_GetQualifierAt 지정된 인덱스에서 한정자를 가져옵니다.
MI_QualifierSet_GetQualifierCount 한정자 집합의 한정자 수를 가져옵니다.
MI_Serializer_Close serializer 개체를 닫고 연결된 내부 메모리를 해제합니다.
MI_Serializer_SerializeClass serializer를 만들 때 지정된 형식으로 버퍼로 MI_Class 직렬화합니다. 클래스와 모든 부모 클래스가 직렬화되는지 아니면 자식 클래스만 제어하도록 플래그에 옵션을 전달할 수 있습니다.
MI_Serializer_SerializeInstance serializer를 만들 때 지정된 형식으로 버퍼에 MI_Instance 직렬화합니다. 클래스가 버퍼와 인스턴스로 직렬화되는지 제어하기 위해 플래그에 옵션을 전달할 수 있습니다.

MI_Server_GetSystemName 서버의 시스템 이름을 가져옵니다.
MI_Server_GetVersion 공급자를 생성할 때 사용되는 MI_VERSION 매크로의 값을 가져옵니다.
MI_Session_AssociatorInstances 특정 키 인스턴스와 연결된 인스턴스를 찾습니다.
MI_Session_Close 세션을 닫고 연결된 모든 메모리를 해제합니다.
MI_Session_CreateInstance 세션이 나타내는 서버에 인스턴스를 만듭니다.
MI_Session_DeleteInstance 세션이 나타내는 서버의 인스턴스를 삭제합니다.
MI_Session_EnumerateClasses 지정된 세션의 클래스를 열거합니다.
MI_Session_EnumerateInstances 클래스와 연결된 모든 인스턴스(세션이 나타내는 서버)를 열거합니다.
MI_Session_GetApplication 지정된 세션을 만드는 데 사용된 애플리케이션 핸들을 가져옵니다.
MI_Session_GetClass 특정 클래스 이름을 기반으로 MI_Class 선언을 가져옵니다.
MI_Session_GetInstance 세션이 나타내는 서버에서 지정된 인스턴스를 가져옵니다.
MI_Session_Invoke 공급자에서 메서드를 호출합니다.

MI_Session_ModifyInstance 세션이 나타내는 서버의 기존 인스턴스를 업데이트.
MI_Session_QueryInstances 쿼리 식을 기반으로 인스턴스 집합을 쿼리합니다.
MI_Session_ReferenceInstances 지정된 키 인스턴스를 참조하는 연결 개체를 찾습니다.
MI_Session_Subscribe 세션이 나타내는 서버의 표시를 구독합니다.
MI_Session_TestConnection 세션이 나타내는 서버와 통신하여 연결을 테스트하여 응답 여부를 확인합니다.
MI_SubscriptionDeliveryOptions_AddDeliveryCredentials 푸시 표시 결과를 제공하기 위해 클라이언트에 다시 연결할 때 사용할 배달 자격 증명에 대한 구독 옵션을 설정합니다.
MI_SubscriptionDeliveryOptions_Clone MI_SubscriptionDeliveryOptions 구조체의 복사본을 만듭니다.
MI_SubscriptionDeliveryOptions_Delete 지정된 구독 배달 옵션 구조를 삭제합니다.
MI_SubscriptionDeliveryOptions_GetBookmark 이전에 설정된 구독 책갈피를 가져옵니다.
MI_SubscriptionDeliveryOptions_GetCredentialsAt 지정된 인덱스 기반 이전에 추가된 자격 증명을 가져옵니다.
MI_SubscriptionDeliveryOptions_GetCredentialsCount 이전에 추가한 자격 증명 수를 가져옵니다. (MI_SubscriptionDeliveryOptions_GetCredentialsCount)
MI_SubscriptionDeliveryOptions_GetCredentialsPasswordAt 지정된 인덱스 기반 이전에 추가된 자격 증명 암호를 가져옵니다.

MI_SubscriptionDeliveryOptions_GetDateTime 이전에 설정된 날짜/시간 옵션을 가져옵니다.
MI_SubscriptionDeliveryOptions_GetDeliveryDestination 이전에 설정된 구독 배달 대상을 가져옵니다.
MI_SubscriptionDeliveryOptions_GetDeliveryPortNumber 이전에 설정된 배달 포트 번호를 가져옵니다.
MI_SubscriptionDeliveryOptions_GetDeliveryRetryAttempts 배달 재시도 횟수를 가져옵니다.
MI_SubscriptionDeliveryOptions_GetDeliveryRetryInterval 배달 재시도 간격을 가져오고 배달을 다시 시도하기 전에 대기할 시간을 가져옵니다.
MI_SubscriptionDeliveryOptions_GetExpirationTime 배달 만료 값(타임스탬프 또는 간격으로 표현할 수 있는)을 가져옵니다.
MI_SubscriptionDeliveryOptions_GetHeartbeatInterval 배달 하트비트 간격을 가져옵니다.
MI_SubscriptionDeliveryOptions_GetInterval 지정된 옵션의 배달 간격을 가져옵니다.
MI_SubscriptionDeliveryOptions_GetMaximumLatency 클라이언트에 전달하기 전에 서버에서 결과를 보유할 최대 시간을 가져옵니다.
MI_SubscriptionDeliveryOptions_GetNumber 명명된 숫자 옵션의 값을 가져옵니다.
MI_SubscriptionDeliveryOptions_GetOption 명명된 옵션의 값을 가져옵니다.
MI_SubscriptionDeliveryOptions_GetOptionAt 지정된 인덱스에서 옵션을 가져옵니다.

MI_SubscriptionDeliveryOptions_GetOptionCount 이전에 설정된 옵션의 수를 가져옵니다.
MI_SubscriptionDeliveryOptions_GetString 명명된 문자열 옵션의 값을 가져옵니다.
MI_SubscriptionDeliveryOptions_SetBookmark 구독 표시 배달에 대한 책갈피를 설정합니다.
MI_SubscriptionDeliveryOptions_SetDateTime 명명된 DateTime 옵션의 값을 설정합니다.
MI_SubscriptionDeliveryOptions_SetDeliveryDestination 표시가 전달될 대상 엔드포인트를 설정합니다.
MI_SubscriptionDeliveryOptions_SetDeliveryPortNumber 구독 배달 포트 번호를 설정합니다.
MI_SubscriptionDeliveryOptions_SetDeliveryRetryAttempts 푸시 배달 구독이 결과를 배달하려고 시도하는 횟수를 설정합니다.
MI_SubscriptionDeliveryOptions_SetDeliveryRetryInterval 푸시 배달을 위한 구독에 대한 배달 재시도 간격을 설정합니다.
MI_SubscriptionDeliveryOptions_SetExpirationTime 구독 만료 시간(구독이 종료되는 경우)을 설정합니다.
MI_SubscriptionDeliveryOptions_SetHeartbeatInterval 하트비트 간격을 설정합니다.
MI_SubscriptionDeliveryOptions_SetInterval 명명된 간격 옵션의 값을 설정합니다.
MI_SubscriptionDeliveryOptions_SetMaximumLatency 클라이언트에 전달하기 전에 서버에서 결과를 보유할 최대 시간을 설정합니다.

MI_SubscriptionDeliveryOptions_SetNumber 전용 함수에서 다루지 않는 명명된 숫자 옵션의 값을 설정합니다.
MI_SubscriptionDeliveryOptions_SetString 전용 함수에서 다루지 않는 명명된 문자열 옵션의 값을 설정합니다.
MI_Utilities_CimErrorFromErrorCode 운영 체제별 오류 코드를 CIM 오류 인스턴스에 매핑합니다.
MI_Utilities_MapErrorToMiErrorCategory 운영 체제별 오류 코드를 오류 범주에 매핑합니다.

콜백 함수

MI_Deserializer_ClassObjectNeeded 역직렬화 중에 요청된 클래스 개체를 제공하는 데 사용됩니다.

구조체

MI_Application 초기화된 인프라를 나타냅니다.
MI_ApplicationFT MI_Application 구조에 사용되는 지원 구조체입니다. "MI_Application_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_Array 배열을 나타내는 일반화된 형식입니다. 데이터 요소 형식을 제외한 모든 배열의 크기가 같으므로 일반화할 수 있습니다.

MI_ArrayField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ArrayField)
MI_BooleanA MI_Boolean 형식의 배열을 나타냅니다.
MI_BooleanAField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_BooleanAField)
MI_BooleanField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_BooleanField)
MI_Char16A MI_Char16 형식의 배열을 나타냅니다. (MI_Char16A)
MI_Char16AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Char16AField)
MI_Char16Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Char16Field)
MI_Class 인스턴스의 스키마를 나타냅니다.
MI_ClassDecl 이 구조체는 클래스 선언을 간략하게 설명합니다. 클래스 이름 및 계층, 속성, 한정자 및 메서드를 포함합니다.
MI_ClassFT MI_Class 구조에 사용되는 지원 구조체입니다. "MI_Class_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_ClientFT_V1 클라이언트 함수 테이블.
MI_ConstBooleanA MI_ConstBoolean 형식의 배열을 나타냅니다.

MI_ConstBooleanAField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstBooleanAField)
MI_ConstBooleanField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstBooleanField)
MI_ConstChar16A MI_Char16 형식의 배열을 나타냅니다. (MI_ConstChar16A)
MI_ConstChar16AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstChar16AField)
MI_ConstChar16Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstChar16Field)
MI_ConstDatetimeA MI_Datatime 형식의 배열을 나타냅니다.
MI_ConstDatetimeAField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstDatetimeAField)
MI_ConstDatetimeField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstDatetimeField)
MI_ConstInstanceA MI_Instance 형식의 배열을 나타냅니다. (MI_ConstInstanceA)
MI_ConstInstanceAField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstInstanceAField)
MI_ConstInstanceField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstInstanceField)
MI_ConstReal32A MI_Real32 형식의 배열을 나타냅니다. (MI_ConstReal32A)

MI_ConstReal32AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstReal32AField)
MI_ConstReal32Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstReal32Field)
MI_ConstReal64A MI_Real64 형식의 배열을 나타냅니다. (MI_ConstReal64A)
MI_ConstReal64AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstReal64AField)
MI_ConstReal64Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstReal64Field)
MI_ConstReferenceA MI_Instance 형식의 배열을 나타냅니다. (MI_ConstReferenceA)
MI_ConstReferenceAField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstReferenceAField)
MI_ConstReferenceField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstReferenceField)
MI_ConstSint16A MI_Sint16 형식의 배열을 나타냅니다. (MI_ConstSint16A)
MI_ConstSint16AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstSint16AField)
MI_ConstSint16Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstSint16Field)
MI_ConstSint32A MI_Sint32 형식의 배열을 나타냅니다. (MI_ConstSint32A)

MI_ConstSint32AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstSint32AField)
MI_ConstSint32Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstSint32Field)
MI_ConstSint64A MI_Sint64 형식의 배열을 나타냅니다. (MI_ConstSint64A)
MI_ConstSint64AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstSint64AField)
MI_ConstSint64Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstSint64Field)
MI_ConstSint8A MI_Sint8 형식의 배열을 나타냅니다. (MI_ConstSint8A)
MI_ConstSint8AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstSint8AField)
MI_ConstSint8Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstSint8Field)
MI_ConstStringA MI_Char 형식의 배열을 나타냅니다.
MI_ConstStringAField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstStringAField)
MI_ConstStringField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstStringField)
MI_ConstUint16A MI_Uint16A 형식의 배열을 나타냅니다.

MI_ConstUint16AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstUint16AField)
MI_ConstUint16Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstUint16Field)
MI_ConstUint32A MI_Uint32 형식의 배열을 나타냅니다. (MI_ConstUint32A)
MI_ConstUint32AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstUint32AField)
MI_ConstUint32Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstUint32Field)
MI_ConstUint64A MI_Uint64 형식의 배열을 나타냅니다. (MI_ConstUint64A)
MI_ConstUint64AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstUint64AField)
MI_ConstUint64Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstUint64Field)
MI_ConstUint8A MI_Uint8 형식의 배열을 나타냅니다. (MI_ConstUint8A)
MI_ConstUint8AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstUint8AField)
MI_ConstUint8Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ConstUint8Field)
MI_Context 공급자가 수행해야 하는 작업에 대한 컨텍스트를 보유합니다.

MI_ContextFT MI_Context 구조체에 사용되는 지원 구조체입니다. 이름 접두사 "MI_Context_"와 함께 함수를 사용하여 이러한 구조를 조작합니다.
MI_Datetime MI_Timestamp 및 MI_Interval 공용 구조체를 나타냅니다.
MI_DatetimeA MI_Datetime 형식의 배열을 나타냅니다.
MI_DatetimeAField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_DatetimeAField)
MI_DatetimeField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_DatetimeField)
MI_Deserializer MI_Application_NewDeserializer 만든 역직렬화 개체입니다. 개체 자체는 내부적으로 사용되므로 수동으로 사용하거나 변경하면 안 됩니다.
MI_DeserializerFT MI_ClientFT_V1 구조에 사용되는 지원 구조체입니다. 이름 접두사 "MI_Deserializer_"와 함께 함수를 사용하여 이러한 구조를 조작합니다.
MI_DestinationOptions 대상 옵션 집합을 나타냅니다. 대상 옵션은 작업이 서버와 통신하는 방식을 정의하는 구성 집합입니다.
MI_DestinationOptionsFT MI_DestinationOptions 구조체에 사용되는 지원 구조체입니다. "MI_DestinationOptions_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_FeatureDecl MI_PropertyDeclMI_ParameterDecland MI_MethodDecl 구조체에 공통적인 속성을 포함합니다.
MI_Filter MI_FilterFT 함수 테이블에 대한 참조를 포함합니다.

MI_FilterFT MI_Filter 구조체에 사용되는 지원 구조체입니다. "MI_Filter_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_HostedProvider 클라이언트 애플리케이션에서 공급자의 호스팅을 나타냅니다.
MI_HostedProviderFT MI_HostedProvider 구조체에 사용되는 지원 구조체입니다. "MI_HostedProvider_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_Instance 이 구조는 CIM 인스턴스를 나타냅니다. 이 개체는 직접 액세스해서는 안 됩니다. 대신 MI_Instance_* 함수를 사용해야 합니다.
MI_InstanceA MI_Instance 구조체의 배열을 나타냅니다.
MI_InstanceAField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_InstanceAField)
MI_InstanceExFT MI_InstanceFT 구조를 확장합니다.
MI_InstanceField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_InstanceField)
MI_InstanceFT MI_Instance 구조체에 사용되는 지원 구조체입니다. 이름 접두사 MI_Instance_ 함수를 사용하여 이러한 구조를 조작합니다.
MI_Interval MI_Interval 시간 간격을 나타냅니다.
MI_MethodDecl CIM 메서드를 나타냅니다.

MI_Module 공급자가 생성한 이 개체는 공급자 관리자가 이 모듈 내에서 공급자를 관리하는 데 필요한 모든 데이터를 포함합니다.
MI_ObjectDecl MI_ClassDecl 및 MI_PropertyDecl 구조체에 공통적인 속성을 포함합니다.
MI_Operation 단일 작업 실행을 나타냅니다. 이 개체는 작업에 대한 작업을 수행하기 위한 내부 함수 테이블을 보유합니다.
MI_OperationCallbacks 작업을 수행하기 위한 모든 콜백 함수 포인터를 보유하는 구조체입니다.
MI_OperationFT MI_Operation 구조체에 사용되는 지원 구조체입니다. "MI_Operation_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_OperationOptions 작업 옵션 집합을 나타냅니다.
MI_OperationOptionsFT MI_OperationOptions 구조체에 사용되는 지원 구조체입니다. "MI_OperationOptions_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_ParameterDecl CIM 메서드 매개 변수를 나타냅니다.
MI_ParameterSet 클래스 정의의 메서드 매개 변수를 보유합니다.
MI_ParameterSetFT MI_ParameterSet 구조체에 사용되는 지원 구조체입니다. 이름 접두사 MI_ParameterSet_ 함수를 사용하여 이러한 구조를 조작합니다.
MI_PropertyDecl 클래스 선언의 클래스 속성(요소)을 나타냅니다.

MI_PropertySet 속성 이름 집합을 구현합니다.
MI_PropertySetFT MI_PropertySet 구조에 사용되는 지원 구조체입니다. "MI_PropertySet_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_ProviderFT MI_ClassDecl 및 MI_Module 구조에 사용되는 지원 구조체입니다.
MI_Qualifier CIM 한정자를 나타냅니다.
MI_QualifierDecl CIM 한정자 선언을 나타냅니다.
MI_QualifierSet 개발자가 클래스 정의의 한정자를 볼 수 있습니다.
MI_QualifierSetFT MI_QualifierSet 구조체에 사용되는 지원 구조체입니다. "MI_QualifierSet_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_Real32A MI_Real32 형식의 배열을 나타냅니다. (MI_Real32A)
MI_Real32AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Real32AField)
MI_Real32Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Real32Field)
MI_Real64A MI_Real64 형식의 배열을 나타냅니다. (MI_Real64A)
MI_Real64AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Real64AField)

MI_Real64Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Real64Field)
MI_ReferenceA MI_Instance 형식에 대한 포인터 배열을 나타냅니다.
MI_ReferenceAField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ReferenceAField)
MI_ReferenceField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_ReferenceField)
MI_SchemaDecl CIM 클래스 및 CIM 한정자 선언을 포함하는 CIM 스키마의 스키마 개체를 나타냅니다.
MI_Serializer 특정 serialization 기술에 연결된 개체입니다.
MI_SerializerFT MI_ClientFT_V1 구조에 사용되는 지원 구조체입니다. "MI_Serializer_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_Server 이 구조체는 _Context, Instance, PropertySet 및 Filter의 모든 형식에 대한 기본 함수 테이블을 정의합니다.
MI_ServerFT MI_Server 구조에 사용되는 지원 구조체입니다. "MI_Server_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_Session 대상과 연결되고 연결된 자격 증명 및 옵션 집합이 있는 개체입니다. .
MI_SessionCallbacks 로깅 및 오류 메시지를 처리하는 콜백 함수 포인터용 컨테이너입니다.

MI_SessionFT 세션 개체의 모든 작업에 대한 함수 테이블입니다.
MI_Sint16A MI_Sint16 형식의 배열을 나타냅니다. (MI_Sint16A)
MI_Sint16AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Sint16AField)
MI_Sint16Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Sint16Field)
MI_Sint32A MI_Sint32 형식의 배열을 나타냅니다. (MI_Sint32A)
MI_Sint32AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Sint32AField)
MI_Sint32Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Sint32Field)
MI_Sint64A MI_Sint64 형식의 배열을 나타냅니다. (MI_Sint64A)
MI_Sint64AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Sint64AField)
MI_Sint64Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Sint64Field)
MI_Sint8A MI_Sint8 형식의 배열을 나타냅니다. (MI_Sint8A)
MI_Sint8AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Sint8AField)

MI_Sint8Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Sint8Field)
MI_StringA null로 끝나는 MI_Char* 문자열에 대한 포인터 배열을 나타냅니다.
MI_StringAField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_StringAField)
MI_StringField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_StringField)
MI_SubscriptionDeliveryOptions 구독 옵션 개체는 구독 작업에 전달하는 데 사용되는 구성 옵션을 저장합니다.
MI_SubscriptionDeliveryOptionsFT MI_SubscriptionDeliveryOptions 구조에 사용되는 지원 구조체입니다. "MI_SubscriptionDeliveryOptions_" 이름 접두사로 함수를 사용하여 이러한 구조를 조작합니다.
MI_Timestamp MI_Timestamp 타임스탬프 또는 특정 시점을 지정합니다.
MI_Uint16A MI_Uint16 형식의 배열을 나타냅니다.
MI_Uint16AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Uint16AField)
MI_Uint16Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Uint16Field)
MI_Uint32A MI_Uint32 형식의 배열을 나타냅니다. (MI_Uint32A)
MI_Uint32AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Uint32AField)

MI_Uint32Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Uint32Field)
MI_Uint64A MI_Uint64 형식의 배열을 나타냅니다. (MI_Uint64A)
MI_Uint64AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Uint64AField)
MI_Uint64Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Uint64Field)
MI_Uint8A MI_Uint8 형식의 배열을 나타냅니다. (MI_Uint8A)
MI_Uint8AField MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Uint8AField)
MI_Uint8Field MI_Instance 구조체 내의 속성을 나타냅니다. (MI_Uint8Field)
MI_UserCredentials 사용자의 자격 증명입니다. 여기에는 인증 유형과 사용자 이름 및 암호 또는 인증서 지문이 포함됩니다.
MI_UsernamePasswordCreds 구독 작업에 사용되는 사용자 이름/암호 조합입니다.
MI_UtilitiesFT MI_ClientFT_V1 구조에 사용되는 지원 구조체입니다. 이름 접두사 "MI_Utilities_"와 함께 함수를 사용하여 이러한 구조를 조작합니다.
MI_Value 모든 CIM 데이터 형식의 통합입니다.

열거형

MI_CallbackMode WriteError 및 PromptUser 함수에 대한 CIM 확장의 콜백 모드를 정의합니다.
MI_CancellationReason 작업이 취소되는 이유를 시스템에 알리기 위해 작업 취소 요청에 전달할 값입니다. 서비스가 종료되는 경우 이러한 값 중 하나를 공급자에게 전달할 수도 있습니다.
MI_DestinationOptions_ImpersonationType DCOM 프로토콜 처리기에서 서버에서 가장을 수행하는 방법을 지정하는 데 사용됩니다.
MI_ErrorCategory 이 열거형은 CIM 확장에 대한 오류 범주를 정의합니다.
MI_LocaleType 로캘을 설정하고 가져오는 경우 로캘 유형이 필요합니다.
MI_OperationCallback_ResponseType MI_CallbackMode MI_CALLBACKMODE_INQUIRE 경우 콜백에서 이러한 값 중 하나를 사용할 수 있습니다.
MI_PromptType CIM 확장에 대한 프롬프트 형식을 정의합니다.
MI_ProviderArchitecture 이 열거형은 서버에서 사용되는 WMI 공급자 아키텍처를 정의합니다.
MI_Result 함수 반환 코드를 정의합니다.
MI_SubscriptionDeliveryType 밀어넣기 또는 끌어오기 구독 배달 유형을 구분합니다. DCOM 프로토콜을 사용할 때는 지원되지 않습니다.
MI_Type 이러한 값은 CIM 데이터 형식에 대한 한정자, 속성, 참조, 매개 변수 및 메서드 반환 값의 데이터 형식을 지정합니다.

micaut.h 헤더

아티클2023. 03. 08.

이 헤더는 태블릿 PC에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Tablet PC

micaut.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

_IMathInputControlEvents 수학 입력 제어 이벤트 처리기를 노출합니다.
IMathInputControl 잉크 입력을 해석된 수학 출력으로 변환하는 메서드를 노출합니다.

열거형

MICUIELEMENT MIC(수학 입력 컨트롤)의 UI(사용자 인터페이스) 요소를 지정합니다.
MICUIELEMENTSTATE MIC(수학 입력 컨트롤)의 단추 상태를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

midles.h 헤더

아티클2023. 08. 24.

이 헤더는 RPC(원격 프로시저 호출)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- RPC(원격 프로시저 호출)

midles.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

MesBufferHandleReset MesBufferHandleReset 함수는 버퍼 serialization에 대한 핸들을 다시 초기화합니다.
MesDecodeBufferHandleCreate MesDecodeBufferHandleCreate 함수는 디코딩 핸들을 만들고 직렬화의 (고정) 버퍼 스타일에 맞게 초기화합니다.
MesDecodeIncrementalHandleCreate MesDecodeIncrementalHandleCreate 함수는 serialization의 증분 스타일에 대한 디코딩 핸들을 만듭니다.
MesEncodeDynBufferHandleCreate MesEncodeDynBufferHandleCreate 함수는 인코딩 핸들을 만든 다음 동적 버퍼 스타일의 serialization을 위해 초기화합니다.
MesEncodeFixedBufferHandleCreate MesEncodeFixedBufferHandleCreate 함수는 인코딩 핸들을 만든 다음 고정 버퍼 스타일의 serialization을 위해 초기화합니다.
MesEncodeIncrementalHandleCreate MesEncodeIncrementalHandleCreate 함수는 인코딩을 만든 다음 직렬화의 증분 스타일에 맞게 초기화합니다.
MesHandleFree MesHandleFree 함수는 serialization 핸들에 의해 할당된 메모리를 해제합니다.

MesIncrementalHandleReset MesIncrementalHandleReset 함수는 증분 serialization에 대한 핸들을 다시 초기화합니다.
MesInqProcEncodingId MesInqProcEncodingId 함수는 인코딩의 ID를 제공합니다.
NdrMesProcEncodeDecode2 NdrMesProcEncodeDecode2를 변경하거나 사용할 수 없습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mileffects.h 헤더

아티클2023. 08. 24.

이 헤더는 WPF 비트맵 효과에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- WPF 비트맵 효과

mileffects.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IMILBitmapEffect WPF(Windows Presentation Foundation) 비트맵 효과를 정의하는 메서드를 노출합니다.
IMILBitmapEffectConnections 비트맵 효과에 의해 노출된 입력 및 출력 커넥터를 검색하는 데 사용되는 메서드를 노출합니다.
IMILBitmapEffectConnectionsInfo 비트맵 효과에 의해 노출되는 입력 및 출력 핀에 대한 정보를 검색하는 메서드를 노출합니다.
IMILBitmapEffectConnector 효과 출력 핀을 정의하는 메서드를 노출합니다.
IMILBitmapEffectConnectorInfo 특정 입력 또는 출력 커넥터 핀에 대한 정보를 검색하는 메서드를 노출합니다.
IMILBitmapEffectEvents 효과 이벤트를 정의하는 메서드를 노출합니다.
IMILBitmapEffectFactory WPF(Windows Presentation Foundation) Microsoft Win32 비트맵 효과 개체를 만드는 데 사용되는 메서드를 노출합니다.
IMILBitmapEffectGroup 효과 그룹에 액세스하는 데 사용되는 메서드를 노출합니다.
IMILBitmapEffectGroupImpl

효과 그룹을 정의하는 메서드를 노출합니다.
IMILBitmapEffectImpl OUT IMILBitmapEffect 개체를 정의하는 메서드를 노출합니다.
IMILBitmapEffectInputConnector 입력 연결을 정의하는 메서드를 노출합니다.
IMILBitmapEffectInteriorInputConnector 내부 입력 커넥터를 정의하는 메서드를 노출합니다.
IMILBitmapEffectInteriorOutputConnector 내부 출력 커넥터를 정의하는 메서드를 노출합니다.
IMILBitmapEffectOutputConnector 출력 커넥터를 정의하는 메서드를 노출합니다. (IMILBitmapEffectOutputConnector)
IMILBitmapEffectOutputConnectorImpl 출력 커넥터를 정의하는 메서드를 노출합니다. (IMILBitmapEffectOutputConnectorImpl)
IMILBitmapEffectPrimitive 비트맵 효과의 출력을 만드는 메서드를 노출합니다. WPF(타사 Windows Presentation Foundation) 비트맵 효과를 만들려면 이 인터페이스를 구현해야 합니다.
IMILBitmapEffectPrimitiveImpl 비트맵 효과의 상태를 보고하는 메서드를 노출합니다. WPF(타사 Windows Presentation Foundation) 비트맵 효과를 만들려면 이 인터페이스를 구현해야 합니다.
IMILBitmapEffectRenderContext IMILBitmapEffectRenderContext 개체를 정의하는 메서드를 노출합니다.
IMILBitmapEffectRenderContextImpl IMILBitmapEffectRenderContext를 정의하는 메서드를 노출합니다.
IMILBitmapEffects 효과의 열거형을 정의하는 메서드를 노출합니다.

구조체

MilMatrix3x2D 3x3 행렬을 나타냅니다.
MILMatrixF 4x4 아핀 변환 행렬을 나타냅니다.
MilPoint2D 2차원 공간의 x 및 y 좌표 쌍을 나타냅니다. M
MilRectD 사각형의 너비, 높이 및 위치를 설명합니다. M

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

minidumpapiset.h 헤더

아티클2023. 08. 24.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

minidumpapiset.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

MiniDumpReadDumpStream 사용자 모드 미니덤프 파일에서 스트림을 읽습니다.
MiniDumpWriteDump 사용자 모드 미니덤프 정보를 지정된 파일에 씁니다.

콜백 함수

MINIDUMP_CALLBACK_ROUTINE MiniDumpWriteDump와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. 확장된 미니덤프 정보를 받습니다.

구조체

MINIDUMP_CALLBACK_INFORMATION MiniDumpWriteDump 함수에서 사용할 수 있는 선택적 콜백 함수에 대한 포인터를 포함합니다.
MINIDUMP_CALLBACK_INPUT MiniDumpCallback 함수에서 사용하는 정보를 포함합니다.

MINIDUMP_CALLBACK_OUTPUT MiniDumpCallback 함수에서 반환된 정보를 포함합니다.
MINIDUMP_DIRECTORY 미니덤프 파일의 특정 데이터 스트림에 액세스하는 데 필요한 정보를 포함합니다.
MINIDUMP_EXCEPTION 예외 정보를 포함합니다.
MINIDUMP_EXCEPTION_INFORMATION MiniDumpWriteDump 함수로 미니덤프 파일에 기록한 예외 정보를 포함합니다.
MINIDUMP_EXCEPTION_STREAM 예외 정보 스트림을 나타냅니다.
MINIDUMP_FUNCTION_TABLE_DESCRIPTOR 함수 테이블 스트림을 나타냅니다.
MINIDUMP_FUNCTION_TABLE_STREAM 함수 테이블 스트림의 헤더를 나타냅니다.
MINIDUMP_HANDLE_DATA_STREAM 핸들 데이터 스트림의 헤더를 나타냅니다.
MINIDUMP_HANDLE_DESCRIPTOR 미니덤프가 쓰여졌을 때의 개별 시스템 핸들 상태를 포함합니다.
MINIDUMP_HANDLE_DESCRIPTOR_2 미니덤프가 쓰여졌을 때의 개별 시스템 핸들 상태를 설명합니다.
MINIDUMP_HANDLE_OBJECT_INFORMATION 핸들의 개체별 정보를 포함합니다.
MINIDUMP_HANDLE_OPERATION_LIST 핸들 작업 목록을 포함합니다.
MINIDUMP_HEADER

미니덤프 파일의 헤더 정보를 포함합니다.
MINIDUMP_INCLUDE_MODULE_CALLBACK 콜백 형식이 IncludeModuleCallback인 경우, MiniDumpCallback 함수에 대한 정보를 포함합니다.
MINIDUMP_INCLUDE_THREAD_CALLBACK 콜백 형식이 IncludeThreadCallback인 경우 MiniDumpCallback 함수에 대한 정보를 포함합니다.
MINIDUMP_IO_CALLBACK I/O 콜백 정보를 포함합니다.
MINIDUMP_LOCATION_DESCRIPTOR 미니덤프 파일 내에서 데이터 스트림의 위치를 설명하는 정보를 포함합니다. (MINIDUMP_LOCATION_DESCRIPTOR)
MINIDUMP_LOCATION_DESCRIPTOR64 미니덤프 파일 내에서 데이터 스트림의 위치를 설명하는 정보를 포함합니다. M
MINIDUMP_MEMORY_DESCRIPTOR 메모리 범위를 설명합니다. (MINIDUMP_MEMORY_DESCRIPTOR)
MINIDUMP_MEMORY_DESCRIPTOR64 메모리 범위를 설명합니다. M
MINIDUMP_MEMORY_INFO 메모리 영역을 설명합니다.
MINIDUMP_MEMORY_INFO_LIST 메모리 영역 목록을 포함합니다.
MINIDUMP_MEMORY_LIST 메모리 범위 목록을 포함합니다. (MINIDUMP_MEMORY_LIST)
MINIDUMP_MEMORY64_LIST 메모리 범위 목록을 포함합니다. M
MINIDUMP_MISC_INFO 다양한 정보를 포함합니다.

MINIDUMP_MISC_INFO_2 기타 정보 스트림의 정보를 나타냅니다.
MINIDUMP_MODULE 특정 모듈의 정보를 포함합니다.
MINIDUMP_MODULE_CALLBACK 콜백 형식이 ModuleCallback인 경우 MiniDumpCallback 함수에 대한 모듈 정보를 포함합니다.
MINIDUMP_MODULE_LIST 모듈 목록을 포함합니다.
MINIDUMP_READ_MEMORY_FAILURE_CALLBACK 실패한 메모리 읽기 작업에 대한 정보를 포함합니다.
MINIDUMP_STRING 문자열을 설명합니다.
MINIDUMP_SYSTEM_INFO 프로세서 및 운영 체제 정보를 포함합니다.
MINIDUMP_THREAD 특정 스레드의 정보를 포함합니다.
MINIDUMP_THREAD_CALLBACK 콜백 형식이 ThreadCallback인 경우 MiniDumpCallback 함수의 스레드 정보를 포함합니다.
MINIDUMP_THREAD_EX 특정 스레드의 확장 정보를 포함합니다.
MINIDUMP_THREAD_EX_CALLBACK 콜백 형식이 ThreadExCallback인 경우 MiniDumpCallback 함수의 확장 스레드 정보를 포함합니다.
MINIDUMP_THREAD_EX_LIST 스레드 목록을 포함합니다. (MINIDUMP_THREAD_EX_LIST)
MINIDUMP_THREAD_INFO

스레드 상태 정보를 포함합니다.
MINIDUMP_THREAD_INFO_LIST 스레드 목록을 포함합니다. (MINIDUMP_THREAD_INFO_LIST)
MINIDUMP_THREAD_LIST 스레드 목록을 포함합니다. (MINIDUMP_THREAD_LIST)
MINIDUMP_UNLOADED_MODULE 언로드된 모듈에 대한 정보를 포함합니다. 이 정보를 통해 더 이상 로드되지 않는 코드를 호출하는 문제를 진단할 수 있습니다.
MINIDUMP_UNLOADED_MODULE_LIST 언로드된 모듈 목록을 포함합니다.
MINIDUMP_USER_STREAM 데이터 스트림에 저장된 사용자 정의 정보를 포함합니다.
MINIDUMP_USER_STREAM_INFORMATION MiniDumpWriteDump 함수에서 사용하는 사용자 데이터 스트림 목록을 포함합니다.

열거형

MINIDUMP_CALLBACK_TYPE MiniDumpCallback 함수에서 반환되는 정보의 형식을 식별합니다.
MINIDUMP_HANDLE_OBJECT_INFORMATION_TYPE 개체별 정보의 형식을 식별합니다.
MINIDUMP_SECONDARY_FLAGS 미니덤프의 보조 플래그를 지정합니다.
MINIDUMP_STREAM_TYPE 미니덤프 데이터 스트림의 형식을 나타냅니다.

MINIDUMP_TYPE MiniDumpWriteDump 함수에 의해 미니덤프 파일에 기록될 정보의 형식을 식별합니다.
MODULE_WRITE_FLAGS MiniDumpWriteDump 함수에 의해 미니덤프 파일에 기록될 모듈 정보의 형식을 식별합니다.
THREAD_WRITE_FLAGS MiniDumpWriteDump 함수에 의해 미니덤프 파일에 기록될 스레드 정보의 형식을 식별합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

minwinbase.h 헤더

아티클2023. 08. 24.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지
- 시스템 서비스

minwinbase.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

LocalDiscard 지정된 로컬 메모리 개체를 삭제합니다. 메모리 개체의 잠금 수는 0이어야 합니다.

콜백 함수

LPOVERLAPPED_COMPLETION_ROUTINE ReadFileEx 및 WriteFileEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. 비동기 입출력(I/O) 작업이 완료 또는 취소되고 호출 스레드가 경고 가능한 상태일 때 호출됩니다.

구조체

CREATE_PROCESS_DEBUG_INFO 디버거에서 사용할 수 있는 프로세스 만들기 정보를 포함합니다.
CREATE_THREAD_DEBUG_INFO 디버거에서 사용할 수 있는 스레드 생성 정보를 포함합니다.
DEBUG_EVENT 디버깅 이벤트를 설명합니다.

EXCEPTION_DEBUG_INFO 디버거에서 사용할 수 있는 예외 정보를 포함합니다.
EXIT_PROCESS_DEBUG_INFO 종료 프로세스에 대한 종료 코드를 포함합니다.
EXIT_THREAD_DEBUG_INFO 종료 스레드에 대한 종료 코드를 포함합니다.
FILETIME 1601년 1월 1일(UTC) 이후의 100나노초 간격 수를 나타내는 64비트 값을 포함합니다.
LOAD_DLL_DEBUG_INFO 방금 로드된 DLL(동적 연결 라이브러리)에 대한 정보를 포함합니다.
OUTPUT_DEBUG_STRING_INFO 디버깅 문자열의 주소, 형식 및 길이(바이트)를 포함합니다.
OVERLAPPED 비동기(또는 겹치는) 입력 및 출력(I/O)에 사용되는 정보를 포함합니다.
OVERLAPPED_ENTRY GetQueuedCompletionStatusEx 함수 호출로 반환된 정보를 포함합니다.
PROCESS_HEAP_ENTRY 힙 요소에 대한 정보를 포함합니다.
REASON_CONTEXT 전원 요청에 대한 정보를 포함합니다. 이 구조체는 PowerCreateRequest 및 SetWaitableTimerEx 함수에서 사용됩니다.
RIP_INFO RIP 디버그 이벤트를 발생시킨 오류를 포함합니다.
Systemtime 월, 일, 연도, 평일, 시간, 분, 초 및 밀리초에 대한 개별 멤버를 사용하여 날짜 및 시간을 지정합니다. 시간은 호출되는 함수에 따라 UTC(협정 세계시) 또는 현지 시간입니다.

UNLOAD_DLL_DEBUG_INFO 방금 언로드된 DLL(동적 연결 라이브러리)에 대한 정보를 포함합니다.
WIN32_FIND_DATAA FindFirstFile, FindFirstFileEx 또는 FindNextFile 함수에서 찾은 파일에 대한 정보를 포함합니다. (ANSI)
WIN32_FIND_DATAW FindFirstFile, FindFirstFileEx 또는 FindNextFile 함수에서 찾은 파일에 대한 정보를 포함합니다. (유니코드)

열거형

FILE_INFO_BY_HANDLE_CLASS GetFileInformationByHandleEx가 검색하거나 SetFileInformationByHandle이 설정해야 하는 파일 정보의 유형을 식별합니다.
FINDEX_INFO_LEVELS 반환된 데이터의 정보 수준을 지정하기 위해 FindFirstFileEx 함수와 함께 사용되는 값을 정의합니다.
FINDEX_SEARCH_OPS 수행할 필터링의 유형을 지정하기 위해 FindFirstFileEx 함수와 함께 사용할 값을 정의합니다.
GET_FILEEX_INFO_LEVELS 반환된 데이터의 정보 수준을 지정하기 위해 GetFileAttributesEx 및 GetFileAttributesTransacted 함수와 함께 사용할 값을 정의합니다.
READ_DIRECTORY_NOTIFY_INFORMATION_CLASS ReadDirectoryChangesExW 함수를 호출하는 애플리케이션이 요청할 수 있는 정보 유형을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mmc.h 헤더

아티클2023. 03. 08.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Hyper-V WMI 공급자
- Microsoft 관리 콘솔 2.0

mmc.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IColumnData IColumnData 인터페이스는 MMC 1.2에 도입되었습니다.
IComponent IComponent 인터페이스를 사용하면 MMC가 스냅인과 통신할 수 있습니다. IComponentData 인터페이스와 마찬가지로 IComponent는 일반적으로 보기 수준에서 구현되며 결과 창에 표시되는 항목과 밀접하게 연결됩니다.
IComponent2 스냅인으로 구현된 IComponent2 인터페이스는 MMC 2.0에 도입되어 IComponent 인터페이스를 대체합니다.
IComponentData IComponentData 인터페이스를 사용하면 MMC가 스냅인과 통신할 수 있습니다. IComponent 인터페이스와 마찬가지로 IComponentData는 일반적으로 문서 수준에서 구현되며 범위 창에 표시되는 항목(폴더)과 밀접하게 연결됩니다.
IComponentData2 IComponentData2 인터페이스는 IComponentData 인터페이스를 대체합니다.
IConsole 콘솔과의 통신을 사용하도록 설정합니다.
IConsole2 IConsole2 인터페이스는 MMC 1.1에서 도입되었습니다.

IConsole3 IConsole3 인터페이스는 IConsole2 인터페이스를 대체합니다. IConsole3 인터페이스에는 범위 노드를 프로그래밍 방식으로 이름 바꾸기 모드로 배치할 수 있는 IConsole3::RenameScopeItem 메서드가 포함되어 있습니다.
IConsoleNameSpace 스냅인을 사용하여 범위 창에서 동적 하위 컨테이너를 열거할 수 있습니다. 특정 스냅인은 하위 컨테이너로 자격이 있는 항목을 결정합니다.
IConsoleNameSpace2 IConsoleNameSpace2 인터페이스는 MMC 1.1에 도입되었습니다.
IConsolePower IConsolePower 인터페이스는 전원 관리를 지원하는 운영 체제에서 실행 상태 및 유휴 타이머를 제어합니다.
IConsolePowerSink IConsolePowerSink 인터페이스는 전원 관리 메시지를 모니터링하고 응답합니다.
IConsoleVerb IConsoleVerb 인터페이스를 사용하면 스냅인을 통해 잘라내기, 복사, 붙여넣기, 삭제, 속성, 이름 바꾸기, 새로 고침 및 인쇄를 비롯한 표준 동사를 사용할 수 있습니다. 항목을 선택하면 스냅인에서 이러한 동사의 상태를 업데이트할 수 있습니다.
IContextMenuCallback IContextMenuCallback 인터페이스는 상황에 맞는 메뉴에 메뉴 항목을 추가하는 데 사용됩니다.
IContextMenuCallback2 IContextMenuCallback2 인터페이스는 상황에 맞는 메뉴에 메뉴 항목을 추가하는 데 사용됩니다. 이 인터페이스는 IContextMenuCallback을 대체합니다.
IContextMenuProvider IContextMenuProvider 인터페이스는 해당 메뉴에 항목을 추가하고 확장에서 해당 메뉴를 확장하고 결과 상황에 맞는 메뉴를 표시하기 위해 새 상황에 맞는 메뉴를 만드는 메서드를 구현합니다.
IControlbar IControlbar 인터페이스는 도구 모음 및 기타 컨트롤을 만드는 방법을 제공합니다.

IDisplayHelp IDisplayHelp 인터페이스는 MMC 버전 1.1에 도입되었습니다.
IEnumTASK IEnumTASK 인터페이스는 MMC 1.1에서 도입되었습니다.
IExtendContextMenu IExtendContextMenu 인터페이스를 사용하면 스냅인에서 기존 상황에 맞는 메뉴에 항목을 추가할 수 있습니다.
IExtendControlbar IExtendControlbar 인터페이스를 사용하면 확장에서 콘솔에 컨트롤 막대를 추가할 수 있습니다. 이렇게 하면 도구 모음 또는 기타 사용자 인터페이스 개선 사항을 추가하여 스냅인의 기능과 모양을 개선할 수 있습니다.
IExtendPropertySheet 스냅인 구성 요소가 항목의 속성 시트에 페이지를 추가할 수 있도록 합니다.
IExtendPropertySheet2 IExtendPropertySheet2 인터페이스는 MMC 1.1에 도입되었습니다.
IExtendTaskPad IExtendTaskPad 인터페이스는 MMC 1.1에 도입되었습니다.
IExtendView IExtendView 인터페이스는 확장 보기에 대한 정보를 제공합니다.
IHeaderCtrl 열을 조작할 수 있도록 설정하고 콘솔의 결과 보기 창에 표시할 정보의 종류를 나타냅니다.
IHeaderCtrl2 IHeaderCtrl2 인터페이스는 MMC 1.2에 도입되었습니다.
IImageList IImageList 인터페이스를 사용하면 사용자가 콘솔의 결과 또는 범위 창에서 항목의 아이콘으로 사용할 이미지를 삽입할 수 있습니다.

IMenuButton IMenuButton 인터페이스를 사용하면 사용자가 스냅인에 대한 메뉴 단추를 추가하고 관리할 수 있습니다.
IMessageView IMessageView 인터페이스는 MMC 1.2에 도입되었습니다.
IMMCVersionInfo IMMCVersionInfo 인터페이스는 설치된 MMC 애플리케이션에 대한 버전 정보를 제공합니다.
INodeProperties INodeProperties 인터페이스는 노드에 대한 텍스트 전용 속성을 검색합니다.
IPropertySheetCallback IPropertySheetCallback 인터페이스는 속성 시트에 속성 페이지를 추가하기 위해 스냅인에서 사용하는 COM 기반 인터페이스입니다.
IPropertySheetProvider IPropertySheetProvider 인터페이스는 Windows 속성 시트를 COM 개체로 구현합니다.
IRequiredExtensions IRequiredExtensions 인터페이스는 MMC 1.1에 도입되었습니다.
IResultData IResultData 인터페이스를 사용하면 사용자가 결과 보기 창과 연결된 항목을 추가, 제거, 찾기 및 수정할 수 있습니다. 또한 결과 보기 창의 보기 스타일을 조작할 수 있습니다.
IResultData2 IResultData2 인터페이스는 IResultData 인터페이스를 대체합니다. IResultData2 인터페이스에는 결과 노드를 프로그래밍 방식으로 이름 바꾸기 모드로 전환할 수 있는 IResultData2::RenameResultItem 메서드가 포함되어 있습니다.
IResultDataCompare 기본 스냅인에서 결과 창에 정렬된 순서로 표시되는 결과 항목을 비교할 수 있습니다.
IResultDataCompareEx 기본 스냅인에서 결과 창에 정렬된 순서로 표시되는 범위와 결과 항목을 모두 비교할 수 있습니다.

IResultOwnerData IResultOwnerData 인터페이스는 LVS_OWNERDATA 스타일이 설정된 목록 보기 컨트롤인 가상 목록의 사용을 지원합니다.
ISnapinAbout ISnapinAbout 인터페이스를 사용하면 콘솔이 스냅인에서 저작권 및 버전 정보를 가져올 수 있습니다. 또한 콘솔은 이 인터페이스를 사용하여 스냅인에서 정적 폴더에 대한 이미지를 가져옵니다.
ISnapinHelp 스냅인에서 HTML 도움말 지원을 추가할 수 있습니다. (ISnapinHelp)
ISnapinHelp2 스냅인에서 HTML 도움말 지원을 추가할 수 있습니다. (ISnapinHelp2)
IStringTable IStringTable 인터페이스는 MMC 1.1에서 도입되었습니다.
IToolbar IToolbar 인터페이스는 새 도구 모음을 만들고, 항목을 추가하고, 도구 모음을 확장하고, 결과 새 도구 모음을 표시하는 데 사용됩니다. 각 도구 모음은 컨트롤 바 내에서 자체 밴드에 만들어집니다.
IViewExtensionCallback IViewExtensionCallback 인터페이스는 결과 창에 보기를 추가하는 데 사용됩니다.

Functions

IS_SPECIAL_COOKIE IS_SPECIAL_COOKIE 매크로는 스냅인의 IComponent::QueryDataObject 메서드 호출에서 MMC가 전달한 MMC_COOKIE 값이 특별한 쿠키 형식인지 여부를 결정합니다.
IS_SPECIAL_DATAOBJECT 스냅인의 Notify 메서드 호출에서 MMC가 전달한 LPDATAOBJECT가 실제 IDataObject 개체에 대한 포인터 대신 특수 형식의 데이터 개체인지 여부를 확인합니다.

MMCFreeNotifyHandle 속성 시트에서 만든 MMCPropertyChangeNotify 호출에 대한 응답으로 MMC가 스냅인으로 보낸 MMCN_PROPERTY_CHANGE 알림 메시지에 대한 핸들을 해제하기 위해 스냅인에 의해 호출됩니다.
MMCPropertyChangeNotify 스냅인 속성 시트가 항목의 속성이 변경되었음을 IComponent 또는 IComponentData 인터페이스에 알릴 수 있도록 합니다.
MMCPropertyHelp 속성 페이지의 병합된 MMC HTML 도움말 파일에 지정된 HTML 도움말 항목을 표시합니다.
MMCPropPageCallback MMCPropPageCallback 함수는 MFC(Microsoft Foundation Classs) 기반 스냅인에서만 필요합니다. 이 함수는 페이지를 만드는 동안 올바른 모듈 상태를 설정합니다.

구조체

CONTEXTMENUITEM CONTEXTMENUITEM 구조체는 IContextMenuCallback::AddItem 메서드 또는 IContextMenuProvider::AddItem 메서드(IContextMenuCallback에서 상속됨)로 전달되어 새 메뉴 항목, 하위 메뉴 또는 삽입 지점을 정의합니다.
CONTEXTMENUITEM2 CONTEXTMENUITEM2 구조체는 MMC 2.0에서 도입되었습니다.
MENUBUTTONDATA MENUBUTTONDATA 구조체에는 도구 모음에서 단추를 만드는 데 사용되는 값이 포함되어 있습니다.
MMC_COLUMN_DATA MMC_COLUMN_DATA 구조는 MMC 1.2에 도입되었습니다.
MMC_COLUMN_SET_DATA MMC_COLUMN_SET_DATA 구조는 MMC 1.2에 도입되었습니다.

MMC_EXPANDSYNC_STRUCT MMC_EXPANDSYNC_STRUCT 구조는 MMC 1.1에 도입되었습니다.
MMC_EXT_VIEW_DATA MMC_EXT_VIEW_DATA 구조는 MMC 2.0에 도입되었습니다.
MMC_FILTERDATA MMC_FILTERDATA 구조는 MMC 1.2에 도입되었습니다.
MMC_LISTPAD_INFO MMC_LISTPAD_INFO 구조는 MMC 1.1에 도입되었습니다.
MMC_RESTORE_VIEW MMC_RESTORE_VIEW 구조는 MMC 1.1에 도입되었습니다.
MMC_SORT_DATA 열 집합에 있는 단일 열의 열 정렬 데이터를 포함합니다.
MMC_SORT_SET_DATA 정렬 데이터가 영구적으로 저장되는 목록 보기 열 집합을 설정하고 검색하는 데 사용됩니다.
MMC_TASK MMC_TASK 구조는 MMC 1.1에 도입되었습니다.
MMC_TASK_DISPLAY_BITMAP MMC_TASK_DISPLAY_BITMAP 구조는 MMC 1.1에 도입되었습니다.
MMC_TASK_DISPLAY_OBJECT 작업 패드에 작업 또는 배경을 표시하기 위해 이미지 형식과 해당 이미지를 사용하는 데 필요한 모든 데이터를 지정합니다.
MMC_TASK_DISPLAY_SYMBOL MMC_TASK_DISPLAY_SYMBOL 구조는 MMC 1.1에 도입되었습니다.
MMC_VISIBLE_COLUMNS MMC에서 MMCN_COLUMNS_CHANGED 알림과 함께 열 집합의 열이 표시되는 스냅인을 알리는 데 사용됩니다.

MMCBUTTON MMCBUTTON 구조체에는 도구 모음에서 단추를 만드는 데 사용되는 값이 포함되어 있습니다. 이 구조는 일반적인 컨트롤과 관련된 SDK(플랫폼 소프트웨어 개발 키트) 항목에서 설명하는 TBBUTTON 구조와 유사합니다.
RDCOMPARE RDCOMPARE 구조는 MMC 1.2에 도입되었습니다.
RDITEMHDR RDITEMHDR 구조는 MMC 1.2에 도입되었습니다.
RESULT_VIEW_TYPE_INFO RESULT_VIEW_TYPE_INFO 구조는 MMC 2.0에 도입되었습니다.
RESULTDATAITEM RESULTDATAITEM 구조체는 스냅인의 결과 창에서 결과 항목의 특성을 지정하거나 받습니다.
RESULTFINDINFO IResultOwnerData::FindItem 메서드에서 결과 창의 가상 목록에서 키보드 탐색을 지원하는 데 사용됩니다.
SColumnSetID SColumnSetID 구조는 MMC 1.2에 도입되었습니다.
SCOPEDATAITEM SCOPEDATAITEM 구조는 범위 창에 삽입할 항목을 지정합니다.
SMMCDataObjects SMMCDataObjects 구조체는 CCF_MULTI_SELECT_SNAPINS 클립보드 형식에 대한 데이터 형식을 정의합니다.
SMMCObjectTypes SMMCDynamicExtensions 구조는 MMC 1.1에 도입되었습니다.
SNodeID SNodeID 구조체는 MMC 1.1에서 도입되었으며 MMC 1.2의 SNodeID2 구조체로 대체됩니다.

SNodeID2 SNodeID2 구조체는 MMC 1.2에 도입되고 SNodeID 구조를 대체합니다.

열거형

DATA_OBJECT_TYPES DATA_OBJECT_TYPES 열거형은 IComponentData::QueryDataObject 및 IComponent::QueryDataObject의 형식 매개 변수에서 사용하여 지정된 쿠키에 대한 컨텍스트 정보를 가져옵니다.
IconIdentifier IconIdentifier 열거형은 MMC 1.2에서 도입되었습니다.
MMC_ACTION_TYPE MMC_ACTION_TYPE 열거형은 MMC 1.1에서 도입되었습니다.
MMC_BUTTON_STATE MMC_BUTTON_STATE 열거형은 MMC에서 사용할 수 있는 단추의 가능한 상태를 정의합니다. 이러한 값은 IConsoleVerb::GetVerbState, IConsoleVerb::SetVerbState, IToolbar::GetButtonState 및 IToolbar::SetButtonState의 nState 매개 변수에 사용됩니다.
MMC_CONSOLE_VERB MMC_CONSOLE_VERB 열거형은 MMC 동사에 사용할 수 있는 명령 식별자를 정의합니다. 이러한 값은 IConsoleVerb::GetVerbState, IConsoleVerb::SetVerbState 및 IConsoleVerb::SetDefaultVerb의 m_eCmdID 매개 변수에 사용됩니다.
MMC_CONTROL_TYPE MMC_CONTROL_TYPE 열거형은 컨트롤 표시줄에 표시할 수 있는 컨트롤의 종류를 정의합니다. 값은 IControlbar::Attach 및 IControlbar::Create 메서드의 nType 매개 변수에서 사용할 수 있습니다.
MMC_FILTER_CHANGE_CODE MMC_FILTER_CHANGE_CODE 열거형은 MMC 1.2에 도입되었습니다.
MMC_FILTER_TYPE MMC_FILTER_TYPE 열거형은 MMC 1.2에 도입되었습니다.

MMC_MENU_COMMAND_IDS MMC_MENU_COMMAND_IDS 열거형은 MMC에서 예약한 명령 식별자를 정의합니다.
MMC_NOTIFY_TYPE MMC_NOTIFY_TYPE 열거형은 IComponentData::Notify, IComponent::Notify 또는 IExtendControlbar::ControlbarNotify를 호출할 때 콘솔의 노드 관리자가 스냅인으로 보낼 수 있는 사용자 작업의 알림을 정의합니다.
MMC_RESULT_VIEW_STYLE MMC_RESULT_VIEW_STYLE 열거형은 MMC 결과 보기 창에서 보기 스타일을 설정하는 데 사용할 수 있는 Win32 목록 보기 스타일(LVS_*) 플래그를 정의합니다. IResultData::ModifyViewStyle 메서드의 매개 변수 추가 및 제거에 사용할 수 있습니다.
MMC_SCOPE_ITEM_STATE SCOPEDATAITEM 구조체의 nState 멤버를 지정하는 데 사용됩니다.
MMC_TASK_DISPLAY_TYPE MMC_TASK_DISPLAY_TYPE 열거형은 MMC 1.1에서 도입되었습니다.
MMC_VIEW_TYPE MMC_VIEW_TYPE 열거형은 결과 뷰 형식을 지정하고 RESULT_VIEW_TYPE_INFO 구조에 사용됩니다.

mmcobj.h 헤더

아티클2023. 08. 24.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 레거시 Windows 환경 기능
- Microsoft 관리 콘솔 2.0
- 보안 및 ID

mmcobj.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ISnapinProperties ISnapinProperties 인터페이스를 사용하면 스냅인에서 스냅인의 속성을 초기화하고 속성이 추가, 변경 또는 삭제될 때 알림을 받을 수 있습니다.
ISnapinPropertiesCallback ISnapinPropertiesCallback 인터페이스는 스냅인에 대한 속성 이름을 추가합니다. 이 인터페이스는 스냅인을 위해 MMC에 의해 구현됩니다.

구조체

MMC_SNAPIN_PROPERTY MMC_SNAPIN_PROPERTY 구조는 MMC 2.0에서 도입되었습니다.

열거형

_ColumnSortOrder IResultsViewer::SortOrderProperty에서 쿼리를 정렬하는 방법을 나타내거나 설정하는 데 사용됩니다.

_DocumentMode DocumentMode 열거형은 Document.Mode 속성에서 사용되며 문서를 여는 방법을 지정합니다. 이 열거형은 MMC 2.0 Automation 개체 모델에 적용됩니다.
_ExportListOptions ExportListOptions 열거형은 View.ExportList 메서드에서 사용되며 목록 보기 콘텐츠를 파일에 쓸 때 옵션을 지정합니다.
_ListViewMode ListViewMode 열거형은 View.ListViewMode 속성에서 목록 보기를 정의하는 데 사용됩니다.
_ViewOptions ViewOptions 열거형은 Views.Add 메서드에서 사용되며 보기, scope 트리 및 도구 모음의 표시 여부와 보기의 지속성 상태를 지정합니다.
MMC_PROPERTY_ACTION MMC_PROPERTY_ACTION 열거형은 MMC_SNAPIN_PROPERTY 구조체에 포함된 속성에 발생할 수 있는 작업을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mmddk.h 헤더

아티클2023. 08. 24.

멀티미디어 디바이스 개발 키트에 대한 파일을 포함합니다.

자세한 내용은 다음을 참조하세요.

- 오디오 디바이스 DDI 참조
- 휴먼 인터페이스 디바이스 참조
- Windows 멀티미디어

mmddk.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

mmGetCurrentTask mmGetCurrentTask 함수는 더 이상 사용되지 않습니다. 애플리케이션에서 이 함수를 사용하면 안 됩니다.
mmTaskBlock mmTaskBlock 함수는 더 이상 사용되지 않습니다. 애플리케이션에서 이 함수를 사용하면 안 됩니다.
mmTaskCreate mmTaskCreate 함수는 더 이상 사용되지 않습니다. 애플리케이션에서 이 함수를 사용하면 안 됩니다.
mmTaskSignal mmTaskSignal 함수는 더 이상 사용되지 않습니다. 애플리케이션에서 이 함수를 사용하면 안 됩니다.
mmTaskYield mmTaskYield 함수는 더 이상 사용되지 않습니다. 애플리케이션에서 이 함수를 사용하면 안 됩니다.

구조체

JOYREGHWVALUES JOYREGHWVALUES(mmddk.h) 구조체에는 하드웨어에서 반환하는 값 범위(보정에 의해 채워짐)가 포함됩니다.
MDEVICECAPSEX MDEVICECAPSEX 구조에는 PnP(플러그 앤 플레이) 디바이스 드라이버에 대한 디바이스 기능 정보가 포함되어 있습니다.
MIDIOPENDESC MIDIOPENDESC 구조체는 MIDI 디바이스를 여는 방법에 대한 정보를 제공하는 클라이언트로 채워진 구조체입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mmdeviceapi.h 헤더

아티클2023. 08. 24.

이 헤더는 핵심 오디오 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 핵심 오디오 API

mmdeviceapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IActivateAudioInterfaceAsyncOperation WASAPI 인터페이스를 활성화하는 비동기 작업을 나타내며 활성화 결과를 검색하는 메서드를 제공합니다.
IActivateAudioInterfaceCompletionHandler WASAPI 인터페이스의 활성화가 완료되었음을 나타내는 콜백을 제공합니다.
IAudioSystemEffectsPropertyChangeNotificationClient 오디오 시스템 효과 속성이 변경될 때 알림을 수신하기 위해 클라이언트에서 구현하는 콜백 인터페이스입니다.
IAudioSystemEffectsPropertyStore 오디오 시스템 효과 저장소를 관리하고 오디오 시스템 효과 속성이 변경되면 알림을 등록할 수 있는 액세스를 제공합니다.
IMMDevice IMMDevice 인터페이스는 멀티미디어 디바이스 리소스의 일반 기능을 캡슐화합니다.
IMMDeviceCollection IMMDeviceCollection 인터페이스는 멀티미디어 디바이스 리소스의 컬렉션을 나타냅니다.
IMMDeviceEnumerator IMMDeviceEnumerator 인터페이스는 멀티미디어 디바이스 리소스를 열거하는 메서드를 제공합니다.
IMMEndpoint

IMMEndpoint 인터페이스는 오디오 엔드포인트 디바이스를 나타냅니다.
IMMNotificationClient IMMNotificationClient 인터페이스는 오디오 엔드포인트 디바이스가 추가 또는 제거되거나, 엔드포인트 디바이스의 상태 또는 속성이 변경되거나, 엔드포인트 디바이스에 할당된 기본 역할이 변경될 때 알림을 제공합니다.

Functions

ActivateAudioInterfaceAsync Windows 스토어 앱이 WASAPI 제품군의 기존 COM(구성 요소 개체 모델) 인터페이스에 액세스할 수 있도록 합니다.

구조체

AudioExtensionParams 이 구조체는 IShellPropSheetExt::AddPages를 통해 제어판 엔드포인트 확장 속성 페이지로 전달되며 엔드포인트 PropertyPages를 만드는 데 사용됩니다.
DIRECTX_AUDIO_ACTIVATION_PARAMS DIRECTX_AUDIO_ACTIVATION_PARAMS 구조체는 DirectSound 스트림의 초기화 매개 변수를 지정합니다.

열거형

AUDIO_SYSTEMEFFECTS_PROPERTYSTORE_TYPE 오디오 시스템 효과 속성 저장소의 유형을 지정합니다.
EDataFlow EDataFlow 열거형은 오디오 엔드포인트 디바이스와 애플리케이션 간에 오디오 데이터가 흐르는 방향을 나타내는 상수를 정의합니다.

EndpointFormFactor EndpointFormFactor 열거형은 오디오 엔드포인트 디바이스의 일반적인 물리적 특성을 나타내는 상수를 정의합니다.
ERole ERole 열거형은 시스템이 오디오 엔드포인트 디바이스에 할당한 역할을 나타내는 상수를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mmeapi.h 헤더

아티클2023. 08. 24.

이 헤더는 Windows 멀티미디어에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 멀티미디어

mmeapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

auxGetDevCaps auxGetDevCaps 함수(mmeapi.h)는 지정된 보조 출력 디바이스의 기능을 검색합니다.
auxGetDevCapsA auxGetDevCaps 함수는 지정된 보조 출력 디바이스의 기능을 검색합니다. (auxGetDevCapsA)
auxGetDevCapsW auxGetDevCapsW(유니코드) 함수(mmeapi.h)는 지정된 보조 출력 디바이스의 기능을 검색합니다.
auxGetNumDevs auxGetNumDevs 함수는 시스템에 있는 보조 출력 디바이스의 수를 검색합니다.
auxGetVolume auxGetVolume 함수는 지정된 보조 출력 디바이스의 현재 볼륨 설정을 검색합니다.
auxOutMessage auxOutMessage 함수는 지정된 보조 출력 디바이스에 메시지를 보냅니다. 또한 이 함수는 메시지의 일부로 전달된 디바이스 식별자에서 오류 검사를 수행합니다.
auxSetVolume auxSetVolume 함수는 지정된 보조 출력 디바이스의 볼륨을 설정합니다.
MEVT_EVENTPARM MEVT_EVENTPARM 매크로는 MIDIEVENT 구조체의 dwEvent 멤버에 지정된 값에서 이벤트 매개 변수 또는 길이를 검색합니다.

MEVT_EVENTTYPE MEVT_EVENTTYPE 매크로는 MIDIEVENT 구조체의 dwEvent 멤버에 지정된 값에서 이벤트 형식을 검색합니다.
midiConnect midiConnect 함수는 MIDI 입력 디바이스를 MIDI를 통해 또는 출력 디바이스에 연결하거나 MIDI를 통해 디바이스를 MIDI 출력 디바이스에 연결합니다.
midiDisconnect midiDisconnect 함수는 MIDI를 통해 또는 출력 디바이스에서 MIDI 입력 디바이스의 연결을 끊거나 MIDI 출력 디바이스에서 MIDI를 통해 디바이스의 연결을 끊습니다.
midiInAddBuffer midiInAddBuffer 함수는 지정된 열린 MIDI 입력 디바이스에 입력 버퍼를 보냅니다. 이 함수는 시스템 전용 메시지에 사용됩니다.
midiInClose midiInClose 함수는 지정된 MIDI 입력 디바이스를 닫습니다.
midiInGetDevCaps midiInGetDevCaps 함수(mmeapi.h)는 지정된 MIDI 입력 디바이스의 기능을 결정합니다.
midiInGetDevCapsA midiInGetDevCaps 함수는 지정된 MIDI 입력 디바이스의 기능을 결정합니다. (midiInGetDevCapsA)
midiInGetDevCapsW midiInGetDevCapsW(유니코드) 함수(mmeapi.h)는 지정된 MIDI 입력 디바이스의 기능을 결정합니다.
midiInGetErrorText midiInGetErrorText 함수(mmeapi.h)는 지정된 오류 코드로 식별된 오류에 대한 텍스트 설명을 검색합니다.
midiInGetErrorTextA midiInGetErrorText 함수는 지정된 오류 코드로 식별된 오류에 대한 텍스트 설명을 검색합니다. (midiInGetErrorTextA)
midiInGetErrorTextW

midiInGetErrorTextW(유니코드) 함수(mmeapi.h)는 지정된 오류 코드로 식별된 오류에 대한 텍스트 설명을 검색합니다.
midiInGetID midiInGetID 함수는 지정된 MIDI 입력 디바이스에 대한 디바이스 식별자를 가져옵니다.
midiInGetNumDevs midiInGetNumDevs 함수는 시스템의 MIDI 입력 디바이스 수를 검색합니다.
midiInMessage midiInMessage 함수는 MIDI 디바이스 드라이버에 메시지를 보냅니다.
midiInOpen midiInOpen 함수는 지정된 MIDI 입력 디바이스를 엽니다.
midiInPrepareHeader midiInPrepareHeader 함수는 MIDI 입력을 위한 버퍼를 준비합니다.
midiInReset midiInReset 함수는 지정된 MIDI 입력 디바이스에서 입력을 중지합니다.
midiInStart midiInStart 함수는 지정된 MIDI 입력 디바이스에서 MIDI 입력을 시작합니다.
midiInStop midiInStop 함수는 지정된 MIDI 입력 디바이스에서 MIDI 입력을 중지합니다.
midiInUnprepareHeader midiInUnprepareHeader 함수는 midiInPrepareHeader 함수에서 수행하는 준비를 정리합니다.
midiOutCacheDrumPatches midiOutCacheDrumPatches 함수는 내부 MIDI 신시사이저 디바이스가 지정된 키 기반 타악기 패치 집합을 미리 로드하고 캐시하도록 요청합니다.
midiOutCachePatches midiOutCachePatches 함수는 내부 MIDI 신시사이저 디바이스가 지정된 패치 집합을 미리 로드하고 캐시하도록 요청합니다.

midiOutClose midiOutClose 함수는 지정된 MIDI 출력 디바이스를 닫습니다.
midiOutGetDevCaps midiOutGetDevCaps 함수(mmeapi.h)는 지정된 MIDI 출력 디바이스를 쿼리하여 해당 기능을 확인합니다.
midiOutGetDevCapsA midiOutGetDevCaps 함수는 지정된 MIDI 출력 디바이스를 쿼리하여 해당 기능을 확인합니다. (midiOutGetDevCapsA)
midiOutGetDevCapsW midiOutGetDevCapsW(유니코드) 함수(mmeapi.h)는 지정된 MIDI 출력 디바이스를 쿼리하여 해당 기능을 확인합니다.
midiOutGetErrorText midiOutGetErrorText(mmeapi.h) 함수는 지정된 오류 코드로 식별된 오류에 대한 텍스트 설명을 검색합니다.
midiOutGetErrorTextA midiOutGetErrorText 함수는 지정된 오류 코드로 식별된 오류에 대한 텍스트 설명을 검색합니다. (midiOutGetErrorTextA)
midiOutGetErrorTextW midiOutGetErrorTextW(유니코드) 함수(mmeapi.h)는 지정된 오류 코드로 식별된 오류에 대한 텍스트 설명을 검색합니다.
midiOutGetID midiOutGetID 함수는 지정된 MIDI 출력 디바이스에 대한 디바이스 식별자를 검색합니다.
midiOutGetNumDevs midiOutGetNumDevs 함수는 시스템에 있는 MIDI 출력 디바이스 수를 검색합니다.
midiOutGetVolume midiOutGetVolume 함수는 MIDI 출력 디바이스의 현재 볼륨 설정을 검색합니다.
midiOutLongMsg midiOutLongMsg 함수는 지정된 MIDI 출력 디바이스에 시스템 전용 MIDI 메시지를 보냅니다.

midiOutMessage midiOutMessage 함수는 MIDI 디바이스 드라이버에 메시지를 보냅니다. 이 함수는 MIDI API에서 지원되지 않는 드라이버 관련 메시지에만 사용됩니다.
midiOutOpen midiOutOpen 함수는 재생을 위해 MIDI 출력 디바이스를 엽니다.
midiOutPrepareHeader midiOutPrepareHeader 함수는 출력을 위해 MIDI 시스템 전용 또는 스트림 버퍼를 준비합니다.
midiOutReset midiOutReset 함수는 지정된 MIDI 출력 디바이스에 대한 모든 MIDI 채널의 모든 노트를 해제합니다.
midiOutSetVolume midiOutSetVolume 함수는 MIDI 출력 디바이스의 볼륨을 설정합니다.
midiOutShortMsg midiOutShortMsg 함수는 지정된 MIDI 출력 디바이스에 짧은 MIDI 메시지를 보냅니다.
midiOutUnprepareHeader midiOutUnprepareHeader 함수는 midiOutPrepareHeader 함수에서 수행하는 준비를 정리합니다.
midiStreamClose midiStreamClose 함수는 열린 MIDI 스트림을 닫습니다.
midiStreamOpen midiStreamOpen 함수는 출력을 위해 MIDI 스트림을 엽니다. 기본적으로 디바이스는 일시 중지 모드로 열립니다. 이 함수에서 검색한 스트림 핸들은 스트림에 대한 모든 후속 참조에서 사용해야 합니다.
midiStreamOut midiStreamOut 함수는 MIDI 데이터의 스트림(버퍼)을 MIDI 출력 디바이스로 재생하거나 큐에 대기합니다.
midiStreamPause midiStreamPause 함수는 지정된 MIDI 스트림의 재생을 일시 중지합니다.

midiStreamPosition midiStreamPosition 함수는 MIDI 스트림의 현재 위치를 검색합니다.
midiStreamProperty midiStreamProperty 함수는 MIDI 출력 디바이스와 연결된 MIDI 데이터 스트림의 속성을 설정하거나 검색합니다.
midiStreamRestart midiStreamRestart 함수는 일시 중지된 MIDI 스트림을 다시 시작합니다.
midiStreamStop midiStreamStop 함수는 지정된 MIDI 출력 디바이스에 대한 모든 MIDI 채널의 모든 노트를 해제합니다.
mixerClose mixerClose 함수는 지정된 믹서 디바이스를 닫습니다.
mixerGetControlDetails mixerGetControlDetails 함수(mmeapi.h)는 오디오 라인과 연결된 단일 컨트롤에 대한 세부 정보를 검색합니다.
mixerGetControlDetailsA mixerGetControlDetails 함수는 오디오 라인과 연결된 단일 컨트롤에 대한 세부 정보를 검색합니다. (mixerGetControlDetailsA)
mixerGetControlDetailsW mixerGetControlDetailsW(유니코드) 함수(mmeapi.h)는 오디오 라인과 연결된 단일 컨트롤에 대한 세부 정보를 검색합니다.
mixerGetDevCaps mixerGetDevCaps 함수(mmeapi.h)는 지정된 믹서 디바이스를 쿼리하여 해당 기능을 확인합니다.
mixerGetDevCapsA mixerGetDevCaps 함수는 지정된 믹서 디바이스를 쿼리하여 해당 기능을 확인합니다. (mixerGetDevCapsA)
mixerGetDevCapsW

mixerGetDevCapsW(유니코드) 함수(mmeapi.h)는 지정된 믹서 디바이스를 쿼리하여 해당 기능을 확인합니다.
mixerGetID mixerGetID 함수는 지정된 디바이스 핸들과 연결된 믹서 디바이스에 대한 디바이스 식별자를 검색합니다.
mixerGetLineControls mixerGetLineControls 함수(mmeapi.h)는 오디오 라인과 연결된 하나 이상의 컨트롤을 검색합니다.
mixerGetLineControlsA mixerGetLineControls 함수는 오디오 라인과 연결된 하나 이상의 컨트롤을 검색합니다. (mixerGetLineControlsA)
mixerGetLineControlsW mixerGetLineControlsW(유니코드) 함수는 오디오 라인과 연결된 하나 이상의 컨트롤을 검색합니다. (mixerGetLineControlsW)
mixerGetLineInfo mixerGetLineInfo 함수는 믹서 디바이스의 특정 줄에 대한 정보를 검색하고 함수를 수정하는 매개 변수를 포함합니다.
mixerGetLineInfoA mixerGetLineInfo 함수는 믹서 디바이스의 특정 줄에 대한 정보를 검색합니다. (mixerGetLineInfoA)
mixerGetLineInfoW mixerGetLineInfoW(유니코드) 함수는 믹서 디바이스의 특정 줄에 대한 정보를 검색합니다. (mixerGetLineInfoW)
mixerGetNumDevs mixerGetNumDevs 함수는 시스템에 있는 믹서 디바이스 수를 검색합니다.
mixerMessage mixerMessage 함수는 사용자 지정 믹서 드라이버 메시지를 믹서 드라이버에 직접 보냅니다.
mixerOpen mixerOpen 함수는 지정된 믹서 디바이스를 열고 애플리케이션이 핸들을 닫을 때까지 디바이스가 제거되지 않도록 합니다.

mixerSetControlDetails mixerSetControlDetails 함수는 오디오 라인과 연결된 단일 컨트롤의 속성을 설정합니다.
waveInAddBuffer waveInAddBuffer 함수는 지정된 파형 오디오 입력 디바이스에 입력 버퍼를 보냅니다. 버퍼가 채워지면 애플리케이션에 알림이 표시됩니다.
waveInClose waveInClose 함수는 지정된 파형 오디오 입력 디바이스를 닫습니다.
waveInGetDevCaps waveInGetDevCaps 함수는 지정된 파형 오디오 입력 디바이스의 기능을 검색합니다.
waveInGetErrorText waveInGetErrorText 함수는 지정된 오류 번호로 식별된 오류에 대한 텍스트 설명을 검색합니다.
waveInGetID waveInGetID 함수는 지정된 파형 오디오 입력 디바이스에 대한 디바이스 식별자를 가져옵니다.
waveInGetNumDevs waveInGetNumDevs 함수는 시스템에 있는 파형 오디오 입력 디바이스의 수를 반환합니다.
waveInGetPosition waveInGetPosition은 Windows Vista를 기준으로 더 이상 사용할 수 없습니다.
waveInMessage waveInMessage 함수는 파형 오디오 입력 디바이스 드라이버에 메시지를 보냅니다.
waveInOpen waveInOpen 함수는 녹음을 위해 지정된 파형 오디오 입력 디바이스를 엽니다.
waveInPrepareHeader waveInPrepareHeader 함수는 파형 오디오 입력을 위한 버퍼를 준비합니다.
waveInReset waveInReset 함수는 지정된 파형 오디오 입력 디바이스의 입력을 중지하고 현재 위치를 0으로 다시 설정합니다. 보류 중인 모든 버퍼는 완료된 것으로 표시되고 애플리케이션에 반환됩니다.

waveInStart waveInStart 함수는 지정된 파형 오디오 입력 디바이스에서 입력을 시작합니다.
waveInStop waveInStop 함수는 파형 오디오 입력을 중지합니다.
waveInUnprepareHeader waveInUnprepareHeader 함수는 waveInPrepareHeader 함수에 의해 수행되는 준비를 정리합니다.
waveOutBreakLoop waveOutBreakLoop 함수는 지정된 파형 오디오 출력 디바이스에서 루프를 중단하고 드라이버 목록의 다음 블록으로 재생을 계속할 수 있도록 합니다.
waveOutClose waveOutClose 함수는 지정된 파형 오디오 출력 디바이스를 닫습니다.
waveOutGetDevCaps waveOutGetDevCaps 함수는 지정된 파형 오디오 출력 디바이스의 기능을 검색합니다.
waveOutGetErrorText waveOutGetErrorText 함수는 지정된 오류 번호로 식별된 오류에 대한 텍스트 설명을 검색합니다.
waveOutGetID waveOutGetID 함수는 지정된 파형 오디오 출력 디바이스에 대한 디바이스 식별자를 검색합니다.
waveOutGetNumDevs waveOutGetNumDevs 함수는 시스템에 있는 파형 오디오 출력 디바이스의 수를 검색합니다.
waveOutGetPitch waveOutGetPitch 함수는 지정된 파형 오디오 출력 디바이스에 대한 현재 피치 설정을 검색합니다.
waveOutGetPlaybackRate waveOutGetPlaybackRate 함수는 지정된 파형 오디오 출력 디바이스에 대한 현재 재생 속도를 검색합니다.
waveOutGetPosition waveOutGetPosition 함수는 지정된 파형 오디오 출력 디바이스의 현재 재생 위치를 검색합니다.

waveOutGetVolume waveOutGetVolume 함수는 지정된 파형 오디오 출력 디바이스의 현재 볼륨 수준을 검색합니다.
waveOutMessage waveOutMessage 함수는 파형 오디오 출력 디바이스 드라이버에 메시지를 보냅니다.
waveOutOpen waveOutOpen 함수는 재생을 위해 지정된 파형 오디오 출력 디바이스를 엽니다.
waveOutPause waveOutPause 함수는 지정된 파형 오디오 출력 디바이스에서 재생을 일시 중지합니다. 현재 위치가 저장됩니다. waveOutRestart 함수를 사용하여 현재 위치에서 재생을 다시 시작합니다.
waveOutPrepareHeader waveOutPrepareHeader 함수는 재생을 위해 파형 오디오 데이터 블록을 준비합니다.
waveOutReset waveOutReset 함수는 지정된 파형 오디오 출력 디바이스에서 재생을 중지하고 현재 위치를 0으로 다시 설정합니다. 보류 중인 모든 재생 버퍼는 완료(WHDR_DONE)로 표시되고 애플리케이션에 반환됩니다.
waveOutRestart waveOutRestart 함수는 일시 중지된 파형 오디오 출력 디바이스에서 재생을 다시 시작합니다.
waveOutSetPitch waveOutSetPitch 함수는 지정된 파형 오디오 출력 디바이스의 피치를 설정합니다.
waveOutSetPlaybackRate waveOutSetPlaybackRate 함수는 지정된 파형 오디오 출력 디바이스의 재생 속도를 설정합니다.
waveOutSetVolume waveOutSetVolume 함수는 지정된 파형 오디오 출력 디바이스의 볼륨 수준을 설정합니다.
waveOutUnprepareHeader waveOutUnprepareHeader 함수는 waveOutPrepareHeader 함수에서 수행하는 준비를 정리합니다. 이 함수는 디바이스 드라이버가 데이터 블록으로 완료된 후에 호출되어야 합니다. 버퍼를 해제하기 전에 이 함수를 호출해야 합니다.

waveOutWrite waveOutWrite 함수는 데이터 블록을 지정된 파형 오디오 출력 디바이스로 보냅니다.

구조체

AUXCAPS AUXCAPS 구조는 보조 출력 디바이스의 기능을 설명합니다. (AUXCAPS)
AUXCAPSA AUXCAPS 구조는 보조 출력 디바이스의 기능을 설명합니다. (AUXCAPSA)
AUXCAPSW AUXCAPS 구조는 보조 출력 디바이스의 기능을 설명합니다. (AUXCAPSW)
MIDIEVENT MIDIEVENT 구조체는 스트림 버퍼의 MIDI 이벤트를 설명합니다.
MIDIHDR MIDIHDR 구조체는 MIDI 시스템 배타적 또는 스트림 버퍼를 식별하는 데 사용되는 헤더를 정의합니다.
MIDIINCAPS MIDIINCAPS 구조는 MIDI 입력 디바이스의 기능을 설명합니다. (MIDIINCAPS)
MIDIINCAPSA MIDIINCAPS 구조는 MIDI 입력 디바이스의 기능을 설명합니다. (MIDIINCAPSA)
MIDIINCAPSW MIDIINCAPS 구조는 MIDI 입력 디바이스의 기능을 설명합니다. (MIDIINCAPSW)
MIDIOUTCAPS MIDIOUTCAPS 구조는 MIDI 출력 디바이스의 기능을 설명합니다. (MIDIOUTCAPS)
MIDIOUTCAPSA

MIDIOUTCAPS 구조는 MIDI 출력 디바이스의 기능을 설명합니다. (MIDIOUTCAPSA)
MIDIOUTCAPSW MIDIOUTCAPS 구조는 MIDI 출력 디바이스의 기능을 설명합니다. (MIDIOUTCAPSW)
MIDIPROPTEMPO MIDIPROPTEMPO 구조체에는 스트림에 대한 tempo 속성이 포함됩니다.
MIDIPROPTIMEDIV MIDIPROPTIMEDIV 구조체에는 스트림의 시간 나누기 속성이 포함됩니다.
MIDISTRMBUFFVER MIDISTRMBUFFVER 구조체에는 MEVT_VERSION 형식의 긴 MIDI 이벤트에 대한 버전 정보가 포함되어 있습니다.
MIXERCAPS MIXERCAPS 구조는 믹서 디바이스의 기능을 설명합니다. (MIXERCAPS)
MIXERCAPSA MIXERCAPS 구조는 믹서 디바이스의 기능을 설명합니다. (MIXERCAPSA)
MIXERCAPSW MIXERCAPS 구조는 믹서 디바이스의 기능을 설명합니다. (MIXERCAPSW)
MIXERCONTROL MIXERCONTROL 구조체는 오디오 라인에 대한 단일 컨트롤의 상태 및 메트릭을 설명합니다. (MIXERCONTROL, *PMIXERCONTROL, FAR *PMIXERCONTROL)
MIXERCONTROLA MIXERCONTROL 구조체는 오디오 라인에 대한 단일 컨트롤의 상태 및 메트릭을 설명합니다. (mixercontrola)
MIXERCONTROLDETAILS MIXERCONTROLDETAILS 구조체는 오디오 믹서 컨트롤의 상태 정보를 검색하거나 설정하는 컨트롤 세부 구조체를 나타냅니다. (MIXERCONTROLDETAILS)
MIXERCONTROLDETAILS_BOOLEAN

MIXERCONTROLDETAILS_BOOLEAN 구조체는 오디오 믹서 컨트롤에 대한 부울 컨트롤 속성을 검색하고 설정합니다.
MIXERCONTROLDETAILS_LISTTEXT MIXERCONTROLDETAILS_LISTTEXT 구조체는 다중 항목 컨트롤에 대한 목록 텍스트, 레이블 텍스트 및/또는 대역 범위 정보를 검색합니다.
MIXERCONTROLDETAILS_LISTTEXTA MIXERCONTROLDETAILS 구조체는 오디오 믹서 컨트롤의 상태 정보를 검색하거나 설정하는 컨트롤 세부 구조체를 나타냅니다. (mixercontroldetails_listtexta)
MIXERCONTROLDETAILS_LISTTEXTW MIXERCONTROLDETAILS 구조체는 오디오 믹서 컨트롤의 상태 정보를 검색하거나 설정하는 컨트롤 세부 구조체를 나타냅니다. (mixercontroldetails_listtextw)
MIXERCONTROLDETAILS_SIGNED MIXERCONTROLDETAILS_SIGNED 구조체는 오디오 믹서 컨트롤에 대한 서명된 형식 컨트롤 속성을 검색하고 설정합니다.
MIXERCONTROLDETAILS_UNSIGNED MIXERCONTROLDETAILS_UNSIGNED 구조체는 오디오 믹서 컨트롤에 대한 서명되지 않은 형식 컨트롤 속성을 검색하고 설정합니다.
MIXERCONTROLW MIXERCONTROL 구조체는 오디오 라인에 대한 단일 컨트롤의 상태 및 메트릭을 설명합니다. (mixercontrolw)
MIXERLINE MIXERLINE 구조체는 오디오 라인의 상태 및 메트릭을 설명합니다. (MIXERLINE)
MIXERLINEA MIXERLINE 구조체는 오디오 라인의 상태 및 메트릭을 설명합니다. (MIXERLINEA)
MIXERLINECONTROLS MIXERLINECONTROLS 구조체에는 오디오 라인의 컨트롤에 대한 정보가 포함됩니다. (MIXERLINECONTROLS)
MIXERLINECONTROLSA

MIXERLINECONTROLS 구조체에는 오디오 라인의 컨트롤에 대한 정보가 포함됩니다. (MIXERLINECONTROLSA)
MIXERLINECONTROLSW MIXERLINECONTROLS 구조체에는 오디오 라인의 컨트롤에 대한 정보가 포함됩니다. (MIXERLINECONTROLSW)
MIXERLINEW MIXERLINE 구조체는 오디오 라인의 상태 및 메트릭을 설명합니다. (MIXERLINEW)
PCMWAVEFORMAT PCMWAVEFORMAT 구조는 PCM 파형 오디오 데이터의 데이터 형식을 설명합니다. 이 구조체는 WAVEFORMATEX 구조체로 대체되었습니다.
WAVEFORMAT WAVEFORMAT 구조체는 파형 오디오 데이터의 형식을 설명합니다. 이 구조체는 WAVEFORMATEX 구조체로 대체되었습니다.
WAVEFORMATEX WAVEFORMATEX 구조체는 파형 오디오 데이터의 형식을 정의합니다.
WAVEHDR WAVEHDR 구조체는 파형 오디오 버퍼를 식별하는 데 사용되는 헤더를 정의합니다.
WAVEINCAPS WAVEINCAPS 구조체는 파형 오디오 입력 디바이스의 기능을 설명합니다. (WAVEINCAPS)
WAVEINCAPSA WAVEINCAPS 구조체는 파형 오디오 입력 디바이스의 기능을 설명합니다. (WAVEINCAPSA)
WAVEINCAPSW WAVEINCAPS 구조체는 파형 오디오 입력 디바이스의 기능을 설명합니다. (WAVEINCAPSW)
WAVEOUTCAPS WAVEOUTCAPS 구조체는 파형 오디오 출력 디바이스의 기능을 설명합니다. (WAVEOUTCAPS)
WAVEOUTCAPSA WAVEOUTCAPS 구조체는 파형 오디오 출력 디바이스의 기능을 설명합니다. (WAVEOUTCAPSA)

WAVEOUTCAPSW WAVEOUTCAPS 구조체는 파형 오디오 출력 디바이스의 기능을 설명합니다. (WAVEOUTCAPSW)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mmiscapi.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows 멀티미디어에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 멀티미디어

mmiscapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

CloseDriver 설치 가능한 드라이버를 닫습니다.
DefDriverProc 설치 가능한 드라이버에서 처리되지 않는 모든 메시지에 대한 기본 처리를 제공합니다. 이 함수는 설치 가능한 드라이버의 DriverProc 함수 내에서만 사용됩니다. (DefDriverProc)
DriverCallback 콜백 함수를 호출하거나, 창에 메시지를 보내거나, 스레드 차단을 해제합니다. 작업은 알림 플래그의 값에 따라 달라집니다. 이 함수는 설치 가능한 드라이버의 DriverProc 함수 내에서만 사용됩니다.
DrvDefDriverProc 설치 가능한 드라이버에서 처리되지 않는 모든 메시지에 대한 기본 처리를 제공합니다. 이 함수는 설치 가능한 드라이버의 DriverProc 함수 내에서만 사용됩니다. D
DrvGetModuleHandle 설치 가능한 드라이버가 포함된 모듈의 인스턴스 핸들을 검색합니다. 이 함수는 이전 버전의 Windows와의 호환성을 위해 제공됩니다.
GetDriverModuleHandle 설치 가능한 드라이버가 포함된 모듈의 인스턴스 핸들을 검색합니다.
mmioAdvance mmioAdvance 함수는 mmioGetInfo 함수를 사용하여 직접 I/O 버퍼 액세스를 위해 설정된 파일의 I/O 버퍼를 진행합니다.

mmioAscend mmioAscend 함수는 mmioDescend 함수로 내림차순으로 내림차순으로 또는 mmioCreateChunk 함수를 사용하여 만든 RIFF 파일의 청크에서 오름차순으로 올라갑니다.
mmioClose mmioClose 함수는 mmioOpen 함수를 사용하여 연 파일을 닫습니다.
mmioCreateChunk mmioCreateChunk 함수는 mmioOpen 함수를 사용하여 연 RIFF 파일에 청크를 만듭니다.
mmioDescend mmioDescend 함수는 mmioOpen 함수를 사용하여 연 RIFF 파일의 청크로 내림차순으로 내려갑니다. 지정된 청크를 검색할 수도 있습니다.
mmioFlush mmioFlush 함수는 버퍼가 기록된 경우 파일의 I/O 버퍼를 디스크에 씁니다.
mmioGetInfo mmioGetInfo 함수는 mmioOpen 함수를 사용하여 연 파일에 대한 정보를 검색합니다. 이 정보를 사용하면 파일이 버퍼링된 I/O용으로 열려 있는 경우 애플리케이션이 I/O 버퍼에 직접 액세스할 수 있습니다.
mmioInstallIOProc mmioInstallIOProc 함수는 사용자 지정 I/O 프로시저를 설치하거나 제거하고 해당 4자 코드를 사용하여 설치된 I/O 프로시저를 찾습니다.
mmioInstallIOProcA mmioInstallIOProc 함수는 사용자 지정 I/O 프로시저를 설치하거나 제거합니다. 또한 이 함수는 해당 4자 코드를 사용하여 설치된 I/O 프로시저를 찾습니다. (mmioInstallIOProcA)
mmioInstallIOProcW mmioInstallIOProcW(유니코드) 함수는 사용자 지정 I/O 프로시저를 설치하거나 제거하고 해당 코드를 사용하여 설치된 I/O 프로시저를 찾습니다.
mmioOpen mmioOpen 함수는 버퍼링되지 않거나 버퍼링된 I/O에 대한 파일을 엽니다. 파일을 만듭니다. 는 파일을 삭제합니다. 또는 파일이 있는지 여부를 확인합니다.

mmioOpenA mmioOpen 함수는 버퍼링되지 않거나 버퍼링된 I/O에 대한 파일을 엽니다. 파일을 만듭니다. 는 파일을 삭제합니다. 또는 파일이 있는지 여부를 확인합니다. (mmioOpenA)
mmioOpenW mmioOpenW(유니코드) 함수는 버퍼링되지 않거나 버퍼링된 I/O에 대한 파일을 엽니다. 파일을 만듭니다. 는 파일을 삭제합니다. 또는 파일이 있는지 여부를 확인합니다. (mmioOpenW)
mmioRead mmioRead 함수는 mmioOpen 함수를 사용하여 연 파일에서 지정된 바이트 수를 읽습니다.
mmioRename mmioRename 함수는 지정된 파일의 이름을 바꾸고 파일 이름을 포함하는 문자열을 수정하는 매개 변수를 포함합니다.
mmioRenameA mmioRename 함수는 지정된 파일의 이름을 바꿉니다. (mmioRenameA)
mmioRenameW mmioRenameW(유니코드) 함수는 지정된 파일의 이름을 바꾸고 파일 이름을 포함하는 문자열을 수정하는 매개 변수를 포함합니다. (mmioRenameW)
mmioSeek mmioSeek 함수는 mmioOpen 함수를 사용하여 연 파일의 현재 파일 위치를 변경합니다.
mmioSendMessage mmioSendMessage 함수는 지정된 파일과 연결된 I/O 프로시저에 메시지를 보냅니다.
mmioSetBuffer mmioSetBuffer 함수는 버퍼링된 I/O를 사용하거나 사용하지 않도록 설정하거나 mmioOpen 함수를 사용하여 연 파일의 버퍼 또는 버퍼 크기를 변경합니다.
mmioSetInfo mmioSetInfo 함수는 mmioGetInfo 함수에서 mmioOpen 함수를 사용하여 연 파일에 대해 검색한 정보를 업데이트합니다. 이 함수를 사용하여 버퍼링된 I/O용으로 열린 파일의 직접 버퍼 액세스를 종료합니다.

mmioStringToFOURCC mmioStringToFOURCC 함수는 null로 끝나는 문자열을 4자 코드로 변환하고 변환을 수정하는 매개 변수를 포함합니다. (mmioStringToFOURCCW)
mmioStringToFOURCCA mmioStringToFOURCC 함수는 null로 끝나는 문자열을 4자 코드로 변환합니다. (mmioStringToFOURCCA)
mmioStringToFOURCCW mmioStringToFOURCCW(유니코드) 함수는 null로 끝나는 문자열을 4자 코드로 변환합니다. (mmioStringToFOURCCW)
mmioWrite mmioWrite 함수는 mmioOpen 함수를 사용하여 연 파일에 지정된 바이트 수를 씁니다.
OpenDriver 설치 가능한 드라이버의 인스턴스를 열고 드라이버의 기본 설정 또는 드라이버별 값을 사용하여 인스턴스를 초기화합니다.
SendDriverMessage 설치 가능한 드라이버에 지정된 메시지를 보냅니다.
sndOpenSound 지정된 사운드 이벤트를 엽니다.

콜백 함수

DRIVERPROC 설치 가능한 드라이버에 대한 드라이버 메시지를 처리합니다. DriverProc은 드라이버 제공 함수입니다.
MMIOPROC MMIOProc 함수는 mmioInstallIOProc 함수에 의해 설치된 사용자 지정 입력/출력(I/O) 프로시저입니다.

구조체

DRVCONFIGINFO 설치 가능한 드라이버와 연결된 레지스트리 키 및 값 이름을 포함합니다.
MMCKINFO MMCKINFO 구조체는 RIFF 파일의 청크에 대한 정보를 포함합니다.

mmreg.h 헤더

아티클2023. 08. 24.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 오디오 디바이스 DDI 참조
- Directshow
- Windows 멀티미디어

mmreg.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

HEAACWAVEFORMAT AudioSpecificConfig() 데이터를 포함하는 AAC 또는 HE-AAC 스트림에 대한 형식 데이터를 포함합니다.
HEAACWAVEINFO AAC(고급 오디오 코딩) 또는 High-Efficiency HE-AAC(Advanced Audio Coding) 스트림에 대한 형식 데이터를 포함합니다.
MPEG1WAVEFORMAT MPEG1WAVEFORMAT 구조체는 MPEG-1 오디오 데이터의 형식을 설명합니다.
MPEGLAYER3WAVEFORMAT MPEGLAYER3WAVEFORMAT 구조는 MP3(MPEG 오디오 계층 3) 오디오 형식을 설명합니다.
PCMWAVEFORMAT PCMWAVEFORMAT 구조체는 PCM 파형 오디오 데이터의 데이터 형식을 설명합니다. 이 구조체는 WAVEFORMATEX 구조체로 대체됩니다.
WAVEFILTER WAVEFILTER 구조체는 파형 오디오 데이터에 대한 필터를 정의합니다.
WAVEFORMAT WAVEFORMAT 구조체는 파형 오디오 데이터의 형식을 설명합니다. 이 구조체는 WAVEFORMATEX 구조체로 대체됩니다.

WAVEFORMATEX WAVEFORMATEX 구조체는 웨이브 오디오 스트림의 데이터 형식을 지정합니다.
WAVEFORMATEXTENSIBLE WAVEFORMATEXTENSIBLE 구조체는 WAVEFORMATEX에서 허용하는 것보다 두 개 이상의 채널 이상의 샘플 해상도를 갖는 형식에 대해 파형 오디오 데이터의 형식을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mmsystem.h 헤더

아티클2023. 08. 24.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows GDI
- Windows 멀티미디어

mmsystem.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

디빈덱스 DIBINDEX 매크로는 인덱스를 DIB 색 테이블의 항목으로 가져오고 지정된 인덱스와 연결된 색을 지정하는 COLORREF 값을 반환합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mobsync.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

mobsync.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ISyncMgrEnumItems SYNCMGRITEM 구조의 배열을 통해 열거하는 메서드를 노출합니다.
ISyncMgrRegister 애플리케이션이 동기화 관리자에 등록할 수 있도록 메서드를 노출합니다. ISyncMgrRegister 인터페이스를 통해 또는 레지스트리에 직접 등록하여 이 작업을 수행할 수 있습니다.
ISyncMgrSynchronize 등록된 애플리케이션 또는 서비스가 동기화 관리자로부터 알림을 받을 수 있도록 하는 메서드를 노출합니다.
ISyncMgrSynchronizeCallback 동기화 프로세스를 관리하는 메서드를 노출합니다.
ISyncMgrSynchronizeInvoke 등록된 애플리케이션이 동기화 관리자를 호출하여 항목을 업데이트할 수 있도록 하는 메서드를 노출합니다.

구조체

SYNCMGRHANDLERINFO ISyncMgrSynchronize::GetHandlerInfo 메서드에서 사용할 처리기에 대한 정보를 제공합니다.

SYNCMGRITEM ISyncMgrEnumItems 인터페이스에서 열거되는 항목에 대한 정보를 제공합니다.
SYNCMGRLOGERRORINFO ISyncMgrSynchronizeCallback::LogError 메서드에서 사용할 오류 정보를 제공합니다.
SYNCMGRPROGRESSITEM 동기화가 진행되는 동안 상태 정보를 제공합니다. 이 구조는 ISyncMgrSynchronizeCallback::Progress 메서드와 함께 사용되며 단일 동기화 항목에 해당합니다.

열거형

SYNCMGRFLAG SYNCMGRFLAG 열거형 값은 동기화 이벤트가 시작된 방법을 나타내기 위해 ISyncMgrSynchronize::Initialize 메서드에 사용됩니다.
SYNCMGRHANDLERFLAGS SYNCMGRHANDLERINFO 구조체에서 현재 처리기에 적용되는 플래그로 사용됩니다.
SYNCMGRINVOKEFLAGS SYNCMGRINVOKEFLAGS 열거형 값은 ISyncMgrSynchronizeInvoke::UpdateItems 메서드에서 동기화 관리자를 호출하는 방법을 지정합니다.
SYNCMGRITEMFLAGS SYNCMGRITEM 구조체의 현재 항목에 대한 정보를 지정합니다.
SYNCMGRLOGLEVEL SYNCMGRLOGLEVEL 열거형 값은 ISyncMgrSynchronizeCallback::LogError 메서드에서 사용할 오류 수준을 지정합니다.
SYNCMGRREGISTERFLAGS SYNCMGRREGISTERFLAGS 열거형 값은 ISyncMgrRegister 인터페이스의 메서드에서 처리기가 알림을 받도록 등록된 이벤트를 식별하는 데 사용됩니다.

SYNCMGRSTATUS ISyncMgrSynchronize::SetItemStatus 메서드에서 항목의 업데이트된 상태를 지정하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mpconfig.h 헤더

아티클2023. 08. 24.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Directshow
- Hyper-V WMI 공급자

mpconfig.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IMixerPinConfig IMixerPinConfig 인터페이스는 오버레이 믹서 필터의 입력 핀에 노출되며 다양한 방법으로 비디오 스트림을 조작하는 메서드를 포함합니다.
IMixerPinConfig2 IMixerPinConfig2 인터페이스는 오버레이 믹서의 입력 핀에 노출되며 VGA 칩이 지원하는 경우 비디오 색 컨트롤을 조작하는 메서드를 포함합니다. 이 인터페이스는 IMixerPinConfig 인터페이스에서 파생됩니다. 애플리케이션은 이 인터페이스를 사용하여 여러 비디오 스트림을 혼합할 때 비디오 색 컨트롤을 가져와서 설정합니다.

열거형

AM_ASPECT_RATIO_MODE 디스플레이 창에서 비디오 이미지의 가로 세로 비율을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mprapi.h 헤더

아티클2023. 03. 16.

이 헤더는 라우팅 및 원격 액세스 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 라우팅 및 원격 액세스 서비스

mprapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

MprAdminAcceptNewConnection 원격 액세스 서비스는 새 사용자가 전화를 걸어 RAS 인증을 성공적으로 완료할 때마다 MprAdminAcceptNewConnection 함수를 호출합니다. MprAdminAcceptNewConnection은 사용자가 연결할 수 있는지 여부를 결정합니다.
MprAdminAcceptNewConnection2 원격 액세스 서비스는 새 사용자가 전화를 걸어 RAS 인증을 성공적으로 완료할 때마다 MprAdminAcceptNewConnection2 함수를 호출합니다. MprAdminAcceptNewConnection2는 사용자가 연결할 수 있는지 여부를 결정합니다.
MprAdminAcceptNewConnection3 원격 액세스 서비스는 새 사용자가 전화를 걸어 RAS 인증을 성공적으로 완료할 때마다 MprAdminAcceptNewConnection3 함수를 호출합니다. MprAdminAcceptNewConnection3은 사용자가 연결할 수 있는지 여부를 결정합니다.
MprAdminAcceptNewConnectionEx RAS(원격 액세스 서비스)는 새 사용자가 전화를 걸어 RAS 인증을 성공적으로 완료할 때마다 MprAdminAcceptNewConnectionEx 함수를 호출합니다. MprAdminAcceptNewConnectionEx는 사용자가 연결할 수 있는지 여부를 결정합니다.
MprAdminAcceptNewLink RAS(원격 액세스 서비스)는 특정 연결에 대한 링크를 만들 때마다 MprAdminAcceptNewLink 함수를 호출합니다.

MprAdminAcceptReauthentication 원격 액세스 서비스는 클라이언트의 격리 상태가 변경 될 때마다 MprAdminAcceptReauthentication 함수를 호출합니다.
MprAdminAcceptReauthenticationEx RAS(원격 액세스 서비스)는 클라이언트의 격리 상태가 변경 될 때마다 MprAdminAcceptReauthenticationEx 함수를 호출합니다.
MprAdminBufferFree MprAdminBufferFree 함수는 _MprAdminDeviceEnum, MprAdminInterfaceGetInfo, MprAdminInterfaceDeviceGetInfo, MprAdminInterfaceEnum, MprAdminServerGetInfo, MprAdminInterfaceTransportGetInfo 및 MprAdminTransportGetInfo에서 반환된 메모리 버퍼를 해제합니다.
MprAdminConnectionClearStats MprAdminConnectionClearStats 함수는 지정된 연결에 대한 통계 카운터를 다시 설정합니다.
MprAdminConnectionEnum MprAdminConnectionEnum 함수는 모든 활성 연결을 열거합니다.
MprAdminConnectionEnumEx MprAdminConnectionEnumEx 함수는 지정된 RRAS 서버에 대한 활성 연결을 열거합니다.
MprAdminConnectionGetInfo 특정 연결에 대한 데이터를 검색합니다.
MprAdminConnectionGetInfoEx 지정된 RRAS 서버의 특정 연결에 대한 연결 정보를 검색합니다.
MprAdminConnectionHangupNotification 원격 액세스 서비스는 지정된 연결에 대한 마지막 링크가 해체된 후 MprAdminConnectionHangupNotification 함수를 호출합니다.
MprAdminConnectionHangupNotification2 원격 액세스 서비스는 지정된 연결에 대한 마지막 링크가 해체된 후 MprAdminConnectionHangupNotification2 함수를 호출합니다.

MprAdminConnectionHangupNotification3 원격 액세스 서비스는 지정된 연결에 대한 마지막 링크가 해체된 후 MprAdminConnectionHangupNotification3 함수를 호출합니다.
MprAdminConnectionHangupNotificationEx RAS(원격 액세스 서비스)는 지정된 연결에 대한 마지막 링크가 해체된 후 MprAdminConnectionHangupNotificationEx 함수를 호출합니다.
MprAdminConnectionRemoveQuarantine 필터가 IAS(인터넷 인증 서비스) 정책의 결과로 적용된 경우 MprAdminConnectionRemoveQuarantine 함수는 전화 접속된 RAS 클라이언트에서 격리 필터를 제거합니다.
MprAdminDeregisterConnectionNotification MprAdminDeregisterConnectionNotification 함수는 MprAdminRegisterConnectionNotification을 사용하여 이전에 등록된 이벤트 개체의 등록을 취소합니다. 등록을 취소하면 인터페이스가 연결되거나 연결이 끊어지면 이 이벤트가 더 이상 신호를 받지 않습니다.
MprAdminDeviceEnum 함수는 이름과 형식을 반환할 수 있는 컴퓨터에 설치된 RAS 지원 디바이스를 열거하기 위해 호출됩니다.
MprAdminEstablishDomainRasServer MprAdminEstablishDomainRasServer 함수는 지정된 컴퓨터를 도메인의 원격 액세스 서버로 설정합니다. 이 함수는 도메인에 가입된 컴퓨터에서만 실행되어야 합니다.
MprAdminGetErrorString MprAdminGetErrorString 함수는 Mprerror.h의 라우터 오류와 연결된 문자열을 반환합니다.
MprAdminGetIpAddressForUser RAS는 IP 주소가 필요한 각 사용자에 대해 MprAdminGetIpAddressForUser 함수를 한 번 호출합니다.
MprAdminGetIpv6AddressForUser RAS는 IPv6 주소가 필요한 각 사용자에 대해 MprAdminGetIpv6AddressForUser 함수를 한 번 호출합니다.

MprAdminGetPDCServer MprAdminGetPDCServer 함수는 도메인 이름 또는 서버 이름에서 마스터 UAS(사용자 계정 하위 시스템)가 있는 서버의 이름을 검색합니다. 도메인 이름 매개 변수 또는 서버 이름 매개 변수는 NULL일 수 있지만 둘 다일 수는 없습니다.
MprAdminInitializeDll RRAS(라우팅 및 원격 액세스 서비스)가 시작되면 관리 DLL에서 내보낸 MprAdminInitializeDll 함수를 호출합니다. 이 함수를 사용하여 DLL에 필요한 초기화를 수행합니다.
MprAdminInitializeDllEx RRAS(라우팅 및 원격 액세스 서비스)가 시작되면 관리 DLL에서 내보낸 MprAdminInitializeDll 함수를 호출합니다.
MprAdminInterfaceConnect MprAdminInterfaceConnect 함수는 지정된 WAN 인터페이스에 대한 연결을 만듭니다.
MprAdminInterfaceCreate MprAdminInterfaceCreate 함수는 지정된 서버에 인터페이스를 만듭니다.
MprAdminInterfaceDelete MprAdminInterfaceDelete 함수는 지정된 서버에서 인터페이스를 삭제합니다.
MprAdminInterfaceDeviceGetInfo MprAdminInterfaceDeviceGetInfo는 라우터 수요 다이얼 인터페이스에 사용되는 디바이스에 대한 정보를 검색합니다.
MprAdminInterfaceDeviceSetInfo MprAdminInterfaceDeviceSetInfo는 라우터 수요 다이얼 인터페이스에 사용되는 디바이스를 만들거나 수정합니다.
MprAdminInterfaceDisconnect MprAdminInterfaceDisconnect 함수는 연결된 WAN 인터페이스의 연결을 끊습니다.
MprAdminInterfaceEnum MprAdminInterfaceEnum 함수는 지정된 서버의 모든 인터페이스를 열거합니다.

MprAdminInterfaceGetCredentials MprAdminInterfaceGetCredentials 함수를 사용하여 지정된 요청 다이얼 인터페이스에서 전화를 걸기 위한 도메인, 사용자 이름 및 암호를 검색합니다.
MprAdminInterfaceGetCredentialsEx MprAdminInterfaceGetCredentialsEx 함수를 사용하여 지정된 인터페이스에 대한 확장 자격 증명 정보를 검색합니다. 이 함수를 사용하여 EAP(Extensible Authentication Protocols)에 사용되는 자격 증명 정보를 검색합니다.
MprAdminInterfaceGetCustomInfoEx 지정된 서버에서 지정된 수요 다이얼 인터페이스에 대한 터널별 구성을 검색합니다.
MprAdminInterfaceGetHandle MprAdminInterfaceGetHandle 함수는 지정된 인터페이스에 대한 핸들을 검색합니다.
MprAdminInterfaceGetInfo MprAdminInterfaceGetInfo 함수는 지정된 서버에서 지정된 인터페이스에 대한 정보를 검색합니다.
MprAdminInterfaceQueryUpdateResult MprAdminInterfaceQueryUpdateResult 함수는 지정된 라우터 관리자에게 마지막 요청의 결과를 반환하여 인터페이스에 대한 경로를 업데이트합니다. 자세한 내용은 MprAdminInterfaceUpdateRoutes를 참조하세요.
MprAdminInterfaceSetCredentials MprAdminInterfaceSetCredentials 함수를 사용하여 지정된 수요 다이얼 인터페이스에서 전화 걸기에 사용할 도메인, 사용자 이름 및 암호를 설정합니다.
MprAdminInterfaceSetCredentialsEx MprAdminInterfaceSetCredentialsEx 함수를 사용하여 인터페이스에 대한 확장 자격 증명 정보를 설정합니다. 이 함수를 사용하여 EAP(Extensible Authentication Protocols)에 사용되는 자격 증명 정보를 설정합니다.
MprAdminInterfaceSetCustomInfoEx 지정된 서버의 지정된 수요 다이얼 인터페이스에 대한 터널별 사용자 지정 구성을 설정합니다.
MprAdminInterfaceSetInfo MprAdminInterfaceSetInfo 함수는 지정된 서버의 지정된 인터페이스에 대한 정보를 설정합니다.

MprAdminInterfaceTransportAdd MprAdminInterfaceTransportAdd 함수는 지정된 인터페이스에 전송(예: IP 또는 IPX)을 추가합니다.
MprAdminInterfaceTransportGetInfo MprAdminInterfaceTransportGetInfo 함수는 지정된 인터페이스에서 실행되는 전송에 대한 정보를 검색합니다.
MprAdminInterfaceTransportRemove MprAdminInterfaceTransportRemove 함수는 지정된 인터페이스에서 전송(예: IP 또는 IPX)을 제거합니다.
MprAdminInterfaceTransportSetInfo MprAdminInterfaceTransportSetInfo 함수는 지정된 인터페이스에서 실행되는 전송에 대한 정보를 설정합니다.
MprAdminInterfaceUpdatePhonebookInfo MprAdminInterfaceUpdatePhonebookInfo 함수는 라우터가 지정된 수요 다이얼 인터페이스에서 변경한 내용을 선택하도록 강제합니다. 요청 전화 접속 인터페이스에 대한 전화 번호부 항목이 변경된 후 이 함수를 호출합니다.
MprAdminInterfaceUpdateRoutes MprAdminInterfaceUpdateRoutes 함수는 지정된 라우터 관리자에게 지정된 인터페이스에 대한 라우팅 정보를 업데이트하도록 요청합니다.
MprAdminIsDomainRasServer MprAdminIsDomainRasServer 함수는 지정된 컴퓨터가 도메인의 원격 액세스 서버로 등록되었는지 여부에 대한 정보를 반환합니다.
MprAdminIsServiceInitialized MprAdminIsServiceInitialized 함수는 호출 프로세스에 액세스 권한이 있는 경우 RRAS 서비스가 지정된 서버에서 실행 중인지 여부를 확인합니다.
MprAdminIsServiceRunning MprAdminIsServiceRunning 함수는 호출 프로세스에 액세스 권한이 있는 경우 RRAS 서비스가 지정된 서버에서 실행 중인지 여부를 확인합니다.
MprAdminLinkHangupNotification RAS는 특정 연결에 대한 링크가 해체될 때마다 MprAdminLinkHangupNotification 함수를 호출합니다.

MprAdminMIBBufferFree MprAdminMIBBufferFree 함수는 MprAdminMIBEntryGet, MprAdminMIBEntryGetFirst, MprAdminMIBEntryGetNext 함수에서 반환된 버퍼를 해제합니다.
MprAdminMIBEntryCreate MprAdminMIBEntryCreate 함수는 라우팅 프로토콜 또는 라우터 관리자가 내보낸 변수 중 하나에 대한 항목을 만듭니다.
MprAdminMIBEntryDelete MprAdminMIBEntryDelete 함수는 라우팅 프로토콜 또는 라우터 관리자가 내보낸 변수 중 하나에 대한 항목을 삭제합니다.
MprAdminMIBEntryGet MprAdminMIBEntryGet 함수는 라우팅 프로토콜 또는 라우터 관리자가 내보낸 변수 중 하나의 값을 검색합니다.
MprAdminMIBEntryGetFirst MprAdminMIBEntryGetFirst 함수는 프로토콜 또는 라우터 관리자가 내보낸 일부 변수 집합의 첫 번째 변수를 검색합니다. 호출을 서비스하는 모듈이 먼저 정의합니다.
MprAdminMIBEntryGetNext MprAdminMIBEntryGetNext 함수는 프로토콜 또는 라우터 관리자가 내보낸 일부 변수 집합의 다음 변수를 검색합니다. 호출을 서비스하는 모듈은 다음에 정의합니다.
MprAdminMIBEntrySet MprAdminMIBEntrySet 함수는 라우팅 프로토콜 또는 라우터 관리자가 내보낸 변수 중 하나의 값을 설정합니다.
MprAdminMIBGetTrapInfo MprAdminMIBGetTrapInfo 함수는 트랩에 대한 자세한 정보를 위해 트랩 이벤트를 설정하는 모듈을 쿼리합니다.
MprAdminMIBServerConnect MprAdminMIBServerConnect 함수는 관리되는 라우터에 대한 연결을 설정합니다. 이 호출은 서버에 대한 다른 호출 전에 수행해야 합니다. 이 함수에서 반환된 핸들은 후속 MIB 호출에 사용됩니다.
MprAdminMIBServerDisconnect MprAdminMIBServerDisconnect 함수는 MprAdminMIBServerConnect에 대한 이전 호출로 인한 연결의 연결을 끊습니다.

MprAdminMIBSetTrapInfo MprAdminMIBSetTrapInfo 함수는 TRAP을 발급해야 할 때마다 신호를 수신하는 이벤트에 대한 핸들을 지정합니다.
MprAdminPortClearStats MprAdminPortClearStats 함수는 지정된 포트에 대한 통계를 다시 설정합니다.
MprAdminPortDisconnect MprAdminPortDisconnect 함수는 특정 포트에서 연결을 끊습니다.
MprAdminPortEnum 특정 연결의 모든 활성 포트 또는 RAS에서 사용하거나 현재 사용할 수 있는 모든 포트를 열거합니다.
MprAdminPortGetInfo MprAdminPortGetInfo 함수는 특정 포트에 대한 정보를 가져옵니다.
MprAdminPortReset MprAdminPortReset 함수는 지정된 포트에 연결된 통신 디바이스를 다시 설정합니다.
MprAdminRegisterConnectionNotification MprAdminRegisterConnectionNotification 함수는 DDM(Demand Dial Manager)에 이벤트 개체를 등록하므로 인터페이스가 연결되거나 연결이 끊어지면 이벤트가 신호를 보냅니다.
MprAdminReleaseIpAddress MprAdminReleaseIpAddress 함수는 사용자가 연결을 끊고 사용자의 IP 주소가 해제될 때 호출됩니다.
MprAdminReleaseIpv6AddressForUser MprAdminReleaseIpv6AddressForUser 함수는 IPv6 주소를 해제해야 하는 각 사용자에 대해 한 번 호출됩니다.
MprAdminSendUserMessage MprAdminSendUserMessage 함수는 지정된 연결에 연결된 사용자에게 메시지를 보냅니다.
MprAdminServerConnect MprAdminServerConnect 함수는 해당 라우터를 관리하기 위해 라우터에 대한 연결을 설정합니다.

MprAdminServerDisconnect MprAdminServerDisconnect 함수는 MprAdminServerConnect에 대한 이전 호출로 인한 연결의 연결을 끊습니다.
MprAdminServerGetCredentials MprAdminServerGetCredentials 함수는 지정된 서버에 대한 미리 공유된 키를 검색합니다.
MprAdminServerGetInfo MprAdminServerGetInfo 함수는 지정된 RRAS 서버에 대한 정보를 검색합니다.
MprAdminServerGetInfoEx MprAdminServerGetInfoEx 함수는 지정된 RRAS 서버에 대한 포트 정보를 검색합니다.
MprAdminServerSetCredentials MprAdminServerSetCredentials 함수는 지정된 서버에 대한 미리 공유된 키를 설정합니다.
MprAdminServerSetInfo MprAdminServerSetInfo 함수는 RRAS 서비스가 실행 중일 때 L2TP, PPTP 및 SSTP 디바이스에 대한 포트 수를 설정하는 데 사용됩니다.
MprAdminServerSetInfoEx MprAdminServerSetInfoEx 함수는 지정된 RRAS 서버에서 포트 정보를 설정합니다.
MprAdminTerminateDll RAS가 종료되면 관리 DLL에서 내보낸 MprAdminTerminateDll 함수를 호출합니다. 이 함수를 사용하여 DLL에 필요한 정리를 수행합니다.
MprAdminTransportCreate MprAdminTransportCreate 함수는 새 전송을 로드하고 전송에 대한 라우터 관리자를 시작합니다.
MprAdminTransportGetInfo MprAdminTransportGetInfo 함수는 지정된 전송에 대한 전역 정보, 기본 클라이언트 인터페이스 정보 또는 둘 다를 검색합니다.
MprAdminTransportSetInfo MprAdminTransportSetInfo 함수는 지정된 전송에 대해 전역 정보 또는 기본 클라이언트 인터페이스 정보 또는 둘 다를 설정합니다.

MprAdminUserGetInfo MprAdminUserGetInfo 함수는 특정 사용자에 대한 모든 RAS 정보를 검색합니다.
MprAdminUserSetInfo MprAdminUserSetInfo 함수는 지정된 사용자에 대한 RAS 정보를 설정합니다.
MprConfigBufferFree MprConfigBufferFree 함수는 버퍼를 해제합니다. MprConfigXEnum, MprConfigXGetInfo
MprConfigFilterGetInfo 지정된 전송 프로토콜 형식에 대한 정적 필터링 정보를 반환합니다.
MprConfigFilterSetInfo 지정된 전송 프로토콜 형식에 대한 정적 필터링 정보를 설정합니다.
MprConfigGetFriendlyName MprConfigGetFriendlyName 함수는 지정된 GUID 이름에 해당하는 인터페이스의 이름을 반환합니다.
MprConfigGetGuidName MprConfigGetGuidName 함수는 지정된 식별 이름에 해당하는 인터페이스의 GUID 이름을 반환합니다.
MprConfigInterfaceCreate MprConfigInterfaceCreate 함수는 지정된 라우터 구성에서 라우터 인터페이스를 만듭니다.
MprConfigInterfaceDelete MprConfigInterfaceDelete 함수는 라우터 구성에서 라우터 인터페이스를 제거합니다. 이 인터페이스와 연결된 모든 전송 정보도 제거됩니다.
MprConfigInterfaceEnum MprConfigInterfaceEnum 함수는 라우터에 대해 구성된 인터페이스를 열거합니다.
MprConfigInterfaceGetCustomInfoEx 지정된 인터페이스에 대한 사용자 지정 IKEv2 정책 구성을 검색합니다.

MprConfigInterfaceGetHandle MprConfigInterfaceGetHandle 함수는 지정된 라우터 구성에서 지정된 인터페이스의 구성에 대한 핸들을 검색합니다.
MprConfigInterfaceGetInfo MprConfigInterfaceGetInfo 함수는 라우터에서 지정된 인터페이스에 대한 구성을 검색합니다.
MprConfigInterfaceSetCustomInfoEx 지정된 인터페이스에 대한 사용자 지정 IKEv2 정책 구성을 설정합니다.
MprConfigInterfaceSetInfo MprConfigInterfaceSetInfo 함수는 지정된 인터페이스에 대한 구성을 설정합니다.
MprConfigInterfaceTransportAdd MprConfigInterfaceTransportAdd 함수는 라우터의 인터페이스 구성에 전송 프로토콜을 추가합니다.
MprConfigInterfaceTransportEnum MprConfigInterfaceTransportEnum 함수는 지정된 인터페이스에 구성된 전송을 열거합니다.
MprConfigInterfaceTransportGetHandle MprConfigInterfaceTransportGetHandle 함수는 지정된 라우터 구성에서 인터페이스의 전송 구성에 대한 핸들을 검색합니다.
MprConfigInterfaceTransportGetInfo MprConfigInterfaceTransportGetInfo 함수는 지정된 인터페이스에서 지정된 클라이언트에 대한 구성 정보를 검색합니다.
MprConfigInterfaceTransportRemove MprConfigInterfaceTransportRemove 함수는 라우터의 지정된 인터페이스 구성에서 지정된 전송을 제거합니다.
MprConfigInterfaceTransportSetInfo MprConfigInterfaceTransportSetInfo 함수는 지정된 인터페이스 및 전송 프로토콜에서 클라이언트에 대한 구성 정보를 업데이트합니다.

MprConfigServerBackup MprConfigServerBackup 함수는 라우터에 대한 라우터 관리자, 인터페이스 및 전화 번호부 구성의 백업을 만듭니다.
MprConfigServerConnect MprConfigServerConnect 함수는 구성할 라우터에 연결합니다.
MprConfigServerDisconnect MprConfigServerDisconnect 함수는 MprConfigServerConnect에 대한 이전 호출로 인한 연결의 연결을 끊습니다.
MprConfigServerGetInfo MprConfigServerGetInfo 함수는 지정된 라우터에 대한 서버 수준 구성 정보를 검색합니다.
MprConfigServerGetInfoEx MprConfigServerGetInfoEx 함수는 지정된 서버에 대한 포트 정보를 검색합니다.
MprConfigServerInstall MprConfigServerInstall 함수는 라우팅 및 원격 액세스 서비스를 기본 구성으로 구성합니다.
MprConfigServerRestore MprConfigServerRestore 함수는 MprConfigServerBackup에 대한 이전 호출에서 만든 백업에서 라우터 관리자, 인터페이스 및 전화 번호부 구성을 복원합니다.
MprConfigServerSetInfo MprConfigServerSetInfo 함수는 L2TP, PPTP 및 SSTP 포트에 대한 포트 수를 설정하고 RRAS 서비스가 실행되지 않을 때 레지스트리에서 RRAS를 사용하거나 사용하지 않도록 설정하여 다음에 시스템을 다시 시작할 때 선택할 수 있도록 하는 데 사용됩니다.
MprConfigServerSetInfoEx MprConfigServerSetInfoEx 함수는 지정된 RRAS 서버에서 포트 정보를 설정합니다.
MprConfigTransportCreate MprConfigTransportCreate 함수는 지정된 전송을 지정된 라우터 구성에 있는 전송 프로토콜 목록에 추가합니다.

MprConfigTransportDelete MprConfigTransportDelete 함수는 지정된 라우터 구성에 있는 전송 목록에서 지정된 전송을 제거합니다.
MprConfigTransportEnum MprConfigTransportEnum 함수는 라우터에 구성된 전송을 열거합니다.
MprConfigTransportGetHandle MprConfigTransportGetHandle 함수는 지정된 라우터 구성에서 지정된 전송 프로토콜의 구성에 대한 핸들을 검색합니다.
MprConfigTransportGetInfo MprConfigTransportGetInfo 함수는 라우터에서 지정된 전송 프로토콜에 대한 구성을 검색합니다.
MprConfigTransportSetInfo MprConfigTransportSetInfo 함수는 지정된 라우터 구성에서 지정된 전송 프로토콜에 대한 구성을 변경합니다.
MprInfoBlockAdd MprInfoBlockAdd 함수는 새 블록을 추가하여 기존 헤더와 동일한 새 헤더를 만듭니다.
MprInfoBlockFind MprInfoBlockFind 함수는 정보 헤더에서 지정된 블록을 찾고 블록에 대한 정보를 검색합니다.
MprInfoBlockQuerySize MprInfoBlockQuerySize 함수는 정보 헤더의 크기를 반환합니다.
MprInfoBlockRemove MprInfoBlockRemove 함수는 지정된 블록이 제거된 기존 헤더와 동일한 새 헤더를 만듭니다.
MprInfoBlockSet MprInfoBlockSet은 지정된 블록이 수정된 기존 헤더와 동일한 새 헤더를 만듭니다.
MprInfoCreate MprInfoCreate 함수는 새 정보 헤더를 만듭니다.

MprInfoDelete MprInfoDelete 함수는 MprInfoCreate를 사용하여 만들거나 MprInfoBlockAdd, MprInfoBlockRemove 또는 MprInfoBlockSet에서 검색한 정보 헤더를 삭제합니다.
MprInfoDuplicate MprInfoDuplicate 함수는 기존 정보 헤더를 복제합니다.
MprInfoRemoveAll MprInfoRemoveAll 함수는 지정된 헤더에서 모든 정보 블록을 제거합니다.

구조체

AUTH_VALIDATION_EX SSTP(Secure Socket Tunneling Protocol) 연결 설정 중에 클라이언트가 PPP(지점 간) 인증을 바이패스하도록 설정하는 데 사용됩니다.
IKEV2_CONFIG_PARAMS IKEv2(Internet Key Exchange 버전 2) 디바이스(RFC 4306)에 대한 매개 변수를 얻거나 설정하는 데 사용됩니다.
IKEV2_PROJECTION_INFO IKE(인터넷 키 교환) 협상 중에 얻은 정보를 포함합니다. (IKEV2_PROJECTION_INFO)
IKEV2_PROJECTION_INFO2 IKE(인터넷 키 교환) 협상 중에 얻은 정보를 포함합니다. (IKEV2_PROJECTION_INFO2)
IKEV2_TUNNEL_CONFIG_PARAMS1 IKEv2(Internet Key Exchange 버전 2) 디바이스에 대한 터널 매개 변수를 얻거나 설정하는 데 사용됩니다.
IKEV2_TUNNEL_CONFIG_PARAMS2 IKEv2(Internet Key Exchange 버전 2) 디바이스에 대한 터널 매개 변수를 가져오거나 설정합니다.

L2TP_CONFIG_PARAMS0 RAS 서버에서 L2TP(계층 2 터널링 프로토콜)에 대한 디바이스 구성을 가져와서 설정하는 데 사용됩니다. (L2TP_CONFIG_PARAMS0)
L2TP_CONFIG_PARAMS1 RAS 서버에서 L2TP(계층 2 터널링 프로토콜)에 대한 디바이스 구성을 가져와서 설정하는 데 사용됩니다. (L2TP_CONFIG_PARAMS1)
MPR_CREDENTIALSEX_0 MPR_CREDENTIALSEX_0 구조에는 EAP(Extensible Authentication Protocols)에서 사용하는 정보와 같은 확장 자격 증명 정보가 포함됩니다.
MPR_CREDENTIALSEX_1 MPR_CREDENTIALSEX_1 구조에는 수요 다이얼 인터페이스에서 사용하는 미리 공유된 키가 포함되어 있습니다.
MPR_DEVICE_0 MPR_DEVICE_0 구조는 다중 연결 수요 다이얼 인터페이스의 링크에 사용되는 디바이스에 대한 정보를 저장합니다.
MPR_DEVICE_1 MPR_DEVICE_1 구조는 다중 연결 수요 다이얼 인터페이스의 링크에 사용되는 디바이스에 대한 정보를 저장합니다. MPR_DEVICE_0 정보 외에도 MPR_DEVICE_1 전화 번호 정보를 포함합니다.
MPR_FILTER_0 정적 필터 구성 정보를 포함합니다.
MPR_IF_CUSTOMINFOEX0 수요 다이얼 인터페이스에 대한 터널별 사용자 지정 구성을 가져오거나 설정합니다.
MPR_IFTRANSPORT_0 MPR_IFTRANSPORT_0 구조에는 특정 인터페이스 전송에 대한 정보가 포함됩니다.
MPR_INTERFACE_0 MPR_INTERFACE_0 구조에는 특정 라우터 인터페이스에 대한 정보가 포함됩니다.
MPR_INTERFACE_1 MPR_INTERFACE_1 구조에는 특정 라우터 인터페이스에 대한 구성 및 상태 정보가 포함됩니다.

MPR_INTERFACE_2 라우터 수요 다이얼 인터페이스에 대한 데이터를 포함합니다. (MPR_INTERFACE_2)
MPR_INTERFACE_3 라우터 수요 다이얼 인터페이스에 대한 데이터를 포함합니다. (MPR_INTERFACE_3)
MPR_SERVER_0 MPR_SERVER_0 구조는 디바이스에 대한 정보를 검색하는 데 사용됩니다.
MPR_SERVER_1 디바이스에서 PPTP(지점 간 터널링 프로토콜) 및 계층 2 터널링 프로토콜(L2TP)에 사용할 수 있는 포트 수를 검색하고 설정하는 데 사용됩니다.
MPR_SERVER_2 디바이스에서 PPTP(지점 간 터널링 프로토콜), L2TP(계층 2 터널링 프로토콜) 및 SSTP(Secure Socket Tunneling Protocol)에 사용할 수 있는 포트 수를 검색하고 설정하는 데 사용됩니다.
MPR_SERVER_EX0 RAS 서버의 구성을 가져오기 또는 설정하는 데 사용됩니다. (MPR_SERVER_EX0)
MPR_SERVER_EX1 RAS 서버의 구성을 가져오기 또는 설정하는 데 사용됩니다. (MPR_SERVER_EX1)
MPR_SERVER_SET_CONFIG_EX0 RAS 서버의 터널 구성 정보를 얻거나 설정하는 데 사용됩니다. (MPR_SERVER_SET_CONFIG_EX0)
MPR_SERVER_SET_CONFIG_EX1 RAS 서버의 터널 구성 정보를 얻거나 설정하는 데 사용됩니다. (MPR_SERVER_SET_CONFIG_EX1)
MPR_TRANSPORT_0 MPR_TRANSPORT_0 구조에는 특정 전송에 대한 정보가 포함됩니다.
MPRAPI_ADMIN_DLL_CALLBACKS MPRAPI_ADMIN_DLL_CALLBACKS 구조는 MprAdminInitializeDllEx 함수에서 콜백 함수 포인터를 등록하는 데 사용됩니다.

MPRAPI_OBJECT_HEADER mprAdminConnectionEnumEx 메서드에서 사용하는 RAS_CONNECTION_EX, MPR_SERVER_EX, MPR_SERVER_SET_CONFIG_EX, RAS_UPDATE_CONNECTION, AUTH_VALIDATION_EX 구조체 및 구조체 버전에 대한 구조체 버전을 정의합니다.
MPRAPI_TUNNEL_CONFIG_PARAMS0 RAS 서버에서 터널 매개 변수의 구성을 얻거나 설정하는 데 사용됩니다. (MPRAPI_TUNNEL_CONFIG_PARAMS0)
MPRAPI_TUNNEL_CONFIG_PARAMS1 RAS 서버에서 터널 매개 변수의 구성을 얻거나 설정하는 데 사용됩니다. (MPRAPI_TUNNEL_CONFIG_PARAMS1)
PPP_ATCP_INFO PPP_ATCP_INFO 구조에는 PPP AppleTalk 프로젝션 작업의 결과가 포함됩니다.
PPP_CCP_INFO PPP_CCP_INFO 구조에는 CCP(압축 제어 프로토콜) 협상 결과를 설명하는 정보가 포함되어 있습니다.
PPP_INFO PPP_INFO 구조체는 연결에 대한 다양한 PPP(Point to Point) 프로젝션 작업의 결과를 보고하는 데 사용됩니다.
PPP_INFO_2 PPP_INFO_2 구조체는 연결에 대한 다양한 PPP(Point to Point) 프로젝션 작업의 결과를 보고하는 데 사용됩니다.
PPP_INFO_3 PPP_INFO_3 구조체는 연결에 대한 다양한 PPP(Point to Point) 프로젝션 작업의 결과를 보고하는 데 사용됩니다.
PPP_IPCP_INFO PPP_IPCP_INFO 구조에는 PPP IP(인터넷 프로토콜) 협상 결과가 포함됩니다.
PPP_IPCP_INFO2 PPP_IPCP_INFO2 구조에는 PPP IP(인터넷 프로토콜) 협상 결과가 포함됩니다.

PPP_IPV6_CP_INFO IPv6 제어 프로토콜 협상 결과를 포함합니다.
PPP_IPXCP_INFO PPP_IPXCP_INFO 구조에는 PPP IPX(Internetwork Packet Exchange) 프로젝션 작업의 결과가 포함됩니다.
PPP_LCP_INFO PPP_LCP_INFO 구조에는 PPP LCP(링크 제어 프로토콜) 협상 결과를 설명하는 정보가 포함되어 있습니다.
PPP_NBFCP_INFO PPP_NBFCP_INFO 구조체에는 PPP NBF(NetBEUI Framer) 프로젝션 작업의 결과가 포함됩니다.
PPP_PROJECTION_INFO SSTP(Secure Socket Tunneling Protocol), PPTP(Point to Point Tunneling Protocol) 및 L2TP(Layer 2 Tunneling Protocol)에 대한 PPP(지점 간) 협상 중에 얻은 정보를 포함합니다. (PPP_PROJECTION_INFO)
PPP_PROJECTION_INFO2 SSTP(Secure Socket Tunneling Protocol), PPTP(Point to Point Tunneling Protocol) 및 L2TP(Layer 2 Tunneling Protocol)에 대한 PPP(지점 간) 협상 중에 얻은 정보를 포함합니다. (PPP_PROJECTION_INFO2)
PPTP_CONFIG_PARAMS RAS 서버에서 PPTP(지점 간 터널링 프로토콜)에 대한 디바이스 구성을 가져와서 설정하는 데 사용됩니다.
PROJECTION_INFO RAS_CONNECTION_EX 구조체에서 PPP_PROJECTION_INFO 및 IKEV2_PROJECTION_INFO 구조체의 자리 표시자로 사용됩니다.
PROJECTION_INFO2 RAS_CONNECTION_4 구조체에서 PPP_PROJECTION_INFO2 및 IKEV2_PROJECTION_INFO2 구조체의 자리 표시자로 사용됩니다.

RAS_CONNECTION_0 RAS_CONNECTION_0 구조에는 사용자 이름 또는 도메인과 같은 특정 연결에 대한 일반 정보가 포함됩니다. 전송되거나 받은 바이트와 같은 특정 연결에 대한 자세한 내용은 RAS_CONNECTION_1 참조하세요.
RAS_CONNECTION_1 RAS_CONNECTION_1 구조에는 수신된 오류 수 및 바이트와 같은 특정 연결에 대한 자세한 정보가 포함됩니다. 사용자 이름 또는 도메인과 같은 특정 연결에 대한 자세한 내용은 RAS_CONNECTION_0 참조하세요.
RAS_CONNECTION_2 RAS_CONNECTION_2 구조에는 연결을 식별하는 GUID(Globally Unique Identifier)를 포함하여 연결에 대한 정보가 포함됩니다.
RAS_CONNECTION_3 RAS_CONNECTION_3 구조에는 연결의 연결 및 격리 상태를 식별하는 GUID(Globally Unique Identifier)를 포함하여 연결에 대한 정보가 포함됩니다.
RAS_CONNECTION_4 연결에 연결된 _the 사용자 이름, 도메인, GUID(Globally Unique Identifier), NAP(네트워크 액세스 보호) 격리 상태, 패킷 통계, PPP(Point-to-Point) 및 IKEv2(인터넷 키 교환 버전 2) 관련 정보가 포함된 연결에 대한 특정 정보를 포함합니다.
RAS_CONNECTION_EX 연결과 연결된 :_the 사용자 이름, 도메인 및 GUID(Globally Unique Identifier), NAP(네트워크 액세스 보호) 격리 상태, 패킷 통계, PPP(Point-to-Point) 및 IKEv2(Internet Key Exchange 버전 2) 관련 정보가 포함된 연결에 대한 특정 정보를 포함합니다.
RAS_PORT_0 RAS_PORT_0 구조에는 포트 조건 및 포트 이름과 같은 특정 RAS 포트에 대한 일반 정보가 포함됩니다. 선 속도 또는 오류와 같은 특정 포트에 대한 자세한 내용은 RAS_PORT_1 참조하세요.
RAS_PORT_1 RAS_PORT_1 구조에는 선 속도 또는 오류와 같은 특정 RAS 포트에 대한 자세한 정보가 포함되어 있습니다. 포트 조건 또는 포트 이름과 같은 포트에 대한 자세한 내용은 RAS_PORT_0 참조하세요.
RAS_UPDATE_CONNECTION 활성 RAS 연결을 업데이트하는 데 사용됩니다.

RAS_USER_0 RAS_USER_0 구조에는 특정 원격 액세스 서비스 사용자에 대한 정보가 포함됩니다.
RAS_USER_1 RAS_USER_1 구조에는 특정 원격 액세스 서비스 사용자에 대한 정보가 포함됩니다. RAS_USER_1 구조체는 RAS_USER_1 추가 멤버인 bfPrivilege2를 지원한다는 점을 제외하고 RAS_USER_0 구조체와 비슷합니다.
ROUTER_CUSTOM_IKEv2_POLICY0 IKEv2 기본 모드 및 빠른 모드 정책 구성을 포함합니다.
ROUTER_IKEv2_IF_CUSTOM_CONFIG0 IKEv2 터널 기반 수요 다이얼 인터페이스에 대한 IKEv2 터널 구성 매개 변수를 가져오거나 설정합니다.
SSTP_CERT_INFO SSTP(Secure Socket Tunneling Protocol) 기반 인증서에 대한 정보를 포함합니다.
SSTP_CONFIG_PARAMS RAS 서버에서 SSTP(Secure Socket Tunneling Protocol)에 대한 디바이스 구성을 가져와서 설정하는 데 사용됩니다.

열거형

MPRAPI_OBJECT_TYPE MPRAPI_OBJECT_HEADER 구조체의 구조체 형식을 지정합니다.
RAS_HARDWARE_CONDITION RAS_HARDWARE_CONDITION 열거형 유형은 지정된 RAS 포트에 대한 하드웨어 상태 정보를 지정합니다.
RAS_PORT_CONDITION RAS_PORT_CONDITION 열거형 형식은 지정된 RAS 포트의 연결 조건에 대한 정보를 지정합니다.

RAS_QUARANTINE_STATE RAS_QUARANTINE_STATE 열거형 형식은 클라이언트 연결의 격리 상태를 나타냅니다.
ROUTER_CONNECTION_STATE 라우터에서 인터페이스의 가능한 상태를 열거합니다.
ROUTER_INTERFACE_TYPE ROUTER_INTERFACE_TYPE 형식은 라우터에서 다양한 종류의 인터페이스를 열거합니다.

msaatext.h 헤더

아티클2023. 03. 08.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 텍스트 서비스 프레임워크 비교
- Windows 접근성 기능

msaatext.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAccClientDocMgr 클라이언트 애플리케이션이 문서를 검색하는 메서드를 노출합니다.
IAccDictionary 문자열 조작에 대한 메서드를 노출합니다.
IAccServerDocMgr 클라이언트 애플리케이션에서 문서에 액세스할 수 있도록 하는 메서드를 노출합니다.
ICoCreatedLocally 로컬 개체에 대한 정보를 반환하는 메서드를 노출합니다.
ICoCreateLocally 클라이언트 애플리케이션이 서버 컨텍스트에서 도우미 개체를 만들 수 있도록 하는 메서드를 노출합니다.
ITfMSAAControl ITfMSAAControl 인터페이스는 TSF에서 불필요한 오버헤드를 방지하기 위해 Microsoft Active Accessibility에서 TSF 컨트롤에서 문서를 추가하거나 제거하는 데 사용됩니다. 이 인터페이스는 다른 애플리케이션에서 사용하지 않는 것이 좋습니다.
IVersionInfo 액세스 가능한 요소에 대한 버전 정보를 제공하는 메서드를 노출합니다.

msacm.h 헤더

아티클2023. 08. 24.

이 헤더는 Windows 멀티미디어에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 멀티미디어

msacm.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

acmDriverAdd acmDriverAdd 함수(msacm.h)는 사용 가능한 ACM 드라이버 목록에 드라이버를 추가합니다. (acmDriverAddW)
acmDriverAddA acmDriverAdd 함수는 사용 가능한 ACM 드라이버 목록에 드라이버를 추가합니다. (acmDriverAddA)
acmDriverAddW acmDriverAddW(유니코드) 함수(msacm.h)는 사용 가능한 ACM 드라이버 목록에 드라이버를 추가합니다. (acmDriverAddW)
acmDriverClose acmDriverClose 함수는 이전에 연 ACM 드라이버 instance 닫습니다. 함수가 성공하면 핸들이 무효화됩니다.
acmDriverDetails acmDriverDetails 함수(msacm.h)는 지정된 ACM 드라이버를 쿼리하여 해당 기능을 확인합니다. (acmDriverDetailsW)
acmDriverDetailsA acmDriverDetails 함수는 지정된 ACM 드라이버를 쿼리하여 해당 기능을 확인합니다. (acmDriverDetailsA)
acmDriverDetailsW acmDriverDetails(유니코드) 함수는 지정된 ACM 드라이버를 쿼리하여 해당 기능을 확인합니다. (acmDriverDetailsW)

acmDriverEnum acmDriverEnum 함수는 사용 가능한 ACM 드라이버를 열거하여 더 이상 드라이버가 없거나 콜백 함수가 FALSE를 반환할 때까지 계속됩니다.
acmDriverID acmDriverID 함수는 열린 ACM 드라이버 instance 또는 스트림 핸들과 연결된 ACM 드라이버 식별자의 핸들을 반환합니다.
acmDriverMessage acmDriverMessage 함수는 지정된 ACM 드라이버 instance 사용자 정의 메시지를 보냅니다.
acmDriverOpen acmDriverOpen 함수는 지정된 ACM 드라이버를 열고 드라이버와 통신하는 데 사용할 수 있는 드라이버 instance 핸들을 반환합니다.
acmDriverPriority acmDriverPriority 함수는 ACM 드라이버의 우선 순위와 상태를 수정합니다.
acmDriverRemove acmDriverRemove 함수는 사용 가능한 ACM 드라이버 목록에서 ACM 드라이버를 제거합니다. 호출 애플리케이션에 대해서만 드라이버가 제거됩니다. 드라이버가 전역적으로 설치된 경우에도 다른 애플리케이션에서 드라이버를 사용할 수 있습니다.
acmFilterChoose acmFilterChoose 함수는 사용자가 파형 오디오 필터를 선택할 수 있는 ACM 정의 대화 상자를 만듭니다.
acmFilterChooseA acmFilterChoose 함수는 사용자가 파형 오디오 필터를 선택할 수 있는 ACM 정의 대화 상자를 만듭니다. (acmFilterChooseA)
acmFilterChooseW acmFilterChoose(유니코드) 함수는 사용자가 파형 오디오 필터를 선택할 수 있는 ACM 정의 대화 상자를 만듭니다. (acmFilterChooseW)
acmFilterDetails acmFilterDetails 함수는 특정 파형 오디오 필터 태그가 있는 필터에 대한 세부 정보를 ACM에 쿼리합니다.

acmFilterDetailsA acmFilterDetails 함수는 특정 파형 오디오 필터 태그가 있는 필터에 대한 세부 정보를 ACM에 쿼리합니다. (acmFilterDetailsA)
acmFilterDetailsW acmFilterDetails(유니코드) 함수는 특정 파형 오디오 필터 태그가 있는 필터에 대한 세부 정보를 ACM에 쿼리합니다. (acmFilterDetailsW)
acmFilterEnum acmFilterEnum 함수는 ACM 드라이버에서 지정된 필터 태그에 사용할 수 있는 파형 오디오 필터를 열거합니다. 이 함수는 필터 태그에 더 이상 적합한 필터가 없거나 콜백 함수가 FALSE를 반환할 때까지 열거를 계속합니다.
acmFilterEnumA acmFilterEnum 함수는 ACM 드라이버에서 지정된 필터 태그에 사용할 수 있는 파형 오디오 필터를 열거합니다. 이 함수는 필터 태그에 더 이상 적합한 필터가 없거나 콜백 함수가 FALSE를 반환할 때까지 열거를 계속합니다. (acmFilterEnumA)
acmFilterEnumW acmFilterEnum(유니코드) 함수는 ACM 드라이버에서 지정된 필터 태그에 사용할 수 있는 파형 오디오 필터를 열거합니다. 이 함수는 필터 태그에 더 이상 적합한 필터가 없거나 콜백 함수가 FALSE를 반환할 때까지 열거를 계속합니다. (acmFilterEnumW)
acmFilterTagDetails acmFilterTagDetails 함수는 특정 파형 오디오 필터 태그에 대한 세부 정보를 ACM에 쿼리합니다.
acmFilterTagDetailsA acmFilterTagDetails 함수는 특정 파형 오디오 필터 태그에 대한 세부 정보를 ACM에 쿼리합니다. (acmFilterTagDetailsA)
acmFilterTagDetailsW acmFilterTagDetails(유니코드) 함수는 특정 파형 오디오 필터 태그에 대한 세부 정보를 ACM에 쿼리합니다. (acmFilterTagDetailsW)
acmFilterTagEnum acmFilterTagEnum 함수는 ACM 드라이버에서 사용할 수 있는 파형 오디오 필터 태그를 열거합니다. 이 함수는 더 이상 적합한 필터 태그가 없거나 콜백 함수가 FALSE를 반환할 때까지 열거를 계속합니다.

acmFilterTagEnumA acmFilterTagEnum 함수는 ACM 드라이버에서 사용할 수 있는 파형 오디오 필터 태그를 열거합니다. 이 함수는 더 이상 적합한 필터 태그가 없거나 콜백 함수가 FALSE를 반환할 때까지 열거를 계속합니다. (acmFilterTagEnumA)
acmFilterTagEnumW acmFilterTagEnum(유니코드) 함수는 ACM 드라이버에서 사용할 수 있는 파형 오디오 필터 태그를 열거합니다. 이 함수는 더 이상 적합한 필터 태그가 없거나 콜백 함수가 FALSE를 반환할 때까지 열거를 계속합니다. (acmFilterTagEnumW)
acmFormatChoose acmFormatChoose 함수는 사용자가 파형 오디오 형식을 선택할 수 있는 ACM 정의 대화 상자를 만듭니다.
acmFormatChooseA acmFormatChoose 함수는 사용자가 파형 오디오 형식을 선택할 수 있는 ACM 정의 대화 상자를 만듭니다. (acmFormatChooseA)
acmFormatChooseW acmFormatChoose(유니코드) 함수는 사용자가 파형 오디오 형식을 선택할 수 있는 ACM 정의 대화 상자를 만듭니다. (acmFormatChooseW)
acmFormatDetails acmFormatDetails 함수는 특정 파형 오디오 형식 태그에 대한 형식 세부 정보를 ACM에 쿼리합니다.
acmFormatDetailsA acmFormatDetails 함수는 특정 파형 오디오 형식 태그에 대한 형식 세부 정보를 ACM에 쿼리합니다. (acmFormatDetailsA)
acmFormatDetailsW acmFormatDetails(유니코드) 함수는 특정 파형 오디오 형식 태그에 대한 형식 세부 정보를 ACM에 쿼리합니다. (acmFormatDetailsW)
acmFormatEnum acmFormatEnum 함수는 ACM 드라이버에서 지정된 형식 태그에 사용할 수 있는 파형 오디오 형식을 열거합니다. 이 함수는 서식 태그에 더 이상 적합한 형식이 없거나 콜백 함수가 FALSE를 반환할 때까지 계속 열거합니다.

acmFormatEnumA acmFormatEnum 함수는 ACM 드라이버에서 지정된 형식 태그에 사용할 수 있는 파형 오디오 형식을 열거합니다. 이 함수는 서식 태그에 더 이상 적합한 형식이 없거나 콜백 함수가 FALSE를 반환할 때까지 계속 열거합니다. (acmFormatEnumA)
acmFormatEnumW acmFormatEnum(유니코드) 함수는 ACM 드라이버에서 지정된 형식 태그에 사용할 수 있는 파형 오디오 형식을 열거합니다. 이 함수는 서식 태그에 더 이상 적합한 형식이 없거나 콜백 함수가 FALSE를 반환할 때까지 계속 열거합니다. (acmFormatEnumW)
acmFormatSuggest acmFormatSuggest 함수는 ACM 또는 지정된 ACM 드라이버를 쿼리하여 제공된 원본 형식에 대한 대상 형식을 제안합니다.
acmFormatTagDetails acmFormatTagDetails 함수는 특정 파형 오디오 형식 태그에 대한 자세한 내용을 ACM에 쿼리합니다.
acmFormatTagDetailsA acmFormatTagDetails 함수는 특정 파형 오디오 형식 태그에 대한 자세한 내용을 ACM에 쿼리합니다. (acmFormatTagDetailsA)
acmFormatTagDetailsW acmFormatTagDetails(유니코드) 함수는 특정 파형 오디오 형식 태그에 대한 자세한 내용을 ACM에 쿼리합니다. (acmFormatTagDetailsW)
acmFormatTagEnum acmFormatTagEnum 함수는 ACM 드라이버에서 사용할 수 있는 파형 오디오 형식 태그를 열거합니다. 이 함수는 더 이상 적합한 형식 태그가 없거나 콜백 함수가 FALSE를 반환할 때까지 계속 열거합니다.
acmFormatTagEnumA acmFormatTagEnum 함수는 ACM 드라이버에서 사용할 수 있는 파형 오디오 형식 태그를 열거합니다. 이 함수는 더 이상 적합한 형식 태그가 없거나 콜백 함수가 FALSE를 반환할 때까지 계속 열거합니다. (acmFormatTagEnumA)
acmFormatTagEnumW acmFormatTagEnum(유니코드) 함수는 ACM 드라이버에서 사용할 수 있는 파형 오디오 형식 태그

를 열거합니다. 이 함수는 더 이상 적합한 형식 태그가 없거나 콜백 함수가 FALSE를 반환할 때까지 계속 열거합니다. (acmFormatTagEnumW)
acmGetVersion acmGetVersion 함수는 ACM의 버전 번호를 반환합니다.
acmMetrics acmMetrics 함수는 ACM 또는 관련 ACM 개체에 대한 다양한 메트릭을 반환합니다.
acmStreamClose acmStreamClose 함수는 ACM 변환 스트림을 닫습니다. 함수가 성공하면 핸들이 무효화됩니다.
acmStreamConvert acmStreamConvert 함수는 지정된 변환 스트림에서 변환을 수행하도록 ACM에 요청합니다. 변환은 스트림이 열린 방법에 따라 동기 또는 비동기일 수 있습니다.
acmStreamMessage acmStreamMessage 함수는 ACM 드라이버에 드라이버 관련 메시지를 보냅니다.
acmStreamOpen acmStreamOpen 함수는 ACM 변환 스트림을 엽니다. 변환 스트림은 지정된 오디오 형식에서 다른 오디오 형식으로 데이터를 변환하는 데 사용됩니다.
acmStreamPrepareHeader acmStreamPrepareHeader 함수는 ACM 스트림 변환을 위해 ACMSTREAMHEADER 구조를 준비합니다.
acmStreamReset acmStreamReset 함수는 지정된 ACM 스트림에 대한 변환을 중지합니다. 보류 중인 모든 버퍼는 완료된 것으로 표시되고 애플리케이션에 반환됩니다.
acmStreamSize acmStreamSize 함수는 ACM 스트림의 원본 또는 대상 버퍼에 권장되는 크기를 반환합니다.
acmStreamUnprepareHeader acmStreamUnprepareHeader 함수는 ACM 스트림에 대한 acmStreamPrepareHeader 함수에서 수행하는 준비를 정리합니다.

콜백 함수

ACMDRIVERENUMCB acmDriverEnumCallback 함수는 acmDriverEnum 함수와 함께 사용되는 콜백 함수를 지정합니다. acmDriverEnumCallback 이름은 애플리케이션 정의 함수 이름의 자리 표시자입니다.
ACMDRIVERPROC acmDriverProc 함수는 ACM 드라이버와 함께 사용되는 콜백 함수를 지정합니다.
ACMFILTERCHOOSEHOOKPROC acmFilterChooseHookProc 함수는 acmFilterChoose 대화 상자를 연결하는 사용자 정의 함수를 지정합니다.
ACMFILTERENUMCB ACMFILTERENUMCB 콜백 함수는 acmFilterEnum 함수와 함께 사용되는 콜백 함수를 지정합니다.
ACMFILTERENUMCBA acmFilterEnumCallback 함수는 acmFilterEnum 함수와 함께 사용되는 콜백 함수를 지정합니다. acmFilterEnumCallback 이름은 애플리케이션 정의 함수 이름의 자리 표시자입니다. (ACMFILTERENUMCBA)
ACMFILTERENUMCBW ACMFILTERENUMCBW(유니코드) 콜백 함수는 acmFilterEnum 함수와 함께 사용되는 콜백 함수를 지정합니다.
ACMFILTERTAGENUMCB ACMFILTERTAGENUMCB 콜백 함수는 acmFilterTagEnum 함수와 함께 사용되는 콜백 함수를 지정합니다.
ACMFILTERTAGENUMCBA acmFilterTagEnumCallback 함수는 acmFilterTagEnum 함수와 함께 사용되는 콜백 함수를 지정합니다. acmFilterTagEnumCallback 함수 이름은 애플리케이션 정의 함수 이름의 자리 표시자입니다. (ACMFILTERTAGENUMCBA)
ACMFILTERTAGENUMCBW ACMFILTERTAGENUMCBW(유니코드) 콜백 함수는 acmFilterTagEnum 함수와 함께 사용되는 콜백 함수를 지정합니다.

ACMFORMATCHOOSEHOOKPROC acmFormatChooseHookProc 함수는 acmFormatChoose 대화 상자를 후크하는 사용자 정의 함수를 지정합니다. acmFormatChooseHookProc 이름은 애플리케이션 정의 이름의 자리 표시자입니다.
ACMFORMATENUMCB ACMFORMATENUMCB 콜백 함수는 acmFormatEnum 함수와 함께 사용되는 콜백 함수를 지정합니다.
ACMFORMATENUMCBA acmFormatEnumCallback 함수는 acmFormatEnum 함수와 함께 사용되는 콜백 함수를 지정합니다. acmFormatEnumCallback 이름은 애플리케이션 정의 함수 이름의 자리 표시자입니다. (ACMFORMATENUMCBA)
ACMFORMATENUMCBW ACMFORMATENUMCBW(유니코드) 콜백 함수는 acmFormatEnum 함수와 함께 사용되는 콜백 함수를 지정합니다.
ACMFORMATTAGENUMCB ACMFORMATTAGENUMCB 콜백 함수는 acmFormatTagEnum 함수와 함께 사용되는 콜백 함수를 지정합니다.
ACMFORMATTAGENUMCBA acmFormatTagEnumCallback 함수는 acmFormatTagEnum 함수와 함께 사용되는 콜백 함수를 지정합니다. acmFormatTagEnumCallback 이름은 애플리케이션 정의 함수 이름의 자리 표시자입니다. (ACMFORMATTAGENUMCBA)
ACMFORMATTAGENUMCBW ACMFORMATTAGENUMCBW(유니코드) 콜백 함수는 acmFormatTagEnum 함수와 함께 사용되는 콜백 함수를 지정합니다.

구조체

ACMDRIVERDETAILS ACMDRIVERDETAILS 구조체는 ACM 드라이버의 기능을 설명합니다.
ACMFILTERCHOOSE

ACMFILTERCHOOSE 구조체에는 ACM이 시스템 정의 파형 오디오 필터 선택 대화 상자를 초기화하는 데 사용하는 정보가 포함되어 있습니다.
ACMFILTERDETAILS ACMFILTERDETAILS 구조체는 ACM 드라이버에 대한 특정 필터 태그에 대한 파형 오디오 필터를 자세히 설명합니다.
ACMFILTERTAGDETAILS ACMFILTERTAGDETAILS 구조체는 ACM 필터 드라이버에 대한 파형 오디오 필터 태그를 자세히 설명합니다.
ACMFORMATCHOOSE ACMFORMATCHOOSE 구조체에는 ACM이 시스템 정의 파형 오디오 형식 선택 대화 상자를 초기화하는 데 사용하는 정보가 포함되어 있습니다.
ACMFORMATDETAILS ACMFORMATDETAILS 구조체는 ACM 드라이버의 특정 형식 태그에 대한 파형 오디오 형식을 자세히 설명합니다.
ACMFORMATTAGDETAILS ACMFORMATTAGDETAILS 구조체는 ACM 드라이버에 대한 파형 오디오 형식 태그를 자세히 설명합니다.
ACMSTREAMHEADER ACMSTREAMHEADER 구조는 변환 스트림에 대한 ACM 변환 원본 및 대상 버퍼 쌍을 식별하는 데 사용되는 헤더를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

msajtransport.h 헤더

아티클2023. 08. 24.

이 헤더는 AllJoyn API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- AllJoyn API

msajtransport.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AllJoynCloseBusHandle 명명된 파이프 핸들을 닫습니다.
AllJoynConnectToBus AllJoyn 라우터 노드 서비스 명명된 파이프를 열고 PIPE_NOWAIT 설정합니다.
AllJoynEnumEvents TCP 소켓 WSAEnumNetworkEvents 기능과 유사한 AllJoyn 전송 기능을 제공합니다.
AllJoynEventSelect TCP 소켓 WSAEventSelect 기능과 유사한 AllJoyn 전송 기능을 제공합니다.
AllJoynReceiveFromBus 명명된 파이프를 통해 버스에서 데이터를 받습니다.
AllJoynSendToBus 명명된 파이프를 통해 버스로 데이터를 보냅니다. 이 API의 호출자는 bytesTransferred가 요청된 바이트보다 작은 경우 검사 이 API를 다시 호출하여 나머지 데이터를 다시 보냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

msapofxproxy.h 헤더

아티클2023. 08. 24.

이 헤더는 오디오 디바이스 DDI 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 오디오 디바이스 DDI 참조

msapofxproxy.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

KSP_PINMODE KSP_PINMODE 구조체는 핀 팩터리에 대해 핀 속성 및 지원되는 오디오 처리 모드를 지정합니다.

열거형

KSPROPERTY_AUDIOEFFECTSDISCOVERY KSPROPERTY_AUDIOEFFECTSDISCOVERY 열거형은 API(오디오 처리 개체) 목록에서 사용되는 상수를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mscat.h 헤더

아티클2023. 08. 24.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

mscat.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CryptCATAdminAcquireContext 카탈로그 관리자 컨텍스트에 대한 핸들을 가져옵니다.
CryptCATAdminAcquireContext2 지정된 해시 알고리즘 및 해시 정책에 대한 카탈로그 관리자 컨텍스트에 대한 핸들을 가져옵니다.
CryptCATAdminAddCatalog 카탈로그 데이터베이스에 카탈로그를 추가합니다.
CryptCATAdminCalcHashFromFileHandle 파일의 해시를 계산합니다.
CryptCATAdminCalcHashFromFileHandle2 지정된 알고리즘을 사용하여 파일의 해시를 계산합니다.
CryptCATAdminEnumCatalogFromHash 지정된 해시를 포함하는 카탈로그를 열거합니다.
CryptCATAdminReleaseCatalogContext 이전에 CryptCATAdminAddCatalog 함수에서 반환한 카탈로그 컨텍스트에 대한 핸들을 해제합니다.
CryptCATAdminReleaseContext 이전에 CryptCATAdminAcquireContext 함수에서 할당한 핸들을 해제합니다.
CryptCATAdminRemoveCatalog

카탈로그 파일을 삭제하고 Windows 카탈로그 데이터베이스에서 해당 카탈로그의 항목을 제거합니다.
CryptCATAdminResolveCatalogPath 지정된 카탈로그의 정규화된 경로를 검색합니다.
CryptCATCatalogInfoFromContext 지정된 카탈로그 컨텍스트에서 카탈로그 정보를 검색합니다.
CryptCATCDFClose CDF(카탈로그 정의 파일)를 닫고 해당 CRYPTCATCDF 구조체에 대한 메모리를 해제합니다.
CryptCATCDFEnumCatAttributes CDF(카탈로그 정의 파일)의 CatalogHeader 섹션 내에서 카탈로그 수준 특성을 열거합니다.
CryptCATCDFOpen CRYPTCATCDF 구조를 읽고 초기화하기 위해 기존 CDF(카탈로그 정의 파일)를 엽니다.
CryptCATClose 이전에 CryptCATOpen 함수에서 연 카탈로그 핸들을 닫습니다.
CryptCATEnumerateAttr 카탈로그의 멤버와 연결된 특성을 열거합니다. 이 함수에는 연결된 가져오기 라이브러리가 없습니다.
CryptCATEnumerateCatAttr 카탈로그와 연결된 특성을 열거합니다. 이 함수에는 연결된 가져오기 라이브러리가 없습니다.
CryptCATEnumerateMember 카탈로그의 멤버를 열거합니다.
CryptCATGetAttrInfo 카탈로그 멤버의 특성에 대한 정보를 검색합니다.
CryptCATGetMemberInfo 카탈로그의 PKCS에서 멤버 정보를 검색합니다.
CryptCATHandleFromStore

메모리에서 카탈로그 핸들을 검색합니다.
CryptCATOpen 카탈로그를 열고 열려 있는 카탈로그에 컨텍스트 핸들을 반환합니다.
CryptCATPersistStore 지정된 카탈로그 저장소의 정보를 서명되지 않은 카탈로그 파일에 저장합니다.
CryptCATPutAttrInfo 특성에 대한 메모리를 할당하고 카탈로그 멤버에 추가합니다.
CryptCATPutCatAttrInfo 카탈로그 파일 특성에 대한 메모리를 할당하고 카탈로그에 추가합니다.
CryptCATPutMemberInfo 카탈로그 멤버에 대한 메모리를 할당하고 카탈로그에 추가합니다.
CryptCATStoreFromHandle 카탈로그 핸들에서 CRYPTCATSTORE 구조를 검색합니다.
IsCatalogFile 지정된 파일이 카탈로그 파일인지 여부를 나타내는 부울 값을 검색합니다.

콜백 함수

PFN_CDF_PARSE_ERROR_CALLBACK CDF(카탈로그 정의 파일)를 구문 분석하는 동안 카탈로그 정의 함수 오류에 대해 호출됩니다.

구조체

CATALOG_INFO

CATALOG_INFO 구조체에는 카탈로그 파일의 이름이 포함됩니다. 이 구조체는 CryptCATCatalogInfoFromContext 함수에서 사용됩니다.
CRYPTCATATTRIBUTE CRYPTCATATTRIBUTE 구조체는 카탈로그 특성을 정의합니다. 이 구조체는 CryptCATEnumerateAttr 및 CryptCATEnumerateCatAttr 함수에서 사용됩니다.
CRYPTCATCDF CDF(카탈로그 정의 파일)에서 서명된 카탈로그 파일(.cat)을 만드는 데 사용되는 정보를 포함합니다.
CRYPTCATMEMBER CRYPTCATMEMBER 구조체는 카탈로그 멤버에 대한 정보를 제공합니다. 이 구조체는 CryptCATGetMemberInfo 및 CryptCATEnumerateAttr 함수에서 사용됩니다.
CRYPTCATSTORE 카탈로그 파일을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mschapp.h 헤더

아티클2023. 08. 24.

이 헤더는 MS-CHAP 암호 관리 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- MS-CHAP 암호 관리 API

mschapp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

MSChapSrvChangePassword 사용자 계정의 암호를 변경합니다.
MSChapSrvChangePassword2 MSChapSrvChangePassword2 함수는 상호 암호화를 지원하면서 사용자 계정의 암호를 변경합니다.

구조체

CYPHER_BLOCK CYPHER_BLOCK OWF(단방향 함수) 암호 해시에 대한 스토리지의 기본 단위입니다.
ENCRYPTED_LM_OWF_PASSWORD ENCRYPTED_LM_OWF_PASSWORD 사용자의 암호화된 LM(Lan Manager) OWF(단방향 함수) 암호 해시를 저장합니다.
LM_OWF_PASSWORD LM_OWF_PASSWORD 사용자 암호의 LM(Lan Manage) OWF(단방향 함수)를 저장합니다.
SAMPR_ENCRYPTED_USER_PASSWORD SAMPR_ENCRYPTED_USER_PASSWORD 사용자의 암호화된 암호를 저장합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

msclus.h 헤더

아티클2023. 03. 08.

이 헤더는 Windows 클러스터링에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 클러스터링

msclus.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ISClusPartitionEx 실제 디스크 리소스의 파티션에 대한 확장된 정보를 제공합니다.

구조체

CLUSTER_CREATE_GROUP_INFO CLUSTER_CREATE_GROUP_INFO 구조를 사용하면 호출자가 새 그룹을 만들 때 추가 속성을 제공할 수 있습니다.
CLUSTER_ENUM_ITEM CLUSTER_ENUM_ITEM 구조에는 클러스터 개체의 속성이 포함되어 있으며 ClusterEnumEx 및 ClusterNodeEnumEx 함수에서 클러스터를 열거하는 데 사용됩니다.
CLUSTER_GROUP_ENUM_ITEM CLUSTER_GROUP_ENUM_ITEM 구조에는 클러스터 그룹의 속성이 포함됩니다. (CLUSTER_GROUP_ENUM_ITEM)
CLUSTER_MEMBERSHIP_INFO CLUSTER_MEMBERSHIP_INFO 구조는 클러스터의 멤버 자격 정보를 나타냅니다. (CLUSTER_MEMBERSHIP_INFO)
CLUSTER_RESOURCE_ENUM_ITEM CLUSTER_RESOURCE_ENUM_ITEM 구조체는 클러스터 리소스의 속성을 나타내며 ClusterResourceEnumEx 함수에서 클러스터 리소스를 열거하는 데 사용됩니다.

GROUP_FAILURE_INFO GROUP_FAILURE_INFO 구조체는 그룹 오류에 대한 장애 조치(failover) 시도에 대한 정보를 나타냅니다. (GROUP_FAILURE_INFO)
GROUP_FAILURE_INFO_BUFFER GROUP_FAILURE_INFO_BUFFER 구조체는 GROUP_FAILURE_INFO 구조체에 대한 버퍼를 나타냅니다. (GROUP_FAILURE_INFO_BUFFER)
NOTIFY_FILTER_AND_TYPE NOTIFY_FILTER_AND_TYPE 구조는 CreateClusterNotifyPortV2 함수에서 만든 알림 포트에 대한 필터를 나타냅니다.
RESOURCE_FAILURE_INFO RESOURCE_FAILURE_INFO 구조체는 리소스에 대한 장애 조치(failover) 시도에 대한 정보를 나타내며 RESOURCE_FAILURE_INFO_BUFFER 구조체에서 사용됩니다.
RESOURCE_FAILURE_INFO_BUFFER RESOURCE_FAILURE_INFO_BUFFER 구조체는 리소스 오류에 대한 버퍼를 나타냅니다. (RESOURCE_FAILURE_INFO_BUFFER)
RESOURCE_TERMINAL_FAILURE_INFO_BUFFER RESOURCE_TERMINAL_FAILURE_INFO_BUFFER 구조체는 리소스의 터미널 오류에 대한 버퍼를 나타냅니다.

열거형

CLCTL_CODES CLCTL_CODES(msclus.h)는 제어 코드가 수행할 수 있는 작업을 열거합니다.
CLUS_CHARACTERISTICS CLUS_CHARACTERISTICS(msclus.h)는 리소스 종류 및 리소스의 특성을 열거합니다.
CLUS_FLAGS CLUS_FLAGS(msclus.h)는 리소스 또는 그룹을 핵심 리소스로 식별합니다.

CLUS_GROUP_START_SETTING CLUS_GROUP_START_SETTING(msclus.h)는 클러스터 그룹에 대한 시작 설정을 열거합니다.
CLUS_RESSUBCLASS CLUS_RESSUBCLASS 열거형(msclus.h)은 공유 리소스를 관리하는 리소스 하위 클래스를 식별합니다.
CLUS_RESSUBCLASS_NETWORK CLUS_RESSUBCLASS_NETWORK 열거형(msclus.h)은 IP 주소 공급자를 관리하는 리소스 하위 클래스를 식별합니다.
CLUS_RESSUBCLASS_STORAGE CLUS_RESSUBCLASS_STORAGE 열거형(msclus.h)은 공유 버스를 관리하는 리소스 하위 클래스를 식별합니다.
CLUSCTL_CLUSTER_CODES CLUSCTL_CLUSTER_CODES 열거형(msclus.h)은 ClusterControl 함수에서 사용하는 클러스터 제어 코드를 열거합니다.
CLUSCTL_GROUP_CODES CLUSCTL_GROUP_CODES 열거형(msclus.h)은 그룹 제어 코드를 열거합니다.
CLUSCTL_GROUPSET_CODES CLUSCTL_GROUPSET_CODES 열거형(msclus.h)은 ClusterGroupSetControl 함수에서 사용하는 groupset 컨트롤 코드를 열거합니다.
CLUSCTL_NETINTERFACE_CODES CLUSCTL_NETINTERFACE_CODES 열거형(msclus.h)은 네트워크 인터페이스 제어 코드를 열거합니다.
CLUSCTL_NETWORK_CODES CLUSCTL_NETWORK_CODES 열거형(msclus.h)은 네트워크 제어 코드를 열거합니다.
CLUSCTL_NODE_CODES CLUSCTL_NODE_CODES 열거형(msclus.h)은 노드 제어 코드를 열거합니다.
CLUSCTL_RESOURCE_CODES CLUSCTL_RESOURCE_CODES 열거형(msclus.h)은 리소스 제어 코드를 열거합니다.

CLUSCTL_RESOURCE_TYPE_CODES CLUSCTL_RESOURCE_TYPE_CODES 열거형(msclus.h)은 리소스 종류 제어 코드를 열거합니다.
CLUSGROUP_TYPE CLUSGROUP_TYPE 열거형(msclus.h)은 만들 클러스터 그룹의 유형을 지정합니다.
CLUSPROP_PIFLAGS CLUSPROP_PIFLAGS 열거형(msclus.h)은 디스크 파티션 정보를 나타냅니다.
CLUSTER_CHANGE CLUSTER_CHANGE 열거형(msclus.h)은 반환된 알림의 유형을 설명합니다.
CLUSTER_CHANGE_CLUSTER_V2 CLUSTER_CHANGE_CLUSTER_V2 열거형(msclus.h)은 클러스터에 대해 생성된 알림 목록을 정의합니다.
CLUSTER_CHANGE_GROUP_V2 CLUSTER_CHANGE_GROUP_V2 열거형(msclus.h)은 그룹에 대해 생성된 알림 목록을 정의합니다.
CLUSTER_CHANGE_GROUPSET_V2 CLUSTER_CHANGE_GROUPSET_V2 열거형(msclus.h)은 그룹 집합에 대해 생성된 알림 목록을 정의합니다.
CLUSTER_CHANGE_NETINTERFACE_V2 CLUSTER_CHANGE_NETINTERFACE_V2 열거형(msclus.h)은 클러스터 네트워크 인터페이스에 대해 생성된 알림 집합을 정의합니다.
CLUSTER_CHANGE_NETWORK_V2 CLUSTER_CHANGE_NETWORK_V2 열거형(msclus.h)은 클러스터 네트워크에 대해 생성된 알림을 정의합니다.
CLUSTER_CHANGE_NODE_UPGRADE_PHASE_V2 CLUSTER_CHANGE_NODE_UPGRADE_PHASE_V2 열거형(msclus.h)은 클러스터 노드 업그레이드에 대해 생성된 알림을 정의합니다.
CLUSTER_CHANGE_NODE_V2 CLUSTER_CHANGE_NODE_V2 열거형(msclus.h)은 클러스터 노드에 대해 생성된 알림을 정의합니다.

CLUSTER_CHANGE_QUORUM_V2 CLUSTER_CHANGE_QUORUM_V2 열거형(msclus.h)은 쿼럼 관련 정보에 대해 생성된 알림을 정의합니다.
CLUSTER_CHANGE_REGISTRY_V2 CLUSTER_CHANGE_REGISTRY_V2 열거형(msclus.h)은 레지스트리 키에 대해 생성된 알림을 정의합니다.
CLUSTER_CHANGE_RESOURCE_TYPE_V2 CLUSTER_CHANGE_RESOURCE_TYPE_V2 열거형(msclus.h)은 리소스 종류에 대해 생성된 알림 집합을 정의합니다.
CLUSTER_CHANGE_RESOURCE_V2 CLUSTER_CHANGE_RESOURCE_V2 열거형(msclus.h)이 리소스에 대해 생성된 알림 목록을 정의하는 방법을 알아봅니다.
CLUSTER_CHANGE_SHARED_VOLUME_V2 CLUSTER_CHANGE_SHARED_VOLUME_V2 열거형(msclus.h)은 클러스터 공유 볼륨에 대해 생성된 알림을 정의합니다.
CLUSTER_CHANGE_SPACEPORT_V2 CLUSTER_CHANGE_SPACEPORT_V2 열거형(msclus.h)은 장애 조치(failover) 클러스터와 관련이 있으며 자세한 내용은 확인할 수 있습니다.
CLUSTER_CONTROL_OBJECT CLUSTER_CONTROL_OBJECT(msclus.h)는 컨트롤 코드가 적용되는 클러스터 개체 유형을 나타내는 컨트롤 코드의 8비트 개체 구성 요소입니다.
CLUSTER_ENUM CLUSTER_ENUM 열거형(msclus.h)은 열거되는 클러스터 개체의 형식을 설명합니다.
CLUSTER_GROUP_AUTOFAILBACK_TYPE AutoFailbackType 그룹 공통 속성은 CLUSTER_GROUP_AUTOFAILBACK_TYPE 사용하여 그룹이 다시 온라인 상태가 된 후 노드 소유자에게 장애 복구(failback)해야 하는지를 지정합니다.
CLUSTER_GROUP_ENUM CLUSTER_GROUP_ENUM 열거형(msclus.h)은 ClusterGroupEnum 및 ClusterGroupOpenEnum 함수에 의해 열거되는 클러스터 개체의 형식을 설명합니다.

CLUSTER_GROUP_PRIORITY CLUSTER_GROUP_PRIORITY 열거형(msclus.h)은 그룹의 우선 순위 수준을 지정합니다. (CLUSTER_GROUP_PRIORITY)
CLUSTER_GROUP_STATE CLUSTER_GROUP_STATE 열거형(msclus.h)은 그룹의 가능한 상태를 열거합니다.
CLUSTER_MGMT_POINT_TYPE CLUSTER_MGMT_POINT_TYPE 열거형(msclus.h)은 클러스터에 대한 관리 지점의 형식을 지정합니다.
CLUSTER_NETINTERFACE_STATE CLUSTER_NETINTERFACE_STATE 열거형(msclus.h)은 네트워크 인터페이스 상태의 가능한 값을 열거합니다.
CLUSTER_NETWORK_ENUM CLUSTER_NETWORK_ENUM(msclus.h)는 ClusterNetworkEnum 및 ClusterNetworkOpenEnum 함수에 의해 열거되는 클러스터 개체의 형식을 설명합니다.
CLUSTER_NETWORK_ROLE CLUSTER_NETWORK_ROLE 열거형(msclus.h)은 클러스터에서 네트워크가 수행하는 역할을 설명합니다.
CLUSTER_NETWORK_STATE CLUSTER_NETWORK_STATE 열거형(msclus.h)은 네트워크 상태의 가능한 값을 열거합니다.
CLUSTER_NODE_DRAIN_STATUS CLUSTER_NODE_DRAIN_STATUS 열거형(msclus.h)은 노드 드레이닝 상태의 가능한 값을 열거합니다.
CLUSTER_NODE_ENUM CLUSTER_NODE_ENUM(msclus.h)는 ClusterNodeEnum 및 ClusterNodeOpenEnum 함수에 의해 열거되는 클러스터 개체의 형식을 설명합니다.
CLUSTER_NODE_STATE CLUSTER_NODE_STATE 열거형(msclus.h)은 클러스터 노드의 상태를 설명합니다.

CLUSTER_NODE_STATUS CLUSTER_NODE_STATUS 열거형(msclus.h)은 클러스터 노드의 상태를 설명합니다.
CLUSTER_NOTIFICATIONS_VERSION CLUSTER_NOTIFICATIONS_VERSION 열거형(msclus.h)은 다양한 버전의 클러스터 알림 열거형을 정의합니다.
CLUSTER_OBJECT_TYPE CLUSTER_OBJECT_TYPE 열거형(msclus.h)은 알림이 요청되거나 생성되는 개체의 형식을 정의합니다.
CLUSTER_PROPERTY_FORMAT CLUSTER_PROPERTY_FORMAT 열거형(msclus.h)은 속성 목록에서 속성 값의 데이터 형식을 지정합니다.
CLUSTER_PROPERTY_SYNTAX CLUSTER_PROPERTY_SYNTAX 열거형(msclus.h)은 속성 목록의 구문 구조에 대해 가능한 값을 제공합니다.
CLUSTER_PROPERTY_TYPE CLUSTER_PROPERTY_TYPE 열거형(msclus.h)은 클러스터 속성 목록에서 지원하는 속성 형식을 정의합니다.
CLUSTER_QUORUM_TYPE CLUSTER_QUORUM_TYPE 열거형(msclus.h)은 만들 쿼럼 리소스의 형식입니다.
CLUSTER_QUORUM_VALUE CLUSTER_QUORUM_VALUE(msclus.h)는 ClusterControl 함수에서 반환하는 값을 두 개의 특정 제어 코드로 열거합니다.
CLUSTER_REG_COMMAND CLUSTER_REG_COMMAND(msclus.h)는 클러스터 조인을 시도할 때 로컬 노드가 수행할 수 있는 클러스터 레지스트리 명령을 열거합니다.
CLUSTER_RESOURCE_CLASS CLUSTER_RESOURCE_CLASS 열거형(msclus.h)은 리소스의 클래스를 정의합니다.

CLUSTER_RESOURCE_CREATE_FLAGS CLUSTER_RESOURCE_CREATE_FLAGS 열거형(msclus.h)은 지정된 리소스를 할당할 리소스 모니터를 결정합니다.
CLUSTER_RESOURCE_EMBEDDED_FAILURE_ACTION CLUSTER_RESOURCE_EMBEDDED_FAILURE_ACTION 열거형(msclus.h)은 리소스에 포함된 오류가 있을 때 수행할 수 있는 다양한 작업을 지정합니다.
CLUSTER_RESOURCE_ENUM CLUSTER_RESOURCE_ENUM(msclus.h)는 ClusterResourceEnum 또는 ClusterResourceOpenEnum 함수에 의해 열거되는 클러스터 개체의 형식을 설명합니다.
CLUSTER_RESOURCE_RESTART_ACTION CLUSTER_RESOURCE_RESTART_ACTION(msclus.h)는 RestartAction 리소스 공용 속성에서 리소스가 실패할 경우 클러스터 서비스의 작업을 지정하는 데 사용됩니다.
CLUSTER_RESOURCE_STATE CLUSTER_RESOURCE_STATE 열거형(msclus.h)은 리소스의 작동 상태를 설명합니다.
CLUSTER_RESOURCE_STATE_CHANGE_REASON CLUSTER_RESOURCE_STATE_CHANGE_REASON(msclus.h)는 리소스 상태 변경 이유를 설명하기 위해 두 개의 특정 제어 코드에서 사용됩니다.
CLUSTER_RESOURCE_TYPE_ENUM CLUSTER_RESOURCE_TYPE_ENUM 열거형(msclus.h)은 두 가지 특정 함수로 열거되는 클러스터 개체의 형식을 설명합니다.
CLUSTER_SHARED_VOLUME_SNAPSHOT_STATE CLUSTER_SHARED_VOLUME_SNAPSHOT_STATE 열거형(msclus.h)은 공유 볼륨에 대한 다양한 스냅샷 상태를 지정합니다.
NODE_CLUSTER_STATE NODE_CLUSTER_STATE 열거형(msclus.h)은 클러스터의 상태를 나타냅니다.

msctf.h 헤더

아티클2023. 08. 24.

이 헤더는 Text Services Framework에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 텍스트 서비스 프레임워크 비교

msctf.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumITfCompositionView IEnumITfCompositionView 인터페이스는 TSF 관리자에 의해 구현되어 컴퍼지션 뷰 개체의 열거형을 제공합니다.
IEnumTfContexts IEnumTfContexts 인터페이스는 컨텍스트 개체의 열거를 제공하기 위해 TSF 관리자에 의해 구현됩니다.
IEnumTfContextViews 구현되지 않았습니다. (IEnumTfContextViews)
IEnumTfDisplayAttributeInfo IEnumTfDisplayAttributeInfo 인터페이스는 TSF 관리자가 표시 특성 정보 개체의 열거형을 제공하기 위해 구현됩니다.
IEnumTfDocumentMgrs IEnumTfDocumentMgrs 인터페이스는 TSF 관리자에 의해 구현되어 문서 관리자 개체의 열거형을 제공합니다.
IEnumTfFunctionProviders IEnumTfFunctionProviders 인터페이스는 함수 공급자 개체의 열거형을 제공하기 위해 TSF 관리자에 의해 구현됩니다.
IEnumTfInputProcessorProfiles IEnumTfInputProcessorProfiles 인터페이스는 TSF 관리자에 의해 구현되고 애플리케이션 또는 textservices에서 사용됩니다. ITfInputProcessorProfileMgr::EnumProfiles에서 이 인터페이스를 검색하고 등록된 프로필을 열거할 수 있습니다.

IEnumTfLanguageProfiles IEnumTfLanguageProfiles 인터페이스는 언어 프로필의 열거를 제공하기 위해 TSF 관리자에 의해 구현됩니다.
IEnumTfProperties IEnumTfProperties 인터페이스는 속성 개체의 열거형을 제공하기 위해 TSF 관리자에 의해 구현됩니다.
IEnumTfPropertyValue IEnumTfPropertyValue 인터페이스는 TSF 관리자에 의해 구현되어 속성 값의 열거형을 제공합니다.
IEnumTfRanges IEnumTfRanges 인터페이스는 범위 개체의 열거형을 제공하기 위해 TSF 관리자에 의해 구현됩니다.
IEnumTfUIElements IEnumTfUIElements 인터페이스는 TSF 관리자에 의해 구현되고 애플리케이션 또는 텍스트 서비스에서 사용됩니다. 이 인터페이스는 ITfUIElementMgr::EnumUIElements에서 검색하고 등록된 UI 요소를 열거할 수 있습니다.
ITextStoreACPServices ITextStoreACPServices 인터페이스는 ACP 기반 애플리케이션에 다양한 서비스를 제공하기 위해 TSF 관리자에 의해 구현됩니다.
ITfActiveLanguageProfileNotifySink ITfActiveLanguageProfileNotifySink 인터페이스는 활성 언어 또는 텍스트 서비스가 변경될 때 알림을 수신하기 위해 애플리케이션에 의해 구현됩니다.
ITfCandidateListUIElement ITfCandidateListUIElement 인터페이스는 후보 목록 UI가 있는 텍스트 서비스에 의해 구현됩니다.
ITfCandidateListUIElementBehavior 이 인터페이스는 후보 목록 UI가 있는 텍스트 서비스에 의해 구현되며 해당 UI는 애플리케이션에서 제어할 수 있습니다. 애플리케이션 QI ITfUIElement에서이 인터페이스 및 후보 목록 동작을 제어 합니다.
ITfCategoryMgr ITfCategoryMgr 인터페이스는 텍스트 서비스의 개체 범주를 관리합니다. TSF 관리자는 이 인터페이스를 구현합니다.

ITfCleanupContextDurationSink ITfCleanupContextDurationSink 인터페이스는 컨텍스트 정리 작업이 수행될 때 알림을 수신하기 위해 텍스트 서비스에 의해 구현됩니다.
ITfCleanupContextSink ITfCleanupContextSink 인터페이스는 컨텍스트 정리 작업이 발생할 때 알림을 수신하도록 텍스트 서비스에 의해 구현됩니다. 이 알림 싱크는 IID_ITfCleanupContextSink ITfSourceSingle::AdviseSingleSink를 호출하여 설치됩니다.
ITfClientId ITfClientId 인터페이스는 TSF 관리자에 의해 구현됩니다. 이 인터페이스는 TSF 개체에 대한 클라이언트 식별자를 가져오는 데 사용됩니다. 이 인터페이스의 instance IID_ITfClientId 사용하여 스레드 관리자를 쿼리하여 가져옵니다.
ITfCompartment ITfCompartment 인터페이스는 TSF 관리자에 의해 구현되며 클라이언트(애플리케이션 및 텍스트 서비스)에서 TSF 구획에서 데이터를 가져오고 설정하는 데 사용됩니다.
ITfCompartmentEventSink ITfCompartmentEventSink 인터페이스는 클라이언트(애플리케이션 또는 텍스트 서비스)에서 구현되며 TSF 관리자가 구획 데이터가 변경될 때 클라이언트에 알리는 데 사용됩니다.
ITfCompartmentMgr ITfCompartmentMgr 인터페이스는 TSF 관리자에 의해 구현되며 클라이언트(애플리케이션 및 텍스트 서비스)에서 TSF 구획을 가져오고 조작하는 데 사용됩니다.
ITfComposition ITfComposition 인터페이스는 TSF 관리자에 의해 구현되며 텍스트 서비스에서 컴퍼지션에 대한 데이터를 가져오고 종료하는 데 사용됩니다. 이 인터페이스의 instance ITfContextComposition::StartComposition 메서드에서 제공됩니다.
ITfCompositionSink ITfCompositionSink 인터페이스는 컴퍼지션이 종료될 때 알림을 수신하기 위해 텍스트 서비스에 의해 구현됩니다.
ITfCompositionView ITfCompositionView 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션에서 컴퍼지션 뷰에 대한 데이터를 가져오는 데 사용됩니다. 이 인터페이스의 instance ITfContextOwnerCompositionSink 메서드 중 하나에서 제공됩니다.

ITfConfigureSystemKeystrokeFeed ITfConfigureSystemKeystrokeFeed 인터페이스는 키 입력 처리를 사용하거나 사용하지 않도록 설정하기 위해 TSF 관리자에 의해 구현됩니다.
ITfContext ITfContext 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션 및 텍스트 서비스에서 편집 컨텍스트에 액세스하는 데 사용됩니다.
ITfContextComposition ITfContextComposition 인터페이스는 TSF 관리자에 의해 구현되며 텍스트 서비스에서 컴퍼지션을 만들고 조작하는 데 사용됩니다. 이 인터페이스의 instance ITfContext::QueryInterface에서 IID_ITfContextComposition 제공합니다.
ITfContextKeyEventSink ITfContextKeyEventSink 인터페이스는 입력 컨텍스트에서 발생하는 키보드 이벤트 알림을 수신하기 위해 텍스트 서비스에 의해 구현됩니다.
ITfContextOwner ITfContextOwner 인터페이스는 텍스트 저장소 없이 텍스트 입력을 수신하기 위해 애플리케이션 또는 텍스트 서비스에 의해 구현됩니다. 이 인터페이스의 instance 애플리케이션이 ITfSource::AdviseSink 메서드를 호출할 때 가져옵니다.
ITfContextOwnerCompositionServices ITfContextOwnerCompositionServices 인터페이스는 TSF 관리자에 의해 구현되며 컨텍스트 소유자가 텍스트 서비스에서 만든 컴퍼지션을 조작하는 데 사용됩니다.
ITfContextOwnerCompositionSink ITfContextOwnerCompositionSink 인터페이스는 컴퍼지션 관련 알림을 수신하기 위해 애플리케이션에 의해 구현됩니다.
ITfContextOwnerServices ITfContextOwnerServices 인터페이스는 관리자가 구현하고 컨텍스트 소유자 역할을 하는 텍스트 서비스 또는 애플리케이션에서 사용됩니다.
ITfContextView ITfContextView 인터페이스는 TSF 관리자에 의해 구현되며 클라이언트(애플리케이션 또는 텍스트 서비스)에서 컨텍스트 뷰에 대한 정보를 가져오는 데 사용됩니다.
ITfCreatePropertyStore

ITfCreatePropertyStore 인터페이스는 속성 저장소 데이터의 지속성을 지원하기 위해 텍스트 서비스에 의해 구현됩니다.
ITfDisplayAttributeInfo ITfDisplayAttributeInfo 인터페이스는 표시 특성 데이터를 제공하기 위해 텍스트 서비스에 의해 구현됩니다. 이 인터페이스는 텍스트가 표시되는 방식을 결정해야 하는 모든 구성 요소(가장 자주 애플리케이션)에서 사용됩니다.
ITfDisplayAttributeMgr ITfDisplayAttributeMgr 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션에서 디스플레이 특성을 가져오고 열거하는 데 사용됩니다. 개별 표시 특성은 ITfDisplayAttributeInfo 인터페이스를 통해 액세스됩니다.
ITfDisplayAttributeNotifySink ITfDisplayAttributeNotifySink 인터페이스는 디스플레이 특성 정보가 업데이트될 때 알림을 수신하도록 애플리케이션에 의해 구현됩니다.
ITfDisplayAttributeProvider ITfDisplayAttributeProvider 인터페이스는 텍스트 서비스에서 구현되며 TSF 관리자가 개별 표시 특성 정보 개체를 열거하고 가져오는 데 사용됩니다.
ITfDocumentMgr ITfDocumentMgr 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션 또는 텍스트 서비스에서 텍스트 컨텍스트를 만들고 관리하는 데 사용됩니다. 이 인터페이스의 instance 가져오려면 ITfThreadMgr::CreateDocumentMgr을 호출합니다.
ITfEditRecord ITfEditRecord 인터페이스는 TSF 관리자에 의해 구현되며 텍스트 편집 싱크에서 편집 세션 중에 변경된 내용을 확인하는 데 사용됩니다.
ITfEditSession ITfEditSession 인터페이스는 텍스트 서비스에 의해 구현되며 TSF 관리자가 컨텍스트의 텍스트 및 속성을 읽고/또는 수정하는 데 사용됩니다.
ITfEditTransactionSink ITfEditTransactionSink 인터페이스는 텍스트 서비스에 의해 구현되며 TSF 관리자가 편집 트랜잭션을 지원하는 데 사용됩니다.
ITfFunction ITfFunction 인터페이스는 개별 함수 인터페이스의 기본 인터페이스입니다.

ITfFunctionProvider ITfFunctionProvider 인터페이스는 다양한 함수 개체를 제공하기 위해 애플리케이션 또는 텍스트 서비스에 의해 구현됩니다.
ITfInputProcessorProfileActivationSink ITfInputProcessorProfileActivationSink 인터페이스는 프로필이 변경될 때 알림을 수신하기 위해 애플리케이션에 의해 구현됩니다.
ITfInputProcessorProfileMgr ITfInputProcessorProfileMgr 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션 또는 텍스트 서비스에서 하나 이상의 텍스트 서비스의 언어 프로필을 조작하는 데 사용됩니다.
ITfInputProcessorProfiles ITfInputProcessorProfiles 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션 또는 텍스트 서비스에서 하나 이상의 텍스트 서비스의 언어 프로필을 조작하는 데 사용됩니다.
ITfInputProcessorProfilesEx 이 인터페이스는 TSF 관리자에 의해 구현되며 텍스트 서비스 또는 애플리케이션에서 언어 프로필의 표시 설명을 설정하는 데 사용됩니다.
ITfInputProcessorProfileSubstituteLayout 이 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션 또는 텍스트 서비스에서 텍스트 서비스 프로필의 대체 입력 로캘 식별자(키보드 레이아웃)를 조작하는 데 사용됩니다.
ITfInsertAtSelection ITfInsertAtSelection 인터페이스는 관리자가 구현하며 텍스트 서비스에서 텍스트 또는 포함된 개체를 컨텍스트에 삽입하는 데 사용됩니다. 텍스트 서비스는 ITfContext::QueryInterface를 호출하여 이 인터페이스를 가져옵니다.
ITfKeyEventSink ITfKeyEventSink 인터페이스는 키보드 및 포커스 이벤트 알림을 수신하기 위해 텍스트 서비스에 의해 구현됩니다. 이 이벤트 싱크를 설치하려면 ITfKeystrokeMgr::AdviseKeyEventSink를 호출합니다.
ITfKeystrokeMgr ITfKeystrokeMgr 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션 및 텍스트 서비스에서 키보드 관리자와 상호 작용하는 데 사용됩니다.
ITfKeyTraceEventSink

ITfKeyTraceEventSink 인터페이스는 대상에 의해 이벤트가 처리되기 전에 키 스트로크 이벤트 알림을 수신하기 위해 애플리케이션 또는 텍스트 서비스에 의해 구현됩니다.
ITfLanguageProfileNotifySink ITfLanguageProfileNotifySink 인터페이스는 언어 프로필이 변경될 때 알림을 수신하기 위해 애플리케이션에 의해 구현됩니다.
ITfMessagePump ITfMessagePump 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션에서 애플리케이션 메시지 큐에서 메시지를 가져오는 데 사용됩니다.
ITfMouseSink ITfMouseSink 인터페이스는 마우스 이벤트 알림을 수신하기 위해 텍스트 서비스에 의해 구현됩니다. 마우스 이벤트 싱크는 ITfMouseTracker 인터페이스 중 하나의 ITfMouseTracker::AdviseMouseSink 메서드와 함께 설치됩니다.
ITfMouseTracker ITfMouseTracker 인터페이스는 TSF 관리자에 의해 구현되며 텍스트 서비스에서 마우스 이벤트 알림 싱크를 관리하는 데 사용됩니다. 이 인터페이스의 instance IID_ITfMouseTracker 대한 ITfContext 개체를 쿼리하여 가져옵니다.
ITfMouseTrackerACP ITfMouseTrackerACP 인터페이스는 마우스 이벤트 싱크를 지원하기 위해 애플리케이션에 의해 구현됩니다.
ITfPersistentPropertyLoaderACP ITfPersistentPropertyLoaderACP 인터페이스는 애플리케이션에서 구현되며 TSF 관리자가 속성을 비동기적으로 로드하는 데 사용합니다.
ITfPreservedKeyNotifySink ITfPreservedKeyNotifySink 인터페이스는 키가 유지되거나, 보존되지 않거나, 유지된 키 설명이 변경될 때 알림을 수신하기 위해 애플리케이션 또는 TSF 텍스트 서비스에 의해 구현됩니다.
ITfProperty ITfProperty 인터페이스는 TSF 관리자에 의해 구현되며 클라이언트(애플리케이션 또는 텍스트 서비스)에서 속성 값을 수정하는 데 사용됩니다.
ITfPropertyStore ITfPropertyStore 인터페이스는 텍스트 서비스에 의해 구현되며 TSF 관리자가 비정적 속성 값을 제공하는 데 사용됩니다. 이 인터페이스의 instance ITfProperty::SetValueStore에 전달됩니다.

ITfQueryEmbedded ITfQueryEmbedded 인터페이스는 TSF 관리자에 의해 구현되며 텍스트 서비스에서 컨텍스트가 포함된 개체를 허용할 수 있는지 여부를 결정하는 데 사용됩니다.
ITfRange ITfRange 인터페이스는 텍스트 서비스 및 애플리케이션에서 지정된 컨텍스트 내에서 텍스트를 참조하고 조작하는 데 사용됩니다. 인터페이스 ID가 IID_ITfRange.
ITfRangeACP ITfRangeACP 인터페이스는 TSF 관리자에 의해 구현되며 ACP(애플리케이션 문자 위치) 기반 애플리케이션에서 범위 개체에 액세스하고 조작하는 데 사용됩니다.
ITfRangeBackup ITfRangeBackup 인터페이스는 TSF 관리자에 의해 구현되며 텍스트 서비스에서 범위 개체에 포함된 데이터의 백업 복사본을 만드는 데 사용됩니다.
ITfReadingInformationUIElement ITfCandidateListUIElement 인터페이스는 가까운 캐럿에서 정보 UI를 읽기 위한 UI가 있는 텍스트 서비스에 의해 구현됩니다.
ITfReadOnlyProperty ITfReadOnlyProperty 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션 또는 텍스트 서비스에서 속성 데이터를 가져오는 데 사용됩니다.
ITfReverseConversion 지정된 문자열의 역방향 변환을 수행합니다.
ITfReverseConversionList 지정된 문자열을 만드는 데 필요한 키 입력 시퀀스의 목록을 나타냅니다.
ITfReverseConversionMgr 역방향 변환을 수행하는 데 사용되는 ITfReverseConversion 개체에 대한 액세스를 제공합니다.
ITfSource ITfSource 인터페이스는 TSF 관리자에 의해 구현됩니다. 애플리케이션 및 텍스트 서비스에서 advise 싱크를 설치하고 제거하는 데 사용됩니다.
ITfSourceSingle

ITfSourceSingle 인터페이스는 TSF 관리자에 의해 구현됩니다.
ITfStatusSink ITfStatusSink 인터페이스는 전역 문서 상태 변경 내용을 지원합니다. 이 조언 싱크는 IID_ITfStatusSink ITfSource::AdviseSink를 호출하여 설치됩니다. 텍스트 서비스는 필요에 따라 이 인터페이스를 구현할 수 있습니다.
ITfTextEditSink ITfTextEditSink 인터페이스는 읽기/쓰기 액세스와 관련된 편집 세션의 완료를 지원합니다.
ITfTextInputProcessor ITfTextInputProcessor 인터페이스는 텍스트 서비스에서 구현되며 TSF 관리자가 텍스트 서비스를 활성화하고 비활성화하는 데 사용됩니다.
ITfTextInputProcessorEx ITfTextInputProcessorEx 인터페이스는 텍스트 서비스에 의해 구현되며 TSF 관리자가 텍스트 서비스를 활성화하고 비활성화하는 데 사용됩니다.
ITfTextLayoutSink ITfTextLayoutSink 인터페이스는 애플리케이션의 컨텍스트 레이아웃 변경을 지원합니다. IID_ITfTextLayoutSink ITfSource::AdviseSink를 호출하여 이 조언 싱크를 설치합니다. 텍스트 서비스는 필요에 따라 이 인터페이스를 구현할 수 있습니다.
ITfThreadFocusSink ITfThreadFocusSink 인터페이스는 스레드가 UI 포커스를 받거나 잃을 때 알림을 수신하기 위해 애플리케이션 또는 TSF 텍스트 서비스에 의해 구현됩니다.
ITfThreadMgr ITfThreadMgr은 TSF 관리자가 구현한 기본 개체를 정의합니다. ITfThreadMgr은 애플리케이션 및 텍스트 서비스에서 텍스트 서비스를 활성화 및 비활성화하고, 문서 관리자를 만들고, 문서 컨텍스트 포커스를 유지하는 데 사용됩니다.
ITfThreadMgr2 ITfThreadMgr2는 TSF 관리자가 구현하는 기본 개체를 정의합니다. ITfThreadMgr2는 애플리케이션 및 텍스트 서비스에서 텍스트 서비스를 활성화 및 비활성화하고, 문서 관리자를 만들고, 문서 컨텍스트 포커스를 유지하는 데 사용됩니다.
ITfThreadMgrEventSink ITfThreadMgrEventSink 인터페이스는 애플리케이션 또는 TSF 텍스트 서비스에서 구현되어 특정 스

레드 관리자 이벤트에 대한 알림을 받습니다. IID_ITfThreadMgrEventSink TSF 관리자 ITfSource::AdviseSink에 전화하여 이 조언 싱크를 설치합니다.
ITfThreadMgrEx ITfThreadMgrEx 인터페이스는 애플리케이션에서 일부 플래그로 textservices를 활성화하는 데 사용됩니다. ITfThreadMgrEx는 ITfThreadMgr에서 QI로 가져올 수 있습니다.
ITfToolTipUIElement ITfToolTipUIElement 인터페이스는 UI에 도구 설명을 표시하려는 텍스트 서비스에 의해 구현됩니다.
ITfTransitoryExtensionSink ITfTransitoryExtensionSink 인터페이스는 일시적 확장을 흐리게 사용하는 애플리케이션에 의해 구현됩니다. 애플리케이션은 이 싱크 인터페이스를 사용하여 일시적인 확장에서 발생한 변경 내용을 추적할 수 있습니다.
ITfTransitoryExtensionUIElement ITfTransitoryExtensionUIElement 인터페이스는 일시적인 확장의 UI를 제공하는 TSF 관리자에 의해 구현됩니다.
ITfUIElement ITfUIElement 인터페이스는 UIElement 개체의 기본 인터페이스이며 텍스트 서비스에 의해 구현됩니다.
ITfUIElementMgr ITfUIElementMgr 인터페이스는 TSF 관리자에 의해 구현되며 애플리케이션 또는 텍스트 서비스에서 사용됩니다. 애플리케이션 및 텍스트 서비스는 IID_ITfUIElementMgr ITfThreadMgr::QueryInterface를 통해 이 인터페이스를 가져올 수 있습니다.
ITfUIElementSink ITfUIElementSink 인터페이스는 UI 요소가 변경될 때 알림을 수신하기 위해 애플리케이션에 의해 구현됩니다.

Functions

TF_CreateCategoryMgr TF_CreateCategoryMgr 함수는 COM을 초기화할 필요 없이 범주 관리자 개체를 만듭니다. 호출 스

레드는 MSCTF.DLL 소유하는 개체에 대한 참조 수를 유지해야 하므로 사용량을 신중하게 수행해야 합니다.
TF_CreateDisplayAttributeMgr TF_CreateDisplayAttributeMgr 함수는 COM을 초기화할 필요 없이 표시 특성 관리자 개체를 만드는 데 사용됩니다.
TF_CreateInputProcessorProfiles TF_CreateInputProcessorProfiles 함수는 COM을 초기화할 필요 없이 입력 프로세서 프로필 개체를 만드는 데 사용됩니다.
TF_CreateLangBarItemMgr TF_CreateLangBarItemMgr 함수는 COM을 초기화할 필요 없이 언어 모음 항목 관리자 개체를 만드는 데 사용됩니다.
TF_CreateLangBarMgr TF_CreateLangBarMgr 함수는 COM을 초기화하지 않고도 언어 모음 관리자 개체를 만듭니다. 호출 프로세스는 Msctf.dll 소유한 개체에 대한 적절한 참조 수를 유지해야 하므로 이 메서드를 사용하지 않는 것이 좋습니다.
TF_CreateThreadMgr TF_CreateThreadMgr 함수는 COM을 초기화할 필요 없이 스레드 관리자 개체를 만듭니다. 호출 프로세스는 Msctf.dll 소유한 개체에 대한 적절한 참조 수를 유지해야 하므로 이 메서드를 사용하지 않는 것이 좋습니다.
TF_GetThreadMgr TF_GetThreadMgr 함수는 호출 스레드 내에서 이전에 만든 스레드 관리자 개체의 복사본을 가져옵니다.

구조체

TF_DA_COLOR TF_DA_COLOR 구조체에는 텍스트 범위에 대한 표시 특성에 사용되는 색 데이터가 포함되어 있습니다.
TF_DISPLAYATTRIBUTE

TF_DISPLAYATTRIBUTE 구조에는 텍스트를 렌더링하기 위한 표시 특성 데이터가 포함되어 있습니다.
TF_HALTCOND TF_HALTCOND 구조체는 범위 이동 조건을 포함하는 데 사용됩니다.
TF_INPUTPROCESSORPROFILE 이 구조체에는 입력 프로세서 프로필에 대한 데이터가 포함됩니다.
TF_LANGUAGEPROFILE TF_LANGUAGEPROFILE 구조에는 언어 프로필에 대한 정보가 포함됩니다.
TF_PERSISTENT_PROPERTY_HEADER_ACP TF_PERSISTENT_PROPERTY_HEADER_ACP 구조체는 속성 헤더 데이터를 제공하는 데 사용됩니다.
TF_PRESERVEDKEY TF_PRESERVEDKEY 구조체는 유지된 키를 나타냅니다.
TF_PROPERTYVAL TF_PROPERTYVAL 구조에는 속성 값 데이터가 포함됩니다. 이 구조체는 IEnumTfPropertyValue::Next 메서드와 함께 사용됩니다.
TF_SELECTION TF_SELECTION 구조에는 텍스트 선택 데이터가 포함됩니다.
TF_SELECTIONSTYLE TF_SELECTIONSTYLE 구조체는 선택 영역의 스타일을 나타냅니다.

열거형

TF_DA_ATTR_INFO TF_DA_ATTR_INFO 열거형의 요소는 TF_DISPLAYATTRIBUTE 구조에서 텍스트 변환 데이터를 지정하는 데 사용됩니다.
TF_DA_COLORTYPE

TF_DA_COLORTYPE 열거형의 요소는 TF_DA_COLOR 구조에 포함된 색의 형식을 지정합니다.
TF_DA_LINESTYLE TF_DA_LINESTYLE 열거형의 요소는 TF_DA_COLOR 구조체에서 표시 특성의 밑줄 스타일을 지정합니다.
TfActiveSelEnd TfActiveSelEnd 열거형의 요소는 선택한 텍스트 범위의 끝을 활성 상태로 지정합니다.
TfAnchor TfAnchor 열거형의 요소는 ITfRange 개체의 시작 앵커 또는 끝 앵커를 지정합니다.
TfGravity TfGravity 열거형의 요소는 ITfRange 개체의 앵커와 연결된 중력 유형을 지정합니다.
TfLayoutCode TfLayoutCode 열거형의 요소는 ITfTextLayoutSink::OnLayoutChange 알림에서 레이아웃 변경 유형을 지정합니다.
TfShiftDir TfShiftDir 열거형의 요소는 범위 앵커가 이동되는 방향을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

msctfmonitorapi.h 헤더

아티클2023. 08. 24.

이 헤더는 Text Services Framework에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 텍스트 서비스 프레임워크 비교

msctfmonitorapi.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

InitLocalMsCtfMonitor InitLocalMsCtfMonitor 함수는 현재 데스크톱에서 TextServicesFramework를 초기화하고 필요한 경우 부동 언어 모음을 준비합니다. 이 함수는 앱의 데스크톱에서 호출되어야 합니다.
UninitLocalMsCtfMonitor UninitLocalMsCtfMonitor 함수는 현재 데스크톱에서 TextServicesFramework를 초기화하지 않습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

msdrm.h 헤더

아티클2023. 03. 08.

이 헤더는 Active Directory Rights Management Services SDK에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Active Directory Rights Management Services SDK

msdrm.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

DRMAcquireAdvisories 제출된 라이선스에 필요한 해지 목록을 검색합니다.
DRMAcquireIssuanceLicenseTemplate 서버에서 발급 라이선스 템플릿을 비동기적으로 검색합니다.
DRMAcquireLicense 최종 사용자 라이선스 또는 클라이언트 라이선스 인증서를 비동기적으로 획득하려고 시도합니다.
DRMActivate 컴퓨터에 대한 lockbox 및 컴퓨터 인증서 또는 사용자에 대한 권한 계정 인증서를 가져옵니다.
DRMAddLicense 임시 또는 영구 라이선스 저장소에 최종 사용자 라이선스를 추가합니다.
DRMAddRightWithUser 발급 라이선스의 사용자에게 권한을 할당합니다.
DRMAttest DRMAttest 함수는 더 이상 지원되지 않으며 E_NOTIMPL 반환합니다.
DRMCheckSecurity 실행 중인 모든 수준의 보안 검사에 대한 S_OK 반환합니다.

DRMClearAllRights 기존 발급 라이선스에서 모든 권한을 제거합니다.
DRMCloseEnvironmentHandle 환경 핸들을 닫습니다.
DRMCloseHandle DRMLoadLibrary를 사용하여 로드된 DRMCreate* 함수 및 라이브러리로 만든 개체에 대한 핸들을 닫습니다.
DRMClosePubHandle 이전에 만든 DRMPUBHANDLE을 닫습니다.
DRMCloseQueryHandle 언바운드 라이선스 개체에 대한 핸들을 닫습니다.
DRMCloseSession 클라이언트 세션 또는 라이선스 스토리지 세션을 닫습니다.
DRMConstructCertificateChain 임의의 수의 인증서에서 인증서 체인을 빌드합니다.
DRMCreateBoundLicense 애플리케이션에서 로컬로 저장된 라이선스에 대한 권한을 검사하거나 행사할 수 있습니다.
DRMCreateClientSession 라이선스 스토리지 세션을 호스트하고 활성화 및 기타 함수 호출에 사용되는 클라이언트 세션을 만듭니다.
DRMCreateEnablingBitsDecryptor 콘텐츠 데이터의 암호를 해독하는 데 사용되는 암호 해독 개체를 만듭니다.
DRMCreateEnablingBitsEncryptor 콘텐츠 데이터를 암호화하는 데 사용되는 AD RMS 암호화 개체를 만듭니다.
DRMCreateEnablingPrincipal 라이선스에 바인딩하는 데 필요한 사용 주체를 만듭니다.

DRMCreateIssuanceLicense 발급 라이선스를 처음부터, 템플릿에서 또는 서명된 발급 라이선스에서 만듭니다.
DRMCreateLicenseStorageSession 라이선스를 획득하거나 조작하는 데 필요한 라이선스 스토리지 세션을 만듭니다.
DRMCreateRight 사용자 또는 그룹에 부여된 권한을 정의하는 XrML 권한을 만듭니다.
DRMCreateUser 권한이 부여되는 사용자를 만듭니다.
DRMDecode base64와 같은 일반적인 알고리즘으로 인코딩된 문자열을 디코딩합니다.
DRMDeconstructCertificateChain 인증서 체인에서 지정된 인증서를 검색합니다.
DRMDecrypt 암호화된 콘텐츠의 암호를 해독합니다.
DRMDeleteLicense 라이선스, 클라이언트 라이선스 인증서, 해지 목록 또는 발급 라이선스 템플릿을 삭제합니다.
DRMDuplicateEnvironmentHandle 환경 핸들의 복사본을 만듭니다.
DRMDuplicateHandle DRMHANDLE의 복사본을 만듭니다.
DRMDuplicatePubHandle DRMPUBHANDLE의 복사본을 만듭니다.
DRMDuplicateSession 클라이언트 또는 라이선스 스토리지 세션을 복제합니다.

DRMEncode base64와 같은 공용 인코딩 메서드를 사용하여 데이터를 인코딩합니다.
DRMEncrypt 데이터를 암호화합니다.
DRMEnumerateLicense 유효한 라이선스, 컴퓨터 인증서 또는 권한 계정 인증서, 현재 사용자에 대한 해지 목록 또는 발급 라이선스 템플릿을 열거합니다.
DRMGetApplicationSpecificData 임의의 애플리케이션 관련 정보의 이름-값 쌍을 검색합니다.
DRMGetBoundLicenseAttribute 라이선스 XrML에서 바인딩된 라이선스 특성을 검색합니다.
DRMGetBoundLicenseAttributeCount 라이선스에서 특성 발생 횟수를 검색합니다.
DRMGetBoundLicenseObject 바인딩된 라이선스에서 개체를 반환합니다.
DRMGetBoundLicenseObjectCount 라이선스의 지정된 분기 내에 있는 개체의 발생 횟수를 검색합니다.
DRMGetCertificateChainCount 인증서 체인의 인증서 수를 검색합니다.
DRMGetClientVersion Active Directory Rights Management Services 클라이언트 소프트웨어의 버전 번호와 계층이 프로덕션 또는 사전 프로덕션용인지 여부를 반환합니다.
DRMGetEnvironmentInfo 보안 환경에 대한 정보를 반환합니다.
DRMGetInfo 개체 암호화 또는 암호 해독에 대한 정보를 검색합니다.

DRMGetIntervalTime 최종 사용자 라이선스를 갱신하기 전에 전달할 수 있는 발급에서 일 수를 검색합니다.
DRMGetIssuanceLicenseInfo 발급 라이선스에서 다양한 정보를 검색합니다.
DRMGetIssuanceLicenseTemplate 기존 발급 라이선스에서 발급 라이선스 템플릿을 가져옵니다.
DRMGetMetaData 발급 라이선스에서 메타데이터를 검색합니다.
DRMGetNameAndDescription 발급 라이선스에서 언어별 이름 및 설명을 검색합니다.
DRMGetOwnerLicense DRMGetSignedIssuanceLicense를 호출하여 만든 소유자 라이선스를 검색합니다.
DRMGetProcAddress 라이브러리에 있는 함수의 주소를 반환합니다. GetProcAddress 함수의 보안 버전입니다.
DRMGetRevocationPoint 발급 라이선스의 해지 지점에 대한 정보를 검색합니다.
DRMGetRightExtendedInfo 오른쪽에 연결된 사용자 지정 이름-값 쌍을 검색합니다.
DRMGetRightInfo 이전에 만든 권한에 대한 정보를 가져옵니다.
DRMGetSecurityProvider lockbox의 경로를 검색합니다.
DRMGetServiceLocation 정품 인증 또는 라이선스 취득과 같은 다양한 권한 관리 서비스를 수행할 수 있는 서버의 URL을 검색합니다.

DRMGetSignedIssuanceLicense 온라인으로 또는 오프라인으로 서명된 발급 라이선스를 획득하거나 나중에 서명할 수 있는 서명되지 않은 발급 라이선스를 생성합니다.
DRMGetSignedIssuanceLicenseEx 오프라인으로 서명된 발급 라이선스를 획득합니다.
DRMGetTime 권한 관리 시스템에서 보안 시간을 요청합니다.
DRMGetUnboundLicenseAttribute 기본 XrML에서 언바운드 라이선스 특성을 검색합니다.
DRMGetUnboundLicenseAttributeCount 언바운드 라이선스의 개체 내 특성 발생 횟수를 검색합니다.
DRMGetUnboundLicenseObject 언바운드 라이선스에서 지정된 형식의 개체를 검색합니다.
DRMGetUnboundLicenseObjectCount 라이선스의 지정된 분기 내에 있는 개체의 인스턴스 수를 계산합니다.
DRMGetUsagePolicy 애플리케이션 이름, 버전 또는 기타 애플리케이션 특성에 따라 콘텐츠에 대한 액세스를 요구하거나 거부하는 사용 정책을 가져옵니다.
DRMGetUserInfo 사용자에 대한 정보를 가져옵니다.
DRMGetUserRights 발급 라이선스에서 사용자/오른쪽 쌍을 검색합니다.
DRMGetUsers 발급 라이선스에서 특정 사용자를 검색합니다.
DRMInitEnvironment 모든 권한 관리 호출에 대한 보안 환경을 만듭니다.

DRMIsActivated 현재 사용자 또는 컴퓨터가 활성화되었는지 여부를 나타냅니다.
DRMIsWindowProtected 창이 보호된 환경과 연결되어 있는지 여부를 나타냅니다.
DRMLoadLibrary 자격 증명에 의해 결정된 대로 승인된 라이브러리에 핸들을 로드합니다.
DRMParseUnboundLicense 애플리케이션이 개체 및 특성을 탐색할 수 있도록 언바운드 라이선스에 대한 핸들을 만듭니다.
DRMRegisterContent AD RMS로 보호된 문서가 있거나 더 이상 표시되지 않는다는 것을 AD RMS(Active Directory Rights Management Services) 클라이언트에 알릴 수 있습니다.
DRMRegisterProtectedWindow 보호된 환경에 창을 등록합니다.
DRMRegisterRevocationList 클라이언트에서 권한 해지 목록을 등록합니다.
DRMRepair 컴퓨터 또는 사용자에 대해 이전에 만든 인증서를 삭제하여 클라이언트 컴퓨터를 복구합니다.
DRMSetApplicationSpecificData 발급 라이선스가 콘텐츠 사용 애플리케이션에서 사용할 임의의 이름-값 쌍을 저장할 수 있도록 허용합니다.
DRMSetGlobalOptions 전송 프로토콜을 지정된 값으로 설정하고 선택적으로 서버 lockbox를 사용할지 여부를 지정합니다.
DRMSetIntervalTime 최종 사용자 라이선스를 갱신하기까지 경과할 수 있는 발급 일 수를 지정합니다.

DRMSetMetaData 발급 라이선스에 애플리케이션별 메타데이터를 추가합니다.
DRMSetNameAndDescription 애플리케이션에서 라이선스의 이름과 설명을 여러 언어로 지정할 수 있습니다.
DRMSetRevocationPoint 해지 목록을 가져올 새로 고침 빈도 및 위치를 설정합니다.
DRMSetUsagePolicy 애플리케이션 이름, 버전 또는 기타 환경 특성에 따라 콘텐츠에 대한 액세스를 요구하거나 거부하는 사용 정책을 설정합니다.
DRMVerify 더 이상 지원되지 않으며 E_NOTIMPL 반환합니다.

msdrmdefs.h 헤더

아티클2023. 08. 24.

이 헤더는 Active Directory Rights Management Services SDK에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Active Directory Rights Management Services SDK

msdrmdefs.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

콜백 함수

DRMCALLBACK AD RMS SDK에 포함된 일부 함수는 구현해야 하는 콜백 함수를 사용하여 애플리케이션에 상태 정보와 라이선스를 제공합니다. 콜백 구문은 다음과 같습니다.

구조체

DRM_ACTSERV_INFO DRM_ACTSERV_INFO 구조체는 활성화 서버에 대한 정보를 저장합니다.
DRM_CLIENT_VERSION_INFO AD RMS(Active Directory Rights Management Services) 클라이언트 및 계층 구조(예: 프로덕션 또는 사전 프로덕션)의 버전에 대한 정보를 받습니다.
DRM_LICENSE_ACQ_DATA 비실용 라이선스 취득 중에 라이선스 취득 데이터를 보유합니다.
DRMBOUNDLICENSEPARAMS DRMCreateBoundLicense에서 라이선스에 바인딩하는 데 사용됩니다.
DRMID 개체를 식별합니다.

열거형

DRM_DISTRIBUTION_POINT_INFO DRMGetIssuanceLicenseInfo를 호출할 때 정보를 검색할 배포 지점의 유형을 지정합니다.
DRM_STATUS_MSG 사용자 지정 콜백 함수에서 콜백 함수가 호출되는 이유를 지정하는 데 사용됩니다.
DRM_USAGEPOLICY_TYPE DRMGetUsagePolicy 및 DRMSetUsagePolicy 함수와 함께 사용 정책 유형을 지정하는 데 사용됩니다.
DRMATTESTTYPE DRMATTESTTYPE 열거형은 데이터 Blob에 대해 만들 서명의 종류를 지정합니다.
DRMENCODINGTYPE DRMENCODINGTYPE 열거형은 라이선스에 사용되는 가능한 인코딩 유형을 식별합니다.
DRMGLOBALOPTIONS 전송 프로토콜에 사용되는 프로토콜과 서버 lockbox가 사용되는지 여부를 지정하기 위한 값을 정의합니다. 이 열거형은 DRMSetGlobalOptions 함수에서 사용됩니다.
DRMSECURITYPROVIDERTYPE DRMSECURITYPROVIDERTYPE 열거형은 사용되는 보안 DRM 환경의 유형을 지정합니다.
DRMSPECTYPE DRMSPECTYPE 열거형은 사용되는 보안 또는 라이브러리 공급자 유형을 나타냅니다.
DRMTIMETYPE DRMTIMETYPE 열거형은 시간 형식을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

msi.h 헤더

아티클2023. 03. 16.

이 헤더는 애플리케이션 설치 및 서비스에 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 애플리케이션 설치 및 서비스

msi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

MsiAdvertiseProductA MsiAdvertiseProduct 함수는 보급 스크립트를 생성하거나 컴퓨터에 제품을 보급합니다. (ANSI)
MsiAdvertiseProductExA MsiAdvertiseProductEx 함수는 보급 스크립트를 생성하거나 컴퓨터에 제품을 보급합니다. (ANSI)
MsiAdvertiseProductExW MsiAdvertiseProductEx 함수는 보급 스크립트를 생성하거나 컴퓨터에 제품을 보급합니다. (유니코드)
MsiAdvertiseProductW MsiAdvertiseProduct 함수는 보급 스크립트를 생성하거나 컴퓨터에 제품을 보급합니다. (유니코드)
MsiAdvertiseScriptA MsiAdvertiseScript 함수는 보급된 스크립트 파일을 지정된 위치에 복사합니다. (ANSI)
MsiAdvertiseScriptW MsiAdvertiseScript 함수는 보급된 스크립트 파일을 지정된 위치에 복사합니다. (유니코드)
MsiApplyMultiplePatchesA 패치를 받을 수 있는 제품에 하나 이상의 패치를 적용합니다. (ANSI)
MsiApplyMultiplePatchesW 패치를 받을 수 있는 제품에 하나 이상의 패치를 적용합니다. (유니코드)

MsiApplyPatchA 패치 패키지를 받을 수 있는 자격으로 패치 패키지에 나열된 각 제품에 대해 MsiApplyPatch 함수는 설치를 호출하고 PATCH 속성을 패치 패키지의 경로로 설정합니다. (ANSI)
MsiApplyPatchW 패치 패키지를 받을 수 있는 자격으로 패치 패키지에 나열된 각 제품에 대해 MsiApplyPatch 함수는 설치를 호출하고 PATCH 속성을 패치 패키지의 경로로 설정합니다. (유니코드)
MsiBeginTransactionA MsiBeginTransaction 함수는 다중 패키지 설치의 트랜잭션 처리를 시작하고 트랜잭션에 대한 식별자를 반환합니다. (ANSI)
MsiBeginTransactionW MsiBeginTransaction 함수는 다중 패키지 설치의 트랜잭션 처리를 시작하고 트랜잭션에 대한 식별자를 반환합니다. (유니코드)
MsiCloseAllHandles MsiCloseAllHandles 함수는 현재 스레드에서 할당한 열려 있는 모든 설치 핸들을 닫습니다. 이 함수는 진단 함수이며 정리에 사용하면 안 됩니다.
MsiCloseHandle MsiCloseHandle 함수는 열린 설치 핸들을 닫습니다.
MsiCollectUserInfoA MsiCollectUserInfo 함수는 설치 마법사에서 사용자 정보 및 제품 ID를 가져오고 저장합니다. (ANSI)
MsiCollectUserInfoW MsiCollectUserInfo 함수는 설치 마법사에서 사용자 정보 및 제품 ID를 가져오고 저장합니다. (유니코드)
MsiConfigureFeatureA MsiConfigureFeature 함수는 제품 기능에 대해 설치된 상태를 구성합니다. (ANSI)
MsiConfigureFeatureW MsiConfigureFeature 함수는 제품 기능에 대해 설치된 상태를 구성합니다. (유니코드)
MsiConfigureProductA MsiConfigureProduct 함수는 제품을 설치하거나 제거합니다. (ANSI)

MsiConfigureProductExA 제품을 설치하거나 제거합니다. (MsiConfigureProductExA)
MsiConfigureProductExW 제품을 설치하거나 제거합니다. (MsiConfigureProductExW)
MsiConfigureProductW MsiConfigureProduct 함수는 제품을 설치하거나 제거합니다. (유니코드)
MsiDetermineApplicablePatchesA MsiDetermineApplicablePatches 함수는 패치 파일, XML 파일 및 XML Blob 집합을 가져와서 지정된 Windows Installer 패키지 및 시퀀스에 적용되는 패치를 결정합니다. (ANSI)
MsiDetermineApplicablePatchesW MsiDetermineApplicablePatches 함수는 패치 파일, XML 파일 및 XML Blob 집합을 가져와서 지정된 Windows Installer 패키지 및 시퀀스에 적용되는 패치를 결정합니다. (유니코드)
MsiDeterminePatchSequenceA 패치 파일, XML 파일 및 XML Blob 집합을 가져와서 지정된 설치된 제품에 대한 패치에 가장 적합한 애플리케이션 시퀀스를 결정합니다. (ANSI)
MsiDeterminePatchSequenceW 패치 파일, XML 파일 및 XML Blob 집합을 가져와서 지정된 설치된 제품에 대한 패치에 가장 적합한 애플리케이션 시퀀스를 결정합니다. (유니코드)
MsiEnableLogA MsiEnableLog 함수는 호출 프로세스에서 시작된 모든 후속 설치에 대한 로그 모드를 설정합니다. (ANSI)
MsiEnableLogW MsiEnableLog 함수는 호출 프로세스에서 시작된 모든 후속 설치에 대한 로그 모드를 설정합니다. (유니코드)
MsiEndTransaction MsiEndTransaction 함수는 MsiBeginTransaction 함수에서 연 트랜잭션에 속하는 모든 설치를 커밋하거나 롤백할 수 있습니다.

MsiEnumClientsA MsiEnumClients 함수는 지정된 설치된 구성 요소에 대한 클라이언트를 열거합니다. 함수는 호출할 때마다 하나의 제품 코드를 검색합니다. (ANSI)
MsiEnumClientsExA MsiEnumClientsEx 함수는 지정된 구성 요소를 사용하는 설치된 애플리케이션을 열거합니다. 함수는 호출할 때마다 애플리케이션에 대한 제품 코드를 검색합니다. (ANSI)
MsiEnumClientsExW MsiEnumClientsEx 함수는 지정된 구성 요소를 사용하는 설치된 애플리케이션을 열거합니다. 함수는 호출할 때마다 애플리케이션에 대한 제품 코드를 검색합니다. (유니코드)
MsiEnumClientsW MsiEnumClients 함수는 지정된 설치된 구성 요소에 대한 클라이언트를 열거합니다. 함수는 호출할 때마다 하나의 제품 코드를 검색합니다. (유니코드)
MsiEnumComponentQualifiersA MsiEnumComponentQualifiers 함수는 지정된 구성 요소에 대해 보급된 한정자를 열거합니다. 이 함수는 호출할 때마다 한정자를 검색합니다. (ANSI)
MsiEnumComponentQualifiersW MsiEnumComponentQualifiers 함수는 지정된 구성 요소에 대해 보급된 한정자를 열거합니다. 이 함수는 호출할 때마다 한정자를 검색합니다. (유니코드)
MsiEnumComponentsA MsiEnumComponents 함수는 모든 제품에 대해 설치된 구성 요소를 열거합니다. 이 함수는 호출할 때마다 하나의 구성 요소 코드를 검색합니다. (ANSI)
MsiEnumComponentsExA MsiEnumComponentsEx 함수는 설치된 구성 요소를 열거합니다. 함수는 호출할 때마다 한 구성 요소에 대한 구성 요소 코드를 검색합니다. 구성 요소 코드는 구성 요소, 버전 및 언어에 고유한 문자열 GUID입니다. (ANSI)
MsiEnumComponentsExW MsiEnumComponentsEx 함수는 설치된 구성 요소를 열거합니다. 함수는 호출할 때마다 한 구성 요소에 대한 구성 요소 코드를 검색합니다. 구성 요소 코드는 구성 요소, 버전 및 언어에 고유한 문자열 GUID입니다. (유니코드)

MsiEnumComponentsW MsiEnumComponents 함수는 모든 제품에 대해 설치된 구성 요소를 열거합니다. 이 함수는 호출할 때마다 하나의 구성 요소 코드를 검색합니다. (유니코드)
MsiEnumFeaturesA MsiEnumFeatures 함수는 지정된 제품에 대해 게시된 기능을 열거합니다. 이 함수는 호출할 때마다 하나의 기능 ID를 검색합니다. (ANSI)
MsiEnumFeaturesW MsiEnumFeatures 함수는 지정된 제품에 대해 게시된 기능을 열거합니다. 이 함수는 호출할 때마다 하나의 기능 ID를 검색합니다. (유니코드)
MsiEnumPatchesA MsiEnumPatches 함수는 제품에 적용된 모든 패치를 열거합니다. (ANSI)
MsiEnumPatchesExA 특정 컨텍스트 또는 모든 컨텍스트에서 모든 패치를 열거합니다. (ANSI)
MsiEnumPatchesExW 특정 컨텍스트 또는 모든 컨텍스트에서 모든 패치를 열거합니다. (유니코드)
MsiEnumPatchesW MsiEnumPatches 함수는 제품에 적용된 모든 패치를 열거합니다. (유니코드)
MsiEnumProductsA MsiEnumProducts 함수는 현재 보급되거나 설치된 모든 제품을 열거합니다. 사용자별 및 컴퓨터별 설치 컨텍스트와 보급 알림 모두에 설치된 제품이 열거됩니다. (ANSI)
MsiEnumProductsExA 지정된 컨텍스트에서 현재 보급되거나 설치된 제품의 하나 또는 모든 인스턴스를 열거합니다. (ANSI)
MsiEnumProductsExW 지정된 컨텍스트에서 현재 보급되거나 설치된 제품의 하나 또는 모든 인스턴스를 열거합니다. (유니코드)

MsiEnumProductsW MsiEnumProducts 함수는 현재 보급되거나 설치된 모든 제품을 열거합니다. 사용자별 및 컴퓨터별 설치 컨텍스트와 보급 알림 모두에 설치된 제품이 열거됩니다. (유니코드)
MsiEnumRelatedProductsA MsiEnumRelatedProducts 함수는 지정된 업그레이드 코드를 사용하여 제품을 열거합니다. 이 함수는 속성 테이블에 지정된 UpgradeCode 속성이 있는 현재 설치 및 보급된 제품을 나열합니다. (ANSI)
MsiEnumRelatedProductsW MsiEnumRelatedProducts 함수는 지정된 업그레이드 코드를 사용하여 제품을 열거합니다. 이 함수는 속성 테이블에 지정된 UpgradeCode 속성이 있는 현재 설치 및 보급된 제품을 나열합니다. (유니코드)
MsiExtractPatchXMLDataA MsiExtractPatchXMLData 함수는 패치가 대상 시스템에 적용되는지 여부를 확인하는 데 사용할 수 있는 패치에서 정보를 추출합니다. (ANSI)
MsiExtractPatchXMLDataW MsiExtractPatchXMLData 함수는 패치가 대상 시스템에 적용되는지 여부를 확인하는 데 사용할 수 있는 패치에서 정보를 추출합니다. (유니코드)
MsiGetComponentPathA MsiGetComponentPath 함수는 설치된 구성 요소의 전체 경로를 반환합니다. 구성 요소의 키 경로가 레지스트리 키인 경우 레지스트리 키가 반환됩니다. (ANSI)
MsiGetComponentPathExA 설치된 구성 요소의 전체 경로를 반환합니다. (ANSI)
MsiGetComponentPathExW 설치된 구성 요소의 전체 경로를 반환합니다. (유니코드)
MsiGetComponentPathW MsiGetComponentPath 함수는 설치된 구성 요소의 전체 경로를 반환합니다. 구성 요소의 키 경로가 레지스트리 키인 경우 레지스트리 키가 반환됩니다. (유니코드)
MsiGetFeatureInfoA 기능에 대한 설명 정보를 반환합니다. (ANSI)

MsiGetFeatureInfoW 기능에 대한 설명 정보를 반환합니다. (유니코드)
MsiGetFeatureUsageA MsiGetFeatureUsage 함수는 제품 기능에 대한 사용 메트릭을 반환합니다. (ANSI)
MsiGetFeatureUsageW MsiGetFeatureUsage 함수는 제품 기능에 대한 사용 메트릭을 반환합니다. (유니코드)
MsiGetFileHashA MsiGetFileHash 함수는 파일 경로를 사용하여 해당 파일의 128비트 해시를 반환합니다. 제작 도구는 MsiGetFileHash를 사용하여 MsiFileHash 테이블을 채우는 데 필요한 파일 해시를 가져올 수 있습니다. (ANSI)
MsiGetFileHashW MsiGetFileHash 함수는 파일 경로를 사용하여 해당 파일의 128비트 해시를 반환합니다. 제작 도구는 MsiGetFileHash를 사용하여 MsiFileHash 테이블을 채우는 데 필요한 파일 해시를 가져올 수 있습니다. (유니코드)
MsiGetFileSignatureInformationA MsiGetFileSignatureInformation 함수는 디지털 서명된 파일의 경로를 가져와서 파일의 서명자 인증서 및 해시를 반환합니다. (ANSI)
MsiGetFileSignatureInformationW MsiGetFileSignatureInformation 함수는 디지털 서명된 파일의 경로를 가져와서 파일의 서명자 인증서 및 해시를 반환합니다. (유니코드)
MsiGetFileVersionA MsiGetFileVersion은 설치 관리자가 데이터베이스에서 찾을 것으로 예상되는 형식으로 버전 문자열 및 언어 문자열을 반환합니다. (ANSI)
MsiGetFileVersionW MsiGetFileVersion은 설치 관리자가 데이터베이스에서 찾을 것으로 예상되는 형식으로 버전 문자열 및 언어 문자열을 반환합니다. (유니코드)
MsiGetPatchFileListA MsiGetPatchFileList 함수는 세미콜론으로 구분된 .msp 파일 목록을 제공하고 패치로 업데이트할 수 있는 파일 목록을 검색합니다. (ANSI)

MsiGetPatchFileListW MsiGetPatchFileList 함수는 세미콜론으로 구분된 .msp 파일 목록을 제공하고 패치로 업데이트할 수 있는 파일 목록을 검색합니다. (유니코드)
MsiGetPatchInfoA MsiGetPatchInfo 함수는 패치에 대한 정보를 반환합니다. (ANSI)
MsiGetPatchInfoExA 제품의 지정된 인스턴스에 패치를 적용하는 방법에 대한 정보를 쿼리합니다. (ANSI)
MsiGetPatchInfoExW 제품의 지정된 인스턴스에 패치를 적용하는 방법에 대한 정보를 쿼리합니다. (유니코드)
MsiGetPatchInfoW MsiGetPatchInfo 함수는 패치에 대한 정보를 반환합니다. (유니코드)
MsiGetProductCodeA MsiGetProductCode 함수는 애플리케이션의 설치되거나 보급된 구성 요소의 구성 요소 코드를 사용하여 애플리케이션의 제품 코드를 반환합니다. (ANSI)
MsiGetProductCodeW MsiGetProductCode 함수는 애플리케이션의 설치되거나 보급된 구성 요소의 구성 요소 코드를 사용하여 애플리케이션의 제품 코드를 반환합니다. (유니코드)
MsiGetProductInfoA MsiGetProductInfo 함수는 게시된 제품 및 설치된 제품에 대한 제품 정보를 반환합니다. (ANSI)
MsiGetProductInfoExA 보급된 제품 및 설치된 제품의 제품 정보를 반환합니다. (ANSI)
MsiGetProductInfoExW 보급된 제품 및 설치된 제품의 제품 정보를 반환합니다. (유니코드)
MsiGetProductInfoFromScriptA MsiGetProductInfoFromScript 함수는 Windows Installer 스크립트 파일에 대한 제품 정보를 반환합니다. (ANSI)

MsiGetProductInfoFromScriptW MsiGetProductInfoFromScript 함수는 Windows Installer 스크립트 파일에 대한 제품 정보를 반환합니다. (유니코드)
MsiGetProductInfoW MsiGetProductInfo 함수는 게시된 제품 및 설치된 제품에 대한 제품 정보를 반환합니다. (유니코드)
MsiGetProductPropertyA MsiGetProductProperty 함수는 제품 속성을 검색합니다. 이러한 속성은 제품 데이터베이스에 있습니다. (ANSI)
MsiGetProductPropertyW MsiGetProductProperty 함수는 제품 속성을 검색합니다. 이러한 속성은 제품 데이터베이스에 있습니다. (유니코드)
MsiGetShortcutTargetA MsiGetShortcutTarget 함수는 바로 가기를 검사하고 제품, 기능 이름 및 구성 요소를 반환합니다(사용 가능한 경우). (ANSI)
MsiGetShortcutTargetW MsiGetShortcutTarget 함수는 바로 가기를 검사하고 제품, 기능 이름 및 구성 요소를 반환합니다(사용 가능한 경우). (유니코드)
MsiGetUserInfoA MsiGetUserInfo 함수는 설치된 제품에 대해 등록된 사용자 정보를 반환합니다. (ANSI)
MsiGetUserInfoW MsiGetUserInfo 함수는 설치된 제품에 대해 등록된 사용자 정보를 반환합니다. (유니코드)
MsiInstallMissingComponentA MsiInstallMissingComponent 함수는 예기치 않게 누락된 파일을 설치합니다. (ANSI)
MsiInstallMissingComponentW MsiInstallMissingComponent 함수는 예기치 않게 누락된 파일을 설치합니다. (유니코드)
MsiInstallMissingFileA MsiInstallMissingFile 함수는 예기치 않게 누락된 파일을 설치합니다. (ANSI)

MsiInstallMissingFileW MsiInstallMissingFile 함수는 예기치 않게 누락된 파일을 설치합니다. (유니코드)
MsiInstallProductA 제품을 설치하거나 제거합니다. (MsiInstallProductA)
MsiInstallProductW 제품을 설치하거나 제거합니다. (MsiInstallProductW)
MsiIsProductElevatedA MsiIsProductElevated 함수는 제품이 관리되는지 여부를 반환합니다. (ANSI)
MsiIsProductElevatedW MsiIsProductElevated 함수는 제품이 관리되는지 여부를 반환합니다. (유니코드)
MsiJoinTransaction MsiJoinTransaction 함수는 Windows Installer가 현재 프로세스를 다중 패키지 설치를 설치하는 트랜잭션의 소유자로 만들도록 요청합니다.
MsiLocateComponentA MsiLocateComponent 함수는 제품 코드 없이 설치된 구성 요소의 전체 경로를 반환합니다. (ANSI)
MsiLocateComponentW MsiLocateComponent 함수는 제품 코드 없이 설치된 구성 요소의 전체 경로를 반환합니다. (유니코드)
MsiNotifySidChangeA MsiNotifySidChange 함수는 사용자 SID에 대한 변경 내용으로 Windows Installer 내부 정보를 알리고 업데이트합니다. (ANSI)
MsiNotifySidChangeW MsiNotifySidChange 함수는 사용자 SID에 대한 변경 내용으로 Windows Installer 내부 정보를 알리고 업데이트합니다. (유니코드)
MsiOpenPackageA MsiOpenPackage 함수는 제품 데이터베이스에 액세스하는 함수와 함께 사용할 패키지를 엽니다. (ANSI)

MsiOpenPackageExA MsiOpenPackageEx 함수는 제품 데이터베이스에 액세스하는 함수와 함께 사용할 패키지를 엽니다. (ANSI)
MsiOpenPackageExW MsiOpenPackageEx 함수는 제품 데이터베이스에 액세스하는 함수와 함께 사용할 패키지를 엽니다. (유니코드)
MsiOpenPackageW MsiOpenPackage 함수는 제품 데이터베이스에 액세스하는 함수와 함께 사용할 패키지를 엽니다. (유니코드)
MsiOpenProductA MsiOpenProduct 함수는 제품 데이터베이스에 액세스하는 함수와 함께 사용할 제품을 엽니다. 핸들이 더 이상 필요하지 않은 경우 MsiCloseHandle 함수를 핸들로 호출해야 합니다. (ANSI)
MsiOpenProductW MsiOpenProduct 함수는 제품 데이터베이스에 액세스하는 함수와 함께 사용할 제품을 엽니다. 핸들이 더 이상 필요하지 않은 경우 MsiCloseHandle 함수를 핸들로 호출해야 합니다. (유니코드)
MsiProcessAdvertiseScriptA MsiProcessAdvertiseScript 함수는 지정된 위치에 보급 스크립트 파일을 처리합니다. (ANSI)
MsiProcessAdvertiseScriptW MsiProcessAdvertiseScript 함수는 지정된 위치에 보급 스크립트 파일을 처리합니다. (유니코드)
MsiProvideAssemblyA MsiProvideAssembly 함수는 어셈블리가 포함된 Windows Installer 구성 요소의 전체 경로를 반환합니다. 이 함수는 원본을 묻는 메시지를 표시하고 필요한 모든 설치를 수행합니다. MsiProvideAssembly는 기능에 대한 사용 횟수를 증분합니다. (ANSI)
MsiProvideAssemblyW MsiProvideAssembly 함수는 어셈블리가 포함된 Windows Installer 구성 요소의 전체 경로를 반환합니다. 이 함수는 원본을 묻는 메시지를 표시하고 필요한 모든 설치를 수행합니다. MsiProvideAssembly는 기능에 대한 사용 횟수를 증분합니다. (유니코드)
MsiProvideComponentA MsiProvideComponent 함수는 필요한 설치를 수행하여 전체 구성 요소 경로를 반환합니다. 이 함수는 필요한 경우 원본을 묻는 메시지를 표시하고 기능에 대한 사용 횟수를 증분합니다. (ANSI)

MsiProvideComponentW MsiProvideComponent 함수는 필요한 설치를 수행하여 전체 구성 요소 경로를 반환합니다. 이 함수는 필요한 경우 원본을 묻는 메시지를 표시하고 기능에 대한 사용 횟수를 증분합니다. (유니코드)
MsiProvideQualifiedComponentA MsiProvideQualifiedComponent 함수는 정규화된 구성 요소에 대한 전체 구성 요소 경로를 반환하고 필요한 모든 설치를 수행합니다. 이 함수는 필요한 경우 원본을 묻는 메시지를 표시하고 기능에 대한 사용 횟수를 증분합니다. (ANSI)
MsiProvideQualifiedComponentExA MsiProvideQualifiedComponentEx 함수는 제품에서 게시하고 필요한 모든 설치를 수행하는 정규화된 구성 요소의 전체 구성 요소 경로를 반환합니다. (ANSI)
MsiProvideQualifiedComponentExW MsiProvideQualifiedComponentEx 함수는 제품에서 게시하고 필요한 모든 설치를 수행하는 정규화된 구성 요소의 전체 구성 요소 경로를 반환합니다. (유니코드)
MsiProvideQualifiedComponentW MsiProvideQualifiedComponent 함수는 정규화된 구성 요소에 대한 전체 구성 요소 경로를 반환하고 필요한 모든 설치를 수행합니다. 이 함수는 필요한 경우 원본을 묻는 메시지를 표시하고 기능에 대한 사용 횟수를 증분합니다. (유니코드)
MsiQueryComponentStateA MsiQueryComponentState 함수는 구성 요소에 대해 설치된 상태를 반환합니다. (ANSI)
MsiQueryComponentStateW MsiQueryComponentState 함수는 구성 요소에 대해 설치된 상태를 반환합니다. (유니코드)
MsiQueryFeatureStateA MsiQueryFeatureState 함수는 제품 기능에 대해 설치된 상태를 반환합니다. (ANSI)
MsiQueryFeatureStateExA MsiQueryFeatureStateEx 함수는 제품 기능에 대해 설치된 상태를 반환합니다. (ANSI)
MsiQueryFeatureStateExW MsiQueryFeatureStateEx 함수는 제품 기능에 대해 설치된 상태를 반환합니다. (유니코드)

MsiQueryFeatureStateW MsiQueryFeatureState 함수는 제품 기능에 대해 설치된 상태를 반환합니다. (유니코드)
MsiQueryProductStateA MsiQueryProductState 함수는 제품에 대해 설치된 상태를 반환합니다. (ANSI)
MsiQueryProductStateW MsiQueryProductState 함수는 제품에 대해 설치된 상태를 반환합니다. (유니코드)
MsiReinstallFeatureA 기능을 다시 설치합니다. (ANSI)
MsiReinstallFeatureW 기능을 다시 설치합니다. (유니코드)
MsiReinstallProductA 제품을 다시 설치합니다. (ANSI)
MsiReinstallProductW 제품을 다시 설치합니다. (유니코드)
MsiRemovePatchesA 단일 제품에서 하나 이상의 패치를 제거합니다. (ANSI)
MsiRemovePatchesW 단일 제품에서 하나 이상의 패치를 제거합니다. (유니코드)
MsiSetExternalUIA MsiSetExternalUI 함수는 외부 사용자 인터페이스 처리기를 사용하도록 설정합니다. (ANSI)
MsiSetExternalUIRecord MsiSetExternalUIRecord 함수는 외부 UI(사용자 인터페이스) 처리기를 사용하도록 설정합니다.
MsiSetExternalUIW MsiSetExternalUI 함수는 외부 사용자 인터페이스 처리기를 사용하도록 설정합니다. (유니코드)

MsiSetInternalUI MsiSetInternalUI 함수는 설치 관리자의 내부 사용자 인터페이스를 사용하도록 설정합니다. 그런 다음 이 사용자 인터페이스는 이 프로세스에서 사용자 인터페이스 생성 설치 관리자 함수에 대한 모든 후속 호출에 사용됩니다. 자세한 내용은 사용자 인터페이스 수준을 참조하세요.
MsiSourceListAddMediaDiskA MsiSourceListAddMediaDisk 함수는 등록된 제품 또는 패치의 미디어 원본 디스크를 추가하거나 업데이트합니다. (ANSI)
MsiSourceListAddMediaDiskW MsiSourceListAddMediaDisk 함수는 등록된 제품 또는 패치의 미디어 원본 디스크를 추가하거나 업데이트합니다. (유니코드)
MsiSourceListAddSourceA 지정된 사용자 컨텍스트에서 제품 또는 패치에 대해 지정된 유형의 원본을 포함하는 유효한 네트워크 원본 목록에 추가합니다. (ANSI)
MsiSourceListAddSourceExA 지정된 컨텍스트에서 패치 또는 제품의 원본 집합을 추가하거나 순서를 변경합니다. 지정된 컨텍스트에 없는 패치에 대한 원본 목록을 만들 수도 있습니다. (ANSI)
MsiSourceListAddSourceExW 지정된 컨텍스트에서 패치 또는 제품의 원본 집합을 추가하거나 순서를 변경합니다. 지정된 컨텍스트에 없는 패치에 대한 원본 목록을 만들 수도 있습니다. (유니코드)
MsiSourceListAddSourceW 지정된 사용자 컨텍스트에서 제품 또는 패치에 대해 지정된 유형의 원본을 포함하는 유효한 네트워크 원본 목록에 추가합니다. (유니코드)
MsiSourceListClearAllA MsiSourceListClearAll 함수는 지정된 컨텍스트의 패치 또는 제품의 원본 목록에서 모든 네트워크 원본을 제거합니다. 자세한 내용은 원본 복원력을 참조하세요. (ANSI)
MsiSourceListClearAllExA MsiSourceListClearAllEx 함수는 지정된 제품 또는 패치 인스턴스에 대해 지정된 소스 형식의 모든 기존 원본을 제거합니다. (ANSI)

MsiSourceListClearAllExW MsiSourceListClearAllEx 함수는 지정된 제품 또는 패치 인스턴스에 대해 지정된 소스 형식의 모든 기존 원본을 제거합니다. (유니코드)
MsiSourceListClearAllW MsiSourceListClearAll 함수는 지정된 컨텍스트의 패치 또는 제품의 원본 목록에서 모든 네트워크 원본을 제거합니다. 자세한 내용은 원본 복원력을 참조하세요. (유니코드)
MsiSourceListClearMediaDiskA MsiSourceListClearMediaDisk 함수는 특정 컨텍스트에서 제품 또는 패치에 대한 미디어 소스에서 기존 등록된 디스크를 제거하는 기능을 제공합니다. (ANSI)
MsiSourceListClearMediaDiskW MsiSourceListClearMediaDisk 함수는 특정 컨텍스트에서 제품 또는 패치에 대한 미디어 소스에서 기존 등록된 디스크를 제거하는 기능을 제공합니다. (유니코드)
MsiSourceListClearSourceA 지정된 컨텍스트에서 제품 또는 패치에 대한 기존 소스를 제거합니다. (ANSI)
MsiSourceListClearSourceW 지정된 컨텍스트에서 제품 또는 패치에 대한 기존 소스를 제거합니다. (유니코드)
MsiSourceListEnumMediaDisksA MsiSourceListEnumMediaDisks 함수는 패치 또는 제품의 미디어 원본에 등록된 디스크 목록을 열거합니다. (ANSI)
MsiSourceListEnumMediaDisksW MsiSourceListEnumMediaDisks 함수는 패치 또는 제품의 미디어 원본에 등록된 디스크 목록을 열거합니다. (유니코드)
MsiSourceListEnumSourcesA MsiSourceListEnumSources 함수는 지정된 패치 또는 제품의 원본 목록에 있는 원본을 열거합니다. (ANSI)
MsiSourceListEnumSourcesW MsiSourceListEnumSources 함수는 지정된 패치 또는 제품의 원본 목록에 있는 원본을 열거합니다. (유니코드)

MsiSourceListForceResolutionA MsiSourceListForceResolution 함수는 다음에 원본이 필요할 때 설치 관리자가 원본 목록에서 유효한 제품 원본을 검색하도록 합니다. (ANSI)
MsiSourceListForceResolutionExA MsiSourceListForceResolutionEx 함수는 "LastUsedSource"라는 속성의 등록을 제거합니다. (ANSI)
MsiSourceListForceResolutionExW MsiSourceListForceResolutionEx 함수는 "LastUsedSource"라는 속성의 등록을 제거합니다. (유니코드)
MsiSourceListForceResolutionW MsiSourceListForceResolution 함수는 다음에 원본이 필요할 때 설치 관리자가 원본 목록에서 유효한 제품 원본을 검색하도록 합니다. (유니코드)
MsiSourceListGetInfoA MsiSourceListGetInfo 함수는 특정 컨텍스트에서 제품 또는 패치의 원본 목록에 대한 정보를 검색합니다. (ANSI)
MsiSourceListGetInfoW MsiSourceListGetInfo 함수는 특정 컨텍스트에서 제품 또는 패치의 원본 목록에 대한 정보를 검색합니다. (유니코드)
MsiSourceListSetInfoA 특정 컨텍스트에서 제품 또는 패치의 원본 목록에 대한 정보를 설정합니다. (ANSI)
MsiSourceListSetInfoW 특정 컨텍스트에서 제품 또는 패치의 원본 목록에 대한 정보를 설정합니다. (유니코드)
MsiUseFeatureA MsiUseFeature 함수는 특정 기능에 대한 사용 횟수를 증가시키고 해당 기능에 대한 설치 상태를 나타냅니다. 이 함수는 기능을 사용하려는 애플리케이션의 의도를 나타내는 데 사용해야 합니다. (ANSI)
MsiUseFeatureExA MsiUseFeatureEx 함수는 특정 기능에 대한 사용 횟수를 증가시키고 해당 기능에 대한 설치 상태를 나타냅니다. 이 함수는 기능을 사용하려는 애플리케이션의 의도를 나타내는 데 사용해야 합니다. (ANSI)

MsiUseFeatureExW MsiUseFeatureEx 함수는 특정 기능에 대한 사용 횟수를 증가시키고 해당 기능에 대한 설치 상태를 나타냅니다. 이 함수는 기능을 사용하려는 애플리케이션의 의도를 나타내는 데 사용해야 합니다. (유니코드)
MsiUseFeatureW MsiUseFeature 함수는 특정 기능에 대한 사용 횟수를 증가시키고 해당 기능에 대한 설치 상태를 나타냅니다. 이 함수는 기능을 사용하려는 애플리케이션의 의도를 나타내는 데 사용해야 합니다. (유니코드)
MsiVerifyPackageA MsiVerifyPackage 함수는 지정된 파일이 설치 패키지인지 확인합니다. (ANSI)
MsiVerifyPackageW MsiVerifyPackage 함수는 지정된 파일이 설치 패키지인지 확인합니다. (유니코드)

콜백 함수

INSTALLUI_HANDLER_RECORD INSTALLUI_HANDLER_RECORD 함수 프로토타입은 설치 관리자가 진행률 알림 및 오류 메시지를 호출하는 콜백 함수를 정의합니다. MsiSetExternalUIRecord 함수를 호출하여 레코드 기반 UI(외부 사용자 인터페이스) 처리기를 사용하도록 설정합니다.
INSTALLUI_HANDLERA INSTALLUI_HANDLER 함수 프로토타입은 설치 관리자가 진행률 알림 및 오류 메시지를 호출하는 콜백 함수를 정의합니다. (ANSI)
INSTALLUI_HANDLERW INSTALLUI_HANDLER 함수 프로토타입은 설치 관리자가 진행률 알림 및 오류 메시지를 호출하는 콜백 함수를 정의합니다. (유니코드)

구조체

MSIFILEHASHINFO MSIFILEHASHINFO 구조체에는 MsiGetFileHash에서 반환되고 MsiFileHash 테이블에 사용되는 파일 해시 정보가 포함됩니다.
MSIPATCHSEQUENCEINFOA MSIPATCHSEQUENCEINFO 구조체는 MsiDeterminePatchSequence 및 MsiDetermineApplicablePatches 함수에서 사용됩니다. (ANSI)
MSIPATCHSEQUENCEINFOW MSIPATCHSEQUENCEINFO 구조체는 MsiDeterminePatchSequence 및 MsiDetermineApplicablePatches 함수에서 사용됩니다. (유니코드)

msime.h 헤더

아티클2023. 08. 24.

이 헤더는 Windows 애플리케이션에 대한 국제화에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 애플리케이션에 대한 국제화

msime.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IFECommon IFECommon 인터페이스는 다양한 언어에 공통적인 IME 관련 서비스를 제공합니다.
IFEDictionary IFEDictionary 인터페이스를 사용하면 클라이언트가 Microsoft IME 사용자 사전에 액세스할 수 있습니다.
IFELanguage IFELanguage 인터페이스는 Microsoft IME를 사용하여 언어 처리 서비스를 제공합니다.

Functions

CreateIFECommonInstance IFECommon 인터페이스에 대한 포인터를 반환합니다.
CreateIFEDictionaryInstance IFEDictionary 인터페이스에 대한 포인터를 반환합니다.
CreateIFELanguageInstance IFELanguage 인터페이스에 대한 포인터를 반환합니다.

구조체

IMEDLG Microsoft IME의 사전 도구를 호출하거나 앱에서 대화 상자 창을 Word 때 사용됩니다.
IMESHF 열린 사용자 사전 파일의 헤더입니다. 버전, 제목, 설명 및 저작권과 같은 사용자 사전의 속성을 가져오는 데 사용됩니다.
IMEWRD Microsoft IME 사전의 Word 데이터에 있는 단어에 대한 데이터를 포함합니다.
POSTBL 공용 POS(음성의 일부) 테이블의 항목입니다.

열거형

IMEUCT IMEWRD 구조의 사용자 주석 유형입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

msimeapi.h 헤더

아티클2023. 03. 08.

이 헤더는 Windows 애플리케이션에 대한 Internationalization에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 애플리케이션에 대한 국제화

msimeapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IImePlugInDictDictionaryList IME 플러그 인 사전 목록에 대한 액세스를 제공합니다.

msinkaut.h 헤더

아티클2023. 03. 08.

이 헤더는 태블릿 PC에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Tablet PC

msinkaut.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IInkCollector . (IInkCollector)
IInkCursor 태블릿 커서에 대한 일반 정보를 나타냅니다.
IInkCursorButton 태블릿에서 장치를 가리키고 선택하는 단추에 대한 일반적인 정보를 나타냅니다.
IInkCursorButtons IInkCursor 인터페이스에 대한 IInkCursorButton 개체의 컬렉션을 나타냅니다.
IInkCursors IInkCursor 개체의 컬렉션을 나타냅니다.
IInkCustomStrokes 사용자 정의 InkStrokes 컬렉션의 컬렉션을 포함합니다.
IInkDisp . (IInkDisp)
IInkDrawingAttributes . (IInkDrawingAttributes)

IInkExtendedProperties 애플리케이션 정의 데이터를 포함하는 IInkExtendedProperty 개체의 컬렉션을 나타냅니다.
IInkExtendedProperty 태블릿 PC 개체 모델 내의 다양한 개체에 사용자 고유의 데이터를 추가하는 기능을 나타냅니다.
IInkGesture 제스처 인식에서 반환된 제스처의 특정 속성을 쿼리하는 기능을 나타냅니다.
IInkLineInfo IInkLineInfo 인터페이스는 텍스트 잉크 개체(tInk)의 표시 속성 및 인식 결과 목록에 대한 액세스를 제공합니다.
IInkOverlay . (IInkOverlay)
IInkPicture . (IInkPicture)
IInkRecognitionAlternate 인식기 사전과 비교되는 잉크 세그먼트에 대해 가능한 단어 일치를 나타냅니다.
IInkRecognitionAlternates 잉크 세그먼트에 대해 가능한 단어 일치를 나타내는 IInkRecognitionAlternate 개체를 포함합니다.
IInkRecognitionResult 인식의 결과를 나타냅니다. 필기 잉크를 인식하는 결과는 IInkRecognitionResult 개체에 반환됩니다.
IInkRecognizer 잉크 또는 필기를 처리하고 스트로크를 텍스트 또는 제스처로 변환하는 기능을 나타냅니다. 인식기는 실제 필기 인식을 수행하는 데 사용되는 InkRecognizerContext 개체를 만듭니다.
IInkRecognizer2 IInkWordList2 인터페이스에 멤버를 추가합니다.

IInkRecognizerContext . (IInkRecognizerContext)
IInkRecognizerContext2 InkRecognizerContext 클래스에 멤버를 추가합니다.
IInkRecognizerGuide . (IInkRecognizerGuide)
IInkRecognizers . (IInkRecognizers)
IInkRectangle . (IInkRectangle)
IInkRenderer . (IInkRenderer)
IInkStrokeDisp 단일 잉크 스트로크를 나타냅니다. 스트로크는 디지타이저가 알려진 잉크 표시의 좌표와 속성을 나타내는 캡처하는 속성 및 점 데이터 집합입니다.
IInkStrokes . (IInkStrokes)
IInkTablet 태블릿 디바이스 메시지 또는 이벤트를 수신하는 태블릿 PC의 디지타이저 디바이스를 나타냅니다.
IInkTablet2 IInkTablet 인터페이스를 확장합니다.
IInkTablet3 입력에 대해 다중 터치 쿼리를 사용하도록 설정합니다.
IInkTablets . (IInkTablets)

IInkTransform . (IInkTransform)
IInkWordList . (IInkWordList)
IInkWordList2 InkWordList 클래스에 멤버를 추가합니다.

구조체

INKMETRIC 텍스트 잉크 개체(tInk)의 표시 속성을 지정합니다.
InkRecoGuide 더 이상 사용되지 않습니다. 인식 가이드에 대한 정보를 나타냅니다. 대신 WritingBox 속성, DrawnBox 속성, Rows 속성, Columns 속성 및 Midline 속성 [InkRecognizerGuide 클래스] 속성을 사용합니다.

열거형

InkApplicationGesture 애플리케이션별 제스처 집합에 대한 관심을 설정하는 값을 정의합니다. 애플리케이션 제스처는 애플리케이션을 지원하도록 선택할 수 있는 제스처입니다.
InkBoundingBoxMode 잉크의 경계 상자를 계산하는 데 사용되는 그리기 특성과 같은 스트로크의 특성을 지정합니다. 경계 상자는 InkDisp 개체의 모든 점을 포함하는 가장 작은 사각형입니다.
InkClipboardFormats 클립보드에 저장된 잉크의 형식을 지정합니다.

InkClipboardModes 클립보드의 복사 옵션을 지정합니다.
InkCollectionMode 잉크, 제스처 또는 잉크 및 제스처가 사용자가 쓸 때 인식되는지 여부를 결정하는 값을 정의합니다.
InkCollectorEventInterest 잉크 수집기에서 이벤트가 발생했는지 여부와 발생한 이벤트를 지정하는 데 사용되는 값을 정의합니다.
InkCursorButtonState 커서 단추의 상태를 지정하는 값을 정의합니다.
InkExtractFlags InkDisp 개체에서 스트로크를 제거하는 방법을 결정합니다.
InkMouseButton 누른 마우스 단추를 지정합니다. (InkMouseButton)
InkMousePointer 표시할 마우스 포인터의 유형을 지정합니다.
InkOverlayAttachMode 현재 레이어 뒤 또는 앞에 새 InkOverlay 개체를 연결할 위치를 지정합니다.
InkOverlayEditingMode InkOverlay 개체 및 InkPicture 컨트롤의 동작 모드를 지정합니다.
InkOverlayEraserMode InkOverlay 개체 및 InkPicture 컨트롤에서 잉크를 지우는 방법을 지정합니다. 이 모드는 InkOverlayEditingMode가 Delete로 설정된 경우에 사용됩니다.
InkPenTip 펜 팁 도형을 지정합니다.
InkPersistenceCompressionMode InkDisp 개체를 직렬화된 형식으로 저장하는 데 사용되는 압축 모드의 값을 정의합니다.

InkPersistenceFormat 잉크를 유지하는 방법을 지정합니다.
InkPictureSizeMode 그림이 InkPicture 컨트롤 내에서 동작하는 방식을 지정합니다.
InkRasterOperation 그려진 잉크에서 래스터 작업을 수행하기 위한 값을 정의합니다.
InkRecognitionAlternatesSelection 구현되지 않았습니다. (InkRecognitionAlternatesSelection)
InkRecognitionConfidence 인식기에서 인식 결과에 있는 신뢰도 수준을 나타냅니다.
InkRecognitionModes 인식기가 잉크를 해석하고 결과 문자열을 결정하는 방법을 지정합니다.
InkRecognitionStatus 인식 중에 오류가 발생했는지 여부와 오류가 발생한 경우를 지정합니다.
InkRecognizerCapabilities 인식기의 특성을 지정합니다. 또한 이 열거형을 사용하여 설치된 인식기를 검색할 때 사용할 특성을 결정합니다.
InkRecognizerCharacterAutoCompletionMode 문자 입력 모드의 형식을 지정합니다.
InkSelectionConstants 점 또는 패킷 값 그룹 내의 첫 번째 요소 또는 모든 요소를 사용할지 여부를 지정합니다.
InkShiftKeyModifierFlags 누른 한정자 키를 지정합니다.
InkSystemGesture 운영 체제별 제스처 집합에 대한 관심을 지정합니다.

SelectionHitResult 적중 횟수 테스트 중에 적중된 선택 영역의 일부(있는 경우)를 지정합니다.
TabletHardwareCapabilities 태블릿 PC의 하드웨어 기능을 지정합니다.
TabletPropertyMetricUnit 속성의 측정 단위를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

msinkaut15.h 헤더

아티클2023. 03. 08.

이 헤더는 태블릿 PC에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Tablet PC

msinkaut15.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IInkDivider . (IInkDivider)
IInkDivisionResult InkDivider 개체에 포함된 스트로크 컬렉션의 레이아웃 분석을 나타냅니다.
IInkDivisionUnit IInkDivisionResult 개체 내의 단일 구조 요소를 나타냅니다.
IInkDivisionUnits IInkDivisionResult 개체에 포함된 IInkDivisionUnit 개체의 컬렉션을 포함합니다.

열거형

InkDivisionType IInkDivisionResult 개체 내의 구조 형식에 대한 값을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

msiquery.h 헤더

아티클2023. 03. 16.

이 헤더는 애플리케이션 설치 및 서비스에 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 애플리케이션 설치 및 서비스

msiquery.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

MsiCreateRecord MsiCreateRecord 함수는 지정된 수의 필드를 사용하여 새 레코드 개체를 만듭니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다.
MsiCreateTransformSummaryInfoA MsiCreateTransformSummaryInfo 함수는 유효성 검사 및 오류 조건을 포함하도록 기존 변환의 요약 정보를 만듭니다. 이 함수를 실행하면 MsiGetLastErrorRecord를 사용하여 액세스할 수 있는 오류 레코드가 설정됩니다. (ANSI)
MsiCreateTransformSummaryInfoW MsiCreateTransformSummaryInfo 함수는 유효성 검사 및 오류 조건을 포함하도록 기존 변환의 요약 정보를 만듭니다. 이 함수를 실행하면 MsiGetLastErrorRecord를 사용하여 액세스할 수 있는 오류 레코드가 설정됩니다. (유니코드)
MsiDatabaseApplyTransformA MsiDatabaseApplyTransform 함수는 데이터베이스에 변환을 적용합니다. (ANSI)
MsiDatabaseApplyTransformW MsiDatabaseApplyTransform 함수는 데이터베이스에 변환을 적용합니다. (유니코드)
MsiDatabaseCommit MsiDatabaseCommit 함수는 데이터베이스에 변경 내용을 커밋합니다.
MsiDatabaseExportA MsiDatabaseExport 함수는 열려 있는 데이터베이스에서 텍스트 보관 파일로 Microsoft Installer 테이블을 내보냅니다. (ANSI)

MsiDatabaseExportW MsiDatabaseExport 함수는 열려 있는 데이터베이스에서 텍스트 보관 파일로 Microsoft Installer 테이블을 내보냅니다. (유니코드)
MsiDatabaseGenerateTransformA MsiDatabaseGenerateTransform 함수는 두 데이터베이스 간의 차이점이 있는 변환 파일을 생성합니다. (ANSI)
MsiDatabaseGenerateTransformW MsiDatabaseGenerateTransform 함수는 두 데이터베이스 간의 차이점이 있는 변환 파일을 생성합니다. (유니코드)
MsiDatabaseGetPrimaryKeysA MsiDatabaseGetPrimaryKeys 함수는 지정된 테이블에 대한 모든 기본 키 열의 이름을 포함하는 레코드를 반환합니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다. (ANSI)
MsiDatabaseGetPrimaryKeysW MsiDatabaseGetPrimaryKeys 함수는 지정된 테이블에 대한 모든 기본 키 열의 이름을 포함하는 레코드를 반환합니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다. (유니코드)
MsiDatabaseImportA MsiDatabaseImport 함수는 설치 관리자 텍스트 보관 파일을 열려 있는 데이터베이스 테이블로 가져옵니다. (ANSI)
MsiDatabaseImportW MsiDatabaseImport 함수는 설치 관리자 텍스트 보관 파일을 열려 있는 데이터베이스 테이블로 가져옵니다. (유니코드)
MsiDatabaseIsTablePersistentA MsiDatabaseIsTablePersistent 함수는 특정 테이블의 상태를 설명하는 열거형을 반환합니다. (ANSI)
MsiDatabaseIsTablePersistentW MsiDatabaseIsTablePersistent 함수는 특정 테이블의 상태를 설명하는 열거형을 반환합니다. (유니코드)
MsiDatabaseMergeA MsiDatabaseMerge 함수는 두 데이터베이스를 병합하여 중복 행을 허용합니다. (ANSI)

MsiDatabaseMergeW MsiDatabaseMerge 함수는 두 데이터베이스를 병합하여 중복 행을 허용합니다. (유니코드)
MsiDatabaseOpenViewA MsiDatabaseOpenView 함수는 데이터베이스 쿼리를 준비하고 뷰 개체를 만듭니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다. (ANSI)
MsiDatabaseOpenViewW MsiDatabaseOpenView 함수는 데이터베이스 쿼리를 준비하고 뷰 개체를 만듭니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다. (유니코드)
MsiDoActionA MsiDoAction 함수는 기본 제공 작업, 사용자 지정 작업 또는 사용자 인터페이스 마법사 작업을 실행합니다. (ANSI)
MsiDoActionW MsiDoAction 함수는 기본 제공 작업, 사용자 지정 작업 또는 사용자 인터페이스 마법사 작업을 실행합니다. (유니코드)
MsiEnableUIPreview MsiEnableUIPreview 함수를 사용하면 사용자 인터페이스의 미리 보기 모드를 사용하여 사용자 인터페이스 대화 상자를 쉽게 작성할 수 있습니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다.
MsiEnumComponentCostsA MsiEnumComponentCosts 함수는 구성 요소를 설치하는 데 필요한 드라이브당 디스크 공간을 열거합니다. (ANSI)
MsiEnumComponentCostsW MsiEnumComponentCosts 함수는 구성 요소를 설치하는 데 필요한 드라이브당 디스크 공간을 열거합니다. (유니코드)
MsiEvaluateConditionA MsiEvaluateCondition 함수는 속성 이름과 값을 포함하는 조건식을 평가합니다. (ANSI)
MsiEvaluateConditionW MsiEvaluateCondition 함수는 속성 이름과 값을 포함하는 조건식을 평가합니다. (유니코드)

MsiFormatRecordA MsiFormatRecord 함수는 형식 문자열을 사용하여 레코드 필드 데이터 및 속성의 형식을 지정합니다. (ANSI)
MsiFormatRecordW MsiFormatRecord 함수는 형식 문자열을 사용하여 레코드 필드 데이터 및 속성의 형식을 지정합니다. (유니코드)
MsiGetActiveDatabase MsiGetActiveDatabase 함수는 설치에 대한 활성 데이터베이스를 반환합니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 읽기 전용 핸들을 반환합니다.
MsiGetComponentStateA MsiGetComponentState 함수는 구성 요소의 상태를 가져옵니다. (ANSI)
MsiGetComponentStateW MsiGetComponentState 함수는 구성 요소의 상태를 가져옵니다. (유니코드)
MsiGetDatabaseState MsiGetDatabaseState 함수는 데이터베이스의 상태를 반환합니다.
MsiGetFeatureCostA MsiGetFeatureCost 함수는 기능 및 선택한 자식 및 부모 기능에 필요한 디스크 공간을 반환합니다. (ANSI)
MsiGetFeatureCostW MsiGetFeatureCost 함수는 기능 및 선택한 자식 및 부모 기능에 필요한 디스크 공간을 반환합니다. (유니코드)
MsiGetFeatureStateA MsiGetFeatureState 함수는 기능의 요청된 상태를 가져옵니다. (ANSI)
MsiGetFeatureStateW MsiGetFeatureState 함수는 기능의 요청된 상태를 가져옵니다. (유니코드)
MsiGetFeatureValidStatesA MsiGetFeatureValidStates 함수는 유효한 설치 상태를 반환합니다. (ANSI)

MsiGetFeatureValidStatesW MsiGetFeatureValidStates 함수는 유효한 설치 상태를 반환합니다. (유니코드)
MsiGetLanguage MsiGetLanguage 함수는 현재 실행 중인 설치의 숫자 언어를 반환합니다.
MsiGetLastErrorRecord MsiGetLastErrorRecord 함수는 호출 프로세스에 대해 마지막으로 반환된 오류 레코드를 반환합니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다.
MsiGetMode MsiGetMode 함수는 테이블에 나열된 대로 설치 관리자가 현재 지정된 모드에서 실행 중인지 여부를 확인하는 데 사용됩니다.
MsiGetPropertyA MsiGetProperty 함수는 설치 관리자 속성의 값을 가져옵니다. (ANSI)
MsiGetPropertyW MsiGetProperty 함수는 설치 관리자 속성의 값을 가져옵니다. (유니코드)
MsiGetSourcePathA MsiGetSourcePath 함수는 디렉터리 테이블의 폴더에 대한 전체 원본 경로를 반환합니다. (ANSI)
MsiGetSourcePathW MsiGetSourcePath 함수는 디렉터리 테이블의 폴더에 대한 전체 원본 경로를 반환합니다. (유니코드)
MsiGetSummaryInformationA MsiGetSummaryInformation 함수는 설치 관리자 데이터베이스에 대한 _SummaryInformation 스트림에 대한 핸들을 가져옵니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다. (ANSI)
MsiGetSummaryInformationW MsiGetSummaryInformation 함수는 설치 관리자 데이터베이스에 대한 _SummaryInformation 스트림에 대한 핸들을 가져옵니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다. (유니코드)

MsiGetTargetPathA MsiGetTargetPath 함수는 디렉터리 테이블의 폴더에 대한 전체 대상 경로를 반환합니다. (ANSI)
MsiGetTargetPathW MsiGetTargetPath 함수는 디렉터리 테이블의 폴더에 대한 전체 대상 경로를 반환합니다. (유니코드)
MsiOpenDatabaseA MsiOpenDatabase 함수는 데이터 액세스를 위한 데이터베이스 파일을 엽니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다. (ANSI)
MsiOpenDatabaseW MsiOpenDatabase 함수는 데이터 액세스를 위한 데이터베이스 파일을 엽니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다. (유니코드)
MsiPreviewBillboardA MsiPreviewBillboard 함수는 표시된 대화 상자에 호스트 컨트롤이 있는 광고판을 표시합니다. (ANSI)
MsiPreviewBillboardW MsiPreviewBillboard 함수는 표시된 대화 상자에 호스트 컨트롤이 있는 광고판을 표시합니다. (유니코드)
MsiPreviewDialogA MsiPreviewDialog 함수는 대화 상자를 모덜리스 및 비활성으로 표시합니다. (ANSI)
MsiPreviewDialogW MsiPreviewDialog 함수는 대화 상자를 모덜리스 및 비활성으로 표시합니다. (유니코드)
MsiProcessMessage MsiProcessMessage 함수는 처리를 위해 설치 관리자에 오류 레코드를 보냅니다.
MsiRecordClearData MsiRecordClearData 함수는 레코드의 모든 필드를 null로 설정합니다.
MsiRecordDataSize MsiRecordDataSize 함수는 레코드 필드의 길이를 반환합니다. 개수에는 종료 null 문자가 포함되지 않습니다.

MsiRecordGetFieldCount 레코드의 필드 수를 반환합니다.
MsiRecordGetInteger MsiRecordGetInteger 함수는 레코드 필드에서 정수 값을 반환합니다.
MsiRecordGetStringA MsiRecordGetString 함수는 레코드 필드의 문자열 값을 반환합니다. (ANSI)
MsiRecordGetStringW MsiRecordGetString 함수는 레코드 필드의 문자열 값을 반환합니다. (유니코드)
MsiRecordIsNull null 레코드 필드를 보고합니다.
MsiRecordReadStream MsiRecordReadStream 함수는 레코드 스트림 필드에서 버퍼로 바이트를 읽습니다.
MsiRecordSetInteger 레코드 필드를 정수 필드로 설정합니다.
MsiRecordSetStreamA MsiRecordSetStream 함수는 파일에서 레코드 스트림 필드를 설정합니다. 스트림 데이터는 임시 필드에 삽입할 수 없습니다. (ANSI)
MsiRecordSetStreamW MsiRecordSetStream 함수는 파일에서 레코드 스트림 필드를 설정합니다. 스트림 데이터는 임시 필드에 삽입할 수 없습니다. (유니코드)
MsiRecordSetStringA MsiRecordSetString 함수는 지정된 필드에 문자열을 복사합니다. (ANSI)
MsiRecordSetStringW MsiRecordSetString 함수는 지정된 필드에 문자열을 복사합니다. (유니코드)
MsiSequenceA MsiSequence 함수는 지정된 테이블에 설명된 대로 다른 작업 시퀀스를 실행합니다. (ANSI)

MsiSequenceW MsiSequence 함수는 지정된 테이블에 설명된 대로 다른 작업 시퀀스를 실행합니다. (유니코드)
MsiSetComponentStateA MsiSetComponentState 함수는 구성 요소를 요청된 상태로 설정합니다. (ANSI)
MsiSetComponentStateW MsiSetComponentState 함수는 구성 요소를 요청된 상태로 설정합니다. (유니코드)
MsiSetFeatureAttributesA MsiSetFeatureAttributes 함수는 런타임에 기능의 기본 특성을 수정할 수 있습니다. 기능의 기본 특성은 기능 테이블의 특성 열에 작성됩니다. (ANSI)
MsiSetFeatureAttributesW MsiSetFeatureAttributes 함수는 런타임에 기능의 기본 특성을 수정할 수 있습니다. 기능의 기본 특성은 기능 테이블의 특성 열에 작성됩니다. (유니코드)
MsiSetFeatureStateA MsiSetFeatureState 함수는 기능을 지정된 상태로 설정합니다. (ANSI)
MsiSetFeatureStateW MsiSetFeatureState 함수는 기능을 지정된 상태로 설정합니다. (유니코드)
MsiSetInstallLevel MsiSetInstallLevel 함수는 전체 제품 설치에 대한 설치 수준을 설정합니다.
MsiSetMode MsiSetMode 함수는 내부 엔진 부울 상태를 설정합니다.
MsiSetPropertyA MsiSetProperty 함수는 설치 속성의 값을 설정합니다. (ANSI)
MsiSetPropertyW MsiSetProperty 함수는 설치 속성의 값을 설정합니다. (유니코드)
MsiSetTargetPathA MsiSetTargetPath 함수는 디렉터리 테이블의 폴더에 대한 전체 대상 경로를 설정합니다. (ANSI)

MsiSetTargetPathW MsiSetTargetPath 함수는 디렉터리 테이블의 폴더에 대한 전체 대상 경로를 설정합니다. (유니코드)
MsiSummaryInfoGetPropertyA MsiSummaryInfoGetProperty 함수는 요약 정보 스트림에서 단일 속성을 가져옵니다. (ANSI)
MsiSummaryInfoGetPropertyCount MsiSummaryInfoGetPropertyCount 함수는 요약 정보 스트림의 기존 속성 수를 반환합니다.
MsiSummaryInfoGetPropertyW MsiSummaryInfoGetProperty 함수는 요약 정보 스트림에서 단일 속성을 가져옵니다. (유니코드)
MsiSummaryInfoPersist MsiSummaryInfoPersist 함수는 변경된 요약 정보를 요약 정보 스트림에 다시 씁니다.
MsiSummaryInfoSetPropertyA MsiSummaryInfoSetProperty 함수는 단일 요약 정보 속성을 설정합니다. (ANSI)
MsiSummaryInfoSetPropertyW MsiSummaryInfoSetProperty 함수는 단일 요약 정보 속성을 설정합니다. (유니코드)
MsiVerifyDiskSpace MsiVerifyDiskSpace 함수는 현재 설치에 충분한 디스크 공간이 있는지 확인합니다.
MsiViewClose MsiViewClose 함수는 실행된 뷰에 대한 결과 집합을 해제합니다.
MsiViewExecute MsiViewExecute 함수는 SQL 뷰 쿼리를 실행하고 필요한 매개 변수를 제공합니다.
MsiViewFetch MsiViewFetch 함수는 뷰에서 다음 순차 레코드를 가져옵니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다.
MsiViewGetColumnInfo MsiViewGetColumnInfo 함수는 열 이름 또는 정의가 포함된 레코드를 반환합니다. 이 함수는 MsiCloseHandle을 사용하여 닫아야 하는 핸들을 반환합니다.

MsiViewGetErrorA MsiViewGetError 함수는 MsiViewModify 함수에서 발생한 오류를 반환합니다. (ANSI)
MsiViewGetErrorW MsiViewGetError 함수는 MsiViewModify 함수에서 발생한 오류를 반환합니다. (유니코드)
MsiViewModify MsiViewModify 함수는 가져온 레코드를 업데이트합니다.

msopc.h 헤더

아티클2023. 08. 25.

이 헤더는 패키징에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 포장

msopc.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IOpcCertificateEnumerator CERT_CONTEXT 구조체에 대한 포인터의 읽기 전용 열거자입니다.
IOpcCertificateSet 서명과 함께 사용할 순서가 지정되지 않은 인증서 집합입니다.
IOpcDigitalSignature 패키지 디지털 서명을 나타냅니다.
IOpcDigitalSignatureEnumerator IOpcDigitalSignature 인터페이스 포인터의 읽기 전용 열거자입니다.
IOpcDigitalSignatureManager 패키징 API 개체로 표시되는 패키지에 대한 패키징 디지털 서명 인터페이스에 대한 액세스를 제공합니다.
IOpcFactory 패키징 API 개체를 만들고 패키지 저장 및 로드를 지원합니다.
IOpcPackage 패키지를 나타내고 패키지의 파트 및 관계에 액세스하는 메서드를 제공합니다.
IOpcPart 데이터가 들어 있고 관계 부분이 아닌 파트를 나타냅니다.
IOpcPartEnumerator

IOpcPart 인터페이스 포인터의 읽기 전용 열거자입니다.
IOpcPartSet 관계 부분이 아닌 패키지의 파트를 나타내는 파트 개체에 대한 순서가 지정되지 않은 IOpcPart 인터페이스 포인터 집합입니다.
IOpcPartUri 파트의 부품 이름을 나타냅니다.
IOpcRelationship 요소 또는 패키지인 원본과 대상 간의 링크인 관계를 나타냅니다.
IOpcRelationshipEnumerator IOpcRelationship 인터페이스 포인터의 읽기 전용 열거자입니다.
IOpcRelationshipSelector 관계 부분에서 서명을 위해 참조할 관계를 선택하는 방법을 나타냅니다.
IOpcRelationshipSelectorEnumerator IOpcRelationshipSelector 인터페이스 포인터의 읽기 전용 열거자입니다.
IOpcRelationshipSelectorSet 서명에 대한 관계를 식별하는 데 사용되는 선택 조건을 나타내는 순서가 지정되지 않은 IOpcRelationshipSelector 인터페이스 포인터 집합입니다.
IOpcRelationshipSet 관계 부분을 관계 개체에 대한 IOpcRelationship 인터페이스 포인터의 순서가 지정되지 않은 집합으로 나타냅니다.
IOpcSignatureCustomObject 서명되었거나 서명될 애플리케이션별 Object 요소를 나타냅니다.
IOpcSignatureCustomObjectEnumerator IOpcSignatureCustomObject 인터페이스 포인터의 읽기 전용 열거자입니다.
IOpcSignatureCustomObjectSet 애플리케이션별 Object 요소의 XML 태그를 포함하는 순서가 지정되지 않은 IOpcSignatureCustomObject 인터페이스 포인터 집합입니다.

IOpcSignaturePartReference 서명되었거나 서명될 파트에 대한 참조를 나타냅니다.
IOpcSignaturePartReferenceEnumerator IOpcSignaturePartReference 인터페이스 포인터의 읽기 전용 열거자입니다.
IOpcSignaturePartReferenceSet 서명할 파트에 대한 참조를 나타내는 순서가 지정되지 않은 IOpcSignaturePartReference 인터페이스 포인터 집합입니다.
IOpcSignatureReference 서명되었거나 서명될 XML 태그에 대한 참조를 나타냅니다.
IOpcSignatureReferenceEnumerator IOpcSignatureReference 인터페이스 포인터의 읽기 전용 열거자입니다.
IOpcSignatureReferenceSet 서명할 XML 요소에 대한 참조를 나타내는 순서가 지정되지 않은 IOpcSignatureReference 인터페이스 포인터 집합입니다.
IOpcSignatureRelationshipReference 서명되었거나 서명될 관계가 포함된 관계 부분에 대한 참조를 나타냅니다.
IOpcSignatureRelationshipReferenceEnumerator IOpcSignatureRelationshipReference 인터페이스 포인터의 읽기 전용 열거자입니다.
IOpcSignatureRelationshipReferenceSet 서명할 관계가 포함된 관계 파트에 대한 참조를 나타내는 순서가 지정되지 않은 IOpcSignatureRelationshipReference 인터페이스 포인터 집합입니다.
IOpcSigningOptions 서명을 생성하는 데 필요한 정보를 설정하고 액세스하는 메서드를 제공합니다.
IOpcUri 패키지 루트 또는 패키지 루트를 기준으로 하는 파트의 URI를 나타냅니다.

열거형

OPC_CANONICALIZATION_METHOD XML 태그에 적용할 정식화 메서드에 대해 설명합니다.
OPC_CERTIFICATE_EMBEDDING_OPTION 서명에 사용되는 인증서의 스토리지 위치를 설명합니다.
OPC_COMPRESSION_OPTIONS 파트 콘텐츠를 압축하는 방법을 설명합니다.
OPC_READ_FLAGS 패키지 구성 요소를 캐싱하고 ECMA-376 OpenXML, 1판, 2부:_Open OPC(패키징 규칙) 규칙 요구 사항에 대해 유효성을 검사하기 위한 읽기 설정을 설명합니다.
OPC_RELATIONSHIP_SELECTOR IOpcRelationshipSelector::GetSelectionCriterion 메서드의 selectionCriterion 매개 변수를 해석하는 방법을 설명합니다.
OPC_RELATIONSHIPS_SIGNING_OPTION IOpcSignatureRelationshipReference 인터페이스가 나타내는 참조가 관계 집합 개체에서 관계 개체로 표시되는 관계의 전체 또는 하위 집합을 참조하는지 여부를 설명합니다.
OPC_SIGNATURE_TIME_FORMAT 서명이 만들어진 시점의 레코드인 signingTime 매개 변수를 IOpcDigitalSignature::GetSigningTime 메서드를 해석하는 방법을 설명합니다.
OPC_SIGNATURE_VALIDATION_RESULT 서명의 상태 나타냅니다.
OPC_STREAM_IO_MODE 스트림의 읽기/쓰기 상태 설명합니다.
OPC_URI_TARGET_MODE 관계의 대상 모드를 나타냅니다.
OPC_WRITE_FLAGS

serialization 개체가 패키지를 생성하는 데 사용하는 인코딩 메서드에 대해 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

msp.h 헤더

아티클2023. 08. 25.

이 헤더는 전화 통신 애플리케이션 프로그래밍 인터페이스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 전화 통신 애플리케이션 프로그래밍 인터페이스

msp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ITMSPAddress ITMSPAddress(msp.h) 인터페이스는 MSP에 의해 구현되며 TAPI DLL에 대한 미디어 서비스 공급자를 나타냅니다.
ITPluggableTerminalEventSink ITPluggableTerminalEventSink(msp.h) 인터페이스는 플러그형 터미널의 변경 내용을 클라이언트 애플리케이션에 알리는 메시지를 발생시키는 메서드를 제공합니다.
ITPluggableTerminalEventSinkRegistration ITPluggableTerminalEventSinkRegistration(msp.h) 인터페이스는 플러그형 터미널 이벤트에 대한 클라이언트 애플리케이션을 등록 및 등록 취소합니다.

구조체

MSP_EVENT_INFO MSP_EVENT_INFO(msp.h) 구조체는 GetEvent 메서드에서 반환하는 이벤트 유형을 정의합니다.

열거형

MSP_ADDRESS_EVENT

MSP_EVENT ME_ADDRESS_EVENT 경우 MSP_ADDRESS_EVENT(msp.h) 열거형 상수는 GetEvent 메서드에 의해 MSP_EVENT_INFO 구조체 내에서 반환됩니다.
MSP_CALL_EVENT MSP_EVENT ME_CALL_EVENT 때 MSP_CALL_EVENT(msp.h) 열거형 상수는 GetEvent 메서드에 의해 MSP_EVENT_INFO 구조체 내에서 반환됩니다.
MSP_CALL_EVENT_CAUSE MSP_CALL_EVENT_CAUSE(msp.h) 열거형 상수는 MSP_EVENT ME_CALL_EVENT 때 GetEvent 메서드에 의해 MSP_EVENT_INFO 구조체 내에서 반환됩니다.
MSP_EVENT MSP_EVENT(msp.h) 열거자는 MSP_EVENT_INFO 포함된 이벤트 정보의 유형을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mspaddr.h 헤더

아티클2023. 08. 25.

이 헤더는 전화 통신 애플리케이션 프로그래밍 인터페이스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 전화 통신 애플리케이션 프로그래밍 인터페이스

mspaddr.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

CMSPAddress CMSPAddress 클래스는 MSP 주소 개체를 구현하고 ITMSPAddress 인터페이스를 지원합니다. 이 클래스는 MSPaddr.h에 정의되어 있습니다. 다음은 클래스를 문서화합니다.

Functions

CreateMSPCallHelper CreateMSPCallHelper 도우미 템플릿 함수는 CreateMSPCall의 파생 클래스 구현에서 호출됩니다.
ShutdownMSPCallHelper ShutdownMSPCallHelper 도우미 템플릿 함수는 파생 클래스의 ShutdownMSPCall 구현에서 호출됩니다.

구조체

MSPEVENTITEM MSPEVENTITEM 구조체에는 MSP 이벤트에 대한 링크 포인터 및 이벤트 유형이 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mspcall.h 헤더

아티클2023. 08. 25.

이 헤더는 전화 통신 애플리케이션 프로그래밍 인터페이스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 전화 통신 애플리케이션 프로그래밍 인터페이스

mspcall.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

CMSPCallBase CMSPCallBase 클래스는 호출 개체의 제네릭 구현을 제공합니다. ITStreamControl 인터페이스를 지원합니다. 스트림 목록을 유지하는 간단한 기본 클래스로 사용됩니다. 이 클래스는 MSPcall.h에 정의되어 있습니다.
CMSPCallMultiGraph CMSPCallBase에서 파생된 이 클래스는 각 스트림에 필터 그래프를 사용하는 호출을 정의합니다. 이러한 추가적인 특이성을 통해 이 클래스는 CMSPCallBase보다 더 많은 호출 기능을 구현할 수 있습니다. MSPcall.h에 정의됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

msports.h 헤더

아티클2023. 08. 25.

이 헤더는 직렬 컨트롤러 드라이버 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 직렬 컨트롤러 드라이버 참조

msports.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

ComDBClaimNextFreePort ComDBClaimNextFreePort는 아직 사용되지 않는 가장 낮은 COM 포트 번호를 반환합니다.
ComDBClaimPort ComDBClaimPort는 사용되지 않는 COM 포트 번호를 COM 포트 데이터베이스에서 "사용 중"으로 기록합니다.
ComDBClose ComDBClose는 COM 포트 데이터베이스에 대한 핸들을 닫습니다.
ComDBGetCurrentPortUsage ComDBGetCurrentPortUsage는 COM 포트 데이터베이스에서 현재 "사용 중"으로 기록된 COM 포트 번호에 대한 정보를 반환합니다.
ComDBOpen ComDBOpen은 COM 포트 데이터베이스에 대한 핸들을 반환합니다.
ComDBReleasePort ComDBReleasePort는 COM 포트 데이터베이스에서 COM 포트 번호를 해제합니다.
ComDBResizeDatabase ComDBResizeDatabase는 COM 포트 데이터베이스의 크기를 조정합니다.
SerialDisplayAdvancedSettings

SerialDisplayAdvancedSettings는 지정된 COM 포트 디바이스에 대해 시스템에서 제공하는 고급 설정 대화 상자를 표시합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mspstrm.h 헤더

아티클2023. 08. 25.

이 헤더는 전화 통신 애플리케이션 프로그래밍 인터페이스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 전화 통신 애플리케이션 프로그래밍 인터페이스

mspstrm.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

CMSPStream CMSPStream 클래스는 애플리케이션이 하위 스트림을 시작, 일시 중지 또는 중지하고 터미널을 선택하거나 선택 취소할 수 있는 메서드를 노출합니다. MSPstrm.h에 정의됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

msrdc.h 헤더

아티클2023. 08. 25.

이 헤더는 원격 차등 압축에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 차등 압축

msrdc.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IFindSimilarResults ISimilarity::FindSimilarFileId 메서드에서 반환된 파일 목록에서 정보를 검색하는 메서드를 제공합니다.
IRdcComparator 두 개의 서명 스트림(시드 및 원본)을 비교하고 대상 파일을 만드는 데 필요한 원본 및 시드 파일 데이터 청크 목록을 생성하는 데 사용됩니다.
IRdcFileReader 동기화되는 데이터가 디스크에 파일로 존재하지 않을 수 있으므로 IRdcFileReader 인터페이스는 파일 핸들과 동등한 값을 제공하는 데 사용됩니다.
IRdcFileWriter 파일에서 읽고 파일에 쓸 추상 인터페이스입니다.
IRdcGenerator 입력 데이터를 처리하고 생성기에서 사용하는 매개 변수를 읽는 데 사용됩니다.
IRdcGeneratorFilterMaxParameters FilterMax 생성기에서 사용하는 매개 변수를 설정하고 검색합니다.
IRdcGeneratorParameters 모든 유형의 생성기 매개 변수에 대한 제네릭 인터페이스입니다. 모든 생성기 매개 변수 개체는 이 인터페이스를 지원해야 합니다.
IRdcLibrary

RDC를 사용하기 위한 기본 인터페이스입니다.
IRdcSignatureReader 서명을 생성하는 데 사용되는 서명 및 매개 변수를 읽습니다.
IRdcSimilarityGenerator 서명 생성기가 유사성 데이터를 생성하고 유사성 데이터가 생성된 후 검색할 수 있도록 하는 메서드를 정의합니다.
ISimilarity 파일별 유사성 데이터 및 파일 ID를 유사성 파일에 저장하고 검색하는 메서드를 정의합니다.
ISimilarityFileIdTable 유사성 파일 ID 정보를 저장하고 검색하는 메서드를 정의합니다.
ISimilarityReportProgress RDC가 유사성 작업의 현재 완료 비율을 보고하는 메서드를 정의합니다.
ISimilarityTableDumpState ISimilarityTraitsTable::BeginDump 메서드에서 반환된 유사성 특성 목록에서 정보를 검색하는 메서드를 제공합니다.
ISimilarityTraitsMappedView RDC 애플리케이션이 유사성 특성 테이블 파일의 매핑된 보기를 조작하기 위해 구현할 수 있는 메서드를 제공합니다.
ISimilarityTraitsMapping RDC 애플리케이션이 유사성 특성 테이블 파일에 대한 파일 매핑 개체를 만들고 조작하기 위해 구현할 수 있는 메서드를 제공합니다.
ISimilarityTraitsTable 파일별 유사성 데이터를 저장하고 유사성 조회를 수행하는 메서드를 정의합니다.

구조체

FindSimilarFileIndexResults

ISimilarityTraitsTable::FindSimilarFileIndex 메서드가 일치하는 파일에 대해 반환한 파일 인덱스 정보를 포함합니다.
RdcBufferPointer 버퍼에 대해 설명합니다.
RdcNeed 두 데이터 집합을 동기화하는 데 필요한 청크에 대한 정보를 포함합니다.
RdcNeedPointer RdcNeed 구조체의 배열에 대해 설명합니다.
RdcSignature 단일 서명과 이를 생성하는 데 사용되는 청크의 길이를 포함합니다.
RdcSignaturePointer RdcSignature 구조체의 배열에 대해 설명합니다.
SimilarityData 파일의 유사성 데이터를 포함합니다.
SimilarityDumpData ISimilarityTableDumpState::GetNextData 메서드에 의해 파일에 대해 반환된 유사성 정보를 포함합니다.
SimilarityFileId 파일의 유사성 파일 ID를 포함합니다.
SimilarityMappedViewInfo 유사성 매핑 보기에 대한 정보를 포함합니다.

열거형

GeneratorParametersType 지원되는 생성기 매개 변수 집합을 정의합니다.

RDC_ErrorCode RDC별 오류 코드 집합을 정의합니다.
RdcCreatedTables 유사성 특성 테이블, 유사성 파일 ID 테이블 또는 둘 다의 상태를 설명하는 값을 정의합니다.
RdcMappingAccessMode RDC 파일 매핑 개체에 대한 액세스 모드 값을 정의합니다.
RdcNeedType 원격 복사본을 생성하는 데 사용되는 데이터 청크 집합을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mssip.h 헤더

아티클2023. 08. 25.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

mssip.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CryptSIPAddProvider CryptSIPAddProvider 함수는 SIP(주체 인터페이스 패키지)를 구현하는 지정된 DLL 파일에서 내보낸 함수를 등록합니다.
CryptSIPCreateIndirectData 제공된 SIP_SUBJECTINFO 구조체의 해시, 다이제스트 알고리즘 및 인코딩 특성이 포함된 SIP_INDIRECT_DATA 구조를 반환합니다. 해시는 데이터에 대한 간접 참조로 사용할 수 있습니다.
CryptSIPGetCaps SIP(주체 인터페이스 패키지)의 기능을 검색합니다.
CryptSIPGetSignedDataMsg 파일에서 Authenticode 서명을 검색합니다.
CryptSIPLoad SIP(주체 인터페이스 패키지)를 구현하는 DLL(동적 연결 라이브러리)을 로드하고 적절한 라이브러리 내보내기 함수를 SIP_DISPATCH_INFO 구조에 할당합니다.
CryptSIPPutSignedDataMsg 대상 파일에 Authenticode 서명을 저장합니다.
CryptSIPRemoveProvider CryptSIPAddProvider 함수에 대한 이전 호출에 의해 추가된 SIP(주체 인터페이스 패키지) DLL 파일의 레지스트리 세부 정보를 제거합니다.
CryptSIPRemoveSignedDataMsg

지정된 Authenticode 서명을 제거합니다.
CryptSIPRetrieveSubjectGuid 지정된 파일의 헤더 정보를 기반으로 GUID를 검색합니다.
CryptSIPRetrieveSubjectGuidForCatalogFile 지정된 파일과 연결된 주체 GUID를 검색합니다.
CryptSIPVerifyIndirectData 제공된 주체에 대해 간접 해시 데이터의 유효성을 검사합니다.

콜백 함수

pCryptSIPGetCaps 기능을 보고하기 위해 SIP(주체 인터페이스 패키지)에 의해 구현됩니다.
pfnIsFileSupported 레지스트리에 나열된 SIP(주체 인터페이스 패키지)를 쿼리하여 파일 형식을 처리하는 SIP를 확인합니다. (pfnIsFileSupported)
pfnIsFileSupportedName 레지스트리에 나열된 SIP(주체 인터페이스 패키지)를 쿼리하여 파일 형식을 처리하는 SIP를 확인합니다. (pfnIsFileSupportedName)

구조체

MS_ADDINFO_BLOB 메모리 내 BLOB 주체 유형에 대한 추가 정보를 제공합니다.
MS_ADDINFO_CATALOGMEMBER 카탈로그 멤버 주체 유형에 대한 추가 정보를 제공합니다.

MS_ADDINFO_FLAT 플랫 또는 엔드 투 엔드 제목 형식에 대한 추가 정보를 제공합니다.
SIP_ADD_NEWPROVIDER SIP(주체 인터페이스 패키지)를 정의합니다. 이 구조체는 CryptSIPAddProvider 함수에서 사용됩니다.
SIP_CAP_SET_V2 SIP_CAP_SET_V2 구조는 SIP(주체 인터페이스 패키지)의 기능을 정의합니다. (SIP_CAP_SET_V2 구조)
SIP_CAP_SET_V3 SIP_CAP_SET_V3 구조는 SIP(주체 인터페이스 패키지)의 기능을 정의합니다. (SIP_CAP_SET_V3 구조)
SIP_DISPATCH_INFO 애플리케이션이 SIP(주체 인터페이스 패키지) 작업을 수행하는 데 사용하는 CryptSIPLoad 함수에 의해 할당된 함수 포인터 집합을 포함합니다.
SIP_INDIRECT_DATA 해시된 주체 정보의 다이제스트를 포함합니다.
SIP_SUBJECTINFO 주체 정보 데이터를 SIP(주체 인터페이스 패키지) API에 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mstask.h 헤더

아티클2023. 08. 25.

이 헤더는 작업 스케줄러에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 작업 Scheduler

mstask.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumWorkItems 예약된 작업 폴더에서 작업을 열거하는 메서드를 제공합니다.
IProvideTaskPage 작업의 속성 시트 설정에 액세스하는 메서드를 제공합니다.
IScheduledWorkItem 특정 작업 항목을 관리하는 방법을 제공합니다.
Itask 작업을 실행하고, 작업 정보를 가져오거나 설정하고, 작업을 종료하는 메서드를 제공합니다. IScheduledWorkItem 인터페이스에서 파생되며 해당 인터페이스의 모든 메서드를 상속합니다.
ITaskScheduler 작업을 예약하는 메서드를 제공합니다.
ITaskTrigger 작업에 대한 트리거에 액세스하고 설정하는 메서드를 제공합니다. 트리거는 작업 시작 시간, 반복 조건 및 태스크 실행 시기를 제어하는 기타 매개 변수를 지정합니다.

구조체

매일

작업이 실행되는 간격(일)을 정의합니다.
MONTHLYDATE 태스크가 실행되는 월의 날짜를 정의합니다.
MONTHLYDOW 작업이 월, 주 및 요일별로 실행되는 날짜를 정의합니다.
TASK_TRIGGER 예약된 작업 항목을 실행하는 시간을 정의합니다.
TRIGGER_TYPE_UNION TASK_TRIGGER 구조체의 Type 멤버 내에서 트리거의 호출 일정을 정의합니다.
주간 작업 호출 사이의 간격(주)을 정의합니다.

열거형

TASK_TRIGGER_TYPE 작업과 연결된 트리거 유형을 정의합니다.
작업 페이지 검색할 작업 페이지의 유형을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mstcpip.h 헤더

아티클2023. 11. 22.

이 헤더는 Windows 소켓 2에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 소켓 2

mstcpip.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

주장
IN4_ADDR_EQUAL
IN4_CLASSA
IN4_CLASSB
IN4_CLASSC
IN4_CLASSD
IN4_IS_ADDR_6TO4ELIGIBLE
IN4_IS_ADDR_BROADCAST
IN4_IS_ADDR_LINKLOCAL
IN4_IS_ADDR_LOOPBACK
IN4_IS_ADDR_MC_ADMINLOCAL
IN4_IS_ADDR_MC_LINKLOCAL

IN4_IS_ADDR_MC_SITELOCAL
IN4_IS_ADDR_MULTICAST
IN4_IS_ADDR_RFC1918
IN4_IS_ADDR_SITELOCAL
IN4_IS_ADDR_UNSPECIFIED
IN4_IS_UNALIGNED_ADDR_6TO4ELIGIBLE
IN4_IS_UNALIGNED_ADDR_BROADCAST
IN4_IS_UNALIGNED_ADDR_LINKLOCAL
IN4_IS_UNALIGNED_ADDR_LOOPBACK
IN4_IS_UNALIGNED_ADDR_MULTICAST
IN4_IS_UNALIGNED_ADDR_RFC1918
IN4_IS_UNALIGNED_ADDR_UNSPECIFIED
IN4_MULTICAST
IN4_UNALIGNED_ADDR_EQUAL
IN4_UNCANONICALIZE_SCOPE_ID
IN4ADDR_ISANY
IN4ADDR_ISEQUAL

IN4ADDR_ISLOOPBACK
IN4ADDR_ISUNSPECIFIED
IN4ADDR_SCOPE_ID
IN4ADDR_SETANY
IN4ADDR_SETLOOPBACK
IN4ADDR_SETSOCKADDR
IN6_EXTRACT_V4ADDR_FROM_6TO4
IN6_EXTRACT_V4ADDR_FROM_ISATAP
IN6_GET_ADDR_V4COMPAT
IN6_GET_ADDR_V4MAPPED
IN6_IS_ADDR_6TO4
IN6_IS_ADDR_ALLNODESONLINK
IN6_IS_ADDR_ALLNODESONNODE
IN6_IS_ADDR_ALLROUTERSONLINK
IN6_IS_ADDR_ISATAP
IN6_IS_ADDR_SOLICITEDNODE
IN6_IS_ADDR_TEREDO

IN6_MULTICAST_SCOPE
IN6_PREFIX_EQUAL
IN6_SET_6TO4_PREFIX
IN6_SET_ADDR_SOLICITEDNODE
IN6_SET_ADDR_V4COMPAT
IN6_SET_ADDR_V4MAPPED
IN6_SET_ISATAP_IDENTIFIER
IN6_UNCANONICALIZE_SCOPE_ID
IN6ADDR_IS6TO4
IN6ADDR_ISISATAP
IN6ADDR_ISTEREDO
IN6ADDR_ISV4MAPPED
IN6ADDR_SETSOCKADDR
IN6ADDR_SETV4MAPPED

구조체

ASSOCIATE_NAMERES_CONTEXT_INPUT

정규화된 도메인 이름에 대한 전송 설정 ID 및 핸들을 포함합니다.
INET_PORT_RANGE SIO_ACQUIRE_PORT_RESERVATION IOCTL에서 TCP 또는 UDP 포트 블록에 대한 런타임 예약을 획득하는 데 사용하는 입력 데이터를 제공합니다.
INET_PORT_RESERVATION_INSTANCE TCP 또는 UDP 포트 블록에 대한 포트 예약 및 토큰을 포함합니다.
INET_PORT_RESERVATION_TOKEN TCP 또는 UDP 포트 블록에 대한 포트 예약 토큰을 포함합니다.
REAL_TIME_NOTIFICATION_SETTING_INPUT Windows 스토어 앱에서 백그라운드 네트워크 알림을 수신하기 위해 ControlChannelTrigger와 함께 사용되는 TCP 소켓에 대한 REAL_TIME_NOTIFICATION_CAPABILITY 전송 설정에 적용할 입력 설정을 제공합니다.
REAL_TIME_NOTIFICATION_SETTING_OUTPUT Windows 스토어 앱에서 백그라운드 네트워크 알림을 수신하기 위해 ControlChannelTrigger와 함께 사용되는 TCP 소켓에 대한 REAL_TIME_NOTIFICATION_CAPABILITY 전송 설정에 대한 쿼리의 출력 설정을 제공합니다.
SOCKET_PEER_TARGET_NAME 피어 대상의 IP 주소와 이름 및 소켓에서 사용할 보안 프로토콜 유형을 포함합니다.
SOCKET_SECURITY_QUERY_INFO WSAQuerySocketSecurity 함수에서 반환된 보안 정보를 포함합니다.
SOCKET_SECURITY_QUERY_TEMPLATE WSAQuerySocketSecurity 함수에서 사용하는 보안 템플릿을 포함합니다.
SOCKET_SECURITY_SETTINGS 소켓에 대한 일반 보안 요구 사항을 지정합니다.
SOCKET_SECURITY_SETTINGS_IPSEC IPsec과 관련된 다양한 보안 요구 사항 및 설정을 지정합니다.
TCP_INFO_v0

소켓에 대해 수집된 TCP(Transmission Control Protocol) 통계를 포함합니다.
TCP_INFO_v1 소켓에 대해 수집된 TCP(Transmission Control Protocol) 통계를 포함합니다. (버전 1.0)
TCP_INITIAL_RTO_PARAMETERS 소켓에서 사용할 초기 RTO(다시 전송 시간 제한) 매개 변수를 구성하기 위해 SIO_TCP_INITIAL_RTO IOCTL에서 사용하는 데이터를 지정합니다.
TIMESTAMPING_CONFIG SIO_TIMESTAMPING IOCTL에서 데이터그램 소켓에 대한 타임스탬프 수신을 구성하는 데 사용하는 입력 구조에 대해 설명합니다.
TRANSPORT_SETTING_ID TRANSPORT_SETTING_ID 구조체(mstcpip.h)는 특정 IOCTL이 소켓에 대한 전송 설정을 적용하거나 쿼리하는 데 사용하는 전송 설정 ID를 지정합니다.

열거형

CONTROL_CHANNEL_TRIGGER_STATUS Windows 스토어 앱에서 백그라운드 네트워크 알림을 수신하기 위해 ControlChannelTrigger와 함께 사용되는 TCP 소켓에 대한 REAL_TIME_NOTIFICATION_CAPABILITY 전송 설정에 대한 쿼리의 상태 지정합니다.
SOCKET_SECURITY_PROTOCOL 네트워크 트래픽을 보호하기 위해 소켓에서 사용할 보안 프로토콜 유형을 나타냅니다.
SOCKET_USAGE_TYPE 소켓의 사용 유형을 지정하는 데 사용됩니다.
TCPSTATE TCP(Transmission Control Protocol) 연결의 가능한 상태를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

mswmdm.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows Media 디바이스 관리자 11 SDK에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Media 디바이스 관리자 11 SDK

mswmdm.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IComponentAuthenticate IComponentAuthenticate 인터페이스는 Windows Media 디바이스 관리자 모듈 간에 안전하고 암호화된 통신을 제공합니다.
IMDServiceProvider IMDServiceProvider 인터페이스는 Windows Media 디바이스 관리자 서비스 공급자에 연결하는 데 사용하는 초기 인터페이스입니다.
IMDServiceProvider2 IMDServiceProvider2 인터페이스는 지정된 디바이스 경로 이름에 대한 IMDSPDevice 개체를 가져오는 방법을 제공하여 IMDServiceProvider 인터페이스를 확장합니다. 디바이스 경로 이름은 PnP(플러그 앤 플레이) 하위 시스템에 있습니다.
IMDServiceProvider3 IMDServiceProvider3 인터페이스는 디바이스 열거형 기본 설정을 위한 메서드를 제공하여 IMDServiceProvider2 인터페이스를 확장합니다.
IMDSPDevice IMDSPDevice 인터페이스는 미디어 디바이스와의 instance 기반 연결을 제공합니다.
IMDSPDevice2 IMDSPDevice2 인터페이스는 확장된 형식을 가져오고, PnP(플러그 앤 플레이) 디바이스 이름을 가져오고, 속성 페이지를 사용하도록 설정하고, 이름에서 스토리지 매체에 대한 포인터를 가져올 수 있도록 하여 IMDSPDevice를 확장합니다.
IMDSPDevice3

IMDSPDevice3 인터페이스는 Windows 미디어 플레이어 동기화해야 하는 디바이스에 대해 지원되어야 합니다.
IMDSPDeviceControl IMDSPDeviceControl 인터페이스는 디바이스를 제어하는 메서드를 제공합니다.
IMDSPDirectTransfer IMDSPDirectTransfer 인터페이스를 사용하면 Windows Media 디바이스 관리자 서비스 공급자에게 콘텐츠 전송을 위임할 수 있습니다.
IMDSPEnumDevice IMDSPEnumDevice 인터페이스는 미디어 디바이스를 열거하는 데 사용됩니다.
IMDSPEnumStorage IMDSPEnumStorage 인터페이스는 디바이스의 스토리지 미디어를 열거하는 데 사용됩니다.
IMDSPObject IMDSPObject 인터페이스는 스토리지 미디어와 데이터 전송을 관리합니다. Open, Read, Write 및 Close 메서드는 스토리지 개체가 파일인 경우에만 유효합니다.
IMDSPObject2 Windows Media 디바이스 관리자 IMDSPObject2를 사용하여 파일 읽기 및 쓰기를 보다 효율적으로 수행할 수 있습니다. 참고 서비스 공급자가 값이 1인 UseExtendedWmdm 디바이스 매개 변수를 추가하지 않으면 Windows Media 디바이스 관리자 이 인터페이스를 호출하지 않습니다. 이에 대한 자세한 내용은 디바이스 매개 변수를 참조하세요. .
IMDSPObjectInfo IMDSPObjectInfo 인터페이스는 IMDSPDeviceControl 인터페이스에서 스토리지 매체의 재생 가능한 개체를 참조하거나 액세스하는 방법을 설명하는 매개 변수를 가져오고 설정하는 메서드를 제공합니다.
IMDSPRevoked IMDSPRevoked 인터페이스는 업데이트된 구성 요소를 다운로드할 수 있는 URL을 검색합니다. 이 인터페이스를 구현하는 것은 선택 사항입니다. 자세한 내용은 필수 및 선택적 인터페이스를 참조하세요.
IMDSPStorage IMDSPStorage 인터페이스는 디바이스의 스토리지 매체와 인스턴스 기반 연결을 제공합니다.
IMDSPStorage2

IMDSPStorage2 인터페이스는 확장 특성을 가져오고 설정하는 메서드를 제공하고 이름에서 스토리지 매체에 대한 포인터를 가져올 수 있도록 하여 IMDSPStorage를 확장합니다.
IMDSPStorage3 IMDSPStorage3 인터페이스는 메타데이터를 지원하여 IMDSPStorage2를 확장합니다.
IMDSPStorage4 IMDSPStorage4 인터페이스는 가상 스토리지(예: 재생 목록 및 앨범) 및 메타데이터를 지원하기 위해 IMDSPStorage3을 확장합니다. 참고 서비스 공급자가 값이 1인 UseExtendedWmdm 디바이스 매개 변수를 추가하지 않으면 Windows Media 디바이스 관리자 이 인터페이스를 호출하지 않습니다. 이에 대한 자세한 내용은 디바이스 매개 변수를 참조하세요. .
IMDSPStorageGlobals IMDSPStorage 인터페이스에서 가져온 IMDSPStorageGlobals 인터페이스는 스토리지 매체에 대한 전역 정보를 검색하는 메서드를 제공합니다. 여기에는 사용 가능한 공간의 양, 매체의 일련 번호 등이 포함될 수 있습니다.
ISCPSecureAuthenticate ISCPSecureAuthenticate 인터페이스는 보안 콘텐츠 공급자의 기본 인터페이스로, Windows Media는 보안 콘텐츠 공급자를 인증하고 보안 콘텐츠 공급자에 의해 인증되도록 쿼리를 디바이스 관리자.
ISCPSecureAuthenticate2 ISCPSecureAuthenticate2 인터페이스는 세션 개체를 가져오는 방법을 제공하여 ISCPSecureAuthenticate를 확장합니다.
ISCPSecureExchange ISCPSecureExchange 인터페이스는 콘텐츠와 연결된 보안 콘텐츠 및 권한을 교환하는 데 사용됩니다. 보안 콘텐츠 공급자는 이 인터페이스를 구현하고 보안 Windows Media 디바이스 관리자 구현에서 해당 메서드를 호출합니다.
ISCPSecureExchange2 ISCPSecureExchange2 인터페이스는 새 버전의 TransferContainerData 메서드를 제공하여 ISCPSecureExchange를 확장합니다.
ISCPSecureExchange3 ISCPSecureExchange3 인터페이스는 향상된 데이터 교환 성능과 전송 완료 콜백 메서드를 제공하여 ISCPSecureExchange2를 확장합니다.
ISCPSecureQuery

ISCPSecureQuery 인터페이스는 Windows Media 디바이스 관리자 쿼리하여 보안 콘텐츠의 소유권을 확인합니다.
ISCPSecureQuery2 ISCPSecureQuery2 인터페이스는 보안 콘텐츠 공급자가 콘텐츠를 담당하는지 여부를 결정하는 기능을 통해 ISCPSecureQuery를 확장하고, 그렇다면 해지된 구성 요소를 업데이트하고 해지된 구성 요소를 결정하는 URL을 제공합니다.
ISCPSecureQuery3 ISCPSecureQuery3 인터페이스는 권한을 검색하고 명확한 채널에서 의사 결정을 내리기 위한 새로운 메서드 집합을 제공하여 ISCPSecureQuery2를 확장합니다.
ISCPSession ISCPSession 인터페이스는 여러 작업에 대한 효율적인 공통 상태 관리를 제공합니다. SCP(보안 콘텐츠 공급자) 세션은 여러 파일을 전송할 때 유용합니다.
IWMDeviceManager IWMDeviceManager 인터페이스는 애플리케이션에 대한 최상위 Windows Media 디바이스 관리자 인터페이스입니다.
IWMDeviceManager2 IWMDeviceManager2 인터페이스는 IWMDeviceManager 인터페이스를 확장합니다.
IWMDeviceManager3 IWMDeviceManager3 인터페이스는 디바이스 열거형 기본 설정을 설정하는 메서드를 제공하여 IWMDeviceManager2 인터페이스를 확장합니다.
IWMDMDevice IWMDMDevice 인터페이스는 단일 휴대용 디바이스를 검사하고 탐색하는 메서드를 제공합니다. 인터페이스를 사용하여 디바이스에 대한 정보를 얻고 해당 스토리지를 열거할 수 있습니다. IWMDMDevice2는 이 인터페이스의 기능을 확장합니다.
IWMDMDevice2 IWMDMDevice2 인터페이스는 디바이스에서 지원하는 비디오 형식을 얻고, 이름에서 스토리지를 찾고, 속성 페이지를 사용할 수 있도록 하여 IWMDMDevice를 확장합니다.
IWMDMDevice3 IWMDMDevice3 인터페이스는 디바이스에서 속성을 쿼리하고, 디바이스 I/O 제어 코드를 전송하는 메서드를 제공하고, 스토리지를 검색하고 디바이스 형식 기능을 검색하는 업그레이드된 메서드를 제공하여 IWMDMDevice2를 확장합니다.

IWMDMDeviceControl IWMDMDeviceControl 인터페이스는 디바이스에서 재생을 제어하는 메서드를 제공합니다.
IWMDMDeviceSession IWMDMDeviceSession 인터페이스는 여러 작업을 한 세션에 번들로 묶어 디바이스 작업의 효율성을 향상시킵니다.
IWMDMEnumDevice IWMDMEnumDevice 인터페이스는 컴퓨터에 연결된 휴대용 디바이스를 열거합니다. 이 인터페이스를 가져오려면 IWMDeviceManager::EnumDevices를 호출합니다.
IWMDMEnumStorage IWMDMEnumStorage 인터페이스는 디바이스의 스토리지를 열거합니다.
IWMDMMetaData IWMDMMetaData 인터페이스는 스토리지의 메타데이터 속성(예: 아티스트, 앨범, 장르 등)을 설정하고 검색합니다.
IWMDMNotification 선택적 애플리케이션 구현 IWMDMNotification 인터페이스를 사용하면 디바이스 또는 메모리 스토리지(예: RAM 카드)가 컴퓨터에서 연결되거나 연결이 끊어지면 애플리케이션 및 서비스 공급자가 알림을 받을 수 있습니다. 참고 이 메서드는 등록된 플러그 앤 플레이 디바이스에 대해서만 호출됩니다. 다른 디바이스 도착 또는 출발 시 이 인터페이스가 호출되지 않습니다. 이 인터페이스 GUID는 mssachlp.lib에 올바르게 정의되지 않았습니다. 따라서 다음을 수행해야 합니다.
IWMDMObjectInfo IWMDMObjectInfo 인터페이스는 IWMDMDeviceControl 인터페이스에서 디바이스에서 재생 가능한 파일을 처리하는 방법을 제어하는 정보를 가져오고 설정합니다. 이 인터페이스는 재생할 수 없는 파일을 위한 것이 아닙니다.
IWMDMOperation 이 선택적 애플리케이션 구현 IWMDMOperation 인터페이스를 사용하면 애플리케이션에서 파일을 전송하는 동안 컴퓨터에서 데이터를 읽거나 쓰는 방법을 제어할 수 있습니다.
IWMDMOperation2 선택적 애플리케이션 구현 IWMDMOperation2 인터페이스는 확장 특성을 가져와서 설정하는 메서드를 제공하여 IWMDMOperation을 확장합니다.
IWMDMOperation3

선택적 애플리케이션 구현 IWMDMOperation3 인터페이스는 효율성을 높이기 위해 암호화되지 않은 데이터를 전송하는 새로운 방법을 제공하여 IWMDMOperation을 확장합니다.
IWMDMProgress 애플리케이션에서 구현한 선택적 IWMDMProgress를 사용하면 애플리케이션이 미디어 또는 파일 전송 서식 지정과 같은 작업의 진행률을 추적할 수 있습니다.
IWMDMProgress2 선택적 애플리케이션 구현 IWMDMProgress2 인터페이스는 상태 표시기를 제공하여 IWMDMProgress::End 메서드를 확장합니다.
IWMDMProgress3 선택적 애플리케이션 구현 IWMDMProgress3 인터페이스는 모니터링되는 이벤트를 지정하고 컨텍스트별 정보를 허용하는 추가 입력 매개 변수를 제공하여 IWMDMProgress2를 확장합니다. 이 콜백 인터페이스를 구현하는 애플리케이션은 새 메서드 외에도 이전 버전과의 호환성을 위해 IWMDMProgress 및 IWMDMProgress2에 해당하는 메서드에 대한 구현을 제공해야 합니다.
IWMDMRevoked IWMDMRevoked 인터페이스는 해지 오류로 인해 전송이 실패하는 경우 업데이트된 구성 요소를 다운로드할 수 있는 URL을 검색합니다.
IWMDMStorage IWMDMStorage 인터페이스의 instance 디바이스에서 스토리지(파일, 폴더 또는 재생 목록과 같은 데이터 또는 컬렉션 개체의 일반 이름)를 검사하고 탐색하는 메서드를 제공합니다.
IWMDMStorage2 IWMDMStorage2 인터페이스는 이름으로 자식 스토리지를 가져와 확장 특성을 가져와서 설정할 수 있도록 하여 IWMDMStorage를 확장합니다. IWMDMStorage3 인터페이스는 메타데이터를 지원하여 이 인터페이스를 확장합니다.
IWMDMStorage3 IWMDMStorage3 인터페이스는 메타데이터를 노출하여 IWMDMStorage2를 확장합니다.
IWMDMStorage4 IWMDMStorage4 인터페이스는 스토리지에 사용 가능한 메타데이터의 하위 집합을 검색하고 다른 스토리지에 대한 참조 목록을 설정 및 검색하는 메서드를 제공하여 IWMDMStorage3을 확장합니다.
IWMDMStorageControl

IWMDMStorageControl 인터페이스는 스토리지, 디바이스 또는 디바이스와 컴퓨터 간에 파일을 삽입, 삭제 또는 이동하는 데 사용됩니다.
IWMDMStorageControl2 IWMDMStorageControl2 인터페이스는 스토리지에 콘텐츠를 삽입할 때 대상 파일의 이름을 설정할 수 있도록 하여 IWMDMStorageControl을 확장합니다.
IWMDMStorageControl3 IWMDMStorageControl3 인터페이스는 IWMDMMetaData 인터페이스 포인터를 허용하는 Insert 메서드를 제공하여 IWMDMStorageControl2를 확장합니다.
IWMDMStorageGlobals IWMDMStorageGlobals 인터페이스는 디바이스의 스토리지 매체(예: 플래시 ROM 카드)에 대한 전역 정보를 검색하는 방법을 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mswsock.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows 소켓 2에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 소켓 2

mswsock.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

Acceptex AcceptEx 함수(mswsock.h)는 새 연결을 수락하고 로컬 및 원격 주소를 반환하며 클라이언트 애플리케이션에서 보낸 첫 번째 데이터 블록을 받습니다.
GetAcceptExSockaddrs GetAcceptExSockaddrs 함수(mswsock.h)는 AcceptEx 함수 호출에서 가져온 데이터를 구문 분석하고 로컬 및 원격 주소를 sockaddr 구조에 전달합니다.
TransmitFile TransmitFile 함수(mswsock.h)는 연결된 소켓 핸들을 통해 파일 데이터를 전송합니다.
WSARecvEx WSARecvEx 함수(mswsock.h)는 연결된 소켓 또는 바인딩된 연결 없는 소켓에서 데이터를 받습니다.

콜백 함수

LPFN_CONNECTEX ConnectEx 함수는 지정된 소켓에 대한 연결을 설정하고 연결이 설정되면 필요에 따라 데이터를 보냅니다.
LPFN_DISCONNECTEX 소켓의 연결을 닫고 소켓 핸들을 다시 사용할 수 있습니다. 참고: 이 함수는 Windows 소켓 사양에 대한 Microsoft 전용 확장입니다.

LPFN_RIOCLOSECOMPLETIONQUEUE Winsock 등록 I/O 확장을 사용하여 요청을 보내고 받아 I/O 완료 알림에 사용되는 기존 완료 큐를 닫습니다.
LPFN_RIOCREATECOMPLETIONQUEUE Winsock 등록 I/O 확장에 사용할 특정 크기의 I/O 완성 큐를 만듭니다.
LPFN_RIOCREATEREQUESTQUEUE Winsock 등록 I/O 확장에 사용할 지정된 소켓 및 I/O 완성 큐를 사용하여 등록된 I/O 소켓 설명자를 만듭니다.
LPFN_RIODEQUEUECOMPLETION Winsock 등록 I/O 확장에 사용할 I/O 완료 큐에서 항목을 제거합니다.
LPFN_RIODEREGISTERBUFFER Winsock 등록 I/O 확장에 사용되는 등록된 버퍼를 등록 취소합니다.
LPFN_RIONOTIFY Winsock 등록 I/O 확장에 사용할 I/O 완료 큐에 알림 동작에 사용할 메서드를 등록합니다.
LPFN_RIORECEIVE Winsock 등록 I/O 확장에 사용하기 위해 연결된 등록된 I/O TCP 소켓 또는 바인딩된 등록된 I/O UDP 소켓에서 네트워크 데이터를 받습니다.
LPFN_RIORECEIVEEX 연결된 등록된 I/O TCP 소켓 또는 바인딩된 등록된 I/O UDP 소켓에서 Winsock 등록 I/O 확장에 사용할 추가 옵션을 사용하여 네트워크 데이터를 받습니다.
LPFN_RIOREGISTERBUFFER Winsock 등록 I/O 확장에 사용할 지정된 버퍼를 사용하여 등록된 버퍼 설명자인 RIO_BUFFERID 등록합니다.
LPFN_RIORESIZECOMPLETIONQUEUE Winsock 등록 I/O 확장에 사용하기 위해 I/O 완료 큐의 크기를 더 크거나 작게 조정합니다.
LPFN_RIORESIZEREQUESTQUEUE Winsock 등록 I/O 확장에 사용하기 위해 요청 큐의 크기를 더 크거나 작게 조정합니다.

LPFN_RIOSEND Winsock 등록 I/O 확장에 사용하기 위해 연결된 등록된 I/O TCP 소켓 또는 바인딩된 등록된 I/O UDP 소켓에 네트워크 데이터를 보냅니다.
LPFN_RIOSENDEX 연결된 등록된 I/O TCP 소켓 또는 바인딩된 등록된 I/O UDP 소켓에서 Winsock 등록 I/O 확장에 사용할 추가 옵션을 사용하여 네트워크 데이터를 보냅니다.
LPFN_TRANSMITPACKETS 연결된 소켓을 통해 메모리 내 데이터 또는 파일 데이터를 전송합니다.
LPFN_WSARECVMSG *_LPFN_WSARECVMSG_*는 함수 포인터 형식입니다. 앱에서 일치하는 **WSARecvMsg** 콜백 함수를 구현합니다. 시스템은 콜백 함수를 사용하여 연결된 소켓을 통해 메모리 내 데이터 또는 파일 데이터로 전송합니다.

구조체

RIO_EXTENSION_FUNCTION_TABLE Winsock 등록 I/O 확장을 구현하는 함수에 대한 정보를 포함합니다.
RIO_NOTIFICATION_COMPLETION Winsock 등록 I/O 확장을 사용하여 네트워크 데이터를 보내거나 받기 위해 RIONotify 함수와 함께 사용할 I/O 완성 방법을 지정합니다.
TRANSMIT_FILE_BUFFERS TRANSMIT_FILE_BUFFERS 구조체(mswsock.h)는 TransmitFile 함수 파일 전송 작업 중 파일 데이터 전후에 전송할 데이터를 지정합니다.
TRANSMIT_PACKETS_ELEMENT TransmitPackets 함수에서 전송할 단일 데이터 요소를 지정합니다.

열거형

RIO_NOTIFICATION_COMPLETION_TYPE Winsock 등록 I/O 확장을 사용하여 데이터를 보내거나 받을 때 RIONotify 함수와 함께 사용할 완료 큐 알림 유형을 지정합니다.

mswsockdef.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows 소켓 2에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 소켓 2

mswsockdef.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

RIO_BUF Winsock 등록 I/O 확장을 사용하여 네트워크 데이터를 보내거나 받는 데 사용되는 등록된 버퍼의 일부를 지정합니다.
RIORESULT Winsock 등록 I/O 확장에 사용되는 요청 완료 결과를 나타내는 데 사용되는 데이터를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

msxml.h 헤더

아티클2023. 03. 08.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

msxml.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IXMLElementCollection 인덱싱된 액세스에 대한 XML 요소 컬렉션을 지원합니다.

msxml6.h 헤더

아티클2023. 08. 25.

이 헤더는 XML HTTP 확장 요청에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- XML HTTP 확장 요청

msxml6.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IXMLHTTPRequest2 HTTP 요청을 구성 및 보내고 콜백을 사용하여 HTTP 응답 처리 중에 알림을 수신하는 데 필요한 메서드 및 속성을 제공합니다. 참고 이 인터페이스는 Windows Phone 8.1에서 지원됩니다. .
IXMLHTTPRequest2Callback HTTP 요청 및 응답 처리에 영향을 주는 이벤트의 미해결 IXMLHTTPRequest2 요청을 사용하여 애플리케이션에 알리는 콜백을 정의합니다. 참고 이 인터페이스는 Windows Phone 8.1에서 지원됩니다. .
IXMLHTTPRequest3 HTTP 요청을 구성 및 보내고 콜백을 사용하여 HTTP 응답 처리 중에 알림을 수신하는 데 필요한 메서드 및 속성을 제공합니다.
IXMLHTTPRequest3Callback HTTP 요청 및 응답 처리에 영향을 주는 이벤트의 미해결 IXMLHTTPRequest3 요청을 사용하여 애플리케이션에 알리는 콜백을 정의합니다.

구조체

XHR_CERT 인코딩된 인증서를 가리키는 버퍼를 정의합니다.
XHR_COOKIE

SetCookie 메서드를 호출하여 HTTP 쿠키 jar에 추가하거나 GetCookie 메서드를 호출하여 HTTP 쿠키 jar에서 검색할 수 있는 쿠키를 정의합니다.

열거형

XHR_AUTH 프록시에 연결하거나 HTTP 서버에 연결하는 데 인증을 사용할지 여부를 지정합니다.
XHR_CERT_ERROR_FLAG IXMLHTTPRequest3Callback 인터페이스에서 OnServerCertificateReceived 메서드를 처리하여 서버와 SSL 협상 중에 서버 인증서 오류를 나타내는 플래그를 정의합니다.
XHR_CERT_IGNORE_FLAG IXMLHTTPRequest3 인터페이스에서 SetProperty 메서드를 호출하여 특정 인증서 오류를 무시하도록 나가는 HTTP 요청에 할당할 수 있는 플래그를 정의합니다.
XHR_COOKIE_FLAG GetCookie 메서드를 호출하여 SETCookie 메서드 또는 HTTP 쿠키 jar에서 쿼리를 호출하여 HTTP 쿠키 jar의 쿠키에 할당할 수 있는 플래그 집합을 정의합니다.
XHR_COOKIE_STATE 쿠키의 상태를 지정합니다.
XHR_CRED_PROMPT 인증을 위해 사용자에게 자격 증명 프롬프트를 허용할지 여부를 지정합니다.
XHR_PROPERTY SetProperty 메서드를 호출하여 나가는 HTTP 요청에 할당할 수 있는 속성을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mtpext.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows Media 디바이스 관리자 11 SDK에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Media 디바이스 관리자 11 SDK

mtpext.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

MTP_COMMAND_DATA_IN MTP_COMMAND_DATA_IN 구조에는 IWMDMDevice3::D eviceIoControl 메서드를 사용하여 디바이스로 전송되는 MTP(미디어 전송 프로토콜) 사용자 지정 명령이 포함되어 있습니다.
MTP_COMMAND_DATA_OUT MTP_COMMAND_DATA_OUT 구조에는 IWMDMDevice3::D eviceIoControl에 대한 호출을 종료할 때 디바이스 드라이버가 채우는 MTP(미디어 전송 프로토콜) 응답이 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

mtxdm.h 헤더

아티클2023. 08. 25.

이 헤더는 COM+(Component Services)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM+(구성 요소 서비스)

mtxdm.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

GetDispenserManager 디스펜서 관리자의 IDispenserManager 인터페이스를 검색합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

muiload.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows 애플리케이션에 대한 Internationalization에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 애플리케이션에 대한 국제화

muiload.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

FreeMUILibrary LoadMUILibrary에서 로드한 리소스 모듈에 대한 핸들을 해제합니다.
GetUILanguageFallbackList 언어 이름으로 표시되는 UI 언어의 대체 목록을 가져옵니다.
LoadMUILibraryA 특정 LN(언어 중립) 파일과 연결된 언어별 리소스에 대한 핸들을 반환합니다. (ANSI)
LoadMUILibraryW 특정 LN(언어 중립) 파일과 연결된 언어별 리소스에 대한 핸들을 반환합니다. (유니코드)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

namedpipeapi.h 헤더

아티클2023. 08. 25.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID
- 시스템 서비스

namedpipeapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CallNamedPipeW 메시지 형식 파이프에 연결하고 파이프의 instance 사용할 수 없는 경우 대기하고 파이프에 쓰고 파이프에서 읽은 다음 파이프를 닫습니다. (CallNamedPipeW)
ConnectNamedPipe 명명된 파이프 서버 프로세스가 클라이언트 프로세스가 명명된 파이프의 instance 연결되기를 기다릴 수 있도록 합니다.
CreateNamedPipeW CreateNamedPipeW(유니코드) 함수(winbase.h)는 명명된 파이프의 instance 만들고 후속 파이프 작업에 대한 핸들을 반환합니다.
CreatePipe 익명 파이프를 만들고 파이프의 읽기 및 쓰기 끝에 대한 핸들을 반환합니다.
DisconnectNamedPipe 클라이언트 프로세스에서 명명된 파이프 instance 서버 끝의 연결을 끊습니다.
GetNamedPipeClientComputerNameW GetNamedPipeClientComputerNameW(유니코드) 함수(winbase.h)는 지정된 명명된 파이프에 대한 클라이언트 컴퓨터 이름을 검색합니다.
GetNamedPipeHandleStateW GetNamedPipeHandleStateW(유니코드) 함수(winbase.h)는 지정된 명명된 파이프에 대한 정보를 검색합니다.

GetNamedPipeInfo 지정된 명명된 파이프에 대한 정보를 검색합니다.
ImpersonateNamedPipeClient 명명된 파이프 클라이언트 애플리케이션을 가장합니다.
PeekNamedPipe 명명된 파이프 또는 익명 파이프의 데이터를 파이프에서 제거하지 않고 버퍼에 복사합니다.
SetNamedPipeHandleState 지정된 명명된 파이프의 읽기 모드 및 차단 모드를 설정합니다. 지정된 핸들이 명명된 파이프의 클라이언트 끝에 있고 명명된 파이프 서버 프로세스가 원격 컴퓨터에 있는 경우 함수를 사용하여 로컬 버퍼링을 제어할 수도 있습니다.
TransactNamedPipe 메시지를 쓰고 지정된 명명된 파이프에서 메시지를 읽는 함수를 단일 작업으로 결합합니다.
WaitNamedPipeW WaitNamedPipeW(유니코드) 함수(winbase.h)는 시간 제한 간격이 경과하거나 지정된 명명된 파이프의 instance 연결에 사용할 수 있게 될 때까지 대기합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

namespaceapi.h 헤더

아티클2023. 08. 25.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

namespaceapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AddSIDToBoundaryDescriptor 지정된 경계 설명자에 SID(보안 식별자)를 추가합니다.
ClosePrivateNamespace 열려 있는 네임스페이스 핸들을 닫습니다.
CreateBoundaryDescriptorW CreateBoundaryDescriptorW(유니코드) 함수(namespaceapi.h)는 경계 설명자를 만듭니다.
CreatePrivateNamespaceW CreatePrivateNamespaceW(유니코드) 함수(namespaceapi.h)는 프라이빗 네임스페이스를 만듭니다.
DeleteBoundaryDescriptor 지정된 경계 설명자를 삭제합니다.
OpenPrivateNamespaceW OpenPrivateNamespaceW(유니코드) 함수(namespaceapi.h)는 프라이빗 네임스페이스를 엽니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

naptypes.h 헤더

아티클2023. 03. 08.

이 헤더는 네트워크 액세스 보호에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 액세스 보호

naptypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

CorrelationId SoHResponses와 SoHRequests를 페어링하는 데 사용되며 SoH 교환을 고유하게 설명합니다.
CountedString 정의된 길이로 null로 끝나는 문자열을 정의합니다.
FailureCategoryMapping SHV(시스템 상태 유효성 검사기)에서 반환되는 선택적 준수 상태 정보를 포함합니다.
FixupInfo Sysytem Health Agent(SHA)에 대한 수정 정보를 포함합니다.
Ipv4Address IPv4 주소를 정의합니다.
Ipv6Address IPv6 주소를 정의합니다.
IsolationInfo 컴퓨터 또는 연결의 격리 상태를 정의합니다.
IsolationInfoEx 컴퓨터 또는 연결의 확장된 격리 상태를 정의합니다.

NapComponentRegistrationInfo SHA, SHV 또는 적용 클라이언트와 같은 등록된 NAP 구성 요소를 정의합니다.
NetworkSoH 유선 SoH 프로토콜을 정의합니다.
PrivateData 불투명 데이터 Blob을 저장하고 검색하는 데 사용됩니다.
ResultCodes 결과 코드 목록을 정의합니다.
Soh SoH(Health 문) 데이터를 포함합니다.
SoHAttribute SHA/SHV와 NAP 시스템 간의 SoH 프로토콜을 정의합니다.
SystemHealthAgentState SHA의 동적 상태를 저장합니다.

열거형

ExtendedIsolationState 연결의 확장된 격리 상태를 설명합니다.
FailureCategory 오류의 원인을 나타냅니다.
FixupState SHA(시스템 상태 에이전트)의 수정 상태를 정의합니다.
IsolationState 연결의 격리 상태를 설명합니다.

NapNotifyType NapAgent 서비스에서 보낸 서비스 알림 유형을 열거합니다.
RemoteConfigurationType 구성 요소에 대해 가능한 원격 구성의 유형을 설명합니다.

natupnp.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows 방화벽 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 방화벽 기술

natupnp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

INATEventManager INATEventManager 인터페이스는 UPnP 기술을 사용하는 NAT 애플리케이션에 대한 메서드를 제공하여 NAT에 콜백 인터페이스를 등록합니다. NAT 구성이 변경되면 시스템은 이러한 인터페이스에서 메서드를 호출합니다.
INATExternalIPAddressCallback INATExternalIPAddressCallback 인터페이스는 UPnP 기술을 사용하여 NAT 애플리케이션에 의해 구현됩니다. NAT 컴퓨터의 외부 IP 주소가 변경되면 시스템에서 호출하는 메서드를 제공합니다.
INATNumberOfEntriesCallback INATNumberOfEntriesCallback 인터페이스는 포트 매핑 수가 변경되면 시스템에서 호출하는 메서드를 제공합니다.
IStaticPortMapping IStaticPortMapping 인터페이스는 특정 포트 매핑에 대한 정보를 검색하고 변경하는 메서드를 제공합니다.
IStaticPortMappingCollection IStaticPortMappingCollection 인터페이스는 정적 포트 매핑 컬렉션을 관리하는 메서드를 제공합니다.
IUPnPNAT IUPnPNAT 인터페이스는 UPnP를 사용하여 NAT(네트워크 주소 변환)를 관리하기 위한 기본 인터페이스입니다. IUPnPNAT 인터페이스는 UPnP 기술을 사용하여 NAT API의 다른 모든 인터페이스에 직접 또는 간접적으로 액세스를 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

nb30.h 헤더

아티클2023. 08. 25.

이 헤더는 NetBIOS에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Netbios

nb30.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

Netbios Netbios 함수는 지정된 NCB(네트워크 제어 블록)를 해석하고 실행합니다.

구조체

ACTION_HEADER ACTION_HEADER 구조에는 작업에 대한 정보가 포함됩니다. 이 작업은 표준 전송 인터페이스의 확장입니다.
ADAPTER_STATUS ADAPTER_STATUS 구조에는 네트워크 어댑터에 대한 정보가 포함됩니다.
FIND_NAME_BUFFER FIND_NAME_BUFFER 구조에는 로컬 네트워크 세션에 대한 정보가 포함됩니다.
FIND_NAME_HEADER FIND_NAME_HEADER 구조에는 네트워크 이름에 대한 정보가 포함됩니다. 이 구조체 뒤에는 이름을 설명하는 데 필요한 만큼의 FIND_NAME_BUFFER 구조가 잇습니다.
LANA_ENUM LANA_ENUM 구조체에는 현재 LAN 어댑터의 숫자가 포함됩니다.
NAME_BUFFER

NAME_BUFFER 구조에는 로컬 네트워크 이름에 대한 정보가 포함됩니다. 하나 이상의 NAME_BUFFER 구조체는 애플리케이션이 NCB 구조체의 ncb_command 멤버에서 NCBASTAT 명령을 지정하는 경우 ADAPTER_STATUS 구조체를 따릅니다.
Ncb NCB 구조체는 네트워크 제어 블록을 나타냅니다.
SESSION_BUFFER SESSION_BUFFER 구조에는 로컬 네트워크 세션에 대한 정보가 포함됩니다. 하나 이상의 SESSION_BUFFER 구조체는 애플리케이션이 NCB 구조체의 ncb_command 멤버에서 NCBSSTAT 명령을 지정하는 경우 SESSION_HEADER 구조체를 따릅니다.
SESSION_HEADER SESSION_HEADER 구조에는 네트워크 세션에 대한 정보가 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ncrypt.h 헤더

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

> **별칭**
>
> - NCryptBuffer (BCryptBuffer의 별칭)
> - NCryptBufferDesc (BCryptBufferDesc의 별칭)

ncrypt.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

NCryptCreateClaim 키 증명 클레임을 만듭니다.
NCryptCreatePersistedKey 새 키를 만들고 지정된 키 스토리지 공급자에 저장합니다.
NCryptDecrypt 암호화된 데이터 블록을 해독합니다.
NCryptDeleteKey 스토리지에서 CNG 키를 삭제합니다.
NCryptDeriveKey 비밀 핸들에서 키를 파생합니다. (NCryptDeriveKey)
NCryptEncrypt 데이터 블록을 암호화합니다. (NCryptEncrypt)
NCryptEnumAlgorithms 지정된 키 스토리지 공급자가 지원하는 알고리즘의 이름을 가져옵니다.

NCryptEnumKeys 공급자가 저장한 키의 이름을 가져옵니다.
NCryptEnumStorageProviders 등록된 키 스토리지 공급자의 이름을 가져옵니다.
NCryptExportKey CNG 키를 메모리 BLOB으로 내보냅니다.
NCryptFinalizeKey CNG 키 스토리지 키를 완료합니다.
NCryptFreeBuffer CNG 키 스토리지 공급자가 할당한 메모리 블록을 해제합니다.
NCryptFreeObject CNG 키 스토리지 개체를 해제합니다.
NCryptGetProperty 키 스토리지 개체의 명명된 속성 값을 검색합니다.
NCryptImportKey 메모리 BLOB에서 암호화 API - CNG(차세대) 키를 가져옵니다.
NCryptIsAlgSupported CNG 키 스토리지 공급자가 특정 암호화 알고리즘을 지원하는지 여부를 결정합니다.
NCryptIsKeyHandle 지정된 핸들이 CNG 키 핸들인지 여부를 확인합니다.
NCryptKeyDerivation 지정된 키 파생 함수를 사용하여 다른 키에서 키를 만듭니다.
NCryptNotifyChangeKey 키 변경 알림을 만들거나 제거합니다.
NCryptOpenKey

지정된 CNG 키 스토리지 공급자에 있는 키를 엽니다.
NCryptOpenStorageProvider CNG 키 스토리지 공급자를 로드하고 초기화합니다.
NCryptSecretAgreement 프라이빗 및 공개 키에서 비밀 계약 값을 만듭니다. (NCryptSecretAgreement)
NCryptSetProperty CNG 키 스토리지 개체의 명명된 속성 값을 설정합니다.
NCryptSignHash 해시 값의 서명을 만듭니다. (NCryptSignHash)
NCryptTranslateHandle CryptoAPI 핸들을 CNG 키 핸들로 변환합니다.
NCryptVerifyClaim 키 증명 클레임을 확인합니다.
NCryptVerifySignature 지정된 서명이 지정된 해시와 일치하는지 확인합니다. (NCryptVerifySignature)

구조

테이블 확장

NCRYPT_ALLOC_PARA 데이터를 할당하고 해제하는 데 사용할 수 있는 사용자 지정 함수를 지정할 수 있습니다.
NCRYPT_KEY_BLOB_HEADER 키 BLOB을 포함합니다.
NCRYPT_SUPPORTED_LENGTHS 키에 대한 길이 정보를 포함하기 위해 NCRYPT_LENGTHS_PROPERTY 속성과 함께 사용됩니다.

NCRYPT_UI_POLICY 키에 대한 강력한 키 사용자 인터페이스 정보를 포함하기 위해 NCRYPT_UI_POLICY_PROPERTY 속성과 함께 사용됩니다.
NCryptAlgorithmName CNG 알고리즘에 대한 정보를 포함하는 데 사용됩니다.
NCryptKeyName CNG 키에 대한 정보를 포함하는 데 사용됩니다.
NCryptProviderName CNG 키 스토리지 공급자의 이름을 포함하는 데 사용됩니다.

Last updated on 2025. 08. 16.

ncryptprotect.h 헤더

아티클2023. 08. 25.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

ncryptprotect.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

NCryptCloseProtectionDescriptor 보호 설명자 개체를 0으로 해제하고 해당 핸들을 해제합니다.
NCryptCreateProtectionDescriptor 보호 설명자 개체에 대한 핸들을 검색합니다.
NCryptGetProtectionDescriptorInfo 보호 설명자 규칙 문자열을 검색합니다.
NCryptProtectSecret 지정된 보호 설명자로 데이터를 암호화합니다.
NCryptQueryProtectionDescriptorName 등록된 설명자 표시 이름과 연결된 보호 설명자 규칙 문자열을 검색합니다.
NCryptRegisterProtectionDescriptorName 보호 설명자에 대한 표시 이름 및 연결된 규칙 문자열을 등록합니다.
NCryptStreamClose NCryptStreamOpenToProtect 또는 NCryptStreamOpenToUnprotect 함수를 사용하여 연 데이터 보호 스트림 개체를 닫습니다.
NCryptStreamOpenToProtect 지정된 보호 설명자로 대량의 데이터를 암호화하는 데 사용할 수 있는 스트림 개체를 엽니다.
NCryptStreamOpenToUnprotect

많은 양의 데이터를 암호화에 사용되는 것과 동일한 보호 설명자로 암호 해독하는 데 사용할 수 있는 스트림 개체를 엽니다. (NCryptStreamOpenToUnprotect)
NCryptStreamOpenToUnprotectEx 많은 양의 데이터를 암호화에 사용되는 것과 동일한 보호 설명자로 암호 해독하는 데 사용할 수 있는 스트림 개체를 엽니다. (NCryptStreamOpenToUnprotectEx)
NCryptStreamUpdate 데이터 블록을 암호화하고 암호 해독합니다.
NCryptUnprotectSecret 데이터를 지정된 보호 설명자로 암호 해독합니다.

콜백 함수

PFNCryptStreamOutputCallback NCryptStreamOpenToProtect 또는 NCryptStreamOpenToUnprotect 함수를 사용하여 시작된 작업에서 암호화되거나 암호 해독된 데이터를 받습니다.

구조체

NCRYPT_PROTECT_STREAM_INFO NCryptStreamOpenToProtect 및 NCryptStreamOpenToUnprotect 함수에서 처리된 데이터 블록을 애플리케이션에 전달하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ndattrib.h 헤더

아티클2023. 08. 25.

이 헤더는 네트워크 진단 프레임워크에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 진단 프레임워크

ndattrib.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

DIAG_SOCKADDR Windows 소켓 통신에 참여하는 컴퓨터의 IP(인터넷 프로토콜) 주소를 저장합니다.
HELPER_ATTRIBUTE HELPER_ATTRIBUTE 구조에는 지원되는 모든 NDF 데이터 형식이 포함됩니다.
LIFE_TIME LIFE_TIME 구조에는 시작 시간과 종료 시간이 포함됩니다.
OCTET_STRING OCTET_STRING 구조체에는 바이트 데이터 문자열에 대한 포인터가 포함되어 있습니다.
RepairInfo RepairInfo 구조에는 특정 복구 옵션에 필요한 데이터가 포함되어 있습니다.
RepairInfoEx 인시던트 근본 원인을 resolve 데 사용할 수 있는 자세한 복구 정보가 포함되어 있습니다.
RootCauseInfo 인시던트 근본 원인에 대한 자세한 정보를 포함합니다.
ShellCommandInfo ShellCommandInfo 구조에는 수동 복구 옵션을 위해 추가 애플리케이션을 시작하는 데 필요한 데이터가 포함되어 있습니다.
UiInfo

UiInfo 구조체는 복구 메시지를 사용자에게 표시하는 데 사용됩니다.

열거형

ATTRIBUTE_TYPE ATTRIBUTE_TYPE 열거형은 도우미 특성에 대해 가능한 값을 정의합니다.
REPAIR_RISK REPAIR_RISK 열거형은 복구 변경 내용이 영구적인지 여부와 취소할 수 있는지 여부를 지정합니다.
REPAIR_SCOPE REPAIR_SCOPE 열거형은 지정된 복구에 대한 수정 scope 설명합니다.
UI_INFO_TYPE UI_INFO_TYPE 열거형은 사용자 인터페이스 작업을 수행하는 복구를 식별합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ndfapi.h 헤더

아티클2023. 03. 08.

이 헤더는 네트워크 진단 프레임워크에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 진단 프레임워크

ndfapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

NdfCancelIncident 기존 인시던트에서 이전에 호출된 불필요한 함수를 취소하는 데 사용됩니다.
NdfCloseIncident 해결 후 NDF(네트워크 진단 프레임워크) 인시던트를 닫는 데 사용됩니다.
NdfCreateConnectivityIncident 일반적인 인터넷 연결 문제를 진단합니다.
NdfCreateDNSIncident 특정 호스트 이름을 해결하는 데 발생하는 이름 확인 문제를 진단합니다.
NdfCreateGroupingIncident 피어 투 피어 그룹화 기능 문제를 진단하는 세션을 만듭니다.
NdfCreateInboundIncident 특정 애플리케이션 또는 서비스에 대한 인바운드 연결을 진단하는 세션을 만듭니다.
NdfCreateIncident 애플리케이션에 통합된 NDF 기능을 테스트합니다.
NdfCreateNetConnectionIncident NetConnection 도우미 클래스를 사용하여 연결 문제를 진단합니다.

NdfCreatePnrpIncident PNRP(피어 이름 확인 프로토콜) 서비스의 문제를 진단하는 세션을 만듭니다.
NdfCreateSharingIncident 특정 네트워크 공유 액세스의 네트워크 문제를 진단합니다.
NdfCreateWebIncident 웹 연결 문제를 진단합니다. (NdfCreateWebIncident)
NdfCreateWebIncidentEx 웹 연결 문제를 진단합니다. (NdfCreateWebIncidentEx)
NdfCreateWinSockIncident Microsoft에서 제공하는 Winsock 도우미 클래스에 대한 액세스를 제공합니다.
NdfDiagnoseIncident 사용자 인터페이스를 표시하지 않고 인시던트 근본 원인을 진단합니다.
NdfExecuteDiagnosis NdfExecuteDiagnosis 함수는 발생한 인시던트 근본 원인을 진단하는 데 사용됩니다.
NdfGetTraceFile 진단 세션에서 ETW(Windows용 이벤트 추적) 이벤트가 포함된 ETL(이벤트 추적 로그) 파일이 포함된 경로를 검색하는 데 사용됩니다.
NdfRepairIncident 사용자 인터페이스를 표시하지 않고 인시던트를 복구합니다.

ndhelper.h 헤더

아티클2023. 08. 25.

이 헤더는 네트워크 진단 프레임워크에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 진단 프레임워크

ndhelper.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

INetDiagHelper INetDiagHelper 인터페이스는 네트워크 관련 문제의 진단 및 해결과 관련된 정보를 캡처하고 제공하는 메서드를 제공합니다.
INetDiagHelperEx INetDiagHelper 인터페이스를 확장하여 네트워크 관련 문제의 진단 및 해결과 관련된 정보를 캡처하고 제공하는 메서드를 제공합니다.
INetDiagHelperInfo INetDiagHelperInfo 인터페이스는 도우미 클래스에 필요한 정보가 있고 올바른 도우미 클래스를 선택했는지 확인해야 하는 경우 NDF(네트워크 진단 프레임워크)에서 호출되는 메서드를 제공합니다.
INetDiagHelperUtilFactory NDF(네트워크 진단 프레임워크)에서 사용하는 예약된 메서드를 제공합니다.

구조체

DiagnosticsInfo DiagnosticsInfo 구조에는 진단 시간의 추정치와 호출 플래그가 포함됩니다.
HelperAttributeInfo HelperAttributeInfo 구조체에는 도우미 특성의 이름과 해당 형식이 포함됩니다.

가설 HYPOTHESIS 구조에는 다른 도우미 클래스에 대한 가설을 NDF에 제출하는 데 사용되는 데이터가 포함되어 있습니다.
HypothesisResult 도우미 클래스에서 반환된 가설에 대한 정보를 포함합니다.

열거형

DIAGNOSIS_STATUS DIAGNOSIS_STATUS 열거형은 구성 요소의 상태가 결정된 도우미 클래스에 제출된 가설의 결과를 설명합니다.
PROBLEM_TYPE PROBLEM_TYPE 열거형은 도우미 클래스가 나타내는 문제의 유형을 설명합니다.
REPAIR_STATUS REPAIR_STATUS 열거형은 도우미 클래스가 복구 옵션을 시도한 결과를 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ndkinfo.h 헤더

아티클2023. 03. 08.

이 헤더는 네트워크 드라이버, Windows Vista 및 이후 버전에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 드라이버, Windows Vista 이상

ndkinfo.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

NDK_ADAPTER_INFO NDK_ADAPTER_INFO 구조는 NDK 어댑터의 제한 및 기능에 대한 정보를 지정합니다.
NDK_VERSION NDK_VERSION 구조체는 NDK 인터페이스에서 주 버전과 부 버전을 지정합니다.

netcon.h 헤더

아티클2023. 03. 08.

이 헤더는 Windows 방화벽 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 방화벽 기술

netcon.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumNetSharingEveryConnection IEnumNetSharingEveryConnection 인터페이스는 Connections 폴더의 모든 연결을 열거하는 메서드를 제공합니다.
IEnumNetSharingPortMapping IEnumNetSharingPortMapping 인터페이스는 특정 연결에 대한 포트 매핑을 열거하는 메서드를 제공합니다.
IEnumNetSharingPrivateConnection IEnumNetSharingPrivateConnection 인터페이스는 현재 구성된 비공개 공유 연결을 열거하는 메서드를 제공합니다.
IEnumNetSharingPublicConnection IEnumNetSharingPublicConnection 인터페이스는 현재 구성된 공개적으로 공유된 연결을 열거하는 메서드를 제공합니다.
INetConnection INetConnection 인터페이스는 네트워크 연결을 관리하는 메서드를 제공합니다.
INetConnectionProps INetConnectionProps 인터페이스를 사용하여 연결에 대한 속성을 검색합니다.
INetSharingConfiguration INetSharingConfiguration 인터페이스는 연결 공유, 포트 매핑 및 인터넷 연결 방화벽을 관리하는 메서드를 제공합니다.

INetSharingEveryConnectionCollection INetSharingEveryConnectionCollection 인터페이스를 사용하면 VBScript 및 JScript와 같은 언어를 스크립팅하여 연결 폴더의 모든 연결을 열거할 수 있습니다.
INetSharingManager INetSharingManager 인터페이스는 Manager 개체의 기본 인터페이스입니다. INetSharingManager는 공유가 설치되어 있는지 확인하고, 포트 매핑을 관리하고, 퍼블릭 및 프라이빗 연결에 대한 열거형 인터페이스를 가져오는 메서드를 제공합니다.
INetSharingPortMapping INetSharingPortMapping 인터페이스는 특정 포트 매핑을 관리하기 위한 메서드를 제공합니다.
INetSharingPortMappingCollection INetSharingPortMappingCollection 인터페이스를 사용하면 VBScript 및 JScript와 같은 언어를 스크립팅하여 포트 매핑을 열거할 수 있습니다.
INetSharingPortMappingProps INetSharingPortMappingProps 인터페이스는 네트워크 연결 포트 매핑의 속성을 검색하고 설정하는 메서드를 제공합니다.
INetSharingPrivateConnectionCollection INetSharingPrivateConnectionCollection 인터페이스를 사용하면 VBScript 및 JScript와 같은 언어를 스크립팅하여 프라이빗 연결을 열거할 수 있습니다.
INetSharingPublicConnectionCollection INetSharingPublicConnectionCollection 인터페이스를 사용하면 VBScript 및 JScript와 같은 언어를 스크립팅하여 공용 연결을 열거할 수 있습니다.

함수

NcFreeNetconProperties NcFreeNetconProperties 함수는 NETCON_PROPERTIES 구조와 연결된 메모리를 해제합니다.
NcIsValidConnectionName NcIsValidConnectionName 함수는 전달된 연결 이름이 유효한지 확인합니다.

구조체

NETCON_PROPERTIES NETCON_PROPERTIES 구조체는 네트워크 연결의 속성을 설명하는 값을 저장합니다.

열거형

NETCON_CHARACTERISTIC_FLAGS NETCON_CHARACTERISTIC_FLAGS 열거형 유형은 네트워크 연결에 대해 가능한 특성을 지정합니다.
NETCON_MEDIATYPE NETCON_MEDIATYPE 값은 컴퓨터가 네트워크에 연결할 수 있는 방법을 열거합니다.
NETCON_STATUS NETCON_STATUS 형식은 네트워크 연결에 대해 가능한 상태 조건을 열거합니다.
NETCON_TYPE NETCON_TYPE 형식은 다양한 종류의 네트워크 연결을 열거합니다.
SHARINGCONNECTIONTYPE SHARINGCONNECTIONTYPE 형식의 값은 가능한 유형의 공유 연결을 열거합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

netfw.h 헤더

이 헤더는 Windows 방화벽 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 방화벽 기술

netfw.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

테이블 확장

INetFwAuthorizedApplication INetFwAuthorizedApplication 인터페이스는 권한이 부여된 애플리케이션의 속성에 대한 액세스를 방화벽에 여는 기능을 제공합니다.
INetFwAuthorizedApplications INetFwAuthorizedApplications 인터페이스는 방화벽에서 권한 있는 열린 포트의 애플리케이션 컬렉션에 대한 액세스를 제공합니다.
INetFwIcmpSettings INetFwIcmpSettings 인터페이스는 ICMP 패킷을 제어하는 설정에 대한 액세스를 제공합니다.
INetFwMgr INetFwMgr 인터페이스는 컴퓨터의 방화벽 설정에 대한 액세스를 제공합니다.
INetFwOpenPort INetFwOpenPort 인터페이스는 방화벽에서 열린 포트의 속성에 대한 액세스를 제공합니다.
INetFwOpenPorts INetFwOpenPorts 인터페이스는 표준 Automation 컬렉션 인터페이스입니다.
INetFwPolicy INetFwPolicy 인터페이스는 방화벽 정책에 대한 액세스를 제공합니다.
INetFwPolicy2 방화벽 정책에 액세스하려면

INetFwProduct 타사 방화벽 등록의 속성에 액세스하려면
INetFwProducts Windows 방화벽에 타사 방화벽 제품을 등록하고 등록된 제품을 열거하기 위한 방법 및 속성에 액세스하려면
INetFwProfile INetFwProfile 인터페이스는 방화벽 설정 프로필에 대한 액세스를 제공합니다.
INetFwRemoteAdminSettings INetFwRemoteAdminSettings 인터페이스는 원격 관리를 제어하는 설정에 대한 액세스를 제공합니다.
INetFwRule 규칙의 속성으로.
INetFwRule2 애플리케이션 또는 서비스에서 INetFwRule의 모든 속성과 NET_FW_EDGE_TRAVERSAL_TYPE 지정된 방화벽 규칙의 네 가지 에지 속성에 액세스할 수 있습니다.
INetFwRule3 애플리케이션 또는 서비스가 INetFwRule2의 모든 속성에 액세스하고 앱 컨테이너의 요구 사항에 대한 액세스를 제공할 수 있도록 허용합니다.
INetFwRules 방화벽 규칙의 컬렉션입니다.
INetFwService INetFwService 인터페이스는 방화벽을 통해 수신 대기할 수 있는 권한이 부여될 수 있는 서비스의 속성에 대한 액세스를 제공합니다.
INetFwServiceRestriction Windows 서비스 강화 네트워킹 규칙에 액세스합니다.
INetFwServices INetFwServices 인터페이스는 방화벽을 통해 수신 대기 권한이 부여될 수 있는 서비스 컬렉션에 대한 액세스를 제공하는 표준 Automation 인터페이스입니다.

Functions

테이블 확장

NetworkIsolationDiagnoseConnectFailureAndGetInfo NetworkIsolationDiagnoseConnectFailureAndGetInfo 함수는 누락된 기능으로 인한 네트워크 격리 연결 실패에 대한 정보를 가져옵니다.
NetworkIsolationEnumAppContainers NetworkIsolationEnumAppContainers 함수는 시스템에 생성된 모든 앱 컨테이너를 열거합니다.
NetworkIsolationEnumerateAppContainerRules 앱 컨테이너와 관련된 모든 규칙을 열거합니다.
NetworkIsolationFreeAppContainers NetworkIsolationFreeAppContainers 함수는 하나 이상의 앱 컨테이너에 할당된 메모리 리소스를 해제하는 데 사용됩니다.
NetworkIsolationGetAppContainerConfig NetworkIsolationGetAppContainerConfig 함수는 하나 이상의 앱 컨테이너에 대한 구성 정보를 검색하는 데 사용됩니다.
NetworkIsolationGetEnterpriseIdAsync WIP(Windows Information Protection) 또는 MDAG(Microsoft Defender Application Guard) 시나리오의 컨텍스트에서 네트워크 격리 엔드포인트를 기반으로 하는 엔터프라이즈 ID를 가져옵니다.
NetworkIsolationGetEnterpriseIdClose 이 API는 NetworkIsolationGetEnterpriseIdAsync에서 반환된 핸들을 닫고 작업을 동기화하는 데 사용됩니다.
NetworkIsolationRegisterForAppContainerChanges NetworkIsolationRegisterForAppContainerChanges 함수는 앱 컨테이너 변경 내용에 대한 알림 배달을 등록하는 데 사용됩니다.
NetworkIsolationSetAppContainerConfig NetworkIsolationSetAppContainerConfig 함수는 하나 이상의 앱 컨테이너 구성을 설정하는 데 사용됩니다.
NetworkIsolationSetupAppContainerBinaries

NetworkIsolationSetupAppContainerBinaries 함수는 소프트웨어 설치 관리자가 앱 컨테이너에서 실행되는 애플리케이션의 이미지 경로에 대한 정보를 제공하는 데 사용됩니다.
NetworkIsolationUnregisterForAppContainerChanges NetworkIsolationUnregisterForAppContainerChanges 함수는 앱 컨테이너 변경 등록을 취소하고 알림 수신을 중지하는 데 사용됩니다.

콜백 함수

테이블 확장

PAC_CHANGES_CALLBACK_FN PAC_CHANGES_CALLBACK_FN 콜백 함수는 앱 컨테이너 변경 알림 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다.
PFN_FWADDDYNAMICKEYWORDADDRESS0 지정된 동적 키워드 주소를 추가하기 위해 호출하는 서비스의 진입점의 함수 포인터 형식입니다.
PFN_FWDELETEDYNAMICKEYWORDADDRESS0 지정된 ID를 사용하여 동적 키워드 주소를 삭제하기 위해 호출하는 서비스의 진입점의 함수 포인터 형식입니다.
PFN_FWENUMDYNAMICKEYWORDADDRESSBYID0 ID별로 특정 동적 키워드 주소를 열거하기 위해 호출하는 서비스의 진입점의 함수 포인터 형식입니다.
PFN_FWENUMDYNAMICKEYWORDADDRESSESBYTYPE0 형식별로 동적 키워드 주소를 열거하기 위해 호출하는 서비스의 진입점의 함수 포인터 형식입니다. 전달된 열거형 플래그를 기반으로 개체의 특정 하위 집합을 요청할 수 있습니다.
PFN_FWFREEDYNAMICKEYWORDADDRESSDATA0 서비스에서 할당한 동적 키워드 주소 데이터 구조체를 해제하기 위해 호출하는 서비스의 진입점의 함수 포인터 형식입니다.
PFN_FWUPDATEDYNAMICKEYWORDADDRESS0 동적 키워드 주소를 입력 ID로 업데이트하기 위해 호출하는 서비스의 진입점의 함수 포인터 형식입니다.

구조체

테이블 확장

FW_DYNAMIC_KEYWORD_ADDRESS_DATA0 **열거형** API가 호출되면 클라이언트에 반환된 데이터를 보유합니다.
FW_DYNAMIC_KEYWORD_ADDRESS0 클라이언트가 IP 주소 목록을 포함하는 동적 키워드 주소를 만들 수 있습니다.
INET_FIREWALL_AC_BINARIES INET_FIREWALL_AC_BINARIES 구조에는 앱 컨테이너에서 실행되는 애플리케이션에 대한 이진 경로가 포함되어 있습니다.
INET_FIREWALL_AC_CAPABILITIES INET_FIREWALL_AC_CAPABILITIES 구조에는 앱 컨테이너의 기능에 대한 정보가 포함되어 있습니다.
INET_FIREWALL_AC_CHANGE INET_FIREWALL_AC_CHANGE 구조에는 앱 컨테이너의 변경 내용에 대한 정보가 포함되어 있습니다. (INET_FIREWALL_AC_CHANGE)
INET_FIREWALL_APP_CONTAINER INET_FIREWALL_APP_CONTAINER 구조에는 특정 앱 컨테이너에 대한 정보가 포함됩니다. (INET_FIREWALL_APP_CONTAINER)

열거형

테이블 확장

FW_DYNAMIC_KEYWORD_ADDRESS_ENUM_FLAGS 열거형 작업에 포함할 개체의 종류를 지정하는 상수를 정의합니다.
FW_DYNAMIC_KEYWORD_ADDRESS_FLAGS IP 주소를 확인하는 방법을 지정하는 상수를 정의합니다.

FW_DYNAMIC_KEYWORD_ORIGIN_TYPE **FW_DYNAMIC_KEYWORD_ADDRESS_DATA0** 동적 키워드 주소 개체의 원본을 지정하는 상수를 정의합니다.
INET_FIREWALL_AC_CHANGE_TYPE INET_FIREWALL_AC_CHANGE_TYPE 열거형은 발생한 앱 컨테이너 변경 유형을 지정합니다. (INET_FIREWALL_AC_CHANGE_TYPE)
INET_FIREWALL_AC_CREATION_TYPE INET_FIREWALL_AC_CREATION_TYPE 열거형은 알림이 전달될 앱 컨테이너 만들기 이벤트의 유형을 지정합니다.
NETISO_ERROR_TYPE NETISO_ERROR_TYPE 열거형은 네트워크 격리 작업과 관련된 오류 유형을 지정합니다.
NETISO_FLAG NETISO_FLAG 열거형은 앱 컨테이너에 대해 이진 파일을 반환해야 하는지 여부를 지정합니다. (NETISO_FLAG)

Last updated on 2025. 10. 01.

netioapi.h 헤더

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- IP 도우미
- 관리 정보 기반
- 네트워크 드라이버, Windows Vista 이상

netioapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

CancelMibChangeNotify2 IP 인터페이스 변경, IP 주소 변경, IP 경로 변경, Teredo 포트 변경 및 유니캐스트 IP 주소 테이블이 안정적이며 검색할 수 있는 경우 변경 알림에 대한 등록을 취소합니다.
ConvertInterfaceAliasToLuid 네트워크 인터페이스의 인터페이스 별칭 이름을 인터페이스의 LUID(로컬 고유 식별자)로 변환합니다.
ConvertInterfaceGuidToLuid 네트워크 인터페이스의 GUID(Globally Unique Identifier)를 인터페이스의 LUID(로컬 고유 식별자)로 변환합니다.
ConvertInterfaceIndexToLuid 네트워크 인터페이스의 로컬 인덱스를 인터페이스의 LUID(로컬 고유 식별자)로 변환합니다.
ConvertInterfaceLuidToAlias 네트워크 인터페이스에 대한 LUID(로컬 고유 식별자)를 인터페이스 별칭으로 변환합니다.
ConvertInterfaceLuidToGuid 네트워크 인터페이스에 대한 LUID(로컬 고유 식별자)를 인터페이스의 GUID(Globally Unique Identifier)로 변환합니다.
ConvertInterfaceLuidToIndex 네트워크 인터페이스에 대한 LUID(로컬 고유 식별자)를 인터페이스의 로컬 인덱스로 변환합니다.

ConvertInterfaceLuidToNameA 네트워크 인터페이스의 LUID(로컬 고유 식별자)를 ANSI 인터페이스 이름으로 변환합니다.
ConvertInterfaceLuidToNameW 네트워크 인터페이스의 LUID(로컬 고유 식별자)를 유니코드 인터페이스 이름으로 변환합니다.
ConvertInterfaceNameToLuidA ANSI 네트워크 인터페이스 이름을 인터페이스의 LUID(로컬 고유 식별자)로 변환합니다.
ConvertInterfaceNameToLuidW 유니코드 네트워크 인터페이스 이름을 인터페이스의 LUID(로컬 고유 식별자)로 변환합니다.
ConvertIpv4MaskToLength IPv4 서브넷 마스크를 IPv4 접두사 길이로 변환합니다.
ConvertLengthToIpv4Mask IPv4 접두사 길이를 IPv4 서브넷 마스크로 변환합니다.
CreateAnycastIpAddressEntry 로컬 컴퓨터에 새 애니캐스트 IP 주소 항목을 추가합니다.
CreateIpForwardEntry2 로컬 컴퓨터에 새 IP 경로 항목을 만듭니다.
CreateIpNetEntry2 로컬 컴퓨터에 새 인접 IP 주소 항목을 만듭니다.
CreateSortedAddressPairs 제공된 잠재적 IP 대상 주소 목록을 사용하고, 대상 주소를 호스트 컴퓨터의 로컬 IP 주소와 페어링하고, 두 피어 간의 통신에 가장 적합한 주소 쌍에 따라 쌍을 정렬합니다.
CreateUnicastIpAddressEntry 로컬 컴퓨터에 새 유니캐스트 IP 주소 항목을 추가합니다.
DeleteAnycastIpAddressEntry 로컬 컴퓨터에서 기존 anycast IP 주소 항목을 삭제합니다.

DeleteIpForwardEntry2 로컬 컴퓨터에서 IP 경로 항목을 삭제합니다.
DeleteIpNetEntry2 로컬 컴퓨터에서 인접 IP 주소 항목을 삭제합니다.
DeleteUnicastIpAddressEntry 로컬 컴퓨터에서 기존 유니캐스트 IP 주소 항목을 삭제합니다.
FlushIpNetTable2 FlushIpNetTable2 함수는 로컬 컴퓨터의 IP 인접 테이블을 플러시합니다.
FlushIpPathTable FlushIpPathTable 함수는 로컬 컴퓨터의 IP 경로 테이블을 플러시합니다.
FreeInterfaceDnsSettings GetInterfaceDnsSettings에서 반환된 설정 개체를 해제합니다.
FreeMibTable 네트워크 인터페이스, 주소 및 경로의 테이블을 반환하는 함수에서 할당한 버퍼를 해제합니다(예: GetIfTable2 및 GetAnycastIpAddressTable).
GetAnycastIpAddressEntry 로컬 컴퓨터에서 기존 anycast IP 주소 항목에 대한 정보를 검색합니다.
GetAnycastIpAddressTable 로컬 컴퓨터에서 애니캐스트 IP 주소 테이블을 검색합니다.
GetBestRoute2 지정된 대상 IP 주소에 대한 최상의 경로를 위해 로컬 컴퓨터에서 IP 경로 항목을 검색합니다.
GetCurrentThreadCompartmentId 다음에 사용하도록 예약됩니다. 이 함수를 사용하지 마세요. (GetCurrentThreadCompartmentId)
GetDefaultCompartmentId GetDefaultCompartmentId 함수는 로컬 컴퓨터의 기본 네트워크 라우팅 구획 식별자를 검색합니다.

GetIfEntry2 로컬 컴퓨터에서 지정된 인터페이스에 대한 정보를 검색합니다.
GetIfEntry2Ex 로컬 컴퓨터에서 지정된 인터페이스에 대해 지정된 수준의 정보를 검색합니다.
GetIfStackTable 인터페이스 스택에서 네트워크 인터페이스의 관계를 지정하는 네트워크 인터페이스 스택 행 항목 테이블을 검색합니다.
GetIfTable2 MIB-II 인터페이스 테이블을 검색합니다. (GetIfTable2)
GetIfTable2Ex MIB-II 인터페이스 테이블을 검색합니다. (GetIfTable2Ex)
GetInterfaceDnsSettings *인터페이스* 매개 변수에 지정된 인터페이스에서 DNS 설정을 검색합니다.
GetInvertedIfStackTable 인터페이스 스택에서 네트워크 인터페이스의 관계를 지정하는 반전된 네트워크 인터페이스 스택 행 항목의 테이블을 검색합니다.
GetIpForwardEntry2 로컬 컴퓨터에서 IP 경로 항목에 대한 정보를 검색합니다.
GetIpForwardTable2 GetIpForwardTable2 함수는 로컬 컴퓨터에서 IP 경로 항목을 검색합니다.
GetIpInterfaceEntry 로컬 컴퓨터에서 지정된 인터페이스에 대한 IP 정보를 검색합니다.
GetIpInterfaceTable 로컬 컴퓨터에서 IP 인터페이스 항목을 검색합니다.
GetIpNetEntry2 로컬 컴퓨터에서 인접 IP 주소 항목에 대한 정보를 검색합니다.

GetIpNetTable2 GetIpNetTable2 함수는 로컬 컴퓨터에서 IP 인접 테이블을 검색합니다.
GetIpNetworkConnectionBandwidthEstimates 지정된 인터페이스에서 네트워크 연결에 대한 기록 대역폭 예상값을 검색합니다.
GetIpPathEntry 로컬 컴퓨터에서 IP 경로 항목에 대한 정보를 검색합니다.
GetIpPathTable GetIpPathTable 함수는 로컬 컴퓨터에서 IP 경로 테이블을 검색합니다.
GetMulticastIpAddressEntry 로컬 컴퓨터에서 기존 멀티캐스트 IP 주소 항목에 대한 정보를 검색합니다.
GetMulticastIpAddressTable 로컬 컴퓨터에서 멀티캐스트 IP 주소 테이블을 검색합니다.
GetNetworkConnectivityHint 애플리케이션 또는 서비스에서 발생할 수 있는 네트워크 연결의 집계 수준 및 비용을 검색합니다.
GetNetworkConnectivityHintForInterface 지정된 인터페이스에 대한 네트워크 연결 수준 및 비용을 검색합니다.
GetNetworkInformation 다음에 사용하도록 예약됩니다. 이 함수를 사용하지 마세요. (GetNetworkInformation)
GetSessionCompartmentId 다음에 사용하도록 예약됩니다. 이 함수를 사용하지 마세요. (GetSessionCompartmentId)
GetTeredoPort 로컬 컴퓨터에서 Teredo 클라이언트에서 사용하는 동적 UDP 포트 번호를 검색합니다.
GetUnicastIpAddressEntry 로컬 컴퓨터에서 기존 유니캐스트 IP 주소 항목에 대한 정보를 검색합니다.
GetUnicastIpAddressTable

로컬 컴퓨터에서 유니캐스트 IP 주소 테이블을 검색합니다.
if_indextoname 네트워크 인터페이스의 로컬 인덱스를 ANSI 인터페이스 이름으로 변환합니다.
if_nametoindex 네트워크 인터페이스의 ANSI 인터페이스 이름을 인터페이스의 로컬 인덱스로 변환합니다.
InitializeIpForwardEntry 로컬 컴퓨터의 IP 경로 항목에 대한 기본값을 사용하여 MIB_IPFORWARD_ROW2 구조를 초기화합니다.
InitializeIpInterfaceEntry 기본값을 사용하여 MIB_IPINTERFACE_ROW 항목의 멤버를 초기화합니다.
InitializeUnicastIpAddressEntry 로컬 컴퓨터의 유니캐스트 IP 주소 항목에 대한 기본값을 사용하여 MIB_UNICASTIPADDRESS_ROW 구조를 초기화합니다.
NotifyIpInterfaceChange 로컬 컴퓨터의 모든 IP 인터페이스, IPv4 인터페이스 또는 IPv6 인터페이스 변경에 대한 알림을 받도록 등록합니다.
NotifyNetworkConnectivityHintChange 집계 네트워크 연결 수준 및 비용 힌트가 변경될 때 호출할 애플리케이션 정의 콜백 함수를 등록합니다.
NotifyRouteChange2 로컬 컴퓨터의 IP 경로 항목 변경에 대한 알림을 받도록 등록합니다.
NotifyStableUnicastIpAddressTable 로컬 컴퓨터에서 안정적인 유니캐스트 IP 주소 테이블을 검색합니다.
NotifyTeredoPortChange 로컬 컴퓨터의 Teredo 서비스 포트에 대해 Teredo 클라이언트에서 사용하는 UDP 포트 번호 변경에 대한 알림을 받도록 등록합니다.
NotifyUnicastIpAddressChange 로컬 컴퓨터의 모든 유니캐스트 IP 인터페이스, 유니캐스트 IPv4 주소 또는 유니캐스트 IPv6 주소에 대한 변경 내용에 대한 알림을 받도록 등록합니다.

ResolveIpNetEntry2 로컬 컴퓨터의 인접 IP 주소 항목에 대한 실제 주소를 확인합니다. (ResolveIpNetEntry2)
SetCurrentThreadCompartmentId 다음에 사용하도록 예약됩니다. 이 함수를 사용하지 마세요. (SetCurrentThreadCompartmentId)
SetInterfaceDnsSettings *Settings* 매개 변수에 지정된 인터페이스별 DNS 설정을 설정합니다.
SetIpForwardEntry2 로컬 컴퓨터에서 IP 경로 항목의 속성을 설정합니다.
SetIpInterfaceEntry 로컬 컴퓨터에서 IP 인터페이스의 속성을 설정합니다.
SetIpNetEntry2 로컬 컴퓨터에서 기존 인접 IP 주소 항목의 실제 주소를 설정합니다.
SetNetworkInformation 다음에 사용하도록 예약됩니다. 이 함수를 사용하지 마세요. (SetNetworkInformation)
SetSessionCompartmentId 다음에 사용하도록 예약됩니다. 이 함수를 사용하지 마세요. (SetSessionCompartmentId)
SetUnicastIpAddressEntry 로컬 컴퓨터에서 기존 유니캐스트 IP 주소 항목의 속성을 설정합니다.

콜백 함수

테이블 확장

PNETWORK_CONNECTIVITY_HINT_CHANGE_CALLBACK 네트워크 집계 연결 수준 및 비용 힌트가 변경되면 애플리케이션 정의 함수가 호출됩니다.

구조체

테이블 확장

DNS_DOH_SERVER_SETTINGS DNS over-HTTPS 서버에 대해 설명합니다.
DNS_INTERFACE_SETTINGS **SetInterfaceDnsSettings 함수를 호출하거나 GetInterfaceDnsSettings 함수를 호출하여** 지정된 인터페이스에 대해 검색하여 지정된 인터페이스에서 구성할 수 있는 DNS 설정을 나타냅니다. (DNS_INTERFACE_SETTINGS)
DNS_INTERFACE_SETTINGS_EX **SetInterfaceDnsSettings 함수를 호출하거나 GetInterfaceDnsSettings 함수를 호출하여** 지정된 인터페이스에 대해 검색하여 지정된 인터페이스에서 구성할 수 있는 DNS 설정을 나타냅니다. (DNS_INTERFACE_SETTINGS_EX)
DNS_INTERFACE_SETTINGS3 **SetInterfaceDnsSettings 함수를 호출하거나 GetInterfaceDnsSettings 함수를 호출하여** 지정된 인터페이스에 대해 검색하여 지정된 인터페이스에서 구성할 수 있는 DNS 설정을 나타냅니다. (DNS_INTERFACE_SETTINGS3)
DNS_SERVER_PROPERTY **DNS_INTERFACE_SETTINGS3** 구조에서 설정되고 **SetInterfaceDnsSettings** 함수를 통해 구성된 DNS 서버 속성에 대해 설명합니다.
DNS_SERVER_PROPERTY_TYPES DNS 서버 속성에 대한 포인터를 포함합니다. 속성의 형식은 DNS_SERVER_PROPERTY::Type의 값에 따라 달라집니다.
IP_ADDRESS_PREFIX IP 주소 접두사를 저장합니다. (IP_ADDRESS_PREFIX)
MIB_ANYCASTIPADDRESS_ROW 애니캐스트 IP 주소에 대한 정보를 저장합니다.
MIB_ANYCASTIPADDRESS_TABLE anycast IP 주소 항목의 테이블을 포함합니다.

MIB_IF_ROW2 특정 인터페이스에 대한 정보를 저장합니다. (MIB_IF_ROW2)
MIB_IF_TABLE2 논리 및 물리적 인터페이스 항목의 테이블을 포함합니다.
MIB_IFSTACK_ROW 두 네트워크 인터페이스 간의 관계를 나타냅니다. (MIB_IFSTACK_ROW)
MIB_IFSTACK_TABLE 네트워크 인터페이스 스택 행 항목의 테이블을 포함합니다. 인터페이스 스택에서 네트워크 인터페이스의 관계를 지정합니다.
MIB_INVERTEDIFSTACK_ROW 두 네트워크 인터페이스 간의 관계를 나타냅니다. (MIB_INVERTEDIFSTACK_ROW)
MIB_INVERTEDIFSTACK_TABLE 반전된 네트워크 인터페이스 스택 행 항목의 테이블을 포함합니다. 그러면 인터페이스 스택의 네트워크 인터페이스 관계가 역순으로 지정됩니다.
MIB_IP_NETWORK_CONNECTION_BANDWIDTH_ESTIMATES 네트워크 연결에 대한 TCP/IP 스택에서 계산한 대역폭 예상값에 대한 읽기 전용 정보를 포함합니다.
MIB_IPFORWARD_ROW2 IP 경로 항목에 대한 정보를 저장합니다.
MIB_IPFORWARD_TABLE2 IP 경로 항목의 테이블을 포함합니다.
MIB_IPINTERFACE_ROW 네트워크 인터페이스에 특정 IP 주소 패밀리에 대한 인터페이스 관리 정보를 저장합니다.
MIB_IPINTERFACE_TABLE IP 인터페이스 항목의 테이블을 포함합니다.
MIB_IPNET_ROW2 인접 IP 주소에 대한 정보를 저장합니다.

MIB_IPNET_TABLE2 인접 IP 주소 항목의 테이블을 포함합니다.
MIB_IPPATH_ROW IP 경로 항목에 대한 정보를 저장합니다.
MIB_IPPATH_TABLE IP 경로 항목의 테이블을 포함합니다.
MIB_MULTICASTIPADDRESS_ROW 멀티캐스트 IP 주소에 대한 정보를 저장합니다.
MIB_MULTICASTIPADDRESS_TABLE 멀티캐스트 IP 주소 항목의 테이블을 포함합니다.
MIB_UNICASTIPADDRESS_ROW 유니캐스트 IP 주소에 대한 정보를 저장합니다.
MIB_UNICASTIPADDRESS_TABLE 유니캐스트 IP 주소 항목의 테이블을 포함합니다.

열거형

테이블 확장

DNS_SERVER_PROPERTY_TYPE DNS_SERVER_PROPERTY::P roperty 멤버에 있는 속성의 유효성을 지정하는 상수를 정의합니다.
MIB_IF_TABLE_LEVEL MIB_IF_TABLE_LEVEL 열거형 형식은 검색할 인터페이스 정보의 수준을 정의합니다.
MIB_NOTIFICATION_TYPE 알림이 발생할 때 콜백 함수에 전달되는 알림 유형을 정의합니다.

Last updated on 2025. 10. 01.

netiodef.h 헤더

아티클2023. 03. 16.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 드라이버, Windows Vista 이상
- Windows 필터링 플랫폼

netiodef.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

열거형

DL_ADDRESS_TYPE DL_ADDRESS_TYPE 열거형(netiodef.h)은 데이터 링크 계층 주소의 형식을 지정합니다.

netlistmgr.h 헤더

아티클2023. 03. 16.

이 헤더는 네트워크 목록 관리자에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 목록 관리자

netlistmgr.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumNetworkConnections IEnumNetworkConnections 인터페이스는 네트워크 연결에 대한 표준 열거자를 제공합니다. 네트워크 내의 활성, 연결 끊김 또는 모든 네트워크 연결을 열거합니다. 이 인터페이스는 INetwork 인터페이스에서 가져올 수 있습니다.
IEnumNetworks IEnumNetworks 인터페이스는 네트워크의 표준 열거자입니다. 로컬 컴퓨터에서 사용할 수 있는 모든 네트워크를 열거합니다. 이 인터페이스는 INetworkListManager 인터페이스에서 가져올 수 있습니다.
INetwork INetwork 인터페이스는 로컬 컴퓨터의 네트워크를 나타냅니다. 또한 비슷한 네트워크 서명이 있는 네트워크 연결 컬렉션을 나타낼 수도 있습니다.
INetwork2 **INetwork2** 인터페이스는 로컬 컴퓨터의 네트워크를 나타냅니다. 또한 비슷한 네트워크 서명이 있는 네트워크 연결 컬렉션을 나타낼 수도 있습니다.
INetworkConnection INetworkConnection 인터페이스는 단일 네트워크 연결을 나타냅니다.
INetworkConnection2 **INetworkConnection2** 인터페이스는 단일 네트워크 연결을 나타냅니다.
INetworkConnectionCost 연결과 연결된 현재 네트워크 비용 및 데이터 계획 상태를 쿼리합니다.

INetworkConnectionCostEvents 이 인터페이스는 애플리케이션에 연결에 대한 비용 및 데이터 계획 상태 변경 이벤트를 알립니다.
INetworkConnectionEvents INetworkConnectionEvents 인터페이스는 클라이언트가 네트워크 연결 관련 이벤트를 가져오기 위해 구현하는 메시지 싱크 인터페이스입니다. 하위 수준 이벤트(예: 인증 변경)에 관심이 있는 애플리케이션은 이 인터페이스를 구현해야 합니다.
INetworkCostManager 이 인터페이스를 사용하여 컴퓨터 전체 인터넷 연결에 사용되는 연결 또는 연결의 특정 대상에 대한 라우팅의 첫 번째 홉과 관련된 컴퓨터 차원의 비용 및 데이터 계획 상태 정보를 쿼리합니다.
INetworkCostManagerEvents 이 인터페이스는 머신 전체 비용 및 데이터 계획 관련 이벤트를 애플리케이션에 알립니다.
INetworkEvents INetworkEvents는 클라이언트가 네트워크 관련 이벤트를 가져오기 위해 구현하는 알림 싱크 인터페이스입니다. 이러한 API는 각 이벤트가 발생할 때 자동으로 호출되는 모든 콜백 함수입니다.
INetworkListManager INetworkListManager 인터페이스는 네트워크 목록 관리 기능을 수행하는 메서드 집합을 제공합니다.
INetworkListManagerEvents INetworkListManagerEvents는 클라이언트가 전반적인 컴퓨터 상태 관련 이벤트를 가져오기 위해 구현하는 메시지 싱크 인터페이스입니다. 인터넷 연결과 같은 상위 수준 이벤트에 관심이 있는 애플리케이션은 이 인터페이스를 구현합니다.

구조체

NLM_DATAPLAN_STATUS NLM_DATAPLAN_STATUS 구조는 운송업체에서 제공하는 현재 데이터 플랜 상태 정보를 저장합니다.

NLM_SIMULATED_PROFILE_INFO SetSimulatedProfileInfo에서 RDP 자식 세션에서 현재 인터넷 연결 프로필 값을 재정의하는 데 사용되는 값을 지정하여 특정 계량된 인터넷 연결 조건의 시뮬레이션을 지원하는 데 사용됩니다.
NLM_SOCKADDR NLM_SOCKADDR 구조에는 IPv4/IPv6 대상 주소가 포함됩니다.
NLM_USAGE_DATA NLM_USAGE_DATA 구조체는 계획의 데이터 사용량을 나타내는 정보를 저장합니다.

열거형

NLM_CONNECTION_COST NLM_CONNECTION_COST 열거형은 Windows 8 Cost API에서 지원되는 비용 수준 및 비용 플래그 집합을 지정합니다.
NLM_CONNECTION_PROPERTY_CHANGE NLM_CONNECTION PROPERTY_CHANGE 열거형은 네트워크 연결의 속성에 대한 변경 내용을 정의하는 플래그 집합입니다.
NLM_CONNECTIVITY NLM_Connectivity 열거형은 연결 관련 매개 변수가 변경 될 때마다 알림을 제공하는 플래그 집합입니다.
NLM_DOMAIN_AUTHENTICATION_KIND 도메인 인증 방법을 지정하는 상수를 정의합니다.
NLM_DOMAIN_TYPE NLM_DOMAIN_TYPE 열거형은 네트워크의 도메인 유형을 지정하는 플래그 집합입니다.
NLM_ENUM_NETWORK NLM_ENUM_NETWORK 열거형에는 열거되는 네트워크 유형을 지정하는 플래그 집합이 포함되어 있습니다.

NLM_INTERNET_CONNECTIVITY NLM_INTERNET_CONNECTIVITY 열거형은 IPv4 또는 IPv6 네트워크 연결에 대한 추가 데이터를 제공하는 플래그 집합을 정의합니다.
NLM_NETWORK_CATEGORY NLM_NETWORK_CATEGORY 열거형은 네트워크의 범주 유형을 지정하는 플래그 집합입니다.
NLM_NETWORK_CLASS NLM_NETWORK_CLASS 열거형은 네트워크가 식별되었는지를 지정하는 플래그 집합을 정의합니다.
NLM_NETWORK_PROPERTY_CHANGE NLM_NETWORK_PROPERTY_CHANGE 열거형은 네트워크의 속성에 대한 변경 내용을 정의하는 플래그 집합입니다.

netsh.h 헤더

아티클2023. 08. 25.

이 헤더는 NetShell에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Netshell

netsh.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

MatchEnumTag 법적 값 테이블을 검색하여 특정 토큰과 일치하는 값을 찾습니다.
MatchToken 사용자가 입력한 문자열이 특정 문자열과 일치하는지 여부를 결정합니다.
PreprocessCommand 인수 문자열을 구문 분석하고 필요한 모든 태그가 있는지 확인합니다.
PrintError NetShell 콘솔에 시스템 또는 애플리케이션 오류 메시지를 표시합니다.
PrintMessage NetShell 콘솔에 출력을 표시합니다.
PrintMessageFromModule NetShell 콘솔에 지역화된 출력을 표시합니다.
RegisterContext NetShell에 도우미 컨텍스트를 등록합니다.
RegisterHelper 도우미의 노출된 InitHelperDll 함수 내에서 호출되고 NetShell 컨텍스트에 도우미를 등록합니다.

콜백 함수

FN_HANDLE_CMD 도우미에 대한 명령 함수입니다.
NS_CONTEXT_COMMIT_FN 도우미에 대한 커밋 함수입니다.
NS_CONTEXT_CONNECT_FN 도우미를 위한 연결 함수입니다.
NS_CONTEXT_DUMP_FN 도우미에 대한 덤프 함수입니다.
NS_DLL_INIT_FN 도우미의 초기 로드를 수행하기 위해 NetShell에서 호출됩니다.
NS_DLL_STOP_FN 도우미 DLL에 대한 DLL 중지 함수입니다.
NS_HELPER_START_FN 도우미의 시작 함수입니다.
NS_HELPER_STOP_FN 도우미에 대한 중지 함수입니다.
NS_OSVERSIONCHECK 도우미를 위한 운영 체제 검사 함수입니다.

구조체

CMD_ENTRY 도우미 명령을 정의합니다.
CMD_GROUP_ENTRY 도우미 명령 그룹을 정의합니다.

NS_CONTEXT_ATTRIBUTES 컨텍스트의 특성을 정의합니다.
NS_HELPER_ATTRIBUTES 도우미의 특성을 제공합니다.
TAG_TYPE PreprocessCommand 함수에 사용되는 태그를 지정합니다.

열거형

NS_CMD_FLAGS NetShell에서 사용할 수 있는 명령 플래그를 지정합니다.
NS_REQS 이벤트 수를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

networkisolation.h 헤더

아티클2024. 05. 07.

이 헤더는 Windows 방화벽 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 방화벽 기술

networkisolation.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

NetworkIsolationDiagnoseConnectFailure 호출 애플리케이션에 대상 서버에 대한 네트워크 연결을 만드는 데 필요한 AppContainer 기능이 있는지 여부를 결정합니다.
NetworkIsolationDiagnoseConnectFailureAndGetInfo NetworkIsolationDiagnoseConnectFailureAndGetInfo 함수는 누락된 기능으로 인한 네트워크 격리 연결 실패에 대한 정보를 가져옵니다.
NetworkIsolationEnumAppContainers NetworkIsolationEnumAppContainers 함수는 시스템에서 만든 모든 앱 컨테이너를 열거합니다. (NetworkIsolationEnumAppContainers)
NetworkIsolationFreeAppContainers NetworkIsolationFreeAppContainers 함수는 하나 이상의 앱 컨테이너에 할당된 메모리 리소스를 해제하는 데 사용됩니다. (NetworkIsolationFreeAppContainers)
NetworkIsolationGetAppContainerConfig NetworkIsolationGetAppContainerConfig 함수는 하나 이상의 앱 컨테이너에 대한 구성 정보를 검색하는 데 사용됩니다.
NetworkIsolationRegisterForAppContainerChanges NetworkIsolationRegisterForAppContainerChanges 함수는 앱 컨테이너의 변경 내용과 관련된 알림 배달을 등록하는 데 사용됩니다.
NetworkIsolationSetAppContainerConfig

NetworkIsolationSetAppContainerConfig 함수는 하나 이상의 앱 컨테이너 구성을 설정하는 데 사용됩니다. (NetworkIsolationSetAppContainerConfig)
NetworkIsolationSetupAppContainerBinaries NetworkIsolationSetupAppContainerBinaries 함수는 소프트웨어 설치 관리자가 앱 컨테이너에서 실행되는 애플리케이션의 이미지 경로에 대한 정보를 제공하는 데 사용됩니다.
NetworkIsolationUnregisterForAppContainerChanges NetworkIsolationUnregisterForAppContainerChanges 함수는 앱 컨테이너 변경 등록을 취소하고 알림 수신을 중지하는 데 사용됩니다.

콜백 함수

테이블 확장

PAC_CHANGES_CALLBACK_FN PAC_CHANGES_CALLBACK_FN 콜백 함수는 앱 컨테이너 변경 알림 프로세스에 사용자 지정 동작을 추가하는 데 사용됩니다.

구조체

테이블 확장

INET_FIREWALL_AC_BINARIES INET_FIREWALL_AC_BINARIES 구조에는 앱 컨테이너에서 실행되는 애플리케이션에 대한 이진 경로가 포함됩니다. (INET_FIREWALL_AC_BINARIES)
INET_FIREWALL_AC_CAPABILITIES INET_FIREWALL_AC_CAPABILITIES 구조에는 앱 컨테이너의 기능에 대한 정보가 포함되어 있습니다. (INET_FIREWALL_AC_CAPABILITIES)
INET_FIREWALL_AC_CHANGE INET_FIREWALL_AC_CHANGE 구조에는 앱 컨테이너의 변경 내용에 대한 정보가 포함됩니다. (INET_FIREWALL_AC_CHANGE)

INET_FIREWALL_APP_CONTAINER INET_FIREWALL_APP_CONTAINER 구조에는 특정 앱 컨테이너에 대한 정보가 포함됩니다. (INET_FIREWALL_APP_CONTAINER)

열거형

테이블 확장

INET_FIREWALL_AC_CHANGE_TYPE INET_FIREWALL_AC_CHANGE_TYPE 열거형은 발생한 앱 컨테이너 변경 유형을 지정합니다. (INET_FIREWALL_AC_CHANGE_TYPE)
INET_FIREWALL_AC_CREATION_TYPE INET_FIREWALL_AC_CREATION_TYPE 열거형은 알림이 전달될 앱 컨테이너 만들기 이벤트의 유형을 지정합니다.
NETISO_ERROR_TYPE NETISO_ERROR_TYPE 열거형은 네트워크 격리 작업과 관련된 오류 유형을 지정합니다. (NETISO_ERROR_TYPE)
NETISO_FLAG NETISO_FLAG 열거형은 앱 컨테이너에 대해 이진 파일을 반환해야 하는지 여부를 지정합니다. (NETISO_FLAG)

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

newdev.h 헤더

아티클2023. 08. 25.

이 헤더는 디바이스 및 드라이버 설치 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 디바이스 및 드라이버 설치 참조

newdev.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DiInstallDevice DiInstallDevice 함수는 시스템에 있는 지정된 디바이스의 드라이버 저장소에 미리 설치된 지정된 드라이버를 설치합니다.
DiInstallDriverA DiInstallDriver 함수는 드라이버 저장소에 드라이버를 사전 설치한 다음 드라이버가 지원하는 시스템에 있는 디바이스에 드라이버를 설치합니다. (ANSI)
DiInstallDriverW DiInstallDriver 함수는 드라이버 저장소에 드라이버를 사전 설치한 다음 드라이버가 지원하는 시스템에 있는 디바이스에 드라이버를 설치합니다. (유니코드)
DiRollbackDriver DiRollbackDriver 함수는 지정된 디바이스에 설치된 드라이버를 롤백합니다.
DiShowUpdateDevice DiShowUpdateDevice 함수는 지정된 디바이스에 대한 하드웨어 업데이트 마법사를 표시합니다.
DiUninstallDevice DiUninstallDevice 함수는 디바이스를 제거하고 시스템에서 디바이스 노드(devnode)를 제거합니다.
DiUninstallDriverA DiUninstallDriver 함수는 해당 디바이스를 다른 일치하는 드라이버 패키지(사용 가능한 경우)와 함께 설치하여 설치된 모든 디바이스에서 드라이버 패키지를 제거하거나, 일치하는 다른 드라이버 패키지를 사용할 수 없는 경우 null 드라이버를 제거합니다. 그런 다음 지정된 드라이버 패키지가 드라이버 저장소에서 제거됩니다. (ANSI)

DiUninstallDriverW DiUninstallDriver 함수는 해당 디바이스를 다른 일치하는 드라이버 패키지(사용 가능한 경우)와 함께 설치하여 설치된 모든 디바이스에서 드라이버 패키지를 제거하거나, 일치하는 다른 드라이버 패키지를 사용할 수 없는 경우 null 드라이버를 제거합니다. 그런 다음 지정된 드라이버 패키지가 드라이버 저장소에서 제거됩니다. (유니코드)
UpdateDriverForPlugAndPlayDevicesA INF 파일과 하드웨어 ID가 지정된 경우 UpdateDriverForPlugAndPlayDevices 함수는 하드웨어 ID와 일치하는 디바이스에 대해 업데이트된 드라이버를 설치합니다. (ANSI)
UpdateDriverForPlugAndPlayDevicesW INF 파일과 하드웨어 ID가 지정된 경우 UpdateDriverForPlugAndPlayDevices 함수는 하드웨어 ID와 일치하는 디바이스에 대해 업데이트된 드라이버를 설치합니다. (유니코드)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

nldef.h 헤더

아티클2023. 08. 25.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- IP 도우미
- 관리 정보 기반
- 네트워크 드라이버, Windows Vista 이상

nldef.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

NL_BANDWIDTH_INFORMATION TCP/IP 스택에 의해 결정된 사용 가능한 대역폭 예상 및 관련 분산에 대한 읽기 전용 정보를 포함합니다.
NL_INTERFACE_OFFLOAD_ROD IP 인터페이스의 오프로드 기능을 나타내는 플래그 집합을 지정합니다.
NL_NETWORK_CONNECTIVITY_HINT 네트워크 연결 수준, 네트워크 연결에 대한 사용 요금 및 비용 요소를 반영하는 다른 멤버에 대해 설명합니다.

열거형

NL_ADDRESS_TYPE NL_ADDRESS_TYPE 열거형 유형은 네트워크 계층의 IP 주소 유형을 지정합니다.
NL_DAD_STATE IP_DAD_STATE 열거형은 IPv4 또는 IPv6 주소에 대한 DAD(중복 주소 검색) 상태에 대한 정보를 지정합니다.
NL_LINK_LOCAL_ADDRESS_BEHAVIOR

NL_LINK_LOCAL_ADDRESS_BEHAVIOR 열거형 형식은 링크 로컬 주소 동작을 정의합니다.
NL_NEIGHBOR_STATE NL_NEIGHBOR_STATE 열거형 형식은 RFC 2461, 섹션 7.3.2에 설명된 대로 네트워크 계층 인접 IP 주소의 상태를 정의합니다.
NL_NETWORK_CONNECTIVITY_COST_HINT 네트워크 연결에 대한 사용 요금에 대한 힌트를 지정하는 상수를 정의합니다.
NL_NETWORK_CONNECTIVITY_LEVEL_HINT 네트워크 연결 수준에 대한 힌트를 지정하는 상수를 정의합니다.
NL_PREFIX_ORIGIN IP_PREFIX_ORIGIN 열거형은 IPv4 또는 IPv6 주소 접두사 원본을 지정하고 IP_ADAPTER_UNICAST_ADDRESS 구조에 사용됩니다.
NL_ROUTE_ORIGIN NL_ROUTE_ORIGIN 열거형 형식은 IP 경로의 원본을 정의합니다.
NL_ROUTE_PROTOCOL NL_ROUTE_PROTOCOL 열거형 형식은 RFC 4292에 설명된 대로 IP 경로가 추가된 라우팅 메커니즘을 정의합니다.
NL_ROUTER_DISCOVERY_BEHAVIOR NL_ROUTER_DISCOVERY_BEHAVIOR 열거형 형식은 RFC 2461에 설명된 대로 라우터 검색 동작을 정의합니다.
NL_SUFFIX_ORIGIN IP_SUFFIX_ORIGIN 열거형은 IPv4 또는 IPv6 주소 접미사의 원본을 지정하고 IP_ADAPTER_UNICAST_ADDRESS 구조에 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

notificationactivationcallback.h 헤더

아티클2023. 08. 25.

이 헤더는 클래식 데스크톱 애플리케이션에 대한 타일, 배지 및 알림에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 클래식 데스크톱 애플리케이션에 대한 타일, 배지 및 알림

notificationactivationcallback.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

INotificationActivationCallback 알림 센터에서 알림을 통해 앱이 트리거될 때 알림 메시지를 받습니다.

구조체

NOTIFICATION_USER_INPUT_DATA 사용자가 알림 센터에서 알림 메시지와 상호 작용하는 방법에 대한 정보를 포함합니다. 이 구조체는 Activate에서 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

npapi.h 헤더

아티클2023. 03. 08.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

npapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

AddConnectNotify 각 추가 연결 작업(WNetAddConnection, WNetAddConnection2 및 WNetAddConnection3)이 MPR(다중 공급자 라우터)에 의해 시도되기 전과 후에 호출됩니다.
CancelConnectNotify 각 연결 취소 작업 전후에 CancelConnectNotify를 호출합니다(WNetCancelConnection 및 WNetCancelConnection2).
NPAddConnection 로컬 디바이스를 네트워크 리소스에 연결합니다. (NPAddConnection)
NPAddConnection3 로컬 디바이스를 네트워크 리소스에 연결합니다. (NPAddConnection3)
NPCancelConnection 네트워크 연결을 끊습니다.
NPCloseEnum 열거형을 닫습니다.
NPDeviceMode 디바이스의 부모 창을 지정합니다. 이 창은 디바이스에서 시작된 모든 대화 상자를 소유합니다.
NPDirectoryNotify 네트워크 공급자에게 특정 디렉터리 작업을 알 수 있습니다.

NPEnumResource NPOpenEnum에서 반환된 핸들을 기반으로 열거형을 수행합니다.
NPFMXEditPerm 네트워크 공급업체가 자체 사용 권한 편집기 대화 상자를 제공할 수 있도록 합니다.
NPFMXGetPermCaps 권한 편집기의 기능을 검색합니다. 반환 값은 파일 관리자에서 사용할 보안 메뉴 항목을 나타내는 비트 마스크입니다.
NPFMXGetPermHelp 파일 관리자의 보안 메뉴에서 메뉴 항목을 선택하고 F1 키를 누르면 사용 권한 편집기 대화 상자의 도움말 파일 및 도움말 컨텍스트를 검색합니다.
NPFormatNetworkName 컨트롤에 표시할 공급자별 형식으로 네트워크 이름 서식을 지정합니다.
NPGetCaps 네트워크에서 지원되는 서비스에 대한 정보를 반환합니다.
NPGetConnection 연결에 대한 정보를 검색합니다.
NPGetConnection3 현재 연결이 끊어진 경우에도 네트워크 연결에 대한 정보를 검색합니다.
NPGetConnectionPerformance 네트워크 리소스에 액세스하는 데 사용되는 연결의 예상 성능에 대한 정보를 반환합니다. 요청은 현재 연결된 네트워크 리소스에 대해서만 가능합니다.
NPGetDirectoryType 네트워크 디렉터리의 유형을 결정합니다.
NPGetPropertyText 네트워크 리소스의 속성 대화 상자에 추가할 단추의 이름을 검색합니다.

NPGetResourceInformation WNet API를 통해 액세스되는 네트워크 리소스의 부분을 리소스 유형과 관련된 API를 통해 액세스하는 부분과 구분합니다.
NPGetResourceParent 찾아보기 계층에서 지정된 네트워크 리소스의 부모를 검색합니다.
NPGetUniversalName 네트워크 리소스의 범용 이름을 검색합니다. NPGetUniversalName 함수는 UNC 형식 또는 이전의 원격 이름 형식으로 이 유니버설 이름을 검색할 수 있습니다.
NPGetUser 네트워크 연결을 설정하는 데 사용되는 현재 기본 사용자 이름 또는 사용자 이름의 값을 검색합니다.
NPLogonNotify MPR은 이 함수를 호출하여 자격 증명 관리자에게 로그온 이벤트가 발생했음을 알리고 자격 증명 관리자가 로그온 스크립트를 반환할 수 있도록 합니다.
NPOpenEnum 네트워크 리소스 또는 기존 연결의 열거형을 엽니다. 열거형에 유효한 핸들을 가져오려면 NPOpenEnum 함수를 호출해야 합니다.
NPPasswordChangeNotify MPR은 이 함수를 호출하여 자격 증명 관리자에게 암호 변경 이벤트를 알립니다.
NPPropertyDialog 사용자가 NPPropertyDialog 함수를 사용하여 추가된 단추를 클릭할 때 호출됩니다. NPPropertyDialog 함수는 파일 및 디렉터리 네트워크 속성에 대해서만 호출됩니다.
NPSearchDialog 네트워크 공급업체가 연결 대화 상자에 표시되는 계층적 보기를 넘어 고유한 형태의 검색 및 검색을 제공할 수 있습니다.
WNetSetLastErrorA 확장 오류 정보를 설정합니다. 네트워크 공급자는 SetLastError 대신 이 함수를 호출해야 합니다. (ANSI)

WNetSetLastErrorW 확장 오류 정보를 설정합니다. 네트워크 공급자는 SetLastError 대신 이 함수를 호출해야 합니다. (유니코드)

구조체

NOTIFYADD NOTIFYADD 구조에는 네트워크 연결 작업의 세부 정보가 포함됩니다. AddConnectNotify 함수에서 사용됩니다.
NOTIFYCANCEL NOTIFYCANCEL 구조에는 네트워크 연결 끊기 작업의 세부 정보가 포함됩니다. CancelConnectNotify 함수에서 사용됩니다.
NOTIFYINFO NOTIFYINFO 구조에는 네트워크 연결 또는 연결 끊기 작업에 대한 상태 정보가 포함됩니다. AddConnectNotify 및 CancelConnectNotify 함수에서 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

nsemail.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows 소켓 2에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 소켓 2

nsemail.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

NAPI_DOMAIN_DESCRIPTION_BLOB NS_EMAIL 네임스페이스에 대한 네임스페이스 공급자가 처리하는 도메인에 대해 설명합니다.
NAPI_PROVIDER_INSTALLATION_BLOB NS_EMAIL 네임스페이스에 대한 네임스페이스 공급자를 설치하는 데 필요한 정보를 포함합니다.

열거형

NAPI_PROVIDER_LEVEL 지정된 도메인에 대한 NS_EMAIL 네임스페이스 공급자의 공급자 기관 수준을 지정합니다.
NAPI_PROVIDER_TYPE 네임스페이스 공급자에 대해 예상되는 호스팅 유형을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

nspapi.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows 소켓 2에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 소켓 2

nspapi.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

EnumProtocolsA EnumProtocols 함수는 로컬 호스트에서 활성 상태인 지정된 네트워크 프로토콜 집합에 대한 정보를 검색합니다. (ANSI)
EnumProtocolsW EnumProtocols 함수는 로컬 호스트에서 활성 상태인 지정된 네트워크 프로토콜 집합에 대한 정보를 검색합니다. (유니코드)
GetAddressByNameA Windows 소켓 2를 기준으로 GetAddressByName을 더 이상 사용할 수 없습니다. (ANSI)
GetAddressByNameW Windows 소켓 2를 기준으로 GetAddressByName을 더 이상 사용할 수 없습니다. (유니코드)
GetNameByTypeA GetNameByType 함수는 지정된 서비스 유형에 대한 네트워크 서비스의 이름을 검색합니다. (ANSI)
GetNameByTypeW GetNameByType 함수는 지정된 서비스 유형에 대한 네트워크 서비스의 이름을 검색합니다. (유니코드)
GetServiceA GetService 함수는 기본 네임스페이스 집합 또는 지정된 네임스페이스의 컨텍스트에서 네트워크 서비스에 대한 정보를 검색합니다. (ANSI)
GetServiceW

GetService 함수는 기본 네임스페이스 집합 또는 지정된 네임스페이스의 컨텍스트에서 네트워크 서비스에 대한 정보를 검색합니다. (유니코드)
GetTypeByNameA GetTypeByName 함수는 이름으로 지정된 네트워크 서비스에 대한 서비스 유형 GUID를 검색합니다. (ANSI)
GetTypeByNameW GetTypeByName 함수는 이름으로 지정된 네트워크 서비스에 대한 서비스 유형 GUID를 검색합니다. (유니코드)
SetServiceA SetService 함수는 하나 이상의 네임스페이스 내에서 네트워크 서비스를 레지스트리에서 등록하거나 제거합니다. (ANSI)
SetServiceW SetService 함수는 하나 이상의 네임스페이스 내에서 네트워크 서비스를 레지스트리에서 등록하거나 제거합니다. (유니코드)

구조체

BLOB 이진 큰 개체에서 파생된 BLOB 구조체(nspapi.h)에는 데이터 블록에 대한 정보가 포함됩니다.
CSADDR_INFO CSADDR_INFO 구조체(nspapi.h)에는 소켓, 네트워크 서비스 또는 네임스페이스 공급자에 대한 Windows 소켓 주소 정보가 포함됩니다.
NS_SERVICE_INFOA 지정된 네임스페이스 또는 기본 네임스페이스 집합의 컨텍스트에서 네트워크 서비스 또는 네트워크 서비스 유형에 대한 정보를 포함합니다. (ANSI)
NS_SERVICE_INFOW 지정된 네임스페이스 또는 기본 네임스페이스 집합의 컨텍스트에서 네트워크 서비스 또는 네트워크 서비스 유형에 대한 정보를 포함합니다. (유니코드)

PROTOCOL_INFOA 프로토콜에 대한 정보를 포함합니다. (ANSI)
PROTOCOL_INFOW 프로토콜에 대한 정보를 포함합니다. (유니코드)
SERVICE_ADDRESS 서비스에 대한 주소 정보를 포함합니다. 구조체는 RPC(원격 프로시저 호출), 명명된 파이프 및 소켓을 포함하여 다양한 유형의 IPC(Interprocess Communication) 메커니즘 및 해당 주소 양식을 수용할 수 있습니다.
SERVICE_ADDRESSES SERVICE_ADDRESSES 구조에는 SERVICE_ADDRESS 데이터 구조의 배열이 포함되어 있습니다.
SERVICE_INFOA 네트워크 서비스 또는 네트워크 서비스 유형에 대한 정보를 포함합니다. (ANSI)
SERVICE_INFOW 네트워크 서비스 또는 네트워크 서비스 유형에 대한 정보를 포함합니다. (유니코드)
SERVICE_TYPE_INFO_ABSA SERVICE_TYPE_INFO_ABS 구조에는 네트워크 서비스 유형에 대한 정보가 포함됩니다. SERVICE_TYPE_INFO_ABS 사용하여 네임스페이스에 네트워크 서비스 유형을 추가합니다. (ANSI)
SERVICE_TYPE_INFO_ABSW SERVICE_TYPE_INFO_ABS 구조에는 네트워크 서비스 유형에 대한 정보가 포함됩니다. SERVICE_TYPE_INFO_ABS 사용하여 네임스페이스에 네트워크 서비스 유형을 추가합니다. (유니코드)
SERVICE_TYPE_VALUE_ABSA 네트워크 서비스 유형 값에 대한 정보를 포함합니다. 이 정보는 네임스페이스에만 해당될 수 있습니다. (ANSI)
SERVICE_TYPE_VALUE_ABSW 네트워크 서비스 유형 값에 대한 정보를 포함합니다. 이 정보는 네임스페이스에만 해당될 수 있습니다. (유니코드)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ntddkbd.h 헤더

아티클2023. 08. 25.

키보드 디바이스에 액세스하기 위한 모든 상수 및 유형을 정의하는 포함 파일입니다.

자세한 내용은 다음을 참조하세요.

- 휴먼 인터페이스 디바이스 참조

ntddkbd.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

IOCTLs

IOCTL_KEYBOARD_QUERY_ATTRIBUTES IOCTL_KEYBOARD_QUERY_ATTRIBUTES 요청은 키보드 특성에 대한 정보를 반환합니다.
IOCTL_KEYBOARD_QUERY_EXTENDED_ATTRIBUTES IOCTL_KEYBOARD_QUERY_EXTENDED_ATTRIBUTES 요청은 확장된 키보드 특성에 대한 정보를 반환합니다.
IOCTL_KEYBOARD_QUERY_INDICATOR_TRANSLATION IOCTL_KEYBOARD_QUERY_INDICATOR_TRANSLATION 요청은 스캔 코드와 키보드 표시기 간의 매핑에 대한 정보를 반환합니다.
IOCTL_KEYBOARD_QUERY_INDICATORS IOCTL_KEYBOARD_QUERY_INDICATORS 요청은 키보드 표시기 정보를 반환합니다.
IOCTL_KEYBOARD_QUERY_TYPEMATIC IOCTL_KEYBOARD_QUERY_TYPEMATIC 요청은 키보드 형식 설정을 반환합니다.
IOCTL_KEYBOARD_SET_INDICATORS IOCTL_KEYBOARD_SET_INDICATORS 요청은 키보드 표시기를 설정합니다.
IOCTL_KEYBOARD_SET_TYPEMATIC IOCTL_KEYBOARD_SET_TYPEMATIC 요청은 키보드 형식 설정을 설정합니다.

구조체

KEYBOARD_ATTRIBUTES KEYBOARD_ATTRIBUTES 키보드의 특성을 지정합니다.
KEYBOARD_EXTENDED_ATTRIBUTES KEYBOARD_EXTENDED_ATTRIBUTES 키보드의 확장 특성을 지정합니다.
KEYBOARD_INDICATOR_PARAMETERS KEYBOARD_INDICATOR_PARAMETERS 키보드 표시기 LED의 상태를 지정합니다.
KEYBOARD_INDICATOR_TRANSLATION KEYBOARD_INDICATOR_TRANSLATION 키보드 스캔 코드와 LED 표시기 간의 디바이스별 가변 길이 매핑 배열을 지정합니다.
KEYBOARD_INPUT_DATA KEYBOARD_INPUT_DATA 하나의 키보드 입력 데이터 패킷을 포함합니다.
KEYBOARD_TYPEMATIC_PARAMETERS KEYBOARD_TYPEMATIC_PARAMETERS 키보드의 형식 설정을 지정합니다.
KEYBOARD_UNIT_ID_PARAMETER KEYBOARD_UNIT_ID_PARAMETER Kbdclass가 키보드에 할당하는 단위 ID를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ntddmou.h 헤더

아티클2023. 03. 08.

마우스 디바이스에 액세스하기 위한 모든 상수 및 형식을 정의하는 포함 파일입니다.

자세한 내용은 다음을 참조하세요.

- 휴먼 인터페이스 디바이스 참조

ntddmou.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

IOCTLs

IOCTL_MOUSE_QUERY_ATTRIBUTES IOCTL_MOUSE_QUERY_ATTRIBUTES 요청은 마우스 특성에 대한 정보를 반환합니다.

구조체

MOUSE_ATTRIBUTES MOUSE_ATTRIBUTES 마우스 디바이스의 특성을 지정합니다.
MOUSE_INPUT_DATA MOUSE_INPUT_DATA 마우스 입력 데이터의 한 패킷을 포함합니다.
MOUSE_UNIT_ID_PARAMETER MOUSE_UNIT_ID_PARAMETER Mouclass가 마우스에 할당하는 단위 ID를 지정합니다.

ntddpsch.h 헤더

이 헤더는 QOS(서비스 품질)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 서비스 품질(QOS)

ntddpsch.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조

테이블 확장

PS_ADAPTER_STATS PS_ADAPTER_STATS 구조체는 지정된 어댑터에 대한 통계 패킷 쉐이퍼 정보를 제공합니다. PS_ADAPTER_STATS 구조체는 PS_COMPONENT_STATS 구조와 함께 사용됩니다.
PS_COMPONENT_STATS PS_COMPONENT_STATS 구조를 사용하면 애플리케이션이 TC 사용 흐름에 대한 통계 정보를 가져올 수 있습니다.
PS_CONFORMER_STATS PS_CONFORMER_STATS 구조체는 특정 흐름에 대한 통계 패킷 쉐이퍼 정보를 제공합니다. PS_CONFORMER_STATS 구조체는 PS_COMPONENT_STATS 구조체와 함께 사용됩니다.
PS_DRRSEQ_STATS PS_DRRSEQ_STATS 구조는 NIC(네트워크 인터페이스 카드) 및 패킷 시퀀서 및 패킷 쉐이퍼 통계를 제공합니다. PS_DRRSEQ_STATS 구조체는 PS_COMPONENT_STATS 구조와 함께 사용됩니다.
PS_FLOW_STATS PS_FLOW_STATS 구조체는 특정 흐름에 대한 통계 패킷 쉐이퍼 정보를 제공합니다. PS_FLOW_STATS 구조체는 PS_COMPONENT_STATS 구조와 함께 사용됩니다.
PS_SHAPER_STATS PS_SHAPER_STATS 구조는 컴퓨터의 패킷 쉐이퍼 구성 요소에 대한 통계 패킷 쉐이퍼 정보를 제공합니다. PS_SHAPER_STATS 구조체는 PS_COMPONENT_STATS 구조체와 함께 사용됩니다.

Last updated on 2025. 10. 08.

ntdef.h 헤더

아티클2023. 08. 26.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 설치 가능한 파일 시스템 DDI 참조
- 커널 모드 드라이버 참조

ntdef.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CONTAINING_RECORD CONTAINING_RECORD 매크로는 포함된 구조체 내의 필드의 형식과 주소를 고려하여 구조체의 instance 기본 주소를 반환합니다.
FIELD_OFFSET FIELD_OFFSET 매크로는 알려진 구조체 형식의 명명된 필드의 바이트 오프셋을 반환합니다. (FIELD_OFFSET 매크로)
InitializeObjectAttributes InitializeObjectAttributes 매크로는 핸들을 여는 루틴에 대한 개체 핸들의 속성을 지정하는 불투명 OBJECT_ATTRIBUTES 구조를 초기화합니다.

구조체

LIST_ENTRY LIST_ENTRY 구조체는 이중으로 연결된 목록의 항목을 설명하거나 이러한 목록의 헤더 역할을 합니다.
Luid LUID 구조체는 로컬 컴퓨터에서 고유하도록 보장되는 식별자를 지정하는 불투명 구조체입니다. 자세한 내용은 Microsoft Windows SDK 설명서의 LUID에 대한 참조 페이지를 참조하세요.
OBJECT_ATTRIBUTES

OBJECT_ATTRIBUTES 구조체는 개체 및/또는 반환 핸들을 만드는 루틴에 의해 개체 또는 개체 핸들에 적용할 수 있는 특성을 지정합니다.
SINGLE_LIST_ENTRY SINGLE_LIST_ENTRY 구조체는 적절하게 연결된 목록의 항목을 설명하거나 이러한 목록의 헤더 역할을 합니다.
STRING ANSI_STRING 구조체는 ANSI 문자열에 사용되는 계산된 문자열을 정의합니다.
UNICODE_STRING UNICODE_STRING 구조체는 유니코드 문자열을 정의하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ntdsapi.h 헤더

아티클2023. 08. 25.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Active Directory Domain Services
- Active Directory LDS(Lightweight Directory Services)

ntdsapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DsAddSidHistoryA 한 도메인에서 보안 주체의 기본 계정 SID(보안 식별자)를 검색하여 다른 포리스트의 다른 도메인에 있는 보안 주체의 sIDHistory 특성에 추가합니다. (ANSI)
DsAddSidHistoryW 한 도메인에서 보안 주체의 기본 계정 SID(보안 식별자)를 검색하여 다른 포리스트의 다른 도메인에 있는 보안 주체의 sIDHistory 특성에 추가합니다. (유니코드)
DsBindA 도메인 컨트롤러에 바인딩합니다. (ANSI)
DsBindByInstanceA AD LDS 또는 Active Directory instance 명시적으로 바인딩합니다. (ANSI)
DsBindByInstanceW AD LDS 또는 Active Directory instance 명시적으로 바인딩합니다. (유니코드)
DsBindingSetTimeout DsBindingSetTimeout 함수는 지정된 바인딩 핸들을 사용하는 모든 RPC 호출에서 적용되는 시간 제한 값을 설정합니다. 시간 제한 값보다 더 많은 시간이 필요한 RPC 호출이 취소됩니다.
DsBindToISTGA 로컬 컴퓨터의 도메인에서 Inter-Site ISTG(토폴로지 생성기) 역할을 보유하는 컴퓨터에 바인딩합니다. (ANSI)

DsBindToISTGW 로컬 컴퓨터의 도메인에서 Inter-Site ISTG(토폴로지 생성기) 역할을 보유하는 컴퓨터에 바인딩합니다. (유니코드)
DsBindW 도메인 컨트롤러에 바인딩합니다. (유니코드)
DsBindWithCredA 지정된 자격 증명을 사용하여 도메인 컨트롤러에 바인딩합니다. (ANSI)
DsBindWithCredW 지정된 자격 증명을 사용하여 도메인 컨트롤러에 바인딩합니다. (유니코드)
DsBindWithSpnA 상호 인증을 위해 지정된 자격 증명 및 특정 SPN(서비스 사용자 이름)을 사용하여 도메인 컨트롤러에 바인딩합니다. (DsBindWithSpnA)
DsBindWithSpnExA 상호 인증을 위해 지정된 자격 증명 및 특정 SPN(서비스 사용자 이름)을 사용하여 도메인 컨트롤러에 바인딩합니다. (DsBindWithSpnExA)
DsBindWithSpnExW 상호 인증을 위해 지정된 자격 증명 및 특정 SPN(서비스 사용자 이름)을 사용하여 도메인 컨트롤러에 바인딩합니다. (DsBindWithSpnExW)
DsBindWithSpnW 상호 인증을 위해 지정된 자격 증명 및 특정 SPN(서비스 사용자 이름)을 사용하여 도메인 컨트롤러에 바인딩합니다. (DsBindWithSpnW)
DsClientMakeSpnForTargetServerA 인증에 사용할 특정 서버를 식별하는 SPN(서비스 사용자 이름)을 생성합니다. (ANSI)
DsClientMakeSpnForTargetServerW 인증에 사용할 특정 서버를 식별하는 SPN(서비스 사용자 이름)을 생성합니다. (유니코드)
DsCrackNamesA 디렉터리 서비스 개체 이름의 배열을 한 형식에서 다른 형식으로 변환합니다. (ANSI)

DsCrackNamesW 디렉터리 서비스 개체 이름의 배열을 한 형식에서 다른 형식으로 변환합니다. (유니코드)
DsFreeDomainControllerInfoA DsFreeDomainControllerInfo 함수는 도메인의 도메인 컨트롤러에 대한 데이터에 대해 DsGetDomainControllerInfo에서 할당한 메모리를 해제합니다. (ANSI)
DsFreeDomainControllerInfoW DsFreeDomainControllerInfo 함수는 도메인의 도메인 컨트롤러에 대한 데이터에 대해 DsGetDomainControllerInfo에서 할당한 메모리를 해제합니다. (유니코드)
DsFreeNameResultA DS_NAME_RESULT 구조체에서 보유한 메모리를 해제합니다. (ANSI)
DsFreeNameResultW DS_NAME_RESULT 구조체에서 보유한 메모리를 해제합니다. (유니코드)
DsFreePasswordCredentials DsMakePasswordCredentials 함수에 의해 자격 증명 구조에 할당된 메모리를 해제합니다.
DsFreeSchemaGuidMapA DsMapSchemaGuids 함수가 DS_SCHEMA_GUID_MAP 구조에 할당한 메모리를 해제합니다. (ANSI)
DsFreeSchemaGuidMapW DsMapSchemaGuids 함수가 DS_SCHEMA_GUID_MAP 구조에 할당한 메모리를 해제합니다. (유니코드)
DsFreeSpnArrayA DsGetSpn 함수에서 반환된 배열을 해제합니다. (ANSI)
DsFreeSpnArrayW DsGetSpn 함수에서 반환된 배열을 해제합니다. (유니코드)
DsGetDomainControllerInfoA 도메인의 도메인 컨트롤러에 대한 데이터를 검색합니다. (ANSI)
DsGetDomainControllerInfoW

도메인의 도메인 컨트롤러에 대한 데이터를 검색합니다. (유니코드)
DsGetSpnA DsGetSpn 함수는 하나 이상의 SPN(서비스 사용자 이름)의 배열을 생성합니다. 배열의 각 이름은 서비스의 instance 식별합니다. 이러한 SPN은 DsWriteAccountSpn 함수를 사용하여 DS(디렉터리 서비스)에 등록할 수 있습니다. (ANSI)
DsGetSpnW DsGetSpn 함수는 하나 이상의 SPN(서비스 사용자 이름)의 배열을 생성합니다. 배열의 각 이름은 서비스의 instance 식별합니다. 이러한 SPN은 DsWriteAccountSpn 함수를 사용하여 DS(디렉터리 서비스)에 등록할 수 있습니다. (유니코드)
DsInheritSecurityIdentityA SrcPrincipal의 objectSid 및 sidHistory 특성을 DstPrincipal의 sidHistory에 추가한 다음 SrcPrincipal을 모두 단일 트랜잭션에서 삭제합니다. (ANSI)
DsInheritSecurityIdentityW SrcPrincipal의 objectSid 및 sidHistory 특성을 DstPrincipal의 sidHistory에 추가한 다음 SrcPrincipal을 모두 단일 트랜잭션에서 삭제합니다. (유니코드)
DsListDomainsInSiteA 사이트의 모든 도메인을 나열합니다. (ANSI)
DsListDomainsInSiteW 사이트의 모든 도메인을 나열합니다. (유니코드)
DsListInfoForServerA DsListInfoForServer 함수는 서버에 대한 기타 데이터를 나열합니다. (ANSI)
DsListInfoForServerW DsListInfoForServer 함수는 서버에 대한 기타 데이터를 나열합니다. (유니코드)
DsListRolesA DsListRoles 함수는 서버에서 인식하는 역할을 나열합니다. (ANSI)
DsListRolesW DsListRoles 함수는 서버에서 인식하는 역할을 나열합니다. (유니코드)
DsListServersForDomainInSiteA

사이트의 도메인에 있는 모든 서버를 나열합니다. (ANSI)
DsListServersForDomainInSiteW 사이트의 도메인에 있는 모든 서버를 나열합니다. (유니코드)
DsListServersInSiteA 사이트의 모든 서버를 나열합니다. (ANSI)
DsListServersInSiteW 사이트의 모든 서버를 나열합니다. (유니코드)
DsListSitesA 엔터프라이즈 포리스트의 모든 사이트를 나열합니다. (ANSI)
DsListSitesW 엔터프라이즈 포리스트의 모든 사이트를 나열합니다. (유니코드)
DsMakePasswordCredentialsA DsBindWithCred 함수에 사용하기에 적합한 자격 증명 핸들을 생성합니다. (ANSI)
DsMakePasswordCredentialsW DsBindWithCred 함수에 사용하기에 적합한 자격 증명 핸들을 생성합니다. (유니코드)
DsMapSchemaGuidsA 디렉터리 서비스 스키마 개체의 GUID를 표시 이름으로 변환합니다. (ANSI)
DsMapSchemaGuidsW 디렉터리 서비스 스키마 개체의 GUID를 표시 이름으로 변환합니다. (유니코드)
DsQuerySitesByCostA 한 사이트와 하나 이상의 다른 사이트 간의 통신 비용을 가져옵니다. (ANSI)
DsQuerySitesByCostW 한 사이트와 하나 이상의 다른 사이트 간의 통신 비용을 가져옵니다. (유니코드)
DsQuerySitesFree DsQuerySitesByCost 함수에 의해 할당된 메모리를 해제합니다.

DsRemoveDsDomainA 디렉터리 서비스의 전역 영역에서 도메인 명명 컨텍스트의 모든 추적을 제거합니다. (ANSI)
DsRemoveDsDomainW 디렉터리 서비스의 전역 영역에서 도메인 명명 컨텍스트의 모든 추적을 제거합니다. (유니코드)
DsRemoveDsServerA DsRemoveDsServer 함수는 디렉터리 서비스의 전역 영역에서 DSA(디렉터리 서비스 에이전트)의 모든 추적을 제거합니다. (ANSI)
DsRemoveDsServerW DsRemoveDsServer 함수는 디렉터리 서비스의 전역 영역에서 DSA(디렉터리 서비스 에이전트)의 모든 추적을 제거합니다. (유니코드)
DsReplicaAddA 대상 명명 컨텍스트에 복제 원본 참조를 추가합니다. (ANSI)
DsReplicaAddW 대상 명명 컨텍스트에 복제 원본 참조를 추가합니다. (유니코드)
DsReplicaConsistencyCheck KCC(지식 일관성 검사기)를 호출하여 복제 토폴로지를 확인합니다.
DsReplicaDelA NC(대상 명명 컨텍스트)에서 복제 원본 참조를 제거합니다. (ANSI)
DsReplicaDelW NC(대상 명명 컨텍스트)에서 복제 원본 참조를 제거합니다. (유니코드)
DsReplicaFreeInfo DsReplicaGetInfo 또는 DsReplicaGetInfo2 함수에 의해 할당된 복제 상태 데이터 구조를 해제합니다.
DsReplicaGetInfo2W 디렉터리 서비스에서 복제 상태 데이터를 검색합니다. 이 함수를 사용하면 검색할 항목이 1000개 이상인 경우 결과의 페이징을 허용합니다.

DsReplicaGetInfoW 디렉터리 서비스에서 복제 상태 데이터를 검색합니다.
DsReplicaModifyA 대상 명명 컨텍스트에 대한 기존 복제 원본 참조를 수정합니다. (ANSI)
DsReplicaModifyW 대상 명명 컨텍스트에 대한 기존 복제 원본 참조를 수정합니다. (유니코드)
DsReplicaSyncA NC(대상 명명 컨텍스트)를 원본 중 하나와 동기화합니다. (ANSI)
DsReplicaSyncAllA 필요에 따라 전이적 복제를 사용하여 서버를 다른 모든 서버와 동기화합니다. (ANSI)
DsReplicaSyncAllW 필요에 따라 전이적 복제를 사용하여 서버를 다른 모든 서버와 동기화합니다. (유니코드)
DsReplicaSyncW NC(대상 명명 컨텍스트)를 원본 중 하나와 동기화합니다. (유니코드)
DsReplicaUpdateRefsA 원본 명명 컨텍스트에서 대상에 대한 복제 참조를 추가하거나 제거합니다. (ANSI)
DsReplicaUpdateRefsW 원본 명명 컨텍스트에서 대상에 대한 복제 참조를 추가하거나 제거합니다. (유니코드)
DsReplicaVerifyObjectsA 원본을 사용하여 명명 컨텍스트에 대한 모든 개체를 확인합니다. (ANSI)
DsReplicaVerifyObjectsW 원본을 사용하여 명명 컨텍스트에 대한 모든 개체를 확인합니다. (유니코드)
DsServerRegisterSpnA DsServerRegisterSpn 함수는 호스트 기반 서비스에 대해 두 개의 SPN을 구성합니다. (ANSI)
DsServerRegisterSpnW

DsServerRegisterSpn 함수는 호스트 기반 서비스에 대해 두 개의 SPN을 구성합니다. (유니코드)
DsUnBindA DsUnBind 함수는 도메인 컨트롤러가 있는 RPC 세션을 찾아 DS(디렉터리 서비스)에 대한 핸들을 바인딩 해제합니다. (ANSI)
DsUnBindW DsUnBind 함수는 도메인 컨트롤러가 있는 RPC 세션을 찾아 DS(디렉터리 서비스)에 대한 핸들을 바인딩 해제합니다. (유니코드)
DsWriteAccountSpnA Active Directory Domain Services 지정된 사용자 또는 컴퓨터 계정 개체의 servicePrincipalName 특성에 SPN(서비스 사용자 이름) 배열을 씁니다. (ANSI)
DsWriteAccountSpnW Active Directory Domain Services 지정된 사용자 또는 컴퓨터 계정 개체의 servicePrincipalName 특성에 SPN(서비스 사용자 이름) 배열을 씁니다. (유니코드)

구조체

DS_DOMAIN_CONTROLLER_INFO_1A DS_DOMAIN_CONTROLLER_INFO_1 구조에는 도메인 컨트롤러에 대한 데이터가 포함됩니다. 이 구조체는 DsGetDomainControllerInfo 함수에 의해 반환됩니다. (ANSI)
DS_DOMAIN_CONTROLLER_INFO_1W DS_DOMAIN_CONTROLLER_INFO_1 구조에는 도메인 컨트롤러에 대한 데이터가 포함됩니다. 이 구조체는 DsGetDomainControllerInfo 함수에 의해 반환됩니다. (유니코드)
DS_DOMAIN_CONTROLLER_INFO_2A DS_DOMAIN_CONTROLLER_INFO_2 구조에는 도메인 컨트롤러에 대한 데이터가 포함됩니다. 이 구조체는 DsGetDomainControllerInfo 함수에 의해 반환됩니다. (ANSI)
DS_DOMAIN_CONTROLLER_INFO_2W DS_DOMAIN_CONTROLLER_INFO_2 구조에는 도메인 컨트롤러에 대한 데이터가 포함됩니다. 이 구조체는 DsGetDomainControllerInfo 함수에 의해 반환됩니다. (유니코드)

DS_DOMAIN_CONTROLLER_INFO_3A DS_DOMAIN_CONTROLLER_INFO_3 구조에는 도메인 컨트롤러에 대한 데이터가 포함됩니다. 이 구조체는 DsGetDomainControllerInfo 함수에 의해 반환됩니다. (ANSI)
DS_DOMAIN_CONTROLLER_INFO_3W DS_DOMAIN_CONTROLLER_INFO_3 구조에는 도메인 컨트롤러에 대한 데이터가 포함됩니다. 이 구조체는 DsGetDomainControllerInfo 함수에 의해 반환됩니다. (유니코드)
DS_NAME_RESULT_ITEMA DS_NAME_RESULT_ITEM 구조에는 연결된 오류 및 도메인 데이터와 함께 DsCrackNames 함수로 변환된 이름이 포함됩니다. (ANSI)
DS_NAME_RESULT_ITEMW DS_NAME_RESULT_ITEM 구조에는 연결된 오류 및 도메인 데이터와 함께 DsCrackNames 함수로 변환된 이름이 포함됩니다. (유니코드)
DS_NAME_RESULTA DS_NAME_RESULT 구조체는 DsCrackNames 함수와 함께 함수로 변환된 이름을 포함하는 데 사용됩니다. (ANSI)
DS_NAME_RESULTW DS_NAME_RESULT 구조체는 DsCrackNames 함수와 함께 함수로 변환된 이름을 포함하는 데 사용됩니다. (유니코드)
DS_REPL_ATTR_META_DATA DS_REPL_ATTR_META_DATA 구조체는 DsReplicaGetInfo 및 DsReplicaGetInfo2 함수와 함께 개체 특성에 대한 복제 상태 데이터를 포함하는 데 사용됩니다.
DS_REPL_ATTR_META_DATA_2 DS_REPL_ATTR_META_DATA_2 구조체는 DsReplicaGetInfo 및 DsReplicaGetInfo2 함수와 함께 개체 특성에 대한 복제 상태 데이터를 포함하는 데 사용됩니다.
DS_REPL_ATTR_META_DATA_BLOB DS_REPL_ATTR_META_DATA_BLOB 구조체는 개체 특성에 대한 복제 상태 데이터를 포함하는 데 사용됩니다.
DS_REPL_ATTR_VALUE_META_DATA DS_REPL_ATTR_VALUE_META_DATA 구조는 DsReplicaGetInfo2 함수와 함께 특성 값 컬렉션에 대한 메타데이터를 제공하는 데 사용됩니다.

DS_REPL_ATTR_VALUE_META_DATA_2 DsReplicaGetInfo2 함수와 함께 특성 값 컬렉션에 대한 메타데이터를 제공하는 데 사용됩니다.
DS_REPL_ATTR_VALUE_META_DATA_EXT 특성 복제 값 컬렉션에 대한 메타데이터를 제공합니다.
DS_REPL_CURSOR DS_REPL_CURSOR 구조에는 DsReplicaGetInfo 및 DsReplicaGetInfo2 함수에서 반환된 지정된 명명 컨텍스트의 모든 복제본과 관련된 인바운드 복제 상태 데이터가 포함됩니다.
DS_REPL_CURSOR_2 DS_REPL_CURSOR_2 구조에는 DsReplicaGetInfo2 함수에서 반환된 지정된 명명 컨텍스트의 모든 복제본과 관련된 인바운드 복제 상태 데이터가 포함됩니다. 이 구조체는 DS_REPL_CURSOR 구조체의 향상된 버전입니다.
DS_REPL_CURSOR_3W DS_REPL_CURSOR_3 구조에는 DsReplicaGetInfo2 함수에서 반환된 지정된 명명 컨텍스트의 모든 복제본과 관련된 인바운드 복제 상태 데이터가 포함됩니다.
DS_REPL_CURSOR_BLOB DS_REPL_CURSOR_BLOB 구조에는 지정된 명명 컨텍스트의 모든 복제본과 관련된 인바운드 복제 상태 데이터가 포함됩니다.
DS_REPL_CURSORS DS_REPL_CURSORS 구조는 DsReplicaGetInfo 및 DsReplicaGetInfo2 함수와 함께 사용하여 지정된 명명 컨텍스트의 모든 복제본과 관련된 복제 상태 데이터를 제공합니다.
DS_REPL_CURSORS_2 DS_REPL_CURSORS_2 구조는 DsReplicaGetInfo2 함수와 함께 사용하여 지정된 명명 컨텍스트의 모든 복제본과 관련된 복제 상태 데이터를 제공합니다.
DS_REPL_CURSORS_3W DS_REPL_CURSORS_3 구조는 DsReplicaGetInfo2 함수와 함께 사용하여 지정된 명명 컨텍스트의 모든 복제본과 관련된 복제 상태 데이터를 제공합니다.
DS_REPL_KCC_DSA_FAILURESW DS_REPL_KCC_DSA_FAILURES 구조에는 DsReplicaGetInfo 및 DsReplicaGetInfo2 함수에서 반환한 대로 인바운드 복제 파트너와 관련된 복제 상태 데이터가 포함된 DS_REPL_KCC_DSA_FAILURE 구조의 배열이 포함되어 있습니다.

DS_REPL_KCC_DSA_FAILUREW DS_REPL_KCC_DSA_FAILURE 구조에는 DsReplicaGetInfo 및 DsReplicaGetInfo2 함수에서 반환한 특정 인바운드 복제 파트너에 대한 복제 상태 데이터가 포함됩니다.
DS_REPL_KCC_DSA_FAILUREW_BLOB 특정 인바운드 복제 파트너와 관련된 복제 상태 데이터를 포함합니다.
DS_REPL_NEIGHBORSW DS_REPL_NEIGHBORS 구조는 DsReplicaGetInfo 및 DsReplicaGetInfo2 함수와 함께 사용하여 컨텍스트 및 원본 서버 쌍 명명을 위한 인바운드 복제 상태 데이터를 제공합니다.
DS_REPL_NEIGHBORW DsReplicaGetInfo 및 DsReplicaGetInfo2 함수에서 반환된 특정 명명 컨텍스트 및 원본 서버 쌍에 대한 인바운드 복제 상태 데이터를 포함합니다.
DS_REPL_NEIGHBORW_BLOB 특정 명명 컨텍스트 및 원본 서버 쌍에 대한 인바운드 복제 상태 데이터를 포함합니다.
DS_REPL_OBJ_META_DATA DS_REPL_OBJ_META_DATA 구조체에는 DS_REPL_ATTR_META_DATA 구조체의 배열이 포함되어 있습니다. 이러한 구조에는 지정된 개체에 대한 과거 및 현재 특성에 대한 복제 상태 데이터가 포함됩니다.
DS_REPL_OBJ_META_DATA_2 DS_REPL_OBJ_META_DATA_2 구조체에는 DsReplicaGetInfo2 함수에서 반환한 대로 지정된 개체에 대한 특성(과거 및 현재)에 대한 복제 상태 데이터가 포함된 DS_REPL_ATTR_META_DATA_2 구조체의 배열이 포함되어 있습니다.
DS_REPL_OPW DS_REPL_OP 구조는 DsReplicaGetInfo 또는 DsReplicaGetInfo2 함수에서 반환된 대로 현재 실행 중이거나 실행 보류 중인 복제 작업을 설명합니다.
DS_REPL_OPW_BLOB DS_REPL_OPW_BLOB 구조는 현재 실행 중이거나 실행 보류 중인 복제 작업을 설명합니다.
DS_REPL_PENDING_OPSW DsReplicaGetInfo 및 DsReplicaGetInfo2 함수에서 반환한 대로 현재 실행되고 큐에 대기 중인 복제 작업을 설명하는 DS_REPL_OP 구조의 배열을 포함합니다.

DS_REPL_QUEUE_STATISTICSW 복제 큐 통계를 포함하는 데 사용됩니다.
DS_REPL_VALUE_META_DATA 특성 값 복제 메타데이터를 포함하기 위해 DS_REPL_ATTR_VALUE_META_DATA 구조와 함께 사용됩니다.
DS_REPL_VALUE_META_DATA_2 특성 값 복제 메타데이터를 포함하기 위해 DS_REPL_ATTR_VALUE_META_DATA_2 구조와 함께 사용됩니다.
DS_REPL_VALUE_META_DATA_BLOB 특성 값 복제 메타데이터를 포함하는 데 사용됩니다.
DS_REPL_VALUE_META_DATA_BLOB_EXT 특성 값 복제 메타데이터를 포함합니다.
DS_REPL_VALUE_META_DATA_EXT DS_REPL_ATTR_VALUE_META_DATA_EXT 구조에 대한 특성 복제 메타 데이터를 포함합니다.
DS_REPSYNCALL_ERRINFOA DS_REPSYNCALL_ERRINFO 구조체는 복제 중에 DsReplicaSyncAll 함수에 의해 생성된 오류를 포함하기 위해 DS_REPSYNCALL_UPDATE 구조체와 함께 사용됩니다. (ANSI)
DS_REPSYNCALL_ERRINFOW DS_REPSYNCALL_ERRINFO 구조체는 복제 중에 DsReplicaSyncAll 함수에 의해 생성된 오류를 포함하기 위해 DS_REPSYNCALL_UPDATE 구조체와 함께 사용됩니다. (유니코드)
DS_REPSYNCALL_SYNCA DS_REPSYNCALL_SYNC 구조는 DsReplicaSyncAll 함수를 통해 원본 서버와 대상 서버 간에 수행되는 단일 복제 작업을 식별합니다. (ANSI)
DS_REPSYNCALL_SYNCW DS_REPSYNCALL_SYNC 구조는 DsReplicaSyncAll 함수를 통해 원본 서버와 대상 서버 간에 수행되는 단일 복제 작업을 식별합니다. (유니코드)
DS_REPSYNCALL_UPDATEA

DS_REPSYNCALL_UPDATE 구조에는 DsReplicaSyncAll 함수에서 수행하는 복제에 대한 상태 데이터가 포함됩니다. (ANSI)
DS_REPSYNCALL_UPDATEW DS_REPSYNCALL_UPDATE 구조에는 DsReplicaSyncAll 함수에서 수행하는 복제에 대한 상태 데이터가 포함됩니다. (유니코드)
DS_SCHEMA_GUID_MAPA DsMapSchemaGuids 호출 결과를 포함합니다. (ANSI)
DS_SCHEMA_GUID_MAPW DsMapSchemaGuids 호출 결과를 포함합니다. (유니코드)
DS_SITE_COST_INFO DS_SITE_COST_INFO 구조는 DsQuerySitesByCost 함수와 함께 통신 비용 데이터를 포함하는 데 사용됩니다.

열거형

DS_KCC_TASKID KCC(지식 일관성 검사기)가 실행할 수 있는 작업을 지정합니다.
DS_NAME_ERROR DS_NAME_ERROR 열거형은 DS_NAME_RESULT_ITEM 구조체의 상태 멤버가 반환하는 오류를 정의합니다. DsCrackNames 함수에서 이름을 변환하는 동안 발생할 수 있는 잠재적 오류입니다.
DS_NAME_FLAGS DS_NAME_FLAGS 열거형은 이름 구문을 해독하는 방법을 정의하는 데 사용됩니다. 이러한 플래그는 DsCrackNames 함수에서 사용됩니다.
DS_NAME_FORMAT DS_NAME_FORMAT 열거형은 DsCrackNames 함수의 입력 및 출력 이름에 사용할 형식을 제공합니다.
DS_REPL_INFO_TYPE

DS_REPL_INFO_TYPE 열거형은 DsReplicaGetInfo 및 DsReplicaGetInfo2 함수와 함께 사용하여 검색할 복제 데이터의 형식을 지정합니다.
DS_REPL_OP_TYPE 복제 큐의 지정된 항목이 나타내는 복제 작업의 유형을 나타내는 데 사용됩니다.
DS_REPSYNCALL_ERROR DS_REPSYNCALL_ERROR 열거형은 복제 프로세스에서 오류가 발생한 위치를 나타내기 위해 DS_REPSYNCALL_ERRINFO 구조와 함께 사용됩니다.
DS_REPSYNCALL_EVENT DS_REPSYNCALL_EVENT 열거형은 DS_REPSYNCALL_UPDATE 구조가 나타내는 이벤트를 정의하기 위해 DS_REPSYNCALL_UPDATE 구조체와 함께 사용됩니다.
DS_SPN_NAME_TYPE DS_SPN_NAME_TYPE 열거형은 DsGetSPN 함수에서 SPN 작성 형식을 식별하는 데 사용됩니다.
DS_SPN_WRITE_OP DS_SPN_WRITE_OP 열거형은 DsWriteAccountSpn 함수에서 수행해야 하는 쓰기 작업의 유형을 식별합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ntenclv.h 헤더

아티클2024. 02. 05.

이 헤더는 System Services 및 Secure Enclave에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스
- 보안 Enclave(신뢰할 수 있는 실행)

ntenclv.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

테이블 확장

ENCLAVE_IDENTITY enclave의 기본 모듈의 ID를 설명합니다.
ENCLAVE_INFORMATION 현재 실행 중인 enclave에 대한 정보를 포함합니다.
VBS_ENCLAVE_REPORT EnclaveGetAttestationReport 함수를 호출하여 생성된 보고서에 포함된 서명된 문의 형식을 설명합니다.
VBS_ENCLAVE_REPORT_MODULE enclave에 대해 로드된 모듈에 대해 설명합니다.
VBS_ENCLAVE_REPORT_PKG_HEADER EnclaveGetAttestationReport 함수를 호출하여 생성된 보고서의 내용을 설명합니다.
VBS_ENCLAVE_REPORT_VARDATA_HEADER EnclaveGetAttestationReport 함수가 생성하는 보고서에 포함된 변수 데이터 블록의 형식을 설명합니다.

열거형

테이블 확장

ENCLAVE_SEALING_IDENTITY_POLICY 다른 Enclave가 Enclave에 대해 EnclaveSealData를 호출하여 데이터의 봉인을 해제하는 Enclave와 관련되어야 하는 방법을 지정하는 값을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

ntioring_x.h 헤더

아티클2023. 08. 25.

I/O 링 작업에 대한 형식을 제공합니다.

이 헤더는 데이터 액세스 및 스토리지에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지

ntioring_x.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

IORING_BUFFER_INFO I/O 링에 등록할 수 있는 데이터 버퍼를 나타냅니다.
IORING_REGISTERED_BUFFER BuildIoRingRegisterBuffers를 호출하여 I/O 링에 등록된 버퍼를 나타냅니다.

열거형

IORING_FEATURE_FLAGS I/O 링 API 버전에 대한 기능 지원을 나타냅니다.
IORING_OP_CODE I/O 링 작업의 형식을 지정합니다.
IORING_VERSION IORING API 버전을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ntlsa.h 헤더

아티클2023. 08. 25.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

ntlsa.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

LsaGetAppliedCAPIDs 특정 컴퓨터에 적용된 모든 CAP의 CAPID(중앙 액세스 정책) 식별자 배열을 반환합니다.
LsaLookupPrivilegeValue LSA(로컬 보안 기관)에서 지정된 권한 이름을 나타내는 데 사용하는 LUID(로컬 고유 식별자)를 검색합니다.
LsaQueryCAPs 지정된 ID에 대한 중앙 액세스 정책(CAP)을 반환합니다.

콜백 함수

LSA_FREE_LSA_HEAP AllocateLsaHeap에서 이전에 할당한 힙 메모리의 할당을 취소합니다.

구조체

CENTRAL_ACCESS_POLICY 중앙 액세스 정책 항목 집합을 포함하는 중앙 액세스 정책을 나타냅니다.

CENTRAL_ACCESS_POLICY_ENTRY 보안 설명자 및 준비된 보안 설명자 목록을 포함하는 중앙 액세스 정책 항목을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ntmsapi.h 헤더

아티클2023. 03. 08.

이 헤더는 Data Access 및 Storage에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지

ntmsapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AccessNtmsLibraryDoor AccessNtmsLibraryDoor 함수는 지정된 라이브러리의 문을 잠금 해제합니다. 라이브러리가 사용 중인 경우 RSM은 요청을 큐에 대기시키고 성공적으로 반환합니다.
AddNtmsMediaType AddNtmsMediaType 함수는 현재 라이브러리 개체에 관계가 없는 경우 지정된 미디어 형식을 지정된 라이브러리에 추가합니다. 그런 다음, 시스템 미디어 풀이 없는 경우 이 함수는 시스템 미디어 풀을 만듭니다.
AllocateNtmsMedia AllocateNtmsMedia 함수는 사용 가능한 미디어를 할당합니다.
BeginNtmsDeviceChangeDetection BeginNtmsDeviceChangeDetection 함수를 사용하면 애플리케이션에서 디바이스 변경 검색 세션을 시작할 수 있습니다.
CancelNtmsLibraryRequest CancelNtmsLibraryRequest 함수는 CleanNtmsDrive 함수 호출과 같은 미해결 RSM 요청을 취소합니다. 라이브러리가 사용 중인 경우 RSM은 취소를 큐에 대기시키고 성공을 반환합니다.
CancelNtmsOperatorRequest CancelNtmsOperatorRequest 함수는 지정된 RSM 연산자 요청을 취소합니다.
ChangeNtmsMediaType ChangeNtmsMediaType 함수는 지정된 PMID를 지정된 대상 미디어 풀로 이동하고 PMID의 미디어 형식 식별자를 대상 미디어 풀의 미디어 형식으로 설정합니다.

CleanNtmsDrive CleanNtmsDrive 함수는 청소를 위해 지정된 드라이브에 대한 정리 요청을 큐에 대기합니다.
CloseNtmsNotification CloseNtmsNotification 함수는 지정된 열린 알림 채널을 닫습니다.
CloseNtmsSession CloseNtmsSession 함수는 지정된 RSM 세션을 닫습니다.
CreateNtmsMediaA CreateNtmsMedia 함수는 오프라인 미디어의 새 부분에 대한 PMID 및 측면(또는 측면)을 만듭니다. 미디어는 lpPhysicalMedia에 지정된 미디어 풀에 배치됩니다. (ANSI)
CreateNtmsMediaPool CreateNtmsMediaPool 함수는 새 애플리케이션 미디어 풀을 만듭니다. (CreateNtmsMediaPool)
CreateNtmsMediaPoolA CreateNtmsMediaPool 함수는 새 애플리케이션 미디어 풀을 만듭니다. (CreateNtmsMediaPoolA)
CreateNtmsMediaPoolW CreateNtmsMediaPoolW(유니코드) 함수는 새 애플리케이션 미디어 풀을 만듭니다. (CreateNtmsMediaPoolW)
CreateNtmsMediaW CreateNtmsMedia 함수는 오프라인 미디어의 새 부분에 대한 PMID 및 측면(또는 측면)을 만듭니다. 미디어는 lpPhysicalMedia에 지정된 미디어 풀에 배치됩니다. (유니코드)
DeallocateNtmsMedia DeallocateNtmsMedia 함수는 지정된 논리 미디어와 연결된 쪽의 할당을 취소합니다.
DecommissionNtmsMedia DecommissionNtmsMedia 함수는 사용 가능한 상태에서 서비스 해제됨 상태로 한 쪽을 이동합니다.
DeleteNtmsDrive DeleteNtmsDrive 함수는 RSM 데이터베이스에서 드라이브를 삭제합니다. 드라이브에는 NTMS_NOT_PRESENT dwOperationalState가 있어야 합니다.

DeleteNtmsLibrary DeleteNtmsLibrary 함수는 라이브러리 및 라이브러리에 포함된 모든 디바이스를 RSM 데이터베이스에서 삭제합니다. 라이브러리의 모든 미디어가 오프라인 라이브러리로 이동됩니다.
DeleteNtmsMedia DeleteNtmsMedia 함수는 데이터베이스에서 지정된 미디어에 대한 모든 참조를 제거하여 RSM에서 오프라인 미디어의 물리적 부분을 삭제합니다.
DeleteNtmsMediaPool DeleteNtmsMediaPool 함수는 지정된 애플리케이션 미디어 풀을 삭제합니다.
DeleteNtmsMediaType DeleteNtmsMediaType 함수는 지정된 미디어 형식의 물리적 미디어 개체가 라이브러리에 포함되어 있지 않은 경우 지정된 라이브러리에서 지정된 미디어 형식 관계를 삭제합니다.
DeleteNtmsRequests DeleteNtmsRequests 함수는 RSM 데이터베이스에서 요청 또는 요청 목록을 삭제합니다.
DisableNtmsObject DisableNtmsObject 함수는 지정된 RSM 개체를 사용하지 않도록 설정합니다.
DismountNtmsDrive DismountNtmsDrive 함수는 지정된 드라이브의 미디어를 스토리지 슬롯으로 이동하는 명령을 큐에 넣습니다. 이 함수는 MountNtmsMedia 함수와 페어링되어야 합니다.
DismountNtmsMedia DismountNtmsMedia 함수는 드라이브의 지정된 미디어를 스토리지로 이동하는 명령을 큐에 대기합니다. 이 함수는 MountNtmsMedia 함수와 페어링되어야 합니다.
EjectDiskFromSADriveA EjectDiskFromSADrive 함수는 독립 실행형 이동식 드라이브에 있는 미디어를 배출합니다. (ANSI)
EjectDiskFromSADriveW EjectDiskFromSADrive 함수는 독립 실행형 이동식 드라이브에 있는 미디어를 배출합니다. (유니코드)
EjectNtmsCleaner EjectNtmsCleaner 함수는 현재 예약된 클리너 슬롯에서 청소 카트리지를 배출합니다.

EjectNtmsMedia EjectNtmsMedia 함수는 현재 라이브러리의 포트에서 지정된 매체를 배출합니다. 라이브러리가 사용 중인 경우 RSM은 EjectNtmsMedia를 큐에 대기시키고 성공을 반환합니다.
EnableNtmsObject EnableNtmsObject 함수는 지정된 개체를 사용하도록 설정합니다.
EndNtmsDeviceChangeDetection EndNtmsDeviceChangeDetection 함수는 SetNtmsDeviceChangeDetection 함수를 사용하여 지정된 대상 디바이스에 대한 디바이스 변경 검색을 종료하고 변경 검색 핸들을 닫습니다.
EnumerateNtmsObject EnumerateNtmsObject 함수는 lpContainerId 매개 변수에 포함된 RSM 개체를 열거합니다.
ExportNtmsDatabase ExportNtmsDatabase 함수는 RSM 데이터베이스 디렉터리에 일관된 데이터베이스 파일 집합을 만듭니다.
GetNtmsMediaPoolNameA GetNtmsMediaPoolName 함수는 지정된 미디어 풀의 전체 이름 계층 구조를 검색합니다. (ANSI)
GetNtmsMediaPoolNameW GetNtmsMediaPoolName 함수는 지정된 미디어 풀의 전체 이름 계층 구조를 검색합니다. (유니코드)
GetNtmsObjectAttributeA GetNtmsObjectAttribute 함수는 지정된 RSM 개체에서 확장 특성(명명된 프라이빗 데이터)을 검색합니다. (ANSI)
GetNtmsObjectAttributeW GetNtmsObjectAttribute 함수는 지정된 RSM 개체에서 확장 특성(명명된 프라이빗 데이터)을 검색합니다. (유니코드)
GetNtmsObjectInformation GetNtmsObjectInformation 함수는 지정된 개체에 대한 개체의 정보 구조를 반환합니다. (GetNtmsObjectInformation)

GetNtmsObjectInformationA GetNtmsObjectInformation 함수는 지정된 개체에 대한 개체의 정보 구조를 반환합니다. (GetNtmsObjectInformationA)
GetNtmsObjectInformationW GetNtmsObjectInformationW(유니코드) 함수는 지정된 개체에 대한 개체의 정보 구조를 반환합니다. (GetNtmsObjectInformationW)
GetNtmsObjectSecurity GetNtmsObjectSecurity 함수는 지정된 RSM 개체에 대한 보안 설명자를 읽습니다.
GetNtmsRequestOrder GetNtmsRequestOrder 함수는 지정된 요청이 라이브러리 큐에서 처리되는 순서를 가져옵니다.
GetNtmsUIOptionsA GetNtmsUIOptions 함수는 지정된 형식의 사용자 인터페이스가 지정된 개체에 대해 전달되는 컴퓨터 이름 목록을 가져옵니다. (ANSI)
GetNtmsUIOptionsW GetNtmsUIOptions 함수는 지정된 형식의 사용자 인터페이스가 지정된 개체에 대해 전달되는 컴퓨터 이름 목록을 가져옵니다. (유니코드)
GetVolumesFromDriveA GetVolumesFromDrive 함수는 지정된 이동식 스토리지 미디어 드라이브에 대한 볼륨 및 드라이브 문자를 검색합니다. (ANSI)
GetVolumesFromDriveW GetVolumesFromDrive 함수는 지정된 이동식 스토리지 미디어 드라이브에 대한 볼륨 및 드라이브 문자를 검색합니다. (유니코드)
IdentifyNtmsSlot IdentifiNtmsSlot 함수는 라이브러리의 지정된 슬롯에 있는 미디어를 식별합니다. 식별이 완료되면 명령이 반환됩니다.
ImportNtmsDatabase ImportNtmsDatabase 함수를 사용하면 RSM이 다음에 RSM을 다시 시작할 때 데이터베이스 내보내기 디렉터리에서 데이터베이스 파일을 가져옵니다.

InjectNtmsCleaner InjectNtmsCleaner 함수를 사용하면 지정된 라이브러리 단위에 클리너 카트리지를 삽입할 수 있습니다.
InjectNtmsMedia InjectNtmsMedia 함수를 사용하면 지정된 라이브러리의 포트에 미디어를 삽입할 수 있습니다. 라이브러리가 사용 중인 경우 RSM은 InjectNtmsMedia를 큐에 대기시키고 성공을 반환합니다.
InventoryNtmsLibrary InventoryNtmsLibrary 함수는 지정된 라이브러리의 인벤토리를 큐에 대기합니다. 라이브러리가 사용 중인 경우 RSM은 InventoryNtmsLibrary를 큐에 대기시키고 성공을 반환합니다.
MountNtmsMedia MountNtmsMedia 함수는 하나 이상의 미디어를 동기적으로 탑재합니다.
MoveToNtmsMediaPool MoveToNtmsMediaPool 함수는 지정된 미디어를 현재 미디어 풀에서 지정된 미디어 풀로 이동합니다.
OpenNtmsNotification OpenNtmsNotification 함수는 지정된 형식의 개체에 대한 RSM 개체 변경 알림을 수신하는 채널을 엽니다.
OpenNtmsSessionA OpenNtmsSession 함수는 RSM 서버를 사용하여 세션을 설정합니다. (ANSI)
OpenNtmsSessionW OpenNtmsSession 함수는 RSM 서버를 사용하여 세션을 설정합니다. (유니코드)
ReleaseNtmsCleanerSlot ReleaseNtmsCleanerSlot 함수는 정리 카트리지에 대한 기존 슬롯 예약을 제거합니다. 그런 다음 슬롯을 데이터 카트리지에 사용할 수 있습니다.
ReserveNtmsCleanerSlot ReserveNtmsCleanerSlot 함수는 드라이브 클리너 카트리지를 위해 라이브러리 단위에 단일 슬롯을 예약합니다.

SatisfyNtmsOperatorRequest SatisfyNtmsOperatorRequest 함수는 지정된 RSM 연산자 요청을 완료합니다.
SetNtmsDeviceChangeDetection SetNtmsDeviceChangeDetection 함수는 변경 검색을 위해 하나 이상의 대상 디바이스를 설정합니다.
SetNtmsMediaComplete SetNtmsMediaComplete 함수는 논리 미디어의 일부를 완료로 표시합니다.
SetNtmsObjectAttributeA SetNtmsObjectAttribute 함수는 지정된 RSM 개체에 확장 특성(명명된 프라이빗 데이터)을 만듭니다. (ANSI)
SetNtmsObjectAttributeW SetNtmsObjectAttribute 함수는 지정된 RSM 개체에 확장 특성(명명된 프라이빗 데이터)을 만듭니다. (유니코드)
SetNtmsObjectInformation SetNtmsObjectInformation 함수는 지정된 개체의 정보 구조를 변경합니다. (SetNtmsObjectInformation)
SetNtmsObjectInformationA SetNtmsObjectInformation 함수는 지정된 개체의 정보 구조를 변경합니다. (SetNtmsObjectInformationA)
SetNtmsObjectInformationW SetNtmsObjectInformationW(유니코드) 함수는 지정된 개체의 정보 구조를 변경합니다. (SetNtmsObjectInformationW)
SetNtmsObjectSecurity SetNtmsObjectSecurity 함수는 지정된 RSM 개체에 대한 보안 설명자를 씁니다.
SetNtmsRequestOrder SetNtmsRequestOrder 함수는 지정된 요청이 라이브러리 큐에서 처리되는 순서를 설정합니다.

SetNtmsUIOptionsA SetNtmsUIOptions 함수는 지정된 형식의 UI가 지정된 개체에 대해 전달되는 컴퓨터 이름 목록을 수정합니다. (ANSI)
SetNtmsUIOptionsW SetNtmsUIOptions 함수는 지정된 형식의 UI가 지정된 개체에 대해 전달되는 컴퓨터 이름 목록을 수정합니다. (유니코드)
SubmitNtmsOperatorRequestA SubmitNtmsOperatorRequest 함수는 RSM 연산자 요청을 제출합니다. (ANSI)
SubmitNtmsOperatorRequestW SubmitNtmsOperatorRequest 함수는 RSM 연산자 요청을 제출합니다. (유니코드)
SwapNtmsMedia SwapNtmsMedia 함수는 지정된 두 LMID와 연결된 측면을 교환합니다. 지정된 LMID는 동일한 미디어 풀에 있어야 합니다.
UpdateNtmsOmidInfo UpdateNtmsOmidInfo 함수는 새로 할당된 매체에 쓴 직후 레이블 정보로 RSM 데이터베이스를 업데이트합니다.
WaitForNtmsNotification WaitForNtmsNotification 함수는 다음 개체 변경 알림을 기다립니다.
WaitForNtmsOperatorRequest WaitForNtmsOperatorRequest 함수는 지정된 RSM 연산자 요청을 기다립니다.

구조체

NTMS_ALLOCATION_INFORMATION NTMS_ALLOCATION_INFORMATION 구조에는 미디어를 가져온 원본 미디어 풀에 대한 정보가 포함됩니다.

NTMS_CHANGERINFORMATIONA NTMS_CHANGERINFORMATION 구조체는 로봇 체인저 개체와 관련된 속성을 정의합니다. (ANSI)
NTMS_CHANGERINFORMATIONW NTMS_CHANGERINFORMATION 구조체는 로봇 체인저 개체와 관련된 속성을 정의합니다. (유니코드)
NTMS_CHANGERTYPEINFORMATIONA NTMS_CHANGERTYPEINFORMATION 구조는 RSM에서 지원하는 로봇 체인저 유형과 관련된 속성을 정의합니다. (ANSI)
NTMS_CHANGERTYPEINFORMATIONW NTMS_CHANGERTYPEINFORMATION 구조는 RSM에서 지원하는 로봇 체인저 유형과 관련된 속성을 정의합니다. (유니코드)
NTMS_COMPUTERINFORMATION NTMS_COMPUTERINFORMATION 구조는 RSM 서버와 관련된 속성을 정의합니다.
NTMS_DRIVEINFORMATIONA NTMS_DRIVEINFORMATION 구조체는 드라이브 개체와 관련된 속성을 정의합니다. (ANSI)
NTMS_DRIVEINFORMATIONW NTMS_DRIVEINFORMATION 구조체는 드라이브 개체와 관련된 속성을 정의합니다. (유니코드)
NTMS_DRIVETYPEINFORMATIONA NTMS_DRIVETYPEINFORMATION 구조는 RSM에서 지원하는 드라이브 유형과 관련된 속성을 정의합니다. (ANSI)
NTMS_DRIVETYPEINFORMATIONW NTMS_DRIVETYPEINFORMATION 구조는 RSM에서 지원하는 드라이브 유형과 관련된 속성을 정의합니다. (유니코드)
NTMS_IEDOORINFORMATION NTMS_IEDOORINFORMATION 구조체는 insert/eject door 개체와 관련된 속성을 정의합니다.
NTMS_IEPORTINFORMATION NTMS_IEPORTINFORMATION 구조체는 삽입/배출 포트 개체와 관련된 속성을 정의합니다.

NTMS_LIBRARYINFORMATION NTMS_LIBRARYINFORMATION 구조체는 라이브러리 개체와 관련된 속성을 정의합니다.
NTMS_LIBREQUESTINFORMATIONA NTMS_LIBREQUESTINFORMATION 구조는 RSM에 큐에 대기 중인 작업 요청과 관련된 속성을 정의합니다. (ANSI)
NTMS_LIBREQUESTINFORMATIONW NTMS_LIBREQUESTINFORMATION 구조는 RSM에 큐에 대기 중인 작업 요청과 관련된 속성을 정의합니다. (유니코드)
NTMS_LMIDINFORMATION NTMS_LMIDINFORMATION 구조체는 논리 미디어 개체와 관련된 속성을 정의합니다.
NTMS_MEDIAPOOLINFORMATION NTMS_MEDIAPOOLINFORMATION 구조체는 미디어 풀 개체와 관련된 속성을 정의합니다.
NTMS_MEDIATYPEINFORMATION NTMS_MEDIATYPEINFORMATION 구조체는 RSM에서 지원하는 미디어 형식과 관련된 속성을 정의합니다.
NTMS_NOTIFICATIONINFORMATION NTMS_NOTIFICATIONINFORMATION 구조체는 RSM 데이터베이스에서 발생한 개체 및 작업을 정의합니다.
NTMS_OBJECTINFORMATIONA NTMS_OBJECTINFORMATION 구조는 애플리케이션이 RSM 디바이스, 미디어 및 시스템 컨트롤(예: 라이브러리, 드라이브, 미디어, 운영자 요청)에 대해 가져와서 설정할 수 있는 속성을 정의합니다. RSM 데이터베이스에 있는 개체의 일반적인 구조입니다. (ANSI)
NTMS_OBJECTINFORMATIONW NTMS_OBJECTINFORMATION 구조는 애플리케이션이 RSM 디바이스, 미디어 및 시스템 컨트롤(예: 라이브러리, 드라이브, 미디어, 운영자 요청)에 대해 가져와서 설정할 수 있는 속성을 정의합니다. RSM 데이터베이스에 있는 개체의 일반적인 구조입니다. (유니코드)
NTMS_OPREQUESTINFORMATIONA NTMS_OPREQUESTINFORMATION 구조는 RSM에 대한 연산자 요청 시스템 제어와 관련된 속성을 정의합니다. (ANSI)

NTMS_OPREQUESTINFORMATIONW NTMS_OPREQUESTINFORMATION 구조는 RSM에 대한 연산자 요청 시스템 제어와 관련된 속성을 정의합니다. (유니코드)
NTMS_PARTITIONINFORMATIONA NTMS_PARTITIONINFORMATION 구조체는 측면 개체와 관련된 속성을 정의합니다. (ANSI)
NTMS_PARTITIONINFORMATIONW NTMS_PARTITIONINFORMATION 구조체는 측면 개체와 관련된 속성을 정의합니다. (유니코드)
NTMS_PMIDINFORMATIONA NTMS_PMIDINFORMATION 구조체는 물리적 미디어 개체와 관련된 속성을 정의합니다. (ANSI)
NTMS_PMIDINFORMATIONW NTMS_PMIDINFORMATION 구조체는 물리적 미디어 개체와 관련된 속성을 정의합니다. (유니코드)
NTMS_STORAGESLOTINFORMATION NTMS_STORAGESLOTINFORMATION 구조체는 스토리지 슬롯 개체와 관련된 속성을 정의합니다.

열거형

NtmsObjectsTypes NtmsObjectsTypes 열거형 형식은 RSM 개체의 형식을 지정합니다.

ntmsmli.h 헤더

아티클2023. 08. 25.

이 헤더는 데이터 액세스 및 스토리지에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지

ntmsmli.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

콜백 함수

CLAIMMEDIALABEL ClaimMediaLabel 콜백 함수는 미디어의 연결된 애플리케이션에서 지정된 미디어 레이블을 만들었는지 여부를 결정합니다.
MAXMEDIALABEL MaxMediaLabel 콜백 함수는 미디어 레이블 라이브러리에서 지원하는 애플리케이션에 대한 미디어 레이블의 최대 크기를 결정합니다.

구조체

MediaLabelInfo MediaLabelInfo 구조체는 테이프 OMID에 대한 정보를 RSM 데이터베이스에 전달합니다. 미디어 레이블 라이브러리는 라이브러리가 인식하는 모든 미디어 레이블에 대해 이 구조를 채웁니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ntquery.h 헤더

아티클2023. 08. 25.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 인덱싱 서비스
- Windows 셸

ntquery.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

BindIFilterFromStorage 지정된 스토리지 개체에 대한 IFilter 인터페이스 포인터를 검색합니다. 이는 문서의 내용을 필터링하고 IStorage 인터페이스를 통해 액세스할 수 있는 포함된 OLE 개체를 처리할 때 특히 유용합니다.
BindIFilterFromStream 지정된 스토리지 개체에 대한 IFilter 인터페이스 포인터를 검색합니다. 이는 문서의 내용을 필터링하고 IStream 인터페이스를 통해 액세스할 수 있는 포함된 OLE 개체를 처리할 때 특히 유용합니다.
LoadIFilter 개체의 경로 이름에서 IFilter를 검색합니다.
SetCatalogState 백업 또는 기타 용도로 카탈로그 상태를 설정합니다.

구조체

CI_STATE 인덱싱 서비스 카탈로그의 현재 상태를 나타냅니다.
CIPROPERTYDEF 이름, 형식 및 ID(속성 식별자) 정보를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ntsecapi.h 헤더

아티클2023. 03. 08.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

ntsecapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AuditComputeEffectivePolicyBySid 지정된 보안 주체에 대한 하나 이상의 하위 범주에 대한 유효 감사 정책을 계산합니다. 이 함수는 시스템 감사 정책을 사용자별 정책과 결합하여 효과적인 감사 정책을 계산합니다.
AuditComputeEffectivePolicyByToken 지정된 토큰과 연결된 보안 주체에 대한 하나 이상의 하위 범주에 대한 유효 감사 정책을 계산합니다. 이 함수는 시스템 감사 정책을 사용자별 정책과 결합하여 효과적인 감사 정책을 계산합니다.
AuditEnumerateCategories 사용 가능한 감사 정책 범주를 열거합니다.
AuditEnumeratePerUserPolicy 사용자별 감사 정책이 지정된 사용자를 열거합니다.
AuditEnumerateSubCategories 사용 가능한 감사 정책 하위 범주를 열거합니다.
AuditFree 지정된 버퍼에 대해 감사 함수에 의해 할당된 메모리를 해제합니다.
AuditLookupCategoryGuidFromCategoryId 지정된 감사 정책 범주를 나타내는 GUID 구조를 검색합니다.
AuditLookupCategoryIdFromCategoryGuid 지정된 감사 정책 범주를 나타내는 POLICY_AUDIT_EVENT_TYPE 열거형의 요소를 검색합니다.

AuditLookupCategoryNameA 지정된 감사 정책 범주의 표시 이름을 검색합니다. (ANSI)
AuditLookupCategoryNameW 지정된 감사 정책 범주의 표시 이름을 검색합니다. (유니코드)
AuditLookupSubCategoryNameA 지정된 감사 정책 하위 범주의 표시 이름을 검색합니다. (ANSI)
AuditLookupSubCategoryNameW 지정된 감사 정책 하위 범주의 표시 이름을 검색합니다. (유니코드)
AuditQueryGlobalSaclA 감사 메시지에 대한 액세스를 위임하는 SACL(전역 시스템 액세스 제어 목록)을 검색합니다. (ANSI)
AuditQueryGlobalSaclW 감사 메시지에 대한 액세스를 위임하는 SACL(전역 시스템 액세스 제어 목록)을 검색합니다. (유니코드)
AuditQueryPerUserPolicy 지정된 보안 주체에 대한 하나 이상의 감사 정책 하위 범주에서 사용자별 감사 정책을 검색합니다.
AuditQuerySecurity 감사 정책에 대한 액세스를 위임하는 보안 설명자를 검색합니다.
AuditQuerySystemPolicy 하나 이상의 감사 정책 하위 범주에 대한 시스템 감사 정책을 검색합니다.
AuditSetGlobalSaclA 감사 메시지에 대한 액세스를 위임하는 SACL(전역 시스템 액세스 제어 목록)을 설정합니다. (ANSI)
AuditSetGlobalSaclW 감사 메시지에 대한 액세스를 위임하는 SACL(전역 시스템 액세스 제어 목록)을 설정합니다. (유니코드)
AuditSetPerUserPolicy 지정된 보안 주체에 대한 하나 이상의 감사 하위 범주에서 사용자별 감사 정책을 설정합니다.

AuditSetSecurity 감사 정책에 대한 액세스를 위임하는 보안 설명자를 설정합니다.
AuditSetSystemPolicy 하나 이상의 감사 정책 하위 범주에 대한 시스템 감사 정책을 설정합니다.
LsaAddAccountRights 계정에 하나 이상의 권한을 할당합니다.
LsaCallAuthenticationPackage 로그온 애플리케이션에서 인증 패키지와 통신하는 데 사용됩니다.
LsaClose LsaClose 함수는 Policy 또는 TrustedDomain 개체에 대한 핸들을 닫습니다.
LsaConnectUntrusted LSA 서버에 대한 신뢰할 수 없는 연결을 설정합니다.
LsaCreateTrustedDomainEx LsaCreateTrustedDomainEx 함수는 새 TrustedDomain 개체를 만들어 신뢰할 수 있는 새 도메인을 설정합니다.
LsaDeleteTrustedDomain LsaDeleteTrustedDomain 함수는 시스템의 신뢰할 수 있는 도메인 목록에서 신뢰할 수 있는 도메인을 제거하고 연결된 TrustedDomain 개체를 삭제합니다.
LsaDeregisterLogonProcess 호출자의 로그온 애플리케이션 컨텍스트를 삭제하고 LSA 서버에 대한 연결을 닫습니다.
LsaEnumerateAccountRights LsaEnumerateAccountRights 함수는 계정에 할당된 권한을 열거합니다.
LsaEnumerateAccountsWithUserRight 지정된 권한을 가진 LSA(로컬 보안 기관) 정책 개체의 데이터베이스에 있는 계정을 반환합니다.
LsaEnumerateLogonSessions 기존 로그온 세션 식별자(LUID) 집합 및 세션 수를 검색합니다.

LsaEnumerateTrustedDomains LsaEnumerateTrustedDomains 함수는 로그온 자격 증명을 인증하기 위해 신뢰할 수 있는 도메인의 이름과 SID를 검색합니다.
LsaEnumerateTrustedDomainsEx 로컬 시스템에서 신뢰하는 도메인에 대한 정보를 반환합니다.
LsaFreeMemory LsaFreeMemory 함수는 LSA 함수 호출에 의해 출력 버퍼에 할당된 메모리를 해제합니다.
LsaFreeReturnBuffer 이전에 LSA에서 할당한 버퍼에서 사용하는 메모리를 해제합니다.
LsaGetLogonSessionData 지정된 로그온 세션에 대한 정보를 검색합니다.
LsaLogonUser 저장된 자격 증명 정보를 사용하여 보안 주체의 로그온 데이터를 인증합니다.
LsaLookupAuthenticationPackage 인증 패키지의 고유 식별자를 가져옵니다.
LsaLookupNames 사용자, 그룹 또는 로컬 그룹 이름의 배열에 해당하는 SID(보안 식별자)를 검색합니다.
LsaLookupNames2 지정된 계정 이름의 SID(보안 식별자)를 검색합니다. LsaLookupNames2는 Windows 포리스트의 모든 도메인에 있는 모든 계정에 대한 SID를 조회할 수 있습니다.
LsaLookupSids SID(보안 식별자) 배열에 해당하는 이름을 조회합니다. LsaLookupSids가 SID에 해당하는 이름을 찾을 수 없는 경우 함수는 SID를 문자 형식으로 반환합니다.
LsaLookupSids2 SID(보안 식별자) 배열에 해당하는 이름을 조회하고 인터넷 공급자 ID를 지원합니다. LsaLookupSids2가 SID에 해당하는 이름을 찾을 수 없는 경우 함수는 SID를 문자 형식으로 반환합니다.

LsaNtStatusToWinError LsaNtStatusToWinError 함수는 LSA 함수에서 반환한 NTSTATUS 코드를 Windows 오류 코드로 변환합니다.
LsaOpenPolicy 로컬 또는 원격 시스템의 Policy 개체에 대한 핸들을 엽니다.
LsaOpenTrustedDomainByName LsaOpenTrustedDomainByName 함수는 원격 신뢰할 수 있는 도메인의 LSA 정책 핸들을 엽니다. 원격 머신의 LSA 정책을 설정하거나 쿼리하기 위해 이 핸들을 LSA 함수 호출에 전달할 수 있습니다.
LsaQueryDomainInformationPolicy Policyobject에서 도메인 정보를 검색합니다.
LsaQueryForestTrustInformation 지정된 로컬 보안 기관 TrustedDomain 개체에 대한 포리스트 트러스트 정보를 검색합니다.
LsaQueryInformationPolicy Policy 개체에 대한 정보를 검색합니다.
LsaQueryTrustedDomainInfo LsaQueryTrustedDomainInfo 함수는 신뢰할 수 있는 도메인에 대한 정보를 검색합니다.
LsaQueryTrustedDomainInfoByName LsaQueryTrustedDomainInfoByName 함수는 신뢰할 수 있는 도메인에 대한 정보를 반환합니다.
LsaRegisterLogonProcess LSA 서버에 대한 연결을 설정하고 호출자가 로그온 애플리케이션인지 확인합니다.
LsaRegisterPolicyChangeNotification LsaRegisterPolicyChangeNotification 함수는 LSA(로컬 보안 기관)에 이벤트 핸들을 등록합니다. 이 이벤트 핸들은 표시된 LSA 정책이 수정될 때마다 신호를 수신합니다.
LsaRemoveAccountRights 계정에서 하나 이상의 권한을 제거합니다.

LsaRetrievePrivateData LSA 프라이빗 데이터 함수를 사용하지 마세요. 대신 CryptProtectData 및 CryptUnprotectData 함수를 사용합니다. (LsaRetrievePrivateData)
LsaSetDomainInformationPolicy 도메인 정보를 Policyobject로 설정합니다.
LsaSetForestTrustInformation 지정된 로컬 보안 기관 TrustedDomain 개체에 대한 포리스트 트러스트 정보를 설정합니다.
LsaSetInformationPolicy Policy 개체의 정보를 수정합니다.
LsaSetTrustedDomainInfoByName LsaSetTrustedDomainInfoByName 함수는 TrustedDomain 개체의 값을 설정합니다.
LsaSetTrustedDomainInformation LsaSetTrustedDomainInformation 함수는 신뢰할 수 있는 도메인에 대한 정책 개체의 정보를 수정합니다.
LsaStorePrivateData LSA 프라이빗 데이터 함수를 사용하지 마세요. 대신 CryptProtectData 및 CryptUnprotectData 함수를 사용합니다. (LsaStorePrivateData)
LsaUnregisterPolicyChangeNotification LsaUnregisterPolicyChangeNotification 함수는 이전에 등록된 알림 이벤트를 사용하지 않도록 설정합니다.
RtlDecryptMemory RtlEncryptMemory 함수에 의해 이전에 암호화된 메모리 콘텐츠를 해독합니다.
RtlEncryptMemory 메모리 콘텐츠를 암호화합니다.
RtlGenRandom 의사 난수를 생성합니다.

콜백 함수

PSAM_INIT_NOTIFICATION_ROUTINE InitializeChangeNotify 함수는 암호 필터 DLL에 의해 구현됩니다. 이 함수는 DLL을 초기화합니다.
PSAM_PASSWORD_FILTER_ROUTINE 암호 필터 DLL에 의해 구현됩니다. 이 함수에서 반환되는 값은 시스템에서 새 암호를 허용하는지 여부를 결정합니다.
PSAM_PASSWORD_NOTIFICATION_ROUTINE 암호 필터 DLL에 의해 구현됩니다. 암호가 변경되었음을 DLL에 알릴 수 있습니다.

구조체

AUDIT_POLICY_INFORMATION 보안 이벤트 유형 및 해당 형식을 감사할 시기를 지정합니다.
DOMAIN_PASSWORD_INFORMATION 암호의 최소 길이 및 고유 암호가 있어야 하는 방법과 같은 도메인의 암호 정책에 대한 정보를 포함합니다.
KERB_ADD_BINDING_CACHE_ENTRY_EX_REQUEST 사용자가 특정 DC(도메인 컨트롤러)에 바인딩하여 Kerberos 도메인 바인딩 캐시를 재정의할 수 있습니다.
KERB_ADD_BINDING_CACHE_ENTRY_REQUEST 바인딩 캐시 항목을 추가할 메시지를 지정합니다.
KERB_ADD_CREDENTIALS_REQUEST 로그온 세션에 대한 추가 서버 자격 증명을 추가, 제거 또는 바꿀 메시지를 지정합니다.
KERB_ADD_CREDENTIALS_REQUEST_EX 로그온 세션에 대한 추가 서버 자격 증명을 추가, 제거 또는 바꿀 메시지와 해당 자격 증명과 연결할 SPN(서비스 사용자 이름)을 지정합니다.

KERB_BINDING_CACHE_ENTRY_DATA 바인딩 캐시 항목에 대한 데이터를 지정합니다.
KERB_CERTIFICATE_HASHINFO 인증서 해시의 페이로드 정보를 제공합니다.
KERB_CERTIFICATE_INFO 인증서 정보를 포함합니다.
KERB_CERTIFICATE_LOGON 스마트 카드 로그온 세션에 대한 정보를 포함합니다. (KERB_CERTIFICATE_LOGON)
KERB_CERTIFICATE_S4U_LOGON 사용자용 서비스(S4U) 로그온에 대한 인증서에 대한 정보를 포함합니다.
KERB_CERTIFICATE_UNLOCK_LOGON 대화형 스마트 카드 로그온 세션 중에 잠긴 워크스테이션의 잠금을 해제하는 데 사용되는 정보를 포함합니다.
KERB_CHANGEPASSWORD_REQUEST 암호를 변경하는 데 사용되는 정보를 포함합니다.
KERB_CLEANUP_MACHINE_PKINIT_CREDS_REQUEST 컴퓨터에서 PKINIT 디바이스 자격 증명을 정리합니다.
KERB_CRYPTO_KEY Kerberos 암호화 세션 키에 대한 정보를 포함합니다.
KERB_EXTERNAL_NAME 외부 이름에 대한 정보를 포함합니다.
KERB_EXTERNAL_TICKET 외부 티켓에 대한 정보를 포함합니다.
KERB_INTERACTIVE_LOGON 대화형 로그온 세션에 대한 정보를 포함합니다.

KERB_INTERACTIVE_PROFILE KERB_INTERACTIVE_PROFILE 구조에는 대화형 로그온 프로필에 대한 정보가 포함됩니다. 이 구조체는 LsaLogonUser 함수에서 사용됩니다.
KERB_INTERACTIVE_UNLOCK_LOGON 대화형 로그온 세션 중에 잠긴 워크스테이션의 잠금을 해제하는 데 사용되는 정보를 포함합니다.
KERB_PURGE_BINDING_CACHE_REQUEST 바인딩 캐시에 대한 요청을 삭제합니다.
KERB_PURGE_TKT_CACHE_REQUEST 티켓 캐시에서 항목을 삭제하는 데 사용되는 정보를 포함합니다.
KERB_QUERY_BINDING_CACHE_REQUEST 바인딩 캐시를 쿼리하는 데 사용되는 정보를 포함합니다.
KERB_QUERY_BINDING_CACHE_RESPONSE 바인딩 캐시를 쿼리한 결과를 포함합니다.
KERB_QUERY_DOMAIN_EXTENDED_POLICIES_REQUEST 확장 정책에 대한 도메인을 쿼리하는 데 사용되는 정보를 포함합니다.
KERB_QUERY_DOMAIN_EXTENDED_POLICIES_RESPONSE 지정된 도메인의 확장된 정책에 대한 쿼리 결과를 포함합니다.
KERB_QUERY_TKT_CACHE_REQUEST 티켓 캐시를 쿼리하는 데 사용되는 정보를 포함합니다.
KERB_QUERY_TKT_CACHE_RESPONSE 티켓 캐시를 쿼리한 결과를 포함합니다.
KERB_RETRIEVE_TKT_REQUEST 티켓을 검색하는 데 사용되는 정보를 포함합니다.
KERB_RETRIEVE_TKT_RESPONSE 티켓 검색의 응답을 포함합니다.

KERB_S4U_LOGON 사용자용 서비스(S4U) 로그온에 대한 정보를 포함합니다.
KERB_SMART_CARD_LOGON 스마트 카드 로그온 세션에 대한 정보를 포함합니다. (KERB_SMART_CARD_LOGON)
KERB_SMART_CARD_UNLOCK_LOGON 스마트 카드 로그온 세션 중에 잠긴 워크스테이션의 잠금을 해제하는 데 사용되는 정보를 포함합니다.
KERB_TICKET_CACHE_INFO 캐시된 Kerberos 티켓에 대한 정보를 포함합니다. Kerberos 티켓은 인터넷 RFC 4120에 정의되어 있습니다. 자세한 내용은 http://www.ietf.org 를 참조하세요.
KERB_TICKET_LOGON 네트워크 로그온에 대한 프로필 정보를 포함합니다.
KERB_TICKET_PROFILE KERB_TICKET_PROFILE 구조에는 대화형 로그온 프로필에 대한 정보가 포함됩니다. 이 구조체는 LsaLogonUser에서 반환됩니다.
KERB_TICKET_UNLOCK_LOGON 워크스테이션의 잠금을 해제하는 정보를 포함합니다.
LSA_AUTH_INFORMATION LSA_AUTH_INFORMATION 구조에는 신뢰할 수 있는 도메인에 대한 인증 정보가 포함됩니다.
LSA_ENUMERATION_INFORMATION LSA_ENUMERATION_INFORMATION 구조체는 SID에 대한 포인터를 반환하기 위해 LsaEnumerateAccountsWithUserRight 함수와 함께 사용됩니다.
LSA_FOREST_TRUST_BINARY_DATA 로컬 보안 기관 포리스트 트러스트 작업에 사용되는 이진 데이터를 포함합니다.
LSA_FOREST_TRUST_COLLISION_INFORMATION 로컬 보안 기관 포리스트 트러스트 충돌에 대한 정보를 포함합니다.

LSA_FOREST_TRUST_COLLISION_RECORD 로컬 보안 기관 포리스트 트러스트 충돌에 대한 정보를 포함합니다.
LSA_FOREST_TRUST_DOMAIN_INFO 도메인에 대한 식별 정보를 포함합니다.
LSA_FOREST_TRUST_INFORMATION 로컬 보안 기관 포리스트 트러스트 정보를 포함합니다.
LSA_FOREST_TRUST_RECORD 로컬 보안 기관 포리스트 트러스트 레코드를 나타냅니다.
LSA_LAST_INTER_LOGON_INFO 로그온 세션에 대한 정보를 포함합니다. (LSA_LAST_INTER_LOGON_INFO)
LSA_TRANSLATED_SID LsaLookupNames 함수와 함께 계정을 식별하는 SID에 대한 정보를 반환하는 데 사용됩니다.
MSV1_0_INTERACTIVE_LOGON 대화형 로그온에 대한 정보를 포함합니다.
MSV1_0_INTERACTIVE_PROFILE MSV1_0_INTERACTIVE_PROFILE 구조에는 대화형 로그온 프로필에 대한 정보가 포함됩니다. 이 구조체는 LsaLogonUser 함수에서 사용됩니다.
MSV1_0_LM20_LOGON 네트워크 로그온에 사용되는 로그온 정보를 포함합니다.
MSV1_0_LM20_LOGON_PROFILE 네트워크 로그온 세션에 대한 정보를 포함합니다.
MSV1_0_SUBAUTH_LOGON 하위 인증 DLL에서 사용됩니다.
MSV1_0_SUBAUTH_REQUEST 하위 인증 패키지에 전달할 정보를 포함합니다.

MSV1_0_SUBAUTH_RESPONSE 하위 인증 패키지의 응답을 포함합니다.
MSV1_0_SUPPLEMENTAL_CREDENTIAL MSV1_0_SUPPLEMENTAL_CREDENTIAL 구조는 Kerberos 또는 사용자 지정 인증 패키지에서 MSV1_0 자격 증명을 전달하는 데 사용됩니다.
PKU2U_CERT_BLOB PKU2U 인증서 데이터를 지정합니다.
PKU2U_CERTIFICATE_S4U_LOGON S4U 로그온에 사용되는 인증서를 지정합니다.
PKU2U_CREDUI_CONTEXT PKU2U 클라이언트 컨텍스트를 지정합니다.
POLICY_AUDIT_EVENTS_INFO POLICY_AUDIT_EVENTS_INFO 구조는 시스템의 감사 규칙을 설정하고 쿼리하는 데 사용됩니다.
POLICY_AUDIT_SID_ARRAY Windows 사용자 또는 그룹을 나타내는 SID 구조의 배열을 지정합니다.
POLICY_LSA_SERVER_ROLE_INFO LSA 서버의 역할을 설정하고 쿼리하는 데 사용됩니다.
POLICY_MODIFICATION_INFO POLICY_MODIFICATION_INFO 구조체는 LSA 데이터베이스의 생성 시간 및 마지막 수정에 대한 정보를 쿼리하는 데 사용됩니다.
POLICY_PRIMARY_DOMAIN_INFO PolicyPrimaryDomainInformation 값 및 POLICY_PRIMARY_DOMAIN_INFO 구조체는 사용되지 않습니다. PolicyDnsDomainInformation 및 POLICY_DNS_DOMAIN_INFO 구조를 대신 사용합니다.
SECURITY_LOGON_SESSION_DATA 로그온 세션에 대한 정보를 포함합니다. (SECURITY_LOGON_SESSION_DATA)

TRUSTED_DOMAIN_AUTH_INFORMATION TRUSTED_DOMAIN_AUTH_INFORMATION 구조는 신뢰할 수 있는 도메인에 대한 인증 정보를 검색하는 데 사용됩니다. InformationClass 매개 변수가 TrustedDomainAuthInformation으로 설정된 경우 LsaQueryTrustedDomainInfo 함수는 이 구조를 사용합니다.
TRUSTED_DOMAIN_FULL_INFORMATION 신뢰할 수 있는 도메인에 대한 전체 정보를 검색하는 데 사용됩니다.
TRUSTED_DOMAIN_INFORMATION_EX 신뢰할 수 있는 도메인에 대한 확장 정보를 검색하는 데 사용됩니다.
TRUSTED_DOMAIN_NAME_INFO 신뢰할 수 있는 도메인의 이름을 쿼리하거나 설정하는 데 사용됩니다.
TRUSTED_PASSWORD_INFO TRUSTED_PASSWORD_INFO 구조는 신뢰할 수 있는 도메인에 대한 암호를 쿼리하거나 설정하는 데 사용됩니다.
TRUSTED_POSIX_OFFSET_INFO Posix 사용자 및 그룹 식별자를 생성하는 데 사용되는 값을 쿼리하거나 설정하는 데 사용됩니다.

열거형

KERB_CERTIFICATE_INFO_TYPE 제공되는 인증서 정보의 유형을 지정합니다.
KERB_LOGON_SUBMIT_TYPE 요청되는 로그온 유형을 식별합니다.
KERB_PROFILE_BUFFER_TYPE 반환된 로그온 프로필의 유형을 나열합니다.
KERB_PROTOCOL_MESSAGE_TYPE LsaCallAuthenticationPackage 함수를 호출하여 Kerberos 인증 패키지로 보낼 수 있는 메시지 유형을 나열합니다.

LSA_FOREST_TRUST_COLLISION_RECORD_TYPE 로컬 보안 기관 포리스트 트러스트 레코드 간에 발생할 수 있는 충돌 유형을 정의합니다.
LSA_FOREST_TRUST_RECORD_TYPE 로컬 보안 기관 포리스트 트러스트 레코드의 형식을 정의합니다.
MSV1_0_LOGON_SUBMIT_TYPE 요청되는 로그온의 종류를 나타냅니다.
MSV1_0_PROFILE_BUFFER_TYPE 반환된 로그온 프로필의 종류를 나열합니다.
MSV1_0_PROTOCOL_MESSAGE_TYPE LsaCallAuthenticationPackage 함수를 호출하여 MSV1_0 인증 패키지로 보낼 수 있는 메시지 유형을 나열합니다.
PKU2U_LOGON_SUBMIT_TYPE PKU2U_CERTIFICATE_S4U_LOGON 구조에 전달된 로그온 메시지의 유형을 나타냅니다.
POLICY_AUDIT_EVENT_TYPE POLICY_AUDIT_EVENT_TYPE 열거형은 시스템에서 감사할 수 있는 이벤트 유형을 나타내는 값을 정의합니다.
POLICY_DOMAIN_INFORMATION_CLASS 정책 도메인 정보의 유형을 정의합니다.
POLICY_INFORMATION_CLASS Policy 개체에서 설정하거나 쿼리할 정보 유형을 나타내는 값을 정의합니다.
POLICY_LSA_SERVER_ROLE LSA 서버의 역할을 나타내는 값을 정의합니다.
POLICY_NOTIFICATION_INFORMATION_CLASS POLICY_NOTIFICATION_INFORMATION_CLASS 열거형은 애플리케이션이 변경 알림을 요청할 수 있는 정책 정보 및 정책 도메인 정보의 유형을 정의합니다.

POLICY_SERVER_ENABLE_STATE POLICY_SERVER_ENABLE_STATE 열거형은 LSA 서버의 상태를 나타냅니다.즉, 사용 여부입니다. 일부 작업은 사용하도록 설정된 LSA 서버에서만 수행할 수 있습니다.
SECURITY_LOGON_TYPE 로그온 프로세스에서 요청한 로그온 유형을 나타냅니다.
TRUSTED_INFORMATION_CLASS TRUSTED_INFORMATION_CLASS 열거형 형식은 신뢰할 수 있는 도메인에 대해 설정하거나 쿼리할 정보 유형을 나타내는 값을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ntsecpkg.h 헤더

아티클2023. 08. 25.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보
- 보안 및 ID

ntsecpkg.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CredMarshalTargetInfo 지정된 대상을 바이트 값 배열로 직렬화합니다.

콜백 함수

CredFreeCredentialsFn 보안 패키지에서 사용하는 자격 증명을 저장하는 데 사용되는 메모리를 해제합니다.
CrediUnmarshalandDecodeStringFn 마샬링된 문자열을 원래 형식으로 다시 변환하고, 숨을 들이지 않은 문자열의 암호를 해독합니다.
CredReadDomainCredentialsFn 자격 증명 관리자에서 도메인 자격 증명을 읽습니다.
CredReadFn 자격 증명 관리자에서 자격 증명을 읽습니다.
CredWriteFn 지정된 자격 증명을 자격 증명 관리자에 씁니다.
KspDeleteContextFn 보안 컨텍스트를 삭제합니다.

KspMakeSignatureFn 지정된 메시지 및 보안 컨텍스트에 따라 서명을 생성합니다.
KspVerifySignatureFn 받은 메시지가 서명에 따라 올바른지 확인합니다.
LSA_ADD_CREDENTIAL 로그온 세션에 자격 증명을 추가합니다.
LSA_ALLOCATE_CLIENT_BUFFER 클라이언트의 주소 공간에 버퍼를 할당합니다.
LSA_ALLOCATE_LSA_HEAP 힙에 메모리를 할당합니다. LSA에 다시 전달된 일부 정보는 이 함수를 사용하여 할당되어야 합니다.
LSA_ALLOCATE_PRIVATE_HEAP 프라이빗 힙에 메모리를 할당합니다.
LSA_ALLOCATE_SHARED_MEMORY AllocateSharedMemory 함수는 CreateSharedMemory 함수를 호출하여 이전에 예약한 메모리 섹션에서 공유 메모리 블록을 할당합니다.
LSA_AP_CALL_PACKAGE LSA에 대한 신뢰할 수 있는 연결이 있는 로그온 애플리케이션이 LsaCallAuthenticationPackage 함수를 호출하고 인증 패키지의 식별자를 지정하면 LSA(로컬 보안 기관)에서 호출됩니다.
LSA_AP_CALL_PACKAGE_PASSTHROUGH LsaCallAuthenticationPackage 함수로 전송된 통과 로그온 요청에 대한 디스패치 함수입니다.
LSA_AP_INITIALIZE_PACKAGE 시스템 초기화 중에 LSA(로컬 보안 기관)에서 한 번 호출하여 인증 패키지를 초기화할 기회를 제공합니다.
LSA_AP_LOGON_TERMINATED 로그온 세션이 종료되면 인증 패키지를 알리는 데 사용됩니다. 로그온 세션을 참조하는 마지막 토큰이 삭제되면 로그온 세션이 종료됩니다.

LSA_AP_LOGON_USER LSA_AP_LOGON_USER(ntsecpkg.h) 콜백 함수는 사용자의 로그온 자격 증명을 인증합니다.
LSA_AP_LOGON_USER_EX LSA_AP_LOGON_USER_EX(ntsecpkg.h) 콜백 함수는 사용자의 로그온 자격 증명을 인증합니다.
LSA_AP_LOGON_USER_EX2 사용자의 초기 로그온에서 사용자 로그온 시도를 인증하는 데 사용됩니다. 사용자에 대한 새 로그온 세션이 설정되고 사용자에 대한 유효성 검사 정보가 반환됩니다.
LSA_AUDIT_ACCOUNT_LOGON AuditAccountLogon 함수는 외부 보안 주체 이름을 Windows 계정에 매핑하는 것을 나타내는 감사 레코드를 생성합니다.
LSA_AUDIT_LOGON AuditLogon 함수는 로그온 시도를 감사하는 데 사용됩니다.
LSA_CALL_PACKAGE CallPackage 함수는 다른 보안 패키지를 호출하여 해당 서비스에 액세스하는 데 사용됩니다.
LSA_CALL_PACKAGE_PASSTHROUGH CallPackagePassthrough 함수는 다른 보안 패키지를 호출하여 해당 서비스에 액세스하는 데 사용됩니다.
LSA_CALL_PACKAGEEX CallPackageEx 함수는 다른 보안 패키지를 호출하여 해당 서비스에 액세스하는 데 사용됩니다.
LSA_CANCEL_NOTIFICATION CancelNotification 함수는 이전에 등록된 알림을 취소합니다.
LSA_CLIENT_CALLBACK LSA(로컬 보안 기관) 모드 보안 패키지가 해당 사용자 모드 패키지로 다시 호출하고 해당 DLL에서 함수를 호출할 수 있도록 허용합니다.
LSA_CLOSE_SAM_USER SAM(보안 계정 관리자) 사용자 계정에 대한 핸들을 닫습니다.

LSA_CONVERT_AUTH_DATA_TO_TOKEN ConvertAuthDataToToken 함수는 GetAuthDataForUser 또는 GetUserAuthData 함수에서 반환된 권한 부여 데이터에서 액세스 토큰을 만듭니다.
LSA_COPY_FROM_CLIENT_BUFFER 클라이언트 프로세스의 주소 공간에서 현재 프로세스의 버퍼로 정보를 복사합니다.
LSA_COPY_TO_CLIENT_BUFFER 현재 프로세스의 버퍼에서 클라이언트 프로세스의 주소 공간으로 정보를 복사합니다.
LSA_CRACK_SINGLE_NAME CrackSingleName 함수는 이름을 한 형식에서 다른 형식으로 변환합니다.
LSA_CREATE_LOGON_SESSION 로그온 세션을 만듭니다.
LSA_CREATE_SHARED_MEMORY CreateSharedMemory 함수는 클라이언트 프로세스 및 보안 패키지에서 공유하는 메모리 섹션을 만듭니다.
LSA_CREATE_THREAD LSA(로컬 보안 기관)에서 사용해야 하는 Windows CreateThread 함수에 대한 래퍼입니다.
LSA_CREATE_TOKEN CreateToken 함수는 SSP/AP에서 SpAcceptLsaModeContext에 대한 호출을 처리하는 동안 토큰을 만드는 데 사용됩니다.
LSA_CREATE_TOKEN_EX SpAcceptLsaModeContext에 대한 호출을 처리하는 동안 토큰을 만듭니다.
LSA_DELETE_CREDENTIAL 기존 자격 증명을 삭제합니다.
LSA_DELETE_LOGON_SESSION 사용자의 인증 정보가 합법적인지 여부를 확인하면서 만든 로그온 세션을 정리합니다.
LSA_DELETE_SHARED_MEMORY

DeleteSharedMemory 함수는 클라이언트와 보안 패키지에서 공유하는 메모리 섹션을 해제합니다.
LSA_DUPLICATE_HANDLE DuplicateHandle 함수는 중복 핸들을 만듭니다. 반환된 중복 항목은 호출자의 프로세스 공간에 있습니다.
LSA_EXPAND_AUTH_DATA_FOR_DOMAIN 지정된 사용자 인증 데이터의 도메인 그룹을 확장합니다.
LSA_FREE_CLIENT_BUFFER AllocateClientBuffer 함수를 사용하여 이전에 할당된 클라이언트 버퍼를 해제합니다.
LSA_FREE_LSA_HEAP FreeReturnBuffer 함수는 LSA(로컬 보안 기관)에서 할당한 버퍼를 해제하고 보안 패키지로 반환하는 데 사용됩니다. 반환된 버퍼의 정보가 더 이상 필요하지 않은 경우 패키지는 이 함수를 호출합니다.
LSA_FREE_PRIVATE_HEAP AllocatePrivateHeap 함수를 사용하여 할당된 메모리를 해제합니다.
LSA_FREE_SHARED_MEMORY FreeSharedMemory 함수는 AllocateSharedMemory 함수에서 이전에 할당한 공유 메모리 블록을 해제합니다.
LSA_GET_AUTH_DATA_FOR_USER GetAuthDataForUser 함수는 SAM(보안 계정 관리자) 데이터베이스에서 사용자에 대한 인증 정보를 검색하여 ConvertAuthDataToken 함수에 적합한 형식으로 만듭니다.
LSA_GET_CALL_INFO GetCallInfo 함수는 최신 함수 호출에 대한 정보를 검색합니다.
LSA_GET_CLIENT_INFO GetClientInfo 함수는 스레드 및 프로세스 ID와 같은 클라이언트 프로세스에 대한 정보와 클라이언트의 상태 및 권한을 나타내는 플래그를 가져옵니다.
LSA_GET_CREDENTIALS 로그온 세션과 연결된 자격 증명을 검색합니다.
LSA_GET_USER_AUTH_DATA

GetUserAuthData 함수는 단일 버퍼에서 사용자에 대한 권한 부여 데이터를 반환합니다.
LSA_MAP_BUFFER SecBuffer 구조를 SSP/AP(보안 지원 공급자/인증 패키지)의 주소 공간에 매핑합니다.
LSA_OPEN_SAM_USER SAM(보안 계정 관리자) 데이터베이스에서 사용자 계정에 대한 핸들을 검색합니다.
LSA_OPEN_TOKEN_BY_LOGON_ID 지정된 사용자 로그온과 연결된 사용자 액세스 토큰을 엽니다.
LSA_PROTECT_MEMORY 지정된 메모리 버퍼를 암호화합니다.
LSA_REGISTER_NOTIFICATION 보안 패키지에 알림을 제공하는 메커니즘을 제공합니다. 알림은 이벤트 개체가 신호를 받으면 또는 특정 시스템 이벤트 중에 고정된 간격으로 발생할 수 있습니다.
LSA_UPDATE_PRIMARY_CREDENTIALS 한 보안 패키지에 대한 메커니즘을 제공하여 로그온 세션에 대한 자격 증명이 변경되었음을 다른 패키지에 알립니다.
SpAcceptCredentialsFn 인증된 보안 주체에 대해 저장된 자격 증명을 보안 패키지에 전달하기 위해 LSA(로컬 보안 기관)에서 호출합니다.
SpAcceptLsaModeContextFn 서버 및 클라이언트에서 공유하는 보안 컨텍스트를 만드는 데 사용되는 서버 디스패치 함수입니다.
SpAcquireCredentialsHandleFn 보안 주체의 자격 증명에 대한 핸들을 가져오기 위해 호출됩니다.
SpAddCredentialsFn 보안 주체에 대한 자격 증명을 추가하는 데 사용됩니다.
SpApplyControlTokenFn 보안 컨텍스트에 컨트롤 토큰을 적용합니다. 이 함수는 현재 LSA(로컬 보안 기관)에서 호출되지 않습니다.

SpCompleteAuthTokenFn 인증 토큰을 완료합니다. S
SpDeleteCredentialsFn 보안 패키지의 기본 또는 추가 자격 증명 목록에서 자격 증명을 삭제합니다.
SpExchangeMetaDataFn 보안 지원 공급자에게 메타데이터를 보냅니다.
SpExportSecurityContextFn 보안 컨텍스트를 다른 프로세스로 내보냅니다.
SpFormatCredentialsFn 사용자 개체에 저장할 자격 증명의 형식을 지정합니다.
SpFreeCredentialsHandleFn SpAcquireCredentialsHandle 함수를 호출하여 획득한 자격 증명을 해제합니다.
SpGetContextTokenFn 가장할 토큰을 가져옵니다.
SpGetCredentialsFn 사용자 개체에서 기본 및 추가 자격 증명을 검색합니다.
SpGetCredUIContextFn 자격 증명 공급자에서 컨텍스트 정보를 검색합니다. (SpGetCredUIContextFn)
SpGetExtendedInformationFn 보안 패키지에 대한 확장 정보를 제공합니다.
SpGetInfoFn 보안 패키지에 대한 일반 정보(예: 이름 및 기능)를 제공합니다.
SpGetUserInfoFn 로그온 세션에 대한 정보를 검색합니다.
SpImportSecurityContextFn

다른 프로세스에서 보안 컨텍스트를 가져옵니다.
SpInitializeFn LSA(로컬 보안 기관)에서 일반 보안 정보와 지원 기능의 디스패치 테이블을 제공하는 보안 패키지를 한 번 호출합니다.
SpInitLsaModeContextFn 서버와 클라이언트 간에 보안 컨텍스트를 설정하는 데 사용되는 클라이언트 디스패치 함수입니다.
SpInitUserModeContextFn 압축된 LSA(로컬 보안 기관) 모드 컨텍스트에서 사용자 모드 보안 컨텍스트를 만듭니다.
SpInstanceInitFn SSP/AP에서 사용자 모드 보안 패키지를 초기화합니다.
SpLsaModeInitializeFn SSP/AP DLL의 각 보안 패키지에서 구현하는 함수에 대한 포인터를 LSA에 제공합니다.
SpMarshallSupplementalCredsFn 공용 형식의 추가 자격 증명을 로컬 프로시저 호출에 적합한 형식으로 변환합니다.
SpQueryContextAttributesFn 보안 컨텍스트의 특성을 검색합니다.
SpQueryCredentialsAttributesFn 자격 증명의 특성을 검색합니다.
SpQueryMetaDataFn 보안 컨텍스트를 시작할 때 SSP(보안 지원 공급자)에서 메타데이터를 가져옵니다.
SpSaveCredentialsFn 추가 자격 증명을 사용자 개체에 저장합니다.
SpSealMessageFn 클라이언트와 서버 간에 교환되는 메시지를 암호화합니다.
SpSetExtendedInformationFn 보안 패키지에 대한 확장 정보를 설정합니다.

SpUnsealMessageFn 이전에 SpSealMessage 함수로 암호화된 메시지의 암호를 해독합니다.
SpUpdateCredentialsFn 지정된 컨텍스트와 연결된 자격 증명을 업데이트. (SpUpdateCredentialsFn)
SpUserModeInitializeFn 보안 지원 공급자/인증 패키지(SSP/AP) DLL이 클라이언트/서버 애플리케이션의 프로세스 공간에 로드될 때 호출됩니다. 이 함수는 SSP/AP DLL의 각 보안 패키지에 대한 SECPKG_USER_FUNCTION_TABLE 테이블을 제공합니다.
SpValidateTargetInfoFn 지정된 SECPKG_TARGETINFO 구조체가 유효한 대상을 나타내는지 확인합니다.

구조체

ENCRYPTED_CREDENTIALW 암호화된 자격 증명을 나타냅니다.
LSA_DISPATCH_TABLE Windows 인증 패키지가 호출할 수 있는 LSA(로컬 보안 기관) 함수에 대한 포인터를 포함합니다.
LSA_SECPKG_FUNCTION_TABLE 보안 패키지에서 호출할 수 있는 LSA 함수에 대한 포인터를 포함합니다. LSA(로컬 보안 기관)는 패키지의 SpInitialize 함수를 호출할 때 이 구조를 보안 패키지에 전달합니다.
LSA_TOKEN_INFORMATION_NULL 인증되지 않은 시스템 액세스가 필요한 경우에 사용됩니다.
LSA_TOKEN_INFORMATION_V1 버전 2 Windows 토큰 개체에 인증 패키지가 배치할 수 있고 LSA_TOKEN_INFORMATION_V1 대체된 정보를 포함합니다.
LSA_TOKEN_INFORMATION_V3

LSA 토큰에 클레임 지원을 추가하고 인증 패키지가 버전 3 Windows 토큰 개체에 배치할 수 있고 LSA_TOKEN_INFORMATION_V1 대체한 정보를 포함합니다.
SECPKG_BYTE_VECTOR 바이트 벡터 정보를 지정합니다.
SECPKG_CALL_INFO 현재 실행 중인 호출에 대한 정보를 포함합니다.
SECPKG_CLIENT_INFO SECPKG_CLIENT_INFO 구조에는 보안 패키지의 클라이언트에 대한 정보가 있습니다. 이 구조체는 GetClientInfo 함수에서 사용됩니다.
SECPKG_CONTEXT_THUNKS SECPKG_CONTEXT_THUNKS 구조에는 LSA 모드에서 실행할 QueryContextAttributes(일반) 호출에 대한 정보가 포함되어 있습니다. 이 구조체는 SpGetExtendedInformation 및 SpSetExtendedInformation 함수에서 사용됩니다.
SECPKG_CREDENTIAL 자격 증명을 지정합니다.
SECPKG_DLL_FUNCTIONS SECPKG_DLL_FUNCTIONS 구조에는 클라이언트/서버 애플리케이션을 사용하여 in-process를 실행하는 동안 보안 패키지가 호출할 수 있는 LSA 함수에 대한 포인터가 포함되어 있습니다.
SECPKG_EVENT_NOTIFY SECPKG_EVENT_NOTIFY 구조에는 보안 이벤트에 대한 정보가 포함됩니다. 이 구조는 이벤트 알림을 수신하기 위해 등록된 함수에 전달됩니다. 이벤트 알림 함수는 RegisterNotification 함수를 호출하여 등록됩니다.
SECPKG_EVENT_PACKAGE_CHANGE SECPKG_EVENT_PACKAGE_CHANGE 구조에는 보안 패키지 가용성의 변경 내용에 대한 정보가 포함되어 있습니다.
SECPKG_EXTENDED_INFORMATION SECPKG_EXTENDED_INFORMATION 구조체는 선택적 패키지 기능에 대한 정보를 보관하는 데 사용됩니다. 이 구조체는 SpGetExtendedInformation 및 SpSetExtendedInformation 함수에서 사용됩니다.
SECPKG_EXTRA_OIDS

확장 보안 패키지에 대한 개체 식별자(ID)를 포함합니다.
SECPKG_FUNCTION_TABLE SECPKG_FUNCTION_TABLE 구조에는 보안 패키지가 구현해야 하는 LSA 함수에 대한 포인터가 포함되어 있습니다. LSA(로컬 보안 기관)는 SpLsaModeInitialize 함수를 호출할 때 SSP/AP DLL에서 이 구조를 가져옵니다.
SECPKG_GSS_INFO SECPKG_GSS_INFO 구조에는 GSS 호환 협상에 사용되는 정보가 포함되어 있습니다.
SECPKG_MUTUAL_AUTH_LEVEL SECPKG_MUTUAL_AUTH_LEVEL 구조에는 보안 패키지에서 사용하는 인증 수준이 포함됩니다.
SECPKG_NEGO2_INFO NEGO2 협상에 사용되는 확장 패키지 정보를 포함합니다.
SECPKG_PARAMETERS SECPKG_PARAMETERS 구조에는 컴퓨터 시스템에 대한 정보가 포함됩니다. 이 구조체는 SpInitialize 함수에서 사용됩니다.
SECPKG_PRIMARY_CRED SECPKG_PRIMARY_CRED 구조에는 기본 자격 증명이 포함됩니다. 이 구조체는 LsaApLogonUserEx2 및 SpAcceptCredentials 함수에서 사용됩니다.
SECPKG_SERIALIZED_OID 보안 패키지의 OID(개체 식별자)를 포함합니다.
SECPKG_SHORT_VECTOR 짧은 벡터 정보를 지정합니다.
SECPKG_SUPPLEMENTAL_CRED SECPKG_SUPPLEMENTAL_CRED 구조에는 보안 패키지에서 인식하는 추가 자격 증명이 포함되어 있습니다.
SECPKG_SUPPLEMENTAL_CRED_ARRAY SECPKG_SUPPLEMENTAL_CRED_ARRAY 구조에는 추가 자격 증명 정보가 포함되어 있습니다. 이 구조체는 LsaApLogonUserEx2 및 UpdateCredentials 함수에서 사용됩니다.
SECPKG_SUPPLIED_CREDENTIAL

제공된 자격 증명을 지정합니다.
SECPKG_TARGETINFO 인증 요청의 대상을 지정합니다.
SECPKG_USER_FUNCTION_TABLE SECPKG_USER_FUNCTION_TABLE 구조에는 클라이언트/서버 애플리케이션을 사용하여 프로세스 실행을 지원하기 위해 보안 패키지가 구현하는 함수에 대한 포인터가 포함되어 있습니다. 이 구조체는 SpUserModeInitialize 함수에서 제공됩니다.
SECPKG_WOW_CLIENT_DLL WOW 인식 32비트 DLL에 대한 경로를 포함합니다.
SECURITY_USER_DATA SecurityUserData 구조에는 보안 지원 공급자/인증 패키지의 사용자에 대한 정보가 포함되어 있습니다. 이 구조체는 SpGetUserInfo 함수에서 사용됩니다.

열거형

LSA_TOKEN_INFORMATION_TYPE 로그온 토큰에 포함할 수 있는 정보 수준을 지정합니다.
SECPKG_EXTENDED_INFORMATION_CLASS SECPKG_EXTENDED_INFORMATION_CLASS 열거형은 보안 패키지에 대해 설정하거나 가져올 정보 유형을 설명합니다. 이 열거형은 SpGetExtendedInformation 및 SpSetExtendedInformation 함수에서 사용됩니다.
SECPKG_NAME_TYPE SECPKG_NAME_TYPE 열거형은 계정에 지정된 이름의 유형을 설명하는 데 사용됩니다. SECPKG_NAME_TYPE 열거형은 GetAuthDataForUser 및 OpenSamUser 함수에서 사용됩니다.
SECPKG_SESSIONINFO_TYPE 세션 정보의 형식을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

nvme.h 헤더

아티클2023. 03. 20.

이 헤더는 데이터 액세스 및 스토리지에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지
- NVMe 드라이브 작업

nvme.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

NVM_SET_LIST NVME 특성 설정 명령에 대한 항목 배열을 포함합니다.
NVME_ADMIN_COMPLETION_QUEUE_BASE_ADDRESS 관리 완료 큐의 기본 메모리 주소를 포함합니다.
NVME_ADMIN_QUEUE_ATTRIBUTES 관리 제출 큐 및 관리 완료 큐에 대한 AQA(관리 큐 특성)를 포함합니다.
NVME_ADMIN_SUBMISSION_QUEUE_BASE_ADDRESS 관리 제출 큐의 기본 메모리 주소를 포함합니다.
NVME_AUTO_POWER_STATE_TRANSITION_ENTRY 허용되는 각 32개 전원 상태에 대한 유휴 시간 및 전원 상태 전환에 대한 정보를 지정하는 64비트 항목을 포함합니다.
NVME_CDW10_ABORT 이전에 관리 제출 큐 또는 I/O 제출 큐에 제출된 특정 명령을 중단하는 데 사용되는 중단 명령에 대한 매개 변수를 포함합니다.
NVME_CDW10_CREATE_IO_QUEUE I/O 완료 큐 만들기 및 IO 제출 큐 만들기 명령에 사용되는 매개 변수를 포함합니다.

NVME_CDW10_DATASET_MANAGEMENT NVME_CDW10_DATASET_MANAGEMENT 구조에는 논리 블록 범위에 대한 특성을 나타내는 데이터 세트 관리 명령에 대한 매개 변수가 포함되어 있습니다.
NVME_CDW10_DIRECTIVE_RECEIVE 지시문 수신 명령에 대한 매개 변수를 포함합니다.
NVME_CDW10_DIRECTIVE_SEND 지시문 보내기 명령에 대한 매개 변수를 포함합니다.
NVME_CDW10_FIRMWARE_ACTIVATE 펌웨어 커밋 명령에 사용되는 매개 변수를 포함합니다.
NVME_CDW10_FIRMWARE_DOWNLOAD NVME_CDW10_FIRMWARE_DOWNLOAD 구조에는 새 펌웨어 이미지를 컨트롤러에 복사하는 펌웨어 이미지 다운로드 명령에 대한 매개 변수가 포함되어 있습니다.
NVME_CDW10_FORMAT_NVM NVM 미디어 형식을 낮게 지정하는 데 사용되는 NVM 형식 명령에 대한 매개 변수를 포함합니다.
NVME_CDW10_GET_FEATURES 지정된 기능의 특성을 검색하는 기능 가져오기 명령에 대한 매개 변수를 포함합니다.
NVME_CDW10_GET_LOG_PAGE NVME_CDW10_GET_LOG_PAGE 구조에는 요청된 로그 페이지가 포함된 데이터 버퍼를 반환하는 로그 페이지 가져오기 명령에 대한 매개 변수가 포함되어 있습니다.
NVME_CDW10_GET_LOG_PAGE_V13 NVME_CDW10_GET_LOG_PAGE_V13 구조에는 요청된 로그 페이지가 포함된 데이터 버퍼를 반환하는 로그 페이지 가져오기 명령에 대한 매개 변수가 포함되어 있습니다.
NVME_CDW10_IDENTIFY NVM 하위 시스템, 컨트롤러 또는 네임스페이스에 대한 정보를 설명하는 데이터 버퍼를 반환하는 Identify 명령에 대한 매개 변수를 포함합니다.
NVME_CDW10_SECURITY_SEND_RECEIVE 관리 명령 보안 보내기 및 보안 수신에 사용되는 매개 변수를 포함합니다.

NVME_CDW10_SET_FEATURES NVME_CDW10_SET_FEATURES 구조에는 지정된 기능의 특성을 설정하는 기능 설정 명령에 대한 매개 변수가 포함되어 있습니다.
NVME_CDW11_CREATE_IO_CQ 관리 완료 큐를 제외하고 모든 I/O 완료 큐를 만드는 데 사용되는 I/O 완료 큐 만들기 명령에 대한 매개 변수를 포함합니다.
NVME_CDW11_CREATE_IO_SQ IO 제출 큐를 만드는 데 사용되는 IO 제출 큐 만들기 명령에 대한 매개 변수를 포함합니다.
NVME_CDW11_DATASET_MANAGEMENT NVME_CDW11_DATASET_MANAGEMENT 구조에는 논리 블록 범위에 대한 특성을 나타내는 데이터 세트 관리 명령에 대한 매개 변수가 포함되어 있습니다.
NVME_CDW11_DIRECTIVE_RECEIVE 지시문 수신 명령에 대한 매개 변수를 포함합니다.
NVME_CDW11_DIRECTIVE_SEND 지시문 보내기 명령에 대한 매개 변수를 포함합니다.
NVME_CDW11_FEATURE_ARBITRATION 명령 중재를 제어하는 중재 기능에 대한 값을 포함합니다.
NVME_CDW11_FEATURE_ASYNC_EVENT_CONFIG 호스트에 대한 비동기 이벤트 알림을 트리거하는 이벤트를 제어하는 비동기 이벤트 구성 기능에 대한 매개 변수를 포함합니다.
NVME_CDW11_FEATURE_AUTO_POWER_STATE_TRANSITION 자율 전원 상태 전환에 대한 설정을 구성하는 자율 전원 상태 전환 기능에 대한 매개 변수를 포함합니다.
NVME_CDW11_FEATURE_HOST_MEMORY_BUFFER 호스트가 컨트롤러에서 단독으로 사용할 호스트 메모리의 일부를 할당하는 메커니즘을 제공하는 호스트 메모리 버퍼 기능에 대한 매개 변수를 포함합니다.
NVME_CDW11_FEATURE_INTERRUPT_COALESCING 인터럽트 병합 설정을 구성하는 인터럽트 병합 기능에 대한 매개 변수를 포함합니다.

NVME_CDW11_FEATURE_INTERRUPT_VECTOR_CONFIG 특정 인터럽트 벡터와 관련된 설정을 구성하는 인터럽트 벡터 구성 기능에 대한 매개 변수를 포함합니다.
NVME_CDW11_FEATURE_LBA_RANGE_TYPE 기능 설정 명령에서 LBA 범위 유형 기능의 LBA 범위 수를 지정하는 매개 변수를 포함합니다.
NVME_CDW11_FEATURE_NON_OPERATIONAL_POWER_STATE 비작동 전원 상태에 대해 허용 모드를 사용할 수 있는지 여부를 나타내는 비작동 전원 상태 기능에 대한 매개 변수를 포함합니다.
NVME_CDW11_FEATURE_NUMBER_OF_QUEUES 호스트가 이 컨트롤러에 대해 요청하는 I/O 완료 큐 및 I/O 제출 큐 수를 나타내는 큐 수 기능에 대한 매개 변수를 포함합니다.
NVME_CDW11_FEATURE_POWER_MANAGEMENT 호스트가 전원 상태를 구성할 수 있도록 하는 전원 관리 기능에 대한 값을 포함합니다.
NVME_CDW11_FEATURE_SUPPORTED_CAPABILITY 지정된 기능의 지원되는 기능을 설명하는 기능 가져오기 명령에서 반환되는 값을 포함합니다.
NVME_CDW11_FEATURE_TEMPERATURE_THRESHOLD 온도 임계값 초과 및 최대 9개의 온도 값에 대한 온도 임계값 미만을 설정하는 데 사용되는 온도 임계값 기능에 대한 매개 변수를 포함합니다.
NVME_CDW11_FEATURE_VOLATILE_WRITE_CACHE 컨트롤러에서 지원되고 존재하는 경우 휘발성 쓰기 캐시를 제어하는 휘발성 쓰기 캐시 기능에 대한 매개 변수를 포함합니다.
NVME_CDW11_FEATURE_WRITE_ATOMICITY_NORMAL 원자성 작업에 대한 컨트롤러의 지원을 정의하는 **AWUN**(Atomic Write Unit Normal) 및 네임스페이스 **NAWUN**(Atomic Write Unit Normal) 매개 변수의 작업을 제어하는 쓰기 원자성 표준 기능에 대한 매개 변수를 포함합니다.
NVME_CDW11_FEATURES 지정된 기능의 특성을 검색하거나 설정하는 기능 가져오기 및 기능 설정 명령에 대한 매개 변수를 포함합니다.

NVME_CDW11_FIRMWARE_DOWNLOAD 새 펌웨어 이미지(전체 또는 부분적으로)를 컨트롤러에 복사하는 데 사용되는 펌웨어 이미지 다운로드 명령에 대한 매개 변수를 포함합니다.
NVME_CDW11_GET_LOG_PAGE NVME_CDW11_GET_LOG_PAGE 구조에는 요청된 로그 페이지가 포함된 데이터 버퍼를 반환하는 로그 페이지 가져오기 명령에 대한 매개 변수가 포함되어 있습니다.
NVME_CDW11_IDENTIFY Identify 명령에 대한 매개 변수를 포함합니다.
NVME_CDW11_SECURITY_RECEIVE 보안 수신 명령에 사용되는 매개 변수를 포함합니다.
NVME_CDW11_SECURITY_SEND 보안 보내기 명령에 사용되는 매개 변수를 포함합니다.
NVME_CDW12_DIRECTIVE_RECEIVE 지시문 수신 명령에 스트림 리소스를 할당하기 위한 매개 변수를 포함합니다.
NVME_CDW12_DIRECTIVE_RECEIVE_STREAMS_ALLOCATE_RESOURCES 지시문 수신 명령에서 스트림 리소스를 할당하는 데 사용되는 네임스페이스 스트림을 요청하기 위한 매개 변수를 포함합니다.
NVME_CDW12_DIRECTIVE_SEND 지시문 보내기 명령에 지시문을 사용하도록 설정하기 위한 매개 변수를 포함합니다.
NVME_CDW12_DIRECTIVE_SEND_IDENTIFY_ENABLE_DIRECTIVE 지시문 보내기 명령에서 지시문을 지정하고 사용하도록 설정하기 위한 매개 변수를 포함합니다.
NVME_CDW12_FEATURE_HOST_MEMORY_BUFFER 호스트 메모리 버퍼의 크기를 지정하는 호스트 메모리 버퍼 기능에 대한 매개 변수를 포함합니다.
NVME_CDW12_FEATURES 지정된 기능의 특성을 설정하는 기능 설정 명령에 대한 매개 변수를 포함합니다.
NVME_CDW12_GET_LOG_PAGE

NVME_CDW12_READ_WRITE 지정된 LBA(논리 블록 주소)에 대해 NVM 컨트롤러에서 데이터 및 메타데이터(해당하는 경우)를 읽거나 쓰는 NVME 읽기 및 NVME 쓰기 명령에 대한 매개 변수를 포함합니다.
NVME_CDW13_FEATURE_HOST_MEMORY_BUFFER 호스트 메모리 설명자 목록의 실제 위치의 하위 32비트를 지정하는 호스트 메모리 버퍼 기능에 대한 매개 변수를 포함합니다.
NVME_CDW13_FEATURES 지정된 기능의 특성을 설정하는 기능 설정 명령에 대한 매개 변수를 포함합니다.
NVME_CDW13_GET_LOG_PAGE
NVME_CDW13_READ_WRITE 지정된 LBA(논리 블록 주소)에 대해 NVM 컨트롤러에서 데이터 및 메타데이터(해당하는 경우)를 읽거나 쓰는 NVME 읽기 및 NVME 쓰기 명령에 대한 매개 변수를 포함합니다.
NVME_CDW14_FEATURE_HOST_MEMORY_BUFFER 호스트 메모리 설명자 목록의 실제 위치의 상위 32비트를 지정하는 호스트 메모리 버퍼 기능에 대한 매개 변수를 포함합니다.
NVME_CDW14_FEATURES NVME_CDW14_FEATURES 구조체에는 지정된 기능의 특성을 설정하는 기능 설정 명령에 대한 매개 변수가 포함되어 있습니다.
NVME_CDW15_FEATURE_HOST_MEMORY_BUFFER 호스트 메모리 설명자 목록의 항목 수를 지정하는 호스트 메모리 버퍼 기능에 대한 매개 변수를 포함합니다.
NVME_CDW15_FEATURES NVME_CDW15_FEATURES 구조체에는 지정된 기능의 특성을 설정하는 기능 설정 명령에 대한 매개 변수가 포함되어 있습니다.
NVME_CDW15_READ_WRITE 지정된 LBA(논리 블록 주소)에 대해 NVM 컨트롤러에서 데이터 및 메타데이터(해당하는 경우)를 읽거나 쓰는 NVME 읽기 및 NVME 쓰기 명령에 대한 매개 변수를 포함합니다.

NVME_CHANGED_NAMESPACE_LIST_LOG 로그 페이지를 마지막으로 읽은 이후 네임스페이스 식별 정보를 변경한 컨트롤러의 네임스페이스를 설명하는 변경된 네임스페이스 목록 로그 페이지에 대한 데이터를 포함합니다.
NVME_COMMAND 관리 명령 및 NVM 명령 집합의 모든 명령에 대한 매개 변수를 포함합니다.
NVME_COMMAND_DWORD0 모든 관리 명령 및 NVM 명령에 공통적인 매개 변수를 포함합니다.
NVME_COMMAND_EFFECTS_DATA 명령의 선택적 기능을 포함하여 관리 또는 I/O 명령의 전반적인 가능한 효과를 설명하는 정보를 포함합니다.
NVME_COMMAND_EFFECTS_LOG 컨트롤러가 지원하는 명령과 이러한 명령이 NVM 하위 시스템의 상태에 미치는 영향을 설명하는 정보를 포함합니다.
NVME_COMMAND_STATUS 명령의 상태에 대한 정보를 포함합니다.
NVME_COMPLETION_DW0_ASYNC_EVENT_REQUEST 완료 큐 항목의 DWord 0에서 관리 완료 큐에 게시되는 비동기 이벤트에 대한 정보를 포함합니다. 비동기 이벤트는 호스트 소프트웨어에 상태, 오류 및 상태 정보를 알리는 데 사용됩니다.
NVME_COMPLETION_DW0_DIRECTIVE_RECEIVE_STREAMS_ALLOCATE_RESOURCES 지시문 수신 명령의 할당된 스트림 리소스 수에 대한 정보를 포함합니다.
NVME_COMPLETION_ENTRY 완료 큐의 크기가 16바이트인 항목을 지정합니다.
NVME_COMPLETION_QUEUE_HEAD_DOORBELL 완료 큐 y에 대한 헤드 항목 포인터를 업데이트하는 초인종 레지스터를 정의 *합니다*.
NVME_CONTEXT_ATTRIBUTES LBA 범위(논리 블록 범위)에 대한 선택적 컨텍스트 특성을 지정합니다.

NVME_CONTROLLER_CAPABILITIES 소프트웨어를 호스트하는 컨트롤러의 기본 기능을 지정하는 읽기 전용 값을 포함합니다.
NVME_CONTROLLER_CONFIGURATION 컨트롤러에 대한 읽기/쓰기 구성 설정을 포함합니다.
NVME_CONTROLLER_LIST 컨트롤러 식별자의 순서가 지정된 목록을 포함합니다.
NVME_CONTROLLER_MEMORY_BUFFER_LOCATION NVME_CONTROLLER_REGISTERS 구조체의 **CMBLOC** 필드에 있는 선택적 컨트롤러 메모리 버퍼 위치 레지스터의 위치를 정의합니다.
NVME_CONTROLLER_MEMORY_BUFFER_SIZE 선택적 컨트롤러 메모리 버퍼 레지스터의 크기를 정의하고 NVME_CONTROLLER_REGISTERS 구조체의 **CMBSZ** 필드에 사용됩니다.
NVME_CONTROLLER_REGISTERS 컨트롤러에 대한 레지스터 맵을 지정합니다.
NVME_CONTROLLER_STATUS 컨트롤러 상태를 나타내는 값을 포함합니다.
NVME_DEVICE_SELF_TEST_LOG 디바이스 자체 테스트의 상태, 완료율 및 결과를 설명하는 디바이스 자체 테스트 로그 페이지의 정보를 지정하는 필드가 포함되어 있습니다.
NVME_DEVICE_SELF_TEST_RESULT_DATA 디바이스 Self-Test 작업의 결과에 대한 데이터를 포함합니다.
NVME_DIRECTIVE_IDENTIFY_RETURN_PARAMETERS Identify 지시문에 대한 반환 매개 변수를 설명하는 필드를 포함합니다.
NVME_DIRECTIVE_IDENTIFY_RETURN_PARAMETERS_DESCRIPTOR Identify 지시문에 대한 반환 매개 변수를 설명하는 값을 포함합니다.

NVME_DIRECTIVE_STREAMS_GET_STATUS_DATA 현재 열려 있는 스트림의 식별자를 포함합니다.
NVME_DIRECTIVE_STREAMS_RETURN_PARAMETERS Streams 지시문에 대한 반환 매개 변수를 포함합니다.
NVME_ENDURANCE_GROUP_LOG 지구력 그룹 정보 로그 페이지에서 지구력 그룹에서 읽고 쓰는 데이터의 양을 나타내는 정보를 지정하는 필드를 포함합니다.
NVME_ERROR_INFO_LOG 오류 정보 로그 페이지의 정보를 지정하는 필드가 포함되어 있습니다.
NVME_FIRMWARE_SLOT_INFO_LOG 펌웨어 슬롯 정보 로그 페이지의 정보를 지정하는 필드가 포함되어 있습니다.
NVME_HEALTH_INFO_LOG SMART/Health Information Log 페이지에 포함된 정보를 지정하는 필드가 포함되어 있습니다.
NVME_HOST_MEMORY_BUFFER_DESCRIPTOR_ENTRY 호스트 메모리 설명자 목록에서 단일 항목의 매개 변수를 정의합니다.
NVME_IDENTIFY_CONTROLLER_DATA 컨트롤러 기능, 기능, 명령 집합 특성 및 전원 상태 설명자를 나타내는 값을 포함합니다.
NVME_IDENTIFY_NAMESPACE_DATA 특정 네임스페이스와 관련된 기능 및 설정을 나타내는 값을 포함합니다.
NVME_LBA_FORMAT NVM 형식 명령의 일부로 NVM 미디어에 적용할 LBA 형식을 지정하는 매개 변수를 포함합니다.
NVME_LBA_RANGE 시작 LBA에서 지정한 연속 논리 블록의 컬렉션과 논리 블록 수를 정의하는 매개 변수를 포함합니다.

NVME_LBA_RANGET_TYPE_ENTRY 기능 설정 명령의 LBA 범위 유형 기능에 대한 LBA(논리 블록 주소) 범위 목록에서 단일 항목을 지정하는 매개 변수를 포함합니다.
NVME_NVM_SUBSYSTEM_RESET NVM 하위 시스템 재설정을 시작하는 기능을 호스트 소프트웨어에 제공하는 매개 변수를 지정합니다.
NVME_POWER_STATE_DESC 각 전원 상태의 특성을 설명하는 필드를 포함합니다.
NVME_PRP_ENTRY 실제 메모리 페이지에 대한 포인터의 PBAO(페이지 기준 주소 및 오프셋)를 지정하는 필드가 포함되어 있습니다.
NVME_RESERVATION_NOTIFICATION_LOG 예약 알림 로그 페이지의 정보를 지정하는 필드가 포함되어 있습니다.
NVME_SCSI_NAME_STRING SCSI 이름 문자열 식별자를 생성하는 데 사용되는 정보를 포함합니다.
NVME_SET_ATTRIBUTES_ENTRY 특성을 설정하기 위한 정보를 지정하는 필드가 포함되어 있습니다.
NVME_SUBMISSION_QUEUE_TAIL_DOORBELL 제출 큐 y에 대한 Tail 항목 포인터를 업데이트하는 초인종 레지스터를 정의 *합니다*.
NVME_TELEMETRY_HOST_INITIATED_LOG 원격 분석 Host-Initiated 로그 페이지에서 정보를 지정하는 필드를 포함합니다.
NVME_VERSION 컨트롤러 구현에서 지원하는 NVM Express 사양의 버전 번호를 지정하는 필드가 포함되어 있습니다.

열거형

NVME_ACCESS_FREQUENCIES LBA(논리 블록 주소 지정) 범위에 대한 읽기 및 쓰기 액세스 빈도를 나타내는 값을 정의합니다.
NVME_ACCESS_LATENCIES 읽기 및 쓰기 작업의 대기 시간을 나타내는 값을 정의합니다.
NVME_ADMIN_COMMANDS 관리 명령 집합에서 명령을 지정하는 값을 정의합니다. 관리 명령 집합에는 관리 제출 큐에 제출할 수 있는 명령이 포함되어 있습니다.
NVME_ASYNC_EVENT_ERROR_STATUS_CODES 일반적인 오류 이벤트 유형을 나타내는 값을 포함합니다.
NVME_ASYNC_EVENT_HEALTH_STATUS_CODES SMART/Health 상태 이벤트 유형을 나타내는 값을 포함합니다.
NVME_ASYNC_EVENT_IO_COMMAND_SET_STATUS_CODES I/O 명령 집합 이벤트 유형을 나타내는 값을 포함합니다.
NVME_ASYNC_EVENT_NOTICE_CODES 알림 이벤트 유형을 나타내는 값을 포함합니다.
NVME_ASYNC_EVENT_TYPES 비동기 이벤트 유형을 나타내는 값을 포함합니다.
NVME_CC_SHN_SHUTDOWN_NOTIFICATIONS CC(컨트롤러 구성) 종료 알림을 나타내는 값을 포함합니다.
NVME_CMBSZ_SIZE_UNITS 컨트롤러 메모리 버퍼의 크기를 나타내는 크기 단위를 지정하는 값을 포함합니다.
NVME_COMMAND_EFFECT_SBUMISSION_EXECUTION_LIMITS 연결된 명령에 대한 명령 제출 및 실행 권장 사항을 나타내는 값을 포함합니다.
NVME_CSTS_SHST_SHUTDOWN_STATUS NVME_CONTROLLER_CONFIGURATION 구조에서 **SHN** 필드를 설정하는 호스트에 의해 시작된 종료 처리의 상태를 나타내는 값을 포함합니다.

NVME_DIRECTIVE_RECEIVE_IDENTIFY_OPERATIONS 식별 작업에 대한 지시문 형식을 지정하는 값을 포함합니다.
NVME_DIRECTIVE_RECEIVE_STREAMS_OPERATIONS Streams 작업에 대한 지시문 형식을 나타내는 값을 포함합니다.
NVME_DIRECTIVE_SEND_IDENTIFY_OPERATIONS 식별 작업으로 전송되는 지시문 형식을 지정하는 값을 포함합니다.
NVME_DIRECTIVE_SEND_STREAMS_OPERATIONS Streams 작업으로 전송된 지시문 형식을 나타내는 값을 포함합니다.
NVME_DIRECTIVE_TYPES 지시문의 형식을 나타내는 값을 포함합니다.
NVME_FEATURE_VALUE_CODES **NVME_ADMIN_COMMAND_GET_FEATURES 및 NVME_ADMIN_COMMAND_SET_FEATURES 관리** 명령으로 검색하거나 구성해야 하는 기능 값을 나타내는 값을 포함합니다.
NVME_FEATURES NVME_ADMIN_COMMAND_GET_FEATURES 검색하거나 구성해야 하는 기능을 나타내고 **관리** 명령을 **NVME_ADMIN_COMMAND_SET_FEATURES** 값을 포함합니다.
NVME_FIRMWARE_ACTIVATE_ACTIONS 펌웨어 이미지 다운로드 명령 또는 이전에 다운로드하고 배치한 이미지에서 다운로드한 이미지에서 수행되는 작업을 나타내는 값을 포함합니다.
NVME_FUSED_OPERATION_CODES 명령이 융합 작업의 첫 번째 또는 두 번째 명령인지 여부를 나타내는 값을 포함합니다.
NVME_IDENTIFY_CNS_CODES NVME_CDW10_IDENTIFY 구조체의 컨트롤러 또는 CNS(네임스페이스 구조) 멤버에서 반환될 컨트롤러 또는 네임스페이스 정보의 유형을 나타내는 값을 포함합니다.
NVME_LBA_RANGE_TYPES NVME_LBA_RANGET_TYPE_ENTRY 구조체의 LBA(논리 블록 주소 지정) 범위 유형을 나타내는 값을 포함합니다.

NVME_LOG_PAGES 로그 페이지 가져오기 NVME_ADMIN_COMMAND_GET_LOG_PAGE **관리** 명령에서 검색할 수 있는 로그 페이지를 나타내는 값을 포함합니다.
NVME_NVM_COMMANDS NVM 명령 집합의 명령을 나타내는 값을 포함합니다.
NVME_NVM_QUEUE_PRIORITIES 컨트롤러에서 지원하는 경우 중재 메커니즘에서 고려할 I/O 제출 큐에 할당할 수 있는 우선 순위를 나타내는 값을 포함합니다.
NVME_PROTECTION_INFORMATION_TYPES 엔드 투 엔드 데이터 보호를 사용할 수 있는지 여부를 나타내는 값을 포함하며, 이 경우 보호 정보 유형을 지정합니다.
NVME_RESERVATION_NOTIFICATION_TYPES 예약 알림 로그 페이지의 예약 알림 유형을 나타내는 값을 포함합니다.
NVME_SECURE_ERASE_SETTINGS NVM 형식 명령의 일부로 보안 지우기 작업의 형식을 수행할지 여부를 지정하는 값을 포함합니다.
NVME_STATUS_COMMAND_SPECIFIC_CODES 특정 명령과 관련된 상태를 나타내는 값을 포함합니다.
NVME_STATUS_GENERIC_COMMAND_CODES 다양한 유형의 명령에서 제네릭인 명령과 연결된 상태를 나타내는 값을 포함합니다.
NVME_STATUS_MEDIA_ERROR_CODES NVM 미디어와 연결된 오류를 나타내거나 데이터 무결성 유형 오류를 나타내는 값을 포함합니다.
NVME_STATUS_TYPES 명령이 완료될 때 컨트롤러가 완료 큐 항목에 게시하는 상태 코드의 유형을 나타내는 값을 포함합니다.
NVME_TEMPERATURE_THRESHOLD_TYPES 전체 디바이스(컨트롤러 및 NVM 포함)의 온도에 대한 임계값 유형을 나타내는 값을 포함합니다.

oaidl.h 헤더

아티클2023. 08. 25.

이 헤더는 Automation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Automation

oaidl.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICreateErrorInfo 오류 정보를 반환합니다.
ICreateTypeInfo 형식 설명을 통해 정의된 형식 정보를 만들고 관리하기 위한 도구를 제공합니다. (ICreateTypeInfo)
ICreateTypeInfo2 형식 설명을 통해 정의된 형식 정보를 만들고 관리하기 위한 도구를 제공합니다. (ICreateTypeInfo2)
ICreateTypeLib 형식 정보를 포함하는 구성 요소 또는 파일을 만들고 관리하는 메서드를 제공합니다. (ICreateTypeLib)
ICreateTypeLib2 형식 정보를 포함하는 구성 요소 또는 파일을 만들고 관리하는 메서드를 제공합니다. (ICreateTypeLib2)
IDispatch 자동화를 지원하는 프로그래밍 도구 및 기타 애플리케이션에 개체, 메서드 및 속성을 노출합니다.
IEnumVARIANT 다른 유형의 개체 및 내장 형식 컬렉션을 포함하여 변형 컬렉션을 열거하는 메서드를 제공합니다.
IErrorInfo 자세한 상황별 오류 정보를 제공합니다.

IErrorLog 클라이언트와 개체 간에 자세한 오류 정보를 전달합니다.
IPropertyBag 개체가 속성을 영구적으로 저장할 수 있는 속성 모음을 개체에 제공합니다.
IRecordInfo 특정 UDT의 구조를 설명합니다.
ISupportErrorInfo 오류 정보가 호출 체인으로 올바르게 전파될 수 있는지 확인합니다. 오류 처리 인터페이스를 사용하는 Automation 개체는 ISupportErrorInfo를 구현해야 합니다.
ITypeChangeEvents 클라이언트가 ITypeInfo, ITypeInfo2, ICreateTypeInfo 및 ICreateTypeInfo2 인터페이스를 구현하는 개체에 대한 형식 변경 알림을 구독할 수 있습니다.
Itypecomp ITypeComp 인터페이스는 구조체 및 인터페이스에 바인딩하고 인스턴스화할 때 컴파일러가 필요로 하는 정보에 빠르게 액세스할 수 있는 방법을 제공합니다.
Itypeinfo 개체에 대한 정보를 읽는 데 사용됩니다. (ITypeInfo)
ITypeInfo2 개체에 대한 정보를 읽는 데 사용됩니다. (ITypeInfo2)
ITypeLib 개체 집합을 설명하는 데이터인 형식 라이브러리를 나타냅니다. (ITypeLib)
ITypeLib2 개체 집합을 설명하는 데이터인 형식 라이브러리를 나타냅니다. (ITypeLib2)

Functions

BSTR_UserFree RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다. (BSTR_UserFree)
BSTR_UserFree64 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다. (BSTR_UserFree64)
BSTR_UserMarshal BSTR 개체를 RPC 버퍼로 마샬링합니다. (BSTR_UserMarshal)
BSTR_UserMarshal64 BSTR 개체를 RPC 버퍼로 마샬링합니다. (BSTR_UserMarshal64)
BSTR_UserSize BSTR 개체의 와이어 크기를 계산하고 해당 핸들과 데이터를 가져옵니다. (BSTR_UserSize)
BSTR_UserSize64 BSTR 개체의 와이어 크기를 계산하고 해당 핸들과 데이터를 가져옵니다. (BSTR_UserSize64)
BSTR_UserUnmarshal RPC 버퍼에서 BSTR 개체를 언마샬링합니다. (BSTR_UserUnmarshal)
BSTR_UserUnmarshal64 RPC 버퍼에서 BSTR 개체를 언마샬링합니다. (BSTR_UserUnmarshal64)
VARIANT_UserFree RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다. (VARIANT_UserFree)
VARIANT_UserFree64 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다. (VARIANT_UserFree64)
VARIANT_UserMarshal VARIANT 개체를 RPC 버퍼로 마샬링합니다. (VARIANT_UserMarshal)
VARIANT_UserMarshal64 VARIANT 개체를 RPC 버퍼로 마샬링합니다. (VARIANT_UserMarshal64)
VARIANT_UserSize

VARIANT 개체의 와이어 크기를 계산하고 해당 핸들과 데이터를 가져옵니다. (VARIANT_UserSize)
VARIANT_UserSize64 VARIANT 개체의 와이어 크기를 계산하고 해당 핸들과 데이터를 가져옵니다. (VARIANT_UserSize64)
VARIANT_UserUnmarshal RPC 버퍼에서 VARIANT 개체를 언마샬링합니다. (VARIANT_UserUnmarshal)
VARIANT_UserUnmarshal64 RPC 버퍼에서 VARIANT 개체를 숨기지 않습니다. (VARIANT_UserUnmarshal64)

구조체

ARRAYDESC 배열, 해당 요소 형식 및 해당 차원에 대해 설명합니다.
BINDPTR 포인터를 설명합니다.
CUSTDATA 사용자 지정 데이터를 나타냅니다.
CUSTDATAITEM 사용자 지정 데이터 항목을 나타냅니다.
DISPPARAMS 메서드 또는 속성에 전달된 인수를 포함합니다.
ELEMDESC ELEMDESC 구조체에는 변수, 함수 또는 함수 매개 변수에 대한 형식 설명 및 프로세스 전송 정보가 포함됩니다. (ELEMDESC)
EXCEPINFO IDispatch::Invoke 중에 발생한 예외에 대해 설명합니다.

FUNCDESC 함수를 설명합니다. (FUNCDESC)
PARAMDESC 구조체 요소, 매개 변수, 프로세스간 함수 반환 값을 전달하는데 필요한 정보가 들어 있습니다.
PARAMDESCEX 매개 변수의 기본값에 대한 정보를 포함합니다.
Safearray 안전한 배열을 나타냅니다.
SAFEARRAYBOUND 배열의 한 차원의 범위를 나타냅니다.
TLIBATTR 형식 라이브러리에 대한 정보를 포함합니다. 이 구조의 정보는 형식 라이브러리를 식별하고 멤버 이름에 대한 국가 언어 지원을 제공하는 데 사용됩니다.
TYPEATTR 형식의 특성을 포함합니다.
TYPEDESC 변수의 형식, 함수의 반환 형식 또는 함수 매개 변수의 형식에 대해 설명합니다.
VARDESC 변수, 상수 또는 데이터 멤버를 설명합니다.
변형 VARIANTARG는 DISPPARAMS 내에서 전달된 인수와 참조로 전달할 수 없는 변형 데이터를 지정하는 VARIANT에 대해 설명합니다.

열거형

CALLCONV METHODDATA 구조에 설명된 멤버 함수에서 사용하는 호출 규칙을 식별합니다.
DESCKIND 바인딩되는 형식 설명을 식별합니다.
FUNCFLAGS 함수 플래그를 지정합니다.
FUNCKIND 함수 형식을 지정합니다.
INVOKEKIND 함수가 호출되는 방식을 지정합니다.
LIBFLAGS 형식 라이브러리에 해당하는 플래그를 정의합니다.
SYSKIND 대상 운영 체제 플랫폼을 식별합니다.
TYPEFLAGS 형식 플래그입니다.
TYPEKIND 형식을 지정합니다.
VARFLAGS 변수 플래그를 지정합니다.
VARKIND 변수 형식을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

objbase.h 헤더

아티클2023. 08. 25.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)
- 개발자 정보
- 구조적 저장소

objbase.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

BindMoniker 모니커를 통해 개체를 찾고, 비활성 상태이면 개체를 활성화하고, 해당 개체의 지정된 인터페이스에 대한 포인터를 검색합니다.
CLSIDFromProgIDEx COMClassStore 정책을 사용하도록 설정하면 CLSIDFromProgIDEx 함수(objbase.h)가 자동 설치를 트리거합니다.
CoAllowSetForegroundWindow 이 함수는 전경 권한(포그라운드 창을 설정하는 권한)을 한 프로세스에서 다른 프로세스로 전달합니다. 포그라운드 권한이 있는 프로세스는 이 함수를 호출하여 해당 권한을 로컬 COM 서버 프로세스에 전달할 수 있습니다.
CoDosDateTimeToFileTime 시간과 날짜의 MS-DOS 표현을 Windows에서 사용하는 FILETIME 구조로 변환합니다.
CoFileTimeNow CoFileTimeNow 함수(objbase.h)는 현재 시간을 FILETIME 구조체로 반환합니다.
CoFileTimeToDosDateTime FILETIME을 MS-DOS 날짜 및 시간 값으로 변환합니다.
CoFreeAllLibraries

CoLoadLibrary 함수(CoGetClassObject에서 내부적으로 호출)를 사용하여 로드된 모든 DLL을 현재 사용 중인지 여부에 관계없이 해제합니다.
CoFreeLibrary 로드될 때 명시적으로 해제되도록 지정된 라이브러리를 해제합니다.
CoGetInstanceFromFile 새 개체를 만들고 IPersistFile::Load를 사용하여 파일에서 초기화합니다.
CoGetInstanceFromIStorage 새 개체를 만들고 IPersistFile::Load에 대한 내부 호출을 통해 스토리지 개체에서 초기화합니다.
CoGetObject 표시 이름을 라는 개체를 식별하는 모니커로 변환한 다음 모니커로 식별된 개체에 바인딩합니다.
CoGetSystemSecurityPermissions 머신 전체 시작 및 액세스 권한의 보안 설명자의 기본값과 시작 및 액세스 제한을 반환합니다.
CoInitialize 현재 스레드에서 COM 라이브러리를 초기화하고 동시성 모델을 STA(단일 스레드 아파트)로 식별합니다.
CoInstall 요청된 COM 서버 애플리케이션을 설치합니다.
CoIsOle1Class 지정된 CLSID가 OLE 1 개체를 나타내는지 여부를 확인합니다.
CoLoadLibrary 호출자의 프로세스에 특정 DLL을 로드합니다.
CoRegisterChannelHook 채널 후크를 등록합니다.
CoRegisterInitializeSpy IInitializeSpy 인터페이스의 구현을 등록합니다. 개발자가 COM 아파트에서 초기화 및 정리를 수행할 수 있도록 IInitializeSpy 인터페이스는 무시됩니다.

CoRegisterMallocSpy IMallocSpy 인터페이스의 구현을 등록한 후 OLE가 해당 IMalloc 메서드를 호출할 때마다 래퍼 메서드를 호출하도록 요구합니다.
CoRegisterMessageFilter 현재 스레드에서 동시성 문제를 처리하는 데 사용되는 IMessageFilter 인터페이스의 instance OLE에 등록합니다.
CoRevokeInitializeSpy IInitializeSpy 인터페이스의 등록된 구현을 취소합니다.
CoRevokeMallocSpy 등록된 IMallocSpy 개체를 취소합니다.
CoTreatAsClass 한 클래스의 개체가 다른 클래스의 개체로 처리되는 에뮬레이션을 설정하거나 제거합니다.
CreateAntiMoniker 새 안티 모니커를 만들고 반환합니다.
CreateBindCtx IBindCtx(바인딩 컨텍스트 개체)의 구현에 대한 포인터를 반환합니다. 이 개체는 특정 모니커 바인딩 작업에 대한 정보를 저장합니다.
CreateClassMoniker 지정된 클래스를 참조하는 클래스 모니커를 만듭니다.
CreateDataAdviseHolder CreateDataAdviseHolder 함수(objbase.h)는 데이터 조언 소유자 개체에서 IDataAdviseHolder의 OLE 구현에 대한 포인터를 검색합니다.
CreateDataCache 데이터 캐시의 OLE 제공 구현의 새 instance 대한 포인터를 검색합니다.
CreateFileMoniker 지정된 경로를 기반으로 파일 모니커를 만듭니다.

CreateGenericComposite 두 모니커의 제네릭 컴퍼지션을 수행하고 결과 복합 모니커에 대한 포인터를 제공합니다.
CreateItemMoniker 포함하는 개체(일반적으로 복합 문서) 내에서 개체를 식별하는 항목 모니커를 만듭니다.
CreateObjrefMoniker 개체에 대한 포인터를 기반으로 OBJREF 모니커를 만듭니다.
CreatePointerMoniker 개체에 대한 포인터를 기반으로 포인터 모니커를 만듭니다.
GetClassFile 지정된 파일 이름과 연결된 CLSID를 반환합니다.
GetRunningObjectTable ROT(로컬 실행 개체 테이블)의 IRunningObjectTable 인터페이스에 대한 포인터를 반환합니다.
IsEqualGUID IsEqualGUID 매크로(objbase.h)는 두 GUID가 같은지 여부를 결정합니다.
MkParseDisplayName 문자열을 문자열로 명명된 개체를 식별하는 모니커로 변환합니다.
MonikerCommonPrefixWith 이 모니커(이 모니커 개체의 데이터로 구성된 모니커)가 다른 모니커와 공유하는 공통 접두사를 기반으로 새 모니커를 만듭니다.
MonikerRelativePathTo 첫 번째 지정된 모니커(또는 구조가 비슷한 모니커)의 끝에 구성될 때 두 번째 지정된 모니커를 생성하는 모니커를 제공합니다.
StgGetIFillLockBytesOnFile 임시 파일에서 래퍼 개체를 엽니다.
StgGetIFillLockBytesOnILockBytes 호출자가 제공하는 바이트 배열 개체에 새 래퍼 개체를 만듭니다.

StgOpenAsyncDocfileOnIFillLockBytes 호출자가 제공하는 바이트 배열 래퍼 개체에서 기존 루트 비동기 스토리지 개체를 엽니다.
StgOpenLayoutDocfile 섹터 데이터를 모니터링할 수 있는 ILockBytes 구현에서 복합 파일을 엽니다.

열거형

COINIT 이 스레드에서 만든 개체에 대한 수신 호출에 사용되는 동시성 모델을 결정합니다. 이 동시성 모델은 아파트 스레드 또는 다중 스레드일 수 있습니다.
COMSD CoGetSystemSecurityPermissions를 호출할 때 가져올 COM 보안 설명자의 유형을 결정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

objectarray.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

objectarray.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IObjectArray 클라이언트가 IUnknown을 지원하는 개체 컬렉션의 항목에 액세스할 수 있도록 하는 메서드를 노출합니다.
IObjectCollection 클라이언트가 컬렉션에서 IUnknown을 지원하는 개체를 추가하고 제거할 수 있는 메서드를 제공하여 IObjectArray 인터페이스를 확장합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

objidl.h 헤더

아티클2023. 03. 16.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)
- 구조적 저장소
- Windows 런타임 C++ 참조

objidl.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAdviseSink 컨테이너 및 기타 개체가 관심 있는 개체에서 발생하는 데이터 변경 내용, 보기 변경 내용 및 복합 문서 변경에 대한 알림을 받을 수 있도록 합니다.
IAdviseSink2 IAdviseSink2 인터페이스는 IAdviseSink 인터페이스의 확장이며, 연결된 개체의 모니커 변경을 처리하기 위해 OnLinkSrcChange 메서드를 계약에 추가합니다.
IAgileObject IAgileObject 인터페이스(objidl.h)는 아파트 전체에서 인터페이스를 agile로 표시합니다.
IAgileReference 개체에 대한 Agile 참조를 검색할 수 있습니다.
IApartmentShutdown 아파트 종료 알림 처리기의 등록을 사용하도록 설정합니다.
IBindCtx 특정 모니커 바인딩 작업에 대한 정보를 저장하는 개체인 바인딩 컨텍스트에 대한 액세스를 제공합니다.
IBlockingLock 파일과 같은 공유 리소스에 일시적으로 단독 액세스를 제공하는 데 사용할 수 있는 세마포를 제공합니다.

ICallFactory ICallFactory 인터페이스(objidl.h)는 비동기 인터페이스의 메서드에 대한 호출을 처리하기 위한 호출 개체를 만듭니다.
ICancelMethodCalls ICancelMethodCalls::Cancel 메서드(objidl.h)는 아웃바운드 메서드 호출에 대한 취소 요청을 관리하고 서버 스레드에서 해당 메서드 호출의 현재 상태를 모니터링합니다.
IClassActivator 클래스 개체를 검색하는 메서드를 지정합니다.
IClientSecurity 개체의 각 개별 인터페이스 프록시에 대한 보안 설정을 클라이언트에 제어할 수 있습니다.
IComThreadingInfo IComThreadingInfo 인터페이스(objidl.h)를 사용하면 호출자가 실행 중인 아파트 및 스레드에 대한 아파트 유형, 스레드 유형 및 스레드 GUID를 가져올 수 있습니다.
IContext IContext 인터페이스(objidl.h)는 COM+ 컨텍스트 속성 설정을 지원합니다.
IDataAdviseHolder 데이터 개체와 하나 이상의 조언 싱크 간의 권고 연결을 만들고 관리합니다.
Idataobject 데이터 전송 및 데이터 변경 내용 알림을 사용하도록 설정합니다.
IDirectWriterLock IDirectWriterLock 인터페이스를 사용하면 단일 작성기가 직접 모드로 열린 루트 스토리지 개체에 대한 단독 쓰기 액세스 권한을 얻는 동시에 여러 판독기에서 동시에 액세스할 수 있습니다.
IEnumContextProps IEnumContextProps 인터페이스(objidl.h)는 COM+ 개체 컨텍스트와 연결된 컨텍스트 속성을 열거하는 메커니즘을 제공합니다.
IEnumFORMATETC 지정된 데이터 개체에서 지원하는 형식 및 미디어를 정의하는 FORMATETC 구조를 열거합니다.

IEnumMoniker 모니커 테이블의 모니커 또는 모니커 구성 요소를 열거합니다.
IEnumSTATDATA 데이터 개체에 대한 권고 연결 정보를 열거합니다.
IEnumSTATSTG STATSTG 구조체의 배열을 열거합니다.
IEnumString IEnumString 인터페이스(objidl.h)는 문자열을 열거합니다. LPWSTR은 와이드 또는 유니코드 문자의 0으로 끝나는 문자열에 대한 포인터를 나타내는 형식입니다.
IEnumUnknown IEnumUnknown 인터페이스(objidl.h)는 IUnknown 인터페이스를 사용하여 개체를 열거합니다.
IExternalConnection IExternalConnection 인터페이스(objidl.h)는 서버 개체의 마샬링된 연결 및 외부 연결 수를 관리합니다.
IFastRundown IFastRundown 인터페이스(objidl.h)는 인터페이스를 빠른 런다운 동작에 적합한 것으로 표시합니다.
IFillLockBytes IFillLockBytes 인터페이스를 사용하면 코드를 다운로드하여 구조화된 스토리지 바이트 배열에 비동기적으로 데이터를 쓸 수 있습니다.
IForegroundTransfer 전경 창을 COM 서버를 호스트하는 프로세스로 전송합니다.
IGlobalInterfaceTable 프로세스의 모든 아파트가 프로세스의 다른 아파트의 개체에 구현된 인터페이스에 액세스할 수 있도록 합니다.
IGlobalOptions IGlobalOptions 인터페이스(objidl.h)는 COM(구성 요소 개체 모델) 런타임의 전역 속성을 설정하고 쿼리합니다.

IInitializeSpy COM 아파트에 들어오거나 나갈 때 초기화 또는 정리를 수행합니다.
IInternalUnknown IInternalUnknown 인터페이스(objidl.h)는 프록시의 내부 인터페이스에 액세스해야 하는 경량 클라이언트 쪽 처리기에서만 사용됩니다.
ILayoutStorage ILayoutStorage 인터페이스를 사용하면 애플리케이션이 느린 링크를 통해 효율적으로 다운로드할 수 있도록 복합 파일의 레이아웃을 최적화할 수 있습니다.
ILockBytes ILockBytes 인터페이스는 디스크 파일, 전역 메모리 또는 데이터베이스와 같은 일부 물리적 스토리지에서 지원되는 바이트 배열 개체에 구현됩니다.
IMalloc IMalloc 인터페이스(objidl.h)는 메모리를 할당, 해제 및 관리합니다.
IMallocSpy 애플리케이션 개발자가 메모리 할당을 모니터링(감시)하고, 메모리 누수 검색, IMalloc 메서드 호출에서 메모리 오류를 시뮬레이션할 수 있습니다.
IMarshal IMarshal 인터페이스(objidl.h)를 사용하면 COM 개체가 인터페이스 포인터의 마샬링을 정의하고 관리할 수 있습니다.
IMarshalingStream IMarshalingStream 인터페이스(objidl.h)는 사용자 지정 마샬링된 개체 및 unmarshalers에 대한 마샬링 컨텍스트에 대한 추가 정보를 제공합니다.
IMessageFilter 동기 호출의 응답을 기다리는 동안 수신 및 발신 COM 메시지를 선택적으로 처리할 수 있는 기능을 COM 서버 및 애플리케이션에 제공합니다.
Imoniker COM 개체를 고유하게 식별하는 정보가 포함된 모니커 개체를 사용할 수 있습니다.

IMultiQI IMultiQI 인터페이스(objidl.h)를 사용하면 클라이언트가 단일 RPC 호출을 사용하여 여러 인터페이스에 대한 개체 프록시 또는 처리기를 쿼리할 수 있습니다.
INoMarshal INoMarshal 인터페이스(objidl.h)는 전역 인터페이스 테이블에 마샬링되거나 저장되는 것을 지원하지 않는 개체를 표시합니다.
IObjContext IObjContext 인터페이스(objidl.h)는 컨텍스트에서 다양한 작업을 수행합니다.
IPersist 시스템에 영구적으로 저장할 수 있는 개체의 CLSID를 제공합니다. 개체가 마샬링의 기본 구현에 사용되므로 클라이언트 프로세스에서 사용할 개체 처리기를 지정할 수 있습니다.
IPersistFile 스토리지 개체 또는 스트림이 아닌 디스크 파일에서 개체를 로드하거나 디스크 파일에 저장할 수 있도록 합니다.
IPersistStorage 컨테이너 애플리케이션이 포함된 개체 중 하나에 스토리지 개체를 전달하고 스토리지 개체를 로드하고 저장할 수 있도록 합니다.
IPersistStream 스토리지 요구 사항에 맞게 간단한 직렬 스트림을 사용하는 개체를 저장하고 로드할 수 있습니다.
IPipeByte IPipeByte 인터페이스(objidl.h)는 8비트 너비의 바이트 형식의 데이터를 전송합니다.
IPipeDouble IPipeDouble 인터페이스(objidl.h)는 너비가 64비트인 double 형식의 데이터를 전송합니다.
IPipeLong IPipeLong 인터페이스(objidl.h)는 너비가 32비트인 긴 정수 형식의 데이터를 전송합니다.
IProcessInitControl IProcessInitControl 인터페이스(objidl.h)는 프로세스 초기화 시간 제한 간격을 지정합니다.

IProcessLock ISurrogateService에서 시간 제한으로 인해 프로세스가 종료되지 않도록 하는 데 사용됩니다.
IProgressNotify 애플리케이션 및 기타 개체가 다운로드 작업 진행 중 변경 내용에 대한 알림을 받을 수 있도록 합니다.
IPSFactoryBuffer IPSFactoryBuffer 인터페이스(objidl.h)는 COM 개체 프록시 및 스텁을 만들기 위한 사용자 지정 메서드를 제공합니다. 이 인터페이스는 마샬링할 수 없습니다.
IRootStorage IRootStorage 인터페이스에는 스토리지 개체를 다른 기본 파일로 전환하고 스토리지 개체를 해당 파일에 저장하는 단일 메서드가 포함되어 있습니다.
IROTData 모니커에 의해 구현되어 ROT(실행 중인 개체 테이블)가 모니커를 서로 비교할 수 있도록 합니다.
IRpcChannelBuffer IRpcChannelBuffer 인터페이스(objidl.h)는 COM 클라이언트 프록시와 COM 서버 스텁 간에 데이터를 마샬링합니다.
IRpcOptions IRpcOptions 인터페이스(objidl.h)를 사용하면 호출자가 COM이 RPC(원격 프로시저 호출)를 처리하는 방법을 제어하는 다양한 속성의 값을 설정하거나 쿼리할 수 있습니다.
IRpcProxyBuffer IRpcProxyBuffer(objidl.h)는 COM 구성 요소 간에 데이터를 마샬링하는 데 사용되는 RPC 프록시를 제어합니다.
IRpcStubBuffer IRpcStubBuffer 인터페이스(objidl.h)는 COM 구성 요소 간에 데이터를 마샬링하는 데 사용되는 RPC 스텁을 제어합니다.
IRunnableObject 컨테이너가 포함된 개체의 실행을 제어할 수 있도록 합니다.

IRunningObjectTable 각 워크스테이션에서 전역적으로 액세스할 수 있는 조회 테이블인 ROT(실행 중인 개체 테이블)에 대한 액세스를 관리합니다.
ISequentialStream ISequentialStream 인터페이스는 스트림 개체에 대한 간소화된 순차적 액세스를 지원합니다. IStream 인터페이스는 ISequentialStream에서 읽기 및 쓰기 메서드를 상속합니다.
IServerSecurity IServerSecurity(objidl.h) 인터페이스는 서버에서 클라이언트를 인증하고 클라이언트의 가장을 관리하는 데 사용됩니다.
IStdMarshalInfo IStdMarshalInfo(objidl.h) 인터페이스는 표준 마샬링 중에 대상 프로세스에서 사용할 처리기를 식별하는 CLSID를 검색합니다.
IStorage IStorage 인터페이스는 구조적 스토리지 개체의 생성 및 관리를 지원합니다.
IStream IStream 인터페이스를 사용하면 데이터를 읽고 작성하여 개체를 스트리밍할 수 있습니다.
ISurrogate ISurrogate(objidl.h) 인터페이스는 새 DLL 서버를 기존 서로게이트에 동적으로 로드하고 더 이상 필요하지 않은 경우 서로게이트를 해제하는 데 사용됩니다.
ISurrogateService COM+ 애플리케이션을 초기화, 시작 및 릴리스하는 데 사용됩니다. 카탈로그를 새로 고치고 프로세스를 종료할 수도 있습니다.
ISynchronize ISynchronize(objidl.h) 인터페이스는 이벤트 발생에 대한 개체 간의 비동기 통신을 제공합니다.
ISynchronizeContainer ISynchronizeContainer(objidl.h) 인터페이스는 서명되지 않은 동기화 개체 그룹을 관리합니다.
ISynchronizeEvent ISynchronizeEvent(objidl.h) 인터페이스는 동기화 개체에 이벤트 핸들을 할당합니다.

ISynchronizeHandle ISynchronizeHandle(objidl.h) 인터페이스는 동기화 개체와 연결된 핸들을 검색합니다.

Functions

HBITMAP_UserFree RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HBITMAP_UserFree64 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HBITMAP_UserMarshal HBITMAP 개체를 RPC 버퍼로 마샬링합니다.
HBITMAP_UserMarshal64 HBITMAP 개체를 RPC 버퍼로 마샬링합니다.
HBITMAP_UserSize HBITMAP 개체의 와이어 크기를 계산하고 해당 핸들과 데이터를 가져옵니다.
HBITMAP_UserSize64 HBITMAP 개체의 와이어 크기를 계산하고 해당 핸들과 데이터를 가져옵니다.
HBITMAP_UserUnmarshal RPC 버퍼에서 HBITMAP 개체의 숨을 풉니다.
HBITMAP_UserUnmarshal64 RPC 버퍼에서 HBITMAP 개체의 숨을 풉니다.

구조체

BIND_OPTS 모니커 바인딩 작업 중에 사용되는 매개 변수를 포함합니다.
BIND_OPTS2 BIND_OPTS2 구조체에는 모니커 바인딩 작업 중에 사용되는 매개 변수가 포함되어 있습니다.
BIND_OPTS3 BIND_OPTS3 구조체에는 모니커 바인딩 작업 중에 사용되는 매개 변수가 포함됩니다.
COSERVERINFO COSERVERINFO(objidl.h) 구조는 활성화 함수에 대한 원격 컴퓨터 리소스를 식별합니다.
DVTARGETDEVICE 데이터를 구성하는 대상 디바이스에 대한 정보를 지정합니다. DVTARGETDEVICE에는 Windows 대상 디바이스에 대한 충분한 정보가 포함되어 있으므로 CreateDC 함수를 사용하여 HDC(디바이스 컨텍스트)에 대한 핸들을 만들 수 있습니다.
Formatetc 일반화된 클립보드 형식을 나타냅니다.
INTERFACEINFO 들어오는 호출에 대한 정보를 포함합니다.
MULTI_QI MULTI_QI(objidl.h) 구조체는 여러 인터페이스에 대한 쿼리의 인터페이스를 나타냅니다.
RemSNB RemSNB 구조체는 SNB 데이터 형식을 마샬링하는 데 사용됩니다. IStorage 인터페이스(Storag.idl)에 정의됩니다.
RPCOLEMESSAGE RPCOLEMESSAGE(objidl.h) 구조체에는 호출 인수 마샬링 및 COM 구성 요소 간의 반환 값이 포함됩니다.
SOLE_AUTHENTICATION_INFO SOLE_AUTHENTICATION_INFO(objidl.h) 구조는 인증 서비스, 권한 부여 서비스 및 지정된 서비스에 대한 정보를 식별합니다.

SOLE_AUTHENTICATION_LIST SOLE_AUTHENTICATION_LIST(objidl.h) 구조는 각 인증 서비스에 사용할 기본 인증 정보를 나타냅니다.
SOLE_AUTHENTICATION_SERVICE SOLE_AUTHENTICATION_SERVICE(objidl.h) 구조는 서버가 클라이언트와 통신하는 데 사용할 인증 서비스를 식별합니다.
STATDATA 각 권고 연결을 지정하는 데 사용되는 정보를 포함합니다.
Statstg 열린 스토리지, 스트림 또는 바이트 배열 개체에 대한 통계 데이터를 포함합니다.
StorageLayout 이름, 위치 및 길이를 포함한 단일 데이터 블록에 대해 설명합니다.
uSTGMEDIUM uSTGMEDIUM 구조체는 IAdviseSink, IDataObject 및 IOleCache 인터페이스의 데이터 전송 작업에 사용되는 일반화된 전역 메모리 핸들을 나타냅니다.

열거형

ACTIVATIONTYPE ACTIVATIONTYPE 열거형(objidl.h)은 활성화 가능한 클래스에 대한 활성화 종류를 지정합니다.
ADVF 데이터 변경 내용의 캐싱 및 알림을 제어하는 플래그입니다.
APTTYPE APTTYPE 열거형(objidl.h)은 다양한 유형의 아파트를 지정합니다.
APTTYPEQUALIFIER APTTYPEQUALIFIER 열거형(objidl.h)은 가능한 COM 아파트 유형 한정자 집합을 지정합니다.

BIND_FLAGS 모니커 바인딩 작업의 여러 측면을 제어합니다.
CALLTYPE IMessageFilter::HandleInComingCall에서 사용하는 호출 형식을 지정합니다.
CO_MARSHALING_CONTEXT_ATTRIBUTES CO_MARSHALING_CONTEXT_ATTRIBUTES 열거형(objidl.h)은 특정 메서드를 사용하여 쿼리할 수 있는 마샬링 컨텍스트 특성 중 하나를 식별합니다.
DATADIR 데이터 흐름의 방향을 지정합니다. 이 메서드는 결과 열거자가 열거할 수 있는 형식을 결정합니다.
EOLE_AUTHENTICATION_CAPABILITIES EOLE_AUTHENTICATION_CAPABILITIES 열거형(objidl.h)은 CoInitializeSecurity 및 IClientSecurity::SetBlanket의 다양한 기능을 지정합니다.
EXTCONN EXTCONN 열거형(objidl.h)은 포함된 개체에 존재하는 외부 연결의 형식을 지정합니다.
GLOBALOPT_PROPERTIES GLOBALOPT_PROPERTIES 열거형(objidl.h)은 IGlobalOptions 인터페이스를 사용하여 설정하거나 쿼리할 수 있는 프로세스 전역 옵션을 식별합니다.
GLOBALOPT_UNMARSHALING_POLICY_VALUES GLOBALOPT_UNMARSHALING_POLICY_VALUES 열거형(objidl.h)은 COM unmarshaling 정책 전역 옵션에 대한 값을 제공합니다.
LOCKTYPE LOCKTYPE 열거형 값은 지정된 바이트 범위에 대해 요청된 잠금 유형을 나타냅니다. 값은 ILockBytes::LockRegion 및 IStream::LockRegion 메서드에서 사용됩니다.
MKRREDUCE 모니커를 줄일 정도를 지정합니다.
MKSYS 모니커의 클래스를 나타냅니다.

PENDINGMSG IMessageFilter::MessagePending 메서드의 반환 값을 지정합니다.
PENDINGTYPE IMessageFilter::MessagePending 메서드의 중첩 수준을 나타냅니다.
SERVERCALL 서버 호출의 상태를 나타냅니다.
STGTY STGTY 열거형 값은 STATSTG 구조체의 형식 멤버에서 스토리지 요소의 형식을 나타내는 데 사용됩니다. 스토리지 요소는 스토리지 개체, 스트림 개체 또는 LOCKBYTES(바이트 배열 개체)입니다.
STREAM_SEEK STREAM_SEEK 열거형 값은 새 검색 포인터 위치를 계산할 원본을 지정합니다.
THDTYPE THDTYPE 열거형(objidl.h)은 특정 스레드가 메시지 루프를 지원하는지 여부를 나타냅니다.
TYMED 데이터 전송에 사용되는 스토리지 매체의 유형을 나타냅니다. STGMEDIUM 또는 FORMATETC 구조체에서 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

objidlbase.h 헤더

아티클2023. 03. 16.

이 헤더는 COM(구성 요소 개체 모델)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)

objidlbase.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAgileObject IAgileObject(objidlbase.h) 인터페이스는 아파트 전체에서 인터페이스를 민첩하게 표시합니다.
ICallFactory ICallFactory(objidlbase.h) 인터페이스는 비동기 인터페이스의 메서드에 대한 호출을 처리하기 위한 호출 개체를 만듭니다.
ICancelMethodCalls ICancelMethodCalls(objidlbase.h) 인터페이스는 아웃바운드 메서드 호출에 대한 취소 요청을 관리하고 해당 호출의 현재 상태를 모니터링합니다.
IComThreadingInfo IComThreadingInfo(objidlbase.h) 인터페이스를 사용하면 호출자가 실행 중인 아파트 및 스레드에 대한 다음 정보를 얻을 수 있습니다.
IContext IContext(objidlbase.h) 인터페이스는 COM+ 컨텍스트 속성 설정을 지원합니다.
IEnumContextProps IEnumContextProps(objidlbase.h) 인터페이스는 COM+ 개체 컨텍스트와 연결된 컨텍스트 속성을 열거하는 메커니즘을 제공합니다.
IEnumString IEnumString(objidlbase.h) 인터페이스는 문자열을 열거합니다. LPWSTR은 와이드 또는 유니코드 문자의 0으로 끝나는 문자열에 대한 포인터를 나타내는 형식입니다.

IEnumUnknown IEnumUnknown(objidlbase.h) 인터페이스는 IUnknown 인터페이스를 사용하여 개체를 열거하고 여러 개체가 포함된 구성 요소의 개체를 열거할 수 있습니다.
IExternalConnection IExternalConnection(objidlbase.h) 인터페이스는 서버 개체의 마샬링된 연결 또는 외부 연결 수를 관리합니다.
IFastRundown IFastRundown(objidlbase.h) 인터페이스는 인터페이스를 빠른 런다운 동작에 적합한 것으로 표시합니다.
IGlobalOptions IGlobalOptions(objidlbase.h) 인터페이스는 COM(구성 요소 개체 모델) 런타임의 전역 속성을 설정하고 쿼리합니다.
IInternalUnknown IInternalUnknown(objidlbase.h) 인터페이스는 프록시의 내부 인터페이스에 액세스해야 하는 간단한 클라이언트 쪽 처리기에 사용됩니다.
IMalloc IMalloc(objidlbase.h) 인터페이스는 메모리를 할당, 해제 및 관리합니다.
IMarshal IMarshal(objidlbase.h) 인터페이스를 사용하면 COM 개체가 인터페이스 포인터의 마샬링을 정의하고 관리할 수 있습니다.
IMarshalingStream IMarshalingStream(objidlbase.h) 인터페이스는 사용자 지정 마샬링된 개체 및 unmarshalers에 대한 마샬링 컨텍스트에 대한 추가 정보를 제공합니다.
IMultiQI IMultiQI(objidlbase.h) 인터페이스를 사용하면 클라이언트가 단일 RPC 호출을 사용하여 여러 인터페이스에 대해 개체 프록시 또는 처리기를 쿼리할 수 있습니다.
INoMarshal INoMarshal(objidlbase.h) 인터페이스는 전역 인터페이스 테이블에 마샬링되거나 저장되는 것을 지원하지 않는 개체를 표시합니다.

IObjContext IObjContext(objidlbase.h) 인터페이스는 컨텍스트에서 다양한 작업을 수행합니다.
IPipeByte IPipeByte(objidlbase.h) 인터페이스는 바이트 형식(너비 8비트)의 데이터를 전송합니다.
IPipeDouble IPipeDouble(objidlbase.h) 인터페이스는 이중 형식(너비 64비트)의 데이터를 전송합니다.
IPipeLong IPipeLong(objidlbase.h) 인터페이스는 긴 정수 형식(너비 32비트)의 데이터를 전송합니다.
IProcessInitControl IProcessInitControl(objidlbase.h) 인터페이스는 프로세스 초기화 제한 시간 간격을 지정합니다.
IPSFactoryBuffer IPSFactoryBuffer(objidlbase.h) 인터페이스는 COM 개체 프록시 및 스텁을 만들기 위한 사용자 지정 메서드를 제공합니다. 이 인터페이스는 마샬링할 수 없습니다.
IRpcChannelBuffer IRpcChannelBuffer(objidlbase.h) 인터페이스는 COM 클라이언트 프록시와 COM 서버 스텁 간에 데이터를 마샬링합니다.
IRpcOptions IRpcOptions(objidlbase.h) 인터페이스를 사용하면 호출자가 COM이 RPC(원격 프로시저 호출)를 처리하는 방법을 제어하는 다양한 속성의 값을 설정하거나 쿼리할 수 있습니다.
IRpcProxyBuffer IRpcProxyBuffer(objidlbase.h) 인터페이스는 COM 구성 요소 간에 데이터를 마샬링하는 데 사용되는 RPC 프록시를 제어합니다.
IRpcStubBuffer IRpcStubBuffer(objidlbase.h) 인터페이스는 COM 구성 요소 간에 데이터를 마샬링하는 데 사용되는 RPC 스텁을 제어합니다.
IServerSecurity IServerSecurity(objidlbase.h) 인터페이스는 서버에서 클라이언트를 인증하고 클라이언트의 가장을 관리하는 데 사용됩니다.

IStdMarshalInfo IStdMarshalInfo(objidlbase.h) 인터페이스는 표준 마샬링 중에 대상 프로세스에서 사용할 처리기를 식별하는 CLSID를 검색합니다.
ISurrogate ISurrogate(objidlbase.h) 인터페이스는 새 DLL 서버를 기존 서로게이트에 동적으로 로드하고 더 이상 필요하지 않은 경우 서로게이트를 해제하는 데 사용됩니다.
ISynchronize ISynchronize(objidlbase.h) 인터페이스는 이벤트 발생에 대한 개체 간의 비동기 통신을 제공합니다.
ISynchronizeContainer ISynchronizeContainer(objidlbase.h) 인터페이스는 서명되지 않은 동기화 개체 그룹을 관리합니다.
ISynchronizeEvent ISynchronizeEvent(objidlbase.h) 인터페이스는 동기화 개체에 이벤트 핸들을 할당합니다.
ISynchronizeHandle ISynchronizeHandle(objidlbase.h) 인터페이스는 동기화 개체와 연결된 핸들을 검색합니다.

구조체

COSERVERINFO COSERVERINFO(objidlbase.h) 구조체는 활성화 함수에 대한 원격 컴퓨터 리소스를 식별합니다.
MULTI_QI MULTI_QI(objidlbase.h) 구조체는 여러 인터페이스에 대한 쿼리의 인터페이스를 나타냅니다.
RPCOLEMESSAGE RPCOLEMESSAGE(objidlbase.h) 구조체에는 COM 구성 요소 간의 호출 인수 마샬링 및 반환 값이 포함됩니다.
SOLE_AUTHENTICATION_INFO SOLE_AUTHENTICATION_INFO(objidlbase.h) 구조는 지정된 인증 서비스에 대한 인증 서비스, 권한 부여 서비스 및 인증 정보를 식별합니다.

SOLE_AUTHENTICATION_LIST SOLE_AUTHENTICATION_LIST(objidlbase.h) 구조는 각 인증 서비스에 사용할 기본 인증 정보를 나타냅니다.
SOLE_AUTHENTICATION_SERVICE SOLE_AUTHENTICATION_SERVICE(objidlbase.h) 구조는 서버가 클라이언트와 통신하는 데 사용할 인증 서비스를 식별합니다.

열거형

APTTYPE APTTYPE(objidlbase.h) 열거형은 다양한 유형의 아파트를 지정합니다.
APTTYPEQUALIFIER APTTYPEQUALIFIER(objidlbase.h) 열거형은 가능한 COM 아파트 유형 한정자 집합을 지정합니다.
CO_MARSHALING_CONTEXT_ATTRIBUTES CO_MARSHALING_CONTEXT_ATTRIBUTES(objidlbase.h) 열거형은 GetMarshalingContextAttribute 메서드를 사용하여 쿼리할 수 있는 특성 중 하나를 식별합니다.
EOLE_AUTHENTICATION_CAPABILITIES EOLE_AUTHENTICATION_CAPABILITIES(objidlbase.h) 열거형은 CoInitializeSecurity 및 IClientSecurity::SetBlanket의 다양한 기능을 지정합니다.
EXTCONN EXTCONN(objidlbase.h) 열거형은 포함된 개체에 존재하는 외부 연결의 유형을 지정합니다.
GLOBALOPT_PROPERTIES GLOBALOPT_PROPERTIES(objidlbase.h) 열거형은 IGlobalOptions 인터페이스를 사용하여 설정하거나 쿼리할 수 있는 프로세스 전역 옵션을 식별합니다.
GLOBALOPT_UNMARSHALING_POLICY_VALUES GLOBALOPT_UNMARSHALING_POLICY_VALUES(objidlbase.h) 열거형은 COM 경계 해제 정책 전역 옵션에 대한 값을 제공합니다.

THDTYPE THDTYPE(objidlbase.h) 열거형은 특정 스레드가 메시지 루프를 지원하는지 여부를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

objpath.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows Management Instrumentation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Management Instrumentation

objpath.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

CObjectPathParser 원격 컴퓨터 이름, 네임스페이스 및 클래스를 포함할 수 있는 WMI 경로를 구문 분석합니다. 이 개체는 사용하지 않는 것이 좋습니다. 대신 IWbemPath COM 인터페이스를 사용합니다.

열거형

ObjectParserFlags CObjectPathParser에 대한 생성자에서 사용하는 플래그입니다.

objsel.h 헤더

아티클2023. 08. 25.

이 헤더는 Active Directory Domain Services 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Active Directory Domain Services

objsel.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDsObjectPicker IDsObjectPicker 인터페이스는 애플리케이션에서 개체 선택 대화 상자를 초기화하고 표시하는 데 사용됩니다. 이 인터페이스의 instance 만들려면 아래와 같이 CLSID_DsObjectPicker 클래스 식별자를 사용하여 CoCreateInstance를 호출합니다.
IDsObjectPickerCredentials IDsObjectPickerCredentials 인터페이스를 사용하면 이 인터페이스를 구현하는 IDsObjectPicker 개체에 대한 자격 증명을 재정의할 수 있습니다.

구조체

DS_SELECTION DS_SELECTION 구조체에는 사용자가 개체 선택 대화 상자에서 선택한 개체에 대한 데이터가 포함됩니다. DS_SELECTION_LIST 구조체에는 DS_SELECTION 구조체의 배열이 포함되어 있습니다.
DS_SELECTION_LIST DS_SELECTION_LIST 구조에는 사용자가 개체 선택 대화 상자에서 선택한 개체에 대한 데이터가 포함됩니다.
DSOP_FILTER_FLAGS 지정된 scope 또는 범위에 대해 사용자에게 표시되는 개체의 유형을 나타내는 플래그를 포함합니다.

DSOP_INIT_INFO DSOP_INIT_INFO 구조에는 개체 선택 대화 상자를 초기화하는 데 필요한 데이터가 포함되어 있습니다. 이 구조체는 IDsObjectPicker::Initialize 메서드와 함께 사용됩니다.
DSOP_SCOPE_INIT_INFO DSOP_SCOPE_INIT_INFO 구조체는 동일한 특성을 가진 하나 이상의 scope 형식을 설명합니다.
DSOP_UPLEVEL_FILTER_FLAGS DSOP_UPLEVEL_FILTER_FLAGS 구조체에는 상위 수준 scope 사용할 필터를 나타내는 플래그가 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ocidl.h 헤더

아티클2023. 03. 22.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)
- Directshow
- Windows 검색

ocidl.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAdviseSinkEx 이 인터페이스는 IAdviseSink에서 파생되어 개체의 뷰 상태 변경 내용을 싱크에 알리기 위한 확장을 제공합니다.
IClassFactory2 모든 종류의 개체 서버에서 클래스 팩터리 개체가 라이선스를 통해 개체 생성을 제어할 수 있도록 합니다.
IConnectionPoint 연결 가능한 개체에 대한 연결점을 지원합니다. (IConnectionPoint)
IConnectionPointContainer 연결 가능한 개체에 대한 연결점을 지원합니다. (IConnectionPointContainer)
IEnumConnectionPoints 연결점을 열거합니다.
IEnumConnections 연결 가능한 개체에 대한 현재 연결을 열거합니다.
IEnumOleUndoUnits 실행 취소 또는 다시 실행 스택에서 실행 취소 단위를 열거합니다.

IFont Windows 글꼴 개체 주위에 래퍼를 제공합니다.
IFontDisp Automation을 통해 글꼴 개체의 속성을 노출합니다. IFont 메서드의 하위 집합을 제공합니다.
IObjectWithSite 컨테이너에서 개체와 해당 사이트 간의 통신을 지원하는 간단한 방법을 제공합니다.
IOleControl 컨트롤 개체에서 키보드 니모닉, 주변 속성 및 이벤트를 지원하기 위한 기능을 제공합니다.
IOleControlSite 사이트 개체가 컨테이너 내의 각 포함된 컨트롤을 관리할 수 있도록 하는 메서드를 제공합니다.
IOleInPlaceObjectWindowless 창 없는 개체가 창 메시지를 처리하고 끌어서 놓기 작업에 참여할 수 있도록 합니다. IOleInPlaceObject 인터페이스에서 파생되고 확장됩니다.
IOleInPlaceSiteEx 개체가 활성화되고 비활성화될 때 개체가 화면에서 불필요한 깜박임을 방지할 수 있도록 하는 추가 활성화 및 비활성화 알림 메서드 집합을 제공합니다.
IOleInPlaceSiteWindowless IOleInPlaceSiteEx 인터페이스를 확장합니다.
IOleParentUndoUnit 실행 취소 단위가 자식 실행 취소 단위를 포함하도록 설정합니다.
IOleUndoManager IOleUndoManager 인터페이스를 사용하면 컨테이너가 포함된 컨트롤 내에서 발생하는 작업에 대해 다단계 실행 취소 및 다시 실행 작업을 구현할 수 있습니다.
IOleUndoUnit 실행 취소 단위의 기본 인터페이스 역할을 합니다. 실행 취소 단위는 단일 작업을 실행 취소하거나 다시 실행하는 데 필요한 정보를 캡슐화합니다.

IPerPropertyBrowsing 개체에서 제공하는 속성 페이지의 정보를 검색합니다.
IPersistPropertyBag IPropertyBag 및 IErrorlog를 사용하여 개별 속성 기반 지속성 메커니즘을 정의합니다.
IPersistStreamInit 초기화 메서드를 추가하는 IPersistStream을 대체합니다.
IPicture 그림 개체 및 해당 속성을 관리합니다. 그림 개체는 비트맵, 아이콘 및 메타파일의 언어 중립적 추상화를 제공합니다.
IPictureDisp Automation을 통해 그림 개체의 속성을 노출합니다. IPicture 메서드를 통해 사용할 수 있는 기능의 하위 집합을 제공합니다.
IPointerInactive 개체가 대부분의 시간 동안 비활성 상태로 유지되지만 끌어서 놓기를 포함하여 마우스와의 상호 작용에 계속 참여할 수 있습니다.
IPropertyNotifySink 송신 인터페이스로 IPropertyNotifySink를 지원하는 개체에서 속성 변경에 대한 알림을 수신하기 위해 싱크 개체에 의해 구현됩니다.
IPropertyPage 속성 시트 내의 특정 페이지를 관리하는 속성 페이지 개체의 주요 기능을 제공합니다.
IPropertyPage2 페이지에서 속성의 초기 선택을 지원하기 위한 IPropertyPage 확장입니다.
IPropertyPageSite 속성 페이지 사이트 개체의 주요 기능을 제공합니다.
IProvideClassInfo 형식 라이브러리에서 개체의 coclass 항목에 대한 형식 정보에 대한 액세스를 제공합니다.

IProvideClassInfo2 기본 이벤트 집합에 대한 개체의 나가는 인터페이스 IID를 더 빠르고 쉽게 검색할 수 있도록 하는 IProvideClassInfo 확장입니다.
IProvideMultipleClassInfo 동작을 결정하는 여러 코클래스가 있을 수 있는 구성 요소에서 형식 정보를 더 빠르고 쉽게 검색할 수 있도록 하는 IProvideClassInfo2 확장입니다.
IQuickActivate 컨트롤 및 컨테이너를 사용하여 컨트롤 로드에서 성능 병목 현상을 방지할 수 있습니다. 컨트롤과 해당 컨테이너 간의 로드 시간 또는 초기화 시간 핸드셰이크를 단일 호출로 결합합니다.
ISimpleFrameSite 다른 중첩된 컨트롤에 대한 간단한 컨테이너 역할을 하는 간단한 프레임 컨트롤을 제공합니다.
ISpecifyPropertyPages 개체가 속성 페이지를 지원한다는 것을 나타냅니다.
IViewObjectEx IViewObject2에서 파생된 확장은 직사각형이 아닌 개체 및 투명 개체에 대한 향상된 깜박임 없는 그리기, 직사각형이 아닌 개체에 대한 적중 테스트 및 컨트롤 크기 조정을 지원합니다.

Functions

HPALETTE_UserFree RPC 스텁 파일(HPALETTE_UserFree)에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HPALETTE_UserFree64 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다. (HPALETTE_UserFree64)
HPALETTE_UserMarshal HPALETTE 개체를 RPC 버퍼로 마샬링합니다. (HPALETTE_UserMarshal)
HPALETTE_UserMarshal64 HPALETTE 개체를 RPC 버퍼로 마샬링합니다. (HPALETTE_UserMarshal64)

HPALETTE_UserSize HPALETTE 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다. (HPALETTE_UserSize)
HPALETTE_UserSize64 HPALETTE 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다. (HPALETTE_UserSize64)
HPALETTE_UserUnmarshal RPC 버퍼에서 HPALETTE 개체를 숨기지 않습니다. (HPALETTE_UserUnmarshal)
HPALETTE_UserUnmarshal64 RPC 버퍼에서 HPALETTE 개체를 숨기지 않습니다. (HPALETTE_UserUnmarshal64)

구조체

CADWORD 속성에 대해 미리 정의된 문자열 중 하나에 해당하는 값을 가져오는 데 사용할 수 있는 값의 계산된 배열을 지정합니다.
CALPOLESTR 속성이 허용할 수 있는 미리 정의된 문자열을 지정하는 데 사용되는 개수 지정된 문자열 배열을 지정합니다.
CAUUID 개체가 표시하려는 속성 페이지에 대한 CLSID 배열을 수신하는 데 사용되는 UUID 또는 GUID 형식의 계산된 배열을 지정합니다.
CONNECTDATA 지정된 연결점에 대해 존재하는 연결을 설명합니다.
CONTROLINFO 컨트롤의 키보드 니모닉 및 키보드 동작을 설명하는 매개 변수를 포함합니다. 구조체는 IOleControl::GetControlInfo 메서드 중에 채워집니다.

DVASPECTINFO GDI를 보다 효율적으로 사용하여 비활성 개체의 렌더링을 최적화하기 위해 IViewObject::D raw 메서드에서 사용하는 정보를 포함합니다.
DVEXTENTINFO IViewObjectEx::GetNaturalExtent에 사용되는 크기 조정 데이터를 나타냅니다.
LICINFO 라이선스를 지원하는 클래스 팩터리의 라이선스 동작을 설명하는 매개 변수를 포함합니다. 구조체는 IClassFactory2::GetLicInfo 메서드를 호출하여 채워집니다.
Pointf HIMETRIC으로 표현된 부동 소수점 및 제어 단위로 표현된 컨테이너 단위 간에 변환하는 데 사용되는 정보를 포함합니다.
PROPPAGEINFO 속성 프레임에 대한 속성 페이지를 설명하는 데 사용되는 매개 변수를 포함합니다. 속성 페이지는 IPropertyPage::GetPageInfo 메서드의 호출자 제공 구조를 채웁니다.
QACONTAINER IQuickActivate::QuickActivate에 대한 컨테이너 정보를 지정합니다.
QACONTROL IQuickActivate::QuickActivate에 대한 제어 정보를 지정합니다.

열거형

ACTIVATEFLAGS 개체가 창 없는 개체로 활성화되는지 여부를 나타냅니다. IOleInPlaceSiteEx::OnInPlaceActivateEx에서 사용됩니다.
DVASPECT2 그리기 프로세스를 최적화하는 데 사용되는 새 그리기 측면을 지정합니다.

DVASPECTINFOFLAG 개체가 자체의 최적화된 그리기를 지원할 수 있는지 여부를 나타냅니다.
DVEXTENTMODE 크기 조정 모드가 콘텐츠인지 정수 크기 조정인지 여부를 나타냅니다.
GUIDKIND IProvideClassInfo2의 개체에서 요청된 정보의 종류를 지정하는 데 사용되는 플래그입니다.
HITRESULT 위치가 개체의 이미지 내에 있는지 여부를 나타냅니다.
OLEDCFLAGS 개체가 요청한 디바이스 컨텍스트에 대한 추가 정보를 컨테이너에 지정합니다.
PICTUREATTRIBUTES IPicture::get_Attributes 메서드를 통해 반환되는 그림 개체의 특성을 지정합니다.
POINTERINACTIVE 개체의 활성화 정책을 나타내며 IPointerInactive::GetActivationPolicy 메서드에 사용됩니다.
QACONTAINERFLAGS 컨테이너에서 제공하는 주변 속성을 나타냅니다. QACONTAINER 구조체의 dwAmbientFlags 멤버에 사용됩니다.
READYSTATE ReadyState 속성은 MSWebDVD 개체의 ReadyState를 검색합니다.
UASFLAGS 부모 실행 취소 단위에 대한 정보를 제공합니다.
VIEWSTATUS 개체의 불투명도와 개체에서 지원하는 그리기 측면을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ole.h 헤더

아티클2023. 03. 16.

이 헤더는 COM(구성 요소 개체 모델)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)

ole.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

OleCreate OleCreate 함수(ole.h)는 CLSID로 식별되는 포함된 개체를 만듭니다. 새 개체를 삽입할 수 있는 메뉴 항목을 구현할 수 있습니다.
OleCreateFromFile OleCreateFromFile 함수(ole.h)는 명명된 파일의 내용에서 포함된 개체를 만듭니다.
OleDraw OleDraw 함수(ole.h)를 사용하면 개체를 더 쉽게 그릴 수 있습니다. IViewObject::D raw를 직접 호출하는 대신 사용할 수 있습니다.
OleLoadFromStream OleLoadFromStream 함수(ole.h)는 스트림에서 개체를 로드합니다.
OleSaveToStream OleSaveToStream 함수(ole.h)는 IPersistStream 인터페이스가 있는 개체를 지정된 스트림에 저장합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ole2.h 헤더

아티클2023. 03. 16.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)
- 구조적 저장소

ole2.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CreateDataAdviseHolder CreateDataAdviseHolder 함수(ole2.h)는 데이터 조언 소유자 개체에서 IDataAdviseHolder의 OLE 구현에 대한 포인터를 검색합니다.
CreateOleAdviseHolder 복합 문서 알림을 관리하기 위한 자문 소유자 개체를 만듭니다. IOleAdviseHolder 인터페이스의 개체 OLE 구현에 대한 포인터를 반환합니다.
Dodragdrop OLE 끌어서 놓기 작업을 수행합니다.
IsAccelerator 지정된 키 입력이 지정된 액셀러레이터 테이블의 액셀러레이터에 매핑되는지 여부를 결정합니다.
OleBuildVersion 이 함수는 사용되지 않습니다.
OleConvertIStorageToOLESTREAM OleConvertIStorageToOLESTREAM 함수는 지정된 스토리지 개체를 OLE 2 구조적 스토리지에서 OLE 1 스토리지 개체 모델로 변환하지만 프레젠테이션 데이터는 포함하지 않습니다. 이는 여러 호환성 함수 중 하나입니다.
OleConvertIStorageToOLESTREAMEx OleConvertIStorageToOLESTREAMEx 함수는 지정된 스토리지 개체를 OLE 2 구조적 스토리지에서 프레젠테이션 데이터를 포함한 OLE 1 스토리지 개체 모델로 변환합니다.

OleConvertOLESTREAMToIStorage 프레젠테이션 데이터를 지정하지 않고 지정된 개체를 OLE 1 스토리지 모델에서 OLE 2 구조적 스토리지 개체로 변환합니다.
OleConvertOLESTREAMToIStorageEx OleConvertOLESTREAMToIStorageEx 함수는 지정된 개체를 OLE 1 스토리지 모델에서 프레젠테이션 데이터를 포함한 OLE 2 구조적 스토리지 개체로 변환합니다. 이는 여러 호환성 함수 중 하나입니다.
OleCreate OleCreate 함수(ole2.h)는 CLSID로 식별되는 포함된 개체를 만듭니다. 최종 사용자가 개체를 삽입할 수 있도록 하는 메뉴 항목을 구현할 수 있습니다.
OleCreateDefaultHandler 기본 포함 처리기의 새 인스턴스를 만듭니다. 이 인스턴스는 포함된 개체가 실행 중 상태가 될 때 로컬 서버를 만들도록 초기화됩니다.
OleCreateEmbeddingHelper OLE 기본 개체 처리기 조각과 함께 집계된 애플리케이션 제공 코드를 사용하여 OLE 포함 도우미 개체를 만듭니다. 이 도우미 개체는 호출자가 결정한 특정 컨텍스트 및 역할에서 만들고 사용할 수 있습니다.
OleCreateEx OleCreate에서 지원하는 단일 형식 대신 여러 프레젠테이션 형식 또는 데이터를 캐싱해야 하는 컨테이너의 개체를 보다 효율적으로 인스턴스화할 수 있도록 지원하여 OleCreate 기능을 확장합니다.
OleCreateFromData 클립보드 또는 OLE 끌어서 놓기 작업의 일부로 검색된 데이터 전송 개체에서 포함된 개체를 만듭니다. OLE 끌어서 놓기 작업에서 붙여넣기를 구현하는 데 사용됩니다.
OleCreateFromDataEx OleCreateFromData에서 지원하는 단일 형식 대신 여러 형식의 프레젠테이션 또는 데이터를 캐싱해야 하는 컨테이너의 개체를 보다 효율적으로 인스턴스화할 수 있도록 지원하여 OleCreateFromData 기능을 확장합니다.
OleCreateFromFile OleCreateFromFile 함수(ole2.h)는 명명된 파일의 내용에서 포함된 개체를 만듭니다.

OleCreateFromFileEx OleCreateFromFile에서 지원하는 단일 형식 대신 여러 프레젠테이션 형식 또는 데이터를 캐싱해야 하는 컨테이너의 개체를 보다 효율적으로 인스턴스화할 수 있도록 지원하여 OleCreateFromFile 기능을 확장합니다.
OleCreateLink OLE 복합 문서 연결된 개체를 만듭니다.
OleCreateLinkEx OleCreateLink에서 지원하는 단일 형식 대신 여러 형식의 프레젠테이션 또는 데이터를 캐싱해야 하는 컨테이너의 개체를 보다 효율적으로 인스턴스화하도록 지원하여 OleCreateLink 기능을 확장합니다.
OleCreateLinkFromData 클립보드 또는 OLE 끌어서 놓기 작업의 일부로 검색된 데이터 전송 개체에서 연결된 개체를 만듭니다.
OleCreateLinkFromDataEx OleCreateLinkFromData에서 지원하는 단일 형식 대신 여러 형식의 프레젠테이션 또는 데이터를 캐싱해야 하는 컨테이너의 개체를 보다 효율적으로 인스턴스화할 수 있도록 지원하여 OleCreateLinkFromData 기능을 확장합니다.
OleCreateLinkToFile 파일에 연결된 개체를 만듭니다.
OleCreateLinkToFileEx OleCreateLinkToFile에서 지원하는 단일 형식 대신 여러 형식의 프레젠테이션 또는 데이터를 캐싱해야 하는 컨테이너의 개체를 보다 효율적으로 인스턴스화할 수 있도록 지원하여 OleCreateLinkToFile 기능을 확장합니다.
OleCreateMenuDescriptor 메뉴 메시지 및 명령을 디스패치할 때 사용할 OLE 메뉴 설명자(즉, 메뉴를 설명하는 OLE 제공 데이터 구조)를 만들고 반환합니다.
OleCreateStaticFromData 데이터 전송 개체에서 네이티브 데이터가 없는 표현만 포함하는 정적 개체를 만듭니다.

OleDestroyMenuDescriptor OleCreateMenuDescriptor 함수에 의해 할당된 공유 메뉴 설명자를 해제하기 위해 컨테이너에서 호출됩니다.
OleDoAutoConvert 레지스트리에서 해당 개체 클래스에 대한 자동 변환이 설정된 경우 개체를 새 클래스로 자동으로 변환합니다.
OleDraw OleDraw 함수(ole2.h)를 사용하면 개체를 더 쉽게 그릴 수 있습니다. IViewObject::D raw를 직접 호출하는 대신 사용할 수 있습니다.
OleDuplicateData 지정된 핸들에 있는 데이터를 복제하고 중복된 데이터에 대한 핸들을 반환합니다. 원본 데이터는 클립보드 형식입니다. 이 함수를 사용하여 IDataObject와 같은 일부 데이터 전송 인터페이스를 구현할 수 있습니다.
OleFlushClipboard 클립보드 종료 시퀀스를 수행합니다. 또한 OleSetClipboard 함수에 의해 클립보드에 배치된 IDataObject 포인터를 해제합니다.
OleGetAutoConvert 지정된 CLSID의 개체에 대해 레지스트리가 다른 CLSID로 자동으로 변환되도록 설정되었는지 여부를 확인하고, 이 경우 새 CLSID를 검색합니다.
OleGetClipboard 클립보드의 내용에 액세스하는 데 사용할 수 있는 데이터 개체를 검색합니다.
OleGetClipboardWithEnterpriseInfo Windows Information Protection 지원 애플리케이션이 데이터 및 원본 애플리케이션에 대한 Windows Information Protection 정보와 함께 OLE 클립보드에서 IDataObject를 검색할 수 있도록 합니다.
OleGetIconOfClass 지정된 CLSID에 대한 아이콘 및 문자열 레이블이 포함된 메타파일의 핸들을 반환합니다.
OleGetIconOfFile 지정된 파일 이름에 대한 아이콘 및 문자열 레이블이 포함된 메타파일의 핸들을 반환합니다.

OleInitialize 현재 아파트에서 COM 라이브러리를 초기화하고, 동시성 모델을 STA(단일 스레드 아파트)로 식별하고, 아래 설명 섹션에 설명된 추가 기능을 사용하도록 설정합니다.
OleIsCurrentClipboard 이전에 OleSetClipboard 함수에 의해 클립보드에 배치된 데이터 개체 포인터가 클립보드에 있는지 여부를 확인합니다.
OleIsRunning 복합 문서 개체가 현재 실행 중 상태인지 여부를 확인합니다.
OleLoad 지정된 스토리지 개체 내에 중첩된 개체를 메모리에 로드합니다.
OleLoadFromStream OleLoadFromStream 함수(ole2.h)는 스트림에서 개체를 로드합니다.
OleLockRunning 이미 실행 중인 개체를 실행 상태로 잠그거나 실행 상태에서 잠금을 해제합니다. (OleLockRunning)
OleMetafilePictFromIconAndLabel 지정된 아이콘과 레이블이 그려지는 메타파일을 만듭니다.
OleNoteObjectVisible 개체를 실행 상태로 유지하는 외부 참조를 증가 또는 감소합니다.
OleQueryCreateFromData 데이터 개체에 OleCreateFromData 또는 OleCreateStaticFromData 함수 호출을 통해 포함된 개체가 될 수 있는 형식 중 하나가 있는지 확인합니다.
OleQueryLinkFromData 클립보드 데이터 개체에서 OLE 연결된 개체(OLE 포함 개체가 아닌)를 만들 수 있는지 여부를 결정합니다.
OleRegEnumFormatEtc OLE 개체 서버가 시스템 레지스트리에 등록한 데이터 형식을 열거하는 데 사용할 수 있는 열거형 개체를 만듭니다.

OleRegEnumVerbs 지정된 클래스에 대해 등록된 동사의 열거형을 제공합니다. 사용자 지정 DLL 개체 애플리케이션의 개발자는 이 함수를 사용하여 기본 개체 처리기의 동작을 에뮬레이트합니다.
OleRegGetMiscStatus 레지스트리에서 지정된 CLSID에서 지원하는 프레젠테이션 및 동작에 대한 기타 정보를 반환합니다.
OleRegGetUserType 레지스트리에서 지정된 클래스의 사용자 유형을 가져옵니다.
OleRun OLE 복합 문서 개체를 실행 상태로 전환합니다.
OleSave 트랜잭션 모드에서 열린 개체를 지정된 스토리지 개체에 저장합니다.
OleSaveToStream OleSaveToStream 함수(ole2.h)는 IPersistStream 인터페이스가 있는 개체를 지정된 스트림에 저장합니다.
OleSetAutoConvert 해당 클래스의 개체가 로드될 때 다른 클래스로 자동 변환하기 위한 CLSID를 지정합니다.
OleSetClipboard 특정 데이터 개체에 대한 포인터를 클립보드에 배치합니다. 이렇게 하면 OleGetClipboard 함수에서 데이터 개체에 액세스할 수 있습니다.
OleSetContainedObject 포함된 개체에 대한 링크를 지원하는 컨테이너에 대해 참조 계산이 올바르게 수행되도록 개체가 OLE 컨테이너에 포함되어 있음을 알 수 있습니다. (OleSetContainedObject)
OleSetMenuDescriptor 컨테이너의 프레임 창에서 OLE 디스패치 코드를 설치하거나 제거합니다.
OleTranslateAccelerator 개체 애플리케이션에서 호출하면 개체의 컨테이너가 컨테이너의 가속기 테이블에 따라 가속기를 변환할 수 있습니다.

OleUninitialize 아파트의 COM 라이브러리를 닫고, 클래스 팩터리, 기타 COM 개체 또는 아파트에 보관된 서버를 해제하고, 아파트에서 RPC를 사용하지 않도록 설정하고, 아파트가 유지 관리하는 모든 리소스를 해제합니다.
ReadFmtUserTypeStg ReadFmtUserTypeStg 함수는 WriteFmtUserTypeStg 함수를 사용하여 이전에 저장한 클립보드 형식 및 사용자 유형을 반환합니다.
RegisterDragDrop 지정된 창을 OLE 끌어서 놓기 작업의 대상이 될 수 있는 창으로 등록하고 삭제 작업에 사용할 IDropTarget 인스턴스를 지정합니다.
ReleaseStgMedium 지정된 스토리지 매체를 해제합니다.
RevokeDragDrop OLE 끌어서 놓기 작업의 잠재적 대상으로 지정된 애플리케이션 창의 등록을 취소합니다.
SetConvertStg SetConvertStg 함수는 스토리지 개체의 변환 비트를 설정하여 개체가 열릴 때 새 클래스로 변환될 것임을 나타냅니다. GetConvertStg 함수를 호출하여 설정을 검색할 수 있습니다.
WriteFmtUserTypeStg WriteFmtUserTypeStg 함수는 스토리지 개체에 클립보드 형식 및 사용자 형식을 씁니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

oleacc.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows 접근성 기능에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 접근성 기능

oleacc.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAccessible 클라이언트 애플리케이션에서 사용자 인터페이스 요소와 해당 자식에 액세스할 수 있도록 하는 메서드 및 속성을 노출합니다.
IAccessibleHandler 개체 ID에서 액세스 가능한 요소를 검색하는 메서드를 노출합니다.
IAccessibleWindowlessSite Microsoft ActiveX 컨트롤 사이트는 이 인터페이스를 구현하여 Microsoft Active Accessibility 구현이 있는 창 없는 ActiveX 컨트롤이 접근성을 표현할 수 있도록 합니다.
IAccIdentity 액세스 가능한 요소에 대한 고유 식별자를 제공하는 메서드를 노출합니다.
IAccPropServer 액세스 가능한 요소에 대한 속성 값을 검색하는 메서드를 노출합니다.
IAccPropServices 액세스 가능한 요소에 주석을 추가하고 ID 문자열을 조작하기 위한 메서드를 노출합니다.

Functions

AccessibleChildren

액세스 가능한 컨테이너 개체 내에서 각 자식의 자식 ID 또는 IDispatch를 검색합니다.
AccessibleObjectFromEvent 클라이언트의 이벤트 후크 함수에서 현재 처리 중인 이벤트를 생성한 개체에 대한 IAccessible 인터페이스의 주소를 검색합니다.
AccessibleObjectFromPoint 화면의 지정된 지점에 표시되는 개체에 대한 IAccessible 인터페이스 포인터의 주소를 검색합니다.
AccessibleObjectFromWindow 지정된 창과 연결된 개체에 대해 지정된 인터페이스의 주소를 검색합니다.
AccNotifyTouchInteraction AT(보조 기술) 애플리케이션이 사용자의 터치 제스처의 결과로 Windows Automation API(예: Microsoft UI 자동화)를 통해 UI와 상호 작용하고 있음을 시스템에 알릴 수 있습니다.
AccSetRunningUtilityState AT(보조 기술) 애플리케이션의 현재 상태가 일반적으로 시스템에서 제공하는 기능에 영향을 미치는지 여부를 나타내는 시스템 값을 설정합니다.
CreateStdAccessibleObject 시스템 제공 사용자 인터페이스 요소의 지정된 형식의 메서드 및 속성을 사용하여 액세스 가능한 개체를 만듭니다.
CreateStdAccessibleProxyA 시스템에서 제공하는 사용자 인터페이스 요소의 지정된 클래스의 속성과 메서드가 있는 액세스 가능한 개체를 만듭니다. (ANSI)
CreateStdAccessibleProxyW 시스템에서 제공하는 사용자 인터페이스 요소의 지정된 클래스의 속성과 메서드가 있는 액세스 가능한 개체를 만듭니다. (유니코드)
GetOleaccVersionInfo Microsoft Active Accessibility 파일 Oleacc.dll 버전 번호 및 빌드 번호를 검색합니다.
GetRoleTextA 지정된 역할 값에 대한 개체의 역할을 설명하는 지역화된 문자열을 검색합니다. (ANSI)
GetRoleTextW

지정된 역할 값에 대한 개체의 역할을 설명하는 지역화된 문자열을 검색합니다. (유니코드)
GetStateTextA 미리 정의된 단일 상태 비트 플래그에 대한 개체의 상태를 설명하는 지역화된 문자열을 검색합니다. 상태 값은 하나 이상의 비트 플래그 조합이므로 클라이언트는 이 함수를 두 번 이상 호출하여 모든 상태 문자열을 검색합니다. (ANSI)
GetStateTextW 미리 정의된 단일 상태 비트 플래그에 대한 개체의 상태를 설명하는 지역화된 문자열을 검색합니다. 상태 값은 하나 이상의 비트 플래그 조합이므로 클라이언트는 이 함수를 두 번 이상 호출하여 모든 상태 문자열을 검색합니다. (유니코드)
LresultFromObject 지정된 개체에 대한 핸들과 유사한 참조를 반환합니다. 서버는 WM_GETOBJECT 처리할 때 이 참조를 반환합니다.
ObjectFromLresult 이전에 생성된 개체 참조를 기반으로 액세스 가능한 개체에 대해 요청된 인터페이스 포인터를 검색합니다.
WindowFromAccessibleObject IAccessible 인터페이스의 특정 instance 해당하는 창 핸들을 검색합니다.

구조체

MSAAMENUINFO 서버 개발자가 소유자가 그린 메뉴 항목의 이름을 노출하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

oleauto.h 헤더

아티클2023. 08. 25.

이 헤더는 Automation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Automation

oleauto.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

BstrFromVector BSTR을 반환하여 BSTR의 문자에 벡터의 각 요소를 할당합니다.
ClearCustData 사용자 지정 데이터 항목을 보유하는 데 사용되는 메모리를 해제합니다.
CreateDispTypeInfo IDispatch 구현에 사용할 간소화된 형식 정보를 만듭니다.
CreateErrorInfo 제네릭 오류 개체의 instance 만듭니다.
CreateStdDispatch 단일 함수 호출을 통해 IDispatch 인터페이스의 표준 구현을 만듭니다. 이렇게 하면 Automation을 통해 개체를 쉽게 노출할 수 있습니다.
CreateTypeLib ICreateTypeLib 인터페이스를 지원하는 새 개체 instance 대한 액세스를 제공합니다.
CreateTypeLib2 현재 파일 형식으로 형식 라이브러리를 만듭니다.
DispCallFunc 사용자 지정된 Invoke에 대한 머신 독립성 기능을 제공하는 Invoke에 대한 하위 수준 도우미입니다. (DispCallFunc)

DispGetIDsOfNames 사용자 지정된 Invoke에 대한 머신 독립성 기능을 제공하는 Invoke에 대한 하위 수준 도우미입니다. (DispGetIDsOfNames)
DispGetParam DISPPARAMS 구조체에서 매개 변수를 검색하여 명명된 매개 변수와 위치 매개 변수를 모두 확인하고 매개 변수를 지정된 형식으로 강제 변환합니다.
DispInvoke 인터페이스에 대한 형식 정보가 제공되면 인터페이스에서 멤버 함수를 자동으로 호출합니다.
DosDateTimeToVariantTime 시간의 MS-DOS 표현을 변형에 저장된 날짜 및 시간 표현으로 변환합니다.
GetActiveObject OLE에 등록된 실행 중인 개체에 대한 포인터를 검색합니다.
GetAltMonthNames 보조(대체) 월 이름을 검색합니다.
GetErrorInfo 현재 논리 스레드에서 SetErrorInfo에 대한 이전 호출에서 설정한 오류 정보 포인터를 가져옵니다.
GetRecordInfoFromGuids 형식 라이브러리를 로드하지 않고 형식 정보의 GUID를 전달하여 UDT에 대한 IRecordInfo 인터페이스에 대한 포인터를 반환합니다.
GetRecordInfoFromTypeInfo 형식 정보를 전달하여 UDT의 IRecordInfo 인터페이스에 대한 포인터를 반환합니다.
LHashValOfName 이름의 해시 값을 계산합니다. (LHashValOfName)
LHashValOfNameSys 이름의 해시 값을 계산합니다. (LHashValOfNameSys)
LHashValOfNameSysA

지정된 이름의 해시 값을 계산합니다.
LoadRegTypeLib 레지스트리 정보를 사용하여 형식 라이브러리를 로드합니다.
LoadTypeLib 형식 라이브러리를 로드하고 등록합니다.
LoadTypeLibEx 형식 라이브러리를 로드하고(선택 사항) 시스템 레지스트리에 등록합니다. .
OaBuildVersion OLE Automation의 빌드 버전을 검색합니다.
OaEnablePerUserTLibRegistration RegisterTypeLib 함수가 WINDOWS Vista SP1(서비스 팩 1), Windows Server 2008 이상 운영 체제 버전에서 기본 레지스트리 매핑을 재정의할 수 있도록 합니다.
QueryPathOfRegTypeLib 등록된 형식 라이브러리의 경로를 검색합니다.
RegisterActiveObject 개체를 해당 클래스의 활성 개체로 등록합니다.
RegisterTypeLib 형식 라이브러리에 대한 정보를 시스템 레지스트리에 추가합니다.
RegisterTypeLibForUser 호출하는 사용자가 사용할 형식 라이브러리를 등록합니다.
RevokeActiveObject 개체의 상태 활성으로 종료합니다.
SafeArrayAccessData 배열의 잠금 수를 증가시키고 배열 데이터에 대한 포인터를 검색합니다.
SafeArrayAddRef 지정된 안전 배열에 대한 설명자의 고정 참조 수를 1씩 늘리고, 안전한 배열의 설명자에 의해 결정

된 대로 해당 데이터가 동적으로 할당된 경우 지정된 안전 배열에 대한 데이터의 고정 참조 수를 1씩 늘릴 수 있습니다.
SafeArrayAllocData SafeArrayAllocDescriptor로 만든 설명자를 기반으로 안전한 배열에 대한 메모리를 할당합니다.
SafeArrayAllocDescriptor 안전한 배열 설명자에 대한 메모리를 할당합니다.
SafeArrayAllocDescriptorEx 배열 데이터를 할당하지 않고 VT_RECORD 포함하여 유효한 변형 형식의 배열에 대한 안전한 배열 설명자를 만듭니다.
SafeArrayCopy 기존 안전 배열의 복사본을 만듭니다.
SafeArrayCopyData 대상 배열의 리소스를 해제한 후 원본 배열을 지정된 대상 배열에 복사합니다.
SafeArray만들기 새 배열 설명자를 만들고, 배열에 대한 데이터를 할당 및 초기화하고, 새 배열 설명자에 대한 포인터를 반환합니다.
SafeArrayCreateEx 지정된 VARTYPE, 차원 및 범위 수에서 안전한 배열 설명자를 만들고 반환합니다.
SafeArrayCreateVector 1차원 배열을 만듭니다. SafeArrayCreateVector를 사용하여 만든 안전한 배열은 고정 크기이므로 상수 FADF_FIXEDSIZE 항상 설정됩니다.
SafeArrayCreateVectorEx 지정된 VARTYPE 및 범위의 1차원 안전 배열을 만들고 반환합니다.
SafeArrayDestroy 기존 배열 설명자와 배열의 모든 데이터를 삭제합니다.
SafeArrayDestroyData 지정된 안전 배열의 모든 데이터를 삭제합니다.

SafeArrayDestroyDescriptor 지정된 안전 배열의 설명자를 제거합니다.
SafeArrayGetDim 배열의 차수를 가져옵니다.
SafeArrayGetElement 배열의 단일 요소를 검색합니다.
SafeArrayGetElemsize 요소의 크기를 가져옵니다.
SafeArrayGetIID 지정된 안전 배열 내에 포함된 인터페이스의 GUID를 가져옵니다.
SafeArrayGetLBound 지정된 안전 배열의 모든 차원에 대한 하한을 가져옵니다.
SafeArrayGetRecordInfo 지정된 안전 배열에 포함된 UDT의 IRecordInfo 인터페이스를 검색합니다.
SafeArrayGetUBound 지정된 안전 배열의 모든 차원에 대한 상한을 가져옵니다.
SafeArrayGetVartype 지정된 안전 배열에 저장된 VARTYPE을 가져옵니다.
SafeArrayLock 배열의 잠금 수를 증가시키고 배열 데이터에 대한 포인터를 배열 설명자의 pvData에 배치합니다.
SafeArrayPtrOfIndex 배열 요소에 대한 포인터를 가져옵니다.
SafeArrayPutElement 배열의 지정된 위치에 데이터 요소를 저장합니다.
SafeArrayRedim

지정된 안전 배열의 가장 오른쪽(최적)의 경계를 변경합니다.
SafeArrayReleaseData 지정된 안전 배열 데이터에 대한 고정 참조 수를 1씩 줄입니다. 해당 수가 0에 도달하면 해당 데이터의 메모리가 더 이상 해제되지 않습니다.
SafeArrayReleaseDescriptor 지정된 안전 배열의 설명자에 대한 고정 참조 수를 1씩 줄입니다. 해당 수가 0에 도달하면 해당 설명자의 메모리가 더 이상 해제되지 않습니다.
SafeArraySetIID 지정된 안전 배열에 대한 인터페이스의 GUID를 설정합니다.
SafeArraySetRecordInfo 지정된 안전 배열의 레코드 정보를 설정합니다.
SafeArrayUnaccessData 배열의 잠금 수를 감소시키고 SafeArrayAccessData에서 검색한 포인터를 무효화합니다.
SafeArrayUnlock 해제하거나 크기를 조정할 수 있도록 배열의 잠금 수를 감소합니다.
SetErrorInfo 실행의 현재 논리 스레드에 대한 오류 정보 개체를 설정합니다.
SysAddRefString 지정된 문자열에 대한 고정 참조 수를 1씩 늘입니다.
SysAllocString 새 문자열을 할당하고 전달된 문자열을 복사합니다.
SysAllocStringByteLen ANSI 문자열을 입력으로 사용하고 ANSI 문자열을 포함하는 BSTR을 반환합니다. ANSI-유니코드 변환을 수행하지 않습니다.
SysAllocStringLen 새 문자열을 할당하고, 전달된 문자열에서 지정된 수의 문자를 복사하고, null 종료 문자를 추가합니다.

SysFreeString SysAllocString, SysAllocStringByteLen, SysReAllocString, SysAllocStringLen 또는 SysReAllocStringLen에서 이전에 할당한 문자열의 할당을 취소합니다.
SysReAllocString 이전에 할당된 문자열을 두 번째 문자열의 크기로 다시 할당하고 두 번째 문자열을 재할당된 메모리에 복사합니다.
SysReAllocStringLen 이전 BSTR에서 지정된 수의 문자를 포함하는 새 BSTR을 만들고 이전 BSTR을 해제합니다.
SysReleaseString 지정된 문자열의 고정 참조 수를 1씩 줄입니다. 이 개수가 0에 도달하면 해당 문자열의 메모리가 더 이상 해제되지 않습니다.
SysStringByteLen BSTR의 길이(바이트)를 반환합니다.
SysStringLen BSTR의 길이를 반환합니다.
SystemTimeToVariantTime 시스템 시간을 변형 표현으로 변환합니다.
UnRegisterTypeLib 시스템 레지스트리에서 형식 라이브러리 정보를 제거합니다. 이 API를 사용하여 애플리케이션이 자신을 제대로 제거할 수 있도록 합니다.
UnRegisterTypeLibForUser RegisterTypeLibForUser를 사용하여 등록된 형식 라이브러리 정보를 제거합니다.
VarAbs 변형의 절대값을 반환합니다.
VarAdd 두 변형의 합계를 반환합니다.

VarAnd 정수 계열 형식의 두 변형 간에 비트 And 연산을 수행합니다.
VarBoolFromCy 통화 값을 부울 값으로 변환합니다.
VarBoolFromDate 날짜 값을 부울 값으로 변환합니다.
VarBoolFromDec 10진수 값을 부울 값으로 변환합니다.
VarBoolFromDisp IDispatch instance 기본 속성을 부울 값으로 변환합니다.
VarBoolFromI1 char 값을 부울 값으로 변환합니다.
VarBoolFromI2 짧은 값을 부울 값으로 변환합니다.
VarBoolFromI4 long 값을 부울 값으로 변환합니다.
VarBoolFromI8 8 바이트 정수 값을 부울 값으로 변환합니다.
VarBoolFromR4 float 값을 부울 값으로 변환합니다.
VarBoolFromR8 double 값을 부울 값으로 변환합니다.
VarBoolFromStr OLECHAR 문자열을 부울 값으로 변환합니다.
VarBoolFromUI1

부호 없는 char 값을 부울 값으로 변환합니다.
VarBoolFromUI2 부호 없는 짧은 값을 부울 값으로 변환합니다.
VarBoolFromUI4 부호 없는 long 값을 부울 값으로 변환합니다.
VarBoolFromUI8 부울 값으로 8 바이트 부호 없는 정수 값을 변환합니다.
VarBstrCat BSTR 형식의 두 변형을 연결하고 결과 BSTR을 반환합니다.
VarBstrCmp BSTR 형식의 두 가지 변형을 비교합니다.
VarBstrFromBool 부울 값을 BSTR 값으로 변환합니다.
VarBstrFromCy 통화 값을 BSTR 값으로 변환합니다.
VarBstrFromDate 날짜 값을 BSTR 값으로 변환합니다.
VarBstrFromDec 10진수 값을 BSTR 값으로 변환합니다.
VarBstrFromDisp IDispatch instance 기본 속성을 BSTR 값으로 변환합니다.
VarBstrFromI1 char 값을 BSTR 값으로 변환합니다.
VarBstrFromI2 짧은 값을 BSTR 값으로 변환합니다.

VarBstrFromI4 긴 값을 BSTR 값으로 변환합니다.
VarBstrFromI8 부호 없는 8 바이트 정수 값을 BSTR 값으로 변환합니다. (VarBstrFromI8)
VarBstrFromR4 float 값을 BSTR 값으로 변환합니다.
VarBstrFromR8 double 값을 BSTR 값으로 변환합니다.
VarBstrFromUI1 부호 없는 char 값을 BSTR 값으로 변환합니다.
VarBstrFromUI2 부호 없는 짧은 값을 BSTR 값으로 변환합니다.
VarBstrFromUI4 부호 없는 long 값을 BSTR 값으로 변환합니다.
VarBstrFromUI8 부호 없는 8 바이트 정수 값을 BSTR 값으로 변환합니다. (VarBstrFromUI8)
VarCat 두 변형을 연결하고 결과를 반환합니다.
VarCmp 두 변형을 비교합니다.
VarCyAbs 형식 통화 변형의 절대값을 검색합니다.
VarCyAdd 통화 형식의 두 가지 변형을 추가합니다.
VarCyCmp

통화 형식의 두 가지 변형을 비교합니다.
VarCyCmpR8 형식 통화의 변형을 double 형식 값과 비교합니다.
VarCyFix 형식 통화 변형의 정수 부분을 검색합니다. (VarCyFix)
VarCyFromBool 부울 값을 통화 값으로 변환합니다.
VarCyFromDate 날짜 값을 통화 값으로 변환합니다.
VarCyFromDec 10진수 값을 통화 값으로 변환합니다.
VarCyFromDisp IDispatch instance 기본 속성을 통화 값으로 변환합니다.
VarCyFromI1 char 값을 통화 값으로 변환합니다.
VarCyFromI2 짧은 값을 통화 값으로 변환합니다.
VarCyFromI4 long 값을 통화 값으로 변환합니다.
VarCyFromI8 8 바이트 정수 값을 통화 값으로 변환합니다.
VarCyFromR4 float 값을 통화 값으로 변환합니다.
VarCyFromR8 double 값을 통화 값으로 변환합니다.

VarCyFromStr OLECHAR 문자열을 통화 값으로 변환합니다.
VarCyFromUI1 부호 없는 char 값을 통화 값으로 변환합니다.
VarCyFromUI2 부호 없는 짧은 값을 통화 값으로 변환합니다.
VarCyFromUI4 부호 없는 long 값을 통화 값으로 변환합니다.
VarCyFromUI8 부호 없는 8 바이트 정수 값을 통화 값으로 변환합니다.
VarCyInt 형식 통화 변형의 정수 부분을 검색합니다. (VarCyInt)
VarCyMul 통화 형식의 두 변형을 곱합니다.
VarCyMulI4 통화 값을 32비트 정수로 곱합니다.
VarCyMulI8 통화 값을 64비트 정수로 곱합니다.
VarCyNeg 형식 통화의 변형에 대해 논리적 부정을 수행합니다.
VarCyRound 형식 통화의 변형을 지정된 소수 자릿수로 반올림합니다.
VarCySub 두 가지 형식 통화 변형을 뺍니다.
VarDateFromBool

부울 값을 날짜 값으로 변환합니다.
VarDateFromCy 통화 값을 날짜 값으로 변환합니다.
VarDateFromDec 10진수 값을 날짜 값으로 변환합니다.
VarDateFromDisp IDispatch instance 기본 속성을 날짜 값으로 변환합니다.
VarDateFromI1 char 값을 날짜 값으로 변환합니다.
VarDateFromI2 짧은 값을 날짜 값으로 변환합니다.
VarDateFromI4 long 값을 날짜 값으로 변환합니다.
VarDateFromI8 부호 없는 8 바이트 정수 값을 날짜 값으로 변환합니다.
VarDateFromR4 float 값을 날짜 값으로 변환합니다.
VarDateFromR8 double 값을 날짜 값으로 변환합니다.
VarDateFromStr OLECHAR 문자열을 날짜 값으로 변환합니다.
VarDateFromUdate MS-DOS 형식에서 변형 형식으로 변환된 시간과 날짜를 변환합니다. (VarDateFromUdate)
VarDateFromUdateEx MS-DOS 형식에서 변형 형식으로 변환된 시간과 날짜를 변환합니다. (VarDateFromUdateEx)

VarDateFromUI1 부호 없는 char 값을 날짜 값으로 변환합니다.
VarDateFromUI2 부호 없는 짧은 값을 날짜 값으로 변환합니다.
VarDateFromUI4 서명되지 않은 long 값을 날짜 값으로 변환합니다.
VarDateFromUI8 8 바이트 부호 없는 값을 날짜 값으로 변환합니다.
VarDecAbs 10진수 형식의 변형 절대값을 검색합니다.
VarDecAdd 10진수 형식의 두 가지 변형을 추가합니다.
VarDecCmp 10진수 형식의 두 변형을 비교합니다.
VarDecCmpR8 10진수 형식의 변형과 double 형식의 값을 비교합니다.
VarDecDiv 10진수 형식의 두 변형을 나눕니다.
VarDecFix 10진수 형식의 변형 정수 부분을 검색합니다. (VarDecFix)
VarDecFromBool 부울 값을 10진수 값으로 변환합니다.
VarDecFromCy 통화 값을 10진수 값으로 변환합니다.
VarDecFromDate

날짜 값을 10진수 값으로 변환합니다.
VarDecFromDisp IDispatch instance 기본 속성을 10진수 값으로 변환합니다.
VarDecFromI1 char 값을 10진수 값으로 변환합니다.
VarDecFromI2 짧은 값을 10진수 값으로 변환합니다.
VarDecFromI4 long 값을 10진수 값으로 변환합니다.
VarDecFromI8 8 바이트 정수 값을 10진수 값으로 변환합니다.
VarDecFromR4 float 값을 10진수 값으로 변환합니다.
VarDecFromR8 double 값을 10진수 값으로 변환합니다.
VarDecFromStr OLECHAR 문자열을 10진수 값으로 변환합니다.
VarDecFromUI1 부호 없는 char 값을 10진수 값으로 변환합니다.
VarDecFromUI2 부호 없는 짧은 값을 10진수 값으로 변환합니다.
VarDecFromUI4 부호 없는 long 값을 10진수 값으로 변환합니다.
VarDecFromUI8 부호 없는 8 바이트 정수 값을 10진수 값으로 변환합니다.

VarDecInt 10진수 형식의 변형 정수 부분을 검색합니다. (VarDecInt)
VarDecMul 10진수 형식의 두 변형을 곱합니다.
VarDecNeg 10진수 형식의 변형에 대해 논리적 부정을 수행합니다.
VarDecRound 10진수 형식의 변형을 지정된 소수 자릿수로 반올림합니다.
VarDecSub 10진수 형식의 두 변형을 뺍니다.
VarDiv 두 변형을 나눈 결과를 반환합니다.
VarEqv 두 변형에 대해 비트 동등성을 수행합니다.
VarFix 변형의 정수 부분을 반환합니다. (VarFix)
VarFormat 형식 문자열을 구문 분석하여 변형 형식을 문자열 형식으로 지정합니다.
VarFormatCurrency 통화 값이 포함된 변형의 형식을 문자열 형식으로 지정합니다.
VarFormatDateTime 명명된 날짜 및 시간 정보를 포함하는 변형의 형식을 문자열로 지정합니다.
VarFormatFromTokens 토큰화된 형식 문자열을 가져와 변형에 적용하여 형식이 지정된 출력 문자열을 생성합니다.
VarFormatNumber

숫자가 포함된 변형의 형식을 문자열 형식으로 지정합니다.
VarFormatPercent 백분율이 포함된 변형의 형식을 문자열 형식으로 지정합니다.
VarI1FromBool 부울 값을 char 값으로 변환합니다.
VarI1FromCy 통화 값을 char 값으로 변환합니다.
VarI1FromDate 날짜 값을 char 값으로 변환합니다.
VarI1FromDec 10진수 값을 char 값으로 변환합니다.
VarI1FromDisp IDispatch instance 기본 속성을 char 값으로 변환합니다.
VarI1FromI2 짧은 값을 char 값으로 변환합니다.
VarI1FromI4 long 값을 char 값으로 변환합니다.
VarI1FromI8 8 바이트 정수 값을 char 값으로 변환합니다.
VarI1FromR4 float 값을 char 값으로 변환합니다.
VarI1FromR8 double 값을 char 값으로 변환합니다.
VarI1FromStr OLECHAR 문자열을 char 값으로 변환합니다.

VarI1FromUI1 부호 없는 char 값을 char 값으로 변환합니다.
VarI1FromUI2 부호 없는 짧은 값을 char 값으로 변환합니다.
VarI1FromUI4 부호 없는 long 값을 char 값으로 변환합니다.
VarI1FromUI8 부호 없는 8 바이트 정수 값을 char 값으로 변환합니다.
VarI2FromBool 부울 값을 짧은 값으로 변환합니다.
VarI2FromCy 통화 값을 짧은 값으로 변환합니다.
VarI2FromDate 날짜 값을 짧은 값으로 변환합니다.
VarI2FromDec 10진수 값을 짧은 값으로 변환합니다.
VarI2FromDisp IDispatch instance 기본 속성을 짧은 값으로 변환합니다.
VarI2FromI1 char 값을 짧은 값으로 변환합니다.
VarI2FromI4 long 값을 짧은 값으로 변환합니다.
VarI2FromI8 8 바이트 정수 값을 짧은 값으로 변환합니다.
VarI2FromR4

float 값을 짧은 값으로 변환합니다.
VarI2FromR8 double 값을 짧은 값으로 변환합니다.
VarI2FromStr OLECHAR 문자열을 짧은 값으로 변환합니다.
VarI2FromUI1 부호 없는 char 값을 짧은 값으로 변환합니다.
VarI2FromUI2 부호 없는 짧은 값을 짧은 값으로 변환합니다.
VarI2FromUI4 부호 없는 long 값을 짧은 값으로 변환합니다.
VarI2FromUI8 부호 없는 8 바이트 정수 값을 짧은 값으로 변환합니다.
VarI4FromBool 부울 값을 long 값으로 변환합니다.
VarI4FromCy 통화 값을 long 값으로 변환합니다.
VarI4FromDate 날짜 값을 long 값으로 변환합니다.
VarI4FromDec 10진수 값을 long 값으로 변환합니다.
VarI4FromDisp IDispatch instance 기본 속성을 long 값으로 변환합니다.
VarI4FromI1 char 값을 long 값으로 변환합니다.

VarI4FromI2 짧은 값을 long 값으로 변환합니다.
VarI4FromI4 int 값을 long 값으로 변환합니다.
VarI4FromI8 8 바이트 정수 값을 long 값으로 변환합니다.
VarI4FromR4 float 값을 long 값으로 변환합니다.
VarI4FromR8 double 값을 long 값으로 변환합니다.
VarI4FromStr OLECHAR 문자열을 long 값으로 변환합니다.
VarI4FromUI1 부호 없는 char 값을 long 값으로 변환합니다.
VarI4FromUI2 부호 없는 짧은 값을 long 값으로 변환합니다.
VarI4FromUI4 부호 없는 long 값을 long 값으로 변환합니다.
VarI4FromUI8 부호 없는 8 바이트 정수 값을 long로 변환합니다.
VarI8FromBool 부울 값을 8 바이트 정수 값으로 변환합니다.
VarI8FromCy 통화 값을 8 바이트 정수 값으로 변환합니다.
VarI8FromDate

날짜 값을 8 바이트 정수 값으로 변환합니다.
VarI8FromDec 10진수 값을 8 바이트 정수 값으로 변환합니다.
VarI8FromDisp IDispatch instance 기본 속성을 8 바이트 정수 값으로 변환합니다.
VarI8FromI1 char 값을 8 바이트 정수 값으로 변환합니다.
VarI8FromI2 짧은 값을 8 바이트 정수 값으로 변환합니다.
VarI8FromR4 float 값을 8 바이트 정수 값으로 변환합니다.
VarI8FromR8 double 값을 8 바이트 정수 값으로 변환합니다.
VarI8FromStr OLECHAR 문자열을 8 바이트 정수 값으로 변환합니다.
VarI8FromUI1 부호 없는 바이트 값을 8 바이트 정수 값으로 반전합니다.
VarI8FromUI2 부호 없는 짧은 값을 8 바이트 정수 값으로 변환합니다.
VarI8FromUI4 부호 없는 long 값을 8 바이트 정수 값으로 변환합니다.
VarI8FromUI8 부호 없는 8 바이트 정수 값을 8 바이트 정수 값으로 변환합니다.
VariantChangeType 변형을 한 형식에서 다른 형식으로 변환합니다.

VariantChangeTypeEx LCID를 사용하여 변형을 한 형식에서 다른 형식으로 변환합니다.
VariantClear 변형을 지웁니다.
VariantCopy 대상 변형을 해제하고 원본 변형의 복사본을 만듭니다.
VariantCopyInd 대상 변형을 해제하고 원본 변형의 복사본을 만들어 원본이 VT_BYREF 지정되는 경우 필요한 간접 참조를 수행합니다.
VariantInit 변형을 초기화합니다.
VariantTimeToDosDateTime 날짜 및 시간의 변형 표현을 MS-DOS 날짜 및 시간 값으로 변환합니다.
VariantTimeToSystemTime 시간의 변형 표현을 시스템 시간 값으로 변환합니다.
VarIdiv 모든 형식의 두 변형을 정수로 변환한 다음, 결과를 구분하여 반환합니다.
VarImp 두 변형에 대해 비트 의미를 수행합니다.
VarInt 변형의 정수 부분을 반환합니다. (VarInt)
VarMod 두 변형을 나누고 나머지만 반환합니다.
VarMonthName 지역화된 월 이름을 포함하는 문자열을 반환합니다.

VarMul 두 변형을 곱한 결과를 반환합니다.
VarNeg 변형에서 논리적 부정을 수행합니다.
VarNot 변형에 대해 부정하지 않는 비트 연산을 수행합니다.
VarNumFromParseNum 구문 분석된 결과를 변형으로 변환합니다.
VarOr 두 변형에 대해 논리적 분리를 수행합니다.
VarParseNumFromStr 문자열을 구문 분석하고 나타내는 숫자에 대한 형식 독립적 설명을 만듭니다.
VarPow 두 가지 변형으로 전원 함수를 수행한 결과를 반환합니다.
VarR4CmpR8 float 및 double 형식의 두 가지 변형을 비교합니다.
VarR4FromBool 부울 값을 float 값으로 변환합니다.
VarR4FromCy 통화 값을 float 값으로 변환합니다.
VarR4FromDate 날짜 값을 float 값으로 변환합니다.
VarR4FromDec 10진수 값을 float 값으로 변환합니다.
VarR4FromDisp

IDispatch instance 기본 속성을 float 값으로 변환합니다.
VarR4FromI1 char 값을 float 값으로 반전합니다.
VarR4FromI2 짧은 값을 float 값으로 변환합니다.
VarR4FromI4 long 값을 float 값으로 변환합니다.
VarR4FromI8 8 바이트 정수 값을 float 값으로 변환합니다.
VarR4FromR8 double 값을 float 값으로 변환합니다.
VarR4FromStr OLECHAR 문자열을 float 값으로 변환합니다.
VarR4FromUI1 부호 없는 char 값을 float 값으로 변환합니다.
VarR4FromUI2 부호 없는 짧은 값을 float 값으로 변환합니다.
VarR4FromUI4 부호 없는 long 값을 float 값으로 변환합니다.
VarR4FromUI8 부호 없는 8 바이트 정수 값을 float 값으로 변환합니다.
VarR8FromBool 부울 값을 double 값으로 변환합니다.
VarR8FromCy 통화 값을 double 값으로 변환합니다.

VarR8FromDate 날짜 값을 double 값으로 변환합니다.
VarR8FromDec 10진수 값을 double 값으로 변환합니다.
VarR8FromDisp IDispatch instance 기본 속성을 double 값으로 변환합니다.
VarR8FromI1 char 값을 double 값으로 변환합니다.
VarR8FromI2 짧은 값을 double 값으로 변환합니다.
VarR8FromI4 long 값을 double 값으로 변환합니다.
VarR8FromI8 8 바이트 정수 값을 double 값으로 변환합니다.
VarR8FromR4 float 값을 double 값으로 변환합니다.
VarR8FromStr OLECHAR 문자열을 double 값으로 변환합니다.
VarR8FromUI1 부호 없는 char 값을 double 값으로 변환합니다.
VarR8FromUI2 부호 없는 짧은 값을 double 값으로 변환합니다.
VarR8FromUI4 부호 없는 long 값을 double 값으로 변환합니다.
VarR8FromUI8

부호 없는 8 바이트 정수 값을 double 값으로 변환합니다.
VarR8Pow double 형식의 변형에 대해 전원 함수를 수행합니다.
VarR8Round double 형식의 변형을 지정된 소수 자릿수로 반올림합니다.
VarRound 변형을 지정된 소수 자릿수로 반올림합니다.
VarSub 두 가지 변형을 뺍니다.
VarTokenizeFormatString 실제 서식 문자열을 VarFormatFromTokens를 사용하여 변형의 서식을 지정하는 데 사용할 수 있는 일련의 토큰으로 구문 분석합니다.
VarUdateFromDate 변형 형식에서 MS-DOS 형식으로 변환된 시간과 날짜를 변환합니다.
VarUI1FromBool 부울 값을 부호 없는 char 값으로 변환합니다.
VarUI1FromCy 통화 값을 부호 없는 char 값으로 변환합니다.
VarUI1FromDate 날짜 값을 부호 없는 char 값으로 변환합니다.
VarUI1FromDec 10진수 값을 부호 없는 char 값으로 변환합니다.
VarUI1FromDisp IDispatch instance 기본 속성을 부호 없는 char 값으로 변환합니다.
VarUI1FromI1 char 값을 부호 없는 char 값으로 변환합니다.

VarUI1FromI2 짧은 값을 부호 없는 char 값으로 변환합니다.
VarUI1FromI4 long 값을 부호 없는 char 값으로 변환합니다.
VarUI1FromI8 8 바이트 정수 값을 바이트 값으로 변환합니다.
VarUI1FromR4 부동 소수자 값을 부호 없는 char 값으로 변환합니다.
VarUI1FromR8 double 값을 부호 없는 char 값으로 변환합니다.
VarUI1FromStr OLECHAR 문자열을 부호 없는 문자 문자열로 변환합니다.
VarUI1FromUI2 부호 없는 짧은 값을 부호 없는 char 값으로 변환합니다.
VarUI1FromUI4 부호 없는 long 값을 부호 없는 char 값으로 변환합니다.
VarUI1FromUI8 부호 없는 8 바이트 정수 값을 바이트 값으로 변환합니다.
VarUI2FromBool 부울 값을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromCy 통화 값을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromDate 날짜 값을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromDec

10진수 값을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromDisp IDispatch instance 기본 속성을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromI1 char 값을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromI2 짧은 값을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromI4 long 값을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromI8 8 바이트 정수 값을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromR4 부동 소수자 값을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromR8 double 값을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromStr OLECHAR 문자열을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromUI1 부호 없는 char 값을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromUI4 부호 없는 long 값을 부호 없는 짧은 값으로 변환합니다.
VarUI2FromUI8 부호 없는 8 바이트 정수 값을 부호 없는 짧은 값으로 변환합니다.
VarUI4FromBool 부울 값을 부호 없는 long 값으로 변환합니다.

VarUI4FromCy 통화 값을 부호 없는 long 값으로 변환합니다.
VarUI4FromDate 날짜 값을 부호 없는 long 값으로 변환합니다.
VarUI4FromDec 10진수 값을 부호 없는 long 값으로 변환합니다.
VarUI4FromDisp IDispatch instance 기본 속성을 부호 없는 long 값으로 변환합니다.
VarUI4FromI1 char 값을 부호 없는 long 값으로 변환합니다.
VarUI4FromI2 짧은 값을 부호 없는 long 값으로 변환합니다.
VarUI4FromI4 long 값을 부호 없는 long 값으로 변환합니다.
VarUI4FromI8 8 바이트 정수 값을 부호 없는 long 값으로 변환합니다.
VarUI4FromR4 부동 소수자 값을 부호 없는 long 값으로 변환합니다.
VarUI4FromR8 double 값을 부호 없는 long 값으로 변환합니다.
VarUI4FromStr OLECHAR 문자열을 부호 없는 long 값으로 변환합니다.
VarUI4FromUI1 부호 없는 char 값을 부호 없는 long 값으로 변환합니다.
VarUI4FromUI2

부호 없는 짧은 값을 부호 없는 long 값으로 변환합니다.
VarUI4FromUI8 부호 없는 8 바이트 정수 값을 부호 없는 long 값으로 변환합니다.
VarUI8FromBool VARIANT_BOOL 값을 8 바이트 부호 없는 정수 값으로 변환합니다.
VarUI8FromCy 통화 값을 8 바이트 부호 없는 정수 값으로 변환합니다.
VarUI8FromDate 날짜 값을 8 바이트 부호 없는 정수 값으로 변환합니다.
VarUI8FromDec 10진수 값을 8 바이트 부호 없는 정수 값으로 변환합니다.
VarUI8FromDisp IDispatch instance 기본 속성을 8 바이트 부호 없는 정수 값으로 변환합니다.
VarUI8FromI1 char 값을 8 바이트 부호 없는 정수 값으로 변환합니다.
VarUI8FromI2 짧은 값을 8 바이트 부호 없는 정수 값으로 변환합니다.
VarUI8FromI8 8 바이트 정수 값을 8 바이트 부호 없는 정수 값으로 변환합니다.
VarUI8FromR4 부동 소수점 값을 부호 없는 8 바이트 정수 값으로 변환합니다.
VarUI8FromR8 double 값을 8 바이트 부호 없는 정수 값으로 변환합니다.
VarUI8FromStr OLECHAR 문자열을 8 바이트 부호 없는 정수 값으로 변환합니다.

VarUI8FromUI1 바이트 값을 8 바이트 부호 없는 정수 값으로 변환합니다.
VarUI8FromUI2 부호 없는 짧은 값을 8 바이트 부호 없는 정수 값으로 변환합니다.
VarUI8FromUI4 부호 없는 long 값을 8 바이트 부호 없는 정수 값으로 변환합니다.
VarWeekdayName 평일의 지역화된 이름을 포함하는 문자열을 반환합니다.
VarXor 두 가지 변형에 대해 논리적 제외를 수행합니다.
VectorFromBstr BSTR의 각 문자를 벡터의 요소에 할당하는 벡터를 반환합니다.

구조체

INTERFACEDATA 개체의 속성 및 메서드에 대해 설명합니다.
METHODDATA 메서드 또는 속성에 대해 설명합니다.
NUMPARSE 숫자 구문 분석 정보를 지정합니다.
PARAMDATA 메서드 또는 속성에서 허용하는 매개 변수에 대해 설명합니다.

열거형

REGKIND 형식 라이브러리를 등록하는 방법을 제어합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

olectl.h 헤더

아티클2023. 08. 25.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Automation
- COM(구성 요소 개체 모델)
- Windows 및 메시지

olectl.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DllRegisterServer 이 서버 모듈에서 지원되는 모든 클래스에 대한 레지스트리 항목을 만들도록 In-Process 서버에 지시합니다.
DllUnregisterServer In Process 서버에 DllRegisterServer를 통해 만든 항목만 제거하도록 지시합니다.
OleCreateFontIndirect FONTDESC 구조체의 글꼴 속성에 대한 초기 설명을 사용하여 표준 글꼴 개체를 만들고 초기화합니다.
OleCreatePictureIndirect PICTDESC 구조체에 따라 초기화된 새 그림 개체를 만듭니다.
OleCreatePropertyFrame 새 속성 프레임, 즉 부모가 hwndOwner인 속성 시트 대화 상자를 호출합니다. 여기서 대화 상자는 부모 창의 지점(x,y)에 배치되고 캡션 lpszCaption이 있습니다.
OleCreatePropertyFrameIndirect OleCreatePropertyFrame을 호출할 때와 같이 별도의 매개 변수를 지정하지 않고 매개 변수가 포함된 구조체(OCPFIPARAMS)를 기반으로 속성 시트 대화 상자를 만듭니다.
OleIconToCursor 아이콘을 커서로 변환합니다.

OleLoadPicture 새 그림 개체를 만들고 스트림의 내용에서 초기화합니다. 이는 NULL을 첫 번째 매개 변수로 사용하여 OleCreatePictureIndirect를 호출한 다음 IPersistStream::Load를 호출하는 것과 같습니다. (OleLoadPicture)
OleLoadPictureEx 새 그림 개체를 만들고 스트림의 내용에서 초기화합니다. 이는 NULL을 첫 번째 매개 변수로 사용하여 OleCreatePictureIndirect를 호출한 다음 IPersistStream::Load를 호출하는 것과 같습니다. (OleLoadPictureEx)
OleLoadPictureFile 디스크의 그림 파일에서 IPictureDisp 개체를 만듭니다.
OleLoadPictureFileEx 파일에서 그림을 로드합니다.
OleLoadPicturePath 새 그림 개체를 만들고 스트림의 내용에서 초기화합니다. 이는 OleCreatePictureIndirect(NULL, ...) 뒤에 IPersistStream::Load를 호출하는 것과 같습니다.
OleSavePictureFile 그림을 파일에 저장합니다.
OleTranslateColor OLE_COLOR 형식을 COLORREF로 변환합니다.

구조체

FONTDESC OleCreateFontIndirect 함수를 통해 글꼴 개체를 만드는 데 사용되는 매개 변수를 포함합니다.
OCPFIPARAMS OleCreatePropertyFrameIndirect 함수를 통해 속성 시트 대화 상자를 호출하는 데 사용되는 매개 변수를 포함합니다.

PICTDESC OleCreatePictureIndirect 함수를 통해 그림 개체를 만드는 매개 변수를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

oledbguid.h 헤더

아티클2023. 03. 08.

이 헤더는 인덱싱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 인덱싱 서비스

oledbguid.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

Dbid DBID 구조체는 데이터베이스 개체를 식별하는 다양한 방법을 캡슐화합니다.

열거형

DBKINDENUM DBKINDENUM 열거형 형식은 데이터베이스 개체를 식별하는 데 사용할 GUID, 속성 번호 또는 속성 이름의 조합을 지정합니다.

oledlg.h 헤더

아티클2023. 03. 16.

이 헤더는 COM(구성 요소 개체 모델)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)

oledlg.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IOleUILinkContainerA 컨테이너에서 구현되고 OLE 일반 대화 상자에서 사용됩니다. 컨테이너의 링크를 관리하는 데 필요한 메서드를 제공하여 이러한 대화 상자를 지원합니다. (ANSI)
IOleUILinkContainerW 컨테이너에서 구현되고 OLE 일반 대화 상자에서 사용됩니다. 컨테이너의 링크를 관리하는 데 필요한 메서드를 제공하여 이러한 대화 상자를 지원합니다. (유니코드)
IOleUILinkInfoA IOleUILinkContainer 인터페이스의 확장입니다. 개체가 마지막으로 업데이트된 시간(IOleUILinkContainer에서 제공하지 않는 링크 정보)을 반환합니다. (ANSI)
IOleUILinkInfoW IOleUILinkContainer 인터페이스의 확장입니다. 개체가 마지막으로 업데이트된 시간(IOleUILinkContainer에서 제공하지 않는 링크 정보)을 반환합니다. (유니코드)
IOleUIObjInfoA 컨테이너에서 구현되고 컨테이너의 개체 속성 대화 상자 및 변환 대화 상자에서 사용됩니다. (ANSI)
IOleUIObjInfoW 컨테이너에서 구현되고 컨테이너의 개체 속성 대화 상자 및 변환 대화 상자에서 사용됩니다. (유니코드)

Functions

OleUIAddVerbMenuA 지정한 개체의 동사 메뉴를 지정된 메뉴에 추가합니다. (ANSI)
OleUIAddVerbMenuW 지정한 개체의 동사 메뉴를 지정된 메뉴에 추가합니다. (유니코드)
OleUIBusyA 사용자가 동시성을 관리할 수 있도록 표준 사용 중인 대화 상자를 호출합니다. (ANSI)
OleUIBusyW 사용자가 동시성을 관리할 수 있도록 표준 사용 중인 대화 상자를 호출합니다. (유니코드)
OleUICanConvertOrActivateAs 레지스트리에 지정된 CLSID를 변환하거나 활성화하는 데 사용할 수 있는 OLE 개체 클래스가 있는지 확인합니다.
OleUIChangeIconA 사용자가 아이콘 파일, 실행 파일 또는 DLL에서 아이콘을 선택할 수 있도록 하는 표준 변경 아이콘 대화 상자를 호출합니다. (ANSI)
OleUIChangeIconW 사용자가 아이콘 파일, 실행 파일 또는 DLL에서 아이콘을 선택할 수 있도록 하는 표준 변경 아이콘 대화 상자를 호출합니다. (유니코드)
OleUIChangeSourceA 원본 변경 대화 상자를 호출하여 사용자가 링크의 원본을 변경할 수 있도록 합니다. (ANSI)
OleUIChangeSourceW 원본 변경 대화 상자를 호출하여 사용자가 링크의 원본을 변경할 수 있도록 합니다. (유니코드)
OleUIConvertA 사용자가 지정된 단일 개체의 형식 또는 지정된 개체 클래스의 모든 OLE 개체 형식을 변경할 수 있도록 표준 변환 대화 상자를 호출합니다. (ANSI)
OleUIConvertW 사용자가 지정된 단일 개체의 형식 또는 지정된 개체 클래스의 모든 OLE 개체 형식을 변경할 수 있도록 표준 변환 대화 상자를 호출합니다. (유니코드)

OleUIEditLinksA 표준 링크 대화 상자를 호출하여 사용자가 컨테이너의 연결된 개체를 수정할 수 있도록 합니다. (ANSI)
OleUIEditLinksW 표준 링크 대화 상자를 호출하여 사용자가 컨테이너의 연결된 개체를 수정할 수 있도록 합니다. (유니코드)
OleUIInsertObjectA 사용자가 개체 원본 및 클래스 이름과 개체를 자체 또는 아이콘으로 표시하는 옵션을 선택할 수 있는 표준 개체 삽입 대화 상자를 호출합니다. (ANSI)
OleUIInsertObjectW 사용자가 개체 원본 및 클래스 이름과 개체를 자체 또는 아이콘으로 표시하는 옵션을 선택할 수 있는 표준 개체 삽입 대화 상자를 호출합니다. (유니코드)
OleUIObjectPropertiesA 개체에 대한 일반, 보기 및 링크 정보를 표시하는 개체 속성 대화 상자를 호출합니다. (ANSI)
OleUIObjectPropertiesW 개체에 대한 일반, 보기 및 링크 정보를 표시하는 개체 속성 대화 상자를 호출합니다. (유니코드)
OleUIPasteSpecialA 표준 붙여넣기 특수 대화 상자를 호출하여 사용자가 붙여넣거나 붙여넣을 클립보드 개체의 형식을 선택할 수 있도록 합니다. (ANSI)
OleUIPasteSpecialW 표준 붙여넣기 특수 대화 상자를 호출하여 사용자가 붙여넣거나 붙여넣을 클립보드 개체의 형식을 선택할 수 있도록 합니다. (유니코드)
OleUIPromptUserA 지정된 템플릿이 있는 대화 상자를 표시하고 사용자의 응답(단추 식별자)을 반환합니다. 이 함수는 OLE 경고 메시지(예: 등록되지 않은 클래스)를 표시하는 데 사용됩니다. (ANSI)
OleUIPromptUserW 지정된 템플릿이 있는 대화 상자를 표시하고 사용자의 응답(단추 식별자)을 반환합니다. 이 함수는 OLE 경고 메시지(예: 등록되지 않은 클래스)를 표시하는 데 사용됩니다. (유니코드)

OleUIUpdateLinksA 링크 컨테이너의 모든 링크를 업데이트 업데이트 프로세스의 진행률을 보여 주는 대화 상자를 표시합니다. 사용자가 중지 단추를 누르거나 모든 링크가 처리될 때 프로세스가 중지됩니다. (ANSI)
OleUIUpdateLinksW 링크 컨테이너의 모든 링크를 업데이트 업데이트 프로세스의 진행률을 보여 주는 대화 상자를 표시합니다. 사용자가 중지 단추를 누르거나 모든 링크가 처리될 때 프로세스가 중지됩니다. (유니코드)

구조체

OLEUIBUSYA OLE 사용자 인터페이스 라이브러리가 사용 중인 대화 상자를 초기화하는 데 사용하는 정보와 대화 상자가 해제될 때 라이브러리가 정보를 반환할 수 있는 공간이 포함되어 있습니다. (ANSI)
OLEUIBUSYW OLE 사용자 인터페이스 라이브러리가 사용 중인 대화 상자를 초기화하는 데 사용하는 정보와 대화 상자가 해제될 때 라이브러리가 정보를 반환할 수 있는 공간이 포함되어 있습니다. (유니코드)
OLEUICHANGEICONA OLE 사용자 인터페이스 라이브러리가 변경 아이콘 대화 상자를 초기화하는 데 사용하는 정보를 포함하며, 대화 상자가 해제될 때 라이브러리가 정보를 반환할 수 있는 공간이 포함되어 있습니다. (ANSI)
OLEUICHANGEICONW OLE 사용자 인터페이스 라이브러리가 변경 아이콘 대화 상자를 초기화하는 데 사용하는 정보를 포함하며, 대화 상자가 해제될 때 라이브러리가 정보를 반환할 수 있는 공간이 포함되어 있습니다. (유니코드)
OLEUICHANGESOURCEA 표준 원본 변경 대화 상자를 초기화하는 데 사용되는 정보를 포함합니다. (ANSI)
OLEUICHANGESOURCEW 표준 원본 변경 대화 상자를 초기화하는 데 사용되는 정보를 포함합니다. (유니코드)

OLEUICONVERTA OLE 사용자 인터페이스 라이브러리가 변환 대화 상자를 초기화하는 데 사용하는 정보와 대화 상자가 해제될 때 라이브러리가 정보를 반환할 수 있는 공간을 포함합니다. (ANSI)
OLEUICONVERTW OLE 사용자 인터페이스 라이브러리가 변환 대화 상자를 초기화하는 데 사용하는 정보와 대화 상자가 해제될 때 라이브러리가 정보를 반환할 수 있는 공간을 포함합니다. (유니코드)
OLEUIEDITLINKSA OLE 사용자 인터페이스 라이브러리가 링크 편집 대화 상자를 초기화하는 데 사용하는 정보를 포함하고, 대화 상자가 해제될 때 라이브러리가 정보를 반환할 수 있는 공간을 포함합니다. (ANSI)
OLEUIEDITLINKSW OLE 사용자 인터페이스 라이브러리가 링크 편집 대화 상자를 초기화하는 데 사용하는 정보를 포함하고, 대화 상자가 해제될 때 라이브러리가 정보를 반환할 수 있는 공간을 포함합니다. (유니코드)
OLEUIGNRLPROPSA 개체 속성 대화 상자의 일반 탭을 초기화합니다. (ANSI)
OLEUIGNRLPROPSW 개체 속성 대화 상자의 일반 탭을 초기화합니다. (유니코드)
OLEUIINSERTOBJECTA OLE 사용자 인터페이스 라이브러리가 개체 삽입 대화 상자를 초기화하는 데 사용하는 정보와 대화 상자가 해제될 때 라이브러리가 정보를 반환할 수 있는 공간을 포함합니다. (ANSI)
OLEUIINSERTOBJECTW OLE 사용자 인터페이스 라이브러리가 개체 삽입 대화 상자를 초기화하는 데 사용하는 정보와 대화 상자가 해제될 때 라이브러리가 정보를 반환할 수 있는 공간을 포함합니다. (유니코드)
OLEUILINKPROPSA 개체 속성 대화 상자의 링크 탭을 초기화하는 데 사용되는 정보를 포함합니다. (ANSI)
OLEUILINKPROPSW 개체 속성 대화 상자의 링크 탭을 초기화하는 데 사용되는 정보를 포함합니다. (유니코드)
OLEUIOBJECTPROPSA 표준 개체 속성 대화 상자를 초기화하는 데 사용되는 정보를 포함합니다. (ANSI)

OLEUIOBJECTPROPSW 표준 개체 속성 대화 상자를 초기화하는 데 사용되는 정보를 포함합니다. (유니코드)
OLEUIPASTEENTRYA 특수 붙여넣기 대화 상자의 OLEUIPASTESPECIAL 구조에 지정할 항목의 배열입니다. (ANSI)
OLEUIPASTEENTRYW 특수 붙여넣기 대화 상자의 OLEUIPASTESPECIAL 구조에 지정할 항목의 배열입니다. (유니코드)
OLEUIPASTESPECIALA OLE 사용자 인터페이스 라이브러리에서 특수 붙여넣기 대화 상자를 초기화하는 데 사용하는 정보와 대화 상자가 해제될 때 라이브러리가 정보를 반환할 수 있는 공간이 포함되어 있습니다. (ANSI)
OLEUIPASTESPECIALW OLE 사용자 인터페이스 라이브러리에서 특수 붙여넣기 대화 상자를 초기화하는 데 사용하는 정보와 대화 상자가 해제될 때 라이브러리가 정보를 반환할 수 있는 공간이 포함되어 있습니다. (유니코드)
OLEUIVIEWPROPSA 개체 속성 대화 상자의 보기 탭을 초기화하는 데 사용되는 정보를 포함합니다. (ANSI)
OLEUIVIEWPROPSW 개체 속성 대화 상자의 보기 탭을 초기화하는 데 사용되는 정보를 포함합니다. (유니코드)

열거형

OLEUIPASTEFLAG 이 형식을 붙여넣을 때 사용할 수 있는 사용자 옵션과 선택 항목 그룹 또는 목록(붙여넣기, 링크 붙여넣기 등) 내의 사용자 옵션을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

oleidl.h 헤더

아티클2023. 08. 25.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)
- Windows 셸

oleidl.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDropSource IDropSource 인터페이스는 애플리케이션에서 끌어서 놓기 작업을 제공하기 위해 구현하는 인터페이스 중 하나입니다.
IDropSourceNotify IDropSourceNotify 인터페이스는 IDropSource 개체에서 구현되어 사용자가 잠재적인 놓기 대상 창으로 마우스를 끌거나 끌 때 OLE에서 알림을 받습니다.
Idroptarget IDropTarget 인터페이스는 애플리케이션에서 끌어서 놓기 작업을 제공하기 위해 구현하는 인터페이스 중 하나입니다.
IEnterpriseDropTarget 놓기 대상 애플리케이션에서 구현되는 경우 이 인터페이스는 OLE 끌어서 놓기 엔진에 놓기 대상 애플리케이션이 엔터프라이즈 보호 정책을 평가할 것인지 여부를 결정하는 기능을 제공하고 OLE 끌어서 놓기 엔진에 드롭 소스 애플리케이션의 엔터프라이즈 ID를 놓기 대상 애플리케이션에 제공하는 방법을 제공합니다.
IEnumOLEVERB 개체에 사용할 수 있는 다른 동사를 오름차순으로 열거합니다. IEnumOLEVERB 인터페이스를 구현하는 열거자는 IOleObject::EnumVerbs에서 반환됩니다.
IOleAdviseHolder 개체 서버에서 권고 연결 및 복합 문서 알림을 관리합니다.
IOleCache

개체 내부에 캐시되는 프레젠테이션 데이터를 제어합니다. 캐시된 프레젠테이션 데이터는 서버 애플리케이션이 실행되고 있지 않거나 사용할 수 없는 경우에도 개체의 컨테이너에서 사용할 수 있습니다.
IOleCache2 개체 클라이언트가 IOleCache::Cache를 사용하여 만든 각 캐시를 선택적으로 업데이트할 수 있도록 합니다.
IOleCacheControl 캐시의 적절한 유지 관리를 제공합니다. 실행 중인 개체의 IDataObject 구현을 캐시에 연결하여 캐시를 유지 관리하므로 캐시가 실행 중인 개체에서 알림을 받을 수 있습니다.
IOleClientSite 포함된 개체가 표시 사이트의 위치 및 범위, 모니커, 사용자 인터페이스 및 컨테이너에서 제공하는 기타 리소스에 대한 정보를 가져오는 기본 수단을 제공합니다.
IOleContainer 복합 문서에서 개체를 열거하거나 실행 중인 상태에서 컨테이너를 잠급니다. 컨테이너 및 개체 애플리케이션은 모두 이 인터페이스를 구현합니다.
IOleInPlaceActiveObject 현재 위치 개체와 연결된 애플리케이션의 가장 바깥 부분 프레임 창과 포함된 개체가 포함된 애플리케이션 내의 문서 창 간에 직접 통신 채널을 제공합니다.
IOleInPlaceFrame 컨테이너의 최상위 프레임 창을 제어합니다.
IOleInPlaceObject 내부 개체의 활성화 및 비활성화를 관리하고 표시할 내부 개체 크기를 결정합니다.
IOleInPlaceSite 컨테이너와 개체의 현재 위치 클라이언트 사이트 간의 상호 작용을 관리합니다. 클라이언트 사이트는 포함된 개체의 표시 사이트이며 개체에 대한 위치 및 개념 정보를 제공합니다.
IOleInPlaceUIWindow 컨테이너 애플리케이션에서 구현되고 개체 애플리케이션에서 문서 또는 프레임 창의 테두리 공간을 협상하는 데 사용됩니다.
IOleItemContainer

항목 모니커가 식별하는 개체에 바인딩할 때 사용됩니다.
IOleLink 연결된 개체가 연결과 관련된 함수를 컨테이너에 제공할 수 있도록 합니다.
IOleObject 포함된 개체가 컨테이너에 기본 기능을 제공하고 통신하는 주체 수단으로 사용됩니다.
IOleWindow IOleWindow 인터페이스는 애플리케이션이 현재 위치 활성화에 참여하는 다양한 창에 대한 핸들을 가져오고 상황에 맞는 도움말 모드를 입력하고 종료할 수 있는 메서드를 제공합니다.
IParseDisplayName 표시 가능한 이름 문자열을 구문 분석하여 사용자 지정 모니커 구현을 위한 모니커로 변환합니다.
IViewObject 호출자에게 데이터 개체를 전달하지 않고 개체가 직접 표시되도록 설정합니다. 또한 이 인터페이스는 뷰 개체의 변경 내용을 호출자에게 알 수 있도록 advise 싱크와의 연결을 만들고 관리할 수 있습니다.
IViewObject2 개체의 지정된 뷰에 대한 드로잉 크기를 반환하는 IViewObject 인터페이스의 확장입니다. IOleObject::GetExtent 대신 이 메서드를 호출하여 개체가 아직 실행되고 있지 않은 경우 개체가 실행되지 않도록 방지할 수 있습니다.

Functions

CLIPFORMAT_UserFree CLIPFORMAT_UserFree 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
CLIPFORMAT_UserFree64 CLIPFORMAT_UserFree64 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
CLIPFORMAT_UserMarshal

CLIPFORMAT_UserMarshal 함수(oleidl.h)는 CLIPFORMAT 개체를 RPC 버퍼로 마샬링합니다.
CLIPFORMAT_UserMarshal64 CLIPFORMAT_UserMarshal64 함수(oleidl.h)는 CLIPFORMAT 개체를 RPC 버퍼로 마샬링합니다.
CLIPFORMAT_UserSize CLIPFORMAT_UserSize 함수(oleidl.h)는 CLIPFORMAT 개체의 와이어 크기를 계산하고 해당 핸들과 데이터를 가져옵니다.
CLIPFORMAT_UserSize64 CLIPFORMAT_UserSize64 함수(oleidl.h)는 CLIPFORMAT 개체의 와이어 크기를 계산하고 해당 핸들과 데이터를 가져옵니다.
CLIPFORMAT_UserUnmarshal CLIPFORMAT_UserUnmarshal 함수(oleidl.h)는 RPC 버퍼에서 CLIPFORMAT 개체를 숨깁니다.
CLIPFORMAT_UserUnmarshal64 CLIPFORMAT_UserUnmarshal64 함수(oleidl.h)는 RPC 버퍼에서 CLIPFORMAT 개체를 숨깁니다.
HACCEL_UserFree HACCEL_UserFree 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HACCEL_UserFree64 HACCEL_UserFree64 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HACCEL_UserMarshal HACCEL을 RPC 버퍼로 마샬링합니다.
HACCEL_UserMarshal64 HACCEL 개체를 RPC 버퍼로 마샬링합니다.
HACCEL_UserSize HACCEL_UserSize 함수(oleidl.h)는 HACCEL 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
HACCEL_UserSize64

HACCEL_UserSize64 함수(oaild.h)는 HACCEL 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
HACCEL_UserUnmarshal HACCEL_UserUnmarshal 함수(oleidl.h)는 RPC 버퍼에서 HACCEL 개체를 숨기지 않습니다.
HACCEL_UserUnmarshal64 HACCEL_UserUnmarshal64 함수(oleidl.h)는 RPC 버퍼에서 HACCEL 개체를 숨기지 않습니다.
HDC_UserFree HDC_UserFree 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HDC_UserFree64 HDC_UserFree64 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HDC_UserMarshal HDC_UserMarshal 함수(oleidl.h)는 HDC 개체를 RPC 버퍼로 마샬링합니다.
HDC_UserMarshal64 HDC_UserMarshal64 함수(oleidl.h)는 HDC 개체를 RPC 버퍼로 마샬링합니다.
HDC_UserSize HDC_UserSize 함수(oleidl.h)는 HDC 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
HDC_UserSize64 HDC_UserSize64 함수(oleidl.h)는 HDC 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
HDC_UserUnmarshal HDC_UserUnmarshal 함수(oleidl.h)는 RPC 버퍼에서 HDC 개체를 숨기지 않습니다.
HDC_UserUnmarshal64 HDC_UserUnmarshal64 함수(oleidl.h)는 RPC 버퍼에서 HDC 개체를 숨기지 않습니다.
HGLOBAL_UserFree HGLOBAL_UserFree 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.

HGLOBAL_UserFree64 HGLOBAL_UserFree64 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HGLOBAL_UserMarshal HGLOBAL을 RPC 버퍼로 마샬링합니다.
HGLOBAL_UserMarshal64 HGLOBAL 개체를 RPC 버퍼로 마샬링합니다.
HGLOBAL_UserSize HGLOBAL_UserSize 함수(oleidl.h)는 HGLOBAL 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
HGLOBAL_UserSize64 HGLOBAL_UserSize64 함수(oleidl.h)는 HGLOBAL 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
HGLOBAL_UserUnmarshal HGLOBAL_UserUnmarshal 함수(oleidl.h)는 RPC 버퍼에서 HGLOBAL 개체를 숨기지 않습니다.
HGLOBAL_UserUnmarshal64 HGLOBAL_UserUnmarshal64 함수(oleidl.h)는 RPC 버퍼에서 HGLOBAL 개체를 숨기지 않습니다.
HMENU_UserFree HMENU_UserFree 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HMENU_UserFree64 HMENU_UserFree64 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HMENU_UserMarshal HMENU_UserMarshal 함수(oleidl.h)는 HMENU를 RPC 버퍼로 마샬링합니다.
HMENU_UserMarshal64 HMENU_UserMarshal64 함수(oleidl.h)는 HMENU를 RPC 버퍼로 마샬링합니다.

HMENU_UserSize HMENU_UserSize 함수(oleidl.h)는 HMENU 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
HMENU_UserSize64 HMENU_UserSize64 함수(oleidl.h)는 HMENU 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
HMENU_UserUnmarshal HMENU_UserUnmarshal 함수(oleidl.h)는 RPC 버퍼에서 HMENU 개체를 숨기지 않습니다.
HMENU_UserUnmarshal64 HMENU_UserUnmarshal64 함수(oleidl.h)는 RPC 버퍼에서 HMENU 개체를 숨기지 않습니다.
HWND_UserFree HWND_UserFree 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HWND_UserFree64 HWND_UserFree64 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HWND_UserMarshal HWND_UserMarshal 함수(oleidl.h)는 HWND 개체를 RPC 버퍼로 마샬링합니다.
HWND_UserMarshal64 HWND_UserMarshal64 함수(oleidl.h)는 HWND 개체를 RPC 버퍼로 마샬링합니다.
HWND_UserSize HWND_UserSize 함수(oleidl.h)는 HWND 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
HWND_UserSize64 HWND_UserSize64 함수(oleidl.h)는 HWND 개체의 와이어 크기를 계산하고 핸들과 데이터를 가져옵니다.
HWND_UserUnmarshal HWND_UserUnmarshal 함수(oleidl.h)는 RPC 버퍼에서 HWND 개체를 숨기지 않습니다.

HWND_UserUnmarshal64 HWND_UserUnmarshal64 함수(oleidl.h)는 RPC 버퍼에서 HWND 개체를 숨기지 않습니다.
STGMEDIUM_UserFree STGMEDIUM_UserFree 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
STGMEDIUM_UserFree64 STGMEDIUM_UserFree64 함수(oleidl.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
STGMEDIUM_UserMarshal STGMEDIUM_UserMarshal 함수(oleidl.h)는 STGMEDIUM 개체를 RPC 버퍼로 마샬링합니다.
STGMEDIUM_UserMarshal64 STGMEDIUM_UserMarshal64 함수(oleidl.h)는 STGMEDIUM 개체를 RPC 버퍼로 마샬링합니다.
STGMEDIUM_UserSize STGMEDIUM_UserSize 함수(oleidl.h)는 STGMEDIUM 개체의 와이어 크기를 계산하고 해당 핸들과 데이터를 가져옵니다.
STGMEDIUM_UserSize64 STGMEDIUM_UserSize64 함수(oleidl.h)는 STGMEDIUM 개체의 와이어 크기를 계산하고 해당 핸들과 데이터를 가져옵니다.
STGMEDIUM_UserUnmarshal STGMEDIUM_UserUnmarshal 함수(oleidl.h)는 RPC 버퍼에서 STGMEDIUM 개체를 숨기지 않습니다.
STGMEDIUM_UserUnmarshal64 STGMEDIUM_UserUnmarshal64 함수(oleidl.h)는 RPC 버퍼에서 STGMEDIUM 개체를 숨기지 않습니다.

구조체

OBJECTDESCRIPTOR

CF_OBJECTDESRIPTOR 및 CF_LINKSRCDESCRIPTOR 파일 형식에 사용되는 데이터 구조를 나타냅니다.
OLEINPLACEFRAMEINFO 현재 위치 세션 동안 컨테이너에서 지원하는 가속기에 대한 정보를 포함합니다. 구조체는 IOleInPlaceSite::GetWindowContext 메서드 및 OleTranslateAccelerator 함수에 사용됩니다.
OLEMENUGROUPWIDTHS 현재 위치 편집 세션 동안 컨테이너와 개체 서버 간에 공유되는 메뉴의 6개 메뉴 그룹 각각에 있는 메뉴 항목 수를 나타냅니다. 공유 메뉴를 빌드하기 위한 메커니즘입니다.
OLEVERB 개체가 지원하는 동사를 정의합니다. IOleObject::EnumVerbs 메서드는 개체에 대해 이러한 구조를 열거할 수 있는 열거자를 만들고 열거자의 IEnumOLEVERB에 대한 포인터를 제공합니다.

열거형

BINDSPEED 호출자가 개체에 바인딩될 때까지 대기하는 시간을 대략적으로 나타냅니다.
DISCARDCACHE 더티 비트가 설정된 경우 메모리에서 삭제할 캐시로 수행할 작업을 지정합니다.
OLECLOSE 닫기 전에 개체를 저장해야 하는지 여부를 나타냅니다.
OLECONTF 열거할 개체의 형식을 나타냅니다.
OLEGETMONIKER IOleObject::GetMoniker 및 IOleClientSite::GetMoniker 메서드의 동작 측면을 제어합니다.
OLELINKBIND 링크 원본에 대한 바인딩 작업을 제어합니다.
OLEMISC

개체 또는 개체 클래스의 기타 특성을 설명합니다.
OLERENDER 새로 만든 개체에 대해 요청된 캐싱 유형을 나타냅니다.
OLEUPDATE 연결된 개체가 연결된 개체의 캐시된 데이터를 자동으로 업데이트하는지 또는 컨테이너가 IOleObject::Update 또는 IOleLink::Update 메서드를 호출할 때만 업데이트하는지 여부를 나타냅니다. 상수는 IOleLink 인터페이스에서 사용됩니다.
OLEVERBATTRIB 개체에 대해 지정된 동사의 특성을 설명합니다.
OLEWHICHMK 개체의 모니커가 설정되거나 검색되는 부분을 나타냅니다.
USERCLASSTYPE 개체 클래스와 연결된 표시 이름의 다양한 변형을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ondemandconnroutehelper.h 헤더

아티클2023. 08. 25.

이 헤더는 네트워크 목록 관리자에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네트워크 목록 관리자

ondemandconnroutehelper.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

FreeInterfaceContextTable 이 함수는 GetInterfaceContextTableForHostName 함수를 사용하여 검색된 인터페이스 컨텍스트 테이블을 해제합니다.
GetInterfaceContextTableForHostName 이 함수는 지정된 호스트 이름 및 연결 프로필 필터에 대한 인터페이스 컨텍스트 테이블을 검색합니다.
OnDemandGetRoutingHint OnDemandGetRoutingHint 함수는 경로 요청 캐시에서 대상을 조회하고 일치하는 항목이 발견되면 해당 인터페이스 ID를 반환합니다.
OnDemandRegisterNotification OnDemandRegisterNotification 함수를 사용하면 경로 요청 캐시가 수정될 때 애플리케이션이 등록 알림을 받을 수 있습니다.
OnDemandUnRegisterNotification OnDemandUnregisterNotification 함수를 사용하면 애플리케이션이 알림을 등록 취소하고 리소스를 클린 수 있습니다.

구조체

NET_INTERFACE_CONTEXT NET_INTERFACE_CONTEXT_TABLE 구조체의 일부인 인터페이스 컨텍스트입니다.

NET_INTERFACE_CONTEXT_TABLE NET_INTERFACE_CONTEXT 구조체의 테이블입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

oobenotification.h 헤더

아티클2023. 03. 08.

이 헤더는 OOBE API에서 설정 상태 및 알림에 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 설정 상태 및 알림에 대한 OOBE API

oobenotification.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

OOBEComplete OOBE(Windows 시작)가 완료되었는지 여부를 확인합니다.
RegisterWaitUntilOOBECompleted OOBE(Windows 시작)가 완료되면 호출할 콜백을 등록합니다.
UnregisterWaitUntilOOBECompleted RegisterWaitUntilOOBECompleted를 통해 이전에 등록된 콜백을 등록 취소합니다.

콜백 함수

OOBE_COMPLETED_CALLBACK RegisterWaitUntilOOBECompleted 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.

opmapi.h 헤더

아티클2023. 08. 25.

이 헤더는 Microsoft Media Foundation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft Media Foundation

opmapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IOPMVideoOutput OPM(출력 보호 관리자) 세션에 대한 비디오 출력을 나타냅니다.

Functions

OPMGetVideoOutputForTarget 지정된 어댑터의 VidPN 대상에 대한 비디오 출력 개체를 반환합니다.
OPMGetVideoOutputsFromHMONITOR 특정 HMONITOR 핸들과 연결된 각 물리적 모니터에 대한 OPM(Output Protection Manager) 개체를 만듭니다.
OPMGetVideoOutputsFromIDirect3DDevice9Object 특정 Direct3D 디바이스와 연결된 각 물리적 모니터에 대한 OPM(출력 보호 관리자) 개체를 만듭니다.

구조체

OPM_ACP_AND_CGMSA_SIGNALING

OPM_GET_ACP_AND_CGMSA_SIGNALING 쿼리의 결과를 포함합니다.
OPM_ACTUAL_OUTPUT_FORMAT OPM(출력 보호 관리자)의 OPM_GET_ACTUAL_OUTPUT_FORMAT 쿼리 결과를 포함합니다.
OPM_CONFIGURE_PARAMETERS OPM(출력 보호 관리자) 또는 COPP(Certified Output Protection Manager) 명령을 포함합니다.
OPM_CONNECTED_HDCP_DEVICE_INFORMATION OPM_GET_CONNECTED_HDCP_DEVICE_INFORMATION 쿼리의 결과를 포함합니다.
OPM_COPP_COMPATIBLE_GET_INFO_PARAMETERS IOPMVideoOutput::COPPCompatibleGetInformation 메서드에 대한 매개 변수를 포함합니다.
OPM_ENCRYPTED_INITIALIZATION_PARAMETERS OPM(출력 보호 관리자) 세션에 대한 암호화된 초기화 매개 변수를 포함합니다. (OPM_ENCRYPTED_INITIALIZATION_PARAMETERS).
OPM_GET_CODEC_INFO_INFORMATION OPM_GET_CODEC_INFO_INFORMATION(opmapi.h)에는 OPM_GET_CODEC_INFO 쿼리의 결과가 포함됩니다.
OPM_GET_CODEC_INFO_PARAMETERS OPM_GET_CODEC_INFO_PARAMETERS(opmapi.h)에는 OPM_GET_CODEC_INFO 명령에 대한 정보가 포함되어 있습니다.
OPM_GET_INFO_PARAMETERS OPM_GET_INFO_PARAMETERS(opmapi.h)에는 IOPMVideoOutput::GetInformation 메서드에 대한 매개 변수가 포함되어 있습니다.
OPM_HDCP_KEY_SELECTION_VECTOR High-Bandwidth HDCP(Digital Content Protection) 수신기에 대한 KSV(키 선택 벡터)를 포함합니다.
OPM_OMAC OPM_OMAC(opmapi.h)에는 OPM(출력 보호 관리자) 메시지에 대한 MAC(메시지 인증 코드)가 포함되어 있습니다.
OPM_OUTPUT_ID_DATA

OPM_GET_OUTPUT_ID 상태 요청의 결과를 포함합니다.
OPM_RANDOM_NUMBER OPM_RANDOM_NUMBER(opmapi.h)에는 OPM(출력 보호 관리자)에 사용할 128비트 난수가 포함되어 있습니다.
OPM_REQUESTED_INFORMATION OPM_REQUESTED_INFORMATION(opmapi.h)에는 OPM(출력 보호 관리자) 상태 요청의 결과가 포함됩니다.
OPM_SET_ACP_AND_CGMSA_SIGNALING_PARAMETERS OPM(출력 보호 관리자)의 OPM_SET_ACP_AND_CGMSA_SIGNALING 명령에 대한 정보를 포함합니다.
OPM_SET_HDCP_SRM_PARAMETERS OPM_SET_HDCP_SRM 명령에 대한 매개 변수를 포함합니다.
OPM_SET_PROTECTION_LEVEL_PARAMETERS OPM(출력 보호 관리자)의 OPM_SET_PROTECTION_LEVEL 명령에 대한 데이터를 포함합니다.
OPM_STANDARD_INFORMATION OPM_STANDARD_INFORMATION(opmapi.h)에는 OPM(출력 보호 관리자) 상태 요청의 결과가 포함됩니다.

열거형

OPM_ACP_PROTECTION_LEVEL ACP(아날로그 복사 보호)에 대한 보호 수준을 지정합니다.
OPM_DPCP_PROTECTION_LEVEL DisplayPort 콘텐츠 보호(DPCP)에 대한 보호 수준을 지정합니다.
OPM_HDCP_PROTECTION_LEVEL High-Bandwidth HDCP(Digital Content Protection)에 대한 보호 수준을 지정합니다.
OPM_IMAGE_ASPECT_RATIO_EN300294

ETSI EN 300 294의 가로 세로 비율을 지정합니다.
OPM_VIDEO_OUTPUT_SEMANTICS IOPMVideoOutput 인터페이스가 OPM(출력 보호 관리자) 의미 체계 또는 COPP(Certified Output Protection Protocol) 의미 체계를 사용할지 여부를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

p2p.h 헤더

아티클2023. 08. 25.

이 헤더는 피어 투 피어에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 피어 투 피어

p2p.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

PeerCollabAddContact 피어의 연락처 목록에 연락처를 추가합니다.
PeerCollabAsyncInviteContact 보안 연결을 통해 보낸 사람의 피어 공동 작업 활동에 참가하기 위해 신뢰할 수 있는 피어 연락처에 초대를 보냅니다. 초대 응답의 가용성은 비동기 이벤트를 통해 업데이트됩니다.
PeerCollabAsyncInviteEndpoint 지정된 피어 엔드포인트에 초대를 보내 보낸 사람의 피어 공동 작업 활동에 참가합니다. 초대에 대한 응답의 가용성은 비동기 이벤트를 통해 업데이트됩니다.
PeerCollabCancelInvitation 이전에 발신자가 연락처에 보낸 초대를 취소합니다.
PeerCollabCloseHandle 피어 협업 활동 초대에 대한 핸들을 닫습니다.
PeerCollabDeleteContact 호출자와 연결된 로컬 연락처 저장소에서 연락처를 삭제합니다.
PeerCollabDeleteEndpointData 제공된 엔드포인트 데이터와 일치하는 호출 피어 노드에서 피어 엔드포인트 데이터를 삭제합니다.
PeerCollabDeleteObject 호출 엔드포인트에서 피어 개체를 삭제합니다.

PeerCollabEnumApplicationRegistrationInfo 피어 애플리케이션 정보를 검색하는 데 사용되는 열거형 핸들을 가져옵니다.
PeerCollabEnumApplications 특정 피어의 엔드포인트에 등록된 애플리케이션을 포함하는 열거형에 대한 핸들을 반환합니다.
PeerCollabEnumContacts 호출 피어에서 현재 사용할 수 있는 모든 피어 공동 작업 네트워크 연락처를 포함하는 열거형 집합에 대한 핸들을 반환합니다.
PeerCollabEnumEndpoints 특정 피어 연락처와 연결된 엔드포인트를 포함하는 열거형에 대한 핸들을 반환합니다.
PeerCollabEnumObjects 특정 피어의 엔드포인트와 연결된 피어 개체를 포함하는 열거형에 대한 핸들을 반환합니다.
PeerCollabEnumPeopleNearMe 호출 피어의 서브넷에서 현재 사용할 수 있는 모든 피어 협업 네트워크 "내 근처에 있는 사람" 엔드포인트가 포함된 열거형 집합에 대한 핸들을 반환합니다.
PeerCollabExportContact 피어 이름과 연결된 연락처 데이터를 문자열 버퍼로 내보냅니다. 버퍼에는 XML 형식의 연락처 데이터가 포함됩니다.
PeerCollabGetAppLaunchInfo 연락처 이름, 피어 엔드포인트 및 초대 요청을 포함하여 피어 애플리케이션 시작 정보를 가져옵니다.
PeerCollabGetApplicationRegistrationInfo 애플리케이션별 등록 정보를 가져옵니다.
PeerCollabGetContact 연락처의 피어 이름이 지정된 피어 연락처에 대한 정보를 가져옵니다.
PeerCollabGetEndpointName 이전에 PeerCollabSetEndpointName을 호출하여 설정한 대로 호출 피어의 현재 엔드포인트 이름을 검색합니다.

PeerCollabGetEventData 피어에서 발생한 피어 협업 이벤트와 연결된 데이터를 가져옵니다.
PeerCollabGetInvitationResponse 이전에 피어 협업 활동에 참여하도록 초대된 피어에서 응답을 가져옵니다.
PeerCollabGetPresenceInfo 특정 연락처와 연결된 엔드포인트에 대한 현재 상태 정보를 검색합니다.
PeerCollabGetSigninOptions 피어의 현재 로그인된 피어 공동 작업 네트워크 현재 상태 옵션을 가져옵니다.
PeerCollabInviteContact 피어 공동 작업 활동에 참가하기 위한 초대를 신뢰할 수 있는 연락처에 보냅니다. 이 호출은 동기적이며, 성공하면 연락처로부터 응답을 받습니다.
PeerCollabInviteEndpoint 지정된 피어 엔드포인트에 초대를 보내 보낸 사람의 피어 공동 작업 활동에 참가합니다. 이 호출은 동기적이며, 성공하면 피어 엔드포인트에서 응답을 가져옵니다.
PeerCollabParseContact 연락처 XML 데이터가 포함된 유니코드 문자열 버퍼를 PEER_CONTACT 데이터 구조로 구문 분석합니다.
PeerCollabQueryContactData 제공된 피어 엔드포인트에 대한 연락처 정보를 검색합니다.
PeerCollabRefreshEndpointData 호출 피어 노드를 새 엔드포인트 데이터로 업데이트.
PeerCollabRegisterApplication 피어 공동 작업에서 시작할 수 있도록 로컬 컴퓨터에 애플리케이션을 등록합니다.
PeerCollabRegisterEvent 애플리케이션을 피어 협업 인프라에 등록하여 특정 피어 협업 이벤트에 대한 콜백을 받습니다.
PeerCollabSetEndpointName

피어 애플리케이션에서 사용하는 현재 엔드포인트의 이름을 설정합니다.
PeerCollabSetObject 피어 협업 네트워크에서 사용되는 피어 데이터 개체를 만들거나 업데이트합니다.
PeerCollabSetPresenceInfo 발신자의 현재 상태 정보를 보고 있는 모든 연락처에 업데이트.
PeerCollabShutdown 피어 협업 인프라를 종료하고 연결된 모든 리소스를 해제합니다.
PeerCollabSignin 피어를 호스트된 인터넷(서버리스 프레즌스) 또는 서브넷("사람 Near Me") 피어 협업 네트워크 프레즌스 공급자에 서명합니다.
PeerCollabSignout 특정 유형의 피어 협업 네트워크 프레즌스 공급자에서 피어를 서명합니다.
PeerCollabStartup 피어 협업 인프라를 초기화합니다.
PeerCollabSubscribeEndpointData 사용 가능한 엔드포인트에 대한 구독을 만듭니다.
PeerCollabUnregisterApplication 로컬 컴퓨터에서 피어의 특정 애플리케이션을 등록 취소합니다.
PeerCollabUnregisterEvent 특정 피어 협업 이벤트에 대한 알림에서 애플리케이션을 등록 취소합니다.
PeerCollabUnsubscribeEndpointData PeerCollabSubscribeEndpointData를 사용하여 만든 엔드포인트에 대한 구독을 제거합니다.
PeerCollabUpdateContact 호출자의 로컬 연락처 저장소에 지정된 피어 연락처와 연결된 정보를 업데이트.
PeerCreatePeerName

PeerCreatePeerName 함수는 지정된 피어 ID 및 분류자의 기존 이름을 기반으로 새 이름을 만듭니다. 그러나 PeerCreatePeerName을 호출하면 새 ID가 만들어지지 않습니다.
PeerEndEnumeration PeerEndEnumeration 함수는 열거형(예: 레코드 또는 멤버 열거형)을 해제하고 열거형과 연결된 모든 리소스의 할당을 취소합니다.
PeerEnumGroups PeerEnumGroups 함수는 특정 피어 ID와 연결된 모든 피어 그룹을 열거하는 데 사용되는 피어 열거형 핸들을 만들고 반환합니다.
PeerEnumIdentities PeerEnumIdentities 함수는 특정 사용자에 속한 모든 피어 ID를 열거하는 데 사용되는 피어 열거형 핸들을 만들고 반환합니다.
PeerFreeData PeerFreeData 함수는 데이터 블록을 할당 취소하고 메모리 풀에 반환합니다. PeerFreeData 함수를 사용하여 피어 Identity Manager, 피어 그룹화 및 피어 협업 API가 반환하는 데이터를 해제합니다.
PeerGetItemCount PeerGetItemCount 함수는 피어 열거형의 항목 수를 반환합니다.
PeerGetNextItem PeerGetNextItem 함수는 피어 열거형에서 특정 개수의 항목을 반환합니다.
PeerGraphAddRecord PeerGraphAddRecord 함수는 피어 그래프에 새 레코드를 추가합니다. 이 함수와 함께 추가된 레코드는 피어 그래프의 각 노드로 전송됩니다.
PeerGraphClose PeerGraphClose 함수는 PeerGraphCreate 또는 PeerGraphOpen 호출에서 반환된 피어 그래프 핸들을 무효화하고 지정된 피어 그래프에 대한 모든 네트워크 연결을 닫습니다.
PeerGraphCloseDirectConnection PeerGraphCloseDirectConnection 함수는 지정된 직접 연결을 닫습니다.
PeerGraphConnect PeerGraphConnect 함수는 피어 그래프에서 지정된 노드에 연결을 시도합니다.

PeerGraph만들기 PeerGraphCreate 함수는 새 피어 그래프를 만듭니다. 애플리케이션은 피어 그래프에 대한 정보와 피어 그래프에서 사용하는 보안 유형을 지정할 수 있습니다. 피어 그래프에 대한 핸들이 반환되지만 네트워크 연결은 설정되지 않습니다.
PeerGraphDelete PeerGraphDelete 함수는 지정된 피어 그래프와 연결된 데이터를 삭제합니다.
PeerGraphDeleteRecord PeerGraphDeleteRecord 함수는 레코드를 피어 그래프 내에서 삭제된 것으로 표시합니다. 레코드는 로컬 노드에서 함수 호출을 사용할 수 없습니다(예: PeerGraphGetRecord 및 PeerGraphEnumRecords 호출).
PeerGraphEndEnumeration PeerGraphEndEnumeration 함수는 열거형 핸들을 해제하고 열거형과 연결된 리소스를 해제합니다.
PeerGraphEnumConnections PeerGraphEnumConnections 함수는 로컬 노드의 연결을 열거하는 데 사용되는 열거형 핸들을 만들고 반환합니다.
PeerGraphEnumNodes PeerGraphEnumNodes 함수는 피어 그래프에서 노드를 열거하는 데 사용되는 열거형 핸들을 만들고 반환합니다.
PeerGraphEnumRecords PeerGraphEnumRecords 함수는 특정 유형의 레코드, 사용자 또는 둘 다의 레코드를 열거하는 데 사용되는 열거형 핸들을 만들고 반환합니다. 열거형은 열거형이 수행될 때 레코드의 스냅샷 제공합니다.
PeerGraphExportDatabase PeerGraphExportDatabase 함수는 피어 그래프 데이터베이스를 다른 컴퓨터로 이동할 수 있는 파일로 내보냅니다. PeerGraphImportDatabase를 사용하면 피어 그래프 데이터베이스를 다른 컴퓨터로 가져올 수 있습니다.
PeerGraphFreeData PeerGraphFreeData 함수는 여러 Peer Graphing API 함수가 반환하는 리소스를 해제합니다.
PeerGraphGetEventData

PeerGraphGetEventData 함수는 피어 이벤트를 검색합니다. 애플리케이션은 반환 값 PEER_S_NO_EVENT_DATA 반환될 때까지 이 함수를 호출합니다. 이는 호출이 성공했지만 검색할 피어 이벤트가 더 이상 없음을 나타냅니다.
PeerGraphGetItemCount PeerGraphGetItemCount 함수는 열거형의 항목 수를 검색합니다.
PeerGraphGetNextItem 다음 함수를 호출하여 만든 열거형의 다음 항목 또는 항목을 가져옵니다.
PeerGraphGetNodeInfo PeerGraphGetNodeInfo 함수는 특정 노드에 대한 정보를 검색합니다.
PeerGraphGetProperties PeerGraphGetProperties 함수는 현재 피어 그래프 속성을 검색합니다.
PeerGraphGetRecord PeerGraphGetRecord 함수는 지정된 레코드 ID를 기반으로 특정 레코드를 검색합니다. 반환된 레코드는 PeerGraphFreeData를 호출하여 해제해야 합니다.
PeerGraphGetStatus PeerGraphGetStatus 함수는 피어 그래프의 현재 상태 반환합니다.
PeerGraphImportDatabase PeerGraphImportDatabase 함수는 피어 그래프 데이터베이스의 정보가 포함된 파일을 가져옵니다. 이 함수는 애플리케이션이 아직 PeerGraphListen 또는 PeerGraphConnect 함수를 호출하지 않은 경우에만 호출할 수 있습니다.
PeerGraphListen PeerGraphListen 함수는 피어 그래프가 들어오는 연결을 수신 대기하기 시작해야 했음을 나타냅니다.
PeerGraphOpen PeerGraphOpen 함수는 로컬 노드 또는 원격 노드에서 이전에 만든 피어 그래프를 엽니다. 피어 그래프에 대한 핸들이 반환되지만 네트워크 연결이 설정되지 않았습니다.
PeerGraphOpenDirectConnection PeerGraphOpenDirectConnection 함수를 사용하면 애플리케이션이 피어 그래프의 노드와 직접 연결을 설정할 수 있습니다.

PeerGraphPeerTimeToUniversalTime PeerGraphPeerTimeToUniversalTime 함수는 피어 그래프 유지 관리 참조 시간 값을 피어의 컴퓨터에 표시하기에 적합한 지역화된 시간 값으로 변환합니다.
PeerGraphRegisterEvent PeerGraphRegisterEvent 함수는 피어 그래프 및 이벤트 유형과 관련된 변경 내용에 대한 알림을 받을 피어의 요청을 등록합니다.
PeerGraphSearchRecords PeerGraphSearchRecords 함수는 피어 그래프에서 특정 레코드를 검색합니다.
PeerGraphSendData PeerGraphSendData 함수는 인접 노드 또는 직접 연결된 노드로 데이터를 보냅니다.
PeerGraphSetNodeAttributes PeerGraphSetNodeAttributes 함수는 로컬 노드에 대한 PEER_NODE_INFO 구조의 특성을 설정합니다.
PeerGraphSetPresence PeerGraphSetPresence 함수는 특정 노드에 대한 현재 상태 레코드 게시를 명시적으로 켜거나 끕니다.
PeerGraphSetProperties PeerGraphSetProperties 함수는 피어 그래프 속성을 설정합니다.
PeerGraphShutdown PeerGraphShutdown 함수는 PeerGraphStartup 호출에 의해 할당된 모든 리소스를 정리합니다. PeerGraphStartup에 대한 각 호출에 대해 PeerGraphShutdown에 대한 호출이 있어야 합니다.
PeerGraphStartup PeerGraphStartup 함수는 피어 그래프 인프라에 호출 애플리케이션에 필요한 피어 프로토콜 버전을 나타냅니다.
PeerGraphUniversalTimeToPeerTime PeerGraphUniversalTimeToPeerTime 함수는 피어의 컴퓨터에서 공용 피어 그래프 시간 값으로 유니버설 시간 값을 변환합니다.
PeerGraphUnregisterEvent

PeerGraphUnregisterEvent 함수는 애플리케이션에 피어 그래프 및 레코드 형식과 관련된 변경 내용에 대한 알림을 더 이상 받지 않도록 요청합니다.
PeerGraphUpdateRecord PeerGraphUpdateRecord 함수는 피어 그래프의 레코드를 업데이트한 다음, 피어 그래프의 각 노드에 레코드를 플러드합니다.
PeerGraphValidateDeferredRecords PeerGraphValidateDeferredRecords 함수는 피어 그래프 인프라에 보안 모듈의 유효성을 검사하기 위해 지연된 레코드를 다시 제출할 때임을 나타냅니다.
PeerGroupAddRecord PeerGroupAddRecord 함수는 참여하는 모든 피어에 전파되는 새 레코드를 피어 그룹에 추가합니다.
PeerGroupClose PeerGroupClose 함수는 PeerGroupCreate, PeerGroupJoin 또는 PeerGroupOpen 함수에 대한 이전 호출에서 얻은 피어 그룹 핸들을 무효화합니다.
PeerGroupCloseDirectConnection PeerGroupCloseDirectConnection 함수는 두 피어 간의 특정 직접 연결을 닫습니다.
PeerGroupConnect PeerGroupConnect 함수는 피어 그룹에 대한 PNRP 검색을 시작하고 연결을 시도합니다. 이 함수가 성공적으로 호출되면 피어는 피어 그룹의 다른 멤버와 통신할 수 있습니다.
PeerGroupConnectByAddress 알려진 IPv6 주소를 가진 다른 피어가 참여하는 피어 그룹에 연결하려고 시도합니다.
PeerGroup만들기 PeerGroupCreate 함수는 새 피어 그룹을 만듭니다.
PeerGroupCreateInvitation PeerGroupCreateInvitation 함수는 지정된 피어에서 그룹을 조인하는 데 사용할 수 있는 XML 문자열을 반환합니다.
PeerGroupCreatePasswordInvitation 지정된 피어에서 일치하는 암호를 사용하여 그룹을 조인하는 데 사용할 수 있는 XML 문자열을 반환합니다.

PeerGroupDelete PeerGroupDelete 함수는 피어 그룹과 연결된 로컬 데이터 및 인증서를 삭제합니다.
PeerGroupDeleteRecord PeerGroupDeleteRecord 함수는 피어 그룹에서 레코드를 삭제합니다. 작성자와 관리 역할의 다른 멤버는 특정 레코드를 삭제할 수 있습니다.
PeerGroupEnumConnections PeerGroupEnumConnections 함수는 피어에서 현재 활성화된 연결의 열거형을 만듭니다.
PeerGroupEnumMembers PeerGroupEnumMembers 함수는 사용 가능한 피어 그룹 멤버 및 연결된 멤버 자격 정보의 열거형을 만듭니다.
PeerGroupEnumRecords PeerGroupEnumRecords 함수는 피어 그룹 레코드의 열거형을 만듭니다.
PeerGroupExportConfig PeerGroupExportConfig 함수는 피어에 대한 그룹 구성을 ID, 그룹 이름 및 ID에 대한 GMC를 포함하는 XML 문자열로 내보냅니다.
PeerGroupExportDatabase PeerGroupExportDatabase 함수는 피어 그룹 데이터베이스를 특정 파일로 내보냅니다. 이 파일은 다른 컴퓨터로 전송되고 PeerGroupImportDatabase 함수를 사용하여 가져올 수 있습니다.
PeerGroupGetEventData PeerGroupGetEventData 함수를 사용하면 애플리케이션이 그룹화 이벤트에서 반환된 데이터를 검색할 수 있습니다.
PeerGroupGetProperties PeerGroupGetProperties 함수는 지정된 그룹의 속성에 대한 정보를 검색합니다.
PeerGroupGetRecord PeerGroupGetRecord 함수는 특정 그룹 레코드를 검색합니다.
PeerGroupGetStatus PeerGroupGetStatus 함수는 그룹의 현재 상태 검색합니다.

PeerGroupImportConfig PeerGroupImportConfig 함수는 제공된 XML 구성 문자열의 특정 설정에 따라 ID에 대한 피어 그룹 구성을 가져옵니다.
PeerGroupImportDatabase PeerGroupImportDatabase 함수는 로컬 파일에서 피어 그룹 데이터베이스를 가져옵니다.
PeerGroupIssueCredentials PeerGroupIssueCredentials 함수는 GMC를 포함한 자격 증명을 특정 ID에 발급하고, 필요에 따라 초대된 피어가 피어 그룹에 조인하는 데 사용할 수 있는 초대 XML 문자열을 반환합니다.
PeerGroupJoin PeerGroupJoin 함수는 PeerGroupConnect 또는 PeerGroupConnectByAddress를 호출하기 전에 초대가 포함된 피어를 준비하여 기존 피어 그룹에 조인합니다.
PeerGroupOpen PeerGroupOpen 함수는 피어가 만들거나 조인한 피어 그룹을 엽니다. 피어 그룹을 연 후 피어는 이벤트 알림에 등록할 수 있습니다.
PeerGroupOpenDirectConnection PeerGroupOpenDirectConnection 함수는 피어 그룹의 다른 피어와 직접 연결을 설정합니다.
PeerGroupParseInvitation PeerGroupParseInvitation 함수는 특정 초대의 세부 정보가 포함된 PEER_INVITATION_INFO 구조를 반환합니다.
PeerGroupPasswordJoin PeerGroupConnect 또는 PeerGroupConnectByAddress를 호출하기 전에 암호로 보호된 피어 그룹에 조인할 초대 및 올바른 암호를 사용하여 피어를 준비합니다.
PeerGroupPeerTimeToUniversalTime PeerGroupPeerTimeToUniversalTime 함수는 피어 그룹 유지 관리 참조 시간 값을 피어 컴퓨터에 표시하기에 적합한 지역화된 시간 값으로 변환합니다.
PeerGroupRegisterEvent PeerGroupRegisterEvent 함수는 특정 피어 그룹 이벤트에 대한 피어를 등록합니다.
PeerGroupSearchRecords

PeerGroupSearchRecords 함수는 로컬 피어 그룹 데이터베이스에서 제공된 조건과 일치하는 레코드를 검색합니다.
PeerGroupSendData PeerGroupSendData 함수는 인접 또는 직접 연결을 통해 멤버에게 데이터를 보냅니다.
PeerGroupSetProperties PeerGroupSetProperties 함수는 현재 피어 그룹 속성을 설정합니다. 이 API 버전 1.0에서는 피어 그룹의 작성자만 이 작업을 수행할 수 있습니다.
PeerGroupShutdown PeerGroupShutdown 함수는 PeerGroupStartup을 사용하여 만든 피어 그룹을 닫고 할당된 리소스를 삭제합니다.
PeerGroupStartup PeerGroupStartup 함수는 요청된 버전의 피어 인프라를 사용하여 피어 그룹을 시작합니다.
PeerGroupUniversalTimeToPeerTime PeerGroupUniversalTimeToPeerTime 함수는 로컬 시간 값을 피어의 컴퓨터에서 공통 피어 그룹 시간 값으로 변환합니다.
PeerGroupUnregisterEvent PeerGroupUnregisterEvent 함수는 제공된 이벤트 핸들과 연결된 피어 이벤트의 알림에서 피어를 등록 취소합니다.
PeerGroupUpdateRecord PeerGroupUpdateRecord 함수는 특정 피어 그룹 내의 레코드를 업데이트합니다.
PeerHostNameToPeerName PeerNameToPeerHostName에서 반환된 호스트 이름을 나타내는 피어 이름 문자열로 디코딩합니다.
PeerIdentity만들기 PeerIdentityCreate 함수는 새 피어 ID를 만들고 해당 이름을 반환합니다.
PeerIdentityDelete PeerIdentityDelete 함수는 피어 ID를 영구적으로 삭제합니다. 여기에는 지정된 피어 ID와 연결된 모든 인증서, 개인 키 및 모든 그룹 정보를 제거하는 것이 포함됩니다.

PeerIdentityExport PeerIdentityExport 함수를 사용하면 사용자가 하나의 피어 ID를 내보낼 수 있습니다. 그러면 사용자는 피어 ID를 다른 컴퓨터로 전송할 수 있습니다.
PeerIdentityGetCryptKey PeerIdentityGetCryptKey 함수는 CSP(암호화 서비스 공급자)에 대한 핸들을 검색합니다.
PeerIdentityGetDefault PeerIdentityGetDefault 함수는 현재 사용자의 기본 피어 이름 집합을 검색합니다.
PeerIdentityGetFriendlyName PeerIdentityGetFriendlyName 함수는 피어 ID의 이름을 반환합니다.
PeerIdentityGetXML PeerIdentityGetXML 함수는 피어 ID에 대한 설명을 반환합니다. 그러면 타사에 전달되고 피어 ID를 피어 그룹에 초대하는 데 사용할 수 있습니다. 이 정보는 XML 조각으로 반환됩니다.
PeerIdentityImport PeerIdentityImport 함수는 하나의 피어 ID를 가져옵니다. 피어 ID가 컴퓨터에 있으면 PEER_E_ALREADY_EXISTS 반환됩니다.
PeerIdentitySetFriendlyName PeerIdentitySetFriendlyName 함수는 지정된 피어 ID의 이름을 수정합니다. 식별 이름은 사람이 읽을 수 있는 이름입니다.
PeerNameToPeerHostName 제공된 피어 이름을 getaddrinfo Windows Sockets 함수에 대한 후속 호출과 함께 사용할 수 있는 형식으로 인코딩합니다.
PeerPnrpEndResolve PeerPnrpStartResolve에 대한 이전 호출로 시작된 비동기 PNRP 확인 작업에 대한 핸들을 닫습니다.
PeerPnrpGetCloudInfo 호출 피어가 참여하는 PNRP(피어 이름 확인 프로토콜) 클라우드에 대한 정보를 검색합니다.
PeerPnrpGetEndpoint 비동기 피어 이름 확인 작업 중에 확인된 피어 엔드포인트 주소를 검색합니다.

PeerPnrpRegister PNRP 클라우드에 피어를 등록하고 등록 업데이트에 사용할 수 있는 핸들을 반환합니다.
PeerPnrpResolve 특정 피어 이름에 대해 등록된 엔드포인트 주소를 가져옵니다.
PeerPnrpShutdown PNRP(피어 이름 확인 프로토콜) 서비스의 실행 중인 instance 종료하고 연결된 모든 리소스를 해제합니다.
PeerPnrpStartResolve 비동기 피어 이름 확인 작업을 시작합니다.
PeerPnrpStartup 호출 피어에 대한 PNRP(피어 이름 확인 프로토콜) 서비스를 시작합니다.
PeerPnrpUnregister PNRP 클라우드에서 피어를 등록 취소합니다.
PeerPnrpUpdateRegistration 이름에 대한 PNRP 등록 정보를 업데이트.

콜백 함수

PFNPEER_FREE_SECURITY_DATA PFNPEER_FREE_SECURITY_DATA 콜백은 피어 그래프 인프라가 PFNPEER_SECURE_RECORD 및 PFNPEER_VALIDATE_RECORD 콜백에서 반환된 데이터를 해제하기 위해 호출하는 함수를 지정합니다.
PFNPEER_SECURE_RECORD PFNPEER_SECURE_RECORD 콜백은 피어 그래프 인프라가 레코드를 보호하기 위해 호출하는 함수를 지정합니다.
PFNPEER_VALIDATE_RECORD

PFNPEER_VALIDATE_RECORD 콜백은 Peer Graphing Infrastructure가 레코드의 유효성을 검사하기 위해 호출하는 함수를 지정합니다.

구조체

PEER_ADDRESS PEER_ADDRESS 구조체는 IP 주소에 대한 정보를 지정합니다.
PEER_APP_LAUNCH_INFO PEER_APP_LAUNCH_INFO 구조에는 이전 피어 초대 요청에서 연락처가 제공한 피어 애플리케이션 시작 정보가 포함됩니다.
PEER_APPLICATION PEER_APPLICATION 구조에는 피어 협업 네트워크 내에서 신뢰할 수 있는 연락처와 등록 및 공유할 수 있는 로컬로 설치된 소프트웨어 애플리케이션 또는 구성 요소를 설명하는 데이터가 포함되어 있습니다.
PEER_APPLICATION_REGISTRATION_INFO PEER_APPLICATION_REGISTRATION_INFO 구조에는 로컬 컴퓨터에 등록하기 위한 피어 애플리케이션 정보가 포함되어 있습니다.
PEER_COLLAB_EVENT_DATA PEER_COLLAB_EVENT_DATA 구조체(p2p.h)에는 피어에서 발생할 수 있는 각 가능한 피어 협업 네트워크 이벤트에 대한 변형 데이터가 포함됩니다.
PEER_COLLAB_EVENT_REGISTRATION PEER_COLLAB_EVENT_REGISTRATION 구조에는 피어가 특정 피어 협업 네트워크 이벤트에 등록하는 데 사용하는 데이터가 포함됩니다.
PEER_CONNECTION_INFO PEER_CONNECTION_INFO 구조에는 연결에 대한 정보가 포함됩니다. 이 구조체는 피어 그래프를 열거하거나 연결을 그룹화할 때 반환됩니다.
PEER_CONTACT PEER_CONTACT 구조에는 특정 연락처에 대한 정보가 포함됩니다.

PEER_CREDENTIAL_INFO PEER_CREDENTIAL_INFO 구조는 피어의 보안 자격 증명을 가져오고 발급하는 데 사용되는 정보를 정의합니다.
PEER_DATA PEER_DATA 구조에는 이진 데이터가 포함됩니다.
PEER_ENDPOINT PEER_ENDPOINT 구조체에는 피어 엔드포인트의 주소와 식별 이름이 포함됩니다.
PEER_EVENT_APPLICATION_CHANGED_DATA PEER_EVENT_APPLICATION_CHANGED_DATA 구조에는 피어 협업 네트워크에 참여하는 피어에서 PEER_EVENT_ENDPOINT_APPLICATION_CHANGED 또는 PEER_EVENT_MY_APPLICATION_CHANGED 이벤트가 발생할 때 반환되는 정보가 포함됩니다.
PEER_EVENT_CONNECTION_CHANGE_DATA 다음 피어 이벤트 중 하나가 트리거되는 경우 PEER_EVENT_CONNECTION_CHANGE_DATA 구조를 가리킵니다.
PEER_EVENT_ENDPOINT_CHANGED_DATA PEER_EVENT_ENDPOINT_CHANGED_DATA 구조에는 피어 협업 네트워크에 참여하는 피어에서 PEER_EVENT_ENDPOINT_CHANGED 또는 PEER_EVENT_MY_ENDPOINT_CHANGED 이벤트가 발생할 때 반환되는 정보가 포함됩니다.
PEER_EVENT_INCOMING_DATA 다음 피어 이벤트 중 하나가 트리거되는 경우 PEER_EVENT_INCOMING_DATA 구조를 가리킵니다.
PEER_EVENT_MEMBER_CHANGE_DATA PEER_EVENT_MEMBER_CHANGE_DATA 구조에는 피어 그룹 멤버의 상태 변경 사항을 설명하는 데이터가 포함되어 있습니다.
PEER_EVENT_NODE_CHANGE_DATA PEER_EVENT_NODE_CHANGE_DATA 구조체는 PEER_GRAPH_EVENT_NODE_CHANGE 이벤트가 트리거되는 경우 데이터에 대한 포인터를 포함합니다.
PEER_EVENT_OBJECT_CHANGED_DATA PEER_EVENT_OBJECT_CHANGED_DATA 구조에는 피어 협업 네트워크에 참여하는 피어에서 PEER_EVENT_ENDPOINT_OBJECT_CHANGED 또는 PEER_EVENT_MY_OBJECT_CHANGED 이벤트가 발생할 때 반환되는 정보가 포함됩니다.

PEER_EVENT_PEOPLE_NEAR_ME_CHANGED_DATA PEER_EVENT_PEOPLE_NEAR_ME_CHANGED_DATA 구조에는 서브넷별 피어 협업 네트워크에 참여하는 피어에서 PEER_EVENT_PEOPLE_NEAR_ME_CHANGED 이벤트가 발생할 때 반환되는 정보가 포함됩니다.
PEER_EVENT_PRESENCE_CHANGED_DATA PEER_EVENT_PRESENCE_CHANGED_DATA 구조에는 피어 협업 네트워크에 참여하는 피어에서 PEER_EVENT_ENDPOINT_PRESENCE_CHANGED 또는 PEER_EVENT_MY_PRESENCE_CHANGED 이벤트가 발생할 때 반환되는 정보가 포함됩니다.
PEER_EVENT_RECORD_CHANGE_DATA 다음 피어 이벤트 중 하나가 트리거되는 경우 PEER_EVENT_RECORD_CHANGE_DATA 구조를 가리킵니다.
PEER_EVENT_REQUEST_STATUS_CHANGED_DATA PEER_EVENT_REQUEST_STATUS_CHANGED_DATA 구조에는 피어 협업 네트워크에 참여하는 피어에서 PEER_EVENT_REQUEST_STATUS_CHANGED 이벤트가 발생할 때 반환되는 정보가 포함됩니다.
PEER_EVENT_SYNCHRONIZED_DATA PEER_EVENT_SYNCHRONIZED_DATA PEER_GRAPH_EVENT_RECORD_CHANGE 또는 PEER_GROUP_EVENT_RECORD_CHANGE 이벤트가 트리거되는 경우 PEER_GRAPH_EVENT_DATA 구조체의 공용 구조체에서 가리켰습니다.
PEER_EVENT_WATCHLIST_CHANGED_DATA PEER_EVENT_WATCHLIST_CHANGED_DATA 구조에는 피어 협업 네트워크에 참여하는 피어에서 PEER_EVENT_WATCHLIST_CHANGED 이벤트가 발생할 때 반환되는 정보가 포함됩니다.
PEER_GRAPH_EVENT_DATA PEER_GRAPH_EVENT_DATA 구조에는 피어 이벤트와 연결된 데이터가 포함됩니다.
PEER_GRAPH_EVENT_REGISTRATION PEER_GRAPH_EVENT_REGISTRATION 구조는 피어 이벤트 알림을 등록하는 동안 사용됩니다. 등록하는 동안 애플리케이션에 알림이 필요한 피어 이벤트를 지정합니다.
PEER_GRAPH_PROPERTIES PEER_GRAPH_PROPERTIES 구조에는 피어 그래프, ID, scope 및 기타 정보의 정책에 대한 데이터가 포함됩니다.
PEER_GROUP_EVENT_DATA

PEER_GROUP_EVENT_DATA 구조체(p2p.h)에는 발생한 특정 피어 그룹 이벤트에 대한 정보가 포함됩니다.
PEER_GROUP_EVENT_REGISTRATION PEER_GROUP_EVENT_REGISTRATION 구조체는 멤버가 등록할 수 있는 특정 피어 그룹 이벤트를 정의합니다.
PEER_GROUP_PROPERTIES PEER_GROUP_PROPERTIES 구조에는 피어 그룹의 멤버 자격 정책에 대한 데이터가 포함됩니다.
PEER_INVITATION PEER_INVITATION 구조에는 피어 협업 작업을 시작하거나 조인하는 요청이 포함됩니다.
PEER_INVITATION_INFO PEER_INVITATION_INFO 구조는 피어 그룹에 가입하기 위한 초대에 대한 정보를 정의합니다.
PEER_INVITATION_RESPONSE PEER_INVITATION_RESPONSE 구조에는 피어 공동 작업 참여 초대에 대한 응답이 포함됩니다.
PEER_MEMBER PEER_MEMBER 구조에는 피어 그룹의 멤버를 설명하는 정보가 포함되어 있습니다.
PEER_NAME_PAIR PEER_NAME_PAIR 구조체에는 PeerGetNextItem에 대한 호출 결과가 포함됩니다.
PEER_NODE_INFO PEER_NODE_INFO 구조에는 피어 그래프의 특정 노드와 관련된 정보가 포함됩니다.
PEER_OBJECT PEER_OBJECT 구조에는 피어 협업 네트워크 내에서 신뢰할 수 있는 연락처와 공유할 수 있는 애플리케이션별 런타임 정보가 포함되어 있습니다.
PEER_PEOPLE_NEAR_ME 동일한 논리 또는 가상 서브넷의 피어에 대한 정보를 포함합니다.
PEER_PNRP_CLOUD_INFO PNRP(피어 이름 확인 프로토콜) 클라우드에 대한 정보를 포함합니다.

PEER_PNRP_ENDPOINT_INFO 피어 엔드포인트와 연결된 IP 주소 및 데이터를 포함합니다.
PEER_PNRP_REGISTRATION_INFO PNRP 클라우드에 등록할 때 피어 ID에서 제공하는 정보를 포함합니다.
PEER_PRESENCE_INFO PEER_PRESENCE_INFO 구조에는 특정 피어 프레즌스 정보가 포함됩니다.
PEER_RECORD PEER_RECORD 구조에는 애플리케이션에서 사용하는 레코드 개체가 포함됩니다.
PEER_SECURITY_INTERFACE PEER_SECURITY_INTERFACE 구조는 Peer Graphing API를 호출하여 레코드의 유효성을 검사하고, 보호하고, 해제하는 데 사용하는 보안 인터페이스를 지정합니다.
PEER_VERSION_DATA PEER_VERSION_DATA 구조에는 피어 그래프 및 그룹화 API에 대한 버전 정보가 포함되어 있습니다.

열거형

PEER_APPLICATION_REGISTRATION_TYPE PEER_APPLICATION_REGISTRATION_TYPE 열거형은 피어 애플리케이션 등록 플래그 집합을 정의합니다.
PEER_CHANGE_TYPE PEER_CHANGE_TYPE 열거형은 피어 이벤트의 피어 개체, 엔드포인트 또는 애플리케이션에서 수행된 변경 내용 집합을 정의합니다. 변경 형식과 연결된 피어 이벤트를 한정하는 데 사용됩니다.
PEER_COLLAB_EVENT_TYPE PEER_COLLAB_EVENT_TYPE 열거형은 피어 협업 네트워크 이벤트 인프라를 통해 피어에서 발생할 수 있는 이벤트 집합을 정의합니다.
PEER_CONNECTION_FLAGS PEER_CONNECTION_FLAGS 열거형은 피어가 가질 수 있는 연결 유형을 지정합니다.

PEER_CONNECTION_STATUS PEER_CONNECTION_STATUS 열거형은 피어 직접 또는 인접 연결의 상태 지정합니다.
PEER_GRAPH_EVENT_TYPE PEER_GRAPH_EVENT_TYPE 열거형은 애플리케이션에 알림을 받을 피어 이벤트 유형을 지정합니다.
PEER_GRAPH_PROPERTY_FLAGS PEER_GRAPH_PROPERTY_FLAGS 열거형은 피어 그래프의 속성을 지정합니다.
PEER_GRAPH_SCOPE PEER_GRAPH_SCOPE 열거형은 피어 그래프의 네트워크 scope 지정합니다.
PEER_GRAPH_STATUS_FLAGS PEER_GRAPH_STATUS_FLAGS 열거형은 피어 그래프 내 노드의 현재 상태 표시하는 플래그 집합입니다.
PEER_GROUP_AUTHENTICATION_SCHEME 피어 그룹에 조인하는 피어를 인증하는 데 사용할 수 있는 인증 체계 집합을 정의합니다.
PEER_GROUP_EVENT_TYPE PEER_GROUP_EVENT_TYPE 열거형에는 피어 그룹 내에서 발생할 수 있는 특정 피어 이벤트 유형이 포함됩니다.
PEER_GROUP_ISSUE_CREDENTIAL_FLAGS . (PEER_GROUP_ISSUE_CREDENTIAL_FLAGS)
PEER_GROUP_PROPERTY_FLAGS PEER_GROUP_PROPERTY_FLAGS 플래그는 다양한 피어 그룹 멤버 자격 설정을 지정하는 데 사용됩니다.
PEER_GROUP_STATUS PEER_GROUP_STATUS 플래그는 피어 그룹에 연결이 있는지 여부를 나타냅니다.
PEER_INVITATION_RESPONSE_TYPE 피어 협업 작업을 시작하기 위해 초대에 수신된 응답 유형을 정의합니다.
PEER_MEMBER_CHANGE_TYPE

PEER_MEMBER_CHANGE_TYPE 열거형은 피어에 대해 가능한 피어 그룹 멤버 자격 및 현재 상태 집합을 정의합니다.
PEER_MEMBER_FLAGS PEER_MEMBER_FLAGS 플래그를 사용하면 애플리케이션이 PeerGroupEnumMembers 함수가 호출될 때 모든 멤버 또는 현재 멤버만 열거해야 하는지 여부를 지정하거나 피어 그룹 내에 멤버가 있는지 여부를 나타낼 수 있습니다.
PEER_NODE_CHANGE_TYPE PEER_NODE_CHANGE_TYPE 열거형은 변경 상태 피어 노드 그래프의 형식을 지정합니다.
PEER_PRESENCE_STATUS PEER_PRESENCE_STATUS 열거형은 피어 협업 네트워크에 참여하는 피어에서 사용할 수 있는 가능한 프레즌스 상태 설정 집합을 정의합니다.
PEER_PUBLICATION_SCOPE 피어 개체 또는 데이터 게시에 대한 범위 집합을 정의합니다.
PEER_RECORD_CHANGE_TYPE PEER_RECORD_CHANGE_TYPE 열거형은 레코드에 발생할 수 있는 변경 내용을 지정합니다.
PEER_RECORD_FLAGS PEER_RECORD_FLAGS 열거형은 피어 레코드 동작에 대한 플래그 집합을 지정합니다.
PEER_SIGNIN_FLAGS PEER_SIGNIN_FLAGS 열거형은 피어가 피어 협업 네트워크에 로그인할 때 사용할 수 있는 피어 프레즌스 게시 동작 집합을 정의합니다.
PEER_WATCH_PERMISSION 피어 연락처가 연락처에서 현재 상태 업데이트를 받을 수 있는지 여부를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

pathcch.h 헤더

아티클2024. 01. 10.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

pathcch.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

PathAllocCanonicalize 경로 문자열을 정식 형식으로 변환합니다. 이 함수는 힙의 결과를 반환한다는 점에서 PathCchCanonicalize 및 PathCchCanonicalizeEx와 다릅니다.
PathAllocCombine 두 개의 경로 조각을 단일 경로로 연결합니다.
PathCchAddBackslash 문자열 끝에 백슬래시를 추가하여 경로에 대한 올바른 구문을 만듭니다. (PathCchAddBackslash)
PathCchAddBackslashEx 문자열 끝에 백슬래시를 추가하여 경로에 대한 올바른 구문을 만듭니다. (PathCchAddBackslashEx)
PathCchAddExtension 경로 문자열에 파일 이름 확장명을 추가합니다. 이 함수는 PathAddExtension과 다릅니다. "\", "\?" 및 "\?\UNC&따옴표; 접두사.
PathCchAppend 한 경로를 다른 경로의 끝에 추가합니다. 이 함수는 길이가 MAX_PATH 최종 경로로 제한된다는 점에서 PathCchAppendEx와 다릅니다. 이 함수는 "\", "\?&따옴표로 묶은 경로를 허용한다는 점에서 PathAppend와 다릅니다. 및 "\?\UNC&따옴표; 접두사.
PathCchAppendEx 한 경로를 다른 경로의 끝에 추가합니다. 이 함수는 더 긴 최종 경로를 생성할 수 있다는 점에서

PathCchAppend와 다릅니다. 이 함수는 "\", "\?&따옴표로 묶은 경로를 허용한다는 점에서 PathAppend와 다릅니다. 및 "\?\UNC&따옴표; 접두사.
PathCchCanonicalize 경로 문자열을 정식 형식으로 변환합니다. 이 함수는 길이가 MAX_PATH 최종 경로로 제한된다는 점에서 PathCchCanonicalizeEx와 다릅니다. 이 함수는 호출자가 스택에 저장된 반환된 문자열의 크기를 선언해야 한다는 점에서 PathAllocCanonicalize와 다릅니다. 이 함수는 PathCanonicalize와 다릅니다. "\", "\?" 및 "\?\UNC&따옴표; 접두사.
PathCchCanonicalizeEx "." 및 ".."와 같은 탐색 요소를 제거하여 경로를 간소화하여 직접적이고 올바른 형식의 경로를 생성합니다. 이 함수는 더 긴 최종 경로를 생성할 수 있다는 점에서 PathCchCanonicalize와 다릅니다. 이 함수는 호출자가 스택에 저장된 반환된 문자열의 크기를 선언해야 한다는 점에서 PathAllocCanonicalize와 다릅니다. 이 함수는 PathCanonicalize와 다릅니다. "\", "\?" 및 "\?\UNC&따옴표; 접두사.
PathCchCombine 두 개의 경로 조각을 단일 경로로 결합합니다. (PathCchCombine)
PathCchCombineEx 두 개의 경로 조각을 단일 경로로 결합합니다. (PathCchCombineEx)
PathCchFindExtension 경로를 검색하여 ".exe" 또는 ".ini"과 같은 파일 이름 확장명을 찾습니다.
PathCchIsRoot 경로 문자열이 볼륨의 루트를 참조하는지 여부를 결정합니다. 이 함수는 "\", "\?" 및 "\?\UNC" 접두사를 사용하여 경로를 허용한다는 점에서 PathIsRoot와 다릅니다.
PathCchRemoveBackslash 경로 문자열의 끝에서 후행 백슬래시를 제거합니다. 이 함수는 PathRemoveBackslash와 다릅니다. "\", "\?" 및 "\?\UNC&따옴표; 접두사.
PathCchRemoveBackslashEx 경로 문자열의 끝에서 후행 백슬래시를 제거합니다. 이 함수는 문자열의 새 끝에 대한 포인터를 반환하고 버퍼에 남아 있는 사용되지 않는 문자 수를 보고할 수 있다는 점에서 PathCchRemoveBackslash와 다릅니다. 이 함수는 PathRemoveBackslash와 다릅니다. "\", "\?" 및 "\?\UNC&따옴표; 접두사.
PathCchRemoveExtension

경로에서 파일 이름 확장명을 제거합니다(있는 경우). 이 함수는 PathRemoveExtension과 다릅니다. "\", "\?" 및 "\?\UNC&따옴표; 접두사.
PathCchRemoveFileSpec 해당 요소가 파일 이름인지 디렉터리 이름인지 여부에 관계없이 경로 문자열의 마지막 요소를 제거합니다.
PathCchRenameExtension 경로 문자열의 끝에 있는 파일 이름의 확장명을 새 확장명으로 바꿉니다.
PathCchSkipRoot 드라이브 문자 또는 UNC(유니버설 명명 규칙) 서버/공유 경로 요소 다음에 있는 경로의 첫 번째 문자에 대한 포인터를 검색합니다. 이 함수는 PathSkipRoot가 "\", "\?&따옴표로 된 경로를 허용한다는 점에서 다릅니다. 및 "\?\UNC&따옴표; 접두사.
PathCchStripPrefix "\?&따옴표;를 제거합니다. 파일 경로의 접두사(있는 경우)입니다.
PathCchStripToRoot 루트 정보를 제외한 경로의 모든 파일 및 디렉터리 요소를 제거합니다. 이 함수는 PathStripToRoot와 다릅니다. "\", "\?" 및 "\?\UNC&따옴표; 접두사.
PathIsUNCEx 경로 문자열이 드라이브 문자를 기반으로 하는 경로와 달리 유효한 UNC(범용 명명 규칙) 경로인지 확인합니다. 이 함수는 경로에서 서버 이름을 추출할 수도 있다는 점에서 PathIsUNC와 다릅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

pchannel.h 헤더

아티클2023. 03. 08.

이 헤더는 원격 데스크톱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 데스크톱 서비스

pchannel.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

CHANNEL_PDU_HEADER 가상 채널의 서버 끝에서 수신되는 데이터 블록에 대한 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

pdh.h 헤더

아티클2023. 08. 25.

이 헤더는 성능 카운터에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 성능 카운터

pdh.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

PdhAddCounterA 지정된 카운터를 쿼리에 추가합니다. (ANSI)
PdhAddCounterW 지정된 카운터를 쿼리에 추가합니다. (유니코드)
PdhAddEnglishCounterA 지정된 언어 중립 카운터를 쿼리에 추가합니다. (ANSI)
PdhAddEnglishCounterW 지정된 언어 중립 카운터를 쿼리에 추가합니다. (유니코드)
PdhBindInputDataSourceA 로그 데이터를 읽기 위해 하나 이상의 이진 로그 파일을 함께 바인딩합니다. (ANSI)
PdhBindInputDataSourceW 로그 데이터를 읽기 위해 하나 이상의 이진 로그 파일을 함께 바인딩합니다. (유니코드)
PdhBrowseCountersA 사용자가 쿼리에 추가할 카운터를 하나 이상 선택하는 데 사용할 수 있는 카운터 찾아보기 대화 상자를 표시합니다. 데이터 원본에 대한 핸들을 사용하려면 PdhBrowseCountersH 함수를 사용합니다. (ANSI)
PdhBrowseCountersHA 사용자가 쿼리에 추가할 카운터를 하나 이상 선택하는 데 사용할 수 있는 카운터 찾아보기 대화 상

자를 표시합니다. 이 함수는 데이터 원본에 대한 핸들 사용을 지원한다는 점을 제외하고 PdhBrowseCounters 함수와 동일합니다. (ANSI)
PdhBrowseCountersHW 사용자가 쿼리에 추가할 카운터를 하나 이상 선택하는 데 사용할 수 있는 카운터 찾아보기 대화 상자를 표시합니다. 이 함수는 데이터 원본에 대한 핸들 사용을 지원한다는 점을 제외하고 PdhBrowseCounters 함수와 동일합니다. (유니코드)
PdhBrowseCountersW 사용자가 쿼리에 추가할 카운터를 하나 이상 선택하는 데 사용할 수 있는 카운터 찾아보기 대화 상자를 표시합니다. 데이터 원본에 대한 핸들을 사용하려면 PdhBrowseCountersH 함수를 사용합니다. (유니코드)
PdhCalculateCounterFromRawValue 두 개의 원시 카운터 값의 표시 가능한 값을 계산합니다.
PdhCloseLog 지정된 로그 파일을 닫습니다.
PdhCloseQuery 지정된 쿼리에 포함된 모든 카운터를 닫고, 쿼리와 관련된 모든 핸들을 닫고, 쿼리와 연결된 모든 메모리를 해제합니다.
PdhCollectQueryData 지정된 쿼리의 모든 카운터에 대한 현재 원시 데이터 값을 수집하고 각 카운터의 상태 코드를 업데이트합니다. (PdhCollectQueryData)
PdhCollectQueryDataEx 별도 스레드를 사용하여 지정된 쿼리의 모든 카운터에 대한 현재 원시 데이터 값을 수집합니다. 그런 다음, 함수는 애플리케이션 정의 이벤트를 알리고 반환하기 전에 지정된 시간 간격을 기다립니다.
PdhCollectQueryDataWithTime 지정된 쿼리의 모든 카운터에 대한 현재 원시 데이터 값을 수집하고 각 카운터의 상태 코드를 업데이트합니다. (PdhCollectQueryDataWithTime)
PdhComputeCounterStatistics 원시 값 배열에서 카운터에 대한 통계를 계산합니다.
PdhConnectMachineA

지정된 컴퓨터에 연결합니다. (ANSI)
PdhConnectMachineW 지정된 컴퓨터에 연결합니다. (유니코드)
PdhEnumLogSetNamesA DSN 내에서 로그 집합의 이름을 열거합니다. (ANSI)
PdhEnumLogSetNamesW DSN 내에서 로그 집합의 이름을 열거합니다. (유니코드)
PdhEnumMachinesA 로그 파일의 카운터와 연결된 컴퓨터 이름 목록을 반환합니다. (PdhEnumMachinesA)
PdhEnumMachinesHA 로그 파일의 카운터와 연결된 컴퓨터 이름 목록을 반환합니다. (PdhEnumMachinesHA)
PdhEnumMachinesHW 로그 파일의 카운터와 연결된 컴퓨터 이름 목록을 반환합니다. (PdhEnumMachinesHW)
PdhEnumMachinesW 로그 파일의 카운터와 연결된 컴퓨터 이름 목록을 반환합니다. (PdhEnumMachinesW)
PdhEnumObjectItemsA 지정된 컴퓨터 또는 지정된 로그 파일에 있는 지정된 개체의 카운터와 instance 이름을 반환합니다. 데이터 원본에 대한 핸들을 사용하려면 PdhEnumObjectItemsH 함수를 사용합니다. (ANSI)
PdhEnumObjectItemsHA 지정된 컴퓨터 또는 지정된 로그 파일에 있는 지정된 개체의 카운터와 instance 이름을 반환합니다. 이 함수는 데이터 원본에 대한 핸들 사용을 지원한다는 점을 제외하고 PdhEnumObjectItems 함수와 동일합니다. (ANSI)
PdhEnumObjectItemsHW 지정된 컴퓨터 또는 지정된 로그 파일에 있는 지정된 개체의 카운터와 instance 이름을 반환합니다. 이 함수는 데이터 원본에 대한 핸들 사용을 지원한다는 점을 제외하고 PdhEnumObjectItems 함수와 동일합니다. (유니코드)
PdhEnumObjectItemsW

지정된 컴퓨터 또는 지정된 로그 파일에 있는 지정된 개체의 카운터와 instance 이름을 반환합니다. 데이터 원본에 대한 핸들을 사용하려면 PdhEnumObjectItemsH 함수를 사용합니다. (유니코드)
PdhEnumObjectsA 지정된 컴퓨터 또는 지정된 로그 파일에서 사용할 수 있는 개체 목록을 반환합니다. 데이터 원본에 핸들을 사용하려면 PdhEnumObjectsH 함수를 사용합니다. (ANSI)
PdhEnumObjectsHA 지정된 컴퓨터 또는 지정된 로그 파일에서 사용할 수 있는 개체 목록을 반환합니다. 이 함수는 데이터 원본에 대한 핸들 사용을 지원한다는 점을 제외하고 PdhEnumObjects와 동일합니다. (ANSI)
PdhEnumObjectsHW 지정된 컴퓨터 또는 지정된 로그 파일에서 사용할 수 있는 개체 목록을 반환합니다. 이 함수는 데이터 원본에 대한 핸들 사용을 지원한다는 점을 제외하고 PdhEnumObjects와 동일합니다. (유니코드)
PdhEnumObjectsW 지정된 컴퓨터 또는 지정된 로그 파일에서 사용할 수 있는 개체 목록을 반환합니다. 데이터 원본에 핸들을 사용하려면 PdhEnumObjectsH 함수를 사용합니다. (유니코드)
PdhExpandCounterPathA 카운터 경로의 와일드카드 문자열과 일치하는 카운터 및 인스턴스에 대해 지정된 컴퓨터(또는 지정된 컴퓨터가 없는 경우 로컬 컴퓨터)를 검사합니다. (ANSI)
PdhExpandCounterPathW 카운터 경로의 와일드카드 문자열과 일치하는 카운터 및 인스턴스에 대해 지정된 컴퓨터(또는 지정된 컴퓨터가 없는 경우 로컬 컴퓨터)를 검사합니다. (유니코드)
PdhExpandWildCardPathA 지정된 컴퓨터 또는 로그 파일을 검사하고 와일드카드 문자가 포함된 지정된 카운터 경로와 일치하는 카운터 경로를 반환합니다. 데이터 원본에 핸들을 사용하려면 PdhExpandWildCardPathH 함수를 사용합니다. (ANSI)
PdhExpandWildCardPathHA 지정된 컴퓨터 또는 로그 파일을 검사하고 와일드카드 문자가 포함된 지정된 카운터 경로와 일치하는 카운터 경로를 반환합니다. 이 함수는 데이터 원본에 대한 핸들 사용을 지원한다는 점을 제외하고 PdhExpandWildCardPath 함수와 동일합니다. (ANSI)
PdhExpandWildCardPathHW 지정된 컴퓨터 또는 로그 파일을 검사하고 와일드카드 문자가 포함된 지정된 카운터 경로와 일치

하는 카운터 경로를 반환합니다. 이 함수는 데이터 원본에 대한 핸들 사용을 지원한다는 점을 제외하고 PdhExpandWildCardPath 함수와 동일합니다. (유니코드)
PdhExpandWildCardPathW 지정된 컴퓨터 또는 로그 파일을 검사하고 와일드카드 문자가 포함된 지정된 카운터 경로와 일치하는 카운터 경로를 반환합니다. 데이터 원본에 핸들을 사용하려면 PdhExpandWildCardPathH 함수를 사용합니다. (유니코드)
PdhFormatFromRawValue 지정된 원시 카운터 값에 대해 표시 가능한 값을 계산합니다.
PdhGetCounterInfoA 데이터 크기, 카운터 유형, 경로 및 사용자가 제공한 데이터 값과 같은 카운터에 대한 정보를 검색합니다. (ANSI)
PdhGetCounterInfoW 데이터 크기, 카운터 유형, 경로 및 사용자가 제공한 데이터 값과 같은 카운터에 대한 정보를 검색합니다. (유니코드)
PdhGetCounterTimeBase 지정된 카운터의 시간 기반을 반환합니다.
PdhGetDataSourceTimeRangeA 시간 범위, 항목 수 및 해당하는 경우 지정된 입력 원본의 성능 데이터를 포함하는 버퍼의 크기를 결정합니다. 데이터 원본에 대한 핸들을 사용하려면 PdhGetDataSourceTimeRangeH 함수를 사용합니다. (ANSI)
PdhGetDataSourceTimeRangeH 시간 범위, 항목 수 및 해당하는 경우 지정된 입력 원본의 성능 데이터를 포함하는 버퍼의 크기를 결정합니다. 이 함수는 데이터 원본에 대한 핸들 사용을 지원한다는 점을 제외하고 PdhGetDataSourceTimeRange 함수와 동일합니다.
PdhGetDataSourceTimeRangeW 시간 범위, 항목 수 및 해당하는 경우 지정된 입력 원본의 성능 데이터를 포함하는 버퍼의 크기를 결정합니다. 데이터 원본에 대한 핸들을 사용하려면 PdhGetDataSourceTimeRangeH 함수를 사용합니다. (유니코드)
PdhGetDefaultPerfCounterA 지정된 개체의 기본 카운터 이름을 검색합니다. 이 이름은 카운터 찾아보기 대화 상자에서 초기 카

운터 선택을 설정하는 데 사용할 수 있습니다. 데이터 원본에 핸들을 사용하려면 PdhGetDefaultPerfCounterH 함수를 사용합니다. (ANSI)
PdhGetDefaultPerfCounterHA 지정된 개체의 기본 카운터 이름을 검색합니다. (ANSI)
PdhGetDefaultPerfCounterHW 지정된 개체의 기본 카운터 이름을 검색합니다. (유니코드)
PdhGetDefaultPerfCounterW 지정된 개체의 기본 카운터 이름을 검색합니다. 이 이름은 카운터 찾아보기 대화 상자에서 초기 카운터 선택을 설정하는 데 사용할 수 있습니다. 데이터 원본에 핸들을 사용하려면 PdhGetDefaultPerfCounterH 함수를 사용합니다. (유니코드)
PdhGetDefaultPerfObjectA 기본 개체의 이름을 검색합니다. 이 이름은 카운터 찾아보기 대화 상자에서 초기 개체 선택을 설정하는 데 사용할 수 있습니다. 데이터 원본에 핸들을 사용하려면 PdhGetDefaultPerfObjectH 함수를 사용합니다. (ANSI)
PdhGetDefaultPerfObjectHA 기본 개체의 이름을 검색합니다. (ANSI)
PdhGetDefaultPerfObjectHW 기본 개체의 이름을 검색합니다. (유니코드)
PdhGetDefaultPerfObjectW 기본 개체의 이름을 검색합니다. 이 이름은 카운터 찾아보기 대화 상자에서 초기 개체 선택을 설정하는 데 사용할 수 있습니다. 데이터 원본에 핸들을 사용하려면 PdhGetDefaultPerfObjectH 함수를 사용합니다. (유니코드)
PdhGetDllVersion 현재 설치된 Pdh.dll 파일의 버전을 반환합니다.
PdhGetFormattedCounterArrayA 서식이 지정된 카운터 값의 배열을 반환합니다. instance 이름의 와일드카드 문자가 포함된 카운터의 카운터 값에 서식을 지정하려면 이 함수를 사용합니다. (ANSI)
PdhGetFormattedCounterArrayW

서식이 지정된 카운터 값의 배열을 반환합니다. instance 이름의 와일드카드 문자가 포함된 카운터의 카운터 값에 서식을 지정하려면 이 함수를 사용합니다. (유니코드)
PdhGetFormattedCounterValue 지정된 카운터의 표시 가능한 값을 계산합니다.
PdhGetLogFileSize 지정된 로그 파일의 크기를 반환합니다.
PdhGetRawCounterArrayA 지정된 카운터에서 원시 값 배열을 반환합니다. instance 이름의 와일드카드 문자가 포함된 카운터의 원시 카운터 값을 검색하려는 경우 이 함수를 사용합니다. (ANSI)
PdhGetRawCounterArrayW 지정된 카운터에서 원시 값 배열을 반환합니다. instance 이름의 와일드카드 문자가 포함된 카운터의 원시 카운터 값을 검색하려는 경우 이 함수를 사용합니다. (유니코드)
PdhGetRawCounterValue 카운터의 현재 원시 값을 반환합니다.
PdhIsRealTimeQuery 지정된 쿼리가 실시간 쿼리인지 확인합니다.
PdhLookupPerfIndexByNameA 지정된 카운터 이름에 해당하는 카운터 인덱스 를 반환합니다. (ANSI)
PdhLookupPerfIndexByNameW 지정된 카운터 이름에 해당하는 카운터 인덱스 를 반환합니다. (유니코드)
PdhLookupPerfNameByIndexA 지정된 인덱스 에 해당하는 성능 개체 이름 또는 카운터 이름을 반환합니다. (ANSI)
PdhLookupPerfNameByIndexW 지정된 인덱스 에 해당하는 성능 개체 이름 또는 카운터 이름을 반환합니다. (유니코드)
PdhMakeCounterPathA PDH_COUNTER_PATH_ELEMENTS 구조체에 지정된 멤버를 사용하여 전체 카운터 경로를 만듭니다. (ANSI)

PdhMakeCounterPathW PDH_COUNTER_PATH_ELEMENTS 구조체에 지정된 멤버를 사용하여 전체 카운터 경로를 만듭니다. (유니코드)
PdhOpenLogA 읽기 또는 쓰기를 위해 지정된 로그 파일을 엽니다. (ANSI)
PdhOpenLogW 읽기 또는 쓰기를 위해 지정된 로그 파일을 엽니다. (유니코드)
PdhOpenQueryA 성능 데이터 컬렉션을 관리하는 데 사용되는 새 쿼리를 만듭니다. 데이터 원본에 핸들을 사용하려면 PdhOpenQueryH 함수를 사용합니다. (ANSI)
PdhOpenQueryH 성능 데이터 컬렉션을 관리하는 데 사용되는 새 쿼리를 만듭니다. 이 함수는 데이터 원본에 대한 핸들 사용을 지원한다는 점을 제외하고 PdhOpenQuery 함수와 동일합니다.
PdhOpenQueryW 성능 데이터 컬렉션을 관리하는 데 사용되는 새 쿼리를 만듭니다. 데이터 원본에 핸들을 사용하려면 PdhOpenQueryH 함수를 사용합니다. (유니코드)
PdhParseCounterPathA 카운터 경로의 요소를 구문 분석하고 결과를 PDH_COUNTER_PATH_ELEMENTS 구조에 저장합니다. (ANSI)
PdhParseCounterPathW 카운터 경로의 요소를 구문 분석하고 결과를 PDH_COUNTER_PATH_ELEMENTS 구조에 저장합니다. (유니코드)
PdhParseInstanceNameA instance 문자열의 요소를 구문 분석합니다. (ANSI)
PdhParseInstanceNameW instance 문자열의 요소를 구문 분석합니다. (유니코드)
PdhReadRawLogRecord 지정된 이진 추적 로그 파일의 정보를 읽습니다.

PdhRemoveCounter 쿼리에서 카운터를 제거합니다.
PdhSelectDataSourceA 사용자에게 성능 데이터의 원본을 지정하라는 메시지를 표시하는 대화 상자 창을 표시합니다. (ANSI)
PdhSelectDataSourceW 사용자에게 성능 데이터의 원본을 지정하라는 메시지를 표시하는 대화 상자 창을 표시합니다. (유니코드)
PdhSetCounterScaleFactor 서식이 지정된 카운터 값을 요청할 때 지정된 카운터의 계산 값에 적용되는 배율 인수를 설정합니다. PDH_FMT_NOSCALE 플래그가 설정되면 이 배율 인수는 무시됩니다.
PdhSetDefaultRealTimeDataSource 실시간 데이터의 원본을 지정합니다.
PdhSetQueryTimeRange 로그 파일에서 읽을 수 있는 샘플을 지정된 시간 범위 내의 샘플로 제한합니다.
PdhUpdateLogA 현재 쿼리에 대한 카운터 데이터를 수집하고 데이터를 로그 파일에 씁니다. (ANSI)
PdhUpdateLogFileCatalog 로그 파일 카탈로그의 정보를 로그 파일의 성능 데이터와 동기화합니다.
PdhUpdateLogW 현재 쿼리에 대한 카운터 데이터를 수집하고 데이터를 로그 파일에 씁니다. (유니코드)
PdhValidatePathA 카운터 경로에 지정된 컴퓨터에 카운터가 있는지 확인합니다. (ANSI)
PdhValidatePathExA 지정된 카운터가 컴퓨터 또는 로그 파일에 있는지 확인합니다. (ANSI)
PdhValidatePathExW

지정된 카운터가 컴퓨터 또는 로그 파일에 있는지 확인합니다. (유니코드)
PdhValidatePathW 카운터 경로에 지정된 컴퓨터에 카운터가 있는지 확인합니다. (유니코드)

콜백 함수

CounterPathCallBack 애플리케이션은 CounterPathCallBack 함수를 구현하여 찾아보기 대화 상자에서 반환된 카운터 경로 문자열을 처리합니다.

구조체

PDH_BROWSE_DLG_CONFIG_A PDH_BROWSE_DLG_CONFIG 구조는 PdhBrowseCounters 함수에서 성능 카운터 찾아보기 대화 상자를 구성하는 데 사용됩니다. (ANSI)
PDH_BROWSE_DLG_CONFIG_HA PDH_BROWSE_DLG_CONFIG_H 구조는 PdhBrowseCountersH 함수에서 성능 카운터 찾아보기 대화 상자를 구성하는 데 사용됩니다. (ANSI)
PDH_BROWSE_DLG_CONFIG_HW PDH_BROWSE_DLG_CONFIG_H 구조는 PdhBrowseCountersH 함수에서 성능 카운터 찾아보기 대화 상자를 구성하는 데 사용됩니다. (유니코드)
PDH_BROWSE_DLG_CONFIG_W PDH_BROWSE_DLG_CONFIG 구조는 PdhBrowseCounters 함수에서 성능 카운터 찾아보기 대화 상자를 구성하는 데 사용됩니다. (유니코드)
PDH_COUNTER_INFO_A PDH_COUNTER_INFO 구조에는 카운터의 속성을 설명하는 정보가 포함되어 있습니다. 이 정보에는 카운터 경로도 포함됩니다. (ANSI)

PDH_COUNTER_INFO_W PDH_COUNTER_INFO 구조에는 카운터의 속성을 설명하는 정보가 포함되어 있습니다. 이 정보에는 카운터 경로도 포함됩니다. (유니코드)
PDH_COUNTER_PATH_ELEMENTS_A PDH_COUNTER_PATH_ELEMENTS 구조체에는 카운터 경로의 구성 요소가 포함됩니다. (ANSI)
PDH_COUNTER_PATH_ELEMENTS_W PDH_COUNTER_PATH_ELEMENTS 구조체에는 카운터 경로의 구성 요소가 포함됩니다. (유니코드)
PDH_DATA_ITEM_PATH_ELEMENTS_A PDH_DATA_ITEM_PATH_ELEMENTS 구조에는 특정 데이터 항목의 경로 요소가 포함됩니다. (ANSI)
PDH_DATA_ITEM_PATH_ELEMENTS_W PDH_DATA_ITEM_PATH_ELEMENTS 구조에는 특정 데이터 항목의 경로 요소가 포함됩니다. (유니코드)
PDH_FMT_COUNTERVALUE PDH_FMT_COUNTERVALUE 구조에는 카운터의 계산된 값과 해당 상태 포함됩니다.
PDH_FMT_COUNTERVALUE_ITEM_A PDH_FMT_COUNTERVALUE_ITEM 구조체에는 카운터의 instance 이름과 서식이 지정된 값이 포함됩니다. (ANSI)
PDH_FMT_COUNTERVALUE_ITEM_W PDH_FMT_COUNTERVALUE_ITEM 구조체에는 카운터의 instance 이름과 서식이 지정된 값이 포함됩니다. (유니코드)
PDH_RAW_COUNTER PDH_RAW_COUNTER 구조체는 카운터 공급자로부터 수집된 데이터를 반환합니다. 데이터에 대해 번역, 서식 또는 기타 해석이 수행되지 않습니다.
PDH_RAW_COUNTER_ITEM_A PDH_RAW_COUNTER_ITEM 구조체에는 카운터의 instance 이름과 원시 값이 포함됩니다. (ANSI)
PDH_RAW_COUNTER_ITEM_W PDH_RAW_COUNTER_ITEM 구조체에는 카운터의 instance 이름과 원시 값이 포함됩니다. (유니코드)

PDH_RAW_LOG_RECORD PDH_RAW_LOG_RECORD 구조에는 이진 추적 로그 파일 레코드에 대한 정보가 포함됩니다.
PDH_STATISTICS PDH_STATISTICS 구조체에는 원시 카운터 값 배열에 대한 최소값, 최대값 및 평균 값이 포함됩니다.
PDH_TIME_INFO PDH_TIME_INFO 구조에는 성능 데이터 샘플링에 적용되는 시간 간격에 대한 정보가 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

peerdist.h 헤더

아티클2023. 08. 25.

이 헤더는 피어 투 피어에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 피어 투 피어

peerdist.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

PeerDistClientAddContentInformation PeerDistClientAddContentInformation 함수는 PeerDistClientOpenContent에서 연 콘텐츠 핸들과 연결된 콘텐츠 정보를 추가합니다.
PeerDistClientAddData PeerDistClientAddData 함수는 로컬 캐시에 콘텐츠를 제공하는 데 사용됩니다.
PeerDistClientBlockRead PeerDistClientBlockRead 함수는 콘텐츠 데이터 블록을 읽습니다.
PeerDistClientCancelAsyncOperation PeerDistClientCancelAsyncOperation 함수는 OVERLAPPED 구조체와 연결된 비동기 작업 및 PeerDistClientOpenContent에서 반환된 콘텐츠 핸들을 취소합니다.
PeerDistClientCloseContent PeerDistClientCloseContent 함수는 PeerDistClientOpenContent에서 연 콘텐츠 핸들을 닫습니다.
PeerDistClientCompleteContentInformation PeerDistClientCompleteContentInformation 함수는 콘텐츠 정보를 추가하는 프로세스를 완료합니다.
PeerDistClientFlushContent PeerDistClientFlushContent 함수를 사용하면 클라이언트가 연결된 PEERDIST_CONTENT_TAG 사용하여 PeerDistClientAddData 함수를 사용하여 로컬 캐시에 추가된 콘텐츠를 제거할 수 있습니다.
PeerDistClientGetInformationByHandle

PeerDistClientGetInformationByHandle 함수는 특정 콘텐츠 핸들에 대한 피어 배포 서비스에서 추가 정보를 검색합니다.
PeerDistClientOpenContent PeerDistClientOpenContent 함수가 열리고 PEERDIST_CONTENT_HANDLE 반환합니다. 클라이언트는 콘텐츠 핸들을 사용하여 피어 배포 서비스에서 데이터를 검색합니다.
PeerDistClientStreamRead PeerDistClientStreamRead는 콘텐츠 스트림에서 바이트 시퀀스를 읽습니다.
PeerDistGetOverlappedResult PeerDistGetOverlappedResult 함수는 비동기 작업의 결과를 검색합니다.
PeerDistGetStatus PeerDistGetStatus 함수는 피어 배포 서비스의 현재 상태 반환합니다.
PeerDistGetStatusEx PeerDistGetStatusEx 함수는 피어 배포 서비스의 현재 상태 및 기능을 반환합니다.
PeerDistRegisterForStatusChangeNotification PeerDistRegisterForStatusChangeNotification 함수는 피어 배포 서비스 상태 변경 알림을 요청합니다.
PeerDistRegisterForStatusChangeNotificationEx PeerDistRegisterForStatusChangeNotificationEx 함수는 피어 배포 서비스 상태 변경 알림을 요청합니다.
PeerDistServerCancelAsyncOperation PeerDistServerCancelAsyncOperation 함수는 콘텐츠 식별자 및 OVERLAPPED 구조체와 연결된 비동기 작업을 취소합니다.
PeerDistServerCloseContentInformation PeerDistServerCloseContentInformation 함수는 PeerDistServerOpenContentInformation에서 연 핸들을 닫습니다.
PeerDistServerCloseStreamHandle PeerDistServerCloseStreamHandle 함수는 PeerDistServerPublishStream에서 반환된 핸들을 닫습니다.

PeerDistServerOpenContentInformation PeerDistServerOpenContentInformation 함수는 PEERDIST_CONTENTINFO_HANDLE 엽니다. 클라이언트는 핸들을 사용하여 콘텐츠 정보를 검색합니다.
PeerDistServerOpenContentInformationEx PeerDistServerOpenContentInformationEx 함수는 PEERDIST_CONTENTINFO_HANDLE 엽니다. 클라이언트는 핸들을 사용하여 콘텐츠 정보를 검색합니다.
PeerDistServerPublishAddToStream PeerDistServerPublishAddToStream 함수는 게시 스트림에 데이터를 추가합니다.
PeerDistServerPublishCompleteStream PeerDistServerPublishCompleteStream 함수는 스트림에 데이터를 추가하는 프로세스를 완료합니다.
PeerDistServerPublishStream PeerDistServerPublishStream 함수는 피어 배포 서비스에 게시할 새 스트림을 초기화합니다.
PeerDistServerRetrieveContentInformation PeerDistServerRetrieveContentInformation 함수는 PeerDistServerOpenContentInformation에서 반환된 핸들과 연결된 인코딩된 콘텐츠 정보를 검색합니다.
PeerDistServerUnpublish PeerDistServerUnpublish 함수는 PeerDistServerPublishStream을 통해 만든 게시를 제거합니다.
PeerDistShutdown PeerDistShutdown 함수는 PeerDistStartup 호출에 의해 할당된 리소스를 해제합니다. PeerDistStartup 호출에서 반환된 각 핸들은 PeerDistShutdown에 대한 일치 호출로 닫혀야 합니다.
PeerDistStartup PeerDistStartup 함수는 다른 모든 피어 배포 API에 전달되어야 하는 새 피어 배포 instance 핸들을 만듭니다.
PeerDistUnregisterForStatusChangeNotification PeerDistUnregisterForStatusChangeNotification 함수는 지정된 핸들과 연결된 세션에 대한 상태 변경 알림을 등록 취소합니다.

구조체

PEERDIST_CLIENT_BASIC_INFO PEERDIST_CLIENT_BASIC_INFO 구조는 동일한 콘텐츠를 동시에 다운로드하는 클라이언트가 많은지 여부를 나타냅니다.
PEERDIST_CONTENT_TAG PEERDIST_CONTENT_TAG 구조에는 PeerDistClientOpenContent API에 대한 입력으로 클라이언트 제공 콘텐츠 태그가 포함됩니다.
PEERDIST_PUBLICATION_OPTIONS PEERDIST_PUBLICATION_OPTIONS 구조에는 API 버전 정보 및 가능한 옵션 플래그를 포함한 게시 옵션이 포함됩니다.
PEERDIST_RETRIEVAL_OPTIONS PEER_RETRIEVAL_OPTIONS 구조에는 검색할 콘텐츠 정보의 버전이 포함되어 있습니다.
PEERDIST_STATUS_INFO PEERDIST_STATUS_INFO 구조에는 로컬 컴퓨터에서 BranchCache 서비스의 현재 상태 및 기능에 대한 정보가 포함되어 있습니다.

열거형

PEERDIST_CLIENT_INFO_BY_HANDLE_CLASS PEERDIST_CLIENT_INFO_BY_HANDLE_CLASS 열거형은 가능한 클라이언트 정보 값을 정의합니다.
PEERDIST_STATUS PEERDIST_STATUS 열거형은 피어 배포 서비스의 가능한 상태 값을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

peninputpanel.h 헤더

아티클2023. 03. 08.

이 헤더는 태블릿 PC에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Tablet PC

peninputpanel.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IHandwrittenTextInsertion 애플리케이션의 사용자 지정 텍스트 입력 코드에서 텍스트 필드와 Text Services 백업 저장소 모두에 텍스트를 삽입하는 데 사용됩니다.
IPenInputPanel . (IPenInputPanel)
ITextInputPanel 태블릿 PC 입력 패널의 모양과 동작을 제어합니다.
ITextInputPanelEventSink ITextInputPanel 인터페이스 이벤트를 처리하는 메서드를 정의합니다.
ITextInputPanelRunInfo 텍스트 입력 패널이 현재 실행 중인지 확인하는 메서드를 제공합니다.

열거형

CorrectionMode 태블릿 PC 입력 패널의 수정 모드를 지정합니다.
수정 처리 삽입 후 수정 빗이 표시하는 입력 패널을 기준으로 방향을 지정합니다.

EventMask 주의를 끌 수 있는 ITextInputPanel 인터페이스의 이벤트입니다.
InPlaceDirection 텍스트 입력 필드를 기준으로 In-Place 입력 패널의 기본 방향을 지정합니다.
InPlaceState 태블릿 PC 입력 패널의 In-Place 상태 값을 지정합니다.
InteractionMode 태블릿 PC 입력 패널에 대해 사용자가 선택할 수 있는 상호 작용 모드를 지정합니다.
PanelInputArea 태블릿 PC 입력 패널의 기본 입력 영역을 나타내는 값을 지정합니다.
PanelType PenInputPanel 개체에서 현재 사용할 수 있는 입력 유형을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

perflib.h 헤더

아티클2023. 08. 25.

이 헤더는 성능 카운터에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 성능 카운터

perflib.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

PerfAddCounters 지정된 쿼리에 성능 카운터 사양을 추가합니다.
PerfCloseQueryHandle PerfOpenQueryHandle을 호출하여 연 쿼리 핸들을 닫습니다.
PerfCreateInstance 지정된 카운터 집합의 instance 만듭니다.
PerfDecrementULongCounterValue 값이 4 바이트 부호 없는 정수인 카운터의 값을 감소합니다. 공급자는 이 함수를 사용합니다.
PerfDecrementULongLongCounterValue 값이 부호 없는 8 바이트 정수인 카운터의 값을 감소합니다. 공급자는 이 함수를 사용합니다.
PerfDeleteCounters 지정된 쿼리에서 지정된 성능 카운터 사양을 제거합니다.
PerfDeleteInstance PerfCreateInstance 함수에서 만든 카운터 집합의 instance 삭제합니다.
PerfEnumerateCounterSet 지정된 시스템에 등록된 카운터 집합의 카운터 집합 식별자를 가져옵니다. 카운터 집합 식별자는 GUID(Globally Unique Identifiers)입니다.
PerfEnumerateCounterSetInstances

지정된 시스템에 설정된 카운터의 활성 인스턴스 이름과 식별자를 가져옵니다.
PerfIncrementULongCounterValue 값이 4 바이트 부호 없는 정수인 카운터의 값을 증분합니다. 공급자는 이 함수를 사용합니다.
PerfIncrementULongLongCounterValue 값이 부호 없는 8 바이트 정수인 카운터의 값을 증분합니다. 공급자는 이 함수를 사용합니다.
PerfOpenQueryHandle 지정된 시스템에서 쿼리를 참조하는 핸들을 만듭니다. 쿼리는 카운터 사양의 목록입니다.
PerfQueryCounterData 지정된 쿼리의 카운터 사양과 일치하는 성능 카운터의 값을 가져옵니다.
PerfQueryCounterInfo 지정된 쿼리의 카운터 사양을 가져옵니다.
PerfQueryCounterSetRegistrationInfo 지정된 시스템의 카운터 집합에 대한 정보를 가져옵니다.
PerfQueryInstance 지정된 카운터 집합 instance 대한 포인터를 검색합니다. 공급자는 이 함수를 사용합니다.
PerfSetCounterRefValue 값이 실제 데이터에 대한 포인터인 카운터의 값을 업데이트. 공급자는 이 함수를 사용합니다.
PerfSetCounterSetInfo 특정 카운터 집합의 레이아웃을 지정합니다.
PerfSetULongCounterValue 값이 4 바이트 부호 없는 정수인 카운터의 값을 업데이트. 공급자는 이 함수를 사용합니다.
PerfSetULongLongCounterValue 값이 8 바이트 부호 없는 정수인 카운터의 값을 업데이트. 공급자는 이 함수를 사용합니다.
PerfStartProvider 공급자를 등록합니다. (PerfStartProvider)

PerfStartProviderEx 공급자를 등록합니다. (PerfStartProviderEx)
PerfStopProvider 등록된 공급자 목록에서 공급자의 등록을 제거하고 공급자와 연결된 모든 리소스를 해제합니다.

콜백 함수

PERF_MEM_ALLOC 공급자는 PERFLIB에 대한 사용자 지정 메모리 관리를 제공하기 위해 이 함수를 구현합니다. A
PERF_MEM_FREE 공급자는 PERFLIB에 대한 사용자 지정 메모리 관리를 제공하기 위해 이 함수를 구현합니다. F
PERFLIBREQUEST 공급자는 소비자가 쿼리에서 카운터 추가 또는 제거와 같은 특정 작업을 수행할 때 알림을 수신하도록 이 함수를 구현할 수 있습니다.

구조체

PERF_COUNTER_DATA 구조를 포함하는 PERF_COUNTER_DATA 블록에 대한 정보를 포함합니다.
PERF_COUNTER_HEADER 구조를 포함하는 PERF_COUNTER_HEADER 블록에 대한 정보를 포함합니다.
PERF_COUNTER_IDENTIFIER 구조를 포함하는 PERF_COUNTER_IDENTIFIER 블록에 대한 정보를 포함합니다.
PERF_COUNTER_IDENTITY

소비자가 쿼리에서 카운터를 추가하거나 제거할 때 공급자의 콜백으로 전송되는 카운터를 정의합니다.
PERF_COUNTER_INFO 공급자가 사용하는 카운터에 대한 정보를 정의합니다. CTRPP 도구는 지정한 스키마에 따라 이 구조를 자동으로 생성합니다.
PERF_COUNTER_REG_INFO 성능 카운터에 대한 등록 정보를 제공합니다.
PERF_COUNTERSET_INFO 공급자가 사용하는 카운터 집합에 대한 정보를 정의합니다. CTRPP 도구는 지정한 스키마에 따라 이 구조를 자동으로 생성합니다.
PERF_COUNTERSET_INSTANCE 카운터 집합의 instance 정의합니다.
PERF_COUNTERSET_REG_INFO 구조를 포함하는 PERF_COUNTERSET_REG_INFO 블록에 대한 정보를 포함합니다.
PERF_DATA_HEADER 구조를 포함하는 PERF_DATA_HEADER 블록에 대한 정보를 제공합니다.
PERF_INSTANCE_HEADER 구조를 포함하는 PERF_INSTANCE_HEADER 블록에 대한 정보를 제공합니다.
PERF_MULTI_COUNTERS 구조를 포함하는 PERF_MULTI_COUNTERS 블록에 대한 정보를 제공합니다.
PERF_MULTI_INSTANCES 구조를 포함하는 PERF_MULTI_INSTANCES 블록에 대한 정보를 제공합니다.
PERF_PROVIDER_CONTEXT 공급자 컨텍스트 정보를 정의합니다.
PERF_STRING_BUFFER_HEADER 구조를 포함하는 PERF_STRING_BUFFER_HEADER 블록에 대한 정보를 제공합니다.

PERF_STRING_COUNTER_HEADER PERF_STRING_BUFFER_HEADER 블록에서 표시된 성능 카운터의 이름 또는 도움말 문자열이 포함된 문자열이 시작되는 위치를 나타냅니다.

열거형

PerfCounterDataType PerfQueryCounterData 함수가 함수가 출력으로 생성하는 PERF_DATA_HEADER 블록의 일부로 포함하는 PERF_COUNTER_HEADER 블록의 콘텐츠 형식을 나타냅니다.
PerfRegInfoType PerfQueryCounterSetRegistrationInfo 함수를 호출하여 성능 카운터 집합에 대해 요청할 수 있는 정보 유형을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

photoacquire.h 헤더

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 애플리케이션 설치 및 서비스
- 사진 취득

photoacquire.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Interfaces

테이블 확장

IPhotoAcquire IPhotoAcquire 인터페이스는 디바이스에서 사진을 가져오는 방법을 제공합니다.
IPhotoAcquireDeviceSelectionDialog 이미지를 가져올 디바이스를 선택하는 대화 상자를 제공합니다.
IPhotoAcquireItem IPhotoAcquireItem 인터페이스는 디바이스에서 획득할 때 항목 작업을 위한 메서드를 제공합니다.
IPhotoAcquireOptionsDialog IPhotoAcquireOptionsDialog 인터페이스는 사용자가 파일 이름 형식과 같은 사진 취득 설정뿐만 아니라 이미지를 회전할지, 태그 이름을 묻는 메시지를 표시할지 또는 가져온 후 카메라에서 사진을 지울지 여부를 선택할 수 있는 옵션 대화 상자를 표시하는 데 사용됩니다.
IPhotoAcquirePlugin 이미지 취득을 위해 Windows Vista UI(사용자 인터페이스)와 함께 실행할 플러그 인을 만들려는 경우 IPhotoAcquirePlugin 인터페이스를 구현합니다. 플러그 인을 사용하도록 설정하려면 레지스트리 설정이 필요합니다.
IPhotoAcquireProgressCB 획득 프로세스의 다양한 단계에서 추가 처리를 수행하려는 경우 IPhotoAcquireProgressCB 인터페이스를 구현할 수 있습니다.
IPhotoAcquireSettings IPhotoAcquireSettings 인터페이스는 파일 이름 형식과 같은 이미지 획득 설정을 사용하는 데 사용됩니다.

IPhotoAcquireSource IPhotoAcquireSource 인터페이스는 디바이스에서 항목을 획득하는 데 사용됩니다.
IPhotoProgressDialog 이미지를 열거하거나 가져올 때 표시될 수 있는 진행률 대화 상자를 제공합니다. 대화 상자는 모달이며 자체 스레드에서 실행됩니다.
IUserInputString IUserInputString 인터페이스는 사용자에게 문자열을 요청할 때 생성된 개체(예: 태그 이름을 가져올 때)를 나타냅니다.

열거 목록

테이블 확장

DEVICE_SELECTION_DEVICE_TYPE DEVICE_SELECTION_DEVICE_TYPE 열거형 유형은 선택한 디바이스의 유형을 나타냅니다.
ERROR_ADVISE_MESSAGE_TYPE ERROR_ADVISE_MESSAGE_TYPE 열거형 형식은 IPhotoAcquireProgressCB::ErrorAdvise의 nMessageType 매개 변수에 전달할 수 있는 오류 값의 형식을 나타냅니다.
ERROR_ADVISE_RESULT ERROR_ADVISE_RESULT 열거형 형식은 IPhotoAcquireProgressCB::ErrorAdvise의 pnErrorAdviseResult 매개 변수에 할당할 수 있는 오류 값의 형식을 나타냅니다.
PROGRESS_DIALOG_CHECKBOX_ID PROGRESS_DIALOG_CHECKBOX_ID 열거형 형식은 IPhotoProgressDialog 개체의 확인란을 나타냅니다.
PROGRESS_DIALOG_IMAGE_TYPE PROGRESS_DIALOG_IMAGE_TYPE 열거형 형식은 IPhotoProgressDialog::SetImage에 설정된 이미지 형식을 나타냅니다.
USER_INPUT_STRING_TYPE

USER_INPUT_STRING_TYPE 열거형 형식은 IPhotoAcquireProgressCB::GetUserInput의 사용자로부터 가져올 문자열 형식을 나타냅니다.

Last updated on 2025. 10. 08.

physicalmonitorenumerationapi.h 헤더

아티클2023. 08. 25.

이 헤더는 모니터 구성에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 구성 모니터링

physicalmonitorenumerationapi.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DestroyPhysicalMonitor 실제 모니터에 대한 핸들을 닫습니다.
DestroyPhysicalMonitors 물리적 모니터 핸들의 배열을 닫습니다.
GetNumberOfPhysicalMonitorsFromHMONITOR HMONITOR 모니터 핸들과 연결된 실제 모니터 수를 검색합니다.
GetNumberOfPhysicalMonitorsFromIDirect3DDevice9 Direct3D 디바이스와 연결된 실제 모니터 수를 검색합니다.
GetPhysicalMonitorsFromHMONITOR HMONITOR 모니터 핸들과 연결된 물리적 모니터를 검색합니다.
GetPhysicalMonitorsFromIDirect3DDevice9 Direct3D 디바이스와 연결된 물리적 모니터를 검색합니다.

구조체

PHYSICAL_MONITOR 실제 모니터에 해당하는 핸들 및 텍스트 설명을 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

pla.h 헤더

아티클2023. 03. 08.

이 헤더는 성능 로그 및 경고에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 성능 로그 및 경고

pla.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAlertDataCollector 카운터 값이 지정된 임계값을 초과할 때마다 성능 카운터를 모니터링하고 작업을 수행합니다. 경고 데이터 수집기를 만들려면 IDataCollectorCollection::CreateDataColl IDataCollectorCollection::CreateDataCollectorFromXml 메서드를 호출합니다. CreateDataCollectorFromXml에 전달하는 XML에 대한 자세한 내용은 비고를 참조하세요.
IApiTracingDataCollector Win32 호출을 Kernel32.dll, Advapi32.dll, Gdi32.dll 및 User32.dll 기록합니다.
IConfigurationDataCollector 컬렉션 시 컴퓨터 설정을 수집합니다.
IDataCollector XML을 사용하여 수집기 속성을 설정하고 검색하고, 로그 파일 이름을 지정하고, 로그 파일의 위치를 검색합니다. 이 인터페이스는 다음 데이터 수집기가 파생되는 추상 IAlertDataCollectorIApiTracingDataCollectorIConfigurationDataCollectorIPerformanceCounterDataCollectorITraceDataCollector
IDataCollectorCollection DataCollector 개체의 컬렉션을 관리합니다. 이 인터페이스를 얻으려면 IDataCollectorSet::D ataCollectors 속성에 액세스합니다.
IDataCollectorSet 집합의 모든 데이터 수집기 개체에 공통적인 구성 정보를 관리합니다. 집합에서 데이터 수집기를 추가하고 제거합니다. 데이터 수집을 시작합니다. 이 인터페이스는 사 터페이스입니다.
IDataCollectorSetCollection DataCollectorSet 개체의 컬렉션을 관리합니다. 이 인터페이스를 얻으려면 coCreateInstance 함수를 호출하여 __uuidof(DataCollectorSetCollection)를 클래스 식별자로 __uuidof(IDataCollectorSetCollection)을 인터페이스 식별자로 전달합니다. 그런 다음 컬렉션을 채웁니다. IDataCollectorSetCollection::GetDataCollectorSets 메서드를 호
IDataManager 보고서 생성, 데이터 보존 정책 및 데이터 전송을 포함하여 데이터 수집기에서 생성된 데이터를 관리합니다. 이 인터페이스를 얻으려면 IDataCollectorSet::D ataManage 니다.
IFolderAction 두 조건(연령 및 크기)이 모두 충족되는 경우 데이터 관리자가 데이터 수집기 집합의 루트 경로 아래에 있는 각 폴더에 대해 수행할 작업을 지정합니다. 이 인터페이스를 IFolderActionCollection::CreateFolderAction 메서드를 호출합니다.
IFolderActionCollection FolderAction 개체의 컬렉션을 관리합니다. 이 인터페이스를 얻으려면 IDataManager::FolderActions 속성에 액세스합니다.
IPerformanceCounterDataCollector 쿼리할 성능 카운터와 카운터 데이터가 기록되는 로그 파일을 지정합니다. 이 데이터 수집기를 만들려면 IDataCollectorCollection::CreateDataCollector 또는 IDataCollectorCollection::CreateDataCollectorFromXml 메서드를 호출합니다.
ISchedule 데이터 수집기 집합이 실행되는 시기를 지정합니다. 이 인터페이스를 얻으려면 IScheduleCollection::CreateSchedule 메서드를 호출합니다.
IScheduleCollection Schedule 개체의 컬렉션을 관리합니다. 이 인터페이스를 얻으려면 IDataCollectorSet::Schedules 속성에 액세스합니다.

ITraceDataCollector 등록된 공급자로부터 추적 이벤트를 수집합니다. 이 인터페이스는 추적 세션을 정의합니다.
ITraceDataProvider 추적 세션에서 사용하도록 설정할 추적 공급자를 지정합니다.
ITraceDataProviderCollection TraceDataProvider 개체의 컬렉션을 관리합니다. 이 인터페이스를 얻으려면 ITraceDataCollector::TraceDataProviders 속성에 액세스합니다. CoCreateInstance 함수를 호 TraceDataProviderCollection 개체의 새 인스턴스를 만들 수도 있습니다.
IValueMap 이름/값 쌍의 컬렉션을 관리합니다. 이 인터페이스를 얻으려면 다음 속성 또는 메서드 중 하나에 액세스합니다. IDataCollector::SetXmlIDataCollectorSet::CommitIDataCollectorSet::SetXmlITraceDataProvider::KeywordsAllITraceDataProvider::KeywordsAnyITraceDataProvider::LevelIT roperties
IValueMapItem 이름/값 쌍을 정의합니다. 이 인터페이스를 가져오려면 IValueMap::Item 속성을 호출합니다. 이 인터페이스를 만들려면 IValueMap::CreateValueMapItem 메서드를 호출

열거형

AutoPathFormat 파일 이름 또는 하위 디렉터리 이름을 데코레이트하는 방법을 정의합니다.
ClockType 이벤트를 추적할 때 사용할 클록 해상도를 정의합니다.
CommitMode 데이터 수집기 집합에 대한 변경 내용을 커밋할 때 수행할 작업을 정의합니다.
DataCollectorSetStatus 데이터 수집기 집합의 실행 상태를 정의합니다.
DataCollectorType 데이터 수집기 형식을 정의합니다.
DataManagerSteps 데이터 관리자가 실행할 때 수행하는 작업을 정의합니다.
Fileformat 로그 파일의 데이터 형식을 정의합니다.
FolderActionSteps 연령 및 크기 제한이 모두 충족되는 경우 데이터 관리자가 수행하는 작업을 정의합니다.
ResourcePolicy 디스크 리소스 제한 중 하나를 초과할 때 폴더를 삭제하는 방법을 정의합니다.
StreamMode 추적 이벤트가 전달되는 위치를 정의합니다.
ValueMapType 값의 형식을 정의합니다.

평일 데이터 수집기 집합을 실행할 요일을 정의합니다.

playtomanagerinterop.h 헤더

아티클2023. 03. 08.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

playtomanagerinterop.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IPlayToManagerInterop 여러 창을 관리하는 Windows 스토어 앱에서 PlayToManager 메서드에 액세스할 수 있습니다.

pnpxassoc.h 헤더

아티클2023. 03. 16.

이 헤더는 다른 디바이스 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 기타 디바이스 기술

pnpxassoc.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IPNPXAssociation PnP-X 디바이스에 대한 연결 데이터베이스 항목을 관리하는 메서드를 정의합니다.
IPNPXDeviceAssociation PnP-X 디바이스에 대한 연결 데이터베이스 항목을 관리하는 메서드를 정의합니다. 이러한 메서드는 해당 PnP devnode가 변경되면 알림을 보냅니다.

pnrpdef.h 헤더

아티클2023. 03. 16.

이 헤더는 피어 투 피어에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 피어 투 피어

pnrpdef.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

PNRP_CLOUD_ID PNRP_CLOUD_ID 구조에는 네트워크 클라우드를 정의하는 값이 포함됩니다.

열거형

PNRP_CLOUD_FLAGS PNRP_CLOUD_FLAGS 열거형은 클라우드 이름의 유효성을 지정합니다.
PNRP_CLOUD_STATE PNRP_CLOUD_STATE 열거형은 PNRP 클라우드가 있을 수 있는 다양한 상태를 지정합니다.
PNRP_RESOLVE_CRITERIA PNRP_RESOLVE_CRITERIA 열거형은 PNRP가 검색을 확인하는 데 사용하는 조건을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

pnrpns.h 헤더

아티클2023. 08. 25.

이 헤더는 피어 투 피어에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 피어 투 피어

pnrpns.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

PNRPCLOUDINFO PNRPCLOUDINFO 구조체는 WSAQUERYSET 구조체의 lpBlob 멤버를 가리킨다.
PNRPINFO_V1 PNRPINFO_V1 구조체는 WSAQUERYSET 구조체의 lpBlob 멤버를 가리켰습니다. (PNRPINFO_V1)
PNRPINFO_V2 PNRPINFO_V1 구조체는 WSAQUERYSET 구조체의 lpBlob 멤버를 가리켰습니다. P

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

portabledeviceapi.h 헤더

아티클2023. 03. 16.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- WPD 애플리케이션 프로그래밍 인터페이스
- WPD Automation 개체 모델

portabledeviceapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumPortableDeviceObjectIDs IEnumPortableDeviceObjectIDs 인터페이스는 이식 가능한 디바이스의 개체를 열거합니다. 디바이스에서 IPortableDeviceContent::EnumObjects를 호출하여 처음에 이 인터페이스를 가져옵니다.
IPortableDevice IPortableDevice 인터페이스는 휴대용 디바이스에 대한 액세스를 제공합니다.
IPortableDeviceCapabilities IPortableDeviceCapabilities는 지원되는 형식, 명령 및 기능 개체를 포함하여 다양한 디바이스 기능을 인터페이스합니다. IPortableDevice::Capabilities를 호출하여 디바이스에서 이 인터페이스를 검색할 수 있습니다.
IPortableDeviceContent IPortableDeviceContent 인터페이스는 디바이스에서 콘텐츠를 만들고, 열거하고, 검사하고, 삭제하는 메서드를 제공합니다. 이 인터페이스를 얻으려면 IPortableDevice::Content를 호출합니다.
IPortableDeviceContent2 디바이스에서 찾은 콘텐츠에 대한 액세스를 제공하는 추가 메서드를 정의합니다.
IPortableDeviceDataStream IPortableDeviceDataStream 인터페이스는 데이터 전송에 사용되는 IStream에 추가 메서드를 노출합니다.
IPortableDeviceDispatchFactory WPD Automation Device 개체를 인스턴스화할 수 있는 팩터리를 나타냅니다.

IPortableDeviceEventCallback 애플리케이션이 IPortableDevice::Advise를 호출하여 수신하도록 등록한 경우 비동기 콜백을 수신하도록 애플리케이션에서 구현한 IPortableDeviceEventCallback 인터페이스입니다.
IPortableDeviceManager 컴퓨터에 연결된 디바이스를 열거하고 제조업체, 식별 이름 및 설명을 포함하여 설치 정보를 요청하는 간단한 방법을 제공합니다.
IPortableDeviceProperties IPortableDeviceProperties 인터페이스는 디바이스 또는 디바이스 자체의 개체에서 속성을 검색, 추가 또는 삭제합니다.
IPortableDevicePropertiesBulk IPortableDevicePropertiesBulk 인터페이스는 디바이스의 여러 개체에서 비동기적으로 여러 속성을 쿼리하거나 설정합니다.
IPortableDevicePropertiesBulkCallback IPortableDevicePropertiesBulkCallback 인터페이스는 IPortableDevicePropertiesBulk 인터페이스를 사용하여 시작된 비동기 작업의 진행률을 추적하기 위해 애플리케이션에 의해 구현됩니다. 애플리케이션이 IPortableDevicePropertiesBulk::Start를 호출한 후 Windows 이식 가능 디바이스는 IPortableDevicePropertiesBulkCallback::OnStart를 먼저 호출한 다음 작업이 완료되거나 애플리케이션이 IPortableDevicePropertiesBulk:Cancel을 호출하거나 OnProgress에 대한 오류 값을 반환할 때까지 정보를 사용하여 IPortableDevicePropertiesBulkCallback::OnProgress를 반복적으로 호출합니다. 마지막으로 작업이 성공적으로 완료되었는지 여부에 관계없이 Windows 이식 가능한 디바이스는 IPortableDevicePropertiesBulkCallback::OnEnd를 호출합니다.
IPortableDeviceResources IPortableDeviceResources 인터페이스는 개체의 원시 데이터에 대한 액세스를 제공합니다. 이 인터페이스를 사용하여 개체에서 리소스를 읽거나 씁니다. 이 인터페이스를 얻으려면 IPortableDeviceContent::Transfer를 호출합니다.
IPortableDeviceService 서비스에 대한 액세스를 제공합니다.
IPortableDeviceServiceCapabilities 서비스의 기능을 설명하는 정보를 검색합니다.
IPortableDeviceServiceManager 서비스와 연결된 디바이스 및 디바이스에 있는 서비스 목록을 검색합니다.

IPortableDeviceServiceMethodCallback 애플리케이션이 콜백 메서드의 완료를 추적하는 데 사용하는 메서드를 포함합니다. 서비스 메서드를 비동기적으로 호출하는 애플리케이션은 이 인터페이스를 구현하고 IPortableDeviceServiceMethods::InvokeAsync에 매개 변수로 제공할 수 있습니다.
IPortableDeviceServiceMethods 서비스에서 메서드 호출을 호출하거나 취소합니다.
IPortableDeviceUnitsStream IPortableDeviceUnitsStream 인터페이스는 프레임 또는 밀리초와 같은 대체 단위를 사용하여 스트림에서 작동하거나 검색하는 방법을 제공합니다.
IPortableDeviceWebControl Windows 스토어 앱에서 WPD Automation Device 개체를 인스턴스화할 수 있는 팩터리를 나타냅니다.

portabledeviceconnectapi.h 헤더

아티클2023. 08. 25.

이 헤더는 WPD 애플리케이션 프로그래밍 인터페이스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- WPD 애플리케이션 프로그래밍 인터페이스

portabledeviceconnectapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IPortableDeviceConnector 페어링된 MTP/Bluetooth 디바이스에 대한 연결 관리 및 속성 검색에 사용되는 메서드를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

powerbase.h 헤더

아티클2023. 03. 08.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

powerbase.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CallNtPowerInformation 전원 정보를 설정하거나 검색합니다.
GetPwrCapabilities 시스템 전원 기능에 대한 정보를 검색합니다.
PowerDeterminePlatformRoleEx 지정된 플랫폼에 대한 컴퓨터 역할을 결정합니다.
PowerRegisterSuspendResumeNotification 시스템이 일시 중단되거나 다시 시작될 때 알림을 받도록 등록합니다.
PowerUnregisterSuspendResumeNotification 시스템이 일시 중단되거나 다시 시작될 때 알림을 받도록 등록을 취소합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

powersetting.h 헤더

아티클2023. 08. 25.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

powersetting.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

EFFECTIVE_POWER_MODE_CALLBACK 유효 전원 모드 콜백을 위한 함수 클래스입니다.
PowerGetActiveScheme 활성 전원 구성표를 검색하고 스키마를 식별하는 GUID를 반환합니다.
PowerReadACValue 지정된 전원 설정에 대한 AC 전원 값을 검색합니다.
PowerReadDCValue 지정된 전원 설정에 대한 DC 전원 값을 검색합니다.
PowerRegisterForEffectivePowerModeNotifications 효과적인 전원 모드 변경 알림을 수신하도록 콜백을 등록합니다.
PowerSetActiveScheme 현재 사용자의 활성 전원 구성표를 설정합니다.
PowerSettingRegisterNotification 전원 설정이 변경될 때 알림을 받도록 등록합니다.
PowerSettingUnregisterNotification 전원 설정이 변경될 때 알림을 받을 등록을 취소합니다.
PowerUnregisterFromEffectivePowerModeNotifications

유효 전원 모드 변경 알림에서 등록을 취소합니다. 이 함수는 정리 코드에서 호출되며 등록을 취소하기 전에 모든 콜백이 완료될 때까지 기다립니다.
PowerWriteACValueIndex 지정된 전원 설정의 AC 값 인덱스 를 설정합니다.
PowerWriteDCValueIndex 지정된 전원 설정의 DC 인덱스 를 설정합니다.

열거형

EFFECTIVE_POWER_MODE 시스템이 실행 중인 유효 전원 모드를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

powrprof.h 헤더

아티클2023. 08. 25.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

powrprof.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CanUserWritePwrScheme 현재 사용자에게 전원 구성표를 작성할 수 있는 충분한 권한이 있는지 여부를 결정합니다.
DeletePwrScheme 지정된 전원 구성표를 삭제합니다.
DevicePowerClose 디바이스 목록의 모든 노드를 해제하고 디바이스 목록을 삭제합니다.
DevicePowerEnumDevices 지정된 조건을 충족하는 시스템의 디바이스를 열거합니다.
DevicePowerOpen 모든 디바이스를 쿼리하여 디바이스 목록을 초기화합니다.
DevicePowerSetDeviceState 지정된 디바이스에서 지정된 데이터를 수정합니다.
EnumPwrSchemes 모든 전원 구성표를 열거합니다.
GetActivePwrScheme 활성 전원 구성표의 인덱스를 검색합니다.
GetCurrentPowerPolicies

현재 시스템 전원 정책 설정을 검색합니다.
GetPwrDiskSpindownRange 디스크 스핀다운 범위를 검색합니다.
IsPwrHibernateAllowed 컴퓨터가 최대 절전 모드를 지원하는지 여부를 결정합니다.
IsPwrShutdownAllowed 컴퓨터가 소프트 끄기 전원 상태를 지원하는지 여부를 결정합니다.
IsPwrSuspendAllowed 컴퓨터가 절전 모드 상태를 지원하는지 여부를 결정합니다.
PowerCanRestoreIndividualDefaultPowerScheme 필요한 경우 복원할 수 있도록 현재 사용자가 지정된 전원 구성표의 데이터에 액세스할 수 있는지 여부를 확인합니다.
PowerCreatePossibleSetting 지정된 전원 설정에 대해 가능한 설정 값을 만듭니다.
PowerCreateSetting 지정된 전원 설정에 대한 설정 값을 만듭니다.
PowerDeleteScheme 데이터베이스에서 지정된 전원 구성표를 삭제합니다.
PowerDeterminePlatformRole Windows 7, Windows Server 2008 R2, Windows Vista 또는 Windows Server 2008에 대한 컴퓨터 역할을 결정합니다.
PowerDuplicateScheme 기존 전원 구성표를 복제합니다.
PowerEnumerate 전원 구성표에서 지정된 요소를 열거합니다.
PowerImportPowerScheme

파일에서 전원 구성표를 가져옵니다.
PowerIsSettingRangeDefined 지정된 전원 설정이 가능한 값의 범위를 나타내는지 여부를 쿼리합니다.
PowerReadACDefaultIndex 지정된 전원 설정의 기본 AC 인덱스를 검색합니다.
PowerReadACValueIndex 지정된 전원 설정의 AC 인덱스를 검색합니다.
PowerReadDCDefaultIndex 지정된 전원 설정의 기본 DC 인덱스를 검색합니다.
PowerReadDCValueIndex 지정된 전원 설정의 DC 값 인덱스를 검색합니다.
PowerReadDescription 지정된 전원 설정, 하위 그룹 또는 체계에 대한 설명을 검색합니다.
PowerReadFriendlyName 지정된 전원 설정, 하위 그룹 또는 구성표의 이름을 검색합니다.
PowerReadIconResourceSpecifier 지정된 전원 설정, 하위 그룹 또는 체계에 대한 아이콘 리소스를 검색합니다.
PowerReadPossibleDescription 전원 설정 값의 가능한 선택 항목 중 하나에 대한 설명을 검색합니다.
PowerReadPossibleFriendlyName 전원 설정 값의 가능한 선택 항목 중 하나에 대한 식별 이름을 검색합니다.
PowerReadPossibleValue 전원 설정의 가능한 값에 대한 값을 검색합니다.
PowerReadSettingAttributes 지정된 전원 설정의 현재 특성을 반환합니다.

PowerReadValueIncrement 전원 설정 최소값과 최댓값 사이의 유효한 값에 대한 증분을 검색합니다.
PowerReadValueMax 지정된 전원 설정의 최대값을 검색합니다.
PowerReadValueMin 지정된 전원 설정의 최소값을 검색합니다.
PowerReadValueUnitsSpecifier 값 범위를 지원하는 전원 설정의 단위를 설명하는 데 사용되는 문자열을 읽습니다.
PowerRemovePowerSetting 지정된 전원 설정을 삭제합니다.
PowerReplaceDefaultPowerSchemes 기본 전원 구성표를 현재 사용자의 전원 구성표로 바꿉니다.
PowerReportThermalEvent 운영 체제에 열 이벤트를 알 수 있습니다.
PowerRestoreDefaultPowerSchemes 시스템의 전원 구성표를 기본 전원 구성표로 바꿉니다. 모든 현재 전원 구성표 및 설정이 삭제되고 기본 시스템 전원 구성표로 대체됩니다.
PowerRestoreIndividualDefaultPowerScheme 현재 사용자의 특정 전원 구성표를 기본 사용자(HKEY_USERS 저장됨)의 전원 구성표로 바꿉니다. 기본값).
PowerSettingAccessCheck 그룹 정책에 대한 쿼리는 지정된 전원 설정을 재정의합니다.
PowerSettingAccessCheckEx 그룹 정책에 대한 쿼리는 지정된 전원 설정을 재정의하고 설정에 대해 요청된 액세스를 지정합니다.
PowerWriteACDefaultIndex

지정된 전원 설정의 기본 AC 인덱스 를 설정합니다.
PowerWriteDCDefaultIndex 지정된 전원 설정의 기본 DC 인덱스 를 설정합니다.
PowerWriteDescription 지정된 전원 설정, 하위 그룹 또는 스키마에 대한 설명을 설정합니다.
PowerWriteFriendlyName 지정된 전원 설정, 하위 그룹 또는 구성표의 이름을 설정합니다.
PowerWriteIconResourceSpecifier 지정된 전원 설정, 하위 그룹 또는 스키마에 대한 아이콘 리소스를 설정합니다.
PowerWritePossibleDescription 전원 설정 값의 가능한 선택 항목 중 하나에 대한 설명을 설정합니다.
PowerWritePossibleFriendlyName 전원 설정의 가능한 지정된 설정에 대한 이름을 설정합니다.
PowerWritePossibleValue 전원 설정의 가능한 값에 대한 값을 설정합니다.
PowerWriteSettingAttributes 전원 키의 전원 특성을 설정합니다.
PowerWriteValueIncrement 전원 설정 최소값과 최댓값 사이의 유효한 값에 대한 증분을 설정합니다.
PowerWriteValueMax 지정된 전원 설정의 최대값을 설정합니다.
PowerWriteValueMin 지정된 전원 설정의 최소값을 설정합니다.
PowerWriteValueUnitsSpecifier 값 범위를 지원하는 전원 설정의 단위를 설명하는 데 사용되는 문자열을 씁니다.

ReadGlobalPwrPolicy 현재 전역 전원 정책 설정을 검색합니다.
ReadProcessorPwrScheme 지정된 전원 구성표에 대한 프로세서 전원 정책 설정을 검색합니다.
ReadPwrScheme 지정된 전원 구성표에 고유한 전원 정책 설정을 검색합니다.
SetActivePwrScheme 활성 전원 구성표를 설정합니다.
SetSuspendState 전원을 종료하여 시스템을 일시 중단합니다. Hibernate 매개 변수에 따라 시스템은 일시 중단(절전 모드) 상태 또는 최대 절전 모드(S4)를 입력합니다.
WriteGlobalPwrPolicy 전역 전원 정책 설정을 씁니다.
WriteProcessorPwrScheme 지정된 전원 구성표에 대한 프로세서 전원 정책 설정을 씁니다.
WritePwrScheme 지정된 전원 구성표에 고유한 정책 설정을 씁니다.

콜백 함수

DEVICE_NOTIFY_CALLBACK_ROUTINE 애플리케이션의 DeviceNotifyCallbackRoutine 콜백 함수는 전원 알림을 수신하는 데 사용됩니다.

구조체

DEVICE_NOTIFY_SUBSCRIBE_PARAMETERS 전원 알림에 등록할 때 사용되는 매개 변수를 포함합니다.
GLOBAL_MACHINE_POWER_POLICY 모든 사용자의 모든 전원 구성표에 적용되는 전역 컴퓨터 전원 정책 설정을 포함합니다.
GLOBAL_POWER_POLICY 모든 전원 구성표에 적용되는 전역 전원 정책 설정을 포함합니다.
GLOBAL_USER_POWER_POLICY 사용자의 모든 전원 구성표에 적용되는 전역 사용자 전원 정책 설정을 포함합니다.
MACHINE_POWER_POLICY 컴퓨터의 각 전원 구성표에 고유한 컴퓨터 전원 정책 설정을 포함합니다.
MACHINE_PROCESSOR_POWER_POLICY 시스템이 AC 전원 또는 배터리 전원에서 실행되는 동안 적용되는 프로세서 전원 정책 설정을 포함합니다.
POWER_POLICY 각 전원 구성표에 고유한 전원 정책 설정을 포함합니다.
THERMAL_EVENT 열 이벤트를 포함합니다.
USER_POWER_POLICY 사용자의 각 전원 구성표에 고유한 전원 정책 설정을 포함합니다.

열거형

POWER_DATA_ACCESSOR PowerEnumerate 및 PowerSettingAccessCheck에서 사용하는 열거형 값입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

presentation.h 헤더

아티클2023. 08. 15.

이 헤더는 컴퍼지션 스왑 체인에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 컴퍼지션 스왑 체인

presentation.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICompositionFramePresentStatistics 시스템에서 특정 현재 내의 특정 콘텐츠 instance 표시하는 방법을 설명합니다.
IIndependentFlipFramePresentStatistics 시스템에서 특정 독립 플립 프레젠테이션 내에서 특정 콘텐츠의 instance 표시하는 방법을 설명합니다.
IPresentationBuffer 프레젠테이션 관리자에 등록된 프레젠테이션 버퍼에 대해 설명합니다.
IPresentationContent 모든 프레젠테이션 콘텐츠의 기본 표현을 설명합니다.
IPresentationFactory 이 인터페이스는 프레젠테이션에 대한 시스템 지원을 쿼리하고 프레젠테이션 관리자를 만드는 데 사용됩니다.
IPresentationManager 프레젠테이션을 관리하기 위한 메서드를 정의합니다.
IPresentationSurface `IPresentationContent` 시스템 작성기와 공유하고 전역 시각적 트리에 콘텐츠로 표시할 수 있는 단일 바인딩된 버퍼가 있는 을 설명합니다.
IPresentStatistics 현재 통계 큐의 단일 항목을 나타냅니다.

IPresentStatusPresentStatistics DWM으로 구성된 프레임 동안 현재가 처리된 방법을 설명합니다.

Functions

CreatePresentationFactory 프레젠테이션 팩터리를 만듭니다.
DECLSPEC_XFGVIRT DECLSPEC_XFGVIRT 함수는 presentation.h 헤더와 함께 컴퍼지션 스왑 체인으로 개발하는 데 사용됩니다.
IIndependentFlipFramePresentStatistics_GetDisplayedTime_Proxy IIndependentFlipFramePresentStatistics_GetDisplayedTime_Proxy 함수(presentation.h)는 현재가 표시된 시간을 가져옵니다.
IIndependentFlipFramePresentStatistics_GetDisplayedTime_Stub IIndependentFlipFramePresentStatistics_GetDisplayedTime_Stub 함수는 현재가 표시된 시간을 가져옵니다.
IIndependentFlipFramePresentStatistics_GetOutputAdapterLUID_Proxy IIndependentFlipFramePresentStatistics_GetOutputAdapterLUID_Proxy 함수는 독립 대칭 이동이 발생한 디스플레이 어댑터를 나타냅니다.
IIndependentFlipFramePresentStatistics_GetOutputAdapterLUID_Stub IIndependentFlipFramePresentStatistics_GetOutputAdapterLUID_Stub 함수는 독립 대칭 이동이 발생한 디스플레이 어댑터를 나타냅니다.
IIndependentFlipFramePresentStatistics_GetPresentDuration_Proxy IIndependentFlipFramePresentStatistics_GetPresentDuration_Proxy 함수는 현재가 표시된 실제 시간을 가져옵니다.
IIndependentFlipFramePresentStatistics_GetPresentDuration_Stub IIndependentFlipFramePresentStatistics_GetPresentDuration_Stub 함수는 현재가 표시된 실제 시간을 가져옵니다.

구조체

CompositionFrameDisplayInstance 단일 출력에 표시되는 콘텐츠의 단일 instance 나타냅니다.

열거형

CompositionFrameInstanceKind 컴퍼지션 프레임에서 콘텐츠가 사용된 방법을 나타내는 상수를 정의합니다.
PresentStatus 현재 상태 지정하는 상수를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

presentationtypes.h 헤더

아티클2023. 08. 15.

이 헤더는 컴퍼지션 스왑 체인에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 컴퍼지션 스왑 체인

presentationtypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DECLSPEC_XFGVIRT DECLSPEC_XFGVIRT 함수는 컴퍼지션 스왑 체인에서 사용하는 presentationtypes.h 헤더의 일부입니다.

구조체

PresentationTransform 3-by-2 행렬에 의해 정의된 임의의 아핀 2D 변환을 나타냅니다. (PresentationTransform)
SystemInterruptTime 시스템이 마지막으로 시작된 이후의 시간(100ns 간격)을 나타냅니다.

열거형

PresentStatisticsKind 통계가 적용되는 프레젠테이션 표면의 종류를 나타내는 상수를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

print3dmanagerinterop.h 헤더

아티클2023. 03. 08.

이 헤더는 WinRT 상호 운용성에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- WinRT 상호 운용성
- Print3DManager 클래스

print3dmanagerinterop.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IPrinting3DManagerInterop WinRT Print3DManager 개체와의 상호 운용성을 사용하도록 설정합니다.

printmanagerinterop.h 헤더

아티클2023. 03. 08.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

printmanagerinterop.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IPrintManagerInterop 여러 창을 관리하는 Windows 스토어 앱에서 PrintManager 메서드에 액세스할 수 있습니다.

prnasnot.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows GDI에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows GDI

prnasnot.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IPrintAsyncNotifyCallback 인쇄 스풀러에서 호스트되는 애플리케이션 및 구성 요소에서 사용하는 통신 채널을 만들고 관리합니다.
IPrintAsyncNotifyChannel 인쇄 스풀러에서 호스트되는 구성 요소가 애플리케이션에 알림을 보내는 데 사용하는 통신 채널을 나타냅니다. 채널이 양방향인 경우 애플리케이션은 동일한 채널을 사용하여 응답을 구성 요소로 다시 보낼 수 있습니다.
IPrintAsyncNotifyDataObject 알림 채널에서 보낸 데이터를 캡슐화합니다.

함수

CreatePrintAsyncNotifyChannel 인쇄 드라이버 또는 포트 모니터와 같은 인쇄 스풀러 호스팅 인쇄 구성 요소와 구성 요소에서 알림을 받는 애플리케이션 간에 통신 채널을 만듭니다.
RegisterForPrintAsyncNotifications 애플리케이션이 프린터 드라이버, 인쇄 프로세서 및 포트 모니터와 같은 인쇄 스풀러 호스팅 인쇄 구성 요소에서 알림을 등록할 수 있도록 합니다.

UnRegisterForPrintAsyncNotifications 등록한 애플리케이션이 인쇄 스풀러 호스팅 인쇄 구성 요소에서 알림을 수신하여 등록을 취소할 수 있도록 합니다.

열거형

PrintAsyncNotifyConversationStyle 애플리케이션과 인쇄 스풀러 호스팅 구성 요소(예: 프린터 드라이버, 인쇄 프로세서 및 포트 모니터) 간의 통신이 양방향인지 아니면 단방향인지를 지정합니다.
PrintAsyncNotifyError 비동기 알림 실패 후 반환된 HRESULT의 오류 코드 부분을 지정합니다.
PrintAsyncNotifyUserFilter 알림이 인쇄 스풀러 호스팅 발신자와 동일한 사용자와 연결된 수신 대기 애플리케이션으로만 이동하거나 더 광범위한 수신 대기 애플리케이션 집합으로 이동하는지 여부를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

prntvpt.h 헤더

아티클2023. 08. 25.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows GDI
- XPS 문서

prntvpt.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

PTCloseProvider 인쇄 티켓 공급자 핸들을 닫습니다.
PTConvertDevModeToPrintTicket DEVMODE 구조를 IStream 내의 인쇄 티켓으로 변환합니다.
PTConvertPrintTicketToDevMode 인쇄 티켓을 DEVMODE 구조체로 변환합니다.
PTGetPrintCapabilities XML 인쇄 스키마에 따라 서식이 지정된 프린터의 기능을 검색합니다.
PTGetPrintDeviceCapabilities XML 인쇄 스키마에 따라 형식이 지정된 디바이스 프린터의 기능을 검색합니다.
PTGetPrintDeviceResources XML 인쇄 스키마를 준수하여 서식이 지정된 프린터의 인쇄 디바이스 리소스를 검색합니다.
PTMergeAndValidatePrintTicket 두 인쇄 티켓을 병합하고 유효한 실행 가능한 인쇄 티켓을 반환합니다.
PTOpenProvider 인쇄 티켓 공급자의 instance 엽니다. (PTOpenProvider)

PTOpenProviderEx 인쇄 티켓 공급자의 instance 엽니다. (PTOpenProviderEx)
PTQuerySchemaVersionSupport 지정한 프린터에서 지원하는 인쇄 스키마의 최고(최신) 버전을 검색합니다.
PTReleaseMemory 인쇄 티켓 및 인쇄 기능과 관련된 버퍼를 해제합니다.

열거형

EDefaultDevmodeType 인쇄 티켓이 가능한 모든 설정을 지정하지 않는 경우 사용자가 기본값의 원본으로 사용할 DEVMODE를 지정할 수 있습니다.
EPrintTicketScope 인쇄 티켓의 scope 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

processenv.h 헤더

아티클2023. 08. 25.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지
- 시스템 서비스

processenv.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

ExpandEnvironmentStringsA 환경 변수 문자열을 확장하고 현재 사용자에 대해 정의된 값으로 바꿉니다. (ANSI)
ExpandEnvironmentStringsW 환경 변수 문자열을 확장하고 현재 사용자에 대해 정의된 값으로 바꿉니다. (유니코드)
FreeEnvironmentStringsA 환경 문자열 블록을 해제합니다. (ANSI)
FreeEnvironmentStringsW 환경 문자열 블록을 해제합니다. (유니코드)
GetCommandLineA 현재 프로세스에 대한 명령줄 문자열을 검색합니다. (ANSI)
GetCommandLineW 현재 프로세스에 대한 명령줄 문자열을 검색합니다. (유니코드)
GetEnvironmentStrings GetEnvironmentStrings 함수(processenv.h)는 현재 프로세스에 대한 환경 변수를 검색합니다.
GetEnvironmentStringsW GetEnvironmentStringsW(유니코드) 함수(processenv.h)는 현재 프로세스에 대한 환경 변수를 검색합니다.

GetEnvironmentVariableA 호출 프로세스의 환경 블록에서 지정된 변수의 내용을 검색합니다. (GetEnvironmentVariableA)
GetEnvironmentVariableW GetEnvironmentVariableW(유니코드) 함수(processenv.h)는 호출 프로세스의 환경 블록에서 지정된 변수의 내용을 검색합니다.
NeedCurrentDirectoryForExePathA 현재 디렉터리를 지정된 실행 파일의 검색 경로에 포함할지 여부를 결정합니다. (ANSI)
NeedCurrentDirectoryForExePathW 현재 디렉터리를 지정된 실행 파일의 검색 경로에 포함할지 여부를 결정합니다. (유니코드)
SearchPathA 지정된 경로에서 지정된 파일을 검색합니다. (ANSI)
SearchPathW 지정된 경로에서 지정된 파일을 검색합니다. (유니코드)
SetEnvironmentStringsW SetEnvironmentStringsW(유니코드) 함수(processenv.h)는 현재 프로세스에 대한 호출 프로세스의 환경 문자열을 설정합니다.
SetEnvironmentVariableA 현재 프로세스에 대해 지정된 환경 변수의 내용을 설정합니다. (SetEnvironmentVariableA)
SetEnvironmentVariableW SetEnvironmentVariableW(유니코드) 함수(processenv.h)는 현재 프로세스에 대해 지정된 환경 변수의 내용을 설정합니다.
SetStdHandleEx SetStdHandleEx 함수(processenv.h)는 입력, 출력 또는 오류 스트림에 대한 핸들을 설정합니다.

피드백

이 페이지가 도움이 되었나요?

Yes No

Microsoft Q&A에서 도움말 보기

processsnapshot.h 헤더

아티클2023. 08. 25.

이 헤더는 프로세스 스냅샷에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 프로세스 스냅샷

processsnapshot.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

PssCaptureSnapshot 대상 프로세스의 스냅샷 캡처합니다.
PssDuplicateSnapshot 스냅샷 핸들을 한 프로세스에서 다른 프로세스로 복제합니다.
PssFreeSnapshot 스냅샷 해제합니다.
PssQuerySnapshot 스냅샷 쿼리합니다.
PssWalkMarker만들기 워크 마커를 만듭니다.
PssWalkMarkerFree PssWalkMarkerCreate에서 만든 워크 마커를 해제합니다.
PssWalkMarkerGetPosition 워크 마커의 현재 위치를 반환합니다.
PssWalkMarkerSeekToBeginning 걷기 마커를 다시 시작 부분으로 되돌아 줍니다.
PssWalkMarkerSetPosition

워크 마커의 위치를 설정합니다.
PssWalkSnapshot 현재 워크 위치에서 정보를 반환하고 다음 위치로 이동 마커를 진행합니다.

구조체

PSS_ALLOCATOR 프로세스 스냅샷 함수가 내부 워크 마커 구조를 할당하고 해제하는 데 사용하는 사용자 지정 함수를 지정합니다.
PSS_AUXILIARY_PAGE_ENTRY PssWalkSnapshot에서 반환된 보조 페이지 항목 정보를 보유합니다.
PSS_AUXILIARY_PAGES_INFORMATION PssQuerySnapshot에서 반환된 보조 페이지 정보를 보유합니다.
PSS_HANDLE_ENTRY PssWalkSnapshot에서 반환된 핸들에 대한 정보를 보유합니다.
PSS_HANDLE_INFORMATION PssQuerySnapshot에서 반환된 핸들 정보를 보유합니다.
PSS_HANDLE_TRACE_INFORMATION PssQuerySnapshot에서 반환된 핸들 추적 정보를 보유합니다.
PSS_PERFORMANCE_COUNTERS PssQuerySnapshot에서 반환된 성능 카운터를 보유합니다.
PSS_PROCESS_INFORMATION PssQuerySnapshot에서 반환된 프로세스 정보를 보유합니다.
PSS_THREAD_ENTRY PssWalkSnapshotPssWalkSnapshot에서 반환된 스레드 정보를 보유합니다.

PSS_THREAD_INFORMATION PssQuerySnapshot에서 반환된 스레드 정보를 보유합니다.
PSS_VA_CLONE_INFORMATION PssQuerySnapshot에서 반환된 VA(가상 주소) 복제 정보를 보유합니다.
PSS_VA_SPACE_ENTRY PssWalkSnapshot이 VA(가상 주소) 지역에 대해 반환한 MEMORY_BASIC_INFORMATION 보유합니다.
PSS_VA_SPACE_INFORMATION PssQuerySnapshot에서 반환된 VA(가상 주소) 공간 정보를 보유합니다.

열거형

PSS_CAPTURE_FLAGS PssCaptureSnapshot 캡처를 지정하는 플래그입니다.
PSS_DUPLICATE_FLAGS PssDuplicateSnapshot에서 사용할 중복 플래그입니다.
PSS_HANDLE_FLAGS PSS_HANDLE_ENTRY 구조체의 유효한 부분을 지정하는 플래그입니다.
PSS_OBJECT_TYPE PSS_HANDLE_ENTRY 구조체의 개체 형식을 지정합니다.
PSS_PROCESS_FLAGS 프로세스를 설명하는 플래그입니다.
PSS_QUERY_INFORMATION_CLASS PssQuerySnapshot 함수가 반환하는 정보를 지정합니다.
PSS_THREAD_FLAGS

스레드를 설명하는 플래그입니다.
PSS_WALK_INFORMATION_CLASS PssWalkSnapshot 함수가 반환하는 정보를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

processthreadsapi.h 헤더

아티클2023. 03. 08.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 프로세스 및 스레드
- 원격 데스크톱 서비스
- 보안 및 ID

processthreadsapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CreateProcessA 새 프로세스 및 기본 스레드를 만듭니다. 새 프로세스는 호출 프로세스의 보안 컨텍스트에서 실행됩니다. (ANSI)
CreateProcessAsUserA 새 프로세스 및 기본 스레드를 만듭니다. 새 프로세스는 지정된 토큰으로 표시되는 사용자의 보안 컨텍스트에서 실행됩니다. (ANSI)
CreateProcessAsUserW 새 프로세스 및 기본 스레드를 만듭니다. 새 프로세스는 지정된 토큰으로 표시되는 사용자의 보안 컨텍스트에서 실행됩니다. (유니코드)
CreateProcessW 새 프로세스 및 기본 스레드를 만듭니다. 새 프로세스는 호출 프로세스의 보안 컨텍스트에서 실행됩니다. (유니코드)
CreateRemoteThread 다른 프로세스의 가상 주소 공간에서 실행되는 스레드를 만듭니다.
CreateRemoteThreadEx 다른 프로세스의 가상 주소 공간에서 실행되는 스레드를 만들고 필요에 따라 프로세서 그룹 선호도와 같은 확장 특성을 지정합니다.

CreateThread 호출 프로세스의 가상 주소 공간 내에서 실행할 스레드를 만듭니다.
DeleteProcThreadAttributeList 프로세스 및 스레드 만들기에 대해 지정된 특성 목록을 삭제합니다.
ExitProcess 호출 프로세스 및 모든 스레드를 종료합니다.
ExitThread 호출 스레드를 종료합니다.
FlushInstructionCache 지정된 프로세스에 대한 명령 캐시를 플러시합니다.
FlushProcessWriteBuffers 현재 프로세스의 스레드를 실행하는 각 프로세서의 쓰기 큐를 플러시합니다.
GetCurrentProcess 현재 프로세스에 대한 의사 핸들을 검색합니다.
GetCurrentProcessId 호출 프로세스의 프로세스 식별자를 검색합니다.
GetCurrentProcessorNumber 이 함수를 호출하는 동안 현재 스레드가 실행 중인 프로세서 수를 검색합니다.
GetCurrentProcessorNumberEx 호출 스레드가 실행 중인 논리 프로세서의 프로세서 그룹 및 수를 검색합니다.
GetCurrentProcessToken 프로세스와 연결된 액세스 토큰을 참조하는 약식 방법으로 사용할 수 있는 의사 핸들을 검색합니다.
GetCurrentThread 호출 스레드에 대한 의사 핸들을 검색합니다.

GetCurrentThreadEffectiveToken 스레드에 대해 현재 적용되는 토큰을 참조하는 약식 방법으로 사용할 수 있는 의사 핸들을 검색합니다. 즉, 스레드 토큰이 있는 경우 스레드 토큰이고, 그렇지 않으면 프로세스 토큰입니다.
GetCurrentThreadId 호출 스레드의 스레드 식별자를 검색합니다.
GetCurrentThreadStackLimits 현재 스레드에 대해 시스템에서 할당한 스택의 경계를 검색합니다.
GetCurrentThreadToken 현재 스레드에 할당된 가장 토큰을 참조하는 약식 방법으로 사용할 수 있는 의사 핸들을 검색합니다.
GetExitCodeProcess 지정된 프로세스의 종료 상태를 검색합니다.
GetExitCodeThread 지정된 스레드의 종료 상태를 검색합니다.
GetMachineTypeAttributes 지정된 아키텍처가 현재 시스템에서 기본적으로 또는 모든 형태의 호환성 또는 에뮬레이션 계층에서 지원되는지 쿼리합니다.
GetPriorityClass 지정된 프로세스의 우선 순위 클래스를 검색합니다. 이 값은 각 프로세스 스레드의 우선 순위 값과 더불어 각 스레드의 기본 우선 순위 수준을 결정합니다.
GetProcessDefaultCpuSetMasks SetProcessDefaultCpuSetMasks 또는 SetProcessDefaultCpuSets로 설정된 프로세스 기본 집합의 CPU 집합 목록을 검색합니다.
GetProcessDefaultCpuSets SetProcessDefaultCpuSets에서 설정한 프로세스 기본 집합의 CPU 집합 목록을 검색합니다.
GetProcessHandleCount 지정된 프로세스에 속하는 열린 핸들의 수를 검색합니다.

GetProcessId 지정된 프로세스의 프로세스 식별자를 검색합니다.
GetProcessIdOfThread 지정된 스레드와 연결된 프로세스의 프로세스 식별자를 검색합니다.
GetProcessInformation 지정된 프로세스에 대한 정보를 검색합니다. (GetProcessInformation)
GetProcessMitigationPolicy 호출 프로세스에 대한 완화 정책 설정을 검색합니다.
GetProcessPriorityBoost 지정된 프로세스의 우선 순위 향상 제어 상태를 검색합니다.
GetProcessShutdownParameters 현재 호출 프로세스에 대한 종료 매개 변수를 검색합니다.
GetProcessTimes 지정된 프로세스의 타이밍 정보를 검색합니다.
GetProcessVersion 지정된 프로세스가 실행될 것으로 예상되는 시스템의 주 버전 및 부 버전 번호를 검색합니다.
GetStartupInfoW 호출 프로세스를 만들 때 지정된 STARTUPINFO 구조체의 내용을 검색합니다.
GetSystemCpuSetInformation 애플리케이션이 시스템에서 사용 가능한 CPU 집합 및 현재 상태를 쿼리할 수 있도록 허용합니다.
GetSystemTimes 시스템 타이밍 정보를 검색합니다. 다중 프로세서 시스템에서 반환되는 값은 모든 프로세서에서 지정된 시간의 합계입니다.
GetThreadContext 지정된 스레드의 컨텍스트를 검색합니다.

GetThreadDescription SetThreadDescription을 호출하여 스레드에 할당된 설명을 검색합니다.
GetThreadId 지정된 스레드의 스레드 식별자를 검색합니다.
GetThreadIdealProcessorEx 지정된 스레드에 적합한 프로세서의 프로세서 번호를 검색합니다.
GetThreadInformation 지정된 스레드에 대한 정보를 검색합니다. (GetThreadInformation)
GetThreadIOPendingFlag 지정된 스레드에 보류 중인 I/O 요청이 있는지 여부를 확인합니다.
GetThreadPriority 지정된 스레드의 우선 순위 값을 검색합니다. 이 값은 스레드 프로세스의 우선 순위 클래스와 함께 스레드의 기본 우선 순위 수준을 결정합니다.
GetThreadPriorityBoost 지정된 스레드의 우선 순위 상승 제어 상태를 검색합니다.
GetThreadSelectedCpuSetMasks SetThreadSelectedCpuSetMasks 또는 SetThreadSelectedCpuSets를 사용하여 할당이 설정된 경우 지정된 스레드의 명시적 CPU 집합 할당을 반환합니다.
GetThreadSelectedCpuSets SetThreadSelectedCpuSets API를 사용하여 할당이 설정된 경우 지정된 스레드의 명시적 CPU 집합 할당을 반환합니다.
GetThreadTimes 지정된 스레드에 대한 타이밍 정보를 검색합니다.
InitializeProcThreadAttributeList 프로세스 및 스레드 만들기에 대해 지정된 특성 목록을 초기화합니다.

IsProcessCritical 지정된 프로세스가 중요한 프로세스로 간주되는지 여부를 결정합니다.
IsProcessorFeaturePresent 지정된 프로세서 기능이 현재 컴퓨터에서 지원되는지 여부를 결정합니다.
OpenProcess 기존 로컬 프로세스 개체를 엽니다.
OpenProcessToken 프로세스와 연결된 액세스 토큰을 엽니다.
OpenThread 기존 스레드 개체를 엽니다.
OpenThreadToken 스레드와 연결된 액세스 토큰을 엽니다.
ProcessIdToSessionId 지정된 프로세스와 연결된 원격 데스크톱 서비스 세션을 검색합니다.
QueryProcessAffinityUpdateMode 지정된 프로세스의 선호도 업데이트 모드를 검색합니다.
QueryProtectedPolicy 보호된 정책과 연결된 값을 쿼리합니다.
QueueUserAPC 지정된 스레드의 APC 큐에 APC(사용자 모드 비동기 프로시저 호출) 개체를 추가합니다. (QueueUserAPC)
QueueUserAPC2 지정된 스레드의 APC 큐에 APC(사용자 모드 비동기 프로시저 호출) 개체를 추가합니다. (QueueUserAPC2)

ResumeThread 스레드의 일시 중단 횟수를 감소합니다. 일시 중단 횟수가 0으로 감소하면 스레드 실행이 다시 시작됩니다.
SetPriorityClass 지정된 프로세스의 우선 순위 클래스를 설정합니다. 이 값은 프로세스의 각 스레드의 우선 순위 값과 함께 각 스레드의 기본 우선 순위 수준을 결정합니다.
SetProcessAffinityUpdateMode 지정된 프로세스의 선호도 업데이트 모드를 설정합니다.
SetProcessDefaultCpuSetMasks SetProcessDefaultCpuSetMasks 함수(processthreadsapi.h)는 지정된 프로세스의 스레드에 대한 기본 CPU 집합 할당을 설정합니다.
SetProcessDefaultCpuSets SetProcessDefaultCpuSets 함수(processthreadsapi.h)는 지정된 프로세스의 스레드에 대한 기본 CPU 집합 할당을 설정합니다.
SetProcessDynamicEHContinuationTargets 지정된 프로세스에 대한 동적 예외 처리 연속 대상을 설정합니다.
SetProcessDynamicEnforcedCetCompatibleRanges 지정된 프로세스에 대해 동적으로 적용되는 CETCOMPAT 범위를 설정합니다.
SetProcessInformation 지정된 프로세스에 대한 정보를 설정합니다.
SetProcessMitigationPolicy 호출 프로세스에 대한 완화 정책을 설정합니다. 완화 정책을 사용하면 프로세스가 다양한 유형의 공격에 대해 자신을 강화할 수 있습니다.
SetProcessPriorityBoost 지정된 프로세스의 스레드 우선 순위를 일시적으로 높일 수 있는 시스템 기능을 사용하지 않도록 설정하거나 사용하도록 설정합니다.

SetProcessShutdownParameters 현재 호출하는 프로세스에 대한 종료 매개 변수를 설정합니다. 이 함수는 시스템의 다른 프로세스와 관련된 프로세스에 대한 종료 순서를 설정합니다.
SetProtectedPolicy 보호된 정책을 설정합니다.
SetThreadContext 지정된 스레드의 컨텍스트를 설정합니다.
SetThreadDescription 스레드에 설명을 할당합니다.
SetThreadIdealProcessor 스레드에 대한 기본 프로세서를 설정합니다. 시스템은 가능하면 원하는 프로세서에서 스레드를 예약합니다.
SetThreadIdealProcessorEx 지정된 스레드에 이상적인 프로세서를 설정하고 필요에 따라 이전 이상적인 프로세서를 검색합니다.
SetThreadInformation 지정된 스레드에 대한 정보를 설정합니다.
SetThreadPriority 지정된 스레드의 우선 순위 값을 설정합니다. 이 값은 스레드 프로세스의 우선 순위 클래스와 함께 스레드의 기본 우선 순위 수준을 결정합니다.
SetThreadPriorityBoost 스레드의 우선 순위를 일시적으로 높일 수 있는 시스템 기능을 사용하지 않도록 설정하거나 활성화합니다.
SetThreadSelectedCpuSetMasks 지정된 스레드에 대해 선택한 CPU 집합 할당을 설정합니다. 이 할당은 프로세스가 설정된 경우 프로세스 기본 할당을 재정의합니다. (SetThreadSelectedCpuSetMasks)

SetThreadSelectedCpuSets 지정된 스레드에 대해 선택한 CPU 집합 할당을 설정합니다. 이 할당은 프로세스가 설정된 경우 프로세스 기본 할당을 재정의합니다. (SetThreadSelectedCpuSets)
SetThreadStackGuarantee 스택 오버플로 예외 중에 사용할 수 있는 호출 스레드 또는 파이버와 연결된 스택의 최소 크기를 설정합니다.
SetThreadToken 스레드에 가장 토큰을 할당합니다. 함수는 스레드가 가장 토큰 사용을 중지하도록 할 수도 있습니다.
SuspendThread 지정된 스레드를 일시 중단합니다.
SwitchToThread 호출 스레드가 현재 프로세서에서 실행할 준비가 되어 있는 다른 스레드에 실행 명령을 내리도록 합니다. 운영 체제는 실행할 다음 스레드를 선택합니다.
TerminateProcess 지정된 프로세스와 모든 스레드를 종료합니다.
TerminateThread 스레드를 종료합니다.
TlsAlloc TLS(스레드 로컬 스토리지) 인덱스를 할당합니다. 각 스레드가 인덱스에 대한 자체 슬롯을 수신하기 때문에 프로세스의 모든 스레드는 이후에 이 인덱스를 사용하여 스레드에 로컬인 값을 저장하고 검색할 수 있습니다.
TlsFree 스레드 TLS(로컬 스토리지) 인덱스를 해제하여 다시 사용할 수 있도록 합니다.
TlsGetValue 지정된 TLS 인덱스에 대한 호출 스레드의 TLS(스레드 로컬 스토리지) 슬롯에서 값을 검색합니다. 각 프로세스의 스레드에는 각 TLS 인덱스에 대한 슬롯이 있습니다.

TlsSetValue 지정된 TLS 인덱스의 호출 스레드 TLS(로컬 스토리지) 슬롯에 값을 저장합니다. 각 프로세스의 스레드에는 각 TLS 인덱스에 대한 슬롯이 있습니다.
UpdateProcThreadAttribute 프로세스 및 스레드 만들기를 위한 특성 목록에서 지정된 특성을 업데이트.

구조체

APP_MEMORY_INFORMATION 단일 시점의 앱 메모리 사용량을 나타냅니다. 이 구조체는 PROCESS_INFORMATION_CLASS 클래스에서 사용됩니다.
MEMORY_PRIORITY_INFORMATION 스레드 또는 프로세스의 메모리 우선 순위를 지정합니다.
PROCESS_INFORMATION 새로 만든 프로세스 및 기본 스레드에 대한 정보를 포함합니다. CreateProcess, CreateProcessAsUser, CreateProcessWithLogonW 또는 CreateProcessWithTokenW 함수와 함께 사용됩니다.
PROCESS_LEAP_SECOND_INFO 시스템에서 양수 윤초를 처리하는 방법을 지정합니다.
PROCESS_MACHINE_INFORMATION 프로세스의 아키텍처를 지정하고 해당 코드 아키텍처가 호스트 운영 체제의 사용자 모드, 커널 모드 및/또는 WoW64에서 실행될 수 있는지 여부를 지정합니다.
PROCESS_MEMORY_EXHAUSTION_INFO 할당이 메모리를 커밋하지 못하는 경우 애플리케이션에서 프로세스를 종료하도록 구성할 수 있습니다. 이 구조체는 PROCESS_INFORMATION_CLASS 클래스에서 사용됩니다.
PROCESS_POWER_THROTTLING_STATE 제한 정책과 해당 프로세스에 전원 관리가 적용될 때 대상 프로세스에 적용하는 방법을 지정합니다.

PROCESS_PROTECTION_LEVEL_INFORMATION PPL(Protected Process Light)을 사용할지 여부를 지정합니다.
STARTUPINFOA 생성 시 프로세스에 대한 주 창의 창 스테이션, 바탕 화면, 표준 핸들 및 모양을 지정합니다. (ANSI)
STARTUPINFOW 생성 시 프로세스에 대한 주 창의 창 스테이션, 바탕 화면, 표준 핸들 및 모양을 지정합니다. (유니코드)
THREAD_POWER_THROTTLING_STATE 제한 정책과 해당 스레드에 전원 관리가 적용될 때 대상 스레드에 적용하는 방법을 지정합니다.

열거형

MACHINE_ATTRIBUTES 호스트 운영 체제에서 코드 아키텍처를 실행할 수 있는 방법을 지정합니다. 둘 이상의 비트를 설정할 수 있습니다.
PROCESS_INFORMATION_CLASS 프로세스 정보의 특정 클래스를 나타냅니다.
PROCESS_MEMORY_EXHAUSTION_TYPE 다양한 메모리 소모 유형을 나타냅니다.
QUEUE_USER_APC_FLAGS QUEUE_USER_APC_FLAGS 열거형(processthreadsapi.h)은 APC(사용자 모드 비동기 프로시저 호출) 개체에 대한 한정자 플래그를 지정합니다.
THREAD_INFORMATION_CLASS THREAD_INFORMATION_CLASS 열거형(processthreadsapi.h)은 지원되는 스레드 형식의 컬렉션을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

processtopologyapi.h 헤더

아티클2023. 03. 08.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

processtopologyapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

GetProcessGroupAffinity 지정된 프로세스의 프로세서 그룹 선호도를 검색합니다.
GetThreadGroupAffinity 지정된 스레드의 프로세서 그룹 선호도를 검색합니다.
SetThreadGroupAffinity 지정된 스레드에 대한 프로세서 그룹 선호도를 설정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

profileapi.h 헤더

아티클2023. 08. 25.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

profileapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

QueryPerformanceCounter 시간 간격 측정에 사용할 수 있는 고해상도(<1us) 타임스탬프를 나타내는 성능 카운터의 현재 값을 검색합니다.
QueryPerformanceFrequency 성능 카운터의 빈도를 검색합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

profinfo.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

profinfo.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

PROFILEINFOA 사용자 프로필을 로드하거나 언로드할 때 사용되는 정보를 포함합니다. (ANSI)
PROFILEINFOW 사용자 프로필을 로드하거나 언로드할 때 사용되는 정보를 포함합니다. (유니코드)

projectedfslib.h 헤더

아티클2023. 08. 25.

이 헤더는 프로젝션된 파일 시스템에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 프로젝션된 파일 시스템

projectedfslib.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

PrjAllocateAlignedBuffer 가상화 instance 스토리지 디바이스의 메모리 맞춤 요구 사항을 충족하는 버퍼를 할당합니다.
PrjClearNegativePathCache 가상화 instance 음수 경로 캐시(활성 상태인 경우)를 제거합니다.
PrjCompleteCommand 공급자가 이전에 HRESULT_FROM_WIN32(ERROR_IO_PENDING)을 반환한 콜백 처리를 완료했음을 나타냅니다.
PrjDeleteFile 공급자가 로컬 파일 시스템에 캐시된 항목을 삭제할 수 있도록 합니다.
PrjDoesNameContainWildCards 이름에 와일드카드 문자가 포함되어 있는지 여부를 확인합니다.
PrjFileNameCompare 두 파일 이름을 비교하고 상대 데이터 정렬 순서를 나타내는 값을 반환합니다.
PrjFileNameMatch 파일 이름이 검색 패턴과 일치하는지 여부를 결정합니다.
PrjFillDirEntryBuffer 열거형에 하나의 파일 또는 디렉터리에 대한 정보를 제공합니다.
PrjFillDirEntryBuffer2

하나의 파일 또는 디렉터리에 대한 정보를 열거형에 제공하고 호출자가 확장 정보를 지정할 수 있도록 합니다.
PrjFreeAlignedBuffer 할당된 버퍼를 해제합니다.
PrjGetOnDiskFileState 파일 또는 디렉터리에 대한 디스크 내 파일 상태를 가져옵니다.
PrjGetVirtualizationInstanceInfo 가상화 instance 대한 정보를 검색합니다.
PrjMarkDirectoryAsPlaceholder 기존 디렉터리를 디렉터리 자리 표시자로 변환합니다.
PrjStartVirtualizing ProjFS 가상화 instance 구성하고 이를 시작하여 I/O를 서비스하고 공급자에서 콜백을 호출할 수 있도록 합니다.
PrjStopVirtualizing 실행 중인 ProjFS 가상화 instance 중지하여 I/O를 서비스하거나 공급자의 콜백을 포함할 수 없습니다.
PrjUpdateFileIfNeeded 공급자가 로컬 파일 시스템에 캐시된 항목을 업데이트할 수 있도록 합니다.
PrjWriteFileData PrjWriteFileData 함수는 PRJ_GET_FILE_DATA_CB 콜백 호출에서 요청된 데이터를 제공합니다. (PrjWriteFileData)
PrjWritePlaceholderInfo 파일 또는 디렉터리 메타데이터를 ProjFS로 보냅니다.
PrjWritePlaceholderInfo2 파일 또는 디렉터리 메타데이터를 ProjFS로 보내고 호출자가 확장 정보를 지정할 수 있도록 합니다.

콜백 함수

PRJ_CANCEL_COMMAND_CB 콜백의 이전 호출에 의한 작업을 취소해야 한다는 것을 공급자에게 알릴 수 있습니다.
PRJ_END_DIRECTORY_ENUMERATION_CB 디렉터리 열거가 끝났다는 것을 공급자에게 알릴 수 있습니다.
PRJ_GET_DIRECTORY_ENUMERATION_CB 공급자로부터 디렉터리 열거 정보를 요청합니다.
PRJ_GET_FILE_DATA_CB 파일의 기본 데이터 스트림의 콘텐츠를 요청합니다.
PRJ_GET_PLACEHOLDER_INFO_CB 공급자의 파일 또는 디렉터리에 대한 정보를 요청합니다.
PRJ_NOTIFICATION_CB 파일 시스템 작업에 대한 알림을 공급자에게 전달합니다.
PRJ_QUERY_FILE_NAME_CB 지정된 파일 경로가 공급자의 백업 저장소에 있는지 여부를 확인합니다.
PRJ_START_DIRECTORY_ENUMERATION_CB 디렉터리 열거가 시작되고 있음을 공급자에게 알릴 수 있습니다.

구조체

PRJ_CALLBACK_DATA 모든 작업 콜백에 대해 공급자에 전달되는 표준 정보를 정의합니다.
PRJ_CALLBACKS 공급자가 콜백의 구현을 저장하는 콜백 루틴 집합입니다.
PRJ_COMPLETE_COMMAND_EXTENDED_PARAMETERS 특정 콜백을 완료하는 데 필요한 매개 변수를 지정합니다.

PRJ_EXTENDED_INFO 디렉터리 열거형 및 자리 표시자 정보에 대한 선택적 확장 정보를 지정합니다.
PRJ_FILE_BASIC_INFO 항목에 대한 기본 정보입니다.
PRJ_NOTIFICATION_MAPPING 디렉터리("알림 루트"라고 함)와 비트 마스크로 표현되는 알림 집합 간의 페어링인 알림 매핑에 대해 설명합니다.
PRJ_NOTIFICATION_PARAMETERS 알림에 대한 추가 매개 변수입니다.
PRJ_PLACEHOLDER_INFO 자리 표시자 파일 또는 디렉터리에 대한 메타데이터 버퍼입니다.
PRJ_PLACEHOLDER_VERSION_INFO 자리 표시자 파일의 내용을 고유하게 식별하는 정보입니다.
PRJ_STARTVIRTUALIZING_OPTIONS 가상화 instance 시작할 때 제공하는 옵션입니다.
PRJ_VIRTUALIZATION_INSTANCE_INFO 가상화 instance 대한 정보입니다.

열거형

PRJ_CALLBACK_DATA_FLAGS 열거형에서 반환되는 내용을 제어하는 플래그입니다.
PRJ_COMPLETE_COMMAND_TYPE 명령 유형을 지정합니다.
PRJ_EXT_INFO_TYPE

확장 정보 유형을 지정합니다.
PRJ_FILE_STATE 항목의 상태입니다.
PRJ_NOTIFICATION 콜백에서 알림을 보낼 때 지정된 알림 값입니다.
PRJ_NOTIFY_TYPES 파일 또는 폴더의 변경 사항을 설명하는 알림 유형입니다.
PRJ_PLACEHOLDER_ID 자리 표시자 식별자의 길이를 정의합니다.
PRJ_STARTVIRTUALIZING_FLAGS 가상화 instance 시작할 때 제공할 플래그입니다.
PRJ_UPDATE_FAILURE_CAUSES 업데이트가 실패한 이유에 대한 설명입니다.
PRJ_UPDATE_TYPES 디스크의 파일 또는 디렉터리 상태를 고려하여 업데이트를 허용할지 여부를 지정하는 플래그입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

proofofpossessioncookieinfo.h 헤더

아티클2023. 04. 09.

이 헤더는 Windows 인터넷에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 인터넷

proofofpossessioncookieinfo.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IProofOfPossessionCookieInfoManager 소유 증명 쿠키 만들기를 지원합니다.
IProofOfPossessionCookieInfoManager2 WebAccount에 대한 소유 증명 쿠키 만들기를 지원합니다.

Functions

FreeProofOfPossessionCookieInfoArray

구조체

ProofOfPossessionCookieInfo

propapi.h 헤더

아티클2023. 03. 16.

이 헤더는 Structured Storage에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 구조적 저장소

propapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

StgPropertyLengthAsVariant StgPropertyLengthAsVariant 함수는 SERIALIZEDPROPERTYVALUE를 검사하고 이 속성이 PROPVARIANT로 차지하는 메모리 양을 반환합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

propidl.h 헤더

아티클2023. 08. 25.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 구조적 저장소
- Windows 셸
- Windows 속성 시스템

propidl.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumSTATPROPSETSTG IEnumSTATPROPSETSTG 인터페이스는 현재 IPropertySetStorage instance 관리되는 속성 집합에 대한 통계 데이터를 포함하는 STATPROPSETSTG 구조의 배열을 반복합니다.
IEnumSTATPROPSTG IEnumSTATPROPSTG 인터페이스는 STATPROPSTG 구조체의 배열을 반복합니다. STATPROPSTG 구조에는 속성 집합의 속성에 대한 통계 데이터가 포함됩니다.
IPropertySetStorage IPropertySetStorage 인터페이스는 IPropertyStorage 인터페이스의 인스턴스를 지원하는 속성 집합 스토리지를 만들고, 열고, 삭제하고, 열거합니다.
IPropertyStorage IPropertyStorage 인터페이스는 단일 속성 집합의 영구 속성을 관리합니다. (IPropertyStorage 인터페이스)

Functions

FreePropVariantArray PROPVARIANT 구조체 배열에서 사용하는 메모리 및 참조를 해제합니다.
PropVariantClear

PROPVARIANT 구조를 지웁니다.
PropVariantCopy PROPVARIANT 구조체의 복사본을 만듭니다.
PropVariantInit PropVariantInit 함수는 PROPVARIANT 구조를 초기화합니다. 참고 이 함수는 제공된 ole2.h 헤더 파일을 포함하여 사용할 수 있는 매크로로 구현됩니다.
StgConvertPropertyToVariant SERIALIZEDPROPERTYVALUE 데이터 형식을 PROPVARIANT 데이터 형식으로 변환합니다.
StgConvertVariantToProperty PROPVARIANT 데이터 형식을 SERIALIZEDPROPERTYVALUE 데이터 형식으로 변환합니다.

구조체

PROPSPEC PROPSPEC 구조체는 많은 IPropertyStorage 메서드에서 속성 식별자(ID) 또는 연결된 문자열 이름으로 속성을 지정하는 데 사용됩니다.
PROPVARIANT PROPVARIANT 구조체는 IPropertyStorage의 ReadMultiple 및 WriteMultiple 메서드에서 속성 집합에 있는 속성의 형식 태그 및 값을 정의하는 데 사용됩니다.
SERIALIZEDPROPERTYVALUE 직렬화된 PROPVARIANT 구조를 나타내는 임의 형식의 메모리 범위입니다.
STATPROPSETSTG STATPROPSETSTG 구조체에는 속성 집합에 대한 정보가 포함됩니다. (STATPROPSETSTG 구조체)
STATPROPSTG STATPROPSTG 구조체에는 속성 집합의 단일 속성에 대한 데이터가 포함됩니다. 이 데이터는 속성 ID 및 형식 태그이며 속성과 연결될 수 있는 선택적 문자열 이름입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

propidlbase.h 헤더

아티클2023. 08. 25.

이 헤더는 Structured Storage에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 구조적 저장소

propidlbase.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumSTATPROPSETSTG IEnumSTATPROPSETSTG 인터페이스는 현재 IPropertySetStorage instance 관리되는 속성 집합에 대한 통계 데이터를 포함하는 STATPROPSETSTG 구조체의 배열을 반복합니다.
IEnumSTATPROPSTG IEnumSTATPROPSTG 인터페이스는 속성 집합의 속성에 대한 통계 데이터를 포함하는 STATPROPSTG 구조체의 배열을 반복합니다.
IPropertyStorage IPropertyStorage 인터페이스는 단일 속성 집합의 영구 속성을 관리합니다. (IPropertyStorage 인터페이스)

구조체

PROPSPEC PROPSPEC 구조체는 IPropertyStorage의 많은 메서드에서 속성 식별자(ID) 또는 연결된 문자열 이름으로 속성을 지정하는 데 사용됩니다.
PROPVARIANT PROPVARIANT 구조체는 IPropertyStorage의 ReadMultiple 및 WriteMultiple 메서드에서 속성 집합에 있는 속성의 형식 태그 및 값을 정의하는 데 사용됩니다.
STATPROPSETSTG STATPROPSETSTG 구조체에는 속성 집합에 대한 정보가 포함됩니다. (STATPROPSETSTG 구조체)

STATPROPSTG STATPROPSTG 구조체에는 속성 집합의 단일 속성에 대한 데이터가 포함됩니다. 이 데이터는 속성 ID 및 형식 태그이며 속성과 연결할 수 있는 선택적 문자열 이름입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

propkeydef.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

propkeydef.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DEFINE_PROPERTYKEY FMTID(형식 식별자) 및 PID(속성 식별자)를 속성 키를 나타내는 PROPERTYKEY 구조로 압축하는 데 사용됩니다.
IsEqualPropertyKey 두 PROPERTYKEY 구조체의 멤버를 비교하고 같은지 여부를 반환합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

propsys.h 헤더

아티클2023. 08. 25.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 오디오 디바이스 DDI 참조
- Windows 셸
- Windows 속성 시스템

propsys.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICreateObject 지정된 클래스의 개체를 만드는 메서드를 노출합니다.
IDelayedPropertyStoreFactory 속성 액세스가 느릴 수 있는 상황에서 지정된 IPropertyStore 개체를 만드는 메서드를 노출합니다.
IInitializeWithFile 파일 경로를 사용하여 속성 처리기, 미리 보기 처리기 또는 미리 보기 처리기와 같은 처리기를 초기화하는 메서드를 노출합니다.
IInitializeWithStream 스트림을 사용하여 속성 처리기, 썸네일 처리기 또는 미리 보기 처리기와 같은 처리기를 초기화하는 메서드를 노출합니다.
INamedPropertyStore 명명된 속성을 가져와서 설정하는 메서드를 노출합니다.
IObjectWithPropertyKey 속성 키를 가져오고 설정하기 위한 메서드를 노출합니다.
IPersistSerializedPropStorage 나중에 사용할 수 있도록 직렬화된 속성 스토리지 데이터를 유지하며 지속형 데이터를 새 속성 저장소 instance 복원하는 메서드를 노출합니다. (IPersistSerializedPropStorage)

IPersistSerializedPropStorage2 나중에 사용할 수 있도록 직렬화된 속성 스토리지 데이터를 유지하며 지속형 데이터를 새 속성 저장소 instance 복원하는 메서드를 노출합니다. (IPersistSerializedPropStorage2)
IPropertyChange 단일 속성에 대한 변경 사항을 캡슐화하는 메서드를 노출합니다.
IPropertyChangeArray IFileOperation에 전달될 수 있는 여러 변경 작업에 대한 메서드를 노출합니다.
IPropertyDescription 개별 속성 설명 세부 정보를 열거하고 검색하는 메서드를 노출합니다. (IPropertyDescription)
IPropertyDescription2 개별 속성 설명 세부 정보를 열거하고 검색하는 메서드를 노출합니다. (IPropertyDescription2)
IPropertyDescriptionAliasInfo 항목에 대한 "정렬 기준" 열 속성을 가져오는 메서드를 노출합니다. 이 인터페이스는 지정된 속성에 대한 기본 또는 보조 정렬 열을 검색하려는 UI 개체에서 사용됩니다.
IPropertyDescriptionList 목록으로 제공되는 속성 설명 컬렉션에서 정보를 추출하는 메서드를 노출합니다.
IPropertyDescriptionRelatedPropertyInfo IPropertyDescription 인터페이스를 검색하는 메서드를 제공합니다.
IPropertyDescriptionSearchInfo 속성에 대한 검색 관련 정보를 노출합니다.
IPropertyEnumType 열거형 정보에서 데이터를 추출하는 메서드를 노출합니다. IPropertyEnumType은 런타임에 프로그래밍 방식으로 속성 스키마의 enum 및 enumRange 요소에 대한 액세스를 제공합니다.
IPropertyEnumType2 열거형 정보에서 데이터를 추출하는 메서드를 노출합니다. IPropertyEnumType2는 IPropertyEnumType을 확장합니다.

IPropertyEnumTypeList 속성에 대해 가능한 값을 열거하는 메서드를 노출합니다.
IPropertyStore 이 인터페이스는 속성 값을 열거하고 조작하는 데 사용되는 메서드를 노출합니다.
IPropertyStoreCache 처리기가 각 속성의 다양한 상태를 관리할 수 있도록 하는 메서드를 노출합니다.
IPropertyStoreCapabilities 사용자가 UI에서 속성을 편집할 수 있는지 여부를 결정하는 메서드를 노출합니다.
IPropertyStoreFactory 메서드를 노출하여 IPropertyStore 개체를 가져옵니다.
IPropertySystem 속성 설명을 얻고, 속성 스키마를 등록 및 등록 취소하고, 속성 설명을 열거하고, 형식이 엄격한 방식으로 속성 값의 서식을 지정하는 메서드를 노출합니다.

Functions

PropVariantToWinRTPropertyValue PROPVARIANT 구조에서 Windows 런타임 속성 값으로 데이터를 추출합니다.
PSCoerceToCanonicalValue 속성 설명에 따라 속성 값을 정식 값으로 변환합니다.
PSCreateAdapterFromPropertyStore IPropertyStore에서 어댑터를 만듭니다.
PSCreateDelayedMultiplexPropertyStore 여러 속성 저장소를 포함하는 읽기 전용 지연 바인딩 속성 저장소를 만듭니다.
PSCreateMemoryPropertyStore

메모리 내 속성 저장소를 만듭니다.
PSCreateMultiplexPropertyStore 여러 속성 저장소가 포함된 읽기 전용 속성 저장소를 만듭니다. 각 저장소는 IPropertyStore 또는 IPropertySetStorage를 지원해야 합니다.
PSCreatePropertyChangeArray IPropertyChange 개체 집합에 대한 컨테이너를 만듭니다. 이 컨테이너는 IFileOperation과 함께 사용하여 파일 집합에 속성 변경 내용 집합을 적용할 수 있습니다.
PSCreatePropertyStoreFromObject IPropertyStore 또는 IPropertySetStorage를 지원하는 개체의 IUnknown 인터페이스를 허용합니다. 개체가 IPropertySetStorage를 지원하는 경우 IPropertyStore를 지원하게 래핑됩니다.
PSCreatePropertyStoreFromPropertySetStorage IPropertyStore 인터페이스에서 IPropertySetStorage 인터페이스를 래핑합니다.
PSCreateSimplePropertyChange 간단한 속성 변경을 만듭니다.
PSEnumeratePropertyDescriptions 스키마 하위 시스템의 IPropertySystem::EnumeratePropertyDescriptions를 호출하는 래퍼 API입니다.
PSFormatForDisplay PROPVARIANT 구조체에 저장된 속성 값의 형식이 지정된 유니코드 문자열 표현을 가져옵니다. 호출자는 출력 버퍼를 할당해야 합니다.
PSFormatForDisplayAlloc PROPVARIANT 구조체에 저장된 속성 값의 형식이 지정된 유니코드 문자열 표현을 가져옵니다. 이 함수는 출력 문자열에 대한 메모리를 할당합니다.
PSFormatPropertyValue 속성 저장소에 저장된 속성 값의 형식이 지정된 유니코드 문자열 표현을 가져옵니다. 이 함수는 출력 문자열에 대한 메모리를 할당합니다.
PSGetImageReferenceForValue 지정된 속성에 대한 속성 설명 인터페이스의 instance 가져옵니다.

PSGetItemPropertyHandler 셸 항목에 대한 속성 처리기를 검색합니다. (PSGetItemPropertyHandler)
PSGetItemPropertyHandlerWithCreateObject 셸 항목에 대한 속성 처리기를 검색합니다. (PSGetItemPropertyHandlerWithCreateObject)
PSGetNamedPropertyFromPropertyStorage 속성 이름으로 직렬화된 속성 스토리지에서 값을 가져옵니다.
PSGetNameFromPropertyKey PROPERTYKEY가 지정된 경우 속성의 정식 이름을 검색합니다.
PSGetPropertyDescription PROPERTYKEY 구조체로 지정된 속성에 대한 속성 설명 인터페이스의 instance 가져옵니다.
PSGetPropertyDescriptionByName 지정된 속성 이름에 대한 속성 설명 인터페이스의 instance 가져옵니다.
PSGetPropertyDescriptionListFromString 지정된 속성 목록에 대한 속성 설명 목록 인터페이스의 instance 가져옵니다.
PSGetPropertyFromPropertyStorage 직렬화된 속성 스토리지에 저장된 속성의 값을 가져옵니다.
PSGetPropertyKeyFromName 정식 속성 이름의 속성 키를 가져옵니다.
PSGetPropertySystem IPropertySystem을 구현하는 하위 시스템 개체의 instance 가져옵니다.
PSGetPropertyValue 속성 저장소에서 속성 값을 가져옵니다.
PSLookupPropertyHandlerCLSID 컴퓨터별 등록된 파일 속성 처리기의 CLSID(클래스 식별자)를 가져옵니다.
PSPropertyBag_Delete

속성 모음에서 속성을 삭제합니다.
PSPropertyBag_ReadBOOL 속성 모음에서 속성의 BOOL 데이터 값을 읽습니다.
PSPropertyBag_ReadBSTR 속성 모음의 속성에서 BSTR 데이터 값을 읽습니다.
PSPropertyBag_ReadDWORD 속성 모음의 속성에서 DWORD 데이터 값을 읽습니다.
PSPropertyBag_ReadGUID 속성 모음의 속성에서 GUID 데이터 값을 읽습니다.
PSPropertyBag_ReadInt 속성 모음의 속성에서 int 데이터 값을 읽습니다.
PSPropertyBag_ReadLONG 속성 모음의 속성에서 LONG 데이터 값을 읽습니다.
PSPropertyBag_ReadPOINTL 지정된 속성 모음의 POINTL 구조에 저장된 속성 좌표를 검색합니다.
PSPropertyBag_ReadPOINTS 지정된 속성 모음의 POINTS 구조에 저장된 속성 좌표를 검색합니다.
PSPropertyBag_ReadPropertyKey 지정된 속성 모음에서 속성의 속성 키를 읽습니다.
PSPropertyBag_ReadRECTL 지정된 속성 모음에 포함된 속성에 저장된 사각형의 좌표를 검색합니다.
PSPropertyBag_ReadSHORT 속성 모음에서 속성의 SHORT 데이터 값을 읽습니다.
PSPropertyBag_ReadStr 속성 모음에서 속성의 문자열 데이터 값을 읽습니다.

PSPropertyBag_ReadStrAlloc 속성 모음의 속성에서 문자열 데이터 값을 읽고 읽는 문자열에 대한 메모리를 할당합니다.
PSPropertyBag_ReadStream 지정된 속성 모음에 포함된 지정된 속성에 저장된 데이터 스트림을 읽습니다.
PSPropertyBag_ReadType 속성 모음에 저장된 속성의 데이터 값 형식을 읽습니다.
PSPropertyBag_ReadULONGLONG 속성 모음의 속성에서 ULONGLONG 데이터 값을 읽습니다.
PSPropertyBag_ReadUnknown 속성 모음에서 알 수 없는 데이터 값의 지정된 속성을 읽습니다.
PSPropertyBag_WriteBOOL 속성 모음에 있는 속성의 BOOL 값을 설정합니다.
PSPropertyBag_WriteBSTR 속성 모음에 있는 속성의 BSTR 값을 설정합니다.
PSPropertyBag_WriteDWORD 속성 모음에 있는 속성의 DWORD 값을 설정합니다.
PSPropertyBag_WriteGUID 속성 모음에서 속성의 GUID 값을 설정합니다.
PSPropertyBag_WriteInt 속성 모음에서 속성의 int 값을 설정합니다.
PSPropertyBag_WriteLONG 속성 모음에 있는 속성의 LONG 값을 설정합니다.
PSPropertyBag_WritePOINTL 속성 좌표를 지정된 속성 모음의 APOINTL 구조에 저장합니다.
PSPropertyBag_WritePOINTS

속성 좌표를 지정된 속성 모음의 APOINTS 구조에 저장합니다.
PSPropertyBag_WritePropertyKey 속성 모음에서 속성의 속성 키 값을 설정합니다.
PSPropertyBag_WriteRECTL 속성 모음의 속성에 사각형의 좌표를 저장합니다.
PSPropertyBag_WriteSHORT 속성 모음에 있는 속성의 SHORT 값을 설정합니다.
PSPropertyBag_WriteStr 속성 모음에 있는 속성의 문자열 값을 설정합니다.
PSPropertyBag_WriteStream 속성 모음의 속성에 데이터 스트림을 씁니다.
PSPropertyBag_WriteULONGLONG 속성 모음에 있는 속성의 ULONGLONG 값을 설정합니다.
PSPropertyBag_WriteUnknown 알 수 없는 데이터 값의 속성을 속성 모음에 씁니다.
PSPropertyKeyFromString 문자열을 PROPERTYKEY 구조체로 변환합니다.
PSRefreshPropertySchema 지원되지 않습니다. 이 함수를 호출하는 것은 유효하지만 함수를 수행하기 위해 구현되지 않으므로 그렇게 할 이유가 없습니다.
PSRegisterPropertySchema 속성 설명 스키마 파일이 추가된 것을 스키마 하위 시스템에 알릴 수 있습니다. (PSRegisterPropertySchema)
PSSetPropertyValue 속성 저장소의 속성 값을 설정합니다.
PSStringFromPropertyKey

해당 속성의 키에서 속성을 식별하는 문자열을 만듭니다.
PSUnregisterPropertySchema 속성 설명 스키마 파일의 제거를 스키마 하위 시스템에 알릴 수 있습니다.
WinRTPropertyValueToPropVariant Windows 런타임 속성 값의 콘텐츠를 PROPVARIANT 구조체로 복사합니다.

열거형

GETPROPERTYSTOREFLAGS IShellItem2::GetPropertyStore 또는 IPropertyStoreFactory::GetPropertyStore와 같이 속성 저장소를 만드는 메서드에 의해 검색된 속성 저장소 개체를 수정하는 플래그를 나타냅니다.
PKA_FLAGS 속성 변경 배열 동작에 대해 설명합니다.
PROPDESC_AGGREGATION_TYPE 여러 항목을 선택할 때 속성 값이 표시되는 방법을 설명합니다.
PROPDESC_COLUMNINDEX_TYPE 속성을 인덱싱할 수 있는지 여부 또는 방법을 나타냅니다.
PROPDESC_CONDITION_TYPE Windows Vista의 쿼리 작성기 UI에 속성을 표시할 때 사용할 조건 유형에 대해 설명하지만 Windows 7 이상에서는 표시되지 않습니다.
PROPDESC_ENUMFILTER 반환되는 속성 설명의 필터링된 목록을 설명합니다.
PROPDESC_FORMAT_FLAGS 속성 설명 도우미 함수(예: PSFormatForDisplay)에서 속성 문자열의 형식을 나타내는 데 사용됩니다.
PROPDESC_RELATIVEDESCRIPTION_TYPE

displayInfo 요소의 relativeDescriptionType 특성에 의해 결정되는 속성 설명에 대한 상대 설명 형식을 설명합니다.
PROPDESC_SEARCHINFO_FLAGS 속성이 Windows Search에서 인덱싱되는지 여부와 방법을 결정합니다.
PROPDESC_TYPE_FLAGS 속성의 .propdesc 파일에서 typeInfo 요소의 특성을 설명합니다.
PROPDESC_VIEW_FLAGS 이러한 플래그는 속성 설명 목록 문자열의 속성을 설명합니다.
PSC_STATE 속성의 상태를 지정합니다. 메모리 내 속성 저장소 캐시를 호스팅하는 코드에 의해 수동으로 설정됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

propvarutil.h 헤더

아티클2023. 03. 16.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 구조적 저장소
- Windows 속성 시스템

propvarutil.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

ClearPropVariantArray 배열에 저장된 PROPVARIANT 구조의 배열에 사용되는 메모리 및 참조를 해제합니다.
ClearVariantArray 배열에 저장된 VARIANT 구조의 배열에서 사용하는 메모리 및 참조를 해제합니다.
InitPropVariantFromBoolean 지정된 부울 값을 사용하여 지정된 PROPVARIANT 구조를 VT_BOOL 초기화합니다.
InitPropVariantFromBooleanVector 지정된 부울 벡터에서 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromBuffer 버퍼의 내용을 사용하여 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromCLSID CLSID(클래스 식별자)를 기반으로 PROPVARIANT 구조를 초기화합니다.

InitPropVariantFromDouble 지정된 double 값을 기반으로 PROPVARIANT 구조를 초기화합니다.	
InitPropVariantFromDoubleVector 지정된 double 값 벡터를 기반으로 PROPVARIANT 구조를 초기화합니다.	
InitPropVariantFromFileTime FILETIME 구조에 저장된 정보를 기반으로 PROPVARIANT 구조를 초기화합니다.	
InitPropVariantFromFileTimeVector FILETIME 값의 지정된 벡터에서 PROPVARIANT 구조를 초기화합니다.	
InitPropVariantFromGUIDAsBuffer GUID를 기반으로 PROPVARIANT 구조를 초기화합니다. 구조체가 VT_VECTOR	VT_UI1.
InitPropVariantFromGUIDAsString GUID를 기반으로 PROPVARIANT 구조를 초기화합니다. 구조체는 VT_LPWSTR 초기화됩니다.	
InitPropVariantFromInt16 16비트 정수 값을 기반으로 PROPVARIANT 구조를 초기화합니다.	
InitPropVariantFromInt16Vector 지정된 16비트 정수 값의 벡터를 기반으로 PROPVARIANT 구조를 초기화합니다.	
InitPropVariantFromInt32 32비트 정수 값을 기반으로 PROPVARIANT 구조를 초기화합니다.	
InitPropVariantFromInt32Vector 32비트 정수 값의 벡터를 기반으로 PROPVARIANT 구조를 초기화합니다.	

InitPropVariantFromInt64 지정된 Int64 값을 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromInt64Vector Int64 값의 벡터를 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromPropVariantVectorElem 지정된 PROPVARIANT 벡터 요소를 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromResource 실행 파일에 포함된 문자열 리소스를 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromString 지정된 문자열을 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromStringAsVector 지정된 문자열에서 PROPVARIANT 구조를 초기화합니다. 문자열은 세미콜론으로 구분된 목록으로 구문 분석됩니다(예::_"A; B; C").
InitPropVariantFromStringVector 지정된 문자열 벡터에서 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromStrRet STRRET 구조에 저장된 문자열을 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromUInt16 부호 없는 16비트 정수 값을 기반으로 PROPVARIANT 구조를 초기화합니다.

InitPropVariantFromUInt16Vector 부호 없는 16비트 정수 값의 벡터를 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromUInt32 부호 없는 32비트 정수 값을 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromUInt32Vector 부호 없는 32비트 정수 값의 벡터를 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromUInt64 부호 없는 64비트 정수 값을 사용하여 PROPVARIANT 구조를 초기화합니다.
InitPropVariantFromUInt64Vector 부호 없는 64비트 정수의 벡터를 기반으로 PROPVARIANT 구조를 초기화합니다.
InitPropVariantVectorFromPropVariant 다른 PROPVARIANT에 저장된 값을 사용하여 PROPVARIANT 구조에서 벡터 요소를 초기화합니다.
InitVariantFromBoolean 부울 값을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromBooleanArray 부울 값 배열에서 VARIANT 구조를 초기화합니다.
InitVariantFromBuffer 버퍼의 내용을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromDispatch IDispatch 개체의 인스턴스를 기반으로 VARIANT 구조를 초기화합니다.

InitVariantFromDosDateTime Microsoft MS-DOS에서 사용하는 형식으로 지정된 날짜 및 시간을 사용하여 VARIANT 구조를 초기화합니다. 날짜 및 시간 값은 VARIANT에 날짜 및 시간을 저장하는 데 사용되는 형식으로 변환됩니다.	
InitVariantFromDouble DOUBLE 형식의 값을 사용하여 VARIANT 구조를 초기화합니다.	
InitVariantFromDoubleArray DOUBLE 형식의 값 배열을 사용하여 VARIANT 구조를 초기화합니다.	
InitVariantFromFileTime FILETIME 구조체의 내용을 사용하여 VARIANT 구조를 초기화합니다.	
InitVariantFromFileTimeArray FILETIME 구조의 배열을 사용하여 VARIANT 구조를 초기화합니다.	
InitVariantFromGUIDAsBuffer GUID를 기반으로 VARIANT 구조를 초기화합니다. 구조체가 VT_ARRAY 초기화됩니다.	VT_UI1.
InitVariantFromGUIDAsString GUID를 기반으로 VARIANT 구조를 초기화합니다. 구조체는 VT_BSTR 형식으로 초기화됩니다.	
InitVariantFromInt16 16비트 정수 값을 사용하여 VARIANT 구조를 초기화합니다.	
InitVariantFromInt16Array 16비트 정수 값 배열을 사용하여 VARIANT 구조를 초기화합니다.	

InitVariantFromInt32 32비트 정수 값을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromInt32Array 32비트 정수 값 배열을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromInt64 64비트 정수 값을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromInt64Array 64비트 정수 값 배열을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromResource 실행 파일에 포함된 문자열 리소스를 기반으로 VARIANT 구조를 초기화합니다.
InitVariantFromString 문자열을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromStringArray 문자열 배열을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromStrRet STRRET 구조에 저장된 문자열을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromUInt16 부호 없는 16비트 정수 값을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromUInt16Array 부호 없는 16비트 정수 값 배열을 사용하여 VARIANT 구조를 초기화합니다.

InitVariantFromUInt32 부호 없는 32비트 정수 값을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromUInt32Array 부호 없는 32비트 정수 값 배열을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromUInt64 부호 없는 64비트 정수 값을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromUInt64Array 부호 없는 64비트 정수 값 배열을 사용하여 VARIANT 구조를 초기화합니다.
InitVariantFromVariantArrayElem 다른 VARIANT 구조에 저장된 값을 사용하여 VARIANT 구조를 초기화합니다.
IsPropVariantString 지정된 PROPVARIANT 구조체가 문자열 형식인지 여부를 지정합니다.
IsPropVariantVector PROPVARIANT 구조체에 벡터 형식이 있는지 여부를 지정합니다.
IsVariantArray 변형이 배열인지 여부를 지정합니다.
IsVariantString 변형이 문자열인지 여부를 지정합니다.
IsVarTypeFloat VARTYPE이 float인지 여부를 반환합니다.

IsVarTypeInteger VARTYPE이 정수인지 여부를 반환합니다.		
IsVarTypeNumber VARTYPE이 숫자인지 여부를 지정합니다.		
IsVarTypeSignedInteger VARTYPE이 부 서명된 정수인지 여부를 반환합니다.		
IsVarTypeUnsignedInteger VARTYPE이 부호 없는 정수인지 여부를 반환합니다.		
PropVariantChangeType PROPVARIANT 구조체로 저장된 값을 다른 변형 형식의 해당 값으로 강제 변환합니다.		
PropVariantCompare 기본 비교 단위 및 설정에 따라 두 PROPVARIANT 구조를 비교합니다.		
PropVariantCompareEx 호출자가 지정된 비교 단위 및 플래그에 따라 두 PROPVARIANT 구조를 비교할 수 있도록 하여 PropVariantCompare를 확장합니다.		
PropVariantGetBooleanElem VT_VECTOR VT_BOOL 형식의 PROPVARIANT 구조에서 단일 부울 요소를 추출합니다.	VT_BOOL 또는 VT_ARRAY	VT_BOOL.
PropVariantGetDoubleElem VT_R8 형식의 PROPVARIANT 구조에서 단일 double 요소를 추출합니다VT_VECTOR	VT_R8 또는 VT_ARRAY	VT_R8.
PropVariantGetElem 다른 PROPVARIANT 구조의 지정된 요소를 기반으로 PROPVARIANT 구조를 초기화합니다.		

PropVariantGetElementCount PROPVARIANT 구조체의 요소 수를 검색합니다.			
PropVariantGetFileTimeElem VT_FILETIME 형식의 PROPVARIANT 구조에서 단일 FILETIME 요소를 추출합니다VT_VECTOR	VT_FILETIME 또는 VT_ARRAY	VT_FILETIME.	
PropVariantGetInt16Elem VT_I2 형식의 PROPVARIANT 구조에서 단일 Int16 요소를 추출합니다VT_VECTOR	VT_I2 또는 VT_ARRAY	VT_I2.	
PropVariantGetInt32Elem VT_VECTOR VT_I4 형식의 PROPVARIANT에서 단일 Int32 요소를 추출합니다.	VT_I4 또는 VT_ARRAY	VT_I4.	
PropVariantGetInt64Elem VT_I8 형식의 PROPVARIANT 구조에서 단일 Int64 요소를 추출 VT_VECTOR합니다.	VT_I8 또는 VT_ARRAY	VT_I8.	
PropVariantGetStringElem VT_LPWSTR, VT_BSTR, VT_VECTOR 형식의 PROPVARIANT 구조에서 단일 유니코드 문자열 요소를 추출합니다.	VT_LPWSTR, VT_VECTOR	VT_BSTR 또는 VT_ARRAY	VT_BSTR.
PropVariantGetUInt16Elem VT_VECTOR VT_U12 형식의 PROPVARIANT 구조에서 부호 없는 단일 Int16 요소를 추출합니다.	VT_U12 또는 VT_ARRAY	VT_U12.	
PropVariantGetUInt32Elem VT_UI4 형식의 PROPVARIANT 구조에서 부호 없는 단일 Int32 요소를 추출합니다VT_VECTOR	VT_UI4 또는 VT_ARRAY	VT_UI4.	
PropVariantGetUInt64Elem VT_UI8 형식의 PROPVARIANT 구조에서 부호 없는 단일 Int64 요소를 추출합니다VT_VECTOR	VT_UI8 또는 VT_ARRAY	VT_UI8.	
PropVariantToBoolean PROPVARIANT 구조체의 부울 속성 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.			

PropVariantToBooleanVector PROPVARIANT 구조에서 부울 벡터를 추출합니다.		
PropVariantToBooleanVectorAlloc PROPVARIANT 구조에서 새로 할당된 부울 벡터로 데이터를 추출합니다.		
PropVariantToBooleanWithDefault PROPVARIANT 구조체의 부울 속성 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.		
PropVariantToBSTR PROPVARIANT 구조체의 BSTR 속성 값을 추출합니다.		
PropVariantToBuffer VT_VECTOR 형식의 PROPVARIANT 구조에서 버퍼 값을 추출합니다.	VT_UI1 또는 VT_ARRRAY	VT_UI1.
PropVariantToCLSID PROPVARIANT 구조체의 CLSID(클래스 식별자) 속성 값을 추출합니다.		
PropVariantToDouble PROPVARIANT 구조체에서 이중 값을 추출합니다.		
PropVariantToDoubleVector PROPVARIANT 구조에서 double의 벡터를 추출합니다.		
PropVariantToDoubleVectorAlloc PROPVARIANT 구조에서 새로 할당된 이중 벡터로 데이터를 추출합니다.		
PropVariantToDoubleWithDefault PROPVARIANT 구조체의 이중 속성 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.		

PropVariantToFileTime	PROPVARIANT 구조체에서 FILETIME 구조를 추출합니다.
PropVariantToFileTimeVector	PROPVARIANT 구조에서 FILETIME 벡터로 데이터를 추출합니다.
PropVariantToFileTimeVectorAlloc	PROPVARIANT 구조에서 새로 할당된 FILETIME 벡터로 데이터를 추출합니다.
PropVariantToGUID	PROPVARIANT 구조체에서 GUID 값을 추출합니다.
PropVariantToInt16	PROPVARIANT 구조체의 Int16 속성 값을 추출합니다.
PropVariantToInt16Vector	PROPVARIANT 구조에서 Int16 값의 벡터를 추출합니다.
PropVariantToInt16VectorAlloc	PROPVARIANT 구조에서 새로 할당된 Int16 벡터로 데이터를 추출합니다.
PropVariantToInt16WithDefault	PROPVARIANT 구조체의 Int16 속성 값을 추출합니다. 현재 값이 없으면 지정된 기본값이 반환됩니다.
PropVariantToInt32	PROPVARIANT 구조체의 Int32 속성 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
PropVariantToInt32Vector	PROPVARIANT 구조체에서 long 값의 벡터를 추출합니다.

PropVariantToInt32VectorAlloc PROPVARIANT 구조에서 새로 할당된 Int32 벡터로 데이터를 추출합니다.
PropVariantToInt32WithDefault PROPVARIANT 구조체에서 Int32 값을 추출합니다. 현재 값이 없으면 지정된 기본값이 반환됩니다.
PropVariantToInt64 PROPVARIANT 구조체에서 LONGLONG 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
PropVariantToInt64Vector PROPVARIANT 구조에서 Int64 벡터로 데이터를 추출합니다.
PropVariantToInt64VectorAlloc PROPVARIANT 구조에서 새로 할당된 LONGLONG 벡터로 데이터를 추출합니다.
PropVariantToInt64WithDefault PROPVARIANT 구조체의 Int64 속성 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.
PropVariantToString PROPVARIANT 구조체에서 문자열 값을 추출합니다.
PropVariantToStringAlloc PROPVARIANT 구조체에서 문자열 속성 값을 추출합니다.
PropVariantToStringVector PROPVARIANT 구조체에서 문자열 벡터를 추출합니다.
PropVariantToStringVectorAlloc PROPVARIANT 구조에서 새로 할당된 벡터의 새로 할당된 문자열로 데이터를 추출합니다.

PropVariantToStringWithDefault PROPVARIANT 구조체의 문자열 속성 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.
PropVariantToStrRet PROPVARIANT 구조체에서 문자열을 추출하여 STRRET 구조에 배치합니다.
PropVariantToUInt16 PROPVARIANT 구조체에서 부호 없는 짧은 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
PropVariantToUInt16Vector PROPVARIANT 구조체에서 부호 없는 짧은 벡터로 데이터를 추출합니다.
PropVariantToUInt16VectorAlloc PROPVARIANT 구조에서 새로 할당된 부호 없는 짧은 벡터로 데이터를 추출합니다.
PropVariantToUInt16WithDefault PROPVARIANT 구조체에서 부호 없는 짧은 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.
PropVariantToUInt32 PROPVARIANT 구조체에서 ULONG 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
PropVariantToUInt32Vector PROPVARIANT 구조에서 ULONG 벡터로 데이터를 추출합니다.
PropVariantToUInt32VectorAlloc PROPVARIANT 구조에서 새로 할당된 ULONG 벡터로 데이터를 추출합니다.
PropVariantToUInt32WithDefault PROPVARIANT 구조체에서 ULONG 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.

PropVariantToUInt64 PROPVARIANT 구조체에서 UInt64 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
PropVariantToUInt64Vector PROPVARIANT 구조에서 ULONGLONG 벡터로 데이터를 추출합니다.
PropVariantToUInt64VectorAlloc PROPVARIANT 구조에서 새로 할당된 ULONGLONG 벡터로 데이터를 추출합니다.
PropVariantToUInt64WithDefault PROPVARIANT 구조에서 ULONGLONG 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.
PropVariantToVariant PROPVARIANT 구조체의 내용을 VARIANT 구조체로 변환합니다.
StgDeserializePropVariant StgDeserializePropVariant 함수는 SERIALIZEDPROPERTYVALUE 데이터 형식을 PROPVARIANT 데이터 형식으로 변환합니다.
StgSerializePropVariant StgSerializePropVariant 함수는 PROPVARIANT 데이터 형식을 SERIALIZEDPROPERTYVALUE 데이터 형식으로 변환합니다.
VariantCompare 기본 비교 규칙에 따라 두 가지 변형 구조를 비교합니다.
VariantGetBooleanElem 변형 구조체에서 단일 부울 요소를 추출합니다.
VariantGetDoubleElem 변형 구조체에서 하나의 double 요소를 추출합니다.

VariantGetElem 지정된 variant 요소에서 VARIANT 구조를 초기화합니다.
VariantGetElementCount 변형 구조체의 요소 수를 검색합니다.
VariantGetInt16Elem 변형 구조체에서 단일 Int16 요소를 추출합니다.
VariantGetInt32Elem 변형 구조체에서 단일 Int32 요소를 추출합니다.
VariantGetInt64Elem 변형 구조체에서 단일 Int64 요소를 추출합니다.
VariantGetStringElem 변형 구조체에서 단일 와이드 문자열 요소를 추출합니다.
VariantGetUInt16Elem 변형 구조체에서 서명되지 않은 단일 Int16 요소를 추출합니다.
VariantGetUInt32Elem 변형 구조체에서 서명되지 않은 단일 Int32 요소를 추출합니다.
VariantGetUInt64Elem 변형 구조체에서 서명되지 않은 단일 Int64 요소를 추출합니다.
VariantToBoolean VARIANT 구조체에서 부울 속성의 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
VariantToBooleanArray VARIANT 구조체에서 부울 값 배열을 추출합니다.

VariantToBooleanArrayAlloc BOOL 값의 배열을 할당한 다음 VARIANT 구조에서 해당 배열로 데이터를 추출합니다.	
VariantToBooleanWithDefault VARIANT 구조체에서 BOOL 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.	
VariantToBuffer VT_ARRRAY 형식의 VARIANT 구조에 저장된 버퍼의 내용을 추출합니다.	VT_UI1.
VariantToDosDateTime VARIANT 구조에서 Microsoft MS-DOS 형식의 날짜 및 시간 값을 추출합니다.	
VariantToDouble VARIANT 구조체에서 DOUBLE 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.	
VariantToDoubleArray VARIANT 구조에서 DOUBLE 값 배열을 추출합니다.	
VariantToDoubleArrayAlloc DOUBLE 값 배열을 할당한 다음 VARIANT 구조에서 해당 배열로 데이터를 추출합니다.	
VariantToDoubleWithDefault VARIANT 구조체에서 DOUBLE 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.	
VariantToFileTime 변형 구조체에서 FILETIME 구조를 추출합니다.	
VariantToGUID 변형 구조체의 GUID 속성 값을 추출합니다.	

VariantToInt16	변형 구조체의 Int16 속성 값을 추출합니다. 값을 추출할 수 없는 경우 이 함수에서 기본값을 할당합니다.
VariantToInt16Array	벡터 구조에서 Int16 배열로 데이터를 추출합니다.
VariantToInt16ArrayAlloc	벡터 구조에서 새로 할당된 Int16 배열로 데이터를 추출합니다.
VariantToInt16WithDefault	변형 구조체의 Int16 속성 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.
VariantToInt32	변형 구조체의 Int32 속성 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
VariantToInt32Array	벡터 구조에서 Int32 배열로 데이터를 추출합니다.
VariantToInt32ArrayAlloc	벡터 구조에서 새로 할당된 Int32 배열로 데이터를 추출합니다.
VariantToInt32WithDefault	변형 구조체의 Int32 속성 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.
VariantToInt64	변형 구조체의 Int64 속성 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
VariantToInt64Array	벡터 구조에서 Int64 배열로 데이터를 추출합니다.

VariantToInt64ArrayAlloc	벡터 구조에서 새로 할당된 Int64 배열로 데이터를 추출합니다.
VariantToInt64WithDefault	변형 구조체의 Int64 속성 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.
VariantToPropVariant	VARIANT 구조체의 내용을 PROPVARIANT 구조체에 복사합니다.
VariantToString	변형 구조체의 변형 값을 문자열로 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
VariantToStringAlloc	변형 구조체의 변형 값을 새로 할당된 문자열로 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
VariantToStringArray	벡터 구조에서 문자열 배열로 데이터를 추출합니다.
VariantToStringArrayAlloc	벡터 구조에서 새로 할당된 문자열 배열로 데이터를 추출합니다.
VariantToStringWithDefault	변형 구조체의 문자열 속성 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.
VariantToStrRet	원본 변형이 VT_BSTR 경우 문자열을 추출하여 STRRET 구조에 배치합니다.
VariantToUInt16	변형 구조체의 부호 없는 Int16 속성 값을 추출합니다. 값을 추출할 수 없는 경우 이 함수에서 기본값을 할당합니다.

VariantToUInt16Array 벡터 구조에서 서명되지 않은 Int16 배열로 데이터를 추출합니다.
VariantToUInt16ArrayAlloc 벡터 구조에서 새로 할당된 부호 없는 Int16 배열로 데이터를 추출합니다.
VariantToUInt16WithDefault 변형 구조체의 부호 없는 Int16 속성 값을 추출합니다. 값이 없으면 지정된 기본값이 반환됩니다.
VariantToUInt32 변형 구조체의 부호 없는 Int32 속성 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
VariantToUInt32Array 벡터 구조에서 부호 없는 Int32 배열로 데이터를 추출합니다.
VariantToUInt32ArrayAlloc 벡터 구조에서 새로 할당된 부호 없는 Int32 배열로 데이터를 추출합니다.
VariantToUInt32WithDefault 변형 구조체의 부호 없는 Int32 속성 값을 추출합니다. 현재 값이 없으면 지정된 기본값이 반환됩니다.
VariantToUInt64 변형 구조체의 부호 없는 Int64 속성 값을 추출합니다. 값을 추출할 수 없는 경우 기본값이 할당됩니다.
VariantToUInt64Array 벡터 구조에서 서명되지 않은 Int64 배열로 데이터를 추출합니다.
VariantToUInt64ArrayAlloc 벡터 구조에서 새로 할당된 부호 없는 Int64 배열로 데이터를 추출합니다.

VariantToUInt64WithDefault 변형 구조체의 부호 없는 Int64 속성 값을 추출합니다. 현재 값이 없으면 지정된 기본값이 반환됩니다.

열거형

PROPVAR_COMPARE_UNIT 이러한 플래그는 특정 PROPVARIANT 구조 비교와 연결됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

provider.h 헤더

아티클2023. 08. 25.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 레거시 Windows 환경 기능
- Windows Management Instrumentation

provider.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

공급자 Provider 클래스는 프레임워크 공급자가 지원하는 클래스 또는 클래스의 기본 클래스입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

prsht.h 헤더

아티클2023. 08. 25.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft 관리 콘솔 2.0
- Windows 컨트롤

prsht.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CreatePropertySheetPageA 속성 시트에 대한 새 페이지를 만듭니다. (ANSI)
CreatePropertySheetPageW 속성 시트에 대한 새 페이지를 만듭니다. (유니코드)
DestroyPropertySheetPage 속성 시트 페이지를 삭제합니다. 애플리케이션에서 PropertySheet 함수에 전달되지 않은 페이지에 대해 이 함수를 호출해야 합니다.
PropertySheetA 속성 시트를 만들고 지정된 속성 시트 헤더 구조체에 정의된 페이지를 추가합니다. (ANSI)
PropertySheetW 속성 시트를 만들고 지정된 속성 시트 헤더 구조체에 정의된 페이지를 추가합니다. (유니코드)
PropSheet_AddPage 기존 속성 시트의 끝에 새 페이지를 추가합니다. 이 매크로를 사용하거나 PSM_ADDPAGE 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_Apply 하나 이상의 페이지가 변경되었으며 변경 내용의 유효성을 검사하고 기록해야 함을 나타내는 적용 단추의 선택을 시뮬레이션합니다. 이 매크로를 사용하거나 PSM_APPLY 메시지를 명시적으로 보낼 수 있습니다.

PropSheet_CancelToClose 가장 최근의 PSN_APPLY 알림을 취소할 수 없으므로 변경한 경우에 사용됩니다. PSM_CANCELTOCLOSE 메시지를 명시적으로 보낼 수도 있습니다.
PropSheet_Changed 페이지의 정보가 변경되었음을 속성 시트에 알립니다. 이 매크로를 사용하거나 PSM_CHANGED 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_EnableWizButtons Aero 마법사에서 단추를 사용하거나 사용하지 않도록 설정합니다. 이 매크로를 사용하거나 PSM_ENABLEWIZBUTTONS 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_GetCurrentPageHwnd 속성 시트의 현재 페이지 창에 대한 핸들을 검색합니다. 이 매크로를 사용하거나 PSM_GETCURRENTPAGEHWND 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_GetResult PropertySheet에 의해 모달 속성 시트에 반환된 정보를 검색하기 위해 모덜리스 속성 시트에서 사용됩니다. 이 매크로를 사용하거나 PSM_GETRESULT 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_GetTabControl 속성 시트의 탭 컨트롤에 대한 핸들을 검색합니다. 이 매크로를 사용하거나 PSM_GETTABCONTROL 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_HwndToIndex 속성 시트 페이지의 창 핸들을 가져와서 0부터 시작하는 인덱스로 반환합니다. 이 매크로를 사용하거나 PSM_HWNDTOINDEX 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_IdToIndex 속성 시트 페이지의 리소스 식별자(ID)를 가져와서 0부터 시작하는 인덱스로 반환합니다. 이 매크로를 사용하거나 PSM_IDTOINDEX 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_IndexToHwnd 속성 시트 페이지의 인덱스를 가져와서 해당 창 핸들을 반환합니다. 이 매크로를 사용하거나 PSM_INDEXTOHWND 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_IndexToId 속성 시트 페이지의 인덱스 를 가져와서 해당 리소스 식별자(ID)를 반환합니다. 이 매크로를 사용하거나 PSM_INDEXTOID 메시지를 명시적으로 보낼 수 있습니다.

PropSheet_IndexToPage 속성 시트 페이지의 인덱스를 가져오고 해당 HPROPSHEETPAGE 핸들을 반환합니다. 이 매크로를 사용하거나 PSM_INDEXTOPAGE 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_InsertPage 기존 속성 시트에 새 페이지를 삽입합니다. 페이지는 지정된 인덱스 또는 지정된 페이지 뒤에 삽입할 수 있습니다. 이 매크로를 사용하거나 PSM_INSERTPAGE 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_IsDialogMessage 속성 시트 대화 상자에 메시지를 전달하고 대화 상자에서 메시지를 처리했는지 여부를 나타냅니다. 이 매크로를 사용하거나 PSM_ISDIALOGMESSAGE 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_PageToIndex 속성 시트 페이지의 HPROPSHEETPAGE 핸들을 가져와서 0부터 시작하는 인덱스로 반환합니다. 이 매크로를 사용하거나 PSM_PAGETOINDEX 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_PressButton 속성 시트 단추의 선택을 시뮬레이션합니다. 이 매크로를 사용하거나 PSM_PRESSBUTTON 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_QuerySiblings 속성 시트가 각 페이지에 PSM_QUERYSIBLINGS 메시지를 보내도록 합니다. 이 매크로를 사용하거나 PSM_QUERYSIBLINGS 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_RebootSystem 변경 내용을 적용하려면 시스템을 다시 시작해야 함을 나타냅니다. 이 매크로를 사용하거나 PSM_REBOOTSYSTEM 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_RecalcPageSizes 페이지가 추가되거나 제거된 후 표준 또는 마법사 속성 시트의 페이지 크기를 다시 계산합니다. 이 매크로를 사용하거나 PSM_RECALCPAGESIZES 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_RemovePage 속성 시트에서 페이지를 제거합니다. 이 매크로를 사용하거나 PSM_REMOVEPAGE 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_RestartWindows 변경 내용을 적용하려면 Windows를 다시 시작해야 함을 나타내는 PSM_RESTARTWINDOWS 메시

지를 보냅니다. 이 매크로를 사용하거나 PSM_RESTARTWINDOWS 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_SetButtonText Aero 마법사에서 단추의 텍스트를 설정합니다. 이 매크로를 사용하거나 PSM_SETBUTTONTEXT 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_SetCurSel 속성 시트에서 지정된 페이지를 활성화합니다. 이 매크로를 사용하거나 PSM_SETCURSEL 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_SetCurSelByID 페이지의 리소스 식별자를 기반으로 속성 시트에서 지정된 페이지를 활성화합니다. 이 매크로를 사용하거나 PSM_SETCURSELID 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_SetFinishText 마법사에서 마침 단추의 텍스트를 설정하고, 이 단추를 표시하고 사용하도록 설정하며, 다음 및 뒤로 단추를 숨깁니다. 이 매크로를 사용하거나 PSM_SETFINISHTEXT 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_SetHeaderSubTitle 마법사 내부 페이지의 헤더에 대한 부제 텍스트를 설정합니다. 이 매크로를 사용하거나 PSM_SETHEADERSUBTITLE 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_SetHeaderTitle 마법사 내부 페이지의 헤더에 대한 제목 텍스트를 설정합니다. 이 매크로를 사용하거나 PSM_SETHEADERTITLE 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_SetNextText 마법사에서 다음 단추의 텍스트를 설정합니다. 이 매크로를 사용하거나 PSM_SETNEXTTEXT 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_SetTitle 속성 시트의 제목을 설정합니다. 이 매크로를 사용하거나 PSM_SETTITLE 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_SetWizButtons PSM_SETWIZBUTTONS 메시지를 게시하여 마법사에서 뒤로, 다음 및 마침 단추를 사용하거나 사용하지 않도록 설정합니다. 이 매크로를 사용하거나 PSM_SETWIZBUTTONS 메시지를 명시적으로 보낼 수 있습니다.

PropSheet_ShowWizButtons 마법사에서 단추를 표시하거나 숨깁니다. 이 매크로를 사용하거나 PSM_SHOWWIZBUTTONS 메시지를 명시적으로 보낼 수 있습니다.
PropSheet_UnChanged 페이지의 정보가 이전에 저장된 상태로 되돌아갔음을 속성 시트에 알립니다. 이 매크로를 사용하거나 PSM_UNCHANGED 메시지를 명시적으로 보낼 수 있습니다.

콜백 함수

LPFNADDPROPSHEETPAGE 속성 시트 확장에서 속성 시트에 페이지를 추가하는 데 사용하는 애플리케이션 정의 콜백 함수를 지정합니다.
LPFNPSPCALLBACKA 페이지를 만들 때와 페이지를 제거하려고 할 때 속성 시트에서 호출하는 애플리케이션 정의 콜백 함수를 지정합니다. 애플리케이션은 이 함수를 사용하여 페이지에 대한 초기화 및 정리 작업을 수행할 수 있습니다. (ANSI)
LPFNPSPCALLBACKW 페이지를 만들 때와 페이지를 제거하려고 할 때 속성 시트에서 호출하는 애플리케이션 정의 콜백 함수를 지정합니다. 애플리케이션은 이 함수를 사용하여 페이지에 대한 초기화 및 정리 작업을 수행할 수 있습니다. (유니코드)
PFNPROPSHEETCALLBACK 속성 시트를 만들고 초기화할 때 시스템에서 호출하는 애플리케이션 정의 콜백 함수입니다.

구조체

PROPSHEETHEADERA_V2 ANSI(PROPSHEETHEADERA_V2) 구조체는 속성 시트의 프레임과 페이지를 정의합니다.
PROPSHEETHEADERW_V2

PROPSHEETHEADERW_V2(유니코드) 구조체는 속성 시트의 프레임과 페이지를 정의합니다.
PROPSHEETPAGEA_V2 ANSI(PROPSHEETPAGEA_V2) 구조체는 속성 시트의 페이지를 정의합니다.
PROPSHEETPAGEA_V3 ANSI(PROPSHEETPAGEA_V3) 구조체는 속성 시트의 페이지를 정의합니다.
PROPSHEETPAGEA_V4 ANSI(PROPSHEETPAGEA_V4) 구조체는 속성 시트의 페이지를 정의합니다.
PROPSHEETPAGEW_V2 PROPSHEETPAGEW_V2(유니코드) 구조체는 속성 시트의 페이지를 정의합니다.
PROPSHEETPAGEW_V3 PROPSHEETPAGEW_V3(유니코드) 구조체는 속성 시트의 페이지를 정의합니다.
PROPSHEETPAGEW_V4 PROPSHEETPAGEW_V4(유니코드) 구조체는 속성 시트의 페이지를 정의합니다.
PSHNOTIFY 속성 시트 알림 메시지에 대한 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

psapi.h 헤더

아티클2024. 01. 11.

이 헤더는 PSAPI(프로세스 상태 API)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 프로세스 상태 API(PSAPI)

psapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

EmptyWorkingSet 지정된 프로세스의 작업 집합에서 가능한 한 많은 페이지를 제거합니다.
EnumDeviceDrivers 시스템의 각 디바이스 드라이버에 대한 부하 주소를 검색합니다.
EnumPageFilesA 시스템에 설치된 각 페이지 파일에 대한 콜백 루틴을 호출합니다. (ANSI)
EnumPageFilesW 시스템에 설치된 각 페이지 파일에 대한 콜백 루틴을 호출합니다. (유니코드)
EnumProcesses 시스템의 각 프로세스 개체에 대한 프로세스 식별자를 검색합니다.
EnumProcessModules 지정된 프로세스의 각 모듈에 대한 핸들을 검색합니다.
EnumProcessModulesEx 지정된 필터 조건을 충족하는 지정된 프로세스의 각 모듈에 대한 핸들을 검색합니다.
GetDeviceDriverBaseNameA 지정된 디바이스 드라이버의 기본 이름을 검색합니다. (ANSI)

GetDeviceDriverBaseNameW 지정된 디바이스 드라이버의 기본 이름을 검색합니다. (유니코드)
GetDeviceDriverFileNameA 지정된 디바이스 드라이버에 사용할 수 있는 경로를 검색합니다. (ANSI)
GetDeviceDriverFileNameW 지정된 디바이스 드라이버에 사용할 수 있는 경로를 검색합니다. (유니코드)
GetMappedFileNameA 지정된 주소가 지정된 프로세스의 주소 공간에 있는 메모리 매핑된 파일 내에 있는지 확인합니다. 이 경우 함수는 메모리 매핑 파일의 이름을 반환합니다. (ANSI)
GetMappedFileNameW 지정된 주소가 지정된 프로세스의 주소 공간에 있는 메모리 매핑된 파일 내에 있는지 확인합니다. 이 경우 함수는 메모리 매핑 파일의 이름을 반환합니다. (유니코드)
GetModuleBaseNameA 지정된 모듈의 기본 이름을 검색합니다. (ANSI)
GetModuleBaseNameW 지정된 모듈의 기본 이름을 검색합니다. (유니코드)
GetModuleFileNameExA 지정된 모듈을 포함하는 파일의 정규화된 경로를 검색합니다. (ANSI)
GetModuleFileNameExW 지정된 모듈을 포함하는 파일의 정규화된 경로를 검색합니다. (유니코드)
GetModuleInformation MODULEINFO 구조에서 지정된 모듈에 대한 정보를 검색합니다.
GetPerformanceInfo PERFORMANCE_INFORMATION 구조에 포함된 성능 값을 검색합니다.
GetProcessImageFileNameA 지정된 프로세스에 대한 실행 파일의 이름을 검색합니다. (ANSI)

GetProcessImageFileNameW 지정된 프로세스에 대한 실행 파일의 이름을 검색합니다. (유니코드)
GetProcessMemoryInfo 지정된 프로세스의 메모리 사용량에 대한 정보를 검색합니다.
GetWsChanges 이 함수 또는 InitializeProcessForWsWatch 함수가 마지막으로 호출된 이후 지정된 프로세스의 작업 집합에 추가된 페이지에 대한 정보를 검색합니다.
GetWsChangesEx 이 함수 또는 InitializeProcessForWsWatch 함수가 마지막으로 호출된 이후 지정된 프로세스의 작업 집합에 추가된 페이지에 대한 확장 정보를 검색합니다.
InitializeProcessForWsWatch 지정된 프로세스의 작업 집합에 대한 모니터링을 시작합니다.
QueryWorkingSet 지정된 프로세스의 작업 집합에 현재 추가된 페이지에 대한 정보를 검색합니다.
QueryWorkingSetEx 지정된 프로세스의 주소 공간에 있는 특정 가상 주소의 페이지에 대한 확장 정보를 검색합니다.

콜백 함수

테이블 확장

PENUM_PAGE_FILE_CALLBACKA EnumPageFiles 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (ANSI)
PENUM_PAGE_FILE_CALLBACKW EnumPageFiles 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (유니코드)

구조체

테이블 확장

ENUM_PAGE_FILE_INFORMATION 페이지 파일에 대한 정보를 포함합니다.
MODULEINFO 모듈 로드 주소, 크기 및 진입점을 포함합니다.
PERFORMANCE_INFORMATION 성능 정보를 포함합니다. (PERFORMANCE_INFORMATION)
PROCESS_MEMORY_COUNTERS 프로세스에 대한 메모리 통계를 포함합니다.
PROCESS_MEMORY_COUNTERS_EX 프로세스에 대한 확장 메모리 통계를 포함합니다.
PROCESS_MEMORY_COUNTERS_EX2 프로세스에 대한 확장 메모리 통계를 포함합니다. PROCESS_MEMORY_COUNTERS_EX 확장합니다.
PSAPI_WORKING_SET_BLOCK 페이지에 대한 작업 집합 정보를 포함합니다.
PSAPI_WORKING_SET_EX_BLOCK 페이지에 대한 확장 작업 집합 정보를 포함합니다.
PSAPI_WORKING_SET_EX_INFORMATION 프로세스에 대한 확장 작업 집합 정보를 포함합니다.
PSAPI_WORKING_SET_INFORMATION 프로세스에 대한 작업 집합 정보를 포함합니다.
PSAPI_WS_WATCH_INFORMATION 프로세스 작업 집합에 추가된 페이지에 대한 정보를 포함합니다.
PSAPI_WS_WATCH_INFORMATION_EX 프로세스 작업 집합에 추가된 페이지에 대한 확장된 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

pwm.h 헤더

아티클2023. 08. 25.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

pwm.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

IOCTLs

IOCTL_PWM_CONTROLLER_GET_ACTUAL_PERIOD PWM(Pulse Width Modulation) 컨트롤러의 유효 출력 신호 기간을 출력 채널에서 측정할 때 검색합니다.
IOCTL_PWM_CONTROLLER_GET_INFO PWM(Pulse Width Modulation) 컨트롤러에 대한 정보를 검색합니다. 컨트롤러가 초기화된 후에는 이 정보가 변경되지 않습니다.
IOCTL_PWM_CONTROLLER_SET_DESIRED_PERIOD PWM(Pulse Width Modulation) 컨트롤러의 출력 신호 기간을 제안된 값으로 설정합니다.
IOCTL_PWM_PIN_GET_ACTIVE_DUTY_CYCLE_PERCENTAGE 핀 또는 채널의 현재 의무 주기 백분율을 검색합니다. 컨트롤 코드는 백분율을 PWM_PIN_GET_ACTIVE_DUTY_CYCLE_PERCENTAGE_OUTPUT 구조체로 반환합니다.
IOCTL_PWM_PIN_GET_POLARITY 핀 또는 채널의 현재 신호 극성을 검색합니다. 컨트롤 코드는 신호 극성을 PWM_PIN_GET_POLARITY_OUTPUT 구조체로 가져옵니다. 신호 극성은 PWM_POLARITY 열거형에 정의된 대로 활성 높음 또는 활성 낮음입니다.
IOCTL_PWM_PIN_IS_STARTED 핀 또는 채널의 신호 생성 상태를 검색합니다. 각 핀의 상태는 PWM_PIN_IS_STARTED_OUTPUT 구조체로 시작되거나 중지됩니다.
IOCTL_PWM_PIN_SET_ACTIVE_DUTY_CYCLE_PERCENTAGE 컨트롤러 핀 또는 채널에 대해 원하는 의무 주기 백분율 값을 설정합니다. 컨트롤 코드는 백분율을 PWM_PIN_SET_ACTIVE_DUTY_CYCLE_PERCENTAGE_INPUT 구조체로 지정합니다.

IOCTL_PWM_PIN_SET_POLARITY 핀 또는 채널의 신호 극성을 설정합니다. 제어 코드는 PWM_PIN_SET_POLARITY_INPUT 구조에 따라 신호 극성을 설정합니다. 신호 극성은 PWM_POLARITY 열거형에 정의된 대로 활성 높음 또는 활성 낮음입니다.
IOCTL_PWM_PIN_START 핀 또는 채널에서 PWM(Pulse Width Modulation) 신호 생성을 시작합니다. 핀이 시작되었는지 여부를 검사 IOCTL_PWM_PIN_IS_STARTED 사용합니다.
IOCTL_PWM_PIN_STOP 핀 또는 채널에서 PWM(Pulse Width Modulation) 신호 생성을 중지합니다. 핀이 시작되었는지 여부를 검사 IOCTL_PWM_PIN_IS_STARTED 사용합니다.

구조체

PWM_CONTROLLER_INFO PWM(Pulse Width Modulation) 컨트롤러의 특징을 나타내는 정적 정보를 나타냅니다.
PWM_CONTROLLER_SET_DESIRED_PERIOD_INPUT PWM(Pulse Width Modulation) 컨트롤러에 대해 제안된 신호 기간에 대한 입력 값을 포함합니다.
PWM_CONTROLLER_SET_DESIRED_PERIOD_OUTPUT PWM(Pulse Width Modulation) 컨트롤러의 유효 출력 신호 기간을 포함합니다.
PWM_PIN_SET_POLARITY_INPUT 핀 또는 채널의 극성에 대해 원하는 값을 포함합니다.

열거형

PWM_POLARITY 의무 주기와 관련하여 PWM(Pulse Width Modulation) 신호 수준 극성을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

qmgr.h 헤더

아티클2023. 08. 25.

이 헤더는 Background Intelligent Transfer Service에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Background Intelligent Transfer Service

qmgr.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IBackgroundCopyCallback1 이벤트가 발생할 때 알림을 수신하도록 IBackgroundCopyCallback1 인터페이스를 구현합니다.
IBackgroundCopyGroup IBackgroundCopyGroup 인터페이스를 사용하여 그룹을 관리합니다. 그룹에 다운로드 작업이 포함됩니다. 예를 들어 그룹에 작업을 추가하고, 그룹의 속성을 설정하고, 다운로드 큐에서 그룹을 시작 및 중지합니다.
IBackgroundCopyJob1 IBackgroundCopyJob1 인터페이스를 사용하여 작업에 파일을 추가하고 작업의 상태 검색합니다.
IBackgroundCopyQMgr IBackgroundCopyQMgr 인터페이스를 사용하여 새 그룹을 만들거나, 기존 그룹을 검색하거나, 큐의 모든 그룹을 열거합니다. 그룹에 다운로드 작업이 포함되어 있습니다.
IEnumBackgroundCopyGroups IEnumBackgroundCopyGroups 인터페이스를 사용하여 다운로드 큐의 그룹 목록을 열거합니다. IEnumBackgroundCopyGroups 인터페이스 포인터를 얻으려면 IBackgroundCopyQMgr::EnumGroups 메서드를 호출합니다.
IEnumBackgroundCopyJobs1 IEnumBackgroundCopyJobs1 인터페이스를 사용하여 그룹의 작업 목록을 열거합니다. IEnumBackgroundCopyJobs1 인터페이스 포인터를 얻으려면 IBackgroundCopyGroup::EnumJobs 메서드를 호출합니다.

구조체

FILESETINFO FILESETINFO 구조체는 다운로드할 파일의 원격 및 로컬 이름을 식별합니다.

열거형

GROUPPROP GROUPPROP 열거형은 그룹 속성 값을 검색하고 설정하기 위한 상수 값을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

qos.h 헤더

아티클2023. 08. 25.

이 헤더는 QOS(서비스 품질)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 서비스 품질(QOS)

qos.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

FLOWSPEC FLOWSPEC 구조는 RSVP SP에 서비스 매개 변수의 품질을 제공합니다.
QOS_OBJECT_HDR QOS_OBJECT_HDR QOS 개체는 각 QOS 개체에 연결됩니다. 개체 형식과 길이를 지정합니다.
QOS_SD_MODE QOS 개체 QOS_SD_MODE 트래픽 제어 패킷 쉐이퍼 구성 요소의 동작을 정의합니다.
QOS_SHAPING_RATE QOS 개체 QOS_SHAPING_RATE 지정된 흐름에 균일한 트래픽 셰이핑 속도를 적용하도록 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

qos2.h 헤더

아티클2023. 03. 08.

이 헤더는 QOS(서비스 품질)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 서비스 품질(QOS)

qos2.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

QOSAddSocketToFlow 트래픽에 대한 새 흐름을 추가합니다.
QOSCancel QOSSetFlow와 같이 보류 중인 겹치는 작업을 취소합니다.
QOSCloseHandle QOSCloseHandle 함수는 QOSCreateHandle 함수에서 반환하는 핸들을 닫습니다.
QOSCreateHandle 이 함수는 QOS 하위 시스템 및 QOSHandle 매개 변수를 초기화합니다. QOSHandle 매개 변수는 다른 QOS 함수를 호출할 때 사용됩니다. QOSCreateHandle은 다른 함수 앞에 호출되어야 합니다.
QOSEnumerateFlows 모든 기존 흐름을 열거합니다.
QOSNotifyFlow 알림을 받을 호출 애플리케이션을 등록합니다.
QOSQueryFlow 특정 흐름에 대한 정보를 요청합니다.
QOSRemoveSocketFromFlow QOS 하위 시스템에 이전에 추가된 흐름이 종료되었음을 알 수 있습니다.

QOSSetFlow 애플리케이션에서 애플리케이션의 패킷의 우선 순위를 지정하고 흐름 트래픽을 변경하도록 QOS 하위 시스템을 요청하기 위해 호출됩니다.
QOSStartTrackingClient QOSStartTrackingClient 함수는 QOS 하위 시스템에 새 클라이언트가 있음을 알 수 있습니다.
QOSStopTrackingClient QOSStopTrackingClient 함수는 QoS 하위 시스템에 이전에 QOSStartTrackingClient 함수를 사용한 클라이언트 추적을 중지하도록 알 수 있습니다. 흐름이 현재 진행 중인 경우 이 함수는 흐름에 영향을 미치지 않습니다.

구조체

QOS_FLOW_FUNDAMENTALS QOS_FLOW_FUNDAMENTALS 구조에는 흐름에 대한 기본 정보가 포함되어 있습니다.
QOS_FLOWRATE_OUTGOING QOS_FLOWRATE_OUTGOING 구조는 QOSSetFlow 함수에서 흐름 속도 정보를 설정하는 데 사용됩니다.
QOS_PACKET_PRIORITY 흐름 트래픽의 우선 순위를 나타내는 QOS_PACKET_PRIORITY 구조체입니다.
QOS_VERSION QOS_VERSION 구조는 QOS 프로토콜의 버전을 나타냅니다.

열거형

QOS_FLOWRATE_REASON QOS_FLOWRATE_REASON 열거형은 흐름의 대역폭이 변경된 이유를 나타냅니다.

QOS_NOTIFY_FLOW QOS_NOTIFY_FLOW 열거형은 QOSNotifyFlow 함수가 알림을 보내기 위해 존재해야 하는 상황을 지정합니다.
QOS_QUERY_FLOW QOS_QUERY_FLOW 열거형은 QOSQueryFlow 함수가 요청할 정보의 유형을 나타냅니다.
QOS_SET_FLOW QOS_SET_FLOW 열거형은 흐름에 대해 변경되는 내용을 나타냅니다.
QOS_SHAPING QOS_SHAPING 열거형은 흐름의 셰이핑 동작을 정의합니다.
QOS_TRAFFIC_TYPE QOS_TRAFFIC_TYPE 열거형은 다양한 트래픽 유형을 정의합니다. 각 흐름에는 단일 트래픽 유형이 있습니다. 이렇게 하면 QOS 하위 시스템이 각 형식에 사용자 지정 정책을 적용할 수 있습니다.

qosobjs.h 헤더

아티클2023. 08. 25.

이 헤더는 QOS(서비스 품질)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 서비스 품질(QOS)

qosobjs.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

QOS_DIFFSERV QOS_DIFFSERV 트래픽 제어 개체는 차별화된 서비스 모드에서 작동할 때 패킷 스케줄러에 대한 필터를 지정하는 데 사용됩니다.
QOS_DIFFSERV_RULE QOS_DIFFSERV_RULE 구조는 지정된 흐름에 대한 Diffserv 규칙을 제공하기 위해 QOS_DIFFSERV 트래픽 제어 개체와 함께 사용됩니다.
QOS_DS_CLASS QOS_DS_CLASS 트래픽 제어 개체를 사용하면 애플리케이션 개발자가 지정된 흐름과 연결된 IP 패킷에 대한 기본 Diffserv 코드 포인트(DSCP) 값을 재정의할 수 있습니다. 기본적으로 DSCP 값은 흐름의 ServiceType에서 파생됩니다.
QOS_FRIENDLY_NAME QOS_FRIENDLY_NAME 트래픽 제어 개체는 친숙한 이름을 흐름과 연결합니다.
QOS_TCP_TRAFFIC QOS_TCP_TRAFFIC 구조는 지정된 흐름에 대한 IP 우선 순위 및 UserPriority 매핑을 TCP 트래픽에 대한 시스템 기본값으로 설정해야 함을 나타내는 데 사용됩니다.
QOS_TRAFFIC_CLASS QOS_TRAFFIC_CLASS 트래픽 제어 개체는 지정된 흐름의 트래픽을 분류하는 패킷에 기록된 기본 UserPriority 값을 재정의하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

qospol.h 헤더

아티클2023. 03. 08.

이 헤더는 QOS(서비스 품질)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 서비스 품질(QOS)

qospol.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

IDPE_ATTR IDPE_ATTR 구조에는 ID 정책 요소 특성 정보가 포함됩니다.

qossp.h 헤더

아티클2023. 08. 25.

이 헤더는 QOS(서비스 품질)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 서비스 품질(QOS)

qossp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

AD_GENERAL_PARAMS AD_GENERAL_PARAMS 구조체에는 RSVP Adspec 개체에 포함된 일반 특성화 매개 변수가 포함되어 있습니다.
CONTROL_SERVICE CONTROL_SERVICE 구조에는 지원되는 RSVP 서비스 유형이 포함됩니다.
FLOWDESCRIPTOR FLOWDESCRIPTOR 구조체는 지정된 FLOWSPEC에 대해 하나 이상의 필터를 지정합니다.
IN_ADDR_IPV4 IN_ADDR_IPV4 공용 구조체는 RSVP FILTERSPEC에 사용할 IPv4 주소를 저장합니다.
IN_ADDR_IPV6 IN_ADDR_IPV6 구조체는 RSVP FILTERSPEC에 사용할 IPv6 주소를 저장합니다.
PARAM_BUFFER PARAM_BUFFER 구조체는 CONTROL_SERVICE 구조에 포함할 수 있는 매개 변수 버퍼의 형식을 설명합니다.
QOS_DESTADDR QOS_DESTADDR QOS 개체는 송신 소켓에 대한 연결 함수 호출을 방지하기 위해 WSAIoctl(SIO_SET_QOS) 함수를 호출하는 동안 사용됩니다.
RSVP_ADSPEC

QOS 개체 RSVP_ADSPEC RSVP 기능 및 사용 가능한 서비스와 관련된 발신자와 수신자 간의 데이터 경로를 따라 네트워크 디바이스를 설명하는 정보를 제공하거나 검색하는 방법을 제공합니다.
RSVP_FILTERSPEC RSVP_FILTERSPEC 구조는 RSVP FILTERSPEC 정보를 제공합니다.
RSVP_FILTERSPEC_V4 RSVP_FILTERSPEC_V4 구조체는 IPv4 주소에 FILTERSPEC에 대한 정보를 저장합니다.
RSVP_FILTERSPEC_V4_GPI RSVP_FILTERSPEC_V4_GPI 구조는 지정된 FILTERSPEC에 대한 일반 포트 식별자 정보를 제공합니다.
RSVP_FILTERSPEC_V6 RSVP_FILTERSPEC_V6 구조체는 IPv6 주소에 FILTERSPEC에 대한 정보를 저장합니다.
RSVP_FILTERSPEC_V6_FLOW RSVP_FILTERSPEC_V6_FLOW 구조체는 IPv6 FILTERSPEC에 대한 흐름 레이블 정보를 제공합니다.
RSVP_FILTERSPEC_V6_GPI RSVP_FILTERSPEC_V6_GPI 구조체는 IPv6 주소에서 지정된 FILTERSPEC에 대한 일반 포트 식별자 정보를 제공합니다.
RSVP_POLICY RSVP_POLICY 구조체는 하나 이상의 정의되지 않은 정책 요소를 저장합니다.
RSVP_POLICY_INFO RSVP_POLICY_INFO 구조체는 RSVP에서 검색된 정의되지 않은 정책 요소를 저장합니다.
RSVP_RESERVE_INFO RSVP_RESERVE_INFO QOS 개체는 ProviderSpecific 버퍼를 통해 지정된 흐름에 대한 RSVP 동작을 세분화된 수준에서 지정하거나 수정할 수 있게 하며 흐름에 대한 기본 RSVP 스타일 설정을 재정의할 수 있도록 합니다.
RSVP_STATUS_INFO QOS 개체 RSVP_STATUS_INFO 모니터링 FD_QOS 이벤트와 관련된 이벤트 알림과 오류 정보를 포함하여 지정된 흐름에 대한 RSVP의 상태 대한 정보를 제공합니다.

열거형

FilterType FilterType 열거형은 RSVP FILTERSPEC에 사용되는 필터 유형을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

radialcontrollerinterop.h 헤더

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- windows 런타임 C++ 참조

radialcontrollerinterop.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Interfaces

테이블 확장

IRadialControllerConfigurationInterop WinRT RadialControllerConfiguration 개체와의 상호 운용성을 사용하도록 설정하고 RadialController 메뉴를 사용자 지정하기 위해 RadialControllerConfiguration 멤버에 대한 액세스를 제공합니다.
IRadialControllerInterop WinRT RadialController 개체와의 상호 운용성을 사용하도록 설정하고 상호 작용 환경을 사용자 지정하기 위해 RadialController 멤버에 대한 액세스를 제공합니다.

Last updated on 2025. 10. 08.

ras.h 헤더

아티클2023. 08. 25.

이 헤더는 라우팅 및 원격 액세스 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 라우팅 및 원격 액세스 서비스

ras.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

RasClearConnectionStatistics RasClearConnectionStatistics 함수는 지정된 RAS 연결에 대해 누적된 통계를 지웁니다.
RasClearLinkStatistics RasClearLinkStatistics 함수는 RAS 멀티링크 연결에서 지정된 링크에 대해 누적된 통계를 지웁니다.
RasConnectionNotificationA RasConnectionNotification 함수는 RAS 연결을 만들거나 종료할 때 시스템이 신호 상태로 설정하는 이벤트 개체를 지정합니다. (ANSI)
RasConnectionNotificationW RasConnectionNotification 함수는 RAS 연결을 만들거나 종료할 때 시스템이 신호 상태로 설정하는 이벤트 개체를 지정합니다. (유니코드)
RasCreatePhonebookEntryA RasCreatePhonebookEntry 함수는 새 전화 번호부 항목을 만듭니다. 함수는 사용자가 전화 번호부 항목에 대한 정보를 입력하는 대화 상자를 표시합니다. (ANSI)
RasCreatePhonebookEntryW RasCreatePhonebookEntry 함수는 새 전화 번호부 항목을 만듭니다. 함수는 사용자가 전화 번호부 항목에 대한 정보를 입력하는 대화 상자를 표시합니다. (유니코드)
RasDeleteEntryA RasDeleteEntry 함수는 전화 번호부에서 항목을 삭제합니다. (ANSI)

RasDeleteEntryW RasDeleteEntry 함수는 전화 번호부에서 항목을 삭제합니다. (유니코드)
RasDeleteSubEntryA RasDeleteSubEntry 함수는 지정된 전화 번호부 항목에서 지정된 하위 엔트리를 삭제합니다. (ANSI)
RasDeleteSubEntryW RasDeleteSubEntry 함수는 지정된 전화 번호부 항목에서 지정된 하위 엔트리를 삭제합니다. (유니코드)
RasDialA RasDial 함수는 RAS 클라이언트와 RAS 서버 간에 RAS 연결을 설정합니다. 연결 데이터에는 콜백 및 사용자 인증 정보가 포함됩니다. (ANSI)
RasDialW RasDial 함수는 RAS 클라이언트와 RAS 서버 간에 RAS 연결을 설정합니다. 연결 데이터에는 콜백 및 사용자 인증 정보가 포함됩니다. (유니코드)
RasEditPhonebookEntryA RasEditPhonebookEntry 함수는 기존 전화 번호부 항목을 편집합니다. 함수는 사용자가 기존 정보를 수정할 수 있는 대화 상자를 표시합니다. (ANSI)
RasEditPhonebookEntryW RasEditPhonebookEntry 함수는 기존 전화 번호부 항목을 편집합니다. 함수는 사용자가 기존 정보를 수정할 수 있는 대화 상자를 표시합니다. (유니코드)
RasEnumAutodialAddressesA RasEnumAutodialAddresses 함수는 AutoDial 매핑 데이터베이스의 모든 주소 목록을 반환합니다. (ANSI)
RasEnumAutodialAddressesW RasEnumAutodialAddresses 함수는 AutoDial 매핑 데이터베이스의 모든 주소 목록을 반환합니다. (유니코드)
RasEnumConnectionsA RasEnumConnections 함수는 모든 활성 RAS 연결을 나열합니다. 각 연결의 핸들과 전화 번호부 항목 이름을 반환합니다. (ANSI)

RasEnumConnectionsW RasEnumConnections 함수는 모든 활성 RAS 연결을 나열합니다. 각 연결의 핸들과 전화 번호부 항목 이름을 반환합니다. (유니코드)
RasEnumDevicesA RasEnumDevices 함수는 사용 가능한 모든 RAS 지원 디바이스의 이름과 유형을 반환합니다. (ANSI)
RasEnumDevicesW RasEnumDevices 함수는 사용 가능한 모든 RAS 지원 디바이스의 이름과 유형을 반환합니다. (유니코드)
RasEnumEntriesA RasEnumEntries 함수는 원격 액세스 전화 번호부의 모든 항목 이름을 나열합니다. (ANSI)
RasEnumEntriesW RasEnumEntries 함수는 원격 액세스 전화 번호부의 모든 항목 이름을 나열합니다. (유니코드)
RasFreeEapUserIdentityA RasFreeEapUserIdentity 함수를 사용하여 RasGetEapUserIdentity에서 반환된 메모리 버퍼를 해제합니다. (ANSI)
RasFreeEapUserIdentityW RasFreeEapUserIdentity 함수를 사용하여 RasGetEapUserIdentity에서 반환된 메모리 버퍼를 해제합니다. (유니코드)
RasGetAutodialAddressA RasGetAutodialAddress 함수는 AutoDial 매핑 데이터베이스의 네트워크 주소와 연결된 모든 AutoDial 항목에 대한 정보를 검색합니다. (ANSI)
RasGetAutodialAddressW RasGetAutodialAddress 함수는 AutoDial 매핑 데이터베이스의 네트워크 주소와 연결된 모든 AutoDial 항목에 대한 정보를 검색합니다. (유니코드)
RasGetAutodialEnableA RasGetAutodialEnable 함수는 지정된 TAPI 전화 걸기 위치에 대해 AutoDial 기능을 사용할 수 있는지 여부를 나타냅니다. (ANSI)
RasGetAutodialEnableW

RasGetAutodialEnable 함수는 지정된 TAPI 전화 걸기 위치에 대해 AutoDial 기능을 사용할 수 있는지 여부를 나타냅니다. (유니코드)
RasGetAutodialParamA RasGetAutodialParam 함수는 AutoDial 매개 변수의 값을 검색합니다. (ANSI)
RasGetAutodialParamW RasGetAutodialParam 함수는 AutoDial 매개 변수의 값을 검색합니다. (유니코드)
RasGetConnectionStatistics RasGetConnectionStatistics 함수는 지정된 연결에 대한 누적 연결 통계를 검색합니다.
RasGetConnectStatusA RasGetConnectStatus 함수는 지정된 원격 액세스 연결의 현재 상태 대한 정보를 검색합니다. 애플리케이션은 이 호출을 사용하여 비동기 RasDial 호출이 완료되는 시기를 결정할 수 있습니다. (ANSI)
RasGetConnectStatusW RasGetConnectStatus 함수는 지정된 원격 액세스 연결의 현재 상태 대한 정보를 검색합니다. 애플리케이션은 이 호출을 사용하여 비동기 RasDial 호출이 완료되는 시기를 결정할 수 있습니다. (유니코드)
RasGetCountryInfoA RasGetCountryInfo 함수는 국가/지역의 Windows 전화 통신 목록에서 국가/지역별 전화 걸기 정보를 검색합니다. (ANSI)
RasGetCountryInfoW RasGetCountryInfo 함수는 국가/지역의 Windows 전화 통신 목록에서 국가/지역별 전화 걸기 정보를 검색합니다. (유니코드)
RasGetCredentialsA RasGetCredentials 함수는 지정된 RAS 전화 번호부 항목과 연결된 사용자 자격 증명을 검색합니다. (ANSI)
RasGetCredentialsW RasGetCredentials 함수는 지정된 RAS 전화 번호부 항목과 연결된 사용자 자격 증명을 검색합니다. (유니코드)
RasGetCustomAuthDataA

RasGetCustomAuthData 함수를 사용하여 연결 관련 인증 정보를 검색합니다. 이 정보는 특정 사용자에만 해당되지 않습니다. (ANSI)
RasGetCustomAuthDataW RasGetCustomAuthData 함수를 사용하여 연결 관련 인증 정보를 검색합니다. 이 정보는 특정 사용자에만 해당되지 않습니다. (유니코드)
RasGetEapUserDataA RasGetEapUserData 함수를 사용하여 지정된 전화 번호부 항목에 대한 사용자별 EAP(확장 가능 인증 프로토콜) 정보를 검색합니다. (ANSI)
RasGetEapUserDataW RasGetEapUserData 함수를 사용하여 지정된 전화 번호부 항목에 대한 사용자별 EAP(확장 가능 인증 프로토콜) 정보를 검색합니다. (유니코드)
RasGetEapUserIdentityA RasGetEapUserIdentity 함수는 현재 사용자의 ID 정보를 검색합니다. 이 정보를 사용하여 EAP(확장 가능한 인증 프로토콜)가 필요한 전화 번호부 항목으로 RasDial을 호출합니다. (ANSI)
RasGetEapUserIdentityW RasGetEapUserIdentity 함수는 현재 사용자의 ID 정보를 검색합니다. 이 정보를 사용하여 EAP(확장 가능한 인증 프로토콜)가 필요한 전화 번호부 항목으로 RasDial을 호출합니다. (유니코드)
RasGetEntryDialParamsA RasGetEntryDialParams 함수는 지정된 전화 번호부 항목에 대한 RasDial 또는 RasSetEntryDialParams 함수에 대한 마지막으로 성공한 호출로 저장된 연결 정보를 검색합니다. (ANSI)
RasGetEntryDialParamsW RasGetEntryDialParams 함수는 지정된 전화 번호부 항목에 대한 RasDial 또는 RasSetEntryDialParams 함수에 대한 마지막으로 성공한 호출로 저장된 연결 정보를 검색합니다. (유니코드)
RasGetEntryPropertiesA RasGetEntryProperties 함수는 전화 번호부 항목의 속성을 검색합니다. (ANSI)
RasGetEntryPropertiesW RasGetEntryProperties 함수는 전화 번호부 항목의 속성을 검색합니다. (유니코드)
RasGetErrorStringA

RasGetErrorString 함수는 지정된 RAS 오류 값에 대한 오류 메시지 문자열을 가져옵니다. (ANSI)
RasGetErrorStringW RasGetErrorString 함수는 지정된 RAS 오류 값에 대한 오류 메시지 문자열을 가져옵니다. (유니코드)
RasGetLinkStatistics RasGetLinkStatistics 함수는 RAS 다중 링크 연결에서 지정된 링크에 대한 누적 통계를 검색합니다.
RasGetNapStatus 지정된 원격 액세스 연결에 대한 NAP(네트워크 액세스 보호) 연결 상태 변수를 검색합니다.
RasGetProjectionInfoEx 로컬 클라이언트의 모든 RAS 연결에 대한 PPP(지점 간 프로토콜) 또는 IKEv2(Internet Key Exchange 버전 2) 원격 액세스 프로젝션 작업에 대한 정보를 가져옵니다.
RasGetSubEntryHandleA RasGetSubEntryHandle 함수는 멀티링크 연결의 지정된 하위 엔트리에 대한 연결 핸들을 검색합니다. (ANSI)
RasGetSubEntryHandleW RasGetSubEntryHandle 함수는 멀티링크 연결의 지정된 하위 엔트리에 대한 연결 핸들을 검색합니다. (유니코드)
RasGetSubEntryPropertiesA RasGetSubEntryProperties 함수는 지정된 전화 번호부 항목에 대한 하위 엔트리에 대한 정보를 검색합니다. (ANSI)
RasGetSubEntryPropertiesW RasGetSubEntryProperties 함수는 지정된 전화 번호부 항목에 대한 하위 엔트리에 대한 정보를 검색합니다. (유니코드)
RasHangUpA RasHangUp 함수는 원격 액세스 연결을 종료합니다. 연결은 RAS 연결 핸들로 지정됩니다. 함수는 핸들과 연결된 모든 RASAPI32.DLL 리소스를 해제합니다. (ANSI)
RasHangUpW RasHangUp 함수는 원격 액세스 연결을 종료합니다. 연결은 RAS 연결 핸들로 지정됩니다. 함수는 핸들과 연결된 모든 RASAPI32.DLL 리소스를 해제합니다. (유니코드)

RasInvokeEapUI RasInvokeEapUI 함수는 사용자로부터 EAP(확장 가능 인증 프로토콜) 정보를 가져오는 사용자 지정 사용자 인터페이스를 표시합니다.
RasRenameEntryA RasRenameEntry 함수는 전화 번호부에 있는 항목의 이름을 변경합니다. (ANSI)
RasRenameEntryW RasRenameEntry 함수는 전화 번호부에 있는 항목의 이름을 변경합니다. (유니코드)
RasSetAutodialAddressA RasSetAutodialAddress 함수는 AutoDial 매핑 데이터베이스에 주소를 추가할 수 있습니다. 또는 함수는 데이터베이스의 기존 주소와 연결된 데이터를 삭제하거나 수정할 수 있습니다. (ANSI)
RasSetAutodialAddressW RasSetAutodialAddress 함수는 AutoDial 매핑 데이터베이스에 주소를 추가할 수 있습니다. 또는 함수는 데이터베이스의 기존 주소와 연결된 데이터를 삭제하거나 수정할 수 있습니다. (유니코드)
RasSetAutodialEnableA RasSetAutodialEnable 함수는 지정된 TAPI 전화 걸기 위치에 대해 AutoDial 기능을 사용하거나 사용하지 않도록 설정합니다. (ANSI)
RasSetAutodialEnableW RasSetAutodialEnable 함수는 지정된 TAPI 전화 걸기 위치에 대해 AutoDial 기능을 사용하거나 사용하지 않도록 설정합니다. (유니코드)
RasSetAutodialParamA RasSetAutodialParam 함수는 AutoDial 매개 변수의 값을 설정합니다. (ANSI)
RasSetAutodialParamW RasSetAutodialParam 함수는 AutoDial 매개 변수의 값을 설정합니다. (유니코드)
RasSetCredentialsA 지정된 RAS 전화 번호부 항목과 연결된 사용자 자격 증명을 설정합니다. (ANSI)
RasSetCredentialsW 지정된 RAS 전화 번호부 항목과 연결된 사용자 자격 증명을 설정합니다. (유니코드)

RasSetCustomAuthDataA RasSetCustomAuthData 함수를 사용하여 연결 관련 인증 정보를 설정합니다. 이 정보는 특정 사용자에 한정되어서는 안 됩니다. (ANSI)
RasSetCustomAuthDataW RasSetCustomAuthData 함수를 사용하여 연결 관련 인증 정보를 설정합니다. 이 정보는 특정 사용자에 한정되어서는 안 됩니다. (유니코드)
RasSetEapUserDataA RasSetEapUserData 함수를 사용하여 레지스트리에 지정된 전화 번호부 항목에 대한 사용자별 EAP(확장 가능 인증 프로토콜) 정보를 저장합니다. (ANSI)
RasSetEapUserDataW RasSetEapUserData 함수를 사용하여 레지스트리에 지정된 전화 번호부 항목에 대한 사용자별 EAP(확장 가능 인증 프로토콜) 정보를 저장합니다. (유니코드)
RasSetEntryDialParamsA RasSetEntryDialParams 함수는 지정된 전화 번호부 항목에 대한 RasDial 또는 RasSetEntryDialParams 함수에 대한 마지막으로 성공한 호출로 저장된 연결 정보를 변경합니다. (ANSI)
RasSetEntryDialParamsW RasSetEntryDialParams 함수는 지정된 전화 번호부 항목에 대한 RasDial 또는 RasSetEntryDialParams 함수에 대한 마지막으로 성공한 호출로 저장된 연결 정보를 변경합니다. (유니코드)
RasSetEntryPropertiesA RasSetEntryProperties 함수는 전화 번호부의 항목에 대한 연결 정보를 변경하거나 새 전화 번호부 항목을 만듭니다. (ANSI)
RasSetEntryPropertiesW RasSetEntryProperties 함수는 전화 번호부의 항목에 대한 연결 정보를 변경하거나 새 전화 번호부 항목을 만듭니다. (유니코드)
RasSetSubEntryPropertiesA RasSetSubEntryProperties 함수는 새 하위 엔트리를 만들거나 지정된 전화 번호부 항목의 기존 하위 엔트리를 수정합니다. (ANSI)
RasSetSubEntryPropertiesW

RasSetSubEntryProperties 함수는 새 하위 엔트리를 만들거나 지정된 전화 번호부 항목의 기존 하위 엔트리를 수정합니다. (유니코드)
RasUpdateConnection RasUpdateConnection 함수는 IKEv2(Internet Key Exchange 버전 2) 연결의 터널 엔드포인트를 업데이트합니다.
RasValidateEntryNameA RasValidateEntryName 함수는 연결 항목 이름의 형식의 유효성을 검사합니다. 이름에는 공백이 아닌 영숫자 문자가 하나 이상 포함되어야 합니다. (ANSI)
RasValidateEntryNameW RasValidateEntryName 함수는 연결 항목 이름의 형식의 유효성을 검사합니다. 이름에는 공백이 아닌 영숫자 문자가 하나 이상 포함되어야 합니다. (유니코드)

콜백 함수

ORASADFUNC ORASADFunc 함수는 자동 완성을 위해 사용자 지정된 사용자 인터페이스를 제공하는 데 사용되는 애플리케이션 정의 콜백 함수입니다.
PFNRASFREEBUFFER 사용자 지정 스크립팅 DLL은 RasFreeBuffer를 호출하여 RasGetBuffer에 대한 이전 호출에 의해 할당된 메모리 버퍼를 해제합니다.
PFNRASGETBUFFER 사용자 지정 스크립팅 DLL은 RasGetBuffer를 호출하여 서버에 연결된 포트를 통해 데이터를 보내거나 받기 위한 메모리를 할당합니다.
PFNRASRECEIVEBUFFER 사용자 지정 스크립팅 DLL은 RasReceiveBuffer 함수를 호출하여 지정된 포트를 통해 서버에서 데이터를 받을 준비가 되었음을 RAS에 알릴 수 있습니다.
PFNRASRETRIEVEBUFFER 사용자 지정 스크립팅 DLL은 RasRetrieveBuffer 함수를 호출하여 지정된 포트를 통해 RAS 서버에서 받은 데이터를 가져옵니다.

PFNRASSENDBUFFER 사용자 지정 스크립팅 DLL은 RasSendBuffer 함수를 호출하여 지정된 포트를 통해 서버로 데이터를 보냅니다.
PFNRASSETCOMMSETTINGS 사용자 지정 스크립팅 DLL에서 RasSetCommSettings를 호출하여 연결에 대한 포트의 설정을 변경합니다.
RASADFUNCA RASADFunc 함수는 자동 진단에 대한 사용자 지정 사용자 인터페이스를 제공하는 데 사용되는 애플리케이션 정의 콜백 함수입니다. (ANSI)
RASADFUNCW RASADFunc 함수는 자동 진단에 대한 사용자 지정 사용자 인터페이스를 제공하는 데 사용되는 애플리케이션 정의 콜백 함수입니다. (유니코드)
RasCustomDeleteEntryNotifyFn RasCustomDeleteEntryNotify 함수는 타사 사용자 지정 전화 걸기 DLL에서 내보내는 애플리케이션 정의 함수입니다. 이 함수를 사용하면 타사 공급업체가 전화 번호부 항목을 관리하기 위한 사용자 지정 대화 상자를 구현할 수 있습니다.
RasCustomDialFn RasCustomDial 함수는 타사 사용자 지정 전화 걸기 DLL에서 내보내는 애플리케이션 정의 함수입니다. 이 함수를 사용하면 타사 공급업체가 사용자 지정 원격 액세스 전화 걸기 루틴을 구현할 수 있습니다.
RasCustomHangUpFn RasCustomHangUp 함수는 타사 사용자 지정 전화 걸기 DLL에서 내보내는 애플리케이션 정의 함수입니다. 이 함수를 사용하면 타사 공급업체가 사용자 지정 연결 중단 루틴을 구현할 수 있습니다.
RasCustomScriptExecuteFn RAS는 RASEO_CustomScript 옵션이 설정된 전화 번호부 항목에 대한 연결을 설정할 때 RasCustomScriptExecute 함수를 호출합니다.
RASDIALFUNC RasDialFunc 콜백 함수는 RAS 연결 프로세스 중에 상태 변경이 발생할 때 RasDial 함수에 의해 호출됩니다.
RASDIALFUNC1

RasDialFunc1 함수는 원격 액세스 연결 프로세스 중에 상태 변경이 발생할 때 RasDial 함수에 의해 호출됩니다.
RASDIALFUNC2 RasDialFunc2 콜백 함수는 원격 액세스 연결 프로세스 중에 상태 변경이 발생할 때 RasDial 함수 호출에 의해 호출됩니다.

구조체

RAS_PROJECTION_INFO RAS 연결에 대한 PPP(지점 간) 또는 IKEv2(인터넷 키 교환 버전 2) 프로젝션 정보를 포함합니다.
RAS_STATS RAS_STATS 구조체는 단일 링크 RAS 연결 또는 다중 링크 RAS 연결의 링크 중 하나에 대한 통계를 저장합니다.
RASDEVSPECIFICINFO 서버 유효성 검사를 위한 쿠키를 보내고 PPP(지점 및 지점 및 지점) 인증을 우회하는 데 사용됩니다.
RASIKEV2_PROJECTION_INFO IKE(인터넷 키 교환) 협상 중에 얻은 프로젝션 정보를 포함합니다.
RASNAPSTATE 원격 액세스 연결에 대한 NAP(네트워크 액세스 보호) 변수입니다.
RASPPP_PROJECTION_INFO 인터넷 프로토콜 버전 4(IPv4) 및 IPv6 프로젝션 작업 및 PPP LCP(링크 제어 프로토콜)/멀티링크 및 CCP(압축 제어 프로토콜) 협상 중에 얻은 정보를 포함합니다.

열거형

RASPROJECTION_INFO_TYPE

RAS_PROJECTION_INFO 구조체에서 RASPPP_PROJECTION_INFO 또는 RASIKEV2_PROJECTION_INFO 구조를 나타내는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

rasdlg.h 헤더

아티클2023. 08. 25.

이 헤더는 라우팅 및 원격 액세스 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 라우팅 및 원격 액세스 서비스

rasdlg.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

RasDialDlgA RasDialDlg 함수는 지정된 전화 번호부 항목과 로그온한 사용자의 자격 증명을 사용하여 RAS 연결을 설정합니다. 함수는 연결 작업의 상태를 나타내는 대화 상자 스트림을 표시합니다. (ANSI)
RasDialDlgW RasDialDlg 함수는 지정된 전화 번호부 항목과 로그온한 사용자의 자격 증명을 사용하여 RAS 연결을 설정합니다. 함수는 연결 작업의 상태를 나타내는 대화 상자 스트림을 표시합니다. (유니코드)
RasEntryDlgA RasEntryDlg 함수는 사용자가 전화 번호부 항목을 조작할 수 있는 모달 속성 시트를 표시합니다. (ANSI)
RasEntryDlgW RasEntryDlg 함수는 사용자가 전화 번호부 항목을 조작할 수 있는 모달 속성 시트를 표시합니다. (유니코드)
RasPhonebookDlgA RasPhonebookDlg 함수는 기본 전화 접속 네트워킹 대화 상자를 표시합니다. (ANSI)
RasPhonebookDlgW RasPhonebookDlg 함수는 기본 전화 접속 네트워킹 대화 상자를 표시합니다. (유니코드)

콜백 함수

RasCustomDialDlgFn RasCustomDialDlg 함수는 타사 사용자 지정 전화 걸기 DLL에서 내보내는 애플리케이션 정의 함수입니다. 이 함수를 사용하면 타사 공급업체에서 사용자 지정 RAS 연결 대화 상자를 구현할 수 있습니다.
RasCustomEntryDlgFn RasCustomEntryDlg 함수는 타사 사용자 지정 전화 걸기 DLL에서 내보내는 애플리케이션 정의 함수입니다. 이 기능을 사용하면 타사 공급업체가 전화 번호부 항목을 관리하기 위한 사용자 지정 대화 상자를 구현할 수 있습니다.
RASPBDLGFUNCA RasPBDlgFunc 함수는 RasPhonebookDlg 대화 상자가 열려 있는 동안 사용자 활동에 대한 알림을 받는 애플리케이션 정의 콜백 함수입니다. (ANSI)
RASPBDLGFUNCW RasPBDlgFunc 함수는 RasPhonebookDlg 대화 상자가 열려 있는 동안 사용자 활동에 대한 알림을 받는 애플리케이션 정의 콜백 함수입니다. (유니코드)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

raseapif.h 헤더

아티클2023. 08. 25.

이 헤더는 확장 가능한 인증 프로토콜에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 확장할 수 있는 인증 프로토콜

raseapif.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

RasEapFreeMemory RAS 연결 관리자는 RasEapFreeMemory를 호출하여 RasEapInvokeConfigUI, RasEapGetIdentity 및 RasEapInvokeInteractiveUI에서 반환된 메모리 버퍼를 해제합니다.
RasEapGetIdentity RAS 연결 관리자는 RasEapGetIdentity 함수를 호출하여 인증을 요청하는 사용자의 ID 정보를 가져옵니다.
RasEapGetInfo RAS 연결 관리자는 RasEapGetInfo를 호출하여 지정된 인증 프로토콜에 대한 함수 포인터 집합을 가져옵니다.
RasEapInvokeConfigUI RAS 연결 관리자는 RasEapInvokeConfigUI 함수를 호출하여 사용자로부터 구성 정보를 가져오는 대화 상자를 표시합니다.
RasEapInvokeInteractiveUI RAS 연결 관리자는 RasEapInvokeInteractiveUI 함수를 호출하여 사용자로부터 인증 데이터를 가져오는 대화 상자를 표시합니다.

구조체

PPP_EAP_INFO

PPP_EAP_INFO 구조는 EAP DLL에 있는 함수에 대한 포인터를 포함하여 인증 프로토콜에 대한 정보를 연결 관리자 제공합니다.
PPP_EAP_INPUT PPP_EAP_INPUT 구조는 RAS 연결 관리자 Service PPP 구현과 EAP 간의 상호 작용에 사용됩니다.
PPP_EAP_OUTPUT 인증 프로토콜은 PPP_EAP_OUTPUT 구조를 사용하여 RasEapMakeMessage 호출에서 반환할 때 요청을 전달하고 정보를 연결 관리자 상태.
PPP_EAP_PACKET PPP_EAP_PACKET 구조는 인증 프로토콜에서 처리되는 패킷에 대한 정보를 지정합니다.
RAS_AUTH_ATTRIBUTE RAS_AUTH_ATTRIBUTE 구조는 EAP 세션 중에 RAS_AUTH_ATTRIBUTE_TYPE 유형의 인증 특성을 전달하는 데 사용됩니다.

열거형

PPP_EAP_ACTION PPP_EAP_ACTION 열거형 형식은 연결 관리자 인증 프로토콜을 대신하여 수행해야 하는 작업을 지정합니다.
RAS_AUTH_ATTRIBUTE_TYPE RAS_AUTH_ATTRIBUTE_TYPE 열거형 형식은 세션 인증에 사용되는 특성 값을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

rasshost.h 헤더

아티클2023. 08. 25.

이 헤더는 라우팅 및 원격 액세스 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 라우팅 및 원격 액세스 서비스

rasshost.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

RasSecurityDialogBegin RasSecurityDialogBegin 함수는 원격 사용자가 연결을 시도할 때 RAS 서버가 호출하는 타사 RAS 보안 DLL 진입점입니다. 이렇게 하면 보안 DLL이 원격 사용자의 인증을 시작할 수 있습니다.
RasSecurityDialogComplete RasSecurityDialogComplete 함수는 RAS 서버에 타사 보안 인증 트랜잭션의 결과를 알릴 수 있습니다.
RasSecurityDialogEnd RasSecurityDialogEnd 함수는 RAS 서버가 인증 트랜잭션을 종료하기 위해 호출하는 타사 RAS 보안 DLL 진입점입니다.
RasSecurityDialogGetInfo RasSecurityDialogGetInfo 함수는 RAS 서버에서 포트에 대한 정보를 가져오기 위해 RAS 보안 DLL에 의해 호출됩니다.
RasSecurityDialogReceive RasSecurityDialogReceive 함수는 보안 문제에 대한 원격 사용자의 응답을 수신하는 비동기 작업을 시작합니다.
RasSecurityDialogSend RasSecurityDialogSend 함수는 원격 컴퓨터의 터미널 창에 표시할 메시지를 보냅니다. 타사 RAS 보안 DLL은 원격 사용자 인증의 일부로 이 메시지를 보냅니다.

구조체

RAS_SECURITY_INFO RAS_SECURITY_INFO 구조는 RAS 보안 DLL 인증 트랜잭션과 연결된 RAS 포트에 대한 정보를 반환하기 위해 RasSecurityDialogGetInfo 함수와 함께 사용됩니다.
SECURITY_MESSAGE SECURITY_MESSAGE 구조는 RAS 보안 DLL 인증 트랜잭션의 결과를 나타내기 위해 RasSecurityDialogComplete 함수와 함께 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

rdpappcontainerclient.h 헤더

아티클2023. 08. 25.

이 헤더는 원격 데스크톱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 데스크톱 서비스

rdpappcontainerclient.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IRemoteDesktopClient RDP(원격 데스크톱 프로토콜) 앱 컨테이너 클라이언트 컨트롤을 구성하고 사용하는 데 사용되는 메서드 및 속성을 제공합니다.
IRemoteDesktopClientActions RDP(원격 데스크톱 프로토콜) 앱 컨테이너 클라이언트 컨트롤과 상호 작용하는 데 사용되는 메서드를 제공합니다.
IRemoteDesktopClientSettings RDP(원격 데스크톱 프로토콜) 앱 컨테이너 클라이언트 컨트롤에 대한 연결 설정을 구성하는 데 필요한 메서드를 제공합니다.
IRemoteDesktopClientTouchPointer RDP(원격 데스크톱 프로토콜) 앱 컨테이너 클라이언트 컨트롤의 터치 포인터 기능을 제어하는 데 필요한 속성을 제공합니다.

열거형

RemoteActionType 원격 세션으로 보낼 작업입니다.
SnapshotEncodingType RDP(원격 데스크톱 프로토콜) 앱 컨테이너 클라이언트에 사용되는 인코딩 유형은 스냅샷.

SnapshotFormatType RDP(원격 데스크톱 프로토콜) 앱 컨테이너 클라이언트에 사용되는 데이터 형식은 스냅샷.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

rdpencomapi.h 헤더

아티클2023. 10. 04.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 데스크톱 서비스
- Windows 데스크톱 공유

rdpencomapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

_IRDPSessionEvents 이벤트가 발생할 때 알림을 받도록 이 인터페이스를 구현합니다.
IRDPSRAPIApplication 프로세스 내에서 공유 가능한 창을 그룹화합니다. 각 애플리케이션 개체에는 창 개체 목록이 포함됩니다. 애플리케이션 개체가 공유되면 모든 창이 공유됩니다.
IRDPSRAPIApplicationFilter 창 및 프로세스 수준에서 공유 데스크톱 영역을 관리합니다. 애플리케이션은 열거자를 사용하여 공유할 수 있는 세션의 개체 목록을 표시할 수 있습니다.
IRDPSRAPIApplicationList 애플리케이션 목록을 관리합니다.
IRDPSRAPIAttendee 참석자 개체는 클라이언트가 세션에 연결하고 인증된 결과로 만들어집니다. 참석자 개체가 만들어지면 참석자 목록에 자동으로 추가됩니다.
IRDPSRAPIAttendeeDisconnectInfo 참석자의 연결이 끊긴 이유에 대한 정보를 포함합니다.
IRDPSRAPIAttendeeManager 참석자 개체를 관리합니다.
IRDPSRAPIAudioStream

공동 작업 공유자 Microsoft ActiveX 컨트롤에서 공동 작업 뷰어 컨트롤로 오디오 스트림을 보낼 수 있습니다.
IRDPSRAPIClipboardUseEvents 공유자 쪽에서 이 인터페이스를 구현하여 클립보드 사용을 추적하거나 제어합니다. 클립보드 공유를 사용하도록 설정하지 않으면 이 인터페이스가 적용되지 않습니다. 속성에 설명된 SetClipboardRedirectCallback 속성에 대한 값을 설정해야 합니다.
IRDPSRAPIDebug Microsoft에서 원격 데스크톱 ActiveX 클라이언트를 테스트하는 데 사용됩니다.
IRDPSRAPIFrameBuffer 프레임 버퍼 크기 및 형식에 대한 데이터를 제공하고 콘텐츠를 검색할 수 있도록 합니다.
IRDPSRAPIInvitation 초대를 사용하면 사용자 또는 사용자 그룹이 세션에 연결할 수 있습니다. 참석자가 세션에 연결하면 클라이언트는 티켓과 암호를 보냅니다. 이 두 가지 정보는 참석자를 인증하는 데 사용됩니다.
IRDPSRAPIInvitationManager 초대 개체를 관리합니다.
IRDPSRAPIPerfCounterLogger 클라이언트 애플리케이션이 사용자 지정 성능 로깅을 구현할 수 있도록 합니다.
IRDPSRAPIPerfCounterLoggingManager IRDPSRAPIPerfCounterLogger 개체를 관리합니다.
IRDPSRAPISessionProperties 이 인터페이스를 사용하여 세션 속성을 얻거나 설정합니다.
IRDPSRAPISharingSession 협업 세션을 시작하기 위해 애플리케이션에서 만들어야 하는 기본 개체입니다. (IRDPSRAPISharingSession)
IRDPSRAPISharingSession2 협업 세션을 시작하기 위해 애플리케이션에서 만들어야 하는 기본 개체입니다. (IRDPSRAPISharingSession2)
IRDPSRAPITcpConnectionInfo

뷰어 및 공유자 쪽에서 TCP 연결 정보를 검색하는 메서드를 지원합니다.
IRDPSRAPITransportStream 스트림을 사용하여 작업을 수행하는 메서드를 노출합니다.
IRDPSRAPITransportStreamBuffer 데이터를 보내고 받기 위해 IRDPSRAPITransportStream 인터페이스에서 만들고 사용합니다.
IRDPSRAPITransportStreamEvents 스트림 인터페이스(IRDPSRAPITransportStream)에서 호출한 메서드를 노출하여 RDP(원격 데스크톱 프로토콜) 스택에 이벤트 완료에 대해 알립니다.
IRDPSRAPIViewer 뷰어 쪽에서 사용되는 ActiveX 인터페이스입니다.
IRDPSRAPIVirtualChannel 가상 채널을 관리합니다.
IRDPSRAPIVirtualChannelManager 가상 채널 목록을 관리합니다.
IRDPSRAPIWindow 공유 가능한 창에 대한 일대일 매핑을 나타냅니다.
IRDPSRAPIWindowList 창 목록을 관리합니다.
IRDPViewerInputSink 마우스 및 키보드 이벤트를 보내고 터치 입력을 지원합니다.

열거형

ATTENDEE_DISCONNECT_REASON 참석자가 세션에서 연결이 끊긴 이유에 대한 값을 정의합니다.

CHANNEL_ACCESS_ENUM 채널에 대해 참석자에게 부여된 액세스 유형에 대한 값을 정의합니다.
CHANNEL_FLAGS 채널에서 데이터를 보내는 방법에 대한 값을 정의합니다.
CHANNEL_PRIORITY 채널에서 패킷을 보내는 데 사용되는 우선 순위에 대한 값을 정의합니다.
CTRL_LEVEL 참석자가 세션에 대해 갖는 제어 수준을 정의합니다.
RDPENCOMAPI_ATTENDEE_FLAGS 참석자 유형에 대한 값을 정의합니다.
RDPSRAPI_APP_FLAGS 애플리케이션 유형에 대한 값을 정의합니다.
RDPSRAPI_KBD_CODE_TYPE 키보드 인코딩 유형에 대한 값을 정의합니다.
RDPSRAPI_KBD_SYNC_FLAG Caps Lock 키와 같이 키보드의 상태를 제어하는 키 유형에 대한 값을 정의합니다.
RDPSRAPI_MOUSE_BUTTON_TYPE 마우스 단추 유형에 대한 값을 정의합니다.
RDPSRAPI_WND_FLAGS 창 유형에 대한 값을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

realtimeapiset.h 헤더

아티클2023. 08. 25.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

realtimeapiset.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

ConvertAuxiliaryCounterToPerformanceCounter 지정된 보조 카운터 값을 해당 성능 카운터 값으로 변환합니다. 필요에 따라 대기 시간 및 가능한 최대 드리프트로 인한 나노초 단위의 예상 변환 오류를 제공합니다.
ConvertPerformanceCounterToAuxiliaryCounter 지정된 성능 카운터 값을 해당 보조 카운터 값으로 변환합니다. 필요에 따라 대기 시간 및 가능한 최대 드리프트로 인한 나노초 단위의 예상 변환 오류를 제공합니다.
QueryAuxiliaryCounterFrequency 보조 카운터 빈도를 쿼리합니다.
QueryIdleProcessorCycleTime 시스템의 각 프로세서의 유휴 스레드에 대한 주기 시간을 검색합니다.
QueryIdleProcessorCycleTimeEx 지정된 프로세서 그룹의 각 논리 프로세서에서 유휴 스레드에 대해 누적된 주기 시간을 검색합니다.
QueryInterruptTime 현재 인터럽트 시간 수를 가져옵니다.
QueryInterruptTimePrecise QueryInterruptTime보다 더 정확한 형식으로 현재 인터럽트 시간 수를 가져옵니다.
QueryProcessCycleTime 지정된 프로세스의 모든 스레드에 대한 주기 시간의 합계를 검색합니다.

QueryThreadCycleTime 지정된 스레드의 주기 시간을 검색합니다.
QueryUnbiasedInterruptTime 현재 편견 없는 인터럽트 시간 수를 100나노초 단위로 가져옵니다. 편견 없는 인터럽트 시간 수는 시스템이 절전 모드 또는 최대 절전 모드에서 보내는 시간을 포함하지 않습니다.
QueryUnbiasedInterruptTimePrecise QueryUnbiasedInterruptTime보다 더 정확한 형식으로 현재 편견 없는 인터럽트 시간 수를 가져옵니다. 편견 없는 인터럽트 시간 수는 시스템이 절전 모드 또는 최대 절전 모드에서 보내는 시간을 포함하지 않습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

recapis.h 헤더

아티클2023. 08. 25.

이 헤더는 태블릿 PC에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Tablet PC

recapis.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AddStroke RecognizerContext에 잉크 스트로크를 추가합니다.
AddWordsToWordList 단어 목록에 하나 이상의 단어를 추가합니다.
AdviseInkChange 스트로크가 추가되거나 삭제되었으므로 인식기가 잉크 처리를 중지합니다.
CloneContext 원본과 동일한 설정을 포함하는 인식기 컨텍스트를 만듭니다. 새 인식기 컨텍스트에는 원본의 잉크 또는 인식 결과가 포함되지 않습니다.
CreateContext 인식기 컨텍스트를 만듭니다.
CreateRecognizer 인식기를 만듭니다.
DestroyAlternate 이 함수는 사용되지 않으며 사용자 지정 애플리케이션 인식기에서 구현할 필요가 없습니다.
DestroyContext 인식기 컨텍스트를 삭제합니다.
DestroyRecognizer

인식기를 삭제합니다.
DestroyWordList 현재 단어 목록을 삭제합니다.
EndInkInput 컨텍스트에 더 이상 잉크가 추가되지 않음을 나타냅니다. 이 함수를 호출한 후에는 컨텍스트에 스트로크를 추가할 수 없습니다.
GetAllRecognizers 모든 인식기를 가져옵니다.
GetBestResultString 최상의 결과 문자열을 검색합니다.
GetContextPreferenceFlags 컨텍스트 기본 설정 플래그를 가져옵니다.
GetContextPropertyList 인식기가 지원하는 속성 목록을 검색합니다.
GetContextPropertyValue 인식기 컨텍스트에서 지정된 속성 값을 반환합니다.
GetEnabledUnicodeRanges 컨텍스트에서 사용하도록 설정된 유니코드 지점 범위 목록을 검색합니다. SetEnabledUnicodeRanges 함수를 호출하여 활성화된 범위를 지정하지 않으면 이 함수는 인식기 기본 유니코드 점 범위를 반환합니다.
GetGuide 박스형, 줄 지어 또는 자유형 입력에 사용되는 가이드를 검색합니다.
GetLatticePtr 현재 결과의 격자에 대한 포인터를 검색합니다.
GetLeftSeparator 인식기 컨텍스트의 왼쪽 구분 기호를 가져옵니다.

GetPreferredPacketDescription 인식기에서 사용하는 패킷 속성이 포함된 패킷 설명을 검색합니다.
GetRecoAttributes 인식기의 특성을 검색합니다.
GetResultPropertyList 인식기가 결과 범위에 대해 반환할 수 있는 속성 목록을 검색합니다.
GetRightSeparator 인식기 컨텍스트에 대한 오른쪽 구분 기호를 가져옵니다.
GetUnicodeRanges 인식기가 지원하는 유니코드 지점의 범위를 반환합니다.
IsStringSupported 전달된 단어, 날짜, 시간, 숫자 또는 기타 텍스트가 사전에 포함되어 있는지 여부를 나타내는 값을 반환합니다. 이 테스트의 결과는 factoid 설정에 따라 달라집니다.
LoadCachedAttributes 인식기의 캐시된 특성을 로드합니다.
MakeWordList 단어 목록을 만듭니다.
처리 잉크 인식을 동기적으로 수행합니다.
ResetContext 컨텍스트에서 현재 잉크 및 인식 결과를 삭제합니다. 인식기 컨텍스트의 현재 설정이 유지됩니다.
SetCACMode 문자 또는 단어 인식에 대한 문자 자동 완성 모드를 지정합니다. 문자 자동 완성을 설정한 후에는 해제할 수 없습니다.
SetContextPropertyValue 인식기 컨텍스트에 속성을 추가합니다. 속성이 이미 있는 경우 해당 값이 수정됩니다.

SetEnabledUnicodeRanges 컨텍스트에서 하나 이상의 유니코드 지점 범위를 사용하도록 설정합니다.
SetFactoid 인식기가 결과에 대한 검색을 제한하는 데 사용하는 factoid를 지정합니다. 입력 필드에 날짜가 포함된 경우와 같이 입력 필드가 알려진 형식인 경우 factoid를 지정합니다.
SetFlags 인식기가 잉크를 해석하고 결과 문자열을 결정하는 방법을 지정합니다. 잉크를 처음으로 처리하기 전에 이 함수를 호출합니다. 따라서 Process 함수를 호출하기 전에 SetFlags 함수를 호출합니다.
SetGuide 박스형 또는 줄 지어 입력에 사용할 인식 가이드를 설정합니다. 컨텍스트에 스트로크를 추가하기 전에 이 함수를 호출해야 합니다.
SetTextContext 인식기 컨텍스트에 포함된 텍스트 앞과 뒤의 텍스트 문자열을 제공합니다. 잉크를 처음으로 처리하기 전에 이 함수를 호출합니다. 따라서 Process 함수를 호출하기 전에 SetTextContext 함수를 호출합니다.
SetWordList 인식할 현재 인식기 컨텍스트의 단어 목록을 설정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

reconcil.h 헤더

아티클2023. 08. 25.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 레거시 Windows 환경 기능
- Windows 셸

reconcil.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

INotifyReplica 개체의 작성자에게 후속 조정의 대상이 될 수 있음을 개체에 알리는 수단을 제공하는 메서드를 노출합니다. 서류 가방 조정자는 이 인터페이스 구현을 담당합니다.
IReconcilableObject 지정된 문서를 조정하는 메서드를 노출합니다. 서류 가방 조정자는 이 인터페이스 구현을 담당합니다.
IReconcileInitiator 초기자에게 진행 상황을 알리고, 종료 개체를 설정하고, 지정된 버전의 문서를 요청하는 수단을 서류 가방 조정자에게 제공하는 메서드를 노출합니다. 초기자는 이 인터페이스를 구현해야 합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

rectypes.h 헤더

아티클2023. 08. 25.

이 헤더는 태블릿 PC에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Tablet PC

rectypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

CHARACTER_RANGE 유니코드 점(문자) 범위를 지정합니다.
LATTICE_METRICS 기준선 및 중간선 높이를 설명합니다.
LINE_SEGMENT 기준선 또는 중간선과 같은 선 인식 세그먼트의 시작점과 끝점을 설명합니다.
RECO_ATTRS 인식기의 특성을 검색하거나 설치된 인식기를 검색할 때 사용할 특성을 지정합니다.
RECO_GUIDE 인식기 잉크의 경계를 정의합니다.
RECO_LATTICE 격자에 진입점 역할을합니다.
RECO_LATTICE_COLUMN 격자의 열을 나타냅니다.
RECO_LATTICE_ELEMENT 일반적으로 한 단어 또는 한 개의 동아시아 문자에 해당합니다. 그러나 요소는 제스처, 도형 또는 기타 코드에 해당할 수도 있습니다.
RECO_LATTICE_PROPERTIES

속성 구조에 대한 포인터 배열을 포함합니다.
RECO_LATTICE_PROPERTY 격자에 사용되는 속성을 포함합니다.
RECO_RANGE 구조체는 사용되지 않습니다.

열거형

ALT_BREAKS 최상의 결과 문자열에서 대체를 만드는 방법을 지정합니다.
CONFIDENCE_LEVEL 인식기에서 인식 결과에 있는 신뢰도 수준을 나타냅니다.
LINE_METRICS 인식 세그먼트에 있는 줄을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

refptrco.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows Management Instrumentation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Management Instrumentation

refptrco.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

TRefPointerCollection TRefPointerCollection 클래스는 개체에 대한 포인터를 수집하는 컨테이너 클래스입니다. 이러한 포인터는 열거할 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

relogger.h 헤더

아티클2023. 08. 25.

이 헤더는 이벤트 추적에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 이벤트 추적

relogger.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ITraceEvent 특정 이벤트와 관련된 데이터에 대한 액세스를 제공합니다.
ITraceEventCallback ETW에서 추적 프로세스가 시작, 종료 및 로그 이벤트를 시작할 때 재로거에 정보를 제공하는 데 사용됩니다.
ITraceRelogger ETW 추적 스트림에서 이벤트를 조작하고 다시 로깅할 수 있도록 다시 로깅 기능에 대한 액세스를 제공합니다.

Functions

DECLSPEC_XFGVIRT

구조체

ETW_BUFFER_CONTEXT 이벤트에 대한 컨텍스트 정보를 제공합니다. (ETW_BUFFER_CONTEXT)

EVENT_DESCRIPTOR 이벤트를 정의하는 메타데이터를 포함합니다.
EVENT_HEADER EVENT_HEADER 구조체(relogger.h)는 이벤트에 대한 정보를 정의합니다.
EVENT_HEADER_EXTENDED_DATA_ITEM EVENT_HEADER_EXTENDED_DATA_ITEM 구조체(relogger.h)는 ETW가 이벤트 데이터의 일부로 수집하는 확장 데이터를 정의합니다.
EVENT_RECORD EVENT_RECORD 구조체(relogger.h)는 ETW에서 제공하는 이벤트의 레이아웃을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

remotesystemadditionalinfo.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

remotesystemadditionalinfo.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

HSTRING_UserFree HSTRING_UserFree 함수는 RPC 스텁 파일에서 호출할 때 서버 쪽의 리소스를 해제합니다. (HSTRING_UserFree)
HSTRING_UserFree64 HSTRING_UserFree64 함수는 RPC 스텁 파일에서 호출할 때 서버 쪽의 리소스를 해제합니다. (HSTRING_UserFree64)
HSTRING_UserMarshal HSTRING_UserMarshal 함수는 HSTRING 개체를 RPC 버퍼로 마샬링합니다. (HSTRING_UserMarshal)
HSTRING_UserMarshal64 HSTRING_UserMarshal64 함수는 HSTRING 개체를 RPC 버퍼로 마샬링합니다. (HSTRING_UserMarshal64)
HSTRING_UserSize HSTRING_UserSize 함수는 HSTRING 개체의 와이어 크기를 계산하고 핸들과 데이터를 검색합니다. (HSTRING_UserSize)
HSTRING_UserSize64 HSTRING_UserSize64 함수는 HSTRING 개체의 와이어 크기를 계산하고 핸들과 데이터를 검색합니다. (HSTRING_UserSize64)
HSTRING_UserUnmarshal

HSTRING_UserUnmarshal 함수는 RPC 버퍼에서 HSTRING 개체를 숨기지 않습니다. (HSTRING_UserUnmarshal)
HSTRING_UserUnmarshal64 HSTRING_UserUnmarshal64 함수는 RPC 버퍼에서 HSTRING 개체를 숨기지 않습니다. (HSTRING_UserUnmarshal64)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

rend.h 헤더

아티클2023. 08. 25.

이 헤더는 전화 통신 애플리케이션 프로그래밍 인터페이스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 전화 통신 애플리케이션 프로그래밍 인터페이스

rend.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumDialableAddrs IEnumDialableAddrs 인터페이스는 COM 표준 열거형 메서드를 제공하여 디렉터리에서 사용 가능한 전화 걸기 주소를 검색하고 사용합니다. ITDirectoryObject::EnumerateDialableAddrs 메서드는 이 인터페이스에 대한 포인터를 반환합니다.
IEnumDirectory IEnumDirectory 인터페이스는 ITDirectory 인터페이스에 대한 COM 표준 열거형 메서드를 제공합니다. ITRendezvous::EnumerateDefaultDirectories 메서드는 IEnumDirectory에 대한 포인터를 반환합니다.
IEnumDirectoryObject IEnumDirectoryObject 인터페이스는 ITDirectoryObject 인터페이스에 대한 COM 표준 열거형 메서드를 제공합니다. ITDirectory::EnumerateDirectoryObjects 메서드는 IEnumDirectoryObject에 대한 포인터를 반환합니다.
ITDirectory ITDirectory 인터페이스는 특정 디렉터리에 해당하는 Directory 개체에 의해 노출됩니다.
ITDirectoryObject ITDirectoryObject 인터페이스는 ITDirectory 인터페이스를 사용하여 추가 및 삭제할 수 있는 모든 개체에서 지원하는 공통 인터페이스입니다.
ITDirectoryObjectConference ITDirectoryObjectConference 인터페이스는 애플리케이션이 회의 세부 정보를 설정하고 가져올 수 있도록 하는 메서드를 제공합니다. ITDirectoryObjectConference 인터페이스는 ITDirectoryObject에서 QueryInterface를 호출하여 만들어집니다.

ITDirectoryObjectUser ITDirectoryObjectUser 인터페이스는 User 개체에서 지원하는 공통 인터페이스입니다. 이 인터페이스는 ITDirectoryObject에서 QueryInterface를 호출하여 만듭니다.
ITILSConfig ITILSConfig 인터페이스를 사용하면 ILS 디렉터리를 구성할 수 있습니다.
ITRendezvous ITRendezvous 인터페이스는 랑데부 컨트롤에 대한 기본 인터페이스입니다. 애플리케이션은 이 인터페이스에서 COM CoCreateInstance 함수를 호출하여 Rendezvous 개체를 만듭니다.

열거형

DIRECTORY_OBJECT_TYPE DIRECTORY_OBJECT_TYPE 열거형은 디렉터리 개체가 회의인지 사용자인지를 설명하는 설명자입니다.
DIRECTORY_TYPE DIRECTORY_TYPE 열거형은 디렉터리 서버의 유형을 나타내는 데 사용됩니다.
RND_ADVERTISING_SCOPE RND_ADVERTISING_SCOPE 열거형의 멤버는 회의 공지 사항의 배포 범위를 지정합니다. 값은 ITDirectoryObjectConference 인터페이스의 advertising scope 속성에 해당합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

rendezvoussession.h 헤더

아티클2023. 08. 25.

이 헤더는 원격 지원에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 지원

rendezvoussession.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

DRendezvousSessionEvents IRendezvousSession 개체에서 생성되거나 수신되는 이벤트를 처리합니다.
IRendezvousApplication IM(인스턴트 메시징) 애플리케이션에서 원격 지원 세션을 만드는 데 사용하는 메서드를 노출합니다.
IRendezvousSession 세션에 대한 데이터를 보내고 종료할 수 있는 메서드를 노출합니다.

열거형

RENDEZVOUS_SESSION_FLAGS 세션 초대에 사용할 수 있는 플래그 목록을 제공합니다.
RENDEZVOUS_SESSION_STATE 세션 초대의 가능한 상태 코드 목록을 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

resapi.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows 클러스터링에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 클러스터링

resapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CloseClusterCryptProvider CSP(암호화 서비스 공급자)에 대한 핸들을 닫습니다. PCLOSE_CLUSTER_CRYPT_PROVIDER 형식은 이 함수에 대한 포인터를 정의합니다.
CLRES_V1_FUNCTION_TABLE Resource API 버전 1.0에 대한 함수 테이블을 초기화합니다.
CLRES_V2_FUNCTION_TABLE_SET Resource API 버전 2.0에 대한 함수 테이블을 초기화합니다.
CLRES_V3_FUNCTION_TABLE_SET Resource API 버전 3.0에 대한 함수 테이블을 초기화합니다.
CLRES_V4_FUNCTION_TABLE_SET Resource API 버전 4.0에 대한 함수 테이블을 초기화합니다.
ClusAddClusterHealthFault ClusAddClusterHealthFault 함수는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부인 상태 오류 유틸리티 함수입니다.
ClusGetClusterHealthFaults ClusGetClusterHealthFaults 함수는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부인 상태 오류 유틸리티 함수입니다.
ClusRemoveClusterHealthFault

ClusRemoveClusterHealthFault 함수는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부인 상태 오류 유틸리티 함수입니다.
ClusterClearBackupStateForSharedVolume 클러스터 공유 볼륨의 백업 상태를 지웁니다.
ClusterDecrypt CSP(암호화 서비스 공급자)에 대한 검사점 데이터 암호를 해독합니다.
ClusterEncrypt CSP(암호화 서비스 공급자)에 대한 검사점 데이터를 암호화합니다.
ClusterGetVolumeNameForVolumeMountPoint ClusterGetVolumeNameForVolumeMountPoint를 변경하거나 사용할 수 없습니다. 대신 GetVolumeNameForVolumeMountPoint를 사용합니다.
ClusterGetVolumePathName ClusterGetVolumePathName을 변경하거나 사용할 수 없습니다. 대신 GetVolumePathName을 사용합니다.
ClusterIsPathOnSharedVolume 경로가 클러스터 공유 볼륨에 있는지 여부를 확인합니다.
ClusterPrepareSharedVolumeForBackup ClusterPrepareSharedVolumeForBackup을 변경하거나 사용할 수 없습니다.
ClusWorkerCheckTerminate 작업자 스레드가 가능한 한 빨리 종료되어야 하는지 여부를 결정합니다. PCLUSAPIClusWorkerCheckTerminate 형식은 이 함수에 대한 포인터를 정의합니다.
ClusWorkerCreate 작업자 스레드를 만듭니다. PCLUSAPI_CLUS_WORKER_CREATE 형식은 이 함수에 대한 포인터를 정의합니다.
ClusWorkersTerminate 여러 작업자 스레드가 지정된 시간 제한까지 종료될 때까지 기다립니다.
ClusWorkerTerminate

작업자 스레드가 지정된 시간 제한까지 종료될 때까지 기다립니다. (ClusWorkerTerminate)
ClusWorkerTerminateEx 작업자 스레드가 지정된 시간 제한까지 종료될 때까지 기다립니다. (ClusWorkerTerminateEx)
FreeClusterCrypt FreeClusterCrypt 함수는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부인 상태 오류 유틸리티 함수입니다.
FreeClusterHealthFault FreeClusterHealthFault 함수는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부인 상태 오류 유틸리티 함수입니다.
FreeClusterHealthFaultArray FreeClusterHealthFaultArray 함수는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부인 상태 오류 유틸리티 함수입니다.
InitializeClusterHealthFault InitializeClusterHealthFault 함수는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부인 상태 오류 유틸리티 함수입니다.
InitializeClusterHealthFaultArray InitializeClusterHealthFaultArray 함수는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부인 상태 오류 유틸리티 함수입니다.
OpenClusterCryptProvider 클러스터 리소스에 대한 검사점 데이터의 암호화를 관리하기 위해 CSP(암호화 서비스 공급자)에 대한 핸들을 엽니다. POPEN_CLUSTER_CRYPT_PROVIDER 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilAddUnknownProperties 클러스터 데이터베이스에서 알 수 없는 속성 집합을 검색하고 속성 목록의 끝에 추가합니다.
ResUtilCreateDirectoryTree 이미 존재하는 디렉터리를 건너뛰고 경로에 지정된 모든 디렉터리를 만듭니다. PRESUTIL_CREATE_DIRECTORY_TREE 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilDupParameterBlock 한 매개 변수 블록에서 다른 매개 변수 블록으로 데이터의 멤버 단위 복사본을 수행합니다.

ResUtilDupString null로 종료된 유니코드 문자열을 복제합니다.
ResUtilEnumPrivateProperties 클러스터 개체의 프라이빗 속성 이름을 검색합니다. PRESUTIL_ENUM_PRIVATE_PROPERTIES 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilEnumProperties 클러스터 개체의 속성 이름을 열거합니다. PRESUTIL_ENUM_PROPERTIES 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilEnumResources 로컬 클러스터의 모든 리소스를 열거하고 각 리소스에 대해 사용자 정의 작업을 시작합니다. PRESUTIL_ENUM_RESOURCES 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilEnumResourcesEx 지정된 클러스터의 모든 리소스를 열거하고 각 리소스에 대해 사용자 정의 작업을 시작합니다. PRESUTIL_ENUM_RESOURCES_EX 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilEnumResourcesEx2 지정된 클러스터의 모든 리소스를 열거하고 각 리소스에 대해 사용자 정의 작업을 시작합니다. PRESUTIL_ENUM_RESOURCES_EX2 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilExpandEnvironmentStrings 환경 변수에 대한 확장되지 않은 참조가 포함된 문자열을 확장합니다. PRESUTIL_EXPAND_ENVIRONMENT_STRINGS 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilFindBinaryProperty 속성 목록에서 지정된 이진 속성을 찾고 속성 값을 반환할 수도 있습니다. PRESUTIL_FIND_BINARY_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilFindDependentDiskResourceDriveLetter 리소스의 실제 디스크 종속성과 연결된 드라이브 문자를 검색합니다. PRESUTIL_FIND_DEPENDENT_DISK_RESOURCE_DRIVE_LETTER 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilFindDwordProperty 속성 목록에서 서명되지 않은 long 속성 값을 찾습니다. PRESUTIL_FIND_DWORD_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.

ResUtilFindExpandedSzProperty 속성 목록에서 확장된 문자열 속성 값을 찾습니다. PRESUTIL_FIND_EXPANDED_SZ_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilFindExpandSzProperty 속성 목록에서 확장 가능한 문자열 속성을 찾습니다. PRESUTIL_FIND_EXPAND_SZ_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilFindFileTimeProperty TBD. PRESUTIL_FIND_FILETIME_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilFindLongProperty 속성 목록에서 서명된 long 속성 값을 찾습니다. PRESUTIL_FIND_LONG_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilFindMultiSzProperty 속성 목록에서 여러 문자열 속성을 찾습니다. PRESUTIL_FIND_MULTI_SZ_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilFindSzProperty 속성 목록에서 문자열 속성을 찾습니다. PRESUTIL_FIND_SZ_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilFindULargeIntegerProperty 속성 목록에서 큰 정수 속성 값을 가져옵니다. PRESUTIL_FIND_ULARGEINTEGER_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilFreeEnvironment ResUtilGetEnvironmentWithNetName을 사용하여 만든 환경 변수 블록을 삭제합니다. PRESUTIL_FREE_ENVIRONMENT 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilFreeParameterBlock ResUtilDupParameterBlock에 의해 매개 변수 블록에 할당된 메모리의 할당을 취소합니다.
ResUtilGetAllProperties 클러스터 개체에 대한 모든 기본 및 알 수 없는 속성을 포함하는 속성 목록을 반환합니다. PRESUTIL_GET_ALL_PROPERTIES 형식은 이 함수에 대한 포인터를 정의합니다.

ResUtilGetBinaryProperty 속성 목록에서 이진 속성을 검색하고 목록의 다음 속성에 대한 포인터를 진행합니다. PRESUTIL_GET_BINARY_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetBinaryValue 클러스터 데이터베이스에서 이진 값을 반환합니다.
ResUtilGetClusterRoleState 특정 역할이 클러스터에 할당되었는지 여부를 결정합니다.
ResUtilGetCoreClusterResources 핵심 네트워크 이름, IP 주소 및 쿼럼 리소스에 대한 핸들을 반환합니다. PRESUTIL_GET_CORE_CLUSTER_RESOURCES 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetCoreClusterResourcesEx 코어, 네트워크 이름, IP 주소 및 쿼럼 리소스에 대한 핸들을 반환합니다. PRESUTIL_GET_CORE_CLUSTER_RESOURCES_EX 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetDwordProperty 속성 목록에서 DWORD 속성을 검색하고 목록의 다음 속성에 대한 포인터를 진행합니다. PRESUTIL_GET_DWORD_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetDwordValue 클러스터 데이터베이스에서 숫자 값을 반환합니다.
ResUtilGetEnvironmentWithNetName 리소스가 클러스터 네트워크 이름을 사용하여 위치를 식별하도록 리소스에 대한 환경 데이터를 조정합니다.
ResUtilGetExpandSzValue 클러스터 데이터베이스에서 확장 가능한 문자열 값을 반환합니다.
ResUtilGetFileTimeProperty TBD. PRESUTIL_GET_FILETIME_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetLongProperty TBD. PRESUTIL_GET_LONG_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.

ResUtilGetMultiSzProperty 속성 목록에서 여러 문자열 속성을 검색하고 목록의 다음 속성에 대한 포인터를 진행합니다. PRESUTIL_GET_MULTI_SZ_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetMultiSzValue 클러스터 데이터베이스에서 여러 문자열 값을 반환합니다.
ResUtilGetPrivateProperties 클러스터 개체에 대한 프라이빗 속성을 반환합니다. PRESUTIL_GET_PRIVATE_PROPERTIES 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetProperties 클러스터 데이터베이스에서 속성 테이블에 지정된 속성을 검색하고 속성 목록에 반환합니다. PRESUTIL_GET_PROPERTIES 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetPropertiesToParameterBlock 클러스터 데이터베이스에서 속성 테이블에 지정된 속성을 검색하고 매개 변수 블록에 반환합니다.
ResUtilGetProperty 클러스터 데이터베이스에서 지정된 속성을 반환합니다. PRESUTIL_GET_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetPropertyFormats 지정된 속성 집합의 형식을 설명하는 속성 형식 목록을 반환합니다. PRESUTIL_GET_PROPERTY_FORMATS 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetPropertySize 지정된 속성에 필요한 총 바이트 수를 반환합니다.
ResUtilGetQwordValue ResUtilGetQwordValue 함수는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부인 상태 오류 유틸리티 함수입니다.
ResUtilGetResourceDependency 지정된 리소스의 종속성을 열거하고 지정된 형식의 종속성에 대한 핸들을 반환합니다. PRESUTIL_GET_RESOURCE_DEPENDENCY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetResourceDependencyByClass

지정된 클러스터에서 지정된 리소스의 종속성을 열거하고 지정된 리소스 클래스와 일치하는 종속성에 대한 핸들을 반환합니다. PRESUTIL_GET_RESOURCE_DEPENDENCY_BY_CLASS 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetResourceDependencyByClassEx 지정된 클러스터에서 지정된 리소스의 종속성을 열거하고 지정된 리소스 클래스와 일치하는 종속성에 대한 핸들을 반환합니다. PRESUTIL_GET_RESOURCE_DEPENDENCY_BY_CLASS_EX 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetResourceDependencyByName 지정된 클러스터에서 지정된 리소스의 종속성을 열거하고 지정된 형식의 종속성에 대한 핸들을 반환합니다. PRESUTIL_GET_RESOURCE_DEPENDENCY_BY_NAME 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetResourceDependencyByNameEx 지정된 클러스터에서 지정된 리소스의 종속성을 열거하고 지정된 형식의 종속성에 대한 핸들을 반환합니다. PRESUTIL_GET_RESOURCE_DEPENDENCY_BY_NAME_EX 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetResourceDependencyEx 지정된 리소스의 종속성을 열거하고 지정된 형식의 종속성에 대한 핸들을 반환합니다. PRESUTIL_GET_RESOURCE_DEPENDENCY_EX 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetResourceDependentIPAddressProps 지정된 리소스에 대해 찾은 첫 번째 IP 주소 종속성의 프라이빗 속성을 검색합니다. PRESUTIL_GET_RESOURCE_DEPENDENTIP_ADDRESS_PROPS 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetResourceName 리소스의 이름을 반환합니다. PRESUTIL_GET_RESOURCE_NAME 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetResourceNameDependency 로컬 클러스터에서 지정된 리소스의 종속성을 열거하고 지정된 리소스 유형의 종속성에 대한 핸들을 반환합니다. PRESUTIL_GET_RESOURCE_NAME_DEPENDENCY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetResourceNameDependencyEx 로컬 클러스터에서 지정된 리소스의 종속성을 열거하고 지정된 리소스 유형의 종속성에 대한 핸들을 반환합니다. PRESUTIL_GET_RESOURCE_NAME_DEPENDENCY_EX 형식은 이 함수에 대한 포인터를 정의합니다.

ResUtilGetSzProperty 속성 목록에서 문자열 속성을 검색하고 목록의 다음 속성에 대한 포인터를 진행합니다. PRESUTIL_GET_SZ_PROPERTY 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilGetSzValue 클러스터 데이터베이스에서 문자열 값을 반환합니다.
ResUtilInitializeResourceStatus RESOURCE_STATUS 구조를 초기화합니다.
ResUtilIsPathValid 경로가 구문적으로 유효한지 확인합니다.
ResUtilIsResourceClassEqual 지정된 리소스의 리소스 클래스가 지정된 리소스 클래스와 같은지 테스트합니다. PRESUTIL_IS_RESOURCE_CLASS_EQUAL 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilLeftPaxosIsLessThanRight 지정된 Paxos 태그에 지정된 다른 Paxos 태그보다 이전 클러스터 구성 정보가 포함되어 있는지 여부를 나타냅니다.
ResUtilPaxosComparer 두 Paxos 태그를 비교하고 동일한 값이 있는지 여부를 나타냅니다.
ResUtilPropertyListFromParameterBlock 속성 테이블 및 매개 변수 블록에서 속성 목록을 생성합니다.
ResUtilRemoveResourceServiceEnvironment 서비스에서 환경 데이터를 제거합니다. 이 함수는 리소스 DLL에서 호출해야 합니다. PRESUTIL_REMOVE_RESOURCE_SERVICE_ENVIRONMENT 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilResourcesEqual 두 개의 리소스 핸들이 동일한 리소스를 나타내는지 테스트합니다. PRESUTIL_RESOURCES_EQUAL 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilResourceTypesEqual

리소스 종류가 지정된 리소스의 리소스 종류 이름과 일치하는지 테스트합니다. PRESUTIL_RESOURCE_TYPES_EQUAL 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilSetBinaryValue 클러스터 데이터베이스에서 이진 값을 설정합니다.
ResUtilSetDwordValue 클러스터 데이터베이스의 숫자 값을 설정합니다. PRESUTIL_SET_DWORD_VALUE 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilSetExpandSzValue 클러스터 데이터베이스에서 확장 가능한 문자열 값을 설정합니다. PRESUTIL_SET_EXPAND_SZ_VALUE 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilSetMultiSzValue 클러스터 데이터베이스에서 여러 문자열 값을 설정합니다. PRESUTIL_SET_MULTI_SZ_VALUE 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilSetPrivatePropertyList 클러스터 개체의 프라이빗 속성을 설정합니다.
ResUtilSetPropertyParameterBlock 매개 변수 블록에서 클러스터 데이터베이스의 속성을 설정합니다. (ResUtilSetPropertyParameterBlock)
ResUtilSetPropertyParameterBlockEx 매개 변수 블록에서 클러스터 데이터베이스의 속성을 설정합니다. (ResUtilSetPropertyParameterBlockEx)
ResUtilSetPropertyTable 속성 테이블의 속성 목록을 기반으로 클러스터 데이터베이스의 속성을 설정합니다.
ResUtilSetPropertyTableEx 속성 테이블의 속성 목록을 기반으로 클러스터 데이터베이스의 속성을 설정합니다.
ResUtilSetQwordValue TBD. PRESUTIL_SET_QWORD_VALUE 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilSetResourceServiceEnvironment

서비스에서 클러스터 네트워크 이름을 사용하여 해당 위치를 식별하도록 서비스의 환경 데이터를 조정합니다. 이 함수는 리소스 DLL에서 호출해야 합니다. PRESUTIL_SET_RESOURCE_SERVICE_ENVIRONMENT 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilSetResourceServiceStartParameters 지정된 서비스의 시작 매개 변수를 조정하여 클러스터 리소스로 올바르게 작동합니다. 리소스 DLL에서 호출해야 합니다. PRESUTIL_SET_RESOURCE_SERVICE_START_PARAMETERS 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilSetResourceServiceStartParametersEx 지정된 서비스의 시작 매개 변수를 조정하여 클러스터 리소스로 올바르게 작동합니다. 리소스 DLL에서 호출해야 합니다. PRESUTIL_SET_RESOURCE_SERVICE_START_PARAMETERS_EX 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilSetSzValue 클러스터 데이터베이스에서 문자열 값을 설정합니다. PRESUTIL_SET_SZ_VALUE 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilSetUnknownProperties 클러스터 데이터베이스에 클러스터 개체의 알 수 없는 속성을 저장합니다.
ResUtilSetValueEx 클러스터 데이터베이스의 값을 설정합니다.
ResUtilStartResourceService 서비스를 시작합니다. PRESUTIL_START_RESOURCE_SERVICE 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilStopResourceService 명명된 서비스를 중지합니다. PRESUTIL_STOP_RESOURCE_SERVICE 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilStopService 핸들로 식별된 서비스를 중지합니다. PRESUTIL_STOP_SERVICE 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilTerminateServiceProcessFromResDll 리소스 DLL에 의해 클러스터 리소스로 관리되는 서비스의 프로세스를 종료하려고 시도합니다. PRESUTIL_TERMINATE_SERVICE_PROCESS_FROM_RES_DLL 형식은 이 함수에 대한 포인터를 정의합니다.

ResUtilVerifyPrivatePropertyList 속성 목록의 형식이 올바른지 확인합니다.
ResUtilVerifyPropertyTable 속성 테이블을 사용하여 속성 목록의 형식이 올바르게 지정되었는지 확인합니다.
ResUtilVerifyResourceService 명명된 서비스가 시작되거나 현재 실행 중인지 확인합니다. PRESUTIL_VERIFY_RESOURCE_SERVICE 형식은 이 함수에 대한 포인터를 정의합니다.
ResUtilVerifyService 핸들로 식별된 서비스가 시작 중인지 현재 실행 중인지 확인합니다. PRESUTIL_VERIFY_SERVICE 형식은 이 함수에 대한 포인터를 정의합니다.

콜백 함수

LPRESOURCE_CALLBACK LPRESOURCE_CALLBACK 콜백 함수는 다른 함수에서 액세스할 수 있으며 첫 번째 함수가 완료된 후에 호출됩니다. (LPRESOURCE_CALLBACK)
LPRESOURCE_CALLBACK_EX LPRESOURCE_CALLBACK_EX 콜백 함수는 다른 함수에서 액세스할 수 있으며 첫 번째 함수가 완료된 후 호출됩니다. (LPRESOURCE_CALLBACK_EX)
PARBITRATE_ROUTINE 노드가 쿼럼 리소스의 소유권을 다시 얻을 수 있도록 허용합니다.
PBEGIN_RESCALL_AS_USER_ROUTINE PBEGIN_RESCALL_AS_USER_ROUTINE 형식은 이 콜백 함수에 대한 포인터를 정의합니다.
PBEGIN_RESCALL_ROUTINE 리소스 제어 코드에 대한 호출을 시작합니다. PBEGIN_RESCALL_ROUTINE 형식은 이 콜백 함수에 대한 포인터를 정의합니다.
PBEGIN_RESTYPECALL_AS_USER_ROUTINE

PBEGIN_RESTYPECALL_AS_USER_ROUTINE 형식은 이 콜백 함수에 대한 포인터를 정의합니다.
PBEGIN_RESTYPECALL_ROUTINE 리소스 제어 코드에 대한 호출을 시작합니다. PBEGIN_RESTYPECALL_ROUTINE 형식은 이 콜백 함수에 대한 포인터를 정의합니다.
PCANCEL_ROUTINE 리소스에 대한 작업을 취소합니다.
PCHANGE_RES_TYPE_PROCESS_FOR_DUMPS PCHANGE_RES_TYPE_PROCESS_FOR_DUMPS 형식은 이 함수에 대한 포인터를 정의합니다.
PCHANGE_RESOURCE_PROCESS_FOR_DUMPS PCHANGE_RESOURCE_PROCESS_FOR_DUMPS 형식은 이 함수에 대한 포인터를 정의합니다.
PCLOSE_ROUTINE 리소스를 닫습니다.
PCLUSAPI_CLUS_WORKER_TERMINATE 작업자 스레드를 종료합니다. PCLUSAPI_CLUS_WORKER_TERMINATE 형식은 이 함수에 대한 포인터를 정의합니다.
PEND_CONTROL_CALL 리소스 제어 코드 작업이 완료되면 호출됩니다. PEND_CONTROL_CALL 형식은 이 함수에 대한 포인터를 정의합니다.
PEND_TYPE_CONTROL_CALL 리소스 종류 제어 코드 작업이 완료되면 호출됩니다. PEND_TYPE_CONTROL_CALL 형식은 이 함수에 대한 포인터를 정의합니다.
PEXTEND_RES_CONTROL_CALL 리소스 제어 코드 호출에 대한 시간 제한을 확장합니다. PEXTEND_RES_CONTROL_CALL 형식은 이 함수에 대한 포인터를 정의합니다.
PEXTEND_RES_TYPE_CONTROL_CALL 리소스 종류 제어 코드에 대한 호출에 대한 시간 제한을 확장합니다. PEXTEND_RES_TYPE_CONTROL_CALL 형식은 이 함수에 대한 포인터를 정의합니다.
PIS_ALIVE_ROUTINE

리소스를 사용할 수 있는지 여부를 결정합니다.
PLOG_EVENT_ROUTINE 클러스터 로그에 이벤트를 기록합니다.
PLOOKS_ALIVE_ROUTINE 리소스를 사용할 수 있는 것으로 보이는지 여부를 결정합니다.
POFFLINE_ROUTINE POFFLINE_ROUTINE 콜백 함수는 정리 처리가 완료된 후 리소스를 사용할 수 없음으로 표시합니다.
POFFLINE_V2_ROUTINE POFFLINE_V2_ROUTINE 콜백 함수는 정리 처리가 완료된 후 리소스를 사용할 수 없음으로 표시합니다.
PONLINE_ROUTINE PONLINE_ROUTINE 콜백 함수는 리소스를 사용할 수 있는 것으로 표시합니다. (PONLINE_ROUTINE)
PONLINE_V2_ROUTINE PONLINE_V2_ROUTINE 콜백 함수는 리소스를 사용할 수 있는 것으로 표시합니다. (PONLINE_V2_ROUTINE)
POPEN_ROUTINE POPEN_ROUTINE 콜백 함수는 리소스를 엽니다. POPEN_ROUTINE 형식은 이 함수에 대한 포인터를 정의합니다.
POPEN_V2_ROUTINE POPEN_V2_ROUTINE 콜백 함수는 리소스를 엽니다. POPEN_V2_ROUTINE 형식은 이 함수에 대한 포인터를 정의합니다.
PQUORUM_RESOURCE_LOST 쿼럼 리소스의 제어가 손실되었을 때 호출됩니다.
PRAISE_RES_TYPE_NOTIFICATION TBD. PRAISE_RES_TYPE_NOTIFICATION 형식은 이 함수에 대한 포인터입니다.
PRELEASE_ROUTINE

중재에서 쿼럼 리소스를 해제합니다.
PRESOURCE_CONTROL_ROUTINE 리소스에 적용되는 작업을 수행합니다.
PRESOURCE_TYPE_CONTROL_ROUTINE 리소스 종류에 적용되는 작업을 수행합니다.
PSET_INTERNAL_STATE 리소스의 내부 상태를 설정합니다.
PSET_RESOURCE_INMEMORY_NODELOCAL_PROPERTIES_ROUTINE PSET_RESOURCE_INMEMORY_NODELOCAL_PROPERTIES_ROUTINE 콜백 함수는 리소스 종류 제어 코드 작업이 완료되면 호출됩니다.
PSET_RESOURCE_LOCKED_MODE_ROUTINE 잠긴 모드가 리소스에 대해 구성되었음을 보고합니다.
PSET_RESOURCE_STATUS_ROUTINE PSET_RESOURCE_STATUS_ROUTINE 콜백 함수는 리소스의 상태 업데이트하기 위해 호출됩니다.
PSET_RESOURCE_STATUS_ROUTINE_EX PSET_RESOURCE_STATUS_ROUTINE_EX 콜백 함수는 리소스의 상태 업데이트하기 위해 호출됩니다. (PSET_RESOURCE_STATUS_ROUTINE_EX)
PSIGNAL_FAILURE_ROUTINE 리소스 instance 오류가 있었다고 보고합니다. PSIGNAL_FAILURE_ROUTINE 형식은 이 함수에 대한 포인터를 정의합니다.
PSTARTUP_EX_ROUTINE 함수 테이블과 버전 번호가 포함된 구조를 반환하여 리소스 DLL을 로드합니다.
PSTARTUP_ROUTINE 함수 테이블과 버전 번호를 포함하는 구조를 반환하여 리소스 DLL을 로드합니다.
PTERMINATE_ROUTINE 즉시 정리 처리가 완료될 때까지 기다리지 않고 리소스를 사용할 수 없음으로 표시합니다.

PWORKER_START_ROUTINE 지정된 콜백 루틴을 사용하여 작업자 스레드를 초기화합니다. PWORKER_START_ROUTINE 형식은 이 함수에 대한 포인터를 정의합니다.

구조체

CLRES_CALLBACK_FUNCTION_TABLE StartupEx 콜백 함수에 대한 함수 테이블을 나타냅니다.
CLRES_FUNCTION_TABLE 모든 버전의 Resource API에 대한 함수 테이블을 설명합니다.
CLRES_V1_FUNCTIONS 시작을 제외한 모든 Resource API 버전 1.0 진입점에 대한 포인터를 포함합니다.
CLRES_V2_FUNCTIONS StartupEx를 제외한 모든 Resource API 버전 2.0 진입점에 대한 포인터를 포함합니다.
CLRES_V3_FUNCTIONS StartupEx를 제외한 모든 Resource API 버전 3.0 진입점에 대한 포인터를 포함합니다.
CLRES_V4_FUNCTIONS StartupEx를 제외한 모든 Resource API 버전 4.0 진입점에 대한 포인터를 포함합니다.
CLUS_WORKER 작업자 스레드에 대한 정보를 포함합니다.
CLUSTER_HEALTH_FAULT CLUSTER_HEALTH_FAULT 구조는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부입니다.
CLUSTER_HEALTH_FAULT_ARRAY CLUSTER_HEALTH_FAULT_ARRAY 구조는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부입니다.

GET_OPERATION_CONTEXT_PARAMS CLUSCTL_RESOURCE_GET_OPERATION_CONTEXT 제어 코드의 입력으로 사용되는 컨텍스트 매개 변수를 나타냅니다.
MONITOR_STATE MONITOR_STATE 구조는 Windows 클러스터링에서 사용하는 resapi.h 헤더의 일부입니다.
PaxosTagCStruct 클러스터에서 파일 공유 감시를 사용할 때 노드의 클러스터 구성 버전에 대한 정보를 저장하는 클러스터 노드의 Paxos 태그 값을 포함합니다.
POST_UPGRADE_VERSION_INFO 클러스터 서비스에 대한 업그레이드 후 상태 정보를 나타냅니다.
RESOURCE_STATUS 온라인 상태가 되거나 오프라인으로 전환되는 리소스에 대한 정보를 포함합니다. 이 구조체는 콜백 함수 SetResourceStatus에 대한 매개 변수로 사용됩니다.
RESOURCE_STATUS_EX 온라인 상태가 되거나 오프라인으로 전환되는 리소스에 대한 정보를 포함합니다. 이 구조체는 콜백 함수 SetResourceStatusEx에 대한 매개 변수로 사용됩니다.
RESUTIL_FILETIME_DATA FILETIME에 허용되는 기본값, 최대값 및 최소값을 설명합니다.
RESUTIL_LARGEINT_DATA 서명된 큰 정수에 허용되는 기본값, 최대값 및 최소값을 설명합니다.
RESUTIL_PROPERTY_ITEM 클러스터 개체 속성에 대한 정보를 포함합니다. RESUTIL_PROPERTY_ITEM 구조체의 배열은 속성 작업에 사용할 수 있는 속성 테이블을 형성합니다.
RESUTIL_ULARGEINT_DATA 부호 없는 큰 정수에 허용되는 기본값, 최대값 및 최소값을 설명합니다.
WitnessTagHelper PaxosTagCStruct 구조체의 유효성을 검사하는 데 사용되는 정보를 포함합니다.

WitnessTagUpdateHelper PaxosTagCStruct 구조체를 업데이트하고 유효성을 검사하는 데 사용되는 정보를 포함합니다.

열거형

CLUSTER_RESOURCE_APPLICATION_STATE 리소스 애플리케이션 상태를 열거합니다.
CLUSTER_ROLE 표준 클러스터 역할의 이름을 포함합니다.
CLUSTER_ROLE_STATE ResUtilGetClusterRoleState 함수의 잠재적 반환 값을 정의합니다.
FAILURE_TYPE 클러스터 리소스에 대한 오류 유형을 정의합니다.
LOG_LEVEL LogEvent 콜백 함수에 전달된 로그 이벤트의 심각도를 나타냅니다.
RESDLL_CONTEXT_OPERATION_TYPE GET_OPERATION_CONTEXT_PARAMS 구조체에 대한 다양한 유형의 컨텍스트 작업을 지정합니다.
RESOURCE_EXIT_STATE 리소스의 가능한 종료 상태를 열거합니다.
RESOURCE_MONITOR_STATE RESOURCE_MONITOR_STATE 열거형은 애플리케이션이 실행 중일 때 발생하는 계측 작업의 보기를 제공합니다.
VM_RESDLL_CONTEXT 수행할 가상 머신에 대한 작업을 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

resourceindexer.h 헤더

아티클2023. 08. 25.

이 헤더는 메뉴 및 기타 리소스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 메뉴 및 기타 리소스

resourceindexer.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CreateResourceIndexer 프로젝트 파일의 루트 및 확장 DLL의 지정된 경로에 대한 새 리소스 인덱서를 만듭니다.
DestroyIndexedResults IndexFilePath 메서드가 반환한 매개 변수를 해제합니다.
DestroyResourceIndexer 지정된 리소스 인덱서와 연결된 계산 리소스를 해제합니다.
IndexFilePath 파일 및 폴더 명명 규칙에 대한 파일 경로를 인덱싱합니다.

구조체

IndexedResourceQualifier 리소스가 적절한 컨텍스트를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

restartmanager.h 헤더

아티클2023. 08. 25.

이 헤더는 Restart Manager에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 관리자 다시 시작

restartmanager.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

RmAddFilter 애플리케이션 또는 서비스에 적용되는 종료 또는 다시 시작 작업을 수정합니다.
RmCancelCurrentTask 현재 RmShutdown 또는 RmRestart 작업을 취소합니다. 이 함수는 RmStartSession 함수를 호출하여 세션을 시작한 애플리케이션에서 호출해야 합니다.
RmEndSession 다시 시작 관리자 세션을 종료합니다.
RmGetFilterList RmAddFilter 함수에서 이미 적용한 종료 및 다시 시작 작업에 대한 수정 사항을 나열합니다.
RmGetList 현재 다시 시작 관리자 세션에 등록된 리소스를 사용하는 모든 애플리케이션 및 서비스의 목록을 가져옵니다.
RmJoinSession 보조 설치 관리자를 기존 Restart Manager 세션에 조인합니다.
RmRegisterResources 리소스를 다시 시작 관리자 세션에 등록합니다.
RmRemoveFilter RmAddFilter 함수를 사용하여 적용된 종료 또는 다시 시작 작업에 대한 수정 사항을 제거합니다.

RmRestart RmShutdown 함수에 의해 종료되고 RegisterApplicationRestart 함수를 사용하여 다시 시작되도록 등록된 애플리케이션 및 서비스를 다시 시작합니다.
RmShutdown 애플리케이션의 종료를 시작합니다.
RmStartSession 새 다시 시작 관리자 세션을 시작합니다.

콜백 함수

RM_WRITE_STATUS_CALLBACK RM_WRITE_STATUS_CALLBACK 함수는 다시 시작 관리자를 제어하는 사용자 인터페이스에서 구현할 수 있습니다.

구조체

RM_FILTER_INFO 다시 시작 또는 종료 작업을 수정하는 방법에 대한 정보를 포함합니다.
RM_PROCESS_INFO 다시 시작 관리자에 등록할 애플리케이션에 대해 설명합니다.
RM_UNIQUE_PROCESS PID 및 프로세스가 시작된 시간을 기준으로 프로세스를 고유하게 식별합니다.

열거형

RM_APP_STATUS Restart Manager에서 수행하는 애플리케이션의 현재 상태 설명합니다.
RM_APP_TYPE RM_PROCESS_INFO 구조체에서 설명하는 애플리케이션 유형을 지정합니다.
RM_FILTER_ACTION 다시 시작 또는 종료 작업에 적용되는 수정 유형을 지정합니다.
RM_FILTER_TRIGGER 애플리케이션 또는 서비스에 대한 다시 시작 또는 종료 작업을 설명합니다.
RM_REBOOT_REASON 시스템을 다시 시작해야 하는 이유를 설명합니다.
RM_SHUTDOWN_TYPE 애플리케이션의 종료를 구성합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

restrictederrorinfo.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

restrictederrorinfo.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ILanguageExceptionErrorInfo RoOriginateLanguageException 호출을 사용하여 오류 정보에 저장된 IUnknown 포인터를 검색할 수 있습니다.
ILanguageExceptionErrorInfo2 언어 프로젝션이 ILanguageExceptionErrorInfo와 같이 오류 정보를 제공하고 검색할 수 있도록 하며, 언어 경계를 넘어 작업할 때 추가적인 이점이 있습니다.
ILanguageExceptionStackBackTrace 프로젝션에서 해당 예외에 대한 사용자 지정 스택 추적을 제공할 수 있습니다.
ILanguageExceptionTransform 다른 예외를 catch하는 catch 처리기의 컨텍스트에서 throw되는 예외의 모든 컨텍스트를 시스템에서 언어 프로젝션을 사용할 수 있도록 허용합니다.
IRestrictedErrorInfo 제한된 오류 정보를 포함하여 오류의 세부 정보를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

richedit.h 헤더

이 헤더는 Windows 컨트롤에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 컨트롤

richedit.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

HyphenateProc HyphenateProc 함수는 애플리케이션이며 EM_SETHYPHENATEINFO 메시지와 함께 사용되는 정의된 콜백 함수입니다. Microsoft Rich Edit 컨트롤에서 하이픈 넣기를 수행하는 방법을 결정합니다.

콜백 함수

테이블 확장

AutoCorrectProc AutoCorrectProc 함수는 EM_SETAUTOCORRECTPROC 메시지와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
EDITSTREAMCALLBACK EditStreamCallback 함수는 EM_STREAMIN 및 EM_STREAMOUT 메시지와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
EDITWORDBREAKPROCEX EditWordBreakProcEx 함수는 EM_SETWORDBREAKPROCEX 메시지와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.

구조

테이블 확장

BIDIOPTIONS 서식 있는 편집 컨트롤에 대한 양방향 정보를 포함합니다. 이 구조는 EM_GETBIDIOPTIONS 및 EM_SETBIDIOPTIONS 메시지에서 컨트롤에 대한 양방향 정보를 가져와서 설정하는 데 사용됩니다.
CHARFORMAT2A ANSI(CHARFORMAT2A) 구조체(richedit.h)에는 서식 있는 편집 컨트롤의 문자 서식에 대한 정보가 포함되어 있습니다.
CHARFORMAT2A 서식 있는 편집 컨트롤의 문자 서식에 대한 정보를 포함합니다. (CHARFORMAT2A)
CHARFORMAT2W CHARFORMAT2W(유니코드) 구조체(richedit.h)에는 서식 있는 편집 컨트롤의 문자 서식에 대한 정보가 포함되어 있습니다.
CHARFORMAT2W 서식 있는 편집 컨트롤의 문자 서식에 대한 정보를 포함합니다. (CHARFORMAT2W)
CHARFORMATA 서식 있는 편집 컨트롤의 문자 서식에 대한 정보를 포함합니다. (CHARFORMATA)
CHARFORMATW 서식 있는 편집 컨트롤의 문자 서식에 대한 정보를 포함합니다. (CHARFORMATW)
CHARRANGE 서식 있는 편집 컨트롤의 문자 범위를 지정합니다. cpMin 및 cpMax 멤버가 같으면 범위가 비어 있습니다. cpMin이 0이고 cpMax가 -1인 경우 범위에 모든 항목이 포함됩니다.
CLIPBOARDFORMAT 클립보드 형식을 지정합니다. 이 구조는 EN_CLIPFORMAT 알림에 포함됩니다.
COMPCOLOR 컴퍼지션 문자열에 대한 색 설정을 포함합니다.
EDITSTREAM 애플리케이션이 EM_STREAMIN 또는 EM_STREAMOUT 메시지의 서식 있는 편집 컨트롤에 전달하는 정보를 포함합니다. 리치 편집 컨트롤은 정보를 사용하여 데이터 스트림을 컨트롤로 또는 외부로 전송합니다.

ENCORRECTTEXT 수정할 선택한 텍스트에 대한 정보를 포함합니다.
ENDCOMPOSITIONNOTIFY 다양한 편집 컨트롤의 EN_ENDCOMPOSITION 알림 코드에 대한 정보를 포함합니다.
엔드롭파일 EN_DROPFILES 알림 코드와 연결된 정보를 포함합니다. 서식 있는 편집 컨트롤은 WM_DROPFILES 메시지를 받으면 이 알림 코드를 보냅니다.
ENLINK 다양한 편집 컨트롤의 EN_LINK 알림 코드에 대한 정보를 포함합니다.
ENLOWFIRTF Microsoft Rich Edit 컨트롤에서 지원되지 않는 RTF(서식 있는 텍스트 형식) 키워드에 대한 정보를 포함합니다.
ENOLEOPFAILED 실패한 작업에 대한 정보를 포함합니다.
보호됨 EN_PROTECTED 알림 코드와 연결된 정보를 포함합니다. 리치 편집 컨트롤은 사용자가 보호된 텍스트를 편집하려고 할 때 이 알림을 보냅니다.
ENSAVECLIPBOARD 클립보드의 개체 및 텍스트에 대한 정보를 포함합니다.
FINDTEXTA 서식 있는 편집 컨트롤의 검색 작업에 대한 정보를 포함합니다. 이 구조는 EM_FINDTEXT 메시지와 함께 사용됩니다. (ANSI)
FINDTEXTEXA 서식 있는 편집 컨트롤에서 검색할 텍스트에 대한 정보를 포함합니다. 이 구조체는 EM_FINDTEXTEX 메시지와 함께 사용됩니다. (ANSI)
FINDTEXTEXW 서식 있는 편집 컨트롤에서 검색할 텍스트에 대한 정보를 포함합니다. 이 구조체는 EM_FINDTEXTEX 메시지와 함께 사용됩니다. (유니코드)

FINDTEXTW 서식 있는 편집 컨트롤의 검색 작업에 대한 정보를 포함합니다. 이 구조는 EM_FINDTEXT 메시지와 함께 사용됩니다. (유니코드)
FORMATRANGE 서식 있는 편집 컨트롤이 특정 디바이스의 출력 형식을 지정하는 데 사용하는 정보입니다. 이 구조는 EM_FORMATRANGE 메시지와 함께 사용됩니다.
GETCONTEXTMENUEX IRichEditOleCallback::GetContextMenu 메서드에 전달되는 상황에 맞는 메뉴 정보를 포함합니다.
GETTEXTEX 서식 있는 편집 컨트롤에서 텍스트를 가져오는 데 사용되는 정보를 포함합니다. EM_GETTEXTEX 메시지와 함께 사용되는 이 구조체입니다.
GETTEXTLENGTHEX 서식 있는 편집 컨트롤의 텍스트 길이를 계산하는 방법에 대한 정보를 포함합니다. 이 구조체는 EM_GETTEXTLENGTHEX 메시지의 wParam에 전달됩니다.
하이픈ATEINFO Microsoft Rich Edit 컨트롤의 하이픈 넣기에 대한 정보를 포함합니다.
HYPHRESULT Microsoft Rich Edit 컨트롤의 하이픈 넣기 결과에 대한 정보를 포함합니다.
IMECOMPTEXT Microsoft Rich Edit 컨트롤의 IME(입력 메서드 편집기) 컴퍼지션 텍스트에 대한 정보를 포함합니다.
MSGFILTER 키보드 또는 마우스 이벤트에 대한 정보를 포함합니다. 다양한 편집 컨트롤은 이 구조를 EN_MSGFILTER 알림 코드의 일부로 부모 창으로 보내어 부모가 메시지를 변경하거나 처리하지 못하도록 합니다.
NMHDR NMHDR(richedit.h) 구조에는 알림 메시지에 대한 정보가 포함됩니다.
개체 지정 개체 위치 정보를 포함합니다.

PARAFORMAT 서식 있는 편집 컨트롤의 단락 서식 특성에 대한 정보를 포함합니다. (PARAFORMAT)
PARAFORMAT2 서식 있는 편집 컨트롤의 단락 서식 특성에 대한 정보를 포함합니다. P
PARAFORMAT2 서식 있는 편집 컨트롤의 단락 서식 특성에 대한 정보를 포함합니다. (PARAFORMAT2)
문장 부호 서식 있는 편집 컨트롤에 사용되는 문장 부호에 대한 정보를 포함합니다.
REPASTESPECIAL 붙여넣은 개체의 표시 측면이 개체의 내용 또는 개체를 나타내는 아이콘을 기반으로 하는지 여부를 식별하는 정보를 포함합니다.
REQRESIZE 서식 있는 편집 컨트롤의 요청된 크기를 포함합니다. 풍부한 편집 컨트롤은 EN_REQUESTRESIZE 알림 코드의 일부로 이 구조를 부모 창으로 보냅니다.
RICHEDIT_IMAGE_PARAMETERS EM_INSERTIMAGE 메시지에서 삽입할 이미지의 특성을 정의합니다.
SELCHANGE EN_SELCHANGE 알림 코드와 연결된 정보를 포함합니다. 서식 있는 편집 컨트롤은 현재 선택 영역이 변경되면 이 알림을 부모 창으로 보냅니다.
SETTEXTEX 텍스트 설정에 사용할 코드 페이지(있는 경우)를 지정하고, 텍스트가 컨트롤의 모든 텍스트를 바꿀지 아니면 선택 영역만 바꿀지, 실행 취소 상태를 유지할지 여부를 지정합니다. 이 구조는 EM_SETTEXTEX 메시지와 함께 사용됩니다.
TABLECELLPARMS 표 행에 있는 셀의 특성을 정의합니다.
TABLEROWPARMS 테이블의 행 특성을 정의합니다.

TEXTRANGEA 서식 있는 편집 컨트롤의 텍스트 범위입니다. 이 구조는 EM_GETTEXTRANGE 메시지에 의해 채워집니다. lpstrText 멤버가 가리키는 버퍼는 모든 문자와 종료 null 문자를 받을 수 있을 만큼 커야 합니다. (ANSI)
TEXTRANGEW 서식 있는 편집 컨트롤의 텍스트 범위입니다. 이 구조는 EM_GETTEXTRANGE 메시지에 의해 채워집니다. lpstrText 멤버가 가리키는 버퍼는 모든 문자와 종료 null 문자를 받을 수 있을 만큼 커야 합니다. (유니코드)

열거 목록

테이블 확장

KHYPH 서식 있는 편집 컨트롤에서 하이픈 넣기를 수행하는 방법을 지정하는 데 사용되는 값을 포함합니다. HyphenateProc 콜백 함수는 이 열거형 형식을 사용합니다.
TEXTMODE 서식 있는 편집 컨트롤의 텍스트 모드를 나타냅니다. EM_SETTEXTMODE 및 EM_GETTEXTMODE 메시지는 이 열거형 형식을 사용합니다.
UNDONAMEID 실행 취소하거나 다시 실행할 수 있는 서식 있는 편집 컨트롤 작업의 형식을 나타내는 값을 포함합니다. EM_GETREDONAME 및 EM_GETUNDONAME 메시지는 이 열거형 형식을 사용하여 값을 반환합니다.

Last updated on 2025. 10. 08.

richole.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows 컨트롤에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 컨트롤

richole.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IRichEditOle IRichEditOle 인터페이스는 풍부한 편집 컨트롤의 COM(구성 요소 개체 모델) 기능을 노출합니다. 인터페이스는 EM_GETOLEINTERFACE 메시지를 전송하여 가져올 수 있습니다. 이 인터페이스에는 다음과 같은 메서드가 있습니다.
IRichEditOleCallback IRichEditOleCallback 인터페이스는 클라이언트에서 OLE 관련 정보를 검색하기 위해 서식 있는 텍스트 편집 컨트롤에서 사용됩니다.

구조체

REOBJECT 서식 있는 편집 컨트롤의 OLE 또는 이미지 개체에 대한 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

roapi.h 헤더

아티클2023. 08. 25.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)
- Windows 런타임 C++ 참조

roapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

GetActivationFactory 지정된 런타임 클래스에 대한 활성화 팩터리를 가져옵니다. (WinRTGetActivationFactory)
초기화 Windows 런타임 API를 사용하도록 스레드를 초기화합니다.
RoActivateInstance 지정된 Windows 런타임 클래스를 활성화합니다.
RoGetActivationFactory 지정된 런타임 클래스에 대한 활성화 팩터리를 가져옵니다. (RoGetActivationFactory)
RoGetApartmentIdentifier 현재 아파트의 고유 식별자를 가져옵니다.
RoInitialize 지정된 동시성 모델을 사용하여 현재 스레드의 Windows 런타임 초기화합니다.
RoRegisterActivationFactories Windows 런타임 exe 서버에 대한 배열 out-of-process 활성화 팩터리를 등록합니다.
RoRegisterForApartmentShutdown 현재 아파트가 종료될 때 호출할 IApartmentShutdown 콜백을 등록합니다.

RoRevokeActivationFactories Windows 런타임 등록된 활성화 팩터리 배열을 제거합니다.
RoUninitialize 현재 스레드의 Windows 런타임 닫습니다.
RoUnregisterForApartmentShutdown 이전에 등록된 IApartmentShutdown 인터페이스의 등록을 취소합니다.

열거형

RO_INIT_TYPE 이 스레드에서 만든 개체에 대한 수신 호출에 사용되는 동시성 모델을 결정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

robuffer.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

robuffer.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

RoGetBufferMarshaler IBuffer 인터페이스가 마샬링될 때 IBuffer 인터페이스와 연결된 의미 체계를 구현하는 표준 IBuffer 마샬러를 제공합니다.

구조체

IBufferByteAccess 버퍼를 바이트 배열로 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

roerrorapi.h 헤더

아티클2023. 03. 08.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

roerrorapi.h 헤더에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

roerrorapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

GetRestrictedErrorInfo 현재 논리 스레드에서 SetRestrictedErrorInfo에 대한 이전 호출에서 설정한 제한된 오류 정보 개체를 가져옵니다.
IsErrorPropagationEnabled coreApplication.UnhandledErrorDetected 이벤트가 Windows 런타임 API 이벤트에 대한 콜백 함수로 등록된 대리자에 의해 반환되는 오류에 대해 발생하는지 또는 비동기 메서드의 완료에 대해 발생하는지 여부를 나타냅니다.
RoCaptureErrorContext RoFailFastWithErrorContext 함수에 대한 이후 호출에 사용할 수 있도록 현재 오류 컨텍스트를 저장합니다.
RoClearError 현재 TEB(스레드 환경 블록)에서 기존 오류 정보를 제거합니다.
RoFailFastWithErrorContext 현재 프로세스에서 연속할 수 없는 예외를 발생합니다.
RoGetErrorReportingFlags Windows 런타임 오류 함수의 현재 보고 동작을 가져옵니다.
RoGetMatchingRestrictedErrorInfo

RoInspectCapturedStackBackTrace 디버거가 대상 프로세스에서 호출 스택을 검사하는 방법을 제공합니다.
RoInspectThreadErrorInfo 오류가 발생한 지점에서 호출 스택을 나타내는 오류 개체를 가져옵니다.
RoOriginateError 연결된 디버거에 오류 및 정보 문자열을 보고합니다. (RoOriginateError)
RoOriginateErrorW 연결된 디버거에 오류 및 정보 문자열을 보고합니다. (RoOriginateErrorW)
RoOriginateLanguageException 오류, 정보 문자열 및 오류 개체를 연결된 디버거에 보고합니다.
RoReportFailedDelegate 대리자 오류가 발생할 때 전역 오류 처리기를 트리거합니다.
RoReportUnhandledError 처리되지 않은 예외가 발생할 때 전역 오류 처리기를 트리거합니다.
RoResolveRestrictedErrorInfoReference 지정된 참조에 따라 IRestrictedErrorInfo 인터페이스 포인터를 반환합니다.
RoSetErrorReportingFlags Windows 런타임 오류 함수의 보고 동작을 설정합니다.
RoTransformError 수정된 오류 및 정보 문자열을 연결된 디버거에 보고합니다.
RoTransformErrorW 변환된 오류와 정보 문자열을 연결된 디버거에 보고합니다.
SetRestrictedErrorInfo 현재 스레드에 대한 제한된 오류 정보 개체를 설정합니다.

콜백 함수

PINSPECT_MEMORY_CALLBACK RoInspectCapturedStackBackTrace 함수에서 사용하는 콜백에 대한 함수 포인터를 제공합니다.

열거형

RO_ERROR_REPORTING_FLAGS RoOriginateError 및 RoTransformError 함수의 동작을 지정합니다.
RoErrorReportingFlags RoOriginateError 및 RoTransformError 함수의 동작을 지정합니다.

rometadata.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

rometadata.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

MetaDataGetDispenser 디스펜서 클래스를 만듭니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

rometadataapi.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

rometadataapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IMetaDataAssemblyImport 어셈블리 매니페스트에 액세스하여 이를 검사하는 메서드를 제공합니다.
IMetaDataDispenser 새 메타데이터 범위를 만들거나 기존 메타데이터 범위를 여는 메서드를 제공합니다.
IMetaDataDispenserEx IMetaDataDispenser 인터페이스를 확장하여 메타데이터 API가 현재 메타데이터 scope 작동하는 방식을 제어하는 기능을 제공합니다.
IMetaDataImport PE(이식 가능) 파일이나 형식 라이브러리 또는 독립 실행형 런타임 메타데이터 이진과 같은 기타 소스에서 기존 메타데이터를 가져오고 조작하는 메서드를 제공합니다.
IMetaDataImport2 제네릭 형식 작업 기능을 제공하도록 IMetaDataImport 인터페이스를 확장합니다.
IMetaDataTables 테이블에서 메타데이터 정보를 스토리지 및 검색하는 메서드를 제공합니다.
IMetaDataTables2 메타데이터 스트림 작업을 위한 메서드를 포함하도록 IMetaDataTables를 확장합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

rometadataresolution.h 헤더

아티클2025. 01. 23.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- windows 런타임 C++ 참조

rometadataresolution.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

테이블 확장

RoCreatePropertySetSerializer IPropertySetSerializer의 새 인스턴스를 만듭니다.
RoGetMetaDataFile 지정된 형식 이름의 ABI(애플리케이션 이진 인터페이스)를 설명하는 메타데이터 파일을 찾아 검색합니다.
RoIsApiContractMajorVersionPresent 지정된 이름 및 주 버전 번호가 있는 API 계약이 있는지 여부를 나타내려면 true 또는 false를 반환합니다.
RoIsApiContractPresent 지정된 이름과 주 버전 및 부 버전 번호가 있는 API 계약이 있는지 여부를 나타내려면 true 또는 false를 반환합니다.
RoParseTypeName 매개 변수가 있는 형식의 경우 형식 이름과 기존 형식 매개 변수를 구문 분석합니다.
RoResolveNamespace Windows 런타임에서 지원하는 프로그래밍 언어에서 지정된 Windows 런타임 네임스페이스의 직접 자식, 형식 및 하위 네임스페이스를 결정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

roparameterizediid.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

roparameterizediid.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

RoFreeParameterizedTypeExtra RoGetParameterizedTypeInstanceIID에 의해 할당된 핸들을 해제합니다.
RoGetParameterizedTypeInstanceIID 매개 변수가 있는 인터페이스 또는 대리자가 지정된 형식 인수로 인스턴스화될 때 발생하는 인터페이스 또는 대리자 형식의 IID(인터페이스 식별자)를 계산합니다.
RoParameterizedTypeExtraGetTypeSignature 지정된 핸들을 사용하여 RoGetParameterizedTypeInstanceIID에 대한 마지막 호출에서 IID를 계산하는 데 사용되는 형식 서명을 가져옵니다.

구조체

IRoMetaDataLocator RoGetParameterizedTypeInstanceIID 함수가 런타임 메타데이터에 액세스할 수 있도록 합니다.
IRoSimpleMetaDataBuilder 메타데이터 로케이터에 검색된 메타데이터의 대상을 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

roregistrationapi.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

roregistrationapi.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

RoGetActivatableClassRegistration 클래스 등록 정보를 검색할 수 있습니다.
RoGetServerActivatableClasses 호출 프로세스의 패키지 ID 아래에 등록된 지정된 EXE(실행 파일) 서버에 대해 등록된 활성화 가능한 클래스를 검색합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

routprot.h 헤더

아티클2023. 08. 25.

이 헤더는 라우팅 및 원격 액세스 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 라우팅 및 원격 액세스 서비스

routprot.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

콜백 함수

PADD_INTERFACE AddInterface 함수는 라우팅 프로토콜에서 관리할 인터페이스를 추가합니다.
PCONNECT_CLIENT 라우터 관리자는 클라이언트가 라우팅 프로토콜이 실행 중인 인터페이스에 연결할 때 ConnectClient 함수를 호출합니다.
PDELETE_INTERFACE DeleteInterface 함수는 라우팅 프로토콜에서 관리하는 집합에서 인터페이스를 제거합니다.
PDISCONNECT_CLIENT 라우터 관리자는 클라이언트가 라우팅 프로토콜이 실행 중인 인터페이스에서 연결을 끊을 때 DisconnectClient 함수를 호출합니다.
PDO_UPDATE_ROUTES DoUpdateRoutes 함수는 지정된 인터페이스를 통해 라우팅 정보 업데이트를 수행하여 정적 경로 정보를 얻기 위해 라우팅 프로토콜을 요청합니다.
PGET_EVENT_MESSAGE GetEventMessage 함수는 라우팅 프로토콜의 메시지 큐에서 항목을 가져옵니다. 라우팅 프로토콜은 큐를 사용하여 라우터 관리자에게 비동기 이벤트를 알릴 수 있습니다.
PGET_GLOBAL_INFO GetGlobalInfo 함수는 라우팅 프로토콜에 의해 유지되는 전역(인터페이스별 구성 정보가 아닌)을 검색합니다.

PGET_INTERFACE_INFO GetInterfaceInfo 함수는 특정 인터페이스에 대한 라우팅 프로토콜에 의해 유지되는 구성 정보를 가져옵니다.
PGET_MFE_STATUS 라우터 관리자는 GetMfeStatus 함수를 호출하여 지정된 인터페이스, 그룹 주소 및 원본 주소에 대한 MFE(멀티캐스트 전달 항목)의 상태 가져옵니다.
PGET_NEIGHBORS 라우터 관리자는 GetNeighbors 함수를 호출하여 지정된 인터페이스를 통해 연결된 네트워크에 대한 쿼리를 가져옵니다.
PINTERFACE_STATUS 라우터 관리자는 InterfaceStatus 함수를 호출하여 인터페이스의 상태 변경합니다.
PMIB_CREATE MibCreate 함수는 SNMP MIB 스타일 만들기 요청을 라우팅 프로토콜에 전달합니다.
PMIB_DELETE MibDelete 함수는 SNMP MIB 스타일 삭제 요청을 라우팅 프로토콜에 전달합니다.
PMIB_GET MibGet 함수는 SNMP MIB 스타일 Get Request를 라우팅 프로토콜 DLL에 전달합니다.
PMIB_GET_FIRST MibGetFirst 함수는 SNMP MIB 스타일 Get First Request를 라우팅 프로토콜에 전달합니다.
PMIB_GET_NEXT MibGetNext 함수는 SNMP MIB 스타일 Get Next Request를 라우팅 프로토콜에 전달합니다.
PMIB_GET_TRAP_INFO MibGetTrapInfo 함수는 트랩 이벤트를 설정하는 모듈을 쿼리하여 트랩에 대한 자세한 내용을 확인합니다.
PMIB_SET MibSet 함수는 SNMP MIB 스타일 Set Request를 라우팅 프로토콜에 전달합니다.

PMIB_SET_TRAP_INFO MibSetTrapInfo 함수는 핸들에서 TRAP을 발급해야 할 때마다 신호를 수신하는 이벤트에 전달합니다.
PQUERY_POWER QueryPower 함수는 나중에 사용하도록 예약되어 있습니다.
PREGISTER_PROTOCOL RegisterProtocol 함수는 라우터 관리자에 라우팅 프로토콜을 등록합니다. 또한 라우팅 프로토콜이 지원하는 기능을 라우터 관리자에게 알릴 수 있습니다.
PSET_GLOBAL_INFO SetGlobalInfo 함수는 라우팅 프로토콜에 의해 유지되는 전역(인터페이스별 구성 정보가 아닌)을 설정합니다. 이 정보의 형식은 라우팅 프로토콜과 관련이 있습니다.
PSET_INTERFACE_INFO SetInterfaceInfo 함수는 라우팅 프로토콜에서 관리하는 특정 인터페이스의 구성을 설정합니다.
PSET_POWER SetPower 함수는 나중에 사용하도록 예약되어 있습니다.
PSTART_COMPLETE 라우터 관리자는 StartComplete 함수를 호출하여 라우팅 프로토콜에 초기화가 완료되고 모든 인터페이스가 추가되었음을 알릴 수 있습니다. 라우팅 프로토콜은 프로토콜별 동작을 시작하기 전에 이 호출을 기다려야 합니다.
PSTART_PROTOCOL StartProtocol 함수는 라우팅 프로토콜의 기능을 초기화합니다.
PSTOP_PROTOCOL StopProtocol 함수를 사용하면 라우팅 프로토콜이 순서대로 종료됩니다.

구조체

IP_ADAPTER_BINDING_INFO

IP_ADAPTER_BINDING_INFO 구조에는 특정 네트워크 어댑터에 대한 IP 관련 정보가 포함되어 있습니다.
IP_LOCAL_BINDING IP_LOCAL_BINDING 구조에는 어댑터에 대한 IP 주소 정보가 포함되어 있습니다.
IPV6_ADAPTER_BINDING_INFO IPV6_ADAPTER_BINDING_INFO 구조에는 특정 네트워크 어댑터에 대한 IPv6 관련 정보가 포함되어 있습니다.
IPV6_LOCAL_BINDING IPV6_LOCAL_BINDING 구조에는 어댑터에 대한 IPv6 주소 정보가 포함되어 있습니다.
메시지 MESSAGE 공용 구조체에는 라우팅 프로토콜의 메시지 큐를 통해 라우터 관리자에게 보고된 이벤트에 대한 정보가 포함되어 있습니다.
MPR50_ROUTING_CHARACTERISTICS MPR_ROUTING_CHARACTERISTICS 구조에는 라우터 관리자에 라우팅 프로토콜을 등록하는 데 사용되는 정보가 포함되어 있습니다. (MPR50_ROUTING_CHARACTERISTICS)
MPR60_ROUTING_CHARACTERISTICS MPR_ROUTING_CHARACTERISTICS 구조에는 라우터 관리자에 라우팅 프로토콜을 등록하는 데 사용되는 정보가 포함되어 있습니다. M
SUPPORT_FUNCTIONS_50 SUPPORT_FUNCTIONS_50 구조체(routprot.h)는 라우터 관리자가 제공하는 함수에 대한 포인터 집합을 라우팅 프로토콜에 전달합니다.
SUPPORT_FUNCTIONS_60 SUPPORT_FUNCTIONS_60 구조체(routprot.h)는 라우터 관리자가 제공하는 함수에 대한 포인터 집합을 라우팅 프로토콜에 전달합니다.
UPDATE_COMPLETE_MESSAGE UPDATE_COMPLETE_MESSAGE 구조에는 업데이트 작업의 완료 상태 설명하는 정보가 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

rpc.h 헤더

아티클2023. 08. 25.

이 헤더는 RPC(원격 프로시저 호출)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- RPC(원격 프로시저 호출)

rpc.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

RpcExcept RpcExcept 문은 RPC 애플리케이션에 대한 구조적 예외 처리를 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

rpcasync.h 헤더

아티클2023. 08. 25.

이 헤더는 RPC(원격 프로시저 호출)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- RPC(원격 프로시저 호출)

rpcasync.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

I_RpcExceptionFilter 예외가 치명적인지 아니면 치명적이 아닌지 확인합니다.
RpcAsyncAbortCall 서버는 RpcAsyncAbortCall을 호출하여 비동기 호출을 중단합니다.
RpcAsyncCancelCall 클라이언트는 RpcAsyncCancelCall 함수를 호출하여 비동기 호출을 취소합니다.
RpcAsyncCompleteCall 클라이언트와 서버는 RpcAsyncCompleteCall 함수를 호출하여 비동기 원격 프로시저 호출을 완료합니다.
RpcAsyncGetCallHandle RpcAsyncGetCallHandle 매크로는 비동기 원격 프로시저 호출에 대한 바인딩 핸들을 반환합니다.
RpcAsyncGetCallStatus 클라이언트는 RpcAsyncGetCallStatus 함수를 호출하여 비동기 원격 호출의 현재 상태 확인합니다.
RpcAsyncInitializeHandle 클라이언트는 RpcAsyncInitializeHandle 함수를 호출하여 비동기 호출에 사용할 RPC_ASYNC_STATE 구조를 초기화합니다.
RpcAsyncRegisterInfo RpcAsyncRegisterInfo 함수는 사용되지 않습니다.

RpcBindingBind RpcBindingBind 함수는 RPC 서버에 연결하고 바인딩합니다.
RpcBindingUnbind 이전에 RpcBindingBind에 의해 바인딩된 바인딩 핸들을 바인딩 해제합니다.
RpcErrorAddRecord RpcErrorAddRecord 함수는 확장된 오류 정보 레코드 체인에 확장 오류 정보를 추가합니다.
RpcErrorClearInformation RpcErrorClearInformation 함수는 현재 스레드에 대한 모든 확장된 오류 정보를 지웁니다.
RpcErrorEndEnumeration RpcErrorEndEnumeration 함수는 확장된 오류 정보의 열거를 종료하고 열거를 위해 RPC에서 할당한 모든 리소스를 해제합니다.
RpcErrorGetNextRecord RpcErrorGetNextRecord 함수는 열거형 핸들에 대한 다음 확장 오류 정보 레코드를 검색합니다.
RpcErrorGetNumberOfRecords RpcErrorGetNumberOfRecords 함수는 확장된 오류 정보의 레코드 수를 반환합니다.
RpcErrorLoadErrorInfo RpcErrorLoadErrorInfo 함수는 RpcErrorSaveErrorInfo 호출에서 얻은 BLOB을 확장된 오류 정보로 변환합니다.
RpcErrorResetEnumeration RpcErrorResetEnumeration 함수는 모든 In-Process 열거형에 대한 열거형 커서를 다시 설정하여 RpcErrorGetNextRecord에 대한 후속 호출이 첫 번째 확장된 오류 정보 레코드를 검색하도록 프로세스를 다시 설정합니다.
RpcErrorSaveErrorInfo RpcErrorSaveErrorInfo 함수는 열거형 핸들에 대한 모든 오류 정보를 BLOB으로 반환합니다.
RpcErrorStartEnumeration RpcErrorStartEnumeration 함수는 확장된 오류 정보의 열거를 시작합니다.

RpcFreeAuthorizationContext RpcFreeAuthorizationContext 함수는 RpcGetAuthorizationContextForClient 함수에 대한 이전 호출에서 얻은 Authz 컨텍스트를 해제합니다.
RpcGetAuthorizationContextForClient RpcGetAuthorizationContextForClient 함수는 고성능 인증을 위해 Authz 함수와 함께 사용할 수 있는 RPC 클라이언트에 대한 Authz 컨텍스트를 반환합니다. ncalrpc 및 ncacn_* 프로토콜 시퀀스에 대해서만 지원됩니다.
RpcServerInqCallAttributesA RpcServerInqCallAttributes 함수는 클라이언트 보안 컨텍스트 특성을 가져오는 RPC 서버 호출입니다. (ANSI)
RpcServerInqCallAttributesW RpcServerInqCallAttributes 함수는 클라이언트 보안 컨텍스트 특성을 가져오는 RPC 서버 호출입니다. (유니코드)
RpcServerSubscribeForNotification RPC 알림을 위해 서버를 구독합니다.
RpcServerUnsubscribeForNotification RPC 알림에서 서버 구독을 취소합니다.
RpcSsContextLockExclusive RpcSsContextLockExclusive 함수를 사용하면 애플리케이션이 전용 모드에서 컨텍스트 핸들을 사용하기 시작할 수 있습니다.
RpcSsContextLockShared RpcSsContextLockShared 함수를 사용하면 애플리케이션이 공유 모드에서 컨텍스트 핸들을 사용하기 시작할 수 있습니다.

콜백 함수

RPCNOTIFICATION_ROUTINE

RPCNOTIFICATION_ROUTINE 함수는 비동기 이벤트에 대한 응답을 사용자 지정하는 기능을 사용하여 비동기 RPC를 활용하는 프로그램을 제공합니다.

구조체

RPC_ASYNC_NOTIFICATION_INFO 비동기 원격 프로시저 호출에 대한 알림 정보를 포함합니다. 이 알림 정보는 IOC(I/O 완료 포트), Windows APC(비동기 프로시저 호출), Windows 메시징 및 Windows 이벤트 알림에 대해 구성할 수 있습니다.
RPC_ASYNC_STATE RPC_ASYNC_STATE 구조체는 비동기 원격 프로시저 호출의 상태를 유지합니다. RPC_ASYNC_STATE 비동기 호출을 대기, 쿼리, 회신 또는 취소하는 데 사용되는 이 구조체에 대한 핸들입니다.
RPC_CALL_ATTRIBUTES_V1_A RPC_CALL_ATTRIBUTES_V1 구조체는 RpcServerInqCallAttributes 함수에 대한 매개 변수를 제공합니다. Windows XP 및 Windows Server 2003 운영 체제용 ANSI 및 유니코드 버전에서 구현됩니다. (ANSI)
RPC_CALL_ATTRIBUTES_V1_W RPC_CALL_ATTRIBUTES_V1 구조체는 RpcServerInqCallAttributes 함수에 대한 매개 변수를 제공합니다. Windows XP 및 Windows Server 2003 운영 체제용 ANSI 및 유니코드 버전에서 구현됩니다. (유니코드)
RPC_CALL_ATTRIBUTES_V2_A RPC_CALL_ATTRIBUTES_V2 구조체는 RpcServerInqCallAttributes 함수에 대한 매개 변수를 제공합니다. 버전 2는 로컬 주소 및 클라이언트 프로세스 ID에 대한 지원을 지정합니다. (ANSI)
RPC_CALL_ATTRIBUTES_V2_W RPC_CALL_ATTRIBUTES_V2 구조체는 RpcServerInqCallAttributes 함수에 대한 매개 변수를 제공합니다. 버전 2는 로컬 주소 및 클라이언트 프로세스 ID에 대한 지원을 지정합니다. (유니코드)
RPC_CALL_LOCAL_ADDRESS_V1 호출이 이루어진 로컬 주소에 대한 정보를 포함합니다.
RPC_EE_INFO_PARAM RPC_EE_INFO_PARAM 구조는 확장된 오류 정보를 저장하는 데 사용됩니다.

RPC_ERROR_ENUM_HANDLE RPC_ERROR_ENUM_HANDLE 구조체는 확장 오류 정보를 처리하기 위해 RpcError* 함수에서 사용하는 열거형 핸들을 제공합니다.
RPC_EXTENDED_ERROR_INFO RPC_EXTENDED_ERROR_INFO 구조는 확장된 오류 정보를 저장하는 데 사용됩니다.

열거형

RPC_ASYNC_EVENT RPC_ASYNC_EVENT 열거형 형식은 RPC 애플리케이션이 받을 수 있는 비동기 알림 이벤트를 설명합니다.
RPC_NOTIFICATION_TYPES RPC_NOTIFICATION_TYPES 열거형 형식에는 클라이언트 프로그램에서 사용할 비동기 알림의 메서드를 지정하는 값이 포함되어 있습니다.
RPC_NOTIFICATIONS RPC_NOTIFICATIONS 열거형은 서버가 RPC에서 받을 수 있는 알림을 지정합니다.
RpcCallClientLocality 가능한 RPC 클라이언트 지역 집합을 지정합니다.
RpcCallType RPC 호출 형식 집합을 지정합니다.
RpcLocalAddressFormat RPC에서 지원하는 가능한 로컬 IP 주소 형식을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

rpcdce.h 헤더

아티클2023. 08. 25.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)
- RPC(원격 프로시저 호출)

rpcdce.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DceErrorInqText DceErrorInqText 함수(rpcdce.h)는 상태 코드에 대한 메시지 텍스트를 반환합니다.
DceErrorInqTextA DceErrorInqText 함수는 상태 코드에 대한 메시지 텍스트를 반환합니다. (DceErrorInqTextA)
DceErrorInqTextW DceErrorInqTextW(유니코드) 함수(rpcdce.h)는 상태 코드에 대한 메시지 텍스트를 반환합니다.
RpcBindingCopy RpcBindingCopy 함수는 바인딩 정보를 복사하고 새 바인딩 핸들을 만듭니다.
RpcBindingCreateA RpcBindingCreate 함수는 제공된 템플릿을 기반으로 새로운 빠른 RPC 바인딩 핸들을 만듭니다. (ANSI)
RpcBindingCreateW RpcBindingCreate 함수는 제공된 템플릿을 기반으로 새로운 빠른 RPC 바인딩 핸들을 만듭니다. (유니코드)
RpcBindingFree RpcBindingFree 함수는 바인딩 핸들 리소스를 해제합니다.
RpcBindingFromStringBinding

RpcBindingFromStringBinding 함수(rpcdce.h)는 바인딩 핸들의 문자열 표현에서 바인딩 핸들을 반환합니다.
RpcBindingFromStringBindingA 바인딩 핸들의 문자열 표현에서 바인딩 핸들을 반환합니다. (RpcBindingFromStringBindingA)
RpcBindingFromStringBindingW RpcBindingFromStringBindingW(유니코드) 함수(rpcdce.h)는 바인딩 핸들의 문자열 표현에서 바인딩 핸들을 반환합니다.
RpcBindingInqAuthClient RpcBindingInqAuthClient 함수(rpcdce.h)는 원격 프로시저 호출을 수행한 인증된 클라이언트의 보안 주체 이름 또는 권한 특성을 가져옵니다.
RpcBindingInqAuthClientA 서버 애플리케이션은 RpcBindingInqAuthClient 함수를 호출하여 원격 프로시저 호출을 수행한 인증된 클라이언트의 보안 주체 이름 또는 권한 특성을 가져옵니다. (RpcBindingInqAuthClientA)
RpcBindingInqAuthClientEx RpcBindingInqAuthClientEx 함수(rpcdce.h)는 원격 프로시저를 호출한 클라이언트 프로그램에 대한 확장된 정보를 가져옵니다.
RpcBindingInqAuthClientExA 서버 애플리케이션은 RpcBindingInqAuthClientEx 함수를 호출하여 원격 프로시저를 호출한 클라이언트 프로그램에 대한 확장된 정보를 가져옵니다. (RpcBindingInqAuthClientExA)
RpcBindingInqAuthClientExW RpcBindingInqAuthClientExW(유니코드) 함수(rpcdce.h)는 원격 프로시저 호출을 수행한 클라이언트 프로그램에 대한 확장된 정보를 가져옵니다.
RpcBindingInqAuthClientW RpcBindingInqAuthClientW(유니코드) 함수(rpcdce.h)는 원격 프로시저 호출을 수행한 클라이언트의 보안 주체 이름 또는 권한 특성을 가져옵니다.
RpcBindingInqAuthInfo RpcBindingInqAuthInfo 함수(rpcdce.h)는 바인딩 핸들에서 인증 및 권한 부여 정보를 반환합니다.
RpcBindingInqAuthInfoA

RpcBindingInqAuthInfo 함수는 바인딩 핸들에서 인증 및 권한 부여 정보를 반환합니다. (RpcBindingInqAuthInfoA)
RpcBindingInqAuthInfoExA RpcBindingInqAuthInfoEx 함수는 바인딩 핸들에서 인증, 권한 부여 및 보안 서비스 품질 정보를 반환합니다. (ANSI)
RpcBindingInqAuthInfoExW RpcBindingInqAuthInfoEx 함수는 바인딩 핸들에서 인증, 권한 부여 및 보안 서비스 품질 정보를 반환합니다. (유니코드)
RpcBindingInqAuthInfoW RpcBindingInqAuthInfoW(유니코드) 함수(rpcdce.h)는 바인딩 핸들에서 인증 및 권한 부여 정보를 반환합니다.
RpcBindingInqObject RpcBindingInqObject 함수는 바인딩 핸들에서 개체 UUID를 반환합니다.
RpcBindingInqOption RPC 클라이언트 프로세스는 RpcBindingInqOption을 사용하여 지정된 바인딩 핸들에 대한 바인딩 옵션의 현재 값을 확인합니다.
RpcBindingReset RpcBindingReset 함수는 호스트가 지정되었지만 해당 호스트의 서버가 지정되지 않도록 바인딩 핸들을 다시 설정합니다.
RpcBindingServerFromClient 애플리케이션은 RpcBindingServerFromClient를 호출하여 클라이언트 바인딩 핸들을 부분적으로 바인딩된 서버 바인딩 핸들로 변환합니다.
RpcBindingSetAuthInfo RpcBindingSetAuthInfo 함수(rpcdce.h)는 바인딩 핸들의 인증 및 권한 부여 정보를 설정합니다.
RpcBindingSetAuthInfoA RpcBindingSetAuthInfo 함수는 바인딩 핸들의 인증 및 권한 부여 정보를 설정합니다. (RpcBindingSetAuthInfoA)
RpcBindingSetAuthInfoExA

RpcBindingSetAuthInfoEx 함수는 바인딩 핸들의 인증, 권한 부여 및 보안 서비스 품질 정보를 설정합니다. (ANSI)
RpcBindingSetAuthInfoExW RpcBindingSetAuthInfoEx 함수는 바인딩 핸들의 인증, 권한 부여 및 보안 서비스 품질 정보를 설정합니다. (유니코드)
RpcBindingSetAuthInfoW RpcBindingSetAuthInfoW(유니코드) 함수(rpcdce.h)는 바인딩 핸들의 인증 및 권한 부여 정보를 설정합니다.
RpcBindingSetObject RpcBindingSetObject 함수는 바인딩 핸들에서 개체 UUID 값을 설정합니다.
RpcBindingSetOption RpcBindingSetOption 함수를 사용하면 클라이언트 애플리케이션이 바인딩 핸들에 메시지 큐 옵션을 지정할 수 있습니다.
RpcBindingToStringBinding RpcBindingToStringBinding 함수(rpcdce.h)는 바인딩 핸들의 문자열 표현을 반환합니다.
RpcBindingToStringBindingA RpcBindingToStringBinding 함수는 바인딩 핸들의 문자열 표현을 반환합니다. (RpcBindingToStringBindingA)
RpcBindingToStringBindingW RpcBindingToStringBindingW(유니코드) 함수(rpcdce.h)는 바인딩 핸들의 문자열 표현을 반환합니다.
RpcBindingVectorFree RpcBindingVectorFree 함수는 벡터와 벡터 자체에 포함된 바인딩 핸들을 해제합니다.
RpcCancelThread RpcCancelThread 함수는 스레드를 취소합니다. RpcCancelThread 함수는 비동기 RPC 호출을 취소하는 데 사용하면 안 됩니다. 대신 RpcAsyncCancelCall 함수를 사용하여 비동기 RPC 호출을 취소합니다.
RpcCancelThreadEx RpcCancelThreadEx 함수는 스레드 실행을 중지합니다.

RpcEpRegister RpcEpRegister 함수(rpcdce.h)는 로컬 엔드포인트 맵 데이터베이스의 서버 주소 정보를 추가하거나 대체합니다.
RpcEpRegisterA RpcEpRegister 함수는 로컬 엔드포인트 맵 데이터베이스의 서버 주소 정보를 추가하거나 대체합니다. (RpcEpRegisterA)
RpcEpRegisterNoReplace RpcEpRegisterNoReplace 함수(rpcdce.h)는 로컬 엔드포인트 맵 데이터베이스에 서버 주소 정보를 추가합니다.
RpcEpRegisterNoReplaceA RpcEpRegisterNoReplace 함수는 로컬 엔드포인트 맵 데이터베이스에 서버 주소 정보를 추가합니다. (RpcEpRegisterNoReplaceA)
RpcEpRegisterNoReplaceW RpcEpRegisterNoReplaceW(유니코드) 함수(rpcdce.h)는 로컬 엔드포인트 맵 데이터베이스에 서버 주소 정보를 추가합니다.
RpcEpRegisterW RpcEpRegisterW(유니코드) 함수(rpcdce.h)는 로컬 엔드포인트 맵 데이터베이스의 서버 주소 정보를 추가하거나 대체합니다.
RpcEpResolveBinding RpcEpResolveBinding 함수는 부분적으로 바인딩된 서버 바인딩 핸들을 완전히 바인딩된 서버 바인딩 핸들로 확인합니다.
RpcEpUnregister RpcEpUnregister 함수는 로컬 엔드포인트 맵 데이터베이스에서 서버 주소 정보를 제거합니다.
RpcExceptionFilter 예외가 치명적이거나 치명적이 아닌지 여부를 결정하는 기본 예외 필터입니다.
RpcIfIdVectorFree RpcIfIdVectorFree 함수는 벡터와 벡터에 포함된 인터페이스 식별 구조를 해제합니다.
RpcIfInqId

RpcIfInqId 함수는 인터페이스 사양의 인터페이스 식별 부분을 반환합니다.
RpcImpersonateClient 클라이언트 원격 프로시저 호출을 처리하는 서버 스레드는 RpcImpersonateClient 함수를 호출하여 활성 클라이언트를 가장할 수 있습니다.
RpcImpersonateClientContainer RpcImpersonateClientContainer를 변경하거나 사용할 수 없습니다.
RpcMgmtEnableIdleCleanup RpcMgmtEnableIdleCleanup 함수를 사용하면 RPC가 클라이언트에서 네트워크 연결과 같은 유휴 리소스를 닫을 수 있습니다.
RpcMgmtEpEltInqBegin RpcMgmtEpEltInqBegin 함수는 엔드포인트 맵에서 요소를 보기 위한 조회 컨텍스트를 만듭니다.
RpcMgmtEpEltInqDone RpcMgmtEpEltInqDone 함수는 엔드포인트 맵에서 요소를 보기 위한 조회 컨텍스트를 삭제합니다.
RpcMgmtEpEltInqNext RpcMgmtEpEltInqNext 함수(rpcdce.h)는 엔드포인트 맵에서 하나의 요소를 반환합니다.
RpcMgmtEpEltInqNextA RpcMgmtEpEltInqNext 함수는 엔드포인트 맵에서 하나의 요소를 반환합니다. (RpcMgmtEpEltInqNextA)
RpcMgmtEpEltInqNextW RpcMgmtEpEltInqNextW(유니코드) 함수(rpcdce.h)는 엔드포인트 맵에서 하나의 요소를 반환합니다.
RpcMgmtEpUnregister RpcMgmtEpUnregister 함수는 엔드포인트 맵에서 서버 주소 정보를 제거합니다.
RpcMgmtInqComTimeout RpcMgmtInqComTimeout 함수는 바인딩 핸들에서 바인딩 통신 제한 시간 값을 반환합니다.
RpcMgmtInqDefaultProtectLevel RpcMgmtInqDefaultProtectLevel 함수는 인증 서비스에 대한 기본 인증 수준을 반환합니다.

RpcMgmtInqIfIds RpcMgmtInqIfIds 함수는 서버에서 제공하는 인터페이스의 식별자를 포함하는 벡터를 반환합니다.
RpcMgmtInqServerPrincName RpcMgmtInqServerPrincName 함수(rpcdce.h)는 서버의 보안 주체 이름을 반환합니다.
RpcMgmtInqServerPrincNameA RpcMgmtInqServerPrincName 함수는 서버의 보안 주체 이름을 반환합니다. (RpcMgmtInqServerPrincNameA)
RpcMgmtInqServerPrincNameW RpcMgmtInqServerPrincNameW(유니코드) 함수(rpcdce.h)는 서버의 보안 주체 이름을 반환합니다.
RpcMgmtInqStats RpcMgmtInqStats 함수는 RPC 런타임 통계를 반환합니다.
RpcMgmtIsServerListening RpcMgmtIsServerListening 함수는 서버가 원격 프로시저 호출을 수신 대기하는지 여부를 알려줍니다.
RpcMgmtSetAuthorizationFn RpcMgmtSetAuthorizationFn 함수는 서버의 관리 함수에 대한 원격 호출을 처리하기 위한 권한 부여 함수를 설정합니다.
RpcMgmtSetCancelTimeout RpcMgmtSetCancelTimeout 함수는 취소를 전달한 후 시간이 초과되기 전에 대기할 시간에 하한을 설정합니다.
RpcMgmtSetComTimeout RpcMgmtSetComTimeout 함수는 바인딩 핸들에서 바인딩 통신 제한 시간 값을 설정합니다.
RpcMgmtSetServerStackSize RpcMgmtSetServerStackSize 함수는 RPC 런타임에서 만든 서버 스레드의 스택 크기를 지정합니다.
RpcMgmtStatsVectorFree RpcMgmtStatsVectorFree 함수는 통계 벡터를 해제합니다.

RpcMgmtStopServerListening RpcMgmtStopServerListening 함수는 서버에 원격 프로시저 호출 수신 대기를 중지하도록 지시합니다. 이 함수는 자동 수신 대기 인터페이스에 영향을 미치지 않습니다. 자세한 내용은 RpcServerRegisterIfEx를 참조하세요.
RpcMgmtWaitServerListen RpcMgmtWaitServerListen 함수는 일반적으로 RpcServerListen과 연결된 대기 작업을 수행합니다.
RpcNetworkInqProtseqs RpcNetworkInqProtseqs 함수(rpcdce.h)는 RPC 런타임 라이브러리와 운영 체제 모두에서 지원하는 모든 프로토콜 시퀀스를 반환합니다.
RpcNetworkInqProtseqsA RpcNetworkInqProtseqs 함수는 RPC 런타임 라이브러리와 운영 체제 모두에서 지원하는 모든 프로토콜 시퀀스를 반환합니다. (RpcNetworkInqProtseqsA)
RpcNetworkInqProtseqsW RpcNetworkInqProtseqsW(유니코드) 함수(rpcdce.h)는 RPC 런타임 라이브러리와 운영 체제 모두에서 지원하는 모든 프로토콜 시퀀스를 반환합니다.
RpcNetworkIsProtseqValid RpcNetworkIsProtseqValid 함수(rpcdce.h)는 RPC 런타임 라이브러리와 운영 체제가 지정된 프로토콜 시퀀스를 지원하는지 알려줍니다.
RpcNetworkIsProtseqValidA RpcNetworkIsProtseqValid 함수는 지정된 프로토콜 시퀀스가 RPC 런타임 라이브러리와 운영 체제 모두에서 지원되는지 여부를 알려줍니다. 서버 애플리케이션은 종종 RpcNetworkInqProtseqs를 사용합니다. (RpcNetworkIsProtseqValidA)
RpcNetworkIsProtseqValidW RpcNetworkIsProtseqValidW(유니코드) 함수(rpcdce.h)는 RPC 런타임 라이브러리와 운영 체제가 지정된 프로토콜 시퀀스를 지원하는지 알려줍니다.
RpcNsBindingInqEntryName RpcNsBindingInqEntryName 함수(rpcdce.h)는 바인딩 핸들이 제공된 항목 이름을 반환합니다.
RpcNsBindingInqEntryNameA RpcNsBindingInqEntryName 함수는 바인딩 핸들이 제공된 항목 이름을 반환합니다. (RpcNsBindingInqEntryNameA)

RpcNsBindingInqEntryNameW RpcNsBindingInqEntryNameW(유니코드) 함수(rpcdce.h)는 바인딩 핸들이 제공된 항목 이름을 반환합니다.
RpcObjectInqType RpcObjectInqType 함수는 개체의 형식을 반환합니다.
RpcObjectSetInqFn RpcObjectSetInqFn 함수는 개체 조회 함수를 등록합니다. null 값은 이전에 등록된 개체 조회 함수를 해제합니다.
RpcObjectSetType RpcObjectSetType 함수는 개체의 형식을 할당합니다.
RpcProtseqVectorFree RpcProtseqVectorFree 함수(rpcdce.h)는 벡터와 벡터 자체에 포함된 프로토콜 시퀀스를 해제합니다.
RpcProtseqVectorFreeA RpcProtseqVectorFree 함수는 벡터와 벡터 자체에 포함된 프로토콜 시퀀스를 해제합니다. (RpcProtseqVectorFreeA)
RpcProtseqVectorFreeW RpcProtseqVectorFreeW(유니코드) 함수(rpcdce.h)는 벡터와 벡터 자체에 포함된 프로토콜 시퀀스를 해제합니다.
RpcRaiseException RpcRaiseException 함수를 사용하여 예외를 발생합니다. 함수는 호출자에게 반환되지 않습니다.
RpcRevertContainerImpersonation RpcRevertContainerImpersonation을 변경하거나 사용할 수 없습니다.
RpcRevertToSelf RpcImpersonateClient를 호출하고 클라이언트 가장이 필요한 작업을 완료한 후 서버는 RpcRevertToSelf를 호출하여 가장을 종료하고 자체 보안 ID를 다시 설정합니다.
RpcRevertToSelfEx RpcRevertToSelfEx 함수를 사용하면 서버가 클라이언트를 가장한 다음, 클라이언트를 가장하는 호

출이 원래 RPC에서 디스패치된 스레드가 아닌 스레드에서 올 수 있는 다중 스레드 작업에서 되돌리기 수 있습니다.
RpcServerCompleteSecurityCallback RpcServerCompleteSecurityCallback 함수는 비동기 보안 콜백을 완료합니다.
RpcServerInqBindingHandle RpcServerInqBindingHandle이 호출되는 스레드에서 서비스되는 RPC 호출에 대한 바인딩 핸들을 가져옵니다.
RpcServerInqBindings RpcServerInqBindings 함수는 원격 프로시저 호출을 받을 수 있는 바인딩 핸들을 반환합니다.
RpcServerInqDefaultPrincName RpcServerInqDefaultPrincName 함수(rpcdce.h)는 지정된 인증 서비스에 대한 기본 보안 주체 이름을 가져옵니다.
RpcServerInqDefaultPrincNameA RpcServerInqDefaultPrincName 함수는 지정된 인증 서비스의 기본 보안 주체 이름을 가져옵니다. (RpcServerInqDefaultPrincNameA)
RpcServerInqDefaultPrincNameW RpcServerInqDefaultPrincNameW(유니코드) 함수(rpcdce.h)는 지정된 인증 서비스에 대한 기본 보안 주체 이름을 가져옵니다.
RpcServerInqIf RpcServerInqIf 함수는 인터페이스에 대해 등록된 EPV(관리자 진입점 벡터)를 반환합니다.
RpcServerInterfaceGroupActivate RpcServerInterfaceGroupActivate 함수는 RPC 서버 런타임에 인터페이스 그룹의 인터페이스 및 엔드포인트를 등록하고 호출 수신 대기를 시작하도록 지시합니다.
RpcServerInterfaceGroupClose RpcServerInterfaceGroupClose 함수는 인터페이스 그룹을 해제하는 데 사용됩니다.
RpcServerInterfaceGroupCreate RpcServerInterfaceGroupCreate 함수(rpcdce.h)는 서버 애플리케이션에 대한 RPC 서버 인터페이스 그룹을 만듭니다.

RpcServerInterfaceGroupCreateA RpcServerInterfaceGroupCreate 함수는 서버 애플리케이션에 대한 RPC 서버 인터페이스 그룹을 만듭니다. (RpcServerInterfaceGroupCreateA)
RpcServerInterfaceGroupCreateW RpcServerInterfaceGroupCreateW(유니코드) 함수(rpcdce.h)는 서버 애플리케이션에 대한 RPC 서버 인터페이스 그룹을 만듭니다.
RpcServerInterfaceGroupDeactivate RpcServerInterfaceGroupDeactivate 함수는 RPC 런타임에 지정된 인터페이스 그룹을 닫으려고 시도하도록 지시하고, 선택적으로 미해결 클라이언트 작업이 있는 경우 작업을 중단합니다.
RpcServerInterfaceGroupInqBindings RpcServerInterfaceGroupInqBindings 함수는 지정된 인터페이스 그룹에 대해 원격 프로시저 호출을 수신할 수 있는 바인딩 핸들을 반환합니다.
RpcServerListen RpcServerListen 함수는 원격 프로시저 호출을 수신 대기하도록 RPC 런타임 라이브러리에 신호를 보냅니다. 이 함수는 자동 수신 대기 인터페이스에 영향을 미치지 않습니다. 해당 기능이 필요한 경우 RpcServerRegisterIfEx를 사용합니다.
RpcServerRegisterAuthInfo RpcServerRegisterAuthInfo 함수(rpcdce.h)는 RPC 런타임 라이브러리에 인증 정보를 등록합니다.
RpcServerRegisterAuthInfoA RpcServerRegisterAuthInfo 함수는 RPC 런타임 라이브러리에 인증 정보를 등록합니다. (RpcServerRegisterAuthInfoA)
RpcServerRegisterAuthInfoW RpcServerRegisterAuthInfoW(유니코드) 함수(rpcdce.h)는 RPC 런타임 라이브러리에 인증 정보를 등록합니다.
RpcServerRegisterIf RpcServerRegisterIf 함수는 RPC 런타임 라이브러리에 인터페이스를 등록합니다.
RpcServerRegisterIf2 RpcServerRegisterIf2 함수는 RPC 런타임 라이브러리에 인터페이스를 등록합니다.

RpcServerRegisterIf3 RpcServerRegisterIf3 함수는 RPC 런타임 라이브러리에 인터페이스를 등록합니다.
RpcServerRegisterIfEx RpcServerRegisterIfEx 함수는 RPC 런타임 라이브러리에 인터페이스를 등록합니다.
RpcServerTestCancel 서버는 RpcServerTestCancel을 호출하여 클라이언트 취소 요청을 테스트합니다.
RpcServerUnregisterIf RpcServerUnregisterIf 함수는 RPC 런타임 라이브러리 레지스트리에서 인터페이스를 제거합니다.
RpcServerUnregisterIfEx RpcServerUnregisterIfEx 함수는 RPC 런타임 라이브러리 레지스트리에서 인터페이스를 제거합니다. 이 함수는 RpcServerUnregisterIf 함수의 기능을 확장합니다.
RpcServerUseAllProtseqs RpcServerUseAllProtseqs 함수는 RPC 런타임 라이브러리에 지원되는 모든 프로토콜 시퀀스를 사용하여 원격 프로시저 호출을 수신하도록 지시합니다.
RpcServerUseAllProtseqsEx RpcServerUseAllProtseqsEx 함수는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위해 지원되는 모든 프로토콜 시퀀스를 사용하도록 지시합니다.
RpcServerUseAllProtseqsIf RpcServerUseAllProtseqsIf 함수는 원격 프로시저 호출을 수신하기 위해 인터페이스 사양에서 지정된 모든 프로토콜 시퀀스 및 엔드포인트를 사용하도록 RPC 런타임 라이브러리에 지시합니다.
RpcServerUseAllProtseqsIfEx RpcServerUseAllProtseqsIfEx 함수는 원격 프로시저 호출을 수신하기 위해 인터페이스 사양에서 지정된 모든 프로토콜 시퀀스 및 엔드포인트를 사용하도록 RPC 런타임 라이브러리에 지시합니다.
RpcServerUseProtseq RpcServerUseProtseq 함수(rpcdce.h)는 RPC 런타임 라이브러리에 지정된 프로토콜 시퀀스를 사용하여 원격 프로시저 호출을 받도록 지시합니다.
RpcServerUseProtseqA

RpcServerUseProtseq 함수는 RPC 런타임 라이브러리에 지정된 프로토콜 시퀀스를 사용하여 원격 프로시저 호출을 수신하도록 지시합니다. (RpcServerUseProtseqA)
RpcServerUseProtseqEp RpcServerUseProtseqEp 함수(rpcdce.h)는 RPC 런타임 라이브러리에 지정된 프로토콜 시퀀스 및 엔드포인트를 사용하여 원격 프로시저 호출을 받도록 지시합니다.
RpcServerUseProtseqEpA RpcServerUseProtseqEp 함수는 RPC 런타임 라이브러리에 지정된 엔드포인트와 결합된 지정된 프로토콜 시퀀스를 사용하여 원격 프로시저 호출을 받도록 지시합니다. (RpcServerUseProtseqEpA)
RpcServerUseProtseqEpEx RpcServerUseProtseqEpEx 함수(rpcdce.h)는 RPC 런타임 라이브러리에 지정된 프로토콜 시퀀스와 엔드포인트를 사용하여 원격 프로시저 호출을 받도록 지시합니다.
RpcServerUseProtseqEpExA RpcServerUseProtseqEpEx 함수는 RPC 런타임 라이브러리에 지정된 엔드포인트와 결합된 지정된 프로토콜 시퀀스를 사용하여 원격 프로시저 호출을 받도록 지시합니다. (RpcServerUseProtseqEpExA)
RpcServerUseProtseqEpExW RpcServerUseProtseqEpExW(유니코드) 함수(rpcdce.h)는 RPC 런타임 라이브러리에 지정된 프로토콜 시퀀스와 엔드포인트를 사용하여 원격 프로시저 호출을 받도록 지시합니다.
RpcServerUseProtseqEpW RpcServerUseProtseqEpW(유니코드) 함수(rpcdce.h)는 RPC 런타임 라이브러리에 지정된 프로토콜 시퀀스 및 엔드포인트를 사용하여 원격 프로시저 호출을 받도록 지시합니다.
RpcServerUseProtseqEx RpcServerUseProtseqEx 함수(rpcdce.h)는 RPC 런타임 라이브러리에 지정된 프로토콜 시퀀스를 사용하여 원격 프로시저 호출을 받도록 지시합니다.
RpcServerUseProtseqExA RpcServerUseProtseqEx 함수는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위해 지정된 프로토콜 시퀀스를 사용하도록 지시합니다. (RpcServerUseProtseqExA)
RpcServerUseProtseqExW RpcServerUseProtseqExW(유니코드) 함수(rpcdce.h)는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위해 지정된 프로토콜 시퀀스를 사용하도록 지시합니다.

RpcServerUseProtseqIf RpcServerUseProtseqIf 함수(rpcdce.h)는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위한 인터페이스 사양에서 지정된 프로토콜 시퀀스 및 엔드포인트를 사용하도록 지시합니다.
RpcServerUseProtseqIfA RpcServerUseProtseqIf 함수는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위한 인터페이스 사양의 엔드포인트와 결합된 지정된 프로토콜 시퀀스를 사용하도록 지시합니다. (RpcServerUseProtseqIfA)
RpcServerUseProtseqIfEx RpcServerUseProtseqIfEx 함수(rpcdce.h)는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위한 인터페이스 사양에서 지정된 프로토콜 시퀀스 및 엔드포인트를 사용하도록 지시합니다.
RpcServerUseProtseqIfExA RpcServerUseProtseqIfEx 함수는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위한 인터페이스 사양의 엔드포인트와 결합된 지정된 프로토콜 시퀀스를 사용하도록 지시합니다. (RpcServerUseProtseqIfExA)
RpcServerUseProtseqIfExW RpcServerUseProtseqIfExW(유니코드) 함수(rpcdce.h)는 원격 프로시저 호출을 수신하기 위한 인터페이스 사양에서 지정된 프로토콜 시퀀스 및 엔드포인트를 사용하도록 RPC 런타임 라이브러리에 지시합니다.
RpcServerUseProtseqIfW RpcServerUseProtseqIfW(유니코드) 함수(rpcdce.h)는 원격 프로시저 호출을 수신하기 위한 인터페이스 사양에서 지정된 프로토콜 시퀀스 및 엔드포인트를 사용하도록 RPC 런타임 라이브러리에 지시합니다.
RpcServerUseProtseqW RpcServerUseProtseqW(유니코드) 함수(rpcdce.h)는 RPC 런타임 라이브러리에 원격 프로시저 호출을 수신하기 위해 지정된 프로토콜 시퀀스를 사용하도록 지시합니다.
RpcSsDontSerializeContext RpcSsDontSerializeContext 함수는 동일한 컨텍스트 핸들에서 서버 관리자 루틴에 디스패치된 여러 호출의 런타임 serialization을 사용하지 않도록 설정합니다.
RpcStringBindingCompose RpcStringBindingCompose 함수(rpcdce.h)는 문자열 바인딩 핸들을 만듭니다.

RpcStringBindingComposeA RpcStringBindingCompose 함수는 문자열 바인딩 핸들을 만듭니다. (RpcStringBindingComposeA)
RpcStringBindingComposeW RpcStringBindingComposeW(유니코드) 함수(rpcdce.h)는 문자열 바인딩 핸들을 만듭니다.
RpcStringBindingParse RpcStringBindingParse 함수(rpcdce.h)는 개체 UUID 부분과 문자열 바인딩의 주소 부분을 별도의 문자열로 반환합니다.
RpcStringBindingParseA RpcStringBindingParse 함수는 개체 UUID 부분과 문자열 바인딩의 주소 부분을 별도의 문자열로 반환합니다. (RpcStringBindingParseA)
RpcStringBindingParseW RpcStringBindingParseW(유니코드) 함수(rpcdce.h)는 개체 UUID 부분과 문자열 바인딩의 주소 부분을 별도의 문자열로 반환합니다.
RpcStringFree RpcStringFree 함수(rpcdce.h)는 RPC 런타임 라이브러리에서 할당한 문자 문자열을 해제합니다.
RpcStringFreeA RpcStringFree 함수는 RPC 런타임 라이브러리에서 할당한 문자열을 해제합니다. (RpcStringFreeA)
RpcStringFreeW RpcStringFreeW(유니코드) 함수(rpcdce.h)는 RPC 런타임 라이브러리에서 할당한 문자열을 해제합니다.
RpcTestCancel RpcTestCancel 함수는 취소 표시를 확인합니다.
UuidCompare 애플리케이션은 UuidCompare 함수를 호출하여 두 UUID를 비교하고 순서를 결정합니다. 반환된 값은 순서를 제공합니다.
UuidCreate UuidCreate 함수는 새 UUID를 만듭니다.

UuidCreateNil UuidCreateNil 함수는 nil 반환 UUID를 만듭니다.
UuidCreateSequential UuidCreateSequential 함수는 새 UUID를 만듭니다.
UuidEqual 애플리케이션은 UuidEqual 함수를 호출하여 두 UUID를 비교하고 동일한지 여부를 확인합니다.
UuidFromString UuidFromString 함수(rpcdce.h)는 문자열을 UUID로 변환합니다.
UuidFromStringA UuidFromString 함수는 문자열을 UUID로 변환합니다. (UuidFromStringA)
UuidFromStringW UuidFromStringW(유니코드) 함수(rpcdce.h)는 문자열을 UUID로 변환합니다.
UuidHash 애플리케이션은 UuidHash 함수를 호출하여 지정된 UUID에 대한 해시 값을 생성합니다.
UuidIsNil 애플리케이션은 UuidIsNil 함수를 호출하여 지정된 UUID가 Nil 반환 UUID인지 여부를 확인합니다.
UuidToString UuidToString 함수(rpcdce.h)는 UUID를 문자열로 변환합니다.
UuidToStringA UuidToString 함수는 UUID를 문자열로 변환합니다. (UuidToStringA)
UuidToStringW UuidToStringW(유니코드) 함수(rpcdce.h)는 UUID를 문자열로 변환합니다.

콜백 함수

RPC_AUTH_KEY_RETRIEVAL_FN RPC_AUTH_KEY_RETRIEVAL_FN 함수는 암호화 키를 반환하는 서버 애플리케이션 제공 루틴의 주소를 지정하는 함수의 프로토타입입니다.
RPC_IF_CALLBACK_FN RPC_IF_CALLBACK_FN 애플리케이션에서 제공하는 보안 콜백 함수의 프로토타입입니다. 프로그램은 정의하는 각 인터페이스에 대한 콜백 함수를 제공할 수 있습니다.
RPC_INTERFACE_GROUP_IDLE_CALLBACK_FN RPC_INTERFACE_GROUP_IDLE_CALLBACK_FN 정의된 각 인터페이스 그룹에 대해 구현할 수 있는 사용자 정의 콜백입니다. 이 콜백은 인터페이스 그룹의 유휴 상태가 변경되었음을 감지할 때 RPC 런타임에서 호출됩니다.
RPC_MGMT_AUTHORIZATION_FN 이 RPC_MGMT_AUTHORIZATION_FN 서버 프로그램에서 사용자 지정 RPC 권한 부여 기술을 구현할 수 있도록 합니다.
RPC_OBJECT_INQ_FN RPC_OBJECT_INQ_FN 함수는 기본 개체 UUID를 UUID 매핑 형식으로 쉽게 대체할 수 있는 함수의 프로토타입입니다.

구조체

RPC_BINDING_HANDLE_OPTIONS_V1 RPC 바인딩 핸들을 만들 추가 옵션이 포함되어 있습니다.
RPC_BINDING_HANDLE_SECURITY_V1_A RPC 바인딩 핸들을 만들 기본 보안 옵션을 포함합니다. (ANSI)
RPC_BINDING_HANDLE_SECURITY_V1_W RPC 바인딩 핸들을 만들 기본 보안 옵션을 포함합니다. (유니코드)
RPC_BINDING_HANDLE_TEMPLATE_V1_A RPC 바인딩 핸들을 만들 기본 옵션을 포함합니다. (ANSI)

RPC_BINDING_HANDLE_TEMPLATE_V1_W RPC 바인딩 핸들을 만들 기본 옵션을 포함합니다. (유니코드)
RPC_BINDING_VECTOR RPC_BINDING_VECTOR 구조에는 서버 애플리케이션이 원격 프로시저 호출을 받을 수 있는 바인딩 핸들 목록이 포함되어 있습니다.
RPC_ENDPOINT_TEMPLATE RPC_ENDPOINT_TEMPLATE 구조체(rpcdce.h)는 프로토콜 시퀀스 및 이름을 포함하여 RPC 인터페이스 그룹 서버 엔드포인트의 속성을 지정합니다.
RPC_ENDPOINT_TEMPLATEA 프로토콜 시퀀스 및 이름을 포함하여 RPC 인터페이스 그룹 서버 엔드포인트의 속성을 지정합니다. (RPC_ENDPOINT_TEMPLATEA)
RPC_ENDPOINT_TEMPLATEW RPC_ENDPOINT_TEMPLATEW(유니코드) 구조체(rpcdce.h)는 프로토콜 시퀀스 및 이름을 포함하여 RPC 인터페이스 그룹 서버 엔드포인트의 속성을 지정합니다.
RPC_HTTP_TRANSPORT_CREDENTIALS_A RPC_HTTP_TRANSPORT_CREDENTIALS 구조는 RPC/HTTP를 사용할 때 RPC 프록시 서버에 인증하기 위한 추가 자격 증명을 정의합니다. (ANSI)
RPC_HTTP_TRANSPORT_CREDENTIALS_V2_A RPC_HTTP_TRANSPORT_CREDENTIALS_V2 구조는 RPC/HTTP를 사용할 때 RPC 프록시 서버 또는 HTTP 프록시 서버에 인증하기 위한 추가 자격 증명을 정의합니다. RPC_HTTP_TRANSPORT_CREDENTIALS_V2 HTTP 프록시 서버에 대한 인증을 허용하여 RPC_HTTP_TRANSPORT_CREDENTIALS 확장합니다. (ANSI)
RPC_HTTP_TRANSPORT_CREDENTIALS_V2_W RPC_HTTP_TRANSPORT_CREDENTIALS_V2 구조는 RPC/HTTP를 사용할 때 RPC 프록시 서버 또는 HTTP 프록시 서버에 인증하기 위한 추가 자격 증명을 정의합니다. RPC_HTTP_TRANSPORT_CREDENTIALS_V2 HTTP 프록시 서버에 대한 인증을 허용하여 RPC_HTTP_TRANSPORT_CREDENTIALS 확장합니다. (유니코드)
RPC_HTTP_TRANSPORT_CREDENTIALS_V3_A RPC_HTTP_TRANSPORT_CREDENTIALS_V3 구조는 RPC/HTTP를 사용할 때 RPC 프록시 서버 또는 HTTP 프록시 서버에 인증하기 위한 추가 자격 증명을 정의합니다.

RPC_HTTP_TRANSPORT_CREDENTIALS_V3 임의의 자격 증명 양식을 사용할 수 있도록 하여 RPC_HTTP_TRANSPORT_CREDENTIALS_V2 확장합니다. (ANSI)
RPC_HTTP_TRANSPORT_CREDENTIALS_V3_W RPC_HTTP_TRANSPORT_CREDENTIALS_V3 구조는 RPC/HTTP를 사용할 때 RPC 프록시 서버 또는 HTTP 프록시 서버에 인증하기 위한 추가 자격 증명을 정의합니다. RPC_HTTP_TRANSPORT_CREDENTIALS_V3 임의의 자격 증명 양식을 사용할 수 있도록 하여 RPC_HTTP_TRANSPORT_CREDENTIALS_V2 확장합니다. (유니코드)
RPC_HTTP_TRANSPORT_CREDENTIALS_W RPC_HTTP_TRANSPORT_CREDENTIALS 구조는 RPC/HTTP를 사용할 때 RPC 프록시 서버에 인증하기 위한 추가 자격 증명을 정의합니다. (유니코드)
RPC_IF_ID RPC_IF_ID 구조체에는 인터페이스 UUID 및 인터페이스의 주 및 부 버전 번호가 포함됩니다.
RPC_IF_ID_VECTOR RPC_IF_ID_VECTOR 구조에는 서버에서 제공하는 인터페이스 목록이 포함되어 있습니다.
RPC_INTERFACE_TEMPLATE RPC_INTERFACE_TEMPLATE(rpcdce.h) 구조체는 RPC 인터페이스 그룹 서버 인터페이스를 정의합니다.
RPC_INTERFACE_TEMPLATEA RPC 인터페이스 그룹 서버 인터페이스를 정의합니다. (RPC_INTERFACE_TEMPLATEA)
RPC_INTERFACE_TEMPLATEW RPC_INTERFACE_TEMPLATEW(유니코드) 구조체(rpcdce.h)는 RPC 인터페이스 그룹 서버 인터페이스를 정의합니다.
RPC_POLICY RPC_POLICY 구조에는 다중 호스트 컴퓨터에서 바인딩을 결정하는 플래그와 ncacn_ip_tcp 및 ncadg_ip_udp 프로토콜을 사용할 때 포트 할당이 포함됩니다.
RPC_PROTSEQ_VECTOR RPC_PROTSEQ_VECTOR 구조에는 RPC 런타임 라이브러리가 원격 프로시저 호출을 보내고 받는 데 사용하는 프로토콜 시퀀스 목록이 포함되어 있습니다.
RPC_SECURITY_QOS

RPC_SECURITY_QOS 구조는 바인딩 핸들에서 서비스 품질 설정을 정의합니다. Windows 버전의 버전 가용성에 대한 설명 을 참조하세요.
RPC_SECURITY_QOS_V2_A RPC_SECURITY_QOS_V2 구조는 바인딩 핸들에서 버전 2 보안 서비스 품질 설정을 정의합니다. Windows 버전의 버전 가용성에 대한 설명 을 참조하세요. (ANSI)
RPC_SECURITY_QOS_V2_W RPC_SECURITY_QOS_V2 구조는 바인딩 핸들에서 버전 2 보안 서비스 품질 설정을 정의합니다. Windows 버전의 버전 가용성에 대한 설명 을 참조하세요. (유니코드)
RPC_SECURITY_QOS_V3_A RPC_SECURITY_QOS_V3 구조는 바인딩 핸들에서 버전 3 보안 서비스 품질 설정을 정의합니다. Windows 버전의 버전 가용성에 대한 설명 을 참조하세요. (ANSI)
RPC_SECURITY_QOS_V3_W RPC_SECURITY_QOS_V3 구조는 바인딩 핸들에서 버전 3 보안 서비스 품질 설정을 정의합니다. Windows 버전의 버전 가용성에 대한 설명 을 참조하세요. (유니코드)
RPC_SECURITY_QOS_V4_A RPC_SECURITY_QOS_V4 구조는 바인딩 핸들에서 버전 4 보안 서비스 품질 설정을 정의합니다. Windows 버전의 버전 가용성에 대한 설명 을 참조하세요. (ANSI)
RPC_SECURITY_QOS_V4_W RPC_SECURITY_QOS_V4 구조는 바인딩 핸들에서 버전 4 보안 서비스 품질 설정을 정의합니다. Windows 버전의 버전 가용성에 대한 설명 을 참조하세요. (유니코드)
RPC_SECURITY_QOS_V5_A RPC_SECURITY_QOS_V5 구조는 바인딩 핸들에서 버전 5 보안 서비스 품질 설정을 정의합니다. Windows 버전의 버전 가용성에 대한 설명 을 참조하세요. (ANSI)
RPC_SECURITY_QOS_V5_W RPC_SECURITY_QOS_V5 구조는 바인딩 핸들에서 버전 5 보안 서비스 품질 설정을 정의합니다. Windows 버전의 버전 가용성에 대한 설명 을 참조하세요. (유니코드)
RPC_STATS_VECTOR RPC_STATS_VECTOR 구조에는 서버별로 RPC 런타임 라이브러리의 통계가 포함됩니다.
SEC_WINNT_AUTH_IDENTITY_A

SEC_WINNT_AUTH_IDENTITY 구조를 사용하면 인증을 위해 특정 사용자 이름과 암호를 런타임 라이브러리에 전달할 수 있습니다. 구조는 Windows 및 Macintosh에 유효합니다. (ANSI)
SEC_WINNT_AUTH_IDENTITY_W SEC_WINNT_AUTH_IDENTITY 구조를 사용하면 인증을 위해 특정 사용자 이름과 암호를 런타임 라이브러리에 전달할 수 있습니다. 구조는 Windows 및 Macintosh에 유효합니다. (유니코드)
UUID_VECTOR UUID_VECTOR 구조체에는 UUID 목록이 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

rpcdcep.h 헤더

아티클2023. 03. 16.

이 헤더는 RPC(원격 프로시저 호출)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- RPC(원격 프로시저 호출)

rpcdcep.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

I_RpcBindingInqLocalClientPID 클라이언트 프로세스 ID를 가져옵니다.

구조체

RPC_C_OPT_COOKIE_AUTH_DESCRIPTOR RPC_C_OPT_COOKIE_AUTH_DESCRIPTOR 구조에는 RPC/HTTP 트래픽의 헤더에 삽입되는 쿠키가 포함되어 있습니다.
RPC_CLIENT_INTERFACE RPC_CLIENT_INTERFACE 구조는 런타임 라이브러리와 스텁 간의 프라이빗 인터페이스의 일부입니다. Microsoft RPC를 사용하는 대부분의 분산 애플리케이션에는 이 구조가 필요하지 않습니다.
RPC_DISPATCH_TABLE RPC_DISPATCH_TABLE 구조는 런타임 라이브러리와 스텁 간의 프라이빗 인터페이스의 일부입니다. Microsoft RPC를 사용하는 대부분의 분산 애플리케이션에는 이 구조가 필요하지 않습니다.
RPC_MESSAGE RPC_MESSAGE 구조에는 NDR과 RPC 또는 OLE 런타임의 나머지 부분 간에 공유되는 정보가 포함됩니다.

rpcndr.h 헤더

아티클2023. 08. 25.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보
- RPC(원격 프로시저 호출)

rpcndr.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

Ndr64AsyncClientCall Ndr64AsyncClientCall은 애플리케이션에서 직접 호출할 수 없습니다.
Ndr64AsyncServerCallAll Ndr64AsyncServerCallAll은 애플리케이션에서 직접 호출할 수 없습니다.
NdrAsyncClientCall NdrAsyncClientCall 함수는 /Oi 및 /Oic 모드 스텁의 비동기 클라이언트 쪽 진입점입니다.
NdrAsyncClientCall2 NdrAsyncClientCall2를 변경하거나 사용할 수 없습니다.
NdrAsyncServerCall NdrAsyncServerCall은 애플리케이션에서 직접 호출할 수 없습니다.
NdrClearOutParameters NdrClearOutParameters 함수는 out 매개 변수의 리소스를 해제하고 서버에 대한 RPC 호출이 실패하면 메모리를 지웁니다.
NdrClientCall NdrClientCall 함수는 /Oicf 모드 스텁의 클라이언트 쪽 진입점입니다.
NdrClientCall2 NdrClientCall2 함수는 /Oicf 모드 스텁의 클라이언트 쪽 진입점입니다.

NdrClientCall3 NdrClientCall3을 변경하거나 사용할 수 없습니다.
NdrClientCall4 NdrClientCall4를 변경하거나 사용할 수 없습니다.
NdrComplexArrayBufferSize NdrComplexArrayBufferSize 함수는 복잡한 배열을 마샬링하는 데 필요한 버퍼 크기를 바이트 단위로 계산합니다.
NdrComplexArrayMarshall NdrComplexArrayMarshall 함수는 복잡한 배열을 네트워크 버퍼로 마샬링합니다.
NdrComplexArrayUnmarshall NdrComplexArrayUnmarshall 함수는 네트워크 버퍼에서 메모리로 복잡한 배열을 숨기지 않습니다.
NdrComplexStructBufferSize NdrComplexStructBufferSize 함수는 복잡한 구조를 마샬링하는 데 필요한 버퍼 크기를 바이트 단위로 계산합니다.
NdrComplexStructMarshall NdrComplexStructMarshall 함수는 복잡한 구조를 네트워크 버퍼로 마샬링합니다.
NdrComplexStructUnmarshall NdrComplexStructUnmarshall 함수는 복잡한 구조를 네트워크 버퍼에서 메모리로 분리합니다.
NdrConformantArrayBufferSize NdrConformantArrayBufferSize 함수는 준수 배열을 마샬링하는 데 필요한 버퍼 크기를 바이트 단위로 계산합니다.
NdrConformantArrayMarshall NdrConformantArrayMarshall 함수는 규칙 배열을 네트워크 버퍼로 마샬링합니다.
NdrConformantArrayUnmarshall NdrConformantArrayUnmarshall 함수는 규칙적인 배열을 unmarshalshals합니다.
NdrConformantStringBufferSize

NdrConformantStringBufferSize 함수는 규칙 문자열을 마샬링하는 데 필요한 버퍼의 크기를 바이트 단위로 계산합니다.
NdrConformantStringMarshall NdrConformantStringMarshall 함수는 준수 문자열을 서버로 보낼 네트워크 버퍼로 마샬링합니다.
NdrConformantStringUnmarshall NdrConformantStringUnmarshall 함수는 네트워크 버퍼에서 메모리로 규칙 문자열을 숨기지 않습니다.
NdrContextHandleInitialize 새 RPC 컨텍스트 핸들을 초기화합니다.
NdrContextHandleMemorySize 로컬 메모리에 표시된 대로 제공된 RPC 컨텍스트 핸들의 크기를 반환합니다.
NdrContextHandleSize NdrContextHandleSize 함수는 제공된 RPC 컨텍스트 핸들의 크기를 반환합니다.
NdrConvert NdrConvert 함수는 네트워크 버퍼를 보낸 사람의 데이터 표현에서 받는 사람의 데이터 표현(다른 경우)으로 변환합니다.
NdrDcomAsyncClientCall NdrDcomAsyncClientCall이 변경되거나 사용할 수 없을 수 있습니다.
NdrDcomAsyncClientCall2 NdrDcomAsyncClientCall2를 변경하거나 사용할 수 없습니다.
NdrGetUserMarshalInfo NdrGetUserMarshalInfo 함수는 wire_marshal 및 user_marshal 도우미 함수에 대한 추가 정보를 제공합니다.
NdrInterfacePointerBufferSize NdrInterfacePointerBufferSize 함수는 인터페이스 포인터를 마샬링하는 데 필요한 버퍼 크기를 바이트 단위로 계산합니다.
NdrInterfacePointerFree

NdrInterfacePointerFree 함수는 인터페이스 포인터를 해제합니다.
NdrInterfacePointerMarshall NdrInterfacePointerMarshall 함수는 인터페이스 포인터를 서버로 보낼 네트워크 버퍼로 마샬링합니다.
NdrInterfacePointerUnmarshall NdrInterfacePointerUnmarshall 함수는 인터페이스 포인터에서 참조하는 데이터를 네트워크 버퍼에서 메모리로 분리합니다.
NdrMesProcEncodeDecode4 NdrMesProcEncodeDecode4를 변경하거나 사용할 수 없습니다.
NdrOleAllocate NdrOleAllocate 함수는 RPC에서 개체 인터페이스에 대한 메모리를 할당하는 데 사용됩니다. 이 함수는 CoTaskMemAlloc 함수에 대한 래퍼입니다.
NdrOleFree NdrOleFree 함수는 CoTaskMemFree 함수에 대한 래퍼입니다.
NdrPointerBufferSize NdrPointerBufferSize 함수는 모든 항목에 대한 최상위 포인터에 필요한 버퍼 크기를 바이트 단위로 계산합니다.
NdrPointerFree NdrPointerFree 함수는 메모리를 해제합니다.
NdrPointerMarshall NdrPointerMarshall 함수는 모든 항목에 대한 최상위 포인터를 마샬링합니다. 구조체, 배열 또는 공용 구조체에 포함된 포인터는 NdrPointerMarshall을 직접 호출합니다.
NdrPointerUnmarshall NdrPointerUnmarshall 함수는 모든 항목에 대한 최상위 포인터를 언마샬링합니다. 구조체, 배열 또는 공용 구조체에 포함된 포인터는 NdrPointerUnmarshall을 직접 호출합니다.
NdrServerCall2 NdrServerCall2는 애플리케이션에서 직접 호출할 수 없습니다.
NdrServerCallAll

NdrServerCallAll은 애플리케이션에서 직접 호출할 수 없습니다.
NdrSimpleStructBufferSize NdrSimpleStructBufferSize 함수는 간단한 구조를 마샬링하는 데 필요한 버퍼 크기를 바이트 단위로 계산합니다.
NdrSimpleStructMarshall NdrSimpleStructMarshall 함수는 간단한 구조를 네트워크 버퍼로 마샬링합니다.
NdrSimpleStructUnmarshall NdrSimpleStructUnmarshall 함수는 네트워크 버퍼에서 메모리로 간단한 구조를 숨기지 않습니다.
NdrSimpleTypeMarshall NdrSimpleTypeMarshall 함수는 간단한 형식을 마샬링합니다.
NdrSimpleTypeUnmarshall NdrSimpleTypeUnmarshall 함수는 단순 형식을 unmarshalls합니다.
NdrStubCall2 NdrStubCall2 함수는 /Oicf 모드 스텁의 서버 쪽 진입점입니다.
NdrStubCall3 NdrStubCall3은 애플리케이션에서 직접 호출할 수 없습니다.
NdrUserMarshalBufferSize NdrUserMarshalBufferSize 함수는 사용자 마샬링 개체를 마샬링하는 데 필요한 버퍼의 크기를 바이트 단위로 계산합니다.
NdrUserMarshalFree NdrUserMarshalFree 함수는 사용자 마샬링 개체를 해제합니다.
NdrUserMarshalMarshall NdrUserMarshalMarshall 함수는 제공된 데이터 버퍼를 마샬링합니다.
NdrUserMarshalUnmarshall NdrUserMarshalUnmarshall 함수는 사용자 정의 unmarshal 루틴을 호출하여 특성을 사용하여 데이터를 언마샬링합니다.

RpcSmAllocate RpcSmAllocate 함수는 RPC 스텁 메모리 관리 함수 내에서 메모리를 할당하고 할당된 메모리 또는 NULL에 대한 포인터를 반환합니다.
RpcSmClientFree RpcSmClientFree 함수는 클라이언트 스텁에서 반환된 메모리를 해제합니다.
RpcSmDestroyClientContext RpcSmDestroyClientContext 함수는 컨텍스트 핸들에 대한 클라이언트 메모리 리소스를 회수하고 컨텍스트 핸들을 NULL로 만듭니다.
RpcSmDisableAllocate RpcSmDisableAllocate 함수는 스텁 메모리 관리 환경 내에서 리소스와 메모리를 해제합니다.
RpcSmEnableAllocate RpcSmEnableAllocate 함수는 스텁 메모리 관리 환경을 설정합니다.
RpcSmFree RpcSmFree 함수는 RpcSmAllocate에서 할당한 메모리를 해제합니다.
RpcSmGetThreadHandle RpcSmGetThreadHandle 함수는 스텁 메모리 관리 환경에 대한 스레드 핸들 또는 NULL을 반환합니다.
RpcSmSetClientAllocFree RpcSmSetClientAllocFree 함수를 사용하면 클라이언트 스텁에서 사용하는 메모리 할당 및 해제 메커니즘을 사용할 수 있습니다.
RpcSmSetThreadHandle RpcSmSetThreadHandle 함수는 스텁 메모리 관리 환경에 대한 스레드 핸들을 설정합니다.
RpcSmSwapClientAllocFree RpcSmSwapClientAllocFree 함수는 클라이언트 스텁의 메모리 할당 및 메모리 해제 메커니즘을 클라이언트에서 제공하는 메커니즘과 교환합니다.
RpcSsAllocate RpcSsAllocate 함수는 RPC 스텁 메모리 관리 함수 내에서 메모리를 할당하고 할당된 메모리 또는 NULL에 대한 포인터를 반환합니다.

RpcSsDestroyClientContext RpcSsDestroyClientContext 함수는 서버에 연결하지 않고 클라이언트가 더 이상 필요로 하지 않는 컨텍스트 핸들을 삭제합니다.
RpcSsDisableAllocate RpcSsDisableAllocate 함수는 스텁 메모리 관리 환경 내에서 리소스와 메모리를 해제합니다.
RpcSsEnableAllocate RpcSsEnableAllocate 함수는 스텁 메모리 관리 환경을 설정합니다.
RpcSsFree RpcSsFree 함수는 RpcSsAllocate에서 할당한 메모리를 해제합니다.
RpcSsGetThreadHandle RpcSsGetThreadHandle 함수는 스텁 메모리 관리 환경에 대한 스레드 핸들을 반환합니다.
RpcSsSetClientAllocFree RpcSsSetClientAllocFree 함수는 클라이언트 스텁에서 사용하는 메모리 할당 및 해제 메커니즘을 사용하도록 설정합니다.
RpcSsSetThreadHandle RpcSsSetThreadHandle 함수는 스텁 메모리 관리 환경에 대한 스레드 핸들을 설정합니다.
RpcSsSwapClientAllocFree RpcSsSwapClientAllocFree 함수는 클라이언트 스텁에서 사용하는 메모리 할당 및 릴리스 메커니즘을 클라이언트에서 제공하는 메커니즘과 교환합니다.
RpcUserFree RpcUserFree를 변경하거나 사용할 수 없습니다.

구조체

MIDL_STUB_DESC MIDL_STUB_DESC 구조체는 클라이언트와 서버 간의 RPC 호출과 관련된 인터페이스 스텁에 대한 정보를 포함하는 MIDL 생성 구조체입니다.

MIDL_STUB_MESSAGE MIDL에서 생성되며 RPC 스텁의 현재 상태 포함합니다. 애플리케이션은 MIDL_STUB_MESSAGE 구조를 직접 수정하지 않습니다.
NDR_USER_MARSHAL_INFO NDR_USER_MARSHAL_INFO 구조체는 wire_marshal 및 user_marshal 도우미 함수에 전달할 수 있는 RPC 호출의 상태에 대한 정보를 보유합니다.
NDR_USER_MARSHAL_INFO_LEVEL1 NDR_USER_MARSHAL_INFO_LEVEL1 구조체는 wire_marshal 및 user_marshal 도우미 함수에 전달할 수 있는 RPC 호출의 상태에 대한 정보를 보유합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

rpcnsi.h 헤더

이 헤더는 RPC(원격 프로시저 호출)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- RPC(원격 프로시저 호출)

rpcnsi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

RpcNsBindingExportA RpcNsBindingExport 함수는 여러 바인딩 핸들과 서버에 대한 여러 개체를 사용하여 이름 서비스 데이터베이스 항목을 설정합니다. (ANSI)
RpcNsBindingExportPnPA RpcNsBindingExportPnP 함수는 플러그 앤 플레이를 지원하는 서버에 대해 여러 바인딩 핸들과 여러 개체를 사용하여 이름 서비스 데이터베이스 항목을 설정합니다. (ANSI)
RpcNsBindingExportPnPW RpcNsBindingExportPnP 함수는 플러그 앤 플레이를 지원하는 서버에 대해 여러 바인딩 핸들과 여러 개체를 사용하여 이름 서비스 데이터베이스 항목을 설정합니다. (유니코드)
RpcNsBindingExportW RpcNsBindingExport 함수는 여러 바인딩 핸들과 서버에 대한 여러 개체를 사용하여 이름 서비스 데이터베이스 항목을 설정합니다. (유니코드)
RpcNsBindingImportBeginA RpcNsBindingImportBegin 함수는 지정된 인터페이스 및 개체를 제공하는 서버에 대한 클라이언트 호환 바인딩 핸들을 가져오기 위한 가져오기 컨텍스트를 만듭니다. (ANSI)
RpcNsBindingImportBeginW RpcNsBindingImportBegin 함수는 지정된 인터페이스 및 개체를 제공하는 서버에 대한 클라이언트 호환 바인딩 핸들을 가져오기 위한 가져오기 컨텍스트를 만듭니다. (유니코드)
RpcNsBindingImportDone RpcNsBindingImportDone 함수는 클라이언트가 호환되는 서버 검색을 완료했음을 알리고 가져오기 컨텍스트를 삭제합니다.

RpcNsBindingImportNext RpcNsBindingImportNext 함수는 인터페이스(및 선택적으로 이름 서비스 데이터베이스의 개체)를 조회하고 호환되는 서버의 바인딩 핸들을 반환합니다(있는 경우).
RpcNsBindingLookupBeginA RpcNsBindingLookupBegin 함수는 인터페이스 및 개체에 대한 조회 컨텍스트를 만듭니다. (ANSI)
RpcNsBindingLookupBeginW RpcNsBindingLookupBegin 함수는 인터페이스 및 개체에 대한 조회 컨텍스트를 만듭니다. (유니코드)
RpcNsBindingLookupDone RpcNsBindingLookupDone 함수는 클라이언트가 호환되는 서버 찾기를 완료하고 조회 컨텍스트를 삭제했음을 나타냅니다.
RpcNsBindingLookupNext RpcNsBindingLookupNext 함수는 지정된 인터페이스 및 선택적으로 개체에 대해 호환되는 바인딩 핸들 목록을 반환합니다.
RpcNsBindingSelect RpcNsBindingSelect 함수는 호환되는 바인딩 핸들 목록에서 바인딩 핸들을 반환합니다.
RpcNsBindingUnexportA RpcNsBindingUnexport 함수는 이름 서비스 데이터베이스의 항목에서 인터페이스 및 개체에 대한 바인딩 핸들을 제거합니다. (ANSI)
RpcNsBindingUnexportPnPA RpcNsBindingUnexportPnP 함수는 이름 서비스 데이터베이스의 항목에서 플러그 앤 플레이 인터페이스 및 개체에 대한 바인딩 핸들을 제거합니다. (ANSI)
RpcNsBindingUnexportPnPW RpcNsBindingUnexportPnP 함수는 이름 서비스 데이터베이스의 항목에서 플러그 앤 플레이 인터페이스 및 개체에 대한 바인딩 핸들을 제거합니다. (유니코드)
RpcNsBindingUnexportW RpcNsBindingUnexport 함수는 이름 서비스 데이터베이스의 항목에서 인터페이스 및 개체에 대한 바인딩 핸들을 제거합니다. (유니코드)
RpcNsEntryExpandNameA

RpcNsEntryExpandName 함수는 이름 서비스 항목 이름을 확장합니다. 이 함수는 Active Directory에서 지원됩니다. (ANSI)
RpcNsEntryExpandNameW RpcNsEntryExpandName 함수는 이름 서비스 항목 이름을 확장합니다. 이 함수는 Active Directory에서 지원됩니다. (유니코드)
RpcNsEntryObjectInqBeginA RpcNsEntryObjectInqBegin 함수는 이름 서비스 데이터베이스 항목의 개체에 대한 조회 컨텍스트를 만듭니다. (ANSI)
RpcNsEntryObjectInqBeginW RpcNsEntryObjectInqBegin 함수는 이름 서비스 데이터베이스 항목의 개체에 대한 조회 컨텍스트를 만듭니다. (유니코드)
RpcNsEntryObjectInqDone RpcNsEntryObjectInqDone 함수는 이름 서비스 데이터베이스 항목의 개체에 대한 조회 컨텍스트를 삭제합니다.
RpcNsEntryObjectInqNext RpcNsEntryObjectInqNext 함수는 이름 서비스 데이터베이스 항목에서 한 번에 하나의 개체를 반환합니다.
RpcNsGroupDeleteA RpcNsGroupDelete 함수는 그룹 특성을 삭제합니다. (ANSI)
RpcNsGroupDeleteW RpcNsGroupDelete 함수는 그룹 특성을 삭제합니다. (유니코드)
RpcNsGroupMbrAddA RpcNsGroupMbrAdd 함수는 그룹에 항목 이름을 추가합니다. 필요한 경우 항목을 만듭니다. (ANSI)
RpcNsGroupMbrAddW RpcNsGroupMbrAdd 함수는 그룹에 항목 이름을 추가합니다. 필요한 경우 항목을 만듭니다. (유니코드)
RpcNsGroupMbrInqBeginA RpcNsGroupMbrInqBegin 함수는 그룹 구성원을 보기 위한 조회 컨텍스트를 만듭니다. (ANSI)
RpcNsGroupMbrInqBeginW

RpcNsGroupMbrInqBegin 함수는 그룹 구성원을 보기 위한 조회 컨텍스트를 만듭니다. (유니코드)
RpcNsGroupMbrInqDone RpcNsGroupMbrInqDone 함수는 그룹에 대한 조회 컨텍스트를 삭제합니다.
RpcNsGroupMbrInqNextA RpcNsGroupMbrInqNext 함수는 그룹에서 한 번에 하나의 항목 이름을 반환합니다. (ANSI)
RpcNsGroupMbrInqNextW RpcNsGroupMbrInqNext 함수는 그룹에서 한 번에 하나의 항목 이름을 반환합니다. (유니코드)
RpcNsGroupMbrRemoveA RpcNsGroupMbrRemove 함수는 그룹에서 항목 이름을 제거합니다. (ANSI)
RpcNsGroupMbrRemoveW RpcNsGroupMbrRemove 함수는 그룹에서 항목 이름을 제거합니다. (유니코드)
RpcNsMgmtBindingUnexportA RpcNsMgmtBindingUnexport 함수는 이름 서비스 데이터베이스의 항목에서 여러 바인딩 핸들 및 개체를 제거합니다. (ANSI)
RpcNsMgmtBindingUnexportW RpcNsMgmtBindingUnexport 함수는 이름 서비스 데이터베이스의 항목에서 여러 바인딩 핸들 및 개체를 제거합니다. (유니코드)
RpcNsMgmtEntryCreateA RpcNsMgmtEntryCreate 함수는 이름 서비스 데이터베이스 항목을 만듭니다. (ANSI)
RpcNsMgmtEntryCreateW RpcNsMgmtEntryCreate 함수는 이름 서비스 데이터베이스 항목을 만듭니다. (유니코드)
RpcNsMgmtEntryDeleteA RpcNsMgmtEntryDelete 함수는 이름 서비스 데이터베이스 항목을 삭제합니다. (ANSI)
RpcNsMgmtEntryDeleteW RpcNsMgmtEntryDelete 함수는 이름 서비스 데이터베이스 항목을 삭제합니다. (유니코드)
RpcNsMgmtEntryInqIfIdsA

RpcNsMgmtEntryInqIfIds 함수는 이름 서비스 데이터베이스 항목으로 내보낸 인터페이스 목록을 반환합니다. (ANSI)
RpcNsMgmtEntryInqIfIdsW RpcNsMgmtEntryInqIfIds 함수는 이름 서비스 데이터베이스 항목으로 내보낸 인터페이스 목록을 반환합니다. (유니코드)
RpcNsMgmtHandleSetExpAge RpcNsMgmtHandleSetExpAge 함수는 이름 서비스 데이터의 로컬 복사본에 대한 이름 서비스 핸들의 만료 기간을 설정합니다.
RpcNsMgmtInqExpAge RpcNsMgmtInqExpAge 함수는 이름 서비스 데이터의 로컬 복사본에 대한 전역 만료 기간을 반환합니다.
RpcNsMgmtSetExpAge RpcNsMgmtSetExpAge 함수는 이름 서비스 데이터의 로컬 복사본에 대한 애플리케이션의 전역 만료 기간을 수정합니다.
RpcNsProfileDeleteA RpcNsProfileDelete 함수는 프로필 특성을 삭제합니다. (ANSI)
RpcNsProfileDeleteW RpcNsProfileDelete 함수는 프로필 특성을 삭제합니다. (유니코드)
RpcNsProfileEltAddA RpcNsProfileEltAdd 함수는 프로필에 요소를 추가합니다. 필요한 경우 항목을 만듭니다. (ANSI)
RpcNsProfileEltAddW RpcNsProfileEltAdd 함수는 프로필에 요소를 추가합니다. 필요한 경우 항목을 만듭니다. (유니코드)
RpcNsProfileEltInqBeginA RpcNsProfileEltInqBegin 함수는 프로필의 요소를 보기 위한 조회 컨텍스트를 만듭니다. (ANSI)
RpcNsProfileEltInqBeginW RpcNsProfileEltInqBegin 함수는 프로필의 요소를 보기 위한 조회 컨텍스트를 만듭니다. (유니코드)
RpcNsProfileEltInqDone RpcNsProfileEltInqDone 함수는 프로필의 요소를 보기 위한 조회 컨텍스트를 삭제합니다.

RpcNsProfileEltInqNextA RpcNsProfileEltInqNext 함수는 프로필에서 한 번에 하나의 요소를 반환합니다. (ANSI)
RpcNsProfileEltInqNextW RpcNsProfileEltInqNext 함수는 프로필에서 한 번에 하나의 요소를 반환합니다. (유니코드)
RpcNsProfileEltRemoveA RpcNsProfileEltRemove 함수는 프로필에서 요소를 제거합니다. (ANSI)
RpcNsProfileEltRemoveW RpcNsProfileEltRemove 함수는 프로필에서 요소를 제거합니다. (유니코드)

Last updated on 2025. 10. 08.

rpcproxy.h 헤더

아티클2023. 03. 16.

이 헤더는 RPC(원격 프로시저 호출)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- RPC(원격 프로시저 호출)

rpcproxy.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

CStdAsyncStubBuffer_AddRef IRpcStubBuffer::AddRef 메서드를 구현합니다.
CStdAsyncStubBuffer_Connect IRpcStubBuffer::Connect 메서드를 구현하고 서버 개체를 스텁에 연결합니다. (CStdAsyncStubBuffer_Connect)
CStdAsyncStubBuffer_Disconnect IRpcStubBuffer::D isconnect 메서드를 구현하고 스텁에서 서버 개체의 연결을 끊습니다. (CStdAsyncStubBuffer_Disconnect)
CStdAsyncStubBuffer_Invoke IRpcStubBuffer::Invoke 메서드를 구현합니다.
CStdAsyncStubBuffer_QueryInterface IRpcStubBuffer::QueryInterface 메서드를 구현합니다. (CStdAsyncStubBuffer_QueryInterface)
CStdAsyncStubBuffer_Release IRpcStubBuffer::Release 메서드를 구현합니다. (CStdAsyncStubBuffer_Release)
CStdAsyncStubBuffer2_Connect IRpcStubBuffer::Connect 메서드를 구현하고 서버 개체를 스텁에 연결합니다. (CStdAsyncStubBuffer2_Connect)

CStdAsyncStubBuffer2_Disconnect IRpcStubBuffer::D isconnect 메서드를 구현하고 스텁에서 서버 개체의 연결을 끊습니다. (CStdAsyncStubBuffer2_Disconnect)
CStdAsyncStubBuffer2_Release IRpcStubBuffer::Release 메서드를 구현합니다. (CStdAsyncStubBuffer2_Release)
CStdStubBuffer_AddRef CStdStubBuffer_AddRef 함수는 IRpcStubBuffer::AddRef 메서드를 구현합니다.
CStdStubBuffer_Connect CStdStubBuffer_Connect 함수는 IRpcStubBuffer::Connect 메서드를 구현하고 서버 개체를 스텁에 연결합니다.
CStdStubBuffer_CountRefs CStdStubBuffer_CountRefs 함수는 IRpcStubBuffer::CountRefs 메서드를 구현합니다.
CStdStubBuffer_DebugServerQueryInterface CStdStubBuffer_DebugServerQueryInterface 함수는 IRpcStubBuffer::D ebugServerQueryInterface 메서드를 구현합니다.
CStdStubBuffer_DebugServerRelease CStdStubBuffer_DebugServerRelease 함수는 IRpcStubBuffer::D ebugServerRelease 메서드를 구현합니다.
CStdStubBuffer_Disconnect CStdStubBuffer_Disconnect 함수는 IRpcStubBuffer::D isconnect 메서드를 구현하고 스텁에서 서버 개체의 연결을 끊습니다.
CStdStubBuffer_Invoke CStdStubBuffer_Invoke 함수는 IRpcStubBuffer::Invoke 메서드를 구현합니다.
CStdStubBuffer_IsIIDSupported CStdStubBuffer_IsIIDSupported 함수는 IRpcStubBuffer::IsIIDSupported 메서드를 구현합니다.
CStdStubBuffer_QueryInterface CStdStubBuffer_QueryInterface 함수는 IRpcStubBuffer::QueryInterface 메서드를 구현합니다.

CStdStubBuffer2_Connect IRpcStubBuffer::Connect 메서드를 구현하고 서버 개체를 스텁에 연결합니다. (CStdStubBuffer2_Connect)
CStdStubBuffer2_CountRefs IRpcStubBuffer::CountRefs 메서드를 구현합니다.
CStdStubBuffer2_Disconnect IRpcStubBuffer::D isconnect 메서드를 구현하고 스텁에서 서버 개체의 연결을 끊습니다. (CStdStubBuffer2_Disconnect)
CStdStubBuffer2_QueryInterface IRpcStubBuffer::QueryInterface 메서드를 구현합니다. (CStdStubBuffer2_QueryInterface)
NdrCStdStubBuffer_Release NdrCStdStubBuffer_Release 함수는 IRpcStubBuffer::Release 메서드를 구현합니다.
NdrCStdStubBuffer2_Release NdrCStdStubBuffer2_Release 함수는 IRpcStubBuffer::Release 메서드를 구현합니다.
NdrDllCanUnloadNow NdrDllCanUnloadNow 함수는 프록시 및 스텁을 구현하는 DLL이 여전히 사용 중인지 여부를 결정합니다. 그렇지 않은 경우 호출자는 메모리에서 DLL을 안전하게 언로드할 수 있습니다.
NdrDllGetClassObject NdrDllGetClassObject 함수는 프록시 또는 스텁의 클래스 개체를 검색합니다.
NdrDllRegisterProxy NdrDllRegisterProxy 함수는 프록시 DLL에 포함된 인터페이스에 대한 레지스트리 항목을 만듭니다.
NdrDllUnregisterProxy NdrDllUnregisterProxy 함수는 프록시 DLL에 포함된 인터페이스에 대한 레지스트리 항목을 제거합니다.
NdrProxyErrorHandler NdrProxyErrorHandler 함수는 RPC 기능 코드를 사용하여 예외를 HRESULT에 매핑합니다.

NdrProxyForwardingFunction10 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction10)
NdrProxyForwardingFunction11 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction11)
NdrProxyForwardingFunction12 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction12)
NdrProxyForwardingFunction13 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction13)
NdrProxyForwardingFunction14 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction14)
NdrProxyForwardingFunction15 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction15)
NdrProxyForwardingFunction16 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction16)
NdrProxyForwardingFunction17 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction17)
NdrProxyForwardingFunction18 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction18)
NdrProxyForwardingFunction19 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction19)
NdrProxyForwardingFunction20 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction20)
NdrProxyForwardingFunction21 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction21)

NdrProxyForwardingFunction22 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction22)
NdrProxyForwardingFunction23 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction23)
NdrProxyForwardingFunction24 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction24)
NdrProxyForwardingFunction25 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction25)
NdrProxyForwardingFunction26 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction26)
NdrProxyForwardingFunction27 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction27)
NdrProxyForwardingFunction28 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction28)
NdrProxyForwardingFunction29 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction29)
NdrProxyForwardingFunction3 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction3)
NdrProxyForwardingFunction30 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction30)
NdrProxyForwardingFunction31 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction31)
NdrProxyForwardingFunction32 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction32)

NdrProxyForwardingFunction4 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction4)
NdrProxyForwardingFunction5 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction5)
NdrProxyForwardingFunction6 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction6)
NdrProxyForwardingFunction7 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction7)
NdrProxyForwardingFunction8 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction8)
NdrProxyForwardingFunction9 COM 프록시에 대한 스텁 함수입니다. (NdrProxyForwardingFunction9)
NdrProxyFreeBuffer NdrProxyFreeBuffer 함수는 RPC 버퍼를 해제합니다.
NdrProxyGetBuffer NdrProxyGetBuffer 함수는 IRpcChannelBuffer::GetBuffer 메서드 호출을 사용하여 COM에서 네트워크 버퍼를 가져옵니다.
NdrProxyInitialize NdrProxyInitialize 함수는 개체 메서드에 대한 프록시를 초기화합니다.
NdrProxySendReceive NdrProxySendReceive 함수는 서버에 메시지를 보낸 다음 회신을 기다립니다.
NdrStubForwardingFunction NdrStubForwardingFunction 함수는 기본 인터페이스에 정의된 서버 쪽 개체 메서드의 진입점입니다.
NdrStubGetBuffer NdrStubGetBuffer 함수는 RPC 채널에서 버퍼를 검색합니다.

NdrStubInitialize NdrStubInitialize 함수는 경계를 풀기 전에 서버 스텁에서 호출됩니다. 일부 스텁 메시지 필드를 설정합니다.
ObjectStublessClient10 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient10)
ObjectStublessClient11 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient11)
ObjectStublessClient12 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient12)
ObjectStublessClient13 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient13)
ObjectStublessClient14 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient14)
ObjectStublessClient15 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient15)
ObjectStublessClient16 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient16)
ObjectStublessClient17 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient17)
ObjectStublessClient18 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient18)
ObjectStublessClient19 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient19)
ObjectStublessClient20 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient20)

ObjectStublessClient21 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient21)
ObjectStublessClient22 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient22)
ObjectStublessClient23 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient23)
ObjectStublessClient24 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient24)
ObjectStublessClient25 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient25)
ObjectStublessClient26 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient26)
ObjectStublessClient27 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient27)
ObjectStublessClient28 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient28)
ObjectStublessClient29 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient29)
ObjectStublessClient3 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient3)
ObjectStublessClient30 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient30)
ObjectStublessClient31 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient31)

ObjectStublessClient32 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient32)
ObjectStublessClient4 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient4)
ObjectStublessClient5 ObjectStublessClient5 함수는 COM 프록시에 대한 스텁 함수입니다.
ObjectStublessClient6 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient6)
ObjectStublessClient7 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient7)
ObjectStublessClient8 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient8)
ObjectStublessClient9 COM 프록시에 대한 스텁 함수입니다. (ObjectStublessClient9)

구조체

ProxyFileInfo ProxyFileInfo 구조체에는 프록시 DLL의 인터페이스 프록시에 대한 정보가 포함됩니다.

rpcssl.h 헤더

아티클2023. 08. 25.

이 헤더는 RPC(원격 프로시저 호출)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- RPC(원격 프로시저 호출)

rpcssl.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

RpcCertGeneratePrincipalName RpcCertGeneratePrincipalName 함수(rpcssl.h)는 서버 프로그램에서 보안 인증서의 주체 이름을 생성하는 데 사용됩니다.
RpcCertGeneratePrincipalNameA 서버 프로그램은 RpcCertGeneratePrincipalName 함수를 사용하여 보안 인증서에 대한 주체 이름을 생성합니다. (RpcCertGeneratePrincipalNameA)
RpcCertGeneratePrincipalNameW RpcCertGeneratePrincipalNameW(유니코드) 함수(rpcssl.h)는 서버 프로그램에서 보안 인증서의 주체 이름을 생성하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

rrascfg.h 헤더

아티클2023. 08. 25.

이 헤더는 확장 가능한 인증 프로토콜에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 확장할 수 있는 인증 프로토콜

rrascfg.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEAPProviderConfig EAP 공급자에 대한 UI입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

rtlsupportapi.h 헤더

아티클2023. 03. 16.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

rtlsupportapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

RtlRaiseException 예외를 발생합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

rtmv2.h 헤더

아티클2023. 03. 16.

이 헤더는 라우팅 및 원격 액세스 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 라우팅 및 원격 액세스 서비스

rtmv2.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

RTM_IPV4_GET_ADDR_AND_LEN RTM_IPV4_GET_ADDR_AND_LEN 매크로는 제네릭 net 주소와 길이를 IPv4 RTM_NET_ADDRESS 구조체 및 길이로 변환합니다.
RTM_IPV4_GET_ADDR_AND_MASK RTM_IPV4_GET_ADDR_AND_MASK 매크로는 제네릭 RTM_NET_ADDRESS 구조와 길이를 IPv4 주소 및 마스크로 변환합니다.
RTM_IPV4_LEN_FROM_MASK RTM_IPV4_LEN_FROM_MASK 매크로는 IPv4 마스크를 제네릭 경로 길이로 변환합니다.
RTM_IPV4_MAKE_NET_ADDRESS RTM_IPV4_MAKE_NET_ADDRESS 매크로는 IPv4 주소와 길이를 제네릭 RTM_NET_ADDRESS 구조체로 변환합니다.
RTM_IPV4_MASK_FROM_LEN RTM_IPV4_MASK_FROM_LEN 매크로는 제네릭 경로 길이를 IPv4 마스크로 변환합니다.
RTM_IPV4_SET_ADDR_AND_LEN RTM_IPV4_SET_ADDR_AND_LEN 매크로는 IPv4 주소와 길이를 제네릭 RTM_NET_ADDRESS 구조체로 변환합니다.
RTM_IPV4_SET_ADDR_AND_MASK RTM_IPV4_SET_ADDR_AND_MASK 매크로는 IPv4 주소와 마스크를 제네릭 RTM_NET_ADDRESS 구조체로 변환합니다.

RTM_SIZE_OF_DEST_INFO RTM_SIZE_OF_DEST_INFO 매크로는 대상 정보 구조(RTM_DEST_INFO)의 크기를 반환합니다.
RTM_SIZE_OF_ROUTE_INFO RTM_SIZE_OF_ROUTE_INFO 매크로는 RTM_ROUTE_INFO 경로 정보 구조의 크기를 반환합니다.
RtmAddNextHop RtmAddNextHop 함수는 새 다음 홉 항목을 추가하거나 기존 다음 홉 항목을 클라이언트의 다음 홉 목록에 업데이트합니다.
RtmAddRouteToDest RtmAddRouteToDest 함수는 라우팅 테이블에 새 경로를 추가하거나 라우팅 테이블의 기존 경로를 업데이트합니다. 최상의 경로가 변경되면 변경 알림이 생성됩니다.
RtmBlockMethods RtmBlockMethods 함수는 지정된 대상, 경로 또는 다음 홉 또는 모든 대상, 경로 및 다음 홉에 대한 메서드 실행을 차단하거나 차단 해제합니다.
RtmCreateDestEnum RtmCreateDestEnum 함수는 라우팅 테이블에서 대상의 열거를 시작합니다. 클라이언트는 하나 이상의 뷰 또는 모든 보기에 대한 대상을 열거할 수 있습니다.
RtmCreateNextHopEnum RtmCreateNextHopEnum은 다음 홉 목록에서 다음 홉을 열거합니다.
RtmCreateRouteEnum RtmCreateRouteEnum 함수는 라우팅 테이블의 특정 대상 또는 대상 범위에 대한 경로의 열거형을 만듭니다. 클라이언트는 하나 이상의 뷰 또는 모든 보기에 대한 경로를 열거할 수 있습니다.
RtmCreateRouteList RtmCreateRouteList 함수는 호출자가 소유한 경로의 복사본을 유지할 수 있는 목록을 만듭니다.
RtmCreateRouteListEnum RtmCreateRouteListEnum 함수는 지정된 경로 목록에 경로의 열거형을 만듭니다.
RtmDeleteEnumHandle RtmDeleteEnumHandle 함수는 지정된 열거형 핸들을 삭제하고 열거형에 할당된 모든 리소스를 해제합니다.

RtmDeleteNextHop RtmDeleteNextHop 함수는 다음 홉 목록에서 다음 홉을 삭제합니다.
RtmDeleteRouteList RtmDeleteRouteList 함수는 클라이언트별 경로 목록에서 모든 경로를 제거한 다음 목록에 할당된 모든 리소스를 해제합니다.
RtmDeleteRouteToDest RtmDeleteRouteToDest 함수는 라우팅 테이블에서 경로를 삭제하고 최상의 경로가 변경된 경우 해당 대상에 대한 최상의 경로 정보를 업데이트합니다. 최상의 경로가 변경되면 변경 알림이 생성됩니다.
RtmDeregisterEntity RtmDeregisterEntity 함수는 라우팅 테이블 관리자 인스턴스 및 주소 패밀리에서 클라이언트의 등록을 취소합니다.
RtmDeregisterFromChangeNotification RtmDeregisterFromChangeNotification 함수는 변경 알림에서 클라이언트의 등록을 취소하고 알림에 할당된 모든 리소스를 해제합니다.
RtmFindNextHop RtmFindNextHop 함수는 클라이언트의 다음 홉 목록에서 특정 다음 홉을 찾습니다.
RtmGetChangedDests RtmGetChangedDests 함수는 변경된 정보가 있는 대상 집합을 반환합니다.
RtmGetChangeStatus RtmGetChangeStatus 함수는 RtmGetChangedDests를 사용하여 검색되지 않은 보류 중인 변경 내용이 있는지 확인합니다.
RtmGetDestInfo RtmGetDestInfo 함수는 대상에 대한 정보를 반환합니다.
RtmGetEntityInfo RtmGetEntityInfo 함수는 이전에 등록된 클라이언트에 대한 정보를 반환합니다.

RtmGetEntityMethods RtmGetEntityMethods 함수는 지정된 클라이언트를 쿼리하여 다른 클라이언트가 호출할 수 있는 메서드를 결정합니다.
RtmGetEnumDests RtmGetEnumDests 함수는 지정된 열거형의 다음 대상 집합을 검색합니다.
RtmGetEnumNextHops RtmGetEnumNextHops 함수는 지정된 열거형에서 다음 홉 집합을 검색합니다.
RtmGetEnumRoutes RtmGetEnumRoutes 함수는 지정된 열거형의 다음 경로 집합을 검색합니다.
RtmGetExactMatchDestination RtmGetExactMatchDestination 함수는 라우팅 테이블에서 지정된 네트워크 주소 및 서브넷 마스크와 정확히 일치하는 대상을 검색합니다. 정확히 일치하는 항목이 발견되면 해당 대상에 대한 정보가 반환됩니다.
RtmGetExactMatchRoute RtmGetExactMatchRoute 함수는 라우팅 테이블에서 지정된 경로와 정확히 일치하는 경로를 검색합니다.
RtmGetLessSpecificDestination RtmGetLessSpecificDestination 함수는 대상 접두사로 지정된 다음 최적 일치(가장 긴) 접두사로 대상에 대한 라우팅 테이블을 검색합니다. 요청된 대상 정보가 반환됩니다.
RtmGetListEnumRoutes RtmGetListEnumRoutes 함수는 지정된 경로 목록의 경로 집합을 열거합니다.
RtmGetMostSpecificDestination RtmGetMostSpecificDestination 함수는 라우팅 테이블에서 지정된 네트워크 주소 및 서브넷 마스크와 정확히 일치하는 대상을 검색합니다. 정확한 일치 항목을 찾을 수 없으면 최상의 접두사는 일치합니다.
RtmGetNextHopInfo RtmGetNextHopInfo 함수는 지정된 다음 홉에 대한 정보를 반환합니다.

RtmGetNextHopPointer RtmGetNextHopPointer 함수는 지정된 다음 홉에 대한 직접 포인터를 가져옵니다. 포인터를 사용하면 다음 홉 소유자가 라우팅 테이블 관리자의 RTM_NEXTHOP_INFO 구조에 직접 읽기 권한을 부여할 수 있습니다.
RtmGetOpaqueInformationPointer RtmGetOpaqueInformationPointer 함수는 이 클라이언트용으로 예약된 대상의 불투명 정보 필드에 대한 포인터를 반환합니다.
RtmGetRegisteredEntities RtmGetRegisteredEntities 함수는 라우팅 테이블 관리자의 지정된 인스턴스 및 지정된 주소 패밀리에 등록된 모든 클라이언트에 대한 정보를 반환합니다.
RtmGetRouteInfo RtmGetRouteInfo 함수는 지정된 경로에 대한 정보를 반환합니다.
RtmGetRoutePointer RtmGetRoutePointer 함수는 경로 읽기 액세스의 소유자를 허용하는 경로에 대한 직접 포인터를 가져옵니다.
RtmHoldDestination RtmHoldDestination 함수는 일정 시간 동안 보류 상태로 배치할 대상을 표시합니다. 보류는 보기에서 대상의 마지막 경로가 삭제된 경우에만 발생합니다.
RtmIgnoreChangedDests RtmIgnoreChangedDests 함수는 이미 발생한 경우 각 대상에 대한 다음 변경 사항을 건너뜁니다.
RtmInsertInRouteList RtmInsertInRouteList 함수는 지정된 경로 집합을 클라이언트의 경로 목록에 삽입합니다. 경로가 이미 다른 목록에 있는 경우 경로가 이전 목록에서 제거되고 새 목록에 삽입됩니다.
RtmInvokeMethod RtmInvokeMethod 함수는 다른 클라이언트에서 내보낸 메서드를 호출합니다.
RtmIsBestRoute RtmIsBestRoute 함수는 지정된 경로가 대상에 가장 적합한 경로인 보기 집합을 반환합니다.

RtmIsMarkedForChangeNotification RtmIsMarkedForChangeNotification 함수는 라우팅 테이블 관리자를 쿼리하여 대상이 이전에 RtmMarkDestForChangeNotification에 대한 호출로 표시되었는지 확인합니다.
RtmLockDestination RtmLockDestination 함수는 라우팅 테이블의 대상을 잠그거나 잠금 해제합니다. 불투명 포인터를 변경하는 동안 대상을 보호하려면 이 함수를 사용합니다.
RtmLockNextHop RtmLockNextHop 함수는 다음 홉을 잠그거나 잠금을 해제합니다. 다음 홉을 변경하기 전에 다음 홉의 소유자가 다음 홉을 잠그도록 이 함수를 호출해야 합니다. 다음 홉에 대한 포인터가 반환됩니다.
RtmLockRoute RtmLockRoute 함수는 라우팅 테이블에서 경로를 잠그거나 잠금을 해제합니다. 이렇게 하면 클라이언트가 클라이언트가 소유한 불투명 경로 포인터에 필요한 변경을 수행하는 동안 경로가 보호됩니다.
RtmMarkDestForChangeNotification RtmMarkDestForChangeNotification 함수는 클라이언트의 대상을 표시합니다.
RtmReferenceHandles RtmReferenceHandles 함수는 라우팅 관리자가 해당 개체에 액세스하는 데 사용한 하나 이상의 핸들이 가리키는 개체에 대한 참조 횟수를 증가합니다.
RtmRegisterEntity RtmRegisterEntity 함수는 특정 주소 패밀리에 대한 라우팅 테이블 관리자 인스턴스에 클라이언트를 등록합니다.
RtmRegisterForChangeNotification RtmRegisterForChangeNotification 함수는 클라이언트가 지정된 유형의 변경 내용에 대한 변경 알림을 받아야 한다고 라우팅 테이블 관리자에게 알릴 수 있습니다.
RtmReleaseChangedDests RtmReleaseChangedDests 함수는 변경된 대상 핸들을 해제합니다.
RtmReleaseDestInfo RtmReleaseDestInfo 함수는 대상 구조를 해제합니다.

RtmReleaseDests RtmReleaseDests 함수는 대상 핸들을 해제합니다.
RtmReleaseEntities RtmReleaseEntities 함수는 RtmGetRegisteredEntities에서 반환된 클라이언트 핸들을 해제합니다.
RtmReleaseEntityInfo RtmReleaseEntityInfo 함수는 클라이언트 구조를 해제합니다.
RtmReleaseNextHopInfo RtmReleaseNextHopInfo 함수는 다음 홉 구조를 해제합니다.
RtmReleaseNextHops RtmReleaseNextHops 함수는 다음 홉 핸들을 해제합니다.
RtmReleaseRouteInfo RtmReleaseRouteInfo 함수는 경로 구조를 해제합니다.
RtmReleaseRoutes RtmReleaseRoutes 함수는 경로 핸들을 해제합니다.
RtmUpdateAndUnlockRoute RtmUpdateAndUnlockRoute 함수는 대상 경로 집합에서 경로의 위치를 업데이트하고 대상에 가장 적합한 경로 정보를 조정합니다.

콜백 함수

_ENTITY_METHOD RTM_ENTITY_EXPORT_METHOD 콜백은 클라이언트에서 내보낸 모든 메서드의 프로토타입입니다.
_EVENT_CALLBACK RTM_EVENT_CALLBACK 콜백은 라우팅 테이블 관리자가 클라이언트에 지정된 이벤트가 발생했음을 알리는 데 사용됩니다.

구조체

RTM_DEST_INFO RTM_DEST_INFO 구조는 라우팅 테이블 관리자에 등록된 클라이언트와 대상 정보를 교환하는 데 사용됩니다.
RTM_ENTITY_EXPORT_METHODS RTM_ENTITY_EXPORT_METHODS 구조에는 클라이언트에서 내보낸 메서드 집합이 포함됩니다.
RTM_ENTITY_ID RTM_ENTITY_ID 구조는 라우팅 테이블 관리자에 대한 클라이언트를 고유하게 식별하는 데 사용됩니다. 프로토콜 식별자와 인스턴스 식별자는 클라이언트를 고유하게 식별하는 데 사용되는 값입니다.
RTM_ENTITY_INFO RTM_ENTITY_INFO 구조는 라우팅 테이블 관리자와 클라이언트 정보를 교환하는 데 사용됩니다.
RTM_ENTITY_METHOD_INPUT RTM_ENTITY_METHOD_INPUT 구조체는 메서드를 호출할 때 클라이언트에 정보를 전달하는 데 사용됩니다.
RTM_ENTITY_METHOD_OUTPUT RTM_ENTITY_METHOD_OUTPUT 구조는 라우팅 테이블 관리자가 메서드를 호출할 때 호출 클라이언트에 정보를 전달하는 데 사용됩니다.
RTM_NET_ADDRESS RTM_NET_ADDRESS 구조는 주소 패밀리의 라우팅 테이블 관리자에게 주소 정보를 전달하는 데 사용됩니다. 주소 패밀리는 8바이트 미만의 연속 주소 마스크에서만 사용해야 합니다.
RTM_NEXTHOP_INFO RTM_NEXTHOP_INFO 구조는 라우팅 테이블 관리자와 다음 홉 정보를 교환하는 데 사용됩니다.
RTM_NEXTHOP_LIST RTM_NEXTHOP_LIST 구조에는 경로에서 동일한 비용 경로를 결정하는 데 사용되는 다음 홉 목록이 포함되어 있습니다.

RTM_PREF_INFO RTM_PREF_INFO 구조에는 두 경로를 비교할 때 사용되는 정보가 포함됩니다. 기본 설정 멤버의 값은 메트릭 멤버의 값보다 가중치가 더 높습니다.
RTM_REGN_PROFILE RTM_REGN_PROFILE 구조에는 등록 프로세스 중에 반환되는 정보가 포함됩니다. 이 정보는 이후 함수 호출(예: RtmGetEnumRoutes 호출에 의해 반환될 수 있는 최대 경로 수)에 사용됩니다.
RTM_ROUTE_INFO RTM_ROUTE_INFO 구조는 라우팅 테이블 관리자와 경로 정보를 교환하는 데 사용됩니다. 읽기 전용 정보를 변경하지 마세요.

열거형

RTM_EVENT_TYPE RTM_EVENT_TYPE 열거형은 라우팅 테이블 관리자가 RTM_EVENT_CALLBACK 콜백 사용에 대해 클라이언트에 알릴 수 있는 이벤트를 열거합니다.

rtscom.h 헤더

아티클2023. 08. 25.

이 헤더는 태블릿 PC에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Tablet PC

rtscom.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDynamicRenderer RealTimeStylus 클래스 개체에서 데이터를 처리할 때 태블릿 펜 데이터를 실시간으로 표시합니다.
IGestureRecognizer 제스처를 인식하고 입력 큐에 제스처 데이터를 추가하여 이벤트에 반응합니다.
IRealTimeStylus 디지타이저의 스타일러스 패킷 데이터를 실시간으로 처리합니다.
IRealTimeStylus2 IRealTimeStylus 인터페이스를 확장합니다.
IRealTimeStylus3 IRealTimeStylus3 인터페이스를 사용하면 실시간 스타일러스에 멀티 터치를 사용할 수 있습니다.
IRealTimeStylusSynchronization RealTimeStylus 클래스 개체에 대한 액세스를 동기화합니다.
IStrokeBuilder 인터페이스를 사용하여 패킷 데이터에서 프로그래밍 방식으로 스트로크를 만듭니다.
IStylusAsyncPlugin RealTimeStylus Class 개체의 비동기 플러그 인 컬렉션에 추가할 수 있는 비동기 플러그 인을 나타냅니다.
IStylusPlugin

RealTimeStylus 클래스 이벤트에 대한 알림을 수신하여 해당 이벤트를 기반으로 사용자 지정 처리를 수행할 수 있습니다.
IStylusSyncPlugin RealTimeStylus Class 개체의 동기 플러그 인 컬렉션에 추가할 수 있는 동기 플러그 인을 나타냅니다.

구조체

StylusInfo 스타일러스에 대한 정보를 제공합니다.

열거형

RealTimeStylusDataInterest 플러그 인이 수신하는 이벤트 알림을 지정하기 위해 플러그 인에서 사용하는 값을 정의합니다.
RealTimeStylusLockType RealTimeStylus Class 개체의 멤버 및 속성을 수정하지 않도록 보호하는 RealTimeStylus Class 개체 내의 잠금을 지정합니다.
StylusQueue 스타일러스 데이터가 추가되는 큐를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

rtworkq.h 헤더

아티클2023. 08. 25.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

rtworkq.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IRtwqAsyncCallback 비동기 메서드가 완료되면 애플리케이션에 알리기 위한 콜백 인터페이스입니다. (IRtwqAsyncCallback)
IRtwqAsyncResult 비동기 작업의 결과에 대한 정보를 제공합니다. (IRtwqAsyncResult)
IRtwqPlatformEvents 이벤트 관련 플랫폼 작업 큐를 제공합니다.

Functions

RtwqAddPeriodicCallback 고정 간격으로 호출할 콜백 함수를 설정합니다. (RtwqAddPeriodicCallback)
RtwqAllocateSerialWorkQueue 작업 항목을 직렬화하도록 보장되는 다른 작업 큐 위에 가상 작업 큐를 만듭니다. 직렬 작업 큐는 기존 다중 스레드 작업 큐를 래핑합니다. 직렬 작업 큐는 FIFO(선점) 실행 순서를 적용합니다.
RtwqAllocateWorkQueue 새 작업 큐를 만듭니다. (RtwqAllocateWorkQueue)
RtwqBeginRegisterWorkQueueWithMMCSS

작업 큐를 MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업에 연결합니다. (RtwqBeginRegisterWorkQueueWithMMCSS)
RtwqBeginUnregisterWorkQueueWithMMCSS MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업에서 작업 큐를 등록 취소합니다. (RtwqBeginUnregisterWorkQueueWithMMCSS)
RtwqCancelDeadline 이전에 RtwqSetDeadline으로 설정된 마감일을 취소합니다.
RtwqCancelWorkItem RtwqScheduleWorkItem으로 예약된 비동기 작업을 취소하려고 시도합니다.
RtwqCreateAsyncResult 비동기 결과 개체를 만듭니다. 비동기 메서드를 구현하는 경우 이 함수를 사용합니다. (RtwqCreateAsyncResult)
RtwqEndRegisterWorkQueueWithMMCSS 작업 큐를 MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업과 연결하는 비동기 요청을 완료합니다. (RtwqEndRegisterWorkQueueWithMMCSS)
RtwqGetWorkQueueMMCSSClass 현재 이 작업 큐와 연결된 MMCSS(멀티미디어 클래스 Scheduler Service) 클래스를 검색합니다. (RtwqGetWorkQueueMMCSSClass)
RtwqGetWorkQueueMMCSSPriority 작업 큐의 상대 스레드 우선 순위를 가져옵니다. (RtwqGetWorkQueueMMCSSPriority)
RtwqGetWorkQueueMMCSSTaskId 현재 이 작업 큐와 연결된 MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업 식별자를 검색합니다. (RtwqGetWorkQueueMMCSSTaskId)
RtwqInvokeCallback 콜백 메서드를 호출하여 비동기 작업을 완료합니다. (RtwqInvokeCallback)
RtwqJoinWorkQueue 작업 큐를 입력/출력(I/O) 핸들과 연결합니다.
RtwqLockPlatform

플랫폼에 보류 중인 비동기 항목이 있음을 나타내는 참조를 추가합니다. 활성 비동기 항목이 있는 경우 RtwqShutdown 함수를 차단합니다.
RtwqLockSharedWorkQueue 공유 작업 큐를 가져오고 잠가줍니다. (RtwqLockSharedWorkQueue)
RtwqLockWorkQueue 작업 큐를 잠근다. (RtwqLockWorkQueue)
RtwqPutWaitingWorkItem 이벤트가 신호를 받을 때까지 기다리는 작업 항목을 큐에 대기합니다. (RtwqPutWaitingWorkItem)
RtwqPutWorkItem 작업 큐에 비동기 작업을 배치합니다. (RtwqPutWorkItem)
RtwqRegisterPlatformEvents 앱이 RtwqStartup 및 RtwqShutdown 함수를 수신 대기하도록 설정합니다.
RtwqRegisterPlatformWithMMCSS 표준 플랫폼 작업 큐를 MMCSS(멀티미디어 클래스 스케줄러 서비스)에 등록합니다.
RtwqRemovePeriodicCallback RtwqAddPeriodicCallback 함수에서 설정한 콜백 함수를 취소합니다.
RtwqScheduleWorkItem 지정된 간격 후에 완료될 비동기 작업을 예약합니다. (RtwqScheduleWorkItem)
RtwqSetDeadline 작업 큐의 작업을 완료해야 하는 최종 기한을 설정합니다. (RtwqSetDeadline)
RtwqSetDeadline2 작업 큐의 작업을 완료해야 하는 최종 기한을 설정합니다. (RtwqSetDeadline2)
RtwqSetLongRunning 앱이 이 작업 큐에서 장기 실행 작업이 발생한다는 힌트를 제출한다는 것을 나타냅니다.
RtwqShutdown

플랫폼을 종료합니다. RtwqStartup을 호출할 때마다 이 함수를 한 번 호출합니다. 작업 큐 스레드에서 이 함수를 호출하지 마세요.
RtwqStartup 플랫폼을 초기화합니다.
RtwqUnjoinWorkQueue 입력/출력(I/O) 핸들에서 작업 큐를 연결 해제합니다.
RtwqUnlockPlatform RtwqLockPlatform 함수에 대한 호출로 잠긴 플랫폼의 잠금을 해제합니다.
RtwqUnlockWorkQueue 작업 큐의 잠금을 해제합니다. (RtwqUnlockWorkQueue)
RtwqUnregisterPlatformEvents 콜백 플랫폼에서 수신기 이벤트를 등록 취소합니다.
RtwqUnregisterPlatformFromMMCSS MMCSS(멀티미디어 클래스 스케줄러 서비스) 작업에서 플랫폼 작업 큐의 등록을 취소합니다.

열거형

RTWQ_WORKQUEUE_TYPE 만들 RtwqAllocateWorkQueue 함수의 작업 큐 유형을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

sas.h 헤더

아티클2023. 08. 25.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

sas.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

SendSAS SAS(보안 주의 시퀀스)를 시뮬레이션합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

sbtsv.h 헤더

아티클2023. 03. 08.

이 헤더는 원격 데스크톱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 데스크톱 서비스

sbtsv.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ITsSbBaseNotifySink 상태 및 오류 메시지를 보고하는 메서드를 RD 연결 브로커(원격 데스크톱 연결 브로커)에 노출합니다.
ITsSbClientConnection RDC(원격 데스크톱 연결) 클라이언트에서 들어오는 연결 요청에 대한 상태 정보를 저장하는 메서드 및 속성을 노출합니다.
ITsSbClientConnectionPropertySet 클라이언트 연결의 사용자 지정 속성을 적절하게 정의하는 데 사용할 수 있습니다.
ITsSbEnvironment 대상 컴퓨터를 호스트하는 환경에 대한 정보가 포함된 메서드 및 속성을 노출합니다. 이 인터페이스는 가상 머신을 호스트하는 물리적 서버에 대한 정보를 저장하는 데 사용할 수 있습니다.
ITsSbEnvironmentPropertySet 대상 컴퓨터를 적절하게 호스트하는 환경의 사용자 지정 속성을 정의하는 데 사용할 수 있습니다.
ITsSbFilterPluginStore 플러그 인 저장소 필터링
ITsSbGenericNotifySink 완료를 보고하고 RD 연결 브로커(원격 데스크톱 연결 브로커)에서 대기 시간을 가져오는 메서드를 노출합니다.

ITsSbGlobalStore RD 연결 브로커(원격 데스크톱 연결 브로커) 저장소에 추가된 대상 컴퓨터, 세션, 환경 및 팜을 쿼리하는 메서드를 노출합니다.
ITsSbLoadBalanceResult 부하 분산 알고리즘에서 반환된 대상 이름을 저장하는 메서드 및 속성을 노출합니다.
ITsSbLoadBalancing 사용자 지정 부하 분산 알고리즘을 제공하는 데 사용할 수 있는 메서드를 노출합니다.
ITsSbLoadBalancingNotifySink 부하 분산 알고리즘의 결과를 반환하는 메서드를 RD 연결 브로커(원격 데스크톱 연결 브로커)에 노출합니다.
ITsSbOrchestration RD 연결 브로커(원격 데스크톱 연결 브로커)가 클라이언트를 리디렉션하기 전에 대상이 준비되었는지 확인하는 데 사용하는 메서드를 노출합니다.
ITsSbOrchestrationNotifySink 대상이 연결에 대해 준비되면 ITsSbTarget 개체를 RD 연결 브로커(원격 데스크톱 연결 브로커)에 반환하는 메서드를 노출합니다.
ITsSbPlacement 환경을 준비하는 메서드(가상 머신을 호스트하는 컴퓨터)를 노출합니다.
ITsSbPlacementNotifySink 환경에 대한 정보를 반환하는 메서드를 RD 연결 브로커(원격 데스크톱 연결 브로커)에 노출합니다.
ITsSbPlugin 플러그 인을 초기화하고 종료하는 메서드를 노출합니다.
ITsSbPluginNotifySink RD 연결 브로커(원격 데스크톱 연결 브로커)에 플러그 인의 초기화 또는 종료를 알리는 메서드를 노출합니다.
ITsSbPluginPropertySet 사용자 지정 플러그 인 속성을 적절하게 정의하는 데 사용할 수 있습니다.

ITsSbPropertySet 사용자 지정 속성을 적절하게 정의하는 데 사용할 수 있습니다.
ITsSbProvider 원격 데스크톱 가상화에 사용되는 개체의 기본 구현을 만드는 메서드를 노출합니다.
ITsSbProvisioning 가상 머신을 만들고 유지 관리하는 메서드를 노출합니다.
ITsSbProvisioningPluginNotifySink RD 연결 브로커(원격 데스크톱 연결 브로커)에 가상 머신 프로비저닝에 대해 알리는 메서드를 노출합니다.
ITsSbResourceNotification RD 연결 브로커(원격 데스크톱 연결 브로커)가 세션, 대상 및 클라이언트 연결 개체에서 발생하는 상태 변경 내용을 플러그 인에 알리는 데 사용하는 메서드를 노출합니다. (ITsSbResourceNotification)
ITsSbResourceNotificationEx RD 연결 브로커(원격 데스크톱 연결 브로커)가 세션, 대상 및 클라이언트 연결 개체에서 발생하는 상태 변경 내용을 플러그 인에 알리는 데 사용하는 메서드를 노출합니다. (ITsSbResourceNotificationEx)
ITsSbResourcePlugin RD 연결 브로커(원격 데스크톱 연결 브로커)의 기능을 확장하는 메서드를 노출합니다.
ITsSbResourcePluginStore 리소스 플러그 인이 세션 및 대상과 같은 개체를 저장할 수 있도록 하는 메서드를 노출합니다.
ITsSbServiceNotification RD 연결 브로커(원격 데스크톱 연결 브로커)가 RD 연결 브로커 자체에서 발생하는 상태 변경 내용을 플러그 인에 알리는 데 사용하는 메서드를 노출합니다.
ITsSbSession 사용자 세션에 대한 정보를 저장하는 속성을 노출합니다.
ITsSbTarget 대상에 대한 구성 및 상태 정보를 저장하는 속성을 노출합니다.

ITsSbTargetPropertySet 이 인터페이스에서 파생하여 사용자 지정 대상 속성 집합을 정의합니다.
ITsSbTaskInfo 원격 데스크톱 연결 브로커가 플러그 인의 큐를 설정하는 데 사용하는 속성을 노출합니다.
ITsSbTaskPlugin 원격 데스크톱 연결 브로커 플러그 인에 대한 작업 큐를 업데이트하는 메서드를 노출합니다.
ITsSbTaskPluginNotifySink 태스크에 대한 상태 및 오류 메시지를 보고하는 메서드를 RD 연결 브로커(원격 데스크톱 연결 브로커)에 노출합니다.

열거형

TS_SB_SORT_BY 정렬 순서를 지정합니다. EnumerateTargets 메서드에서 매개 변수로 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

scclient.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows Media 디바이스 관리자 11 SDK에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Media 디바이스 관리자 11 SDK

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

scesvc.h 헤더

아티클2023. 03. 08.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

scesvc.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ISceSvcAttachmentData ISceSvcAttachmentData 인터페이스는 보안 구성 스냅인에서 지정된 보안 서비스에 대한 구성 및 분석 데이터를 검색합니다.
ISceSvcAttachmentPersistInfo ISceSvcAttachmentPersistInfo 인터페이스는 첨부 파일 스냅인에서 수정된 구성 또는 분석 정보를 검색합니다.

콜백 함수

PFSCE_FREE_INFO PFSCE_QUERY_INFO 호출할 때 보안 구성 도구 집합에서 할당한 버퍼의 메모리를 해제합니다.
PFSCE_LOG_INFO 구성 로그 파일 또는 분석 로그 파일에 메시지를 기록합니다.
PFSCE_QUERY_INFO 보안 구성 파일 또는 분석 데이터베이스에서 서비스 관련 정보를 쿼리합니다.
PFSCE_SET_INFO 서비스별 구성 및 분석 정보를 설정하거나 덮어씁니다.

구조체

SCESVC_ANALYSIS_INFO 분석 정보를 포함합니다.
SCESVC_ANALYSIS_LINE SCESVC_ANALYSIS_LINE 구조체에는 SCESVC_ANALYSIS_INFO 구조체로 지정된 특정 줄의 키, 값 및 값 길이가 포함됩니다.
SCESVC_CALLBACK_INFO SCESVC_CALLBACK_INFO 구조에는 쿼리, 설정 및 사용 가능한 정보에 대한 불투명 데이터베이스 핸들 및 콜백 함수 포인터가 포함되어 있습니다.
SCESVC_CONFIGURATION_INFO SCESVC_CONFIGURATION_INFO 구조는 서비스에 대한 구성 정보를 제공합니다. 이 구조는 구성 정보를 지정할 때 PFSCE_QUERY_INFO 및 PFSCE_SET_INFO 함수에서 사용됩니다.
SCESVC_CONFIGURATION_LINE SCESVC_CONFIGURATION_LINE 구조에는 구성 데이터 줄에 대한 정보가 포함됩니다. SCESVC_CONFIGURATION_INFO 구조체에서 사용됩니다.

열거형

SCESVC_INFO_TYPE SCESVC_INFO_TYPE 열거형은 PFSCE_QUERY_INFO 및 PFSCE_SET_INFO 보안 데이터베이스에서 요청되거나 보안 데이터베이스에 전달된 정보의 유형을 나타내는 데 사용됩니다. 다음 값 중 하나일 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

schannel.h 헤더

아티클2023. 03. 08.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

schannel.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

SslCrackCertificate 지정된 인증서 BLOB에 포함된 정보가 포함된 X509Certificate 구조를 반환합니다.
SslEmptyCacheA Schannel 캐시에서 지정된 문자열을 제거합니다. (ANSI)
SslEmptyCacheW Schannel 캐시에서 지정된 문자열을 제거합니다. (유니코드)
SslFreeCertificate SslCrackCertificate 함수에 대한 이전 호출에 의해 할당된 인증서를 해제합니다.
SslGetServerIdentity 서버의 ID를 가져옵니다.

구조체

CRYPTO_SETTINGS 비활성화된 암호화 설정을 나타냅니다.
SCH_CRED_PUBLIC_CERTCHAIN SCH_CRED_PUBLIC_CERTCHAIN 구조에는 단일 인증서가 포함됩니다. 이 인증서에서 인증 체인을 빌드할 수 있습니다.

SCH_CRED_SECRET_PRIVKEY 클라이언트 또는 서버를 인증하는 데 필요한 프라이빗 키 정보를 포함합니다.
SCH_CREDENTIALS Schannel 자격 증명에 대한 데이터를 포함합니다. (SCH_CREDENTIALS)
SCHANNEL_ALERT_TOKEN InitializeSecurityContext(Schannel) 함수 또는 AcceptSecurityContext(Schannel) 함수에 대한 호출 대상으로 보낼 SSL(Secure Sockets Layer Protocol) 또는 TLS(전송 계층 보안 프로토콜) 경고를 생성합니다.
SCHANNEL_CERT_HASH Schannel에서 사용하는 인증서에 대한 해시 저장소 데이터를 포함합니다.
SCHANNEL_CERT_HASH_STORE Schannel이 커널 모드에서 사용하는 인증서에 대한 해시 저장소 데이터를 포함합니다.
SCHANNEL_CLIENT_SIGNATURE InitializeSecurityContext(Schannel) 함수를 호출할 때 클라이언트 인증서의 프라이빗 키에 액세스할 수 없는 경우 클라이언트 서명을 지정합니다(이 경우 함수는 SEC_I_SIGNATURE_NEEDED 반환).
SCHANNEL_CRED Schannel 자격 증명에 대한 데이터를 포함합니다. (SCHANNEL_CRED)
SCHANNEL_SESSION_TOKEN InitializeSecurityContext(Schannel) 함수 또는 AcceptSecurityContext(Schannel) 함수를 호출하여 만든 인증 세션에 다시 연결을 사용할 수 있는지 여부를 지정합니다.
SecPkgContext_CipherInfo 암호 정보 구조입니다. 이는 QueryContextAttributes(Schannel) 함수에서 SECPKG_ATTR_CIPHER_INFO ulAttribute에 의해 반환됩니다.
SecPkgContext_ConnectionInfo SecPkgContext_ConnectionInfo 구조에는 프로토콜 및 암호 정보가 포함됩니다. 이 구조체는 InitializeSecurityContext(Schannel) 함수에서 사용됩니다. 이 특성은 SSP(Schannel 보안 지원 공급자)에서만 지원됩니다.

SecPkgContext_EapKeyBlock EAP TLS 인증 프로토콜에서 사용하는 키 데이터를 포함합니다.
SecPkgContext_EapPrfInfo PRF(의사 권한 함수)를 지정하고 EAP(Extensible Authentication Protocol) TLS(전송 계층 보안 프로토콜) 인증 프로토콜에서 사용하는 키 데이터를 추출합니다.
SecPkgContext_EarlyStart SecPkgContext_EarlyStart 구조에는 보안 컨텍스트에서 False Start 기능을 사용할지 여부에 대한 정보가 포함되어 있습니다.
SecPkgContext_IssuerListInfoEx SecPkgContext_IssuerListInfoEx 구조에는 신뢰할 수 있는 CA(인증 기관) 목록이 있습니다.
SecPkgContext_KeyingMaterial SecPkgContext_KeyingMaterial 구조체입니다.
SecPkgContext_KeyingMaterialInfo SecPkgContext_KeyingMaterialInfo 구조에는 보안 컨텍스트에서 내보낼 수 있는 키 지정 자료에 대한 정보가 포함됩니다.
SecPkgContext_SessionAppData 세션 컨텍스트에 대한 애플리케이션 데이터를 저장합니다.
SecPkgContext_SessionInfo 세션이 다시 연결인지 여부를 지정하고 세션을 식별하는 값을 검색합니다.
SecPkgContext_SupportedSignatures Schannel 연결에서 지원하는 서명 알고리즘을 지정합니다.
TLS_PARAMETERS TLS 매개 변수 제한을 나타냅니다.
X509Certificate X.509 인증서를 나타냅니다.

열거형

eTlsAlgorithmUsage 암호화 설정을 사용하지 않도록 설정하는 데 사용되는 알고리즘을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

schedule.h 헤더

아티클2023. 03. 08.

이 헤더는 Active Directory Domain Services 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Active Directory Domain Services

schedule.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

일정 DsReplicaAdd 및 DsReplicaModify 함수와 함께 복제 원본에 대한 복제 일정 데이터를 포함하는 데 사용됩니다.
SCHEDULE_HEADER 복제 원본에 대한 복제 일정 데이터를 포함하는 데 사용됩니다.

scrnsave.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

scrnsave.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DefScreenSaverProc 화면 보호기 애플리케이션이 처리하지 않는 메시지에 대한 기본 처리를 제공합니다.
RegisterDialogClasses 화면 보호기 구성 대화 상자에 필요한 비표준 창 클래스를 등록합니다.
ScreenSaverConfigureDialog 화면 보호기 구성 대화 상자로 전송된 메시지를 받습니다. 사용자 구성을 허용하는 화면 보호기에서 이 함수를 정의해야 합니다.
ScreenSaverProc 지정된 화면 보호기 창으로 전송된 메시지를 받습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

scserver.h 헤더

아티클2023. 08. 25.

이 헤더는 Windows Media 디바이스 관리자 11 SDK에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Media 디바이스 관리자 11 SDK

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

sddl.h 헤더

아티클2023. 08. 25.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

sddl.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

ConvertSecurityDescriptorToStringSecurityDescriptorA 보안 설명자를 문자열 형식으로 변환합니다. 문자열 형식을 사용하여 보안 설명자를 저장하거나 전송할 수 있습니다. (ANSI)
ConvertSecurityDescriptorToStringSecurityDescriptorW 보안 설명자를 문자열 형식으로 변환합니다. 문자열 형식을 사용하여 보안 설명자를 저장하거나 전송할 수 있습니다. (유니코드)
ConvertSidToStringSidA SID(보안 식별자)를 표시, 스토리지 또는 전송에 적합한 문자열 형식으로 변환합니다. (ANSI)
ConvertSidToStringSidW SID(보안 식별자)를 표시, 스토리지 또는 전송에 적합한 문자열 형식으로 변환합니다. (유니코드)
ConvertStringSecurityDescriptorToSecurityDescriptorA 문자열 형식 보안 설명자를 유효한 기능 보안 설명자로 변환합니다. (ANSI)
ConvertStringSecurityDescriptorToSecurityDescriptorW 문자열 형식 보안 설명자를 유효한 기능 보안 설명자로 변환합니다. (유니코드)
ConvertStringSidToSidA 문자열 형식 SID(보안 식별자)를 유효한 기능 SID로 변환합니다. 이 함수를 사용하여 ConvertSidToStringSid 함수가 문자열 형식으로 변환된 SID를 검색할 수 있습니다. (ANSI)
ConvertStringSidToSidW

문자열 형식 SID(보안 식별자)를 유효한 기능 SID로 변환합니다. 이 함수를 사용하여 ConvertSidToStringSid 함수가 문자열 형식으로 변환된 SID를 검색할 수 있습니다. (유니코드)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

sdoias.h 헤더

아티클2023. 08. 25.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- CIMWin32 WMI 공급자
- Hyper-V WMI 공급자 클러스터링 장애 조치(failover)
- iSCSI 대상 서버 공급자
- 네트워크 정책 서버
- WhqlProvider 공급자
- Windows Installer 공급자
- wlbsprov
- NFS용 WMI 공급자

sdoias.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ISdo ISdo 인터페이스를 사용하여 SDO(서버 데이터 개체) 정보를 저장, 검색 및 업데이트합니다.
ISdoCollection ISdoCollection 인터페이스를 사용하여 SDO 개체 컬렉션을 조작합니다.
ISdoDictionaryOld ISdoDictionaryOld 인터페이스를 사용하여 RADIUS(원격 액세스 전화 접속 사용자 서비스) 특성의 사전을 조작합니다.
ISdoMachine ISdoMachine 인터페이스를 사용하여 SDO 컴퓨터에 연결하고, SDO 컴퓨터에 대한 정보를 가져오고, 다른 SDO 개체에 대한 인터페이스를 가져옵니다.
ISdoServiceControl ISdoServiceControl 인터페이스를 사용하여 SDO 컴퓨터에서 관리되는 서비스를 제어합니다.

열거형

ACCOUNTINGPROPERTIES ACCOUNTINGPROPERTIES 형식의 값은 기록되는 패킷 유형과 로그 파일의 특성을 제어하는 속성을 열거합니다.
ATTRIBUTEFILTER NPS 사전 특성의 사용 사례입니다.
ATTRIBUTEID ATTRIBUTEID 열거형 형식은 SDO API에서 지원하는 RADIUS 특성을 열거합니다.
ATTRIBUTEINFO ATTRIBUTEINFO 형식의 값은 지정된 특성의 특성을 열거합니다.
ATTRIBUTEPROPERTIES ATTRIBUTEPROPERTIES 형식의 값은 RADIUS 사전 특성에 대한 속성을 열거합니다.
ATTRIBUTERESTRICTIONS ATTRIBUTERESTRICTIONS 열거형 형식의 값은 특정 특성을 사용하는 방법에 대한 제한을 지정합니다.
ATTRIBUTESYNTAX ATTRIBUTESYNTAX 열거형 형식의 각 값은 가능한 특성 구문을 지정합니다.
AUTHENTICATION_TYPE AUTHENTICATION_TYPE 열거형 형식의 값은 인증 방법을 지정하는 데 사용됩니다.
CLIENTPROPERTIES CLIENTPROPERTIES 형식의 값은 RADIUS 클라이언트의 속성을 열거합니다. SDO 컴퓨터는 RADIUS 서버입니다.
CONDITIONPROPERTIES CONDITIONPROPERTIES 열거형 형식의 값은 NAP(네트워크 액세스 정책) 조건의 속성을 지정합니다.
DICTIONARYPROPERTIES DICTIONARYPROPERTIES 속성의 값은 특성 사전과 연결된 속성을 열거합니다.

IASCOMMONPROPERTIES IASCOMMONPROPERTIES 열거형 형식의 값은 모든 SDO 개체에 있는 속성을 열거합니다.
IASCOMPONENTPROPERTIES IASCOMPONENTPROPERTIES 열거형 형식의 값은 SDO 개체의 식별자를 열거합니다.
IASDATASTORE IASDATASTORE 열거형의 값은 SDO 데이터에 대한 가능한 스토리지 위치를 나타냅니다.
IASDOMAINTYPE IASDOMAINTYPE 열거형 형식의 값은 SDO 컴퓨터가 도메인의 일부인지 여부와 도메인 유형(있는 경우)을 지정합니다.
IASOSTYPE IASOSTYPE 열거형 형식의 값은 클라이언트가 인증을 요청하는 운영 체제 유형(SDO 컴퓨터)을 지정합니다.
IASPROPERTIES IASPROPERTIES 열거형 형식의 값은 NPS와 관련된 속성을 열거합니다.
IDENTITY_TYPE MS_ATTRIBUTE_IDENTITY_TYPE 사용할 수 있는 다양한 값입니다.
NAMESPROPERTIES NAMESPROPERTIES 열거형 형식의 값은 네트워크 액세스를 요청하는 사용자의 이름과 관련된 속성을 열거합니다.
NAPPROPERTIES NAPPROPERTIES 열거형 형식의 값은 NAP(네트워크 액세스 정책)에 대한 속성을 지정합니다.
NEW_LOG_FILE_FREQUENCY NEW_LOG_FILE_FREQUENCY 열거형 형식의 값은 새 로그 파일을 만드는 빈도를 지정합니다.
NTEVENTLOGPROPERTIES NTEVENTLOGPROPERTIES 열거형 형식의 값은 NT 이벤트 로그에 기록해야 하는 이벤트 유형을 열거합니다.

NTSAMPROPERTIES NTSAMPROPERTIES 열거형 형식의 값은 NT SAM(보안 계정 관리자)과 관련된 속성을 지정합니다.
POLICYPROPERTIES POLICYPROPERTIES 열거형 형식의 값은 NAP(네트워크 액세스 정책)의 속성을 열거합니다.
PROFILEPROPERTIES PROFILEPROPERTIES 열거형 형식은 프로필과 연결된 속성을 열거합니다.
PROTOCOLPROPERTIES PROTOCOLPROPERTIES 열거형 형식의 값은 인증 프로토콜의 속성을 열거합니다.
RADIUSPROPERTIES RADIUSPROPERTIES 열거형 형식의 값은 Microsoft RADIUS 프로토콜 SDO의 속성을 열거합니다.
RADIUSPROXYPROPERTIES RADIUSPROXYPROPERTIES 열거형 형식의 값은 RADIUS 프록시 서비스와 관련된 속성을 열거합니다.
RADIUSSERVERGROUPPROPERTIES RADIUSSERVERGROUPPROPERTIES 열거형 형식의 값은 RADIUS 서버 그룹의 속성을 열거합니다.
RADIUSSERVERPROPERTIES RADIUSSERVERPROPERTIES 열거형 형식의 값은 RADIUS 서버의 속성(즉, SDO 컴퓨터)을 열거합니다.
REMEDIATIONSERVERGROUPPROPERTIES REMEDIATIONSERVERGROUPPROPERTIES 열거형 형식의 값은 수정 서버 그룹의 속성을 열거합니다.
REMEDIATIONSERVERPROPERTIES REMEDIATIONSERVERPROPERTIES 열거형 형식의 값은 수정 서버의 속성을 열거합니다.
REMEDIATIONSERVERSPROPERTIES REMEDIATIONSERVERSPROPERTIES 열거형 형식의 값은 수정 서버 그룹 집합의 속성을 열거합니다.
SERVICE_TYPE

SERVICE_TYPE 열거형 형식의 값은 SDO API에서 관리되는 서비스 유형을 지정합니다.
SHV_COMBINATION_TYPE SHV_COMBINATION_TYPE 열거형 형식은 SHV(시스템 상태 유효성 검사기) 조합의 형식을 지정합니다.
SHVTEMPLATEPROPERTIES SHVTEMPLATEPROPERTIES 열거형 형식의 값은 SHV(시스템 상태 유효성 검사기) 템플릿의 속성을 열거합니다.
USERPROPERTIES USERPROPERTIES 열거형 형식의 값은 SDO API에서 지원하는 사용자 속성을 열거합니다.
VENDORPROPERTIES VENDORPROPERTIES 열거형 형식의 값은 vendors 컬렉션에 있는 개체의 속성을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

searchapi.h 헤더

아티클2023. 08. 25.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 레거시 Windows 환경 기능
- Windows 검색

searchapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumSearchRoots 카탈로그의 검색 루트(예: SystemIndex)를 열거하는 메서드를 제공합니다.
IEnumSearchScopeRules scope 규칙을 열거합니다.
IOpLockStatus 인덱싱하는 동안 항목에 대해 Microsoft WDS(Windows Desktop Search)에서 사용하는 기회 잠금을 검사 메서드를 제공합니다.
IProtocolHandlerSite 프로토콜 처리기의 IUrlAccessor 개체에 대한 메서드를 제공하여 URL 항목에 대한 적절한 필터에 대해 필터 디먼을 쿼리합니다.
IRowsetEvents 이벤트 알림을 받기 위한 메서드를 노출합니다.
IRowsetPrioritization 이 쿼리에서 지정한 scope 대한 현재 인덱서 우선 순위 수준을 설정하거나 검색합니다.
ISearchCatalogManager 다시 인덱싱 또는 시간 제한 설정과 같은 목적으로 검색 카탈로그를 관리하는 메서드를 제공합니다.
ISearchCatalogManager2

ISearchCatalogManager 인터페이스를 확장하여 검색 카탈로그를 관리합니다(예: 다시 인덱싱 또는 시간 제한 설정).
ISearchCrawlScopeManager 크롤링 및/또는 watch 컨테이너의 검색 엔진에 알리는 메서드와 크롤링 또는 감시 시 포함하거나 제외할 해당 컨테이너 아래의 항목을 제공합니다.
ISearchCrawlScopeManager2 ISearchCrawlScopeManager 인터페이스의 기능을 확장합니다.
ISearchItemsChangedSink 인덱싱된 항목의 변경 내용에 대한 알림을 제공합니다. 또한 변경된 항목에 대해 모니터링되는 계층적 scope 대한 알림을 제공합니다.
ISearchLanguageSupport 동의어 사전 정보에 액세스하는 메서드를 제공합니다.
ISearchManager Search Service 제어하는 메서드를 제공합니다. 이 인터페이스는 카탈로그 전체에서 검색 엔진에 영향을 주는 설정 및 개체를 관리합니다.
ISearchManager2 애플리케이션이 Windows Search 인덱서에서 사용자 지정 카탈로그를 만들고 삭제할 수 있도록 설정했습니다.
ISearchNotifyInlineSite Search Service 카탈로그 및 인덱스 상태 업데이트를 알림 공급자에게 보내는 데 사용하는 메서드를 제공합니다.
ISearchPersistentItemsChangedSink 항목 업데이트가 필요한 인덱서에 경고하는 변경 알림을 전달하는 메서드를 제공합니다.
ISearchProtocol IUrlAccessor 개체를 호출, 초기화 및 관리하는 메서드를 제공합니다. (ISearchProtocol)
ISearchProtocol2 IUrlAccessor 개체를 호출, 초기화 및 관리하는 메서드를 제공합니다. (ISearchProtocol2)
ISearchProtocolThreadContext

이 선택적 인터페이스를 사용하면 프로토콜 처리기가 프로토콜 호스트에서 필터링하는 데 사용되는 스레드에서 작업을 수행할 수 있습니다.
ISearchQueryHelper 사용자 입력에서 쿼리를 빌드하고, 쿼리를 Windows Search SQL로 변환하고, 연결 문자열을 가져와 창 검색 인덱스로의 연결을 초기화하는 메서드를 제공합니다.
ISearchRoot 검색 루트를 조작하는 메서드를 제공합니다. 속성 멤버에 대한 변경 내용은 검색 루트 아래에 있는 모든 URL에 적용됩니다. URL은 검색 루트 URL과 일치하거나 해당 URL의 계층적 자식인 경우 검색 루트 아래에 속합니다.
ISearchScopeRule 크롤링 및 인덱싱에 대한 scope 규칙을 정의하는 메서드를 제공합니다.
ISearchViewChangedSink 구현되지 않았습니다. (ISearchViewChangedSink)
IUrlAccessor 수집기가 필터 호스트에 URL을 제공하는 콘텐츠 원본의 개별 항목을 처리하는 메서드를 제공합니다.
IUrlAccessor2 IUrlAccessor 인터페이스의 기능을 확장합니다.
IUrlAccessor3 IUrlAccessor3::GetImpersonationSidBlobs 메서드를 사용하여 IUrlAccessor2 인터페이스의 기능을 확장하여 지정된 URL에 대한 SID(사용자 보안 식별자)를 식별합니다.
IUrlAccessor4 항목의 콘텐츠를 인덱싱해야 하는지 여부를 식별하는 IUrlAccessor4::ShouldIndexItemContent 메서드를 사용하여 IUrlAccessor3 인터페이스의 기능을 확장합니다.

구조체

AUTHENTICATION_INFO

콘텐츠 액세스에 대한 보안 인증 정보를 설명합니다.
INCREMENTAL_ACCESS_INFO 마지막 액세스 날짜 및 수정 시간과 같은 증분 크롤링에서 사용하는 액세스 정보를 포함합니다.
ITEM_INFO 현재 항목에 대한 IUrlAccessor 개체에 전달된 정보를 포함합니다. 예를 들어 애플리케이션 이름 및 카탈로그 이름입니다.
PROXY_INFO 프록시에 대한 정보를 저장합니다. ISearchProtocol에서 사용됩니다.
SEARCH_COLUMN_PROPERTIES 이 구조체는 구현되지 않습니다.
SEARCH_ITEM_CHANGE 인덱싱된 항목의 변경 내용을 지정합니다.
SEARCH_ITEM_INDEXING_STATUS 인덱싱할 문서의 상태 설명합니다.
SEARCH_ITEM_PERSISTENT_CHANGE 인덱싱할 항목에서 발생한 변경의 종류에 대한 정보를 포함합니다. 이 구조체는 ISearchPersistentItemsChangedSink::OnItemsChanged 메서드와 함께 사용하여 변경된 내용에 대한 정보를 인덱서에 전달합니다.
TIMEOUT_INFO 연결 및 데이터에 대한 제한 시간 값을 저장합니다.

열거형

AUTH_TYPE 콘텐츠 액세스에 대한 인증 유형을 설명합니다.
CatalogPausedReason

ISearchCatalogManager::GetCatalogStatus에서 카탈로그가 일시 중지된 이유를 확인하는 데 사용됩니다.
CatalogStatus ISearchCatalogManager::GetCatalogStatus에서 카탈로그의 현재 상태를 확인하는 데 사용됩니다.
CLUSION_REASON 이러한 플래그는 URL이 현재 크롤링 scope 포함되거나 제외되는 이유를 열거합니다.
FOLLOW_FLAGS 크롤링 또는 인덱싱 시 동작을 정의하는 데 사용됩니다. 이러한 플래그는 ISearchCrawlScopeManager::AddDefaultScopeRule 및 ISearchCrawlScopeManager::AddUserScopeRule 메서드에서 사용됩니다.
PRIORITY_LEVEL IRowsetPrioritization 인터페이스에서 쿼리에 지정된 scope 대한 현재 인덱서 우선 순위 수준을 설정하거나 검색하는 데 사용됩니다.
PROXY_ACCESS ISearchManager에서 상태 프록시 사용에 사용됩니다.
ROWSETEVENT_ITEMSTATE 행 집합의 검색 조건과 일치하는 항목이 현재 해당 행 집합에 있는지 여부를 설명합니다.
ROWSETEVENT_TYPE 행 집합의 데이터에 대한 변경 형식을 설명합니다.
SEARCH_INDEXING_PHASE 현재 검색 인덱싱 단계의 상태 지정합니다.
SEARCH_KIND_OF_CHANGE 원본 싱크가 항목이 변경되었음을 클라이언트에 알립니다.
SEARCH_NOTIFICATION_PRIORITY 변경된 항목을 처리하는 우선 순위를 나타냅니다.
SEARCH_QUERY_SYNTAX 쿼리 구문의 형식을 지정합니다. (SEARCH_QUERY_SYNTAX)

SEARCH_TERM_EXPANSION 검색어에 대한 와일드카드 옵션을 나타냅니다. ISearchQueryHelper::get_QueryTermExpansion 및 ISearchQueryHelper::p ut_QueryTermExpansion 메서드에서 사용됩니다.
tagPRIORITIZE_FLAGS PrioritizeMatchingURL에서 인덱서가 이전에 인덱싱하지 못한 항목을 처리하는 방법을 지정하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

secext.h 헤더

아티클2023. 08. 25.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

secext.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

GetComputerObjectNameA 지정된 형식으로 로컬 컴퓨터의 이름을 검색합니다. (ANSI)
GetComputerObjectNameW 지정된 형식으로 로컬 컴퓨터의 이름을 검색합니다. (유니코드)
GetUserNameExA 호출 스레드와 연결된 사용자 또는 기타 보안 주체의 이름을 검색합니다. 반환된 이름의 형식을 지정할 수 있습니다. (ANSI)
GetUserNameExW 호출 스레드와 연결된 사용자 또는 기타 보안 주체의 이름을 검색합니다. 반환된 이름의 형식을 지정할 수 있습니다. (유니코드)
TranslateNameA 디렉터리 서비스 개체 이름을 한 형식에서 다른 형식으로 변환합니다. (ANSI)
TranslateNameW 디렉터리 서비스 개체 이름을 한 형식에서 다른 형식으로 변환합니다. (유니코드)

열거형

EXTENDED_NAME_FORMAT

디렉터리 서비스 개체 이름의 형식을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

securityappcontainer.h 헤더

아티클2023. 08. 25.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

securityappcontainer.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

GetAppContainerNamedObjectPath 앱 컨테이너의 명명된 개체 경로를 검색합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

securitybaseapi.h 헤더

아티클2023. 03. 16.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 이벤트 추적
- 보안 및 ID

securitybaseapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AccessCheck 보안 설명자가 액세스 토큰으로 식별된 클라이언트에 지정된 액세스 권한 집합을 부여할지 여부를 결정합니다. (AccessCheck)
AccessCheckAndAuditAlarmW 보안 설명자가 호출 스레드에서 가장하는 클라이언트에 대해 지정된 액세스 권한 집합을 부여할지 여부를 결정합니다.
AccessCheckByType 보안 설명자가 액세스 토큰으로 식별된 클라이언트에 지정된 액세스 권한 집합을 부여할지 여부를 결정합니다. (AccessCheckByType)
AccessCheckByTypeAndAuditAlarmW 보안 설명자가 호출 스레드에서 가장하는 클라이언트에 대해 지정된 액세스 권한 집합을 부여할지 여부를 결정합니다. (AccessCheckByTypeAndAuditAlarmW)
AccessCheckByTypeResultList 보안 설명자가 액세스 토큰으로 식별된 클라이언트에 지정된 액세스 권한 집합을 부여할지 여부를 결정합니다. (AccessCheckByTypeResultList)
AccessCheckByTypeResultListAndAuditAlarmByHandleW AccessCheckByTypeResultListAndAuditAlarmByHandleW(유니코드) 함수(securitybaseapi.h)는 보안 설명자가 호출 스레드가 가장하는 클라이언트에 대한 액세스 권한을 부여할지 여부를 결정합니다.

AccessCheckByTypeResultListAndAuditAlarmW 보안 설명자가 호출 스레드에서 가장하는 클라이언트에 대해 지정된 액세스 권한 집합을 부여할지 여부를 결정합니다. (AccessCheckByTypeResultListAndAuditAlarmW)
AddAccessAllowedAce 액세스 제어 목록(ACL)에 ACE(액세스 허용 액세스 제어 항목)를 추가합니다. 지정된 SID(보안 식별자)에 대한 액세스 권한이 부여됩니다.
AddAccessAllowedAceEx DACL(임의 액세스 제어 목록)의 끝에 ACE(액세스 허용 액세스 제어 항목)를 추가합니다. (AddAccessAllowedAceEx)
AddAccessAllowedObjectAce DACL(임의 액세스 제어 목록)의 끝에 ACE(액세스 허용 액세스 제어 항목)를 추가합니다. (AddAccessAllowedObjectAce)
AddAccessDeniedAce 액세스 거부 ACE(액세스 거부 액세스 제어 항목)를 ACL(액세스 제어 목록)에 추가합니다. 지정된 SID(보안 식별자)에 대한 액세스가 거부됩니다.
AddAccessDeniedAceEx DACL(임의 액세스 제어 목록)의 끝에 ACE(액세스 거부 액세스 제어 항목)를 추가합니다.
AddAccessDeniedObjectAce DACL(임의 액세스 제어 목록)의 끝에 ACE(액세스 거부 액세스 제어 항목)를 추가합니다. 새 ACE는 개체 또는 개체의 속성 집합 또는 속성에 대한 액세스를 거부할 수 있습니다.
AddAce 지정된 ACL(액세스 제어 목록)에 하나 이상의 ACL(액세스 제어 항목)을 추가합니다.
AddAuditAccessAce 시스템 ACL(액세스 제어 목록)에 ACE(시스템 감사 액세스 제어 항목)를 추가합니다. 지정된 SID(보안 식별자)의 액세스가 감사됩니다.
AddAuditAccessAceEx SACL(시스템 액세스 제어 목록)의 끝에 ACE(시스템 감사 액세스 제어 항목)를 추가합니다. (AddAuditAccessAceEx)

AddAuditAccessObjectAce SACL(시스템 액세스 제어 목록)의 끝에 ACE(시스템 감사 액세스 제어 항목)를 추가합니다. (AddAuditAccessObjectAce)
AddMandatoryAce 지정된 SACL(시스템 액세스 제어 목록)에 ACE(SYSTEM_MANDATORY_LABEL_ACE 액세스 제어 항목)를 추가합니다.
AddResourceAttributeAce SACL(시스템 액세스 제어 목록)의 끝에 ACE(SYSTEM_RESOURCE_ATTRIBUTE_ACEaccess 컨트롤 항목)를 추가합니다.
AddScopedPolicyIDAce SACL(시스템 액세스 제어 목록)의 끝에 ACE(SYSTEM_SCOPED_POLICY_ID_ACEaccess 컨트롤 항목)를 추가합니다.
AdjustTokenGroups 지정된 액세스 토큰에 이미 있는 그룹을 사용하거나 사용하지 않도록 설정합니다. 액세스 토큰에서 그룹을 사용하거나 사용하지 않도록 설정하려면 TOKEN_ADJUST_GROUPS 액세스가 필요합니다.
AdjustTokenPrivileges 지정된 액세스 토큰에서 권한을 사용하거나 사용하지 않도록 설정합니다. 액세스 토큰에서 권한을 사용하거나 사용하지 않도록 설정하려면 TOKEN_ADJUST_PRIVILEGES 액세스가 필요합니다.
AllocateAndInitializeSid 최대 8개의 하위 권한이 있는 SID(보안 식별자)를 할당하고 초기화합니다.
AllocateLocallyUniqueId LUID(로컬 고유 식별자)를 할당합니다.
AreAllAccessesGranted 요청된 액세스 권한 집합이 부여되었는지 여부를 확인합니다. 액세스 권한은 액세스 마스크에서 비트 플래그로 표시됩니다.
AreAnyAccessesGranted 요청된 액세스 권한 집합이 부여되었는지 여부를 테스트합니다. 액세스 권한은 액세스 마스크에서 비트 플래그로 표시됩니다.

CheckTokenCapability 지정된 토큰의 기능을 확인합니다.
CheckTokenMembership 액세스 토큰에서 지정된 SID(보안 식별자)를 사용할 수 있는지 여부를 결정합니다.
CheckTokenMembershipEx 지정된 토큰에서 지정된 SID를 사용할 수 있는지 여부를 결정합니다.
ConvertToAutoInheritPrivateObjectSecurity 보안 설명자와 ACL(액세스 제어 목록)을 상속 가능한 ACL(액세스 제어 항목)의 자동 전파를 지원하는 형식으로 변환합니다.
CopySid SID(보안 식별자)를 버퍼에 복사합니다.
CreatePrivateObjectSecurity 새 프라이빗 개체에 대한 자체 상대 보안 설명자를 할당하고 초기화합니다. 보호된 서버는 새 프라이빗 개체를 만들 때 이 함수를 호출합니다.
CreatePrivateObjectSecurityEx 이 함수를 호출하는 리소스 관리자가 만든 새 프라이빗 개체에 대한 자체 상대 보안 설명자를 할당하고 초기화합니다. (CreatePrivateObjectSecurityEx)
CreatePrivateObjectSecurityWithMultipleInheritance 이 함수를 호출하는 리소스 관리자가 만든 새 프라이빗 개체에 대한 자체 상대 보안 설명자를 할당하고 초기화합니다. (CreatePrivateObjectSecurityWithMultipleInheritance)
CreateRestrictedToken 기존 액세스 토큰의 제한된 버전인 새 액세스 토큰을 만듭니다. 제한된 토큰에는 SID(보안 식별자), 삭제된 권한 및 제한 SID 목록이 있을 수 있습니다.
CreateWellKnownSid 미리 정의된 별칭에 대한 SID를 만듭니다.
CveEventWrite 사용자 모드 애플리케이션에서 보안 취약성 악용 시도가 감지될 때 이벤트를 게시하기 위한 추적 함수입니다.

DeleteAce ACL(액세스 제어 목록)에서 ACE(액세스 제어 항목)를 삭제합니다.
DeriveCapabilitySidsFromName 이 함수는 기능 이름에서 두 개의 SID 배열을 생성합니다. 하나는 NT 기관이 있는 배열 그룹 SID이고, 다른 하나는 AppAuthority를 사용하는 기능 SID 배열입니다.
DestroyPrivateObjectSecurity 프라이빗 개체의 보안 설명자를 삭제합니다.
DuplicateToken 이미 존재하는 토큰을 복제하는 새 액세스 토큰을 만듭니다.
DuplicateTokenEx 기존 토큰을 복제하는 새 액세스 토큰을 만듭니다. 이 함수는 기본 토큰 또는 가장 토큰을 만들 수 있습니다.
EqualDomainSid 두 SID가 동일한 도메인에 있는지 여부를 확인합니다.
EqualPrefixSid 같음의 두 SID(보안 식별자) 접두사 값을 테스트합니다. SID 접두사는 마지막 하위 인증 값을 제외한 전체 SID입니다.
EqualSid 같음의 두 SID(보안 식별자) 값을 테스트합니다. 동일한 것으로 간주되려면 두 SID가 정확히 일치해야 합니다.
FindFirstFreeAce ACL(액세스 제어 목록)에서 첫 번째 무료 바이트에 대한 포인터를 검색합니다.
FreeSid AllocateAndInitializeSid 함수를 사용하여 이전에 할당된 SID(보안 식별자)를 해제합니다.
GetAce ACL(액세스 제어 목록)에서 ACE(액세스 제어 항목)에 대한 포인터를 가져옵니다.

GetAclInformation ACL(액세스 제어 목록)에 대한 정보를 검색합니다.
GetAppContainerAce 패키지 또는 기능 SID가 있는지 여부를 나타내는 값을 검색합니다.
GetCachedSigningLevel 캐시된 서명 수준을 검색합니다.
GetFileSecurityW 파일 또는 디렉터리의 보안에 대한 지정된 정보를 가져옵니다. 가져온 정보는 호출자의 액세스 권한 및 권한에 의해 제한됩니다. (GetFileSecurityW)
GetKernelObjectSecurity 커널 개체를 보호하는 보안 설명자의 복사본을 검색합니다.
GetLengthSid 유효한 SID(보안 식별자)의 길이(바이트)를 반환합니다.
GetPrivateObjectSecurity 개인 개체의 보안 설명자에서 정보를 검색합니다.
GetSecurityDescriptorControl 보안 설명자 컨트롤 및 수정 정보를 검색합니다.
GetSecurityDescriptorDacl 지정된 보안 설명자에서 DACL(임의 액세스 제어 목록)에 대한 포인터를 검색합니다.
GetSecurityDescriptorGroup 보안 설명자에서 기본 그룹 정보를 검색합니다.
GetSecurityDescriptorLength 구조적으로 유효한 보안 설명자의 길이(바이트)를 반환합니다. 길이에는 연결된 모든 구조체의 길이가 포함됩니다.
GetSecurityDescriptorOwner 보안 설명자에서 소유자 정보를 검색합니다.

GetSecurityDescriptorRMControl 리소스 관리자 컨트롤 비트를 검색합니다.
GetSecurityDescriptorSacl 지정된 보안 설명자에서 SACL(시스템 액세스 제어 목록)에 대한 포인터를 검색합니다.
GetSidIdentifierAuthority 지정된 SID(보안 식별자)의 SID_IDENTIFIER_AUTHORITY 구조체에 대한 포인터를 반환합니다.
GetSidLengthRequired 지정된 수의 하위 권한이 있는 SID를 저장하는 데 필요한 버퍼의 길이(바이트)를 반환합니다.
GetSidSubAuthority SID(보안 식별자)에서 지정된 하위 권한에 대한 포인터를 반환합니다. 하위 인증 값은 RID(상대 식별자)입니다.
GetSidSubAuthorityCount 하위 권한 수를 포함하는 SID(보안 식별자) 구조체의 멤버에 대한 포인터를 반환합니다.
GetTokenInformation 액세스 토큰에 대한 지정된 유형의 정보를 검색합니다. 호출 프로세스에는 정보를 얻기 위한 적절한 액세스 권한이 있어야 합니다.
GetWindowsAccountDomainSid SID(보안 식별자)를 수신하고 해당 SID의 도메인을 나타내는 SID를 반환합니다.
ImpersonateAnonymousToken 지정된 스레드가 시스템의 익명 로그온 토큰을 가장할 수 있도록 합니다.
ImpersonateLoggedOnUser 호출 스레드가 로그온한 사용자의 보안 컨텍스트를 가장할 수 있도록 합니다. 사용자는 토큰 핸들로 표시됩니다.
ImpersonateSelf 호출 프로세스의 보안 컨텍스트를 가장하는 액세스 토큰을 가져옵니다. 토큰은 호출 스레드에 할당됩니다.

InitializeAcl 새 ACL 구조를 초기화합니다.
InitializeSecurityDescriptor 새 보안 설명자를 초기화합니다.
InitializeSid SID(보안 식별자)를 초기화합니다.
IsTokenRestricted 토큰에 제한된 SID(보안 식별자) 목록이 포함되어 있는지 여부를 나타냅니다.
IsValidAcl ACL(액세스 제어 목록)의 유효성을 검사합니다.
IsValidSecurityDescriptor 보안 설명자의 구성 요소가 유효한지 여부를 확인합니다.
IsValidSid 수정 번호가 알려진 범위 내에 있고 하위 인증 수가 최대값보다 작은지 확인하여 SID(보안 식별자)의 유효성을 검사합니다.
IsWellKnownSid SID를 잘 알려진 SID와 비교하고 일치하는 경우 TRUE를 반환합니다.
MakeAbsoluteSD 자체 상대 형식의 보안 설명자를 템플릿으로 사용하여 절대 형식의 보안 설명자를 만듭니다.
MakeSelfRelativeSD 절대 형식의 보안 설명자를 템플릿으로 사용하여 자체 상대 형식의 보안 설명자를 만듭니다.
MapGenericMask 액세스 마스크의 일반 액세스 권한을 특정 및 표준 액세스 권한에 매핑합니다. 함수는 GENERIC_MAPPING 구조에 제공된 매핑을 적용합니다.

ObjectCloseAuditAlarmW 프라이빗 개체에 대한 핸들이 삭제되면 보안 이벤트 로그에 감사 메시지를 생성합니다. (ObjectCloseAuditAlarmW)
ObjectDeleteAuditAlarmW ObjectDeleteAuditAlarmW(유니코드) 함수(securitybaseapi.h)는 개체가 삭제되면 감사 메시지를 생성합니다.
ObjectOpenAuditAlarmW 클라이언트 애플리케이션이 개체에 대한 액세스 권한을 얻거나 새 개체를 만들려고 할 때 감사 메시지를 생성합니다. (ObjectOpenAuditAlarmW)
ObjectPrivilegeAuditAlarmW 보안 이벤트 로그에 감사 메시지를 생성합니다. (ObjectPrivilegeAuditAlarmW)
PrivilegeCheck 액세스 토큰에서 지정된 권한 집합을 사용할 수 있는지 여부를 결정합니다.
PrivilegedServiceAuditAlarmW 보안 이벤트 로그에 감사 메시지를 생성합니다. (PrivilegedServiceAuditAlarmW)
QuerySecurityAccessMask 지정된 개체 보안 정보를 쿼리하는 데 필요한 액세스 권한을 나타내는 액세스 마스크를 만듭니다.
RevertToSelf 클라이언트 애플리케이션의 가장을 종료합니다.
SetAclInformation ACL(액세스 제어 목록)에 대한 정보를 설정합니다.
SetCachedSigningLevel 캐시된 서명 수준을 설정합니다.
SetFileSecurityW SetFileSecurityW(유니코드) 함수(securitybaseapi.h)는 파일 또는 디렉터리 개체의 보안을 설정합니다.

SetKernelObjectSecurity 커널 개체의 보안을 설정합니다.
SetPrivateObjectSecurity 프라이빗 개체의 보안 설명자를 수정합니다.
SetPrivateObjectSecurityEx 이 함수를 호출하는 리소스 관리자가 유지 관리하는 프라이빗 개체의 보안 설명자를 수정합니다.
SetSecurityAccessMask 지정된 개체 보안 정보를 설정하는 데 필요한 액세스 권한을 나타내는 액세스 마스크를 만듭니다.
SetSecurityDescriptorControl 보안 설명자의 제어 비트를 설정합니다. 함수는 ACE의 자동 상속과 관련된 컨트롤 비트만 설정할 수 있습니다.
SetSecurityDescriptorDacl DACL(임의 액세스 제어 목록)의 정보를 설정합니다. DACL이 보안 설명자에 이미 있는 경우 DACL이 대체됩니다.
SetSecurityDescriptorGroup 절대 형식 보안 설명자의 기본 그룹 정보를 설정하여 보안 설명자에 이미 있는 기본 그룹 정보를 바꿉니다.
SetSecurityDescriptorOwner 절대 형식 보안 설명자의 소유자 정보를 설정합니다. 보안 설명자에 이미 있는 소유자 정보를 대체합니다.
SetSecurityDescriptorRMControl SECURITY_DESCRIPTOR 구조에서 리소스 관리자 컨트롤 비트를 설정합니다.
SetSecurityDescriptorSacl SACL(시스템 액세스 제어 목록)의 정보를 설정합니다. 보안 설명자에 SACL이 이미 있는 경우 대체됩니다.
SetTokenInformation 지정된 액세스 토큰에 대한 다양한 유형의 정보를 설정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

sensapi.h 헤더

아티클2023. 08. 26.

이 헤더는 시스템 이벤트 알림 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 이벤트 알림 서비스

sensapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

IsDestinationReachableA IsDestinationReachable 함수는 지정된 대상에 연결할 수 있는지 여부를 결정하고 대상에 대한 QOC(연결 품질) 정보를 제공합니다. (ANSI)
IsDestinationReachableW IsDestinationReachable 함수는 지정된 대상에 연결할 수 있는지 여부를 결정하고 대상에 대한 QOC(연결 품질) 정보를 제공합니다. (유니코드)
IsNetworkAlive IsNetworkAlive 함수는 로컬 시스템이 네트워크에 연결되어 있는지 여부를 확인하고 네트워크 연결 유형(예: LAN, WAN 또는 둘 다)을 식별합니다.

구조체

QOCINFO QOCINFO 구조체는 IsDestinationReachable 함수에 의해 반환되며 호출자에게 연결 품질 정보를 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

sensevts.h 헤더

아티클2023. 08. 26.

이 헤더는 System Event Notification Service에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 이벤트 알림 서비스

sensevts.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ISensLogon ISensLogon 인터페이스는 SENS에서 발생한 로그온 이벤트를 처리합니다.
ISensLogon2 ISensLogon2 인터페이스는 SENS에서 발생한 로그온 이벤트를 처리합니다.
ISensNetwork ISensNetwork 인터페이스는 SENS(시스템 이벤트 알림 서비스)에서 발생한 네트워크 이벤트를 처리합니다.
ISensOnNow ISensOnNow 인터페이스는 SENS(시스템 이벤트 알림 서비스)에서 발생한 AC 및 배터리 전원 이벤트를 처리합니다.

구조체

SENS_QOCINFO SENS_QOCINFO 구조체는 ISensNetwork::ConnectionMade 메서드에서 사용됩니다. 이 구조체에는 SENS를 구독하는 애플리케이션의 싱크 개체에 대한 연결 품질 정보가 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

sensorsapi.h 헤더

> **Important**
>
> 대신 **UWP 센서 API** 를 사용합니다.
>
> COM 기반 센서 API는 더 이상 사용되지 않으며 새 애플리케이션에서 사용해서는 안 됩니다. 추가 기능 또는 향상된 기능은 계획되지 않으며 지원이 제한됩니다.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 위치 API
- 센서 API
- 센서 디바이스 드라이버 참조
- winsensors_com_ref

sensorsapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Interfaces

테이블 확장

ILocationPermissions 사용자가 위치 설정을 변경할 수 있도록 하는 시스템 설정의 상태를 제공합니다.
ISensor 센서를 나타냅니다.
ISensorCollection 컴퓨터에 연결된 모든 센서와 같은 센서 컬렉션을 나타냅니다.
ISensorDataReport 센서 데이터 보고서를 나타냅니다. 센서 데이터 보고서에는 센서에서 생성된 데이터 필드 값과 데이터 보고서가 만들어진 시기를 나타내는 타임스탬프를 포함합니다.
ISensorEvents 센서 이벤트를 수신하려는 경우 구현해야 하는 콜백 인터페이스입니다.

ISensorManager 사용 가능한 센서를 검색 및 검색하는 방법과 센서 관리자 이벤트를 요청하는 방법을 제공합니다.
ISensorManagerEvents 센서 관리자 이벤트를 수신하기 위한 콜백 인터페이스입니다.

열거 목록

테이블 확장

LOCATION_DESIRED_ACCURACY LOCATION_DESIRED_ACCURACY 열거형 형식은 SENSOR_PROPERTY_LOCATION_DESIRED_ACCURACY 속성에 대한 값을 정의합니다.
MagnetometerAccuracy 자력계의 정확도를 지정합니다.
SensorConnectionType 센서 디바이스 연결 유형을 정의합니다.
SensorState 센서에 대해 가능한 작동 상태를 정의합니다.

Last updated on 2025. 09. 21.

servprov.h 헤더

아티클2023. 08. 26.

이 헤더는 COM(구성 요소 개체 모델)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)

servprov.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IServiceProvider IServiceProvider 인터페이스는 GUID 식별 서비스를 찾는 일반 액세스 메커니즘을 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

sessdirpublictypes.h 헤더

아티클2023. 08. 26.

이 헤더는 원격 데스크톱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 데스크톱 서비스

sessdirpublictypes.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

CLIENT_DISPLAY RDC(원격 데스크톱 연결) 클라이언트 표시에 대한 정보를 포함합니다. (CLIENT_DISPLAY)
TSSD_ConnectionPoint 대상의 IP 주소를 정의합니다.

열거형

RDV_TASK_STATUS 작업의 상태 나타내기 위해 IRDVTaskPluginNotifySink::OnTaskStateChange 메서드와 함께 사용됩니다.
TARGET_CHANGE_TYPE 대상에서 발생한 변경 유형을 지정합니다.
TARGET_STATE 대상의 상태를 나타냅니다.
TARGET_TYPE 대상이 풀 또는 팜에 속하는지 여부를 나타냅니다.
TSSD_AddrV46Type IP 주소 유형을 나타냅니다.

TSSESSION_STATE 세션의 상태를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

setupapi.h 헤더

아티클2023. 03. 16.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 애플리케이션 설치 및 서비스
- 디바이스 및 드라이버 설치 참조

setupapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

InstallHinfSectionA InstallHinfSection은 .inf 파일의 섹션을 실행하는 데 사용할 수 있는 Setupapi.dll 내보낸 진입점 함수입니다. InstallHinfSection은 주의 섹션에 설명된 대로 Rundll32.exe 유틸리티를 호출하여 호출할 수 있습니다. (ANSI)
InstallHinfSectionW InstallHinfSection은 .inf 파일의 섹션을 실행하는 데 사용할 수 있는 Setupapi.dll 내보낸 진입점 함수입니다. InstallHinfSection은 주의 섹션에 설명된 대로 Rundll32.exe 유틸리티를 호출하여 호출할 수 있습니다. (유니코드)
SetupAddInstallSectionToDiskSpaceListA SetupAddInstallSectionToDiskSpaceList 함수는 INF 파일의 설치 섹션에서 CopyFile 및 DelFile 줄을 검색합니다. 그런 다음, 함수는 해당 섹션에 지정된 파일 작업을 디스크 공간 목록에 추가합니다. (ANSI)
SetupAddInstallSectionToDiskSpaceListW SetupAddInstallSectionToDiskSpaceList 함수는 INF 파일의 설치 섹션에서 CopyFile 및 DelFile 줄을 검색합니다. 그런 다음, 함수는 해당 섹션에 지정된 파일 작업을 디스크 공간 목록에 추가합니다. (유니코드)
SetupAddSectionToDiskSpaceListA SetupAddSectionToDiskSpaceList 함수는 INF 파일의 파일 복사 또는 파일 삭제 섹션에 나열된 모든 파일 삭제 또는 복사 작업을 디스크 공간 목록에 추가합니다. (ANSI)

SetupAddSectionToDiskSpaceListW SetupAddSectionToDiskSpaceList 함수는 INF 파일의 파일 복사 또는 파일 삭제 섹션에 나열된 모든 파일 삭제 또는 복사 작업을 디스크 공간 목록에 추가합니다. (유니코드)
SetupAddToDiskSpaceListA SetupAddToDiskSpaceList 함수는 단일 삭제 또는 복사 작업을 디스크 공간 목록에 추가합니다. INF 파일의 섹션에 있는 모든 파일 작업을 추가하려면 SetupAddSectionToDiskSpaceList 또는 SetupAddInstallSectionToDiskSpaceList를 사용합니다. (ANSI)
SetupAddToDiskSpaceListW SetupAddToDiskSpaceList 함수는 단일 삭제 또는 복사 작업을 디스크 공간 목록에 추가합니다. INF 파일의 섹션에 있는 모든 파일 작업을 추가하려면 SetupAddSectionToDiskSpaceList 또는 SetupAddInstallSectionToDiskSpaceList를 사용합니다. (유니코드)
SetupAddToSourceListA SetupAddToSourceList 함수는 현재 사용자 또는 시스템의 설치 원본 목록에 값을 추가합니다. 값이 이미 있는 경우 중복 항목이 만들어지지 않도록 먼저 제거됩니다. (ANSI)
SetupAddToSourceListW SetupAddToSourceList 함수는 현재 사용자 또는 시스템의 설치 원본 목록에 값을 추가합니다. 값이 이미 있는 경우 중복 항목이 만들어지지 않도록 먼저 제거됩니다. (유니코드)
SetupAdjustDiskSpaceListA SetupAdjustDiskSpaceList 함수는 지정된 드라이브에 필요한 공간의 양을 조정합니다. (ANSI)
SetupAdjustDiskSpaceListW SetupAdjustDiskSpaceList 함수는 지정된 드라이브에 필요한 공간의 양을 조정합니다. (유니코드)
SetupBackupErrorA SetupBackupError 함수는 사용자에게 백업 오류를 알리는 대화 상자를 생성합니다. (ANSI)
SetupBackupErrorW SetupBackupError 함수는 사용자에게 백업 오류를 알리는 대화 상자를 생성합니다. (유니코드)
SetupCancelTemporarySourceList SetupCancelTemporarySourceList 함수는 임시 목록 및 찾아보기 없음 동작을 취소하고 표준 목록 동작을 다시 설정합니다.

SetupCloseFileQueue SetupCloseFileQueue 함수는 설치 파일 큐를 제거합니다.
SetupCloseInfFile SetupCloseInfFile 함수는 SetupOpenInfFile 호출로 열린 INF 파일을 닫습니다. 이 함수는 SetupOpenAppendInfFile을 호출하여 추가된 모든 INF 파일을 닫습니다.
SetupCloseLog SetupCloseLog 함수는 로그 파일을 닫습니다.
SetupCommitFileQueueA SetupCommitFileQueue 함수는 설치 파일 큐에 대기 중인 파일 작업을 수행합니다. (ANSI)
SetupCommitFileQueueW SetupCommitFileQueue 함수는 설치 파일 큐에 대기 중인 파일 작업을 수행합니다. (유니코드)
SetupConfigureWmiFromInfSectionA SetupConfigureWmiFromInfSection 함수는 [DDInstall.WMI] 섹션에 전달될 때 INF 파일에 의해 노출되는 WMI 데이터의 보안을 구성합니다. (ANSI)
SetupConfigureWmiFromInfSectionW SetupConfigureWmiFromInfSection 함수는 [DDInstall.WMI] 섹션에 전달될 때 INF 파일에 의해 노출되는 WMI 데이터의 보안을 구성합니다. (유니코드)
SetupCopyErrorA SetupCopyError 함수는 사용자에게 복사 파일 오류를 알리는 대화 상자를 생성합니다. (ANSI)
SetupCopyErrorW SetupCopyError 함수는 사용자에게 복사 파일 오류를 알리는 대화 상자를 생성합니다. (유니코드)
SetupCopyOEMInfA SetupCopyOEMInf 함수는 지정된 .inf 파일을 %windir%/Inf 디렉터리에 복사합니다. (ANSI)
SetupCopyOEMInfW SetupCopyOEMInf 함수는 지정된 .inf 파일을 %windir%/Inf 디렉터리에 복사합니다. (유니코드)

SetupCreateDiskSpaceListA SetupCreateDiskSpaceList 함수는 디스크 공간 목록을 만듭니다. (ANSI)
SetupCreateDiskSpaceListW SetupCreateDiskSpaceList 함수는 디스크 공간 목록을 만듭니다. (유니코드)
SetupDecompressOrCopyFileA SetupDecompressOrCopyFile 함수는 파일을 복사하여 필요한 경우 압축을 풉니다. (ANSI)
SetupDecompressOrCopyFileW SetupDecompressOrCopyFile 함수는 파일을 복사하여 필요한 경우 압축을 풉니다. (유니코드)
SetupDefaultQueueCallbackA SetupDefaultQueueCallback 함수는 설치 API에 포함된 기본 큐 콜백 루틴입니다. 이 함수를 사용하여 SetupCommitFileQueue 함수에서 보낸 알림을 처리할 수 있습니다. (ANSI)
SetupDefaultQueueCallbackW SetupDefaultQueueCallback 함수는 설치 API에 포함된 기본 큐 콜백 루틴입니다. 이 함수를 사용하여 SetupCommitFileQueue 함수에서 보낸 알림을 처리할 수 있습니다. (유니코드)
SetupDeleteErrorA SetupDeleteError 함수는 삭제 오류를 사용자에게 알리는 대화 상자를 생성합니다. (ANSI)
SetupDeleteErrorW SetupDeleteError 함수는 삭제 오류를 사용자에게 알리는 대화 상자를 생성합니다. (유니코드)
SetupDestroyDiskSpaceList SetupDestroyDiskSpaceList 함수는 디스크 공간 목록을 삭제하고 할당된 리소스를 해제합니다.
SetupDiAskForOEMDisk SetupDiAskForOEMDisk 함수는 사용자에게 OEM 설치 디스크의 경로를 요청하는 대화 상자를 표시합니다.
SetupDiBuildClassInfoList SetupDiBuildClassInfoList 함수는 로컬 컴퓨터에 설치된 클래스를 식별하는 설치 클래스 GUID 목록을 반환합니다.

SetupDiBuildClassInfoListExA SetupDiBuildClassInfoListEx 함수는 로컬 시스템 또는 원격 시스템에 설치된 모든 클래스를 포함하는 설치 클래스 GUID 목록을 반환합니다. (ANSI)
SetupDiBuildClassInfoListExW SetupDiBuildClassInfoListEx 함수는 로컬 시스템 또는 원격 시스템에 설치된 모든 클래스를 포함하는 설치 클래스 GUID 목록을 반환합니다. (유니코드)
SetupDiBuildDriverInfoList SetupDiBuildDriverInfoList 함수는 특정 디바이스 또는 디바이스 정보 집합에 대한 전역 클래스 드라이버 목록과 연결된 드라이버 목록을 빌드합니다.
SetupDiCallClassInstaller SetupDiCallClassInstaller 함수는 지정된 설치 요청(DIF 코드)을 사용하여 적절한 클래스 설치 관리자 및 등록된 모든 공동 설치 관리자를 호출합니다.
SetupDiCancelDriverInfoSearch SetupDiCancelDriverInfoSearch 함수는 다른 스레드에서 현재 진행 중인 드라이버 목록 검색을 취소합니다.
SetupDiChangeState SetupDiChangeState 함수는 DIF_PROPERTYCHANGE 설치 요청에 대한 기본 처리기입니다.
SetupDiClassGuidsFromNameA SetupDiClassGuidsFromName 함수는 지정된 클래스 이름과 연결된 GUID를 검색합니다. 이 목록은 현재 시스템에 설치된 클래스를 기반으로 빌드됩니다. (ANSI)
SetupDiClassGuidsFromNameExA SetupDiClassGuidsFromNameEx 함수는 지정된 클래스 이름과 연결된 GUID를 검색합니다. 이 결과 목록에는 로컬 또는 원격 컴퓨터에 현재 설치된 클래스가 포함됩니다. (ANSI)
SetupDiClassGuidsFromNameExW SetupDiClassGuidsFromNameEx 함수는 지정된 클래스 이름과 연결된 GUID를 검색합니다. 이 결과 목록에는 로컬 또는 원격 컴퓨터에 현재 설치된 클래스가 포함됩니다. (유니코드)
SetupDiClassGuidsFromNameW SetupDiClassGuidsFromName 함수는 지정된 클래스 이름과 연결된 GUID를 검색합니다. 이 목록은 현재 시스템에 설치된 클래스를 기반으로 빌드됩니다. (유니코드)

SetupDiClassNameFromGuidA SetupDiClassNameFromGuid 함수는 클래스 GUID와 연결된 클래스 이름을 검색합니다. (ANSI)
SetupDiClassNameFromGuidExA SetupDiClassNameFromGuidEx 함수는 클래스 GUID와 연결된 클래스 이름을 검색합니다. 클래스는 로컬 또는 원격 컴퓨터에 설치할 수 있습니다. (ANSI)
SetupDiClassNameFromGuidExW SetupDiClassNameFromGuidEx 함수는 클래스 GUID와 연결된 클래스 이름을 검색합니다. 클래스는 로컬 또는 원격 컴퓨터에 설치할 수 있습니다. (유니코드)
SetupDiClassNameFromGuidW SetupDiClassNameFromGuid 함수는 클래스 GUID와 연결된 클래스 이름을 검색합니다. (유니코드)
SetupDiCreateDeviceInfoA SetupDiCreateDeviceInfo 함수는 새 디바이스 정보 요소를 만들고 지정된 디바이스 정보 집합에 새 멤버로 추가합니다. (ANSI)
SetupDiCreateDeviceInfoList SetupDiCreateDeviceInfoList 함수는 빈 디바이스 정보 집합을 만들고 필요에 따라 집합을 디바이스 설정 클래스 및 최상위 창에 연결합니다.
SetupDiCreateDeviceInfoListExA SetupDiCreateDeviceInfoList 함수는 원격 또는 로컬 컴퓨터에 설정된 빈 디바이스 정보를 만들고 필요에 따라 집합을 디바이스 설정 클래스와 연결합니다. (ANSI)
SetupDiCreateDeviceInfoListExW SetupDiCreateDeviceInfoList 함수는 원격 또는 로컬 컴퓨터에 설정된 빈 디바이스 정보를 만들고 필요에 따라 집합을 디바이스 설정 클래스와 연결합니다. (유니코드)
SetupDiCreateDeviceInfoW SetupDiCreateDeviceInfo 함수는 새 디바이스 정보 요소를 만들고 지정된 디바이스 정보 집합에 새 멤버로 추가합니다. (유니코드)
SetupDiCreateDeviceInterfaceA SetupDiCreateDeviceInterface 함수는 로컬 시스템 또는 원격 시스템에 디바이스 인터페이스를 등록합니다. (ANSI)

SetupDiCreateDeviceInterfaceRegKeyA SetupDiCreateDeviceInterfaceRegKey 함수는 디바이스 인터페이스에 대한 정보를 저장하기 위한 레지스트리 키를 만들고 키에 대한 핸들을 반환합니다. (ANSI)
SetupDiCreateDeviceInterfaceRegKeyW SetupDiCreateDeviceInterfaceRegKey 함수는 디바이스 인터페이스에 대한 정보를 저장하기 위한 레지스트리 키를 만들고 키에 대한 핸들을 반환합니다. (유니코드)
SetupDiCreateDeviceInterfaceW SetupDiCreateDeviceInterface 함수는 로컬 시스템 또는 원격 시스템에 디바이스 인터페이스를 등록합니다. (유니코드)
SetupDiCreateDevRegKeyA SetupDiCreateDevRegKey 함수는 디바이스별 구성 정보에 대한 레지스트리 키를 만들고 키에 대한 핸들을 반환합니다. (ANSI)
SetupDiCreateDevRegKeyW SetupDiCreateDevRegKey 함수는 디바이스별 구성 정보에 대한 레지스트리 키를 만들고 키에 대한 핸들을 반환합니다. (유니코드)
SetupDiDeleteDeviceInfo SetupDiDeleteDeviceInfo 함수는 디바이스 정보 집합에서 디바이스 정보 요소를 삭제합니다. 이 함수는 실제 디바이스를 삭제하지 않습니다.
SetupDiDeleteDeviceInterfaceData SetupDiDeleteDeviceInterfaceData 함수는 디바이스 정보 집합에서 디바이스 인터페이스를 삭제합니다.
SetupDiDeleteDeviceInterfaceRegKey SetupDiDeleteDeviceInterfaceRegKey 함수는 애플리케이션 및 드라이버가 인터페이스 관련 정보를 저장하는 데 사용하는 레지스트리 하위 키를 삭제합니다.
SetupDiDeleteDevRegKey SetupDiDeleteDevRegKey 함수는 디바이스 정보 요소와 연결된 지정된 사용자 액세스 가능 레지스트리 키를 삭제합니다.
SetupDiDestroyClassImageList SetupDiDestroyClassImageList 함수는 SetupDiGetClassImageList 또는 SetupDiGetClassImageListEx에 대한 호출로 작성된 클래스 이미지 목록을 삭제합니다.

SetupDiDestroyDeviceInfoList SetupDiDestroyDeviceInfoList 함수는 디바이스 정보 집합을 삭제하고 연결된 모든 메모리를 해제합니다.
SetupDiDestroyDriverInfoList SetupDiDestroyDriverInfoList 함수는 드라이버 목록을 삭제합니다.
SetupDiDrawMiniIcon SetupDiDrawMiniIcon 함수는 요청된 위치에 지정된 미니 아이콘을 그립니다.
SetupDiEnumDeviceInfo SetupDiEnumDeviceInfo 함수는 디바이스 정보 집합의 디바이스 정보 요소를 지정하는 SP_DEVINFO_DATA 구조를 반환합니다.
SetupDiEnumDeviceInterfaces SetupDiEnumDeviceInterfaces 함수는 디바이스 정보 집합에 포함된 디바이스 인터페이스를 열거합니다.
SetupDiEnumDriverInfoA SetupDiEnumDriverInfo 함수는 드라이버 목록의 멤버를 열거합니다. (ANSI)
SetupDiEnumDriverInfoW SetupDiEnumDriverInfo 함수는 드라이버 목록의 멤버를 열거합니다. (유니코드)
SetupDiGetActualModelsSectionA SetupDiGetActualModelsSection 함수는 디바이스 INF 파일에서 디바이스를 설치할 때 사용할 적절한 데코레이팅된 INF 모델 섹션을 검색합니다. (ANSI)
SetupDiGetActualModelsSectionW SetupDiGetActualModelsSection 함수는 디바이스 INF 파일에서 디바이스를 설치할 때 사용할 적절한 데코레이팅된 INF 모델 섹션을 검색합니다. (유니코드)
SetupDiGetActualSectionToInstallA SetupDiGetActualSectionToInstall 함수는 로컬 컴퓨터의 디바이스 INF 파일에서 디바이스를 설치할 때 사용할 적절한 INF DDInstall 섹션을 검색합니다. (ANSI)

SetupDiGetActualSectionToInstallExA SetupDiGetActualSectionToInstallEx 함수는 지정된 운영 체제 및 프로세서 아키텍처에 대한 디바이스를 설치하는 INF DDInstall 섹션의 이름을 검색합니다. (ANSI)
SetupDiGetActualSectionToInstallExW SetupDiGetActualSectionToInstallEx 함수는 지정된 운영 체제 및 프로세서 아키텍처에 대한 디바이스를 설치하는 INF DDInstall 섹션의 이름을 검색합니다. (유니코드)
SetupDiGetActualSectionToInstallW SetupDiGetActualSectionToInstall 함수는 로컬 컴퓨터의 디바이스 INF 파일에서 디바이스를 설치할 때 사용할 적절한 INF DDInstall 섹션을 검색합니다. (유니코드)
SetupDiGetClassBitmapIndex SetupDiGetClassBitmapIndex 함수는 지정된 클래스에 대해 제공된 미니 아이콘의 인덱스를 검색합니다.
SetupDiGetClassDescriptionA SetupDiGetClassDescription 함수는 지정된 설치 클래스 GUID와 연결된 클래스 설명을 검색합니다. (ANSI)
SetupDiGetClassDescriptionExA SetupDiGetClassDescriptionEx 함수는 로컬 또는 원격 컴퓨터에 설치된 설치 클래스에 대한 설명을 검색합니다. (ANSI)
SetupDiGetClassDescriptionExW SetupDiGetClassDescriptionEx 함수는 로컬 또는 원격 컴퓨터에 설치된 설치 클래스에 대한 설명을 검색합니다. (유니코드)
SetupDiGetClassDescriptionW SetupDiGetClassDescription 함수는 지정된 설치 클래스 GUID와 연결된 클래스 설명을 검색합니다. (유니코드)
SetupDiGetClassDevPropertySheetsA SetupDiGetClassDevPropertySheets 함수는 디바이스 정보 요소의 속성 시트 또는 디바이스 정보 집합의 디바이스 설정 클래스에 대한 핸들을 검색합니다. (ANSI)
SetupDiGetClassDevPropertySheetsW SetupDiGetClassDevPropertySheets 함수는 디바이스 정보 요소의 속성 시트 또는 디바이스 정보 집합의 디바이스 설정 클래스에 대한 핸들을 검색합니다. (유니코드)

SetupDiGetClassDevsA SetupDiGetClassDevs 함수는 로컬 컴퓨터에 대해 요청된 디바이스 정보 요소가 포함된 디바이스 정보 집합에 대한 핸들을 반환합니다. (ANSI)
SetupDiGetClassDevsExA SetupDiGetClassDevsEx 함수는 로컬 또는 원격 컴퓨터에 대해 요청된 디바이스 정보 요소가 포함된 디바이스 정보 집합에 대한 핸들을 반환합니다. (ANSI)
SetupDiGetClassDevsExW SetupDiGetClassDevsEx 함수는 로컬 또는 원격 컴퓨터에 대해 요청된 디바이스 정보 요소가 포함된 디바이스 정보 집합에 대한 핸들을 반환합니다. (유니코드)
SetupDiGetClassDevsW SetupDiGetClassDevs 함수는 로컬 컴퓨터에 대해 요청된 디바이스 정보 요소가 포함된 디바이스 정보 집합에 대한 핸들을 반환합니다. (유니코드)
SetupDiGetClassImageIndex SetupDiGetClassImageIndex 함수는 지정된 클래스의 클래스 이미지 목록 내에서 인덱스를 검색합니다.
SetupDiGetClassImageList SetupDiGetClassImageList 함수는 설치된 모든 클래스에 대한 비트맵이 포함된 이미지 목록을 빌드하고 데이터 구조에서 목록을 반환합니다.
SetupDiGetClassImageListExA SetupDiGetClassImageListEx 함수는 로컬 또는 원격 시스템에 설치된 모든 클래스에 대한 비트맵의 이미지 목록을 빌드합니다. (ANSI)
SetupDiGetClassImageListExW SetupDiGetClassImageListEx 함수는 로컬 또는 원격 시스템에 설치된 모든 클래스에 대한 비트맵의 이미지 목록을 빌드합니다. (유니코드)
SetupDiGetClassInstallParamsA SetupDiGetClassInstallParams 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대한 클래스 설치 매개 변수를 검색합니다. (ANSI)
SetupDiGetClassInstallParamsW SetupDiGetClassInstallParams 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대한 클래스 설치 매개 변수를 검색합니다. (유니코드)

SetupDiGetClassPropertyExW SetupDiGetClassPropertyEx 함수는 로컬 또는 원격 컴퓨터에서 디바이스 설정 클래스 또는 디바이스 인터페이스 클래스에 대한 클래스 속성을 검색합니다.
SetupDiGetClassPropertyKeys SetupDiGetClassPropertyKeys 함수는 디바이스 설정 클래스 또는 디바이스 인터페이스 클래스에 대해 설정된 디바이스 속성을 나타내는 디바이스 속성 키의 배열을 검색합니다.
SetupDiGetClassPropertyKeysExW SetupDiGetClassPropertyKeysEx 함수는 로컬 또는 원격 컴퓨터에서 디바이스 설정 클래스 또는 디바이스 인터페이스 클래스에 대해 설정된 디바이스 속성을 나타내는 디바이스 속성 키의 배열을 검색합니다.
SetupDiGetClassPropertyW SetupDiGetClassProperty 함수는 디바이스 설정 클래스 또는 디바이스 인터페이스 클래스에 대해 설정된 디바이스 속성을 검색합니다.
SetupDiGetClassRegistryPropertyA SetupDiGetClassRegistryProperty 함수는 레지스트리에서 지정된 디바이스 설정 클래스에 대한 속성을 검색합니다. (ANSI)
SetupDiGetClassRegistryPropertyW SetupDiGetClassRegistryProperty 함수는 레지스트리에서 지정된 디바이스 설정 클래스에 대한 속성을 검색합니다. (유니코드)
SetupDiGetCustomDevicePropertyA SetupDiGetCustomDeviceProperty 함수는 레지스트리에서 지정된 사용자 지정 디바이스 속성을 검색합니다. (ANSI)
SetupDiGetCustomDevicePropertyW SetupDiGetCustomDeviceProperty 함수는 레지스트리에서 지정된 사용자 지정 디바이스 속성을 검색합니다. (유니코드)
SetupDiGetDeviceInfoListClass SetDiGetDeviceInfoListClass 함수는 집합에 연결된 클래스가 있는 경우 디바이스 정보 집합과 연결된 디바이스 설정 클래스에 대한 GUID를 검색합니다.

SetupDiGetDeviceInfoListDetailA SetupDiGetDeviceInfoListDetail 함수는 클래스 GUID, 원격 컴퓨터 핸들 및 원격 컴퓨터 이름을 포함하여 디바이스 정보 집합과 관련된 정보를 검색합니다. (ANSI)
SetupDiGetDeviceInfoListDetailW SetupDiGetDeviceInfoListDetail 함수는 클래스 GUID, 원격 컴퓨터 핸들 및 원격 컴퓨터 이름을 포함하여 디바이스 정보 집합과 관련된 정보를 검색합니다. (유니코드)
SetupDiGetDeviceInstallParamsA SetupDiGetDeviceInstallParams 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대한 디바이스 설치 매개 변수를 검색합니다. (ANSI)
SetupDiGetDeviceInstallParamsW SetupDiGetDeviceInstallParams 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대한 디바이스 설치 매개 변수를 검색합니다. (유니코드)
SetupDiGetDeviceInstanceIdA SetupDiGetDeviceInstanceId 함수는 디바이스 정보 요소와 연결된 디바이스 인스턴스 ID를 검색합니다. (ANSI)
SetupDiGetDeviceInstanceIdW SetupDiGetDeviceInstanceId 함수는 디바이스 정보 요소와 연결된 디바이스 인스턴스 ID를 검색합니다. (유니코드)
SetupDiGetDeviceInterfaceAlias SetupDiGetDeviceInterfaceAlias 함수는 지정된 디바이스 인터페이스의 별칭을 반환합니다.
SetupDiGetDeviceInterfaceDetailA SetupDiGetDeviceInterfaceDetail 함수는 디바이스 인터페이스에 대한 세부 정보를 반환합니다. (ANSI)
SetupDiGetDeviceInterfaceDetailW SetupDiGetDeviceInterfaceDetail 함수는 디바이스 인터페이스에 대한 세부 정보를 반환합니다. (유니코드)
SetupDiGetDeviceInterfacePropertyKeys SetupDiGetDeviceInterfacePropertyKeys 함수는 디바이스 인터페이스에 대해 설정된 디바이스 속성을 나타내는 디바이스 속성 키 배열을 검색합니다.

SetupDiGetDeviceInterfacePropertyW SetupDiGetDeviceInterfaceProperty 함수는 디바이스 인터페이스에 대해 설정된 디바이스 속성을 검색합니다.
SetupDiGetDevicePropertyKeys SetupDiGetDevicePropertyKeys 함수는 디바이스 인스턴스에 대해 설정된 디바이스 속성을 나타내는 디바이스 속성 키의 배열을 검색합니다.
SetupDiGetDevicePropertyW SetupDiGetDeviceProperty 함수는 디바이스 인스턴스 속성을 검색합니다.
SetupDiGetDeviceRegistryPropertyA SetupDiGetDeviceRegistryProperty 함수는 지정된 플러그 앤 플레이 디바이스 속성을 검색합니다. (ANSI)
SetupDiGetDeviceRegistryPropertyW SetupDiGetDeviceRegistryProperty 함수는 지정된 플러그 앤 플레이 디바이스 속성을 검색합니다. (유니코드)
SetupDiGetDriverInfoDetailA SetupDiGetDriverInfoDetail 함수는 디바이스 정보 집합의 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대한 드라이버 정보 세부 정보를 검색합니다. (ANSI)
SetupDiGetDriverInfoDetailW SetupDiGetDriverInfoDetail 함수는 디바이스 정보 집합의 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대한 드라이버 정보 세부 정보를 검색합니다. (유니코드)
SetupDiGetDriverInstallParamsA SetupDiGetDriverInstallParams 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대한 드라이버 설치 매개 변수를 검색합니다. (ANSI)
SetupDiGetDriverInstallParamsW SetupDiGetDriverInstallParams 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대한 드라이버 설치 매개 변수를 검색합니다. (유니코드)
SetupDiGetHwProfileFriendlyNameA SetupDiGetHwProfileFriendlyName 함수는 하드웨어 프로필 ID와 연결된 이름을 검색합니다. (ANSI)

SetupDiGetHwProfileFriendlyNameExA SetupDiGetHwProfileFriendlyNameEx 함수는 로컬 또는 원격 컴퓨터의 하드웨어 프로필 ID와 연결된 이름을 검색합니다. (ANSI)
SetupDiGetHwProfileFriendlyNameExW SetupDiGetHwProfileFriendlyNameEx 함수는 로컬 또는 원격 컴퓨터의 하드웨어 프로필 ID와 연결된 이름을 검색합니다. (유니코드)
SetupDiGetHwProfileFriendlyNameW SetupDiGetHwProfileFriendlyName 함수는 하드웨어 프로필 ID와 연결된 이름을 검색합니다. (유니코드)
SetupDiGetHwProfileList SetupDiGetHwProfileList 함수는 현재 정의된 모든 하드웨어 프로필 ID 목록을 검색합니다.
SetupDiGetHwProfileListExA SetupDiGetHwProfileListEx 함수는 로컬 또는 원격 컴퓨터에서 현재 정의된 모든 하드웨어 프로필 ID 목록을 검색합니다. (ANSI)
SetupDiGetHwProfileListExW SetupDiGetHwProfileListEx 함수는 로컬 또는 원격 컴퓨터에서 현재 정의된 모든 하드웨어 프로필 ID 목록을 검색합니다. (유니코드)
SetupDiGetINFClassA SetupDiGetINFClass 함수는 지정된 디바이스 INF 파일의 클래스를 반환합니다. (ANSI)
SetupDiGetINFClassW SetupDiGetINFClass 함수는 지정된 디바이스 INF 파일의 클래스를 반환합니다. (유니코드)
SetupDiGetSelectedDevice SetupDiGetSelectedDevice 함수는 디바이스 정보 집합에서 선택한 디바이스 정보 요소를 검색합니다.
SetupDiGetSelectedDriverA SetupDiGetSelectedDriver 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대해 선택한 드라이버를 검색합니다. (ANSI)

SetupDiGetSelectedDriverW SetupDiGetSelectedDriver 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대해 선택한 드라이버를 검색합니다. (유니코드)
SetupDiInstallClassA SetupDiInstallClass 함수는 지정된 INF 파일의 ClassInstall32 섹션을 설치합니다. (ANSI)
SetupDiInstallClassExA SetupDiInstallClassEx 함수는 클래스 설치 관리자 또는 인터페이스 클래스를 설치합니다. (ANSI)
SetupDiInstallClassExW SetupDiInstallClassEx 함수는 클래스 설치 관리자 또는 인터페이스 클래스를 설치합니다. (유니코드)
SetupDiInstallClassW SetupDiInstallClass 함수는 지정된 INF 파일의 ClassInstall32 섹션을 설치합니다. (유니코드)
SetupDiInstallDevice SetupDiInstallDevice 함수는 DIF_INSTALLDEVICE 설치 요청에 대한 기본 처리기입니다.
SetupDiInstallDeviceInterfaces SetupDiInstallDeviceInterfaces 함수는 DIF_INSTALLINTERFACES 설치 요청에 대한 기본 처리기입니다.
SetupDiInstallDriverFiles SetupDiInstallDriverFiles 함수는 DIF_INSTALLDEVICEFILES 설치 요청에 대한 기본 처리기입니다.
SetupDiLoadClassIcon SetupDiLoadClassIcon 함수는 지정된 클래스의 큰 아이콘과 미니 아이콘을 모두 로드합니다.
SetupDiLoadDeviceIcon SetupDiLoadDeviceIcon 함수는 지정된 디바이스의 아이콘을 검색합니다.
SetupDiOpenClassRegKey SetupDiOpenClassRegKey 함수는 설치 클래스 레지스트리 키 또는 특정 클래스의 하위 키를 엽니다.

SetupDiOpenClassRegKeyExA SetupDiOpenClassRegKeyEx 함수는 디바이스 설정 클래스 레지스트리 키, 디바이스 인터페이스 클래스 레지스트리 키 또는 특정 클래스의 하위 키를 엽니다. 이 함수는 로컬 컴퓨터 또는 원격 컴퓨터에서 지정된 키를 엽니다. (ANSI)
SetupDiOpenClassRegKeyExW SetupDiOpenClassRegKeyEx 함수는 디바이스 설정 클래스 레지스트리 키, 디바이스 인터페이스 클래스 레지스트리 키 또는 특정 클래스의 하위 키를 엽니다. 이 함수는 로컬 컴퓨터 또는 원격 컴퓨터에서 지정된 키를 엽니다. (유니코드)
SetupDiOpenDeviceInfoA SetupDiOpenDeviceInfo 함수는 디바이스 정보 집합에 아직 없는 경우 디바이스 인스턴스에 대한 디바이스 정보 요소를 디바이스 정보 집합에 추가하고 디바이스 정보 집합에서 디바이스 인스턴스의 디바이스 정보 요소를 식별하는 정보를 검색합니다. (ANSI)
SetupDiOpenDeviceInfoW SetupDiOpenDeviceInfo 함수는 디바이스 정보 집합에 아직 없는 경우 디바이스 인스턴스에 대한 디바이스 정보 요소를 디바이스 정보 집합에 추가하고 디바이스 정보 집합에서 디바이스 인스턴스의 디바이스 정보 요소를 식별하는 정보를 검색합니다. (유니코드)
SetupDiOpenDeviceInterfaceA SetupDiOpenDeviceInterface 함수는 디바이스 인터페이스에 대한 정보를 검색하고 로컬 시스템 또는 원격 시스템에 대해 설정된 지정된 디바이스 정보에 인터페이스를 추가합니다. (ANSI)
SetupDiOpenDeviceInterfaceRegKey SetupDiOpenDeviceInterfaceRegKey 함수는 애플리케이션 및 드라이버가 디바이스 인터페이스와 관련된 정보를 저장하는 데 사용하는 레지스트리 하위 키를 엽니다.
SetupDiOpenDeviceInterfaceW SetupDiOpenDeviceInterface 함수는 디바이스 인터페이스에 대한 정보를 검색하고 로컬 시스템 또는 원격 시스템에 대해 설정된 지정된 디바이스 정보에 인터페이스를 추가합니다. (유니코드)
SetupDiOpenDevRegKey SetupDiOpenDevRegKey 함수는 디바이스별 구성 정보에 대한 레지스트리 키를 엽니다.
SetupDiRegisterCoDeviceInstallers SetupDiRegisterCoDeviceInstallers 함수는 DIF_REGISTER_COINSTALLERS 기본 처리기입니다.

SetupDiRegisterDeviceInfo SetupDiRegisterDeviceInfo 함수는 DIF_REGISTERDEVICE 요청에 대한 기본 처리기입니다.
SetupDiRemoveDevice SetupDiRemoveDevice 함수는 DIF_REMOVE 설치 요청에 대한 기본 처리기입니다.
SetupDiRemoveDeviceInterface SetupDiRemoveDeviceInterface 함수는 시스템에서 등록된 디바이스 인터페이스를 제거합니다.
SetupDiRestartDevices SetupDiRestartDevices 함수는 지정된 디바이스를 다시 시작하거나 필요한 경우 동일한 함수에서 작동하는 모든 디바이스를 다시 시작하고 지정된 디바이스를 작동하는 드라이버를 필터링합니다.
SetupDiSelectBestCompatDrv SetupDiSelectBestCompatDrv 함수는 DIF_SELECTBESTCOMPATDRV 설치 요청에 대한 기본 처리기입니다.
SetupDiSelectDevice SetupDiSelectDevice 함수는 DIF_SELECTDEVICE 요청에 대한 기본 처리기입니다.
SetupDiSelectOEMDrv SetupDiSelectOEMDrv 함수는 디바이스 정보 집합에 대한 드라이버 또는 사용자가 제공한 OEM 경로를 사용하는 특정 디바이스 정보 요소를 선택합니다.
SetupDiSetClassInstallParamsA SetupDiSetClassInstallParams 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대한 클래스 설치 매개 변수를 설정하거나 지웁니다. (ANSI)
SetupDiSetClassInstallParamsW SetupDiSetClassInstallParams 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대한 클래스 설치 매개 변수를 설정하거나 지웁니다. (유니코드)
SetupDiSetClassPropertyExW SetupDiSetClassPropertyEx 함수는 로컬 또는 원격 컴퓨터에서 디바이스 설정 클래스 또는 디바이스 인터페이스 클래스에 대한 디바이스 속성을 설정합니다.

SetupDiSetClassPropertyW SetupDiSetClassProperty 함수는 디바이스 설정 클래스 또는 디바이스 인터페이스 클래스에 대한 클래스 속성을 설정합니다.
SetupDiSetClassRegistryPropertyA SetupDiSetClassRegistryProperty 함수는 레지스트리에서 지정된 디바이스 클래스 속성을 설정합니다. (ANSI)
SetupDiSetClassRegistryPropertyW SetupDiSetClassRegistryProperty 함수는 레지스트리에서 지정된 디바이스 클래스 속성을 설정합니다. (유니코드)
SetupDiSetDeviceInstallParamsA SetupDiSetDeviceInstallParams 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대한 디바이스 설치 매개 변수를 설정합니다. (ANSI)
SetupDiSetDeviceInstallParamsW SetupDiSetDeviceInstallParams 함수는 디바이스 정보 집합 또는 특정 디바이스 정보 요소에 대한 디바이스 설치 매개 변수를 설정합니다. (유니코드)
SetupDiSetDeviceInterfaceDefault SetupDiSetDeviceInterfaceDefault 함수는 디바이스 인터페이스 클래스의 기본 인터페이스로 디바이스 인터페이스를 설정합니다.
SetupDiSetDeviceInterfacePropertyW SetupDiSetDeviceInterfaceProperty 함수는 디바이스 인터페이스의 디바이스 속성을 설정합니다.
SetupDiSetDevicePropertyW SetupDiSetDeviceProperty 함수는 디바이스 인스턴스 속성을 설정합니다.
SetupDiSetDeviceRegistryPropertyA SetupDiSetDeviceRegistryProperty 함수는 디바이스에 대한 플러그 앤 플레이 디바이스 속성을 설정합니다. (ANSI)
SetupDiSetDeviceRegistryPropertyW SetupDiSetDeviceRegistryProperty 함수는 디바이스에 대한 플러그 앤 플레이 디바이스 속성을 설정합니다. (유니코드)

SetupDiSetDriverInstallParamsA SetupDiSetDriverInstallParams 함수는 드라이버 정보 요소에 대한 드라이버 설치 매개 변수를 설정합니다. (ANSI)
SetupDiSetDriverInstallParamsW SetupDiSetDriverInstallParams 함수는 드라이버 정보 요소에 대한 드라이버 설치 매개 변수를 설정합니다. (유니코드)
SetupDiSetSelectedDevice SetupDiSetSelectedDevice 함수는 디바이스 정보 집합의 선택된 멤버로 디바이스 정보 요소를 설정합니다. 이 함수는 일반적으로 설치 마법사에서 사용됩니다.
SetupDiSetSelectedDriverA SetupDiSetSelectedDriver 함수는 디바이스 정보 요소에 대해 선택한 드라이버 또는 디바이스 정보 집합에 대해 선택한 클래스 드라이버를 설정하거나 다시 설정합니다. (ANSI)
SetupDiSetSelectedDriverW SetupDiSetSelectedDriver 함수는 디바이스 정보 요소에 대해 선택한 드라이버 또는 디바이스 정보 집합에 대해 선택한 클래스 드라이버를 설정하거나 다시 설정합니다. (유니코드)
SetupDiUnremoveDevice SetupDiUnremoveDevice 함수는 DIF_UNREMOVE 설치 요청에 대한 기본 처리기입니다.
SetupDuplicateDiskSpaceListA SetupDuplicateDiskSpaceList 함수는 디스크 공간 목록을 새 독립 디스크 공간 목록으로 복제합니다. (ANSI)
SetupDuplicateDiskSpaceListW SetupDuplicateDiskSpaceList 함수는 디스크 공간 목록을 새 독립 디스크 공간 목록으로 복제합니다. (유니코드)
SetupEnumInfSectionsA SetupEnumInfSections 함수는 INF 파일에서 섹션 이름을 검색합니다. (ANSI)
SetupEnumInfSectionsW SetupEnumInfSections 함수는 INF 파일에서 섹션 이름을 검색합니다. (유니코드)

SetupFindFirstLineA SetupFindFirstLine 함수는 INF 파일의 지정된 섹션에서 줄을 찾습니다. Key 매개 변수가 NULL이면 SetupFindFirstLine은 섹션의 첫 번째 줄을 반환합니다. (ANSI)
SetupFindFirstLineW SetupFindFirstLine 함수는 INF 파일의 지정된 섹션에서 줄을 찾습니다. Key 매개 변수가 NULL이면 SetupFindFirstLine은 섹션의 첫 번째 줄을 반환합니다. (유니코드)
SetupFindNextLine SetupFindNextLine은 ContextIn.Line을 기준으로 INF 파일 섹션에서 다음 줄의 위치를 반환합니다.
SetupFindNextMatchLineA SetupFindNextMatchLine 함수는 지정된 키와 일치하는 ContextIn.Line을 기준으로 INF 파일에서 다음 줄의 위치를 반환합니다. (ANSI)
SetupFindNextMatchLineW SetupFindNextMatchLine 함수는 지정된 키와 일치하는 ContextIn.Line을 기준으로 INF 파일에서 다음 줄의 위치를 반환합니다. (유니코드)
SetupFreeSourceListA SetupFreeSourceList 함수는 원본 목록에 할당된 시스템 리소스를 해제합니다. (ANSI)
SetupFreeSourceListW SetupFreeSourceList 함수는 원본 목록에 할당된 시스템 리소스를 해제합니다. (유니코드)
SetupGetBinaryField SetupGetBinaryField 함수는 지정된 필드에서 줄 끝까지 INF 파일 섹션의 한 줄에서 이진 데이터를 검색합니다.
SetupGetFieldCount SetupGetFieldCount 함수는 INF 파일에서 지정된 줄의 필드 수를 검색합니다.
SetupGetFileCompressionInfoA SetupGetFileCompressionInfo 함수는 물리적 파일을 검사하여 압축되었는지 확인하고 압축되지 않은 대상 파일의 전체 경로, 크기 및 크기를 가져옵니다. (ANSI)

SetupGetFileCompressionInfoExA SetupGetFileCompressionInfoEx 함수는 잠재적으로 압축된 파일을 검사하고 압축 형식, 파일의 전체 경로(파일 이름 포함), 압축된 크기 및 압축되지 않은 대상 파일의 크기를 가져옵니다. (ANSI)
SetupGetFileCompressionInfoExW SetupGetFileCompressionInfoEx 함수는 잠재적으로 압축된 파일을 검사하고 압축 형식, 파일의 전체 경로(파일 이름 포함), 압축된 크기 및 압축되지 않은 대상 파일의 크기를 가져옵니다. (유니코드)
SetupGetFileCompressionInfoW SetupGetFileCompressionInfo 함수는 물리적 파일을 검사하여 압축되었는지 확인하고 압축되지 않은 대상 파일의 전체 경로, 크기 및 크기를 가져옵니다. (유니코드)
SetupGetFileQueueCount SetupGetFileQueueCount 함수는 설치 파일 큐에서 개수를 가져옵니다.
SetupGetFileQueueFlags SetupGetFileQueueFlags 함수는 설치 파일 큐에서 플래그를 가져옵니다.
SetupGetInfDriverStoreLocationA SetupGetInfDriverStoreLocation 함수는 드라이버 저장소에서 시스템 INF 파일 디렉터리의 지정된 INF 파일 또는 드라이버 저장소의 지정된 INF 파일에 해당하는 INF 파일의 정규화된 파일 이름(디렉터리 경로 및 파일 이름)을 검색합니다. (ANSI)
SetupGetInfDriverStoreLocationW SetupGetInfDriverStoreLocation 함수는 드라이버 저장소에서 시스템 INF 파일 디렉터리의 지정된 INF 파일 또는 드라이버 저장소의 지정된 INF 파일에 해당하는 INF 파일의 정규화된 파일 이름(디렉터리 경로 및 파일 이름)을 검색합니다. (유니코드)
SetupGetInfFileListA SetupGetInfFileList 함수는 호출자가 지정한 디렉터리에 있는 INF 파일 목록을 호출 제공 버퍼에 반환합니다. (ANSI)
SetupGetInfFileListW SetupGetInfFileList 함수는 호출자가 지정한 디렉터리에 있는 INF 파일 목록을 호출 제공 버퍼에 반환합니다. (유니코드)
SetupGetInfInformationA SetUpGetInfInformation 함수는 지정된 INF 파일의 SP_INF_INFORMATION 구조를 버퍼에 반환합니다. (ANSI)

SetupGetInfInformationW SetUpGetInfInformation 함수는 지정된 INF 파일의 SP_INF_INFORMATION 구조를 버퍼에 반환합니다. (유니코드)
SetupGetInfPublishedNameA SetupGetInfPublishedName 함수는 시스템 INF 파일 디렉터리에서 드라이버 저장소의 지정된 INF 파일 또는 시스템 INF 파일 디렉터리의 지정된 INF 파일에 해당하는 INF 파일의 정규화된 파일 이름(디렉터리 경로 및 파일 이름)을 검색합니다. (ANSI)
SetupGetInfPublishedNameW SetupGetInfPublishedName 함수는 시스템 INF 파일 디렉터리에서 드라이버 저장소의 지정된 INF 파일 또는 시스템 INF 파일 디렉터리의 지정된 INF 파일에 해당하는 INF 파일의 정규화된 파일 이름(디렉터리 경로 및 파일 이름)을 검색합니다. (유니코드)
SetupGetIntField SetupGetIntField 함수는 INF 파일에 있는 줄의 지정된 필드에서 정수 값을 검색합니다.
SetupGetLineByIndexA SetupGetLineByIndex 함수는 INF 파일의 지정된 섹션에서 인덱스 값으로 줄을 찾습니다. (ANSI)
SetupGetLineByIndexW SetupGetLineByIndex 함수는 INF 파일의 지정된 섹션에서 인덱스 값으로 줄을 찾습니다. (유니코드)
SetupGetLineCountA SetupGetLineCount 함수는 INF 파일의 지정된 섹션에 있는 줄 수를 반환합니다. (ANSI)
SetupGetLineCountW SetupGetLineCount 함수는 INF 파일의 지정된 섹션에 있는 줄 수를 반환합니다. (유니코드)
SetupGetLineTextA SetupGetLineText 함수는 INF 파일의 줄 내용을 압축 형식으로 반환합니다. (ANSI)
SetupGetLineTextW SetupGetLineText 함수는 INF 파일의 줄 내용을 압축 형식으로 반환합니다. (유니코드)

SetupGetMultiSzFieldA SetupGetMultiSzField 함수는 지정된 필드에서 줄 끝까지 INF 파일 줄에 저장된 여러 문자열을 검색합니다. (ANSI)
SetupGetMultiSzFieldW SetupGetMultiSzField 함수는 지정된 필드에서 줄 끝까지 INF 파일 줄에 저장된 여러 문자열을 검색합니다. (유니코드)
SetupGetNonInteractiveMode SetupGetNonInteractiveMode 함수는 호출자의 프로세스가 대화 상자와 같은 사용자 인터페이스 구성 요소를 통해 사용자와 상호 작용할 수 있는지 여부를 나타내는 SetupAPI 비대화형 플래그 값을 반환합니다.
SetupGetSourceFileLocationA SetupGetSourceFileLocation 함수는 INF 파일에 나열된 원본 파일의 위치를 검색합니다. (ANSI)
SetupGetSourceFileLocationW SetupGetSourceFileLocation 함수는 INF 파일에 나열된 원본 파일의 위치를 검색합니다. (유니코드)
SetupGetSourceFileSizeA SetupGetSourceFileSize 함수는 INF 파일에 나열된 소스 파일의 압축되지 않은 크기를 읽습니다. (ANSI)
SetupGetSourceFileSizeW SetupGetSourceFileSize 함수는 INF 파일에 나열된 소스 파일의 압축되지 않은 크기를 읽습니다. (유니코드)
SetupGetSourceInfoA SetupGetSourceInfo 함수는 INF 파일에 나열된 원본에 대한 경로, 태그 파일 또는 미디어 설명을 검색합니다. (ANSI)
SetupGetSourceInfoW SetupGetSourceInfo 함수는 INF 파일에 나열된 원본에 대한 경로, 태그 파일 또는 미디어 설명을 검색합니다. (유니코드)
SetupGetStringFieldA SetupGetStringField 함수는 INF 파일의 지정된 줄 필드에서 문자열을 검색합니다. (ANSI)

SetupGetStringFieldW SetupGetStringField 함수는 INF 파일의 지정된 줄 필드에서 문자열을 검색합니다. (유니코드)
SetupGetTargetPathA SetupGetTargetPath 함수는 파일 목록 섹션의 대상 디렉터리를 결정합니다. (ANSI)
SetupGetTargetPathW SetupGetTargetPath 함수는 파일 목록 섹션의 대상 디렉터리를 결정합니다. (유니코드)
SetupGetThreadLogToken SetupGetThreadLogToken 함수는 이 함수가 호출된 스레드에 대한 로그 토큰을 검색합니다.
SetupInitDefaultQueueCallback SetupInitDefaultQueueCallback 함수는 설치 API에 포함된 기본 큐 콜백 루틴에서 사용하는 컨텍스트를 초기화합니다.
SetupInitDefaultQueueCallbackEx 진행률 메시지를 수락하기 위해 콜백 함수에 추가 창이 제공된다는 점을 제외하고 SetupInitDefaultQueueCallback과 동일한 방식으로 설치 API에 포함된 기본 큐 콜백 루틴에서 사용하는 컨텍스트를 초기화합니다.
SetupInitializeFileLogA SetupInitializeFileLog 함수는 설치 작업 및 결과를 기록하기 위해 파일을 초기화합니다. 시스템이 Windows의 일부로 설치된 파일 또는 다른 파일을 추적하는 시스템 로그일 수 있습니다. (ANSI)
SetupInitializeFileLogW SetupInitializeFileLog 함수는 설치 작업 및 결과를 기록하기 위해 파일을 초기화합니다. 시스템이 Windows의 일부로 설치된 파일 또는 다른 파일을 추적하는 시스템 로그일 수 있습니다. (유니코드)
SetupInstallFileA SetupInstallFile 함수는 SetupFindXXXLine에서 반환된 INFCONTEXT 또는 파일 이름 및 경로에 의해 명시적으로 지정된 대로 파일을 설치합니다. (ANSI)
SetupInstallFileExA SetupInstallFileEx 함수는 SetupFindXXXLine에서 반환된 INFCONTEXT 또는 파일 이름 및 경로 정보에 의해 명시적으로 지정된 대로 파일을 설치합니다. (ANSI)

SetupInstallFileExW SetupInstallFileEx 함수는 SetupFindXXXLine에서 반환된 INFCONTEXT 또는 파일 이름 및 경로 정보에 의해 명시적으로 지정된 대로 파일을 설치합니다. (유니코드)
SetupInstallFilesFromInfSectionA SetupInstallFilesFromInfSection 함수는 설치 섹션에 나열된 파일 복사, 파일 삭제 및 파일 이름 바꾸기 섹션에 지정된 설치에 대한 모든 파일을 큐에 대기합니다. (ANSI)
SetupInstallFilesFromInfSectionW SetupInstallFilesFromInfSection 함수는 설치 섹션에 나열된 파일 복사, 파일 삭제 및 파일 이름 바꾸기 섹션에 지정된 설치에 대한 모든 파일을 큐에 대기합니다. (유니코드)
SetupInstallFileW SetupInstallFile 함수는 SetupFindXXXLine에서 반환된 INFCONTEXT 또는 파일 이름 및 경로에 의해 명시적으로 지정된 대로 파일을 설치합니다. (유니코드)
SetupInstallFromInfSectionA SetupInstallFromInfSection 함수는 INF 파일 설치 섹션의 모든 지시문을 수행합니다. (ANSI)
SetupInstallFromInfSectionW SetupInstallFromInfSection 함수는 INF 파일 설치 섹션의 모든 지시문을 수행합니다. (유니코드)
SetupInstallServicesFromInfSectionA SetupInstallServicesFromInfSection 함수는 INF 파일의 서비스 섹션에 나열된 서비스 설치 섹션에 지정된 서비스 설치 및 삭제 작업을 수행합니다. (ANSI)
SetupInstallServicesFromInfSectionExA SetupInstallServicesFromInfSectionEx 함수는 INF 파일의 서비스 섹션에 나열된 서비스 설치 섹션에 지정된 서비스 설치 및 삭제 작업을 수행합니다. (ANSI)
SetupInstallServicesFromInfSectionExW SetupInstallServicesFromInfSectionEx 함수는 INF 파일의 서비스 섹션에 나열된 서비스 설치 섹션에 지정된 서비스 설치 및 삭제 작업을 수행합니다. (유니코드)
SetupInstallServicesFromInfSectionW SetupInstallServicesFromInfSection 함수는 INF 파일의 서비스 섹션에 나열된 서비스 설치 섹션에 지정된 서비스 설치 및 삭제 작업을 수행합니다. (유니코드)

SetupIterateCabinetA SetupIterateCabinet 함수는 캐비닛의 모든 파일을 반복하고 찾은 각 파일에 대한 콜백 함수에 알림을 보냅니다. (ANSI)
SetupIterateCabinetW SetupIterateCabinet 함수는 캐비닛의 모든 파일을 반복하고 찾은 각 파일에 대한 콜백 함수에 알림을 보냅니다. (유니코드)
SetupLogErrorA SetupLogError 함수는 로그 파일에 오류 메시지를 씁니다. (ANSI)
SetupLogErrorW SetupLogError 함수는 로그 파일에 오류 메시지를 씁니다. (유니코드)
SetupLogFileA SetupLogFile 함수는 로그 파일에 항목을 추가합니다. (ANSI)
SetupLogFileW SetupLogFile 함수는 로그 파일에 항목을 추가합니다. (유니코드)
SetupOpenAppendInfFileA SetupOpenAppendInfFile 함수는 INF 파일의 정보를 SetupOpenInfFile에서 이전에 연 INF 파일에 추가합니다. (ANSI)
SetupOpenAppendInfFileW SetupOpenAppendInfFile 함수는 INF 파일의 정보를 SetupOpenInfFile에서 이전에 연 INF 파일에 추가합니다. (유니코드)
SetupOpenFileQueue SetupOpenFileQueue 함수는 설치 파일 큐를 만듭니다.
SetupOpenInfFileA SetupOpenInfFile 함수는 INF 파일을 열고 핸들을 반환합니다. (ANSI)
SetupOpenInfFileW SetupOpenInfFile 함수는 INF 파일을 열고 핸들을 반환합니다. (유니코드)

SetupOpenLog SetupOpenLog 함수가 로그 파일을 엽니다.
SetupOpenMasterInf SetupOpenMasterInf 함수는 Windows와 함께 제공되는 파일에 대한 파일 및 레이아웃 정보를 포함하는 마스터 INF 파일을 엽니다.
SetupPromptForDiskA SetupPromptForDisk 함수는 사용자에게 디스크를 묻는 대화 상자를 표시합니다. (ANSI)
SetupPromptForDiskW SetupPromptForDisk 함수는 사용자에게 디스크를 묻는 대화 상자를 표시합니다. (유니코드)
SetupPromptReboot SetupPromptReboot 함수는 필요에 따라 커밋된 파일 큐의 파일이 파일 작업 중에 사용 중인지 여부에 따라 시스템을 다시 부팅할지 사용자에게 묻습니다.
SetupQueryDrivesInDiskSpaceListA SetupQueryDrivesInDiskSpaceList 함수는 디스크 공간 목록에 나열된 파일 작업에서 참조하는 드라이브 목록으로 버퍼를 채웁니다. (ANSI)
SetupQueryDrivesInDiskSpaceListW SetupQueryDrivesInDiskSpaceList 함수는 디스크 공간 목록에 나열된 파일 작업에서 참조하는 드라이브 목록으로 버퍼를 채웁니다. (유니코드)
SetupQueryFileLogA SetupQueryFileLog 함수는 설치 파일 로그에서 정보를 반환합니다. (ANSI)
SetupQueryFileLogW SetupQueryFileLog 함수는 설치 파일 로그에서 정보를 반환합니다. (유니코드)
SetupQueryInfFileInformationA SetupQueryInfFileInformation 함수는 SP_INF_INFORMATION 구조체에서 버퍼로 INF 파일 이름을 반환합니다. (ANSI)
SetupQueryInfFileInformationW SetupQueryInfFileInformation 함수는 SP_INF_INFORMATION 구조체에서 버퍼로 INF 파일 이름을 반환합니다. (유니코드)

SetupQueryInfOriginalFileInformationA SetupQueryInfOriginalFileInformation 함수는 OEM INF 파일의 원래 이름을 반환합니다. (ANSI)
SetupQueryInfOriginalFileInformationW SetupQueryInfOriginalFileInformation 함수는 OEM INF 파일의 원래 이름을 반환합니다. (유니코드)
SetupQueryInfVersionInformationA SetupQueryInfVersionInformation 함수는 SP_INF_INFORMATION 구조체에서 버퍼로 INF 파일 버전 정보를 반환합니다. (ANSI)
SetupQueryInfVersionInformationW SetupQueryInfVersionInformation 함수는 SP_INF_INFORMATION 구조체에서 버퍼로 INF 파일 버전 정보를 반환합니다. (유니코드)
SetupQuerySourceListA SetupQuerySourceList 함수는 현재 설치 원본 목록을 쿼리합니다. 목록은 시스템 및 사용자별 목록에서 작성되며 임시 목록에 의해 재정의될 수 있습니다(SetupSetSourceList 참조). (ANSI)
SetupQuerySourceListW SetupQuerySourceList 함수는 현재 설치 원본 목록을 쿼리합니다. 목록은 시스템 및 사용자별 목록에서 작성되며 임시 목록에 의해 재정의될 수 있습니다(SetupSetSourceList 참조). (유니코드)
SetupQuerySpaceRequiredOnDriveA SetupQuerySpaceRequiredOnDrive 함수는 디스크 공간 목록을 검사하여 특정 드라이브에 대해 나열된 모든 파일 작업을 수행하는 데 필요한 공간을 결정합니다. (ANSI)
SetupQuerySpaceRequiredOnDriveW SetupQuerySpaceRequiredOnDrive 함수는 디스크 공간 목록을 검사하여 특정 드라이브에 대해 나열된 모든 파일 작업을 수행하는 데 필요한 공간을 결정합니다. (유니코드)
SetupQueueCopyA SetupQueueCopy 함수는 설치 파일 큐에 단일 파일 복사 작업을 추가합니다. (ANSI)
SetupQueueCopyIndirectA SetupQueueCopyIndirect 함수는 추가 매개 변수를 구조체(SP_FILE_COPY_PARAMS)로 전달하는 확장된 형태의 SetupQueueCopy입니다. 이 외에는 동작이 동일합니다. (ANSI)

SetupQueueCopyIndirectW SetupQueueCopyIndirect 함수는 추가 매개 변수를 구조체(SP_FILE_COPY_PARAMS)로 전달하는 확장된 형태의 SetupQueueCopy입니다. 이 외에는 동작이 동일합니다. (유니코드)
SetupQueueCopySectionA SetupQueueCopySection 함수는 복사를 위해 INF 파일의 섹션에 있는 모든 파일을 설치 큐에 배치합니다. (ANSI)
SetupQueueCopySectionW SetupQueueCopySection 함수는 복사를 위해 INF 파일의 섹션에 있는 모든 파일을 설치 큐에 배치합니다. (유니코드)
SetupQueueCopyW SetupQueueCopy 함수는 설치 파일 큐에 단일 파일 복사 작업을 추가합니다. (유니코드)
SetupQueueDefaultCopyA SetupQueueDefaultCopy 함수는 INF 파일에 지정된 기본 원본 미디어 및 대상을 사용하여 복사를 위해 설치 파일 큐에 단일 파일을 추가합니다. (ANSI)
SetupQueueDefaultCopyW SetupQueueDefaultCopy 함수는 INF 파일에 지정된 기본 원본 미디어 및 대상을 사용하여 복사를 위해 설치 파일 큐에 단일 파일을 추가합니다. (유니코드)
SetupQueueDeleteA SetupQueueDelete 함수는 개별 파일 삭제 작업을 설치 파일 큐에 배치합니다. (ANSI)
SetupQueueDeleteSectionA SetupQueueDeleteSection 함수는 삭제를 위해 INF 파일의 섹션에 있는 모든 파일을 큐에 대기합니다. 섹션은 올바른 파일 삭제 형식이어야 하며 INF 파일에 DestinationDirs 섹션이 포함되어야 합니다. (ANSI)
SetupQueueDeleteSectionW SetupQueueDeleteSection 함수는 삭제를 위해 INF 파일의 섹션에 있는 모든 파일을 큐에 대기합니다. 섹션은 올바른 파일 삭제 형식이어야 하며 INF 파일에 DestinationDirs 섹션이 포함되어야 합니다. (유니코드)
SetupQueueDeleteW SetupQueueDelete 함수는 개별 파일 삭제 작업을 설치 파일 큐에 배치합니다. (유니코드)

SetupQueueRenameA SetupQueueRename 함수는 설치 파일 큐에 개별 파일 이름 바꾸기 작업을 배치합니다. (ANSI)
SetupQueueRenameSectionA SetupQueueRenameSection 함수는 이름을 바꾸기 위해 INF 파일의 섹션을 큐에 대기합니다. 섹션은 올바른 이름 바꾸기 목록 섹션 형식이어야 하며 INF 파일에 DestinationDirs 섹션이 포함되어야 합니다. (ANSI)
SetupQueueRenameSectionW SetupQueueRenameSection 함수는 이름을 바꾸기 위해 INF 파일의 섹션을 큐에 대기합니다. 섹션은 올바른 이름 바꾸기 목록 섹션 형식이어야 하며 INF 파일에 DestinationDirs 섹션이 포함되어야 합니다. (유니코드)
SetupQueueRenameW SetupQueueRename 함수는 설치 파일 큐에 개별 파일 이름 바꾸기 작업을 배치합니다. (유니코드)
SetupRemoveFileLogEntryA SetupRemoveFileLogEntry 함수는 파일 로그에서 항목 또는 섹션을 제거합니다. (ANSI)
SetupRemoveFileLogEntryW SetupRemoveFileLogEntry 함수는 파일 로그에서 항목 또는 섹션을 제거합니다. (유니코드)
SetupRemoveFromDiskSpaceListA SetupRemoveFromDiskSpaceList 함수는 디스크 공간 목록에서 파일 삭제 또는 복사 작업을 제거합니다. (ANSI)
SetupRemoveFromDiskSpaceListW SetupRemoveFromDiskSpaceList 함수는 디스크 공간 목록에서 파일 삭제 또는 복사 작업을 제거합니다. (유니코드)
SetupRemoveFromSourceListA SetupRemoveFromSourceList 함수는 현재 사용자 또는 시스템의 설치 원본 목록에서 값을 제거합니다. 시스템 및 사용자 목록은 런타임에 병합됩니다. (ANSI)
SetupRemoveFromSourceListW SetupRemoveFromSourceList 함수는 현재 사용자 또는 시스템의 설치 원본 목록에서 값을 제거합니다. 시스템 및 사용자 목록은 런타임에 병합됩니다. (유니코드)

SetupRemoveInstallSectionFromDiskSpaceListA SetupRemoveInstallSectionFromDiskSpaceList 함수는 INF 파일의 설치 섹션에서 CopyFiles 및 DelFiles 줄을 검색하고 디스크 공간 목록에서 해당 섹션에 지정된 파일 작업을 제거합니다. (ANSI)
SetupRemoveInstallSectionFromDiskSpaceListW SetupRemoveInstallSectionFromDiskSpaceList 함수는 INF 파일의 설치 섹션에서 CopyFiles 및 DelFiles 줄을 검색하고 디스크 공간 목록에서 해당 섹션에 지정된 파일 작업을 제거합니다. (유니코드)
SetupRemoveSectionFromDiskSpaceListA SetupRemoveSectionFromDiskSpaceList 함수는 디스크 공간 목록에서 INF 파일의 파일 복사 섹션에 나열된 파일 삭제 또는 복사 작업을 제거합니다. (ANSI)
SetupRemoveSectionFromDiskSpaceListW SetupRemoveSectionFromDiskSpaceList 함수는 디스크 공간 목록에서 INF 파일의 파일 복사 섹션에 나열된 파일 삭제 또는 복사 작업을 제거합니다. (유니코드)
SetupRenameErrorA RenameError 함수는 파일 이름 바꾸기 오류를 사용자에게 알리는 대화 상자를 생성합니다. (ANSI)
SetupRenameErrorW RenameError 함수는 파일 이름 바꾸기 오류를 사용자에게 알리는 대화 상자를 생성합니다. (유니코드)
SetupScanFileQueueA SetupScanFileQueue 함수는 설치 파일 큐를 검색하여 복사 목록의 각 노드에서 작업을 수행합니다. 작업은 플래그 집합으로 지정됩니다. 큐가 커밋되기 전이나 후에 이 함수를 호출할 수 있습니다. (ANSI)
SetupScanFileQueueW SetupScanFileQueue 함수는 설치 파일 큐를 검색하여 복사 목록의 각 노드에서 작업을 수행합니다. 작업은 플래그 집합으로 지정됩니다. 큐가 커밋되기 전이나 후에 이 함수를 호출할 수 있습니다. (유니코드)
SetupSetDirectoryIdA SetupSetDirectoryId 함수는 INF 파일의 디렉터리 식별자를 특정 디렉터리에 연결합니다. (ANSI)
SetupSetDirectoryIdExA SetupSetDirectoryIdEx 함수는 INF 파일의 디렉터리 식별자를 특정 디렉터리에 연결합니다. (ANSI)

SetupSetDirectoryIdExW SetupSetDirectoryIdEx 함수는 INF 파일의 디렉터리 식별자를 특정 디렉터리에 연결합니다. (유니코드)
SetupSetDirectoryIdW SetupSetDirectoryId 함수는 INF 파일의 디렉터리 식별자를 특정 디렉터리에 연결합니다. (유니코드)
SetupSetFileQueueAlternatePlatformA SetupSetFileQueueAlternatePlatform 함수는 파일 큐를 함수를 실행하는 플랫폼과 다른 대상 플랫폼과 연결합니다. 이 작업은 네이티브가 아닌 서명 확인을 사용하도록 설정하기 위해 수행됩니다. (ANSI)
SetupSetFileQueueAlternatePlatformW SetupSetFileQueueAlternatePlatform 함수는 파일 큐를 함수를 실행하는 플랫폼과 다른 대상 플랫폼과 연결합니다. 이 작업은 네이티브가 아닌 서명 확인을 사용하도록 설정하기 위해 수행됩니다. (유니코드)
SetupSetFileQueueFlags SetupSetFileQueueFlags 함수는 설치 파일 큐의 플래그를 설정합니다.
SetupSetNonInteractiveMode SetupSetNonInteractiveMode 함수는 SetupAPI가 호출자의 컨텍스트에서 사용자와 상호 작용할 수 있는지 여부를 결정하는 비대화형 SetupAPI 플래그를 설정합니다.
SetupSetPlatformPathOverrideA SetupSetPlatformPathOverride 함수는 다른 컴퓨터의 INF로 작업할 때 대상 머신에 대한 플랫폼 경로 재정의를 설정하는 데 사용됩니다. (ANSI)
SetupSetPlatformPathOverrideW SetupSetPlatformPathOverride 함수는 다른 컴퓨터의 INF로 작업할 때 대상 머신에 대한 플랫폼 경로 재정의를 설정하는 데 사용됩니다. (유니코드)
SetupSetSourceListA SetupSetSourceList 함수를 사용하면 호출자가 현재 사용자 또는 시스템(모든 사용자에게 공통)에 대한 설치 원본 목록을 설정할 수 있습니다. (ANSI)

SetupSetSourceListW SetupSetSourceList 함수를 사용하면 호출자가 현재 사용자 또는 시스템(모든 사용자에게 공통)에 대한 설치 원본 목록을 설정할 수 있습니다. (유니코드)
SetupSetThreadLogToken SetupSetThreadLogToken 함수는 이 함수가 호출된 스레드에 대해 로그 토큰으로 표시되는 로그 컨텍스트를 설정합니다.
SetupTermDefaultQueueCallback 큐 커밋이 완료된 후 SetupTermDefaultQueueCallback 함수가 호출됩니다. SetupInitDefaultQueueCallback 또는 SetupInitDefaultQueueCallbackEx에 대한 이전 호출에 의해 할당된 리소스를 해제합니다.
SetupTerminateFileLog SetupTerminateFileLog 함수는 파일 로그와 연결된 리소스를 해제합니다.
SetupUninstallNewlyCopiedInfs SetupUninstallNewlyCopiedInfs 함수는 지정된 파일 큐의 커밋 중에 이전에 설치된 INF 파일(.inf), 미리 컴파일된 INF 파일(.pnf) 및 카탈로그 파일(.cat)을 제거합니다.
SetupUninstallOEMInfA SetupUninstallOEMInf 함수는 지정된 .inf 파일 및 연결된 모든 .pnf 파일을 제거합니다. (ANSI)
SetupUninstallOEMInfW SetupUninstallOEMInf 함수는 지정된 .inf 파일 및 연결된 모든 .pnf 파일을 제거합니다. (유니코드)
SetupVerifyInfFileA SetupVerifyInfFile 함수는 해당 카탈로그를 사용하여 지정된 INF 파일의 디지털 서명을 확인합니다. 대체 플랫폼에 대해 확인을 수행할 수 있습니다. (ANSI)
SetupVerifyInfFileW SetupVerifyInfFile 함수는 해당 카탈로그를 사용하여 지정된 INF 파일의 디지털 서명을 확인합니다. 대체 플랫폼에 대해 확인을 수행할 수 있습니다. (유니코드)
SetupWriteTextLog SetupWriteTextLog 함수는 SetupAPI 텍스트 로그에 로그 항목을 씁니다.

SetupWriteTextLogError SetupWriteTextLogError 함수는 SetupAPI 관련 오류 또는 Win32 시스템 오류에 대한 정보를 SetupAPI 텍스트 로그에 씁니다.
SetupWriteTextLogInfLine SetupWriteTextLogInfLine 함수는 지정된 INF 파일 줄의 텍스트가 포함된 SetupAPI 텍스트 로그에 로그 항목을 씁니다.

콜백 함수

PSP_FILE_CALLBACK_A FileCallback 콜백 함수는 몇 가지 설정 함수에서 사용됩니다. (ANSI)
PSP_FILE_CALLBACK_W FileCallback 콜백 함수는 몇 가지 설정 함수에서 사용됩니다. (유니코드)

구조체

CABINET_INFO_A CABINET_INFO 구조체는 캐비닛 파일에 대한 정보를 저장합니다. SetupIterateCabinet 함수는 캐비닛 콜백 루틴에 SPFILENOTIFY_NEEDNEWCABINET 알림을 보낼 때 이 구조를 매개 변수로 지정합니다. (ANSI)
CABINET_INFO_W CABINET_INFO 구조체는 캐비닛 파일에 대한 정보를 저장합니다. SetupIterateCabinet 함수는 캐비닛 콜백 루틴에 SPFILENOTIFY_NEEDNEWCABINET 알림을 보낼 때 이 구조를 매개 변수로 지정합니다. (유니코드)
FILE_IN_CABINET_INFO_A FILE_IN_CABINET_INFO 구조체는 캐비닛에 있는 파일에 대한 정보를 제공합니다. (ANSI)

FILE_IN_CABINET_INFO_W FILE_IN_CABINET_INFO 구조체는 캐비닛에 있는 파일에 대한 정보를 제공합니다. (유니코드)
FILEPATHS_A FILEPATHS 구조는 원본 및 대상 경로 정보를 저장합니다. 설치 함수는 FILEPATHS 구조를 콜백 루틴으로 전송된 여러 알림의 매개 변수로 보냅니다. 자세한 내용은 알림을 참조하세요. (ANSI)
FILEPATHS_SIGNERINFO_A FILEPATHS_SINGNERINFO 구조는 원본 및 대상 경로 정보와 파일 서명 정보를 저장합니다. (ANSI)
FILEPATHS_SIGNERINFO_W FILEPATHS_SINGNERINFO 구조는 원본 및 대상 경로 정보와 파일 서명 정보를 저장합니다. (유니코드)
FILEPATHS_W FILEPATHS 구조는 원본 및 대상 경로 정보를 저장합니다. 설치 함수는 FILEPATHS 구조를 콜백 루틴으로 전송된 여러 알림의 매개 변수로 보냅니다. 자세한 내용은 알림을 참조하세요. (유니코드)
INFCONTEXT INFCONTEXT 구조체는 SetupGetLineText와 같은 함수가 INF 파일을 탐색하는 데 사용하는 컨텍스트 정보를 저장합니다.
SOURCE_MEDIA_A SOURCE_MEDIA 구조는 원본 미디어 정보를 전달하기 위해 SPFILENOTIFY_NEEDMEDIA 알림과 함께 사용됩니다. (ANSI)
SOURCE_MEDIA_W SOURCE_MEDIA 구조는 원본 미디어 정보를 전달하기 위해 SPFILENOTIFY_NEEDMEDIA 알림과 함께 사용됩니다. (유니코드)
SP_ALTPLATFORM_INFO_V1 이 구조는 대체 플랫폼에 대한 정보를 SetupQueryInfOriginalFileInformation에 전달하는 데 사용됩니다.
SP_ALTPLATFORM_INFO_V2 SP_ALTPLATFORM_INFO_V2 구조체는 대체 플랫폼에 대한 정보를 SetupQueryInfOriginalFileInformation에 전달하는 데 사용됩니다.

SP_CLASSIMAGELIST_DATA SP_CLASSIMAGELIST_DATA 구조체는 클래스 이미지 목록을 설명합니다.
SP_CLASSINSTALL_HEADER SP_CLASSINSTALL_HEADER 클래스 설치 매개 변수 구조의 첫 번째 멤버입니다. 여기에는 나머지 설치 매개 변수 구조의 형식을 정의하는 디바이스 설치 요청 코드가 포함되어 있습니다.
SP_DETECTDEVICE_PARAMS SP_DETECTDEVICE_PARAMS 구조는 DIF_DETECT 설치 요청에 해당합니다.
SP_DEVICE_INTERFACE_DATA SP_DEVICE_INTERFACE_DATA 구조는 디바이스 정보 집합에서 디바이스 인터페이스를 정의합니다.
SP_DEVICE_INTERFACE_DETAIL_DATA_A SP_DEVICE_INTERFACE_DETAIL_DATA 구조체에는 디바이스 인터페이스에 대한 경로가 포함됩니다. (ANSI)
SP_DEVICE_INTERFACE_DETAIL_DATA_W SP_DEVICE_INTERFACE_DETAIL_DATA 구조체에는 디바이스 인터페이스에 대한 경로가 포함됩니다. (유니코드)
SP_DEVINFO_DATA SP_DEVINFO_DATA 구조는 디바이스 정보 집합의 멤버인 디바이스 인스턴스를 정의합니다.
SP_DEVINFO_LIST_DETAIL_DATA_A SP_DEVINFO_LIST_DETAIL_DATA 구조에는 연결된 설치 클래스 GUID(연결된 설치 클래스가 있는 경우)와 같은 디바이스 정보 집합에 대한 정보가 포함됩니다. (ANSI)
SP_DEVINFO_LIST_DETAIL_DATA_W SP_DEVINFO_LIST_DETAIL_DATA 구조에는 연결된 설치 클래스 GUID(연결된 설치 클래스가 있는 경우)와 같은 디바이스 정보 집합에 대한 정보가 포함됩니다. (유니코드)
SP_DEVINSTALL_PARAMS_A SP_DEVINSTALL_PARAMS 구조에는 특정 디바이스 정보 요소와 연결되거나 디바이스 정보 집합과 전역적으로 연결된 디바이스 설치 매개 변수가 포함됩니다. (ANSI)

SP_DEVINSTALL_PARAMS_W SP_DEVINSTALL_PARAMS 구조에는 특정 디바이스 정보 요소와 연결되거나 디바이스 정보 집합과 전역적으로 연결된 디바이스 설치 매개 변수가 포함됩니다. (유니코드)
SP_DRVINFO_DATA_V1_A SP_DRVINFO_DATA 구조에는 드라이버에 대한 정보가 포함됩니다. 이 구조는 특정 디바이스 인스턴스 또는 전역적으로 디바이스 정보 집합과 연결할 수 있는 드라이버 정보 목록의 멤버입니다. (sp_drvinfo_data_v1_a)
SP_DRVINFO_DATA_V1_W SP_DRVINFO_DATA 구조에는 드라이버에 대한 정보가 포함됩니다. 이 구조는 특정 디바이스 인스턴스 또는 전역적으로 디바이스 정보 집합과 연결할 수 있는 드라이버 정보 목록의 멤버입니다. (sp_drvinfo_data_v1_w)
SP_DRVINFO_DATA_V2_A SP_DRVINFO_DATA 구조에는 드라이버에 대한 정보가 포함됩니다. 이 구조는 특정 디바이스 인스턴스 또는 전역적으로 디바이스 정보 집합과 연결할 수 있는 드라이버 정보 목록의 멤버입니다. (sp_drvinfo_data_v2_a)
SP_DRVINFO_DATA_V2_W SP_DRVINFO_DATA 구조에는 드라이버에 대한 정보가 포함됩니다. 이 구조는 특정 디바이스 인스턴스 또는 전역적으로 디바이스 정보 집합과 연결할 수 있는 드라이버 정보 목록의 멤버입니다. (sp_drvinfo_data_v2_w)
SP_DRVINFO_DETAIL_DATA_A SP_DRVINFO_DETAIL_DATA 구조에는 특정 드라이버 정보 구조에 대한 자세한 정보가 포함됩니다. (ANSI)
SP_DRVINFO_DETAIL_DATA_W SP_DRVINFO_DETAIL_DATA 구조에는 특정 드라이버 정보 구조에 대한 자세한 정보가 포함됩니다. (유니코드)
SP_DRVINSTALL_PARAMS SP_DRVINSTALL_PARAMS 구조에는 특정 드라이버 정보 요소와 연결된 드라이버 설치 매개 변수가 포함됩니다.
SP_FILE_COPY_PARAMS_A SP_FILE_COPY_PARAMS 구조는 단일 파일 복사 작업을 설명합니다. (ANSI)

SP_FILE_COPY_PARAMS_W SP_FILE_COPY_PARAMS 구조는 단일 파일 복사 작업을 설명합니다. (유니코드)
SP_INF_INFORMATION SP_INF_INFORMATION 구조체는 스타일, 구성 INF 파일 수 및 버전 데이터를 포함하여 INF 파일에 대한 정보를 저장합니다.
SP_INF_SIGNER_INFO_V1_A SP_INF_SIGNER_INFO 구조체는 INF 파일의 디지털 서명에 대한 정보를 저장합니다. (sp_inf_signer_info_v1_a)
SP_INF_SIGNER_INFO_V1_W SP_INF_SIGNER_INFO 구조체는 INF 파일의 디지털 서명에 대한 정보를 저장합니다. (sp_inf_signer_info_v1_w)
SP_INF_SIGNER_INFO_V2_A SP_INF_SIGNER_INFO 구조체는 INF 파일의 디지털 서명에 대한 정보를 저장합니다. (sp_inf_signer_info_v2_a)
SP_INF_SIGNER_INFO_V2_W SP_INF_SIGNER_INFO 구조체는 INF 파일의 디지털 서명에 대한 정보를 저장합니다. (sp_inf_signer_info_v2_w)
SP_NEWDEVICEWIZARD_DATA SP_NEWDEVICEWIZARD_DATA 구조는 설치 관리자가 사용자 지정 페이지를 추가하여 하드웨어 설치 마법사의 작업을 확장하는 데 사용됩니다. DIF_NEWDEVICEWIZARD_XXX 설치 요청과 함께 사용됩니다.
SP_ORIGINAL_FILE_INFO_A SP_ORIGINAL_FILE_INFO 구조체는 SetupQueryInfOriginalFileInformation에서 반환된 원래 INF 파일 이름 및 카탈로그 파일 정보를 받습니다. (ANSI)
SP_ORIGINAL_FILE_INFO_W SP_ORIGINAL_FILE_INFO 구조체는 SetupQueryInfOriginalFileInformation에서 반환된 원래 INF 파일 이름 및 카탈로그 파일 정보를 받습니다. (유니코드)
SP_POWERMESSAGEWAKE_PARAMS_A SP_POWERMESSAGEWAKE_PARAMS 구조는 DIF_POWERMESSAGEWAKE 설치 요청에 해당합니다. (ANSI)

SP_POWERMESSAGEWAKE_PARAMS_W SP_POWERMESSAGEWAKE_PARAMS 구조는 DIF_POWERMESSAGEWAKE 설치 요청에 해당합니다. (유니코드)
SP_PROPCHANGE_PARAMS SP_PROPCHANGE_PARAMS 구조는 DIF_PROPERTYCHANGE 설치 요청에 해당합니다.
SP_PROPSHEETPAGE_REQUEST SP_PROPSHEETPAGE_REQUEST 구조체를 SetupAPI DLL의 ExtensionPropSheetPageProc 진입점에 첫 번째 매개 변수(lpv)로 전달할 수 있습니다.
SP_REGISTER_CONTROL_STATUSA SP_REGISTER_CONTROL_STATUS 구조에는 Windows 2000에서 DLL을 자체 등록하기 위해 RegisterDlls INF 지시문을 사용하여 등록 또는 등록 취소되는 파일에 대한 정보가 포함되어 있습니다. (ANSI)
SP_REGISTER_CONTROL_STATUSW SP_REGISTER_CONTROL_STATUS 구조에는 Windows 2000에서 DLL을 자체 등록하기 위해 RegisterDlls INF 지시문을 사용하여 등록 또는 등록 취소되는 파일에 대한 정보가 포함되어 있습니다. (유니코드)
SP_REMOVEDEVICE_PARAMS SP_REMOVEDEVICE_PARAMS 구조는 DIF_REMOVE 설치 요청에 해당합니다.
SP_SELECTDEVICE_PARAMS_A SP_SELECTDEVICE_PARAMS 구조는 DIF_SELECTDEVICE 설치 요청에 해당합니다. (ANSI)
SP_SELECTDEVICE_PARAMS_W SP_SELECTDEVICE_PARAMS 구조는 DIF_SELECTDEVICE 설치 요청에 해당합니다. (유니코드)
SP_TROUBLESHOOTER_PARAMS_A SP_TROUBLESHOOTER_PARAMS 구조는 DIF_TROUBLESHOOTER 설치 요청에 해당합니다. (ANSI)
SP_TROUBLESHOOTER_PARAMS_W SP_TROUBLESHOOTER_PARAMS 구조는 DIF_TROUBLESHOOTER 설치 요청에 해당합니다. (유니코드)

SP_UNREMOVEDEVICE_PARAMS SP_UNREMOVEDEVICE_PARAMS 구조는 DIF_UNREMOVE 설치 요청에 해당합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

sfc.h 헤더

아티클2023. 08. 26.

이 헤더는 애플리케이션 설치 및 서비스에 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 애플리케이션 설치 및 서비스

sfc.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

SfcGetNextProtectedFile 보호된 파일의 전체 목록을 검색합니다.
SfcIsFileProtected 지정된 파일이 보호되는지 여부를 결정합니다.
SfcIsKeyProtected 지정된 레지스트리 키가 보호되는지 여부를 확인합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

shappmgr.h 헤더

아티클2023. 08. 26.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

shappmgr.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAppPublisher 제어판 프로그램 추가/제거를 통해 애플리케이션을 게시하는 메서드를 노출합니다. 이 용도로 구현된 보안 주체 인터페이스입니다.
IEnumPublishedApps 게시된 애플리케이션을 제어판 프로그램 추가/제거에 열거하는 메서드를 노출합니다. 이 인터페이스를 노출하는 개체는 IAppPublisher::EnumApps를 통해 요청됩니다.
IPublishedApp 제어판 프로그램 추가/제거에 애플리케이션을 나타내는 메서드를 노출합니다.
IPublishedApp2 추가 설치 방법을 제공하여 IPublishedApp 인터페이스를 확장합니다.
IShellApp 애플리케이션에 대한 일반적인 정보를 제공하는 메서드를 프로그램 추가/제거 애플리케이션에 노출합니다.

구조체

APPINFODATA 게시된 애플리케이션에 대한 정보를 프로그램 추가/제거 제어판 유틸리티에 제공합니다.

PUBAPPINFO 애플리케이션 게시자에서 제어판 프로그램 추가/제거에 게시된 애플리케이션에 대한 정보를 제공합니다.
SLOWAPPINFO 제어판 프로그램 추가/제거에 대한 특수 애플리케이션 정보를 제공합니다. 이 구조는 게시된 애플리케이션에는 적용되지 않습니다.

열거형

APPACTIONFLAGS 애플리케이션 게시자가 지원하는 애플리케이션 관리 작업을 지정합니다. 이러한 플래그는 IShellApp::GetPossibleActions에 전달되는 비트 마스크입니다.
APPINFODATAFLAGS IShellApp::GetAppInfo에서 반환할 애플리케이션 정보를 지정합니다. 이러한 플래그는 APPINFODATA 구조체의 dwMask 멤버에 사용되는 비트 마스크입니다.
PUBAPPINFOFLAGS PUBAPPINFO 구조체에서 유효한 멤버를 지정합니다. 이러한 플래그는 dwMask 멤버에서 설정되고 IPublishedApp::GetPublishedAppInfo에 전달되는 비트 마스크입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

sharewindowcommandsourceinterop.h 헤더

아티클2024. 08. 08.

Windows.UI.Shell 네임스페이스를 사용하는 데스크톱 interop용 API입니다.

sharewindowcommandsourceinterop.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

테이블 확장

IShareWindowCommandEventArgsInterop 창 식별자(창 핸들)에 해당하는 ShareWindowCommandSource 개체를 가져옵니다. (IShareWindowCommandEventArgsInterop)
IShareWindowCommandSourceInterop ShareWindowCommandSource 개체를 가져오도록 허용하는 네이티브 상호 운용 인터페이스입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

shcore.h 헤더

아티클2023. 03. 08.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

shcore.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CreateRandomAccessStreamOnFile 파일에 대한 Windows 런타임 임의 액세스 스트림을 만듭니다.
CreateRandomAccessStreamOverStream IStream 기본 구현 주위에 Windows 런타임 임의 액세스 스트림을 만듭니다.
CreateStreamOverRandomAccessStream Windows 런타임 IRandomAccessStream 개체 주위에 IStream을 만듭니다.

열거형

BSOS_OPTIONS COM(구성 요소 개체 모델) IStream을 캡슐화하는 RandomAccessStream의 동작을 지정합니다.

shdeprecated.h 헤더

아티클2023. 08. 26.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

shdeprecated.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IBrowserService 더 이상 사용되지 않습니다. (IBrowserService)
IBrowserService2 더 이상 사용되지 않습니다. (IBrowserService2)
IBrowserService3 더 이상 사용되지 않습니다. (IBrowserService3)
IBrowserService4 더 이상 사용되지 않습니다. (IBrowserService4)
IExpDispSupport 더 이상 사용되지 않습니다. 속성 검색, 키보드 가속기 변환 및 특정 이벤트에 대한 연결점 결정을 허용하는 메서드를 노출합니다. (IExpDispSupport)
IExpDispSupportXP 더 이상 사용되지 않습니다. 속성 검색, 키보드 가속기 변환 및 특정 이벤트에 대한 연결점 결정을 허용하는 메서드를 노출합니다. (IExpDispSupportXP)
IShellService 더 이상 사용되지 않습니다. IShellService 특정 인터페이스를 구현하는 서비스 구성 요소가 Windows 인터넷 Explorer 및 Windows Explorer 같은 여러 클라이언트 간에 공유되는 경우 소유권을 선언하는 한 가지 메서드를 노출합니다.
ITrackShellMenu

메뉴와 함께 도구 모음 단추를 조정하고 팝업 메뉴를 표시하는 기능을 제공하여 IShellMenu 인터페이스를 확장하는 메서드를 노출합니다.
ITravelEntry 더 이상 사용되지 않습니다. 브라우저의 여행 기록에서 개별 항목을 식별, 호출 및 업데이트하는 메서드를 노출합니다.
ITravelLog 더 이상 사용되지 않습니다. 브라우저에서 이동 레코드를 유지 관리하고 조작하는 메서드를 노출합니다.

구조체

BASEBROWSERDATALH BASEBROWSERDATALH 구조체에는 기본 클래스의 보호된 멤버가 포함됩니다. (BASEBROWSERDATALH 구조체)
BASEBROWSERDATAXP BASEBROWSERDATAXP 구조체에는 기본 클래스의 보호된 멤버가 포함됩니다. (BASEBROWSERDATAXP 구조체)
FOLDERSETDATA 더 이상 사용되지 않습니다. IBrowserService2::GetFolderSetData에서 사용되는 데이터입니다.
TOOLBARITEM 더 이상 사용되지 않습니다. 도구 모음 항목을 정의하기 위해 IBrowserService2::_GetToolbarItem, IBrowserService2::v_MayGetNextToolbarFocus 및 IBrowserService2::_SetFocus 사용되는 데이터입니다.

열거형

BNSTATE 더 이상 사용되지 않습니다. IBrowserService::SetNavigateState 및 IBrowserService::GetNavigateState에서 탐색 상태를 지정하는 데 사용됩니다.

SECURELOCKCODE 더 이상 사용되지 않습니다. 이 열거형은 BASEBROWSERDATA 구조체에서 기본 브라우저의 잠금 아이콘 상태 나타내는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

shellapi.h 헤더

아티클2023. 03. 16.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸
- Windows 속성 시스템

shellapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

AssocCreateForClasses IQueryAssociations 인터페이스를 구현하는 개체를 검색합니다.
CommandLineToArgvW 유니코드 명령줄 문자열을 구문 분석하고 표준 C 런타임 argv 및 argc 값과 유사한 방식으로 이러한 인수의 수와 함께 명령줄 인수에 대한 포인터 배열을 반환합니다.
DoEnvironmentSubstA 하나 이상의 환경 변수에 대한 참조가 포함된 입력 문자열을 구문 분석하고 완전히 확장된 값으로 바꿉니다. (ANSI)
DoEnvironmentSubstW 하나 이상의 환경 변수에 대한 참조가 포함된 입력 문자열을 구문 분석하고 완전히 확장된 값으로 바꿉니다. (유니코드)
DragAcceptFiles 창에서 삭제된 파일을 허용하는지 여부를 등록합니다.
DragFinish 시스템이 애플리케이션에 파일 이름을 전송하는 데 사용하기 위해 할당한 메모리를 해제합니다.
DragQueryFileA 성공적으로 끌어서 놓기 작업으로 인해 삭제된 파일의 이름을 검색합니다. (ANSI)

DragQueryFileW 성공적으로 끌어서 놓기 작업으로 인해 삭제된 파일의 이름을 검색합니다. (유니코드)
DragQueryPoint 끌어서 놓기 작업 중에 파일을 삭제할 때 마우스 포인터의 위치를 검색합니다.
DuplicateIcon 지정된 아이콘의 중복을 만듭니다.
ExtractAssociatedIconA 파일에 리소스로 저장된 아이콘 또는 파일의 연결된 실행 파일에 저장된 아이콘에 대한 핸들을 가져옵니다. (ANSI)
ExtractAssociatedIconExA ExtractAssociatedIconEx를 변경하거나 사용할 수 없습니다. (ANSI)
ExtractAssociatedIconExW ExtractAssociatedIconEx를 변경하거나 사용할 수 없습니다. (유니코드)
ExtractAssociatedIconW 파일에 리소스로 저장된 아이콘 또는 파일의 연결된 실행 파일에 저장된 아이콘에 대한 핸들을 가져옵니다. (유니코드)
ExtractIconA 지정된 실행 파일, DLL 또는 아이콘 파일에서 아이콘에 대한 핸들을 가져옵니다. 크거나 작은 아이콘에 대한 핸들 배열을 검색하려면 ExtractIconEx 함수를 사용합니다. (ANSI)
ExtractIconExA ExtractIconEx 함수는 지정된 실행 파일, DLL 또는 아이콘 파일에서 추출된 크거나 작은 아이콘에 대한 핸들 배열을 만듭니다. (ANSI)
ExtractIconExW ExtractIconEx 함수는 지정된 실행 파일, DLL 또는 아이콘 파일에서 추출된 크거나 작은 아이콘에 대한 핸들 배열을 만듭니다. (유니코드)
ExtractIconW 지정된 실행 파일, DLL 또는 아이콘 파일에서 아이콘에 대한 핸들을 가져옵니다. 크거나 작은 아이콘에 대한 핸들 배열을 검색하려면 ExtractIconEx 함수를 사용합니다. (유니코드)

FindExecutableA 특정 문서 파일과 연결된 실행 파일(.exe) 파일의 이름과 핸들을 검색합니다. (ANSI)
FindExecutableW 특정 문서 파일과 연결된 실행 파일(.exe) 파일의 이름과 핸들을 검색합니다. (유니코드)
InitNetworkAddressControl 네트워크 주소 제어 창 클래스를 초기화합니다.
NetAddr_DisplayErrorTip 네트워크 주소 컨트롤과 연결된 풍선 팁에 오류 메시지를 표시합니다.
NetAddr_GetAddress 네트워크 주소가 지정된 형식 및 형식을 준수하는지 여부를 나타냅니다.
NetAddr_GetAllowType 지정된 네트워크 주소 컨트롤에서 허용하는 네트워크 주소 유형을 검색합니다.
NetAddr_SetAllowType 지정된 네트워크 주소 컨트롤에서 허용하는 네트워크 주소 유형을 설정합니다.
SHAppBarMessage 앱바 메시지를 시스템에 보냅니다.
SHCreateProcessAsUserW 지정된 실행 파일을 실행하는 새 사용자 모드 프로세스 및 기본 스레드를 만듭니다.
Shell_NotifyIconA 작업 표시줄의 상태 영역에 메시지를 보냅니다. (ANSI)
Shell_NotifyIconGetRect 알림 아이콘의 경계 사각형의 화면 좌표를 가져옵니다.
Shell_NotifyIconW 작업 표시줄의 상태 영역에 메시지를 보냅니다. (유니코드)

ShellAboutA ShellAbout 대화 상자를 표시합니다. (ANSI)
ShellAboutW ShellAbout 대화 상자를 표시합니다. (유니코드)
ShellExecuteA 지정된 파일에 대한 작업을 수행합니다. (ShellExecuteA)
ShellExecuteExA 지정된 파일에 대한 작업을 수행합니다. (ShellExecuteExA)
ShellExecuteExW 지정된 파일에 대해 작업을 수행합니다. (ShellExecuteExW)
ShellExecuteW 지정된 파일에 대해 작업을 수행합니다. (ShellExecuteW)
ShellMessageBoxA ShellMessageBox를 변경하거나 사용할 수 없습니다. (ANSI)
ShellMessageBoxW ShellMessageBox를 변경하거나 사용할 수 없습니다. (유니코드)
SHEmptyRecycleBinA 지정된 드라이브의 휴지통을 비웁니다. (ANSI)
SHEmptyRecycleBinW 지정된 드라이브의 휴지통을 비웁니다. (유니코드)
SHEnumerateUnreadMailAccountsA 읽지 않은 전자 메일이 있는 사용자 계정을 열거합니다. (ANSI)
SHEnumerateUnreadMailAccountsW 읽지 않은 전자 메일이 있는 사용자 계정을 열거합니다. (유니코드)

SHEvaluateSystemCommandTemplate CreateProcess 또는 ShellExecute 호출에 사용되는 매개 변수의 엄격한 유효성 검사를 적용합니다.
SHFileOperationA 파일 시스템 개체를 복사, 이동, 이름 바꾸기 또는 삭제합니다. 이 함수는 Windows Vista에서 IFileOperation으로 대체되었습니다.
SHFileOperationW 파일 시스템 개체를 복사, 이동, 이름 바꾸기 또는 삭제합니다. Windows Vista 이상 릴리스에서는 이 함수 대신 IFileOperation을 사용하는 것이 좋습니다.
SHFreeNameMappings SHFileOperation 함수에서 검색한 파일 이름 매핑 개체를 해제합니다.
SHGetDiskFreeSpaceExA 디스크 볼륨에 대한 디스크 공간 정보를 검색합니다. (ANSI)
SHGetDiskFreeSpaceExW 디스크 볼륨에 대한 디스크 공간 정보를 검색합니다. (유니코드)
SHGetDriveMedia 지정된 드라이브에 있는 미디어의 형식을 반환합니다.
SHGetFileInfoA 파일, 폴더, 디렉터리 또는 드라이브 루트와 같은 파일 시스템의 개체에 대한 정보를 검색합니다. (ANSI)
SHGetFileInfoW 파일, 폴더, 디렉터리 또는 드라이브 루트와 같은 파일 시스템의 개체에 대한 정보를 검색합니다. (유니코드)
SHGetImageList 이미지 목록을 검색합니다.
SHGetLocalizedName Shell 폴더에 있는 파일의 지역화된 이름을 검색합니다.

SHGetNewLinkInfoA 바로 가기의 제안된 대상에 따라 새 바로 가기의 이름을 만듭니다. 이 함수는 바로 가기를 만들지 않고 이름만 만듭니다. (ANSI)
SHGetNewLinkInfoW 바로 가기의 제안된 대상에 따라 새 바로 가기의 이름을 만듭니다. 이 함수는 바로 가기를 만들지 않고 이름만 만듭니다. (유니코드)
SHGetPropertyStoreForWindow 특정 창의 속성 컬렉션을 나타내는 개체를 검색하여 해당 속성을 쿼리하거나 설정할 수 있습니다.
SHGetStockIconInfo 시스템 정의 셸 아이콘에 대한 정보를 검색합니다.
SHGetUnreadMailCountA 모든 전자 메일 계정에 대해 지정된 사용자의 읽지 않은 메시지 수를 검색합니다. (ANSI)
SHGetUnreadMailCountW 모든 전자 메일 계정에 대해 지정된 사용자의 읽지 않은 메시지 수를 검색합니다. (유니코드)
SHInvokePrinterCommandA 프린터 개체에서 명령을 실행합니다. (ANSI)
SHInvokePrinterCommandW 프린터 개체에서 명령을 실행합니다. (유니코드)
SHIsFileAvailableOffline 파일 또는 폴더를 오프라인으로 사용할 수 있는지 여부를 결정합니다. 또한 이 함수는 네트워크, 로컬 오프라인 파일 캐시 또는 두 위치 모두에서 파일을 열지 여부를 결정합니다.
SHLoadNonloadedIconOverlayIdentifiers 오버레이 정보가 필요한 다음 작업 중에 생성에 실패했거나 시작 시 생성할 수 없는 아이콘 오버레이 식별자를 로드해야 한다는 신호를 Shell에 알릴 수 있습니다. 이미 로드된 식별자는 영향을 받지 않습니다.
SHQueryRecycleBinA 지정된 드라이브에 대한 휴지통의 크기와 그 안에 있는 항목 수를 검색합니다. (ANSI)

SHQueryRecycleBinW 지정된 드라이브에 대한 휴지통의 크기와 그 안에 있는 항목 수를 검색합니다. (유니코드)
SHQueryUserNotificationState 현재 사용자의 컴퓨터 상태를 확인하여 알림 보내기가 적절한지 여부를 확인합니다.
SHRemoveLocalizedName Shell 폴더에 있는 파일의 지역화된 이름을 제거합니다.
SHSetLocalizedName Shell 폴더에 있는 파일의 지역화된 이름을 설정합니다.
SHSetUnreadMailCountA 레지스트리에 지정된 전자 메일 계정에 대한 현재 사용자의 읽지 않은 메시지 수를 저장합니다. (ANSI)
SHSetUnreadMailCountW 레지스트리에 지정된 전자 메일 계정에 대한 현재 사용자의 읽지 않은 메시지 수를 저장합니다. (유니코드)
SHTestTokenMembership CheckTokenMembership을 사용하여 지정된 토큰이 지정된 RID가 있는 로컬 그룹의 구성원인지 여부를 테스트합니다.

구조체

APPBARDATA 시스템 앱바 메시지에 대한 정보를 포함합니다.
ASSOCIATIONELEMENT AssocCreateForClasses에서 지정된 파일 연결에 대한 IQueryAssociations 인터페이스를 검색하는 데 사용하는 정보를 정의합니다.
NC_ADDRESS 네트워크 주소를 설명하는 정보를 포함합니다.

NOTIFYICONDATAA 시스템에서 알림 영역에 알림을 표시해야 하는 정보를 포함합니다. Shell_NotifyIcon 사용합니다. (ANSI)
NOTIFYICONDATAW 시스템에서 알림 영역에 알림을 표시해야 하는 정보를 포함합니다. Shell_NotifyIcon 사용합니다. (유니코드)
NOTIFYICONIDENTIFIER 경계 사각형을 검색할 아이콘을 식별하기 위해 Shell_NotifyIconGetRect 사용하는 정보를 포함합니다.
OPEN_PRINTER_PROPS_INFOA 프린터의 속성 페이지에서 특정 속성 시트를 식별하고 해당 속성 시트가 모달이어야 하는지 여부를 식별합니다. 필요에 따라 SHInvokePrinterCommand 함수와 함께 사용됩니다. (ANSI)
OPEN_PRINTER_PROPS_INFOW 프린터의 속성 페이지에서 특정 속성 시트를 식별하고 해당 속성 시트가 모달이어야 하는지 여부를 식별합니다. 필요에 따라 SHInvokePrinterCommand 함수와 함께 사용됩니다. (유니코드)
SHCREATEPROCESSINFOW 프로세스를 만드는 데 SHCreateProcessAsUserW에 필요한 정보를 포함합니다.
SHELLEXECUTEINFOA ShellExecuteEx에서 사용하는 정보를 포함합니다. (ANSI)
SHELLEXECUTEINFOW ShellExecuteEx에서 사용하는 정보를 포함합니다. (유니코드)
SHFILEINFOA 파일 개체에 대한 정보를 포함합니다. (ANSI)
SHFILEINFOW 파일 개체에 대한 정보를 포함합니다. (유니코드)
SHFILEOPSTRUCTA SHFileOperation 함수가 파일 작업을 수행하는 데 사용하는 정보를 포함합니다. (ANSI)

SHFILEOPSTRUCTW SHFileOperation 함수가 파일 작업을 수행하는 데 사용하는 정보를 포함합니다. (유니코드)
SHNAMEMAPPINGA SHFileOperation 함수에서 이동, 복사 또는 이름을 바꾼 각 파일에 대한 이전 및 새 경로 이름을 포함합니다. (ANSI)
SHNAMEMAPPINGW SHFileOperation 함수에서 이동, 복사 또는 이름을 바꾼 각 파일에 대한 이전 및 새 경로 이름을 포함합니다. (유니코드)
SHQUERYRBINFO SHQueryRecycleBin 함수에서 검색한 크기 및 항목 개수 정보를 포함합니다.
슈스톡티코닌포 주식 셸 아이콘을 검색하는 데 사용되는 정보를 받습니다. 이 구조체는 SHGetStockIconInfo 호출에 사용됩니다.

열거형

QUERY_USER_NOTIFICATION_STATE 알림 보내기의 적정성과 관련하여 현재 사용자의 컴퓨터 상태를 지정합니다. SHQueryUserNotificationState에서 사용됩니다.
슈스톡ICONID SHGetStockIconInfo에서 검색할 스톡 시스템 아이콘을 식별하는 데 사용됩니다.

shellhandwriting.h 헤더

아티클2023. 11. 23.

먼저 포커스를 컨트롤로 설정하지 않고 펜을 켜거나 가까이에서 텍스트 편집 컨트롤로 수동 입력할 수 있습니다. 시스템은 의도를 결정하고, 입력 대상을 식별하고, 잉크 스트로크를 렌더링하고, 잉크를 텍스트(또는 텍스트 수정 제스처)로 인식하고, 사용 가능한 경우 텍스트 후보를 제안하고, 컨트롤의 편집 필드에 새 텍스트 또는 수정된 텍스트를 삽입합니다.

이 헤더는 잉크 입력에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 잉크 입력

TSF(Text Services Framework)와 UIA(UI 자동화)는 모두 ShellHandwriting 기능을 지원하는 데 사용됩니다.

다음 단계에서는 ShellHandwriting 기능에 사용되는 기본 프로세스를 설명합니다.

1. **옵트아웃 결정:** 애플리케이션이 ShellHandwriting을 지원하는지 확인합니다. 일반적으로 UIA(강력한 UI 자동화) 구현이 없는 애플리케이션 또는 사용자 지정 잉크 처리가 있는 애플리케이션입니다.
2. **의도 결정:** 앱이 ShellHandwriting을 지원하는 경우 시스템에서 펜 아래쪽 포인터 입력을 앱으로 전달해야 하는지 또는 필기용으로 사용해야 하는지 여부를 결정합니다.
3. **탭 결정:** 펜 입력을 필기용으로 사용해야 하는 경우 실행 가능한 컨트롤(예: 입력 우선 순위를 나타내는 단추)에서 입력을 받았는지 여부와 텍스트를 받을 수 있는 컨트롤(편집, ComboBox 또는 문서 형식)을 찾을 수 있는지, 읽기 전용이 아닌, 키보드 포커스를 허용할 수 있는지 테스트합니다.
4. **대상 결정:** 입력을 필기에 사용할 수 있는 경우 입력의 일부 주위에 경계 상자가 만들어지고 잉크 인식 결과(대상 결정)를 받는 데 가장 적합한 편집 컨트롤을 결정합니다.
5. **최종 결정:** 대상 편집 컨트롤을 식별하고 포커스를 지정하면 잉크 인식 결과가 컨트롤에 복사되기 전에 다양한 컨텍스트 표시기(예: 입력 scope, 언어, 기존 텍스트 경계 사각형)를 사용하여 적합성을 확인합니다.

shellhandwriting.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IHandwritingInputRoutingCallback 입력 대상의 UI 스레드에서 펜 입력 처리 시스템 스레드를 분리하기 위한 입력 라우팅 지원을 제공합니다.
ITfDetermineProximateHandwritingTargetArgs 필기 응답 동작 및 근접 대상 정보를 제공합니다.
ITfFocusHandwritingTargetArgs 필기 응답 동작 및 대상 정보를 제공합니다.
ITfHandwriting 먼저 포커스를 편집 컨트롤로 설정하지 않고도 근접 편집 필드에 펜으로 잉크 입력을 지원합니다. 잉크 스트로크가 렌더링되고, 필기 인식이 수행되고(텍스트 또는 편집 제스처로) 편집 필드에 텍스트가 입력됩니다.
ITfHandwritingRequest 애플리케이션은 이 인터페이스를 사용하여 필기 요청 후에 발생한 펜 입력을 평가했음을 시스템에 알려야 합니다.
ITfHandwritingSink 포커스가 편집 필드로 설정되기 전에 사용자 지정 필기 동작을 지정하는 기능을 지원합니다.

Functions

GetHandwritingStrokeIdForPointer 지정된 포인터 ID와 연결된 잉크 스트로크의 고유 ID를 검색합니다.
RegisterHandwritingInputRoutingCallback

열거형

TfHandwritingFocusTargetResponse

FocusHandwritingTarget 함수를 호출할 때 IHandwritingInputRoutingCallback 인터페이스를 구현하는 클라이언트가 응답하는 방법을 지정합니다.
TfHandwritingState 시스템에서 필기를 처리하는 방법을 지정합니다.
TfInputEvaluation 펜 입력을 인식하는 방법을 지정합니다.
TfProximateHandwritingTargetResponse 필기 대상 개체의 근접 위치에 따라 지원되는 필기 동작을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

shellscalingapi.h 헤더

아티클2023. 08. 26.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 높은 DPI
- Windows 셸

shellscalingapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

GetDpiForMonitor 디스플레이의 dpi(인치당 점)를 쿼리합니다.
GetDpiForShellUIComponent 현재 배율 계수 및 PROCESS_DPI_AWARENESS 따라 SHELL_UI_COMPONENT 차지하는 dpi(인치당 점)를 검색합니다.
GetProcessDpiAwareness 지정된 프로세스의 dpi(인치당 점) 인식을 검색합니다.
GetScaleFactorForDevice 디스플레이 디바이스의 기본 배율 인수를 가져옵니다.
GetScaleFactorForMonitor 특정 모니터의 배율 인수를 가져옵니다. 이 함수는 GetScaleFactorForDevice를 대체합니다.
RegisterScaleChangeEvent 눈금이 변경되었을 때 트리거되는 이벤트에 등록합니다. 이 함수는 RegisterScaleChangeNotifications를 대체합니다.
RegisterScaleChangeNotifications 정보 크기 조정이 변경될 때 콜백을 받을 창을 등록합니다.
RevokeScaleChangeNotifications 창의 등록을 취소하여 정보 크기 조정이 변경되면 콜백을 수신하지 못하게 합니다.

SetProcessDpiAwareness 프로세스 기본 DPI 인식 수준을 설정합니다. 이는 해당 DPI_AWARENESS_CONTEXT 값으로 SetProcessDpiAwarenessContext를 호출하는 것과 같습니다.
UnregisterScaleChangeEvent RegisterScaleChangeEvent를 통해 등록된 확장 변경 이벤트에 대한 등록을 취소합니다. 이 함수는 RevokeScaleChangeNotifications를 대체합니다.

열거형

DISPLAY_DEVICE_TYPE 디바이스가 기본 또는 몰입형 디스플레이 유형인지 여부를 나타냅니다.
MONITOR_DPI_TYPE 모니터의 dpi(인치당 점) 설정을 식별합니다.
PROCESS_DPI_AWARENESS dpi(인치당 점) 인식 값을 식별합니다. DPI 인식은 애플리케이션이 DPI에 대해 수행하는 크기 조정 작업과 시스템에서 수행하는 작업을 나타냅니다.
SCALE_CHANGE_FLAGS 발생한 크기 조정 변경을 나타내는 데 사용되는 플래그입니다.
SHELL_UI_COMPONENT 셸에 필요한 UI 구성 요소의 유형을 식별합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

shidfact.h 헤더

아티클2023. 08. 26.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

shidfact.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

CItemIDFactory Shell 데이터 원본과 상호 작용하기 위한 메서드를 노출합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

shimgdata.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

shimgdata.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IShellImageData 이미지 데이터를 표시, 조작 및 설명하는 메서드 및 속성을 노출합니다.
IShellImageDataAbort IShellImageData 프로세스를 중단하는 데 사용되는 단일 메서드를 노출합니다.
IShellImageDataFactory 다양한 이미지 원본을 기반으로 IShellImageData 인스턴스를 만드는 메서드를 노출합니다.

shldisp.h 헤더

아티클2023. 08. 26.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 레거시 Windows 환경 기능
- Windows 셸
- Windows 사이드바

shldisp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAutoComplete 자동 완성 개체(CLSID_AutoComplete)에 의해 노출됩니다. 이 인터페이스를 사용하면 애플리케이션에서 개체를 초기화, 사용 및 사용하지 않도록 설정할 수 있습니다.
IAutoComplete2 IAutoComplete를 확장합니다. 이 인터페이스를 사용하면 자동 완성 개체의 클라이언트가 자동 완성 작동 방식을 제어하는 여러 옵션을 검색하고 설정할 수 있습니다.
IDataObjectAsyncCapability 일반적으로 동기식 인터페이스가 비동기적으로 작동하도록 설정합니다.
IShellFolderViewDual 보기를 수정하고 현재 폴더의 항목을 선택하는 메서드를 노출합니다. (IShellFolderViewDual)
IShellFolderViewDual2 보기를 수정하고 현재 폴더의 항목을 선택하는 메서드를 노출합니다. (IShellFolderViewDual2)
IShellFolderViewDual3 현재 폴더 뷰를 수정하는 메서드를 노출합니다.

열거형

AUTOCOMPLETEOPTIONS 자동 완성을 둘러싼 옵션에 대해 IAutoComplete2::GetOptions 및 IAutoComplete2::SetOptions에서 사용하는 값을 지정합니다.
ShellFolderViewOptions ViewOptions 속성에서 반환되는 보기 옵션을 지정합니다.
ShellSpecialFolderConstants 특수 폴더를 식별하는 고유한 시스템 독립적 값을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

shlobj.h 헤더

아티클2023. 08. 26.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 레거시 Windows 환경 기능
- Windows 셸
- Windows 컨트롤

shlobj.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IActiveDesktopP Windows 데스크톱을 관리하는 메서드를 노출합니다.
IADesktopP2 Windows 데스크톱을 관리하는 메서드를 제공합니다.
IColumnProvider Windows Explorer 세부 정보 보기에서 사용자 지정 열을 추가하도록 설정하는 메서드를 노출합니다.
ICopyHookA 복사 후크 처리기를 만드는 메서드를 노출합니다. (ANSI)
ICopyHookW 복사 후크 처리기를 만드는 메서드를 노출합니다. (유니코드)
ICurrentWorkingDirectory 클라이언트가 개체의 현재 작업 디렉터리를 검색하거나 설정할 수 있도록 하는 메서드를 노출합니다.
IDefViewFrame IUnknown 기능에만 사용됩니다. 그것은 자신의 방법이 없습니다.
IDockingWindowFrame

프레임에 IDockingWindow 개체 추가를 지원하는 메서드를 노출합니다. 브라우저에서 구현됩니다.
IDocViewSite IShellView 인터페이스에서 사이트 개체로 사용됩니다.
IFileViewerA 등록된 파일 뷰어가 파일을 표시하거나 인쇄해야 할 때 알림을 받을 수 있는 인터페이스를 지정하는 메서드를 노출합니다. (ANSI)
IFileViewerSite 파일 뷰어가 현재 고정된 창에 대한 핸들을 검색하거나 새 고정 창을 설정할 수 있는 인터페이스를 지정하는 메서드를 노출합니다.
IFileViewerW 등록된 파일 뷰어가 파일을 표시하거나 인쇄해야 할 때 알림을 받을 수 있는 인터페이스를 지정하는 메서드를 노출합니다. (유니코드)
INewShortcutHookA 메서드를 노출하여 새 인터넷 바로 가기를 만듭니다. (ANSI)
INewShortcutHookW 메서드를 노출하여 새 인터넷 바로 가기를 만듭니다. (유니코드)
IShellFolderBand IShellFolderBand를 변경하거나 사용할 수 없습니다.
IShellImageStore 더 이상 사용되지 않습니다. 이미지 캐시를 조작하는 메서드를 노출합니다.
IThumbnailCapture HTML 배경 화면의 썸네일 표현을 가져오는 메서드를 노출합니다.

Functions

DriveType

DriveType 함수는 드라이브 번호에 따라 드라이브 유형을 결정합니다. (DriveType 함수)
GetFileNameFromBrowse GetFileNameFromBrowse 함수는 사용자가 열 파일의 드라이브, 디렉터리 및 이름을 지정할 수 있도록 열기 대화 상자를 만듭니다.
PathIsSlowA PathIsSlow를 변경하거나 사용할 수 없습니다. (ANSI)
PathIsSlowW PathIsSlow를 변경하거나 사용할 수 없습니다. (유니코드)
PathProcessCommand 더 이상 사용되지 않습니다. 명령줄이 포함된 문자열을 처리하고 필요한 경우 인수가 연결된 적절한 따옴표가 붙은 문자열을 생성합니다.
PerUserInit 내 문서 및 기타 특수 폴더를 만들고, 필요에 따라 초기화하고, 내 문서에 대한 바로 가기로 보내기 메뉴 항목을 만듭니다.
SHChangeNotifyRegisterThread 스레드의 비동기 레지스터 및 등록 취소를 사용하도록 설정합니다.
SHCreatePropSheetExtArray SHCreatePropSheetExtArray를 변경하거나 사용할 수 없습니다.
SHCreateQueryCancelAutoPlayMoniker 더 이상 사용되지 않습니다. QueryCancelAutoPlay 클래스 모니커를 만든 다음, ROT(실행 중인 개체 테이블)에 IQueryCancelAutoPlay 처리기를 등록하는 데 사용할 수 있습니다.
SHLoadOLE 더 이상 사용되지 않습니다. 이전 버전과의 호환성을 위해 제공됩니다.
SHMultiFileProperties 파일 집합에 대한 병합된 속성 시트를 표시합니다. 모든 파일에 공통적인 속성 값이 표시되고 다른 속성 값은 문자열(여러 값)을 표시합니다.
SHOpenPropSheetA

SHOpenPropSheet를 변경하거나 사용할 수 없습니다. (ANSI)
SHOpenPropSheetW SHOpenPropSheet를 변경하거나 사용할 수 없습니다. (유니코드)
SHRunControlPanel 제어판 항목을 엽니다.
SoftwareUpdateMessageBox 애플리케이션이 업데이트되었음을 사용자에게 알리는 데 사용할 수 있는 표준 메시지 상자를 표시합니다.

구조체

AASHELLMENUFILENAME 메뉴 파일 이름에 대한 정보를 포함하는 가변 크기 구조체입니다.
AASHELLMENUITEM 메뉴 항목에 대한 정보를 포함합니다. (AASHELLMENUITEM)
BANDINFOSFB 폴더 대역에 대한 정보를 포함합니다. 이 구조체는 IShellFolderBand::GetBandInfoSFB 및 IShellFolderBand::SetBandInfoSFB 메서드와 함께 사용됩니다.
FVSHOWINFO 파일 뷰어가 파일을 표시하는 데 사용하는 정보를 포함합니다.
SFV_SETITEMPOS 항목의 위치 정보를 저장합니다. 메시지 SFVM_SETITEMPOS 함께 사용됩니다.
SHChangeProductKeyAsIDList SHChangeProductKeyAsIDList 구조체
SHCOLUMNDATA

특정 파일을 식별하는 정보를 포함합니다. 특정 파일에 대한 데이터를 요청할 때 IColumnProvider::GetItemData에서 사용됩니다.
SHCOLUMNINFO 열의 속성에 대한 정보를 포함합니다. IColumnProvider::GetColumnInfo에서 사용됩니다.
SHCOLUMNINIT IColumnProvider::Initialize에 초기화 정보를 전달합니다.
TBINFO SFVM_GETBUTTONINFO 알림과 함께 도구 모음에 추가할 단추 수와 추가 방법을 지정하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

shlobj_core.h 헤더

아티클2023. 03. 16.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Active Directory Domain Services
- 레거시 Windows 환경 기능
- Windows 셸
- Windows 속성 시스템

shlobj_core.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IACList 후보 문자열이 계층 구조로 구성될 때 자동 완성의 효율성을 향상시키는 메서드를 노출합니다.
IACList2 자동 완성 개체의 클라이언트가 옵션 플래그를 검색하고 설정할 수 있도록 IACList 인터페이스를 확장합니다.
IActiveDesktop 클라이언트 프로그램에서 로컬 컴퓨터의 데스크톱 항목과 배경 화면을 관리할 수 있습니다.
IDockingWindowSite 하나 이상의 IDockingWindow 개체에 대한 테두리 공간을 관리하는 메서드를 노출합니다. 이 인터페이스는 브라우저에서 구현되며 IOleInPlaceUIWindow 인터페이스와 유사합니다.
IExtractIconA 클라이언트가 폴더의 개체 중 하나와 연결된 아이콘을 검색할 수 있도록 하는 메서드를 노출합니다. (ANSI)
IExtractIconW 클라이언트가 폴더의 개체 중 하나와 연결된 아이콘을 검색할 수 있도록 하는 메서드를 노출합니다. (유니코드)

INamedPropertyBag 개체가 해당 속성을 저장할 수 있는 지정된 속성 모음을 개체에 제공하는 메서드를 노출합니다.
IObjMgr 클라이언트가 서버 개체에서 관리하는 개체의 컬렉션에서 개체를 추가하거나 제거할 수 있도록 하는 메서드를 노출합니다.
IProgressDialog 애플리케이션에서 진행률 대화 상자를 표시할 수 있는 옵션을 제공하는 메서드를 노출합니다.
IQueryInfo 셸이 IShellFolder 구현에 있는 항목에 대한 플래그 및 정보 팁 정보를 검색하는 데 사용하는 메서드를 노출합니다. 정보 팁은 일반적으로 도구 설명 컨트롤 내에 표시됩니다.
ISearchContext 사용자 지정 정보를 검색 후크에 채널하는 메서드를 노출합니다.
IShellChangeNotify 항목의 ID가 변경되면 셸 네임스페이스 확장을 알리는 메서드를 노출합니다.
IShellDetails 폴더의 항목에 대한 자세한 정보를 제공하기 위해 Shell 폴더에 의해 노출됩니다.
IShellFolderView 셸 폴더 뷰를 조작하는 메서드를 노출합니다.
IShellFolderViewCB 폴더 뷰에 이벤트를 알리고 그에 따라 보기를 수정할 수 있도록 시스템 폴더 뷰 개체(SHCreateShellFolderView를 통해 반환된 IShellView 개체)를 사용하여 구현된 폴더 보기와 Windows 탐색기 간의 통신을 허용하는 메서드를 노출합니다.
IShellIconOverlay 네임스페이스 확장에서 포함하는 개체에 대한 아이콘 오버레이를 지정하는 데 사용되는 메서드를 노출합니다.
IShellIconOverlayManager IShellIconOverlayManager를 변경하거나 사용할 수 없습니다.

IURLSearchHook 브라우저에서 알 수 없는 URL 프로토콜의 주소를 변환하는 데 사용되는 메서드를 노출합니다.
IURLSearchHook2 검색 컨텍스트 개체를 사용하여 알 수 없는 URL 프로토콜의 주소를 변환하기 위해 브라우저에서 사용하는 메서드를 노출합니다.

Functions

AssocGetDetailsOfPropKey 네임스페이스 확장에서 제공하는 파일 연결 정보를 사용하여 지정된 속성 키의 값을 검색합니다.
CDefFolderMenu_Create2 선택한 파일 폴더 개체 그룹에 대한 상황에 맞는 메뉴를 만듭니다.
CIDLData_CreateFromIDArray CIDLData_CreateFromIDArray 변경되거나 사용할 수 없습니다.
DAD_AutoScroll 이미지를 끌 때 창을 스크롤합니다.
DAD_DragEnterEx 끌어서 놓기 작업 중에 지정된 창의 업데이트를 잠그고 창 내의 지정된 위치에 끌기 이미지를 표시합니다. (DAD_DragEnterEx)
DAD_DragEnterEx2 끌어서 놓기 작업 중에 지정된 창에 업데이트를 잠그고 창 내의 지정된 위치에 끌어서 놓기 이미지를 표시합니다.
DAD_DragLeave DAD_DragEnterEx 함수에 의해 잠긴 창의 잠금을 해제합니다.
DAD_DragMove 끌어서 놓기 작업 중에 끌기 이미지를 이동합니다.

DAD_SetDragImage 끌기 이미지를 설정합니다.
DAD_ShowDragImage 끌기 이미지를 표시하거나 숨깁니다. (DAD_ShowDragImage)
DriveType DriveType 함수는 드라이브 번호에 따라 드라이브 유형을 결정합니다. (DriveType)
GetFileNameFromBrowse GetFileNameFromBrowse 함수는 사용자가 열 파일의 드라이브, 디렉터리 및 이름을 지정할 수 있도록 열기 대화 상자를 만듭니다. (GetFileNameFromBrowse)
IDListContainerIsConsistent IDList의 컨테이너 구조가 유효한지 확인합니다.
ILAppendID ITEMIDLIST 구조체에 SHITEMID 구조체를 추가하거나 앞에 추가합니다.
ILClone ITEMIDLIST 구조를 복제합니다.
ILCloneChild 자식 ITEMIDLIST 구조를 복제합니다.
ILCloneFirst ITEMIDLIST 구조체의 첫 번째 SHITEMID 구조를 복제합니다.
ILCloneFull 전체 또는 절대 ITEMIDLIST 구조를 복제합니다.
ILCombine 두 ITEMIDLIST 구조를 결합합니다.
ILCreateFromPath ILCreateFromPath 함수는 지정된 파일 경로와 연결된 ITEMIDLIST 구조를 반환합니다.

ILCreateFromPathA 지정된 파일 경로와 연결된 ITEMIDLIST 구조를 반환합니다. (ILCreateFromPathA)
ILCreateFromPathW ILCreateFromPathW(유니코드) 함수는 지정된 파일 경로와 연결된 ITEMIDLIST 구조를 반환합니다.
ILFindChild 지정된 ITEMIDLIST 구조체가 다른 ITEMIDLIST 구조체의 자식인지 여부를 확인합니다.
ILFindLastID ITEMIDLIST 구조체의 마지막 SHITEMID 구조체에 대한 포인터를 반환합니다.
ILFree 셸에서 할당한 ITEMIDLIST 구조를 해제합니다.
ILGetNext ITEMIDLIST 구조체에서 다음 SHITEMID 구조를 검색합니다. (ILGetNext)
ILGetSize ITEMIDLIST 구조체의 크기(바이트)를 반환합니다.
ILIsAligned 32비트 아키텍처의 DWORD 및 64비트 아키텍처의 QWORD인 포인터 경계에 상수 ITEMIDLIST가 정렬되어 있는지 확인합니다.
ILIsChild PIDL(항목 식별자 목록)에 대한 포인터가 정확히 하나의 SHITEMID가 있는 PIDL인 자식 PIDL인지 확인합니다.
ILIsEmpty ITEMIDLIST 구조체가 비어 있는지 확인합니다.
ILIsEqual 두 ITEMIDLIST 구조체가 이진 비교에서 같은지 테스트합니다.
ILIsParent ITEMIDLIST 구조체가 다른 ITEMIDLIST 구조체의 부모인지 여부를 테스트합니다.

ILLoadFromStream 더 이상 사용되지 않습니다. 스트림에서 ITEMIDLIST 구조를 로드합니다.
ILLoadFromStreamEx 이 함수는 변경되거나 사용할 수 없습니다.
ILLoadFromStreamEx ILLoadFromStreamEx 함수는 IStream에서 ITEMIDLIST를 로드합니다. (ILLoadFromStreamEx r1)
ILLoadFromStreamEx ILLoadFromStreamEx 함수는 IStream에서 PIDL(항목 식별자 목록)에 대한 자식 포인터를 로드합니다. (ILLoadFromStreamEx r2)
ILNext ITEMIDLIST 구조체에서 다음 SHITEMID 구조를 검색합니다. (ILNext)
ILRemoveLastID ITEMIDLIST 구조체에서 마지막 SHITEMID 구조를 제거합니다.
ILSaveToStream ITEMIDLIST 구조를 스트림에 저장합니다.
ILSkip 정렬되지 않은 상수의 상대 ITEMIDLIST 구조에서 지정된 바이트 수를 건너뜁니다.
IsNetDrive 드라이브가 네트워크 드라이브인지 테스트합니다.
IsUserAnAdmin IsUserAnAdmin을 변경하거나 사용할 수 없습니다.
OpenRegStream OpenRegStream을 변경하거나 사용할 수 없습니다. 대신 SHOpenRegStream2 또는 SHOpenRegStream을 사용합니다.
PathCleanupSpec PathCleanupSpec을 변경하거나 사용할 수 없습니다.

PathGetShortPath PathGetShortPath를 변경하거나 사용할 수 없습니다.
PathIsExe PathIsExe를 변경하거나 사용할 수 없습니다.
PathMakeUniqueName 템플릿에서 고유한 경로 이름을 만듭니다.
PathResolve PathResolve를 변경하거나 사용할 수 없습니다.
PathYetAnotherMakeUniqueName 기존 파일 이름을 기반으로 고유한 파일 이름을 만듭니다.
PickIconDlg PickIconDlg를 변경하거나 사용할 수 없습니다.
PifMgr_CloseProperties PifMgr_OpenProperties 사용하여 연 애플리케이션 속성을 닫습니다.
PifMgr_GetProperties .pif 파일에서 지정된 데이터 블록을 반환합니다.
PifMgr_OpenProperties Microsoft MS-DOS 애플리케이션과 연결된 .pif 파일을 열고 애플리케이션의 속성에 대한 핸들을 반환합니다.
PifMgr_SetProperties .pif 파일의 데이터 블록에 값을 할당합니다.
ReadCabinetState ReadCabinetState를 변경하거나 사용할 수 없습니다.
RealDriveType RealDriveType을 변경하거나 사용할 수 없습니다.

RestartDialog 사용자에게 Windows를 다시 시작하라는 메시지를 표시하는 대화 상자를 표시합니다. 사용자가 단추를 클릭하면 함수는 ExitWindowsEx를 호출하여 Windows를 다시 시작하려고 시도합니다.
RestartDialogEx 사용자에게 Windows를 다시 시작하도록 요청하는 대화 상자를 표시합니다. 사용자가 단추를 클릭하면 함수는 ExitWindowsEx를 호출하여 Windows를 다시 시작하려고 시도합니다.
SHAddFromPropSheetExtArray SHCreatePropSheetExtArray에서 만든 속성 시트 확장 배열에 페이지를 추가합니다.
SHAddToRecentDocs 가장 최근에 가장 자주 사용되는 항목을 추적하기 위해 항목에 액세스했음을 시스템에 알립니다. 이 함수를 사용하여 모든 사용량 현황 데이터를 지울 수도 있습니다.
SHAlloc 셸의 힙에서 메모리를 할당합니다.
SHBindToFolderIDListParent 폴더 형식으로 지정된 셸 네임스페이스 항목과 해당 폴더를 기준으로 하는 항목 식별자 목록이 있는 경우 이 함수는 네임스페이스 항목의 부모에 바인딩하고 선택적으로 항목 식별자 목록의 최종 구성 요소에 대한 포인터를 반환합니다.
SHBindToFolderIDListParentEx 호출자가 바인딩 컨텍스트를 지정할 수 있도록 하여 SHBindToFolderIDListParent 함수를 확장합니다.
SHBindToObject 셸 네임스페이스 IShellFolder::BindToObject 메서드를 사용하여 지정된 개체를 검색하고 바인딩합니다.
SHBindToParent PIDL(정규화된 항목 식별자 목록)에 대한 포인터를 사용하고 부모 개체에 대해 지정된 인터페이스 포인터를 반환합니다.
SHBrowseForFolderA 사용자가 Shell 폴더를 선택할 수 있는 대화 상자를 표시합니다. (ANSI)

SHBrowseForFolderW 사용자가 Shell 폴더를 선택할 수 있는 대화 상자를 표시합니다. (유니코드)
SHChangeNotification_Lock 셸 변경 알림 이벤트와 연결된 공유 메모리를 잠가 옵니다.
SHChangeNotification_Unlock 변경 알림에 대한 공유 메모리 잠금을 해제합니다.
SHChangeNotify 애플리케이션이 수행한 이벤트의 시스템에 알 수 있습니다. 애플리케이션은 셸에 영향을 줄 수 있는 작업을 수행하는 경우 이 함수를 사용해야 합니다.
SHChangeNotifyDeregister SHChangeNotify 메시지를 수신하지 못하도록 클라이언트의 창 프로세스를 등록 취소합니다.
SHChangeNotifyRegister 파일 시스템에서 알림을 지원하는 경우 파일 시스템 또는 Shell에서 알림을 받을 창을 등록합니다.
SHCloneSpecialIDList SHCloneSpecialIDList를 변경하거나 사용할 수 없습니다. 대신 SHGetSpecialFolderLocation을 사용합니다.
SHCLSIDFromString CLSID(클래스 식별자)의 문자열 형식을 사용하여 해당 CLSID를 만듭니다.
SHCoCreateInstance SHCoCreateInstance를 변경하거나 사용할 수 없습니다. 대신 CoCreateInstance를 사용합니다.
SHCreateDataObject 부모 폴더에 데이터 개체를 만듭니다.
SHCreateDefaultContextMenu Shell의 기본 상황에 맞는 메뉴 구현을 나타내는 개체를 만듭니다.
SHCreateDirectory 새 파일 시스템 폴더를 만듭니다.

SHCreateDirectoryExA 선택적 보안 특성을 사용하여 새 파일 시스템 폴더를 만듭니다. (ANSI)
SHCreateDirectoryExW 선택적 보안 특성을 사용하여 새 파일 시스템 폴더를 만듭니다. (유니코드)
SHCreateFileExtractIconA SHCreateFileExtractIcon을 변경하거나 사용할 수 없습니다. (ANSI)
SHCreateFileExtractIconW SHCreateFileExtractIcon을 변경하거나 사용할 수 없습니다. (유니코드)
SHCreateShellFolderView 기본 Shell 폴더 뷰 개체(DefView)의 새 인스턴스를 만듭니다.
SHCreateShellFolderViewEx 기본 Shell 폴더 뷰 개체의 새 인스턴스를 만듭니다. 이 함수 대신 SHCreateShellFolderView를 사용하는 것이 좋습니다.
SHCreateShellItem IShellItem 개체를 만듭니다.
SHCreateStdEnumFmtEtc SHCreateStdEnumFmtEtc를 변경하거나 사용할 수 없습니다.
SHDefExtractIconA 파일에서 아이콘을 추출하는 기본 처리기를 제공합니다. (ANSI)
SHDefExtractIconW 파일에서 아이콘을 추출하는 기본 처리기를 제공합니다. (유니코드)
SHDestroyPropSheetExtArray SHCreatePropSheetExtArray에서 만든 배열을 가리키는 속성 시트 처리기를 해제합니다.
SHDoDragDrop 끌어서 놓기 작업을 실행합니다. 요청 시 원본 끌어서 만들기와 끌어서 이미지를 지원합니다.

Shell_GetCachedImageIndex Shell_GetCachedImageIndex 변경되거나 사용할 수 없습니다.
Shell_GetCachedImageIndexA Shell_GetCachedImageIndex 변경되거나 사용할 수 없습니다. (ANSI)
Shell_GetCachedImageIndexW Shell_GetCachedImageIndex 변경되거나 사용할 수 없습니다. (유니코드)
Shell_GetImageLists 크고 작은 아이콘에 대한 시스템 이미지 목록을 검색합니다.
Shell_MergeMenus Shell_MergeMenus 변경되거나 사용할 수 없습니다.
SHFind_InitMenuPopup SHFind_InitMenuPopup 변경되거나 사용할 수 없습니다.
SHFindFiles SHFindFiles를 변경하거나 사용할 수 없습니다.
SHFlushSFCache SHFlushSFCache를 변경하거나 사용할 수 없습니다.
SHFormatDrive SHFormatDrive를 변경하거나 사용할 수 없습니다.
SHFree SHAlloc에서 할당한 메모리를 해제합니다.
SHGetAttributesFromDataObject SHGetAttributesFromDataObject를 변경하거나 사용할 수 없습니다.
SHGetDataFromIDListA 상대 식별자 목록에서 확장 속성 데이터를 검색합니다. (ANSI)

SHGetDataFromIDListW 상대 식별자 목록에서 확장 속성 데이터를 검색합니다. (유니코드)
SHGetDesktopFolder 셸 네임스페이스의 루트인 데스크톱 폴더에 대한 IShellFolder 인터페이스를 검색합니다.
SHGetFolderLocation 더 이상 사용되지 않습니다. 폴더의 경로를 ITEMIDLIST 구조체로 검색합니다.
SHGetFolderPathA 더 이상 사용되지 않습니다. (SHGetFolderPathA)
SHGetFolderPathAndSubDirA 폴더의 경로를 가져오고 사용자가 제공한 하위 폴더 경로를 추가합니다. (ANSI)
SHGetFolderPathAndSubDirW 폴더의 경로를 가져오고 사용자가 제공한 하위 폴더 경로를 추가합니다. (유니코드)
SHGetFolderPathW 더 이상 사용되지 않습니다. (SHGetFolderPathW)
SHGetIconOverlayIndexA 시스템 이미지 목록에서 오버레이 아이콘의 인덱스 를 반환합니다. (ANSI)
SHGetIconOverlayIndexW 시스템 이미지 목록에서 오버레이 아이콘의 인덱스 를 반환합니다. (유니코드)
SHGetInstanceExplorer 호스트된 셸 확장 및 기타 구성 요소가 호스트 프로세스가 조기에 닫히지 않도록 하는 인터페이스를 검색합니다.
SHGetKnownFolderIDList 알려진 폴더의 경로를 ITEMIDLIST 구조체로 검색합니다.
SHGetKnownFolderItem 알려진 폴더를 나타내는 IShellItem 개체를 검색합니다.

SHGetKnownFolderPath 폴더의 KNOWNFOLDERID로 식별된 알려진 폴더의 전체 경로를 검색합니다.
SHGetMalloc 셸의 IMalloc 인터페이스에 대한 포인터를 검색합니다.
SHGetPathFromIDListA 항목 식별자 목록을 파일 시스템 경로로 변환합니다. (ANSI)
SHGetPathFromIDListEx 항목 식별자 목록을 파일 시스템 경로로 변환합니다. 이 함수는 문자열 버퍼의 초기 크기를 설정하고 아래 옵션을 선언할 수 있도록 하여 SHGetPathFromIDList를 확장합니다.
SHGetPathFromIDListW 항목 식별자 목록을 파일 시스템 경로로 변환합니다. (유니코드)
SHGetRealIDL SHGetRealIDL을 변경하거나 사용할 수 없습니다.
SHGetSetFolderCustomSettings SHGetSetFolderCustomSettings를 변경하거나 사용할 수 없습니다.
SHGetSetSettings SHGetSetSettings를 변경하거나 사용할 수 없습니다.
SHGetSettings 현재 셸 옵션 설정을 검색합니다.
SHGetSpecialFolderLocation SHGetSpecialFolderLocation은 지원되지 않으며 나중에 변경되거나 사용할 수 없습니다. 대신 SHGetFolderLocation을 사용합니다.
SHGetSpecialFolderPathA SHGetSpecialFolderPath는 지원되지 않습니다. 대신 ShGetFolderPath를 사용합니다. (ANSI)
SHGetSpecialFolderPathW SHGetSpecialFolderPath는 지원되지 않습니다. 대신 ShGetFolderPath를 사용합니다. (유니코드)

SHHandleUpdateImage SHHandleUpdateImage를 변경하거나 사용할 수 없습니다.
SHILCreateFromPath SHILCreateFromPath를 변경하거나 사용할 수 없습니다.
SHLimitInputEdit 편집 컨트롤의 유효한 문자에 대한 제한을 설정합니다.
SHLoadInProc Shell 프로세스의 컨텍스트 내에서 지정된 개체 클래스의 인스턴스를 만듭니다. Windows Vista 이상:_This 함수가 비활성화되었으며 E_NOTIMPL 반환합니다.
SHMapPIDLToSystemImageListIndex SHMapPIDLToSystemImageListIndex를 변경하거나 사용할 수 없습니다.
SHObjectProperties SHObjectProperties를 변경하거나 사용할 수 없습니다.
SHOpenFolderAndSelectItems 특정 폴더에 지정된 항목이 선택된 Windows 탐색기 창을 엽니다.
SHOpenWithDialog 다음으로 열기 대화 상자를 표시합니다.
SHParseDisplayName Shell 네임스페이스 개체의 표시 이름을 항목 식별자 목록으로 변환하고 개체의 특성을 반환합니다. 이 함수는 문자열을 PIDL(항목 식별자 목록)에 대한 포인터로 변환하는 기본 메서드입니다.
SHPathPrepareForWriteA 경로가 있는지 확인합니다. (ANSI)
SHPathPrepareForWriteW 경로가 있는지 확인합니다. (유니코드)
SHPropStgCreate 요청된 속성 집합 작업에 대한 코드 페이지 검색 또는 할당의 적절한 처리를 보장합니다.

SHPropStgReadMultiple 더 이상 사용되지 않는 속성 집합에 대해 ANSI 및 유니코드 변환이 제대로 처리되도록 IPropertyStorage::ReadMultiple 함수를 래핑합니다.
SHPropStgWriteMultiple 더 이상 사용되지 않는 속성 집합에 대해 ANSI 및 유니코드 변환이 제대로 처리되도록 IPropertyStorage::WriteMultiple 함수를 래핑합니다.
SHReplaceFromPropSheetExtArray 페이지를 바꾸기 위해 속성 시트 확장 배열의 각 속성 시트를 요청합니다. 각 페이지는 최대 하나의 대체가 허용됩니다.
SHRestricted SHRestricted는 변경되거나 사용할 수 없습니다.
SHSetFolderPathA 더 이상 사용되지 않습니다. CSIDL로 식별된 시스템 폴더에 새 경로를 할당합니다. (ANSI)
SHSetFolderPathW 더 이상 사용되지 않습니다. CSIDL로 식별된 시스템 폴더에 새 경로를 할당합니다. (유니코드)
SHSetInstanceExplorer 호스트된 셸 확장 및 기타 구성 요소가 호스트 프로세스가 조기에 닫히지 않도록 하는 인터페이스를 제공합니다.
SHSetKnownFolderPath 알려진 폴더를 새 위치로 리디렉션합니다.
SHShellFolderView_Message SHShellFolderView_Message 변경되거나 사용할 수 없습니다.
SHStartNetConnectionDialogA SHStartNetConnectionDialog를 변경하거나 사용할 수 없습니다. (ANSI)
SHStartNetConnectionDialogW SHStartNetConnectionDialog를 변경하거나 사용할 수 없습니다. (유니코드)

SHUpdateImageA 셸에 시스템 이미지 목록의 이미지가 변경되었음을 알 수 있습니다. (ANSI)
SHUpdateImageW 셸에 시스템 이미지 목록의 이미지가 변경되었음을 알 수 있습니다. (유니코드)
SHValidateUNC SHValidateUNC를 변경하거나 사용할 수 없습니다.
SignalFileOpen SignalFileOpen을 변경하거나 사용할 수 없습니다.
StgMakeUniqueName 템플릿에서 스트림 또는 스토리지 개체의 고유한 이름을 만듭니다.
Win32DeleteFile Win32DeleteFile을 변경하거나 사용할 수 없습니다.
WriteCabinetState WriteCabinetState를 변경하거나 사용할 수 없습니다.

콜백 함수

BFFCALLBACK Active Directory Domain Services 컨테이너 브라우저 대화 상자에서 이벤트 알림을 받습니다.
LPFNDFMCALLBACK LPFNDFMCALLBACK을 변경하거나 사용할 수 없습니다.
LPFNVIEWCALLBACK 시스템 폴더 뷰 개체에서 사용하는 콜백 함수의 프로토타입을 정의합니다. 이 함수는 기본적으로 IShellFolderViewCB의 기능을 복제합니다.

구조체

AUTO_SCROLL_DATA AUTO_SCROLL_DATA 변경되거나 사용할 수 없습니다.
BROWSEINFOA SHBrowseForFolder 함수에 대한 매개 변수를 포함하고 사용자가 선택한 폴더에 대한 정보를 받습니다. (ANSI)
BROWSEINFOW SHBrowseForFolder 함수에 대한 매개 변수를 포함하고 사용자가 선택한 폴더에 대한 정보를 받습니다. (유니코드)
캐비닛스테이트 CABINETSTATE는 변경되거나 사용할 수 없습니다.
CIDA 하나 이상의 Shell 네임스페이스 개체의 PIDL(항목 식별자 목록)에 포인터를 전송하기 위해 CFSTR_SHELLIDLIST 클립보드 형식과 함께 사용됩니다.
COMPONENT Windows 2000에서 구성 요소에 대한 정보를 보관하는 데 사용됩니다. 이 구조체는 IE4COMPONENT 구조를 대체합니다.
COMPONENTSOPT 바탕 화면 항목 옵션을 포함합니다.
COMPPOS 구성 요소의 위치 및 크기에 대한 정보를 보유합니다.
COMPSTATEINFO Windows 2000에서 구성 요소의 상태에 대한 정보를 보관하는 데 사용됩니다.
CSFV SHCreateShellFolderViewEx 함수와 함께 사용됩니다.

DATABLOCK_HEADER IShellLinkDataList에서 사용하는 일부 추가 데이터 구조의 헤더 역할을 합니다.
DEFCONTEXTMENU SHCreateDefaultContextMenu에서 사용하는 상황에 맞는 메뉴 정보를 포함합니다.
DETAILSINFO Shell 폴더 항목에 대한 세부 정보를 포함합니다. SFVM_GETDETAILSOF 알림과 함께 사용됩니다.
DFMICS DFM_INVOKECOMMANDEX 사용하는 추가 인수를 포함합니다.
DROPDESCRIPTION drop 개체에 대한 이미지 및 함께 제공되는 텍스트를 설명합니다.
DROPFILES CF_HDROP 클립보드 형식을 정의합니다. 다음 데이터는 파일 이름의 이중 null로 끝나는 목록입니다.
EXP_DARWIN_LINK IShellLinkDataList에서 사용하는 추가 데이터 블록을 보유합니다. 링크의 Windows Installer ID를 보유합니다.
EXP_PROPERTYSTORAGE 셸 링크 상태에 대한 정보를 저장합니다. 이 구조는 EXP_PROPERTYSTORAGE_SIG 태그가 지정된 추가 데이터 섹션에 사용됩니다.
EXP_SPECIAL_FOLDER IShellLinkDataList에서 사용하는 추가 데이터 블록을 보유합니다. 특수 폴더 정보를 보유합니다.
EXP_SZ_LINK IShellLinkDataList에서 사용하는 추가 데이터 블록을 보유합니다. 아이콘 또는 대상에 대해 확장 가능한 환경 문자열을 보유합니다.
FILE_ATTRIBUTES_ARRAY CFSTR_FILE_ATTRIBUTES_ARRAY 대한 클립보드 형식 정의를 포함합니다.

FILEDESCRIPTORA Microsoft ActiveX 끌어서 놓기 작업 중에 클립보드를 통해 복사되는 파일의 속성을 설명합니다. (ANSI)
FILEDESCRIPTORW Microsoft ActiveX 끌어서 놓기 작업 중에 클립보드를 통해 복사되는 파일의 속성을 설명합니다. (유니코드)
FILEGROUPDESCRIPTORA CF_FILEGROUPDESCRIPTOR 클립보드 형식을 정의합니다. (ANSI)
FILEGROUPDESCRIPTORW CF_FILEGROUPDESCRIPTOR 클립보드 형식을 정의합니다. (유니코드)
IE4COMPONENT Microsoft Internet Explorer 4.0 및 Microsoft Internet Explorer 4.01에서 구성 요소에 대한 정보를 보관하는 데 사용됩니다. Windows 2000에서는 구성 요소 구조로 대체됩니다.
ITEMSPACING display:_small 및 large에 사용할 수 있는 두 가지 가능한 아이콘 간격 크기의 크기를 저장합니다. IShellFolderView::GetItemSpacing에서 사용됩니다.
NRESARRAY CF_NETRESOURCE 클립보드 형식을 정의합니다.
NT_CONSOLE_PROPS IShellLinkDataList에서 사용하는 추가 데이터 블록을 보유합니다. 콘솔 속성을 보유합니다.
NT_FE_CONSOLE_PROPS IShellLinkDataList에서 사용하는 추가 데이터 블록을 보유합니다. 콘솔의 코드 페이지가 있습니다.
OPENASINFO SHOpenWithDialog 함수에 대한 정보를 저장합니다.
PROPPRG 이 구조체에는 .pif 파일의 정보가 포함됩니다. PifMgr_GetProperties 사용됩니다.

QCMINFO 메뉴 항목을 Windows 탐색기 메뉴로 병합하기 위한 정보를 포함합니다.
SFV_CREATE 이 구조체는 SHCreateShellFolderView 함수와 함께 사용됩니다.
SFVM_HELPTOPIC_DATA HTML 도움말 파일의 이름과 해당 파일의 토픽을 포함합니다. SFVM_GETHELPTOPIC 알림과 함께 사용됩니다. 이 구조에는 유니코드 문자열이 필요합니다.
SFVM_PROPPAGE_DATA 개체의 속성 시트에 추가할 페이지의 세부 정보를 포함합니다.
SHARDAPPIDINFO SHAddToRecentDocs에서 이 경우 항목을 IShellItem과 연결된 프로세스로 식별하는 데 사용하는 데이터를 포함합니다.
SHARDAPPIDINFOIDLIST SHAddToRecentDocs에서 PIDL(항목 식별자 목록)에 대한 절대 포인터와 연결된 프로세스를 통해 이 경우 항목을 식별하는 데 사용하는 데이터를 포함합니다.
SHARDAPPIDINFOLINK SHAddToRecentDocs에서 IShellLink를 통해 항목과 연결된 프로세스를 모두 식별하는 데 사용하는 데이터를 포함합니다.
SHChangeDWORDAsIDList SHChangeDWORDAsIDList를 변경하거나 사용할 수 없습니다.
SHChangeNotifyEntry 변경 알림에 대한 정보를 포함하고 받습니다. 이 구조체는 SHChangeNotifyRegister 함수 및 SFVM_QUERYFSNOTIFY 알림과 함께 사용됩니다.
SHChangeUpdateImageIDList SHChangeUpdateImageIDList를 변경하거나 사용할 수 없습니다.
SHDESCRIPTIONID SHGetDataFromIDList 호출에 대한 응답으로 항목 데이터를 받습니다.

SHELLFLAGSTATE 현재 셸 설정을 나타내는 플래그 집합을 포함합니다. 이 구조체는 SHGetSettings 함수와 함께 사용됩니다.
SHELLSTATEA 셸 상태에 대한 설정을 포함합니다. 이 구조체는 SHGetSetSettings 함수와 함께 사용됩니다. (ANSI)
SHELLSTATEW 셸 상태에 대한 설정을 포함합니다. 이 구조체는 SHGetSetSettings 함수와 함께 사용됩니다. (유니코드)
SHFOLDERCUSTOMSETTINGS 사용자 지정 폴더 설정을 보유합니다. 이 구조체는 SHGetSetFolderCustomSettings 함수와 함께 사용됩니다.
배경 화면옵트 배경 화면 표시 옵션이 포함되어 있습니다. IActiveDesktop 인터페이스의 멤버와 함께 사용됩니다.

열거형

AUTOCOMPLETELISTOPTIONS 자동 완성 목록에 대해 열거되는 개체를 지정합니다.
DROPIMAGETYPE DROPDESCRIPTION 구조와 함께 드롭 이미지를 지정하는 데 사용되는 값입니다.
IESHORTCUTFLAGS 브라우저에서 바로 가기를 처리하는 방법을 지정합니다.
KNOWN_FOLDER_FLAG 알려진 폴더에 대한 특수 검색 옵션을 지정하는 상수를 정의합니다. 이러한 값은 병렬 의미가 있는 CSIDL 값을 대체합니다.
제한 이러한 플래그는 SHRestricted 함수와 함께 사용됩니다.

SCNRT_STATUS SHChangeNotifyRegisterThread에 대해 비동기 레지스터 및 등록 취소를 사용하거나 사용하지 않도록 설정할지 여부를 나타냅니다.
파편 사용 통계가 추적되는 항목을 식별하기 위해 pv 매개 변수에서 SHAddToRecentDocs에서 전달한 데이터의 해석을 나타냅니다.
SHELL_LINK_DATA_FLAGS 옵션 설정을 지정합니다. IShellLinkDataList::GetFlags 및 IShellLinkDataList::SetFlags와 함께 사용됩니다.

shlwapi.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

shlwapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IQueryAssociations 파일 형식 또는 프로토콜을 정의하고 애플리케이션과 연결하여 레지스트리에 저장된 정보를 검색하는 프로세스를 간소화하는 메서드를 노출합니다.

Functions

AssocCreate IQueryAssociations 개체에 대한 포인터를 반환합니다.
AssocGetPerceivedType 확장에 따라 파일의 인식된 형식을 검색합니다.
AssocIsDangerous 파일 형식이 잠재적인 보안 위험으로 간주되는지 여부를 결정합니다.
AssocQueryKeyA 레지스트리에서 파일 또는 프로토콜 연결과 관련된 키를 검색하고 검색합니다. (ANSI)
AssocQueryKeyW 레지스트리에서 파일 또는 프로토콜 연결과 관련된 키를 검색하고 검색합니다. (유니코드)

AssocQueryStringA 레지스트리에서 파일 또는 프로토콜 연결 관련 문자열을 검색하고 검색합니다. (AssocQueryStringA)
AssocQueryStringByKeyA 레지스트리에서 지정된 키부터 파일 연결 관련 문자열을 검색하고 검색합니다. (ANSI)
AssocQueryStringByKeyW 레지스트리에서 지정된 키부터 파일 연결 관련 문자열을 검색하고 검색합니다. (유니코드)
AssocQueryStringW 레지스트리에서 파일 또는 프로토콜 연결 관련 문자열을 검색하고 검색합니다. (AssocQueryStringW)
ChrCmpIA 두 문자를 비교합니다. 대/소문자를 구분하지 않고 비교합니다. (ANSI)
ChrCmpIW 두 문자를 비교합니다. 대/소문자를 구분하지 않고 비교합니다. (유니코드)
ColorAdjustLuma RGB 값의 광도를 변경합니다. 색조와 채도는 영향을 받지 않습니다.
ColorHLSToRGB 색을 HLS(색조 채도)에서 RGB 형식으로 변환합니다.
ColorRGBToHLS 색을 RGB에서 HLS(색조 채도) 형식으로 변환합니다.
ConnectToConnectionPoint 클라이언트의 싱크와 연결 지점 컨테이너 간의 연결을 설정하거나 종료합니다.
DllInstall DLL에 대한 설치 및 설정을 처리합니다.
GetAcceptLanguagesA 언어 기본 설정을 지정할 때 웹 사이트에서 사용되는 문자열을 검색합니다. (ANSI)

GetAcceptLanguagesW 언어 기본 설정을 지정할 때 웹 사이트에서 사용되는 문자열을 검색합니다. (유니코드)
GetMenuPosFromID GetMenuPosFromID를 변경하거나 사용할 수 없습니다.
GetProcessReference SetProcessReference에서 제공하는 프로세스별 개체를 검색하여 참조 수를 증가시켜 프로세스를 활성 상태로 유지합니다.
HashData 데이터 배열을 해시합니다.
IntlStrEqNA 지역화된 두 문자열의 시작 부분에서 지정된 문자 수의 대/소문자를 구분하여 비교합니다. (ANSI)
IntlStrEqNIA 지역화된 두 문자열의 시작 부분에서 지정된 수의 문자를 대/소문자를 구분하지 않는 비교를 수행합니다. (ANSI)
IntlStrEqNIW 지역화된 두 문자열의 시작 부분에서 지정된 수의 문자를 대/소문자를 구분하지 않는 비교를 수행합니다. (유니코드)
IntlStrEqNW 지역화된 두 문자열의 시작 부분에서 지정된 문자 수의 대/소문자를 구분하여 비교합니다. (유니코드)
IntlStrEqWorkerA 지역화된 두 문자열의 시작 부분에서 지정된 문자 수를 비교합니다. (ANSI)
IntlStrEqWorkerW 지역화된 두 문자열의 시작 부분에서 지정된 문자 수를 비교합니다. (유니코드)
IsCharSpaceA 문자가 공백을 나타내는지 여부를 결정합니다. (ANSI)

IsCharSpaceW 문자가 공백을 나타내는지 여부를 결정합니다. (유니코드)
IsInternetESCEnabled Windows Internet Explorer가 향상된 보안 구성에 있는지 여부를 확인합니다.
Iso 지정된 운영 체제 및 운영 체제 기능을 확인합니다.
IStream_Copy 스트림을 다른 스트림에 복사합니다.
IStream_Read 지정된 스트림에서 바이트를 읽고 모든 바이트를 성공적으로 읽었는지 여부를 나타내는 값을 반환합니다.
IStream_ReadPidl IStream 개체에서 PIDLIST_RELATIVE 개체로 PIDL(항목 식별자 목록)에 대한 포인터를 읽습니다.
IStream_ReadStr 스트림에서 읽고 문자열에 씁니다.
IStream_Reset 지정된 스트림의 검색 위치를 스트림의 시작 부분으로 이동합니다.
IStream_Size 지정된 스트림의 크기(바이트)를 검색합니다.
IStream_Write 버퍼에서 지정된 스트림에 알 수 없는 형식의 데이터를 씁니다.
IStream_WritePidl PCUIDLIST_RELATIVE 개체의 PIDL(항목 식별자 목록)에 대한 포인터를 IStream 개체에 씁니다.
IStream_WriteStr 문자열에서 읽고 스트림에 씁니다.

IUnknown_AtomicRelease COM(구성 요소 개체 모델) 포인터를 해제하고 NULL로 설정합니다.
IUnknown_GetSite 지정된 개체의 IObjectWithSite::GetSite 메서드를 호출합니다.
IUnknown_GetWindow GetWindow 메서드가 있는 다양한 인터페이스를 쿼리하여 COM(Component Object Model) 개체에서 창 핸들을 검색하려고 시도합니다.
IUnknown_QueryService 지정된 개체에서 서비스에 대한 인터페이스를 검색합니다.
IUnknown_Set COM(구성 요소 개체 모델) 인터페이스 포인터의 값을 변경하고 이전 인터페이스를 해제합니다.
IUnknown_SetSite IObjectWithSite::SetSite 메서드를 호출하여 지정된 개체의 사이트를 설정합니다.
MAKEDLLVERULL DLL 버전 정보를 ULONGLONG 값으로 압축하는 데 사용됩니다.
ParseURLA URL의 기본적인 구문 분석을 수행합니다. (ANSI)
ParseURLW URL의 기본적인 구문 분석을 수행합니다. (유니코드)
PathAddBackslashA 문자열 끝에 백슬래시를 추가하여 경로에 대한 올바른 구문을 만듭니다. (PathAddBackslashA)
PathAddBackslashW 문자열 끝에 백슬래시를 추가하여 경로에 대한 올바른 구문을 만듭니다. (PathAddBackslashW)
PathAddExtensionA 경로 문자열에 파일 이름 확장명을 추가합니다. (ANSI)

PathAddExtensionW 경로 문자열에 파일 이름 확장명을 추가합니다. (유니코드)
PathAppendA 한 경로를 다른 경로의 끝에 추가합니다. (ANSI)
PathAppendW 한 경로를 다른 경로의 끝에 추가합니다. (유니코드)
PathBuildRootA 지정된 드라이브 번호에서 루트 경로를 만듭니다. (ANSI)
PathBuildRootW 지정된 드라이브 번호에서 루트 경로를 만듭니다. (유니코드)
PathCanonicalizeA "." 및 ".."와 같은 탐색 요소를 제거하여 경로를 간소화하여 올바른 형식의 직접 경로를 생성합니다. (ANSI)
PathCanonicalizeW "." 및 ".."와 같은 탐색 요소를 제거하여 경로를 간소화하여 올바른 형식의 직접 경로를 생성합니다. (유니코드)
PathCombineA 올바른 형식의 경로를 나타내는 두 문자열을 하나의 경로로 연결합니다. 또한 상대 경로 요소를 연결합니다. (ANSI)
PathCombineW 올바른 형식의 경로를 나타내는 두 문자열을 하나의 경로로 연결합니다. 또한 상대 경로 요소를 연결합니다. (유니코드)
PathCommonPrefixA 두 경로를 비교하여 공통 접두사 공유 여부를 확인합니다. 접두사는 다음 형식 중 하나입니다. _"C:\", ".", "..", "... \". (ANSI)
PathCommonPrefixW 두 경로를 비교하여 공통 접두사 공유 여부를 확인합니다. 접두사는 다음 형식 중 하나입니다. _"C:\", ".", "..", "... \". (유니코드)

PathCompactPathA 경로 구성 요소를 줄임표로 바꿔 지정된 픽셀 너비 내에 맞게 파일 경로를 자립니다. (ANSI)
PathCompactPathExA 경로 구성 요소를 줄임표로 바꿔 특정 수의 문자에 맞게 경로를 자립니다. (ANSI)
PathCompactPathExW 경로 구성 요소를 줄임표로 바꿔 특정 수의 문자에 맞게 경로를 자립니다. (유니코드)
PathCompactPathW 경로 구성 요소를 줄임표로 바꿔 지정된 픽셀 너비 내에 맞게 파일 경로를 자립니다. (유니코드)
PathCreateFromUrlA 파일 URL을 Microsoft MS-DOS 경로로 변환합니다. (ANSI)
PathCreateFromUrlAlloc 파일 URL에서 경로를 만듭니다.
PathCreateFromUrlW 파일 URL을 Microsoft MS-DOS 경로로 변환합니다. (유니코드)
PathFileExistsA 파일 또는 폴더와 같은 파일 시스템 개체의 경로가 유효한지 여부를 확인합니다. (ANSI)
PathFileExistsW 파일 또는 폴더와 같은 파일 시스템 개체의 경로가 유효한지 여부를 확인합니다. (유니코드)
PathFindExtensionA 경로를 검색하여 확장을 검색합니다. (ANSI)
PathFindExtensionW 경로를 검색하여 확장을 검색합니다. (유니코드)
PathFindFileNameA 경로에서 파일 이름을 검색합니다. (ANSI)

PathFindFileNameW 경로에서 파일 이름을 검색합니다. (유니코드)
PathFindNextComponentA 경로를 구문 분석하고 첫 번째 백슬래시 뒤에 있는 해당 경로 부분을 반환합니다. (ANSI)
PathFindNextComponentW 경로를 구문 분석하고 첫 번째 백슬래시 뒤에 있는 해당 경로 부분을 반환합니다. (유니코드)
PathFindOnPathA 파일을 검색합니다. (ANSI)
PathFindOnPathW 파일을 검색합니다. (유니코드)
PathFindSuffixArrayA 지정된 파일 이름에 접미사 목록 중 하나가 있는지 여부를 확인합니다. (ANSI)
PathFindSuffixArrayW 지정된 파일 이름에 접미사 목록 중 하나가 있는지 여부를 확인합니다. (유니코드)
PathGetArgsA 지정된 경로 내에서 명령줄 인수를 찾습니다. (ANSI)
PathGetArgsW 지정된 경로 내에서 명령줄 인수를 찾습니다. (유니코드)
PathGetCharTypeA 경로와 관련된 문자 형식을 결정합니다. (ANSI)
PathGetCharTypeW 경로와 관련된 문자 형식을 결정합니다. (유니코드)
PathGetDriveNumberA 'A'에서 'Z' 범위 내의 드라이브 문자를 검색하고 해당 드라이브 번호를 반환합니다. (ANSI)

PathGetDriveNumberW 'A'에서 'Z' 범위 내의 드라이브 문자를 검색하고 해당 드라이브 번호를 반환합니다. (유니코드)
PathIsContentTypeA 파일의 등록된 콘텐츠 형식이 지정된 콘텐츠 형식과 일치하는지 여부를 확인합니다. 이 함수는 지정된 파일 형식의 콘텐츠 형식을 가져오고 해당 문자열을 pszContentType과 비교합니다. 대/소문자를 구분하지 않고 비교합니다. (ANSI)
PathIsContentTypeW 파일의 등록된 콘텐츠 형식이 지정된 콘텐츠 형식과 일치하는지 여부를 확인합니다. 이 함수는 지정된 파일 형식의 콘텐츠 형식을 가져오고 해당 문자열을 pszContentType과 비교합니다. 대/소문자를 구분하지 않고 비교합니다. (유니코드)
PathIsDirectoryA 경로가 유효한 디렉터리인지 확인합니다. (ANSI)
PathIsDirectoryEmptyA 지정된 경로가 빈 디렉터리인지 여부를 확인합니다. (ANSI)
PathIsDirectoryEmptyW 지정된 경로가 빈 디렉터리인지 여부를 확인합니다. (유니코드)
PathIsDirectoryW 경로가 유효한 디렉터리인지 확인합니다. (유니코드)
PathIsFileSpecA 경로 구분 문자(예: ':' 또는 '')를 검색합니다. 경로 구분 문자가 없는 경우 경로는 파일 사양 경로로 간주됩니다.
PathIsFileSpecW 경로 구분 문자(예: ':' 또는 '\')를 검색합니다. 경로 구분 문자가 없는 경우 경로는 파일 사양 경로로 간주됩니다.
PathIsHTMLFileA 파일이 HTML 파일인지 여부를 확인합니다. 파일의 확장명용으로 등록된 콘텐츠 형식에 따라 결정됩니다. (ANSI)

PathIsHTMLFileW 파일이 HTML 파일인지 여부를 확인합니다. 파일의 확장명용으로 등록된 콘텐츠 형식에 따라 결정됩니다. (유니코드)
PathIsLFNFileSpecA 파일 이름이 긴 형식인지 여부를 확인합니다. (ANSI)
PathIsLFNFileSpecW 파일 이름이 긴 형식인지 여부를 확인합니다. (유니코드)
PathIsNetworkPathA 경로 문자열이 네트워크 리소스를 나타내는지 여부를 결정합니다. (ANSI)
PathIsNetworkPathW 경로 문자열이 네트워크 리소스를 나타내는지 여부를 결정합니다. (유니코드)
PathIsPrefixA 경로를 검사하여 pszPrefix에서 전달된 접두사 값으로 시작하는지 확인합니다. (ANSI)
PathIsPrefixW 경로를 검사하여 pszPrefix에서 전달된 접두사 값으로 시작하는지 확인합니다. (유니코드)
PathIsRelativeA 경로를 검색하고 상대 경로인지 확인합니다. (ANSI)
PathIsRelativeW 경로를 검색하고 상대 경로인지 확인합니다. (유니코드)
PathIsRootA 경로 문자열이 볼륨의 루트를 참조하는지 여부를 결정합니다. (ANSI)
PathIsRootW 경로 문자열이 볼륨의 루트를 참조하는지 여부를 결정합니다. (유니코드)
PathIsSameRootA 두 경로를 비교하여 공통 루트 구성 요소가 있는지 확인합니다. (ANSI)

PathIsSameRootW 두 경로를 비교하여 공통 루트 구성 요소가 있는지 확인합니다. (유니코드)
PathIsSystemFolderA 기존 폴더에 시스템 폴더로 만드는 특성이 포함되어 있는지 확인합니다. 또는 이 함수는 특정 특성이 폴더를 시스템 폴더로 한정하는지를 나타냅니다. (ANSI)
PathIsSystemFolderW 기존 폴더에 시스템 폴더로 만드는 특성이 포함되어 있는지 확인합니다. 또는 이 함수는 특정 특성이 폴더를 시스템 폴더로 한정하는지를 나타냅니다. (유니코드)
PathIsUNCA 경로 문자열이 드라이브 문자를 기반으로 하는 경로가 아닌 유효한 UNC(범용 명명 규칙) 경로인지 확인합니다. (ANSI)
PathIsUNCServerA 문자열이 서버 경로에 대해서만 유효한 UNC(범용 명명 규칙)인지 확인합니다. (ANSI)
PathIsUNCServerShareA 문자열이 유효한 UNC(유니버설 명명 규칙) 공유 경로인 \server\share인지 확인합니다. (ANSI)
PathIsUNCServerShareW 문자열이 유효한 UNC(유니버설 명명 규칙) 공유 경로인 \server\share인지 확인합니다. (유니코드)
PathIsUNCServerW 문자열이 서버 경로에 대해서만 유효한 UNC(범용 명명 규칙)인지 확인합니다. (유니코드)
PathIsUNCW 경로 문자열이 드라이브 문자를 기반으로 하는 경로가 아닌 유효한 UNC(범용 명명 규칙) 경로인지 확인합니다. (유니코드)
PathIsURLA 지정된 문자열을 테스트하여 유효한 URL 형식을 준수하는지 확인합니다. (ANSI)
PathIsURLW 지정된 문자열을 테스트하여 유효한 URL 형식을 준수하는지 확인합니다. (유니코드)

PathMakePrettyA 모든 대문자 경로를 모든 소문자로 변환하여 경로에 일관된 모양을 제공합니다. (ANSI)
PathMakePrettyW 모든 대문자 경로를 모든 소문자로 변환하여 경로에 일관된 모양을 제공합니다. (유니코드)
PathMakeSystemFolderA 기존 폴더에 시스템 폴더가 될 적절한 특성을 제공합니다. (ANSI)
PathMakeSystemFolderW 기존 폴더에 시스템 폴더가 될 적절한 특성을 제공합니다. (유니코드)
PathMatchSpecA Microsoft MS-DOS 와일드카드 일치 유형을 사용하여 문자열을 검색합니다. (ANSI)
PathMatchSpecExA 경로의 파일 이름을 하나 이상의 파일 이름 패턴과 일치합니다. (ANSI)
PathMatchSpecExW 경로의 파일 이름을 하나 이상의 파일 이름 패턴과 일치합니다. (유니코드)
PathMatchSpecW Microsoft MS-DOS 와일드카드 일치 유형을 사용하여 문자열을 검색합니다. (유니코드)
PathParseIconLocationA 파일 위치 및 아이콘 인덱스가 포함된 파일 위치 문자열을 구문 분석하고 별도의 값을 반환합니다. (ANSI)
PathParseIconLocationW 파일 위치 및 아이콘 인덱스가 포함된 파일 위치 문자열을 구문 분석하고 별도의 값을 반환합니다. (유니코드)
PathQuoteSpacesA 공백의 경로를 검색합니다. 공백이 발견되면 전체 경로가 따옴표로 묶입니다. (ANSI)
PathQuoteSpacesW 공백의 경로를 검색합니다. 공백이 발견되면 전체 경로가 따옴표로 묶입니다. (유니코드)

PathRelativePathToA 한 파일 또는 폴더에서 다른 파일로의 상대 경로를 만듭니다. (ANSI)
PathRelativePathToW 한 파일 또는 폴더에서 다른 파일로의 상대 경로를 만듭니다. (유니코드)
PathRemoveArgsA 지정된 경로에서 인수를 제거합니다. (ANSI)
PathRemoveArgsW 지정된 경로에서 인수를 제거합니다. (유니코드)
PathRemoveBackslashA 지정된 경로에서 후행 백슬래시를 제거합니다. (ANSI)
PathRemoveBackslashW 지정된 경로에서 후행 백슬래시를 제거합니다. (유니코드)
PathRemoveBlanksA 문자열에서 선행 및 후행 공백을 모두 제거합니다. (ANSI)
PathRemoveBlanksW 문자열에서 선행 및 후행 공백을 모두 제거합니다. (유니코드)
PathRemoveExtensionA 경로에서 파일 이름 확장명을 제거합니다(있는 경우). (ANSI)
PathRemoveExtensionW 경로에서 파일 이름 확장명을 제거합니다(있는 경우). (유니코드)
PathRemoveFileSpecA 경로에 있는 경우 후행 파일 이름과 백슬래시를 제거합니다. (ANSI)
PathRemoveFileSpecW 경로에 있는 경우 후행 파일 이름과 백슬래시를 제거합니다. (유니코드)

PathRenameExtensionA 파일 이름의 확장명을 새 확장명으로 바꿉니다. 파일 이름에 확장명을 포함하지 않으면 확장명은 문자열의 끝에 연결됩니다. (ANSI)
PathRenameExtensionW 파일 이름의 확장명을 새 확장명으로 바꿉니다. 파일 이름에 확장명을 포함하지 않으면 확장명은 문자열의 끝에 연결됩니다. (유니코드)
PathSearchAndQualifyA 지정된 경로의 형식이 올바르게 지정되고 정규화되었는지 여부를 확인합니다. (ANSI)
PathSearchAndQualifyW 지정된 경로의 형식이 올바르게 지정되고 정규화되었는지 여부를 확인합니다. (유니코드)
PathSetDlgItemPathA PathCompactPath를 사용하여 경로가 컨트롤에 맞는지 확인하여 창 또는 대화 상자에서 자식 컨트롤의 텍스트를 설정합니다. (ANSI)
PathSetDlgItemPathW PathCompactPath를 사용하여 경로가 컨트롤에 맞는지 확인하여 창 또는 대화 상자에서 자식 컨트롤의 텍스트를 설정합니다. (유니코드)
PathSkipRootA 드라이브 문자 또는 UNC(범용 명명 규칙) 서버/공유 경로 요소 다음에 있는 경로의 첫 번째 문자에 대한 포인터를 검색합니다. (ANSI)
PathSkipRootW 드라이브 문자 또는 UNC(범용 명명 규칙) 서버/공유 경로 요소 다음에 있는 경로의 첫 번째 문자에 대한 포인터를 검색합니다. (유니코드)
PathStripPathA 정규화된 경로 및 파일의 경로 부분을 제거합니다. (ANSI)
PathStripPathW 정규화된 경로 및 파일의 경로 부분을 제거합니다. (유니코드)
PathStripToRootA 루트 정보를 제외한 경로의 모든 파일 및 디렉터리 요소를 제거합니다. (ANSI)

PathStripToRootW 루트 정보를 제외한 경로의 모든 파일 및 디렉터리 요소를 제거합니다. (유니코드)
PathUndecorateA 경로 문자열에서 장식을 제거합니다. (ANSI)
PathUndecorateW 경로 문자열에서 장식을 제거합니다. (유니코드)
PathUnExpandEnvStringsA 정규화된 경로의 특정 폴더 이름을 연결된 환경 문자열로 바꿉니다. (ANSI)
PathUnExpandEnvStringsW 정규화된 경로의 특정 폴더 이름을 연결된 환경 문자열로 바꿉니다. (유니코드)
PathUnmakeSystemFolderA 시스템 폴더로 만드는 폴더에서 특성을 제거합니다. 이 폴더는 파일 시스템에 실제로 있어야 합니다. (ANSI)
PathUnmakeSystemFolderW 시스템 폴더로 만드는 폴더에서 특성을 제거합니다. 이 폴더는 파일 시스템에 실제로 있어야 합니다. (유니코드)
PathUnquoteSpacesA 경로의 시작과 끝에서 따옴표를 제거합니다. (ANSI)
PathUnquoteSpacesW 경로의 시작과 끝에서 따옴표를 제거합니다. (유니코드)
QISearch IUnknown::QueryInterface 메서드의 테이블 기반 구현입니다.
SetProcessReference 호스트된 셸 확장 및 기타 구성 요소가 호스트 프로세스가 조기에 닫히지 않도록 하는 COM(구성 요소 개체 모델) 개체를 제공합니다.

SHAllocShared SHAllocShared를 변경하거나 사용할 수 없습니다.
SHAnsiToAnsi ANSI 문자열을 복사합니다.
SHAnsiToUnicode ANSI 코드 페이지에서 유니코드 코드 페이지로 문자열을 변환합니다.
SHAutoComplete 자동 완성을 사용하여 URL 또는 파일 시스템 경로를 완료하도록 시스템 편집 컨트롤에 지시합니다.
SHCopyKeyA 원본 하위 키의 하위 키와 값을 대상 키에 재귀적으로 복사합니다. SHCopyKey는 키의 보안 특성을 복사하지 않습니다. (ANSI)
SHCopyKeyW 원본 하위 키의 하위 키와 값을 대상 키에 재귀적으로 복사합니다. SHCopyKey는 키의 보안 특성을 복사하지 않습니다. (유니코드)
SHCreateMemStream CreateStreamOnHGlobal과 유사한 프로세스를 사용하여 메모리 스트림을 만듭니다.
SHCreateShellPalette 지정된 디바이스 컨텍스트에 대한 하프톤 팔레트를 만듭니다.
SHCreateStreamOnFileA SHCreateStreamOnFile을 변경하거나 사용할 수 없습니다. 대신 SHCreateStreamOnFileEx를 사용합니다. (ANSI)
SHCreateStreamOnFileEx 파일을 열거나 만들고 해당 파일을 읽거나 쓸 스트림을 검색합니다.
SHCreateStreamOnFileW SHCreateStreamOnFile을 변경하거나 사용할 수 없습니다. 대신 SHCreateStreamOnFileEx를 사용합니다. (유니코드)

SHCreateThread 스레드를 만듭니다.
SHCreateThreadRef COM(구성 요소 개체 모델) 개체에 대한 스레드별 참조를 만듭니다.
SHCreateThreadWithHandle 새 스레드를 만들고 해당 핸들을 검색합니다.
SHDeleteEmptyKeyA 빈 키를 삭제합니다. (ANSI)
SHDeleteEmptyKeyW 빈 키를 삭제합니다. (유니코드)
SHDeleteKeyA 하위 키 및 모든 하위 항목을 삭제합니다. 이 함수는 레지스트리에서 키와 모든 키의 값을 제거합니다. (ANSI)
SHDeleteKeyW 하위 키 및 모든 하위 항목을 삭제합니다. 이 함수는 레지스트리에서 키와 모든 키의 값을 제거합니다. (유니코드)
SHDeleteValueA 지정된 레지스트리 키에서 명명된 값을 삭제합니다. (ANSI)
SHDeleteValueW 지정된 레지스트리 키에서 명명된 값을 삭제합니다. (유니코드)
SHEnumKeyExA 지정된 열린 레지스트리 키의 하위 키를 열거합니다. (SHEnumKeyExA)
SHEnumKeyExW 지정된 열린 레지스트리 키의 하위 키를 열거합니다. (SHEnumKeyExW)
SHEnumValueA 지정된 열린 레지스트리 키의 값을 열거합니다. (ANSI)

SHEnumValueW 지정된 열린 레지스트리 키의 값을 열거합니다. (유니코드)
SHFormatDateTimeA SHFormatDateTime을 변경하거나 사용할 수 없습니다. (ANSI)
SHFormatDateTimeW SHFormatDateTime을 변경하거나 사용할 수 없습니다. (유니코드)
SHFreeShared SHFreeShared를 변경하거나 사용할 수 없습니다.
SHGetAssocKeys IQueryAssociations 개체와 연결된 클래스 하위 키의 배열을 검색합니다.
SHGetInverseCMAP 하프톤 팔레트에 대한 역색 테이블 매핑을 검색합니다.
SHGetThreadRef SHSetThreadRef에서 설정한 스레드별 개체 참조를 검색합니다.
SHGetValueA 레지스트리 값을 검색합니다. (SHGetValueA)
SHGetValueW 레지스트리 값을 검색합니다. (SHGetValueW)
SHGetViewStatePropertyBag SHGetViewStatePropertyBag을 변경하거나 사용할 수 없습니다.
SHGlobalCounterDecrement 전역 카운터를 감소합니다.
SHGlobalCounterGetValue 전역 카운터의 현재 값을 가져옵니다.

SHGlobalCounterIncrement 전역 카운터를 증분합니다.
SHIsLowMemoryMachine 지원 안 됨 (SHIsLowMemoryMachine)
SHLoadIndirectString 간접 문자열('@' 기호로 시작하는 문자열)의 형태로 해당 리소스를 지정하면 지정된 텍스트 리소스를 추출합니다.
SHLocalStrDupA 새로 할당된 메모리에서 문자열의 복사본을 만듭니다. (SHLocalStrDupA)
SHLocalStrDupW 새로 할당된 메모리에 문자열의 복사본을 만듭니다. (SHLocalStrDupW)
SHLockShared SHLockShared를 변경하거나 사용할 수 없습니다.
SHMessageBoxCheckA SHMessageBoxCheck를 변경하거나 사용할 수 없습니다. (ANSI)
SHMessageBoxCheckW SHMessageBoxCheck를 변경하거나 사용할 수 없습니다. (유니코드)
SHOpenRegStream2A 레지스트리 값을 열고 값을 읽거나 쓰는 데 사용할 수 있는 스트림을 제공합니다. 이 함수는 SHOpenRegStream을 대체합니다. (ANSI)
SHOpenRegStream2W 레지스트리 값을 열고 값을 읽거나 쓰는 데 사용할 수 있는 스트림을 제공합니다. 이 함수는 SHOpenRegStream을 대체합니다. (유니코드)
SHOpenRegStreamA 더 이상 사용되지 않습니다. (SHOpenRegStreamA)

SHOpenRegStreamW 더 이상 사용되지 않습니다. (SHOpenRegStreamW)
SHQueryInfoKeyA 지정된 레지스트리 키에 대한 정보를 검색합니다. (ANSI)
SHQueryInfoKeyW 지정된 레지스트리 키에 대한 정보를 검색합니다. (유니코드)
SHQueryValueExA 레지스트리 키를 열고 특정 값을 쿼리합니다. (ANSI)
SHQueryValueExW 레지스트리 키를 열고 특정 값을 쿼리합니다. (유니코드)
SHRegCloseUSKey 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 사용자별 레지스트리 하위 키에 대한 핸들을 닫습니다.
SHRegCreateUSKeyA 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 레지스트리 하위 키를 만들거나 엽니다. (ANSI)
SHRegCreateUSKeyW 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 레지스트리 하위 키를 만들거나 엽니다. (유니코드)
SHRegDeleteEmptyUSKeyA 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 빈 레지스트리 하위 키를 삭제합니다. (ANSI)
SHRegDeleteEmptyUSKeyW 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 빈 레지스트리 하위 키를 삭제합니다. (유니코드)
SHRegDeleteUSValueA 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 레지스트리 하위 키 값을 삭제합니다. (ANSI)

SHRegDeleteUSValueW 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 레지스트리 하위 키 값을 삭제합니다. (유니코드)
SHRegDuplicateHKey 레지스트리 키의 HKEY 핸들을 복제합니다.
SHRegEnumUSKeyA 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 레지스트리 하위 키의 하위 키를 열거합니다. (ANSI)
SHRegEnumUSKeyW 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 레지스트리 하위 키의 하위 키를 열거합니다. (유니코드)
SHRegEnumUSValueA 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 지정된 레지스트리 하위 키의 값을 열거합니다. (ANSI)
SHRegEnumUSValueW 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 지정된 레지스트리 하위 키의 값을 열거합니다. (유니코드)
SHRegGetBoolUSValueA 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)의 레지스트리 하위 키에서 부울 값을 검색합니다. (ANSI)
SHRegGetBoolUSValueW 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)의 레지스트리 하위 키에서 부울 값을 검색합니다. (유니코드)
SHRegGetBoolValueFromHKCUHKLM 레지스트리 키 값을 평가하고 값이 존재하고 예상 상태가 실제 상태와 일치하는지 여부를 반영하는 부울 값을 반환합니다.
SHRegGetIntW 레지스트리에서 숫자 문자열 값을 읽고 정수로 변환합니다.

SHRegGetPathA 레지스트리에서 파일 경로를 검색하여 필요에 따라 환경 변수를 확장합니다. (ANSI)
SHRegGetPathW 레지스트리에서 파일 경로를 검색하여 필요에 따라 환경 변수를 확장합니다. (유니코드)
SHRegGetUSValueA 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)의 레지스트리 하위 키에서 값을 검색합니다. (ANSI)
SHRegGetUSValueW 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)의 레지스트리 하위 키에서 값을 검색합니다. (유니코드)
SHRegGetValueA 레지스트리 값을 검색합니다. (SHRegGetValueA)
SHRegGetValueFromHKCUHKLM 레지스트리에서 지정된 정보를 가져옵니다.
SHRegGetValueW 레지스트리 값을 검색합니다. (SHRegGetValueW)
SHRegOpenUSKeyA 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 레지스트리 하위 키를 엽니다. (ANSI)
SHRegOpenUSKeyW 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 레지스트리 하위 키를 엽니다. (유니코드)
SHRegQueryInfoUSKeyA 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 지정된 레지스트리 하위 키에 대한 정보를 검색합니다. (ANSI)
SHRegQueryInfoUSKeyW 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 지정된 레지스트리 하위 키에 대한 정보를 검색합니다. (유니코드)

SHRegQueryUSValueA 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 열린 레지스트리 하위 키와 연결된 지정된 이름의 형식 및 데이터를 검색합니다. (ANSI)
SHRegQueryUSValueW 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 열린 레지스트리 하위 키와 연결된 지정된 이름의 형식 및 데이터를 검색합니다. (유니코드)
SHRegSetPathA 파일 경로를 사용하고, 폴더 이름을 환경 문자열로 바꾸고, 결과 문자열을 레지스트리에 배치합니다. (ANSI)
SHRegSetPathW 파일 경로를 사용하고, 폴더 이름을 환경 문자열로 바꾸고, 결과 문자열을 레지스트리에 배치합니다. (유니코드)
SHRegSetUSValueA 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 레지스트리 하위 키 값을 설정합니다. (ANSI)
SHRegSetUSValueW 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)에서 레지스트리 하위 키 값을 설정합니다. (유니코드)
SHRegSetValue 지원 안 됨 (SHRegSetValue)
SHRegWriteUSValueA 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)의 레지스트리 하위 키에 값을 씁니다. (ANSI)
SHRegWriteUSValueW 사용자별 하위 트리(HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE)의 레지스트리 하위 키에 값을 씁니다. (유니코드)
SHReleaseThreadRef 스레드 프로시저가 반환되기 전에 스레드 참조를 해제합니다.

SHSendMessageBroadcastA 시스템의 모든 최상위 창에 메시지를 보냅니다. (ANSI)
SHSendMessageBroadcastW 시스템의 모든 최상위 창에 메시지를 보냅니다. (유니코드)
SHSetThreadRef COM(Component Object Model) 개체에 대한 스레드별 참조를 저장합니다. 이렇게 하면 호출자가 스레드의 수명을 제어하여 호출자가 준비되기 전에 Windows가 스레드를 종료하지 않도록 할 수 있습니다.
SHSetValueA 레지스트리 키의 값을 설정합니다. (ANSI)
SHSetValueW 레지스트리 키의 값을 설정합니다. (유니코드)
SHSkipJunction 바인딩 컨텍스트를 검사하여 특정 구성 요소 개체에 바인딩하는 것이 안전한지 확인합니다.
SHStrDupA 새로 할당된 메모리에 문자열의 복사본을 만듭니다. (SHStrDupA)
SHStrDupW 새로 할당된 메모리에 문자열의 복사본을 만듭니다. (SHStrDupW)
SHStripMneumonicA 문자열에서 니모닉 마커를 제거합니다. (ANSI)
SHStripMneumonicW 문자열에서 니모닉 마커를 제거합니다. (유니코드)
SHUnicodeToAnsi 유니코드 코드 페이지에서 ANSI 코드 페이지로 문자열을 변환합니다.
SHUnicodeToUnicode 유니코드 문자열을 복사합니다.

SHUnlockShared SHUnlockShared를 변경하거나 사용할 수 없습니다.
StopWatchFlush StopWatchFlush를 변경하거나 사용할 수 없습니다.
StopWatchMode StopWatchMode를 변경하거나 사용할 수 없습니다.
StrCatBuffA 한 문자열에서 다른 문자열의 끝까지 문자를 복사하고 추가합니다. (ANSI)
StrCatBuffW 한 문자열에서 다른 문자열의 끝까지 문자를 복사하고 추가합니다. (유니코드)
StrCatChainW 두 유니코드 문자열을 연결합니다. 동일한 버퍼에 대해 반복되는 연결이 필요할 때 사용됩니다.
StrCatW 한 문자열을 다른 문자열에 추가합니다.
StrChrA 문자열에서 지정된 문자와 일치하는 첫 번째 문자를 검색합니다. 비교는 대/소문자를 구분합니다. (ANSI)
StrChrIA 문자열에서 지정된 문자와 일치하는 첫 번째 문자를 검색합니다. 대/소문자를 구분하지 않고 비교합니다. (ANSI)
StrChrIW 문자열에서 지정된 문자와 일치하는 첫 번째 문자를 검색합니다. 대/소문자를 구분하지 않고 비교합니다. (유니코드)
StrChrNIW 문자열에서 지정된 문자의 첫 번째 발생을 검색합니다. 대/소문자를 구분하지 않고 비교합니다.

StrChrNW 문자열에서 지정된 문자의 첫 번째 발생을 검색합니다. 비교는 대/소문자를 구분합니다.
StrChrW 문자열에서 지정된 문자와 일치하는 첫 번째 문자를 검색합니다. 비교는 대/소문자를 구분합니다. (유니코드)
StrCmpCA ASCII(C 런타임) 데이터 정렬 규칙을 사용하여 문자열을 비교합니다. 비교는 대/소문자를 구분합니다. (ANSI)
StrCmpCW ASCII(C 런타임) 데이터 정렬 규칙을 사용하여 문자열을 비교합니다. 비교는 대/소문자를 구분합니다. (유니코드)
StrCmpICA ASCII(C 런타임) 데이터 정렬 규칙을 사용하여 두 문자열을 비교합니다. 대/소문자를 구분하지 않고 비교합니다. (ANSI)
StrCmpICW ASCII(C 런타임) 데이터 정렬 규칙을 사용하여 두 문자열을 비교합니다. 대/소문자를 구분하지 않고 비교합니다. (유니코드)
StrCmpIW 두 문자열을 비교하여 동일한지 확인합니다. 대/소문자를 구분하지 않고 비교합니다.
StrCmpLogicalW 두 유니코드 문자열을 비교합니다. 문자열의 숫자는 텍스트가 아닌 숫자 콘텐츠로 간주됩니다. 이 테스트는 대/소문자를 구분하지 않습니다.
StrCmpNA 두 문자열의 시작 부분에서 지정된 수의 문자를 비교하여 동일한지 확인합니다. 비교는 대/소문자를 구분합니다. StrNCmp 매크로는 이름에서만 이 함수와 다릅니다. (ANSI)
StrCmpNCA ASCII(C 런타임) 데이터 정렬 규칙을 사용하여 두 문자열의 시작 부분에서 지정된 수의 문자를 비교합니다. 비교는 대/소문자를 구분합니다. (ANSI)

StrCmpNCW ASCII(C 런타임) 데이터 정렬 규칙을 사용하여 두 문자열의 시작 부분에서 지정된 수의 문자를 비교합니다. 비교는 대/소문자를 구분합니다. (유니코드)
StrCmpNIA 두 문자열의 시작 부분에서 지정된 수의 문자를 비교하여 동일한지 확인합니다. 대/소문자를 구분하지 않고 비교합니다. StrNCmpI 매크로는 이름에서만 이 함수와 다릅니다. (ANSI)
StrCmpNICA ASCII(C 런타임) 데이터 정렬 규칙을 사용하여 두 문자열의 시작 부분에서 지정된 수의 문자를 비교합니다. 대/소문자를 구분하지 않고 비교합니다. (ANSI)
StrCmpNICW ASCII(C 런타임) 데이터 정렬 규칙을 사용하여 두 문자열의 시작 부분에서 지정된 수의 문자를 비교합니다. 대/소문자를 구분하지 않고 비교합니다. (유니코드)
StrCmpNIW 두 문자열의 시작 부분에서 지정된 수의 문자를 비교하여 동일한지 확인합니다. 대/소문자를 구분하지 않고 비교합니다. StrNCmpI 매크로는 이름에서만 이 함수와 다릅니다. (유니코드)
StrCmpNW 두 문자열의 시작 부분에서 지정된 수의 문자를 비교하여 동일한지 확인합니다. 비교는 대/소문자를 구분합니다. StrNCmp 매크로는 이름에서만 이 함수와 다릅니다. (유니코드)
StrCmpW 두 문자열을 비교하여 동일한지 확인합니다. 비교는 대/소문자를 구분합니다.
StrCpyNW 한 문자열의 시작 부분에서 다른 문자열로 지정된 수의 문자를 복사합니다. 참고: 이 함수 또는 StrNCpy 매크로는 사용하지 마세요.
StrCpyW 한 문자열을 다른 문자열에 복사합니다. (StrCpyW)
StrCSpnA 문자열에서 문자 그룹의 첫 번째 발생을 검색합니다. 검색 방법은 대/소문자를 구분하며 종료되는 NULL 문자는 검색 패턴 일치에 포함됩니다. (ANSI)

StrCSpnIA 문자열에서 문자 그룹의 첫 번째 발생을 검색합니다. 검색 방법은 대/소문자를 구분하지 않으며 종결 NULL 문자는 검색 패턴 일치에 포함됩니다. (ANSI)
StrCSpnIW 문자열에서 문자 그룹의 첫 번째 발생을 검색합니다. 검색 방법은 대/소문자를 구분하지 않으며 종결 NULL 문자는 검색 패턴 일치에 포함됩니다. (유니코드)
StrCSpnW 문자열에서 문자 그룹의 첫 번째 발생을 검색합니다. 검색 방법은 대/소문자를 구분하며 종료되는 NULL 문자는 검색 패턴 일치에 포함됩니다. (유니코드)
StrDupA 문자열을 복제합니다. (ANSI)
StrDupW 문자열을 복제합니다. (유니코드)
StrFormatByteSize64A 숫자 값을 크기에 따라 크기 값(바이트, 킬로바이트, 메가바이트 또는 기가바이트)으로 표현되는 숫자를 나타내는 문자열로 변환합니다.
StrFormatByteSizeA 숫자 값을 크기에 따라 크기 값(바이트, 킬로바이트, 메가바이트 또는 기가바이트)으로 표현되는 숫자를 나타내는 문자열로 변환합니다. 한 매개 변수 형식의 StrFormatByteSizeW와 다릅니다.
StrFormatByteSizeEx 크기에 따라 숫자 값을 바이트, 킬로바이트, 메가바이트 또는 기가바이트 단위의 숫자를 나타내는 문자열로 변환합니다.
StrFormatByteSizeW 숫자 값을 크기에 따라 크기 값(바이트, 킬로바이트, 메가바이트 또는 기가바이트)으로 표현되는 숫자를 나타내는 문자열로 변환합니다. 한 매개 변수 형식의 StrFormatByteSizeA와 다릅니다.
StrFormatKBSizeA 숫자 값을 크기 값(KB)으로 표현된 숫자를 나타내는 문자열로 변환합니다. (ANSI)

StrFormatKBSizeW 숫자 값을 크기 값(KB)으로 표현된 숫자를 나타내는 문자열로 변환합니다. (유니코드)
StrFromTimeIntervalA 밀리초 단위로 지정된 시간 간격을 문자열로 변환합니다. (ANSI)
StrFromTimeIntervalW 밀리초 단위로 지정된 시간 간격을 문자열로 변환합니다. (유니코드)
StrIsIntlEqualA 두 문자열의 시작 부분에서 지정된 수의 문자를 비교하여 동일한지 확인합니다. (ANSI)
StrIsIntlEqualW 두 문자열의 시작 부분에서 지정된 수의 문자를 비교하여 동일한지 확인합니다. (유니코드)
StrNCatA 한 문자열의 시작부터 다른 문자열의 끝까지 지정된 수의 문자를 추가합니다. (ANSI)
StrNCatW 한 문자열의 시작부터 다른 문자열의 끝까지 지정된 수의 문자를 추가합니다. (유니코드)
StrPBrkA 문자열에서 지정된 버퍼에 포함된 문자의 첫 번째 항목을 검색합니다. 이 검색에는 종료 null 문자가 포함되지 않습니다. (ANSI)
StrPBrkW 문자열에서 지정된 버퍼에 포함된 문자의 첫 번째 항목을 검색합니다. 이 검색에는 종료 null 문자가 포함되지 않습니다. (유니코드)
StrRChrA 문자열에서 지정된 문자의 마지막 발생을 검색합니다. 비교는 대/소문자를 구분합니다. (ANSI)
StrRChrIA 문자열에서 지정된 문자의 마지막 발생을 검색합니다. 대/소문자를 구분하지 않고 비교합니다. (ANSI)

StrRChrIW 문자열에서 지정된 문자의 마지막 발생을 검색합니다. 대/소문자를 구분하지 않고 비교합니다. (유니코드)
StrRChrW 문자열에서 지정된 문자의 마지막 발생을 검색합니다. 비교는 대/소문자를 구분합니다. (유니코드)
StrRetToBSTR 문자열을 포함하거나 가리키는 IShellFolder::GetDisplayNameOf에서 반환된 STRRET 구조를 수락하고 해당 문자열을 BSTR로 반환합니다.
StrRetToBufA IShellFolder::GetDisplayNameOf에서 반환된 STRRET 구조를 문자열로 변환하고 결과를 버퍼에 배치합니다. (ANSI)
StrRetToBufW IShellFolder::GetDisplayNameOf에서 반환된 STRRET 구조를 문자열로 변환하고 결과를 버퍼에 배치합니다. (유니코드)
StrRetToStrA IShellFolder::GetDisplayNameOf에서 반환된 STRRET 구조를 사용하고 표시 이름을 포함하는 할당된 문자열에 대한 포인터를 반환합니다. (ANSI)
StrRetToStrW IShellFolder::GetDisplayNameOf에서 반환된 STRRET 구조를 사용하고 표시 이름을 포함하는 할당된 문자열에 대한 포인터를 반환합니다. (유니코드)
StrRStrIA 문자열 내에서 지정된 부분 문자열의 마지막 항목을 검색합니다. 대/소문자를 구분하지 않고 비교합니다. (ANSI)
StrRStrIW 문자열 내에서 지정된 부분 문자열의 마지막 항목을 검색합니다. 대/소문자를 구분하지 않고 비교합니다. (유니코드)
StrSpnA 지정된 버퍼에 포함된 문자로 구성된 문자열 내에서 부분 문자열의 길이를 가져옵니다. (ANSI)

StrSpnW 지정된 버퍼에 포함된 문자로 구성된 문자열 내에서 부분 문자열의 길이를 가져옵니다. (유니코드)
StrStrA 문자열 내에서 부분 문자열의 첫 번째 발생을 찾습니다. 비교는 대/소문자를 구분합니다. (StrStrA)
StrStrIA 문자열 내에서 부분 문자열의 첫 번째 발생을 찾습니다. 대/소문자를 구분하지 않고 비교합니다. (ANSI)
StrStrIW 문자열 내에서 부분 문자열의 첫 번째 발생을 찾습니다. 대/소문자를 구분하지 않고 비교합니다. (유니코드)
StrStrNIW 문자열 내에서 부분 문자열의 첫 번째 발생을 찾습니다. 비교는 대/소문자를 구분합니다.
StrStrNW 문자열 내에서 부분 문자열의 첫 번째 발생을 찾습니다. 비교는 대/소문자를 구분합니다. (StrStrNW)
StrStrW 문자열 내에서 부분 문자열의 첫 번째 발생을 찾습니다. 비교는 대/소문자를 구분합니다. (StrStrW)
StrToInt64ExA 10진수 또는 16진수 값을 나타내는 문자열을 64비트 정수로 변환합니다. (ANSI)
StrToInt64ExW 10진수 또는 16진수 값을 나타내는 문자열을 64비트 정수로 변환합니다. (유니코드)
StrToIntA 10진수를 나타내는 문자열을 정수로 변환합니다. StrToLong 매크로는 이 함수와 동일합니다. (ANSI)
StrToIntExA 10진수 또는 16진수를 나타내는 문자열을 정수로 변환합니다. (ANSI)

StrToIntExW 10진수 또는 16진수를 나타내는 문자열을 정수로 변환합니다. (유니코드)
StrToIntW 10진수를 나타내는 문자열을 정수로 변환합니다. StrToLong 매크로는 이 함수와 동일합니다. (유니코드)
StrTrimA 문자열에서 지정된 선행 및 후행 문자를 제거합니다. (ANSI)
StrTrimW 문자열에서 지정된 선행 및 후행 문자를 제거합니다. (유니코드)
UrlApplySchemeA 지정된 URL 문자열에 대한 구성표를 결정하고 적절한 접두사로 문자열을 반환합니다. (ANSI)
UrlApplySchemeW 지정된 URL 문자열에 대한 구성표를 결정하고 적절한 접두사로 문자열을 반환합니다. (유니코드)
UrlCanonicalizeA URL 문자열을 정규 형식으로 변환합니다. (ANSI)
UrlCanonicalizeW URL 문자열을 정규 형식으로 변환합니다. (유니코드)
UrlCombineA 상대 URL 및 기본값을 제공하면 정식 형식으로 URL을 반환합니다. (ANSI)
UrlCombineW 상대 URL 및 기본값을 제공하면 정식 형식으로 URL을 반환합니다. (유니코드)
UrlCompareA 두 URL 문자열을 대/소문자를 구분하여 비교합니다. (ANSI)
UrlCompareW 두 URL 문자열을 대/소문자를 구분하여 비교합니다. (유니코드)

UrlCreateFromPathA Microsoft MS-DOS 경로를 정식화된 URL로 변환합니다. (ANSI)
UrlCreateFromPathW Microsoft MS-DOS 경로를 정식화된 URL로 변환합니다. (유니코드)
UrlEscapeA 인터넷("안전하지 않은" 문자)을 전송하는 동안 변경될 수 있는 URL의 문자 또는 서로게이트 쌍을 해당 이스케이프 시퀀스로 변환합니다. (ANSI)
UrlEscapeSpaces 공백 문자를 해당 이스케이프 시퀀스로 변환하는 매크로입니다.
UrlEscapeW 인터넷("안전하지 않은" 문자)을 전송하는 동안 변경될 수 있는 URL의 문자 또는 서로게이트 쌍을 해당 이스케이프 시퀀스로 변환합니다. (유니코드)
UrlFixupW UrlFixupW를 변경하거나 사용할 수 없습니다.
UrlGetLocationA URL에서 위치를 검색합니다. (ANSI)
UrlGetLocationW URL에서 위치를 검색합니다. (유니코드)
UrlGetPartA URL 문자열을 허용하고 해당 URL의 지정된 부분을 반환합니다. (ANSI)
UrlGetPartW URL 문자열을 허용하고 해당 URL의 지정된 부분을 반환합니다. (유니코드)
UrlHashA URL 문자열을 해시합니다. (ANSI)
UrlHashW URL 문자열을 해시합니다. (유니코드)

UrlIsA URL이 지정된 형식인지 여부를 테스트합니다. (ANSI)
UrlIsFileUrlA URL을 테스트하여 파일 URL인지 확인합니다. (ANSI)
UrlIsFileUrlW URL을 테스트하여 파일 URL인지 확인합니다. (유니코드)
UrlIsNoHistoryA URL이 브라우저가 일반적으로 탐색 기록에 포함하지 않는 URL인지 여부를 반환합니다. (ANSI)
UrlIsNoHistoryW URL이 브라우저가 일반적으로 탐색 기록에 포함하지 않는 URL인지 여부를 반환합니다. (유니코드)
UrlIsOpaqueA URL이 불투명한지 여부를 반환합니다. (ANSI)
UrlIsOpaqueW URL이 불투명한지 여부를 반환합니다. (유니코드)
UrlIsW URL이 지정된 형식인지 여부를 테스트합니다. (유니코드)
UrlUnescapeA 이스케이프 시퀀스를 다시 일반 문자로 변환합니다. (ANSI)
UrlUnescapeInPlace 이스케이프 시퀀스를 다시 일반 문자로 변환하고 원래 문자열을 덮어씁니다.
UrlUnescapeW 이스케이프 시퀀스를 다시 일반 문자로 변환합니다. (유니코드)
어느 플랫폼 어떤Platform을 변경하거나 사용할 수 없습니다.

wnsprintfA 가변 길이 인수 목록을 사용하고 인수 값을 printf 스타일 형식 문자열로 반환합니다. (ANSI)
wnsprintfW 가변 길이 인수 목록을 사용하고 인수 값을 printf 스타일 형식 문자열로 반환합니다. (유니코드)
wvnsprintfA 인수 목록을 가져와 인수 값을 printf 스타일 형식 문자열로 반환합니다. (ANSI)
wvnsprintfW 인수 목록을 가져와 인수 값을 printf 스타일 형식 문자열로 반환합니다. (유니코드)

콜백 함수

DLLGETVERSIONPROC 애플리케이션이 DLL 관련 버전 정보를 가져올 수 있도록 많은 Windows 셸 DLL에서 구현됩니다.

구조체

DLLVERSIONINFO DLL 관련 버전 정보를 받습니다.
DLLVERSIONINFO2 DLL 관련 버전 정보를 받습니다. DllGetVersion 함수와 함께 사용됩니다.
PARSEDURLA ParseURL 함수에서 구문 분석된 URL을 반환하는 데 사용됩니다. (ANSI)
PARSEDURLW ParseURL 함수에서 구문 분석된 URL을 반환하는 데 사용됩니다. (유니코드)

QITAB QISearch 함수에서 단일 인터페이스를 설명하는 데 사용됩니다.

열거형

ASSOCDATA IQueryAssociations::GetData에서 반환할 데이터 형식을 정의하는 데 사용됩니다.
ASSOCKEY IQueryAssociations::GetKey에서 반환할 키의 형식을 지정합니다.
ASSOCSTR IQueryAssociations::GetString에서 반환할 문자열의 형식을 정의하는 데 사용됩니다.
FILETYPEATTRIBUTEFLAGS 파일 연결 PROGID 레지스트리 키의 EditFlags 값에 사용되는 FILETYPEATTRIBUTEFLAGS 상수임을 나타냅니다.
SHGLOBALCOUNTER 다양한 전역 카운터 또는 공유 변수에 대한 식별자입니다. 각 전역 카운터는 SHGlobalCounterIncrement 및 SHGlobalCounterDecrement를 사용하여 증가하거나 감소할 수 있습니다.
SHREGDEL_FLAGS 항목을 삭제할 기본 키를 나타내는 값 집합을 제공합니다.
SHREGENUM_FLAGS 열거형에 사용할 기본 키를 나타내는 값 집합을 제공합니다.
tagSFBS_FLAGS StrFormatByteSizeEx 함수가 재생되지 않은 숫자의 반올림을 처리하는 방법을 지정합니다.
URL_SCHEME URL 구성표를 지정하는 데 사용됩니다.

shobjidl.h 헤더

아티클2023. 08. 26.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)
- Windows 셸
- Windows 속성 시스템

shobjidl.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAccessibilityDockingService Windows 스토어 앱이 표시되고 스냅되지 않거나 시작 관리자가 표시될 때 애플리케이션 창을 모니터 아래쪽에 도킹합니다.
IAccessibilityDockingServiceCallback 접근성 창 도킹 이벤트를 받습니다.
IAccessibleObject 접근성 애플리케이션에서 사용할 수 있는 메서드를 노출합니다.
IApplicationAssociationRegistrationUI 사용자가 연결을 사용자 지정할 수 있는 고급 연결 대화 상자를 시작하는 메서드를 노출합니다.
IAutoCompleteDropDown 편집 컨트롤에서 사용자가 입력한 문자열에 대한 가능한 완성을 포함하는 자동 완성 드롭다운 목록의 표시 상태를 클라이언트가 다시 설정하거나 쿼리할 수 있도록 하는 메서드를 노출합니다.
IBandHost 밴드를 만들고 삭제하고 가용성을 지정하는 메서드를 노출합니다.
ICDBurn 시스템에 CD에 쓰기 위한 하드웨어, CD 기록기 디바이스의 드라이브 문자가 있는지 여부를 확인하고 프로그래밍 방식으로 CD 쓰기 세션을 시작하는 메서드를 노출합니다.

ICDBurnExt ICDBurnExt를 변경하거나 사용할 수 없습니다.
ICommDlgBrowser3 ICommDlgBrowser2의 기능을 확장하고 셸 브라우저를 호스트할 때 공통 파일 대화 상자에서 사용합니다.
IComputerInfoChangeNotify 지원되지 않습니다. 이 인터페이스는 이후 버전의 Windows에서 없을 수 있습니다. (IComputerInfoChangeNotify)
IDeskBand2 deskband 개체에서 반투명 효과를 사용하도록 설정하고 쿼리하는 메서드를 노출합니다.
IDesktopGadget 설치된 가젯을 사용자의 데스크톱에 프로그래밍 방식으로 추가할 수 있는 메서드를 노출합니다.
IDragSourceHelper2 IDragSourceHelper에 기능을 추가하는 메서드를 노출합니다. 이 메서드는 IDragSourceHelper 개체에 대한 끌어서 놓기 작업의 특성을 설정합니다.
IDynamicHWHandler 자동 실행에서 호출됩니다. 사용자에게 표시하기 전에 등록된 처리기에 대한 동적 정보를 가져오는 메서드를 노출합니다.
IEnumerableView 뷰의 콘텐츠를 열거하고 열거 완료 시 콜백에서 알림을 받는 메서드를 노출합니다. 이 인터페이스를 사용하면 보기의 클라이언트가 보기의 폴더 콘텐츠 목록을 공유하려고 시도할 수 있습니다.
IEnumReadyCallback 열거가 완료되면 뷰가 구현자에게 알릴 수 있도록 하는 메서드를 노출합니다.
IFileDialog2 호출자가 일반 파일 대화 상자에서 검색할 수 있는 제한된 특정 위치의 이름을 지정하고 취소 단추에 레이블로 표시할 대체 텍스트를 지정할 수 있는 메서드를 제공하여 IFileDialog 인터페이스를 확장합니다.
IFileDialogControlEvents

애플리케이션이 공통 파일 대화 상자에 추가한 컨트롤과 관련된 이벤트에 대해 애플리케이션에 알림을 받을 수 있도록 하는 메서드를 노출합니다.
IFolderBandPriv IFolderBandPriv는 요구 사항 섹션에 지정된 운영 체제에서 사용할 수 있습니다. 이후 버전에서는 변경되거나 제공되지 않을 수 있습니다.
IFolderViewHost 창에서 IFolderView 개체를 호스트하는 메서드를 노출합니다.
IFolderViewOptions Windows 7 이상 보기와 관련된 폴더 보기 옵션을 제어할 수 있는 메서드를 노출합니다.
IHWEventHandler 자동 실행에서 호출되어 등록된 미디어 형식의 처리를 구현합니다.
IHWEventHandler2 IHWEventHandler 인터페이스를 확장하여 디바이스 처리기에 대한 UAC(사용자 계정 컨트롤) 권한 상승을 처리합니다.
IImageRecompress 이미지를 다시 압축하는 메서드를 노출합니다.
IInsertItem IInsertItem을 변경하거나 사용할 수 없습니다.
INameSpaceTreeAccessible 네임스페이스 트리 컨트롤에서 셸 항목에 대한 접근성 작업을 수행하는 메서드를 노출합니다.
INameSpaceTreeControl2 셸 네임스페이스 항목에 사용할 트리뷰 컨트롤의 표시 스타일을 가져와서 설정하는 메서드를 제공하여 INameSpaceTreeControl 인터페이스를 확장합니다.
INameSpaceTreeControlCustomDraw 사용자가 사용자 지정 네임스페이스 트리 컨트롤 및 해당 항목을 그릴 수 있도록 하는 메서드를 노출합니다.
INameSpaceTreeControlDropHandler

끌어서 놓기에 대한 처리기 메서드를 노출합니다.
INameSpaceTreeControlEvents INameSpaceTreeControl 이벤트를 처리하기 위한 메서드를 노출합니다.
IPreviousVersionsInfo Windows Server 2003에서 제공하는 섀도 복사본 기술로 되돌리기 위해 저장된 이전 버전의 서버 파일 또는 폴더를 확인하는 메서드를 노출합니다.
IPublishingWizard 온라인 인쇄 마법사, 웹 게시 마법사 및 네트워크 위치 추가 마법사를 사용하는 메서드를 노출합니다. Windows Vista에서 IPublishingWizard는 더 이상 웹 게시 마법사 또는 온라인 인쇄 마법사를 지원하지 않습니다.
IQueryCancelAutoPlay 자동 실행 또는 자동 실행을 프로그래밍 방식으로 재정의하는 메서드를 노출합니다. 이렇게 하면 미디어를 삽입할 때 시작되는 콘텐츠의 위치와 형식을 사용자 지정할 수 있습니다.
IQueryCodePage ANSI 코드 페이지의 숫자 값(코드 페이지 식별자)을 가져오고 설정합니다.
IResultsFolder 데이터 개체의 항목을 보유하는 메서드를 노출합니다.
ISearchBoxInfo 호출자가 검색 상자에 입력한 정보를 검색할 수 있도록 하는 메서드를 노출합니다.
IShellRunDll 지원되지 않습니다. (IShellRunDll)
IShellView3 IShellView2::CreateViewWindow2를 대체하는 메서드를 제공하여 IShellView2의 기능을 확장합니다.
IStartMenuPinnedList 시작 메뉴 또는 작업 표시줄에서 애플리케이션 바로 가기를 고정 해제하는 메서드를 노출합니다.
IStreamAsync

비동기 스트림에 대한 입력/출력(I/O)을 관리하는 메서드를 노출합니다.
IStreamUnbufferedInfo 바이트 맞춤을 지원하기 위해 섹터 크기를 결정하는 메서드를 노출합니다.
ITrayDeskBand 데스크밴드를 표시, 숨기기 및 쿼리하는 메서드를 노출합니다.
IUserAccountChangeCallback 사용자 계정을 나타내는 그림이 변경되면 호출되는 메서드를 노출합니다.
IUserNotification2 알림 정보를 설정한 다음 작업 표시줄의 알림 영역과 함께 표시되는 풍선에서 사용자에게 해당 알림을 표시하는 메서드를 노출합니다. (IUserNotification2)
IUserNotificationCallback 알림 풍선에서 마우스 클릭 또는 바로 가기 메뉴 액세스를 처리하기 위한 메서드를 노출합니다. IUserNotification2::Show와 함께 사용됩니다.
IUseToBrowseItem 이 인터페이스는 지원되지 않습니다. 사용하지 마십시오.
IVisualProperties 시각적 속성을 설정하고 가져오는 메서드를 노출합니다.
IWebWizardExtension 마법사 확장의 초기 URL 및 오류 발생 시 특정 URL을 설정하는 메서드를 노출하여 IWizardExtension 인터페이스를 확장합니다.
IWizardExtension 서버 쪽 콘텐츠 페이지를 호스트하는 웹 게시 마법사 및 온라인 인쇄 순서 마법사와 같은 마법사에서 사용됩니다. 이 인터페이스는 지원되는 확장 페이지를 지정하고 해당 페이지로 들어오고 나가는 메서드를 노출합니다.
IWizardSite 마법사 확장에서 자신과 마법사의 나머지 부분 사이의 테두리를 탐색하는 데 사용하는 메서드를 노출합니다.

Functions

SHAddDefaultPropertiesByExt 지정된 파일 확장 프로그램에 대해 등록된 대로 속성 저장소에 기본 속성을 추가합니다.
SHCreateDefaultPropertiesOp 아직 설정되지 않은 셸 항목의 기본 속성을 설정하는 파일 작업을 만듭니다.
SHResolveFolderPathInLibrary 이동되거나 이름이 변경된 라이브러리 폴더의 대상 위치를 resolve 시도합니다.
SHSetDefaultProperties 셸 항목에 기본 속성 집합을 적용합니다.

구조체

NSTCCUSTOMDRAW INameSpaceTreeControlCustomDraw 메서드에서 사용하는 사용자 지정 그리기 구조체입니다.
OVERLAPPED 비동기(겹치는) 입력/출력(I/O)에 사용되는 정보를 포함합니다.

열거형

FOLDERVIEWOPTIONS IFolderViewOptions 인터페이스의 메서드에서 Windows 7 이상 시스템에서 기본적으로 지원되지 않는 Windows Vista 옵션을 활성화하고 새 Windows 7 옵션을 비활성화하는 데 사용됩니다.
NSTCSTYLE2 INameSpaceTreeControl2의 메서드에서 셸 네임스페이스 트리뷰에서 확장된 표시 스타일을 지정하는 데 사용됩니다.

SYNC_ENGINE_STATE_FLAGS 파일 인덱서에서 속성 저장소의 PKEY_StorageProviderStatus 값에 내부 엔진 상태를 노출하기 위해 동기화 엔진에서 사용하는 값을 지정합니다. 속성을 업데이트하려면 먼저 GPS_EXTRINSICPROPERTIES 플래그를 사용하여 IShellItem2::GetPropertyStore를 호출합니다. 다음으로 반환된 개체의 IPropertyStore::SetValue 메서드를 호출하여 PKEY_StorageProviderStatus 키를 지정하여 이러한 SYNC_ENGINE_STATE_FLAGS 사용하여 속성의 비트 마스크 값을 설정합니다.
UNDOCK_REASON 도킹된 접근성 앱 창이 도킹 해제된 이유를 나타내는 값입니다. IAccessibilityDockingServiceCallback::Undocked에서 사용됩니다.
VPCOLORFLAGS 색의 사용을 지정합니다. IVisualProperties 메서드에서 사용됩니다.
VPWATERMARKFLAGS 워터마크 플래그를 지정합니다. IVisualProperties::SetWatermark에서 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

shobjidl_core.h 헤더

아티클2023. 03. 11.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸
- Windows 속성 시스템
- Windows 런타임 C++ 참조
- Windows 검색
- Windows 사이드바
- NFS용 WMI 공급자

shobjidl_core.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IActionProgress 진행률 기반 작업이 상속할 수 있는 추상 기본 클래스를 나타냅니다.
IActionProgressDialog 진행률 대화 상자를 초기화하고 중지하는 메서드를 노출합니다.
IApplicationActivationManager 시작, 파일 및 프로토콜 확장에 대해 Windows 스토어 앱을 활성화하는 메서드를 제공합니다. 일반적으로 디버거 및 디자인 도구에서 이 인터페이스를 사용합니다.
IApplicationAssociationRegistration 특정 파일 연결 형식에 대한 기본 애플리케이션을 쿼리하고 설정하는 메서드와 특정 연결 수준에서 프로토콜을 노출합니다.
IApplicationDesignModeSettings 개발 도구 애플리케이션이 실제 하드웨어 없이 다양한 폼 팩터에 대한 디자인 모드에서 실행되는 Windows 스토어 앱을 테스트하기 위해 네이티브 디스플레이 해상도, 디바이스 배율 인수 및 애플리케이션 보기 상태와 같은 시스템 및 사용자 상태를 동적으로 스푸핑할 수 있도록 합니다. 또한 일반적으로 사용자가 제어하는 상태의 변경 내용을 테스트하여 다양한 시나리오에서 Windows 스토어 앱을 테스트할 수 있습니다.

IApplicationDesignModeSettings2 개발 도구 애플리케이션이 실제 하드웨어 없이 다양한 폼 팩터에 대한 디자인 모드에서 실행되는 Windows 스토어 앱을 테스트하기 위해 Windows 스토어 앱에 보고되는 네이티브 디스플레이 해상도, 디바이스 배율 및 애플리케이션 보기 레이아웃과 같은 시스템 및 사용자 상태를 동적으로 제어할 수 있도록 합니다. 또한 일반적으로 사용자가 제어하는 상태의 변경 내용을 테스트하여 다양한 시나리오에서 Windows 스토어 앱을 테스트할 수 있습니다.
IApplicationDestinations 애플리케이션이 점프 목록의 최근 또는 자주 사용하는 범주에서 하나 또는 모든 대상을 제거할 수 있도록 하는 메서드를 노출합니다.
IApplicationDocumentLists 애플리케이션이 점프 목록에서 최근 또는 자주 사용하는 범주의 콘텐츠를 검색할 수 있도록 하는 메서드를 노출합니다.
IAppVisibility 디스플레이에 유니버설 Windows 플랫폼 앱이 표시되는지 여부를 확인하는 기능을 제공합니다.
IAppVisibilityEvents 애플리케이션이 디스플레이의 상태 변경 내용 및 시작 화면 표시 유형에서 변경 내용에 대한 알림을 받을 수 있도록 합니다.
IAssocHandler 파일 연결 대화 상자 또는 메뉴를 사용하여 작업에 대한 메서드를 노출합니다.
IAssocHandlerInvoker 연결된 애플리케이션 처리기를 호출하는 메서드를 노출합니다.
IAttachmentExecute 클라이언트 애플리케이션과 함께 작동하는 메서드를 노출하여 전자 메일 및 메시징 첨부 파일을 통해 안전한 파일 다운로드 및 교환을 제공하는 사용자 환경을 제공합니다.
IBandSite 밴드 개체를 제어하는 메서드를 노출합니다.
IBrowserFrameOptions 브라우저 또는 호스트가 IShellView에 지원되는 보기 동작 종류를 요청할 수 있습니다.

ICategorizer 항목 식별자 목록에 대한 정보를 가져오는 데 사용되는 메서드를 노출합니다.
ICategoryProvider IShellFolder에 등록된 범주 목록을 노출합니다.
IColumnManager Windows 탐색기 세부 정보 보기에서 열을 검사하고 조작할 수 있는 메서드를 노출합니다. 각 열은 속성 이름을 지정하는 PROPERTYKEY 구조체에서 참조됩니다.
ICommDlgBrowser 셸 브라우저를 호스트할 때 사용할 공통 파일 대화 상자에서 노출됩니다.
ICommDlgBrowser2 ICommDlgBrowser의 기능을 확장합니다. 이 인터페이스는 셸 브라우저를 호스트할 때 공통 파일 대화 상자에 의해 노출됩니다. IShellBrowser 개체에서 QueryInterface를 호출하여 ICommDlgBrowser2에 대한 포인터를 가져올 수 있습니다.
IContactManagerInterop 여러 창을 관리하는 앱에서 ContactManager 메서드에 액세스할 수 있습니다.
IContextMenu Shell 개체와 연결된 바로 가기 메뉴를 만들거나 병합하는 메서드를 노출합니다.
IContextMenu2 Shell 개체와 연결된 바로 가기(상황에 맞는) 메뉴를 만들거나 병합하는 메서드를 노출합니다. 클라이언트 개체가 소유자가 그린 메뉴 항목과 연결된 메시지를 처리할 수 있는 메서드를 추가하여 IContextMenu를 확장합니다.
IContextMenu3 Shell 개체와 연결된 바로 가기 메뉴를 만들거나 병합하는 메서드를 노출합니다. 클라이언트 개체가 소유자 그리기 메뉴 항목과 연결된 메시지를 처리하고 해당 메시지 처리에서 반환 값을 수락하여 IContextMenu2를 확장할 수 있습니다.
IContextMenuCB 상황에 맞는 메뉴의 콜백을 사용하도록 설정하는 메서드를 노출합니다. 예를 들어 상승이 필요한 menuItem에 방패 아이콘을 추가합니다.

IContextMenuSite SHCreateShellFolderView를 사용하여 만든 기본 폴더 뷰에 의해 구현됩니다.
ICreateProcessInputs ICreatingProcess 인터페이스에서 생성되는 프로세스의 일부 매개 변수를 변경하는 데 사용됩니다.
ICreatingProcess ShellExecuteEx 및 IContextMenu에서 호출자가 생성되는 프로세스의 일부 매개 변수를 변경할 수 있도록 하는 데 사용됩니다.
ICurrentItem 항목에 대해 IShellFolder::BindToObject를 호출하여 가져옵니다. 항목이 이전 시간에 항목의 스냅샷을 나타내는 경우 이 인터페이스는 항목의 현재 버전을 가져옵니다.
ICustomDestinationList 애플리케이션이 작업 표시줄에 표시할 대상 및 작업을 비롯한 사용자 지정 점프 목록을 제공할 수 있도록 하는 메서드를 노출합니다.
IDataObjectProvider DataPackage가 상호 운용성을 지원하는 데 사용하는 DataPackage 개체의 IDataObject 인터페이스를 설정하거나 검색할 수 있는 메서드를 제공합니다. DataPackage 개체는 앱에서 다른 앱에 데이터를 제공하는 데 사용됩니다.
IDataTransferManagerInterop 여러 창을 관리하는 Windows 스토어 앱에서 DataTransferManager 메서드에 액세스할 수 있습니다.
IDefaultExtractIconInit 개체와 연결된 기본 아이콘을 설정하는 메서드를 노출합니다.
IDefaultFolderMenuInitialize 바로 가기 메뉴 정보를 얻고 설정하는 데 사용되는 메서드를 제공합니다. 이 정보는 DEFCONTEXTMENU 구조를 통해 SHCreateDefaultContextMenu에 제공된 것과 동일합니다.
IDelegateFolder 대리자 폴더에 할당 및 사용 가능한 항목 ID에 필요한 IMalloc 인터페이스가 제공되는 메서드를 노출합니다.

IDelegateItem 항목 경로의 기본 표현을 즉시 가져오는 데 사용됩니다.
IDeskBand 밴드 개체에 대한 정보를 가져오는 데 사용됩니다.
IDeskBandInfo 기본 Desk 대역폭을 가져오는 메서드를 노출합니다.
IDeskBar 책상 바 조작을 가능하게 하는 메서드를 노출합니다.
IDesktopWallpaper . (IDesktopWallpaper)
IDestinationStreamFactory 속성에 변경 내용을 적용하기 전에 스트림 또는 파일을 수동으로 복사하는 메서드를 노출합니다.
IDisplayItem UI에 표시될 항목 이름과 같은 표시 속성을 가져오는 데 사용할 현재 항목의 버전을 찾는 메서드를 노출합니다.
IDockingWindow 표시, 숨기기 및 임박한 제거를 포함하여 도킹 창 개체에 변경 내용을 알리는 메서드를 노출합니다. 이 인터페이스는 Windows 탐색기 창의 테두리 공간 내에 도킹할 수 있는 창 개체에 의해 구현됩니다.
IDragSourceHelper 애플리케이션이 셸 끌어서 놓기 작업 중에 표시할 이미지를 지정할 수 있도록 셸에 의해 노출됩니다.
IDropTargetHelper 이미지가 대상 창 위에 있는 동안 놓기 대상이 끌기 이미지를 표시할 수 있도록 하는 메서드를 노출합니다.
IEnumAssocHandlers 특정 파일 이름 확장자와 연결된 처리기 컬렉션을 열거할 수 있는 메서드를 노출합니다.

IEnumExplorerCommand IExplorerCommandProvider에서 제공합니다. 이 인터페이스에는 명령 모음에 넣을 명령의 열거형이 포함되어 있습니다.
IEnumExtraSearch 클라이언트가 폴더에 사용 가능한 검색 개체를 결정하는 데 사용하는 표준 OLE 열거자입니다.
IEnumFullIDList Shell 폴더에 있는 항목의 PIDL(항목 식별자 목록)에 대한 포인터를 열거하는 표준 메서드 집합을 노출합니다.
IEnumIDList Shell 폴더에 있는 항목의 PIDL(항목 식별자 목록)에 대한 포인터를 열거하는 데 사용되는 표준 메서드 집합을 노출합니다.
IEnumObjects 메서드를 노출하여 알 수 없는 개체를 열거합니다.
IEnumResources 리소스 열거형 메서드를 노출합니다.
IEnumShellItems IShellItem 인터페이스의 열거형을 노출합니다. 이 인터페이스는 일반적으로 IEnumShellItems 메서드를 호출하여 가져옵니다.
IExecuteCommand 명령 동사와 관련된 지정된 상태 또는 매개 변수를 설정하는 메서드와 해당 동사를 호출하는 메서드를 노출합니다.
IExecuteCommandApplicationHostEnvironment 애플리케이션이 호스트가 데스크톱 또는 몰입형 모드에 있는지 여부를 확인할 수 있도록 하는 단일 메서드를 제공합니다.
IExecuteCommandHost IExplorerCommand 기반 셸 동사 처리기가 애플리케이션이 호출된 호스트 구성 요소의 UI 모드를 쿼리할 수 있도록 하는 메서드를 제공합니다.

IExplorerBrowser IExplorerBrowser는 탐색할 수 있거나 데이터 개체의 보기를 호스트할 수 있는 브라우저 개체입니다. 전체 기능을 갖춘 브라우저 개체로서 자동 이동 로그도 지원합니다.
IExplorerBrowserEvents 탐색기 브라우저 탐색 및 보기 만들기 이벤트의 알림에 대한 메서드를 노출합니다.
IExplorerCommand 명령 모양을 가져오는 메서드를 노출하거나, 하위 명령을 열거하거나, 명령을 호출합니다.
IExplorerCommandProvider 탐색기 명령 및 명령 열거자를 만드는 메서드를 노출합니다.
IExplorerCommandState 명령 상태를 검색할 수 있는 단일 메서드를 노출합니다.
IExplorerPaneVisibility IShellFolder 구현에서 Windows 탐색기에서 사용하여 표시되는 창에 대한 제안을 보기에 제공합니다.
IExtractImage Shell 폴더에서 썸네일 이미지를 요청하는 메서드를 노출합니다.
IExtractImage2 IExtractImage의 기능을 확장합니다.
IFileDialog 일반 파일 대화 상자에서 결과를 초기화, 표시 및 가져오는 메서드를 노출합니다.
IFileDialogCustomize 애플리케이션이 공통 파일 대화 상자에 컨트롤을 추가할 수 있도록 하는 메서드를 노출합니다.
IFileDialogEvents 공통 파일 대화 상자 내에서 이벤트 알림을 허용하는 메서드를 노출합니다.

IFileIsInUse 다른 애플리케이션에서 사용 중인 파일에 대한 정보를 얻거나 닫기 위해 호출할 수 있는 메서드를 노출합니다.
IFileOpenDialog 열려 있는 대화 상자와 관련된 메서드를 추가하여 IFileDialog 인터페이스를 확장합니다.
IFileOperation 진행률 및 오류 대화 상자를 제공하는 메서드뿐만 아니라 셸 항목을 복사, 이동, 이름 바꾸기, 만들기 및 삭제하는 메서드를 노출합니다. 이 인터페이스는 SHFileOperation 함수를 대체합니다.
IFileOperationProgressSink IFileOperation 호출자가 해당 인터페이스를 통해 수행하는 작업의 세부 정보를 모니터링하는 데 사용하는 풍부한 알림 시스템을 제공하는 메서드를 노출합니다.
IFileSaveDialog 파일과 함께 유지할 메타데이터 컬렉션을 지원하는 메서드를 포함하는 저장 대화 상자와 관련된 메서드를 추가하여 IFileDialog 인터페이스를 확장합니다.
IFileSyncMergeHandler . (IFileSyncMergeHandler)
IFileSystemBindData IShellFolder::P arseDisplayName에 대한 호출을 최적화하기 위한 파일 시스템 정보를 저장하는 메서드를 노출합니다.
IFileSystemBindData2 IShellFolder::P arseDisplayName 호출을 최적화하기 위한 파일 시스템 정보를 저장하는 IFileSystemBindData를 확장합니다. 이 인터페이스는 기능 집합을 추가하거나 파일 ID 또는 CLSID(접합 클래스 식별자)를 가져옵니다.
IFolderFilter 서버 애플리케이션에 의해 Shell 폴더의 열거형을 필터링하는 방법을 지정하기 위해 클라이언트에서 노출합니다.
IFolderFilterSite 클라이언트가 Shell 폴더 열거형을 필터링하는 방법을 지정할 수 있도록 호스트에서 내보냅니다.

IFolderView 폴더의 표시 옵션에 대한 정보를 검색하고, 해당 폴더에서 지정된 항목을 선택하고, 폴더의 보기 모드를 설정하는 메서드를 노출합니다. (IFolderView)
IFolderView2 폴더의 표시 옵션에 대한 정보를 검색하고, 해당 폴더에서 지정된 항목을 선택하고, 폴더의 보기 모드를 설정하는 메서드를 노출합니다. (IFolderView2)
IFolderViewSettings 폴더 보기 설정을 가져오는 메서드를 노출합니다.
IFrameworkInputPane 입력 창의 상태 변경 및 위치를 앱에 알릴 수 있는 메서드를 제공합니다.
IFrameworkInputPaneHandler 입력 창(화상 키보드 또는 필기 패널)이 표시되거나 숨겨질 때 앱에 알림을 표시할 수 있습니다. 이렇게 하면 입력 창에 의해 가려진 입력 영역(예: 텍스트 상자)이 없도록 앱 창이 디스플레이를 조정할 수 있습니다.
IHandlerActivationHost . (IHandlerActivationHost)
IHandlerInfo 처리기에 대한 정보를 제공하는 메서드를 IHandlerActivationHost 인터페이스의 메서드에 제공합니다.
IHomeGroup 컴퓨터의 HomeGroup 멤버 자격 상태를 결정하고 공유 마법사를 표시하는 메서드를 노출합니다.
IIdentityName 메서드를 노출하여 두 항목을 비교하여 동일한지 확인합니다.
IInitializeCommand 애플리케이션 지정 명령 이름과 등록된 속성을 사용하여 IExplorerCommandState, IExecuteCommand 또는 IDropTarget을 구현하는 개체를 초기화하는 데 사용되는 단일 메서드를 노출합니다.

IInitializeWithBindCtx 바인딩 컨텍스트를 사용하여 속성 처리기, 썸네일 처리기 또는 미리 보기 처리기와 같은 처리기를 초기화하는 메서드를 노출합니다.
IInitializeWithItem IShellItem을 사용하여 속성 처리기, 미리 보기 처리기 또는 미리 보기 처리기와 같은 처리기를 초기화하는 데 사용되는 메서드를 노출합니다.
IInitializeWithPropertyStore 속성 저장소를 사용하여 속성 처리기, 썸네일 처리기 또는 미리 보기 처리기와 같은 처리기를 초기화하는 메서드를 노출합니다.
IInitializeWithWindow 클라이언트가 데스크톱 애플리케이션에서 사용되는 Windows 런타임 개체에 소유자 창을 제공할 수 있는 메서드를 노출합니다.
IInputObject Shell에 포함된 사용자 입력 개체에 대한 UI 활성화 및 프로세스 가속기를 변경하는 메서드를 노출합니다.
IInputObject2 전역 가속기를 처리하여 IInputObject를 확장하는 메서드를 노출합니다.
IInputObjectSite Shell에 포함된 사용자 입력 개체에 대한 포커스 변경 내용을 전달하는 데 사용되는 메서드를 노출합니다.
IIOCancelInformation 진행률 대화 상자에서 프로세스 스레드에 취소 창 메시지를 게시하는 메서드를 노출합니다.
IItemNameLimits 유효하고 잘못된 문자 목록 또는 네임스페이스의 이름 최대 길이를 검색합니다. 유효성 검사 구문 분석 및 번역에 이 인터페이스를 사용합니다.
IKnownFolder 애플리케이션이 알려진 폴더의 범주, 유형, GUID, PIDL(항목 식별자 목록) 값에 대한 포인터, 리디렉션 기능 및 정의에 대한 정보를 검색할 수 있도록 하는 메서드를 노출합니다.

IKnownFolderManager 기존 알려진 폴더를 만들거나 열거하거나 관리하는 메서드를 노출합니다.
ILaunchSourceAppUserModelId AppUserModelId를 검색하는 메서드를 제공합니다.
ILaunchSourceViewSizePreference 원본 애플리케이션에 대한 정보를 검색하는 메서드를 제공합니다.
ILaunchTargetMonitor . (ILaunchTargetMonitor)
ILaunchTargetViewSizePreference 새 애플리케이션 창에 대한 기본 보기 크기를 검색하는 메서드를 제공합니다.
IMenuBand COM(구성 요소 개체 모델) 개체가 적절한 메시지를 받고 번역할 수 있도록 하는 메서드를 노출합니다.
IMenuPopup IMenuPopup을 변경하거나 사용할 수 없습니다.
IModalWindow 모달 창을 나타내는 메서드를 노출합니다.
INameSpaceTreeControl 셸 항목 트리에서 노드를 보고 조작하는 데 사용되는 메서드를 노출합니다.
INameSpaceTreeControlFolderCapabilities 폴더의 System.IsPinnedToNameSpaceTree 필터링 지원 상태를 검색하는 단일 메서드를 노출합니다.
INamespaceWalk 지정된 루트 노드에서 네임스페이스를 안내하는 메서드를 노출합니다. 연습의 깊이를 지정하고 이동한 모든 노드의 ID를 포함하는 선택적 배열이 반환됩니다.

INamespaceWalkCB INamespaceWalk와 함께 사용되는 메서드를 노출하는 콜백 인터페이스입니다.
INamespaceWalkCB2 네임스페이스 워크를 완료하는 데 필요한 메서드를 사용하여 INamespaceWalkCB를 확장합니다. 이 메서드는 연습 중에 수집된 데이터를 제거합니다.
INewMenuClient Windows 7 메뉴에서 항목을 조작할 수 있는 메서드를 노출합니다.
INewWindowManager 다른 창에서 시작된 창을 표시할지 차단할지 여부를 결정하는 메서드를 노출하여 팝업 창을 제어할 수 있습니다.
IObjectProvider 메서드를 노출하여 다른 개체의 GUID로 명명된 개체를 검색합니다. QueryService와 달리 이 인터페이스는 해당 기능을 다른 개체에 위임하지 않습니다.
IObjectWithAppUserModelID 사용자 지정 IAssocHandler 개체의 구현자가 명시적 애플리케이션 사용자 모델 ID(AppUserModelID)에 대한 액세스를 제공할 수 있도록 하는 메서드를 노출합니다.
IObjectWithBackReferences 개체가 보유한 백 참조와 상호 작용하는 메서드를 제공합니다.
IObjectWithCancelEvent 지원되지 않습니다. 호출자에게 호출된 개체가 작업의 취소를 나타내는 신호를 받을 이벤트를 제공합니다.
IObjectWithFolderEnumMode 구문 분석된 항목의 열거 모드를 가져와서 설정하는 메서드를 노출합니다.
IObjectWithProgID 개체와 연결된 ProgID에 대한 액세스를 제공하는 메서드를 노출합니다.
IObjectWithSelection 셸 항목 배열로 표시되는 선택한 항목을 얻거나 설정하는 메서드를 노출합니다.

IOpenControlPanel 제어판 보기 상태, 개별 제어판 항목의 경로를 검색하고 제어판 자체 또는 개별 제어판 항목을 여는 메서드를 노출합니다.
IOpenSearchSource 사용자 지정 클라이언트 쪽 OpenSearch 데이터 원본에서 검색 결과를 가져오는 메서드를 노출합니다.
IOperationsProgressDialog 진행률 대화 상자를 가져오기, 설정 및 쿼리하는 메서드를 노출합니다.
IPackageDebugSettings 디버거 개발자가 일시 중단 또는 다시 시작과 같은 Windows 스토어 앱의 수명 주기를 제어할 수 있습니다.
IPackageExecutionStateChangeNotification Windows 스토어 앱 디버깅 중에 패키지 상태 변경 알림을 받을 수 있습니다.
IParentAndItem 부모 및 부모의 자식 ID를 가져오고 설정하는 메서드를 노출합니다. IParentAndItem은 일반적으로 IShellItems에서 구현되지만 IShellItem에만 해당되지는 않습니다.
IParseAndCreateItem IParseAndCreateItem 인터페이스
IPersistFolder Shell 폴더 개체를 초기화하는 메서드를 노출합니다.
IPersistFolder2 Shell 폴더 개체에서 정보를 가져오는 메서드를 노출합니다.
IPersistFolder3 폴더 개체가 폴더 바로 가기의 기본이 아닌 처리를 구현할 수 있도록 하여 IPersistFolder 및 IPersistFolder2 인터페이스를 확장합니다.
IPersistIDList 항목 식별자 목록을 유지하는 데 사용되는 메서드를 노출합니다.

IPreviewHandler 서식 있는 미리 보기 표시를 위한 메서드를 노출합니다.
IPreviewHandlerFrame 미리 보기 처리기가 바로 가기 키를 호스트에 전달할 수 있도록 합니다. 이 인터페이스는 바로 가기 키 목록을 검색하고 호스트가 바로 가기 키를 처리하도록 지시합니다.
IPreviewHandlerVisuals 미리 보기 처리기에 색 및 글꼴 정보를 적용하는 메서드를 노출합니다.
IPreviewItem 미리 보기 창에 표시될 항목을 식별합니다.
IProfferService 개체가 동일한 호스트의 다른 개체에 서비스를 제공하는 일반적인 메커니즘을 노출합니다.
IPropertyUI 개발자는 대신 IPropertyDescription을 사용해야 합니다. (IPropertyUI)
IQueryContinue 개체가 작업을 계속할 수 있는 권한을 클라이언트에 쿼리하는 간단한 표준 메커니즘을 제공하는 메서드를 노출합니다.
IRegTreeItem Tree-View 컨트롤 창 스타일 플래그가 설정된 트리 뷰 컨트롤에서 항목의 상태를 검색하고 설정하는 메서드를 노출합니다.
IRelatedItem 특정 관계를 사용하여 관련 항목을 파생시키는 메서드를 노출합니다.
IRemoteComputer 원격 개체에서 호출될 때 네임스페이스 확장을 열거하거나 초기화하는 메서드를 노출합니다. 이 인터페이스는 예를 들어 원격 프린터 가상 폴더를 초기화하는 데 사용됩니다.
IResolveShellLink 애플리케이션에서 Shell 폴더 개체가 해당 항목 중 하나에 대한 링크를 확인하도록 요청할 수 있는 메서드를 노출합니다.

IRunnableTask 백그라운드 스레드에서 작업을 수행할 수 있도록 개체에 의해 노출될 수 있는 자유 스레드 인터페이스입니다.
ISearchFolderItemFactory 검색 폴더를 만들고 수정하는 메서드를 노출합니다.
ISharingConfigurationManager 사용자(C:\Users) 또는 공용(C:\Users\Public) 폴더에 대한 컴퓨터의 기본 공유 설정에 대한 정보를 설정하고 검색하는 메서드를 노출합니다. 프린터 공유를 제어할 수 있는 메서드 집합도 노출합니다.
IShellBrowser 셸 뷰의 호스트(IShellView를 구현하는 개체)에 의해 구현됩니다. 호스팅하는 보기에 대한 서비스 및 탐색기 창의 컨텍스트에서 실행되는 다른 개체를 제공하는 메서드를 노출합니다.
IShellExtInit 속성 시트, 바로 가기 메뉴 및 끌어서 놓기 처리기(기본값이 아닌 끌어서 놓기 작업 중에 바로 가기 메뉴에 항목을 추가하는 확장)에 대한 셸 확장을 초기화하는 메서드를 노출합니다.
IShellFolder 모든 Shell 네임스페이스 폴더 개체에 의해 노출되는 해당 메서드는 폴더를 관리하는 데 사용됩니다.
IShellFolder2 IShellFolder의 기능을 확장합니다. 해당 메서드는 Shell 폴더의 내용에 대한 다양한 정보를 제공합니다.
IShellIcon IShellFolder 개체에 대한 아이콘 인덱스를 가져오는 메서드를 노출합니다.
IShellIconOverlayIdentifier 아이콘 오버레이 처리기와 셸 간의 모든 통신을 처리하는 메서드를 노출합니다.
IShellItem 셸 항목에 대한 정보를 검색하는 메서드를 노출합니다. IShellItem 및 IShellItem2는 새 코드에서 항목의 기본 표현입니다.

IShellItem2 항목의 다양한 속성 값을 검색하는 메서드를 사용하여 IShellItem을 확장합니다. IShellItem 및 IShellItem2는 새 코드에서 항목의 기본 표현입니다.
IShellItemArray Shell 항목 배열을 만들고 조작하는 메서드를 노출합니다.
IShellItemFilter 서버 애플리케이션으로 셸 항목의 열거형을 필터링하는 방법을 지정하기 위해 클라이언트에서 노출합니다.
IShellItemImageFactory 셸 항목에 대한 아이콘 또는 축소판 그림을 반환하는 메서드를 노출합니다. 요청된 항목에 미리 보기 또는 아이콘을 사용할 수 없는 경우 셸에서 클래스별 아이콘을 제공할 수 있습니다.
IShellItemResources 셸 항목 리소스를 조작하고 쿼리하는 메서드를 노출합니다.
IShellLibrary 라이브러리를 만들고 관리하기 위한 메서드를 노출합니다.
IShellLinkA 셸 링크를 만들고, 수정하고, 확인하는 메서드를 노출합니다. (ANSI)
IShellLinkDataList 애플리케이션이 셸 링크에 추가 데이터 블록을 연결할 수 있도록 하는 메서드를 노출합니다. 이러한 메서드는 데이터 블록을 추가, 복사 또는 제거합니다.
IShellLinkW 셸 링크를 만들고, 수정하고, 확인하는 메서드를 노출합니다. (유니코드)
IShellMenu 시작 메뉴 및 즐겨찾기 메뉴와 같은 셸 메뉴와 상호 작용하는 메서드를 노출합니다.
IShellMenuCallback 메뉴 밴드에서 메시지를 받는 메서드를 노출하는 콜백 인터페이스입니다.

IShellPropSheetExt 속성 시트 처리기가 파일 개체에 대해 표시된 속성 시트의 페이지를 추가하거나 바꿀 수 있도록 하는 메서드를 노출합니다.
IShellTaskScheduler IShellTaskScheduler를 변경하거나 사용할 수 없습니다.
IShellView Windows 탐색기 또는 폴더 창에 보기를 표시하는 메서드를 노출합니다.
IShellView2 IShellView의 기능을 확장합니다.
ISuspensionDependencyManager . (ISuspensionDependencyManager)
ITaskbarList 작업 표시줄을 제어하는 메서드를 노출합니다. 작업 표시줄에서 항목을 동적으로 추가, 제거 및 활성화할 수 있습니다.
ITaskbarList2 창을 전체 화면 표시로 표시하는 메서드를 노출하여 ITaskbarList 인터페이스를 확장합니다.
ITaskbarList3 Windows 7에 추가된 통합 시작 및 전환 작업 표시줄 단추 기능을 지원하는 메서드를 노출하여 ITaskbarList2를 확장합니다.
ITaskbarList4 호출자가 탭 미리 보기 및 피킹 기능에 대한 두 가지 속성 값을 제어할 수 있는 메서드를 제공하여 ITaskbarList3을 확장합니다.
IThumbnailHandlerFactory 항목의 썸네일 처리기를 검색하는 메서드를 노출합니다. 자식 IDList에 사용되는 추출기를 지정하려면 이 인터페이스를 구현합니다.
ITransferAdviseSink 상태 수집 및 실패 정보를 지원하는 메서드를 노출합니다.

ITransferDestination 복사 또는 이동 작업에 대한 대상 셸 항목을 만드는 메서드를 노출합니다. 이 인터페이스는 ITransferDestination::Advise 메서드를 제공하여 파일 작업을 보다 강력하게 제어할 수 있도록 제공됩니다.
ITransferMediumItem 복사 엔진에서 QueryInterface를 호출하여 인터페이스 ITransferDestination 또는 인터페이스 ITransferSource에 대한 포인터를 반환할 항목을 가져오는 데 사용됩니다. 복사, 이동 또는 삭제 작업을 위해 이러한 인터페이스를 쿼리하고 열거할 수 있습니다.
ITransferSource 복사, 이동, 재활용 등을 포함하여 IShellItem을 조작하는 메서드를 노출합니다. 이 인터페이스는 ITransferSource::Advise 메서드를 제공하여 파일 작업을 보다 강력하게 제어할 수 있도록 제공됩니다.
IUpdateIDList 폴더 개체의 자식에 대한 ITEMIDLIST를 업데이트하는 메서드를 제공합니다.
IUserNotification 알림 정보를 설정한 다음 작업 표시줄의 알림 영역과 함께 표시되는 풍선에서 사용자에게 해당 알림을 표시하는 메서드를 노출합니다. (IUserNotification)
IViewStateIdentityItem 뷰 사용자 지정이 기억될 항목인 정식 지속성 항목을 제공합니다.
IVirtualDesktopManager 애플리케이션이 가상 작업 영역을 형성하는 창 그룹과 상호 작용할 수 있도록 하는 메서드를 노출합니다.

Functions

FreeIDListArray PIDL(항목 식별자 목록) 목록 배열에 대한 포인터에서 사용하는 메모리를 해제합니다.

FreeIDListArrayChild 자식 항목 ID에 대한 포인터 배열의 메모리 공간을 해제합니다. 이렇게 하면 배열 내의 PITEMID_CHILDs 및 배열 자체가 모두 해제됩니다.
FreeIDListArrayFull PIDL(항목 식별자 목록) 배열에 대한 포인터의 메모리 공간을 해제합니다. 이렇게 하면 배열 내의 PIDLIST_ABSOLUTEs 및 배열 자체가 모두 해제됩니다.
FreeKnownFolderDefinitionFields IKnownFolder::GetFolderDefinition에서 결과에서 할당된 필드를 해제합니다.
GetCurrentProcessExplicitAppUserModelID 현재 프로세스에 대한 애플리케이션 정의 명시적 애플리케이션 사용자 모델 ID(AppUserModelID)를 검색합니다.
SetContractDelegateWindow 기본 포그라운드 창이 아닌 앱 창을 앱의 계약과 연결합니다. 네이티브 C++에서 Windows 스토어 앱을 작성하는 개발자인 경우 이 함수를 사용합니다.
SetCurrentProcessExplicitAppUserModelID 작업 표시줄에 대한 현재 프로세스를 식별하는 고유한 애플리케이션 정의 애플리케이션 사용자 모델 ID(AppUserModelID)를 지정합니다. 이 식별자를 사용하면 애플리케이션이 연결된 프로세스와 창을 단일 작업 표시줄 단추로 그룹화할 수 있습니다.
SHAddFolderPathToLibrary 라이브러리에 폴더를 추가합니다.
SHAssocEnumHandlers 지정된 파일 이름 확장명 처리기 집합에 대한 열거형 개체를 반환합니다.
SHAssocEnumHandlersForProtocolByApplication 지정된 프로토콜과 연결된 처리기에 대한 액세스를 제공하는 열거형 인터페이스를 가져옵니다.
SHCreateAssociationRegistration Windows에서 제공하는 인터페이스의 스톡 구현을 기반으로 IApplicationAssociationRegistration 개체를 만듭니다.

SHCreateDefaultExtractIcon IDefaultExtractIconInit 인터페이스를 통해 기본값을 추가로 구성할 수 있는 표준 아이콘 추출기를 만듭니다.
SHCreateItemFromIDList 포인터에서 PIDL(항목 식별자 목록)에 대한 Shell 항목 개체를 만들고 초기화합니다. 결과 셸 항목 개체는 IShellItem 인터페이스를 지원합니다.
SHCreateItemFromParsingName 구문 분석 이름에서 셸 항목 개체를 만들고 초기화합니다.
SHCreateItemFromRelativeName 상대 구문 분석 이름에서 Shell 항목 개체를 만들고 초기화합니다.
SHCreateItemInKnownFolder 알려진 폴더 내에 있는 단일 파일에 대한 Shell 항목 개체를 만듭니다.
SHCreateItemWithParent 부모 폴더와 자식 항목 ID가 지정된 셸 항목을 만듭니다.
SHCreateLibrary IShellLibrary 개체를 만듭니다.
SHCreateShellItemArray Shell 항목 배열 개체를 만듭니다.
SHCreateShellItemArrayFromDataObject 데이터 개체에서 Shell 항목 배열 개체를 만듭니다.
SHCreateShellItemArrayFromIDLists ITEMIDLIST 구조 목록에서 Shell 항목 배열 개체를 만듭니다.
SHCreateShellItemArrayFromShellItem 단일 셸 항목에서 한 요소의 배열을 만듭니다.
SHGetIDListFromObject 개체의 PIDL(항목 식별자 목록)에 대한 포인터를 검색합니다.

SHGetItemFromDataObject IDataObject에서 지정한 항목을 기반으로 IShellItem 또는 관련 개체를 만듭니다.
SHGetItemFromObject 개체에 대한 IShellItem을 검색합니다.
SHGetNameFromIDList IDList로 식별된 항목의 표시 이름을 검색합니다.
SHGetPropertyStoreFromIDList PIDL(항목 식별자 목록)에 대한 포인터에서 IPropertyStore 또는 관련 인터페이스를 지원하는 개체를 검색합니다.
SHGetPropertyStoreFromParsingName 경로 또는 구문 분석 이름이 지정된 항목에 대한 속성 저장소를 반환합니다.
SHGetTemporaryPropertyForItem 지정된 항목에 대한 임시 속성을 검색합니다. 임시 속성은 항목에 다시 유지되지 않고 IShellItem 개체의 수명 동안만 속성을 보유하는 읽기/쓰기 저장소입니다.
SHLoadLibraryFromItem 지정된 라이브러리 정의 파일에서 IShellLibrary 개체를 만들고 로드합니다.
SHLoadLibraryFromKnownFolder 지정된 알려진 폴더 ID에 대한 IShellLibrary 개체를 만들고 로드합니다.
SHLoadLibraryFromParsingName 지정된 경로에 대한 IShellLibrary 개체를 만들고 로드합니다.
SHRemoveFolderPathFromLibrary 라이브러리에서 폴더를 제거합니다.
SHResolveLibrary 라이브러리의 모든 위치(이동 또는 이름이 변경된 위치)를 확인합니다.
SHSaveLibraryInFolderPath IShellLibrary 개체를 디스크에 저장합니다.

SHSetTemporaryPropertyForItem 지정된 항목에 대한 임시 속성을 설정합니다. 임시 속성은 항목에 다시 쓰는 대신 IShellItem 개체의 수명 동안만 속성을 보유하는 읽기/쓰기 저장소에 유지됩니다.
SHShowManageLibraryUI 사용자가 라이브러리 폴더 및 기본 저장 위치를 관리할 수 있는 라이브러리 관리 대화 상자를 표시합니다.
SHSimpleIDListFromPath 더 이상 사용되지 않습니다. 경로를 전달할 때 ITEMIDLIST 구조체에 대한 포인터를 반환합니다.

구조체

BANDSITEINFO 밴드 사이트에 대한 정보를 포함합니다. 이 구조체는 IBandSite::GetBandSiteInfo 및 IBandSite::SetBandSiteInfo 메서드와 함께 사용됩니다.
CATEGORY_INFO 범주 정보를 포함합니다. 구성 요소 범주는 CATID(공통 범주 식별자)를 공유하는 논리적으로 관련된 COM(구성 요소 개체 모델) 클래스 그룹입니다.
CM_COLUMNINFO 열 정보를 정의합니다. IColumnManager 인터페이스의 멤버에서 사용됩니다.
CMINVOKECOMMANDINFO IContextMenu::InvokeCommand에서 바로 가기 메뉴 명령을 호출하는 데 필요한 정보를 포함합니다.
CMINVOKECOMMANDINFOEX 바로 가기 메뉴 명령에 대한 확장된 정보를 포함합니다. 이 구조체는 유니코드 값을 사용할 수 있는 CMINVOKECOMMANDINFO의 확장 버전입니다.
DELEGATEITEMID 표준 ITEMIDLIST 구조 대신 대리자 폴더에서 사용됩니다.

DESKBANDINFO 밴드 개체에 대한 정보를 받습니다. 이 구조체는 사용되지 않는 IDeskBand::GetBandInfo 메서드와 함께 사용됩니다.
EXTRASEARCH IEnumExtraSearch 열거자 개체가 Shell Folder 개체에서 지원하는 검색 개체에 대한 정보를 반환하는 데 사용됩니다.
FOLDERSETTINGS 폴더 보기 정보를 포함합니다.
KNOWNFOLDER_DEFINITION 알려진 폴더의 세부 정보를 정의합니다.
PERSIST_FOLDER_TARGET_INFO 폴더 바로 가기의 대상 폴더 및 해당 특성을 지정합니다. 이 구조체는 IPersistFolder3::GetFolderTargetInfo 및 IPersistFolder3::InitializeEx에서 사용됩니다.
PREVIEWHANDLERFRAMEINFO 액셀러레이터 테이블 구조체. IPreviewHandlerFrame::GetWindowContext에서 사용됩니다.
SHDRAGIMAGE 끌기 이미지를 만드는 데 필요한 정보를 포함합니다.
SHELL_ITEM_RESOURCE 셸 항목 리소스를 정의합니다.
SMCSHCHANGENOTIFYSTRUCT 변경 알림에 대한 정보를 포함합니다. IShellMenuCallback::CallbackSM에서 사용됩니다.
SMDATA 메뉴 대역의 정보를 포함합니다.
SMINFO 메뉴 밴드의 항목에 대한 정보를 포함합니다.

SORTCOLUMN 폴더 뷰에 표시되는 열을 정렬하는 방법에 대한 정보를 저장합니다.
SV2CVW2_PARAMS IShellView2::CreateViewWindow2 메서드에 대한 매개 변수를 보유합니다.
THUMBBUTTON ITaskbarList3 인터페이스의 메서드에서 창의 축소판 그림 표현에 포함된 도구 모음에 사용되는 단추를 정의하는 데 사용됩니다.

열거형

_BROWSERFRAMEOPTIONS IBrowserFrameOptions::GetFrameOptions 메서드와 함께 사용됩니다.
_EXPCMDSTATE EXPCMDSTATE 값은 셸 항목의 명령 상태를 나타냅니다.
_EXPLORERPANESTATE 지정된 Windows 탐색기 창의 현재 상태를 가져오기 위해 IExplorerPaneVisibility::GetPaneState에서 사용하는 플래그를 나타냅니다.
_FILEOPENDIALOGOPTIONS 열기 또는 저장 대화 상자에서 사용할 수 있는 옵션 집합을 정의합니다.
_KF_DEFINITION_FLAGS 특정 알려진 폴더 동작을 지정하는 플래그입니다. KNOWNFOLDER_DEFINITION 구조와 함께 사용됩니다.
_KF_REDIRECT_FLAGS IKnownFolderManager::Redirect에서 리디렉션된 폴더에 대한 사용 권한 및 소유권과 같은 알려진 폴더 리디렉션의 세부 정보를 지정하는 데 사용하는 플래그입니다.

_KF_REDIRECTION_CAPABILITIES 알려진 폴더의 현재 리디렉션 기능을 지정하는 플래그입니다. IKnownFolder::GetRedirectionCapabilities에서 사용됩니다.
_NSTCITEMSTATE 트리 항목의 상태를 지정합니다. 이러한 값은 INameSpaceTreeControl 인터페이스의 메서드에서 사용됩니다.
_NSTCSTYLE 지정된 네임스페이스 트리 컨트롤의 특징을 설명합니다.
_PROPERTYUI_FLAGS 속성 기능을 지정합니다.
_SHCONTF 열거형에 포함된 항목의 유형을 결정합니다. 이러한 값은 IShellFolder::EnumObjects 메서드와 함께 사용됩니다.
_SHGDNF IShellFolder::GetDisplayNameOf 및 IShellFolder::SetNameOf 메서드와 함께 사용되는 값을 정의하여 해당 메서드에서 사용하는 파일 또는 폴더 이름의 형식을 지정합니다.
_SICHINTF 두 셸 항목을 비교하는 방법을 결정하는 데 사용됩니다. IShellItem::Compare는 이 열거형 형식을 사용합니다.
_SPBEGINF IActionProgress::Begin에서 사용되는 이러한 상수는 사용하거나 사용하지 않도록 설정할 특정 UI 작업을 지정합니다.
_SVGIO IFolderView::Items, IFolderView::ItemCount 및 IShellView::GetItemObject 메서드와 함께 사용하여 컬렉션의 항목을 제한하거나 제어합니다.
_SVSIF 적용할 선택 유형을 지정하기 위해 IFolderView, IFolderView2, IShellView 및 IShellView2에서 사용하는 플래그를 나타냅니다.

_TRANSFER_SOURCE_FLAGS ITransferSource 및 ITransferDestination 인터페이스의 메서드에서 파일 작업을 제어하는 데 사용됩니다.
APPLICATION_VIEW_ORIENTATION 창(앱 보기)에 대한 표시 방향 모드 집합을 정의합니다. IApplicationDesignModeSettings2::GetApplicationViewOrientation 및 IApplicationDesignModeSettings2::SetApplicationViewOrientation에서 사용됩니다.
APPLICATION_VIEW_SIZE_PREFERENCE 가능한 일반 창(앱 보기) 크기 기본 설정 집합을 정의합니다. ILaunchSourceViewSizePreference::GetSourceViewSizePreference 및 ILaunchTargetViewSizePreference::GetTargetViewSizePreference에서 사용됩니다.
APPLICATION_VIEW_STATE Windows 스토어 앱의 현재 보기 상태를 나타냅니다. IApplicationDesignModeSettings::SetApplicationViewState 및 IApplicationDesignModeSettings::IsApplicationViewStateSupported에서 사용됩니다.
ASSOCIATIONLEVEL 파일 이름 확장명 기본 연결의 원본을 지정합니다. IApplicationAssociationRegistration 인터페이스의 메서드에서 사용됩니다.
ASSOCIATIONTYPE 애플리케이션에 대한 연결 유형을 지정합니다. IApplicationAssociationRegistration 인터페이스의 메서드에서 사용됩니다.
ATTACHMENT_ACTION 사용자 확인 시 수행할 작업을 나타내기 위해 IAttachmentExecute::P rompt와 함께 사용할 플래그 집합을 제공합니다.
ATTACHMENT_PROMPT 표시할 프롬프트 UI 유형을 나타내기 위해 IAttachmentExecute::P rompt와 함께 사용할 플래그 집합을 제공합니다.
CATEGORYINFO_FLAGS CATEGORY_INFO 구조체에 사용할 플래그 집합을 제공합니다.

CATSORT_FLAGS 범주 데이터를 정렬하는 메서드를 지정합니다.
CM_ENUM_FLAGS IColumnManager 인터페이스의 멤버가 요청 중인 열 집합을 지정하는 데 사용되며, 현재 표시되는 열 집합 중 하나 또는 모두만 지정합니다.
CM_MASK IColumnManager::SetColumnInfo를 호출하는 동안 설정해야 하는 CM_COLUMNINFO 구조체의 값을 나타냅니다.
CM_SET_WIDTH_VALUE 너비 값을 픽셀 단위로 지정하고 기본값 및 자동 크기 조정에 대한 특별 지원을 포함합니다. CM_COLUMNINFO 구조를 통해 IColumnManager 인터페이스의 멤버가 사용합니다.
CM_STATE 열 상태 값을 지정합니다. CM_COLUMNINFO 구조를 통해 IColumnManager 인터페이스의 멤버가 사용합니다.
DATAOBJ_GET_ITEM_FLAGS SHGetItemFromDataObject 함수에서 소스 개체 처리와 관련된 옵션을 지정하는 데 사용되는 값입니다.
DEF_SHARE_ID ISharingConfigurationManager 인터페이스의 메서드에 의해 작동하는 폴더를 지정하는 값입니다.
DEFAULT_FOLDER_MENU_RESTRICTIONS . (DEFAULT_FOLDER_MENU_RESTRICTIONS)
DEFAULTSAVEFOLDERTYPE 기본 저장 위치를 지정합니다.
DESKTOP_WALLPAPER_POSITION 바탕 화면 배경 화면을 표시하는 방법을 지정합니다.
EXPLORER_BROWSER_FILL_FLAGS 이러한 플래그는 IExplorerBrowser::FillFromObject와 함께 사용됩니다.

EXPLORER_BROWSER_OPTIONS 이러한 플래그는 IExplorerBrowser::GetOptions 및 IExplorerBrowser::SetOptions와 함께 사용됩니다.
FDAP 목록 배치를 지정합니다.
FDE_OVERWRITE_RESPONSE 일반 파일 대화 상자를 사용하여 저장 작업 중에 덮어쓰기 요청에 대한 애플리케이션의 응답을 나타내기 위해 IFileDialogEvents::OnOverwrite 메서드에서 사용하는 값을 지정합니다.
FDE_SHAREVIOLATION_RESPONSE 파일을 열거나 저장할 때 발생하는 공유 위반에 대한 애플리케이션의 응답을 나타내기 위해 IFileDialogEvents::OnShareViolation 메서드에서 사용하는 값을 지정합니다.
FFFP_MODE 일치 조건을 설명합니다. IKnownFolderManager 인터페이스의 메서드에서 사용됩니다.
FILE_USAGE_TYPE 사용 중인 파일이 사용되는 방식을 나타내기 위해 IFileIsInUse::GetUsage에서 사용하는 상수입니다.
FOLDER_ENUM_MODE IObjectWithFolderEnumMode::GetMode 및 IObjectWithFolderEnumMode::SetMode 메서드에서 폴더의 표시 모드를 가져와서 설정하는 데 사용됩니다.
FOLDERFLAGS 폴더 보기 옵션을 지정하는 플래그 집합입니다. 플래그는 서로 독립적이며 모든 조합에서 사용할 수 있습니다.
FOLDERLOGICALVIEWMODE IFolderViewSettings::GetViewMode 및 ISearchFolderItemFactory::SetFolderLogicalViewMode에서 뷰 모드를 설명하는 데 사용됩니다.
FOLDERVIEWMODE 폴더 뷰 유형을 지정합니다.

KF_CATEGORY 알려진 폴더 시스템에 등록된 폴더를 분류할 수 있는 범주를 나타내는 값입니다.
LIBRARYFOLDERFILTER 폴더 항목을 필터링하는 옵션을 정의합니다.
LIBRARYMANAGEDIALOGOPTIONS SHShowManageLibraryUI에서 라이브러리를 저장할 때 이름 충돌을 처리하는 옵션을 정의하는 데 사용됩니다.
LIBRARYOPTIONFLAGS 라이브러리 옵션을 지정합니다.
LIBRARYSAVEFLAGS 라이브러리를 저장할 때 이름 충돌을 처리하는 옵션을 지정합니다.
MONITOR_APP_VISIBILITY 디스플레이에 Windows 스토어 앱 대신 데스크톱 창이 표시되는지 여부를 지정합니다.
NSTCFOLDERCAPABILITIES 트리 항목의 상태를 지정합니다. 이러한 값은 INameSpaceTreeControlFolderCapabilities 인터페이스의 메서드에서 사용됩니다.
NWMF INewWindowManager::EvaluateNewWindow에서 사용하는 플래그입니다. 이러한 값은 팝업 창을 표시할지 여부를 결정하는 요소입니다.
PACKAGE_EXECUTION_STATE . (PACKAGE_EXECUTION_STATE)
PDOPSTATUS 작업 상태 플래그를 제공합니다.
PLACEHOLDER_STATES 자리 표시자 파일에 있을 수 있는 상태를 지정합니다. System.FilePlaceholderStatus(PKEY_FilePlaceholderStatus) 속성을 통해 이 값을 검색합니다.

SHARE_ROLE 사용자 또는 공용 폴더에 할당된 액세스 권한을 지정합니다. CreateShare 및 GetSharePermissions에서 사용됩니다.
SIGDN IShellItem::GetDisplayName 및 SHGetNameFromIDList를 통해 검색할 항목의 표시 이름 형식을 요청합니다.
SPACTION IActionProgress 인터페이스를 사용하여 진행률을 사용자에게 표시해야 하는 수행 중인 작업에 대해 설명합니다.
SPTEXT IActionProgress 인터페이스에 제공되는 설명 텍스트의 형식을 지정합니다.
STPFLAG ITaskbarList4::SetTabProperties 메서드에서 탭 속성을 지정하는 데 사용됩니다.
SVUIA_STATUS 브라우저 보기의 상태를 설정하기 위해 IBrowserService2::_UIActivateView 메서드와 함께 사용됩니다.
SYNC_TRANSFER_STATUS System.SyncTransferStatus 속성에 사용할 수 있는 상태 값을 지정합니다.
tagDESKBANDCID 이러한 명령 ID는 IOleCommandTarget::Exec를 사용하여 밴드 개체의 컨테이너로 보낼 수 있습니다.
THUMBBUTTONFLAGS THUMBBUTTON에서 단추의 특정 상태 및 동작을 제어하는 데 사용됩니다.
THUMBBUTTONMASK THUMBBUTTON 구조체에서 유효한 데이터를 포함하는 해당 구조체의 멤버를 지정하는 데 사용됩니다.

shtypes.h 헤더

아티클2023. 06. 08.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

shtypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

COMDLG_FILTERSPEC 일반적으로 요소를 필터링하는 데 사용됩니다.
ITEMIDLIST 항목 식별자 목록을 포함합니다.
LOGFONTA ANSI(LOGFONTA) 구조체는 글꼴의 특성을 정의합니다. (LOGFONTA 구조체)
LOGFONTW LOGFONTW(유니코드) 구조체는 글꼴의 특성을 정의합니다. (LOGFONTW 구조체)
SHELLDETAILS Shell 폴더의 항목에 대한 자세한 정보를 보고합니다.
SHITEMID 항목 식별자를 정의합니다.
STRRET IShellFolder 인터페이스 메서드에서 반환된 문자열을 포함합니다.

열거형

DEVICE_SCALE_FACTOR 스푸핑된 디바이스 배율 인수를 백분율로 나타냅니다. IApplicationDesignModeSettings::SetApplicationViewState 및 IApplicationDesignModeSettings::IsApplicationViewStateSupported에서 사용됩니다.
인식 파일의 인식된 형식을 지정합니다. 이 상수 집합은 AssocGetPerceivedType 함수에서 사용됩니다.
SHCOLSTATE 속성을 처리하는 방법을 설명합니다. 이러한 값은 Shtypes.h에 정의되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

slpublic.h 헤더

아티클2023. 08. 26.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

slpublic.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

SLAcquireGenuineTicket SLS(소프트웨어 라이선싱 서버)에서 획득한 XrML 정품 티켓을 가져옵니다.
SLActivateProduct SLS(소프트웨어 라이선스 서버)에서 사용 라이선스를 획득합니다.
SLClose SLC(소프트웨어 라이선싱 클라이언트) 컨텍스트 핸들을 닫습니다.
SLConsumeRight 애플리케이션이 로컬로 저장된 라이선스에 대한 권한을 행사할 수 있도록 합니다.
SLDepositMigrationBlob SLGatherMigrationBlob 함수를 사용하여 이전에 수집하고 수집한 라이선스 정보를 입금합니다.
SLDepositOfflineConfirmationId 오프라인 정품 인증을 위해 IID(설치 ID) 및 CID(확인 ID)를 입금합니다. (SLDepositOfflineConfirmationId)
SLDepositOfflineConfirmationIdEx 오프라인 정품 인증을 위해 IID(설치 ID) 및 CID(확인 ID)를 입금합니다. (SLDepositOfflineConfirmationIdEx)
SLFireEvent 지정된 이벤트를 등록된 수신기에 보냅니다.

SLGatherMigrationBlob 제공된 파일 핸들에 대한 라이선스 정보를 수집합니다. 이 라이선스 정보는 나중에 SLDepositMigrationBlob 함수를 사용하여 적용하거나 입금할 수 있습니다.
SLGenerateOfflineInstallationId IID(설치 ID)를 생성합니다.
SLGenerateOfflineInstallationIdEx IID(설치 ID)를 생성합니다.
SLGetApplicationInformation 지정된 애플리케이션에 대한 정보를 가져옵니다.
SLGetApplicationPolicy SLPersistApplicationPolicies 함수와 함께 저장되고 SLLoadApplicationPolicies 함수를 사용하여 로드된 집합에서 정책을 쿼리합니다.
SLGetAuthenticationResult 인증 결과를 가져옵니다.
SLGetGenuineInformation Windows 컴퓨터의 정품 상태에 대한 정보를 가져옵니다.
SLGetGenuineInformationEx Windows 컴퓨터의 정품 상태 대한 정보를 지정합니다. (SLGetGenuineInformationEx)
SLGetInstalledProductKeyIds 이 함수는 지정된 제품 SKU ID와 연결된 제품 키 ID 목록을 반환합니다.
SLGetLicense 라이선스 파일 BLOB을 반환합니다.
SLGetLicenseFileId 라이선스 BLOB이 이미 설치되어 있는지 확인합니다.
SLGetLicenseInformation 지정된 라이선스 정보를 가져옵니다.

SLGetLicensingStatusInformation 지정된 애플리케이션 또는 SKU의 라이선스 상태 가져옵니다.
SLGetPKeyId 제품과 연결된 등록된 제품 키 ID를 가져옵니다.
SLGetPKeyInformation 지정된 제품 키의 정보를 가져옵니다.
SLGetPolicyInformation 오른쪽이 성공적으로 사용된 후 정책 정보를 가져옵니다. (SLGetPolicyInformation)
SLGetPolicyInformationDWORD 오른쪽이 성공적으로 사용된 후 정책 정보를 가져옵니다. (SLGetPolicyInformationDWORD)
SLGetProductSkuInformation 지정된 제품 SKU에 대한 정보를 가져옵니다.
SLGetReferralInformation 지정된 제품에 대한 조회 정보를 가져옵니다.
SLGetServerStatus 지정된 URL 및 RequestType에 따라 서버 상태 확인합니다.
SLGetServiceInformation 전역 데이터 정보를 가져옵니다.
SLGetSLIDList 입력 쿼리 ID 형식 및 ID 값에 따라 SLID 목록을 가져옵니다.
SLGetWindowsInformation 소프트웨어 구성 요소의 라이선스 정책에서 이름-값 쌍의 값 부분을 검색합니다.
SLGetWindowsInformationDWORD 소프트웨어 구성 요소의 라이선스 정책에서 이름-값 쌍의 DWORD 값 부분을 검색합니다.
SLInstallLicense

지정된 라이선스를 저장하고 라이선스 파일 ID를 반환합니다.
SLInstallProofOfPurchase 제품 키를 SL에 등록합니다.
SLInstallProofOfPurchaseEx 제품 키를 SL에 등록합니다.
SLIsGenuineLocal 지정된 애플리케이션이 정품 Windows 설치인지 확인합니다.
SLIsGenuineLocalEx 지정된 애플리케이션 설치가 정품인지 확인합니다.
SLLoadApplicationPolicies SLGetApplicationPolicy 함수에서 사용할 SLPersistApplicationPolicies 함수를 사용하여 설정된 애플리케이션 정책을 로드합니다.
SLOpen SLC(소프트웨어 라이선스 클라이언트)를 초기화하고 SLC를 SLS(소프트웨어 라이선스 서비스)에 연결합니다.
SLPersistApplicationPolicies 빠른 정책 액세스를 위해 현재 사용된 정책을 디스크에 저장합니다.
SLPersistRTSPayloadOverride 온라인 및 전화 정품 인증을 위해 지정된 제품과 정보를 연결합니다.
SLQueryLicenseValueFromApp 지정된 구성 요소 정책의 값을 가져옵니다.
SLReArm 이 함수는 애플리케이션 활성화를 다시 정렬합니다.
SLRegisterEvent SL 서비스에 이벤트를 등록합니다.
SLSetAuthenticationData

인증 데이터를 설정합니다.
SLSetCurrentProductKey 현재 제품 키를 이전에 설치된 제품 키로 설정합니다.
SLSetGenuineInformation Windows 컴퓨터의 정품 상태 대한 정보를 지정합니다. (SLSetGenuineInformation)
SLUninstallLicense 라이선스 파일 ID 및 대상 사용자 옵션으로 지정된 라이선스를 제거합니다.
SLUninstallProofOfPurchase 제품 키 정보를 등록 취소합니다.
SLUnloadApplicationPolicies SLLoadApplicationPolicies 함수에서 반환된 정책 컨텍스트 핸들을 해제합니다.
SLUnregisterEvent SL 서비스에서 등록된 이벤트를 등록 취소합니다.

구조체

SL_ACTIVATION_INFO_HEADER 제품 정품 인증 정보를 지정합니다.
SL_AD_ACTIVATION_INFO 라이선스의 소매 또는 Active Directory 전화 활성화에 사용되는 정보를 지정합니다.
SL_LICENSING_STATUS 라이선스 상태 나타냅니다. (SL_LICENSING_STATUS)
SL_NONGENUINE_UI_OPTIONS SLIsGenuineLocal 함수가 설치가 정품이 아님을 나타내는 경우 대화 상자를 표시하는 애플리케이션을 지정합니다.

열거형

SL_ACTIVATION_TYPE 라이선스에 대한 오프라인 활성화 유형을 나타냅니다.
SL_GENUINE_STATE 애플리케이션 설치의 상태를 지정합니다.
SLDATATYPE SLGetWindowsInformation 함수에서 반환되는 버퍼의 데이터 형식을 지정합니다.
슬라이드타입 소프트웨어 라이선스 ID의 유형을 나타냅니다.
SLLICENSINGSTATUS 라이선스 상태 나타냅니다. (SLLICENSINGSTATUS)
SLREFERRALTYPE SLGetReferralInformation 함수를 사용하여 쿼리할 수 있는 정보 유형을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

smbclnt.h 헤더

아티클2023. 08. 26.

이 헤더는 Windows 클러스터링에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 클러스터링

smbclnt.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

RegisterAppInstance 프로세스에 대한 AppInstance ID를 등록합니다.
SetAppInstanceCsvFlags 애플리케이션 instance 연결에 영향을 주는 플래그를 설정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

snmp.h 헤더

아티클2023. 08. 26.

이 헤더는 단순 네트워크 관리 프로토콜에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Simple Network Management Protocol

snmp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

SnmpExtensionClose Microsoft SNMP 서비스는 SnmpExtensionClose 함수를 호출하여 SNMP 확장 에이전트가 리소스의 할당을 취소하고 작업을 종료하도록 요청합니다. 이 함수는 SNMP 확장 에이전트 API의 요소입니다.
SnmpExtensionInit Microsoft SNMP 서비스는 SnmpExtensionInit 함수를 호출하여 SNMP 확장 에이전트 DLL을 초기화합니다. 이 함수는 SNMP 확장 에이전트 API의 요소입니다.
SnmpExtensionInitEx Microsoft SNMP 서비스는 SnmpExtensionInitEx 함수를 호출하여 SNMP 확장 에이전트가 지원하는 MIB(추가 관리 정보 기반) 하위 트리를 식별합니다. 이 함수는 SNMP 확장 에이전트 API의 요소입니다.
SnmpExtensionMonitor Microsoft SNMP 서비스는 SnmpExtensionMonitor 함수를 호출하여 SNMP 확장 에이전트에 서비스의 내부 카운터 및 매개 변수에 대한 보기를 제공합니다. 이 함수는 SNMP 확장 에이전트 API의 요소입니다.
SnmpExtensionQuery Microsoft SNMP 서비스는 SnmpExtensionQuery 함수를 호출하여 SNMP 확장 에이전트의 등록된 MIB 하위 트리 중 하나 이상의 변수를 포함하는 SNMP 요청을 resolve. 이 함수는 SNMP 확장 에이전트 API의 요소입니다.
SnmpExtensionQueryEx Microsoft SNMP 서비스는 SnmpExtensionQueryEx 함수를 호출하여 SNMP 확장 에이전트에 의해

등록된 하나 이상의 MIB 하위 트리에서 변수를 지정하는 SNMP 요청을 처리합니다. 이 함수는 SNMP 확장 에이전트 API의 요소입니다.
SnmpExtensionTrap Microsoft SNMP 서비스는 SnmpExtensionTrap 함수를 호출하여 서비스가 SNMP 확장 에이전트에 대한 트랩을 생성하는 데 필요한 정보를 검색합니다.
SnmpSvcGetUptime SnmpSvcGetUptime 함수는 SNMP 서비스가 실행되고 있는 센티미터 수를 검색합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpSvcSetLogLevel SnmpSvcSetLogLevel 함수는 SnmpUtilDbgPrint 함수를 사용하여 SNMP 서비스 및 SNMP 확장 에이전트에서 디버그 출력의 세부 수준을 조정합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpSvcSetLogType SnmpSvcSetLogType 함수는 SnmpUtilDbgPrint 함수를 사용하여 SNMP 서비스 및 SNMP 확장 에이전트의 디버그 출력 대상을 조정합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilAsnAnyCpy SnmpUtilAsnAnyCpy 함수는 pAnySrc 매개 변수가 가리키는 변수를 pAnyDst 매개 변수에 복사합니다. 함수는 대상의 복사본에 필요한 메모리를 할당합니다. SnmpUtilAsnAnyCpy 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilAsnAnyFree SnmpUtilAsnAnyFree 함수는 지정된 AsnAny 구조에 할당된 메모리를 해제합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilDbgPrint SnmpUtilDbgPrint 함수를 사용하면 SNMP 서비스에서 출력을 디버깅할 수 있습니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilIdsToA SnmpUtilIdsToA 함수는 OID(개체 식별자)를 null로 끝나는 문자열로 변환합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilMemAlloc SnmpUtilMemAlloc 함수는 프로세스 힙에서 동적 메모리를 할당합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.

SnmpUtilMemFree SnmpUtilMemFree 함수는 지정된 메모리 개체를 해제합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilMemReAlloc SnmpUtilMemReAlloc 함수는 지정된 메모리 개체의 크기를 변경합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilOctetsCmp SnmpUtilOctetsCmp 함수는 두 개의 옥텟 문자열을 비교합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilOctetsCpy SnmpUtilOctetsCpy 함수는 pOctetsSrc 매개 변수가 가리키는 변수를 pOctetsDst 매개 변수가 가리키는 변수에 복사합니다.
SnmpUtilOctetsFree SnmpUtilOctetsFree 함수는 지정된 8진수 문자열에 할당된 메모리를 해제합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilOctetsNCmp SnmpUtilOctetsNCmp 함수는 두 개의 옥텟 문자열을 비교합니다. 함수는 nChars 매개 변수로 지정된 하위 identifier 수에 도달할 때까지 문자열의 하위 identifier를 비교합니다. SnmpUtilOctetsNCmp는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilOidAppend SnmpUtilOidAppend 함수는 원본 개체 식별자를 대상 개체 식별자에 추가합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilOidCmp SnmpUtilOidCmp 함수는 두 개체 식별자를 비교합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilOidCpy SnmpUtilOidCpy 함수는 pOidSrc 매개 변수가 가리키는 변수를 pOidDst 매개 변수에 복사하여 대상의 복사본에 필요한 메모리를 할당합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilOidFree

SnmpUtilOidFree 함수는 지정된 개체 식별자에 할당된 메모리를 해제합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilOidNCmp SnmpUtilOidNCmp 함수는 두 개체 식별자를 비교합니다.
SnmpUtilOidToA SnmpUtilOidToA 함수는 OID(개체 식별자)를 null로 끝나는 문자열로 변환합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilPrintAsnAny SnmpUtilPrintAsnAny 함수는 Any 매개 변수의 값을 표준 출력에 출력합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilPrintOid SnmpUtilPrintOid 함수는 지정된 OID(개체 식별자)의 형식을 지정하고 결과를 표준 출력 디바이스에 출력합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilVarBindCpy SnmpUtilVarBindCpy 함수는 지정된 SnmpVarBind 구조를 복사하고 대상 구조에 필요한 모든 메모리를 할당합니다. SnmpUtilVarBindCpy 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilVarBindFree SnmpUtilVarBindFree 함수는 SnmpVarBind 구조에 할당된 메모리를 해제합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilVarBindListCpy SnmpUtilVarBindListCpy 함수는 지정된 SnmpVarBindList 구조를 복사하고 대상의 복사본에 필요한 메모리를 할당합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.
SnmpUtilVarBindListFree SnmpUtilVarBindListFree 함수는 SnmpVarBindList 구조에 할당된 메모리를 해제합니다. 이 함수는 SNMP 유틸리티 API의 요소입니다.

구조체

AsnAny AsnAny 구조체에는 SNMP 변수 형식 및 값이 포함됩니다. 이 구조체는 많은 SNMP 함수에서 매개 변수로 사용되는 SnmpVarBind 구조체의 멤버입니다. 이 구조체는 WinSNMP API 함수에서 사용되지 않습니다.
AsnObjectIdentifier AsnObjectIdentifier 구조체는 개체 식별자를 나타냅니다. 이 구조체는 여러 SNMP 함수에서 사용됩니다. 이 구조체는 WinSNMP API 함수에서 사용되지 않습니다.
AsnOctetString AsnOctetString 구조체에는 일반적으로 바이트인 옥텟 수량이 포함됩니다. 이 구조체는 여러 SNMP 함수에서 사용됩니다. 이 구조체는 WinSNMP API 함수에서 사용되지 않습니다.
SnmpVarBind SnmpVarBind 구조체는 SNMP 변수 바인딩을 나타냅니다. 이 구조체는 여러 SNMP 함수에서 사용됩니다. 이 구조체는 WinSNMP API 함수에서 사용되지 않습니다.
SnmpVarBindList SnmpVarBindList 구조체는 SNMP 변수 바인딩 목록을 나타냅니다. 이 구조체는 여러 SNMP 함수에서 사용됩니다. 이 구조체는 WinSNMP API 함수에서 사용되지 않습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

socketapi.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows 소켓 2에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 소켓 2

socketapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

SetSocketMediaStreamingMode 서비스 품질이 필요한 스트리밍 미디어를 전송하는 데 네트워크를 사용할지 여부를 나타냅니다.

spatialaudioclient.h 헤더

아티클2023. 03. 16.

이 헤더는 핵심 오디오 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 핵심 오디오 API

spatialaudioclient.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAudioFormatEnumerator 지원되는 오디오 형식의 목록을 제공합니다. 가장 선호하는 형식은 목록의 첫 번째 형식입니다. ISpatialAudioClient::GetSupportedAudioObjectFormatEnumerator를 호출하여 이 인터페이스에 대한 참조를 가져옵니다.
ISpatialAudioClient ISpatialAudioClient 인터페이스를 사용하면 클라이언트가 3D 공간의 위치에서 오디오를 내보내는 오디오 스트림을 만들 수 있습니다.
ISpatialAudioClient2 **ISpatialAudioClient2** 인터페이스는 ISpatialAudioClient에서 상속되며 큰 오디오 버퍼 오프로드에 대한 지원을 쿼리하는 메서드를 추가합니다.
ISpatialAudioObject 사용자를 기준으로 3D 공간의 위치에서 렌더링할 오디오 데이터를 제공하는 개체를 나타냅니다.
ISpatialAudioObjectBase 사용자를 기준으로 3D 공간의 위치에서 렌더링할 오디오 데이터를 제공하는 개체를 나타내는 기본 인터페이스입니다.
ISpatialAudioObjectRenderStream 스트림 시작, 중지 및 재설정을 포함하여 공간 오디오 개체 렌더링 스트림을 제어하는 메서드를 제공합니다.
ISpatialAudioObjectRenderStreamBase 스트림 시작, 중지 및 재설정을 포함하여 공간 오디오 개체 렌더링 스트림을 제어하는 메서드를 제공하는 기본 인터페이스입니다.

ISpatialAudioObjectRenderStreamNotify 공간 오디오 클라이언트가 ISpatialAudioObjectRenderStream의 상태 변경에 응답하도록 알림을 제공합니다.

구조체

SpatialAudioClientActivationParams 공간 오디오 렌더링 스트림에 대한 선택적 활성화 매개 변수를 나타냅니다. ISpatialAudioClient 인터페이스를 활성화할 때 이 구조를 ActivateAudioInterfaceAsync에 전달합니다.
SpatialAudioObjectRenderStreamActivationParams 공간 오디오 렌더링 스트림에 대한 활성화 매개 변수를 나타냅니다. 스트림을 활성화할 때 ISpatialAudioClient::ActivateSpatialAudioStream에 이 구조를 전달합니다.
SpatialAudioObjectRenderStreamActivationParams2 공간 오디오 렌더링 스트림에 대한 활성화 매개 변수를 나타내며 스트림 옵션을 지정할 수 있는 기능으로 SpatialAudioObjectRenderStreamActivationParams를 확장합니다.

열거형

AudioObjectType ISpatialAudioObject의 형식을 지정합니다.
SPATIAL_AUDIO_STREAM_OPTIONS ActivateSpatialAudioStream 호출에 대한 오디오 스트림 옵션을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

spatialaudiohrtf.h 헤더

아티클2023. 08. 26.

이 헤더는 핵심 오디오 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 핵심 오디오 API

spatialaudiohrtf.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ISpatialAudioObjectForHrtf HRTF(헤드 상대 전송 함수)를 기준으로 3D 공간의 위치에서 렌더링할 오디오 데이터를 제공하는 개체를 나타냅니다.
ISpatialAudioObjectRenderStreamForHrtf 스트림 시작, 중지 및 초기화를 포함하여 Hrtf 공간 오디오 개체 렌더링 스트림을 제어하는 메서드를 제공합니다.

구조체

SpatialAudioHrtfActivationParams ISpatialAudioRenderStreamForHrtf에 대한 활성화 매개 변수를 지정합니다.
SpatialAudioHrtfActivationParams2 공간 오디오 렌더링 스트림에 대한 활성화 매개 변수를 나타내며 스트림 옵션을 지정할 수 있는 기능으로 SpatialAudioHrtfActivationParams를 확장합니다.
SpatialAudioHrtfDirectivity ISpatialAudioObjectForHrtf에 대한 전방향 모델을 나타냅니다. 전방향 방출은 크기 조정 필드의 값을 기반으로 형식 필드에 지정된 직접성 모델과 선형으로 보간됩니다.
SpatialAudioHrtfDirectivityCardioid ISpatialAudioObjectForHrtf에 대한 심장 모양의 직접성 모델을 나타냅니다.

SpatialAudioHrtfDirectivityCone ISpatialAudioObjectForHrtf에 대한 원뿔 모양의 직접성 모델을 나타냅니다.
SpatialAudioHrtfDirectivityUnion ISpatialAudioObjectForHrtf에 대한 공간 오디오 직접성 모델을 정의합니다.
SpatialAudioHrtfDistanceDecay ISpatialAudioObjectForHrtf의 위치에서 수신기 위치까지의 거리에 적용되는 감쇠 모델을 나타냅니다.

열거형

SpatialAudioHrtfDirectivityType ISpatialAudioObjectForHrtf에서 소리를 내보낸 셰이프를 지정합니다.
SpatialAudioHrtfDistanceDecayType ISpatialAudioObjectForHrtf의 위치에서 수신기 위치까지의 거리에 적용된 감쇠 유형을 지정합니다.
SpatialAudioHrtfEnvironmentType ISpatialAudioObjectForHrtf에 대해 오디오가 처리될 때 시뮬레이션되는 음향 환경의 유형을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

spatialaudiometadata.h 헤더

아티클2023. 08. 26.

이 헤더는 핵심 오디오 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 핵심 오디오 API

spatialaudiometadata.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ISpatialAudioMetadataClient ISpatialAudioMetadataItems, ISpatialAudioMetadataWriter, ISpatialAudioMetadataReader 및 ISpatialAudioMetadataCopier 개체를 만들기 위한 클래스 팩터리를 제공합니다.
ISpatialAudioMetadataCopier 원본 SpatialAudioMetadataItems에서 대상 SpatialAudioMetadataItems로 메타데이터 항목의 전체 또는 하위 집합을 복사하는 메서드를 제공합니다.
ISpatialAudioMetadataItems 공간 오디오 메타데이터 항목의 버퍼를 나타냅니다.
ISpatialAudioMetadataItemsBuffer 데이터의 현재 위치 스토리지를 위해 SpatialAudioMetadataItems에 버퍼를 연결하는 메서드를 제공합니다.
ISpatialAudioMetadataReader ISpatialAudioMetadataItems 개체에서 공간 오디오 메타데이터 항목 및 항목 명령 값 쌍을 추출하는 메서드를 제공합니다.
ISpatialAudioMetadataWriter 해당 오디오 프레임 범위 내에 배치된 공간 오디오 메타데이터 항목을 저장하는 메서드를 제공합니다.
ISpatialAudioObjectForMetadataCommands 공간 오디오에 대한 메타데이터 명령을 작성하는 데 사용됩니다.

ISpatialAudioObjectForMetadataItems 프레임 정확 배치를 사용하여 버퍼당 여러 메타데이터 항목이 필요한 애플리케이션에 대한 공간 오디오 메타데이터를 작성하는 데 사용됩니다.
ISpatialAudioObjectRenderStreamForMetadata 스트림 시작, 중지 및 다시 설정을 포함하여 메타데이터에 대한 공간 오디오 개체 렌더링 스트림을 제어하는 메서드를 제공합니다.

구조체

SpatialAudioMetadataItemsInfo ISpatialAudioMetadataItems 개체에 대한 정보를 제공합니다. GetInfo를 호출하여 이 구조체의 복사본을 가져옵니다.
SpatialAudioObjectRenderStreamForMetadataActivationParams 메타데이터에 대한 공간 오디오 렌더링 스트림에 대한 활성화 매개 변수를 나타냅니다. 스트림을 활성화할 때 이 구조를 ISpatialAudioClient::ActivateSpatialAudioStream에 전달합니다.
SpatialAudioObjectRenderStreamForMetadataActivationParams2 스트림 옵션을 지정할 수 있는 기능으로 SpatialAudioObjectRenderStreamForMetadataActivationParams를 확장하여 메타데이터에 대한 공간 오디오 렌더링 스트림에 대한 활성화 매개 변수를 나타냅니다.

열거형

SpatialAudioMetadataCopyMode ISpatialAudioMetadataCopier::CopyMetadataForFrames를 호출할 때 사용되는 복사 모드를 지정합니다.
SpatialAudioMetadataWriterOverflowMode ISpatialAudioMetadataWriter가 클라이언트를 초기화할 때 지정한 것보다 더 많은 항목을 메타데이터 버퍼에 쓰려고 할 때 원하는 동작을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

spatialinteractionmanagerinterop.h 헤더

아티클2023. 08. 26.

이 헤더는 Windows Mixed Reality 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Mixed Reality

spatialinteractionmanagerinterop.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ISpatialInteractionManagerInterop WinRT SpatialInteractionManager 개체와의 상호 운용성을 사용하도록 설정하고, 손, 모션 컨트롤러 및 시스템 음성 명령의 사용자 입력에 액세스하기 위해 SpatialInteractionManager 멤버에 대한 액세스를 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

spellcheck.h 헤더

아티클2023. 03. 08.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 애플리케이션에 대한 국제화
- 맞춤법 검사기 API

spellcheck.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumSpellingError 맞춤법 오류의 열거형입니다.
IOptionDescription 맞춤법 검사기 옵션에 대한 설명을 나타냅니다.
ISpellChecker 특정 언어에 대한 특정 맞춤법 검사기를 나타냅니다.
ISpellChecker2 추가된 단어 사전 또는 무시 목록에서 단어를 제거하는 기능이 추가된 특정 언어의 특정 맞춤법 검사기를 나타냅니다.
ISpellCheckerChangedEventHandler 호출자가 맞춤법 검사기의 상태가 변경되었다는 알림에 대한 처리기를 만들 수 있습니다.
ISpellCheckerFactory 맞춤법 검사기(ISpellChecker)를 인스턴스화하고 지원되는 언어를 결정하는 기능을 제공하는 팩터리입니다.
ISpellingError 맞춤법 오류에 대한 정보를 제공합니다.
IUserDictionariesRegistrar 사용자 사전의 등록을 관리합니다.

열거형

CORRECTIVE_ACTION 맞춤법 오류에 대해 수행할 수정 작업의 유형을 식별합니다.
WORDLIST_TYPE 맞춤법 검사기에서 사용하는 단어 목록 유형 중 하나를 식별합니다.

spellcheckprovider.h 헤더

아티클2023. 03. 08.

이 헤더는 Windows 애플리케이션에 대한 Internationalization에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 애플리케이션에 대한 국제화

spellcheckprovider.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IComprehensiveSpellCheckProvider 공급자가 필요에 따라 보다 포괄적인 맞춤법 검사 기능을 지원할 수 있습니다.
ISpellCheckProvider 맞춤법 검사 인프라에서 사용할 특정 언어에 대한 특정 맞춤법 검사기 공급자를 나타냅니다.
ISpellCheckProviderFactory 맞춤법 검사기(ISpellCheckProvider)를 인스턴스화하고 지원되는 언어를 결정하는 기능을 제공하는 팩터리입니다.

sporder.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows 소켓 2에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 소켓 2

sporder.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WSCWriteNameSpaceOrder 사용 가능한 Windows 소켓(Winsock) 2 네임스페이스 공급자의 순서를 변경합니다. 네임스페이스 공급자의 순서는 이름 확인을 위해 열거하거나 쿼리할 때 네임스페이스의 우선 순위를 결정합니다.
WSCWriteNameSpaceOrder32 32비트 카탈로그에서 사용 가능한 Windows 소켓(Winsock) 2 네임스페이스 공급자의 순서를 변경합니다.
WSCWriteProviderOrder 사용 가능한 전송 공급자의 순서를 다시 지정하는 데 사용됩니다.
WSCWriteProviderOrder32 사용 가능한 32비트 전송 공급자의 순서를 다시 지정하는 데 사용됩니다.

srpapi.h 헤더

아티클2023. 08. 26.

이 헤더는 WIP(Windows Information Protection)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- WIP(Windows Information Protection)

srpapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

SrpCloseThreadNetworkContext SrpCreateThreadNetworkContext에서 선택적으로 반환되었을 수 있는 원래 컨텍스트로 스레드를 다시 복원합니다.
SrpCreateThreadNetworkContext 엔터프라이즈 ID를 현재 스레드의 데이터 컨텍스트로 설정합니다. 프로세스 컨텍스트에 동일한 엔터프라이즈 ID가 이미 있는 경우에만 허용됩니다. 필요에 따라 기존 스레드 토큰을 반환합니다.
SrpDisablePermissiveModeFileEncryption 현재 스레드에서 파일 암호화에 대한 허용 모드를 사용하지 않도록 설정합니다.
SrpDoesPolicyAllowAppExecution 소프트웨어 제한 정책에 따라 패키지된 앱을 실행할 수 있는지 여부를 평가합니다.
SrpEnablePermissiveModeFileEncryption 현재 스레드 및 이 스레드가 만들거나 게시하는 모든 스레드에서 파일 암호화에 대한 허용 모드를 사용하도록 설정합니다.
SrpGetEnterpriseIds 지정된 토큰에 대한 엔터프라이즈 식별자 목록을 가져옵니다.
SrpGetEnterprisePolicy 정책은 Windows 10 버전 1607에 적용할 수 있습니다.앱의 엔터프라이즈 정책에 대한 정보를 가져옵니다.

SrpIsTokenService 서비스가 토큰 서비스인지 여부를 식별합니다.
SrpSetTokenEnterpriseId 토큰에 대한 데이터 의도를 설정합니다. 호출자 프로세스는 제공된 엔터프라이즈 ID에 대해 엔터프라이즈가 허용되어야 합니다.

열거형

ENTERPRISE_DATA_POLICIES 앱이 WIP(Windows Information Protection)에 대해 인식되는지 여부와 앱이 정책에 의해 관리되는지 여부를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

srrestoreptapi.h 헤더

아티클2023. 08. 26.

이 헤더는 시스템 복원에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 복원

srrestoreptapi.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

SRRemoveRestorePoint 지정된 복원 지점을 삭제합니다.
SRSetRestorePointA 시스템 복원에서 복원 지점을 만들 수 있도록 변경 내용 집합의 시작과 끝을 지정합니다. (ANSI)
SRSetRestorePointW 시스템 복원에서 복원 지점을 만들 수 있도록 변경 내용 집합의 시작과 끝을 지정합니다. (유니코드)

구조체

RESTOREPOINTINFOA SRSetRestorePoint 함수에서 사용하는 정보를 포함합니다. (ANSI)
RESTOREPOINTINFOW SRSetRestorePoint 함수에서 사용하는 정보를 포함합니다. (유니코드)
STATEMGRSTATUS SRSetRestorePoint 함수에서 사용하는 상태 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

sspi.h 헤더

아티클2023. 08. 26.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

sspi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AcceptSecurityContext 전송 애플리케이션의 서버 구성 요소가 서버와 원격 클라이언트 간에 보안 컨텍스트를 설정할 수 있도록 합니다.
AcquireCredentialsHandleA AcquireCredentialsHandle(CredSSP) 함수는 보안 주체의 기존 자격 증명에 대한 핸들을 획득합니다. (ANSI)
AcquireCredentialsHandleW AcquireCredentialsHandle(CredSSP) 함수는 보안 주체의 기존 자격 증명에 대한 핸들을 획득합니다. (유니코드)
AddCredentialsA ANSI(AddCredentialsA)는 자격 증명 목록에 자격 증명을 추가합니다.
AddCredentialsW AddCredentialsW(유니코드)는 자격 증명 목록에 자격 증명을 추가합니다.
AddSecurityPackageA Microsoft Negotiate에서 지원하는 공급자 목록에 보안 지원 공급자를 추가합니다. (ANSI)
AddSecurityPackageW Microsoft Negotiate에서 지원하는 공급자 목록에 보안 지원 공급자를 추가합니다. (유니코드)
ApplyControlToken 보안 컨텍스트에 컨트롤 토큰을 적용하는 방법을 제공합니다.

ChangeAccountPasswordA 지정된 보안 지원 공급자를 사용하여 Windows 도메인 계정의 암호를 변경합니다. (ANSI)
ChangeAccountPasswordW 지정된 보안 지원 공급자를 사용하여 Windows 도메인 계정의 암호를 변경합니다. (유니코드)
CompleteAuthToken 인증 토큰을 완료합니다. (CompleteAuthToken)
DecryptMessage 다이제스트를 사용하여 메시지의 암호를 해독합니다.
DelegateSecurityContext 지정된 서버에 보안 컨텍스트를 위임합니다.
DeleteSecurityContext InitializeSecurityContext(일반) 함수 또는 AcceptSecurityContext(일반) 함수에 대한 이전 호출로 시작된 지정된 보안 컨텍스트와 연결된 로컬 데이터 구조를 삭제합니다.
DeleteSecurityPackageA Microsoft Negotiate에서 지원하는 공급자 목록에서 보안 지원 공급자를 삭제합니다. (ANSI)
DeleteSecurityPackageW Microsoft Negotiate에서 지원하는 공급자 목록에서 보안 지원 공급자를 삭제합니다. (유니코드)
EncryptMessage 다이제스트를 사용하여 개인 정보를 제공하도록 메시지를 암호화합니다.
EnumerateSecurityPackagesA 클라이언트에서 사용할 수 있는 보안 패키지에 대한 정보를 제공하는 SecPkgInfo 구조체의 배열을 반환합니다. (ANSI)
EnumerateSecurityPackagesW 클라이언트에서 사용할 수 있는 보안 패키지에 대한 정보를 제공하는 SecPkgInfo 구조체의 배열을 반환합니다. (유니코드)
ExportSecurityContext

ExportSecurityContext 함수는 나중에 ImportSecurityContext를 호출하여 다른 프로세스로 가져올 수 있는 보안 컨텍스트의 직렬화된 표현을 만듭니다.
FreeContextBuffer 보안 패키지 함수의 호출자가 보안 패키지에서 할당한 메모리 버퍼를 해제할 수 있도록 합니다.
FreeCredentialsHandle 보안 시스템에 자격 증명이 더 이상 필요하지 않음을 알 수 있습니다.
ImpersonateSecurityContext 서버가 AcceptSecurityContext(일반) 또는 QuerySecurityContextToken을 호출하여 이전에 가져온 토큰을 사용하여 클라이언트를 가장할 수 있습니다.
ImportSecurityContextA 보안 컨텍스트를 가져옵니다. 보안 컨텍스트는 ExportSecurityContext에 대한 이전 호출을 통해 ImportSecurityContext를 호출하는 프로세스로 내보내야 합니다. (ANSI)
ImportSecurityContextW 보안 컨텍스트를 가져옵니다. 보안 컨텍스트는 ExportSecurityContext에 대한 이전 호출을 통해 ImportSecurityContext를 호출하는 프로세스로 내보내야 합니다. (유니코드)
InitializeSecurityContextA 자격 증명 핸들에서 클라이언트 쪽 아웃바운드 보안 컨텍스트를 시작합니다. (ANSI)
InitializeSecurityContextW 자격 증명 핸들에서 클라이언트 쪽 아웃바운드 보안 컨텍스트를 시작합니다. (유니코드)
InitSecurityInterfaceA InitSecurityInterface 함수는 SSPI 디스패치 테이블에 대한 포인터를 반환합니다. 이 함수를 사용하면 클라이언트가 인터페이스 구현에 직접 바인딩하지 않고 SSPI를 사용할 수 있습니다. (ANSI)
InitSecurityInterfaceW InitSecurityInterface 함수는 SSPI 디스패치 테이블에 대한 포인터를 반환합니다. 이 함수를 사용하면 클라이언트가 인터페이스 구현에 직접 바인딩하지 않고 SSPI를 사용할 수 있습니다. (유니코드)
MakeSignature 메시지의 암호화 체크섬을 생성하고 메시지 손실 또는 삽입을 방지하기 위한 시퀀싱 정보도 포함합니다.

QueryContextAttributesA 전송 애플리케이션이 보안 컨텍스트의 특정 특성에 대해 CredSSP(자격 증명 보안 지원 공급자) 보안 패키지를 쿼리할 수 있습니다. (ANSI)
QueryContextAttributesExA ANSI(QueryContextAttributesExA) 함수(sspi.h)를 사용하면 전송 애플리케이션이 보안 컨텍스트의 특정 특성에 대한 보안 패키지를 쿼리할 수 있습니다.
QueryContextAttributesExW QueryContextAttributesExW(유니코드) 함수(sspi.h)를 사용하면 전송 애플리케이션이 보안 컨텍스트의 특정 특성에 대한 보안 패키지를 쿼리할 수 있습니다.
QueryContextAttributesW 전송 애플리케이션이 보안 컨텍스트의 특정 특성에 대해 CredSSP(자격 증명 보안 지원 공급자) 보안 패키지를 쿼리할 수 있습니다. (유니코드)
QueryCredentialsAttributesA 자격 증명과 연결된 이름과 같은 자격 증명의 특성을 검색합니다. (ANSI)
QueryCredentialsAttributesExA 보안 컨텍스트의 특성을 쿼리합니다.
QueryCredentialsAttributesExW 보안 컨텍스트의 특성을 쿼리합니다.
QueryCredentialsAttributesW 자격 증명과 연결된 이름과 같은 자격 증명의 특성을 검색합니다. (유니코드)
QuerySecurityContextToken 클라이언트 보안 컨텍스트에 대한 액세스 토큰을 가져오고 직접 사용합니다.
QuerySecurityPackageInfoA 지정된 보안 패키지에 대한 정보를 검색합니다. 이 정보에는 인증 정보, 자격 증명 및 컨텍스트의 크기에 대한 범위가 포함됩니다. (ANSI)
QuerySecurityPackageInfoW 지정된 보안 패키지에 대한 정보를 검색합니다. 이 정보에는 인증 정보, 자격 증명 및 컨텍스트의 크기에 대한 범위가 포함됩니다. (유니코드)

RevertSecurityContext 보안 패키지가 호출자의 가장을 중단하고 자체 보안 컨텍스트를 복원할 수 있도록 허용합니다.
SaslAcceptSecurityContext 보안 지원 공급자 인터페이스 AcceptSecurityContext(일반) 함수에 대한 표준 호출을 래핑하고 SASL 서버 쿠키 만들기를 포함합니다.
SaslEnumerateProfilesA SASL 인터페이스를 제공하는 패키지를 나열합니다. (ANSI)
SaslEnumerateProfilesW SASL 인터페이스를 제공하는 패키지를 나열합니다. (유니코드)
SaslGetContextOption 지정된 SASL 컨텍스트의 지정된 속성을 검색합니다.
SaslGetProfilePackageA 지정된 패키지에 대한 패키지 정보를 반환합니다. (ANSI)
SaslGetProfilePackageW 지정된 패키지에 대한 패키지 정보를 반환합니다. (유니코드)
SaslIdentifyPackageA 지정된 SASL 협상 버퍼와 일치하는 협상 접두사를 반환합니다. (ANSI)
SaslIdentifyPackageW 지정된 SASL 협상 버퍼와 일치하는 협상 접두사를 반환합니다. (유니코드)
SaslInitializeSecurityContextA 보안 지원 공급자 인터페이스 InitializeSecurityContext(일반) 함수에 대한 표준 호출을 래핑하고 서버에서 SASL 서버 쿠키를 처리합니다. (ANSI)
SaslInitializeSecurityContextW 보안 지원 공급자 인터페이스 InitializeSecurityContext(일반) 함수에 대한 표준 호출을 래핑하고 서버에서 SASL 서버 쿠키를 처리합니다. (유니코드)
SaslSetContextOption

지정된 SASL 컨텍스트에 대해 지정된 속성의 값을 설정합니다.
SetContextAttributesA 전송 애플리케이션이 보안 패키지에 대한 보안 컨텍스트의 특성을 설정할 수 있도록 합니다. 이 함수는 Schannel 보안 패키지에서만 지원됩니다. (ANSI)
SetContextAttributesW 전송 애플리케이션이 보안 패키지에 대한 보안 컨텍스트의 특성을 설정할 수 있도록 합니다. 이 함수는 Schannel 보안 패키지에서만 지원됩니다. (유니코드)
SetCredentialsAttributesA 자격 증명과 연결된 이름과 같은 자격 증명의 특성을 설정합니다. (ANSI)
SetCredentialsAttributesW 자격 증명과 연결된 이름과 같은 자격 증명의 특성을 설정합니다. (유니코드)
SspiAcceptSecurityContextAsync 전송 애플리케이션의 서버 구성 요소가 서버와 원격 클라이언트 간에 보안 컨텍스트를 비동기적으로 설정할 수 있습니다.
SspiAcquireCredentialsHandleAsyncA 보안 주체의 기존 자격 증명에 대한 핸들을 비동기적으로 획득합니다. (ANSI)
SspiAcquireCredentialsHandleAsyncW 보안 주체의 기존 자격 증명에 대한 핸들을 비동기적으로 획득합니다. (유니코드)
SspiAsyncContextRequiresNotify 지정된 비동기 컨텍스트에 호출 완료 시 알림이 필요한지 여부를 결정합니다.
SspiCompareAuthIdentities 지정된 두 자격 증명을 비교합니다.
SspiCopyAuthIdentity 지정된 불투명 자격 증명 구조의 복사본을 만듭니다.
SspiCreateAsyncContext 비동기 호출을 추적하는 데 사용되는 SspiAsyncContext의 instance 만듭니다.

SspiDecryptAuthIdentity 지정된 암호화된 자격 증명의 암호를 해독합니다.
SspiDecryptAuthIdentityEx SEC_WINNT_AUTH_IDENTITY_OPAQUE 구조체의 암호를 해독합니다.
SspiDeleteSecurityContextAsync SspiInitializeSecurityContextAsync 함수 또는 SspiAcceptSecurityContextAsync 함수에 대한 이전 호출에서 시작된 지정된 보안 컨텍스트와 연결된 로컬 데이터 구조를 삭제합니다.
SspiEncodeAuthIdentityAsStrings 지정된 인증 ID를 세 개의 문자열로 인코딩합니다.
SspiEncodeStringsAsAuthIdentity 3개의 자격 증명 문자열 집합을 인증 ID 구조로 인코딩합니다.
SspiEncryptAuthIdentity 지정된 ID 구조를 암호화합니다.
SspiEncryptAuthIdentityEx SEC_WINNT_AUTH_IDENTITY_OPAQUE 구조를 암호화합니다.
SspiExcludePackage 지정된 SSP(보안 지원 공급자)를 제외하도록 수정된 지정된 ID 구조의 복사본인 새 ID 구조를 만듭니다.
SspiFreeAsyncContext SspiCreateAsyncContext 함수 호출에서 만든 컨텍스트를 해제합니다.
SspiFreeAuthIdentity 지정된 ID 구조에 할당된 메모리를 해제합니다.
SspiFreeCredentialsHandleAsync 자격 증명 핸들을 해제합니다.
SspiGetAsyncCallStatus 제공된 컨텍스트와 연결된 비동기 호출의 현재 상태 가져옵니다.

SspiGetCredUIContext 자격 증명 공급자에서 컨텍스트 정보를 검색합니다. (SspiGetCredUIContext)
SspiGetTargetHostName 지정된 대상과 연결된 호스트 이름을 가져옵니다.
SspiInitializeSecurityContextAsyncA 비동기 보안 컨텍스트를 초기화합니다. (ANSI)
SspiInitializeSecurityContextAsyncW 비동기 보안 컨텍스트를 초기화합니다. (유니코드)
SspiIsAuthIdentityEncrypted 지정된 ID 구조가 암호화되었는지 여부를 나타냅니다.
SspiIsPromptingNeeded InitializeSecurityContext 또는 AcceptSecurityContext 함수를 호출한 후 반환된 오류가 SspiPromptForCredentials 함수에 대한 추가 호출이 필요한지 여부를 나타냅니다.
SspiLocalFree 지정된 버퍼와 연결된 메모리를 해제합니다.
SspiMarshalAuthIdentity 지정된 ID 구조를 바이트 배열로 직렬화합니다.
SspiPrepareForCredRead 지정된 ID 구조에서 대상 이름 및 자격 증명 형식을 생성합니다.
SspiPrepareForCredWrite CredWrite 함수 호출에서 매개 변수 값으로 전달할 수 있는 ID 구조에서 값을 생성합니다.
SspiPromptForCredentialsA SSPI(보안 지원 공급자 인터페이스) 애플리케이션에서 사용자에게 자격 증명을 입력하라는 메시지를 표시할 수 있습니다. (ANSI)
SspiPromptForCredentialsW

SSPI(보안 지원 공급자 인터페이스) 애플리케이션에서 사용자에게 자격 증명을 입력하라는 메시지를 표시할 수 있습니다. (유니코드)
SspiReinitAsyncContext 다시 사용할 비동기 컨텍스트를 표시합니다.
SspiSetAsyncNotifyCallback 비동기 호출 완료 시 알림을 받은 콜백을 등록합니다.
SspiUnmarshalAuthIdentity 지정된 바이트 값 배열을 ID 구조로 역직렬화합니다.
SspiUnmarshalCredUIContext ICredentialProvider::SetSerialization 메서드에 대한 이전 호출 중에 자격 증명 공급자가 가져온 자격 증명 정보를 역직렬화합니다.
SspiUpdateCredentials 지정된 컨텍스트와 연결된 자격 증명을 업데이트. (SspiUpdateCredentials)
SspiValidateAuthIdentity 지정된 ID 구조가 유효한지 여부를 나타냅니다.
SspiZeroAuthIdentity 지정된 ID 구조와 연결된 메모리 블록을 0으로 채웁니다.
VerifySignature MakeSignature 함수를 사용하여 서명된 메시지가 올바른 순서로 수신되었으며 수정되지 않은지 확인합니다.

콜백 함수

SspiAsyncNotifyCallback 비동기 SSPI 호출 완료를 알리는 데 사용되는 콜백입니다.

구조체

CREDUIWIN_MARSHALED_CONTEXT ICredentialProvider::SetSerialization 메서드를 사용하여 serialize된 자격 증명 정보를 지정합니다.
SEC_APPLICATION_PROTOCOL_LIST 애플리케이션 프로토콜 목록을 저장합니다.
SEC_APPLICATION_PROTOCOLS 애플리케이션 프로토콜 목록의 배열을 저장합니다.
SEC_CERTIFICATE_REQUEST_CONTEXT 인증서 요청 컨텍스트를 저장합니다.
SEC_CHANNEL_BINDINGS 보안 컨텍스트에 대한 채널 바인딩 정보를 지정합니다.
SEC_DTLS_MTU DTLS MTU를 저장합니다.
SEC_FLAGS 보안 플래그를 포함합니다.
SEC_NEGOTIATION_INFO 보안 협상 정보를 저장합니다.
SEC_PRESHAREDKEY 미리 공유된 키 정보를 포함합니다.
SEC_PRESHAREDKEY_IDENTITY 미리 공유된 키의 ID를 포함합니다.
SEC_SRTP_MASTER_KEY_IDENTIFIER SRTP master 키 식별자를 저장합니다.
SEC_SRTP_PROTECTION_PROFILES SRTP 보호 프로필을 저장합니다.
SEC_TOKEN_BINDING

토큰 바인딩 정보를 저장합니다.
SEC_TRAFFIC_SECRETS 연결에 대한 트래픽 비밀을 포함합니다.
SEC_WINNT_AUTH_BYTE_VECTOR 인증 구조에서 데이터의 바이트 오프셋 및 배열 길이를 지정합니다.
SEC_WINNT_AUTH_CERTIFICATE_DATA 직렬화된 인증서 정보를 지정합니다.
SEC_WINNT_AUTH_DATA 인증 데이터를 지정합니다.
SEC_WINNT_AUTH_DATA_PASSWORD 직렬화된 암호를 지정합니다.
SEC_WINNT_AUTH_DATA_TYPE_SMARTCARD_CONTEXTS_DATA 스마트 카드 컨텍스트에 대한 인증 데이터를 포함합니다.
SEC_WINNT_AUTH_FIDO_DATA FIDO 인증에 대한 데이터를 포함합니다.
SEC_WINNT_AUTH_IDENTITY_A 인증을 위해 특정 사용자 이름 및 암호를 런타임 라이브러리에 전달할 수 있습니다. (ANSI)
SEC_WINNT_AUTH_IDENTITY_EX2 인증 ID에 대한 정보를 포함합니다.
SEC_WINNT_AUTH_IDENTITY_EXA ANSI(SEC_WINNT_AUTH_IDENTITY_EXA) 구조에는 사용자에 대한 정보가 포함됩니다.
SEC_WINNT_AUTH_IDENTITY_EXW SEC_WINNT_AUTH_IDENTITY_EXW(유니코드) 구조에는 사용자에 대한 정보가 포함됩니다.
SEC_WINNT_AUTH_IDENTITY_INFO 인증에 대한 ID 정보를 포함합니다.

SEC_WINNT_AUTH_IDENTITY_W 인증을 위해 특정 사용자 이름 및 암호를 런타임 라이브러리에 전달할 수 있습니다. (유니코드)
SEC_WINNT_AUTH_NGC_DATA 인증을 위한 NGC 데이터를 포함합니다.
SEC_WINNT_AUTH_PACKED_CREDENTIALS 직렬화된 자격 증명을 지정합니다.
SEC_WINNT_AUTH_PACKED_CREDENTIALS_EX 직렬화된 자격 증명 및 자격 증명을 지원하는 보안 패키지 목록을 지정합니다.
SEC_WINNT_AUTH_SHORT_VECTOR USHORT 값 배열의 오프셋 및 문자 수를 지정합니다.
SEC_WINNT_CREDUI_CONTEXT 비직렬화된 자격 증명 정보를 지정합니다.
SEC_WINNT_CREDUI_CONTEXT_VECTOR SEC_WINNT_CREDUI_CONTEXT 구조에서 자격 증명 컨텍스트 데이터의 오프셋 및 크기를 지정합니다.
SecBuffer 보안 패키지에 전달하기 위해 전송 애플리케이션에서 할당한 버퍼에 대해 설명합니다.
SecBufferDesc SecBufferDesc 구조체는 전송 애플리케이션에서 보안 패키지로 전달할 SecBuffer 구조체의 배열을 설명합니다.
SecHandle 보안 핸들을 나타냅니다.
SecPkgContext_AccessToken 현재 보안 컨텍스트에 대한 액세스 토큰에 대한 핸들을 반환합니다.
SecPkgContext_ApplicationProtocol 보안 컨텍스트의 애플리케이션 프로토콜에 대한 정보를 포함합니다.

SecPkgContext_AuthorityA SecPkgContext_Authority 구조체에는 인증 기관(사용 가능한 경우)의 이름이 포함됩니다. (ANSI)
SecPkgContext_AuthorityW SecPkgContext_Authority 구조체에는 인증 기관(사용 가능한 경우)의 이름이 포함됩니다. (유니코드)
SecPkgContext_AuthzID 보안 컨텍스트의 AuthzID에 대한 정보를 포함합니다.
SecPkgContext_Bindings 보안 컨텍스트에 대한 채널 바인딩 정보를 포함하는 구조를 지정합니다.
SecPkgContext_ClientSpecifiedTarget QueryContextAttributes(다이제스트) 함수를 호출할 때 초기 대상의 SPN(서비스 사용자 이름)을 지정합니다.
SecPkgContext_CredentialNameA 자격 증명 이름 및 형식을 포함합니다.
SecPkgContext_CredentialNameW 보안 컨텍스트의 자격 증명 이름에 대한 정보입니다.
SecPkgContext_CredInfo 클라이언트 컨텍스트를 만드는 데 사용되는 자격 증명 유형을 지정합니다.
SecPkgContext_DceInfo SecPkgContext_DceInfo 구조에는 DCE 서비스에서 사용하는 권한 부여 데이터가 포함됩니다. QueryContextAttributes(일반) 함수는 이 구조를 사용합니다.
SecPkgContext_Flags SecPkgContext_Flags 구조에는 현재 보안 컨텍스트의 플래그에 대한 정보가 포함됩니다. 이 구조체는 QueryContextAttributes(일반)에서 반환됩니다.
SecPkgContext_KeyInfoA SecPkgContext_KeyInfo 구조에는 보안 컨텍스트에서 사용되는 세션 키에 대한 정보가 포함됩니다. (ANSI)

SecPkgContext_KeyInfoW SecPkgContext_KeyInfo 구조에는 보안 컨텍스트에서 사용되는 세션 키에 대한 정보가 포함됩니다. (유니코드)
SecPkgContext_LastClientTokenStatus InitializeSecurityContext 함수에 대한 최신 호출의 토큰이 클라이언트의 마지막 토큰인지 여부를 지정합니다.
SecPkgContext_Lifespan SecPkgContext_Lifespan 구조는 보안 컨텍스트의 수명을 나타냅니다. QueryContextAttributes(일반) 함수는 이 구조를 사용합니다.
SecPkgContext_LogoffTime 보안 컨텍스트의 로그오프 시간입니다.
SecPkgContext_NamesA SecPkgContext_Names 구조는 보안 컨텍스트와 연결된 사용자의 이름을 나타냅니다. QueryContextAttributes(일반) 함수는 이 구조를 사용합니다. (ANSI)
SecPkgContext_NamesW SecPkgContext_Names 구조는 보안 컨텍스트와 연결된 사용자의 이름을 나타냅니다. QueryContextAttributes(일반) 함수는 이 구조를 사용합니다. (유니코드)
SecPkgContext_NativeNamesA 클라이언트 및 서버 보안 주체 이름을 포함합니다.
SecPkgContext_NativeNamesW 보안 컨텍스트에서 클라이언트 및 서버의 네이티브 이름입니다.
SecPkgContext_NegoKeys 협상된 보안 패키지 키를 보유합니다.
SecPkgContext_NegoPackageInfo 협상된 애플리케이션 패키지에 대한 정보를 보유합니다.
SecPkgContext_NegoStatus 클라이언트 컨텍스트를 만드는 마지막 시도의 오류 상태 지정합니다.

SecPkgContext_NegotiatedTlsExtensions SecPkgContext_NegotiatedTlsExtensions 구조에는 현재(D)TLS 연결에 대해 협상된 (D)TLS 확장에 대한 정보가 포함되어 있습니다.
SecPkgContext_NegotiationInfoA SecPkgContext_NegotiationInfo 구조에는 설정되거나 설정된 보안 패키지에 대한 정보가 포함되며, 협상 상태 보안 패키지를 설정할 수도 있습니다. (ANSI)
SecPkgContext_NegotiationInfoW SecPkgContext_NegotiationInfo 구조에는 설정되거나 설정된 보안 패키지에 대한 정보가 포함되며, 협상 상태 보안 패키지를 설정할 수도 있습니다. (유니코드)
SecPkgContext_PackageInfoA 애플리케이션 패키지 정보를 보유합니다.
SecPkgContext_PackageInfoW 패키지 정보를 보유합니다.
SecPkgContext_PasswordExpiry SecPkgContext_PasswordExpiry 구조에는 보안 컨텍스트에 사용되는 암호 또는 기타 자격 증명의 만료에 대한 정보가 포함되어 있습니다. 이 구조체는 QueryContextAttributes(일반)에서 반환됩니다.
SecPkgContext_ProtoInfoA SecPkgContext_ProtoInfo 구조에는 사용 중인 프로토콜에 대한 정보가 있습니다. (ANSI)
SecPkgContext_ProtoInfoW SecPkgContext_ProtoInfo 구조에는 사용 중인 프로토콜에 대한 정보가 있습니다. (유니코드)
SecPkgContext_SessionKey SecPkgContext_SessionKey 구조에는 보안 컨텍스트에 사용되는 세션 키에 대한 정보가 포함되어 있습니다. 이 구조체는 QueryContextAttributes(일반) 함수에 의해 반환됩니다.
SecPkgContext_Sizes SecPkgContext_Sizes 구조체는 메시지 지원 함수에 사용되는 중요한 구조체의 크기를 나타냅니다. QueryContextAttributes(일반) 함수는 이 구조를 사용합니다.
SecPkgContext_StreamSizes

메시지 지원 함수와 함께 사용할 스트림의 다양한 부분의 크기를 나타냅니다. QueryContextAttributes(일반) 함수는 이 구조를 사용합니다.
SecPkgContext_SubjectAttributes 보안 특성 정보를 반환합니다.
SecPkgContext_Target 대상 정보를 보유합니다.
SecPkgContext_TargetInformation 보안 컨텍스트에 사용되는 자격 증명에 대한 정보를 반환합니다.
SecPkgContext_UserFlags 사용자 플래그를 보유합니다.
SecPkgCredentials_Cert 인증서 자격 증명을 지정합니다. QueryCredentialsAttributes 함수는 이 구조를 사용합니다.
SecPkgCredentials_KdcProxySettingsW 자격 증명에 대한 Kerberos 프록시 설정을 지정합니다.
SecPkgCredentials_NamesA SecPkgCredentials_Names 구조체는 컨텍스트와 연결된 사용자의 이름을 보유합니다. QueryCredentialsAttributes 함수는 이 구조를 사용합니다. (ANSI)
SecPkgCredentials_NamesW SecPkgCredentials_Names 구조체는 컨텍스트와 연결된 사용자의 이름을 보유합니다. QueryCredentialsAttributes 함수는 이 구조를 사용합니다. (유니코드)
SecPkgCredentials_SSIProviderA SecPkgCredentials_SSIProvider 구조체는 컨텍스트와 연결된 SSI 공급자 정보를 보유합니다. QueryCredentialsAttributes 함수는 이 구조를 사용합니다. (ANSI)
SecPkgCredentials_SSIProviderW SecPkgCredentials_SSIProvider 구조체는 컨텍스트와 연결된 SSI 공급자 정보를 보유합니다. QueryCredentialsAttributes 함수는 이 구조를 사용합니다. (유니코드)
SecPkgInfoA

SecPkgInfo 구조체는 이름 및 기능과 같은 보안 패키지에 대한 일반적인 정보를 제공합니다. (ANSI)
SecPkgInfoW SecPkgInfo 구조체는 이름 및 기능과 같은 보안 패키지에 대한 일반적인 정보를 제공합니다. (유니코드)
SECURITY_INTEGER SECURITY_INTEGER 숫자 값을 보유하는 구조체입니다. 다른 형식을 정의하는 데 사용됩니다.
SECURITY_PACKAGE_OPTIONS 보안 패키지에 대한 정보를 지정합니다.
SECURITY_STRING 커널 작업의 문자열 인터페이스로 사용되며 UNICODE_STRING 구조체의 복제본입니다.
SecurityFunctionTableA SecurityFunctionTable 구조체는 SSPI에 정의된 함수에 대한 포인터를 포함하는 디스패치 테이블입니다. (ANSI)
SecurityFunctionTableW SecurityFunctionTable 구조체는 SSPI에 정의된 함수에 대한 포인터를 포함하는 디스패치 테이블입니다. (유니코드)

열거형

SEC_APPLICATION_PROTOCOL_NEGOTIATION_EXT 협상된 애플리케이션 프로토콜 확장을 나타냅니다.
SEC_APPLICATION_PROTOCOL_NEGOTIATION_STATUS SEC 애플리케이션 프로토콜 협상의 상태 설명합니다.
SEC_TRAFFIC_SECRET_TYPE 사용된 트래픽 비밀 유형을 나타냅니다.
SecDelegationType

사용되는 위임 유형입니다.
SECPKG_ATTR_LCT_STATUS InitializeSecurityContext 함수에 대한 최신 호출의 토큰이 클라이언트의 마지막 토큰인지 여부를 나타냅니다.
SECPKG_CRED_CLASS 클라이언트 컨텍스트에서 사용되는 자격 증명의 유형을 나타냅니다. SECPKG_CRED_CLASS 열거형은 SecPkgContext_CredInfo 구조체에서 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

stm.h 헤더

아티클2023. 08. 26.

이 헤더는 라우팅 및 원격 액세스 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 라우팅 및 원격 액세스 서비스

stm.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

콜백 함수

PBLOCK_CONVERT_SERVICES_TO_STATIC BlockConvertServicesToStatic 함수는 지정된 인터페이스에서 받은 모든 서비스를 정적으로 변환합니다.
PBLOCK_DELETE_STATIC_SERVICES BlockDeleteStaticServices 함수는 지정된 인터페이스와 연결된 모든 정적 서비스를 삭제합니다.
PCLOSE_SERVICE_ENUMERATION_HANDLE CloseServiceEnumerationHandle 함수는 열거형을 종료하고 연결된 리소스를 해제합니다.
PCREATE_SERVICE_ENUMERATION_HANDLE CreateServiceEnumerationHandle 함수는 빠르고 변경 허용 열거형 함수를 사용할 수 있는 핸들을 반환합니다.
PCREATE_STATIC_SERVICE CreateStaticService 함수는 테이블에 정적 서비스를 추가합니다.
PDELETE_STATIC_SERVICE DeleteStaticService 함수는 테이블에서 정적 서비스를 삭제합니다.
PENUMERATE_GET_NEXT_SERVICE EnumerateGetNextService 함수는 CreateServiceEnumerationHandle로 시작된 열거형에서 다음 서비스 항목을 반환합니다.
PGET_FIRST_ORDERED_SERVICE

GetFirstOrderedService 함수는 테이블의 지정된 서비스 하위 집합에서 지정된 순서로 첫 번째 서비스를 반환합니다.
PGET_NEXT_ORDERED_SERVICE GetNextOrderedService 함수는 테이블의 서비스 하위 집합에서 다음 서비스를 반환합니다. 반환된 서비스는 지정된 순서 지정 메서드를 사용하여 지정된 입력 서비스 이후의 다음 서비스입니다.
PGET_SERVICE_COUNT GetServiceCount 함수는 테이블의 서비스 수를 반환합니다.
PIS_SERVICE IsService 함수는 지정된 형식 및 이름의 서비스가 서비스 테이블에 있는지 확인하고 필요에 따라 서비스의 매개 변수를 반환합니다.

구조체

IPX_SERVER_ENTRY IPX_SERVER_ENTRY 구조는 특정 IPX 서비스를 설명합니다.
IPX_SERVICE IPX_SERVICE 구조에는 IPX 서비스에 대한 정보가 포함되어 있으며 이 정보를 얻은 인터페이스 및 프로토콜을 식별합니다.
MPR40_SERVICE_CHARACTERISTICS MPR40_SERVICE_CHARACTERISTICS(stm.h) 구조에는 라우터 관리자에 라우팅 프로토콜을 등록하는 데 사용되는 정보가 포함되어 있습니다.
MPR50_SERVICE_CHARACTERISTICS MPR50_SERVICE_CHARACTERISTICS(stm.h) 구조에는 라우터 관리자에 라우팅 프로토콜을 등록하는 데 사용되는 정보가 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

storageprovider.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

storageprovider.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IStorageProviderHandler 특정 파일 또는 폴더와 연결된 IStorageProviderPropertyHandler를 검색합니다.
IStorageProviderPropertyHandler 파일 또는 폴더와 연결된 속성의 컬렉션을 제공합니다.

storprop.h 헤더

아티클2023. 08. 26.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

storprop.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CdromDisableDigitalPlayback 지정된 CD-ROM 또는 DVD 드라이브에 대한 디지털 재생을 사용하지 않도록 설정합니다.
CdromEnableDigitalPlayback 지정된 CD-ROM 또는 DVD 드라이브에 디지털 재생을 사용하도록 설정합니다.
CdromIsDigitalPlaybackEnabled 지정된 CD-ROM 또는 DVD 드라이브에 대해 디지털 재생을 사용할 수 있는지 여부를 결정합니다.
CdromKnownGoodDigitalPlayback 지정된 CD-ROM 또는 DVD 드라이브에 양호한 것으로 알려진 디지털 재생이 있는지 여부를 확인합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

stralign.h 헤더

아티클2023. 03. 08.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

stralign.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

uaw_wcslen null로 끝나는 유니코드 문자열의 문자 수를 검색합니다.

stringapiset.h 헤더

아티클2023. 08. 26.

이 헤더는 Windows 애플리케이션에 대한 Internationalization에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 애플리케이션에 대한 국제화

stringapiset.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CompareStringEx 이름으로 지정된 로캘에 대해 두 개의 유니코드(와이드 문자) 문자열을 비교합니다. 주의 CompareStringEx를 잘못 사용하면 애플리케이션의 보안이 손상됩니다.
CompareStringOrdinal 두 유니코드 문자열을 비교하여 이진 동등성을 테스트합니다.
CompareStringW CompareStringW(유니코드) 함수(stringapiset.h)는 식별자로 지정된 로캘에 대해 두 문자 문자열을 비교합니다.
FoldStringW 한 유니코드 문자열을 다른 문자열에 매핑하여 지정된 변환을 수행합니다. (FoldStringW)
GetStringTypeExW 지정된 소스 문자열의 문자에 대한 문자 형식 정보를 검색합니다.
GetStringTypeW 지정된 유니코드 소스 문자열의 문자에 대한 문자 형식 정보를 검색합니다.
MultiByteToWideChar 문자 문자열을 UTF-16(와이드 문자) 문자열에 매핑합니다.
WideCharToMultiByte UTF-16(와이드 문자) 문자열을 새 문자열에 매핑합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

strmif.h 헤더

아티클2023. 08. 26.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Directshow
- Microsoft TV 기술

strmif.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAMAnalogVideoDecoder IAMAnalogVideoDecoder 인터페이스는 비디오 캡처 필터에서 아날로그-디지털 변환 프로세스에 대한 정보를 설정하고 검색합니다. WDM 비디오 캡처 필터는 디바이스가 아날로그 비디오 캡처 디바이스인 경우 이 인터페이스를 노출합니다.
IAMAnalogVideoEncoder 참고 이 인터페이스는 더 이상 사용되지 않습니다. 참고 Microsoft는 이 인터페이스의 구현을 제공하지 않습니다.
IAMAsyncReaderTimestampScaling 풀 모드 소스 필터를 사용하여 더 큰 파일 크기를 지원할 수 있습니다.
IAMAudioInputMixer IAMAudioInputMixer 인터페이스는 이동 및 음량과 같은 오디오 캡처 속성을 제어합니다. 및 는 줄 또는 마이크와 같은 특정 오디오 입력을 사용하거나 사용하지 않도록 설정합니다. 오디오 캡처 필터는 필터 자체뿐만 아니라 각 입력 핀에 이 인터페이스를 노출합니다. 오디오 캡처 필터의 입력 핀은 실제 하드웨어 연결을 나타냅니다. 다른 DirectShow 필터에 연결되어 있지 않습니다. 핀 이름은 입력 형식을 나타냅니다. 예를 들어 "Line In" 또는 "Microphone"가 있습니다. 다음과 같이 IAMAudioInputMixer 인터페이스를 사용합니다. 특정 입력에 대한 설정을 제어하려면 핀의 인터페이스를 사용합니다. 여러 입력을 사용할 때 전체 속성을 설정하려면 필터에서 인터페이스를 사용합니다. 입력을 사용하거나 사용하지 않도록 설정하려면 해당 핀의 IAMAudioInputMixer::p ut_Enable 메서드를 호출합니다. 이 인터페이스의 일부 메서드는 기본 하드웨어의 기능에 따라 실패할 수 있습니다. 필터 Developers:_Implement 이 인터페이스는 오디오 캡처 필터의 각 입력 핀에 있습니다. 오디오 캡처 필터 자체에서 이 인터페이스를 구현하여 혼합 후 전체 오디오 설정을 제어할 수도 있습니다.
IAMAudioRendererStats

IAMAudioRendererStats 인터페이스는 오디오 렌더러 필터에서 통계 성능 정보를 검색합니다. 이 인터페이스는 개발 중에 오디오 렌더러의 성능 데이터를 기록하는 데 사용됩니다.
IAMBufferNegotiation IAMBufferNegotiation 인터페이스는 각 버퍼의 만들기 및 크기를 위해 필터에 대한 버퍼 수를 요청합니다.
IAMCameraControl IAMCameraControl 인터페이스는 확대/축소, 이동, 조리개 조정 또는 셔터 속도와 같은 카메라 설정을 제어합니다. 이 인터페이스를 가져오려면 카메라를 제어하는 필터를 쿼리합니다.
IAMCertifiedOutputProtection IAMCertifiedOutputProtection 인터페이스는 COPP(Certified Output Protection Protocol) 메시지를 그래픽 드라이버로 보냅니다.
IAMClockAdjust IAMClockAdjust 인터페이스는 참조 클록을 조정합니다. 시스템 참조 클록은 이 인터페이스를 노출합니다.
IAMClockSlave IAMClockSlave 인터페이스는 다른 클록과 속도를 일치시킬 때 오디오 렌더러의 허용 오차를 제어합니다. 오디오 렌더러가 다른 클록과 속도를 일치시키는 경우 오디오가 지정된 허용 오차의 양까지 드리프트할 수 있습니다.
IAMCopyCaptureFileProgress IAMCopyCaptureFileProgress 인터페이스는 ICaptureGraphBuilder2::CopyCaptureFile 메서드에서 사용하는 콜백 인터페이스입니다. CopyCaptureFile 메서드를 완료하는 데 시간이 오래 걸릴 수 있으므로 애플리케이션은 이 인터페이스를 구현하여 복사 작업의 진행률에 대한 정기적인 알림을 받을 수 있습니다. 애플리케이션이 이 정보를 받을 필요가 없는 경우 인터페이스를 구현할 필요가 없습니다.
IAMCrossbar IAMCrossbar 인터페이스는 신호를 아날로그 또는 디지털 소스에서 비디오 캡처 필터로 라우팅합니다. 이 인터페이스는 아날로그 비디오 크로스바 필터에 의해 구현됩니다.
IAMDecoderCaps IAMDecoderCaps 인터페이스는 MPEG 디코더 필터에서 기능 정보를 반환합니다.
IAMDeviceRemoval

IAMDeviceRemoval 인터페이스는 Filter Graph Manager가 캡처 디바이스에 대한 디바이스 제거 이벤트에 등록하는 방법을 제공합니다.
IAMDevMemoryAllocator 참고 이 인터페이스는 더 이상 AVI 분할기에서 지원되지 않습니다. 참고 이 인터페이스는 AVI 파일을 직접 하드웨어 메모리로 읽어야 하는 이전 하드웨어 디코더를 지원하도록 정의되었습니다.
IAMDevMemoryControl 참고 이 인터페이스는 더 이상 AVI 분할기에서 지원되지 않습니다. 참고 AVI 파일을 하드웨어 메모리로 직접 읽어야 하는 특정 이전 하드웨어 디코더를 지원하도록 정의되었습니다.
IAMDroppedFrames IAMDroppedFrames 인터페이스는 삭제된 프레임 수와 배달된 프레임 수를 포함하여 비디오 캡처 필터에서 성능 정보를 검색합니다. 애플리케이션은 이 인터페이스를 사용하여 런타임에 캡처 성능을 확인할 수 있습니다.
IAMExtDevice IAMExtDevice 인터페이스는 DV 카메라 또는 VTR(비디오 테이프 리코더)과 같은 외부 디바이스를 제어합니다.
IAMExtTransport IAMExtTransport 인터페이스는 VTR(비디오 테이프 사각형) 또는 캠코더의 전송을 제어합니다.
IAMFilterGraphCallback IAMFilterGraphCallback 인터페이스는 그래프를 빌드하는 동안 콜백 메커니즘을 제공합니다. 이 인터페이스를 사용하려면 애플리케이션 또는 클라이언트 개체에서 인터페이스를 구현합니다.
IAMFilterMiscFlags IAMFilterMiscFlags 인터페이스는 필터가 원본 필터인지 렌더러인지 쿼리합니다.
IAMGraphBuilderCallback IAMGraphBuilderCallback 인터페이스는 그래프를 빌드하는 동안 콜백 메커니즘을 제공합니다. 이 인터페이스를 사용하려면 애플리케이션 또는 클라이언트 개체에서 인터페이스를 구현합니다.
IAMGraphStreams IAMGraphStreams 인터페이스는 라이브 원본을 렌더링하는 필터 그래프를 제어합니다.
IAMLatency IAMLatency 인터페이스는 필터가 그래프에 도입하는 대기 시간의 양을 보고합니다.

IAMOpenProgress IAMOpenProgress 인터페이스는 파일 열기 작업의 진행률을 보고하고 애플리케이션이 작업을 취소할 수 있도록 합니다. 네트워크를 통해 파일을 여는 필터는 이 인터페이스를 노출할 수 있습니다.
IAMOverlayFX IAMOverlayFX 인터페이스는 비디오 오버레이가 사용자의 화면에 표시되는 방식을 제어합니다. 오버레이 믹서 필터는 이 인터페이스를 구현합니다.
IAMovieSetup 참고 이 인터페이스는 더 이상 사용되지 않습니다. (IAMovieSetup)
IAMPhysicalPinInfo 참고 이 인터페이스는 더 이상 사용되지 않습니다. (IAMPhysicalPinInfo)
IAMPluginControl 기본 설정 및 차단된 필터 목록을 제어합니다.
IAMPushSource IAMPushSource 인터페이스는 라이브 원본을 렌더링하는 필터 그래프를 동기화합니다.
IAMResourceControl IAMResourceControl 인터페이스가 열리고 디바이스가 실제로 필요하기 전에 오디오 디바이스 리소스를 보유하므로 재생을 보장하거나 애플리케이션에서 디바이스를 사용할 수 없다는 것을 미리 학습할 수 있습니다. 다음 필터는 이 인터페이스를 구현합니다.오디오 캡처 필터. DirectSound 렌더러 필터. 오디오 렌더러(WaveOut) 필터입니다.
IAMStreamConfig IAMStreamConfig 인터페이스는 오디오 및 비디오 모두에 대해 특정 캡처 및 압축 필터의 출력 형식을 설정합니다.
IAMStreamControl IAMStreamControl 인터페이스는 필터의 개별 스트림을 제어합니다.
IAMStreamSelect IAMStreamSelect 인터페이스는 파서 필터에서 사용 가능한 스트림 중에서 선택합니다.
IAMTimecodeDisplay

IAMTimecodeDisplay 인터페이스는 외부 SMPTE/MIDI 시간 코드 표시 디바이스를 제어합니다. DirectShow는 현재 이 인터페이스를 구현하는 필터를 제공하지 않습니다.
IAMTimecodeGenerator IAMTimecodeGenerator 인터페이스는 외부 SMPTE/MIDI 시간 코드 생성기가 필터 그래프에 데이터를 제공하는 방법을 제어합니다. DirectShow는 현재 이 인터페이스를 구현하는 필터를 제공하지 않습니다.
IAMTimecodeReader IAMTimecodeReader 인터페이스는 외부 디바이스에서 SMPTE 또는 MIDI 시간 코드를 읽습니다. MSDV 및 MSTape 드라이버는 외부 DV 또는 MPEG-2 캠코더에서 시간 코드를 읽기 위해 이 인터페이스를 지원합니다.
IAMTuner IAMTuner 인터페이스는 TV 튜너를 제어합니다.
IAMTunerNotification 참고 TV 튜너 필터가 콜백 메커니즘을 구현하지 않으므로 이 콜백 인터페이스는 더 이상 사용되지 않습니다. .
IAMTVAudio IAMTVAudio 인터페이스는 텔레비전 소스의 오디오를 제어합니다. TV 오디오 필터는 이 인터페이스를 구현합니다. 애플리케이션은 이를 사용하여 SAP(보조 오디오 프로그램) 및 스테레오 또는 모노 선택을 비롯한 TV 오디오 설정을 제어할 수 있습니다.
IAMTVAudioNotification 참고 TV 오디오 필터가 콜백 메커니즘을 구현하지 않으므로 이 콜백 인터페이스는 더 이상 사용되지 않습니다. .
IAMTVTuner IAMTVTuner 인터페이스는 TV 튜너를 제어합니다.
IAMVfwCaptureDialogs IAMVfwCaptureDialogs 인터페이스는 VFW(Windows용 비디오) 캡처 드라이버에서 제공하는 대화 상자를 표시합니다. VFW 캡처 필터는 이 인터페이스를 구현합니다.
IAMVfwCompressDialogs IAMVfwCompressDialogs 인터페이스는 VFW(Windows용 비디오) 코덱에서 제공하는 대화 상자를 표시합니다.

IAMVideoCompression IAMVideoCompression 인터페이스는 비디오 압축 속성을 설정하고 검색합니다.
IAMVideoControl IAMVideoControl 인터페이스는 사용 가능한 프레임 속도 및 이미지 방향을 열거하는 것과 같은 특정 비디오 캡처 작업을 제어합니다.
IAMVideoDecimationProperties IAMVideoDecimationProperties 인터페이스는 오버레이 믹서가 비디오 decimation을 수행하는 방법을 제어합니다. 비디오 창이 표시되는 비디오의 네이티브 크기보다 작으면 비디오 렌더러는 들어오는 비디오 즉, 비디오를 더 작은 크기로 축소해야 합니다. 10진수는 다음 위치 중 하나에서 수행할 수 있습니다. VGA 칩의 오버레이 하드웨어입니다. 비디오 포트에 기본 제공되는 스케일러입니다(연결이 비디오 포트를 통해 인 경우). 렌더러에 비디오를 제공하는 디코더입니다. 애플리케이션은 성능을 최적화하기 위해 이 인터페이스에서 메서드를 호출하여 특정 10진수 전략을 선택할 수 있습니다. 그러나 대부분의 애플리케이션은 이 인터페이스를 사용할 기회가 없습니다. 애플리케이션이 특정 하드웨어를 지원하도록 설계되지 않은 한 오버레이 믹서 필터의 기본 동작을 삭제하도록 수정할 이유가 없습니다.
IAMVideoProcAmp IAMVideoProcAmp 인터페이스는 밝기, 대비, 색조, 채도, 감마 및 선명도와 같은 들어오는 비디오 신호의 품질을 조정합니다. WDM 비디오 캡처 필터는 하드웨어가 이미지 조정을 지원하는 경우 이 인터페이스를 노출합니다.
IAsyncReader IAsyncReader 인터페이스는 필터에서 비동기 데이터 요청을 수행합니다. 이 인터페이스는 비동기 읽기 작업을 수행하는 출력 핀에 의해 노출됩니다.
IBaseFilter IBaseFilter 인터페이스는 DirectShow 필터의 기본 인터페이스입니다.
IBPCSatelliteTuner 참고 이 인터페이스는 구현되지 않았으며 더 이상 사용되지 않습니다. IBPCSatelliteTuner 인터페이스는 위성 TV 튜닝을 지원합니다.
ICaptureGraphBuilder 참고 이 인터페이스는 더 이상 사용되지 않습니다. (ICaptureGraphBuilder)
ICaptureGraphBuilder2 ICaptureGraphBuilder2 인터페이스 빌드는 그래프 및 기타 사용자 지정 필터 그래프를 캡처합니다.

ICodecAPI ICodecAPI 인터페이스(strmif.h)는 인코더 또는 디코더 필터에서 설정을 설정하고 검색합니다.
IConfigAviMux IConfigAviMux 인터페이스는 AVI Mux 필터를 구성합니다.
IConfigInterleaving IConfigInterleaving 인터페이스는 AVI Mux 필터가 오디오 및 비디오 샘플을 인터리브하는 방법을 제어합니다.
ICreateDevEnum ICreateDevEnum 인터페이스는 비디오 캡처 디바이스 또는 오디오 캡처 디바이스와 같은 필터 범주에 대한 열거자를 만듭니다.
IDDrawExclModeVideo IDDrawExclModeVideo 인터페이스를 사용하면 DirectDraw 전용 전체 화면 모드에서 비디오 재생이 가능합니다.
IDDrawExclModeVideoCallback IDDrawExclModeVideoCallback 인터페이스는 IDDrawExclModeVideo 인터페이스에 대한 콜백 인터페이스입니다. 이 콜백 인터페이스를 사용하면 애플리케이션이 비디오 표시 유형, 크기 및 위치를 조정할 수 있도록 오버레이 위치, 크기, 표시 유형 등에 대한 변경 내용에 대한 동기 알림을 받을 수 있습니다. 이렇게 하면 시작, 끝 또는 재생 중에 색상 키 플래시가 발생하지 않습니다. 애플리케이션은 인터페이스를 구현해야 합니다. 재생에 문제가 발생하므로 어떤 메서드도 비디오 처리를 차단하거나 느리게 하는 것이 중요합니다. IDDrawExclModeVideo를 지원하는 필터를 작성하거나 애플리케이션이 적시에 색 키를 그릴 수 있도록 콜백을 생성해야 하는 경우 이 인터페이스를 사용합니다.
IDecimateVideoImage IDecimateVideoImage 인터페이스는 디코더 필터의 10진수를 지정합니다.
IDistributorNotify IDistributorNotify 인터페이스를 사용하면 필터 그래프가 변경될 때 플러그 인 배포자에게 알림을 받을 수 있습니다. 애플리케이션은 이 인터페이스를 사용하지 않습니다.
IDrawVideoImage 참고 이 인터페이스는 더 이상 사용되지 않습니다. (IDrawVideoImage)
IDvdCmd

IDvdCmd 인터페이스는 DVD 명령이 시작되거나 끝날 때까지 기다립니다. DVD 탐색기는 이 인터페이스를 노출하는 동기화 개체를 만듭니다.
IDvdControl 참고 이 인터페이스는 더 이상 사용되지 않습니다. (IDvdControl)
IDvdControl2 IDvdControl2 인터페이스는 DVD-Video 타이틀을 탐색하고 재생합니다.
IDvdGraphBuilder IDvdGraphBuilder 인터페이스는 DVD-Video 재생을 위한 필터 그래프를 빌드합니다.
IDvdInfo 참고 이 인터페이스는 더 이상 사용되지 않습니다. (IDvdInfo)
IDvdInfo2 IDvdInfo2 인터페이스는 DVD 디스크의 특성 또는 DVD 재생 및 탐색의 현재 상태를 보고합니다.
IDvdState IDvdState 인터페이스는 현재 상태를 캐시합니다. 이 인터페이스를 구현하는 개체를 DVD 책갈피라고 합니다. 재생 위치, 사용자의 자녀 수준 및 DVD 지역을 포함하는 DVD 상태를 저장하고 복원하는 데 사용할 수 있습니다.
IDVEnc IDVEnc 인터페이스는 DV 비디오 인코더 필터의 속성을 설정하고 검색합니다.
IDVRGB219 IDVRGB219 인터페이스는 DV 비디오 인코더 및 DV 비디오 디코더 필터의 동적 범위를 제어합니다.
IDVSplitter DV(디지털 비디오) 스트림의 프레임 속도를 다운그레이드합니다.
IEncoderAPI IEncoderAPI는 더 이상 사용할 수 없습니다. (IEncoderAPI)
IEnumFilters IEnumFilters 인터페이스는 필터 그래프의 필터를 열거합니다.

IEnumMediaTypes IEnumMediaTypes 인터페이스는 핀의 기본 미디어 형식을 열거합니다.
IEnumPins 필터의 핀을 열거합니다. IBaseFilter::EnumPins 메서드는 이 인터페이스를 반환합니다.
IEnumRegFilters 참고 이 인터페이스는 더 이상 사용되지 않습니다. (IEnumRegFilters)
IEnumStreamIdMap IEnumStreamIdMap 인터페이스는 MPEG-2 Demultiplexer의 IMPEG2StreamIdMap::MapStreamId 메서드에 의해 생성된 스트림 ID 맵의 표준 COM 컬렉션에서 구현됩니다.
IFileSinkFilter IFileSinkFilter 인터페이스는 파일에 미디어 스트림을 쓰는 필터에 구현됩니다.
IFileSinkFilter2 IFileSinkFilter2 인터페이스는 IFileSinkFilter 인터페이스를 확장합니다.
IFileSourceFilter IFileSourceFilter 인터페이스는 원본 필터에 의해 노출되어 렌더링할 미디어 파일의 파일 이름과 미디어 형식을 설정합니다.
IFilterChain IFilterChain 인터페이스는 필터 그래프에서 필터 체인을 시작, 중지 또는 제거하는 메서드를 제공합니다.
IFilterGraph IFilterGraph 인터페이스는 필터 그래프를 빌드하는 메서드를 제공합니다.
IFilterGraph2 IFilterGraph2 인터페이스는 필터 그래프를 빌드하는 메서드를 포함하는 IFilterGraph 및 IGraphBuilder 인터페이스를 확장합니다. Filter Graph Manager는 이 인터페이스를 구현합니다.
IFilterGraph3 IFilterGraph3 인터페이스는 필터 그래프를 빌드하는 메서드를 포함하는 IFilterGraph2 인터페이스를 확장합니다. Filter Graph Manager는 이 인터페이스를 구현합니다.

IFilterMapper 참고 이 인터페이스는 더 이상 사용되지 않습니다. (IFilterMapper)
IFilterMapper2 필터를 등록 및 등록 취소하고 레지스트리에서 필터를 찾습니다.
IFilterMapper3 IFilterMapper3 인터페이스는 IFilterMapper2 인터페이스를 확장합니다. Filter Mapper 개체는 이 인터페이스를 구현합니다. 참고 이 인터페이스는 더 이상 사용되지 않습니다. .
IGetCapabilitiesKey IGetCapabilitiesKey 인터페이스를 사용하면 애플리케이션이 인코더 필터의 instance 만들지 않고도 레지스트리에서 소프트웨어 또는 하드웨어 코덱의 기능을 검색할 수 있습니다.
IGraphBuilder 이 인터페이스는 애플리케이션이 필터 그래프를 빌드할 수 있도록 하는 메서드를 제공합니다.
IGraphConfig Filter Graph Manager는 동적 그래프 빌드를 지원하기 위해 IGraphConfig를 노출합니다.
IGraphConfigCallback IGraphConfigCallback 인터페이스에는 IGraphConfig::Reconfigure에 전달된 콜백 메서드가 포함되어 있습니다. 호출자(애플리케이션 또는 필터)는 이 인터페이스를 구현합니다. 자세한 내용은 IGraphConfig를 참조하세요.
IGraphVersion IGraphVersion 인터페이스는 필터 그래프 관리자에서 구현되어 플러그 인 배포자 및 애플리케이션이 그래프가 변경된 시기를 알 수 있는 방법을 제공합니다.
IIPDVDec IIPDVDec 인터페이스는 DV 비디오 디코더 필터에서 속성을 설정하고 검색하는 메서드를 제공합니다.
IMediaEventSink Filter Graph Manager에 필터 그래프 내에서 발생하는 이벤트를 알 수 있습니다.
IMediaFilter

IMediaFilter 인터페이스는 필터의 스트리밍 상태를 제어합니다. 모든 DirectShow 필터는 이 인터페이스를 구현합니다.
IMediaPropertyBag IMediaPropertyBag 인터페이스는 Media Property Bag 개체에 의해 노출됩니다.
IMediaSample IMediaSample 인터페이스는 미디어 샘플에서 속성을 설정하고 검색합니다.
IMediaSample2 IMediaSample2 인터페이스는 미디어 샘플에서 속성을 설정하고 검색합니다. 이 인터페이스는 IMediaSample 인터페이스를 상속합니다.
IMediaSample2Config IMediaSample2Config 인터페이스는 VRAM 캡처 버퍼를 나타내는 Direct3D 표면에 대한 포인터를 반환합니다.
IMediaSeeking IMediaSeeking 인터페이스에는 스트림 내의 위치를 찾고 재생 속도를 설정하는 메서드가 포함되어 있습니다.
IMemAllocator IMemAllocator 인터페이스는 핀 간에 데이터를 이동하기 위해 미디어 샘플을 할당합니다. 이 인터페이스는 입력 핀이 IMemInputPin 인터페이스를 노출할 때 할당자를 공유하는 핀에서 사용됩니다.
IMemAllocatorCallbackTemp IMemAllocatorCallbackTemp 인터페이스를 사용하면 샘플이 할당자의 무료 목록에 반환될 때마다 필터가 할당자로부터 콜백 알림을 받을 수 있습니다. 이 인터페이스의 사용은 더 이상 사용되지 않습니다.
IMemAllocatorNotifyCallbackTemp 샘플이 할당자의 무료 목록에 반환될 때마다 필터가 할당자로부터 콜백 알림을 받을 수 있도록 합니다.
IMemInputPin IMemInputPin 인터페이스는 미디어 데이터를 입력 핀에 전달합니다.
IMpeg2Demultiplexer

이 인터페이스는 MPEG-2 Demultiplexer 필터(Demux)에서 구현되며 프로그램 스트림 모드와 전송 스트림 모드 모두에서 사용됩니다.
IMPEG2StreamIdMap 이 인터페이스는 MPEG-2 Demultiplexer 필터(Demux)의 각 출력 핀에서 구현되며 프로그램 스트림 모드에서만 사용됩니다.
IOverlay IOverlay 인터페이스는 필터가 비디오를 올바른 창 위치에 배치하는 동안 비디오 메모리에 직접 쓸 수 있도록 정보를 제공합니다.
IOverlayNotify IOverlayNotify 인터페이스는 렌더링 창에 대한 변경 알림과 함께 디코더와 같은 업스트림 필터를 제공합니다.
IOverlayNotify2 IOverlayNotify2 인터페이스는 IOverlayNotify 인터페이스에서 파생됩니다.
IPersistMediaPropertyBag IPersistMediaPropertyBag 인터페이스는 Audio-Video AVI(Interleaved) 스트림에서 INFO 및 DISP 청크를 설정하고 검색합니다.
Ipin 이 인터페이스는 모든 입력 및 출력 핀에 의해 노출됩니다. 필터 그래프 관리자는 이 인터페이스를 사용하여 핀을 연결하고 플러시 작업을 수행합니다.
IPinConnection 이 인터페이스는 필터가 계속 실행되는 동안 입력 핀을 다시 연결하는 메서드를 제공합니다.
IPinFlowControl 활성 출력 핀에서 데이터 흐름을 차단합니다.
IQualityControl IQualityControl 인터페이스는 품질 제어를 지원합니다.
IReferenceClock IReferenceClock 인터페이스는 필터 그래프에 대한 참조 시간을 제공합니다. 참조 클록 역할을 할 수 있는 필터는 이 인터페이스를 노출할 수 있습니다.

IReferenceClockTimerControl IReferenceClockTimerControl 인터페이스는 참조 클록에서 사용하는 타이머 기간을 변경합니다. 이 인터페이스는 DirectShow 시스템 참조 시계에 의해 노출됩니다.
IRegisterServiceProvider IRegisterServiceProvider 인터페이스는 개체를 필터 그래프 관리자에 서비스로 등록합니다. 자세한 내용은 IRegisterServiceProvider::RegisterService를 참조하세요.
IResourceConsumer IResourceConsumer 인터페이스는 IResourceManager 인터페이스를 사용하는 개체에 대한 콜백 메커니즘을 제공합니다. 개체는 IResourceManager 인터페이스를 사용하여 필터 그래프 관리자에서 리소스를 요청하는 경우 IResourceConsumer를 구현해야 합니다.
IResourceManager IResourceManager 인터페이스는 시스템 리소스에 대한 경합을 해결합니다. 필터 그래프 관리자는 이 인터페이스를 노출합니다.
ISeekingPassThru ISeekingPassThru 인터페이스는 한 입력 필터 검색을 구현하는 도우미 개체를 만듭니다.
IStreamBuilder IStreamBuilder 인터페이스를 사용하면 출력 핀이 필터 그래프 관리자에게 핀 자체가 필터 그래프의 다운스트림 섹션을 빌드할 것임을 알릴 수 있습니다.
IVideoEncoder IVideoEncoder 인터페이스는 선택적으로 비디오 인코더 필터에 의해 노출됩니다.
IVideoFrameStep IVideoFrameStep 인터페이스는 비디오 스트림을 단계합니다.
IVMRAspectRatioControl IVMRAspectRatioControl 인터페이스는 비디오 혼합 렌더러 필터 7(VMR-7)이 원본 비디오의 가로 세로 비율을 유지하는지 여부를 제어합니다.
IVMRDeinterlaceControl IVMRDeinterlaceControl 인터페이스는 VMR-7(Video Mixing Renderer Filter 7)을 사용하여 고급 하드웨어 가속 디인터레이싱을 지원합니다.

IVMRFilterConfig IVMRFilterConfig 인터페이스는 VMR-7(Video Mixing Renderer Filter 7)의 운영 모드 및 비디오 렌더링 메커니즘을 구성하는 데 사용됩니다.
IVMRImageCompositor IVMRImageCompositor 인터페이스는 비디오 혼합 렌더러 필터 7(VMR-7)의 기본 작성자에 의해 구현됩니다.
IVMRImagePresenter IVMRImagePresenter 인터페이스는 VMR-7(Video Mixing Renderer Filter 7)의 기본 Allocator-Presenter 의해 구현됩니다.
IVMRImagePresenterConfig IVMRImagePresenterConfig 인터페이스는 VMR-7(Video Mixing Renderer Filter 7)에서 사용하는 할당자-발표자에 대한 렌더링 기본 설정을 위한 메서드를 제공합니다. 애플리케이션은 이 인터페이스를 직접 사용하면 안 됩니다.
IVMRImagePresenterExclModeConfig IVMRImagePresenterExclModeConfig 인터페이스는 IVMRImagePresenterConfig에서 상속되며 단독 모드 Allocator-Presenter에서 렌더링 기본 설정을 설정하고 검색하는 메서드를 제공합니다.
IVMRMixerBitmap IVMRMixerBitmap 인터페이스를 사용하면 VMR-7(비디오 혼합 렌더러 필터 7)을 사용할 때 애플리케이션이 비트맵 또는 DirectDraw 표면의 정적 이미지를 비디오 스트림에 혼합할 수 있습니다.
IVMRMixerControl IVMRMixerControl 인터페이스를 사용하면 애플리케이션이 비디오 혼합 렌더러 필터 7(VMR-7)에서 들어오는 비디오 스트림을 조작할 수 있습니다.
IVMRMonitorConfig IVMRMonitorConfig 인터페이스는 비디오 혼합 렌더러 필터 7(VMR-7)에 의해 구현됩니다.
IVMRSurface IVMRSurface 인터페이스는 VMR-7(Video Mixing Renderer Filter 7)에서 사용하는 미디어 샘플에서 구현됩니다.
IVMRSurfaceAllocator IVMRSurfaceAllocator 인터페이스는 비디오 혼합 렌더러 필터 7(VMR-7)의 기본 할당자-발표자에 의해 구현됩니다.

IVMRSurfaceAllocatorNotify IVMRSurfaceAllocatorNotify 인터페이스는 비디오 혼합 렌더러 필터 7(VMR-7)에 의해 구현됩니다.
IVMRVideoStreamControl IVMRVideoStreamControl 인터페이스는 VMR-7(Video Mixing Renderer Filter 7)의 각 입력 핀에 구현됩니다.
IVMRWindowlessControl IVMRWindowlessControl 인터페이스는 VMR-7(비디오 혼합 렌더러 필터 7)이 컨테이너 창 내에서 비디오 스트림을 렌더링하는 방법을 제어합니다.
IVPManager IVPManager 인터페이스는 VPM(비디오 포트 관리자)에서 구현됩니다.

구조체

ALLOCATOR_PROPERTIES ALLOCATOR_PROPERTIES 구조체는 할당자의 개수, 크기, 맞춤 및 접두사 속성을 설명합니다.
AM_DVD_RENDERSTATUS AM_DVD_RENDERSTATUS 구조에는 DVD-Video 재생의 상태 나타내는 코드가 포함되어 있습니다. 이러한 코드는 IDvdGraphBuilder::RenderDvdVideoVolume 메서드에서 사용됩니다.
AM_MEDIA_TYPE AM_MEDIA_TYPE 구조는 미디어 샘플의 형식을 설명합니다.
AM_SAMPLE2_PROPERTIES AM_SAMPLE2_PROPERTIES 구조는 미디어 샘플의 속성을 설명합니다. IMediaSample2 인터페이스는 이 구조를 사용합니다.
AM_STREAM_INFO AM_STREAM_INFO 구조에는 스트림 제어 정보가 포함됩니다.
AMCOPPCommand

AMCOPPCommand 구조체에는 COPP(Certified Output Protection Protocol) 명령이 포함되어 있습니다.
AMCOPPSignature AMCOPPSignature 구조체에는 IAMCertifiedOutputProtection::SessionSequenceStart 메서드에 필요한 서명이 포함되어 있습니다.
AMCOPPStatusInput AMCOPPStatusInput 구조에는 COPP(Certified Output Protection Protocol) 상태 요청이 포함되어 있습니다.
AMCOPPStatusOutput AMCOPPStatusOutput 구조에는 COPP(Certified Output Protection Protocol) 상태 요청의 결과가 포함됩니다.
AUDIO_STREAM_CONFIG_CAPS AUDIO_STREAM_CONFIG_CAPS 구조는 다양한 오디오 형식을 설명합니다. 오디오 압축 및 캡처 필터는 이 구조를 사용하여 생성할 수 있는 형식을 설명합니다.
CodecAPIEventData CodecAPIEventData 구조체(strmif.h)는 EC_CODECAPI_EVENT 이벤트에 대한 이벤트 데이터를 포함합니다. 이 이벤트는 ICodecAPI 인터페이스를 지원하는 코덱에서 전송됩니다.
COLORKEY COLORKEY 구조는 렌더러와 다른 필터 간에 색 키 정보를 전달합니다.
DDCOLORKEY 색 키를 값 범위로 설명합니다.
DVD_AudioAttributes DVD_AudioAttributes 구조체는 IDvdInfo2::GetAudioAttributes에서 디스크의 다양한 오디오 특성을 수신하는 데 사용됩니다.
DVD_DECODER_CAPS DVD_DECODER_CAPS 구조는 DVD 디코더의 기능을 나타냅니다.
DVD_HMSF_TIMECODE DVD_HMSF_TIMECODE 구조체는 DVD 시간 코드로 시간, 분, 초 및 프레임을 제공합니다.

DVD_KaraokeAttributes DVD_KaraokeAttributes 구조에는 노래방 오디오 스트림에 대한 정보가 포함됩니다. IDvdInfo2::GetKaraokeAttributes 메서드는 지정된 스트림에 대한 DVD_KaraokeAttributes 구조를 채웁니다.
DVD_MenuAttributes DVD_MenuAttributes 구조에는 DVD 메뉴에 대한 정보가 포함되어 있습니다. IDvdInfo2::GetTitleAttributes 메서드는 지정된 스트림에 대한 DVD_MenuAttributes 구조를 채웁니다.
DVD_MUA_Coeff DVD_MUA_Coeff 구조체는 다중 채널 오디오 스트림의 한 채널에 대한 혼합 계수를 정의합니다. DVD_MultichannelAudioAttributes 구조체에는 스트림의 각 채널에 대해 하나씩 8개의 DVD_MUA_Coeff 구조체 배열이 포함됩니다.
DVD_MUA_MixingInfo DVD_MUA_MixingInfo 구조는 지정된 타이틀의 한 오디오 스트림에 있는 채널에 대한 서라운드 사운드 혼합 정보를 설명합니다.
DVD_MultichannelAudioAttributes DVD_MultichannelAudioAttributes 구조는 지정된 타이틀 내에서 한 오디오 스트림의 다중 채널 특성을 설명합니다.
DVD_PLAYBACK_LOCATION DVD_PLAYBACK_LOCATION 구조는 DVD 재생 위치를 나타냅니다.
DVD_PLAYBACK_LOCATION2 DVD_PLAYBACK_LOCATION2 구조는 DVD 재생 위치를 나타냅니다.
DVD_SubpictureAttributes DVD_SubpictureAttributes 구조에는 DVD 하위 이미지에 대한 정보가 포함되어 있습니다. IDvdInfo2::GetSubpictureAttributes 메서드는 지정된 스트림에 대한 DVD_SubpictureAttributes 구조를 채웁니다.
DVD_TIMECODE DVD_TIMECODE 구조에는 DVD 시간 코드(시간, 분, 초 및 프레임)가 포함됩니다.
DVD_TitleAttributes DVD_TitleAttributes 구조에는 DVD 제목에 대한 정보가 포함되어 있습니다.

DVD_VideoAttributes DVD_VideoAttributes 구조는 현재 제목 또는 메뉴에 대한 비디오 스트림의 특성을 설명합니다.
DVINFO DVINFO 구조는 DV(디지털 비디오) 스트림의 형식을 설명합니다.
FILTER_INFO FILTER_INFO 구조에는 필터에 대한 정보가 포함됩니다.
NORMALIZEDRECT NORMALIZEDRECT 구조체는 혼합 작업에서 VMR 필터와 함께 사용하여 컴퍼지션 공간에서 비디오 사각형의 위치를 지정합니다.
PIN_INFO PIN_INFO 구조체에는 핀에 대한 정보가 포함됩니다.
품질 품질 구조는 렌더러의 홍수 또는 기근을 나타내고 렌더러의 성능을 최적화하기 위해 삭제하거나 추가할 프레임의 백분율을 지정하여 품질 메시지를 설명합니다.
REGFILTER2 REGFILTER2 구조에는 필터를 등록하기 위한 정보가 포함되어 있습니다.
REGFILTERPINS REGFILTERPINS 구조에는 필터를 등록하기 위한 핀 정보가 포함되어 있습니다.
REGFILTERPINS2 REGFILTERPINS2 구조에는 IFilterMapper2 인터페이스를 통해 필터를 등록하기 위한 정보가 포함되어 있습니다.
REGPINMEDIUM REGPINMEDIUM 구조체는 IFilterMapper2 인터페이스를 통해 등록을 위한 핀 매체를 설명합니다.
REGPINTYPES REGPINTYPES 구조에는 필터를 등록하기 위한 미디어 형식 정보가 포함되어 있습니다.
STREAM_ID_MAP

STREAM_ID_MAP 구조는 MPEG-2 프로그램 스트림 내의 기본 스트림을 설명합니다. IEnumStreamIdMap 인터페이스 메서드와 함께 사용됩니다.
TIMECODE TIMECODE 구조에는 기본 시간 코드 프레임 수 정보가 포함됩니다. T
TIMECODE TIMECODE 구조에는 기본 시간 코드 프레임 수 정보가 포함됩니다. (TIMECODE)
TIMECODE_SAMPLE TIMECODE_SAMPLE 구조에는 전체 시간 코드 정보가 포함됩니다.
VIDEO_STREAM_CONFIG_CAPS VIDEO_STREAM_CONFIG_CAPS 구조는 다양한 비디오 형식을 설명합니다. 비디오 압축 및 비디오 캡처 필터는 이 구조를 사용하여 생성할 수 있는 형식을 설명합니다.
VMRALLOCATIONINFO VMRALLOCATIONINFO 구조체는 VMR-7 필터의 IVMRSurfaceAllocator::AllocateSurface 메서드에 사용됩니다.
VMRALPHABITMAP VMRALPHABITMAP 구조는 애플리케이션이 복합 비디오 프레임에 표시할 정적 알파 혼합 비트맵을 제공할 때 VMR-7 필터의 IVMRMixerBitmap 메서드에 사용됩니다.
VMRDeinterlaceCaps VMRDeinterlaceCaps 구조는 디인터레이싱 모드의 기능을 설명합니다.
VMRFrequency VMRFrequency 구조는 비디오 스트림의 빈도를 설명합니다. 빈도는 비율로 설명됩니다. 예를 들어 29.97fps의 NTSC 프레임 속도는 30,000:1001로 표시됩니다.
VMRGUID VMRGUID 구조체는 VMRMONITORINFO 구조체의 멤버이며 시스템에서 모니터를 식별하는 데 사용됩니다(VMR-7에만 해당).
VMRMONITORINFO VMRMONITORINFO 구조체는 IVMRMonitorConfig::GetAvailableMonitors 메서드에서 시스템의 모니터에 대한 정보를 설정하고 검색하는 데 사용됩니다(VMR-7에만 해당).

VMRPRESENTATIONINFO VMRPRESENTATIONINFO 구조체는 IVMRImagePresenter::P resentImage 메서드(VMR-7에만 해당)에 사용됩니다.
VMRVideoDesc 이 항목은 Windows XP 서비스 팩 1 이상에 적용됩니다. VMRVideoDesc 구조체는 디인터레이스할 비디오 스트림을 설명합니다.
VMRVIDEOSTREAMINFO 이 항목은 Windows XP 이상에 적용됩니다. VMRVIDEOSTREAMINFO 구조체는 이미지 작성기에서 IVMRImageCompositor::CompositeImage에 대한 VMR-7 필터 호출에 사용됩니다.

열거형

_AM_AUDIO_RENDERER_STAT_PARAM _AM_AUDIO_RENDERER_STAT_PARAM 열거형은 오디오 렌더러에서 검색할 성능 정보를 지정합니다.
_AM_FILTER_MISC_FLAGS _AM_FILTER_MISC_FLAGS 열거형에는 필터가 원본 필터인지 렌더러 필터인지를 나타내는 플래그가 포함되어 있습니다.
_AM_INTF_SEARCH_FLAGS 필터 그래프에서 인터페이스를 찾으려고 할 때 검색할 개체의 형식을 지정합니다.
_AM_OVERLAY_NOTIFY_FLAGS AM_OVERLAY_NOTIFY_FLAGS 열거형은 오버레이가 변경되었거나 변경하려고 하는 내용을 나타냅니다.
_AM_PIN_FLOW_CONTROL_BLOCK_FLAGS 출력 핀에서 데이터 흐름을 차단하는 방법을 지정하는 플래그를 정의합니다.
_AM_PUSHSOURCE_FLAGS 라이브 소스 필터의 동작을 나타냅니다.

_AMRESCTL_RESERVEFLAGS 현재 예약 중인 리소스 수를 증가 또는 감소할지 여부를 지정합니다.
_DVENCODERFORMAT DV(디지털 비디오) 형식을 나타냅니다.
_DVENCODERRESOLUTION DV(디지털 비디오) 인코딩 해상도를 나타냅니다.
_DVENCODERVIDEOFORMAT DV(디지털 비디오) 인코딩에 대한 비디오 표준(NTSC 또는 PAL)을 나타냅니다.
_DVRESOLUTION DV(디지털 비디오) 디코딩 해상도를 나타냅니다.
_REM_FILTER_FLAGS 필터 그래프에서 필터를 제거하는 방법을 지정합니다.
AM_DVD_GRAPH_FLAGS AM_DVD_GRAPH_FLAGS 열거형은 DVD 탐색기에서 DVD 재생 그래프를 빌드하는 방법을 지정합니다. 이러한 플래그는 IDvdGraphBuilder::RenderDvdVideoVolume 메서드와 함께 사용됩니다.
AM_DVD_STREAM_FLAGS DVD 스트림 유형(비디오, 오디오 또는 하위픽처)을 설명합니다.
AM_GRAPH_CONFIG_RECONNECT_FLAGS 필터 그래프를 동적으로 다시 빌드할 때 필터를 다시 연결하는 방법을 지정합니다.
AM_SEEKING_SEEKING_CAPABILITIES 미디어 스트림의 검색 기능을 지정합니다.
AM_STREAM_INFO_FLAGS AM_STREAM_INFO_FLAGS 열거형은 핀의 스트림 제어 상태 나타내는 플래그를 정의합니다.
AMOVERLAYFX DirectDraw 하드웨어 오버레이 표면에 미치는 영향을 지정합니다.

AMTunerModeType TV 튜너(케이블 또는 안테나)의 빈도를 지정합니다. (AMTunerModeType)
AnalogVideoStandard AnalogVideoStandard 열거형은 아날로그 텔레비전 신호의 형식을 지정합니다.
CameraControlFlags CameraControlFlags 열거형은 카메라 설정이 수동으로 또는 자동으로 제어되는지 여부를 정의합니다.
CameraControlProperty CameraControlProperty 열거형은 카메라의 설정을 지정합니다.
CompressionCaps 비디오 압축 기능을 나타냅니다.
DECIMATION_USAGE 오버레이 믹서 필터 필터가 비디오 이미지를 더 작은 크기로 스케일 다운하는 데 사용하는 전략을 설명합니다.
DVD_AUDIO_APPMODE IDvdInfo2::GetAudioAttributes 호출에서 검색된 현재 오디오 모드를 나타냅니다.
DVD_AUDIO_FORMAT DVD의 오디오 형식을 나타냅니다.
DVD_AUDIO_LANG_EXT 오디오 스트림에 오디오 언어 확장이 포함되어 있는지 여부를 나타내는 플래그를 정의합니다.
DVD_CMD_FLAGS DVD 탐색기 필터 필터가 명령 동기화를 처리하는 방법을 제어하는 플래그를 정의합니다.
DVD_DISC_SIDE DVD 디스크의 측면을 나타냅니다.
DVD_DOMAIN DVD 도메인을 정의합니다.

DVD_FRAMERATE DVD가 초당 25 또는 30프레임을 재생하도록 제작되었는지 여부를 나타냅니다.
DVD_KARAOKE_ASSIGNMENT 오디오 스트림에 대한 스피커 구성을 정의합니다.
DVD_KARAOKE_CONTENTS 비트 OR 연산에서 사용되는 경우 노래방 제목에 있는 오디오 스트림의 각 채널 콘텐츠를 설명하는 플래그를 지정합니다.
DVD_KARAOKE_DOWNMIX IDvdControl2::SelectKaraokeAudioPresentationMode 메서드에서 사용하는 플래그를 정의하여 각 보조 채널이 다운믹싱되는 스피커(있는 경우)를 제어합니다.
DVD_MENU_ID IDvdControl2::ShowMenu 호출에서 DVD 메뉴를 지정합니다.
DVD_NavCmdType DVD 탐색 명령 유형을 정의합니다.
DVD_OPTION_FLAG DVD_OPTION_FLAG 열거형은 DVD 탐색기 필터의 동작을 제어하는 플래그를 정의합니다. 이러한 플래그를 설정하려면 IDvdControl2::SetOption을 호출합니다.
DVD_PARENTAL_LEVEL DVD 사양에 정의된 일반 보호자 수준에 대한 플래그를 식별합니다.
DVD_PREFERRED_DISPLAY_MODE 사용자가 선호하는 창 가로 세로 비율 및 변환 방법을 나타냅니다.
DVD_RELATIVE_BUTTON DVD 메뉴 탐색 및 기타 작업에 사용되는 4가지 방향 단추를 식별합니다.
DVD_SUBPICTURE_CODING 하위 항목 스트림에 포함된 콘텐츠의 종류를 나타냅니다.
DVD_SUBPICTURE_LANG_EXT

지정된 하위픽처 스트림에서 가능한 언어 확장을 정의합니다.
DVD_SUBPICTURE_TYPE 하위 항목 스트림에 포함된 콘텐츠의 종류를 결정하는 데 사용되는 플래그를 정의합니다.
DVD_TextCharSet 지정된 문자열을 인코딩할 문자 집합을 정의합니다.
DVD_TextStringType DVD 텍스트 문자열 형식의 하위 집합을 정의합니다.
DVD_TIMECODE_FLAGS DVD가 재생되도록 작성된 프레임 속도를 나타냅니다.
DVD_TITLE_APPMODE DVD 타이틀이 노래방 제목인지 여부를 나타냅니다. 이 열거형은 애플리케이션이 IDvdInfo2::GetTitleAttributes 메서드를 호출할 때 채워지는 DVD_TitleAttributes 구조체의 멤버입니다.
DVD_VIDEO_COMPRESSION 가능한 DVD 비디오 압축 유형을 정의합니다.
FILTER_STATE 필터의 상태 또는 필터 그래프의 상태를 지정합니다.
InterleavingMode 비디오 프레임 및 오디오 샘플을 디스크에 쓰는 방법을 지정합니다.
PhysicalConnectorType 실제 핀 유형(오디오 또는 비디오)을 지정합니다.
PIN_DIRECTION 핀의 방향을 나타냅니다.
QualityMessageType 품질 메시지 유형을 설명합니다.
tagAM_SAMPLE_PROPERTY_FLAGS

AM_SAMPLE2_PROPERTIES 구조체의 dwSampleFlags 및 dwStreamId 멤버에 대한 값을 지정합니다. 이러한 값은 미디어 샘플의 속성을 설명합니다.
TunerInputType TV 튜너(케이블 또는 안테나)의 빈도를 지정합니다. (TunerInputType)
TVAudioMode TV 오디오 컨트롤의 모드를 지정합니다.
VALID_UOP_FLAG 현재 DVD에서 허용되는 UOP(사용자 작업) 명령을 나타냅니다.
VfwCaptureDialogs Windows용 비디오 캡처 드라이버에 있을 수 있는 대화 상자를 지정합니다.
VfwCompressDialogs Windows용 비디오 압축(코덱) 드라이버에 있을 수 있는 대화 상자를 지정합니다.
VideoControlFlags 비디오 디바이스의 비디오 작업 모드를 지정합니다.
VIDEOENCODER_BITRATE_MODE VIDEOENCODER_BITRATE_MODE 열거형 형식은 IEncoderAPI 인터페이스에서 지원하는 세 가지 비트 전송률 유형을 정의합니다.
VideoProcAmpFlags VideoProcAmpFlags 열거형은 특정 비디오 속성이 수동으로 또는 자동으로 제어되는지 여부를 나타냅니다.
VideoProcAmpProperty VideoProcAmpProperty 열거형은 비디오 캡처 디바이스의 비디오 속성을 지정합니다.
VMR_ASPECT_RATIO_MODE VMR_ASPECT_RATIO_MODE 열거형 형식은 비디오 혼합 렌더러 필터 7이 원본 비디오의 가로 세로 비율을 유지하는지 여부를 설명합니다.
VMRDeinterlacePrefs

VMRDeinterlacePrefs 열거형 형식은 애플리케이션에서 설정한 메서드를 사용할 수 없는 경우 비디오 혼합 렌더러 필터 7(VMR-7)에서 사용하는 디인터레이싱 메서드를 설명합니다.
VMRDeinterlaceTech VMRDeinterlaceTech 열거형 형식은 비디오 스트림을 디인터레이싱하는 데 사용되는 알고리즘을 설명합니다. 플래그는 상호 배타적이지 않습니다. 드라이버는 플래그의 조합을 설정할 수 있습니다.
VMRMixerPrefs VMRMixerPrefs 열거형에는 DirectDraw 화면에서 비디오 이미지를 만들 때 사용할 소수점, 필터링 및 색 공간 정보를 지정하는 플래그가 포함되어 있습니다.
VMRMode VMRMode 열거형 형식은 IVMRFilterConfig::GetRenderingMode 및 IVMRFilterConfig::SetRenderingMode 메서드를 호출하여 VMR-7(비디오 혼합 렌더러 필터 7) 렌더링 모드를 검색하거나 지정하는 데 사용됩니다.
VMRPresentationFlags VMRPresentationFlags 열거형 형식은 VMRPRESENTATIONINFO 구조의 멤버입니다.
VMRRenderPrefs VMRRenderPrefs 열거형 형식은 IVMRFilterConfig::GetRenderingPrefs 및 IVMRFilterConfig::SetRenderingPrefs 메서드와 함께 사용하여 기본 렌더링 기본 설정을 가져옵니다.
VMRSurfaceAllocationFlags VMRSurfaceAllocationFlags 열거형은 IVMRSurfaceAllocator::AllocateSurface 메서드와 함께 사용하여 표면 생성 매개 변수를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

strsafe.h 헤더

아티클2023. 08. 26.

이 헤더는 메뉴 및 기타 리소스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 메뉴 및 기타 리소스

strsafe.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

StringCbCatA 한 문자열을 다른 문자열에 연결합니다. (StringCbCatA)
StringCbCatExA 한 문자열을 다른 문자열에 연결합니다. (StringCbCatExA)
StringCbCatExW 한 문자열을 다른 문자열에 연결합니다. (StringCbCatExW)
StringCbCatNA 한 문자열에서 다른 문자열로 지정된 바이트 수를 연결합니다. (StringCbCatNA)
StringCbCatNExA 한 문자열에서 다른 문자열로 지정된 바이트 수를 연결합니다. (StringCbCatNExA)
StringCbCatNExW 한 문자열에서 다른 문자열로 지정된 바이트 수를 연결합니다. (StringCbCatNExW)
StringCbCatNW 한 문자열에서 다른 문자열로 지정된 바이트 수를 연결합니다. (StringCbCatNW)
StringCbCatW 한 문자열을 다른 문자열에 연결합니다. (StringCbCatW)
StringCbCopyA

한 문자열을 다른 문자열에 복사합니다. (StringCbCopyA)
StringCbCopyExA 한 문자열을 다른 문자열에 복사합니다. (StringCbCopyExA)
StringCbCopyExW 한 문자열을 다른 문자열에 복사합니다. (StringCbCopyExW)
StringCbCopyNA 지정된 바이트 수를 한 문자열에서 다른 문자열로 복사합니다. (StringCbCopyNA)
StringCbCopyNExA 지정된 바이트 수를 한 문자열에서 다른 문자열로 복사합니다. (StringCbCopyNExA)
StringCbCopyNExW 지정된 바이트 수를 한 문자열에서 다른 문자열로 복사합니다. (StringCbCopyNExW)
StringCbCopyNW 지정된 바이트 수를 한 문자열에서 다른 문자열로 복사합니다. (StringCbCopyNW)
StringCbCopyW 한 문자열을 다른 문자열에 복사합니다. (StringCbCopyW)
StringCbGetsA 줄 바꿈 문자('\n')를 포함하여 stdin에서 한 줄의 텍스트를 가져옵니다. (StringCbGetsA)
StringCbGetsExA 줄 바꿈 문자('\n')를 포함하여 stdin에서 한 줄의 텍스트를 가져옵니다. (StringCbGetsExA)
StringCbGetsExW 줄 바꿈 문자('\n')를 포함하여 stdin에서 한 줄의 텍스트를 가져옵니다. (StringCbGetsExW)
StringCbGetsW 줄 바꿈 문자('\n')를 포함하여 stdin에서 한 줄의 텍스트를 가져옵니다. (StringCbGetsW)
StringCbLengthA 문자열이 지정된 길이(바이트)를 초과하는지 여부를 결정합니다. (ANSI)

StringCbLengthW 문자열이 지정된 길이(바이트)를 초과하는지 여부를 결정합니다. (유니코드)
StringCbPrintf_lA 서식이 지정된 데이터를 지정된 문자열에 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCbPrintf_lA)
StringCbPrintf_lExA 서식이 지정된 데이터를 지정된 문자열에 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCbPrintf_lExA)
StringCbPrintf_lExW 서식이 지정된 데이터를 지정된 문자열에 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCbPrintf_lExW)
StringCbPrintf_lW 서식이 지정된 데이터를 지정된 문자열에 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCbPrintf_lW)
StringCbPrintfA 서식이 지정된 데이터를 지정된 문자열에 씁니다. (StringCbPrintfA)
StringCbPrintfExA 서식이 지정된 데이터를 지정된 문자열에 씁니다. (StringCbPrintfExA)
StringCbPrintfExW 서식이 지정된 데이터를 지정된 문자열에 씁니다. (StringCbPrintfExW)
StringCbPrintfW 서식이 지정된 데이터를 지정된 문자열에 씁니다. (StringCbPrintfW)
StringCbVPrintf_lA 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCbVPrintf_lA)
StringCbVPrintf_lExA 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCbVPrintf_lExA)

StringCbVPrintf_lExW 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCbVPrintf_lExW)
StringCbVPrintf_lW 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCbVPrintf_lW)
StringCbVPrintfA 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. (StringCbVPrintfA)
StringCbVPrintfExA 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. (StringCbVPrintfExA)
StringCbVPrintfExW 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. (StringCbVPrintfExW)
StringCbVPrintfW 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. (StringCbVPrintfW)
StringCchCatA 한 문자열을 다른 문자열에 연결합니다. (StringCchCatA)
StringCchCatExA 한 문자열을 다른 문자열에 연결합니다. (StringCchCatExA)
StringCchCatExW 한 문자열을 다른 문자열에 연결합니다. (StringCchCatExW)
StringCchCatNA 한 문자열에서 다른 문자열로 지정된 수의 문자를 연결합니다. (StringCchCatNA)
StringCchCatNExA 한 문자열에서 다른 문자열로 지정된 수의 문자를 연결합니다. (StringCchCatNExA)

StringCchCatNExW 한 문자열에서 다른 문자열로 지정된 수의 문자를 연결합니다. (StringCchCatNExW)
StringCchCatNW 한 문자열에서 다른 문자열로 지정된 수의 문자를 연결합니다. (StringCchCatNW)
StringCchCatW 한 문자열을 다른 문자열에 연결합니다. (StringCchCatW)
StringCchCopyA 한 문자열을 다른 문자열에 복사합니다. (StringCchCopyA)
StringCchCopyExA 한 문자열을 다른 문자열에 복사합니다. (StringCchCopyExA)
StringCchCopyExW 한 문자열을 다른 문자열에 복사합니다. (StringCchCopyExW)
StringCchCopyNA 한 문자열에서 다른 문자열로 지정된 수의 문자를 복사합니다. (StringCchCopyNA)
StringCchCopyNExA 한 문자열에서 다른 문자열로 지정된 수의 문자를 복사합니다. (StringCchCopyNExA)
StringCchCopyNExW 한 문자열에서 다른 문자열로 지정된 수의 문자를 복사합니다. (StringCchCopyNExW)
StringCchCopyNW 한 문자열에서 다른 문자열로 지정된 수의 문자를 복사합니다. (StringCchCopyNW)
StringCchCopyW 한 문자열을 다른 문자열에 복사합니다. (StringCchCopyW)
StringCchGetsA 줄 바꿈 문자('\n')를 포함하여 stdin에서 텍스트 한 줄을 가져옵니다. (StringCchGetsA)
StringCchGetsExA

줄 바꿈 문자('\n')를 포함하여 stdin에서 텍스트 한 줄을 가져옵니다. (StringCchGetsExA)
StringCchGetsExW 줄 바꿈 문자('\n')를 포함하여 stdin에서 텍스트 한 줄을 가져옵니다. (StringCchGetsExW)
StringCchGetsW 줄 바꿈 문자('\n')를 포함하여 stdin에서 텍스트 한 줄을 가져옵니다. (StringCchGetsW)
StringCchLengthA 문자열이 지정된 길이(문자)를 초과하는지 여부를 결정합니다. (ANSI)
StringCchLengthW 문자열이 지정된 길이(문자)를 초과하는지 여부를 결정합니다. (유니코드)
StringCchPrintf_lA 서식이 지정된 데이터를 지정된 문자열에 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCchPrintf_lA)
StringCchPrintf_lExA 서식이 지정된 데이터를 지정된 문자열에 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCchPrintf_lExA)
StringCchPrintf_lExW 서식이 지정된 데이터를 지정된 문자열에 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCchPrintf_lExW)
StringCchPrintf_lW 서식이 지정된 데이터를 지정된 문자열에 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCchPrintf_lW)
StringCchPrintfA 서식이 지정된 데이터를 지정된 문자열에 씁니다. (StringCchPrintfA)
StringCchPrintfExA 서식이 지정된 데이터를 지정된 문자열에 씁니다. (StringCchPrintfExA)
StringCchPrintfExW 서식이 지정된 데이터를 지정된 문자열에 씁니다. (StringCchPrintfExW)

StringCchPrintfW 서식이 지정된 데이터를 지정된 문자열에 씁니다. (StringCchPrintfW)
StringCchVPrintf_lA 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCchVPrintf_lA)
StringCchVPrintf_lExA 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCchVPrintf_lExA)
StringCchVPrintf_lExW 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCchVPrintf_lExW)
StringCchVPrintf_lW 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. 대상 버퍼의 크기는 이 버퍼의 끝을 지나서 작성되지 않도록 함수에 제공됩니다. (StringCchVPrintf_lW)
StringCchVPrintfA 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. (StringCchVPrintfA)
StringCchVPrintfExA 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. (StringCchVPrintfExA)
StringCchVPrintfExW 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. (StringCchVPrintfExW)
StringCchVPrintfW 인수 목록에 대한 포인터를 사용하여 지정된 문자열에 서식이 지정된 데이터를 씁니다. (StringCchVPrintfW)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

structuredquery.h 헤더

아티클2023. 08. 26.

이 헤더는 Windows Search에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 검색

structuredquery.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IConditionFactory 쿼리 문자열을 구문 분석하여 가져온 조건 트리를 만들거나 확인하는 메서드를 제공합니다.
IConditionFactory2 IConditionFactory의 기능을 확장합니다. IConditionFactory2는 쿼리 문자열을 구문 분석하여 얻은 조건 트리를 만들거나 해결하는 메서드를 제공합니다.
IConditionGenerator 명명된 엔터티를 처리하고 특수 조건을 생성하는 메서드를 제공합니다.
IEntity 스키마의 엔터티 형식에 대한 정보를 검색하는 메서드를 제공합니다.
IInterval 간격의 제한을 가져오는 메서드를 제공합니다.
IMetaData IEntity, IRelationship 또는 ISchemaProvider 개체에서 문자열의 키/값 쌍을 검색하는 메서드를 제공합니다.
INamedEntity 명명된 엔터티의 값 또는 값에 대한 기본 구를 가져오는 메서드를 제공합니다.
INamedEntityCollector IConditionGenerator 개체로 식별되는 명명된 엔터티를 누적하는 메서드를 제공합니다.

IQueryParser 입력 문자열을 IQuerySolution 개체로 구문 분석하는 메서드를 제공합니다.
IQueryParserManager IQueryParser 개체에 대한 옵션을 만들고, 초기화하고, 변경하는 메서드를 제공합니다.
IQuerySolution 구문 분석된 쿼리의 해석에 대한 정보를 검색하는 메서드를 제공합니다.
IRelationship 스키마 속성에 대한 정보를 검색하는 메서드를 제공합니다.
ISchemaLocalizerSupport 지정된 문자열에서 키워드를 지역화하는 메서드를 제공합니다.
ISchemaProvider 검색할 수 있는 스키마 리포지토리를 제공합니다.
ITokenCollection 단어 분리기를 사용하여 발생하는 토큰을 가져옵니다.

구조체

HITRANGE 쿼리 검색 조건이 인덱싱된 데이터와 일치하는 경우 일치하는 데이터의 범위를 식별합니다.

열거형

CASE_REQUIREMENT 쿼리에 대한 키워드의 사례 요구 사항(있는 경우)을 지정합니다.

CONDITION_CREATION_OPTIONS 조건 트리 노드의 형식을 나타내기 위해 다음 인터페이스와 함께 사용할 플래그 집합을 제공합니다._ICondition, ICondition2, IConditionFactory, IConditionFactory2 및 IConditionGenerator.
INTERVAL_LIMIT_KIND 이러한 값은 IInterval::GetLimits에서 쌍으로 반환되어 상한 및 하한이 있는 범위를 지정합니다.
NAMED_ENTITY_CERTAINTY 명명된 엔터티에 대한 확실성 수준을 정의합니다.
QUERY_PARSER_MANAGER_OPTION IQueryParserManager::SetOption에서 구문 분석 옵션을 설정하는 데 사용됩니다. 스키마 및 지역화 옵션을 지정하는 데 사용할 수 있습니다.
STRUCTURED_QUERY_MULTIOPTION 개별 옵션을 나타내기 위해 IQueryParser::SetMultiOption에서 사용하는 플래그 집합입니다.
STRUCTURED_QUERY_PARSE_ERROR 구문 분석 오류를 식별하기 위해 IQuerySolution::GetErrors와 함께 사용할 플래그 집합입니다. 각 구문 분석 오류는 쿼리 문자열을 구문 분석할 때 하나 이상의 토큰이 무시되었음을 나타냅니다.
STRUCTURED_QUERY_RESOLVE_OPTION 조건 트리로 데이터를 확인하는 옵션입니다.
STRUCTURED_QUERY_SINGLE_OPTION 개별 옵션을 나타내기 위해 IQueryParser::SetOption 및 IQueryParser::GetOption과 함께 사용할 플래그 집합입니다.
STRUCTURED_QUERY_SYNTAX 쿼리 구문의 형식을 지정합니다. (STRUCTURED_QUERY_SYNTAX)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

structuredquerycondition.h 헤더

아티클2023. 08. 26.

이 헤더는 Windows Search에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 검색

structuredquerycondition.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICondition 검색 조건에 대한 정보를 검색하는 메서드를 제공합니다.
ICondition2 ICondition 인터페이스의 기능을 확장합니다. ICondition2는 검색 조건에 대한 정보를 검색하는 메서드를 제공합니다.
IRichChunk 데이터 청크를 문자열 및 PROPVARIANT 값으로 나타냅니다.

열거형

CONDITION_OPERATION ICondition::GetComparisonInfo에서 작업을 나타내기 위해 다음 메서드와 함께 사용할 플래그 집합을 제공합니다. ICondition2::GetLeafConditionInfo, IConditionFactory::MakeLeaf, IConditionFactory2::CreateBooleanLeaf, IConditionFactory2::CreateIntegerLeaf, IConditionFactory2::MakeLeaf, IConditionFactory2::CreateStringLeaf 및 IConditionGenerator::GenerateForLeaf.
CONDITION_TYPE 조건 트리 노드:_ICondition::GetConditionType, IConditionFactory::MakeAndOr, IConditionFactory2::CreateCompoundFromArray 및 IConditionFactory2::CreateCompoundFromObjectArray의 형식을 나타내기 위해 다음 메서드와 함께 사용할 플래그 집합을 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

subauth.h 헤더

아티클2023. 08. 26.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

subauth.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

Msv1_0SubAuthenticationFilter 도메인 컨트롤러와 관련된 사용자 로그온 인증을 수행합니다.
Msv1_0SubAuthenticationRoutine 클라이언트/서버별 인증을 수행합니다.
Msv1_0SubAuthenticationRoutineEx LogonUser 함수를 호출하여 하위 인증을 요청할 때 원격 액세스 서비스 인증을 수행합니다.
Msv1_0SubAuthenticationRoutineGeneric LsaCallAuthenticationPackage 함수를 호출하여 하위 인증이 요청되면 원격 액세스 서비스 인증을 수행합니다.

구조체

NETLOGON_LOGON_IDENTITY_INFO 로그온 하위 인증을 위해 사용자에 대한 정보를 전달하는 데 사용됩니다.
OLD_LARGE_INTEGER 64비트 부가 정수 값을 두 개의 32비트 정수로 나타내는 데 사용됩니다.
SR_SECURITY_DESCRIPTOR SR_SECURITY_DESCRIPTOR 구조에는 사용자의 보안 권한에 대한 정보가 포함됩니다.

UNICODE_STRING 다양한 LSA(로컬 보안 기관) 함수에서 유니코드 문자열을 지정하는 데 사용됩니다.
USER_ALL_INFORMATION 세션 사용자에 대한 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

subsmgr.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows Search에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 검색

subsmgr.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

ITEMPROP Windows 속성 시스템의 속성에 대한 정보를 저장하고 IItemPropertyBag 인터페이스에서 사용됩니다.

swdevice.h 헤더

아티클2023. 03. 08.

이 헤더는 소프트웨어 디바이스 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 소프트웨어 디바이스 API

swdevice.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

SwDeviceClose 소프트웨어 디바이스 핸들을 닫습니다. 핸들이 닫히면 SwDevice의 수명이 SWDeviceLifetimeHandle인 경우 PnP는 디바이스를 "플러깅 해제" 프로세스를 시작합니다. 디바이스는 더 이상 부모 디바이스의 자식으로 보고되지 않습니다.
SwDeviceCreate 소프트웨어 디바이스의 열거형을 시작합니다.
SwDeviceGetLifetime 소프트웨어 디바이스의 수명을 가져옵니다.
SwDeviceInterfacePropertySet 소프트웨어 디바이스 인터페이스의 속성을 설정합니다.
SwDeviceInterfaceRegister 소프트웨어 디바이스에 대한 디바이스 인터페이스를 등록하고 필요에 따라 해당 인터페이스의 속성을 설정합니다.
SwDeviceInterfaceSetState 소프트웨어 디바이스에 대한 디바이스 인터페이스를 사용하거나 사용하지 않도록 설정합니다.
SwDevicePropertySet 소프트웨어 디바이스의 속성을 설정합니다.
SwDeviceSetLifetime 소프트웨어 디바이스의 수명을 관리합니다.

SwMemFree 다른 소프트웨어 디바이스 API 함수가 할당한 메모리를 해제합니다.

콜백 함수

SW_DEVICE_CREATE_CALLBACK 레지스트리에서 지원되는 디바이스를 제공하고 호출자가 hSwDevice 핸들을 사용하여 소프트웨어 디바이스 API 함수를 호출할 수 있도록 합니다.

swdevicedef.h 헤더

아티클2023. 03. 08.

이 헤더는 소프트웨어 디바이스 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 소프트웨어 디바이스 API

swdevicedef.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

SW_DEVICE_CREATE_INFO PnP가 소프트웨어 디바이스를 만드는 데 사용하는 정보를 설명합니다.

synchapi.h 헤더

아티클2023. 08. 26.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

synchapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AcquireSRWLockExclusive 전용 모드에서 SRW(슬림 판독기/기록기) 잠금을 획득합니다.
AcquireSRWLockShared 공유 모드에서 SRW(슬림 판독기/기록기) 잠금을 획득합니다.
CancelWaitableTimer 지정된 대기 가능 타이머를 비활성 상태로 설정합니다.
CreateEventA 명명되거나 명명되지 않은 이벤트 개체를 만들거나 엽니다. (ANSI)
CreateEventExA 명명되거나 명명되지 않은 이벤트 개체를 만들거나 열고 개체에 대한 핸들을 반환합니다. (ANSI)
CreateEventExW 명명되거나 명명되지 않은 이벤트 개체를 만들거나 열고 개체에 대한 핸들을 반환합니다. (유니코드)
CreateEventW 명명되거나 명명되지 않은 이벤트 개체를 만들거나 엽니다. (유니코드)
CreateMutexA 명명되거나 명명되지 않은 뮤텍스 개체를 만들거나 엽니다. (ANSI)
CreateMutexExA

명명되거나 명명되지 않은 뮤텍스 개체를 만들거나 열고 개체에 대한 핸들을 반환합니다. (ANSI)
CreateMutexExW 명명되거나 명명되지 않은 뮤텍스 개체를 만들거나 열고 개체에 대한 핸들을 반환합니다. (유니코드)
CreateMutexW 명명되거나 명명되지 않은 뮤텍스 개체를 만들거나 엽니다. (유니코드)
CreateSemaphoreExW 명명되거나 명명되지 않은 세마포 개체를 만들거나 열고 개체에 대한 핸들을 반환합니다. (CreateSemaphoreExW)
CreateSemaphoreW 명명되거나 명명되지 않은 세마포 개체를 만들거나 엽니다. (CreateSemaphoreW)
CreateWaitableTimerExW 대기 가능한 타이머 개체를 만들거나 열고 개체에 대한 핸들을 반환합니다.
CreateWaitableTimerW 대기 가능한 타이머 개체를 만들거나 엽니다.
DeleteCriticalSection 소유하지 않은 중요한 섹션 개체에서 사용하는 모든 리소스를 해제합니다.
DeleteSynchronizationBarrier 동기화 장벽을 삭제합니다.
EnterCriticalSection 지정된 임계 영역 개체의 소유권을 기다립니다. 함수가 호출 스레드가 소유권을 부여받는 시기를 반환합니다.
EnterSynchronizationBarrier 최대 스레드 수가 장벽에 들어갈 때까지 호출 스레드가 동기화 장벽에서 대기하도록 합니다.
InitializeConditionVariable 조건 변수를 초기화합니다.

InitializeCriticalSection 중요한 섹션 개체를 초기화합니다.
InitializeCriticalSectionAndSpinCount 중요한 섹션 개체를 초기화하고 중요한 섹션의 스핀 수를 설정합니다.
InitializeCriticalSectionEx 스핀 수 및 선택적 플래그를 사용하여 중요한 섹션 개체를 초기화합니다.
InitializeSRWLock SRW(슬림 판독기/기록기) 잠금을 초기화합니다.
InitializeSynchronizationBarrier 새 동기화 장벽을 초기화합니다.
InitOnceBeginInitialize 일회성 초기화를 시작합니다.
InitOnceComplete InitOnceBeginInitialize 함수로 시작된 일회성 초기화를 완료합니다.
InitOnceExecuteOnce 지정된 함수를 한 번 성공적으로 실행합니다. 동일한 일회성 초기화 구조를 지정하는 다른 스레드는 현재 스레드에서 실행되는 동안 지정된 함수를 실행할 수 없습니다.
InitOnceInitialize 일회성 초기화 구조를 초기화합니다.
LeaveCriticalSection 지정된 중요 섹션 개체의 소유권을 해제합니다.
OpenEventA 명명된 기존 이벤트 개체를 엽니다. (ANSI)
OpenEventW 명명된 기존 이벤트 개체를 엽니다. (유니코드)

OpenMutexW 명명된 기존 뮤텍스 개체를 엽니다.
OpenSemaphoreW 명명된 기존 세마포 개체를 엽니다.
OpenWaitableTimerW 기존 명명된 대기 가능한 타이머 개체를 엽니다.
ReleaseMutex 지정된 뮤텍스 개체의 소유권을 해제합니다.
ReleaseSemaphore 지정된 세마포 개체의 개수를 지정된 양만큼 늘입니다.
ReleaseSRWLockExclusive 독점 모드에서 획득한 SRW(슬림 판독기/기록기) 잠금을 해제합니다.
ReleaseSRWLockShared 공유 모드에서 획득한 SRW(슬림 판독기/기록기) 잠금을 해제합니다.
ResetEvent 지정된 이벤트 개체를 부호 없는 상태로 설정합니다.
SetCriticalSectionSpinCount 지정된 중요 섹션의 스핀 수를 설정합니다.
SetEvent 지정된 이벤트 개체를 신호 상태로 설정합니다.
SetWaitableTimer 지정된 대기 가능 타이머를 활성화합니다. 기한이 되면 타이머에 신호가 전송되고 타이머를 설정하는 스레드는 선택적 완료 루틴을 호출합니다.
SetWaitableTimerEx 지정된 대기 가능 타이머를 활성화하고 타이머에 대한 컨텍스트 정보를 제공합니다. 기한이 되면 타이머에 신호가 전송되고 타이머를 설정하는 스레드는 선택적 완료 루틴을 호출합니다.

SignalObjectAndWait 한 개체에 신호를 전송하고 다른 개체를 단일 작업으로 기다립니다.
Sleep 제한 시간 간격이 경과할 때까지 현재 스레드의 실행을 일시 중단합니다.
SleepConditionVariableCS 지정된 조건 변수에서 절전 모드로 전환하고 지정된 중요 섹션을 원자성 연산으로 해제합니다.
SleepConditionVariableSRW 지정된 조건 변수에서 절전 모드로 전환하고 지정된 잠금을 원자성 작업으로 해제합니다.
SleepEx 지정된 조건이 충족될 때까지 현재 스레드를 일시 중단합니다.
TryAcquireSRWLockExclusive 전용 모드에서 SRW(슬림 판독기/기록기) 잠금을 획득하려고 시도합니다. 호출에 성공하면 호출 스레드가 잠금의 소유권을 맡습니다.
TryAcquireSRWLockShared 공유 모드에서 SRW(슬림 판독기/기록기) 잠금을 획득하려고 시도합니다. 호출에 성공하면 호출 스레드가 잠금의 소유권을 맡습니다.
TryEnterCriticalSection 차단하지 않고 중요한 섹션을 입력하려고 시도합니다. 호출이 성공하면 호출 스레드는 중요한 섹션의 소유권을 맡습니다.
WaitForMultipleObjects 지정된 개체 중 하나 또는 전부가 신호 상태 또는 시간 제한 간격이 경과할 때까지 기다립니다.
WaitForMultipleObjectsEx 지정된 개체 중 하나 또는 전부가 신호 상태에 있거나, I/O 완료 루틴 또는 APC(비동기 프로시저 호출)가 스레드에 큐에 대기되거나 시간 제한 간격이 경과할 때까지 기다립니다.
WaitForSingleObject 지정된 개체가 신호 상태가 되거나 시간 제한 간격이 경과할 때까지 기다립니다.

WaitForSingleObjectEx 지정된 개체가 신호 상태에 있거나, I/O 완료 루틴 또는 APC(비동기 프로시저 호출)가 스레드에 큐에 대기되거나 시간 제한 간격이 경과할 때까지 기다립니다.
WaitOnAddress 지정된 주소의 값이 변경될 때까지 기다립니다.
WakeAllConditionVariable 지정된 조건 변수에서 대기 중인 모든 스레드를 절 해제합니다.
WakeByAddressAll 주소 값이 변경되기를 기다리는 모든 스레드를 절 해제합니다.
WakeByAddressSingle 주소 값이 변경되는 것을 기다리는 스레드 하나를 절 해제합니다.
WakeConditionVariable 지정된 조건 변수에서 대기 중인 단일 스레드를 절 해제합니다.

콜백 함수

PINIT_ONCE_FN 애플리케이션 정의 콜백 함수입니다. InitOnceExecuteOnce 함수를 호출할 때 이 함수에 대한 포인터를 지정합니다.
PTIMERAPCROUTINE 애플리케이션에서 정의한 타이머 완료 루틴입니다. SetWaitableTimer 함수를 호출할 때 이 주소를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

syncmgr.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

syncmgr.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumSyncMgrConflict 충돌 열거형 메서드를 노출합니다.
IEnumSyncMgrEvents 동기화 이벤트 열거형 메서드를 노출합니다.
IEnumSyncMgrSyncItems 처리기에서 관리하는 동기화 항목 개체를 열거하는 메서드를 노출합니다.
ISyncMgrConflict 충돌 저장소에서 검색된 충돌에 대한 정보를 제공하고 충돌을 해결할 수 있는 메서드를 노출합니다.
ISyncMgrConflictFolder 충돌 개체의 충돌 ID 목록을 가져오는 메서드를 노출합니다.
ISyncMgrConflictItems 충돌 항목 데이터 및 항목 수를 가져오는 메서드를 노출합니다.
ISyncMgrConflictPresenter 사용자에게 충돌을 표시하는 메서드를 노출합니다.
ISyncMgrConflictResolutionItems 항목 정보 및 항목 수를 가져오는 메서드를 노출합니다.

ISyncMgrConflictResolveInfo 동기화 관리자 충돌 해결에 대한 정보를 가져와서 설정하는 메서드를 노출합니다.
ISyncMgrConflictStore 처리기가 Conflicts 폴더에 표시되는 충돌을 제공할 수 있도록 하는 메서드를 노출합니다.
ISyncMgrControl 애플리케이션 또는 처리기가 동기화를 시작하거나 중지하거나, 동기화 센터에 처리기 또는 항목 집합에 대한 변경 내용을 알리거나, 속성 값의 변경 내용을 알릴 수 있는 메서드를 노출합니다.
ISyncMgrEvent 이벤트 저장소에서 데이터를 검색하는 메서드를 노출합니다. 이벤트 저장소를 사용하면 동기화 센터에서 저장소에 있는 모든 이벤트의 열거자를 가져올 뿐만 아니라 개별 이벤트를 검색할 수 있습니다.
ISyncMgrEventLinkUIOperation 동기화 결과 폴더에서 이벤트 링크를 클릭할 때 호출되는 메서드를 제공합니다.
ISyncMgrEventStore 처리기가 기본 동기화 센터 이벤트 저장소를 사용하는 대신 자체 이벤트 저장소를 제공하고 자체 동기화 이벤트를 관리할 수 있도록 하는 메서드를 노출합니다. 이러한 이벤트는 동기화 결과 폴더에 표시됩니다.
ISyncMgrHandler 동기화 처리기에 의해 구현된 기본 인터페이스를 구성하는 메서드를 노출합니다.
ISyncMgrHandlerCollection 동기화 처리기 ID의 열거자를 제공하고 해당 동기화 처리기를 인스턴스화하는 메서드를 노출합니다.
ISyncMgrHandlerInfo 처리기가 동기화 센터에 속성 및 상태 정보를 제공할 수 있도록 하는 메서드를 노출합니다.
ISyncMgrResolutionHandler 동기화 충돌을 관리하는 메서드를 노출합니다. 이 인터페이스를 구현하여 동기화 충돌 처리기를 생성합니다. 충돌 해결 UI(사용자 인터페이스)는 사용자에게 표시되는 충돌을 해결하기 위해 이 인터페이스를 호출합니다.

ISyncMgrScheduleWizardUIOperation 처리기가 처리기의 동기화 일정 마법사를 표시할 수 있도록 하는 메서드를 노출합니다.
ISyncMgrSessionCreator 처리기 또는 외부 애플리케이션이 동기화가 시작되었음을 동기화 센터에 알리고 진행률 및 이벤트를 보고할 수 있는 단일 메서드를 노출합니다.
ISyncMgrSyncCallback 동기화 프로세스에서 진행률 및 이벤트를 동기화 센터에 보고하거나 프로세스가 취소되었는지 여부를 쿼리할 수 있는 메서드를 노출합니다.
ISyncMgrSyncItem 처리기가 동기화 항목을 독립적인 개체로 관리할 수 있도록 단일 동기화 항목에서 작동 및 정보를 검색하는 메서드를 노출합니다.
ISyncMgrSyncItemContainer 포함된 항목에 대한 정보를 처리기에 제공하는 메서드를 노출합니다.
ISyncMgrSyncItemInfo 단일 동기화 항목에 대한 속성 및 상태 정보를 제공하는 메서드를 노출합니다.
ISyncMgrSyncResult ISyncMgrControl을 호출하는 애플리케이션이 ISyncMgrControl::StartHandlerSync 또는 ISyncMgrControl::StartItemSync 호출의 결과를 가져오는 데 사용할 수 있는 메서드를 노출합니다.
ISyncMgrUIOperation 동기화 센터에서 요청할 때 동기화 처리기 또는 동기화 항목이 UI 개체를 표시할 수 있는 메서드를 노출합니다.

Functions

FreeConfirmConflictItem CONFIRM_CONFLICT_ITEM 구조에 할당된 리소스를 해제합니다.

구조체

CONFIRM_CONFLICT_ITEM 충돌 항목 구조를 정의합니다.
CONFIRM_CONFLICT_RESULT_INFO 충돌 결과 정보 구조를 정의합니다.
SYNCMGR_CONFLICT_ID_INFO 충돌 ID 정보 구조를 설명합니다.

열거형

SYNCMGR_CANCEL_REQUEST 동기화를 취소하라는 사용자의 요청에 대해 설명합니다.
SYNCMGR_CONFLICT_ITEM_TYPE 충돌 항목 유형에 대해 설명합니다.
SYNCMGR_CONTROL_FLAGS ISyncMgrControl의 특정 메서드에 대해 요청된 작업을 수행하는 방법을 지정합니다.
SYNCMGR_EVENT_FLAGS 동기화 이벤트에 대한 플래그를 지정합니다.
SYNCMGR_EVENT_LEVEL 동기화 센터에 보고되는 이벤트의 유형을 지정합니다.
SYNCMGR_HANDLER_CAPABILITIES 처리기에 대해 수행할 수 있는 동작과 관련된 처리기의 기능을 지정합니다.
SYNCMGR_HANDLER_POLICIES 기본 정책에서 벗어나는 동기화 처리기에서 지정한 정책을 열거합니다.

SYNCMGR_HANDLER_TYPE 처리기의 형식을 지정합니다. ISyncMgrHandlerInfo::GetType에서 사용됩니다.
SYNCMGR_ITEM_CAPABILITIES 항목에 대해 수행할 수 있는 작업을 지정합니다.
SYNCMGR_ITEM_POLICIES 그룹 정책에 의해 사용하도록 설정하거나 사용하지 않도록 설정할 수 있는 방법을 제어하는 항목의 정책을 지정합니다.
SYNCMGR_PRESENTER_CHOICE 사용자가 동기화 관리자 충돌 해결에 대해 어떤 선택을 하는지 설명합니다. ISyncMgrConflictPresenter에서 사용됩니다.
SYNCMGR_PRESENTER_NEXT_STEP 동기화 관리자 충돌 해결에서 발생하는 다음 단계를 설명합니다. ISyncMgrConflictPresenter에서 사용됩니다.
SYNCMGR_PROGRESS_STATUS 동기화 프로세스의 현재 진행 상태를 지정합니다. ISyncMgrSyncCallback::ReportProgress에서 사용됩니다.
SYNCMGR_RESOLUTION_ABILITIES 수행할 기능 및 충돌 해결 활동을 나타냅니다. ISyncMgrResolutionHandler::QueryAbilities와 함께 사용됩니다.
SYNCMGR_RESOLUTION_FEEDBACK 동기화 관리자 확인 피드백에 대해 설명합니다. ISyncMgrResolutionHandler에서 사용됩니다.
SYNCMGR_SYNC_CONTROL_FLAGS ISyncMgrControl::StartHandlerSync 및 ISyncMgrControl::StartItemSync에서 사용하는 플래그를 나타냅니다.

syncregistration.h 헤더

아티클2023. 08. 26.

이 헤더는 Windows Sync에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 동기화

syncregistration.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumSyncProviderConfigUIInfos 동기화 공급자를 빌드하고 등록하는 데 사용되는 구성 UI 정보가 포함된 ISyncProviderConfigUIInfo 개체를 열거합니다.
IEnumSyncProviderInfos 동기화 공급자의 instance 만드는 데 사용되는 정보가 포함된 ISyncProviderInfo 개체를 열거합니다.
IRegisteredSyncProvider 등록된 동기화 공급자를 나타냅니다. 이 인터페이스는 동기화 공급자의 작성기에 의해 구현됩니다.
ISyncProviderConfigUI 동기화 공급자를 빌드하고 등록하는 데 사용되는 구성 UI 정보를 나타냅니다.
ISyncProviderConfigUIInfo 동기화 공급자 구성 UI의 instance 만드는 데 필요한 정보 및 속성을 나타냅니다.
ISyncProviderInfo 동기화 공급자의 instance 만드는 데 필요한 정보 및 속성을 나타냅니다.
ISyncProviderRegistration 동기화 공급자 등록을 나타냅니다.
ISyncRegistrationChange 동기화 공급자 또는 동기화 공급자 구성 UI 등록에 대한 변경 사항을 나타냅니다. 변경 내용은 등록 이벤트로 보고됩니다.

구조체

SyncProviderConfigUIConfiguration 동기화 공급자 구성 UI에 대한 정보를 나타냅니다.
SyncProviderConfiguration 동기화 공급자 구성에 대한 정보를 나타냅니다.

열거형

SYNC_REGISTRATION_EVENT 다양한 유형의 동기화 등록 이벤트를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

sysinfoapi.h 헤더

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- System Services
- Windows 사이드바

sysinfoapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

EnumSystemFirmwareTables 지정된 형식의 모든 시스템 펌웨어 테이블을 열거합니다.
GetComputerNameExA 로컬 컴퓨터와 연결된 NetBIOS 또는 DNS 이름을 검색합니다. 이름은 시스템이 레지스트리에서 읽을 때 시스템 시작 시 설정됩니다. (ANSI)
GetComputerNameExW 로컬 컴퓨터와 연결된 NetBIOS 또는 DNS 이름을 검색합니다. 이름은 시스템이 레지스트리에서 읽을 때 시스템 시작 시 설정됩니다. (유니코드)
GetDeveloperDriveEnablementState 개발자 드라이브를 사용할 수 있는지 여부를 나타내는 값을 가져옵니다.
GetIntegratedDisplaySize 기본 제공 화면의 대각선 크기를 인치 단위로 가장 잘 예측합니다.
GetLocalTime 현재 로컬 날짜 및 시간을 검색합니다.
GetLogicalProcessorInformation 논리 프로세서 및 관련 하드웨어에 대한 정보를 검색합니다.
GetLogicalProcessorInformationEx 논리 프로세서 및 관련 하드웨어의 관계에 대한 정보를 검색합니다.

GetNativeSystemInfo WOW64에서 실행되는 애플리케이션에 현재 시스템에 대한 정보를 검색합니다.
GetPhysicallyInstalledSystemMemory 컴퓨터에 물리적으로 설치된 RAM의 양을 검색합니다.
GetProcessorSystemCycleTime 프로세서가 활성화된 이후 지정된 프로세서 그룹의 각 프로세서가 DPC(지연 프로시저 호출) 및 ISR(인터럽트 서비스 루틴)을 실행하는 데 소요된 주기 시간을 검색합니다.
GetProductInfo 로컬 컴퓨터에서 운영 체제의 제품 유형을 검색하고 지정된 운영 체제에서 지원하는 제품 유형에 형식을 매핑합니다.
GetRuntimeAttestationReport 서명된 런타임 증명 보고서 검색
GetSystemDirectoryA 시스템 디렉터리의 경로를 검색합니다. (ANSI)
GetSystemDirectoryW 시스템 디렉터리의 경로를 검색합니다. (유니코드)
GetSystemFirmwareTable 펌웨어 테이블 공급자에서 지정된 펌웨어 테이블을 검색합니다.
GetSystemInfo 현재 시스템에 대한 정보를 검색합니다.
GetSystemTime 현재 시스템 날짜 및 시간을 UTC(협정 세계시) 형식으로 검색합니다.
GetSystemTimeAdjustment 시스템에서 정기적인 시간 조정을 해당 시간 시계에 적용하는지 여부를 결정하고 이러한 조정의 값과 기간을 가져옵니다.
GetSystemTimeAdjustmentPrecise

시스템이 시간 시계에 주기적이고 프로그래밍된 시간 조정을 적용하는지 여부를 결정하고 이러한 조정의 값과 기간을 얻습니다.
GetSystemTimeAsFileTime 현재 시스템 날짜 및 시간을 검색합니다. 정보는 UTC(협정 세계시) 형식입니다.
GetSystemTimePreciseAsFileTime GetSystemTimePreciseAsFileTime 함수는 가능한 전체 자릿수(<1us)가 가장 높은 현재 시스템 날짜 및 시간을 검색합니다. 검색된 정보는 UTC(협정 세계시) 형식입니다.
GetSystemWindowsDirectoryA 다중 사용자 시스템에서 공유 Windows 디렉터리의 경로를 검색합니다. (ANSI)
GetSystemWindowsDirectoryW 다중 사용자 시스템에서 공유 Windows 디렉터리의 경로를 검색합니다. (유니코드)
GetTickCount 시스템이 시작된 후 경과된 시간(최대 49.7일)을 검색합니다.
GetTickCount64 시스템이 시작된 후 경과된 시간(밀리초)을 검색합니다.
GetVersion Windows 8.1 릴리스에서는 GetVersion API의 동작이 운영 체제 버전에 대해 반환되는 값으로 변경되었습니다. 이제 GetVersion 함수에서 반환되는 값은 애플리케이션의 매니페스트 방식에 따라 달라집니다.
GetVersionExA Windows 8.1 릴리스에서는 GetVersionEx API의 동작이 운영 체제 버전에 대해 반환되는 값으로 변경되었습니다. 이제 GetVersionEx 함수에서 반환되는 값은 애플리케이션이 매니페스트되는 방법에 따라 달라집니다. (ANSI)
GetVersionExW Windows 8.1 릴리스에서는 GetVersionEx API의 동작이 운영 체제 버전에 대해 반환되는 값으로 변경되었습니다. 이제 GetVersionEx 함수에서 반환되는 값은 애플리케이션이 매니페스트되는 방법에 따라 달라집니다. (유니코드)
GetWindowsDirectoryA Windows 디렉터리의 경로를 검색합니다. (ANSI)

GetWindowsDirectoryW Windows 디렉터리의 경로를 검색합니다. (유니코드)
GlobalMemoryStatusEx 물리적 메모리와 가상 메모리의 현재 사용량에 대한 정보를 검색합니다. (GlobalMemoryStatusEx)
InstallELAMCertificateInfo 빌드 시 ELAM 드라이버에 연결된 리소스 파일에 지정된 인증서 정보를 설치합니다.
IsUserCetAvailableInEnvironment 지정된 환경에서 사용자 모드 하드웨어 적용 스택 보호를 사용할 수 있는지 여부를 쿼리합니다.
SetComputerNameA 로컬 컴퓨터의 새 NetBIOS 이름을 설정합니다. 이름은 레지스트리에 저장되며 다음번에 사용자가 컴퓨터를 다시 시작할 때 이름 변경이 적용됩니다. (ANSI)
SetComputerNameExA 로컬 컴퓨터의 새 NetBIOS 또는 DNS 이름을 설정합니다. (ANSI)
SetComputerNameExW 로컬 컴퓨터의 새 NetBIOS 또는 DNS 이름을 설정합니다. (유니코드)
SetComputerNameW 로컬 컴퓨터의 새 NetBIOS 이름을 설정합니다. 이름은 레지스트리에 저장되며 다음번에 사용자가 컴퓨터를 다시 시작할 때 이름 변경이 적용됩니다. (유니코드)
SetLocalTime 현재 현지 시간 및 날짜를 설정합니다.
SetSystemTime 현재 시스템 시간과 날짜를 설정합니다. 시스템 시간은 UTC(협정 세계시)로 표시됩니다.
SetSystemTimeAdjustment 시스템의 시간 시계에 대한 주기적인 시간 조정을 사용하거나 사용하지 않도록 설정합니다. 사용하도록 설정하면 이러한 시간 조정을 사용하여 시간을 다른 시간 원본 정보와 동기화할 수 있습니다. (SetSystemTimeAdjustment)

SetSystemTimeAdjustmentPrecise 시스템의 시간 시계에 대한 주기적인 시간 조정을 사용하거나 사용하지 않도록 설정합니다. 사용하도록 설정하면 이러한 시간 조정을 사용하여 시간을 다른 시간 원본 정보와 동기화할 수 있습니다. (SetSystemTimeAdjustmentPrecise)

구조

테이블 확장

MEMORYSTATUSEX 확장 메모리를 포함하여 실제 메모리와 가상 메모리의 현재 상태에 대한 정보를 포함합니다.
SYSTEM_INFO 현재 컴퓨터 시스템에 대한 정보를 포함합니다. 여기에는 프로세서의 아키텍처 및 유형, 시스템의 프로세서 수, 페이지 크기 및 기타 정보가 포함됩니다.

열거 목록

테이블 확장

COMPUTER_NAME_FORMAT 컴퓨터 이름의 유형을 지정합니다.
DEVELOPER_DRIVE_ENABLEMENT_STATE 개발자 드라이브 사용 상태의 가능한 값 열거형입니다.

Last updated on 2026. 01. 23.

systemmediatransportcontrolsinterop.h 헤더

아티클2023. 08. 26.

이 헤더는 WinRT 상호 운용성에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- WinRT 상호 운용성
- SystemMediaTransportControls 클래스

systemmediatransportcontrolsinterop.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ISystemMediaTransportControlsInterop 앱에서 ISystemMediaTransportControls 인터페이스의 instance 가져올 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

systemtopologyapi.h 헤더

아티클2023. 08. 26.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

systemtopologyapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

GetNumaHighestNodeNumber 현재 숫자가 가장 높은 노드를 검색합니다.
GetNumaNodeProcessorMask2 지정된 노드의 다중 그룹 프로세서 마스크를 검색합니다.
GetNumaNodeProcessorMaskEx 노드가 속한 프로세서 그룹에 관계없이 노드의 프로세서 마스크를 검색합니다.
GetNumaProximityNodeEx 지정된 근접 식별자에 해당하는 NUMA 노드 번호를 USHORT 값으로 검색합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

t2embapi.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows GDI에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows GDI

t2embapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

TTCharToUnicode 8비트 문자 코드 값의 배열을 16비트 유니코드 값으로 변환합니다.
TTDeleteEmbeddedFont 포함된 글꼴 hFontReference에서 사용하는 메모리를 해제합니다.
TTEmbedFont 하위 집합 와이드 문자(16비트) 글꼴을 포함하는 글꼴 구조를 만듭니다. 디바이스 컨텍스트(hDC)의 현재 글꼴은 글꼴 정보를 제공합니다.
TTEmbedFontEx 하위 집합 UCS-4 문자(32비트) 글꼴을 포함하는 글꼴 구조를 만듭니다. 디바이스 컨텍스트(hDC)의 현재 글꼴은 글꼴 정보를 제공합니다.
TTEmbedFontFromFileA 하위 집합 와이드 문자(16비트) 글꼴을 포함하는 글꼴 구조를 만듭니다. 외부 파일은 글꼴 정보를 제공합니다.
TTEnableEmbeddingForFacename 서체 제외 목록에서 얼굴 이름을 추가하거나 제거합니다.
TTGetEmbeddedFontInfo 포함 권한과 같은 포함된 글꼴에 대한 정보를 검색합니다. TTGetEmbeddedFontInfo는 TTLoadEmbeddedFont와 동일한 작업을 수행하지만 포함된 글꼴에 대한 내부 데이터 구조를 할당하지는 않습니다.

TTGetEmbeddingType 글꼴의 포함 권한을 가져옵니다.
TTGetNewFontName TTLoadEmbeddedFont를 통해 로드된 글꼴의 패밀리 이름을 가져옵니다.
TTIsEmbeddingEnabled 서체 제외 목록에 지정된 글꼴이 포함되어 있는지 여부를 확인합니다.
TTIsEmbeddingEnabledForFacename 지정된 글꼴에 포함을 사용할 수 있는지 여부를 확인합니다.
TTLoadEmbeddedFont 문서 스트림에서 포함된 글꼴을 읽고 설치합니다. 또한 클라이언트가 글꼴의 포함 권한을 추가로 제한할 수 있습니다.
TTRunValidationTests 지정된 크기 범위에서 와이드 문자(16비트) 글꼴의 일부 또는 모든 문자 모양 데이터의 유효성을 검사합니다.
TTRunValidationTestsEx 지정된 크기 범위에서 UCS-4 문자(32비트) 글꼴의 일부 또는 모든 문자 모양 데이터의 유효성을 검사합니다.

구조체

TTEMBEDINFO TTEMBEDINFO 구조체에는 포함된 글꼴 개체가 합법적으로 참조될 수 있는 URL 목록이 포함되어 있습니다.
TTLOADINFO TTLOADINFO 구조체에는 포함된 글꼴 개체를 가져온 URL이 포함됩니다.

TTVALIDATIONTESTSPARAMS TTVALIDATIONTESTSPARAMS 구조체에는 Microsoft OpenType 글꼴을 테스트하기 위한 매개 변수가 포함되어 있습니다.
TTVALIDATIONTESTSPARAMSEX TTVALIDATIONTESTSPARAMSEX 구조체에는 Microsoft OpenType 글꼴을 테스트하기 위한 매개 변수가 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

tabflicks.h 헤더

아티클2023. 08. 26.

이 헤더는 태블릿 PC에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Tablet PC

tabflicks.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

FLICK_DATA 펜 플릭에 대한 정보를 포함합니다.
FLICK_POINT 펜 터치에 대한 정보를 제공합니다.

열거형

FLICKACTION_COMMANDCODE 펜 플릭에 할당할 수 있는 가능한 영화 동작을 정의합니다.
FLICKDIRECTION 펜 플릭이 발생한 방향을 정의합니다.
FLICKMODE Flick 제스처가 사용되는 위치를 설명합니다.
KEYMODIFIER 플릭 제스처가 발생했을 때 누른 한정자 키를 결정합니다.
SCROLLDIRECTION 펜 플릭에 대한 스크롤 명령의 방향을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

tapi.h 헤더

아티클2023. 03. 08.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- TAPI 2.2
- 전화 통신 애플리케이션 프로그래밍 인터페이스
- 전화 통신 서비스 공급자

tapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

lineAccept lineAccept 함수는 지정된 제공된 호출을 수락합니다. 필요에 따라 지정된 사용자-사용자 정보를 통화 당사자에게 보낼 수 있습니다.
lineAddProvider lineAddProvider 함수(tapi.h)는 새 전화 통신 서비스 공급자를 전화 통신 시스템에 설치합니다.
lineAddProviderA lineAddProvider 함수는 전화 통신 시스템에 새 전화 통신 서비스 공급자를 설치합니다. (lineAddProviderA)
lineAddProviderW lineAddProviderW(유니코드) 함수(tapi.h)는 전화 통신 시스템에 새 전화 통신 서비스 공급자를 설치합니다.
lineAddToConference lineAddToConference 함수는 hConsultCall로 지정된 호출을 hConfCall로 지정된 전화 회의에 추가합니다.
lineAgentSpecific lineAgentSpecific 함수를 사용하면 애플리케이션이 주소와 연결된 에이전트 처리기의 전용 처리기 관련 함수에 액세스할 수 있습니다.

lineAnswer lineAnswer 함수는 지정된 제품 호출에 응답합니다.
lineBlindTransfer lineBlindTransfer 함수(tapi.h)는 지정된 대상 주소에 대한 지정된 호출의 블라인드 또는 단일 단계 전송을 수행합니다.
lineBlindTransferA lineBlindTransfer 함수는 지정된 대상 주소에 대한 지정된 호출의 블라인드 또는 단일 단계 전송을 수행합니다. (lineBlindTransferA)
lineBlindTransferW lineBlindTransfer(유니코드) 함수(tapi.h)는 지정된 대상 주소에 대한 지정된 호출의 블라인드 또는 단일 단계 전송을 수행합니다.
lineClose lineClose 함수는 지정된 열린 선 디바이스를 닫습니다.
lineCompleteCall lineCompleteCall 함수는 정상적으로 연결할 수 없는 호출을 대신 완료하는 방법을 지정합니다.
lineCompleteTransfer lineCompleteTransfer 함수는 상담 통화에 연결된 당사자에게 지정된 호출의 전송을 완료합니다.
lineConfigDialog lineConfigDialog 함수(tapi.h)를 사용하면 지정된 줄 디바이스의 공급자가 대화 상자를 표시합니다.
lineConfigDialogA lineConfigDialog 함수는 지정된 줄 디바이스의 공급자가 사용자가 줄 디바이스와 관련된 매개 변수를 구성할 수 있도록 대화 상자(애플리케이션의 hwndOwner에 연결됨)를 표시하도록 합니다. (lineConfigDialogA)
lineConfigDialogEdit lineConfigDialogEdit 함수(tapi.h)를 사용하면 지정된 줄 디바이스의 공급자가 대화 상자를 표시합니다.

lineConfigDialogEditA lineConfigDialogEdit 함수를 사용하면 지정된 줄 디바이스의 공급자가 사용자가 줄 디바이스와 관련된 매개 변수를 구성할 수 있도록 대화 상자(애플리케이션의 hwndOwner에 연결됨)를 표시합니다. (lineConfigDialogEditA)
lineConfigDialogEditW lineConfigDialogEditW(유니코드) 함수(tapi.h)를 사용하면 지정된 줄 디바이스의 공급자가 대화 상자를 표시합니다.
lineConfigDialogW lineConfigDialogW(유니코드) 함수(tapi.h)를 사용하면 지정된 줄 디바이스의 공급자가 대화 상자를 표시합니다.
lineConfigProvider lineConfigProvider 함수를 사용하면 서비스 공급자가 구성 대화 상자를 표시합니다.
lineCreateAgentA lineCreateAgent 함수는 새 에이전트 개체를 만듭니다. LINEPROXYREQUEST_CREATEAGENT 형식의 LINEPROXYREQUEST 구조를 참조하여 등록된 프록시 함수 처리기로 보낼 LINE_PROXYREQUEST 메시지를 생성합니다. (ANSI)
lineCreateAgentSessionA lineCreateAgentSession 함수는 새 AgentSession 개체를 만듭니다. LINEPROXYREQUEST_CREATEAGENTSESSION 형식의 LINEPROXYREQUEST 구조를 참조하여 등록된 프록시 함수 처리기로 보낼 LINE_PROXYREQUEST 메시지를 생성합니다. (ANSI)
lineCreateAgentSessionW lineCreateAgentSession 함수는 새 AgentSession 개체를 만듭니다. LINEPROXYREQUEST_CREATEAGENTSESSION 형식의 LINEPROXYREQUEST 구조를 참조하여 등록된 프록시 함수 처리기로 보낼 LINE_PROXYREQUEST 메시지를 생성합니다. (유니코드)
lineCreateAgentW lineCreateAgent 함수는 새 에이전트 개체를 만듭니다. LINEPROXYREQUEST_CREATEAGENT 형식의 LINEPROXYREQUEST 구조를 참조하여 등록된 프록시 함수 처리기로 보낼 LINE_PROXYREQUEST 메시지를 생성합니다. (유니코드)
lineDeallocateCall 지정된 호출 핸들의 할당을 취소합니다.

lineDevSpecific lineDevSpecific 함수를 사용하면 서비스 공급자가 다른 TAPI 함수에서 제공하지 않는 기능에 액세스할 수 있습니다.
lineDevSpecificFeature lineDevSpecificFeature 함수를 사용하면 서비스 공급자가 다른 TAPI 함수에서 제공하지 않는 기능에 액세스할 수 있습니다.
lineDial lineDial 함수(tapi.h)는 지정된 통화에서 지정된 전화 걸기 가능 번호로 전화를 켭니다.
lineDialA lineDial 함수는 지정된 통화에서 지정된 전화 걸기 가능 번호로 전화를 켭니다. (lineDialA)
lineDialW lineDialW(유니코드) 함수(tapi.h)는 지정된 통화에서 지정된 전화 걸기 가능 번호로 전화를 켭니다.
lineDrop lineDrop 함수는 지정된 호출을 삭제하거나 연결을 끊습니다. 애플리케이션에는 통화 연결 끊기의 일부로 전송할 사용자-사용자 정보를 지정하는 옵션이 있습니다.
lineForward lineForward 함수(tapi.h)는 지정된 전달 지침에 따라 지정된 줄의 지정된 주소로 향하는 호출을 전달합니다.
lineForwardA lineForward 함수는 지정된 전달 지침에 따라 지정된 줄에서 지정된 주소로 향하는 호출을 전달합니다. (lineForwardA)
lineForwardW lineForwardW(유니코드) 함수는 지정된 전달 지침에 따라 지정된 줄에서 지정된 주소로 향하는 호출을 전달합니다.
lineGatherDigits lineGatherDigits 함수(tapi.h)는 지정된 호출에서 버퍼링된 숫자 수집을 시작합니다.

lineGatherDigitsA lineGatherDigits 함수는 지정된 호출에서 버퍼링된 숫자 수집을 시작합니다. 애플리케이션은 수집할 숫자와 최대 자릿수를 배치할 버퍼를 지정합니다. (lineGatherDigitsA)
lineGatherDigitsW lineGatherDigitsW(유니코드) 함수는 지정된 호출에서 버퍼링된 숫자 수집을 시작합니다.
lineGenerateDigits lineGenerateDigits 함수(tapi.h)는 지정된 신호 모드를 사용하여 지정된 호출에서 지정된 숫자의 생성을 인밴드 톤으로 시작합니다.
lineGenerateDigitsA 지정된 신호 모드를 사용하여 지정된 호출에서 지정된 숫자를 인밴드 톤으로 생성합니다. (lineGenerateDigitsA)
lineGenerateDigitsW lineGenerateDigitsW(유니코드) 함수는 지정된 신호 모드를 사용하여 호출 시 지정된 숫자를 인밴드 톤으로 생성합니다.
lineGenerateTone lineGenerateTone 함수는 지정된 호출에 대해 지정된 인밴드 톤을 생성합니다.
lineGetAddressCaps lineGetAddressCaps 함수(tapi.h)는 지정된 줄 디바이스에서 지정된 주소를 쿼리하여 전화 통신 기능을 확인합니다.
lineGetAddressCapsA lineGetAddressCaps 함수는 지정된 줄 디바이스에서 지정된 주소를 쿼리하여 전화 통신 기능을 확인합니다. (lineGetAddressCapsA)
lineGetAddressCapsW lineGetAddressCapsW(유니코드) 함수(tapi.h)는 지정된 줄 디바이스에서 지정된 주소를 쿼리하여 전화 통신 기능을 확인합니다.
lineGetAddressID lineGetAddressID 함수(tapi.h)는 지정된 줄에서 주소와 연결된 주소 식별자를 다른 형식으로 반환합니다.

lineGetAddressIDA lineGetAddressID 함수는 지정된 줄에서 주소와 연결된 주소 식별자를 다른 형식으로 반환합니다. (lineGetAddressIDA)
lineGetAddressIDW lineGetAddressIDW(유니코드) 함수(tapi.h)는 지정된 줄에서 주소와 연결된 주소 식별자를 다른 형식으로 반환합니다.
lineGetAddressStatus lineGetAddressStatus 함수(tapi.h)를 사용하면 애플리케이션이 현재 상태에 대해 지정된 주소를 쿼리할 수 있습니다.
lineGetAddressStatusA lineGetAddressStatus 함수를 사용하면 애플리케이션이 지정된 주소의 현재 상태를 쿼리할 수 있습니다. (lineGetAddressStatusA)
lineGetAddressStatusW lineGetAddressStatusW 함수(tapi.h)를 사용하면 애플리케이션이 현재 상태에 대해 지정된 주소를 쿼리할 수 있습니다.
lineGetAgentActivityListA lineGetAgentActivityList 함수는 lineSetAgentActivity를 사용하여 현재 에이전트가 실제로 수행하는 함수를 나타내기 위해 애플리케이션에서 선택할 수 있는 활동의 ID를 가져옵니다. (ANSI)
lineGetAgentActivityListW lineGetAgentActivityList 함수는 lineSetAgentActivity를 사용하여 현재 에이전트가 실제로 수행하는 함수를 나타내기 위해 애플리케이션에서 선택할 수 있는 활동의 ID를 가져옵니다. (유니코드)
lineGetAgentCapsA lineGetAgentCaps 함수는 지정된 줄 디바이스에서 지원되는 에이전트 관련 기능을 가져옵니다. 특정 에이전트의 이름이 지정된 경우 기능에는 에이전트가 로그인할 수 있는 ACD 그룹 목록이 포함됩니다. (ANSI)
lineGetAgentCapsW lineGetAgentCaps 함수는 지정된 줄 디바이스에서 지원되는 에이전트 관련 기능을 가져옵니다. 특정 에이전트의 이름이 지정된 경우 기능에는 에이전트가 로그인할 수 있는 ACD 그룹 목록이 포함됩니다. (유니코드)

lineGetAgentGroupListA lineGetAgentGroupList 함수는 현재 워크스테이션에 로그인한 에이전트가 자동 호출 배포자에 로그인할 수 있는 에이전트 그룹(큐, 감독자, 기술 수준 등의 조합)의 ID를 가져옵니다. (ANSI)
lineGetAgentGroupListW lineGetAgentGroupList 함수는 현재 워크스테이션에 로그인한 에이전트가 자동 호출 배포자에 로그인할 수 있는 에이전트 그룹(큐, 감독자, 기술 수준 등의 조합)의 ID를 가져옵니다. (유니코드)
lineGetAgentInfo lineGetAgentInfo 함수는 특정 에이전트 핸들과 연결된 ACD 정보를 포함하는 구조를 반환합니다.
lineGetAgentSessionInfo lineGetAgentSessionInfo 함수는 특정 에이전트 세션 핸들과 연결된 ACD 정보를 포함하는 구조를 반환합니다.
lineGetAgentSessionList lineGetAgentSessionList 함수는 지정된 에이전트에 대해 생성된 에이전트 세션 목록을 반환합니다.
lineGetAgentStatusA lineGetAgentStatus 함수는 지정된 주소에서 에이전트 관련 상태를 가져옵니다. (ANSI)
lineGetAgentStatusW lineGetAgentStatus 함수는 지정된 주소에서 에이전트 관련 상태를 가져옵니다. (유니코드)
lineGetAppPriority lineGetAppPriority 함수(tapi.h)를 사용하면 애플리케이션이 특정 미디어 모드의 핸드오프 우선 순위 목록에 있는지 여부를 확인할 수 있습니다.
lineGetAppPriorityA 애플리케이션이 특정 미디어 모드 또는 보조 전화 통신 요청 모드의 핸드오프 우선 순위 목록에 있는지 여부와 우선 순위 목록에서 해당 위치를 확인할 수 있도록 합니다. (lineGetAppPriorityA)
lineGetAppPriorityW lineGetAppPriorityW(유니코드) 함수를 사용하면 애플리케이션이 특정 미디어 모드의 핸드오프 우선 순위 목록에 있는지 여부를 확인할 수 있습니다.

lineGetCallInfo lineGetCallInfo 함수(tapi.h)를 사용하면 애플리케이션이 지정된 호출에 대한 고정 정보를 가져올 수 있습니다.
lineGetCallInfoA lineGetCallInfo 함수를 사용하면 애플리케이션이 지정된 호출에 대한 고정 정보를 가져올 수 있습니다. (lineGetCallInfoA)
lineGetCallInfoW lineGetCallInfoW(유니코드) 함수(tapi.h)를 사용하면 애플리케이션이 지정된 호출에 대한 고정 정보를 가져올 수 있습니다.
lineGetCallStatus lineGetCallStatus 함수는 지정된 호출의 현재 상태를 반환합니다.
lineGetConfRelatedCalls lineGetConfRelatedCalls 함수는 지정된 호출과 동일한 전화 회의 호출의 일부인 호출 핸들 목록을 반환합니다.
lineGetCountry lineGetCountry 함수(tapi.h)는 저장된 전화 걸기 규칙 및 지정된 국가/지역과 관련된 기타 정보를 가져옵니다.
lineGetCountryA lineGetCountry 함수는 저장된 전화 걸기 규칙 및 지정된 국가/지역, 국가/지역 목록의 첫 번째 국가/지역 또는 모든 국가/지역과 관련된 기타 정보를 가져옵니다. (lineGetCountryA)
lineGetCountryW lineGetCountryW(유니코드) 함수는 저장된 전화 걸기 규칙 및 지정된 국가/지역과 관련된 기타 정보를 가져옵니다.
lineGetDevCaps lineGetDevCaps 함수(tapi.h)는 지정된 줄 디바이스를 쿼리하여 전화 통신 기능을 확인합니다.
lineGetDevCapsA lineGetDevCaps 함수는 지정된 줄 디바이스를 쿼리하여 전화 통신 기능을 확인합니다. 반환된 정보는 회선 디바이스의 모든 주소에 유효합니다. (lineGetDevCapsA)

lineGetDevCapsW lineGetDevCapsW(유니코드) 함수(tapi.h)는 지정된 줄 디바이스를 쿼리하여 전화 통신 기능을 확인합니다.
lineGetDevConfig lineGetDevConfig 함수(tapi.h)는 줄 및 디바이스 클래스와 관련된 내용인 "불투명" 데이터 구조 개체를 반환합니다.
lineGetDevConfigA lineGetDevConfig 함수는 줄(서비스 공급자) 및 디바이스 클래스와 관련된 "불투명" 데이터 구조 개체를 반환합니다. (lineGetDevConfigA)
lineGetDevConfigW lineGetDevConfigW(유니코드) 함수는 줄 및 디바이스 클래스와 관련된 내용인 "불투명" 데이터 구조 개체를 반환합니다.
lineGetGroupListA lineGetGroupList 함수는 ACD 시스템에서 사용할 수 있는 ACD 그룹 목록을 반환합니다. (ANSI)
lineGetGroupListW lineGetGroupList 함수는 ACD 시스템에서 사용할 수 있는 ACD 그룹 목록을 반환합니다. (유니코드)
lineGetIcon lineGetIcon 함수(tapi.h)를 사용하면 애플리케이션이 사용자에게 표시할 서비스 라인 디바이스별(또는 공급자별) 아이콘을 검색할 수 있습니다.
lineGetIconA lineGetIcon 함수를 사용하면 애플리케이션이 사용자에게 표시할 서비스 라인 디바이스별(또는 공급자별) 아이콘을 검색할 수 있습니다. (lineGetIconA)
lineGetIconW lineGetIconW(유니코드) 함수를 사용하면 애플리케이션이 사용자에게 표시할 서비스 라인 디바이스별 아이콘을 검색할 수 있습니다.
lineGetID lineGetID 함수(tapi.h)는 선택한 줄, 주소 또는 호출과 연결된 지정된 디바이스 클래스에 대한 디바이스 식별자를 반환합니다.

lineGetIDA lineGetID 함수는 선택한 줄, 주소 또는 호출과 연결된 지정된 디바이스 클래스에 대한 디바이스 식별자를 반환합니다. (lineGetIDA)
lineGetIDW lineGetIDW(유니코드) 함수(tapi.h)는 선택한 줄, 주소 또는 호출과 연결된 지정된 디바이스 클래스에 대한 디바이스 식별자를 반환합니다.
lineGetLineDevStatus lineGetLineDevStatus 함수(tapi.h)를 사용하면 애플리케이션이 지정된 열린 줄 디바이스의 현재 상태를 쿼리할 수 있습니다.
lineGetLineDevStatusA lineGetLineDevStatus 함수를 사용하면 애플리케이션이 지정된 열린 줄 디바이스의 현재 상태를 쿼리할 수 있습니다. (lineGetLineDevStatusA)
lineGetLineDevStatusW lineGetLineDevStatusW(유니코드) 함수를 사용하면 애플리케이션이 지정된 열린 줄 디바이스의 현재 상태를 쿼리할 수 있습니다.
lineGetMessage lineGetMessage 함수는 이벤트 핸들 알림 메커니즘을 사용하는 애플리케이션에 배달 대기 중인 다음 TAPI 메시지를 반환합니다(자세한 내용은 lineInitializeEx 참조).
lineGetNewCalls lineGetNewCalls 함수는 애플리케이션에 현재 핸들이 없는 지정된 줄 또는 주소의 호출에 대한 호출 핸들을 반환합니다. 애플리케이션에는 이러한 호출에 대한 모니터 권한이 부여됩니다.
lineGetNumRings lineGetNumRings 함수는 호출에 응답하기 전에 지정된 주소에서 들어오는 호출이 울리는 링 수를 결정합니다.
lineGetProviderList lineGetProviderList 함수(tapi.h)는 현재 전화 통신 시스템에 설치된 서비스 공급자 목록을 반환합니다.
lineGetProviderListA lineGetProviderList 함수는 현재 전화 통신 시스템에 설치된 서비스 공급자 목록을 반환합니다. (lineGetProviderListA)

lineGetProviderListW lineGetProviderListW(유니코드) 함수(tapi.h)는 현재 전화 통신 시스템에 설치된 서비스 공급자 목록을 반환합니다.
lineGetProxyStatus lineGetProxyStatus 함수는 지정된 디바이스에 대해 현재 서비스되고 있는 프록시 요청 형식 목록을 반환합니다.
lineGetQueueInfo lineGetQueueInfo 함수는 특정 큐와 연결된 ACD 정보를 포함하는 구조를 반환합니다.
lineGetQueueListA lineGetQueueList 함수는 특정 ACD 그룹과 연결된 큐 목록을 반환합니다. (ANSI)
lineGetQueueListW lineGetQueueList 함수는 특정 ACD 그룹과 연결된 큐 목록을 반환합니다. (유니코드)
lineGetRequest lineGetRequest 함수(tapi.h)는 지정된 요청 모드에 대한 다음 프록시별 요청을 검색합니다.
lineGetRequestA 지정된 요청 모드에 대한 다음 프록시별 요청을 검색합니다. (lineGetRequestA)
lineGetRequestW lineGetRequestW(유니코드) 함수(tapi.h)는 지정된 요청 모드에 대한 다음 프록시별 요청을 검색합니다.
lineGetStatusMessages lineGetStatusMessages 함수를 사용하면 애플리케이션이 지정된 줄 또는 주소의 상태 변경과 관련된 이벤트에 대해 수신하도록 설정된 알림 메시지를 쿼리할 수 있습니다.
lineGetTranslateCaps lineGetTranslateCaps 함수(tapi.h)는 주소 변환 기능을 반환합니다.
lineGetTranslateCapsA lineGetTranslateCaps 함수는 주소 변환 기능을 반환합니다. (lineGetTranslateCapsA)

lineGetTranslateCapsW lineGetTranslateCapsW(유니코드) 함수(tapi.h)는 주소 변환 기능을 반환합니다.
lineHandoff lineHandoff 함수(tapi.h)는 다른 애플리케이션에 대해 지정된 호출의 소유권을 부여합니다.
lineHandoffA lineHandoff 함수는 다른 애플리케이션에 대해 지정된 호출의 소유권을 부여합니다. 애플리케이션은 파일 이름으로 직접 지정하거나 지정된 미디어 모드의 호출을 처리하는 우선 순위가 가장 높은 애플리케이션으로 간접적으로 지정할 수 있습니다. (lineHandoffA)
lineHandoffW lineHandoffW(유니코드) 함수(tapi.h)는 다른 애플리케이션에 대해 지정된 호출의 소유권을 제공합니다.
lineHold lineHold 함수는 지정된 호출을 보류합니다.
lineInitialize lineInitialize 함수는 사용되지 않습니다. API 버전 1.3 및 1.4를 사용하는 애플리케이션과의 이전 버전과의 호환성을 위해 Tapi.dll 계속 내보내고 Tapi32.dll.
lineInitializeExA lineInitializeEx 함수는 줄 추상화의 후속 사용을 위해 애플리케이션의 TAPI 사용을 초기화합니다. (ANSI)
lineInitializeExW lineInitializeEx 함수는 줄 추상화의 후속 사용을 위해 애플리케이션의 TAPI 사용을 초기화합니다. (유니코드)
lineMakeCall lineMakeCall 함수(tapi.h)는 지정된 줄에 지정된 대상 주소에 대한 호출을 배치합니다.
lineMakeCallA lineMakeCall 함수는 지정된 줄에서 지정된 대상 주소에 대한 호출을 배치합니다. 필요에 따라 기본 호출 설정 매개 변수를 제외한 모든 호출 매개 변수를 요청할 경우 호출 매개 변수를 지정할 수 있습니다. (lineMakeCallA)

lineMakeCallW lineMakeCallW(유니코드) 함수(tapi.h)는 지정된 줄에서 지정된 대상 주소에 대한 호출을 배치합니다.
lineMonitorDigits lineMonitorDigits 함수는 호출 시 수신된 숫자의 버퍼되지 않은 검색을 사용하도록 설정하고 사용하지 않도록 설정합니다. 지정된 숫자 모드의 숫자가 검색될 때마다 검색된 숫자를 나타내는 메시지가 애플리케이션에 전송됩니다.
lineMonitorMedia lineMonitorMedia 함수는 지정된 호출에서 미디어 유형(모드)의 검색을 활성화하고 사용하지 않도록 설정합니다. 미디어 유형이 검색되면 애플리케이션에 메시지가 전송됩니다. 자세한 내용은 ITLegacyCallMediaControl::MonitorMedia를 참조하세요.
lineMonitorTones lineMonitorTones 함수는 호출 시 인밴드 톤 검색을 활성화하고 사용하지 않도록 설정합니다. 지정된 톤이 검색될 때마다 애플리케이션에 메시지가 전송됩니다.
lineNegotiateAPIVersion lineNegotiateAPIVersion 함수를 사용하면 애플리케이션에서 사용할 API 버전을 협상할 수 있습니다.
lineNegotiateExtVersion lineNegotiateExtVersion 함수를 사용하면 애플리케이션이 지정된 줄 디바이스에서 사용할 확장 버전을 협상할 수 있습니다. 애플리케이션이 확장을 지원하지 않는 경우 이 작업을 호출할 필요가 없습니다.
lineOpen lineOpen 함수(tapi.h)는 디바이스 식별자가 지정한 선 디바이스를 열고 해당 열린 선 디바이스에 대한 줄 핸들을 반환합니다.
lineOpenA lineOpen 함수는 디바이스 식별자가 지정한 줄 디바이스를 열고 열린 해당 줄 디바이스에 대한 줄 핸들을 반환합니다. 이 줄 핸들은 라인 디바이스의 후속 작업에 사용됩니다. (lineOpenA)
lineOpenW lineOpenW(유니코드) 함수(tapi.h)는 디바이스 식별자에 지정된 선 디바이스를 열고 해당 열린 선 디바이스에 대한 줄 핸들을 반환합니다.

linePark linePark 함수(tapi.h)는 지정된 파크 모드에 따라 지정된 호출을 파킹합니다.
lineParkA linePark 함수는 지정된 공원 모드에 따라 지정된 호출을 주차합니다. (lineParkA)
lineParkW lineParkW(유니코드) 함수(tapi.h)는 지정된 파크 모드에 따라 지정된 호출을 파킹합니다.
linePickup linePickup 함수(tapi.h)는 지정된 대상 주소에서 호출 경고를 선택하고 선택한 호출에 대한 호출 핸들을 반환합니다.
linePickupA linePickup 함수는 지정된 대상 주소에서 호출 경고를 선택하고 선택한 호출에 대한 호출 핸들을 반환합니다. (linePickupA)
linePickupW linePickupW(유니코드) 함수(tapi.h)는 지정된 대상 주소에서 호출 경고를 선택하고 픽업된 호출에 대한 호출 핸들을 반환합니다.
linePrepareAddToConference linePrepareAddToConference 함수(tapi.h)는 다른 파티의 추가를 위해 기존 전화 회의를 준비합니다.
linePrepareAddToConferenceA linePrepareAddToConference 함수는 다른 파티의 추가를 위해 기존 전화 회의를 준비합니다. (linePrepareAddToConferenceA)
linePrepareAddToConferenceW linePrepareAddToConferenceW(유니코드) 함수(tapi.h)는 다른 파티의 추가를 위해 기존 전화 회의를 준비합니다.
lineProxyMessage lineProxyMessage 함수는 등록된 프록시 요청 처리기에서 해당 역할과 관련된 TAPI 메시지를 생성하는 데 사용됩니다.

lineProxyResponse 서버의 ACD 에이전트 처리기와 같이 등록된 프록시 처리기의 프록시 요청 완료를 나타냅니다.
lineRedirect lineRedirect 함수(tapi.h)는 지정된 제품 호출을 지정된 대상 주소로 리디렉션합니다.
lineRedirectA lineRedirect 함수는 지정된 제품 호출을 지정된 대상 주소로 리디렉션합니다. (lineRedirectA)
lineRedirectW lineRedirectW(유니코드) 함수(tapi.h)는 지정된 제품 호출을 지정된 대상 주소로 리디렉션합니다.
lineRegisterRequestRecipient lineRegisterRequestRecipient 함수는 호출하는 애플리케이션을 지정된 요청 모드에 대한 요청의 수신자로 등록합니다.
lineReleaseUserUserInfo lineReleaseUserUserInfo 함수는 애플리케이션이 LINECALLINFO 구조에 포함된 사용자-사용자 정보를 처리했음을 서비스 공급자에게 알릴 수 있습니다.
lineRemoveFromConference lineRemoveFromConference 함수는 현재 속한 전화 회의 호출에서 지정된 호출을 제거합니다. 전화 회의의 나머지 통화는 영향을 받지 않습니다.
lineRemoveProvider lineRemoveProvider 함수는 전화 통신 시스템에서 기존 전화 통신 서비스 공급자를 제거합니다.
lineSecureCall lineSecureCall 함수는 호출의 미디어 스트림에 영향을 줄 수 있는 중단 또는 간섭으로부터 호출을 보호합니다.
lineSendUserUserInfo lineSendUserUserInfo 함수는 지정된 호출에서 원격 당사자에게 사용자-사용자 정보를 보냅니다.
lineSetAgentActivity lineSetAgentActivity 함수는 특정 주소와 연결된 에이전트 활동 코드를 설정합니다.

lineSetAgentGroup lineSetAgentGroup 함수는 에이전트가 특정 주소에 로그인되는 에이전트 그룹을 설정합니다.
lineSetAgentMeasurementPeriod lineSetAgentMeasurementPeriod 함수는 특정 에이전트와 연결된 측정 기간을 설정합니다.
lineSetAgentSessionState lineSetAgentSessionState 함수는 특정 에이전트 세션 핸들과 연결된 에이전트 세션 상태를 설정합니다.
lineSetAgentState lineSetAgentState 함수는 특정 주소와 연결된 에이전트 상태를 설정합니다.
lineSetAgentStateEx lineSetAgentStateEx 함수는 특정 에이전트 핸들과 연결된 에이전트 상태를 설정합니다.
lineSetAppPriority lineSetAppPriority 함수(tapi.h)를 사용하면 애플리케이션이 핸드오프 우선 순위 목록에서 우선 순위를 설정할 수 있습니다.
lineSetAppPriorityA 애플리케이션이 특정 미디어 유형 또는 보조 전화 통신 요청 모드의 핸드오프 우선 순위 목록에서 우선 순위를 설정하거나 우선 순위 목록에서 자신을 제거할 수 있도록 합니다. (lineSetAppPriorityA)
lineSetAppPriorityW lineSetAppPriorityW(유니코드) 함수(tapi.h)를 사용하면 애플리케이션이 핸드오프 우선 순위 목록에서 우선 순위를 설정할 수 있습니다.
lineSetAppSpecific lineSetAppSpecific 함수를 사용하면 애플리케이션이 지정된 호출의 호출 정보 레코드의 애플리케이션별 필드를 설정할 수 있습니다.
lineSetCallData lineSetCallData 함수는 LINECALLINFO에서 CallData 멤버를 설정합니다.

lineSetCallParams lineSetCallParams 함수를 사용하면 애플리케이션이 전달자 모드 및/또는 기존 호출의 속도 매개 변수를 변경할 수 있습니다.
lineSetCallPrivilege lineSetCallPrivilege 함수는 애플리케이션의 권한을 지정된 권한으로 설정합니다.
lineSetCallQualityOfService lineSetCallQualityOfService 함수를 사용하면 애플리케이션이 기존 호출에 대한 서비스 매개 변수의 품질(예약된 용량 및 성능 보장)을 변경하려고 시도할 수 있습니다.
lineSetCallTreatment lineSetCallTreatment 함수는 응답이 없거나 대기 중인 통화에서 파티 소리를 설정합니다. 기본 매개 변수 유효성 검사를 제외하고 서비스 공급자에 대한 TAPI의 직선 통과입니다.
lineSetCurrentLocation lineSetCurrentLocation 함수는 주소 변환의 컨텍스트로 사용되는 위치를 설정합니다.
lineSetDevConfig lineSetDevConfig 함수(tapi.h)를 사용하면 애플리케이션이 라인 디바이스에서 미디어 스트림 디바이스의 구성을 이전 설정으로 복원할 수 있습니다.
lineSetDevConfigA lineSetDevConfig 함수를 사용하면 애플리케이션이 라인 디바이스의 미디어 스트림 디바이스 구성을 lineGetDevConfig를 사용하여 이전에 가져온 설정으로 복원할 수 있습니다. (lineSetDevConfigA)
lineSetDevConfigW lineSetDevConfigW(유니코드) 함수(tapi.h)를 사용하면 애플리케이션이 라인 디바이스에서 미디어 스트림 디바이스의 구성을 이전 설정으로 복원할 수 있습니다.
lineSetLineDevStatus lineSetLineDevStatus 함수는 줄 디바이스 상태를 설정합니다.
lineSetMediaControl lineSetMediaControl 함수는 지정된 줄, 주소 또는 호출과 연결된 미디어 스트림에서 제어 작업을 사용하거나 사용하지 않도록 설정합니다.

lineSetMediaMode lineSetMediaMode 함수는 LINECALLINFO 구조에서 지정된 호출의 미디어 형식을 설정합니다. 자세한 내용은 ITLegacyCallMediaControl::SetMediaType을 참조하세요.
lineSetNumRings lineSetNumRings 함수는 들어오는 호출에 응답하기 전에 발생해야 하는 링 수를 설정합니다.
lineSetQueueMeasurementPeriod lineSetQueueMeasurementPeriod 함수는 특정 큐와 연결된 측정 기간을 설정합니다.
lineSetStatusMessages lineSetStatusMessages 함수를 사용하면 애플리케이션이 지정된 줄 또는 해당 주소의 상태 변경과 관련된 이벤트에 대해 수신할 알림 메시지를 지정할 수 있습니다.
lineSetTerminal lineSetTerminal 함수를 사용하면 애플리케이션이 라우팅할 지정된 줄, 주소 또는 호출과 관련된 터미널 정보를 지정할 수 있습니다.
lineSetTollList lineSetTollList 함수(tapi.h)는 통행료 목록을 조작합니다.
lineSetTollListA lineSetTollList 함수는 통행료 목록을 조작합니다. (lineSetTollListA)
lineSetTollListW lineSetTollListW(유니코드) 함수(tapi.h)는 통행료 목록을 조작합니다.
lineSetupConference lineSetupConference 함수(tapi.h)는 타사 추가를 위한 전화 회의를 설정합니다.
lineSetupConferenceA lineSetupConference 함수는 타사 추가를 위한 전화 회의를 설정합니다. (lineSetupConferenceA)
lineSetupConferenceW lineSetupConferenceW(유니코드) 함수(tapi.h)는 타사 추가를 위한 전화 회의를 설정합니다.

lineSetupTransfer lineSetupTransfer 함수(tapi.h)는 hCall 매개 변수로 지정된 호출의 전송을 시작합니다.
lineSetupTransferA lineSetupTransfer 함수는 hCall 매개 변수로 지정된 호출의 전송을 시작합니다. (lineSetupTransferA)
lineSetupTransferW lineSetupTransferW(유니코드) 함수(tapi.h)는 hCall 매개 변수로 지정된 호출의 전송을 시작합니다.
lineShutdown lineShutdown 함수는 API의 선 추상화에 대한 애플리케이션의 사용을 종료합니다.
lineSwapHold lineSwapHold 함수는 지정된 활성 호출을 상담 보류의 지정된 호출로 바꿉니다.
lineTranslateAddress lineTranslateAddress 함수(tapi.h)는 지정된 주소를 다른 형식으로 변환합니다.
lineTranslateAddressA lineTranslateAddress 함수는 지정된 주소를 다른 형식으로 변환합니다. (lineTranslateAddressA)
lineTranslateAddressW lineTranslateAddressW(유니코드) 함수(tapi.h)는 지정된 주소를 다른 형식으로 변환합니다.
lineTranslateDialog lineTranslateDialog 함수(tapi.h)는 사용자가 전화 번호 위치를 변경하고, 통화 카드 매개 변수를 조정하고, 효과를 볼 수 있도록 하는 애플리케이션 모달 대화 상자를 표시합니다.
lineTranslateDialogA lineTranslateDialog 함수는 사용자가 전화를 걸 전화 번호의 현재 위치를 변경하고, 위치 및 통화 카드 매개 변수를 조정하고, 효과를 볼 수 있도록 하는 애플리케이션 모달 대화 상자를 표시합니다. (lineTranslateDialogA)
lineTranslateDialogW lineTranslateDialogW(유니코드) 함수(tapi.h)는 사용자가 전화 번호 위치를 변경하고, 통화 카드 매개 변수를 조정하고, 효과를 볼 수 있는 애플리케이션 모달 대화 상자를 표시합니다.

lineUncompleteCall lineUncompleteCall 함수는 지정된 줄에서 지정된 호출 완료 요청을 취소합니다.
lineUnhold lineUnhold 함수는 지정된 보류된 호출을 검색합니다.
lineUnpark lineUnpark 함수(tapi.h)는 지정된 주소에 대기된 호출을 검색하고 해당 호출 핸들을 반환합니다.
lineUnparkA lineUnpark 함수는 지정된 주소에 주차된 호출을 검색하고 해당 호출 핸들을 반환합니다. (lineUnparkA)
lineUnparkW lineUnparkW(유니코드) 함수(tapi.h)는 지정된 주소에 주차된 호출을 검색하고 해당 호출 핸들을 반환합니다.
phoneClose phoneClose 함수는 지정된 열린 휴대폰 디바이스를 닫습니다.
phoneConfigDialog phoneConfigDialog 함수(tapi.h)를 사용하면 지정된 디바이스의 공급자가 사용자가 관련 매개 변수를 구성할 수 있는 모달 대화 상자를 표시합니다.
phoneConfigDialogA phoneConfigDialog 함수를 사용하면 지정된 휴대폰 디바이스의 공급자가 사용자가 휴대폰 디바이스와 관련된 매개 변수를 구성할 수 있는 모달 대화 상자를 표시합니다. (phoneConfigDialogA)
phoneConfigDialogW phoneConfigDialogW(유니코드) 함수(tapi.h)를 사용하면 지정된 디바이스의 공급자가 사용자가 관련 매개 변수를 구성할 수 있는 모달 대화 상자를 표시합니다.
phoneDevSpecific phoneDevSpecific 함수는 다른 TAPI 함수에 설명되지 않은 기능을 제공하기 위해 전화 통신 API 구현을 사용하도록 설정하는 일반 확장 메커니즘으로 사용됩니다. 이러한 확장의 의미는 디바이스별로 다릅니다.

phoneGetButtonInfo phoneGetButtonInfo 함수(tapi.h)는 지정된 단추에 대한 정보를 반환합니다.
phoneGetButtonInfoA phoneGetButtonInfo 함수는 지정된 단추에 대한 정보를 반환합니다. (phoneGetButtonInfoA)
phoneGetButtonInfoW phoneGetButtonInfoW(유니코드) 함수(tapi.h)는 지정된 단추에 대한 정보를 반환합니다.
phoneGetData phoneGetData 함수는 열린 휴대폰 디바이스의 지정된 위치에서 지정된 버퍼로 정보를 업로드합니다.
phoneGetDevCaps phoneGetDevCaps 함수(tapi.h)는 지정된 휴대폰 디바이스를 쿼리하여 전화 통신 기능을 확인합니다.
phoneGetDevCapsA phoneGetDevCaps 함수는 지정된 휴대폰 디바이스를 쿼리하여 전화 통신 기능을 확인합니다. (phoneGetDevCapsA)
phoneGetDevCapsW phoneGetDevCapsW(유니코드) 함수(tapi.h)는 지정된 전화 장치를 쿼리하여 전화 통신 기능을 확인합니다.
phoneGetDisplay phoneGetDisplay 함수는 지정된 휴대폰 디스플레이의 현재 콘텐츠를 반환합니다.
phoneGetGain phoneGetGain 함수는 지정된 휴대폰의 hookswitch 디바이스의 마이크 게인 설정을 반환합니다.
phoneGetHookSwitch phoneGetHookSwitch 함수는 지정된 개방형 휴대폰 디바이스의 현재 후크위치 모드를 반환합니다.
phoneGetIcon phoneGetIcon 함수(tapi.h)를 사용하면 애플리케이션이 사용자에게 표시될 수 있는 서비스 전화 디바이스별(또는 공급자별) 아이콘을 검색할 수 있습니다.

phoneGetIconA phoneGetIcon 함수를 사용하면 애플리케이션이 사용자에게 표시할 수 있는 서비스 전화 장치별(또는 공급자별) 아이콘을 검색할 수 있습니다. (phoneGetIconA)
phoneGetIconW phoneGetIconW(유니코드) 함수(tapi.h)를 사용하면 애플리케이션이 사용자에게 표시될 수 있는 서비스 전화 디바이스별(또는 공급자별) 아이콘을 검색할 수 있습니다.
phoneGetID phoneGetID 함수(tapi.h)는 지정된 휴대폰 디바이스와 연결된 지정된 디바이스 클래스에 대한 디바이스 식별자를 반환합니다.
phoneGetIDA phoneGetID 함수는 지정된 휴대폰 디바이스와 연결된 지정된 디바이스 클래스에 대한 디바이스 식별자를 반환합니다. (phoneGetIDA)
phoneGetIDW phoneGetIDW(유니코드) 함수(tapi.h)는 지정된 휴대폰 디바이스와 연결된 지정된 디바이스 클래스에 대한 디바이스 식별자를 반환합니다.
phoneGetLamp phoneGetLamp 함수는 지정된 램프의 현재 램프 모드를 반환합니다.
phoneGetMessage phoneGetMessage 함수는 이벤트 핸들 알림 메커니즘을 사용하는 애플리케이션에 배달 대기 중인 다음 TAPI 메시지를 반환합니다(자세한 내용은 phoneInitializeEx 참조).
phoneGetRing phoneGetRing 함수를 사용하면 애플리케이션이 현재 링 모드와 같이 지정된 열린 전화 디바이스를 쿼리할 수 있습니다.
phoneGetStatus phoneGetStatus 함수(tapi.h)를 사용하면 애플리케이션이 지정된 개방형 휴대폰 디바이스의 전체 상태를 쿼리할 수 있습니다.
phoneGetStatusA phoneGetStatus 함수를 사용하면 애플리케이션이 지정된 개방형 휴대폰 디바이스의 전체 상태를 쿼리할 수 있습니다. (phoneGetStatusA)

phoneGetStatusMessages phoneGetStatusMessages 함수는 지정된 전화 디바이스에서 변경된 전화 상태가 애플리케이션에 대한 콜백을 생성하는 것을 반환합니다.
phoneGetStatusW phoneGetStatusW(유니코드) 함수(tapi.h)를 사용하면 애플리케이션이 지정된 개방형 휴대폰 디바이스의 전체 상태를 쿼리할 수 있습니다.
phoneGetVolume phoneGetVolume 함수는 지정된 휴대폰의 hookswitch 디바이스의 볼륨 설정을 반환합니다.
phoneInitialize phoneInitialize 함수는 사용되지 않습니다. TAPI 버전 1.3 및 1.4를 사용하는 애플리케이션과의 이전 버전과의 호환성을 위해 Tapi.dll 계속 내보내고 Tapi32.dll.
phoneInitializeExA phoneInitializeEx 함수는 휴대폰 추상화의 후속 사용을 위해 애플리케이션의 TAPI 사용을 초기화합니다. (ANSI)
phoneInitializeExW phoneInitializeEx 함수는 휴대폰 추상화의 후속 사용을 위해 애플리케이션의 TAPI 사용을 초기화합니다. (유니코드)
phoneNegotiateAPIVersion phoneNegotiateAPIVersion을 사용하면 애플리케이션이 지정된 전화 디바이스에 사용할 API 버전을 협상할 수 있습니다.
phoneNegotiateExtVersion phoneNegotiateExtVersion 함수를 사용하면 애플리케이션이 지정된 휴대폰 디바이스와 함께 사용할 확장 버전을 협상할 수 있습니다. 애플리케이션에서 확장을 지원하지 않는 경우 이 작업을 호출할 필요가 없습니다.
phoneOpen phoneOpen 함수는 지정된 휴대폰 디바이스를 엽니다.
phoneSetButtonInfo phoneSetButtonInfo 함수(tapi.h)는 지정된 휴대폰의 지정된 단추에 대한 정보를 설정합니다.

phoneSetButtonInfoA phoneSetButtonInfo 함수는 지정된 휴대폰의 지정된 단추에 대한 정보를 설정합니다. (phoneSetButtonInfoA)
phoneSetButtonInfoW phoneSetButtonInfoW(유니코드) 함수(tapi.h)는 지정된 휴대폰의 지정된 단추에 대한 정보를 설정합니다.
phoneSetData phoneSetData 함수는 지정된 버퍼의 정보를 선택한 데이터 식별자의 열린 휴대폰 디바이스에 다운로드합니다.
phoneSetDisplay phoneSetDisplay 함수를 사용하면 지정된 열린 휴대폰 디바이스에 지정된 문자열이 표시됩니다.
phoneSetGain phoneSetGain 함수는 지정된 후크위치 디바이스의 마이크 게인을 지정된 게인 수준으로 설정합니다.
phoneSetHookSwitch phoneSetHookSwitch 함수는 지정된 열린 휴대폰의 hookswitch 디바이스의 후크 상태를 지정된 모드로 설정합니다. 나열된 hookswitch 디바이스의 hookswitch 상태만 영향을 받습니다.
phoneSetLamp phoneSetLamp 함수를 사용하면 지정된 램프 모드에서 지정된 열린 전화 장치에 지정된 램프가 켜집니다.
phoneSetRing phoneSetRing 함수는 지정된 링 모드 및 볼륨을 사용하여 지정된 오픈 폰 디바이스를 울립니다.
phoneSetStatusMessages phoneSetStatusMessages 함수를 사용하면 애플리케이션에서 선택한 상태 이벤트에 대해 지정된 전화 디바이스를 모니터링할 수 있습니다.
phoneSetVolume phoneSetVolume 함수는 지정된 hookswitch 디바이스의 스피커 구성 요소 볼륨을 지정된 수준으로 설정합니다.

phoneShutdown phoneShutdown 함수는 TAPI의 휴대폰 추상화에 대한 애플리케이션의 사용을 종료합니다.
TAPIERROR_FORMATMESSAGE TAPIERROR_FORMATMESSAGE 매크로는 FormatMessage 함수에서 사용할 수 있는 표준 TAPI 오류 코드에 대한 식별자를 생성합니다.
tapiGetLocationInfo tapiGetLocationInfo 함수(tapi.h)는 사용자가 전화 통신 제어판 위치 매개 변수에 설정한 국가, 지역 및 도시 코드를 반환합니다.
tapiGetLocationInfoA tapiGetLocationInfo 함수는 사용자가 전화 통신 제어판 현재 위치 매개 변수에 설정한 국가 또는 지역 코드 및 도시(지역) 코드를 반환합니다. (tapiGetLocationInfoA)
tapiGetLocationInfoW tapiGetLocationInfoW(유니코드) 함수(tapi.h)는 사용자가 전화 통신 제어판 위치 매개 변수에서 설정한 국가, 지역 및 도시 코드를 반환합니다.
tapiRequestDrop tapiRequestMediaCall에 대한 이전 호출에서 수행한 호출 요청을 닫습니다.
tapiRequestMakeCall tapiRequestMakeCall 함수(tapi.h)는 음성 통화의 설정을 요청합니다.
tapiRequestMakeCallA tapiRequestMakeCall 함수는 음성 통화의 설정을 요청합니다. 호출 관리자 애플리케이션은 요청 애플리케이션을 대신하여 호출을 설정한 다음, 사용자의 호출 관리자 애플리케이션에 의해 제어됩니다. (tapiRequestMakeCallA)
tapiRequestMakeCallW tapiRequestMakeCallW(유니코드) 함수(tapi.h)는 음성 통화의 설정을 요청합니다.

콜백 함수

LINECALLBACK lineCallback 함수는 애플리케이션에서 제공하는 함수 이름의 자리 표시자입니다.
PHONECALLBACK phoneCallback 함수는 애플리케이션에서 제공하는 함수 이름의 자리 표시자입니다.

구조체

LINEADDRESSCAPS LINEADDRESSCAPS 구조체는 지정된 주소의 기능을 설명합니다. lineGetAddressCaps 함수와 TSPI_lineGetAddressCaps 함수는 LINEADDRESSCAPS 구조를 반환합니다.
LINEADDRESSSTATUS LINEADDRESSSTATUS 구조체는 주소의 현재 상태를 설명합니다. lineGetAddressStatus 함수 및 TSPI_lineGetAddressStatus 함수는 LINEADDRESSSTATUS 구조를 반환합니다.
LINEAGENTACTIVITYENTRY LINEAGENTACTIVITYENTRY 구조는 단일 ACD 에이전트 작업을 설명합니다. LINEAGENTACTIVITYLIST 구조체는 LINEAGENTACTIVITYENTRY 구조체의 배열을 포함할 수 있습니다.
LINEAGENTACTIVITYLIST LINEAGENTACTIVITYLIST 구조는 ACD 에이전트 활동 목록을 설명합니다. 이 구조체는 LINEAGENTACTIVITYENTRY 구조체의 배열을 포함할 수 있습니다. lineGetAgentActivityList 함수는 LINEAGENTACTIVITYLIST 구조를 반환합니다.
LINEAGENTCAPS LINEAGENTCAPS 구조는 ACD 에이전트의 기능을 설명합니다. lineGetAgentCaps 함수는 LINEAGENTCAPS 구조를 반환합니다.
LINEAGENTENTRY LINEAGENTENTRY 구조는 개별 ACD 에이전트를 설명합니다. LINEAGENTLIST 구조체는 LINEAGENTENTRY 구조체의 배열을 포함할 수 있습니다.

LINEAGENTGROUPENTRY LINEAGENTGROUPENTRY 구조체는 ACD 에이전트 그룹에 대한 정보를 제공합니다. LINEAGENTGROUPLIST 구조체는 LINEAGENTGROUPENTRY 구조체의 배열을 포함할 수 있습니다.
LINEAGENTGROUPLIST LINEAGENTGROUPLIST 구조체는 ACD 에이전트 그룹 목록을 설명합니다. 이 구조체는 LINEAGENTGROUPENTRY 구조체의 배열을 포함할 수 있습니다.
LINEAGENTINFO LINEAGENTINFO 구조체에는 ACD 에이전트에 대한 정보가 포함되어 있습니다. lineGetAgentInfo 함수는 LINEAGENTINFO 구조를 반환합니다.
LINEAGENTLIST LINEAGENTLIST 구조는 ACD 에이전트 목록을 설명합니다. 이 구조체는 LINEAGENTENTRY 구조체의 배열을 포함할 수 있습니다.
LINEAGENTSESSIONENTRY LINEAGENTSESSIONENTRY 구조체는 ACD 에이전트 세션을 설명합니다. LINEAGENTSESSIONLIST 구조체에는 LINEAGENTSESSIONENTRY 구조체의 배열이 포함될 수 있습니다.
LINEAGENTSESSIONINFO LINEAGENTSESSIONINFO 구조체에는 ACD 에이전트 세션에 대한 정보가 포함되어 있습니다. lineGetAgentSessionInfo 함수는 LINEAGENTSESSIONINFO 구조를 반환합니다.
LINEAGENTSESSIONLIST LINEAGENTSESSIONLIST 구조체는 ACD 에이전트 세션 목록을 설명합니다. 이 구조체는 LINEAGENTSESSIONENTRY 구조체의 배열을 포함할 수 있습니다. lineGetAgentSessionList 함수는 LINEAGENTSESSIONLIST 구조를 반환합니다.
LINEAGENTSTATUS LINEAGENTSTATUS 구조체는 ACD 에이전트의 현재 상태를 설명합니다. lineGetAgentStatus 함수는 LINEAGENTSTATUS 구조를 반환합니다.
LINEAPPINFO LINEAPPINFO 구조에는 현재 실행 중인 애플리케이션에 대한 정보가 포함되어 있습니다. LINEDEVSTATUS 구조체는 LINEAPPINFO 구조체의 배열을 포함할 수 있습니다.

LINECALLHUBTRACKINGINFO LINECALLHUBTRACKINGINFO 구조에는 호출 허브에서 사용할 수 있는 추적 유형을 보고하는 정보가 포함되어 있습니다. 이 구조는 TAPI 버전 2.2 이상을 협상하는 애플리케이션에만 노출됩니다.
LINECALLINFO LINECALLINFO 구조에는 호출에 대한 정보가 포함됩니다.
LINECALLLIST LINECALLLIST 구조는 호출 핸들 목록을 설명합니다. 이 형식의 구조체는 lineGetNewCalls 및 lineGetConfRelatedCalls 함수에 의해 반환됩니다.
LINECALLPARAMS LINECALLPARAMS 구조는 lineMakeCall 및 TSPI_lineMakeCall 함수를 사용하여 호출할 때 제공되는 매개 변수를 설명합니다. LINECALLPARAMS 구조체는 lineOpen 함수와 같은 다른 작업에서도 매개 변수로 사용됩니다.
LINECALLSTATUS LINECALLSTATUS 구조체는 호출의 현재 상태를 설명합니다.
LINECALLTREATMENTENTRY LINECALLTREATMENTENTRY 구조는 현재 통화에서 음악, 녹음된 알림 또는 무음과 같은 통화 처리 유형에 대한 정보를 제공합니다. LINEADDRESSCAPS 구조체는 LINECALLTREATMENTENTRY 구조체의 배열을 포함할 수 있습니다.
LINECARDENTRY LINECARDENTRY 구조는 호출 카드를 설명합니다. LINETRANSLATECAPS 구조체는 LINECARDENTRY 구조체의 배열을 포함할 수 있습니다.
LINECOUNTRYENTRY 단일 국가/지역 항목에 대한 데이터를 제공합니다.
LINECOUNTRYLIST LINECOUNTRYLIST 구조는 국가/지역 목록을 설명합니다. 이 구조체는 LINECOUNTRYENTRY 구조체의 배열을 포함할 수 있습니다. LINECOUNTRYLIST는 lineGetCountry 함수에 의해 반환됩니다.
LINEDEVCAPS LINEDEVCAPS 구조는 라인 디바이스의 기능을 설명합니다. lineGetDevCaps 함수와 TSPI_lineGetDevCaps 함수는 LINEDEVCAPS 구조를 반환합니다.

LINEDEVSTATUS LINEDEVSTATUS 구조는 라인 디바이스의 현재 상태를 설명합니다. lineGetLineDevStatus 함수와 TSPI_lineGetLineDevStatus 함수는 LINEDEVSTATUS 구조를 반환합니다.
LINEDIALPARAMS LINEDIALPARAMS 구조체는 전화 걸기 관련 필드의 컬렉션을 지정합니다. lineSetCallParams 함수 또는 TSPI_lineSetCallParams 함수를 호출하여 LINEDIALPARAMS 구조를 사용하여 호출에 대한 매개 변수를 설정합니다.
LINEEXTENSIONID LINEEXTENSIONID 구조체는 확장 식별자를 설명합니다.
LINEFORWARD LINEFORWARD 구조는 전달 명령의 항목을 설명합니다. LINEFORWARDLIST 및 LINEADDRESSSTATUS 구조체는 LINEFORWARD 구조체의 배열을 포함할 수 있습니다.
LINEFORWARDLIST LINEFORWARDLIST 구조는 전달 지침 목록을 설명합니다. 이 구조체는 LINEFORWARD 구조체의 배열을 포함할 수 있습니다. lineForward 및 TSPI_lineForward 함수는 LINEFORWARDLIST 구조를 사용합니다.
LINEGENERATETONE LINEGENERATETONE 구조체에는 생성할 톤에 대한 정보가 포함됩니다. 이 구조체는 lineGenerateTone 및 TSPI_lineGenerateTone 함수에서 사용됩니다.
LINEINITIALIZEEXPARAMS LINEINITIZALIZEEXPARAMS 구조체는 LINEINITIALIZEEX를 사용하여 호출할 때 제공되는 매개 변수를 설명합니다.
LINELOCATIONENTRY 주소 번역 컨텍스트를 제공하는 데 사용되는 위치를 설명합니다.
LINEMEDIACONTROLCALLSTATE LINEMEDIACONTROLCALLSTATE 구조는 하나 이상의 호출 상태로 전환을 검색할 때 실행할 미디어 작업을 설명합니다. lineSetMediaControl 및 TSPI_lineSetMediaControl 함수는 이 구조를 사용합니다.

LINEMEDIACONTROLDIGIT LINEMEDIACONTROLDIGIT 구조체는 숫자를 검색할 때 실행할 미디어 동작을 설명합니다. 배열의 항목으로 사용됩니다. lineSetMediaControl 및 TSPI_lineSetMediaControl 함수는 이 구조를 사용합니다.
LINEMEDIACONTROLMEDIA LINEMEDIACONTROLMEDIA 구조는 미디어 형식 변경을 검색할 때 실행할 미디어 동작을 설명합니다. 배열의 항목으로 사용됩니다. lineSetMediaControl 및 TSPI_lineSetMediaControl 함수는 이 구조를 사용합니다.
LINEMEDIACONTROLTONE LINEMEDIACONTROLTONE 구조는 톤이 감지되었을 때 실행할 미디어 동작을 설명합니다. 배열의 항목으로 사용됩니다. lineSetMediaControl 및 TSPI_lineSetMediaControl 함수는 이 구조를 사용합니다.
LINEMESSAGE LINEMESSAGE 구조체에는 애플리케이션이 현재 열려 있는 줄의 상태 변경을 지정하는 매개 변수 값이 포함되어 있습니다. lineGetMessage 함수는 LINEMESSAGE 구조를 반환합니다.
LINEMONITORTONE LINEMONITORTONE 구조는 모니터링할 톤을 설명합니다. 이 항목은 배열의 항목으로 사용됩니다. lineMonitorTones 및 TSPI_lineMonitorTones 함수는 이 구조를 사용합니다.
LINEPROVIDERENTRY LINEPROVIDERENTRY 구조는 단일 서비스 공급자 항목에 대한 정보를 제공합니다. 이러한 구조체의 배열은 lineGetProviderList 함수에서 반환하는 LINEPROVIDERLIST 구조의 일부로 반환됩니다.
LINEPROVIDERLIST LINEPROVIDERLIST 구조는 서비스 공급자 목록을 설명합니다. 이 형식의 구조는 lineGetProviderList 함수에 의해 반환됩니다. LINEPROVIDERLIST 구조체는 LINEPROVIDERENTRY 구조의 배열을 포함할 수 있습니다.
LINEPROXYREQUEST LINEPROXYREQUEST 구조에는 프록시 요청을 만드는 애플리케이션의 매개 변수 값이 포함됩니다. 여러 TAPI 콜 센터 함수는 LINEPROXYREQUEST 구조를 참조하는 LINE_PROXYREQUEST 메시지를 생성합니다.

LINEPROXYREQUESTLIST LINEPROXYREQUESTLIST 구조는 프록시 요청 목록을 설명합니다. lineGetProxyStatus 함수는 LINEPROXYREQUESTLIST 구조를 반환합니다.
LINEQUEUEENTRY LINEQUEUEENTRY 구조는 단일 큐 항목에 대한 정보를 제공합니다. LINEQUEUELIST 구조체에는 LINEQUEUEENTRY 구조의 배열이 포함될 수 있습니다. 이 구조에는 TAPI 3.0 버전 협상이 필요합니다.
LINEQUEUEINFO LINEQUEUEINFO 구조는 줄 디바이스의 큐에 대한 정보를 제공합니다. lineGetQueueInfo 함수는 LINEQUEUEINFO 구조를 반환합니다. 이 구조에는 TAPI 3.0 버전 협상이 필요합니다.
LINEQUEUELIST LINEQUEUELIST 구조는 큐 목록을 설명합니다. 이 구조체는 LINEQUEUEENTRY 구조의 배열을 포함할 수 있습니다. lineGetQueueList 함수는 LINEQUEUELIST 구조를 반환합니다. LINEQUEUELIST에는 TAPI 3.0 버전 협상이 필요합니다.
LINEREQMAKECALL LINEREQMAKECALL 구조는 lineGetRequest 함수에 대한 호출로 시작된 요청을 설명합니다.
LINEREQMEDIACALL lineGetRequest 함수에 대한 호출로 시작된 요청에 대해 설명합니다. 이 데이터 구조는 사용되지 않으며 사용하면 안 됩니다.
LINETERMCAPS LINETERMCAPS 구조는 라인의 터미널 디바이스 기능을 설명합니다. LINEDEVCAPS 구조체에는 LINETERMCAPS 구조체의 배열이 포함될 수 있습니다.
LINETRANSLATECAPS LINETRANSLATECAPS 구조체는 주소 변환 기능을 설명합니다.
LINETRANSLATEOUTPUT LINETRANSLATEOUTPUT 구조체는 주소 변환의 결과를 설명합니다. lineTranslateAddress 함수는 이 구조를 사용합니다.
PHONEBUTTONINFO PHONEBUTTONINFO 구조에는 전화 장치의 단추에 대한 정보가 포함되어 있습니다. 이 구조는 여러 TAPI 및 TSPI 함수에서 사용됩니다.

PHONECAPS PHONECAPS 구조는 휴대폰 디바이스의 기능을 설명합니다. phoneGetDevCaps 및 TSPI_phoneGetDevCaps 함수는 이 구조를 반환합니다.
PHONEEXTENSIONID PHONEEXTENSIONID 구조체는 확장 식별자를 설명합니다.
PHONEINITIALIZEEXPARAMS PHONEINITIALIZEEXPARAMS 구조에는 애플리케이션과 TAPI 간의 연결을 설정하는 데 사용되는 매개 변수가 포함되어 있습니다. 예를 들어 애플리케이션이 선택한 이벤트 알림 메커니즘입니다. phoneInitializeEx 함수는 이 구조를 사용합니다.
PHONEMESSAGE PHONEMESSAGE 구조에는 애플리케이션에 배달하기 위해 큐에 대기된 다음 메시지가 포함됩니다. phoneGetMessage 함수는 이 구조를 반환합니다.
PHONESTATUS PHONESTATUS 구조는 전화 장치의 현재 상태를 설명합니다. phoneGetStatus 및 TSPI_phoneGetStatus 함수는 이 구조를 반환합니다.
VARSTRING VARSTRING 구조체는 고정 크기 문자열을 반환하는 데 사용됩니다. 이 클래스는 라인 디바이스 클래스와 전화 장치 클래스에서 모두 사용됩니다.

tapi3.h 헤더

아티클2023. 03. 08.

이 헤더는 전화 통신 애플리케이션 프로그래밍 인터페이스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 전화 통신 애플리케이션 프로그래밍 인터페이스

tapi3.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumACDGroup IEnumACDGroup(tapi3.h) 인터페이스는 ITACDGroup 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다.
IEnumAgent IEnumAgent(tapi3.h)는 ITAgent에 대한 열거자 인터페이스입니다.
IEnumAgentHandler IEnumAgentHandler(tapi3.h) 인터페이스는 ITAgentHandler 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다.
IEnumAgentSession IEnumAgentSession(tapi3.h) 인터페이스는 ITAgentSession 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다.
IEnumQueue IEnumQueue(tapi3.h) 인터페이스는 ITQueue 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다.
ITACDGroup ITACDGroup(tapi3.h) 개체는 ACD 파일럿, 분할 또는 그룹을 반영합니다. ACD는 스위칭 시스템 내에서 호출을 큐에 대기하고 배포하는 메커니즘입니다.
ITACDGroupEvent ITACDGroupEvent(tapi3.h) 인터페이스에는 ACD(자동 호출 배포) 그룹 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.

ITAgent ITAgent(tapi3.h) 인터페이스는 IDispatch 인터페이스에서 상속됩니다. 에이전트는 콜 센터의 핵심입니다.
ITAgentEvent ITAgentEvent(tapi3.h) 인터페이스에는 에이전트 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITAgentHandler ITAgentHandler(tapi3.h) 인터페이스는 에이전트 개체를 만들고 ACD(자동 호출 배포) 그룹을 열거하는 메서드를 제공합니다.
ITAgentHandlerEvent ITAgentHandlerEvent(tapi3.h) 인터페이스에는 에이전트 처리기 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITAgentSession ITAgentSession(tapi3.h)의 메서드를 사용하면 애플리케이션이 통계를 검색할 수 있습니다. 에이전트 세션은 에이전트, 그룹 및 주소 간의 연결을 나타냅니다.
ITAgentSessionEvent ITAgentSessionEvent 인터페이스(tapi3.h)에는 에이전트 세션 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITAllocatorProperties ITAllocatorProperties 인터페이스(tapi3.h)는 MST(미디어 스트리밍 터미널)의 버퍼 할당자 속성을 최종 사용자 또는 서버 애플리케이션에 노출합니다.
ITAMMediaFormat ITAMMediaFormat 인터페이스(tapi3.h)는 DirectShow 미디어 형식을 설정하고 가져옵니다.
ITMSPAddress ITMSPAddress 인터페이스(tapi3.h)는 MSP에 의해 구현되며 TAPI DLL에 대한 미디어 서비스 공급자를 나타냅니다.
ITPluggableTerminalEventSink ITPluggableTerminalEventSink 인터페이스(tapi3.h)는 플러그형 터미널의 변경 내용을 클라이언트 애플리케이션에 알리는 메시지를 발생시키는 메서드를 제공합니다.

ITPluggableTerminalEventSinkRegistration ITPluggableTerminalEventSinkRegistration 인터페이스(tapi3.h)는 플러그형 터미널 이벤트에 대한 클라이언트 애플리케이션을 등록 및 등록 취소합니다.
ITQueue ITQueue 인터페이스(tapi3.h)는 큐에 관한 정보를 가져오고 설정합니다.
ITQueueEvent ITQueueEvent 인터페이스(tapi3.h)에는 ACD(자동 호출 배포) 큐 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITTAPICallCenter ITTAPICallCenter 인터페이스(tapi3.h)는 콜 센터 컨트롤에 대한 진입점을 제공합니다.

구조체

MSP_EVENT_INFO MSP_EVENT_INFO 구조체(tapi3.h)는 GetEvent 메서드에서 반환되는 이벤트 형식을 정의합니다.

열거형

ACDGROUP_EVENT ACDGROUP_EVENT 열거형(tapi3.h)은 ACD 그룹 이벤트를 설명합니다.
ACDQUEUE_EVENT ACDQUEUE_EVENT 열거형(tapi3.h)은 ACD 큐 이벤트를 설명합니다.
AGENT_EVENT AGENT_EVENT 열거형(tapi3.h)은 에이전트 이벤트를 설명합니다.

AGENT_SESSION_EVENT AGENT_SESSION_EVENT 열거형(tapi3.h)은 에이전트 세션 이벤트를 설명합니다.
AGENT_SESSION_STATE AGENT_SESSION_STATE 열거형(tapi3.h)은 ITAgentSession::get_State 및 ITAgentSession::p ut_State 메서드에서 사용하는 에이전트 세션 표시기를 정의합니다.
AGENT_STATE AGENT_STATE 열거형(tapi3.h)은 ITAgent::p ut_State 및 ITAgent::get_State 메서드에서 에이전트 상태를 설명하는 데 사용됩니다.
AGENTHANDLER_EVENT AGENTHANDLER_EVENT 열거형(tapi3.h)은 에이전트 처리기 이벤트를 설명합니다.
MSP_ADDRESS_EVENT MSP_ADDRESS_EVENT 열거형(tapi3.h) 상수는 MSP_EVENT ME_ADDRESS_EVENT 때 GetEvent 메서드에 의해 MSP_EVENT_INFO 구조체 내에서 반환됩니다.
MSP_CALL_EVENT MSP_CALL_EVENT 열거형(tapi3.h) 상수는 MSP_EVENT ME_CALL_EVENT 때 GetEvent 메서드에 의해 MSP_EVENT_INFO 구조체 내에 반환됩니다.
MSP_CALL_EVENT_CAUSE MSP_EVENT ME_CALL_EVENT 때 MSP_CALL_EVENT_CAUSE 열거형(tapi3.h) 상수는 GetEvent 메서드에 의해 MSP_EVENT_INFO 구조체 내에서 반환됩니다.
MSP_EVENT MSP_EVENT 열거자(tapi3.h)는 MSP_EVENT_INFO 포함된 이벤트 정보의 형식을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

tapi3cc.h 헤더

아티클2023. 08. 26.

이 헤더는 전화 통신 애플리케이션 프로그래밍 인터페이스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 전화 통신 애플리케이션 프로그래밍 인터페이스

tapi3cc.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumACDGroup IEnumACDGroup 인터페이스(tapi3cc.h)는 ITACDGroup 인터페이스에 대한 COM 표준 열거형 메서드를 제공합니다.
IEnumAgent IEnumAgent(tapi3cc.h)는 ITAgent의 열거자 인터페이스입니다.
IEnumAgentHandler IEnumAgentHandler 인터페이스(tapi3cc.h)는 ITAgentHandler 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다.
IEnumAgentSession IEnumAgentSession 인터페이스(tapi3cc.h)는 ITAgentSession 인터페이스에 대한 COM 표준 열거형 메서드를 제공합니다.
IEnumQueue IEnumQueue 인터페이스(tapi3cc.h)는 ITQueue 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다.
ITACDGroup ITACDGroup 인터페이스(tapi3cc.h)는 전환 시스템 내에서 호출을 큐에 대기하고 배포하는 ACD(자동 호출 분포) 메커니즘을 처리합니다.
ITACDGroupEvent ITACDGroupEvent 인터페이스(tapi3cc.h)에는 ACD(자동 호출 배포) 그룹 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.

ITAgent ITAgent 인터페이스(tapi3cc.h)는 들어오는 호출을 수신 및 처리하고 고객 또는 잠재 고객에게 발신 통화를 하는 에이전트 개체를 처리합니다.
ITAgentEvent ITAgentEvent 인터페이스(tapi3cc.h)에는 에이전트 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITAgentHandler ITAgentHandler 인터페이스(tapi3cc.h)는 에이전트 개체를 만들고 ACD(자동 호출 배포) 그룹을 열거하는 메서드를 제공합니다.
ITAgentHandlerEvent ITAgentHandlerEvent 인터페이스(tapi3cc.h)에는 에이전트 처리기 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITAgentSession ITAgentSession 인터페이스(tapi3cc.h)는 에이전트, 그룹 및 주소 간의 연결을 나타내는 에이전트 세션을 처리합니다.
ITAgentSessionEvent ITAgentSessionEvent 인터페이스(tapi3cc.h)에는 에이전트 세션 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITQueue ITQueue 인터페이스(tapi3cc.h)는 큐와 관련된 정보를 가져오고 설정합니다.
ITQueueEvent ITQueueEvent 인터페이스(tapi3cc.h)에는 ACD(자동 호출 배포) 큐 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITTAPICallCenter ITTAPICallCenter 인터페이스(tapi3cc.h)는 콜 센터 컨트롤에 진입점을 제공합니다.

열거형

ACDGROUP_EVENT ACDGROUP_EVENT 열거형(tapi3cc.h)은 ACD 그룹 이벤트를 설명합니다.
ACDQUEUE_EVENT ACDQUEUE_EVENT 열거형(tapi3cc.h)은 ACD 큐 이벤트를 설명합니다.
AGENT_EVENT AGENT_EVENT 열거형(tapi3cc.h)은 에이전트 이벤트를 설명합니다.
AGENT_SESSION_EVENT AGENT_SESSION_EVENT 열거형(tapi3cc.h)은 에이전트 세션 이벤트를 설명합니다.
AGENT_SESSION_STATE 이 AGENT_SESSION_STATE 열거형(tapi3cc.h)은 ITAgentSession::get_State 및 ITAgentSession::put_State 메서드에서 사용하는 에이전트 세션 표시기를 정의합니다.
AGENT_STATE AGENT_STATE 열거형(tapi3cc.h)은 ITAgent::p ut_State 및 ITAgent::get_State 메서드에서 에이전트 상태를 설명하는 데 사용됩니다.
AGENTHANDLER_EVENT AGENTHANDLER_EVENT 열거형(tapi3cc.h)은 에이전트 처리기 이벤트를 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

tapi3ds.h 헤더

아티클2023. 03. 08.

이 헤더는 전화 통신 애플리케이션 프로그래밍 인터페이스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 전화 통신 애플리케이션 프로그래밍 인터페이스

tapi3ds.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ITAllocatorProperties ITAllocatorProperties 인터페이스(tapi3ds.h)는 MST(미디어 스트리밍 터미널)의 버퍼 할당자 속성을 최종 사용자 또는 서버 애플리케이션에 노출합니다.
ITAMMediaFormat ITAMMediaFormat 인터페이스(tapi3ds.h)는 DirectShow 미디어 형식을 설정하고 가져옵니다.

tapi3if.h 헤더

아티클2023. 03. 08.

이 헤더는 전화 통신 애플리케이션 프로그래밍 인터페이스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 전화 통신 애플리케이션 프로그래밍 인터페이스

tapi3if.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumAddress IEnumAddress 인터페이스는 ITAddress 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다. ITTAPI::EnumerateAddresses 및 ITAgentHandler::EnumerateUsableAddresses 메서드는 IEnumAddress에 대한 포인터를 반환합니다.
IEnumBstr IEnumBstr 인터페이스는 BSTR 문자열을 열거하는 COM 표준 메서드를 제공합니다.
IEnumCall IEnumCall 인터페이스는 ITCallInfo 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다. ITCallHub::EnumerateCalls 및 ITAddress::EnumerateCalls 메서드는 IEnumCall에 대한 포인터를 반환합니다.
IEnumCallHub IEnumCallHub 인터페이스는 ITCallHub 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다. ITTAPI::EnumerateCallHubs 메서드는 IEnumCallHub에 대한 포인터를 반환합니다.
IEnumCallingCard IEnumCallingCard 인터페이스는 ITCallingCard 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다. ITAddressTranslation::EnumerateCallingCards 메서드는 IEnumCallingCard에 대한 포인터를 반환합니다.
IEnumLocation IEnumLocation 인터페이스는 ITLocationInfo 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다. ITAddressTranslation::EnumerateLocations 메서드는 IEnumLocation에 대한 포인터를 반환합니다.

IEnumPhone IEnumPhone 인터페이스는 ITPhone 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다. ITAddress2::EnumeratePhones 및 ITTAPI2::EnumeratePhones 메서드는 IEnumPhone에 대한 포인터를 반환합니다.
IEnumPluggableSuperclassInfo IEnumPluggableSuperclassInfo 인터페이스는 ITPluggableTerminalSuperclassInfo 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다. ITTerminalSupport2::EnumeratePluggableSuperclasses 메서드는 IEnumPluggableSuperclassInfo에 대한 포인터를 반환합니다.
IEnumPluggableTerminalClassInfo IEnumPluggableTerminalClassInfo 인터페이스는 ITPluggableTerminalClassInfo 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다. ITTerminalSupport2::EnumeratePluggableTerminalClasses 메서드는 IEnumPluggableTerminalClassInfo에 대한 포인터를 반환합니다.
IEnumStream IEnumStream 인터페이스는 ITStream 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다. ITStreamControl::EnumerateStreams 및 ITParticipant::EnumerateStreams 메서드는 IEnumStream에 대한 포인터를 반환합니다.
IEnumSubStream IEnumSubStream 인터페이스는 ITSubStream 인터페이스에 대한 COM 표준 열거 메서드를 제공합니다. ITSubStreamControl::EnumerateSubStreams 메서드는 IEnumSubStream에 대한 포인터를 반환합니다.
IEnumTerminal IEnumTerminal 인터페이스는 ITTerminal 인터페이스에 대한 COM 표준 열거형 메서드를 제공합니다.
IEnumTerminalClass IEnumTerminalClass 인터페이스는 사용 가능한 동적 터미널 클래스를 검색하고 사용하는 COM 표준 열거형 메서드를 제공합니다. ITTerminalSupport::EnumerateDynamicTerminalClasses 메서드는 이 인터페이스에 대한 포인터를 반환합니다.
ITAddress ITAddress 인터페이스는 Address 개체의 기본 인터페이스입니다. 애플리케이션은 이 인터페이스를 사용하여 Address 개체에 대한 정보를 얻고 사용합니다.

ITAddress2 ITAddress2 인터페이스는 ITAddress 인터페이스에서 파생됩니다. ITAddress2는 전화 디바이스를 지원하기 위해 Address 개체에 메서드를 추가합니다. TAPI 3.1에서 열거된 모든 Address 개체는 이 인터페이스를 자동으로 구현합니다.
ITAddressCapabilities ITAddressCapabilities 인터페이스는 주소의 기능에 대한 정보를 가져오는 데 사용됩니다. Address 개체에 있으며 애플리케이션은 Address 개체에서 QueryInterface를 호출하여 액세스할 수 있습니다.
ITAddressDeviceSpecificEvent ITAddressDeviceSpecificEvent는 애플리케이션이 디바이스별 이벤트에 대한 정보를 검색할 수 있도록 하는 메서드를 노출합니다.
ITAddressEvent ITAddressEvent 인터페이스에는 주소 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITAddressTranslation ITAddressTranslation 인터페이스는 호출 주소를 다른 형식으로 변환할 수 있는 메서드를 제공합니다. 예를 들어 애플리케이션은 전화를 걸기 전에 주소를 정식 주소에서 전화 접속 가능 주소로 변환해야 할 수 있습니다.
ITAddressTranslationInfo 주소 변환 데이터를 확인하는 데 사용됩니다.
ITASRTerminalEvent ITASRTerminalEvent 인터페이스에는 발생한 자동 음성 인식 터미널 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITAutomatedPhoneControl ITAutomatedPhoneControl은 TAPI 전화 개체에 의해 노출되는 완전 OLE 자동화 및 스크립트 가능 인터페이스입니다.
ITBasicAudioTerminal ITBasicAudioTerminal 인터페이스는 애플리케이션이 터미널의 기본 사운드 특성을 제어할 수 있도록 하는 메서드를 제공합니다.

ITBasicCallControl ITBasicCallControl 인터페이스는 애플리케이션에서 호출 개체에 대한 기본 전화 통신 작업을 연결, 응답 및 수행하는 데 사용됩니다.
ITBasicCallControl2 ITBasicCallControl2 인터페이스는 ITBasicCallControl 인터페이스의 확장입니다.
ITCallHub ITCallHub 인터페이스는 CallHub 개체와 관련된 정보를 검색하는 메서드를 제공합니다. IEnumCallHub::Next 및 ITTapi::get_CallHubs 메서드는 ITCallHub 인터페이스를 만듭니다.
ITCallHubEvent ITCallHubEvent 인터페이스에는 CallHub 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITCallInfo ITCallInfo 인터페이스는 Call 개체와 관련된 다양한 정보를 가져오고 설정합니다. ITAddress::get_Calls 및 IEnumCall::Next 메서드는 ITCallInfo 인터페이스를 만듭니다.
ITCallInfo2 ITCallInfo2 인터페이스는 ITCallInfo 인터페이스의 확장입니다. ITCallInfo2는 애플리케이션이 호출별로 이벤트 필터링을 설정할 수 있도록 하는 추가 메서드를 제공합니다.
ITCallInfoChangeEvent ITCallInfoChangeEvent 인터페이스에는 호출 정보 변경 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITCallingCard ITCallingCard 인터페이스는 전화 통신 통화 카드와 관련된 정보를 검색하는 메서드를 제공합니다.
ITCallMediaEvent ITCallMediaEvent 인터페이스에는 미디어 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITCallNotificationEvent ITCallNotificationEvent 인터페이스에는 호출 알림 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.

ITCallStateEvent ITCallStateEvent 인터페이스에는 호출 상태 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITCollection ITCollection 인터페이스를 사용하면 Visual Basic으로 작성된 애플리케이션과 같은 Automation 클라이언트 애플리케이션에서 컬렉션 정보를 검색할 수 있습니다.
ITCollection2 ITCollection2 인터페이스는 ITCollection 인터페이스의 확장입니다. ITCollection2는 컬렉션을 수정하기 위한 추가 메서드를 노출합니다.
ITCustomTone ITCustomTone 인터페이스는 일부 전화 세트에서 사용할 수 있는 사용자 지정 톤을 자세히 제어할 수 있는 메서드를 노출합니다.
ITDetectTone ITDetectTone 인터페이스는 애플리케이션이 TAPI 서버에서 톤 이벤트를 생성해야 하는 톤 및 톤 특성을 지정할 수 있도록 하는 메서드를 노출합니다.
ITDigitDetectionEvent ITDigitDetectionEvent 인터페이스에는 DTMF 숫자 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITDigitGenerationEvent ITDigitGenerationEvent 인터페이스에는 숫자 생성 이벤트를 설명하는 메서드가 포함되어 있습니다.
ITDigitsGatheredEvent ITDigitsGatheredEvent 인터페이스는 TAPI 서버가 서버가 애플리케이션에 필요한 숫자를 수집했음을 나타내는 이벤트를 보낼 때 애플리케이션이 데이터를 검색할 수 있도록 하는 메서드를 노출합니다.
ITDispatchMapper ITDispatchMapper 인터페이스를 사용하면 한 인터페이스의 디스패치 포인터와 다른 인터페이스의 GUID를 고려할 때 애플리케이션이 개체에서 다른 인터페이스의 디스패치 포인터를 검색할 수 있습니다.

ITFileTerminalEvent ITFileTerminalEvent 인터페이스에는 발생한 파일 터미널 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITFileTrack ITFileTrack 인터페이스는 애플리케이션이 파일 터미널 트랙과 관련된 정보를 가져와서 설정할 수 있도록 하는 메서드를 노출합니다. ITFileTerminalEvent::get_Track 메서드는 ITFileTrack 인터페이스를 만듭니다.
ITForwardInformation ITForwardInformation 인터페이스는 착신 전환의 설정 및 구현을 위한 메서드를 제공합니다.
ITForwardInformation2 ITForwardInformation2 인터페이스는 전달 정보를 제어하기 위한 추가 메서드를 제공하는 메서드를 노출합니다. 기본 전달 제어 메서드는 ITForwardInformation을 참조하세요.
ITLegacyAddressMediaControl ITLegacyAddressMediaControl 인터페이스는 디바이스 및 해당 구성에 직접 액세스해야 하는 레거시 애플리케이션을 지원하기 위해 제공됩니다. Address 개체에 의해 노출되며 ITAddress에서 QueryInterface를 호출하여 만들 수 있습니다.
ITLegacyAddressMediaControl2 ITLegacyAddressMediaControl2 인터페이스는 ITLegacyAddressMediaControl 인터페이스에서 파생됩니다. ITLegacyAddressMediaControl2는 줄 디바이스와 관련된 매개 변수의 구성을 허용하는 추가 메서드를 제공합니다.
ITLegacyCallMediaControl ITLegacyCallMediaControl 인터페이스는 디바이스와 직접 통신해야 하는 레거시 애플리케이션을 지원합니다. 이 인터페이스는 호출 개체에 노출되며 ITBasicCallControl에서 QueryInterface를 호출하여 만들 수 있습니다.
ITLegacyCallMediaControl2 ITLegacyCallMediaControl2 인터페이스는 ITLegacyCallMediaControl 인터페이스의 확장입니다. ITLegacyCallMediaControl2는 주로 톤 감지 및 생성을 위한 추가 메서드를 제공합니다.
ITLegacyWaveSupport ITLegacyWaveSupport 인터페이스를 사용하면 애플리케이션이 Wave API를 사용하여 레거시 TSP(TAPI 3 이전)에서 만든 터미널을 제어할 수 있는지 여부를 검색할 수 있습니다.

ITLocationInfo ITLocationInfo 인터페이스는 통화 당사자의 위치와 관련된 정보를 가져오는 데 사용됩니다. 이 정보는 제어판 아래에 있는 전화 통신 애플릿을 사용하여 입력되는 위치 정보입니다.
ITMediaControl ITMediaControl 인터페이스는 미디어 파일 터미널에 대한 일반 인터페이스입니다. 인터페이스는 애플리케이션이 재생과 같은 현재 작업을 시작, 중지 또는 일시 중지할 수 있도록 하는 메서드를 노출합니다.
ITMediaPlayback ITMediaPlayback 인터페이스는 애플리케이션이 재생할 파일 목록을 설정하고 가져올 수 있도록 하는 재생 관련 메서드를 제공합니다. 이 인터페이스는 ITTerminal에서 QueryInterface를 호출하여 만듭니다.
ITMediaRecord ITMediaRecord 인터페이스는 애플리케이션이 기록할 파일의 이름을 설정하고 가져올 수 있도록 하는 기록 관련 메서드를 제공합니다.
ITMediaSupport ITMediaSupport 인터페이스는 애플리케이션이 이 인터페이스를 노출하는 주소 개체에 대한 미디어 지원 기능을 검색할 수 있도록 하는 메서드를 제공합니다.
ITMultiTrackTerminal 이 ITMultiTrackTerminal 인터페이스는 모든 멀티트랙 터미널에 노출됩니다. 인터페이스에는 트랙을 열거, 생성 및 제거하는 메서드가 포함됩니다. ITMultiTrackTerminal 인터페이스는 ITTerminal에서 QueryInterface를 호출하여 만들어집니다.
ITPhone ITPhone 인터페이스는 TAPI 3.1 개체 모델의 새 Phone 개체에 대한 기본 인터페이스입니다.
ITPhoneDeviceSpecificEvent ITPhoneDeviceSpecificEvent는 애플리케이션이 휴대폰 디바이스 관련 이벤트에 대한 정보를 검색할 수 있도록 하는 메서드를 노출합니다.
ITPhoneEvent ITPhoneEvent 인터페이스에는 발생한 전화 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.

ITPluggableTerminalClassInfo ITPluggableTerminalClassInfo 인터페이스는 애플리케이션이 플러그형 터미널과 관련된 정보를 검색할 수 있도록 하는 메서드를 노출합니다.
ITPluggableTerminalSuperclassInfo ITPluggableTerminalSuperclassInfo 인터페이스는 플러그형 터미널 클래스의 이름과 CLSID를 가져오는 메서드를 노출합니다.
ITPrivateEvent ITPrivateEvent 인터페이스는 애플리케이션이 프라이빗 개체에 의해 생성된 이벤트를 처리할 수 있도록 하는 메서드를 노출합니다.
ITQOSEvent ITQOSEvent 인터페이스에는 QOS(서비스 품질) 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITRequest ITRequest 인터페이스를 사용하면 애플리케이션에서 보조 전화 통신을 사용할 수 있습니다. 보조 전화 통신은 전화 통신을 사용하는 애플리케이션에 개발자가 전화 통신을 완전히 읽지 않아도 전화를 걸 수 있는 간단한 메커니즘을 제공합니다.
ITRequestEvent ITRequestEvent 인터페이스에는 애플리케이션이 보조 전화 통신 요청 이벤트를 수신하고 처리할 수 있도록 하는 메서드가 포함되어 있습니다.
ITScriptableAudioFormat ITScriptableAudioFormat 인터페이스는 스크립팅 가능한 클라이언트가 트랙의 오디오 형식을 얻거나 오디오 형식을 설정하는 데 사용됩니다. 인터페이스는 WAVEFORMATEX 구조체의 각 멤버에 대한 속성을 제공합니다.
ITStaticAudioTerminal ITStaticAudioTerminal 인터페이스는 TAPI 3.1 MSP가 모든 정적 오디오 터미널에 노출해야 하는 인터페이스입니다. 인터페이스는 휴대폰 디바이스를 지원하는 데 필요한 정적 오디오 터미널 개체에 대한 메서드를 정의합니다.
ITStream ITStream 인터페이스는 애플리케이션이 스트림에 대한 정보를 검색할 수 있도록 하는 메서드를 노출합니다. 스트림을 시작, 일시 중지 또는 중지하려면 스트림에서 터미널을 선택하거나 선택 취소하려면 스트림에서 선택한 터미널 목록을 가져옵니다.

ITStreamControl ITStreamControl 인터페이스는 호출의 미디어 스트리밍 기능을 나타내며 애플리케이션이 스트림을 열거, 생성 또는 제거할 수 있도록 하는 메서드를 노출합니다.
ITSubStream ITSubStream은 ITStream의 구성 요소이며, 애플리케이션에서 미디어 스트리밍을 더 세밀하게 제어할 수 있습니다.
ITSubStreamControl ITSubStreamControl 인터페이스는 애플리케이션이 하위 스트림을 열거, 만들기 또는 제거할 수 있도록 하는 메서드를 노출합니다. 많은 MSP가 이 인터페이스를 지원하지 않습니다.
ITTAPI ITTAPI 인터페이스는 TAPI 개체의 기본 인터페이스입니다. TAPI 개체는 CoCreateInstance에 의해 만들어집니다. CoCreateInstance에 대한 자세한 내용은 COM의 설명서를 참조하세요. 다른 모든 TAPI 3 개체는 TAPI 3 자체에서 만들어집니다.
ITTAPI2 ITTAPI2 인터페이스는 ITTAPI 인터페이스에서 파생됩니다. 휴대폰 디바이스를 지원하기 위해 TAPI 개체에 추가 메서드를 추가합니다.
ITTAPIEventNotification ITTAPIEventNotification 인터페이스는 애플리케이션이 이벤트 정보의 처리를 제어할 수 있도록 하는 나가는 인터페이스입니다.
ITTAPIObjectEvent ITTAPIObjectEvent 인터페이스에는 TAPI 개체 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITTAPIObjectEvent2 ITTAPIObjectEvent2 인터페이스는 ITTAPIObjectEvent 인터페이스의 확장입니다. ITTAPIObjectEvent2는 TAPI 개체 이벤트를 발생시킨 전화 개체의 ITPhone 인터페이스에 대한 포인터를 반환하는 추가 메서드를 노출합니다.
ITTerminal ITTerminal 인터페이스는 터미널 개체의 기본 인터페이스입니다.

ITTerminalSupport ITTerminalSupport 인터페이스는 MSP가 있는 경우에만 Address 개체에 노출됩니다. 이 인터페이스의 메서드를 사용하면 애플리케이션이 사용 가능한 터미널을 검색하고/또는 터미널을 만들고 필요한 터미널 개체에 대한 포인터를 가져올 수 있습니다.
ITTerminalSupport2 ITTerminalSupport2 인터페이스는 ITTerminalSupport 인터페이스에서 파생됩니다. ITTerminalSupport2는 C, C++, 스크립팅 애플리케이션을 통해 플러그형 터미널 클래스 및 슈퍼클래스에 대한 정보 검색을 지원합니다.
ITToneDetectionEvent ITToneDetectionEvent 인터페이스는 애플리케이션이 톤 감지 이벤트에 대한 정보를 검색할 수 있도록 하는 메서드를 노출합니다.
ITToneTerminalEvent ITToneTerminalEvent 인터페이스에는 발생한 톤 터미널 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.
ITTTSTerminalEvent ITTTSTerminalEvent 인터페이스에는 발생한 TTS(Text to Speech) 터미널 이벤트에 대한 설명을 검색하는 메서드가 포함되어 있습니다.

구조체

TAPI_CUSTOMTONE TAPI_CUSTOMTONE 구조에는 사용자 지정 톤을 정의하는 매개 변수가 포함되어 있습니다.
TAPI_DETECTTONE TAPI_DETECTTONE 구조는 모니터링할 톤을 설명합니다. 이 항목은 배열의 항목으로 사용됩니다.

열거형

ADDRESS_CAPABILITY ADDRESS_CAPABILITY 열거형의 멤버는 ITAddressCapabilities::get_AddressCapability 메서드에서 필요한 주소 기능을 나타내는 데 사용됩니다.
ADDRESS_CAPABILITY_STRING ADDRESS_CAPABILITY_STRING 열거형은 문자열로 설명된 주소 기능을 확인하는 데 사용됩니다.
ADDRESS_EVENT ADDRESS_EVENT 열거형은 주소 이벤트를 설명합니다. ITAddressEvent::get_Event 메서드는 발생한 주소 이벤트의 유형을 나타내기 위해 이 열거형의 멤버를 반환합니다.
ADDRESS_STATE ADDRESS_STATE 열거형은 ITAddress::get_State 메서드에서 주소 상태를 확인하는 데 사용됩니다.
CALL_MEDIA_EVENT CALL_MEDIA_EVENT 열거형은 통화 미디어 이벤트를 설명합니다. ITCallMediaEvent::get_Event 메서드는 발생한 호출 미디어 이벤트의 유형을 나타내기 위해 이 열거형의 멤버를 반환합니다.
CALL_MEDIA_EVENT_CAUSE CALL_MEDIA_EVENT_CAUSE 열거형은 ITCallMediaEvent::get_Cause 메서드에서 디바이스 시간 제한과 같은 미디어 이벤트의 원인에 대한 설명을 반환하는 데 사용됩니다.
CALL_NOTIFICATION_EVENT CALL_NOTIFICATION_EVENT 열거형은 통화 알림 이벤트를 설명합니다. ITCallNotificationEvent::get_Event 메서드는 발생한 호출 알림 이벤트의 유형을 나타내기 위해 이 열거형의 멤버를 반환합니다.
CALL_PRIVILEGE CALL_PRIVILEGE 멤버는 ITCallInfo::get_Privilege 메서드에 의해 반환되며 현재 애플리케이션이 현재 호출을 소유하거나 모니터링하는 시기를 나타냅니다.
CALL_STATE CALL_STATE 열거형은 ITCallInfo::get_CallState 및 ITCallStateEvent::get_State 메서드에서 사용됩니다.
CALL_STATE_EVENT_CAUSE CALL_STATE_EVENT_CAUSE 열거형은 ITCallStateEvent::get_Cause 메서드에 의해 반환됩니다.

CALLHUB_EVENT CALLHUB_EVENT 열거형은 CallHub 이벤트를 설명합니다. ITCallHubEvent::get_Event 메서드는 이 열거형의 멤버를 반환하여 발생한 CallHub 이벤트의 유형을 나타냅니다.
CALLHUB_STATE CALLHUB_STATE 열거형은 ITCallHub::get_State 메서드에서 반환하는 상태 표시기입니다.
CALLINFO_BUFFER CALLINFO_BUFFER 열거형은 ITCallInfo::GetCallInfoBuffer 메서드 또는 ITCallInfo::SetCallInfoBuffer 메서드에서 액세스하는 버퍼 유형을 나타냅니다.
CALLINFO_LONG CALLINFO_LONG 열거형은 LONG 형식의 호출 정보를 설정하고 가져오는 ITCallInfo 메서드에서 사용됩니다.
CALLINFO_STRING CALLINFO_STRING 열거형은 문자열 사용과 관련된 호출 정보를 설정하고 가져오는 ITCallInfo 메서드에서 사용됩니다.
CALLINFOCHANGE_CAUSE CALLINFOCHANGE_CAUSE 열거형은 ITCallInfoChangeEvent::get_Cause 메서드에서 변경된 호출 정보의 형식에 대한 설명을 반환하는 데 사용됩니다.
DISCONNECT_CODE DISCONNECT_CODE 열거형은 ITBasicCallControl::D isconnect 메서드에서 사용됩니다.
FINISH_MODE FINISH_MODE 열거형은 애플리케이션에서 필요한 호출 완료 유형을 나타내는 데 사용됩니다. TAPI DLL이 수행하는 작업은 통화 전송이 완료되고 있는지 또는 전화 회의에 통화가 추가되는지에 따라 달라집니다.
FT_STATE_EVENT_CAUSE FT_STATE_EVENT_CAUSE 열거형은 파일 터미널 이벤트의 형식을 나타냅니다.
FULLDUPLEX_SUPPORT FULLDUPLEX_SUPPORT 열거형은 지정된 터미널이 전체 이중 작업을 지원하는지 여부를 나타내기 위해 레거시 TSP와 상호 작용하는 애플리케이션에서 사용됩니다. 이 열거형은 ITLegacyWaveSupport::IsFullDuplex 메서드에 의해 반환됩니다.

PHONE_BUTTON_FUNCTION PHONE_BUTTON_FUNCTION 열거형은 단추의 함수에 대한 자세한 정보를 제공합니다.
PHONE_BUTTON_MODE PHONE_BUTTON_MODE 열거형은 전화 단추의 모드를 설명합니다.
PHONE_BUTTON_STATE PHONE_BUTTON_STATE 열거형은 전화 단추의 상태를 설명합니다.
PHONE_EVENT PHONE_EVENT 열거형은 전화 이벤트의 유형을 나타냅니다.
PHONE_HOOK_SWITCH_DEVICE PHONE_HOOK_SWITCH_DEVICE 열거형은 휴대폰 디바이스의 스위치 후크 유형을 나타내는 데 사용됩니다.
PHONE_HOOK_SWITCH_STATE PHONE_HOOK_SWITCH_STATE 열거형은 전화 후크위치 상태의 표시기를 제공합니다.
PHONE_LAMP_MODE PHONE_LAMP_MODE 열거형은 전화 램프의 상태를 표시합니다.
PHONE_PRIVILEGE PHONE_PRIVILEGE 열거형은 현재 휴대폰 디바이스와 관련하여 애플리케이션의 권한 상태를 나타냅니다.
PHONE_TONE PHONE_TONE 열거형은 전화 톤을 식별합니다.
PHONECAPS_BUFFER PHONECAPS_BUFFER 열거형은 버퍼에서 설명하는 전화 기능을 설정하거나 가져오는 메서드에서 사용됩니다.
PHONECAPS_LONG PHONECAPS_LONG 열거형은 긴 값으로 설명된 전화 기능을 설정하거나 가져오는 메서드에서 사용됩니다.

PHONECAPS_STRING PHONECAPS_STRING 열거형은 문자열에서 설명하는 전화 기능을 설정하거나 가져오는 메서드에서 사용됩니다.
QOS_EVENT QOS_EVENT 열거형은 QOS(서비스 품질) 이벤트를 설명합니다. ITQOSEvent::get_Event 메서드는 이 열거형의 멤버를 반환하여 발생한 QOS 이벤트의 유형을 나타냅니다.
QOS_SERVICE_LEVEL QOS_SERVICE_LEVEL 열거형은 ITBasicCallControl::SetQOS 메서드에서 호출에 대한 서비스 품질 요구 사항을 나타내는 데 사용됩니다.
TAPI_EVENT TAPI 개체에서 변경이 발생했음을 애플리케이션에 알리는 데 사용됩니다.
TAPI_GATHERTERM TAPI_GATHERTERM 열거형은 TAPI 서버가 호출에서 숫자 수집을 종료한 이유를 설명하는 데 사용됩니다.
TAPI_OBJECT_TYPE TAPI_OBJECT_TYPE 열거형은 이벤트에 관련된 TAPI 개체의 형식을 나타내는 데 사용됩니다.
TAPI_TONEMODE TAPI_TONEMODE 열거형은 선 톤을 생성할 때 사용되는 다양한 선택을 설명하는 데 사용됩니다.
TAPIOBJECT_EVENT TAPIOBJECT_EVENT 열거형은 TAPI 개체 이벤트를 설명합니다. ITTAPIObjectEvent::get_Event 메서드는 이 열거형의 멤버를 반환하여 발생한 TAPI 개체 이벤트의 형식을 나타냅니다.
TERMINAL_DIRECTION TERMINAL_DIRECTION 열거형은 로컬 컴퓨터 또는 터미널의 방향 기능과 관련하여 미디어 스트림의 방향을 설명하는 데 사용됩니다.
TERMINAL_MEDIA_STATE TERMINAL_MEDIA_STATE 열거형은 파일 터미널의 상태를 나타냅니다.

TERMINAL_STATE TERMINAL_STATE 열거형은 터미널 디바이스의 현재 상태를 설명합니다. 이 열거형은 ITTerminal::get_State 메서드에서 반환됩니다.
TERMINAL_TYPE TERMINAL_TYPE 열거형은 터미널의 형식을 설명합니다. 이 열거형은 ITTerminal::get_TerminalType 메서드에서 반환됩니다.

taskschd.h 헤더

아티클2023. 08. 26.

이 헤더는 작업 스케줄러에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 작업 Scheduler

taskschd.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAction 모든 작업 개체에서 상속되는 공통 속성을 제공합니다.
IActionCollection 태스크에서 수행하는 작업을 포함합니다.
IBootTrigger 시스템이 시작될 때 작업을 시작하는 트리거를 나타냅니다.
IComHandlerAction 처리기를 발생시키는 작업을 나타냅니다.
IDailyTrigger 일별 일정에 따라 작업을 시작하는 트리거를 나타냅니다.
IEmailAction 전자 메일 메시지를 보내는 작업을 나타냅니다.
IEventTrigger 시스템 이벤트가 발생할 때 작업을 시작하는 트리거를 나타냅니다.
IExecAction 명령줄 작업을 실행하는 작업을 나타냅니다.
IIdleSettings

컴퓨터가 유휴 상태일 때 작업 스케줄러가 작업을 수행하는 방법을 지정합니다.
IIdleTrigger 컴퓨터가 유휴 상태로 전환되면 작업을 시작하는 트리거를 나타냅니다.
ILogonTrigger 사용자가 로그온할 때 작업을 시작하는 트리거를 나타냅니다.
IMaintenanceSettings 자동 유지 관리 중에 작업 스케줄러가 작업을 수행하는 데 사용하는 설정을 제공합니다.
IMonthlyDOWTrigger 월별 요일 일정에 따라 작업을 시작하는 트리거를 나타냅니다.
IMonthlyTrigger 월별 일정에 따라 작업을 시작하는 트리거를 나타냅니다.
INetworkSettings 작업 스케줄러 서비스에서 네트워크 프로필을 가져오는 데 사용하는 설정을 제공합니다.
IPrincipal 보안 주체에 대한 보안 자격 증명을 제공합니다.
IPrincipal2 보안 자격 증명에 적용된 확장 설정을 보안 주체에 제공합니다.
IRegisteredTask 작업을 즉시 실행하고, 태스크의 실행 중인 인스턴스를 가져와서 작업을 등록하는 데 사용되는 자격 증명과 작업을 설명하는 속성을 얻거나 설정하는 데 사용되는 메서드를 제공합니다.
IRegisteredTaskCollection 등록된 모든 작업을 포함합니다.
IRegistrationInfo 작업을 설명하는 데 사용할 수 있는 관리 정보를 제공합니다.
IRegistrationTrigger 작업이 등록되거나 업데이트될 때 작업을 시작하는 트리거를 나타냅니다.

IRepetitionPattern 작업이 실행되는 빈도와 작업이 시작된 후 반복 패턴이 반복되는 기간을 정의합니다.
IRunningTask 에서 정보를 얻고 실행 중인 작업을 제어하는 메서드를 제공합니다.
IRunningTaskCollection 실행 중인 작업을 제어하는 데 사용되는 컬렉션을 제공합니다.
ISessionStateChangeTrigger 콘솔 연결 또는 연결 끊기, 원격 연결 또는 연결 끊기 또는 워크스테이션 잠금 또는 잠금 해제 알림에 대한 작업을 트리거합니다.
IShowMessageAction 작업이 활성화될 때 메시지 상자를 표시하는 작업을 나타냅니다.
ITaskDefinition 작업 설정, 트리거, 작업 및 등록 정보와 같은 작업의 모든 구성 요소를 정의합니다.
ITaskFolder 폴더에 작업을 등록(만들기)하고, 폴더에서 작업을 제거하고, 폴더에서 하위 폴더를 만들거나 제거하는 데 사용되는 메서드를 제공합니다.
ITaskFolderCollection 작업을 포함하는 폴더 컬렉션에 대한 정보 및 제어를 제공합니다.
ITaskHandler COM 처리기를 관리하기 위해 작업 스케줄러 서비스에서 호출하는 메서드를 정의합니다.
ITaskHandlerStatus COM 처리기에서 작업 스케줄러에 처리기의 상태 알리는 데 사용되는 메서드를 제공합니다.
ITaskNamedValueCollection ITaskNamedValuePair 인터페이스 이름-값 쌍의 컬렉션을 포함합니다.
ITaskNamedValuePair 이름이 값과 연결된 이름-값 쌍을 만듭니다.

ITaskService 등록된 작업을 관리하기 위한 작업 스케줄러 서비스에 대한 액세스를 제공합니다.
ITaskSettings 작업 스케줄러 서비스에서 작업을 수행하는 데 사용하는 설정을 제공합니다.
ITaskSettings2 작업 스케줄러가 작업을 실행하는 데 사용하는 확장 설정을 제공합니다. (ITaskSettings2)
ITaskSettings3 작업 스케줄러가 작업을 실행하는 데 사용하는 확장 설정을 제공합니다. (ITaskSettings3)
ITaskVariables 태스크에서 시작하는 작업 처리기 및 외부 실행 파일에 매개 변수로 전달할 수 있는 작업 변수를 정의합니다.
ITimeTrigger 특정 날짜 및 시간에 작업을 시작하는 트리거를 나타냅니다.
ITrigger 모든 트리거 개체에서 상속되는 공통 속성을 제공합니다.
ITriggerCollection 작업의 트리거를 추가, 제거 및 가져오는 데 사용되는 메서드를 제공합니다.
IWeeklyTrigger 주별 일정에 따라 작업을 시작하는 트리거를 나타냅니다.

열거형

TASK_ACTION_TYPE 태스크가 수행할 수 있는 작업 유형을 정의합니다.
TASK_COMPATIBILITY

작업 스케줄러 또는 태스크와 호환되는 AT 명령의 버전을 정의합니다.
TASK_CREATION 작업 스케줄러 서비스가 작업을 만들거나 업데이트하거나 사용하지 않도록 설정하는 방법을 정의합니다.
TASK_ENUM_FLAGS 작업 스케줄러가 등록된 작업을 통해 열거하는 방법을 정의합니다.
TASK_INSTANCES_POLICY 작업 스케줄러가 작업의 새 instance 시작할 때 작업의 기존 인스턴스를 처리하는 방법을 정의합니다.
TASK_LOGON_TYPE 작업을 실행하는 데 필요한 로그온 기술을 정의합니다.
TASK_PROCESSTOKENSID_TYPE 태스크에서 사용할 수 있는 SID(프로세스 보안 식별자) 유형을 정의합니다.
TASK_RUN_FLAGS 작업을 실행하는 방법을 정의합니다.
TASK_RUNLEVEL_TYPE 작업이 실행될 권한 수준으로 지정하는 LUA 권한 상승 플래그를 정의합니다.
TASK_SESSION_STATE_CHANGE_TYPE 시작할 작업을 트리거하는 데 사용할 수 있는 터미널 서버 세션 상태 변경 종류를 정의합니다.
TASK_STATE 등록된 작업이 있을 수 있는 다양한 상태를 정의합니다.
TASK_TRIGGER_TYPE2 태스크에서 사용할 수 있는 트리거 유형을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

tbs.h 헤더

아티클2023. 08. 26.

이 헤더는 TPM 기본 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- TPM 기본 서비스

tbs.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

Tbsi_Context_Create TBS에 명령을 전달하는 데 사용할 수 있는 컨텍스트 핸들을 만듭니다.
Tbsi_Get_OwnerAuth 로컬 레지스트리에서 정보를 사용할 수 있는 경우 TPM의 소유자 권한 부여를 검색합니다.
Tbsi_Get_TCG_Log TCG 로그라고도 하는 최신 WBCL(Windows 부팅 구성 로그)을 검색합니다.
Tbsi_Get_TCG_Log_Ex 지정된 형식의 TCG 로그라고도 하는 WBCL(Windows 부팅 구성 로그)을 가져옵니다.
Tbsi_GetDeviceInfo 컴퓨터에서 TPM의 버전을 가져옵니다.
Tbsi_Physical_Presence_Command TBS를 통해 드라이버에 물리적 프레즌스 ACPI 명령을 전달합니다.
Tbsi_Revoke_Attestation ELAM 드라이버가 정책 위반(예: 루트킷)을 검색하는 경우 PCR을 무효화합니다.
Tbsip_Cancel_Commands 지정된 컨텍스트에 대한 모든 미해결 명령을 취소합니다.
Tbsip_Context_Close

컨텍스트 핸들을 닫습니다. 이 핸들은 TBS의 컨텍스트와 연결된 리소스를 해제하고 TBS와 통신하는 데 사용되는 바인딩 핸들을 닫습니다.
Tbsip_Submit_Command 처리를 위해 TPM(신뢰할 수 있는 플랫폼 모듈) 명령을 TBS(TPM 기본 서비스)에 제출합니다.

구조체

TBS_CONTEXT_PARAMS TBS 컨텍스트 구현의 버전을 지정합니다.
TBS_CONTEXT_PARAMS2 TBS 컨텍스트 구현의 버전을 지정합니다. 애플리케이션이 두 버전의 TPM에서 모두 작동하는 경우 이 구조를 사용해야 합니다.
TPM_DEVICE_INFO TPM 버전에 대한 정보를 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

tcpestats.h 헤더

아티클2023. 08. 26.

이 헤더는 IP 도우미에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- IP 도우미

tcpestats.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

TCP_ESTATS_BANDWIDTH_ROD_v0 TCP 연결에 대한 대역폭 추정에 대한 확장된 TCP 통계에 대한 읽기 전용 동적 정보를 포함합니다.
TCP_ESTATS_BANDWIDTH_RW_v0 TCP 연결에 대한 대역폭 추정에 대한 확장된 TCP 통계에 대한 읽기/쓰기 구성 정보를 포함합니다.
TCP_ESTATS_DATA_ROD_v0 TCP 연결에 대한 데이터 전송에 대한 확장된 TCP 통계에 대한 읽기 전용 동적 정보를 포함합니다.
TCP_ESTATS_DATA_RW_v0 TCP 연결에 대한 데이터 전송에 대한 확장된 TCP 통계에 대한 읽기/쓰기 구성 정보를 포함합니다.
TCP_ESTATS_FINE_RTT_ROD_v0 TCP 연결에 대한 RTT(세분화된 왕복 시간) 추정에 대한 확장된 TCP 통계에 대한 읽기 전용 동적 정보를 포함합니다.
TCP_ESTATS_FINE_RTT_RW_v0 TCP 연결에 대한 RTT(세분화된 왕복 시간) 추정 통계에 대한 확장된 TCP 통계에 대한 읽기/쓰기 구성 정보를 포함합니다.
TCP_ESTATS_OBS_REC_ROD_v0 TCP 연결에 대한 원격 수신기에서 관찰된 확장된 TCP 통계에 대한 읽기 전용 동적 정보를 포함합니다.
TCP_ESTATS_OBS_REC_RW_v0

TCP 연결에 대한 원격 수신기에서 관찰된 확장된 TCP 통계에 대한 읽기/쓰기 구성 정보를 포함합니다.
TCP_ESTATS_PATH_ROD_v0 TCP 연결에 대한 네트워크 경로 측정에 대한 확장된 TCP 통계에 대한 읽기 전용 동적 정보를 포함합니다.
TCP_ESTATS_PATH_RW_v0 TCP 연결의 경로 측정에 대한 확장된 TCP 통계에 대한 읽기/쓰기 구성 정보를 포함합니다.
TCP_ESTATS_REC_ROD_v0 TCP 연결에 대한 로컬 수신기의 확장된 TCP 통계에 대한 읽기 전용 동적 정보를 포함합니다.
TCP_ESTATS_REC_RW_v0 TCP 연결에 대한 로컬 수신기의 확장된 TCP 통계에 대한 읽기/쓰기 구성 정보를 포함합니다.
TCP_ESTATS_SEND_BUFF_ROD_v0 TCP 연결에 대한 출력 큐에 대한 확장된 TCP 통계에 대한 읽기 전용 동적 정보를 포함합니다.
TCP_ESTATS_SEND_BUFF_RW_v0 TCP 연결에 대한 출력 큐에 대한 확장된 TCP 통계에 대한 읽기/쓰기 구성 정보를 포함합니다.
TCP_ESTATS_SND_CONG_ROD_v0 TCP 연결에 대한 보낸 사람 정체 관련 데이터에 대한 확장된 TCP 통계에 대한 읽기 전용 동적 정보를 포함합니다.
TCP_ESTATS_SND_CONG_ROS_v0 TCP 연결의 최대 정체 기간에 대한 확장된 TCP 통계에 대한 읽기 전용 정적 정보를 포함합니다.
TCP_ESTATS_SND_CONG_RW_v0 TCP 연결에 대한 보낸 사람 정체에 대한 확장된 TCP 통계에 대한 읽기/쓰기 구성 정보를 포함합니다.
TCP_ESTATS_SYN_OPTS_ROS_v0 TCP 연결에 대한 SYN 교환의 확장된 TCP 통계에 대한 읽기 전용 정적 정보를 포함합니다.

열거형

TCP_BOOLEAN_OPTIONAL TCP 연결에 대한 읽기/쓰기 정보의 멤버를 업데이트할 때 호출자가 지정할 수 있는 상태를 정의합니다.
TCP_ESTATS_TYPE 요청되거나 설정되는 TCP 연결에 대한 확장 통계 유형을 정의합니다.
TCP_SOFT_ERROR TCP 연결에 기록된 치명적이거나 일시적이지 않은 오류의 이유를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

tcpioctl.h 헤더

아티클2023. 08. 26.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

tcpioctl.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

IOCTLs

IOCTL_TCP_QUERY_INFORMATION_EX TCP/IP 드라이버에서 정보를 검색합니다.

구조체

IFEntry 16~23페이지의 RFC(Internet Engineering Task Force) 1213에 지정된 대로 TCP/IP 네트워크(MIB-II)의 네트워크 관리를 위한 관리 정보 기반의 단순 네트워크 관리 프로토콜 버전 2(SNMPv2) 표준에 따라 TCP/IP 인터페이스를 설명합니다.
IPAddrEntry IETF(인터넷 엔지니어링 태스크 포스) RFC(주석 요청) 2011에 지정된 대로 IP(인터넷 프로토콜)에 대한 MIB-II(관리 정보 베이스) 정보 그룹의 일부를 구현합니다. (IPAddrEntry)
IPInterfaceInfo IPInterfaceInfo 구조에는 IOCTL_TCP_QUERY_INFORMATION_EX 제어 코드에서 반환되는 특정 IP 인터페이스에 대한 정보가 포함되어 있습니다.
IPSNMPInfo IETF(인터넷 엔지니어링 태스크 포스) RFC(주석 요청) 2011에 지정된 대로 IP(인터넷 프로토콜)에 대한 MIB-II(관리 정보 베이스) 정보 그룹의 일부를 구현합니다. (IPSNMPInfo)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

tcpmib.h 헤더

아티클2023. 08. 26.

이 헤더는 관리 정보 베이스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 관리 정보 기반

tcpmib.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

MIB_TCP6ROW IPv6 TCP 연결을 설명하는 정보를 포함합니다. (MIB_TCP6ROW)
MIB_TCP6ROW_OWNER_MODULE 소유권 데이터가 있는 PID(특정 프로세스 ID)에 바인딩된 IPv6 TCP 연결을 설명하는 정보를 포함합니다.
MIB_TCP6ROW_OWNER_PID 특정 프로세스 ID(PID)와 연결된 IPv6 TCP 연결을 설명하는 정보를 포함합니다.
MIB_TCP6ROW2 IPv6 TCP 연결을 설명하는 정보를 포함합니다. (MIB_TCP6ROW2)
MIB_TCP6TABLE 로컬 컴퓨터의 IPv6에 대한 TCP 연결 테이블을 포함합니다.
MIB_TCP6TABLE_OWNER_MODULE 사용 가능한 소유권 데이터와 함께 이러한 PID에 바인딩된 프로세스 ID(PID) 테이블과 IPv6 TCP 링크 컨텍스트를 포함합니다.
MIB_TCP6TABLE_OWNER_PID 이러한 PID에 바인딩된 컨텍스트인 프로세스 ID(PID) 및 IPv6 TCP 링크의 테이블이 포함되어 있습니다.
MIB_TCP6TABLE2 로컬 컴퓨터의 IPv6 TCP 연결 테이블을 포함합니다.

MIB_TCPROW_LH MIB_TCPROW_LH(tcpmib.h)에는 IPv4 TCP 연결을 설명하는 정보가 포함되어 있습니다.
MIB_TCPROW_OWNER_MODULE 소유권 데이터, IPv4 주소, TCP 연결에서 사용하는 포트 및 연결과 연결된 특정 프로세스 ID(PID)를 사용하여 IPv4 TCP 연결을 설명하는 정보를 포함합니다.
MIB_TCPROW_OWNER_PID IPv4 주소와의 IPv4 TCP 연결, TCP 연결에서 사용되는 포트 및 연결과 연결된 PID(특정 프로세스 ID)를 설명하는 정보를 포함합니다.
MIB_TCPROW_W2K MIB_TCPROW_W2K(tcpmib.h)에는 IPv4 TCP 연결을 설명하는 정보가 포함되어 있습니다.
MIB_TCPROW2 IPv4 TCP 연결을 설명하는 정보를 포함합니다.
MIB_TCPSTATS_LH MIB_TCPSTATS_LH(tcpmib.h)에는 로컬 컴퓨터에서 실행되는 TCP 프로토콜에 대한 통계가 포함되어 있습니다.
MIB_TCPSTATS_W2K MIB_TCPSTATS_W2K(tcpmib.h)에는 로컬 컴퓨터에서 실행되는 TCP 프로토콜에 대한 통계가 포함되어 있습니다.
MIB_TCPSTATS2 로컬 컴퓨터에서 실행되는 TCP 프로토콜에 대한 통계를 포함합니다. (MIB_TCPSTATS2)
MIB_TCPTABLE 로컬 컴퓨터의 IPv4에 대한 TCP 연결 테이블이 포함되어 있습니다.
MIB_TCPTABLE_OWNER_MODULE PID(프로세스 ID) 테이블과 PID에 바인딩된 IPv4 TCP 링크 컨텍스트 및 사용 가능한 모든 소유권 데이터를 포함합니다.
MIB_TCPTABLE_OWNER_PID 이러한 PID에 바인딩된 컨텍스트인 프로세스 ID(PID) 및 IPv4 TCP 링크의 테이블이 포함되어 있습니다.

MIB_TCPTABLE2 로컬 컴퓨터의 IPv4 TCP 연결 테이블을 포함합니다.

열거형

TCP_CONNECTION_OFFLOAD_STATE TCP 연결에 사용할 수 있는 TCP 오프로드 상태를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

tdh.h 헤더

아티클2023. 03. 08.

이 TDH(추적 데이터 도우미) 헤더는 이벤트 추적에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 이벤트 추적

tdh.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

EMI_MAP_FORMAT 이벤트 맵 형식을 검색하는 매크로입니다.
EMI_MAP_INPUT 이벤트 맵 입력을 검색하는 매크로입니다.
EMI_MAP_NAME 이벤트 맵 이름을 검색하는 매크로입니다.
EMI_MAP_OUTPUT 이벤트 맵 출력을 검색하는 매크로입니다.
PEI_PROVIDER_NAME PEI(공급자 이벤트 정보) 이름을 검색하는 매크로입니다.
PFI_FIELD_MESSAGE PFI(공급자 필드 정보) 필드 메시지를 검색하는 매크로입니다.
PFI_FIELD_NAME PFI(공급자 필드 정보) 필드 이름을 검색하는 매크로입니다.
PFI_FILTER_MESSAGE PFI(공급자 필드 정보) 필드 메시지를 필터링하는 매크로입니다.

PFI_PROPERTY_NAME PFI(공급자 필드 정보) 속성 이름을 검색하는 매크로입니다.
TdhAggregatePayloadFilters EnableTraceEx2 함수와 함께 사용할 단일 데이터 구조로 단일 공급자에 대한 여러 페이로드 필터를 집계합니다.
TdhCleanupPayloadEventFilterDescriptor TdhAggregatePayloadFilters 함수를 사용하여 만든 페이로드 필터의 집계된 구조를 해제합니다.
TdhCloseDecodingHandle 입력 디코딩 핸들과 연결된 모든 리소스를 해제합니다.
TdhCreatePayloadFilter EnableTraceEx2 함수와 함께 사용할 단일 페이로드에 대한 단일 필터를 만듭니다.
TdhDeletePayloadFilter TdhCreatePayloadFilter 함수에 의해 단일 페이로드 필터에 할당된 메모리를 해제합니다.
TdhEnumerateManifestProviderEvents 공급자 매니페스트에 있는 이벤트 목록을 검색합니다.
TdhEnumerateProviderFieldInformation 지정된 공급자에 대해 지정된 필드 메타데이터를 검색합니다.
TdhEnumerateProviderFilters 매니페스트에 정의된 지정된 공급자가 지정한 필터를 열거합니다.
TdhEnumerateProviders 컴퓨터에 MOF 클래스 또는 매니페스트 파일을 등록한 공급자 목록을 검색합니다.
TdhEnumerateProvidersForDecodingSource 컴퓨터에 MOF 클래스 또는 매니페스트 파일을 등록한 공급자 목록을 검색합니다.
TdhFormatProperty 표시할 속성 값의 형식을 지정합니다.

TdhGetDecodingParameter 디코딩 매개 변수의 값을 검색합니다.
TdhGetEventInformation 이벤트에 대한 메타데이터를 검색합니다.
TdhGetEventMapInformation 이벤트에 포함된 이벤트 맵에 대한 정보를 검색합니다.
TdhGetManifestEventInformation 매니페스트의 이벤트에 대한 메타데이터를 검색합니다.
TdhGetProperty 이벤트 데이터에서 속성 값을 검색합니다.
TdhGetPropertySize 이벤트 데이터에서 하나 이상의 속성 값의 크기를 검색합니다.
TdhGetWppMessage EVENT_RECORD 구조에 포함된 형식의 WPP 메시지를 검색합니다.
TdhGetWppProperty WPP 메시지와 연결된 특정 속성을 검색합니다.
TdhLoadManifest 로그 파일을 디코딩하는 데 사용되는 매니페스트를 로드합니다.
TdhLoadManifestFromBinary NULL로 끝나는 경로를 특정 이벤트 공급자를 디코딩하는 데 필요한 메타데이터 리소스가 포함된 이진 파일로 이동합니다.
TdhLoadManifestFromMemory 메모리에서 매니페스트를 로드합니다.
TdhOpenDecodingHandle 디코딩 핸들을 엽니다.

TdhQueryProviderFieldInformation 지정된 값과 일치하는 필드 값에 대한 이벤트 설명에서 지정된 필드에 대한 정보를 검색합니다.
TdhSetDecodingParameter 디코딩 매개 변수의 값을 설정합니다.
TdhUnloadManifest TdhLoadManifest 함수에 의해 로드된 매니페스트를 언로드합니다.
TdhUnloadManifestFromMemory 메모리에서 매니페스트를 언로드합니다.
TEI_ACTIVITYID_NAME TEI(추적 이벤트 정보) 활동 ID 이름을 검색하는 매크로입니다.
TEI_CHANNEL_NAME TEI(추적 이벤트 정보) 채널 이름을 검색하는 매크로입니다.
TEI_EVENT_MESSAGE TEI(추적 이벤트 정보) 메시지를 검색하는 매크로입니다.
TEI_KEYWORDS_NAME TEI(추적 이벤트 정보) 키워드 이름을 검색하는 매크로입니다.
TEI_LEVEL_NAME TEI(추적 이벤트 정보) 수준 이름을 검색하는 매크로입니다.
TEI_MAP_NAME TEI(추적 이벤트 정보) 맵 이름을 검색하는 매크로입니다.
TEI_OPCODE_NAME TEI(추적 이벤트 정보) opcode 이름을 검색하는 매크로입니다.
TEI_PROPERTY_NAME TEI(추적 이벤트 정보) 속성 이름을 검색하는 매크로입니다.

TEI_PROVIDER_MESSAGE TEI(추적 이벤트 정보) 공급자 메시지를 검색하는 매크로입니다.
TEI_PROVIDER_NAME TEI(추적 이벤트 정보) 공급자 이름을 검색하는 매크로입니다.
TEI_RELATEDACTIVITYID_NAME TEI(추적 이벤트 정보) 관련 활동 ID 이름을 검색하는 매크로입니다.
TEI_TASK_NAME TEI(추적 이벤트 정보) 작업 이름을 검색하는 매크로입니다.

구조체

EVENT_MAP_ENTRY 단일 값 맵 항목을 정의합니다.
EVENT_MAP_INFO 이벤트 맵에 대한 메타데이터를 정의합니다.
EVENT_PROPERTY_INFO 이벤트 또는 필터의 단일 속성에 대한 정보를 제공합니다.
PAYLOAD_FILTER_PREDICATE 추적 세션의 단일 필드를 필터링하는 방법을 설명하는 이벤트 페이로드 필터 조건자를 정의합니다.
PROPERTY_DATA_DESCRIPTOR 검색할 속성을 정의합니다.
PROVIDER_ENUMERATION_INFO 컴퓨터에 MOF 또는 매니페스트를 등록한 공급자의 배열을 정의합니다.
PROVIDER_EVENT_INFO 공급자 매니페스트의 이벤트 배열을 정의합니다.

PROVIDER_FIELD_INFO 필드 정보를 정의합니다.
PROVIDER_FIELD_INFOARRAY 요청된 필드에 대한 메타데이터 정보를 정의합니다.
PROVIDER_FILTER_INFO 필터 및 해당 데이터를 정의합니다.
TDH_CONTEXT 이벤트를 구문 분석하는 데 필요한 추가 정보를 정의합니다.
TRACE_EVENT_INFO 이벤트에 대한 정보를 정의합니다.
TRACE_PROVIDER_INFO 공급자의 GUID 및 이름을 정의합니다.

열거형

_TDH_IN_TYPE 추적 데이터 도우미(TDH)에 대해 지원되는 [in] 형식을 정의합니다.
_TDH_OUT_TYPE 추적 데이터 도우미(TDH)에 대해 지원되는 [out] 형식을 정의합니다.
DECODING_SOURCE 이벤트 데이터의 원본을 정의합니다.
EVENT_FIELD_TYPE 검색할 공급자 정보를 정의합니다.
MAP_FLAGS 맵이 값 맵, 비트맵 또는 패턴 맵인지 여부를 나타내는 상수 값을 정의합니다.

MAP_VALUETYPE 값 맵 값이 ULONG 데이터 형식 또는 문자열에 있는지를 정의합니다.
PAYLOAD_OPERATOR TDH(추적 데이터 도우미)에 대해 지원되는 페이로드 연산자를 정의합니다.
PROPERTY_FLAGS 속성이 구조체 또는 배열에 포함되어 있는지 여부를 정의합니다.
TDH_CONTEXT_TYPE 컨텍스트 형식을 정의합니다.
TEMPLATE_FLAGS 이벤트 데이터의 레이아웃을 나타내는 상수 값을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

tdiinfo.h 헤더

아티클2023. 08. 26.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

tdiinfo.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

TCP_REQUEST_QUERY_INFORMATION_EX_W2K TCP_REQUEST_QUERY_INFORMATION_EX_W2K 구조체(tdiinfo.h)는 IOCTL_TCP_QUERY_INFORMATION_EX 제어 코드에 대한 입력을 포함합니다.
TCP_REQUEST_QUERY_INFORMATION_EX_XP TCP_REQUEST_QUERY_INFORMATION_EX_XP 구조체(tdiinfo.h)에는 IOCTL_TCP_QUERY_INFORMATION_EX 제어 코드에 대한 입력이 포함됩니다.
TDIEntityID IOCTL_TCP_QUERY_INFORMATION_EX 제어 코드를 사용하여 검색된 TDI 드라이버에 대한 정보를 나타내는 TDIObjectID 구조체의 일부를 포함합니다.
TDIObjectID TCP 드라이버에서 요청되는 정보의 종류를 지정하기 위해 IOCTL_TCP_QUERY_INFORMATION_EX 제어 코드와 함께 사용되는 TCP_REQUEST_QUERY_INFORMATION_EX 구조체의 일부를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

termmgr.h 헤더

아티클2023. 08. 26.

이 헤더는 전화 통신 애플리케이션 프로그래밍 인터페이스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 전화 통신 애플리케이션 프로그래밍 인터페이스

termmgr.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ITPluggableTerminalClassRegistration ITPluggableTerminalClassRegistration 인터페이스는 플러그형 터미널에 대한 레지스트리 항목을 만들고 수정하고 삭제할 수 있는 메서드를 노출합니다.
ITPluggableTerminalInitialization ITPluggableTerminalInitialization 인터페이스는 플러그형 터미널에 의해 구현되며 터미널 관리자가 터미널을 초기화할 수 있도록 합니다. ITPluggableTerminalInitialization 인터페이스는 ITTerminal에서 QueryInterface를 호출하여 만듭니다.
ITPluggableTerminalSuperclassRegistration ITPluggableTerminalSuperclassRegistration 인터페이스는 터미널 슈퍼클래스(이름 및 CLSID)에 대한 정보를 가져와서 설정하는 메서드를 노출합니다.
ITTerminalControl ITTerminalControl 인터페이스는 MSP 핸들을 가져와 필터 그래프에서 터미널을 연결 및 연결 해제하고 렌더러를 실행하거나 중지하는 메서드를 제공합니다.
ITTerminalManager ITTerminalManager 인터페이스는 MSP에서 동적 터미널을 만드는 데 사용됩니다.
ITTerminalManager2 ITTerminalManager2 인터페이스는 현재 시스템에 등록된 플러그형 터미널 클래스에 대한 정보를 검색하는 메서드를 노출합니다. ITTerminalManager2는 ITTerminalManager 인터페이스에서 파생됩니다.

열거형

TMGR_DIRECTION TMGR_DIRECTION 열거형은 플러그형 터미널 메서드에서 터미널 방향을 나타내는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

textserv.h 헤더

아티클2023. 08. 26.

이 헤더는 Windows 컨트롤에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 컨트롤

textserv.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

ITextHost ITextHost 인터페이스는 텍스트 호스트 서비스를 가져오기 위해 텍스트 서비스 개체에서 사용됩니다.
ITextHost2 ITextHost2 인터페이스는 ITextHost 인터페이스를 확장합니다.
ITextServices TOM(텍스트 개체 모델)을 확장하여 창 없는 작업에 대한 추가 기능을 제공합니다.
ITextServices2 ITextServices2 인터페이스는 ITextServices 인터페이스를 확장합니다.

인터페이스

IRichEditUiaInformation 창 없는 서식 있는 편집 컨트롤에 대한 Microsoft UI 자동화 접근성 정보를 제공합니다.
IRicheditUiaOverrides 창 없는 서식 있는 편집 컨트롤의 호스트 컨테이너가 컨트롤의 Microsoft UI 자동화 접근성 속성을 재정의할 수 있도록 합니다.
IRicheditWindowlessAccessibility

창 없는 서식 있는 편집 컨트롤의 호스트 컨테이너를 사용하여 컨트롤의 부모에 대한 Microsoft UI 자동화 공급자를 가져올 수 있습니다.

Functions

CreateTextServices CreateTextServices 함수는 텍스트 서비스 개체의 인스턴스를 만듭니다. 텍스트 서비스 개체는 ITextServices 및 TOM(텍스트 개체 모델)을 비롯한 다양한 인터페이스를 지원합니다.
ShutdownTextServices 텍스트 서비스 instance 호스트의 연결을 끊습니다.

구조체

CHANGENOTIFY EN_CHANGE 알림 코드와 연결된 정보를 포함합니다. 창이 없는 서식 있는 편집 컨트롤은 컨트롤의 콘텐츠가 변경되면 호스트 창으로 이 알림을 보냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

textstor.h 헤더

아티클2023. 08. 26.

이 헤더는 Text Services Framework에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 텍스트 서비스 프레임워크 비교

textstor.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAnchor IAnchor 인터페이스는 TSF 관리자에 의해 구현됩니다. Microsoft 활성 접근성의 클라이언트는 IAnchor 앵커 개체를 사용하여 텍스트 스트림 내에서 텍스트 범위를 구분합니다.
ITextStoreACP ITextStoreACP 인터페이스는 애플리케이션에서 구현되며 TSF 관리자가 TSF의 텍스트 스트림 또는 텍스트 저장소를 조작하는 데 사용됩니다.
ITextStoreACP2 ITextStoreACP2 인터페이스는 애플리케이션에서 구현되며 TSF 관리자가 TSF의 텍스트 스트림 또는 텍스트 저장소를 조작하는 데 사용됩니다.
ITextStoreACPSink ITextStoreACPSink 인터페이스는 TSF 관리자에 의해 구현되며 ACP 기반 애플리케이션에서 특정 이벤트가 발생할 때 관리자에게 알리는 데 사용됩니다. 관리자는 ITextStoreACP::AdviseSink를 호출하여 이 조언 싱크를 설치합니다.
ITextStoreAnchor ITextStoreAnchor 인터페이스는 Microsoft Active Accessibility 클라이언트에 의해 구현되며 TSF 관리자가 텍스트 스트림을 조작하는 데 사용됩니다.
ITextStoreAnchorSink ITextStoreAnchorSink 인터페이스는 TSF 관리자에 의해 구현되며 앵커 기반 애플리케이션에서 특정 이벤트가 발생할 때 관리자에게 알리는 데 사용됩니다. 관리자는 ITextStoreAnchor::AdviseSink를 호출하여 이 조언 싱크를 설치합니다.

구조체

TS_ATTRVAL TS_ATTRVAL 구조에는 문서 특성 값이 포함됩니다.
TS_RUNINFO TS_RUNINFO 구조체는 텍스트 실행 데이터의 속성을 지정합니다.
TS_SELECTION_ACP TS_SELECTION_ACP 구조에는 ACP 기반 텍스트 선택 데이터가 포함됩니다.
TS_SELECTION_ANCHOR TS_SELECTION_ANCHOR 구조에는 앵커 기반 텍스트 선택 데이터가 포함됩니다.
TS_SELECTIONSTYLE TS_SELECTIONSTYLE 구조체는 선택 영역의 스타일을 나타냅니다.
TS_STATUS TS_STATUS 구조에는 문서 상태 데이터가 포함됩니다.
TS_TEXTCHANGE TS_TEXTCHANGE 구조에는 텍스트 변경 데이터가 포함됩니다.

열거형

TsActiveSelEnd TsActiveSelEnd 열거형의 요소는 현재 텍스트 저장소 선택 영역의 끝을 지정합니다.
TsGravity TsGravity 열거형의 요소는 IAnchor 개체와 연결된 중력 유형을 지정합니다.
TsLayoutCode TsLayoutCode 열거형의 요소는 ITextStoreACPSink::OnLayoutChange 또는 ITextStoreAnchorSink::OnLayoutChange 알림에서 레이아웃 변경 유형을 지정하는 데 사용됩니다.

TsRunType TsRunType 열거형의 요소는 텍스트 실행이 표시되거나 숨겨지거나 텍스트 실행에 포함된 프라이빗 데이터 형식인지 여부를 지정합니다.
TsShiftDir TsShiftDir 열거형의 요소는 앵커가 이동되는 방향을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

thrdbase.h 헤더

아티클2023. 08. 26.

이 헤더는 Windows Management Instrumentation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Management Instrumentation

thrdbase.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

CThreadBase CThreadBase 클래스는 WMI 공급자 프레임워크에 대한 내부 스레드 안전 메커니즘을 제공하는 기본 클래스입니다. 이 클래스는 내부적으로 호출됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

threadpoolapiset.h 헤더

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- System Services

threadpoolapiset.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

CallbackMayRunLong 콜백이 빠르게 반환되지 않을 수 있음을 나타냅니다.
CancelThreadpoolIo StartThreadpoolIo 함수에서 알림을 취소합니다.
CloseThreadpool 지정된 스레드 풀을 닫습니다.
CloseThreadpoolCleanupGroup 지정된 정리 그룹을 닫습니다.
CloseThreadpoolCleanupGroupMembers 지정된 정리 그룹의 멤버를 해제하고, 모든 콜백 함수가 완료될 때까지 기다린 다음, 선택적으로 처리 중인 콜백 함수를 취소합니다.
CloseThreadpoolIo 지정된 I/O 완성 개체를 해제합니다.
CloseThreadpoolTimer 지정된 타이머 개체를 해제합니다.
CloseThreadpoolWait 지정된 대기 개체를 해제합니다.
CloseThreadpoolWork

지정된 작업 개체를 해제합니다.
CreateThreadpool 콜백을 실행할 새 스레드 풀을 할당합니다.
CreateThreadpoolCleanupGroup 애플리케이션이 하나 이상의 스레드 풀 콜백을 추적하는 데 사용할 수 있는 정리 그룹을 만듭니다.
CreateThreadpoolIo 새 I/O 완성 개체를 만듭니다.
CreateThreadpoolTimer 새 타이머 개체를 만듭니다.
CreateThreadpoolWait 새 대기 개체를 만듭니다.
CreateThreadpoolWork 새 작업 개체를 만듭니다.
DisassociateCurrentThreadFromCallback 현재 실행 중인 콜백 함수와 콜백을 시작한 개체 간의 연결을 제거합니다. 현재 스레드는 더 이상 개체를 대신하여 콜백을 실행하는 것으로 계산되지 않습니다.
FreeLibraryWhenCallbackReturns 현재 콜백이 완료될 때 스레드 풀이 언로드할 DLL을 지정합니다.
IsThreadpoolTimerSet 지정된 타이머 개체가 현재 설정되어 있는지 여부를 확인합니다.
LeaveCriticalSectionWhenCallbackReturns 현재 콜백이 완료될 때 스레드 풀이 해제되는 중요한 섹션을 지정합니다.
QueryThreadpoolStackInformation 지정된 스레드 풀의 스레드에 대한 스택 예약 및 커밋 크기를 검색합니다.
ReleaseMutexWhenCallbackReturns 현재 콜백이 완료될 때 스레드 풀이 해제될 뮤텍스를 지정합니다.

ReleaseSemaphoreWhenCallbackReturns 현재 콜백이 완료될 때 스레드 풀이 해제될 세마포를 지정합니다.
SetEventWhenCallbackReturns 현재 콜백이 완료될 때 스레드 풀이 설정하는 이벤트를 지정합니다.
SetThreadpoolStackInformation 지정된 스레드 풀의 새 스레드에 대한 스택 예약 및 커밋 크기를 설정합니다. 기존 스레드에 대한 스택 예약 및 커밋 크기는 변경되지 않습니다.
SetThreadpoolThreadMaximum 지정된 스레드 풀이 콜백을 처리하기 위해 할당할 수 있는 최대 스레드 수를 설정합니다.
SetThreadpoolThreadMinimum 지정된 스레드 풀에서 콜백을 처리하는 데 사용할 수 있도록 해야 하는 최소 스레드 수를 설정합니다.
SetThreadpoolTimer 타이머 개체를 설정하여 이전 타이머(있는 경우)를 대체합니다. 작업자 스레드는 지정된 시간 제한이 만료된 후 타이머 개체의 콜백을 호출합니다. (SetThreadpoolTimer)
SetThreadpoolTimerEx 타이머 개체를 설정하여 이전 타이머(있는 경우)를 대체합니다. 작업자 스레드는 지정된 시간 제한이 만료된 후 타이머 개체의 콜백을 호출합니다. (SetThreadpoolTimerEx)
SetThreadpoolWait 대기 개체를 설정합니다. 이전 대기 개체(있는 경우)를 대체합니다. 작업자 스레드는 핸들이 신호를 받은 후 또는 지정된 시간 제한이 만료된 후 대기 개체의 콜백 함수를 호출합니다. (SetThreadpoolWait)
SetThreadpoolWaitEx 이전 대기 개체(있는 경우)를 대체하는 대기 개체를 설정합니다. 작업자 스레드는 핸들이 신호를 받은 후 또는 지정된 시간 제한이 만료된 후 대기 개체의 콜백 함수를 호출합니다. (SetThreadpoolWaitEx)
StartThreadpoolIo 지정된 I/O 완료 개체에 대해 I/O 작업이 시작될 수 있음을 스레드 풀에 알깁니다. 작업자 스레드는 이 개체에 바인딩된 파일 핸들에서 작업이 완료된 후 I/O 완성 개체의 콜백 함수를 호출합니다.
SubmitThreadpoolWork 스레드 풀에 작업 개체를 게시합니다. 작업자 스레드는 작업 개체의 콜백 함수를 호출합니다.

TrySubmitThreadpoolCallback 스레드 풀 작업자 스레드가 지정된 콜백 함수를 호출하도록 요청합니다.
WaitForThreadpoolIoCallbacks 미해결 I/O 완료 콜백이 완료될 때까지 대기하고 선택적으로 아직 실행이 시작되지 않은 보류 중인 콜백을 취소합니다.
WaitForThreadpoolTimerCallbacks 미해결 타이머 콜백이 완료될 때까지 대기하고 선택적으로 아직 실행이 시작되지 않은 보류 중인 콜백을 취소합니다.
WaitForThreadpoolWaitCallbacks 미해결 대기 콜백이 완료될 때까지 대기하고 선택적으로 아직 실행이 시작되지 않은 보류 중인 콜백을 취소합니다.
WaitForThreadpoolWorkCallbacks 미해결 작업 콜백이 완료될 때까지 대기하고 선택적으로 아직 실행이 시작되지 않은 보류 중인 콜백을 취소합니다.

Last updated on 2025. 10. 08.

threadpoollegacyapiset.h 헤더

아티클2023. 08. 26.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

threadpoollegacyapiset.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

ChangeTimerQueueTimer CreateTimerQueueTimer 함수에서 만든 타이머 큐 타이머를 업데이트.
CreateTimerQueue 타이머에 대한 큐를 만듭니다.
CreateTimerQueueTimer 타이머 큐 타이머를 만듭니다. 이 타이머는 지정된 기한에 만료된 다음 지정된 기간마다 만료됩니다. 타이머가 만료되면 콜백 함수가 호출됩니다.
DeleteTimerQueue 타이머 큐를 삭제합니다. 큐의 보류 중인 타이머는 모두 취소되고 삭제됩니다. (DeleteTimerQueue)
DeleteTimerQueueEx 타이머 큐를 삭제합니다. 큐의 보류 중인 타이머는 모두 취소되고 삭제됩니다. (DeleteTimerQueueEx)
DeleteTimerQueueTimer 타이머 큐에서 타이머를 제거하고 필요에 따라 타이머를 삭제하기 전에 현재 실행 중인 타이머 콜백 함수가 완료되기를 기다립니다.
QueueUserWorkItem 작업 항목을 스레드 풀의 작업자 스레드에 큐에 대기합니다.
UnregisterWaitEx RegisterWaitForSingleObject 함수에서 발급한 등록된 대기 작업을 취소합니다. (UnregisterWaitEx)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

thumbcache.h 헤더

아티클2023. 08. 26.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

thumbcache.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ISharedBitmap 비트맵에 액세스하기 위한 메모리 효율적인 메서드를 노출합니다. 이 인터페이스는 HBITMAP 개체에 대한 씬 래퍼로 사용되므로 해당 개체를 참조 횟수로 계산하고 기본 데이터가 변경되지 않도록 보호할 수 있습니다.
IThumbnailCache 애플리케이션 간에 공유되는 시스템 미리 보기 캐시에 대한 메서드를 노출합니다.
IThumbnailCachePrimer . (IThumbnailCachePrimer)
IThumbnailProvider 썸네일 이미지를 가져오기 위한 메서드를 노출하며 썸네일 처리기에 대해 구현될 예정입니다. 이 인터페이스를 구현하는 개체도 IInitializeWithStream을 구현해야 합니다.
IThumbnailSettings 썸네일 공급자가 썸네일 요청의 사용자 컨텍스트를 확인할 수 있도록 하는 메서드를 제공합니다.

구조체

WTS_THUMBNAILID 시스템 썸네일 캐시의 썸네일에 대한 고유 식별자를 포함합니다.

열거형

WTS_CONTEXTFLAGS 썸네일 추출의 컨텍스트를 지정합니다. IThumbnailSettings::SetContext에서 사용됩니다.
WTS_FLAGS IThumbnailCache::GetThumbnail에서 썸네일 이미지의 추출 및 표시 옵션을 지정하는 데 사용되는 값입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

thumbnailstreamcache.h 헤더

아티클2023. 08. 26.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

thumbnailstreamcache.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IThumbnailStreamCache 썸네일 스트림을 가져오거나 설정합니다. 이 인터페이스는 내부 전용이며 사진 애플리케이션에서만 호출할 수 있습니다.

열거형

ThumbnailStreamCacheOptions IThumbnailStreamCache 인터페이스에서 사용하는 캐시 옵션을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

timeapi.h 헤더

아티클2023. 08. 26.

이 헤더는 Windows 멀티미디어에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 멀티미디어

timeapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

timeBeginPeriod timeBeginPeriod 함수는 주기적 타이머에 대한 최소 해상도를 요청합니다.
timeEndPeriod timeEndPeriod 함수는 이전에 설정된 최소 타이머 해상도를 지웁니다.
timeGetDevCaps timeGetDevCaps 함수는 타이머 디바이스를 쿼리하여 해상도를 확인합니다.
timeGetSystemTime timeGetSystemTime 함수는 시스템 시간을 밀리초 단위로 검색합니다.
timeGetTime timeGetTime 함수는 시스템 시간을 밀리초 단위로 검색합니다. 시스템 시간은 Windows가 시작된 이후 경과된 시간입니다.

구조체

TIMECAPS TIMECAPS 구조에는 타이머의 해상도에 대한 정보가 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

timeprov.h 헤더

아티클2023. 08. 26.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

timeprov.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

TimeProvClose 시간 공급자 관리자가 시간 공급자를 종료하기 위해 호출하는 콜백 함수입니다.
TimeProvCommand 시간 공급자 관리자가 시간 공급자에게 명령을 보내기 위해 호출하는 콜백 함수입니다.
TimeProvOpen 시간 공급자 DLL이 로드될 때 시간 공급자 관리자가 호출하는 콜백 함수입니다.

콜백 함수

AlertSamplesAvailFunc 사용할 수 있는 새 시간 샘플이 있음을 시스템에 알 수 있습니다.
GetTimeSysInfoFunc 시스템 시간 상태 정보를 검색합니다.
LogTimeProvEventFunc 이벤트 로그에 시간 공급자 이벤트를 기록합니다.
SetProviderStatusFunc 시간 공급자의 상태 정보를 설정합니다.

SetProviderStatusInfoFreeFunc SetProviderStatusInfo 구조를 해제합니다.

구조체

SetProviderStatusInfo SetProviderStatusFunc 함수에서 사용되는 구조체입니다.
TimeProvSysCallbacks 시간 공급자가 사용할 함수에 대한 포인터를 포함합니다.
TimeSample 시간 샘플을 나타냅니다.
TpcGetSamplesArgs TimeProvCommand 함수에서 사용되는 구조체입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

timezoneapi.h 헤더

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- System Services

timezoneapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

EnumDynamicTimeZoneInformation 레지스트리에 저장된 DYNAMIC_TIME_ZONE_INFORMATION 항목을 열거합니다.
FileTimeToSystemTime 파일 시간을 시스템 시간 형식으로 변환합니다. 시스템 시간은 UTC(협정 세계시)를 기반으로 합니다.
GetDynamicTimeZoneInformation 현재 표준 시간대 및 동적 일광 절약 시간 설정을 검색합니다. 이러한 설정은 UTC(협정 세계시)와 현지 시간 간의 변환을 제어합니다.
GetDynamicTimeZoneInformationEffectiveYears DYNAMIC_TIME_ZONE_INFORMATION 유효한 항목이 있는 연도로 표현된 범위를 가져옵니다.
GetTimeZoneInformation 현재 표준 시간대 설정을 검색합니다. 이러한 설정은 UTC(협정 세계시)와 현지 시간 간의 변환을 제어합니다.
GetTimeZoneInformationForYear 지정된 연도 및 표준 시간대에 대한 표준 시간대 설정을 검색합니다. 이러한 설정은 UTC(협정 세계시)와 현지 시간 간의 변환을 제어합니다.
SetDynamicTimeZoneInformation 현재 표준 시간대 및 동적 일광 절약 시간 설정을 설정합니다. 이러한 설정은 UTC(협정 세계시)에서 현지 시간으로의 변환을 제어합니다.
SetTimeZoneInformation

현재 표준 시간대 설정을 설정합니다. 이러한 설정은 UTC(협정 세계시)에서 현지 시간으로의 변환을 제어합니다.
SystemTimeToFileTime 시스템 시간을 파일 시간 형식으로 변환합니다. 시스템 시간은 UTC(협정 세계시)를 기반으로 합니다.
SystemTimeToTzSpecificLocalTime UTC(협정 세계시)의 시간을 지정된 표준 시간대의 해당 현지 시간으로 변환합니다.
SystemTimeToTzSpecificLocalTimeEx 동적 일광 절약 시간 설정이 있는 UTC(협정 세계시)의 시간을 지정된 표준 시간대의 해당 현지 시간으로 변환합니다.
TzSpecificLocalTimeToSystemTime 지정된 현지 시간을 UTC(협정 세계시)의 해당 시간으로 변환합니다.
TzSpecificLocalTimeToSystemTimeEx DST(동적 일광 절약 시간제) 설정을 사용하여 지정된 현지 시간을 UTC(협정 세계시)의 해당 시간으로 변환합니다.

구조

테이블 확장

DYNAMIC_TIME_ZONE_INFORMATION 표준 시간대 및 동적 일광 절약 시간에 대한 설정을 지정합니다.
TIME_ZONE_INFORMATION 표준 시간대에 대한 설정을 지정합니다.

Last updated on 2025. 12. 18.

tlhelp32.h 헤더

아티클2023. 08. 26.

이 헤더는 도구 도움말 라이브러리에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 도구 도움말 라이브러리

tlhelp32.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CreateToolhelp32Snapshot 이러한 프로세스에서 사용되는 힙, 모듈 및 스레드뿐만 아니라 지정된 프로세스의 스냅샷 사용합니다.
Heap32First 프로세스에서 할당한 힙의 첫 번째 블록에 대한 정보를 검색합니다.
Heap32ListFirst 지정된 프로세스에 의해 할당된 첫 번째 힙에 대한 정보를 검색합니다.
Heap32ListNext 프로세스에서 할당한 다음 힙에 대한 정보를 검색합니다.
Heap32Next 프로세스에서 할당한 힙의 다음 블록에 대한 정보를 검색합니다.
Module32First Module32First 함수(tlhelp32.h)는 프로세스와 연결된 첫 번째 모듈에 대한 정보를 검색합니다.
Module32FirstW Module32FirstW(유니코드) 함수(tlhelp32.h)는 프로세스와 연결된 첫 번째 모듈에 대한 정보를 검색합니다.
Module32Next Module32Next 함수(tlhelp32.h)는 프로세스 또는 스레드와 연결된 다음 모듈에 대한 정보를 검색합니다.

Module32NextW Module32NextW(유니코드) 함수(tlhelp32.h)는 프로세스 또는 스레드와 연결된 다음 모듈에 대한 정보를 검색합니다.
Process32First Process32First 함수(tlhelp32.h)는 시스템 스냅샷 발생한 첫 번째 프로세스에 대한 정보를 검색합니다.
Process32FirstW Process32FirstW(유니코드) 함수(tlhelp32.h)는 시스템 스냅샷 발생한 첫 번째 프로세스에 대한 정보를 검색합니다.
Process32Next Process32Next 함수(tlhelp32.h)는 시스템 스냅샷 기록된 다음 프로세스에 대한 정보를 검색합니다.
Process32NextW Process32NextW(유니코드) 함수(tlhelp32.h)는 시스템 스냅샷 기록된 다음 프로세스에 대한 정보를 검색합니다.
Thread32First 시스템 스냅샷 발생한 프로세스의 첫 번째 스레드에 대한 정보를 검색합니다.
Thread32Next 시스템 메모리 스냅샷 발생한 프로세스의 다음 스레드에 대한 정보를 검색합니다.
Toolhelp32ReadProcessMemory 다른 프로세스에 할당된 메모리를 애플리케이션 제공 버퍼에 복사합니다.

구조체

HEAPENTRY32 검사 중인 힙의 한 항목(블록)을 설명합니다.
HEAPLIST32 지정된 프로세스에서 사용하는 힙을 열거하는 목록의 항목을 설명합니다.

MODULEENTRY32 MODULEENTRY32(tlhelp32.h) 구조체는 지정된 프로세스에 속하는 모듈 목록의 항목을 설명합니다.
MODULEENTRY32W MODULEENTRY32W(유니코드) 구조체(tlhelp32.h)는 지정된 프로세스에 속하는 모듈 목록의 항목을 설명합니다.
PROCESSENTRY32 PROCESSENTRY32(tlhelp32.h) 구조체는 스냅샷 수행되었을 때 시스템 주소 공간에 있는 프로세스 목록의 항목을 설명합니다.
PROCESSENTRY32W PROCESSENTRY32W(유니코드) 구조체(tlhelp32.h)는 스냅샷 수행되었을 때 시스템 주소 공간에 있는 프로세스 목록의 항목을 설명합니다.
THREADENTRY32 스냅샷 수행되었을 때 시스템에서 실행되는 스레드 목록의 항목을 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

tlogstg.h 헤더

아티클2023. 08. 26.

이 헤더는 Windows 셸에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 셸

tlogstg.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

WINDOWDATA 창 데이터를 저장합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

tokenbinding.h 헤더

아티클2023. 03. 08.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

tokenbinding.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

TokenBindingDeleteAllBindings 호출하는 사용자 또는 앱 컨테이너와 연결된 모든 토큰 바인딩 키를 삭제합니다.
TokenBindingDeleteBinding 지정된 대상 문자열과 연결된 토큰 바인딩 키를 삭제합니다.
TokenBindingGenerateBinding 토큰 바인딩에 대해 지정된 키 형식, 토큰 바인딩 키를 만들고 검색하기 위한 대상 식별자 문자열 및 고유 데이터를 사용하여 내보낸 공개 키와 서명을 포함하는 하나의 토큰 바인딩을 생성합니다.
TokenBindingGenerateID 키 형식에서 서명 알고리즘을 추출하고 내보낸 공개 키를 복사하여 토큰 바인딩 식별자를 생성합니다.
TokenBindingGenerateMessage 토큰 바인딩 목록을 어셈블하고 클라이언트 디바이스에 대한 최종 메시지를 서버에 생성합니다.
TokenBindingGetKeyTypesClient 클라이언트 디바이스에서 지원하는 키 유형의 목록을 검색합니다.
TokenBindingGetKeyTypesServer 서버에서 지원하는 키 유형의 목록을 검색합니다.
TokenBindingVerifyMessage 토큰 바인딩 메시지의 유효성을 검사하고 메시지에 포함된 토큰 바인딩을 확인합니다.

구조체

TOKENBINDING_IDENTIFIER 토큰 바인딩 메시지 교환에서 발생하는 토큰 바인딩 식별자를 나타내는 정보를 포함합니다.
TOKENBINDING_KEY_TYPES 클라이언트 디바이스 또는 서버에서 지원하는 토큰 바인딩 키 형식의 모든 조합을 포함합니다.
TOKENBINDING_RESULT_DATA 토큰 바인딩을 생성하거나 토큰 바인딩 메시지에서 토큰 바인딩 중 하나를 확인한 결과에 대한 데이터를 포함합니다.
TOKENBINDING_RESULT_LIST TokenBindingVerifyMessage에서 확인한 각 토큰 바인딩에 대한 결과를 포함합니다.

열거형

TOKENBINDING_EXTENSION_FORMAT 확장 데이터를 해석하는 데 사용할 수 있는 형식을 지정합니다.
TOKENBINDING_TYPE 토큰 바인딩에 사용할 수 있는 형식을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

tom.h 헤더

이 헤더는 Windows 컨트롤에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 컨트롤

tom.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Interfaces

테이블 확장

ITextDisplays 이 TOM(텍스트 개체 모델) 엔진 인스턴스에 대한 표시 컬렉션을 나타냅니다.
ITextDocument ITextDocument 인터페이스는 TOM(텍스트 개체 모델) 최상위 인터페이스로, 활성 상태인지 여부에 관계없이 문서의 모든 텍스트 영역에 대한 활성 선택 영역 및 범위 개체를 검색합니다.
ITextDocument2 ITextDocument 인터페이스를 확장하여 IME(입력 메서드 편집기)가 다양한 편집 컨트롤을 구동할 수 있도록 하는 메서드와 ITextDisplays, ITextRange2, ITextFont2, ITextPara2 등과 같은 다른 인터페이스를 검색하는 메서드를 추가합니다.
ITextFont TOM(텍스트 개체 모델) 서식 있는 텍스트 범위 특성은 ITextFont 및 ITextPara 쌍의 이중 인터페이스를 통해 액세스됩니다. (ITextFont)
ITextFont2 TOM(텍스트 개체 모델)에서 애플리케이션은 ITextFont 및 ITextPara 쌍의 이중 인터페이스를 사용하여 텍스트 범위 특성에 액세스합니다.
ITextPara TOM(텍스트 개체 모델) 서식 있는 텍스트 범위 특성은 ITextFont 및 ITextPara 쌍의 이중 인터페이스를 통해 액세스됩니다. (ITextPara)
ITextPara2 TOM(텍스트 개체 모델) 서식 있는 텍스트 범위 특성은 ITextFont 및 ITextPara 쌍의 이중 인터페이스를 통해 액세스됩니다. (ITextPara2)

ITextRange ITextRange 개체는 프로그램에서 텍스트 영역의 텍스트를 선택한 다음 해당 텍스트를 검사하거나 변경할 수 있는 강력한 편집 및 데이터 바인딩 도구입니다.
ITextRange2 ITextRange2 인터페이스는 ITextRange에서 파생되며, 해당 개체는 프로그램에서 스토리의 텍스트를 선택한 다음 해당 텍스트를 검사하거나 변경할 수 있는 강력한 편집 및 데이터 바인딩 도구입니다.
ITextRow ITextRow 인터페이스는 하나 이상의 동일한 테이블 행을 삽입하고 테이블 행 속성을 검색하고 변경하는 메서드를 제공합니다. 비분석 행을 삽입하려면 서로 다른 각 행 구성에 대해 ITextRow::Insert를 호출합니다.
ITextSelection 텍스트 선택 영역은 선택 영역이 강조 표시된 텍스트 범위입니다.
ITextSelection2 현재 이 인터페이스에는 ITextRange2에서 상속된 메서드 이외의 메서드가 없습니다.
ITextStory ITextStory 인터페이스 메서드는 부모 ITextServices 인스턴스에 저장된 여러 스토리에서 공유 데이터에 액세스하는 데 사용됩니다.
ITextStoryRanges ITextStoryRanges 인터페이스의 목적은 ITextDocument의 스토리를 열거하는 것입니다.
ITextStoryRanges2 ITextStoryRanges2 인터페이스는 ITextDocument의 스토리를 열거합니다.
ITextStrings ITextStrings 인터페이스는 서식 있는 텍스트를 조작하는 데 유용한 서식 있는 텍스트 문자열 컬렉션을 나타냅니다.

열거 목록

테이블 확장

MANCODE 수학적 영숫자 코드를 나타냅니다.
OBJECTTYPE TOM(텍스트 개체 모델) 콘텐츠의 개체 형식을 식별하는 값을 정의합니다.
tomConstants TOM(텍스트 개체 모델) API와 함께 사용되는 값을 정의합니다.

Last updated on 2025. 10. 08.

tpcshrd.h 헤더

아티클2023. 08. 26.

이 헤더는 태블릿 PC에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Tablet PC

tpcshrd.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

PACKET_DESCRIPTION 특정 태블릿 인식기 컨텍스트에 대한 패킷의 내용을 설명합니다. 이 구조를 사용하여 패킷에 포함된 데이터에 액세스하지 마세요. 이 구조체는 패킷의 내용을 설명합니다.
PACKET_PROPERTY 태블릿 드라이버에서 보고하는 패킷 속성을 설명합니다.
PROPERTY_METRICS 패킷 속성의 범위와 해상도를 정의합니다.
STROKE_RANGE InkDisp 개체의 스트로크 범위를 지정합니다.
SYSTEM_EVENT_DATA 태블릿 시스템 이벤트에 대한 정보를 포함합니다.

열거형

PROPERTY_UNITS 속성의 측정 단위에 대한 상수 값을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

tpmvscmgr.h 헤더

아티클2023. 08. 26.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

tpmvscmgr.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ITpmVirtualSmartCardManager TPM 가상 스마트 카드를 관리합니다.
ITpmVirtualSmartCardManagerStatusCallback 호출자에게 요청된 작업의 진행률 또는 결과 오류를 알 수 있습니다.

열거형

TPMVSCMGR_ERROR TPM 가상 스마트 카드 관리자의 오류 컨텍스트를 나타내는 미리 정의된 오류 코드를 제공합니다.
TPMVSCMGR_STATUS TPM 가상 스마트 카드 관리자의 진행률을 나타내는 미리 정의된 상태 코드를 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

traceloggingactivity.h 헤더

아티클2023. 08. 26.

이 헤더는 TraceLogging에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- TraceLogging

traceloggingactivity.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

TraceLoggingActivity 활동 중에 ETW 이벤트 로깅을 지원합니다. 모든 이벤트는 수동으로 태그를 지정하거나 중첩해야 합니다.
TraceLoggingThreadActivity 활동 중에 ETW 이벤트 로깅을 지원합니다. 이벤트는 이 활동에서 자동으로 태그가 지정되거나 중첩됩니다.
TraceLoggingThreadActivityIdSetter ETW가 해당 스레드의 모든 이벤트를 활동 ID로 표시할 수 있도록 작업 ID로 스레드에 태그를 지정합니다.

Functions

TraceLoggingFunction 현재 함수의 이름을 따서 명명된 TraceLoggingThreadActivity를 만들고 활동에 대한 Start 이벤트를 작성합니다. 중지 작업은 현재 scope 끝에 기록됩니다.
TraceLoggingWriteStart 활동을 시작하고 시작 이벤트를 기록합니다.
TraceLoggingWriteStop 활동을 중지하고 중지 이벤트를 기록합니다.

TraceLoggingWriteTagged 연결된 ETW 활동 ID를 사용하여 이벤트를 기록합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

traceloggingprovider.h 헤더

아티클2023. 08. 26.

TraceLogging 은 매니페스트 없이 디코딩할 수 있는 자체 설명 이벤트를 로깅하기 위한 시스템입니다. Windows에서 TraceLogging은 ETW(Windows용 이벤트 추적) 이벤트를 생성하는 데 사용됩니다.

Windows SDK의 **TraceLoggingProvider.h** 헤더에는 C 또는 C++를 사용하여 커널 및 사용자 모드 코드에 대한 TraceLogging 인코딩 ETW 이벤트를 생성하는 매크로 및 인라인 함수가 있습니다.

> ⓘ **참고**
>
> **TraceLoggingProvider.h** 에는 공급자 이름, 이벤트 이름 및 필드 이름과 같은 이벤트 특성에 대한 컴파일 시간 상수 값이 필요합니다. 런타임 오버헤드를 최소화하기 위해 **TraceLoggingProvider.h** 는 컴파일 시간에 해당 데이터 구조를 빌드하고 정보를 읽기 전용 메모리에 저장합니다. 런타임 동적 이벤트를 생성해야 하는 경우 TraceLoggingDynamic ↗ 과 같은 다른 TraceLogging 구현을 사용해야 합니다.

빠른 시작

- .c 또는 .cpp 파일에서 TRACELOGGING_DEFINE_PROVIDER 매크로를 사용하여 전역 공급자 핸들을 선언합니다. 공급자 핸들은 ETW에 대한 구성 요소의 연결을 나타냅니다.
- 구성 요소 시작 시(예: , `wmain`, `DllMain`또는 `DriverEntry`)에서 `main`TraceLoggingRegister 함수를 사용하여 ETW에 대한 구성 요소의 연결을 엽니다.
- 구성 요소 종료 시 TraceLoggingUnregister 함수를 사용하여 구성 요소의 ETW 연결을 닫습니다.
- 구성 요소를 실행하는 동안 TraceLoggingWrite 매크로를 사용하여 TraceLogging으로 인코딩된 ETW 이벤트를 생성합니다.
- 필요에 따라 헤더의 TRACELOGGING_DECLARE_PROVIDER 매크로를 사용하여 공급자 핸들을 전달 선언하여 구성 요소의 다른 부분에서 사용할 수 있습니다.
- WPR, 추적 로그 또는 추적 보기와 같은 도구를 사용하여 추적을 수집합니다.
- WPA, tracefmt 또는 traceview와 같은 도구를 사용하여 추적을 디코딩하고 봅니다.

예제

```
C
```

```
#include <windows.h> // or <wdm.h> for kernel-mode.
#include <winmeta.h> // For event level definitions.
#include <TraceLoggingProvider.h>

TRACELOGGING_DEFINE_PROVIDER( // defines g_hProvider
    g_hProvider, // Name of the provider handle
    "MyCompany.MyComponent", // Human-readable name for the provider
    // {ce5fa4ea-ab00-5402-8b76-9f76ac858fb5}
    (0xce5fa4ea,0xab00,0x5402,0x8b,0x76,0x9f,0x76,0xac,0x85,0x8f,0xb5));

int main(int argc, char* argv[]) // or DriverEntry for kernel-mode.
{
    TraceLoggingRegister(g_hProvider);

    TraceLoggingWrite(
        g_hProvider,
        "MyEvent1",
        TraceLoggingLevel(WINEVENT_LEVEL_WARNING), // Levels defined in
<winmeta.h>
        TraceLoggingKeyword(MyEventCategories), // Provider-defined
categories
        TraceLoggingString(argv[0], "arg0"), // field name is "arg0"
        TraceLoggingInt32(argc)); // field name is implicitly "argc"

    TraceLoggingUnregister(g_hProvider);
    return 0;
}
```

자세한 내용은 다음을 참조하세요.

- TraceLogging
- TraceLogging 래퍼 매크로
- TraceLoggingWrite
- TRACELOGGING_DECLARE_PROVIDER
- TRACELOGGING_DEFINE_PROVIDER

traceloggingprovider.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

TRACELOGGING_DECLARE_PROVIDER TraceLogging 공급자에 대한 핸들을 전달 선언합니다.
TRACELOGGING_DEFINE_PROVIDER

TraceLogging 공급자에 대한 핸들을 정의합니다.
TRACELOGGING_DEFINE_PROVIDER_STORAGE 사용자가 정의할 TraceLogging 공급자 핸들에 대한 정적 스토리지를 예약합니다. 이 매크로보다 TRACELOGGING_DEFINE_PROVIDER 것이 좋습니다.
TraceLoggingBinary 이진 데이터가 있는 필드를 이벤트에 추가하는 TraceLogging 래퍼 매크로입니다.
TraceLoggingChannel 이벤트의 채널을 설정하는 TraceLogging 래퍼 매크로입니다.
TraceLoggingCustom 사용자 지정 serializer를 사용하여 압축된 필드를 이벤트에 추가하는 TraceLogging 래퍼 매크로입니다.
TraceLoggingCustomAttribute 이벤트에 대한 사용자 지정 정보를 PDB에 추가하는 TraceLogging 래퍼 매크로입니다.
TraceLoggingDescription 이벤트에 대한 설명을 설정하는 TraceLogging 래퍼 매크로입니다.
TraceLoggingEventTag 이벤트에 대한 이벤트 태그를 설정하는 TraceLogging 래퍼 매크로입니다.
TraceLoggingKeyword 이벤트에 대한 키워드(keyword) 설정하는 TraceLogging 래퍼 매크로입니다.
TraceLoggingLevel 이벤트의 수준을 설정하는 TraceLogging 래퍼 매크로
TraceLoggingOpcode 이벤트에 대한 opcode를 설정하는 TraceLogging 래퍼 매크로
TraceLoggingOptionGroup TRACELOGGING_DEFINE_PROVIDER 공급자 그룹을 지정하는 데 사용할 TraceLogging 매크로입니다.

TraceLoggingProviderEnabled 추적 소비자가 이 공급자의 이벤트를 수신 대기하는지 여부를 확인하는 추적 로깅 매크로입니다.
TraceLoggingProviderId TraceLogging 공급자의 공급자 ID를 가져옵니다.
TraceLoggingRegister 이벤트를 기록하는 데 사용할 수 있도록 TraceLogging 공급자를 등록합니다.
TraceLoggingRegisterEx ETW 사용 콜백을 지정하여 이벤트를 기록하는 데 사용할 수 있도록 TraceLogging 공급자를 등록합니다.
TraceLoggingSetInformation EventSetInformation을 호출하여 TraceLogging 공급자를 구성합니다.
TraceLoggingSocketAddress 소켓 주소가 있는 필드를 이벤트에 추가하는 TraceLogging 래퍼 매크로입니다.
TraceLoggingStruct 이벤트에 다른 필드가 포함된 필드를 추가하는 TraceLogging 래퍼 매크로입니다.
TraceLoggingUnregister TraceLogging 공급자의 등록을 취소합니다.
TraceLoggingValue 이벤트에 자동으로 추론된 형식의 필드를 추가하는 C++용 TraceLogging 래퍼 매크로입니다.
TraceLoggingWrite TraceLogging 이벤트를 내보낸다.
TraceLoggingWriteActivity 지정된 활동 ID를 사용하여 TraceLogging 이벤트를 내보낸다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

traffic.h 헤더

이 헤더는 QOS(서비스 품질)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 서비스 품질(QOS)

traffic.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

TcAddFilter TcAddFilter 함수는 필터와 일치하는 패킷을 연결된 흐름으로 보낼 수 있는 기존 흐름과 새 필터를 연결합니다.
TcAddFlow TcAddFlow 함수는 지정된 인터페이스에 새 흐름을 추가합니다.
TcCloseInterface TcCloseInterface 함수는 TcOpenInterface 호출을 사용하여 이전에 연 인터페이스를 닫습니다. TcCloseInterface 호출을 사용하여 인터페이스를 닫기 전에 특정 인터페이스의 모든 흐름과 필터를 닫아야 합니다.
TcDeleteFilter TcDeleteFilter 함수는 이전에 TcAddFilter 함수와 함께 추가된 필터를 삭제합니다.
TcDeleteFlow TcDeleteFlow 함수는 TcAddFlow 함수와 함께 추가된 흐름을 삭제합니다. 클라이언트는 흐름을 삭제하기 전에 흐름과 연결된 모든 필터를 삭제해야 합니다. 그렇지 않으면 오류가 반환되고 함수는 흐름을 삭제하지 않습니다.
TcDeregisterClient TcDeregisterClient 함수는 TCI(트래픽 제어 인터페이스)를 사용하여 클라이언트를 등록 취소합니다.
TcEnumerateFlows TcEnumerateFlows 함수는 설치된 흐름과 연결된 필터를 인터페이스에 열거합니다.
TcEnumerateInterfaces

TcEnumerateInterfaces 함수는 모든 트래픽 제어(사용 가능한 네트워크 인터페이스)를 열거합니다. 클라이언트는 ClNotifyHandler 함수를 통해 인터페이스 변경에 대한 알림을 받습니다.
TcGetFlowNameA TcGetFlowName 함수는 호출 클라이언트에서 만든 흐름의 이름을 제공합니다. (ANSI)
TcGetFlowNameW TcGetFlowName 함수는 호출 클라이언트에서 만든 흐름의 이름을 제공합니다. (유니코드)
TcModifyFlow TcModifyFlow 함수는 기존 흐름을 수정합니다. TcModifyFlow를 호출할 때 새 Flowspec 매개 변수 및 모든 트래픽 제어 개체를 채워야 합니다.
TcOpenInterfaceA TcOpenInterface 함수가 인터페이스를 엽니다. (ANSI)
TcOpenInterfaceW TcOpenInterface 함수가 인터페이스를 엽니다. (유니코드)
TcQueryFlowA TcQueryFlow 함수는 흐름 이름에 따라 특정 흐름 매개 변수의 값에 대한 트래픽 제어를 쿼리합니다. 흐름의 이름은 TcEnumerateFlows 함수 또는 TcGetFlowName 함수에서 검색할 수 있습니다. (ANSI)
TcQueryFlowW TcQueryFlow 함수는 흐름 이름에 따라 특정 흐름 매개 변수의 값에 대한 트래픽 제어를 쿼리합니다. 흐름의 이름은 TcEnumerateFlows 함수 또는 TcGetFlowName 함수에서 검색할 수 있습니다. (유니코드)
TcQueryInterface TcQueryInterface 함수는 관련 인터페이스별 매개 변수에 대한 트래픽 제어를 쿼리합니다.
TcRegisterClient TcRegisterClient 함수는 TCI(트래픽 제어 인터페이스)에 클라이언트를 등록하는 데 사용됩니다. TcRegisterClient 함수는 클라이언트가 TCI에 대해 하는 첫 번째 함수 호출이어야 합니다.
TcSetFlowA TcSetFlow 함수는 지정된 흐름에 대한 개별 매개 변수를 설정합니다. (ANSI)
TcSetFlowW

TcSetFlow 함수는 지정된 흐름에 대한 개별 매개 변수를 설정합니다. (유니코드)
TcSetInterface TcSetInterface 함수는 지정된 인터페이스에 대한 개별 매개 변수를 설정합니다.

콜백 함수

테이블 확장

TCI_ADD_FLOW_COMPLETE_HANDLER ClAddFlowComplete 함수는 트래픽 제어에서 TcAddFlow 함수에 대한 이전 호출의 완료를 클라이언트에 알리는 데 사용됩니다.
TCI_DEL_FLOW_COMPLETE_HANDLER ClDeleteFlowComplete 함수는 트래픽 제어에서 TcDeleteFlow 함수에 대한 이전 호출이 완료되었음을 클라이언트에 알리는 데 사용됩니다.
TCI_MOD_FLOW_COMPLETE_HANDLER ClModifyFlowComplete 함수는 트래픽 제어에서 TcModifyFlow 함수에 대한 이전 호출이 완료되었음을 클라이언트에 알리는 데 사용됩니다.
TCI_NOTIFY_HANDLER ClNotifyHandler 함수는 트래픽 제어에서 흐름 삭제, 필터 매개 변수 변경 또는 인터페이스 닫기를 비롯한 다양한 트래픽 제어 특정 이벤트를 클라이언트에 알리는 데 사용됩니다.

구조

테이블 확장

ADDRESS_LIST_DESCRIPTOR ADDRESS_LIST_DESCRIPTOR 구조는 지정된 인터페이스에 대한 네트워크 주소 설명자 정보를 제공합니다.
ENUMERATION_BUFFER

ENUMERATION_BUFFER 구조에는 흐름 이름, 흐름과 연결된 필터 수 및 흐름과 연결된 필터 배열을 포함하여 지정된 흐름과 관련된 정보가 포함됩니다.
IP_PATTERN IP_PATTERN 구조체는 IP 프로토콜에 특정 패턴 또는 해당 마스크를 적용합니다. IP_PATTERN 구조 지정은 패킷 필터 애플리케이션의 트래픽 제어 인터페이스에서 사용됩니다.
IPX_PATTERN IPX_PATTERN 구조체는 IPX 프로토콜에 특정 패턴 또는 해당 마스크를 적용합니다. IPX_PATTERN 구조 지정은 패킷 필터의 애플리케이션에서 트래픽 제어 인터페이스에서 사용됩니다.
TC_GEN_FILTER TC_GEN_FILTER 구조는 특정 패킷 특성 또는 조건 집합과 일치하는 필터를 만듭니다. 이 필터는 이후에 특성 조건을 충족하는 패킷을 특정 흐름과 연결하는 데 사용할 수 있습니다.
TC_GEN_FLOW TC_GEN_FLOW 구조체는 트래픽 제어 인터페이스와 함께 사용할 제네릭 흐름을 만듭니다. 흐름은 이 구조체의 멤버를 통해 사용자 지정됩니다.
TC_IFC_DESCRIPTOR TC_IFC_DESCRIPTOR 구조체는 인터페이스를 열거하는 데 사용되는 인터페이스 식별자입니다.
TCI_CLIENT_FUNC_LIST TCI_CLIENT_FUNC_LIST 구조는 트래픽 제어 인터페이스에서 클라이언트 콜백 함수를 등록한 다음 액세스하는 데 사용됩니다. TCI_CLIENT_FUNC_LIST 각 멤버는 클라이언트에서 제공하는 콜백 함수에 대한 포인터입니다.

Last updated on 2025. 10. 08.

transportsettingcommon.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows 소켓 2에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 소켓 2

transportsettingcommon.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

TRANSPORT_SETTING_ID TRANSPORT_SETTING_ID(transportsettingcommon.h) 구조체는 SIO_APPLY_TRANSPORT_SETTING 및 SIO_QUERY_TRANSPORT_SETTING IOCTL에서 사용하는 전송 설정 ID를 지정합니다.

tsgauthenticationengine.h 헤더

아티클2023. 03. 08.

이 헤더는 원격 데스크톱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 데스크톱 서비스

tsgauthenticationengine.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ITSGAuthenticateUserSink 인증 이벤트에 대해 RD 게이트웨이(원격 데스크톱 게이트웨이)에 알리는 메서드를 노출합니다.
ITSGAuthenticationEngine RD 게이트웨이(원격 데스크톱 게이트웨이)에 대한 사용자를 인증하는 메서드를 노출합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

tsgpolicyengine.h 헤더

아티클2023. 08. 26.

이 헤더는 원격 데스크톱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 데스크톱 서비스

tsgpolicyengine.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ITSGAccountingEngine 연결에 대한 세션 생성 또는 종료에 대한 정보를 제공하는 메서드를 노출합니다.
ITSGAuthorizeConnectionSink 연결 권한을 부여하려는 시도의 결과를 RD 게이트웨이(원격 데스크톱 게이트웨이)에 알리는 메서드를 노출합니다.
ITSGAuthorizeResourceSink 리소스 권한 부여 시도의 결과에 대해 RD 게이트웨이(원격 데스크톱 게이트웨이)에 알리는 메서드를 노출합니다.
ITSGPolicyEngine 연결 및 리소스에 권한을 부여하는 메서드를 노출합니다.

구조체

AAAccountingData 이 구조에는 연결 이벤트에 대한 정보가 포함되어 있습니다.

열거형

AAAccountingDataType ITSGAccountingEngine::D oAccounting 메서드가 알림을 받는 이벤트 유형을 지정합니다.
AAAuthSchemes RD 게이트웨이(원격 데스크톱 게이트웨이)에 연결하는 데 사용되는 인증 유형을 지정합니다.
PolicyAttributeType 연결과 연결된 리디렉션 설정을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

tspi.h 헤더

아티클2023. 08. 26.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- TAPI 2.2
- 전화 통신 서비스 공급자

tspi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

TSPI_lineAccept TSPI_lineAccept 함수는 지정된 제공된 호출을 수락합니다. 필요에 따라 지정된 사용자-사용자 정보를 통화 당사자에게 보낼 수 있습니다.
TSPI_lineAddToConference TSPI_lineAddToConference 함수는 hdConsultCall로 지정된 호출을 hdConfCall로 지정된 전화 회의에 추가합니다.
TSPI_lineAnswer TSPI_lineAnswer 함수는 지정된 제품 호출에 응답합니다.
TSPI_lineBlindTransfer TSPI_lineBlindTransfer 함수는 지정된 대상 주소에 대한 지정된 호출의 블라인드 또는 단일 단계 전송을 수행합니다.
TSPI_lineClose TSPI_lineClose 함수는 디바이스에서 모든 미해결 호출 및 비동기 작업을 완료하거나 중단한 후 지정된 열린 선 디바이스를 닫습니다.
TSPI_lineCloseCall TSPI_lineCloseCall 함수는 호출에서 미해결 비동기 작업을 모두 완료하거나 중단한 후 호출의 할당을 취소합니다.
TSPI_lineCloseMSPInstance TSPI_lineCloseMSPInstance 함수는 TAPI 3 DLL에 MSP(미디어 서비스 공급자) 호출 instance 닫도록 지시합니다. 이 함수에는 TAPI 3.0 버전 협상이 필요합니다.

TSPI_lineCompleteCall TSPI_lineCompleteCall 함수는 정상적으로 연결할 수 없는 호출을 대신 완료하는 방법을 지정하는 데 사용됩니다.
TSPI_lineCompleteTransfer TSPI_lineCompleteTransfer 함수는 상담 통화에 연결된 당사자에게 지정된 통화의 전송을 완료합니다.
TSPI_lineConditionalMediaDetection TSPI_lineConditionalMediaDetection 함수는 클라이언트 애플리케이션이 lineOpen 함수 호출에서 LINEMAPPER를 dwDeviceID로 사용할 때마다 TAPI에서 호출됩니다.
TSPI_lineConfigDialog TSPI_lineConfigDialog 함수는 사용되지 않습니다. TAPI 버전 1.4 또는 이전 서비스 공급자는 이 TSPI 함수를 구현할 수 있습니다. TAPI 버전 2.0 이상 TSP는 TUISPI_lineConfigDialog 구현합니다.
TSPI_lineConfigDialogEdit TSPI_lineConfigDialogEdit 함수는 사용되지 않습니다. TAPI 버전 1.4 또는 이전 서비스 공급자는 이 TSPI 함수를 구현할 수 있습니다. TAPI 버전 2.0 이상 TSP는 TUISPI_lineConfigDialogEdit 구현합니다.
TSPI_lineCreateMSPInstance TSPI_lineCreateMSPInstance 함수는 TAPI 3 DLL에 특정 라인 디바이스에 대한 MSP(미디어 서비스 공급자) instance 만들도록 지시합니다. 이 함수는 MSP 호출에 대한 TSP 핸들을 반환합니다. 이 함수에는 TAPI 3.0 버전 협상이 필요합니다.
TSPI_lineDevSpecific TSPI_lineDevSpecific 함수는 서비스 공급자가 다른 작업에 설명되지 않은 기능에 대한 액세스를 제공할 수 있도록 하는 일반적인 확장 메커니즘으로 사용됩니다.
TSPI_lineDevSpecificFeature TSPI_lineDevSpecificFeature 함수는 서비스 공급자가 다른 작업에 설명되지 않은 기능에 대한 액세스를 제공할 수 있도록 하는 확장 메커니즘으로 사용됩니다.
TSPI_lineDial TSPI_lineDial 함수는 지정된 통화에서 지정된 전화 걸기 번호로 전화를 켭니다.
TSPI_lineDrop TSPI_lineDrop 함수는 지정된 호출을 삭제하거나 연결을 끊습니다.

TSPI_lineForward TSPI_lineForward 함수는 지정된 전달 지침에 따라 지정된 줄의 지정된 주소로 향하는 호출을 전달합니다.
TSPI_lineGatherDigits TSPI_lineGatherDigits 함수는 지정된 호출에서 버퍼링된 숫자 수집을 시작합니다. TAPI는 수집할 숫자와 최대 자릿수를 배치할 버퍼를 지정합니다.
TSPI_lineGenerateDigits TSPI_lineGenerateDigits 함수는 지정된 신호 모드를 사용하여 지정된 호출에서 지정된 숫자를 대역내 톤으로 생성하기 시작합니다.
TSPI_lineGenerateTone TSPI_lineGenerateTone 함수는 지정된 호출을 통해 지정된 톤 인밴드를 생성합니다.
TSPI_lineGetAddressCaps TSPI_lineGetAddressCaps 함수는 지정된 줄 디바이스에서 지정된 주소를 쿼리하여 전화 통신 기능을 확인합니다.
TSPI_lineGetAddressID TSPI_lineGetAddressID 함수는 지정된 줄에서 주소와 연결된 주소 식별자를 다른 형식으로 반환합니다.
TSPI_lineGetAddressStatus TSPI_lineGetAddressStatus 함수는 현재 상태 대해 지정된 주소를 쿼리합니다.
TSPI_lineGetCallAddressID TSPI_lineGetCallAddressID 함수는 표시된 호출의 주소 식별자를 검색합니다.
TSPI_lineGetCallHubTracking TSPI_lineGetCallHubTracking 함수는 서비스 공급자에 대한 CallHub 추적의 현재 상태를 반환합니다. 이 함수에는 TAPI 3.0 버전 협상이 필요합니다.
TSPI_lineGetCallIDs TSPI_lineGetCallIDs 함수는 서비스 공급자에 대한 호출 식별자를 반환합니다. 이 함수에는 TAPI 3.0 버전 협상이 필요합니다.
TSPI_lineGetCallInfo

TSPI_lineGetCallInfo 함수는 지정된 호출에 대한 자세한 정보를 반환합니다.
TSPI_lineGetCallStatus TSPI_lineGetCallStatus 함수는 지정된 호출의 현재 상태 반환합니다.
TSPI_lineGetDevCaps TSPI_lineGetDevCaps 함수는 지정된 줄 디바이스를 쿼리하여 전화 통신 기능을 확인합니다. 반환된 정보는 줄 디바이스의 모든 주소에 유효합니다.
TSPI_lineGetDevConfig TSPI_lineGetDevConfig 함수는 줄(서비스 공급자) 및 디바이스 클래스와 관련된 데이터 구조 개체를 반환하여 줄 디바이스와 일대일로 연결된 디바이스의 현재 구성을 제공합니다.
TSPI_lineGetExtensionID TSPI_lineGetExtensionID 함수는 서비스 공급자가 표시된 줄 디바이스에 대해 지원하는 확장 식별자를 반환합니다.
TSPI_lineGetIcon TSPI_lineGetIcon 함수는 사용자에게 표시할 서비스 라인 디바이스별 아이콘을 검색합니다.
TSPI_lineGetID TSPI_lineGetID 함수는 선택한 줄, 주소 또는 호출과 연결된 지정된 디바이스 클래스에 대한 디바이스 식별자를 반환합니다.
TSPI_lineGetLineDevStatus TSPI_lineGetLineDevStatus 함수는 현재 상태 대해 지정된 열린 선 디바이스를 쿼리합니다. 반환된 정보는 줄의 모든 주소에 전역입니다.
TSPI_lineGetNumAddressIDs TSPI_lineGetNumAddressIDs 함수는 표시된 줄에서 지원되는 주소 식별자 수를 검색합니다.
TSPI_lineHold TSPI_lineHold 함수는 지정된 호출을 보류합니다.
TSPI_lineMakeCall TSPI_lineMakeCall 함수는 지정된 줄에서 지정된 대상 주소에 대한 호출을 배치합니다. 필요에 따라 기본 호출 설정 매개 변수를 제외한 모든 항목이 요청되면 호출 매개 변수를 지정할 수 있습니다.
TSPI_lineMonitorDigits

TSPI_lineMonitorDigits 함수는 호출에서 수신된 숫자의 버퍼되지 않은 검색을 사용하거나 사용하지 않도록 설정합니다.
TSPI_lineMonitorMedia TSPI_lineMonitorMedia 함수는 지정된 호출에서 미디어 유형 검색을 사용하거나 사용하지 않도록 설정합니다. 미디어 유형이 검색되면 LINE_MONITORMEDIA 메시지가 TAPI로 전송됩니다.
TSPI_lineMonitorTones TSPI_lineMonitorTones 함수는 호출에서 인밴드 톤 검색을 활성화하고 사용하지 않도록 설정합니다. 지정된 톤이 검색될 때마다 TAPI를 통해 클라이언트 애플리케이션에 메시지가 전송됩니다.
TSPI_lineMSPIdentify TSPI_lineMSPIdentify 함수는 모든 줄에 대해 연결된 MSP CLSID를 결정합니다. 이 함수에는 TAPI 3.0 버전 협상이 필요합니다.
TSPI_lineNegotiateExtVersion TSPI_lineNegotiateExtVersion 함수는 가능한 확장 버전 범위가 지정된 경우 서비스 공급자가 이 디바이스에 대해 작동할 수 있는 가장 높은 확장 버전 번호를 반환합니다.
TSPI_lineNegotiateTSPIVersion TSPI_lineNegotiateTSPIVersion 함수는 가능한 SPI 버전의 범위를 감안할 때 서비스 공급자가 이 디바이스에 대해 작동할 수 있는 가장 높은 SPI 버전을 반환합니다.
TSPI_lineOpen TSPI_lineOpen 함수는 디바이스 식별자가 지정된 줄 디바이스를 열고 디바이스에 대한 서비스 공급자의 핸들을 반환합니다.
TSPI_linePark TSPI_linePark 함수는 지정된 대기 모드에 따라 지정된 호출을 주차합니다.
TSPI_linePickup TSPI_linePickup 함수는 지정된 대상 주소에서 호출 경고를 선택하고 선택한 호출에 대한 호출 핸들을 반환합니다.
TSPI_linePrepareAddToConference TSPI_linePrepareAddToConference 함수는 다른 파티의 추가를 위해 기존 전화 회의를 준비합니다. 그것은 새로운, 임시 상담 전화를 만듭니다. 새로운 상담 전화는 이후에 컨퍼런스 콜에 추가할 수 있습니다.
TSPI_lineReceiveMSPData

TSPI_lineReceiveMSPData 함수 서비스 공급자는 MSP(미디어 서비스 공급자)가 보낸 데이터를 받습니다. 이 함수에는 TAPI 3.0 버전 협상이 필요합니다.
TSPI_lineRedirect TSPI_lineRedirect 함수는 지정된 제품 호출을 지정된 대상 주소로 리디렉션합니다.
TSPI_lineReleaseUserUserInfo TSPI_lineReleaseUserUserInfo 함수는 LINECALLINFO 구조에 포함된 사용자-사용자 정보가 처리되었으며 이후에 수신된 사용자-사용자 정보를 해당 구조에 기록할 수 있음을 서비스 공급자에게 알릴 수 있습니다.
TSPI_lineRemoveFromConference TSPI_lineRemoveFromConference 함수는 현재 속한 전화 회의 호출에서 지정된 호출을 제거합니다. 전화 회의의 나머지 통화는 영향을 받지 않습니다.
TSPI_lineSecureCall TSPI_lineSecureCall 함수는 호출의 미디어 스트림에 영향을 줄 수 있는 중단 또는 간섭으로부터 호출을 보호합니다.
TSPI_lineSelectExtVersion TSPI_lineSelectExtVersion 함수는 표시된 줄 디바이스에 대해 표시된 확장 버전을 선택합니다. 후속 요청은 해당 확장 버전에 따라 작동합니다.
TSPI_lineSendUserUserInfo TSPI_lineSendUserUserInfo 함수는 지정된 호출에서 원격 당사자에게 사용자-사용자 정보를 보냅니다.
TSPI_lineSetAppSpecific TSPI_lineSetAppSpecific 함수는 지정된 호출의 LINECALLINFO 구조체의 애플리케이션별 필드를 설정합니다.
TSPI_lineSetCallData TSPI_lineSetCallData 함수 서비스 공급자는 호출과 관련된 정보와 함께 표시된 호출 데이터를 저장한 후 TSPI_lineGetCallInfo 호출될 때마다 전달합니다.
TSPI_lineSetCallHubTracking TSPI_lineSetCallHubTracking 함수는 CallHub 추적 모드를 설정합니다. 이 함수에는 TAPI 3.0 버전 협상이 필요합니다.
TSPI_lineSetCallParams

TSPI_lineSetCallParams 함수는 기존 호출에 대한 특정 매개 변수를 설정합니다.
TSPI_lineSetCallQualityOfService TSPI_lineSetCallQualityOfService 함수 서비스 공급자는 스위치를 사용하여 호출 시 QOS를 재협상하려고 시도합니다. 원하는 QOS를 사용할 수 없는 경우 함수가 실패하지만 호출은 이전 QOS에서 계속됩니다.
TSPI_lineSetCallTreatment TSPI_lineSetCallTreatment 함수 서비스 공급자는 표시된 dwCallTreatment를 LINECALLINFO에 저장하고 업데이트된 정보를 나타내는 LINE_CALLINFO 메시지를 보냅니다.
TSPI_lineSetDefaultMediaDetection TSPI_lineSetDefaultMediaDetection 프로시저는 표시된 줄에 대해 검색할 새로운 미디어 유형 집합을 서비스 공급자에게 알려줍니다(이전 집합을 대체).
TSPI_lineSetDevConfig TSPI_lineSetDevConfig 함수는 이전에 TSPI_lineGetDevConfig 사용하여 가져온 데이터 구조에서 라인 디바이스와 일대일로 연결된 디바이스의 구성을 복원합니다.
TSPI_lineSetLineDevStatus TSPI_lineSetLineDevStatus 서비스 공급자는 디바이스 상태 표시된 대로 설정하여 새 상태 나타내는 적절한 LINE_LINEDEVSTATE 메시지를 보냅니다.
TSPI_lineSetMediaControl TSPI_lineSetMediaControl 함수는 지정된 줄, 주소 또는 호출과 연결된 미디어 스트림에서 제어 작업을 사용하거나 사용하지 않도록 설정합니다.
TSPI_lineSetMediaMode TSPI_lineSetMediaMode 함수는 호출의 LINECALLINFO 구조에 저장된 대로 호출의 미디어를 변경합니다.
TSPI_lineSetStatusMessages TSPI_lineSetStatusMessages 함수를 사용하면 TAPI가 지정된 줄 또는 해당 주소에 대한 상태 변경 내용과 관련된 이벤트에 대해 서비스 공급자가 생성해야 하는 알림 메시지를 지정할 수 있습니다.
TSPI_lineSetTerminal TSPI_lineSetTerminal 함수를 사용하면 TAPI가 지정된 줄, 주소 또는 호출과 관련된 터미널 정보를 라우팅할 터미널 정보를 지정할 수 있습니다.
TSPI_lineSetupConference

TSPI_lineSetupConference 함수는 타사 추가를 위한 전화 회의를 설정합니다.
TSPI_lineSetupTransfer TSPI_lineSetupTransfer 함수는 hdCall에서 지정한 호출의 전송을 시작합니다. 그것은 상담 전화를 설정합니다, lphdConsultCall, 있는 파티는 전송의 대상이 될 수있는 전화를 걸 수 있습니다.
TSPI_lineSwapHold TSPI_lineSwapHold 함수는 지정된 활성 통화를 상담 보류 중 지정된 통화로 바꿉니다.
TSPI_lineUncompleteCall TSPI_lineUncompleteCall 함수는 지정된 줄에서 지정된 호출 완료 요청을 취소하는 데 사용됩니다.
TSPI_lineUnhold TSPI_lineUnhold 함수는 지정된 보류된 호출을 검색합니다.
TSPI_lineUnpark TSPI_lineUnpark 함수는 지정된 주소에 대기된 호출을 검색하고 해당 호출 핸들을 반환합니다.
TSPI_phoneClose TSPI_phoneClose 함수는 디바이스에서 모든 미해결 비동기 작업을 완료하거나 중단한 후 지정된 열린 휴대폰 디바이스를 닫습니다.
TSPI_phoneConfigDialog TSPI_phoneConfigDialog 함수는 사용되지 않습니다. TAPI 버전 1.4 이하 서비스 공급자는 이 TSPI 함수를 구현할 수 있습니다. TAPI 버전 2.0 이상 TSP는 TUISPI_phoneConfigDialog 구현합니다.
TSPI_phoneDevSpecific TSPI_phoneDevSpecific 함수는 다른 작업에 설명되지 않은 기능을 제공하기 위해 전화 통신 API 구현을 사용하도록 설정하는 일반 확장 메커니즘으로 사용됩니다. 이러한 확장의 의미는 디바이스별로 다릅니다.
TSPI_phoneGetButtonInfo TSPI_phoneGetButtonInfo 함수는 지정된 단추에 대한 정보를 반환합니다.
TSPI_phoneGetData TSPI_phoneGetData 함수는 열린 휴대폰 디바이스의 지정된 위치에서 지정된 버퍼로 정보를 업로드합니다.
TSPI_phoneGetDevCaps

TSPI_phoneGetDevCaps 함수는 지정된 전화 장치를 쿼리하여 전화 통신 기능을 확인합니다.
TSPI_phoneGetDisplay TSPI_phoneGetDisplay 함수는 지정된 휴대폰 디스플레이의 현재 내용을 반환합니다.
TSPI_phoneGetExtensionID TSPI_phoneGetExtensionID 함수는 서비스 공급자가 표시된 전화 디바이스에 대해 지원하는 확장 식별자를 검색합니다.
TSPI_phoneGetGain TSPI_phoneGetGain 함수는 지정된 휴대폰의 후크위치 디바이스 마이크의 게인 설정을 반환합니다.
TSPI_phoneGetHookSwitch TSPI_phoneGetHookSwitch 함수는 지정된 열린 휴대폰 디바이스의 현재 후크위치 모드를 반환합니다.
TSPI_phoneGetIcon TSPI_phoneGetIcon 함수는 사용자에게 표시할 서비스 전화 디바이스별(또는 공급자별) 아이콘을 검색합니다.
TSPI_phoneGetID TSPI_phoneGetID 함수는 지정된 휴대폰 디바이스와 연결된 지정된 디바이스 클래스에 대한 디바이스 식별자를 반환합니다.
TSPI_phoneGetLamp TSPI_phoneGetLamp 함수는 지정된 램프의 현재 램프 모드를 반환합니다.
TSPI_phoneGetRing TSPI_phoneGetRing 함수를 사용하면 애플리케이션이 현재 링 모드로 지정된 개방형 휴대폰 디바이스를 쿼리할 수 있습니다.
TSPI_phoneGetStatus TSPI_phoneGetStatus 함수는 전체 상태 대해 지정된 개방형 휴대폰 디바이스를 쿼리합니다.
TSPI_phoneGetVolume TSPI_phoneGetVolume 함수는 지정된 휴대폰의 후크위치 디바이스의 볼륨 설정을 반환합니다.
TSPI_phoneNegotiateExtVersion

TSPI_phoneNegotiateExtVersion 함수는 가능한 확장 버전의 범위를 감안할 때 서비스 공급자가 이 디바이스에 대해 작동할 수 있는 가장 높은 확장 버전 번호를 반환합니다.
TSPI_phoneNegotiateTSPIVersion TSPI_phoneNegotiateTSPIVersion 함수는 가능한 SPI 버전의 범위를 감안할 때 서비스 공급자가 이 디바이스에 대해 작동할 수 있는 가장 높은 SPI 버전을 반환합니다.
TSPI_phoneOpen TSPI_phoneOpen 함수는 디바이스 식별자가 지정된 휴대폰 디바이스를 열고, 디바이스에 대한 서비스 공급자의 불투명 핸들을 반환하고, PHONEEVENT 프로시저에 대한 후속 호출에 사용할 디바이스에 대한 TAPI의 불투명 핸들을 유지합니다.
TSPI_phoneSelectExtVersion TSPI_phoneSelectExtVersion 함수는 표시된 휴대폰 디바이스에 대해 표시된 확장 버전을 선택합니다. 후속 요청은 해당 확장 버전에 따라 작동합니다.
TSPI_phoneSetButtonInfo TSPI_phoneSetButtonInfo 함수는 지정된 휴대폰의 지정된 단추에 대한 정보를 설정합니다.
TSPI_phoneSetData TSPI_phoneSetData 함수는 지정된 버퍼의 정보를 선택한 데이터 식별자의 열린 휴대폰 디바이스에 다운로드합니다.
TSPI_phoneSetDisplay TSPI_phoneSetDisplay 함수를 사용하면 지정된 열린 휴대폰 디바이스에 지정된 문자열이 표시됩니다.
TSPI_phoneSetGain TSPI_phoneSetGain 함수는 지정된 후크위치 디바이스의 마이크 게인을 지정된 게인 수준으로 설정합니다.
TSPI_phoneSetHookSwitch TSPI_phoneSetHookSwitch 함수는 지정된 열린 휴대폰의 후크위치 디바이스의 후크 상태를 지정된 모드로 설정합니다. 나열된 후크위치 디바이스의 hookswitch 상태만 영향을 받습니다.
TSPI_phoneSetLamp TSPI_phoneSetLamp 함수는 지정된 램프 모드에서 지정된 열린 휴대폰 디바이스에 지정된 램프를 설정합니다.
TSPI_phoneSetRing

TSPI_phoneSetRing 함수는 지정된 링 모드 및 볼륨을 사용하여 지정된 열린 휴대폰 디바이스를 울립니다.

TSPI_phoneSetStatusMessages

TSPI_phoneSetStatusMessages 함수를 사용하면 서비스 공급자가 현재 관심 없는 상태 메시지를 애플리케이션에 필터링합니다.

TSPI_phoneSetVolume

TSPI_phoneSetVolume 함수는 지정된 후크위치 디바이스의 스피커 구성 요소 볼륨을 지정된 수준으로 설정합니다.

TSPI_providerConfig

TSPI_providerConfig 함수는 사용되지 않습니다. TAPI 버전 1.4 이하 서비스 공급자는 이 TSPI 함수를 구현할 수 있습니다. TAPI 버전 2.0 이상 TSP는 TUISPI_providerConfig 구현합니다.

TSPI_providerCreateLineDevice

TSPI_providerCreateLineDevice 함수는 서비스 공급자로부터 LINE_CREATE 메시지를 수신하여 TAPI에 의해 호출되며, 이를 통해 새 줄 디바이스를 동적으로 만들 수 있습니다.

TSPI_providerCreatePhoneDevice

TSPI_providerCreatePhoneDevice 함수는 서비스 공급자로부터 PHONE_CREATE 메시지를 수신하여 새 휴대폰 디바이스를 동적으로 만들 수 있도록 TAPI에 의해 호출됩니다.

TSPI_providerEnumDevices

TAPI는 TSPI_providerInit 전에 TSPI_providerEnumDevices 함수를 호출하여 서비스 공급자가 지원하는 회선 및 전화 디바이스의 수를 확인합니다.

TSPI_providerFreeDialogInstance

TSPI_providerFreeDialogInstance 함수는 hdDlgInst와 연결된 대화 상자가 종료되었음을 서비스 공급자에게 알릴 수 있습니다.

TSPI_providerGenericDialogData

TSPI_providerGenericDialogData 함수는 TUISPIDLLCALLBACK 함수를 통해 애플리케이션 컨텍스트에서 실행되는 UI DLL에서 전송된 서비스 공급자 데이터를 제공합니다.

TSPI_providerInit

TSPI_providerInit 함수는 서비스 공급자를 초기화하고 후속 작업에 필요한 매개 변수를 제공합니다.

TSPI_providerInstall

TSPI_providerInstall 함수는 사용되지 않습니다. TAPI 버전 1.4 이하 서비스 공급자는 이 TSPI 함수를 구현할 수 있습니다. TAPI 버전 2.0 이상 TSP는 TUISPI_providerInstall 구현합니다.
TSPI_providerRemove TSPI_providerRemove 함수는 사용되지 않습니다. TAPI 버전 1.4 이하 서비스 공급자는 이 TSPI 함수를 구현할 수 있습니다. TAPI 버전 2.0 이상 TSP는 TUISPI_providerRemove 구현합니다.
TSPI_providerShutdown TSPI_providerShutdown 함수는 서비스 공급자를 종료합니다. 서비스 공급자는 진행 중인 모든 활동을 종료하고 할당된 리소스를 해제합니다.
TSPI_providerUIIdentify TSPI_providerUIIdentify 함수는 서비스 공급자에서 정규화된 경로를 추출하여 서비스 공급자의 UI DLL 구성 요소를 로드합니다.
TUISPI_lineConfigDialog TUISPI_lineConfigDialog 함수를 사용하면 지정된 줄 디바이스의 공급자가 모달 대화 상자를 hwndOwner의 자식 창으로 표시하여 사용자가 줄 디바이스와 관련된 매개 변수를 구성할 수 있도록 합니다.
TUISPI_lineConfigDialogEdit TUISPI_lineConfigDialogEdit 함수를 사용하면 지정된 줄 디바이스의 공급자가 모달 대화 상자를 hwndOwner의 자식 창으로 표시하여 사용자가 줄 디바이스와 관련된 매개 변수를 구성할 수 있도록 합니다.
TUISPI_phoneConfigDialog TUISPI_phoneConfigDialog 함수를 사용하면 지정된 휴대폰 디바이스 공급자가 모달 대화 상자를 hwndOwner의 자식 창으로 표시하여 사용자가 휴대폰 디바이스와 관련된 매개 변수를 구성할 수 있도록 합니다.
TUISPI_providerConfig TUISPI_providerConfig 함수는 호출 애플리케이션의 컨텍스트에서 실행해야 하는 UI 요소를 구현합니다. 이 함수는 버전 2.0 이상에서 TSPI_providerConfig 함수를 더 이상 사용되지 않습니다(버전 1.4 이하에서 지원됨).
TUISPI_providerGenericDialog 서비스 공급자가 LINE_CREATEDIALOGINSTANCE 메시지를 보낼 때 UI DLL의 TUISPI_providerGenericDialog 함수가 호출됩니다.
TUISPI_providerGenericDialogData

서비스 공급자가 LINE_SENDDIALOGINSTANCEDATA 메시지를 보낼 때 UI DLL의 TUISPI_providerGenericDialogData 함수가 호출됩니다.
TUISPI_providerInstall TUISPI_providerInstall 함수의 구현은 서비스 공급자가 공급자의 추가 "조각"을 올바른 디렉터리에 설치하고 공급자에게 필요한 레지스트리 항목을 설정할 수 있는 기회입니다.
TUISPI_providerRemove TUISPI_providerRemove 함수는 사용자에게 서비스 공급자 제거를 확인하도록 요청합니다. 이 함수는 버전 2.0 이상에서 TSPI_providerRemove 함수를 더 이상 사용되지 않습니다(버전 1.4 이하에서 지원됨).

콜백 함수

ASYNC_COMPLETION Completion_Proc TAPI에서 구현하고 서비스 공급자에게 TSPI_providerInit 매개 변수로 제공하는 콜백 함수입니다.
LINEEVENT Line_Event TAPI에서 구현하고 서비스 공급자에게 TSPI_lineOpen 매개 변수로 제공하는 콜백 함수입니다. 서비스 공급자는 이 함수를 호출하여 줄 또는 줄의 호출에서 발생하는 이벤트를 보고합니다.
PHONEEVENT TAPI에서 구현하고 서비스 공급자에게 TSPI_phoneOpen 매개 변수로 제공하는 콜백 함수를 Phone_Event. 서비스 공급자는 이 함수를 호출하여 휴대폰에서 발생하는 이벤트를 보고합니다.
TUISPIDLLCALLBACK DllCallbackProc 함수는 UI DLL에 의해 호출되어 프라이빗 매개 변수 블록을 서비스 공급자에게 보냅니다.

구조체

TUISPICREATEDIALOGINSTANCEPARAMS

TUISPI 데이터 구조는 아래에 정의되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

tspubplugin2com.h 헤더

아티클2023. 08. 26.

이 헤더는 원격 데스크톱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 데스크톱 서비스

tspubplugin2com.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ItsPubPlugin2 RemoteApp 및 데스크톱 연결 사용자가 사용할 수 있는 리소스에 대한 정보를 제공하는 메서드를 지정합니다.

구조체

pluginResource2 RemoteApp 및 데스크톱 연결에서 사용자에게 할당할 수 있는 리소스에 대한 추가 정보를 포함합니다.
pluginResource2FileAssociation RemoteApp 및 데스크톱 연결의 파일 연결에 대한 정보를 포함합니다.

열거형

TSPUB_PLUGIN_PD_ASSIGNMENT_TYPE 개인 데스크톱 해상도에 대한 할당 유형을 지정합니다.
TSPUB_PLUGIN_PD_RESOLUTION_TYPE 요청되는 개인 데스크톱 해상도의 유형을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

tspubplugincom.h 헤더

아티클2023. 08. 26.

이 헤더는 원격 데스크톱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 데스크톱 서비스

tspubplugincom.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ItsPubPlugin RemoteApp 및 데스크톱 연결 사용자가 사용할 수 있는 리소스에 대한 정보를 제공하는 속성 및 메서드를 노출합니다.

구조체

pluginResource RemoteApp 및 데스크톱 연결에서 사용자에게 할당할 수 있는 리소스에 대한 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

tssbx.h 헤더

이 헤더는 원격 데스크톱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 데스크톱 서비스

tssbx.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Interfaces

테이블 확장

IWTSSBPlugin 터미널 서비스 세션 브로커(TS 세션 브로커)의 기능을 확장하는 데 사용됩니다. TS 세션 브로커의 리디렉션 논리를 재정의하는 플러그 인을 제공하려는 경우 이 인터페이스를 구현합니다.

구조

테이블 확장

WTSSBX_IP_ADDRESS 네트워크 리소스의 IP 주소에 대한 정보를 포함합니다.
WTSSBX_MACHINE_CONNECT_INFO 원격 연결을 허용하는 컴퓨터에 대한 정보를 포함합니다.
WTSSBX_MACHINE_INFO 컴퓨터 및 현재 상태에 대한 정보를 포함합니다.
WTSSBX_SESSION_INFO RD 연결 브로커(원격 데스크톱 연결 브로커)에서 사용할 수 있는 세션에 대한 정보를 포함합니다.

열거 목록

테이블 확장

WTSSBX_ADDRESS_FAMILY 리디렉션에 사용되는 네트워크 주소의 주소 패밀리를 나타내는 값을 포함합니다.
WTSSBX_MACHINE_DRAIN RD 세션 호스트(원격 데스크톱 세션 호스트) 서버의 드레이닝 상태를 나타내는 값을 포함합니다.
WTSSBX_MACHINE_SESSION_MODE RD 세션 호스트(원격 데스크톱 세션 호스트) 서버의 세션 모드를 나타내는 값을 포함합니다.
WTSSBX_MACHINE_STATE 서버의 현재 상태를 나타내는 값을 포함합니다.
WTSSBX_NOTIFICATION_TYPE RD 세션 호스트(원격 데스크톱 세션 호스트) 서버 또는 사용자 세션에서 발생한 상태 변경 유형을 나타내는 값을 포함합니다.
WTSSBX_SESSION_STATE 사용자 세션의 연결 상태를 나타내는 값을 포함합니다.

Last updated on 2025. 10. 08.

tsuserex.h 헤더

아티클2023. 03. 08.

이 헤더는 원격 데스크톱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 데스크톱 서비스

tsuserex.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IADsTSUserEx 원격 데스크톱 서비스 사용자 속성을 검사하고 구성하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

tsvirtualchannels.h 헤더

아티클2023. 03. 08.

이 헤더는 원격 데스크톱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 데스크톱 서비스

tsvirtualchannels.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWTSBitmapRenderer 동적 가상 채널 플러그 인에서 비트맵을 렌더링하는 데 사용됩니다.
IWTSBitmapRendererCallback 동적 가상 채널 플러그 인은 렌더링 영역의 크기가 변경될 때 알림을 받도록 이 인터페이스를 구현합니다.
IWTSBitmapRenderService 이 서비스는 서버의 매핑된 창에 해당하는 클라이언트에서 시각적 매핑을 만드는 데 사용됩니다.
IWTSListener DVC(동적 가상 채널) 연결에 대한 각 수신기에 대한 구성 설정을 관리합니다.
IWTSListenerCallback 특정 수신기에서 들어오는 요청에 대해 RDC(원격 데스크톱 연결) 클라이언트 플러그 인에 알리는 데 사용됩니다.
IWTSPlugin RDC(원격 데스크톱 연결) 클라이언트 플러그 인을 RDC(원격 데스크톱 연결) 클라이언트에서 로드할 수 있습니다.
IWTSPluginServiceProvider 동적 가상 채널 플러그 인이 다양한 원격 데스크톱 클라이언트 서비스를 쿼리하는 방법을 제공합니다.

IWTSVirtualChannel 채널 상태를 제어하고 채널에 쓰는 데 사용됩니다.
IWTSVirtualChannelCallback 채널 상태 변경 내용 또는 수신된 데이터에 대한 알림을 받습니다.
IWTSVirtualChannelManager 모든 RDC(원격 데스크톱 연결) 클라이언트 플러그 인 및 DVC(동적 가상 채널) 수신기를 관리합니다.

구조체

BITMAP_RENDERER_STATISTICS RemoteFX 미디어 리디렉션 비트맵 렌더러에 대한 통계를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

tvout.h 헤더

아티클2023. 08. 26.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 디스플레이 디바이스 참조
- Windows GDI

tvout.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

VIDEOPARAMETERS 비디오 미니포트 드라이버는 IOCTL 요청이 IOCTL_VIDEO_HANDLE_VIDEOPARAMETERS 때 VIDEO_REQUEST_PACKET InputBuffer 멤버의 VIDEOPARAMETERS 구조에 대한 포인터를 받습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

txfw32.h 헤더

아티클2023. 08. 26.

이 헤더는 데이터 액세스 및 스토리지에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지

txfw32.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

TxfGetThreadMiniVersionForCreate 후속 만들기가 열려도록 설정된 MiniVersion을 반환합니다.
TxfLogCreateFileReadContext 복제 레코드를 읽는 데 사용할 컨텍스트를 만듭니다.
TxfLogCreateRangeReadContext 복제 레코드를 읽는 데 필요한 컨텍스트를 만듭니다.
TxfLogDestroyReadContext TxfLogCreateFileReadContext 함수에 의해 만들어진 읽기 컨텍스트를 닫습니다.
TxfLogReadRecords 로그에서 다시 실행 레코드를 읽습니다.
TxfSetThreadMiniVersionForCreate 후속 만들기를 열어야 하는 MiniVersion을 설정합니다.

구조체

TXF_ID 리소스 관리자의 컨텍스트 내의 고유 식별자를 나타냅니다.

TXF_LOG_RECORD_AFFECTED_FILE 트랜잭션의 영향을 받은 파일에 대한 정보를 포함합니다.
TXF_LOG_RECORD_BASE 기본 레코드 정보를 포함합니다.
TXF_LOG_RECORD_TRUNCATE 잘린 작업에 대한 레코드를 포함합니다.
TXF_LOG_RECORD_WRITE 쓰기 작업에 대한 레코드를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

txlogpub.h 헤더

아티클2023. 08. 26.

이 헤더는 COM(구성 요소 개체 모델)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)

txlogpub.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IFileBasedLogInit ILog의 파일 기반 구현 instance 초기화합니다.
Ilog 일반 하위 수준 로깅 기능을 제공합니다.

열거형

RECORD_READING_POLICY 로그에서 레코드를 읽을 순서에 대한 힌트를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ual.h 헤더

아티클2023. 03. 08.

이 헤더는 사용자 액세스 로깅에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 사용자 액세스 로깅

ual.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

UalInstrument UAL_DATA_BLOB 구조체의 정보를 사용하여 지정된 데이터를 UAL(사용자 액세스 로깅) 프레임워크에 기록합니다.
UalRegisterProduct UAL(사용자 액세스 로깅)에 제품을 등록합니다.
UalStart UAL(사용자 액세스 로깅) 세션을 시작합니다.
UalStop UAL(사용자 액세스 로깅) 세션을 중지합니다.

구조체

UAL_DATA_BLOB UAL(사용자 액세스 로깅) 세션에 대한 정보를 지정합니다.

udpmib.h 헤더

아티클2023. 08. 26.

이 헤더는 관리 정보 베이스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 관리 정보 기반

udpmib.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

MIB_UDP6ROW 로컬 컴퓨터의 IPv6에 대한 UDP(사용자 데이터그램 프로토콜) 수신기 테이블의 항목을 포함합니다.
MIB_UDP6ROW_OWNER_MODULE 로컬 컴퓨터의 IPv6에 대한 UDP(사용자 데이터그램 프로토콜) 수신기 테이블의 항목을 포함합니다. 이 항목에는 UDP 엔드포인트에 대한 bind 함수 호출을 실행한 모든 사용 가능한 소유권 데이터 및 PID(프로세스 ID)도 포함됩니다.
MIB_UDP6ROW_OWNER_PID 로컬 컴퓨터의 IPv6에 대한 UDP(사용자 데이터그램 프로토콜) 수신기 테이블의 항목을 포함합니다. 항목에는 UDP 엔드포인트에 대한 바인딩 함수 호출을 실행한 PID(프로세스 ID)도 포함됩니다.
MIB_UDP6TABLE 로컬 컴퓨터의 IPv6에 대한 UDP(사용자 데이터그램 프로토콜) 수신기 테이블을 포함합니다.
MIB_UDP6TABLE_OWNER_MODULE 로컬 컴퓨터의 IPv6에 대한 UDP(사용자 데이터그램 프로토콜) 수신기 테이블을 포함합니다. 또한 테이블에는 사용 가능한 모든 소유권 데이터와 각 UDP 엔드포인트에 대한 bind 함수 호출을 실행한 PID(프로세스 ID)가 포함됩니다.
MIB_UDP6TABLE_OWNER_PID 로컬 컴퓨터의 IPv6에 대한 UDP(사용자 데이터그램 프로토콜) 수신기 테이블을 포함합니다. 테이블에는 각 UDP 엔드포인트에 대한 bind 함수 호출을 실행한 PID(프로세스 ID)도 포함되어 있습니다.
MIB_UDPROW

로컬 컴퓨터의 IPv4용 UDP(사용자 데이터그램 프로토콜) 수신기 테이블의 항목을 포함합니다.
MIB_UDPROW_OWNER_MODULE 로컬 컴퓨터의 IPv4 UDP(사용자 데이터그램 프로토콜) 수신기 테이블의 항목을 포함합니다. 이 항목에는 UDP 엔드포인트에 대한 bind 함수 호출을 실행한 모든 사용 가능한 소유권 데이터 및 PID(프로세스 ID)도 포함됩니다.
MIB_UDPROW_OWNER_PID 로컬 컴퓨터의 IPv4용 UDP(사용자 데이터그램 프로토콜) 수신기 테이블의 항목을 포함합니다. 항목에는 UDP 엔드포인트에 대한 바인딩 함수 호출을 실행한 PID(프로세스 ID)도 포함됩니다.
MIB_UDPSTATS 로컬 컴퓨터에서 실행되는 UDP(사용자 데이터그램 프로토콜)에 대한 통계를 포함합니다. (MIB_UDPSTATS)
MIB_UDPSTATS2 로컬 컴퓨터에서 실행되는 UDP(사용자 데이터그램 프로토콜)에 대한 통계를 포함합니다. (MIB_UDPSTATS2)
MIB_UDPTABLE 로컬 컴퓨터의 IPv4용 UDP(사용자 데이터그램 프로토콜) 수신기 테이블을 포함합니다.
MIB_UDPTABLE_OWNER_MODULE 로컬 컴퓨터의 IPv4용 UDP(사용자 데이터그램 프로토콜) 수신기 테이블을 포함합니다. 또한 테이블에는 사용 가능한 모든 소유권 데이터와 각 UDP 엔드포인트에 대한 bind 함수 호출을 실행한 PID(프로세스 ID)가 포함됩니다.
MIB_UDPTABLE_OWNER_PID 로컬 컴퓨터의 IPv4용 UDP(사용자 데이터그램 프로토콜) 수신기 테이블을 포함합니다. 테이블에는 각 UDP 엔드포인트에 대한 bind 함수 호출을 실행한 PID(프로세스 ID)도 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

uianimation.h 헤더

아티클2023. 08. 26.

이 헤더는 Windows 애니메이션 관리자에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 애니메이션 관리자

uianimation.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IUIAnimationInterpolator 사용자 지정 보간기를 만드는 메서드를 정의합니다.
IUIAnimationInterpolator2 사용자 지정 보간기를 만들기 위한 메서드를 정의하는 IUIAnimationInterpolator 인터페이스를 확장합니다. IUIAnimationInterpolator2는 지정된 차원의 보간을 지원합니다.
IUIAnimationLoopIterationChangeHandler2 스토리보드 루프 반복 이벤트를 처리하는 메서드를 정의합니다.
IUIAnimationManager 애니메이션을 만들고 관리하기 위한 중앙 인터페이스를 제공하는 애니메이션 관리자를 정의합니다.
IUIAnimationManager2 여러 차원에서 애니메이션을 만들고 관리하기 위한 중앙 인터페이스를 제공하는 애니메이션 관리자를 정의합니다.
IUIAnimationManagerEventHandler 애니메이션 관리자에 대한 상태 업데이트를 처리하는 메서드를 정의합니다.
IUIAnimationManagerEventHandler2 애니메이션 관리자에 대한 업데이트를 처리하는 메서드를 정의합니다.
IUIAnimationPrimitiveInterpolation

사용자 지정 보간기가 입방 다항 곡선 형식의 전환 정보를 애니메이션 관리자로 제공할 수 있도록 하는 메서드를 정의합니다.
IUIAnimationPriorityComparison 애니메이션 관리자가 일정 충돌을 resolve 데 사용하는 우선 순위 비교를 위한 메서드를 정의합니다.
IUIAnimationPriorityComparison2 우선 순위 비교를 통해 일정 충돌을 해결하는 메서드를 정의합니다.
IUIAnimationStoryboard 서로 상대적으로 동기화되는 전환 그룹을 포함하는 스토리보드를 정의합니다.
IUIAnimationStoryboard2 서로 상대적으로 동기화되는 전환 그룹을 포함하는 스토리보드를 정의합니다. 이 섹션에서 TopicDescriptionAbandon MethodTerminates the storyboard, release all related animation variables, and remove the storyboard from the schedule. AddKeyframeAfterTransition 메서드 지정된 전환이 끝날 때 키 프레임을 추가합니다. AddKeyframeAtOffset 메서드 기존 키 프레임에서 지정된 오프셋의 키 프레임을 추가합니다. AddTransition 메서드 스토리보드로의 전환을 추가합니다. AddTransitionAtKeyframe 메서드 지정된 키프레임에서 시작하는 전환을 추가합니다. AddTransitionBetweenKeyframes 메서드 두 키 프레임 간의 전환을 추가합니다. Conclude Method 진행 중인 키 프레임 루프의 현재 반복(루프가 UI_ANIMATION_REPEAT_INDEFINITELY 설정됨)을 완료하고, 루프를 종료하고, 스토리보드를 계속 진행합니다. Finish Method 지정한 시간 내에 스토리보드를 정의하여 필요한 경우 스토리보드를 압축합니다. GetElapsedTime MethodGets 스토리보드 재생을 시작한 이후 경과된 시간입니다. GetStatus 메서드스토리보드의 상태 설정합니다. GetTag 메서드 스토리보드에 대한 태그를 설정합니다. HoldVariable MethodDirects storyboard는 스토리보드가 끝날 때까지 지정된 애니메이션 변수를 최종 값으로 유지합니다. RepeatBetweenKeyframes 메서드 두 키 프레임 간에 루프를 만듭니다. Schedule MethodDirects the storyboard to scheduleself for play. SetSkipDuration 메서드 애니메이션을 시작할 스토리보드의 시작 부분에서 오프셋을 지정합니다. SetLongestAcceptableDelay 메서드 예약된 스토리보드가 시작되기 전에 허용되는 가장 긴 지연 시간을 설정합니다. SetStoryboardEventHandler 메서드스토리보드 이벤트에 대한 처리기를 지정합니다. SetTag 메서드스토리보드의 태그를 설정합니다. .
IUIAnimationStoryboardEventHandler 스토리보드에 대한 상태 및 업데이트 이벤트를 처리하는 메서드를 정의합니다.
IUIAnimationStoryboardEventHandler2 스토리보드 이벤트를 처리하는 메서드를 정의합니다.
IUIAnimationTimer 애니메이션 타이밍을 관리하기 위한 서비스를 제공하는 애니메이션 타이머를 정의합니다.

IUIAnimationTimerClientEventHandler 타이머 클라이언트 상태 변경 내용과 관련된 이벤트를 처리하는 메서드를 정의합니다.
IUIAnimationTimerEventHandler 타이밍 이벤트를 처리하는 메서드를 정의합니다.
IUIAnimationTimerUpdateHandler 타이밍 업데이트 이벤트를 처리하는 메서드를 정의합니다.
IUIAnimationTransition 시간이 지남에 따라 애니메이션 변수가 변경되는 방식을 결정하는 전환을 정의합니다.
IUIAnimationTransition2 전환을 정의하는 IUIAnimationTransition 인터페이스를 확장합니다. IUIAnimationTransition2 전환은 지정된 차원의 시간에 따라 애니메이션 변수가 변경되는 방식을 결정합니다.
IUIAnimationTransitionFactory 사용자 지정 보간기에서 전환을 만드는 메서드를 정의합니다.
IUIAnimationTransitionFactory2 사용자 지정 보간기에서 전환을 만드는 메서드를 정의합니다. 는 지정된 차원의 전환 생성을 지원합니다.
IUIAnimationTransitionLibrary 표준 전환 라이브러리를 정의합니다.
IUIAnimationTransitionLibrary2 지정된 차원에 대한 표준 전환 라이브러리를 정의합니다.
IUIAnimationVariable 애니메이션할 수 있는 시각적 요소를 나타내는 애니메이션 변수를 정의합니다.
IUIAnimationVariable2 여러 차원에서 애니메이션할 수 있는 시각적 요소를 나타내는 애니메이션 변수를 정의합니다.
IUIAnimationVariableChangeHandler 애니메이션 변수 업데이트와 관련된 이벤트를 처리하는 메서드를 정의합니다.

IUIAnimationVariableChangeHandler2 애니메이션 변수 업데이트 이벤트를 처리하는 메서드를 정의합니다. IUIAnimationVariableChangeHandler2는 지정된 차원에서 발생하는 이벤트를 처리합니다.
IUIAnimationVariableCurveChangeHandler2 애니메이션 곡선 업데이트 이벤트를 처리하는 메서드를 정의합니다.
IUIAnimationVariableIntegerChangeHandler 애니메이션 변수 업데이트 이벤트를 처리하는 메서드를 정의합니다.
IUIAnimationVariableIntegerChangeHandler2 애니메이션 변수 업데이트 이벤트를 처리하는 메서드를 정의합니다. IUIAnimationVariableIntegerChangeHandler2는 지정된 차원에서 발생하는 이벤트를 처리합니다.

구조체

__MIDL___MIDL_itf_UIAnimation_0000_0002_0003 스토리보드 내의 시간 오프셋을 나타내는 키 프레임을 정의합니다.

열거형

UI_ANIMATION_DEPENDENCIES 지정된 입력에 따라 보간기의 측면을 정의합니다.
UI_ANIMATION_IDLE_BEHAVIOR 애니메이션 관리자가 유휴 상태일 때 타이머의 동작을 정의합니다.
UI_ANIMATION_MANAGER_STATUS 애니메이션 관리자의 작업 상태 정의합니다.
UI_ANIMATION_MODE

애니메이션 모드를 정의합니다.
UI_ANIMATION_PRIORITY_EFFECT 우선 순위 비교가 false를 반환하는 경우 스토리보드에 미치는 잠재적 효과를 정의합니다.
UI_ANIMATION_REPEAT_MODE 루프 반복에 대한 패턴을 정의합니다.
UI_ANIMATION_ROUNDING_MODE 애니메이션 변수의 값이 부동 소수점 형식에서 정수 형식으로 변환될 때 사용할 반올림 모드를 정의합니다.
UI_ANIMATION_SCHEDULING_RESULT 스토리보드 일정에 대한 결과를 정의합니다.
UI_ANIMATION_SLOPE 애니메이션 기울기 특성을 정의합니다.
UI_ANIMATION_STORYBOARD_STATUS 스토리보드의 상태 정의합니다.
UI_ANIMATION_TIMER_CLIENT_STATUS 타이머 클라이언트에 대한 활동 상태 정의합니다.
UI_ANIMATION_UPDATE_RESULT 애니메이션 업데이트에 대한 결과를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

uiautomationclient.h 헤더

아티클2023. 08. 26.

이 헤더는 Windows 접근성 기능에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 접근성 기능

uiautomationclient.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IUIAutomation Microsoft UI 자동화 클라이언트 애플리케이션이 UI 자동화 요소를 검색, 액세스 및 필터링할 수 있도록 하는 메서드를 노출합니다.
IUIAutomation2 IUIAutomation 인터페이스를 확장하여 Microsoft UI 자동화 기능을 제어하는 추가 메서드를 노출합니다.
IUIAutomation3 IUIAutomation2 인터페이스를 확장하여 Microsoft UI 자동화 기능을 제어하는 추가 메서드를 노출합니다.
IUIAutomation4 IUIAutomation3 인터페이스를 확장하여 Microsoft UI 자동화 기능을 제어하는 추가 메서드를 노출합니다.
IUIAutomation5 IUIAutomation4 인터페이스를 확장하여 Microsoft UI 자동화 기능을 제어하는 추가 메서드를 노출합니다.
IUIAutomation6 IUIAutomation5 인터페이스를 확장하여 Microsoft UI 자동화 기능을 제어하는 추가 메서드를 노출합니다.
IUIAutomationActiveTextPositionChangedEventHandler 활성 텍스트 위치가 변경되면 발생하는 Microsoft UI 자동화 이벤트를 처리하는 메서드를 노출합니다.

IUIAutomationAndCondition Microsoft UI 자동화 클라이언트 애플리케이션이 AND 기반 속성 조건에 대한 정보를 검색하는 데 사용할 수 있는 속성 및 메서드를 노출합니다.
IUIAutomationAnnotationPattern 문서에서 주석의 속성에 대한 액세스를 제공합니다.
IUIAutomationBoolCondition TRUE(모든 요소 선택) 또는 FALSE(요소가 선택되지 않음)일 수 있는 조건을 나타냅니다.
IUIAutomationCacheRequest 캐시 요청의 속성 및 메서드를 노출합니다. 클라이언트 애플리케이션은 이 인터페이스를 사용하여 Microsoft UI 자동화 요소를 가져올 때 캐시할 속성 및 컨트롤 패턴을 지정합니다.
IUIAutomationChangesEventHandler 하나 이상의 Microsoft UI 자동화 변경 이벤트를 처리하는 메서드를 노출합니다.
IUIAutomationCondition UI 자동화 트리에서 요소를 검색할 때 필터링에 사용되는 조건에 대한 기본 인터페이스입니다.
IUIAutomationCustomNavigationPattern 사용자 지정 탐색 순서를 지원하는 컨트롤에 Microsoft UI 자동화 클라이언트의 액세스를 지원하는 메서드를 노출합니다.
IUIAutomationDockPattern 자식 요소를 서로 상대적으로 가로 및 세로로 정렬할 수 있도록 하는 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationDragPattern 끌어서 놓기 작업의 일부로 끌 수 있는 요소에 대해 UI 자동화 공급자가 노출하는 정보에 대한 액세스를 제공합니다.
IUIAutomationDropTargetPattern 끌어서 놓기 작업의 놓기 대상이 될 수 있는 요소에 대해 Microsoft UI 자동화 공급자가 노출하는 끌어서 놓기 정보에 대한 액세스를 제공합니다.
IUIAutomationElement UI 항목을 나타내는 UI 자동화 요소에 대한 메서드 및 속성을 노출합니다.

IUIAutomationElement2 IUIAutomationElement 인터페이스를 확장합니다.
IUIAutomationElement3 IUIAutomationElement2 인터페이스를 확장합니다.
IUIAutomationElement4 IUIAutomationElement3 인터페이스를 확장합니다.
IUIAutomationElement5 IUIAutomationElement4 인터페이스를 확장하여 현재 및 캐시된 랜드마크 데이터에 대한 액세스를 제공합니다.
IUIAutomationElement6 IUIAutomationElement5 인터페이스를 확장하여 현재 및 캐시된 전체 설명에 대한 액세스를 제공합니다.
IUIAutomationElement7 IUIAutomationElement6 인터페이스를 확장합니다.
IUIAutomationElement8 IUIAutomationElement7 인터페이스를 확장합니다.
IUIAutomationElement9 IUIAutomationElement8 인터페이스를 확장합니다.
IUIAutomationElementArray UI 자동화 요소의 컬렉션을 나타냅니다.
IUIAutomationEventHandler Microsoft UI 자동화 이벤트를 처리하는 메서드를 노출합니다.
IUIAutomationEventHandlerGroup IUIAutomation6에 정의된 CreateEventHandlerGroup 및 AddEventHandlerGroup 메서드를 통해 대량 등록을 위해 컬렉션에 하나 이상의 이벤트를 추가하는 메서드를 노출합니다.
IUIAutomationExpandCollapsePattern

시각적으로 확장하여 콘텐츠를 표시하고 축소하여 콘텐츠를 숨길 수 있는 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationFocusChangedEventHandler 키보드 포커스가 다른 UI 자동화 요소로 이동할 때 발생하는 이벤트를 처리하는 메서드를 노출합니다.
IUIAutomationGridItemPattern IUIAutomationGridPattern 인터페이스를 지원하는 그리드 스타일 컨테이너의 자식 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationGridPattern 행과 열로 트래버스할 수 있는 2차원 논리 좌표계로 구성된 자식 컨트롤 컬렉션의 컨테이너 역할을 하는 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationInvokePattern 클라이언트 애플리케이션이 컨트롤의 동작(일반적으로 단추)을 호출할 수 있도록 하는 메서드를 노출합니다.
IUIAutomationItemContainerPattern 가상 목록과 같은 컨테이너에서 항목을 검색하는 메서드를 노출합니다.
IUIAutomationLegacyIAccessiblePattern Microsoft UI 자동화 클라이언트가 MSAA(Microsoft Active Accessibility) 서버에서 UI 정보를 검색할 수 있도록 하는 메서드 및 속성을 노출합니다.
IUIAutomationMultipleViewPattern 동일한 정보 또는 자식 컨트롤 집합의 여러 표현 간에 전환할 수 있는 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationNotCondition 다른 조건의 음수인 조건을 나타냅니다.
IUIAutomationNotificationEventHandler Microsoft UI 자동화 알림 이벤트를 처리하는 메서드를 노출합니다.
IUIAutomationObjectModelPattern 컨트롤 또는 애플리케이션에서 구현하는 기본 개체 모델에 대한 액세스를 제공합니다. (IUIAutomationObjectModelPattern)

IUIAutomationOrCondition 여러 조건으로 구성된 조건을 나타내며, 그 중 하나 이상이 true여야 합니다.
IUIAutomationPropertyChangedEventHandler 속성이 변경되면 발생하는 Microsoft UI 자동화 이벤트를 처리하는 메서드를 노출합니다.
IUIAutomationPropertyCondition UI 자동화 요소를 찾는 데 사용되는 속성 값을 기반으로 하는 조건을 나타냅니다.
IUIAutomationProxyFactory UI 자동화에 대한 기본 지원이 없는 UI 요소에 대한 Microsoft UI 자동화 공급자를 만드는 개체의 속성과 메서드를 노출합니다. 이 인터페이스는 프록시에 의해 구현됩니다.
IUIAutomationProxyFactoryEntry Microsoft UI 자동화에서 유지 관리하는 테이블의 프록시 팩터리를 나타내며 클라이언트 애플리케이션에서 IUIAutomationProxyFactory 개체와 상호 작용하는 데 사용할 수 있는 속성과 메서드를 노출합니다.
IUIAutomationProxyFactoryMapping 프록시 팩터리 테이블에 대한 속성 및 메서드를 노출합니다. 각 테이블 항목은 IUIAutomationProxyFactoryEntry 인터페이스로 표시됩니다. 항목은 시스템에서 프록시를 사용하려고 시도하는 순서대로 표시됩니다.
IUIAutomationRangeValuePattern 값 범위를 제공하는 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationScrollItemPattern 스크롤 가능한 보기의 항목을 보기의 표시되는 부분에 배치할 수 있도록 하는 메서드를 노출합니다.
IUIAutomationScrollPattern 자식 요소 컬렉션에 대해 스크롤 가능한 컨테이너 역할을 하는 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationSelectionItemPattern IUIAutomationSelectionPattern을 지원하는 컨테이너 컨트롤의 선택 가능한 자식 항목에 대한 액세스를 제공합니다.

IUIAutomationSelectionPattern 선택 가능한 자식 항목이 포함된 컨트롤에 대한 액세스를 제공합니다. 이 요소의 자식은 IUIAutomationSelectionItemPattern을 지원합니다.
IUIAutomationSelectionPattern2 IUIAutomationSelectionPattern 인터페이스를 확장하여 선택한 항목에 대한 정보를 제공합니다.
IUIAutomationSpreadsheetItemPattern 클라이언트 애플리케이션이 스프레드시트의 항목(셀)에 대한 정보를 검색할 수 있도록 합니다.
IUIAutomationSpreadsheetPattern 클라이언트 애플리케이션이 스프레드시트의 항목(셀)에 액세스할 수 있도록 합니다.
IUIAutomationStructureChangedEventHandler Microsoft UI 자동화 트리 구조가 변경되면 발생하는 이벤트를 처리하는 메서드를 노출합니다.
IUIAutomationStylesPattern Microsoft UI 자동화 클라이언트가 문서의 요소와 연결된 비주얼 스타일을 검색할 수 있도록 합니다.
IUIAutomationSynchronizedInputPattern 컨트롤의 키보드 또는 마우스 입력에 대한 액세스를 제공합니다.
IUIAutomationTableItemPattern IUIAutomationTablePattern을 지원하는 컨테이너의 자식 요소에 대한 액세스를 제공합니다.
IUIAutomationTablePattern 자식 요소 컬렉션의 컨테이너 역할을 하는 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationTextChildPattern 다른 텍스트 기반 컨트롤의 자식 또는 하위 항목인 텍스트 기반 컨트롤(또는 텍스트에 포함된 개체)에 대한 액세스를 제공합니다.
IUIAutomationTextEditPattern 예를 들어 자동 고침을 수행하거나 IME(입력기)를 통한 입력 작성을 지원하는 컨트롤과 같이, 텍스트를 수정하는 컨트롤에 대한 액세스를 제공합니다.

IUIAutomationTextEditTextChangedEventHandler Microsoft UI Automation이 텍스트 편집 컨트롤에서 텍스트 변경 이벤트를 보고할 때 발생하는 이벤트를 처리하는 메서드를 노출합니다.
IUIAutomationTextPattern 텍스트가 포함된 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationTextPattern2 IUIAutomationTextPattern 인터페이스를 확장합니다.
IUIAutomationTextRange IUIAutomationTextPattern 인터페이스를 지원하는 컨테이너의 연속 텍스트 범위에 대한 액세스를 제공합니다. 클라이언트 애플리케이션은 IUIAutomationTextRange 인터페이스를 사용하여 텍스트 범위에서 포함된 개체를 선택, 비교 및 검색할 수 있습니다.
IUIAutomationTextRange2 IUIAutomationTextRange 인터페이스를 확장하여 Microsoft UI 자동화 클라이언트가 프로그래밍 방식으로 상황에 맞는 메뉴를 호출할 수 있도록 합니다.
IUIAutomationTextRange3 텍스트 범위에서 기본 서식 있는 텍스트 데이터에 더 빠르게 액세스할 수 있도록 IUIAutomationTextRange2 인터페이스를 확장합니다.
IUIAutomationTextRangeArray IUIAutomationTextRange 개체의 컬렉션을 나타냅니다.
IUIAutomationTogglePattern 상태 집합을 순환하고 설정된 후 상태를 유지할 수 있는 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationTransformPattern 이동, 크기 조정 또는 회전할 수 있는 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationTransformPattern2 IUIAutomationTransformPattern 인터페이스를 확장하여 Microsoft UI 자동화 클라이언트가 컨트롤의 뷰포트 확대/축소 기능에 프로그래밍 방식으로 액세스할 수 있도록 합니다.
IUIAutomationTreeWalker

UI 자동화 클라이언트 애플리케이션이 데스크톱에서 UI 자동화 요소를 보고 탐색하는 데 사용하는 속성과 메서드를 노출합니다.
IUIAutomationValuePattern 범위에 포함되지 않고 문자열로 나타낼 수 있는 값이 포함된 컨트롤에 대한 액세스를 제공합니다.
IUIAutomationVirtualizedItemPattern Microsoft UI 자동화 트리의 자리 표시자 자동화 요소가 나타내는 항목인 가상화된 항목을 나타냅니다.
IUIAutomationWindowPattern 창의 기본 기능에 대한 액세스를 제공합니다.

구조체

ExtendedProperty 확장 속성에 대한 정보를 포함합니다.

열거형

AutomationElementMode AutomationElementMode(uiautomationclient.h) 열거형에는 UI 자동화 요소를 반환할 때 사용할 참조 형식을 지정하는 값이 포함되어 있습니다.
CoalesceEventsOptions 액세스 가능한 기술 클라이언트가 모든 이벤트를 수신하는지 또는 중복 이벤트가 검색되고 필터링되는 하위 집합을 수신하는지 여부를 나타내는 CoalesceEvents 속성에 사용할 수 있는 값을 포함합니다.
ConnectionRecoveryBehaviorOptions ConnectionRecoveryBehavior 속성에 대한 가능한 값을 포함하며, 이는 공급자가 응답하지 않는 경우 액세스 가능한 기술 클라이언트가 공급자 요청 시간 제한을 조정하는지 여부를 나타냅니다.

PropertyConditionFlags PropertyConditionFlags(uiautomationclient.h) 열거형에는 속성 조건을 만드는 데 사용되는 값이 포함됩니다.
TreeScope TreeScope(uiautomationclient.h) 열거형에는 Microsoft UI 자동화 트리에서 다양한 작업의 scope 지정하는 값이 포함되어 있습니다.
TreeTraversalOptions TreeTraversalOptions(uiautomationclient.h) 열거형은 트리 탐색 순서를 사용자 지정하는 데 사용할 수 있는 값을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

uiautomationcore.h 헤더

이 헤더는 Windows 접근성 기능에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 접근성 기능

uiautomationcore.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Interfaces

테이블 확장

IAccessibleEx Microsoft UI Automation에서 호출한 메서드를 노출하여 Microsoft Active Accessibility를 지원하는 컨트롤에 대한 추가 정보를 검색합니다.
IAccessibleHostingElementProviders Microsoft Active Accessibility 개체는 개체가 Microsoft UI 자동화를 구현하는 창 없는 Microsoft ActiveX 컨트롤을 포함하는 접근성 트리의 루트인 경우 이 인터페이스를 구현합니다.
IAnnotationProvider 문서에서 주석의 속성을 노출합니다.
ICustomNavigationProvider 사용자 지정 탐색 순서를 지원하는 컨트롤에 대한 액세스를 제공합니다.
IDockProvider 도킹 컨테이너의 요소에 대한 액세스를 제공합니다.
IDragProvider Microsoft UI 자동화 요소가 끌어서 놓기 작업의 일부로 끌 수 있는 요소로 자신을 설명할 수 있도록 합니다.
IDropTargetProvider Microsoft UI 자동화 요소가 UI 자동화 끌어서 놓기 작업의 일부로 끌어온 요소의 삭제를 받을 수 있는 요소로 자신을 설명할 수 있도록 합니다.
IExpandCollapseProvider

시각적으로 확장하여 콘텐츠를 표시하고 축소하여 콘텐츠를 숨기는 컨트롤에 대한 액세스를 제공합니다.
IGridItemProvider IGridProvider를 구현하는 컨테이너의 개별 자식 컨트롤에 대한 액세스를 제공합니다.
IGridProvider 키보드를 사용하여 트래버스할 수 있는 2차원 논리 좌표계로 구성된 자식 요소 컬렉션의 컨테이너 역할을 하는 컨트롤(즉, Microsoft UI 자동화 클라이언트가 인접한 컨트롤로 이동할 수 있음)에 대한 액세스를 제공합니다.
IInvokeProvider 명확한 단일 작업을 시작하거나 수행하고 활성화될 때 상태를 유지 관리하지 않는 컨트롤에 대한 액세스를 제공합니다.
IItemContainerProvider 가상 목록 보기와 같은 다른 컨트롤의 컨테이너 역할을 하는 컨트롤에 대한 액세스를 제공합니다.
ILegacyIAccessibleProvider Microsoft UI 자동화 클라이언트가 Microsoft Active Accessibility 요소의 기본 IAccessible 구현에 액세스할 수 있도록 합니다.
IMultipleViewProvider 동일한 정보 또는 자식 컨트롤 집합의 여러 표현을 제공하고 전환할 수 있는 컨트롤에 대한 액세스를 제공합니다.
IObjectModelProvider 컨트롤 또는 애플리케이션에서 구현하는 기본 개체 모델에 대한 액세스를 제공합니다. (IObjectModelProvider)
IProxyProviderWinEventHandler WinEvents를 처리하기 위해 프록시 공급자가 구현하는 메서드를 노출합니다.
IProxyProviderWinEventSink 프록시 공급자가 이벤트를 발생시켜서 사용하는 메서드를 노출합니다.
IRangeValueProvider 범위 내의 값으로 설정할 수 있는 컨트롤에 대한 액세스를 제공합니다.
IRawElementProviderAdviseEvents

Microsoft UI 자동화 클라이언트 애플리케이션이 해당 조각의 이벤트 수신 대기를 시작하거나 종료할 때 조각의 루트 요소에 알리기 위해 호출되는 메서드를 노출합니다.
IRawElementProviderFragment 목록 상자 또는 목록 항목과 같이 둘 이상의 수준 깊이 구조에 속하는 UI 요소에 메서드 및 속성을 노출합니다. Microsoft UI 자동화 공급자가 구현합니다.
IRawElementProviderFragmentRoot 조각의 루트 요소에 메서드 및 속성을 노출합니다.
IRawElementProviderHostingAccessibles 이 인터페이스는 공급자가 Microsoft Active Accessibility를 지원하는 창 없는 컨트롤을 포함하는 접근성 트리의 루트인 경우 Microsoft UI 자동화 공급자에 의해 구현됩니다.
IRawElementProviderHwndOverride 조각의 UI 자동화 트리 내에서 창 기반 요소의 위치를 변경할 수 있는 메서드를 노출합니다.
IRawElementProviderSimple 간단한 UI 요소를 노출하는 메서드 및 속성을 정의합니다.
IRawElementProviderSimple2 프로그래밍 방식으로 상황에 맞는 메뉴를 호출할 수 있도록 IRawElementProviderSimple 인터페이스를 확장합니다.
IRawElementProviderSimple3 IRawElementProviderSimple2 인터페이스를 확장하여 접근성 있는 기술에 기본 설정 콘텐츠 형식을 말해야 하는 방법에 대한 메타데이터를 검색할 수 있도록 합니다.
IRawElementProviderWindowlessSite Microsoft ActiveX 컨트롤 사이트는 이 인터페이스를 구현하여 Microsoft UI 자동화 지원 ActiveX 컨트롤이 접근성을 표현할 수 있도록 합니다.
IScrollItemProvider IScrollProvider를 구현하는 컨테이너의 개별 자식 컨트롤에 대한 액세스를 제공합니다.
IScrollProvider 자식 개체 컬렉션에 대해 스크롤 가능한 컨테이너 역할을 하는 컨트롤에 대한 액세스를 제공합니다.
ISelectionItemProvider

ISelectionProvider를 구현하는 컨테이너의 선택 가능한 개별 자식 컨트롤에 대한 액세스를 제공합니다.
ISelectionProvider 선택 가능한 개별 자식 항목 컬렉션의 컨테이너 역할을 하는 컨트롤에 대한 액세스를 제공합니다.
ISelectionProvider2 ISelectionItemProvider 인터페이스를 확장하여 선택한 항목에 대한 정보를 제공합니다.
ISpreadsheetItemProvider 스프레드시트의 항목(셀)에 대한 정보에 대한 액세스를 제공합니다.
ISpreadsheetProvider 스프레드시트의 항목(셀)에 대한 액세스를 제공합니다.
IStylesProvider 문서의 내용과 연결된 비주얼 스타일에 대한 액세스를 제공합니다.
ISynchronizedInputProvider Microsoft UI 자동화 클라이언트 애플리케이션에서 마우스 또는 키보드 입력을 특정 UI 요소로 보낼 수 있습니다.
ITableItemProvider ITableProvider를 구현하는 컨테이너의 자식 컨트롤에 대한 액세스를 제공합니다.
ITableProvider 자식 요소 컬렉션의 컨테이너 역할을 하는 컨트롤에 대한 액세스를 제공합니다.
ITextChildProvider 다른 텍스트 기반 컨트롤의 자식 또는 하위 항목인 텍스트 기반 컨트롤(또는 텍스트에 포함된 개체)에 대한 액세스를 제공합니다.
ITextEditProvider Microsoft UI 자동화 공급자가 프로그래밍 방식 텍스트 편집 작업을 노출할 수 있도록 ITextProvider 인터페이스를 확장합니다.
ITextProvider 텍스트를 포함하는 컨트롤에 대한 액세스를 제공합니다.

ITextProvider2 ITextProvider 인터페이스를 확장하여 Microsoft UI 자동화 공급자가 주석의 대상인 텍스트 콘텐츠와 공급자에 속하는 케어에 대한 정보를 노출할 수 있도록 합니다.
ITextRangeProvider ITextProvider 또는 ITextProvider2를 구현하는 텍스트 컨테이너의 연속 텍스트 범위에 대한 액세스를 제공합니다.
ITextRangeProvider2 Microsoft UI 자동화 공급자가 상황에 맞는 메뉴를 호출할 수 있도록 ITextRangeProvider 인터페이스를 확장합니다.
IToggleProvider 상태 집합을 순환하고 설정된 후 상태를 유지할 수 있는 컨트롤에 대한 액세스를 제공합니다.
ITransformProvider 2차원 공간 내에서 이동, 크기 조정 및/또는 회전할 수 있는 컨트롤에 대한 액세스를 제공합니다.
ITransformProvider2 ITransformProvider 인터페이스를 확장하여 Microsoft UI 자동화 공급자가 컨트롤의 뷰포트 확대/축소 기능을 지원하는 속성을 노출할 수 있도록 합니다.
IUIAutomationClientConnectionCallback UI 자동화 클라이언트가 연결되거나 연결을 끊을 때 동기 알림을 수신하는 기능을 지원합니다.
IUIAutomationClientInfo 연결된 UI 자동화 클라이언트에 대한 정보를 수신하는 기능을 지원합니다.
IUIAutomationClientInfoSource 연결 콜백 등록 및 등록 취소 및 현재 연결된 UIA 클라이언트 열거를 지원합니다.
IUIAutomationPatternHandler 클라이언트 API 래퍼 개체를 반환하고 실제 공급자 인스턴스에 대한 속성 및 메서드 요청을 숨기지 않습니다.
IUIAutomationPatternInstance 컨트롤 패턴 개체를 나타냅니다. 클라이언트 API 래퍼는 이 인터페이스를 사용하여 GetProperty 및 CallMethod 메서드를 기준으로 모든 속성 및 메서드 호출을 구현합니다.

IUIAutomationRegistrar 새 컨트롤 패턴, 속성 및 이벤트를 등록하기 위한 메서드를 노출합니다.
IValueProvider 범위에 걸쳐 있지 않고 문자열로 나타낼 수 있는 내장 값이 있는 컨트롤에 대한 액세스를 제공합니다.
IVirtualizedItemProvider Microsoft UI 자동화 트리의 자리 표시자 자동화 요소로 표시되는 항목인 가상화된 항목에 대한 액세스를 제공합니다.
IWindowProvider 컨트롤의 기본 창 기반 기능에 대한 액세스를 제공합니다.

구조

테이블 확장

UiaChangeInfo 발생한 UI 자동화 변경에 대한 데이터를 포함합니다.
UiaPoint 점의 좌표를 포함합니다.
UiaRect 사각형의 위치와 크기를 화면 좌표로 포함합니다.
UIAutomationEventInfo 사용자 지정 이벤트에 대한 정보를 포함합니다.
UIAutomationMethodInfo 사용자 지정 컨트롤 패턴에서 지원하는 메서드에 대한 정보를 포함합니다.
UIAutomationParameter 사용자 지정 컨트롤 패턴의 매개 변수에 대한 정보를 포함합니다.

UIAutomationPatternInfo 사용자 지정 컨트롤 패턴에 대한 정보를 포함합니다.
UIAutomationPropertyInfo 사용자 지정 속성에 대한 정보를 포함합니다.

열거 목록

테이블 확장

ActiveEnd 현재 선택한 텍스트를 나타내는 텍스트 범위를 기준으로 캐럿의 위치를 나타내는 SelectionActiveEnd 텍스트 특성에 사용할 수 있는 값을 포함합니다.
AnimationStyle AnimationStyle 텍스트 특성에 대한 값을 포함합니다.
Bulletstyle BulletStyle 텍스트 특성에 대한 값을 포함합니다.
CapStyle CapStyle 텍스트 특성의 값을 지정하는 값을 포함합니다.
CaretBidiMode CaretBidiMode 텍스트 특성에 대해 가능한 값을 포함하며, 이는 caret이 왼쪽에서 오른쪽으로 또는 오른쪽에서 왼쪽으로 흐르는 텍스트에 있는지 여부를 나타냅니다.
CaretPosition 텍스트 범위의 텍스트 줄을 기준으로 하는 caret의 위치를 나타내는 CaretPosition 텍스트 특성에 대해 가능한 값을 포함합니다.
DockPosition Dock 컨트롤 패턴이 나타내는 도킹 창의 위치를 지정하는 값을 포함합니다.
ExpandCollapseState

확장 및 축소할 수 있는 UI 요소의 상태를 지정하는 값을 포함합니다.
FillType FillType 특성에 대한 값을 포함합니다.
FlowDirections TextFlowDirections 텍스트 특성에 대한 값을 포함합니다.
HorizontalTextAlignment 가로 텍스트 맞춤 값을 정의합니다.
LiveSetting LiveSetting 속성에 사용할 수 있는 값을 포함합니다. 이 속성은 라이브 지역의 일부인 공급자 요소에 의해 구현됩니다.
NavigateDirection Microsoft UI 자동화 트리 내에서 탐색 방향을 지정하는 데 사용되는 값을 포함합니다.
NotificationKind 알림 이벤트의 형식을 나타내는 값과 이벤트 처리에 대한 힌트를 수신기에 정의합니다.
NotificationProcessing 알림을 처리하는 방법을 나타내는 값을 정의합니다.
OrientationType 컨트롤의 방향을 지정하는 값을 포함합니다.
OutlineStyles OutlineStyle 텍스트 특성에 대한 값을 포함합니다.
ProviderOptions UI 자동화 공급자의 형식을 지정하는 값을 포함합니다. IRawElementProviderSimple::P roviderOptions 속성은 이 열거형을 사용합니다.
RowOrColumnMajor 테이블의 데이터를 주로 행 또는 열별로 읽을지 여부를 지정하는 값을 포함합니다.
SayAsInterpretAs

텍스트 음성 변환 엔진이 특정 데이터를 해석하는 방법을 나타내는 값을 정의합니다.
ScrollAmount 스크롤할 방향과 거리를 지정하는 값을 포함합니다.
StructureChangeType Microsoft UI 자동화 트리 구조의 변경 유형을 지정하는 값을 포함합니다.
SupportedTextSelection 지원되는 텍스트 선택 특성을 지정하는 값을 포함합니다.
SynchronizedInputType 동기화된 입력의 형식을 지정하는 값을 포함합니다.
TextDecorationLineStyle OverlineStyle, StrikethroughStyle 및 UnderlineStyle 텍스트 특성을 지정하는 값을 포함합니다.
TextEditChangeType 텍스트 편집 이벤트가 발생하거나 처리될 때 컨트롤에서 수행하는 텍스트 편집 변경에 대해 설명합니다.
TextPatternRangeEndpoint 텍스트 범위의 엔드포인트를 지정하는 값을 포함합니다.
TextUnit 탐색을 위해 텍스트 단위를 지정하는 값을 포함합니다.
ToggleState 토글 컨트롤 패턴을 구현하는 Microsoft UI Automation 요소의 토글 상태를 지정하는 값을 포함합니다.
UIAutomationType Microsoft UI 자동화 데이터 형식을 나타내는 데 사용되는 값을 포함합니다.
VisualEffects VisualEffects 특성에 대한 값을 포함합니다.
WindowInteractionState 사용자 상호 작용을 위해 창의 현재 상태를 지정하는 값을 포함합니다.

WindowVisualState 창의 시각적 상태를 지정하는 값을 포함합니다.
ZoomUnit 지정된 단위로 컨트롤의 뷰포트를 확대/축소하는 IUIAutomationTransformPattern2::ZoomByUnit 메서드에 대한 가능한 값을 포함합니다.

Last updated on 2026. 01. 13.

uiautomationcoreapi.h 헤더

아티클2023. 08. 26.

이 헤더는 Windows 접근성 기능에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 접근성 기능

uiautomationcoreapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DockPattern_SetDockPosition 도킹 컨테이너 내의 요청된 dockPosition에서 UI 자동화 요소를 도킹합니다.
ExpandCollapsePattern_Collapse UI 자동화 요소의 모든 하위 노드, 컨트롤 또는 콘텐츠를 숨깁니다.
ExpandCollapsePattern_Expand 더 많은 정보를 표시할 수 있도록 화면에서 컨트롤을 확장합니다.
GridPattern_GetItem 표에 있는 항목의 노드를 가져옵니다.
InvokePattern_Invoke 요청을 보내 컨트롤을 활성화하고 해당하는 명확한 단일 작업을 시작합니다. (InvokePattern_Invoke)
ItemContainerPattern_FindItemByProperty 지정된 속성 값을 기반으로 포함하는 노드 내의 노드를 검색합니다.
LegacyIAccessiblePattern_DoDefaultAction 요소에 대한 Microsoft Active Accessibility 기본 작업을 수행합니다. (LegacyIAccessiblePattern_DoDefaultAction)
LegacyIAccessiblePattern_GetIAccessible UI 자동화 요소에 해당하는 IAccessible 개체를 검색합니다.

LegacyIAccessiblePattern_Select Microsoft 활성 접근성 선택을 수행합니다. (LegacyIAccessiblePattern_Select)
LegacyIAccessiblePattern_SetValue 노드에 대한 Microsoft Active Accessibility 값 속성을 설정합니다.
MultipleViewPattern_GetViewName 컨트롤별 뷰 이름을 검색합니다. (MultipleViewPattern_GetViewName)
MultipleViewPattern_SetCurrentView 컨트롤을 다른 레이아웃으로 설정합니다.
RangeValuePattern_SetValue 숫자 범위가 있는 컨트롤의 값을 설정합니다.
ScrollItemPattern_ScrollIntoView 컨테이너의 표시되는 영역(뷰포트) 내에 UI 자동화 요소를 표시하기 위해 컨테이너 개체의 콘텐츠 영역을 스크롤합니다.
ScrollPattern_Scroll 지정한 ScrollAmount 콘텐츠 영역의 현재 표시되는 영역을 가로, 세로 또는 둘 다 스크롤합니다.
ScrollPattern_SetScrollPercent 컨테이너를 가로, 세로 또는 둘 다의 특정 위치로 스크롤합니다.
SelectionItemPattern_AddToSelection 선택되지 않은 요소를 컨트롤의 선택 영역에 추가합니다.
SelectionItemPattern_RemoveFromSelection 선택 컨테이너의 선택 영역에서 요소를 제거합니다.
SelectionItemPattern_Select 선택 컨테이너에서 요소를 선택합니다.
SynchronizedInputPattern_Cancel UI 자동화 공급자가 마우스 또는 키보드 입력에 대한 수신 대기를 중지하도록 합니다.

SynchronizedInputPattern_StartListening UI 자동화 공급자가 마우스 또는 키보드 입력을 수신 대기하도록 합니다.
TextPattern_get_DocumentRange 전체 문서의 텍스트 범위를 가져옵니다.
TextPattern_get_SupportedTextSelection 텍스트 컨테이너의 콘텐츠를 선택하고 선택 취소할 수 있는지 여부를 확인합니다.
TextPattern_GetSelection 텍스트 패턴을 지원하는 텍스트 컨테이너에서 선택한 텍스트의 현재 범위를 가져옵니다.
TextPattern_GetVisibleRanges 텍스트 컨테이너에서 각 텍스트 범위가 부분적으로 표시되는 첫 번째 줄로 시작되고 부분적으로 표시되는 마지막 줄의 끝까지 이어지는 서로 분리된 텍스트 범위의 배열을 검색합니다.
TextPattern_RangeFromChild 지정된 노드에 걸쳐 있는 텍스트 범위를 가져옵니다.
TextPattern_RangeFromPoint 지정된 화면 좌표에 가장 가까운 퇴화(빈) 텍스트 범위를 검색합니다. (TextPattern_RangeFromPoint)
TextRange_AddToSelection 호출하는 텍스트 범위 시작 및 끝 엔드포인트에 해당하는 보조 텍스트를 강조 표시하여 여러 개의 연결되지 않은 선택을 지원하는 텍스트 컨테이너의 강조 표시된 텍스트의 기존 컬렉션에 추가합니다.
TextRange_Clone 텍스트 범위를 복사합니다.
TextRange_Compare 두 텍스트 범위를 비교합니다.
TextRange_CompareEndpoints 두 텍스트 범위에 동일한 엔드포인트가 있는지 여부를 나타내는 값을 반환합니다.

TextRange_ExpandToEnclosingUnit 텍스트 범위를 문자, Word, 선 또는 페이지와 같은 더 크거나 작은 단위로 확장합니다.
TextRange_FindAttribute 지정된 텍스트 특성을 지원하는 텍스트의 첫 번째 부분을 지정된 방향으로 검색합니다.
TextRange_FindText 클라이언트에서 검색하는 텍스트가 포함된 지정된 방향으로 첫 번째 텍스트 범위를 반환합니다.
TextRange_GetAttributeValue 텍스트 범위의 텍스트 특성 값을 가져옵니다.
TextRange_GetBoundingRectangles 범위를 묶을 수 있는 경계 사각형의 최소 개수를 검색합니다(줄당 하나의 사각형).
TextRange_GetChildren 지정된 텍스트 범위 내에 포함된 모든 UI 자동화 요소를 반환합니다.
TextRange_GetEnclosingElement 범위를 포함하는 다음으로 작은 공급자에 대한 노드를 반환합니다.
TextRange_GetText 지정된 문자 수까지 텍스트 범위의 텍스트를 반환합니다.
TextRange_Move 클라이언트에서 요청한 지정된 단위 수를 텍스트 범위로 이동합니다.
TextRange_MoveEndpointByRange 한 범위의 엔드포인트를 다른 범위의 엔드포인트로 이동합니다.
TextRange_MoveEndpointByUnit 범위의 엔드포인트를 지정된 단위 수로 이동합니다.
TextRange_RemoveFromSelection 여러 개의 연결되지 않은 선택을 지원하는 텍스트 컨테이너의 선택한 텍스트 컬렉션에서 호출 텍스트 범위 TextPatternRangeEndpoint_Start 및 TextPatternRangeEndpoint_End 엔드포인트에 해당하는 선택한 텍스트를 제거합니다.

TextRange_ScrollIntoView 지정된 범위가 뷰포트에 표시되도록 텍스트를 스크롤합니다.
TextRange_Select 텍스트 범위를 선택합니다.
TogglePattern_Toggle 컨트롤을 지원되는 다음 상태로 전환합니다.
TransformPattern_Move 요소를 화면의 지정된 위치로 이동합니다.
TransformPattern_Resize 화면의 요소 크기를 조정합니다.
TransformPattern_Rotate 화면에서 요소를 회전합니다.
UiaAddEvent UI 자동화 트리의 노드에서 이벤트에 대한 수신기를 추가합니다.
UiaClientsAreListening 클라이언트 애플리케이션이 Microsoft UI 자동화 이벤트를 구독하는지 여부를 나타내는 값을 가져옵니다.
UiaDisconnectAllProviders 호출 프로세스와 연결된 모든 공급자가 보유한 모든 Microsoft UI 자동화 리소스를 해제합니다.
UiaDisconnectProvider 특정 공급자가 Microsoft UI 자동화 개체에 보유하는 모든 참조를 해제합니다.
UiaEventAddWindow 이벤트 수신기에 창을 추가합니다.
UiaEventRemoveWindow 이벤트 수신기에서 창을 제거합니다.

UiaFind 검색 조건과 일치하는 하나 이상의 UI 자동화 노드를 검색합니다.
UiaGetErrorDescription 클라이언트에 전달할 수 있도록 오류 문자열을 가져옵니다. 이 메서드는 클라이언트에서 직접 사용되지 않습니다.
UiaGetPatternProvider 컨트롤 패턴을 검색합니다.
UiaGetPropertyValue UI 자동화 속성의 값을 검색합니다.
UiaGetReservedMixedAttributeValue Microsoft UI 자동화 텍스트 특성의 값이 텍스트 범위 내에서 달라진다는 것을 나타내는 예약된 값을 검색합니다.
UiaGetReservedNotSupportedValue Microsoft UI 자동화 속성 또는 텍스트 특성이 지원되지 않음을 나타내는 예약된 값을 검색합니다.
UiaGetRootNode 루트 UI 자동화 노드를 검색합니다.
UiaGetRuntimeId UI 자동화 노드의 런타임 식별자를 검색합니다.
UiaGetUpdatedCache 속성 값 및 컨트롤 패턴의 캐시를 업데이트.
UiaHasServerSideProvider 창에 Microsoft UI 자동화 공급자 구현이 있는지 여부를 확인합니다.
UiaHostProviderFromHwnd 창에 대한 호스트 공급자를 가져옵니다.
UiaHPatternObjectFromVariant VARIANT 형식에서 컨트롤 패턴 개체를 가져옵니다.

UiaHTextRangeFromVariant VARIANT 형식에서 텍스트 범위를 가져옵니다.
UiaHUiaNodeFromVariant VARIANT 형식에서 HUIANODE를 가져옵니다.
UiaIAccessibleFromProvider Microsoft UI 자동화 공급자를 대신하여 Microsoft Active Accessibility 데이터를 제공하는 IAccessible 구현을 검색합니다.
UiaLookupId PROPERTYID, PATTERNID, CONTROLTYPEID, TEXTATTRIBUTEID 또는 EVENTID가 필요한 메서드에서 사용할 수 있는 정수 식별자를 가져옵니다.
UiaNavigate 선택적으로 캐시된 정보를 검색하여 UI 자동화 트리를 탐색합니다.
UiaNodeFromFocus 현재 입력 포커스가 있는 UI 요소에 대한 UI 자동화 노드를 검색합니다.
UiaNodeFromHandle 창과 연결된 UI 자동화 노드를 검색합니다.
UiaNodeFromPoint 지정된 지점에서 요소의 UI 자동화 노드를 검색합니다.
UiaNodeFromProvider 원시 요소 공급자에 대한 UI 자동화 노드를 검색합니다.
UiaNodeRelease 메모리에서 노드를 삭제합니다.
UiaPatternRelease 메모리에서 할당된 패턴 개체를 삭제합니다.
UiaProviderForNonClient

창의 전체 비 클라이언트 영역 또는 창의 비 클라이언트 영역에 있는 컨트롤에 대한 공급자를 가져옵니다.
UiaProviderFromIAccessible 지정된 Microsoft Active Accessibility 개체를 기반으로 Microsoft UI 자동화 공급자를 만듭니다.
UiaRaiseActiveTextPositionChangedEvent 공급자가 텍스트 컨트롤에 프로그래밍 방식으로 변경된 텍스트가 있음을 Microsoft UI 자동화 코어에 알리기 위해 호출됩니다. (UiaRaiseActiveTextPositionChangedEvent)
UiaRaiseAsyncContentLoadedEvent Microsoft UI 자동화 코어에 콘텐츠가 비동기적으로 로드되고 있음을 알리기 위해 공급자가 호출합니다.
UiaRaiseAutomationEvent 수신기에 이벤트를 알 수 있습니다.
UiaRaiseAutomationPropertyChangedEvent 공급자가 호출하여 요소 속성이 변경되었음을 Microsoft UI 자동화 코어에 알립니다.
UiaRaiseChangesEvent 공급자가 Microsoft UI 자동화 코어에 변경이 발생했음을 알리기 위해 호출됩니다.
UiaRaiseNotificationEvent 공급자가 알림 이벤트를 시작하기 위해 호출합니다.
UiaRaiseStructureChangedEvent Microsoft UI 자동화 코어에 트리 구조가 변경되었음을 알리기 위해 공급자가 호출합니다.
UiaRaiseTextEditTextChangedEvent 공급자가 텍스트 컨트롤에 프로그래밍 방식으로 변경된 텍스트가 있음을 Microsoft UI 자동화 코어에 알리기 위해 호출됩니다. (UiaRaiseTextEditTextChangedEvent)
UiaRectIsEmpty 사각형에 모든 좌표가 0으로 설정되어 있는지 여부를 지정하는 부울 값을 가져옵니다.
UiaRectSetEmpty UiaRect 구조체의 요소를 0으로 설정합니다.

UiaRegisterProviderCallback 요소에 대한 공급자를 가져오기 위해 UI 자동화에서 호출하는 애플리케이션 정의 메서드를 등록합니다.
UiaRemoveEvent UI 자동화 트리의 노드에서 이벤트에 대한 수신기를 제거합니다.
UiaReturnRawElementProvider 창에 대한 UI 자동화 공급자에 대한 인터페이스를 가져옵니다.
UiaSetFocus 입력 포커스를 UI의 지정된 요소로 설정합니다.
UiaTextRangeRelease 메모리에서 할당된 텍스트 범위 개체를 삭제합니다.
ValuePattern_SetValue 요소의 텍스트 값을 설정합니다.
VirtualizedItemPattern_Realize 가상 항목을 UI 자동화 요소로 완전히 액세스할 수 있도록 만듭니다. (VirtualizedItemPattern_Realize)
WindowPattern_Close 열린 창을 닫습니다.
WindowPattern_SetWindowVisualState 창의 시각적 상태를 설정합니다. 예를 들어 창을 최대화합니다.
WindowPattern_WaitForInputIdle 지정된 시간 동안 또는 연결된 프로세스가 유휴 상태가 될 때까지 중 먼저 완료되는 시간 동안 코드 호출을 차단합니다. (WindowPattern_WaitForInputIdle)

콜백 함수

UiaEventCallback 클라이언트가 구독한 이벤트가 발생할 때 UI 자동화에서 호출되는 클라이언트 구현 함수입니다.
UiaProviderCallback 요소에 대한 클라이언트 쪽 공급자를 가져오기 위해 UI 자동화에서 호출하는 애플리케이션 정의 함수입니다.

구조체

UiaAndOrCondition 참고 이 구조체는 더 이상 사용되지 않습니다. 복잡한 조건에 대한 정보를 포함합니다.
UiaAsyncContentLoadedEventArgs 참고 이 구조체는 더 이상 사용되지 않습니다. 콘텐츠가 UI 요소에 의해 비동기적으로 로드될 때 발생하는 이벤트에 대한 정보를 포함합니다.
UiaCacheRequest 참고 이 구조체는 더 이상 사용되지 않습니다. UI 자동화 요소에 대한 데이터를 캐시하는 요청에 대한 정보를 포함합니다.
UiaCondition 참고 이 구조체는 더 이상 사용되지 않습니다. 조건에 대한 정보를 포함합니다.
UiaEventArgs 참고 이 구조체는 더 이상 사용되지 않습니다. Microsoft UI 자동화 이벤트에 대한 정보를 포함합니다.
UiaFindParams 참고 이 구조체는 더 이상 사용되지 않습니다. UiaFind 함수에 사용되는 매개 변수를 포함합니다.
UiaNotCondition 참고 이 구조체는 더 이상 사용되지 않습니다. 음수 조건에 대한 정보를 포함합니다.
UiaPropertyChangedEventArgs

참고 이 구조체는 더 이상 사용되지 않습니다. Microsoft UI 자동화 요소 속성 변경이 발생할 때 발생하는 이벤트에 대한 정보를 포함합니다.
UiaPropertyCondition 참고 이 구조체는 더 이상 사용되지 않습니다. 일치하는 속성이 있는 UI 자동화 요소를 찾는 데 사용되는 조건에 대한 정보를 포함합니다.
UiaStructureChangedEventArgs 참고 이 구조체는 더 이상 사용되지 않습니다. Microsoft UI 자동화 트리의 구조가 변경될 때 발생하는 이벤트에 대한 정보를 포함합니다.
UiaWindowClosedEventArgs 참고 이 구조체는 더 이상 사용되지 않습니다. 하나 이상의 창이 닫히면 발생하는 이벤트에 대한 정보를 포함합니다.

열거형

AsyncContentLoadedState 콘텐츠의 비동기 로드 진행률을 설명하는 값을 포함합니다.
AutomationElementMode AutomationElementMode(uiautomationcoreapi.h) 열거형에는 UI 자동화 요소를 반환할 때 사용할 참조 형식을 지정하는 값이 포함되어 있습니다.
AutomationIdentifierType UiaLookupId 함수에 사용되는 값을 포함합니다.
ConditionType UiaCondition 형식을 지정하는 값을 포함합니다.
EventArgsType UiaEventArgs 구조체에서 설명하는 이벤트 유형을 지정하는 값을 포함합니다.
NormalizeState UiaGetUpdatedCache의 동작을 지정하는 값을 포함합니다.

PropertyConditionFlags PropertyConditionFlags(uiautomationcoreapi.h) 열거형에는 속성 조건을 만드는 데 사용되는 값이 포함됩니다.
ProviderType 클라이언트 쪽(프록시) UI 자동화 공급자의 형식을 지정하는 값을 포함합니다.
TreeScope TreeScope(uiautomationcoreapi.h) 열거형에는 Microsoft UI 자동화 트리에서 다양한 작업의 scope 지정하는 값이 포함되어 있습니다.
TreeTraversalOptions TreeTraversalOptions(uiautomationcoreapi.h) 열거형은 트리 탐색 순서를 사용자 지정하는 데 사용할 수 있는 값을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

uiribbon.h 헤더

아티클2023. 08. 26.

이 헤더는 Windows 리본 프레임워크에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 리본 프레임워크

uiribbon.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IUIApplication IUIApplication 인터페이스는 애플리케이션에서 구현되며 Windows 리본 프레임워크에 대한 콜백 진입점 메서드를 정의합니다.
IUICollection IUICollection 인터페이스는 리본 프레임워크에 의해 구현됩니다.
IUICollectionChangedEvent IUICollectionChangedEvent 인터페이스는 애플리케이션에서 구현되며 런타임에 컬렉션의 변경 내용을 처리하는 데 필요한 메서드를 정의합니다.
IUICommandHandler IUICommandHandler 인터페이스는 애플리케이션에서 구현되며 명령 정보를 수집하고 Windows 리본 프레임워크에서 명령 이벤트를 처리하는 메서드를 정의합니다.
IUIContextualUI IUIContextualUI 인터페이스는 리본 프레임워크에서 구현되며 컨텍스트 팝업 보기의 핵심 기능을 제공합니다.
IUIEventingManager IUIEventingManager 인터페이스는 리본 프레임워크에서 구현되며 리본 이벤트에 등록하는 애플리케이션에 대한 알림 기능을 제공합니다.
IUIEventLogger IUIEventLogger 인터페이스는 애플리케이션에서 구현되며 리본 이벤트 콜백 메서드를 정의합니다.

IUIFramework IUIFramework 인터페이스는 Windows 리본 프레임워크에서 구현되며 프레임워크의 핵심 기능을 제공하는 메서드를 정의합니다.
IUIImage IUIImage 인터페이스는 애플리케이션에서 구현되며 Windows 리본 프레임워크 의 리본 및 컨텍스트 팝업 UI에 표시할 이미지를 검색하는 메서드를 정의합니다.
IUIImageFromBitmap IUIImageFromBitmap은 IUIImage 개체를 만드는 메서드를 정의하는 Windows 리본 프레임워크에서 구현하는 팩터리 인터페이스입니다.
IUIRibbon IUIRibbon 인터페이스는 Windows 리본 프레임워크에서 구현되며 리본 메뉴의 설정 및 속성을 지정하는 기능을 제공합니다.
IUISimplePropertySet IUISimplePropertySet은 속성 키로 식별된 값을 검색하는 메서드를 정의하는 읽기 전용 인터페이스입니다.

구조체

UI_EVENTPARAMS 리본 이벤트에 대한 정보를 포함합니다.
UI_EVENTPARAMS_COMMAND 이벤트와 연결된 명령에 대한 정보를 포함합니다.

열거형

UI_COLLECTIONCHANGE 컬렉션에 적용할 수 있는 변경 형식을 식별하는 값을 지정합니다.

UI_COMMANDTYPE 리본 컨트롤과 연결된 명령의 형식을 식별하는 값을 지정합니다.
UI_CONTEXTAVAILABILITY 상황에 맞는 탭의 가용성을 식별하는 값을 지정합니다.
UI_CONTROLDOCK QAT(빠른 실행 도구 모음)의 도크 상태를 식별하는 값을 지정합니다.
UI_EVENTLOCATION 리본 컨트롤과 연결된 이벤트가 시작될 수 있는 위치를 식별합니다.
UI_EVENTTYPE 리본 메뉴와 연결된 이벤트 유형을 식별합니다.
UI_EXECUTIONVERB 사용자가 명령에서 시작할 수 있는 작업에 매핑되는 실행 ID를 식별하는 값을 지정합니다.
UI_FONTDELTASIZE 강조 표시된 텍스트 실행의 글꼴 크기를 늘리거나 감소시켜야 하는지 여부를 식별하는 값을 지정합니다.
UI_FONTPROPERTIES Strikethrough와 같은 FontControl의 글꼴 속성 상태를 식별하는 값을 지정합니다.
UI_FONTUNDERLINE FontControl의 밑줄 상태를 식별하는 값을 지정합니다.
UI_FONTVERTICALPOSITION FontControl의 세로 맞춤 상태를 식별하는 값을 지정합니다.
UI_INVALIDATIONS 무효화할 명령의 측면을 식별하는 값을 지정합니다.
UI_OWNERSHIP Windows 리본 프레임워크에서 이미지를 만드는 소유권 조건을 식별하는 값을 지정합니다.

UI_SWATCHCOLORMODE 견본에 표준 모드 또는 단색 모드가 있는지 여부를 지정합니다.
UI_SWATCHCOLORTYPE DropDownColorPicker 또는 FontControl 색 선택기(텍스트 색 또는 텍스트 강조 표시)의 색 견본을 채우는 방법을 식별하는 값을 지정합니다. 참고 권장 사항입니다.
UI_VIEWTYPE 리본 프레임워크 뷰를 식별하는 값을 지정합니다.
UI_VIEWVERB 리본 프레임워크 보기에서 완료할 작업 유형을 식별하는 값을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

uiviewsettingsinterop.h 헤더

아티클2023. 08. 26.

이 헤더는 WinRT 상호 운용성에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- WinRT 상호 운용성
- UIViewSettings 클래스

uiviewsettingsinterop.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IUIViewSettingsInterop WinRT UIViewSettings 개체와의 상호 운용성을 사용하도록 설정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

unknwn.h 헤더

아티클2023. 03. 16.

이 헤더는 COM(구성 요소 개체 모델)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)

unknwn.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IClassFactory IClassFactory 인터페이스를 사용하면 개체 클래스를 만들 수 있습니다.
IUnknown 클라이언트는 QueryInterface 메서드를 통해 지정된 개체의 다른 인터페이스에 대한 포인터를 가져와 AddRef 및 Release 메서드를 통해 개체의 존재를 관리할 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

unknwnbase.h 헤더

아티클2023. 03. 16.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)
- RPC(원격 프로시저 호출)

unknwnbase.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IClassFactory IClassFactory 인터페이스(unknwnbase.h)를 사용하면 개체 클래스를 만들 수 있습니다.

Functions

IUnknown_AddRef_Proxy IUnknown_AddRef_Proxy 함수는 모든 인터페이스 프록시에 대해 AddRef 메서드를 구현합니다.
IUnknown_QueryInterface_Proxy IUnknown_QueryInterface_Proxy 함수는 모든 인터페이스 프록시에 대해 QueryInterface 메서드를 구현합니다.
IUnknown_Release_Proxy IUnknown_Release_Proxy 함수는 모든 인터페이스 프록시에 대해 Release 메서드를 구현합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

upnp.h 헤더

아티클2023. 03. 16.

이 헤더는 UPnP API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- UPnP API

upnp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IUPnPAddressFamilyControl IUPnPAddressFamilyControl 인터페이스는 Device Finder 개체의 주소 패밀리 플래그에 액세스합니다.
IUPnPAsyncResult IUPnPAsyncResult 인터페이스는 완료된 비동기 I/O 작업의 UPnP 제어 지점에 알리는 데 사용됩니다.
IUPnPDescriptionDocument IUPnPDescriptionDocument 인터페이스를 사용하면 애플리케이션이 디바이스 설명을 로드할 수 있습니다.
IUPnPDescriptionDocumentCallback IUPnPDescriptionDocumentCallback 인터페이스를 사용하면 UPnP 프레임워크가 비동기 로드 작업의 결과를 애플리케이션에 전달할 수 있습니다.
IUPnPDevice IUPnPDevice 인터페이스를 사용하면 애플리케이션이 특정 디바이스에 대한 정보를 검색할 수 있습니다.
IUPnPDeviceDocumentAccess IUPnPDeviceDocumentAccess 인터페이스를 사용하면 애플리케이션이 디바이스 설명 문서의 URL을 가져올 수 있습니다.
IUPnPDeviceDocumentAccessEx 특정 디바이스에 대한 전체 XML 디바이스 설명 문서를 가져오는 메서드를 제공합니다.

IUPnPDeviceFinder IUPnPDeviceFinder 인터페이스를 사용하면 애플리케이션에서 디바이스를 찾을 수 있습니다.
IUPnPDeviceFinderAddCallbackWithInterface IUPnPDeviceFinderAddCallbackWithInterface 인터페이스를 사용하면 UPnP 프레임워크가 애플리케이션과 통신할 수 있습니다.
IUPnPDeviceFinderCallback IUPnPDeviceFinderCallback 인터페이스를 사용하면 UPnP 프레임워크가 비동기 검색 결과를 애플리케이션에 전달할 수 있습니다.
IUPnPDevices IUPnPDevices 인터페이스는 디바이스 컬렉션을 열거합니다.
IUPnPHttpHeaderControl 호출자가 디바이스에 대한 HTTP 요청으로 전송되는 추가 HTTP 헤더를 지정할 수 있도록 합니다.
IUPnPService IUPnPService 인터페이스를 사용하면 애플리케이션이 상태 변수를 쿼리하고 서비스 인스턴스에서 작업을 호출할 수 있습니다.
IUPnPServiceAsync 이 인터페이스를 사용하여 상태 변수를 비동기적으로 쿼리하고 서비스 인스턴스에서 작업을 호출합니다.
IUPnPServiceCallback IUPnPServiceCallback 인터페이스는 서비스 개체의 클라이언트에 이벤트 알림을 보내는 데 사용됩니다.
IUPnPServiceDocumentAccess 이 인터페이스를 사용하여 SCPD(서비스 제어 프로토콜 설명) 문서를 검색하여 UPnP 제어 지점 애플리케이션에 제공하여 서비스에서 지원하는 작업을 노출하고 상태 변수에 대한 정보를 제공합니다.
IUPnPServiceEnumProperty 이 인터페이스를 사용하여 IUPnPServices 개체에서 열거된 IUPnPService 개체에서 SCPD(서비스 제어 프로토콜 설명) 다운로드 및 이벤트 구독을 지연할 수 있습니다.

IUPnPServices IUPnPServices 인터페이스는 서비스 컬렉션을 열거합니다.

upnphost.h 헤더

아티클2023. 08. 26.

이 헤더는 UPnP API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- UPnP API

upnphost.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IUPnPDeviceControl IUPnPDeviceControl 인터페이스는 디바이스 및 해당 서비스 개체에 대한 관리의 중심 지점입니다.
IUPnPDeviceProvider IUPnPDeviceProvider 인터페이스를 사용하면 디바이스 공급자가 해당 처리를 시작하고 중지할 수 있습니다.
IUPnPEventSink IUPnPEventSink 인터페이스를 사용하면 호스트된 서비스가 디바이스 호스트에 이벤트 알림을 보낼 수 있습니다.
IUPnPEventSource IUPnPEventSource 인터페이스를 사용하면 디바이스 호스트가 호스트된 서비스에 대한 이벤트 구독을 관리할 수 있습니다.
IUPnPRegistrar IUPnPRegistrar 인터페이스는 디바이스 호스트의 컨텍스트에서 실행되는 디바이스를 등록합니다.
IUPnPRemoteEndpointInfo IUPnPRemoteEndpointInfo 인터페이스를 사용하면 호스트된 디바이스가 요청자(즉, 제어점) 및 요청에 대한 정보를 가져올 수 있습니다.
IUPnPReregistrar IUPnPReregistrar 인터페이스를 사용하면 애플리케이션이 디바이스 호스트에 UPnP 기반 디바이스를 다시 등록할 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

urlmon.h 헤더

아티클2023. 08. 26.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)
- Windows 셸

urlmon.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CreateAsyncBindCtx 비동기 모니커에 사용할 비동기 바인딩 컨텍스트를 만듭니다.
CreateFormatEnumerator FORMATETC 구조체의 정적 배열을 통해 IEnumFORMATETC를 구현하는 개체를 만듭니다.

구조체

SOFTDISTINFO 소프트웨어 업데이트에 대한 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

usbuser.h 헤더

아티클2023. 08. 26.

이 헤더는 USB 드라이버 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- USB 드라이버 참조

usbuser.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

IOCTLs

IOCTL_GET_HCD_DRIVERKEY_NAME IOCTL_GET_HCD_DRIVERKEY_NAME I/O 제어 요청은 USB 호스트 컨트롤러 드라이버에 대한 레지스트리의 드라이버 키 이름을 검색합니다.
IOCTL_USB_DIAGNOSTIC_MODE_OFF IOCTL_USB_DIAGNOSTIC_MODE_OFF I/O 컨트롤은 더 이상 사용되지 않습니다. 사용하지 마십시오.
IOCTL_USB_DIAGNOSTIC_MODE_ON IOCTL_USB_DIAGNOSTIC_MODE_ON I/O 컨트롤은 더 이상 사용되지 않습니다. 사용하지 마십시오.
IOCTL_USB_GET_ROOT_HUB_NAME IOCTL_USB_GET_ROOT_HUB_NAME I/O 컨트롤 요청은 루트 허브의 기호 링크 이름을 검색하기 위해 USB_ROOT_HUB_NAME 구조체와 함께 사용됩니다. IOCTL_USB_GET_ROOT_HUB_NAME 사용자 모드 I/O 제어 요청입니다.
IOCTL_USB_USER_REQUEST IOCTL_USB_USER_REQUEST I/O 제어 요청은 사용자 모드 애플리케이션과 커널 모드 드라이버 모두에서 사용할 수 있습니다.

구조체

USB_BANDWIDTH_INFO

USB_BANDWIDTH_INFO 구조는 할당된 대역폭에 대한 정보를 검색하기 위해 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용됩니다.
USB_BUS_STATISTICS_0 USB_BUS_STATISTICS_0 구조는 버스 통계를 검색하기 위해 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용됩니다.
USB_CLOSE_RAW_DEVICE_PARAMETERS 이 구조는 지원되지 않습니다. USB_CLOSE_RAW_DEVICE_PARAMETERS 구조는 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용하여 버스의 디바이스에 대한 원시 액세스를 닫습니다.
USB_CONTROLLER_INFO_0 USB_CONTROLLER_INFO_0 구조는 USB 호스트 컨트롤러에 대한 정보를 검색하기 위해 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용됩니다.
USB_DRIVER_VERSION_PARAMETERS USB_DRIVER_VERSION_PARAMETERS 구조는 버전 정보를 검색하기 위해 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용됩니다.
USB_PASS_THRU_PARAMETERS USB_PASS_THRU_PARAMETERS 구조는 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용하여 공급업체별 명령을 호스트 컨트롤러 미니포트 드라이버에 전달합니다.
USB_POWER_INFO USB_POWER_INFO 구조는 호스트 컨트롤러 전원 정책이 표시된 시스템 전원 상태에 대해 지정하는 디바이스 전원 상태를 검색하기 위해 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용됩니다.
USB_UNICODE_NAME USB_UNICODE_NAME 구조체에는 기호 링크 이름을 지정하는 유니코드 문자열이 포함되어 있습니다.
USBUSER_BANDWIDTH_INFO_REQUEST USBUSER_BANDWIDTH_INFO_REQUEST 구조는 할당된 대역폭에 대한 정보를 검색하기 위해 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용됩니다.
USBUSER_BUS_STATISTICS_0_REQUEST USBUSER_BUS_STATISTICS_0_REQUEST 구조는 버스 통계를 검색하기 위해 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용됩니다.

USBUSER_CONTROLLER_UNICODE_NAME USBUSER_CONTROLLER_UNICODE_NAME 구조는 USB 호스트 컨트롤러 드라이버키 이름을 검색하기 위해 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용됩니다.
USBUSER_GET_DRIVER_VERSION USBUSER_GET_DRIVER_VERSION 구조는 드라이버 및 인터페이스 버전 정보를 읽기 위해 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용됩니다.
USBUSER_PASS_THRU_REQUEST USBUSER_PASS_THRU_REQUEST 구조는 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용하여 공급업체별 명령을 호스트 컨트롤러 미니포트 드라이버에 보냅니다.
USBUSER_POWER_INFO_REQUEST USBUSER_POWER_INFO_REQUEST 구조는 호스트 컨트롤러 및 루트 허브의 전원 상태와 특정 시스템 상태의 관계에 관한 전원 정책 정보를 검색하기 위해 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용됩니다.
USBUSER_REQUEST_HEADER USBUSER_REQUEST_HEADER 구조는 usb 호스트 컨트롤러 드라이버에 사용자 모드 요청을 보내기 위해 IOCTL_USB_USER_REQUEST I/O 제어 요청과 함께 사용됩니다.

열거형

USB_USER_ERROR_CODE USB_USER_ERROR_CODE 열거형은 USB 사용자 모드가 실패할 때 보고하는 오류 코드를 나열합니다.
WDMUSB_POWER_STATE WDMUSB_POWER_STATE 열거형은 호스트 컨트롤러 또는 루트 허브의 전원 상태를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

useractivityinterop.h 헤더

아티클2023. 08. 15.

이 헤더는 WinRT 상호 운용성에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- WinRT 상호 운용성
- UserActivity 클래스

useractivityinterop.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IUserActivityInterop WinRT UserActivity 개체와의 상호 운용성을 사용하도록 설정합니다.
IUserActivityRequestManagerInterop WinRT UserActivityRequestManager 개체와의 상호 운용성을 사용하도록 설정합니다.
IUserActivitySourceHostInterop IUserActivitySourceHostInterop 인터페이스를 사용하면 WinRT와 UserActivityChannel 또는 UserActivity 개체와의 상호 운용성을 사용할 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

userconsentverifierinterop.h 헤더

아티클2023. 08. 26.

이 헤더는 WinRT 상호 운용성에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- WinRT 상호 운용성
- UserConsentVerifier 클래스

userconsentverifierinterop.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IUserConsentVerifierInterop WinRT UserConsentVerifier 클래스 개체와의 상호 운용성을 사용하도록 설정하고 UserConsentVerifier 멤버에 대한 액세스를 제공하여 현재 사용자를 확인합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

userenv.h 헤더

아티클2023. 07. 26.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 그룹 정책 API
- Windows 셸

userenv.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CreateAppContainerProfile AppContainer에 대한 사용자별 앱별 프로필을 만듭니다.
CreateEnvironmentBlock 지정된 사용자의 환경 변수를 검색합니다. 그런 다음 이 블록을 CreateProcessAsUser 함수에 전달할 수 있습니다.
CreateProfile 새 사용자 프로필을 만듭니다.
DeleteAppContainerProfile 지정된 사용자별 앱별 프로필을 삭제합니다.
DeleteProfileA 지정된 컴퓨터에서 사용자 프로필 및 모든 사용자 관련 설정을 삭제합니다. 호출자에게는 사용자의 프로필을 삭제할 수 있는 관리자 권한이 있어야 합니다. (ANSI)
DeleteProfileW 지정된 컴퓨터에서 사용자 프로필 및 모든 사용자 관련 설정을 삭제합니다. 호출자에게는 사용자의 프로필을 삭제할 수 있는 관리자 권한이 있어야 합니다. (유니코드)
DeriveAppContainerSidFromAppContainerName 지정된 프로필의 SID를 가져옵니다.
DeriveRestrictedAppContainerSidFromAppContainerSidAndRestrictedName

DeriveRestrictedAppContainerSidFromAppContainerSidAndRestrictedName은 나중에 사용하도록 예약되어 있습니다.
DestroyEnvironmentBlock CreateEnvironmentBlock 함수에서 만든 환경 변수를 해제합니다.
EnterCriticalPolicySection EnterCriticalPolicySection 함수는 애플리케이션이 정책 설정을 안전하게 읽을 수 있도록 정책 애플리케이션을 일시 중지합니다.
ExpandEnvironmentStringsForUserA 지정된 사용자에 대해 설정된 환경 블록을 사용하여 원본 문자열을 확장합니다. (ANSI)
ExpandEnvironmentStringsForUserW 지정된 사용자에 대해 설정된 환경 블록을 사용하여 원본 문자열을 확장합니다. (유니코드)
FreeGPOListA FreeGPOList 함수는 지정된 GPO 목록을 해제합니다. (ANSI)
FreeGPOListW FreeGPOList 함수는 지정된 GPO 목록을 해제합니다. (유니코드)
GetAllUsersProfileDirectoryA 모든 사용자가 공유하는 프로그램 데이터가 포함된 디렉터리의 루트 경로를 검색합니다. (ANSI)
GetAllUsersProfileDirectoryW 모든 사용자가 공유하는 프로그램 데이터가 포함된 디렉터리의 루트 경로를 검색합니다. (유니코드)
GetAppContainerFolderPath 지정된 앱 컨테이너에 대한 로컬 앱 데이터 폴더의 경로를 가져옵니다.
GetAppContainerRegistryLocation 앱 컨테이너와 연결된 레지스트리 스토리지의 위치를 가져옵니다.
GetAppliedGPOListA GetAppliedGPOList 함수는 지정된 사용자 또는 컴퓨터에 적용된 GPO 목록을 검색합니다. (ANSI)

GetAppliedGPOListW GetAppliedGPOList 함수는 지정된 사용자 또는 컴퓨터에 적용된 GPO 목록을 검색합니다. (유니코드)
GetDefaultUserProfileDirectoryA 기본 사용자 프로필의 루트 경로를 검색합니다. (ANSI)
GetDefaultUserProfileDirectoryW 기본 사용자 프로필의 루트 경로를 검색합니다. (유니코드)
GetGPOListA GetGPOList 함수는 지정된 사용자 또는 컴퓨터의 GPO 목록을 검색합니다. (ANSI)
GetGPOListW GetGPOList 함수는 지정된 사용자 또는 컴퓨터의 GPO 목록을 검색합니다. (유니코드)
GetProfilesDirectoryA 사용자 프로필이 저장되는 루트 디렉터리의 경로를 검색합니다. (ANSI)
GetProfilesDirectoryW 사용자 프로필이 저장되는 루트 디렉터리의 경로를 검색합니다. (유니코드)
GetProfileType 현재 사용자에 대해 로드된 프로필 유형을 검색합니다.
GetUserProfileDirectoryA 지정된 사용자 프로필의 루트 디렉터리에 대한 경로를 검색합니다. (ANSI)
GetUserProfileDirectoryW 지정된 사용자 프로필의 루트 디렉터리에 대한 경로를 검색합니다. (유니코드)
LeaveCriticalPolicySection LeaveCriticalPolicySection 함수는 정책의 백그라운드 애플리케이션을 다시 시작합니다. 이 함수는 정책 섹션에 대한 핸들을 닫습니다.
LoadUserProfileA

지정된 사용자의 프로필을 로드합니다. 프로필은 로컬 사용자 프로필 또는 로밍 사용자 프로필일 수 있습니다. (ANSI)
LoadUserProfileW 지정된 사용자의 프로필을 로드합니다. 프로필은 로컬 사용자 프로필 또는 로밍 사용자 프로필일 수 있습니다. (유니코드)
ProcessGroupPolicyCompleted ProcessGroupPolicyCompleted 함수는 지정된 확장이 정책 적용을 완료했음을 시스템에 알릴 수 있습니다.
ProcessGroupPolicyCompletedEx ProcessGroupPolicyCompletedEx 함수는 지정된 정책 확장이 정책 적용을 완료했음을 시스템에 알릴 수 있습니다. 또한 함수는 RSoP(결과 정책 집합) 로깅의 상태 보고합니다.
RefreshPolicy RefreshPolicy 함수를 사용하면 정책이 클라이언트 컴퓨터에 즉시 적용됩니다.
RefreshPolicyEx RefreshPolicyEx 함수를 사용하면 정책이 컴퓨터에 즉시 적용됩니다. 확장 함수를 사용하면 적용할 정책 새로 고침 유형을 지정할 수 있습니다.
RegisterGPNotification RegisterGPNotification 함수를 사용하면 정책이 변경될 때 애플리케이션이 알림을 받을 수 있습니다. 정책 변경이 발생하면 지정된 이벤트 개체가 신호 상태로 설정됩니다.
RsopAccessCheckByType RSoPAccessCheckByType 함수는 보안 설명자가 RSOPTOKEN으로 식별된 클라이언트에 지정된 액세스 권한 집합을 부여하는지 여부를 결정합니다.
RsopFileAccessCheck RSoPFileAccessCheck 함수는 파일의 보안 설명자가 RSOPTOKEN으로 식별된 클라이언트에 지정된 파일 액세스 권한 집합을 부여하는지 여부를 결정합니다.
RsopResetPolicySettingStatus RSoPResetPolicySettingStatus 함수는 RSOP_PolicySetting instance RSOP_PolicySettingStatus instance 연결을 해제합니다.
RsopSetPolicySettingStatus RSoPSetPolicySettingStatus 함수는 RSOP_PolicySettingStatus instance RSOP_PolicySettingLink

instance 만듭니다. 함수는 RSOP_PolicySetting instance RSOP_PolicySettingStatus 연결(연결)합니다.
UnloadUserProfile LoadUserProfile 함수에 의해 로드된 사용자의 프로필을 언로드합니다. 호출자는 컴퓨터에 대한 관리 권한이 있어야 합니다. 자세한 내용은 LoadUserProfile 함수의 주의 섹션을 참조하세요.
UnregisterGPNotification UnregisterGPNotification 함수는 정책 변경 알림 수신에서 지정된 정책 알림 핸들의 등록을 취소합니다.

콜백 함수

PFNGENERATEGROUPPOLICY GenerateGroupPolicy 콜백 함수는 계획 모드에서 RSoP 데이터를 생성할 때 각 정책 확장이 내보내야 하는 애플리케이션 정의 콜백 함수입니다.
PFNPROCESSGROUPPOLICY ProcessGroupPolicy 함수는 정책을 적용할 때 사용되는 애플리케이션 정의 콜백 함수입니다.
PFNPROCESSGROUPPOLICYEX ProcessGroupPolicyEx 함수는 정책을 적용할 때 사용되는 애플리케이션 정의 콜백 함수입니다.
PFNSTATUSMESSAGECALLBACK StatusMessageCallback 함수는 정책을 적용할 때 상태 메시지를 표시하는 데 사용되는 애플리케이션 정의 콜백 함수입니다.

구조체

GROUP_POLICY_OBJECTA GROUP_POLICY_OBJECT 구조체는 GPO 목록의 GPO에 대한 정보를 제공합니다. (ANSI)
GROUP_POLICY_OBJECTW

GROUP_POLICY_OBJECT 구조체는 GPO 목록의 GPO에 대한 정보를 제공합니다. (유니코드)
POLICYSETTINGSTATUSINFO POLICYSETTINGSTATUSINFO 구조는 정책 설정 이벤트에 대한 정보를 제공합니다.
RSOP_TARGET RSOP_TARGET 구조에는 GenerateGroupPolicy 함수에 필요한 컴퓨터 및 사용자 정보가 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

usp10.h 헤더

아티클2023. 08. 26.

이 헤더는 Windows 애플리케이션에 대한 국제화에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 애플리케이션에 대한 국제화

usp10.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

ScriptApplyDigitSubstitution 지정된 숫자 대체 설정을 지정된 스크립트 컨트롤 및 스크립트 상태 구조에 적용합니다.
ScriptApplyLogicalWidth 실행에 대한 사전 너비 배열을 사용하고 조정된 사전 문자 모양 너비의 배열을 생성합니다.
ScriptBreak 줄 바꿈을 결정하기 위한 정보를 검색합니다.
ScriptCacheGetHeight 현재 캐시된 글꼴의 높이를 검색합니다.
ScriptCPtoX 실행의 왼쪽 끝 또는 선행 가장자리에서 논리 문자 클러스터의 선행 또는 후행 가장자리로 x 오프셋을 생성합니다.
ScriptFreeCache 스크립트 캐시를 해제합니다.
ScriptGetCMap TrueType cmap 테이블 또는 이전 스타일 글꼴에 대해 구현된 표준 cmap 테이블에 따라 문자열에서 유니코드 문자의 문자 모양 인덱스를 검색합니다.
ScriptGetFontAlternateGlyphs

지정된 OpenType 기능을 통해 액세스할 수 있는 지정된 문자에 대한 대체 문자 모양 목록을 검색합니다.
ScriptGetFontFeatureTags OpenType 처리를 위해 정의된 쓰기 시스템의 입력 기능 목록을 검색합니다. 목록을 구성하는 입력 기능 태그는 제공된 디바이스 컨텍스트 또는 캐시의 글꼴에서 검색됩니다.
ScriptGetFontLanguageTags 지정한 항목에 사용할 수 있고 OpenType 처리를 위해 지정된 스크립트 태그에서 지원되는 언어 태그 목록을 검색합니다. 목록을 구성하는 태그는 지정된 디바이스 컨텍스트 또는 캐시의 글꼴에서 검색됩니다.
ScriptGetFontProperties 글꼴에서 사용하는 특수 문자 모양에 대한 글꼴 캐시에서 정보를 검색합니다.
ScriptGetFontScriptTags OpenType 처리를 위해 글꼴에서 사용할 수 있는 스크립트 목록을 검색합니다. 목록을 구성하는 스크립트는 제공된 디바이스 컨텍스트에 있는 글꼴 또는 현재 실행의 글꼴을 처리하는 스크립트 셰이핑 엔진에서 검색됩니다.
ScriptGetGlyphABCWidth 지정된 문자 모양의 ABC 너비를 검색합니다.
ScriptGetLogicalWidths 특정 글꼴의 문자 모양 앞으로 너비를 논리적 너비로 변환합니다.
ScriptGetProperties 현재 스크립트에 대한 정보를 검색합니다.
ScriptIsComplex 유니코드 문자열에 복잡한 스크립트 처리가 필요한지 여부를 결정합니다.
ScriptItemize 유니코드 문자열을 개별적으로 셰이프 가능한 항목으로 분할합니다.
ScriptItemizeOpenType 유니코드 문자열을 개별적으로 셰이프 가능한 항목으로 나누고 OpenType 처리를 위해 각 셰이프 가능한 항목에 대한 기능 태그 배열을 제공합니다.

ScriptJustify ScriptTextOut 함수에 전달할 때 텍스트 근거를 허용하는 고급 너비 테이블을 만듭니다.
ScriptLayout 실행 포함 수준 배열을 시각적-논리 위치 및/또는 논리적-시각적 위치의 맵으로 변환합니다.
ScriptPlace ScriptShape의 출력에서 문자 모양 사전 너비 및 2차원 오프셋 정보를 생성합니다.
ScriptPlaceOpenType ScriptShapeOpenType의 출력에서 OpenType 정보를 사용하여 유니코드 실행에 대한 문자 모양 및 시각적 특성을 생성합니다.
ScriptPositionSingleGlyph OpenType 처리를 위해 글꼴에 제공된 지정된 기능을 사용하여 단일 조정으로 단일 문자 모양을 배치합니다. 대부분의 경우 애플리케이션은 이 함수를 사용하여 선의 시작 또는 끝에서 문자 모양을 광학적으로 정렬합니다.
ScriptRecordDigitSubstitution NLS(국가 언어 지원) 네이티브 숫자 및 숫자 대체 설정을 읽고 SCRIPT_DIGITSUBSTITUTE 구조로 기록합니다. 자세한 내용은 숫자 셰이프를 참조하세요.
ScriptShape 유니코드 실행에 대한 문자 모양 및 시각적 특성을 생성합니다.
ScriptShapeOpenType OpenType 정보를 사용하여 유니코드 실행에 대한 문자 모양 및 시각적 특성을 생성합니다. 각 실행은 이 함수에 대한 하나의 호출로 구성됩니다.
ScriptString_pcOutChars 클리핑 후 문자열의 길이에 대한 포인터를 반환합니다.
ScriptString_pLogAttr 분석된 문자열에 대한 논리 특성 버퍼에 대한 포인터를 반환합니다.
ScriptString_pSize 분석된 문자열의 SIZE 구조체에 대한 포인터를 반환합니다.

ScriptStringAnalyse 일반 텍스트 문자열을 분석합니다.
ScriptStringCPtoX 문자 위치의 선행 또는 후행 가장자리에 대한 x 좌표를 검색합니다.
ScriptStringFree SCRIPT_STRING_ANALYSIS 구조를 해제합니다.
ScriptStringGetLogicalWidths 시각적 너비를 논리적 너비로 변환합니다.
ScriptStringGetOrder 원래 문자 위치를 문자 모양 위치에 매핑하는 배열을 만듭니다.
ScriptStringOut ScriptStringAnalyse에 대한 이전 호출에서 생성된 문자열을 표시하고 필요에 따라 강조 표시를 추가합니다.
ScriptStringValidate SCRIPT_STRING_ANALYSIS 구조체에서 잘못된 시퀀스를 확인합니다.
ScriptStringXtoCP x 좌표를 문자 위치로 변환합니다.
ScriptSubstituteSingleGlyph OpenType 처리를 위해 동일한 문자 모양의 대체 형식으로 단일 문자 모양을 대체할 수 있습니다.
ScriptTextOut 지정된 스크립트 셰이프에 대한 텍스트를 표시하고 정보를 배치합니다.
ScriptXtoCP 실행의 x 오프셋에서 논리 문자 클러스터의 선행 또는 후행 가장자리를 생성합니다.

구조체

GOFFSET 결합 문자 모양의 x 및 y 오프셋을 포함합니다.
OPENTYPE_FEATURE_RECORD 실행에 적용할 단일 OpenType 기능에 대한 정보를 포함합니다.
SCRIPT_ANALYSIS 유니코드 문자열, 즉 "item"의 일부를 포함합니다.
SCRIPT_CHARPROP 실행의 단일 문자에 대한 정보(입력 문자열)를 포함합니다. 이 정보는 문자 문자 모양이 실행의 주변 문자에 의해 영향을 받는지 나타냅니다.
SCRIPT_CONTROL 여러 Uniscribe 함수(예: ScriptItemize)에 대한 스크립트 컨트롤 플래그를 포함합니다.
SCRIPT_DIGITSUBSTITUTE 네이티브 숫자 및 숫자 대체 설정을 포함합니다.
SCRIPT_FONTPROPERTIES 현재 글꼴의 속성에 대한 정보를 포함합니다.
SCRIPT_GLYPHPROP 출력 문자 모양 배열의 일부인 문자 모양에 대한 정보를 포함합니다.
SCRIPT_ITEM 항목의 첫 번째 문자에 대한 문자열 오프셋이 있는 SCRIPT_ANALYSIS 구조를 포함하여 스크립트 항목을 포함합니다.
SCRIPT_LOGATTR 텍스트를 편집하고 서식을 지정할 때 유용한 논리 문자의 특성을 포함합니다.
SCRIPT_PROPERTIES 각 스크립트에 대한 특수 처리에 대한 정보를 포함합니다.
SCRIPT_STATE 스크립트 상태 정보를 포함합니다.

SCRIPT_TABDEF ScriptStringAnalyse에 대한 탭 위치의 정의를 포함합니다.
SCRIPT_VISATTR ScriptShape에서 생성된 클러스터 및 근거 지점을 식별하는 시각적 개체(문자 모양) 특성을 포함합니다.
TEXTRANGE_PROPERTIES 실행에 적용할 OpenType 기능 그룹을 포함합니다.

열거형

SCRIPT_JUSTIFY 애플리케이션에서 근거를 구현하는 데 필요한 문자 모양 특성 정보를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

utilapiset.h 헤더

아티클2023. 08. 26.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스
- Windows 사이드바

utilapiset.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

Beep 스피커에서 간단한 톤을 생성합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

uxtheme.h 헤더

아티클2023. 08. 26.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 높은 DPI
- 터치 입력
- Windows 컨트롤

uxtheme.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

BeginBufferedAnimation 버퍼링된 애니메이션 작업을 시작합니다. 애니메이션은 지정된 시간 동안 두 버퍼의 콘텐츠 간 교차 페이드로 구성됩니다.
BeginBufferedPaint 버퍼링된 페인트 작업을 시작합니다.
BeginPanningFeedback 패닝 제스처의 영향을 받는 대상 창에 대한 피드백을 보내도록 시스템에 알립니다.
BufferedPaintClear 버퍼에서 지정된 사각형을 ARGB = {0,0,0,0}으로 지웁니다.
BufferedPaintInit 현재 스레드에 대해 버퍼링된 페인팅을 초기화합니다.
BufferedPaintRenderAnimation 버퍼링된 페인트 애니메이션의 다음 프레임을 그립니다.
BufferedPaintSetAlpha 알파를 지정된 사각형의 지정된 값으로 설정합니다. 알파는 대상 DC(대상 디바이스 컨텍스트)에서 버퍼와 혼합할 때 적용되는 투명도의 양을 제어합니다.
BufferedPaintStopAllAnimations

지정된 창에 대해 버퍼링된 모든 애니메이션을 중지합니다.
BufferedPaintUnInit 현재 스레드에 대한 버퍼링된 페인팅을 닫습니다. BeginBufferedPaint에 대한 호출이 더 이상 필요하지 않은 후 BufferedPaintInit에 대한 각 호출에 대해 한 번씩 호출됩니다.
CloseThemeData 테마 데이터 핸들을 닫습니다.
DrawThemeBackground 지정된 컨트롤 파트에 대한 시각적 스타일로 정의된 테두리 및 채우기를 그립니다.
DrawThemeBackgroundEx 지정된 컨트롤 파트에 대한 시각적 스타일로 정의된 배경 이미지를 그립니다.
DrawThemeEdge 직사각형의 시각적 스타일로 정의된 하나 이상의 모서리를 그립니다.
DrawThemeIcon 시각적 스타일로 정의된 아이콘 효과를 사용하여 이미지 목록에서 이미지를 그립니다.
DrawThemeParentBackground 부분적으로 투명하거나 알파가 혼합된 자식 컨트롤로 덮힌 부모 컨트롤의 파트를 그립니다.
DrawThemeParentBackgroundEx 부분적으로 투명하거나 알파가 혼합된 자식 컨트롤에서 부모 컨트롤이 표시되는 파트를 그리는 데 사용됩니다. WM_ERASEBKGND 메시지와 WM_PRINTCLIENT를 보냅니다.
DrawThemeText 시각적 스타일에서 정의한 색과 글꼴을 사용하여 텍스트를 그립니다.
메서드의 시각적 스타일에서 정의한 색과 글꼴을 사용하여 텍스트를 그립니다. 추가 텍스트 형식 옵션을 허용하여 DrawThemeText를 확장합니다.
EnableThemeDialogTexture 대화 상자 창 배경의 시각적 스타일을 사용하거나 사용하지 않도록 설정합니다.

EnableTheming Windows Vista에서 Windows 7:_Enables 또는 현재 및 이후 세션의 현재 사용자에 대한 시각적 스타일을 사용하지 않도록 설정합니다. Windows 8 이상:_This 함수는 아무 것도 수행하지 않습니다. 시각적 스타일은 항상 Windows 8 이상에서 사용하도록 설정됩니다.
EndBufferedAnimation 버퍼링된 애니메이션 작업의 첫 번째 프레임을 렌더링하고 애니메이션 타이머를 시작합니다.
EndBufferedPaint 버퍼링된 페인트 작업을 완료하고 연결된 버퍼링된 페인트 핸들을 해제합니다.
EndPanningFeedback 진행 중이거나 BeginPanningFeedback 및 UpdatePanningFeedback에 의해 설정된 기존 애니메이션을 종료합니다.
GetBufferedPaintBits 버퍼가 DIB(디바이스 독립적 비트맵)인 경우 버퍼 비트맵에 대한 포인터를 검색합니다.
GetBufferedPaintDC 그림판 디바이스 컨텍스트(DC)를 가져옵니다. 이는 BeginBufferedPaint에서 검색한 것과 동일한 값입니다.
GetBufferedPaintTargetDC DC(대상 디바이스 컨텍스트)를 검색합니다.
GetBufferedPaintTargetRect BeginBufferedPaint에서 지정한 대상 사각형을 검색합니다.
GetCurrentThemeName 현재 시각적 스타일의 이름을 검색하고 선택적으로 색 구성표 이름과 크기 이름을 검색합니다.
GetThemeAnimationProperty 스토리보드 ID 및 대상 ID를 기반으로 하는 테마 애니메이션 속성을 가져옵니다.
GetThemeAnimationTransform 스토리보드 ID, 대상 ID 및 변환 인덱스 기반의 애니메이션 변환 작업을 가져옵니다.

GetThemeAppProperties 현재 애플리케이션에서 시각적 스타일이 적용되는 방식을 제어하는 속성 플래그를 검색합니다.
GetThemeBackgroundContentRect 시각적 스타일로 정의된 배경에 대한 콘텐츠 영역의 크기를 검색합니다.
GetThemeBackgroundExtent 콘텐츠 영역에 따라 시각적 스타일로 정의된 배경의 크기와 위치를 계산합니다.
GetThemeBackgroundRegion 지정된 사각형으로 둘러싸인 일반 또는 부분적으로 투명한 배경에 대한 영역을 계산합니다.
GetThemeBitmap 특정 테마, 파트, 상태 및 속성과 관련된 비트맵을 검색합니다.
GetThemeBool 테마 데이터의 SysMetrics 섹션에서 BOOL 속성 값을 검색합니다.
GetThemeColor 색 속성 값을 검색합니다.
GetThemeDocumentationProperty 지정된 테마 파일의 설명서 섹션에서 테마 속성 값을 검색합니다.
GetThemeEnumValue 열거된 유형 속성의 값을 검색합니다.
GetThemeFilename 파일 이름 속성 값을 검색합니다.
GetThemeFont 글꼴 속성 값을 검색합니다.
GetThemeInt int 속성 값을 검색합니다.
GetThemeIntList

시각적 스타일에서 int 데이터 목록을 검색합니다.
GetThemeMargins MARGINS 속성 값을 검색합니다.
GetThemeMetric 메트릭 속성의 값을 검색합니다.
GetThemePartSize 시각적 스타일로 정의된 파트의 원래 크기를 계산합니다.
GetThemePosition 위치 속성의 값을 검색합니다.
GetThemePropertyOrigin 속성에 대한 테마 속성 정의의 위치를 검색합니다.
GetThemeRect RECT 속성 값을 검색합니다.
GetThemeStream 지정된 파트, 상태 및 속성에서 시작하여 지정된 테마에 해당하는 데이터 스트림을 검색합니다.
GetThemeString 문자열 속성의 값을 검색합니다.
GetThemeSysBool 시스템 메트릭의 부울 값을 검색합니다.
GetThemeSysColor 시스템 색의 값을 검색합니다.
GetThemeSysColorBrush 시스템 색 브러시를 검색합니다.
GetThemeSysFont 시스템 글꼴의 LOGFONT를 검색합니다.

GetThemeSysInt 시스템 int의 값을 검색합니다.
GetThemeSysSize 테마 데이터에서 시스템 크기 메트릭 값을 검색합니다.
GetThemeSysString 시스템 문자열의 값을 검색합니다.
GetThemeTextExtent 시각적 스타일 글꼴로 렌더링될 때 지정된 텍스트의 크기와 위치를 계산합니다.
GetThemeTextMetrics 특정 파트의 시각적 스타일로 지정된 글꼴에 대한 정보를 검색합니다.
GetThemeTimingFunction 타이밍 함수 식별자를 기반으로 미리 정의된 타이밍 함수를 가져옵니다.
GetThemeTransitionDuration 지정된 전환의 기간을 가져옵니다.
GetWindowTheme 시각적 스타일이 적용된 창에 대한 테마 핸들을 검색합니다.
HitTestThemeBackground 시각적 스타일로 지정된 배경의 지점에 대한 적중 테스트 코드를 검색합니다.
IsAppThemed 현재 애플리케이션의 사용자 인터페이스가 시각적 스타일을 사용하여 표시되는지 여부를 보고합니다.
IsCompositionActive DWM(바탕 화면 창 관리자) 구성 효과를 테마에 사용할 수 있는지 여부를 결정합니다.
IsThemeActive 현재 애플리케이션의 시각적 스타일이 활성화되어 있는지 테스트합니다.

IsThemeBackgroundPartiallyTransparent 시각적 스타일로 지정된 배경에 투명한 부분이 있는지 또는 알파가 혼합된 부분이 있는지 검색합니다.
IsThemeDialogTextureEnabled 지정된 대화창이 배경 텍스처링을 지원하는지 여부를 보고합니다.
IsThemePartDefined 시각적 스타일에 지정된 파트 및 상태에 대한 정의된 매개 변수가 있는지 여부를 검색합니다.
OpenThemeData 창 및 관련 클래스에 대한 테마 데이터를 엽니다.
OpenThemeDataEx 지정된 테마 클래스에 대한 창과 연결된 테마 데이터를 엽니다.
OpenThemeDataForDpi 특정 DPI와 연결된 테마 핸들을 여는 OpenThemeData의 변형입니다.
SetThemeAppProperties 호출 애플리케이션에서 시각적 스타일이 구현되는 방식을 결정하는 플래그를 설정합니다.
SetWindowTheme 창에서 해당 클래스가 일반적으로 사용하는 것과 다른 시각적 스타일 정보 집합을 사용하도록 합니다.
SetWindowThemeAttribute 시각적 스타일이 지정된 창에 적용되는 방식을 제어하는 특성을 설정합니다.
SetWindowThemeNonClientAttributes 비클라이언트 특성을 설정하여 시각적 스타일이 지정된 창에 적용되는 방식을 제어합니다.
UpdatePanningFeedback 패닝 제스처로 인한 창 상태에 대해 클라이언트를 업데이트합니다. 이 함수는 BeginPanningFeedback 호출 후에만 호출할 수 있습니다.

구조체

BP_ANIMATIONPARAMS BeginBufferedPaint에서 사용하는 BP_PAINTPARAMS 구조체에 대한 애니메이션 매개 변수를 정의합니다.
BP_PAINTPARAMS BeginBufferedPaint에 대한 페인트 작업 매개 변수를 정의합니다.
DTBGOPTS DrawThemeBackgroundEx 함수에 대한 옵션을 정의합니다.
DTTOPTS DrawThemeTextEx 함수에 대한 옵션을 정의합니다.
INTLIST 시각적 스타일의 int 데이터 항목 목록 또는 배열을 포함합니다.
MARGINS 시각적 스타일이 적용된 창의 여백을 정의하기 위해 GetThemeMargins 함수에 의해 반환됩니다.
WTA_OPTIONS 창 시각적 스타일 특성을 설정하는 데 사용되는 옵션을 정의합니다.

열거형

BP_ANIMATIONSTYLE BP_ANIMATIONPARAMS 구조에서 애니메이션 옵션을 선언하는 데 사용됩니다.
BP_BUFFERFORMAT 버퍼의 형식을 지정합니다. BeginBufferedAnimation 및 BeginBufferedPaint에서 사용됩니다.
PROPERTYORIGIN 속성이 발견된 위치를 지정하기 위해 GetThemePropertyOrigin에 의해 반환됩니다.

THEMESIZE 비주얼 스타일 부분에 대해 검색할 크기 값의 형식을 식별합니다.
WINDOWTHEMEATTRIBUTETYPE 창에 설정할 시각적 스타일 특성의 형식을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

vbinterf.h 헤더

아티클2023. 08. 26.

이 헤더는 COM(구성 요소 개체 모델)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)

vbinterf.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IGetOleObject Visual Basic 컨테이너의 OLE 컨트롤에 대한 액세스를 가져옵니다.
IGetVBAObject IVBFormat 인터페이스에 대한 액세스를 가져옵니다.
IVBFormat Visual Basic 컨테이너의 컨트롤에 대한 서식 지정 지원을 제공합니다.
IVBGetControl Visual Basic 컨테이너의 모든 컨트롤에 대한 액세스를 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

vdmdbg.h 헤더

아티클2023. 08. 26.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

vdmdbg.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

VDMEnumProcessWOW 16비트 Windows 작업을 실행하는 모든 가상 DOS 머신을 열거합니다.
VDMEnumTaskWOWEx 특정 VDM(가상 DOS 머신) 내에서 작업을 열거합니다.

콜백 함수

PROCESSENUMPROC VDMEnumProcessWOW가 열거하는 각 VDM(가상 DOS 머신)에 대한 정보를 수신하도록 이 함수를 구현합니다.
TASKENUMPROCEX VDMEnumTaskWOWEx가 열거하는 각 작업에 대한 정보를 수신하도록 이 함수를 구현합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

vds.h 헤더

아티클2023. 08. 26.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Hyper-V WMI 공급자
- 스토리지 볼륨 공급자
- 시스템 서비스

vds.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumVdsObject IEnumVdsObject 인터페이스(vds.h)는 지정된 형식의 VDS 개체 집합을 통해 열거합니다.
IVdsAdvancedDisk 파티션을 만들고 삭제하고 파티션 특성을 수정합니다.
IVdsAdvancedDisk2 파티션 형식을 변경하는 메서드를 제공합니다.
IVdsAdviseSink IVdsAdviseSink 인터페이스(vds.h)는 VDS 알림을 받습니다.
IVdsAsync IVdsAsync 인터페이스(vds.h)는 비동기 작업을 관리합니다.
IVdsController IVdsController 인터페이스(vds.h)는 컨트롤러에서 쿼리 및 구성 작업을 수행하기 위한 메서드를 제공합니다.
IVdsControllerControllerPort IVdsControllerControllerPort 인터페이스(vds.h)는 IVdsController 인터페이스를 구현하는 클래스에 대한 컨트롤러 포트를 열거하는 메서드를 제공합니다.
IVdsControllerPort

IVdsControllerPort 인터페이스(vds.h)는 컨트롤러 포트에서 쿼리 및 구성 작업을 수행하기 위한 메서드를 제공합니다.
IVdsCreatePartitionEx 기본 디스크에 파티션을 만듭니다.
IVdsDisk 기본 및 동적 디스크를 쿼리하고 구성하는 메서드를 제공합니다.
IVdsDisk2 디스크의 SAN 모드를 오프라인 또는 온라인으로 설정하는 방법을 제공합니다.
IVdsDisk3 디스크의 위치 경로를 포함하여 디스크에 대한 속성 정보를 검색하는 메서드를 제공합니다.
IVdsDiskOnline 디스크를 온라인 상태로 전환하고 오프라인으로 전환할 수 있는 메서드를 제공합니다. Windows Vista: Windows Vista SP1(서비스 팩 1)이 있는 Windows Vista까지는 이 인터페이스가 지원되지 않습니다. 대신 IVdsDisk2를 사용합니다.
IVdsDiskPartitionMF 파티션에서 파일 시스템 관리 작업을 수행하는 메서드를 제공합니다.
IVdsDiskPartitionMF2 추가 서식 옵션을 사용하여 파티션의 서식을 지정하는 메서드를 제공합니다.
IVdsDrive IVdsDrive 인터페이스(vds.h)는 드라이브에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsDrive2 IVdsDrive2 인터페이스(vds.h)는 드라이브의 속성을 쿼리하는 메서드를 제공합니다.
IVdsHbaPort 로컬 시스템에서 HBA 포트를 쿼리하고 상호 작용하는 메서드를 제공합니다.
IVdsHwProvider IVdsHwProvider 인터페이스(vds.h)는 하드웨어 공급자에서 쿼리, 다시 열거 및 새로 고침 작업을 수행하는 메서드를 제공합니다.

IVdsHwProviderStoragePools IVdsHwProviderStoragePools 인터페이스(vds.h)는 스토리지 풀에서 LUN을 만들고 하드웨어 공급자가 관리하는 스토리지 풀을 열거하는 메서드를 제공합니다.
IVdsHwProviderType IVdsHwProviderType 인터페이스(vds.h)는 하드웨어 공급자 유형을 검색하는 메서드를 제공합니다.
IVdsHwProviderType2 IVdsHwProviderType2 인터페이스(vds.h)는 구현되지 않습니다. 대신 IVdsHwProviderType을 사용합니다.
IVdsIscsiInitiatorAdapter 로컬 시스템에서 iSCSI 초기자 어댑터를 쿼리하고 상호 작용하는 메서드를 제공합니다.
IVdsIscsiInitiatorPortal 로컬 시스템의 iSCSI 초기자 포털을 쿼리하고 상호 작용하는 메서드를 제공합니다.
IVdsIscsiPortal IVdsIscsiPortal 인터페이스(vds.h)는 iSCSI 포털에서 쿼리 및 구성 작업을 수행하기 위한 메서드를 제공합니다.
IVdsIscsiPortalGroup IVdsIscsiPortalGroup 인터페이스(vds.h)는 iSCSI 포털 그룹에서 쿼리 및 구성 서비스를 수행하기 위한 메서드를 제공합니다.
IVdsIscsiPortalLocal iSCSI 포털에서 로컬 초기자별 IPSEC 사전 공유 키를 설정하는 메서드를 제공합니다.
IVdsIscsiTarget IVdsIscsiTarget 인터페이스(vds.h)는 iSCSI 대상에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsLun IVdsLun 인터페이스(vds.h)는 LUN(논리 단위 번호)에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsLun2

IVdsLun2 인터페이스(vds.h)는 LUN(논리 단위 번호) 힌트를 적용하고 쿼리하는 메서드를 제공합니다.
IVdsLunControllerPorts IVdsLunControllerPorts 인터페이스(vds.h)는 LUN에서 컨트롤러 포트 구성 작업을 수행하기 위한 메서드를 제공합니다.
IVdsLunIscsi IVdsLunIscsi 인터페이스(vds.h)는 iSCSI LUN에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsLunMpio IVdsLunMpio 인터페이스(vds.h)는 MPIO 확장을 사용하여 LUN에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsLunNaming IVdsLunNaming 인터페이스(vds.h)는 IVdsLun 인터페이스를 구현하는 클래스의 LUN 이름을 지정하는 메서드를 제공합니다.
IVdsLunNumber IVdsLunNumber 인터페이스(vds.h)는 LUN에 대한 LUN 번호를 쿼리하는 메서드를 제공합니다.
IVdsLunPlex IVdsLunPlex 인터페이스(vds.h)는 LUN 플렉스에서 쿼리 및 구성 작업을 수행하기 위한 메서드를 제공합니다.
IVdsMaintenance IVdsMaintenance 인터페이스(vds.h)는 하위 시스템, 컨트롤러, LUN 또는 드라이브에서 유지 관리 작업을 수행하는 메서드를 제공합니다.
IVdsOpenVDisk 가상 디스크를 관리하기 위한 메서드를 정의합니다. (IVdsOpenVDisk)
IVdsPack 디스크 및 볼륨이 포함된 팩에서 관리 작업을 쿼리하고 수행하는 메서드를 제공합니다.
IVdsPack2 팩에 정렬된 볼륨을 만드는 메서드를 제공합니다.

IVdsProvider IVdsProvider 인터페이스(vds.h)는 하드웨어 또는 소프트웨어 공급자의 속성을 반환합니다.
IVdsProviderSupport IVdsProviderSupport 인터페이스(vds.h)는 공급자가 지원하는 VDS 인터페이스 버전을 나타내는 메서드를 제공합니다.
IVdsRemovable CD-ROM과 같은 이동식 디스크를 쿼리하고 꺼내는 메서드를 제공합니다.
IVdsService VDS를 쿼리하고 상호 작용하는 메서드를 제공합니다.
IVdsServiceHba 로컬 시스템에서 HBA 포트를 쿼리하는 메서드를 제공합니다.
IVdsServiceInitialization 이 인터페이스는 시스템 사용을 위해 예약되어 있습니다.
IVdsServiceIscsi CHAP 보안 설정을 설정하고 대상에 로그인하는 기능을 포함하여 로컬 초기자 서비스와 인터페이스하는 메서드를 제공합니다.
IVdsServiceLoader 서비스 로더 개체를 인스턴스화합니다.
IVdsServiceSAN 운영 체제에 대한 온라인 및 오프라인 SAN 정책을 관리하는 방법을 제공합니다.
IVdsServiceUninstallDisk 기본 및 동적 디스크를 제거하는 메서드를 제공합니다.
IVdsStoragePool IVdsStoragePool 인터페이스(vds.h)는 정보를 쿼리하고 스토리지 풀에 대한 관련 개체를 열거하는 메서드를 제공합니다.
IVdsSubSystem

IVdsSubSystem 인터페이스(vds.h)는 하위 시스템에 대한 쿼리 및 구성 작업을 수행하기 위한 메서드를 제공합니다.
IVdsSubSystem2 IVdsSubSystem2 인터페이스(vds.h)는 VDS_HINTS2 및 VDS_SUB_SYSTEM_PROP2 구조를 사용하여 하위 시스템에 대한 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsSubSystemImportTarget 하위 시스템에 대한 기본 VSS 가져오기 대상을 쿼리하고 구성하는 메서드를 제공합니다.
IVdsSubSystemInterconnect IVdsSubSystemInterconnect 인터페이스(vds.h)는 하위 시스템이 지원하는 상호 연결 형식을 쿼리하는 메서드를 제공합니다.
IVdsSubSystemIscsi IVdsSubSystemIscsi 인터페이스(vds.h)는 하위 시스템의 iSCSI 대상 및 포털을 쿼리하고 구성하는 메서드를 제공합니다.
IVdsSubSystemNaming IVdsSubSystemNaming 인터페이스(vds.h)는 IVdsSubSystem 인터페이스를 구현하는 클래스에 대한 하위 시스템의 이름을 지정하는 메서드를 제공합니다.
IVdsSwProvider 소프트웨어 공급자와 관련된 작업을 수행하는 메서드를 제공합니다.
IVdsVDisk 가상 디스크를 관리하기 위한 메서드를 정의합니다. (IVdsVDisk)
IVdsVdProvider 가상 디스크를 만들고 관리하기 위한 메서드를 정의합니다.
IVdsVolume 볼륨을 관리하는 메서드를 제공합니다.
IVdsVolume2 볼륨 GUID를 포함하여 볼륨 속성 정보를 반환하는 메서드를 제공합니다.
IVdsVolumeMF

볼륨 개체에서 액세스 경로 및 파일 시스템 작업을 수행하는 메서드를 제공합니다.
IVdsVolumeMF2 볼륨 개체에 대한 추가 파일 시스템 관리 작업을 수행하는 메서드를 제공합니다. (IVdsVolumeMF2)
IVdsVolumeMF3 볼륨 개체에 대한 추가 파일 시스템 관리 작업을 수행하는 메서드를 제공합니다. (IVdsVolumeMF3)
IVdsVolumeOnline 단일 볼륨을 복구하는 메서드를 제공합니다.
IVdsVolumePlex 볼륨 플렉스를 유지 관리하기 위한 메서드를 제공합니다.
IVdsVolumeShrink 볼륨 축소를 지원하는 메서드를 제공합니다.

구조체

CHANGE_ATTRIBUTES_PARAMETERS 파티션 스타일의 파티션 매개 변수를 정의합니다. (CHANGE_ATTRIBUTES_PARAMETERS)
CHANGE_PARTITION_TYPE_PARAMETERS 파티션의 형식을 변경할 때 사용할 매개 변수에 대해 설명합니다.
CREATE_PARTITION_PARAMETERS 파티션 스타일의 파티션 매개 변수를 정의합니다. (CREATE_PARTITION_PARAMETERS)
VDS_ASYNC_OUTPUT VDS_ASYNC_OUTPUT 구조체(vds.h)는 비동기 개체의 출력을 정의합니다. 출력 요소는 작업 유형에 따라 달라집니다.
VDS_CONTROLLER_NOTIFICATION VDS_CONTROLLER_NOTIFICATION 구조체(vds.h)는 컨트롤러 이벤트의 세부 정보를 정의합니다.

VDS_CONTROLLER_PROP VDS_CONTROLLER_PROP 구조체(vds.h)는 컨트롤러 개체의 속성을 정의합니다.
VDS_CREATE_VDISK_PARAMETERS 가상 디스크를 만들 때 사용할 매개 변수를 포함합니다.
VDS_DISK_EXTENT 디스크 익스텐트 속성을 정의합니다.
VDS_DISK_FREE_EXTENT 디스크의 사용 가능한 범위에 대해 설명합니다.
VDS_DISK_NOTIFICATION VDS_DISK_NOTIFICATION 구조체(vds.h)는 디스크 이벤트의 세부 정보를 정의합니다.
VDS_DISK_PROP 디스크 개체의 속성을 정의합니다.
VDS_DISK_PROP2 디스크 개체의 속성을 정의합니다. 이 구조체는 위치 경로도 포함하고 디스크가 오프라인인 경우 오프라인인 이유를 제외하고 VDS_DISK_PROP 구조와 동일합니다.
VDS_DRIVE_EXTENT VDS_DRIVE_EXTENT 구조체(vds.h)는 드라이브 익스텐트 속성을 정의합니다.
VDS_DRIVE_LETTER_NOTIFICATION VDS_DRIVE_LETTER_NOTIFICATION 구조체(vds.h)는 드라이브 문자 이벤트의 세부 정보를 정의합니다.
VDS_DRIVE_LETTER_PROP 드라이브 문자의 속성을 정의합니다.
VDS_DRIVE_NOTIFICATION VDS_DRIVE_NOTIFICATION 구조체(vds.h)는 드라이브 이벤트의 세부 정보를 정의합니다.
VDS_DRIVE_PROP VDS_DRIVE_PROP 구조체(vds.h)는 드라이브 개체의 속성을 정의합니다.

VDS_DRIVE_PROP2 VDS_DRIVE_PROP2 구조체(vds.h)는 드라이브 개체의 속성을 정의합니다.
VDS_FILE_SYSTEM_FORMAT_SUPPORT_PROP 볼륨 서식 지정에 지원되는 파일 시스템에 대한 정보를 제공합니다.
VDS_FILE_SYSTEM_NOTIFICATION VDS_FILE_SYSTEM_NOTIFICATION 구조체(vds.h)는 파일 시스템 이벤트의 세부 정보를 정의합니다.
VDS_FILE_SYSTEM_PROP 파일 시스템의 속성을 정의합니다.
VDS_FILE_SYSTEM_TYPE_PROP 파일 시스템 형식의 속성을 정의합니다.
VDS_HBAPORT_PROP VDS_HBAPORT_PROP 구조체(vds.h)는 HBA 포트의 속성을 정의합니다.
VDS_HINTS VDS_HINTS 구조체(vds.h)는 LUN 또는 LUN 플렉스에 대한 자동 매핑 힌트를 정의합니다.
VDS_HINTS2 VDS_HINTS2 구조체(vds.h)에는 스토리지 풀의 LUN에 대한 자동 매핑 힌트가 포함되어 있습니다.
VDS_INPUT_DISK 입력 디스크의 세부 정보를 정의합니다.
VDS_IPADDRESS VDS_IPADDRESS 구조체(vds.h)는 IP 주소와 포트를 정의합니다.
VDS_ISCSI_INITIATOR_ADAPTER_PROP VDS_ISCSI_INITIATOR_ADAPTER_PROP 구조체(vds.h)는 iSCSI 초기자 어댑터의 속성을 정의합니다.
VDS_ISCSI_INITIATOR_PORTAL_PROP VDS_ISCSI_INITIATOR_PORTAL_PROP 구조체(vds.h)는 iSCSI 초기자 포털의 속성을 정의합니다.
VDS_ISCSI_IPSEC_KEY

VDS_ISCSI_IPSEC_KEY 구조체(vds.h)는 지원되지 않습니다. 이 구조체는 나중에 사용하도록 예약되어 있습니다.
VDS_ISCSI_PORTAL_PROP VDS_ISCSI_PORTAL_PROP 구조체(vds.h)는 iSCSI 포털의 속성을 정의합니다. (VDS_ISCSI_PORTAL_PROP)
VDS_ISCSI_PORTALGROUP_PROP VDS_ISCSI_PORTALGROUP_PROP 구조체(vds.h)는 iSCSI 포털 그룹의 속성을 정의합니다.
VDS_ISCSI_SHARED_SECRET VDS_ISCSI_SHARED_SECRET 구조체(vds.h)는 CHAP 공유 비밀을 정의합니다.
VDS_ISCSI_TARGET_PROP VDS_ISCSI_TARGET_PROP 구조체(vds.h)는 iSCSI 대상의 속성을 정의합니다.
VDS_LUN_NOTIFICATION VDS_LUN_NOTIFICATION 구조체(vds.h)는 LUN 알림의 세부 정보를 정의합니다.
VDS_LUN_PLEX_PROP VDS_LUN_PLEX_PROP 구조체(vds.h)는 LUN plex 개체의 속성을 정의합니다.
VDS_LUN_PROP VDS_LUN_PROP 구조체(vds.h)는 LUN 개체의 속성을 정의합니다.
VDS_MOUNT_POINT_NOTIFICATION VDS_MOUNT_POINT_NOTIFICATION 구조체(vds.h)는 드라이브 문자 또는 볼륨 GUID 경로가 변경되어 소프트웨어 공급자가 반환한 알림 정보를 나타냅니다.
VDS_NOTIFICATION VDS_NOTIFICATION 구조체(vds.h)는 각 알림 대상 유형과 관련된 VDS 알림 구조를 정의합니다.
VDS_PACK_NOTIFICATION VDS_PACK_NOTIFICATION 구조체(vds.h)는 팩 이벤트의 세부 정보를 정의합니다.
VDS_PACK_PROP pack 개체의 속성을 정의합니다.

VDS_PARTITION_INFO_GPT GPT(GUID 파티션 테이블) 파티션의 세부 정보를 정의합니다.
VDS_PARTITION_INFO_MBR MBR(master 부팅 레코드) 파티션의 세부 정보를 정의합니다.
VDS_PARTITION_INFORMATION_EX 이 구조체는 시스템 사용을 위해 예약되어 있습니다. (VDS_PARTITION_INFORMATION_EX)
VDS_PARTITION_NOTIFICATION VDS_PARTITION_NOTIFICATION 구조체(vds.h)는 파티션 이벤트의 세부 정보를 정의합니다.
VDS_PARTITION_PROP 파티션의 속성을 정의합니다.
VDS_PATH_ID VDS_PATH_ID 구조체(vds.h)는 경로에 대한 고유 ID를 정의합니다.
VDS_PATH_INFO VDS_PATH_INFO 구조체(vds.h)는 LUN 경로에 대한 정보를 정의합니다.
VDS_PATH_POLICY VDS_PATH_POLICY 구조체(vds.h)는 특정 경로에 적용되는 부하 분산 정책을 정의합니다.
VDS_POOL_ATTRIBUTES VDS_POOL_ATTRIBUTES 구조체(vds.h)는 스토리지 풀의 특성을 정의합니다.
VDS_POOL_CUSTOM_ATTRIBUTES VDS_POOL_CUSTOM_ATTRIBUTES 구조체(vds.h)는 스토리지 풀의 사용자 지정 특성을 정의합니다.
VDS_PORT_NOTIFICATION VDS_PORT_NOTIFICATION 구조체(vds.h)는 컨트롤러 포트 이벤트의 세부 정보를 정의합니다.
VDS_PORT_PROP VDS_PORT_PROP 구조체(vds.h)는 컨트롤러 개체의 포트 속성을 정의합니다.
VDS_PORTAL_GROUP_NOTIFICATION

VDS_PORTAL_GROUP_NOTIFICATION 구조체(vds.h)는 iSCSI 포털 이벤트의 세부 정보를 정의합니다.
VDS_PORTAL_NOTIFICATION VDS_PORTAL_NOTIFICATION 구조체(vds.h)는 iSCSI 포털 이벤트의 세부 정보를 정의합니다.
VDS_PROVIDER_PROP VDS_PROVIDER_PROP 구조체(vds.h)는 공급자 개체의 속성을 정의합니다.
VDS_REPARSE_POINT_PROP 볼륨 개체의 재분석 지점 속성을 정의합니다.
VDS_SERVICE_PROP 서비스 개체의 속성을 정의합니다.
VDS_STORAGE_POOL_DRIVE_EXTENT VDS_STORAGE_POOL_DRIVE_EXTENT 구조체(vds.h)는 스토리지 풀에서 사용할 수 있는 드라이브 익스텐트 를 정의합니다.
VDS_STORAGE_POOL_PROP VDS_STORAGE_POOL_PROP 구조체(vds.h)는 스토리지 풀 개체의 속성을 정의합니다.
VDS_SUB_SYSTEM_NOTIFICATION VDS_SUB_SYSTEM_NOTIFICATION 구조체(vds.h)는 하위 시스템 이벤트의 세부 정보를 정의합니다.
VDS_SUB_SYSTEM_PROP VDS_SUB_SYSTEM_PROP 구조체(vds.h)는 하위 시스템 개체의 속성을 정의합니다.
VDS_SUB_SYSTEM_PROP2 VDS_SUB_SYSTEM_PROP2 구조체(vds.h)는 하위 시스템 개체의 속성을 정의합니다.
VDS_TARGET_NOTIFICATION VDS_TARGET_NOTIFICATION 구조체(vds.h)는 iSCSI 대상 이벤트의 세부 정보를 정의합니다.
VDS_VDISK_PROPERTIES 가상 디스크의 속성을 정의합니다.
VDS_VOLUME_NOTIFICATION

VDS_VOLUME_NOTIFICATION 구조체(vds.h)는 볼륨 이벤트의 세부 정보를 정의합니다.
VDS_VOLUME_PLEX_PROP 볼륨 플렉스 개체의 속성을 정의합니다.
VDS_VOLUME_PROP 볼륨 개체의 속성을 정의합니다.
VDS_VOLUME_PROP2 볼륨 개체의 속성을 정의합니다. 이 구조체는 볼륨 GUID도 포함한다는 점을 제외하고 VDS_VOLUME_PROP 구조체와 동일합니다.
VDS_WWN VDS_WWN 구조체(vds.h)는 WWN(전 세계 이름)을 정의합니다. 이 구조체는 ANSI HBA API에서 정의한 HBA_WWN 구조체에 해당합니다.

열거형

__VDS_PARTITION_STYLE 이 열거형은 명시적으로 사용되지 않습니다.
VDS_ASYNC_OUTPUT_TYPE VDS_ASYNC_OUTPUT_TYPE(vds.h) 열거형은 개체가 처리할 수 있는 작업 집합을 정의합니다.
VDS_CONTROLLER_STATUS VDS_CONTROLLER_STATUS(vds.h) 열거형은 컨트롤러에 대한 개체 상태 값 집합을 정의합니다.
VDS_DISK_EXTENT_TYPE 디스크 익스텐트 유형 집합을 정의합니다. 형식은 파티션, 볼륨 또는 사용 가능한 공간일 수 있습니다.
VDS_DISK_FLAG 디스크 개체의 유효한 플래그 집합을 정의합니다. 참고 VDS_DF_READ_ONLY 경우를 제외하고 이러한 플래그는 IVdsDisk::SetFlags 메서드를 사용하여 설정하거나 IVdsDisk::ClearFlags 메서드를 사용하여 지울 수 없습니다.

VDS_DISK_OFFLINE_REASON 디스크가 오프라인 상태가 되는 이유 집합을 정의합니다.
VDS_DISK_STATUS 디스크에 대한 개체 상태 값 집합을 정의합니다.
VDS_DRIVE_FLAG VDS_DRIVE_FLAG(vds.h) 열거형은 드라이브 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_DRIVE_LETTER_FLAG 드라이브 문자에 대한 유효한 플래그 집합을 정의합니다.
VDS_DRIVE_STATUS VDS_DRIVE_STATUS(vds.h) 열거형은 드라이브에 대한 개체 상태 값 집합을 정의합니다.
VDS_FILE_SYSTEM_FLAG 파일 시스템의 유효한 플래그 집합을 정의합니다.
VDS_FILE_SYSTEM_FORMAT_SUPPORT_FLAG 볼륨 서식 지정에 지원되는 파일 시스템의 속성을 정의합니다.
VDS_FILE_SYSTEM_PROP_FLAG 파일 시스템 압축의 세부 정보를 정의합니다.
VDS_FILE_SYSTEM_TYPE VDS_FILE_SYSTEM_TYPE(vds.h) 열거형은 파일 시스템에 유효한 형식 집합을 정의합니다.
VDS_FORMAT_OPTION_FLAGS IVdsDiskPartitionMF2::FormatPartitionEx2 메서드에 대한 유효한 서식 옵션 집합을 정의합니다.
VDS_HBAPORT_SPEED_FLAG VDS_HBAPORT_SPEED_FLAG(vds.h) 열거형은 HBA 포트에서 지원하는 속도를 결정하기 위한 유효한 플래그 집합을 정의합니다.
VDS_HBAPORT_STATUS VDS_HBAPORT_STATUS(vds.h) 열거형은 HBA 포트에 대한 유효한 상태 집합을 정의합니다.

VDS_HBAPORT_TYPE VDS_HBAPORT_TYPE(vds.h) 열거형은 HBA 포트에 유효한 형식 집합을 정의합니다.
VDS_HEALTH VDS_HEALTH(vds.h) 열거형은 VDS 개체의 상태 값 집합을 정의합니다.
VDS_HWPROVIDER_TYPE VDS_HWPROVIDER_TYPE(vds.h) 열거형은 하드웨어 공급자에 대한 유효한 형식 집합을 정의합니다.
VDS_INTERCONNECT_FLAG VDS_INTERCONNECT_FLAG(vds.h) 열거형은 하위 시스템이 지원할 수 있는 상호 연결 형식 집합을 정의합니다.
VDS_IPADDRESS_TYPE VDS_IPADDRESS_TYPE(vds.h) 열거형은 IP 주소에 유효한 형식 집합을 정의합니다.
VDS_ISCSI_AUTH_TYPE VDS_ISCSI_AUTH_TYPE(vds.h) 열거형은 iSCSI 대상에 로그인할 때 인증에 유효한 형식 집합을 정의합니다.
VDS_ISCSI_IPSEC_FLAG VDS_ISCSI_IPSEC_FLAG(vds.h) 열거형은 지원되지 않지만 나중에 사용하도록 예약되어 있습니다.
VDS_ISCSI_LOGIN_FLAG VDS_ISCSI_LOGIN_FLAG(vds.h) 열거형은 iSCSI 대상 로그인 옵션을 지정하기 위한 유효한 플래그 집합을 정의합니다.
VDS_ISCSI_LOGIN_TYPE VDS_ISCSI_LOGIN_TYPE(vds.h) 열거형은 iSCSI 대상에 로그인하기 위한 유효한 형식 집합을 정의합니다.
VDS_ISCSI_PORTAL_STATUS VDS_ISCSI_PORTAL_STATUS(vds.h) 열거형은 iSCSI 포털에 유효한 상태 값 집합을 정의합니다.
VDS_LOADBALANCE_POLICY_ENUM VDS_LOADBALANCE_POLICY_ENUM(vds.h) 열거형은 경로에 대한 유효한 부하 분산 정책 집합을 정의합니다.

VDS_LUN_FLAG VDS_LUN_FLAG(vds.h) 열거형은 LUN 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_LUN_PLEX_FLAG VDS_LUN_PLEX_FLAG(vds.h) 열거형은 LUN plex 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_LUN_PLEX_STATUS VDS_LUN_PLEX_STATUS(vds.h) 열거형은 LUN 플렉스에 대한 개체 상태 값 집합을 정의합니다.
VDS_LUN_PLEX_TYPE VDS_LUN_PLEX_TYPE(vds.h) 열거형은 LUN 플렉스에 유효한 형식 집합을 정의합니다.
VDS_LUN_RESERVE_MODE 지원되지 않습니다. 이 열거형은 나중에 사용하도록 예약되어 있습니다. (VDS_LUN_RESERVE_MODE)
VDS_LUN_STATUS VDS_LUN_STATUS(vds.h) 열거형은 LUN에 대한 개체 상태 값 집합을 정의합니다.
VDS_LUN_TYPE VDS_LUN_TYPE(vds.h) 열거형은 LUN에 유효한 형식 집합을 정의합니다.
VDS_MAINTENANCE_OPERATION VDS_MAINTENANCE_OPERATION(vds.h) 열거형은 유효한 하위 시스템 유지 관리 작업 집합을 정의합니다.
VDS_NOTIFICATION_TARGET_TYPE VDS_NOTIFICATION_TARGET_TYPE(vds.h) 열거형은 VDS 알림의 유효한 대상 유형(주체) 집합을 정의합니다.
VDS_OBJECT_TYPE VDS_OBJECT_TYPE(vds.h) 열거형은 VDS 개체의 유효한 형식 집합을 정의합니다.
VDS_PACK_FLAG 팩 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_PACK_STATUS

팩에 대한 개체 상태 값 집합을 정의합니다.
VDS_PARTITION_FLAG 파티션에 대한 유효한 플래그 집합을 정의합니다.
VDS_PARTITION_STYLE 파티션 스타일 값 집합을 정의합니다.
VDS_PATH_STATUS VDS_PATH_STATUS(vds.h) 열거형은 포트에 유효한 상태 값 집합을 정의합니다.
VDS_PORT_STATUS VDS_PORT_STATUS(vds.h) 열거형은 포트에 대한 개체 상태 값 집합을 정의합니다.
VDS_PROVIDER_FLAG VDS_PROVIDER_FLAG(vds.h) 열거형은 공급자 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_PROVIDER_LBSUPPORT_FLAG VDS_PROVIDER_LBSUPPORT_FLAG(vds.h) 열거형은 하드웨어 공급자가 지원하는 부하 분산 정책을 나타내는 유효한 플래그 집합을 지정합니다.
VDS_PROVIDER_TYPE VDS_PROVIDER_TYPE(vds.h) 열거형은 공급자에 유효한 형식 집합을 정의합니다.
VDS_QUERY_PROVIDER_FLAG 공급자 쿼리 작업에 유효한 플래그 집합을 정의합니다. 호출자는 하드웨어 공급자, 소프트웨어 공급자 또는 둘 다를 쿼리할 수 있습니다.
VDS_RAID_TYPE VDS_RAID_TYPE(vds.h) 열거형은 스토리지 풀의 기본 RAID 유형을 지정하는 데 사용할 수 있는 집합 열거형 값을 정의합니다.
VDS_RECOVER_ACTION VDS_RECOVER_ACTION(vds.h) 열거형은 시스템 사용을 위해 예약되어 있습니다.
VDS_SAN_POLICY 유효한 디스크 SAN 정책 플래그 집합을 정의합니다.

VDS_SERVICE_FLAG 서비스 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_STORAGE_POOL_STATUS VDS_STORAGE_POOL_STATUS(vds.h) 열거형은 스토리지 풀에 대한 개체 상태 값 집합을 정의합니다.
VDS_STORAGE_POOL_TYPE VDS_STORAGE_POOL_TYPE(vds.h) 열거형은 스토리지 풀 유형 집합을 정의합니다.
VDS_SUB_SYSTEM_FLAG VDS_SUB_SYSTEM_FLAG(vds.h) 열거형은 하위 시스템 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_SUB_SYSTEM_STATUS VDS_SUB_SYSTEM_STATUS(vds.h) 열거형은 하위 시스템에 대한 개체 상태 값 집합을 정의합니다.
VDS_SUB_SYSTEM_SUPPORTED_RAID_TYPE_FLAG VDS_SUB_SYSTEM_SUPPORTED_RAID_TYPE_FLAG(vds.h) 열거형은 하위 시스템에 의해 지원될 수 있는 RAID 수준 집합을 정의합니다.
VDS_TRANSITION_STATE VDS_TRANSITION_STATE(vds.h) 열거형은 VDS 개체에 대한 유효한 전환 상태 값 집합을 정의합니다.
VDS_VDISK_STATE 가상 디스크 개체의 상태 값 집합을 정의합니다.
VDS_VERSION_SUPPORT_FLAG VDS_VERSION_SUPPORT_FLAG(vds.h) 열거형은 지원되는 VDS 인터페이스 버전을 나타냅니다.
VDS_VOLUME_FLAG 볼륨 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_VOLUME_PLEX_STATUS 볼륨 플렉스에 대한 개체 상태 값 집합을 정의합니다.

VDS_VOLUME_PLEX_TYPE 볼륨 플렉스에 유효한 형식 집합을 정의합니다.
VDS_VOLUME_STATUS 볼륨에 대한 개체 상태 값 집합을 정의합니다.
VDS_VOLUME_TYPE 볼륨 개체의 유효한 형식 집합을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

vdshwprv.h 헤더

아티클2023. 03. 08.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

vdshwprv.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumVdsObject IEnumVdsObject(vdshwprv.h) 인터페이스는 지정된 형식의 VDS 개체 집합을 통해 열거됩니다.
IVdsAdmin 공급자를 VDS에 등록합니다.
IVdsAdviseSink IVdsAdviseSink(vdshwprv.h) 인터페이스는 VDS 알림을 받습니다.
IVdsAsync IVdsAsync(vdshwprv.h) 인터페이스는 비동기 작업을 관리합니다. 비동기 작업을 시작하는 메서드는 IVdsAsync 인터페이스에 대한 포인터를 반환합니다.
IVdsController IVdsController(vdshwprv.h) 인터페이스는 컨트롤러에서 쿼리 및 구성 작업을 수행하기 위한 메서드를 제공합니다.
IVdsControllerControllerPort IVdsControllerControllerPort(vdshwprv.h) 인터페이스는 IVdsController 인터페이스를 구현하는 클래스에 대한 컨트롤러 포트를 열거하는 메서드를 제공합니다.
IVdsControllerPort IVdsControllerPort(vdshwprv.h) 인터페이스는 컨트롤러 포트에서 쿼리 및 구성 작업을 수행하기 위한 메서드를 제공합니다.

IVdsDrive IVdsDrive(vdshwprv.h) 인터페이스는 드라이브에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsDrive2 IVdsDrive2(vdshwprv.h) 인터페이스는 드라이브의 속성을 쿼리하는 메서드를 제공합니다.
IVdsHwProvider IVdsHwProvider(vdshwprv.h) 인터페이스는 하드웨어 공급자에서 쿼리, 다시 열거 및 새로 고침 작업을 수행하는 메서드를 제공합니다.
IVdsHwProviderPrivate VDS가 하드웨어 공급자가 지정된 LUN을 관리하는지 여부를 확인할 수 있도록 하는 메서드를 제공합니다.
IVdsHwProviderPrivateMpio 특정 HBA 포트에서 시작된 경로의 상태를 공급자로 설정하는 메서드를 제공합니다.
IVdsHwProviderStoragePools IVdsHwProviderStoragePools 인터페이스(vdshwprv.h)는 스토리지 풀에서 LUN을 만들고 하드웨어 공급자가 관리하는 스토리지 풀을 열거하는 메서드를 제공합니다.
IVdsHwProviderType IVdsHwProviderType 인터페이스(vdshwprv.h)는 하드웨어 공급자 유형을 검색하는 메서드를 제공합니다.
IVdsHwProviderType2 IVdsHwProviderType2 인터페이스(vdshwprv.h)가 구현되지 않았습니다. 대신 IVdsHwProviderType 인터페이스(vds.h)를 사용합니다.
IVdsIscsiPortal IVdsIscsiPortal 인터페이스(vdshwprv.h)는 iSCSI 포털에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsIscsiPortalGroup IVdsIscsiPortalGroup 인터페이스(vdshwprv.h)는 iSCSI 포털 그룹에서 쿼리 및 구성 서비스를 수행하는 메서드를 제공합니다.

IVdsIscsiTarget IVdsIscsiTarget 인터페이스(vdshwprv.h)는 iSCSI 대상에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsLun IVdsLun 인터페이스(vdshwprv.h)는 LUN(논리 단위 번호)에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsLun2 IVdsLun2 인터페이스(vdshwprv.h)는 LUN(논리 단위 번호) 힌트를 적용하고 쿼리하는 메서드를 제공합니다.
IVdsLunControllerPorts IVdsLunControllerPorts 인터페이스(vdshwprv.h)는 LUN에서 컨트롤러 포트 구성 작업을 수행하는 메서드를 제공합니다.
IVdsLunIscsi IVdsLunIscsi 인터페이스(vdshwprv.h)는 iSCSI LUN에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsLunMpio IVdsLunMpio 인터페이스(vdshwprv.h)는 MPIO 확장을 사용하여 LUN에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsLunNaming IVdsLunNaming 인터페이스(vdshwprv.h)는 IVdsLun 인터페이스를 구현하는 클래스의 LUN 이름을 지정하는 메서드를 제공합니다.
IVdsLunNumber IVdsLunNumber 인터페이스(vdshwprv.h)는 LUN에 대한 LUN 번호를 쿼리하는 메서드를 제공합니다.
IVdsLunPlex IVdsLunPlex 인터페이스(vdshwprv.h)는 LUN 플렉스에서 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsMaintenance IVdsMaintenance 인터페이스(vdshwprv.h)는 하위 시스템, 컨트롤러, LUN 또는 드라이브에서 유지 관리 작업을 수행하는 메서드를 제공합니다.

IVdsProvider IVdsProvider 인터페이스(vdshwprv.h)는 하드웨어 또는 소프트웨어 공급자의 속성을 반환합니다.
IVdsProviderPrivate VDS가 공급자 개체에 대해 기타 작업을 수행할 수 있도록 하는 메서드를 제공합니다.
IVdsProviderSupport IVdsProviderSupport 인터페이스(vdshwprv.h)는 공급자가 지원하는 VDS 인터페이스의 버전을 나타내는 메서드를 제공합니다.
IVdsStoragePool IVdsStoragePool 인터페이스(vdshwprv.h)는 정보를 쿼리하고 스토리지 풀에 대한 관련 개체를 열거하는 메서드를 제공합니다.
IVdsSubSystem IVdsSubSystem 인터페이스(vdshwprv.h)는 하위 시스템에 대해 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsSubSystem2 IVdsSubSystem2 인터페이스는 VDS_HINTS2 및 VDS_SUB_SYSTEM_PROP2 구조를 사용하여 하위 시스템에 대한 쿼리 및 구성 작업을 수행하는 메서드를 제공합니다.
IVdsSubSystemInterconnect IVdsSubSystemInterconnect 인터페이스(vdshwprv.h)는 하위 시스템에 의해 지원되는 상호 연결 형식을 쿼리하는 메서드를 제공합니다.
IVdsSubSystemIscsi IVdsSubSystemIscsi 인터페이스(vdshwprv.h)는 하위 시스템의 iSCSI 대상 및 포털을 쿼리하고 구성하는 메서드를 제공합니다.
IVdsSubSystemNaming IVdsSubSystemNaming 인터페이스(vdshwprv.h)는 IVdsSubSystem 인터페이스를 구현하는 클래스에 대한 하위 시스템의 이름을 지정하는 메서드를 제공합니다.

구조체

VDS_ASYNC_OUTPUT VDS_ASYNC_OUTPUT 구조체(vdshwprv.h)는 비동기 개체의 출력을 정의합니다. 출력 요소는 작업 유형에 따라 달라집니다.
VDS_CONTROLLER_NOTIFICATION VDS_CONTROLLER_NOTIFICATION 구조체(vdshwprv.h)는 컨트롤러 이벤트의 세부 정보를 정의합니다.
VDS_CONTROLLER_PROP VDS_CONTROLLER_PROP 구조체(vdshwprv.h)는 컨트롤러 개체의 속성을 정의합니다.
VDS_DISK_NOTIFICATION VDS_DISK_NOTIFICATION 구조체(vdshwprv.h)는 디스크 이벤트의 세부 정보를 정의합니다.
VDS_DRIVE_EXTENT VDS_DRIVE_EXTENT 구조체(vdshwprv.h)는 드라이브 익스텐트의 속성을 정의합니다.
VDS_DRIVE_LETTER_NOTIFICATION VDS_DRIVE_LETTER_NOTIFICATION 구조체(vdshwprv.h)는 드라이브 문자 이벤트의 세부 정보를 정의합니다.
VDS_DRIVE_NOTIFICATION VDS_DRIVE_NOTIFICATION 구조체(vdshwprv.h)는 드라이브 이벤트의 세부 정보를 정의합니다.
VDS_DRIVE_PROP VDS_DRIVE_PROP 구조체(vdshwprv.h)는 드라이브 개체의 속성을 정의합니다.
VDS_DRIVE_PROP2 VDS_DRIVE_PROP2 구조체(vdshwprv.h)는 드라이브 개체의 속성을 정의합니다.
VDS_FILE_SYSTEM_NOTIFICATION VDS_FILE_SYSTEM_NOTIFICATION 구조체(vdshwprv.h)는 파일 시스템 이벤트의 세부 정보를 정의합니다.
VDS_HBAPORT_PROP VDS_HBAPORT_PROP 구조체(vdshwprv.h)는 HBA 포트의 속성을 정의합니다.

VDS_HINTS VDS_HINTS 구조체(vdshwprv.h)는 LUN 또는 LUN 플렉스에 대한 자동 매핑 힌트를 정의합니다.
VDS_HINTS2 VDS_HINTS2 구조체(vdshwprv.h)에는 스토리지 풀의 LUN에 대한 자동 매핑 힌트가 포함되어 있습니다.
VDS_IPADDRESS VDS_IPADDRESS 구조체(vdshwprv.h)는 IP 주소와 포트를 정의합니다.
VDS_ISCSI_INITIATOR_ADAPTER_PROP VDS_ISCSI_INITIATOR_ADAPTER_PROP 구조체(vdshwprv.h)는 iSCSI 초기자 어댑터의 속성을 정의합니다.
VDS_ISCSI_INITIATOR_PORTAL_PROP VDS_ISCSI_INITIATOR_PORTAL_PROP 구조체(vdshwprv.h)는 iSCSI 초기자 포털의 속성을 정의합니다.
VDS_ISCSI_IPSEC_KEY VDS_ISCSI_IPSEC_KEY 구조체(vdshwprv.h)는 지원되지 않으며 나중에 사용하도록 예약되어 있습니다.
VDS_ISCSI_PORTAL_PROP VDS_ISCSI_PORTAL_PROP 구조체(vdshwprv.h)는 iSCSI 포털의 속성을 정의합니다.
VDS_ISCSI_PORTALGROUP_PROP VDS_ISCSI_PORTALGROUP_PROP 구조체(vdshwprv.h)는 iSCSI 포털 그룹의 속성을 정의합니다.
VDS_ISCSI_SHARED_SECRET VDS_ISCSI_SHARED_SECRET 구조체(vdshwprv.h)는 CHAP 공유 비밀을 정의합니다.
VDS_ISCSI_TARGET_PROP VDS_ISCSI_TARGET_PROP 구조체(vdshwprv.h)는 iSCSI 대상의 속성을 정의합니다.
VDS_LUN_NOTIFICATION VDS_LUN_NOTIFICATION 구조체(vdshwprv.h)는 LUN 알림의 세부 정보를 정의합니다.

VDS_LUN_PLEX_PROP VDS_LUN_PLEX_PROP 구조체(vdshwprv.h)는 LUN 플렉스 개체의 속성을 정의합니다.
VDS_LUN_PROP VDS_LUN_PROP 구조체(vdshwprv.h)는 LUN 개체의 속성을 정의합니다.
VDS_MOUNT_POINT_NOTIFICATION VDS_MOUNT_POINT_NOTIFICATION 구조체(vdshwprv.h)는 드라이브 문자 또는 볼륨 GUID 경로가 변경되어 소프트웨어 공급자가 반환한 데이터를 나타냅니다.
VDS_NOTIFICATION VDS_NOTIFICATION 구조(vdshwprv.h)는 각 알림 대상 유형(제목)과 관련된 VDS 알림 구조를 정의합니다.
VDS_PACK_NOTIFICATION VDS_PACK_NOTIFICATION 구조체(vdshwprv.h)는 팩 이벤트의 세부 정보를 정의합니다.
VDS_PARTITION_NOTIFICATION VDS_PARTITION_NOTIFICATION 구조체(vdshwprv.h)는 파티션 이벤트의 세부 정보를 정의합니다.
VDS_PATH_ID VDS_PATH_ID 구조체(vdshwprv.h)는 경로에 대한 고유 ID를 정의합니다.
VDS_PATH_INFO VDS_PATH_INFO 구조체(vdshwprv.h)는 LUN 경로에 대한 정보를 정의하고 IVdsLunMpio::GetPathInfo 메서드의 ppPaths 매개 변수에 반환됩니다.
VDS_PATH_POLICY VDS_PATH_POLICY 구조체(vdshwprv.h)는 특정 경로에 적용되는 부하 분산 정책을 정의합니다.
VDS_POOL_ATTRIBUTES VDS_POOL_ATTRIBUTES 구조체(vdshwprv.h)는 스토리지 풀의 특성을 정의합니다.
VDS_POOL_CUSTOM_ATTRIBUTES VDS_POOL_CUSTOM_ATTRIBUTES 구조체(vdshwprv.h)는 스토리지 풀의 사용자 지정 특성을 정의합니다.

VDS_PORT_NOTIFICATION VDS_PORT_NOTIFICATION 구조체(vdshwprv.h)는 컨트롤러 포트 이벤트의 세부 정보를 정의합니다.
VDS_PORT_PROP VDS_PORT_PROP 구조체(vdshwprv.h)는 컨트롤러 개체의 포트 속성을 정의합니다.
VDS_PORTAL_GROUP_NOTIFICATION VDS_PORTAL_GROUP_NOTIFICATION 구조체(vdshwprv.h)는 iSCSI 포털 이벤트의 세부 정보를 정의합니다.
VDS_PORTAL_NOTIFICATION VDS_PORTAL_NOTIFICATION 구조체(vdshwprv.h)는 iSCSI 포털 이벤트의 세부 정보를 정의합니다.
VDS_PROVIDER_PROP VDS_PROVIDER_PROP 구조체(vdshwprv.h)는 공급자 개체의 속성을 정의합니다.
VDS_STORAGE_POOL_DRIVE_EXTENT VDS_STORAGE_POOL_DRIVE_EXTENT 구조체(vdshwprv.h)는 스토리지 풀에서 사용할 수 있는 드라이브 범위를 정의합니다.
VDS_STORAGE_POOL_PROP VDS_STORAGE_POOL_PROP 구조체(vdshwprv.h)는 스토리지 풀 개체의 속성을 정의합니다.
VDS_SUB_SYSTEM_NOTIFICATION VDS_SUB_SYSTEM_NOTIFICATION 구조체(vdshwprv.h)는 하위 시스템 이벤트의 세부 정보를 정의합니다.
VDS_SUB_SYSTEM_PROP VDS_SUB_SYSTEM_PROP 구조체(vdshwprv.h)는 하위 시스템 개체의 속성을 정의합니다.
VDS_SUB_SYSTEM_PROP2 VDS_SUB_SYSTEM_PROP2 구조체(vdshwprv.h)는 하위 시스템 개체의 속성을 정의합니다.
VDS_TARGET_NOTIFICATION VDS_TARGET_NOTIFICATION 구조체(vdshwprv.h)는 iSCSI 대상 이벤트의 세부 정보를 정의합니다.

VDS_VOLUME_NOTIFICATION VDS_VOLUME_NOTIFICATION 구조체(vdshwprv.h)는 볼륨 이벤트의 세부 정보를 정의합니다.
VDS_WWN VDS_WWN 구조체(vdshwprv.h)는 WWN(전 세계 이름)을 정의합니다. 이 구조체는 ANSI HBA API에서 정의한 HBA_WWN 구조체에 해당합니다.

열거형

VDS_ASYNC_OUTPUT_TYPE VDS_ASYNC_OUTPUT_TYPE 열거형(vdshwprv.h)은 개체가 처리할 수 있는 작업 집합을 정의합니다.
VDS_CONTROLLER_STATUS VDS_CONTROLLER_STATUS 열거형(vdshwprv.h)은 컨트롤러의 개체 상태 값 집합을 정의합니다.
VDS_DRIVE_FLAG VDS_DRIVE_FLAG 열거형(vdshwprv.h)은 드라이브 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_DRIVE_STATUS VDS_DRIVE_STATUS 열거형(vdshwprv.h)은 드라이브의 개체 상태 값 집합을 정의합니다.
VDS_FILE_SYSTEM_TYPE VDS_FILE_SYSTEM_TYPE 열거형(vdshwprv.h)은 파일 시스템에 유효한 형식 집합을 정의합니다.
VDS_HBAPORT_SPEED_FLAG VDS_HBAPORT_SPEED_FLAG 열거형(vdshwprv.h)은 HBA 포트에서 지원하는 속도를 결정하기 위한 유효한 플래그 집합을 정의합니다.
VDS_HBAPORT_STATUS VDS_HBAPORT_STATUS 열거형(vdshwprv.h)은 HBA 포트에 대한 유효한 상태 집합을 정의합니다.
VDS_HBAPORT_TYPE VDS_HBAPORT_TYPE 열거형(vdshwprv.h)은 HBA 포트에 유효한 형식 집합을 정의합니다.

VDS_HEALTH VDS_HEALTH 열거형(vdshwprv.h)은 VDS 개체의 상태 값 집합을 정의합니다.
VDS_HWPROVIDER_TYPE VDS_HWPROVIDER_TYPE 열거형(vdshwprv.h)은 하드웨어 공급자에 대한 유효한 형식 집합을 정의합니다.
VDS_INTERCONNECT_FLAG VDS_INTERCONNECT_FLAG 열거형(vdshwprv.h)은 하위 시스템이 지원할 수 있는 상호 연결 형식 집합을 정의합니다.
VDS_IPADDRESS_TYPE VDS_IPADDRESS_TYPE 열거형(vdshwprv.h)은 IP 주소에 유효한 형식 집합을 정의합니다.
VDS_ISCSI_AUTH_TYPE VDS_ISCSI_AUTH_TYPE 열거형(vdshwprv.h)은 iSCSI 대상에 로그인할 때 인증에 유효한 형식 집합을 정의합니다.
VDS_ISCSI_IPSEC_FLAG VDS_ISCSI_IPSEC_FLAG 열거형(vdshwprv.h)은 지원되지 않습니다. 이 열거형은 나중에 사용하도록 예약되어 있습니다.
VDS_ISCSI_LOGIN_FLAG VDS_ISCSI_LOGIN_FLAG 열거형(vdshwprv.h)은 iSCSI 대상 로그인 옵션을 지정하기 위한 유효한 플래그 집합을 정의합니다.
VDS_ISCSI_LOGIN_TYPE VDS_ISCSI_LOGIN_TYPE 열거형(vdshwprv.h)은 iSCSI 대상에 로그인하기 위한 유효한 형식 집합을 정의합니다.
VDS_ISCSI_PORTAL_STATUS VDS_ISCSI_PORTAL_STATUS 열거형(vdshwprv.h)은 iSCSI 포털의 유효한 상태 값 집합을 정의합니다.
VDS_LOADBALANCE_POLICY_ENUM VDS_LOADBALANCE_POLICY_ENUM 열거형(vdshwprv.h)은 경로에 대한 유효한 부하 분산 정책 집합을 정의합니다.

VDS_LUN_FLAG VDS_LUN_FLAG 열거형(vdshwprv.h)은 LUN 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_LUN_PLEX_FLAG VDS_LUN_PLEX_FLAG 열거형(vdshwprv.h)은 LUN 플렉스 개체에 유효한 플래그 집합을 정의합니다.
VDS_LUN_PLEX_STATUS VDS_LUN_PLEX_STATUS 열거형(vdshwprv.h)은 LUN 플렉스에 대한 개체 상태 값 집합을 정의합니다.
VDS_LUN_PLEX_TYPE VDS_LUN_PLEX_TYPE 열거형(vdshwprv.h)은 LUN 플렉스에 유효한 형식 집합을 정의합니다.
VDS_LUN_STATUS VDS_LUN_STATUS 열거형(vdshwprv.h)은 LUN에 대한 개체 상태 값 집합을 정의합니다.
VDS_LUN_TYPE VDS_LUN_TYPE 열거형(vdshwprv.h)은 LUN에 유효한 형식 집합을 정의합니다.
VDS_MAINTENANCE_OPERATION VDS_MAINTENANCE_OPERATION 열거형(vdshwprv.h)은 유효한 하위 시스템 유지 관리 작업 집합을 정의합니다.
VDS_NOTIFICATION_TARGET_TYPE VDS_NOTIFICATION_TARGET_TYPE 열거형(vdshwprv.h)은 VDS 알림의 유효한 대상 유형 집합을 정의합니다.
VDS_OBJECT_TYPE VDS_OBJECT_TYPE 열거형(vdshwprv.h)은 VDS 개체의 유효한 형식 집합을 정의합니다.
VDS_PATH_STATUS VDS_PATH_STATUS 열거형(vdshwprv.h)은 포트에 대한 유효한 상태 값 집합을 정의합니다.
VDS_PORT_STATUS VDS_PORT_STATUS 열거형(vdshwprv.h)은 포트에 대한 개체 상태 값 집합을 정의합니다.

VDS_PROVIDER_FLAG VDS_PROVIDER_FLAG 열거형(vdshwprv.h)은 공급자 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_PROVIDER_LBSUPPORT_FLAG VDS_PROVIDER_LBSUPPORT_FLAG 열거형(vdshwprv.h)은 하드웨어 공급자가 지원하는 부하 분산 정책을 나타내는 유효한 플래그 집합을 지정합니다.
VDS_PROVIDER_TYPE VDS_PROVIDER_TYPE 열거형(vdshwprv.h)은 공급자에 대한 유효한 형식 집합을 정의합니다.
VDS_RAID_TYPE VDS_RAID_TYPE 열거형(vdshwprv.h)은 스토리지 풀의 기본 RAID 유형을 지정하는 데 사용할 수 있는 집합 열거형 값을 정의합니다.
VDS_RECOVER_ACTION VDS_RECOVER_ACTION 열거형(vdshwprv.h)은 시스템 사용을 위해 예약되어 있습니다.
VDS_STORAGE_POOL_STATUS VDS_STORAGE_POOL_STATUS 열거형(vdshwprv.h)은 스토리지 풀에 대한 개체 상태 값 집합을 정의합니다.
VDS_STORAGE_POOL_TYPE VDS_STORAGE_POOL_TYPE 열거형(vdshwprv.h)은 스토리지 풀 유형의 집합을 정의합니다.
VDS_SUB_SYSTEM_FLAG VDS_SUB_SYSTEM_FLAG 열거형(vdshwprv.h)은 하위 시스템 개체에 대한 유효한 플래그 집합을 정의합니다.
VDS_SUB_SYSTEM_STATUS VDS_SUB_SYSTEM_STATUS 열거형(vdshwprv.h)은 하위 시스템의 개체 상태 값 집합을 정의합니다.
VDS_SUB_SYSTEM_SUPPORTED_RAID_TYPE_FLAG VDS_SUB_SYSTEM_SUPPORTED_RAID_TYPE_FLAG 열거형(vdshwprv.h)은 하위 시스템에 의해 지원될 수 있는 RAID 수준 집합을 정의합니다.

VDS_TRANSITION_STATE VDS_TRANSITION_STATE 열거형(vdshwprv.h)은 VDS 개체에 대한 유효한 전환 상태 값 집합을 정의합니다.
VDS_VERSION_SUPPORT_FLAG VDS_VERSION_SUPPORT_FLAG 열거형(vdshwprv.h)은 지원되는 VDS 인터페이스 버전을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

vdslun.h 헤더

아티클2023. 08. 27.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

vdslun.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

VDS_INTERCONNECT 물리적 상호 연결의 주소 데이터를 정의합니다.
VDS_LUN_INFORMATION LUN 또는 디스크에 대한 정보를 정의합니다. 애플리케이션은 이 구조를 사용하여 항상 LUN을 고유하게 식별할 수 있습니다.
VDS_STORAGE_DEVICE_ID_DESCRIPTOR 스토리지 디바이스에 대해 하나 이상의 스토리지 식별자를 정의합니다(일반적으로 디바이스의 클래스가 아닌 instance).
VDS_STORAGE_IDENTIFIER 특정 코드 집합 및 형식을 사용하여 스토리지 디바이스를 정의합니다.

열거형

VDS_INTERCONNECT_ADDRESS_TYPE 실제 상호 연결의 유효한 주소 형식 집합을 정의합니다.
VDS_STORAGE_BUS_TYPE 스토리지 디바이스의 유효한 버스 유형 집합을 정의합니다.
VDS_STORAGE_IDENTIFIER_CODE_SET

스토리지 식별자의 유효한 코드 집합(인코딩)의 집합을 정의합니다.
VDS_STORAGE_IDENTIFIER_TYPE 스토리지 식별자에 대한 유효한 형식 집합을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

verrsrc.h 헤더

아티클2023. 08. 27.

이 헤더는 메뉴 및 기타 리소스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 메뉴 및 기타 리소스

verrsrc.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

VS_FIXEDFILEINFO 파일에 대한 버전 정보를 포함합니다. 이 정보는 언어 및 코드 페이지 독립적입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

versionhelpers.h 헤더

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- System Services

versionhelpers.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

기능

테이블 확장

IsActiveSessionCountLimited
IsWindows10OrGreater 현재 OS 버전이 Windows 10 버전과 일치하는지 또는 그보다 큰지를 나타냅니다.
IsWindows7OrGreater 현재 OS 버전이 Windows 7 버전과 일치하는지 또는 그보다 큰지를 나타냅니다.
IsWindows7SP1OrGreater 현재 OS 버전이 WINDOWS 7 SP1(서비스 팩 1) 버전과 일치하는지 또는 그보다 큰지를 나타냅니다.
IsWindows8OrGreater 현재 OS 버전이 Windows 8 버전과 일치하거나 그보다 큰지 여부를 나타냅니다.
IsWindows8Point1OrGreater 현재 OS 버전이 Windows 8.1 버전과 일치하거나 그보다 큰지 여부를 나타냅니다.
IsWindowsServer 현재 OS가 Windows Server 릴리스인지를 나타냅니다.
IsWindowsVersionOrGreater 현재 OS 버전이 제공된 버전 정보와 일치하거나 그보다 큰지 여부를 나타냅니다.
IsWindowsVistaOrGreater 현재 OS 버전이 Windows Vista 버전과 일치하는지 또는 그보다 큰지를 나타냅니다.

IsWindowsVistaSP1OrGreater 현재 OS 버전이 WINDOWS Vista SP1(서비스 팩 1) 버전과 일치하는지 또는 그보다 큰지를 나타냅니다.
IsWindowsVistaSP2OrGreater 현재 OS 버전이 WINDOWS Vista SP2(서비스 팩 2) 버전과 일치하는지 또는 그보다 큰지를 나타냅니다.
IsWindowsXPOrGreater 현재 OS 버전이 Windows XP 버전과 일치하는지 또는 그보다 큰지를 나타냅니다.
IsWindowsXPSP1OrGreater 현재 OS 버전이 WINDOWS XP SP1(서비스 팩 1) 버전과 일치하는지 또는 그보다 큰지를 나타냅니다.
IsWindowsXPSP2OrGreater 현재 OS 버전이 WINDOWS XP SP2(서비스 팩 2) 버전과 일치하는지 또는 그보다 큰지를 나타냅니다.
IsWindowsXPSP3OrGreater 현재 OS 버전이 WINDOWS XP SP3(서비스 팩 3) 버전과 일치하는지 또는 그보다 큰지를 나타냅니다.

Last updated on 2025. 05. 09.

vfw.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows 멀티미디어에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 멀티미디어

vfw.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAVIEditStream IAVIEditStream 인터페이스는 편집 가능한 스트림 조작 및 수정을 지원합니다. 다음 사용자 지정 메서드 외에도 IUnknown::QueryInterface, IUnknown::AddRef, IUnknown::Release를 사용합니다.
IAVIFile IAVIFile 인터페이스는 파일 및 파일 헤더 열기 및 조작, 스트림 인터페이스 만들기 및 가져오기를 지원합니다. 다음 사용자 지정 메서드 외에도 IUnknown::QueryInterface, IUnknown::AddRef 및 IUnknown::Release를 사용합니다.
IAVIStream IAVIStream 인터페이스는 파일 내에서 데이터 스트림을 만들고 조작할 수 있습니다. 다음 사용자 지정 메서드 외에도 IUnknown::QueryInterface, IUnknown::AddRef, IUnknown::Release를 사용합니다.
IAVIStreaming IAVIStreaming 인터페이스는 스트리밍 작업에서 재생을 위해 열린 데이터 스트림 준비를 지원합니다. 다음 사용자 지정 메서드 외에도 IUnknown::QueryInterface, IUnknown::AddRef, IUnknown::Release를 사용합니다.
IGetFrame IGetFrame 인터페이스는 열린 스트림에서 개별 프레임의 추출, 압축 해제 및 표시를 지원합니다.

Functions

AVIBuildFilterA AVIBuildFilter 함수는 GetOpenFileName 또는 GetSaveFileName 함수에서 이후에 사용되는 필터 사양을 빌드합니다. (ANSI)
AVIBuildFilterW AVIBuildFilter 함수는 GetOpenFileName 또는 GetSaveFileName 함수에서 이후에 사용되는 필터 사양을 빌드합니다. (유니코드)
AVIClearClipboard AVIClearClipboard 함수는 클립보드에서 AVI 파일을 제거합니다.
AVIFileAddRef AVIFileAddRef 함수는 AVI 파일의 참조 수를 증가합니다.
AVIFileCreateStream AVIFileCreateStream 함수(vfw.h)는 기존 파일에 새 스트림을 만들고 새 스트림에 대한 인터페이스를 만듭니다.
AVIFileCreateStreamA AVIFileCreateStream 함수는 기존 파일에 새 스트림을 만들고 새 스트림에 대한 인터페이스를 만듭니다. (AVIFileCreateStreamA)
AVIFileCreateStreamW AVIFileCreateStreamW(유니코드) 함수(vfw.h)는 기존 파일에 새 스트림을 만들고 새 스트림에 대한 인터페이스를 만듭니다.
AVIFileEndRecord AVIFileEndRecord 함수는 비디오의 1:1 인터리브 요소를 오디오 데이터에 사용하는 인터리브 파일을 작성할 때 레코드의 끝을 표시합니다. (비디오의 각 프레임은 동일한 양의 오디오 데이터와 산재합니다.)
AVIFileExit AVIFileExit 함수는 AVIFile 라이브러리를 종료하고 라이브러리에 대한 참조 수를 감소합니다.
AVIFileGetStream AVIFileGetStream 함수는 지정된 AVI 파일과 연결된 스트림 인터페이스의 주소를 반환합니다.
AVIFileInfo

AVIFileInfo 함수(vfw.h)는 AVI 파일에 대한 정보를 가져옵니다.
AVIFileInfoA AVIFileInfo 함수는 AVI 파일에 대한 정보를 가져옵니다. (AVIFileInfoA)
AVIFileInfoW AVIFileInfoW 함수(vfw.h)는 AVI 파일에 대한 정보를 가져옵니다.
AVIFileInit AVIFileInit 함수는 AVIFile 라이브러리를 초기화합니다.
AVIFileOpen AVIFileOpen 함수(vfw.h)는 AVI 파일을 열고 액세스하는 데 사용되는 파일 인터페이스의 주소를 반환합니다.
AVIFileOpenA AVIFileOpen 함수는 AVI 파일을 열고 액세스하는 데 사용되는 파일 인터페이스의 주소를 반환합니다. (AVIFileOpenA)
AVIFileOpenW AVIFileOpenW 함수(vfw.h)는 AVI 파일을 열고 액세스하는 데 사용되는 파일 인터페이스의 주소를 반환합니다.
AVIFileReadData AVIFileReadData 함수는 작성자 또는 저작권 정보와 같은 전체 파일에 적용되는 선택적 헤더 데이터를 읽습니다.
AVIFileRelease AVIFileRelease 함수는 AVI 파일 인터페이스 핸들의 참조 수를 감소시키고 개수가 0에 도달하면 파일을 닫습니다.
AVIFileWriteData AVIFileWriteData 함수는 보조 데이터(일반 헤더, 형식 및 스트림 데이터 제외)를 파일에 씁니다.
AVIGetFromClipboard AVIGetFromClipboard 함수는 클립보드에서 AVI 파일을 복사합니다.
AVIMakeCompressedStream AVIMakeCompressedStream 함수는 압축되지 않은 스트림 및 압축 필터에서 압축 스트림을 만들고

압축된 스트림에 대한 포인터의 주소를 반환합니다. 이 함수는 오디오 및 비디오 압축을 지원합니다.
AVIMakeFileFromStreams AVIMakeFileFromStreams 함수는 별도의 스트림에서 AVIFile 인터페이스 포인터를 생성합니다.
AVIMakeStreamFromClipboard AVIMakeStreamFromClipboard 함수는 클립보드의 스트림 데이터에서 편집 가능한 스트림을 만듭니다.
AVIPutFileOnClipboard AVIPutFileOnClipboard 함수는 AVI 파일을 클립보드에 복사합니다.
AVISaveA AVISave 함수는 다른 파일 또는 메모리의 데이터 스트림을 결합하여 파일을 빌드합니다. (ANSI)
AVISaveOptions AVISaveOptions 함수는 파일에 대한 저장 옵션을 검색하여 버퍼에 반환합니다.
AVISaveOptionsFree AVISaveOptionsFree 함수는 AVISaveOptions 함수에서 할당한 리소스를 해제합니다.
AVISaveVA AVISaveV 함수는 다른 파일 또는 메모리의 데이터 스트림을 결합하여 파일을 빌드합니다. (ANSI)
AVISaveVW AVISaveV 함수는 다른 파일 또는 메모리의 데이터 스트림을 결합하여 파일을 빌드합니다. (유니코드)
AVISaveW AVISave 함수는 다른 파일 또는 메모리의 데이터 스트림을 결합하여 파일을 빌드합니다. (유니코드)
AVIStreamAddRef AVIStreamAddRef 함수는 AVI 스트림의 참조 수를 증가합니다.
AVIStreamBeginStreaming

AVIStreamBeginStreaming 함수는 스트리밍에 사용되는 매개 변수를 지정하고 스트림 처리기가 스트리밍을 준비할 수 있도록 합니다.
AVIStreamCreate AVIStreamCreate 함수는 파일과 연결되지 않은 스트림을 만듭니다.
AVIStreamDataSize AVIStreamDataSize 매크로는 지정된 스트림에 대한 선택적 헤더 데이터를 검색하는 데 필요한 버퍼 크기(바이트)를 결정합니다.
AVIStreamEnd AVIStreamEnd 매크로는 스트림의 끝과 연결된 샘플을 계산합니다.
AVIStreamEndStreaming AVIStreamEndStreaming 함수는 스트리밍을 종료합니다.
AVIStreamEndTime AVIStreamEndTime 매크로는 스트림의 끝을 나타내는 시간을 반환합니다.
AVIStreamFindSample AVIStreamFindSample 함수는 지정된 위치를 기준으로 샘플(키 프레임, 무정형 프레임 또는 형식 변경이 포함된 프레임)의 위치를 반환합니다.
AVIStreamFormatSize AVIStreamFormatSize 매크로는 샘플에 대한 형식 정보를 스트림에 저장하는 데 필요한 버퍼 크기(바이트)를 결정합니다.
AVIStreamGetFrame AVIStreamGetFrame 함수는 압축 해제된 비디오 프레임의 주소를 반환합니다.
AVIStreamGetFrameClose AVIStreamGetFrameClose 함수는 비디오 프레임의 압축을 해제하는 데 사용되는 리소스를 해제합니다.
AVIStreamGetFrameOpen AVIStreamGetFrameOpen 함수는 지정된 비디오 스트림에서 비디오 프레임의 압축을 풀도록 준비합니다.
AVIStreamInfoA

AVIStreamInfo 함수는 스트림 헤더 정보를 가져옵니다. (ANSI)
AVIStreamInfoW AVIStreamInfo 함수는 스트림 헤더 정보를 가져옵니다. (유니코드)
AVIStreamIsKeyFrame AVIStreamIsKeyFrame 매크로는 지정된 스트림의 샘플이 키 프레임인지 여부를 나타냅니다.
AVIStreamLength AVIStreamLength 함수는 스트림의 길이를 반환합니다.
AVIStreamLengthTime AVIStreamLengthTime 매크로는 스트림의 길이를 시간 단위로 반환합니다.
AVIStreamNearestKeyFrame AVIStreamNearestKeyFrame 매크로는 스트림에서 지정된 위치 또는 앞에 있는 키 프레임을 찾습니다.
AVIStreamNearestKeyFrameTime AVIStreamNearestKeyFrameTime 매크로는 스트림에서 지정된 시간에 가장 가까운(또는 이전) 키 프레임의 시작 부분에 해당하는 시간을 결정합니다.
AVIStreamNearestSample AVIStreamNearestSample 매크로는 스트림에서 지정된 위치 또는 그 이전의 가장 가까운 무중단 샘플을 찾습니다.
AVIStreamNearestSampleTime AVIStreamNearestSampleTime 매크로는 스트림에서 지정된 시간에 가장 가까운 샘플의 시작 부분에 해당하는 시간을 결정합니다.
AVIStreamNextKeyFrame AVIStreamNextKeyFrame 매크로는 스트림에서 지정된 위치에 따라 다음 키 프레임을 찾습니다.
AVIStreamNextKeyFrameTime AVIStreamNextKeyFrameTime 매크로는 지정된 시간부터 스트림에서 다음 키 프레임의 시간을 반환합니다.
AVIStreamNextSample

AVIStreamNextSample 매크로는 스트림의 지정된 위치에서 다음 흠 없는 샘플을 찾습니다.
AVIStreamNextSampleTime AVIStreamNextSampleTime 매크로는 샘플이 스트림의 다음 샘플로 변경되는 시간을 반환합니다. 이 매크로는 스트림에서 다음 흥미로운 시간을 찾습니다.
AVIStreamOpenFromFileA AVIStreamOpenFromFile 함수는 파일에서 단일 스트림을 엽니다. (ANSI)
AVIStreamOpenFromFileW AVIStreamOpenFromFile 함수는 파일에서 단일 스트림을 엽니다. (유니코드)
AVIStreamPrevKeyFrame AVIStreamPrevKeyFrame 매크로는 스트림에서 지정된 위치 앞에 오는 키 프레임을 찾습니다.
AVIStreamPrevKeyFrameTime AVIStreamPrevKeyFrameTime 매크로는 지정된 시간부터 스트림에서 이전 키 프레임의 시간을 반환합니다.
AVIStreamPrevSample AVIStreamPrevSample 매크로는 스트림에서 지정된 위치 앞에 오는 가장 가까운 비어 있지 않은 샘플을 찾습니다.
AVIStreamPrevSampleTime AVIStreamPrevSampleTime 매크로는 스트림에서 지정된 시간보다 앞에 오는 가장 가까운 비어 있지 않은 샘플의 시간을 결정합니다.
AVIStreamRead AVIStreamRead 함수는 스트림 유형에 따라 스트림에서 오디오, 비디오 또는 기타 데이터를 읽습니다.
AVIStreamReadData AVIStreamReadData 함수는 스트림에서 선택적 헤더 데이터를 읽습니다.
AVIStreamReadFormat AVIStreamReadFormat 함수는 스트림 형식 데이터를 읽습니다.
AVIStreamRelease

AVIStreamRelease 함수는 AVI 스트림 인터페이스 핸들의 참조 수를 감소시키고 개수가 0에 도달하면 스트림을 닫습니다.
AVIStreamSampleSize AVIStreamRelease 매크로는 스트림에서 하나의 정보 샘플을 저장하는 데 필요한 버퍼의 크기를 결정합니다. 크기는 lPos로 지정된 위치의 샘플에 해당합니다.
AVIStreamSampleToSample AVIStreamSampleToSample 매크로는 두 번째 스트림에서 발생하는 샘플과 동시에 발생하는 스트림의 샘플을 반환합니다.
AVIStreamSampleToTime AVIStreamSampleToTime 함수는 스트림 위치를 샘플에서 밀리초로 변환합니다.
AVIStreamSetFormat AVIStreamSetFormat 함수는 지정된 위치에 있는 스트림의 형식을 설정합니다.
AVIStreamStart AVIStreamStart 함수는 스트림의 시작 샘플 번호를 반환합니다.
AVIStreamStartTime AVIStreamStartTime 매크로는 스트림의 첫 번째 샘플의 시작 시간을 반환합니다.
AVIStreamTimeToSample AVIStreamTimeToSample 함수는 밀리초에서 샘플로 변환합니다.
AVIStreamWrite AVIStreamWrite 함수는 스트림에 데이터를 씁니다.
AVIStreamWriteData AVIStreamWriteData 함수는 선택적 헤더 정보를 스트림에 씁니다.
capCaptureAbort capCaptureAbort 매크로는 캡처 작업을 중지합니다. 이 매크로를 사용하거나 WM_CAP_ABORT 메시지를 명시적으로 보낼 수 있습니다.
capCaptureGetSetup

capCaptureGetSetup 매크로는 스트리밍 캡처 매개 변수의 현재 설정을 검색합니다. 이 매크로를 사용하거나 WM_CAP_GET_SEQUENCE_SETUP 메시지를 명시적으로 보낼 수 있습니다.
capCaptureSequence capCaptureSequence 매크로는 비디오 및 오디오 캡처를 파일로 스트리밍하기 시작합니다. 이 매크로를 사용하거나 WM_CAP_SEQUENCE 메시지를 명시적으로 보낼 수 있습니다.
capCaptureSequenceNoFile capCaptureSequenceNoFile 매크로는 파일에 데이터를 쓰지 않고 스트리밍 비디오 캡처를 시작합니다. 이 매크로를 사용하거나 WM_CAP_SEQUENCE_NOFILE 메시지를 명시적으로 보낼 수 있습니다.
capCaptureSetSetup capCaptureSetSetup 매크로는 스트리밍 캡처에 사용되는 구성 매개 변수를 설정합니다. 이 매크로를 사용하거나 WM_CAP_SET_SEQUENCE_SETUP 메시지를 명시적으로 보낼 수 있습니다.
capCaptureSingleFrame capCaptureSingleFrame 매크로는 capCaptureSingleFrameOpen 매크로를 사용하여 연 캡처 파일에 단일 프레임을 추가합니다. 이 매크로를 사용하거나 WM_CAP_SINGLE_FRAME 메시지를 명시적으로 보낼 수 있습니다.
capCaptureSingleFrameClose capCaptureSingleFrameClose 매크로는 capCaptureSingleFrameOpen 매크로에서 연 캡처 파일을 닫습니다. 이 매크로를 사용하거나 WM_CAP_SINGLE_FRAME_CLOSE 메시지를 명시적으로 보낼 수 있습니다.
capCaptureSingleFrameOpen capCaptureSingleFrameOpen 매크로는 단일 프레임 캡처를 위한 캡처 파일을 엽니다. 캡처 파일의 모든 이전 정보를 덮어씁니다. 이 매크로를 사용하거나 WM_CAP_SINGLE_FRAME_OPEN 메시지를 명시적으로 보낼 수 있습니다.
capCaptureStop capCaptureStop 매크로는 캡처 작업을 중지합니다. 이 매크로를 사용하거나 WM_CAP_STOP 메시지를 명시적으로 보낼 수 있습니다.
capCreateCaptureWindowA capCreateCaptureWindow 함수는 캡처 창을 만듭니다. (ANSI)
capCreateCaptureWindowW capCreateCaptureWindow 함수는 캡처 창을 만듭니다. (유니코드)

capDlgVideoCompression capDlgVideoCompression 매크로는 사용자가 캡처 프로세스 중에 사용할 압축기를 선택할 수 있는 대화 상자를 표시합니다.
capDlgVideoDisplay capDlgVideoDisplay 매크로는 사용자가 비디오 출력을 설정하거나 조정할 수 있는 대화 상자를 표시합니다.
capDlgVideoFormat capDlgVideoFormat 매크로는 사용자가 비디오 형식을 선택할 수 있는 대화 상자를 표시합니다.
capDlgVideoSource capDlgVideoSource 매크로는 사용자가 비디오 원본을 제어할 수 있는 대화 상자를 표시합니다.
capDriverConnect capDriverConnect 매크로는 캡처 창을 캡처 드라이버에 연결합니다. 이 매크로를 사용하거나 WM_CAP_DRIVER_CONNECT 메시지를 명시적으로 보낼 수 있습니다.
capDriverDisconnect capDriverDisconnect 매크로는 캡처 창에서 캡처 드라이버의 연결을 끊습니다. 이 매크로를 사용하거나 WM_CAP_DRIVER_DISCONNECT 메시지를 명시적으로 보낼 수 있습니다.
capDriverGetCaps capDriverGetCaps 매크로는 현재 캡처 창에 연결된 캡처 드라이버의 하드웨어 기능을 반환합니다. 이 매크로를 사용하거나 WM_CAP_DRIVER_GET_CAPS 메시지를 명시적으로 보낼 수 있습니다.
capDriverGetName capDriverGetName 매크로는 캡처 창에 연결된 캡처 드라이버의 이름을 반환합니다. 이 매크로를 사용하거나 WM_CAP_DRIVER_GET_NAME 메시지를 명시적으로 호출할 수 있습니다.
capDriverGetVersion capDriverGetVersion 매크로는 캡처 창에 연결된 캡처 드라이버의 버전 정보를 반환합니다. 이 매크로를 사용하거나 WM_CAP_DRIVER_GET_VERSION 메시지를 명시적으로 보낼 수 있습니다.
capEditCopy capEditCopy 매크로는 비디오 프레임 버퍼 및 연결된 팔레트의 내용을 클립보드에 복사합니다. 이 매크로를 사용하거나 WM_CAP_EDIT_COPY 메시지를 명시적으로 보낼 수 있습니다.

capFileAlloc capFileAlloc 매크로는 지정된 크기의 캡처 파일을 만듭니다(사전 할당). 이 매크로를 사용하거나 WM_CAP_FILE_ALLOCATE 메시지를 명시적으로 보낼 수 있습니다.
capFileGetCaptureFile capFileGetCaptureFile 매크로는 현재 캡처 파일의 이름을 반환합니다. 이 매크로를 사용하거나 WM_CAP_FILE_GET_CAPTURE_FILE 메시지를 명시적으로 호출할 수 있습니다.
capFileSaveAs capFileSaveAs 매크로는 캡처 파일의 내용을 다른 파일에 복사합니다. 이 매크로를 사용하거나 WM_CAP_FILE_SAVEAS 메시지를 명시적으로 호출할 수 있습니다.
capFileSaveDIB capFileSaveDIB 매크로는 현재 프레임을 DIB 파일에 복사합니다. 이 매크로를 사용하거나 WM_CAP_FILE_SAVEDIB 메시지를 명시적으로 호출할 수 있습니다.
capFileSetCaptureFile capFileSetCaptureFile 매크로는 비디오 캡처에 사용되는 파일의 이름을 지정합니다. 이 매크로를 사용하거나 WM_CAP_FILE_SET_CAPTURE_FILE 메시지를 명시적으로 호출할 수 있습니다.
capFileSetInfoChunk capFileSetInfoChunk 매크로는 정보 청크를 설정하고 지웁니다. 캡처하는 동안 정보 청크를 AVI 파일에 삽입하여 텍스트 문자열 또는 사용자 지정 데이터를 포함할 수 있습니다. 이 매크로를 사용하거나 WM_CAP_FILE_SET_INFOCHUNK 메시지를 명시적으로 호출할 수 있습니다.
capGetAudioFormat capGetAudioFormat 매크로는 오디오 형식을 가져옵니다. 이 매크로를 사용하거나 WM_CAP_GET_AUDIOFORMAT 메시지를 명시적으로 호출할 수 있습니다.
capGetAudioFormatSize capGetAudioFormatSize 매크로는 오디오 형식의 크기를 가져옵니다. 이 매크로를 사용하거나 WM_CAP_GET_AUDIOFORMAT 메시지를 명시적으로 호출할 수 있습니다.
capGetDriverDescriptionA capGetDriverDescription 함수는 캡처 드라이버의 버전 설명을 검색합니다. (ANSI)
capGetDriverDescriptionW capGetDriverDescription 함수는 캡처 드라이버의 버전 설명을 검색합니다. (유니코드)

capGetMCIDeviceName capGetMCIDeviceName 매크로는 capSetMCIDeviceName 매크로를 사용하여 이전에 설정한 MCI 디바이스의 이름을 검색합니다. 이 매크로를 사용하거나 WM_CAP_GET_MCI_DEVICE 메시지를 명시적으로 호출할 수 있습니다.
capGetStatus capGetStatus 매크로는 캡처 창의 상태 검색합니다. 이 매크로를 사용하거나 WM_CAP_GET_STATUS 메시지를 명시적으로 호출할 수 있습니다.
capGetUserData capGetUserData 매크로는 캡처 창과 연결된 LONG_PTR 데이터 값을 검색합니다. 이 매크로를 사용하거나 WM_CAP_GET_USER_DATA 메시지를 명시적으로 호출할 수 있습니다.
capGetVideoFormat capGetVideoFormat 매크로는 사용 중인 비디오 형식의 복사본을 검색합니다. 이 매크로를 사용하거나 WM_CAP_GET_VIDEOFORMAT 메시지를 명시적으로 호출할 수 있습니다.
capGetVideoFormatSize capGetVideoFormatSize 매크로는 비디오 형식에 필요한 크기를 검색합니다. 이 매크로를 사용하거나 WM_CAP_GET_VIDEOFORMAT 메시지를 명시적으로 호출할 수 있습니다.
capGrabFrame capGrabFrame 매크로는 캡처 드라이버에서 단일 프레임을 검색하고 표시합니다. 캡처 후 오버레이 및 미리 보기를 사용할 수 없습니다. 이 매크로를 사용하거나 WM_CAP_GRAB_FRAME 메시지를 명시적으로 호출할 수 있습니다.
capGrabFrameNoStop capGrabFrameNoStop 매크로는 캡처 디바이스에서 압축되지 않은 단일 프레임으로 프레임 버퍼를 채우고 표시합니다.
capOverlay capOverlay 매크로는 오버레이 모드를 사용하거나 사용하지 않도록 설정합니다. 오버레이 모드에서는 하드웨어 오버레이를 사용하여 비디오가 표시됩니다. 이 매크로를 사용하거나 WM_CAP_SET_OVERLAY 메시지를 명시적으로 호출할 수 있습니다.
capPaletteAuto capPaletteAuto 매크로는 캡처 드라이버가 비디오 프레임을 샘플링하고 자동으로 새 팔레트를 만들 것을 요청합니다. 이 매크로를 사용하거나 WM_CAP_PAL_AUTOCREATE 메시지를 명시적으로 호출할 수 있습니다.

capPaletteManual capPaletteManual 매크로는 캡처 드라이버가 비디오 프레임을 수동으로 샘플링하고 새 팔레트를 만들 것을 요청합니다. 이 매크로를 사용하거나 WM_CAP_PAL_MANUALCREATE 메시지를 명시적으로 호출할 수 있습니다.
capPaletteOpen capPaletteOpen 매크로는 색상표 파일에서 새 팔레트를 로드하고 캡처 드라이버에 전달합니다.
capPalettePaste capPalettePaste 매크로는 클립보드에서 팔레트를 복사하여 캡처 드라이버에 전달합니다. 이 매크로를 사용하거나 WM_CAP_PAL_PASTE 메시지를 명시적으로 호출할 수 있습니다.
capPaletteSave capPaletteSave 매크로는 현재 색상표를 팔레트 파일에 저장합니다. 색상표 파일은 일반적으로 파일 이름 확장명 를 사용합니다. Pal. 이 매크로를 사용하거나 WM_CAP_PAL_SAVE 메시지를 명시적으로 보낼 수 있습니다.
capPreview capPreview 매크로는 미리 보기 모드를 사용하거나 사용하지 않도록 설정합니다.
capPreviewRate capPreviewRate 매크로는 프레임 표시 속도를 미리 보기 모드로 설정합니다. 이 매크로를 사용하거나 WM_CAP_SET_PREVIEWRATE 메시지를 명시적으로 호출할 수 있습니다.
capPreviewScale capPreviewScale 매크로는 미리 보기 비디오 이미지의 크기 조정을 사용하거나 사용하지 않도록 설정합니다.
capSetAudioFormat capSetAudioFormat 매크로는 스트리밍 또는 단계 캡처를 수행할 때 사용할 오디오 형식을 설정합니다. 이 매크로를 사용하거나 WM_CAP_SET_AUDIOFORMAT 메시지를 명시적으로 호출할 수 있습니다.
capSetCallbackOnCapControl capSetCallbackOnCapControl 매크로는 애플리케이션에서 콜백 함수를 설정하여 정확한 기록 제어를 제공합니다. 이 매크로를 사용하거나 WM_CAP_SET_CALLBACK_CAPCONTROL 메시지를 명시적으로 호출할 수 있습니다.
capSetCallbackOnError

capSetCallbackOnError 매크로는 클라이언트 애플리케이션에서 오류 콜백 함수를 설정합니다. 오류가 발생하면 AVICap에서 이 절차를 호출합니다. 이 매크로를 사용하거나 WM_CAP_SET_CALLBACK_ERROR 메시지를 명시적으로 호출할 수 있습니다.
capSetCallbackOnFrame capSetCallbackOnFrame 매크로는 애플리케이션에서 미리 보기 콜백 함수를 설정합니다. AVICap은 캡처 창이 미리 보기 프레임을 캡처할 때 이 절차를 호출합니다. 이 매크로를 사용하거나 WM_CAP_SET_CALLBACK_FRAME 메시지를 명시적으로 호출할 수 있습니다.
capSetCallbackOnStatus capSetCallbackOnStatus 매크로는 애플리케이션에서 상태 콜백 함수를 설정합니다. AVICap은 캡처 창이 변경 상태 때마다 이 절차를 호출합니다. 이 매크로를 사용하거나 WM_CAP_SET_CALLBACK_STATUS 메시지를 명시적으로 호출할 수 있습니다.
capSetCallbackOnVideoStream capSetCallbackOnVideoStream 매크로는 애플리케이션에서 콜백 함수를 설정합니다. AVICap은 비디오 버퍼가 채워지면 스트리밍 캡처 중에 이 절차를 호출합니다. 이 매크로를 사용하거나 WM_CAP_SET_CALLBACK_VIDEOSTREAM 메시지를 명시적으로 호출할 수 있습니다.
capSetCallbackOnWaveStream capSetCallbackOnWaveStream 매크로는 애플리케이션에서 콜백 함수를 설정합니다.
capSetCallbackOnYield capSetCallbackOnYield 매크로는 애플리케이션에서 콜백 함수를 설정합니다. AVICap은 스트리밍 캡처 중에 캡처 창이 생성되면 이 절차를 호출합니다. 이 매크로를 사용하거나 WM_CAP_SET_CALLBACK_YIELD 메시지를 명시적으로 호출할 수 있습니다.
capSetMCIDeviceName capSetMCIDeviceName 매크로는 데이터를 캡처하는 데 사용할 MCI 비디오 디바이스의 이름을 지정합니다. 이 매크로를 사용하거나 WM_CAP_SET_MCI_DEVICE 메시지를 명시적으로 호출할 수 있습니다.
capSetScrollPos capSetScrollPos 매크로는 캡처 창에 표시할 비디오 프레임의 부분을 정의합니다.
capSetUserData capSetUserData 매크로는 LONG_PTR 데이터 값을 캡처 창과 연결합니다. 이 매크로를 사용하거나 WM_CAP_SET_USER_DATA 메시지를 명시적으로 호출할 수 있습니다.
capSetVideoFormat

capSetVideoFormat 매크로는 캡처된 비디오 데이터의 형식을 설정합니다. 이 매크로를 사용하거나 WM_CAP_SET_VIDEOFORMAT 메시지를 명시적으로 호출할 수 있습니다.
CreateEditableStream CreateEditableStream 함수는 편집 가능한 스트림을 만듭니다. 다른 스트림 편집 함수를 사용하기 전에 이 함수를 사용합니다.
DrawDibBegin DrawDib 함수는 DrawDib DC의 매개 변수를 변경하거나 새 DrawDib DC를 초기화합니다.
DrawDibChangePalette DrawDibChangePalette 함수는 DIB 그리기에 사용되는 색상표 항목을 설정합니다.
DrawDibClose DrawDibClose 함수는 DrawDib DC를 닫고 DrawDib가 할당한 리소스를 해제합니다.
DrawDibDraw DrawDibDraw 함수는 화면에 DIB를 그립니다.
DrawDibEnd DrawDibEnd 함수는 DrawDibBegin 또는 DrawDibDraw 함수로 설정된 DrawDib DC의 플래그 및 기타 설정을 지웁니다.
DrawDibGetBuffer DrawDibGetBuffer 함수는 압축 해제를 위해 DrawDib에서 사용하는 버퍼의 위치를 검색합니다.
DrawDibGetPalette DrawDibGetPalette 함수는 DrawDib DC에서 사용하는 팔레트를 검색합니다.
DrawDibOpen DrawDibOpen 함수는 사용할 DrawDib 라이브러리를 열고 그리기용 DrawDib DC를 만듭니다.
DrawDibProfileDisplay DrawDibProfileDisplay 함수는 DrawDib 함수를 사용할 때 디스플레이 시스템의 설정을 결정합니다.
DrawDibRealize DrawDibRealize 함수는 지정된 DC와 함께 사용할 DrawDib DC의 팔레트를 인식합니다.

DrawDibSetPalette DrawDibSetPalette 함수는 DIB 그리기에 사용되는 팔레트를 설정합니다.
DrawDibStart DrawDibStart 함수는 스트리밍 재생을 위해 DrawDib DC를 준비합니다.
DrawDibStop DrawDibStop 함수는 스트리밍 재생을 위해 DrawDib DC에서 사용하는 리소스를 해제합니다.
DrawDibTime DrawDibTime 함수는 그리기 작업에 대한 타이밍 정보를 검색하고 디버그 작업 중에 사용됩니다.
DrawDibUpdate DrawDibUpdate 매크로는 DrawDib 오프 스크린 버퍼의 마지막 프레임을 그립니다.
EditStreamClone EditStreamClone 함수는 중복 편집 가능한 스트림을 만듭니다.
EditStreamCopy EditStreamCopy 함수는 편집 가능한 스트림(또는 그 일부)을 임시 스트림에 복사합니다.
EditStreamCut EditStreamCut 함수는 편집 가능한 스트림의 전체 또는 일부를 삭제하고 스트림의 삭제된 부분에서 임시 편집 가능한 스트림을 만듭니다.
EditStreamPaste EditStreamPaste 함수는 한 스트림에서 스트림(또는 그 일부)을 복사하여 지정된 위치에 있는 다른 스트림 내에 붙여넣습니다.
EditStreamSetInfoA EditStreamSetInfo 함수는 편집 가능한 스트림의 특성을 변경합니다. (ANSI)
EditStreamSetInfoW EditStreamSetInfo 함수는 편집 가능한 스트림의 특성을 변경합니다. (유니코드)
EditStreamSetNameA EditStreamSetName 함수는 설명 문자열을 스트림에 할당합니다. (ANSI)

EditStreamSetNameW EditStreamSetName 함수는 설명 문자열을 스트림에 할당합니다. (유니코드)
GetOpenFileNamePreviewA GetOpenFileNamePreview 함수는 열기 대화 상자를 사용하여 파일을 선택합니다. 또한 대화 상자를 사용하면 사용자가 현재 지정된 AVI 파일을 미리 볼 수 있습니다. 이 함수는 GetOpenFileName 함수에 있는 기능을 보강합니다. (ANSI)
GetOpenFileNamePreviewW GetOpenFileNamePreview 함수는 열기 대화 상자를 사용하여 파일을 선택합니다. 또한 대화 상자를 사용하면 사용자가 현재 지정된 AVI 파일을 미리 볼 수 있습니다. 이 함수는 GetOpenFileName 함수에 있는 기능을 보강합니다. (유니코드)
GetSaveFileNamePreviewA GetSaveFileNamePreview 함수는 다른 이름으로 저장 대화 상자를 사용하여 파일을 선택합니다. 대화 상자를 사용하면 사용자가 현재 지정된 파일을 미리 볼 수도 있습니다. 이 함수는 GetSaveFileName 함수에 있는 기능을 보강합니다. (ANSI)
GetSaveFileNamePreviewW GetSaveFileNamePreview 함수는 다른 이름으로 저장 대화 상자를 사용하여 파일을 선택합니다. 대화 상자를 사용하면 사용자가 현재 지정된 파일을 미리 볼 수도 있습니다. 이 함수는 GetSaveFileName 함수에 있는 기능을 보강합니다. (유니코드)
ICAbout ICAbout 매크로는 비디오 압축 드라이버에 정보 대화 상자를 표시하도록 알 수 있습니다. 이 매크로를 사용하거나 ICM_ABOUT 메시지를 명시적으로 호출할 수 있습니다.
ICClose ICClose 함수는 압축기 또는 압축 해제기를 닫습니다.
ICCompress ICCompress 함수는 단일 비디오 이미지를 압축합니다.
ICCompressBegin ICCompressBegin 매크로는 비디오 압축 드라이버에 데이터 압축을 준비하도록 알 수 있습니다. 이 매크로를 사용하거나 ICM_COMPRESS_BEGIN 메시지를 명시적으로 호출할 수 있습니다.
ICCompressEnd ICCompressEnd 매크로는 압축을 위해 할당된 압축 및 사용 가능한 리소스를 종료하도록 비디오 압

축 드라이버에 알 보냅니다. 이 매크로를 사용하거나 ICM_COMPRESS_END 메시지를 명시적으로 호출할 수 있습니다.
ICCompressGetFormat ICCompressGetFormat 매크로는 비디오 압축 드라이버에서 압축된 데이터의 출력 형식을 요청합니다. 이 매크로를 사용하거나 ICM_COMPRESS_GET_FORMAT 메시지를 명시적으로 호출할 수 있습니다.
ICCompressGetFormatSize ICCompressGetFormatSize 매크로는 비디오 압축 드라이버에서 압축된 데이터의 출력 형식 크기를 요청합니다. 이 매크로를 사용하거나 ICM_COMPRESS_GET_FORMAT 메시지를 명시적으로 호출할 수 있습니다.
ICCompressGetSize ICCompressGetSize 매크로는 비디오 압축 드라이버가 지정된 출력 형식으로 압축될 때 데이터 프레임의 최대 크기를 제공하도록 요청합니다. 이 매크로를 사용하거나 ICM_COMPRESS_GET_SIZE 메시지를 명시적으로 호출할 수 있습니다.
ICCompressorChoose ICCompressorChoose 함수는 사용자가 압축기를 선택할 수 있는 대화 상자를 표시합니다. 이 함수는 등록된 모든 압축기를 표시하거나 특정 형식을 지원하는 압축기만 나열할 수 있습니다.
ICCompressorFree ICCompressorFree 함수는 다른 VCM 함수에서 사용하는 COMPVARS 구조의 리소스를 해제합니다.
ICCompressQuery ICCompressQuery 매크로는 비디오 압축 드라이버를 쿼리하여 특정 입력 형식을 지원하는지 또는 특정 입력 형식을 특정 출력 형식으로 압축할 수 있는지 확인합니다.
ICConfigure ICConfigure 매크로는 비디오 압축 드라이버에 구성 대화 상자를 표시하도록 알 수 있습니다. 이 매크로를 사용하거나 ICM_CONFIGURE 메시지를 명시적으로 보낼 수 있습니다.
ICDecompress ICDecompress 함수는 단일 비디오 프레임의 압축을 풉니다.
ICDecompressBegin ICDecompressBegin 매크로는 비디오 압축 해제 드라이버에 데이터 압축 해제를 준비하도록 알 수 있습니다. 이 매크로를 사용하거나 ICM_DECOMPRESS_BEGIN 메시지를 명시적으로 호출할 수 있습니다.

ICDecompressEnd ICDecompressEnd 매크로는 비디오 압축 해제 드라이버에 압축 해제를 종료하도록 알리고 압축 해제를 위해 할당된 리소스를 해제합니다. 이 매크로를 사용하거나 ICM_DECOMPRESS_END 메시지를 명시적으로 호출할 수 있습니다.
ICDecompressEx ICDecompressEx 함수는 단일 비디오 프레임의 압축을 풉니다.
ICDecompressExBegin ICDecompressExBegin 함수는 데이터 압축을 풀기 위한 압축 해제기를 준비합니다.
ICDecompressExEnd ICDecompressExEnd 매크로는 압축 해제를 위해 할당된 압축 해제 및 사용 가능한 리소스를 종료하도록 비디오 압축 해제 드라이버에 알 보냅니다. 이 매크로를 사용하거나 ICM_DECOMPRESSEX_END 메시지를 명시적으로 호출할 수 있습니다.
ICDecompressExQuery ICDecompressExQuery 함수는 압축 해제기가 특정 형식으로 데이터를 압축 해제할 수 있는지 여부를 결정합니다.
ICDecompressGetFormat ICDecompressGetFormat 매크로는 비디오 압축 해제 드라이버에서 압축 해제된 데이터의 출력 형식을 요청합니다. 이 매크로를 사용하거나 ICM_DECOMPRESS_GET_FORMAT 메시지를 명시적으로 호출할 수 있습니다.
ICDecompressGetFormatSize ICDecompressGetFormatSize 매크로는 비디오 압축 해제 드라이버에서 압축 해제된 데이터의 출력 형식 크기를 요청합니다. 이 매크로를 사용하거나 ICM_DECOMPRESS_GET_FORMAT 메시지를 명시적으로 호출할 수 있습니다.
ICDecompressGetPalette ICDecompressGetPalette 매크로는 비디오 압축 해제 드라이버가 출력 BITMAPINFOHEADER 구조체의 색 테이블을 제공해 줄 것을 요청합니다. 이 매크로를 사용하거나 ICM_DECOMPRESS_GET_PALETTE 메시지를 명시적으로 호출할 수 있습니다.
ICDecompressOpen ICDecompressOpen 매크로는 지정된 형식과 호환되는 압축 해제기를 엽니다.
ICDecompressQuery

ICDecompressQuery 매크로는 비디오 압축 해제 드라이버를 쿼리하여 특정 입력 형식을 지원하는지 또는 특정 입력 형식을 특정 출력 형식으로 압축 해제할 수 있는지 확인합니다.
ICDecompressSetPalette ICDecompressSetPalette 매크로는 색상표를 사용하는 형식으로 압축을 풀 때 사용할 비디오 압축 해제 드라이버의 팔레트를 지정합니다. 이 매크로를 사용하거나 ICM_DECOMPRESS_SET_PALETTE 메시지를 명시적으로 호출할 수 있습니다.
ICDraw ICDraw 함수는 그리기 위해 이미지를 압축 해제합니다.
ICDrawBegin ICDrawBegin 함수는 렌더러를 초기화하고 그리기 대상을 준비합니다.
ICDrawChangePalette ICDrawChangePalette 매크로는 렌더링 드라이버에 영화 팔레트가 변경되고 있음을 알립니다. 이 매크로를 사용하거나 ICM_DRAW_CHANGEPALETTE 메시지를 명시적으로 호출할 수 있습니다.
ICDrawEnd ICDrawEnd 매크로는 렌더링 드라이버에 현재 이미지를 화면으로 압축 해제하고 압축 해제 및 그리기에 할당된 리소스를 해제하도록 알립니다. 이 매크로를 사용하거나 ICM_DRAW_END 메시지를 명시적으로 호출할 수 있습니다.
ICDrawFlush ICDrawFlush 매크로는 렌더링 드라이버에 그리기를 기다리는 모든 이미지 버퍼의 콘텐츠를 렌더링하도록 알립니다. 이 매크로를 사용하거나 ICM_DRAW_FLUSH 메시지를 명시적으로 호출할 수 있습니다.
ICDrawGetTime ICDrawGetTime 매크로는 그리기 프레임의 타이밍을 제어하는 렌더링 드라이버를 요청하여 내부 클록의 현재 값을 반환합니다. 이 매크로를 사용하거나 ICM_DRAW_GETTIME 메시지를 명시적으로 호출할 수 있습니다.
ICDrawOpen ICDrawOpen 매크로는 지정된 형식으로 이미지를 그릴 수 있는 드라이버를 엽니다.
ICDrawQuery ICDrawQuery 매크로는 렌더링 드라이버를 쿼리하여 특정 형식으로 데이터를 렌더링할 수 있는지 확인합니다. 이 매크로를 사용하거나 ICM_DRAW_QUERY 메시지를 명시적으로 호출할 수 있습니다.

ICDrawRealize ICDrawRealize 매크로는 렌더링 드라이버에 그리는 동안 그리기 팔레트를 실현하도록 알립니다. 이 매크로를 사용하거나 ICM_DRAW_REALIZE 메시지를 명시적으로 호출할 수 있습니다.
ICDrawRenderBuffer ICDrawRenderBuffer 매크로는 렌더링 드라이버에 전달된 프레임을 그리도록 알립니다. 이 매크로를 사용하거나 ICM_DRAW_RENDERBUFFER 메시지를 명시적으로 호출할 수 있습니다.
ICDrawSetTime ICDrawSetTime 매크로는 그리기 프레임의 타이밍을 처리하는 렌더링 드라이버에 동기화 정보를 제공합니다.
ICDrawStart ICDrawStart 매크로는 렌더링 드라이버에 그리기 프레임 타이밍에 대한 내부 클록을 시작하도록 알립니다. 이 매크로를 사용하거나 ICM_DRAW_START 메시지를 명시적으로 호출할 수 있습니다.
ICDrawStartPlay ICDrawStartPlay 매크로는 렌더링 드라이버에 재생 작업의 시작 및 종료 시간을 제공합니다. 이 매크로를 사용하거나 ICM_DRAW_START_PLAY 메시지를 명시적으로 호출할 수 있습니다.
ICDrawStop ICDrawStop 매크로는 렌더링 드라이버에 그리기 프레임 타이밍에 대한 내부 클록을 중지하도록 알립니다. 이 매크로를 사용하거나 ICM_DRAW_STOP 메시지를 명시적으로 호출할 수 있습니다.
ICDrawStopPlay ICDrawStopPlay 매크로는 재생 작업이 완료되면 렌더링 드라이버에 알 수 있습니다. 이 매크로를 사용하거나 ICM_DRAW_STOP_PLAY 메시지를 명시적으로 호출할 수 있습니다.
ICDrawSuggestFormat ICDrawSuggestFormat 함수는 그리기 처리기에 입력 데이터 형식을 제안하도록 알립니다.
ICDrawWindow ICDrawWindow 매크로는 렌더링 드라이버에 ICM_DRAW_BEGIN 메시지에 대해 지정된 창을 다시 그려야 한다는 것을 알립니다. 창이 이동되었거나 일시적으로 가려집니다. 이 매크로를 사용하거나 ICM_DRAW_WINDOW 메시지를 명시적으로 호출할 수 있습니다.
ICGetBuffersWanted ICGetBuffersWanted 매크로는 드라이버에 할당할 버퍼 수를 쿼리합니다. 이 매크로를 사용하거나 ICM_GETBUFFERSWANTED 메시지를 명시적으로 호출할 수 있습니다.

ICGetDefaultKeyFrameRate ICGetDefaultKeyFrameRate 매크로는 비디오 압축 드라이버에서 기본 키 프레임 간격(또는 기본 설정)을 쿼리합니다. 이 매크로를 사용하거나 ICM_GETDEFAULTKEYFRAMERATE 메시지를 명시적으로 호출할 수 있습니다.
ICGetDefaultQuality ICGetDefaultQuality 매크로는 비디오 압축 드라이버를 쿼리하여 기본 품질 설정을 제공합니다. 이 매크로를 사용하거나 ICM_GETDEFAULTQUALITY 메시지를 명시적으로 호출할 수 있습니다.
ICGetDisplayFormat ICGetDisplayFormat 함수는 압축된 이미지를 표시하는 데 사용할 수 있는 최상의 형식을 결정합니다. 또한 열린 압축기의 핸들을 지정하지 않은 경우 함수는 압축기를 엽니다.
ICGetInfo ICGetInfo 함수는 압축기 정보를 가져옵니다.
ICGetState ICGetState 매크로는 비디오 압축 드라이버를 쿼리하여 메모리 블록에서 현재 구성을 반환합니다. 이 매크로를 사용하거나 ICM_GETSTATE 메시지를 명시적으로 호출할 수 있습니다.
ICGetStateSize ICGetStateSize 매크로는 비디오 압축 드라이버를 쿼리하여 구성 정보를 검색하는 데 필요한 메모리 양을 확인합니다. 이 매크로를 사용하거나 ICM_GETSTATE 메시지를 명시적으로 호출할 수 있습니다.
ICImageCompress ICImageCompress 함수는 이미지를 지정된 크기로 압축합니다. 이 함수에는 초기화 함수가 필요하지 않습니다.
ICImageDecompress ICImageDecompress 함수는 초기화 함수를 사용하지 않고 이미지를 압축 해제합니다.
ICInfo ICInfo 함수는 설치된 특정 압축기에 대한 정보를 검색하거나 설치된 압축기를 열거합니다.
ICInstall ICInstall 함수는 새 압축기 또는 압축 해제기를 설치합니다.

ICLocate ICLocate 함수는 지정된 형식으로 이미지를 처리할 수 있는 압축기 또는 압축 해제기를 찾거나 지정된 형식의 이미지를 하드웨어로 직접 압축 해제할 수 있는 드라이버를 찾습니다.
ICOpen ICOpen 함수는 압축기 또는 압축 해제기를 엽니다.
ICOpenFunction ICOpenFunction 함수는 함수로 정의된 압축기 또는 압축 해제기를 엽니다.
ICQueryAbout ICQueryAbout 매크로는 비디오 압축 드라이버를 쿼리하여 정보 대화 상자가 있는지 확인합니다. 이 매크로를 사용하거나 ICM_ABOUT 메시지를 명시적으로 호출할 수 있습니다.
ICQueryConfigure ICQueryConfigure 매크로는 비디오 압축 드라이버를 쿼리하여 구성 대화 상자가 있는지 확인합니다. 이 매크로를 사용하거나 ICM_CONFIGURE 메시지를 명시적으로 보낼 수 있습니다.
ICRemove ICRemove 함수는 설치된 압축기를 제거합니다.
ICSendMessage ICSendMessage 함수는 압축기로 메시지를 보냅니다.
ICSeqCompressFrame ICSeqCompressFrame 함수는 프레임 시퀀스에서 한 프레임을 압축합니다.
ICSeqCompressFrameEnd ICSeqCompressFrameEnd 함수는 ICSeqCompressFrameStart 및 ICSeqCompressFrame 함수를 사용하여 시작된 시퀀스 압축을 종료합니다.
ICSeqCompressFrameStart ICSeqCompressFrameStart 함수는 ICSeqCompressFrame 함수를 사용하여 프레임 시퀀스를 압축하기 위한 리소스를 초기화합니다.
ICSetState ICSetState 매크로는 비디오 압축 드라이버에 압축기의 상태를 설정하도록 알 수 있습니다. 이 매크로를 사용하거나 ICM_SETSTATE 메시지를 명시적으로 호출할 수 있습니다.

ICSetStatusProc ICSetStatusProc 함수는 상태 콜백 함수의 주소를 압축기로 보냅니다. 압축기는 긴 작업 중에 이 함수를 호출합니다.
MCIWndCanConfig MCIWndCanConfig 매크로는 MCI 디바이스가 구성 대화 상자를 표시할 수 있는지 여부를 결정합니다. 이 매크로를 사용하거나 MCIWNDM_CAN_CONFIG 메시지를 명시적으로 보낼 수 있습니다.
MCIWndCanEject MCIWndCanEject 매크로는 MCI 디바이스가 미디어를 배출할 수 있는지 여부를 결정합니다. 이 매크로를 사용하거나 MCIWNDM_CAN_EJECT 메시지를 명시적으로 보낼 수 있습니다.
MCIWndCanPlay MCIWndCanPlay 매크로는 MCI 디바이스가 다른 종류의 데이터 파일 또는 콘텐츠를 재생할 수 있는지 여부를 결정합니다. 이 매크로를 사용하거나 MCIWNDM_CAN_PLAY 메시지를 명시적으로 보낼 수 있습니다.
MCIWndCanRecord MCIWndCanRecord 매크로는 MCI 디바이스가 기록을 지원하는지 여부를 결정합니다. 이 매크로를 사용하거나 MCIWNDM_CAN_RECORD 메시지를 명시적으로 보낼 수 있습니다.
MCIWndCanSave MCIWndCanSave 매크로는 MCI 디바이스가 데이터를 저장할 수 있는지 여부를 결정합니다. 이 매크로를 사용하거나 MCIWNDM_CAN_SAVE 메시지를 명시적으로 보낼 수 있습니다.
MCIWndCanWindow MCIWndCanWindow 매크로는 MCI 디바이스가 창 지향 MCI 명령을 지원하는지 여부를 결정합니다. 이 매크로를 사용하거나 MCIWNDM_CAN_WINDOW 메시지를 명시적으로 보낼 수 있습니다.
MCIWndChangeStyles MCIWndChangeStyles 매크로는 MCIWnd 창에서 사용하는 스타일을 변경합니다. 이 매크로를 사용하거나 MCIWNDM_CHANGESTYLES 메시지를 명시적으로 보낼 수 있습니다.
MCIWndClose MCIWndClose 매크로는 MCIWnd 창과 연결된 MCI 디바이스 또는 파일을 닫습니다.
MCIWndCreateA MCIWndCreate 함수는 MCIWnd 창 클래스를 등록하고 MCI 서비스를 사용하기 위한 MCIWnd 창을

만듭니다. MCIWndCreate는 MCI 디바이스 또는 파일(예: AVI 파일)을 열고 MCIWnd 창과 연결할 수도 있습니다. (ANSI)
MCIWndCreateW MCIWndCreate 함수는 MCIWnd 창 클래스를 등록하고 MCI 서비스를 사용하기 위한 MCIWnd 창을 만듭니다. MCIWndCreate는 MCI 디바이스 또는 파일(예: AVI 파일)을 열고 MCIWnd 창과 연결할 수도 있습니다. (유니코드)
MCIWndDestroy MCIWndDestroy 매크로는 MCIWnd 창과 연결된 MCI 디바이스 또는 파일을 닫고 창을 삭제합니다. 이 매크로를 사용하거나 WM_CLOSE 메시지를 명시적으로 보낼 수 있습니다.
MCIWndEject MCIWndEject 매크로는 MCI 디바이스에 명령을 보내 미디어를 배출합니다. 이 매크로를 사용하거나 MCIWNDM_EJECT 메시지를 명시적으로 보낼 수 있습니다.
MCIWndEnd MCIWndEnd 매크로는 현재 위치를 콘텐츠의 끝으로 이동합니다. 이 매크로를 사용하거나 MCI_SEEK 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetActiveTimer MCIWndGetActiveTimer 매크로는 MCIWnd 창이 활성 창일 때 사용되는 업데이트 기간을 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETACTIVETIMER 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetAlias MCIWndGetAlias 매크로는 mciSendString 함수를 사용하여 MCI 디바이스 또는 파일을 여는 데 사용되는 별칭을 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETALIAS 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetDest MCIWndGetDest 매크로는 재생 중에 AVI 파일의 이미지를 확대하거나 확대하는 데 사용되는 대상 사각형의 좌표를 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GET_DEST 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetDevice MCIWndGetDevice 매크로는 현재 MCI 디바이스의 이름을 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETDEVICE 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetDeviceID MCIWndGetDeviceID 매크로는 mciSendCommand 함수와 함께 사용할 현재 MCI 디바이스의 식별

자를 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETDEVICEID 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetEnd MCIWndGetEnd 매크로는 MCI 디바이스 또는 파일의 콘텐츠 끝 위치를 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETEND 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetError MCIWndGetError 매크로는 마지막으로 발생한 MCI 오류를 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETERROR 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetFileName MCIWndGetFileName 매크로는 MCI 디바이스에서 사용하는 파일 이름을 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETFILENAME 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetInactiveTimer MCIWndGetInactiveTimer 매크로는 MCIWnd 창이 비활성 창일 때 사용되는 업데이트 기간을 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETINACTIVETIMER 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetLength MCIWndGetLength 매크로는 MCI 디바이스에서 현재 사용되는 콘텐츠 또는 파일의 길이를 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETLENGTH 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetMode MCIWndGetMode 매크로는 MCI 디바이스의 현재 작동 모드를 검색합니다. MCI 디바이스에는 상수로 지정된 여러 운영 모드가 있습니다. 이 매크로를 사용하거나 MCIWNDM_GETMODE 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetPalette MCIWndGetPalette 매크로는 MCI 디바이스에서 사용하는 팔레트의 핸들을 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETPALETTE 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetPosition MCIWndGetPosition 매크로는 MCI 디바이스의 콘텐츠 내에서 현재 위치의 숫자 값을 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETPOSITION 메시지를 명시적으로 보낼 수 있습니다.
MCIWndGetPositionString MCIWndGetPositionString 매크로는 MCI 디바이스의 콘텐츠 내에서 현재 위치의 숫자 값을 검색합니다.

MCIWndGetRepeat

MCIWndGetRepeat 매크로는 연속 재생이 활성화되었는지 여부를 결정합니다. 이 매크로를 사용하거나 MCIWNDM_GETREPEAT 메시지를 명시적으로 보낼 수 있습니다.

MCIWndGetSource

MCIWndGetSource 매크로는 재생 중에 AVI 파일의 이미지를 자르는 데 사용되는 원본 사각형의 좌표를 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GET_SOURCE 메시지를 명시적으로 보낼 수 있습니다.

MCIWndGetSpeed

MCIWndGetSpeed 매크로는 MCI 디바이스의 재생 속도를 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETSPEED 메시지를 명시적으로 보낼 수 있습니다.

MCIWndGetStart

MCIWndGetStart 매크로는 MCI 디바이스 또는 파일의 콘텐츠 시작 위치를 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETSTART 메시지를 명시적으로 보낼 수 있습니다.

MCIWndGetStyles

MCIWndGetStyles 매크로는 창에서 사용하는 현재 MCIWnd 창 스타일을 지정하는 플래그를 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETSTYLES 메시지를 명시적으로 보낼 수 있습니다.

MCIWndGetTimeFormat

MCIWndGetTimeFormat 매크로는 MCI 디바이스의 현재 시간 형식을 숫자 값과 문자열로 _as 두 가지 형식으로 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETTIMEFORMAT 메시지를 명시적으로 보낼 수 있습니다.

MCIWndGetVolume

MCIWndGetVolume 매크로는 MCI 디바이스의 현재 볼륨 설정을 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETVOLUME 메시지를 명시적으로 보낼 수 있습니다.

MCIWndGetZoom

MCIWndGetZoom 매크로는 MCI 디바이스에서 사용하는 현재 확대/축소 값을 검색합니다. 이 매크로를 사용하거나 MCIWNDM_GETZOOM 메시지를 명시적으로 보낼 수 있습니다.

MCIWndHome

MCIWndHome 매크로는 현재 위치를 콘텐츠의 시작 부분으로 이동합니다. 이 매크로를 사용하거나 MCI_SEEK 명령을 명시적으로 보낼 수 있습니다.

MCIWndNew

MCIWndNew 매크로는 현재 MCI 디바이스에 대한 새 파일을 만듭니다. 이 매크로를 사용하거나 MCIWNDM_NEW 메시지를 명시적으로 보낼 수 있습니다.
MCIWndOpen MCIWndOpen 매크로는 MCI 디바이스를 열고 MCIWnd 창과 연결합니다.
MCIWndOpenDialog MCIWndOpenDialog 매크로는 사용자가 지정한 데이터 파일과 해당 유형의 MCI 디바이스를 열고 MCIWnd 창에 연결합니다.
MCIWndOpenInterface MCIWndOpenInterface 매크로는 지정된 인터페이스와 연결된 데이터 스트림 또는 파일을 MCIWnd 창에 연결합니다. 이 매크로를 사용하거나 MCIWNDM_OPENINTERFACE 메시지를 명시적으로 보낼 수 있습니다.
MCIWndPause MCIWndPause 매크로는 MCI 디바이스에 명령을 보내 재생 또는 녹화를 일시 중지합니다.
MCIWndPlay MCIWndPlay 매크로는 MCI 디바이스에 명령을 보내 콘텐츠의 현재 위치에서 재생을 시작합니다. 이 매크로를 사용하거나 MCI_PLAY 명령을 명시적으로 보낼 수 있습니다.
MCIWndPlayFrom MCIWndPlayFrom 매크로는 지정된 위치에서 콘텐츠 끝까지 또는 다른 명령이 재생을 중지할 때까지 MCI 디바이스의 콘텐츠를 재생합니다. 이 매크로를 사용하거나 MCIWNDM_PLAYFROM 메시지를 명시적으로 보낼 수 있습니다.
MCIWndPlayFromTo MCIWndPlayFromTo 매크로는 지정된 시작 위치와 종료 위치 간에 콘텐츠의 일부를 재생합니다.
MCIWndPlayReverse MCIWndPlayReverse 매크로는 현재 위치에서 시작하여 콘텐츠의 시작 부분에서 또는 다른 명령이 재생을 중지할 때까지 현재 콘텐츠를 역방향으로 재생합니다.
MCIWndPlayTo MCIWndPlayTo 매크로는 현재 위치에서 지정된 종료 위치까지 또는 다른 명령이 재생을 중지할 때까지 MCI 디바이스의 콘텐츠를 재생합니다.
MCIWndPutDest MCIWndPutDest 매크로는 재생 중에 AVI 파일의 이미지를 확대하거나 확대하는 데 사용되는 대상

사각형의 좌표를 다시 정의합니다. 이 매크로를 사용하거나 MCIWNDM_PUT_DEST 메시지를 명시적으로 보낼 수 있습니다.
MCIWndPutSource MCIWndPutSource 매크로는 재생 중에 AVI 파일의 이미지를 자르는 데 사용되는 원본 사각형의 좌표를 다시 정의합니다. 이 매크로를 사용하거나 MCIWNDM_PUT_SOURCE 메시지를 명시적으로 보낼 수 있습니다.
MCIWndRealize MCIWndRealize 매크로는 전경 또는 백그라운드에서 MCI 창이 실현되는 방식을 제어합니다. 또한 이 매크로를 사용하면 MCI 창의 색상표가 프로세스에서 실현됩니다. 이 매크로를 사용하거나 MCIWNDM_REALIZE 메시지를 명시적으로 보낼 수 있습니다.
MCIWndRecord MCIWndRecord 매크로는 MCI 디바이스를 사용하여 콘텐츠 기록을 시작합니다. 녹음/녹화 프로세스는 콘텐츠의 현재 위치에서 시작하여 기록 기간 동안 기존 데이터를 덮어씁니다.
MCIWndRegisterClass MCIWndRegisterClass 함수는 MCI 창 클래스 MCIWND_WINDOW_CLASS 등록합니다.
MCIWndResume MCIWndResume 매크로는 일시 중지된 모드에서 재생 또는 녹화 콘텐츠를 다시 시작합니다. 이 매크로는 콘텐츠의 현재 위치에서 재생 또는 기록을 다시 시작합니다. 이 매크로를 사용하거나 MCI_RESUME 명령을 명시적으로 보낼 수 있습니다.
MCIWndReturnString MCIWndReturnString 매크로는 MCI 디바이스로 전송된 최신 MCI 문자열 명령에 대한 회신을 검색합니다. 회신의 정보는 null로 끝나는 문자열로 제공됩니다. 이 매크로를 사용하거나 MCIWNDM_RETURNSTRING 메시지를 명시적으로 보낼 수 있습니다.
MCIWndSave MCIWndSave 매크로는 MCI 디바이스에서 현재 사용되는 콘텐츠를 저장합니다.
MCIWndSaveDialog MCIWndSaveDialog 매크로는 MCI 디바이스에서 현재 사용하는 콘텐츠를 저장합니다. 이 매크로는 사용자가 콘텐츠를 저장할 파일 이름을 선택할 수 있도록 저장 대화 상자를 표시합니다. 이 매크로를 사용하거나 MCI_SAVE 명령을 명시적으로 보낼 수 있습니다.
MCIWndSeek

MCIWndSeek 매크로는 재생 위치를 콘텐츠의 지정된 위치로 이동합니다. 이 매크로를 사용하거나 MCI_SEEK 명령을 명시적으로 사용할 수 있습니다.

MCIWndSendString

MCIWndSendString 매크로는 MCIWnd 창과 연결된 디바이스에 문자열 형식의 MCI 명령을 보냅니다. 이 매크로를 사용하거나 MCIWNDM_SENDSTRING 메시지를 명시적으로 보낼 수 있습니다.

MCIWndSetActiveTimer

MCIWndSetActiveTimer 매크로는 MCIWnd 창에서 트랙바를 업데이트하고, 창 제목 표시줄에 표시되는 위치 정보를 업데이트하고, MCIWnd 창이 활성 상태일 때 부모 창에 알림 메시지를 보내는 데 MCIWnd에서 사용하는 업데이트 기간을 설정합니다. 이 매크로를 사용하거나 MCIWNDM_SETACTIVETIMER 메시지를 명시적으로 보낼 수 있습니다.

MCIWndSetInactiveTimer

MCIWndSetInactiveTimer 매크로는 MCIWnd에서 MCIWnd 창에서 트랙바를 업데이트하고, 창 제목 표시줄에 표시되는 위치 정보를 업데이트하고, MCIWnd 창이 비활성 상태일 때 부모 창에 알림 메시지를 보내는 데 사용하는 업데이트 기간을 설정합니다. 이 매크로를 사용하거나 MCIWNDM_SETINACTIVETIMER 메시지를 명시적으로 보낼 수 있습니다.

MCIWndSetOwner

MCIWndSetOwner 매크로는 MCIWnd 창과 연결된 알림 메시지를 수신하도록 창을 설정합니다. 이 매크로를 사용하거나 MCIWNDM_SETOWNER 메시지를 명시적으로 보낼 수 있습니다.

MCIWndSetPalette

MCIWndSetPalette 매크로는 MCIWnd 창과 연결된 MCI 디바이스에 팔레트 핸들을 보냅니다. 이 매크로를 사용하거나 MCIWNDM_SETPALETTE 메시지를 명시적으로 보낼 수 있습니다.

MCIWndSetRepeat

MCIWndSetRepeat 매크로는 연속 재생과 연결된 반복 플래그를 설정합니다. 이 매크로를 사용하거나 MCIWNDM_SETREPEAT 메시지를 명시적으로 보낼 수 있습니다.

MCIWndSetSpeed

MCIWndSetSpeed 매크로는 MCI 디바이스의 재생 속도를 설정합니다. 이 매크로를 사용하거나 MCIWNDM_SETSPEED 메시지를 명시적으로 보낼 수 있습니다.

MCIWndSetTimeFormat

MCIWndSetTimeFormat 매크로는 MCI 디바이스의 시간 형식을 설정합니다. 이 매크로를 사용하거나 MCIWNDM_SETTIMEFORMAT 메시지를 명시적으로 보낼 수 있습니다.

MCIWndSetTimers

MCIWndSetTimers 매크로는 MCIWnd에서 MCIWnd 창에서 트랙바를 업데이트하고, 창 제목 표시줄에 표시되는 위치 정보를 업데이트하고, 부모 창에 알림 메시지를 보내는 데 사용하는 업데이트 기간을 설정합니다.
MCIWndSetVolume MCIWndSetVolume 매크로는 MCI 디바이스의 볼륨 수준을 설정합니다. 이 매크로를 사용하거나 MCIWNDM_SETVOLUME 메시지를 명시적으로 보낼 수 있습니다.
MCIWndSetZoom MCIWndSetZoom 매크로는 확대/축소 요소에 따라 비디오 이미지의 크기를 조정합니다. 이 마르코는 일정한 가로 세로 비율을 유지하면서 MCIWnd 창의 크기를 조정합니다. 이 매크로를 사용하거나 MCIWNDM_SETZOOM 메시지를 명시적으로 보낼 수 있습니다.
MCIWndStep MCIWndStep 매크로는 콘텐츠의 현재 위치를 지정된 증분 단위로 앞으로 또는 뒤로 이동합니다. 이 매크로를 사용하거나 MCI_STEP 명령을 명시적으로 보낼 수 있습니다.
MCIWndStop MCIWndStop 매크로는 MCIWnd 창과 연결된 MCI 디바이스의 콘텐츠 재생 또는 기록을 중지합니다. 이 매크로를 사용하거나 MCI_STOP 명령을 명시적으로 보낼 수 있습니다.
MCIWndUseFrames MCIWndUseFrames 매크로는 MCI 디바이스의 시간 형식을 프레임으로 설정합니다. 이 매크로를 사용하거나 MCIWNDM_SETTIMEFORMAT 메시지를 명시적으로 보낼 수 있습니다.
MCIWndUseTime MCIWndUseTime 매크로는 MCI 디바이스의 시간 형식을 밀리초로 설정합니다. 이 매크로를 사용하거나 MCIWNDM_SETTIMEFORMAT 메시지를 명시적으로 보낼 수 있습니다.
MCIWndValidateMedia MCIWndValidateMedia 매크로는 콘텐츠의 시작 및 끝 위치, 콘텐츠의 현재 위치 및 현재 시간 형식에 따라 트랙바를 업데이트합니다.
mmioFOURCC mmioFOURCC 매크로는 4자를 4자 코드로 변환합니다.
StretchDIB StretchDIB 함수는 디바이스 독립적 비트맵을 한 메모리 위치에서 다른 메모리 위치로 복사하고 대상 사각형에 맞게 이미지 크기를 조정합니다.

콜백 함수

CAPCONTROLCALLBACK capControlCallback 함수는 스트리밍 캡처를 시작하고 종료하기 위해 정밀도 제어에 사용되는 콜백 함수입니다. capControlCallback 이름은 애플리케이션에서 제공하는 함수 이름의 자리 표시자입니다.
CAPERRORCALLBACKA capErrorCallback 함수는 비디오 캡처에 사용되는 오류 콜백 함수입니다. capErrorCallback 이름은 애플리케이션에서 제공하는 함수 이름의 자리 표시자입니다. (ANSI)
CAPERRORCALLBACKW capErrorCallback 함수는 비디오 캡처에 사용되는 오류 콜백 함수입니다. capErrorCallback 이름은 애플리케이션에서 제공하는 함수 이름의 자리 표시자입니다. (유니코드)
CAPSTATUSCALLBACKA capStatusCallback 함수는 비디오 캡처에 사용되는 상태 콜백 함수입니다. capStatusCallback 이름은 애플리케이션에서 제공하는 함수 이름의 자리 표시자입니다. (ANSI)
CAPSTATUSCALLBACKW capStatusCallback 함수는 비디오 캡처에 사용되는 상태 콜백 함수입니다. capStatusCallback 이름은 애플리케이션에서 제공하는 함수 이름의 자리 표시자입니다. (유니코드)
CAPVIDEOCALLBACK capVideoStreamCallback 함수는 캡처된 비디오 프레임을 선택적으로 처리하기 위해 스트리밍 캡처와 함께 사용되는 콜백 함수입니다. capVideoStreamCallback 이름은 애플리케이션에서 제공하는 함수 이름의 자리 표시자입니다.
CAPWAVECALLBACK capWaveStreamCallback 함수는 필요에 따라 오디오 데이터의 버퍼를 처리하기 위해 스트리밍 캡처와 함께 사용되는 콜백 함수입니다. capWaveStreamCallback 이름은 애플리케이션에서 제공하는 함수 이름의 자리 표시자입니다.
CAPYIELDCALLBACK capYieldCallback 함수는 비디오 캡처에 사용되는 수익률 콜백 함수입니다. capYieldCallback 이름은 애플리케이션에서 제공하는 함수 이름의 자리 표시자입니다.

구조체

AVICOMPRESSOPTIONS AVICOMPRESSOPTIONS 구조체에는 스트림 및 스트림이 압축 및 저장되는 방법에 대한 정보가 포함됩니다. 이 구조체는 데이터를 AVIMakeCompressedStream 함수(또는 AVIMakeCompressedStream을 사용하는 AVISave 함수)에 전달합니다.
AVIFILEINFOA AVIFILEINFO 구조체에는 전체 AVI 파일에 대한 전역 정보가 포함됩니다. (ANSI)
AVIFILEINFOW AVIFILEINFO 구조체에는 전체 AVI 파일에 대한 전역 정보가 포함됩니다. (유니코드)
AVISTREAMINFOA AVISTREAMINFO 구조체에는 단일 스트림에 대한 정보가 포함됩니다. (ANSI)
AVISTREAMINFOW AVISTREAMINFO 구조체에는 단일 스트림에 대한 정보가 포함됩니다. (유니코드)
CAPDRIVERCAPS CAPDRIVERCAPS 구조체는 캡처 드라이버의 기능을 정의합니다. 애플리케이션은 캡처 창을 캡처 드라이버에 연결할 때마다 WM_CAP_DRIVER_GET_CAPS 메시지 또는 capDriverGetCaps 매크로를 사용하여 드라이버 기능의 복사본을 CAPDRIVERCAPS 구조에 배치해야 합니다.
CAPINFOCHUNK CAPINFOCHUNK 구조에는 AVI 캡처 파일 내에서 정보 청크를 정의하는 데 사용할 수 있는 매개 변수가 포함되어 있습니다. WM_CAP_FILE_SET_INFOCHUNK 메시지 또는 capSetInfoChunk 매크로는 CAPINFOCHUNK 구조를 캡처 창으로 보내는 데 사용됩니다.
CAPSTATUS CAPSTATUS 구조체는 캡처 창의 현재 상태를 정의합니다.
CAPTUREPARMS CAPTUREPARMS 구조에는 스트리밍 비디오 캡처 프로세스를 제어하는 매개 변수가 포함되어 있습니다. 이 구조는 캡처 속도, 캡처하는 동안 사용할 버퍼 수 및 캡처가 종료되는 방법에 영향을 주는 매개 변수를 가져와서 설정하는 데 사용됩니다.
COMPVARS COMPVARS 구조체는 ICCompressorChoose, ICSeqCompressFrame 및 ICCompressorFree와 같은 함수에 대한 압축기 설정을 설명합니다.

DRAWDIBTIME DRAWDIBTIME 구조에는 DrawDib 작업 집합을 수행하기 위한 경과된 타이밍 정보가 포함됩니다. DrawDibTime 함수는 호출될 때마다 각 작업에 대한 개수 및 경과된 시간 값을 다시 설정합니다.
ICCOMPRESS ICCOMPRESS 구조에는 ICM_COMPRESS 메시지와 함께 사용되는 압축 매개 변수가 포함되어 있습니다.
ICCOMPRESSFRAMES ICCOMPRESSFRAMES 구조체에는 ICM_COMPRESS_FRAMES_INFO 메시지와 함께 사용되는 압축 매개 변수가 포함되어 있습니다.
ICDECOMPRESS ICDECOMPRESS 구조에는 ICM_DECOMPRESS 메시지와 함께 사용되는 압축 해제 매개 변수가 포함되어 있습니다.
ICDECOMPRESSEX ICDECOMPRESSEX 구조체에는 ICM_DECOMPRESSEX 메시지와 함께 사용되는 압축 해제 매개 변수가 포함되어 있습니다.
ICDRAW ICDRAW 구조체에는 비디오 데이터를 화면에 그리기 위한 매개 변수가 포함되어 있습니다. 이 구조체는 ICM_DRAW 메시지와 함께 사용됩니다.
ICDRAWBEGIN ICDRAWBEGIN 구조체에는 ICM_DRAW_BEGIN 메시지와 함께 사용되는 압축 해제 매개 변수가 포함되어 있습니다.
ICDRAWSUGGEST ICDRAWSUGGEST 구조에는 적절한 입력 형식을 제안하기 위해 ICM_DRAW_SUGGESTFORMAT 메시지와 함께 사용되는 압축 매개 변수가 포함되어 있습니다.
ICINFO ICINFO 구조에는 비디오 압축 드라이버에서 제공하는 압축 매개 변수가 포함되어 있습니다. 드라이버는 ICM_GETINFO 메시지를 받을 때 구조를 채우거나 업데이트합니다.
ICOPEN ICOPEN 구조에는 압축 또는 압축 해제되는 데이터 스트림, 드라이버의 버전 번호 및 드라이버 사용 방법에 대한 정보가 포함됩니다.

ICSETSTATUSPROC ICSETSTATUSPROC 구조에는 ICM_SET_STATUS_PROC 메시지와 함께 사용되는 상태 정보가 포함되어 있습니다.
VIDEOHDR VIDEOHDR 구조체는 capVideoStreamCallback 함수에서 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

virtdisk.h 헤더

아티클2023. 08. 27.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 스토리지 드라이버 DDI 참조
- 가상 하드 디스크
- 가상 스토리지

virtdisk.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AddVirtualDiskParent OPEN_VIRTUAL_DISK_FLAG_CUSTOM_DIFF_CHAIN 플래그를 사용하여 열린 가상 디스크에 부모를 연결합니다.
ApplySnapshotVhdSet VHD 집합 파일에 현재 가상 디스크의 스냅샷 적용합니다.
AttachVirtualDisk 첨부 파일을 수행할 적절한 VHD 공급자를 찾아 VHD(가상 하드 디스크) 또는 CD 또는 DVD 이미지 파일(ISO)을 연결합니다.
BreakMirrorVirtualDisk 이전에 시작된 미러 작업을 중단하고 미러 활성 가상 디스크로 설정합니다.
CompactVirtualDisk VHD(가상 하드 디스크) 백업 저장소 파일의 크기를 줄입니다.
CreateVirtualDisk 기본 매개 변수를 사용하거나 기존 가상 디스크 또는 실제 디스크를 사용하여 VHD(가상 하드 디스크) 이미지 파일을 만듭니다.
DeleteSnapshotVhdSet VHD 집합 파일에서 스냅샷 삭제합니다.

DeleteVirtualDiskMetadata 가상 디스크에서 메타데이터를 삭제합니다.
DetachVirtualDisk 작업을 수행하기 위해 적절한 가상 디스크 공급자를 찾아 VHD(가상 하드 디스크) 또는 CD 또는 DVD 이미지 파일(ISO)을 분리합니다.
EnumerateVirtualDiskMetadata 가상 디스크와 연결된 메타데이터를 열거합니다.
ExpandVirtualDisk 고정되거나 동적으로 확장 가능한 VHD(가상 하드 디스크)의 크기를 늘입니다.
GetStorageDependencyInformation VHD(가상 하드 디스크) 또는 CD 또는 DVD 이미지 파일(ISO) 또는 해당 디스크와 부모 디스크 또는 볼륨에 포함된 볼륨 간의 관계를 반환합니다.
GetVirtualDiskInformation VHD에 대한 정보를 검색합니다.
GetVirtualDiskMetadata 가상 디스크에서 지정된 메타데이터를 검색합니다.
GetVirtualDiskOperationProgress 비동기 VHD(가상 하드 디스크) 작업의 진행률을 확인합니다.
GetVirtualDiskPhysicalPath VHD(가상 하드 디스크) 또는 CD 또는 DVD 이미지 파일(ISO)이 포함된 물리적 디바이스 개체의 경로를 검색합니다.
MergeVirtualDisk 차이점 체인의 자식 VHD(가상 하드 디스크)를 체인에 있는 하나 이상의 부모 가상 디스크와 병합합니다.
MirrorVirtualDisk 가상 디스크에 대한 미러 작업을 시작합니다.

ModifyVhdSet 가상 디스크 파일의 내부 콘텐츠를 수정합니다. 활성 리프를 설정하거나 스냅샷 항목을 수정하는 데 사용할 수 있습니다.
OpenVirtualDisk 사용할 VHD(가상 하드 디스크) 또는 CD 또는 DVD 이미지 파일(ISO)을 엽니다.
QueryChangesVirtualDisk RCT(복원력 있는 변경 내용 추적)로 추적되는 VHD(가상 하드 디스크)의 지정된 영역에 대한 변경 내용에 대한 정보를 검색합니다.
RawSCSIVirtualDisk 포함된 SCSI 요청을 가상 하드 디스크에 직접 발급합니다.
ResizeVirtualDisk 가상 디스크의 크기를 조정합니다.
SetVirtualDiskInformation VHD(가상 하드 디스크)에 대한 정보를 설정합니다.
SetVirtualDiskMetadata 가상 디스크에 대한 메타데이터 항목을 설정합니다.
TakeSnapshotVhdSet VHD 집합 파일에 대한 현재 가상 디스크의 스냅샷 만듭니다.

구조체

APPLY_SNAPSHOT_VHDSET_PARAMETERS 적용할 새 스냅샷 대한 정보를 나타내는 스냅샷 매개 변수를 포함합니다.
ATTACH_VIRTUAL_DISK_PARAMETERS VHD(가상 하드 디스크) 연결 요청 매개 변수를 포함합니다.

COMPACT_VIRTUAL_DISK_PARAMETERS VHD(가상 하드 디스크) 압축 매개 변수를 포함합니다.
CREATE_VIRTUAL_DISK_PARAMETERS 새로 만든 가상 디스크에 대한 제어 및 정보를 제공하는 VHD(가상 하드 디스크) 만들기 매개 변수를 포함합니다.
DELETE_SNAPSHOT_VHDSET_PARAMETERS VHD 집합에서 삭제할 스냅샷 지정하는 스냅샷 삭제 매개 변수를 포함합니다.
EXPAND_VIRTUAL_DISK_PARAMETERS 가상 디스크 확장 요청 매개 변수를 포함합니다.
GET_VIRTUAL_DISK_INFO VHD(가상 하드 디스크) 정보를 포함합니다.
MERGE_VIRTUAL_DISK_PARAMETERS VHD(가상 하드 디스크) 병합 요청 매개 변수를 포함합니다.
MIRROR_VIRTUAL_DISK_PARAMETERS 요청 매개 변수를 미러 VHD(가상 하드 디스크)를 포함합니다.
MODIFY_VHDSET_PARAMETERS VHD 집합을 변경하는 방법을 나타내는 VHD 집합 수정 매개 변수를 포함합니다.
OPEN_VIRTUAL_DISK_PARAMETERS 가상 디스크 열기 요청 매개 변수를 포함합니다.
QUERY_CHANGES_VIRTUAL_DISK_RANGE RCT(복원력 있는 변경 내용 추적)에 의해 추적된 대로 변경된 VHD(가상 하드 디스크)의 영역을 식별합니다.
RAW_SCSI_VIRTUAL_DISK_PARAMETERS 원시 SCSI 가상 디스크 요청 매개 변수를 포함합니다.
RAW_SCSI_VIRTUAL_DISK_RESPONSE 원시 SCSI 가상 디스크 응답 매개 변수를 포함합니다.

RESIZE_VIRTUAL_DISK_PARAMETERS ResizeVirtualDisk 함수에 대한 매개 변수를 포함합니다.
SET_VIRTUAL_DISK_INFO SetVirtualDiskInformation 함수를 호출하여 VHD 속성을 설정할 때 사용할 VHD(가상 하드 디스크) 정보를 포함합니다.
STORAGE_DEPENDENCY_INFO VHD(가상 하드 디스크) 스토리지 종속성 정보를 포함합니다.
STORAGE_DEPENDENCY_INFO_TYPE_1 유형 1에 대한 VHD(가상 하드 디스크) 스토리지 종속성 정보를 포함합니다.
STORAGE_DEPENDENCY_INFO_TYPE_2 형식 2에 대한 VHD 또는 ISO 스토리지 종속성 정보를 포함합니다.
TAKE_SNAPSHOT_VHDSET_PARAMETERS 만들 새 스냅샷 대한 정보를 나타내는 스냅샷 매개 변수를 포함합니다.
VIRTUAL_DISK_PROGRESS GetVirtualDiskOperationProgress 함수에서 사용하는 현재 VHD(가상 하드 디스크) 작업에 대한 진행률 및 결과 데이터를 포함합니다.
VIRTUAL_STORAGE_TYPE 가상 스토리지 디바이스의 유형 및 공급자(공급업체)를 포함합니다.

열거형

APPLY_SNAPSHOT_VHDSET_FLAG ApplySnapshotVhdSet 함수의 동작에 영향을 주는 플래그를 포함합니다.
APPLY_SNAPSHOT_VHDSET_VERSION ApplySnapshotVhdSet 함수에 사용할 수 있는 매개 변수 버전을 열거합니다.

ATTACH_VIRTUAL_DISK_FLAG 가상 디스크 연결 요청 플래그를 포함합니다.
ATTACH_VIRTUAL_DISK_VERSION VHD 함수 호출에 사용할 VHD(가상 하드 디스크) ATTACH_VIRTUAL_DISK_PARAMETERS 구조체의 버전을 포함합니다.
COMPACT_VIRTUAL_DISK_FLAG 가상 디스크 압축 요청 플래그를 포함합니다.
COMPACT_VIRTUAL_DISK_VERSION VHD 함수 호출에 사용할 VHD(가상 하드 디스크) COMPACT_VIRTUAL_DISK_PARAMETERS 구조체의 버전을 포함합니다.
CREATE_VIRTUAL_DISK_FLAG VHD(가상 하드 디스크) 만들기 플래그를 포함합니다.
CREATE_VIRTUAL_DISK_VERSION 가상 디스크 함수 호출에 사용할 가상 디스크 CREATE_VIRTUAL_DISK_PARAMETERS 구조의 버전을 포함합니다.
DELETE_SNAPSHOT_VHDSET_FLAG DeleteSnapshotVhdSet 함수의 동작에 영향을 주는 플래그를 포함합니다.
DELETE_SNAPSHOT_VHDSET_VERSION 가상 디스크 함수 호출에 사용할 DELETE_SNAPHSOT_VHDSET_PARAMETERS 구조체의 버전을 포함합니다.
DEPENDENT_DISK_FLAG VHD(가상 하드 디스크) 종속성 정보 플래그를 포함합니다.
DETACH_VIRTUAL_DISK_FLAG 가상 디스크 분리 요청 플래그를 포함합니다.
EXPAND_VIRTUAL_DISK_FLAG VHD(가상 하드 디스크) 확장 요청 플래그를 포함합니다.

EXPAND_VIRTUAL_DISK_VERSION 가상 디스크 함수 호출에 사용할 가상 디스크 EXPAND_VIRTUAL_DISK_PARAMETERS 구조체의 버전을 포함합니다.
GET_STORAGE_DEPENDENCY_FLAG VHD(가상 하드 디스크) 스토리지 종속성 요청 플래그를 포함합니다.
GET_VIRTUAL_DISK_INFO_VERSION 검색할 수 있는 VHD(가상 하드 디스크) 정보의 종류를 포함합니다.
MERGE_VIRTUAL_DISK_FLAG VHD(가상 하드 디스크) 병합 요청 플래그를 포함합니다.
MERGE_VIRTUAL_DISK_VERSION VHD 함수 호출에 사용할 VHD(가상 하드 디스크) MERGE_VIRTUAL_DISK_PARAMETERS 구조체의 버전을 포함합니다.
MIRROR_VIRTUAL_DISK_FLAG VHD(가상 하드 디스크) 미러 요청 플래그를 포함합니다.
MIRROR_VIRTUAL_DISK_VERSION MirrorVirtualDisk 함수에서 사용하는 가상 디스크 MIRROR_VIRTUAL_DISK_PARAMETERS 구조체의 버전을 포함합니다.
MODIFY_VHDSET_FLAG ModifyVhdSet 함수의 동작에 영향을 주는 플래그를 포함합니다.
MODIFY_VHDSET_VERSION 가상 디스크 함수 호출에 사용할 MODIFY_VHDSET_PARAMETERS 구조체의 버전을 포함합니다.
OPEN_VIRTUAL_DISK_FLAG VHD(가상 하드 디스크) 또는 CD 또는 ISO(DVD 이미지 파일) 열린 요청 플래그를 포함합니다.
OPEN_VIRTUAL_DISK_VERSION 가상 디스크 함수 호출에 사용할 가상 디스크 OPEN_VIRTUAL_DISK_PARAMETERS 구조의 버전을 포함합니다.

RAW_SCSI_VIRTUAL_DISK_FLAG RawSCSIVirtualDisk 함수의 동작에 영향을 주는 플래그를 포함합니다.
RAW_SCSI_VIRTUAL_DISK_VERSION 가상 디스크 함수 호출에 사용할 RAW_SCSI_VIRTUAL_DISK_PARAMETERS 구조체의 버전을 포함합니다.
RESIZE_VIRTUAL_DISK_FLAG ResizeVirtualDisk 함수에 사용할 수 있는 플래그를 열거합니다.
RESIZE_VIRTUAL_DISK_VERSION ResizeVirtualDisk 함수에 사용할 수 있는 매개 변수 버전을 열거합니다.
SET_VIRTUAL_DISK_INFO_VERSION VHD 함수 호출에 사용할 가상 디스크 SET_VIRTUAL_DISK_INFO 구조체의 버전을 포함합니다.
STORAGE_DEPENDENCY_INFO_VERSION VHD 함수 호출에 사용할 VHD(가상 하드 디스크) STORAGE_DEPENDENCY_INFO 구조체의 버전을 포함합니다.
TAKE_SNAPSHOT_VHDSET_FLAG TakeSnapshotVhdSet 함수의 동작에 영향을 주는 플래그를 포함합니다.
TAKE_SNAPSHOT_VHDSET_VERSION TakeSnapshotVhdSet 함수에 사용할 수 있는 매개 변수 버전을 열거합니다.
VIRTUAL_DISK_ACCESS_MASK VIRTUAL_DISK_ACCESS_MASK VHD(가상 하드 디스크) 또는 CD 또는 DVD 이미지 파일(ISO)에 대한 액세스 권한을 지정하기 위한 비트 마스크를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

vmgenerationcounter.h 헤더

아티클2023. 08. 27.

이 헤더는 Hyper-V 관리에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Hyper-V 관리

vmgenerationcounter.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

IOCTLs

IOCTL_VMGENCOUNTER_READ IOCTL_VMGENCOUNTER_READ 제어 코드는 가상 머신 생성 식별자를 검색합니다.

구조체

VM_GENCOUNTER 가상 머신 생성 식별자에 대해 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

vsadmin.h 헤더

아티클2023. 08. 27.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

vsadmin.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IVssAdmin IVssAdmin 인터페이스는 VSS에 등록된 공급자를 관리합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

vsbackup.h 헤더

아티클2023. 08. 27.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

vsbackup.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

IVssBackupComponents IVssBackupComponents 인터페이스는 요청자가 파일 상태 대한 작성자를 폴링하고 백업/복원 작업을 실행하는 데 사용됩니다.
IVssBackupComponentsEx 요청자가 여러 기록기 인스턴스를 사용하여 백업 및 복원 작업을 실행할 수 있는 메서드를 제공합니다.
IVssBackupComponentsEx2 요청자가 백업 및 복원 작업을 실행하는 데 사용할 수 있는 추가 메서드를 정의합니다.
IVssBackupComponentsEx3 요청자가 LUN 다시 동기화를 수행하고 확장 기록기 상태 정보를 반환하는 데 사용할 수 있는 추가 메서드를 정의합니다.
IVssBackupComponentsEx4 요청자의 UNC 파일 공유 경로 처리를 지원하는 추가 메서드를 정의합니다.
IVssExamineWriterMetadata IVssExamineWriterMetadata 인터페이스는 요청자가 특정 작성기 instance 메타데이터를 검사할 수 있는 C++(COM 아님) 인터페이스입니다.
IVssExamineWriterMetadataEx 작성기 instance 이름 및 특정 작성기 instance 대한 기타 기본 정보를 검색하는 메서드를 제공합니다.

IVssExamineWriterMetadataEx2 특정 기록기 instance 대한 버전 정보 및 기타 기본 정보를 검색하는 메서드를 정의합니다.
IVssWMComponent IVssWMComponent는 기록기 메타데이터 문서에 저장된 구성 요소 정보에 액세스할 수 있는 C++(COM 아님) 인터페이스입니다.
IVssWriterComponentsExt IVssWriterComponentsExt 인터페이스는 요청자가 백업과 관련된 기록기의 구성 요소에 액세스하고 수정하는 데 사용하는 C++(COM 아님) 인터페이스입니다.

Functions

CreateVssBackupComponents CreateVssBackupComponents 함수(vsbackup.h)는 IVssBackupComponents 인터페이스 개체를 만들고 포인터를 반환합니다.
CreateVssBackupComponentsInternal CreateVssBackupComponents 함수(vsbackup.h)는 IVssBackupComponents 인터페이스 개체를 만들고 포인터를 반환합니다.
CreateVssExamineWriterMetadata CreateVssExamineWriterMetadata 함수(vsbackup.h)는 IVssExamineWriterMetadata 개체를 만듭니다.
CreateVssExamineWriterMetadataInternal CreateVssExamineWriterMetadata 함수(vsbackup.h)는 IVssExamineWriterMetadata 개체를 만듭니다.
IsVolumeSnapshotted IsVolumeSnapshotted 함수(vsbackup.h)는 지정된 볼륨에 대한 섀도 복사본이 있는지 여부를 결정합니다.
IsVolumeSnapshottedInternal

IsVolumeSnapshottedInternal 함수(vsbackup.h)는 지정된 볼륨에 대한 섀도 복사본이 있는지 여부를 결정합니다.
ShouldBlockRevert 지정된 볼륨에서 되돌리기 작업을 차단해야 하는 기록기에 대한 레지스트리를 확인합니다.
VssFreeSnapshotProperties VssFreeSnapshotProperties 함수(vsbackup.h)는 수명 주기 관리의 일환으로 VSS_SNAPSHOT_PROP 구조체의 내용을 해제하는 데 사용됩니다.
VssFreeSnapshotPropertiesInternal VssFreeSnapshotPropertiesInternal 함수(vsbackup.h)는 수명 주기 관리의 일부로 VSS_SNAPSHOT_PROP 구조체의 내용을 해제하는 데 사용됩니다.

구조체

VSS_COMPONENTINFO 지정된 구성 요소에 대한 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

vsmgmt.h 헤더

아티클2023. 08. 27.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

vsmgmt.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IVssDifferentialSoftwareSnapshotMgmt 애플리케이션이 시스템 섀도 복사본 공급자에 의해 생성된 섀도 복사본 스토리지 영역을 쿼리하고 관리할 수 있도록 하는 메서드를 포함합니다.
IVssDifferentialSoftwareSnapshotMgmt2 애플리케이션이 시스템 섀도 복사본 공급자에 의해 생성된 섀도 복사본 스토리지 영역을 쿼리하고 관리할 수 있도록 하는 추가 메서드를 정의합니다.
IVssDifferentialSoftwareSnapshotMgmt3 애플리케이션이 VSS의 섀도 복사본 보호 기능을 사용할 수 있도록 하는 메서드를 정의합니다.
IVssEnumMgmtObject 열거된 개체 목록에서 반복하고 다른 작업을 수행하는 메서드를 포함합니다. (IVssEnumMgmtObject)
IVssSnapshotMgmt 인터페이스를 반환하여 섀도 복사본 공급자를 추가로 구성하는 메서드를 제공합니다.
IVssSnapshotMgmt2 섀도 복사본 스토리지 영역의 최소 크기를 검색하는 메서드를 제공합니다.

구조체

VSS_DIFF_AREA_PROP 원래 파일 데이터를 포함하는 볼륨과 섀도 복사본 스토리지 영역을 포함하는 볼륨 간의 연결을 설명합니다.
VSS_DIFF_VOLUME_PROP 섀도 복사본 스토리지 영역 볼륨에 대해 설명합니다.
VSS_MGMT_OBJECT_PROP 볼륨, 섀도 복사본 스토리지 볼륨 또는 섀도 복사본 스토리지 영역의 속성을 정의합니다.
VSS_VOLUME_PROP 섀도 복사본 원본 볼륨의 속성을 포함합니다.
VSS_VOLUME_PROTECTION_INFO 볼륨의 섀도 복사본 보호 수준에 대한 정보를 포함합니다.

열거형

VSS_MGMT_OBJECT_TYPE VSS_MGMT_OBJECT_PROP 구조 내에서 VSS_MGMT_OBJECT_UNION 공용 구조체에 대한 비난.
VSS_PROTECTION_FAULT 섀도 복사본 보호 오류 집합을 정의합니다.
VSS_PROTECTION_LEVEL 볼륨 섀도 복사본 보호 수준 집합을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

vsprov.h 헤더

아티클2023. 08. 27.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

vsprov.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IVssFileShareSnapshotProvider . (IVssFileShareSnapshotProvider)
IVssHardwareSnapshotProvider VSS에서 볼륨을 LUN에 매핑하고, 섀도 복사 프로세스 중에 생성된 LUN을 검색하고, SAN에서 LUN을 전송하는 데 사용하는 메서드를 포함합니다.
IVssHardwareSnapshotProviderEx VSS에서 하드웨어 공급자에게 LUN 상태 변경 내용을 알리는 데 사용하는 추가 메서드를 제공합니다.
IVssProviderCreateSnapshotSet IVssProviderCreateSnapshotSet 인터페이스에는 섀도 복사본을 만드는 동안 사용되는 메서드가 포함되어 있습니다.
IVssProviderNotifications IVssProviderNotifications 인터페이스는 VSS에 등록된 공급자를 관리합니다.
IVssSoftwareSnapshotProvider VSS에서 섀도 복사본 볼륨을 관리하는 데 사용하는 메서드를 포함합니다. 모든 소프트웨어 공급자는 이 인터페이스를 지원해야 합니다.

피드백

이 페이지가 도움이 되었나요?

Yes No

Microsoft Q&A에서 도움말 보기

vss.h 헤더

아티클2023. 08. 27.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

vss.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IVssAsync IVssAsync 인터페이스는 비동기 작업을 시작하는 메서드를 통해 호출 애플리케이션에 반환됩니다. 이 메서드는 백그라운드에서 실행되며 일반적으로 완료하는 데 시간이 오래 걸립니다.
IVssEnumObject 열거된 개체 목록에서 반복하고 다른 작업을 수행하는 메서드를 포함합니다. (IVssEnumObject)

구조체

VSS_OBJECT_PROP 공급자, 볼륨, 섀도 복사본 또는 섀도 복사본 집합의 속성을 정의합니다.
VSS_PROVIDER_PROP 섀도 복사본 공급자 속성을 지정합니다.
VSS_SNAPSHOT_PROP 섀도 복사본 또는 섀도 복사본 집합의 속성을 포함합니다.

열거형

VSS_APPLICATION_LEVEL 애플리케이션 수준, 기록기에 동결 알림을 받은 섀도 복사본을 만드는 과정에서의 지점을 나타냅니다.
VSS_BACKUP_SCHEMA 작성기에서 참여할 수 있는 백업 작업 유형을 나타내는 데 사용됩니다.
VSS_BACKUP_TYPE 수행할 백업 유형을 나타냅니다.
VSS_FILE_SPEC_BACKUP_TYPE 기록기에서 특정 백업 작업에 대한 지원을 나타내는 데 사용됩니다.
VSS_HARDWARE_OPTIONS 섀도 복사본 LUN 플래그를 정의합니다.
VSS_OBJECT_TYPE 요청자가 개체를 섀도 복사본 집합, 섀도 복사본 또는 공급자로 식별하는 데 사용됩니다.
VSS_PROVIDER_CAPABILITIES 이 열거형은 나중에 사용할 수 있습니다.
VSS_PROVIDER_TYPE 공급자 유형을 지정합니다.
VSS_RECOVERY_OPTIONS 요청자가 다시 동기화 작업을 수행하는 방법을 지정하는 데 사용됩니다.
VSS_RESTORE_TYPE 요청자가 수행하려는 복원 작업의 유형을 나타내는 데 사용됩니다.
VSS_ROLLFORWARD_TYPE VSS_ROLLFORWARD_TYPE 열거형은 요청자가 수행하려는 롤 포워드 작업의 유형을 나타내는 데 사용됩니다.
VSS_SNAPSHOT_COMPATIBILITY

VSS_SNAPSHOT_COMPATIBILITY 열거형은 섀도 복사된 볼륨에 대해 사용하지 않도록 설정된 볼륨 제어 또는 파일 I/O 작업을 나타냅니다.
VSS_SNAPSHOT_CONTEXT 섀도 복사본을 만들거나 쿼리하거나 삭제하는 방법과 기록기 참여 정도를 지정합니다.
VSS_SNAPSHOT_PROPERTY_ID 섀도 복사본에 설정할 속성을 지정합니다.
VSS_SNAPSHOT_STATE 지정된 섀도 복사 작업의 상태를 지정합니다.
VSS_VOLUME_SNAPSHOT_ATTRIBUTES 섀도 복사본에 대해 추가 특성을 지정할 수 있습니다.
VSS_WRITER_STATE 작성기의 현재 상태를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

vswriter.h 헤더

아티클2023. 08. 27.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

vswriter.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

CVssWriter CVssWriter 클래스는 작성기가 VSS 및 기타 작성기와 상태를 동기화하는 인터페이스를 정의하는 추상 기본 클래스입니다.
CVssWriterEx CVssWriterEx 클래스는 작성기가 VSS 및 기타 작성기와 상태를 동기화하는 인터페이스를 정의하는 추상 기본 클래스입니다.
CVssWriterEx2 CVssWriterEx2 클래스는 작성기가 VSS 및 기타 작성기와 상태를 동기화하는 인터페이스를 정의하는 추상 기본 클래스입니다.
IVssComponent IVssComponent 인터페이스는 요청자의 백업 구성 요소 문서에 포함된 구성 요소에 대한 정보를 검사하고 수정하는 메서드를 포함하는 C++(COM 아님) 인터페이스입니다.
IVssComponentEx 요청자의 백업 구성 요소 문서에 포함된 구성 요소에 대한 정보를 검사하고 수정하기 위한 추가 메서드를 정의합니다.
IVssComponentEx2 구성 요소 수준 기록기 오류를 보고하고 검색하기 위한 추가 메서드를 정의합니다.
IVssCreateExpressWriterMetadata IVssCreateExpressWriterMetadata 인터페이스는 빠른 작성기에 대한 기록기 메타데이터 문서를 생성하는 메서드를 포함하는 COM 인터페이스입니다.

IVssCreateWriterMetadata IVssCreateWriterMetadata 인터페이스는 Identify 이벤트에 대한 응답으로 기록기 메타데이터 문서를 생성하는 메서드를 포함하는 C++(COM 아님) 인터페이스입니다. CVssWriter::OnIdentify 메서드에서만 사용됩니다.
IVssCreateWriterMetadataEx IVssCreateWriterMetadataEx 인터페이스는 섀도 복사본을 만들 때 명시적으로 제외될 파일 집합을 보고하는 메서드를 정의하는 C++(COM 아님) 인터페이스입니다.
IVssExpressWriter VSS Express Writer에 대한 메타데이터를 관리하는 메서드를 정의합니다.
IVssWMDependency IVssWMDependency는 IVssWMComponent 인터페이스에서 반환되고 다른 작성기에서 관리하는 구성 요소에 대한 명시적 기록기 구성 요소 종속성이 있는 구성 요소를 백업하거나 복원할 때 애플리케이션에서 사용하는 C++(COM 아님) 인터페이스입니다.
IVssWMFiledesc IVssWMFiledesc 인터페이스는 여러 쿼리 메서드에 의해 호출 애플리케이션에 반환된 C++(COM 아님) 인터페이스입니다. 파일 또는 파일 집합(파일 집합)에 대한 자세한 정보를 제공합니다.
IVssWriterComponents 구성 요소 정보를 가져오고 수정하는 데 사용되는 메서드를 포함합니다.

Functions

CreateVssExpressWriter CreateVssExpressWriter 함수(vswriter.h)는 IVssExpressWriter 인터페이스 개체를 만들고 포인터를 반환합니다.
CreateVssExpressWriterInternal CreateVssExpressWriterInternal 함수(vswriter.h)는 IVssExpressWriter 인터페이스 개체를 만들고 포인터를 반환합니다.
CreateWriter

이 함수는 시스템 사용을 위해 예약되어 있습니다. (CreateWriter)
CreateWriterEx 이 함수는 시스템 사용을 위해 예약되어 있습니다. (CreateWriterEx)

열거형

VSS_ALTERNATE_WRITER_STATE 지정된 작성기에 연결된 대체 작성기가 있는지 여부를 나타내는 데 사용됩니다.
VSS_COMPONENT_FLAGS 기록기에서 자동 복구에 대한 지원을 나타내는 데 사용됩니다.
VSS_COMPONENT_TYPE 섀도 복사 백업 작업과 함께 사용되는 구성 요소의 유형을 지정합니다.
VSS_FILE_RESTORE_STATUS 파일 복원 작업의 상태 집합을 정의합니다.
VSS_RESTORE_TARGET 복원 시 작성기에서 선택한 구성 요소에 포함된 모든 파일과 정의한 구성 요소 집합의 모든 파일을 복원하는 방법을 나타내는 데 사용됩니다.
VSS_RESTOREMETHOD_ENUM 기록기에서 백업할 때 기록기 메타데이터 문서를 통해 기본 파일 복원 방법을 지정하는 데 사용됩니다.
VSS_SOURCE_TYPE 기록기가 관리하는 데이터 형식을 지정합니다.
VSS_SUBSCRIBE_MASK 작성기에서 수신할 이벤트를 나타냅니다.
VSS_USAGE_TYPE

호스트 시스템에서 VSS 작업과 관련된 기록기에서 관리하는 데이터를 사용하는 방법을 지정합니다.
VSS_WRITERRESTORE_ENUM 복원 작업 중에 생성된 이벤트를 처리할 조건을 요청자에게 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

waasapi.h 헤더

아티클2023. 08. 27.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

waasapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWaaSAssessor Microsoft의 최신 빌드와 현재 디바이스에서 실행 중인 빌드를 비교하여 OS 업데이트 평가를 가져옵니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

waasapitypes.h 헤더

아티클2023. 08. 27.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

waasapitypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

OSUpdateAssessment OSUpdateAssessment 구조는 대상 디바이스의 OS가 얼마나 최신 상태인지 정의합니다.
UpdateAssessment UpdateAssessment에는 설치된 OS의 최신 정보를 평가하는 정보가 포함되어 있습니다.

열거형

UpdateAssessmentStatus 디바이스의 OS가 얼마나 최신 상태인지 설명합니다.
UpdateImpactLevel 오래된 OS를 실행하는 디바이스의 높음, 중간 또는 낮은 영향을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wabapi.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows 주소록에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 주소록

wabapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWABExtInit 사용하지 마십시오. 이 인터페이스는 표시되는 WAB(Windows Address Book) 개체(예: 속성 시트 또는 상황에 맞는 메뉴)를 ndicate합니다.
IWABObject 사용하지 마십시오. 이 인터페이스는 메모리 할당 함수 및 데이터베이스 유지 관리 함수에 대한 함수 포인터를 포함하는 WAB(Windows Address Book) 개체에 대한 액세스를 제공합니다.

콜백 함수

WABOpen 사용하지 마십시오. 여러 개체 인터페이스를 통해 주소록에 대한 액세스를 제공합니다. 루트 인터페이스는 IAddrBook의 MAPI 구현 하위 집합인 IAddrBook입니다.
WABOpenEx WABOpenEx는 Windows Vista를 기준으로 더 이상 사용할 수 없습니다.

구조체

WAB_PARAM 사용하지 마십시오. WABOpen에 전달할 입력 정보를 포함합니다.

WABEXTDISPLAY 사용하지 마십시오. WAB(Windows 주소록)에서 사용자의 IContextMenu 인터페이스 및 IShellPropSheetExt 인터페이스 구현을 초기화하는 데 사용됩니다.
WABIMPORTPARAM 사용하지 마십시오. .wab 파일 가져오기에 대한 정보를 제공하는 가져오기에 전달된 구조체입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wabdefs.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows 주소록에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 주소록

wabdefs.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IABContainer 사용하지 마십시오. 이 인터페이스는 주소록 컨테이너에 대한 액세스를 제공합니다. 애플리케이션은 인터페이스의 메서드를 호출하여 이름 확인을 수행하고 받는 사람을 만들고, 복사하고, 삭제합니다.
IDistList 사용하지 마십시오. 이 인터페이스는 수정 가능한 주소록 컨테이너의 배포 목록에 대한 액세스를 제공하는 데 사용됩니다. 인터페이스는 이름 확인을 수행하는 것 외에도 메일 그룹을 만들고, 복사하고, 삭제하는 메서드를 제공합니다.
IMailUser 사용하지 마십시오. 이 인터페이스는 메일 사용자 개체에 대한 액세스를 제공합니다.
IMAPITable 사용하지 마십시오. 이 인터페이스는 WAB(Windows 주소록) 컨테이너 및 배포 목록의 콘텐츠 테이블에 사용됩니다.

구조체

SRestriction 사용하지 마십시오. 테이블 보기를 특정 행으로 제한하기 위한 필터에 대해 설명합니다.
ADRENTRY

ADRENTRY(wabdefs.h) - 사용하지 마세요. 하나 이상의 받는 사람에 속하는 0개 이상의 속성을 설명합니다.
ADRLIST ADRLIST(wabdefs.h) - 사용하지 마세요. 하나 이상의 받는 사람에 속하는 0개 이상의 속성을 설명합니다. G
ADRPARM 사용하지 마십시오. 일반 주소 대화 상자의 표시 및 동작에 대해 설명합니다.
Entryid 사용하지 마십시오. MAPI 개체에 대한 항목 식별자 정보를 포함합니다.
SBinaryArray 사용하지 마십시오. MAPI 개체를 나타내는 항목 식별자의 배열입니다. SBinaryArray와 동일한 구현을 사용합니다.
SPropProblem 사용하지 마십시오. 속성과 관련된 작업과 관련된 오류를 설명합니다.
SPropProblemArray 사용하지 마십시오. 하나 이상의 SPropProblem 구조체 배열을 포함합니다.
SPropTagArray 사용하지 마십시오. 속성 태그의 배열을 포함합니다.
SPropValue 사용하지 마십시오. 속성 태그 값을 포함합니다.
슬로우 (동음이의) 사용하지 마십시오. 특정 개체에 대해 선택한 속성을 포함하는 테이블의 행에 대해 설명합니다.
SRowSet 사용하지 마십시오. SRow 구조체의 배열을 포함합니다. 각 SRow 구조는 테이블의 행을 설명합니다.
SSortOrder

사용하지 마십시오. 정렬 키로 사용할 열과 정렬 방향을 모두 설명하는 테이블 행을 정렬하는 방법을 정의합니다.
SSortOrderSet 사용하지 마십시오. 표준 또는 분류된 정렬에 사용할 테이블의 키 컬렉션을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wabiab.h 헤더

아티클2023. 03. 08.

이 헤더는 Windows 주소록에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 주소록

wabiab.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAddrBook 사용하지 마십시오. (IAddrBook)

wabmem.h 헤더

아티클2023. 08. 27.

이 헤더는 MAPI(Windows 메시징)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- MAPI(Windows 메시징)

wabmem.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

콜백 함수

MAPIFREEBUFFER MAPIFreeBuffer 함수(wabmem.h)는 메시징 시스템에서 할당한 메모리를 해제합니다. (MAPIFreeBuffer)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wabtags.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows 주소록에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 주소록

wabtags.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

열거형

Gender 사용하지 마십시오. Gender 열거형은 PR_GENDER 속성에 사용할 수 있는 값을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wbemcli.h 헤더

아티클2023. 08. 27.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 분산 트랜잭션 코디네이터 WMI 공급자
- 원격 데스크톱 서비스
- Windows 데이터 액세스 구성 요소 WMI 공급자
- Windows Management Instrumentation
- Windows 사이드바

wbemcli.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IEnumWbemClassObject IEnumWbemClassObject 인터페이스는 CIM(Common Information Model) 개체를 열거하는 데 사용되며 표준 COM 열거자와 비슷합니다.
IMofCompiler Mofd.dll 구현된 IMofCompiler 인터페이스는 MOF(Managed Object Format) 컴파일러 및 MOF 파일을 컴파일하는 다른 애플리케이션에서 사용하는 COM 인터페이스를 제공합니다.
IUnsecuredApartment IUnsecuredApartment 인터페이스는 클라이언트 프로세스에서 비동기 호출을 만드는 프로세스를 간소화하는 데 사용됩니다.
IWbemBackupRestore IWbemBackupRestore 인터페이스는 WMI 리포지토리의 콘텐츠를 백업하고 복원합니다.
IWbemBackupRestoreEx IWbemBackupRestoreEx 인터페이스는 리포지토리의 콘텐츠를 백업하고 복원합니다.
IWbemCallResult IWbemServices 인터페이스의 semisynchronous 호출에 사용합니다. 이러한 호출을 할 때, 호출된 IWbemServices 메서드는 IWbemCallResult 개체와 함께 즉시 반환됩니다.

IWbemClassObject 클래스 정의와 클래스 개체 인스턴스를 모두 포함하고 조작합니다.
IWbemConfigureRefresher IWbemConfigureRefresher 인터페이스는 클라이언트 코드에서 열거자, 개체 및 중첩된 새로 고침을 새로 고침에 추가하는 데 사용됩니다.
IWbemContext IWbemContext 인터페이스는 필요에 따라 WMI에 IWbemServices 호출을 제출할 때 공급자에게 추가 컨텍스트 정보를 전달하는 데 사용됩니다. IWbemServices의 모든 기본 호출은 이 형식의 개체를 가리키는 선택적 매개 변수를 사용합니다.
IWbemHiPerfEnum 새로 고침 작업에서 인스턴스 개체의 열거형에 대한 신속한 액세스를 제공하는 용도로 사용됩니다.
IWbemLocator IWbemLocator 인터페이스를 사용하여 특정 호스트 컴퓨터에서 WMI용 IWbemServices 인터페이스에 대한 초기 네임스페이스 포인터를 가져옵니다.
IWbemObjectAccess 개체의 메서드 및 속성에 대한 액세스를 제공합니다.
IWbemObjectSink IWbemObjectSink 인터페이스는 WMI 프로그래밍 모델 내의 모든 유형의 알림을 수신할 수 있는 싱크 인터페이스를 만듭니다.
IWbemObjectSinkEx WMI 프로그래밍 모델 내에서 모든 유형의 알림을 받을 수 있는 싱크 인터페이스를 만듭니다.
IWbemObjectTextSrc IWbemObjectTextSrc 인터페이스는 IWbemClassObject 인스턴스를 다른 텍스트 형식으로 변환하는 데 사용됩니다.
IWbemQualifierSet 단일 속성 또는 전체 개체(클래스 또는 인스턴스)에 대한 명명된 한정자의 전체 집합용 컨테이너 역할을 합니다.
IWbemRefresher

열거자 또는 리프레셔 개체 같은 새로 고칠 수 있는 개체를 새로 고치는 진입점을 제공합니다.
IWbemServices 클라이언트 및 공급자가 WMI 서비스에 액세스하는 용도로 사용합니다. 인터페이스는 WMI 및 WMI 공급자에 의해 구현되며 기본 WMI 인터페이스입니다.
IWbemShutdown IWbemShutdown 인터페이스는 개체의 instance 삭제할 준비가 되었음을 공급자에게 나타냅니다. 공급자는 이 호출을 사용하여 현재 참조하는 리소스를 해제할 수 있습니다.
IWbemStatusCodeText IWbemStatusCodeText 인터페이스는 오류 코드 또는 오류가 발생한 하위 시스템의 이름에 대한 텍스트 문자열 설명을 추출합니다.
IWbemUnsecuredApartment 클라이언트 애플리케이션에서 Unsecapp.exe 비동기 콜백에 대한 액세스 검사를 수행하는지 여부를 확인할 수 있습니다.

구조체

WBEM_COMPILE_STATUS_INFO IMofCompiler 인터페이스의 오류를 설명합니다.

열거형

CIMTYPE_ENUMERATION 여러 다른 CIM 데이터 형식을 지정하는 값을 정의합니다.
WBEM_BACKUP_RESTORE_FLAGS IWbemBackupRestore::Restore 메서드 및 IWbemBackupRestoreEx::Restore 메서드에 사용되는 플래그를 포함합니다.
WBEM_CHANGE_FLAG_TYPE

메서드 매개 변수 플래그를 포함합니다.
WBEM_COMPARISON_FLAG IWbemClassObject::CompareTo 메서드를 사용할 때 수행할 비교를 정의하는 플래그를 포함합니다.
WBEM_COMPILER_OPTIONS IMofCompiler 메서드에 대한 옵션 플래그를 포함합니다.
WBEM_CONDITION_FLAG_TYPE IWbemClassObject::GetNames 메서드와 함께 사용되는 플래그를 포함합니다.
WBEM_CONNECT_OPTIONS IWbemLocator::ConnectServer 메서드에 대한 플래그를 포함합니다.
WBEM_FLAVOR_TYPE 한정자 버전을 나열합니다.
WBEM_GENERIC_FLAG_TYPE 범용 메서드 매개 변수 플래그를 포함합니다.
WBEM_GENUS_TYPE 클래스와 인스턴스를 구분하는 데 사용되는 상수를 포함합니다.
WBEM_LIMITS 일부 제한 값을 정의합니다.
WBEM_QUERY_FLAG_TYPE 쿼리 또는 열거자를 정의하는 데 사용되는 플래그를 포함합니다.
WBEM_REFRESHER_FLAGS refresher 메서드의 동작을 수정하는 플래그를 포함합니다.
WBEM_SECURITY_FLAGS 보안 액세스 수준을 설정하는 데 사용되는 플래그를 포함합니다.
WBEM_STATUS_TYPE 비동기 작업의 상태 설명합니다.

WBEM_TEXT_FLAG_TYPE IWbemClassObject::GetObjectText 메서드의 실행을 제어하는 플래그를 포함합니다.
WBEM_TIMEOUT_TYPE IEnumWbemClassObject::Next 메서드의 시간 제한을 지정하는 데 사용되는 값을 포함합니다.
WBEM_UNSECAPP_FLAG_TYPE IWbemUnsecuredApartment::CreateSinkStub 메서드를 사용할 때 콜백에 대한 액세스 검사를 제어하는 데 사용됩니다.
WBEMSTATUS WMI API의 메서드에서 반환된 오류 및 상태 코드를 포함합니다.
WMI_OBJ_TEXT SWbemObjectEx.GetText_ 사용할 유효한 개체 텍스트 형식을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wbemdisp.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows Management Instrumentation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Management Instrumentation

wbemdisp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

열거형

WbemAuthenticationLevelEnum 보안 인증 수준을 정의합니다.
WbemChangeFlagEnum 클래스 또는 인스턴스에 대한 쓰기 작업을 수행하는 방법을 정의합니다.
WbemCimtypeEnum 속성 값의 유효한 CIM 형식을 정의합니다.
WbemComparisonFlagEnum 개체 비교를 위해 SWbemObject.CompareTo_에서 사용하는 설정을 정의합니다.
WbemConnectOptionsEnum 원격 컴퓨터에서 WMI에 대한 연결이 실패할 때 SWbemLocator.ConnectServer 메서드 호출에서 매개 변수로 사용되는 보안 플래그를 정의합니다.
WbemErrorEnum WMI용 스크립팅 API 호출에서 반환할 수 있는 오류를 정의합니다.
WbemFlagEnum SWbemServices.ExecQuery, SWbemServices.ExecQueryAsync, SWbemServices.SubclassesOf 및 SWbemServices.InstancesOf에서 사용되는 상수를 정의합니다.
WbemImpersonationLevelEnum 보안 가장 수준을 정의합니다. 이러한 상수는 SWbemSecurity와 함께 사용됩니다.

WbemObjectTextFormatEnum SWbemObjectEx.GetText_에서 사용할 유효한 개체 텍스트 형식을 정의합니다.
WbemPrivilegeEnum 권한을 정의합니다. 이러한 상수는 SWbemSecurity와 함께 사용되어 일부 작업에 필요한 권한을 부여합니다.
WbemQueryFlagEnum 호출에서 반환되는 개체 수를 결정하는 열거형 또는 쿼리의 깊이를 정의합니다.
WbemTextFlagEnum 생성된 개체 텍스트의 내용을 정의하며 SWbemObject.GetObjectText_에서 사용됩니다.
WbemTimeout 제한 시간 상수를 정의합니다. 이 상수는 SWbemEventSource.NextEvent에서 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wbemglue.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows Management Instrumentation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Management Instrumentation

wbemglue.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

CWbemGlueFactory CWbemGlueFactory 클래스는 WMI 공급자 프레임워크의 일부입니다. 공급자 프레임워크는 이 인터페이스의 메서드를 내부에 구현하여 공급자에 대한 클래스의 새 인스턴스를 만듭니다.
CWbemProviderGlue CWbemProviderGlue는 WMI(Windows Management Instrumentation) API의 COM(구성 요소 개체 모델) 인터페이스를 Provider 클래스에서 파생된 클래스와 연결하고 공급자가 서로 쿼리하는 데 사용할 메서드를 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wbemprov.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows Management Instrumentation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Management Instrumentation

wbemprov.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWbemDecoupledBasicEventProvider IWbemDecoupledBasicEventProvider 인터페이스는 분리된 공급자를 WMI에 등록하는 공동 생성 가능한 인터페이스입니다. 만든 개체는 IWbemDecoupledRegistrar::Register의 pUnknown 인수에 전달되어야 합니다.
IWbemDecoupledRegistrar IWbemDecoupledRegistrar 인터페이스는 분리된 공급자를 WMI와 연결합니다. 이 인터페이스를 사용하면 프로세스 호스팅 공급자가 인터페이스의 상호 운용성 수명을 정의하고 다른 공급자와 공존할 수 있습니다.
IWbemEventConsumerProvider 이벤트 소비자 공급자용 기본 인터페이스를 제공합니다. 이 인터페이스와 FindConsumer 메서드를 사용하면, 이벤트 소비자 공급자는 지정된 이벤트를 수신해야 하는 이벤트 소비자를 표시할 수 있습니다.
IWbemEventProvider IWbemEventProvider 인터페이스를 사용하여 이벤트 공급자와의 통신을 시작합니다. Windows Management는 소비자가 공급자가 생성한 이벤트를 수신하는 데 관심을 표시한 경우 이 인터페이스의 공급자 구현을 호출합니다.
IWbemEventProviderQuerySink IWbemEventProviderQuerySink 인터페이스는 성능을 최적화하기 위해 현재 활성 상태인 이벤트 쿼리 필터 종류를 알고자 하는 이벤트 공급자에 의해 선택적으로 구현됩니다.
IWbemEventProviderSecurity IWbemEventProviderSecurity 인터페이스는 필요에 따라 해당 이벤트에 대한 소비자 액세스를 제한

하려는 이벤트 공급자에 의해 구현됩니다. 이 인터페이스를 사용하는 시기에 대한 자세한 내용은 WMI 이벤트 보안을 참조하세요.
IWbemEventSink 제한된 쿼리 집합을 사용하여 이벤트 공급자와의 통신을 시작합니다.
IWbemHiPerfProvider 새로 고칠 수 있는 개체 및 열거자를 공급자가 제공할 수 있게 합니다.
IWbemPropertyProvider WMI 클래스의 인스턴스에서 개별 속성을 검색하고 업데이트할 수 있게 합니다.
IWbemProviderIdentity 공급자가 동일한 CLSID 값으로 둘 이상의 이름(__Win32Provider 여러 인스턴스)을 사용하여 자신을 등록하는 경우 IWbemProviderIdentity 인터페이스는 이벤트 공급자에 의해 구현됩니다.
IWbemProviderInit IWbemProviderInit 인터페이스는 Windows Management에서 공급자를 초기화하기 위해 호출됩니다. IWbemProviderInit를 구현하려면 모든 공급자가 필요합니다.
IWbemProviderInitSink IWbemProviderInitSink 인터페이스는 WMI에서 구현되며 공급자가 초기화 상태 보고하도록 호출합니다.
IWbemUnboundObjectSink IWbemUnboundObjectSink 인터페이스는 모든 논리 이벤트 소비자가 구현합니다. 이벤트 개체 전달을 허용하는 단순 싱크 인터페이스입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wbemtime.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows Management Instrumentation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Management Instrumentation

wbemtime.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

WBEMTime WBEMTime 클래스는 다양한 Windows 및 ANSI C 런타임 형식 간의 변환을 용이하게 합니다. 자세한 내용은 WBEMTimeSpan 클래스 메서드도 참조하세요.
WBEMTimeSpan WBEMTimeSpan 클래스는 시간 범위를 나노초 단위로 보유합니다.

wcmapi.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows 연결 관리자 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 연결 관리자

wcmapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WcmFreeMemory WCM 함수에서 할당한 메모리 리소스를 해제하는 데 사용됩니다.
WcmGetProfileList 기본 설정 순서로 프로필 목록을 검색합니다.
WcmQueryProperty 지정된 WCM 속성의 값을 검색합니다.
WcmSetProfileList 프로필 목록 또는 프로필 목록의 하위 집합을 다시 정렬합니다.
WcmSetProperty WCM 속성의 값을 설정합니다.

구조체

WCM_BILLING_CYCLE_INFO 청구 주기 대한 정보를 지정합니다.
WCM_CONNECTION_COST_DATA 연결 비용에 대한 정보를 지정합니다.

WCM_DATAPLAN_STATUS 네트워크 연결에 대한 구독 정보를 지정합니다.
WCM_POLICY_VALUE 정책의 현재 값에 대한 정보를 포함합니다.
WCM_PROFILE_INFO 특정 프로필에 대한 정보를 포함합니다.
WCM_PROFILE_INFO_LIST 기본 설정 순서로 프로필 목록을 포함합니다.
WCM_TIME_INTERVAL 시간 간격을 정의합니다.
WCM_USAGE_DATA 연결 사용과 관련된 정보를 포함합니다.

열거형

WCM_CONNECTION_COST 연결 비용 유형 및 플래그를 결정합니다.
WCM_CONNECTION_COST_SOURCE 연결 비용 정보를 제공하는 원본을 지정합니다.
WCM_MEDIA_TYPE 연결에 대한 미디어 유형을 지정합니다.
WCM_PROPERTY 연결의 속성을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wcmconfig.h 헤더

아티클2023. 08. 27.

이 헤더는 SMI(설정 관리 인프라)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- SMI(설정 관리 인프라)

wcmconfig.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IItemEnumerator 설정 및 특성 컬렉션의 항목을 열거합니다.
ISettingsContext 다른 SMI API를 통해 변경된 설정을 저장하는 데 사용되는 백업 저장소에 대한 인터페이스이며 표현에서 직렬화 및 역직렬화하는 작업을 제공합니다.
ISettingsEngine 네임스페이스를 열고 열 방법을 제어하기 위한 중앙 인터페이스입니다.
ISettingsIdentity 열거나 사용할 네임스페이스를 식별합니다.
ISettingsItem 설정 트리를 탐색하고, 특정 설정에 대한 메타데이터를 검색하고, 해당 값을 검색하거나 수정합니다.
ISettingsNamespace 네임스페이스 instance 설정을 설정하고, 검색하고, 유효성을 검사하고, 변경 내용을 저장하는 작업을 수행합니다.
ISettingsResult 다양한 작업에서 반환된 오류 및 경고에 대한 코드와 설명을 검색합니다.
ITargetInfo 오프라인 대상 정보, 특히 파일 및 레지스트리 위치와 wow64 정보를 정의합니다.

열거형

WcmDataType ISettingsItem::GetDataType 메서드에서 반환된 데이터 형식을 열거합니다.
WcmNamespaceAccess 네임스페이스에 액세스하는 방법을 선택하기 위해 ISettingsEngine::GetNamespace 메서드에 전달된 옵션에 대해 설명합니다.
WcmNamespaceEnumerationFlags 열거형 플래그의 형식을 설명합니다.
WcmRestrictionFacets ISettingsItem::GetRestrictionFacets 메서드에서 반환할 수 있는 패싯 값을 열거합니다.
WcmSettingType ISettingsItem::GetSettingType 메서드에서 반환되는 설정 형식을 설명하고 호출하는 ISettingsItem 인터페이스에 대한 개체 모델 형식을 정의합니다.
WcmTargetMode 다양한 대상 모드를 열거합니다.
WcmUserStatus 사용자의 상태 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wcndevice.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows Connect Now 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Connect Now

wcndevice.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWCNConnectNotify Windows Connect Now 연결 세션이 완료되면 이 인터페이스를 사용하여 성공 또는 실패 알림을 받습니다.
IWCNDevice 이 인터페이스를 사용하여 디바이스를 구성하고 세션을 시작합니다.

구조체

WCN_VENDOR_EXTENSION_SPEC WCN_VENDOR_EXTENSION_SPEC 구조에는 공급업체 확장을 정의하는 데이터가 포함되어 있습니다.

열거형

WCN_PASSWORD_TYPE WCN_PASSWORD_TYPE 열거형은 WPS 세션에서 사용할 인증을 정의합니다.
WCN_SESSION_STATUS WPS 세션의 결과 상태를 정의합니다.

wcntypes.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows Connect Now 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Connect Now

wcntypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

WCN_VALUE_TYPE_PRIMARY_DEVICE_TYPE WCN_VALUE_TYPE_PRIMARY_DEVICE_TYPE 구조에는 범주, 하위 범주 및 제조업체별 OUI(조직 ID)에 따라 디바이스 유형을 식별하는 정보가 포함됩니다.

열거형

WCN_ATTRIBUTE_TYPE WCN_ATTRIBUTE_TYPE 열거형은 Wi-Fi 보호된 설치에 대해 정의된 특성 버퍼 형식을 정의합니다. 각 특성 버퍼가 차지하는 전체 크기에는 추가 4바이트(ID 2바이트, 길이 2바이트)가 포함됩니다.
WCN_VALUE_TYPE_ASSOCIATION_STATE WCN_VALUE_TYPE_ASSOCIATION_STATE 열거형은 검색 요청 중에 무선 스테이션의 가능한 연결 상태를 정의합니다.
WCN_VALUE_TYPE_AUTHENTICATION_TYPE WCN_VALUE_TYPE_AUTHENTICATION_TYPE 열거형은 등록자(액세스 지점 또는 스테이션)에서 지원하는 인증 유형을 정의합니다.
WCN_VALUE_TYPE_BOOLEAN WCN_VALUE_TYPE_BOOLEAN 열거형은 디바이스 설정 및 연결 중에 발생하는 true/false 조건을 나타내는 데 사용되는 값을 정의합니다.

WCN_VALUE_TYPE_CONFIG_METHODS WCN_VALUE_TYPE_CONFIG_METHODS 열거형은 등록자 또는 등록 기관에서 지원하는 구성 메서드를 정의합니다.
WCN_VALUE_TYPE_CONFIGURATION_ERROR WCN_VALUE_TYPE_CONFIGURATION_ERROR 열거형은 WLAN을 구성하고 연결하는 동안 디바이스에 반환되는 가능한 오류 값을 정의합니다.
WCN_VALUE_TYPE_CONNECTION_TYPE WCN_VALUE_TYPE_CONNECTION_TYPE.
WCN_VALUE_TYPE_DEVICE_PASSWORD_ID WCN_VALUE_TYPE_DEVICE_PASSWORD_ID 열거형은 암호의 원본 또는 '형식'을 지정하는 값을 정의합니다.
WCN_VALUE_TYPE_ENCRYPTION_TYPE WCN_VALUE_TYPE_ENCRYPTION_TYPE 열거형은 지원되는 WLAN 암호화 유형을 정의합니다.
WCN_VALUE_TYPE_RF_BANDS WCN_VALUE_TYPE_RF_BANDS 열거형은 등록자가 검색 요청을 보낼 수 있는 가능한 무선 주파수 대역을 정의합니다.
WCN_VALUE_TYPE_VERSION 지원되는 WPS(Wi-Fi Protected Setup) 버전을 정의합니다.
WCN_VALUE_TYPE_WI_FI_PROTECTED_SETUP_STATE WCN_VALUE_TYPE_WI_FI_PROTECTED_SETUP_STATE 열거형입니다.

wcsplugin.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows Color 시스템에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 색 시스템

wcsplugin.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IDeviceModelPlugIn IDeviceModelPlugIn COM(구성 요소 개체 모델) 인터페이스에 대해 정의된 메서드에 대해 설명합니다.
IGamutMapModelPlugIn IGamutMapModelPlugIn COM(구성 요소 개체 모델) 인터페이스에 대해 정의된 메서드에 대해 설명합니다.

구조체

BlackInformation 검은색 채널이 있는 디바이스 모델에 대한 정보를 포함합니다.
GamutBoundaryDescription 영역 경계를 정의합니다.
GamutShell 인덱싱된 삼각형 목록으로 표시되는 영역 셸을 정의하는 정보를 포함합니다. 꼭짓점 버퍼에는 꼭짓점 데이터가 포함됩니다.
GamutShellTriangle 꼭짓점 버퍼에 액세스하기 위한 세 가지 꼭짓점 인덱스를 포함합니다.

JabColorF JabColorF(wcsplugin.h)는 구조체입니다.
JChColorF JChColorF(wcsplugin.h)는 구조체입니다.
PrimaryJabColors 이 구조체는 Jab 좌표의 8가지 기본 색을 포함합니다.
PrimaryXYZColors 이 구조체에는 XYZ 좌표의 8가지 기본 색이 포함되어 있습니다.
XYZColorF XYZColorF(wcsplugin.h)는 구조체입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wct.h 헤더

아티클2023. 03. 08.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

wct.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

CloseThreadWaitChainSession 지정된 WCT 세션을 닫고 미해결 비동기 작업을 취소합니다.
GetThreadWaitChain 지정된 스레드에 대한 대기 체인을 검색합니다.
OpenThreadWaitChainSession 새 WCT 세션을 만듭니다.
RegisterWaitChainCOMCallback WCT에 대한 COM 콜백 함수를 등록합니다.

콜백 함수

PWAITCHAINCALLBACK 대기 체인을 수신하는 애플리케이션 정의 콜백 함수입니다. OpenThreadWaitChainSession 함수를 호출할 때 이 주소를 지정합니다.

구조체

WAITCHAIN_NODE_INFO 대기 체인의 노드를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wdsbp.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows Deployment Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Deployment Services

wdsbp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WdsBpAddOption 패킷에 옵션을 추가합니다.
WdsBpCloseHandle 지정된 핸들을 닫습니다.
WdsBpGetOptionBuffer DHCP 패킷 옵션에 추가해야 하는 버퍼에 정보를 복사합니다.
WdsBpInitialize WDS 네트워크 부팅 프로그램에 대한 옵션을 생성합니다.
WdsBpParseInitialize 네트워크 부팅 프로그램에서 보낸 패킷에 대한 핸들을 받습니다. (WdsBpParseInitialize)
WdsBpParseInitializev6 네트워크 부팅 프로그램에서 보낸 패킷에 대한 핸들을 받습니다. (WdsBpParseInitializev6)
WdsBpQueryOption 구문 분석된 패킷에서 옵션 값을 반환합니다.

wdsclientapi.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows 배포 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Deployment Services

wdsclientapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WdsCliAuthorizeSession WDS 서버가 있는 세션을 인증된 세션으로 변환합니다.
WdsCliCancelTransfer WDS 전송 작업을 취소합니다.
WdsCliClose WDS 세션 또는 이미지에 대한 핸들을 닫고 리소스를 해제합니다.
WdsCliCreateSession WDS 서버를 사용하여 새 세션을 시작합니다.
WdsCliFindFirstImage WDS 서버에 저장된 이미지의 열거형을 시작하고 첫 번째 이미지를 참조하는 찾기 핸들을 반환합니다.
WdsCliFindNextImage 찾기 핸들의 참조를 WDS 서버에 저장된 다음 이미지로 이동합니다.
WdsCliFreeStringArray 이 함수는 WdsCliObtainDriverPackages 함수에 의해 할당되는 문자열 값 배열을 해제하는 데 사용할 수 있습니다.
WdsCliGetDriverQueryXml 이 함수는 WdsCliObtainDriverPackagesEx 함수를 사용하여 WDS 서버에서 드라이버 패키지를 쿼리하는 데 사용할 수 있는 XML 문자열을 생성합니다.

WdsCliGetEnumerationFlags 현재 이미지 핸들의 이미지 열거형 플래그를 반환합니다.
WdsCliGetImageArchitecture 현재 이미지의 프로세서 아키텍처를 반환합니다.
WdsCliGetImageDescription 현재 이미지에 대한 설명을 반환합니다.
WdsCliGetImageGroup 현재 이미지에 대한 이미지 그룹의 이름을 반환합니다.
WdsCliGetImageHalName 현재 이미지의 HAL(하드웨어 추상화 계층) 이름을 반환합니다.
WdsCliGetImageHandleFromFindHandle 이미지 열거형의 현재 이미지에 대한 이미지 핸들을 반환합니다.
WdsCliGetImageHandleFromTransferHandle 완료된 전송 핸들에서 이미지 핸들을 반환합니다. 핸들은 서버에서 클라이언트로 전송된 이미지의 로컬 복사본에 대한 것입니다.
WdsCliGetImageIndex 현재 이미지의 WIM(Windows Imaging Format) 파일 내에서 인덱스 값을 반환합니다.
WdsCliGetImageLanguage 현재 이미지의 기본 언어를 반환합니다.
WdsCliGetImageLanguages 현재 이미지에서 지원하는 언어 배열을 반환합니다.
WdsCliGetImageLastModifiedTime 현재 이미지의 마지막 수정 시간을 반환합니다.
WdsCliGetImageName 현재 이미지의 이름을 반환합니다.

WdsCliGetImageNamespace 현재 이미지의 네임스페이스를 반환합니다.
WdsCliGetImagePath 현재 이미지가 포함된 파일의 경로를 반환합니다.
WdsCliGetImageSize 현재 이미지의 크기를 반환합니다.
WdsCliGetImageVersion 현재 이미지의 버전을 반환합니다.
WdsCliGetTransferSize 현재 파일 전송의 크기를 반환합니다.
WdsCliInitializeLog WDS 클라이언트에 대한 로깅을 초기화합니다.
WdsCliLog WDS 서버에 로그 이벤트를 보냅니다.
WdsCliObtainDriverPackages 이 함수는 이 컴퓨터에서 사용할 수 있는 드라이버 패키지(INF 파일)인 WDS 이미지에서 가져옵니다.
WdsCliObtainDriverPackagesEx 이 함수는 지정된 WDS 드라이버 쿼리 XML에 적용할 수 있는 드라이버 패키지(INF 파일)를 가져옵니다.
WdsCliRegisterTrace 디버깅 메시지를 받을 콜백 함수를 등록합니다.
WdsCliTransferFile 멀티캐스트 전송 프로토콜을 사용하여 WDS 서버에서 WDS 클라이언트로 파일을 전송합니다.
WdsCliTransferImage WDS 서버에서 WDS 클라이언트로 이미지를 전송합니다.

WdsCliWaitForTransfer 이미지 또는 파일 전송이 완료되기를 기다립니다.

콜백 함수

PFN_WdsCliCallback 파일 또는 이미지 전송 중에 WDS에서 진행률 알림 및 오류 메시지를 호출할 수 있는 콜백 함수를 정의합니다.
PFN_WdsCliTraceFunction WDS 클라이언트에서 디버깅 메시지를 받을 수 있는 콜백 함수를 정의합니다.

구조체

WDS_CLI_CRED 클라이언트 세션에 권한을 부여하는 데 사용되는 자격 증명을 포함합니다.

열거형

__unnamed_enum_1 __unnamed_enum_1(wdsclientapi.h) 열거형은 WdsCliLog 함수에서 사용됩니다.
__unnamed_enum_2 __unnamed_enum_2(wdsclientapi.h) 열거형은 WdsCliLog 함수에서 사용됩니다.

wdspxe.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows 배포 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Deployment Services

wdspxe.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

PxeAsyncRecvDone 클라이언트 요청을 비동기적으로 처리한 결과를 전달합니다. PxeProviderRecvRequest 함수가 ERROR_IO_PENDING 반환하는 경우에만 이 함수를 호출해야 합니다.
PxeDhcpAppendOption 회신 패킷에 DHCP 옵션을 추가합니다. (PxeDhcpAppendOption)
PxeDhcpAppendOptionRaw 회신 패킷에 DHCP 옵션을 추가합니다. (PxeDhcpAppendOptionRaw)
PxeDhcpGetOptionValue DHCP 패킷에서 옵션 값을 검색합니다.
PxeDhcpGetVendorOptionValue DHCP 패킷의 공급업체별 정보 필드(43)에서 옵션 값을 검색합니다.
PxeDhcpInitialize 응답 패킷을 DHCP 회신 패킷으로 초기화합니다.
PxeDhcpIsValid 패킷이 DHCP 패킷인지 확인합니다.
PxeDhcpv6AppendOption 회신 패킷에 DHCPv6 옵션을 추가합니다. (PxeDhcpv6AppendOption)
PxeDhcpv6AppendOptionRaw

회신 패킷에 DHCPv6 옵션을 추가합니다. (PxeDhcpv6AppendOptionRaw)
PxeDhcpv6CreateRelayRepl RELAY-REPL 메시지를 생성합니다.
PxeDhcpv6GetOptionValue DHCPv6 패킷에서 옵션 값을 검색합니다.
PxeDhcpv6GetVendorOptionValue DHCPv6 패킷의 OPTION_VENDOR_OPTS(17) 필드에서 옵션 값을 검색합니다.
PxeDhcpv6Initialize 응답 패킷을 DHCPv6 회신 패킷으로 초기화합니다.
PxeDhcpv6IsValid 패킷이 유효한 DHCPv6 패킷인지 확인합니다.
PxeDhcpv6ParseRelayForw 이 함수는 공급자가 RELAY-FORW 메시지와 중첩된 OPTION_RELAY_MSG 메시지를 구문 분석하는 데 사용할 수 있습니다.
PxeGetServerInfo PXE 서버에 대한 정보를 반환합니다. (PxeGetServerInfo)
PxeGetServerInfoEx PXE 서버에 대한 정보를 반환합니다. (PxeGetServerInfoEx)
PxePacketAllocate PxeSendReply 함수를 사용하여 보낼 패킷을 할당합니다.
PxePacketFree PxePacketAllocate 함수에서 할당한 패킷을 해제합니다.
PxeProviderEnumClose PxeProviderEnumFirst 함수에서 연 공급자의 열거형을 닫습니다.
PxeProviderEnumFirst 등록된 공급자의 열거형을 시작합니다.

PxeProviderEnumNext 등록된 공급자를 열거합니다.
PxeProviderFreeInfo PxeProviderEnumNext 함수에서 할당한 메모리를 해제합니다.
PxeProviderQueryIndex 등록된 공급자 목록에서 지정된 공급자의 인덱스를 반환합니다.
PxeProviderRegister 공급자를 시스템에 등록합니다.
PxeProviderSetAttribute 공급자의 특성을 지정합니다.
PxeProviderUnRegister 등록된 공급자 목록에서 공급자를 제거합니다.
PxeRegisterCallback 다른 알림 이벤트에 대한 콜백 함수를 등록합니다.
PxeSendReply 클라이언트 요청에 패킷을 보냅니다.
PxeTrace PXE 로그에 추적 항목을 추가합니다.

구조체

PXE_ADDRESS 패킷의 네트워크 주소를 지정합니다.
PXE_DHCP_MESSAGE

PXE_DHCP_MESSAGE 구조는 Windows Deployment Services PXE 서버 API와 함께 사용할 수 있습니다.
PXE_DHCP_OPTION PXE_DHCP_OPTION 구조는 Windows Deployment Services PXE 서버 API와 함께 사용할 수 있습니다.
PXE_DHCPV6_MESSAGE DHCPV6 메시지입니다.
PXE_DHCPV6_MESSAGE_HEADER PXE_DHCPV6_MESSAGE 구조체와 PXE_DHCPV6_RELAY_MESSAGE 구조체 간의 공통 필드를 설명합니다.
PXE_DHCPV6_NESTED_RELAY_MESSAGE OPTION_RELAY_MSG 메시지에 중첩된 패킷에 대해 설명합니다.
PXE_DHCPV6_OPTION PXE_DHCPV6_OPTION 구조는 Windows Deployment Services PXE 서버 API와 함께 사용할 수 있습니다.
PXE_DHCPV6_RELAY_MESSAGE DHCPV6 릴레이 메시지를 제공합니다.
PXE_PROVIDER 공급자에 대해 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wdstci.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows Deployment Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Deployment Services

wdstci.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

WdsTransportClientAddRefBuffer 멀티캐스트 클라이언트가 소유한 버퍼에 대한 참조 수를 증분합니다.
WdsTransportClientCancelSession 클라이언트의 세션과 연결된 리소스를 해제합니다. (WdsTransportClientCancelSession)
WdsTransportClientCloseSession 클라이언트의 세션과 연결된 리소스를 해제합니다. (WdsTransportClientCloseSession)
WdsTransportClientCompleteReceive 데이터 블록의 모든 처리가 완료되었으며 멀티캐스트 클라이언트가 캐시에서 이 데이터 블록을 제거하여 추가 수신을 위한 공간을 확보할 수 있음을 나타냅니다.
WdsTransportClientInitialize 멀티캐스트 클라이언트를 초기화합니다.
WdsTransportClientInitializeSession 멀티캐스트 파일 전송을 시작합니다. (WdsTransportClientInitializeSession)
WdsTransportClientQueryStatus 멀티캐스트 클라이언트에서 진행 중이거나 완전한 멀티캐스트 전송의 현재 상태를 검색합니다.
WdsTransportClientRegisterCallback 멀티캐스트 클라이언트에 콜백을 등록합니다.

WdsTransportClientReleaseBuffer 멀티캐스트 클라이언트가 소유한 버퍼의 참조 수를 줄입니다.
WdsTransportClientShutdown 멀티캐스트 클라이언트를 종료합니다.
WdsTransportClientStartSession 멀티캐스트 파일 전송을 시작합니다. (WdsTransportClientStartSession)
WdsTransportClientWaitForCompletion 멀티캐스트 세션이 완료되거나 지정된 시간 제한에 도달할 때까지 차단합니다.

콜백 함수

PFN_WdsTransportClientReceiveContents PFN_WdsTransportClientReceiveContents 콜백은 멀티캐스트 클라이언트에서 데이터 블록을 사용할 준비가 되었음을 나타내는 데 사용됩니다.
PFN_WdsTransportClientReceiveMetadata PFN_WdsTransportClientReceiveMetadata 소비자가 파일에 대한 메타데이터 형식 정보를 수신하도록 등록할 수 있는 선택적 콜백입니다. 이 정보는 콘텐츠 공급자가 제공하며 멀티캐스트 클라이언트 및 서버에 불투명합니다.
PFN_WdsTransportClientSessionComplete PFN_WdsTransportClientSessionCompete 콜백은 더 이상 콜백이 소비자에게 전송되지 않으며 세션이 성공적으로 완료되었거나 복구할 수 없는 오류가 발생했음을 나타내기 위해 클라이언트에서 사용됩니다.
PFN_WdsTransportClientSessionStart PFN_WdsTransportClientSessionStart 콜백은 멀티캐스트 세션이 시작될 때 호출되어 파일에 대한 파일 크기 및 기타 서버 쪽 정보를 소비자에게 나타냅니다. (PFN_WdsTransportClientSessionStart)

PFN_WdsTransportClientSessionStartEx PFN_WdsTransportClientSessionStart 콜백은 멀티캐스트 세션이 시작될 때 호출되어 파일에 대한 파일 크기 및 기타 서버 쪽 정보를 소비자에게 나타냅니다. (PFN_WdsTransportClientSessionStartEx)

구조체

TRANSPORTCLIENT_SESSION_INFO 이 구조는 PFN_WdsTransportClientSessionStartEx 콜백 함수에서 사용됩니다.
WDS_TRANSPORTCLIENT_REQUEST 이 구조체는 WdsTransportClientStartSession 함수에서 사용됩니다.

열거형

TRANSPORTCLIENT_CALLBACK_ID 이 열거형은 WdsTransportClientRegisterCallback 함수에 의해 수신됩니다.

wdstpdi.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows Deployment Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Deployment Services

wdstpdi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WdsTransportProviderCloseContent 핸들로 지정된 콘텐츠 스트림을 닫습니다.
WdsTransportProviderCloseInstance 핸들로 지정된 콘텐츠 공급자의 instance 닫습니다.
WdsTransportProviderCompareContent 지정된 콘텐츠 이름과 instance 열린 콘텐츠 스트림과 비교하여 동일한지 확인합니다.
WdsTransportProviderCreateInstance 콘텐츠 공급자의 새 instance 엽니다.
WdsTransportProviderDumpState 전송 공급자에게 해당 상태 요약 및 기타 관련 정보를 추적 로그에 인쇄하도록 지시합니다.
WdsTransportProviderGetContentMetadata 열린 콘텐츠 스트림에 대한 메타데이터를 검색합니다.
WdsTransportProviderGetContentSize 열린 콘텐츠 스트림의 크기를 검색합니다.
WdsTransportProviderInitialize 콘텐츠 공급자를 초기화합니다.

WdsTransportProviderOpenContent 콘텐츠 스트림의 새 정적 보기를 엽니다.
WdsTransportProviderReadContent 열려 있는 콘텐츠 스트림에서 콘텐츠를 읽습니다.
WdsTransportProviderRefreshSettings 전송 공급자에게 관련 설정을 다시 읽도록 지시합니다.
WdsTransportProviderShutdown 콘텐츠 공급자를 종료합니다.
WdsTransportProviderUserAccessCheck 사용자의 토큰에 따라 콘텐츠 스트림에 대한 액세스를 지정합니다.
WdsTransportServerAllocateBuffer 메모리에 버퍼를 할당합니다.
WdsTransportServerCompleteRead 읽기 작업의 상태 제공합니다.
WdsTransportServerFreeBuffer 버퍼에서 사용하는 메모리를 해제합니다.
WdsTransportServerRegisterCallback 공급자 콜백을 멀티캐스트 서버에 등록합니다.
WdsTransportServerTrace 디버깅 메시지를 보냅니다. (WdsTransportServerTrace)
WdsTransportServerTraceV 디버깅 메시지를 보냅니다. (WdsTransportServerTraceV)

구조체

WDS_TRANSPORTPROVIDER_INIT_PARAMS 이 구조체는 WdsTransportProviderInitialize 콜백 함수에서 사용됩니다. (WDS_TRANSPORTPROVIDER_INIT_PARAMS)
WDS_TRANSPORTPROVIDER_SETTINGS 이 구조체는 WdsTransportProviderInitialize 콜백 함수에서 사용됩니다. (WDS_TRANSPORTPROVIDER_SETTINGS)

열거형

TRANSPORTPROVIDER_CALLBACK_ID 이 구조체는 WdsTransportServerRegisterCallback 함수에서 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wdstptmgmt.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows Deployment Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 배포 서비스

wdstptmgmt.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWdsTransportCacheable 영구 데이터를 처리하는 개체에 대한 캐싱을 제공합니다. 이 인터페이스는 지속형 개체를 나타내는 다른 인터페이스에서 상속할 수 있습니다.
IWdsTransportClient WDS 전송 서버의 전송 세션에 조인된 WDS 클라이언트를 나타냅니다.
IWdsTransportCollection WDS(Windows 배포 서비스) 전송 관리 개체의 컬렉션을 나타냅니다.
IWdsTransportConfigurationManager WDS 전송 서버의 구성을 관리합니다.
IWdsTransportConfigurationManager2 이 인터페이스는 IWdsTransportConfigurationManager 인터페이스에서 상속되며 Windows Server 2008 R2부터 사용할 수 있는 멀티캐스트 세션 정책과 같은 구성 설정으로 확장합니다.
IWdsTransportContent 하나 이상의 세션을 통해 네임스페이스 아래에 전송되는 콘텐츠를 나타냅니다.
IWdsTransportContentProvider 콘텐츠 공급자를 설명하는 데 사용됩니다.
IWdsTransportDiagnosticsPolicy WDS 전송 서버 구성의 진단 정책 부분을 나타냅니다.

IWdsTransportManager WDS 전송 서버를 관리합니다. 이는 WDS(Windows 배포 서비스) 전송 관리 API의 최상위 인터페이스이며 CoCreateInstance 함수를 사용하여 만들 수 있는 유일한 인터페이스입니다.
IWdsTransportMulticastSessionPolicy 이 인터페이스는 WDS 전송 서버 구성의 멀티캐스트 세션 정책 부분을 나타냅니다.
IWdsTransportNamespace WDS 전송 서버의 네임스페이스를 나타냅니다.
IWdsTransportNamespaceAutoCast WDS 전송 서버는 이 인터페이스의 네임스페이스 개체에 의해 호스트되는 멀티캐스트 세션을 만들고, 시작하고, 종료할 수 있습니다.
IWdsTransportNamespaceManager WDS 전송 서버에서 네임스페이스를 관리합니다.
IWdsTransportNamespaceScheduledCast 파생 인터페이스:_IWdsTransportNamespaceScheduledCastManualStart 및 IWdsTransportNamespaceScheduledCastAutoStart에 대한 기본 인터페이스를 나타냅니다.
IWdsTransportNamespaceScheduledCastAutoStart 관리자는 IWdsTransportNamespaceScheduledCastAutoStart 인터페이스의 개체에 대한 전송을 자동으로 시작하는 조건을 지정할 수 있습니다. 애플리케이션은 이러한 조건에 도달할 때까지 네임스페이스 개체의 세션에 계속 조인할 수 있습니다.
IWdsTransportNamespaceScheduledCastManualStart 관리자는 IWdsTransportNamespaceScheduledCastManualStart 인터페이스의 개체에 대한 전송을 시작해야 합니다. 애플리케이션은 관리자가 전송을 시작할 때까지 네임스페이스의 세션에 계속 참가할 수 있습니다.
IWdsTransportServer WDS 전송 서버를 나타냅니다. WDS 클라이언트는 이 인터페이스의 개체를 사용하여 서버에서 설정, 구성 및 네임스페이스 작업을 관리할 수 있습니다.
IWdsTransportServer2 이 인터페이스는 IWdsTransportServer 인터페이스에서 상속되고 확장됩니다. Windows Server 2012부터 사용할 수 있습니다.

IWdsTransportServicePolicy WDS 전송 서버 구성의 서비스 정책 부분을 나타냅니다.
IWdsTransportServicePolicy2 이 인터페이스는 IWdsTransportServicePolicy 인터페이스에서 상속되며 Windows Server 2012부터 확장됩니다.
IWdsTransportSession WDS 전송 서버의 활성 전송 세션을 나타냅니다.
IWdsTransportSetupManager WDS 전송 서버에서 설치 작업을 관리합니다.
IWdsTransportSetupManager2 이 인터페이스는 IWdsTransportSetupManager 인터페이스에서 상속되어 확장합니다. Windows Server 2012부터 사용할 수 있습니다.
IWdsTransportTftpClient 이 인터페이스는 현재 서버에서 활성 상태인 TFTP 클라이언트 세션의 정보를 지정하는 데 사용됩니다.
IWdsTransportTftpManager 이 인터페이스는 현재 TFTP 서버에 연결된 모든 클라이언트를 검색하는 메서드를 제공합니다.

열거형

WDSTRANSPORT_DIAGNOSTICS_COMPONENT_FLAGS 진단 사용하도록 설정된 WDS 구성 요소를 구성합니다. WDS 진단 시스템 이벤트 로그에 이벤트를 기록합니다.
WDSTRANSPORT_DISCONNECT_TYPE WDS 클라이언트가 세션에서 연결이 끊어지면 수행해야 하는 작업을 나타냅니다.
WDSTRANSPORT_FEATURE_FLAGS WDS 서버에 설치된 WDS 기능을 나타냅니다.

WDSTRANSPORT_IP_ADDRESS_SOURCE_TYPE WDS 멀티캐스트 공급자가 새 세션에 대한 멀티캐스트 주소를 가져오는 원본을 나타냅니다.
WDSTRANSPORT_IP_ADDRESS_TYPE IP 주소의 유형을 나타냅니다.
WDSTRANSPORT_NAMESPACE_TYPE 이 네임스페이스에서 다루는 개체를 전송하는 데 사용되는 멀티캐스트 세션의 유형을 결정합니다.
WDSTRANSPORT_NETWORK_PROFILE_TYPE WDS 전송 프로토콜에서 네트워크에서 데이터 전송을 최적화하는 데 사용되는 설정을 정의합니다.
WDSTRANSPORT_PROTOCOL_FLAGS WDS 전송 서버에서 지원하는 프로토콜을 지정합니다.
WDSTRANSPORT_SERVICE_NOTIFICATION 구성 변경 후 설정을 다시 읽는 등 WDS 전송 서비스에 알릴 때 수행해야 하는 작업을 지정합니다.
WDSTRANSPORT_SLOW_CLIENT_HANDLING_TYPE Windows Server 2008 R2에서 실행되는 WDS 전송 서버가 멀티캐스트 전송 속도를 늦추는 클라이언트 컴퓨터를 처리하는 데 사용해야 하는 자동 작업 유형을 지정합니다.
WDSTRANSPORT_TFTP_CAPABILITY WDS TFTP 서버에서 지원하는 기능을 나타냅니다.
WDSTRANSPORT_UDP_PORT_POLICY UDP 포트를 할당할 때 사용해야 하는 정책 WDS 전송 서비스를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

weakreference.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

weakreference.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWeakReference 개체에 대한 약한 참조를 나타냅니다.
IWeakReferenceSource 약한 참조를 검색할 수 있는 원본 개체를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

webapplication.h 헤더

아티클2023. 08. 27.

이 헤더는 디버깅 및 작성 지원에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 디버깅 및 작성 지원

webapplication.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWebApplicationActivation 디버깅 애플리케이션이 활성화를 관리할 수 있도록 합니다.
IWebApplicationAuthoringMode WWAHost 프로세스에 로드할 작성 이진 파일에 대한 전체 로컬 경로를 제공합니다.
IWebApplicationHost WWAHost에서 구현하는 메서드 및 속성을 노출합니다.
IWebApplicationNavigationEvents 디버깅 또는 제작 앱이 탐색 이벤트 알림을 받을 수 있도록 합니다.
IWebApplicationScriptEvents 디버깅 또는 제작 앱이 스크립팅 엔진 이벤트에 대한 알림을 받을 수 있도록 합니다.
IWebApplicationUIEvents 디버깅 또는 제작 앱이 사용자 인터페이스 이벤트에 대한 알림을 받고 사용자 상호 작용이 필요한 이벤트에 응답할 수 있도록 합니다.
IWebApplicationUpdateEvents 작성 앱이 디자이너 이벤트에 대한 알림을 받고 해당 이벤트에 응답할 수 있도록 합니다.

콜백 함수

RegisterAuthoringClientFunctionType 작성 이진 파일로 사용할 DLL(동적 연결 라이브러리)에서 애플리케이션 정의 함수에 대한 포인터를 정의합니다. 앱 호스트가 제작 모드에서 시작되면 이 함수가 호출되어 제작 이진 파일을 초기화합니다.
UnregisterAuthoringClientFunctionType RegisterAuthoringClientFunctionType 함수에 등록된 애플리케이션 정의 함수의 등록을 취소합니다. 이 함수는 앱 호스트가 종료되면 호출됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

webauthenticationcoremanagerinterop.h 헤더

아티클2023. 08. 27.

UWP 앱에서만 사용할 수 있는 WebAuthenticationCoreManager 의 특정 기능에 액세스할 수 있는 Win32 앱을 제공합니다.

webauthenticationcoremanagerinterop.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWebAuthenticationCoreManagerInterop 웹 계정 공급자로부터 토큰을 가져오기 위한 핵심 메서드를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

webauthn.h 헤더

아티클2023. 08. 27.

WebAuthN 및 CTAP 사양의 일부로 Windows Hello 및 외부 보안 키와 통신하기 위한 API를 Win32 앱에 제공합니다.

상수

WebAuthN API에서 사용할 수 있는 **상수를 봅니다** .

webauthn.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WebAuthNAuthenticatorGetAssertion 사용자가 특정 트랜잭션에 동의한 인증자의 어설션을 나타내는 어설션 서명을 생성합니다.
WebAuthNAuthenticatorMakeCredential 관리 인증자에 바인딩된 공개 키 자격 증명 원본을 만들고 자격 증명 프라이빗 키와 연결된 자격 증명 공개 키를 반환합니다.
WebAuthNCancelCurrentOperation 인증자 세션에서 현재 진행 중인 작업을 종료합니다.
WebAuthNDeletePlatformCredential 인증자에 저장된 자격 증명 원본을 제거합니다.
WebAuthNFreeAssertion 이전에 할당된 WebAuthN 어설션을 해제합니다.
WebAuthNFreeCredentialAttestation 이전에 할당된 자격 증명 증명을 해제합니다.
WebAuthNFreePlatformCredentialList 플랫폼 자격 증명 목록에 대한 할당을 해제합니다.

WebAuthNGetApiVersionNumber API 버전 번호를 가져옵니다.
WebAuthNGetCancellationId 취소된 작업의 취소 ID를 가져옵니다.
WebAuthNGetErrorName 오류 코드의 오류 이름을 가져옵니다.
WebAuthNGetPlatformCredentialList 저장된 자격 증명 목록을 가져옵니다.
WebAuthNGetW3CExceptionDOMError 마지막으로 실패한 작업에 대한 W3C DOM 오류 코드를 가져옵니다.
WebAuthNIsUserVerifyingPlatformAuthenticatorAvailable 플랫폼 인증자 서비스를 사용할 수 있는지 확인합니다.

구조체

WEBAUTHN_ASSERTION 어설션을 확인하는 데 필요한 데이터를 포함하는 구조체입니다.
WEBAUTHN_AUTHENTICATOR_GET_ASSERTION_OPTIONS 어설션을 가져오는 옵션이 포함된 구조체입니다.
WEBAUTHN_AUTHENTICATOR_MAKE_CREDENTIAL_OPTIONS WebAuthNAuthenticatorMakeCredential 작업에 대한 옵션입니다.
WEBAUTHN_CLIENT_DATA 인증자로 전송되는 클라이언트 데이터를 포함하는 구조체입니다.
WEBAUTHN_COMMON_ATTESTATION 증명에 대한 공통 데이터를 포함하는 구조체입니다.

WEBAUTHN_COSE_CREDENTIAL_PARAMETER COSE 자격 증명 매개 변수 정보를 포함하는 구조체입니다.
WEBAUTHN_COSE_CREDENTIAL_PARAMETERS 자격 증명에 대한 COSE 매개 변수를 포함하는 구조체입니다.
WEBAUTHN_CRED_BLOB_EXTENSION 자격 증명에 대한 Blob 확장 데이터입니다.
WEBAUTHN_CRED_PROTECT_EXTENSION_IN 자격 증명 보호 확장 정보를 포함합니다.
WEBAUTHN_CRED_WITH_HMAC_SECRET_SALT SALT 값이 있는 자격 증명을 포함하는 구조체입니다.
WEBAUTHN_CREDENTIAL 자격 증명에 대한 정보를 포함합니다.
WEBAUTHN_CREDENTIAL_ATTESTATION 자격 증명에 대한 증명 데이터를 포함합니다.
WEBAUTHN_CREDENTIAL_DETAILS 자격 증명에 대한 데이터를 포함합니다.
WEBAUTHN_CREDENTIAL_DETAILS_LIST 자격 증명 목록입니다.
WEBAUTHN_CREDENTIAL_EX 추가 정보가 포함된 자격 증명에 대한 데이터입니다.
WEBAUTHN_CREDENTIAL_LIST 사용자가 인증자에 등록한 자격 증명 목록입니다.
WEBAUTHN_CREDENTIALS 사용자가 인증자에 등록한 자격 증명의 배열입니다.
WEBAUTHN_EXTENSION

확장에 대한 정보를 포함합니다.
WEBAUTHN_EXTENSIONS 확장 목록을 포함합니다.
WEBAUTHN_GET_CREDENTIALS_OPTIONS WebAuthNGetPlatformCredentialsList 함수에 대한 옵션을 포함합니다.
WEBAUTHN_HMAC_SECRET_SALT Hmac-Secret에 대한 SALT 값을 포함합니다.
WEBAUTHN_HMAC_SECRET_SALT_VALUES HMAC 비밀에 대한 SALT 값을 포함합니다.
WEBAUTHN_RP_ENTITY_INFORMATION 신뢰 당사자에 대한 정보입니다.
WEBAUTHN_USER_ENTITY_INFORMATION 사용자 엔터티에 대한 정보입니다.
WEBAUTHN_X5C X.509 인증서 체인입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

webservices.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows Web Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 웹 서비스

webservices.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WS_STRING_VALUE 상수 문자열이 지정된 WS_STRING 구조를 초기화합니다.
WS_XML_STRING_DICTIONARY_VALUE 연결된 사전 ID가 있는 경우 WS_XML_STRING 구조에 대한 이니셜라이저를 제공합니다.
WS_XML_STRING_VALUE 연결된 사전 ID가 없는 경우 WS_XML_STRING 구조에 대한 이니셜라이저를 제공합니다.
WsAbandonCall 지정된 서비스 프록시에서 지정된 호출을 중단합니다.
WsAbandonMessage 지정된 채널에서 지정된 메시지의 나머지 부분을 건너뜁니다.
WsAbortChannel 지정된 채널에 대해 보류 중인 모든 I/O를 취소합니다.
WsAbortListener 지정된 수신기에 대해 보류 중인 I/O를 취소합니다.
WsAbortServiceHost 지정된 서비스 호스트에서 현재 작업을 모두 중단합니다.
WsAbortServiceProxy

서비스 프록시를 중단하고 서비스 프록시에서 보류 중인 I/O를 취소합니다.
WsAcceptChannel 지정된 수신기에서 다음으로 들어오는 메시지를 수락합니다.
WsAddCustomHeader 지정된 애플리케이션 정의 헤더를 메시지에 추가합니다.
WsAddErrorString 지정된 오류 문자열을 오류 개체에 추가합니다.
WsAddMappedHeader 지정된 매핑된 헤더를 메시지에 추가합니다.
WsAddressMessage 지정된 엔드포인트 주소로 메시지를 주소 지정합니다.
WsAlloc 지정된 힙에서 메모리 세그먼트를 할당합니다.
WsAsyncExecute 비동기 작업을 구현하기 위한 도우미 함수입니다.
WsCall 서비스 프록시에서 내부적으로 지정된 메타데이터에 따라 지정된 인수의 형식을 지정하고 메시지로 보내는 데 사용됩니다. 애플리케이션은 이 함수를 직접 호출해서는 안 됩니다.
WsCheckMustUnderstandHeaders 지정된 헤더가 수신기에서 이해되었는지 확인합니다. 참고 이 함수는 받은 메시지에 대해 모든 헤더를 읽은 후에 호출해야 합니다. .
WsCloseChannel 지정된 채널을 닫습니다.
WsCloseListener 지정된 수신기가 수신 대기를 중지하도록 합니다.
WsCloseServiceHost

지정된 서비스 호스트와의 통신을 닫습니다.
WsCloseServiceProxy 지정된 서비스 프록시와의 통신을 닫습니다.
WsCombineUrl 지정된 URL 참조(절대 또는 상대 URL) 및 지정된 절대 기준 URL에서 절대 URL을 생성합니다.
WsCopyError 지정된 원본에서 지정된 대상으로 오류 개체를 복사합니다.
WsCopyNode 지정된 XML 판독기에서 지정된 XML 기록기로 현재 노드를 복사합니다.
WsCountOf 지정된 배열의 요소 수를 반환합니다.
WsCreateChannel 엔드포인트를 사용하여 메시지 교환을 위한 채널을 만듭니다.
WsCreateChannelForListener 지정된 수신기와 연결된 채널을 만듭니다.
WsCreateError 함수에 전달하여 풍부한 오류 정보를 기록할 수 있는 오류 개체를 만듭니다.
WsCreateFaultFromError 지정된 오류 개체에서 WS_FAULT 생성합니다.
WsCreateHeap 힙 개체를 만듭니다.
WsCreateListener 지정된 속성을 사용하여 수신기를 만듭니다.
WsCreateMessage 지정된 속성을 사용하여 메시지 개체를 만듭니다.

WsCreateMessageForChannel 지정된 채널에 사용할 메시지를 만듭니다.
WsCreateMetadata 메타데이터 문서를 수집하고 처리하는 데 사용되는 메타데이터 개체를 만듭니다.
WsCreateReader 지정된 속성을 사용하여 XML 판독기를 만듭니다.
WsCreateServiceEndpointFromTemplate 정책 템플릿에서 서비스 엔드포인트(WS_SERVICE_ENDPOINT)를 만들기 위한 도우미 루틴입니다.
WsCreateServiceHost 지정된 엔드포인트에 대한 서비스 호스트를 만듭니다.
WsCreateServiceProxy 지정된 속성을 사용하여 서비스 프록시를 만듭니다.
WsCreateServiceProxyFromTemplate 정책 템플릿에서 서비스 프록시를 만들기 위한 도우미 루틴입니다.
WsCreateWriter 는 지정된 속성을 사용하여 XML 기록기를 만듭니다.
WsCreateXmlBuffer XML 데이터를 처리하는 데 사용할 수 있는 XML 버퍼를 만듭니다.
WsCreateXmlSecurityToken 지정된 XML 양식에서 보안 토큰을 만듭니다.
WsDateTimeToFileTime WS_DATETIME 개체를 FILETIME 개체로 변환합니다. FILETIME 개체에 대한 참조는 출력 매개 변수에 의해 반환됩니다.
WsDecodeUrl URL의 구성 요소를 평가하여 "체계"를 확인합니다.

WsEncodeUrl 지정된 WS_URL 구성 요소 파트가 지정된 URL 문자열로 인코딩합니다. 값은 필요에 따라 이스케이프되고, 결합되고, 지정된 WS_HEAP 저장되며 결과는 WS_STRING 반환됩니다.
WsEndReaderCanonicalization 이 함수는 이전 WsStartReaderCanonicalization 함수 호출로 시작된 XML 정식화를 중지합니다. 판독기가 버퍼링한 나머지 정식 바이트는 콜백 함수에 기록됩니다.
WsEndWriterCanonicalization 이 함수는 이전 WsStartWriterCanonicalization 호출로 시작된 XML 정식화를 중지합니다. 작성기가 버퍼링한 나머지 정식 바이트는 콜백 함수에 기록됩니다.
WsFileTimeToDateTime FILETIME 개체에 대한 참조를 사용하여 WS_DATETIME 개체로 변환합니다. WS_DATETIME 개체에 대한 참조는 출력 매개 변수에 의해 반환됩니다.
WsFillBody 메시지에서 읽기에 사용할 수 있는 바이트가 충분한지 확인합니다.
WsFillReader 판독기가 후속 판독기 함수에서 사용할 XML 데이터의 최소 바이트 수를 버퍼링했는지 확인합니다.
WsFindAttribute 현재 요소의 특성을 검색하여 지정된 이름과 네임스페이스가 있는 특성을 검색하고 WsReadStartAttribute에 전달될 수 있는 인덱스를 반환합니다.
WsFlushBody 기록된 모든 누적된 메시지 본문 데이터를 플러시합니다.
WsFlushWriter 충분한 데이터가 버퍼링된 경우 WS_XML_WRITER_STREAM_OUTPUT 콜백을 호출하도록 기록기에 지시합니다.
WsFreeChannel Channel 개체와 연결된 메모리 리소스를 해제합니다.
WsFreeError

WsCreateError를 사용하여 만든 Error 개체와 연결된 메모리 리소스를 해제합니다. 그러면 개체 및 해당 구성 요소가 해제됩니다.
WsFreeHeap 이렇게 하면 WsAlloc을 사용하여 힙 개체 및 할당과 연결된 메모리가 해제됩니다.
WsFreeListener 수신기 개체와 연결된 메모리 리소스를 해제합니다.
WsFreeMessage Message 개체와 연결된 메모리 리소스를 해제합니다.
WsFreeMetadata 메타데이터 개체와 연결된 메모리 리소스를 해제합니다.
WsFreeReader XML_Reader 개체와 연결된 메모리 리소스를 해제합니다.
WsFreeSecurityToken 보안 토큰 개체와 연결된 메모리 리소스를 해제합니다.
WsFreeServiceHost 서비스 호스트 개체와 연결된 메모리를 해제합니다.
WsFreeServiceProxy 서비스 프록시 리소스와 연결된 메모리를 해제합니다.
WsFreeWriter XML 기록기 개체와 연결된 메모리 리소스를 해제합니다.
WsGetChannelProperty 채널 매개 변수에서 참조하는 채널의 속성을 검색합니다.
WsGetCustomHeader 메시지의 애플리케이션 정의 헤더를 찾아 역직렬화합니다.
WsGetDictionary XML Dictionary 개체를 검색합니다. 검색된 사전은 사전 참조 매개 변수에 의해 반환됩니다.

WsGetErrorProperty 오류 매개 변수에서 참조하는 WS_ERROR 개체의 속성을 검색합니다.
WsGetErrorString 오류 개체에서 오류 문자열을 검색합니다.
WsGetFaultErrorDetail WS_ERROR 개체에 저장된 오류 세부 정보를 읽습니다.
WsGetFaultErrorProperty error 매개 변수에서 참조하는 WS_ERROR 개체의 Fault error 속성을 검색합니다.
WsGetHeader 메시지에서 특정 표준 헤더를 찾아 역직렬화합니다.
WsGetHeaderAttributes 이 함수는 ULONG 매개 변수를 판독기가 배치된 헤더 요소의 WS_HEADER_ATTRIBUTES 채웁니다. 메시지의 봉투 버전은 반환할 특성을 결정하는 데 사용됩니다.
WsGetHeapProperty 지정된 힙의 특정 속성을 검색합니다.
WsGetListenerProperty 지정된 수신기 개체 속성을 검색합니다. 검색할 속성은 WS_LISTENER_PROPERTY_ID 입력 매개 변수로 식별됩니다.
WsGetMappedHeader 메시지에서 매핑된 헤더를 찾아 역직렬화합니다.
WsGetMessageProperty 지정된 Message 개체 속성을 검색합니다. 검색할 속성은 WS_MESSAGE_PROPERTY_ID 입력 매개 변수로 식별됩니다.
WsGetMetadataEndpoints 메타데이터 개체 문서 내에 정의된 "엔드포인트"를 반환합니다.
WsGetMetadataProperty

지정된 WS_METADATA 개체 속성을 검색합니다. 검색할 속성은 WS_METADATA_PROPERTY_ID 입력 매개 변수로 식별됩니다.
WsGetMissingMetadataDocumentAddress 이 함수는 메타데이터 개체에서 참조하는 누락된 문서의 주소를 반환합니다.
WsGetNamespaceFromPrefix 이 함수는 바인딩된 접두사에서 네임스페이스를 반환합니다.
WsGetOperationContextProperty 지정된 작업 컨텍스트의 속성을 반환합니다. 이러한 속성의 유효성은 작업 컨텍스트 자체의 수명으로 제한됩니다.
WsGetPolicyAlternativeCount 정책 개체에서 사용할 수 있는 대안 수를 검색합니다. 대체 개수를 사용하여 WsMatchPolicyAlternative를 사용하여 각 대안을 반복할 수 있습니다.
WsGetPolicyProperty 정책 개체의 속성을 검색합니다.
WsGetPrefixFromNamespace 이 함수는 네임스페이스가 바인딩된 접두사를 반환합니다. scope 두 개 이상의 접두사일 수 있으며 이 함수는 해당 접두사 중 하나를 자유롭게 반환할 수 있습니다.
WsGetReaderNode 함수는 XML 판독기의 현재 위치에 있는 XML 노드를 반환합니다.
WsGetReaderPosition 판독기의 현재 위치를 반환합니다. XmlBuffer로 설정된 판독기에서만 사용할 수 있습니다.
WsGetReaderProperty 이 함수는 지정된 XML 판독기의 속성을 반환합니다.
WsGetSecurityContextProperty 지정된 보안 컨텍스트의 속성을 가져옵니다.
WsGetSecurityTokenProperty 보안 토큰에서 필드 또는 속성을 추출합니다.

WsGetServiceHostProperty 지정된 서비스 호스트 속성을 검색합니다. 검색할 속성은 WS_SERVICE_PROPERTY_ID 입력 매개 변수로 식별됩니다.
WsGetServiceProxyProperty 이 함수는 지정된 서비스 프록시 속성을 검색합니다. 검색할 속성은 WS_PROXY_PROPERTY_ID 입력 매개 변수로 식별됩니다.
WsGetWriterPosition 작성기의 현재 위치를 반환합니다. XmlBuffer로 설정된 작성기에서만 사용할 수 있습니다. 버퍼에 쓸 때 위치는 새 데이터가 배치될 xml 노드를 나타냅니다.
WsGetWriterProperty 지정된 XML 기록기 속성을 검색합니다. 검색할 속성은 WS_XML WRITER_PROPERTY_ID 입력 매개 변수로 식별됩니다.
WsGetXmlAttribute 지정된 localName을 사용하여 scope 가장 가까운 xml 특성을 찾아 해당 값을 반환합니다. 반환된 값은 지정된 힙에 배치됩니다.
WsInitializeMessage 이 함수는 처리를 준비하기 위해 메시지의 헤더를 초기화합니다.
WsMarkHeaderAsUnderstood 이 함수는 애플리케이션에서 헤더를 "이해"로 표시합니다.
WsMatchPolicyAlternative 정책 대안이 지정된 정책 제약 조건과 호환되는지 확인합니다.
WsMoveReader moveTo 매개 변수에 지정된 판독기의 현재 위치를 이동합니다. 이 함수는 XmlBuffer로 설정된 판독기에서만 사용할 수 있습니다.
WsMoveWriter moveTo 매개 변수에 지정된 대로 기록기의 현재 위치를 이동합니다.
WsOffsetOf 구조체와 필드의 이름이 지정된 경우 구조체 내 필드의 오프셋(바이트)을 반환합니다.

WsOpenChannel 엔드포인트에 대한 채널을 엽니다.
WsOpenListener 지정된 주소에서 "수신 대기"를 시작합니다. 수신기가 열리면 채널에서 수락할 수 있습니다. 열기에 성공하면 수신기 리소스를 해제하기 전에 WsCloseListener 함수를 사용하여 수신기를 닫아야 합니다.
WsOpenServiceHost 통신을 위해 서비스 호스트를 열고 모든 엔드포인트에서 수신기를 시작합니다. WsOpenSerivceHost가 호출될 때까지 클라이언트 애플리케이션은 서비스 엔드포인트에 연결할 수 없습니다.
WsOpenServiceProxy 서비스 엔드포인트에 대한 서비스 프록시를 엽니다.
WsPullBytes 호출할 콜백을 설정하여 요소 내에 쓸 바이트를 가져옵니다. 일부 인코딩에서는 데이터 복사본을 제거하여 더 효율적일 수 있습니다.
WsPushBytes 요소 내에서 바이트를 쓰기 위해 호출할 콜백을 설정합니다. 일부 인코딩에서는 데이터 복사본을 제거하여 더 효율적일 수 있습니다.
WsReadArray 판독기에서 일련의 요소를 읽고 지정된 값 형식에 따라 해당 콘텐츠를 해석합니다.
WsReadAttribute 지정된 WS_TYPE 값을 생성하는 특성을 읽습니다.
WsReadBody 메시지의 XML Reader에서 값을 역직렬화하는 도우미 함수입니다. WS_MESSAGE_STATE WS_MESSAGE_STATE_READING 설정해야 합니다. 이 함수는 상태 전환을 일으키지 않습니다.
WsReadBytes Reader에서 텍스트를 읽고 base64 사양에 따라 문자를 바이트로 디코딩합니다.
WsReadChars

읽기 권한자에서 지정된 수의 텍스트 문자를 읽습니다.
WsReadCharsUtf8 판독기에서 지정된 수의 텍스트 문자를 읽고 UTF-8로 인코딩된 텍스트를 반환합니다.
WsReadElement 지정된 WS_TYPE 값을 생성하는 요소를 읽습니다.
WsReadEndAttribute 판독기를 읽은 특성이 포함된 요소 노드로 다시 이동합니다.
WsReadEndElement 이 함수는 현재 판독기 노드가 End 요소인지 확인하고 판독기를 다음 노드로 진행합니다.
WsReadEndpointAddressExtension WS_ENDPOINT_ADDRESS 확장을 읽습니다.
WsReadEnvelopeEnd 메시지의 닫는 요소를 읽습니다. 이 작업을 통해 채널 이외의 원본에서 메시지를 읽을 수 있습니다. 원본이 채널인 경우 WsReadMessageEnd를 사용합니다.
WsReadEnvelopeStart 메시지의 헤더를 읽고 본문 요소를 읽을 준비를 합니다.
WsReadMessageEnd 채널에서 메시지의 닫는 요소를 읽습니다.
WsReadMessageStart 채널에서 다음 메시지의 헤더를 읽고 본문 요소를 읽을 준비를 합니다.
WsReadMetadata 메타데이터 요소를 읽고 메타데이터 개체의 메타데이터 문서에 추가합니다.
WsReadNode 이 작업은 판독기를 입력 스트림의 다음 노드로 진행합니다.
WsReadQualifiedName

정규화된 이름을 읽고 XML_READER 현재 네임스페이스 scope 따라 접두사, localName 및 네임스페이스로 구분합니다.
WsReadStartAttribute WsReadValue, WsReadChars 또는 WsReadBytes를 사용하여 콘텐츠를 읽을 수 있도록 판독기를 지정된 특성으로 이동합니다.
WsReadStartElement 이 함수를 호출하면 판독기가 공백을 건너뛰는 시작 요소를 지나서 진행됩니다.
WsReadToStartElement 필요한 경우 공백 및 주석을 건너뛰는 다음 시작 요소로 판독기를 진행합니다. 필요에 따라 요소의 localName 및 네임스페이스를 확인할 수도 있습니다.
WsReadType WS_TYPE_MAPPING 따라 XML에서 지정된 WS_TYPE 값을 읽습니다.
WsReadValue 판독기에서 텍스트를 읽고 지정된 값 형식에 따라 구문 분석합니다.
WsReadXmlBuffer 판독기에서 현재 노드를 WS_XML_BUFFER 읽습니다.
WsReadXmlBufferFromBytes 판독기를 사용하여 인코딩된 바이트 집합을 WS_XML_BUFFER 변환합니다.
WsReceiveMessage 메시지를 수신하고 메시지 본문을 값으로 역직렬화합니다.
WsRegisterOperationForCancel 서비스 작업은 이 함수를 사용하여 취소 알림을 등록할 수 있습니다. 서비스 작업이 실행 중일 때만 이 API를 호출하는 것이 유효합니다. 서비스 작업이 완료된 후 호출하는 동작은 지원되지 않습니다.
WsRemoveCustomHeader 메시지에서 사용자 지정 헤더를 제거합니다. 이 함수는 메시지에 한 번 표시되고 최종 수신자를 대상으로 하는 헤더 유형을 처리하도록 설계되었습니다. 궁극의 수신기 이외의 역할을 대상으로 하는 헤더는 무시됩니다.
WsRemoveHeader

메시지에서 표준 WS_HEADER_TYPE 개체를 제거합니다.
WsRemoveMappedHeader 메시지에서 매핑된 헤더의 모든 인스턴스를 제거합니다.
WsRemoveNode xml 버퍼에서 지정된 위치에 있는 노드를 제거합니다. 요소에 배치된 경우 모든 자식 요소와 해당 끝 요소를 포함한 요소를 제거합니다. 그렇지 않으면 단일 노드를 제거합니다.
WsRequestReply 요청 메시지를 보내고 상관 관계가 있는 회신 메시지를 받는 데 사용됩니다.
WsRequestSecurityToken 페더레이션 시나리오에서 토큰 발급자 역할을 하는 STS(보안 토큰 서비스)에서 보안 토큰을 가져옵니다.
WsResetChannel 다시 사용할 수 있도록 채널을 다시 설정합니다.
WsResetError 오류 개체 매개 변수의 콘텐츠를 해제하지만 오류 개체 매개 변수에 할당된 리소스는 해제하지 않습니다.
WsResetHeap 모든 힙 할당을 해제합니다. WsAlloc을 사용하여 힙에서 수행한 할당은 더 이상 유효하지 않습니다. Heap 개체 자체에 대한 할당은 해제되지 않습니다.
WsResetListener 다시 사용할 수 있도록 수신기 개체를 다시 설정합니다. 이 함수를 사용하려면 수신기 상태를 WS_LISTENER_STATE_CREATED 또는 WS_LISTENER_STATE_CLOSED 설정해야 합니다.
WsResetMessage 메시지 상태를 다시 WS_MESSAGE_STATE_EMPTY 설정합니다. 이 상태에서 Message 개체를 다시 사용할 수 있습니다.
WsResetMetadata 메타데이터 개체 상태를 WS_METADATA_STATE_CREATED 다시 설정합니다. 이 상태에서 메타데이터 개체를 다시 사용할 수 있습니다. 메타데이터 개체를 사용하여 검색된 WS_POLICY 개체가 해제됩니다.

WsResetServiceHost 다시 열 수 있도록 서비스 호스트를 다시 설정합니다.
WsResetServiceProxy 서비스 프록시를 다시 설정합니다.
WsRevokeSecurityContext 보안 컨텍스트를 취소합니다. 서버 쪽에서만 호출할 수 있습니다. 이 보안 컨텍스트를 사용하는 추가 요청이 실패하고 오류가 클라이언트로 전송됩니다.
WsSendFaultMessageForError WS_ERROR 개체가 지정된 경우 오류 메시지를 보냅니다.
WsSendMessage serialization을 사용하여 채널에 메시지를 보내 본문 요소를 작성합니다.
WsSendReplyMessage 받은 메시지에 대한 회신인 메시지를 보냅니다.
WsSetChannelProperty 채널의 속성을 설정합니다.
WsSetErrorProperty WS_ERROR 개체 속성을 설정합니다.
WsSetFaultErrorDetail WS_ERROR 개체에 저장된 오류 세부 정보를 작성합니다.
WsSetFaultErrorProperty WS_ERROR 개체의 Fault 속성을 설정합니다.
WsSetHeader 메시지에 지정된 표준 헤더를 추가하거나 바꿉니다.
WsSetInput XML 판독기 인코딩 및 입력 소스를 설정합니다.

WsSetInputToBuffer 판독기 입력을 지정된 XML 버퍼로 설정합니다. WsSetInputToBuffer에 지정된 판독기 속성은 WsCreateReader에서 설정한 속성을 재정의합니다.
WsSetListenerProperty Listenerobject 속성을 설정합니다.
WsSetMessageProperty 이 작업은 Messageproperty를 설정합니다.
WsSetOutput 작성기에 대한 인코딩 및 출력 콜백을 설정합니다. 콜백은 작성기에 버퍼를 제공하고 비동기 i/o를 수행하는 데 사용됩니다.
WsSetOutputToBuffer 이 작업은 지정된 버퍼의 끝에 기록기를 배치합니다.
WsSetReaderPosition 판독기의 현재 위치를 설정합니다. 위치는 WsGetReaderPosition 또는 WsGetWriterPosition 호출을 통해 가져와야 합니다. 이 함수는 WS_XML_BUFFER 설정된 판독기에서만 사용할 수 있습니다.
WsSetWriterPosition 작성기의 현재 위치를 설정합니다. 위치는 WsGetReaderPosition 또는 WsGetWriterPosition 호출을 통해 가져와야 합니다.
WsShutdownSessionChannel 세션 채널에 대한 메시지의 끝을 알리는 데 사용됩니다.
WsSkipNode 입력 스트림에서 판독기를 진행합니다.
WsStartReaderCanonicalization 이 작업은 지정된 XML 판독기를 표준 또는 "정식" 형식으로 배치하는 프로세스를 시작합니다.
WsStartWriterCanonicalization 지정된 XML 작성기에서 정식화를 시작합니다.

WsTrimXmlWhitespace 문자 시퀀스에서 선행 및 후행 공백을 제거합니다.
WsVerifyXmlNCName 입력 문자열이 유효한 XML NCName인지 확인합니다.
WsWriteArray 이 작업은 일련의 요소를 XML 기록기에 보냅니다.
WsWriteAttribute 형식화된 값을 XML 특성으로 작성합니다.
WsWriteBody 메시지 본문에 값을 씁니다. 이 함수는 값을 메시지의 XML 기록기로 직렬화하는 도우미 함수입니다. 메시지 상태는 WS_MESSAGE_STATE_WRITING 설정해야 합니다. 이 함수는 상태 전환을 일으키지 않습니다.
WsWriteBytes 인코딩에 최적화된 형식으로 작성기에 바이트를 씁니다. 텍스트 인코딩을 작성할 때 base64로 인코딩된 바이트를 내보냅니다. 이진 형식으로 쓸 때 바이트를 직접 내보냅니다.
WsWriteChars 요소 또는 특성에 일련의 문자를 씁니다.
WsWriteCharsUtf8 UTF-8로 인코딩된 일련의 문자를 요소 또는 특성에 씁니다.
WsWriteElement 형식화된 값을 XML 요소로 작성합니다.
WsWriteEndAttribute 이 작업은 현재 요소에 특성을 쓰는 작업을 완료합니다. WsWriteStartAttribute를 호출하는 경우 기록기는 WsWriteEndAttribute가 호출될 때까지 다른 요소 또는 특성을 작성할 수 없습니다.
WsWriteEndCData 기록기에서 CDATA 섹션을 종료합니다.

WsWriteEndElement 기록기에 끝 요소를 씁니다.
WsWriteEndStartElement 작성기가 현재 요소를 커밋하고 추가 특성이 요소에 기록되지 않도록 합니다.
WsWriteEnvelopeEnd 메시지의 닫는 요소를 씁니다.
WsWriteEnvelopeStart 메시지의 현재 헤더 집합을 포함하여 메시지의 시작을 작성하고 본문 요소를 작성할 준비를 합니다.
WsWriteMessageEnd 메시지의 닫는 요소를 채널에 씁니다.
WsWriteMessageStart 메시지의 모든 헤더를 채널에 쓰고 본문 요소를 작성할 준비를 합니다.
WsWriteNode 지정된 노드를 XML 기록기에 씁니다.
WsWriteQualifiedName XML 정규화된 이름을 기록기에 씁니다.
WsWriteStartAttribute 이 작업은 현재 요소에 특성을 쓰기 시작합니다.
WsWriteStartCData 이 작업은 기록기에서 CDATA 섹션을 시작합니다.
WsWriteStartElement 시작 요소를 작성기에 씁니다.
WsWriteText 지정된 텍스트를 XML 기록기에 씁니다.

WsWriteType WS_TYPE_MAPPING 따라 지정된 WS_TYPE 값을 XML에 씁니다.
WsWriteValue 이 작업은 기본 인코딩에서 기본 값에 대한 최상의 표현을 파생하고 파생된 값을 Writer 개체에 전달합니다.
WsWriteXmlBuffer 작성기에 WS_XML_BUFFER 씁니다.
WsWriteXmlBufferToBytes 기록기를 사용하여 WS_XML_BUFFER 인코딩된 바이트 집합으로 변환합니다.
WsWriteXmlnsAttribute Xmlns 특성을 현재 요소에 씁니다.
WsXmlStringEquals 두 WS_XML_STRING 개체를 같음으로 비교합니다. 이 작업은 String 개체에 포함된 문자 값의 서수 비교를 수행합니다.

콜백 함수

WS_ABANDON_MESSAGE_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsAbandonMessage 호출을 처리합니다.
WS_ABORT_CHANNEL_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsAbortChannel 호출을 처리합니다.
WS_ABORT_LISTENER_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsAbortListener 호출을 처리합니다.
WS_ACCEPT_CHANNEL_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsAcceptChannel 호출을 처리합니다.

WS_ASYNC_CALLBACK 비동기 모델과 함께 사용되는 콜백 함수 매개 변수입니다.
WS_ASYNC_FUNCTION WsAsyncExecute와 함께 일련의 비동기 작업에서 호출할 다음 함수를 지정하는 데 사용됩니다. (WS_ASYNC_FUNCTION)
WS_CERT_ISSUER_LIST_NOTIFICATION_CALLBACK 서버에 허용되는 인증서 발급자 목록을 클라이언트에 알립니다.
WS_CERTIFICATE_VALIDATION_CALLBACK WS_CERTIFICATE_VALIDATION_CALLBACK 콜백은 HTTP 서버에 대한 연결이 설정되고 헤더가 전송될 때 인증서의 유효성을 검사하기 위해 호출됩니다.
WS_CLOSE_CHANNEL_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsCloseChannel 호출을 처리합니다.
WS_CLOSE_LISTENER_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsCloseListener 호출을 처리합니다.
WS_CREATE_CHANNEL_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsCreateChannel 호출을 처리합니다.
WS_CREATE_CHANNEL_FOR_LISTENER_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsCreateChannelForListener 호출을 처리합니다.
WS_CREATE_DECODER_CALLBACK 디코더 instance 만드는 것을 처리합니다.
WS_CREATE_ENCODER_CALLBACK 인코더 instance 만드는 것을 처리합니다.
WS_CREATE_LISTENER_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsCreateListener 호출을 처리합니다.
WS_DECODER_DECODE_CALLBACK 메시지를 디코딩합니다.

WS_DECODER_END_CALLBACK 메시지의 끝을 디코딩합니다.
WS_DECODER_GET_CONTENT_TYPE_CALLBACK 메시지의 콘텐츠 형식을 가져옵니다. (WS_DECODER_GET_CONTENT_TYPE_CALLBACK)
WS_DECODER_START_CALLBACK 메시지 디코딩을 시작합니다.
WS_DURATION_COMPARISON_CALLBACK 두 기간을 비교합니다.
WS_DYNAMIC_STRING_CALLBACK 지정된 문자열을 최적화된 형식으로 작성할 수 있는지 여부를 결정합니다.
WS_ENCODER_ENCODE_CALLBACK 메시지를 인코딩합니다.
WS_ENCODER_END_CALLBACK 메시지의 끝을 인코딩합니다.
WS_ENCODER_GET_CONTENT_TYPE_CALLBACK 메시지의 콘텐츠 형식을 가져옵니다. (WS_ENCODER_GET_CONTENT_TYPE_CALLBACK)
WS_ENCODER_START_CALLBACK 메시지 인코딩을 시작합니다.
WS_FREE_CHANNEL_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsFreeChannel 호출을 처리합니다.
WS_FREE_DECODER_CALLBACK 디코더 instance 해제를 처리합니다.
WS_FREE_ENCODER_CALLBACK 인코더 instance 해제를 처리합니다.
WS_FREE_LISTENER_CALLBACK

WS_CUSTOM_CHANNEL_BINDING 대한 WsFreeListener 호출을 처리합니다.
WS_GET_CERT_CALLBACK 보안 런타임에 인증서를 제공합니다.
WS_GET_CHANNEL_PROPERTY_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsGetChannelProperty 호출을 처리합니다.
WS_GET_LISTENER_PROPERTY_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsGetListenerProperty 호출을 처리합니다.
WS_HTTP_REDIRECT_CALLBACK 메시지가 RFC2616 설명된 대로 HTTP 자동 리디렉션 기능을 사용하여 다른 서비스로 자동으로 리디렉션될 때 호출됩니다.
WS_IS_DEFAULT_VALUE_CALLBACK 값이 기본값인지 확인합니다.
WS_MESSAGE_DONE_CALLBACK 메시지가 WsReadEnvelopeStartfunction에 제공된 WS_XML_READER 구조체 또는 WsWriteEnvelopeStart 함수에 제공된 WS_XML_WRITER 구조체의 사용을 완료했음을 호출자에게 알릴 수 있습니다.
WS_OPEN_CHANNEL_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsOpenChannel 호출을 처리합니다.
WS_OPEN_LISTENER_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsOpenListener 호출을 처리합니다.
WS_OPERATION_CANCEL_CALLBACK 서비스 호스트가 중단된 결과로 비동기 서비스 작업 호출의 취소 알림을 제공합니다.
WS_OPERATION_FREE_STATE_CALLBACK 애플리케이션에서 취소 콜백에 등록된 상태 정보를 정리할 수 있습니다.
WS_PROXY_MESSAGE_CALLBACK 입력 메시지의 헤더를 보내려고 하거나 출력 메시지 헤더가 방금 수신될 때 호출됩니다.

WS_PULL_BYTES_CALLBACK WsPullBytes 함수에서 작성해야 하는 데이터를 요청하는 데 사용됩니다.
WS_PUSH_BYTES_CALLBACK WsPushBytes 함수에서 데이터를 작성하도록 요청하는 데 사용됩니다.
WS_READ_CALLBACK WS_XML_READERto 일부 원본에서 버퍼로 읽는 데 사용됩니다.
WS_READ_MESSAGE_END_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsReadMessageEnd 호출을 처리합니다.
WS_READ_MESSAGE_START_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsReadMessageStart 호출을 처리합니다.
WS_READ_TYPE_CALLBACK WS_TYPEhas 지정되면 값을 읽습니다.
WS_RESET_CHANNEL_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsResetChannel 호출을 처리합니다.
WS_RESET_LISTENER_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsResetListener 호출을 처리합니다.
WS_SERVICE_ACCEPT_CHANNEL_CALLBACK 서비스 호스트가 엔드포인트 수신기에서 채널을 수락할 때 호출됩니다.
WS_SERVICE_CLOSE_CHANNEL_CALLBACK 채널이 닫혀 있거나 엔드포인트에서 중단될 때 호출됩니다.
WS_SERVICE_MESSAGE_RECEIVE_CALLBACK defaultMessageHandlerCallback이 설정된 WS_SERVICE_CONTRACT 구성된 엔드포인트에서 WS_MESSAGE 수신될 때 호출됩니다.
WS_SERVICE_SECURITY_CALLBACK 들어오는 메시지의 헤더가 수신되고 본문이 처리되지 않을 때 호출됩니다.

WS_SERVICE_STUB_CALLBACK 서비스 작업 호출에 위임하기 위해 서비스 모델에 의해 호출됩니다.
WS_SET_CHANNEL_PROPERTY_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsSetChannelProperty 호출을 처리합니다.
WS_SET_LISTENER_PROPERTY_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsSetListenerProperty 호출을 처리합니다.
WS_SHUTDOWN_SESSION_CHANNEL_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsShutdownSessionChannel 호출을 처리합니다.
WS_VALIDATE_PASSWORD_CALLBACK 수신자 쪽에서 사용자 이름/암호 쌍의 유효성을 검사합니다.
WS_VALIDATE_SAML_CALLBACK SAML 어설션의 유효성을 검사합니다.
WS_WRITE_CALLBACK WS_XML_WRITER 함수에서 사용자가 결정한 대상에 지정된 버퍼를 쓰는 데 사용됩니다.
WS_WRITE_MESSAGE_END_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsWriteMessageEnd 호출을 처리합니다.
WS_WRITE_MESSAGE_START_CALLBACK WS_CUSTOM_CHANNEL_BINDING 대한 WsWriteMessageStart 호출을 처리합니다.
WS_WRITE_TYPE_CALLBACK WS_CUSTOM_TYPEhas 지정될 때 요소를 작성하기 위해 호출됩니다.

구조체

WS_ANY_ATTRIBUTE 이 형식은 필드에 직접 매핑되지 않은 특성을 저장하는 데 사용됩니다.

WS_ANY_ATTRIBUTES 이 형식은 구조체의 필드에 직접 매핑되지 않은 특성 집합을 저장하는 데 사용됩니다.
WS_ASYNC_CONTEXT 비동기 모델과 함께 비동기 콜백 및 비동기 콜백에 전달될 포인터를 지정하는 데 사용됩니다.
WS_ASYNC_OPERATION WsAsyncExecute와 함께 일련의 비동기 작업에서 호출할 다음 함수를 지정하는 데 사용됩니다. (WS_ASYNC_OPERATION)
WS_ASYNC_STATE WsAsyncExecute에서 비동기 작업의 상태를 관리하는 데 사용됩니다.
WS_ATTRIBUTE_DESCRIPTION C 데이터 형식과 XML 특성 간의 매핑을 나타냅니다.
WS_BOOL_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정합니다. (WS_BOOL_DESCRIPTION)
WS_BUFFERS WS_BYTES 연속 배열을 나타내는 데 사용되는 구조체입니다.
WS_BYTE_ARRAY_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정합니다. (WS_BYTE_ARRAY_DESCRIPTION)
WS_BYTES 바이트 배열을 직렬화하고 역직렬화하는 데 사용됩니다.
WS_BYTES_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정합니다. (WS_BYTES_DESCRIPTION)
WS_CALL_PROPERTY 프록시 속성을 지정합니다. (WS_CALL_PROPERTY)

WS_CAPI_ASYMMETRIC_SECURITY_KEY_HANDLE CAPI 1.0 키 핸들로 비대칭 암호화 키를 지정하기 위한 형식입니다.
WS_CERT_CREDENTIAL 모든 인증서 자격 증명 형식에 대한 추상 기본 형식입니다.
WS_CERT_ENDPOINT_IDENTITY 인증서 엔드포인트 ID에 대한 형식
WS_CERT_MESSAGE_SECURITY_BINDING_CONSTRAINT WS_XML_TOKEN_MESSAGE_SECURITY_BINDING 사용할 수 있는 보안 바인딩 제약 조건입니다.
WS_CERT_SIGNED_SAML_AUTHENTICATOR 예상되는 발급자 인증서 배열을 기반으로 SAML 토큰 인증자를 지정하는 형식입니다.
WS_CERTIFICATE_VALIDATION_CALLBACK_CONTEXT WS_CERTIFICATE_VALIDATION_CALLBACK_CONTEXT 구조에는 HTTP 연결에 대한 인증서의 유효성을 검사하기 위한 콜백 함수 및 상태가 포함됩니다.
WS_CHANNEL_DECODER 수신된 메시지의 콘텐츠 형식 및 인코딩된 바이트를 변환할 수 있는 콜백 집합을 지정하는 데 사용되는 구조체입니다.
WS_CHANNEL_ENCODER 전송된 메시지의 콘텐츠 형식 및 인코딩된 바이트를 변환할 수 있는 콜백 집합을 지정하는 데 사용되는 구조체입니다.
WS_CHANNEL_PROPERTIES WS_CHANNEL_PROPERTY 구조체 집합을 지정합니다.
WS_CHANNEL_PROPERTY 채널별 설정을 지정합니다.
WS_CHANNEL_PROPERTY_CONSTRAINT 특정 채널 속성에 대한 제약 조건을 지정합니다.

WS_CHAR_ARRAY_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정합니다. (WS_CHAR_ARRAY_DESCRIPTION)
WS_CONTRACT_DESCRIPTION 서비스 모델에 대한 서비스 계약에 대한 메타데이터입니다.
WS_CUSTOM_CERT_CREDENTIAL 애플리케이션에 대한 콜백에서 제공할 인증서 자격 증명을 지정하기 위한 형식입니다.
WS_CUSTOM_CHANNEL_CALLBACKS 사용자 지정 채널의 구현을 형성하는 콜백 집합을 지정하는 데 사용되는 구조체입니다.
WS_CUSTOM_HTTP_PROXY WS_CHANNEL_PROPERTY_CUSTOM_HTTP_PROXY 사용하여 채널에 대한 사용자 지정 프록시를 지정하는 데 사용되는 구조체입니다.
WS_CUSTOM_LISTENER_CALLBACKS 사용자 지정 수신기의 구현을 구성하는 콜백 집합을 지정하는 데 사용되는 구조체입니다.
WS_CUSTOM_TYPE_DESCRIPTION C 데이터 형식과 XML 요소 간의 사용자 지정 매핑을 나타냅니다.
WS_DATETIME 이 구조체는 날짜와 시간을 나타내는 데 사용됩니다.
WS_DATETIME_DESCRIPTION 이 형식 설명은 WS_DATETIME_TYPE 사용되며 선택 사항입니다. 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하는 데 사용됩니다.
WS_DECIMAL_DESCRIPTION WS_DECIMAL_TYPE 사용하는 선택적 형식 설명입니다. 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하는 데 사용됩니다.
WS_DEFAULT_VALUE

필드의 기본값을 정의합니다. 이는 WS_FIELD_DESCRIPTION 사용됩니다.
WS_DEFAULT_WINDOWS_INTEGRATED_AUTH_CREDENTIAL 현재 Windows ID를 기반으로 하는 Windows 통합 인증 자격 증명을 제공하기 위한 형식입니다.
WS_DISALLOWED_USER_AGENT_SUBSTRINGS 차단된 UserAgent 하위 문자열의 목록을 지정합니다. WS_LISTENER_PROPERTY_DISALLOWED_USER_AGENT 수신기 속성과 함께 사용됩니다.
WS_DNS_ENDPOINT_IDENTITY DNS 이름으로 표시되는 엔드포인트 ID를 지정하기 위한 형식입니다.
WS_DOUBLE_DESCRIPTION WS_DOUBLE_TYPE 사용되는 선택적 형식 설명입니다. 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하는 데 사용됩니다.
WS_DURATION xsd:duration 데이터 형식을 나타냅니다.
WS_DURATION_DESCRIPTION WS_DURATION_TYPE 사용되는 선택적 형식 설명입니다. 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하는 데 사용됩니다.
WS_ELEMENT_DESCRIPTION C 데이터 형식과 XML 요소 간의 매핑을 나타냅니다.
WS_ENDPOINT_ADDRESS 엔드포인트의 네트워크 주소를 나타냅니다.
WS_ENDPOINT_ADDRESS_DESCRIPTION WS_ENDPOINT_ADDRESS XML 요소 간의 매핑에 대한 정보입니다.
WS_ENDPOINT_IDENTITY 모든 엔드포인트 ID의 기본 형식입니다.
WS_ENDPOINT_POLICY_EXTENSION

이 구조체는 엔드포인트 정책 확장을 지정하는 데 사용됩니다.
WS_ENUM_DESCRIPTION WS_ENUM_TYPE 함께 사용되며 필요한 형식 설명입니다. 열거형의 값을 직렬화하고 역직렬화하는 데 사용되는 정보를 제공합니다.
WS_ENUM_VALUE 열거형(WS_ENUM_DESCRIPTION)의 일부인 단일 값에 대한 serialization 정보를 제공합니다.
WS_ERROR_PROPERTY 오류 특정 설정을 지정합니다.
WS_FAULT 오류는 처리 실패를 전달하는 메시지 본문에 전달되는 값입니다. 오류는 WS_FAULT 구조를 사용하여 모델링됩니다.
WS_FAULT_CODE 오류 코드를 나타냅니다.
WS_FAULT_DESCRIPTION WS_FAULT XML 요소 간의 매핑에 대한 정보입니다.
WS_FAULT_DETAIL_DESCRIPTION 오류 메시지의 세부 요소에 대한 설명입니다.
WS_FAULT_REASON 오류에 대한 설명을 포함합니다.
WS_FIELD_DESCRIPTION 구조체 내의 필드에 대한 serialization 정보를 나타냅니다.
WS_FLOAT_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하기 위해 WS_FLOAT_TYPE 사용하는 선택적 형식 설명입니다.
WS_GUID_DESCRIPTION

역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하기 위해 WS_GUID_TYPE 사용하는 선택적 형식 설명입니다.
WS_HEAP_PROPERTIES WS_HEAP_PROPERTYs 집합을 지정하는 데 사용되는 구조체입니다.
WS_HEAP_PROPERTY 힙 특정 설정을 지정합니다.
WS_HOST_NAMES 호스트 이름 목록을 포함하는 구조체입니다.
WS_HTTP_BINDING_TEMPLATE http 바인딩을 위해 애플리케이션에서 채울 HTTP 템플릿 구조입니다.
WS_HTTP_HEADER_AUTH_BINDING_TEMPLATE 애플리케이션에서 채울 HTTP 헤더 인증 보안 템플릿 정보입니다. WS_HTTP_HEADER_AUTH_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_HTTP_HEADER_AUTH_POLICY_DESCRIPTION http 채널 바인딩을 지정하는 정책에 대해 설명합니다. (WS_HTTP_HEADER_AUTH_POLICY_DESCRIPTION)
WS_HTTP_HEADER_AUTH_SECURITY_BINDING 기본, 다이제스트(RFC 2617) 및 SPNEGO(RFC4559) 프로토콜을 기반으로 대상 서비스 또는 HTTP 프록시 서버에 대한 HTTP 헤더 인증 사용을 지정하기 위한 보안 바인딩 하위 형식입니다.
WS_HTTP_HEADER_AUTH_SECURITY_BINDING_CONSTRAINT WS_HTTP_HEADER_AUTH_SECURITY_BINDING 해당하는 보안 바인딩 제약 조건입니다.
WS_HTTP_HEADER_AUTH_SECURITY_BINDING_POLICY_DESCRIPTION 이 형식 설명은 템플릿 API와 함께 입력 정책 설정에 따라 생성된 템플릿을 설명하는 데 사용됩니다. (WS_HTTP_HEADER_AUTH_SECURITY_BINDING_POLICY_DESCRIPTION)
WS_HTTP_HEADER_AUTH_SECURITY_BINDING_TEMPLATE

HTP 헤더 인증 프로토콜 기반 보안 사용을 지정하기 위한 보안 바인딩 템플릿입니다.
WS_HTTP_HEADER_MAPPING WS_HTTP_MESSAGE_MAPPING 일부로 매핑되는 개별 헤더를 지정합니다.
WS_HTTP_MESSAGE_MAPPING 메시지 개체에서 HTTP 요청 또는 응답을 나타내는 방법입니다.
WS_HTTP_POLICY_DESCRIPTION http 채널 바인딩을 지정하는 정책에 대해 설명합니다. (WS_HTTP_POLICY_DESCRIPTION)
WS_HTTP_REDIRECT_CALLBACK_CONTEXT HTTP 자동 리디렉션 동작을 제어하기 위한 콜백 함수 및 상태를 지정합니다.
WS_HTTP_SSL_BINDING_TEMPLATE 애플리케이션에서 채울 SSL 보안 템플릿 정보입니다. WS_HTTP_SSL_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_HTTP_SSL_HEADER_AUTH_BINDING_TEMPLATE 애플리케이션에서 입력할 사용자 이름/암호 보안 템플릿 정보입니다. WS_HTTP_SSL_HEADER_AUTH_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_HTTP_SSL_HEADER_AUTH_POLICY_DESCRIPTION SSL 전송 보안 및 헤더 인증을 사용하여 http 채널 바인딩을 지정하는 정책에 대해 설명합니다.
WS_HTTP_SSL_KERBEROS_APREQ_BINDING_TEMPLATE 애플리케이션에서 입력할 사용자 이름/암호 보안 템플릿 정보입니다. WS_HTTP_SSL_KERBEROS_APREQ_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_HTTP_SSL_KERBEROS_APREQ_POLICY_DESCRIPTION SSL 전송 보안 및 KERBEROS AP_REQ 메시지 보안을 사용하여 http 채널 바인딩을 지정하는 정책을 설명합니다.
WS_HTTP_SSL_KERBEROS_APREQ_SECURITY_CONTEXT_BINDING_TEMPLATE 애플리케이션에서 채울 보안 템플릿 정보입니다.

WS_HTTP_SSL_KERBEROS_APREQ_SECURITY_CONTEXT_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_HTTP_SSL_KERBEROS_APREQ_SECURITY_CONTEXT_POLICY_DESCRIPTION SSL 전송 보안을 사용하여 http 채널 바인딩을 통해 보안 컨텍스트 메시지 바인딩을 지정하는 정책을 설명합니다. 부트스트랩 채널은 SSL 전송 보안 및 KERBEROS AP_REQ 메시지 보안과 함께 http 채널 바인딩을 사용합니다.
WS_HTTP_SSL_POLICY_DESCRIPTION http 채널 바인딩을 지정하는 정책에 대해 설명합니다. (WS_HTTP_SSL_POLICY_DESCRIPTION)
WS_HTTP_SSL_USERNAME_BINDING_TEMPLATE 애플리케이션에서 입력할 사용자 이름/암호 보안 템플릿 정보입니다. WS_HTTP_SSL_USERNAME_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_HTTP_SSL_USERNAME_POLICY_DESCRIPTION SSL 전송 보안 및 사용자 이름/암호 메시지 보안을 사용하여 http 채널 바인딩을 지정하는 정책을 설명합니다.
WS_HTTP_SSL_USERNAME_SECURITY_CONTEXT_BINDING_TEMPLATE 애플리케이션에서 채울 보안 템플릿 정보입니다. WS_HTTP_SSL_USERNAME_SECURITY_CONTEXT_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_HTTP_SSL_USERNAME_SECURITY_CONTEXT_POLICY_DESCRIPTION SSL 전송 보안을 사용하여 http 채널 바인딩을 통해 보안 컨텍스트 메시지 바인딩을 지정하는 정책을 설명합니다. 부트스트랩 채널은 SSL 전송 보안 및 사용자 이름/암호 메시지 보안과 함께 http 채널 바인딩을 사용합니다.
WS_HTTP_URL HTTP URL을 지정하기 위한 URL 하위 형식입니다.
WS_HTTPS_URL HTTPS URL을 지정하기 위한 URL 하위 형식입니다.
WS_INT16_DESCRIPTION 역직렬화할 수 있는 값 집합에 제약 조건을 지정하기 위해 WS_INT16_TYPE 사용하는 선택적 형식 설명입니다.

WS_INT32_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하기 위해 WS_INT32_TYPE 사용하는 선택적 형식 설명입니다.
WS_INT64_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하기 위해 WS_INT64_TYPE 사용하는 선택적 형식 설명입니다.
WS_INT8_DESCRIPTION 역직렬화할 수 있는 값 집합에 제약 조건을 지정하기 위해 WS_INT8_TYPE 사용하는 선택적 형식 설명입니다.
WS_ISSUED_TOKEN_MESSAGE_SECURITY_BINDING_CONSTRAINT 발급 당사자로부터 발급된 토큰을 가져오는 방법에 대한 정보를 추출하는 데 사용할 수 있는 보안 바인딩 제약 조건입니다.
WS_ITEM_RANGE WS_FIELD_DESCRIPTION 내에서 WS_REPEATING_ELEMENT_FIELD_MAPPING, WS_REPEATING_ELEMENT_CHOICE_FIELD_MAPPING 또는 WS_REPEATING_ANY_ELEMENT_FIELD_MAPPING 사용할 때 나타날 수 있는 최소 및 최대 항목 수를 정의합니다.
WS_KERBEROS_APREQ_MESSAGE_SECURITY_BINDING WS-Security를 사용하여 Kerberos AP_REQ 티켓을 직접(즉, 세션을 설정하지 않고) 보안 토큰으로 사용하도록 지정하기 위한 보안 바인딩 하위 형식입니다.
WS_KERBEROS_APREQ_MESSAGE_SECURITY_BINDING_CONSTRAINT WS_KERBEROS_APREQ_MESSAGE_SECURITY_BINDING 해당하는 보안 바인딩 제약 조건입니다.
WS_KERBEROS_APREQ_MESSAGE_SECURITY_BINDING_POLICY_DESCRIPTION 이 형식 설명은 입력 정책 설정에 따라 생성된 템플릿을 설명하기 위해 템플릿 API와 함께 사용됩니다. (WS_KERBEROS_APREQ_MESSAGE_SECURITY_BINDING_POLICY_DESCRIPTION)
WS_KERBEROS_APREQ_MESSAGE_SECURITY_BINDING_TEMPLATE WS-Security를 사용하여 Kerberos AP_REQ 티켓을 직접(즉, 세션을 설정하지 않고) 보안 토큰으로 사용하도록 지정하기 위한 보안 바인딩 템플릿입니다.

WS_LISTENER_PROPERTIES WS_LISTENER_PROPERTY 구조체 집합을 지정합니다.
WS_LISTENER_PROPERTY 수신기 특정 설정을 지정합니다.
WS_MESSAGE_DESCRIPTION 지정된 작업 설명에 대한 입력/출력 WS_MESSAGE 스키마입니다.
WS_MESSAGE_PROPERTIES WS_MESSAGE_PROPERTY 구조체 집합을 지정합니다.
WS_MESSAGE_PROPERTY 메시지별 설정을 지정합니다.
WS_METADATA_ENDPOINT 메타데이터 문서에서 읽은 단일 엔드포인트에 대한 정보입니다.
WS_METADATA_ENDPOINTS 메타데이터 문서에서 읽은 모든 엔드포인트에 대한 정보입니다.
WS_METADATA_PROPERTY 메타데이터 개체 설정을 지정합니다.
WS_NAMEDPIPE_SSPI_TRANSPORT_SECURITY_BINDING 명명된 파이프 전송과 함께 Windows 통합 인증 프로토콜(예: Kerberos, NTLM 또는 SPNEGO)의 사용을 지정하기 위한 보안 바인딩 하위 유형입니다.
WS_NCRYPT_ASYMMETRIC_SECURITY_KEY_HANDLE 비대칭 암호화 키를 CryptoNG NCRYPT_KEY_HANDLE 지정하기 위한 형식입니다.
WS_NETPIPE_URL net.pipe URL을 지정하기 위한 URL 하위 형식입니다.
WS_NETTCP_URL net.tcp URL을 지정하기 위한 URL 하위 형식입니다.

WS_OPAQUE_WINDOWS_INTEGRATED_AUTH_CREDENTIAL SspiPromptForCredentials 및 관련 API 제품군에서 만든 불투명 핸들로 Windows 통합 인증 자격 증명을 제공하기 위한 형식입니다. 이 기능은 Windows 7 이상에서만 사용할 수 있습니다.
WS_OPERATION_DESCRIPTION 서비스 작업에 대한 메타데이터입니다.
WS_PARAMETER_DESCRIPTION 들어오는/나가는 메시지 필드 설명에 있는 매개 변수의 인덱스입니다.
WS_POLICY_CONSTRAINTS 채널에 대한 정책 제약 조건을 지정합니다.
WS_POLICY_EXTENSION 모든 정책 확장 구조에 대한 기본 클래스입니다. 정책 확장은 사용자 지정 어설션과 같은 애플리케이션에서 직접 처리하는 어설션입니다.
WS_POLICY_PROPERTIES WS_POLICY_PROPERTY 구조체 집합을 지정합니다.
WS_POLICY_PROPERTY 정책 개체 설정을 지정합니다.
WS_PROXY_MESSAGE_CALLBACK_CONTEXT 입력 또는 출력 메시지의 헤더를 각각 연결하거나 검사하려는 애플리케이션의 콜백 함수 및 상태를 지정합니다.
WS_PROXY_PROPERTY 프록시 속성을 지정합니다. (WS_PROXY_PROPERTY)
WS_RAW_SYMMETRIC_SECURITY_KEY_HANDLE 대칭 암호화 키를 원시 바이트로 지정하기 위한 형식입니다.
WS_REQUEST_SECURITY_TOKEN_PROPERTY 발급자에서 보안 토큰을 요청하기 위한 속성을 지정합니다.

WS_REQUEST_SECURITY_TOKEN_PROPERTY_CONSTRAINT 이 구조체는 특정 요청 보안 토큰 속성에 대한 제약 조건 집합을 지정하는 데 사용됩니다. 지정되지 않은 모든 속성 제약 조건은 기본 제약 조건을 사용합니다.
WS_RSA_ENDPOINT_IDENTITY RSA 엔드포인트 ID를 입력합니다.
WS_SAML_AUTHENTICATOR 들어오는 SAML 토큰의 유효성을 검사하기 위해 서버 쪽에서 사용되는 모든 SAML 인증자에 대한 추상 기본 형식입니다.
WS_SAML_MESSAGE_SECURITY_BINDING SAML 어설션을 메시지 보안 토큰으로 사용하도록 지정하기 위한 보안 바인딩 하위 형식입니다.
WS_SECURITY_ALGORITHM_PROPERTY 암호화 알고리즘 설정을 지정합니다.
WS_SECURITY_ALGORITHM_SUITE WS-Security와 함께 사용할 보안 알고리즘 및 키 길이를 정의합니다. 이 설정은 메시지 보안 바인딩 및 혼합 모드 보안 바인딩과 관련이 있습니다.
WS_SECURITY_BINDING 모든 보안 바인딩에 대한 추상 기본 형식입니다.
WS_SECURITY_BINDING_CONSTRAINT 모든 보안 바인딩 제약 조건 구조에 대한 기본 클래스입니다.
WS_SECURITY_BINDING_PROPERTIES 보안 바인딩 설정의 배열을 지정합니다.
WS_SECURITY_BINDING_PROPERTY 보안 바인딩 특정 설정을 지정합니다.
WS_SECURITY_BINDING_PROPERTY_CONSTRAINT 이 구조체는 특정 보안 바인딩 속성에 대한 제약 조건 집합을 지정하는 데 사용됩니다. 지정되지 않은 모든 속성 제약 조건은 기본 제약 조건을 사용합니다.

WS_SECURITY_CONSTRAINTS 이 구조는 보안 관련 제약 조건을 WS_POLICY_CONSTRAINTS 일부로 지정합니다.
WS_SECURITY_CONTEXT_MESSAGE_SECURITY_BINDING WS-SecureConversation을 사용하여 클라이언트와 서버 간에 협상된 보안 컨텍스트 토큰의 사용을 지정하기 위한 보안 바인딩 하위 형식입니다.
WS_SECURITY_CONTEXT_MESSAGE_SECURITY_BINDING_CONSTRAINT WS_SECURITY_CONTEXT_MESSAGE_SECURITY_BINDING 해당하는 보안 바인딩 제약 조건입니다.
WS_SECURITY_CONTEXT_MESSAGE_SECURITY_BINDING_POLICY_DESCRIPTION 이 형식 설명은 입력 정책 설정에 따라 생성된 템플릿을 설명하기 위해 템플릿 API와 함께 사용됩니다. (WS_SECURITY_CONTEXT_MESSAGE_SECURITY_BINDING_POLICY_DESCRIPTION)
WS_SECURITY_CONTEXT_MESSAGE_SECURITY_BINDING_TEMPLATE 애플리케이션 제공 보안 컨텍스트 보안 바인딩의 사용을 지정하기 위한 보안 바인딩 템플릿입니다. (WS_SECURITY_CONTEXT_MESSAGE_SECURITY_BINDING_TEMPLATE)
WS_SECURITY_CONTEXT_PROPERTY WS_SECURITY_CONTEXT 속성을 정의합니다.
WS_SECURITY_CONTEXT_SECURITY_BINDING_POLICY_DESCRIPTION 이 형식 설명은 템플릿 API와 함께 입력 정책 설정에 따라 생성된 보안 컨텍스트 관련 템플릿을 설명하는 데 사용됩니다.
WS_SECURITY_CONTEXT_SECURITY_BINDING_TEMPLATE 애플리케이션 제공 보안 컨텍스트 보안 바인딩의 사용을 지정하기 위한 보안 바인딩 템플릿입니다. (WS_SECURITY_CONTEXT_SECURITY_BINDING_TEMPLATE)
WS_SECURITY_DESCRIPTION 채널(클라이언트 쪽) 또는 수신기(서버 쪽)에 대한 보안 요구 사항을 지정하는 데 사용되는 최상위 구조입니다.
WS_SECURITY_KEY_HANDLE 암호화 키를 지정하는 모든 형식에 대한 추상 기본 형식입니다. 이러한 키는 일반적으로 일반 XML 보안 토큰 또는 사용자 지정 보안 토큰에 대해 지정됩니다.

WS_SECURITY_PROPERTIES 채널 전체 보안 설정의 배열을 지정합니다.
WS_SECURITY_PROPERTY 채널 전체 보안 설정을 지정합니다.
WS_SECURITY_PROPERTY_CONSTRAINT 이 구조체는 특정 보안 속성에 대한 제약 조건 집합을 지정하는 데 사용됩니다. 지정되지 않은 속성 제약 조건은 기본 제약 조건을 사용합니다.
WS_SERVICE_CONTRACT 엔드포인트에서 서비스 계약을 지정합니다.
WS_SERVICE_ENDPOINT 서비스 호스트의 개별 엔드포인트를 나타냅니다. 엔드포인트의 속성은 주소, 바인딩 및 계약을 지정하는 데 사용됩니다.
WS_SERVICE_ENDPOINT_METADATA 엔드포인트의 포트 요소를 나타냅니다. 포트 요소는 serviceName에 지정된 대로 서비스 요소에 대해 생성되고 WS_SERVICE_HOST WS_SERVICE_PROPERTY_METADATA 속성에 대한 serviceNs가 생성됩니다.
WS_SERVICE_ENDPOINT_PROPERTY 서비스별 설정을 지정합니다. (WS_SERVICE_ENDPOINT_PROPERTY)
WS_SERVICE_METADATA 서비스 메타데이터 문서 배열을 지정합니다. WS_STRING 배열로 표시되는 WSDL/XSD 문서의 컬렉션일 수 있습니다.
WS_SERVICE_METADATA_DOCUMENT 서비스 메타데이터를 구성하는 개별 문서를 지정합니다.
WS_SERVICE_PROPERTY 서비스별 설정을 지정합니다. (WS_SERVICE_PROPERTY)
WS_SERVICE_PROPERTY_ACCEPT_CALLBACK 채널이 성공적으로 수락될 때 호출되는 콜백을 지정합니다.

WS_SERVICE_PROPERTY_CLOSE_CALLBACK 채널을 닫을 때 호출되는 콜백을 지정합니다. 자세한 내용은 WS_SERVICE_CLOSE_CHANNEL_CALLBACK 참조하세요.
WS_SERVICE_SECURITY_IDENTITIES 확장된 보호의 유효성을 검사하는 데 사용되는 SPN(서버 보안 주체 이름) 목록입니다.
WS_SOAPUDP_URL soap.udp URL을 지정하기 위한 URL 하위 형식입니다.
WS_SPN_ENDPOINT_IDENTITY SPN(서비스 주체 이름)으로 표시되는 엔드포인트 ID를 지정하기 위한 형식입니다.
WS_SSL_TRANSPORT_SECURITY_BINDING SSL/TLS 프로토콜 기반 전송 보안 사용을 지정하기 위한 보안 바인딩 하위 형식입니다.
WS_SSL_TRANSPORT_SECURITY_BINDING_CONSTRAINT WS_SSL_TRANSPORT_SECURITY_BINDING 해당하는 보안 바인딩 제약 조건입니다.
WS_SSL_TRANSPORT_SECURITY_BINDING_POLICY_DESCRIPTION 이 형식 설명은 템플릿 API와 함께 입력 정책 설정에 따라 생성된 템플릿을 설명하는 데 사용됩니다. (WS_SSL_TRANSPORT_SECURITY_BINDING_POLICY_DESCRIPTION)
WS_SSL_TRANSPORT_SECURITY_BINDING_TEMPLATE SSL/TLS 프로토콜 기반 전송 보안 사용을 지정하기 위한 보안 바인딩 템플릿입니다.
WS_SSPI_TRANSPORT_SECURITY_BINDING_POLICY_DESCRIPTION 이 형식 설명은 템플릿 API와 함께 입력 정책 설정에 따라 생성된 템플릿을 설명하는 데 사용됩니다. (WS_SSPI_TRANSPORT_SECURITY_BINDING_POLICY_DESCRIPTION)
WS_STRING 유니코드 문자 및 길이 배열입니다.

WS_STRING_DESCRIPTION 이 형식 설명은 WS_STRING_TYPE 사용되며 선택 사항입니다. 역직렬화할 수 있는 값 집합에 제약 조건을 지정하는 데 사용됩니다.
WS_STRING_USERNAME_CREDENTIAL 사용자 이름/암호 쌍을 문자열로 제공하기 위한 형식입니다.
WS_STRING_WINDOWS_INTEGRATED_AUTH_CREDENTIAL Windows 자격 증명을 사용자 이름, 암호, 도메인 문자열로 제공하기 위한 형식입니다.
WS_STRUCT_DESCRIPTION C 구조체 형식 및 XML 요소에 매핑되는 방법에 대한 정보입니다. 이는 WS_STRUCT_TYPE 사용됩니다.
WS_SUBJECT_NAME_CERT_CREDENTIAL 인증서의 주체 이름, 저장소 위치 및 저장소 이름을 사용하여 인증서 자격 증명을 지정하는 형식입니다. 포함된 채널 또는 수신기가 열릴 때 지정된 자격 증명이 로드됩니다.
WS_TCP_BINDING_TEMPLATE TCP 바인딩을 위해 애플리케이션에서 채울 TCP 템플릿 구조입니다.
WS_TCP_POLICY_DESCRIPTION http 채널 바인딩을 지정하는 정책에 대해 설명합니다. (WS_TCP_POLICY_DESCRIPTION)
WS_TCP_SSPI_BINDING_TEMPLATE 애플리케이션에서 채울 HTTP 헤더 인증 보안 템플릿 정보입니다. WS_TCP_SSPI_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_TCP_SSPI_KERBEROS_APREQ_BINDING_TEMPLATE 애플리케이션에서 입력할 사용자 이름/암호 보안 템플릿 정보입니다. WS_TCP_SSPI_KERBEROS_APREQ_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_TCP_SSPI_KERBEROS_APREQ_POLICY_DESCRIPTION Windows SSPI 전송 보안 및 kerberos 메시지 보안을 사용하여 TCP 채널 바인딩을 지정하는 정책에 대해 설명합니다.

WS_TCP_SSPI_KERBEROS_APREQ_SECURITY_CONTEXT_BINDING_TEMPLATE 애플리케이션에서 채울 보안 템플릿 정보입니다. WS_TCP_SSPI_KERBEROS_APREQ_SECURITY_CONTEXT_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_TCP_SSPI_KERBEROS_APREQ_SECURITY_CONTEXT_POLICY_DESCRIPTION Windows SSPI 전송 보안과 함께 TCP 채널 바인딩을 사용하여 보안 컨텍스트 메시지 바인딩을 지정하는 정책을 설명합니다. 부트스트랩 채널은 Windows SSPI 전송 보안 및 kerberos 메시지 보안과 함께 TCP 채널 바인딩을 사용합니다.
WS_TCP_SSPI_POLICY_DESCRIPTION Windows SSPI를 사용하여 TCP 채널 바인딩을 지정하는 정책에 대해 설명합니다.
WS_TCP_SSPI_TRANSPORT_SECURITY_BINDING TCP 전송과 함께 Windows 통합 인증 프로토콜(예: Kerberos, NTLM 또는 SPNEGO)의 사용을 지정하기 위한 보안 바인딩 하위 형식입니다.
WS_TCP_SSPI_TRANSPORT_SECURITY_BINDING_CONSTRAINT WS_TCP_SSPI_TRANSPORT_SECURITY_BINDING 해당하는 보안 바인딩 제약 조건입니다.
WS_TCP_SSPI_TRANSPORT_SECURITY_BINDING_TEMPLATE Windows SSPI 프로토콜 기반 전송 보안 사용을 지정하기 위한 보안 바인딩 템플릿입니다.
WS_TCP_SSPI_USERNAME_BINDING_TEMPLATE 애플리케이션에서 입력할 사용자 이름/암호 보안 템플릿 정보입니다. WS_TCP_SSPI_USERNAME_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_TCP_SSPI_USERNAME_POLICY_DESCRIPTION Windows SSPI 전송 보안 및 사용자 이름/암호 메시지 보안을 사용하여 TCP 채널 바인딩을 지정하는 정책을 설명합니다.
WS_TCP_SSPI_USERNAME_SECURITY_CONTEXT_BINDING_TEMPLATE 애플리케이션에서 채울 보안 템플릿 정보입니다. WS_TCP_SSPI_USERNAME_SECURITY_CONTEXT_BINDING_TEMPLATE_TYPE 연결됩니다.
WS_TCP_SSPI_USERNAME_SECURITY_CONTEXT_POLICY_DESCRIPTION

Windows SSPI 전송 보안과 함께 TCP 채널 바인딩을 사용하여 보안 컨텍스트 메시지 바인딩을 지정하는 정책을 설명합니다. 부트스트랩 채널은 Windows SSPI 전송 보안 및 사용자 이름/암호 메시지 보안과 함께 TCP 채널 바인딩을 사용합니다.
WS_THUMBPRINT_CERT_CREDENTIAL 인증서의 지문, 저장소 위치 및 저장소 이름을 사용하여 인증서 자격 증명을 지정하는 형식입니다. 포함된 채널 또는 수신기가 열릴 때 지정된 자격 증명이 로드됩니다.
WS_TIMESPAN 서명된 64비트 시간 간격을 100나노초 단위로 나타냅니다.
WS_TIMESPAN_DESCRIPTION 이 형식 설명은 WS_TIMESPAN_TYPE 사용되며 선택 사항입니다. 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하는 데 사용됩니다.
WS_UINT16_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하기 위해 WS_UINT16_TYPE 사용하는 선택적 형식 설명입니다.
WS_UINT32_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하기 위해 WS_UINT32_TYPE 사용하는 선택적 형식 설명입니다.
WS_UINT64_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하기 위해 WS_UINT64_TYPE 사용하는 선택적 형식 설명입니다.
WS_UINT8_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하기 위해 WS_UINT8_TYPE 사용하는 선택적 형식 설명입니다.
WS_UNION_DESCRIPTION 공용 구조체 형식 내의 선택 항목에 대한 정보입니다. 이는 WS_UNION_TYPE 함께 사용됩니다.
WS_UNION_FIELD_DESCRIPTION 공용 구조체 내의 필드에 대한 serialization 정보를 나타냅니다. WS_UNION_DESCRIPTION 참조하세요.

WS_UNIQUE_ID 고유 ID URI를 나타냅니다.
WS_UNIQUE_ID_DESCRIPTION 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하기 위해 WS_UNIQUE_ID_TYPE 사용하는 선택적 형식 설명입니다.
WS_UNKNOWN_ENDPOINT_IDENTITY 알 수 없는 엔드포인트 ID를 입력합니다. 이 형식은 역직렬화되었지만 이해되지 않은 엔드포인트 ID 형식을 나타내는 데만 사용됩니다.
WS_UPN_ENDPOINT_IDENTITY UPN(사용자 계정 이름)으로 표시되는 엔드포인트 ID를 지정하기 위한 형식입니다.
WS_URL WsDecodeUrl 및 WsEncodeUrl API와 함께 사용되는 모든 URL 구성표에 대한 추상 기본 형식입니다.
WS_USERNAME_CREDENTIAL 모든 사용자 이름/암호 자격 증명에 대한 추상 기본 형식입니다.
WS_USERNAME_MESSAGE_SECURITY_BINDING 애플리케이션 제공 사용자 이름/암호 쌍을 직접(즉, 원샷) 보안 토큰으로 사용하도록 지정하기 위한 보안 바인딩 하위 형식입니다.
WS_USERNAME_MESSAGE_SECURITY_BINDING_CONSTRAINT WS_USERNAME_MESSAGE_SECURITY_BINDING 해당하는 보안 바인딩 제약 조건입니다.
WS_USERNAME_MESSAGE_SECURITY_BINDING_POLICY_DESCRIPTION 이 형식 설명은 입력 정책 설정에 따라 생성된 템플릿을 설명하기 위해 템플릿 API와 함께 사용됩니다. (WS_USERNAME_MESSAGE_SECURITY_BINDING_POLICY_DESCRIPTION)
WS_USERNAME_MESSAGE_SECURITY_BINDING_TEMPLATE 애플리케이션 제공 사용자 이름/암호 쌍을 직접(즉, 원샷) 보안 토큰으로 사용하도록 지정하기 위한 보안 바인딩 템플릿입니다.

WS_UTF8_ARRAY_DESCRIPTION 이 형식 설명은 WS_UTF8_ARRAY_TYPE 사용되며 선택 사항입니다. 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하는 데 사용됩니다.
WS_VOID_DESCRIPTION 직렬화되거나 역직렬화되지 않는 필드에 대한 정보를 지정합니다.
WS_WINDOWS_INTEGRATED_AUTH_CREDENTIAL Windows 통합 인증에 사용되는 모든 자격 증명 형식에 대한 추상 기본 형식입니다.
WS_WSZ_DESCRIPTION 이 형식 설명은 WS_WSZ_TYPE 사용되며 선택 사항입니다. 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하는 데 사용됩니다.
WS_XML_ATTRIBUTE 특성을 나타냅니다(예 <: a:purchaseOrder xmlns:a="http://tempuri.org" id="5">)
WS_XML_BASE64_TEXT base64로 인코딩된 데이터를 나타냅니다.
WS_XML_BOOL_TEXT "true" 또는 "false" 텍스트를 나타내는 부울 값입니다.
WS_XML_BUFFER_PROPERTY XML 버퍼 특정 설정을 지정합니다.
WS_XML_CANONICALIZATION_INCLUSIVE_PREFIXES 단독 XML 정식화 중에 포괄 접두사로 처리해야 하는 XML 접두사 배열입니다. 포괄 접두사 처리는 RFC 3741에 정의되어 있습니다.
WS_XML_CANONICALIZATION_PROPERTY XML 정식화가 수행되는 방법을 제어하는 설정을 지정합니다.
WS_XML_COMMENT_NODE 주석을 나타냅니다.

WS_XML_DATETIME_TEXT xsd:dateTime 형식의 datetime을 나타냅니다.
WS_XML_DECIMAL_TEXT 12 바이트 고정 소수점 값을 나타냅니다.
WS_XML_DICTIONARY 고유한 문자열 집합을 나타냅니다. 이 정보는 이진 인코딩에서 더 압축된 xml 문서를 작성하는 데 사용됩니다.
WS_XML_DOUBLE_TEXT 8 바이트 부동 소수점 값을 나타냅니다.
WS_XML_ELEMENT_NODE xml의 start 요소를 나타냅니다(예:
WS_XML_FLOAT_TEXT 4 바이트 부동 소수점 값을 나타냅니다.
WS_XML_GUID_TEXT "xxxxxxxx-xx" 텍스트로 서식이 지정된 GUID를 나타냅니다.
WS_XML_INT32_TEXT 부제 32비트 정수입니다.
WS_XML_INT64_TEXT 부제 64비트 정수입니다.
WS_XML_LIST_TEXT 단일 공백 문자로 구분된 텍스트 값 목록을 나타냅니다.
WS_XML_NODE 모든 종류의 XML 노드에 대한 기본 형식입니다. XML 노드는 XML의 데이터 단위입니다.
WS_XML_NODE_POSITION XML 버퍼 내의 위치를 나타냅니다.

WS_XML_QNAME XML 이름(요소 또는 특성)을 로컬 이름, 네임스페이스 쌍으로 지정하는 데 사용되는 구조체입니다.
WS_XML_QNAME_DESCRIPTION 이 형식 설명은 WS_XML_QNAME_TYPE 사용되며 선택 사항입니다. 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하는 데 사용됩니다.
WS_XML_QNAME_TEXT "prefix:localName" 텍스트로 서식이 지정된 qname을 나타냅니다.
WS_XML_READER_BINARY_ENCODING 판독기가 읽는 바이트를 이진 xml로 해석해야 함을 나타내는 데 사용됩니다.
WS_XML_READER_BUFFER_INPUT xml 입력의 원본이 버퍼임을 지정합니다.
WS_XML_READER_ENCODING 이 구조체는 다양한 종류의 판독기 인코딩에 대한 기본 형식입니다.
WS_XML_READER_INPUT 판독기가 xml 문서를 구성하는 바이트를 가져올 위치를 지정합니다.
WS_XML_READER_MTOM_ENCODING 판독기가 읽은 바이트를 MTOM 형식으로 해석해야 함을 나타내는 데 사용됩니다.
WS_XML_READER_PROPERTIES WS_XML_READER_PROPERTYs 집합을 지정하는 데 사용되는 구조체입니다.
WS_XML_READER_PROPERTY 판독기별 설정을 지정합니다.
WS_XML_READER_RAW_ENCODING 판독기가 문서의 바이트를 base64로 인코딩된 문자로 표시해야 함을 나타내는 데 사용됩니다.
WS_XML_READER_STREAM_INPUT

콜백에서 xml의 원본을 가져오라고 지정합니다.	
WS_XML_READER_TEXT_ENCODING 판독기가 읽은 바이트를 텍스트 xml로 해석해야 함을 나타내는 데 사용됩니다.	
WS_XML_SECURITY_TOKEN_PROPERTY XML 보안 토큰의 속성을 지정합니다.	
WS_XML_STRING 선택적으로 연결된 사전 정보가 있는 문자열을 나타냅니다. xml API는 WS_XML_STRINGs 사용하여 접두사, localNames 및 네임스페이스를 식별합니다.	
WS_XML_STRING_DESCRIPTION 이 형식 설명은 WS_XML_STRING_TYPE 사용되며 선택 사항입니다. 역직렬화할 수 있는 값 집합에 대한 제약 조건을 지정하는 데 사용됩니다.	
WS_XML_TEXT xml의 텍스트 콘텐츠 노드를 나타냅니다.	
WS_XML_TEXT_NODE 요소, 특성 또는 CDATA 콘텐츠를 나타냅니다.	
WS_XML_TIMESPAN_TEXT "[+ 텍스트로 서식이 지정된 시간 범위를 나타냅니다.	-][d?.]HH:mm:ss[.fffffff]" d는 요일을 나타내는 일련의 숫자입니다.
WS_XML_TOKEN_MESSAGE_SECURITY_BINDING XML 형식으로 애플리케이션에 이미 사용할 수 있는 보안 토큰의 사용을 지정하기 위한 보안 바인딩 하위 형식입니다.	
WS_XML_UINT64_TEXT 부호 없는 64비트 정수입니다.	
WS_XML_UNIQUE_ID_TEXT "urn:uuid:xxxxxxxx-xxxx-xxxx-xxxx-xx" 텍스트로 서식이 지정된 GUID를 나타냅니다.	
WS_XML_UTF16_TEXT	

UTF-16바이트로 인코딩된 텍스트를 나타냅니다.
WS_XML_UTF8_TEXT UTF-8 바이트로 인코딩된 텍스트를 나타냅니다.
WS_XML_WRITER_BINARY_ENCODING 기록기가 바이트를 이진 xml로 내보내야 함을 나타내는 데 사용됩니다.
WS_XML_WRITER_BUFFER_OUTPUT 생성된 바이트를 버퍼에 배치되도록 지정합니다.
WS_XML_WRITER_ENCODING 이 구조체는 모든 종류의 기록기 인코딩에 대한 기본 형식입니다.
WS_XML_WRITER_MTOM_ENCODING 판독기가 MTOM 형식으로 바이트를 내보내야 함을 나타내는 데 사용됩니다. MTOM 형식은 포함된 base64로 인코딩된 텍스트가 아닌 이진 MIME 부분으로 작성된 바이트를 나타냅니다.
WS_XML_WRITER_OUTPUT 작성기에서 xml 문서를 구성하는 바이트를 내보낸 위치를 지정합니다.
WS_XML_WRITER_PROPERTIES WS_XML_WRITER_PROPERTYs 집합을 지정하는 데 사용되는 구조체입니다.
WS_XML_WRITER_PROPERTY 기록기별 설정을 지정합니다.
WS_XML_WRITER_RAW_ENCODING 기록기가 디코딩된 base64 문자에서 바이트를 내보내야 함을 나타내는 데 사용됩니다.
WS_XML_WRITER_STREAM_OUTPUT 생성된 바이트를 콜백으로 보내도록 지정합니다.
WS_XML_WRITER_TEXT_ENCODING 판독기가 바이트를 텍스트 xml로 내보내야 했음을 나타냅니다.

열거형

WS_ADDRESSING_VERSION 주소 지정 헤더에 사용되는 사양의 버전을 식별합니다.
WS_BINDING_TEMPLATE_TYPE 지원되는 다양한 보안 바인딩 조합의 열거형입니다.
WS_CALL_PROPERTY_ID 클라이언트 쪽 서비스 작업에서 호출을 구성하기 위한 선택적 매개 변수입니다.
WS_CALLBACK_MODEL 콜백의 스레딩 동작(예: WS_ASYNC_CALLBACK)을 지정합니다.
WS_CERT_CREDENTIAL_TYPE WS_CERT_CREDENTIAL 하위 형식의 선택기로 사용되는 인증서 자격 증명의 형식입니다.
WS_CHANNEL_BINDING 채널에 사용할 프로토콜 스택을 나타냅니다.
WS_CHANNEL_PROPERTY_ID 각 채널 속성은 ID로 식별되며 연결된 값을 가집니다. 채널을 만들 때 속성을 지정하지 않으면 기본값이 사용됩니다.
WS_CHANNEL_STATE 채널이 있을 수 있는 다양한 상태입니다.
WS_CHANNEL_TYPE 세션인지 여부 및 지원되는 통신 방향과 같은 채널의 기본 특성을 나타냅니다.
WS_CHARSET 문서의 문자 집합을 식별합니다.
WS_COOKIE_MODE HTTP 쿠키를 처리하는 방법을 지정하는 데 사용되는 열거형입니다.
WS_DATETIME_FORMAT

WS_DATETIME 텍스트 형식을 지정합니다.
WS_ENCODING 다양한 인코딩(메시지 형식)입니다.
WS_ENDPOINT_ADDRESS_EXTENSION_TYPE 이는 WS_ENDPOINT_ADDRESS 확장 필드 내의 확장 유형을 식별합니다.
WS_ENDPOINT_IDENTITY_TYPE WS_ENDPOINT_IDENTITY 하위 형식의 선택기로 사용되는 엔드포인트 ID의 형식입니다.
WS_ENVELOPE_VERSION 봉투 구조에 사용되는 사양의 버전입니다.
WS_ERROR_PROPERTY_ID 오류와 연결된 속성 값 집합입니다. WsGetErrorProperty 및 WsSetErrorProperty를 사용하여 설정 및 검색됩니다.
WS_EXCEPTION_CODE 이 구성 요소에서 throw한 구조적 예외 코드입니다. 이러한 예외는 치명적이며 애플리케이션에서 처리해서는 안 됩니다.
WS_EXTENDED_PROTECTION_POLICY 확장된 보호 데이터의 유효성을 검사해야 하는지를 정의합니다.
WS_EXTENDED_PROTECTION_SCENARIO 확장된 보호의 유효성을 검사하는 방법을 정의합니다.
WS_FAULT_DISCLOSURE 오류에 포함된 오류 정보의 양을 제어합니다. 오류 개체는 오류 문자열의 일부로 중요한 데이터를 포함할 수 있으므로 오류 문자열 정보를 모든 오류에 포함하는 것이 항상 적절하지는 않습니다.
WS_FAULT_ERROR_PROPERTY_ID 오류에 대한 정보입니다.
WS_FIELD_MAPPING 구조체의 필드가 XML로 표시되는 방식을 지정합니다. 이는 WS_FIELD_DESCRIPTION 내에서 사용됩니다.

WS_HEADER_TYPE 헤더 유형을 식별합니다.
WS_HEAP_PROPERTY_ID 각 힙 속성은 ID로 식별되며 연결된 값을 가집니다.
WS_HTTP_HEADER_AUTH_TARGET HTTP 헤더 인증 보안 바인딩의 대상을 정의합니다.
WS_HTTP_PROXY_SETTING_MODE 프록시 설정은 바인딩 WS_HTTP_CHANNEL_BINDING 있는 채널에 대한 HTTP 프록시 설정을 나타냅니다. 이는 WS_CHANNEL_PROPERTY_HTTP_PROXY_SETTING_MODE 채널 속성의 일부로 지정됩니다.
WS_IP_VERSION IP 버전을 지정합니다.
WS_LISTENER_PROPERTY_ID 각 수신기 속성은 WS_LISTENER_PROPERTY 형식이며 ID로 식별되며 연결된 값을 가집니다. 수신기를 만들 때 속성을 지정하지 않으면 기본값이 사용됩니다.
WS_LISTENER_STATE 수신기가 있을 수 있는 다른 상태입니다.
WS_MESSAGE_INITIALIZATION WsInitializeMessageshould가 메시지에 추가하는 헤더를 지정합니다.
WS_MESSAGE_PROPERTY_ID 각 메시지 속성은 WS_MESSAGE_PROPERTY 형식이며 ID로 식별되며 연결된 값을 가집니다.
WS_MESSAGE_SECURITY_USAGE 메시지 보안 바인딩이 WS-Security 메커니즘을 사용하여 메시지에 해당하는 보안 토큰을 연결하는 방법을 정의합니다.
WS_MESSAGE_STATE 메시지가 있을 수 있는 다른 상태입니다.

WS_METADATA_EXCHANGE_TYPE 메타데이터 교환 유형을 사용하도록 설정하고 사용하지 않도록 설정하는 방법에 대한 정보입니다.
WS_METADATA_PROPERTY_ID 각 메타데이터 속성은 WS_METADATA_PROPERTY 형식이며 ID로 식별되며 연결된 값을 가집니다. 메타데이터를 만들 때 속성을 지정하지 않으면 기본값이 사용됩니다.
WS_METADATA_STATE 메타데이터 개체의 상태입니다.
WS_MOVE_TO 이 열거형은 xml 문서에 대해 이동하는 다양한 방법을 식별합니다.
WS_OPERATION_CONTEXT_PROPERTY_ID 컨텍스트에서 사용할 수 있는 속성입니다. 컨텍스트의 지정된 지점에서 모든 속성을 사용할 수 있는 것은 아닙니다. 모든 컨텍스트 속성은 WsGetOperationContextProperty를 통해 사용할 수 있습니다.
WS_OPERATION_STYLE 다양한 작업 스타일의 열거형입니다.
WS_PARAMETER_TYPE 다양한 매개 변수 형식입니다.
WS_POLICY_EXTENSION_TYPE 이 열거형의 값은 하위 형식의 WS_POLICY_EXTENSION 식별하는 데 사용됩니다.
WS_POLICY_PROPERTY_ID 각 정책 속성과 관련 값을 식별합니다.
WS_POLICY_STATE 정책 개체의 상태입니다.
WS_PROTECTION_LEVEL 전송 및 수신된 메시지에 필요한 무결성 및 기밀성 수준을 정의합니다.
WS_PROXY_PROPERTY_ID

서비스 프록시를 구성하기 위한 선택적 매개 변수입니다. WS_PROXY_PROPERTY_STATE 제외하면 모든 값은 WS_PROXY_PROPERTY* 매개 변수의 일부로 WsCreateServiceProxy에서만 사용할 수 있습니다.
WS_READ_OPTION 값이 필요한지 여부와 값을 할당하는 방법을 지정합니다.
WS_RECEIVE_OPTION 채널에서 수신할 때 메시지가 필요한지 여부를 지정합니다.
WS_REPEATING_HEADER_OPTION 이 열거형은 헤더가 메시지에 두 번 이상 나타날지 여부를 지정하는 데 사용됩니다.
WS_REQUEST_SECURITY_TOKEN_ACTION WS-Trust를 사용하여 보안 토큰을 협상할 때 사용할 작업 집합을 정의합니다.
WS_REQUEST_SECURITY_TOKEN_PROPERTY_ID 발급자로부터 보안 토큰을 요청하기 위한 속성을 식별합니다. WsRequestSecurityToken과 함께 WS_REQUEST_SECURITY_TOKEN_PROPERTY* 매개 변수의 일부로 사용됩니다.
WS_SAML_AUTHENTICATOR_TYPE 들어오는 SAML 토큰의 유효성을 검사하기 위해 서버 쪽(예: 신뢰 당사자)에서 사용되는 SAML 토큰 인증자의 형식 ID입니다.
WS_SECURE_CONVERSATION_VERSION 메시지 보안 및 혼합 모드 보안에 사용할 WS-SecureConversation 사양 버전을 정의합니다.
WS_SECURITY_ALGORITHM_ID WS-Security와 함께 사용할 보안 알고리즘을 정의합니다. 이러한 값은 메시지 보안 바인딩 및 혼합 모드 보안 바인딩과 관련이 있습니다.
WS_SECURITY_ALGORITHM_PROPERTY_ID 보안 알고리즘 노브를 나타내는 속성을 식별합니다.
WS_SECURITY_ALGORITHM_SUITE_NAME 서명 및 암호화와 같은 작업에 사용되는 보안 알고리즘 모음입니다. 이 열거형의 값은 WS-SecurityPolicy 1.1 섹션 7.1에 정의된 도구 모음에 해당합니다.
WS_SECURITY_BINDING_CONSTRAINT_TYPE

이 열거형의 값은 하위 형식의 WS_SECURITY_BINDING_CONSTRAINT 식별하는 데 사용됩니다.
WS_SECURITY_BINDING_PROPERTY_ID 보안 바인딩 설정을 지정하는 데 사용되는 속성을 식별합니다. 보안 바인딩 설정은 보안 설명에서 사용되는 보안 바인딩에 있습니다.
WS_SECURITY_BINDING_TYPE WS_SECURITY_BINDING 하위 형식의 선택기로 사용되는 보안 바인딩의 형식입니다.
WS_SECURITY_CONTEXT_PROPERTY_ID 보안 컨텍스트 개체의 속성을 식별합니다. 이 열거형은 WsGetSecurityContextProperty와 함께 사용됩니다.
WS_SECURITY_HEADER_LAYOUT WS-Security 보안 헤더의 요소에 적용되는 레이아웃 규칙입니다. 이 설정은 메시지 보안 바인딩 및 혼합 모드 보안 바인딩과 관련이 있습니다.
WS_SECURITY_HEADER_VERSION 메시지 보안 및 혼합 모드 보안에 사용할 WS-Security 사양 버전입니다.
WS_SECURITY_KEY_ENTROPY_MODE 메시지 및 혼합 모드 보안으로 수행된 보안 토큰 협상 중에 발급된 키에 임의성을 기여하는 방법을 정의합니다.
WS_SECURITY_KEY_HANDLE_TYPE 보안 키의 유형입니다.
WS_SECURITY_KEY_TYPE 보안 토큰의 키 형식입니다. 키에 대해 보안 토큰을 쿼리할 때 반환 형식으로 사용됩니다. 보안 토큰 서비스에서 보안 토큰을 요청할 때 필요한 키 형식을 지정하는 데도 사용됩니다.
WS_SECURITY_PROPERTY_ID 채널 전체 보안 설정을 나타내는 속성을 식별합니다. 이 열거형은 WS_SECURITY_PROPERTY 구조 내에서 사용되며, WS_SECURITY_DESCRIPTION 구조 내에서 사용됩니다.
WS_SECURITY_TIMESTAMP_USAGE 메시지 보안 및 혼합 모드 보안을 사용하면 WS-Security 헤더에서 타임스탬프 요소를 생성하고 요구할 시기를 정의합니다.

WS_SECURITY_TOKEN_PROPERTY_ID 보안 토큰에서 추출할 수 있는 필드 및 속성의 키를 정의합니다. 모든 속성이 모든 보안 토큰 형식에 유효하지는 않습니다. WsGetSecurityTokenProperty 함수는 여기에 정의된 값을 키로 사용합니다.
WS_SECURITY_TOKEN_REFERENCE_MODE 메시지 및 혼합 모드 보안 바인딩을 사용하여 서명, 암호화된 항목 및 파생 토큰에서 보안 토큰을 참조하는 데 사용하는 메커니즘입니다.
WS_SERVICE_CANCEL_REASON 취소 이유입니다.
WS_SERVICE_ENDPOINT_PROPERTY_ID 각 속성은 지정된 WS_SERVICE_ENDPOINT 구조를 구성하기 위한 선택적 매개 변수를 나타냅니다. 이 열거형은 WS_SERVICE_ENDPOINT 일부인 WS_SERVICE_ENDPOINT_PROPERTY 구조 내에서 사용됩니다.
WS_SERVICE_HOST_STATE 서비스 호스트가 있을 수 있는 상태입니다.
WS_SERVICE_PROPERTY_ID 서비스 호스트를 구성하기 위한 선택적 매개 변수입니다. 이 열거형은 WsCreateServiceHost를 호출할 때 또는 WsGetServiceHostProperty를 호출할 때 자체에 의해 WS_SERVICE_PROPERTY 구조 내에서 사용됩니다.
WS_SERVICE_PROXY_STATE 서비스 프록시의 상태입니다.
WS_TRACE_API WS_TRACE_API 열거형입니다.
WS_TRANSFER_MODE 전송되거나 수신된 메시지가 스트리밍되거나 버퍼링되는지 여부입니다.
WS_TRUST_VERSION 메시지 보안 및 혼합 모드 보안에 사용할 WS-Trust 사양 버전을 정의합니다.
WS_TYPE

serialization에 지원되는 형식입니다.
WS_TYPE_MAPPING 직렬화되거나 역직렬화될 때 WS_TYPE XML 간 매핑 방법입니다.
WS_URL_SCHEME_TYPE WsDecodeUrl, WsEncodeUrl 및 WsCombineUrl과 함께 사용되는 스키마 집합입니다.
WS_USERNAME_CREDENTIAL_TYPE WS_USERNAME_CREDENTIAL 하위 형식의 선택기로 사용되는 사용자 이름/암호 자격 증명의 형식입니다.
WS_VALUE_TYPE 고정 크기 기본 형식입니다.
WS_WINDOWS_INTEGRATED_AUTH_CREDENTIAL_TYPE WS_WINDOWS_INTEGRATED_AUTH_CREDENTIAL 하위 형식의 선택기로 사용되는 Windows 통합 인증 자격 증명의 형식입니다.
WS_WINDOWS_INTEGRATED_AUTH_PACKAGE Windows 통합 인증에 사용할 특정 SSP 패키지를 정의합니다.
WS_WRITE_OPTION 지정된 스토리지에 값 또는 값에 대한 포인터가 포함되어 있는지 여부와 XML 콘텐츠에서 값을 nil로 나타낼 수 있는지 여부를 지정합니다.
WS_XML_BUFFER_PROPERTY_ID 각 XML 버퍼 속성은 ID로 식별되며 연결된 값을 가집니다.
WS_XML_CANONICALIZATION_ALGORITHM XML 정식화 알고리즘의 값입니다.
WS_XML_CANONICALIZATION_PROPERTY_ID 각 XML 정식화 속성과 관련 값을 식별합니다. 이 열거형은 WsStartReaderCanonicalization 및 WsStartWriterCanonicalization에 대한 매개 변수로 사용되는 WS_XML_CANONICALIZATION_PROPERTY 구조 내에서 사용됩니다.
WS_XML_NODE_TYPE

WS_XML_NODE 구조체의 형식입니다.
WS_XML_READER_ENCODING_TYPE WS_XML_READER_ENCODING 구조체의 형식입니다.
WS_XML_READER_INPUT_TYPE WS_XML_READER_INPUT 구조체의 형식입니다.
WS_XML_READER_PROPERTY_ID 각 XML 판독기 속성이 이고 연결된 값을 식별합니다.
WS_XML_SECURITY_TOKEN_PROPERTY_ID XML 보안 토큰을 만들기 위한 속성 모음의 키입니다. 이 열거형은 WsCreateXmlSecurityToken에 대한 매개 변수로 사용되는 WS_XML_SECURITY_TOKEN_PROPERTY 구조 내에서 사용됩니다.
WS_XML_TEXT_TYPE WS_XML_TEXT 구조체의 형식입니다.
WS_XML_WRITER_ENCODING_TYPE WS_XML_WRITER_ENCODING 구조체의 형식입니다.
WS_XML_WRITER_OUTPUT_TYPE WS_XML_WRITER_OUTPUT 구조체의 형식입니다.
WS_XML_WRITER_PROPERTY_ID 각 xml 기록기 속성은 ID로 식별되며 연결된 값을 가집니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

websocket.h 헤더

아티클2023. 08. 27.

이 헤더는 WebSocket 프로토콜 구성 요소 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- WebSocket 프로토콜 구성 요소 API

websocket.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WebSocketAbortHandle WebSocketCreateClientHandle 또는 WebSocketCreateServerHandle에서 만든 WebSocket 세션 핸들을 중단합니다.
WebSocketBeginClientHandshake 클라이언트 쪽 핸드셰이크를 시작합니다.
WebSocketBeginServerHandshake 서버 쪽 핸드셰이크를 시작합니다.
WebSocketCompleteAction WebSocketGetAction에서 시작한 작업을 완료합니다.
WebSocketCreateClientHandle 클라이언트 쪽 WebSocket 세션 핸들을 만듭니다.
WebSocketCreateServerHandle 서버 쪽 WebSocket 세션 핸들을 만듭니다.
WebSocketDeleteHandle WebSocketCreateClientHandle 또는 WebSocketCreateServerHandle에서 만든 WebSocket 세션 핸들을 삭제합니다.
WebSocketEndClientHandshake 클라이언트 쪽 핸드셰이크를 완료합니다.

WebSocketEndServerHandshake 서버 쪽 핸드셰이크를 완료합니다.
WebSocketGetAction WebSocketSend, WebSocketReceive 또는 WebSocketCompleteAction 호출에서 작업을 반환합니다.
WebSocketGetGlobalProperty 단일 WebSocket 속성을 가져옵니다.
WebSocketReceive 프로토콜 구성 요소 작업 큐에 수신 작업을 추가합니다.
WebSocketSend 프로토콜 구성 요소 작업 큐에 보내기 작업을 추가합니다.

구조체

WEB_SOCKET_BUFFER 특정 WebSocket 작업에 대한 데이터를 포함합니다.
WEB_SOCKET_HTTP_HEADER HTTP 헤더를 포함합니다.
WEB_SOCKET_PROPERTY 단일 WebSocket 속성을 포함합니다.

열거형

WEB_SOCKET_ACTION WebSocket 애플리케이션에서 수행할 작업을 지정합니다.

WEB_SOCKET_ACTION_QUEUE WebSocketGetAction에서 반환된 작업 유형을 지정합니다.
WEB_SOCKET_BUFFER_TYPE WebSocket 프레임 헤더를 생성하는 데 사용되는 비트 값을 지정합니다.
WEB_SOCKET_CLOSE_STATUS WebSocket 닫기 상태 지정합니다.
WEB_SOCKET_PROPERTY_TYPE WebSocket 속성 형식을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

werapi.h 헤더

아티클2023. 07. 27.

이 헤더는 WER(Windows 오류 보고)에서 사용됩니다.

werapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WerAddExcludedApplication 오류 보고에서 제외할 애플리케이션 목록에 지정된 애플리케이션을 추가합니다.
WerFreeString 보고서 키 문자열을 저장하는 데 사용되는 메모리를 해제합니다. 특정 보고서 키 문자열이 사용되고 더 이상 필요하지 않은 경우 WerStoreGetFirstReportKey 또는 WerStoreGetNextReportKey에 대한 각 연속 호출 후에 호출해야 합니다.
WerGetFlags 지정된 프로세스에 대한 오류 보고 설정을 검색합니다.
WerRegisterAdditionalProcess 기본 애플리케이션 프로세스와 함께 WER(Windows 오류 보고) 보고서에 포함할 프로세스를 등록합니다. 필요에 따라 추가 데이터를 가져올 등록된 프로세스 내의 스레드를 지정합니다.
WerRegisterAppLocalDump 애플리케이션에 대한 프로세스 중 하나가 응답을 중지할 때 WER(Windows 오류 보고)이 수집하는 진단 메모리 덤프의 복사본을 저장해야 하는 경로(패키지된 애플리케이션의 LocalFolder를 기준으로)를 등록합니다.
WerRegisterCustomMetadata WER(Windows 오류 보고) 오류 보고서에 대해 수집할 앱별 메타데이터(키/값 문자열 형식)를 등록합니다.
WerRegisterExcludedMemoryBlock WER(Windows 오류 보고) 오류 보고서에서 제외할 메모리 블록(일반적으로 오류 보고서에 기본적으로 포함됨)을 표시합니다.

WerRegisterFile WER(Windows 오류 보고)이 오류 보고서를 만들 때 수집할 파일을 등록합니다.
WerRegisterMemoryBlock WER(Windows 오류 보고)이 오류 보고서를 만들 때 수집할 메모리 블록을 등록합니다.
WerRegisterRuntimeExceptionModule 크래시에 대한 WER(사용자 지정 Windows 오류 보고)을 제공하는 데 사용되는 사용자 지정 런타임 예외 처리기를 등록합니다.
WerRemoveExcludedApplication WER(Windows 오류 보고)에서 제외할 애플리케이션 목록에서 지정된 애플리케이션을 제거합니다.
WerReportAddDump 지정된 형식의 덤프를 지정된 WER(Windows 오류 보고) 보고서에 추가합니다.
WerReportAddFile 지정된 WER(Windows 오류 보고) 보고서에 파일을 추가합니다.
WerReportCloseHandle 지정된 WER(Windows 오류 보고) 보고서를 닫습니다.
WerReportCreate 애플리케이션 이벤트를 설명하는 WER(Windows 오류 보고) 보고서를 만듭니다.
WerReportSetParameter 지정된 WER(Windows 오류 보고) 보고서에 대한 이벤트를 고유하게 식별하는 매개 변수를 설정합니다.
WerReportSetUIOption 지정된 WER(Windows 오류 보고) 보고서에 대한 사용자 인터페이스 옵션을 설정합니다.
WerReportSubmit 지정된 WER(Windows 오류 보고) 보고서를 제출합니다.
WerSetFlags 현재 프로세스에 대한 WER(Windows 오류 보고) 보고 설정을 설정합니다.

WerSetMaxProcessHoldMilliseconds WER(Windows 오류 보고)의 최대 프로세스 보류 시간을 밀리초 단위로 설정합니다.
WerStoreClose 저장된 WER(Windows 오류 보고) 보고서의 컬렉션을 닫습니다.
WerStoreGetFirstReportKey 보고서 저장소의 첫 번째 WER(Windows 오류 보고) 보고서에 대한 참조를 가져옵니다.
WerStoreGetNextReportKey 오류 보고서 저장소의 다음 WER(Windows 오류 보고) 보고서에 대한 참조를 가져옵니다.
WerStoreGetReportCount 오류 보고서 저장소의 WER(Windows 오류 보고) 보고서 수를 가져옵니다.
WerStoreGetSizeOnDisk WER(Windows 오류 보고) 오류 보고서 저장소의 크기를 바이트 단위로 가져옵니다.
WerStoreOpen 저장된 WER(Windows 오류 보고) 오류 보고서의 컬렉션을 엽니다.
WerStorePurge WER(Windows 오류 보고) 보고서 저장소를 제거합니다.
WerStoreQueryReportMetadataV1 저장소의 WER(Windows 오류 보고) 보고서에 대한 메타데이터를 검색합니다.
WerStoreQueryReportMetadataV2 저장소의 WER(Windows 오류 보고) 보고서에 대한 메타데이터를 검색합니다.
WerStoreQueryReportMetadataV3 저장소의 WER(Windows 오류 보고) 보고서에 대한 메타데이터를 검색합니다.
WerStoreUploadReport WER(Windows 오류 보고) 저장소에 보고서를 업로드합니다.
WerUnregisterAdditionalProcess

WER(Windows 오류 보고) 오류 보고서에 포함할 추가 프로세스 목록에서 프로세스를 제거합니다.
WerUnregisterAppLocalDump WerRegisterAppLocalDump 함수를 호출하여 WER(Windows 오류 보고)이 애플리케이션에 대한 프로세스 중 하나가 응답을 중지할 때 수집하는 진단 메모리 덤프의 복사본을 저장하도록 지정하여 만든 등록을 취소합니다.
WerUnregisterCustomMetadata 애플리케이션에 대한 WER(Windows 오류 보고) 중에 수집되는 앱별 메타데이터의 항목을 제거합니다.
WerUnregisterExcludedMemoryBlock 이전에 제외된 것으로 표시된 메모리 블록을 제거합니다. 이 블록은 WER(Windows 오류 보고] 오류 보고서에 다시 포함됩니다.
WerUnregisterFile 현재 프로세스에 대해 생성된 WER(Windows 오류 보고) 보고서에 추가할 파일 목록에서 파일을 제거합니다.
WerUnregisterMemoryBlock 애플리케이션에 대한 WER(Windows 오류 보고) 중에 수집할 데이터 목록에서 메모리 블록을 제거합니다.
WerUnregisterRuntimeExceptionModule WER(Windows 오류 보고) 예외 처리기의 등록을 제거합니다.

콜백 함수

PFN_WER_RUNTIME_EXCEPTION_DEBUGGER_LAUNCH WER(Windows 오류 보고)은 이 함수를 호출하여 디버거 시작 옵션 및 시작 문자열을 사용자 지정할 수 있도록 합니다.
PFN_WER_RUNTIME_EXCEPTION_EVENT WER(Windows 오류 보고)은 이 함수를 호출하여 예외 처리기가 충돌을 클레임하는지 여부를 확인합니다.

PFN_WER_RUNTIME_EXCEPTION_EVENT_SIGNATURE WER(Windows 오류 보고)은 이 함수를 호출하여 문제를 고유하게 설명하는 보고서 매개 변수를 가져옵니다.

구조체

WER_DUMP_CUSTOM_OPTIONS WerReportAddDump 함수에서 수집할 WER(사용자 지정 Windows 오류 보고) 미니덤프 정보를 지정합니다.
WER_DUMP_CUSTOM_OPTIONS_V2 WerReportAddDump 함수에서 수집할 WER(사용자 지정 Windows 오류 보고) 미니덤프 정보를 지정합니다.
WER_DUMP_CUSTOM_OPTIONS_V3 PssCaptureSnapshot 함수에 의해 백그라운드에서 수집할 WER(사용자 지정 Windows 오류 보고) 미니덤프 정보를 지정합니다(프로세스를 일시 중지하지 않음).
WER_EXCEPTION_INFORMATION WerReportAddDump 함수에 대한 WER(Windows 오류 보고) 예외 정보를 포함합니다.
WER_REPORT_INFORMATION WerReportCreate 함수에서 사용하는 WER(Windows 오류 보고) 정보를 포함합니다.
WER_REPORT_INFORMATION_V3 WerReportCreate 함수에서 사용하는 WER(Windows 오류 보고) 정보를 포함합니다.
WER_REPORT_INFORMATION_V4 WerReportCreate 함수에서 사용하는 WER(Windows 오류 보고] 정보를 포함합니다.
WER_REPORT_INFORMATION_V5 WerReportCreate 함수에서 사용하는 WER(Windows 오류 보고) 정보를 포함합니다.
WER_REPORT_METADATA_V1 WER(Windows 오류 보고)에서 생성된 오류 보고서에 대한 정보를 포함합니다.

WER_REPORT_METADATA_V2 WER(Windows 오류 보고)에서 생성된 오류 보고서에 대한 정보를 포함합니다.
WER_REPORT_METADATA_V3 WER(Windows 오류 보고)에서 생성된 오류 보고서에 대한 정보를 포함합니다.
WER_REPORT_PARAMETER WER(Windows 오류 보고)에서 생성된 오류 보고서 매개 변수에 대한 정보를 포함합니다.
WER_REPORT_SIGNATURE WER(Windows 오류 보고)에서 생성된 오류 보고서 서명에 대한 정보를 포함합니다.
WER_RUNTIME_EXCEPTION_INFORMATION 충돌을 클레임할지 여부를 결정하는 데 사용되는 WER(Windows 오류 보고) 예외 정보를 포함합니다.

열거형

REPORT_STORE_TYPES 열 수 있는 WER(Windows 오류 보고) 보고서 저장소의 유형을 정의합니다.
WER_CONSENT WER(가능한 Windows 오류 보고) 사용자 동의 상태를 정의합니다.
WER_DUMP_TYPE WER(가능한 Windows 오류 보고) 미니덤프 형식을 정의합니다.
WER_FILE_TYPE 미니덤프 보고서에 사용할 수 있는 WER(Windows 오류 보고) 파일 형식을 정의합니다.
WER_REGISTER_FILE_TYPE 등록할 수 있는 WER(Windows 오류 보고) 파일 형식을 정의합니다.
WER_REPORT_TYPE

WER(Windows 오류 보고) 보고서 형식을 정의합니다.
WER_REPORT_UI WER(Windows 오류 보고) 보고서의 사용자 지정 가능한 UI 문자열의 인덱스를 정의합니다.
WER_SUBMIT_RESULT WER(Windows 오류 보고) 제출 결과 옵션을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winbase.h 헤더

아티클2023. 03. 08.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 애플리케이션 설치 및 서비스
- 애플리케이션 복구 및 다시 시작
- Backup
- 데이터 액세스 및 스토리지
- 데이터 교환
- 개발자 정보
- eventlogprov
- 하드웨어 카운터 프로파일링
- Windows 애플리케이션에 대한 국제화
- 메뉴 및 기타 리소스
- 작업 레코더
- 원격 데스크톱 서비스
- 보안 및 ID
- 시스템 서비스
- 창 스테이션 및 데스크톱
- Windows 및 메시지

winbase.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

_lclose _lclose 함수는 지정된 파일을 닫아 더 이상 읽기 또는 쓰기에 사용할 수 없습니다. 이 함수는 16비트 버전의 Windows와의 호환성을 위해 제공됩니다. Win32 기반 애플리케이션은 CloseHandle 함수를 사용해야 합니다.
_lcreat 지정된 파일을 만들거나 엽니다.
_llseek 지정된 파일의 파일 포인터 위치를 변경합니다.

_lopen _lopen 함수는 기존 파일을 열고 파일 포인터를 파일의 시작 부분으로 설정합니다. 이 함수는 16비트 버전의 Windows와의 호환성을 위해 제공됩니다. Win32 기반 애플리케이션은 CreateFile 함수를 사용해야 합니다.
_lread _lread 함수는 지정된 파일에서 데이터를 읽습니다. 이 함수는 16비트 버전의 Windows와의 호환성을 위해 제공됩니다. Win32 기반 애플리케이션은 ReadFile 함수를 사용해야 합니다.
_lwrite 지정된 파일에 데이터를 씁니다.
AccessCheckAndAuditAlarmA 보안 설명자가 호출 스레드에서 가장하는 클라이언트에 대해 지정된 액세스 권한 집합을 부여할지 여부를 결정합니다. (AccessCheckAndAuditAlarmA)
AccessCheckByTypeAndAuditAlarmA 보안 설명자가 호출 스레드에서 가장하는 클라이언트에 대해 지정된 액세스 권한 집합을 부여할지 여부를 결정합니다. (AccessCheckByTypeAndAuditAlarmA)
AccessCheckByTypeResultListAndAuditAlarmA 보안 설명자가 호출 스레드에서 가장하는 클라이언트에 대해 지정된 액세스 권한 집합을 부여할지 여부를 결정합니다. (AccessCheckByTypeResultListAndAuditAlarmA)
AccessCheckByTypeResultListAndAuditAlarmByHandleA AccessCheckByTypeResultListAndAuditAlarmByHandleA(ANSI) 함수(winbase.h)는 보안 설명자가 호출 스레드가 가장하는 클라이언트에 지정된 액세스 권한 집합을 부여할지 여부를 결정합니다.
ActivateActCtx ActivateActCtx 함수는 지정된 활성화 컨텍스트를 활성화합니다.
AddAtomA 로컬 원자 테이블에 문자열을 추가하고 문자열을 식별하는 고유 값(원자)을 반환합니다. (ANSI)
AddAtomW 로컬 원자 테이블에 문자열을 추가하고 문자열을 식별하는 고유 값(원자)을 반환합니다. (유니코드)

AddConditionalAce 지정된 ACL(액세스 제어 목록)에 ACE(조건부 액세스 제어 항목)를 추가합니다.
AddIntegrityLabelToBoundaryDescriptor 지정된 경계 설명자에 새 필수 SID(보안 식별자)를 추가합니다.
AddRefActCtx AddRefActCtx 함수는 지정된 활성화 컨텍스트의 참조 수를 증분합니다.
AddSecureMemoryCacheCallback 보안 메모리 범위가 해제되거나 보호가 변경될 때 호출할 콜백 함수를 등록합니다.
ApplicationRecoveryFinished 호출하는 애플리케이션이 데이터 복구를 완료했음을 나타냅니다.
ApplicationRecoveryInProgress 호출하는 애플리케이션이 계속해서 데이터를 복구하고 있음을 나타냅니다.
BackupEventLogA 지정된 이벤트 로그를 백업 파일에 저장합니다. (ANSI)
BackupEventLogW 지정된 이벤트 로그를 백업 파일에 저장합니다. (유니코드)
BackupRead 보안 정보를 포함하여 파일 또는 디렉터리를 백업합니다.
BackupSeek BackupRead 또는 BackupWrite 함수를 사용하여 처음에 액세스한 데이터 스트림에서 전달을 찾습니다.
BackupWrite BackupRead를 사용하여 백업된 파일 또는 디렉터리를 복원합니다.
BeginUpdateResourceA UpdateResource 함수에서 이진 모듈에서 리소스를 추가, 삭제 또는 바꾸는 데 사용할 수 있는 핸들을 검색합니다. (ANSI)

BeginUpdateResourceW UpdateResource 함수에서 이진 모듈에서 리소스를 추가, 삭제 또는 바꾸는 데 사용할 수 있는 핸들을 검색합니다. (유니코드)
BindIoCompletionCallback 스레드 풀이 소유한 I/O 완료 포트를 지정된 파일 핸들과 연결합니다. 이 파일과 관련된 I/O 요청이 완료되면 I/O가 아닌 작업자 스레드가 지정된 콜백 함수를 실행합니다.
BuildCommDCBA 지정된 DCB 구조체를 디바이스 제어 문자열에 지정된 값으로 채웁니다. (ANSI)
BuildCommDCBAndTimeoutsA 디바이스 정의 문자열을 적절한 디바이스 제어 블록 코드로 변환하고 디바이스 제어 블록에 배치합니다. (ANSI)
BuildCommDCBAndTimeoutsW 디바이스 정의 문자열을 적절한 디바이스 제어 블록 코드로 변환하고 디바이스 제어 블록에 배치합니다. (유니코드)
BuildCommDCBW 지정된 DCB 구조체를 디바이스 제어 문자열에 지정된 값으로 채웁니다. (유니코드)
CallNamedPipeA 메시지 형식 파이프에 연결하고 파이프 인스턴스를 사용할 수 없는 경우 대기하고 파이프에 쓰고 파이프에서 읽은 다음 파이프를 닫습니다. (CallNamedPipeA)
CheckNameLegalDOS8Dot3A 지정된 이름을 사용하여 FAT 파일 시스템에 파일을 만들 수 있는지 여부를 결정합니다. (ANSI)
CheckNameLegalDOS8Dot3W 지정된 이름을 사용하여 FAT 파일 시스템에 파일을 만들 수 있는지 여부를 결정합니다. (유니코드)
ClearCommBreak 지정된 통신 디바이스에 대한 문자 전송을 복원하고 전송 라인을 끊지 않는 상태로 만듭니다.
ClearCommError 통신 오류에 대한 정보를 검색하고 통신 디바이스의 현재 상태를 보고합니다.

ClearEventLogA 지정된 이벤트 로그를 지우고 필요에 따라 로그의 현재 복사본을 백업 파일에 저장합니다. (ANSI)
ClearEventLogW 지정된 이벤트 로그를 지우고 필요에 따라 로그의 현재 복사본을 백업 파일에 저장합니다. (유니코드)
CloseEncryptedFileRaw 백업 또는 복원 작업 후 암호화된 파일을 닫고 연결된 시스템 리소스를 해제합니다.
CloseEventLog 지정된 이벤트 로그를 닫습니다. (CloseEventLog)
CommConfigDialogA 드라이버 제공 구성 대화 상자를 표시합니다. (ANSI)
CommConfigDialogW 드라이버 제공 구성 대화 상자를 표시합니다. (유니코드)
ConvertFiberToThread 현재 파이버를 스레드로 변환합니다.
ConvertThreadToFiber 현재 스레드를 파이버로 변환합니다. 다른 파이버를 예약하려면 먼저 스레드를 파이버로 변환해야 합니다. (ConvertThreadToFiber)
ConvertThreadToFiberEx 현재 스레드를 파이버로 변환합니다. 다른 파이버를 예약하려면 먼저 스레드를 파이버로 변환해야 합니다. (ConvertThreadToFiberEx)
CopyContext 원본 컨텍스트 구조(XState 포함)를 초기화된 대상 컨텍스트 구조에 복사합니다.
CopyFile CopyFile 함수(winbase.h)는 기존 파일을 새 파일에 복사합니다.

CopyFile2 기존 파일을 새 파일에 복사하여 콜백 함수를 통해 애플리케이션에 진행 상황을 알립니다. (CopyFile2)
CopyFileA 새 파일에 기존 파일을 복사합니다. (CopyFileA)
CopyFileExA 기존 파일을 새 파일에 복사하여 콜백 함수를 통해 애플리케이션에 진행 상황을 알립니다. (CopyFileExA)
CopyFileExW 기존 파일을 새 파일에 복사하여 콜백 함수를 통해 애플리케이션에 진행 상황을 알립니다. (CopyFileExW)
CopyFileTransactedA 기존 파일을 트랜잭션 작업으로 새 파일에 복사하여 콜백 함수를 통해 애플리케이션에 진행 상황을 알립니다. (ANSI)
CopyFileTransactedW 기존 파일을 트랜잭션 작업으로 새 파일에 복사하여 콜백 함수를 통해 애플리케이션에 진행 상황을 알립니다. (유니코드)
CopyFileW CopyFileW(유니코드) 함수(winbase.h)는 기존 파일을 새 파일에 복사합니다.
CreateActCtxA CreateActCtx 함수는 활성화 컨텍스트를 만듭니다. (ANSI)
CreateActCtxW CreateActCtx 함수는 활성화 컨텍스트를 만듭니다. (유니코드)
CreateBoundaryDescriptorA ANSI(CreateBoundaryDescriptorA) 함수(winbase.h)는 경계 설명자를 만듭니다.
CreateDirectory CreateDirectory 함수(winbase.h)는 새 디렉터리를 만듭니다.

CreateDirectoryExA 지정된 템플릿 디렉터리의 특성을 사용하여 새 디렉터리를 만듭니다. (ANSI)
CreateDirectoryExW 지정된 템플릿 디렉터리의 특성을 사용하여 새 디렉터리를 만듭니다. (유니코드)
CreateDirectoryTransactedA 지정된 템플릿 디렉터리의 특성을 사용하여 트랜잭션 작업으로 새 디렉터리를 만듭니다. (ANSI)
CreateDirectoryTransactedW 지정된 템플릿 디렉터리의 특성을 사용하여 트랜잭션 작업으로 새 디렉터리를 만듭니다. (유니코드)
CreateFiber 파이버 개체를 할당하고, 스택을 할당하고, 지정된 시작 주소(일반적으로 파이버 함수)에서 시작하도록 실행을 설정합니다. 이 함수는 파이버를 예약하지 않습니다. (CreateFiber)
CreateFiberEx 파이버 개체를 할당하고, 스택을 할당하고, 지정된 시작 주소(일반적으로 파이버 함수)에서 시작하도록 실행을 설정합니다. 이 함수는 파이버를 예약하지 않습니다. (CreateFiberEx)
CreateFileMappingA 지정된 파일에 대한 명명되거나 명명되지 않은 파일 매핑 개체를 만들거나 엽니다. (CreateFileMappingA)
CreateFileMappingNumaA 지정된 파일에 대한 명명되거나 명명되지 않은 파일 매핑 개체를 만들거나 열고 실제 메모리에 대한 NUMA 노드를 지정합니다. (CreateFileMappingNumaA)
CreateFileTransactedA 파일, 파일 스트림 또는 디렉터리를 트랜잭션 작업으로 만들거나 엽니다. (ANSI)
CreateFileTransactedW 파일, 파일 스트림 또는 디렉터리를 트랜잭션 작업으로 만들거나 엽니다. (유니코드)
CreateHardLinkA 기존 파일과 새 파일 간에 하드 링크를 설정합니다. (ANSI)

CreateHardLinkTransactedA 트랜잭션 작업으로 기존 파일과 새 파일 간에 하드 링크를 설정합니다. (ANSI)
CreateHardLinkTransactedW 트랜잭션 작업으로 기존 파일과 새 파일 간에 하드 링크를 설정합니다. (유니코드)
CreateHardLinkW 기존 파일과 새 파일 간에 하드 링크를 설정합니다. (유니코드)
CreateJobObjectA 작업 개체를 만들거나 엽니다. (CreateJobObjectA)
CreateMailslotA 지정된 이름의 mailslot을 만들고 mailslot 서버가 mailslot에서 작업을 수행하는 데 사용할 수 있는 핸들을 반환합니다. (ANSI)
CreateMailslotW 지정된 이름의 mailslot을 만들고 mailslot 서버가 mailslot에서 작업을 수행하는 데 사용할 수 있는 핸들을 반환합니다. (유니코드)
CreateNamedPipeA ANSI(CreateNamedPipeA) 함수(winbase.h)는 명명된 파이프의 인스턴스를 만들고 후속 파이프 작업에 대한 핸들을 반환합니다.
CreatePrivateNamespaceA ANSI(CreatePrivateNamespaceA) 함수(winbase.h)는 프라이빗 네임스페이스를 만듭니다.
CreateProcessWithLogonW 새 프로세스와 해당 기본 스레드를 만듭니다. 그런 다음, 새 프로세스는 지정된 자격 증명(사용자, 도메인 및 암호)의 보안 컨텍스트에서 지정된 실행 파일을 실행합니다. 필요에 따라 지정된 사용자에 대한 사용자 프로필을 로드할 수 있습니다.
CreateProcessWithTokenW 새 프로세스와 해당 기본 스레드를 만듭니다. 새 프로세스는 지정된 토큰의 보안 컨텍스트에서 실행됩니다. 필요에 따라 지정된 사용자에 대한 사용자 프로필을 로드할 수 있습니다.
CreateSemaphoreA 명명되거나 명명되지 않은 세마포 개체를 만들거나 엽니다. (CreateSemaphoreA)

CreateSemaphoreExA 명명되거나 명명되지 않은 세마포 개체를 만들거나 열고 개체에 대한 핸들을 반환합니다. (CreateSemaphoreExA)
CreateSymbolicLinkA 기호화 된 링크를 만듭니다. (ANSI)
CreateSymbolicLinkTransactedA 트랜잭션된 작업으로 기호 링크를 만듭니다. (ANSI)
CreateSymbolicLinkTransactedW 트랜잭션된 작업으로 기호 링크를 만듭니다. (유니코드)
CreateSymbolicLinkW 기호화 된 링크를 만듭니다. (유니코드)
CreateTapePartition 테이프의 서식을 다시 지정합니다.
CreateUmsCompletionList UMS(사용자 모드 예약) 완료 목록을 만듭니다.
CreateUmsThreadContext UMS 작업자 스레드를 나타내는 UMS(사용자 모드 예약) 스레드 컨텍스트를 만듭니다.
DeactivateActCtx DeactivateActCtx 함수는 지정된 쿠키에 해당하는 활성화 컨텍스트를 비활성화합니다.
DebugBreakProcess 지정된 프로세스에서 중단점 예외가 발생하도록 합니다. 이렇게 하면 호출 스레드가 디버거가 예외를 처리하도록 신호를 보냅니다.
DebugSetProcessKillOnExit 호출 스레드가 종료될 때 수행할 작업을 설정합니다.
DecryptFileA 암호화된 파일 또는 디렉터리의 암호를 해독합니다. (ANSI)

DecryptFileW 암호화된 파일 또는 디렉터리의 암호를 해독합니다. (유니코드)
DefineDosDeviceA MS-DOS 디바이스 이름을 정의, 재정의 또는 삭제합니다. (DefineDosDeviceA)
DeleteAtom 로컬 문자열 원자의 참조 수를 줄입니다. 원자의 참조 수가 0으로 줄면 DeleteAtom은 로컬 원자 테이블에서 원자와 연결된 문자열을 제거합니다.
DeleteFiber 기존 파이버를 삭제합니다.
DeleteFile DeleteFile 함수(winbase.h)는 기존 파일을 삭제합니다.
DeleteFileTransactedA 기존 파일을 트랜잭션 작업으로 삭제합니다. (ANSI)
DeleteFileTransactedW 기존 파일을 트랜잭션 작업으로 삭제합니다. (유니코드)
DeleteUmsCompletionList 지정된 UMS(사용자 모드 예약) 완료 목록을 삭제합니다. 목록은 비어 있어야 합니다.
DeleteUmsThreadContext 지정된 UMS(사용자 모드 예약) 스레드 컨텍스트를 삭제합니다. 스레드를 종료해야 합니다.
DeleteVolumeMountPointA 드라이브 문자 또는 탑재된 폴더를 삭제합니다. (DeleteVolumeMountPointA)
DequeueUmsCompletionListItems 지정된 UMS 완료 목록에서 UMS(사용자 모드 예약) 작업자 스레드를 검색합니다.
DeregisterEventSource 지정된 이벤트 로그를 닫습니다. (DeregisterEventSource)

DestroyThreadpoolEnvironment 지정된 콜백 환경을 삭제합니다. 새 스레드 풀 개체를 만드는 데 콜백 환경이 더 이상 필요하지 않은 경우 이 함수를 호출합니다. (DestroyThreadpoolEnvironment)
DisableThreadProfiling 스레드 프로파일링을 사용하지 않도록 설정합니다.
DnsHostnameToComputerNameA DNS 스타일 호스트 이름을 NetBIOS 스타일의 컴퓨터 이름으로 변환합니다. (ANSI)
DnsHostnameToComputerNameW DNS 스타일 호스트 이름을 NetBIOS 스타일의 컴퓨터 이름으로 변환합니다. (유니코드)
DosDateTimeToFileTime MS-DOS 날짜 및 시간 값을 파일 시간으로 변환합니다.
EnableProcessOptionalXStateFeatures EnableProcessOptionalXStateFeatures 함수를 사용하면 현재 프로세스에 대한 선택적 XState 기능 집합을 사용할 수 있습니다.
EnableThreadProfiling 지정된 스레드에서 스레드 프로파일링을 사용하도록 설정합니다.
EncryptFileA 파일 또는 디렉터리를 암호화합니다. (ANSI)
EncryptFileW 파일 또는 디렉터리를 암호화합니다. (유니코드)
EndUpdateResourceA UpdateResource를 호출하기 전에 변경한 내용을 커밋하거나 삭제합니다. (ANSI)
EndUpdateResourceW UpdateResource를 호출하기 전에 변경한 내용을 커밋하거나 삭제합니다. (유니코드)
EnterUmsSchedulingMode 호출 스레드를 UMS(사용자 모드 예약) 스케줄러 스레드로 변환합니다.

EnumResourceLanguagesA 이진 모듈과 연결된 지정된 형식 및 이름의 언어별 리소스를 열거합니다. (ANSI)
EnumResourceLanguagesW 이진 모듈과 연결된 지정된 형식 및 이름의 언어별 리소스를 열거합니다. (유니코드)
EnumResourceTypesA 이진 모듈 내에서 리소스 종류를 열거합니다. (ANSI)
EnumResourceTypesW 이진 모듈 내에서 리소스 종류를 열거합니다. (유니코드)
EraseTape 테이프의 전체 또는 일부를 지웁니다.
EscapeCommFunction 지정된 통신 디바이스에 확장 함수를 수행하도록 지시합니다.
ExecuteUmsThread 지정된 UMS 작업자 스레드를 실행합니다.
FatalExit 실행 컨트롤을 디버거로 전송합니다. 이후 디버거의 동작은 사용되는 디버거 유형에 따라 다릅니다.
FileEncryptionStatusA 지정된 파일의 암호화 상태를 검색합니다. (ANSI)
FileEncryptionStatusW 지정된 파일의 암호화 상태를 검색합니다. (유니코드)
FileTimeToDosDateTime 파일 시간을 MS-DOS 날짜 및 시간 값으로 변환합니다.
FindActCtxSectionGuid FindActCtxSectionGuid 함수는 현재 활성화 컨텍스트에서 특정 GUID에 대한 정보를 검색하고 ACTCTX_SECTION_KEYED_DATA 구조를 반환합니다.

FindActCtxSectionStringA FindActCtxSectionString 함수는 현재 활성화 컨텍스트의 특정 문자열에 대한 정보를 검색하고 ACTCTX_SECTION_KEYED_DATA 구조를 반환합니다. (ANSI)
FindActCtxSectionStringW FindActCtxSectionString 함수는 현재 활성화 컨텍스트의 특정 문자열에 대한 정보를 검색하고 ACTCTX_SECTION_KEYED_DATA 구조를 반환합니다. (유니코드)
FindAtomA 로컬 원자 테이블에서 지정된 문자열을 검색하고 해당 문자열과 연결된 원자를 검색합니다. (ANSI)
FindAtomW 로컬 원자 테이블에서 지정된 문자열을 검색하고 해당 문자열과 연결된 원자를 검색합니다. (유니코드)
FindFirstFileNameTransactedW 지정된 파일에 대한 모든 하드 링크의 열거형을 트랜잭션 작업으로 만듭니다. 함수는 FindNextFileNameW 함수에 대한 후속 호출에 사용할 수 있는 열거형에 대한 핸들을 반환합니다.
FindFirstFileTransactedA 디렉터리에서 특정 이름과 일치하는 이름을 트랜잭션 작업으로 사용하여 파일 또는 하위 디렉터리를 검색합니다. (ANSI)
FindFirstFileTransactedW 디렉터리에서 특정 이름과 일치하는 이름을 트랜잭션 작업으로 사용하여 파일 또는 하위 디렉터리를 검색합니다. (유니코드)
FindFirstStreamTransactedW 지정된 파일 또는 디렉터리의 첫 번째 스트림을 트랜잭션 작업으로 열거합니다.
FindFirstVolumeA 컴퓨터에서 볼륨의 이름을 검색합니다. (FindFirstVolumeA)
FindFirstVolumeMountPointA 지정된 볼륨에 탑재된 폴더의 이름을 검색합니다. (ANSI)
FindFirstVolumeMountPointW 지정된 볼륨에 탑재된 폴더의 이름을 검색합니다. (유니코드)

FindNextVolumeA FindFirstVolume 함수를 호출하여 시작된 볼륨 검색을 계속합니다. (FindNextVolumeA)
FindNextVolumeMountPointA FindFirstVolumeMountPoint 함수를 호출하여 시작된 탑재된 폴더 검색을 계속합니다. (ANSI)
FindNextVolumeMountPointW FindFirstVolumeMountPoint 함수를 호출하여 시작된 탑재된 폴더 검색을 계속합니다. (유니코드)
FindResourceA 지정된 모듈에서 지정된 형식과 이름을 가진 리소스의 위치를 결정합니다. (FindResourceA)
FindResourceExA 지정된 모듈에서 지정된 형식, 이름 및 언어를 사용하여 리소스의 위치를 결정합니다. (FindResourceExA)
FindVolumeMountPointClose 지정된 탑재된 폴더 검색 핸들을 닫습니다.
FormatMessage FormatMessage 함수(winbase.h)는 메시지 문자열의 형식을 지정합니다.
FormatMessageA 메시지 문자열의 서식을 지정합니다. (FormatMessageA)
FormatMessageW FormatMessageW(유니코드) 함수(winbase.h)는 메시지 문자열의 형식을 지정합니다.
GetActiveProcessorCount 프로세서 그룹 또는 시스템의 활성 프로세서 수를 반환합니다.
GetActiveProcessorGroupCount 시스템의 활성 프로세서 그룹 수를 반환합니다.
GetApplicationRecoveryCallback 지정된 프로세스에 대해 등록된 콜백 루틴에 대한 포인터를 검색합니다. 반환된 주소는 프로세스의 가상 주소 공간에 있습니다.

GetApplicationRestartSettings 지정된 프로세스에 대해 등록된 다시 시작 정보를 검색합니다.
GetAtomNameA 지정된 로컬 원자와 연결된 문자열의 복사본을 검색합니다. (ANSI)
GetAtomNameW 지정된 로컬 원자와 연결된 문자열의 복사본을 검색합니다. (유니코드)
GetBinaryTypeA 파일이 실행 파일(.exe) 파일인지 여부와 해당하는 경우 실행 파일을 실행하는 하위 시스템을 결정합니다. (ANSI)
GetBinaryTypeW 파일이 실행 파일(.exe) 파일인지 여부와 해당하는 경우 실행 파일을 실행하는 하위 시스템을 결정합니다. (유니코드)
GetCommConfig 통신 디바이스의 현재 구성을 검색합니다.
GetCommMask 지정된 통신 디바이스에 대한 이벤트 마스크 값을 검색합니다.
GetCommModemStatus 모뎀 컨트롤 레지스터 값을 검색합니다.
GetCommPorts 올바른 형식의 COM 포트를 포함하는 배열을 가져옵니다.
GetCommProperties 지정된 통신 디바이스의 통신 속성에 대한 정보를 검색합니다.
GetCommState 지정된 통신 디바이스에 대한 현재 컨트롤 설정을 검색합니다.
GetCommTimeouts 지정된 통신 디바이스에서 모든 읽기 및 쓰기 작업에 대한 제한 시간 매개 변수를 검색합니다.

GetCompressedFileSizeTransactedA 지정된 파일을 트랜잭션 작업으로 저장하는 데 사용되는 디스크 스토리지의 실제 바이트 수를 검색합니다. (ANSI)
GetCompressedFileSizeTransactedW 지정된 파일을 트랜잭션 작업으로 저장하는 데 사용되는 디스크 스토리지의 실제 바이트 수를 검색합니다. (유니코드)
GetComputerNameA 로컬 컴퓨터의 NetBIOS 이름을 검색합니다. 이 이름은 시스템이 레지스트리에서 읽을 때 시스템 시작 시 설정됩니다. (ANSI)
GetComputerNameW 로컬 컴퓨터의 NetBIOS 이름을 검색합니다. 이 이름은 시스템이 레지스트리에서 읽을 때 시스템 시작 시 설정됩니다. (유니코드)
GetCurrentActCtx GetCurrentActCtx 함수는 호출 스레드의 활성 활성화 컨텍스트에 핸들을 반환합니다.
GetCurrentDirectory 현재 프로세스의 현재 디렉터리를 검색합니다.
GetCurrentHwProfileA 로컬 컴퓨터의 현재 하드웨어 프로필에 대한 정보를 검색합니다. (ANSI)
GetCurrentHwProfileW 로컬 컴퓨터의 현재 하드웨어 프로필에 대한 정보를 검색합니다. (유니코드)
GetCurrentUmsThread 호출하는 UMS 스레드의 UMS(사용자 모드 예약) 스레드 컨텍스트를 반환합니다.
GetDefaultCommConfigA 지정된 통신 디바이스에 대한 기본 구성을 검색합니다. (ANSI)
GetDefaultCommConfigW 지정된 통신 디바이스에 대한 기본 구성을 검색합니다. (유니코드)

GetDevicePowerState 지정된 디바이스의 현재 전원 상태를 검색합니다.
GetDllDirectoryA 애플리케이션에 대한 DLL을 찾는 데 사용되는 검색 경로의 애플리케이션별 부분을 검색합니다. (ANSI)
GetDllDirectoryW 애플리케이션에 대한 DLL을 찾는 데 사용되는 검색 경로의 애플리케이션별 부분을 검색합니다. (유니코드)
GetEnabledXStateFeatures x86 또는 x64 프로세서에서 활성화된 XState 기능의 마스크를 가져옵니다.
GetEnvironmentVariable GetEnvironmentVariable 함수(winbase.h)는 호출 프로세스의 환경 블록에서 지정된 변수의 내용을 검색합니다.
GetEventLogInformation 지정된 이벤트 로그에 대한 정보를 검색합니다.
GetFileAttributesTransactedA 지정된 파일 또는 디렉터리에 대한 파일 시스템 특성을 트랜잭션 작업으로 검색합니다. (ANSI)
GetFileAttributesTransactedW 지정된 파일 또는 디렉터리에 대한 파일 시스템 특성을 트랜잭션 작업으로 검색합니다. (유니코드)
GetFileBandwidthReservation 지정된 파일이 있는 볼륨의 대역폭 예약 속성을 검색합니다.
GetFileInformationByHandleEx 지정된 파일의 파일 정보를 검색합니다. (GetFileInformationByHandleEx)
GetFileSecurityA 파일 또는 디렉터리의 보안에 대한 지정된 정보를 가져옵니다. 가져온 정보는 호출자의 액세스 권한 및 권한에 의해 제한됩니다. (GetFileSecurityA)

GetFirmwareEnvironmentVariableA 지정된 펌웨어 환경 변수의 값을 검색합니다. (ANSI)
GetFirmwareEnvironmentVariableExA 지정된 펌웨어 환경 변수 및 해당 특성의 값을 검색합니다. (ANSI)
GetFirmwareEnvironmentVariableExW 지정된 펌웨어 환경 변수 및 해당 특성의 값을 검색합니다. (유니코드)
GetFirmwareEnvironmentVariableW 지정된 펌웨어 환경 변수의 값을 검색합니다. (유니코드)
GetFirmwareType 로컬 컴퓨터의 펌웨어 유형을 검색합니다.
GetFullPathNameTransactedA 지정된 파일의 전체 경로 및 파일 이름을 트랜잭션 작업으로 검색합니다. (ANSI)
GetFullPathNameTransactedW 지정된 파일의 전체 경로 및 파일 이름을 트랜잭션 작업으로 검색합니다. (유니코드)
GetLogicalDriveStringsA 버퍼를 시스템에서 유효한 드라이브를 지정하는 문자열로 채웁니다. (GetLogicalDriveStringsA)
GetLongPathNameTransactedA 지정된 경로를 트랜잭션 작업으로 긴 형식으로 변환합니다. (ANSI)
GetLongPathNameTransactedW 지정된 경로를 트랜잭션 작업으로 긴 형식으로 변환합니다. (유니코드)
GetMailslotInfo 지정된 mailslot에 대한 정보를 검색합니다.
GetMaximumProcessorCount 프로세서 그룹 또는 시스템에 있을 수 있는 논리 프로세서의 최대 수를 반환합니다.

GetMaximumProcessorGroupCount 시스템에 있을 수 있는 프로세서 그룹의 최대 수를 반환합니다.
GetNamedPipeClientComputerNameA GetNamedPipeClientComputerNameA(ANSI) 함수(winbase.h)는 지정된 명명된 파이프에 대한 클라이언트 컴퓨터 이름을 검색합니다.
GetNamedPipeClientProcessId 지정된 명명된 파이프에 대한 클라이언트 프로세스 식별자를 검색합니다.
GetNamedPipeClientSessionId 지정된 명명된 파이프에 대한 클라이언트 세션 식별자를 검색합니다.
GetNamedPipeHandleStateA GETNamedPipeHandleStateA(ANSI) 함수(winbase.h)는 지정된 명명된 파이프에 대한 정보를 검색합니다.
GetNamedPipeServerProcessId 지정된 명명된 파이프에 대한 서버 프로세스 식별자를 검색합니다.
GetNamedPipeServerSessionId 지정된 명명된 파이프에 대한 서버 세션 식별자를 검색합니다.
GetNextUmsListItem 스레드 컨텍스트 목록에서 다음 UMS(사용자 모드 예약) 스레드 컨텍스트를 반환합니다.
GetNumaAvailableMemoryNode 지정된 노드에서 사용할 수 있는 메모리 양을 검색합니다.
GetNumaAvailableMemoryNodeEx USHORT 값으로 지정된 노드에서 사용할 수 있는 메모리 양을 검색합니다.
GetNumaNodeNumberFromHandle 지정된 파일 핸들로 표시되는 파일 또는 I/O 디바이스와 연결된 NUMA 노드를 검색합니다.
GetNumaNodeProcessorMask 지정된 노드의 프로세서 마스크를 검색합니다.

GetNumaProcessorNode 지정된 프로세서의 노드 번호를 검색합니다.
GetNumaProcessorNodeEx 노드 번호를 지정된 논리 프로세서에 대한 USHORT 값으로 검색합니다.
GetNumaProximityNode 지정된 근접 도메인 식별자에 해당하는 NUMA 노드 번호를 검색합니다.
GetNumberOfEventLogRecords 지정된 이벤트 로그의 레코드 수를 검색합니다.
GetOldestEventLogRecord 지정된 이벤트 로그에서 가장 오래된 레코드의 절대 레코드 번호를 검색합니다.
GetPrivateProfileInt GetPrivateProfileInt 함수(winbase.h)는 초기화 파일의 지정된 섹션에서 키와 연결된 정수 를 검색합니다.
GetPrivateProfileIntA 초기화 파일의 지정된 섹션에서 키와 연결된 정수 를 검색합니다. (GetPrivateProfileIntA)
GetPrivateProfileIntW GetPrivateProfileIntW(유니코드) 함수(winbase.h)는 초기화 파일의 지정된 섹션에서 키와 연결된 정수 를 검색합니다.
GetPrivateProfileSection GetPrivateProfileSection 함수(winbase.h)는 초기화 파일의 지정된 섹션에 대한 모든 키와 값을 검색합니다.
GetPrivateProfileSectionA 초기화 파일의 지정된 섹션에 대한 모든 키와 값을 검색합니다. (GetPrivateProfileSectionA)
GetPrivateProfileSectionNames GetPrivateProfileSectionNames 함수(winbase.h)는 초기화 파일에서 모든 섹션의 이름을 검색합니다.

GetPrivateProfileSectionNamesA 초기화 파일에서 모든 섹션의 이름을 검색합니다. (GetPrivateProfileSectionNamesA)
GetPrivateProfileSectionNamesW GetPrivateProfileSectionNamesW(유니코드) 함수(winbase.h)는 초기화 파일에서 모든 섹션의 이름을 검색합니다.
GetPrivateProfileSectionW GetPrivateProfileSectionW(유니코드) 함수(winbase.h)는 초기화 파일의 지정된 섹션에 대한 모든 키와 값을 검색합니다.
GetPrivateProfileString GetPrivateProfileString 함수(winbase.h)는 초기화 파일의 지정된 섹션에서 문자열을 검색합니다.
GetPrivateProfileStringA 초기화 파일의 지정된 섹션에서 문자열을 검색합니다. (GetPrivateProfileStringA)
GetPrivateProfileStringW GetPrivateProfileStringW(유니코드) 함수(winbase.h)는 초기화 파일의 지정된 섹션에서 문자열을 검색합니다.
GetPrivateProfileStruct GetPrivateProfileStruct 함수(winbase.h)는 초기화 파일의 지정된 섹션에서 키와 연결된 데이터를 검색합니다.
GetPrivateProfileStructA 초기화 파일의 지정된 섹션에서 키와 연결된 데이터를 검색합니다. (GetPrivateProfileStructA)
GetPrivateProfileStructW GetPrivateProfileStructW(유니코드) 함수(winbase.h)는 초기화 파일의 지정된 섹션에서 키와 연결된 데이터를 검색합니다.
GetProcessAffinityMask 지정된 프로세스의 프로세스 선호도 마스크와 시스템의 시스템 선호도 마스크를 검색합니다.

GetProcessDEPPolicy 지정된 32비트 프로세스에 대한 DEP(데이터 실행 방지) 및 DEP-ATL thunk 에뮬레이션 설정을 가져옵니다. WINDOWS XP SP3: 현재 프로세스에 대한 DEP 및 DEP-ATL 썽크 에뮬레이션 설정을 가져옵니다.
GetProcessIoCounters 지정된 프로세스에서 수행하는 모든 I/O 작업에 대한 회계 정보를 검색합니다.
GetProfileIntA Win.ini 파일의 지정된 섹션에 있는 키에서 정수 를 검색합니다. (ANSI)
GetProfileIntW Win.ini 파일의 지정된 섹션에 있는 키에서 정수 를 검색합니다. (유니코드)
GetProfileSectionA Win.ini 파일의 지정된 섹션에 대한 모든 키와 값을 검색합니다. (ANSI)
GetProfileSectionW Win.ini 파일의 지정된 섹션에 대한 모든 키와 값을 검색합니다. (유니코드)
GetProfileStringA Win.ini 파일의 지정된 섹션에서 키와 연결된 문자열을 검색합니다. (ANSI)
GetProfileStringW Win.ini 파일의 지정된 섹션에서 키와 연결된 문자열을 검색합니다. (유니코드)
GetShortPathNameA 지정된 경로의 짧은 경로 형식을 검색합니다. (GetShortPathNameA)
GetSystemDEPPolicy 시스템에 대한 DEP(데이터 실행 방지) 정책 설정을 가져옵니다.
GetSystemPowerStatus 시스템의 전원 상태를 검색합니다. 상태는 시스템이 AC 또는 DC 전원에서 실행 중인지 여부, 배터리가 현재 충전 중인지 여부, 남아 있는 배터리 수명, 배터리 절약 모드가 켜지거나 꺼져 있는지 여부를 나타냅니다.

GetSystemRegistryQuota 레지스트리의 현재 크기와 레지스트리가 시스템에서 얻을 수 있는 최대 크기를 검색합니다.
GetTapeParameters 테이프 또는 테이프 드라이브를 설명하는 정보를 검색합니다.
GetTapePosition 테이프의 현재 주소를 논리적 또는 절대 블록으로 검색합니다.
GetTapeStatus 테이프 디바이스가 테이프 명령을 처리할 준비가 되었는지 여부를 확인합니다.
GetTempFileName GetTempFileName 함수(winbase.h)는 임시 파일의 이름을 만듭니다. 고유한 파일 이름이 생성되면 빈 파일이 만들어지고 해당 파일에 대한 핸들이 해제됩니다. 그렇지 않으면 파일 이름만 생성됩니다.
GetThreadEnabledXStateFeatures GetThreadEnabledXStateFeatures 함수는 현재 스레드에 대해 현재 사용하도록 설정된 XState 기능 집합을 반환합니다.
GetThreadSelectorEntry 지정된 선택기 및 스레드에 대한 설명자 테이블 항목을 검색합니다.
GetUmsCompletionListEvent 지정된 UMS(사용자 모드 예약) 완료 목록과 연결된 이벤트에 대한 핸들을 검색합니다.
GetUmsSystemThreadInformation 지정된 스레드가 UMS 스케줄러 스레드, UMS 작업자 스레드 또는 비 UMS 스레드인지 여부를 쿼리합니다.
GetUserNameA 현재 스레드와 연결된 사용자의 이름을 검색합니다. (ANSI)
GetUserNameW 현재 스레드와 연결된 사용자의 이름을 검색합니다. (유니코드)

GetVolumeNameForVolumeMountPointA 지정된 볼륨 탑재 지점(드라이브 문자, 볼륨 GUID 경로 또는 탑재된 폴더)과 연결된 볼륨의 볼륨 GUID 경로를 검색합니다. (GetVolumeNameForVolumeMountPointA)
GetVolumePathNameA 지정된 경로가 탑재되는 볼륨 탑재 지점을 검색합니다. (GetVolumePathNameA)
GetVolumePathNamesForVolumeNameA 지정된 볼륨에 대한 드라이브 문자 및 탑재된 폴더 경로 목록을 검색합니다. (GetVolumePathNamesForVolumeNameA)
GetXStateFeaturesMask CONTEXT 구조 내에 설정된 XState 기능의 마스크를 반환합니다.
GlobalAddAtomA 전역 원자 테이블에 문자열을 추가하고 문자열을 식별하는 고유 값(원자)을 반환합니다. (GlobalAddAtomA)
GlobalAddAtomExA 전역 원자 테이블에 문자열을 추가하고 문자열을 식별하는 고유 값(원자)을 반환합니다. (GlobalAddAtomExA)
GlobalAddAtomExW 전역 원자 테이블에 문자열을 추가하고 문자열을 식별하는 고유 값(원자)을 반환합니다. (GlobalAddAtomExW)
GlobalAddAtomW 전역 원자 테이블에 문자열을 추가하고 문자열을 식별하는 고유 값(원자)을 반환합니다. (GlobalAddAtomW)
GlobalAlloc 힙에서 지정된 바이트 수를 할당합니다. (GlobalAlloc)
GlobalDeleteAtom 전역 문자열 원자의 참조 수를 줄입니다. 원자의 참조 수가 0에 도달하면 GlobalDeleteAtom은 글로벌 원자 테이블에서 원자와 연결된 문자열을 제거합니다.

GlobalDiscard 지정된 전역 메모리 블록을 삭제합니다.
GlobalFindAtomA 전역 원자 테이블에서 지정된 문자열을 검색하고 해당 문자열과 연결된 전역 원자를 검색합니다. (ANSI)
GlobalFindAtomW 전역 원자 테이블에서 지정된 문자열을 검색하고 해당 문자열과 연결된 전역 원자를 검색합니다. (유니코드)
GlobalFlags 지정된 전역 메모리 개체에 대한 정보를 검색합니다.
GlobalFree 지정된 전역 메모리 개체를 해제하고 해당 핸들을 무효화합니다.
GlobalGetAtomNameA 지정된 전역 원자와 연결된 문자열의 복사본을 검색합니다. (ANSI)
GlobalGetAtomNameW 지정된 전역 원자와 연결된 문자열의 복사본을 검색합니다. (유니코드)
GlobalHandle 전역 메모리 블록에 대해 지정된 포인터와 연결된 핸들을 검색합니다.
GlobalLock 전역 메모리 개체를 잠그고 개체 메모리 블록의 첫 번째 바이트에 대한 포인터를 반환합니다.
GlobalMemoryStatus 물리적 메모리와 가상 메모리 모두에 대한 시스템의 현재 사용량에 대한 정보를 검색합니다. (GlobalMemoryStatus)
GlobalReAlloc 지정된 전역 메모리 개체의 크기 또는 특성을 변경합니다. 크기가 증가하거나 감소할 수 있습니다.

GlobalSize 지정된 전역 메모리 개체의 현재 크기(바이트)를 검색합니다.
GlobalUnlock GMEM_MOVEABLE 할당된 메모리 개체와 연결된 잠금 수를 감소합니다.
HasOverlappedIoCompleted 미해결 I/O 작업의 완료를 폴링하는 데 사용할 수 있는 고성능 테스트 작업을 제공합니다.
InitAtomTable 로컬 원자 테이블을 초기화하고 해시 버킷 수를 지정된 크기로 설정합니다.
InitializeContext 필요한 크기와 맞춤을 사용하여 버퍼 내에서 CONTEXT 구조를 초기화합니다.
InitializeContext2 XSTATE 압축 마스크를 지정하는 옵션을 사용하여 필요한 크기와 맞춤을 사용하여 버퍼 내에서 CONTEXT 구조를 초기화합니다.
InitializeThreadpoolEnvironment 콜백 환경을 초기화합니다.
InterlockedExchangeSubtract 두 값의 원자성 빼기 를 수행합니다.
IsBadCodePtr 호출 프로세스에 지정된 주소의 메모리에 대한 읽기 권한이 있는지 여부를 확인합니다.
IsBadReadPtr 호출 프로세스에서 지정된 메모리 범위에 대한 읽기 권한이 있는지 확인합니다. (IsBadReadPtr)
IsBadStringPtrA 호출 프로세스에서 지정된 메모리 범위에 대한 읽기 권한이 있는지 확인합니다. (IsBadStringPtrA)
IsBadStringPtrW 호출 프로세스에서 지정된 메모리 범위에 대한 읽기 권한이 있는지 확인합니다. (IsBadStringPtrW)

IsBadWritePtr 호출 프로세스에서 지정된 메모리 범위에 대한 쓰기 권한이 있는지 확인합니다.
IsNativeVhdBoot OS가 VHD 컨테이너에서 부팅되었는지를 나타냅니다.
IsSystemResumeAutomatic 컴퓨터의 현재 상태를 확인합니다.
IsTextUnicode 버퍼에 유니코드 텍스트 형식이 포함될 가능성이 있는지 확인합니다.
LoadModule 애플리케이션을 로드 및 실행하거나 기존 애플리케이션의 새 인스턴스를 만듭니다.
LoadPackagedLibrary 지정된 패키지된 모듈 및 해당 종속성을 호출 프로세스의 주소 공간에 로드합니다.
LocalAlloc 힙에서 지정된 바이트 수를 할당합니다. (LocalAlloc)
LocalFlags 지정된 로컬 메모리 개체에 대한 정보를 검색합니다.
LocalFree 지정된 로컬 메모리 개체를 해제하고 해당 핸들을 무효화합니다.
LocalHandle 로컬 메모리 개체에 대해 지정된 포인터와 연결된 핸들을 검색합니다.
LocalLock 로컬 메모리 개체를 잠그고 개체 메모리 블록의 첫 번째 바이트에 대한 포인터를 반환합니다.
LocalReAlloc 지정된 로컬 메모리 개체의 크기 또는 특성을 변경합니다. 크기가 증가하거나 감소할 수 있습니다.

LocalSize 지정된 로컬 메모리 개체의 현재 크기(바이트)를 검색합니다.
LocalUnlock LMEM_MOVEABLE 할당된 메모리 개체와 연결된 잠금 수를 감소합니다.
LocateXStateFeature CONTEXT 구조 내에서 XState 기능의 프로세서 상태에 대한 포인터를 검색합니다.
LogonUserA Win32 LogonUser 함수는 사용자를 로컬 컴퓨터에 로그온하려고 시도합니다. LogonUser는 사용자를 가장하는 데 사용할 수 있는 사용자 토큰에 대한 핸들을 반환합니다. (ANSI)
LogonUserExA LogonUserEx 함수는 사용자를 로컬 컴퓨터에 로그온하려고 시도합니다. (ANSI)
LogonUserExW LogonUserEx 함수는 사용자를 로컬 컴퓨터에 로그온하려고 시도합니다. (유니코드)
LogonUserW Win32 LogonUser 함수는 사용자를 로컬 컴퓨터에 로그온하려고 시도합니다. LogonUser는 사용자를 가장하는 데 사용할 수 있는 사용자 토큰에 대한 핸들을 반환합니다. (유니코드)
LookupAccountNameA 시스템 및 계정의 이름을 입력으로 허용합니다. 계정의 SID(보안 식별자)와 계정이 발견된 도메인의 이름을 검색합니다. (ANSI)
LookupAccountNameW 시스템 및 계정의 이름을 입력으로 허용합니다. 계정의 SID(보안 식별자)와 계정이 발견된 도메인의 이름을 검색합니다. (유니코드)
LookupAccountSidA SID(보안 식별자)를 입력으로 허용합니다. 이 SID의 계정 이름과 이 SID가 발견된 첫 번째 도메인의 이름을 검색합니다. (ANSI)
LookupAccountSidLocalA 로컬 컴퓨터에서 지정된 SID에 대한 계정의 이름을 검색합니다. (ANSI)

LookupAccountSidLocalW 로컬 컴퓨터에서 지정된 SID에 대한 계정의 이름을 검색합니다. (유니코드)
LookupAccountSidW SID(보안 식별자)를 입력으로 허용합니다. 이 SID의 계정 이름과 이 SID가 발견된 첫 번째 도메인의 이름을 검색합니다. (유니코드)
LookupPrivilegeDisplayNameA 지정된 권한을 나타내는 표시 이름을 검색합니다. (ANSI)
LookupPrivilegeDisplayNameW 지정된 권한을 나타내는 표시 이름을 검색합니다. (유니코드)
LookupPrivilegeNameA 지정된 LUID(로컬 고유 식별자)에 의해 특정 시스템에 표시되는 권한에 해당하는 이름을 검색합니다. (ANSI)
LookupPrivilegeNameW 지정된 LUID(로컬 고유 식별자)에 의해 특정 시스템에 표시되는 권한에 해당하는 이름을 검색합니다. (유니코드)
LookupPrivilegeValueA 지정된 시스템에서 로컬로 지정된 권한 이름을 나타내는 데 사용되는 LUID(로컬 고유 식별자)를 검색합니다. (ANSI)
LookupPrivilegeValueW 지정된 시스템에서 로컬로 지정된 권한 이름을 나타내는 데 사용되는 LUID(로컬 고유 식별자)를 검색합니다. (유니코드)
lstrcatA 한 문자열을 다른 문자열에 추가합니다. 경고 사용하지 마세요. (ANSI)
lstrcatW 한 문자열을 다른 문자열에 추가합니다. 경고 사용하지 마세요. (유니코드)
lstrcmpA 두 문자 문자열을 비교합니다. 비교는 대/소문자를 구분합니다. (ANSI)

lstrcmpiA 두 문자 문자열을 비교합니다. 대/소문자를 구분하지 않고 비교합니다. (ANSI)
lstrcmpiW 두 문자 문자열을 비교합니다. 대/소문자를 구분하지 않고 비교합니다. (유니코드)
lstrcmpW 두 문자 문자열을 비교합니다. 비교는 대/소문자를 구분합니다. (유니코드)
lstrcpyA 버퍼에 문자열을 복사합니다. (ANSI)
lstrcpynA 원본 문자열에서 버퍼로 지정된 수의 문자를 복사합니다. 경고 사용하지 마세요. (ANSI)
lstrcpynW 원본 문자열에서 버퍼로 지정된 수의 문자를 복사합니다. 경고 사용하지 마세요. (유니코드)
lstrcpyW 버퍼에 문자열을 복사합니다. (유니코드)
lstrlenA 지정된 문자열의 길이를 결정합니다(종료 null 문자는 포함하지 않음). (ANSI)
lstrlenW 지정된 문자열의 길이를 결정합니다(종료 null 문자는 포함하지 않음). (유니코드)
MAKEINTATOM 지정된 원자를 문자열로 변환하므로 원자 또는 문자열을 허용하는 함수에 전달할 수 있습니다.
MapUserPhysicalPagesScatter 이전에 할당된 실제 메모리 페이지를 AWE(주소 창 확장) 지역의 지정된 주소에 매핑합니다. (MapUserPhysicalPagesScatter)
MapViewOfFileExNuma 호출 프로세스의 주소 공간에 파일 매핑 보기를 매핑하고 실제 메모리에 대한 NUMA 노드를 지정합니다.

MoveFile MoveFile 함수(winbase.h)는 자식을 포함하여 기존 파일 또는 디렉터리를 이동합니다.
MoveFileA 자식을 포함하여 기존 파일 또는 디렉터리를 이동합니다. (MoveFileA)
MoveFileExA 자식을 포함하여 기존 파일 또는 디렉터리를 다양한 이동 옵션으로 이동합니다. (ANSI)
MoveFileExW 자식을 포함하여 기존 파일 또는 디렉터리를 다양한 이동 옵션으로 이동합니다. (유니코드)
MoveFileTransactedA 기존 파일 또는 해당 자식을 포함한 디렉터리를 트랜잭션 작업으로 이동합니다. (ANSI)
MoveFileTransactedW 기존 파일 또는 해당 자식을 포함한 디렉터리를 트랜잭션 작업으로 이동합니다. (유니코드)
MoveFileW MoveFileW(유니코드) 함수(winbase.h)는 자식을 포함하여 기존 파일 또는 디렉터리를 이동합니다.
MoveFileWithProgressA 자식을 포함하여 파일 또는 디렉터리를 이동합니다. 진행률 알림을 받는 콜백 함수를 제공할 수 있습니다. (ANSI)
MoveFileWithProgressW 자식을 포함하여 파일 또는 디렉터리를 이동합니다. 진행률 알림을 받는 콜백 함수를 제공할 수 있습니다. (유니코드)
MulDiv 두 개의 32비트 값을 곱한 다음 64비트 결과를 세 번째 32비트 값으로 나눕니다.
NotifyChangeEventLog 지정된 이벤트 로그에 이벤트가 기록될 때 애플리케이션이 알림을 받을 수 있도록 합니다.

ObjectCloseAuditAlarmA 프라이빗 개체에 대한 핸들이 삭제되면 보안 이벤트 로그에 감사 메시지를 생성합니다. (ObjectCloseAuditAlarmA)
ObjectDeleteAuditAlarmA ObjectDeleteAuditAlarmA(ANSI) 함수(winbase.h)는 개체가 삭제되면 감사 메시지를 생성합니다.
ObjectOpenAuditAlarmA 클라이언트 애플리케이션이 개체에 대한 액세스 권한을 얻거나 새 개체를 만들려고 할 때 감사 메시지를 생성합니다. (ObjectOpenAuditAlarmA)
ObjectPrivilegeAuditAlarmA 보안 이벤트 로그에 감사 메시지를 생성합니다. (ObjectPrivilegeAuditAlarmA)
OpenBackupEventLogA BackupEventLog 함수에서 만든 백업 이벤트 로그에 대한 핸들을 엽니다. (ANSI)
OpenBackupEventLogW BackupEventLog 함수에서 만든 백업 이벤트 로그에 대한 핸들을 엽니다. (유니코드)
OpenCommPort 통신 디바이스를 열려고 시도합니다.
OpenEncryptedFileRawA 파일을 백업(내보내기) 또는 복원(가져오기)하기 위해 암호화된 파일을 엽니다. (ANSI)
OpenEncryptedFileRawW 파일을 백업(내보내기) 또는 복원(가져오기)하기 위해 암호화된 파일을 엽니다. (유니코드)
OpenEventLogA 지정된 이벤트 로그에 대한 핸들을 엽니다. (ANSI)
OpenEventLogW 지정된 이벤트 로그에 대한 핸들을 엽니다. (유니코드)
OpenFile 파일을 만들거나, 열거나, 다시 열거나, 삭제합니다.

OpenFileById 지정된 식별자와 일치하는 파일을 엽니다.
OpenFileMappingA 명명된 파일 매핑 개체를 엽니다. (OpenFileMappingA)
OpenJobObjectA 기존 작업 개체를 엽니다. (OpenJobObjectA)
OpenPrivateNamespaceA ANSI(OpenPrivateNamespaceA) 함수(winbase.h)는 프라이빗 네임스페이스를 엽니다.
OperationEnd 애플리케이션이 작업을 종료하려고 했음을 시스템에 알 수 있습니다.
OperationStart 애플리케이션이 작업을 시작하려고 했음을 시스템에 알 수 있습니다.
PowerClearRequest 전원 요청 개체에 대해 지정된 형식의 전원 요청 수를 감소합니다.
PowerCreateRequest 새 전원 요청 개체를 만듭니다.
PowerSetRequest 전원 요청 개체에 대해 지정된 형식의 전원 요청 수를 증분합니다.
PrepareTape 테이프에 액세스하거나 제거할 준비를 합니다.
PrivilegedServiceAuditAlarmA 보안 이벤트 로그에 감사 메시지를 생성합니다. (PrivilegedServiceAuditAlarmA)
PulseEvent 지정된 이벤트 개체를 신호 상태로 설정한 다음 적절한 대기 스레드 수를 해제한 후 서명되지 않은 상태로 다시 설정합니다.

PurgeComm 지정된 통신 리소스의 출력 또는 입력 버퍼에서 모든 문자를 삭제합니다. 리소스에 대한 보류 중인 읽기 또는 쓰기 작업을 종료할 수도 있습니다.
QueryActCtxSettingsW QueryActCtxSettingsW 함수는 활성화 컨텍스트와 쿼리할 특성의 네임스페이스 및 이름을 지정합니다.
QueryActCtxW QueryActCtxW 함수는 활성화 컨텍스트를 쿼리합니다.
QueryDosDeviceA MS-DOS 디바이스 이름에 대한 정보를 검색합니다. (QueryDosDeviceA)
QueryFullProcessImageNameA 지정된 프로세스에 대한 실행 파일 이미지의 전체 이름을 검색합니다. (ANSI)
QueryFullProcessImageNameW 지정된 프로세스에 대한 실행 파일 이미지의 전체 이름을 검색합니다. (유니코드)
QueryThreadProfiling 지정된 스레드에 대해 스레드 프로파일링을 사용할 수 있는지 여부를 결정합니다.
QueryUmsThreadInformation 지정된 UMS(사용자 모드 예약) 작업자 스레드에 대한 정보를 검색합니다.
ReadDirectoryChangesExW 지정된 디렉터리 내의 변경 내용을 설명하는 정보를 검색하며, 해당 정보 유형이 지정된 경우 확장된 정보를 포함할 수 있습니다.
ReadDirectoryChangesW 지정된 디렉터리 내의 변경 내용을 설명하는 정보를 검색합니다.
ReadEncryptedFileRaw 암호화된 파일을 백업(내보내기)합니다.

ReadEventLogA 지정된 이벤트 로그에서 지정된 수의 항목을 읽습니다. (ANSI)
ReadEventLogW 지정된 이벤트 로그에서 지정된 수의 항목을 읽습니다. (유니코드)
ReadThreadProfilingData 스레드와 연결된 지정된 프로파일링 데이터를 읽습니다.
RegisterApplicationRecoveryCallback 복구를 위해 애플리케이션의 활성 인스턴스를 등록합니다.
RegisterApplicationRestart 다시 시작할 애플리케이션의 활성 인스턴스를 등록합니다.
RegisterEventSourceA 지정된 이벤트 로그에 등록된 핸들을 검색합니다. (ANSI)
RegisterEventSourceW 지정된 이벤트 로그에 등록된 핸들을 검색합니다. (유니코드)
RegisterWaitForSingleObject 스레드 풀의 대기 스레드가 개체를 대기하도록 지시합니다.
ReleaseActCtx ReleaseActCtx 함수는 지정된 활성화 컨텍스트의 참조 수를 감소합니다.
RemoveDirectoryTransactedA 트랜잭션 작업으로 기존 빈 디렉터리를 삭제합니다. (ANSI)
RemoveDirectoryTransactedW 트랜잭션 작업으로 기존 빈 디렉터리를 삭제합니다. (유니코드)
RemoveSecureMemoryCacheCallback 이전에 AddSecureMemoryCacheCallback 함수에 등록된 콜백 함수의 등록을 취소합니다.

ReOpenFile 다른 액세스 권한, 공유 모드 및 플래그를 사용하여 지정된 파일 시스템 개체를 다시 엽니다.
ReplaceFileA 한 파일을 다른 파일로 바꾸고 원래 파일의 백업 복사본을 만드는 옵션으로 바꿉니다. (ANSI)
ReplaceFileW 한 파일을 다른 파일로 바꾸고 원래 파일의 백업 복사본을 만드는 옵션으로 바꿉니다. (유니코드)
ReportEventA 지정된 이벤트 로그의 끝에 항목을 씁니다. (ANSI)
ReportEventW 지정된 이벤트 로그의 끝에 항목을 씁니다. (유니코드)
RequestWakeupLatency 효과가 없고 STATUS_NOT_SUPPORTED 반환합니다. 이 함수는 이전 버전의 Windows.Windows Server 2008 및 Windows Vista와의 호환성을 위해서만 제공됩니다. 효과가 없으며 항상 성공을 반환합니다.
SetCommBreak 지정한 통신 디바이스에 대한 문자 전송을 일시 중단하고 ClearCommBreak 함수가 호출될 때까지 전송 라인을 중단 상태로 설정합니다.
SetCommConfig 통신 디바이스의 현재 구성을 설정합니다.
SetCommMask 통신 디바이스에 대해 모니터링할 이벤트 집합을 지정합니다.
SetCommState 디바이스 제어 블록(DCB 구조)의 사양에 따라 통신 디바이스를 구성합니다. 함수는 모든 하드웨어 및 컨트롤 설정을 다시 초기화하지만 출력 또는 입력 큐는 비어 있지 않습니다.
SetCommTimeouts 지정된 통신 디바이스의 모든 읽기 및 쓰기 작업에 대한 제한 시간 매개 변수를 설정합니다.

SetCurrentDirectory 현재 프로세스의 현재 디렉터리를 변경합니다.
SetDefaultCommConfigA 통신 디바이스의 기본 구성을 설정합니다. (ANSI)
SetDefaultCommConfigW 통신 디바이스의 기본 구성을 설정합니다. (유니코드)
SetDllDirectoryA 애플리케이션의 DLL을 찾는 데 사용되는 검색 경로에 디렉터리를 추가합니다. (ANSI)
SetDllDirectoryW 애플리케이션의 DLL을 찾는 데 사용되는 검색 경로에 디렉터리를 추가합니다. (유니코드)
SetEnvironmentVariable SetEnvironmentVariable 함수(winbase.h)는 현재 프로세스에 대해 지정된 환경 변수의 내용을 설정합니다.
SetFileAttributesTransactedA 파일 또는 디렉터리의 특성을 트랜잭션 작업으로 설정합니다. (ANSI)
SetFileAttributesTransactedW 파일 또는 디렉터리의 특성을 트랜잭션 작업으로 설정합니다. (유니코드)
SetFileBandwidthReservation 지정된 파일 스트림의 대역폭을 예약할 것을 요청합니다. 예약은 지정된 파일 핸들의 I/O 요청에 대해 밀리초 단위로 바이트 수로 지정됩니다.
SetFileCompletionNotificationModes 파일 핸들에 대한 알림 모드를 설정하여 지정된 파일에 대해 완료 알림이 작동하는 방식을 지정할 수 있습니다.
SetFileSecurityA SETFileSecurityA(ANSI) 함수(winbase.h)는 파일 또는 디렉터리 개체의 보안을 설정합니다.

SetFileShortNameA 지정된 파일의 짧은 이름을 설정합니다. (ANSI)
SetFileShortNameW 지정된 파일의 짧은 이름을 설정합니다. (유니코드)
SetFirmwareEnvironmentVariableA 지정된 펌웨어 환경 변수의 값을 설정합니다. (ANSI)
SetFirmwareEnvironmentVariableExA 지정된 펌웨어 환경 변수의 값을 이 변수가 저장 및 유지 관리되는 방식을 나타내는 특성으로 설정합니다.
SetFirmwareEnvironmentVariableExW 지정된 펌웨어 환경 변수의 값과 이 변수가 저장 및 유지 관리되는 방식을 나타내는 특성을 설정합니다.
SetFirmwareEnvironmentVariableW 지정된 펌웨어 환경 변수의 값을 설정합니다. (유니코드)
SetHandleCount SetHandleCount 함수는 프로세스에서 사용할 수 있는 파일 핸들 수를 변경합니다.
SetMailslotInfo 지정된 mailslot에서 읽기 작업에 사용하는 제한 시간 값을 설정합니다.
SetProcessAffinityMask 지정된 프로세스의 스레드에 대한 프로세서 선호도 마스크를 설정합니다.
SetProcessDEPPolicy 32비트 프로세스에 대한 DEP(데이터 실행 방지) 및 DEP-ATL unk 에뮬레이션 설정을 변경합니다.
SetSearchPathMode 파일을 찾을 때 SearchPath 함수가 사용하는 프로세스별 모드를 설정합니다.

SetSystemPowerState 전원을 종료하여 시스템을 일시 중단합니다. ForceFlag 매개 변수에 따라 함수는 작업을 즉시 일시 중단하거나 작업을 수행하기 전에 모든 애플리케이션 및 디바이스 드라이버의 권한을 요청합니다.
SetTapeParameters 테이프의 블록 크기를 지정하거나 테이프 디바이스를 구성합니다.
SetTapePosition 지정된 디바이스의 테이프 위치를 설정합니다.
SetThreadAffinityMask 지정된 스레드에 대한 프로세서 선호도 마스크를 설정합니다.
SetThreadExecutionState 애플리케이션이 사용 중임을 시스템에 알릴 수 있으므로 애플리케이션이 실행되는 동안 시스템이 절전 모드로 전환되거나 디스플레이가 꺼지는 것을 방지할 수 있습니다.
SetThreadpoolCallbackCleanupGroup 지정된 정리 그룹을 지정된 콜백 환경에 연결합니다. (SetThreadpoolCallbackCleanupGroup)
SetThreadpoolCallbackLibrary 미해결 콜백이 있는 한 지정된 DLL이 로드된 상태로 유지되도록 합니다. (SetThreadpoolCallbackLibrary)
SetThreadpoolCallbackPersistent 콜백이 영구 스레드에서 실행되도록 지정합니다. (SetThreadpoolCallbackPersistent)
SetThreadpoolCallbackPool 콜백을 생성할 때 사용할 스레드 풀을 설정합니다.
SetThreadpoolCallbackPriority 동일한 스레드 풀의 다른 작업 항목을 기준으로 콜백 함수의 우선 순위를 지정합니다. (SetThreadpoolCallbackPriority)
SetThreadpoolCallbackRunsLong 이 콜백 환경과 연결된 콜백이 빠르게 반환되지 않을 수 있음을 나타냅니다. (SetThreadpoolCallbackRunsLong)

SetUmsThreadInformation 지정된 UMS(사용자 모드 예약) 작업자 스레드에 대한 애플리케이션별 컨텍스트 정보를 설정합니다.
SetupComm 지정된 통신 디바이스에 대한 통신 매개 변수를 초기화합니다.
SetVolumeLabelA 파일 시스템 볼륨의 레이블을 설정합니다. (ANSI)
SetVolumeLabelW 파일 시스템 볼륨의 레이블을 설정합니다. (유니코드)
SetVolumeMountPointA 볼륨을 드라이브 문자 또는 다른 볼륨의 디렉터리에 연결합니다. (ANSI)
SetVolumeMountPointW 볼륨을 드라이브 문자 또는 다른 볼륨의 디렉터리에 연결합니다. (유니코드)
SetXStateFeaturesMask CONTEXT 구조 내에서 설정된 XState 기능의 마스크를 설정합니다.
SwitchToFiber 섬유를 예약합니다. 함수는 파이버에서 호출되어야 합니다.
TransmitCommChar 지정된 통신 디바이스의 출력 버퍼에서 보류 중인 데이터보다 먼저 지정된 문자를 전송합니다.
UmsThreadYield 호출하는 UMS 작업자 스레드가 실행 중인 UMS(사용자 모드 예약) 스케줄러 스레드에 대한 제어를 생성합니다.
UnregisterApplicationRecoveryCallback 복구 목록에서 애플리케이션의 활성 인스턴스를 제거합니다.
UnregisterApplicationRestart 다시 시작 목록에서 애플리케이션의 활성 인스턴스를 제거합니다.

UnregisterWait RegisterWaitForSingleObject 함수에서 발급한 등록된 대기 작업을 취소합니다. (UnregisterWait)
UpdateResourceA PE(이식 가능한 실행 파일) 파일의 리소스를 추가, 삭제 또는 대체합니다. (ANSI)
UpdateResourceW PE(이식 가능한 실행 파일) 파일의 리소스를 추가, 삭제 또는 대체합니다. (유니코드)
VerifyVersionInfoA 운영 체제 버전 요구 사항 집합을 현재 실행 중인 시스템 버전의 해당 값과 비교합니다. (ANSI)
VerifyVersionInfoW 운영 체제 버전 요구 사항 집합을 현재 실행 중인 시스템 버전의 해당 값과 비교합니다. (유니코드)
WaitCommEvent 지정된 통신 디바이스에 대한 이벤트가 발생할 때까지 기다립니다. 이 함수에서 모니터링하는 이벤트 집합은 디바이스 핸들과 연결된 이벤트 마스크에 포함됩니다.
WaitNamedPipeA WAITNamedPipeA(ANSI) 함수(winbase.h)는 시간 제한 간격이 경과하거나 지정된 명명된 파이프의 인스턴스를 연결에 사용할 수 있을 때까지 기다립니다(즉, 파이프의 서버 프로세스에 파이프에 보류 중인 ConnectNamedPipe 작업이 있음).
WinExec 지정된 애플리케이션을 실행합니다.
WinMain 그래픽 Windows 기반 애플리케이션에 대한 사용자가 제공한 진입점입니다.
Wow64GetThreadSelectorEntry 지정된 선택기 및 WOW64 스레드에 대한 설명자 테이블 항목을 검색합니다.
WriteEncryptedFileRaw 암호화된 파일을 복원(가져오기)합니다.

WritePrivateProfileSectionA 초기화 파일에서 지정된 섹션의 키와 값을 바꿉니다. (ANSI)
WritePrivateProfileSectionW 초기화 파일에서 지정된 섹션의 키와 값을 바꿉니다. (유니코드)
WritePrivateProfileStringA 문자열을 초기화 파일의 지정된 섹션에 복사합니다. (ANSI)
WritePrivateProfileStringW 문자열을 초기화 파일의 지정된 섹션에 복사합니다. (유니코드)
WritePrivateProfileStructA 초기화 파일의 지정된 섹션에 있는 키에 데이터를 복사합니다. 데이터를 복사할 때 함수는 체크섬을 계산하고 데이터 끝에 추가합니다. (ANSI)
WritePrivateProfileStructW 초기화 파일의 지정된 섹션에 있는 키에 데이터를 복사합니다. 데이터를 복사할 때 함수는 체크섬을 계산하고 데이터 끝에 추가합니다. (유니코드)
WriteProfileSectionA Win.ini 파일에서 지정된 섹션의 내용을 지정된 키와 값으로 바꿉니다. (ANSI)
WriteProfileSectionW Win.ini 파일에서 지정된 섹션의 내용을 지정된 키와 값으로 바꿉니다. (유니코드)
WriteProfileStringA 문자열을 Win.ini 파일의 지정된 섹션에 복사합니다. (ANSI)
WriteProfileStringW 문자열을 Win.ini 파일의 지정된 섹션에 복사합니다. (유니코드)
WriteTapemark 지정된 수의 파일 표시, 집합 표시, 짧은 파일 표시 또는 긴 파일 표시를 테이프 디바이스에 씁니다.
WTSGetActiveConsoleSessionId 콘솔 세션의 세션 식별자를 검색합니다.

ZombifyActCtx ZombifyActCtx 함수는 지정된 활성화 컨텍스트를 비활성화하지만 할당을 취소하지는 않습니다.

콜백 함수

LPPROGRESS_ROUTINE CopyFileEx, MoveFileTransacted 및 MoveFileWithProgress 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PCOPYFILE2_PROGRESS_ROUTINE CopyFile2 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PFE_EXPORT_FUNC ReadEncryptedFileRaw와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
PFE_IMPORT_FUNC WriteEncryptedFileRaw와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. 시스템은 백업 파일 데이터의 일부를 검색하기 위해 매번 ImportCallback을 하나 이상 호출합니다.
PFIBER_START_ROUTINE CreateFiber 함수와 함께 사용되는 애플리케이션 정의 함수입니다. 섬유의 시작 주소 역할을 합니다.

구조체

ACTCTX_SECTION_KEYED_DATA ACTCTX_SECTION_KEYED_DATA 구조체는 FindActCtxSectionString 및 FindActCtxSectionGuid 함수에서 GUID 또는 32비트 정수 태그가 지정된 활성화 컨텍스트 섹션과 함께 활성화 컨텍스트 정보를 반환하는 데 사용됩니다.
ACTCTXA ACTCTX 구조체는 CreateActCtx 함수에서 활성화 컨텍스트를 만드는 데 사용됩니다. (ANSI)

ACTCTXW ACTCTX 구조체는 CreateActCtx 함수에서 활성화 컨텍스트를 만드는 데 사용됩니다. (유니코드)
COMMCONFIG 통신 디바이스의 구성 상태에 대한 정보를 포함합니다.
COMMPROP 통신 드라이버에 대한 정보를 포함합니다.
COMMTIMEOUTS 통신 디바이스에 대한 제한 시간 매개 변수를 포함합니다.
COMSTAT 통신 디바이스에 대한 정보를 포함합니다.
COPYFILE2_EXTENDED_PARAMETERS CopyFile2 함수에 대한 확장 매개 변수를 포함합니다.
COPYFILE2_MESSAGE 보류 중인 복사 작업에 대한 정보와 함께 CopyFile2ProgressRoutine 콜백 함수에 전달됩니다.
DCB 직렬 통신 디바이스에 대한 제어 설정을 정의합니다.
EVENTLOG_FULL_INFORMATION 이벤트 로그가 가득 찼는지 여부를 나타냅니다.
FILE_ALIGNMENT_INFO 파일에 대한 맞춤 정보를 포함합니다.
FILE_ALLOCATION_INFO 파일에 할당해야 하는 총 바이트 수를 포함합니다.
FILE_ATTRIBUTE_TAG_INFO 요청된 파일 특성 정보를 받습니다. 모든 핸들에 사용됩니다.

FILE_BASIC_INFO 파일에 대한 기본 정보를 포함합니다. 파일 핸들에 사용됩니다.
FILE_COMPRESSION_INFO 파일 압축 정보를 받습니다.
FILE_DISPOSITION_INFO 파일을 삭제해야 하는지 여부를 나타냅니다. 모든 핸들에 사용됩니다.
FILE_END_OF_FILE_INFO 파일의 끝을 설정해야 하는 지정된 값을 포함합니다.
FILE_FULL_DIR_INFO 파일에 대한 디렉터리 정보를 포함합니다. (FILE_FULL_DIR_INFO)
FILE_ID_BOTH_DIR_INFO 지정된 디렉터리의 파일에 대한 정보를 포함합니다.
FILE_ID_DESCRIPTOR 사용 중인 ID 유형을 지정합니다.
FILE_ID_EXTD_DIR_INFO 파일에 대한 식별 정보를 포함합니다. (FILE_ID_EXTD_DIR_INFO)
FILE_ID_INFO 파일에 대한 식별 정보를 포함합니다. (FILE_ID_INFO)
FILE_IO_PRIORITY_HINT_INFO 파일 I/O 작업의 우선 순위 힌트를 지정합니다.
FILE_NAME_INFO 파일 이름을 받습니다.
FILE_REMOTE_PROTOCOL_INFO 파일 원격 프로토콜 정보를 포함합니다.

FILE_RENAME_INFO 파일 이름을 바꿀 이름을 포함합니다.
FILE_STANDARD_INFO 파일에 대한 확장 정보를 받습니다.
FILE_STORAGE_INFO 파일에 대한 디렉터리 정보를 포함합니다. (FILE_STORAGE_INFO)
FILE_STREAM_INFO 지정된 파일에 대한 파일 스트림 정보를 받습니다.
HW_PROFILE_INFOA 하드웨어 프로필에 대한 정보를 포함합니다. (ANSI)
HW_PROFILE_INFOW 하드웨어 프로필에 대한 정보를 포함합니다. (유니코드)
MEMORYSTATUS 실제 메모리와 가상 메모리의 현재 상태에 대한 정보를 포함합니다.
OFSTRUCT OpenFile 함수가 열거나 열려고 시도한 파일에 대한 정보를 포함합니다.
OPERATION_END_PARAMETERS 이 구조체는 OperationEnd 함수에서 사용됩니다.
OPERATION_START_PARAMETERS 이 구조체는 OperationStart 함수에서 사용됩니다.
STARTUPINFOEXA 새 프로세스의 창 스테이션, 데스크톱, 표준 핸들 및 특성을 지정합니다. CreateProcess 및 CreateProcessAsUser 함수와 함께 사용됩니다. (ANSI)
STARTUPINFOEXW 새 프로세스의 창 스테이션, 데스크톱, 표준 핸들 및 특성을 지정합니다. CreateProcess 및 CreateProcessAsUser 함수와 함께 사용됩니다. (유니코드)

SYSTEM_POWER_STATUS 시스템의 전원 상태에 대한 정보를 포함합니다.
UMS_SCHEDULER_STARTUP_INFO UMS(사용자 모드 예약) 스케줄러 스레드에 대한 특성을 지정합니다.
UMS_SYSTEM_THREAD_INFORMATION UMS 스케줄러 스레드, UMS 작업자 스레드 또는 비 UMS 스레드를 지정합니다. GetUmsSystemThreadInformation 함수는 이 구조를 사용합니다.
WIN32_STREAM_ID 스트림 데이터를 포함합니다.

열거형

COPYFILE2_COPY_PHASE 오류 발생 시 복사본의 단계를 나타냅니다.
COPYFILE2_MESSAGE_ACTION CopyFile2ProgressRoutine 콜백 함수에서 반환되어 보류 중인 복사 작업에 대해 수행해야 하는 작업을 나타냅니다.
COPYFILE2_MESSAGE_TYPE COPYFILE2_MESSAGE 구조에서 CopyFile2ProgressRoutine 콜백 함수에 전달된 메시지 유형을 나타냅니다.
FILE_ID_TYPE FILE_ID_DESCRIPTOR 구조체의 공용 구조체에 대한 판별자입니다.
PRIORITY_HINT 파일 I/O 작업에 대한 우선 순위 힌트를 지정하기 위해 FILE_IO_PRIORITY_HINT_INFO 구조에 사용되는 값을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winber.h 헤더

아티클2023. 08. 27.

이 헤더는 Lightweight Directory Access Protocol에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Lightweight Directory Access Protocol

winber.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

ber_alloc_t 새 BerElement 구조를 할당하고 생성합니다.
ber_bvdup ber_bvdup 함수는 제공된 berval 구조체의 복사본을 만듭니다.
ber_bvecfree ber_bvecfree 함수는 버벌 구조의 배열을 해제합니다.
ber_bvfree ber_bvfree 함수(winber.h)는 버벌 구조를 해제합니다.
ber_first_element ber_first_element 함수는 제공된 BerElement 구조에 저장된 SET, SET OF, SEQUENCE 또는 SEQUENCE OF 데이터 값의 순회를 시작하는 데 사용됩니다. 첫 번째 요소의 태그와 길이를 반환합니다.
ber_flatten ber_flatten 함수는 제공된 BerElement 구조체에서 가져온 데이터를 포함하는 새 berval 구조를 할당합니다.
ber_free ber_free 함수는 이전에 ber_alloc_t, ber_init 또는 ldap_first_attribute/ldap_next_attribute 검색 함수로 할당된 BerElement 구조를 해제합니다.

ber_init ber_init 함수는 제공된 berval 구조에서 가져온 데이터를 포함하는 새 BerElement 구조를 할당합니다.
ber_next_element ber_next_element 함수는 제공된 BerElement 구조에 저장된 SET, SET OF, SEQUENCE 또는 SEQUENCE OF 데이터 값을 트래버스하는 데 ber_first_element 함께 사용됩니다. 생성된 형식에서 다음 요소의 태그와 길이를 반환합니다.
ber_peek_tag 제공된 BerElement 구조체에서 구문 분석할 다음 요소의 태그를 반환합니다.
ber_printf BER 요소를 인코딩하는 데 사용되며 sprintf_s 비슷합니다.
ber_scanf ber_scanf 함수는 SSCANF_S 유사한 방식으로 BER 요소를 디코딩합니다.
ber_skip_tag ber_skip_tag 함수는 현재 태그를 건너뛰고 제공된 BerElement 구조체에서 다음 요소의 태그를 반환합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winbio.h 헤더

아티클2023. 03. 08.

이 헤더는 Windows 생체 인식 프레임워크에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 생체 인식 프레임워크

winbio.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WinBioAcquireFocus 창 포커스를 가져옵니다.
WinBioAsyncEnumBiometricUnits 입력 인자 유형과 일치하는 연결된 모든 생체 인식 단위를 비동기적으로 열거합니다.
WinBioAsyncEnumDatabases 지정된 형식과 일치하는 등록된 모든 데이터베이스를 비동기적으로 열거합니다.
WinBioAsyncEnumServiceProviders 설치된 생체 인식 서비스 공급자에 대한 정보를 비동기적으로 반환합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioAsyncMonitorFrameworkChanges 생체 인식 프레임워크의 변경 내용에 대한 비동기 모니터를 시작합니다.
WinBioAsyncOpenFramework 생체 인식 프레임워크에 대한 핸들을 엽니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioAsyncOpenSession 생체 인식 서비스 공급자와 하나 이상의 생체 인식 단위에 비동기적으로 연결합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.

WinBioCancel 지정된 세션에 대해 보류 중인 모든 생체 인식 작업을 취소합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioCaptureSample 생체 인식 샘플을 캡처하고 원시 또는 처리된 데이터로 BIR(생체 인식 정보 레코드)을 채웁니다.
WinBioCaptureSampleWithCallback 생체 인식 샘플을 비동기적으로 캡처하고 원시 또는 처리된 데이터를 BIR(생체 인식 정보 레코드)에 반환합니다.
WinBioCloseFramework WinBioAsyncOpenFramework를 사용하여 이전에 연 프레임워크 핸들을 닫습니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioCloseSession 생체 인식 세션을 닫고 관련 리소스를 해제합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioControlUnit 호출자가 생체 인식 단위에서 공급업체 정의 제어 작업을 수행할 수 있도록 허용합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioControlUnitPrivileged 호출자가 생체 인식 단위에서 권한 있는 공급업체 정의 제어 작업을 수행할 수 있도록 허용합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioDeleteTemplate 템플릿 저장소에서 생체 인식 템플릿을 삭제합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioEnrollBegin 생체 인식 등록 순서를 시작하고 빈 생체 인식 템플릿을 만듭니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioEnrollCapture 생체 인식 샘플을 캡처하고 템플릿에 추가합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.

WinBioEnrollCaptureWithCallback 생체 인식 샘플을 비동기적으로 캡처하여 템플릿에 추가합니다.
WinBioEnrollCommit 보류 중인 생체 인식 템플릿을 마무리하고 등록에 사용되는 생체 인식 단위와 연결된 데이터베이스에 저장합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioEnrollDiscard 등록 순서를 종료하고 보류 중인 생체 인식 템플릿을 삭제합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioEnrollSelect 여러 개인을 나타내는 데이터가 샘플 버퍼에 있을 때 등록하려는 개인을 지정합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioEnumBiometricUnits 입력 형식과 일치하는 연결된 모든 생체 인식 단위를 열거합니다.
WinBioEnumDatabases 지정된 형식과 일치하는 등록된 모든 데이터베이스를 열거합니다.
WinBioEnumEnrollments 지정된 ID 및 생체 인식 단위에 등록된 생체 인식 하위 요소를 검색합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioEnumServiceProviders 설치된 생체 인식 서비스 공급자에 대한 정보를 검색합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioFree Windows 생체 인식 프레임워크 API 함수에 대한 이전 호출을 통해 클라이언트 애플리케이션에 할당된 메모리를 해제합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioGetCredentialState 지정된 사용자에 대해 자격 증명이 설정되었는지 여부를 지정하는 값을 검색합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.

WinBioGetDomainLogonSetting 사용자가 생체 인식 정보를 사용하여 도메인에 로그온할 수 있는지 여부를 지정하는 값을 검색합니다.
WinBioGetEnabledSetting Windows 생체 인식 프레임워크를 현재 사용할 수 있는지 여부를 지정하는 값을 검색합니다.
WinBioGetEnrolledFactors 지정된 사용자가 컴퓨터에 있는 생체 인식 등록에 대한 정보를 가져옵니다.
WinBioGetLogonSetting 사용자가 생체 인식 정보를 사용하여 로그온할 수 있는지 여부를 나타내는 값을 검색합니다.
WinBioGetProperty 세션, 단위 또는 템플릿 속성을 검색합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioIdentify 생체 인식 샘플을 캡처하고 기존 생체 인식 템플릿과 일치하는지 여부를 결정합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioIdentifyWithCallback 생체 인식 샘플을 비동기적으로 캡처하고 기존 생체 인식 템플릿과 일치하는지 여부를 결정합니다.
WinBioLocateSensor 사용자가 대화형으로 선택한 생체 인식 단위의 ID 번호를 검색합니다.
WinBioLocateSensorWithCallback 사용자가 대화형으로 선택한 생체 인식 단위의 ID 번호를 비동기적으로 검색합니다.
WinBioLockUnit 단일 세션에서만 사용할 수 있는 생체 인식 단위를 잠급 수 있습니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioLogonIdentifiedUser 빠른 사용자가 생체 인식 세션에서 수행된 마지막 성공적인 식별 작업과 연결된 계정으로 전환합니다.

WinBioMonitorPresence 지정된 생체 인식 단위에 대한 얼굴 인식 또는 홍채 모니터링 메커니즘을 켭니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioOpenSession 생체 인식 서비스 공급자 및 하나 이상의 생체 인식 단위에 연결합니다.
WinBioRegisterEventMonitor 열린 세션과 연결된 서비스 공급자로부터 이벤트 알림을 수신하도록 콜백 함수를 등록합니다.
WinBioReleaseFocus 창 포커스를 해제합니다.
WinBioRemoveAllCredentials 저장소에서 모든 자격 증명을 제거합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioRemoveAllDomainCredentials 저장소에서 현재 도메인에 대한 모든 사용자 자격 증명을 제거합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioRemoveCredential 지정된 사용자의 생체 인식 로그온 자격 증명을 삭제합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioSetCredential 현재 사용자의 생체 인식 로그온 자격 증명을 저장합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioSetProperty 생체 인식 세션, 단위, 템플릿 또는 계정과 연결된 표준 속성의 값을 설정합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioUnlockUnit 지정된 생체 인식 단위에서 세션 잠금을 해제합니다.
WinBioUnregisterEventMonitor 열린 생체 인식 세션과 연결된 서비스 공급자의 이벤트 알림을 취소합니다.

WinBioVerify 생체 인식 샘플을 캡처하고 샘플이 지정된 사용자 ID에 해당하는지 여부를 확인합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.
WinBioVerifyWithCallback 생체 인식 샘플을 비동기적으로 캡처하고 샘플이 지정된 사용자 ID에 해당하는지 여부를 확인합니다.
WinBioWait 세션에 대해 보류 중인 모든 생체 인식 작업이 완료되거나 취소될 때까지 호출자 실행을 차단합니다. Windows 10 빌드 1607부터 이 함수를 모바일 이미지와 함께 사용할 수 있습니다.

콜백 함수

PWINBIO_ASYNC_COMPLETION_CALLBACK WinBioAsyncOpenSession 또는 WinBioAsyncOpenFramework를 사용하여 시작된 비동기 작업이 완료되었음을 클라이언트 애플리케이션에 알립니다.
PWINBIO_CAPTURE_CALLBACK 비동기 WinBioCaptureSampleWithCallback 함수의 결과를 반환합니다.
PWINBIO_ENROLL_CAPTURE_CALLBACK 비동기 WinBioEnrollCaptureWithCallback 함수의 결과를 반환하기 위해 Windows 생체 인식 프레임워크에서 호출됩니다.
PWINBIO_EVENT_CALLBACK 비동기 WinBioRegisterEventMonitor 함수의 결과를 반환합니다.
PWINBIO_IDENTIFY_CALLBACK 비동기 WinBioIdentifyWithCallback 함수의 결과를 반환합니다.
PWINBIO_LOCATE_SENSOR_CALLBACK 비동기 WinBioLocateSensorWithCallback 함수의 결과를 반환합니다.

PWINBIO_VERIFY_CALLBACK 비동기 WinBioVerifyWithCallback 함수의 결과를 반환합니다.

구조체

WINBIO_ASYNC_RESULT 비동기 작업의 결과를 포함합니다.

열거형

WINBIO_ASYNC_NOTIFICATION_METHOD 비동기 작업에 대한 완료 알림을 클라이언트 애플리케이션에 전달하는 방법을 지정하는 상수를 정의합니다.

winbio_adapter.h 헤더

아티클2023. 03. 08.

이 헤더는 Windows 생체 인식 프레임워크에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 생체 인식 프레임워크

winbio_adapter.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WbioQueryEngineInterface 엔진 어댑터의 WINBIO_ENGINE_INTERFACE 구조체에 대한 포인터를 검색합니다.
WbioQuerySensorInterface 센서 어댑터의 WINBIO_SENSOR_INTERFACE 구조체에 대한 포인터를 검색합니다.
WbioQueryStorageInterface 스토리지 어댑터의 WINBIO_STORAGE_INTERFACE 구조체에 대한 포인터를 검색합니다.

콜백 함수

PIBIO_ENGINE_ACCEPT_SAMPLE_DATA_FN 원시 생체 인식 샘플을 수락하고 기능 집합을 추출합니다.
PIBIO_ENGINE_ACTIVATE_FN 엔진 어댑터에 센서 구성 요소를 유휴 상태로 가져오는 데 필요한 모든 작업을 수행할 수 있는 기회를 제공합니다.
PIBIO_ENGINE_ATTACH_FN 생체 인식 단위의 처리 파이프라인에 엔진 어댑터를 추가합니다.

PIBIO_ENGINE_CHECK_FOR_DUPLICATE_FN 파이프라인의 새 템플릿이 템플릿과 연결된 ID에 관계없이 데이터베이스에 이미 저장된 템플릿을 복제하는지 여부를 결정합니다.
PIBIO_ENGINE_CLEAR_CONTEXT_FN 새 작업을 위해 생체 인식 단위의 처리 파이프라인을 준비합니다. 전자
PIBIO_ENGINE_COMMIT_ENROLLMENT_FN 등록 개체를 마무리하고 템플릿으로 변환한 다음 데이터베이스에 템플릿을 저장합니다.
PIBIO_ENGINE_CONTROL_UNIT_FN 상승된 권한이 필요하지 않은 공급업체 정의 제어 작업을 수행합니다. 전자
PIBIO_ENGINE_CONTROL_UNIT_PRIVILEGED_FN 상승된 권한이 필요한 공급업체 정의 제어 작업을 수행합니다. 전자
PIBIO_ENGINE_CREATE_ENROLLMENT_FN 생체 인식 단위 파이프라인에서 등록 개체를 초기화합니다.
PIBIO_ENGINE_CREATE_KEY_FN HMAC 키를 센서에 푸시하기 위해 Windows 생체 인식 프레임워크에서 호출됩니다. 프레임워크가 EngineAdapterIdentifyFeatureSetSecure를 호출하면 반환된 키 식별자가 생체 인식 단위로 다시 전달됩니다.
PIBIO_ENGINE_DEACTIVATE_FN 엔진 어댑터에 센서 구성 요소를 유휴 상태로 두는 데 필요한 모든 작업을 수행할 수 있는 기회를 제공합니다.
PIBIO_ENGINE_DETACH_FN 파이프라인에 연결된 어댑터 관련 리소스를 해제합니다. 전자
PIBIO_ENGINE_DISCARD_ENROLLMENT_FN 파이프라인에서 중간 등록 상태 정보를 삭제합니다.
PIBIO_ENGINE_EXPORT_ENGINE_DATA_FN 표준 생체 인식 정보 레코드의 엔진에서 가장 최근에 처리된 기능 집합 또는 템플릿의 복사본을 검색합니다.

PIBIO_ENGINE_GET_ENROLLMENT_HASH_FN 파이프라인에서 완료된 등록 템플릿의 해시를 검색합니다.
PIBIO_ENGINE_GET_ENROLLMENT_STATUS_FN 등록 개체를 파이프라인에 커밋할 준비가 되었는지 여부를 결정합니다.
PIBIO_ENGINE_IDENTIFY_ALL_FN 현재 카메라 프레임에 있는 모든 사람의 ID를 결정합니다.
PIBIO_ENGINE_IDENTIFY_FEATURE_SET_FN 현재 기능 집합에서 템플릿을 빌드하고 데이터베이스에서 일치하는 템플릿을 찾습니다.
PIBIO_ENGINE_IDENTIFY_FEATURE_SET_SECURE_FN Windows 생체 인식 프레임워크에서 호출하여 현재 기능 집합에서 템플릿을 빌드하고 데이터베이스에서 일치하는 템플릿을 찾습니다.
PIBIO_ENGINE_NOTIFY_POWER_CHANGE_FN 컴퓨터 전원 상태의 변경에 대한 알림을 받고 그에 따라 엔진 어댑터를 준비합니다.
PIBIO_ENGINE_PIPELINE_CLEANUP_FN 엔진 어댑터에 스토리지 어댑터의 도움이 필요한 정리를 수행할 수 있는 기회를 제공합니다.
PIBIO_ENGINE_PIPELINE_INIT_FN 엔진 어댑터에 불완전하게 남아 있는 초기화를 수행할 수 있는 기회를 제공합니다.
PIBIO_ENGINE_QUERY_CALIBRATION_DATA_FN 엔진 어댑터에서 캡처 후 보정 데이터 집합을 가져옵니다.
PIBIO_ENGINE_QUERY_EXTENDED_ENROLLMENT_STATUS_FN WINBIO_PROPERTY_EXTENDED_ENROLLMENT_STATUS 속성을 쿼리합니다.
PIBIO_ENGINE_QUERY_EXTENDED_INFO_FN 생체 인식 엔진 구성 요소의 기능 및 제한 사항을 결정합니다.
PIBIO_ENGINE_QUERY_HASH_ALGORITHMS_FN 엔진 어댑터에서 지원하는 해시 알고리즘을 나타내는 개체 식별자의 배열을 검색합니다.

PIBIO_ENGINE_QUERY_INDEX_VECTOR_SIZE_FN 엔진 어댑터에서 사용하는 인덱스 벡터의 크기를 검색합니다.
PIBIO_ENGINE_QUERY_PREFERRED_FORMAT_FN 엔진 어댑터에서 선호하는 입력 데이터 형식을 결정합니다.
PIBIO_ENGINE_QUERY_SAMPLE_HINT_FN 등록 템플릿을 생성하는 데 엔진 어댑터에 필요한 올바른 샘플 수를 검색합니다.
PIBIO_ENGINE_REFRESH_CACHE_FN 엔진 어댑터에 메모리에 보관할 수 있는 캐시된 템플릿을 삭제해야 한다고 알립니다.
PIBIO_ENGINE_SELECT_CALIBRATION_FORMAT_FN Windows 생체 인식 프레임워크에서 호출하여 엔진 어댑터에서 사용하려는 센서 어댑터의 보정 형식을 결정합니다.
PIBIO_ENGINE_SET_ACCOUNT_POLICY_FN 엔진 어댑터에서 사용하는 확장 기본 및 사용자별 스푸핑 방지 정책을 설정합니다.
PIBIO_ENGINE_SET_ENROLLMENT_PARAMETERS_FN 엔진 어댑터에 등록 작업에 대한 추가 정보를 제공합니다.
PIBIO_ENGINE_SET_ENROLLMENT_SELECTOR_FN 엔진 어댑터에 현재 등록 작업을 추적할 사람을 알려줍니다.
PIBIO_ENGINE_SET_HASH_ALGORITHM_FN 후속 작업에서 사용할 해시 알고리즘을 선택합니다.
PIBIO_ENGINE_UPDATE_ENROLLMENT_FN 등록 개체에 현재 기능 집합을 추가합니다.
PIBIO_ENGINE_VERIFY_FEATURE_SET_FN 현재 기능 집합의 템플릿을 데이터베이스의 특정 템플릿과 비교합니다.
PIBIO_SENSOR_ACCEPT_CALIBRATION_DATA_FN 엔진 어댑터에서 센서 어댑터로 보정 데이터를 전달합니다.

PIBIO_SENSOR_ACTIVATE_FN 센서 어댑터에게 센서 구성 요소를 유휴 상태에서 해제하는 데 필요한 모든 작업을 수행할 수 있는 기회를 제공합니다.
PIBIO_SENSOR_ATTACH_FN 생체 인식 단위의 처리 파이프라인에 센서 어댑터를 추가합니다.
PIBIO_SENSOR_CANCEL_FN 보류 중인 모든 센서 작업을 취소합니다.
PIBIO_SENSOR_CLEAR_CONTEXT_FN PIBIO_SENSOR_CLEAR_CONTEXT_FN(winbio_adapter.h)는 새 작업을 위해 생체 인식 단위의 처리 파이프라인을 준비합니다.
PIBIO_SENSOR_CONTROL_UNIT_FN PIBIO_SENSOR_CONTROL_UNIT_FN(winbio_adapter.h)는 상승된 권한이 필요하지 않은 공급업체 정의 제어 작업을 수행합니다.
PIBIO_SENSOR_CONTROL_UNIT_PRIVILEGED_FN PIBIO_SENSOR_CONTROL_UNIT_PRIVILEGED_FN 콜백 함수(winbio_adapter.h)는 상승된 권한이 필요한 공급업체 정의 제어 작업을 수행합니다.
PIBIO_SENSOR_DEACTIVATE_FN 센서 어댑터에 센서 구성 요소를 유휴 상태로 두는 데 필요한 모든 작업을 수행할 수 있는 기회를 제공합니다.
PIBIO_SENSOR_DETACH_FN 파이프라인에 연결된 어댑터 특정 리소스를 해제합니다.
PIBIO_SENSOR_EXPORT_SENSOR_DATA_FN 표준 WINBIO_BIR 구조로 포맷된 가장 최근에 캡처된 생체 인식 샘플을 검색합니다.
PIBIO_SENSOR_FINISH_CAPTURE_FN PIBIO_SENSOR_FINISH_CAPTURE_FN 콜백(winbio_adapter.h)은 센서 표시기가 켜지거나 꺼져 있는지 여부를 나타내는 값을 검색합니다.

PIBIO_SENSOR_GET_INDICATOR_STATUS_FN PIBIO_SENSOR_GET_INDICATOR_STATUS_FN 콜백 함수(winbio_adapter.h)는 센서 표시기가 켜지거나 꺼져 있는지 여부를 나타내는 값을 검색합니다.
PIBIO_SENSOR_NOTIFY_POWER_CHANGE_FN 컴퓨터 전원 상태의 변경에 대한 알림을 받고 그에 따라 센서 어댑터를 준비합니다.
PIBIO_SENSOR_PIPELINE_CLEANUP_FN 센서 어댑터는 엔진 또는 스토리지 어댑터 구성 요소의 도움이 필요한 정리를 수행할 수 있는 기회를 제공합니다.
PIBIO_SENSOR_PIPELINE_INIT_FN 센서 어댑터는 불완전하게 유지되고 엔진 또는 스토리지 어댑터 구성 요소의 도움이 필요한 초기화를 수행할 수 있는 기회를 제공합니다.
PIBIO_SENSOR_PUSH_DATA_TO_ENGINE_FN 샘플 버퍼의 현재 내용을 엔진 어댑터에서 사용할 수 있도록 합니다.
PIBIO_SENSOR_QUERY_CALIBRATION_FORMATS_FN 센서 어댑터에서 지원하는 보정 형식 집합을 결정합니다.
PIBIO_SENSOR_QUERY_EXTENDED_INFO_FN 생체 인식 센서 구성 요소의 기능 및 제한 사항을 결정합니다.
PIBIO_SENSOR_QUERY_STATUS_FN 센서 디바이스의 현재 상태에 대한 정보를 검색합니다.
PIBIO_SENSOR_RESET_FN 센서를 다시 초기화합니다.
PIBIO_SENSOR_SET_CALIBRATION_FORMAT_FN 센서 어댑터에 엔진 어댑터에서 특정 보정 데이터 형식을 선택했음을 알 수 있습니다.
PIBIO_SENSOR_SET_INDICATOR_STATUS_FN 센서 표시기를 켜거나 끕니다.

PIBIO_SENSOR_SET_MODE_FN 센서 어댑터 모드를 설정합니다.
PIBIO_SENSOR_START_CAPTURE_FN 비동기 생체 인식 캡처를 시작합니다.
PIBIO_STORAGE_ACTIVATE_FN 스토리지 어댑터에 스토리지 구성 요소를 유휴 상태로 가져오는 데 필요한 모든 작업을 수행할 수 있는 기회를 제공합니다.
PIBIO_STORAGE_ADD_RECORD_FN 데이터베이스에 템플릿을 추가합니다.
PIBIO_STORAGE_ATTACH_FN 생체 인식 단위의 처리 파이프라인에 스토리지 어댑터를 추가합니다.
PIBIO_STORAGE_CLEAR_CONTEXT_FN PIBIO_STORAGE_CLEAR_CONTEXT_FN 콜백 함수(winbio_adapter.h)는 새 작업을 위해 생체 인식 단위의 처리 파이프라인을 준비합니다.
PIBIO_STORAGE_CLOSE_DATABASE_FN 파이프라인과 연결된 데이터베이스를 닫고 관련된 모든 리소스를 해제합니다.
PIBIO_STORAGE_CONTROL_UNIT_FN PIBIO_STORAGE_CONTROL_UNIT_FN 콜백 함수(winbio_adapter.h)는 상승된 권한이 필요하지 않은 공급업체 정의 제어 작업을 수행합니다.
PIBIO_STORAGE_CONTROL_UNIT_PRIVILEGED_FN PIBIO_STORAGE_CONTROL_UNIT_PRIVILEGED_FN 콜백 함수(winbio_adapter.h)는 상승된 권한이 필요한 공급업체 정의 제어 작업을 수행합니다.
PIBIO_STORAGE_CREATE_DATABASE_FN 새 데이터베이스를 만들고 구성합니다.
PIBIO_STORAGE_DEACTIVATE_FN 스토리지 어댑터에 스토리지 구성 요소를 유휴 상태로 두는 데 필요한 모든 작업을 수행할 수 있는 기회를 제공합니다.

PIBIO_STORAGE_DELETE_RECORD_FN 데이터베이스에서 하나 이상의 템플릿을 삭제합니다.
PIBIO_STORAGE_DETACH_FN 파이프라인에 연결된 어댑터 관련 리소스를 해제합니다. S
PIBIO_STORAGE_ERASE_DATABASE_FN 데이터베이스를 지우고 삭제할 수 있는 것으로 표시합니다.
PIBIO_STORAGE_FIRST_RECORD_FN 결과 집합 커서를 집합의 첫 번째 레코드에 배치합니다.
PIBIO_STORAGE_GET_CURRENT_RECORD_FN 파이프라인 결과 집합에서 현재 레코드의 내용을 검색합니다.
PIBIO_STORAGE_GET_DATA_FORMAT_FN 파이프라인과 연결된 현재 데이터베이스에서 사용하는 형식 및 버전 정보를 검색합니다.
PIBIO_STORAGE_GET_DATABASE_SIZE_FN 데이터베이스 크기와 사용 가능한 공간을 검색합니다.
PIBIO_STORAGE_GET_RECORD_COUNT_FN 파이프라인 결과 집합의 템플릿 레코드 수를 검색합니다.
PIBIO_STORAGE_NEXT_RECORD_FN 결과 집합 커서를 하나의 레코드로 진행합니다.
PIBIO_STORAGE_NOTIFY_POWER_CHANGE_FN 컴퓨터 전원 상태의 변경에 대한 알림을 받고 그에 따라 스토리지 어댑터를 준비합니다.
PIBIO_STORAGE_OPEN_DATABASE_FN 스토리지 어댑터에서 사용할 데이터베이스를 엽니다.
PIBIO_STORAGE_PIPELINE_CLEANUP_FN 스토리지 어댑터에 템플릿 데이터베이스를 닫을 준비를 위해 정리를 수행할 수 있는 기회를 제공합니다.

PIBIO_STORAGE_PIPELINE_INIT_FN 스토리지 어댑터에 불완전하게 남아 있는 초기화를 수행할 수 있는 기회를 제공합니다.
PIBIO_STORAGE_QUERY_BY_CONTENT_FN 지정된 인덱스 벡터와 연결된 템플릿에 대해 현재 열려 있는 데이터베이스를 쿼리합니다.
PIBIO_STORAGE_QUERY_BY_SUBJECT_FN 지정된 ID 및 하위 요소와 연결된 템플릿에 대해 현재 열려 있는 데이터베이스를 쿼리합니다.
PIBIO_STORAGE_QUERY_EXTENDED_INFO_FN 생체 인식 스토리지 구성 요소의 기능 및 제한 사항을 결정합니다.

구조체

WINBIO_ENGINE_INTERFACE 사용자 지정 엔진 어댑터 함수에 대한 포인터를 포함합니다.
WINBIO_SENSOR_INTERFACE 사용자 지정 센서 어댑터 함수에 대한 포인터를 포함합니다.
WINBIO_STORAGE_INTERFACE 사용자 지정 스토리지 어댑터 함수에 대한 포인터를 포함합니다.
WINBIO_ADAPTER_INTERFACE_VERSION 엔진, 센서 및 스토리지 어댑터 인터페이스 테이블에 사용되는 주 버전 및 부 버전 번호를 포함합니다.
WINBIO_PIPELINE 단일 생체 인식 단위의 센서, 엔진 및 스토리지 어댑터 구성 요소에서 사용하는 공유 컨텍스트 정보를 포함합니다.
WINBIO_STORAGE_RECORD 표준 형식의 생체 인식 템플릿 및 관련 데이터를 포함합니다.

wincodec.h header

이 헤더는 Windows 이미징 구성 요소에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 이미징 구성 요소

wincodec.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Interfaces

테이블 확장

IWICBitmap 비트맵의 쓰기 가능성 및 정적 메모리 내 표현의 개념을 IWICBitmapSource에 추가하는 메서드를 정의합니다.
IWICBitmapClipper 지정된 사각형 영역에 대한 입력 비트맵의 잘린 버전을 생성하는 메서드를 노출합니다.
IWICBitmapCodecInfo 특정 코덱에 대한 정보를 제공하는 메서드를 노출합니다.
IWICBitmapCodecProgressNotification 인코더 및 디코더에 대한 진행률 알림에 사용되는 메서드를 노출합니다.
IWICBitmapDecoder 디코더를 나타내는 메서드를 노출합니다.
IWICBitmapDecoderInfo 디코더에 대한 정보를 제공하는 메서드를 노출합니다.
IWICBitmapEncoder 썸네일, 프레임 및 색상표와 같은 인코더의 속성을 설정하는 메서드를 정의합니다.
IWICBitmapEncoderInfo 인코더에 대한 정보를 제공하는 메서드를 노출합니다.
IWICBitmapFlipRotator

가로 또는 세로로 대칭 이동하거나 90도 단위로 회전할 수 있는 비트맵 소스를 생성하는 메서드를 제공합니다. 대칭 이동은 회전 전에 수행됩니다.
IWICBitmapFrameChainReader 다양한 형식의 체인을 통해 현재 프레임에 연결된 프레임에 대한 액세스를 제공합니다.
IWICBitmapFrameChainWriter 프레임을 서로 다른 형식의 체인으로 연결합니다.
IWICBitmapFrameDecode 인코딩된 파일의 개별 이미지 프레임을 디코딩하는 메서드를 정의합니다.
IWICBitmapFrameEncode 인코더의 개별 이미지 프레임을 나타냅니다.
IWICBitmapLock Lock 메서드를 지원하는 메서드를 노출합니다.
IWICBitmapScaler 다시 샘플링 또는 필터링 알고리즘을 사용하여 입력 비트맵의 크기가 조정된 버전을 나타냅니다.
IWICBitmapSource 픽셀이 검색되는 원본을 참조하지만 다시 쓸 수 없는 메서드를 노출합니다.
IWICBitmapSourceTransform 기본 IWICBitmapSource 구현에 특정 작업을 오프로드하는 메서드를 노출합니다.
IWICColorContext 색 관리를 위한 메서드를 노출합니다.
IWICColorTransform 한 색 컨텍스트에서 다른 색 컨텍스트로 IWICBitmapSource를 변환하는 메서드를 노출합니다.
IWICComponentInfo 구성 요소 정보를 제공하는 메서드를 노출합니다.
IWICDdsDecoder DDS 이미지 형식과 관련된 정보 및 기능을 제공합니다.

IWICDdsEncoder 인코더에 DDS 형식 특정 정보를 쓸 수 있습니다.
IWICDdsFrameDecode 기본 DXGI_FORMAT 형식의 DDS 이미지 데이터의 단일 프레임과 이미지 데이터에 대한 정보를 제공합니다.
IWICDevelopRaw 원시 코덱 형식의 기능에 대한 액세스를 제공하는 메서드를 노출합니다.
IWICDevelopRawNotificationCallback 원본 이미지 변경 알림에 대한 콜백 메서드를 제공합니다.
IWICDisplayAdaptationControl2
IWICEnumMetadataItem 개별 메타데이터 항목에 대한 열거형 서비스를 제공하는 메서드를 노출합니다.
IWICFastMetadataEncoder 현재 위치 메타데이터 편집에 사용되는 메서드를 노출합니다. 빠른 메타데이터 인코더를 사용하면 이미지를 완전히 다시 인코딩하지 않고도 이미지에 메타데이터를 추가하고 제거할 수 있습니다.
IWICFormatConverter 이미지 데이터를 한 픽셀 형식에서 다른 픽셀 형식으로 변환하여 디더링 및 하프톤을 인덱싱된 형식, 팔레트 변환 및 알파 임계값으로 처리하는 IWICBitmapSource를 나타냅니다.
IWICFormatConverterInfo 픽셀 형식 변환기 정보를 제공하는 메서드를 노출합니다.
IWICImageEncoder ID2D1Image 인터페이스를 IWICBitmapEncoder로 인코딩합니다.
IWICImagingFactory 디코더, 인코더 및 픽셀 형식 변환기와 같은 WIC(Windows 이미징 구성 요소)에 대한 구성 요소를 만드는 데 사용되는 메서드를 노출합니다.
IWICImagingFactory2

IWICImageEncoder를 만드는 기능을 포함하는 WIC 팩터리 인터페이스의 확장입니다.
IWICJpegFrameDecode JPEG 이미지를 디코딩하는 메서드를 노출합니다. SOF(Start Of Frame) 헤더, SOS(Start of Scan) 헤더, 허프만 및 양자화 테이블, 그리고 압축된 JPEG 데이터에 대한 액세스를 제공합니다. 또한 효율적인 임의 액세스를 위해 인덱싱을 사용하도록 설정합니다.
IWICJpegFrameEncode 압축된 JPEG 검사 데이터를 WIC 인코더의 출력 스트림에 직접 쓰기 위한 메서드를 노출합니다. 또한 Huffman 및 양자화 테이블에 대한 액세스를 제공합니다.
IWICMetadataQueryReader 메타데이터 쿼리 식을 사용하여 디코더 또는 해당 이미지 프레임에서 메타데이터 블록 및 항목을 검색하는 메서드를 노출합니다.
IWICMetadataQueryWriter 메타데이터 쿼리 식을 사용하여 메타데이터 블록 및 항목을 인코더 또는 해당 이미지 프레임에 설정하거나 제거하는 메서드를 노출합니다.
IWICPalette 주로 인덱싱된 픽셀 형식에 대한 색 테이블에 액세스하고 빌드하기 위한 메서드를 노출합니다.
IWICPixelFormatInfo 픽셀 형식에 대한 정보를 제공하는 메서드를 노출합니다.
IWICPixelFormatInfo2 픽셀 형식에 대한 추가 정보를 제공하여 IWICPixelFormatInfo를 확장합니다.
IWICPlanarBitmapFrameEncode 평면 구성 요소 이미지 픽셀을 인코더에 쓸 수 있습니다.
IWICPlanarBitmapSourceTransform 픽셀 구성 요소가 별도의 구성 요소 평면에 저장되는 평면 Y'CbCr 픽셀 형식에 대한 액세스를 제공합니다.
IWICPlanarFormatConverter 서식 변환기를 평면 원본으로 초기화할 수 있습니다.
IWICProgressCallback

IWICProgressCallback 인터페이스는 규정 준수에 대해서만 문서화됩니다. 해당 사용은 권장되지 않으며 나중에 변경되거나 사용할 수 없습니다. 대신 RegisterProgressNotification을 사용합니다.
IWICProgressiveLevelControl 점진적 디코딩에 대한 정보를 가져오고 제어하는 메서드를 노출합니다.
IWICStream 이미징 및 메타데이터 콘텐츠를 참조하기 위한 WIC(Windows 이미징 구성 요소) 스트림을 나타냅니다.

Functions

테이블 확장

WICConvertBitmapSource 지정된 IWICBitmapSource에서 원하는 픽셀 형식으로 IWICBitmapSource를 가져옵니다.
WICCreateBitmapFromSection GDI(Windows Graphics Device Interface) 섹션 핸들의 픽셀에 의해 지원되는 IWICBitmapSource를 반환합니다. (WICCreateBitmapFromSection)
WICCreateBitmapFromSectionEx GDI(Windows Graphics Device Interface) 섹션 핸들의 픽셀에 의해 지원되는 IWICBitmapSource를 반환합니다. (WICCreateBitmapFromSectionEx)
WICMapGuidToShortName 지정된 GUID와 연결된 짧은 이름을 가져옵니다.
WICMapSchemaToName 지정된 스키마와 연결된 이름을 가져옵니다.
WICMapShortNameToGuid 지정된 짧은 이름과 연결된 GUID를 가져옵니다.

Callback functions

테이블 확장

PFNProgressNotification 코덱 구성 요소 진행률이 발생할 때 호출되는 애플리케이션 정의 콜백 함수입니다.

Structures

테이블 확장

WICBitmapPattern 특정 형식을 식별하는 데 사용할 수 있는 이미지 파일 내의 패턴을 식별하는 멤버를 포함합니다.
WICBitmapPlane 평면 픽셀 형식에 대한 구성 요소 평면의 픽셀 형식, 버퍼, 스트라이드 및 크기를 지정합니다.
WICBitmapPlaneDescription 구성 요소 평면의 픽셀 형식과 크기를 지정합니다.
WICDdsFormatInfo DDS 형식의 DXGI_FORMAT 및 차단 정보를 지정합니다.
WICDdsParameters 포함된 데이터의 DDS 이미지 차원, DXGI_FORMAT 및 알파 모드를 지정합니다.
WICImageParameters 이미지를 인코딩할 때 일반적으로 사용되는 기본 매개 변수를 재정의하는 데 사용할 수 있는 매개 변수를 정의합니다.
WICJpegFrameHeader JPEG 프레임 헤더를 나타냅니다. (WICJpegFrameHeader)
WICJpegScanHeader JPEG 프레임 헤더를 나타냅니다. (WICJpegScanHeader)
WICRawCapabilitiesInfo

원시 코덱 기능을 정의합니다.
WICRawToneCurve 원시 이미지 톤 곡선을 나타냅니다.
WICRawToneCurvePoint 원시 이미지 톤 곡선 점을 나타냅니다.
WICRect WIC(Windows 이미징 구성 요소) API의 사각형을 나타냅니다.

Enumerations

테이블 확장

WIC8BIMIptcDigestProperties 8BIM IPTC 다이제스트 메타데이터 블록에 있는 메타데이터 항목의 식별자를 지정합니다.
WIC8BIMIptcProperties 8BIM IPTC 블록에 있는 메타데이터 항목의 식별자를 지정합니다.
WIC8BIMResolutionInfoProperties 8BIMResolutionInfo 블록에 있는 메타데이터 항목의 식별자를 지정합니다.
WICBitmapAlphaChannelOption 원하는 알파 채널 사용을 지정합니다.
WICBitmapChainType 체인 유형을 지정하는 상수를 정의합니다.
WICBitmapCreateCacheOption 원하는 캐시 사용을 지정합니다.
WICBitmapDecoderCapabilities 디코더의 기능을 지정합니다.

WICBitmapDitherType 이미지 형식 간에 변환할 때 적용할 디더 알고리즘의 형식을 지정합니다.
WICBitmapEncoderCacheOption 인코더에 사용할 수 있는 캐시 옵션을 지정합니다.
WICBitmapInterpolationMode 이미지 크기를 조정하는 데 사용할 샘플링 또는 필터링 모드를 지정합니다.
WICBitmapLockFlags IWICBitmap에 대한 액세스를 지정합니다.
WICBitmapPaletteType 인덱싱된 이미지 형식에 사용되는 색상표의 유형을 지정합니다.
WICBitmapToneMappingMode
WICBitmapTransformOptions 대칭 이동 및 회전 변환을 지정합니다.
WICColorContextType 색 컨텍스트 형식을 지정합니다.
WICComponentEnumerateOptions 구성 요소 열거 옵션을 지정합니다.
WICComponentSigning 구성 요소 서명 상태를 지정합니다.
WICComponentType WIC(Windows 이미징 구성 요소) 구성 요소의 유형을 지정합니다.
WICDdsAlphaMode DDS 이미지에 포함된 픽셀 색 구성 요소 값의 의미를 지정합니다.
WICDdsDimension

DDS 이미지에 포함된 데이터의 차원 형식을 지정합니다.
WICDecodeOptions 디코딩 옵션을 지정합니다.
WICGainMapProperties
WICGifApplicationExtensionProperties GIF(그래픽 교환 형식) 이미지의 애플리케이션 확장 메타데이터 속성을 지정합니다.
WICGifCommentExtensionProperties GIF(그래픽 교환 형식) 이미지의 주석 확장 메타데이터 속성을 지정합니다.
WICGifGraphicControlExtensionProperties GIF(그래픽 교환 형식) 이미지에 대한 각 프레임 애니메이션 간의 전환을 정의하는 그래픽 컨트롤 확장 메타데이터 속성을 지정합니다.
WICGifImageDescriptorProperties GIF(그래픽 교환 형식) 프레임에 대한 이미지 설명자 메타데이터 속성을 지정합니다.
WICGifLogicalScreenDescriptorProperties GIF(그래픽 교환 형식) 메타데이터의 논리적 화면 설명자 속성을 지정합니다.
WICHeifCompressionOption HEIF(고효율 이미지 형식) 압축 옵션을 지정하는 상수를 정의합니다.
WICHeifHdrProperties HEIF(고효율 이미지 형식) 이미지의 HDR 속성을 지정합니다.
WICHeifProperties HEIF(고효율 이미지 형식) 이미지의 속성을 지정합니다.
WICJpegChrominanceProperties JPEG 크로미낸스 테이블 속성을 지정합니다.
WICJpegCommentProperties JPEG 주석 속성을 지정합니다.

WICJpegIndexingOptions JPEG 이미지를 인덱싱하는 옵션을 지정합니다.
WICJpegLuminanceProperties JPEG 광도 테이블 속성을 지정합니다.
WICJpegScanType JPEG 이미지 검사에서 픽셀 데이터의 메모리 레이아웃을 지정합니다.
WICJpegTransferMatrix Y'Cb'Cr'에서 R'G'B'로의 변환 매트릭스를 지정합니다.
WICJpegYCrCbSubsamplingOption JPEG YCrCB 하위 샘플링 옵션을 지정합니다.
WICNamedWhitePoint 원시 이미지의 명명된 흰색 균형을 지정합니다.
WICPixelFormatNumericRepresentation WIC 픽셀 형식의 숫자 표현에 대한 기본 형식을 지정하는 상수를 정의합니다.
WICPlanarOptions IWICPlanarBitmapSourceTransform 구현에 대한 추가 옵션을 지정합니다.
WICPngBkgdProperties PNG(이식 가능한 네트워크 그래픽) 배경(bKGD) 청크 메타데이터 속성을 지정합니다.
WICPngChrmProperties CIE XYZ 색에 대한 PNG(이식 가능한 네트워크 그래픽) cHRM 청크 메타데이터 속성을 지정합니다.
WICPngFilterOption 압축 최적화에 사용할 수 있는 PNG(이식 가능한 네트워크 그래픽) 필터를 지정합니다.
WICPngGamaProperties PNG(이식 가능한 네트워크 그래픽) gAMA 청크 메타데이터 속성을 지정합니다.
WICPngHistProperties

PNG(이식 가능한 네트워크 그래픽) hIST 청크 메타데이터 속성을 지정합니다.
WICPngIccpProperties PNG(이식 가능한 네트워크 그래픽) iCCP 청크 메타데이터 속성을 지정합니다.
WICPngItxtProperties PNG(이식 가능한 네트워크 그래픽) iTXT 청크 메타데이터 속성을 지정합니다.
WICPngSrgbProperties PNG(이식 가능한 네트워크 그래픽) sRGB 청크 메타데이터 속성을 지정합니다.
WICPngTimeProperties PNG(이식 가능한 네트워크 그래픽) tIME 청크 메타데이터 속성을 지정합니다.
WICProgressNotification 진행률 알림 콜백을 호출해야 하는 시기를 지정합니다.
WICProgressOperation 알림을 받을 진행률 작업을 지정합니다.
WICRawCapabilities 원시 이미지의 기능 지원을 지정합니다.
WICRawParameterSet 원시 코덱에서 사용하는 매개 변수 집합을 지정합니다.
WICRawRenderMode 다음 CopyPixels 호출의 렌더링 의도를 지정합니다.
WICRawRotationCapabilities 코덱의 회전 기능을 지정합니다.
WICSectionAccessLevel Windows GDI(그래픽 디바이스 인터페이스) 섹션의 액세스 수준을 지정합니다.
WICTiffCompressionOption TIFF(태그가 지정된 이미지 파일 형식) 압축 옵션을 지정합니다.

WICWebpAnimProperties WebP 이미지의 애니메이션 속성을 지정합니다.
WICWebpAnmfProperties WebP 이미지의 애니메이션 프레임 속성을 지정합니다.

Last updated on 2025. 07. 16.

wincodecsdk.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows 이미징 구성 요소에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Imaging 구성 요소

wincodecsdk.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWICComponentFactory 구성 요소 개발자가 사용하는 구성 요소를 만드는 메서드를 노출합니다. 여기에는 코덱 및 메타데이터 처리기 개발자가 사용할 메타데이터 판독기, 기록기 및 기타 서비스가 포함됩니다.
IWICMetadataBlockReader 코덱의 모든 최상위 메타데이터 블록에 대한 액세스를 제공하는 메서드를 노출합니다.
IWICMetadataBlockWriter 메타데이터 인코딩을 사용하도록 설정하는 메서드를 노출합니다. 이 인터페이스는 디코더 및 해당 이미지 프레임에 의해 구현됩니다.
IWICMetadataHandlerInfo 등록된 메타데이터 처리기에 대한 기본 정보를 제공하는 메서드를 노출합니다.
IWICMetadataReader 밑줄 메타데이터 콘텐츠에 대한 액세스를 제공하는 메서드를 노출합니다. 이 인터페이스는 ISV(독립 소프트웨어 공급업체)에 의해 구현되어 새 메타데이터 판독기를 만듭니다.
IWICMetadataReaderInfo 등록된 메타데이터 판독기에 대한 기본 정보를 제공하는 메서드를 노출합니다.
IWICMetadataWriter 메타데이터 콘텐츠 작성에 대한 액세스를 제공하는 메서드를 노출합니다. 이는 ISV(독립 소프트웨어 공급업체)에서 새 메타데이터 작성기를 만들기 위해 구현됩니다.

IWICMetadataWriterInfo 등록된 메타데이터 작성기에 대한 기본 정보를 제공하는 메서드를 노출합니다.
IWICPersistStream 추가 로드를 제공하고 WICPersistOptions를 사용하는 메서드를 저장하는 메서드를 노출합니다.
IWICStreamProvider 스트림 공급자에 대한 메서드를 노출합니다.

Functions

WICGetMetadataContentSize 지정된 IWICMetadataWriter에 포함된 메타데이터 콘텐츠의 크기를 반환합니다. 반환된 크기는 헤더 및 메타데이터의 길이를 차지합니다.
WICMatchMetadataContent 지정된 스트림 내의 콘텐츠와 가장 일치하는 지정된 컨테이너 형식 및 공급업체에 대한 메타데이터 형식 GUID를 가져옵니다.
WICSerializeMetadataContent 지정된 스트림에 메타데이터를 씁니다.

구조체

WICMetadataHeader 메타데이터 헤더를 나타냅니다.
WICMetadataPattern 메타데이터 패턴을 나타냅니다.

열거형

WICMetadataCreationOptions 메타데이터 만들기 옵션을 지정합니다.
WICPersistOptions 스트림을 사용하여 구성 요소를 초기화할 때 사용되는 WIC(Windows 이미징 구성 요소) 옵션을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wincred.h 헤더

아티클2023. 08. 27.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

wincred.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CredDeleteA 사용자의 자격 증명 집합에서 자격 증명을 삭제합니다. (ANSI)
CredDeleteW 사용자의 자격 증명 집합에서 자격 증명을 삭제합니다. (유니코드)
CredEnumerateA 사용자의 자격 증명 집합에서 자격 증명을 열거합니다. (ANSI)
CredEnumerateW 사용자의 자격 증명 집합에서 자격 증명을 열거합니다. (유니코드)
CredFindBestCredentialA CredMan(자격 증명 관리) 데이터베이스에서 현재 로그온 세션과 연결되고 지정된 대상 리소스와 가장 일치하는 일반 자격 증명 집합을 검색합니다. (ANSI)
CredFindBestCredentialW CredMan(자격 증명 관리) 데이터베이스에서 현재 로그온 세션과 연결되고 지정된 대상 리소스와 가장 일치하는 일반 자격 증명 집합을 검색합니다. (유니코드)
CredFree CredFree 함수는 자격 증명 관리 함수에서 반환된 버퍼를 해제합니다.
CredGetSessionTypes CredGetSessionTypes 함수는 현재 로그온 세션에서 지원하는 최대 지속성을 반환합니다. 각 자격 증명 유형에 대해 별도의 최대 지속성이 반환됩니다.

CredGetTargetInfoA CredGetTargetInfo 함수는 명명된 대상 컴퓨터에 대해 알려진 모든 대상 이름 정보를 검색합니다. (ANSI)
CredGetTargetInfoW CredGetTargetInfo 함수는 명명된 대상 컴퓨터에 대해 알려진 모든 대상 이름 정보를 검색합니다. (유니코드)
CredIsMarshaledCredentialA 지정된 사용자 이름 문자열이 이전에 CredMarshalCredential에 의해 마샬링된 마샬링된 자격 증명인지 여부를 확인합니다. (ANSI)
CredIsMarshaledCredentialW 지정된 사용자 이름 문자열이 이전에 CredMarshalCredential에 의해 마샬링된 마샬링된 자격 증명인지 여부를 확인합니다. (유니코드)
CredIsProtectedA CredProtect 함수에 대한 이전 호출을 통해 지정된 자격 증명을 암호화할지 여부를 지정합니다. (ANSI)
CredIsProtectedW CredProtect 함수에 대한 이전 호출을 통해 지정된 자격 증명을 암호화할지 여부를 지정합니다. (유니코드)
CredMarshalCredentialA CredMarshalCredential 함수는 자격 증명을 텍스트 문자열로 변환합니다. (ANSI)
CredMarshalCredentialW CredMarshalCredential 함수는 자격 증명을 텍스트 문자열로 변환합니다. (유니코드)
CredPackAuthenticationBufferA 문자열 사용자 이름 및 암호를 인증 버퍼로 변환합니다. (ANSI)
CredPackAuthenticationBufferW 문자열 사용자 이름 및 암호를 인증 버퍼로 변환합니다. (유니코드)
CredProtectA 현재 보안 컨텍스트만 암호를 해독할 수 있도록 지정된 자격 증명을 암호화합니다. (ANSI)

CredProtectW 현재 보안 컨텍스트만 암호를 해독할 수 있도록 지정된 자격 증명을 암호화합니다. (유니코드)
CredReadA 사용자의 자격 증명 집합에서 자격 증명을 읽습니다. (ANSI)
CredReadDomainCredentialsA 사용자의 자격 증명 집합에서 도메인 자격 증명을 읽습니다. (ANSI)
CredReadDomainCredentialsW 사용자의 자격 증명 집합에서 도메인 자격 증명을 읽습니다. (유니코드)
CredReadW 사용자의 자격 증명 집합에서 자격 증명을 읽습니다. (유니코드)
CredRenameA CredRename은 더 이상 지원되지 않습니다. (ANSI)
CredRenameW CredRename은 더 이상 지원되지 않습니다. (유니코드)
CredUICmdLinePromptForCredentialsA 명령줄(콘솔) 애플리케이션에서 작업하는 사용자의 자격 증명 정보를 묻는 메시지를 표시하고 수락합니다. 사용자가 입력한 이름 및 암호는 확인을 위해 호출 애플리케이션에 다시 전달됩니다. (ANSI)
CredUICmdLinePromptForCredentialsW 명령줄(콘솔) 애플리케이션에서 작업하는 사용자의 자격 증명 정보를 묻는 메시지를 표시하고 수락합니다. 사용자가 입력한 이름 및 암호는 확인을 위해 호출 애플리케이션에 다시 전달됩니다. (유니코드)
CredUIConfirmCredentialsA 수집된 자격 증명의 유효성을 확인하기 위해 CredUIPromptForCredentials 또는 CredUICmdLinePromptForCredentials 후에 호출됩니다. (ANSI)
CredUIConfirmCredentialsW 수집된 자격 증명의 유효성을 확인하기 위해 CredUIPromptForCredentials 또는 CredUICmdLinePromptForCredentials 후에 호출됩니다. (유니코드)

CredUIParseUserNameA CredUIParseUserName 함수는 정규화된 사용자 이름에서 도메인 및 사용자 계정 이름을 추출합니다. (ANSI)
CredUIParseUserNameW CredUIParseUserName 함수는 정규화된 사용자 이름에서 도메인 및 사용자 계정 이름을 추출합니다. (유니코드)
CredUIPromptForCredentialsA 사용자의 자격 증명 정보를 허용하는 구성 가능한 대화 상자를 만들고 표시합니다. (ANSI)
CredUIPromptForCredentialsW 사용자의 자격 증명 정보를 허용하는 구성 가능한 대화 상자를 만들고 표시합니다. (유니코드)
CredUIPromptForWindowsCredentialsA 사용자가 로컬 컴퓨터에 설치된 자격 증명 공급자를 사용하여 자격 증명 정보를 제공할 수 있는 구성 가능한 대화 상자를 만들고 표시합니다. (ANSI)
CredUIPromptForWindowsCredentialsW 사용자가 로컬 컴퓨터에 설치된 자격 증명 공급자를 사용하여 자격 증명 정보를 제공할 수 있는 구성 가능한 대화 상자를 만들고 표시합니다. (유니코드)
CredUIReadSSOCredW CredUIReadSSOCredW 함수는 단일 로그온 자격 증명에 대한 사용자 이름을 검색합니다.
CredUIStoreSSOCredW CredUIStoreSSOCredW 함수는 단일 로그온 자격 증명을 저장합니다.
CredUnmarshalCredentialA CredUnmarshalCredential 함수는 마샬링된 자격 증명을 원래 형식으로 다시 변환합니다. (ANSI)
CredUnmarshalCredentialW CredUnmarshalCredential 함수는 마샬링된 자격 증명을 원래 형식으로 다시 변환합니다. (유니코드)
CredUnPackAuthenticationBufferA CredUIPromptForWindowsCredentials 함수를 호출하여 반환된 인증 버퍼를 문자열 사용자 이름 및 암호로 변환합니다. (ANSI)

CredUnPackAuthenticationBufferW CredUIPromptForWindowsCredentials 함수를 호출하여 반환된 인증 버퍼를 문자열 사용자 이름 및 암호로 변환합니다. (유니코드)
CredUnprotectA CredProtect 함수를 사용하여 이전에 암호화된 자격 증명의 암호를 해독합니다. (ANSI)
CredUnprotectW CredProtect 함수를 사용하여 이전에 암호화된 자격 증명의 암호를 해독합니다. (유니코드)
CredWriteA 새 자격 증명을 만들거나 사용자의 자격 증명 집합에서 기존 자격 증명을 수정합니다. (ANSI)
CredWriteDomainCredentialsA 사용자의 자격 증명 집합에 도메인 자격 증명을 씁니다. (ANSI)
CredWriteDomainCredentialsW 사용자의 자격 증명 집합에 도메인 자격 증명을 씁니다. (유니코드)
CredWriteW 새 자격 증명을 만들거나 사용자의 자격 증명 집합에서 기존 자격 증명을 수정합니다. (유니코드)

구조체

CERT_CREDENTIAL_INFO CERT_CREDENTIAL_INFO 구조에는 인증서에 대한 참조가 포함되어 있습니다.
CREDENTIAL_ATTRIBUTEA CREDENTIAL_ATTRIBUTE 구조에는 자격 증명의 애플리케이션 정의 특성이 포함됩니다. 특성은 키워드(keyword) 값 쌍입니다. 특성의 의미를 정의하는 것은 애플리케이션에 달려 있습니다. (ANSI)
CREDENTIAL_ATTRIBUTEW CREDENTIAL_ATTRIBUTE 구조에는 자격 증명의 애플리케이션 정의 특성이 포함됩니다. 특성은 키워드(keyword) 값 쌍입니다. 특성의 의미를 정의하는 것은 애플리케이션에 달려 있습니다. (유니코드)

CREDENTIAL_TARGET_INFORMATIONA CREDENTIAL_TARGET_INFORMATION 구조에는 대상 컴퓨터의 이름, 도메인 및 트리가 포함됩니다. (ANSI)
CREDENTIAL_TARGET_INFORMATIONW CREDENTIAL_TARGET_INFORMATION 구조에는 대상 컴퓨터의 이름, 도메인 및 트리가 포함됩니다. (유니코드)
크레덴티알라 CREDENTIAL 구조에는 개별 자격 증명이 포함됩니다. (ANSI)
CREDENTIALW CREDENTIAL 구조에는 개별 자격 증명이 포함됩니다. (유니코드)
CREDUI_INFOA CREDUI_INFO 구조체는 자격 증명 정보를 가져오는 데 사용되는 대화 상자를 만드는 CredUIPromptForCredentials 함수에 정보를 전달하는 데 사용됩니다. (ANSI)
CREDUI_INFOW CREDUI_INFO 구조체는 자격 증명 정보를 가져오는 데 사용되는 대화 상자를 만드는 CredUIPromptForCredentials 함수에 정보를 전달하는 데 사용됩니다. (유니코드)
USERNAME_TARGET_CREDENTIAL_INFO USERNAME_TARGET_CREDENTIAL_INFO 구조체에는 자격 증명에 대한 참조가 포함되어 있습니다.

열거형

CRED_MARSHAL_TYPE CredMarshalCredential에 의해 마샬링되거나 CredUnmarshalCredential로 마샬링할 자격 증명 유형을 지정합니다.
CRED_PROTECTION_TYPE CredProtect 함수를 사용할 때 자격 증명이 암호화되는 보안 컨텍스트를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wincrypt.h 헤더

아티클2023. 08. 27.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

wincrypt.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CertAddCertificateContextToStore 인증서 저장소에 인증서 컨텍스트를 추가합니다.
CertAddCertificateLinkToStore 인증서 저장소의 링크를 다른 저장소의 인증서 컨텍스트에 추가합니다.
CertAddCRLContextToStore 지정된 인증서 저장소에 CRL(인증서 해지 목록) 컨텍스트를 추가합니다.
CertAddCRLLinkToStore 다른 저장소의 CRL(인증서 해지 목록) 컨텍스트에 저장소의 링크를 추가합니다.
CertAddCTLContextToStore 인증서 저장소에 CTL(인증서 신뢰 목록) 컨텍스트를 추가합니다.
CertAddCTLLinkToStore CertAddCTLLinkToStore 함수는 저장소의 링크를 다른 저장소의 CTL(인증서 신뢰 목록) 컨텍스트에 추가합니다. 이 함수는 CTL 컨텍스트의 중복을 만들고 추가하는 대신 원래 CTL 컨텍스트에 대한 링크를 추가합니다.
CertAddEncodedCertificateToStore 인코딩된 인증서에서 인증서 컨텍스트를 만들고 인증서 저장소에 추가합니다.
CertAddEncodedCertificateToSystemStoreA 지정된 시스템 저장소를 열고 인코딩된 인증서를 추가합니다. (ANSI)

CertAddEncodedCertificateToSystemStoreW 지정된 시스템 저장소를 열고 인코딩된 인증서를 추가합니다. (유니코드)
CertAddEncodedCRLToStore 인코딩된 CRL에서 CRL(인증서 해지 목록) 컨텍스트를 만들고 인증서 저장소에 추가합니다.
CertAddEncodedCTLToStore 인코딩된 CTL에서 CTL(인증서 신뢰 목록) 컨텍스트를 만들고 인증서 저장소에 추가합니다.
CertAddEnhancedKeyUsageIdentifier CertAddEnhancedKeyUsageIdentifier 함수는 인증서의 EKU(향상된 키 사용) 확장 속성에 OID(사용 식별자 개체 식별자)를 추가합니다.
CertAddRefServerOcspResponse HCERT_SERVER_OCSP_RESPONSE 핸들에 대한 참조 수를 증분합니다.
CertAddRefServerOcspResponseContext CERT_SERVER_OCSP_RESPONSE_CONTEXT 구조체에 대한 참조 수를 증분합니다.
CertAddSerializedElementToStore 직렬화된 인증서, CRL(인증서 해지 목록) 또는 CTL(인증서 신뢰 목록) 요소를 저장소에 추가합니다.
CertAddStoreToCollection CertAddStoreToCollection 함수는 형제 인증서 저장소를 컬렉션 인증서 저장소에 추가합니다.
CertAlgIdToOID ALG_ID(CryptoAPI 알고리즘 식별자)를 ASN.1(Abstract Syntax Notation One) OID(개체 식별자) 문자열로 변환합니다.
CertCloseServerOcspResponse OCSP(온라인 인증서 상태 프로토콜) 서버 응답 핸들을 닫습니다.
CertCloseStore 인증서 저장소 핸들을 닫고 저장소의 참조 수를 줄입니다.
CertCompareCertificate 인증서의 발급자 이름과 일련 번호를 비교하여 두 인증서가 동일한지 여부를 확인합니다.

CertCompareCertificateName CertCompareCertificateName 함수는 두 인증서 CERT_NAME_BLOB 구조를 비교하여 동일한지 여부를 확인합니다. CERT_NAME_BLOB 구조는 인증서의 주체 및 발급자용으로 사용됩니다.
CertCompareIntegerBlob CertCompareIntegerBlob 함수는 두 개의 정수 BLOB을 비교하여 같은 숫자 값을 나타내는지 여부를 결정합니다.
CertComparePublicKeyInfo CertComparePublicKeyInfo 함수는 인코딩된 두 공용 키를 비교하여 동일한지 여부를 확인합니다.
CertControlStore 사용 중인 캐시된 저장소의 콘텐츠와 스토리지에 유지되는 해당 저장소의 콘텐츠 간에 차이가 있는 경우 애플리케이션에 알림을 받을 수 있습니다.
CertCreateCertificateChainEngine CertCreateCertificateChainEngine 함수는 애플리케이션에 대한 기본이 아닌 새로운 체인 엔진을 만듭니다.
CertCreateCertificateContext 인코딩된 인증서에서 인증서 컨텍스트를 만듭니다. 만든 컨텍스트는 인증서 저장소에 유지되지 않습니다. 함수는 생성된 컨텍스트 내에서 인코딩된 인증서의 복사본을 만듭니다.
CertCreateContext 인코딩된 바이트에서 지정된 컨텍스트를 만듭니다. 만든 컨텍스트에는 확장 속성이 포함되지 않습니다.
CertCreateCRLContext CertCreateCRLContext 함수는 인코딩된 CRL에서 CRL(인증서 해지 목록) 컨텍스트를 만듭니다. 만든 컨텍스트는 인증서 저장소에 유지되지 않습니다. 생성된 컨텍스트 내에서 인코딩된 CRL의 복사본을 만듭니다.
CertCreateCTLContext CertCreateCTLContext 함수는 인코딩된 CTL에서 CTL(인증서 신뢰 목록) 컨텍스트를 만듭니다. 만든 컨텍스트는 인증서 저장소에 유지되지 않습니다. 함수는 생성된 컨텍스트 내에서 인코딩된 CTL의 복사본을 만듭니다.
CertCreateCTLEntryFromCertificateContextProperties CertCreateCTLEntryFromCertificateContextProperties 함수는 해당 특성이 인증서 컨텍스트의 속성

인 CTL(인증서 신뢰 목록) 항목을 만듭니다. CTL 항목의 SubjectIdentifier는 인증서의 SHA1 해시입니다.
CertCreateSelfSignCertificate 자체 서명된 인증서를 빌드하고 인증서를 나타내는 CERT_CONTEXT 구조체에 대한 포인터를 반환합니다.
CertDeleteCertificateFromStore CertDeleteCertificateFromStore 함수는 인증서 저장소에서 지정된 인증서 컨텍스트를 삭제합니다.
CertDeleteCRLFromStore CertDeleteCRLFromStore 함수는 인증서 저장소에서 지정된 CRL(인증서 해지 목록) 컨텍스트를 삭제합니다.
CertDeleteCTLFromStore CertDeleteCTLFromStore 함수는 인증서 저장소에서 지정된 CTL(인증서 신뢰 목록) 컨텍스트를 삭제합니다.
CertDuplicateCertificateChain CertDuplicateCertificateChain 함수는 체인의 참조 수를 증가시켜 인증서 체인에 대한 포인터를 복제합니다.
CertDuplicateCertificateContext 참조 횟수를 증가시켜 인증서 컨텍스트를 복제합니다.
CertDuplicateCRLContext CertDuplicateCRLContext 함수는 참조 횟수를 증가시켜 CRL(인증서 해지 목록) 컨텍스트를 복제합니다.
CertDuplicateCTLContext CertDuplicateCTLContext 함수는 참조 횟수를 증가시켜 CTL(인증서 신뢰 목록) 컨텍스트를 복제합니다.
CertDuplicateStore 저장소의 참조 수를 증가시켜 저장소 핸들을 복제합니다.
CertEnumCertificateContextProperties CertEnumCertificateContextProperties 함수는 인증서 컨텍스트와 연결된 첫 번째 또는 다음 확장 속성을 검색합니다.

CertEnumCertificatesInStore 인증서 저장소에서 첫 번째 또는 다음 인증서를 검색합니다. 루프에서 사용되는 이 함수는 인증서 저장소의 모든 인증서를 순서대로 검색할 수 있습니다.
CertEnumCRLContextProperties CertEnumCRLContextProperties 함수는 CRL(인증서 해지 목록) 컨텍스트와 연결된 첫 번째 또는 다음 확장 속성을 검색합니다.
CertEnumCRLsInStore CertEnumCRLsInStore 함수는 인증서 저장소에서 첫 번째 또는 다음 CRL(인증서 해지 목록) 컨텍스트를 검색합니다. 루프에서 사용되는 이 함수는 인증서 저장소의 모든 CRL 컨텍스트를 순서대로 검색할 수 있습니다.
CertEnumCTLContextProperties CertEnumCTLContextProperties 함수는 CTL(인증서 신뢰 목록) 컨텍스트와 연결된 첫 번째 또는 다음 확장 속성을 검색합니다. 루프에서 사용되는 이 함수는 CTL 컨텍스트와 연결된 모든 확장 속성을 순서대로 검색할 수 있습니다.
CertEnumCTLsInStore CertEnumCTLsInStore 함수는 인증서 저장소의 첫 번째 또는 다음 CTL(인증서 신뢰 목록) 컨텍스트를 검색합니다. 루프에서 사용되는 이 함수는 인증서 저장소의 모든 CTL 컨텍스트를 순서대로 검색할 수 있습니다.
CertEnumPhysicalStore CertEnumPhysicalStore 함수는 컴퓨터의 실제 저장소를 검색합니다. 함수는 찾은 각 물리적 저장소에 대해 제공된 콜백 함수를 호출합니다.
CertEnumSubjectInSortedCTL 정렬된 CTL(인증서 신뢰 목록)에서 첫 번째 또는 다음 TrustedSubject를 검색합니다.
CertEnumSystemStore CertEnumSystemStore 함수는 사용 가능한 시스템 저장소를 검색합니다. 함수는 찾은 각 시스템 저장소에 대해 제공된 콜백 함수를 호출합니다.
CertEnumSystemStoreLocation CertEnumSystemStoreLocation 함수는 모든 시스템 저장소 위치를 검색합니다. 함수는 찾은 각 시스템 저장소 위치에 대해 제공된 콜백 함수를 호출합니다.
CertFindAttribute

CertFindAttribute 함수는 OID(개체 식별자)로 식별된 대로 CRYPT_ATTRIBUTE 배열에서 첫 번째 특성을 찾습니다.
CertFindCertificateInCRL CertFindCertificateInCRL 함수는 CRL(인증서 해지 목록)에서 지정된 인증서를 검색합니다.
CertFindCertificateInStore dwFindType 및 연결된 pvFindPara에서 설정한 검색 조건과 일치하는 인증서 저장소에서 첫 번째 또는 다음 인증서 컨텍스트를 찾습니다.
CertFindChainInStore 지정된 조건을 충족하는 저장소에서 첫 번째 또는 다음 인증서를 찾습니다.
CertFindCRLInStore dwFindType 매개 변수 및 연결된 pvFindPara 매개 변수에 의해 설정된 검색 조건과 일치하는 인증서 저장소에서 첫 번째 또는 다음 CRL(인증서 해지 목록) 컨텍스트를 찾습니다.
CertFindCTLInStore dwFindType 및 연결된 pvFindPara에서 설정한 검색 조건과 일치하는 첫 번째 또는 다음 CTL(인증서 신뢰 목록) 컨텍스트를 찾습니다.
CertFindExtension CertFindExtension 함수는 OID(개체 식별자)로 식별된 대로 CERT_EXTENSION 배열에서 첫 번째 확장을 찾습니다.
CertFindRDNAttr CertFindRDNAttr 함수는 RDN(상대 고유 이름) 목록에서 OID(개체 식별자)로 식별된 첫 번째 RDN 특성을 찾습니다.
CertFindSubjectInCTL CertFindSubjectInCTL 함수는 CTL(인증서 신뢰 목록)에서 지정된 주체를 찾으려고 시도합니다.
CertFindSubjectInSortedCTL CertFindSubjectInSortedCTL 함수는 CTL(정렬된 인증서 신뢰 목록)에서 지정된 주체를 찾으려고 시도합니다.
CertFreeCertificateChain CertFreeCertificateChain 함수는 참조 횟수를 줄여 인증서 체인을 해제합니다. 참조 수가 0이 되면 체인에 할당된 메모리가 해제됩니다.

CertFreeCertificateChainEngine CertFreeCertificateChainEngine 함수는 인증서 신뢰 엔진을 해제합니다.
CertFreeCertificateChainList 포인터 배열을 체인 컨텍스트로 해제합니다.
CertFreeCertificateContext 참조 횟수를 감소시켜 인증서 컨텍스트를 해제합니다. 참조 수가 0으로 설정되면 CertFreeCertificateContext는 인증서 컨텍스트에서 사용하는 메모리를 해제합니다.
CertFreeCRLContext 참조 횟수를 감소시켜 CRL(인증서 해지 목록) 컨텍스트를 해제합니다.
CertFreeCTLContext 참조 횟수를 감소시켜 CTL(인증서 신뢰 목록) 컨텍스트를 해제합니다.
CertFreeServerOcspResponseContext CERT_SERVER_OCSP_RESPONSE_CONTEXT 구조체에 대한 참조 수를 감소합니다.
CertGetCertificateChain 최종 인증서에서 시작하여 가능한 경우 신뢰할 수 있는 루트 인증서로 돌아가는 인증서 체인 컨텍스트를 빌드합니다.
CertGetCertificateContextProperty 인증서 컨텍스트의 확장 속성에 포함된 정보를 검색합니다.
CertGetCRLContextProperty 지정된 CRL(인증서 해지 목록) 컨텍스트에 대한 확장 속성을 가져옵니다.
CertGetCRLFromStore 지정된 발급자의 인증서 저장소에서 첫 번째 또는 다음 CRL(인증서 해지 목록) 컨텍스트를 가져옵니다.
CertGetCTLContextProperty CTL(인증서 신뢰 목록) 컨텍스트의 확장 속성을 검색합니다.
CertGetEnhancedKeyUsage

EKU(향상된 키 사용) 확장 또는 인증서의 EKU 확장 속성에서 정보를 반환합니다.
CertGetIntendedKeyUsage 인증서에서 의도한 키 사용 바이트를 획득합니다.
CertGetIssuerCertificateFromStore 지정된 주체 인증서의 첫 번째 또는 다음 발급자의 인증서 저장소에서 인증서 컨텍스트를 검색합니다. 이 함수를 사용하는 대신 새 인증서 체인 확인 함수를 사용하는 것이 좋습니다.
CertGetNameStringA 인증서 CERT_CONTEXT 구조체에서 주체 또는 발급자 이름을 가져오고 null로 끝나는 문자열로 변환합니다. (ANSI)
CertGetNameStringW 인증서 CERT_CONTEXT 구조체에서 주체 또는 발급자 이름을 가져오고 null로 끝나는 문자열로 변환합니다. (유니코드)
CertGetPublicKeyLength CertGetPublicKeyLength 함수는 공개 키 BLOB에서 퍼블릭/프라이빗 키의 비트 길이를 획득합니다.
CertGetServerOcspResponseContext 지정된 핸들에 대한 비 차단, 시간 유효한 온라인 인증서 상태 프로토콜(OCSP) 응답 컨텍스트를 검색합니다.
CertGetStoreProperty 저장소 속성을 검색합니다.
CertGetSubjectCertificateFromStore 인증서 저장소에서 발급자와 일련 번호로 고유하게 식별된 주체 인증서 컨텍스트를 반환합니다.
CertGetValidUsages 인증서 배열의 모든 인증서에 대해 유효한 사용법의 교집합으로 구성된 사용량 배열을 반환합니다.
CertIsRDNAttrsInCertificateName CertIsRDNAttrsInCertificateName 함수는 인증서 이름의 특성을 지정된 CERT_RDN 비교하여 모든 특성이 포함되는지 여부를 확인합니다.
CertIsStrongHashToSign

서명 인증서의 지정된 해시 알고리즘과 공개 키를 사용하여 강력한 서명을 수행할 수 있는지 여부를 결정합니다.
CertIsValidCRLForCertificate CertIsValidCRLForCertificate 함수는 CRL을 검사하여 인증서가 해지된 경우 특정 인증서를 포함하는 CRL인지 확인합니다.
CertNameToStrA CERT_NAME_BLOB 구조체의 인코딩된 이름을 null로 끝나는 문자열로 변환합니다. (ANSI)
CertNameToStrW CERT_NAME_BLOB 구조체의 인코딩된 이름을 null로 끝나는 문자열로 변환합니다. (유니코드)
CertOIDToAlgId ALG_ID 식별자가 CNG에서 더 이상 지원되지 않으므로 이 함수 대신 CryptFindOIDInfo 함수를 사용합니다.
CertOpenServerOcspResponse 서버 인증서 체인과 연결된 OCSP(온라인 인증서 상태 프로토콜) 응답에 대한 핸들을 엽니다.
CertOpenStore 지정된 저장소 공급자 유형을 사용하여 인증서 저장소를 엽니다.
CertOpenSystemStoreA 가장 일반적인 시스템 인증서 저장소를 엽니다. 파일 기반 또는 메모리 기반 저장소와 같이 더 복잡한 요구 사항이 있는 인증서 저장소를 열려면 CertOpenStore를 사용합니다. (ANSI)
CertOpenSystemStoreW 가장 일반적인 시스템 인증서 저장소를 엽니다. 파일 기반 또는 메모리 기반 저장소와 같이 더 복잡한 요구 사항이 있는 인증서 저장소를 열려면 CertOpenStore를 사용합니다. (유니코드)
CertRDNValueToStrA CertRDNValueToStr 함수는 CERT_RDN_VALUE_BLOB 이름을 null로 끝나는 문자열로 변환합니다. (ANSI)
CertRDNValueToStrW CertRDNValueToStr 함수는 CERT_RDN_VALUE_BLOB 이름을 null로 끝나는 문자열로 변환합니다. (유니코드)

CertRegisterPhysicalStore 레지스트리 시스템 저장소 컬렉션에 물리적 저장소를 추가합니다.
CertRegisterSystemStore 시스템 저장소를 등록합니다.
CertRemoveEnhancedKeyUsageIdentifier CertRemoveEnhancedKeyUsageIdentifier 함수는 인증서의 EKU(향상된 키 사용) 확장 속성에서 OID(사용 식별자 개체 식별자)를 제거합니다.
CertRemoveStoreFromCollection 컬렉션 저장소에서 형제 인증서 저장소를 제거합니다.
CertResyncCertificateChainEngine 인증서 체인 엔진을 다시 동기화합니다. 이 엔진은 저장소의 엔진을 저장하고 엔진 캐시를 업데이트합니다.
CertRetrieveLogoOrBiometricInfo szOID_LOGOTYPE_EXT 또는 szOID_BIOMETRIC_EXT 인증서 확장에 지정된 로고 또는 생체 인식 정보의 URL 검색을 수행합니다.
CertSaveStore 인증서 저장소를 파일 또는 메모리 BLOB에 저장합니다.
CertSelectCertificateChains 지정된 선택 조건에 따라 인증서 체인을 검색합니다.
CertSerializeCertificateStoreElement CertSerializeCertificateStoreElement 함수는 인증서 컨텍스트의 인코딩된 인증서 및 인코딩된 속성을 직렬화합니다. 나중에 인증서 및 속성을 검색할 수 있도록 결과를 스토리지에 유지할 수 있습니다.
CertSerializeCRLStoreElement CertSerializeCRLStoreElement 함수는 인코딩된 CRL(인증서 해지 목록) 컨텍스트와 해당 속성의 인코딩된 표현을 직렬화합니다.
CertSerializeCTLStoreElement CertSerializeCTLStoreElement 함수는 인코딩된 CTL(인증서 신뢰 목록) 컨텍스트와 해당 속성의 인

코딩된 표현을 직렬화합니다. 나중에 CTL 및 속성을 검색할 수 있도록 결과를 스토리지에 유지할 수 있습니다.
CertSetCertificateContextPropertiesFromCTLEntry 지정된 CTL(인증서 신뢰 목록) 항목의 특성을 사용하여 인증서 컨텍스트의 속성을 설정합니다.
CertSetCertificateContextProperty 지정된 인증서 컨텍스트에 대한 확장 속성을 설정합니다.
CertSetCRLContextProperty 지정된 CRL(인증서 해지 목록) 컨텍스트에 대한 확장 속성을 설정합니다.
CertSetCTLContextProperty 지정된 CTL(인증서 신뢰 목록) 컨텍스트에 대한 확장 속성을 설정합니다.
CertSetEnhancedKeyUsage CertSetEnhancedKeyUsage 함수는 인증서에 대한 향상된 EKU(키 사용량) 속성을 설정합니다.
CertSetStoreProperty CertSetStoreProperty 함수는 저장소 속성을 설정합니다.
CertStrToNameA null로 종료된 X.500 문자열을 인코딩된 인증서 이름으로 변환합니다. (ANSI)
CertStrToNameW null로 종료된 X.500 문자열을 인코딩된 인증서 이름으로 변환합니다. (유니코드)
CertUnregisterPhysicalStore CertUnregisterPhysicalStore 함수는 지정된 시스템 저장소 컬렉션에서 물리적 저장소를 제거합니다. CertUnregisterPhysicalStore를 사용하여 물리적 저장소를 삭제할 수도 있습니다.
CertUnregisterSystemStore CertUnregisterSystemStore 함수는 지정된 시스템 저장소의 등록을 취소합니다.
CertVerifyCertificateChainPolicy 인증서 체인을 검사하여 지정된 유효성 정책 조건 준수를 포함하여 인증서 체인의 유효성을 확인합니다.

CertVerifyCRLRevocation CRL(인증서 해지 목록)을 확인하여 주체의 인증서가 해지되었는지 여부를 확인합니다.
CertVerifyCRLTimeValidity CertVerifyCRLTimeValidity 함수는 CRL의 시간 유효성을 확인합니다.
CertVerifyCTLUsage 주체가 포함된 사용 식별자가 있는 서명되고 시간이 유효한 CTL(인증서 신뢰 목록)을 찾아 주체가 지정된 사용에 대해 신뢰할 수 있는지 확인합니다.
CertVerifyRevocation rgpvContext 배열에 포함된 인증서의 해지 상태 확인합니다. 목록의 인증서가 해지된 것으로 확인되면 추가 검사가 수행되지 않습니다.
CertVerifySubjectCertificateContext CertVerifySubjectCertificateContext 함수는 인증서 발급자의 유효성을 확인하여 인증서에 대해 활성화된 확인 검사를 수행합니다. 이 함수 대신 새 인증서 체인 확인 함수를 사용하는 것이 좋습니다.
CertVerifyTimeValidity CertVerifyTimeValidity 함수는 인증서의 시간 유효성을 확인합니다.
CertVerifyValidityNesting CertVerifyValidityNesting 함수는 주체 인증서의 시간 유효성이 발급자의 시간 유효성 내에 올바르게 중첩되었는지 확인합니다.
CryptAcquireCertificatePrivateKey 인증서에 대한 프라이빗 키를 가져옵니다.
CryptAcquireContextA 특정 CSP(암호화 서비스 공급자) 내의 특정 키 컨테이너에 대한 핸들을 획득하는 데 사용됩니다. 이 반환된 핸들은 선택한 CSP를 사용하는 CryptoAPI 함수 호출에 사용됩니다. (ANSI)
CryptAcquireContextW 특정 CSP(암호화 서비스 공급자) 내의 특정 키 컨테이너에 대한 핸들을 획득하는 데 사용됩니다. 이 반환된 핸들은 선택한 CSP를 사용하는 CryptoAPI 함수 호출에 사용됩니다. (유니코드)
CryptBinaryToStringA 바이트 배열을 형식이 지정된 문자열로 변환합니다. (ANSI)

CryptBinaryToStringW 바이트 배열을 형식이 지정된 문자열로 변환합니다. (유니코드)
CryptCloseAsyncHandle CryptCloseAsyncHandle 함수(wincrypt.h)는 비동기 핸들을 닫습니다.
CryptContextAddRef HCRYPTPROV CSP(암호화 서비스 공급자) 핸들의 참조 수에 하나를 추가합니다.
CryptCreateAsyncHandle CryptCreateAsyncHandle 함수(wincrypt.h)는 비동기 핸들을 만듭니다.
CryptCreateHash 데이터 스트림의 해시를 시작합니다. CSP(암호화 서비스 공급자) 해시 개체에 대한 핸들을 만들어 호출 애플리케이션에 반환합니다.
CryptCreateKeyIdentifierFromCSP 중요 이 API는 더 이상 사용되지 않습니다. (CryptCreateKeyIdentifierFromCSP)
CryptDecodeMessage 암호화 메시지를 디코딩, 암호 해독 및 확인합니다.
CryptDecodeObject CryptDecodeObject 함수는 lpszStructType 매개 변수로 표시된 형식의 구조를 디코딩합니다. CryptDecodeObjectEx는 성능이 크게 향상되어 동일한 기능을 수행하는 API로 사용하는 것이 좋습니다.
CryptDecodeObjectEx lpszStructType 매개 변수로 표시된 형식의 구조를 디코딩합니다.
CryptDecrypt CryptEncrypt 함수를 사용하여 이전에 암호화된 데이터의 암호를 해독합니다.
CryptDecryptAndVerifyMessageSignature CryptDecryptAndVerifyMessageSignature 함수는 메시지의 암호를 해독하고 해당 서명을 확인합니다.

CryptDecryptMessage CryptDecryptMessage 함수는 메시지를 디코딩하고 암호를 해독합니다.
CryptDeriveKey 기본 데이터 값에서 파생된 암호화 세션 키를 생성합니다.
CryptDestroyHash hHash 매개 변수에서 참조하는 해시 개체를 제거합니다.
CryptDestroyKey hKey 매개 변수에서 참조하는 핸들을 해제합니다.
CryptDuplicateHash 중복이 수행되는 시점까지 해시의 정확한 복사본을 만듭니다.
CryptDuplicateKey 키의 정확한 복사본과 키의 상태를 만듭니다.
CryptEncodeObject CryptEncodeObject 함수는 lpszStructType 매개 변수의 값으로 표시된 형식의 구조를 인코딩합니다. CryptEncodeObjectEx는 성능이 크게 향상되어 동일한 기능을 수행하는 API로 사용하는 것이 좋습니다.
CryptEncodeObjectEx lpszStructType 매개 변수의 값으로 표시된 형식의 구조를 인코딩합니다.
CryptEncrypt 데이터를 암호화합니다. 데이터를 암호화하는 데 사용되는 알고리즘은 CSP 모듈에서 보유한 키에 의해 지정되며 hKey 매개 변수에서 참조됩니다.
CryptEncryptMessage CryptEncryptMessage 함수는 메시지를 암호화하고 인코딩합니다.
CryptEnumKeyIdentifierProperties CryptEnumKeyIdentifierProperties 함수는 키 식별자와 해당 속성을 열거합니다.
CryptEnumOIDFunction

CryptEnumOIDFunction 함수는 등록된 OID(개체 식별자) 함수를 열거합니다.
CryptEnumOIDInfo 미리 정의된 OID(개체 식별자) CRYPT_OID_INFO 구조체를 열거합니다. 이 함수는 미리 정의된 구조체와 등록된 모든 구조체 또는 선택한 OID 그룹으로 식별된 구조체만 열거합니다.
CryptEnumProvidersA 중요 이 API는 더 이상 사용되지 않습니다. (CryptEnumProvidersA)
CryptEnumProvidersW 중요 이 API는 더 이상 사용되지 않습니다. (CryptEnumProvidersW)
CryptEnumProviderTypesA 컴퓨터에서 지원되는 CSP(암호화 서비스 공급자)의 첫 번째 또는 다음 유형을 검색합니다. (ANSI)
CryptEnumProviderTypesW 컴퓨터에서 지원되는 CSP(암호화 서비스 공급자)의 첫 번째 또는 다음 유형을 검색합니다. (유니코드)
CryptExportKey CSP(암호화 서비스 공급자)에서 암호화 키 또는 키 쌍을 안전한 방식으로 내보냅니다.
CryptExportPKCS8 PKCS에서 프라이빗 키 내보내기(CryptExportPKCS8)
CryptExportPKCS8Ex PKCS(CryptExportPKCS8Ex)에서 프라이빗 키를 내보냅니다.
CryptExportPublicKeyInfo CryptExportPublicKeyInfo 함수는 공급자의 해당 프라이빗 키와 연결된 공개 키 정보를 내보냅니다. 이 함수의 업데이트된 버전은 CryptExportPublicKeyInfoEx를 참조하세요.
CryptExportPublicKeyInfoEx 공급자의 해당 프라이빗 키와 연결된 공개 키 정보를 내보냅니다.
CryptExportPublicKeyInfoFromBCryptKeyHandle 공급자의 해당 프라이빗 키와 연결된 공개 키 정보를 내보냅니다.

CryptFindCertificateKeyProvInfo 암호화 공급자 및 해당 컨테이너를 열거하여 인증서의 공개 키에 해당하는 프라이빗 키를 찾습니다.
CryptFindLocalizedName 지정된 이름의 지역화된 이름(예: "루트" 시스템 저장소의 지역화 이름)을 찾습니다.
CryptFindOIDInfo 지정된 키 형식 및 키와 일치하는 첫 번째 미리 정의되거나 등록된 CRYPT_OID_INFO 구조를 검색합니다. 검색은 지정된 OID 그룹 내의 OID(개체 식별자)로 제한될 수 있습니다.
CryptFormatObject CryptFormatObject 함수는 인코딩된 데이터의 형식을 지정하고 인증서 인코딩 유형에 따라 할당된 버퍼에 유니코드 문자열을 반환합니다.
CryptFreeOIDFunctionAddress CryptFreeOIDFunctionAddress 함수는 함수 핸들의 참조 수를 감소시켜 CryptGetOIDFunctionAddress 또는 CryptGetDefaultOIDFunctionAddress에서 반환된 핸들을 해제합니다.
CryptGenKey 임의의 암호화 세션 키 또는 퍼블릭/프라이빗 키 쌍을 생성합니다. 키 또는 키 쌍에 대한 핸들이 phKey로 반환됩니다. 그런 다음 키 핸들이 필요한 CryptoAPI 함수에서 필요에 따라 이 핸들을 사용할 수 있습니다.
CryptGenRandom 암호화된 임의 바이트로 버퍼를 채웁니다.
CryptGetAsyncParam CryptGetAsyncParam 함수(wincrypt.h)는 비동기 매개 변수 값을 설정합니다.
CryptGetDefaultOIDDllList CryptGetDefaultOIDDllList 함수는 지정된 함수 집합 및 인코딩 형식에 대해 등록된 OID(기본 개체 식별자) 함수를 포함하는 DLL 파일의 이름 목록을 가져옵니다.
CryptGetDefaultOIDFunctionAddress CryptGetDefaultOIDFunctionAddress 함수는 기본 함수 주소를 포함하는 DLL을 로드합니다.

CryptGetDefaultProviderA 로컬 컴퓨터 또는 현재 사용자에 대해 지정된 공급자 유형의 기본 CSP(암호화 서비스 공급자)를 찾습니다. (ANSI)
CryptGetDefaultProviderW 로컬 컴퓨터 또는 현재 사용자에 대해 지정된 공급자 유형의 기본 CSP(암호화 서비스 공급자)를 찾습니다. (유니코드)
CryptGetHashParam 해시 개체의 작업을 제어하는 데이터를 검색합니다.
CryptGetKeyIdentifierProperty CryptGetKeyIdentifierProperty는 지정된 키 식별자에서 특정 속성을 가져옵니다.
CryptGetKeyParam 키의 작업을 제어하는 데이터를 검색합니다.
CryptGetMessageCertificates CryptGetMessageCertificates 함수는 메시지의 인증서 및 CRL을 포함하는 열린 인증서 저장소의 핸들을 반환합니다. 이 함수는 공급자 형식 CERT_STORE_PROV_PKCS7 사용하여 CertOpenStore를 lpszStoreProvider 매개 변수로 호출합니다.
CryptGetMessageSignerCount CryptGetMessageSignerCount 함수는 서명된 메시지의 서명자 수를 반환합니다.
CryptGetObjectUrl 인증서, CTL(인증서 신뢰 목록) 또는 CRL(인증서 해지 목록)에서 원격 개체의 URL을 가져옵니다.
CryptGetOIDFunctionAddress 등록된 함수 및 설치된 함수 목록에서 인코딩 형식 및 OID(개체 식별자) 일치 항목을 검색합니다.
CryptGetOIDFunctionValue CryptGetOIDFunctionValue 함수는 OID와 연결된 값을 쿼리합니다.
CryptGetProvParam CSP(암호화 서비스 공급자)의 작업을 제어하는 매개 변수를 검색합니다.

CryptGetTimeValidObject 지정된 컨텍스트 및 시간 내에서 유효한 CRL, OCSP 응답 또는 CTL 개체를 검색합니다.
CryptGetUserKey 사용자의 두 퍼블릭/프라이빗 키 쌍 중 하나의 핸들을 검색합니다.
CryptHashCertificate CryptHashCertificate 함수는 서명을 포함하여 인증서의 인코딩된 전체 콘텐츠를 해시합니다.
CryptHashCertificate2 CNG 해시 공급자를 사용하여 데이터 블록을 해시합니다.
CryptHashData 지정된 해시 개체에 데이터를 추가합니다.
CryptHashMessage 메시지의 해시를 만듭니다.
CryptHashPublicKeyInfo CERT_PUBLIC_KEY_INFO 구조에서 공개 키 정보를 인코딩하고 인코딩된 바이트의 해시를 계산합니다.
CryptHashSessionKey 세션 키 개체의 암호화 해시를 계산합니다.
CryptHashToBeSigned 중요 이 API는 더 이상 사용되지 않습니다. (CryptHashToBeSigned)
CryptImportKey 키 BLOB에서 CSP(암호화 서비스 공급자)로 암호화 키를 전송합니다.
CryptImportPKCS8 PKCS에서 프라이빗 키 가져오기
CryptImportPublicKeyInfo 공개 키 정보를 공급자로 변환하고 가져오고 공개 키의 핸들을 반환합니다.

CryptImportPublicKeyInfoEx 중요 이 API는 더 이상 사용되지 않습니다. (CryptImportPublicKeyInfoEx)
CryptImportPublicKeyInfoEx2 공개 키를 OID(공개 키 개체 식별자)에 해당하는 CNG 비대칭 공급자로 가져오고 키에 대한 CNG 핸들을 반환합니다.
CryptInitOIDFunctionSet CryptInitOIDFunctionSet은 제공된 함수 집합 이름으로 식별된 OID 함수 집합의 핸들을 초기화하고 반환합니다.
CryptInstallDefaultContext 지정된 알고리즘의 기본 컨텍스트 공급자가 되도록 특정 공급자를 설치합니다.
CryptInstallOIDFunctionAddress CryptInstallOIDFunctionAddress 함수는 OID(호출 가능한 개체 식별자) 함수 주소 집합을 설치합니다.
CryptMemAlloc CryptMemAlloc 함수는 버퍼에 대한 메모리를 할당합니다. 할당된 버퍼를 반환하는 모든 Crypt32.lib 함수에서 사용됩니다.
CryptMemFree CryptMemFree 함수는 CryptMemAlloc 또는 CryptMemRealloc에서 할당한 메모리를 해제합니다.
CryptMemRealloc CryptMemRealloc 함수는 현재 버퍼에 할당된 메모리를 해제하고 새 버퍼에 대한 메모리를 할당합니다.
CryptMsgCalculateEncodedLength 메시지 유형, 인코딩 매개 변수 및 인코딩할 데이터의 총 길이가 지정된 경우 인코딩된 암호화 메시지에 필요한 최대 바이트 수를 계산합니다.
CryptMsgClose CryptMsgClose 함수는 암호화 메시지 핸들을 닫습니다. 이 함수를 호출할 때마다 메시지의 참조 수가 1씩 줄어듭니다. 참조 수가 0에 도달하면 메시지가 완전히 해제됩니다.
CryptMsgControl

CryptMsgUpdate 함수에 대한 최종 호출로 메시지가 디코딩된 후 제어 작업을 수행합니다.
CryptMsgCountersign 메시지의 기존 서명을 카운터 서명합니다.
CryptMsgCountersignEncoded 기존 PKCS에 대한 카운터 서명
CryptMsgDuplicate CryptMsgDuplicate 함수는 참조 횟수를 증가시켜 암호화 메시지 핸들을 복제합니다.
CryptMsgEncodeAndSignCTL CryptMsgEncodeAndSignCTL 함수는 CTL을 인코딩하고 인코딩된 CTL을 포함하는 서명된 메시지를 만듭니다. 이 함수는 먼저 pCtlInfo가 가리키는 CTL을 인코딩한 다음, CryptMsgSignCTL을 호출하여 인코딩된 메시지에 서명합니다.
CryptMsgGetAndVerifySigner CryptMsgGetAndVerifySigner 함수는 암호화 메시지의 서명을 확인합니다.
CryptMsgGetParam 암호화 메시지가 인코딩되거나 디코딩된 후 메시지 매개 변수를 가져옵니다.
CryptMsgOpenToDecode 디코딩을 위한 암호화 메시지를 열고 열린 메시지의 핸들을 반환합니다.
CryptMsgOpenToEncode 인코딩을 위한 암호화 메시지를 열고 열린 메시지의 핸들을 반환합니다.
CryptMsgSignCTL CryptMsgSignCTL 함수는 인코딩된 CTL을 포함하는 서명된 메시지를 만듭니다.
CryptMsgUpdate 암호화 메시지에 콘텐츠를 추가합니다.
CryptMsgVerifyCountersignatureEncoded SignerInfo 구조체(PKCS에서 정의한 대로)의 측면에서 역개인을 확인합니다.
CryptMsgVerifyCountersignatureEncodedEx

pbSignerInfoCounterSignature 매개 변수에 pbSignerInfo 매개 변수 구조의 encryptedDigest 필드의 암호화된 해시가 포함되어 있음을 확인합니다.
CryptQueryObject 인증서, 인증서 해지 목록 또는 인증서 신뢰 목록과 같은 암호화 API 개체의 내용에 대한 정보를 검색합니다.
CryptRegisterDefaultOIDFunction CryptRegisterDefaultOIDFunction은 지정된 인코딩 형식 및 함수 이름에 대해 호출할 기본 함수를 포함하는 DLL을 등록합니다. CryptRegisterOIDFunction과 달리 DLL에서 내보낼 함수 이름은 재정의할 수 없습니다.
CryptRegisterOIDFunction 지정된 인코딩 형식, 함수 이름 및 OID(개체 식별자)에 대해 호출할 함수가 포함된 DLL을 등록합니다.
CryptRegisterOIDInfo CryptRegisterOIDInfo 함수는 CRYPT_OID_INFO 구조에 지정된 OID 정보를 등록하여 레지스트리에 유지합니다.
CryptReleaseContext CSP(암호화 서비스 공급자) 및 키 컨테이너의 핸들을 해제합니다.
CryptRetrieveObjectByUrlA URL로 지정된 위치에서 PKI(공개 키 인프라) 개체를 검색합니다. (ANSI)
CryptRetrieveObjectByUrlW URL로 지정된 위치에서 PKI(공개 키 인프라) 개체를 검색합니다. (유니코드)
CryptRetrieveTimeStamp 타임스탬프를 요청하고 URL로 지정된 위치에서 TSA(타임스탬핑 기관)로 타임스탬프를 검색합니다.
CryptSetAsyncParam CryptSetAsyncParam 함수(wincrypt.h)는 비동기 매개 변수를 설정합니다.
CryptSetHashParam 초기 해시 콘텐츠 설정 및 특정 해시 알고리즘 선택을 포함하여 해시 개체의 작업을 사용자 지정합니다.

CryptSetKeyIdentifierProperty CryptSetKeyIdentifierProperty 함수는 지정된 키 식별자의 속성을 설정합니다. 이 함수는 pwszComputerName으로 식별된 컴퓨터에서 속성을 설정할 수 있습니다.
CryptSetKeyParam 세션 키 작업의 다양한 측면을 사용자 지정합니다.
CryptSetOIDFunctionValue CryptSetOIDFunctionValue 함수는 지정된 인코딩 형식, 함수 이름, OID 및 값 이름에 대한 값을 설정합니다.
CryptSetProviderA 현재 사용자의 기본 CSP(암호화 서비스 공급자)를 지정합니다. (ANSI)
CryptSetProviderExA 로컬 컴퓨터 또는 현재 사용자에 대해 지정된 공급자 유형의 기본 CSP(암호화 서비스 공급자)를 지정합니다. (ANSI)
CryptSetProviderExW 로컬 컴퓨터 또는 현재 사용자에 대해 지정된 공급자 유형의 기본 CSP(암호화 서비스 공급자)를 지정합니다. (유니코드)
CryptSetProviderW 현재 사용자의 기본 CSP(암호화 서비스 공급자)를 지정합니다. (유니코드)
CryptSetProvParam CSP(암호화 서비스 공급자)의 작업을 사용자 지정합니다. 이 함수는 일반적으로 CSP와 연결된 키 컨테이너에서 보안 설명자를 설정하여 해당 키 컨테이너의 프라이빗 키에 대한 액세스를 제어하는 데 사용됩니다.
CryptSignAndEncodeCertificate 인증서, CRL(인증서 해지 목록), CTL(인증서 신뢰 목록) 또는 인증서 요청을 인코딩하고 서명합니다.
CryptSignAndEncryptMessage CryptSignAndEncryptMessage 함수는 지정된 콘텐츠의 해시를 만들고, 해시에 서명하고, 콘텐츠를 암호화하고, 암호화된 콘텐츠와 서명된 해시를 해시한 다음, 암호화된 콘텐츠와 서명된 해시를 모두 인코딩합니다.

CryptSignCertificate CryptSignCertificate 함수는 인코딩된 서명된 콘텐츠의 "서명할" 정보에 서명합니다.
CryptSignHashA 데이터에 서명합니다. (CryptSignHashA)
CryptSignHashW 데이터에 서명합니다. (CryptSignHashW)
CryptSignMessage CryptSignMessage 함수는 지정된 콘텐츠의 해시를 만들고 해시에 서명한 다음 원본 메시지 콘텐츠와 서명된 해시를 모두 인코딩합니다.
CryptSignMessageWithKey 매개 변수에 지정된 CSP의 프라이빗 키를 사용하여 메시지에 서명합니다.
CryptStringToBinaryA 형식이 지정된 문자열을 바이트 배열로 변환합니다. (ANSI)
CryptStringToBinaryW 형식이 지정된 문자열을 바이트 배열로 변환합니다. (유니코드)
CryptUninstallDefaultContext 중요 이 API는 더 이상 사용되지 않습니다. (CryptUninstallDefaultContext)
CryptUnregisterDefaultOIDFunction CryptUnregisterDefaultOIDFunction은 지정된 인코딩 형식 및 함수 이름에 대해 호출할 기본 함수를 포함하는 DLL의 등록을 제거합니다.
CryptUnregisterOIDFunction 지정된 인코딩 형식, 함수 이름 및 OID에 대해 호출할 함수가 포함된 DLL 등록을 제거합니다.
CryptUnregisterOIDInfo CryptUnregisterOIDInfo 함수는 지정된 CRYPT_OID_INFO OID 정보 구조의 등록을 제거합니다. 등록 취소할 구조체는 구조체의 pszOID 및 dwGroupId 멤버로 식별됩니다.
CryptVerifyCertificateSignature

CERT_PUBLIC_KEY_INFO 구조에서 공개 키를 사용하여 인증서, CRL(인증서 해지 목록) 또는 인증서 요청의 서명을 확인합니다.
CryptVerifyCertificateSignatureEx 발급자의 공개 키를 사용하여 주체 인증서, 인증서 해지 목록, 인증서 요청 또는 keygen 요청의 서명을 확인합니다.
CryptVerifyDetachedMessageHash CryptVerifyDetachedMessageHash 함수는 분리된 해시를 확인합니다.
CryptVerifyDetachedMessageSignature CryptVerifyDetachedMessageSignature 함수는 분리된 서명 또는 서명이 포함된 서명된 메시지를 확인합니다.
CryptVerifyMessageHash CryptVerifyMessageHash 함수는 지정된 콘텐츠의 해시를 확인합니다.
CryptVerifyMessageSignature 서명된 메시지의 서명을 확인합니다.
CryptVerifyMessageSignatureWithKey 지정된 공개 키 정보를 사용하여 서명된 메시지의 서명을 확인합니다.
CryptVerifySignatureA 해시 개체의 서명을 확인합니다. (ANSI)
CryptVerifySignatureW 해시 개체의 서명을 확인합니다. (유니코드)
CryptVerifyTimeStampSignature 지정된 바이트 배열의 타임스탬프를 확인합니다.
GetEncSChannel 이 함수를 사용할 수 없습니다.
PFXExportCertStore 인증서를 내보내고, 사용 가능한 경우 참조된 인증서 저장소에서 연결된 프라이빗 키를 내보냅니다.

PFXExportCertStoreEx 인증서를 내보내고, 사용 가능한 경우 참조된 인증서 저장소에서 연결된 프라이빗 키를 내보냅니다.
PFXImportCertStore PFX BLOB을 가져오고 인증서 및 연결된 프라이빗 키가 포함된 저장소의 핸들을 반환합니다.
PFXIsPFXBlob PFXIsPFXBlob 함수는 BLOB의 외부 계층을 PFX 패킷으로 디코딩하려고 시도합니다.
PFXVerifyPassword PFXVerifyPassword 함수는 BLOB의 외부 계층을 PFX(개인 정보 교환) 패킷으로 디코딩하고 지정된 암호로 암호를 해독하려고 시도합니다. BLOB에서 데이터를 가져오지 않습니다.

콜백 함수

PCRYPT_DECRYPT_PRIVATE_KEY_FUNC 프라이빗 키의 암호를 해독하고 pbClearTextKey 매개 변수에서 암호 해독된 키를 반환합니다.
PCRYPT_ENCRYPT_PRIVATE_KEY_FUNC 프라이빗 키를 암호화하고 pbEncryptedKey 매개 변수에서 암호화된 콘텐츠를 반환합니다.
PCRYPT_RESOLVE_HCRYPTPROV_FUNC phCryptProv 매개 변수를 사용하여 가져올 키를 수신하여 CSP(암호화 서비스 공급자)에 대한 핸들을 반환합니다.
PFN_CERT_CHAIN_FIND_BY_ISSUER_CALLBACK 애플리케이션이 인증서 체인에 추가될 수 있는 인증서를 필터링할 수 있도록 하는 애플리케이션 정의 콜백 함수입니다.
PFN_CERT_CREATE_CONTEXT_SORT_FUNC 컨텍스트를 만들 때 정렬된 각 컨텍스트 항목에 대해 호출됩니다.
PFN_CERT_DLL_OPEN_STORE_PROV_FUNC 저장소 공급자에 의해 구현되며 저장소를 여는 데 사용됩니다.

PFN_CERT_ENUM_PHYSICAL_STORE CertEnumPhysicalStoreCallback 콜백 함수는 CertEnumPhysicalStore를 호출하여 찾은 각 물리적 저장소에 대한 정보를 형식화하고 제공합니다.
PFN_CERT_ENUM_SYSTEM_STORE CertEnumSystemStoreCallback 콜백 함수는 CertEnumSystemStore를 호출하여 찾은 각 시스템 저장소에 대한 정보를 형식화하고 제공합니다.
PFN_CERT_ENUM_SYSTEM_STORE_LOCATION CertEnumSystemStoreLocationCallback 콜백 함수는 CertEnumSystemStoreLocation 호출에서 찾은 각 시스템 저장소 위치에 대한 정보를 형식으로 지정하고 제공합니다.
PFN_CERT_STORE_PROV_CLOSE 저장소의 참조 수가 0으로 감소할 때 CertCloseStore에서 호출되는 애플리케이션 정의 콜백 함수입니다.
PFN_CERT_STORE_PROV_CONTROL CertStoreProvControl 콜백 함수는 CertControlStore API를 지원합니다. 모든 API의 매개 변수는 콜백으로 바로 전달됩니다. 자세한 내용은 CertControlStore를 참조하세요.
PFN_CERT_STORE_PROV_DELETE_CERT 저장소에서 인증서를 삭제하기 전에 CertDeleteCertificateFromStore에서 호출하는 애플리케이션 정의 콜백 함수입니다.
PFN_CERT_STORE_PROV_DELETE_CRL 저장소에서 CRL을 삭제하기 전에 CertDeleteCRLFromStore에서 호출되는 애플리케이션 정의 콜백 함수입니다.
PFN_CERT_STORE_PROV_READ_CERT 공급자의 인증서 컨텍스트 복사본을 읽는 애플리케이션 정의 콜백 함수입니다.
PFN_CERT_STORE_PROV_READ_CRL CRL 컨텍스트의 공급자 복사본을 읽는 애플리케이션 정의 콜백 함수입니다.
PFN_CERT_STORE_PROV_READ_CTL CertStoreProvReadCTL 콜백 함수는 공급자의 CTL 컨텍스트 복사본을 읽고 있는 경우 새 CTL 컨텍스트를 만들기 위해 호출됩니다.

PFN_CERT_STORE_PROV_SET_CERT_PROPERTY 인증서의 속성을 설정하기 전에 CertSetCertificateContextProperty에서 호출되는 애플리케이션 정의 콜백 함수입니다.
PFN_CERT_STORE_PROV_SET_CRL_PROPERTY CRL의 속성을 설정하기 전에 CertSetCRLContextProperty에서 호출되는 애플리케이션 정의 콜백 함수입니다.
PFN_CERT_STORE_PROV_SET_CTL_PROPERTY CertStoreProvSetCTLProperty 콜백 함수는 CTL에서 속성을 설정할 수 있는지 여부를 결정합니다.
PFN_CERT_STORE_PROV_WRITE_CERT 저장소에 추가하기 전에 CertAddEncodedCertificateToStore, CertAddCertificateContextToStore 및 CertAddSerializedElementToStore에서 호출되는 애플리케이션 정의 콜백 함수입니다.
PFN_CERT_STORE_PROV_WRITE_CRL 저장소에 추가하기 전에 CertAddEncodedCRLToStore, CertAddCRLContextToStore 및 CertAddSerializedElementToStore에서 호출되는 애플리케이션 정의 콜백 함수입니다.
PFN_CERT_STORE_PROV_WRITE_CTL CertStoreProvWriteCTL 콜백 함수는 CTL이 저장소에 추가되기 전에 CertAddEncodedCTLToStore, CertAddCTLContextToStore 또는 CertAddSerializedElementToStore에서 호출할 수 있습니다.
PFN_CMSG_CNG_IMPORT_CONTENT_ENCRYPT_KEY 이미 암호가 해독된 CEK(콘텐츠 암호화 키)를 가져옵니다.
PFN_CMSG_CNG_IMPORT_KEY_AGREE 키 계약 수신자를 위한 CEK(콘텐츠 암호화 키)의 암호를 해독합니다.
PFN_CMSG_CNG_IMPORT_KEY_TRANS 키 전송 수신자를 위한 CEK(콘텐츠 암호화 키)를 가져오고 암호 해독합니다.
PFN_CMSG_EXPORT_KEY_AGREE 봉투 메시지의 키 계약 수신자에 대한 콘텐츠 암호화 키를 암호화하고 내보냅니다.
PFN_CMSG_EXPORT_KEY_TRANS 봉투 메시지의 키 전송 수신자에 대한 콘텐츠 암호화 키를 암호화하고 내보냅니다.

PFN_CMSG_EXPORT_MAIL_LIST 봉투 메시지의 메일 그룹 수신자에 대한 콘텐츠 암호화 키를 암호화하고 내보냅니다.
PFN_CMSG_GEN_CONTENT_ENCRYPT_KEY 봉투 메시지의 콘텐츠를 암호화하는 데 사용되는 대칭 키를 생성합니다.
PFN_CMSG_IMPORT_KEY_AGREE 봉투 메시지의 키 전송 받는 사람에 대한 콘텐츠 암호화 키를 가져옵니다. (PFN_CMSG_IMPORT_KEY_AGREE)
PFN_CMSG_IMPORT_KEY_TRANS 봉투 메시지의 키 전송 받는 사람에 대한 콘텐츠 암호화 키를 가져옵니다. (PFN_CMSG_IMPORT_KEY_TRANS)
PFN_CMSG_IMPORT_MAIL_LIST 봉투 메시지의 키 전송 받는 사람에 대한 콘텐츠 암호화 키를 가져옵니다. (PFN_CMSG_IMPORT_MAIL_LIST)
PFN_CRYPT_ENUM_KEYID_PROP CRYPT_ENUM_KEYID_PROP 콜백 함수는 CryptEnumKeyIdentifierProperties 함수와 함께 사용됩니다.
PFN_CRYPT_ENUM_OID_FUNC CRYPT_ENUM_OID_FUNCTION 콜백 함수는 CryptEnumOIDFunction 함수와 함께 사용됩니다.
PFN_CRYPT_ENUM_OID_INFO CRYPT_ENUM_OID_INFO 콜백 함수는 CryptEnumOIDInfo 함수와 함께 사용됩니다.
PFN_CRYPT_EXPORT_PUBLIC_KEY_INFO_EX2_FUNC 공개 키 BLOB을 내보내고 인코딩하기 위해 CryptExportPublicKeyInfoEx에서 호출됩니다.
PFN_CRYPT_EXTRACT_ENCODED_SIGNATURE_PARAMETERS_FUNC 해시 알고리즘 식별자와 선택적으로 서명 매개 변수를 디코딩하고 반환하기 위해 호출됩니다.
PFN_CRYPT_GET_SIGNER_CERTIFICATE CryptGetSignerCertificateCallback 사용자가 제공한 콜백 함수는 CRYPT_VERIFY_MESSAGE_PARA 구조와 함께 사용하여 메시지 서명자의 인증서를 가져와서 확인합니다.

PFN_CRYPT_OBJECT_LOCATOR_PROVIDER_FLUSH 개체가 변경되었음을 지정합니다.
PFN_CRYPT_OBJECT_LOCATOR_PROVIDER_FREE 공급자가 반환한 개체를 해제합니다.
PFN_CRYPT_OBJECT_LOCATOR_PROVIDER_FREE_IDENTIFIER 개체 식별자에 대한 메모리를 해제합니다.
PFN_CRYPT_OBJECT_LOCATOR_PROVIDER_FREE_PASSWORD PFX(개인 정보 교환) 바이트 배열을 암호화하는 데 사용되는 암호를 해제합니다.
PFN_CRYPT_OBJECT_LOCATOR_PROVIDER_GET 개체를 검색합니다.
PFN_CRYPT_OBJECT_LOCATOR_PROVIDER_INITIALIZE 공급자를 초기화합니다.
PFN_CRYPT_OBJECT_LOCATOR_PROVIDER_RELEASE 공급자를 해제합니다.
PFN_CRYPT_SIGN_AND_ENCODE_HASH_FUNC 계산된 해시에 서명하고 인코딩하기 위해 호출됩니다.
PFN_CRYPT_VERIFY_ENCODED_SIGNATURE_FUNC 인코딩된 서명을 해독하고 계산된 해시와 비교하기 위해 호출됩니다.
PFN_IMPORT_PUBLIC_KEY_INFO_EX2_FUNC 공개 키 알고리즘 식별자를 디코딩하고, 알고리즘 공급자를 로드하고, 키 쌍을 가져오기 위해 CryptImportPublicKeyInfoEx2에서 호출됩니다.

구조체

AUTHENTICODE_EXTRA_CERT_CHAIN_POLICY_PARA

파일에 대한 인증서 체인 확인에 사용되는 정책 정보를 보유합니다.
AUTHENTICODE_EXTRA_CERT_CHAIN_POLICY_STATUS AUTHENTICODE_EXTRA_CERT_CHAIN_POLICY_STATUS 구조에는 파일의 체인 확인을 위한 추가 Authenticode 정책 정보가 있습니다.
AUTHENTICODE_TS_EXTRA_CERT_CHAIN_POLICY_PARA AUTHENTICODE_TS_EXTRA_CERT_CHAIN_POLICY_PARA 구조에는 파일의 인증서 체인 확인에 사용할 수 있는 타임스탬프를 정책 정보가 포함됩니다.
BLOBHEADER 키의 BLOB 형식 및 키에서 사용하는 알고리즘을 나타냅니다.
CERT_ACCESS_DESCRIPTION CERT_ACCESS_DESCRIPTION 구조체는 CERT_AUTHORITY_INFO_ACCESS 구조체의 멤버입니다.
CERT_ALT_NAME_ENTRY 다양한 이름 양식 중 하나에 대체 이름을 포함합니다.
CERT_ALT_NAME_INFO CERT_ALT_NAME_INFO 구조는 주체 또는 발급자 인증서, CRL(인증서 해지 목록) 및 CTL(인증서 신뢰 목록)에 대한 확장을 인코딩하고 디코딩하는 데 사용됩니다.
CERT_AUTHORITY_INFO_ACCESS 인증 기관 정보 액세스 및 주체 정보 액세스 인증서 확장을 나타내며 주체 또는 인증서 발급자의 추가 정보 및 서비스에 액세스하는 방법을 지정합니다.
CERT_AUTHORITY_KEY_ID_INFO 인증서 또는 CRL(인증서 해지 목록)에 서명하는 데 사용되는 키를 식별합니다.
CERT_AUTHORITY_KEY_ID2_INFO CERT_AUTHORITY_KEY_ID2_INFO 구조는 인증서 또는 CRL에 서명하는 데 사용되는 키를 식별합니다.
CERT_BASIC_CONSTRAINTS_INFO CERT_BASIC_CONSTRAINTS_INFO 구조에는 인증된 주체가 CA(인증 기관), 최종 엔터티 또는 둘 다로 작동할 수 있는지 여부를 나타내는 정보가 포함되어 있습니다.
CERT_BASIC_CONSTRAINTS2_INFO

CERT_BASIC_CONSTRAINTS2_INFO 구조에는 인증된 주체가 CA 또는 최종 엔터티 역할을 할 수 있는지 여부를 나타내는 정보가 포함되어 있습니다. 주체가 CA 역할을 할 수 있는 경우 인증 경로 길이 제약 조건을 지정할 수도 있습니다.
CERT_BIOMETRIC_DATA 생체 인식 데이터에 대한 정보를 포함합니다.
CERT_BIOMETRIC_EXT_INFO 생체 인식 정보 집합을 포함합니다.
CERT_CHAIN_CONTEXT 연결된 모든 단순 체인에 대한 요약 유효성 데이터를 나타내는 단순 인증서 체인 배열과 트러스트 상태 구조를 포함합니다.
CERT_CHAIN_ELEMENT CERT_CHAIN_ELEMENT 구조는 간단한 인증서 체인의 단일 요소입니다.
CERT_CHAIN_ENGINE_CONFIG 기본이 아닌 인증서 체인 엔진을 빌드하기 위한 매개 변수를 설정합니다. 사용된 엔진은 인증서 체인을 빌드하는 방법을 결정합니다.
CERT_CHAIN_FIND_ISSUER_PARA 인증서 체인을 빌드하기 위해 CertFindChainInStore 함수에 사용되는 정보를 포함합니다.
CERT_CHAIN_PARA CERT_CHAIN_PARA 구조는 인증서 체인을 빌드하는 데 사용할 검색 및 일치 조건을 설정합니다.
CERT_CHAIN_POLICY_PARA 인증서 체인 확인에 대한 정책 기준을 설정하기 위해 CertVerifyCertificateChainPolicy에 사용되는 정보를 포함합니다.
CERT_CHAIN_POLICY_STATUS 인증서 체인의 유효성을 검사할 때 CertVerifyCertificateChainPolicy 함수에서 반환하는 인증서 체인 상태 정보를 보유합니다.
CERT_CONTEXT 인증서의 인코딩된 표현과 디코딩된 표현을 모두 포함합니다.
CERT_CREATE_CONTEXT_PARA

CertCreateContext 함수를 호출할 때 사용할 수 있는 추가 값을 정의합니다.
CERT_CRL_CONTEXT_PAIR CERT_CRL_CONTEXT_PAIR 구조에는 인증서 컨텍스트 및 연결된 CRL 컨텍스트가 포함됩니다.
CERT_DH_PARAMETERS Diffie/Hellman 공개 키 알고리즘과 연결된 매개 변수를 포함합니다.
CERT_DSS_PARAMETERS DSS(디지털 서명 표준) 공개 키 알고리즘과 연결된 매개 변수를 포함합니다.
CERT_ECC_SIGNATURE ECDSA(타원 곡선 디지털 서명 알고리즘) 서명에 대한 r 및 s 값을 포함합니다.
CERT_EXTENSION CERT_EXTENSION 구조에는 인증서, CRL(인증서 해지 목록) 또는 CTL(인증서 신뢰 목록)에 대한 확장 정보가 포함됩니다.
CERT_EXTENSIONS CERT_EXTENSIONS 구조체에는 확장 배열이 포함되어 있습니다.
CERT_GENERAL_SUBTREE CERT_GENERAL_SUBTREE 구조체는 CERT_NAME_CONSTRAINTS_INFO 구조체에 사용됩니다. 이 구조체는 포함하거나 제외할 수 있는 인증서의 ID를 제공합니다.
CERT_HASHED_URL 해시된 URL을 포함합니다.
CERT_ID 인증서를 고유하게 식별하는 유연한 수단으로 사용됩니다.
CERT_INFO 인증서의 정보를 포함합니다.
CERT_ISSUER_SERIAL_NUMBER 인증서에 대한 발급자 및 발급자의 일련 번호를 포함하는 인증서의 고유 식별자 역할을합니다.
CERT_KEY_ATTRIBUTES_INFO

CERT_KEY_ATTRIBUTES_INFO 구조에는 인증되는 공개 키에 대한 선택적 추가 정보가 포함되어 있습니다.
CERT_KEY_CONTEXT CERT_KEY_CONTEXT_PROP_ID 속성과 연결된 데이터를 포함합니다.
CERT_KEY_USAGE_RESTRICTION_INFO CERT_KEY_USAGE_RESTRICTION_INFO 구조에는 인증서의 공개 키 사용에 대한 제한 사항이 포함되어 있습니다. 여기에는 키를 사용할 수 있는 키 및 정책을 사용하기 위한 목적이 포함됩니다.
CERT_KEYGEN_REQUEST_INFO Netscape 키 생성 요청에 저장된 정보를 포함합니다. 주체 및 주체 공개 키 BLOB은 인코딩됩니다.
CERT_LDAP_STORE_OPENED_PARA CERT_STORE_PROV_LDAP 공급자가 CERT_LDAP_STORE_OPENED_FLAG 플래그를 사용하여 쿼리를 수행하는 데 사용할 기존 LDAP 세션과 LDAP 쿼리 문자열을 모두 지정하여 지정하는 경우 CertOpenStore 함수와 함께 사용됩니다.
CERT_LOGOTYPE_AUDIO 오디오 로고 형식에 대한 정보를 포함합니다.
CERT_LOGOTYPE_AUDIO_INFO 오디오 로고 형식에 대한 자세한 정보를 포함합니다.
CERT_LOGOTYPE_DATA 로고 형식 데이터를 포함합니다.
CERT_LOGOTYPE_DETAILS 로고 형식에 대한 추가 정보를 포함합니다.
CERT_LOGOTYPE_EXT_INFO 로고 형식 정보 집합을 포함합니다.
CERT_LOGOTYPE_IMAGE 이미지 로고 형식에 대한 정보를 포함합니다.
CERT_LOGOTYPE_IMAGE_INFO 이미지 로고 형식에 대한 자세한 정보를 포함합니다.

CERT_LOGOTYPE_INFO 로고 형식 데이터에 대한 정보를 포함합니다.
CERT_LOGOTYPE_REFERENCE 로고 형식 참조 정보를 포함합니다.
CERT_NAME_CONSTRAINTS_INFO CERT_NAME_CONSTRAINTS_INFO 구조에는 특히 트러스트에서 허용되거나 제외되는 인증서에 대한 정보가 포함되어 있습니다.
CERT_NAME_INFO 주체 또는 발급자 이름을 포함합니다.
CERT_NAME_VALUE RDN(상대 고유 이름) 특성 값을 포함합니다.
CERT_OR_CRL_BLOB 인터넷 키 Exchange 메시지와 함께 사용할 인증서를 캡슐화합니다.
CERT_OR_CRL_BUNDLE 인터넷 키 Exchange 메시지와 함께 사용할 인증서 배열을 캡슐화합니다.
CERT_OTHER_LOGOTYPE_INFO 미리 정의되지 않은 로고 유형에 대한 정보를 포함합니다.
CERT_PAIR CERT_PAIR 구조에는 인증서와 쌍 간 인증서가 포함됩니다.
CERT_PHYSICAL_STORE_INFO 물리적 인증서 저장소에 대한 정보를 포함합니다.
CERT_POLICIES_INFO CERT_POLICIES_INFO 구조체에는 CERT_POLICY_INFO 배열이 포함되어 있습니다.
CERT_POLICY_CONSTRAINTS_INFO CERT_POLICY_CONSTRAINTS_INFO 구조에는 인증서를 신뢰할 수 있는 것으로 수락하기 위한 설정된 정책이 포함되어 있습니다.

CERT_POLICY_ID CERT_POLICY_ID 구조에는 인증서가 명시적으로 지원하는 인증서 정책 목록과 이러한 정책과 관련된 선택적 한정자 정보가 포함되어 있습니다.
CERT_POLICY_INFO CERT_POLICY_INFO 구조에는 정책을 지정하는 OID(개체 식별자)와 정책 한정자의 선택적 배열이 포함됩니다.
CERT_POLICY_MAPPING 발급자 도메인과 주체 도메인 정책 OID 간의 매핑을 포함합니다.
CERT_POLICY_MAPPINGS_INFO CERT_POLICY_MAPPINGS_INFO 구조는 두 도메인의 정책 OID 간에 매핑을 제공합니다.
CERT_POLICY_QUALIFIER_INFO CERT_POLICY_QUALIFIER_INFO 구조체에는 한정자 및 한정자별 추가 정보를 지정하는 OID(개체 식별자)가 포함되어 있습니다.
CERT_PRIVATE_KEY_VALIDITY CERT_PRIVATE_KEY_VALIDITY 구조는 인증서의 공개 키에 해당하는 프라이빗 키에 대한 유효한 시간 범위를 나타냅니다.
CERT_PUBLIC_KEY_INFO 공개 키와 해당 알고리즘을 포함합니다.
CERT_QC_STATEMENT QC(정규화된 인증서) 문 확장에 포함할 하나 이상의 문 시퀀스의 단일 문을 나타냅니다.
CERT_QC_STATEMENTS_EXT_INFO QC에 대한 정규화된 인증서(QC) 문 확장을 구성하는 하나 이상의 문 시퀀스를 포함합니다.
CERT_RDN CERT_RDN 구조체에는 CERT_RDN_ATTR 구조의 배열로 구성된 RDN(상대 고유 이름)이 포함되어 있습니다.
CERT_RDN_ATTR RDN(상대 고유 이름)의 단일 특성을 포함합니다. 전체 RDN은 CERT_RDN_ATTR 구조의 배열을 포함하는 CERT_RDN 구조체로 표현됩니다.

CERT_REQUEST_INFO CERT_REQUEST_INFO 구조에는 인증서 요청에 대한 정보가 포함됩니다. 주체, 주체 공개 키 및 특성 BLOB이 인코딩됩니다.
CERT_REVOCATION_CHAIN_PARA OCSP(독립 온라인 인증서 상태 프로토콜) 응답 서명자 인증서에 대한 체인을 빌드하는 데 사용되는 매개 변수를 포함합니다.
CERT_REVOCATION_CRL_INFO CRL(인증서 해지 목록) 해지 유형 처리기에 의해 업데이트된 정보를 포함합니다.
CERT_REVOCATION_INFO CERT_CHAIN_ELEMENT 인증서의 해지 상태 나타냅니다.
CERT_REVOCATION_PARA 확인할 컨텍스트의 발급자를 찾는 데 도움이 되도록 CertVerifyRevocation 함수에 대한 호출에 전달됩니다.
CERT_REVOCATION_STATUS 인증서의 해지 상태 대한 정보를 포함합니다.
CERT_SELECT_CHAIN_PARA 체인을 빌드하고 선택하는 데 사용되는 매개 변수를 포함합니다.
CERT_SELECT_CRITERIA CertSelectCertificateChains 함수에 전달되는 선택 조건을 지정합니다.
CERT_SERVER_OCSP_RESPONSE_CONTEXT 인코딩된 OCSP 응답을 포함합니다.
CERT_SIGNED_CONTENT_INFO CERT_SIGNED_CONTENT_INFO 구조에는 서명할 인코딩된 콘텐츠와 서명을 보관할 BLOB이 포함됩니다. ToBeSigned 멤버는 인코딩된 CERT_INFO, CRL_INFO, CTL_INFO 또는 CERT_REQUEST_INFO.
CERT_SIMPLE_CHAIN CERT_SIMPLE_CHAIN 구조에는 체인 요소의 배열과 배열이 나타내는 체인에 대한 요약 신뢰 상태 포함됩니다.

CERT_STORE_PROV_FIND_INFO 많은 저장소 공급자 콜백 함수에서 사용됩니다.
CERT_STORE_PROV_INFO CertOpenStore 함수를 사용하여 저장소를 열 때 설치된 CertDllOpenStoreProv 함수에서 반환되는 정보를 포함합니다.
CERT_STRONG_SIGN_PARA 인증서, CRL(인증서 해지 목록), OCSP(온라인 인증서 상태 프로토콜) 응답 및 PKCS에서 강력한 서명을 검사 데 사용되는 매개 변수를 포함합니다.
CERT_STRONG_SIGN_SERIALIZED_INFO 강력한 서명에 사용할 수 있는 서명 알고리즘/해시 알고리즘 및 공개 키 알고리즘/비트 길이 쌍을 포함합니다.
CERT_SYSTEM_STORE_INFO CERT_SYSTEM_STORE_INFO 구조에는 시스템 저장소와 함께 작동하는 함수에서 사용하는 정보가 포함됩니다. 현재 이 구조에는 필수 정보가 포함되어 있지 않습니다.
CERT_SYSTEM_STORE_RELOCATE_PARA CERT_SYSTEM_STORE_RELOCATE_PARA 구조에는 해당 함수의 dwFlags 매개 변수가 CERT_SYSTEM_STORE_RELOCATE_FLAG 설정된 경우 CertOpenStore에 전달할 데이터가 포함됩니다.
CERT_TEMPLATE_EXT 인증서 템플릿.
CERT_TRUST_LIST_INFO CTL의 유효한 사용을 나타내는 CERT_TRUST_LIST_INFO 구조체입니다.
CERT_TRUST_STATUS 인증서 체인의 인증서에 대한 신뢰 정보, 간단한 인증서 체인에 대한 요약 신뢰 정보 또는 단순 체인 배열에 대한 요약 정보를 포함합니다.
CERT_USAGE_MATCH 인증서 체인을 빌드하는 데 사용할 발급자 인증서를 식별하기 위한 조건을 제공합니다.
CERT_X942_DH_PARAMETERS

Diffie-Hellman 공개 키 알고리즘과 연결된 매개 변수를 포함합니다.
CERT_X942_DH_VALIDATION_PARAMS 필요에 따라 CERT_X942_DH_PARAMETERS 구조체의 멤버가 가리키고 추가 시드 정보를 포함합니다.
CMC_ADD_ATTRIBUTES_INFO 인증서에 추가할 인증서 특성을 포함합니다.
CMC_ADD_EXTENSIONS_INFO 인증서에 추가할 인증서 확장 제어 특성을 포함합니다.
CMC_DATA_INFO 태그가 지정된 다양한 정보를 전달하는 수단을 제공합니다. (CMC_DATA_INFO)
CMC_PEND_INFO CMC_STATUS_INFO 구조체의 가능한 멤버입니다.
CMC_RESPONSE_INFO 태그가 지정된 다양한 정보를 전달하는 수단을 제공합니다. (CMC_RESPONSE_INFO)
CMC_STATUS_INFO CMS를 통해 인증서 관리 메시지에 대한 상태 정보를 포함합니다.
CMC_TAGGED_ATTRIBUTE CMC_DATA_INFO 및 CMC_RESPONSE_INFO 구조체에 사용됩니다. (CMC_TAGGED_ATTRIBUTE)
CMC_TAGGED_CERT_REQUEST CMC_TAGGED_REQUEST 구조체에 사용됩니다.
CMC_TAGGED_CONTENT_INFO CMC_DATA_INFO 및 CMC_RESPONSE_INFO 구조체에 사용됩니다. (CMC_TAGGED_CONTENT_INFO)
CMC_TAGGED_OTHER_MSG CMC_DATA_INFO 및 CMC_RESPONSE_INFO 구조체에 사용됩니다. (CMC_TAGGED_OTHER_MSG)
CMC_TAGGED_REQUEST

CMC_DATA_INFO 구조에서 인증서를 요청하는 데 사용됩니다.
CMS_DH_KEY_INFO Diffie-Hellman 키 정보를 포함하기 위해 CryptSetKeyParam 함수의 KP_CMS_DH_KEY_INFO 매개 변수와 함께 사용됩니다.
CMS_KEY_INFO 사용되지 않습니다.
CMSG_CMS_RECIPIENT_INFO CryptMsgGetParam 함수와 함께 키 전송, 키 계약 또는 메일 목록 봉투 메시지 받는 사람에 대한 정보를 가져오는 데 사용됩니다.
CMSG_CMS_SIGNER_INFO 서명되거나 서명된 메시지와 봉투로 둘러싸인 메시지에 정의된 SignerInfo의 콘텐츠를 포함합니다.
CMSG_CNG_CONTENT_DECRYPT_INFO CNG(Cryptography API:_Next Generation) CEK(콘텐츠 암호화 키)의 가져오기 및 암호 해독을 위해 CryptMsgControl과 OID(개체 식별자) 설치 가능 함수 간에 전달된 모든 관련 정보를 포함합니다.
CMSG_CONTENT_ENCRYPT_INFO PFN_CMSG_GEN_CONTENT_ENCRYPT_KEY, PFN_CMSG_EXPORT_KEY_TRANS, PFN_CMSG_EXPORT_KEY_AGREE 및 PFN_CMSG_EXPORT_MAIL_LIST 함수 간에 공유되는 정보를 포함합니다.
CMSG_CTRL_ADD_SIGNER_UNAUTH_ATTR_PARA 서명된 메시지의 서명자에 인증되지 않은 특성을 추가하는 데 사용됩니다.
CMSG_CTRL_DECRYPT_PARA 키 전송 받는 사람에 대한 봉투 메시지의 암호를 해독하는 데 사용되는 정보를 포함합니다. dwCtrlType 매개 변수가 CMSG_CTRL_DECRYPT 경우 이 구조체는 CryptMsgControl에 전달됩니다.
CMSG_CTRL_DEL_SIGNER_UNAUTH_ATTR_PARA 서명된 메시지 서명자의 인증되지 않은 특성을 삭제하는 데 사용됩니다.
CMSG_CTRL_KEY_AGREE_DECRYPT_PARA 주요 계약 수신자에 대한 정보를 포함합니다.
CMSG_CTRL_KEY_TRANS_DECRYPT_PARA

키 전송 메시지 수신자에 대한 정보를 포함합니다.
CMSG_CTRL_MAIL_LIST_DECRYPT_PARA 메일 목록 메시지 받는 사람에 대한 정보를 포함합니다.
CMSG_CTRL_VERIFY_SIGNATURE_EX_PARA 메시지 서명을 확인하는 데 사용되는 정보를 포함합니다. 서명자 인덱스 및 서명자 공개 키가 포함됩니다.
CMSG_ENVELOPED_ENCODE_INFO 봉투 메시지를 인코딩하는 데 필요한 정보를 포함합니다. dwMsgType 매개 변수가 CMSG_ENVELOPED 경우 CryptMsgOpenToEncode에 전달됩니다.
CMSG_HASHED_ENCODE_INFO 해시된 메시지와 함께 사용됩니다. CryptMsgOpenToEncode 함수의 dwMsgType 매개 변수가 CMSG_ENVELOPED 경우 CryptMsgOpenToEncode 함수에 전달됩니다.
CMSG_KEY_AGREE_ENCRYPT_INFO 봉투 메시지의 모든 주요 계약 수신자에게 적용되는 암호화 정보를 포함합니다.
CMSG_KEY_AGREE_KEY_ENCRYPT_INFO 봉투 메시지의 키 규약 수신자에 대한 암호화된 키를 포함합니다.
CMSG_KEY_AGREE_RECIPIENT_ENCODE_INFO 키 계약 키 관리를 사용하는 메시지 수신자에 대한 정보를 포함합니다.
CMSG_KEY_AGREE_RECIPIENT_INFO 주요 규약 알고리즘에 사용되는 정보를 포함합니다.
CMSG_KEY_TRANS_ENCRYPT_INFO 봉투형 데이터의 키 전송 수신자에 대한 암호화 정보를 포함합니다.
CMSG_KEY_TRANS_RECIPIENT_ENCODE_INFO 메시지 수신자에 대한 인코딩된 키 전송 정보를 포함합니다.
CMSG_KEY_TRANS_RECIPIENT_INFO CMSG_KEY_TRANS_RECIPIENT_INFO 구조에는 주요 전송 알고리즘에 사용되는 정보가 포함되어 있습니다.

CMSG_MAIL_LIST_ENCRYPT_INFO 봉투형 데이터의 메일 그룹 수신자에 대한 암호화 정보를 포함합니다.
CMSG_MAIL_LIST_RECIPIENT_ENCODE_INFO CMSG_MAIL_LIST_RECIPIENT_ENCODE_INFO 구조는 KEK(콘텐츠 키 암호화 키)의 암호를 해독하기 위해 이전에 분산된 대칭 키와 함께 사용됩니다.
CMSG_MAIL_LIST_RECIPIENT_INFO 이전에 분산된 KEK(키 암호화 키)에 사용되는 정보를 포함합니다.
CMSG_RC2_AUX_INFO RC2 암호화 알고리즘에 대한 키의 비트 길이를 포함합니다.
CMSG_RC4_AUX_INFO CMSG_RC4_AUX_INFO 구조에는 RC4 암호화 알고리즘에 대한 키의 비트 길이가 포함됩니다. CMSG_ENVELOPED_ENCODE_INFO pvEncryptionAuxInfo 멤버는 이 구조체의 instance 가리키도록 설정할 수 있습니다.
CMSG_RECIPIENT_ENCODE_INFO 메시지 수신자의 콘텐츠 암호화 키 관리 유형에 대한 정보를 포함합니다.
CMSG_RECIPIENT_ENCRYPTED_KEY_ENCODE_INFO 메시지 내용의 암호를 해독하는 데 필요한 세션 키를 해독하는 데 사용되는 메시지 수신기에 대한 정보를 포함합니다.
CMSG_RECIPIENT_ENCRYPTED_KEY_INFO CMSG_RECIPIENT_ENCRYPTED_KEY_INFO 구조에는 개별 키 계약 수신자에 사용되는 정보가 포함됩니다.
CMSG_SIGNED_ENCODE_INFO dwMsgType이 CMSG_SIGNED 경우 CryptMsgOpenToEncode에 전달할 정보를 포함합니다.
CMSG_SIGNER_ENCODE_INFO 서명자 정보를 포함합니다. dwMsgType 매개 변수가 CMSG_SIGNED 경우 CryptMsgCountersign, CryptMsgCountersignEncoded에 전달되고, 필요에 따라 CMSG_SIGNED_ENCODE_INFO 구조체의 멤버로 CryptMsgOpenToEncode에 전달됩니다.
CMSG_SIGNER_INFO

CMSG_SIGNER_INFO 구조에는 PKCS의 콘텐츠가 포함됩니다.
CMSG_SP3_COMPATIBLE_AUX_INFO SP3 호환 암호화에 필요한 정보를 포함합니다.
CMSG_STREAM_INFO 단일 블록 처리가 아닌 데이터의 스트림 처리를 사용하도록 설정하는 데 사용됩니다.
CRL_CONTEXT CRL_CONTEXT 구조에는 CRL(인증서 해지 목록)의 인코딩된 표현과 디코딩된 표현이 모두 포함됩니다. CryptoAPI 함수에서 반환된 CRL 컨텍스트는 CertFreeCRLContext 함수를 호출하여 해제해야 합니다.
CRL_DIST_POINT 인증서 사용자가 참조하여 인증서가 해지되었는지 여부를 확인할 수 있는 단일 CRL(인증서 해지 목록) 배포 지점을 식별합니다.
CRL_DIST_POINT_NAME CRL을 가져올 수 있는 위치를 식별합니다.
CRL_DIST_POINTS_INFO 인증서 사용자가 인증서 해지 여부를 확인하기 위해 참조할 수 있는 CRL(인증서 해지 목록) 배포 지점 목록을 포함합니다.
CRL_ENTRY 해지된 단일 인증서에 대한 정보를 포함합니다. CRL_INFO 구조체의 멤버입니다.
CRL_FIND_ISSUED_FOR_PARA 주체와 인증서 발급자 모두의 인증서 컨텍스트를 포함합니다.
CRL_INFO CRL(인증서 해지 목록)의 정보를 포함합니다.
CRL_ISSUING_DIST_POINT CRL(인증서 해지 목록)에 나열된 인증서 종류에 대한 정보를 포함합니다.
CROSS_CERT_DIST_POINTS_INFO 동적 교차 인증서를 업데이트하는 데 사용되는 정보를 제공합니다.

CRYPT_AES_128_KEY_STATE AES(Advanced Encryption Standard) 암호에 대한 128비트 대칭 키 정보를 지정합니다.
CRYPT_AES_256_KEY_STATE AES(Advanced Encryption Standard) 암호에 대한 256비트 대칭 키 정보를 지정합니다.
CRYPT_ALGORITHM_IDENTIFIER 프라이빗 키를 암호화하는 데 사용되는 알고리즘을 지정합니다.
CRYPT_ATTRIBUTE CRYPT_ATTRIBUTE 구조체는 하나 이상의 값이 있는 특성을 지정합니다.
CRYPT_ATTRIBUTE_TYPE_VALUE 단일 특성 값을 포함합니다. Value 멤버의 CRYPT_OBJID_BLOB 인코딩됩니다.
CRYPT_ATTRIBUTES 특성 배열을 포함합니다.
CRYPT_BIT_BLOB 바이트 배열로 표시되는 비트 집합을 포함합니다.
CRYPT_BLOB_ARRAY CRYPT_DATA_BLOB 구조체의 배열을 포함합니다.
CRYPT_CONTENT_INFO PKCS에 인코딩된 데이터를 포함합니다.
CRYPT_CONTENT_INFO_SEQUENCE_OF_ANY 인증서의 Netscape 인증서 시퀀스를 나타내는 정보를 포함합니다.
CRYPT_CREDENTIALS CryptRetrieveObjectByUrl 또는 CryptGetTimeValidObject와 같은 원격 개체 검색 함수에 선택적 입력으로 전달할 수 있는 자격 증명에 대한 정보를 포함합니다.
CRYPT_DECODE_PARA CryptDecodeObjectEx 함수에서 메모리 할당 및 메모리 해제 콜백 함수에 대한 액세스를 제공하는 데 사용됩니다.

CRYPT_DECRYPT_MESSAGE_PARA CRYPT_DECRYPT_MESSAGE_PARA 구조에는 메시지 암호 해독에 대한 정보가 포함되어 있습니다.
CRYPT_DEFAULT_CONTEXT_MULTI_OID_PARA CryptInstallDefaultContext 함수와 함께 개체 식별자 문자열의 배열을 포함하는 데 사용됩니다.
CRYPT_ECC_CMS_SHARED_INFO CMS(암호화 메시지 구문) EnvelopedData 콘텐츠 형식에서 ECC(타원 곡선 암호화)를 사용할 때 키 암호화 키 정보를 나타냅니다.
CRYPT_ENCODE_PARA CryptEncodeObjectEx 함수에서 메모리 할당 및 메모리 해제 콜백 함수에 대한 액세스를 제공하는 데 사용됩니다.
CRYPT_ENCRYPT_MESSAGE_PARA 메시지를 암호화하는 데 사용되는 정보를 포함합니다.
CRYPT_ENCRYPTED_PRIVATE_KEY_INFO PKCS의 정보를 포함합니다.
CRYPT_ENROLLMENT_NAME_VALUE_PAIR 사용자를 대신하여 인증서 요청을 만드는 데 사용됩니다.
CRYPT_GET_TIME_VALID_OBJECT_EXTRA_INFO pExtraInfo 매개 변수의 CryptGetTimeValidObject 함수에 전달할 수 있는 선택적 추가 정보를 포함합니다.
CRYPT_HASH_MESSAGE_PARA 해시 메시지에 대한 데이터를 포함합니다.
CRYPT_INTEGER_BLOB CryptoAPI CRYPT_INTEGER_BLOB 구조체(wincrypt.h)는 임의의 바이트 배열에 사용되며 데이터 형식을 포함할 수 있는 개체에 유연성을 제공합니다.
CRYPT_KEY_PROV_INFO CRYPT_KEY_PROV_INFO 구조에는 CSP(암호화 서비스 공급자) 내의 키 컨테이너에 대한 정보가 포함됩니다.

CRYPT_KEY_PROV_PARAM 키 컨테이너 매개 변수에 대한 정보를 포함합니다.
CRYPT_KEY_SIGN_MESSAGE_PARA CSP(암호화 서비스 공급자) 및 메시지에 서명하는 데 사용되는 알고리즘에 대한 정보를 포함합니다.
CRYPT_KEY_VERIFY_MESSAGE_PARA 서명자에 대한 인증서 없이 서명된 메시지를 확인하는 데 필요한 정보를 포함합니다.
CRYPT_MASK_GEN_ALGORITHM RSA PKCS를 생성하는 데 사용되는 알고리즘을 식별합니다.
CRYPT_OBJECT_LOCATOR_PROVIDER_TABLE 개체 위치 공급자가 구현한 함수에 대한 포인터를 포함합니다.
CRYPT_OID_FUNC_ENTRY OID(개체 식별자) 및 관련 함수에 대한 포인터를 포함합니다.
CRYPT_OID_INFO OID(개체 식별자)에 대한 정보를 포함합니다.
CRYPT_PASSWORD_CREDENTIALSA cryptRetrieveObjectByUrl 또는 CryptGetTimeValidObject와 같은 원격 개체 검색 함수에 대한 선택적 입력으로 CRYPT_CREDENTIALS 구조에 사용할 사용자 이름 및 암호 자격 증명을 포함합니다. (ANSI)
CRYPT_PASSWORD_CREDENTIALSW cryptRetrieveObjectByUrl 또는 CryptGetTimeValidObject와 같은 원격 개체 검색 함수에 대한 선택적 입력으로 CRYPT_CREDENTIALS 구조에 사용할 사용자 이름 및 암호 자격 증명을 포함합니다. (유니코드)
CRYPT_PKCS12_PBE_PARAMS PKCS에 대한 암호화 키, IV(초기화 벡터) 또는 MAC(메시지 인증 코드) 키를 만드는 데 사용되는 매개 변수를 포함합니다.
CRYPT_PKCS8_EXPORT_PARAMS 프라이빗 키를 암호화하는 프라이빗 키 및 콜백 함수를 식별합니다.

CRYPT_PKCS8_EXPORT_PARAMS PKCS에서 프라이빗 키를 내보내는 CryptExportPKCS8Ex 함수에 대한 매개 변수로 사용됩니다.
CRYPT_PKCS8_IMPORT_PARAMS PKCS 포함
CRYPT_PRIVATE_KEY_INFO PrivateKey 필드에 지우기 텍스트 프라이빗 키(DER 인코딩됨)를 포함합니다. CRYPT_PRIVATE_KEY_INFO PKCS의 정보를 포함합니다.
CRYPT_PSOURCE_ALGORITHM 알고리즘을 식별하고(선택적으로) RSAES-OAEP 키 암호화에 대한 레이블 값을 식별합니다.
CRYPT_RC2_CBC_PARAMETERS szOID_RSA_RC2CBC 암호화에 사용되는 정보를 포함합니다.
CRYPT_RETRIEVE_AUX_INFO CryptRetrieveObjectByUrl 함수에 전달할 선택적 정보를 포함합니다.
CRYPT_RSA_SSA_PSS_PARAMETERS RSA PKCS에 대한 매개 변수를 포함합니다.
CRYPT_RSAES_OAEP_PARAMETERS RSAES-OAEP 키 암호화에 대한 매개 변수를 포함합니다.
CRYPT_SEQUENCE_OF_ANY 인코딩된 BLOB의 임의 목록을 포함합니다.
CRYPT_SIGN_MESSAGE_PARA CRYPT_SIGN_MESSAGE_PARA 구조에는 지정된 서명 인증서 컨텍스트를 사용하여 메시지에 서명하기 위한 정보가 포함되어 있습니다.
CRYPT_SMART_CARD_ROOT_INFO 인증서 컨텍스트와 연결된 스마트 카드 및 세션 ID를 포함합니다.
CRYPT_SMIME_CAPABILITIES 지원되는 기능의 우선 순위가 지정된 배열을 포함합니다.

CRYPT_SMIME_CAPABILITY CRYPT_SMIME_CAPABILITY 구조체는 단일 기능 및 관련 매개 변수를 지정합니다. 단일 기능은 우선 순위가 지정된 기능 기본 설정 목록을 지정할 수 있는 CRYPT_SMIME_CAPABILITIES 목록으로 그룹화됩니다.
CRYPT_TIME_STAMP_REQUEST_INFO 타임스탬핑에 사용됩니다.
CRYPT_TIMESTAMP_ACCURACY CRYPT_TIMESTAMP_INFO 구조체에서 타임 스탬프 토큰이 만들어진 UTC 시간의 시간 편차 정확도를 나타내기 위해 TSA(타임스탬핑 기관)에서 사용됩니다.
CRYPT_TIMESTAMP_CONTEXT 타임스탬프를 인코딩한 표현과 디코딩된 타임스탬프를 모두 포함합니다.
CRYPT_TIMESTAMP_INFO CMS(암호화 메시지 구문) 형식으로 서명된 데이터 콘텐츠 형식을 포함합니다.
CRYPT_TIMESTAMP_PARA 타임스탬프를 요청하는 추가 매개 변수를 정의합니다.
CRYPT_TIMESTAMP_REQUEST TimeStampReq 형식의 ASN.1(추상 구문 표기법 1) 정의에 해당하는 타임스탬프 요청 구조를 정의합니다.
CRYPT_TIMESTAMP_RESPONSE 내부적으로 ASN.1(추상 구문 표기법 1) Distinguished Encoding Rules(DER) 인코딩 응답을 캡슐화하는 데 사용됩니다.
CRYPT_URL_INFO URL 그룹화에 대한 정보를 포함합니다.
CRYPT_VERIFY_CERT_SIGN_STRONG_PROPERTIES_INFO 공개 키의 길이(비트)와 강력한 서명에 사용되는 서명 및 해시 알고리즘의 이름을 포함합니다.
CRYPT_VERIFY_MESSAGE_PARA CRYPT_VERIFY_MESSAGE_PARA 구조에는 서명된 메시지를 확인하는 데 필요한 정보가 포함되어 있습니다.

CRYPT_X942_OTHER_INFO CRYPT_X942_OTHER_INFO 구조에는 추가 키 생성 정보가 포함되어 있습니다.
CRYPTNET_URL_CACHE_FLUSH_INFO CUC(Cryptnet URL Cache) 서비스에서 URL 캐시 항목을 유지하기 위해 사용하는 만료 정보를 포함합니다.
CRYPTNET_URL_CACHE_PRE_FETCH_INFO CUC(Cryptnet URL Cache) 서비스에서 URL 캐시 항목을 유지하기 위해 사용하는 업데이트 정보를 포함합니다.
CRYPTNET_URL_CACHE_RESPONSE_INFO CUC(Cryptnet URL Cache) 서비스에서 URL 캐시 항목을 유지하기 위해 사용하는 응답 정보를 포함합니다.
CTL_ANY_SUBJECT_INFO 인증서 신뢰 목록(CTL)에서 일치시킬 SubjectAlgorithm 및 CertFindSubjectInCTL 호출의 CTL 항목 중 하나에서 일치시킬 SubjectIdentifier를 포함합니다.
CTL_CONTEXT CTL_CONTEXT 구조체에는 CTL의 인코딩된 표현과 디코딩된 표현이 모두 포함됩니다.
CTL_ENTRY CTL(인증서 신뢰 목록)의 요소입니다.
CTL_FIND_SUBJECT_PARA 인증서 신뢰 목록(CTL)을 찾기 위해 CTL_FIND_SUBJECT dwFindType 매개 변수를 사용하여 CertFindCTLInStore에서 사용하는 데이터를 포함합니다.
CTL_FIND_USAGE_PARA CTL_FIND_SUBJECT_PARA 구조체의 멤버이며 CertFindCTLInStore에서 사용됩니다.
CTL_INFO CTL(인증서 신뢰 목록)에 저장된 정보를 포함합니다.
CTL_USAGE CTL(인증서 신뢰 목록) 확장에 대한 OID(개체 식별자) 배열을 포함합니다.

CTL_USAGE_MATCH 인증서 체인을 빌드하는 데 사용되는 CTL(인증서 신뢰 목록)을 찾기 위한 매개 변수를 제공합니다.
CTL_VERIFY_USAGE_PARA CTL_VERIFY_USAGE_PARA 구조에는 CertVerifyCTLUsage에서 CTL 사용의 유효성을 설정하는 데 사용하는 매개 변수가 포함되어 있습니다.
CTL_VERIFY_USAGE_STATUS CertVerifyCTLUsage에서 반환된 CTL(인증서 신뢰 목록)에 대한 정보를 포함합니다.
DHPRIVKEY_VER3 키 BLOB에 포함된 특정 프라이빗 키와 관련된 정보를 포함합니다.
DHPUBKEY 키 BLOB에 포함된 특정 Diffie-Hellman 공개 키와 관련된 정보를 포함합니다.
DHPUBKEY_VER3 키 BLOB에 포함된 특정 공개 키와 관련된 정보를 포함합니다.
DSSSEED DSS 공개 키의 소수를 확인하는 데 사용할 수 있는 시드 및 카운터 값을 보유합니다.
EV_EXTRA_CERT_CHAIN_POLICY_PARA EV 정책 유효성 검사를 위해 전달되는 매개 변수를 지정합니다. 애플리케이션은 이 구조를 사용하여 확장 유효성 검사 인증서의 정책 한정자 플래그 중 애플리케이션에 중요한 정책 한정자 플래그를 나타내는 힌트를 API에 전달합니다.
EV_EXTRA_CERT_CHAIN_POLICY_STATUS CertVerifyCertificateChainPolicy 함수 호출에서 반환된 정책 플래그를 포함합니다.
HMAC_INFO HMAC_INFO 구조체는 해시 알고리즘과 HMAC 해시를 계산하는 데 사용할 내부 및 외부 문자열을 지정합니다.
HTTPSPolicyCallbackData SSL(Secure Sockets Layer) 클라이언트/서버 인증서 체인의 확인에 사용되는 정책 정보를 보유합니다.

OCSP_BASIC_RESPONSE_ENTRY 단일 인증서에 대한 현재 인증서 상태 포함합니다.
OCSP_BASIC_RESPONSE_INFO RFC 2560에서 지정한 기본 OCSP(온라인 인증서 상태 프로토콜) 응답을 포함합니다.
OCSP_BASIC_REVOKED_INFO 인증서가 해지된 이유를 포함합니다.
OCSP_BASIC_SIGNED_RESPONSE_INFO 서명이 있는 기본 OCSP(온라인 인증서 상태 프로토콜) 응답을 포함합니다.
OCSP_CERT_ID OCSP(온라인 인증서 상태 프로토콜) 요청 또는 응답에서 인증서를 식별하는 정보를 포함합니다.
OCSP_REQUEST_ENTRY OCSP(온라인 인증서 상태 프로토콜) 요청의 단일 인증서에 대한 정보를 포함합니다.
OCSP_REQUEST_INFO RFC 2560에서 지정한 온라인 인증서 상태 프로토콜(OCSP) 요청에 대한 정보를 포함합니다.
OCSP_RESPONSE_INFO 해당 온라인 인증서 상태 프로토콜(OCSP) 요청의 성공 또는 실패를 나타냅니다. 성공적인 요청의 경우 응답 정보의 형식과 값이 포함됩니다.
OCSP_SIGNATURE_INFO OCSP(온라인 인증서 상태 프로토콜) 요청 또는 응답에 대한 서명을 포함합니다.
OCSP_SIGNED_REQUEST_INFO 선택적 서명 정보가 있는 온라인 인증서 상태 프로토콜(OCSP) 요청에 대한 정보를 포함합니다.
PKCS12_PBES2_EXPORT_PARAMS 사용할 암호화 알고리즘에 대한 정보를 제공하기 위해 dwFlags에 대해 PKCS12_EXPORT_PBES2_PARAMS 플래그가 설정된 경우 PFXExportCertStoreEx 함수에 pvPara로 전달됩니다.
PROV_ENUMALGS

CSP(암호화 서비스 공급자)에서 지원하는 알고리즘에 대한 정보를 포함하도록 PP_ENUMALGS 매개 변수를 검색할 때 CryptGetProvParam 함수와 함께 사용됩니다.
PROV_ENUMALGS_EX CSP(암호화 서비스 공급자)에서 지원하는 알고리즘에 대한 정보를 포함하도록 PP_ENUMALGS_EX 매개 변수를 검색할 때 CryptGetProvParam 함수와 함께 사용됩니다.
ROOT_INFO_LUID 암호화 스마트 카드 루트 정보에 대한 LUID(로컬 고유 식별자)를 포함합니다.
RSAPUBKEY RSAPUBKEY 구조에는 키 BLOB에 포함된 특정 공개 키와 관련된 정보가 포함됩니다.
SCHANNEL_ALG SCHANNEL_ALG 구조에는 알고리즘 및 키 크기 정보가 포함됩니다. dwParam이 KP_SCHANNEL_ALG 설정된 경우 CryptSetKeyParam에서 pbData로 전달되는 구조체로 사용됩니다.
SSL_F12_EXTRA_CERT_CHAIN_POLICY_STATUS SSL_F12_EXTRA_CERT_CHAIN_POLICY_STATUS 구조는 체인의 인증서에 약한 암호화가 있는지 확인하고 타사 루트 인증서가 Microsoft 루트 프로그램 요구 사항을 준수하는지 확인합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winddi.h 헤더

아티클2023. 08. 27.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)
- 디스플레이 디바이스 참조
- 시스템 서비스

winddi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

BRUSHOBJ_hGetColorTransform BRUSHOBJ_hGetColorTransform 함수는 지정된 브러시의 색 변환을 검색합니다.
BRUSHOBJ_pvAllocRbrush BRUSHOBJ_pvAllocRbrush 함수는 드라이버가 지정된 브러시를 실현할 수 있는 메모리를 할당합니다.
BRUSHOBJ_pvGetRbrush BRUSHOBJ_pvGetRbrush 함수는 지정된 브러시의 드라이버 실현에 대한 포인터를 검색합니다.
BRUSHOBJ_ulGetBrushColor BRUSHOBJ_ulGetBrushColor 함수는 지정된 단색 브러시의 RGB 색을 반환합니다.
CLIPOBJ_bEnum CLIPOBJ_bEnum 함수는 지정된 클립 영역에서 사각형의 일괄 처리를 열거합니다. CLIPOBJ_cEnumStart 대한 이전 호출은 열거 순서를 결정합니다.
CLIPOBJ_cEnumStart CLIPOBJ_cEnumStart 함수는 지정된 클립 영역에서 사각형을 열거하기 위한 매개 변수를 설정합니다.
CLIPOBJ_ppoGetPath CLIPOBJ_ppoGetPath 함수는 지정된 클립 영역의 윤곽선을 포함하는 PATHOBJ 구조를 만듭니다.

DrvAlphaBlend DrvAlphaBlend 함수는 알파 혼합을 통해 비트 블록 전송 기능을 제공합니다.
DrvAssertMode DrvAssertMode 함수는 지정된 물리적 디바이스의 모드를 PDEV가 초기화될 때 지정된 모드 또는 하드웨어의 기본 모드로 설정합니다.
DrvBitBlt DrvBitBlt 함수는 디바이스 관리 표면 간, GDI 관리 표준 형식 비트맵 간 또는 디바이스 관리형 표면과 GDI 관리 표준 형식 비트맵 간에 일반적인 비트 블록 전송 기능을 제공합니다.
DrvCompletePDEV DrvCompletePDEV 함수는 생성되는 물리적 디바이스의 GDI 핸들을 저장합니다.
DrvCopyBits DrvCopyBits 함수는 디바이스 관리 래스터 표면과 GDI 표준 형식 비트맵 간에 변환됩니다.
DrvCreateDeviceBitmap DrvCreateDeviceBitmap 함수는 비트맵을 만들고 관리합니다.
DrvDeleteDeviceBitmap DrvDeleteDeviceBitmap 함수는 DrvCreateDeviceBitmap에서 만든 디바이스 비트맵을 삭제합니다.
DrvDeriveSurface DrvDeriveSurface 함수는 지정된 DirectDraw 표면에서 GDI 표면을 파생합니다.
DrvDescribePixelFormat DrvDescribePixelFormat 함수는 PIXELFORMATDESCRIPTOR 구조에 픽셀 형식 설명을 작성하여 디바이스 지정 PDEV의 픽셀 형식을 설명합니다.
DrvDestroyFont DrvDestroyFont 함수는 이제 글꼴 실현이 더 이상 필요하지 않으며 드라이버가 할당한 연결된 모든 데이터 구조를 해제할 수 있음을 드라이버에 알려줍니다.
DrvDisableDirectDraw DrvDisableDirectDraw 함수는 DirectDraw 사용을 위해 하드웨어를 사용하지 않도록 설정합니다.

DrvDisableDriver DrvDisableDriver 함수는 GDI에서 드라이버가 더 이상 필요하지 않으며 언로드할 준비가 되었음을 드라이버에 알리는 데 사용됩니다.
DrvDisablePDEV DrvDisablePDEV 함수는 지정된 PDEV가 더 이상 필요하지 않음을 드라이버에 알리기 위해 GDI에서 사용됩니다.
DrvDisableSurface DrvDisableSurface 함수는 GDI에서 현재 디바이스에 대해 DrvEnableSurface에서 만든 표면이 더 이상 필요하지 않음을 드라이버에 알리는 데 사용됩니다.
DrvDitherColor DrvDitherColor 함수는 디바이스 팔레트에 대해 디더링된 브러시를 만들도록 디바이스에 요청합니다.
DrvDrawEscape DrvDrawEscape 함수는 둘 이상의 함수 호출을 제공하는 진입점입니다. 특정 함수는 iEsc 매개 변수의 값에 따라 달라집니다.
DrvEnableDirectDraw DrvEnableDirectDraw 함수를 사용하면 DirectDraw에 하드웨어를 사용할 수 있습니다.
DrvEnableDriver DrvEnableDriver 함수는 드라이버 DLL에서 내보낸 초기 드라이버 진입점입니다.
DrvEnablePDEV DrvEnablePDEV 함수는 물리적 디바이스의 특성에 대한 설명을 GDI에 반환합니다.
DrvEnableSurface DrvEnableSurface 함수는 그릴 표면을 설정하고 지정된 물리적 디바이스와 연결합니다.
DrvEndDoc DrvEndDoc 함수는 렌더링을 위해 드라이버에 문서 보내기를 마쳤을 때 GDI에서 호출됩니다.
DrvEscape DrvEscape 함수는 디바이스 독립적 디바이스 드라이버 인터페이스에서 사용할 수 없는 디바이스에서 정보를 검색하는 데 사용됩니다. 특정 쿼리는 iEsc 매개 변수의 값에 따라 달라집니다.

DrvFillPath DrvFillPath 함수는 닫힌 경로 채우기를 처리하는 선택적 진입점입니다.
DrvFontManagement DrvFontManagement 함수는 PostScript 디바이스에 제공되는 선택적 진입점입니다.
DrvFree DrvFree 함수는 지정된 구조가 더 이상 필요하지 않음을 드라이버에 알리는 데 사용됩니다.
DrvGetDirectDrawInfo DrvGetDirectDrawInfo 함수는 그래픽 하드웨어의 기능을 반환합니다.
DrvGetGlyphMode DrvGetGlyphMode 함수는 GDI에 문자 모양 정보를 캐시하는 방법을 알려줍니다.
DrvGetModes DrvGetModes 함수는 지정된 디바이스에서 지원하는 모드를 나열합니다.
DrvGetTrueTypeFile DrvGetTrueTypeFile 함수는 메모리 매핑된 TrueType 글꼴 파일에 액세스합니다.
DrvGradientFill DrvGradientFill 함수는 지정된 기본 형식을 음영으로 지정합니다.
DrvIcmCheckBitmapBits DrvIcmCheckBitmapBits 함수는 지정된 비트맵의 픽셀이 지정된 변환의 디바이스 영역 내에 있는지 여부를 확인합니다.
DrvIcmCreateColorTransform DrvIcmCreateColorTransform 함수는 ICM 색 변환을 만듭니다.
DrvIcmDeleteColorTransform DrvIcmDeleteColorTransform 함수는 지정된 색 변환을 삭제합니다.
DrvIcmSetDeviceGammaRamp DrvIcmSetDeviceGammaRamp 함수는 지정된 디스플레이 디바이스의 하드웨어 감마 램프를 설정합니다.

DrvLineTo DrvLineTo 함수는 단일 단색 정수 전용 코스메틱 라인을 그립니다.
DrvLoadFontFile DrvLoadFontFile 함수는 글꼴 파일 로드 및 매핑과 관련된 정보를 GDI로부터 받습니다.
DrvMovePointer DrvMovePointer 함수는 포인터를 새 위치로 이동하고 GDI가 포인터 표시를 방해하지 않도록 합니다.
DrvNextBand DrvNextBand 함수는 드라이버가 다음 대역을 프린터로 보낼 수 있도록 물리적 페이지에 대한 밴드 그리기를 마쳤을 때 GDI에 의해 호출됩니다.
DrvNotify DrvNotify 함수를 사용하면 디스플레이 드라이버가 GDI의 특정 정보에 대한 알림을 받을 수 있습니다.
DrvPaint DrvPaint 함수는 더 이상 사용되지 않으며 Windows 2000 이상에서 GDI에서 더 이상 호출되지 않습니다. 새 드라이버는 하나 이상의 DrvFillPath, DrvStrokePath 또는 DrvStrokeAndFillPath를 구현해야 합니다.
DrvPlgBlt DrvPlgBlt 함수는 디바이스 관리형 표면과 GDI 관리형 표면의 조합 간에 회전 비트 블록 전송 기능을 제공합니다.
DrvQueryAdvanceWidths DrvQueryAdvanceWidths 함수는 지정된 문자 모양 집합에 대한 문자 전달 너비를 반환합니다.
DrvQueryDeviceSupport DrvQueryDeviceSupport 함수는 요청된 디바이스 관련 정보를 반환합니다.
DrvQueryDriverInfo DrvQueryDriverInfo 함수는 요청된 드라이버 관련 정보를 반환합니다.
DrvQueryFont DrvQueryFont 함수는 GDI에서 지정된 글꼴에 대한 IFIMETRICS 구조를 가져오는 데 사용됩니다.

DrvQueryFontCaps DrvQueryFontCaps 함수는 글꼴 드라이버의 기능을 정의합니다.
DrvQueryFontData DrvQueryFontData 함수는 실현된 글꼴에 대한 정보를 검색합니다.
DrvQueryFontFile DrvQueryFontFile 함수는 글꼴 파일 정보를 제공합니다.
DrvQueryFontTree DrvQueryFontTree 함수는 glyph variantsA 커닝 핸들에 대한 커닝 쌍 매핑을 포함하여 유니코드에서 문자 모양 핸들로 매핑 중 하나를 정의하는 구조체에 대한 포인터를 GDI에 제공합니다.
DrvQueryPerBandInfo 프린터 그래픽 DLL의 DrvQueryPerBandInfo 함수는 물리적 페이지에 대한 밴드 그리기를 시작하기 전에 GDI에서 호출되므로 드라이버는 GDI에 대역별 정보를 제공할 수 있습니다.
DrvQueryTrueTypeOutline DrvQueryTrueTypeOutline 함수는 네이티브 TrueType 형식의 문자 모양 윤곽선을 검색합니다.
DrvQueryTrueTypeTable DrvQueryTrueTypeTable 함수는 TrueType 글꼴 설명 파일의 특정 테이블에 액세스합니다.
DrvRealizeBrush DrvRealizeBrush 함수는 드라이버가 지정된 표면에 대해 지정된 브러시를 실현하도록 요청합니다.
DrvResetDevice DrvResetDevice 함수는 작동하지 않거나 응답하지 않는 디바이스를 다시 설정합니다.
DrvResetPDEV DrvResetPDEV 함수를 사용하면 Win32 애플리케이션이 ResetDC를 호출할 때 그래픽 드라이버가 이전 PDEV 구조에서 새 PDEV 구조로 드라이버의 상태를 전송할 수 있습니다.
DrvSaveScreenBits DrvSaveScreenBits 함수를 사용하면 디스플레이 드라이버가 표시된 이미지의 지정된 사각형을 저장하거나 복원합니다.

DrvSendPage 프린터 그래픽 DLL의 DrvSendPage 함수는 드라이버가 프린터로 페이지를 보낼 수 있도록 물리적 페이지 그리기를 마쳤을 때 GDI에 의해 호출됩니다.
DrvSetPalette DrvSetPalette 함수는 드라이버가 지정된 디바이스에 대한 팔레트를 실현할 것을 요청합니다.
DrvSetPixelFormat DrvSetPixelFormat 함수는 창의 픽셀 형식을 설정합니다.
DrvSetPointerShape DrvSetPointerShape 함수는 드라이버가 포인터를 디스플레이에서 끌도록 드라이버에 요청하는 데 사용됩니다. 새 포인터 셰이프를 설정하려고 합니다. 새 포인터를 디스플레이에 지정한 위치에 배치합니다.
DrvStartBanding DrvStartBanding 함수는 렌더링을 위해 드라이버에 물리적 페이지의 밴드를 보낼 준비가 되면 GDI에 의해 호출됩니다.
DrvStartDoc DrvStartDoc 함수는 렌더링을 위해 드라이버에 문서 보내기를 시작할 준비가 되면 GDI에서 호출됩니다.
DrvStartPage DrvStartPage 함수는 렌더링을 위해 드라이버에 물리적 페이지의 콘텐츠를 보낼 준비가 되면 GDI에서 호출됩니다.
DrvStretchBlt DrvStretchBlt 함수는 디바이스 관리 표면과 GDI 관리형 표면의 조합 간에 확장 비트 블록 전송 기능을 제공합니다.
DrvStretchBltROP DrvStretchBltROP 함수는 ROP를 사용하여 스트레치 비트 블록 전송을 수행합니다.
DrvStrokeAndFillPath DrvStrokeAndFillPath 함수는 스트로크(윤곽선)를 만들고 경로를 동시에 채웁니다.
DrvStrokePath

DrvStrokePath 함수는 경로를 스트로크(윤곽선)합니다.
DrvSwapBuffers DrvSwapBuffers 함수는 지정된 표면에 창의 연결된 숨겨진 버퍼의 내용을 표시합니다.
DrvSynchronize DrvSynchronize 함수는 GDI가 디바이스 관리 표면에 액세스해야 한다고 드라이버에 알립니다. 이 함수를 사용하면 디바이스의 공동 프로세서에서 수행하는 비동기 그리기 작업을 GDI 액세스와 조정할 수 있습니다.
DrvSynchronizeSurface DrvSynchronizeSurface 함수는 지정된 표면에 GDI를 써야 한다는 것을 드라이버에 알립니다. 이 함수를 사용하면 디바이스의 공동 프로세서에서 수행하는 그리기 작업을 GDI와 조율할 수 있습니다.
DrvTextOut DrvTextOut 함수는 드라이버가 지정된 위치에서 문자 모양 집합을 렌더링하도록 요구하는 GDI의 진입점입니다.
DrvTransparentBlt DrvTransparentBlt 함수는 투명도를 갖춘 비트 블록 전송 기능을 제공합니다.
DrvUnloadFontFile DrvUnloadFontFile 함수는 지정된 글꼴 파일이 더 이상 필요하지 않음을 글꼴 드라이버에 알릴 수 있습니다.
EngAcquireSemaphore EngAcquireSemaphore 함수는 호출 스레드에서 단독 액세스를 위해 세마포와 연결된 리소스를 획득합니다.
EngAllocMem EngAllocMem 함수는 메모리 블록을 할당하고 할당 전에 호출자 제공 태그를 삽입합니다.
EngAllocPrivateUserMem EngAllocPrivateUserMem 함수는 지정된 프로세스의 주소 공간에서 사용자 메모리 블록을 할당하고 할당 전에 호출자가 제공한 태그를 삽입합니다.
EngAllocUserMem EngAllocUserMem 함수는 현재 프로세스의 주소 공간에서 메모리 블록을 할당하고 할당 전에 호출자가 제공한 태그를 삽입합니다.

EngAlphaBlend EngAlphaBlend 함수는 알파 혼합을 통해 비트 블록 전송 기능을 제공합니다.
EngAssociateSurface EngAssociateSurface 함수는 지정된 표면을 지정된 디바이스에 속하는 것으로 표시합니다.
EngBitBlt EngBitBlt 함수는 디바이스 관리 표면 간 또는 디바이스 관리형 표면과 GDI 관리 표준 형식 비트맵 간에 일반적인 비트 블록 전송 기능을 제공합니다.
EngBugCheckEx EngBugCheckEx 함수는 호출자가 복구할 수 없는 오류를 발견하면 제어된 방식으로 시스템을 다운합니다. 그러면 호출자가 계속 실행되면 시스템이 손상됩니다.
EngCheckAbort EngCheckAbort 함수를 사용하면 프린터 그래픽 DLL에서 인쇄 작업을 종료해야 하는지 여부를 확인할 수 있습니다.
EngClearEvent EngClearEvent 함수는 지정된 이벤트 개체를 부호 없는 상태로 설정합니다.
EngComputeGlyphSet EngComputeGlyphSet 함수는 디바이스에서 지원되는 문자 모양 집합을 계산합니다.
EngControlSprites EngControlSprites 함수는 지정된 WNDOBJ 영역에서 스프라이트를 눈물을 흘리거나 다시 그을어 줍니다.
EngCopyBits EngCopyBits 함수는 디바이스 관리 래스터 표면과 GDI 표준 형식 비트맵 간에 변환됩니다.
EngCreateBitmap EngCreateBitmap 함수는 GDI가 비트맵을 만들고 관리해 줄 것을 요청합니다.
EngCreateClip EngCreateClip 함수는 드라이버가 콜백에서 사용하는 CLIPOBJ 구조를 만듭니다.

EngCreateDeviceBitmap EngCreateDeviceBitmap 함수는 GDI에 디바이스 비트맵에 대한 핸들을 만들도록 요청합니다.
EngCreateDeviceSurface EngCreateDeviceSurface 함수는 드라이버가 관리할 디바이스 표면에 대한 핸들을 만들고 반환합니다.
EngCreateDriverObj EngCreateDriverObj 함수는 DRIVEROBJ 구조를 만듭니다.
EngCreateEvent EngCreateEvent 함수는 디스플레이 드라이버와 비디오 미니포트 드라이버 간에 하드웨어 액세스를 동기화하는 데 사용할 수 있는 동기화 이벤트 개체를 만듭니다.
EngCreatePalette EngCreatePalette 함수는 RGB 팔레트를 만들기 위해 GDI에 요청을 보냅니다.
EngCreatePath EngCreatePath 함수는 드라이버의 임시 사용을 위한 경로를 할당합니다.
EngCreateSemaphore EngCreateSemaphore 함수는 세마포 개체를 만듭니다.
EngCreateWnd EngCreateWnd 함수는 hwnd에서 참조하는 창에 대한 WNDOBJ 구조를 만듭니다.
EngDebugBreak EngDebugBreak 함수로 인해 현재 프로세스의 중단점이 발생합니다.
EngDebugPrint EngDebugPrint 함수는 지정된 디버그 메시지를 커널 디버거에 출력합니다.
EngDeleteClip EngDeleteClip 함수는 EngCreateClip에 의해 할당된 CLIPOBJ 구조를 삭제합니다.
EngDeleteDriverObj EngDeleteDriverObj 함수는 디바이스 관리 리소스를 추적하는 데 사용되는 핸들을 해제합니다.

EngDeleteEvent EngDeleteEvent 함수는 지정된 이벤트 개체를 삭제합니다.
EngDeleteFile EngDeleteFile 함수는 파일을 삭제합니다.
EngDeletePalette EngDeletePalette 함수는 GDI에 지정된 팔레트를 삭제하라는 요청을 보냅니다.
EngDeletePath EngDeletePath 함수는 EngCreatePath에서 이전에 할당한 경로를 삭제합니다.
EngDeleteSafeSemaphore EngDeleteSafeSemaphore 함수는 지정된 안전 세마포에 대한 참조를 제거합니다.
EngDeleteSemaphore EngDeleteSemaphore 함수는 시스템의 리소스 목록에서 세마포 개체를 삭제합니다.
EngDeleteSurface EngDeleteSurface 함수는 지정된 표면을 삭제합니다.
EngDeleteWnd EngDeleteWnd 함수는 WNDOBJ 구조를 삭제합니다.
EngDeviceIoControl EngDeviceIoControl 함수는 지정된 비디오 미니포트 드라이버에 제어 코드를 보내 디바이스가 지정된 작업을 수행하도록 합니다.
EngDitherColor EngDitherColor 함수는 지정된 RGB 색과 근사치를 나타내는 표준 8x8 디더를 반환합니다.
EngEnumForms EngEnumForms 함수는 지정된 프린터에서 지원하는 양식을 열거합니다.
EngEraseSurface EngEraseSurface 함수는 GDI를 호출하여 표면을 지웁니다. 표면에 지정된 사각형이 지정된 색으로 채워집니다.

EngFillPath EngFillPath 함수는 경로를 채웁니다.
EngFindImageProcAddress EngFindImageProcAddress 함수는 실행 가능한 모듈 내에서 함수의 주소를 반환합니다.
EngFindResource EngFindResource 함수는 모듈에서 리소스의 위치를 결정합니다.
EngFntCacheAlloc EngFntCacheAlloc 함수는 캐시된 메모리에 저장할 글꼴에 대한 스토리지를 할당합니다.
EngFntCacheFault 글꼴 드라이버에서 글꼴 데이터 캐시를 읽거나 쓰는 동안 오류가 발생하면 EngFntCacheFault 함수는 글꼴 엔진에 오류를 보고합니다.
EngFntCacheLookUp EngFntCacheLookUp 함수는 캐시된 글꼴 파일 데이터의 주소를 검색합니다.
EngFreeMem EngFreeMem 함수는 시스템 메모리 블록을 할당 취소합니다.
EngFreeModule EngFreeModule 함수는 시스템 메모리에서 파일의 매핑을 해제합니다.
EngFreePrivateUserMem EngFreePrivateUserMem 함수는 개인 사용자 메모리 블록을 할당 취소합니다.
EngFreeUserMem EngFreeUserMem 함수는 사용자 메모리 블록을 할당 취소합니다.
EngGetCurrentCodePage EngGetCurrentCodePage 함수는 시스템의 기본 OEM 및 ANSI 코드 페이지를 반환합니다.
EngGetCurrentProcessId EngGetCurrentProcessId 함수는 애플리케이션의 현재 프로세스를 식별합니다.

EngGetCurrentThreadId EngGetCurrentThreadId 함수는 애플리케이션의 현재 스레드를 식별합니다.
EngGetDriverName EngGetDriverName 함수는 드라이버의 DLL 이름을 반환합니다.
EngGetFileChangeTime EngGetFileChangeTime 함수는 파일의 마지막 쓰기 시간을 검색합니다.
EngGetFilePath EngGetFilePath 함수는 지정된 글꼴 파일과 연결된 파일 경로를 결정합니다.
EngGetForm EngGetForm 함수는 지정된 양식에 대한 FORM_INFO_1 세부 정보를 가져옵니다.
EngGetLastError EngGetLastError 함수는 호출 스레드에 대해 GDI가 기록한 마지막 오류 코드를 반환합니다.
EngGetPrinter EngGetPrinter 함수는 지정된 프린터에 대한 정보를 검색합니다.
EngGetPrinterData EngGetPrinterData 함수는 지정된 프린터에 대한 구성 데이터를 검색합니다.
EngGetPrinterDataFileName EngGetPrinterDataFileName 함수는 프린터 데이터 파일의 문자열 이름을 검색합니다.
EngGetPrinterDriver EngGetPrinterDriver 함수는 지정된 프린터에 대한 드라이버 데이터를 검색합니다.
EngGetProcessHandle EngGetProcessHandle 함수는 현재 클라이언트 프로세스에 대한 핸들을 검색합니다.
EngGetType1FontList EngGetType1FontList 함수는 로컬 및 원격으로 모두 설치된 PostScript Type 1 글꼴 목록을 검색합니다.

EngGradientFill EngGradientFill 함수는 지정된 기본 형식을 음영으로 지정합니다.
EngHangNotification EngHangNotification 함수는 시스템에 지정된 디바이스가 작동하지 않거나 응답하지 않음을 알 수 있습니다.
EngInitializeSafeSemaphore EngInitializeSafeSemaphore 함수는 지정된 안전 세마포를 초기화합니다.
EngIsSemaphoreOwned EngIsSemaphoreOwned 함수는 스레드가 지정된 세마포를 보유하는지 여부를 결정합니다.
EngIsSemaphoreOwnedByCurrentThread EngIsSemaphoreOwnedByCurrentThread 함수는 현재 실행 중인 스레드에 지정된 세마포가 있는지 여부를 결정합니다.
EngLineTo EngLineTo 함수는 단일 단색 정수 전용 코스메틱 라인을 그립니다.
EngLoadImage EngLoadImage 함수는 지정된 실행 파일을 커널 모드 메모리에 로드합니다.
EngLoadModule EngLoadModule 함수는 읽기 위해 지정된 데이터 모듈을 시스템 메모리에 로드합니다.
EngLoadModuleForWrite EngLoadModuleForWrite 함수는 지정된 실행 모듈을 쓰기 위해 시스템 메모리에 로드합니다.
EngLockDirectDrawSurface EngLockDirectDrawSurface 함수는 DirectDraw 표면의 커널 모드 핸들을 잠급니다.
EngLockDriverObj EngLockDriverObj 함수는 호출 스레드에 대해 이 개체에 대한 단독 잠금을 만듭니다.
EngLockSurface

EngLockSurface 함수는 지정된 표면에 대한 사용자 개체를 만듭니다. 이 함수는 드라이버가 만드는 표면에 대한 액세스 권한을 제공합니다.
EngLpkInstalled EngLpkInstalled 함수는 언어 팩이 시스템에 설치되어 있는지 여부를 결정합니다.
EngMapEvent EngMapEvent 함수는 사용자 모드 이벤트 개체를 커널 모드에 매핑합니다.
EngMapFile EngMapFile 함수는 파일을 만들거나 열고 시스템 공간에 매핑합니다.
EngMapFontFile EngMapFontFile 함수는 사용되지 않습니다. 대신 EngMapFontFileFD를 사용합니다.
EngMapFontFileFD EngMapFontFileFD 함수는 필요한 경우 글꼴 파일을 시스템 메모리에 매핑하고 파일의 글꼴 데이터의 기본 위치에 대한 포인터를 반환합니다.
EngMapModule EngMapModule 함수는 EngLoadModule, EngLoadModuleForWrite, EngLoadImage 또는 EngMapFile에 의해 로드된 파일의 주소와 크기를 반환합니다.
EngMarkBandingSurface EngMarkBandingSurface 함수는 지정된 표면을 밴딩 표면으로 표시합니다.
EngModifySurface EngModifySurface 함수는 드라이버에서 만든 표면의 특성에 대해 GDI에 알립니다.
EngMovePointer EngMovePointer 함수는 디바이스에서 엔진 관리 포인터를 이동합니다.
EngMulDiv EngMulDiv 함수는 두 개의 32비트 값을 곱한 다음 64비트 결과를 세 번째 32비트 값으로 나눕니다.
EngMultiByteToUnicodeN EngMultiByteToUnicodeN 함수는 현재 ANSI 코드 페이지를 사용하여 지정된 ANSI 소스 문자열을 유니코드 문자열로 변환합니다.

EngMultiByteToWideChar EngMultiByteToWideChar 함수는 지정된 코드 페이지를 사용하여 ANSI 소스 문자열을 와이드 문자열로 변환합니다.
EngPaint EngPaint 함수를 사용하면 GDI가 지정된 영역을 그립니다.
EngPlgBlt EngPlgBlt 함수는 GDI가 회전 비트 블록 전송을 수행하도록 합니다.
EngProbeForRead EngProbeForRead 함수는 읽기 접근성을 위해 구조를 검색합니다.
EngProbeForReadAndWrite EngProbeForReadAndWrite 함수는 읽기 및 쓰기 접근성을 위해 구조를 검색합니다.
EngQueryDeviceAttribute EngQueryDeviceAttribute 함수를 사용하면 드라이버가 디바이스의 특정 특성에 대해 시스템을 쿼리할 수 있습니다.
EngQueryFileTimeStamp EngQueryFileTimeStamp 함수는 파일의 타임스탬프를 반환합니다.
EngQueryLocalTime EngQueryLocalTime 함수는 현지 시간을 쿼리합니다.
EngQueryPalette EngQueryPalette 함수는 특성에 대해 지정된 팔레트를 쿼리합니다.
EngQueryPerformanceCounter EngQueryPerformanceCounter 함수는 성능 카운터를 쿼리합니다.
EngQueryPerformanceFrequency EngQueryPerformanceFrequency 함수는 성능 카운터의 빈도를 쿼리합니다.
EngQuerySystemAttribute EngQuerySystemAttribute 함수는 프로세서별 또는 시스템별 기능을 쿼리합니다.

EngReadStateEvent EngReadStateEvent 함수는 지정된 이벤트 개체:_signaled 또는 부호 없는 이벤트의 현재 상태를 반환합니다.
EngReleaseSemaphore EngReleaseSemaphore 함수는 지정된 세마포를 해제합니다.
EngRestoreFloatingPointState EngRestoreFloatingPointState 함수는 드라이버가 부동 소수점 또는 MMX 하드웨어 지침을 사용한 후 Windows 2000(이상) 커널 부동 소수점 상태를 복원합니다.
EngSaveFloatingPointState EngSaveFloatingPointState 함수는 현재 Windows 2000 이상 커널 부동 소수점 상태를 저장합니다.
EngSecureMem EngSecureMem 함수는 메모리에서 지정된 주소 범위를 잠급니다.
EngSetEvent EngSetEvent 함수는 지정된 이벤트 개체를 신호 상태로 설정하고 이벤트 개체의 이전 상태를 반환합니다.
EngSetLastError EngSetLastError 함수는 GDI가 애플리케이션에서 검색할 수 있는 오류 코드를 보고하도록 합니다.
EngSetPointerShape EngSetPointerShape 함수는 호출 드라이버에 대한 포인터 셰이프를 설정합니다.
EngSetPointerTag EngSetPointerTag 함수는 Windows 2000 이상 운영 체제 버전에서는 사용되지 않습니다.
EngSetPrinterData EngSetPrinterData 함수는 Windows 2000 이상에서 사용되지 않습니다. 이전 버전의 Windows EngSetPrinterData에서는 지정된 프린터의 구성 데이터를 설정합니다.
EngSort EngSort 함수는 지정된 목록에서 빠른 정렬을 수행합니다.

EngStretchBlt EngStretchBlt 함수는 GDI가 확장 비트 블록 전송을 수행하도록 합니다.
EngStretchBltROP EngStretchBltROP 함수는 ROP를 사용하여 스트레치 비트 블록 전송을 수행합니다.
EngStrokeAndFillPath EngStrokeAndFillPath 함수는 GDI가 경로를 채우고 동시에 스트로크하도록 합니다.
EngStrokePath EngStrokePath 함수는 GDI가 지정된 경로를 스트로크할 것을 요청합니다.
EngTextOut EngTextOut 함수를 사용하면 GDI가 지정된 위치에서 문자 모양 집합을 렌더링합니다.
EngTransparentBlt EngTransparentBlt 함수는 투명도를 갖춘 비트 블록 전송 기능을 제공합니다.
EngUnicodeToMultiByteN EngUnicodeToMultiByteN 함수는 현재 ANSI 코드 페이지를 사용하여 지정된 유니코드 문자열을 ANSI 문자열로 변환합니다.
EngUnloadImage EngUnloadImage 함수는 EngLoadImage에서 로드한 이미지를 언로드합니다.
EngUnlockDirectDrawSurface EngUnlockDirectDrawSurface 함수는 지정된 표면에서 잠금을 해제합니다.
EngUnlockDriverObj EngUnlockDriverObj 함수를 사용하면 GDI가 드라이버 개체의 잠금을 해제합니다.
EngUnlockSurface EngUnlockSurface 함수는 GDI가 표면의 잠금을 해제하도록 합니다.
EngUnmapEvent EngUnmapEvent 함수는 매핑된 사용자 모드 이벤트에 할당된 커널 모드 리소스를 정리합니다.

EngUnmapFile EngUnmapFile 함수는 시스템 공간에서 파일 뷰의 매핑을 해제합니다.
EngUnmapFontFile EngUnmapFontFile 함수는 사용되지 않습니다. 대신 EngUnmapFontFileFD를 사용합니다.
EngUnmapFontFileFD EngUnmapFontFileFD 함수는 시스템 메모리에서 지정된 글꼴 파일의 매핑을 해제합니다.
EngUnsecureMem EngUnsecureMem 함수는 메모리에 잠긴 주소 범위를 잠금 해제합니다.
EngWaitForSingleObject EngWaitForSingleObject 함수는 지정된 이벤트 개체가 신호된 상태로 설정되거나 대기 시간이 초과될 때까지 디스플레이 드라이버의 현재 스레드를 대기 상태로 전환합니다.
EngWideCharToMultiByte EngWideCharToMultiByte 함수는 지정된 코드 페이지를 사용하여 와이드 문자열을 ANSI 소스 문자열로 변환합니다.
EngWritePrinter EngWritePrinter 함수를 사용하면 프린터 그래픽 DLL이 데이터 스트림을 프린터 하드웨어로 보낼 수 있습니다.
FLOATOBJ_Add FLOATOBJ_Add 함수는 두 FLOATOBJ를 추가하고 첫 번째 매개 변수의 결과와 함께 를 반환합니다.
FLOATOBJ_AddFloat FLOATOBJ_AddFloat 함수는 FLOATL 형식의 값을 FLOATOBJ에 추가하고 첫 번째 매개 변수의 결과와 함께 를 반환합니다.
FLOATOBJ_AddLong FLOATOBJ_AddLong 함수는 FLOATOBJ에 LONG 형식의 값을 추가하고 첫 번째 매개 변수의 결과와 함께 를 반환합니다.
FLOATOBJ_Div FLOATOBJ_Div 함수는 두 FLOATOBJ를 나누고 첫 번째 매개 변수의 결과와 함께 를 반환합니다.

FLOATOBJ_DivFloat FLOATOBJ_DivFloat 함수는 FLOATOBJ를 FLOATL 형식의 값으로 나누고 첫 번째 매개 변수의 결과와 함께 를 반환합니다.
FLOATOBJ_DivLong FLOATOBJ_DivLong 함수는 FLOATOBJ를 LONG 형식의 값으로 나누고 첫 번째 매개 변수의 결과와 함께 를 반환합니다.
FLOATOBJ_Equal FLOATOBJ_Equal 함수는 두 FLOATOBJ가 같은지 여부를 결정합니다.
FLOATOBJ_EqualLong FLOATOBJ_EqualLong 함수는 FLOATOBJ와 LONG 형식의 값이 같은지 여부를 결정합니다.
FLOATOBJ_GetFloat FLOATOBJ_GetFloat 함수는 지정된 FLOATOBJ의 FLOAT 등가 값을 계산하고 반환합니다.
FLOATOBJ_GetLong FLOATOBJ_GetLong 함수는 지정된 FLOATOBJ의 LONG과 동등한 값을 계산하고 반환합니다.
FLOATOBJ_GreaterThan FLOATOBJ_GreaterThan 함수는 첫 번째 FLOATOBJ가 두 번째 FLOATOBJ보다 큰지 여부를 결정합니다.
FLOATOBJ_GreaterThanLong FLOATOBJ_GreaterThanLong 함수는 FLOATOBJ가 LONG 형식의 값보다 큰지 여부를 결정합니다.
FLOATOBJ_LessThan FLOATOBJ_LessThan 함수는 첫 번째 FLOATOBJ가 두 번째 FLOATOBJ보다 작은지 여부를 결정합니다.
FLOATOBJ_LessThanLong FLOATOBJ_LessThanLong 함수는 FLOATOBJ가 LONG 형식의 값보다 작은지 여부를 결정합니다.
FLOATOBJ_Mul FLOATOBJ_Mul 함수는 두 FLOATOBJ를 곱하고 첫 번째 매개 변수의 결과와 함께 를 반환합니다.

FLOATOBJ_MulFloat FLOATOBJ_MulFloat 함수는 FLOATOBJ를 FLOATL 형식의 값과 곱하고 첫 번째 매개 변수의 결과와 함께 를 반환합니다.
FLOATOBJ_MulLong FLOATOBJ_MulLong 함수는 FLOATOBJ를 LONG 형식의 값과 곱하고 첫 번째 매개 변수의 결과와 함께 를 반환합니다.
FLOATOBJ_Neg FLOATOBJ_Neg 함수는 FLOATOBJ를 부정합니다.
FLOATOBJ_SetFloat FLOATOBJ_SetFloat 함수는 FLOATL 형식의 값을 FLOATOBJ에 할당합니다.
FLOATOBJ_SetLong FLOATOBJ_SetLong 함수는 FLOATOBJ에 LONG 형식의 값을 할당합니다.
FLOATOBJ_Sub FLOATOBJ_Sub 함수는 첫 번째 FLOATOBJ에서 두 번째 FLOATOBJ를 빼고 첫 번째 매개 변수의 결과와 함께 를 반환합니다.
FLOATOBJ_SubFloat FLOATOBJ_SubFloat 함수는 FLOATOBJ에서 FLOATL 형식의 값을 빼고 첫 번째 매개 변수의 결과와 함께 를 반환합니다.
FLOATOBJ_SubLong FLOATOBJ_SubLong 함수는 FLOATOBJ에서 LONG 형식의 값을 빼고 첫 번째 매개 변수의 결과와 함께 를 반환합니다.
FONTOBJ_cGetAllGlyphHandles FONTOBJ_cGetAllGlyphHandles 함수를 사용하면 디바이스 드라이버가 GDI 글꼴의 모든 문자 모양 핸들을 찾을 수 있습니다.
FONTOBJ_cGetGlyphs FONTOBJ_cGetGlyphs 함수는 문자 모양 핸들을 포인터로 변환하여 문자 모양 데이터를 변환하는 글꼴 소비자에 대한 서비스로, 다음 FONTOBJ_cGetGlyphs 호출될 때까지 유효합니다.
FONTOBJ_pfdg

FONTOBJ_pfdg 함수는 지정된 글꼴과 연결된 FD_GLYPHSET 구조체에 대한 포인터를 검색합니다.
FONTOBJ_pifi FONTOBJ_pifi 함수는 지정된 글꼴과 연결된 IFIMETRICS 구조체에 대한 포인터를 검색합니다.
FONTOBJ_pjOpenTypeTablePointer FONTOBJ_pjOpenTypeTablePointer 함수는 OpenType 테이블의 뷰에 대한 포인터를 반환합니다.
FONTOBJ_pQueryGlyphAttrs FONTOBJ_pQueryGlyphAttrs 함수는 글꼴의 문자 모양에 대한 정보를 반환합니다.
FONTOBJ_pvTrueTypeFontFile FONTOBJ_pvTrueTypeFontFile 함수는 TrueType, OpenType 또는 Type1 글꼴 파일의 보기에 대한 사용자 모드 포인터를 검색합니다.
FONTOBJ_pwszFontFilePaths FONTOBJ_pwszFontFilePaths 함수는 글꼴과 연결된 파일 경로를 검색합니다.
FONTOBJ_pxoGetXform FONTOBJ_pxoGetXform 함수는 지정된 글꼴에 대한 개념적-디바이스 변환을 검색합니다.
FONTOBJ_vGetInfo FONTOBJ_vGetInfo 함수는 연결된 글꼴에 대한 정보를 검색합니다.
HT_ComputeRGBGammaTable HT_ComputeRGBGammaTable 함수는 GDI가 감마 번호에 따라 디바이스 빨간색, 녹색 및 파란색 강도를 계산하도록 합니다.
HT_Get8BPPFormatPalette HT_Get8BPPFormatPalette 함수는 픽셀 디바이스 유형당 표준 8비트에서 사용할 하프톤 팔레트를 반환합니다.
HT_Get8BPPMaskPalette HT_Get8BPPMaskPalette 함수는 픽셀당 8비트 디바이스 유형에 대한 마스크 팔레트를 반환합니다.
HTUI_DeviceColorAdjustment 그래픽 디바이스 드라이버에서 HTUI_DeviceColorAdjustment 함수를 사용하여 사용자가 디바이스의 하프톤 속성을 조정할 수 있는 대화 상자를 표시할 수 있습니다.

IsEqualGUID IsEqualGUID 매크로(winddi.h)는 두 GUID가 같은지 여부를 결정합니다.
PALOBJ_cGetColors PALOBJ_cGetColors 함수는 인덱싱된 색상표에서 RGB 색을 복사합니다.
PATHOBJ_bCloseFigure PATHOBJ_bCloseFigure 함수는 현재 위치에서 그림의 첫 번째 지점까지 선을 그려 경로의 열린 그림을 닫습니다.
PATHOBJ_bEnum PATHOBJ_bEnum 함수는 지정된 경로에서 다음 PATHDATA 레코드를 검색하고 경로의 곡선을 열거합니다.
PATHOBJ_bEnumClipLines PATHOBJ_bEnumClipLines 함수는 지정된 경로에서 잘린 선 세그먼트를 열거합니다.
PATHOBJ_bMoveTo PATHOBJ_bMoveTo 함수는 지정된 경로에서 현재 위치를 설정합니다.
PATHOBJ_bPolyBezierTo PATHOBJ_bPolyBezierTo 함수는 경로에 베지어 곡선을 그립니다.
PATHOBJ_bPolyLineTo PATHOBJ_bPolyLineTo 함수는 경로의 현재 위치에서 지정된 점을 통해 선을 그립니다.
PATHOBJ_vEnumStart PATHOBJ_vEnumStart 함수는 지정된 PATHOBJ 구조에 드라이버가 PATHOBJ_bEnum 호출하여 경로의 선 및/또는 곡선을 열거하도록 알립니다.
PATHOBJ_vEnumStartClipLines PATHOBJ_vEnumStartClipLines 함수를 사용하면 드라이버가 지정된 클립 영역에 대해 줄을 잘리도록 요청할 수 있습니다.
PATHOBJ_vGetBounds PATHOBJ_vGetBounds 함수는 지정된 경로에 대한 경계 사각형을 검색합니다.

STROBJ_bEnum STROBJ_bEnum 함수는 문자 모양 ID 및 위치를 열거합니다.
STROBJ_bEnumPositionsOnly STROBJ_bEnumPositionsOnly 함수는 지정된 텍스트 문자열의 문자 모양 ID와 위치를 열거하지만 캐시된 문자 모양 비트맵을 만들지는 않습니다.
STROBJ_bGetAdvanceWidths STROBJ_bGetAdvanceWidths 함수는 지정된 문자열을 구성하는 문자 모양 너비를 지정하는 벡터 배열을 검색합니다.
STROBJ_dwGetCodePage STROBJ_dwGetCodePage 함수는 지정된 STROBJ 구조체와 연결된 코드 페이지를 반환합니다.
STROBJ_fxBreakExtra STROBJ_fxBreakExtra 함수는 양쪽 맞춤 텍스트를 표시 및/또는 인쇄할 때 문자열의 각 공백 문자에 추가할 여분 공간을 검색합니다.
STROBJ_fxCharacterExtra STROBJ_fxCharacterExtra 함수는 표시 및/또는 인쇄할 때 문자열에서 각 문자의 너비를 보강할 추가 공간의 양을 검색합니다.
STROBJ_vEnumStart STROBJ_vEnumStart 함수는 STROBJ_bEnum 대한 후속 호출에서 GDI에서 반환되는 데이터의 폼 또는 형식을 정의합니다.
WNDOBJ_bEnum WNDOBJ_bEnum 함수는 창의 표시되는 영역에서 사각형 일괄 처리를 가져옵니다.
WNDOBJ_cEnumStart WNDOBJ_cEnumStart 함수는 창의 표시되는 영역에서 사각형 열거를 위한 매개 변수를 설정하는 콜백 함수입니다.
WNDOBJ_vSetConsumer WNDOBJ_vSetConsumer 함수는 지정된 WNDOBJ 구조체의 pvConsumer 필드에 드라이버 정의 값을 설정합니다.
XFORMOBJ_bApplyXform

XFORMOBJ_bApplyXform 함수는 지정된 변환 또는 해당 역을 지정된 점 배열에 적용합니다.
XFORMOBJ_iGetFloatObjXform XFORMOBJ_iGetFloatObjXform 함수는 FLOATOBJ 변환을 드라이버에 다운로드합니다.
XFORMOBJ_iGetXform XFORMOBJ_iGetXform 함수는 드라이버에 변환을 다운로드합니다.
XLATEOBJ_cGetPalette XLATEOBJ_cGetPalette 함수는 지정된 색상표에서 RGB 색 또는 비트 필드 형식을 검색합니다.
XLATEOBJ_hGetColorTransform XLATEOBJ_hGetColorTransform 함수는 지정된 변환 개체의 색 변환을 반환합니다.
XLATEOBJ_iXlate XLATEOBJ_iXlate 함수는 원본 팔레트의 색 인덱스를 대상 팔레트에서 가장 가까운 인덱스로 변환합니다.
XLATEOBJ_piVector XLATEOBJ_piVector 함수는 드라이버가 원본 인덱스를 대상 인덱스로 변환하는 데 사용할 수 있는 변환 벡터를 검색합니다.

콜백 함수

PFN_DrvQueryGlyphAttrs DrvQueryGlyphAttrs 함수는 글꼴의 문자 모양에 대한 정보를 반환합니다.
WNDOBJCHANGEPROC **WNDOBJCHANGEPROC** 함수는 GDI가 드라이버에 해당 창의 변경 내용을 알리는 데 사용하는 드라이버 정의 콜백 함수입니다.

구조체

BLENDOBJ BLENDOBJ 구조체는 원본 및 대상 비트맵에 대한 혼합 함수를 지정하여 혼합을 제어합니다.
BRUSHOBJ BRUSHOBJ 구조체에는 브러시 개체를 설명하는 세 개의 공용 멤버가 포함되어 있습니다.
CIECHROMA CIECHROMA 구조는 CIE 색 공간에서 색도 좌표, x 및 y 및 광도 Y를 설명하는 데 사용됩니다.
클립라인 CLIPLINE 구조체를 사용하면 드라이버가 그리기에 사용되는 두 클립 영역 사이의 선 부분에 액세스할 수 있습니다.
CLIPOBJ CLIPOBJ 구조체는 그릴 때 사용되는 클립 영역을 설명합니다.
COLORINFO COLORINFO 구조체는 CIE 좌표 공간에서 디바이스의 색을 정의합니다.
DEVHTADJDATA DEVHTADJDATA 구조체는 HTUI_DeviceColorAdjustment 함수에 대한 입력으로 사용됩니다.
DEVHTINFO DEVHTINFO 구조체는 HTUI_DeviceColorAdjustment 함수에 대한 입력으로 사용됩니다.
DEVINFO DEVINFO 구조체는 드라이버 및 해당 프라이빗 PDEV에 대한 정보를 그래픽 엔진에 제공합니다.
DRIVEROBJ DRIVEROBJ 구조는 GDI 서비스를 사용해야 하는 드라이버에서 할당한 리소스를 추적하는 데 사용됩니다.
DRVENABLEDATA DRVENABLEDATA 구조체에는 DRVFN 구조체 배열과 NT 기반 운영 체제의 그래픽 DDI 버전 번호에 대한 포인터가 포함되어 있습니다.
DRVFN

DRVFN 구조체는 그래픽 드라이버에서 정의한 그래픽 DDI 함수에 대한 포인터를 GDI에 제공하는 데 사용됩니다.

ENG_TIME_FIELDS

ENG_TIME_FIELDS 구조체는 EngQueryLocalTime 함수에서 현지 시간을 반환하는 데 사용됩니다.

ENGSAFESEMAPHORE

ENGSAFESEMAPHORE 구조체는 드라이버에 스레드로부터 안전한 세마포를 제공합니다.

열거형

ENUMRECTS 구조체는 CLIPOBJ_cEnumStart 함수에서 CLIPOBJ_bEnum 함수에 대한 클립 영역의 사각형에 대한 정보를 제공하는 데 사용됩니다.

FD_DEVICEMETRICS

FD_DEVICEMETRICS 구조체는 드라이버 제공 DrvQueryFontData 함수의 iMode 매개 변수가 QFD_MAXEXTENTS 경우 GDI에 디바이스별 글꼴 정보를 제공하는 데 사용됩니다.

FD_GLYPHATTR

FD_GLYPHATTR 구조체는 FONTOBJ_pQueryGlyphAttrs 및 DrvQueryGlyphAttrs 함수의 반환 값을 지정하는 데 사용됩니다.

FD_GLYPHSET

FD_GLYPHSET 구조체는 유니코드 문자에서 문자 모양 핸들로의 매핑을 정의하는 데 사용됩니다.

FD_KERNINGPAIR

FD_KERNINGPAIR 구조체는 커닝 쌍에 대한 정보를 저장하는 데 사용됩니다.

FD_XFORM

FD_XFORM 구조체는 임의의 2차원 글꼴 변환을 설명합니다.

FLOATOBJ

FLOATOBJ 구조체는 부동 소수점 숫자를 에뮬레이트하는 데 사용됩니다.

FLOATOBJ_XFORM

FLOATOBJ_XFORM 구조체는 기하학적 와이드 라인과 같은 임의의 선형 2차원 변환을 설명합니다. (FLOATOBJ_XFORM)

FONTDIFF

FONTDIFF 구조체는 기본 글꼴과 시뮬레이션 중 하나 간에 다른 모든 특성을 설명합니다.
Fontinfo FONTINFO 구조체에는 특정 글꼴에 대한 정보가 포함됩니다.
FONTOBJ FONTOBJ 구조체는 글꼴의 특정 instance 대한 정보에 대한 액세스 권한을 드라이버에 부여하는 데 사용됩니다.
FONTSIM FONTSIM 구조체에는 굵게, 기울임꼴 및 굵은 기울임꼴 시뮬레이션을 설명하는 하나 이상의 FONTDIFF 구조에 대한 오프셋이 포함되어 있습니다.
GAMMARAMP GAMMARAMP 구조체는 DrvIcmSetDeviceGammaRamp에서 특정 디스플레이 디바이스의 하드웨어 감마 램프를 설정하는 데 사용됩니다.
GDIINFO GDIINFO 구조는 지정된 디바이스의 그래픽 기능을 설명합니다.
문자 모양 GLYPHBITS 구조체는 문자 모양 비트맵을 정의하는 데 사용됩니다.
GLYPHDATA GLYPHDATA 구조체에는 개별 문자 모양에 대한 정보가 포함됩니다.
GLYPHDEF GLYPHDEF 공용 구조체는 개별 문자 모양을 식별하고 GLYPHBITS 구조체에 대한 포인터 또는 PATHOBJ 구조체에 대한 포인터를 제공합니다.
문자 모양 GLYPHPOS 구조체는 GDI에서 문자 모양 설명 및 위치를 그래픽 드라이버에 제공하는 데 사용됩니다.
IFIEXTRA IFIEXTRA 구조체는 GDI에서 사용할 수 있는 지정된 서체에 대한 추가 정보를 정의합니다.
IFIMETRICS

IFIMETRICS 구조체는 GDI에서 사용할 수 있는 지정된 서체에 대한 정보를 정의합니다.
LINEATTRS LINEATTRS 구조체는 드라이버의 선 그리기 함수에서 선 특성을 확인하는 데 사용됩니다.
PALOBJ PALOBJ 구조체는 인덱싱된 색상표를 나타내는 사용자 개체입니다.
PATHDATA PATHDATA 구조체는 하위 경로의 전체 또는 일부를 설명합니다.
PATHOBJ PATHOBJ 구조체는 스트로크하거나 채울 선 및 베지어 곡선 집합을 설명하는 데 사용됩니다.
PERBANDINFO PERBANDINFO 구조체는 프린터 그래픽 DLL의 DrvQueryPerBandInfo 함수에 대한 입력으로 사용됩니다.
실행 RUN 구조체는 CLIPLINE 구조체에 의해 잘리지 않는 선형 픽셀 집합을 설명하는 데 사용됩니다.
STROBJ STROBJ 클래스 또는 텍스트 문자열 개체에는 디바이스 드라이버에 대한 문자 모양 핸들 및 위치의 열거형이 포함되어 있습니다.
SURFOBJ SURFOBJ 구조체는 표면의 사용자 개체입니다. 디바이스 드라이버는 일반적으로 Surface 개체가 GDI 비트맵 또는 디바이스 관리 표면을 나타내는 경우에만 Surface 개체에서 메서드를 호출합니다.
TYPE1_FONT TYPE1_FONT 구조에는 PostScript 드라이버가 GDI를 통해 Type1 글꼴에 액세스하는 데 필요한 정보가 포함되어 있습니다.
WCRUN WCRUN 구조체는 유니코드 문자의 실행을 설명합니다.
WNDOBJ

WNDOBJ 구조를 사용하면 드라이버가 창의 위치, 크기 및 표시되는 클라이언트 영역 변경 내용을 추적할 수 있습니다.
XFORML FLOATOBJ_XFORM 구조체는 기하학적 와이드 라인과 같은 임의의 선형 2차원 변환을 설명합니다. F
XFORMOBJ XFORMOBJ 구조체는 기하학적 와이드 라인과 같은 임의의 선형 2차원 변환을 설명합니다.
XLATEOBJ XLATEOBJ 구조체는 색상표에서 다른 색상표로 색 인덱스를 변환하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

windef.h 헤더

아티클2023. 08. 27.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 디바이스 참조 표시
- 높은 DPI
- Windows 접근성 기능
- Windows 및 메시지
- Windows GDI

windef.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

지점 POINT 구조체는 점의 x 및 y 좌표를 정의합니다.
POINTL POINTL 구조체는 점의 x 및 y 좌표를 정의합니다.
POINTS POINTS 구조체는 점의 x 및 y 좌표를 정의합니다.
RECT RECT 구조체는 왼쪽 위와 오른쪽 아래 모서리의 좌표로 사각형을 정의합니다.
RECTL RECTL 구조체는 왼쪽 위와 오른쪽 아래 모서리의 좌표로 사각형을 정의합니다.
SIZE SIZE 구조체는 사각형의 너비와 높이를 정의합니다.

열거형

DPI_AWARENESS 스레드, 프로세스 또는 창에 대한 dpi(인치당 점 수) 설정을 식별합니다.
DPI_HOSTING_BEHAVIOR 창에 대한 DPI 호스팅 동작을 식별합니다. 이 동작을 사용하면 스레드에서 만든 창이 다른 DPI_AWARENESS_CONTEXT 있는 자식 창을 호스트할 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

windns.h 헤더

아티클2025. 01. 09.

이 헤더는 DNS(Domain Name System)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DNS(도메인 이름 시스템)

windns.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

테이블 확장

DnsAcquireContextHandle_A DnsAcquireContextHandle 함수 형식은 자격 증명 집합에 대한 컨텍스트 핸들을 획득합니다. (ANSI)
DnsAcquireContextHandle_W DnsAcquireContextHandle 함수 형식은 자격 증명 집합에 대한 컨텍스트 핸들을 획득합니다. (유니코드)
dnsCancelQuery DnsCancelQuery 함수를 사용하여 DNS 네임스페이스에 대한 보류 중인 쿼리를 취소할 수 있습니다.
dnsCancelQueryRaw DnsQueryRaw를 호출하여 시작된 쿼리를 취소합니다.
DnsExtractRecordsFromMessage_UTF8 DnsExtractRecordsFromMessage 함수 형식은 DNS 메시지에서 RR(리소스 레코드)을 추출하고 해당 레코드를 DNS_RECORD 구조에 저장합니다. (DnsExtractRecordsFromMessage_UTF8)
DnsExtractRecordsFromMessage_W DnsExtractRecordsFromMessage 함수 형식은 DNS 메시지에서 RR(리소스 레코드)을 추출하고 해당 레코드를 DNS_RECORD 구조에 저장합니다. (DnsExtractRecordsFromMessage_W)
dnsFree

DnsQuery 함수를 사용하여 가져온 DNS 레코드에 할당된 메모리를 해제합니다.
dnsFreeCustomServers DnsGetApplicationSettings이전 호출에서 반환된 사용자 지정 서버의 배열을 해제합니다.
dnsFreeProxyName DnsGetProxyInformation 함수를 사용하여 가져온 DNS_PROXY_INFORMATION 구조체의 proxyName 멤버에 할당된 메모리를 해제합니다.
dnsGetApplicationSettings 애플리케이션별 DNS 설정을 검색합니다.
dnsGetProxyInformation DnsGetProxyInformation 함수는 DNS 서버의 이름 확인 정책 테이블에 대한 프록시 정보를 반환합니다.
DnsModifyRecordsInSet_A 함수는 이전에 DNS 서버에 등록되었을 수 있는 RR(리소스 레코드) 집합을 추가, 수정 또는 제거합니다. (DnsModifyRecordsInSet_A)
DnsModifyRecordsInSet_UTF8 함수는 이전에 DNS 서버에 등록되었을 수 있는 RR(리소스 레코드) 집합을 추가, 수정 또는 제거합니다. (DnsModifyRecordsInSet_UTF8)
DnsModifyRecordsInSet_W 함수는 이전에 DNS 서버에 등록되었을 수 있는 RR(리소스 레코드) 집합을 추가, 수정 또는 제거합니다. (DnsModifyRecordsInSet_W)
dnsNameCompare DnsNameCompare 함수는 두 개의 DNS 이름을 비교합니다. (DnsNameCompare_UTF8)
DnsNameCompare_A DnsNameCompare 함수는 두 개의 DNS 이름을 비교합니다. (DnsNameCompare_A)
DnsNameCompare_W DnsNameCompare 함수는 두 개의 DNS 이름을 비교합니다. (DnsNameCompare_W)
DnsQuery_A

DNS 네임스페이스에 대한 일반 쿼리 인터페이스이며 애플리케이션 개발자에게 DNS 쿼리 확인 인터페이스를 제공합니다. (DnsQuery_A)
DnsQuery_UTF8 DNS 네임스페이스에 대한 일반 쿼리 인터페이스이며 애플리케이션 개발자에게 DNS 쿼리 확인 인터페이스를 제공합니다. (DnsQuery_UTF8)
DnsQuery_W DNS 네임스페이스에 대한 일반 쿼리 인터페이스이며 애플리케이션 개발자에게 DNS 쿼리 확인 인터페이스를 제공합니다. (DnsQuery_W)
dnsQueryConfig DnsQueryConfig 함수를 사용하면 애플리케이션 프로그래머가 로컬 컴퓨터 또는 특정 어댑터의 구성을 쿼리할 수 있습니다.
dnsQueryEx DNS 네임스페이스에 대한 비동기 제네릭 쿼리 인터페이스이며 애플리케이션 개발자에게 DNS 쿼리 확인 인터페이스를 제공합니다.
dnsQueryRaw DNS 쿼리를 포함하는 원시 패킷이나 쿼리 이름 및 형식을 허용하는 DNS 쿼리를 수행할 수 있습니다.
dnsQueryRawResultFree DNS_QUERY_RAW_RESULT 구조체 개체에 할당된 메모리를 해제합니다.
dnsRecordCompare DnsRecordCompare 함수는 두 개의 RR(DNS 리소스 레코드)을 비교합니다.
dnsRecordCopyEx DnsRecordCopyEx 함수는 지정된 RR(리소스 레코드)의 복사본을 만듭니다. DnsRecordCopyEx 함수는 복사 작업 중에 문자 인코딩을 변환할 수도 있습니다.
DnsRecordListFree DnsQuery 함수를 사용하여 가져온 DNS 레코드에 할당된 메모리를 해제합니다.
dnsRecordSetCompare DnsRecordSetCompare 함수는 두 개의 RR 집합을 비교합니다.

dnsRecordSetCopyEx DnsRecordSetCopyEx 함수는 지정된 리소스 레코드 집합의 복사본을 만듭니다. DnsRecordSetCopyEx 함수는 복사 작업 중에 문자 인코딩을 변환할 수도 있습니다.
dnsRecordSetDetach DnsRecordSetDetach 함수는 지정된 DNS 레코드 목록에서 첫 번째 레코드 집합을 분리합니다.
dnsReleaseContextHandle DnsReleaseContextHandle 함수는 특정 계정의 자격 증명을 저장하는 데 사용되는 메모리를 해제합니다.
dnsReplaceRecordSetA 기존 RR(리소스 레코드) 집합을 대체합니다. (DnsReplaceRecordSetA)
dnsReplaceRecordSetUTF8 기존 RR(리소스 레코드) 집합을 대체합니다. (DnsReplaceRecordSetUTF8)
dnsReplaceRecordSetW 기존 RR(리소스 레코드) 집합을 대체합니다. (DnsReplaceRecordSetW)
dnsServiceBrowse 로컬 네트워크에서 실행되는 서비스에 대한 DNS-SD 검색을 시작하는 데 사용됩니다.
dnsServiceBrowseCancel 실행 중인 DNS-SD 검색 쿼리를 취소하는 데 사용됩니다.
dnsServiceConstructInstance 이를 설명하는 데이터에서 DNS_SERVICE_INSTANCE 구조를 빌드하는 데 사용됩니다.
dnsServiceCopyInstance DNS_SERVICE_INSTANCE 구조를 복사하는 데 사용됩니다.
dnsServiceDeRegister 등록된 서비스를 제거하는 데 사용됩니다.
dnsServiceFreeInstance DNS_SERVICE_INSTANCE 구조와 연결된 리소스를 해제하는 데 사용됩니다.

dnsServiceRegister 이 디바이스에서 검색 가능한 서비스를 등록하는 데 사용됩니다. (DnsServiceRegister)
dnsServiceRegisterCancel 보류 중인 등록 작업을 취소하는 데 사용됩니다.
dnsServiceResolve 로컬 네트워크에 보급된 서비스에 대한 자세한 정보를 가져오는 데 사용됩니다.
dnsServiceResolveCancel 실행 중인 DNS-SD 쿼리를 취소하는 데 사용됩니다.
dnsSetApplicationSettings 애플리케이션별 DNS 설정을 구성합니다. 여기에는 애플리케이션별 DNS 서버를 시스템 구성 서버에 대한 대체(fallback)로 설정하거나 단독으로 설정하는 기능이 포함됩니다.
dnsStartMulticastQuery 이 디바이스에서 검색 가능한 서비스를 등록하는 데 사용됩니다. (DnsStartMulticastQuery)
dnsStopMulticastQuery DnsStartMulticastQuery 작업을 실행을 중지하는 데 사용됩니다.
dnsValidateName DnsValidateName 매크로(windns.h) 함수는 지정된 DNS 이름의 상태를 확인합니다.
DnsValidateName_A DnsValidateName 함수는 지정된 DNS 이름의 상태를 확인합니다. (DnsValidateName_A)
DnsValidateName_UTF8 DnsValidateName_UTF8 함수(windns.h) 함수는 지정된 DNS 이름의 상태를 확인합니다.
DnsValidateName_W DnsValidateName 함수는 지정된 DNS 이름의 상태를 확인합니다. (DnsValidateName_W)
dnsValidateServerStatus DnsValidateServerStatus 함수는 IP 주소를 적합한 DNS 서버로 유효성을 검사합니다.

DnsWriteQuestionToBuffer_UTF8 DnsWriteQuestionToBuffer 함수 형식은 DNS 쿼리 메시지를 만들고 DNS_MESSAGE_BUFFER 구조에 저장합니다. (DnsWriteQuestionToBuffer_UTF8)
DnsWriteQuestionToBuffer_W DnsWriteQuestionToBuffer 함수 형식은 DNS 쿼리 메시지를 만들고 DNS_MESSAGE_BUFFER 구조에 저장합니다. (DnsWriteQuestionToBuffer_W)

콜백 함수

테이블 확장

DNS_QUERY_COMPLETION_ROUTINE DNS_QUERY_COMPLETION_ROUTINE 콜백은 DNS 쿼리의 결과를 비동기적으로 반환하는 데 사용됩니다.
DNS_QUERY_RAW_COMPLETION_ROUTINE 구현하는 비동기 콜백 함수의 함수 서명입니다. 시스템은 DnsQueryRaw를 호출하여 시작한 쿼리 결과를 사용하여 구현을 호출합니다.
DNS_SERVICE_BROWSE_CALLBACK DNS-SD 쿼리의 결과를 비동기적으로 반환하는 데 사용됩니다.
DNS_SERVICE_REGISTER_COMPLETE 서비스 등록이 완료되었음을 애플리케이션에 알리는 데 사용됩니다.
DNS_SERVICE_RESOLVE_COMPLETE 서비스 확인 작업의 결과를 비동기적으로 반환하는 데 사용됩니다.
MDNS_QUERY_CALLBACK mDNS 쿼리의 결과를 비동기적으로 반환하는 데 사용됩니다.

구조

테이블 확장

DNS_APPLICATION_SETTINGS 애플리케이션별 DNS 설정을 나타냅니다.
DNS_PROXY_INFORMATION DNS 서버의 이름 확인 정책 테이블에 대한 프록시 정보를 포함합니다.
DNS_QUERY_CANCEL DNS_QUERY_CANCEL 구조를 사용하여 비동기 DNS 쿼리를 취소할 수 있습니다.
DNS_QUERY_RAW_CANCEL DNS 원시 쿼리 취소 핸들을 나타냅니다.
DNS_QUERY_RAW_REQUEST DNS 원시 쿼리 요청을 나타냅니다(dnsQueryRaw참조).
DNS_QUERY_RAW_RESULT DNS 원시 쿼리 결과를 나타냅니다(DNS_QUERY_RAW_COMPLETION_ROUTINE참조).
DNS_QUERY_REQUEST DNS_QUERY_REQUEST 구조에는 DnsQueryEx 호출에 사용되는 DNS 쿼리 매개 변수가 포함됩니다.
DNS_QUERY_REQUEST3 DnsQueryEx호출에 사용되는 DNS 쿼리 매개 변수를 포함합니다.
DNS_QUERY_RESULT DNS_QUERY_RESULT 구조에는 DnsQueryEx 호출에서 반환된 DNS 쿼리 결과가 포함됩니다.
DNS_RRSET DNS_RRSET 구조에는 RR(DNS 리소스 레코드) 집합에 대한 정보가 포함됩니다.
DNS_SERVICE_BROWSE_REQUEST DnsServiceBrowse호출에 사용되는 쿼리 매개 변수를 포함합니다.
DNS_SERVICE_CANCEL 비동기 DNS-SD 작업을 취소하는 데 사용됩니다.

DNS_SERVICE_INSTANCE 네트워크에서 실행되는 DNS 서비스를 나타냅니다.
DNS_SERVICE_REGISTER_REQUEST DnsServiceRegister사용하여 서비스를 보급하거나 DnsServiceDeRegister사용하여 광고를 중지하는 데 필요한 정보를 포함합니다.
DNS_SERVICE_RESOLVE_REQUEST DnsServiceResolve호출에 사용되는 쿼리 매개 변수를 포함합니다.
DNS_WIRE_QUESTION DNS_WIRE_QUESTION 구조에는 RFC 1035의 섹션 4.1.2에 지정된 대로 네트워크를 통해 전송되는 DNS 질문에 대한 정보가 포함됩니다.
DNS_WIRE_RECORD DNS_WIRE_RECORD 구조에는 RFC 1035의 섹션 4.1.3에 지정된 대로 네트워크를 통해 전송되는 DNS 와이어 레코드에 대한 정보가 포함됩니다.
IP4_ARRAY IP4_ARRAY 구조체는 IPv4 주소 배열을 저장합니다.
MDNS_QUERY_HANDLE 진행 중인 MDNS 쿼리와 관련된 정보를 포함합니다. 애플리케이션에서 해당 콘텐츠를 수정해서는 안 됩니다.
MDNS_QUERY_REQUEST mDNS 쿼리를 수행하는 데 필요한 정보를 포함합니다.

열거형

테이블 확장

DNS_CONFIG_TYPE DNS_CONFIG_TYPE 열거형은 DNS 구성 형식 정보를 제공합니다.

DNS_FREE_TYPE DNS_FREE_TYPE 열거형은 해제할 데이터 형식을 지정합니다.
DNS_NAME_FORMAT DNS_NAME_FORMAT 열거형은 DNS의 이름 형식 정보를 지정합니다.
DNS_PROXY_INFORMATION_TYPE DNS_PROXY_INFORMATION_TYPE 열거형은 DNS_PROXY_INFORMATION 구조체의 프록시 정보 형식을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

windnsdef.h 헤더

이 헤더는 DNS(Domain Name System)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- DNS(도메인 이름 시스템)

windnsdef.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조

테이블 확장

DNS_A_DATA DNS_A_DATA 구조체는 RFC 1035의 섹션 3.4.1에 지정된 DNS 주소(A) 레코드를 나타냅니다.
DNS_AAAA_DATA DNS_AAAA_DATA 구조체는 RFC 3596에 지정된 대로 AAAA(DNS IPv6) 레코드를 나타냅니다.
DNS_ADDR DNS_ADDR 구조체는 IPv4 또는 IPv6 주소를 저장합니다.
DNS_ADDR_ARRAY IPv4 또는 IPv6 주소의 배열을 저장합니다.
DNS_ATMA_DATA DNS_ATMA_DATA 구조체는 ATMA(DNS ATM 주소) RR(리소스 레코드)을 나타냅니다.
DNS_CUSTOM_SERVER DNS 사용자 지정 서버를 나타냅니다. DNS_CUSTOM_SERVER 개체는 DNS_QUERY_REQUEST3 구조를 통해 dnsQueryEx 전달됩니다.
DNS_DHCID_DATA RFC 4701 섹션 3에 지정된 DNS DHCID(동적 호스트 구성 프로토콜 정보) RR(리소스 레코드)을 나타냅니다.
DNS_DS_DATA RFC 4034의 섹션 2에 지정된 DS RR(리소스 레코드)을 나타내며 DNS_DNSKEY_DATA 내용을 확인하는 데 사용됩니다.

DNS_HEADER DNS_HEADER 구조에는 RFC 1035의 섹션 4.1.1에 지정된 대로 DNS 메시지를 보낼 때 사용되는 DNS 헤더 정보가 포함됩니다.
DNS_KEY_DATA DNS_KEY_DATA 구조체는 RFC 3445에 지정된 DNS 키(KEY) RR(리소스 레코드)을 나타냅니다.
DNS_LOC_DATA DNS_LOC_DATA 구조체는 RFC 1876에 지정된 대로 RR(DNS 위치) 리소스 레코드를 나타냅니다.
DNS_MESSAGE_BUFFER DNS_MESSAGE_BUFFER 구조체는 DNS 쿼리에 대한 메시지 정보를 저장합니다.
DNS_MINFO_DATAA DNS_MINFO_DATA 구조는 RFC 1035의 섹션 3.3.7에 지정된 대로 MINFO(DNS 메일 정보) 레코드를 나타냅니다. (ANSI)
DNS_MINFO_DATAW DNS_MINFO_DATA 구조는 RFC 1035의 섹션 3.3.7에 지정된 대로 MINFO(DNS 메일 정보) 레코드를 나타냅니다. (유니코드)
DNS_MX_DATAA DNS_MX_DATA 구조는 RFC 1035의 섹션 3.3.9에 지정된 대로 DNS MX(메일 교환기) 레코드를 나타냅니다. (ANSI)
DNS_MX_DATAW DNS_MX_DATA 구조는 RFC 1035의 섹션 3.3.9에 지정된 대로 DNS MX(메일 교환기) 레코드를 나타냅니다. (유니코드)
DNS_NAPTR_DATAA DNS_NAPTR_DATA 구조체는 RFC 2915에 지정된 대로 NAPTR(명명 기관 포인터) RR(DNS 리소스 레코드)을 나타냅니다. (ANSI)
DNS_NAPTR_DATAW DNS_NAPTR_DATA 구조체는 RFC 2915에 지정된 대로 NAPTR(명명 기관 포인터) RR(DNS 리소스 레코드)을 나타냅니다. (유니코드)
DNS_NSEC_DATAA

RFC 4034 섹션 4에 지정된 NSEC RR(리소스 레코드)을 나타냅니다. (ANSI)
DNS_NSEC_DATAW RFC 4034 섹션 4에 지정된 NSEC RR(리소스 레코드)을 나타냅니다. (유니코드)
DNS_NULL_DATA DNS_NULL_DATA 구조체는 RFC 1035 섹션 3.3.10에 지정된 대로 DNS 리소스 레코드에 대한 NULL 데이터를 나타냅니다.
DNS_NXT_DATAA DNS_NXT_DATA 구조체는 RFC 2535 섹션 5에 지정된 대로 DNS 다음(NXT) RR(리소스 레코드)을 나타냅니다. (ANSI)
DNS_NXT_DATAW DNS_NXT_DATA 구조체는 RFC 2535 섹션 5에 지정된 대로 DNS 다음(NXT) RR(리소스 레코드)을 나타냅니다. (유니코드)
DNS_OPT_DATA RFC 2671의 섹션 4에 지정된 대로 DNS 옵션(OPT) RR(리소스 레코드)을 나타냅니다.
DNS_PTR_DATAA DNS_PTR_DATA 구조체는 RFC 1035의 섹션 3.3.12에 지정된 대로 PTR(DNS 포인터) 레코드를 나타냅니다. (ANSI)
DNS_PTR_DATAW DNS_PTR_DATA 구조체는 RFC 1035의 섹션 3.3.12에 지정된 대로 PTR(DNS 포인터) 레코드를 나타냅니다. (유니코드)
DNS_RECORD_FLAGS DNS_RECORD_FLAGS 구조체는 DNS_RECORD 구조체에서 사용할 플래그를 설정하는 데 사용됩니다.
DNS_RECORDA DNS RR(리소스 레코드)을 저장합니다. (ANSI)
DNS_RECORDW DNS RR(리소스 레코드)을 저장합니다. (유니코드)
DNS_SIG_DATAA

구조체는 RFC 4034에 지정된 대로 DNSSEC(DNS 보안 확장) SIG(암호화 서명) RR(리소스 레코드)을 나타냅니다. (ANSI)
DNS_SIG_DATAW 구조체는 RFC 4034에 지정된 대로 DNSSEC(DNS 보안 확장) SIG(암호화 서명) RR(리소스 레코드)을 나타냅니다. (유니코드)
DNS_SOA_DATAA DNS_SOA_DATA 구조는 RFC 1035의 섹션 3.3.13에 지정된 대로 SOA(DNS 권한 시작) 레코드를 나타냅니다. (ANSI)
DNS_SOA_DATAW DNS_SOA_DATA 구조는 RFC 1035의 섹션 3.3.13에 지정된 대로 SOA(DNS 권한 시작) 레코드를 나타냅니다. (유니코드)
DNS_SRV_DATAA DNS_SRV_DATA 구조체는 RFC 2782에 지정된 대로 SRV(DNS 서비스) 레코드를 나타냅니다. (ANSI)
DNS_SRV_DATAW DNS_SRV_DATA 구조체는 RFC 2782에 지정된 대로 SRV(DNS 서비스) 레코드를 나타냅니다. (유니코드)
DNS_SVCB_DATA DNS_SVCB_DATA 구조체는 RFC 9460에 지정된 대로 DNS SVCB("서비스 바인딩") 레코드를 나타냅니다.
DNS_SVCB_PARAM SVCB 매개 변수를 나타냅니다.
DNS_SVCB_PARAM_ALPN DNS_SVCB_PARAM_ALPN 구조체는 RFC 9460 섹션 7.1을 나타냅니다.
DNS_SVCB_PARAM_ALPN_ID 미정
DNS_SVCB_PARAM_IPV4 DNS_SVCB_PARAM_IPV4 구조체는 RFC 9460의 섹션 7.3을 나타냅니다.
DNS_SVCB_PARAM_IPV6 DNS_SVCB_PARAM_IPV6 구조체는 RFC 9460의 섹션 7.3을 나타냅니다.

DNS_SVCB_PARAM_MANDATORY DNS_SVCB_PARAM_MANDATORY 구조체는 RFC 9460의 섹션 8을 나타냅니다.
DNS_SVCB_PARAM_UNKNOWN 미정
DNS_TKEY_DATAA DNS_TKEY_DATA 구조는 RFC 2930에 지정된 대로 DNS 확인자와 서버 간에 알고리즘의 공유 비밀 키를 설정하고 삭제하는 데 사용되는 DNS TKEY 리소스 레코드를 나타냅니다. (ANSI)
DNS_TKEY_DATAW DNS_TKEY_DATA 구조는 RFC 2930에 지정된 대로 DNS 확인자와 서버 간에 알고리즘의 공유 비밀 키를 설정하고 삭제하는 데 사용되는 DNS TKEY 리소스 레코드를 나타냅니다. (유니코드)
DNS_TSIG_DATAA DNS_TSIG_DATA 구조는 RFC 2845 및 RFC 3645에 지정된 대로 TSIG(비밀 키 트랜잭션 인증) RR(리소스 레코드)을 나타냅니다. (ANSI)
DNS_TSIG_DATAW DNS_TSIG_DATA 구조는 RFC 2845 및 RFC 3645에 지정된 대로 TSIG(비밀 키 트랜잭션 인증) RR(리소스 레코드)을 나타냅니다. (유니코드)
DNS_TXT_DATAA DNS_TXT_DATA 구조체는 RFC 1035의 섹션 3.3.14에 지정된 대로 TXT(DNS 텍스트) 레코드를 나타냅니다. (ANSI)
DNS_TXT_DATAW DNS_TXT_DATA 구조체는 RFC 1035의 섹션 3.3.14에 지정된 대로 TXT(DNS 텍스트) 레코드를 나타냅니다. (유니코드)
DNS_WINS_DATA DNS_WINS_DATA 구조는 DNS WINS(Windows Internet Name Service) 레코드를 나타냅니다.
DNS_WINSR_DATAA DNS_WINSR_DATA 구조는 DNS WINDOWS WINDOWS 인터넷 이름 서비스 WINSR(역방향 조회) 레코드를 나타냅니다. (ANSI)
DNS_WINSR_DATAW

DNS_WINSR_DATA 구조는 DNS WINDOWS WINDOWS 인터넷 이름 서비스 WINSR(역방향 조회) 레코드를 나타냅니다. (유니코드)
DNS_WKS_DATA DNS_WKS_DATA 구조는 RFC 1035의 섹션 3.4.2에 지정된 대로 DNS WKS(잘 알려진 서비스) 레코드를 나타냅니다.
IP6_ADDRESS IP6_ADDRESS 구조체는 IPv6 주소를 저장합니다. (IP6_ADDRESS)

열거형

테이블 확장

DNS_CHARSET DNS_CHARSET 열거형은 사용된 문자 집합을 지정합니다.
DNS_SECTION DNS_SECTION 열거형은 레코드 플래그 및 DNS 와이어 메시지 헤더 섹션 개수에 대한 인덱스로 사용됩니다.
DNS_SVCB_PARAM_TYPE 미정

Last updated on 2025. 07. 09.

windows.data.pdf.interop.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

windows.data.pdf.interop.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

data::p df::interop::IPdfRendererNative PDF(이식 가능한 문서 형식) 파일의 단일 페이지를 표시하기 위한 고성능 API를 나타냅니다.

Functions

data::p df::interop::P dfCreateRenderer PDF(이식 가능한 문서 형식) 파일의 단일 페이지를 표시하기 위한 IPdfRendererNative 인터페이스의 instance 가져옵니다.
data::p df::interop::P dfRenderParams PDF_RENDER_PARAMS 구조를 채웁니다. PDF_RENDER_PARAMS 구조체는 PDF(이식 가능한 문서 형식) 파일의 단일 페이지를 출력하기 위한 속성 집합을 나타냅니다.

구조체

PDF_RENDER_PARAMS PDF(이식 가능한 문서 형식) 파일의 단일 페이지를 출력하기 위한 속성 집합을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

windows.devices.alljoyn.interop.h 헤더

아티클2023. 08. 27.

이 헤더는 AllJoyn API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- AllJoyn API

windows.devices.alljoyn.interop.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

devices::alljoyn::interop::IWindowsDevicesAllJoynBusAttachmentFactoryInterop 이 인터페이스를 사용하면 참조 소유권을 가져오지 않고 alljoyn_busattachment 만들 수 있습니다.
devices::alljoyn::interop::IWindowsDevicesAllJoynBusAttachmentInterop 이 인터페이스를 사용하면 기본 alljoyn_busattachment 개체를 검색할 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

windows.devices.display.core.interop.h 헤더

아티클2023. 08. 27.

Windows.Devices.Display.Core 네임스페이스와 데스크톱 interop용 API.

windows.devices.display.core.interop.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

devices::d isplay::core::interop::IDisplayDeviceInterop Windows 런타임 Windows.Devices.Display.Core 네임스페이스의 Direct3D(및 기타 그래픽 API)와 API 간에 상호 운용하는 메서드를 제공합니다.
devices::d isplay::core::interop::IDisplayPathInterop Windows 런타임 클래스 DisplayPath와 기타 디스플레이 관련 API 간에 상호 운용하는 메서드를 제공합니다.

Functions

devices::d isplay::core::interop::Int64FromLuid LUID에서 64비트 부제 정수로 변환합니다.
devices::d isplay::core::interop::LuidFromInt64 부 서명된 64비트 정수에서 LUID로 변환합니다.

구조체

Luid 어댑터의 로컬 식별자를 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

windows.foundation.h 헤더

아티클2023. 03. 08.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

windows.foundation.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

foundation::IAsyncAction 비동기 동작을 나타냅니다.
foundation::IClosable 할당된 리소스를 해제하는 메서드를 정의합니다.
foundation::IPropertyValue Windows 런타임 속성 저장소의 값을 나타냅니다.
foundation::IPropertyValueStatics 속성 저장소에 저장할 수 있는 IPropertyValue 개체를 만듭니다.
foundation::IStringable 현재 개체를 문자열로 나타내는 방법을 제공합니다.

구조체

DateTime 일반적으로 날짜와 시간으로 표시된 시간을 나타냅니다.
Point 2차원 평면의 한 점을 정의하는 부동 소수점 X 좌표와 Y 좌표의 정렬된 쌍을 나타냅니다.

Rect 사각형의 위치와 크기를 나타내는 4개의 부동 소수점 숫자 집합을 포함합니다.
크기 높이와 너비를 지정하는 정렬된 부동 소수점 숫자 쌍을 나타냅니다.
TimeSpan 시간 간격을 나타냅니다.

열거형

foundation::P ropertyType IPropertyValue 인터페이스를 구현하는 Windows 런타임 개체에 저장된 형식을 식별합니다.

windows.graphics.capture.interop.h 헤더

아티클2023. 08. 27.

Windows.Graphics.Capture 네임스페이스와 데스크톱 interop용 API.

windows.graphics.capture.interop.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

graphics::capture::interop::IGraphicsCaptureItemInterop GraphicsCaptureItem 정품 인증 팩터리에 대한 데스크톱 interop 인터페이스입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

windows.graphics.directx.direct3d11.interop.h 헤더

IDirect3DDevice 및 IDirect3DSurface와 함께 사용할 Interop 함수입니다.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

windows.graphics.directx.direct3d11.interop.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

graphics::d irectx::d irect3d11::interop::CreateDirect3D11DeviceFromDXGIDevice IDXGIDevice에서 IDirect3DDevice의 instance 만듭니다.
graphics::d irectx::d irect3d11::interop::CreateDirect3D11SurfaceFromDXGISurface IDXGISurface에서 IDirect3DSurface의 instance 만듭니다.
graphics::d irectx::d irect3d11::interop::CreateDirect3DDevice IDXGIDevice에서 IDirect3DDevice의 instance 만듭니다.
graphics::d irectx::d irect3d11::interop::CreateDirect3DSurface IDXGISurface에서 IDirect3DSurface의 instance 만듭니다.
graphics::d irectx::d irect3d11::interop::GetDXGIInterface IDirect3DDevice instance DXGI 인터페이스를 검색합니다.
graphics::d irectx::d irect3d11::interop::GetDXGIInterface IDirect3DSurface instance DXGI 인터페이스를 검색합니다.
graphics::d irectx::d irect3d11::interop::GetDXGIInterfaceFromObject 개체에서 DXGI 인터페이스를 검색합니다.

구조체

테이블 확장

graphics::d irectx::d irect3d11::interop::IDirect3DDxgiInterfaceAccess IDirect3DDxgiInterfaceAccess는 COM 인터페이스로, IDirect3DDevice 또는 IDirect3DSurface를 구현하는 모든 항목에 의해 구현되어야 합니다.

Last updated on 2025. 10. 01.

windows.graphics.display.interop.h 헤더

아티클2023. 08. 27.

Windows.Graphics.Display 네임스페이스와 데스크톱 interop용 API.

windows.graphics.display.interop.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

graphics::d isplay::interop::IDisplayInformationStaticsInterop DisplayInformation 정품 인증 팩터리에 대한 데스크톱 interop 인터페이스입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

windows.graphics.effects.interop.h 헤더

아티클2023. 08. 27.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- w_graph_fx
- Windows 런타임 C++ 참조

windows.graphics.effects.interop.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

graphics::effects::interop::IGraphicsEffectD2D1Interop IGraphicsEffect에 대응하는 를 제공하고 메타데이터 쿼리를 허용하는 네이티브 상호 운용 인터페이스입니다. 이 인터페이스는 C++에서만 사용할 수 있습니다.

열거형

GRAPHICS_EFFECT_PROPERTY_MAPPING 강력한 형식의 효과 속성이 기본 Direct2D 효과 속성에 매핑되는 방식을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

windows.graphics.holographic.interop.h 헤더

2023. 08. 27.

헤더의 API를 `Windows.Graphics.Holographic.Interop.h` 사용하면 Windows Mixed Reality 앱에서 Direct3D 12를 사용할 수 있습니다. 이 헤더에 지정된 인터페이스는 COM 인터페이스 포인터를 사용하여 DirectX COM 개체를 Windows.Graphics.Holographic 네임스페이스의 Windows 런타임 개체에 대한 매개 변수로 전달하므로 Windows Mixed Reality 앱에서 추가 오버헤드 없이 Direct3D 12 버퍼 리소스를 만들고 사용할 수 있습니다.

이 API 집합에 대한 샘플 코드는 Windows Mixed Reality Direct3D 12 앱 템플릿에 포함되어 있습니다. Windows Mixed Reality Direct3D 12 앱 템플릿에는 헤더에 제공되는 대부분의 API에 `Windows.Graphics.Holographic.Interop.h` 대한 상용구 코드가 포함되어 있으며 Windows Mixed Reality PC, HoloLens 2 및 HoloLens 2 에뮬레이터에서 회전하는 큐브를 렌더링합니다.

이 헤더는 Direct3D 12 그래픽에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Direct3D 12 그래픽

windows.graphics.holographic.interop.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

테이블 확장

graphics::holographic::interop::IHolographicCameraInterop HolographicCamera를 확장하여 Direct3D 12에서 홀로그램 렌더링을 위한 백 버퍼로 2D 텍스처 리소스를 만들고 사용할 수 있도록 합니다.
graphics::holographic::interop::IHolographicCameraRenderingParametersInterop 홀로그램 렌더링에 Direct3D 12를 사용하는 애플리케이션의 경우 COM interop을 HolographicCameraRenderingParameters 클래스와 상호 운용할 수 있는 nano-COM 인터페이스입니다.
graphics::holographic::interop::IHolographicQuadLayerInterop 홀로그램 렌더링에 Direct3D 12를 사용하는 앱의 경우 HolographicQuadLayer Windows 런타임 클래스와 COM 상호 운용을 허용하는 nano-COM 인터페이스입니다.

graphics::holographic::interop::IHolographicQuadLayerUpdateParametersInterop 홀로그램 렌더링에 Direct3D 12를 사용하는 애플리케이션에 대해 COM interop을 HolographicQuadLayerUpdateParameters 클래스와 상호 운용할 수 있는 nano-COM 인터페이스입니다.

windows.graphics.imaging.interop.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

windows.graphics.imaging.interop.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

graphics::imaging::interop::ISoftwareBitmapNative 소프트웨어 비트맵을 나타냅니다.
graphics::imaging::interop::ISoftwareBitmapNativeFactory ISoftwareBitmapNative 인스턴스를 만듭니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

windows.media.core.interop.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

windows.media.core.interop.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

media::core::interop::IAudioFrameNative 오디오 데이터의 프레임을 나타냅니다.
media::core::interop::IAudioFrameNativeFactory IAudioFrameNative의 인스턴스를 만듭니다.
media::core::interop::IVideoFrameNative 비디오 데이터의 프레임을 나타냅니다.
media::core::interop::IVideoFrameNativeFactory IVideoFrameNative의 인스턴스를 만듭니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

windows.ui.composition.interop.h 헤더

아티클2023. 06. 16.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- w_ui_comp
- Windows 런타임 C++ 참조

windows.ui.composition.interop.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ui::composition::interop::ICompositionDrawingSurfaceInterop RECT를 사용하여 표면 개체에 그릴 영역을 정의할 수 있는 네이티브 상호 운용 인터페이스입니다. 이 인터페이스는 C++에서만 사용할 수 있습니다.
ui::composition::interop::ICompositionDrawingSurfaceInterop2 컴퍼지션 그리기 표면(또는 컴퍼지션 가상 그리기 표면)의 내용을 다시 읽을 수 있는 네이티브 상호 운용 인터페이스입니다.
ui::composition::interop::ICompositionGraphicsDeviceInterop 그래픽 디바이스를 가져오고 설정할 수 있는 네이티브 상호 운용 인터페이스입니다. 이 인터페이스는 C++에서만 사용할 수 있습니다.
ui::composition::interop::ICompositionTextureInterop TBD
ui::composition::interop::ICompositorInterop 스왑 체인 표면 및 그래픽 디바이스를 만들 수 있는 네이티브 상호 운용 인터페이스입니다. 이 인터페이스는 C++에서만 사용할 수 있습니다.
ui::composition::interop::ICompositorInterop2

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

windows.ui.interop.h 헤더

아티클2025. 05. 20.

이 헤더는 다음과 같은 기술을 사용합니다.

- windows 런타임 C++ 참조

windows.ui.interop.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

기능

테이블 확장

ui::interop::GetWindowFromWindowId
ui::interop::GetWindowIdFromWindow hwnd 인수가 유효한 경우 지정된 창 핸들 HWND에 해당하는 WindowId를 가져옵니다.

windows.ui.viewmanagement.core.coreframeworkinputviewinterop.h 헤더

아티클2023. 03. 08.

이 헤더는 WinRT 상호 운용성에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- WinRT 상호 운용성
- CoreInputView 클래스

windows.ui.viewmanagement.core.coreframeworkinputviewinterop.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ui::viewmanagement::core::coreframeworkinputviewinterop::ICoreFrameworkInputViewInterop ICoreFrameworkInputViewInterop 인터페이스를 결정해야 합니다.

windows.ui.xaml.hosting.desktopwindowxamlsource.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

windows.ui.xaml.hosting.desktopwindowxamlsource.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ui::xaml::hosting::d esktopwindowxamlsource::IDesktopWindowXamlSourceNative WinRT XAML 컨트롤을 호스트하는 **DesktopWindowXamlSource** 개체를 데스크톱 앱의 부모 UI 요소에 연결하는 데 사용할 수 있는 멤버를 제공합니다.
ui::xaml::hosting::d esktopwindowxamlsource::IDesktopWindowXamlSourceNative2 WinRT XAML 프레임워크가 WinRT XAML 컨트롤을 호스트하는 **DesktopWindowXamlSource** 개체에 대한 Windows 메시지를 처리할 수 있도록 하는 메서드를 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

windows.ui.xaml.hosting.referencetracker.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

windows.ui.xaml.hosting.referencetracker.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ui::xaml::hosting::referencetracker::IFindReferenceTargetsCallback IReferenceTracker::FindTrackerTargets의 콜백에 대한 인터페이스를 정의합니다. 이 인터페이스의 구현은 찾은 모든 IReferenceTrackerTarget 인스턴스를 FoundTrackerTarget 메서드에 전달해야 합니다.
ui::xaml::hosting::referencetracker::IReferenceTracker XAML 개체 참조를 관리하기 위해 XAML 프레임워크에서 구현하는 인터페이스를 정의합니다.
ui::xaml::hosting::referencetracker::IReferenceTrackerHost XAML 프레임워크에서 사용하는 GC(가비지 수집) 시스템에서 사용하는 전역 서비스를 제공하는 인터페이스를 정의합니다.
ui::xaml::hosting::referencetracker::IReferenceTrackerManager XAML 개체 참조 관리자에 대한 인터페이스를 정의합니다. 이 인터페이스를 구현하여 XAML 개체에서 IReferenceTracker 인스턴스를 관리합니다.
ui::xaml::hosting::referencetracker::IReferenceTrackerTarget XAML에서 참조되는 가비지 수집기 개체에 의해 구현되는 인터페이스를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

windows.ui.xaml.media.dxinterop.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

windows.ui.xaml.media.dxinterop.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ui::xaml::media::d xinterop::ISurfaceImageSourceManagerNative 동일한 프로세스에서 만든 모든 SurfaceImageSource 개체에서 대량 작업을 수행할 수 있습니다.
ui::xaml::media::d xinterop::ISurfaceImageSourceNative Direct2D 드로잉에 대한 공유 고정 크기 표면의 구현을 제공합니다.
ui::xaml::media::d xinterop::ISurfaceImageSourceNativeWithD2D SurfaceImageSource 또는 VirtualSurfaceImageSource에 표시되는 공유 Microsoft DirectX 표면의 구현을 제공합니다.
ui::xaml::media::d xinterop::ISwapChainBackgroundPanelNative XAML과 DirectX 스왑 체인 간의 상호 운용성을 제공합니다.
ui::xaml::media::d xinterop::ISwapChainPanelNative ISwapChainPanelNative 인터페이스(windows.ui.xaml.media.dxinterop.h)는 XAML과 DirectX 스왑 체인 간의 상호 운용성을 제공합니다.
ui::xaml::media::d xinterop::ISwapChainPanelNative2 ISwapChainPanelNative2 인터페이스(windows.ui.xaml.media.dxinterop.h)는 XAML과 DirectX 스왑 체인 간의 상호 운용성을 제공합니다.
ui::xaml::media::d xinterop::IVirtualSurfaceImageSourceNative DirectX 드로잉에 대한 큰(화면 크기보다 큰) 공유 표면의 구현을 제공합니다.

ui::xaml::media::d xinterop::IVirtualSurfaceUpdatesCallbackNative VirtualSurfaceImageSource가 업데이트를 요청할 때 그리기 동작을 구현하기 위한 인터페이스를 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

windowsceip.h 헤더

아티클2023. 08. 27.

이 헤더는 System Services에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 시스템 서비스

windowsceip.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CeipIsOptedIn 사용자가 CEIP(고객 환경 개선 프로그램)의 일부로 SQM 데이터 수집을 옵트인했는지 여부를 확인합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

windowsdefender.h 헤더

아티클2023. 08. 27.

이 헤더는 레거시 Windows 환경 기능에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 레거시 Windows 환경 기능

windowsdefender.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WDEnable Windows Defender 상태 켜기 또는 끄기로 변경합니다.
WDStatus Windows Defender 현재 상태 반환합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

windowsstoragecom.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

windowsstoragecom.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IOplockBreakingHandler 이 인터페이스는 현재 구현되지 않습니다.
IRandomAccessStreamFileAccessMode 임의 액세스 바이트 스트림을 열기 위해 StorageFile.OpenAsync 메서드를 호출할 때 사용된 파일 액세스 모드에 대한 액세스를 제공합니다.
IStorageFolderHandleAccess 스토리지 폴더의 운영 체제 핸들에 대한 액세스를 제공합니다.
IStorageItemHandleAccess 스토리지 파일의 운영 체제 핸들에 대한 액세스를 제공합니다.
IUnbufferedFileHandleOplockCallback IUnbufferedFileHandleProvider::OpenUnbufferedFileHandle 메서드를 호출하여 가져오는 핸들에 대한 기회 잠금이 끊어질 때 실행할 콜백 메서드를 정의합니다.
IUnbufferedFileHandleProvider StorageFile.OpenAsync 메서드가 만든 임의 액세스 바이트 스트림의 핸들에 대한 액세스를 제공합니다.

열거형

HANDLE_ACCESS_OPTIONS 파일에 대한 핸들의 액세스 수준을 정의합니다.
HANDLE_CREATION_OPTIONS 존재하거나 존재하지 않는 파일에 대해 수행할 작업을 나타냅니다.
HANDLE_OPTIONS 파일 핸들의 플래그를 정의합니다.
HANDLE_SHARING_OPTIONS 파일 핸들의 요청된 공유 모드를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

windowsx.h 헤더

아티클2023. 08. 27.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보
- Windows 및 메시지
- Windows 컨트롤
- Windows GDI

windowsx.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

Button_Enable 단추를 사용하거나 사용하지 않도록 설정합니다.
Button_GetCheck 라디오 단추 또는 확인란의 선택 상태를 가져옵니다. 이 매크로를 사용하거나 BM_GETCHECK 메시지를 명시적으로 보낼 수 있습니다.
Button_GetState 단추 또는 확인란의 상태를 검색합니다. 이 매크로를 사용하거나 BM_GETSTATE 메시지를 명시적으로 보낼 수 있습니다.
Button_GetText 단추의 텍스트를 가져옵니다.
Button_GetTextLength 단추 텍스트의 문자 수를 가져옵니다.
Button_SetCheck 라디오 단추 또는 확인란의 선택 상태를 설정합니다. 이 매크로를 사용하거나 BM_SETCHECK 메시지를 명시적으로 보낼 수 있습니다.
Button_SetState 단추의 강조 표시 상태를 설정합니다. 강조 상태는 사용자가 단추를 누른 것처럼 단추가 강조 표시

되는지 여부를 나타냅니다. 이 매크로를 사용하거나 BM_SETSTATE 메시지를 명시적으로 보낼 수 있습니다.
Button_SetStyle 단추의 스타일을 설정합니다. 이 매크로를 사용하거나 BM_SETSTYLE 메시지를 명시적으로 보낼 수 있습니다.
Button_SetText 단추의 텍스트를 설정합니다.
ComboBox_AddItemData 지정된 위치에 있는 콤보 상자의 목록에 항목 데이터를 추가합니다. 이 매크로를 사용하거나 CB_ADDSTRING 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_AddString 콤보 상자의 목록에 문자열을 추가합니다.
ComboBox_DeleteString 콤보 상자의 목록에서 지정된 위치에 있는 항목을 삭제합니다. 이 매크로를 사용하거나 CB_DELETESTRING 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_Dir 콤보 상자에 표시된 목록에 이름을 추가합니다.
ComboBox_Enable 콤보 상자 컨트롤을 사용하거나 사용하지 않도록 설정합니다.
ComboBox_FindItemData 지정된 항목 데이터가 있는 콤보 상자 목록에서 첫 번째 항목을 찾습니다. 이 매크로를 사용하거나 CB_FINDSTRING 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_FindString 지정된 문자열로 시작하는 콤보 상자 목록에서 첫 번째 문자열을 찾습니다. 이 매크로를 사용하거나 CB_FINDSTRING 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_FindStringExact 대/소문자를 구분하지 않는다는 점을 제외하고 지정된 문자열과 정확히 일치하는 콤보 상자 목록의 첫 번째 문자열을 찾습니다. 이 매크로를 사용하거나 CB_FINDSTRINGEXACT 메시지를 명시적으로 보낼 수 있습니다.

ComboBox_GetCount 콤보 상자의 목록 상자에 있는 항목 수를 가져옵니다. 이 매크로를 사용하거나 CB_GETCOUNT 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_GetCurSel 콤보 상자에서 현재 선택된 항목의 인덱스를 가져옵니다. 이 매크로를 사용하거나 CB_GETCURSEL 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_GetDroppedControlRect 드롭다운 상태에서 콤보 상자의 화면 좌표를 검색합니다. 이 매크로를 사용하거나 CB_GETDROPPEDCONTROLRECT 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_GetDroppedState 콤보 상자 컨트롤의 드롭 목록이 표시되는지 여부를 확인합니다. 이 매크로를 사용하거나 CB_GETDROPPEDSTATE 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_GetExtendedUI 콤보 상자가 기본 UI(사용자 인터페이스)를 사용하는지 또는 확장 UI를 사용하는지 확인합니다. 이 매크로를 사용하거나 CB_GETEXTENDEDUI 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_GetItemData 콤보 상자의 지정된 목록 항목과 연결된 애플리케이션 정의 값을 가져옵니다. 이 매크로를 사용하거나 CB_GETITEMDATA 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_GetItemHeight 콤보 상자에서 목록 항목의 높이를 검색합니다. 이 매크로를 사용하거나 CB_GETITEMHEIGHT 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_GetLBText 콤보 상자의 목록에서 문자열을 가져옵니다. 이 매크로를 사용하거나 CB_GETLBTEXT 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_GetLBTextLen 콤보 상자의 목록에서 문자열의 길이를 가져옵니다. 이 매크로를 사용하거나 CB_GETLBTEXTLEN 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_GetText 콤보 상자 컨트롤에서 텍스트를 검색합니다.

ComboBox_GetTextLength 콤보 상자의 텍스트에서 문자 수를 가져옵니다.
ComboBox_InsertItemData 콤보 상자의 지정된 위치에 목록의 항목 데이터를 삽입합니다. 이 매크로를 사용하거나 CB_INSERTSTRING 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_InsertString 지정된 위치에 있는 콤보 상자의 목록에 문자열을 추가합니다. 이 매크로를 사용하거나 CB_INSERTSTRING 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_LimitText 사용자가 콤보 상자의 편집 컨트롤에 입력할 수 있는 텍스트의 길이를 제한합니다. 이 매크로를 사용하거나 CB_LIMITTEXT 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_ResetContent 콤보 상자의 목록 상자 및 편집 컨트롤에서 모든 항목을 제거합니다. 이 매크로를 사용하거나 CB_RESETCONTENT 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_SelectItemData 지정된 항목 데이터가 있는 항목에 대한 콤보 상자의 목록을 검색합니다. 일치하는 항목이 발견되면 해당 항목이 선택됩니다. 이 매크로를 사용하거나 CB_SELECTSTRING 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_SelectString 지정된 문자열의 문자로 시작하는 항목에 대한 콤보 상자의 목록을 검색합니다. 일치하는 항목이 발견되면 해당 항목이 선택됩니다. 이 매크로를 사용하거나 CB_SELECTSTRING 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_SetCurSel 콤보 상자에서 현재 선택된 항목을 설정합니다. 이 매크로를 사용하거나 CB_SETCURSEL 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_SetExtendedUI 기본 UI(사용자 인터페이스) 또는 CBS_DROPDOWN 또는 CBS_DROPDOWNLIST 스타일이 있는 콤보 상자의 확장 UI를 선택합니다. 이 매크로를 사용하거나 CB_SETEXTENDEDUI 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_SetItemData

콤보 상자에서 지정된 목록 항목과 연결된 애플리케이션 정의 값을 설정합니다. 이 매크로를 사용하거나 CB_SETITEMDATA 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_SetItemHeight 콤보 상자에서 목록 항목 또는 선택 필드의 높이를 설정합니다. 이 매크로를 사용하거나 CB_SETITEMHEIGHT 메시지를 명시적으로 보낼 수 있습니다.
ComboBox_SetText 콤보 상자의 텍스트를 설정합니다.
ComboBox_ShowDropdown 콤보 상자에서 목록을 표시하거나 숨깁니다. 이 매크로를 사용하거나 CB_SHOWDROPDOWN 메시지를 명시적으로 보낼 수 있습니다.
DeleteFont DeleteFont 매크로는 글꼴 개체를 삭제하여 글꼴 개체와 연결된 모든 시스템 리소스를 해제합니다.
Edit_CanUndo 편집 또는 서식 있는 편집 컨트롤의 실행 취소 큐에 작업이 있는지 여부를 결정합니다. 이 매크로를 사용하거나 EM_CANUNDO 메시지를 명시적으로 보낼 수 있습니다.
Edit_EmptyUndoBuffer 편집 또는 서식 있는 편집 컨트롤의 실행 취소 플래그를 다시 설정합니다. 편집 컨트롤 내의 작업을 실행 취소할 수 있을 때마다 실행 취소 플래그가 설정됩니다. 이 매크로를 사용하거나 EM_EMPTYUNDOBUFFER 메시지를 명시적으로 보낼 수 있습니다.
Edit_Enable 편집 컨트롤을 사용하거나 사용하지 않도록 설정합니다.
Edit_FmtLines 여러 줄 편집 컨트롤에서 검색된 텍스트에 소프트 줄 바꿈 문자가 포함되는지 여부를 결정하는 플래그를 설정합니다.
Edit_GetFirstVisibleLine 여러 줄 편집 또는 서식 있는 편집 컨트롤에서 맨 위에 표시되는 줄의 인덱스를 가져옵니다. 이 매크로를 사용하거나 EM_GETFIRSTVISIBLELINE 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetHandle

여러 줄 편집 컨트롤의 텍스트에 대해 현재 할당된 메모리에 대한 핸들을 가져옵니다. 이 매크로를 사용하거나 EM_GETHANDLE 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetLine 편집 또는 서식 있는 편집 컨트롤에서 텍스트 줄을 검색합니다. 이 매크로를 사용하거나 EM_GETLINE 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetLineCount 편집 컨트롤의 텍스트에 있는 줄 수를 가져옵니다. 이 매크로를 사용하거나 EM_GETLINECOUNT 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetModify 편집 또는 서식 있는 편집 컨트롤의 수정 플래그 상태를 가져옵니다. 플래그는 컨트롤의 내용이 수정되었는지 여부를 나타냅니다. 이 매크로를 사용하거나 EM_GETMODIFY 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetPasswordChar 편집 또는 서식 있는 편집 컨트롤의 암호 문자를 가져옵니다. 이 매크로를 사용하거나 EM_GETPASSWORDCHAR 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetRect 편집 컨트롤의 서식 사각형을 가져옵니다. 이 매크로를 사용하거나 EM_GETRECT 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetSel 편집 또는 서식 있는 편집 컨트롤에서 현재 선택 항목의 시작 및 끝 문자 위치를 가져옵니다. 이 매크로를 사용하거나 EM_GETSEL 메시지를 명시적으로 보낼 수 있습니다.
Edit_GetText 편집 컨트롤의 텍스트를 가져옵니다.
Edit_GetTextLength 편집 컨트롤의 텍스트에 있는 문자 수를 가져옵니다.
Edit_GetWordBreakProc 편집 또는 서식 있는 편집 컨트롤의 Wordwrap 함수의 주소를 검색합니다. 이 매크로를 사용하거나 EM_GETWORDBREAKPROC 메시지를 명시적으로 보낼 수 있습니다.
Edit_LimitText

편집 컨트롤에 입력할 수 있는 텍스트의 길이를 제한합니다. 이 매크로를 사용하거나 EM_LIMITTEXT 메시지를 명시적으로 보낼 수 있습니다.
Edit_LineFromChar 여러 줄 편집 또는 서식 있는 편집 컨트롤에서 지정된 문자 인덱스가 포함된 줄의 인덱스를 가져옵니다. 이 매크로를 사용하거나 EM_LINEFROMCHAR 메시지를 명시적으로 보낼 수 있습니다.
Edit_LineIndex 여러 줄 편집 또는 서식 있는 편집 컨트롤에서 지정된 줄 첫 번째 문자의 문자 인덱스를 가져옵니다. 이 매크로를 사용하거나 EM_LINEINDEX 메시지를 명시적으로 보낼 수 있습니다.
Edit_LineLength 편집 또는 서식 있는 편집 컨트롤에서 줄의 길이(문자)를 검색합니다. 이 매크로를 사용하거나 EM_LINELENGTH 메시지를 명시적으로 보낼 수 있습니다.
Edit_ReplaceSel 편집 컨트롤 또는 서식 있는 편집 컨트롤에서 선택한 텍스트를 지정된 텍스트로 바꿉니다. 이 매크로를 사용하거나 EM_REPLACESEL 메시지를 명시적으로 보낼 수 있습니다.
Edit_Scroll 여러 줄 편집 또는 서식 있는 편집 컨트롤에서 텍스트를 세로로 스크롤합니다. 이 매크로를 사용하거나 EM_SCROLL 메시지를 명시적으로 보낼 수 있습니다.
Edit_ScrollCaret 편집 또는 서식 있는 편집 컨트롤에서 캐럿을 뷰로 스크롤합니다. 이 매크로를 사용하거나 EM_SCROLLCARET 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetHandle 여러 줄 편집 컨트롤에서 사용할 메모리의 핸들을 설정합니다. 이 매크로를 사용하거나 EM_SETHANDLE 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetModify 편집 컨트롤에 대한 수정 플래그를 설정하거나 지웁니다. 수정 플래그는 편집 컨트롤 내의 텍스트가 수정되었는지 여부를 나타냅니다. 이 매크로를 사용하거나 EM_SETMODIFY 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetPasswordChar 편집 또는 서식 있는 편집 컨트롤의 암호 문자를 설정하거나 제거합니다. 암호 문자가 설정되면 사용자가 입력한 문자 대신 해당 문자가 표시됩니다. 이 매크로를 사용하거나 EM_SETPASSWORDCHAR 메시지를 명시적으로 보낼 수 있습니다.

Edit_SetReadOnly 편집 또는 서식 있는 편집 컨트롤의 읽기 전용 스타일(ES_READONLY)을 설정하거나 제거합니다. 이 매크로를 사용하거나 EM_SETREADONLY 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetRect 편집 컨트롤의 서식 사각형을 설정합니다. 이 매크로를 사용하거나 EM_SETRECT 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetRectNoPaint 여러 줄 편집 컨트롤의 서식 사각형을 설정합니다. 이 매크로는 편집 컨트롤 창을 다시 그리지 않는다는 점을 제외하고 Edit_SetRect와 동일합니다. 이 매크로를 사용하거나 EM_SETRECTNP 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetSel 편집 또는 서식 있는 편집 컨트롤에서 문자 범위를 선택합니다. 이 매크로를 사용하거나 EM_SETSEL 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetTabStops 여러 줄 편집 또는 서식 있는 편집 컨트롤에서 탭 정지를 설정합니다. 텍스트가 컨트롤에 복사되면 텍스트의 모든 탭 문자로 인해 다음 탭 정지까지 공간이 생성됩니다. 이 매크로를 사용하거나 EM_SETTABSTOPS 메시지를 명시적으로 보낼 수 있습니다.
Edit_SetText 편집 컨트롤의 텍스트를 설정합니다.
Edit_SetWordBreakProc 편집 컨트롤의 기본 단어 줄 바꿈 함수를 애플리케이션 정의 단어 줄 바꿈 함수로 바꿉니다. 이 매크로를 사용하거나 EM_SETWORDBREAKPROC 메시지를 명시적으로 보낼 수 있습니다.
Edit_Undo 편집 또는 서식 있는 편집 컨트롤의 실행 취소 큐에서 마지막 작업을 실행 취소합니다. 이 매크로를 사용하거나 EM_UNDO 메시지를 명시적으로 보낼 수 있습니다.
GET_X_LPARAM 지정된 LPARAM 값에서 서명된 x 좌표를 검색합니다.
GET_Y_LPARAM 지정된 LPARAM 값에서 서명된 y 좌표를 검색합니다.

ListBox_AddItemData 지정된 위치의 목록 상자에 항목 데이터를 추가합니다. 이 매크로를 사용하거나 LB_ADDSTRING 메시지를 명시적으로 보낼 수 있습니다.
ListBox_AddString 목록 상자에 문자열을 추가합니다.
ListBox_DeleteString 목록 상자의 지정된 위치에 있는 항목을 삭제합니다. 이 매크로를 사용하거나 LB_DELETESTRING 메시지를 명시적으로 보낼 수 있습니다.
ListBox_Dir 목록 상자에 표시된 목록에 이름을 추가합니다.
ListBox_Enable 목록 상자 컨트롤을 사용하거나 사용하지 않도록 설정합니다.
ListBox_FindItemData 목록 상자에서 지정된 항목 데이터가 있는 첫 번째 항목을 찾습니다. 이 매크로를 사용하거나 LB_FINDSTRING 메시지를 명시적으로 보낼 수 있습니다.
ListBox_FindString 지정된 문자열로 시작하는 목록 상자에서 첫 번째 문자열을 찾습니다. 이 매크로를 사용하거나 LB_FINDSTRING 메시지를 명시적으로 보낼 수 있습니다.
ListBox_FindStringExact 대/소문자를 구분하지 않는 검색 결과를 제외하고 지정된 문자열과 정확히 일치하는 첫 번째 목록 상자 문자열을 찾습니다. 이 매크로를 사용하거나 LB_FINDSTRINGEXACT 메시지를 명시적으로 보낼 수 있습니다.
ListBox_GetCaretIndex 다중 선택 목록 상자에 포커스 사각형이 있는 목록 상자 항목의 인덱스를 검색합니다. 항목을 선택하거나 선택하지 않을 수 있습니다. 이 매크로를 사용하거나 LB_GETCARETINDEX 메시지를 명시적으로 보낼 수 있습니다.
ListBox_GetCount 목록 상자에 있는 항목의 개수를 가져옵니다. 이 매크로를 사용하거나 LB_GETCOUNT 메시지를 명시적으로 보낼 수 있습니다.

ListBox_GetCurSel 단일 선택 목록 상자에서 현재 선택한 항목의 인덱스 를 가져옵니다. 이 매크로를 사용하거나 LB_GETCURSEL 메시지를 명시적으로 보낼 수 있습니다.
ListBox_GetHorizontalExtent 목록 상자에 가로 스크롤 막대가 있는 경우 목록 상자를 가로로 스크롤할 수 있는 너비(스크롤 가능한 너비)를 가져옵니다. 이 매크로를 사용하거나 LB_GETHORIZONTALEXTENT 메시지를 명시적으로 보낼 수 있습니다.
ListBox_GetItemData 지정된 목록 상자 항목과 연결된 애플리케이션 정의 값을 가져옵니다. 이 매크로를 사용하거나 LB_GETITEMDATA 메시지를 명시적으로 보낼 수 있습니다.
ListBox_GetItemHeight 목록 상자에서 항목의 높이를 검색합니다.
ListBox_GetItemRect 목록 상자에 현재 표시되는 목록 상자 항목의 경계를 이루는 사각형의 크기를 가져옵니다. 이 매크로를 사용하거나 LB_GETITEMRECT 메시지를 명시적으로 보낼 수 있습니다.
ListBox_GetSel 항목을 선택한 상태를 가져옵니다. 이 매크로를 사용하거나 LB_GETSEL 메시지를 명시적으로 보낼 수 있습니다.
ListBox_GetSelCount 다중 선택 목록 상자에서 선택한 항목 수를 가져옵니다. 이 매크로를 사용하거나 LB_GETSELCOUNT 메시지를 명시적으로 보낼 수 있습니다.
ListBox_GetSelItems 다중 선택 목록 상자에서 선택한 항목의 인덱스를 가져옵니다. 이 매크로를 사용하거나 LB_GETSELITEMS 메시지를 명시적으로 보낼 수 있습니다.
ListBox_GetText 목록 상자에서 문자열을 가져옵니다. 이 매크로를 사용하거나 LB_GETTEXT 메시지를 명시적으로 보낼 수 있습니다.
ListBox_GetTextLen 목록 상자에서 문자열의 길이를 가져옵니다. 이 매크로를 사용하거나 LB_GETTEXTLEN 메시지를 명시적으로 보낼 수 있습니다.

ListBox_GetTopIndex 목록 상자에서 가장 먼저 표시되는 항목의 인덱스를 가져옵니다. 이 매크로를 사용하거나 LB_GETTOPINDEX 메시지를 명시적으로 보낼 수 있습니다.
ListBox_InsertItemData 지정된 위치의 목록 상자에 항목 데이터를 삽입합니다. 이 매크로를 사용하거나 LB_INSERTSTRING 메시지를 명시적으로 보낼 수 있습니다.
ListBox_InsertString 지정된 위치의 목록 상자에 문자열을 추가합니다. 이 매크로를 사용하거나 LB_INSERTSTRING 메시지를 명시적으로 보낼 수 있습니다.
ListBox_ResetContent 목록 상자에서 모든 항목을 제거합니다. 이 매크로를 사용하거나 LB_RESETCONTENT 메시지를 명시적으로 보낼 수 있습니다.
ListBox_SelectItemData 목록 상자에서 지정된 항목 데이터가 있는 항목을 검색합니다. 일치하는 항목이 발견되면 해당 항목이 선택됩니다. 이 매크로를 사용하거나 LB_SELECTSTRING 메시지를 명시적으로 보낼 수 있습니다.
ListBox_SelectString 지정된 문자열의 문자로 시작하는 항목에 대한 목록 상자를 검색합니다. 일치하는 항목을 찾으면 해당 항목이 선택됩니다. 이 매크로를 사용하거나 LB_SELECTSTRING 메시지를 명시적으로 보낼 수 있습니다.
ListBox_SelItemRange 다중 선택 목록 상자에서 하나 이상의 연속 항목을 선택하거나 선택 취소합니다. 이 매크로를 사용하거나 LB_SELITEMRANGE 메시지를 명시적으로 보낼 수 있습니다.
ListBox_SetCaretIndex 포커스 사각형을 다중 선택 목록 상자의 지정된 인덱스 항목으로 설정합니다. 항목이 표시되지 않으면 보기로 스크롤됩니다. 이 매크로를 사용하거나 LB_SETCARETINDEX 메시지를 명시적으로 보낼 수 있습니다.
ListBox_SetColumnWidth 여러 열 목록 상자에 있는 모든 열의 너비를 설정합니다. 이 매크로를 사용하거나 LB_SETCOLUMNWIDTH 메시지를 명시적으로 보낼 수 있습니다.

ListBox_SetCurSel 단일 선택 목록 상자에서 현재 선택한 항목을 설정합니다. 이 매크로를 사용하거나 LB_SETCURSEL 메시지를 명시적으로 보낼 수 있습니다.
ListBox_SetHorizontalExtent 목록 상자를 가로로 스크롤할 수 있는 너비(스크롤 가능한 너비)를 설정합니다.
ListBox_SetItemData 지정된 목록 상자 항목과 연결된 애플리케이션 정의 값을 설정합니다. 이 매크로를 사용하거나 LB_SETITEMDATA 메시지를 명시적으로 보낼 수 있습니다.
ListBox_SetItemHeight 목록 상자에서 항목의 높이를 설정합니다.
ListBox_SetSel 다중 선택 목록 상자에서 항목을 선택하거나 선택 취소합니다. 이 매크로를 사용하거나 LB_SETSEL 메시지를 명시적으로 보낼 수 있습니다.
ListBox_SetTabStops 목록 상자에서 탭 중지 위치를 설정합니다. 이 매크로를 사용하거나 LB_SETTABSTOPS 메시지를 명시적으로 보낼 수 있습니다.
ListBox_SetTopIndex 목록 상자에 지정된 항목이 표시되는지 확인합니다. 이 매크로를 사용하거나 LB_SETTOPINDEX 메시지를 명시적으로 보낼 수 있습니다.
ScrollBar_Enable 스크롤 막대 컨트롤을 사용하거나 사용하지 않도록 설정합니다.
ScrollBar_GetPos 지정된 스크롤 막대에서 스크롤 상자(thumb)의 위치를 검색합니다.
ScrollBar_GetRange 스크롤 막대의 범위를 가져옵니다.
ScrollBar_SetPos 지정된 스크롤 막대에서 스크롤 상자(thumb)의 위치를 설정하고, 요청된 경우 스크롤 막대를 다시 그려 스크롤 상자의 새 위치를 반영합니다. (ScrollBar_SetPos)

ScrollBar_SetRange 스크롤 막대의 범위를 설정합니다.
ScrollBar_Show 스크롤 막대 컨트롤을 표시하거나 숨깁니다.
SelectFont SelectFont 매크로는 지정된 DC(디바이스 컨텍스트)에 글꼴 개체를 선택합니다. 새 글꼴 개체는 이전 글꼴 개체를 대체합니다.
Static_Enable 정적 컨트롤을 사용하거나 사용하지 않도록 설정합니다.
Static_GetIcon SS_ICON 스타일이 있는 정적 컨트롤과 연결된 아이콘에 대한 핸들을 검색합니다. 이 매크로를 사용하거나 STM_GETICON 메시지를 명시적으로 보낼 수 있습니다.
Static_GetText 정적 컨트롤의 텍스트를 가져옵니다.
Static_GetTextLength 정적 컨트롤의 텍스트에 있는 문자 수를 가져옵니다.
Static_SetIcon 정적 컨트롤의 아이콘을 설정합니다. 이 매크로를 사용하거나 STM_SETICON 메시지를 명시적으로 보낼 수 있습니다.
Static_SetText 정적 컨트롤의 텍스트를 설정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winefs.h 헤더

아티클2023. 08. 27.

이 헤더는 데이터 액세스 및 스토리지에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지

winefs.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AddUsersToEncryptedFile 지정된 암호화된 파일에 사용자를 추가합니다.
DuplicateEncryptionInfoFile 한 파일 또는 디렉터리에서 다른 파일로 EFS 메타데이터를 복사합니다.
EncryptionDisable 지정된 디렉터리와 해당 디렉터리에 암호화를 사용하거나 사용하지 않도록 설정합니다.
FreeEncryptionCertificateHashList 인증서 해시 목록을 해제합니다.
QueryRecoveryAgentsOnEncryptedFile 복구 에이전트 목록을 검색하여 지정된 파일을 찾습니다.
QueryUsersOnEncryptedFile 사용자 목록을 검색하여 지정된 파일을 찾습니다.
RemoveUsersFromEncryptedFile 지정된 파일에서 지정된 인증서 해시를 제거합니다.
SetUserFileEncryptionKey 사용자의 현재 키를 지정된 인증서로 설정합니다.

구조체

EFS_CERTIFICATE_BLOB 인증서를 포함합니다.
EFS_HASH_BLOB 인증서 해시를 포함합니다.
ENCRYPTION_CERTIFICATE 인증서와 해당 소유자의 SID를 포함합니다.
ENCRYPTION_CERTIFICATE_HASH 인증서 해시를 포함하고 인증서에 대한 정보를 표시합니다.
ENCRYPTION_CERTIFICATE_HASH_LIST 인증서 해시 목록을 포함합니다.
ENCRYPTION_CERTIFICATE_LIST 인증서 목록을 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winenclaveapi.h 헤더

이 헤더는 System Services 및 보안 Enclave에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- System Services
- 보안 Enclave(신뢰할 수 있는 실행)

> **참고 항목**
>
> VBS Enclave에 이러한 API를 사용하려면 Windows 11 빌드 26100.2314 이상 또는 Windows Server 2025 이상이 필요합니다.

winenclaveapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

기능

테이블 확장

EnclaveCopyIntoEnclave 보안되지 않은 주소(enclave 외부)에서 enclave로 데이터를 복사합니다.
EnclaveCopyOutOfEnclave enclave의 데이터를 신뢰할 수 없는 주소(enclave 외부)로 복사합니다.
EnclaveGetAttestationReport 현재 enclave를 설명하고 enclave의 형식을 담당하는 기관에서 서명하는 Enclave 증명 보고서를 가져옵니다.
EnclaveGetEnclaveInformation 현재 실행 중인 enclave에 대한 정보를 가져옵니다.
EnclaveRestrictContainingProcessAccess 이 함수는 enclave에 의한 액세스를 포함하는 프로세스의 주소 공간으로 제한(또는 복원)합니다.
EnclaveSealData 암호화되지 않은 데이터에서 암호화된 이진 큰 개체(Blob)를 생성합니다.

EnclaveUnsealData 암호화된 이진 큰 개체(Blob)의 암호를 해독합니다.
EnclaveUsesAttestedKeys 이 메서드는 Enclave가 안전한 증명 키를 사용하는지 여부를 나타냅니다.
EnclaveVerifyAttestationReport 현재 시스템에서 생성된 증명 보고서를 확인합니다.

Last updated on 2025. 06. 18.

winerror.h 헤더

아티클2023. 08. 27.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)
- Direct2D
- DXGI
- 네트워크 액세스 보호
- RPC(원격 프로시저 호출)
- 시스템 서비스
- 작업 Scheduler
- Windows 애니메이션 관리자
- Windows 이벤트 로그
- Windows 필터링 플랫폼

winerror.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

실패 모든 상태 값에 대한 오류에 대한 일반 테스트를 제공합니다.
HRESULT_CODE 지정된 HRESULT의 코드 부분을 추출합니다.
HRESULT_FACILITY 지정된 HRESULT의 기능을 추출합니다.
HRESULT_FROM_NT NT 상태 값을 HRESULT 값에 매핑합니다.
HRESULT_FROM_WIN32 시스템 오류 코드를 HRESULT 값에 매핑합니다.
HRESULT_SEVERITY 지정된 HRESULT의 심각도 필드를 추출합니다.

IS_ERROR 모든 상태 값에 대한 오류에 대한 일반 테스트를 제공합니다.
MAKE_HRESULT MAKE_HRESULT 매크로(winerror.h)는 구성 요소 조각에서 HRESULT 값을 만듭니다.
MAKE_SCODE 구성 요소 조각에서 SCODE 값을 만듭니다.
SCODE_CODE 지정된 SCODE의 코드 부분을 추출합니다.
SCODE_FACILITY 지정된 SCODE의 기능을 추출합니다.
SCODE_SEVERITY 지정된 SCODE의 심각도 필드를 추출합니다.
성공 모든 상태 값에 대한 성공에 대한 일반 테스트를 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winevt.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows 이벤트 로그에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 이벤트 로그

winevt.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

EvtArchiveExportedLog 지정된 로그 파일의 이벤트에 지역화된 문자열을 추가합니다.
EvtCancel 핸들에서 보류 중인 모든 작업을 취소합니다.
EvtClearLog 지정된 채널에서 모든 이벤트를 제거하고 대상 로그 파일에 씁니다.
EvtClose 열린 핸들을 닫습니다.
EvtCreateBookmark 채널에서 이벤트를 식별하는 책갈피를 만듭니다.
EvtCreateRenderContext 렌더링하려는 이벤트의 정보를 지정하는 컨텍스트를 만듭니다.
EvtExportLog 지정된 채널 또는 로그 파일에서 이벤트를 복사하고 대상 로그 파일에 씁니다.
EvtFormatMessage 메시지 문자열의 형식을 지정합니다. (EvtFormatMessage)
EvtGetChannelConfigProperty

지정된 채널 구성 속성을 가져옵니다.
EvtGetEventInfo 이벤트를 선택한 구조적 XML 쿼리와 이벤트가 포함된 채널 또는 로그 파일을 식별하는 정보를 가져옵니다.
EvtGetEventMetadataProperty 지정된 이벤트 메타데이터 속성을 가져옵니다.
EvtGetExtendedStatus 현재 오류에 대한 확장된 오류 정보가 포함된 문자 메시지를 가져옵니다.
EvtGetLogInfo 채널 또는 로그 파일에 대한 정보를 가져옵니다.
EvtGetObjectArrayProperty 배열의 지정된 개체에서 공급자 메타데이터 속성을 가져옵니다.
EvtGetObjectArraySize 개체 배열의 요소 수를 가져옵니다.
EvtGetPublisherMetadataProperty 지정된 공급자 메타데이터 속성을 가져옵니다.
EvtGetQueryInfo 쿼리가 액세스하려고 시도한 채널 또는 로그 파일 목록을 식별하는 실행한 쿼리에 대한 정보를 가져옵니다. 또한 함수는 각 액세스의 성공 또는 실패를 나타내는 반환 코드 목록을 가져옵니다.
EvtNext 쿼리 또는 구독 결과에서 다음 이벤트를 가져옵니다.
EvtNextChannelPath 열거자에서 채널 이름을 가져옵니다.
EvtNextEventMetadata 열거자에서 이벤트 정의를 가져옵니다.
EvtNextPublisherId

열거자에서 공급자의 식별자를 가져옵니다.
EvtOpenChannelConfig 채널의 구성 속성을 읽거나 수정하는 데 사용하는 핸들을 가져옵니다.
EvtOpenChannelEnum 컴퓨터에 등록된 채널 목록을 열거하는 데 사용하는 핸들을 가져옵니다.
EvtOpenEventMetadataEnum 공급자가 정의하는 이벤트 목록을 열거하는 데 사용하는 핸들을 가져옵니다.
EvtOpenLog 채널 또는 로그 파일에 대한 정보를 가져오는 데 사용할 수 있는 채널 또는 로그 파일에 대한 핸들을 가져옵니다.
EvtOpenPublisherEnum 컴퓨터에서 등록된 공급자 목록을 열거하는 데 사용하는 핸들을 가져옵니다.
EvtOpenPublisherMetadata 지정된 공급자의 메타데이터를 읽는 데 사용하는 핸들을 가져옵니다.
EvtOpenSession 다른 Windows 이벤트 로그 함수를 호출할 때 사용할 수 있는 원격 컴퓨터에 대한 연결을 설정합니다.
EvtQuery 지정된 쿼리 조건과 일치하는 채널 또는 로그 파일에서 이벤트를 검색하는 쿼리를 실행합니다.
EvtRender 지정한 렌더링 컨텍스트에 따라 XML 조각을 렌더링합니다.
EvtSaveChannelConfig 채널 구성에 대한 변경 내용을 저장합니다.
EvtSeek 쿼리 결과 집합의 특정 이벤트를 찾습니다.
EvtSetChannelConfigProperty

채널의 지정된 구성 속성을 설정합니다.
EvtSubscribe 지정된 쿼리 조건과 일치하는 채널 또는 로그 파일에서 현재 및 이후 이벤트를 수신할 구독을 만듭니다.
EvtUpdateBookmark 지정된 이벤트를 식별하는 정보로 책갈피를 업데이트.

콜백 함수

EVT_SUBSCRIBE_CALLBACK EvtSubscribe 함수를 호출하여 쿼리와 일치하는 이벤트를 수신하는 경우 이 콜백을 구현합니다.

구조체

EVT_RPC_LOGIN 원격 컴퓨터에 연결하는 데 사용되는 정보를 포함합니다.
EVT_VARIANT 이벤트 데이터 또는 속성 값을 포함합니다.

열거형

EVT_CHANNEL_CLOCK_TYPE 이벤트 채널을 로깅할 때 사용할 타임스탬프를 지정하는 값을 정의합니다.
EVT_CHANNEL_CONFIG_PROPERTY_ID 채널의 구성 속성을 식별하는 식별자를 정의합니다.

EVT_CHANNEL_ISOLATION_TYPE 채널에 적용할 기본 액세스 권한을 정의합니다.
EVT_CHANNEL_REFERENCE_FLAGS 채널을 참조하는 방법을 지정하는 값을 정의합니다.
EVT_CHANNEL_SID_TYPE 이벤트에 이벤트를 기록한 보안 주체의 SID(보안 식별자)가 포함되어 있는지 여부를 결정하는 값을 정의합니다.
EVT_CHANNEL_TYPE 채널의 형식을 정의합니다.
EVT_EVENT_METADATA_PROPERTY_ID 이벤트 정의의 메타데이터 속성을 식별하는 식별자를 정의합니다.
EVT_EVENT_PROPERTY_ID 검색할 쿼리 정보를 결정하는 값을 정의합니다.
EVT_EXPORTLOG_FLAGS 이벤트가 채널 또는 로그 파일에서 제공되는지 여부를 나타내는 값을 정의합니다.
EVT_FORMAT_MESSAGE_FLAGS 이벤트의 메시지 문자열을 형식으로 지정하는 값을 정의합니다.
EVT_LOG_PROPERTY_ID 채널 또는 로그 파일의 로그 파일 메타데이터 속성을 식별하는 식별자를 정의합니다.
EVT_LOGIN_CLASS 원격 컴퓨터에 연결하는 데 사용할 수 있는 연결 방법 유형을 정의합니다.
EVT_OPEN_LOG_FLAGS 채널 또는 내보낸 로그 파일을 열지 여부를 지정하는 값을 정의합니다.
EVT_PUBLISHER_METADATA_PROPERTY_ID 공급자의 메타데이터 속성을 식별하는 식별자를 정의합니다.

EVT_QUERY_FLAGS 쿼리 결과를 반환하는 방법과 채널 또는 로그 파일에 대해 쿼리할지 여부를 지정하는 값을 정의합니다.
EVT_QUERY_PROPERTY_ID 검색할 수 있는 쿼리 정보를 식별하는 식별자를 정의합니다.
EVT_RENDER_CONTEXT_FLAGS 이벤트에서 액세스할 정보 유형을 지정하는 값을 정의합니다.
EVT_RENDER_FLAGS 렌더링할 항목을 지정하는 값을 정의합니다.
EVT_RPC_LOGIN_FLAGS 원격 컴퓨터에 연결할 때 사용자를 인증하는 데 사용할 수 있는 인증 유형을 정의합니다.
EVT_SEEK_FLAGS 검색할 결과 집합의 상대 위치를 정의합니다.
EVT_SUBSCRIBE_FLAGS 이벤트 구독을 시작할 시기를 지정하는 가능한 값을 정의합니다.
EVT_SUBSCRIBE_NOTIFY_ACTION 구독 서비스가 콜백에 제공할 수 있는 데이터 형식을 정의합니다.
EVT_SYSTEM_PROPERTY_ID 이벤트의 시스템별 속성을 식별하는 식별자를 정의합니다.
EVT_VARIANT_TYPE 변형 데이터 항목의 가능한 데이터 형식을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winfax.h 헤더

아티클2023. 08. 27.

이 헤더는 팩스 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 팩스 서비스

winfax.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

FaxCompleteJobParamsA FaxCompleteJobParams 함수는 팩스 클라이언트 애플리케이션에 대한 FAX_COVERPAGE_INFO 구조와 FAX_JOB_PARAM 구조를 모두 만듭니다. (ANSI)
FaxCompleteJobParamsW FaxCompleteJobParams 함수는 팩스 클라이언트 애플리케이션에 대한 FAX_COVERPAGE_INFO 구조와 FAX_JOB_PARAM 구조를 모두 만듭니다. (유니코드)
FaxConnectFaxServerA FaxConnectFaxServer 함수는 팩스 클라이언트 애플리케이션을 로컬 팩스 서버에 연결합니다. 함수는 작업, 디바이스, 구성 및 문서 관리를 용이하게 하는 다른 팩스 클라이언트 함수를 호출하는 데 필요한 팩스 서버 핸들을 반환합니다. (ANSI)
FaxConnectFaxServerW FaxConnectFaxServer 함수는 팩스 클라이언트 애플리케이션을 로컬 팩스 서버에 연결합니다. 함수는 작업, 디바이스, 구성 및 문서 관리를 용이하게 하는 다른 팩스 클라이언트 함수를 호출하는 데 필요한 팩스 서버 핸들을 반환합니다. (유니코드)
FaxEnableRoutingMethodA FaxEnableRoutingMethod 함수는 특정 팩스 디바이스에 대한 팩스 라우팅 방법을 사용하거나 사용하지 않도록 설정합니다. 팩스 관리 애플리케이션은 일반적으로 디바이스 관리를 위해 이 함수를 호출합니다. (ANSI)
FaxEnableRoutingMethodW FaxEnableRoutingMethod 함수는 특정 팩스 디바이스에 대한 팩스 라우팅 방법을 사용하거나 사용하지 않도록 설정합니다. 팩스 관리 애플리케이션은 일반적으로 디바이스 관리를 위해 이 함수를 호출합니다. (유니코드)

FaxEnumGlobalRoutingInfoA FaxEnumGlobalRoutingInfo 함수는 특정 팩스 서버와 연결된 모든 팩스 라우팅 메서드를 열거합니다. (ANSI)
FaxEnumGlobalRoutingInfoW FaxEnumGlobalRoutingInfo 함수는 특정 팩스 서버와 연결된 모든 팩스 라우팅 메서드를 열거합니다. (유니코드)
FaxEnumJobsA FaxEnumJobs 함수는 클라이언트가 연결된 팩스 서버에서 큐에 대기 중인 모든 및 활성 팩스 작업을 열거합니다. 함수는 팩스 클라이언트 애플리케이션에 각 팩스 작업에 대한 자세한 정보를 반환합니다. (ANSI)
FaxEnumJobsW FaxEnumJobs 함수는 클라이언트가 연결된 팩스 서버에서 큐에 대기 중인 모든 및 활성 팩스 작업을 열거합니다. 함수는 팩스 클라이언트 애플리케이션에 각 팩스 작업에 대한 자세한 정보를 반환합니다. (유니코드)
FaxEnumPortsA FaxEnumPorts 함수는 클라이언트가 연결된 팩스 서버에 현재 연결된 모든 팩스 디바이스를 열거합니다. 함수는 팩스 클라이언트 애플리케이션에 각 팩스 포트에 대한 자세한 정보를 반환합니다. (ANSI)
FaxEnumPortsW FaxEnumPorts 함수는 클라이언트가 연결된 팩스 서버에 현재 연결된 모든 팩스 디바이스를 열거합니다. 함수는 팩스 클라이언트 애플리케이션에 각 팩스 포트에 대한 자세한 정보를 반환합니다. (유니코드)
FaxEnumRoutingMethodsA FaxEnumRoutingMethods 함수는 특정 팩스 디바이스에 대한 모든 팩스 라우팅 메서드를 열거합니다. 함수는 팩스 클라이언트 애플리케이션에 대한 각 라우팅 방법에 대한 정보를 반환합니다. (ANSI)
FaxEnumRoutingMethodsW FaxEnumRoutingMethods 함수는 특정 팩스 디바이스에 대한 모든 팩스 라우팅 메서드를 열거합니다. 함수는 팩스 클라이언트 애플리케이션에 대한 각 라우팅 방법에 대한 정보를 반환합니다. (유니코드)
FaxGetConfigurationA

FaxGetConfiguration 함수는 클라이언트가 연결된 팩스 서버에 대한 전역 구성 설정을 팩스 클라이언트 애플리케이션으로 반환합니다. (ANSI)
FaxGetConfigurationW FaxGetConfiguration 함수는 클라이언트가 연결된 팩스 서버에 대한 전역 구성 설정을 팩스 클라이언트 애플리케이션으로 반환합니다. (유니코드)
FaxGetDeviceStatusA FaxGetDeviceStatus 함수는 관심 있는 팩스 디바이스에 대한 현재 상태 정보를 팩스 클라이언트 애플리케이션으로 반환합니다. (ANSI)
FaxGetDeviceStatusW FaxGetDeviceStatus 함수는 관심 있는 팩스 디바이스에 대한 현재 상태 정보를 팩스 클라이언트 애플리케이션으로 반환합니다. (유니코드)
FaxGetJobA 팩스 클라이언트 애플리케이션은 FaxGetJob 함수를 호출하여 지정된 큐에 대기 중인 또는 활성 팩스 작업에 대한 자세한 정보를 검색합니다. 함수는 FAX_JOB_ENTRY 구조체의 정보를 반환합니다. (ANSI)
FaxGetJobW 팩스 클라이언트 애플리케이션은 FaxGetJob 함수를 호출하여 지정된 큐에 대기 중인 또는 활성 팩스 작업에 대한 자세한 정보를 검색합니다. 함수는 FAX_JOB_ENTRY 구조체의 정보를 반환합니다. (유니코드)
FaxGetLoggingCategoriesA FaxGetLoggingCategories 함수는 클라이언트가 연결된 팩스 서버에 대한 현재 로깅 범주를 팩스 클라이언트 애플리케이션으로 반환합니다. (ANSI)
FaxGetLoggingCategoriesW FaxGetLoggingCategories 함수는 클라이언트가 연결된 팩스 서버에 대한 현재 로깅 범주를 팩스 클라이언트 애플리케이션으로 반환합니다. (유니코드)
FaxGetPortA FaxGetPort 함수는 지정된 팩스 포트에 대한 정보를 팩스 클라이언트 애플리케이션에 반환합니다. (ANSI)
FaxGetPortW FaxGetPort 함수는 지정된 팩스 포트에 대한 정보를 팩스 클라이언트 애플리케이션에 반환합니다. (유니코드)

FaxGetRoutingInfoA FaxGetRoutingInfo 함수는 특정 팩스 디바이스와 연결된 팩스 라우팅 방법에 대한 팩스 클라이언트 애플리케이션 라우팅 정보로 돌아갑니다. (ANSI)
FaxGetRoutingInfoW FaxGetRoutingInfo 함수는 특정 팩스 디바이스와 연결된 팩스 라우팅 방법에 대한 팩스 클라이언트 애플리케이션 라우팅 정보로 돌아갑니다. (유니코드)
FaxPrintCoverPageA FaxPrintCoverPage 함수는 팩스 전송 표지 페이지를 팩스 클라이언트 애플리케이션의 지정된 디바이스 컨텍스트로 인쇄합니다. (ANSI)
FaxPrintCoverPageW FaxPrintCoverPage 함수는 팩스 전송 표지 페이지를 팩스 클라이언트 애플리케이션의 지정된 디바이스 컨텍스트로 인쇄합니다. (유니코드)
FaxRegisterRoutingExtensionW FaxRegisterRoutingExtension 함수는 팩스 서비스에 팩스 라우팅 확장 DLL을 등록합니다. 함수는 새 라우팅 확장 DLL을 사용하도록 팩스 서비스 레지스트리를 구성합니다.
FaxRegisterServiceProviderW FaxRegisterServiceProvider 함수는 팩스 서비스 공급자 DLL을 팩스 서비스에 등록합니다. 함수는 팩스 서비스가 다시 시작될 때 새 팩스 서비스 공급자 DLL을 쿼리하고 사용하도록 팩스 서비스 레지스트리를 구성합니다.
FaxSendDocumentA 팩스 클라이언트 애플리케이션은 FaxSendDocument 함수를 호출하여 발신 팩스 전송을 전송하는 팩스 작업을 큐에 대기합니다. (ANSI)
FaxSendDocumentForBroadcastA 팩스 클라이언트 애플리케이션은 FaxSendDocumentForBroadcast 함수를 호출하여 동일한 송신 팩스 전송을 여러 받는 사람에게 전송하는 여러 팩스 작업을 큐에 대기합니다. (ANSI)
FaxSendDocumentForBroadcastW 팩스 클라이언트 애플리케이션은 FaxSendDocumentForBroadcast 함수를 호출하여 동일한 송신 팩스 전송을 여러 받는 사람에게 전송하는 여러 팩스 작업을 큐에 대기합니다. (유니코드)
FaxSendDocumentW

팩스 클라이언트 애플리케이션은 FaxSendDocument 함수를 호출하여 발신 팩스 전송을 전송하는 팩스 작업을 큐에 대기합니다. (유니코드)
FaxSetConfigurationA 팩스 클라이언트 애플리케이션은 FaxSetConfiguration 함수를 호출하여 클라이언트가 연결된 팩스 서버에 대한 전역 구성 설정을 변경합니다. (ANSI)
FaxSetConfigurationW 팩스 클라이언트 애플리케이션은 FaxSetConfiguration 함수를 호출하여 클라이언트가 연결된 팩스 서버에 대한 전역 구성 설정을 변경합니다. (유니코드)
FaxSetGlobalRoutingInfoA 팩스 관리 애플리케이션은 FaxSetGlobalRoutingInfo 함수를 호출하여 팩스 서버에 전역적으로 적용되는 라우팅 우선 순위와 같은 팩스 라우팅 방법 데이터를 수정합니다. (ANSI)
FaxSetGlobalRoutingInfoW 팩스 관리 애플리케이션은 FaxSetGlobalRoutingInfo 함수를 호출하여 팩스 서버에 전역적으로 적용되는 라우팅 우선 순위와 같은 팩스 라우팅 방법 데이터를 수정합니다. (유니코드)
FaxSetJobA 팩스 클라이언트 애플리케이션은 FaxSetJob 함수를 호출하여 지정된 팩스 작업을 일시 중지, 다시 시작, 취소 또는 다시 시작합니다. (ANSI)
FaxSetJobW 팩스 클라이언트 애플리케이션은 FaxSetJob 함수를 호출하여 지정된 팩스 작업을 일시 중지, 다시 시작, 취소 또는 다시 시작합니다. (유니코드)
FaxSetLoggingCategoriesA 팩스 클라이언트 애플리케이션은 FaxSetLoggingCategories 함수를 호출하여 클라이언트가 연결된 팩스 서버에 대한 현재 로깅 범주를 수정합니다. (ANSI)
FaxSetLoggingCategoriesW 팩스 클라이언트 애플리케이션은 FaxSetLoggingCategories 함수를 호출하여 클라이언트가 연결된 팩스 서버에 대한 현재 로깅 범주를 수정합니다. (유니코드)
FaxSetPortA 팩스 클라이언트 애플리케이션은 FaxSetPort 함수를 호출하여 관심 있는 팩스 포트의 구성을 변경합니다. (ANSI)
FaxSetPortW

팩스 클라이언트 애플리케이션은 FaxSetPort 함수를 호출하여 관심 있는 팩스 포트의 구성을 변경합니다. (유니코드)
FaxSetRoutingInfoA 팩스 관리 애플리케이션은 FaxSetRoutingInfo 함수를 호출하여 특정 팩스 디바이스와 연결된 팩스 라우팅 방법에 대한 라우팅 정보를 수정합니다. (ANSI)
FaxSetRoutingInfoW 팩스 관리 애플리케이션은 FaxSetRoutingInfo 함수를 호출하여 특정 팩스 디바이스와 연결된 팩스 라우팅 방법에 대한 라우팅 정보를 수정합니다. (유니코드)
FaxStartPrintJobA 팩스 클라이언트 애플리케이션은 FaxStartPrintJob 함수를 호출하여 지정된 팩스 프린터에서 아웃바운드 팩스 전송 인쇄를 시작합니다. (ANSI)
FaxStartPrintJobW 팩스 클라이언트 애플리케이션은 FaxStartPrintJob 함수를 호출하여 지정된 팩스 프린터에서 아웃바운드 팩스 전송 인쇄를 시작합니다. (유니코드)

콜백 함수

PFAX_RECIPIENT_CALLBACKA FAX_RECIPIENT_CALLBACK 함수는 FaxSendDocumentForBroadcast 함수가 전송에 대한 사용자별 정보를 검색하기 위해 호출하는 애플리케이션 정의 또는 라이브러리 정의 콜백 함수입니다. (ANSI)
PFAX_RECIPIENT_CALLBACKW FAX_RECIPIENT_CALLBACK 함수는 FaxSendDocumentForBroadcast 함수가 전송에 대한 사용자별 정보를 검색하기 위해 호출하는 애플리케이션 정의 또는 라이브러리 정의 콜백 함수입니다. (유니코드)
PFAX_ROUTING_INSTALLATION_CALLBACKW FaxRoutingInstallationCallback 함수는 FaxRegisterRoutingExtension 함수가 팩스 라우팅 확장 DLL을 설치하기 위해 호출하는 라이브러리 정의 콜백 함수입니다.
PFAXABORT 팩스 클라이언트 애플리케이션은 FaxAbort 함수를 호출하여 팩스 작업을 종료합니다.

PFAXACCESSCHECK 팩스 클라이언트 애플리케이션은 FaxAccessCheck 함수를 호출하여 사용자의 팩스 액세스 권한을 쿼리합니다.
PFAXCLOSE FaxClose 함수는 팩스 핸들을 닫습니다.
PFAXFREEBUFFER FaxFreeBuffer 함수는 팩스 클라이언트 애플리케이션에서 함수를 호출한 결과로 이전에 할당된 버퍼와 연결된 리소스를 해제합니다.
PFAXGETPAGEDATA FaxGetPageData 함수는 팩스 작업에 대한 데이터의 첫 번째 페이지를 팩스 클라이언트 애플리케이션으로 반환합니다.
PFAXINITIALIZEEVENTQUEUE FaxInitializeEventQueue 함수는 호출하는 팩스 클라이언트 애플리케이션에 대한 팩스 이벤트 큐를 만듭니다. 큐를 사용하면 애플리케이션이 팩스 서버에서 비동기 이벤트의 알림을 받을 수 있습니다.
PFAXOPENPORT FaxOpenPort 함수는 팩스 클라이언트 애플리케이션에 팩스 포트 핸들을 반환합니다. 포트 핸들은 애플리케이션이 디바이스 관리 및 팩스 문서 라우팅을 용이하게 하는 다른 팩스 클라이언트 함수를 호출할 때 필요합니다.

구조체

FAX_CONFIGURATIONA FAX_CONFIGURATION 구조에는 팩스 서버의 전역 구성 설정에 대한 정보가 포함되어 있습니다. (ANSI)
FAX_CONFIGURATIONW FAX_CONFIGURATION 구조에는 팩스 서버의 전역 구성 설정에 대한 정보가 포함되어 있습니다. (유니코드)
FAX_CONTEXT_INFOA

FAX_CONTEXT_INFO 구조에는 팩스 프린터 디바이스 컨텍스트에 대한 정보가 포함되어 있습니다. SizeOfStruct 멤버가 필요합니다. 다른 멤버에 대한 정보는 FaxStartPrintJob 함수에 대한 호출을 통해 제공됩니다. (ANSI)
FAX_CONTEXT_INFOW FAX_CONTEXT_INFO 구조에는 팩스 프린터 디바이스 컨텍스트에 대한 정보가 포함되어 있습니다. SizeOfStruct 멤버가 필요합니다. 다른 멤버에 대한 정보는 FaxStartPrintJob 함수에 대한 호출을 통해 제공됩니다. (유니코드)
FAX_COVERPAGE_INFOA FAX_COVERPAGE_INFO 구조에는 팩스 전송의 표지에 표시할 데이터가 포함되어 있습니다. SizeOfStruct 및 CoverPageName 멤버가 필요합니다. 다른 멤버는 선택 사항입니다. (ANSI)
FAX_COVERPAGE_INFOW FAX_COVERPAGE_INFO 구조에는 팩스 전송의 표지에 표시할 데이터가 포함되어 있습니다. SizeOfStruct 및 CoverPageName 멤버가 필요합니다. 다른 멤버는 선택 사항입니다. (유니코드)
FAX_DEVICE_STATUSA FAX_DEVICE_STATUS 구조에는 팩스 장치의 현재 상태 대한 정보가 포함되어 있습니다. (ANSI)
FAX_DEVICE_STATUSW FAX_DEVICE_STATUS 구조에는 팩스 장치의 현재 상태 대한 정보가 포함되어 있습니다. (유니코드)
FAX_EVENTA FAX_EVENT 구조체는 I/O 완료 패킷의 내용을 나타냅니다. 팩스 서버는 완료 패킷을 전송하여 팩스 클라이언트 애플리케이션에 비동기 팩스 서버 이벤트를 알립니다. (ANSI)
FAX_EVENTW FAX_EVENT 구조체는 I/O 완료 패킷의 내용을 나타냅니다. 팩스 서버는 완료 패킷을 전송하여 팩스 클라이언트 애플리케이션에 비동기 팩스 서버 이벤트를 알립니다. (유니코드)
FAX_GLOBAL_ROUTING_INFOA FAX_GLOBAL_ROUTING_INFO 구조에는 팩스 서비스와 관련된 하나의 팩스 라우팅 방법에 대한 정보가 포함되어 있습니다. (ANSI)
FAX_GLOBAL_ROUTING_INFOW FAX_GLOBAL_ROUTING_INFO 구조에는 팩스 서비스와 관련된 하나의 팩스 라우팅 방법에 대한 정보가 포함되어 있습니다. (유니코드)
FAX_JOB_ENTRYA

FAX_JOB_ENTRY 구조는 하나의 팩스 작업을 설명합니다. (ANSI)
FAX_JOB_ENTRYW FAX_JOB_ENTRY 구조는 하나의 팩스 작업을 설명합니다. (유니코드)
FAX_JOB_PARAMA FAX_JOB_PARAM 구조에는 팩스 서버가 개별 팩스 전송을 보내는 데 필요한 정보가 포함되어 있습니다. (ANSI)
FAX_JOB_PARAMW FAX_JOB_PARAM 구조에는 팩스 서버가 개별 팩스 전송을 보내는 데 필요한 정보가 포함되어 있습니다. (유니코드)
FAX_LOG_CATEGORYA FAX_LOG_CATEGORY 구조는 하나의 로깅 범주를 설명합니다. (ANSI)
FAX_LOG_CATEGORYW FAX_LOG_CATEGORY 구조는 하나의 로깅 범주를 설명합니다. (유니코드)
FAX_PORT_INFOA FAX_PORT_INFO 구조체는 하나의 팩스 포트를 설명합니다. 데이터에는 디바이스 식별자, 포트 이름 및 현재 상태 및 스테이션 식별자가 포함됩니다. (ANSI)
FAX_PORT_INFOW FAX_PORT_INFO 구조체는 하나의 팩스 포트를 설명합니다. 데이터에는 디바이스 식별자, 포트 이름 및 현재 상태 및 스테이션 식별자가 포함됩니다. (유니코드)
FAX_PRINT_INFOA FAX_PRINT_INFO 구조에는 팩스 서버가 팩스 전송을 인쇄하는 데 필요한 정보가 포함되어 있습니다. 구조에는 보낸 사람 및 받는 사람 데이터, 선택적 청구 코드 및 배달 보고서 정보가 포함됩니다. (ANSI)
FAX_PRINT_INFOW FAX_PRINT_INFO 구조에는 팩스 서버가 팩스 전송을 인쇄하는 데 필요한 정보가 포함되어 있습니다. 구조에는 보낸 사람 및 받는 사람 데이터, 선택적 청구 코드 및 배달 보고서 정보가 포함됩니다. (유니코드)
FAX_ROUTING_METHODA

FAX_ROUTING_METHOD 구조에는 하나의 팩스 디바이스와 관련된 하나의 팩스 라우팅 방법에 대한 정보가 포함됩니다. (ANSI)
FAX_ROUTING_METHODW FAX_ROUTING_METHOD 구조에는 하나의 팩스 디바이스와 관련된 하나의 팩스 라우팅 방법에 대한 정보가 포함됩니다. (유니코드)
FAX_TIME FAX_TIME 구조체는 현재 시간 및 분 동안 개별 멤버를 사용하는 시간을 나타냅니다. 시간은 UTC(협정 세계시)로 표현됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wingdi.h 헤더

아티클2023. 03. 08.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 교환
- Direct3D 9 그래픽
- Directshow
- 디바이스 참조 표시
- Windows 애플리케이션에 대한 국제화
- Opengl
- Tablet PC
- Windows 색 시스템
- Windows GDI

wingdi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AbortDoc AbortDoc 함수는 현재 인쇄 작업을 중지하고 StartDoc 함수에 대한 마지막 호출 이후 그려진 모든 항목을 지웁니다.
AbortPath AbortPath 함수는 지정된 디바이스 컨텍스트의 모든 경로를 닫고 삭제합니다.
AddFontMemResourceEx AddFontMemResourceEx 함수는 메모리 이미지의 글꼴 리소스를 시스템에 추가합니다.
AddFontResourceA AddFontResource 함수는 지정된 파일의 글꼴 리소스를 시스템 글꼴 테이블에 추가합니다. 이후에 모든 응용 프로그램에서 텍스트 출력에 글꼴을 사용할 수 있습니다. (ANSI)
AddFontResourceExA AddFontResourceEx 함수는 지정된 파일의 글꼴 리소스를 시스템에 추가합니다. AddFontResourceEx 함수를 사용하여 추가된 글꼴은 비공개로 표시될 수 있으며 열거할 수 없습니다. (ANSI)

AddFontResourceExW AddFontResourceEx 함수는 지정된 파일의 글꼴 리소스를 시스템에 추가합니다. AddFontResourceEx 함수를 사용하여 추가된 글꼴은 비공개로 표시될 수 있으며 열거할 수 없습니다. (유니코드)
AddFontResourceW AddFontResource 함수는 지정된 파일의 글꼴 리소스를 시스템 글꼴 테이블에 추가합니다. 이후에 모든 응용 프로그램에서 텍스트 출력에 글꼴을 사용할 수 있습니다. (유니코드)
AlphaBlend AlphaBlend 함수는 투명 또는 반투명 픽셀이 있는 비트맵을 표시합니다.
AngleArc AngleArc 함수는 선 세그먼트와 호를 그립니다.
애니메이션 팔레트 AnimatePalette 함수는 지정된 논리 팔레트의 항목을 대체합니다.
Arc Arc 함수는 타원형 호를 그립니다.
ArcTo ArcTo 함수는 타원형 호를 그립니다.
BeginPath BeginPath 함수는 지정된 디바이스 컨텍스트에서 경로 대괄호를 엽니다.
Bitblt BitBlt 함수는 지정된 원본 디바이스 컨텍스트에서 대상 디바이스 컨텍스트로 픽셀 사각형에 해당하는 색 데이터의 비트 블록 전송을 수행합니다.
CancelDC CancelDC 함수는 지정된 DC(디바이스 컨텍스트)에서 보류 중인 작업을 취소합니다.
CheckColorsInGamut CheckColorsInGamut 함수는 지정된 RGB 삼중 집합이 지정된 디바이스의 출력 영역 내에 있는지 여부를 결정합니다. RGB 삼중은 입력 논리 색 공간에서 해석됩니다.

ChoosePixelFormat ChoosePixelFormat 함수는 디바이스 컨텍스트에서 지원하는 적절한 픽셀 형식을 지정된 픽셀 형식 사양과 일치시키려고 시도합니다.
Chord(현) 코드 함수는 코드(타원의 교차점과 시컨트라고 하는 선 세그먼트로 경계가 지정된 영역)를 그립니다. 코드는 현재 펜을 사용하여 윤곽을 표시하고 현재 브러시를 사용하여 채워집니다.
CloseEnhMetaFile CloseEnhMetaFile 함수는 향상된 메타파일 디바이스 컨텍스트를 닫고 향상된 형식 메타파일을 식별하는 핸들을 반환합니다.
CloseFigure CloseFigure 함수는 경로의 열린 그림을 닫습니다.
CloseMetaFile CloseMetaFile 함수는 메타파일 디바이스 컨텍스트를 닫고 Windows 형식 메타파일을 식별하는 핸들을 반환합니다.
Cmyk CMYK 매크로는 지정된 시안, 자홍색, 노란색 및 검은색 값을 결합하여 CMYK 색 값을 만듭니다.
ColorCorrectPalette ColorCorrectPalette 함수는 지정된 디바이스 컨텍스트에서 WCS 1.0 매개 변수를 사용하여 팔레트의 항목을 수정합니다.
ColorMatchToTarget ColorMatchToTarget 함수를 사용하면 대상 디바이스에 표시되는 색을 미리 볼 수 있습니다.
CombineRgn CombineRgn 함수는 두 영역을 결합하고 결과를 세 번째 지역에 저장합니다. 두 영역은 지정된 모드에 따라 결합됩니다.
CombineTransform CombineTransform 함수는 두 개의 세계 공간을 페이지 공간 변환에 연결합니다.
CopyEnhMetaFileA CopyEnhMetaFile 함수는 고급 형식 메타파일의 내용을 지정된 파일에 복사합니다. (ANSI)

CopyEnhMetaFileW CopyEnhMetaFile 함수는 고급 형식 메타파일의 내용을 지정된 파일에 복사합니다. (유니코드)
CopyMetaFileA CopyMetaFile 함수는 Windows 형식 메타파일의 내용을 지정된 파일에 복사합니다. (ANSI)
CopyMetaFileW CopyMetaFile 함수는 Windows 형식 메타파일의 내용을 지정된 파일에 복사합니다. (유니코드)
CreateBitmap CreateBitmap 함수는 지정된 너비, 높이 및 색 형식(색 평면 및 픽셀당 비트)을 사용하여 비트맵을 만듭니다.
CreateBitmapIndirect CreateBitmapIndirect 함수는 지정된 너비, 높이 및 색 형식(색 평면 및 픽셀당 비트)을 사용하여 비트맵을 만듭니다.
CreateBrushIndirect CreateBrushIndirect 함수는 지정된 스타일, 색 및 패턴을 가진 논리 브러시를 만듭니다.
CreateColorSpaceA CreateColorSpace 함수는 논리적 색 공간을 만듭니다. (ANSI)
CreateColorSpaceW CreateColorSpace 함수는 논리적 색 공간을 만듭니다. (유니코드)
CreateCompatibleBitmap CreateCompatibleBitmap 함수는 지정된 디바이스 컨텍스트와 연결된 디바이스와 호환되는 비트맵을 만듭니다.
CreateCompatibleDC CreateCompatibleDC 함수는 지정된 디바이스와 호환되는 DC(메모리 디바이스 컨텍스트)를 만듭니다.
CreateDCA CreateDC 함수는 지정된 이름을 사용하여 디바이스에 대한 DC(디바이스 컨텍스트)를 만듭니다. (ANSI)

CreateDCW CreateDC 함수는 지정된 이름을 사용하여 디바이스에 대한 DC(디바이스 컨텍스트)를 만듭니다. (유니코드)
CreateDIBitmap CreateDIBitmap 함수는 DIB에서 호환되는 비트맵(DDB)을 만들고 필요에 따라 비트맵 비트를 설정합니다.
CreateDIBPatternBrush CreateDIBPatternBrush 함수는 지정된 DIB(디바이스 독립적 비트맵)로 지정된 패턴이 있는 논리 브러시를 만듭니다.
CreateDIBPatternBrushPt CreateDIBPatternBrushPt 함수는 디바이스 독립적 비트맵(DIB)에서 지정한 패턴이 있는 논리 브러시를 만듭니다.
CreateDIBSection CreateDIBSection 함수는 애플리케이션이 직접 쓸 수 있는 DIB를 만듭니다.
CreateDiscardableBitmap CreateDiscardableBitmap 함수는 지정된 디바이스와 호환되는 삭제 가능한 비트맵을 만듭니다.
CreateEllipticRgn CreateEllipticRgn 함수는 타원형 영역을 만듭니다.
CreateEllipticRgnIndirect CreateEllipticRgnIndirect 함수는 타원형 영역을 만듭니다.
CreateEnhMetaFileA CreateEnhMetaFile 함수는 향상된 형식 메타파일을 위한 디바이스 컨텍스트를 만듭니다. 이 디바이스 컨텍스트를 사용하여 디바이스 독립적 그림을 저장할 수 있습니다. (ANSI)
CreateEnhMetaFileW CreateEnhMetaFile 함수는 향상된 형식 메타파일을 위한 디바이스 컨텍스트를 만듭니다. 이 디바이스 컨텍스트를 사용하여 디바이스 독립적 그림을 저장할 수 있습니다. (유니코드)

CreateFontA CreateFont 함수는 지정된 특성을 가진 논리적 글꼴을 만듭니다. 이후에 논리 글꼴을 모든 디바이스의 글꼴로 선택할 수 있습니다. (ANSI)
CreateFontIndirectA CreateFontIndirect 함수는 지정된 특성을 가진 논리적 글꼴을 만듭니다. 이후에는 디바이스 컨텍스트에 대한 현재 글꼴로 글꼴을 선택할 수 있습니다. (ANSI)
CreateFontIndirectExA CreateFontIndirectEx 함수는 지정된 구조체의 특징이 있는 논리 글꼴을 지정합니다. 이후에는 디바이스 컨텍스트에 대한 현재 글꼴로 글꼴을 선택할 수 있습니다. (ANSI)
CreateFontIndirectExW CreateFontIndirectEx 함수는 지정된 구조체의 특징이 있는 논리 글꼴을 지정합니다. 이후에는 디바이스 컨텍스트에 대한 현재 글꼴로 글꼴을 선택할 수 있습니다. (유니코드)
CreateFontIndirectW CreateFontIndirect 함수는 지정된 특성을 가진 논리적 글꼴을 만듭니다. 이후에는 디바이스 컨텍스트에 대한 현재 글꼴로 글꼴을 선택할 수 있습니다. (유니코드)
CreateFontW CreateFont 함수는 지정된 특성을 가진 논리적 글꼴을 만듭니다. 이후에 논리 글꼴을 모든 디바이스의 글꼴로 선택할 수 있습니다. (유니코드)
CreateHalftonePalette CreateHalftonePalette 함수는 지정된 DC(디바이스 컨텍스트)에 대한 하프톤 팔레트를 만듭니다.
CreateHatchBrush CreateHatchBrush 함수는 지정된 빗살 무늬와 색이 있는 논리 브러시를 만듭니다.
CreateICA CreateIC 함수는 지정된 디바이스에 대한 정보 컨텍스트를 만듭니다. (ANSI)
CreateICW CreateIC 함수는 지정된 디바이스에 대한 정보 컨텍스트를 만듭니다. (유니코드)
CreateMetaFileA CreateMetaFile 함수는 Windows 형식 메타파일의 디바이스 컨텍스트를 만듭니다. (ANSI)

CreateMetaFileW CreateMetaFile 함수는 Windows 형식 메타파일의 디바이스 컨텍스트를 만듭니다. (유니코드)
CreatePalette CreatePalette 함수는 논리 팔레트를 만듭니다.
CreatePatternBrush CreatePatternBrush 함수는 지정된 비트맵 패턴을 사용하여 논리 브러시를 만듭니다. 비트맵은 CreateDIBSection 함수에서 만든 DIB 섹션 비트맵이거나 디바이스 종속 비트맵일 수 있습니다.
CreatePen CreatePen 함수는 지정된 스타일, 너비 및 색이 있는 논리 펜을 만듭니다. 이후에 펜을 디바이스 컨텍스트로 선택하고 선과 곡선을 그리는 데 사용할 수 있습니다.
CreatePenIndirect CreatePenIndirect 함수는 구조체에 지정된 스타일, 너비 및 색이 있는 논리적 화장품 펜을 만듭니다.
CreatePolygonRgn CreatePolygonRgn 함수는 다각형 영역을 만듭니다.
CreatePolyPolygonRgn CreatePolyPolygonRgn 함수는 일련의 다각형으로 구성된 영역을 만듭니다. 다각형은 겹칠 수 있습니다.
CreateRectRgn CreateRectRgn 함수는 사각형 영역을 만듭니다.
CreateRectRgnIndirect CreateRectRgnIndirect 함수는 사각형 영역을 만듭니다.
CreateRoundRectRgn CreateRoundRectRgn 함수는 모서리가 둥근 사각형 영역을 만듭니다.
CreateScalableFontResourceA CreateScalableFontResource 함수는 확장 가능한 글꼴에 대한 글꼴 리소스 파일을 만듭니다. (ANSI)

CreateScalableFontResourceW CreateScalableFontResource 함수는 확장 가능한 글꼴에 대한 글꼴 리소스 파일을 만듭니다. (유니코드)
CreateSolidBrush CreateSolidBrush 함수는 지정된 단색이 있는 논리 브러시를 만듭니다.
DeleteColorSpace DeleteColorSpace 함수는 지정된 색 공간을 제거하고 제거합니다.
DeleteDC DeleteDC 함수는 지정된 DC(디바이스 컨텍스트)를 삭제합니다.
DeleteEnhMetaFile DeleteEnhMetaFile 함수는 향상된 형식 메타파일 또는 향상된 형식 메타파일 핸들을 삭제합니다.
DeleteMetaFile DeleteMetaFile 함수는 Windows 형식 메타파일 또는 Windows 형식 메타파일 핸들을 삭제합니다.
DeleteObject DeleteObject 함수는 논리 펜, 브러시, 글꼴, 비트맵, 지역 또는 팔레트를 삭제하여 개체와 연결된 모든 시스템 리소스를 해제합니다. 개체가 삭제되면 지정된 핸들이 더 이상 유효하지 않습니다.
DescribePixelFormat DescribePixelFormat 함수는 hdc와 연결된 디바이스의 iPixelFormat으로 식별되는 픽셀 형식에 대한 정보를 가져옵니다. 이 함수는 해당 픽셀 형식 데이터를 사용하여 ppfd가 가리키는 PIXELFORMATDESCRIPTOR 구조체의 멤버를 설정합니다.
DeviceCapabilitiesA DeviceCapabilities 함수는 프린터 드라이버의 기능을 검색합니다. (ANSI)
DeviceCapabilitiesW DeviceCapabilities 함수는 프린터 드라이버의 기능을 검색합니다. (유니코드)
DPtoLP DPtoLP 함수는 디바이스 좌표를 논리 좌표로 변환합니다. 변환은 디바이스 컨텍스트의 매핑 모드, 창 및 뷰포트의 원본 및 익스텐트 설정 및 세계 변환에 따라 달라집니다.

DrawEscape DrawEscape 함수는 GDI(그래픽 디바이스 인터페이스)를 통해 직접 사용할 수 없는 지정된 비디오 디스플레이의 그리기 기능을 제공합니다.
타원 타원 함수는 타원을 그립니다. 줄임표의 중심은 지정된 경계 사각형의 중심입니다. 타원은 현재 펜을 사용하여 간략하게 설명되며 현재 브러시를 사용하여 채워집니다.
EndDoc EndDoc 함수는 인쇄 작업을 종료합니다.
EndPage EndPage 함수는 애플리케이션이 페이지에 쓰기를 완료했음을 디바이스에 알합니다. 이 함수는 일반적으로 디바이스 드라이버가 새 페이지로 이동하도록 지시하는 데 사용됩니다.
EndPath EndPath 함수는 경로 대괄호를 닫고 대괄호로 정의된 경로를 지정된 디바이스 컨텍스트로 선택합니다.
EnumEnhMetaFile EnumEnhMetaFile 함수는 각 레코드를 검색하고 지정된 콜백 함수에 전달하여 향상된 형식 메타파일 내에서 레코드를 열거합니다.
EnumFontFamiliesA EnumFontFamilies 함수는 지정된 디바이스에서 사용할 수 있는 지정된 글꼴 패밀리의 글꼴을 열거합니다. (ANSI)
EnumFontFamiliesExA EnumFontFamiliesEx 함수는 LOGFONT 구조에서 지정한 글꼴 특성과 일치하는 시스템에서 고유하게 명명된 모든 글꼴을 열거합니다. EnumFontFamiliesEx는 서체 이름, 문자 집합 또는 둘 다를 기반으로 글꼴을 열거합니다. (ANSI)
EnumFontFamiliesExW EnumFontFamiliesEx 함수는 LOGFONT 구조에서 지정한 글꼴 특성과 일치하는 시스템에서 고유하게 명명된 모든 글꼴을 열거합니다. EnumFontFamiliesEx는 서체 이름, 문자 집합 또는 둘 다를 기반으로 글꼴을 열거합니다. (유니코드)

EnumFontFamiliesW EnumFontFamilies 함수는 지정된 디바이스에서 사용할 수 있는 지정된 글꼴 패밀리의 글꼴을 열거합니다. (유니코드)
EnumFontsA EnumFonts 함수는 지정된 디바이스에서 사용할 수 있는 글꼴을 열거합니다. (ANSI)
EnumFontsW EnumFonts 함수는 지정된 디바이스에서 사용할 수 있는 글꼴을 열거합니다. (유니코드)
EnumICMProfilesA EnumICMProfiles 함수는 시스템에서 지정된 디바이스 컨텍스트에 대해 지원하는 다양한 출력 색 프로필을 열거합니다. (ANSI)
EnumICMProfilesW EnumICMProfiles 함수는 시스템에서 지정된 디바이스 컨텍스트에 대해 지원하는 다양한 출력 색 프로필을 열거합니다. (유니코드)
EnumMetaFile EnumMetaFile 함수는 각 레코드를 검색하고 지정된 콜백 함수에 전달하여 Windows 형식 메타파일 내의 레코드를 열거합니다.
EnumObjects EnumObjects 함수는 지정된 DC(디바이스 컨텍스트)에 사용할 수 있는 펜 또는 브러시를 열거합니다.
EqualRgn EqualRgn 함수는 지정된 두 영역이 동일한지 여부를 확인합니다. 이 함수는 크기와 모양이 같은 경우 두 영역이 동일한 것으로 간주합니다.
Esc 애플리케이션이 GDI를 통해 사용할 수 없는 시스템 정의 디바이스 기능에 액세스할 수 있도록 합니다.
ExcludeClipRect ExcludeClipRect 함수는 지정된 사각형을 뺀 기존 클리핑 영역으로 구성된 새 클리핑 영역을 만듭니다.

ExtCreatePen ExtCreatePen 함수는 지정된 스타일, 너비 및 브러시 특성을 가진 논리적 화장품 또는 기하학적 펜을 만듭니다.
ExtCreateRegion ExtCreateRegion 함수는 지정된 지역 및 변환 데이터에서 영역을 만듭니다.
ExtEscape ExtEscape 함수를 사용하면 애플리케이션이 GDI를 통해 사용할 수 없는 디바이스 기능에 액세스할 수 있습니다.
ExtFloodFill ExtFloodFill 함수는 현재 브러시로 디스플레이 표면의 영역을 채웁니다.
ExtSelectClipRgn ExtSelectClipRgn 함수는 지정된 모드를 사용하여 지정된 영역을 현재 클리핑 영역과 결합합니다.
ExtTextOutA ExtTextOut 함수는 현재 선택한 글꼴, 배경색 및 텍스트 색을 사용하여 텍스트를 그립니다. 필요에 따라 클리핑, 불투명 또는 둘 다에 사용할 차원을 제공할 수 있습니다. (ANSI)
ExtTextOutW ExtTextOut 함수는 현재 선택한 글꼴, 배경색 및 텍스트 색을 사용하여 텍스트를 그립니다. 필요에 따라 클리핑, 불투명 또는 둘 다에 사용할 차원을 제공할 수 있습니다. (유니코드)
FillPath FillPath 함수는 현재 경로의 열려 있는 모든 그림을 닫고 현재 브러시 및 다각형 채우기 모드를 사용하여 경로의 내부를 채웁니다.
FillRgn FillRgn 함수는 지정된 브러시를 사용하여 영역을 채웁니다.
FlattenPath FlattenPath 함수는 선택한 경로의 모든 곡선을 현재 DC(디바이스 컨텍스트)로 변환하여 각 곡선을 선 시퀀스로 변환합니다.

FloodFill FloodFill 함수는 디스플레이 표면의 영역을 현재 브러시로 채웁니다. 영역은 색 매개 변수에 지정된 대로 경계로 간주됩니다.
FrameRgn FrameRgn 함수는 지정된 브러시를 사용하여 지정된 영역 주위에 테두리를 그립니다.
GdiAlphaBlend GdiAlphaBlend 함수는 투명 또는 반투명 픽셀이 있는 비트맵을 표시합니다.
GdiComment GdiComment 함수는 버퍼의 주석을 지정된 고급 형식 메타파일로 복사합니다.
GdiFlush GdiFlush 함수는 호출 스레드의 현재 일괄 처리를 플러시합니다.
GdiGetBatchLimit GdiGetBatchLimit 함수는 호출 스레드의 현재 일괄 처리에 누적될 수 있는 최대 함수 호출 수를 반환합니다. 이 제한을 초과할 때마다 시스템에서 현재 일괄 처리를 플러시합니다.
GdiGradientFill GdiGradientFill 함수는 사각형과 삼각형 구조를 채웁니다.
GdiSetBatchLimit GdiSetBatchLimit 함수는 호출 스레드의 현재 일괄 처리에 누적될 수 있는 함수 호출의 최대 수를 설정합니다. 이 제한을 초과할 때마다 시스템에서 현재 일괄 처리를 플러시합니다.
GdiTransparentBlt GdiTransparentBlt 함수는 지정된 원본 디바이스 컨텍스트에서 대상 디바이스 컨텍스트로 픽셀 사각형에 해당하는 색 데이터의 비트 블록 전송을 수행합니다.
GetArcDirection GetArcDirection 함수는 지정된 디바이스 컨텍스트에 대한 현재 호 방향을 검색합니다. Arc 및 사각형 함수는 호 방향을 사용합니다.
GetAspectRatioFilterEx GetAspectRatioFilterEx 함수는 현재 가로 세로 비율 필터에 대한 설정을 검색합니다.

GetBitmapBits GetBitmapBits 함수는 지정된 디바이스 종속 비트맵의 비트맵 비트를 버퍼에 복사합니다.
GetBitmapDimensionEx GetBitmapDimensionEx 함수는 호환되는 비트맵의 차원을 검색합니다. 검색된 차원은 SetBitmapDimensionEx 함수에 의해 설정되어야 합니다.
GetBkColor GetBkColor 함수는 지정된 디바이스 컨텍스트에 대한 현재 배경색을 반환합니다.
GetBkMode GetBkMode 함수는 지정된 디바이스 컨텍스트에 대한 현재 백그라운드 혼합 모드를 반환합니다. 디바이스 컨텍스트의 배경 혼합 모드는 실선이 아닌 텍스트, 빗살 무늬 브러시 및 펜 스타일에 영향을 줍니다.
GetBoundsRect GetBoundsRect 함수는 지정된 디바이스 컨텍스트에 대해 현재 누적된 경계 사각형을 가져옵니다.
GetBrushOrgEx GetBrushOrgEx 함수는 지정된 디바이스 컨텍스트에 대한 현재 브러시 원본을 검색합니다. 이 함수는 GetBrushOrg 함수를 대체합니다.
GetBValue GetBValue 매크로는 빨강, 녹색, 파랑(RGB) 값의 파란색 구성 요소에 대한 강도 값을 검색합니다.
GetCharABCWidthsA GetCharABCWidths 함수는 현재 TrueType 글꼴에서 지정된 범위의 연속 문자 너비를 논리 단위로 검색합니다. 이 함수는 TrueType 글꼴로만 성공합니다. (ANSI)
GetCharABCWidthsFloatA GetCharABCWidthsFloat 함수는 현재 글꼴에서 지정된 범위의 연속 문자의 너비를 논리 단위로 검색합니다. (ANSI)
GetCharABCWidthsFloatW GetCharABCWidthsFloat 함수는 현재 글꼴에서 지정된 범위의 연속 문자의 너비를 논리 단위로 검색합니다. (유니코드)

GetCharABCWidthsI GetCharABCWidthsI 함수는 현재 TrueType 글꼴에서 지정된 범위의 연속 문자 모양 인덱스의 너비를 논리 단위로 검색합니다. 이 함수는 TrueType 글꼴로만 성공합니다.
GetCharABCWidthsW GetCharABCWidths 함수는 현재 TrueType 글꼴에서 지정된 범위의 연속 문자 너비를 논리 단위로 검색합니다. 이 함수는 TrueType 글꼴로만 성공합니다. (유니코드)
GetCharacterPlacementA GetCharacterPlacement 함수는 문자 너비, 캐리트 위치 지정, 문자열 내 순서 지정 및 문자 모양 렌더링과 같은 문자열에 대한 정보를 검색합니다. (ANSI)
GetCharacterPlacementW GetCharacterPlacement 함수는 문자 너비, 캐리트 위치 지정, 문자열 내 순서 지정 및 문자 모양 렌더링과 같은 문자열에 대한 정보를 검색합니다. (유니코드)
GetCharWidth32A GetCharWidth32 함수는 현재 글꼴에서 지정된 범위의 연속 문자의 너비를 논리적 좌표로 검색합니다. (ANSI)
GetCharWidth32W GetCharWidth32 함수는 현재 글꼴에서 지정된 범위의 연속 문자의 너비를 논리적 좌표로 검색합니다. (유니코드)
GetCharWidthA GetCharWidth 함수는 현재 글꼴에서 지정된 범위의 연속 문자의 너비를 논리적 좌표로 검색합니다. (ANSI)
GetCharWidthFloatA GetCharWidthFloat 함수는 현재 글꼴에서 지정된 범위에서 연속된 문자의 소수 자릿수 너비를 검색합니다. (ANSI)
GetCharWidthFloatW GetCharWidthFloat 함수는 현재 글꼴에서 지정된 범위에서 연속된 문자의 소수 자릿수 너비를 검색합니다. (유니코드)
GetCharWidthI GetCharWidthI 함수는 현재 글꼴에서 지정된 범위의 연속 문자 모양 인덱스의 너비를 논리적 좌표로 검색합니다.

GetCharWidthW GetCharWidth 함수는 현재 글꼴에서 지정된 범위의 연속 문자의 너비를 논리적 좌표로 검색합니다. (유니코드)
GetClipBox GetClipBox 함수는 디바이스에서 현재 표시되는 영역 주위에 그릴 수 있는 가장 좁은 경계 사각형의 크기를 검색합니다.
GetClipRgn GetClipRgn 함수는 지정된 디바이스 컨텍스트에 대한 현재 애플리케이션 정의 클리핑 영역을 식별하는 핸들을 검색합니다.
GetColorAdjustment GetColorAdjustment 함수는 지정된 DC(디바이스 컨텍스트)에 대한 색 조정 값을 검색합니다.
GetColorSpace GetColorSpace 함수는 지정된 디바이스 컨텍스트에서 입력 색 공간에 대한 핸들을 검색합니다.
GetCurrentObject GetCurrentObject 함수는 지정된 DC(디바이스 컨텍스트)로 선택된 지정된 형식의 개체에 대한 핸들을 검색합니다.
GetCurrentPositionEx GetCurrentPositionEx 함수는 논리 좌표에서 현재 위치를 검색합니다.
GetCValue GetCValue 매크로는 CMYK 색 값에서 시안 색 값을 검색합니다.
GetDCBrushColor GetDCBrushColor 함수는 지정된 DC(디바이스 컨텍스트)에 대한 현재 브러시 색을 검색합니다.
GetDCOrgEx GetDCOrgEx 함수는 지정된 DC(디바이스 컨텍스트)에 대한 최종 번역 원본을 검색합니다.
GetDCPenColor GetDCPenColor 함수는 지정된 DC(디바이스 컨텍스트)에 대한 현재 펜 색을 검색합니다.

GetDeviceCaps GetDeviceCaps 함수는 지정된 디바이스에 대한 디바이스별 정보를 검색합니다.
GetDeviceGammaRamp GetDeviceGammaRamp 함수는 하드웨어에서 다운로드 가능한 감마 램프를 지원하는 드라이버가 있는 직접 색 디스플레이 보드의 감마 램프를 가져옵니다.
GetDIBColorTable GetDIBColorTable 함수는 현재 지정된 디바이스 컨텍스트로 선택된 DIB 섹션 비트맵의 색 테이블에 있는 항목 범위에서 RGB(빨강, 녹색, 파랑) 색 값을 검색합니다.
GetDIBits GetDIBits 함수는 지정된 호환 비트맵의 비트를 검색하고 지정된 형식을 사용하여 DIB로 버퍼에 복사합니다.
GetEnhMetaFileA GetEnhMetaFile 함수는 지정된 파일에 저장된 고급 형식 메타파일을 식별하는 핸들을 만듭니다. (ANSI)
GetEnhMetaFileBits GetEnhMetaFileBits 함수는 지정된 고급 형식 메타파일의 콘텐츠를 검색하여 버퍼에 복사합니다.
GetEnhMetaFileDescriptionA GetEnhMetaFileDescription 함수는 고급 형식 메타파일에서 선택적 텍스트 설명을 검색하고 지정된 버퍼에 문자열을 복사합니다. (ANSI)
GetEnhMetaFileDescriptionW GetEnhMetaFileDescription 함수는 고급 형식 메타파일에서 선택적 텍스트 설명을 검색하고 지정된 버퍼에 문자열을 복사합니다. (유니코드)
GetEnhMetaFileHeader GetEnhMetaFileHeader 함수는 지정된 고급 형식 메타파일의 헤더가 포함된 레코드를 검색합니다.
GetEnhMetaFilePaletteEntries GetEnhMetaFilePaletteEntries 함수는 지정된 향상된 메타파일에서 선택적 팔레트 항목을 검색합니다.

GetEnhMetaFilePixelFormat GetEnhMetaFilePixelFormat 함수는 향상된 메타파일의 픽셀 형식 정보를 검색합니다.
GetEnhMetaFileW GetEnhMetaFile 함수는 지정된 파일에 저장된 고급 형식 메타파일을 식별하는 핸들을 만듭니다. (유니코드)
GetFontData GetFontData 함수는 TrueType 글꼴에 대한 글꼴 메트릭 데이터를 검색합니다.
GetFontLanguageInfo GetFontLanguageInfo 함수는 지정된 표시 컨텍스트에 대해 현재 선택된 글꼴에 대한 정보를 반환합니다. 애플리케이션은 일반적으로 이 정보와 GetCharacterPlacement 함수를 사용하여 표시할 문자열을 준비합니다.
GetFontUnicodeRanges GetFontUnicodeRanges 함수는 글꼴에서 지원되는 유니코드 문자에 대한 정보를 반환합니다. 정보는 GLYPHSET 구조체로 반환됩니다.
GetGlyphIndicesA GetGlyphIndices 함수는 문자열을 문자 모양 인덱스의 배열로 변환합니다. 이 함수를 사용하여 글꼴에 문자 모양이 있는지 여부를 확인할 수 있습니다. (ANSI)
GetGlyphIndicesW GetGlyphIndices 함수는 문자열을 문자 모양 인덱스의 배열로 변환합니다. 이 함수를 사용하여 글꼴에 문자 모양이 있는지 여부를 확인할 수 있습니다. (유니코드)
GetGlyphOutlineA GetGlyphOutline 함수는 TrueType 글꼴에서 지정된 디바이스 컨텍스트로 선택된 문자의 윤곽선 또는 비트맵을 검색합니다. (ANSI)
GetGlyphOutlineW GetGlyphOutline 함수는 TrueType 글꼴에서 지정된 디바이스 컨텍스트로 선택된 문자의 윤곽선 또는 비트맵을 검색합니다. (유니코드)
GetGraphicsMode GetGraphicsMode 함수는 지정된 디바이스 컨텍스트에 대한 현재 그래픽 모드를 검색합니다.

GetGValue GetGValue 매크로는 빨강, 녹색, 파랑(RGB) 값의 녹색 구성 요소에 대한 강도 값을 검색합니다.
GetICMProfileA GetICMProfile 함수는 지정된 디바이스 컨텍스트에 대한 현재 출력 색 프로필의 파일 이름을 검색합니다. (ANSI)
GetICMProfileW GetICMProfile 함수는 지정된 디바이스 컨텍스트에 대한 현재 출력 색 프로필의 파일 이름을 검색합니다. (유니코드)
GetKerningPairsA GetKerningPairs 함수는 지정된 디바이스 컨텍스트에 대해 현재 선택된 글꼴에 대한 문자 커닝 쌍을 검색합니다. (ANSI)
GetKerningPairsW GetKerningPairs 함수는 지정된 디바이스 컨텍스트에 대해 현재 선택된 글꼴에 대한 문자 커닝 쌍을 검색합니다. (유니코드)
GetKValue GetKValue 매크로는 CMYK 색 값에서 검은색 값을 검색합니다.
GetLayout GetLayout 함수는 DC(디바이스 컨텍스트)의 레이아웃을 반환합니다.
GetLogColorSpaceA GetLogColorSpace 함수는 지정된 핸들로 식별된 색 공간 정의를 검색합니다. (ANSI)
GetLogColorSpaceW GetLogColorSpace 함수는 지정된 핸들로 식별된 색 공간 정의를 검색합니다. (유니코드)
GetMapMode GetMapMode 함수는 현재 매핑 모드를 검색합니다.
GetMetaFileA GetMetaFile 함수는 지정된 파일에 저장된 메타파일을 식별하는 핸들을 만듭니다. (ANSI)

GetMetaFileBitsEx GetMetaFileBitsEx 함수는 Windows 형식 메타파일의 내용을 검색하고 지정된 버퍼에 복사합니다.
GetMetaFileW GetMetaFile 함수는 지정된 파일에 저장된 메타파일을 식별하는 핸들을 만듭니다. (유니코드)
GetMetaRgn GetMetaRgn 함수는 지정된 디바이스 컨텍스트에 대한 현재 메타레지온을 검색합니다.
GetMiterLimit GetMiterLimit 함수는 지정된 디바이스 컨텍스트에 대한 miter 제한을 검색합니다.
GetMValue GetMValue 매크로는 CMYK 색 값에서 magenta 색 값을 검색합니다.
GetNearestColor GetNearestColor 함수는 지정된 색 값을 사용할 때 표시될 시스템 색상표에서 색을 식별하는 색 값을 검색합니다.
GetNearestPaletteIndex GetNearestPaletteIndex 함수는 지정된 색 값과 가장 일치하는 지정된 논리 팔레트의 항목에 대한 인덱스를 검색합니다.
GetObject GetObject 함수(wingdi.h)는 지정된 그래픽 개체에 대한 정보를 검색합니다.
GetObjectA GetObject 함수는 지정된 그래픽 개체에 대한 정보를 검색합니다. (GetObjectA)
GetObjectType GetObjectType은 지정된 개체의 형식을 검색합니다.
GetObjectW GetObjectW(유니코드) 함수(wingdi.h)는 지정된 그래픽 개체에 대한 정보를 검색합니다.
GetOutlineTextMetricsA GetOutlineTextMetrics 함수는 TrueType 글꼴에 대한 텍스트 메트릭을 검색합니다. (ANSI)

GetOutlineTextMetricsW GetOutlineTextMetrics 함수는 TrueType 글꼴에 대한 텍스트 메트릭을 검색합니다. (유니코드)
GetPaletteEntries GetPaletteEntries 함수는 지정된 논리 팔레트에서 지정된 색상표 항목 범위를 검색합니다.
GetPath GetPath 함수는 지정된 디바이스 컨텍스트로 선택된 경로에 있는 곡선의 제어점 및 선의 엔드포인트를 정의하는 좌표를 검색합니다.
GetPixel GetPixel 함수는 지정된 좌표에서 픽셀의 빨강, 녹색, 파랑(RGB) 색 값을 검색합니다.
GetPixelFormat GetPixelFormat 함수는 지정된 디바이스 컨텍스트의 현재 선택된 픽셀 형식의 인덱스를 가져옵니다.
GetPolyFillMode GetPolyFillMode 함수는 현재 다각형 채우기 모드를 검색합니다.
GetRandomRgn GetRandomRgn 함수는 지정된 디바이스 컨텍스트의 시스템 클리핑 영역을 특정 지역에 복사합니다.
GetRasterizerCaps GetRasterizerCaps 함수는 트루타입 글꼴이 시스템에 설치되어 있는지 여부를 나타내는 플래그를 반환합니다.
GetRegionData GetRegionData 함수는 지정된 버퍼를 지역을 설명하는 데이터로 채웁니다. 이 데이터에는 영역을 구성하는 사각형의 크기가 포함됩니다.
GetRgnBox GetRgnBox 함수는 지정된 지역의 경계 사각형을 검색합니다.
GetROP2 GetROP2 함수는 지정된 디바이스 컨텍스트의 포그라운드 혼합 모드를 검색합니다. 혼합 모드는 펜 또는 내부 색과 이미 화면에 있는 색을 결합하여 새 색을 만드는 방법을 지정합니다.

GetRValue GetRValue 매크로는 빨강, 녹색, 파랑(RGB) 값의 빨간색 구성 요소에 대한 강도 값을 검색합니다.
GetStockObject GetStockObject 함수는 스톡 펜, 브러시, 글꼴 또는 색상표 중 하나에 대한 핸들을 검색합니다.
GetStretchBltMode GetStretchBltMode 함수는 현재 스트레치 모드를 검색합니다. 스트레치 모드는 StretchBlt 함수가 호출될 때 늘어나거나 압축되는 비트맵에서 색 데이터를 추가하거나 제거하는 방법을 정의합니다.
GetSystemPaletteEntries GetSystemPaletteEntries 함수는 지정된 DC(디바이스 컨텍스트)와 연결된 시스템 팔레트에서 색상표 항목 범위를 검색합니다.
GetSystemPaletteUse GetSystemPaletteUse 함수는 지정된 DC(디바이스 컨텍스트)에 대한 시스템(물리적) 팔레트의 현재 상태를 검색합니다.
GetTextAlign GetTextAlign 함수는 지정된 디바이스 컨텍스트에 대한 텍스트 맞춤 설정을 검색합니다.
GetTextCharacterExtra GetTextCharacterExtra 함수는 지정된 디바이스 컨텍스트에 대한 현재 문자 간 간격을 검색합니다.
GetTextCharset 현재 지정된 디바이스 컨텍스트로 선택된 글꼴의 문자 집합 식별자를 검색합니다.
GetTextCharsetInfo 현재 지정된 디바이스 컨텍스트로 선택된 글꼴의 문자 집합에 대한 정보를 검색합니다.
GetTextColor GetTextColor 함수는 지정된 디바이스 컨텍스트에 대한 현재 텍스트 색을 검색합니다.
GetTextExtentExPointA GetTextExtentExPoint 함수는 지정된 공간에 맞는 지정된 문자열의 문자 수를 검색하고 각 문자에 대한 텍스트 범위로 배열을 채웁니다. (ANSI)

GetTextExtentExPointI GetTextExtentExPointI 함수는 지정된 공간에 맞는 지정된 문자열의 문자 수를 검색하고 각 문자에 대한 텍스트 범위로 배열을 채웁니다.
GetTextExtentExPointW GetTextExtentExPoint 함수는 지정된 공간에 맞는 지정된 문자열의 문자 수를 검색하고 각 문자에 대한 텍스트 범위로 배열을 채웁니다. (유니코드)
GetTextExtentPoint32A GetTextExtentPoint32 함수는 지정된 텍스트 문자열의 너비와 높이를 계산합니다. (ANSI)
GetTextExtentPoint32W GetTextExtentPoint32 함수는 지정된 텍스트 문자열의 너비와 높이를 계산합니다. (유니코드)
GetTextExtentPointA GetTextExtentPoint 함수는 지정된 텍스트 문자열의 너비와 높이를 계산합니다. (ANSI)
GetTextExtentPointI GetTextExtentPointI 함수는 지정된 문자 모양 인덱스 배열의 너비와 높이를 계산합니다.
GetTextExtentPointW GetTextExtentPoint 함수는 지정된 텍스트 문자열의 너비와 높이를 계산합니다. (유니코드)
GetTextFaceA GetTextFace 함수는 선택한 글꼴의 서체 이름을 지정된 디바이스 컨텍스트로 검색합니다. (ANSI)
GetTextFaceW GetTextFace 함수는 선택한 글꼴의 서체 이름을 지정된 디바이스 컨텍스트로 검색합니다. (유니코드)
GetTextMetrics GetTextMetrics 함수(wingdi.h)는 지정된 버퍼를 현재 선택한 글꼴에 대한 메트릭으로 채웁니다.
GetTextMetricsA GetTextMetrics 함수는 지정된 버퍼를 현재 선택한 글꼴에 대한 메트릭으로 채웁니다. (GetTextMetricsA)

GetTextMetricsW GetTextMetricsW(유니코드) 함수(wingdi.h)는 지정된 버퍼를 현재 선택한 글꼴에 대한 메트릭으로 채웁니다.
GetViewportExtEx GetViewportExtEx 함수는 지정된 디바이스 컨텍스트에 대한 현재 뷰포트의 x 익스텐트 및 y 익스텐트를 검색합니다.
GetViewportOrgEx GetViewportOrgEx 함수는 지정된 디바이스 컨텍스트에 대한 뷰포트 원본의 x 좌표 및 y 좌표를 검색합니다.
GetWindowExtEx 이 함수는 지정된 디바이스 컨텍스트에 대한 창의 x 익스텐트 및 y 익스텐트 검색합니다.
GetWindowOrgEx GetWindowOrgEx 함수는 지정된 디바이스 컨텍스트에 대한 창 원본의 x 좌표 및 y 좌표를 검색합니다.
GetWinMetaFileBits GetWinMetaFileBits 함수는 메타파일의 향상된 형식 레코드를 Windows 형식 레코드로 변환하고 변환된 레코드를 지정된 버퍼에 저장합니다.
GetWorldTransform GetWorldTransform 함수는 현재 세계 공간을 페이지 공간 변환으로 검색합니다.
GetYValue GetYValue 매크로는 CMYK 색 값에서 노란색 색 값을 검색합니다.
GradientFill GradientFill 함수는 사각형과 삼각형 구조를 채웁니다.
IntersectClipRect IntersectClipRect 함수는 현재 클리핑 영역과 지정된 사각형의 교차점에서 새 클리핑 영역을 만듭니다.
InvertRgn InvertRgn 함수는 지정된 영역의 색을 반전합니다.

LineDDA LineDDA 함수는 지정된 시작점과 끝점으로 정의된 선에 대해 강조 표시해야 하는 픽셀을 결정합니다.
Lineto LineTo 함수는 현재 위치에서 지정된 지점까지 선을 그립니다.
LPtoDP LPtoDP 함수는 논리 좌표를 디바이스 좌표로 변환합니다. 변환은 디바이스 컨텍스트의 매핑 모드, 창 및 뷰포트의 원본 및 익스텐트 설정, 월드 변환에 따라 달라집니다.
메이크포인트 MAKEPOINTS 매크로는 점의 x 좌표와 y 좌표를 포함하는 값을 POINTS 구조체로 변환합니다.
MAKEROP4 MAKEROP4 매크로는 MaskBlt 함수와 함께 사용할 쿼터너리 래스터 작업 코드를 만듭니다.
MaskBlt MaskBlt 함수는 지정된 마스크 및 래스터 작업을 사용하여 원본 및 대상 비트맵의 색 데이터를 결합합니다.
ModifyWorldTransform ModifyWorldTransform 함수는 지정된 모드를 사용하여 디바이스 컨텍스트에 대한 월드 변환을 변경합니다.
MoveToEx MoveToEx 함수는 현재 위치를 지정된 지점으로 업데이트하고 필요에 따라 이전 위치를 반환합니다.
OffsetClipRgn OffsetClipRgn 함수는 디바이스 컨텍스트의 클리핑 영역을 지정된 오프셋으로 이동합니다.
OffsetRgn OffsetRgn 함수는 지정된 오프셋으로 영역을 이동합니다.
OffsetViewportOrgEx OffsetViewportOrgEx 함수는 지정된 가로 및 세로 오프셋을 사용하여 디바이스 컨텍스트에 대한 뷰포트 원본을 수정합니다.

OffsetWindowOrgEx OffsetWindowOrgEx 함수는 지정된 가로 및 세로 오프셋을 사용하여 디바이스 컨텍스트의 창 원본을 수정합니다.
PaintRgn PaintRgn 함수는 디바이스 컨텍스트에 현재 선택된 브러시를 사용하여 지정된 영역을 그립니다.
PALETTEINDEX PALETTEINDEX 매크로는 논리 색 팔레트 항목에 대한 인덱스를 허용하고 지정된 인덱스와 연결된 색을 지정하는 COLORREF 값으로 구성된 색상표 항목 지정자를 반환합니다.
PALETTERGB PALETTERGB 매크로는 빨강, 녹색 및 파랑의 상대적 강도를 나타내는 세 가지 값을 허용하며 상위 바이트에서 2로 구성된 색상표 상대 빨강, 녹색, 파랑(RGB) 지정자 및 3개의 낮은 순서 바이트에서 RGB 값을 반환합니다. 색상표를 사용하는 애플리케이션은 명시적 RGB 값 대신 이 지정자를 색을 예상하는 함수에 전달할 수 있습니다.
PatBlt PatBlt 함수는 지정된 디바이스 컨텍스트에 현재 선택된 브러시를 사용하여 지정된 사각형을 그립니다. 브러시 색과 표면 색 또는 색은 지정된 래스터 연산을 사용하여 결합됩니다.
PathToRegion PathToRegion 함수는 선택한 경로에서 지정된 디바이스 컨텍스트로 영역을 만듭니다. 결과 지역에서는 디바이스 좌표를 사용합니다.
Pie 원형 함수는 타원과 두 방사형의 교차점에 의해 경계가 있는 원형 모양의 웨지를 그립니다. 원형은 현재 펜을 사용하여 윤곽선으로 표시되고 현재 브러시를 사용하여 채워집니다.
PlayEnhMetaFile PlayEnhMetaFile 함수는 지정된 고급 형식 메타파일로 저장된 그림을 표시합니다.
PlayEnhMetaFileRecord PlayEnhMetaFileRecord 함수는 레코드로 식별된 GDI(그래픽 장치 인터페이스) 함수를 실행하여 향상된 메타파일 레코드를 재생합니다.
PlayMetaFile PlayMetaFile 함수는 지정된 디바이스에서 지정된 Windows 형식 메타파일에 저장된 그림을 표시합니다.

PlayMetaFileRecord PlayMetaFileRecord 함수는 해당 레코드에 포함된 GDI(그래픽 디바이스 인터페이스) 함수를 실행하여 Windows 형식 메타파일 레코드를 재생합니다.
PlgBlt PlgBlt 함수는 원본 디바이스 컨텍스트의 지정된 사각형에서 대상 디바이스 컨텍스트의 지정된 parallelogram으로 색 데이터 비트를 비트 블록 전송합니다.
PolyBezier PolyBezier 함수는 하나 이상의 B 지어 곡선을 그립니다.
PolyBezierTo PolyBezierTo 함수는 하나 이상의 B 지어 곡선을 그립니다.
PolyDraw PolyDraw 함수는 선 세그먼트 및 B 지어 곡선 집합을 그립니다.
Polygon 다각형 함수는 직선으로 연결된 둘 이상의 꼭짓점으로 구성된 다각형을 그립니다. 다각형은 현재 펜을 사용하여 윤곽선으로 표시되고 현재 브러시 및 다각형 채우기 모드를 사용하여 채워집니다.
폴리라인 Polyline 함수는 지정된 배열의 점을 연결하여 일련의 선 세그먼트를 그립니다.
PolylineTo PolylineTo 함수는 하나 이상의 직선을 그립니다.
PolyPolygon PolyPolygon 함수는 일련의 닫힌 다각형을 그립니다. 각 다각형은 현재 펜을 사용하여 윤곽선으로 표시되고 현재 브러시 및 다각형 채우기 모드를 사용하여 채워집니다. 이 함수에서 그린 다각형은 겹칠 수 있습니다.
PolyPolyline PolyPolyline 함수는 여러 일련의 연결된 선 세그먼트를 그립니다.
PolyTextOutA PolyTextOut 함수는 지정된 디바이스 컨텍스트에서 현재 선택된 글꼴 및 텍스트 색을 사용하여 여러 문자열을 그립니다. (ANSI)

PolyTextOutW PolyTextOut 함수는 지정된 디바이스 컨텍스트에서 현재 선택된 글꼴 및 텍스트 색을 사용하여 여러 문자열을 그립니다. (유니코드)
PtInRegion PtInRegion 함수는 지정된 지점이 지정된 지역 내에 있는지 여부를 결정합니다.
PtVisible PtVisible 함수는 지정된 지점이 디바이스 컨텍스트의 클리핑 영역 내에 있는지 여부를 결정합니다.
RealizePalette RealizePalette 함수는 현재 논리 팔레트의 색상표 항목을 시스템 팔레트에 매핑합니다.
사각형 사각형 함수는 사각형을 그립니다. 사각형은 현재 펜을 사용하여 윤곽선으로 표시되고 현재 브러시를 사용하여 채워집니다.
RectInRegion RectInRegion 함수는 지정된 사각형의 일부가 영역 경계 내에 있는지 여부를 결정합니다.
RectVisible RectVisible 함수는 지정된 사각형의 일부가 디바이스 컨텍스트의 클리핑 영역 내에 있는지 여부를 결정합니다.
RemoveFontMemResourceEx RemoveFontMemResourceEx 함수는 메모리 이미지 파일에서 추가된 글꼴을 제거합니다.
RemoveFontResourceA RemoveFontResource 함수는 시스템 글꼴 테이블에서 지정된 파일의 글꼴을 제거합니다. (ANSI)
RemoveFontResourceExA RemoveFontResourceEx 함수는 시스템 글꼴 테이블에서 지정된 파일의 글꼴을 제거합니다. (ANSI)
RemoveFontResourceExW RemoveFontResourceEx 함수는 시스템 글꼴 테이블에서 지정된 파일의 글꼴을 제거합니다. (유니코드)

RemoveFontResourceW RemoveFontResource 함수는 시스템 글꼴 테이블에서 지정된 파일의 글꼴을 제거합니다. (유니코드)
ResetDCA ResetDC 함수는 지정된 정보를 사용하여 지정된 프린터 또는 플로터 DC(디바이스 컨텍스트)를 업데이트합니다. (ANSI)
ResetDCW ResetDC 함수는 지정된 정보를 사용하여 지정된 프린터 또는 플로터 DC(디바이스 컨텍스트)를 업데이트합니다. (유니코드)
ResizePalette ResizePalette 함수는 지정된 값에 따라 논리 팔레트의 크기를 늘리거나 줄입니다.
RestoreDC RestoreDC 함수는 DC(디바이스 컨텍스트)를 지정된 상태로 복원합니다. DC는 SaveDC 함수에 대한 이전 호출로 만든 스택에서 상태 정보를 표시하여 복원됩니다.
RGB RGB 매크로는 제공된 인수 및 출력 디바이스의 색 기능에 따라 빨강, 녹색, 파랑(RGB) 색을 선택합니다.
RoundRect RoundRect 함수는 모서리가 둥근 사각형을 그립니다. 사각형은 현재 펜을 사용하여 윤곽선으로 표시되고 현재 브러시를 사용하여 채워집니다.
SaveDC SaveDC 함수는 선택한 개체 및 그래픽 모드(예: 비트맵, 브러시, 팔레트, 글꼴, 펜, 영역, 그리기 모드 및 매핑 모드)를 설명하는 데이터를 컨텍스트 스택에 복사하여 지정된 DC(디바이스 컨텍스트)의 현재 상태를 저장합니다.
ScaleViewportExtEx ScaleViewportExtEx 함수는 지정된 곱셈 및 제수에 의해 형성된 비율을 사용하여 디바이스 컨텍스트에 대한 뷰포트를 수정합니다.
ScaleWindowExtEx ScaleWindowExtEx 함수는 지정된 곱셈 및 제수에 의해 형성된 비율을 사용하여 디바이스 컨텍스트의 창을 수정합니다.

SelectClipPath SelectClipPath 함수는 현재 경로를 디바이스 컨텍스트의 클리핑 영역으로 선택하고 지정된 모드를 사용하여 새 영역을 기존 클리핑 영역과 결합합니다.
SelectClipRgn SelectClipRgn 함수는 지정된 디바이스 컨텍스트에 대한 현재 클리핑 영역으로 영역을 선택합니다.
Selectobject SelectObject 함수는 지정된 DC(디바이스 컨텍스트)로 개체를 선택합니다. 새 개체는 동일한 형식의 이전 개체를 대체합니다.
SelectPalette SelectPalette 함수는 지정된 논리 팔레트를 디바이스 컨텍스트로 선택합니다.
SetAbortProc SetAbortProc 함수는 스풀링 중에 인쇄 작업을 취소할 수 있는 애플리케이션 정의 중단 함수를 설정합니다.
SetArcDirection SetArcDirection은 원호 및 사각형 함수에 사용할 그리기 방향을 설정합니다.
SetBitmapBits SetBitmapBits 함수는 비트맵의 색 데이터 비트를 지정된 값으로 설정합니다.
SetBitmapDimensionEx SetBitmapDimensionEx 함수는 비트맵에 기본 차원을 할당합니다. 이러한 차원은 애플리케이션에서 사용할 수 있습니다. 그러나 시스템에서는 사용되지 않습니다.
SetBkColor SetBkColor 함수는 현재 배경색을 지정된 색 값으로 설정하거나 디바이스가 지정된 색 값을 나타낼 수 없는 경우 가장 가까운 실제 색으로 설정합니다.
SetBkMode SetBkMode 함수는 지정된 디바이스 컨텍스트의 배경 혼합 모드를 설정합니다. 배경 혼합 모드는 실선이 아닌 텍스트, 빗살 무늬 브러시 및 펜 스타일과 함께 사용됩니다.
SetBoundsRect SetBoundsRect 함수는 지정된 디바이스 컨텍스트에 대한 경계 사각형 정보의 누적을 제어합니다.

SetBrushOrgEx SetBrushOrgEx 함수는 GDI가 지정한 디바이스 컨텍스트에 애플리케이션이 선택하는 다음 브러시에 할당하는 브러시 원본을 설정합니다.
SetColorAdjustment SetColorAdjustment 함수는 지정된 값을 사용하여 DC(디바이스 컨텍스트)의 색 조정 값을 설정합니다.
SetColorSpace SetColorSpace 함수는 지정된 디바이스 컨텍스트에 대한 입력 색 공간을 정의합니다.
SetDCBrushColor SetDCBrushColor 함수는 현재 DC(디바이스 컨텍스트) 브러시 색을 지정된 색 값으로 설정합니다. 디바이스에서 지정된 색 값을 나타낼 수 없는 경우 색은 가장 가까운 실제 색으로 설정됩니다.
SetDCPenColor SetDCPenColor 함수는 현재 DC(디바이스 컨텍스트) 펜 색을 지정된 색 값으로 설정합니다. 디바이스에서 지정된 색 값을 나타낼 수 없는 경우 색은 가장 가까운 실제 색으로 설정됩니다.
SetDeviceGammaRamp SetDeviceGammaRamp 함수는 하드웨어에서 다운로드 가능한 감마 램프를 지원하는 드라이버가 있는 직접 색 디스플레이 보드의 감마 램프를 설정합니다.
SetDIBColorTable SetDIBColorTable 함수는 현재 지정된 디바이스 컨텍스트로 선택된 DIB의 색 테이블에 있는 항목 범위에서 RGB(빨강, 녹색, 파랑) 색 값을 설정합니다.
SetDIBits SetDIBits 함수는 지정된 DIB에 있는 색 데이터를 사용하여 호환되는 비트맵(DDB)의 픽셀을 설정합니다.
SetDIBitsToDevice SetDIBitsToDevice 함수는 DIB, JPEG 또는 PNG 이미지의 색 데이터를 사용하여 대상 디바이스 컨텍스트와 연결된 디바이스의 지정된 사각형에 있는 픽셀을 설정합니다.
SetEnhMetaFileBits SetEnhMetaFileBits 함수는 지정된 데이터에서 메모리 기반 고급 형식 메타파일을 만듭니다.

SetGraphicsMode SetGraphicsMode 함수는 지정된 디바이스 컨텍스트에 대한 그래픽 모드를 설정합니다.
SetICMMode SetICMMode 함수를 사용하면 지정된 DC(디바이스 컨텍스트)에서 이미지 색 관리를 사용하거나 사용하지 않도록 설정하거나 쿼리할 수 있습니다.
SetICMProfileA SetICMProfile 함수는 지정된 색 프로필을 지정된 DC(디바이스 컨텍스트)의 출력 프로필로 설정합니다. (ANSI)
SetICMProfileW SetICMProfile 함수는 지정된 색 프로필을 지정된 DC(디바이스 컨텍스트)의 출력 프로필로 설정합니다. (유니코드)
SetLayout SetLayout 함수는 DC(디바이스 컨텍스트)의 레이아웃을 변경합니다.
SetMapMode SetMapMode 함수는 지정된 디바이스 컨텍스트의 매핑 모드를 설정합니다. 매핑 모드는 페이지 공간 단위를 디바이스 공간 단위로 변환하는 데 사용되는 측정 단위를 정의하고 디바이스의 x축과 y축의 방향도 정의합니다.
SetMapperFlags SetMapperFlags 함수는 논리 글꼴을 실제 글꼴에 매핑할 때 글꼴 매퍼가 사용하는 알고리즘을 변경합니다.
SetMetaFileBitsEx SetMetaFileBitsEx 함수는 제공된 데이터에서 메모리 기반 Windows 형식 메타파일을 만듭니다.
SetMetaRgn SetMetaRgn 함수는 지정된 디바이스 컨텍스트의 현재 클리핑 영역을 현재 메타레지온과 교차하고 결합된 지역을 지정된 디바이스 컨텍스트에 대한 새 메타레지온으로 저장합니다.
SetMiterLimit SetMiterLimit 함수는 지정된 디바이스 컨텍스트에 대한 miter 조인 길이 제한을 설정합니다.

SetPaletteEntries SetPaletteEntries 함수는 RGB(빨강, 녹색, 파랑) 색 값 및 플래그를 논리 팔레트의 항목 범위로 설정합니다.
Setpixel SetPixel 함수는 지정된 좌표의 픽셀을 지정된 색으로 설정합니다.
SetPixelFormat SetPixelFormat 함수는 지정된 디바이스 컨텍스트의 픽셀 형식을 iPixelFormat 인덱스에 지정된 형식으로 설정합니다.
SetPixelV SetPixelV 함수는 지정된 좌표의 픽셀을 지정된 색의 가장 가까운 근사값으로 설정합니다. 지점은 클리핑 영역과 디바이스 표면의 표시되는 부분에 있어야 합니다.
SetPolyFillMode SetPolyFillMode 함수는 다각형을 채우는 함수의 다각형 채우기 모드를 설정합니다.
SetRectRgn SetRectRgn 함수는 지정된 좌표를 사용하여 영역을 사각형 영역으로 변환합니다.
SetROP2 SetROP2 함수는 현재 전경 혼합 모드를 설정합니다.
SetStretchBltMode SetStretchBltMode 함수는 지정된 디바이스 컨텍스트에서 비트맵 스트레치 모드를 설정합니다.
SetSystemPaletteUse SetSystemPaletteUse 함수를 사용하면 애플리케이션에서 시스템 팔레트에 2 또는 20개의 정적 색이 포함되어 있는지 여부를 지정할 수 있습니다.
SetTextAlign SetTextAlign 함수는 지정된 디바이스 컨텍스트에 대한 텍스트 맞춤 플래그를 설정합니다.
SetTextCharacterExtra SetTextCharacterExtra 함수는 문자 간 간격을 설정합니다. 시스템에서 텍스트 줄을 작성할 때 구분 문자를 포함하여 문자 간 간격이 각 문자에 추가됩니다.

SetTextColor SetTextColor 함수는 지정된 디바이스 컨텍스트의 텍스트 색을 지정된 색으로 설정합니다.
SetTextJustification SetTextJustification 함수는 시스템에서 텍스트 문자열의 나누기 문자에 추가해야 하는 공간의 크기를 지정합니다. 이 공간은 애플리케이션이 TextOut 또는 ExtTextOut 함수를 호출할 때 추가됩니다.
SetViewportExtEx 지정된 값을 사용하여 디바이스 컨텍스트에 대한 뷰포트의 가로 및 세로 범위를 설정합니다.
SetViewportOrgEx SetViewportOrgEx 함수는 창 원본(0,0)에 매핑되는 디바이스 지점을 지정합니다.
SetWindowExtEx SetWindowExtEx 함수는 지정된 값을 사용하여 디바이스 컨텍스트에 대한 창의 가로 및 세로 익스텐트를 설정합니다.
SetWindowOrgEx SetWindowOrgEx 함수는 뷰포트 원점(0,0)에 매핑되는 창 지점을 지정합니다.
SetWinMetaFileBits SetWinMetaFileBits 함수는 메타파일을 이전 Windows 형식에서 새로운 향상된 형식으로 변환하고 새 메타파일을 메모리에 저장합니다.
SetWorldTransform SetWorldTransform 함수는 지정된 디바이스 컨텍스트에 대한 세계 공간과 페이지 공간 간의 2차원 선형 변환을 설정합니다. 이 변환은 그래픽 출력의 크기를 조정, 회전, 전단 또는 변환하는 데 사용할 수 있습니다.
StartDocA StartDoc 함수는 인쇄 작업을 시작합니다. (ANSI)
StartDocW StartDoc 함수는 인쇄 작업을 시작합니다. (유니코드)
StartPage StartPage 함수는 데이터를 허용하도록 프린터 드라이버를 준비합니다.

Stretchblt StretchBlt 함수는 필요한 경우 원본 사각형에서 대상 사각형의 크기에 맞게 비트맵을 늘이거나 압축하여 대상 사각형으로 비트맵을 복사합니다.
StretchDIBits StretchDIBits 함수는 DIB, JPEG 또는 PNG 이미지의 픽셀 사각형에 대한 색 데이터를 지정된 대상 사각형에 복사합니다.
StrokeAndFillPath StrokeAndFillPath 함수는 경로의 열려 있는 그림을 닫고, 현재 펜을 사용하여 경로의 윤곽선을 스트로크하고, 현재 브러시를 사용하여 내부를 채웁니다.
StrokePath StrokePath 함수는 현재 펜을 사용하여 지정된 경로를 렌더링합니다.
SwapBuffers 지정한 디바이스 컨텍스트에서 참조하는 창의 현재 픽셀 형식에 백 버퍼가 포함된 경우 SwapBuffers 함수는 전면 및 후면 버퍼를 교환합니다.
TextOutA TextOut 함수는 현재 선택한 글꼴, 배경색 및 텍스트 색을 사용하여 지정된 위치에 문자열을 씁니다. (ANSI)
TextOutW TextOut 함수는 현재 선택한 글꼴, 배경색 및 텍스트 색을 사용하여 지정된 위치에 문자열을 씁니다. (유니코드)
TranslateCharsetInfo 문자 집합 정보를 변환하고 대상 구조의 모든 멤버를 적절한 값으로 설정합니다.
TransparentBlt TransparentBlt 함수는 지정된 원본 디바이스 컨텍스트에서 대상 디바이스 컨텍스트로 픽셀 사각형에 해당하는 색 데이터의 비트 블록 전송을 수행합니다.
UnrealizeObject UnrealizeObject 함수는 브러시의 원본을 다시 설정하거나 논리 팔레트를 다시 설정합니다.

UpdateColors UpdateColors 함수는 클라이언트 영역의 현재 색을 현재 실현된 논리 팔레트에 다시 매핑하여 지정된 디바이스 컨텍스트의 클라이언트 영역을 업데이트합니다.
UpdateICMRegKeyA UpdateICMRegKey 함수는 시스템의 색 프로필 및 색 관리 모듈을 관리합니다. (ANSI)
UpdateICMRegKeyW UpdateICMRegKey 함수는 시스템의 색 프로필 및 색 관리 모듈을 관리합니다. (유니코드)
wglCopyContext wglCopyContext 함수는 선택한 렌더링 상태 그룹을 한 OpenGL 렌더링 컨텍스트에서 다른 OpenGL 렌더링 컨텍스트로 복사합니다.
wglCreateContext wglCreateContext 함수는 hdc에서 참조하는 디바이스에 그리기에 적합한 새 OpenGL 렌더링 컨텍스트를 만듭니다. 렌더링 컨텍스트는 디바이스 컨텍스트와 픽셀 형식이 동일합니다.
wglCreateLayerContext wglCreateLayerContext 함수는 디바이스 컨텍스트에서 지정된 레이어 평면에 그리기 위한 새 OpenGL 렌더링 컨텍스트를 만듭니다.
wglDeleteContext wglDeleteContext 함수는 지정된 OpenGL 렌더링 컨텍스트를 삭제합니다.
wglDescribeLayerPlane wglDescribeLayerPlane 함수는 지정된 픽셀 형식의 레이어 평면에 대한 정보를 가져옵니다.
wglGetCurrentContext wglGetCurrentContext 함수는 호출 스레드의 현재 OpenGL 렌더링 컨텍스트에 대한 핸들을 가져옵니다.
wglGetCurrentDC wglGetCurrentDC 함수는 호출 스레드의 현재 OpenGL 렌더링 컨텍스트와 연결된 디바이스 컨텍스트에 대한 핸들을 가져옵니다.
wglGetLayerPaletteEntries 지정된 디바이스 컨텍스트에 대해 지정된 색 인덱스 계층 평면에서 색상표 항목을 검색합니다.

wglGetProcAddress wglGetProcAddress 함수는 현재 OpenGL 렌더링 컨텍스트에서 사용할 OpenGL 확장 함수의 주소를 반환합니다.
wglMakeCurrent wglMakeCurrent 함수는 지정된 OpenGL 렌더링 컨텍스트를 호출 스레드의 현재 렌더링 컨텍스트로 만듭니다.
wglRealizeLayerPalette wglRealizeLayerPalette 함수는 지정된 색인 계층 평면의 색상표 항목을 실제 색상표에 매핑하거나 RGBA 레이어 평면의 팔레트를 초기화합니다.
wglSetLayerPaletteEntries 지정된 디바이스 컨텍스트에 대해 지정된 색 인덱스 계층 평면의 색상표 항목을 설정합니다.
wglShareLists wglShareLists 함수를 사용하면 여러 OpenGL 렌더링 컨텍스트에서 단일 표시 목록 공간을 공유할 수 있습니다.
wglSwapLayerBuffers wglSwapLayerBuffers 함수는 지정된 디바이스 컨텍스트에서 참조하는 창의 오버레이, 언더레이 및 주 평면에서 전면 및 후면 버퍼를 교환합니다.
wglUseFontBitmapsA wglUseFontBitmaps 함수는 현재 OpenGL 렌더링 컨텍스트에서 사용할 비트맵 표시 목록 집합을 만듭니다. (ANSI)
wglUseFontBitmapsW wglUseFontBitmaps 함수는 현재 OpenGL 렌더링 컨텍스트에서 사용할 비트맵 표시 목록 집합을 만듭니다. (유니코드)
wglUseFontOutlinesA wglUseFontOutlines 함수는 현재 렌더링 컨텍스트에서 사용할 디바이스 컨텍스트의 현재 선택된 개요 글꼴의 각 문자 모양에 대해 하나씩 표시 목록 집합을 만듭니다. (ANSI)
wglUseFontOutlinesW wglUseFontOutlines 함수는 현재 렌더링 컨텍스트에서 사용할 디바이스 컨텍스트의 현재 선택된 개요 글꼴의 각 문자 모양에 대해 하나씩 표시 목록 집합을 만듭니다. (유니코드)

WidenPath WidenPath 함수는 현재 경로를 지정된 디바이스 컨텍스트에 현재 선택된 펜을 사용하여 경로를 스트로크한 경우 그릴 영역으로 다시 정의합니다.

콜백 함수

ABORTPROC AbortProc 함수는 SetAbortProc 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
ENHMFENUMPROC EnhMetaFileProc 함수는 EnumEnhMetaFile 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
GOBJENUMPROC EnumObjectsProc 함수는 EnumObjects 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
ICMENUMPROCA EnumICMProfilesProcCallback 콜백은 EnumICMProfiles에서 색 프로필 데이터를 처리하는 애플리케이션 정의 콜백 함수입니다. (ANSI)
ICMENUMPROCW EnumICMProfilesProcCallback 콜백은 EnumICMProfiles에서 색 프로필 데이터를 처리하는 애플리케이션 정의 콜백 함수입니다. (유니코드)
LINEDDAPROC LineDDAProc 함수는 LineDDA 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
MFENUMPROC EnumMetaFileProc 함수는 Windows 형식 메타파일 레코드를 처리하는 애플리케이션 정의 콜백 함수입니다.

구조체

ABC ABC 구조체에는 트루타입 글꼴의 문자 너비가 포함됩니다.
ABCFLOAT ABCFLOAT 구조체에는 글꼴 문자의 A, B 및 C 너비가 포함됩니다.
AXESLISTA AXESLIST 구조체에는 여러 마스터 글꼴의 모든 축에 대한 정보가 포함됩니다. (ANSI)
AXESLISTW AXESLIST 구조체에는 여러 마스터 글꼴의 모든 축에 대한 정보가 포함됩니다. (유니코드)
AXISINFOA AXISINFO 구조체에는 여러 마스터 글꼴의 축에 대한 정보가 포함됩니다. (ANSI)
AXISINFOW AXISINFO 구조체에는 여러 마스터 글꼴의 축에 대한 정보가 포함됩니다. (유니코드)
비트맵 BITMAP 구조체는 비트맵의 형식, 너비, 높이, 색 형식 및 비트 값을 정의합니다.
BITMAPCOREHEADER BITMAPCOREHEADER 구조체에는 DIB의 차원 및 색 형식에 대한 정보가 포함됩니다.
BITMAPCOREINFO BITMAPCOREINFO 구조체는 DIB의 차원 및 색 정보를 정의합니다.
BITMAPFILEHEADER BITMAPFILEHEADER 구조체에는 DIB가 포함된 파일의 형식, 크기 및 레이아웃에 대한 정보가 포함됩니다.
BITMAPINFO BITMAPINFO 구조체는 DIB의 차원 및 색 정보를 정의합니다.
BITMAPINFOHEADER BITMAPINFOHEADER 구조체에는 디바이스 독립적 비트맵(DIB)의 차원 및 색 형식에 대한 정보가 포함됩니다.

BITMAPV4HEADER BITMAPV4HEADER 구조는 비트맵 정보 헤더 파일입니다. BITMAPINFOHEADER 구조의 확장 버전입니다. 애플리케이션은 추가 기능에 BITMAPV5HEADER 구조를 사용할 수 있습니다.
BITMAPV5HEADER BITMAPV5HEADER 구조는 비트맵 정보 헤더 파일입니다. BITMAPINFOHEADER 구조의 확장 버전입니다.
BLENDFUNCTION BLENDFUNCTION 구조체는 소스 및 대상 비트맵에 대한 혼합 함수를 지정하여 혼합을 제어합니다.
CHARSETINFO 문자 집합에 대한 정보를 포함합니다.
CIEXYZ CIEXYZ 구조체에는 지정된 색 공간에서 특정 색의 x, y 및 z 좌표가 포함됩니다.
CIEXYZTRIPLE CIEXYZTRIPLE 구조체에는 지정된 논리적 색 공간에 대한 빨강, 녹색 및 파랑 엔드포인트에 해당하는 세 가지 색의 x,y 및 z 좌표가 포함됩니다.
COLORADJUSTMENT COLORADJUSTMENT 구조체는 스트레치 모드가 HALFTONE인 경우 StretchBlt 및 StretchDIBits 함수에서 사용하는 색 조정 값을 정의합니다. SetColorAdjustment 함수를 호출하여 색 조정 값을 설정할 수 있습니다.
DESIGNVECTOR DESIGNVECTOR 구조는 애플리케이션에서 여러 마스터 글꼴의 축에 대한 값을 지정하는 데 사용됩니다.
DEVMODEA DEVMODE 데이터 구조에는 프린터 또는 디스플레이 디바이스의 초기화 및 환경에 대한 정보가 포함됩니다.
DEVMODEW DEVMODEW 구조는 유니코드(와이드) 문자 집합에서 디스플레이 및 인쇄 디바이스의 특성을 지정하는 데 사용됩니다.

DIBSECTION DIBSECTION 구조에는 CreateDIBSection 함수를 호출하여 만든 DIB에 대한 정보가 포함됩니다.
DISPLAY_DEVICEA DISPLAY_DEVICE 구조체는 EnumDisplayDevices 함수의 iDevNum 매개 변수로 지정된 디스플레이 디바이스에 대한 정보를 받습니다. (ANSI)
DISPLAY_DEVICEW DISPLAY_DEVICE 구조체는 EnumDisplayDevices 함수의 iDevNum 매개 변수로 지정된 디스플레이 디바이스에 대한 정보를 받습니다. (유니코드)
DISPLAYCONFIG_2DREGION DISPLAYCONFIG_2DREGION 구조체는 2차원 공간의 점 또는 오프셋을 나타냅니다.
DISPLAYCONFIG_ADAPTER_NAME DISPLAYCONFIG_ADAPTER_NAME 구조에는 디스플레이 어댑터에 대한 정보가 포함되어 있습니다.
DISPLAYCONFIG_DESKTOP_IMAGE_INFO DISPLAYCONFIG_DESKTOP_IMAGE_INFO 구조체에는 바탕 화면에 표시되는 이미지에 대한 정보가 포함됩니다.
DISPLAYCONFIG_DEVICE_INFO_HEADER DISPLAYCONFIG_DEVICE_INFO_HEADER 구조에는 디바이스에 대한 표시 정보가 포함됩니다.
DISPLAYCONFIG_MODE_INFO DISPLAYCONFIG_MODE_INFO 구조에는 소스 모드 또는 대상 모드 정보가 포함됩니다.
DISPLAYCONFIG_PATH_INFO DISPLAYCONFIG_PATH_INFO 구조체는 대상에서 원본으로의 단일 경로를 설명하는 데 사용됩니다.
DISPLAYCONFIG_PATH_SOURCE_INFO DISPLAYCONFIG_PATH_SOURCE_INFO 구조에는 단일 경로에 대한 원본 정보가 포함됩니다.
DISPLAYCONFIG_PATH_TARGET_INFO DISPLAYCONFIG_PATH_TARGET_INFO 구조에는 단일 경로에 대한 대상 정보가 포함됩니다.

DISPLAYCONFIG_RATIONAL DISPLAYCONFIG_RATIONAL 구조체는 비디오 모드(즉, 세로 동기화 및 가로 동기화)의 세로 및 가로 빈도를 나타내는 소수 자릿수 값을 설명합니다.
DISPLAYCONFIG_SDR_WHITE_LEVEL DISPLAYCONFIG_SDR_WHITE_LEVEL 구조체(wingdi.h)에는 디스플레이의 현재 SDR 흰색 수준에 대한 정보가 포함되어 있습니다.
DISPLAYCONFIG_SET_TARGET_PERSISTENCE DISPLAYCONFIG_SET_TARGET_PERSISTENCE 구조에는 디스플레이 설정에 대한 정보가 포함되어 있습니다.
DISPLAYCONFIG_SOURCE_DEVICE_NAME DISPLAYCONFIG_SOURCE_DEVICE_NAME 구조에는 원본 또는 뷰에 대한 GDI 디바이스 이름이 포함됩니다.
DISPLAYCONFIG_SOURCE_MODE DISPLAYCONFIG_SOURCE_MODE 구조체는 2차원 공간의 점 또는 오프셋을 나타냅니다.
DISPLAYCONFIG_SUPPORT_VIRTUAL_RESOLUTION DISPLAYCONFIG_SUPPORT_VIRTUAL_RESOLUTION 구조에는 모니터에 대한 가상 해상도 지원 상태에 대한 정보가 포함되어 있습니다.
DISPLAYCONFIG_TARGET_BASE_TYPE 지정된 대상 ID에 대한 기본 출력 기술 정보를 지정합니다.
DISPLAYCONFIG_TARGET_DEVICE_NAME DISPLAYCONFIG_TARGET_DEVICE_NAME 구조에는 대상에 대한 정보가 포함됩니다.
DISPLAYCONFIG_TARGET_DEVICE_NAME_FLAGS DISPLAYCONFIG_TARGET_DEVICE_NAME_FLAGS 구조에는 대상 디바이스에 대한 정보가 포함됩니다.
DISPLAYCONFIG_TARGET_MODE DISPLAYCONFIG_TARGET_MODE 구조체는 표시 경로 대상 모드를 설명합니다.

DISPLAYCONFIG_TARGET_PREFERRED_MODE DISPLAYCONFIG_TARGET_PREFERRED_MODE 구조체에는 디스플레이의 기본 설정 모드에 대한 정보가 포함됩니다.
DISPLAYCONFIG_VIDEO_SIGNAL_INFO DISPLAYCONFIG_VIDEO_SIGNAL_INFO 구조에는 디스플레이에 대한 비디오 신호에 대한 정보가 포함됩니다.
DOCINFOA DOCINFO 구조체에는 StartDoc 함수에서 사용하는 입력 및 출력 파일 이름 및 기타 정보가 포함됩니다. (ANSI)
DOCINFOW DOCINFO 구조체에는 StartDoc 함수에서 사용하는 입력 및 출력 파일 이름 및 기타 정보가 포함됩니다. (유니코드)
DRAWPATRECT DRAWPATRECT 구조체는 만들 사각형을 정의합니다.
EMR EMR 구조는 모든 향상된 메타파일 레코드에 대한 기본 구조를 제공합니다. 향상된 메타파일 레코드에는 향상된 형식 메타파일로 그림의 일부를 만드는 데 사용되는 특정 GDI 함수에 대한 매개 변수가 포함됩니다.
EMRABORTPATH AbortPath, BeginPath, EndPath, CloseFigure, FlattenPath, WidenPath, SetMetaRgn, SaveDC 및 RealizePalette 고급 메타파일 레코드에 대한 데이터를 포함합니다.
EMRALPHABLEND EMRALPHABLEND 구조체에는 AlphaBlend 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRANGLEARC EMRANGLEARC 구조체에는 AngleArc 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRARC EMRARC, EMRARCTO, EMRCHORD 및 EMRPIE 구조에는 Arc, ArcTo, 코드 및 원형 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.

EMRBITBLT EMRBITBLT 구조체에는 BitBlt 향상된 메타파일 레코드에 대한 멤버가 포함됩니다. GDI(그래픽 디바이스 인터페이스)는 메타파일 레코드에 저장하기 전에 디바이스 종속 비트맵을 디바이스 독립적 비트맵(DIB)으로 변환합니다.
EMRCOLORCORRECTPALETTE EMRCOLORCORRECTPALETTE 구조체에는 ColorCorrectPalette 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRCOLORMATCHTOTARGET EMRCOLORMATCHTOTARGET 구조체에는 ColorMatchToTarget 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRCREATEBRUSHINDIRECT EMRCREATEBRUSHINDIRECT 구조체에는 CreateBrushIndirect 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRCREATECOLORSPACE EMRCREATECOLORSPACE 구조체에는 CreateColorSpace 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRCREATECOLORSPACEW EMRCREATECOLORSPACEW 구조체에는 CreateColorSpace 향상된 메타파일 레코드에 대한 멤버가 포함됩니다. EMRCREATECOLORSPACE는 유니코드 논리 색 공간이 있고 원시 소스 프로필 데이터를 포함하는 선택적 배열도 있다는 점에서 EMRCREATECOLORSPACE와 다릅니다.
EMRCREATEDIBPATTERNBRUSHPT EMRCREATEDIBPATTERNBRUSHPT 구조체에는 CreateDIBPatternBrushPt 향상된 메타파일 레코드에 대한 멤버가 포함됩니다. BITMAPINFO 구조 다음에는 압축된 DIB(디바이스 독립적 비트맵)를 형성하는 비트맵 비트가 잇습니다.
EMRCREATEMONOBRUSH EMRCREATEMONOBRUSH 구조체에는 CreatePatternBrush(단색 비트맵을 전달한 경우) 또는 CreateDIBPatternBrush(단색 DIB를 전달한 경우) 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRCREATEPALETTE EMRCREATEPALETTE 구조체에는 CreatePalette 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.

EMRCREATEPEN EMRCREATEPEN 구조체에는 CreatePen 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRELLIPSE EMRELLIPSE 및 EMRRECTANGLE 구조체에는 타원 및 Rectangle 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMREOF EMREOF 구조에는 메타파일의 끝을 나타내는 향상된 메타파일 레코드에 대한 데이터가 포함됩니다.
EMREXCLUDECLIPRECT EMREXCLUDECLIPRECT 및 EMRINTERSECTCLIPRECT 구조체에는 ExcludeClipRect 및 IntersectClipRect 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMREXTCREATEFONTINDIRECTW EMREXTCREATEFONTINDIRECTW 구조체에는 CreateFontIndirect 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMREXTCREATEPEN EMREXTCREATEPEN 구조체에는 ExtCreatePen 향상된 메타파일 레코드에 대한 멤버가 포함됩니다. 레코드에 BITMAPINFO 구조가 포함된 경우 압축된 DIB(디바이스 독립적 비트맵)를 형성하는 비트맵 비트가 뒤따릅니다.
EMREXTFLOODFILL EMREXTFLOODFILL 구조체에는 ExtFloodFill 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMREXTSELECTCLIPRGN EMREXTSELECTCLIPRGN 구조체에는 ExtSelectClipRgn 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMREXTTEXTOUTA EMREXTTEXTOUTA 및 EMREXTTEXTOUTW 구조체에는 ExtTextOut, TextOut 또는 DrawText 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRFILLPATH EMRFILLPATH, EMRSTROKEANDFILLPATH 및 EMRSTROKEPATH 구조체에는 FillPath, StrokeAndFillPath 및 StrokePath 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.

EMRFILLRGN EMRFILLRGN 구조체에는 FillRgn 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRFORMAT EMRFORMAT 구조체에는 향상된 메타파일의 그래픽 데이터를 식별하는 정보가 포함되어 있습니다. GDICOMMENT_MULTIFORMATS 향상된 메타파일 공개 주석에는 EMRFORMAT 구조의 배열이 포함되어 있습니다.
EMRFRAMERGN EMRFRAMERGN 구조체에는 FrameRgn 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRGDICOMMENT EMRGDICOMMENT 구조에는 애플리케이션별 데이터가 포함됩니다.
EMRGLSBOUNDEDRECORD EMRGLSBOUNDEDRECORD 구조체에는 OpenGL 함수에서 생성된 향상된 메타파일 레코드에 대한 멤버가 포함됩니다. 메타파일을 재생할 때 크기를 조정해야 하는 픽셀 단위의 정보가 포함된 OpenGL 함수에 대한 데이터가 포함됩니다.
EMRGLSRECORD EMRGLSRECORD 구조체에는 OpenGL 함수에서 생성된 향상된 메타파일 레코드에 대한 멤버가 포함됩니다. OpenGL 뷰포트로 자동으로 스케일링되는 OpenGL 함수에 대한 데이터가 포함되어 있습니다.
EMRGRADIENTFILL EMRGRADIENTFILL 구조체에는 GradientFill 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRINVERTRGN EMRINVERTRGN 및 EMRPAINTRGN 구조체에는 InvertRgn 및 PaintRgn 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRLINETO EMRLINETO 및 EMRMOVETOEX 구조체에는 LineTo 및 MoveToEx 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRMASKBLT EMRMASKBLT 구조체에는 MaskBlt 향상된 메타파일 레코드에 대한 멤버가 포함됩니다. GDI(그래픽 디바이스 인터페이스)는 메타파일 레코드에 저장하기 전에 디바이스 종속 비트맵을 디바이스 독립적 DIB(비트맵)로 변환합니다.

EMRMODIFYWORLDTRANSFORM EMRMODIFYWORLDTRANSFORM 구조체에는 ModifyWorldTransform 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMROFFSETCLIPRGN EMROFFSETCLIPRGN 구조체에는 OffsetClipRgn 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRPIXELFORMAT EMRPIXELFORMAT 구조체에는 SetPixelFormat 향상된 메타파일 레코드에 대한 멤버가 포함됩니다. ENHMETAHEADER의 픽셀 형식 정보는 이 구조를 나타냅니다.
EMRPLGBLT EMRPLGBLT 구조체에는 PlgBlt 고급 메타파일 레코드에 대한 멤버가 포함됩니다. GDI(그래픽 디바이스 인터페이스)는 메타파일 레코드에 저장하기 전에 디바이스 종속 비트맵을 디바이스 독립적 DIB(비트맵)로 변환합니다.
EMRPOLYDRAW EMRPOLYDRAW 구조체에는 PolyDraw 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRPOLYDRAW16 EMRPOLYDRAW16 구조체에는 PolyDraw 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRPOLYLINE EMRPOLYLINE, EMRPOLYBEZIER, EMRPOLYGON, EMRPOLYBEZIERTO 및 EMRPOLYLINETO 구조체에는 Polyline, PolyBezier, Polygon, PolyBezierTo 및 PolylineTo 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRPOLYLINE16 EMRPOLYLINE16, EMRPOLYBEZIER16, EMRPOLYGON16, EMRPOLYBEZIERTO16 및 EMRPOLYLINETO16 구조체에는 Polyline, PolyBezier, Polygon, PolyBezierTo 및 PolylineTo 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRPOLYPOLYLINE EMRPOLYPOLYLINE 및 EMRPOLYPOLYGON 구조체에는 PolyPolyline 및 PolyPolygon 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.

EMRPOLYPOLYLINE16 EMRPOLYPOLYLINE16 및 EMRPOLYPOLYGON16 구조체에는 PolyPolyline 및 PolyPolygon 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRPOLYTEXTOUTA EMRPOLYTEXTOUTA 및 EMRPOLYTEXTOUTW 구조체에는 PolyTextOut 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRRESIZEPALETTE EMRRESIZEPALETTE 구조체에는 ResizePalette 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRRESTOREDC EMRRESTOREDC 구조체에는 RestoreDC 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRROUNDRECT EMRROUNDRECT 구조체에는 RoundRect 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSCALEVIEWPORTEXTEX EMRSCALEVIEWPORTEXTEX 및 EMRSCALEWINDOWEXTEX 구조에는 ScaleViewportExtEx 및 ScaleWindowExtEx 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSELECTCLIPPATH SelectClipPath, SetBkMode, SetMapMode, SetPolyFillMode, SetROP2, SetStretchBltMode, SetTextAlign, SetICMMode 및 SetLayout 고급 메타파일 레코드에 대한 매개 변수를 포함합니다.
EMRSELECTOBJECT EMRSELECTOBJECT 및 EMRDELETEOBJECT 구조체에는 SelectObject 및 DeleteObject 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSELECTPALETTE EMRSELECTPALETTE 구조체에는 SelectPalette 고급 메타파일 레코드에 대한 멤버가 포함됩니다. SelectPalette의 bForceBackground 매개 변수는 항상 TRUE로 기록되므로 색상표가 배경 팔레트로 실현됩니다.
EMRSETARCDIRECTION EMRSETARCDIRECTION 구조체에는 SetArcDirection 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.

EMRSETBKCOLOR EMRSETBKCOLOR 및 EMRSETTEXTCOLOR 구조체에는 SetBkColor 및 SetTextColor 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSETCOLORADJUSTMENT EMRSETCOLORADJUSTMENT 구조체에는 SetColorAdjustment 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSETCOLORSPACE EMRSETCOLORSPACE, EMRSELECTCOLORSPACE 및 EMRDELETECOLORSPACE 구조체에는 SetColorSpace 및 DeleteColorSpace 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSETDIBITSTODEVICE EMRSETDIBITSTODEVICE 구조체에는 SetDIBitsToDevice 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSETICMPROFILE EMRSETICMPROFILE 구조체에는 SetICMProfile 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSETMAPPERFLAGS EMRSETMAPPERFLAGS 구조체에는 SetMapperFlags 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSETMITERLIMIT EMRSETMITERLIMIT 구조체에는 SetMiterLimit 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSETPALETTEENTRIES EMRSETPALETTEENTRIES 구조체에는 SetPaletteEntries 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSETPIXELV EMRSETPIXELV 구조체에는 SetPixelV 향상된 메타파일 레코드에 대한 멤버가 포함됩니다. 향상된 메타파일을 만들면 SetPixel에 대한 호출도 이 레코드에 기록됩니다.
EMRSETVIEWPORTEXTEX EMRSETVIEWPORTEXTEX 및 EMRSETWINDOWEXTEX 구조체에는 SetViewportExtEx 및 SetWindowExtEx 고급 메타파일 레코드에 대한 멤버가 포함됩니다.

EMRSETVIEWPORTORGEX EMRSETVIEWPORTORGEX, EMRSETWINDOWORGEX 및 EMRSETBRUSHORGEX 구조체에는 SetViewportOrgEx, SetWindowOrgEx 및 SetBrushOrgEx 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSETWORLDTRANSFORM EMRSETWORLDTRANSFORM 구조체에는 SetWorldTransform 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRSTRETCHBLT EMRSTRETCHBLT 구조체에는 StretchBlt 고급 메타파일 레코드에 대한 멤버가 포함됩니다. GDI(그래픽 디바이스 인터페이스)는 메타파일 레코드에 저장하기 전에 디바이스 종속 비트맵을 디바이스 독립적 DIB(비트맵)로 변환합니다.
EMRSTRETCHDIBITS EMRSTRETCHDIBITS 구조체에는 StretchDIBits 향상된 메타파일 레코드에 대한 멤버가 포함됩니다.
EMRTEXT EMRTEXT 구조체에는 텍스트 출력에 대한 멤버가 포함됩니다.
EMRTRANSPARENTBLT EMRTRANSPARENTBLT 구조체에는 TransparentBLT 고급 메타파일 레코드에 대한 멤버가 포함됩니다.
ENHMETAHEADER ENHMETAHEADER 구조에는 향상된 메타파일에 저장된 그림의 크기, 향상된 메타파일의 레코드 수, 그림을 만든 장치의 해상도 등과 같은 향상된 메타파일 데이터가 포함됩니다. 이 구조는 항상 향상된 메타파일의 첫 번째 레코드입니다.
ENHMETARECORD ENHMETARECORD 구조체에는 고급 형식 메타파일로 그림의 일부를 만드는 데 사용되는 GDI(그래픽 디바이스 인터페이스) 함수를 설명하는 데이터가 포함되어 있습니다.
ENUMLOGFONTA ENUMLOGFONT 구조체는 글꼴의 특성, 글꼴의 전체 이름 및 글꼴 스타일을 정의합니다. (ANSI)
ENUMLOGFONTEXA ENUMLOGFONTEX 구조체에는 열거형 글꼴에 대한 정보가 포함되어 있습니다. (ANSI)

ENUMLOGFONTEXDVA ENUMLOGFONTEXDV 구조체에는 글꼴을 만드는 데 사용되는 정보가 포함됩니다. (ANSI)
ENUMLOGFONTEXDVW ENUMLOGFONTEXDV 구조체에는 글꼴을 만드는 데 사용되는 정보가 포함됩니다. (유니코드)
ENUMLOGFONTEXW ENUMLOGFONTEX 구조체에는 열거형 글꼴에 대한 정보가 포함되어 있습니다. (유니코드)
ENUMLOGFONTW ENUMLOGFONT 구조체는 글꼴의 특성, 글꼴의 전체 이름 및 글꼴 스타일을 정의합니다. (유니코드)
ENUMTEXTMETRICA ENUMTEXTMETRIC 구조체에는 실제 글꼴에 대한 정보가 포함됩니다. (ANSI)
ENUMTEXTMETRICW ENUMTEXTMETRIC 구조체에는 실제 글꼴에 대한 정보가 포함됩니다. (유니코드)
EXTLOGFONTA EXTLOGFONT 구조체는 글꼴의 특성을 정의합니다. (ANSI)
EXTLOGFONTW EXTLOGFONT 구조체는 글꼴의 특성을 정의합니다. (유니코드)
EXTLOGPEN EXTLOGPEN 구조체는 확장 펜의 펜 스타일, 너비 및 브러시 특성을 정의합니다.
FIXED FIXED 구조체에는 고정 소수점 실수의 정수 부분과 소수 부분이 포함됩니다.
FONTSIGNATURE 지정된 글꼴에서 문자 모양을 제공하는 코드 페이지 및 유니코드 하위 범위를 식별하는 정보를 포함합니다.

GCP_RESULTSA GCP_RESULTS 구조체에는 문자열의 문자에 대한 정보가 포함됩니다. 이 구조체는 GetCharacterPlacement 함수의 결과를 받습니다. 일부 언어의 경우 배열의 첫 번째 요소에 더 많은 언어 종속 정보가 포함될 수 있습니다. (ANSI)
GCP_RESULTSW GCP_RESULTS 구조체에는 문자열의 문자에 대한 정보가 포함됩니다. 이 구조체는 GetCharacterPlacement 함수의 결과를 받습니다. 일부 언어의 경우 배열의 첫 번째 요소에 더 많은 언어 종속 정보가 포함될 수 있습니다. (유니코드)
문자 모양 GLYPHMETRICS 구조체에는 문자 셀의 문자 모양 배치 및 방향에 대한 정보가 포함됩니다.
GLYPHMETRICSFLOAT GLYPHMETRICSFLOAT 구조체에는 문자 셀의 문자 모양 배치 및 방향에 대한 정보가 포함됩니다.
문자 모양 집합 GLYPHSET 구조체에는 유니코드 코드 포인트 범위에 대한 정보가 포함되어 있습니다.
GRADIENT_RECT GRADIENT_RECT 구조체는 GradientFill 함수의 pVertex 배열에 있는 두 꼭짓점의 인덱스를 지정합니다. 이 두 꼭짓점은 사각형의 왼쪽 위와 오른쪽 아래 경계를 형성합니다.
GRADIENT_TRIANGLE GRADIENT_TRIANGLE 구조체는 GradientFill 함수의 pVertex 배열에 있는 세 꼭짓점의 인덱스를 지정합니다. 이 세 꼭짓점은 하나의 삼각형을 형성합니다.
HANDLETABLE HANDLETABLE 구조는 각각 GDI(그래픽 디바이스 인터페이스) 개체를 식별하는 핸들 배열입니다.
KERNINGPAIR KERNINGPAIR 구조체는 커닝 쌍을 정의합니다.
LAYERPLANEDESCRIPTOR LAYERPLANEDESCRIPTOR 구조체는 그리기 표면의 픽셀 형식을 설명합니다.

LOCALESIGNATURE 기본 및 지원되는 문자 집합 및 코드 페이지를 정의하는 두 개의 코드 페이지 비트필드(CPB)를 포함하여 확장된 글꼴 서명 정보를 포함합니다. 이 구조는 일반적으로 글꼴 검사와 로캘 간의 관계를 나타내는 데 사용됩니다.
LOGBRUSH LOGBRUSH 구조체는 실제 브러시의 스타일, 색 및 패턴을 정의합니다. CreateBrushIndirect 및 ExtCreatePen 함수에서 사용됩니다.
LOGBRUSH32 LOGBRUSH32 구조체는 실제 브러시의 스타일, 색 및 패턴을 정의합니다.
LOGCOLORSPACEA LOGCOLORSPACE 구조체에는 논리적 색 공간을 정의하는 정보가 포함되어 있습니다. (ANSI)
LOGCOLORSPACEW LOGCOLORSPACE 구조체에는 논리적 색 공간을 정의하는 정보가 포함되어 있습니다. (유니코드)
LOGFONTA LOGFONT 구조체는 글꼴의 특성을 정의합니다. (ANSI)
LOGFONTW LOGFONT 구조체는 글꼴의 특성을 정의합니다. (유니코드)
LOGPALETTE LOGPALETTE 구조체는 논리 팔레트를 정의합니다.
LOGPEN LOGPEN 구조는 펜의 스타일, 너비 및 색을 정의합니다. CreatePenIndirect 함수는 LOGPEN 구조를 사용합니다.
MAT2 MAT2 구조체에는 GetGlyphOutline 함수에서 사용하는 변환 행렬에 대한 값이 포함됩니다.
METAFILEPICT 클립보드를 통해 메타파일 데이터를 교환하는 데 사용되는 메타파일 그림 형식을 정의합니다.

메타헤더 METAHEADER 구조에는 Windows 형식 메타파일에 대한 정보가 포함되어 있습니다.
METARECORD METARECORD 구조체에는 Windows 형식 메타파일 레코드가 포함되어 있습니다.
NEWTEXTMETRICA NEWTEXTMETRIC 구조에는 실제 글꼴을 설명하는 데이터가 포함되어 있습니다. (ANSI)
NEWTEXTMETRICEXA NEWTEXTMETRICEX 구조체에는 실제 글꼴에 대한 정보가 포함됩니다. (ANSI)
NEWTEXTMETRICEXW NEWTEXTMETRICEX 구조체에는 실제 글꼴에 대한 정보가 포함됩니다. (유니코드)
NEWTEXTMETRICW NEWTEXTMETRIC 구조에는 실제 글꼴을 설명하는 데이터가 포함되어 있습니다. (유니코드)
OUTLINETEXTMETRICA OUTLINETEXTMETRIC 구조에는 TrueType 글꼴을 설명하는 메트릭이 포함되어 있습니다. (ANSI)
OUTLINETEXTMETRICW OUTLINETEXTMETRIC 구조에는 TrueType 글꼴을 설명하는 메트릭이 포함되어 있습니다. (유니코드)
PALETTEENTRY 논리적 팔레트에서 항목의 색과 사용량을 지정합니다.
PANOSE PANOSE 구조체는 TrueType 글꼴에 대한 PANOSE 글꼴 분류 값을 설명합니다. 이러한 특성은 글꼴을 유사한 모양이지만 다른 이름의 다른 글꼴과 연결하는 데 사용됩니다.
PIXELFORMATDESCRIPTOR PIXELFORMATDESCRIPTOR 구조체는 그리기 표면의 픽셀 형식을 설명합니다.
POINTFLOAT POINTFLOAT 구조체에는 점의 x 및 y 좌표가 포함됩니다.

POINTFX POINTFX 구조체에는 TrueType 글꼴의 문자 윤곽선을 설명하는 점의 좌표가 포함되어 있습니다.
POLYTEXTA POLYTEXT 구조체는 PolyTextOut 함수가 텍스트 문자열을 그리는 방법을 설명합니다. (ANSI)
POLYTEXTW POLYTEXT 구조체는 PolyTextOut 함수가 텍스트 문자열을 그리는 방법을 설명합니다. (유니코드)
PSFEATURE_CUSTPAPER PSFEATURE_CUSTPAPER 구조에는 PostScript 드라이버의 사용자 지정 용지 크기에 대한 정보가 포함되어 있습니다. 이 구조는 GET_PS_FEATURESETTING 프린터 이스케이프 함수와 함께 사용됩니다.
PSFEATURE_OUTPUT PSFEATURE_OUTPUT 구조에는 PostScript 드라이버 출력 옵션에 대한 정보가 포함되어 있습니다. 이 구조는 GET_PS_FEATURESETTING 프린터 이스케이프 함수와 함께 사용됩니다.
PSINJECTDATA PSINJECTDATA 구조는 POSTSCRIPT_INJECTION 프린터 이스케이프 함수와 함께 사용되는 입력 버퍼의 헤더입니다.
RASTERIZER_STATUS RASTERIZER_STATUS 구조에는 TrueType이 설치되어 있는지 여부에 대한 정보가 포함됩니다. 이 구조는 애플리케이션이 GetRasterizerCaps 함수를 호출할 때 채워집니다.
RGBQUAD RGBQUAD 구조는 빨강, 녹색 및 파랑의 상대적 강도로 구성된 색을 설명합니다.
RGBTRIPLE RGBTRIPLE 구조는 빨강, 녹색 및 파랑의 상대적 강도로 구성된 색을 설명합니다. BITMAPCOREINFO 구조체의 bmciColors 멤버는 RGBTRIPLE 구조체의 배열로 구성됩니다.
RGNDATA RGNDATA 구조체에는 영역을 구성하는 머리글 및 사각형 배열이 포함됩니다. 사각형은 위에서 아래로, 왼쪽에서 오른쪽으로 정렬됩니다. 겹치지 않습니다.
RGNDATAHEADER RGNDATAHEADER 구조체는 GetRegionData 함수에서 반환된 데이터를 설명합니다.

TEXTMETRICA TEXTMETRIC 구조에는 실제 글꼴에 대한 기본 정보가 포함되어 있습니다. 모든 크기는 논리 단위로 지정됩니다. 즉, 표시 컨텍스트의 현재 매핑 모드에 따라 달라집니다. (ANSI)
TEXTMETRICW TEXTMETRIC 구조에는 실제 글꼴에 대한 기본 정보가 포함되어 있습니다. 모든 크기는 논리 단위로 지정됩니다. 즉, 표시 컨텍스트의 현재 매핑 모드에 따라 달라집니다. (유니코드)
TRIVERTEX TRIVERTEX 구조체에는 색 정보 및 위치 정보가 포함됩니다.
TTPOLYCURVE TTPOLYCURVE 구조체는 TrueType 문자의 윤곽선에 있는 곡선에 대한 정보를 포함합니다.
TTPOLYGONHEADER TTPOLYGONHEADER 구조체는 TrueType 문자 윤곽선에서 컨투어의 시작 위치와 형식을 지정합니다.
WCRANGE WCRANGE 구조체는 유니코드 문자 범위를 지정합니다.
Xform XFORM 구조는 페이지 공간 변환에 대한 세계 공간을 지정합니다.

열거형

DISPLAYCONFIG_DEVICE_INFO_TYPE DISPLAYCONFIG_DEVICE_INFO_TYPE 열거형은 DisplayConfigSetDeviceInfo 또는 DisplayConfigGetDeviceInfo 함수를 통해 구성하거나 가져올 디스플레이 디바이스 정보의 유형을 지정합니다.
DISPLAYCONFIG_MODE_INFO_TYPE DISPLAYCONFIG_MODE_INFO_TYPE 열거형은 DISPLAYCONFIG_MODE_INFO 구조에 포함된 정보가 원본 또는 대상 모드임을 지정합니다.

DISPLAYCONFIG_PIXELFORMAT DISPLAYCONFIG_PIXELFORMAT 열거형은 BPP(픽셀당 다양한 비트) 값의 픽셀 형식을 지정합니다.
DISPLAYCONFIG_ROTATION DISPLAYCONFIG_ROTATION 열거형은 디스플레이의 시계 방향 회전을 지정합니다.
DISPLAYCONFIG_SCALING DISPLAYCONFIG_SCALING 열거형은 VidPN(비디오 제공 네트워크) 현재 경로에 표시되는 콘텐츠에 적용되는 크기 조정 변환을 지정합니다.
DISPLAYCONFIG_SCANLINE_ORDERING DISPLAYCONFIG_SCANLINE_ORDERING 열거형은 디스플레이가 화면에 이미지를 만드는 데 사용하는 메서드를 지정합니다.
DISPLAYCONFIG_TOPOLOGY_ID DISPLAYCONFIG_TOPOLOGY_ID 열거형은 표시 토폴로지의 유형을 지정합니다.
DISPLAYCONFIG_VIDEO_OUTPUT_TECHNOLOGY DISPLAYCONFIG_VIDEO_OUTPUT_TECHNOLOGY 열거형은 대상의 커넥터 유형을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winhttp.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows HTTP 서비스(WinHTTP)에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows HTTP 서비스(WinHTTP)

winhttp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WinHttpAddRequestHeaders HTTP 요청 핸들에 하나 이상의 HTTP 요청 헤더를 추가합니다. (WinHttpAddRequestHeaders)
WinHttpAddRequestHeadersEx HTTP 요청 핸들에 하나 이상의 HTTP 요청 헤더를 추가하여 별도의 이름/값 문자열을 사용할 수 있습니다.
WinHttpCheckPlatform WinHttpCheckPlatform 함수는 현재 플랫폼이 이 버전의 WinHTTP(Microsoft Windows HTTP 서비스)에서 지원되는지 여부를 결정합니다.
WinHttpCloseHandle WinHttpCloseHandle 함수는 단일 **HINTERNET** 핸들을 닫습니다.
WinHttpConnect WinHttpConnect 함수는 HTTP 요청의 초기 대상 서버를 지정하고 초기 대상에 대한 HTTP 세션에 HINTERNET 연결 핸들을 반환합니다.
WinHttpCrackUrl WinHttpCrackUrl 함수는 URL을 호스트 이름 및 경로와 같은 구성 요소 부분으로 구분합니다.
WinHttpCreateProxyResolver WinHttpGetProxyForUrlEx에서 사용할 핸들을 만듭니다.
WinHttpCreateUrl

호스트 이름 및 경로와 같은 구성 요소 부분에서 URL을 만듭니다.
WinHttpDetectAutoProxyConfigUrl PAC(프록시 자동 구성) 파일의 URL을 찾습니다.
WinHttpFreeProxyResult WinHttpFreeProxyResult 함수는 WinHttpGetProxyResult에 대한 이전 호출에서 검색된 데이터를 해제합니다.
WinHttpFreeProxySettingsEx WinHttpGetProxySettingsResultEx에 대한 이전 호출에서 검색된 데이터를 해제합니다.
WinHttpFreeQueryConnectionGroupResult WinHttpQueryConnectionGroup에 대한 이전 호출에 의해 할당된 메모리를 해제합니다.
WinHttpGetDefaultProxyConfiguration 레지스트리에서 기본 WinHTTP 프록시 구성을 검색합니다.
WinHttpGetIEProxyConfigForCurrentUser 현재 사용자의 인터넷 Explorer 프록시 구성을 검색합니다.
WinHttpGetProxyForUrl 지정된 URL에 대한 프록시 데이터를 검색합니다. (WinHttpGetProxyForUrl)
WinHttpGetProxyForUrlEx 지정된 URL에 대한 프록시 데이터를 검색합니다. (WinHttpGetProxyForUrlEx)
WinHttpGetProxyResult WinHttpGetProxyResult 함수는 WinHttpGetProxyForUrlEx 호출 결과를 검색합니다.
WinHttpGetProxySettingsEx 확장 프록시 설정을 검색합니다.
WinHttpGetProxySettingsResultEx WinHttpGetProxySettingsEx 호출 결과를 검색합니다.
WinHttpOpen 애플리케이션의 경우 WinHTTP 함수를 초기화하고 WinHTTP 세션 핸들을 반환합니다.

WinHttpOpenRequest WinHttpOpenRequest 함수는 HTTP 요청 핸들을 만듭니다.
WinHttpQueryAuthSchemes WinHttpQueryAuthSchemes 함수는 서버에서 지원하는 권한 부여 체계를 반환합니다.
WinHttpQueryConnectionGroup http 연결 및 해당 **GUID**의 열거형을 검색합니다.
WinHttpQueryDataAvailable WinHttpReadData를 사용하여 읽을 수 있는 데이터 양(바이트)을 반환합니다.
WinHttpQueryHeaders WinHttpQueryHeaders 함수는 HTTP 요청과 연결된 헤더 정보를 검색합니다.
WinHttpQueryHeadersEx HTTP 요청과 연결된 헤더 정보를 검색합니다. 는 구문 분석된 헤더 이름 및 값 문자열을 검색하는 방법을 제공합니다.
WinHttpQueryOption WinHttpQueryOption 함수는 지정된 핸들에서 인터넷 옵션을 쿼리합니다.
WinHttpReadData WinHttpReadData 함수는 WinHttpOpenRequest 함수에서 연 핸들에서 데이터를 읽습니다.
WinHttpReadDataEx WinHttpOpenRequest 함수에서 연 핸들에서 데이터를 읽습니다.
WinHttpReceiveResponse WinHttpReceiveResponse 함수는 WinHttpSendRequest에서 시작한 HTTP 요청에 대한 응답을 받기 위해 대기합니다.
WinHttpRegisterProxyChangeNotification 유효 프록시 설정이 변경되면 WinHTTP에서 호출하는 콜백 함수를 등록합니다.
WinHttpResetAutoProxy 자동 프록시를 다시 설정합니다.

WinHttpSendRequest 지정된 요청을 HTTP 서버로 보냅니다. (WinHttpSendRequest)
WinHttpSetCredentials WinHttpSetCredentials 함수는 필요한 권한 부여 자격 증명을 서버에 전달합니다.
WinHttpSetDefaultProxyConfiguration 레지스트리에서 기본 WinHTTP 프록시 구성을 설정합니다.
WinHttpSetOption WinHttpSetOption 함수는 인터넷 옵션을 설정합니다.
WinHttpSetStatusCallback WinHttpSetStatusCallback 함수는 작업 중에 진행이 수행되면 WinHTTP가 호출할 수 있는 콜백 함수를 설정합니다.
WinHttpSetTimeouts HTTP 트랜잭션과 관련된 시간 초과를 설정합니다.
WinHttpTimeFromSystemTime HTTP 버전 1.0 사양에 따라 날짜 및 시간의 형식을 지정합니다. (WinHttpTimeFromSystemTime)
WinHttpTimeToSystemTime WinHttpTimeToSystemTime 함수는 HTTP 시간/날짜 문자열을 사용하여 SYSTEMTIME 구조체로 변환합니다.
WinHttpUnregisterProxyChangeNotification WinHttpRegisterProxyChangeNotification을 호출하여 등록된 콜백 함수의 등록을 취소합니다.
WinHttpWebSocketClose WebSocket 연결을 닫습니다.
WinHttpWebSocketCompleteUpgrade WinHttpSendRequest에서 시작한 WebSocket 핸드셰이크를 완료합니다.
WinHttpWebSocketQueryCloseStatus 서버에서 보낸 닫기 상태 검색합니다.

WinHttpWebSocketReceive WebSocket 연결에서 데이터를 받습니다.
WinHttpWebSocketSend WebSocket 연결을 통해 데이터를 보냅니다.
WinHttpWebSocketShutdown WebSocket 서버에 닫기 프레임을 보내 송신 채널을 닫지만 수신 채널은 열린 상태로 둡니다.
WinHttpWriteData WinHttpWriteData 함수는 요청 데이터를 HTTP 서버에 씁니다.

콜백 함수

WINHTTP_PROXY_CHANGE_CALLBACK 애플리케이션 정의 프록시 변경 콜백 함수를 나타냅니다.
WINHTTP_STATUS_CALLBACK 애플리케이션 정의 상태 콜백 함수를 나타냅니다.

구조체

HTTP_VERSION_INFO HTTP_VERSION_INFO 구조에는 전역 HTTP 버전이 포함됩니다.
URL_COMPONENTS URL_COMPONENTS 구조체에는 URL의 구성 요소가 포함됩니다. 이 구조체는 WinHttpCrackUrl 및 WinHttpCreateUrl 함수와 함께 사용됩니다.
WINHTTP_ASYNC_RESULT

WINHTTP_ASYNC_RESULT 구조체에는 비동기 함수 호출의 결과가 포함됩니다. 이 구조체는 WINHTTP_STATUS_CALLBACK 프로토타입과 함께 사용됩니다.
WINHTTP_AUTOPROXY_OPTIONS WINHTTP_AUTOPROXY_OPTIONS 구조체는 WinHttpGetProxyForURL 함수에 PAC(프록시 자동 구성) 파일의 URL을 지정할지 또는 DHCP 또는 DNS 쿼리를 사용하여 네트워크에 대한 URL을 자동으로 찾을지 여부를 나타내는 데 사용됩니다.
WINHTTP_CERTIFICATE_INFO WINHTTP_CERTIFICATE_INFO 구조에는 서버에서 반환된 인증서 정보가 포함됩니다. 이 구조체는 WinHttpQueryOption 함수에서 사용됩니다.
WINHTTP_CONNECTION_GROUP 연결 그룹을 나타냅니다.
WINHTTP_CONNECTION_INFO WINHTTP_CONNECTION_INFO 구조에는 응답을 생성한 요청의 원본 및 대상 IP 주소가 포함됩니다.
WINHTTP_CREDS 서버 및 프록시 인증에 사용되는 사용자 자격 증명 정보를 포함합니다. (WINHTTP_CREDS)
WINHTTP_CREDS_EX 서버 및 프록시 인증에 사용되는 사용자 자격 증명 정보를 포함합니다. (WINHTTP_CREDS_EX)
WINHTTP_CURRENT_USER_IE_PROXY_CONFIG WINHTTP_CURRENT_USER_IE_PROXY_CONFIG 구조에는 인터넷 Explorer 프록시 구성 정보가 포함됩니다.
WINHTTP_EXTENDED_HEADER HTTP 요청 헤더를 이름/값 문자열 쌍으로 나타냅니다.
WINHTTP_HEADER_NAME HTTP 요청 헤더 이름을 나타냅니다.
WINHTTP_HOST_CONNECTION_GROUP 연결 그룹의 컬렉션을 나타냅니다.
WINHTTP_MATCH_CONNECTION_GUID

연결 일치를 위해 연결의 GUID를 나타냅니다.
WINHTTP_PROXY_INFO WINHTTP_PROXY_INFO 구조에는 세션 또는 기본 프록시 구성이 포함됩니다.
WINHTTP_PROXY_RESULT WINHTTP_PROXY_RESULT 구조에는 WinHttpGetProxyResult에서 제공하는 프록시 결과 항목의 컬렉션이 포함됩니다.
WINHTTP_PROXY_RESULT_ENTRY WINHTTP_PROXY_RESULT_ENTRY 구조체에는 WinHttpGetProxyResult 호출의 결과 항목이 포함됩니다.
WINHTTP_PROXY_SETTINGS_EX WINHTTP_PROXY_SETTINGS_EX 구조는 확장 프록시 설정을 나타냅니다.
WINHTTP_PROXY_SETTINGS_PARAM WINHTTP_PROXY_SETTINGS_PARAM 구조는 확장 프록시 설정을 나타냅니다.
WINHTTP_QUERY_CONNECTION_GROUP_RESULT 호스트 연결 그룹의 컬렉션을 나타냅니다.
WINHTTP_REQUEST_STATS WINHTTP_REQUEST_STATS 구조에는 요청에 대한 다양한 통계가 포함되어 있습니다.
WINHTTP_REQUEST_TIMES WINHTTP_REQUEST_TIMES 구조에는 HTTP 요청에 대한 다양한 타이밍 정보가 포함됩니다.
WINHTTP_SECURITY_INFO WINHTTP_SECURITY_INFO 구조에는 HTTP 요청에 대한 다양한 타이밍 정보가 포함됩니다.
WINHTTP_WEB_SOCKET_ASYNC_RESULT WINHTTP_WEB_SOCKET_ASYNC_RESULT WebSocket 작업의 결과 상태 포함합니다.
WINHTTP_WEB_SOCKET_STATUS WINHTTP_WEB_SOCKET_STATUS 열거형에는 WebSocket 작업의 상태 포함됩니다.

열거형

WINHTTP_PROXY_SETTINGS_TYPE 확장 프록시 설정 API에 대한 프록시 설정 유형을 지정하는 상수를 정의합니다.
WINHTTP_REQUEST_STAT_ENTRY WINHTTP_REQUEST_STAT_ENTRY 열거형에는 사용 가능한 요청 통계 유형이 나열됩니다.
WINHTTP_REQUEST_TIME_ENTRY WINHTTP_REQUEST_TIME_ENTRY 열거형에는 사용 가능한 유형의 요청 타이밍 정보가 나열됩니다.
WINHTTP_WEB_SOCKET_BUFFER_TYPE WINHTTP_WEB_SOCKET_BUFFER_TYPE 열거형에는 WebSocket 버퍼 유형이 포함됩니다.
WINHTTP_WEB_SOCKET_CLOSE_STATUS WINHTTP_WEB_SOCKET_CLOSE_STATUS 열거형에는 WebSocket 닫기 작업의 상태 포함됩니다.
WINHTTP_WEB_SOCKET_OPERATION WINHTTP_WEB_SOCKET_OPERATION 열거형에는 WebSocket 작업 유형이 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wininet.h 헤더

아티클2023. 03. 08.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보
- Windows 인터넷

wininet.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

CommitUrlCacheEntryA 지정된 파일의 데이터를 인터넷 캐시에 저장하고 지정된 URL과 연결합니다. (ANSI)
CommitUrlCacheEntryW 지정된 파일의 데이터를 인터넷 캐시에 저장하고 지정된 URL과 연결합니다. (유니코드)
CreateMD5SSOHash CreateMD5SSOHash 함수(wininet.h)는 Microsoft Passport 암호를 가져오고, 특정 문자열을 사용하여 MD5 해시를 만들고, 결과를 반환합니다.
CreateUrlCacheEntryA 지정된 URL 및 파일 이름 확장명을 기반으로 캐시 항목을 저장하기 위한 로컬 파일 이름을 만듭니다. (ANSI)
CreateUrlCacheEntryW 지정된 URL 및 파일 이름 확장명을 기반으로 캐시 항목을 저장하기 위한 로컬 파일 이름을 만듭니다. (유니코드)
CreateUrlCacheGroup 캐시 그룹 ID를 생성합니다.
DeleteUrlCacheEntry DeleteUrlCacheEntry 함수(wininet.h)는 파일이 있는 경우 캐시에서 원본 이름과 연결된 파일을 제거합니다.

DeleteUrlCacheEntryA 파일이 있는 경우 캐시에서 원본 이름과 연결된 파일을 제거합니다. (DeleteUrlCacheEntryA)
DeleteUrlCacheEntryW DeleteUrlCacheEntryW(유니코드) 함수(wininet.h)는 파일이 있는 경우 캐시에서 원본 이름과 연결된 파일을 제거합니다.
DeleteUrlCacheGroup 캐시 인덱스 파일에서 지정된 GROUPID 및 연결된 상태를 해제합니다.
DetectAutoProxyUrl DetectAutoProxyUrl 함수(wininet.h)는 WPAD 자동 프록시 스크립트의 위치를 확인하려고 시도합니다.
FindCloseUrlCache 지정된 캐시 열거형 핸들을 닫습니다.
FindFirstUrlCacheEntryA 인터넷 캐시의 열거형을 시작합니다. (ANSI)
FindFirstUrlCacheEntryExA 인터넷 캐시의 필터링된 열거형을 시작합니다. (ANSI)
FindFirstUrlCacheEntryExW 인터넷 캐시의 필터링된 열거형을 시작합니다. (유니코드)
FindFirstUrlCacheEntryW 인터넷 캐시의 열거형을 시작합니다. (유니코드)
FindFirstUrlCacheGroup 인터넷 캐시에서 캐시 그룹의 열거를 시작합니다.
FindNextUrlCacheEntryA 인터넷 캐시에서 다음 항목을 검색합니다. (ANSI)
FindNextUrlCacheEntryExA FindFirstUrlCacheEntryEx 함수에서 시작된 캐시 열거형에서 다음 캐시 항목을 찾습니다. (ANSI)

FindNextUrlCacheEntryExW FindFirstUrlCacheEntryEx 함수에서 시작된 캐시 열거형에서 다음 캐시 항목을 찾습니다. (유니코드)
FindNextUrlCacheEntryW 인터넷 캐시에서 다음 항목을 검색합니다. (유니코드)
FindNextUrlCacheGroup FindFirstUrlCacheGroup에서 시작한 캐시 그룹 열거형의 다음 캐시 그룹을 검색합니다.
FtpCommandA 명령을 FTP 서버로 직접 보냅니다. (ANSI)
FtpCommandW 명령을 FTP 서버로 직접 보냅니다. (유니코드)
FtpCreateDirectoryA FTP 서버에 새 디렉터리를 만듭니다. (ANSI)
FtpCreateDirectoryW FTP 서버에 새 디렉터리를 만듭니다. (유니코드)
FtpDeleteFileA FTP 서버에 저장된 파일을 삭제합니다. (ANSI)
FtpDeleteFileW FTP 서버에 저장된 파일을 삭제합니다. (유니코드)
FtpFindFirstFileA 지정된 FTP 세션의 지정된 디렉터리를 검색합니다. 파일 및 디렉터리 항목은 WIN32_FIND_DATA 구조의 애플리케이션에 반환됩니다. (ANSI)
FtpFindFirstFileW 지정된 FTP 세션의 지정된 디렉터리를 검색합니다. 파일 및 디렉터리 항목은 WIN32_FIND_DATA 구조의 애플리케이션에 반환됩니다. (유니코드)
FtpGetCurrentDirectoryA 지정된 FTP 세션의 현재 디렉터리를 검색합니다. (ANSI)

FtpGetCurrentDirectoryW 지정된 FTP 세션의 현재 디렉터리를 검색합니다. (유니코드)
FtpGetFileA FTP 서버에서 파일을 검색하고 지정된 파일 이름 아래에 저장하여 프로세스에 새 로컬 파일을 만듭니다. (ANSI)
FtpGetFileSize 요청된 FTP 리소스의 파일 크기를 검색합니다.
FtpGetFileW FTP 서버에서 파일을 검색하고 지정된 파일 이름 아래에 저장하여 프로세스에 새 로컬 파일을 만듭니다. (유니코드)
FtpOpenFileA 읽기 또는 쓰기를 위해 FTP 서버에서 원격 파일에 대한 액세스를 시작합니다. (ANSI)
FtpOpenFileW 읽기 또는 쓰기를 위해 FTP 서버에서 원격 파일에 대한 액세스를 시작합니다. (유니코드)
FtpPutFileA FTP 서버에 파일을 저장합니다. (ANSI)
FtpPutFileW FTP 서버에 파일을 저장합니다. (유니코드)
FtpRemoveDirectoryA FTP 서버에서 지정된 디렉터리를 제거합니다. (ANSI)
FtpRemoveDirectoryW FTP 서버에서 지정된 디렉터리를 제거합니다. (유니코드)
FtpRenameFileA FTP 서버에 저장된 파일의 이름을 바꿉니다. (ANSI)
FtpRenameFileW FTP 서버에 저장된 파일의 이름을 바꿉니다. (유니코드)

FtpSetCurrentDirectoryA FTP 서버에서 다른 작업 디렉터리로 변경합니다. (ANSI)
FtpSetCurrentDirectoryW FTP 서버에서 다른 작업 디렉터리로 변경합니다. (유니코드)
GetUrlCacheEntryInfoA 캐시 항목에 대한 정보를 검색합니다. (ANSI)
GetUrlCacheEntryInfoExA HttpSendRequest 함수에 의해 오프라인 모드로 적용되는 리디렉션을 고려하여 지정된 URL과 연결된 캐시 항목에 대한 정보를 검색합니다. (ANSI)
GetUrlCacheEntryInfoExW HttpSendRequest 함수에 의해 오프라인 모드로 적용되는 리디렉션을 고려하여 지정된 URL과 연결된 캐시 항목에 대한 정보를 검색합니다. (유니코드)
GetUrlCacheEntryInfoW 캐시 항목에 대한 정보를 검색합니다. (유니코드)
GetUrlCacheGroupAttributeA 지정된 캐시 그룹의 특성 정보를 검색합니다. (ANSI)
GetUrlCacheGroupAttributeW 지정된 캐시 그룹의 특성 정보를 검색합니다. (유니코드)
GopherCreateLocatorA 선택기 문자열의 구성 요소 부분에서 Gopher 또는 Gopher+ 로케이터 문자열을 만듭니다. (ANSI)
GopherCreateLocatorW 선택기 문자열의 구성 요소 부분에서 Gopher 또는 Gopher+ 로케이터 문자열을 만듭니다. (유니코드)
GopherFindFirstFileA Gopher 로케이터 및 검색 조건을 사용하여 서버와 세션을 만들고 요청된 문서, 이진 파일, 인덱스 서버 또는 디렉터리 트리를 찾습니다. (ANSI)

GopherFindFirstFileW Gopher 로케이터 및 검색 조건을 사용하여 서버와 세션을 만들고 요청된 문서, 이진 파일, 인덱스 서버 또는 디렉터리 트리를 찾습니다. (유니코드)
GopherGetAttributeA 서버에서 특정 특성 정보를 검색합니다. (ANSI)
GopherGetAttributeW 서버에서 특정 특성 정보를 검색합니다. (유니코드)
GopherGetLocatorTypeA Gopher 로케이터를 구문 분석하고 해당 특성을 결정합니다. (ANSI)
GopherGetLocatorTypeW Gopher 로케이터를 구문 분석하고 해당 특성을 결정합니다. (유니코드)
GopherOpenFileA Gopher 서버에서 Gopher 데이터 파일을 읽기 시작합니다. (ANSI)
GopherOpenFileW Gopher 서버에서 Gopher 데이터 파일을 읽기 시작합니다. (유니코드)
HttpAddRequestHeadersA HTTP 요청 핸들에 하나 이상의 HTTP 요청 헤더를 추가합니다. (HttpAddRequestHeadersA)
HttpAddRequestHeadersW HTTP 요청 핸들에 하나 이상의 HTTP 요청 헤더를 추가합니다. (HttpAddRequestHeadersW)
HttpEndRequestA HttpSendRequestEx에서 시작된 HTTP 요청을 종료합니다. (ANSI)
HttpEndRequestW HttpSendRequestEx에서 시작된 HTTP 요청을 종료합니다. (유니코드)
HttpOpenRequestA HTTP 요청 핸들을 만듭니다. (ANSI)

HttpOpenRequestW HTTP 요청 핸들을 만듭니다. (유니코드)
HttpQueryInfoA HTTP 요청과 연결된 헤더 정보를 검색합니다. (ANSI)
HttpQueryInfoW HTTP 요청과 연결된 헤더 정보를 검색합니다. (유니코드)
HttpSendRequestA 지정된 요청을 HTTP 서버로 보내 호출자가 일반적으로 HttpSendRequestEx에 전달되는 것 이상으로 추가 데이터를 보낼 수 있도록 합니다. (ANSI)
HttpSendRequestExA 지정된 요청을 HTTP 서버로 보냅니다. (HttpSendRequestExA)
HttpSendRequestExW 지정된 요청을 HTTP 서버로 보냅니다. (HttpSendRequestExW)
HttpSendRequestW 지정된 요청을 HTTP 서버로 보내 호출자가 일반적으로 HttpSendRequestEx에 전달되는 것 이상으로 추가 데이터를 보낼 수 있도록 합니다. (유니코드)
InternetAttemptConnect 인터넷에 연결을 시도합니다.
InternetAutodial InternetAutodial 함수(wininet.h)를 사용하면 모뎀이 기본 인터넷 연결에 자동으로 전화를 걸 수 있습니다.
InternetAutodialHangup InternetAutodialHangup 함수(wininet.h)는 자동 전화 접속 연결을 끊습니다.
InternetCanonicalizeUrlA 안전하지 않은 문자와 공백을 이스케이프 시퀀스로 변환하는 것을 포함하는 URL을 정식화합니다. (ANSI)

InternetCanonicalizeUrlW 안전하지 않은 문자와 공백을 이스케이프 시퀀스로 변환하는 것을 포함하는 URL을 정식화합니다. (유니코드)
InternetCheckConnectionA 애플리케이션에서 인터넷에 대한 연결을 설정할 수 있는지 확인할 수 있습니다. (ANSI)
InternetCheckConnectionW 애플리케이션에서 인터넷에 대한 연결을 설정할 수 있는지 확인할 수 있습니다. (유니코드)
InternetClearAllPerSiteCookieDecisions 사이트별로 쿠키에 대한 모든 결정을 지웁니다.
InternetCloseHandle 단일 인터넷 핸들을 닫습니다.
InternetCombineUrlA 기본 및 상대 URL을 단일 URL로 결합합니다. 결과 URL은 정식화됩니다(InternetCanonicalizeUrl 참조). (ANSI)
InternetCombineUrlW 기본 및 상대 URL을 단일 URL로 결합합니다. 결과 URL은 정식화됩니다(InternetCanonicalizeUrl 참조). (유니코드)
InternetConfirmZoneCrossing InternetConfirmZoneCrossing 함수(wininet.h)는 보안 URL과 비보안 URL 간의 변경 내용을 확인합니다.
InternetConfirmZoneCrossingA 보안 URL과 비보안 URL 간의 변경 내용을 확인합니다. 두 URL 간의 보안이 변경되면 항상 사용자에게 알릴 수 있습니다. 일반적으로 애플리케이션은 사용자가 대화 상자와의 상호 작용을 통해 변경 사항을 승인하도록 허용해야 합니다. (InternetConfirmZoneCrossingA)
InternetConfirmZoneCrossingW InternetConfirmZoneCrossingW(유니코드) 함수(wininet.h)는 보안 URL과 비보안 URL 간의 변경 내용을 확인합니다.

InternetConnectA 지정된 사이트에 대한 FTP(파일 전송 프로토콜) 또는 HTTP 세션을 엽니다. (ANSI)
InternetConnectW 지정된 사이트에 대한 FTP(파일 전송 프로토콜) 또는 HTTP 세션을 엽니다. (유니코드)
InternetCrackUrlA URL을 구성 요소 부분에 크래킹합니다. (ANSI)
InternetCrackUrlW URL을 구성 요소 부분에 크래킹합니다. (유니코드)
InternetCreateUrlA 구성 요소 부분에서 URL을 만듭니다. (ANSI)
InternetCreateUrlW 구성 요소 부분에서 URL을 만듭니다. (유니코드)
InternetDial InternetDial 함수(wininet.h)는 모뎀을 사용하여 인터넷에 대한 연결을 시작합니다.
InternetDialA ANSI(InternetDialA) 함수(wininet.h)는 모뎀을 사용하여 인터넷에 대한 연결을 시작합니다.
InternetDialW InternetDialW(유니코드) 함수(wininet.h)는 모뎀을 사용하여 인터넷에 대한 연결을 시작합니다.
InternetEnumPerSiteCookieDecisionA InternetEnumPerSiteCookieDecisionA(ANSI) 함수(wininet.h)는 설정된 사이트별 쿠키 규정을 사용하여 웹 사이트의 도메인 및 쿠키 설정을 검색합니다.
InternetEnumPerSiteCookieDecisionW InternetEnumPerSiteCookieDecisionW(유니코드) 함수(wininet.h)는 설정된 사이트별 쿠키 규정을 사용하여 웹 사이트의 도메인 및 쿠키 설정을 검색합니다.
InternetErrorDlg 적절한 대화 상자가 있는 경우 InternetErrorDlg에 전달되는 오류에 대한 대화 상자를 표시합니다.

InternetFindNextFileA FtpFindFirstFile.Windows XP 및 Windows Server 2003 R2 이전 호출의 결과로 시작된 파일 검색을 계속합니다. 또는 GopherFindFirstFile에 대한 이전 호출의 결과로 파일 검색을 계속합니다. (ANSI)
InternetFindNextFileW FtpFindFirstFile.Windows XP 및 Windows Server 2003 R2 이전 호출의 결과로 시작된 파일 검색을 계속합니다. 또는 GopherFindFirstFile에 대한 이전 호출의 결과로 파일 검색을 계속합니다. (유니코드)
InternetFreeCookies INTERNET_COOKIE2 구조체의 배열을 해제합니다.
InternetGetConnectedState InternetGetConnectedState 함수(wininet.h)는 로컬 시스템의 연결된 상태를 검색하며 사용하지 않는 것이 좋습니다.
InternetGetConnectedStateEx InternetGetConnectedStateEx 함수(wininet.h)는 지정된 인터넷 연결의 연결된 상태를 검색하며 사용하지 않는 것이 좋습니다.
InternetGetConnectedStateExA INTERNETGetConnectedStateExA(ANSI) 함수(wininet.h)는 지정된 인터넷 연결의 연결된 상태를 검색하며 사용하지 않는 것이 좋습니다.
InternetGetConnectedStateExW InternetGetConnectedStateExW(유니코드) 함수(wininet.h)는 지정된 인터넷 연결의 연결된 상태를 검색하며 사용하지 않는 것이 좋습니다.
InternetGetCookieA 지정된 URL에 대한 쿠키를 검색합니다. (ANSI)
InternetGetCookieEx2 지정된 URL과 연결된 하나 이상의 쿠키를 검색합니다.
InternetGetCookieExA 지정된 URL과 연결된 쿠키에 저장된 데이터를 검색합니다. (ANSI)

InternetGetCookieExW 지정된 URL과 연결된 쿠키에 저장된 데이터를 검색합니다. (유니코드)
InternetGetCookieW 지정된 URL에 대한 쿠키를 검색합니다. (유니코드)
InternetGetLastResponseInfoA 이 함수를 호출하는 스레드에서 마지막 오류 설명 또는 서버 응답을 검색합니다. (ANSI)
InternetGetLastResponseInfoW 이 함수를 호출하는 스레드에서 마지막 오류 설명 또는 서버 응답을 검색합니다. (유니코드)
InternetGetPerSiteCookieDecisionA 지정된 도메인의 쿠키에 대한 결정을 검색합니다. (ANSI)
InternetGetPerSiteCookieDecisionW 지정된 도메인의 쿠키에 대한 결정을 검색합니다. (유니코드)
InternetGoOnline InternetGoOnline 함수(wininet.h)는 사용자에게 URL에 대한 연결을 시작할 수 있는 권한을 묻는 메시지를 표시합니다.
InternetGoOnlineA ANSI(InternetGoOnlineA) 함수(wininet.h)는 사용자에게 URL에 대한 연결을 시작할 수 있는 권한을 묻는 메시지를 표시합니다.
InternetGoOnlineW InternetGoOnlineW(유니코드) 함수(wininet.h)는 사용자에게 URL에 대한 연결을 시작할 수 있는 권한을 묻는 메시지를 표시합니다.
InternetHangUp InternetHangUp 함수(wininet.h)는 모뎀에 인터넷 연결을 끊도록 지시합니다.
InternetInitializeAutoProxyDll InternetInitializeAutoProxyDll 함수(wininet.h)는 InternetInitializeAutoProxyDll이라는 두 개의 WinINet 함수 중 하나입니다.

InternetLockRequestFile 사용 중인 파일에 잠금을 배치합니다.
InternetOpenA 애플리케이션의 WinINet 함수 사용을 초기화합니다. (ANSI)
InternetOpenUrlA 전체 FTP 또는 HTTP URL로 지정된 리소스를 엽니다. (ANSI)
InternetOpenUrlW 전체 FTP 또는 HTTP URL로 지정된 리소스를 엽니다. (유니코드)
InternetOpenW 애플리케이션의 WinINet 함수 사용을 초기화합니다. (유니코드)
InternetQueryDataAvailable 서버를 쿼리하여 사용 가능한 데이터의 양을 확인합니다.
InternetQueryOptionA 지정된 핸들에서 인터넷 옵션을 쿼리합니다. (ANSI)
InternetQueryOptionW 지정된 핸들에서 인터넷 옵션을 쿼리합니다. (유니코드)
InternetReadFile InternetOpenUrl, FtpOpenFile 또는 HttpOpenRequest 함수에서 연 핸들에서 데이터를 읽습니다.
InternetReadFileExA InternetOpenUrl 또는 HttpOpenRequest 함수에서 연 핸들에서 데이터를 읽습니다. (ANSI)
InternetReadFileExW InternetOpenUrl 또는 HttpOpenRequest 함수에서 연 핸들에서 데이터를 읽습니다. (유니코드)
InternetSetCookieA 지정된 URL과 연결된 쿠키를 만듭니다. (InternetSetCookieA)

InternetSetCookieEx2 지정된 URL과 연결된 쿠키를 만듭니다. (InternetSetCookieEx2)
InternetSetCookieExA InternetSetCookieEx 함수는 지정된 URL과 연결된 지정된 이름의 쿠키를 만듭니다. 이 함수는 타사 쿠키를 만들 수 있어 InternetSetCookie 함수와 다릅니다. (ANSI)
InternetSetCookieExW InternetSetCookieEx 함수는 지정된 URL과 연결된 지정된 이름의 쿠키를 만듭니다. 이 함수는 타사 쿠키를 만들 수 있어 InternetSetCookie 함수와 다릅니다. (유니코드)
InternetSetCookieW 지정된 URL과 연결된 쿠키를 만듭니다. (InternetSetCookieW)
InternetSetDialState InternetSetDialState 함수(wininet.h)는 지원되지 않으며 사용되지 않으며 사용해서는 안 됩니다.
InternetSetDialStateA ANSI(InternetSetDialStateA) 함수(wininet.h)는 지원되지 않으며 사용되지 않으며 사용해서는 안 됩니다.
InternetSetDialStateW InternetSetDialStateW(유니코드) 함수(wininet.h)는 지원되지 않으며 사용되지 않으며 사용해서는 안 됩니다.
InternetSetFilePointer InternetReadFile의 파일 위치를 설정합니다. 동기 호출입니다. 그러나 InternetReadFile에 대한 후속 호출은 캐시에서 데이터를 사용할 수 없으며 서버가 임의 액세스를 지원하지 않는 경우 보류 중인 데이터를 차단하거나 반환할 수 있습니다.
InternetSetOptionA 인터넷 옵션을 설정합니다. (ANSI)
InternetSetOptionExA 지원되지 않습니다. InternetSetOption 함수를 호출하는 스텁으로만 구현됩니다. InternetSetOptionEx에는 자체 기능이 없습니다. 현재는 이 함수를 사용하지 마세요. (ANSI)

InternetSetOptionExW 지원되지 않습니다. InternetSetOption 함수를 호출하는 스텁으로만 구현됩니다. InternetSetOptionEx에는 자체 기능이 없습니다. 현재는 이 함수를 사용하지 마세요. (유니코드)
InternetSetOptionW 인터넷 옵션을 설정합니다. (유니코드)
InternetSetPerSiteCookieDecisionA 지정된 도메인의 쿠키에 대한 결정을 설정합니다. (ANSI)
InternetSetPerSiteCookieDecisionW 지정된 도메인의 쿠키에 대한 결정을 설정합니다. (유니코드)
InternetSetStatusCallback 작업 중에 진행 상황으로 WinINet 함수가 호출할 수 있는 콜백 함수를 설정합니다.
InternetSetStatusCallbackA ANSI(InternetSetStatusCallbackA) 함수는 작업 중에 진행이 수행되면 WinINet 함수가 호출할 수 있는 콜백 함수를 설정합니다.
InternetSetStatusCallbackW InternetSetStatusCallbackW(유니코드) 함수는 WinINet 함수가 작업 중에 진행률로 호출할 수 있는 콜백 함수를 설정합니다.
InternetTimeFromSystemTime InternetTimeFromSystemTime 함수(wininet.h)는 HTTP 버전 1.0 사양에 따라 날짜 및 시간의 형식을 지정합니다.
InternetTimeFromSystemTimeA HTTP 버전 1.0 사양에 따라 날짜 및 시간의 형식을 지정합니다. (InternetTimeFromSystemTimeA)
InternetTimeFromSystemTimeW InternetTimeFromSystemTimeW(유니코드) 함수(wininet.h)는 HTTP 버전 1.0 사양에 따라 날짜 및 시간의 형식을 지정합니다.
InternetTimeToSystemTime InternetTimeToSystemTime 함수(wininet.h)는 HTTP 시간/날짜 문자열을 SYSTEMTIME 구조체로 변환합니다.

InternetTimeToSystemTimeA HTTP 시간/날짜 문자열을 SYSTEMTIME 구조체로 변환합니다. (InternetTimeToSystemTimeA)
InternetTimeToSystemTimeW InternetTimeToSystemTimeW(유니코드) 함수(wininet.h)는 HTTP 시간/날짜 문자열을 SYSTEMTIME 구조체로 변환합니다.
InternetUnlockRequestFile InternetLockRequestFile을 사용하여 잠긴 파일의 잠금을 해제합니다.
InternetWriteFile 열려 있는 인터넷 파일에 데이터를 씁니다.
PrivacyGetZonePreferenceW PrivacyGetZonePreferenceW(유니코드) 함수(wininet.h)는 지정된 URLZONE 및 PrivacyType에 대한 개인 정보 설정을 검색합니다.
PrivacySetZonePreferenceW PrivacySetZonePreferenceW(유니코드) 함수(wininet.h)는 지정된 URLZONE 및 PrivacyType에 대한 개인 정보 설정을 설정합니다.
ReadUrlCacheEntryStream RetrieveUrlCacheEntryStream 함수를 사용하여 열린 스트림에서 캐시된 데이터를 읽습니다.
ResumeSuspendedDownload ResumeSuspendedDownload 함수는 사용자 인터페이스 대화 상자에서 일시 중단된 요청을 다시 시작합니다.
RetrieveUrlCacheEntryFileA 지정된 URL과 연결된 캐시 항목 파일을 잠급 수 있습니다. (ANSI)
RetrieveUrlCacheEntryFileW 지정된 URL과 연결된 캐시 항목 파일을 잠급 수 있습니다. (유니코드)
RetrieveUrlCacheEntryStreamA 캐시 데이터에 액세스하는 가장 효율적이고 구현 독립적인 방법을 제공합니다. (ANSI)

RetrieveUrlCacheEntryStreamW 캐시 데이터에 액세스하는 가장 효율적이고 구현 독립적인 방법을 제공합니다. (유니코드)
SetUrlCacheEntryGroup SetUrlCacheEntryGroup 함수(wininet.h)는 캐시 그룹에 항목을 추가하거나 캐시 그룹에서 항목을 제거합니다.
SetUrlCacheEntryGroupA 캐시 그룹에 항목을 추가하거나 캐시 그룹에서 항목을 제거합니다. (SetUrlCacheEntryGroupA)
SetUrlCacheEntryGroupW SetUrlCacheEntryGroupW(유니코드) 함수(wininet.h)는 캐시 그룹에 항목을 추가하거나 캐시 그룹에서 항목을 제거합니다.
SetUrlCacheEntryInfoA INTERNET_CACHE_ENTRY_INFO 구조체의 지정된 멤버를 설정합니다. (ANSI)
SetUrlCacheEntryInfoW INTERNET_CACHE_ENTRY_INFO 구조체의 지정된 멤버를 설정합니다. (유니코드)
SetUrlCacheGroupAttributeA 지정된 캐시 그룹의 특성 정보를 설정합니다. (ANSI)
SetUrlCacheGroupAttributeW 지정된 캐시 그룹의 특성 정보를 설정합니다. (유니코드)
UnlockUrlCacheEntryFile UnlockUrlCacheEntryFile 함수(wininet.h)는 캐시에서 사용하기 위해 파일을 검색하는 동안 잠긴 캐시 항목의 잠금을 해제합니다.
UnlockUrlCacheEntryFileA 캐시에서 사용하기 위해 파일을 검색하는 동안 잠긴 캐시 항목의 잠금을 해제합니다. (UnlockUrlCacheEntryFileA)
UnlockUrlCacheEntryFileW UnlockUrlCacheEntryFileW(유니코드) 함수(wininet.h)는 캐시에서 사용하기 위해 파일을 검색하는 동안 잠긴 캐시 항목의 잠금을 해제합니다.

UnlockUrlCacheEntryStream RetrieveUrlCacheEntryStream 함수를 사용하여 검색된 스트림을 닫습니다.

콜백 함수

GOPHER_ATTRIBUTE_ENUMERATOR Gopher 서버에서 특성 정보를 처리하는 콜백 함수의 프로토타입입니다.
INTERNET_STATUS_CALLBACK 이 콜백 함수에 대한 포인터를 정의합니다.

구조체

AUTO_PROXY_SCRIPT_BUFFER AUTO_PROXY_SCRIPT_BUFFER 구조체는 InternetInitializeAutoProxyDll이 열리는 파일을 식별하는 대신 버퍼의 자동 프록시 스크립트를 InternetInitializeAutoProxyDll에 전달하는 데 사용됩니다.
AutoProxyHelperFunctions AutoProxyHelperFunctions 구조는 InternetInitializeAutoProxyDll에 전달할 수 있는 프록시 자동 구성 함수의 v 테이블을 만드는 데 사용됩니다.
AutoProxyHelperVtbl AutoProxyHelperVtbl 구조체는 PAC(프록시 AutoConfig) 도우미 함수에 대한 포인터의 v 테이블을 만듭니다. 폼의 사양 및 프록시 자동 구성 도우미 함수 사용에 대한 자세한 내용은 PAC(탐색기 프록시 자동 구성) 파일 형식 설명서를 참조하세요.
GOPHER_ATTRIBUTE_TYPE 개체에 대한 단일 Gopher 특성의 관련 정보를 포함합니다.
GOPHER_FIND_DATAA GopherFindFirstFile 및 InternetFindNextFile 함수에서 검색한 정보를 포함합니다. (ANSI)

GOPHER_FIND_DATAW GopherFindFirstFile 및 InternetFindNextFile 함수에서 검색한 정보를 포함합니다. (유니코드)
HTTP_VERSION_INFO 전역 HTTP 버전을 포함합니다.
INTERNET_ASYNC_RESULT 비동기 함수 호출의 결과를 포함합니다. 이 구조는 InternetStatusCallback과 함께 사용됩니다.
INTERNET_AUTH_NOTIFY_DATA 인증 요청에 대한 알림 데이터를 포함합니다.
INTERNET_BUFFERSA 데이터 및 헤더 정보를 모두 포함합니다. (ANSI)
INTERNET_BUFFERSW 데이터 및 헤더 정보를 모두 포함합니다. (유니코드)
INTERNET_CACHE_ENTRY_INFOA 인터넷 캐시의 항목에 대한 정보를 포함합니다. (ANSI)
INTERNET_CACHE_ENTRY_INFOW 인터넷 캐시의 항목에 대한 정보를 포함합니다. (유니코드)
INTERNET_CACHE_GROUP_INFOA 특정 캐시 그룹에 대한 정보를 포함합니다. (ANSI)
INTERNET_CACHE_GROUP_INFOW 특정 캐시 그룹에 대한 정보를 포함합니다. (유니코드)
INTERNET_CACHE_TIMESTAMPS 인터넷 캐시에 저장된 리소스에 대한 LastModified 및 만료 시간을 포함합니다.
INTERNET_CERTIFICATE_INFO 서버에서 반환된 인증서 정보를 포함합니다. 이 구조는 InternetQueryOption 함수에서 사용됩니다.

INTERNET_CONNECTED_INFO 전역 온라인/오프라인 상태를 설정하는 정보를 포함합니다.
INTERNET_COOKIE2 INTERNET_COOKIE2 구조에는 쿠키의 구성 요소가 포함됩니다. 이 구조는 InternetGetCookieEx2 및 InternetSetCookieEx2 함수와 함께 사용됩니다.
INTERNET_DIAGNOSTIC_SOCKET_INFO INTERNET_DIAGNOSTIC_SOCKET_INFO 구조체는 INTERNET_OPTION_DIAGNOSTIC_SOCKET_INFO 플래그가 HTTP 요청에 대한 핸들과 함께 전달될 때 InternetQueryOption 함수에 의해 반환됩니다.
INTERNET_PER_CONN_OPTION_LISTA 특정 인터넷 연결에 대한 옵션 목록을 포함합니다. (ANSI)
INTERNET_PER_CONN_OPTION_LISTW 특정 인터넷 연결에 대한 옵션 목록을 포함합니다. (유니코드)
INTERNET_PER_CONN_OPTIONA 옵션의 값을 포함합니다. (ANSI)
INTERNET_PER_CONN_OPTIONW 옵션의 값을 포함합니다. (유니코드)
INTERNET_PROXY_INFO InternetOpen 함수 호출에서 가져온 핸들에 대한 프록시 정보를 가져오거나 설정하기 위해 INTERNET_OPTION_PROXY 값과 함께 제공되는 정보를 포함합니다.
INTERNET_VERSION_INFO 서버의 HTTP 버전 번호를 포함합니다. 이 구조는 internetQueryOption 함수에 INTERNET_OPTION_VERSION 플래그를 전달할 때 사용됩니다.
InternetCookieHistory InternetCookieHistory 구조에는 쿠키 기록이 포함됩니다.
URL_COMPONENTSA URL의 구성 요소를 포함합니다. 이 구조는 InternetCrackUrl 및 InternetCreateUrl 함수와 함께 사용됩니다. (ANSI)

URL_COMPONENTSW URL의 구성 요소를 포함합니다. 이 구조는 InternetCrackUrl 및 InternetCreateUrl 함수와 함께 사용됩니다. (유니코드)

열거형

INTERNET_SCHEME URL_COMPONENTS 구조체의 nScheme 멤버와 함께 사용되는 플래그를 정의합니다.
InternetCookieState InternetCookieState 열거형은 쿠키의 상태를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winineti.h 헤더

아티클2023. 08. 18.

이 헤더는 Windows 인터넷에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 인터넷

wininet.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CreateMD5SSOHash CreateMD5SSOHash 함수(winineti.h)는 지정된 계정 또는 영역에 대한 기본 Microsoft Passport 암호를 가져오고 MD5 해시를 만들고 결과를 반환합니다.
CreateUrlCacheContainerA 지정된 캐시 경로에 지정된 이름, 캐시 접두사 및 컨테이너 형식에 따라 캐시 항목을 저장할 캐시 컨테이너를 만듭니다. (ANSI)
CreateUrlCacheContainerW 지정된 캐시 경로에 지정된 이름, 캐시 접두사 및 컨테이너 형식에 따라 캐시 항목을 저장할 캐시 컨테이너를 만듭니다. (유니코드)
DeleteUrlCacheContainerA 지정된 이름에 따라 캐시 컨테이너(캐시 항목 포함)를 삭제합니다. (ANSI)
DeleteUrlCacheContainerW 지정된 이름에 따라 캐시 컨테이너(캐시 항목 포함)를 삭제합니다. (유니코드)
DetectAutoProxyUrl DetectAutoProxyUrl 함수(winineti.h)는 WPAD 자동 프록시 스크립트의 위치를 확인하려고 시도합니다.
FreeUrlCacheSpaceA 캐시의 공간을 확보합니다. (ANSI)
FreeUrlCacheSpaceW

캐시의 공간을 확보합니다. (유니코드)
GetUrlCacheConfigInfoA 캐시 구성에 대한 정보를 검색합니다. (ANSI)
GetUrlCacheConfigInfoW 캐시 구성에 대한 정보를 검색합니다. (유니코드)
InternetAutodial InternetAutodial 함수(winineti.h)를 사용하면 모뎀이 기본 인터넷 연결에 자동으로 전화를 걸 수 있습니다.
InternetAutodialHangup InternetAutodialHangup 함수(winineti.h)는 자동 전화 접속 연결을 끊습니다.
InternetDial InternetDial 함수(winineti.h)는 모뎀을 사용하여 인터넷에 대한 연결을 시작합니다.
InternetDialA ANSI(InternetDialA) 함수(winineti.h)는 모뎀을 사용하여 인터넷에 대한 연결을 시작합니다.
InternetDialW InternetDialW(유니코드) 함수(winineti.h)는 모뎀을 사용하여 인터넷에 대한 연결을 시작합니다.
InternetEnumPerSiteCookieDecisionA InternetEnumPerSiteCookieDecisionA(ANSI) 함수(winineti.h)는 사이트별 쿠키 규정이 설정된 웹 사이트의 도메인 및 쿠키 설정을 검색합니다.
InternetEnumPerSiteCookieDecisionW InternetEnumPerSiteCookieDecisionW(유니코드) 함수(winineti.h)는 사이트별 쿠키 규정이 설정된 웹 사이트의 도메인 및 쿠키 설정을 검색합니다.
InternetGetConnectedState InternetGetConnectedState 함수(winineti.h)는 로컬 시스템의 연결된 상태를 검색합니다.
InternetGetConnectedStateEx InternetGetConnectedStateEx 함수(winineti.h)는 지정된 인터넷 연결의 연결된 상태를 검색합니다.

InternetGetConnectedStateExA INTERNETGetConnectedStateExA(ANSI) 함수는 지정된 인터넷 연결의 연결된 상태를 검색합니다. 이 API를 사용하지 않는 것이 좋습니다. 대신 INetworkListManager::GetConnectivity 메서드를 사용합니다.
InternetGetConnectedStateExW InternetGetConnectedStateExW(유니코드) 함수는 지정된 인터넷 연결의 연결된 상태를 검색합니다. 이 API를 사용하지 않는 것이 좋습니다. 대신 INetworkListManager::GetConnectivity 메서드를 사용합니다.
InternetGoOnline InternetGoOnline 함수(winineti.h)는 사용자에게 URL에 대한 연결을 시작할 수 있는 권한을 묻는 메시지를 표시합니다.
InternetGoOnlineA ANSI(InternetGoOnlineA) 함수(winineti.h)는 사용자에게 URL에 대한 연결을 시작할 수 있는 권한을 묻는 메시지를 표시합니다.
InternetGoOnlineW InternetGoOnlineW(유니코드) 함수(winineti.h)는 사용자에게 URL에 대한 연결을 시작할 수 있는 권한을 묻는 메시지를 표시합니다.
InternetHangUp InternetHangUp 함수(winineti.h)는 모뎀에 인터넷 연결을 끊도록 지시합니다.
InternetInitializeAutoProxyDll InternetInitializeAutoProxyDll이라는 두 개의 WinINet 함수가 있습니다. 첫 번째는 프록시 구성의 내부 상태를 새로 고치고 두 번째는 WinINet의 자동 프록시 지원의 일부입니다.
InternetSetDialState InternetSetDialState 함수(wininet.h)는 사용되지 않으며 지원되지 않으며 사용하지 마세요.
InternetSetDialStateA ANSI(InternetSetDialStateA) 함수(wininet.h)는 사용되지 않으며 지원되지 않으며 사용하지 마세요.
InternetSetDialStateW InternetSetDialStateW(유니코드) 함수(wininet.h)는 사용되지 않으며 지원되지 않습니다. 사용하지 마세요.

PrivacyGetZonePreferenceW PrivacyGetZonePreferenceW(유니코드) 함수(winineti.h)는 지정된 URLZONE 및 PrivacyType에 대한 개인 정보 설정을 검색합니다.
PrivacySetZonePreferenceW PrivacySetZonePreferenceW 함수(winineti.h)는 지정된 URLZONE 및 PrivacyType에 대한 개인 정보 설정을 설정합니다.

구조체

INTERNET_CACHE_CONFIG_INFOA 인터넷 캐시의 구성에 대한 정보를 포함합니다. (ANSI)
INTERNET_CACHE_CONFIG_INFOW 인터넷 캐시의 구성에 대한 정보를 포함합니다. (유니코드)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winioctl.h 헤더

아티클2023. 03. 08.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지
- 시스템 서비스

winioctl.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

IOCTLs

FSCTL_ALLOW_EXTENDED_DASD_IO 파티션 읽기 또는 쓰기 호출에서 I/O 경계 검사를 수행하지 않도록 파일 시스템 드라이버에 신호를 보냅니다.
FSCTL_CREATE_OR_GET_OBJECT_ID 지정된 파일 또는 디렉터리에 대한 개체 식별자를 검색합니다. 개체 식별자가 없으면 FSCTL_CREATE_OR_GET_OBJECT_ID를 사용하여 만듭니다.
FSCTL_CREATE_USN_JOURNAL 대상 볼륨에 USN(업데이트 시퀀스 번호) 변경 저널 스트림을 만들거나 기존 변경 저널 스트림을 수정합니다.
FSCTL_CSV_CONTROL CSV 제어 작업의 결과를 검색합니다.
FSCTL_CSV_QUERY_DOWN_LEVEL_FILE_SYSTEM_CHARACTERISTICS CSVFS가 프록시인 파일 시스템에 대한 정보를 검색합니다.
FSCTL_DELETE_OBJECT_ID 지정된 파일 또는 디렉터리에서 개체 식별자를 제거합니다.
FSCTL_DELETE_REPARSE_POINT 지정된 파일 또는 디렉터리에서 재분석 지점을 삭제합니다.

FSCTL_DELETE_USN_JOURNAL 볼륨에서 USN(업데이트 시퀀스 번호) 변경 저널을 삭제하거나 변경 저널 삭제 알림을 기다립니다.
FSCTL_DISMOUNT_VOLUME 볼륨이 현재 사용 중인지 여부에 관계없이 볼륨을 분리합니다. 자세한 내용은 주의 섹션을 참조하세요.
FSCTL_DUPLICATE_EXTENTS_TO_FILE 애플리케이션을 대신하여 파일 바이트 범위를 복사하도록 파일 시스템에 지시합니다.
FSCTL_ENUM_USN_DATA 지정된 두 경계 간에 USN(업데이트 시퀀스 번호) 데이터를 열거하여 MFT(마스터 파일 테이블) 레코드를 가져옵니다.
FSCTL_EXTEND_VOLUME 탑재된 볼륨의 크기를 늘립니다.
FSCTL_FILE_LEVEL_TRIM 스토리지 시스템에서 저장할 필요가 없는 지정된 파일 내의 범위를 나타냅니다.
FSCTL_FILESYSTEM_GET_STATISTICS 다양한 파일 시스템 성능 카운터에서 정보를 검색합니다.
FSCTL_FILESYSTEM_GET_STATISTICS_EX 다양한 파일 시스템 성능 카운터에서 정보를 검색합니다. 이 제어 코드에 대한 지원은 Windows 10 시작했습니다.
FSCTL_FIND_FILES_BY_SID 디렉터리에서 작성자 소유자가 지정된 SID와 일치하는 파일을 검색합니다.
FSCTL_GET_BOOT_AREA_INFO 볼륨의 부팅 섹터 위치를 검색합니다.
FSCTL_GET_COMPRESSION 파일 시스템이 스트림별 압축을 지원하는 볼륨에서 파일 또는 디렉터리의 현재 압축 상태를 검색합니다.

FSCTL_GET_INTEGRITY_INFORMATION ReFS 볼륨에서 파일 또는 디렉터리의 무결성 상태를 검색합니다. (FSCTL_GET_INTEGRITY_INFORMATION)
FSCTL_GET_NTFS_FILE_RECORD 사용 중이고 서수 값이 요청된 파일 참조 번호보다 작거나 같은 첫 번째 파일 레코드를 검색합니다.
FSCTL_GET_NTFS_VOLUME_DATA 지정된 NTFS 파일 시스템 볼륨에 대한 정보를 검색합니다.
FSCTL_GET_OBJECT_ID 지정된 파일 또는 디렉터리에 대한 개체 식별자를 검색합니다.
FSCTL_GET_REPAIR NTFS 파일 시스템의 자동 복구 메커니즘에 대한 정보를 검색합니다.
FSCTL_GET_REPARSE_POINT 지정된 핸들로 식별되는 파일 또는 디렉터리와 연결된 재분석 지점 데이터를 검색합니다.
FSCTL_GET_RETRIEVAL_POINTER_BASE 볼륨의 시작을 기준으로 파일 시스템의 첫 번째 LCN(논리 클러스터 번호)에 대한 섹터 오프셋을 반환합니다.
FSCTL_GET_RETRIEVAL_POINTERS 파일 핸들이 지정된 경우 디스트에서 특정 파일의 할당 및 위치를 설명하는 데이터 구조를 검색하거나 볼륨 핸들이 지정된 경우 볼륨에서 잘못된 클러스터의 위치를 검색합니다.
FSCTL_GET_VOLUME_BITMAP 볼륨에서 점유된 클러스터 및 사용 가능한 클러스터의 비트맵을 검색합니다.
FSCTL_INITIATE_REPAIR 단일 파일에서 자동 복구 주기를 시작하도록 NTFS 파일 시스템을 트리거합니다.
FSCTL_IS_CSV_FILE 파일이 CSVFS 볼륨에 저장되는지 여부를 확인하거나 네임스페이스 정보를 검색합니다. (FSCTL_IS_CSV_FILE)

FSCTL_IS_FILE_ON_CSV_VOLUME 파일이 CSVFS 볼륨에 저장되는지 여부를 확인하거나 네임스페이스 정보를 검색합니다. (FSCTL_IS_FILE_ON_CSV_VOLUME)
FSCTL_IS_VOLUME_MOUNTED 지정된 볼륨이 탑재되는지 또는 지정된 파일 또는 디렉터리가 탑재된 볼륨에 있는지 확인합니다.
FSCTL_IS_VOLUME_OWNED_BYCSVFS 볼륨이 CSVFS에 의해 잠겨 있는지 여부를 확인합니다.
FSCTL_LOCK_VOLUME 볼륨을 사용하고 있지 않은 경우 잠급니다.
FSCTL_LOOKUP_STREAM_FROM_CLUSTER NTFS 볼륨 또는 NTFS 볼륨의 파일에 대한 핸들이 지정된 경우 지정된 클러스터를 차지하는 스트림을 설명하는 데이터 구조 체인을 반환합니다.
FSCTL_MAKE_MEDIA_COMPATIBLE 미디어 ROM이 호환되도록 한 번 쓰기 미디어에서 열린 UDF 세션을 닫습니다.
FSCTL_MARK_HANDLE 지정된 파일 또는 디렉터리와 해당 변경 저널 레코드를 해당 파일 또는 디렉터리의 변경 내용에 대한 정보로 표시합니다.
FSCTL_MOVE_FILE 파일의 가상 클러스터 하나 이상을 하나의 논리적 클러스터에서 동일한 볼륨 내의 다른 클러스터로 재배치합니다. 이 작업은 조각 모음 중에 사용됩니다.
FSCTL_OPBATCH_ACK_CLOSE_PENDING 클라이언트 애플리케이션이 파일을 닫을 준비가 되었음을 서버에 알립니다.
FSCTL_OPLOCK_BREAK_ACK_NO_2 파일에 대한 편의적 잠금이 곧 중단된다는 알림에 응답합니다. 이 작업을 사용하여 파일에 대한 모든 편의적 잠금을 해제하고 파일을 열어 둡니다.
FSCTL_OPLOCK_BREAK_ACKNOWLEDGE 파일에 대한 배타적인 편의적 잠금이 곧 중단된다는 알림에 응답합니다. 이 작업을 사용하여 파일이 수준 2 편의적 잠금을 수신해야 함을 나타냅니다.

FSCTL_OPLOCK_BREAK_NOTIFY 호출 애플리케이션이 편의적 잠금 중단이 완료될 때까지 기다릴 수 있도록 합니다.
FSCTL_QUERY_ALLOCATED_RANGES 파일 또는 대체 스트림을 스캔하여 0이 아닌 데이터를 포함할 수 있는 범위를 찾습니다.
FSCTL_QUERY_FILE_SYSTEM_RECOGNITION 볼륨에 대한 파일 시스템 인식 정보를 쿼리합니다.
FSCTL_QUERY_ON_DISK_VOLUME_INFO UDF 관련 볼륨 정보를 요청합니다.
FSCTL_QUERY_REGION_INFO 데이터 계층화를 지원하는 볼륨에 대해 정의된 스토리지 계층 영역을 검색합니다.
FSCTL_QUERY_SPARING_INFO 볼륨의 결함 관리 속성을 검색합니다. UDF 파일 시스템에 사용됩니다.
FSCTL_QUERY_STORAGE_CLASSES 데이터 계층화를 지원하는 볼륨에 대해 정의된 스토리지 계층을 검색합니다.
FSCTL_QUERY_USN_JOURNAL 현재 USN(업데이트 시퀀스 번호) 변경 저널, 해당 레코드 및 용량에 대한 정보를 쿼리합니다.
FSCTL_READ_FILE_USN_DATA 지정된 파일 또는 디렉터리에 대한 USN(업데이트 시퀀스 번호) 변경 저널 정보를 검색합니다.
FSCTL_READ_FROM_PLEX 지정된 플렉스에서 읽습니다.
FSCTL_READ_USN_JOURNAL 지정된 두 USN 값 간에 USN(업데이트 시퀀스 번호) 변경 저널 레코드 세트를 검색합니다.
FSCTL_RECALL_FILE 계층적 스토리지 관리 소프트웨어인 원격 스토리지가 관리하는 스토리지 미디어에서 파일을 다시 호출합니다.

FSCTL_REPAIR_COPIES 사용할 적절한 복사본을 선택하여 데이터 손상을 복구합니다.
FSCTL_REQUEST_BATCH_OPLOCK 파일에 대한 일괄 편의적 잠금을 요청합니다.
FSCTL_REQUEST_FILTER_OPLOCK 파일에 대한 필터 편의적 잠금을 요청합니다.
FSCTL_REQUEST_OPLOCK 파일에 대한 편의적 잠금(oplock)을 요청하고 oplock 중단이 발생했음을 확인합니다.
FSCTL_REQUEST_OPLOCK_LEVEL_1 파일에 대해 수준 1 편의적 잠금을 요청합니다.
FSCTL_REQUEST_OPLOCK_LEVEL_2 파일에 대해 수준 2 편의적 잠금을 요청합니다.
FSCTL_SET_COMPRESSION 파일 시스템이 파일별 및 디렉터리별 압축을 지원하는 볼륨에서 파일 또는 디렉터리의 압축 상태를 설정합니다.
FSCTL_SET_DEFECT_MANAGEMENT 지정된 파일에 대한 소프트웨어 결함 관리 상태를 설정합니다. UDF 파일 시스템에 사용됩니다.
FSCTL_SET_INTEGRITY_INFORMATION ReFS 볼륨에서 파일 또는 디렉터리의 무결성 상태를 검색합니다. (FSCTL_SET_INTEGRITY_INFORMATION)
FSCTL_SET_OBJECT_ID 지정된 파일 또는 디렉터리에 대한 개체 식별자를 설정합니다.
FSCTL_SET_OBJECT_ID_EXTENDED 지정된 파일 또는 디렉터리에 대한 개체 식별자와 연결된 사용자 데이터를 수정합니다.
FSCTL_SET_REPAIR NTFS 파일 시스템의 자동 복구 기능 모드를 설정합니다.

FSCTL_SET_REPARSE_POINT 파일 또는 디렉터리에 재분석 지점을 설정합니다.
FSCTL_SET_SPARSE 표시된 파일을 스파스 또는 스파스 아님으로 표시합니다. 스파스 파일에서 큰 범위의 0에는 디스크 할당이 필요하지 않을 수 있습니다.
FSCTL_SET_ZERO_DATA 파일의 지정된 범위를 0으로 채웁니다.
FSCTL_SET_ZERO_ON_DEALLOCATION NTFS 파일 시스템 파일 핸들이 할당 취소될 때 클러스터가 0으로 채워져야 함을 나타냅니다.
FSCTL_SHRINK_VOLUME 축소 작업을 수행하기 위해 볼륨을 준비하거나, 축소 작업을 커밋하거나, 축소 작업을 종료해야 한다는 신호를 보냅니다.
FSCTL_TXFS_CREATE_MINIVERSION 지정한 파일에 새 miniversion을 만듭니다. 미니버전에서는 트랜잭션 중에 파일의 스냅샷을 참조할 수 있습니다. 트랜잭션이 커밋되거나 롤백되면 미니버전이 삭제됩니다.
FSCTL_TXFS_GET_METADATA_INFO 지정된 파일을 잠근 트랜잭션의 GUID 및 파일에 대한 TxF(트랜잭션 NTFS) 메타데이터를 검색합니다(파일이 잠긴 경우).
FSCTL_TXFS_GET_TRANSACTED_VERSION TXFS_GET_TRANSACTED_VERSION 구조체를 반환합니다. 구조체는 지정된 파일의 가장 최근에 커밋된 버전인 핸들의 버전 번호를 식별합니다.
FSCTL_TXFS_LIST_TRANSACTION_LOCKED_FILES 지정된 트랜잭션에 의해 현재 잠겨 있는 모든 파일의 목록을 반환합니다. 반환 값이 ERROR_MORE_DATA인 경우 이 호출 시 전체 파일 목록을 보관하는 데 필요한 버퍼의 길이를 반환합니다.
FSCTL_TXFS_LIST_TRANSACTIONS 지정된 리소스 관리자에 현재 관련된 모든 트랜잭션의 목록을 반환합니다.

FSCTL_TXFS_MODIFY_RM RM(보조 리소스 관리자)에 대한 로그 모드 및 로그 매개 변수 정보를 설정합니다.
FSCTL_TXFS_QUERY_RM_INFORMATION RM(리소스 관리자)에 대한 정보를 검색합니다.
FSCTL_TXFS_READ_BACKUP_INFORMATION 지정된 파일에 대한 TxF(트랜잭션 NTFS)별 정보를 반환합니다.
FSCTL_TXFS_SAVEPOINT_INFORMATION FSCTL_TXFS_SAVEPOINT_INFORMATION 제어 코드는 지정된 저장점으로의 설정, 지우기 및 롤백을 제어합니다.
FSCTL_TXFS_TRANSACTION_ACTIVE 스냅샷을 만들 때 연결된 볼륨에서 활성 상태인 트랜잭션이 있는지 여부를 나타내는 부울 값을 반환합니다. 이 호출은 읽기 전용 스냅샷 볼륨에만 유효합니다.
FSCTL_TXFS_WRITE_BACKUP_INFORMATION TxF(트랜잭션 NTFS)별 정보를 지정된 파일에 씁니다.
FSCTL_UNLOCK_VOLUME 볼륨의 잠금을 해제합니다.
FSCTL_USN_TRACK_MODIFIED_RANGES 대상 볼륨에서 USN(업데이트 시퀀스 번호) 변경 저널 스트림에 범위 추적 기능을 사용하도록 설정하거나 이미 활성화된 범위 추적 매개 변수를 수정합니다.
FSCTL_WAIT_FOR_REPAIR 지정된 복구가 완료되면 반환됩니다.
FSCTL_WRITE_USN_CLOSE_RECORD 입력 파일에 대한 USN(업데이트 시퀀스 번호) 변경 저널 스트림에 레코드를 생성합니다.
IOCTL_CHANGER_EXCHANGE_MEDIUM 원본 요소에서 하나의 대상으로 미디어 조각을 이동하고 원래 첫 번째 대상에 있는 미디어 부분을 두 번째 대상으로 이동합니다.

IOCTL_CHANGER_GET_ELEMENT_STATUS 모든 요소의 상태 또는 특정 형식의 지정된 수의 요소를 검색합니다.
IOCTL_CHANGER_GET_PARAMETERS 지정된 디바이스의 매개 변수를 검색합니다.
IOCTL_CHANGER_GET_PRODUCT_DATA 지정된 디바이스의 제품 데이터를 검색합니다.
IOCTL_CHANGER_GET_STATUS 지정된 디바이스의 현재 상태를 검색합니다.
IOCTL_CHANGER_INITIALIZE_ELEMENT_STATUS 모든 요소 또는 특정 형식의 지정된 요소의 상태를 초기화합니다.
IOCTL_CHANGER_MOVE_MEDIUM 미디어 조각을 대상으로 이동합니다.
IOCTL_CHANGER_QUERY_VOLUME_TAGS 지정된 요소에 대한 볼륨 태그 정보를 검색합니다.
IOCTL_CHANGER_REINITIALIZE_TRANSPORT 전송 요소를 물리적으로 다시 정정합니다. 재보정에는 전송을 홈 위치로 반환하는 작업이 포함될 수 있습니다.
IOCTL_CHANGER_SET_ACCESS 디바이스의 삽입/배출 포트, 도어 또는 키패드의 상태를 설정합니다.
IOCTL_CHANGER_SET_POSITION 체인저의 로봇 전송 메커니즘을 지정된 요소 주소로 설정합니다. 이렇게 하면 전송을 미리 배치하여 미디어를 이동하거나 교환할 수 있습니다.
IOCTL_DISK_CREATE_DISK CREATE_DISK 구조체의 정보를 사용하여 지정된 디스크 및 디스크 파티션 테이블을 초기화합니다.
IOCTL_DISK_DELETE_DRIVE_LAYOUT 디스크가 섹터 0에서 디스크 끝까지 포맷되도록 마스터 부팅 레코드에서 부팅 서명을 제거합니다.

IOCTL_DISK_FORMAT_TRACKS 플로피 디스크에서 지정된 연속 트랙 세트를 포맷합니다. 추가 매개 변수를 제공하려면 IOCTL_DISK_FORMAT_TRACKS_EXinstead 사용합니다.
IOCTL_DISK_FORMAT_TRACKS_EX 플로피 디스크에서 지정된 연속 트랙 세트를 포맷합니다.
IOCTL_DISK_GET_CACHE_INFORMATION 디스크 캐시 구성 데이터를 검색합니다.
IOCTL_DISK_GET_DISK_ATTRIBUTES 지정된 디스크 디바이스의 특성을 검색합니다.
IOCTL_DISK_GET_DRIVE_GEOMETRY 실제 디스크의 geometry:_type, 실린더 수, 실린더당 트랙 수, 트랙당 섹터 및 섹터당 바이트에 대한 정보를 검색합니다.
IOCTL_DISK_GET_DRIVE_GEOMETRY_EX 실제 디스크의 geometry:_type, 실린더 수, 실린더당 트랙 수, 트랙당 섹터 및 섹터당 바이트에 대한 확장 정보를 검색합니다.
IOCTL_DISK_GET_DRIVE_LAYOUT 디스크의 파티션 테이블에 있는 각 항목에 대한 정보를 검색합니다.
IOCTL_DISK_GET_DRIVE_LAYOUT_EX 디스크의 파티션 테이블에 있는 각 항목에 대한 확장 정보를 검색합니다.
IOCTL_DISK_GET_LENGTH_INFO 지정된 디스크, 볼륨 또는 파티션의 길이를 검색합니다.
IOCTL_DISK_GET_PARTITION_INFO 디스크 파티션의 형식, 크기 및 특성에 대한 정보를 검색합니다.
IOCTL_DISK_GET_PARTITION_INFO_EX 디스크 파티션의 형식, 크기 및 특성에 대한 확장 정보를 검색합니다.

IOCTL_DISK_GROW_PARTITION 지정된 파티션을 확대합니다.
IOCTL_DISK_IS_WRITABLE 지정한 디스크가 쓰기 가능한지 여부를 확인합니다.
IOCTL_DISK_PERFORMANCE 디스크 성능 정보를 제공하는 성능 카운터를 사용하도록 설정합니다.
IOCTL_DISK_PERFORMANCE_OFF 디스크 성능 정보를 제공하는 성능 카운터를 사용하지 않도록 설정합니다.
IOCTL_DISK_REASSIGN_BLOCKS 디스크 디바이스가 하나 이상의 블록을 예비 블록 풀에 매핑하도록 지시합니다. (IOCTL_DISK_REASSIGN_BLOCKS)
IOCTL_DISK_REASSIGN_BLOCKS_EX 디스크 디바이스가 하나 이상의 블록을 예비 블록 풀에 매핑하도록 지시합니다. (IOCTL_DISK_REASSIGN_BLOCKS_EX)
IOCTL_DISK_RESET_SNAPSHOT_INFO 디스크에서 VSS(볼륨 섀도 복사본 서비스) 하드웨어 기반 섀도 복사본("스냅샷"이라고도 함) 정보를 모두 지웁니다.
IOCTL_DISK_SET_CACHE_INFORMATION 디스크 구성 데이터를 설정합니다.
IOCTL_DISK_SET_DISK_ATTRIBUTES 지정된 디스크 디바이스의 특성을 설정합니다.
IOCTL_DISK_SET_DRIVE_LAYOUT 드라이브 레이아웃 및 파티션 정보 데이터에 지정된 대로 디스크를 분할합니다.
IOCTL_DISK_SET_DRIVE_LAYOUT_EX 지정된 드라이브 레이아웃 및 파티션 정보 데이터에 따라 디스크를 분할합니다.

IOCTL_DISK_SET_PARTITION_INFO 지정된 디스크 파티션에 대한 파티션 정보를 설정합니다.
IOCTL_DISK_SET_PARTITION_INFO_EX AT 및 EFI(Extensible Firmware Interface) 파티션에 대한 레이아웃 정보를 포함하여 지정된 디스크 파티션에 대한 파티션 정보를 설정합니다.
IOCTL_DISK_UPDATE_PROPERTIES 캐시된 파티션 테이블을 무효화하고 디바이스를 다시 열거합니다.
IOCTL_DISK_VERIFY 고정 디스크에서 지정된 익스텐트를 확인합니다.
IOCTL_SERIAL_LSRMST_INSERT 애플리케이션이 ReadFile 함수를 통해 획득하는 일반 데이터 스트림에 줄 상태 및 모뎀 상태 값의 배치를 사용하거나 사용하지 않도록 설정합니다.
IOCTL_STORAGE_CHECK_VERIFY 디바이스에 대한 미디어에 액세스할 수 있는지 여부를 결정합니다.
IOCTL_STORAGE_DEVICE_POWER_CAP Windows 애플리케이션은 이 제어 코드를 사용하여 스토리지 디바이스의 최대 작동 전력 소비 수준을 지정할 수 있습니다.
IOCTL_STORAGE_EJECT_MEDIA SCSI 디바이스에서 미디어를 배출합니다.
IOCTL_STORAGE_EJECTION_CONTROL 미디어를 꺼내는 메커니즘을 사용하거나 사용하지 않도록 설정합니다. 메커니즘을 사용하지 않도록 설정하면 드라이브가 잠깁니다.
IOCTL_STORAGE_FIRMWARE_ACTIVATE Windows 애플리케이션은 이 제어 코드를 사용하여 지정된 디바이스에서 펌웨어 이미지를 활성화할 수 있습니다.
IOCTL_STORAGE_FIRMWARE_DOWNLOAD Windows 애플리케이션은 이 제어 코드를 사용하여 대상 디바이스에 펌웨어 이미지를 다운로드할 수 있지만 활성화할 수는 없습니다.

IOCTL_STORAGE_FIRMWARE_GET_INFO Windows 애플리케이션은 이 제어 코드를 사용하여 스토리지 디바이스에서 자세한 펌웨어 정보를 쿼리할 수 있습니다.
IOCTL_STORAGE_GET_DEVICE_NUMBER 디바이스 유형, 디바이스 번호 및 분할 가능한 디바이스의 경우 디바이스의 파티션 번호를 검색합니다.
IOCTL_STORAGE_GET_HOTPLUG_INFO 지정된 디바이스의 핫플러그 구성을 검색합니다.
IOCTL_STORAGE_GET_MEDIA_SERIAL_NUMBER USB 디바이스의 일련 번호를 검색합니다.
IOCTL_STORAGE_GET_MEDIA_TYPES 디바이스에 대한 기하 도형 정보를 검색합니다. (IOCTL_STORAGE_GET_MEDIA_TYPES)
IOCTL_STORAGE_GET_MEDIA_TYPES_EX 디바이스에서 지원하는 미디어 유형에 대한 정보를 검색합니다.
IOCTL_STORAGE_LOAD_MEDIA 디바이스에 미디어를 로드합니다.
IOCTL_STORAGE_MANAGE_DATA_SET_ATTRIBUTES IOCTL_STORAGE_MANAGE_DATA_SET_ATTRIBUTES 제어 코드는 볼륨 관리자 및 스토리지 시스템 디바이스에 특성 정보를 전달합니다.
IOCTL_STORAGE_MCN_CONTROL 미디어 변경 알림을 사용하거나 사용하지 않도록 설정합니다. 미디어 변경 알림을 사용하지 않도록 설정하면 GUID_IO_MEDIA_ARRIVAL 및 GUID_IO_MEDIA_REMOVAL 이벤트가 방지됩니다.
IOCTL_STORAGE_MEDIA_REMOVAL 잠금 기능을 소유한 디바이스에 대해 미디어를 꺼내는 메커니즘을 사용하거나 사용하지 않도록 설정합니다.

IOCTL_STORAGE_PROTOCOL_COMMAND Windows 애플리케이션은 이 제어 코드를 사용하여 스토리지 디바이스 또는 어댑터의 속성을 반환할 수 있습니다. 요청은 디바이스에 대한 조회 데이터 또는 어댑터의 기능 및 제한 사항과 같이 검색할 정보의 종류를 나타냅니다.
IOCTL_STORAGE_QUERY_PROPERTY Windows 애플리케이션은 이 제어 코드를 사용하여 스토리지 디바이스 또는 어댑터의 속성을 반환할 수 있습니다.
IOCTL_STORAGE_READ_CAPACITY 디바이스에 대한 기하 도형 정보를 검색합니다. (IOCTL_STORAGE_READ_CAPACITY)
IOCTL_STORAGE_REINITIALIZE_MEDIA IOCTL_STORAGE_REINITIALIZE_MEDIA ioctl(winioctl.h)은 삭제 프로세스를 스토리지 디바이스로 오프로드합니다.
IOCTL_STORAGE_RPMB_COMMAND IOCTL_STORAGE_RPMB_COMMAND ioctl(winioctl.h)은 기본 스토리지 디바이스에 RPMB 명령을 보냅니다.
IOCTL_STORAGE_SET_HOTPLUG_INFO 지정된 디바이스의 핫플러그 구성을 설정합니다.
IOCTL_STORAGE_SET_TEMPERATURE_THRESHOLD Windows 애플리케이션은 이 제어 코드를 사용하여 디바이스의 온도 임계값을 설정할 수 있습니다(디바이스에서 지원하는 경우).
IOCTL_VOLUME_GET_GPT_ATTRIBUTES 볼륨의 특성을 검색합니다.
IOCTL_VOLUME_GET_VOLUME_DISK_EXTENTS 하나 이상의 디스크에서 지정된 볼륨의 물리적 위치를 검색합니다.
IOCTL_VOLUME_IS_CLUSTERED 지정된 볼륨이 클러스터되었는지 여부를 확인합니다.
IOCTL_VOLUME_OFFLINE 볼륨을 오프라인 상태로 전환합니다.

IOCTL_VOLUME_ONLINE 볼륨을 온라인 상태로 전환합니다.

구조체

BOOT_AREA_INFO FSCTL_GET_BOOT_AREA_INFO 제어 코드에 대한 출력을 포함합니다.
CHANGER_ELEMENT 변경자 요소를 나타냅니다.
CHANGER_ELEMENT_LIST 일반적으로 여러 요소의 상태를 가져오거나 초기화하는 등의 작업에 대해 단일 형식의 요소 범위를 나타냅니다.
CHANGER_ELEMENT_STATUS 지정된 요소의 상태를 나타냅니다. (CHANGER_ELEMENT_STATUS)
CHANGER_ELEMENT_STATUS_EX 지정된 요소의 상태를 나타냅니다. (CHANGER_ELEMENT_STATUS_EX)
CHANGER_EXCHANGE_MEDIUM IOCTL_CHANGER_EXCHANGE_MEDIUM 제어 코드에서 미디어 조각을 대상으로 이동하는 데 사용하는 정보와 원래 첫 번째 대상에 있는 미디어 조각을 두 번째 대상으로 이동하는 정보가 포함되어 있습니다.
CHANGER_INITIALIZE_ELEMENT_STATUS 모든 미디어 체인저 요소 또는 특정 형식의 지정된 요소의 상태를 나타냅니다.
CHANGER_MOVE_MEDIUM IOCTL_CHANGER_MOVE_MEDIUM 제어 코드가 미디어 조각을 대상으로 이동하는 데 사용하는 정보를 포함합니다.

CHANGER_PRODUCT_DATA 체인저 디바이스의 제품 데이터를 나타냅니다. IOCTL_CHANGER_GET_PRODUCT_DATA 제어 코드에서 사용됩니다.
CHANGER_READ_ELEMENT_STATUS IOCTL_CHANGER_GET_ELEMENT_STATUS 제어 코드가 상태를 검색할 요소를 결정하는 데 필요한 정보를 포함합니다.
CHANGER_SEND_VOLUME_TAG_INFORMATION IOCTL_CHANGER_QUERY_VOLUME_TAGS 제어 코드가 검색할 볼륨 정보를 결정하는 데 사용하는 정보를 포함합니다.
CHANGER_SET_ACCESS IOCTL_CHANGER_SET_ACCESS 제어 코드가 디바이스의 삽입/배출 포트, 도어 또는 키패드의 상태를 설정하는 데 필요한 정보를 포함합니다.
CHANGER_SET_POSITION 변경자의 로봇 전송 메커니즘을 지정된 요소 주소로 설정하기 위해 IOCTL_CHANGER_SET_POSITION 제어 코드에 필요한 정보를 포함합니다.
CLASS_MEDIA_CHANGE_CONTEXT 미디어 변경 이벤트와 관련된 정보를 포함합니다.
CREATE_DISK IOCTL_DISK_CREATE_DISK 제어 코드에서 GPT(GUID 파티션 테이블), MBR(마스터 부트레코드) 또는 원시 디스크를 초기화하는 데 사용하는 정보를 포함합니다.
CREATE_DISK_GPT IOCTL_DISK_CREATE_DISK 제어 코드에서 GPT(GUID 파티션 테이블) 디스크를 초기화하는 데 사용하는 정보를 포함합니다.
CREATE_DISK_MBR IOCTL_DISK_CREATE_DISK 제어 코드dptj MBR(마스터 부팅 레코드) 디스크를 초기화하는 데 사용하는 정보를 포함합니다.
CREATE_USN_JOURNAL_DATA USN(업데이트 시퀀스 번호) 변경 저널을 설명하는 정보를 포함합니다.

CSV_CONTROL_PARAM CSV 제어 작업의 형식을 나타냅니다.
CSV_IS_OWNED_BY_CSVFS 볼륨을 CSVFS에서 소유하는지 여부를 결정하는 FSCTL_IS_VOLUME_OWNED_BYCSVFS 제어 코드에 대한 출력을 포함합니다.
CSV_NAMESPACE_INFO 파일에 대한 네임스페이스 정보를 가져오는 FSCTL_IS_CSV_FILE 제어 코드에 대한 출력을 포함합니다.
CSV_QUERY_FILE_REVISION 스트림의 파일이 수정되었는지 여부에 대한 정보를 포함합니다.
CSV_QUERY_MDS_PATH CSV에서 MDS와 통신하는 데 사용하는 경로를 포함합니다.
CSV_QUERY_REDIRECT_STATE 스트림의 파일이 리디렉션되었는지 여부에 대한 정보를 포함합니다.
CSV_QUERY_VETO_FILE_DIRECT_IO_OUTPUT 볼륨이 리디렉션 모드인 이유에 대한 문제 해결 정보를 포함합니다.
DELETE_USN_JOURNAL_DATA FSCTL_DELETE_USN_JOURNAL 제어 코드를 사용한 USN(업데이트 시퀀스 번호) 변경 저널 삭제에 대한 정보를 포함합니다.
DEVICE_COPY_OFFLOAD_DESCRIPTOR 스토리지 디바이스에 대한 복사 오프로드 기능을 포함합니다.
DEVICE_DATA_SET_LB_PROVISIONING_STATE IOCTL_STORAGE_MANAGE_DATA_SET_ATTRIBUTES 제어 코드의 DeviceDsmAction_Allocation 동작에 대한 출력 구조입니다.
DEVICE_DATA_SET_RANGE IOCTL_STORAGE_MANAGE_DATA_SET_ATTRIBUTES 제어 코드와 함께 사용할 데이터 세트 범위 정보를 제공합니다.

DEVICE_DATA_SET_REPAIR_PARAMETERS 복구 작업에 대한 매개 변수를 지정합니다.
DEVICE_DSM_NOTIFICATION_PARAMETERS IOCTL_STORAGE_MANAGE_DATA_SET_ATTRIBUTES 제어 코드에 대한 DeviceDsmAction_Notification 작업에 대한 매개 변수를 포함합니다.
DEVICE_DSM_OFFLOAD_READ_PARAMETERS IOCTL_STORAGE_MANAGE_DATA_SET_ATTRIBUTES 제어 코드에 대한 DeviceDsmAction_OffloadRead 작업에 대한 매개 변수를 포함합니다.
DEVICE_DSM_OFFLOAD_WRITE_PARAMETERS 오프로드 쓰기 작업에 대한 매개 변수를 지정합니다.
DEVICE_LB_PROVISIONING_DESCRIPTOR 스토리지 디바이스에 대한 씬 프로비저닝 기능을 포함합니다.
DEVICE_MANAGE_DATA_SET_ATTRIBUTES IOCTL_STORAGE_MANAGE_DATA_SET_ATTRIBUTES 제어 코드에 대한 입력 구조입니다.
DEVICE_MANAGE_DATA_SET_ATTRIBUTES_OUTPUT IOCTL_STORAGE_MANAGE_DATA_SET_ATTRIBUTES 제어 코드의 출력 구조입니다.
DEVICE_MEDIA_INFO 디바이스에서 지원하는 미디어에 대한 정보를 제공합니다.
DEVICE_POWER_DESCRIPTOR DEVICE_POWER_DESCRIPTOR 구조는 스토리지 디바이스의 전원 기능을 설명합니다.
DEVICE_SEEK_PENALTY_DESCRIPTOR IOCTL_STORAGE_QUERY_PROPERTY 요청과 함께 디바이스에 대한 검색 페널티 설명자 데이터를 검색하는 데 사용됩니다.
DEVICE_TRIM_DESCRIPTOR IOCTL_STORAGE_QUERY_PROPERTY 요청과 함께 디바이스에 대한 트리밍 설명자 데이터를 검색하는 데 사용됩니다.

DEVICE_WRITE_AGGREGATION_DESCRIPTOR 시스템에서 사용하도록 예약되었습니다. (DEVICE_WRITE_AGGREGATION_DESCRIPTOR)
DISK_CACHE_INFORMATION 디스크 캐시에 대한 정보를 제공합니다.
DISK_DETECTION_INFO 검색된 드라이브 매개 변수를 포함합니다.
DISK_EX_INT13_INFO 확장된 Int13 드라이브 매개 변수를 포함합니다.
DISK_EXTENT 디스크 범위를 나타냅니다.
DISK_GEOMETRY 디스크 디바이스 및 미디어의 기하학을 설명합니다.
DISK_GEOMETRY_EX 디스크 디바이스 및 미디어의 확장된 기하학을 설명합니다.
DISK_GROW_PARTITION 파티션 크기를 늘리는 데 사용되는 정보를 포함합니다.
DISK_INT13_INFO 표준 Int13 드라이브 기하학 매개 변수를 포함합니다.
DISK_PARTITION_INFO 디스크 파티션 정보를 포함합니다.
DISK_PERFORMANCE 디스크 성능 정보를 제공합니다.
DRIVE_LAYOUT_INFORMATION 드라이브의 파티션에 대한 정보를 포함합니다.

DRIVE_LAYOUT_INFORMATION_EX 드라이브의 파티션에 대한 확장된 정보를 포함합니다.
DRIVE_LAYOUT_INFORMATION_GPT 드라이브의 GPT(GUID 파티션 테이블) 파티션에 대한 정보를 포함합니다.
DRIVE_LAYOUT_INFORMATION_MBR 드라이브의 MBR(마스터 부트 레코드) 파티션에 대한 정보를 제공합니다.
DUPLICATE_EXTENTS_DATA 블록 복제 작업을 수행하는 FSCTL_DUPLICATE_EXTENTS 제어 코드에 대한 매개 변수를 포함합니다.
EXFAT_STATISTICS exFAT 파일 시스템의 통계 정보를 포함합니다.
FAT_STATISTICS FAT 파일 시스템의 통계 정보를 포함합니다.
FILE_ALLOCATED_RANGE_BUFFER 파일에서 바이트 범위를 나타냅니다.
FILE_LEVEL_TRIM FSCTL_FILE_LEVEL_TRIM 제어 코드에 대한 입력으로 사용됩니다.
FILE_LEVEL_TRIM_OUTPUT FSCTL_FILE_LEVEL_TRIM 제어 코드에 대한 출력으로 사용됩니다.
FILE_LEVEL_TRIM_RANGE 잘라낼 파일의 범위를 지정합니다.
FILE_MAKE_COMPATIBLE_BUFFER 현재 세션을 닫을 디스크를 지정합니다. 이 제어 코드는 UDF 파일 시스템에 사용됩니다. 이 구조체는 FSCTL_MAKE_MEDIA_COMPATIBLE을 호출할 때 입력에 사용됩니다.
FILE_OBJECTID_BUFFER 개체 식별자 및 개체 식별자와 연결된 사용자 정의 메타데이터를 포함합니다.

FILE_QUERY_ON_DISK_VOL_INFO_BUFFER FSCTL_QUERY_ON_DISK_VOLUME_INFO 호출에서 볼륨 정보를 수신합니다.
FILE_QUERY_SPARING_BUFFER 결함 관리 속성을 포함합니다.
FILE_SET_DEFECT_MGMT_BUFFER 설정할 결함 관리 상태를 지정합니다.
FILE_SET_SPARSE_BUFFER 설정할 스파스 상태를 지정합니다.
FILE_STORAGE_TIER 볼륨을 기준으로 스토리지 계층의 식별자를 나타냅니다.
FILE_STORAGE_TIER_REGION 단일 스토리지 계층 영역을 설명합니다.
FILE_SYSTEM_RECOGNITION_INFORMATION FSCTL_QUERY_FILE_SYSTEM_RECOGNITION 제어 코드를 통해 가져온 파일 시스템 인식 정보를 포함합니다.
FILE_ZERO_DATA_INFORMATION 0으로 설정할 파일의 범위를 포함합니다.
FILESYSTEM_STATISTICS 파일 시스템의 통계 정보를 포함합니다.
FILESYSTEM_STATISTICS_EX 파일 시스템의 통계 정보를 포함합니다. 이 구조체에 대한 지원은 Windows 10 시작했습니다.
FIND_BY_SID_DATA FSCTL_FIND_FILES_BY_SID 제어 코드에 대한 데이터를 포함합니다.
FIND_BY_SID_OUTPUT 파일 이름을 나타냅니다.

FORMAT_EX_PARAMETERS 인접한 디스크 트랙 집합의 서식을 지정하는 데 사용되는 정보를 포함합니다. IOCTL_DISK_FORMAT_TRACKS_EX 제어 코드에서 사용됩니다.
FORMAT_PARAMETERS 인접한 디스크 트랙 집합의 서식을 지정하는 데 사용되는 정보를 포함합니다.
FSCTL_GET_INTEGRITY_INFORMATION_BUFFER 파일 또는 디렉터리에 대한 무결성 정보를 포함합니다.
FSCTL_QUERY_REGION_INFO_INPUT 특정 볼륨에 대한 스토리지 스택의 스토리지 계층 영역을 포함합니다.
FSCTL_QUERY_REGION_INFO_OUTPUT 하나 이상의 영역에 대한 정보를 포함합니다.
FSCTL_QUERY_STORAGE_CLASSES_OUTPUT 특정 볼륨의 모든 계층에 대한 정보를 포함합니다.
FSCTL_SET_INTEGRITY_INFORMATION_BUFFER FSCTL_SET_INTEGRITY_INFORMATION 제어 코드를 사용하여 전달된 입력 버퍼입니다.
GET_CHANGER_PARAMETERS 변경자의 매개 변수를 나타냅니다.
GET_DISK_ATTRIBUTES 디스크 디바이스의 특성을 포함합니다.
GET_LENGTH_INFORMATION IOCTL_DISK_GET_LENGTH_INFO 제어 코드에서 사용하는 디스크, 볼륨 또는 파티션 길이 정보를 포함합니다.
GET_MEDIA_TYPES 디바이스에서 지원하는 미디어 유형에 대한 정보를 포함합니다.
LOOKUP_STREAM_FROM_CLUSTER_ENTRY FSCTL_LOOKUP_STREAM_FROM_CLUSTER 제어 코드에서 반환됩니다.

LOOKUP_STREAM_FROM_CLUSTER_INPUT FSCTL_LOOKUP_STREAM_FROM_CLUSTER 제어 코드에 입력으로 전달됩니다.
LOOKUP_STREAM_FROM_CLUSTER_OUTPUT FSCTL_LOOKUP_STREAM_FROM_CLUSTER 제어 코드에서 출력으로 수신됩니다.
MARK_HANDLE_INFO 지정된 파일 또는 디렉터리를 표시하는 데 사용되는 정보 및 변경 내용에 대한 데이터가 있는 USN(업데이트 시퀀스 번호) 변경 저널 레코드를 포함합니다.
MARK_HANDLE_INFO32 지정된 파일 또는 디렉터리를 표시하는 데 사용되는 정보 및 변경 내용에 대한 데이터가 있는 USN(업데이트 시퀀스 번호) 변경 저널 레코드를 포함합니다.
MFT_ENUM_DATA_V0 USN(업데이트 시퀀스 번호) 변경 저널 레코드 열거의 경계 및 시작 위치를 정의하는 정보를 포함합니다.
MFT_ENUM_DATA_V1 ReFS 볼륨에 대한 USN(업데이트 시퀀스 번호) 변경 저널 레코드 열거의 경계 및 시작 위치를 정의하는 정보를 포함합니다.
MOVE_FILE_DATA FSCTL_MOVE_FILE 제어 코드의 입력 데이터를 포함합니다.
NTFS_EXTENDED_VOLUME_DATA 볼륨 데이터를 나타냅니다. N
NTFS_FILE_RECORD_INPUT_BUFFER FSCTL_GET_NTFS_FILE_RECORD 제어 코드에 대한 데이터를 포함합니다.
NTFS_FILE_RECORD_OUTPUT_BUFFER FSCTL_GET_NTFS_FILE_RECORD 제어 코드에서 출력 데이터를 수신합니다.
NTFS_STATISTICS NTFS 파일 시스템의 통계 정보를 포함합니다.

NTFS_STATISTICS_EX NTFS 파일 시스템의 통계 정보를 포함합니다. 이 구조체에 대한 지원은 Windows 10 시작했습니다.
NTFS_VOLUME_DATA_BUFFER 볼륨 데이터를 나타냅니다. (NTFS_VOLUME_DATA_BUFFER)
PARTITION_INFORMATION 디스크 파티션에 대한 정보를 포함합니다.
PARTITION_INFORMATION_EX 표준 AT 스타일 MBR(마스터 부트 레코드) 및 EFI(Extensible Firmware Interface) 디스크에 대한 파티션 정보를 포함합니다.
PARTITION_INFORMATION_GPT GPT(GUID 파티션 테이블) 파티션 정보를 포함합니다.
PARTITION_INFORMATION_MBR MBR(마스터 부트 레코드) 디스크와 관련된 파티션 정보를 포함합니다.
PLEX_READ_DATA_REQUEST 수행할 읽기 작업의 범위와 읽을 플렉스를 나타냅니다.
PREVENT_MEDIA_REMOVAL 이동식 미디어 잠금 데이터를 제공합니다. IOCTL_STORAGE_MEDIA_REMOVAL 제어 코드에서 사용됩니다.
READ_ELEMENT_ADDRESS_INFO 볼륨 태그 정보를 나타냅니다. IOCTL_CHANGER_QUERY_VOLUME_TAGS 제어 코드에서 사용됩니다.
READ_FILE_USN_DATA 애플리케이션에서 지원하는 USN(업데이트 시퀀스 번호) 변경 저널의 버전을 지정합니다.
READ_USN_JOURNAL_DATA_V0 호출 프로세스에 반환할 USN(업데이트 시퀀스 번호) 변경 저널 레코드 세트를 정의하는 정보를 포함합니다. (READ_USN_JOURNAL_DATA_V0)

READ_USN_JOURNAL_DATA_V1 호출 프로세스로 돌아갈 USN(업데이트 시퀀스 번호) 변경 저널 레코드 집합을 정의하는 정보를 포함합니다. R
REASSIGN_BLOCKS 디스크 블록 재할당 데이터를 포함합니다. (REASSIGN_BLOCKS)
REASSIGN_BLOCKS_EX 디스크 블록 재할당 데이터를 포함합니다. (REASSIGN_BLOCKS_EX)
REPAIR_COPIES_INPUT FSCTL_REPAIR_COPIES 제어 코드에 대한 입력 구조입니다.
REPAIR_COPIES_OUTPUT FSCTL_REPAIR_COPIES 제어 코드에서 반환된 복사본 복구 작업의 출력을 포함합니다.
REQUEST_OPLOCK_INPUT_BUFFER 편의적 잠금(oplock)을 요청하거나 FSCTL_REQUEST_OPLOCK 제어 코드로 oplock 중단을 확인하는 정보를 포함합니다.
REQUEST_OPLOCK_OUTPUT_BUFFER FSCTL_REQUEST_OPLOCK 제어 코드에서 반환된 편의적 잠금(oplock) 정보를 포함합니다.
RETRIEVAL_POINTER_BASE FSCTL_GET_RETRIEVAL_POINTER_BASE 제어 코드에 대한 출력을 포함합니다.
RETRIEVAL_POINTERS_BUFFER FSCTL_GET_RETRIEVAL_POINTERS 제어 코드에 대한 출력을 포함합니다.
SET_DISK_ATTRIBUTES 디스크 디바이스에서 설정할 특성을 지정합니다.
SET_PARTITION_INFORMATION 디스크 파티션의 형식을 설정하는 데 사용되는 정보를 포함합니다.
SHRINK_VOLUME_INFORMATION 수행할 볼륨 축소 작업을 지정합니다.

STARTING_LCN_INPUT_BUFFER FSCTL_GET_VOLUME_BITMAP 제어 코드에 대한 시작 LCN을 포함합니다.
STARTING_VCN_INPUT_BUFFER FSCTL_GET_RETRIEVAL_POINTERS 제어 코드에 대한 시작 VCN을 포함합니다.
STORAGE_ACCESS_ALIGNMENT_DESCRIPTOR IOCTL_STORAGE_QUERY_PROPERTY 제어 코드와 함께 사용하여 디바이스에 대한 스토리지 액세스 맞춤 설명자 데이터를 검색합니다.
STORAGE_ADAPTER_DESCRIPTOR IOCTL_STORAGE_QUERY_PROPERTY 제어 코드와 함께 사용하여 디바이스에 대한 스토리지 어댑터 설명자 데이터를 검색합니다.
STORAGE_ADAPTER_SERIAL_NUMBER STORAGE_PROPERTY_ID 정의된 StorageAdapterSerialNumberProperty에 대한 어댑터 일련 번호의 NULL로 종료된 유니코드 문자열입니다.
STORAGE_DESCRIPTOR_HEADER IOCTL_STORAGE_QUERY_PROPERTY 제어 코드와 함께 사용하여 스토리지 디바이스 또는 어댑터의 속성을 검색합니다.
STORAGE_DEVICE_ATTRIBUTES_DESCRIPTOR 다음에 사용하도록 예약됩니다. (STORAGE_DEVICE_ATTRIBUTES_DESCRIPTOR)
STORAGE_DEVICE_DESCRIPTOR IOCTL_STORAGE_QUERY_PROPERTY 제어 코드와 함께 사용하여 디바이스에 대한 스토리지 디바이스 설명자 데이터를 검색합니다.
STORAGE_DEVICE_ID_DESCRIPTOR IOCTL_STORAGE_QUERY_PROPERTY 제어 코드 요청과 함께 사용하여 디바이스에 대한 디바이스 ID 설명자 데이터를 검색합니다.
STORAGE_DEVICE_IO_CAPABILITY_DESCRIPTOR STORAGE_PROPERTY_ID 정의된 StorageDeviceIoCapabilityProperty에 대한 출력 버퍼입니다.

STORAGE_DEVICE_NUMBER 디바이스에 대한 정보를 포함합니다. 이 구조체는 IOCTL_STORAGE_GET_DEVICE_NUMBER 제어 코드에서 사용됩니다.
STORAGE_DEVICE_POWER_CAP 이 구조체는 IOCTL_STORAGE_DEVICE_POWER_CAP 대한 입력 및 출력 버퍼로 사용됩니다.
STORAGE_DEVICE_RESILIENCY_DESCRIPTOR 시스템에서 사용하도록 예약되었습니다. (STORAGE_DEVICE_RESILIENCY_DESCRIPTOR)
STORAGE_HOTPLUG_INFO 디바이스의 핫플러그 정보에 대한 정보를 제공합니다.
STORAGE_HW_FIRMWARE_ACTIVATE 이 구조체에는 활성화할 다운로드한 펌웨어에 대한 정보가 포함되어 있습니다.
STORAGE_HW_FIRMWARE_DOWNLOAD 이 구조에는 대상에 다운로드할 펌웨어 이미지 페이로드가 포함되어 있습니다.
STORAGE_MEDIUM_PRODUCT_TYPE_DESCRIPTOR IOCTL_STORAGE_QUERY_PROPERTY 요청과 함께 사용되어 스토리지 디바이스의 제품 유형을 설명합니다.
STORAGE_MINIPORT_DESCRIPTOR 시스템에서 사용하도록 예약되었습니다. (STORAGE_MINIPORT_DESCRIPTOR)
STORAGE_OFFLOAD_READ_OUTPUT IOCTL_STORAGE_MANAGE_DATA_SET_ATTRIBUTES 제어 코드의 DeviceDsmAction_OffloadRead 동작에 대한 출력 구조입니다.
STORAGE_OFFLOAD_TOKEN 읽기 및 쓰기 작업을 오프로드하는 데 사용되는 파일의 일부를 나타내는 데 사용되는 토큰입니다.
STORAGE_OFFLOAD_WRITE_OUTPUT IOCTL_STORAGE_MANAGE_DATA_SET_ATTRIBUTES 제어 코드의 DeviceDsmAction_OffloadWrite 작업에 대한 출력 구조입니다.

STORAGE_PHYSICAL_ADAPTER_DATA 실제 스토리지 어댑터를 설명합니다.
STORAGE_PHYSICAL_DEVICE_DATA 실제 스토리지 디바이스를 설명합니다.
STORAGE_PHYSICAL_NODE_DATA 스토리지 노드의 물리적 디바이스 데이터를 지정합니다.
STORAGE_PHYSICAL_TOPOLOGY_DESCRIPTOR STORAGE_PHYSICAL_TOPOLOGY_DESCRIPTOR 구조체는 IOCTL_STORAGE_QUERY_PROPERTY 요청에서 반환된 쿼리 결과 구조체 중 하나입니다.
STORAGE_PROPERTY_QUERY IOCTL_STORAGE_QUERY_PROPERTY 제어 코드에 전달된 입력 버퍼로 검색할 스토리지 디바이스 또는 어댑터의 속성을 나타냅니다.
STORAGE_PROTOCOL_COMMAND 이 구조체는 통과 메커니즘을 사용하여 IOCTL_STORAGE_PROTOCOL_COMMAND를 통해 공급업체별 명령을 스토리지 디바이스에 발급할 때 입력 버퍼로 사용됩니다.
STORAGE_PROTOCOL_DATA_DESCRIPTOR 이 구조체는 스토리지 디바이스 또는 어댑터에서 프로토콜별 데이터를 반환하기 위해 IOCTL_STORAGE_QUERY_PROPERTY와 함께 사용됩니다.
STORAGE_PROTOCOL_SPECIFIC_DATA IOCTL_STORAGE_QUERY_PROPERTY 요청의 입력 및 출력 버퍼에 제공되는 프로토콜별 디바이스 데이터를 설명합니다.
STORAGE_RPMB_DATA_FRAME STORAGE_RPMB_DATA_FRAME 구조체(winioctl.h)를 사용하면 애플리케이션이 IOCTL_STORAGE_QUERY_PROPERTY 데이터를 사용하여 RPMB 프레임을 만들 수 있습니다.
STORAGE_RPMB_DESCRIPTOR STORAGE_RPMB_DESCRIPTOR 구조(winioctl.h)는 디바이스에 RPMB 및 RPMB에서 지원하는 최대 페이로드 크기가 포함되어 있는지 여부를 설명합니다.

STORAGE_SPEC_VERSION 스토리지 사양 버전입니다.
STORAGE_TEMPERATURE_DATA_DESCRIPTOR 이 구조체는 스토리지 디바이스 또는 어댑터에서 온도 데이터를 반환하기 위해 IOCTL_STORAGE_QUERY_PROPERTY와 함께 사용됩니다.
STORAGE_TEMPERATURE_INFO 디바이스 온도 데이터를 설명합니다. IOCTL_STORAGE_QUERY_PROPERTY 요청으로 온도 데이터를 쿼리할 때 STORAGE_TEMPERATURE_DATA_DESCRIPTOR의 일부로 반환됩니다.
STORAGE_TEMPERATURE_THRESHOLD 이 구조체는 IOCTL_STORAGE_SET_TEMPERATURE_THRESHOLD를 통해 스토리지 디바이스의 온도 상한 또는 하한 임계값을 설정하는 데 사용됩니다.
STORAGE_WRITE_CACHE_PROPERTY IOCTL_STORAGE_QUERY_PROPERTY 제어 코드와 함께 사용하여 디바이스의 쓰기 캐시 속성에 대한 정보를 검색합니다.
TXFS_CREATE_MINIVERSION_INFO FSCTL_TXFS_CREATE_MINIVERSION에 의해 만들어진 미니버전에 대한 버전 정보를 포함합니다.
TXFS_GET_METADATA_INFO_OUT 생성된 미니버전에 대한 버전 정보를 포함합니다.
TXFS_GET_TRANSACTED_VERSION 지정된 파일의 기본 및 최신 버전에 대한 정보를 포함합니다.
TXFS_LIST_TRANSACTION_LOCKED_FILES 트랜잭션된 기록기에 의해 잠긴 파일의 목록을 포함합니다.
TXFS_LIST_TRANSACTION_LOCKED_FILES_ENTRY 잠긴 트랜잭션에 대한 정보를 포함합니다.
TXFS_LIST_TRANSACTIONS 트랜잭션 목록을 포함합니다.

TXFS_LIST_TRANSACTIONS_ENTRY 트랜잭션에 대한 정보를 포함합니다.
TXFS_MODIFY_RM 보조 리소스 관리자에 대한 로그 매개 변수 및 로깅 모드를 수정할 때 필요한 정보를 포함합니다.
TXFS_QUERY_RM_INFORMATION 리소스 관리자(RM)에 대한 정보를 포함합니다.
TXFS_READ_BACKUP_INFORMATION_OUT 트랜잭션 NTFS(TxF) 특정 구조체를 포함합니다. 이 정보는 TXFS_WRITE_BACKUP_INFORMATION을 호출할 때만 사용해야 합니다. (TXFS_READ_BACKUP_INFORMATION_OUT)
TXFS_SAVEPOINT_INFORMATION FSCTL_TXFS_SAVEPOINT_INFORMATION 구조체는 수행할 작업과 트랜잭션을 지정합니다.
TXFS_TRANSACTION_ACTIVE_INFO 스냅샷을 만들 때 트랜잭션이 활성 상태인지를 나타내는 플래그를 포함합니다.
TXFS_WRITE_BACKUP_INFORMATION 트랜잭션 NTFS(TxF) 특정 구조체를 포함합니다. 이 정보는 TXFS_WRITE_BACKUP_INFORMATION을 호출할 때만 사용해야 합니다. (TXFS_WRITE_BACKUP_INFORMATION)
USN_JOURNAL_DATA_V0 USN(업데이트 시퀀스 번호) 변경 저널, 해당 레코드 및 용량을 나타냅니다. (USN_JOURNAL_DATA_V0)
USN_JOURNAL_DATA_V1 USN(업데이트 시퀀스 번호) 변경 저널, 해당 레코드 및 용량을 나타냅니다. U
USN_JOURNAL_DATA_V2 USN(업데이트 시퀀스 번호) 변경 저널, 해당 레코드 및 용량을 나타냅니다. 이 구조는 FSCTL_QUERY_USN_JOURNAL 제어 코드의 출력 버퍼입니다.
USN_RANGE_TRACK_OUTPUT FSCTL_USN_TRACK_MODIFIED_RANGES 제어 코드에서 반환된 USN(업데이트 시퀀스 번호)을 포함합니다.

USN_RECORD_COMMON_HEADER USN_RECORD_V2, USN_RECORD_V3 및 USN_RECORD_V4에서 공통되는 USN(업데이트 시퀀스 번호) 공통 헤더에 대한 정보를 포함합니다.
USN_RECORD_EXTENT USN(업데이트 시퀀스 번호) 레코드 범위에 대한 오프셋과 길이를 포함합니다.
USN_RECORD_V2 USN(업데이트 시퀀스 번호) 변경 저널 버전 2.0 레코드에 대한 정보를 포함합니다.
USN_RECORD_V3 USN(업데이트 시퀀스 번호) 변경 저널 버전 3.0 레코드에 대한 정보를 포함합니다.
USN_RECORD_V4 USN(업데이트 시퀀스 번호) 변경 저널 버전 4.0 레코드에 대한 정보를 포함합니다. 버전 2.0 및 3.0 레코드는 각각 USN_RECORD_V2(USN_RECORD라고도 함) 및 USN_RECORD_V3 구조로 정의됩니다.
USN_TRACK_MODIFIED_RANGES FSCTL_USN_TRACK_MODIFIED_RANGES 제어 코드를 사용하는 USN(업데이트 시퀀스 번호) 변경 저널의 범위 추적 매개 변수에 대한 정보를 포함합니다.
VERIFY_INFORMATION 디스크 익스텐트 확인에 사용되는 정보를 포함합니다.
VOLUME_BITMAP_BUFFER 디스크를 차지하고 있으며 사용 가능한 클러스터를 나타냅니다.
VOLUME_DISK_EXTENTS 디스크에서의 실제 위치를 나타냅니다.
VOLUME_GET_GPT_ATTRIBUTES_INFORMATION IOCTL_VOLUME_GET_GPT_ATTRIBUTES 제어 코드를 사용하여 가져온 볼륨 특성을 포함합니다.

열거형

CSV_CONTROL_OP FSCTL_CSV_CONTROL 제어 코드와 함께 사용할 CSV 제어 작업의 유형을 지정합니다.
ELEMENT_TYPE 변경자 디바이스의 요소 유형을 지정합니다.
FILE_STORAGE_TIER_MEDIA_TYPE 스토리지 미디어 유형을 지정합니다.
MEDIA_TYPE 다양한 형태의 디바이스 미디어를 나타냅니다.
PARTITION_STYLE 파티션 형식을 나타냅니다.
STORAGE_BUS_TYPE 다양한 유형의 스토리지 버스를 지정합니다.
STORAGE_COMPONENT_HEALTH_STATUS 스토리지 구성 요소의 상태를 지정합니다.
STORAGE_DEVICE_FORM_FACTOR 디바이스의 폼 팩터를 지정합니다.
STORAGE_DEVICE_POWER_CAP_UNITS 최대 전원 임계값의 단위입니다.
STORAGE_MEDIA_TYPE 다양한 유형의 스토리지 미디어를 지정합니다.
STORAGE_PORT_CODE_SET 시스템에서 사용하도록 예약되었습니다. (STORAGE_PORT_CODE_SET)
STORAGE_PROPERTY_ID 스토리지 디바이스 또는 어댑터의 속성을 검색하기 위해 IOCTL_STORAGE_QUERY_PROPERTY 요청에 입력으로 전달된 STORAGE_PROPERTY_QUERY 구조체의 PropertyId 멤버에 가능한 값을 열거합니다.

STORAGE_PROTOCOL_ATA_DATA_TYPE ATA 프로토콜 데이터 형식입니다.
STORAGE_PROTOCOL_NVME_DATA_TYPE IOCTL_STORAGE_QUERY_PROPERTY 요청 중에 쿼리할 NVMe 프로토콜별 데이터의 형식을 설명합니다.
STORAGE_PROTOCOL_TYPE 스토리지 디바이스의 프로토콜을 지정합니다.
STORAGE_QUERY_TYPE IOCTL_STORAGE_QUERY_PROPERTY 제어 코드에 전달된 STORAGE_PROPERTY_QUERY 구조체에서 스토리지 디바이스 또는 어댑터의 속성에 대해 반환되는 정보를 나타내는 데 사용됩니다.
WRITE_CACHE_CHANGE 디바이스의 쓰기 캐시 기능을 변경할 수 있는지 여부를 나타냅니다.
WRITE_CACHE_ENABLE 쓰기 캐시의 활성화 여부를 나타냅니다.
WRITE_CACHE_TYPE 캐시 유형을 지정합니다.
WRITE_THROUGH 스토리지 디바이스가 write-through 캐싱을 지원하는지 여부를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winldap.h 헤더

아티클2023. 03. 08.

이 헤더는 Lightweight Directory Access Protocol에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Lightweight Directory Access Protocol

winldap.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

ber_bvfree ber_bvfree 함수(winldap.h)는 BER(기본 인코딩 규칙)에 따라 인코딩되는 임의의 이진 데이터를 나타내는 berval 구조를 해제합니다.
cldap_open cldap_open 함수(winldap.h)는 연결이 없는 UDP(사용자 데이터그램 프로토콜) 서비스를 통해 LDAP 서버와 세션을 설정합니다.
cldap_openA 연결 없는 UDP(사용자 데이터그램 프로토콜) 서비스를 통해 LDAP 서버로 세션을 설정합니다. (cldap_openA)
cldap_openW cldap_openW(유니코드) 함수(winldap.h)는 연결되지 않은 UDP(사용자 데이터그램 프로토콜) 서비스를 통해 LDAP 서버와 세션을 설정합니다.
ldap_abandon 클라이언트는 ldap_abandon 호출하여 in-process 비동기 LDAP 호출을 취소합니다.
ldap_add ldap_add 함수(winldap.h)는 디렉터리 트리에 비동기 추가 작업을 시작합니다.
ldap_add_ext ldap_add_ext(winldap.h) 함수는 트리에 비동기 추가 작업을 시작합니다.

ldap_add_ext_s ldap_add_ext_s 함수(winldap.h)는 트리에 동기 추가 작업을 시작합니다.
ldap_add_ext_sA ldap_add_ext_s 함수는 트리에 동기 추가 작업을 시작합니다. 추가 작업이 성공하려면 추가된 항목의 부모가 있어야 하거나 부모가 비어 있어야 합니다(루트의 고유 이름과 같음). (ldap_add_ext_sA)
ldap_add_ext_sW ldap_add_ext_sW(유니코드) 함수(winldap.h)는 트리에 동기 추가 작업을 시작합니다.
ldap_add_extA ldap_add_ext 함수는 트리에 비동기 추가 작업을 시작합니다. 추가된 항목의 부모가 있어야 합니다. 또는 추가 작업이 성공하려면 부모가 비어 있어야 합니다(루트의 고유 이름과 같음). (ldap_add_extA)
ldap_add_extW ldap_add_extW(유니코드) 함수(winldap.h)는 트리에 비동기 추가 작업을 시작합니다.
ldap_add_s ldap_add_s 함수(winldap.h)는 트리에 항목을 추가하는 동기 추가 작업을 시작합니다.
ldap_add_sA ldap_add_s 함수는 트리에 항목을 추가하는 동기 추가 작업을 시작합니다. 추가되는 항목의 부모가 이미 있거나 추가 작업이 성공하려면 부모가 비어 있어야 합니다(루트 고유 이름과 같음). (ldap_add_sA)
ldap_add_sW ldap_add_sW(유니코드) 함수(winldap.h)는 트리에 항목을 추가하는 동기 추가 작업을 시작합니다.
ldap_addA 디렉터리 트리에 비동기 추가 작업을 시작합니다. (ldap_addA)
ldap_addW ldap_addW(유니코드) 함수(winldap.h)는 디렉터리 트리에 비동기 추가 작업을 시작합니다.
ldap_bind ldap_bind 함수(winldap.h)는 LDAP 서버를 사용하여 클라이언트를 비동기적으로 인증합니다.

ldap_bind_s ldap_bind_s 함수(winldap.h)는 LDAP 서버에 대한 클라이언트를 동기적으로 인증합니다.
ldap_bind_sA 클라이언트를 LDAP 서버에 동기적으로 인증합니다. (ldap_bind_sA)
ldap_bind_sW ldap_bind_sW(유니코드) 함수(winldap.h)는 LDAP 서버에 대한 클라이언트를 동기적으로 인증합니다.
ldap_bindA LDAP 서버를 사용하여 클라이언트를 비동기적으로 인증합니다. (ldap_bindA)
ldap_bindW ldap_bindW(유니코드) 함수(winldap.h)는 LDAP 서버를 사용하여 클라이언트를 비동기적으로 인증합니다.
ldap_check_filterA ldap_check_filter 함수는 필터 구문을 확인하는 데 사용됩니다. (ANSI)
ldap_check_filterW ldap_check_filter 함수는 필터 구문을 확인하는 데 사용됩니다. (유니코드)
ldap_cleanup 경고 ldap_cleanup 함수는 DLL 언로드 시 예기치 않은 동작을 일으킬 수 있으므로 wldap32.dll 동적으로 로드하고 언로드할 때 리소스를 안전하게 정리할 수 없습니다. 이 때문에 라이브러리의 언로드 시 리소스 누수가 발생할 수 있습니다. 따라서 ldap_cleanup 사용하는 것은 권장되지 않으며 사용자 고유의 위험에 처해 있습니다. .
ldap_close_extended_op ldap_close_extended_op 함수는 ldap_extended_operation 호출하여 만들어진 요청을 종료합니다.
ldap_compare ldap_compare 함수(winldap.h)는 지정된 항목의 특성에 알려진 값이 있는지 여부를 결정합니다.
ldap_compare_ext ldap_compare_ext 함수(winldap.h)는 지정된 항목에 대한 특성이 알려진 값을 보유하는지 여부를 결정합니다.

ldap_compare_ext_s ldap_compare_ext_s 함수(winldap.h)는 지정된 항목에 대한 특성이 알려진 값을 보유하는지 여부를 결정합니다.
ldap_compare_ext_sA ldap_compare_ext_s 함수를 사용하여 지정된 항목의 특성에 알려진 값이 있는지 확인합니다. (ldap_compare_ext_sA)
ldap_compare_ext_sW ldap_compare_ext_sW(유니코드) 함수(winldap.h)는 지정된 항목에 대한 특성이 알려진 값을 보유하는지 여부를 결정합니다.
ldap_compare_extA ldap_compare_ext 함수를 사용하여 지정된 항목의 특성에 알려진 값이 있는지 확인합니다. (ldap_compare_extA)
ldap_compare_extW ldap_compare_extW(유니코드) 함수(winldap.h)는 지정된 항목에 대한 특성이 알려진 값을 보유하는지 여부를 결정합니다.
ldap_compare_s ldap_compare_s 함수(winldap.h)는 지정된 항목에 대한 특성이 알려진 값을 보유하는지 여부를 결정합니다.
ldap_compare_sA ldap_compare_s 함수를 사용하여 지정된 항목의 특성에 알려진 값이 있는지 여부를 확인합니다. (ldap_compare_sA)
ldap_compare_sW ldap_compare_sW(유니코드) 함수(winldap.h)는 지정된 항목의 특성에 알려진 값이 있는지 여부를 결정합니다.
ldap_compareA ldap_compare 함수를 사용하여 지정된 항목의 특성에 알려진 값이 있는지 여부를 확인합니다. (ldap_compareA)
ldap_compareW ldap_compareW(유니코드) 함수(winldap.h)는 지정된 항목의 특성에 알려진 값이 있는지 여부를 결정합니다.

ldap_conn_from_msg 특정 메시지에 대한 LDAP 세션 핸들(연결 포인터)을 반환합니다.
ldap_connect ldap_connect 함수는 서버와의 연결을 설정합니다.
ldap_control_free ldap_control_free 함수(winldap.h)는 LDAPControl 구조를 해제합니다.
ldap_control_freeA ldap_control_free 함수는 LDAPControl 구조를 해제합니다. (ldap_control_freeA)
ldap_control_freeW ldap_control_freeW(유니코드) 함수(winldap.h)는 LDAPControl 구조를 해제합니다.
ldap_controls_free ldap_controls_free 함수(winldap.h)는 LDAPControl 구조체의 배열을 해제합니다.
ldap_controls_freeA LDAPControl 구조체의 배열을 해제합니다. (ldap_controls_freeA)
ldap_controls_freeW ldap_controls_freeW(유니코드) 함수(winldap.h)는 LDAPControl 구조의 배열을 해제합니다.
ldap_count_entries ldap_count_entries 함수는 서버가 반환한 검색 항목 수를 계산합니다.
ldap_count_references ldap_count_references 함수는 검색 요청에 대한 응답으로 서버에서 반환한 하위 참조 수를 계산합니다.
ldap_count_values ldap_count_values 함수(winldap.h)는 목록의 값 수를 계산합니다.
ldap_count_values_len 목록의 값 수를 계산합니다.

ldap_count_valuesA ldap_count_values 함수는 목록의 값 수를 계산합니다. (ldap_count_valuesA)
ldap_count_valuesW ldap_count_valuesW(유니코드) 함수(winldap.h)는 목록의 값 수를 계산합니다.
ldap_create_page_control ldap_create_page_control 함수(winldap.h)는 페이징 결과에 대한 기본 컨트롤을 만듭니다.
ldap_create_page_controlA ldap_create_page_control 함수를 사용하여 페이징 결과에 대한 기본 컨트롤을 만듭니다. 컨트롤에 대한 지원은 LDAP 3에서 유효하지만 페이지 컨트롤이 지원되는지 여부는 특정 서버에 종속됩니다. (ldap_create_page_controlA)
ldap_create_page_controlW ldap_create_page_controlW(유니코드) 함수(winldap.h)는 페이징 결과에 대한 기본 컨트롤을 만듭니다.
ldap_create_sort_control ldap_create_sort_control 함수(winldap.h)는 정렬 키 목록의 형식을 검색 컨트롤로 지정하는 데 사용됩니다.
ldap_create_sort_controlA ldap_create_sort_control 함수는 정렬 키 목록의 형식을 검색 컨트롤로 지정하는 데 사용됩니다. 컨트롤에 대한 지원은 LDAP 3에서 유효하지만 정렬 컨트롤이 지원되는지 여부는 특정 서버에 종속됩니다. (ldap_create_sort_controlA)
ldap_create_sort_controlW ldap_create_sort_controlW(유니코드) 함수(winldap.h)는 정렬 키 목록을 검색 컨트롤로 서식을 지정하는 데 사용됩니다.
ldap_create_vlv_controlA ldap_create_vlv_control 함수는 서버에서 요청 컨트롤(LDAP_CONTROL_VLVREQUEST)을 만드는 데 사용됩니다. (ANSI)
ldap_create_vlv_controlW ldap_create_vlv_control 함수는 서버에서 요청 컨트롤(LDAP_CONTROL_VLVREQUEST)을 만드는 데 사용됩니다. (유니코드)

ldap_delete ldap_delete 함수(winldap.h)는 디렉터리 트리에서 항목을 삭제합니다.
ldap_delete_ext ldap_delete_ext 함수(winldap.h)는 디렉터리 트리에서 리프 항목을 제거하는 확장 루틴입니다.
ldap_delete_ext_s ldap_delete_ext_s 함수(winldap.h)는 디렉터리 트리에서 리프 항목을 제거하는 동기 작업을 수행하는 확장 루틴입니다.
ldap_delete_ext_sA ldap_delete_ext_s 함수는 디렉터리 트리에서 리프 항목을 제거하는 동기 작업을 수행하는 확장 루틴입니다. (ldap_delete_ext_sA)
ldap_delete_ext_sW ldap_delete_ext_sW(유니코드) 함수(winldap.h)는 디렉터리 트리에서 리프 항목을 제거하는 동기 작업을 수행하는 확장 루틴입니다.
ldap_delete_extA ldap_delete_ext 함수는 디렉터리 트리에서 리프 항목을 제거하는 확장 루틴입니다. (ldap_delete_extA)
ldap_delete_extW ldap_delete_extW(유니코드) 함수(winldap.h)는 디렉터리 트리에서 리프 항목을 제거하는 확장 루틴입니다.
ldap_delete_s ldap_delete_s 함수(winldap.h)는 디렉터리 트리에서 리프 항목을 제거하는 동기 작업입니다.
ldap_delete_sA ldap_delete_s 함수는 디렉터리 트리에서 리프 항목을 제거하는 동기 연산입니다. (ldap_delete_sA)
ldap_delete_sW ldap_delete_sW(유니코드) 함수(winldap.h)는 디렉터리 트리에서 리프 항목을 제거하는 동기 작업입니다.
ldap_deleteA ldap_delete 함수는 디렉터리 트리에서 항목을 삭제합니다. (ldap_deleteA)

ldap_deleteW ldap_deleteW(유니코드) 함수(winldap.h)는 디렉터리 트리에서 항목을 삭제합니다.
ldap_dn2ufn ldap_dn2ufn 함수(winldap.h)는 고유 이름을 사용자에게 친숙한 형식으로 변환합니다.
ldap_dn2ufnA 고유 이름을 사용자에게 친숙한 형식으로 변환합니다. (ldap_dn2ufnA)
ldap_dn2ufnW ldap_dn2ufnW(유니코드) 함수(winldap.h)는 고유 이름을 사용자에게 친숙한 형식으로 변환합니다.
ldap_encode_sort_control ldap_encode_sort_control 함수는 정렬 키 목록의 형식을 검색 컨트롤로 지정합니다. 이 함수는 사용되지 않으며 대신 ldap_create_sort_control 함수를 사용합니다.
ldap_encode_sort_controlA ldap_encode_sort_control 함수는 정렬 키 목록의 형식을 검색 컨트롤로 지정합니다. 이 함수는 사용되지 않습니다. 대신 ldap_create_sort_control 사용합니다. (ldap_encode_sort_controlA)
ldap_encode_sort_controlW ldap_encode_sort_controlW(유니코드) 함수는 정렬 키 목록의 형식을 검색 컨트롤로 지정합니다. 이 함수는 사용되지 않으며 대신 ldap_create_sort_controlW(유니코드) 함수를 사용합니다.
ldap_err2string ldap_err2string 함수(winldap.h)는 숫자 LDAP 오류 코드를 오류를 설명하는 null로 끝나는 문자열로 변환합니다.
ldap_err2stringA 숫자 LDAP 오류 코드를 오류를 설명하는 null로 끝나는 문자열로 변환합니다. (ldap_err2stringA)
ldap_err2stringW ldap_err2stringW(유니코드) 함수(winldap.h)는 숫자 LDAP 오류 코드를 오류를 설명하는 null로 끝나는 문자열로 변환합니다.
ldap_escape_filter_element ldap_escape_filter_element 함수(winldap.h)는 필터 요소를 검색 필터에서 안전하게 전달할 수 있는 null로 끝나는 문자열로 변환합니다.

ldap_escape_filter_elementA ldap_escape_filter_element 함수는 필터 요소를 검색 필터에서 안전하게 전달할 수 있는 null로 끝나는 문자열로 변환합니다. (ldap_escape_filter_elementA)
ldap_escape_filter_elementW ldap_escape_filter_elementW(유니코드) 함수(winldap.h)는 필터 요소를 검색 필터에서 안전하게 전달할 수 있는 null로 끝나는 문자열로 변환합니다.
ldap_explode_dn ldap_explode_dn 함수(winldap.h)는 항목 이름을 구성 요소 부분으로 나눕니다.
ldap_explode_dnA ldap_explode_dn 함수는 항목 이름을 구성 요소 부분으로 구분합니다. (ldap_explode_dnA)
ldap_explode_dnW ldap_explode_dnW(유니코드) 함수(winldap.h)는 항목 이름을 구성 요소 부분으로 나눕니다.
ldap_extended_operation ldap_extended_operation 함수(winldap.h)를 사용하면 확장된 LDAP 작업을 서버에 전달할 수 있습니다.
ldap_extended_operation_sA 확장된 LDAP 작업을 서버에 전달하는 데 사용됩니다. (ANSI)
ldap_extended_operation_sW 확장된 LDAP 작업을 서버에 전달하는 데 사용됩니다. (유니코드)
ldap_extended_operationA ldap_extended_operation 함수를 사용하면 확장된 LDAP 작업을 서버에 전달할 수 있습니다. (ldap_extended_operationA)
ldap_extended_operationW ldap_extended_operationW(유니코드) 함수(winldap.h)를 사용하면 확장된 LDAP 작업을 서버에 전달할 수 있습니다.
ldap_first_attribute ldap_first_attribute 함수(winldap.h)는 첫 번째 특성을 반환합니다.

ldap_first_attributeA 첫 번째 특성을 반환합니다. (ldap_first_attributeA)
ldap_first_attributeW ldap_first_attributeW(유니코드) 함수(winldap.h)는 첫 번째 특성을 반환합니다.
ldap_first_entry ldap_first_entry 함수는 메시지의 첫 번째 항목을 반환합니다.
ldap_first_reference 메시지에서 첫 번째 참조를 반환합니다.
ldap_free_controls ldap_free_controls 함수(winldap.h)는 LDAPControl 구조의 배열을 해제하는 사용되지 않는 함수입니다.
ldap_free_controlsA LDAPControl 구조의 배열을 해제하는 사용되지 않는 함수입니다. (ldap_free_controlsA)
ldap_free_controlsW ldap_free_controlsW(유니코드) 함수(winldap.h)는 LDAPControl 구조의 배열을 해제하는 사용되지 않는 함수입니다.
ldap_get_dn ldap_get_dn 함수(winldap.h)는 지정된 항목의 고유 이름을 검색합니다.
ldap_get_dnA ldap_get_dn 함수는 지정된 항목의 고유 이름을 검색합니다. (ldap_get_dnA)
ldap_get_dnW ldap_get_dnW(유니코드) 함수(winldap.h)는 지정된 항목의 고유 이름을 검색합니다.
ldap_get_next_page 비동기 페이징 검색 결과 시퀀스의 다음 페이지를 반환합니다.
ldap_get_next_page_s 동기 페이징된 검색 결과 시퀀스의 다음 페이지를 반환합니다.

ldap_get_option ldap_get_option 함수(winldap.h)는 세션 전체 매개 변수의 현재 값을 검색합니다.
ldap_get_optionW ldap_get_optionW(유니코드) 함수(winldap.h)는 세션 전체 매개 변수의 현재 값을 검색합니다.
ldap_get_paged_count 서버가 검색을 위해 반환한 페이징된 결과 수를 기록합니다.
ldap_get_values ldap_get_values 함수(winldap.h)는 지정된 특성의 값 목록을 검색합니다.
ldap_get_values_len ldap_get_values_len 함수(winldap.h)는 지정된 특성의 값 목록을 검색합니다.
ldap_get_values_lenA ldap_get_values_len 함수는 지정된 특성의 값 목록을 검색합니다. (ldap_get_values_lenA)
ldap_get_values_lenW ldap_get_values_lenW(유니코드) 함수(winldap.h)는 지정된 특성에 대한 값 목록을 검색합니다.
ldap_get_valuesA ldap_get_values 함수는 지정된 특성의 값 목록을 검색합니다. (ldap_get_valuesA)
ldap_get_valuesW ldap_get_valuesW(유니코드) 함수(winldap.h)는 지정된 특성의 값 목록을 검색합니다.
ldap_init ldap_init 함수(winldap.h)는 LDAP 서버를 사용하여 세션을 초기화합니다.
ldap_initA LDAP 서버를 사용하여 세션을 초기화합니다. (ldap_initA)
ldap_initW ldap_initW(유니코드) 함수(winldap.h)는 LDAP 서버를 사용하여 세션을 초기화합니다.

ldap_memfree ldap_memfree 함수(winldap.h)는 LDAP 힙에서 할당된 메모리를 해제합니다.
ldap_memfreeA LDAP 힙에서 할당된 메모리를 해제합니다. (ldap_memfreeA)
ldap_memfreeW ldap_memfreeW(유니코드) 함수(winldap.h)는 LDAP 힙에서 할당된 메모리를 해제합니다.
ldap_modify ldap_modify 함수(winldap.h)는 기존 항목을 수정하는 비동기 작업을 시작합니다.
ldap_modify_ext ldap_modify_ext 함수는 기존 항목을 변경합니다. (ldap_modify_extW)
ldap_modify_ext_s ldap_modify_ext_s 함수(winldap.h)는 기존 항목을 수정하는 비동기 작업을 시작합니다.
ldap_modify_ext_sA ldap_modify_ext_s 함수는 기존 항목을 변경합니다. (ldap_modify_ext_sA)
ldap_modify_ext_sW ldap_modify_ext_sW(유니코드) 함수(winldap.h)는 기존 항목을 수정하는 비동기 작업을 시작합니다.
ldap_modify_extA ldap_modify_ext 함수는 기존 항목을 변경합니다. (ldap_modify_extA)
ldap_modify_extW ldap_modify_extW(유니코드) 함수(winldap.h)는 기존 항목을 수정하는 비동기 작업을 시작합니다.
ldap_modify_s ldap_modify_s 함수(winldap.h)는 기존 항목을 수정하는 비동기 작업을 시작합니다.
ldap_modify_sA ldap_modify_s 함수는 기존 항목을 변경합니다. (ldap_modify_sA)

ldap_modify_sW ldap_modify_sW(유니코드) 함수(winldap.h)는 기존 항목을 수정하는 비동기 작업을 시작합니다.
ldap_modifyA ldap_modify 함수는 기존 항목을 변경합니다. (ldap_modifyA)
ldap_modifyW ldap_modifyW(유니코드) 함수(winldap.h)는 기존 항목을 수정하는 비동기 작업을 시작합니다.
ldap_modrdn ldap_modrdn 함수(winldap.h)는 LDAP 항목의 상대적인 고유 이름을 변경합니다.
ldap_modrdn_s ldap_modrdn_s 함수(winldap.h)는 LDAP 항목의 상대적인 고유 이름을 변경합니다.
ldap_modrdn_sA LDAP 항목의 상대 고유 이름을 변경합니다. (ldap_modrdn_sA)
ldap_modrdn_sW ldap_modrdn_sW(유니코드) 함수(winldap.h)는 LDAP 항목의 상대적인 고유 이름을 변경합니다.
ldap_modrdn2 ldap_modrdn2 함수(winldap.h)는 LDAP 항목의 상대적인 고유 이름을 변경합니다.
ldap_modrdn2_s ldap_modrdn2_s 함수(winldap.h)는 LDAP 항목의 상대적인 고유 이름을 변경합니다.
ldap_modrdn2_sA ldap_modrdn2_s 함수는 LDAP 항목의 상대적인 고유 이름을 변경합니다. (ldap_modrdn2_sA)
ldap_modrdn2_sW ldap_modrdn2_sW(유니코드) 함수(winldap.h)는 LDAP 항목의 상대적인 고유 이름을 변경합니다.
ldap_modrdn2A ldap_modrdn2 함수는 LDAP 항목의 상대적인 고유 이름을 변경합니다. (ldap_modrdn2A)

ldap_modrdn2W ldap_modrdn2W(유니코드) 함수(winldap.h)는 LDAP 항목의 상대적인 고유 이름을 변경합니다.
ldap_modrdnA ldap_modrdn 함수는 LDAP 항목의 상대적인 고유 이름을 변경합니다. (ldap_modrdnA)
ldap_modrdnW ldap_modrdnW(유니코드) 함수(winldap.h)는 LDAP 항목의 상대적인 고유 이름을 변경합니다.
ldap_msgfree ldap_msgfree 함수는 이전 호출에서 얻은 결과를 ldap_result 또는 동기 검색 루틴 중 하나로 해제합니다.
ldap_next_attribute ldap_next_attribute 함수(winldap.h)는 다음 특성을 반환합니다.
ldap_next_attributeA 다음 특성을 반환합니다. (ldap_next_attributeA)
ldap_next_attributeW ldap_next_attributeW(유니코드) 함수(winldap.h)는 다음 특성을 반환합니다.
ldap_next_entry ldap_next_entry 함수는 검색 결과 체인에서 항목을 검색합니다.
ldap_next_reference 검색 결과 체인에서 참조를 검색합니다.
ldap_open ldap_open 함수는 연결 블록을 만들고 초기화한 다음 LDAP 서버에 대한 연결을 엽니다. 대신 ldap_init 함수를 사용하지 않는 것이 좋습니다.
ldap_openA ldap_open 사용하지 않는 것이 좋습니다. (ldap_openA)

ldap_openW ldap_openW(유니코드) 함수는 연결 블록을 만들고 초기화한 다음 LDAP 서버에 대한 연결을 엽니다. 대신 ldap_initW(유니코드) 함수를 사용하지 않는 것이 좋습니다.
ldap_parse_extended_resultA ldap_parse_extended_result LDAP 확장 작업의 결과를 구문 분석합니다. (ANSI)
ldap_parse_extended_resultW ldap_parse_extended_result LDAP 확장 작업의 결과를 구문 분석합니다. (유니코드)
ldap_parse_page_control ldap_parse_page_control 함수(winldap.h)는 검색 결과를 페이지로 구문 분석합니다.
ldap_parse_page_controlA ldap_parse_page_control 검색 결과를 페이지로 구문 분석합니다. (ldap_parse_page_controlA)
ldap_parse_page_controlW ldap_parse_page_controlW(유니코드) 함수(winldap.h)는 검색 결과를 페이지로 구문 분석합니다.
ldap_parse_reference ldap_parse_reference 함수(winldap.h)는 검색 응답 메시지에서 하위 조회 목록을 반환합니다.
ldap_parse_referenceA ldap_parse_reference 함수는 검색 응답 메시지에서 하위 조회 목록을 반환합니다. (ldap_parse_referenceA)
ldap_parse_referenceW ldap_parse_referenceW(유니코드) 함수(winldap.h)는 검색 응답 메시지에서 하위 조회 목록을 반환합니다.
ldap_parse_result ldap_parse_result 함수(winldap.h)는 서버의 응답을 구문 분석하고 적절한 필드를 반환합니다.
ldap_parse_resultA ldap_parse_result 함수는 서버의 응답을 구문 분석하고 적절한 필드를 반환합니다. (ldap_parse_resultA)

ldap_parse_resultW ldap_parse_resultW(유니코드) 함수(winldap.h)는 서버의 응답을 구문 분석하고 적절한 필드를 반환합니다.
ldap_parse_sort_control ldap_parse_sort_control 함수(winldap.h)는 서버에서 반환된 정렬 컨트롤을 구문 분석합니다.
ldap_parse_sort_controlA ldap_parse_sort_control 함수는 서버에서 반환된 정렬 컨트롤을 구문 분석합니다. (ldap_parse_sort_controlA)
ldap_parse_sort_controlW ldap_parse_sort_controlW(유니코드) 함수(winldap.h)는 서버에서 반환된 정렬 컨트롤을 구문 분석합니다.
ldap_parse_vlv_controlA VLV 검색 결과를 찾아 구문 분석하는 데 사용됩니다. (ANSI)
ldap_parse_vlv_controlW VLV 검색 결과를 찾아 구문 분석하는 데 사용됩니다. (유니코드)
ldap_perror 사용되지 않는 함수입니다. 호환성을 위해서만 존재합니다.
ldap_rename_ext ldap_rename_ext 함수(winldap.h)는 디렉터리에 있는 항목의 고유 이름을 변경하는 비동기 작업을 시작합니다.
ldap_rename_ext_s ldap_rename_ext_s 함수(winldap.h)는 디렉터리에 있는 항목의 고유 이름을 변경하는 동기 연산입니다.
ldap_rename_ext_sA ldap_rename_ext_s 함수는 디렉터리에 있는 항목의 고유 이름을 변경하는 동기 연산입니다. 이 함수는 LDAP 3에서 사용할 수 있습니다. (ldap_rename_ext_sA)

ldap_rename_ext_sW ldap_rename_ext_sW(유니코드) 함수(winldap.h)는 디렉터리에 있는 항목의 고유 이름을 변경하는 동기 연산입니다.
ldap_rename_extA ldap_rename_ext 함수는 디렉터리에 있는 항목의 고유 이름을 변경하는 비동기 작업을 시작합니다. 이 함수는 LDAP 3에서 사용할 수 있습니다. (ldap_rename_extA)
ldap_rename_extW ldap_rename_extW(유니코드) 함수(winldap.h)는 디렉터리에 있는 항목의 고유 이름을 변경하는 비동기 작업을 시작합니다.
ldap_result 비동기 작업의 결과를 가져옵니다.
ldap_result2error ldap_result2error 함수는 메시지를 구문 분석하고 오류 코드를 반환합니다.
ldap_sasl_bind_sA ldap_sasl_bind_s 함수는 SASL을 사용하여 LDAP 서버에 클라이언트를 인증하는 동기 함수입니다. (ANSI)
ldap_sasl_bind_sW ldap_sasl_bind_s 함수는 SASL을 사용하여 LDAP 서버에 클라이언트를 인증하는 동기 함수입니다. (유니코드)
ldap_sasl_bindA ldap_sasl_bind SASL을 사용하여 클라이언트를 LDAP 서버에 인증하는 비동기 함수입니다. (ANSI)
ldap_sasl_bindW ldap_sasl_bind SASL을 사용하여 클라이언트를 LDAP 서버에 인증하는 비동기 함수입니다. (유니코드)
ldap_search ldap_search 함수(winldap.h)는 LDAP 디렉터리를 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다.

ldap_search_abandon_page ldap_search_abandon_page 함수는 페이징된 결과 검색을 종료합니다.
ldap_search_ext ldap_search_ext 함수(winldap.h)는 LDAP 디렉터리를 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다.
ldap_search_ext_s ldap_search_ext_s 함수(winldap.h)는 LDAP 디렉터리를 동기적으로 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다.
ldap_search_ext_sA 동기적으로 LDAP 디렉터리를 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다. (ldap_search_ext_sA)
ldap_search_ext_sW ldap_search_ext_sW(유니코드) 함수(winldap.h)는 LDAP 디렉터리를 동기적으로 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다.
ldap_search_extA LDAP 디렉터리를 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다. (ldap_search_extA)
ldap_search_extW ldap_search_extW(유니코드) 함수(winldap.h)는 LDAP 디렉터리를 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다.
ldap_search_init_page ldap_search_init_page 함수(winldap.h)는 간단한 페이징 결과 검색에 대한 검색 블록을 초기화합니다.
ldap_search_init_pageA 간단한 페이징 결과 검색에 대한 검색 블록을 초기화합니다. (ldap_search_init_pageA)
ldap_search_init_pageW ldap_search_init_pageW(유니코드) 함수(winldap.h)는 간단한 페이징 결과 검색에 대한 검색 블록을 초기화합니다.

ldap_search_s ldap_search_s 함수(winldap.h)는 LDAP 디렉터리를 동기적으로 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다.
ldap_search_sA ldap_search_s 함수는 LDAP 디렉터리를 동기적으로 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다. (ldap_search_sA)
ldap_search_st ldap_search_st 함수(winldap.h)는 LDAP 디렉터리를 동기적으로 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다.
ldap_search_stA ldap_search_st 함수는 LDAP 디렉터리를 동기적으로 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다. 추가 매개 변수는 검색에 대한 로컬 시간 초과를 지정합니다. (ldap_search_stA)
ldap_search_stW ldap_search_stW(유니코드) 함수(winldap.h)는 LDAP 디렉터리를 동기적으로 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다.
ldap_search_sW ldap_search_sW(유니코드) 함수(winldap.h)는 LDAP 디렉터리를 동기적으로 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다.
ldap_searchA LDAP 디렉터리를 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다. (ldap_searchA)
ldap_searchW ldap_searchW(유니코드) 함수(winldap.h)는 LDAP 디렉터리를 검색하고 일치하는 각 항목에 대해 요청된 특성 집합을 반환합니다.
ldap_set_option ldap_set_option 함수(winldap.h)는 연결 블록에 대한 옵션을 설정합니다.
ldap_set_optionW ldap_set_optionW(유니코드) 함수(winldap.h)는 연결 블록에 대한 옵션을 설정합니다.

ldap_simple_bind ldap_simple_bind 함수(winldap.h)는 일반 텍스트 암호를 사용하여 서버에 대한 클라이언트를 비동기적으로 인증합니다.
ldap_simple_bind_s ldap_simple_bind_s 함수(winldap.h)는 일반 텍스트 암호를 사용하여 서버에 클라이언트를 동기적으로 인증합니다.
ldap_simple_bind_sA ldap_simple_bind_s 함수는 일반 텍스트 암호를 사용하여 서버에 클라이언트를 동기적으로 인증합니다. (ldap_simple_bind_sA)
ldap_simple_bind_sW ldap_simple_bind_sW(유니코드) 함수(winldap.h)는 일반 텍스트 암호를 사용하여 서버에 클라이언트를 동기적으로 인증합니다.
ldap_simple_bindA 일반 텍스트 암호를 사용하여 서버에 대한 클라이언트를 비동기적으로 인증합니다. (ldap_simple_bindA)
ldap_simple_bindW ldap_simple_bindW(유니코드) 함수(winldap.h)는 일반 텍스트 암호를 사용하여 서버에 대한 클라이언트를 비동기적으로 인증합니다.
ldap_sslinit ldap_sslinit 함수(winldap.h)는 LDAP 서버를 사용하여 SSL(Secure Sockets Layer) 세션을 초기화합니다.
ldap_sslinitA LDAP 서버를 사용하여 SSL(Secure Sockets Layer) 세션을 초기화합니다. (ldap_sslinitA)
ldap_sslinitW ldap_sslinitW(유니코드) 함수(winldap.h)는 LDAP 서버를 사용하여 SSL(Secure Sockets Layer) 세션을 초기화합니다.
ldap_start_tls_sA 활성 LDAP 세션에서 TLS 암호화 사용을 시작하는 데 사용됩니다. (ANSI)

ldap_start_tls_sW 활성 LDAP 세션에서 TLS 암호화 사용을 시작하는 데 사용됩니다. (유니코드)
ldap_stop_tls_s ldap_start_tls_s 호출로 시작된 암호화 작업을 중지합니다.
ldap_ufn2dn ldap_ufn2dn 함수(winldap.h)는 사용자에게 친숙한 이름을 고유 이름으로 변환합니다.
ldap_ufn2dnA 사용자에게 친숙한 이름을 고유 이름으로 변환합니다. (ldap_ufn2dnA)
ldap_ufn2dnW ldap_ufn2dnW(유니코드) 함수(winldap.h)는 사용자에게 친숙한 이름을 고유 이름으로 변환합니다.
ldap_unbind ldap_unbind 함수는 LDAP 세션과 연결된 리소스를 해제합니다.
ldap_unbind_s ldap_unbind_s 함수는 LDAP 세션과 연결된 리소스를 동기적으로 해제합니다.
ldap_value_free ldap_value_free 함수(winldap.h)는 ldap_get_values 반환된 구조를 해제합니다.
ldap_value_free_len 이 ldap_value_free_len ldap_get_values_len 반환된 버벌 구조를 해제합니다.
ldap_value_freeA ldap_get_values 반환된 구조를 해제합니다. (ldap_value_freeA)
ldap_value_freeW ldap_value_freeW(유니코드) 함수(winldap.h)는 ldap_get_values 반환된 구조를 해제합니다.
LdapGetLastError LdapGetLastError 함수는 LDAP 호출에서 반환된 마지막 오류 코드를 검색합니다.

LdapMapErrorToWin32 LdapMapErrorToWin32 함수는 LdapError 값을 가장 가까운 Win32 오류 코드로 변환합니다.
LdapUnicodeToUTF8 유니코드 문자열을 UTF-8로 변환합니다.
LdapUTF8ToUnicode UTF-8 코드 페이지가 없는 모듈의 문자열을 번역하는 데 사용됩니다.

콜백 함수

QUERYCLIENTCERT SSL(Secure Sockets Layer) 연결을 설정할 때 서버가 클라이언트에서 인증서를 요청할 수 있도록 합니다.
VERIFYSERVERCERT 클라이언트가 연결된 서버의 인증서 체인을 평가할 수 있습니다.

구조체

BerElement BER(기본 인코딩 규칙) 인코딩을 수행하는 C++ 클래스 개체입니다.
LDAP LDAP 세션을 나타냅니다.
LDAP_BERVAL berval 구조는 BER(기본 인코딩 규칙)에 따라 인코딩되는 임의의 이진 데이터를 나타냅니다. 버발을 사용하여 null로 끝나는 문자열로 나타낼 수 없는 특성을 나타냅니다.

LDAP_REFERRAL_CALLBACK 연결의 외부 캐싱을 구현하는 데 사용됩니다.
LDAP_TIMEVAL 시간 간격을 나타내는 데 사용됩니다.
LDAPAPIFeatureInfoA 지원되는 모든 LDAP API 확장에 대한 데이터를 검색합니다. (ANSI)
LDAPAPIFeatureInfoW 지원되는 모든 LDAP API 확장에 대한 데이터를 검색합니다. (유니코드)
LDAPAPIInfoA 사용된 API 및 구현에 대한 데이터를 검색합니다. (ANSI)
LDAPAPIInfoW 사용된 API 및 구현에 대한 데이터를 검색합니다. (유니코드)
LDAPControlA 클라이언트 쪽 컨트롤과 서버 컨트롤을 모두 나타냅니다. (ANSI)
LDAPControlW 클라이언트 쪽 컨트롤과 서버 컨트롤을 모두 나타냅니다. (유니코드)
LDAPMessage LDAP 함수에서 결과 및 오류 데이터를 반환하는 데 사용됩니다.
LDAPModA 수정 작업을 수행하는 데 필요한 데이터를 보유합니다. (ANSI)
LDAPModW 수정 작업을 수행하는 데 필요한 데이터를 보유합니다. (유니코드)
LDAPSortKeyA LDAPSortKey 구조체는 정렬 컨트롤에서 사용할 정렬 조건을 저장합니다. (ANSI)

LDAPSortKeyW LDAPSortKey 구조체는 정렬 컨트롤에서 사용할 정렬 조건을 저장합니다. (유니코드)
LDAPVLVInfo VLV(가상 목록 보기) 요청 컨트롤(LDAP_CONTROL_VLVREQUEST)에 대한 검색 매개 변수를 설정하는 데 사용됩니다.

winml.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows Machine Learning에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Machine Learning

winml.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWinMLEvaluationContext WinML 모델에 입력 및 출력을 바인딩하는 컨텍스트를 나타냅니다.
IWinMLModel 해당 메타데이터가 있는 Windows Machine Learning 모델을 나타냅니다. 에는 모델 설명(이름, 작성자, 버전 관리 등)과 예상 입력 및 출력이 포함됩니다.
IWinMLRuntime WinML 모델을 로드하고 평가할 런타임을 나타냅니다.
IWinMLRuntimeFactory 모델 로드 및 평가를 위한 WinML 런타임을 만드는 팩터리를 나타냅니다.

구조체

WINML_BINDING_DESC WinML 바인딩에 대한 설명을 포함합니다.
WINML_IMAGE_BINDING_DESC 형식 이미지의 바인딩에 대한 속성을 포함합니다.
WINML_IMAGE_VARIABLE_DESC

이미지 변수 설명에 대한 속성을 포함합니다.
WINML_MAP_BINDING_DESC 형식 맵의 바인딩에 대한 속성을 포함합니다.
WINML_MAP_VARIABLE_DESC 지도 변수의 설명 속성을 포함합니다.
WINML_MODEL_DESC 모델의 설명 속성을 포함합니다.
WINML_RESOURCE_BINDING_DESC 리소스 바인딩의 설명 속성을 포함합니다.
WINML_SEQUENCE_BINDING_DESC 시퀀스 바인딩의 설명 속성을 포함합니다.
WINML_SEQUENCE_VARIABLE_DESC 시퀀스 변수의 설명 속성을 포함합니다.
WINML_TENSOR_BINDING_DESC 텐서 바인딩의 설명 속성을 포함합니다.
WINML_TENSOR_VARIABLE_DESC 텐서 변수의 설명 속성을 포함합니다.
WINML_VARIABLE_DESC 변수의 설명 속성을 포함합니다.

열거형

WINML_BINDING_TYPE 다양한 유형의 WinML 바인딩을 지정합니다.
WINML_FEATURE_TYPE

다양한 유형의 WinML 기능을 지정합니다.
WINML_RUNTIME_TYPE 다양한 유형의 WinML 런타임을 지정합니다.
WINML_TENSOR_DATA_TYPE WinML 텐서의 다양한 데이터 형식을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winnetwk.h 헤더

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID
- WNet(Windows 네트워킹)

winnetwk.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

MultinetGetConnectionPerformanceA 네트워크 리소스에 액세스하는 데 사용되는 연결의 예상 성능에 대한 정보를 반환합니다. (ANSI)
MultinetGetConnectionPerformanceW 네트워크 리소스에 액세스하는 데 사용되는 연결의 예상 성능에 대한 정보를 반환합니다. (유니코드)
WNetAddConnection2A WNetAddConnection2 함수는 네트워크 리소스에 연결하고 로컬 디바이스를 네트워크 리소스로 리디렉션할 수 있습니다. (ANSI)
WNetAddConnection2W WNetAddConnection2 함수는 네트워크 리소스에 연결하고 로컬 디바이스를 네트워크 리소스로 리디렉션할 수 있습니다. (유니코드)
WNetAddConnection3A WNetAddConnection3 함수는 네트워크 리소스에 연결합니다. 이 함수는 로컬 디바이스를 네트워크 리소스로 리디렉션할 수 있습니다. (ANSI)
WNetAddConnection3W WNetAddConnection3 함수는 네트워크 리소스에 연결합니다. 이 함수는 로컬 디바이스를 네트워크 리소스로 리디렉션할 수 있습니다. (유니코드)
WNetAddConnection4A WNetAddConnection4 함수는 네트워크 리소스에 연결합니다. 이 함수는 로컬 디바이스를 네트워크 리소스로 리디렉션할 수 있습니다. (ANSI)

WNetAddConnection4W WNetAddConnection4 함수는 네트워크 리소스에 연결합니다. 이 함수는 로컬 디바이스를 네트워크 리소스로 리디렉션할 수 있습니다. (유니코드)
WNetAddConnectionA WNetAddConnection 함수를 사용하면 호출하는 애플리케이션이 로컬 디바이스를 네트워크 리소스에 연결할 수 있습니다. 성공적인 연결은 영구적입니다. 즉, 시스템이 후속 로그온 작업 중에 연결을 자동으로 복원합니다. (ANSI)
WNetAddConnectionW WNetAddConnection 함수를 사용하면 호출하는 애플리케이션이 로컬 디바이스를 네트워크 리소스에 연결할 수 있습니다. 성공적인 연결은 영구적입니다. 즉, 시스템이 후속 로그온 작업 중에 연결을 자동으로 복원합니다. (유니코드)
WNetCancelConnection2A WNetCancelConnection2 함수는 기존 네트워크 연결을 취소합니다. 함수를 호출하여 현재 연결되지 않은 기억된 네트워크 연결을 제거할 수도 있습니다. (ANSI)
WNetCancelConnection2W WNetCancelConnection2 함수는 기존 네트워크 연결을 취소합니다. 함수를 호출하여 현재 연결되지 않은 기억된 네트워크 연결을 제거할 수도 있습니다. (유니코드)
WNetCancelConnectionA WNetCancelConnection 함수는 기존 네트워크 연결을 취소합니다. (ANSI)
WNetCancelConnectionW WNetCancelConnection 함수는 기존 네트워크 연결을 취소합니다. (유니코드)
WNetCloseEnum WNetCloseEnum 함수는 WNetOpenEnum 함수 호출로 시작된 네트워크 리소스 열거형을 종료합니다.
WNetConnectionDialog WNetConnectionDialog 함수는 네트워크 리소스에 연결하기 위한 일반 검색 대화 상자를 시작합니다. 이 함수에는 대화 상자의 소유자 창에 대한 핸들이 필요합니다.
WNetConnectionDialog1A WNetConnectionDialog1 함수는 네트워크 리소스에 연결하기 위한 일반 검색 대화 상자를 표시합니다. 이 함수는 대화 상자 매개 변수를 설정하기 위해 CONNECTDLGSTRUCT가 필요합니다. (ANSI)

WNetConnectionDialog1W WNetConnectionDialog1 함수는 네트워크 리소스에 연결하기 위한 일반 검색 대화 상자를 표시합니다. 이 함수는 대화 상자 매개 변수를 설정하기 위해 CONNECTDLGSTRUCT가 필요합니다. (유니코드)
WNetDisconnectDialog WNetDisconnectDialog 함수는 네트워크 리소스에서 연결을 끊기 위한 일반 검색 대화 상자를 시작합니다. 이 함수에는 대화 상자의 소유자 창에 대한 핸들이 필요합니다.
WNetDisconnectDialog1A WNetDisconnectDialog1 함수는 네트워크 리소스의 연결을 끊으려고 시도합니다. (ANSI)
WNetDisconnectDialog1W WNetDisconnectDialog1 함수는 네트워크 리소스의 연결을 끊으려고 시도합니다. (유니코드)
WNetEnumResourceA WNetEnumResource 함수는 WNetOpenEnum 함수에 대한 호출로 시작된 네트워크 리소스의 열거를 계속합니다. (ANSI)
WNetEnumResourceW WNetEnumResource 함수는 WNetOpenEnum 함수에 대한 호출로 시작된 네트워크 리소스의 열거를 계속합니다. (유니코드)
WNetGetConnectionA WNetGetConnection 함수는 로컬 디바이스와 연결된 네트워크 리소스의 이름을 검색합니다. (ANSI)
WNetGetConnectionW WNetGetConnection 함수는 로컬 디바이스와 연결된 네트워크 리소스의 이름을 검색합니다. (유니코드)
WNetGetLastErrorA WNetGetLastError 함수는 WNet 함수에서 설정한 최신 확장 오류 코드를 검색합니다. 네트워크 공급자가 이 오류 코드를 보고했습니다. 일반적으로 SDK 헤더 파일 WinError.h에 포함된 오류 중 하나가 아닙니다. (ANSI)
WNetGetLastErrorW WNetGetLastError 함수는 WNet 함수에서 설정한 최신 확장 오류 코드를 검색합니다. 네트워크 공급자가 이 오류 코드를 보고했습니다. 일반적으로 SDK 헤더 파일 WinError.h에 포함된 오류 중 하나가 아닙니다. (유니코드)

WNetGetNetworkInformationA WNetGetNetworkInformation 함수는 이전 네트워크 열거형에서 이름이 반환된 특정 네트워크 공급자에 대한 확장 정보를 반환합니다. (ANSI)
WNetGetNetworkInformationW WNetGetNetworkInformation 함수는 이전 네트워크 열거형에서 이름이 반환된 특정 네트워크 공급자에 대한 확장 정보를 반환합니다. (유니코드)
WNetGetProviderNameA WNetGetProviderName 함수는 특정 유형의 네트워크에 대한 공급자 이름을 가져옵니다. (ANSI)
WNetGetProviderNameW WNetGetProviderName 함수는 특정 유형의 네트워크에 대한 공급자 이름을 가져옵니다. (유니코드)
WNetGetResourceInformationA 네트워크 리소스에 대한 원격 경로를 제공하는 경우 WNetGetResourceInformation 함수는 리소스를 소유하는 네트워크 공급자를 식별하고 리소스 유형에 대한 정보를 가져옵니다. (ANSI)
WNetGetResourceInformationW 네트워크 리소스에 대한 원격 경로를 제공하는 경우 WNetGetResourceInformation 함수는 리소스를 소유하는 네트워크 공급자를 식별하고 리소스 유형에 대한 정보를 가져옵니다. (유니코드)
WNetGetResourceParentA WNetGetResourceParent 함수는 네트워크 찾아보기 계층 구조에서 네트워크 리소스의 부모를 반환합니다. 검색은 지정된 네트워크 리소스의 위치에서 시작됩니다. (ANSI)
WNetGetResourceParentW WNetGetResourceParent 함수는 네트워크 찾아보기 계층 구조에서 네트워크 리소스의 부모를 반환합니다. 검색은 지정된 네트워크 리소스의 위치에서 시작됩니다. (유니코드)
WNetGetUniversalNameA WNetGetUniversalName 함수는 네트워크 리소스에 대한 드라이브 기반 경로를 사용하고 보다 보편적인 형식의 이름을 포함하는 정보 구조를 반환합니다. (ANSI)
WNetGetUniversalNameW WNetGetUniversalName 함수는 네트워크 리소스에 대한 드라이브 기반 경로를 사용하고 보다 보편적인 형식의 이름을 포함하는 정보 구조를 반환합니다. (유니코드)

WNetGetUserA WNetGetUser 함수는 현재 기본 사용자 이름 또는 네트워크 연결을 설정하는 데 사용되는 사용자 이름을 검색합니다. (ANSI)
WNetGetUserW WNetGetUser 함수는 현재 기본 사용자 이름 또는 네트워크 연결을 설정하는 데 사용되는 사용자 이름을 검색합니다. (유니코드)
WNetOpenEnumA WNetOpenEnum 함수는 네트워크 리소스 또는 기존 연결의 열거형을 시작합니다. WNetEnumResource 함수를 호출하여 열거형을 계속할 수 있습니다. (ANSI)
WNetOpenEnumW WNetOpenEnum 함수는 네트워크 리소스 또는 기존 연결의 열거형을 시작합니다. WNetEnumResource 함수를 호출하여 열거형을 계속할 수 있습니다. (유니코드)
WNetRestoreConnectionW WNetRestoreConnectionW 함수는 네트워크 리소스에 대한 연결을 복원합니다. 이 함수는 필요한 경우 사용자에게 이름과 암호를 묻는 메시지를 표시합니다.
WNetUseConnectionA WNetUseConnection 함수는 네트워크 리소스에 연결합니다. 이 함수는 로컬 디바이스를 네트워크 리소스로 리디렉션할 수 있습니다. (ANSI)
WNetUseConnectionW WNetUseConnection 함수는 네트워크 리소스에 연결합니다. 이 함수는 로컬 디바이스를 네트워크 리소스로 리디렉션할 수 있습니다. (유니코드)

구조

테이블 확장

CONNECTDLGSTRUCTA WNetConnectionDialog1 함수에서 검색 대화 상자 매개 변수를 설정하는 데 사용됩니다. (ANSI)
CONNECTDLGSTRUCTW

WNetConnectionDialog1 함수에서 검색 대화 상자 매개 변수를 설정하는 데 사용됩니다. (유니코드)
DISCDLGSTRUCTA WNetDisconnectDialog1 함수에 사용됩니다. 구조체에는 연결 끊기 시도에 필요한 정보가 포함되어 있습니다. (ANSI)
DISCDLGSTRUCTW WNetDisconnectDialog1 함수에 사용됩니다. 구조체에는 연결 끊기 시도에 필요한 정보가 포함되어 있습니다. (유니코드)
NETCONNECTINFOSTRUCT NETCONNECTINFOSTRUCT 구조에는 네트워크의 성능에 대한 정보가 포함되어 있습니다. NPGetConnectionPerformance 함수에서 사용됩니다.
NETINFOSTRUCT WNetGetNetworkInformation 함수에서 반환된 네트워크 공급자를 설명하는 정보를 포함합니다.
NETRESOURCEA 다음 구조에는 네트워크 리소스에 대한 정보가 포함되어 있습니다. NPOpenEnum 및 NPAddConnection을 비롯한 여러 네트워크 공급자 함수에서 사용됩니다. (ANSI)
NETRESOURCEW 다음 구조에는 네트워크 리소스에 대한 정보가 포함되어 있습니다. NPOpenEnum 및 NPAddConnection을 비롯한 여러 네트워크 공급자 함수에서 사용됩니다. (유니코드)
REMOTE_NAME_INFOA REMOTE_NAME_INFO 구조에는 범용 이름의 원격 형식에 대한 정보가 포함됩니다. NPGetUniversalName 함수에서 사용됩니다. (ANSI)
REMOTE_NAME_INFOW REMOTE_NAME_INFO 구조에는 범용 이름의 원격 형식에 대한 정보가 포함됩니다. NPGetUniversalName 함수에서 사용됩니다. (유니코드)
UNIVERSAL_NAME_INFOA UNIVERSAL_NAME_INFO 구조체에는 범용 이름의 UNC 형식에 대한 정보가 포함됩니다. NPGetUniversalName 함수에서 사용됩니다. (ANSI)
UNIVERSAL_NAME_INFOW

UNIVERSAL_NAME_INFO 구조체에는 범용 이름의 UNC 형식에 대한 정보가 포함됩니다. NPGetUniversalName 함수에서 사용됩니다. (유니코드)

Last updated on 2025. 08. 16.

winnls.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows 애플리케이션에 대한 Internationalization에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 애플리케이션에 대한 국제화

winnls.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CompareString CompareString 함수(winnls.h)는 식별자로 지정된 로캘에 대해 두 개의 문자열을 비교합니다.
CompareStringA 식별자로 지정된 로캘에 대해 두 개의 문자열을 비교합니다. 주의 CompareString을 잘못 사용하면 애플리케이션의 보안이 손상됩니다. (CompareStringA)
ConvertDefaultLocale 기본 로캘 값을 실제 로캘 식별자로 변환합니다.
EnumCalendarInfoA 지정된 로캘의 일정 정보를 열거합니다. 참고 일정 정보 외에 일정 식별자를 받으려면 애플리케이션에서 EnumCalendarInfoEx 함수를 사용해야 합니다. (ANSI)
EnumCalendarInfoExA 식별자가 지정한 로캘의 일정 정보를 열거합니다. 참고 Windows Vista 이상에서만 실행되는 모든 애플리케이션은 이 함수에 대한 기본 설정으로 EnumCalendarInfoExEx를 사용해야 합니다. (ANSI)
EnumCalendarInfoExEx 이름으로 지정된 로캘의 일정 정보를 열거합니다. 참고 애플리케이션은 Windows Vista 이상에서만 실행되도록 설계된 경우 EnumCalendarInfo 또는 EnumCalendarInfoEx에 대한 기본 설정으로 이 함수를 호출해야 합니다. 참고 이 함수는 예를 들어 사용자 지정 로캘로 인해 릴리스 간에 변경되는 데이터를 열거할 수 있습니다. 애플리케이션이 데이터를 유지하거나 전송해야 하는 경우 영구 로캘 데이터 사용을 참조하세요.
EnumCalendarInfoExW

식별자가 지정한 로캘의 일정 정보를 열거합니다. 참고 Windows Vista 이상에서만 실행되는 모든 애플리케이션은 이 함수에 대한 기본 설정으로 EnumCalendarInfoExEx를 사용해야 합니다. (유니코드)
EnumCalendarInfoW 지정된 로캘의 일정 정보를 열거합니다. 참고 일정 정보 외에 일정 식별자를 받으려면 애플리케이션에서 EnumCalendarInfoEx 함수를 사용해야 합니다. (유니코드)
EnumDateFormatsA 지정된 로캘에 사용할 수 있는 긴 날짜, 짧은 날짜 또는 연도/월 형식을 열거합니다. (ANSI)
EnumDateFormatsExA 지정된 로캘에 사용할 수 있는 긴 날짜, 짧은 날짜 또는 연도/월 형식을 열거합니다. 참고 Windows Vista 이상에서만 실행되는 모든 애플리케이션은 이 함수에 대한 기본 설정으로 EnumDateFormatsExEx를 사용해야 합니다. (ANSI)
EnumDateFormatsExEx 이름으로 지정된 로캘에 사용할 수 있는 긴 날짜, 짧은 날짜 또는 연도/월 형식을 열거합니다. 참고 애플리케이션은 Windows Vista 이상에서만 실행되도록 설계된 경우 EnumDateFormats 또는 EnumDateFormatsEx를 기본 설정으로 이 함수를 호출해야 합니다. 참고 이 함수는 예를 들어 사용자 지정 로캘로 인해 릴리스 간에 변경되는 데이터를 열거할 수 있습니다. 애플리케이션이 데이터를 유지하거나 전송해야 하는 경우 영구 로캘 데이터 사용을 참조하세요.
EnumDateFormatsExW 지정된 로캘에 사용할 수 있는 긴 날짜, 짧은 날짜 또는 연도/월 형식을 열거합니다. 참고 Windows Vista 이상에서만 실행되는 모든 애플리케이션은 이 함수에 대한 기본 설정으로 EnumDateFormatsExEx를 사용해야 합니다. (유니코드)
EnumDateFormatsW 지정된 로캘에 사용할 수 있는 긴 날짜, 짧은 날짜 또는 연도/월 형식을 열거합니다. (유니코드)
EnumLanguageGroupLocalesA 지정된 언어 그룹의 로캘을 열거합니다. (ANSI)
EnumLanguageGroupLocalesW 지정된 언어 그룹의 로캘을 열거합니다. (유니코드)
EnumSystemCodePagesA 운영 체제에 설치되거나 지원되는 코드 페이지를 열거합니다. (ANSI)

EnumSystemCodePagesW 운영 체제에 설치되거나 지원되는 코드 페이지를 열거합니다. (유니코드)
EnumSystemGeoID 운영 체제에서 사용할 수 있는 지리적 위치 식별자(GEOID 유형)를 열거합니다.
EnumSystemGeoNames 운영 체제에서 사용할 수 있는 지리적 위치에 대한 두 글자 ISO(국제 표준화 기구) 3166-1 코드 또는 숫자 UN(UN) 시리즈 M, 번호 49(M.49) 코드를 열거합니다.
EnumSystemLanguageGroupsA 운영 체제에 설치되거나 지원되는 언어 그룹을 열거합니다. (ANSI)
EnumSystemLanguageGroupsW 운영 체제에 설치되거나 지원되는 언어 그룹을 열거합니다. (유니코드)
EnumSystemLocalesA 운영 체제에 설치되거나 지원되는 로캘을 열거합니다. 참고 상호 운용성을 위해 애플리케이션은 EnumSystemLocalesEx 함수를 EnumSystemLocales에 선호해야 합니다. Microsoft는 새 로캘에 대한 로캘 식별자 대신 로캘 이름을 사용하도록 마이그레이션하고 있기 때문입니다. Windows Vista 이상에서만 실행되는 모든 애플리케이션은 EnumSystemLocalesEx를 사용해야 합니다. (ANSI)
EnumSystemLocalesEx 운영 체제에 설치되거나 지원되는 로캘을 열거합니다. 참고 애플리케이션은 Windows Vista 이상에서만 실행되도록 설계된 경우 EnumSystemLocales를 기본 설정으로 이 함수를 호출해야 합니다.
EnumSystemLocalesW 운영 체제에 설치되거나 지원되는 로캘을 열거합니다. 참고 상호 운용성을 위해 애플리케이션은 EnumSystemLocalesEx 함수를 EnumSystemLocales에 선호해야 합니다. Microsoft는 새 로캘에 대한 로캘 식별자 대신 로캘 이름을 사용하도록 마이그레이션하고 있기 때문입니다. Windows Vista 이상에서만 실행되는 모든 애플리케이션은 EnumSystemLocalesEx를 사용해야 합니다. (유니코드)
EnumTimeFormatsA 식별자가 지정한 로캘에 사용할 수 있는 시간 형식을 열거합니다. 참고 상호 운용성을 위해 애플리케이션은 EnumTimeFormatsEx 함수를 EnumTimeFormats에 선호해야 합니다. Microsoft는 새 로캘에 대한 로캘 식별자 대신 로캘 이름을 사용하도록 마이그레이션하고 있기 때문입니다. Windows Vista 이상에서만 실행되는 모든 애플리케이션은 EnumTimeFormatsEx를 사용해야 합니다. (ANSI)
EnumTimeFormatsEx

이름으로 지정된 로캘에 사용할 수 있는 시간 형식을 열거합니다. 참고 애플리케이션은 Windows Vista 이상에서만 실행되도록 설계된 경우 EnumTimeFormats에 대한 기본 설정으로 이 함수를 호출해야 합니다. 참고 이 함수는 예를 들어 사용자 지정 로캘로 인해 릴리스 간에 변경되는 데이터를 열거할 수 있습니다. 애플리케이션이 데이터를 유지하거나 전송해야 하는 경우 영구 로캘 데이터 사용을 참조하세요.
EnumTimeFormatsW 식별자가 지정한 로캘에 사용할 수 있는 시간 형식을 열거합니다. 참고 상호 운용성을 위해 애플리케이션은 EnumTimeFormatsEx 함수를 EnumTimeFormats에 선호해야 합니다. Microsoft는 새 로캘에 대한 로캘 식별자 대신 로캘 이름을 사용하도록 마이그레이션하고 있기 때문입니다. Windows Vista 이상에서만 실행되는 모든 애플리케이션은 EnumTimeFormatsEx를 사용해야 합니다. (유니코드)
EnumUILanguagesA 운영 체제에서 사용할 수 있는 사용자 인터페이스 언어를 열거하고 목록의 모든 언어로 콜백 함수를 호출합니다. (ANSI)
EnumUILanguagesW 운영 체제에서 사용할 수 있는 사용자 인터페이스 언어를 열거하고 목록의 모든 언어로 콜백 함수를 호출합니다. (유니코드)
FILEMUIINFO_GET_CULTURE FILEMUIINFO 구조체의 dwLanguageNameOffset 멤버에서 언어 이름 오프셋 정보와 연결된 로캘 이름을 가져옵니다.
FILEMUIINFO_GET_MAIN_TYPEID FILEMUIINFO 구조체의 형식 식별자 크기 및 오프셋과 연결된 기본 모듈 형식 배열 요소를 가져옵니다. 정보는 구조체의 dwTypeIDMainSize 및 dwTypeIDMainOffset 멤버에 제공됩니다.
FILEMUIINFO_GET_MAIN_TYPEIDS FILEMUIINFO 구조체의 dwTypeIDMainOffset 멤버에서 형식 식별자 오프셋 정보와 연결된 기본 모듈 형식 배열을 가져옵니다.
FILEMUIINFO_GET_MAIN_TYPENAMES fileMUIINFO 구조체의 dwTypeNameMainOffset 멤버에 있는 형식 이름 오프셋 정보와 연결된 기본 모듈 이름 다중 문자열 배열을 가져옵니다.
FILEMUIINFO_GET_MUI_TYPEID FILEMUIINFO 구조체의 형식 식별자 크기 및 오프셋 정보와 연결된 MUI 모듈 형식 배열 요소를 가져옵니다. 정보는 구조체의 dwTypeIDMUISize 및 dwTypeIDMUIOffset 멤버에 제공됩니다.

FILEMUIINFO_GET_MUI_TYPEIDS FILEMUIINFO 구조체의 dwTypeIDMUIOffset 멤버에서 형식 식별자 오프셋 정보와 연결된 MUI 모듈 형식 배열을 가져옵니다.
FILEMUIINFO_GET_MUI_TYPENAMES FILEMUIINFO 구조체의 dwTypeNameMUIOffset 멤버에서 형식 이름 오프셋 정보와 연결된 MUI 모듈 이름 다중 문자열 배열을 가져옵니다.
FindNLSString 식별자로 지정된 로캘의 다른 유니코드 문자열에서 유니코드 문자열(와이드 문자) 또는 해당 문자열을 찾습니다. 주의 이진 표현이 매우 다른 문자열은 동일하게 비교할 수 있으므로 이 함수는 특정 보안 문제를 제기할 수 있습니다. 자세한 내용은 보안 고려 사항:_International 기능의 비교 함수에 대한 설명을 참조하세요. 참고 상호 운용성을 위해 Microsoft는 새 로캘에 대한 로캘 식별자 대신 로캘 이름을 사용하도록 마이그레이션하기 때문에 애플리케이션에서 FindNLSStringEx 함수를 선호해야 합니다. FindNLSString은 사용자 지정 로캘을 지원하지만 대부분의 애플리케이션은 이러한 유형의 지원에 FindNLSStringEx를 사용해야 합니다.
FindNLSStringEx 이름으로 지정된 로캘의 다른 유니코드 문자열에서 유니코드 문자열(와이드 문자) 또는 해당 문자열을 찾습니다. 주의 이진 표현이 매우 다른 문자열은 동일하게 비교할 수 있으므로 이 함수는 특정 보안 문제를 제기할 수 있습니다. 자세한 내용은 보안 고려 사항:_International 기능의 비교 함수에 대한 설명을 참조하세요.
FoldStringA 한 유니코드 문자열을 다른 문자열에 매핑하여 지정된 변환을 수행합니다. (FoldStringA)
GetACP 운영 체제에 대한 현재 Windows ANSI 코드 페이지 식별자를 검색합니다. 주의 ANSI API 함수(예: ANSI 버전의 TextOut)는 GetACP를 암시적으로 사용하여 유니코드로 또는 유니코드로 텍스트를 번역합니다.
GetCalendarInfoA 식별자가 지정한 로캘의 일정 정보를 검색합니다. (ANSI)
GetCalendarInfoEx 이름으로 지정된 로캘의 일정 정보를 검색합니다. 참고 애플리케이션은 Windows Vista 이상에서만 실행되도록 설계된 경우 기본 설정으로 이 함수를 GetCalendarInfo로 호출해야 합니다. 참고 이 함수는 예를 들어 사용자 지정 로캘로 인해 릴리스 간에 변경되는 데이터를 검색할 수 있습니다. 애플리케이션이 데이터를 유지하거나 전송해야 하는 경우 영구 로캘 데이터 사용을 참조하세요.

GetCalendarInfoW 식별자가 지정한 로캘의 일정 정보를 검색합니다. (유니코드)
GetCPInfo 유효한 설치 또는 사용 가능한 코드 페이지에 대한 정보를 검색합니다. (GetCPInfo)
GetCPInfoExA 유효한 설치 또는 사용 가능한 코드 페이지에 대한 정보를 검색합니다. (GetCPInfoExA)
GetCPInfoExW 유효한 설치 또는 사용 가능한 코드 페이지에 대한 정보를 검색합니다. (GetCPInfoExW)
GetCurrencyFormatA 숫자 문자열을 식별자로 지정된 로캘의 통화 문자열로 서식을 지정합니다. (ANSI)
GetCurrencyFormatEx 숫자 문자열을 이름으로 지정된 로캘의 통화 문자열로 서식을 지정합니다. 참고 애플리케이션은 Windows Vista 이상에서만 실행되도록 설계된 경우 기본 설정으로 이 함수를 GetCurrencyFormat으로 호출해야 합니다. 참고 이 함수는 예를 들어 사용자 지정 로캘로 인해 릴리스 간에 변경되는 데이터의 서식을 지정할 수 있습니다. 애플리케이션이 데이터를 유지하거나 전송해야 하는 경우 영구 로캘 데이터 사용을 참조하세요.
GetCurrencyFormatW 숫자 문자열을 식별자로 지정된 로캘의 통화 문자열로 서식을 지정합니다. (유니코드)
GetDurationFormat 시간 기간을 식별자로 지정된 로캘의 시간 문자열로 서식을 지정합니다.
GetDurationFormatEx 시간 기간을 이름으로 지정된 로캘의 시간 문자열로 서식을 지정합니다.
GetFileMUIInfo 파일에 대한 리소스 관련 정보를 검색합니다.
GetFileMUIPath 제공된 LN 파일과 연결된 모든 언어별 리소스 파일의 경로를 검색합니다. 애플리케이션은 각 리소스 파일의 경로를 가져오기 위해 이 함수를 반복적으로 호출해야 합니다.

GetGeoInfoA 지정된 지리적 위치에 대한 정보를 검색합니다. (ANSI)
GetGeoInfoEx ISO(국제 표준화 기구) 3166-1 코드 또는 숫자 UN(UN) 시리즈 M, 번호 49(M.49) 코드를 사용하여 지정한 지리적 위치에 대한 정보를 검색합니다.
GetGeoInfoW 지정된 지리적 위치에 대한 정보를 검색합니다. (유니코드)
GetLocaleInfoA 식별자가 지정한 로캘에 대한 정보를 검색합니다. (ANSI)
GetLocaleInfoEx 이름으로 지정된 로캘에 대한 정보를 검색합니다. 참고 애플리케이션은 Windows Vista 이상에서만 실행되도록 설계된 경우 기본 설정으로 이 함수를 GetLocaleInfo로 호출해야 합니다. 참고 이 함수는 예를 들어 사용자 지정 로캘로 인해 릴리스 간에 변경되는 데이터를 검색할 수 있습니다. 애플리케이션이 데이터를 유지하거나 전송해야 하는 경우 영구 로캘 데이터 사용을 참조하세요.
GetLocaleInfoW 식별자가 지정한 로캘에 대한 정보를 검색합니다. (유니코드)
GetNLSVersion 식별자가 지정한 로캘에 대해 지정된 NLS 기능의 현재 버전에 대한 정보를 검색합니다. 참고 상호 운용성을 위해 Microsoft는 새 로캘에 대한 로캘 식별자 대신 로캘 이름을 사용하도록 마이그레이션하기 때문에 애플리케이션에서 GetNLSVersionEx 함수를 GetNLSVersion에 선호해야 합니다. 이 권장 사항은 특히 GetNLSVersionEx가 정렬 동작이 변경되었는지 확인하기에 충분한 정보를 검색하는 사용자 지정 로캘에 적용됩니다. Windows Vista 이상에서만 실행되는 모든 애플리케이션은 GetNLSVersionEx를 사용하거나 GetNLSVersion을 호출하여 추가 정렬 버전 관리 데이터를 가져올 때 적어도 NLSVERSIONINFOEX 구조를 전달해야 합니다.
GetNLSVersionEx 이름으로 지정된 로캘에 대해 지정된 NLS 기능의 현재 버전에 대한 정보를 검색합니다. 참고 애플리케이션은 Windows Vista 이상에서만 실행되도록 설계된 경우 GetNLSVersion을 기본 설정으로 이 함수를 호출해야 합니다.
GetNumberFormatA 숫자 문자열을 식별자로 지정된 로캘에 대해 사용자 지정된 숫자 문자열로 서식을 지정합니다. (ANSI)

GetNumberFormatEx 숫자 문자열을 이름으로 지정된 로캘에 대해 사용자 지정된 숫자 문자열로 서식을 지정합니다. 참고 애플리케이션은 Windows Vista 이상에서만 실행되도록 설계된 경우 GetNumberFormat을 기본 설정으로 이 함수를 호출해야 합니다. 참고 이 함수는 예를 들어 사용자 지정 로캘로 인해 릴리스 간에 변경되는 데이터의 서식을 지정할 수 있습니다. 애플리케이션이 데이터를 유지하거나 전송해야 하는 경우 영구 로캘 데이터 사용을 참조하세요.
GetNumberFormatW 숫자 문자열을 식별자로 지정된 로캘에 대해 사용자 지정된 숫자 문자열로 서식을 지정합니다. (유니코드)
GetOEMCP 운영 체제에 대한 현재 OEM(원본 장비 제조업체) 코드 페이지 식별자를 반환합니다.
GetProcessPreferredUILanguages 프로세스 기본 설정 UI 언어를 검색합니다. 자세한 내용은 사용자 인터페이스 언어 관리를 참조하세요.
GetStringScripts 지정된 유니코드 문자열에 사용되는 스크립트 목록을 제공합니다.
GetStringTypeA 더 이상 사용되지 않습니다. (GetStringTypeA)
GetSystemDefaultLangID 시스템 로캘의 언어 식별자를 반환합니다.
GetSystemDefaultLCID 시스템 로캘의 로캘 식별자를 반환합니다. 참고 Windows Vista 이상에서만 실행되는 모든 애플리케이션은 이 함수를 기본 설정으로 GetSystemDefaultLocaleName을 사용해야 합니다.
GetSystemDefaultLocaleName 시스템 기본 로캘 이름을 검색합니다. 참고 애플리케이션은 이 함수보다 기본 설정으로 GetUserDefaultLocaleName을 호출하는 것이 좋습니다.
GetSystemDefaultUILanguage Windows Vista 이상에서 "설치 언어"라고도 하는 운영 체제의 시스템 기본 UI 언어에 대한 언어 식별자를 검색합니다. 자세한 내용은 사용자 인터페이스 언어 관리를 참조하세요.

GetSystemPreferredUILanguages 시스템 기본 설정 UI 언어를 검색합니다. 자세한 내용은 사용자 인터페이스 언어 관리를 참조하세요.
GetThreadLocale 호출 스레드에 대한 현재 로캘의 로캘 식별자를 반환합니다. 참고 이 함수는 예를 들어 사용자 지정 로캘로 인해 릴리스 간에 변경되는 데이터를 검색할 수 있습니다.
GetThreadPreferredUILanguages 현재 스레드에 대한 스레드 기본 설정 UI 언어를 검색합니다. 자세한 내용은 사용자 인터페이스 언어 관리를 참조하세요.
GetThreadUILanguage 현재 스레드에 대한 첫 번째 사용자 인터페이스 언어의 언어 식별자를 반환합니다.
GetUILanguageInfo 설치된 UI 언어에 대한 다양한 정보를 검색합니다.
GetUserDefaultGeoName 사용자의 기본 지리적 위치에 대한 두 글자 ISO(국제 표준화 기구) 3166-1 코드 또는 숫자 UN(UN) 시리즈 M, 번호 49(M.49) 코드를 검색합니다.
GetUserDefaultLangID 현재 사용자에 대한 지역 형식 설정의 언어 식별자를 반환합니다.
GetUserDefaultLCID 사용자 기본 로캘에 대한 로캘 식별자를 반환합니다. 주의 사용자 기본 로캘이 사용자 지정 로캘인 경우 애플리케이션은 값을 사용하여 데이터에 정확하게 태그를 지정하거나 교환할 수 없습니다.
GetUserDefaultLocaleName 사용자 기본 로캘 이름을 검색합니다. 참고 애플리케이션은 Windows Vista 이상에서만 실행되도록 설계된 경우 기본 설정으로 이 함수를 GetUserDefaultLCID로 호출해야 합니다.
GetUserDefaultUILanguage 현재 사용자의 사용자 UI 언어에 대한 언어 식별자를 반환합니다.
GetUserGeoID 사용자의 지리적 위치에 대한 정보를 검색합니다. 자세한 내용은 지리적 위치 표를 참조하세요.

GetUserPreferredUILanguages 사용자 기본 설정 UI 언어에 대한 정보를 검색합니다. 자세한 내용은 사용자 인터페이스 언어 관리를 참조하세요.
IdnToAscii IDN(국제화된 도메인 이름) 또는 다른 국제화된 레이블을 Punycode 전송 인코딩 구문의 이름을 나타내는 ASCII 문자열의 유니코드(와이드 문자) 표현으로 변환합니다.
IdnToNameprepUnicode IDN(국제화된 도메인 이름) 또는 다른 국제화된 레이블을 네트워크 작업 그룹 RFC 3491에서 지정한 NamePrep 양식으로 변환하지만 Punycode로의 추가 변환은 수행하지 않습니다.
IdnToUnicode IDN(국제화된 도메인 이름) 또는 다른 국제화된 레이블의 Punycode 형식을 일반 유니코드 UTF-16 인코딩 구문으로 변환합니다.
IS_HIGH_SURROGATE 문자가 utF-16 상위 서로게이트 코드 지점인지 여부를 결정합니다(0xd800부터 0xdbff까지 포함).
IS_LOW_SURROGATE 문자가 utF-16 하위 서로게이트 코드 지점인지 여부를 확인합니다(0xdc00부터 0xdfff까지 포함).
IS_SURROGATE_PAIR 지정된 코드 단위가 UTF-16 서로게이트 쌍을 구성하는지 여부를 결정합니다.
IsDBCSLeadByte 지정된 문자가 시스템 기본 Windows ANSI 코드 페이지(CP_ACP)의 리드 바이트인지 확인합니다. 리드 바이트는 코드 페이지의 DBCS(더블 바이트 문자 집합)에서 2 바이트 문자의 첫 번째 바이트입니다.
IsDBCSLeadByteEx 지정된 문자가 잠재적으로 리드 바이트인지 확인합니다. 리드 바이트는 코드 페이지의 DBCS(더블 바이트 문자 집합)에서 2 바이트 문자의 첫 번째 바이트입니다.
IsNLSDefinedString 문자열의 각 문자에 지정된 NLS 기능에 대해 정의된 결과가 있는지 확인합니다.
IsNormalizedString

유니코드 4.0 TR#15에 따라 문자열이 정규화되었는지 확인합니다. 자세한 내용은 유니코드 정규화를 사용하여 문자열 표시를 참조하세요.
IsValidCodePage 지정된 코드 페이지가 유효한지 확인합니다.
IsValidLanguageGroup 언어 그룹이 운영 체제에 설치되거나 지원되는지 확인합니다. 자세한 내용은 NLS 용어를 참조하세요.
IsValidLocale 지정된 로캘이 운영 체제에 설치되거나 지원되는지 확인합니다. 자세한 내용은 로캘 및 언어를 참조하세요.
IsValidLocaleName 운영 체제에 설치되거나 지원되는 로캘에 대해 지정된 로캘 이름이 유효한지 확인합니다. 참고 Windows Vista 이상에서만 실행되는 애플리케이션은 기본 설정으로 이 함수를 IsValidLocale로 호출하여 추가 로캘의 유효성을 확인해야 합니다.
IsValidNLSVersion NLS 버전이 지정된 NLS 함수에 유효한지 확인합니다.
LCIDToLocaleName 로캘 식별자를 로캘 이름으로 변환합니다.
LCMapStringA 식별자로 지정된 로캘의 경우 지정된 변환을 사용하여 한 입력 문자 문자열을 다른 입력 문자열에 매핑하거나 입력 문자열에 대한 정렬 키를 생성합니다. (ANSI)
LCMapStringEx 이름으로 지정된 로캘의 경우 지정된 변환을 사용하여 입력 문자 문자열을 다른 에 매핑하거나 입력 문자열에 대한 정렬 키를 생성합니다. 참고 애플리케이션은 Windows Vista 이상에서만 실행되도록 설계된 경우 LCMapString을 기본 설정으로 이 함수를 호출해야 합니다.
LCMapStringW 식별자로 지정된 로캘의 경우 지정된 변환을 사용하여 한 입력 문자 문자열을 다른 입력 문자열에 매핑하거나 입력 문자열에 대한 정렬 키를 생성합니다. (유니코드)
LoadStringByReference

지원되지 않음 LoadStringByReference를 변경하거나 사용할 수 없습니다. 대신 SHLoadIndirectString을 사용합니다.
LocaleNameToLCID 로캘 이름을 로캘 식별자로 변환합니다.
NormalizeString 유니코드 4.0 TR#15에 따라 텍스트 문자열의 문자를 정규화합니다. 자세한 내용은 유니코드 정규화를 사용하여 문자열 표시를 참조하세요.
NotifyUILanguageChange 지원되지 않음 NotifyUILanguageChange를 변경하거나 사용할 수 없습니다.
ResolveLocaleName 제공된 이름에 대해 가능한 로캘 이름 일치를 찾습니다.
SetCalendarInfoA 일정에 대한 로캘 정보의 항목을 설정합니다. 자세한 내용은 날짜 및 일정을 참조하세요. (ANSI)
SetCalendarInfoW 일정에 대한 로캘 정보의 항목을 설정합니다. 자세한 내용은 날짜 및 일정을 참조하세요. (유니코드)
SetLocaleInfoA 현재 로캘의 사용자 재정의 부분에 있는 정보 항목을 설정합니다. 이 함수는 시스템 기본값을 설정하지 않습니다. (ANSI)
SetLocaleInfoW 현재 로캘의 사용자 재정의 부분에 있는 정보 항목을 설정합니다. 이 함수는 시스템 기본값을 설정하지 않습니다. (유니코드)
SetProcessPreferredUILanguages 애플리케이션 프로세스에 대한 프로세스 기본 설정 UI 언어를 설정합니다. 자세한 내용은 사용자 인터페이스 언어 관리를 참조하세요.
SetThreadLocale 호출 스레드의 현재 로캘을 설정합니다.
SetThreadPreferredUILanguages

현재 스레드에 대한 스레드 기본 설정 UI 언어를 설정합니다. 자세한 내용은 사용자 인터페이스 언어 관리를 참조하세요.
SetThreadUILanguage 현재 스레드에 대한 사용자 인터페이스 언어를 설정합니다.
SetUserGeoID 사용자의 지리적 위치 식별자를 설정합니다. 이 식별자에는 지리적 위치 표에 설명된 값 중 하나가 있어야 합니다.
SetUserGeoName 현재 사용자의 지리적 위치를 지정된 ISO(국제 표준화 기구) 3166-1 코드 또는 숫자 UN(UN) 시리즈 M, 번호 49(M.49) 코드로 설정합니다.
VerifyScripts 두 개의 열거형 스크립트 목록을 비교합니다.

콜백 함수

GEO_ENUMNAMEPROC EnumSystemGeoNames 함수에서 제공하는 열거된 지리적 위치 정보를 처리하는 애플리케이션 정의 콜백 함수입니다.
LANGGROUPLOCALE_ENUMPROCA EnumLanguageGroupLocales 함수에서 제공하는 열거형 언어 그룹 로캘 정보를 처리하는 애플리케이션 정의 콜백 함수입니다. (ANSI)
LANGGROUPLOCALE_ENUMPROCW EnumLanguageGroupLocales 함수에서 제공하는 열거형 언어 그룹 로캘 정보를 처리하는 애플리케이션 정의 콜백 함수입니다. (유니코드)
LOCALE_ENUMPROCEX EnumSystemLocalesEx 함수에서 제공하는 열거형 로캘 정보를 처리하는 애플리케이션 정의 콜백 함수입니다.
UILANGUAGE_ENUMPROCA

EnumUILanguages 함수에서 제공하는 열거형 사용자 인터페이스 언어 정보를 처리하는 애플리케이션 정의 콜백 함수입니다. (ANSI)
UILANGUAGE_ENUMPROCW EnumUILanguages 함수에서 제공하는 열거형 사용자 인터페이스 언어 정보를 처리하는 애플리케이션 정의 콜백 함수입니다. (유니코드)

구조체

CPINFO 코드 페이지에 대한 정보를 포함합니다. 이 구조체는 GetCPInfo 함수에서 사용됩니다.
CPINFOEXA 코드 페이지에 대한 정보를 포함합니다. 이 구조체는 GetCPInfoEx 함수에서 사용됩니다. (ANSI)
CPINFOEXW 코드 페이지에 대한 정보를 포함합니다. 이 구조체는 GetCPInfoEx 함수에서 사용됩니다. (유니코드)
CURRENCYFMTA 통화 문자열의 형식을 정의하는 정보를 포함합니다. GetCurrencyFormat 함수는 이 정보를 사용하여 지정된 로캘에 대한 통화 문자열을 사용자 지정합니다. (ANSI)
CURRENCYFMTW 통화 문자열의 형식을 정의하는 정보를 포함합니다. GetCurrencyFormat 함수는 이 정보를 사용하여 지정된 로캘에 대한 통화 문자열을 사용자 지정합니다. (유니코드)
FILEMUIINFO MUI 사용과 관련된 파일에 대한 정보를 포함합니다.
NLSVERSIONINFO NLSVERSIONINFO 구조체(winnls.h)는 더 이상 사용되지 않으며 사용하면 안 됩니다.
NLSVERSIONINFOEX NLS 기능에 대한 버전 정보를 포함합니다.
NUMBERFMTA

숫자 문자열의 형식을 정의하는 정보를 포함합니다. GetNumberFormat 함수는 이 정보를 사용하여 지정된 로캘에 대한 숫자 문자열을 사용자 지정합니다. (ANSI)
NUMBERFMTW 숫자 문자열의 형식을 정의하는 정보를 포함합니다. GetNumberFormat 함수는 이 정보를 사용하여 지정된 로캘에 대한 숫자 문자열을 사용자 지정합니다. (유니코드)

열거형

NORM_FORM 지원되는 정규화 양식을 지정합니다.
SYSGEOCLASS 지리적 위치 클래스를 지정합니다.
SYSGEOTYPE GetGeoInfo 또는 GetGeoInfoEx 함수에서 요청된 지리적 위치 정보의 유형을 정의합니다.
SYSNLS_FUNCTION NLS 함수 기능을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winnls32.h 헤더

아티클2023. 08. 27.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

winnls32.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WINNLSEnableIME IME(입력 방법 편집기)를 일시적으로 사용하거나 사용하지 않도록 설정하고 동시에 IME가 소유한 모든 창의 표시를 켜거나 끕니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winnt.h 헤더

아티클2023. 03. 08.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 애플리케이션 설치 및 서비스
- Backup
- 데이터 액세스 및 스토리지
- 개발자 정보
- DXGI
- 하드웨어 카운터 프로파일링
- Windows 애플리케이션에 대한 국제화
- 커널 모드 드라이버 참조
- 메뉴 및 기타 리소스
- 보안 및 ID
- 시스템 서비스
- Windows 셸
- Windows Management Instrumentation
- Windows 런타임 C++ 참조

winnt.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

_InlineInterlockedAdd 지정된 LONG 값에 대해 원자성 추가 작업을 수행합니다. 이 작업은 메모리 정렬 의미 체계 획득을 사용하여 수행됩니다.
_InlineInterlockedAdd64 지정된 LONG64 값에 대해 원자성 추가 작업을 수행합니다.
_interlockedbittestandreset 지정된 LONG 값의 지정된 비트를 테스트하고 0으로 설정합니다. 이 작업은 원자성 작업입니다.
_interlockedbittestandreset64 지정된 LONG64 값의 지정된 비트를 테스트하고 0으로 설정합니다. 이 작업은 원자성 작업입니다.

_interlockedbittestandset 지정된 LONG 값의 지정된 비트를 테스트하고 1로 설정합니다. 이 작업은 원자성 작업입니다.
_interlockedbittestandset64 지정된 LONG64 값의 지정된 비트를 테스트하고 1로 설정합니다. 이 작업은 원자성 작업입니다.
C_ASSERT 컴파일 시간에 어설션을 확인합니다.
FIELD_OFFSET FIELD_OFFSET 매크로는 알려진 구조체 형식의 명명된 필드의 바이트 오프셋을 반환합니다. (FIELD_OFFSET 매크로)
GetCurrentFiber 현재 파이버의 주소를 검색합니다.
GetFiberData 현재 파이버와 연결된 파이버 데이터를 검색합니다.
Int32x32To64 서명된 32비트 정수 2개를 곱하여 서명된 64비트 정수 결과를 반환합니다.
Int64ShllMod32 부호 없는 64비트 정수 값에 대해 왼쪽 논리 시프트 작업을 수행합니다. 함수는 시프트 수가 0-31 범위에 있는 왼쪽 논리 교대조에 대해 향상된 시프트 코드를 제공합니다.
Int64ShraMod32 서명된 64비트 정수 값에 대해 올바른 산술 시프트 작업을 수행합니다. 함수는 시프트 수가 0-31 범위에 있는 오른쪽 산술 교대조에 대해 향상된 시프트 코드를 제공합니다.
Int64ShrlMod32 부호 없는 64비트 정수 값에 대해 올바른 논리적 시프트 작업을 수행합니다. 함수는 시프트 수가 0-31 범위에 있는 오른쪽 논리 교대조에 대해 향상된 시프트 코드를 제공합니다.
InterlockedAdd 지정된 LONG 값에 대해 원자성 추가 작업을 수행합니다.

InterlockedAnd 지정된 LONG 값에 대해 원자성 AND 연산을 수행합니다.
InterlockedAnd16 지정된 SHORT 값에 대해 원자성 AND 연산을 수행합니다.
InterlockedAnd64 지정된 LONGLONG 값에 대해 원자성 AND 연산을 수행합니다.
InterlockedAnd8 지정된 char 값에 대해 원자성 AND 연산을 수행합니다.
InterlockedCompareExchange 지정된 값에 대해 원자성 비교 및 교환 작업을 수행합니다. 함수는 두 개의 지정된 32비트 값과 교환을 비교 결과에 따라 다른 32비트 값과 비교합니다.
InterlockedCompareExchange128 지정된 값에 대해 원자성 비교 및 교환 작업을 수행합니다. 함수는 두 개의 지정된 128비트 값과 교환을 비교 결과에 따라 다른 128비트 값과 비교합니다.
InterlockedCompareExchange16 지정된 값에 대해 원자성 비교 및 교환 작업을 수행합니다. 함수는 두 개의 지정된 16비트 값과 교환을 비교 결과에 따라 다른 16비트 값과 비교합니다.
InterlockedCompareExchange64 지정된 값에 대해 원자성 비교 및 교환 작업을 수행합니다. 함수는 두 개의 지정된 64비트 값과 교환을 비교 결과에 따라 다른 64비트 값과 비교합니다.
InterlockedCompareExchangePointer 지정된 값에 대해 원자성 비교 및 교환 작업을 수행합니다. 함수는 두 개의 지정된 포인터 값과 교환을 비교 결과에 따라 다른 포인터 값과 비교합니다.
InterlockedDecrement 지정된 32비트 변수의 값을 원자성 연산으로 감소(1씩 감소)합니다.
InterlockedDecrement16 지정된 16비트 변수의 값을 원자성 연산으로 감소(1씩 감소)합니다.

InterlockedDecrement64 지정된 64비트 변수의 값을 원자성 연산으로 감소(1씩 감소)합니다.
InterlockedExchange 32비트 변수를 지정된 값으로 원자성 연산으로 설정합니다.
InterlockedExchange16 16비트 변수를 지정된 값으로 원자성 연산으로 설정합니다.
InterlockedExchange64 64비트 변수를 지정된 값으로 원자성 연산으로 설정합니다.
InterlockedExchange8 8비트 변수를 지정된 값으로 원자성 연산으로 설정합니다.
InterlockedExchangeAdd 두 개의 32비트 값의 원자성 추가를 수행합니다.
InterlockedExchangeAdd64 두 64비트 값의 원자성 추가를 수행합니다.
InterlockedExchangePointer 주소 쌍을 원자적으로 교환합니다.
InterlockedIncrement 지정된 32비트 변수의 값을 원자성 연산으로 증분(1씩 증가)합니다.
InterlockedIncrement16 지정된 16비트 변수의 값을 원자성 연산으로 증분(1씩 증가)합니다.
InterlockedIncrement64 지정된 64비트 변수의 값을 원자성 연산으로 증분(1씩 증가)합니다.
InterlockedOr 지정된 LONG 값에 대해 원자성 OR 연산을 수행합니다.

InterlockedOr16 지정된 SHORT 값에 대해 원자성 OR 연산을 수행합니다.
InterlockedOr64 지정된 LONGLONG 값에 대해 원자성 OR 연산을 수행합니다.
InterlockedOr8 지정된 char 값에 대해 원자성 OR 연산을 수행합니다.
InterlockedXor 지정된 LONG 값에 대해 원자성 XOR 연산을 수행합니다.
InterlockedXor16 지정된 SHORT 값에 대해 원자성 XOR 연산을 수행합니다.
InterlockedXor64 지정된 LONGLONG 값에 대해 원자성 XOR 연산을 수행합니다.
InterlockedXor8 지정된 char 값에 대해 원자성 XOR 연산을 수행합니다.
IsReparseTagMicrosoft 재분석 지점 태그가 Microsoft 재분석 지점을 나타내는지 여부를 결정합니다.
IsReparseTagNameSurrogate 태그의 연결된 재분석 지점이 다른 명명된 엔터티(예: 탑재된 폴더)의 서로게이트인지 여부를 결정합니다.
LANGIDFROMLCID 로캘 식별자에서 언어 식별자를 검색합니다.
MAKELANGID 기본 언어 식별자 및 하위 언어 식별자에서 언어 식별자를 만듭니다.
MAKELCID 언어 식별자 및 정렬 순서 식별자에서 로캘 식별자를 만듭니다.

MAKESORTLCID 언어 식별자, 정렬 순서 식별자 및 정렬 버전에서 LCID(로캘 식별자)를 생성합니다.
MemoryBarrier CPU가 읽기 및 쓰기 작업을 다시 정렬하지 못하도록 하는 하드웨어 메모리 장벽(펜스)을 만듭니다. 컴파일러가 읽기 및 쓰기 작업을 다시 정렬하지 못할 수도 있습니다.
Multiply128 64비트 정수 2개를 곱하여 128비트 정수 생성
MultiplyExtract128 64비트 정수 2개를 곱하여 128비트 정수를 생성하고, 제품을 지정된 비트 수만큼 오른쪽으로 이동하고, 결과의 낮은 64비트를 반환합니다.
MultiplyHigh 64비트 정수 2개를 곱하여 128비트 정수와 높은 64비트를 가져옵니다.
NtCurrentTeb NtCurrentTeb 루틴은 현재 스레드의 TEB(스레드 환경 블록)에 대한 포인터를 반환합니다.
PopulationCount64 부호 없는 64비트 정수의 1비트(모집단 수)의 수를 계산합니다.
PreFetchCacheLine 가까운 장래에 캐시 라인이 필요할 것임을 프로세서에 나타냅니다.
PRIMARYLANGID 언어 식별자에서 기본 언어 식별자를 추출합니다.
RtlAddFunctionTable 동적 함수 테이블 목록에 동적 함수 테이블을 추가합니다. (RtlAddFunctionTable)
RtlAddGrowableFunctionTable 코드가 포함된 메모리 영역을 나타내는 동적 함수 테이블을 시스템에 알립니다.
RtlCaptureContext 호출자의 컨텍스트에서 컨텍스트 레코드를 검색합니다.

RtlCaptureStackBackTrace RtlCaptureStackBackTrace 루틴은 스택을 위로 이동하고 각 프레임에 대한 정보를 기록하여 스택 뒤로 추적을 캡처합니다.
RtlConvertDeviceFamilyInfoToString 디바이스 패밀리 정보의 문자열 표현을 검색합니다.
RtlDeleteFunctionTable 동적 함수 테이블 목록에서 동적 함수 테이블을 제거합니다.
RtlDeleteGrowableFunctionTable 이전에 보고된 동적 함수 테이블이 더 이상 사용되지 않음을 시스템에 알립니다.
RtlFirstEntrySList 연결된 목록의 첫 번째 항목을 검색합니다. 목록에 대한 액세스는 다중 프로세서 시스템에서 동기화됩니다.
RtlGrowFunctionTable 동적 함수 테이블의 크기가 증가했음을 보고합니다.
RtlInitializeSListHead Singly 연결된 목록의 헤드를 초기화합니다. (RtlInitializeSListHead)
RtlInstallFunctionTableCallback 동적 함수 테이블 목록에 동적 함수 테이블을 추가합니다. (RtlInstallFunctionTableCallback)
RtlInterlockedFlushSList 연결된 목록에서 모든 항목을 제거합니다. 목록에 대한 액세스는 다중 프로세서 시스템에서 동기화됩니다. (RtlInterlockedFlushSList)
RtlInterlockedPopEntrySList 연결된 목록의 앞면에서 항목을 제거합니다. 목록에 대한 액세스는 다중 프로세서 시스템에서 동기화됩니다. (RtlInterlockedPopEntrySList)
RtlInterlockedPushEntrySList 연결된 목록의 맨 앞에 항목을 삽입합니다. 목록에 대한 액세스는 다중 프로세서 시스템에서 동기화됩니다. (RtlInterlockedPushEntrySList)

RtlIsEcCode 제공된 포인터가 가리키는 코드가 ARM64EC(ARM 에뮬레이션 호환)인지 여부를 나타내는 값을 반환합니다.
RtlLookupFunctionEntry 활성 함수 테이블에서 지정된 PC 값에 해당하는 항목을 검색합니다.
RtlPcToFileHeader 지정된 PC 값이 포함된 이미지의 기본 주소를 검색합니다.
RtlQueryDepthSList 지정된 singly 연결된 목록의 항목 수를 검색합니다. (RtlQueryDepthSList)
RtlRestoreContext 호출자의 컨텍스트를 지정된 컨텍스트 레코드로 복원합니다.
RtlUnwind 프로시저 호출 프레임의 해제를 시작합니다. (RtlUnwind)
RtlUnwind2 프로시저 호출 프레임의 해제를 시작합니다. (RtlUnwind2)
RtlUnwindEx 프로시저 호출 프레임의 해제를 시작합니다.
RtlVirtualUnwind 지정된 함수 컨텍스트 앞에 오는 함수의 호출 컨텍스트를 검색합니다.
ShiftLeft128 128비트 왼쪽으로 이동합니다.
ShiftRight128 오른쪽으로 128비트 이동합니다.
SORTIDFROMLCID 로캘 식별자에서 정렬 순서 식별자를 검색합니다.

SORTVERSIONFROMLCID 로캘 식별자에서 정렬 버전을 검색합니다.
SUBLANGID 언어 식별자에서 하위 언어 식별자를 추출합니다.
텍스트 컴파일하는 동안 유니코드가 전처리기 지시문에 의해 정의되는 경우 문자열을 유니코드로 식별합니다. 그렇지 않으면 매크로는 문자열을 ANSI 문자열로 식별합니다.
TpDestroyCallbackEnviron 지정된 콜백 환경을 삭제합니다. 새 스레드 풀 개체를 만드는 데 콜백 환경이 더 이상 필요하지 않은 경우 이 함수를 호출합니다. (TpDestroyCallbackEnviron)
TpInitializeCallbackEnviron 스레드 풀에 대한 콜백 환경을 초기화합니다.
TpSetCallbackActivationContext 콜백 환경에 활성화 컨텍스트를 할당합니다.
TpSetCallbackCleanupGroup 지정된 정리 그룹을 지정된 콜백 환경과 연결합니다. (TpSetCallbackCleanupGroup)
TpSetCallbackFinalizationCallback 콜백 환경이 종료될 때 호출할 함수를 나타냅니다.
TpSetCallbackLongFunction 이 콜백 환경과 연결된 콜백이 빠르게 반환되지 않을 수 있음을 나타냅니다. (TpSetCallbackLongFunction)
TpSetCallbackNoActivationContext 콜백 환경에 활성화 컨텍스트가 없음을 나타냅니다.
TpSetCallbackPersistent 콜백이 영구 스레드에서 실행되도록 지정합니다. (TpSetCallbackPersistent)

TpSetCallbackPriority 동일한 스레드 풀의 다른 작업 항목을 기준으로 콜백 함수의 우선 순위를 지정합니다. (TpSetCallbackPriority)
TpSetCallbackRaceWithDll 지정된 DLL이 미해결 콜백이 있는 한 로드된 상태로 유지되도록 합니다. (TpSetCallbackRaceWithDll)
TpSetCallbackThreadpool 콜백 환경에 스레드 풀을 할당합니다.
UInt32x32To64 부호 없는 32비트 정수 2개를 곱하여 부호 없는 64비트 정수 결과를 반환합니다.
UnsignedMultiply128 부호 없는 64비트 정수 2개를 곱하여 부호 없는 128비트 정수 를 생성합니다.
UnsignedMultiplyExtract128 부호 없는 64비트 정수 2개를 곱하여 부호 없는 128비트 정수를 생성하고, 지정된 비트 수만큼 제품을 오른쪽으로 이동하고, 결과의 낮은 64비트를 반환합니다.
UnsignedMultiplyHigh 64비트 정수 2개를 곱하여 128비트 정수 생성 및 부호 없는 상위 64비트를 가져옵니다.
VER_SET_CONDITION 지정된 운영 체제 버전 특성에 사용할 비교 연산자를 나타내도록 64비트 값의 비트를 설정합니다. 이 매크로는 VerifyVersionInfo 함수의 dwlConditionMask 매개 변수를 빌드하는 데 사용됩니다.
VerSetConditionMask 지정된 운영 체제 버전 특성에 사용할 비교 연산자를 나타내도록 64비트 값의 비트를 설정합니다. 이 함수는 VerifyVersionInfo 함수의 dwlConditionMask 매개 변수를 빌드하는 데 사용됩니다.
YieldProcessor 대기 중인 스레드에 리소스를 제공하도록 프로세서에 신호를 보냅니다.

콜백 함수

PAPCFUNC 애플리케이션 정의 완료 루틴입니다. QueueUserAPC 함수를 호출할 때 이 주소를 지정합니다.
PFLS_CALLBACK_FUNCTION 애플리케이션 정의 함수입니다. FLS 슬롯이 사용 중인 경우 FlsCallback은 파이버 삭제, 스레드 종료 및 FLS 인덱스가 해제되면 호출됩니다.
PSECURE_MEMORY_CACHE_CALLBACK 보안 메모리 범위가 해제되거나 보호가 변경될 때 호출되는 AddSecureMemoryCacheCallback 함수에 이전에 등록된 애플리케이션 정의 함수입니다.
PVECTORED_EXCEPTION_HANDLER 벡터 예외 처리기 역할을 하는 애플리케이션 정의 함수입니다.
RTL_UMS_SCHEDULER_ENTRY_POINT UMS 완료 목록과 연결된 애플리케이션 정의 UMS(사용자 모드 예약) 스케줄러 진입점 함수입니다.

구조체

ACCESS_ALLOWED_ACE 개체에 대한 액세스를 제어하는 DACL(임의 액세스 제어 목록)에 대한 ACE(액세스 제어 항목)를 정의합니다. 액세스 허용 ACE를 사용하면 SID(보안 식별자)로 식별된 특정 트러스티에 대한 개체에 액세스할 수 있습니다.
ACCESS_ALLOWED_CALLBACK_ACE ACCESS_ALLOWED_CALLBACK_ACE 구조체는 개체에 대한 액세스를 제어하는 임의 액세스 제어 목록에 대한 액세스 제어 항목을 정의합니다.
ACCESS_ALLOWED_CALLBACK_OBJECT_ACE 개체, 속성 집합 또는 속성에 대한 허용된 액세스를 제어하는 ACE(액세스 제어 항목)를 정의합니다.
ACCESS_ALLOWED_OBJECT_ACE 개체, 속성 집합 또는 속성에 대한 허용된 액세스를 제어하는 ACE(액세스 제어 항목)를 정의합니다.

ACCESS_DENIED_ACE 개체에 대한 액세스를 제어하는 DACL(임의 액세스 제어 목록)에 대한 ACE(액세스 제어 항목)를 정의합니다. 액세스 거부 ACE는 SID(보안 식별자)로 식별된 특정 트러스티에 대한 개체에 대한 액세스를 거부합니다.
ACCESS_DENIED_CALLBACK_ACE ACCESS_DENIED_CALLBACK_ACE 구조체는 개체에 대한 액세스를 제어하는 임의 액세스 제어 목록에 대한 액세스 제어 항목을 정의합니다.
ACCESS_DENIED_CALLBACK_OBJECT_ACE ACCESS_DENIED_CALLBACK_OBJECT_ACE 구조체는 개체, 속성 집합 또는 속성에 대한 거부된 액세스를 제어하는 액세스 제어 항목을 정의합니다.
ACCESS_DENIED_OBJECT_ACE 개체, 속성 집합 또는 속성에 대한 거부된 액세스를 제어하는 ACE(액세스 제어 항목)를 정의합니다.
ACE_HEADER ACE(액세스 제어 항목)의 형식과 크기를 정의합니다.
ACL ACL(액세스 제어 목록)의 헤더입니다.
ACL_REVISION_INFORMATION ACL 구조체에 대한 수정 정보를 포함합니다.
ACL_SIZE_INFORMATION ACL 구조체의 크기에 대한 정보를 포함합니다.
ACTIVATION_CONTEXT_ASSEMBLY_DETAILED_INFORMATION ACTIVATION_CONTEXT_ASSEMBLY_DETAILED_INFORMATION 구조체는 QueryActCtxW 함수에서 사용됩니다.
ACTIVATION_CONTEXT_COMPATIBILITY_INFORMATION ACTIVATION_CONTEXT_COMPATIBILITY_INFORMATION 구조체는 QueryActCtxW 함수에서 사용됩니다.
ACTIVATION_CONTEXT_DETAILED_INFORMATION ACTIVATION_CONTEXT_DETAILED_INFORMATION 구조체는 QueryActCtxW 함수에서 사용됩니다.

ACTIVATION_CONTEXT_QUERY_INDEX ACTIVATION_CONTEXT_QUERY_INDEX 구조체는 QueryActCtxW 함수에서 사용됩니다.
ACTIVATION_CONTEXT_RUN_LEVEL_INFORMATION ACTIVATION_CONTEXT_RUN_LEVEL_INFORMATION 구조체는 QueryActCtxW 함수에서 사용됩니다.
ADMINISTRATOR_POWER_POLICY 관리자가 전원 정책 설정을 재정의하는 것을 나타냅니다.
ARM64_NT_CONTEXT 프로세서별 레지스터 데이터를 포함합니다. 시스템은 CONTEXT 구조를 사용하여 다양한 내부 작업을 수행합니다. C
ASSEMBLY_FILE_DETAILED_INFORMATION ASSEMBLY_FILE_DETAILED_INFORMATION 구조체는 QueryActCtxW 함수에서 사용됩니다.
BATTERY_REPORTING_SCALE IOCTL_BATTERY_QUERY_STATUS 보고하는 배터리 용량의 세분성을 포함합니다.
CACHE_DESCRIPTOR 캐시 특성을 설명합니다.
CACHE_RELATIONSHIP 캐시 특성을 설명합니다. 이 구조체는 GetLogicalProcessorInformationEx 함수와 함께 사용됩니다.
CLAIM_SECURITY_ATTRIBUTE_FQBN_VALUE 정규화된 이진 이름을 지정합니다.
CLAIM_SECURITY_ATTRIBUTE_OCTET_STRING_VALUE 클레임 보안 특성의 OCTET_STRING 값 형식을 지정합니다.
CLAIM_SECURITY_ATTRIBUTE_RELATIVE_V1 직렬화된 보안 설명자 내에서 지속성을 위해 연속 메모리에 정의된 리소스 특성을 정의합니다.
CLAIM_SECURITY_ATTRIBUTE_V1 토큰 또는 권한 부여 컨텍스트와 연결할 수 있는 보안 특성을 정의합니다.

CLAIM_SECURITY_ATTRIBUTES_INFORMATION 클레임에 대한 보안 특성을 정의합니다.
COMPATIBILITY_CONTEXT_ELEMENT COMPATIBILITY_CONTEXT_ELEMENT 구조체는 QueryActCtxW 함수에서 ACTIVATION_CONTEXT_COMPATIBILITY_INFORMATION 구조체의 일부로 사용됩니다.
컨텍스트 프로세서별 레지스터 데이터를 포함합니다. 시스템은 CONTEXT 구조를 사용하여 다양한 내부 작업을 수행합니다. (CONTEXT)
ENCLAVE_CREATE_INFO_SGX Enclave 형식이 ENCLAVE_TYPE_SGX 때 Enclave를 만드는 데 사용할 아키텍처별 정보가 포함되어 있습니다. 이 정보는 Intel SGX(Software Guard Extensions) 아키텍처 확장에 대한 enclave를 지정합니다.
ENCLAVE_CREATE_INFO_VBS enclave 형식이 VBS(가상화 기반 보안) enclave를 지정하는 ENCLAVE_TYPE_VBS 때 enclave를 만드는 데 사용할 아키텍처별 정보를 포함합니다.
ENCLAVE_INIT_INFO_SGX Enclave 형식이 ENCLAVE_TYPE_SGX 때 enclave를 초기화하는 데 사용할 아키텍처별 정보를 포함합니다. 이 정보는 Intel SGX(Software Guard Extensions) 아키텍처 확장에 대한 enclave를 지정합니다.
ENCLAVE_INIT_INFO_VBS enclave 형식이 VBS(가상화 기반 보안) enclave를 지정하는 ENCLAVE_TYPE_VBS enclave를 초기화하는 데 사용할 아키텍처별 정보를 포함합니다.
EVENTLOGRECORD ReadEventLog 함수에서 반환된 이벤트 레코드에 대한 정보를 포함합니다.
EXCEPTION_POINTERS 예외에 대한 머신 독립적 설명과 예외 시 프로세서 컨텍스트에 대한 컴퓨터 종속 설명이 있는 컨텍스트 레코드가 포함된 예외 레코드를 포함합니다.
EXCEPTION_RECORD 예외를 설명합니다. (EXCEPTION_RECORD)

EXCEPTION_RECORD64 예외를 설명합니다. 전자
FILE_ID_128 128비트 파일 식별자를 정의합니다.
FILE_NOTIFY_EXTENDED_INFORMATION ReadDirectoryChangesExW 함수를 통해 찾은 변경 내용에 대해 설명합니다.
FILE_NOTIFY_INFORMATION ReadDirectoryChangesW 함수를 통해 찾은 변경 내용에 대해 설명합니다.
FILE_SEGMENT_ELEMENT FILE_SEGMENT_ELEMENT 구조체는 분산/수집 읽기/쓰기 작업에 대한 세그먼트 버퍼 구조를 나타냅니다.
FPO_DATA FPO(Frame Pointer Omission) 최적화가 사용될 때 x86 컴퓨터의 함수에 대한 스택 프레임 레이아웃을 나타냅니다. 이 구조체는 호출 프레임의 베이스를 찾는 데 사용됩니다.
GENERIC_MAPPING 개체에 대한 특정 및 표준 액세스 권한에 대한 일반 액세스 권한의 매핑을 정의합니다.
GROUP_AFFINITY 스레드의 선호도와 같은 프로세서 그룹별 선호도를 나타냅니다.
GROUP_RELATIONSHIP 프로세서 그룹에 대한 정보를 나타냅니다. 이 구조체는 GetLogicalProcessorInformationEx 함수와 함께 사용됩니다.
HARDWARE_COUNTER_DATA 하드웨어 카운터 값을 포함합니다.
HEAP_OPTIMIZE_RESOURCES_INFORMATION HeapSetInformation으로 시작된 HeapOptimizeResources 작업에 대한 플래그를 지정합니다.

IMAGE_COFF_SYMBOLS_HEADER COFF 기호 헤더를 나타냅니다.
IMAGE_DATA_DIRECTORY 데이터 디렉터리를 나타냅니다.
IMAGE_DEBUG_DIRECTORY 디버그 디렉터리 형식을 나타냅니다.
IMAGE_ENCLAVE_CONFIG32 32비트 Windows를 실행하는 시스템의 Enclave 구성 형식을 정의합니다. (32비트)
IMAGE_ENCLAVE_CONFIG64 32비트 Windows를 실행하는 시스템의 Enclave 구성 형식을 정의합니다. (64비트)
IMAGE_ENCLAVE_IMPORT Enclave에서 가져올 수 있는 이미지 배열의 항목을 정의합니다.
IMAGE_FILE_HEADER COFF 헤더 형식을 나타냅니다.
IMAGE_FUNCTION_ENTRY 함수 테이블의 항목을 나타냅니다. (IMAGE_FUNCTION_ENTRY)
IMAGE_FUNCTION_ENTRY64 함수 테이블의 항목을 나타냅니다. 나
IMAGE_LOAD_CONFIG_DIRECTORY32 이미지의 로드 구성 데이터를 포함합니다. (32비트)
IMAGE_LOAD_CONFIG_DIRECTORY64 이미지의 로드 구성 데이터를 포함합니다. (64비트)
IMAGE_NT_HEADERS32 PE 헤더 형식을 나타냅니다. (32비트)

IMAGE_NT_HEADERS64 PE 헤더 형식을 나타냅니다. (64비트)
IMAGE_OPTIONAL_HEADER32 선택적 헤더 형식을 나타냅니다. (32비트)
IMAGE_OPTIONAL_HEADER64 선택적 헤더 형식을 나타냅니다. (64비트)
IMAGE_SECTION_HEADER 이미지 섹션 헤더 형식을 나타냅니다.
IO_COUNTERS 프로세스 또는 작업 개체에 대한 I/O 회계 정보를 포함합니다.
JOBOBJECT_ASSOCIATE_COMPLETION_PORT 완료 포트를 작업과 연결하는 데 사용되는 정보를 포함합니다.
JOBOBJECT_BASIC_ACCOUNTING_INFORMATION 작업 개체에 대한 기본 회계 정보를 포함합니다.
JOBOBJECT_BASIC_AND_IO_ACCOUNTING_INFORMATION 작업 개체에 대한 기본 회계 및 I/O 회계 정보를 포함합니다.
JOBOBJECT_BASIC_LIMIT_INFORMATION 작업 개체에 대한 기본 제한 정보를 포함합니다.
JOBOBJECT_BASIC_PROCESS_ID_LIST 작업 개체의 프로세스 식별자 목록을 포함합니다.
JOBOBJECT_BASIC_UI_RESTRICTIONS 작업 개체에 대한 기본 사용자 인터페이스 제한을 포함합니다.
JOBOBJECT_CPU_RATE_CONTROL_INFORMATION 작업 개체에 대한 CPU 속도 제어 정보를 포함합니다. 이 구조체는 JobObjectCpuRateControlInformation 정보 클래스와 함께 SetInformationJobObject 및 QueryInformationJobObject 함수에서 사용됩니다.

JOBOBJECT_END_OF_JOB_TIME_INFORMATION 작업 종료 시간 제한을 초과할 때 시스템에서 수행할 작업을 지정합니다.
JOBOBJECT_EXTENDED_LIMIT_INFORMATION 작업 개체에 대한 기본 및 확장 제한 정보를 포함합니다.
JOBOBJECT_LIMIT_VIOLATION_INFORMATION 작업 개체에 대해 초과된 리소스 알림 제한에 대한 정보를 포함합니다. 이 구조체는 JobObjectLimitViolationInformation 정보 클래스와 함께 QueryInformationJobObject 함수와 함께 사용됩니다.
JOBOBJECT_LIMIT_VIOLATION_INFORMATION_2 작업 개체에 대해 초과된 리소스 알림 제한에 대한 확장된 정보를 포함합니다. 이 구조체는 JobObjectLimitViolationInformation2 정보 클래스와 함께 QueryInformationJobObject 함수와 함께 사용됩니다.
JOBOBJECT_NET_RATE_CONTROL_INFORMATION 작업에 대한 네트워크 트래픽을 제어하는 데 사용되는 정보를 포함합니다. 이 구조체는 JobObjectNetRateControlInformation 정보 클래스와 함께 SetInformationJobObject 및 QueryInformationJobObject 함수에서 사용됩니다.
JOBOBJECT_NOTIFICATION_LIMIT_INFORMATION 작업 개체의 알림 제한에 대한 정보를 포함합니다. 이 구조체는 JobObjectNotificationLimitInformation 정보 클래스와 함께 SetInformationJobObject 및 QueryInformationJobObject 함수에서 사용됩니다.
JOBOBJECT_NOTIFICATION_LIMIT_INFORMATION_2 작업 개체에 대한 알림 제한에 대한 확장된 정보를 포함합니다. 이 구조체는 JobObjectNotificationLimitInformation2 정보 클래스와 함께 SetInformationJobObject 및 QueryInformationJobObject 함수에서 사용됩니다.
JOBOBJECT_SECURITY_LIMIT_INFORMATION 작업 개체에 대한 보안 제한을 포함합니다.
LARGE_INTEGER LARGE_INTEGER 구조체는 부가된 64비트 정수 값을 나타냅니다. (LARGE_INTEGER 공용 구조체 (winnt.h))

LDT_ENTRY 설명자 테이블의 항목을 설명합니다. 이 구조는 x86 기반 시스템에서만 유효합니다.
Luid 어댑터의 로컬 식별자를 설명합니다. (LUID)
LUID_AND_ATTRIBUTES LUID(로컬 고유 식별자) 및 해당 특성을 나타냅니다.
MEM_ADDRESS_REQUIREMENTS 가상 메모리를 관리하는 함수에 대한 확장 매개 변수의 일부로 가장 낮고 가장 높은 기준 주소와 맞춤을 지정합니다.
MEM_EXTENDED_PARAMETER 가상 메모리를 관리하는 함수에 대한 확장 매개 변수를 나타냅니다.
MEMORY_BASIC_INFORMATION 프로세스의 가상 주소 공간에 있는 페이지 범위에 대한 정보를 포함합니다.
MESSAGE_RESOURCE_BLOCK LowId 및 HighId 멤버가 나타내는 범위의 식별자가 있는 메시지 문자열에 대한 정보를 포함합니다.
MESSAGE_RESOURCE_DATA 오류 메시지로 표시하거나 메시지 테이블 리소스의 메시지 상자에 표시할 서식이 지정된 텍스트에 대한 정보를 포함합니다.
MESSAGE_RESOURCE_ENTRY 메시지 테이블 리소스에 대한 오류 메시지 또는 메시지 상자 표시 텍스트를 포함합니다.
NUMA_NODE_RELATIONSHIP 프로세서 그룹의 NUMA 노드에 대한 정보를 나타냅니다. 이 구조체는 GetLogicalProcessorInformationEx 함수와 함께 사용됩니다.
OBJECT_TYPE_LIST 개체 형식의 계층 구조에서 개체 형식 요소를 식별합니다.

OSVERSIONINFOA 운영 체제 버전 정보를 포함합니다. (ANSI)
OSVERSIONINFOEXA 운영 체제 버전 정보를 포함합니다. 이 정보에는 주 버전 및 부 버전 번호, 빌드 번호, 플랫폼 식별자, 제품 제품군 및 시스템에 설치된 최신 서비스 팩에 대한 정보가 포함됩니다. (ANSI)
OSVERSIONINFOEXW 운영 체제 버전 정보를 포함합니다. 이 정보에는 주 버전 및 부 버전 번호, 빌드 번호, 플랫폼 식별자, 제품 제품군 및 시스템에 설치된 최신 서비스 팩에 대한 정보가 포함됩니다. (유니코드)
OSVERSIONINFOW 운영 체제 버전 정보를 포함합니다. (유니코드)
PERFORMANCE_DATA 요청한 스레드 프로파일링 및 하드웨어 카운터 데이터를 포함합니다.
POWER_ACTION_POLICY 시스템 전원 상태를 설정하는 데 사용되는 정보를 포함합니다.
PRIVILEGE_SET 권한 집합을 지정합니다.
PROCESS_DYNAMIC_EH_CONTINUATION_TARGET 동적 예외 처리 연속 대상을 포함합니다.
PROCESS_DYNAMIC_ENFORCED_ADDRESS_RANGE HSP(사용자 모드 하드웨어 적용 스택 보호)와 관련된 다양한 기능에서 사용하는 동적 적용 주소 범위를 포함합니다.
PROCESS_MITIGATION_ASLR_POLICY ASLR(주소 공간 임의화 레이아웃)에 대한 프로세스 완화 정책 설정을 포함합니다.
PROCESS_MITIGATION_BINARY_SIGNATURE_POLICY 이미지의 서명에 따라 이미지 로드에 대한 프로세스 완화 정책 설정을 포함합니다.

PROCESS_MITIGATION_CONTROL_FLOW_GUARD_POLICY CFG(Control Flow Guard)에 대한 프로세스 완화 정책 설정을 포함합니다.
PROCESS_MITIGATION_DEP_POLICY DEP(데이터 실행 방지)에 대한 프로세스 완화 정책 설정을 포함합니다.
PROCESS_MITIGATION_DYNAMIC_CODE_POLICY 동적 코드 생성 및 수정을 제한하기 위한 프로세스 완화 정책 설정을 포함합니다.
PROCESS_MITIGATION_EXTENSION_POINT_DISABLE_POLICY 레거시 확장 지점 DLL에 대한 프로세스 완화 정책 설정을 포함합니다.
PROCESS_MITIGATION_FONT_DISABLE_POLICY 비 시스템 글꼴 로드에 대한 프로세스 완화 정책 설정을 포함합니다.
PROCESS_MITIGATION_IMAGE_LOAD_POLICY 원격 디바이스에서 이미지를 로드하기 위한 프로세스 완화 정책 설정을 포함합니다.
PROCESS_MITIGATION_REDIRECTION_TRUST_POLICY ???. 대한 프로세스 완화 정책 설정을 포함합니다.
PROCESS_MITIGATION_SIDE_CHANNEL_ISOLATION_POLICY 이 데이터 구조는 사이드 채널의 완화와 관련된 프로세스 정책의 상태 제공합니다. 여기에는 투기적 실행 및 페이지 결합과 관련된 측면 채널 공격이 포함될 수 있습니다.
PROCESS_MITIGATION_STRICT_HANDLE_CHECK_POLICY 잘못된 핸들 참조에 새 동작을 적용하는 데 사용됩니다.
PROCESS_MITIGATION_SYSTEM_CALL_DISABLE_POLICY 프로세스에서 호출할 수 있는 시스템 호출에 제한을 적용하는 데 사용됩니다.
PROCESS_MITIGATION_USER_SHADOW_STACK_POLICY 사용자 모드 HSP(하드웨어 적용 스택 보호)에 대한 프로세스 완화 정책 설정을 포함합니다.
PROCESSOR_GROUP_INFO 프로세서 그룹의 프로세서 수와 선호도를 나타냅니다.

PROCESSOR_NUMBER 프로세서 그룹의 논리 프로세서를 나타냅니다.
PROCESSOR_POWER_POLICY 프로세서 성능 제어 및 C 상태에 대한 정보를 포함합니다.
PROCESSOR_POWER_POLICY_INFO 프로세서 C 상태 정책 설정에 대한 정보를 포함합니다.
PROCESSOR_RELATIONSHIP 프로세서 그룹 내의 선호도에 대한 정보를 나타냅니다. 이 구조체는 GetLogicalProcessorInformationEx 함수와 함께 사용됩니다.
QUOTA_LIMITS 사용자가 사용할 수 있는 시스템 리소스의 양을 설명합니다.
REPARSE_GUID_DATA_BUFFER 재분석 지점에 대한 정보를 포함합니다.
RUNTIME_FUNCTION 64비트 Windows의 함수 테이블에 있는 항목을 나타냅니다.
SECURITY_CAPABILITIES 앱 컨테이너의 보안 기능을 정의합니다.
SECURITY_DESCRIPTOR 개체와 연결된 보안 정보를 포함합니다.
SECURITY_QUALITY_OF_SERVICE 클라이언트 가장을 지원하는 데 사용되는 정보를 포함합니다.
SID 사용자 또는 그룹을 고유하게 식별하는 데 사용됩니다.
SID_AND_ATTRIBUTES SID(보안 식별자) 및 해당 특성을 나타냅니다.

SID_AND_ATTRIBUTES_HASH 지정된 SID(보안 식별자) 배열에 대한 해시 값을 지정합니다.
SID_IDENTIFIER_AUTHORITY SID(보안 식별자)의 최상위 기관을 나타냅니다.
SINGLE_LIST_ENTRY Singly 연결된 목록의 항목을 나타냅니다. S
SLIST_ENTRY Singly 연결된 목록의 항목을 나타냅니다. (SLIST_ENTRY)
SYSTEM_ALARM_ACE SYSTEM_ALARM_ACE 구조체는 나중에 사용하도록 예약되어 있습니다.
SYSTEM_ALARM_CALLBACK_ACE SYSTEM_ALARM_CALLBACK_ACE 구조체는 나중에 사용하도록 예약되어 있습니다.
SYSTEM_ALARM_CALLBACK_OBJECT_ACE SYSTEM_ALARM_CALLBACK_OBJECT_ACE 구조체는 나중에 사용하도록 예약되어 있습니다.
SYSTEM_ALARM_OBJECT_ACE SYSTEM_ALARM_OBJECT_ACE 구조체는 나중에 사용하도록 예약되어 있습니다.
SYSTEM_AUDIT_ACE 시스템 수준 알림을 유발하는 액세스 유형을 지정하는 SACL(시스템 액세스 제어 목록)에 대한 ACE(액세스 제어 항목)를 정의합니다.
SYSTEM_AUDIT_CALLBACK_ACE SYSTEM_AUDIT_CALLBACK_ACE 구조는 시스템 수준 알림을 유발하는 액세스 유형을 지정하는 시스템 액세스 제어 목록에 대한 액세스 제어 항목을 정의합니다.
SYSTEM_AUDIT_CALLBACK_OBJECT_ACE SYSTEM_AUDIT_CALLBACK_OBJECT_ACE 구조체는 시스템 액세스 제어 목록에 대한 액세스 제어 항목을 정의합니다.

SYSTEM_AUDIT_OBJECT_ACE SACL(시스템 액세스 제어 목록)에 대한 ACE(액세스 제어 항목)를 정의합니다.
SYSTEM_BATTERY_STATE 시스템 배터리의 현재 상태에 대한 정보를 포함합니다.
SYSTEM_CPU_SET_INFORMATION 이 구조체는 GetSystemCpuSetInformation에서 반환됩니다. 시스템에서 CPU 집합을 열거하고 현재 상태를 확인하는 데 사용됩니다.
SYSTEM_LOGICAL_PROCESSOR_INFORMATION 지정된 프로세서 집합 간의 관계를 설명합니다. 이 구조체는 GetLogicalProcessorInformation 함수와 함께 사용됩니다.
SYSTEM_LOGICAL_PROCESSOR_INFORMATION_EX 논리 프로세서 및 관련 하드웨어의 관계에 대한 정보를 포함합니다. GetLogicalProcessorInformationEx 함수는 이 구조를 사용합니다.
SYSTEM_MANDATORY_LABEL_ACE 보안 개체에 대한 필수 액세스 수준 및 정책을 지정하는 SACL(시스템 액세스 제어 목록)에 대한 ACE(액세스 제어 항목)를 정의합니다.
SYSTEM_POWER_CAPABILITIES 시스템의 전원 기능에 대한 정보를 포함합니다.
SYSTEM_POWER_LEVEL 시스템 배터리 드레이닝 정책 설정에 대한 정보를 포함합니다.
SYSTEM_POWER_POLICY 현재 시스템 전원 정책에 대한 정보를 포함합니다.
SYSTEM_RESOURCE_ATTRIBUTE_ACE 보안 개체에 대한 시스템 리소스 특성을 지정하는 SACL(시스템 액세스 제어 목록)에 대한 ACE(액세스 제어 항목)를 정의합니다.
SYSTEM_SCOPED_POLICY_ID_ACE 보안 개체의 범위가 지정된 정책 식별자를 지정하는 SACL(시스템 액세스 제어 목록)에 대한 ACE(액세스 제어 항목)를 정의합니다.

TAPE_ERASE 지울 파티션에 대해 설명합니다.
TAPE_GET_DRIVE_PARAMETERS 테이프 드라이브에 대해 설명합니다. GetTapeParameters 함수에서 사용됩니다.
TAPE_GET_MEDIA_PARAMETERS 테이프 드라이브의 테이프에 대해 설명합니다. GetTapeParametersfunction에서 사용됩니다.
TAPE_GET_POSITION 테이프의 위치를 설명합니다.
TAPE_PREPARE 테이프를 준비하는 방법을 설명합니다.
TAPE_SET_DRIVE_PARAMETERS 테이프 드라이브에 대해 설명합니다. SetTapeParametersfunction에서 사용됩니다.
TAPE_SET_MEDIA_PARAMETERS 테이프 드라이브의 테이프에 대해 설명합니다. SetTapeParametersfunction에서 사용됩니다.
TAPE_SET_POSITION 테이프를 배치하는 방법과 위치를 설명합니다.
TAPE_WRITE_MARKS 쓸 테이프 표시의 유형과 수를 설명합니다.
TOKEN_ACCESS_INFORMATION 액세스 검사 수행하는 데 필요한 토큰의 모든 정보를 지정합니다.
TOKEN_APPCONTAINER_INFORMATION 앱 컨테이너에 필요한 토큰의 모든 정보를 지정합니다.
TOKEN_AUDIT_POLICY 토큰에 대한 사용자별 감사 정책을 지정합니다.

TOKEN_CONTROL 액세스 토큰을 식별하는 정보를 포함합니다.
TOKEN_DEFAULT_DACL DACL(임의 액세스 제어 목록)을 지정합니다.
TOKEN_DEVICE_CLAIMS 토큰에 대한 디바이스 클레임을 정의합니다.
TOKEN_ELEVATION 토큰에 상승된 권한이 있는지 여부를 나타냅니다.
TOKEN_GROUPS 액세스 토큰의 SID(그룹 보안 식별자)에 대한 정보를 포함합니다.
TOKEN_GROUPS_AND_PRIVILEGES 액세스 토큰의 SID(그룹 보안 식별자) 및 권한에 대한 정보를 포함합니다.
TOKEN_LINKED_TOKEN 토큰에 대한 핸들을 포함합니다. 이 토큰은 GetTokenInformation 함수에서 쿼리하거나 SetTokenInformation 함수에서 설정하는 토큰에 연결됩니다.
TOKEN_MANDATORY_LABEL 토큰에 대한 필수 무결성 수준을 지정합니다.
TOKEN_MANDATORY_POLICY 토큰에 대한 필수 무결성 정책을 지정합니다.
TOKEN_ORIGIN 로그온 세션의 원본에 대한 정보를 포함합니다.
TOKEN_OWNER 새로 만든 개체에 적용할 기본 SID(소유자 보안 식별자)를 포함합니다.
TOKEN_PRIMARY_GROUP 액세스 토큰에 대한 SID(그룹 보안 식별자)를 지정합니다.

TOKEN_PRIVILEGES 액세스 토큰에 대한 권한 집합에 대한 정보를 포함합니다.
TOKEN_SOURCE 액세스 토큰의 원본을 식별합니다.
TOKEN_STATISTICS 액세스 토큰에 대한 정보를 포함합니다.
TOKEN_USER 액세스 토큰과 연결된 사용자를 식별합니다.
TOKEN_USER_CLAIMS 토큰에 대한 사용자 클레임을 정의합니다.
ULARGE_INTEGER ULARGE_INTEGER 구조체는 부호 없는 64비트 정수 값을 나타냅니다. (ULARGE_INTEGER 공용 구조체(winnt.h))
UMS_CREATE_THREAD_ATTRIBUTES UMS(사용자 모드 예약) 작업자 스레드에 대한 특성을 지정합니다.
WOW64_CONTEXT WOW64의 컨텍스트 프레임을 나타냅니다.
WOW64_FLOATING_SAVE_AREA WOW64의 80387 저장 영역을 나타냅니다.
WOW64_LDT_ENTRY 64비트 시스템의 32비트 스레드에 대한 설명자 테이블의 항목을 설명합니다. 이 구조는 64비트 시스템에서만 유효합니다.

열거형

ACL_INFORMATION_CLASS ACL(액세스 제어 목록)에 할당되거나 검색되는 정보의 유형을 지정하는 값을 포함합니다.
ACTCTX_COMPATIBILITY_ELEMENT_TYPE ACTCTX_COMPATIBILITY_ELEMENT_TYPE 열거형은 애플리케이션 매니페스트의 호환성 요소를 설명합니다.
ACTCTX_REQUESTED_RUN_LEVEL ACTCTX_REQUESTED_RUN_LEVEL 열거형은 활성화 컨텍스트의 요청된 실행 수준을 설명합니다.
AUDIT_EVENT_TYPE 감사되는 개체의 형식을 나타내는 값을 정의합니다. AccessCheckByTypeAndAuditAlarm 및 AccessCheckByTypeResultListAndAuditAlarm 함수는 이러한 값을 사용합니다.
COMPARTMENT_ID COMPARTMENT_ID 열거형은 네트워크 라우팅 구획 식별자를 나타냅니다.
FIRMWARE_TYPE 펌웨어 유형을 지정합니다.
HARDWARE_COUNTER_TYPE 프로파일되는 하드웨어 카운터의 유형을 정의합니다.
HEAP_INFORMATION_CLASS 설정하거나 검색할 힙 정보의 클래스를 지정합니다.
JOB_OBJECT_NET_RATE_CONTROL_FLAGS 네트워크 속도 제어에 대한 일정 정책 유형을 지정합니다.
LOGICAL_PROCESSOR_RELATIONSHIP 해당 SYSTEM_LOGICAL_PROCESSOR_INFORMATION 또는 SYSTEM_LOGICAL_PROCESSOR_INFORMATION_EX 구조에서 식별된 프로세서 집합 간의 관계를 나타냅니다.
MANDATORY_LEVEL 가능한 보안 수준을 나열합니다.

MEM_EXTENDED_PARAMETER_TYPE 주소 공간에 파일 매핑에 사용되는 확장 매개 변수의 값을 정의합니다.
POWER_ACTION 시스템 전원 작업 유형을 지정하는 데 사용되는 값을 정의합니다.
POWER_PLATFORM_ROLE OEM의 기본 전원 관리 프로필을 나타냅니다.
PROCESS_MITIGATION_POLICY 다양한 프로세스 완화 정책을 나타냅니다.
PROCESSOR_CACHE_TYPE 해당 CACHE_DESCRIPTOR 구조에서 식별되는 프로세서 캐시의 유형을 나타냅니다.
SECURITY_IMPERSONATION_LEVEL 보안 가장 수준을 지정하는 값을 포함합니다. 보안 가장 수준은 서버 프로세스가 클라이언트 프로세스를 대신하여 작업을 수행할 수 있는 정도를 제어합니다.
SID_NAME_USE SID(보안 식별자) 형식을 지정하는 값을 포함합니다.
SYSTEM_POWER_CONDITION GUID_ACDC_POWER_SOURCE 전원 이벤트에서 현재 전원을 나타내는 데 사용됩니다.
SYSTEM_POWER_STATE 시스템 전원 상태를 지정하는 데 사용되는 값을 정의합니다.
TOKEN_ELEVATION_TYPE GetTokenInformation 함수에서 쿼리하거나 SetTokenInformation 함수에서 설정하는 토큰의 상승 유형을 나타냅니다.
TOKEN_INFORMATION_CLASS 액세스 토큰에 할당되거나 검색되는 정보의 유형을 지정하는 값을 포함합니다.
TOKEN_TYPE 기본 토큰과 가장 토큰을 구분하는 값을 포함합니다.

TRANSACTION_OUTCOME KTM이 트랜잭션에 할당할 수 있는 결과(결과)를 정의합니다.
WELL_KNOWN_SID_TYPE 일반적으로 사용되는 SID(보안 식별자) 목록입니다. 프로그램에서 이러한 값을 CreateWellKnownSid 함수에 전달하여 이 목록에서 SID를 만들 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winperf.h 헤더

아티클2023. 08. 27.

이 헤더는 성능 카운터에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 성능 카운터

winperf.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

콜백 함수

PM_CLOSE_PROC 성능 DLL에 필요한 정리를 수행합니다.
PM_COLLECT_PROC 성능 데이터를 수집하고 소비자에게 반환합니다.

구조체

PERF_COUNTER_BLOCK 개체의 카운터에 대한 원시 성능 카운터 데이터가 포함된 메모리 블록을 설명합니다.
PERF_COUNTER_DEFINITION 성능 카운터에 대해 설명합니다.
PERF_DATA_BLOCK 쿼리한 성능 데이터 블록(예: 공급자가 반환하는 성능 개체 수 및 성능 값을 계산할 때 사용하는 시간 기반 값)에 대해 설명합니다.
PERF_INSTANCE_DEFINITION 성능 개체의 instance 설명합니다.
PERF_OBJECT_TYPE 개체별 성능 정보(예: 개체 인스턴스 수 및 개체가 정의하는 카운터 수)를 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winreg.h 헤더

아티클2023. 03. 08.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보
- Hyper-V WMI 공급자
- 보안 및 ID
- 시스템 서비스

winreg.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AbortSystemShutdownA 시작된 시스템 종료를 중지합니다. (ANSI)
AbortSystemShutdownW 시작된 시스템 종료를 중지합니다. (유니코드)
InitiateShutdownA 지정된 컴퓨터의 종료 및 다시 시작을 시작하고 다시 시작하기 위해 등록된 모든 애플리케이션을 다시 시작합니다. (ANSI)
InitiateShutdownW 지정된 컴퓨터의 종료 및 다시 시작을 시작하고 다시 시작하기 위해 등록된 모든 애플리케이션을 다시 시작합니다. (유니코드)
InitiateSystemShutdownA 지정된 컴퓨터의 종료 및 선택적 다시 시작을 시작합니다. (ANSI)
InitiateSystemShutdownExA 지정된 컴퓨터의 종료 및 선택적 다시 시작을 시작하고 필요에 따라 종료 이유를 기록합니다. (ANSI)

InitiateSystemShutdownExW 지정된 컴퓨터의 종료 및 선택적 다시 시작을 시작하고 필요에 따라 종료 이유를 기록합니다. (유니코드)
InitiateSystemShutdownW 지정된 컴퓨터의 종료 및 선택적 다시 시작을 시작합니다. (유니코드)
RegCloseKey 지정된 레지스트리 키에 대한 핸들을 닫습니다.
RegConnectRegistryA 다른 컴퓨터에서 미리 정의된 레지스트리 키에 대한 연결을 설정합니다. (ANSI)
RegConnectRegistryW 다른 컴퓨터에서 미리 정의된 레지스트리 키에 대한 연결을 설정합니다. (유니코드)
RegCopyTreeA 지정된 레지스트리 키를 해당 값 및 하위 키와 함께 지정된 대상 키에 복사합니다. (ANSI)
RegCopyTreeW 지정된 레지스트리 키를 해당 값 및 하위 키와 함께 지정된 대상 키에 복사합니다. (유니코드)
RegCreateKeyA 지정된 레지스트리 키를 만듭니다. 레지스트리에 키가 이미 있는 경우 함수가 키를 엽니다. (ANSI)
RegCreateKeyExA 지정된 레지스트리 키를 만듭니다. 키가 이미 있는 경우 함수가 키를 엽니다. 키 이름은 대/소문자를 구분하지 않습니다. (ANSI)
RegCreateKeyExW 지정된 레지스트리 키를 만듭니다. 키가 이미 있는 경우 함수가 키를 엽니다. 키 이름은 대/소문자를 구분하지 않습니다. (유니코드)
RegCreateKeyTransactedA 지정된 레지스트리 키를 만들어 트랜잭션과 연결합니다. (ANSI)

RegCreateKeyTransactedW 지정된 레지스트리 키를 만들어 트랜잭션과 연결합니다. (유니코드)
RegCreateKeyW 지정된 레지스트리 키를 만듭니다. 레지스트리에 키가 이미 있는 경우 함수가 키를 엽니다. (유니코드)
RegDeleteKeyA 하위 키와 해당 값을 삭제합니다. (ANSI)
RegDeleteKeyExA 레지스트리의 지정된 플랫폼별 보기에서 하위 키와 해당 값을 삭제합니다. (ANSI)
RegDeleteKeyExW 레지스트리의 지정된 플랫폼별 보기에서 하위 키와 해당 값을 삭제합니다. (유니코드)
RegDeleteKeyTransactedA 레지스트리의 지정된 플랫폼별 보기에서 하위 키와 해당 값을 트랜잭션 작업으로 삭제합니다. (ANSI)
RegDeleteKeyTransactedW 레지스트리의 지정된 플랫폼별 보기에서 하위 키와 해당 값을 트랜잭션 작업으로 삭제합니다. (유니코드)
RegDeleteKeyValueA 지정된 레지스트리 키 및 하위 키에서 지정된 값을 제거합니다. (ANSI)
RegDeleteKeyValueW 지정된 레지스트리 키 및 하위 키에서 지정된 값을 제거합니다. (유니코드)
RegDeleteKeyW 하위 키와 해당 값을 삭제합니다. (유니코드)
RegDeleteTreeA 지정된 키의 하위 키와 값을 재귀적으로 삭제합니다. (ANSI)

RegDeleteTreeW 지정된 키의 하위 키와 값을 재귀적으로 삭제합니다. (유니코드)
RegDeleteValueA 지정된 레지스트리 키에서 명명된 값을 제거합니다. (ANSI)
RegDeleteValueW 지정된 레지스트리 키에서 명명된 값을 제거합니다. (유니코드)
RegDisablePredefinedCache 현재 프로세스에 대한 HKEY_CURRENT_USER 대해 미리 정의된 레지스트리 핸들의 캐싱을 사용하지 않도록 설정합니다.
RegDisablePredefinedCacheEx 현재 프로세스에 대해 미리 정의된 모든 레지스트리 핸들에 대한 핸들 캐싱을 사용하지 않도록 설정합니다.
RegDisableReflectionKey 지정된 키에 대한 레지스트리 리플렉션을 사용하지 않도록 설정합니다. 키에 대한 리플렉션을 사용하지 않도록 설정해도 하위 키의 리플렉션에는 영향을 주지 않습니다.
RegEnableReflectionKey 지정된 비활성화된 키에 대한 레지스트리 리플렉션을 복원합니다. 키에 대한 리플렉션을 복원해도 하위 키의 리플렉션에는 영향을 주지 않습니다.
RegEnumKeyA 지정된 열린 레지스트리 키의 하위 키를 열거합니다. (RegEnumKeyA)
RegEnumKeyExA 지정된 열린 레지스트리 키의 하위 키를 열거합니다. 함수는 호출할 때마다 하나의 하위 키에 대한 정보를 검색합니다. (ANSI)
RegEnumKeyExW 지정된 열린 레지스트리 키의 하위 키를 열거합니다. 함수는 호출할 때마다 하나의 하위 키에 대한 정보를 검색합니다. (유니코드)
RegEnumKeyW 지정된 열린 레지스트리 키의 하위 키를 열거합니다. (RegEnumKeyW)

RegEnumValueA 지정된 열린 레지스트리 키의 값을 열거합니다. 함수는 호출될 때마다 키에 대해 인덱싱된 값 이름과 데이터 블록을 하나씩 복사합니다. (ANSI)
RegEnumValueW 지정된 열린 레지스트리 키의 값을 열거합니다. 함수는 호출될 때마다 키에 대해 인덱싱된 값 이름과 데이터 블록을 하나씩 복사합니다. (유니코드)
RegFlushKey 지정된 열린 레지스트리 키의 특성을 모두 레지스트리에 기록합니다.
RegGetKeySecurity 지정된 열린 레지스트리 키를 보호하는 보안 설명자의 복사본을 검색합니다.
RegGetValueA 지정된 레지스트리 값의 형식 및 데이터를 검색합니다. (ANSI)
RegGetValueW 지정된 레지스트리 값의 형식 및 데이터를 검색합니다. (유니코드)
RegLoadAppKeyA 지정된 레지스트리 하이브를 애플리케이션 하이브로 로드합니다. (ANSI)
RegLoadAppKeyW 지정된 레지스트리 하이브를 애플리케이션 하이브로 로드합니다. (유니코드)
RegLoadKeyA HKEY_USERS 또는 HKEY_LOCAL_MACHINE 아래에 하위 키를 만들고 지정된 레지스트리 하이브의 데이터를 해당 하위 키로 로드합니다. (ANSI)
RegLoadKeyW HKEY_USERS 또는 HKEY_LOCAL_MACHINE 아래에 하위 키를 만들고 지정된 레지스트리 하이브의 데이터를 해당 하위 키로 로드합니다. (유니코드)
RegLoadMUIStringA 지정된 키와 하위 키에서 지정된 문자열을 로드합니다. (ANSI)

RegLoadMUIStringW 지정된 키와 하위 키에서 지정된 문자열을 로드합니다. (유니코드)
RegNotifyChangeKeyValue 지정된 레지스트리 키의 특성 또는 내용에 대한 변경 내용에 대해 호출자에게 알합니다.
RegOpenCurrentUser 현재 스레드가 가장하는 사용자의 HKEY_CURRENT_USER 키에 대한 핸들을 검색합니다.
RegOpenKeyA 지정된 레지스트리 키를 엽니다. (ANSI)
RegOpenKeyExA 지정된 레지스트리 키를 엽니다. 키 이름은 대/소문자를 구분하지 않습니다. (ANSI)
RegOpenKeyExW 지정된 레지스트리 키를 엽니다. 키 이름은 대/소문자를 구분하지 않습니다. (유니코드)
RegOpenKeyTransactedA 지정된 레지스트리 키를 열고 트랜잭션과 연결합니다. (ANSI)
RegOpenKeyTransactedW 지정된 레지스트리 키를 열고 트랜잭션과 연결합니다. (유니코드)
RegOpenKeyW 지정된 레지스트리 키를 엽니다. (유니코드)
RegOpenUserClassesRoot 지정된 사용자의 HKEY_CLASSES_ROOT 키에 대한 핸들을 검색합니다. 사용자는 액세스 토큰으로 식별됩니다.
RegOverridePredefKey 미리 정의된 레지스트리 키를 지정된 레지스트리 키에 매핑합니다.
RegQueryInfoKeyA 지정된 레지스트리 키에 대한 정보를 검색합니다. (ANSI)

RegQueryInfoKeyW 지정된 레지스트리 키에 대한 정보를 검색합니다. (유니코드)
RegQueryMultipleValuesA 열린 레지스트리 키와 연결된 값 이름 목록에 대한 형식 및 데이터를 검색합니다. (ANSI)
RegQueryMultipleValuesW 열린 레지스트리 키와 연결된 값 이름 목록에 대한 형식 및 데이터를 검색합니다. (유니코드)
RegQueryReflectionKey 지정된 키에 대해 리플렉션을 사용하지 않도록 설정했는지 여부를 결정합니다.
RegQueryValueA 지정된 레지스트리 키의 기본값 또는 명명되지 않은 값과 연결된 데이터를 검색합니다. 데이터는 null로 끝나는 문자열이어야 합니다. (ANSI)
RegQueryValueExA 열린 레지스트리 키와 연결된 지정된 값 이름의 형식 및 데이터를 검색합니다. (ANSI)
RegQueryValueExW 열린 레지스트리 키와 연결된 지정된 값 이름의 형식 및 데이터를 검색합니다. (유니코드)
RegQueryValueW 지정된 레지스트리 키의 기본값 또는 명명되지 않은 값과 연결된 데이터를 검색합니다. 데이터는 null로 끝나는 문자열이어야 합니다. (유니코드)
RegRenameKey 지정된 레지스트리 키의 이름을 변경합니다. (RegRenameKey)
RegReplaceKeyA 레지스트리 키 및 모든 하위 키를 지원하는 파일을 다른 파일로 바꿉니다. 따라서 시스템이 다음에 시작될 때 키와 하위 키는 새 파일에 저장된 값을 갖게 됩니다. (ANSI)
RegReplaceKeyW 레지스트리 키 및 모든 하위 키를 지원하는 파일을 다른 파일로 바꿉니다. 따라서 시스템이 다음에 시작될 때 키와 하위 키는 새 파일에 저장된 값을 갖게 됩니다. (유니코드)

RegRestoreKeyA 지정된 파일에서 레지스트리 정보를 읽고 지정된 키 위에 복사합니다. 이 레지스트리 정보는 키와 여러 수준의 하위 키 형식일 수 있습니다. (ANSI)
RegRestoreKeyW 지정된 파일에서 레지스트리 정보를 읽고 지정된 키 위에 복사합니다. 이 레지스트리 정보는 키와 여러 수준의 하위 키 형식일 수 있습니다. (유니코드)
RegSaveKeyA 지정된 키와 모든 하위 키와 값을 표준 형식으로 새 파일에 저장합니다. (ANSI)
RegSaveKeyExA 지정된 키와 모든 하위 키 및 값을 레지스트리 파일에 지정된 형식으로 저장합니다. (ANSI)
RegSaveKeyExW 지정된 키와 모든 하위 키 및 값을 레지스트리 파일에 지정된 형식으로 저장합니다. (유니코드)
RegSaveKeyW 지정된 키와 모든 하위 키와 값을 표준 형식으로 새 파일에 저장합니다. (유니코드)
RegSetKeySecurity 열린 레지스트리 키의 보안을 설정합니다.
RegSetKeyValueA 지정된 레지스트리 키 및 하위 키의 지정된 값에 대한 데이터를 설정합니다. (ANSI)
RegSetKeyValueW 지정된 레지스트리 키 및 하위 키의 지정된 값에 대한 데이터를 설정합니다. (유니코드)
RegSetValueA 지정된 레지스트리 키의 기본값 또는 명명되지 않은 값에 대한 데이터를 설정합니다. 데이터는 텍스트 문자열이어야 합니다. (ANSI)
RegSetValueExA 레지스트리 키 아래에 지정된 값의 데이터 및 형식을 설정합니다. (ANSI)

RegSetValueExW 레지스트리 키 아래에 지정된 값의 데이터 및 형식을 설정합니다. (유니코드)
RegSetValueW 지정된 레지스트리 키의 기본값 또는 명명되지 않은 값에 대한 데이터를 설정합니다. 데이터는 텍스트 문자열이어야 합니다. (유니코드)
RegUnLoadKeyA 레지스트리에서 지정된 레지스트리 키와 해당 하위 키를 언로드합니다. (ANSI)
RegUnLoadKeyW 레지스트리에서 지정된 레지스트리 키와 해당 하위 키를 언로드합니다. (유니코드)

구조체

VALENTA 레지스트리 값에 대한 정보를 포함합니다. RegQueryMultipleValues 함수는 이 구조를 사용합니다. (ANSI)
VALENTW 레지스트리 값에 대한 정보를 포함합니다. RegQueryMultipleValues 함수는 이 구조를 사용합니다. (유니코드)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winsafer.h 헤더

아티클2023. 08. 27.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

winsafer.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

SaferCloseLevel SaferIdentifyLevel 함수 또는 SaferCreateLevel 함수를 사용하여 연 SAFER_LEVEL_HANDLE 닫습니다.
SaferComputeTokenFromLevel SAFER_LEVEL_HANDLE 지정된 제한을 사용하여 토큰을 제한합니다.
SaferCreateLevel SAFER_LEVEL_HANDLE 엽니다.
SaferGetLevelInformation 정책 수준에 대한 정보를 검색합니다.
SaferGetPolicyInformation 정책에 대한 정보를 가져옵니다.
SaferIdentifyLevel 수준에 대한 정보를 검색합니다.
SaferiIsExecutableFileType 지정된 파일이 실행 파일인지 여부를 확인합니다.
SaferRecordEventLogEntry 이벤트 로그에 메시지를 저장합니다.
SaferSetLevelInformation

정책 수준에 대한 정보를 설정합니다.
SaferSetPolicyInformation 전역 정책 컨트롤을 설정합니다.

구조체

SAFER_CODE_PROPERTIES_V1 코드 이미지에서 확인할 코드 이미지 정보 및 조건을 포함합니다. (SAFER_CODE_PROPERTIES_V1)
SAFER_CODE_PROPERTIES_V2 코드 이미지에서 확인할 코드 이미지 정보 및 조건을 포함합니다. S
SAFER_HASH_IDENTIFICATION 해시 식별 규칙을 나타냅니다.
SAFER_IDENTIFICATION_HEADER SAFER_IDENTIFICATION_HEADER 구조체는 SAFER_PATHNAME_IDENTIFICATION, SAFER_HASH_IDENTIFICATION 및 SAFER_URLZONE_IDENTIFICATION 구조체의 헤더로 사용됩니다.
SAFER_PATHNAME_IDENTIFICATION 경로 식별 규칙을 나타냅니다.
SAFER_URLZONE_IDENTIFICATION URL 영역 식별 규칙을 나타냅니다.

열거형

SAFER_IDENTIFICATION_TYPES SAFER_IDENTIFICATION_HEADER 구조체에서 식별할 수 있는 식별 규칙 구조의 가능한 유형을 정의합니다.

SAFER_OBJECT_INFO_CLASS Safer 개체에 대해 요청된 정보의 형식을 정의합니다.
SAFER_POLICY_INFO_CLASS 정책을 쿼리할 수 있는 방법을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winsatcominterfacei.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows 시스템 평가 도구에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 시스템 평가 도구

winsatcominterfacei.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IInitiateWinSATAssessment 평가를 시작합니다.
IProvideWinSATAssessmentInfo 평가의 하위 구성 요소에 대한 요약 정보(예: 점수)를 가져옵니다.
IProvideWinSATResultsInfo 평가 결과(예: 기본 점수 및 평가가 실행된 날짜)에 대한 정보를 가져옵니다.
IProvideWinSATVisuals WinSAT 평가를 그래픽으로 나타내기 위해 사용자 인터페이스에서 사용할 수 있는 요소를 검색합니다.
IQueryAllWinSATAssessments 모든 공식 WinSAT 평가에 대한 세부 정보를 검색합니다.
IQueryRecentWinSATAssessment 가장 최근의 공식 WinSAT 평가 결과에 대한 세부 정보를 검색합니다.
IWinSATInitiateEvents 평가가 완료되거나 진행 중일 때 알림을 받도록 이 인터페이스를 구현합니다.

열거형

WINSAT_ASSESSMENT_STATE 평가의 가능한 상태를 정의합니다.
WINSAT_ASSESSMENT_TYPE 평가의 가능한 하위 구성 요소를 정의합니다.
WINSAT_BITMAP_SIZE WinSAT 점수를 나타내는 데 사용할 비트맵의 크기를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winscard.h 헤더

아티클2023. 08. 27.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

winscard.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

GetOpenCardNameA GetOpenCardName 함수는 스마트 카드 "카드 선택" 대화 상자를 표시합니다. (ANSI)
GetOpenCardNameW GetOpenCardName 함수는 스마트 카드 "카드 선택" 대화 상자를 표시합니다. (유니코드)
SCardAccessStartedEvent 이벤트가 스마트 카드 리소스 관리자가 시작되었음을 알리는 경우 이벤트 핸들을 반환합니다.
SCardAddReaderToGroupA 판독기 그룹에 판독기를 추가합니다. (ANSI)
SCardAddReaderToGroupW 판독기 그룹에 판독기를 추가합니다. (유니코드)
SCardAudit Windows 애플리케이션 로그 Microsoft-Windows-SmartCard-Audit/Authentication에 이벤트 메시지를 씁니다.
SCardBeginTransaction 트랜잭션을 시작합니다.
SCardCancel 특정 리소스 관리자 컨텍스트 내에서 모든 미해결 작업을 종료합니다.
SCardConnectA

호출 애플리케이션과 특정 판독기가 포함하는 스마트 카드 간에 연결(특정 리소스 관리자 컨텍스트 사용)을 설정합니다. 지정된 판독기에서 카드 없으면 오류가 반환됩니다. (ANSI)
SCardConnectW 호출 애플리케이션과 특정 판독기가 포함하는 스마트 카드 간에 연결(특정 리소스 관리자 컨텍스트 사용)을 설정합니다. 지정된 판독기에서 카드 없으면 오류가 반환됩니다. (유니코드)
SCardControl 판독기를 직접 제어할 수 있습니다. SCardConnect를 성공적으로 호출한 후 SCardDisconnect를 성공적으로 호출하기 전에 언제든지 호출할 수 있습니다.
SCardDisconnect 호출 애플리케이션과 대상 판독기에서 스마트 카드 간에 이전에 연 연결을 종료합니다.
SCardEndTransaction 이전에 선언된 트랜잭션을 완료하여 다른 애플리케이션이 카드 상호 작용을 다시 시작할 수 있도록 합니다.
SCardEstablishContext 데이터베이스 작업이 수행되는 리소스 관리자 컨텍스트(scope)를 설정합니다.
SCardForgetCardTypeA 스마트 카드 하위 시스템의 도입된 스마트 카드 제거합니다. (ANSI)
SCardForgetCardTypeW 스마트 카드 하위 시스템의 도입된 스마트 카드 제거합니다. (유니코드)
SCardForgetReaderA 스마트 카드 하위 시스템에 의해 이전에 도입된 판독기를 제어에서 제거합니다. 추가되었을 수 있는 판독기 그룹을 포함하여 스마트 카드 데이터베이스에서 제거됩니다. (ANSI)
SCardForgetReaderGroupA 스마트 카드 하위 시스템의 이전에 도입된 스마트 카드 판독기 그룹을 제거합니다. 이 함수는 그룹의 모든 판독기를 자동으로 지우지만 데이터베이스에 개별 판독기의 존재에는 영향을 주지 않습니다. (ANSI)
SCardForgetReaderGroupW 스마트 카드 하위 시스템의 이전에 도입된 스마트 카드 판독기 그룹을 제거합니다. 이 함수는 그룹

의 모든 판독기를 자동으로 지우지만 데이터베이스에 개별 판독기의 존재에는 영향을 주지 않습니다. (유니코드)
SCardForgetReaderW 스마트 카드 하위 시스템에 의해 이전에 도입된 판독기를 제어에서 제거합니다. 추가되었을 수 있는 판독기 그룹을 포함하여 스마트 카드 데이터베이스에서 제거됩니다. (유니코드)
SCardFreeMemory SCARD_AUTOALLOCATE 길이 지정자를 사용하여 리소스 관리자에서 반환된 메모리를 해제합니다.
SCardGetAttrib 지정된 핸들에 대한 현재 판독기 특성을 검색합니다. 판독기, 드라이버 또는 카드 상태에 영향을 주지 않습니다.
SCardGetCardTypeProviderNameA 지정된 카드 이름 및 공급자 형식에 대한 공급자가 포함된 모듈(동적 링크 라이브러리)의 이름을 반환합니다. (ANSI)
SCardGetCardTypeProviderNameW 지정된 카드 이름 및 공급자 형식에 대한 공급자가 포함된 모듈(동적 링크 라이브러리)의 이름을 반환합니다. (유니코드)
SCardGetDeviceTypeIdA 지정된 판독기 이름에 대한 카드 판독기의 디바이스 유형 식별자를 가져옵니다. 이 함수는 판독기의 상태에 영향을 주지 않습니다. (ANSI)
SCardGetDeviceTypeIdW 지정된 판독기 이름에 대한 카드 판독기의 디바이스 유형 식별자를 가져옵니다. 이 함수는 판독기의 상태에 영향을 주지 않습니다. (유니코드)
SCardGetProviderIdA 지정된 카드 대한 기본 서비스 공급자의 식별자(GUID)를 반환합니다. (ANSI)
SCardGetProviderIdW 지정된 카드 대한 기본 서비스 공급자의 식별자(GUID)를 반환합니다. (유니코드)
SCardGetReaderDeviceInstanceIdA 지정된 판독기 이름에 대한 카드 판독기의 디바이스 instance 식별자를 가져옵니다. 이 함수는 판독기의 상태에 영향을 주지 않습니다. (ANSI)

SCardGetReaderDeviceInstanceIdW 지정된 판독기 이름에 대한 카드 판독기의 디바이스 instance 식별자를 가져옵니다. 이 함수는 판독기의 상태에 영향을 주지 않습니다. (유니코드)
SCardGetReaderIconA 지정된 판독기 이름에 대한 스마트 카드 판독기의 아이콘을 가져옵니다. (ANSI)
SCardGetReaderIconW 지정된 판독기 이름에 대한 스마트 카드 판독기의 아이콘을 가져옵니다. (유니코드)
SCardGetStatusChangeA 특정 판독기 집합에 있는 카드의 현재 가용성이 변경될 때까지 실행을 차단합니다. (ANSI)
SCardGetStatusChangeW 특정 판독기 집합에 있는 카드의 현재 가용성이 변경될 때까지 실행을 차단합니다. (유니코드)
SCardGetTransmitCount 지정된 카드 판독기가 삽입된 이후 완료된 전송 작업의 수를 검색합니다.
SCardIntroduceCardTypeA 스마트 카드 데이터베이스에 추가하여 스마트 카드 하위 시스템(활성 사용자용)에 스마트 카드 도입합니다. (ANSI)
SCardIntroduceCardTypeW 스마트 카드 데이터베이스에 추가하여 스마트 카드 하위 시스템(활성 사용자용)에 스마트 카드 도입합니다. (유니코드)
SCardIntroduceReaderA 기존 스마트 카드 읽기 프로그램의 새 이름을 소개합니다. (ANSI)
SCardIntroduceReaderGroupA 스마트 카드 하위 시스템에 판독기 그룹을 소개합니다. 그러나 스마트 카드 데이터베이스에 판독기를 추가할 때 그룹을 지정할 때까지 판독기 그룹이 만들어지지 않습니다. (ANSI)
SCardIntroduceReaderGroupW 스마트 카드 하위 시스템에 판독기 그룹을 소개합니다. 그러나 스마트 카드 데이터베이스에 판독기를 추가할 때 그룹을 지정할 때까지 판독기 그룹이 만들어지지 않습니다. (유니코드)

SCardIntroduceReaderW 기존 스마트 카드 읽기 프로그램의 새 이름을 소개합니다. (유니코드)
SCardIsValidContext 스마트 카드 컨텍스트 핸들이 유효한지 여부를 결정합니다.
SCardListCardsA 스마트 카드 데이터베이스를 검색하고 사용자가 이전에 시스템에 도입한 명명된 카드 목록을 제공합니다. (ANSI)
SCardListCardsW 스마트 카드 데이터베이스를 검색하고 사용자가 이전에 시스템에 도입한 명명된 카드 목록을 제공합니다. (유니코드)
SCardListInterfacesA 지정된 카드 제공하는 인터페이스 목록을 제공합니다. (ANSI)
SCardListInterfacesW 지정된 카드 제공하는 인터페이스 목록을 제공합니다. (유니코드)
SCardListReaderGroupsA 이전에 시스템에 도입된 판독기 그룹 목록을 제공합니다. (ANSI)
SCardListReaderGroupsW 이전에 시스템에 도입된 판독기 그룹 목록을 제공합니다. (유니코드)
SCardListReadersA 명명된 판독기 그룹 집합 내에서 판독기 목록을 제공하여 중복을 제거합니다. (ANSI)
SCardListReadersW 명명된 판독기 그룹 집합 내에서 판독기 목록을 제공하여 중복을 제거합니다. (유니코드)
SCardListReadersWithDeviceInstanceIdA 디바이스 instance 식별자를 제공한 판독기 목록을 가져옵니다. 이 함수는 판독기의 상태에 영향을 주지 않습니다. (ANSI)
SCardListReadersWithDeviceInstanceIdW

디바이스 instance 식별자를 제공한 판독기 목록을 가져옵니다. 이 함수는 판독기의 상태에 영향을 주지 않습니다. (유니코드)
SCardLocateCardsA rgReaderStates 매개 변수에 나열된 판독기에서 mszCards에 지정된 카드 이름 중 하나와 일치하는 ATR 문자열이 있는 카드 검색하고 결과와 함께 즉시 반환합니다. (ANSI)
SCardLocateCardsByATRA rgReaderStates 매개 변수에 나열된 판독기에서 rgAtrMasks 매개 변수로 지정된 SCARD_ATRMASK 구조 중 하나에 포함된 카드 이름 중 하나와 일치하는 이름의 카드 검색합니다. (ANSI)
SCardLocateCardsByATRW rgReaderStates 매개 변수에 나열된 판독기에서 rgAtrMasks 매개 변수로 지정된 SCARD_ATRMASK 구조 중 하나에 포함된 카드 이름 중 하나와 일치하는 이름의 카드 검색합니다. (유니코드)
SCardLocateCardsW rgReaderStates 매개 변수에 나열된 판독기에서 mszCards에 지정된 카드 이름 중 하나와 일치하는 ATR 문자열이 있는 카드 검색하고 결과와 함께 즉시 반환합니다. (유니코드)
SCardReadCacheA 스마트 카드 Resource Manager 유지 관리하는 전역 캐시에서 이름-값 쌍의 값 부분을 검색합니다. (ANSI)
SCardReadCacheW 스마트 카드 Resource Manager 유지 관리하는 전역 캐시에서 이름-값 쌍의 값 부분을 검색합니다. (유니코드)
SCardReconnect 호출하는 애플리케이션과 스마트 카드 간의 기존 연결을 다시 설정합니다.
SCardReleaseContext 설정된 리소스 관리자 컨텍스트를 닫고 SCARDHANDLE 개체 및 SCARD_AUTOALLOCATE 길이 지정자를 사용하여 할당된 메모리를 포함하여 해당 컨텍스트에서 할당된 모든 리소스를 해제합니다.
SCardReleaseStartedEvent SCardAccessStartedEvent 함수에 대한 이전 호출에서 얻은 핸들에 대한 참조 수를 감소합니다.
SCardRemoveReaderFromGroupA 기존 판독기 그룹에서 판독기 제거 이 함수는 판독기에서 영향을 주지 않습니다. (ANSI)

SCardRemoveReaderFromGroupW 기존 판독기 그룹에서 판독기 제거 이 함수는 판독기에서 영향을 주지 않습니다. (유니코드)
SCardSetAttrib 지정된 핸들에 대해 지정된 판독기 특성을 설정합니다.
SCardSetCardTypeProviderNameA 지정된 카드 이름 및 공급자 형식에 대한 공급자를 포함하는 모듈(동적 링크 라이브러리)의 이름을 지정합니다. (ANSI)
SCardSetCardTypeProviderNameW 지정된 카드 이름 및 공급자 형식에 대한 공급자를 포함하는 모듈(동적 링크 라이브러리)의 이름을 지정합니다. (유니코드)
SCardStatusA 판독기에서 스마트 카드 현재 상태 제공합니다. (ANSI)
SCardStatusW 판독기에서 스마트 카드 현재 상태 제공합니다. (유니코드)
SCardTransmit 스마트 카드 서비스 요청을 보내고 카드 데이터를 다시 수신해야 합니다.
SCardUIDlgSelectCardA 스마트 카드 카드 선택 대화 상자를 표시합니다. (ANSI)
SCardUIDlgSelectCardW 스마트 카드 카드 선택 대화 상자를 표시합니다. (유니코드)
SCardWriteCacheA 스마트 카드 스마트 카드 Resource Manager 유지 관리하는 전역 캐시에 이름-값 쌍을 씁니다. (ANSI)
SCardWriteCacheW 스마트 카드 스마트 카드 Resource Manager 유지 관리하는 전역 캐시에 이름-값 쌍을 씁니다. (유니코드)

구조체

OPENCARD_SEARCH_CRITERIAA OPENCARD_SEARCH_CRITERIA 구조는 호출자가 설정한 요구 사항을 충족하는 카드를 인식하기 위해 SCardUIDlgSelectCard 함수에서 사용됩니다. 그러나 이 구조를 사용하지 않고 SCardUIDlgSelectCard를 호출할 수 있습니다. (ANSI)
OPENCARD_SEARCH_CRITERIAW OPENCARD_SEARCH_CRITERIA 구조는 호출자가 설정한 요구 사항을 충족하는 카드를 인식하기 위해 SCardUIDlgSelectCard 함수에서 사용됩니다. 그러나 이 구조를 사용하지 않고 SCardUIDlgSelectCard를 호출할 수 있습니다. (유니코드)
OPENCARDNAME_EXA OPENCARDNAME_EX 구조에는 SCardUIDlgSelectCard 함수가 스마트 카드 카드 선택 대화 상자를 초기화하는 데 사용하는 정보가 포함되어 있습니다. (ANSI)
OPENCARDNAME_EXW OPENCARDNAME_EX 구조에는 SCardUIDlgSelectCard 함수가 스마트 카드 카드 선택 대화 상자를 초기화하는 데 사용하는 정보가 포함되어 있습니다. (유니코드)
OPENCARDNAMEA GetOpenCardName 함수가 스마트 카드 카드 선택 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. (ANSI)
OPENCARDNAMEW GetOpenCardName 함수가 스마트 카드 카드 선택 대화 상자를 초기화하는 데 사용하는 정보를 포함합니다. (유니코드)
SCARD_ATRMASK SCardLocateCardsByATR 함수에서 카드를 찾는 데 사용됩니다.
SCARD_READERSTATEA 판독기 내에서 스마트 카드를 추적하는 함수에서 사용됩니다. (ANSI)
SCARD_READERSTATEW 판독기 내에서 스마트 카드를 추적하는 함수에서 사용됩니다. (유니코드)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winsnmp.h 헤더

아티클2023. 08. 27.

이 헤더는 단순 네트워크 관리 프로토콜에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Simple Network Management Protocol

winsnmp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

SnmpCancelMsg WinSNMP 애플리케이션은 SnmpCancelMsg 함수를 호출하여 Microsoft WinSNMP 구현이 SNMP 요청 메시지에 대한 재전송 시도 및 시간 제한 알림을 취소하도록 요청합니다.
SnmpCleanup SnmpCleanup 함수는 호출하는 WinSNMP 애플리케이션에 더 이상 구현의 서비스가 필요하지 않음을 Microsoft WinSNMP 구현에 알릴 수 있습니다.
SnmpCleanupEx SnmpCleanupEx 함수는 WinSNMP(Windows SNMP) 애플리케이션 내에서 SnmpStartup 또는 SnmpStartupEx에 대한 미해결 호출이 없는 경우 정리를 수행합니다.
SnmpClose SnmpClose 함수를 사용하면 Microsoft WinSNMP 구현에서 WinSNMP 세션과 연결된 메모리, 리소스 및 데이터 구조의 할당을 취소할 수 있습니다.
SnmpContextToStr WinSNMP SnmpContextToStr 함수는 관리되는 개체 리소스 집합인 SNMP 컨텍스트를 식별하는 문자열을 반환합니다. 함수는 smiOCTETS 구조체의 문자열을 반환합니다.
SnmpCountVbl WinSNMP 애플리케이션은 WinSNMP SnmpCountVbl 함수를 호출하여 지정된 변수 바인딩 목록의 변수 바인딩 항목을 열거합니다.
SnmpCreatePdu WinSNMP SnmpCreatePdu 함수는 SNMP PDU(프로토콜 데이터 단위)를 만들고 초기화합니다.

SnmpCreateSession SnmpCreateSession 함수는 Microsoft WinSNMP 구현에 WinSNMP 애플리케이션에 대한 세션을 열도록 요청합니다.
SnmpCreateVbl WinSNMP SnmpCreateVbl 함수는 호출하는 WinSNMP 애플리케이션에 대한 새 변수 바인딩 목록을 만듭니다.
SnmpDecodeMsg WinSNMP SnmpDecodeMsg 함수는 인코딩된 SNMP 메시지를 해당 구성 요소로 디코딩합니다. 이 함수는 WinSNMP SnmpEncodeMsg 함수의 반대 동작을 수행합니다.
SnmpDeleteVb WinSNMP SnmpDeleteVb 함수는 변수 바인딩 목록에서 변수 바인딩 항목을 제거합니다.
SnmpDuplicatePdu WinSNMP SnmpDuplicatePdu 함수는 PDU 매개 변수가 식별하는 SNMP PDU(프로토콜 데이터 단위)를 복제하여 중복된 PDU에 필요한 메모리를 할당합니다.
SnmpDuplicateVbl WinSNMP SnmpDuplicateVbl 함수는 지정된 WinSNMP 세션에 대한 변수 바인딩 목록을 복사합니다. 이 함수는 복사된 변수 바인딩 목록에 대한 핸들을 반환하고 필요한 메모리를 할당합니다.
SnmpEncodeMsg Microsoft WinSNMP 구현은 WinSNMP SnmpEncodeMsg 함수에 전달된 매개 변수를 사용하여 SNMP 메시지를 인코딩합니다.
SnmpEntityToStr WinSNMP SnmpEntityToStr 함수는 SNMP 관리 엔터티를 식별하는 문자열을 반환합니다.
SnmpFreeContext WinSNMP SnmpFreeContext 함수는 관리되는 개체 리소스 집합인 SNMP 컨텍스트와 연결된 리소스를 해제합니다.
SnmpFreeDescriptor WinSNMP 애플리케이션은 SnmpFreeDescriptor 함수를 사용하여 더 이상 설명자 개체에 액세스할 필요가 없음을 Microsoft WinSNMP 구현에 알릴 수 있습니다.

SnmpFreeEntity WinSNMP SnmpFreeEntity 함수는 SNMP 관리 엔터티와 연결된 리소스를 해제합니다.
SnmpFreePdu WinSNMP SnmpFreePdu 함수는 SnmpCreatePdu 또는 SnmpDuplicatePdu 함수에서 만든 SNMP PDU(프로토콜 데이터 단위)와 연결된 리소스를 해제합니다.
SnmpFreeVbl WinSNMP SnmpFreeVbl 함수는 변수 바인딩 목록과 연결된 리소스를 해제합니다. 이는 WinSNMP 애플리케이션에서 SnmpCreateVbl 함수 또는 SnmpDuplicateVbl 함수를 호출하여 이전에 할당한 리소스입니다.
SnmpGetLastError WinSNMP SnmpGetLastError 함수는 호출 애플리케이션의 마지막 오류 코드 값을 반환합니다. 값은 WinSNMP 애플리케이션에서 실행된 마지막 함수 호출이 실패한 이유를 나타냅니다.
SnmpGetPduData WinSNMP SnmpGetPduData 함수는 지정된 SNMP 프로토콜 데이터 단위(PDU)에서 선택한 데이터 필드를 반환합니다.
SnmpGetRetransmitMode WinSNMP SnmpGetRetransmitMode 함수는 재전송 모드의 현재 설정을 WinSNMP 애플리케이션에 반환합니다.
SnmpGetRetry WinSNMP SnmpGetRetry 함수는 SNMP 메시지 요청의 재전송에 대한 재시도 횟수 값을 단위로 반환합니다. 재시도 횟수는 WinSNMP 애플리케이션이 지정된 관리 엔터티에 대한 SnmpSendMsg 함수에 대해 만드는 호출에 적용됩니다.
SnmpGetTimeout WinSNMP SnmpGetTimeout 함수는 SNMP 메시지 요청 전송에 대한 시간 제한 값을 100초 안에 반환합니다.
SnmpGetTranslateMode WinSNMP SnmpGetTranslateMode 함수는 엔터티 및 컨텍스트 변환 모드의 현재 설정을 WinSNMP 애플리케이션에 반환합니다.
SnmpGetVb WinSNMP 애플리케이션은 SnmpGetVb 함수를 호출하여 변수 바인딩 목록에서 정보를 검색합니

다. 이 WinSNMP 함수는 인덱스 매개 변수로 지정된 변수 바인딩 항목에서 변수 이름과 관련 값을 검색합니다.
SnmpGetVendorInfo WinSNMP 애플리케이션은 SnmpGetVendorInfo 함수를 호출하여 Microsoft WinSNMP 구현에 대한 정보를 검색합니다.
SnmpListen WinSNMP SnmpListen 함수는 WinSNMP 애플리케이션을 SNMP 에이전트로 등록합니다.
SnmpOidCompare WinSNMP SnmpOidCompare 함수는 maxlen 매개 변수에 지정된 길이까지 두 개의 SNMP 개체 식별자를 사전적으로 비교합니다.
SnmpOidCopy WinSNMP SnmpOidCopy 함수는 SNMP 개체 식별자를 복사하여 복사에 필요한 메모리를 할당합니다.
SnmpOidToStr WinSNMP SnmpOidToStr 함수는 SNMP 개체 식별자의 내부 이진 표현을 점선 숫자 문자열 형식(예: "1.2.3.4.5.6")으로 변환합니다.
SnmpOpen SnmpOpen 함수는 Microsoft WinSNMP 구현에 WinSNMP 애플리케이션에 대한 세션을 열도록 요청합니다.
SnmpRecvMsg WinSNMP SnmpRecvMsg 함수는 SNMP 메시지 형식으로 SnmpSendMsg 함수 호출로 제출된 완료된 비동기 요청의 결과를 검색합니다.
SnmpRegister WinSNMP SnmpRegister 함수는 트랩 및 알림 수신을 위해 WinSNMP 애플리케이션을 등록하거나 등록 취소합니다. 애플리케이션은 트랩 및 알림을 등록 및 수신하거나 트랩 및 알림을 등록 취소하고 사용하지 않도록 설정할 수 있습니다.
SnmpSendMsg WinSNMP 애플리케이션은 SnmpSendMsg 함수를 호출하여 Microsoft WinSNMP 구현이 SNMP 메시지 형식으로 SNMP PDU(프로토콜 데이터 단위)를 전송하도록 요청합니다.
SnmpSetPduData

WinSNMP SnmpSetPduData 함수는 지정된 SNMP 프로토콜 데이터 단위(PDU)에서 선택한 데이터 필드를 업데이트합니다.
SnmpSetPort WinSNMP 애플리케이션은 SnmpSetPort 함수를 호출하여 대상 엔터티에 할당된 포트를 변경합니다. SnmpSetPort 함수는 WinSNMP API 버전 2.0의 요소입니다.
SnmpSetRetransmitMode WinSNMP SnmpSetRetransmitMode 함수를 사용하면 WinSNMP 애플리케이션이 재전송 모드를 설정할 수 있습니다.
SnmpSetRetry WinSNMP SnmpSetRetry 함수를 사용하면 WinSNMP 애플리케이션에서 SNMP 메시지 요청 재전송에 대한 재시도 횟수 값을 변경할 수 있습니다.
SnmpSetTimeout WinSNMP SnmpSetTimeout 함수를 사용하면 WinSNMP 애플리케이션이 SNMP 메시지 요청 전송에 대한 제한 시간 값을 변경할 수 있습니다.
SnmpSetTranslateMode WinSNMP SnmpSetTranslateMode 함수를 사용하면 WinSNMP 애플리케이션이 엔터티 및 컨텍스트 변환 모드를 변경할 수 있습니다. 엔터티 및 컨텍스트 변환 모드는 WinSNMP 입력 및 출력 문자열 매개 변수의 해석 및 반환에 영향을 줍니다.
SnmpSetVb WinSNMP SnmpSetVb 함수는 변수 바인딩 목록의 변수 바인딩 항목을 변경합니다. 또한 이 함수는 기존 변수 바인딩 목록에 새 변수 바인딩 항목을 추가합니다.
SnmpStartup SnmpStartup 함수는 Microsoft WinSNMP 구현에 WinSNMP 애플리케이션에 구현의 서비스가 필요하다는 것을 알 수 있습니다.
SnmpStartupEx SnmpStartupEx 함수는 Microsoft WinSNMP 구현에 WinSNMP 애플리케이션에 구현의 서비스가 필요하다는 것을 알 수 있습니다.
SnmpStrToContext WinSNMP SnmpStrToContext 함수는 Microsoft WinSNMP 구현과 관련된 SNMP 컨텍스트 정보에 대한 핸들을 반환합니다.

SnmpStrToEntity WinSNMP SnmpStrToEntity 함수는 Microsoft WinSNMP 구현과 관련된 SNMP 관리 엔터티에 대한 정보에 대한 핸들을 반환합니다.
SnmpStrToOid WinSNMP SnmpStrToOid 함수는 SNMP 개체 식별자의 점선 숫자 문자열 형식(예: "1.2.3.4.5.6")을 내부 이진 표현으로 변환합니다.

콜백 함수

SNMPAPI_CALLBACK Microsoft WinSNMP 구현은 SNMPAPI_CALLBACK 함수를 호출하여 WinSNMP 세션에 SNMP 메시지 또는 비동기 이벤트를 사용할 수 있음을 알립니다.

구조체

smiCNTR64 WinSNMP smiCNTR64 구조체에는 64비트 부호 없는 정수 값이 포함됩니다. 구조체는 64비트 카운터를 나타냅니다.
smiOCTETS WinSNMP smiOCTETS 구조체는 컨텍스트 문자열을 여러 WinSNMP 함수에 전달합니다. 구조체는 인코딩된 SNMP 메시지를 설명하고 받습니다.
smiOID WinSNMP smiOID 구조체는 개체 식별자를 여러 WinSNMP 함수에 전달합니다. 또한 구조체는 SnmpGetVb 함수에 대한 호출에서 변수 바인딩 항목의 변수 이름을 받습니다.
smiVALUE WinSNMP smiVALUE 구조체는 변수 바인딩 항목의 변수 이름과 연결된 값을 설명합니다.
smiVENDORINFO

smiVENDORINFO 구조체에는 Microsoft WinSNMP 구현에 대한 정보가 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winsock.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows 소켓 2에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 소켓 2

winsock.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

__WSAFDIsSet __WSAFDIsSet 함수(winsock.h)는 소켓이 소켓 설명자 집합에 포함되는지 여부를 지정합니다.
AcceptEx AcceptEx 함수(winsock.h)는 새 연결을 수락하고 로컬 및 원격 주소를 반환하며 클라이언트 애플리케이션에서 보낸 첫 번째 데이터 블록을 받습니다.
bind bind 함수(winsock.h)는 로컬 주소를 소켓과 연결합니다.
closesocket closesocket 함수(winsock.h)는 기존 소켓을 닫습니다.
FD_SET FD_SET 매크로(winsock.h) 구조는 Windows 소켓(Winsock) 함수 및 서비스 공급자가 소켓을 집합에 배치하는 데 사용됩니다.
GetAcceptExSockaddrs GetAcceptExSockaddrs 함수(winsock.h)는 AcceptEx 함수 호출에서 얻은 데이터를 구문 분석하고 로컬 및 원격 주소를 sockaddr 구조체에 전달합니다.
gethostbyaddr Windows 소켓 2를 기준으로 gethostbyaddr 함수(winsock.h)는 더 이상 사용하지 않는 것이 좋습니다.
gethostbyname

gethostbyname 함수(winsock.h)는 호스트 데이터베이스에서 호스트 이름에 해당하는 호스트 정보를 검색합니다.
gethostname gethostname 함수(winsock.h)는 로컬 컴퓨터의 표준 호스트 이름을 검색합니다.
getpeername getpeername 함수(winsock.h)는 소켓이 연결된 피어의 주소를 검색합니다.
getprotobyname getprotobyname 함수(winsock.h)는 프로토콜 이름에 해당하는 프로토콜 정보를 검색합니다.
getprotobynumber getprotobynumber 함수(winsock.h)는 프로토콜 번호에 해당하는 프로토콜 정보를 검색합니다.
getservbyname getservbyname 함수(winsock.h)는 서비스 이름 및 프로토콜에 해당하는 서비스 정보를 검색합니다.
getservbyport getservbyport 함수(winsock.h)는 포트 및 프로토콜에 해당하는 서비스 정보를 검색합니다.
getsockname getsockname 함수(winsock.h)는 소켓의 로컬 이름을 검색합니다.
getsockopt getsockopt 함수(winsock.h)는 소켓 옵션을 검색합니다.
htonl htonl 함수(winsock.h)는 u_long 호스트에서 TCP/IP 네트워크 바이트 순서(big-endian)로 변환합니다.
htons htons 함수(winsock.h)는 u_short 호스트에서 TCP/IP 네트워크 바이트 순서(big-endian)로 변환합니다.
inet_addr

inet_addr 함수(winsock.h)는 IPv4 점선 소수점 주소가 포함된 문자열을 IN_ADDR 구조체에 적합한 주소로 변환합니다.
inet_ntoa inet_ntoa 함수(winsock.h)는 (Ipv4) 인터넷 네트워크 주소를 인터넷 표준 점선 소수점 형식의 ASCII 문자열로 변환합니다.
ioctlsocket ioctlsocket 함수(winsock.h)는 소켓의 I/O 모드를 제어합니다.
ntohl ntohl 함수(winsock.h)는 u_long TCP/IP 네트워크 순서에서 호스트 바이트 순서(Intel 프로세서의 little-endian)로 변환합니다.
ntohs ntohs 함수(winsock.h)는 u_short TCP/IP 네트워크 바이트 순서에서 호스트 바이트 순서(Intel 프로세서의 little-endian)로 변환합니다.
Recv recv 함수(winsock.h)는 연결된 소켓 또는 바인딩된 연결 없는 소켓에서 데이터를 수신합니다.
recvfrom recvfrom 함수는 데이터그램을 수신하고 원본 주소를 저장합니다.
Sendto sendto 함수(winsock.h)는 데이터를 특정 대상으로 보냅니다.
setsockopt setsockopt 함수(winsock.h)는 소켓 옵션을 설정합니다.
shutdown shutdown 함수(winsock.h)는 소켓에서 송신 또는 수신을 사용하지 않도록 설정합니다.
TransmitFile TransmitFile 함수(winsock.h)는 연결된 소켓 핸들을 통해 파일 데이터를 전송합니다.
WSAAsyncGetHostByAddr

WSAAsyncGetHostByAddr 함수(winsock.h)는 주소에 해당하는 호스트 정보를 비동기적으로 검색합니다.
WSAAsyncGetHostByName WSAAsyncGetHostByName 함수(winsock.h)는 호스트 이름에 해당하는 호스트 정보를 비동기적으로 검색합니다.
WSAAsyncGetProtoByName WSAAsyncGetProtoByName 함수(winsock.h)는 프로토콜 이름에 해당하는 프로토콜 정보를 비동기적으로 검색합니다.
WSAAsyncGetProtoByNumber WSAAsyncGetProtoByNumber 함수(winsock.h)는 프로토콜 번호에 해당하는 프로토콜 정보를 비동기적으로 검색합니다.
WSAAsyncGetServByName WSAAsyncGetServByName 함수(winsock.h)는 서비스 이름 및 포트에 해당하는 서비스 정보를 비동기적으로 검색합니다.
WSAAsyncGetServByPort WSAAsyncGetServByPort 함수(winsock.h)는 포트 및 프로토콜에 해당하는 서비스 정보를 비동기적으로 검색합니다.
WSAAsyncSelect WSAAsyncSelect 함수(winsock.h)는 소켓에 대한 네트워크 이벤트의 Windows 메시지 기반 알림을 요청합니다.
WSACancelAsyncRequest WSACancelAsyncRequest 함수(winsock.h)는 불완전한 비동기 작업을 취소합니다.
WSACleanup WSACleanup 함수(winsock.h)는 WS2_32.dll 사용을 종료합니다.
WSAGetLastError WSAGetLastError 함수(winsock.h)는 실패한 마지막 Windows 소켓 작업에 대한 오류 상태 반환합니다.
WSARecvEx WSARecvEx 함수(winsock.h)는 연결된 소켓 또는 바인딩된 연결 없는 소켓에서 데이터를 받습니다.

WSASetLastError WSASetLastError 함수(winsock.h)는 WSAGetLastError 함수를 통해 검색할 수 있는 오류 코드를 설정합니다.
WSAStartup WSAStartup 함수(winsock.h)는 프로세스에 의해 Winsock DLL 사용을 시작합니다.

구조체

fd_set fd_set 구조체(winsock.h)는 Windows 소켓(Winsock) 함수 및 서비스 공급자가 소켓을 집합에 배치하는 데 사용됩니다.
HOSTENT HOSTENT(winsock.h) 구조체는 함수에서 호스트 이름, IPv4 주소 등과 같은 지정된 호스트에 대한 정보를 저장하는 데 사용됩니다.
머무르게 LINGER(winsock.h) 구조체는 데이터를 보낼 큐에 대기할 때 해당 소켓이 작동하는 방식을 지정하는 특정 소켓에 대한 정보를 유지 관리합니다.
PROTOENT PROTOENT(winsock.h) 구조체에는 지정된 프로토콜 이름에 해당하는 이름 및 프로토콜 번호가 포함됩니다.
SERVENT SERVENT(winsock.h) 구조체는 지정된 서비스 이름에 대한 이름 및 서비스 번호를 저장하거나 반환하는 데 사용됩니다.
SOCKADDR sockaddr 구조는 선택한 프로토콜에 따라 달라집니다. (sockaddr)
SOCKADDR_IN SOCKADDR_IN(winsock.h) 구조는 선택한 프로토콜에 따라 달라집니다.

TIMEVAL TIMEVAL(winsock.h) 구조체는 시간 간격을 지정하는 데 사용됩니다. BSD(Berkeley Software Distribution) Time.h 헤더 파일과 연결됩니다.
TRANSMIT_FILE_BUFFERS TRANSMIT_FILE_BUFFERS(winsock.h) 구조는 TransmitFile 함수 파일 전송 작업 중 파일 데이터 전후에 전송할 데이터를 지정합니다.
WSADATA WSADATA(winsock.h) 구조체에는 Windows 소켓 구현에 대한 정보가 포함되어 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winsock2.h 헤더

아티클2023. 03. 08.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Vista 이전의 네트워크 드라이버
- 서비스 품질(QOS)
- Windows 소켓 2

winsock2.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

__WSAFDIsSet __WSAFDIsSet 함수(winsock2.h)는 소켓이 소켓 설명자 집합에 포함되는지 여부를 지정합니다.
받아들일 accept 함수는 소켓에서 들어오는 연결 시도를 허용합니다.
bind 바인딩 함수는 로컬 주소를 소켓과 연결합니다. (bind 함수(winsock2.h))
closesocket closesocket 함수는 기존 소켓을 닫습니다. (closesocket 함수(winsock2.h))
connect Connect 함수는 지정된 소켓에 대한 연결을 설정합니다.
FD_SET FD_SET 매크로(winsock2.h)는 Windows 소켓(Winsock) 함수 및 서비스 공급자가 소켓을 집합에 배치하는 데 사용됩니다.
gethostbyaddr gethostbyaddr 함수는 네트워크 주소에 해당하는 호스트 정보를 검색합니다. (gethostbyaddr는 더 이상 권장되지 않으며 대신 getnameinfo를 사용합니다.)

gethostbyname gethostbyname 함수(winsock2.h) 함수는 호스트 데이터베이스에서 호스트 이름에 해당하는 호스트 정보를 검색합니다.
gethostname gethostname 함수(winsock2.h)는 로컬 컴퓨터의 표준 호스트 이름을 검색합니다.
GetHostNameW GetHostNameW 함수는 로컬 컴퓨터의 표준 호스트 이름을 유니코드 문자열로 검색합니다.
getpeername getpeername 함수(winsock2.h)는 소켓이 연결된 피어의 주소를 검색합니다.
getprotobyname getprotobyname 함수(winsock2.h)는 프로토콜 이름에 해당하는 프로토콜 정보를 검색합니다.
getprotobynumber getprotobynumber 함수(winsock2.h)는 프로토콜 번호에 해당하는 프로토콜 정보를 검색합니다.
getservbyname getservbyname 함수(winsock2.h)는 서비스 이름 및 프로토콜에 해당하는 서비스 정보를 검색합니다.
getservbyport getservbyport 함수(winsock2.h)는 포트 및 프로토콜에 해당하는 서비스 정보를 검색합니다.
getsockname getsockname 함수(winsock2.h)는 지정된 소켓 설명자의 로컬 이름을 검색합니다.
getsockopt getsockopt 함수는 모든 형식의 소켓과 연결된 소켓 옵션의 현재 값을 모든 상태에서 검색하고 결과를 optval에 저장합니다.
htond double을 호스트에서 TCP/IP 네트워크 바이트 순서(big-endian)로 변환합니다.

htonf 호스트에서 TCP/IP 네트워크 바이트 순서(big-endian)로 float를 변환합니다.
htonl htonl 함수(winsock2.h)는 u_long 호스트에서 big-endian인 TCP/IP 네트워크 바이트 순서로 변환합니다.
htonll 부호 없는 __int64 호스트에서 TCP/IP 네트워크 바이트 순서(big-endian)로 변환합니다.
htons htons 함수(winsock2.h)는 u_short 호스트에서 big-endian인 TCP/IP 네트워크 바이트 순서로 변환합니다.
inet_addr inet_addr 함수(winsock2.h)는 IPv4 점선 소수점 주소가 포함된 문자열을 IN_ADDR 구조체에 적합한 주소로 변환합니다.
inet_ntoa inet_ntoa 함수(winsock2.h)는 (Ipv4) 인터넷 네트워크 주소를 인터넷 표준 점선 소수점 형식의 ASCII 문자열로 변환합니다.
ioctlsocket ioctlsocket 함수(winsock2.h)는 소켓의 I/O 모드를 제어하며 모든 상태의 모든 소켓에서 사용할 수 있습니다.
listen 수신 대기 함수는 들어오는 연결을 수신 대기하는 상태에 소켓을 배치합니다.
ntohd 서명되지 않은 __int64 TCP/IP 네트워크 순서에서 호스트 바이트 순서(Intel 프로세서의 little-endian)로 변환하고 double을 반환합니다.
ntohf 서명되지 않은 __int32 TCP/IP 네트워크 순서에서 호스트 바이트 순서(Intel 프로세서의 little-endian)로 변환하고 float를 반환합니다.

ntohl ntohl 함수(winsock2.h)는 TCP/IP 네트워크 순서에서 호스트 바이트 순서(Intel 프로세서의 little-endian)로 u_long 변환합니다.
ntohll 서명되지 않은 __int64 TCP/IP 네트워크 순서에서 호스트 바이트 순서(Intel 프로세서의 little-endian)로 변환합니다.
ntohs ntohs 함수(winsock2.h)는 u_short TCP/IP 네트워크 바이트 순서에서 호스트 바이트 순서(Intel 프로세서의 little-endian)로 변환합니다.
ProcessSocketNotifications 소켓 집합을 완료 포트와 연결하고 해당 포트에서 이미 보류 중인 알림을 검색합니다. 연결되면 완료 포트는 지정된 소켓 상태 알림을 받습니다.
Recv recv 함수(winsock2.h)는 연결된 소켓 또는 바운드 연결 없는 소켓에서 데이터를 받습니다.
recvfrom recvfrom 함수는 데이터그램을 수신하고 원본 주소를 저장합니다.
선택 select 함수는 동기 I/O를 수행하기 위해 필요한 경우 대기하는 하나 이상의 소켓 상태를 결정합니다.
send 연결된 소켓에 데이터를 보냅니다. (보내기)
Sendto sendto 함수(winsock2.h)는 소켓에 나가는 데이터를 쓰는 데 사용됩니다.
setsockopt setsockopt 함수(winsock2.h)는 모든 형식의 소켓과 연결된 소켓 옵션의 현재 값을 모든 상태로 설정합니다.

shutdown 종료 함수(winsock2.h)는 수신, 전송 또는 둘 다를 사용하지 않도록 설정하기 위해 모든 유형의 소켓에서 사용됩니다.
socket 소켓 함수는 특정 전송 서비스 공급자에 바인딩된 소켓을 만듭니다.
SocketNotificationRetrieveEvents 이 인라인 도우미 함수는 OVERLAPPED_ENTRY 이벤트 마스크를 검색하는 편의를 위해 제공됩니다.
WSAAccept WSAAccept 함수는 조건부로 조건부 함수의 반환 값에 따라 연결을 허용하고 서비스 흐름 사양의 품질을 제공하며 연결 데이터 전송을 허용합니다.
WSAAddressToStringA sockaddr 구조체의 모든 구성 요소를 사람이 읽을 수 있는 주소의 문자열 표현으로 변환합니다. (ANSI)
WSAAddressToStringW sockaddr 구조체의 모든 구성 요소를 사람이 읽을 수 있는 주소의 문자열 표현으로 변환합니다. (유니코드)
WSAAsyncGetHostByAddr WSAAsyncGetHostByAddr 함수(winsock2.h)는 주소에 해당하는 호스트 정보를 비동기적으로 검색합니다.
WSAAsyncGetHostByName WSAAsyncGetHostByName 함수(winsock2.h)는 호스트 이름에 해당하는 호스트 정보를 비동기적으로 검색합니다.
WSAAsyncGetProtoByName WSAAsyncGetProtoByName 함수(winsock2.h)는 프로토콜 이름에 해당하는 프로토콜 정보를 비동기적으로 검색합니다.
WSAAsyncGetProtoByNumber WSAAsyncGetProtoByNumber 함수(winsock2.h)는 프로토콜 번호에 해당하는 프로토콜 정보를 비동기적으로 검색합니다.

WSAAsyncGetServByName WSAAsyncGetServByName 함수(winsock2.h)는 서비스 이름 및 포트에 해당하는 서비스 정보를 비동기적으로 검색합니다.
WSAAsyncGetServByPort WSAAsyncGetServByPort 함수(winsock2.h)는 포트 및 프로토콜에 해당하는 서비스 정보를 비동기적으로 검색합니다.
WSAAsyncSelect WSAAsyncSelect 함수(winsock2.h)는 소켓에 대한 네트워크 이벤트의 Windows 메시지 기반 알림을 요청합니다.
WSACancelAsyncRequest WSACancelAsyncRequest 함수(winsock2.h)는 불완전한 비동기 작업을 취소합니다.
WSACancelBlockingCall WSACancelBlockingCall 함수는 Windows 소켓 2 사양, 수정 버전 2.2.0에 따라 제거되었습니다.
WSACleanup WSACleanup 함수(winsock2.h)는 WS2_32.dll 사용을 종료합니다.
WSACloseEvent WSACloseEvent 함수는 열린 이벤트 개체 핸들을 닫습니다.
WSAConnect WSAConnect 함수는 다른 소켓 애플리케이션에 대한 연결을 설정하고, 연결 데이터를 교환하며, 지정된 FLOWSPEC 구조에 따라 필요한 서비스 품질을 지정합니다.
WSAConnectByList 대상 주소 집합(호스트 이름 및 포트)이 나타내는 가능한 엔드포인트 컬렉션 중 하나에 대한 연결을 설정합니다.
WSAConnectByNameA 지정된 호스트 및 포트에 대한 연결을 설정합니다. (ANSI)
WSAConnectByNameW 지정된 호스트 및 포트에 대한 연결을 설정합니다. (유니코드)

WSACreateEvent WSACreateEvent 함수는 새 이벤트 개체를 만듭니다.
WSADuplicateSocketA WSADuplicateSocket 함수는 공유 소켓에 대한 새 소켓 설명자를 만드는 데 사용할 수 있는 WSAPROTOCOL_INFO 구조를 반환합니다. WSADuplicateSocket 함수는 QOS 사용 소켓에서 사용할 수 없습니다. (ANSI)
WSADuplicateSocketW WSADuplicateSocket 함수는 공유 소켓에 대한 새 소켓 설명자를 만드는 데 사용할 수 있는 WSAPROTOCOL_INFO 구조를 반환합니다. WSADuplicateSocket 함수는 QOS 사용 소켓에서 사용할 수 없습니다. (유니코드)
WSAEnumNameSpaceProvidersA WSAEnumNameSpaceProviders 함수는 사용 가능한 네임스페이스 공급자에 대한 정보를 검색합니다. (ANSI)
WSAEnumNameSpaceProvidersExA 사용 가능한 네임스페이스 공급자에 대한 정보를 검색합니다. (ANSI)
WSAEnumNameSpaceProvidersExW 사용 가능한 네임스페이스 공급자에 대한 정보를 검색합니다. (유니코드)
WSAEnumNameSpaceProvidersW WSAEnumNameSpaceProviders 함수는 사용 가능한 네임스페이스 공급자에 대한 정보를 검색합니다. (유니코드)
WSAEnumNetworkEvents WSAEnumNetworkEvents 함수는 표시된 소켓에 대한 네트워크 이벤트 발생, 내부 네트워크 이벤트 레코드 지우기 및 이벤트 개체 재설정(선택 사항)을 검색합니다.
WSAEnumProtocolsA WSAEnumProtocols 함수는 사용 가능한 전송 프로토콜에 대한 정보를 검색합니다. (ANSI)
WSAEnumProtocolsW WSAEnumProtocols 함수는 사용 가능한 전송 프로토콜에 대한 정보를 검색합니다. (유니코드)

WSAEventSelect WSAEventSelect 함수는 지정된 FD_XXX 네트워크 이벤트 집합과 연결할 이벤트 개체를 지정합니다.
WSAGetLastError WSAGetLastError 함수(winsock2.h)는 실패한 마지막 Windows 소켓 작업에 대한 오류 상태를 반환합니다.
WSAGetOverlappedResult WSAGetOverlappedResult 함수는 지정된 소켓에서 겹치는 작업의 결과를 검색합니다.
WSAGetQOSByName WSAGetQOSByName 함수는 명명된 템플릿을 기반으로 QOS 구조를 초기화하거나 사용 가능한 템플릿 이름의 열거형을 검색하는 버퍼를 제공합니다.
WSAGetServiceClassInfoA WSAGetServiceClassInfo 함수는 지정된 네임스페이스 공급자에서 지정된 서비스 클래스와 관련된 클래스 정보(스키마)를 검색합니다. (ANSI)
WSAGetServiceClassInfoW WSAGetServiceClassInfo 함수는 지정된 네임스페이스 공급자에서 지정된 서비스 클래스와 관련된 클래스 정보(스키마)를 검색합니다. (유니코드)
WSAGetServiceClassNameByClassIdA WSAGetServiceClassNameByClassId 함수는 지정된 형식과 연결된 서비스의 이름을 검색합니다. 이 이름은 FTP 또는 SNA와 같은 일반 서비스 이름이며 해당 서비스의 특정 인스턴스 이름이 아닙니다. (ANSI)
WSAGetServiceClassNameByClassIdW WSAGetServiceClassNameByClassId 함수는 지정된 형식과 연결된 서비스의 이름을 검색합니다. 이 이름은 FTP 또는 SNA와 같은 일반 서비스 이름이며 해당 서비스의 특정 인스턴스 이름이 아닙니다. (유니코드)
WSAHtonl WSAHtonl 함수는 호스트 바이트 순서에서 네트워크 바이트 순서로 u_long 변환합니다.
WSAHtons WSAHtons 함수는 호스트 바이트 순서에서 네트워크 바이트 순서로 u_short 변환합니다.

WSAInstallServiceClassA WSAInstallServiceClass 함수는 네임스페이스 내에 서비스 클래스 스키마를 등록합니다. (ANSI)
WSAInstallServiceClassW WSAInstallServiceClass 함수는 네임스페이스 내에 서비스 클래스 스키마를 등록합니다. (유니코드)
WSAIoctl WSAIoctl 함수는 소켓의 모드를 제어합니다.
WSAIsBlocking 이 함수는 Windows 소켓 2 사양, 수정 버전 2.2.0에 따라 제거되었습니다. (WSAIsBlocking)
WSAJoinLeaf WSAJoinLeaf 함수는 리프 노드를 다중 포인트 세션에 조인하고, 연결 데이터를 교환하고, 지정된 FLOWSPEC 구조에 따라 필요한 서비스 품질을 지정합니다.
WSALookupServiceBeginA WSALookupServiceBegin 함수는 WSAQUERYSET 구조에 포함된 정보로 제한되는 클라이언트 쿼리를 시작합니다. (ANSI)
WSALookupServiceBeginW WSALookupServiceBegin 함수는 WSAQUERYSET 구조에 포함된 정보로 제한되는 클라이언트 쿼리를 시작합니다. (유니코드)
WSALookupServiceEnd WSALookupServiceEnd 함수는 WSALookupServiceBegin 및 WSALookupServiceNext에 대한 이전 호출 후 핸들을 해제하기 위해 호출됩니다.
WSALookupServiceNextA WSALookupServiceNext 함수는 요청된 서비스 정보를 검색하기 위해 WSALookupServiceBegin에 대한 이전 호출에서 핸들을 가져온 후 호출됩니다. (ANSI)
WSALookupServiceNextW WSALookupServiceNext 함수는 요청된 서비스 정보를 검색하기 위해 WSALookupServiceBegin에 대한 이전 호출에서 핸들을 가져온 후 호출됩니다. (유니코드)
WSANSPIoctl 개발자가 등록된 네임스페이스에 대한 I/O 제어 호출을 수행할 수 있도록 합니다.

WSANtohl WSANtohl 함수는 u_long 네트워크 바이트 순서에서 호스트 바이트 순서로 변환합니다.
WSANtohs WSANtohs 함수는 u_short 네트워크 바이트 순서에서 호스트 바이트 순서로 변환합니다.
WSAPoll WSAPoll 함수는 하나 이상의 소켓 상태를 결정합니다.
WSAProviderConfigChange WSAProviderConfigChange 함수는 공급자 구성이 변경될 때 애플리케이션에 알합니다.
WSARecv 연결된 소켓 또는 바인딩된 연결 없는 소켓에서 데이터를 받습니다. (WSARecv)
WSARecvDisconnect WSARecvDisconnect 함수는 소켓에서 수신을 종료하고 소켓이 연결 지향인 경우 연결 끊기 데이터를 검색합니다.
WSARecvFrom 데이터그램을 수신하고 원본 주소를 저장합니다.
WSARemoveServiceClass WSARemoveServiceClass 함수는 레지스트리에서 서비스 클래스 스키마를 영구적으로 제거합니다.
WSAResetEvent WSAResetEvent 함수는 지정된 이벤트 개체의 상태를 서명되지 않은 상태로 다시 설정합니다.
WSASend 연결된 소켓에 데이터를 보냅니다. (WSASend)
WSASendDisconnect WSASendDisconnect 함수는 소켓에 대한 연결 종료를 시작하고 연결 끊기 데이터를 보냅니다.
WSASendMsg 연결된 소켓과 연결되지 않은 소켓에서 데이터 및 선택적 제어 정보를 보냅니다. 참고: 이 함수는 Windows 소켓 사양에 대한 Microsoft 전용 확장입니다. .

WSASendTo 해당하는 경우 겹치는 I/O를 사용하여 특정 대상으로 데이터를 보냅니다.
WSASetBlockingHook 이 함수는 Windows 소켓 2 사양, 수정 버전 2.2.0에 따라 제거되었습니다. (WSASetBlockingHook)
WSASetEvent WSASetEvent 함수는 지정된 이벤트 개체의 상태를 신호로 설정합니다.
WSASetLastError WSASetLastError 함수(winsock2.h)는 WSAGetLastError 함수를 통해 검색할 수 있는 오류 코드를 설정합니다.
WSASetServiceA WSASetService 함수는 하나 이상의 네임스페이스 내의 서비스 인스턴스를 레지스트리에서 등록하거나 제거합니다. (ANSI)
WSASetServiceW WSASetService 함수는 하나 이상의 네임스페이스 내의 서비스 인스턴스를 레지스트리에서 등록하거나 제거합니다. (유니코드)
WSASocketA WSASocket 함수는 특정 전송 서비스 공급자에 바인딩된 소켓을 만듭니다. (ANSI)
WSASocketW WSASocket 함수는 특정 전송 서비스 공급자에 바인딩된 소켓을 만듭니다. (유니코드)
WSAStartup WSAStartup 함수(winsock2.h)는 프로세스에 의해 Winsock DLL 사용을 시작합니다.
WSAStringToAddressA WSAStringToAddress 함수는 표준 텍스트 프레젠테이션 형식의 네트워크 주소를 sockaddr 구조의 숫자 이진 형식으로 변환하며, 이러한 구조를 사용하는 Windows 소켓 루틴에 전달하는 데 적합합니다. (ANSI)

WSAStringToAddressW WSAStringToAddress 함수는 표준 텍스트 프레젠테이션 형식의 네트워크 주소를 sockaddr 구조의 숫자 이진 형식으로 변환하며, 이러한 구조를 사용하는 Windows 소켓 루틴에 전달하는 데 적합합니다. (유니코드)
WSAUnhookBlockingHook 이 함수는 Windows 소켓 2 사양, 수정 버전 2.2.0에 따라 제거되었습니다. (WSAUnhookBlockingHook)
WSAWaitForMultipleEvents 지정된 이벤트 개체 중 하나 또는 전부가 신호 상태에 있거나, 제한 시간 간격이 만료되거나, I/O 완료 루틴이 실행된 경우를 반환합니다.

콜백 함수

LPWSAOVERLAPPED_COMPLETION_ROUTINE TBD(LPWSAOVERLAPPED_COMPLETION_ROUTINE)

구조체

AFPROTOCOLS AFPROTOCOLS 구조는 애플리케이션 프로그래머가 쿼리를 제한할 수 있는 프로토콜 목록을 제공합니다. AFPROTOCOLS 구조는 쿼리 목적으로만 사용됩니다.
BLOB Binary Large 개체에서 파생된 BLOB 구조체(winsock2.h)에는 데이터 블록에 대한 정보가 포함됩니다.
fd_set Fd_set 구조체(winsock2.h)는 Windows 소켓(Winsock) 함수 및 서비스 공급자가 소켓을 세트에 배치하는 데 사용됩니다.

HOSTENT HOSTENT 구조체(winsock2.h)는 호스트 이름, IPv4 주소 등과 같은 지정된 호스트에 대한 정보를 저장하는 함수에서 사용됩니다.
in_addr in_addr 구조는 IPv4 인터넷 주소를 나타냅니다.
머무르게 LINGER 구조(winsock2.h)는 데이터가 큐에 대기될 때 소켓이 동작하는 방식을 지정하는 특정 소켓에 대한 정보를 유지 관리합니다.
PROTOENT PROTOENT 구조체(winsock2.h)에는 지정된 프로토콜 이름에 해당하는 이름 및 프로토콜 번호가 포함됩니다.
Qos QOS 구조는 QOS 지원 애플리케이션이 특정 흐름에서 전송 및 수신된 트래픽에 대한 서비스 매개 변수의 품질을 지정할 수 있는 방법을 제공합니다.
SERVENT SERVENT 구조체(winsock2.h)는 지정된 서비스 이름에 대한 이름 및 서비스 번호를 저장하거나 반환하는 데 사용됩니다.
SOCK_NOTIFY_REGISTRATION ProcessSocketNotifications 함수에 제공된 정보를 나타냅니다.
TIMEVAL TIMEVAL 구조체(winsock2.h)는 시간 간격을 지정하는 데 사용됩니다. BSD(Berkeley Software Distribution) Time.h 헤더 파일과 연결됩니다.
WSACOMPLETION 등록된 네임스페이스에 대한 I/O 컨트롤 호출에 대한 완료 알림 설정을 지정합니다.
WSADATA WSADATA 구조체(winsock2.h)에는 Windows 소켓 구현에 대한 정보가 포함되어 있습니다.
WSANAMESPACE_INFOA 네임스페이스 공급자에 대한 모든 등록 정보를 포함합니다. (WSANAMESPACE_INFOA)

WSANAMESPACE_INFOEXA 네임스페이스 공급자에 대한 모든 등록 정보를 포함합니다. (WSANAMESPACE_INFOEXA)
WSANAMESPACE_INFOEXW 네임스페이스 공급자에 대한 모든 등록 정보를 포함합니다. (WSANAMESPACE_INFOEXW)
WSANAMESPACE_INFOW 네임스페이스 공급자에 대한 모든 등록 정보를 포함합니다. (WSANAMESPACE_INFOW)
WSANETWORKEVENTS WSANETWORKEVENTS 구조체는 네트워크 이벤트에 대한 소켓의 내부 정보를 저장하는 데 사용됩니다.
WSANSCLASSINFOA WSANSCLASSINFO 구조체는 특정 Windows 소켓 네임스페이스에 대한 개별 매개 변수 정보를 제공합니다. (ANSI)
WSANSCLASSINFOW WSANSCLASSINFO 구조체는 특정 Windows 소켓 네임스페이스에 대한 개별 매개 변수 정보를 제공합니다. (유니코드)
WSAOVERLAPPED 겹치는 I/O 작업의 시작과 이후 완료 사이의 통신 매체를 제공합니다.
WSAPOLLFD WSAPoll 함수에서 사용하는 소켓 정보를 저장합니다.
WSAPROTOCOL_INFOA 지정된 프로토콜에 대한 전체 정보를 저장하거나 검색하는 데 사용됩니다. (ANSI)
WSAPROTOCOL_INFOW 지정된 프로토콜에 대한 전체 정보를 저장하거나 검색하는 데 사용됩니다. (유니코드)
WSAPROTOCOLCHAIN WSAPROTOCOLCHAIN 구조에는 프로토콜 체인을 구성하는 카탈로그 항목 식별자의 계산된 목록이 포함되어 있습니다.

WSAQUERYSET2A 서비스 클래스 ID, 서비스 이름, 적용 가능한 네임스페이스 식별자 및 프로토콜 정보뿐만 아니라 서비스가 수신 대기하는 전송 주소 집합을 포함하여 지정된 서비스에 대한 관련 정보를 제공합니다. (ANSI)
WSAQUERYSET2W 서비스 클래스 ID, 서비스 이름, 적용 가능한 네임스페이스 식별자 및 프로토콜 정보뿐만 아니라 서비스가 수신 대기하는 전송 주소 집합을 포함하여 지정된 서비스에 대한 관련 정보를 제공합니다. (유니코드)
WSAQUERYSETA 서비스 클래스 ID, 서비스 이름, 적용 가능한 네임스페이스 식별자 및 프로토콜 정보뿐만 아니라 서비스가 수신 대기하는 전송 주소 집합을 포함하여 지정된 서비스에 대한 관련 정보를 제공합니다. (ANSI)
WSAQUERYSETW 서비스 클래스 ID, 서비스 이름, 적용 가능한 네임스페이스 식별자 및 프로토콜 정보뿐만 아니라 서비스가 수신 대기하는 전송 주소 집합을 포함하여 지정된 서비스에 대한 관련 정보를 제공합니다. (유니코드)
WSASERVICECLASSINFOA WSASERVICECLASSINFO 구조체에는 지정된 서비스 클래스에 대한 정보가 포함됩니다. Windows Sockets 2의 각 서비스 클래스에는 단일 WSASERVICECLASSINFO 구조가 있습니다. (ANSI)
WSASERVICECLASSINFOW WSASERVICECLASSINFO 구조체에는 지정된 서비스 클래스에 대한 정보가 포함됩니다. Windows Sockets 2의 각 서비스 클래스에는 단일 WSASERVICECLASSINFO 구조가 있습니다. (유니코드)
WSAVERSION WSAVERSION 구조는 Windows 소켓에서 버전 비교를 제공합니다.

열거형

WSAECOMPARATOR Windows 소켓 WSAECOMPARATOR 열거형 형식은 Windows 소켓 2의 버전 비교 의미 체계에 사용됩니다.

winstring.h 헤더

아티클2023. 03. 08.

이 헤더는 Windows 런타임 C++ 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 런타임 C++ 참조

winstring.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

HSTRING_UserFree HSTRING_UserFree 함수(winstring.h)는 RPC 스텁 파일에서 호출할 때 서버 쪽에서 리소스를 해제합니다.
HSTRING_UserMarshal HSTRING_UserMarshal 함수(winstring.h)는 HSTRING 개체를 RPC 버퍼로 마샬링합니다.
HSTRING_UserSize HSTRING_UserSize 함수(winstring.h)는 HSTRING 개체의 와이어 크기를 계산하고 핸들과 데이터를 검색합니다.
HSTRING_UserUnmarshal HSTRING_UserUnmarshal 함수(winstring.h)는 RPC 버퍼에서 HSTRING 개체를 숨기지 않습니다.
WindowsCompareStringOrdinal 지정된 두 HSTRING 개체를 비교하고 정렬 순서로 상대 위치를 나타내는 정수 값을 반환합니다.
WindowsConcatString 지정된 두 문자열을 연결합니다.
WindowsCreateString 지정된 소스 문자열을 기반으로 새 HSTRING을 만듭니다.
WindowsCreateStringReference 지정된 문자열을 기반으로 새 문자열 참조를 만듭니다.

WindowsDeleteString 문자열 버퍼의 참조 수를 줄입니다.
WindowsDeleteStringBuffer HSTRING으로 승격되지 않은 경우 미리 할당된 문자열 버퍼를 삭제합니다.
WindowsDuplicateString 지정된 문자열의 복사본을 만듭니다.
WindowsGetStringLen 지정한 문자열의 길이(유니코드 문자)를 가져옵니다.
WindowsGetStringRawBuffer 지정된 문자열에 대한 지원 버퍼를 검색합니다.
WindowsInspectString 디버거가 다른 주소 공간, 원격 또는 덤프에서 Windows 런타임 HSTRING의 값을 표시하는 방법을 제공합니다. (WindowsInspectString)
WindowsInspectString2 디버거가 다른 주소 공간, 원격 또는 덤프에서 Windows 런타임 HSTRING의 값을 표시하는 방법을 제공합니다. (WindowsInspectString2)
WindowsIsStringEmpty 지정된 문자열이 빈 문자열인지 여부를 나타냅니다.
WindowsPreallocateStringBuffer HSTRING 생성에 사용할 변경 가능한 문자 버퍼를 할당합니다.
WindowsPromoteStringBuffer 지정된 HSTRING_BUFFER HSTRING을 만듭니다.
WindowsReplaceString 지정된 문자열의 모든 문자 집합을 다른 문자 집합으로 바꿔 새 문자열을 만듭니다.
WindowsStringHasEmbeddedNull 지정된 문자열에 null 문자가 포함되어 있는지 여부를 나타냅니다.

WindowsSubstring 지정된 문자열에서 부분 문자열을 검색합니다. 부분 문자열은 지정된 문자 위치에서 시작됩니다.
WindowsSubstringWithSpecifiedLength 지정된 문자열에서 부분 문자열을 검색합니다. 부분 문자열은 지정된 문자 위치에서 시작하고 길이도 지정되어 있습니다.
WindowsTrimStringEnd 원본 문자열에서 지정된 문자 집합의 후행 발생을 모두 제거합니다.
WindowsTrimStringStart 원본 문자열에서 지정된 문자 집합의 선행 항목을 모두 제거합니다.

콜백 함수

PINSPECT_HSTRING_CALLBACK WindowsInspectString 함수에서 사용하는 콜백에 대한 함수 포인터를 제공합니다.

winsvc.h 헤더

아티클2023. 03. 08.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID
- 시스템 서비스

winsvc.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

ChangeServiceConfig2A 서비스의 선택적 구성 매개 변수를 변경합니다. (ANSI)
ChangeServiceConfig2W 서비스의 선택적 구성 매개 변수를 변경합니다. (유니코드)
ChangeServiceConfigA 서비스의 구성 매개 변수를 변경합니다. (ANSI)
ChangeServiceConfigW 서비스의 구성 매개 변수를 변경합니다. (유니코드)
CloseServiceHandle 서비스 제어 관리자 또는 서비스 개체에 대한 핸들을 닫습니다.
ControlService 제어 코드를 서비스로 보냅니다. (ControlService)
ControlServiceExA 제어 코드를 서비스로 보냅니다. (ControlServiceExA)
ControlServiceExW 제어 코드를 서비스로 보냅니다. (ControlServiceExW)

CreateServiceA 서비스 개체를 만들고 지정된 서비스 제어 관리자 데이터베이스에 추가합니다. (ANSI)
CreateServiceW 서비스 개체를 만들고 지정된 서비스 제어 관리자 데이터베이스에 추가합니다. (유니코드)
DeleteService 서비스 제어 관리자 데이터베이스에서 삭제할 지정된 서비스를 표시합니다.
EnumDependentServicesA 지정된 서비스에 따라 달라지는 각 서비스의 이름과 상태를 검색합니다. (ANSI)
EnumDependentServicesW 지정된 서비스에 따라 달라지는 각 서비스의 이름과 상태를 검색합니다. (유니코드)
EnumServicesStatusA 지정된 서비스 제어 관리자 데이터베이스의 서비스를 열거합니다. 각 서비스의 이름과 상태가 제공됩니다. (ANSI)
EnumServicesStatusExA 지정된 서비스 제어 관리자 데이터베이스의 서비스를 열거합니다. 지정된 정보 수준에 따라 추가 데이터와 함께 각 서비스의 이름 및 상태가 제공됩니다. (ANSI)
EnumServicesStatusExW 지정된 서비스 제어 관리자 데이터베이스의 서비스를 열거합니다. 지정된 정보 수준에 따라 추가 데이터와 함께 각 서비스의 이름 및 상태가 제공됩니다. (유니코드)
EnumServicesStatusW 지정된 서비스 제어 관리자 데이터베이스의 서비스를 열거합니다. 각 서비스의 이름과 상태가 제공됩니다. (유니코드)
GetServiceDirectory 서비스가 상태를 읽고/또는 쓸 서비스별 파일 시스템 위치에 대한 경로를 반환합니다.
GetServiceDisplayNameA 지정된 서비스의 표시 이름을 검색합니다. (ANSI)

GetServiceDisplayNameW 지정된 서비스의 표시 이름을 검색합니다. (유니코드)
GetServiceKeyNameA 지정된 서비스의 서비스 이름을 검색합니다. (ANSI)
GetServiceKeyNameW 지정된 서비스의 서비스 이름을 검색합니다. (유니코드)
GetServiceRegistryStateKey 서비스에서 상태를 읽고/또는 쓸 레지스트리 키에 대한 핸들을 반환합니다.
GetSharedServiceDirectory 서비스 및 관련 프로그램이 상태를 읽고/또는 쓸 수 있도록 서비스별 파일 시스템 위치에 대한 경로를 반환합니다.
GetSharedServiceRegistryStateKey 상태를 읽고/또는 쓸 서비스 및 관련 프로그램에 대한 레지스트리 키에 대한 핸들을 반환합니다.
LockServiceDatabase SCM(서비스 제어 관리자) 데이터베이스 잠금의 소유권을 요청합니다. 지정된 시간에는 하나의 프로세스만 잠금을 소유할 수 있습니다.
NotifyBootConfigStatus 부팅 상태를 서비스 제어 관리자에 보고합니다. 부팅 확인 프로그램에서 사용됩니다.
NotifyServiceStatusChangeA 지정된 서비스를 만들거나 삭제하거나 상태가 변경될 때 애플리케이션이 알림을 받을 수 있도록 합니다. (ANSI)
NotifyServiceStatusChangeW 지정된 서비스를 만들거나 삭제하거나 상태가 변경될 때 애플리케이션이 알림을 받을 수 있도록 합니다. (유니코드)
OpenSCManagerA 지정된 컴퓨터에서 서비스 제어 관리자에 대한 연결을 설정하고 지정된 서비스 제어 관리자 데이터베이스를 엽니다. (ANSI)

OpenSCManagerW 지정된 컴퓨터에서 서비스 제어 관리자에 대한 연결을 설정하고 지정된 서비스 제어 관리자 데이터베이스를 엽니다. (유니코드)
OpenServiceA 기존 서비스를 엽니다. (ANSI)
OpenServiceW 기존 서비스를 엽니다. (유니코드)
QueryServiceConfig2A 지정된 서비스의 선택적 구성 매개 변수를 검색합니다. (ANSI)
QueryServiceConfig2W 지정된 서비스의 선택적 구성 매개 변수를 검색합니다. (유니코드)
QueryServiceConfigA 지정된 서비스의 구성 매개 변수를 검색합니다. (ANSI)
QueryServiceConfigW 지정된 서비스의 구성 매개 변수를 검색합니다. (유니코드)
QueryServiceDynamicInformation 현재 서비스 시작과 관련된 동적 정보를 검색합니다.
QueryServiceLockStatusA 지정된 서비스 제어 관리자 데이터베이스의 잠금 상태를 검색합니다. (ANSI)
QueryServiceLockStatusW 지정된 서비스 제어 관리자 데이터베이스의 잠금 상태를 검색합니다. (유니코드)
QueryServiceObjectSecurity 서비스 개체와 연결된 보안 설명자의 복사본을 검색합니다.
QueryServiceStatus 지정된 서비스의 현재 상태를 검색합니다.

QueryServiceStatusEx 지정된 정보 수준에 따라 지정된 서비스의 현재 상태를 검색합니다.
RegisterServiceCtrlHandlerA 서비스 제어 요청을 처리하는 함수를 등록합니다. (ANSI)
RegisterServiceCtrlHandlerExA 확장 서비스 제어 요청을 처리하는 함수를 등록합니다. (ANSI)
RegisterServiceCtrlHandlerExW 확장 서비스 제어 요청을 처리하는 함수를 등록합니다. (유니코드)
RegisterServiceCtrlHandlerW 서비스 제어 요청을 처리하는 함수를 등록합니다. (유니코드)
SetServiceObjectSecurity 서비스 개체의 보안 설명자를 설정합니다.
SetServiceStatus 호출 서비스에 대한 서비스 제어 관리자의 상태 정보를 업데이트.
StartServiceA 서비스를 시작합니다. (ANSI)
StartServiceCtrlDispatcherA 서비스 프로세스의 주 스레드를 서비스 제어 관리자에 연결하여 스레드가 호출 프로세스의 서비스 제어 디스패처 스레드가 됩니다. (ANSI)
StartServiceCtrlDispatcherW 서비스 프로세스의 주 스레드를 서비스 제어 관리자에 연결하여 스레드가 호출 프로세스의 서비스 제어 디스패처 스레드가 됩니다. (유니코드)
StartServiceW 서비스를 시작합니다. (유니코드)
UnlockServiceDatabase 지정된 잠금을 해제하여 서비스 제어 관리자 데이터베이스의 잠금을 해제합니다.

콜백 함수

LPHANDLER_FUNCTION RegisterServiceCtrlHandler 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. 서비스 프로그램은 특정 서비스의 제어 처리기 함수로 사용할 수 있습니다.
LPHANDLER_FUNCTION_EX RegisterServiceCtrlHandlerEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. 서비스 프로그램은 특정 서비스의 제어 처리기 함수로 사용할 수 있습니다.
LPSERVICE_MAIN_FUNCTIONA 서비스의 진입점입니다. (ANSI)
LPSERVICE_MAIN_FUNCTIONW 서비스의 진입점입니다. (유니코드)

구조체

ENUM_SERVICE_STATUS_PROCESSA 서비스 제어 관리자 데이터베이스에 있는 서비스의 이름과 서비스에 대한 정보를 포함합니다. EnumServicesStatusEx 함수에서 사용됩니다. (ANSI)
ENUM_SERVICE_STATUS_PROCESSW 서비스 제어 관리자 데이터베이스에 있는 서비스의 이름과 서비스에 대한 정보를 포함합니다. EnumServicesStatusEx 함수에서 사용됩니다. (유니코드)
ENUM_SERVICE_STATUSA 서비스 제어 관리자 데이터베이스에 있는 서비스의 이름과 해당 서비스에 대한 정보를 포함합니다. EnumDependentServices 및 EnumServicesStatus 함수에서 사용됩니다. (ANSI)
ENUM_SERVICE_STATUSW 서비스 제어 관리자 데이터베이스에 있는 서비스의 이름과 해당 서비스에 대한 정보를 포함합니다. EnumDependentServices 및 EnumServicesStatus 함수에서 사용됩니다. (유니코드)

QUERY_SERVICE_CONFIGA 설치된 서비스에 대한 구성 정보를 포함합니다. QueryServiceConfig 함수에서 사용됩니다. (ANSI)
QUERY_SERVICE_CONFIGW 설치된 서비스에 대한 구성 정보를 포함합니다. QueryServiceConfig 함수에서 사용됩니다. (유니코드)
QUERY_SERVICE_LOCK_STATUSA 서비스 제어 관리자 데이터베이스의 잠금 상태에 대한 정보를 포함합니다. QueryServiceLockStatus 함수에서 사용됩니다. (ANSI)
QUERY_SERVICE_LOCK_STATUSW 서비스 제어 관리자 데이터베이스의 잠금 상태에 대한 정보를 포함합니다. QueryServiceLockStatus 함수에서 사용됩니다. (유니코드)
SC_ACTION 서비스 제어 관리자가 수행할 수 있는 작업을 나타냅니다.
SERVICE_CONTROL_STATUS_REASON_PARAMSA 서비스 제어 매개 변수를 포함합니다. (ANSI)
SERVICE_CONTROL_STATUS_REASON_PARAMSW 서비스 제어 매개 변수를 포함합니다. (유니코드)
SERVICE_DELAYED_AUTO_START_INFO 자동 시작 서비스의 지연된 자동 시작 설정을 포함합니다.
SERVICE_DESCRIPTIONA 서비스 설명을 포함합니다. (ANSI)
SERVICE_DESCRIPTIONW 서비스 설명을 포함합니다. (유니코드)
SERVICE_FAILURE_ACTIONS_FLAG 서비스의 오류 작업 플래그 설정을 포함합니다. 이 설정은 실패 작업을 실행할 시기를 결정합니다.

SERVICE_FAILURE_ACTIONSA 서비스 컨트롤러가 서비스의 각 실패에 대해 수행해야 하는 작업을 나타냅니다. 서비스가 서비스 컨트롤러에 SERVICE_STOPPED 상태를 보고하지 않고 종료될 때 실패한 것으로 간주됩니다. (ANSI)
SERVICE_FAILURE_ACTIONSW 서비스 컨트롤러가 서비스의 각 실패에 대해 수행해야 하는 작업을 나타냅니다. 서비스가 서비스 컨트롤러에 SERVICE_STOPPED 상태를 보고하지 않고 종료될 때 실패한 것으로 간주됩니다. (유니코드)
SERVICE_LAUNCH_PROTECTED_INFO 서비스 보호 유형을 나타냅니다.
SERVICE_NOTIFY_2A 서비스 상태 알림 정보를 나타냅니다. (ANSI)
SERVICE_NOTIFY_2W 서비스 상태 알림 정보를 나타냅니다. (유니코드)
SERVICE_PREFERRED_NODE_INFO 서비스를 실행할 기본 설정 노드를 나타냅니다.
SERVICE_PRESHUTDOWN_INFO preshutdown 설정을 포함합니다.
SERVICE_REQUIRED_PRIVILEGES_INFOA 서비스에 필요한 권한을 나타냅니다. (ANSI)
SERVICE_REQUIRED_PRIVILEGES_INFOW 서비스에 필요한 권한을 나타냅니다. (유니코드)
SERVICE_SID_INFO SID(서비스 보안 식별자)를 나타냅니다.
SERVICE_STATUS 서비스에 대한 상태 정보를 포함합니다.

SERVICE_STATUS_PROCESS 서비스에 대한 프로세스 상태 정보를 포함합니다. ControlServiceEx, EnumServicesStatusEx, NotifyServiceStatusChange 및 QueryServiceStatusEx 함수는 이 구조를 사용합니다.
SERVICE_TABLE_ENTRYA 호출 프로세스에서 실행할 수 있는 서비스에 대한 ServiceMain 함수를 지정합니다. StartServiceCtrlDispatcher 함수에서 사용됩니다. (ANSI)
SERVICE_TABLE_ENTRYW 호출 프로세스에서 실행할 수 있는 서비스에 대한 ServiceMain 함수를 지정합니다. StartServiceCtrlDispatcher 함수에서 사용됩니다. (유니코드)
SERVICE_TIMECHANGE_INFO 시스템 시간 변경 설정을 포함합니다.
SERVICE_TRIGGER 서비스 트리거 이벤트를 나타냅니다. 이 구조체는 SERVICE_TRIGGER_INFO 구조체에서 사용됩니다.
SERVICE_TRIGGER_INFO 서비스에 대한 트리거 이벤트 정보를 포함합니다. 이 구조체는 ChangeServiceConfig2 및 QueryServiceConfig2 함수에서 사용됩니다.
SERVICE_TRIGGER_SPECIFIC_DATA_ITEM 서비스 트리거 이벤트에 대한 트리거 관련 데이터를 포함합니다.

열거형

SERVICE_DIRECTORY_TYPE 서비스별 디렉터리 경로의 형식을 지정합니다.
SERVICE_REGISTRY_STATE_TYPE 서비스 레지스트리 키의 상태 유형을 지정합니다. (SERVICE_REGISTRY_STATE_TYPE)
SERVICE_SHARED_DIRECTORY_TYPE 서비스별 공유 디렉터리 경로의 형식을 지정합니다.

SERVICE_SHARED_REGISTRY_STATE_TYPE 서비스 레지스트리 키의 상태 유형을 지정합니다. (SERVICE_SHARED_REGISTRY_STATE_TYPE)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winsxs.h 헤더

아티클2023. 08. 27.

이 헤더는 애플리케이션 설치 및 서비스에 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 애플리케이션 설치 및 서비스

winsxs.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAssemblyCache IAssemblyCache 인터페이스를 사용하여 병렬 어셈블리를 설치, 제거 또는 쿼리할 수 있습니다. CreateAssemblyCache 함수를 호출하여 IAssemblyCache의 instance 가져옵니다.
IAssemblyCacheItem IAssemblyCacheItem 인터페이스를 사용하여 스트림 기반 설치를 사용하여 병렬 어셈블리를 병렬 저장소에 설치할 수 있습니다.
IAssemblyName IAssemblyName 인터페이스는 병렬 어셈블리 이름을 나타냅니다.

Functions

CreateAssemblyCache CreateAssemblyCache 함수는 IAssemblyCache 인터페이스의 instance 가져옵니다.
CreateAssemblyNameObject CreateAssemblyNameObject 함수는 IAssemblyName 인터페이스의 instance 가져옵니다.

구조체

ASSEMBLY_INFO ASSEMBLY_INFO 구조체에는 병렬 어셈블리 저장소의 어셈블리에 대한 정보가 포함됩니다. 이 정보는 QueryAssemblyInfo 메서드에서 사용됩니다.
FUSION_INSTALL_REFERENCE FUSION_INSTALL_REFERENCE 구조에는 병렬 어셈블리를 참조하는 애플리케이션에 대한 정보가 포함되어 있습니다.

열거형

ASM_CMP_FLAGS ASM_CMP_FLAGS 열거형의 값은 IsEqual 메서드에서 비교할 두 어셈블리 이름의 부분을 지정하는 데 사용됩니다.
ASM_DISPLAY_FLAGS ASM_DISPLAY_FLAGS 열거형의 값은 GetDisplayName 메서드에서 어셈블리 이름의 문자열 표현에 포함할 어셈블리의 전체 이름 부분을 지정하는 데 사용됩니다.
ASM_NAME ASM_NAME 열거형의 값은 side-by-side 어셈블리 이름에 포함된 이름-값 쌍의 속성 ID입니다.
CREATE_ASM_NAME_OBJ_FLAGS CREATE_ASM_NAME_OBJ_FLAGS 열거형은 CreateAssemblyNameObject 함수에서 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winsync.h 헤더

아티클2023. 08. 27.

이 헤더는 Windows 동기화에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 동기화

winsync.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAsynchronousDataRetriever 대상 공급자가 원본 공급자에서 항목 데이터를 비동기적으로 검색하는 메커니즘을 나타냅니다.
IChangeConflict 두 항목 간의 충돌을 나타냅니다.
IChangeUnitException 기술 개체에서 제외할 변경 단위를 나타냅니다.
IChangeUnitListFilterInfo ISyncChangeBatch 개체의 항목에 포함되는 변경 단위를 제어하는 데 사용할 수 있는 필터를 나타냅니다.
IClockVector 기술 구조의 클록 벡터를 나타냅니다.
IClockVectorElement 기술 구조의 클록 벡터 요소를 나타냅니다.
IConstructReplicaKeyMap IReplicaKeyMap 개체에 항목을 추가합니다.
ICoreFragment 특정 변경 단위 집합에 대한 scope 모든 항목에 대한 지식을 나타냅니다.
ICoreFragmentInspector

기술 개체에 포함된 ICoreFragment 개체를 열거합니다.
IDataRetrieverCallback IAsynchronousDataRetriever 메서드에서 처리가 완료되었음을 나타내기 위해 IAsynchronousDataRetriever 개체가 호출할 수 있는 메서드를 나타냅니다.
IEnumChangeUnitExceptions 기술 개체에 저장된 변경 단위 예외를 열거합니다.
IEnumClockVector 클록 벡터에 저장된 클록 벡터 요소를 열거합니다.
IEnumFeedClockVector FeedSync 정보를 포함하는 클록 벡터에 저장된 클록 벡터 요소를 열거합니다.
IEnumRangeExceptions 기술 개체에 저장된 범위 예외를 열거합니다.
IEnumSingleItemExceptions 지식 개체에 저장된 단일 항목 예외를 열거합니다.
IEnumSyncChanges 항목 변경 목록을 열거합니다.
IEnumSyncChangeUnits 변경 단위 목록을 열거합니다.
IFeedClockVector FeedSync 정보를 포함하는 클록 벡터를 나타냅니다.
IFeedClockVectorElement FeedSync 정보를 포함하는 클록 벡터 요소를 나타냅니다.
IFilterRequestCallback 중재는 대상 공급자와 원본 공급자 간의 필터 협상을 필터링합니다.
IForgottenKnowledge 삭제 표시 정리로 인해 잊혀진 지식을 나타냅니다.

IKnowledgeSyncProvider 지식을 사용하여 동기화를 수행하는 동기화 공급자를 나타냅니다.
ILoadChangeContext 항목 저장소에서 로드할 변경 내용에 대한 정보를 나타냅니다.
IProviderConverter 파생 클래스에서 구현되는 경우 는 ISyncProvider 개체를 IKnowledgeSyncProvider 개체로 변환할 수 있는 개체를 나타냅니다.
IRangeException 기술 개체에서 제외할 항목 ID 범위를 나타냅니다.
IRecoverableError 항목이 로드되거나 항목이 저장되었을 때 발생한 복구 가능한 오류를 나타냅니다.
IRecoverableErrorData 복구 가능한 오류에 대한 정보를 나타냅니다.
IReplicaKeyMap 복제본(replica) 키와 복제본(replica) ID 간의 매핑을 나타냅니다.
IRequestFilteredSync 파생 클래스에서 구현되는 경우 는 변경 열거 중에 원본 공급자가 사용할 필터를 지정할 수 있는 대상 공급자를 나타냅니다.
ISingleItemException 기술 개체에서 제외할 항목을 나타냅니다.
ISupportFilteredSync 파생 클래스에서 구현되는 경우 는 필터링된 변경 열거를 지원하고 사용되는 필터 유형을 협상할 수 있는 원본 공급자를 나타냅니다.
ISupportLastWriteTime 항목 또는 변경 단위가 마지막으로 변경된 날짜와 시간을 보고할 수 있는 동기화 공급자를 나타냅니다. 이 기능은 마지막 기록기-승리 충돌 해결을 구현하는 애플리케이션에 유용합니다.

ISyncCallback 애플리케이션에 동기화 이벤트를 알리는 데 사용되는 애플리케이션 콜백을 나타냅니다.
ISyncCallback2 애플리케이션에 동기화 이벤트를 알리는 데 사용되는 추가 애플리케이션 콜백을 나타냅니다.
ISyncChange 항목에 대한 변경 사항을 나타냅니다.
ISyncChangeBatch 변경 내용 집합에 대한 메타데이터를 나타냅니다. (ISyncChangeBatch)
ISyncChangeBatchAdvanced 변경 내용 집합에 대한 추가 정보를 나타냅니다.
ISyncChangeBatchBase 변경 내용 집합에 대한 메타데이터를 나타냅니다. (ISyncChangeBatchBase)
ISyncChangeBatchBase2 ISyncChangeBatchBase 개체의 추가 기능을 나타냅니다.
ISyncChangeBatchWithPrerequisite 변경 일괄 처리와 관련된 필수 구성 요소 지식을 기반으로 하는 변경 일괄 처리에 대한 메타데이터를 나타냅니다.
ISyncChangeBuilder 항목 변경에 대한 추가 데이터를 제공합니다.
ISyncChangeUnit 항목에 포함된 변경 단위의 변경 사항을 나타냅니다.
ISyncChangeWithPrerequisite 변경 내용과 관련된 필수 구성 요소 지식을 기반으로 하는 변경 내용에 대한 메타데이터를 나타냅니다.
ISyncFilterInfo ISyncChangeBatch 개체에 포함된 데이터를 제어하는 데 사용되는 필터에 대한 정보를 나타냅니다.

ISyncFilterInfo2 ISyncChangeBatch 개체에 포함된 변경 내용을 제어하는 데 사용할 수 있는 필터에 대한 추가 정보를 나타냅니다.
ISyncFullEnumerationChange 복구 동기화 중에 ISyncChange 개체에 대한 추가 정보를 나타냅니다.
ISyncFullEnumerationChangeBatch 복구 동기화의 일부로 생성된 변경 내용 집합에 대한 메타데이터를 나타냅니다.
ISynchronousDataRetriever 대상 공급자가 원본 공급자에서 항목 데이터를 검색하는 메커니즘을 나타냅니다.
ISyncKnowledge 복제본(replica) 항목 저장소에 대해 가지고 있는 지식을 나타냅니다.
ISyncKnowledge2 복제본(replica) 항목 저장소에 대한 지식에 대한 추가 정보를 나타냅니다.
ISyncProvider 동기화 세션에서 데이터를 다른 동기화 공급자와 동기화하는 데 사용할 수 있는 동기화 공급자를 나타냅니다.
ISyncSessionExtendedErrorInfo 동기화가 실패하게 된 공급자에 대한 정보를 나타냅니다.
ISyncSessionState 현재 동기화 세션에 대한 정보를 나타냅니다.
ISyncSessionState2 현재 동기화 세션에 대한 추가 정보를 나타냅니다.

구조체

ID_PARAMETER_PAIR 동기화 엔터티 ID의 형식을 나타냅니다.
ID_PARAMETERS 동기화 세션에서 엔터티를 식별하는 데 사용되는 ID 그룹의 형식 스키마를 나타냅니다.
SYNC_RANGE 항목 ID의 범위를 나타냅니다.
SYNC_SESSION_STATISTICS 단일 단방향 동기화 세션에 대한 통계를 나타냅니다.
SYNC_TIME 날짜 및 시간 값을 나타냅니다.
SYNC_VERSION 항목 또는 변경 단위의 버전을 나타냅니다.

열거형

CONFLICT_RESOLUTION_POLICY 동기화 세션에 사용할 동시성 충돌 해결 정책의 옵션을 나타냅니다.
FILTERING_TYPE 필터링된 동기화 중에 변경 일괄 처리에 포함된 정보의 유형을 나타냅니다.
KNOWLEDGE_COOKIE_COMPARISON_RESULT ISyncKnowledge2::CompareToKnowledgeCookie를 사용하여 지식 쿠키를 지식 개체와 비교할 때 가능한 결과를 나타냅니다.
SYNC_FULL_ENUMERATION_ACTION ISyncCallback::OnFullEnumerationNeeded에 대한 응답으로 애플리케이션에서 수행할 작업을 나타냅니다.

SYNC_PROGRESS_STAGE 동기화 세션의 단계를 나타냅니다.
SYNC_PROVIDER_ROLE 동기화 세션의 다른 공급자를 기준으로 공급자의 역할을 나타냅니다.
SYNC_RESOLVE_ACTION 특정 동시성 충돌을 resolve 위해 수행되는 작업을 나타냅니다.
SYNC_SERIALIZATION_VERSION 특정 구성 요소가 호환되는 Microsoft Sync Framework의 버전을 나타냅니다.
SYNC_STATISTICS 구성 요소에 대한 정보를 전달하는 통계 유형을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winternl.h 헤더

아티클2023. 08. 27.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보
- 보안 및 ID
- 시스템 서비스

winternl.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

NtClose 더 이상 사용되지 않습니다. 지정된 핸들을 닫습니다. NtClose는 CloseHandle로 대체됩니다.
NtCreateFile 새 파일 또는 디렉터리를 만들거나 기존 파일, 디바이스, 디렉터리 또는 볼륨을 엽니다.
NtDeviceIoControlFile 더 이상 사용되지 않습니다. 제공된 버퍼에 대한 설명자를 빌드하고 형식화되지 않은 데이터를 파일 핸들과 연결된 디바이스 드라이버에 전달합니다. NtDeviceIoControlFile은 DeviceIoControl로 대체됩니다.
NtNotifyChangeMultipleKeys 레지스트리 키 또는 해당 하위 키가 변경되면 알림을 요청합니다.
NtOpenFile 기존 파일, 디바이스, 디렉터리 또는 볼륨을 열고 파일 개체에 대한 핸들을 반환합니다.
NtQueryInformationProcess 지정된 프로세스에 대한 정보를 검색합니다. (NtQueryInformationProcess)
NtQueryInformationThread 지정된 스레드에 대한 정보를 검색합니다. (NtQueryInformationThread)
NtQueryMultipleValueKey

지정된 다중 값 키에 대한 값을 검색합니다.
NtQueryObject 다양한 종류의 개체 정보를 검색합니다.
NtQuerySystemInformation 지정된 시스템 정보를 검색합니다.
NtQuerySystemTime 현재 시스템 시간을 검색합니다.
NtRenameKey 지정된 레지스트리 키의 이름을 변경합니다. (NtRenameKey)
NtSetInformationKey 지정된 레지스트리 키에 대한 정보를 설정합니다.
NtWaitForSingleObject 더 이상 사용되지 않습니다. 지정된 개체가 신호 상태가 될 때까지 기다립니다. NtWaitForSingleObject는 WaitForSingleObject로 대체됩니다.
RtlAnsiStringToUnicodeString 지정된 ANSI 원본 문자열을 유니코드 문자열로 변환합니다.
RtlCharToInteger 문자열을 정수로 변환합니다.
RtlConvertSidToUnicodeString SID(보안 식별자)를 유니코드 문자 표현으로 변환합니다.
RtlFreeAnsiString RtlUnicodeStringToAnsiString에서 할당한 문자열 버퍼를 해제합니다.
RtlFreeOemString RtlUnicodeStringToOemString에서 할당한 문자열 버퍼를 해제합니다.
RtlFreeUnicodeString

RtlAnsiStringToUnicodeString 또는 RtlUpcaseUnicodeString에서 할당한 문자열 버퍼를 해제합니다.
RtlInitString 계산된 문자열을 초기화합니다.
RtlInitUnicodeString 계산된 유니코드 문자열을 초기화합니다.
RtlIsNameLegalDOS8Dot3 지정된 이름을 사용하여 FAT 파일 시스템에 파일을 만들 수 있는지 여부를 결정합니다.
RtlLocalTimeToSystemTime 지정된 현지 시간을 시스템 시간으로 변환합니다.
RtlNtStatusToDosError 지정된 NTSTATUS 코드를 해당하는 시스템 오류 코드로 변환합니다.
RtlTimeToSecondsSince1970 지정된 64비트 시스템 시간을 1970년 1월 1일 이후의 초 수로 변환합니다.
RtlUnicodeStringToAnsiString 지정된 유니코드 원본 문자열을 ANSI 문자열로 변환합니다.
RtlUnicodeStringToOemString 지정된 유니코드 원본 문자열을 OEM 문자열로 변환합니다. 변환은 OCP(OEM 코드 페이지)와 관련하여 수행됩니다.
RtlUnicodeToMultiByteSize 유니코드 문자열을 ANSI 문자열로 나타내는 데 필요한 바이트 수를 결정합니다.
RtlUniform D.H. Lehmer의 1948년 알고리즘을 사용하여 균일한 난수를 생성합니다.

구조체

Peb 프로세스 정보를 포함합니다.
PEB_LDR_DATA 프로세스에 대해 로드된 모듈에 대한 정보를 포함합니다.
RTL_USER_PROCESS_PARAMETERS 프로세스 매개 변수 정보를 포함합니다.
STRING RtlUnicodeStringToOemString 함수와 함께 사용됩니다.
TEB TEB(스레드 환경 블록)는 스레드의 상태를 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wintrust.h 헤더

아티클2023. 08. 27.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

wintrust.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

OpenPersonalTrustDBDialog 인증서 대화 상자를 표시합니다. (OpenPersonalTrustDBDialog)
OpenPersonalTrustDBDialogEx 인증서 대화 상자를 표시합니다. (OpenPersonalTrustDBDialogEx)
WintrustAddActionID 사용자 시스템에 신뢰 공급자 작업을 추가합니다.
WintrustAddDefaultForUsage 공급자에 대한 기본 사용 식별자 및 콜백 정보를 지정합니다.
WintrustGetDefaultForUsage 기본 사용 식별자 및 콜백 정보를 검색합니다.
WintrustGetRegPolicyFlags 정책 공급자에 대한 정책 플래그를 검색합니다.
WintrustLoadFunctionPointers 지정된 작업 GUID에 대한 함수 진입점을 로드합니다. 이 함수에는 연결된 가져오기 라이브러리가 없습니다.
WintrustRemoveActionID WintrustAddActionID 함수에 의해 추가된 작업을 제거합니다. 이 함수에는 연결된 가져오기 라이브러리가 없습니다.

WintrustSetDefaultIncludePEPageHashes PE 파일에 대한 SIP(주체 인터페이스 패키지) 간접 데이터를 만들 때 페이지 해시가 포함되는지 여부를 결정하는 기본 설정을 설정합니다.
WintrustSetRegPolicyFlags 정책 공급자에 대한 정책 플래그를 설정합니다.
WinVerifyTrust 지정된 개체에 대해 트러스트 확인 작업을 수행합니다.
WinVerifyTrustEx 지정된 개체에 대한 신뢰 확인 작업을 수행하고 WINTRUST_DATA 구조체에 대한 포인터를 사용합니다.
WTHelperCertCheckValidSignature 서명이 유효한지 여부를 확인합니다.
WTHelperCertIsSelfSigned 인증서가 자체 서명되었는지 여부를 확인합니다.
WTHelperGetProvCertFromChain 인증서 체인에서 트러스트 공급자 인증서를 검색합니다.
WTHelperGetProvPrivateDataFromChain 공급자 ID를 사용하여 체인에서 CRYPT_PROVIDER_PRIVDATA 구조를 받습니다.
WTHelperGetProvSignerFromChain 체인에서 인덱스로 서명자 또는 카운터시너를 검색합니다.
WTHelperProvDataFromStateData 지정된 핸들에서 트러스트 공급자 정보를 검색합니다.

구조체

CRYPT_PROVIDER_CERT 공급자 인증서에 대한 정보를 제공합니다.
CRYPT_PROVIDER_DATA WinVerifyTrust와 트러스트 공급자 간에 데이터를 전달하는 데 사용됩니다.
CRYPT_PROVIDER_DEFUSAGE WintrustGetDefaultForUsage 함수에서 공급자의 기본 사용에 대한 콜백 정보를 검색하는 데 사용됩니다.
CRYPT_PROVIDER_FUNCTIONS WinTrust 작업에 대해 CSP(암호화 서비스 공급자)에서 사용하는 함수를 정의합니다.
CRYPT_PROVIDER_PRIVDATA 공급자가 사용할 프라이빗 데이터를 포함합니다.
CRYPT_PROVIDER_REGDEFUSAGE WintrustAddDefaultForUsage 함수에서 공급자의 기본 사용량에 대한 콜백 정보를 등록하는 데 사용됩니다.
CRYPT_PROVIDER_SGNR 서명자 또는 카운터 서명자에 대한 정보를 제공합니다.
CRYPT_PROVIDER_SIGSTATE 정책 공급자와 Wintrust 간에 통신하는 데 사용됩니다.
CRYPT_PROVUI_DATA 공급자에 대한 UI(사용자 인터페이스) 데이터를 제공합니다. 이 구조체는 CRYPT_PROVUI_FUNCS 구조체에서 사용됩니다.
CRYPT_PROVUI_FUNCS 공급자의 UI(사용자 인터페이스) 함수에 대한 정보를 제공합니다. 이 구조체는 CRYPT_PROVIDER_FUNCTIONS 구조체에서 사용됩니다.
CRYPT_REGISTER_ACTIONID 공급자의 함수에 대한 정보를 제공합니다.

CRYPT_TRUST_REG_ENTRY DLL 이름 및 함수 이름으로 공급자 함수를 식별합니다.
SPC_INDIRECT_DATA_CONTENT Authenticode 서명에서 서명된 파일의 다이제스트 및 기타 특성을 저장하는 데 사용됩니다.
WIN_CERTIFICATE 이 구조체는 실행 파일 확인에 사용되는 서명을 캡슐화합니다.
WINTRUST_BLOB_INFO WinVerifyTrust를 호출하여 메모리 BLOB을 확인할 때 사용됩니다.
WINTRUST_CATALOG_INFO WINTRUST_CATALOG_INFO 구조는 WinVerifyTrust를 호출하여 Microsoft 카탈로그의 멤버를 확인할 때 사용됩니다.
WINTRUST_CERT_INFO WinVerifyTrust를 호출하여 CERT_CONTEXT 확인할 때 사용됩니다.
WINTRUST_DATA WinVerifyTrust를 호출하여 필요한 정보를 트러스트 공급자에 전달할 때 사용됩니다.
WINTRUST_FILE_INFO WINTRUST_FILE_INFO 구조는 WinVerifyTrust를 호출하여 개별 파일을 확인할 때 사용됩니다.
WINTRUST_SGNR_INFO WinVerifyTrust를 호출하여 CMSG_SIGNER_INFO 구조를 확인할 때 사용됩니다.
WINTRUST_SIGNATURE_SETTINGS 파일의 서명을 지정하는 데 사용할 수 있습니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winusb.h 헤더

아티클2023. 08. 27.

이 헤더는 USB 드라이버 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- USB 드라이버 참조

winusb.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WinUsb_AbortPipe WinUsb_AbortPipe 함수는 파이프에 대해 보류 중인 모든 전송을 중단합니다. 동기 작업입니다.
WinUsb_ControlTransfer WinUsb_ControlTransfer 함수는 기본 제어 엔드포인트를 통해 제어 데이터를 전송합니다.
WinUsb_FlushPipe WinUsb_FlushPipe 함수는 파이프에 캐시된 모든 데이터를 삭제합니다. 동기 작업입니다.
WinUsb_Free WinUsb_Free 함수는 할당된 모든 리소스를 WinUsb_Initialize 해제합니다. 동기 작업입니다.
WinUsb_GetAdjustedFrameNumber WinUsb_GetAdjustedFrameNumber 함수는 프레임 번호 값 및 타임스탬프를 기반으로 현재 USB 프레임 번호를 계산합니다.
WinUsb_GetAssociatedInterface WinUsb_GetAssociatedInterface 함수는 연결된 인터페이스에 대한 핸들을 검색합니다. 동기 작업입니다.
WinUsb_GetCurrentAlternateSetting WinUsb_GetCurrentAlternateSetting 함수는 인터페이스에 대한 현재 대체 인터페이스 설정을 가져옵니다. 동기 작업입니다.
WinUsb_GetCurrentFrameNumber WinUsb_GetCurrentFrameNumber 함수는 버스의 현재 프레임 번호를 가져옵니다.

WinUsb_GetCurrentFrameNumberAndQpc WinUsb_GetCurrentFrameNumberAndQpc 함수는 프레임 및 마이크로프레임과 동기화된 QPC(시스템 쿼리 성능 카운터) 값을 검색합니다.
WinUsb_GetDescriptor WinUsb_GetDescriptor 함수는 요청된 설명자를 반환합니다. 동기 작업입니다.
WinUsb_GetOverlappedResult WinUsb_GetOverlappedResult 함수는 지정된 파일에서 겹치는 작업의 결과를 검색합니다.
WinUsb_GetPipePolicy WinUsb_GetPipePolicy 함수는 디바이스의 엔드포인트와 연결된 특정 파이프에 대한 정책을 검색합니다. 동기 작업입니다.
WinUsb_GetPowerPolicy WinUsb_GetPowerPolicy 함수는 디바이스에 대한 전원 정책을 검색합니다. 동기 작업입니다.
WinUsb_Initialize WinUsb_Initialize 함수는 파일 핸들로 지정된 디바이스에 대한 WinUSB 핸들을 만듭니다.
WinUsb_QueryDeviceInformation WinUsb_QueryDeviceInformation 함수는 WinUSB 인터페이스 핸들과 연결된 물리적 디바이스에 대한 정보를 가져옵니다.
WinUsb_QueryInterfaceSettings WinUsb_QueryInterfaceSettings 함수는 특정 인터페이스 핸들에 대해 지정된 대체 인터페이스 설정에 대한 인터페이스 설명자를 검색합니다.
WinUsb_QueryPipe WinUsb_QueryPipe 함수는 인터페이스에 대해 지정된 엔드포인트 및 연결된 파이프에 대한 정보를 검색합니다.
WinUsb_QueryPipeEx WinUsb_QueryPipeEx 함수는 인터페이스에 대해 지정된 엔드포인트 및 연결된 파이프에 대한 확장 정보를 검색합니다.
WinUsb_ReadIsochPipe WinUsb_ReadIsochPipe 함수는 등시 OUT 엔드포인트에서 데이터를 읽습니다.

WinUsb_ReadIsochPipeAsap WinUsb_ReadIsochPipeAsap 함수는 등시 OUT 엔드포인트에서 데이터를 읽는 요청을 제출합니다.
WinUsb_ReadPipe WinUsb_ReadPipe 함수는 지정된 파이프에서 데이터를 읽습니다.
WinUsb_RegisterIsochBuffer WinUsb_RegisterIsochBuffer 함수는 등시 전송에 사용할 버퍼를 등록합니다.
WinUsb_ResetPipe WinUsb_ResetPipe 함수는 데이터 토글을 재설정하고 파이프의 중단 조건을 지웁니다.
WinUsb_SetCurrentAlternateSetting WinUsb_SetCurrentAlternateSetting 함수는 인터페이스의 대체 설정을 설정합니다.
WinUsb_SetPipePolicy WinUsb_SetPipePolicy 함수는 디바이스의 엔드포인트와 연결된 특정 파이프에 대한 정책을 설정합니다. 동기 작업입니다.
WinUsb_SetPowerPolicy WinUsb_SetPowerPolicy 함수는 디바이스에 대한 전원 정책을 설정합니다.
WinUsb_StartTrackingForTimeSync WinUsb_StartTrackingForTimeSync 함수는 USB 드라이버 스택에서 USB 버스 프레임 및 마이크로프레임에 대한 연결된 시스템 QPC 시간을 가져오는 시간 동기화 기능을 시작합니다.
WinUsb_StopTrackingForTimeSync WinUsb_StopTrackingForTimeSync 함수는 USB 버스 프레임 및 마이크로프레임에 대한 연결된 시스템 QPC 시간을 가져오는 USB 드라이버 스택의 시간 동기화 기능을 맨 위에 놓습니다.
WinUsb_UnregisterIsochBuffer WinUsb_UnregisterIsochBuffer 함수는 등시 전송에 할당된 WinUsb_RegisterIsochBuffer 모든 리소스를 해제합니다. 동기 작업입니다.
WinUsb_WriteIsochPipe WinUsb_WriteIsochPipe 함수는 지정된 프레임 번호에서 시작하여 호출자가 제공한 버퍼의 내용을 등시 OUT 엔드포인트에 씁니다.

WinUsb_WriteIsochPipeAsap WinUsb_WriteIsochPipeAsap 버퍼의 콘텐츠를 등시 OUT 엔드포인트에 쓰기 위한 요청을 제출합니다.
WinUsb_WritePipe WinUsb_WritePipe 함수는 파이프에 데이터를 씁니다.

구조체

WINUSB_SETUP_PACKET WINUSB_SETUP_PACKET 구조는 USB 설정 패킷을 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winusbio.h 헤더

아티클2023. 03. 08.

이 헤더는 USB 드라이버 참조에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- USB 드라이버 참조

winusbio.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

WINUSB_PIPE_INFORMATION WINUSB_PIPE_INFORMATION 구조에는 WinUsb_QueryPipe 루틴에서 검색하는 파이프 정보가 포함됩니다.
WINUSB_PIPE_INFORMATION_EX WINUSB_PIPE_INFORMATION_EX 구조에는 WinUsb_QueryPipeEx 루틴에서 검색하는 파이프 정보가 포함됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winuser.h 헤더

아티클2023. 03. 08.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 교환
- 데스크톱 창 관리자(DWM)
- 개발자 정보
- 대화 상자
- 디스플레이 디바이스 참조
- 높은 DPI
- 입력 피드백 구성
- 입력 원본 식별
- Windows 애플리케이션에 대한 국제화
- 키보드 및 마우스 입력
- 메뉴 및 기타 리소스
- 모바일 장치 관리 설정 공급자
- 포인터 디바이스 입력 스택
- 포인터 입력 메시지 및 알림
- 원격 데스크톱 서비스
- 보안 및 ID
- 시스템 서비스
- Windows 셸
- 터치 적중 테스트
- 터치 삽입
- 터치 입력
- 창 스테이션 및 데스크톱
- Windows 접근성 기능
- Windows 및 메시지
- Windows 컨트롤
- Windows GDI

winuser.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

ActivateKeyboardLayout 호출 스레드 또는 현재 프로세스에 대한 입력 로캘 식별자(이전의 키보드 레이아웃 핸들)를 설정합니다. 입력 로캘 식별자는 로캘은 물론 키보드의 물리적 레이아웃을 지정합니다.
AddClipboardFormatListener 지정된 창을 시스템 유지 관리 클립보드 형식 수신기 목록에 배치합니다.
AdjustWindowRect 원하는 클라이언트 사각형 크기에 따라 창 사각형의 필요한 크기를 계산합니다. 그런 다음 창 사각형을 CreateWindow 함수에 전달하여 클라이언트 영역이 원하는 크기인 창을 만들 수 있습니다.
AdjustWindowRectEx 클라이언트 사각형의 원하는 크기에 따라 창 사각형의 필요한 크기를 계산합니다. 그런 다음 창 사각형을 CreateWindowEx 함수에 전달하여 클라이언트 영역이 원하는 크기인 창을 만들 수 있습니다.
AdjustWindowRectExForDpi 클라이언트 사각형의 원하는 크기 및 제공된 DPI에 따라 창 사각형의 필요한 크기를 계산합니다.
AllowSetForegroundWindow SetForegroundWindow 함수를 사용하여 지정된 프로세스가 포그라운드 창을 설정할 수 있도록 합니다. 호출 프로세스는 이미 포그라운드 창을 설정할 수 있어야 합니다. 자세한 내용은 이 항목의 뒷부분에 나오는 주의를 참조하십시오.
AnimateWindow 창을 표시하거나 숨길 때 특수 효과를 생성할 수 있습니다. 애니메이션에는 _roll, 슬라이드, 축소 또는 확장, 알파 혼합 페이드의 네 가지 유형이 있습니다.
AnyPopup 소유, 표시, 최상위 팝업 또는 겹치는 창이 화면에 있는지 여부를 나타냅니다. 함수는 호출 애플리케이션의 클라이언트 영역뿐만 아니라 전체 화면을 검색합니다.
AppendMenuA 지정된 메뉴 모음, 드롭다운 메뉴, 하위 메뉴 또는 바로 가기 메뉴의 끝에 새 항목을 추가합니다. 이 함수를 사용하여 메뉴 항목의 콘텐츠, 모양 및 동작을 지정할 수 있습니다. (ANSI)
AppendMenuW 지정된 메뉴 모음, 드롭다운 메뉴, 하위 메뉴 또는 바로 가기 메뉴의 끝에 새 항목을 추가합니다. 이 함수를 사용하여 메뉴 항목의 콘텐츠, 모양 및 동작을 지정할 수 있습니다. (유니코드)

AreDpiAwarenessContextsEqual 두 DPI_AWARENESS_CONTEXT 값이 동일한지 여부를 결정합니다.
ArrangeIconicWindows 지정된 부모 창의 최소화된(상징적인) 자식 창을 모두 정렬합니다.
AttachThreadInput 한 스레드의 입력 처리 메커니즘을 다른 스레드의 입력 처리 메커니즘에 연결하거나 분리합니다.
BeginDeferWindowPos 다중 창 위치 구조에 대한 메모리를 할당하고 구조체에 핸들을 반환합니다.
BeginPaint BeginPaint 함수는 그리기 위해 지정된 창을 준비하고 그림에 대한 정보로 PAINTSTRUCT 구조를 채웁니다.
BlockInput 키보드 및 마우스 입력 이벤트가 애플리케이션에 도달하지 못하도록 차단합니다.
BringWindowToTop 지정된 창을 Z 순서의 맨 위로 가져옵니다. 창이 최상위 창이면 활성화됩니다. 창이 자식 창인 경우 자식 창과 연결된 최상위 부모 창이 활성화됩니다.
BroadcastSystemMessage BroadcastSystemMessage 함수는 지정된 받는 사람에게 메시지를 보냅니다. (BroadcastSystemMessage)
BroadcastSystemMessageA 지정된 받는 사람에게 메시지를 보냅니다. (BroadcastSystemMessageA)
BroadcastSystemMessageExA 지정된 받는 사람에게 메시지를 보냅니다. (BroadcastSystemMessageExA)
BroadcastSystemMessageExW 지정된 받는 사람에게 메시지를 보냅니다. (BroadcastSystemMessageExW)

BroadcastSystemMessageW BroadcastSystemMessageW(유니코드) 함수는 지정된 받는 사람에게 메시지를 보냅니다. (BroadcastSystemMessageW)
CalculatePopupWindowPosition 지정된 앵커 지점, 팝업 창 크기, 플래그 및 선택적 제외 사각형을 사용하여 적절한 팝업 창 위치를 계산합니다.
CallMsgFilterA 지정된 메시지 및 후크 코드를 WH_SYSMSGFILTER 및 WH_MSGFILTER 후크와 연결된 후크 프로시저에 전달합니다. (ANSI)
CallMsgFilterW 지정된 메시지 및 후크 코드를 WH_SYSMSGFILTER 및 WH_MSGFILTER 후크와 연결된 후크 프로시저에 전달합니다. (유니코드)
CallNextHookEx 후크 정보를 현재 후크 체인의 다음 후크 프로시저에 전달합니다. 후크 프로시저는 후크 정보를 처리하기 전이나 후에 이 함수를 호출할 수 있습니다.
CallWindowProcA 지정된 창 프로시저에 메시지 정보를 전달합니다. (ANSI)
CallWindowProcW 지정된 창 프로시저에 메시지 정보를 전달합니다. (유니코드)
CascadeWindows 지정된 부모 창의 지정된 자식 창을 계단식으로 연결합니다.
ChangeClipboardChain 클립보드 뷰어 체인에서 지정된 창을 제거합니다.
ChangeDisplaySettingsA ChangeDisplaySettings 함수는 기본 디스플레이 디바이스의 설정을 지정된 그래픽 모드로 변경합니다. (ANSI)

ChangeDisplaySettingsExA ChangeDisplaySettingsEx 함수는 지정된 디스플레이 디바이스의 설정을 지정된 그래픽 모드로 변경합니다. (ANSI)
ChangeDisplaySettingsExW ChangeDisplaySettingsEx 함수는 지정된 디스플레이 디바이스의 설정을 지정된 그래픽 모드로 변경합니다. (유니코드)
ChangeDisplaySettingsW ChangeDisplaySettings 함수는 기본 디스플레이 디바이스의 설정을 지정된 그래픽 모드로 변경합니다. (유니코드)
ChangeWindowMessageFilter UIPI(사용자 인터페이스 권한 격리) 메시지 필터에서 메시지를 추가하거나 제거합니다.
ChangeWindowMessageFilterEx 지정된 창에 대한 UIPI(사용자 인터페이스 권한 격리) 메시지 필터를 수정합니다.
CharLowerA 문자열 또는 단일 문자를 소문자로 변환합니다. 피연산자가 문자열인 경우 함수는 현재 위치에 있는 문자를 변환합니다. (ANSI)
CharLowerBuffA 버퍼의 대문자를 소문자로 변환합니다. 함수는 현재 위치에 있는 문자를 변환합니다. (ANSI)
CharLowerBuffW 버퍼의 대문자를 소문자로 변환합니다. 함수는 현재 위치에 있는 문자를 변환합니다. (유니코드)
CharLowerW 문자열 또는 단일 문자를 소문자로 변환합니다. 피연산자가 문자열인 경우 함수는 현재 위치에 있는 문자를 변환합니다. (유니코드)
CharNextA 문자열의 다음 문자에 대한 포인터를 검색합니다. 이 함수는 단일 또는 다중 바이트 문자로 구성된 문자열을 처리할 수 있습니다. (ANSI)

CharNextExA 문자열의 다음 문자에 대한 포인터를 검색합니다. 이 함수는 단일 또는 다중 바이트 문자로 구성된 문자열을 처리할 수 있습니다.
CharNextW 문자열의 다음 문자에 대한 포인터를 검색합니다. 이 함수는 단일 또는 다중 바이트 문자로 구성된 문자열을 처리할 수 있습니다. (유니코드)
CharPrevA 문자열에서 이전 문자에 대한 포인터를 검색합니다. 이 함수는 단일 또는 다중 바이트 문자로 구성된 문자열을 처리할 수 있습니다. (ANSI)
CharPrevExA 문자열에서 이전 문자에 대한 포인터를 검색합니다. 이 함수는 단일 또는 다중 바이트 문자로 구성된 문자열을 처리할 수 있습니다.
CharPrevW 문자열에서 이전 문자에 대한 포인터를 검색합니다. 이 함수는 단일 또는 다중 바이트 문자로 구성된 문자열을 처리할 수 있습니다. (유니코드)
CharToOemA 문자열을 OEM 정의 문자 집합으로 변환합니다. 경고 사용하지 마세요. (ANSI)
CharToOemBuffA 문자열에서 지정된 수의 문자를 OEM 정의 문자 집합으로 변환합니다. (ANSI)
CharToOemBuffW 문자열에서 지정된 수의 문자를 OEM 정의 문자 집합으로 변환합니다. (유니코드)
CharToOemW 문자열을 OEM 정의 문자 집합으로 변환합니다. 경고 사용하지 마세요. (유니코드)
CharUpperA 문자열 또는 단일 문자를 대문자로 변환합니다. 피연산자가 문자열인 경우 함수는 현재 위치에 있는 문자를 변환합니다. (ANSI)
CharUpperBuffA 버퍼의 소문자를 대문자로 변환합니다. 함수는 현재 위치에 있는 문자를 변환합니다. (ANSI)

CharUpperBuffW 버퍼의 소문자를 대문자로 변환합니다. 함수는 현재 위치에 있는 문자를 변환합니다. (유니코드)
CharUpperW 문자열 또는 단일 문자를 대문자로 변환합니다. 피연산자가 문자열인 경우 함수는 현재 위치에 있는 문자를 변환합니다. (유니코드)
CheckDlgButton 단추 컨트롤의 확인 상태를 변경합니다.
CheckMenuItem 지정된 메뉴 항목의 검사 표시 특성의 상태를 선택 또는 지우기로 설정합니다.
CheckMenuRadioItem 지정된 메뉴 항목을 확인하고 라디오 항목으로 만듭니다. 동시에 함수는 연결된 그룹의 다른 모든 메뉴 항목을 지우고 해당 항목에 대한 라디오 항목 형식 플래그를 지웁니다.
CheckRadioButton 그룹의 지정된 라디오 단추에 확인 표시를 추가(확인)하고 그룹의 다른 모든 라디오 단추에서 확인 표시를 제거합니다(지우기).
ChildWindowFromPoint 부모 창에 속하는 자식 창 중 지정된 점이 포함된 자식 창이 있는 경우 결정합니다. 검색은 바로 자식 창으로 제한됩니다. 손자, 그리고 더 깊은 하위 창은 검색되지 않습니다.
ChildWindowFromPointEx 지정된 부모 창에 속한 자식 창 중 지정된 점이 포함된 자식 창이 있는 경우 결정합니다.
ClientToScreen ClientToScreen 함수는 지정된 지점의 클라이언트 영역 좌표를 화면 좌표로 변환합니다.
ClipCursor 커서를 화면의 사각형 영역으로 제한합니다.
CloseClipboard 클립보드를 닫습니다.

CloseDesktop 데스크톱 개체에 대한 열린 핸들을 닫습니다.
CloseGestureInfoHandle 제스처 정보 핸들과 연결된 리소스를 닫습니다.
CloseTouchInputHandle 터치 입력 핸들을 닫고, 연결된 프로세스 메모리를 해제하고, 핸들을 무효화합니다.
CloseWindow 지정된 창을 최소화하지만 삭제하지는 않습니다.
CloseWindowStation 열린 창 스테이션 핸들을 닫습니다.
CopyAcceleratorTableA 지정된 가속기 테이블을 복사합니다. 이 함수는 accelerator-table 핸들에 해당하는 accelerator-table 데이터를 가져오거나 accelerator-table 데이터의 크기를 확인하는 데 사용됩니다. (ANSI)
CopyAcceleratorTableW 지정된 가속기 테이블을 복사합니다. 이 함수는 accelerator-table 핸들에 해당하는 accelerator-table 데이터를 가져오거나 accelerator-table 데이터의 크기를 확인하는 데 사용됩니다. (유니코드)
CopyCursor 지정된 커서를 복사합니다.
CopyIcon 지정한 아이콘을 다른 모듈에서 현재 모듈로 복사합니다.
CopyImage 새 이미지(아이콘, 커서 또는 비트맵)를 만들고 지정된 이미지의 특성을 새 이미지에 복사합니다. 필요한 경우 함수는 새 이미지의 원하는 크기에 맞게 비트를 늘립니다.
CopyRect CopyRect 함수는 한 사각형의 좌표를 다른 사각형에 복사합니다.

CountClipboardFormats 현재 클립보드에 있는 다양한 데이터 형식의 수를 검색합니다.
CreateAcceleratorTableA 액셀러레이터 테이블을 만듭니다. (ANSI)
CreateAcceleratorTableW 액셀러레이터 테이블을 만듭니다. (유니코드)
CreateCaret 시스템 caret에 대한 새 셰이프를 만들고 지정된 창에 caret의 소유권을 할당합니다. 캐럿 모양은 선, 블록 또는 비트맵일 수 있습니다.
CreateCursor 지정된 크기, 비트 패턴 및 핫 스폿이 있는 커서를 만듭니다.
CreateDesktopA 새 데스크톱을 만들고, 호출 프로세스의 현재 창 스테이션과 연결하고, 호출 스레드에 할당합니다. (ANSI)
CreateDesktopExA 지정된 힙을 사용하여 새 데스크톱을 만들고, 호출 프로세스의 현재 창 스테이션에 연결하고, 호출 스레드에 할당합니다. (ANSI)
CreateDesktopExW 지정된 힙을 사용하여 새 데스크톱을 만들고, 호출 프로세스의 현재 창 스테이션에 연결하고, 호출 스레드에 할당합니다. (유니코드)
CreateDesktopW 새 데스크톱을 만들고, 호출 프로세스의 현재 창 스테이션과 연결하고, 호출 스레드에 할당합니다. (유니코드)
CreateDialogA 대화 상자 템플릿 리소스에서 모덜리스 대화 상자를 만듭니다. CreateDialog 매크로는 CreateDialogParam 함수를 사용합니다. (ANSI)

CreateDialogIndirectA 메모리의 대화 상자 템플릿에서 모달리스 대화 상자를 만듭니다. CreateDialogIndirect 매크로는 CreateDialogIndirectParam 함수를 사용합니다. (ANSI)
CreateDialogIndirectParamA 메모리의 대화 상자 템플릿에서 모달리스 대화 상자를 만듭니다. (ANSI)
CreateDialogIndirectParamW 메모리의 대화 상자 템플릿에서 모달리스 대화 상자를 만듭니다. (유니코드)
CreateDialogIndirectW 메모리의 대화 상자 템플릿에서 모달리스 대화 상자를 만듭니다. CreateDialogIndirect 매크로는 CreateDialogIndirectParam 함수를 사용합니다. (유니코드)
CreateDialogParamA 대화 상자 템플릿 리소스에서 모달리스 대화 상자를 만듭니다. (ANSI)
CreateDialogParamW 대화 상자 템플릿 리소스에서 모달리스 대화 상자를 만듭니다. (유니코드)
CreateDialogW 대화 상자 템플릿 리소스에서 모달리스 대화 상자를 만듭니다. CreateDialog 매크로는 CreateDialogParam 함수를 사용합니다. (유니코드)
CreateIcon 지정된 크기, 색 및 비트 패턴이 있는 아이콘을 만듭니다.
CreateIconFromResource 아이콘을 설명하는 리소스 비트에서 아이콘 또는 커서를 만듭니다. (CreateIconFromResource)
CreateIconFromResourceEx 아이콘을 설명하는 리소스 비트에서 아이콘 또는 커서를 만듭니다. (CreateIconFromResourceEx)
CreateIconIndirect ICONINFO 구조체에서 아이콘 또는 커서를 만듭니다.

CreateMDIWindowA MDI(다중 문서 인터페이스) 자식 창을 만듭니다. (ANSI)
CreateMDIWindowW MDI(다중 문서 인터페이스) 자식 창을 만듭니다. (유니코드)
CreateMenu 메뉴를 만듭니다. 메뉴는 처음에는 비어 있지만 InsertMenuItem, AppendMenu 및 InsertMenu 함수를 사용하여 메뉴 항목으로 채울 수 있습니다.
CreatePopupMenu 드롭다운 메뉴, 하위 메뉴 또는 바로 가기 메뉴를 만듭니다.
CreateSyntheticPointerDevice 호출 애플리케이션에 대한 포인터 삽입 디바이스를 구성하고 앱이 삽입할 수 있는 최대 동시 포인터 수를 초기화합니다.
CreateWindowA 겹치는 팝업 또는 자식 창을 만듭니다. (ANSI)
CreateWindowExA 확장 창 스타일을 사용하여 겹치는, 팝업 또는 자식 창을 만듭니다. 그렇지 않으면 이 함수는 CreateWindow 함수와 동일합니다. (ANSI)
CreateWindowExW 확장 창 스타일을 사용하여 겹치는, 팝업 또는 자식 창을 만듭니다. 그렇지 않으면 이 함수는 CreateWindow 함수와 동일합니다. (유니코드)
CreateWindowStationA 창 스테이션 개체를 만들고 호출 프로세스와 연결한 다음 현재 세션에 할당합니다. (ANSI)
CreateWindowStationW 창 스테이션 개체를 만들고 호출 프로세스와 연결한 다음 현재 세션에 할당합니다. (유니코드)
CreateWindowW 겹치는 팝업 또는 자식 창을 만듭니다. (유니코드)

DefDlgProcA 기본 대화 상자 창 프로시저를 호출하여 프라이빗 창 클래스가 있는 대화 상자가 처리하지 않는 모든 창 메시지에 대한 기본 처리를 제공합니다. (ANSI)
DefDlgProcW 기본 대화 상자 창 프로시저를 호출하여 프라이빗 창 클래스가 있는 대화 상자가 처리하지 않는 모든 창 메시지에 대한 기본 처리를 제공합니다. (유니코드)
DeferWindowPos 지정된 창에 대해 지정된 다중 창 위치 구조를 업데이트.
DefFrameProcA MDI(다중 문서 인터페이스) 프레임 창의 창 프로시저가 처리되지 않는 모든 창 메시지에 대한 기본 처리를 제공합니다. (ANSI)
DefFrameProcW MDI(다중 문서 인터페이스) 프레임 창의 창 프로시저가 처리되지 않는 모든 창 메시지에 대한 기본 처리를 제공합니다. (유니코드)
DefMDIChildProcA MDI(다중 문서 인터페이스) 자식 창의 창 프로시저가 처리되지 않는 모든 창 메시지에 대한 기본 처리를 제공합니다. (ANSI)
DefMDIChildProcW MDI(다중 문서 인터페이스) 자식 창의 창 프로시저가 처리되지 않는 모든 창 메시지에 대한 기본 처리를 제공합니다. (유니코드)
DefRawInputProc RAWINPUTHEADER 구조체의 크기가 올바른지 확인합니다.
DefWindowProcA 기본 창 프로시저를 호출하여 애플리케이션이 처리하지 않는 모든 창 메시지에 대한 기본 처리를 제공합니다. (ANSI)
DefWindowProcW 기본 창 프로시저를 호출하여 애플리케이션이 처리하지 않는 모든 창 메시지에 대한 기본 처리를 제공합니다. (유니코드)

DeleteMenu 지정된 메뉴에서 항목을 삭제합니다. 메뉴 항목이 메뉴 또는 하위 메뉴를 열면 이 함수는 메뉴 또는 하위 메뉴에 대한 핸들을 삭제하고 메뉴 또는 하위 메뉴에서 사용하는 메모리를 해제합니다.
DeregisterShellHookWindow 셸 후크 메시지를 수신하도록 등록된 지정된 셸 창의 등록을 취소합니다.
DestroyAcceleratorTable 가속기 테이블을 삭제합니다.
DestroyCaret 캐리트의 현재 모양을 제거하고 창에서 캐리트를 해제하고 화면에서 캐리트를 제거합니다.
DestroyCursor 커서를 제거하고 커서가 점유한 메모리를 해제합니다. 이 함수를 사용하여 공유 커서를 삭제하지 마세요.
DestroyIcon 아이콘을 삭제하고 아이콘이 점유한 메모리를 해제합니다.
DestroyMenu 지정된 메뉴를 삭제하고 메뉴가 차지하는 모든 메모리를 해제합니다.
DestroySyntheticPointerDevice 지정된 포인터 삽입 디바이스를 삭제합니다.
DestroyWindow 지정된 창을 삭제합니다.
DialogBoxA 대화 상자 템플릿 리소스에서 모달 대화 상자를 만듭니다. DialogBox는 지정된 콜백 함수가 EndDialog 함수를 호출하여 모달 대화 상자를 종료할 때까지 컨트롤을 반환하지 않습니다. (ANSI)
DialogBoxIndirectA 메모리의 대화 상자 템플릿에서 모달 대화 상자를 만듭니다. DialogBoxIndirect는 지정된 콜백 함수가 EndDialog 함수를 호출하여 모달 대화 상자를 종료할 때까지 컨트롤을 반환하지 않습니다. (ANSI)

DialogBoxIndirectParamA 메모리의 대화 상자 템플릿에서 모달 대화 상자를 만듭니다. (ANSI)
DialogBoxIndirectParamW 메모리의 대화 상자 템플릿에서 모달 대화 상자를 만듭니다. (유니코드)
DialogBoxIndirectW 메모리의 대화 상자 템플릿에서 모달 대화 상자를 만듭니다. DialogBoxIndirect는 지정된 콜백 함수가 EndDialog 함수를 호출하여 모달 대화 상자를 종료할 때까지 컨트롤을 반환하지 않습니다. (유니코드)
DialogBoxParamA 대화 상자 템플릿 리소스에서 모달 대화 상자를 만듭니다. (ANSI)
DialogBoxParamW 대화 상자 템플릿 리소스에서 모달 대화 상자를 만듭니다. (유니코드)
DialogBoxW 대화 상자 템플릿 리소스에서 모달 대화 상자를 만듭니다. DialogBox는 지정된 콜백 함수가 EndDialog 함수를 호출하여 모달 대화 상자를 종료할 때까지 컨트롤을 반환하지 않습니다. (유니코드)
DisableProcessWindowsGhosting 호출 GUI 프로세스에 대한 창 고스팅 기능을 사용하지 않도록 설정합니다. 창 고스팅은 사용자가 응답하지 않는 애플리케이션의 기본 창을 최소화, 이동 또는 닫을 수 있는 Windows 관리자 기능입니다.
DispatchMessage DispatchMessage 함수는 창 프로시저에 메시지를 디스패치합니다. 일반적으로 GetMessage 함수에서 검색한 메시지를 디스패치하는 데 사용됩니다.
DispatchMessageA 창 프로시저에 메시지를 디스패치합니다. 일반적으로 GetMessage 함수에서 검색한 메시지를 디스패치하는 데 사용됩니다. (DispatchMessageA)
DispatchMessageW DispatchMessageW(유니코드) 함수는 창 프로시저에 메시지를 디스패치합니다. 일반적으로 GetMessage 함수에서 검색한 메시지를 디스패치하는 데 사용됩니다.

DisplayConfigGetDeviceInfo DisplayConfigGetDeviceInfo 함수는 디바이스에 대한 디스플레이 구성 정보를 검색합니다.
DisplayConfigSetDeviceInfo DisplayConfigSetDeviceInfo 함수는 대상의 속성을 설정합니다.
DlgDirListA 콤보 상자의 콘텐츠를 지정된 디렉터리의 하위 디렉터리 및 파일 이름으로 바꿉니다. 파일 특성 집합을 지정하여 이름 목록을 필터링할 수 있습니다. 목록에는 필요에 따라 매핑된 드라이브가 포함될 수 있습니다. (ANSI)
DlgDirListComboBoxA 콤보 상자의 콘텐츠를 지정된 디렉터리의 하위 디렉터리 및 파일 이름으로 바꿉니다. 파일 특성 집합을 지정하여 이름 목록을 필터링할 수 있습니다. 이름 목록에는 매핑된 드라이브 문자가 포함될 수 있습니다. (ANSI)
DlgDirListComboBoxW 콤보 상자의 콘텐츠를 지정된 디렉터리의 하위 디렉터리 및 파일 이름으로 바꿉니다. 파일 특성 집합을 지정하여 이름 목록을 필터링할 수 있습니다. 이름 목록에는 매핑된 드라이브 문자가 포함될 수 있습니다. (유니코드)
DlgDirListW 콤보 상자의 콘텐츠를 지정된 디렉터리의 하위 디렉터리 및 파일 이름으로 바꿉니다. 파일 특성 집합을 지정하여 이름 목록을 필터링할 수 있습니다. 목록에는 필요에 따라 매핑된 드라이브가 포함될 수 있습니다. (유니코드)
DlgDirSelectComboBoxExA DlgDirListComboBox 함수를 사용하여 채워진 콤보 상자에서 현재 선택 항목을 검색합니다. 선택 항목은 드라이브 문자, 파일 또는 디렉터리 이름으로 해석됩니다. (ANSI)
DlgDirSelectComboBoxExW DlgDirListComboBox 함수를 사용하여 채워진 콤보 상자에서 현재 선택 항목을 검색합니다. 선택 항목은 드라이브 문자, 파일 또는 디렉터리 이름으로 해석됩니다. (유니코드)
DlgDirSelectExA 단일 선택 목록 상자에서 현재 선택 영역을 검색합니다. 목록 상자가 DlgDirList 함수에 의해 채워졌으며 선택 영역이 드라이브 문자, 파일 이름 또는 디렉터리 이름이라고 가정합니다. (ANSI)

DlgDirSelectExW 단일 선택 목록 상자에서 현재 선택 영역을 검색합니다. 목록 상자가 DlgDirList 함수에 의해 채워졌으며 선택 영역이 드라이브 문자, 파일 이름 또는 디렉터리 이름이라고 가정합니다. (유니코드)
DragDetect 마우스를 캡처하고 사용자가 왼쪽 단추를 놓거나, Esc 키를 누르거나, 지정된 점 주위의 끌기 사각형 밖으로 마우스를 이동할 때까지 이동을 추적합니다.
DrawAnimatedRects 창의 캡션 애니메이션을 적용하여 아이콘의 열기 또는 창 최소화 또는 최대화를 나타냅니다.
DrawCaption DrawCaption 함수는 창 캡션 그립니다.
DrawEdge DrawEdge 함수는 하나 이상의 사각형 가장자리를 그립니다.
DrawFocusRect DrawFocusRect 함수는 사각형에 포커스가 있음을 나타내는 데 사용되는 스타일로 사각형을 그립니다.
DrawFrameControl DrawFrameControl 함수는 지정된 형식과 스타일의 프레임 컨트롤을 그립니다.
DrawIcon 지정된 디바이스 컨텍스트에 아이콘 또는 커서를 그립니다.
DrawIconEx 지정된 디바이스 컨텍스트에 아이콘 또는 커서를 그리고 지정된 래스터 작업을 수행하고 지정된 대로 아이콘 또는 커서를 확장하거나 압축합니다.
DrawMenuBar 지정된 창의 메뉴 모음을 다시 그립니다. 시스템에서 창을 만든 후 메뉴 모음이 변경되면 변경된 메뉴 모음을 그리려면 이 함수를 호출해야 합니다.
DrawStateA DrawState 함수는 이미지를 표시하고 비활성 상태 또는 기본 상태와 같은 상태를 나타내기 위해 시각적 효과를 적용합니다. (ANSI)

DrawStateW DrawState 함수는 이미지를 표시하고 비활성 상태 또는 기본 상태와 같은 상태를 나타내기 위해 시각적 효과를 적용합니다. (유니코드)
Drawtext DrawText 함수는 지정된 사각형에 서식이 지정된 텍스트를 그립니다. (DrawText 함수)
DrawTextA DrawText 함수는 지정된 사각형에 서식이 지정된 텍스트를 그립니다. 지정된 메서드(탭 확장, 문자 양쪽 맞춤, 줄 바꿈 등)에 따라 텍스트의 서식을 지정합니다. (DrawTextA)
DrawTextExA DrawTextEx 함수는 지정된 사각형에 서식이 지정된 텍스트를 그립니다. (ANSI)
DrawTextExW DrawTextEx 함수는 지정된 사각형에 서식이 지정된 텍스트를 그립니다. (유니코드)
DrawTextW DrawTextW(유니코드) 함수는 지정된 사각형에 서식이 지정된 텍스트를 그립니다. (DrawTextW 함수)
EmptyClipboard 클립보드를 비우고 클립보드의 데이터에 대한 핸들을 해제합니다. 그런 다음 함수는 클립보드가 열려 있는 창에 클립보드의 소유권을 할당합니다.
EnableMenuItem 지정된 메뉴 항목을 사용하거나 사용하지 않도록 설정하거나 회색으로 표시합니다.
EnableMouseInPointer 마우스가 포인터 입력 디바이스 역할을 하고 WM_POINTER 메시지를 보낼 수 있도록 합니다.
EnableNonClientDpiScaling 높은 DPI 디스플레이에서 는 지정된 최상위 창의 비 클라이언트 영역 부분의 자동 디스플레이 크기 조정을 사용하도록 설정합니다. 해당 창을 초기화하는 동안 호출해야 합니다.
EnableScrollBar EnableScrollBar 함수는 하나 또는 두 스크롤 막대 화살표를 사용하거나 사용하지 않도록 설정합니다.

EnableWindow 지정된 창 또는 컨트롤에 대한 마우스 및 키보드 입력이 가능하거나 가능하지 않도록 설정합니다. 입력이 가능하지 않으면 창은 입력(예: 마우스 클릭 및 키 누르기)을 받지 않습니다. 입력이 가능하게 설정되면 창은 모든 입력을 받습니다.
EndDeferWindowPos 단일 화면 새로 고침 주기에서 하나 이상의 창의 위치와 크기를 동시에 업데이트합니다.
EndDialog 모달 대화 상자를 제거하여 시스템이 대화 상자에 대한 처리를 종료합니다.
EndMenu 호출 스레드의 활성 메뉴를 종료합니다.
EndPaint EndPaint 함수는 지정된 창에서 그리기 끝을 표시합니다. 이 함수는 BeginPaint 함수를 호출할 때마다 필요하지만 그리기는 완료된 후에만 필요합니다.
EndTask 지정된 창을 강제로 닫습니다.
EnumChildWindows 각 자식 창에 핸들을 차례로 애플리케이션 정의 콜백 함수에 전달하여 지정된 부모 창에 속하는 자식 창을 열거합니다.
EnumClipboardFormats 클립보드에서 현재 사용할 수 있는 데이터 형식을 열거합니다.
EnumDesktopsA 호출 프로세스의 지정된 창 스테이션과 연결된 모든 데스크톱을 열거합니다. 함수는 각 데스크톱의 이름을 애플리케이션 정의 콜백 함수에 차례로 전달합니다. (ANSI)
EnumDesktopsW 호출 프로세스의 지정된 창 스테이션과 연결된 모든 데스크톱을 열거합니다. 함수는 각 데스크톱의 이름을 애플리케이션 정의 콜백 함수에 차례로 전달합니다. (유니코드)

EnumDesktopWindows 지정된 데스크톱과 연결된 모든 최상위 창을 열거합니다. 핸들을 각 창에 차례로 애플리케이션 정의 콜백 함수로 전달합니다.
EnumDisplayDevicesA EnumDisplayDevices 함수를 사용하면 현재 세션의 디스플레이 디바이스에 대한 정보를 가져올 수 있습니다. (ANSI)
EnumDisplayDevicesW EnumDisplayDevices 함수를 사용하면 현재 세션의 디스플레이 디바이스에 대한 정보를 가져올 수 있습니다. (유니코드)
EnumDisplayMonitors EnumDisplayMonitors 함수는 지정된 클리핑 사각형의 교집합 및 디바이스 컨텍스트의 표시 영역에 의해 형성된 영역을 교차하는 디스플레이 모니터(미러링 드라이버와 연결된 보이지 않는 의사 모니터 포함)를 열거합니다. EnumDisplayMonitors는 열거된 각 모니터에 대해 애플리케이션 정의 MonitorEnumProc 콜백 함수를 한 번 호출합니다. GetSystemMetrics(SM_CMONITORS)는 디스플레이 모니터만 계산합니다.
EnumDisplaySettingsA EnumDisplaySettings 함수는 디스플레이 디바이스의 그래픽 모드 중 하나에 대한 정보를 검색합니다. 디스플레이 디바이스의 모든 그래픽 모드에 대한 정보를 검색하려면 이 함수를 일련의 호출합니다. (ANSI)
EnumDisplaySettingsExA EnumDisplaySettingsEx 함수는 디스플레이 디바이스의 그래픽 모드 중 하나에 대한 정보를 검색합니다. 디스플레이 디바이스의 모든 그래픽 모드에 대한 정보를 검색하려면 이 함수를 일련의 호출합니다. (ANSI)
EnumDisplaySettingsExW EnumDisplaySettingsEx 함수는 디스플레이 디바이스의 그래픽 모드 중 하나에 대한 정보를 검색합니다. 디스플레이 디바이스의 모든 그래픽 모드에 대한 정보를 검색하려면 이 함수를 일련의 호출합니다. (유니코드)
EnumDisplaySettingsW EnumDisplaySettings 함수는 디스플레이 디바이스의 그래픽 모드 중 하나에 대한 정보를 검색합니다. 디스플레이 디바이스의 모든 그래픽 모드에 대한 정보를 검색하려면 이 함수를 일련의 호출합니다. (유니코드)

EnumPropsA 지정된 콜백 함수에 하나씩 전달하여 창의 속성 목록에 있는 모든 항목을 열거합니다. EnumProps는 마지막 항목이 열거되거나 콜백 함수가 FALSE를 반환할 때까지 계속됩니다. (ANSI)
EnumPropsExA 지정된 콜백 함수에 하나씩 전달하여 창의 속성 목록에 있는 모든 항목을 열거합니다. EnumPropsEx는 마지막 항목이 열거되거나 콜백 함수가 FALSE를 반환할 때까지 계속됩니다. (ANSI)
EnumPropsExW 지정된 콜백 함수에 하나씩 전달하여 창의 속성 목록에 있는 모든 항목을 열거합니다. EnumPropsEx는 마지막 항목이 열거되거나 콜백 함수가 FALSE를 반환할 때까지 계속됩니다. (유니코드)
EnumPropsW 지정된 콜백 함수에 하나씩 전달하여 창의 속성 목록에 있는 모든 항목을 열거합니다. EnumProps는 마지막 항목이 열거되거나 콜백 함수가 FALSE를 반환할 때까지 계속됩니다. (유니코드)
EnumThreadWindows 각 창에 핸들을 차례로 애플리케이션 정의 콜백 함수에 전달하여 스레드와 연결된 모든 비차일드 창을 열거합니다.
EnumWindows 각 창에 핸들을 차례로 애플리케이션 정의 콜백 함수에 전달하여 화면의 모든 최상위 창을 열거합니다. EnumWindows는 마지막 최상위 창이 열거되거나 콜백 함수가 FALSE를 반환할 때까지 계속됩니다.
EnumWindowStationsA 현재 세션의 모든 창 스테이션을 열거합니다. 함수는 각 창 스테이션의 이름을 차례로 애플리케이션 정의 콜백 함수에 전달합니다. (ANSI)
EnumWindowStationsW 현재 세션의 모든 창 스테이션을 열거합니다. 함수는 각 창 스테이션의 이름을 차례로 애플리케이션 정의 콜백 함수에 전달합니다. (유니코드)
EqualRect EqualRect 함수는 지정된 두 사각형이 왼쪽 위와 오른쪽 아래 모서리의 좌표를 비교하여 같은지 여부를 결정합니다.

EvaluateProximityToPolygon 다각형의 점수를 가능한 터치 대상(터치 접촉 영역을 교차하는 다른 모든 다각형과 비교) 및 다각형 내의 조정된 터치 지점으로 반환합니다.
EvaluateProximityToRect 터치 접촉 영역을 교차하는 다른 모든 사각형과 사각형 내의 조정된 터치 지점에 비해 가능한 터치 대상으로 사각형의 점수를 반환합니다.
ExcludeUpdateRgn ExcludeUpdateRgn 함수는 창의 업데이트된 영역을 클리핑 영역에서 제외하여 창의 잘못된 영역 내에서 그리는 것을 방지합니다.
ExitWindows ExitWindowsEx 함수를 호출하여 대화형 사용자를 로그오프합니다.
ExitWindowsEx 대화형 사용자를 로그오프하거나, 시스템을 종료하거나, 시스템을 종료하고 다시 시작합니다.
FillRect FillRect 함수는 지정된 브러시를 사용하여 사각형을 채웁니다. 이 함수는 왼쪽 및 위쪽 테두리를 포함하지만 사각형의 오른쪽 및 아래쪽 테두리는 제외합니다.
FindWindowA 클래스 이름과 창 이름이 지정된 문자열과 일치하는 최상위 창에 대한 핸들을 검색합니다. 이 함수는 자식 창을 검색하지 않습니다. 이 함수는 대/소문자를 구분하는 검색을 수행하지 않습니다. (ANSI)
FindWindowExA 클래스 이름과 창 이름이 지정된 문자열과 일치하는 창에 대한 핸들을 검색합니다. 함수는 지정된 자식 창 다음에 있는 자식 창부터 시작하여 자식 창을 검색합니다. 이 함수는 대/소문자를 구분하는 검색을 수행하지 않습니다. (ANSI)
FindWindowExW 클래스 이름과 창 이름이 지정된 문자열과 일치하는 창에 대한 핸들을 검색합니다. 함수는 지정된 자식 창 다음에 있는 자식 창부터 시작하여 자식 창을 검색합니다. 이 함수는 대/소문자를 구분하는 검색을 수행하지 않습니다. (유니코드)

FindWindowW 클래스 이름과 창 이름이 지정된 문자열과 일치하는 최상위 창에 대한 핸들을 검색합니다. 이 함수는 자식 창을 검색하지 않습니다. 이 함수는 대/소문자를 구분하는 검색을 수행하지 않습니다. (유니코드)
FlashWindow 지정된 창을 한 번 깜박입니다. 창의 활성 상태는 변경되지 않습니다.
FlashWindowEx 지정된 창을 깜박입니다. 창의 활성 상태는 변경되지 않습니다.
FrameRect FrameRect 함수는 지정된 브러시를 사용하여 지정된 사각형 주위에 테두리를 그립니다. 테두리의 너비와 높이는 항상 하나의 논리 단위입니다.
GET_APPCOMMAND_LPARAM 지정된 LPARAM 값에서 애플리케이션 명령을 검색합니다.
GET_DEVICE_LPARAM 지정된 LPARAM 값에서 입력 디바이스 유형을 검색합니다.
GET_FLAGS_LPARAM 지정된 LPARAM 값에서 특정 가상 키의 상태를 검색합니다. (GET_FLAGS_LPARAM)
GET_KEYSTATE_LPARAM 지정된 LPARAM 값에서 특정 가상 키의 상태를 검색합니다. (GET_KEYSTATE_LPARAM)
GET_KEYSTATE_WPARAM 지정된 WPARAM 값에서 특정 가상 키의 상태를 검색합니다.
GET_NCHITTEST_WPARAM 지정된 WPARAM 값에서 적중 테스트 값을 검색합니다.
GET_POINTERID_WPARAM 지정된 값을 사용하여 포인터 ID를 검색합니다.

GET_RAWINPUT_CODE_WPARAM WM_INPUT wParam에서 입력 코드를 검색합니다.
GET_WHEEL_DELTA_WPARAM 지정된 WPARAM 값에서 휠 델타 값을 검색합니다.
GET_XBUTTON_WPARAM 지정된 WPARAM 값에서 특정 단추의 상태를 검색합니다.
GetActiveWindow 호출 스레드의 메시지 큐에 연결된 활성 창에 대한 창 핸들을 가져옵니다.
GetAltTabInfoA 지정된 창이 애플리케이션 전환(ALT+TAB) 창인 경우 지정된 창에 대한 상태 정보를 검색합니다. (ANSI)
GetAltTabInfoW 지정된 창이 애플리케이션 전환(ALT+TAB) 창인 경우 지정된 창에 대한 상태 정보를 검색합니다. (유니코드)
GetAncestor 지정된 창의 상위 항목에 대한 핸들을 검색합니다.
GetAsyncKeyState 함수가 호출될 때 키가 위쪽 또는 아래쪽인지 여부와 GetAsyncKeyState에 대한 이전 호출 후에 키가 눌렸는지 여부를 결정합니다.
GetAutoRotationState 시스템의 화면 자동 회전 상태(예: 자동 회전 지원 여부 및 사용자가 사용하도록 설정되었는지 여부)가 포함된 AR_STATE 값을 검색합니다.
GetAwarenessFromDpiAwarenessContext DPI_AWARENESS_CONTEXT DPI_AWARENESS 값을 검색합니다.
GetCapture 마우스를 캡처한 창에 대한 핸들(있는 경우)을 가져옵니다. 한 번에 하나의 창만 마우스를 캡처할 수 있습니다. 이 창은 커서가 테두리 내에 있는지 여부에 관계없이 마우스 입력을 받습니다.

GetCaretBlinkTime caret의 픽셀을 반전하는 데 필요한 시간을 검색합니다. 사용자는 이 값을 설정할 수 있습니다.
GetCaretPos 지정한 POINT 구조체에 캐리트의 위치를 복사합니다.
GetCIMSSM 입력 메시지의 원본을 검색합니다(GetCurrentInputMessageSourceInSendMessage).
GetClassInfoA 창 클래스에 대한 정보를 검색합니다. (ANSI)
GetClassInfoExA 창 클래스와 연결된 작은 아이콘에 대한 핸들을 포함하여 창 클래스에 대한 정보를 검색합니다. GetClassInfo 함수는 작은 아이콘에 대한 핸들을 검색하지 않습니다. (ANSI)
GetClassInfoExW 창 클래스와 연결된 작은 아이콘에 대한 핸들을 포함하여 창 클래스에 대한 정보를 검색합니다. GetClassInfo 함수는 작은 아이콘에 대한 핸들을 검색하지 않습니다. (유니코드)
GetClassInfoW 창 클래스에 대한 정보를 검색합니다. (유니코드)
GetClassLongA 지정된 창과 연결된 WNDCLASSEX 구조체에서 지정된 32비트(DWORD) 값을 검색합니다. (ANSI)
GetClassLongPtrA 지정된 창과 연결된 WNDCLASSEX 구조체에서 지정된 값을 검색합니다. (ANSI)
GetClassLongPtrW 지정된 창과 연결된 WNDCLASSEX 구조체에서 지정된 값을 검색합니다. (유니코드)
GetClassLongW 지정된 창과 연결된 WNDCLASSEX 구조체에서 지정된 32비트(DWORD) 값을 검색합니다. (유니코드)

GetClassName GetClassName 함수는 지정된 창이 속한 클래스의 이름을 검색합니다. (GetClassName)
GetClassNameA 지정된 창이 속한 클래스의 이름을 검색합니다. (GetClassNameA)
GetClassNameW GetClassNameW(유니코드) 함수는 지정된 창이 속한 클래스의 이름을 검색합니다. (GetClassNameW)
GetClassWord 지정된 오프셋의 16비트(WORD) 값을 지정된 창이 속한 창 클래스의 추가 클래스 메모리로 검색합니다.
GetClientRect 창 클라이언트 영역의 좌표를 검색합니다.
GetClipboardData 클립보드에서 지정된 형식으로 데이터를 검색합니다. 클립보드가 이전에 열렸어야 합니다.
GetClipboardFormatNameA 클립보드에서 지정된 등록된 형식의 이름을 검색합니다. 함수는 지정된 버퍼에 이름을 복사합니다. (ANSI)
GetClipboardFormatNameW 클립보드에서 지정된 등록된 형식의 이름을 검색합니다. 함수는 지정된 버퍼에 이름을 복사합니다. (유니코드)
GetClipboardOwner 클립보드의 현재 소유자 창 핸들을 검색합니다.
GetClipboardSequenceNumber 현재 창 스테이션의 클립보드 시퀀스 번호를 검색합니다.
GetClipboardViewer 클립보드 뷰어 체인의 첫 번째 창에 대한 핸들을 검색합니다.

GetClipCursor 커서가 제한된 사각형 영역의 화면 좌표를 검색합니다.
GetComboBoxInfo 지정된 콤보 상자에 대한 정보를 검색합니다.
GetCurrentInputMessageSource 입력 메시지의 원본을 검색합니다.
GetCursor 현재 커서에 대한 핸들을 검색합니다.
GetCursorInfo 전역 커서에 대한 정보를 검색합니다.
GetCursorPos 화면 좌표에서 마우스 커서의 위치를 검색합니다.
Getdc GetDC 함수는 지정된 창의 클라이언트 영역 또는 전체 화면에 대한 DC(디바이스 컨텍스트)에 대한 핸들을 검색합니다.
GetDCEx GetDCEx 함수는 지정된 창의 클라이언트 영역 또는 전체 화면에 대한 DC(디바이스 컨텍스트)에 대한 핸들을 검색합니다.
GetDesktopWindow 바탕 화면 창에 대한 핸들을 검색합니다. 바탕 화면 창은 전체 화면을 덮습니다. 바탕 화면 창은 다른 창이 그려지는 위쪽 영역입니다.
GetDialogBaseUnits 시스템 글꼴에서 문자의 평균 너비와 높이인 시스템의 대화 상자 기본 단위를 검색합니다.
GetDialogControlDpiChangeBehavior 대화 상자에서 자식 창의 DPI 크기 조정 동작 재정의를 검색하고 모니터 단위로 설정합니다.

GetDialogDpiChangeBehavior SetDialogDpiChangeBehavior에 대한 이전 호출로 지정된 대화 상자에서 설정되었을 수 있는 플래그를 반환합니다.
GetDisplayAutoRotationPreferences 현재 프로세스에 대한 화면 자동 회전 기본 설정을 검색합니다.
GetDisplayAutoRotationPreferencesByProcessId dwProcessId 매개 변수로 표시된 프로세스에 대한 화면 자동 회전 기본 설정을 검색합니다.
GetDisplayConfigBufferSizes GetDisplayConfigBufferSizes 함수는 QueryDisplayConfig 함수를 호출하는 데 필요한 버퍼의 크기를 검색합니다.
GetDlgCtrlID 지정된 컨트롤의 식별자를 검색합니다.
GetDlgItem 지정된 대화 상자에서 컨트롤에 대한 핸들을 검색합니다.
GetDlgItemInt 대화 상자에서 지정된 컨트롤의 텍스트를 정수 값으로 변환합니다.
GetDlgItemTextA 대화 상자에서 컨트롤과 연결된 제목 또는 텍스트를 검색합니다. (ANSI)
GetDlgItemTextW 대화 상자에서 컨트롤과 연결된 제목 또는 텍스트를 검색합니다. (유니코드)
GetDoubleClickTime 마우스의 현재 두 번 클릭 시간을 가져옵니다.
GetDpiAwarenessContextForProcess 지정된 프로세스에 대한 DPI_AWARENESS_CONTEXT 핸들을 가져옵니다.
GetDpiForSystem 시스템 DPI를 반환합니다.

GetDpiForWindow 지정된 창의 dpi(인치당 점) 값을 반환합니다.
GetDpiFromDpiAwarenessContext 지정된 DPI_AWARENESS_CONTEXT 핸들에서 DPI를 검색합니다. 이렇게 하면 해당 스레드 내에서 만든 창을 검사할 필요 없이 스레드의 DPI를 확인할 수 있습니다.
GetFocus 창이 호출 스레드의 메시지 큐에 연결된 경우 키보드 포커스가 있는 창에 대한 핸들을 가져옵니다.
GetForegroundWindow 전경 창(사용자가 현재 작업 중인 창)에 대한 핸들을 검색합니다. 시스템은 다른 스레드보다 포그라운드 창을 만드는 스레드에 약간 더 높은 우선 순위를 할당합니다.
GetGestureConfig 창에서 Windows Touch 제스처 메시지를 보내는 구성을 검색합니다.
GetGestureExtraArgs GESTUREINFO 핸들에서 제스처에 대한 추가 정보를 검색합니다.
GetGestureInfo 제스처 정보에 대한 핸들이 지정된 GESTUREINFO 구조를 검색합니다.
GetGuiResources 지정된 프로세스에서 사용 중인 GUI(그래픽 사용자 인터페이스) 개체에 대한 핸들 수를 검색합니다.
GetGUIThreadInfo 활성 창 또는 지정된 GUI 스레드에 대한 정보를 검색합니다.
GetIconInfo 지정된 아이콘 또는 커서에 대한 정보를 검색합니다.
GetIconInfoExA 지정된 아이콘 또는 커서에 대한 정보를 검색합니다. GetIconInfoEx는 최신 ICONINFOEX 구조를 사용하여 GetIconInfo를 확장합니다. (ANSI)

GetIconInfoExW 지정된 아이콘 또는 커서에 대한 정보를 검색합니다. GetIconInfoEx는 최신 ICONINFOEX 구조를 사용하여 GetIconInfo를 확장합니다. (유니코드)
GetInputState 호출 스레드의 메시지 큐에 마우스 단추 또는 키보드 메시지가 있는지 여부를 결정합니다.
GetKBCodePage 현재 코드 페이지를 검색합니다.
GetKeyboardLayout 활성 입력 로캘 식별자(이전의 키보드 레이아웃이라고 함)를 검색합니다.
GetKeyboardLayoutList 시스템의 현재 입력 로캘 집합에 해당하는 입력 로캘 식별자(이전의 키보드 레이아웃 핸들)를 가져옵니다. 함수는 식별자를 지정된 버퍼에 복사합니다.
GetKeyboardLayoutNameA 시스템의 활성 입력 로캘 식별자(이전의 키보드 레이아웃이라고 함)의 이름을 검색합니다. (ANSI)
GetKeyboardLayoutNameW 시스템의 활성 입력 로캘 식별자(이전의 키보드 레이아웃이라고 함)의 이름을 검색합니다. (유니코드)
GetKeyboardState 256개 가상 키의 상태를 지정된 버퍼에 복사합니다.
GetKeyboardType 현재 키보드에 대한 정보를 검색합니다.
GetKeyNameTextA 키의 이름을 나타내는 문자열을 가져옵니다. (ANSI)
GetKeyNameTextW 키의 이름을 나타내는 문자열을 가져옵니다. (유니코드)

GetKeyState 지정된 가상 키의 상태를 가져옵니다. 상태 키가 위쪽, 아래쪽 또는 토글되었는지 여부를 지정합니다(키를 누를 때마다 번갈아 가며 켜기, 끄기).
GetLastActivePopup 지정된 창이 소유한 팝업 창이 가장 최근에 활성화되었는지 확인합니다.
GetLastInputInfo 마지막 입력 이벤트의 시간을 가져옵니다.
GetLayeredWindowAttributes 계층적 창의 불투명도 및 투명도 색상 키를 검색합니다.
GetListBoxInfo 지정된 목록 상자에서 열당 항목 수를 검색합니다.
GetMenu 지정된 창에 할당된 메뉴에 대한 핸들을 검색합니다.
GetMenuBarInfo 지정된 메뉴 모음에 대한 정보를 검색합니다.
GetMenuCheckMarkDimensions 기본 검사 표시 비트맵의 차원을 검색합니다.
GetMenuContextHelpId 지정된 메뉴와 연결된 도움말 컨텍스트 식별자를 검색합니다.
GetMenuDefaultItem 지정된 메뉴에서 기본 메뉴 항목을 결정합니다.
GetMenuInfo 지정된 메뉴에 대한 정보를 검색합니다.
GetMenuItemCount 지정된 메뉴의 항목 수를 결정합니다.

GetMenuItemID 메뉴의 지정된 위치에 있는 메뉴 항목의 메뉴 항목 식별자를 검색합니다.
GetMenuItemInfoA 메뉴 항목에 대한 정보를 검색합니다. (ANSI)
GetMenuItemInfoW 메뉴 항목에 대한 정보를 검색합니다. (유니코드)
GetMenuItemRect 지정된 메뉴 항목에 대한 경계 사각형을 검색합니다.
GetMenuState 지정된 메뉴 항목과 연결된 메뉴 플래그를 검색합니다.
GetMenuStringA 지정된 메뉴 항목의 텍스트 문자열을 지정된 버퍼에 복사합니다. (ANSI)
GetMenuStringW 지정된 메뉴 항목의 텍스트 문자열을 지정된 버퍼에 복사합니다. (유니코드)
GetMessage GetMessage 함수는 호출 스레드의 메시지 큐에서 메시지를 검색합니다. (GetMessage)
GetMessageA 호출 스레드의 메시지 큐에서 메시지를 검색합니다. 함수는 게시된 메시지를 검색할 수 있게 될 때까지 들어오는 보낸 메시지를 디스패치합니다. (GetMessageA)
GetMessageExtraInfo 현재 스레드에 대한 추가 메시지 정보를 검색합니다. 추가 메시지 정보는 현재 스레드의 메시지 큐와 연결된 애플리케이션 또는 드라이버 정의 값입니다.
GetMessagePos GetMessage 함수에서 검색한 마지막 메시지의 커서 위치를 검색합니다.
GetMessageTime GetMessage 함수에서 검색한 마지막 메시지의 메시지 시간을 검색합니다.

GetMessageW GetMessageW 함수(유니코드)는 호출 스레드의 메시지 큐에서 메시지를 검색합니다. (GetMessageW)
GetMonitorInfoA GetMonitorInfo 함수는 디스플레이 모니터에 대한 정보를 검색합니다. (ANSI)
GetMonitorInfoW GetMonitorInfo 함수는 디스플레이 모니터에 대한 정보를 검색합니다. (유니코드)
GetMouseMovePointsEx 마우스 또는 펜의 이전 좌표 기록을 최대 64개까지 검색합니다.
GetNextDlgGroupItem 대화 상자에서 지정된 컨트롤 앞에 오거나 뒤에 오는 컨트롤 그룹의 첫 번째 컨트롤에 대한 핸들을 검색합니다.
GetNextDlgTabItem 지정된 컨트롤 앞에 오거나 뒤에 오는 WS_TABSTOP 스타일이 있는 첫 번째 컨트롤에 대한 핸들을 검색합니다.
GetNextWindow Z-Order의 다음 또는 이전 창에 대한 핸들을 검색합니다. 다음 창은 지정된 창 아래에 있습니다. 이전 창이 위에 있습니다.
GetOpenClipboardWindow 현재 클립보드가 열려 있는 창에 대한 핸들을 검색합니다.
GetParent 지정된 창의 부모 또는 소유자에 대한 핸들을 검색합니다.
GetPhysicalCursorPos 실제 좌표에서 커서의 위치를 검색합니다.
GetPointerCursorId 지정된 포인터와 연결된 커서 식별자를 검색합니다.

GetPointerDevice 포인터 디바이스에 대한 정보를 가져옵니다.
GetPointerDeviceCursors 포인터 디바이스와 연결된 커서에 매핑되는 커서 ID를 가져옵니다.
GetPointerDeviceProperties POINTER_DEVICE_INFO 구조에 포함되지 않은 디바이스 속성을 가져옵니다.
GetPointerDeviceRects 포인터 디바이스의 x 및 y 범위(하이메트릭)와 포인터 디바이스가 매핑된 디스플레이의 x 및 y 범위(현재 해상도)를 가져옵니다.
GetPointerDevices 시스템에 연결된 포인터 디바이스에 대한 정보를 가져옵니다.
GetPointerFrameInfo 현재 메시지와 연결된 지정된 포인터에 대한 정보의 전체 프레임을 가져옵니다.
GetPointerFrameInfoHistory 현재 메시지와 연결된 지정된 포인터에 대한 전체 정보 프레임(병합된 입력 프레임 포함)을 가져옵니다.
GetPointerFramePenInfo 현재 메시지와 연결된 지정된 포인터(PT_PEN 형식)에 대한 펜 기반 정보의 전체 프레임을 가져옵니다.
GetPointerFramePenInfoHistory 현재 메시지와 연결된 지정된 포인터(PT_PEN 형식)에 대한 펜 기반 정보(병합된 입력 프레임 포함)의 전체 프레임을 가져옵니다.
GetPointerFrameTouchInfo 현재 메시지와 연결된 지정된 포인터(PT_TOUCH 형식)에 대한 터치 기반 정보의 전체 프레임을 가져옵니다.
GetPointerFrameTouchInfoHistory 현재 메시지와 연결된 지정된 포인터(PT_TOUCH 형식)에 대한 터치 기반 정보(병합된 입력 프레임 포함)의 전체 프레임을 가져옵니다.

GetPointerInfo 현재 메시지와 연결된 지정된 포인터에 대한 정보를 가져옵니다.
GetPointerInfoHistory 지정된 포인터에 대한 현재 메시지에 병합된 개별 입력(있는 경우)과 연결된 정보를 가져옵니다.
GetPointerInputTransform 현재 메시지와 연결된 포인터 정보 좌표에 대한 하나 이상의 변환을 가져옵니다.
GetPointerPenInfo 현재 메시지와 연결된 지정된 포인터(PT_PEN 형식)에 대한 펜 기반 정보를 가져옵니다.
GetPointerPenInfoHistory 지정된 포인터(PT_PEN 형식)에 대한 현재 메시지로 병합된 개별 입력(있는 경우)과 연결된 펜 기반 정보를 가져옵니다.
GetPointerTouchInfo 현재 메시지와 연결된 지정된 포인터(PT_TOUCH 형식)에 대한 터치 기반 정보를 가져옵니다.
GetPointerTouchInfoHistory 지정된 포인터(PT_TOUCH 형식)에 대한 현재 메시지에 병합된 개별 입력(있는 경우)과 연결된 터치 기반 정보를 가져옵니다.
GetPointerType 지정된 포인터의 포인터 형식을 검색합니다.
GetPriorityClipboardFormat 지정된 목록에서 사용 가능한 첫 번째 클립보드 형식을 검색합니다.
GetProcessDefaultLayout 부모 또는 소유자 없이 창을 만들 때 사용되는 기본 레이아웃을 검색합니다.
GetProcessWindowStation 호출 프로세스에 대한 현재 창 스테이션에 대한 핸들을 검색합니다.

GetPropA 지정된 창의 속성 목록에서 데이터 핸들을 검색합니다. 문자열은 검색할 핸들을 식별합니다. SetProp 함수에 대한 이전 호출을 통해 문자열과 핸들을 속성 목록에 추가해야 합니다. (ANSI)
GetPropW 지정된 창의 속성 목록에서 데이터 핸들을 검색합니다. 문자열은 검색할 핸들을 식별합니다. SetProp 함수에 대한 이전 호출을 통해 문자열과 핸들을 속성 목록에 추가해야 합니다. (유니코드)
GetQueueStatus 호출 스레드의 메시지 큐에 있는 메시지 유형을 검색합니다.
GetRawInputBuffer 원시 입력 데이터의 버퍼링된 읽기를 수행합니다.
GetRawInputData 지정된 디바이스에서 원시 입력을 검색합니다.
GetRawInputDeviceInfoA 원시 입력 디바이스에 대한 정보를 검색합니다. (ANSI)
GetRawInputDeviceInfoW 원시 입력 디바이스에 대한 정보를 검색합니다. (유니코드)
GetRawInputDeviceList 시스템에 연결된 원시 입력 디바이스를 열거합니다.
GetRawPointerDeviceData 포인터 디바이스에서 원시 입력 데이터를 가져옵니다.
GetRegisteredRawInputDevices 현재 애플리케이션의 원시 입력 디바이스에 대한 정보를 검색합니다.
GetScrollBarInfo GetScrollBarInfo 함수는 지정된 스크롤 막대에 대한 정보를 검색합니다.

GetScrollInfo GetScrollInfo 함수는 최소 및 최대 스크롤 위치, 페이지 크기, 스크롤 상자(엄지)의 위치를 포함하여 스크롤 막대의 매개 변수를 검색합니다.
GetScrollPos GetScrollPos 함수는 지정된 스크롤 막대에서 스크롤 상자(엄지)의 현재 위치를 검색합니다.
GetScrollRange GetScrollRange 함수는 지정된 스크롤 막대의 현재 최소 및 최대 스크롤 상자(엄지) 위치를 검색합니다.
GetShellWindow 셸의 바탕 화면 창에 대한 핸들을 검색합니다.
GetSubMenu 지정된 메뉴 항목에 의해 활성화된 드롭다운 메뉴 또는 하위 메뉴에 대한 핸들을 검색합니다.
GetSysColor 지정된 표시 요소의 현재 색을 검색합니다.
GetSysColorBrush GetSysColorBrush 함수는 지정된 색 인덱스에 해당하는 논리 브러시를 식별하는 핸들을 검색합니다.
GetSystemDpiForProcess 지정된 프로세스와 연결된 시스템 DPI를 검색합니다. 이는 여러 시스템 인식 프로세스 간에 서로 다른 시스템 DPI 값을 사용하여 DPI에 민감한 정보를 공유하여 발생하는 호환성 문제를 방지하는 데 유용합니다.
GetSystemMenu 애플리케이션이 복사 및 수정을 위해 창 메뉴(시스템 메뉴 또는 컨트롤 메뉴라고도 함)에 액세스할 수 있도록 합니다.
GetSystemMetrics 지정된 시스템 메트릭 또는 시스템 구성 설정을 검색합니다.
GetSystemMetricsForDpi 제공된 DPI를 고려하여 지정된 시스템 메트릭 또는 시스템 구성 설정을 검색합니다.

GetTabbedTextExtentA GetTabbedTextExtent 함수는 문자열의 너비와 높이를 계산합니다. (ANSI)
GetTabbedTextExtentW GetTabbedTextExtent 함수는 문자열의 너비와 높이를 계산합니다. (유니코드)
GetThreadDesktop 지정된 스레드에 할당된 데스크톱에 대한 핸들을 검색합니다.
GetThreadDpiAwarenessContext 현재 스레드에 대한 DPI_AWARENESS_CONTEXT 가져옵니다.
GetThreadDpiHostingBehavior 현재 스레드에서 DPI_HOSTING_BEHAVIOR 검색합니다.
GetTitleBarInfo 지정된 제목 표시줄에 대한 정보를 검색합니다.
GetTopWindow 지정된 부모 창과 연결된 자식 창의 Z 순서를 검사하고 Z 순서 맨 위에 있는 자식 창에 대한 핸들을 검색합니다.
GetTouchInputInfo 특정 터치 입력 핸들과 연결된 터치 입력에 대한 자세한 정보를 검색합니다.
GetUnpredictedMessagePos 터치 예측 처리를 거치기 전에 포인터 데이터를 가져옵니다.
GetUpdatedClipboardFormats 현재 지원되는 클립보드 형식을 검색합니다.
GetUpdateRect GetUpdateRect 함수는 지정된 창의 업데이트 영역을 완전히 묶는 가장 작은 사각형의 좌표를 검색합니다.

GetUpdateRgn GetUpdateRgn 함수는 지정된 지역에 복사하여 창의 업데이트 영역을 검색합니다. 업데이트 영역의 좌표는 창의 왼쪽 위 모서리(즉, 클라이언트 좌표)를 기준으로 합니다.
GetUserObjectInformationA 지정된 창 스테이션 또는 데스크톱 개체에 대한 정보를 검색합니다. (ANSI)
GetUserObjectInformationW 지정된 창 스테이션 또는 데스크톱 개체에 대한 정보를 검색합니다. (유니코드)
GetUserObjectSecurity 지정된 사용자 개체에 대한 보안 정보를 검색합니다.
GetWindow 지정된 창에 대한 지정한 관계(Z-Order 또는 소유자)가 있는 창에 대한 핸들을 검색합니다.
GetWindowContextHelpId 지정된 창과 연결된 도움말 컨텍스트 식별자(있는 경우)를 검색합니다.
GetWindowDC GetWindowDC 함수는 제목 표시줄, 메뉴 및 스크롤 막대를 포함하여 전체 창에 대한 디바이스 컨텍스트(DC)를 검색합니다.
GetWindowDisplayAffinity 지정된 창에 대한 모든 프로세스에서 현재 표시 선호도 설정을 검색합니다.
GetWindowDpiAwarenessContext 창과 연결된 DPI_AWARENESS_CONTEXT 반환합니다.
GetWindowDpiHostingBehavior 지정된 창의 DPI_HOSTING_BEHAVIOR 반환합니다.
GetWindowFeedbackSetting 창에 대한 피드백 구성을 검색합니다.
GetWindowInfo 지정된 창에 대한 정보를 검색합니다. (GetWindowInfo)

GetWindowLongA 지정된 창에 대한 정보를 검색합니다. (GetWindowLongA)
GetWindowLongPtrA 지정된 창에 대한 정보를 검색합니다. 또한 함수는 지정된 오프셋의 값을 추가 창 메모리로 검색합니다. (ANSI)
GetWindowLongPtrW 지정된 창에 대한 정보를 검색합니다. 또한 함수는 지정된 오프셋의 값을 추가 창 메모리로 검색합니다. (유니코드)
GetWindowLongW 지정된 창에 대한 정보를 검색합니다. (GetWindowLongW)
GetWindowModuleFileNameA 지정된 창 핸들과 연결된 모듈의 전체 경로 및 파일 이름을 검색합니다. (ANSI)
GetWindowModuleFileNameW 지정된 창 핸들과 연결된 모듈의 전체 경로 및 파일 이름을 검색합니다. (유니코드)
GetWindowPlacement 표시 상태와 지정된 창의 복원, 최소화 및 최대화된 위치를 검색합니다.
GetWindowRect 지정된 창의 경계 사각형 크기를 검색합니다. 차원은 화면의 왼쪽 위 모서리를 기준으로 하는 화면 좌표로 제공됩니다.
GetWindowRgn GetWindowRgn 함수는 창의 창 영역 복사본을 가져옵니다.
GetWindowRgnBox GetWindowRgnBox 함수는 창의 창 영역에 대한 가장 엄격한 경계 사각형의 크기를 검색합니다.
GetWindowTextA 지정된 창의 제목 표시줄 텍스트(있는 경우)를 버퍼에 복사합니다. 지정한 창이 컨트롤이면 컨트롤의 텍스트가 복사됩니다. 그러나 GetWindowText는 다른 애플리케이션에서 컨트롤의 텍스트를 검색할 수 없습니다. (ANSI)

GetWindowTextLengthA 창에 제목 표시줄이 있는 경우 지정된 창의 제목 표시줄 텍스트의 길이를 문자 단위로 검색합니다. (ANSI)
GetWindowTextLengthW 창에 제목 표시줄이 있는 경우 지정된 창의 제목 표시줄 텍스트의 길이를 문자 단위로 검색합니다. (유니코드)
GetWindowTextW 지정된 창의 제목 표시줄 텍스트(있는 경우)를 버퍼에 복사합니다. 지정한 창이 컨트롤이면 컨트롤의 텍스트가 복사됩니다. 그러나 GetWindowText는 다른 애플리케이션에서 컨트롤의 텍스트를 검색할 수 없습니다. (유니코드)
GetWindowThreadProcessId 지정된 창을 만든 스레드의 식별자와 필요에 따라 창을 만든 프로세스의 식별자를 검색합니다.
GetWindowWord 지정된 오프셋의 16비트(**DWORD**) 값을 추가 창 메모로 검색합니다.
GID_ROTATE_ANGLE_FROM_ARGUMENT GID_ROTATE_ANGLE_FROM_ARGUMENT 매크로는 WM_GESTURE 구조체에서 값을 받을 때 GID_ROTATE ullArgument 값을 해석하는 데 사용됩니다.
GID_ROTATE_ANGLE_TO_ARGUMENT 회전 제스처 메시지의 인수로 라디안 값을 변환합니다.
GrayStringA GrayString 함수는 지정된 위치에 회색 텍스트를 그립니다. (ANSI)
GrayStringW GrayString 함수는 지정된 위치에 회색 텍스트를 그립니다. (유니코드)
HAS_POINTER_CONFIDENCE_WPARAM 지정한 포인터 메시지가 실수로 인한 것이 아니라 의도적인 것으로 간주되는지 확인합니다.
HideCaret 화면에서 캐리트를 제거합니다. 캐리트를 숨기면 현재 도형이 삭제되거나 삽입 지점이 무효화되지 않습니다.

HiliteMenuItem 메뉴 모음의 항목에서 강조 표시를 추가하거나 제거합니다.
InflateRect InflateRect 함수는 지정된 사각형의 너비와 높이를 늘리거나 줄입니다.
InheritWindowMonitor 지정된 창이 다른 창의 모니터를 상속하도록 합니다.
InitializeTouchInjection 호출 애플리케이션에 대한 터치 삽입 컨텍스트를 구성하고 앱이 삽입할 수 있는 최대 동시 접점 수를 초기화합니다.
InjectSyntheticPointerInput 포인터 입력(펜 또는 터치)을 시뮬레이션합니다.
InjectTouchInput 터치 입력을 시뮬레이션합니다.
InSendMessage 현재 창 프로시저가 SendMessage 함수를 호출하여 다른 스레드(동일한 프로세스 또는 다른 프로세스)에서 보낸 메시지를 처리하고 있는지 여부를 확인합니다.
InSendMessageEx 현재 창 프로시저가 다른 스레드(동일한 프로세스 또는 다른 프로세스)에서 보낸 메시지를 처리하고 있는지 여부를 확인합니다.
InsertMenuA 메뉴에 새 메뉴 항목을 삽입하고 다른 항목을 메뉴 아래로 이동합니다. (ANSI)
InsertMenuItemA 메뉴의 지정된 위치에 새 메뉴 항목을 삽입합니다. (ANSI)
InsertMenuItemW 메뉴의 지정된 위치에 새 메뉴 항목을 삽입합니다. (유니코드)

InsertMenuW 메뉴에 새 메뉴 항목을 삽입하고 다른 항목을 메뉴 아래로 이동합니다. (유니코드)
InternalGetWindowText 지정된 창의 제목 표시줄 텍스트(있는 경우)를 버퍼에 복사합니다.
IntersectRect IntersectRect 함수는 두 원본 사각형의 교집합을 계산하고 교차 사각형의 좌표를 대상 사각형에 배치합니다.
InvalidateRect InvalidateRect 함수는 지정된 창의 업데이트 영역에 사각형을 추가합니다. 업데이트 지역은 다시 그려야 하는 창의 클라이언트 영역 부분을 나타냅니다.
InvalidateRgn InvalidateRgn 함수는 창의 현재 업데이트 영역에 추가하여 지정된 지역 내의 클라이언트 영역을 무효화합니다.
InvertRect InvertRect 함수는 사각형 내부의 각 픽셀에 대한 색 값에 대해 논리 NOT 연산을 수행하여 창에서 사각형을 반전합니다.
IS_INTRESOURCE 값이 리소스의 정수 식별자인지 여부를 확인합니다.
IS_POINTER_CANCELED_WPARAM 지정한 포인터 입력이 갑자기 종료되었는지 또는 잘못된지 확인하여 상호 작용이 완료되지 않았음을 나타냅니다.
IS_POINTER_FIFTHBUTTON_WPARAM 지정된 포인터가 다섯 번째 작업을 수행했는지 여부를 확인합니다.
IS_POINTER_FIRSTBUTTON_WPARAM 지정한 포인터가 첫 번째 작업을 수행했는지 여부를 확인합니다.
IS_POINTER_FLAG_SET_WPARAM 포인터 매크로가 지정된 플래그를 설정하는지 여부를 확인합니다.

IS_POINTER_FOURTHBUTTON_WPARAM 지정된 포인터가 네 번째 작업을 수행했는지 여부를 확인합니다.
IS_POINTER_INCONTACT_WPARAM 지정한 포인터가 접촉하는지 여부를 확인합니다.
IS_POINTER_INRANGE_WPARAM 지정된 포인터가 범위에 있는지 확인합니다.
IS_POINTER_NEW_WPARAM 지정된 포인터가 새 포인터인지 여부를 확인합니다.
IS_POINTER_SECONDBUTTON_WPARAM 지정된 포인터가 두 번째 작업을 수행했는지 여부를 확인합니다.
IS_POINTER_THIRDBUTTON_WPARAM 지정된 포인터가 세 번째 작업을 수행했는지 여부를 확인합니다.
IsCharAlphaA 문자가 사전순 문자인지 여부를 결정합니다. 이 결정은 설치 중 또는 제어판 통해 사용자가 선택한 언어의 의미 체계를 기반으로 합니다. (ANSI)
IsCharAlphaNumericA 문자가 사전순인지 숫자 문자인지 여부를 결정합니다. 이 결정은 설치 중 또는 제어판 통해 사용자가 선택한 언어의 의미 체계를 기반으로 합니다. (ANSI)
IsCharAlphaNumericW 문자가 사전순인지 숫자 문자인지 여부를 결정합니다. 이 결정은 설치 중 또는 제어판 통해 사용자가 선택한 언어의 의미 체계를 기반으로 합니다. (유니코드)
IsCharAlphaW 문자가 사전순 문자인지 여부를 결정합니다. 이 결정은 설치 중 또는 제어판 통해 사용자가 선택한 언어의 의미 체계를 기반으로 합니다. (유니코드)
IsCharLowerA 문자가 소문자인지 여부를 확인합니다. 이 결정은 설치 중 또는 제어판 통해 사용자가 선택한 언어의 의미 체계를 기반으로 합니다.

IsCharLowerW IsCharLowerW(유니코드) 함수는 문자가 소문자인지 여부를 결정합니다. (IsCharLowerW)
IsCharUpperA 문자가 대문자인지 여부를 결정합니다. 이 결정은 설치 중 또는 제어판 통해 사용자가 선택한 언어의 의미 체계를 기반으로 합니다. (ANSI)
IsCharUpperW 문자가 대문자인지 여부를 결정합니다. 이 결정은 설치 중 또는 제어판 통해 사용자가 선택한 언어의 의미 체계를 기반으로 합니다. (유니코드)
IsChild 창이 지정된 부모 창의 자식 창인지 하위 창인지 여부를 결정합니다.
IsClipboardFormatAvailable 클립보드에 지정된 형식의 데이터가 포함되어 있는지 여부를 확인합니다.
IsDialogMessageA 메시지가 지정된 대화 상자의 용도인지 여부를 확인하고 메시지인 경우 메시지를 처리합니다. (ANSI)
IsDialogMessageW 메시지가 지정된 대화 상자의 용도인지 여부를 확인하고 메시지인 경우 메시지를 처리합니다. (유니코드)
IsDlgButtonChecked IsDlgButtonChecked 함수는 단추 컨트롤이 선택되었는지 또는 세 가지 상태의 단추 컨트롤이 선택되었는지, 선택되지 않았는지 또는 결정되지 않았는지를 결정합니다.
IsGUIThread 호출 스레드가 이미 GUI 스레드인지 여부를 확인합니다. 필요에 따라 스레드를 GUI 스레드로 변환할 수도 있습니다.
IsHungAppWindow 시스템에서 지정된 애플리케이션이 응답하지 않는 것으로 간주하는지 여부를 결정합니다.
IsIconic 지정된 창이 최소화되는지 여부를 결정합니다(아이콘).

IsImmersiveProcess 프로세스가 Windows 스토어 앱에 속하는지 여부를 결정합니다.
IsMenu 핸들이 메뉴 핸들인지 여부를 확인합니다.
IsMouseInPointerEnabled 마우스가 포인터 입력 디바이스 역할을 하고 WM_POINTER 메시지를 보낼 수 있도록 EnableMouseInPointer가 설정되어 있는지 여부를 나타냅니다.
IsProcessDPIAware IsProcessDPIAware를 변경하거나 사용할 수 없습니다. 대신 GetProcessDPIAwareness를 사용합니다.
IsRectEmpty IsRectEmpty 함수는 지정된 사각형이 비어 있는지 여부를 결정합니다.
IsTouchWindow 지정된 창이 터치 가능인지 확인하고 필요에 따라 창의 터치 기능에 대해 설정된 한정자 플래그를 검색합니다.
IsValidDpiAwarenessContext 지정된 DPI_AWARENESS_CONTEXT 유효하고 현재 시스템에서 지원하는지 확인합니다.
IsWindow 지정된 창 핸들이 기존 창을 식별하는지 여부를 결정합니다.
IsWindowArranged 지정된 창이 정렬되어 있는지(즉, 스냅되었는지 여부)를 결정합니다.
IsWindowEnabled 지정된 창에서 마우스 및 키보드 입력을 사용할 수 있는지 여부를 결정합니다.
IsWindowUnicode 지정된 창이 네이티브 유니코드 창인지 여부를 결정합니다.

IsWindowVisible 지정된 창의 표시 여부를 결정합니다.
IsWinEventHookInstalled 지정된 이벤트에 대한 알림을 받을 수 있는 설치된 WinEvent 후크가 있는지 여부를 결정합니다.
IsWow64Message 현재 스레드의 큐에서 읽은 마지막 메시지가 WOW64 프로세스에서 시작되었는지 여부를 결정합니다.
IsZoomed 창을 최대화할지 여부를 결정합니다.
keybd_event 키 입력을 합성합니다.
KillTimer 지정된 타이머를 삭제합니다.
LoadAcceleratorsA 지정된 가속기 테이블을 로드합니다. (ANSI)
LoadAcceleratorsW 지정된 가속기 테이블을 로드합니다. (유니코드)
LoadBitmapA LoadBitmap 함수는 모듈의 실행 파일에서 지정된 비트맵 리소스를 로드합니다. (ANSI)
LoadBitmapW LoadBitmap 함수는 모듈의 실행 파일에서 지정된 비트맵 리소스를 로드합니다. (유니코드)
LoadCursorA 애플리케이션 instance 연결된 실행 파일(.EXE) 파일에서 지정된 커서 리소스를 로드합니다. (ANSI)
LoadCursorFromFileA 파일에 포함된 데이터를 기반으로 커서를 만듭니다. (ANSI)

LoadCursorFromFileW 파일에 포함된 데이터를 기반으로 커서를 만듭니다. (유니코드)
LoadCursorW 애플리케이션 instance 연결된 실행 파일(.EXE) 파일에서 지정된 커서 리소스를 로드합니다. (유니코드)
LoadIconA 애플리케이션 instance 연결된 실행 파일(.exe) 파일에서 지정된 아이콘 리소스를 로드합니다. (ANSI)
LoadIconW 애플리케이션 instance 연결된 실행 파일(.exe) 파일에서 지정된 아이콘 리소스를 로드합니다. (유니코드)
LoadImageA 아이콘, 커서, 애니메이션 커서 또는 비트맵을 로드합니다. (ANSI)
LoadImageW 아이콘, 커서, 애니메이션 커서 또는 비트맵을 로드합니다. (유니코드)
LoadKeyboardLayoutA 새 입력 로캘 식별자(이전의 키보드 레이아웃)를 시스템에 로드합니다. (ANSI)
LoadKeyboardLayoutW 새 입력 로캘 식별자(이전의 키보드 레이아웃)를 시스템에 로드합니다. (유니코드)
LoadMenuA 애플리케이션 instance 연결된 실행 파일(.exe) 파일에서 지정된 메뉴 리소스를 로드합니다. (ANSI)
LoadMenuIndirectA 지정된 메뉴 템플릿을 메모리에 로드합니다. (ANSI)
LoadMenuIndirectW 지정된 메뉴 템플릿을 메모리에 로드합니다. (유니코드)

LoadMenuW 애플리케이션 instance 연결된 실행 파일(.exe) 파일에서 지정된 메뉴 리소스를 로드합니다. (유니코드)
LoadStringA 지정된 모듈과 연결된 실행 파일에서 문자열 리소스를 로드하고, 문자열을 버퍼에 복사하고, 종료 null 문자를 추가합니다. (ANSI)
LoadStringW 지정된 모듈과 연결된 실행 파일에서 문자열 리소스를 로드하고, 문자열을 버퍼에 복사하고, 종료 null 문자를 추가합니다. (유니코드)
LockSetForegroundWindow 포그라운드 프로세스는 LockSetForegroundWindow 함수를 호출하여 SetForegroundWindow 함수에 대한 호출을 사용하지 않도록 설정할 수 있습니다.
LockWindowUpdate LockWindowUpdate 함수는 지정된 창에서 그리기를 사용하지 않도록 설정하거나 사용하도록 설정합니다. 한 번에 하나의 창만 잠글 수 있습니다.
LockWorkStation 워크스테이션의 디스플레이를 잠근다.
LogicalToPhysicalPoint 창에 있는 점의 논리적 좌표를 실제 좌표로 변환합니다.
LogicalToPhysicalPointForPerMonitorDPI 호출자의 dpi(인치당 점) 인식에 관계없이 창의 점을 논리적 좌표에서 물리적 좌표로 변환합니다.
LookupIconIdFromDirectory 아이콘 또는 커서 데이터를 검색하여 현재 디스플레이 디바이스에 가장 적합한 아이콘 또는 커서를 검색합니다. (LookupIconIdFromDirectory)
LookupIconIdFromDirectoryEx 아이콘 또는 커서 데이터를 검색하여 현재 디스플레이 디바이스에 가장 적합한 아이콘 또는 커서를 검색합니다. (LookupIconIdFromDirectoryEx)

MAKEINTRESOURCEA 정수 값을 리소스 관리 함수와 호환되는 리소스 형식으로 변환합니다. 이 매크로는 리소스 이름을 포함하는 문자열 대신 사용됩니다. (ANSI)
MAKEINTRESOURCEW 정수 값을 리소스 관리 함수와 호환되는 리소스 형식으로 변환합니다. 이 매크로는 리소스 이름을 포함하는 문자열 대신 사용됩니다. (유니코드)
MAKELPARAM 메시지에서 lParam 매개 변수로 사용할 값을 만듭니다. 매크로는 지정된 값을 연결합니다.
MAKELRESULT 창 프로시저에서 반환 값으로 사용할 값을 만듭니다. 매크로는 지정된 값을 연결합니다.
MAKEWPARAM 메시지에서 wParam 매개 변수로 사용할 값을 만듭니다. 매크로는 지정된 값을 연결합니다.
MapDialogRect 지정된 대화 상자 단위를 화면 단위(픽셀)로 변환합니다.
MapVirtualKeyA 가상 키 코드를 스캔 코드 또는 문자 값으로 변환(매핑)하거나 스캔 코드를 가상 키 코드로 변환합니다. (ANSI)
MapVirtualKeyExA 가상 키 코드를 스캔 코드 또는 문자 값으로 변환(매핑)하거나 스캔 코드를 가상 키 코드로 변환합니다. 함수는 입력 언어 및 입력 로캘 식별자를 사용하여 코드를 변환합니다. (ANSI)
MapVirtualKeyExW 가상 키 코드를 스캔 코드 또는 문자 값으로 변환(매핑)하거나 스캔 코드를 가상 키 코드로 변환합니다. 함수는 입력 언어 및 입력 로캘 식별자를 사용하여 코드를 변환합니다. (유니코드)
MapVirtualKeyW 가상 키 코드를 스캔 코드 또는 문자 값으로 변환(매핑)하거나 스캔 코드를 가상 키 코드로 변환합니다. (유니코드)

MapWindowPoints MapWindowPoints 함수는 한 창을 기준으로 한 좌표 공간에서 다른 창을 기준으로 좌표 공간으로 점 집합을 변환(매핑)합니다.
MenuItemFromPoint 지정된 위치에 있는 메뉴 항목(있는 경우)을 결정합니다.
MessageBeep 파형 소리를 재생합니다. 각 사운드 유형에 대한 파형 소리는 레지스트리의 항목으로 식별됩니다.
MessageBox MessageBox 함수는 시스템 아이콘, 단추 집합 및 간단한 애플리케이션 관련 메시지가 포함된 모달 대화 상자를 표시합니다.
MessageBoxA 시스템 아이콘, 단추 집합 및 상태 또는 오류 정보와 같은 간단한 애플리케이션 관련 메시지가 포함된 모달 대화 상자를 표시합니다. 메시지 상자는 사용자가 클릭한 단추를 나타내는 정수 값을 반환합니다. (MessageBoxA)
MessageBoxExA 메시지 상자를 만들고, 표시하고, 작동합니다. (ANSI)
MessageBoxExW 메시지 상자를 만들고, 표시하고, 작동합니다. (유니코드)
MessageBoxIndirectA 메시지 상자를 만들고, 표시하고, 작동합니다. 메시지 상자에는 애플리케이션 정의 메시지 텍스트와 제목, 아이콘 및 미리 정의된 푸시 단추 조합이 포함됩니다. (ANSI)
MessageBoxIndirectW 메시지 상자를 만들고, 표시하고, 작동합니다. 메시지 상자에는 애플리케이션 정의 메시지 텍스트와 제목, 아이콘 및 미리 정의된 푸시 단추 조합이 포함됩니다. (유니코드)
MessageBoxW MessageBoxW(유니코드) 함수는 시스템 아이콘, 단추 집합 및 간단한 애플리케이션 관련 메시지가 포함된 모달 대화 상자를 표시합니다.

ModifyMenuA 기존 메뉴 항목을 변경합니다. (ANSI)
ModifyMenuW 기존 메뉴 항목을 변경합니다. (유니코드)
MonitorFromPoint MonitorFromPoint 함수는 지정된 지점을 포함하는 디스플레이 모니터에 대한 핸들을 검색합니다.
MonitorFromRect MonitorFromRect 함수는 지정된 사각형과 가장 큰 교집합 영역이 있는 디스플레이 모니터에 대한 핸들을 검색합니다.
MonitorFromWindow MonitorFromWindow 함수는 지정된 창의 경계 사각형과 가장 큰 교집합 영역이 있는 디스플레이 모니터에 대한 핸들을 검색합니다.
mouse_event mouse_event 함수는 마우스 동작과 단추 클릭을 합성합니다.
MoveWindow 지정된 창의 위치와 크기를 변경합니다.
MsgWaitForMultipleObjects 지정된 개체 중 하나 또는 전부가 신호 상태이거나 제한 시간 간격이 경과할 때까지 기다립니다. 개체에는 입력 이벤트 개체가 포함될 수 있습니다.
MsgWaitForMultipleObjectsEx 지정된 개체 중 하나 또는 전부가 신호 상태에 있거나, I/O 완료 루틴 또는 APC(비동기 프로시저 호출)가 스레드에 큐에 대기하거나 시간 제한 간격이 경과할 때까지 기다립니다. 개체의 배열에는 입력 이벤트 개체가 포함될 수 있습니다.
NEXTRAWINPUTBLOCK RAWINPUT 구조체 배열에서 다음 구조체의 위치를 검색합니다.
NotifyWinEvent 미리 정의된 이벤트가 발생한 시스템에 신호를 보냅니다. 클라이언트 애플리케이션이 이벤트에 대한 후크 함수를 등록한 경우 시스템은 클라이언트의 후크 함수를 호출합니다.

OemKeyScan OEMASCII 코드 0~0x0FF를 OEM 스캔 코드 및 시프트 상태에 매핑합니다. 이 함수는 프로그램이 키보드 입력을 시뮬레이션하여 OEM 텍스트를 다른 프로그램으로 보낼 수 있도록 하는 정보를 제공합니다.
OemToChara OEM 정의 문자 집합의 문자열을 ANSI 또는 와이드 문자열로 변환합니다. 경고 사용하지 마세요. (ANSI)
OemToCharBuffA OEM 정의 문자 집합에서 문자열의 지정된 수의 문자를 ANSI 또는 와이드 문자열로 변환합니다. (ANSI)
OemToCharBuffW OEM 정의 문자 집합에서 문자열의 지정된 수의 문자를 ANSI 또는 와이드 문자열로 변환합니다. (유니코드)
OemToCharW OEM 정의 문자 집합의 문자열을 ANSI 또는 와이드 문자열로 변환합니다. 경고 사용하지 마세요. (유니코드)
OffsetRect OffsetRect 함수는 지정된 사각형을 지정된 오프셋으로 이동합니다.
OpenClipboard 검사를 위해 클립보드를 열고 다른 애플리케이션이 클립보드 콘텐츠를 수정하지 못하도록 합니다.
OpenDesktopA 지정된 데스크톱 개체를 엽니다. (ANSI)
OpenDesktopW 지정된 데스크톱 개체를 엽니다. (유니코드)
OpenIcon 최소화된(상징적) 창을 이전 크기와 위치로 복원합니다. 그러면 창이 활성화됩니다.
OpenInputDesktop 사용자 입력을 수신하는 데스크톱을 엽니다.

OpenWindowStationA 지정된 창 스테이션을 엽니다. (ANSI)
OpenWindowStationW 지정된 창 스테이션을 엽니다. (유니코드)
PackTouchHitTestingProximityEvaluation 근접 평가 점수와 조정된 터치 포인트 좌표를 WM_TOUCHHITTESTING 콜백의 압축된 값으로 반환합니다.
PaintDesktop PaintDesktop 함수는 지정된 디바이스 컨텍스트의 클리핑 영역을 바탕 화면 패턴 또는 배경 화면으로 채웁니다. 함수는 주로 셸 데스크톱용으로 제공됩니다.
PeekMessageA 큐에 추가되지 않은 들어오는 메시지를 디스패치하고, 게시된 메시지에 대한 스레드 메시지 큐를 확인하고, 메시지를 검색합니다(있는 경우). (ANSI)
PeekMessageW 큐에 추가되지 않은 들어오는 메시지를 디스패치하고, 게시된 메시지에 대한 스레드 메시지 큐를 확인하고, 메시지를 검색합니다(있는 경우). (유니코드)
PhysicalToLogicalPoint 창에 있는 점의 실제 좌표를 논리 좌표로 변환합니다.
PhysicalToLogicalPointForPerMonitorDPI 호출자의 dpi(인치당 점 수) 인식에 관계없이 창의 점을 실제 좌표에서 논리적 좌표로 변환합니다.
POINTSTOPOINT POINTSTOPOINT 매크로는 POINTS 구조체의 내용을 POINT 구조체에 복사합니다.
POINTTOPOINTS POINTTOPOINTS 매크로는 POINT 구조를 POINTS 구조체로 변환합니다.
PostMessageA 지정된 창을 만든 스레드와 연결된 메시지 큐에 메시지를 배치(게시)하고 스레드가 메시지를 처리할 때까지 기다리지 않고 반환합니다. (ANSI)

PostMessageW 지정된 창을 만든 스레드와 연결된 메시지 큐에 메시지를 배치(게시)하고 스레드가 메시지를 처리할 때까지 기다리지 않고 반환합니다. (유니코드)
PostQuitMessage 스레드가 종료(종료)를 요청했음을 시스템에 나타냅니다. 일반적으로 WM_DESTROY 메시지에 대한 응답으로 사용됩니다.
PostThreadMessageA 지정된 스레드의 메시지 큐에 메시지를 게시합니다. 스레드가 메시지를 처리할 때까지 기다리지 않고 반환됩니다. (ANSI)
PostThreadMessageW 지정된 스레드의 메시지 큐에 메시지를 게시합니다. 스레드가 메시지를 처리할 때까지 기다리지 않고 반환됩니다. (유니코드)
PrintWindow PrintWindow 함수는 시각적 창을 지정된 DC(디바이스 컨텍스트), 일반적으로 프린터 DC에 복사합니다.
PrivateExtractIconsA 지정된 파일에서 추출된 아이콘에 대한 핸들 배열을 만듭니다. (ANSI)
PrivateExtractIconsW 지정된 파일에서 추출된 아이콘에 대한 핸들 배열을 만듭니다. (유니코드)
PtInRect PtInRect 함수는 지정된 점이 지정된 사각형 내에 있는지 여부를 결정합니다.
QueryDisplayConfig QueryDisplayConfig 함수는 현재 설정에서 모든 디스플레이 디바이스 또는 보기에 대해 가능한 모든 표시 경로에 대한 정보를 검색합니다.
RealChildWindowFromPoint 지정된 지점에서 자식 창에 대한 핸들을 검색합니다. 검색은 바로 자식 창으로 제한됩니다. 손자와 더 깊은 하위 창은 검색되지 않습니다.

RealGetWindowClassA 창 유형을 지정하는 문자열을 검색합니다. (ANSI)
RealGetWindowClassW 창 유형을 지정하는 문자열을 검색합니다. (유니코드)
RedrawWindow RedrawWindow 함수는 창의 클라이언트 영역에서 지정된 사각형 또는 영역을 업데이트합니다.
RegisterClassA CreateWindow 또는 CreateWindowEx 함수에 대한 호출에서 이후에 사용할 창 클래스를 등록합니다. (RegisterClassA)
RegisterClassExA CreateWindow 또는 CreateWindowEx 함수에 대한 호출에서 이후에 사용할 창 클래스를 등록합니다. (RegisterClassExA)
RegisterClassExW CreateWindow 또는 CreateWindowEx 함수에 대한 호출에서 이후에 사용할 창 클래스를 등록합니다. (RegisterClassExW)
RegisterClassW CreateWindow 또는 CreateWindowEx 함수에 대한 호출에서 이후에 사용할 창 클래스를 등록합니다. (RegisterClassW)
RegisterClipboardFormatA 새 클립보드 형식을 등록합니다. 그런 다음 이 형식을 유효한 클립보드 형식으로 사용할 수 있습니다. (ANSI)
RegisterClipboardFormatW 새 클립보드 형식을 등록합니다. 그런 다음 이 형식을 유효한 클립보드 형식으로 사용할 수 있습니다. (유니코드)
RegisterDeviceNotificationA 창에서 알림을 받을 디바이스 또는 디바이스 유형을 등록합니다. (ANSI)
RegisterDeviceNotificationW 창에서 알림을 받을 디바이스 또는 디바이스 유형을 등록합니다. (유니코드)

RegisterForTooltipDismissNotification 앱 또는 UI 프레임워크가 창을 등록 및 등록 취소하여 도구 설명 창을 해제하는 알림을 받을 수 있습니다.
RegisterHotKey 시스템 전체 핫 키를 정의합니다.
RegisterPointerDeviceNotifications WM_POINTERDEVICECHANGE, WM_POINTERDEVICEINRANGE 및 WM_POINTERDEVICEOUTOFRANGE 포인터 디바이스 알림을 처리하는 창을 등록합니다.
RegisterPointerInputTarget 호출자가 지정된 형식의 모든 포인터 입력이 리디렉션되는 대상 창을 등록할 수 있습니다.
RegisterPointerInputTargetEx RegisterPointerInputTargetEx를 변경하거나 사용할 수 없습니다. 대신 RegisterPointerInputTarget을 사용합니다.
RegisterPowerSettingNotification 애플리케이션을 등록하여 특정 전원 설정 이벤트에 대한 전원 설정 알림을 받습니다.
RegisterRawInputDevices 원시 입력 데이터를 제공하는 디바이스를 등록합니다.
RegisterShellHookWindow 지정된 셸 창을 등록하여 셸 애플리케이션에 유용한 이벤트 또는 알림에 대한 특정 메시지를 받습니다.
RegisterSuspendResumeNotification 시스템이 일시 중단되거나 다시 시작될 때 알림을 받도록 등록합니다. PowerRegisterSuspendResumeNotification과 비슷하지만 사용자 모드에서 작동하며 창 핸들을 사용할 수 있습니다.
RegisterTouchHitTestingWindow WM_TOUCHHITTESTING 알림을 처리하는 창을 등록합니다.
RegisterTouchWindow 창을 터치 가능으로 등록합니다.

RegisterWindowMessageA 시스템 전체에서 고유하도록 보장되는 새 창 메시지를 정의합니다. 메시지 값은 메시지를 보내거나 게시할 때 사용할 수 있습니다. (ANSI)
RegisterWindowMessageW 시스템 전체에서 고유하도록 보장되는 새 창 메시지를 정의합니다. 메시지 값은 메시지를 보내거나 게시할 때 사용할 수 있습니다. (유니코드)
ReleaseCapture 현재 스레드의 창에서 마우스 캡처를 해제하고 일반 마우스 입력 처리를 복원합니다.
ReleaseDC ReleaseDC 함수는 디바이스 컨텍스트(DC)를 해제하여 다른 애플리케이션에서 사용할 수 있게 합니다. ReleaseDC 함수의 효과는 DC 유형에 따라 달라집니다. 일반 및 창 DC만 해제합니다. 클래스 또는 개인 DC에는 영향을 주지 않습니다.
RemoveClipboardFormatListener 시스템이 유지 관리하는 클립보드 형식 수신기 목록에서 지정된 창을 제거합니다.
RemoveMenu 메뉴 항목을 삭제하거나 지정된 메뉴에서 하위 메뉴를 분리합니다.
RemovePropA 지정된 창의 속성 목록에서 항목을 제거합니다. 지정된 문자열은 제거할 항목을 식별합니다. (ANSI)
RemovePropW 지정된 창의 속성 목록에서 항목을 제거합니다. 지정된 문자열은 제거할 항목을 식별합니다. (유니코드)
ReplyMessage SendMessage 함수가 다른 스레드에서 보낸 메시지에 회신합니다.
ScreenToClient ScreenToClient 함수는 화면에서 지정된 지점의 화면 좌표를 클라이언트 영역 좌표로 변환합니다.
ScrollDC ScrollDC 함수는 비트의 직사각형을 수평 및 수직으로 스크롤합니다.

ScrollWindow ScrollWindow 함수는 지정된 창의 클라이언트 영역 콘텐츠를 스크롤합니다.
ScrollWindowEx ScrollWindowEx 함수는 지정된 창의 클라이언트 영역 콘텐츠를 스크롤합니다.
SendDlgItemMessageA 대화 상자에서 지정된 컨트롤에 메시지를 보냅니다. (ANSI)
SendDlgItemMessageW 대화 상자에서 지정된 컨트롤에 메시지를 보냅니다. (유니코드)
SendInput 키 입력, 마우스 동작 및 단추 클릭을 합성합니다.
SendMessage SendMessage 함수는 지정된 메시지를 창 또는 창으로 보냅니다. (SendMessage 함수)
SendMessageA 지정된 메시지를 창 또는 창으로 보냅니다. SendMessage 함수는 지정된 창에 대한 창 프로시저를 호출하며 창 프로시저가 메시지를 처리할 때까지 반환되지 않습니다. (SendMessageA)
SendMessageCallbackA 지정된 메시지를 창 또는 창으로 보냅니다. (SendMessageCallbackA)
SendMessageCallbackW 지정된 메시지를 창 또는 창으로 보냅니다. (SendMessageCallbackW)
SendMessageTimeoutA 지정된 메시지를 하나 이상의 창으로 보냅니다. (ANSI)
SendMessageTimeoutW 지정된 메시지를 하나 이상의 창으로 보냅니다. (유니코드)
SendMessageW SendMessageW(유니코드) 함수는 지정된 메시지를 창 또는 창으로 보냅니다. (SendMessageW)

SendNotifyMessageA 지정된 메시지를 창 또는 창으로 보냅니다. (SendNotifyMessageA)
SendNotifyMessageW 지정된 메시지를 창 또는 창으로 보냅니다. (SendNotifyMessageW)
SetActiveWindow 창을 활성화합니다. 창은 호출 스레드의 메시지 큐에 연결되어야 합니다.
SetAdditionalForegroundBoostProcesses SetAdditionalForegroundBoostProcesses는 여러 프로세스가 데이터 또는 렌더링으로 포그라운드 환경에 기여하는 다중 프로세스 애플리케이션 모델을 사용하는 애플리케이션을 지원하는 성능 지원 API입니다.
SetCapture 마우스 캡처를 현재 스레드에 속하는 지정된 창으로 설정합니다.
SetCaretBlinkTime 캐리트 깜박임 시간을 지정된 시간(밀리초)으로 설정합니다. 깜박임 시간은 캐리트의 픽셀을 반전하는 데 필요한 경과 시간(밀리초)입니다.
SetCaretPos 지정된 좌표로 캐리트를 이동합니다. caret를 소유하는 창이 CS_OWNDC 클래스 스타일로 만들어진 경우 지정된 좌표에는 해당 창과 연결된 디바이스 컨텍스트의 매핑 모드가 적용됩니다.
SetClassLongA 지정된 오프셋의 지정된 32비트(long) 값을 지정된 창이 속한 클래스의 추가 클래스 메모리 또는 WNDCLASSEX 구조체로 바꿉니다. (ANSI)
SetClassLongPtrA 지정된 창이 속한 클래스의 추가 클래스 메모리 또는 WNDCLASSEX 구조체에서 지정된 오프셋의 지정된 값을 바꿉니다. (ANSI)
SetClassLongPtrW 지정된 창이 속한 클래스의 추가 클래스 메모리 또는 WNDCLASSEX 구조체에서 지정된 오프셋의 지정된 값을 바꿉니다. (유니코드)

SetClassLongW 지정된 오프셋의 지정된 32비트(long) 값을 지정된 창이 속한 클래스의 추가 클래스 메모리 또는 WNDCLASSEX 구조체로 바꿉니다. (유니코드)
SetClassWord 지정된 오프셋의 16비트(WORD) 값을 지정된 창이 속한 창 클래스의 추가 클래스 메모리로 바꿉니다.
SetClipboardData 지정된 클립보드 형식으로 클립보드에 데이터를 배치합니다.
SetClipboardViewer 클립보드 뷰어 체인에 지정된 창을 추가합니다. 클립보드 뷰어 창은 클립보드의 콘텐츠가 변경될 때마다 WM_DRAWCLIPBOARD 메시지를 받습니다. 이 함수는 이전 버전의 Windows와 이전 버전과의 호환성을 위해 사용됩니다.
SetCoalescableTimer 지정된 제한 시간 값과 병합 허용 오차 지연을 사용하여 타이머를 만듭니다.
SetCursor 커서 모양을 설정합니다.
SetCursorPos 커서를 지정된 화면 좌표로 이동합니다.
SetDialogControlDpiChangeBehavior 대화 상자에서 자식 창의 기본 모니터별 DPI 크기 조정 동작을 재정의합니다.
SetDialogDpiChangeBehavior Per-Monitor v2 컨텍스트의 대화 상자는 자동으로 DPI 크기 조정됩니다. 이 방법을 사용하면 DPI 변경 동작을 사용자 지정할 수 있습니다.
SetDisplayAutoRotationPreferences 현재 프로세스에 대한 화면 자동 회전 기본 설정을 지정합니다.
SetDisplayConfig SetDisplayConfig 함수는 현재 세션에서 지정된 경로를 단독으로 사용하도록 설정하여 표시 토폴로지, 원본 및 대상 모드를 수정합니다.

SetDlgItemInt 대화 상자에서 컨트롤의 텍스트를 지정된 정수 값의 문자열 표현으로 설정합니다.
SetDlgItemTextA 대화 상자에서 컨트롤의 제목 또는 텍스트를 설정합니다. (ANSI)
SetDlgItemTextW 대화 상자에서 컨트롤의 제목 또는 텍스트를 설정합니다. (유니코드)
SetDoubleClickTime 마우스의 두 번 클릭 시간을 설정합니다.
SetFocus 키보드 포커스를 지정된 창으로 설정합니다. 창은 호출 스레드의 메시지 큐에 연결되어야 합니다.
SetForegroundWindow 지정된 창을 만든 스레드를 포그라운드로 가져오고 창을 활성화합니다.
SetGestureConfig Windows Touch 제스처에 대한 창에서 전송되는 메시지를 구성합니다.
SetKeyboardState 키보드 키 상태 배열을 호출 스레드의 키보드 입력 상태 테이블에 복사합니다. GetKeyboardState 및 GetKeyState 함수에서 액세스하는 동일한 테이블입니다. 이 테이블을 변경해도 다른 스레드에 대한 키보드 입력에는 영향을 주지 않습니다.
SetLastErrorEx 마지막 오류 코드를 설정합니다.
SetLayeredWindowAttributes 계층적 창의 불투명도 및 투명도 색상 키를 설정합니다.
SetMenu 지정된 창에 새 메뉴를 할당합니다.
SetMenuContextHelpId 도움말 컨텍스트 식별자를 메뉴에 연결합니다.

SetMenuDefaultItem 지정된 메뉴의 기본 메뉴 항목을 설정합니다.
SetMenuInfo 지정된 메뉴에 대한 정보를 설정합니다.
SetMenuItemBitmaps 지정된 비트맵을 메뉴 항목과 연결합니다. 메뉴 항목을 선택하든 선택 취소하든 관계없이 시스템은 메뉴 항목 옆에 적절한 비트맵을 표시합니다.
SetMenuItemInfoA 메뉴 항목에 대한 정보를 변경합니다. (ANSI)
SetMenuItemInfoW 메뉴 항목에 대한 정보를 변경합니다. (유니코드)
SetMessageExtraInfo 현재 스레드에 대한 추가 메시지 정보를 설정합니다.
SetParent 지정된 자식 창의 부모 창을 변경합니다.
SetPhysicalCursorPos 실제 좌표에서 커서의 위치를 설정합니다.
SetProcessDefaultLayout 현재 실행 중인 프로세스에 대해서만 부모 또는 소유자 없이 창을 만들 때 기본 레이아웃을 변경합니다.
SetProcessDPIAware SetProcessDPIAware를 변경하거나 사용할 수 없습니다. 대신 SetProcessDPIAwareness를 사용합니다.
SetProcessDpiAwarenessContext 현재 프로세스를 지정된 dpi(인치당 점) 인식 컨텍스트로 설정합니다. DPI 인식 컨텍스트는 DPI_AWARENESS_CONTEXT 값에서 가져옵니다.

SetProcessRestrictionExemption 데스크톱 프로세스가 Windows 스토어 앱 환경과 상호 작용하지 못하도록 하는 제한에서 호출 프로세스를 제외합니다. 이 함수는 개발 및 디버깅 도구에서 사용됩니다.
SetProcessWindowStation 지정된 창 스테이션을 호출 프로세스에 할당합니다.
SetPropA 새 항목을 추가하거나 지정된 창의 속성 목록에서 기존 항목을 변경합니다. (ANSI)
SetPropW 새 항목을 추가하거나 지정된 창의 속성 목록에서 기존 항목을 변경합니다. (유니코드)
SetRect SetRect 함수는 지정된 사각형의 좌표를 설정합니다. 이는 RECT 구조체의 적절한 멤버에 왼쪽, 위쪽, 오른쪽 및 아래쪽 인수를 할당하는 것과 같습니다.
SetRectEmpty SetRectEmpty 함수는 모든 좌표가 0으로 설정된 빈 사각형을 만듭니다.
SetScrollInfo SetScrollInfo 함수는 최소 및 최대 스크롤 위치, 페이지 크기, 스크롤 상자(엄지)의 위치를 포함하여 스크롤 막대의 매개 변수를 설정합니다. 또한 이 함수는 요청된 경우 스크롤 막대를 다시 그립니다.
SetScrollPos SetScrollPos 함수는 지정된 스크롤 막대에서 스크롤 상자(엄지)의 위치를 설정하고 요청 시 스크롤 막대를 다시 그려서 스크롤 상자의 새 위치를 반영합니다.
SetScrollRange SetScrollRange 함수는 지정된 스크롤 막대의 최소 및 최대 스크롤 상자 위치를 설정합니다.
SetSysColors 지정된 표시 요소의 색을 설정합니다.
SetSystemCursor 애플리케이션이 시스템 커서를 사용자 지정할 수 있도록 합니다. id 매개 변수로 지정된 시스템 커서의 내용을 hcur 매개 변수로 지정된 커서의 내용으로 바꾼 다음 hcur를 삭제합니다.

SetThreadCursorCreationScaling 이 스레드에서 생성되는 커서가 의도된 DPI 배율을 설정합니다. 이 값은 표시되는 특정 모니터의 커서 크기를 조정하는 경우 고려됩니다.
SetThreadDesktop 호출 스레드에 지정된 데스크톱을 할당합니다. 데스크톱의 모든 후속 작업은 데스크톱에 부여된 액세스 권한을 사용합니다.
SetThreadDpiAwarenessContext 현재 스레드에 대한 DPI 인식을 제공된 값으로 설정합니다.
SetThreadDpiHostingBehavior 스레드의 DPI_HOSTING_BEHAVIOR 설정합니다. 이 동작을 사용하면 스레드에서 만든 창이 다른 DPI_AWARENESS_CONTEXT 있는 자식 창을 호스트할 수 있습니다.
SetTimer 지정된 시간 제한 값을 사용하여 타이머를 만듭니다.
SetUserObjectInformationA 지정된 창 스테이션 또는 데스크톱 개체에 대한 정보를 설정합니다. (ANSI)
SetUserObjectInformationW 지정된 창 스테이션 또는 데스크톱 개체에 대한 정보를 설정합니다. (유니코드)
SetUserObjectSecurity 사용자 개체의 보안을 설정합니다. 예를 들어 창 또는 DDE 대화일 수 있습니다.
SetWindowContextHelpId 도움말 컨텍스트 식별자를 지정된 창에 연결합니다.
SetWindowDisplayAffinity 표시 선호도 설정을 창과 연결된 hWnd의 커널 모드에 저장합니다.
SetWindowFeedbackSetting 창에 대한 피드백 구성을 설정합니다.

SetWindowLongA 지정된 창의 특성을 변경합니다. 또한 함수는 지정된 오프셋의 32비트(long) 값을 추가 창 메모리로 설정합니다. (ANSI)
SetWindowLongPtrA 지정된 창의 특성을 변경합니다. (ANSI)
SetWindowLongPtrW 지정된 창의 특성을 변경합니다. (유니코드)
SetWindowLongW 지정된 창의 특성을 변경합니다. 또한 함수는 지정된 오프셋의 32비트(long) 값을 추가 창 메모리로 설정합니다. (유니코드)
SetWindowPlacement 표시 상태와 지정된 창의 복원, 최소화 및 최대화된 위치를 설정합니다.
SetWindowPos 자식, 팝업 또는 최상위 창의 크기, 위치 및 Z 순서를 변경합니다. 이러한 창은 화면의 모양에 따라 정렬됩니다. 맨 위 창은 가장 높은 순위를 받고 Z 순서의 첫 번째 창입니다.
SetWindowRgn SetWindowRgn 함수는 창의 창 영역을 설정합니다.
SetWindowsHookExA 애플리케이션 정의 후크 프로시저를 후크 체인에 설치합니다. (ANSI)
SetWindowsHookExW 애플리케이션 정의 후크 프로시저를 후크 체인에 설치합니다. (유니코드)
SetWindowTextA 지정된 창 제목 표시줄의 텍스트를 변경합니다(있는 경우). 지정한 창이 컨트롤이면 컨트롤의 텍스트가 변경됩니다. 그러나 SetWindowText는 다른 애플리케이션에서 컨트롤의 텍스트를 변경할 수 없습니다. (ANSI)

SetWindowTextW 지정된 창 제목 표시줄의 텍스트를 변경합니다(있는 경우). 지정한 창이 컨트롤이면 컨트롤의 텍스트가 변경됩니다. 그러나 SetWindowText는 다른 애플리케이션에서 컨트롤의 텍스트를 변경할 수 없습니다. (유니코드)
SetWinEventHook 이벤트 범위에 대한 이벤트 후크 함수를 설정합니다.
ShowCaret caret의 현재 위치에 있는 화면에 캐리트를 표시합니다. 캐리트가 표시되면 자동으로 깜박이기 시작합니다.
ShowCursor 커서를 표시하거나 숨깁니다. (ShowCursor)
ShowOwnedPopups 지정된 창이 소유한 모든 팝업 창을 표시하거나 숨깁니다.
ShowScrollBar ShowScrollBar 함수는 지정된 스크롤 막대를 표시하거나 숨깁니다.
Showwindow 지정된 창의 표시 상태를 설정합니다.
ShowWindowAsync 작업이 완료 될 때까지 기다리지 않고 창의 표시 상태를 설정합니다.
ShutdownBlockReasonCreate 시스템을 종료할 수 없음을 나타내고 시스템 종료가 시작된 경우 사용자에게 표시할 이유 문자열을 설정합니다.
ShutdownBlockReasonDestroy 시스템을 종료하고 이유 문자열을 해제할 수 있음을 나타냅니다.
ShutdownBlockReasonQuery ShutdownBlockReasonCreate 함수에서 설정한 이유 문자열을 검색합니다.

SkipPointerFrameMessages 지정된 포인터에 대해 가장 최근에 검색된 메시지를 생성한 포인터 입력 프레임을 결정하고 동일한 포인터 입력 프레임에서 생성된 큐에 대기(시도되지 않은) 포인터 입력 메시지를 삭제합니다.
SoundSentry 시각적 신호를 트리거하여 소리가 재생 중임을 나타냅니다.
SubtractRect SubtractRect 함수는 한 사각형을 다른 사각형에서 빼서 형성된 사각형의 좌표를 결정합니다.
SwapMouseButton 마우스 왼쪽 단추와 오른쪽 단추의 의미를 반대로 뒤바꾸거나 복원합니다.
SwitchDesktop 지정된 데스크톱을 표시하고 활성화합니다. 이렇게 하면 데스크톱이 사용자로부터 입력을 받을 수 있습니다.
SwitchToThisWindow 포커스를 지정된 창으로 전환하여 포그라운드로 가져옵니다.
SystemParametersInfoA 시스템 전체 매개 변수 중 하나의 값을 검색하거나 설정합니다. (ANSI)
SystemParametersInfoForDpi 제공된 DPI 값을 고려하여 시스템 전체 매개 변수 중 하나의 값을 검색합니다.
SystemParametersInfoW 시스템 전체 매개 변수 중 하나의 값을 검색하거나 설정합니다. (유니코드)
TabbedTextOutA TabbedTextOut 함수는 지정된 위치에 문자 문자열을 쓰고 탭을 탭 중지 위치 배열에 지정된 값으로 확장합니다. 텍스트는 현재 선택한 글꼴, 배경색 및 텍스트 색으로 작성됩니다. (ANSI)
TabbedTextOutW TabbedTextOut 함수는 지정된 위치에 문자 문자열을 쓰고 탭을 탭 중지 위치 배열에 지정된 값으로 확장합니다. 텍스트는 현재 선택한 글꼴, 배경색 및 텍스트 색으로 작성됩니다. (유니코드)

TileWindows 지정된 부모 창의 지정된 자식 창을 타일로 표시합니다.
ToAscii 지정된 가상 키 코드 및 키보드 상태를 하나 이상의 해당 문자로 변환합니다.
ToAsciiEx 지정된 가상 키 코드 및 키보드 상태를 하나 이상의 해당 문자로 변환합니다. 이 함수는 입력 언어 및 입력 로캘 식별자로 식별된 물리적 키보드 레이아웃을 사용하여 코드를 변환합니다.
TOUCH_COORD_TO_PIXEL 터치 좌표를 픽셀로 변환합니다.
ToUnicode 지정된 가상 키 코드 및 키보드 상태를 하나 이상의 해당 유니코드 문자로 변환합니다. (ToUnicode)
ToUnicodeEx 지정된 가상 키 코드 및 키보드 상태를 하나 이상의 해당 유니코드 문자로 변환합니다. (ToUnicodeEx)
TrackMouseEvent 마우스 포인터가 창을 벗어나거나 지정된 시간 동안 창 위에 있을 때 메시지를 게시합니다.
TrackPopupMenu 지정된 위치에 바로 가기 메뉴를 표시하고 메뉴의 항목 선택을 추적합니다. 바로 가기 메뉴는 화면의 아무 곳에나 나타날 수 있습니다.
TrackPopupMenuEx 지정된 위치에 바로 가기 메뉴를 표시하고 바로 가기 메뉴에서 항목 선택을 추적합니다. 바로 가기 메뉴는 화면의 아무 곳에나 나타날 수 있습니다.
TranslateAcceleratorA 메뉴 명령에 대한 가속기 키를 처리합니다. (ANSI)
TranslateAcceleratorW 메뉴 명령에 대한 가속기 키를 처리합니다. (유니코드)

TranslateMDISysAccel 지정된 MDI 클라이언트 창과 연결된 MDI(다중 문서 인터페이스) 자식 창의 창 메뉴 명령에 대한 가속기 키 입력을 처리합니다.
TranslateMessage 가상 키 메시지를 문자 메시지로 변환합니다. 문자 메시지는 다음에 스레드가 GetMessage 또는 PeekMessage 함수를 호출할 때 읽을 수 있도록 호출 스레드의 메시지 큐에 게시됩니다.
UnhookWindowsHookEx SetWindowsHookEx 함수에 의해 후크 체인에 설치된 후크 프로시저를 제거합니다.
UnhookWinEvent SetWinEventHook에 대한 이전 호출에서 만든 이벤트 후크 함수를 제거합니다.
UnionRect UnionRect 함수는 두 사각형의 공용 구조체를 만듭니다. 공용 구조체는 두 원본 사각형을 모두 포함하는 가장 작은 사각형입니다.
UnloadKeyboardLayout 입력 로캘 식별자(이전의 키보드 레이아웃)를 언로드합니다.
UnregisterClassA 창 클래스의 등록을 취소하여 클래스에 필요한 메모리를 해제합니다. (ANSI)
UnregisterClassW 창 클래스의 등록을 취소하여 클래스에 필요한 메모리를 해제합니다. (유니코드)
UnregisterDeviceNotification 지정된 디바이스 알림 핸들을 닫습니다.
UnregisterHotKey 호출 스레드에 의해 이전에 등록된 핫 키를 해제합니다.
UnregisterPointerInputTarget 호출자가 지정된 형식의 모든 포인터 입력이 리디렉션되는 대상 창의 등록을 취소할 수 있습니다.

UnregisterPointerInputTargetEx UnregisterPointerInputTargetEx를 변경하거나 사용할 수 없습니다. 대신 UnregisterPointerInputTarget을 사용합니다.
UnregisterPowerSettingNotification 전원 설정 알림을 등록 취소합니다.
UnregisterSuspendResumeNotification 시스템이 일시 중단되거나 다시 시작될 때 알림을 받을 등록을 취소합니다. PowerUnregisterSuspendResumeNotification과 유사하지만 사용자 모드에서 작동합니다.
UnregisterTouchWindow 창을 더 이상 터치 가능하지 않은 것으로 등록합니다.
UpdateLayeredWindow 계층적 창의 위치, 크기, 모양, 내용 및 투명도를 업데이트합니다.
UpdateWindow UpdateWindow 함수는 창의 업데이트 영역이 비어 있지 않은 경우 창에 WM_PAINT 메시지를 보내 지정된 창의 클라이언트 영역을 업데이트합니다.
UserHandleGrantAccess 사용자 인터페이스 제한이 있는 작업에 User 개체에 대한 핸들에 대한 액세스 권한을 부여하거나 거부합니다.
ValidateRect ValidateRect 함수는 지정된 창의 업데이트 영역에서 사각형을 제거하여 사각형 내의 클라이언트 영역의 유효성을 검사합니다.
ValidateRgn ValidateRgn 함수는 지정된 창의 현재 업데이트 영역에서 지역을 제거하여 지역 내의 클라이언트 영역의 유효성을 검사합니다.
VkKeyScanA 문자를 현재 키보드의 해당 가상 키 코드 및 시프트 상태로 변환합니다. (ANSI)

VkKeyScanExA 문자를 해당 가상 키 코드 및 시프트 상태로 변환합니다. 이 함수는 입력 언어 및 입력 로캘 식별자로 식별된 물리적 키보드 레이아웃을 사용하여 문자를 변환합니다. (ANSI)
VkKeyScanExW 문자를 해당 가상 키 코드 및 시프트 상태로 변환합니다. 이 함수는 입력 언어 및 입력 로캘 식별자로 식별된 물리적 키보드 레이아웃을 사용하여 문자를 변환합니다. (유니코드)
VkKeyScanW 문자를 현재 키보드의 해당 가상 키 코드 및 시프트 상태로 변환합니다. (유니코드)
WaitForInputIdle 지정된 프로세스가 초기 입력 처리를 완료하고 입력이 보류 중이 아니거나 제한 시간 간격이 경과할 때까지 사용자 입력을 대기할 때까지 기다립니다.
WaitMessage 스레드에 메시지 큐에 다른 메시지가 없는 경우 다른 스레드에 대한 제어를 생성합니다. WaitMessage 함수는 스레드를 일시 중단하고 새 메시지가 스레드의 메시지 큐에 배치될 때까지 반환되지 않습니다.
WindowFromDC WindowFromDC 함수는 지정된 DC(디스플레이 디바이스 컨텍스트)와 연결된 창에 대한 핸들을 반환합니다. 지정된 디바이스 컨텍스트를 사용하는 출력 함수는 이 창에 그립니다.
WindowFromPhysicalPoint 지정된 물리적 지점이 포함된 창에 대한 핸들을 검색합니다.
WindowFromPoint 지정된 점을 포함하는 창에 대한 핸들을 검색합니다.
WinHelpA Windows 도움말(Winhelp.exe)을 시작하고 애플리케이션에서 요청한 도움말의 특성을 나타내는 추가 데이터를 전달합니다. (ANSI)
WinHelpW Windows 도움말(Winhelp.exe)을 시작하고 애플리케이션에서 요청한 도움말의 특성을 나타내는 추가 데이터를 전달합니다. (유니코드)

wsprintfA 지정된 버퍼에 서식이 지정된 데이터를 씁니다. (ANSI)
wsprintfW 지정된 버퍼에 서식이 지정된 데이터를 씁니다. (유니코드)
wvsprintfA 인수 목록에 대한 포인터를 사용하여 지정된 버퍼에 서식이 지정된 데이터를 씁니다. (ANSI)
wvsprintfW 인수 목록에 대한 포인터를 사용하여 지정된 버퍼에 서식이 지정된 데이터를 씁니다. (유니코드)

콜백 함수

DLGPROC CreateDialog 및 DialogBox 함수 패밀리와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
DRAWSTATEPROC DrawStateProc 함수는 DrawState 함수에 대한 복잡한 이미지를 렌더링하는 애플리케이션 정의 콜백 함수입니다.
EDITWORDBREAKPROCA EM_SETWORDBREAKPROC 메시지와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (ANSI)
EDITWORDBREAKPROCW EM_SETWORDBREAKPROC 메시지와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (유니코드)
GRAYSTRINGPROC OutputProc 함수는 GrayString 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
HOOKPROC SetWindowsHookEx 함수와 함께 사용되는 애플리케이션 정의 또는 라이브러리 정의 콜백 함수입니다. 시스템은 SendMessage 함수가 호출된 후 이 함수를 호출합니다. 후크 프로시저는 메시지를 검사할 수 있습니다. 수정할 수 없습니다.

MONITORENUMPROC MonitorEnumProc 함수는 EnumDisplayMonitors 함수에서 호출되는 애플리케이션 정의 콜백 함수입니다.
MSGBOXCALLBACK 애플리케이션에서 정의한 콜백 함수는 메시지 상자에 대한 도움말 이벤트를 처리합니다.
PROPENUMPROCA EnumProps 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (ANSI)
PROPENUMPROCEXA EnumPropsEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (ANSI)
PROPENUMPROCEXW EnumPropsEx 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (유니코드)
PROPENUMPROCW EnumProps 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다. (유니코드)
SENDASYNCPROC SendMessageCallback 함수와 함께 사용되는 애플리케이션 정의 콜백 함수입니다.
TIMERPROC WM_TIMER 메시지를 처리하는 애플리케이션 정의 콜백 함수입니다. TIMERPROC 형식은 이 콜백 함수에 대한 포인터를 정의합니다. TimerProc은 애플리케이션 정의 함수 이름의 자리 표시자입니다.
WINEVENTPROC 액세스 가능한 개체에서 생성된 이벤트에 대한 응답으로 시스템이 호출하는 애플리케이션 정의 콜백(또는 후크) 함수입니다.
Wndproc 애플리케이션에서 정의한 콜백 함수는 창으로 전송된 메시지를 처리합니다.

구조체

가속 가속기 테이블에 사용되는 가속기 키를 정의합니다.
ACCESSTIMEOUT 접근성 기능과 관련된 제한 시간에 대한 정보를 포함합니다.
ALTTABINFO 애플리케이션 전환(ALT+TAB) 창에 대한 상태 정보를 포함합니다.
ANIMATIONINFO 사용자 작업과 관련된 애니메이션 효과에 대해 설명합니다.
AUDIODESCRIPTION 오디오 설명과 관련된 정보를 포함합니다. 이 구조체는 SPI_GETAUDIODESCRIPTION 또는 SPI_SETAUDIODESCRIPTION 작업 값을 지정할 때 SystemParametersInfo 함수와 함께 사용됩니다.
BSMINFO BroadcastSystemMessageEx의 요청을 거부한 창에 대한 정보를 포함합니다.
CBT_CREATEWNDA 창을 만들기 전에 WH_CBT 후크 프로시저 CBTProc에 전달된 정보를 포함합니다. (ANSI)
CBT_CREATEWNDW 창을 만들기 전에 WH_CBT 후크 프로시저 CBTProc에 전달된 정보를 포함합니다. (유니코드)
CBTACTIVATESTRUCT 창이 활성화되기 전에 WH_CBT 후크 프로시저 CBTProc에 전달된 정보를 포함합니다.
CHANGEFILTERSTRUCT ChangeWindowMessageFilterEx 함수를 호출하여 얻은 확장된 결과 정보를 포함합니다.
CLIENTCREATESTRUCT MDI 클라이언트 창의 메뉴 및 첫 번째 MDI(다중 문서 인터페이스) 자식 창에 대한 정보를 포함합니다.
COMBOBOXINFO 콤보 상자 상태 정보를 포함합니다.

COMPAREITEMSTRUCT 소유자가 그린 정렬된 목록 상자 또는 콤보 상자에서 두 항목에 대한 식별자 및 애플리케이션 제공 데이터를 제공합니다.
COPYDATASTRUCT WM_COPYDATA 메시지로 다른 애플리케이션에 전달할 데이터를 포함합니다.
CREATESTRUCTA 애플리케이션의 창 프로시저에 전달되는 초기화 매개 변수를 정의합니다. 이러한 멤버는 CreateWindowEx 함수의 매개 변수와 동일합니다. (ANSI)
CREATESTRUCTW 애플리케이션의 창 프로시저에 전달되는 초기화 매개 변수를 정의합니다. 이러한 멤버는 CreateWindowEx 함수의 매개 변수와 동일합니다. (유니코드)
CURSORINFO 전역 커서 정보를 포함합니다.
커서 셰이프 커서에 대한 정보를 포함합니다.
CWPRETSTRUCT WH_CALLWNDPROCRET 후크 프로시저 CallWndRetProc에 전달되는 메시지 매개 변수를 정의합니다.
CWPSTRUCT WH_CALLWNDPROC 후크 프로시저 CallWndProc에 전달되는 메시지 매개 변수를 정의합니다.
DEBUGHOOKINFO WH_DEBUG 후크 프로시저인 DebugProc에 전달되는 디버깅 정보를 포함합니다.
DELETEITEMSTRUCT 삭제된 목록 상자 또는 콤보 상자 항목을 설명합니다.
DLGITEMTEMPLATE 대화 상자에서 컨트롤의 차원과 스타일을 정의합니다. 이러한 구조체 중 하나 이상이 DLGTEMPLATE 구조체와 결합되어 대화 상자의 표준 템플릿을 형성합니다.

DLGTEMPLATE 대화 상자의 차원과 스타일을 정의합니다.
DRAWITEMSTRUCT 소유자 창에서 소유자 그리기 컨트롤 또는 메뉴 항목을 그리는 방법을 결정하는 데 사용하는 정보를 제공합니다.
DRAWTEXTPARAMS DRAWTEXTPARAMS 구조체에는 DrawTextEx 함수에 대한 확장 서식 옵션이 포함되어 있습니다.
EVENTMSG 시스템 메시지 큐로 전송된 하드웨어 메시지에 대한 정보를 포함합니다. 이 구조는 JournalPlaybackProc 콜백 함수에 대한 메시지 정보를 저장하는 데 사용됩니다.
FILTERKEYS 장애가 있는 사용자가 키보드 반복 속도(RepeatKeys), 수용 지연(SlowKeys) 및 바운스 속도(BounceKeys)를 설정할 수 있도록 하는 FilterKeys 접근성 기능에 대한 정보를 포함합니다.
FLASHWINFO 창에 대한 플래시 상태 및 시스템이 창을 플래시해야 하는 횟수를 포함합니다.
GESTURECONFIG 제스처 메시지 및 이 구성의 유형을 사용하도록 설정하기 위한 구성을 가져오고 설정합니다.
GESTUREINFO 제스처에 대한 정보를 저장합니다.
GESTURENOTIFYSTRUCT WM_GESTURENOTIFY 메시지와 함께 전송되는 경우 제스처에 대한 정보를 전달합니다.
GUITHREADINFO GUI 스레드에 대한 정보를 포함합니다.
HARDWAREINPUT 키보드나 마우스 외에 입력 디바이스에서 생성된 시뮬레이션된 메시지에 대한 정보를 포함합니다.

HELPINFO 상황에 맞는 도움말이 요청된 항목에 대한 정보를 포함합니다.
HELPWININFOA 기본 또는 보조 도움말 창의 크기와 위치를 포함합니다. 애플리케이션은 HELP_SETWINPOS 값으로 WinHelp 함수를 호출하여 이 정보를 설정할 수 있습니다. (ANSI)
HELPWININFOW 기본 또는 보조 도움말 창의 크기와 위치를 포함합니다. 애플리케이션은 HELP_SETWINPOS 값으로 WinHelp 함수를 호출하여 이 정보를 설정할 수 있습니다. (유니코드)
HIGHCONTRASTA 고대비 접근성 기능에 대한 정보를 포함합니다. (ANSI)
HIGHCONTRASTW 고대비 접근성 기능에 대한 정보를 포함합니다. (유니코드)
ICONINFO 아이콘 또는 커서에 대한 정보를 포함합니다.
ICONINFOEXA 아이콘 또는 커서에 대한 정보를 포함합니다. ICONINFO를 확장합니다. GetIconInfoEx에서 사용됩니다. (ANSI)
ICONINFOEXW 아이콘 또는 커서에 대한 정보를 포함합니다. ICONINFO를 확장합니다. GetIconInfoEx에서 사용됩니다. (유니코드)
ICONMETRICSA 아이콘과 연결된 확장 가능한 메트릭을 포함합니다. 이 구조체는 SPI_GETICONMETRICS 또는 SPI_SETICONMETRICS 작업을 지정할 때 SystemParametersInfo 함수와 함께 사용됩니다. (ANSI)
ICONMETRICSW 아이콘과 연결된 확장 가능한 메트릭을 포함합니다. 이 구조체는 SPI_GETICONMETRICS 또는 SPI_SETICONMETRICS 작업을 지정할 때 SystemParametersInfo 함수와 함께 사용됩니다. (유니코드)

INPUT SendInput에서 키 입력, 마우스 이동 및 마우스 클릭과 같은 입력 이벤트를 합성하기 위한 정보를 저장하는 데 사용됩니다.
INPUT_INJECTION_VALUE 입력 삽입 세부 정보를 포함합니다.
INPUT_MESSAGE_SOURCE 입력 메시지의 원본에 대한 정보를 포함합니다.
INPUT_TRANSFORM 메시지 소비자의 변환을 나타내는 행렬을 정의합니다.
KBDLLHOOKSTRUCT 하위 수준의 키보드 입력 이벤트에 대한 정보를 포함합니다.
KEYBDINPUT 시뮬레이션된 키보드 이벤트에 대한 정보를 포함합니다.
LASTINPUTINFO 마지막 입력 시간을 포함합니다.
MDICREATESTRUCTA MDI(다중 문서 인터페이스) 자식 창의 클래스, 제목, 소유자, 위치 및 크기에 대한 정보를 포함합니다. (ANSI)
MDICREATESTRUCTW MDI(다중 문서 인터페이스) 자식 창의 클래스, 제목, 소유자, 위치 및 크기에 대한 정보를 포함합니다. (유니코드)
MDINEXTMENU 활성화할 메뉴에 대한 정보를 포함합니다.
MEASUREITEMSTRUCT 소유자가 그린 컨트롤 또는 메뉴 항목의 차원을 시스템에 알립니다. 이렇게 하면 시스템이 컨트롤과 사용자의 상호 작용을 올바르게 처리할 수 있습니다.

MENUBARINFO 메뉴 모음 정보를 포함합니다.
MENUGETOBJECTINFO 마우스 커서가 있는 메뉴에 대한 정보를 포함합니다.
MENUINFO 메뉴에 대한 정보를 포함합니다.
MENUITEMINFOA 메뉴 항목에 대한 정보를 포함합니다. (MENUITEMINFOA)
MENUITEMINFOW 메뉴 항목에 대한 정보를 포함합니다. (MENUITEMINFOW)
MENUITEMTEMPLATE 메뉴 템플릿에서 메뉴 항목을 정의합니다.
MENUITEMTEMPLATEHEADER 메뉴 템플릿의 헤더를 정의합니다. 전체 메뉴 템플릿은 헤더와 하나 이상의 메뉴 항목 목록으로 구성됩니다.
MINIMIZEDMETRICS 최소화된 창과 연결된 확장 가능한 메트릭을 포함합니다.
MINMAXINFO 창의 최대 크기 및 위치와 최소 및 최대 추적 크기에 대한 정보를 포함합니다.
MONITORINFO MONITORINFO 구조에는 디스플레이 모니터에 대한 정보가 포함되어 있습니다. GetMonitorInfo 함수는 MONITORINFO 구조 또는 MONITORINFOEX 구조체에 정보를 저장합니다. MONITORINFO 구조체는 MONITORINFOEX 구조체의 하위 집합입니다.
MONITORINFOEXA MONITORINFOEX 구조에는 디스플레이 모니터에 대한 정보가 포함되어 있습니다. GetMonitorInfo 함수는 정보를 MONITORINFOEX 구조 또는 MONITORINFO 구조체에 저장합니다. MONITORINFOEX 구조체는 MONITORINFO 구조체의 상위 집합입니다. (ANSI)

MONITORINFOEXW MONITORINFOEX 구조에는 디스플레이 모니터에 대한 정보가 포함되어 있습니다. GetMonitorInfo 함수는 정보를 MONITORINFOEX 구조 또는 MONITORINFO 구조체에 저장합니다. MONITORINFOEX 구조체는 MONITORINFO 구조체의 상위 집합입니다. (유니코드)
MOUSEHOOKSTRUCT WH_MOUSE 후크 프로시저 MouseProc에 전달된 마우스 이벤트에 대한 정보를 포함합니다.
MOUSEHOOKSTRUCTEX WH_MOUSE 후크 프로시저 MouseProc에 전달된 마우스 이벤트에 대한 정보를 포함합니다. 휠 이동 또는 X 단추 사용에 대한 정보를 포함하는 MOUSEHOOKSTRUCT 구조체의 확장입니다.
MOUSEINPUT 시뮬레이션된 마우스 이벤트에 대한 정보를 포함합니다.
마우스 MouseKeys 접근성 기능에 대한 정보를 포함합니다.
MOUSEMOVEPOINT 화면 좌표에서 마우스의 위치에 대한 정보를 포함합니다.
메시지 스레드 메시지 큐의 메시지 정보를 포함합니다.
MSGBOXPARAMSA 메시지 상자를 표시하는 데 사용되는 정보를 포함합니다. MessageBoxIndirect 함수는 이 구조를 사용합니다. (ANSI)
MSGBOXPARAMSW 메시지 상자를 표시하는 데 사용되는 정보를 포함합니다. MessageBoxIndirect 함수는 이 구조를 사용합니다. (유니코드)
MSLLHOOKSTRUCT 하위 수준 마우스 입력 이벤트에 대한 정보를 포함합니다.
MULTIKEYHELPA 검색할 키워드(keyword) Windows 도움말에서 검색할 키워드(keyword) 테이블을 지정합니다. (ANSI)

MULTIKEYHELPW 검색할 키워드(keyword) Windows 도움말에서 검색할 키워드(keyword) 테이블을 지정합니다. (유니코드)
NCCALCSIZE_PARAMS WM_NCCALCSIZE 메시지를 처리하는 동안 애플리케이션이 창의 클라이언트 영역의 크기, 위치 및 유효한 내용을 계산하는 데 사용할 수 있는 정보를 포함합니다.
NMHDR NMHDR 구조에는 알림 메시지에 대한 정보가 포함되어 있습니다. (NMHDR 구조체)
NONCLIENTMETRICSA 비정상 창의 비클라이언트 영역과 연결된 확장 가능한 메트릭을 포함합니다. (ANSI)
NONCLIENTMETRICSW 비정상 창의 비클라이언트 영역과 연결된 확장 가능한 메트릭을 포함합니다. (유니코드)
PAINTSTRUCT PAINTSTRUCT 구조체에는 애플리케이션에 대한 정보가 포함되어 있습니다. 이 정보는 해당 애플리케이션이 소유한 창의 클라이언트 영역을 그리는 데 사용할 수 있습니다.
POINTER_DEVICE_CURSOR_INFO 포인터 디바이스에 대한 커서 ID 매핑을 포함합니다.
POINTER_DEVICE_INFO 포인터 디바이스에 대한 정보를 포함합니다. 이러한 구조체의 배열은 GetPointerDevices 함수에서 반환됩니다. GetPointerDevice 함수에 대한 호출에서 단일 구조체가 반환됩니다.
POINTER_DEVICE_PROPERTY 포인터 기반 디바이스 속성(HID 사용량에 해당하는 HID(휴먼 인터페이스 디바이스) 전역 항목)을 포함합니다.
POINTER_INFO 모든 포인터 형식에 공통적인 기본 포인터 정보를 포함합니다. 애플리케이션은 GetPointerInfo, GetPointerFrameInfo, GetPointerInfoHistory 및 GetPointerFrameInfoHistory 함수를 사용하여 이 정보를 검색할 수 있습니다.

POINTER_PEN_INFO 모든 포인터 형식에 공통적인 기본 펜 정보를 정의합니다.
POINTER_TOUCH_INFO 모든 포인터 형식에 공통적인 기본 터치 정보를 정의합니다.
POINTER_TYPE_INFO 포인터 입력 형식에 대한 정보를 포함합니다.
POWERBROADCAST_SETTING 전원 설정 이벤트와 함께 전송되며 특정 변경 내용에 대한 데이터를 포함합니다.
RAWHID HID(휴먼 인터페이스 디바이스)의 원시 입력 형식을 설명합니다.
RAWINPUT 디바이스의 원시 입력을 포함합니다.
RAWINPUTDEVICE 원시 입력 디바이스에 대한 정보를 정의합니다.
RAWINPUTDEVICELIST 원시 입력 디바이스에 대한 정보를 포함합니다.
RAWINPUTHEADER 원시 입력 데이터의 일부인 헤더 정보를 포함합니다.
RAWKEYBOARD 키보드의 상태에 대한 정보를 포함합니다.
RAWMOUSE 마우스의 상태에 대한 정보를 포함합니다.
RID_DEVICE_INFO 모든 디바이스에서 들어오는 원시 입력 데이터를 정의합니다.

RID_DEVICE_INFO_HID 지정된 HID(휴먼 인터페이스 디바이스)에서 들어오는 원시 입력 데이터를 정의합니다.
RID_DEVICE_INFO_KEYBOARD 지정된 키보드에서 들어오는 원시 입력 데이터를 정의합니다.
RID_DEVICE_INFO_MOUSE 지정된 마우스에서 들어오는 원시 입력 데이터를 정의합니다.
SCROLLBARINFO SCROLLBARINFO 구조체에는 스크롤 막대 정보가 포함되어 있습니다.
SCROLLINFO SCROLLINFO 구조체에는 SetScrollInfo 함수(또는 SBM_SETSCROLLINFO 메시지)가 설정하거나 GetScrollInfo 함수(또는 SBM_GETSCROLLINFO 메시지)가 검색한 스크롤 막대 매개 변수가 포함되어 있습니다.
SERIALKEYSA 직렬 포트에 연결된 통신 지원의 데이터를 시스템에서 키보드 및 마우스 입력을 시뮬레이션하는 명령으로 해석하는 SerialKeys 접근성 기능에 대한 정보를 포함합니다. (ANSI)
SERIALKEYSW 직렬 포트에 연결된 통신 지원의 데이터를 시스템에서 키보드 및 마우스 입력을 시뮬레이션하는 명령으로 해석하는 SerialKeys 접근성 기능에 대한 정보를 포함합니다. (유니코드)
SOUNDSENTRYA SoundSentry 접근성 기능에 대한 정보를 포함합니다. SoundSentry 기능이 켜지면 컴퓨터는 소리가 생성될 때만 시각적 표시를 표시합니다. (ANSI)
SOUNDSENTRYW SoundSentry 접근성 기능에 대한 정보를 포함합니다. SoundSentry 기능이 켜지면 컴퓨터는 소리가 생성될 때만 시각적 표시를 표시합니다. (유니코드)
고정키 StickyKeys 접근성 기능에 대한 정보를 포함합니다.
STYLESTRUCT 창에 대한 스타일을 포함합니다.

TITLEBARINFO 제목 표시줄 정보를 포함합니다.
TITLEBARINFOEX 제목 표시줄의 각 요소의 좌표를 포함하여 TITLEBARINFO 구조에 설명된 정보를 확장합니다.
토글키 ToggleKeys 접근성 기능에 대한 정보를 포함합니다.
TOUCH_HIT_TESTING_INPUT 터치 디지타이저에서 보고한 터치 접촉 영역에 대한 정보를 포함합니다.
TOUCH_HIT_TESTING_PROXIMITY_EVALUATION 터치 접촉 영역을 교차하는 다른 개체를 기준으로 개체가 터치 접촉 영역의 대상인지 여부를 나타내는 적중 테스트 점수를 포함합니다.
TOUCHINPUT 터치 입력에 대한 데이터를 캡슐화합니다.
TOUCHPREDICTIONPARAMETERS 터치 대상을 예측하고 거리 및 속도 데이터를 포함하는 터치 및 제스처 입력을 처리할 때 하드웨어 대기 시간을 보정하는 데 사용할 수 있는 하드웨어 입력 세부 정보가 포함되어 있습니다.
TPMPARAMS TrackPopupMenuEx 함수에 대한 확장 매개 변수를 포함합니다.
TRACKMOUSEEVENT 마우스 포인터가 창을 벗어나거나 지정된 시간 동안 창 위를 가리키는 경우를 TrackMouseEvent 함수가 추적하는 데 사용됩니다.
UPDATELAYEREDWINDOWINFO UpdateLayeredWindowIndirect에서 계층화된 창에 대한 위치, 크기, 모양, 콘텐츠 및 반투명 정보를 제공하는 데 사용됩니다.
USAGE_PROPERTIES 모든 유형의 HID 입력 디바이스에 대한 디바이스 속성(HID 사용량에 해당하는 HID(휴먼 인터페이스 디바이스) 전역 항목)을 포함합니다.

USEROBJECTFLAGS 창 스테이션 또는 데스크톱 핸들에 대한 정보를 포함합니다.
WINDOWINFO 창 정보를 포함합니다.
창 위치 화면에 창의 배치에 대한 정보를 포함합니다.
WINDOWPOS 창의 크기와 위치에 대한 정보를 포함합니다.
WNDCLASSA RegisterClass 함수에 의해 등록된 창 클래스 특성을 포함합니다. (ANSI)
WNDCLASSEXA 창 클래스 정보를 포함합니다. (ANSI)
WNDCLASSEXW 창 클래스 정보를 포함합니다. (유니코드)
WNDCLASSW RegisterClass 함수에 의해 등록된 창 클래스 특성을 포함합니다. (유니코드)
WTSSESSION_NOTIFICATION 세션 변경 알림에 대한 정보를 제공합니다. 서비스는 세션 변경 이벤트에 대한 응답으로 HandlerEx 함수에서 이 구조를 받습니다.

열거형

AR_STATE 시스템의 화면 자동 회전 상태를 나타냅니다. 예를 들어 자동 회전이 지원되는지 여부 및 사용자가 사용하도록 설정할지 여부가 있습니다.

DIALOG_CONTROL_DPI_CHANGE_BEHAVIORS 대화 상자 내의 자식 창에 대한 모니터별 DPI 크기 조정 동작 재정의에 대해 설명합니다. 이 열거형의 값은 비트 필드이며 결합할 수 있습니다.
DIALOG_DPI_CHANGE_BEHAVIORS 모니터별 v2 컨텍스트에서 대화 상자는 자체 크기를 조정하고 자식 창의 위치를 다시 계산하여 DPI 변경에 자동으로 응답합니다(여기서는 다시 레이아웃이라고 함).
FEEDBACK_TYPE 이벤트와 연결된 시각적 피드백을 지정합니다.
INPUT_MESSAGE_DEVICE_TYPE 입력 메시지를 보낸 디바이스의 유형입니다.
INPUT_MESSAGE_ORIGIN_ID 입력 메시지 원본의 ID입니다.
ORIENTATION_PREFERENCE 데스크톱 앱 프로세스에 대한 화면 방향 기본 설정을 나타냅니다.
POINTER_BUTTON_CHANGE_TYPE 포인터와 연결된 단추의 상태 변경을 식별합니다.
POINTER_DEVICE_CURSOR_TYPE 포인터 디바이스 커서 유형을 식별합니다.
POINTER_DEVICE_TYPE 포인터 디바이스 유형을 식별합니다.
POINTER_FEEDBACK_MODE CreateSyntheticPointerDevice에 사용할 수 있는 시각적 피드백 동작을 식별합니다.
tagPOINTER_INPUT_TYPE 포인터 입력 형식을 식별합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winver.h 헤더

아티클2023. 08. 28.

이 헤더는 메뉴 및 기타 리소스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 메뉴 및 기타 리소스

winver.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

GetFileVersionInfoA 지정된 파일의 버전 정보를 검색합니다. (GetFileVersionInfoA)
GetFileVersionInfoExA 지정된 파일의 버전 정보를 검색합니다. (GetFileVersionInfoExA)
GetFileVersionInfoExW 지정된 파일의 버전 정보를 검색합니다. (GetFileVersionInfoExW)
GetFileVersionInfoSizeA 운영 체제가 지정된 파일에 대한 버전 정보를 검색할 수 있는지 여부를 결정합니다. 버전 정보를 사용할 수 있는 경우 GetFileVersionInfoSize는 해당 정보의 크기(바이트)를 반환합니다. (ANSI)
GetFileVersionInfoSizeExA 운영 체제가 지정된 파일에 대한 버전 정보를 검색할 수 있는지 여부를 결정합니다. 버전 정보를 사용할 수 있는 경우 GetFileVersionInfoSizeEx는 해당 정보의 크기를 바이트 단위로 반환합니다. (ANSI)
GetFileVersionInfoSizeExW 운영 체제가 지정된 파일에 대한 버전 정보를 검색할 수 있는지 여부를 결정합니다. 버전 정보를 사용할 수 있는 경우 GetFileVersionInfoSizeEx는 해당 정보의 크기를 바이트 단위로 반환합니다. (유니코드)
GetFileVersionInfoSizeW 운영 체제가 지정된 파일에 대한 버전 정보를 검색할 수 있는지 여부를 결정합니다. 버전 정보를 사용할 수 있는 경우 GetFileVersionInfoSize는 해당 정보의 크기(바이트)를 반환합니다. (유니코드)

GetFileVersionInfoW 지정된 파일의 버전 정보를 검색합니다. (GetFileVersionInfoW)
VerFindFileA 시스템에서 파일의 다른 버전을 찾는지 여부에 따라 파일을 설치할 위치를 결정합니다. 지정된 버퍼에서 VerFindFile이 반환하는 값은 VerInstallFile 함수에 대한 후속 호출에 사용됩니다. (ANSI)
VerFindFileW 시스템에서 파일의 다른 버전을 찾는지 여부에 따라 파일을 설치할 위치를 결정합니다. 지정된 버퍼에서 VerFindFile이 반환하는 값은 VerInstallFile 함수에 대한 후속 호출에 사용됩니다. (유니코드)
VerInstallFileA VerFindFile 함수에서 반환된 정보에 따라 지정된 파일을 설치합니다. VerInstallFile은 필요한 경우 파일을 압축 해제하고, 고유한 파일 이름을 할당하고, 오래된 파일과 같은 오류를 확인합니다. (ANSI)
VerInstallFileW VerFindFile 함수에서 반환된 정보에 따라 지정된 파일을 설치합니다. VerInstallFile은 필요한 경우 파일을 압축 해제하고, 고유한 파일 이름을 할당하고, 오래된 파일과 같은 오류를 확인합니다. (유니코드)
VerLanguageNameA 지정된 이진 Microsoft 언어 식별자와 연결된 언어에 대한 설명 문자열을 검색합니다. (ANSI)
VerLanguageNameW 지정된 이진 Microsoft 언어 식별자와 연결된 언어에 대한 설명 문자열을 검색합니다. (유니코드)
VerQueryValueA 지정된 버전 정보 리소스에서 지정된 버전 정보를 검색합니다. (ANSI)
VerQueryValueW 지정된 버전 정보 리소스에서 지정된 버전 정보를 검색합니다. (유니코드)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

winwlx.h 헤더

아티클2023. 08. 28.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

winwlx.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WlxActivateUserShell 사용자 셸 프로그램을 활성화합니다.
WlxDisconnectNotify Winlogon은 터미널 서비스 네트워크 세션의 연결이 끊어지면 이 함수를 호출합니다.
WlxDisplayLockedNotice GINA에서 워크스테이션을 잠근 사람 및 잠금 시기와 같은 잠금에 대한 정보를 표시할 수 있습니다.
WlxDisplaySASNotice Winlogon은 사용자가 로그온되지 않은 경우 이 함수를 호출합니다.
WlxDisplayStatusMessage Winlogon은 GINA DLL이 메시지를 표시해야 하는 경우 이 함수를 호출합니다.
WlxGetConsoleSwitchCredentials Winlogon은 이 함수를 호출하여 현재 로그온한 사용자의 자격 증명을 읽어 대상 세션으로 투명하게 전송합니다.
WlxGetStatusMessage Winlogon은 이 함수를 호출하여 GINA DLL에서 표시되는 상태 메시지를 가져옵니다.
WlxInitialize Winlogon은 컴퓨터에 있는 각 창 스테이션에 대해 이 함수를 한 번 호출합니다. 현재 운영 체제는 워크스테이션당 하나의 창 스테이션을 지원합니다.

WlxIsLockOk Winlogon은 워크스테이션을 잠그기 전에 이 함수를 호출합니다.
WlxIsLogoffOk Winlogon은 사용자가 로그오프 작업을 시작할 때 이 함수를 호출합니다.
WlxLoggedOnSAS Winlogon은 사용자가 로그온되고 워크스테이션이 잠겨 있지 않은 동안 SAS(보안 주의 순서) 이벤트를 수신할 때 이 함수를 호출합니다.
WlxLoggedOutSAS Winlogon은 사용자가 로그온하지 않는 동안 SAS(보안 주의 순서) 이벤트를 수신할 때 이 함수를 호출합니다.
WlxLogoff Winlogon은 이 함수를 호출하여 GINA에 이 워크스테이션의 로그오프 작업을 알리고 GINA가 필요할 수 있는 모든 로그오프 작업을 수행할 수 있도록 합니다.
WlxNegotiate WlxNegotiate 함수는 대체 GINA DLL에 의해 구현되어야 합니다. Winlogon에서 GINA DLL에 대한 첫 번째 호출입니다. WlxNegotiate를 사용하면 GINA가 설치된 Winlogon 버전을 지원하는지 확인할 수 있습니다.
WlxNetworkProviderLoad Winlogon은 이 함수를 호출하여 유효한 인증 및 식별 정보를 수집합니다.
WlxReconnectNotify Winlogon은 터미널 서비스 네트워크 세션이 다시 연결될 때 이 함수를 호출합니다.
WlxRemoveStatusMessage Winlogon은 이 함수를 호출하여 GINA DLL에 상태 메시지 표시를 중지하도록 지시합니다.
WlxScreenSaverNotify Winlogon은 화면 보호기가 활성화되기 직전에 이 함수를 호출하여 GINA가 화면 보호기 프로그램과 상호 작용할 수 있도록 합니다.
WlxShutdown

Winlogon은 종료 직전에 이 함수를 호출하여 GINA가 판독기에서 스마트 카드 꺼내는 것과 같은 모든 종료 작업을 수행할 수 있도록 합니다.
WlxStartApplication Winlogon은 시스템에서 사용자의 컨텍스트에서 애플리케이션을 시작해야 할 때 이 함수를 호출합니다.
WlxWkstaLockedSAS Winlogon은 SAS(보안 주의 순서)를 수신하고 워크스테이션이 잠겨 있을 때 이 함수를 호출합니다.

콜백 함수

PWLX_ASSIGN_SHELL_PROTECTION 새로 로그온한 사용자의 셸 프로그램에 보호를 할당하기 위해 GINA에서 호출합니다.
PWLX_CHANGE_PASSWORD_NOTIFY GINA가 암호를 변경했음을 나타내기 위해 호출됩니다.
PWLX_CHANGE_PASSWORD_NOTIFY_EX GINA에서 호출하여 특정 네트워크 공급자(또는 모든 네트워크 공급자)에게 암호가 변경되었음을 알립니다.
PWLX_CLOSE_USER_DESKTOP GINA에서 호출하여 대체 사용자 데스크톱을 닫고 데스크톱이 닫힌 후 클린.
PWLX_CREATE_USER_DESKTOP GINA에서 사용자를 위한 대체 애플리케이션 데스크톱을 만들기 위해 호출됩니다.
PWLX_DIALOG_BOX 대화 상자 템플릿에서 모달 대화 상자를 만들기 위해 GINA에서 호출됩니다.
PWLX_DIALOG_BOX_INDIRECT 메모리의 대화 상자 템플릿에서 모달 대화 상자를 만들기 위해 GINA에서 호출합니다.
PWLX_DIALOG_BOX_INDIRECT_PARAM

GINA에서 호출하여 대화 상자 컨트롤을 초기화한 다음 메모리의 대화 상자 템플릿에서 모달 대화 상자를 만듭니다.
PWLX_DIALOG_BOX_PARAM 대화 상자 컨트롤을 초기화한 다음 대화 상자 템플릿 리소스에서 모달 대화 상자를 만들기 위해 GINA에서 호출합니다.
PWLX_DISCONNECT 터미널 서비스를 사용하는 경우 대체 GINA DLL에 의해 호출됩니다. GINA는 터미널 서비스 네트워크 세션에서 연결을 끊기 위해 이 함수를 호출합니다.
PWLX_GET_OPTION 옵션의 현재 값을 검색하기 위해 GINA에서 호출됩니다.
PWLX_GET_SOURCE_DESKTOP Winlogon이 Winlogon 데스크톱으로 전환되기 전에 현재 데스크톱의 이름과 핸들을 확인하기 위해 GINA에서 호출합니다.
PWLX_MESSAGE_BOX 메시지 상자를 만들고 표시하고 작동하기 위해 GINA에서 호출합니다.
PWLX_QUERY_CLIENT_CREDENTIALS 터미널 서비스를 사용하는 경우 대체 GINA DLL에 의해 호출됩니다. GINA는 이 함수를 호출하여 인터넷 커넥터 라이선스를 사용하지 않는 원격 터미널 서비스 클라이언트의 자격 증명을 검색합니다.
PWLX_QUERY_CONSOLESWITCH_CREDENTIALS 임시 세션의 Winlogon에서 대상 세션의 Winlogon으로 전송된 자격 증명을 읽기 위해 GINA에서 호출합니다.
PWLX_QUERY_IC_CREDENTIALS 터미널 서비스를 사용하는 경우 대체 GINA DLL에 의해 호출됩니다. GINA는 이 함수를 호출하여 터미널 서버가 인터넷 커넥터 라이선스를 사용하고 있는지 확인하고 자격 증명 정보를 검색합니다.
PWLX_QUERY_TERMINAL_SERVICES_DATA 사용자가 로그온한 후 터미널 서비스 사용자 구성 정보를 검색하기 위해 GINA에서 호출됩니다.
PWLX_QUERY_TS_LOGON_CREDENTIALS 터미널 서비스를 사용하는 경우 자격 증명 정보를 검색하기 위해 대체 GINA DLL에 의해 호출됩니

다. 그런 다음 GINA DLL은 이 정보를 사용하여 로그온 상자를 자동으로 채우고 사용자를 로그인하려고 시도할 수 있습니다.
PWLX_SAS_NOTIFY GINA가 Winlogon에 SAS(보안 주의 순서) 이벤트를 알리기 위해 호출합니다.
PWLX_SET_CONTEXT_POINTER GINA에서 호출하여 Winlogon에서 전달한 컨텍스트 포인터를 GINA 함수에 대한 모든 이후 호출에 대한 첫 번째 매개 변수로 지정합니다.
PWLX_SET_OPTION 옵션 값을 설정하기 위해 GINA에서 호출됩니다.
PWLX_SET_RETURN_DESKTOP GINA에서 호출하여 현재 SAS(보안 주의 순서) 이벤트 처리 함수가 완료될 때 Winlogon이 전환할 대체 애플리케이션 데스크톱을 지정합니다.
PWLX_SET_TIMEOUT 대화 상자와 연결된 제한 시간을 변경하기 위해 GINA에서 호출합니다. 기본 제한 시간은 2분입니다.
PWLX_SWITCH_DESKTOP_TO_USER 애플리케이션 데스크톱으로 전환하기 위해 GINA에서 호출됩니다.
PWLX_SWITCH_DESKTOP_TO_WINLOGON GINA DLL을 Winlogon 데스크톱으로 전환할 수 있습니다.
PWLX_USE_CTRL_ALT_DEL GINA에서 호출하여 Winlogon에 표준 CTRL+ALT+DEL 키 조합을 SAS(보안 주의 시퀀스)로 사용하도록 지시합니다.
PWLX_WIN31_MIGRATE 터미널 서비스를 사용하는 경우 대체 GINA DLL에서 호출됩니다. GINA는 이 함수를 호출하여 터미널 서비스 클라이언트의 설정을 완료합니다.

구조체

WLX_CLIENT_CREDENTIALS_INFO_V1_0 WlxQueryClientCredentials 또는 WlxQueryInetConnectorCredentials 호출에서 반환되는 클라이언트 자격 증명을 포함합니다.
WLX_CLIENT_CREDENTIALS_INFO_V2_0 WlxQueryTsLogonCredentials 호출에서 반환된 클라이언트 자격 증명을 포함합니다.
WLX_CONSOLESWITCH_CREDENTIALS_INFO_V1_0 WlxGetConsoleSwitchCredentials 호출에서 반환된 클라이언트 자격 증명을 포함합니다.
WLX_DESKTOP GINA DLL과 Winlogon 간에 데스크톱 정보를 전달하는 데 사용됩니다.
WLX_DISPATCH_VERSION_1_0 WlxInitialize 호출에서 GINA DLL에 전달된 Winlogon 버전 1.0 함수 디스패치 테이블의 형식을 정의합니다.
WLX_DISPATCH_VERSION_1_1 WlxInitialize 호출에서 GINA DLL에 전달된 Winlogon 버전 1.1 함수 디스패치의 형식을 정의합니다.
WLX_DISPATCH_VERSION_1_2 WlxInitialize 호출에서 GINA DLL에 전달된 Winlogon 버전 1.2 함수 디스패치 테이블의 형식을 정의합니다.
WLX_DISPATCH_VERSION_1_3 WlxInitialize 호출에서 GINA DLL에 전달된 Winlogon 버전 1.3 함수 디스패치 테이블의 형식을 정의합니다.
WLX_DISPATCH_VERSION_1_4 WlxInitialize 호출에서 GINA DLL에 전달된 Winlogon 버전 1.4 함수 디스패치 테이블의 형식을 정의합니다.
WLX_MPR_NOTIFY_INFO 네트워크 공급자에게 식별 및 인증 정보를 제공합니다.
WLX_NOTIFICATION_INFO 이 구조체는 Winlogon 이벤트에 대한 정보를 저장합니다.

WLX_PROFILE_V1_0 초기 환경을 설정하는 데 사용되는 정보를 포함합니다.
WLX_PROFILE_V2_0 WLX_PROFILE_V1_0 제공한 정보 외에도 프로필 정보를 포함합니다.
WLX_TERMINAL_SERVICES_DATA 터미널 서비스 사용자 구성 정보를 GINA에 제공하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wlanapi.h 헤더

아티클2024. 03. 08.

이 헤더는 네이티브 와이파이에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 네이티브 와이파이

wlanapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

테이블 확장

WFDCancelOpenSession 애플리케이션이 완료되지 않은 보류 중인 WFDStartOpenSession 함수를 취소하려고 했음을 나타냅니다.
WFDCloseHandle Wi-Fi Direct 서비스에 대한 핸들을 닫습니다.
WFDCloseSession 이전에 WFDStartOpenSession 함수를 성공적으로 호출한 후 세션을 닫습니다.
WFDOpenHandle Wi-Fi Direct 서비스에 대한 핸들을 열고 사용할 Wi-FI Direct API 버전을 협상합니다.
WFDOpenLegacySession Wi-Fi Direct 레거시 디바이스에 대한 저장된 프로필을 검색하고 적용합니다.
WFDStartOpenSession 이전에 Windows 페어링 환경을 통해 페어링된 특정 Wi-Fi Direct 디바이스에 대한 주문형 연결을 시작합니다.
WFDUpdateDeviceVisibility 지정된 Wi-Fi 직접 디바이스 노드에 대한 Wi-Fi 직접 디바이스 주소에 대한 디바이스 표시 유형을 업데이트.

WlanAllocateMemory 메모리를 할당합니다.
WlanCloseHandle 서버에 대한 연결을 닫습니다.
WlanConnect 특정 네트워크에 연결하려고 시도합니다.
WlanDeleteProfile 로컬 컴퓨터에서 무선 인터페이스에 대한 무선 프로필을 삭제합니다.
WlanDeviceServiceCommand OEM 또는 IHV 구성 요소가 특정 무선 LAN 인터페이스에서 디바이스 서비스와 통신할 수 있도록 허용합니다.
WlanDisconnect 현재 네트워크에서 인터페이스 연결을 끊습니다.
WlanEnumInterfaces 로컬 컴퓨터에서 현재 사용하도록 설정된 모든 무선 LAN 인터페이스를 열거합니다.
WlanExtractPsdIEDataList 비콘에 포함된 원시 IE 데이터에서 PSD(근접 서비스 검색) IE(정보 요소) 데이터 목록을 추출합니다.
WlanFreeMemory 메모리를 해제합니다.
WlanGetAvailableNetworkList 무선 LAN 인터페이스에서 사용 가능한 네트워크 목록을 검색합니다.
WlanGetFilterList 그룹 정책 또는 사용자 권한 목록을 검색합니다.
WlanGetInterfaceCapability 인터페이스의 기능을 검색합니다.

WlanGetNetworkBssList 지정된 무선 LAN 인터페이스에서 무선 네트워크 또는 네트워크의 기본 BSS(서비스 집합) 항목 목록을 검색합니다.
WlanGetProfile 지정된 무선 프로필에 대한 모든 정보를 검색합니다.
WlanGetProfileCustomUserData 무선 프로필과 연결된 사용자 지정 사용자 데이터를 가져옵니다.
WlanGetProfileList 프로필 목록을 검색합니다.
WlanGetSecuritySettings 구성 가능한 개체와 연결된 보안 설정을 가져옵니다.
WlanGetSupportedDeviceServices 지정된 무선 LAN 인터페이스에서 지원되는 디바이스 서비스 목록을 검색합니다.
WlanHostedNetworkForceStart 요청을 애플리케이션의 호출 핸들과 연결하지 않고 무선 호스트 네트워크를 wlan_hosted_network_active 상태로 전환합니다.
WlanHostedNetworkForceStop 요청을 애플리케이션의 호출 핸들과 연결하지 않고 무선 호스트 네트워크를 wlan_hosted_network_idle 전환합니다.
WlanHostedNetworkInitSettings 이러한 설정이 아직 구성되지 않은 경우 무선 호스트 네트워크에서 네트워크 연결 설정(예: SSID 및 최대 피어 수)을 저장하도록 구성하고 유지합니다.
WlanHostedNetworkQueryProperty 무선 호스트 네트워크의 현재 정적 속성을 쿼리합니다.
WlanHostedNetworkQuerySecondaryKey 무선 호스트 네트워크에서 사용하도록 구성된 보조 보안 키를 쿼리합니다.

WlanHostedNetworkQueryStatus 무선 호스트 네트워크의 현재 상태 쿼리합니다.
WlanHostedNetworkRefreshSecuritySettings 무선 호스트 네트워크 보안 설정의 구성 가능 및 자동 생성 부분을 새로 고칩니다.
WlanHostedNetworkSetProperty 무선 호스트 네트워크의 정적 속성을 설정합니다.
WlanHostedNetworkSetSecondaryKey 무선 호스트 네트워크에서 사용할 보조 보안 키를 구성합니다.
WlanHostedNetworkStartUsing 무선 호스트 네트워크를 시작합니다.
WlanHostedNetworkStopUsing 무선 호스트 네트워크를 중지합니다.
WlanIhvControl WLAN 드라이버 또는 서비스의 IHV(독립 하드웨어 공급업체) 제어를 위한 메커니즘을 제공합니다.
WlanOpenHandle 서버에 대한 연결을 엽니다.
WlanQueryAutoConfigParameter 자동 구성 서비스의 매개 변수에 대한 쿼리입니다.
WlanQueryInterface WlanQueryInterface 함수는 지정된 인터페이스의 다양한 매개 변수를 쿼리합니다.
WlanReasonCodeToString 지정된 이유 코드를 설명하는 문자열을 검색합니다.
WlanRegisterDeviceServiceNotification 관리자 권한 또는 User-Mode 드라이버 프레임워크(UMDF) 드라이버가 있는 사용자 모드 클라이언트가 관심 있는 디바이스 서비스에 해당하는 원치 않는 알림을 등록할 수 있습니다.

WlanRegisterNotification 모든 무선 인터페이스에서 알림을 등록 및 등록 취소하는 데 사용됩니다.
WlanRegisterVirtualStationNotification 가상 스테이션에서 알림을 등록 및 등록 취소하는 데 사용됩니다.
WlanRenameProfile 지정된 프로필의 이름을 바꿉니다.
WlanSaveTemporaryProfile 임시 프로필을 프로필 저장소에 저장합니다.
WlanScan 표시된 인터페이스에서 사용 가능한 네트워크에 대한 검사를 요청합니다.
WlanSetAutoConfigParameter 자동 구성 서비스에 대한 매개 변수를 설정합니다.
WlanSetFilterList 허용/거부 목록을 설정합니다.
WlanSetInterface 사용자 구성 가능한 매개 변수를 설정합니다.
WlanSetProfile 특정 프로필의 콘텐츠를 설정합니다.
WlanSetProfileCustomUserData 프로필과 연결된 사용자 지정 사용자 데이터를 설정합니다.
WlanSetProfileEapUserData 원시 EAP 데이터에 지정된 대로 EAP(확장 가능한 인증 프로토콜) 사용자 자격 증명을 설정합니다.
WlanSetProfileEapXmlUserData XML 문자열에 지정된 대로 EAP(확장 가능 인증 프로토콜) 사용자 자격 증명을 설정합니다.
WlanSetProfileList

프로필의 기본 설정 순서를 설정합니다.
WlanSetProfilePosition 기본 설정 목록에서 지정된 단일 프로필의 위치를 설정합니다.
WlanSetPsdIEDataList PSD(근접 서비스 검색) IE(정보 요소) 데이터 목록을 설정합니다.
WlanSetSecuritySettings 구성 가능한 개체에 대한 보안 설정을 지정합니다.
WlanUIEditProfile 무선 프로필 UI(사용자 인터페이스)를 표시합니다.

콜백 함수

테이블 확장

WFD_OPEN_SESSION_COMPLETE_CALLBACK WFDStartOpenSession 작업이 완료된 경우 WFDStartOpenSession 함수에서 호출되는 콜백 함수를 정의합니다.
WLAN_NOTIFICATION_CALLBACK 알림 콜백 함수의 유형을 정의합니다.

구조체

테이블 확장

DOT11_NETWORK 사용 가능한 무선 네트워크에 대한 정보를 포함합니다. (DOT11_NETWORK)
DOT11_NETWORK_LIST

802.11 무선 네트워크 목록을 포함합니다.
WLAN_ASSOCIATION_ATTRIBUTES 연결에 대한 연결 특성을 포함합니다.
WLAN_AUTH_CIPHER_PAIR_LIST 인증 및 암호 알고리즘 쌍 목록을 포함합니다.
WLAN_AVAILABLE_NETWORK 사용 가능한 무선 네트워크에 대한 정보를 포함합니다. (WLAN_AVAILABLE_NETWORK)
WLAN_AVAILABLE_NETWORK_LIST 사용 가능한 네트워크에 대한 정보 배열을 포함합니다.
WLAN_BSS_ENTRY BSS(기본 서비스 집합)에 대한 정보를 포함합니다.
WLAN_BSS_LIST BSS(기본 서비스 집합) 항목 목록을 포함합니다.
WLAN_CONNECTION_ATTRIBUTES 무선 연결의 특성을 정의합니다.
WLAN_CONNECTION_NOTIFICATION_DATA 연결 관련 알림에 대한 정보를 포함합니다.
WLAN_CONNECTION_PARAMETERS WlanConnect 함수를 사용할 때 사용되는 매개 변수를 지정합니다.
WLAN_CONNECTION_QOS_INFO TBD
WLAN_COUNTRY_OR_REGION_STRING_LIST 지원되는 국가 또는 지역 문자열 목록을 포함합니다.
WLAN_DEVICE_SERVICE_GUID_LIST 디바이스 서비스 GUID의 배열을 포함합니다.

WLAN_DEVICE_SERVICE_NOTIFICATION_DATA 디바이스 서비스 알림을 나타내는 구조체입니다.
WLAN_HOSTED_NETWORK_CONNECTION_SETTINGS 무선 호스트 네트워크의 연결 설정에 대한 정보를 포함합니다.
WLAN_HOSTED_NETWORK_DATA_PEER_STATE_CHANGE 무선 호스티드 네트워크의 데이터 피어에 대한 네트워크 상태 변경에 대한 정보를 포함합니다.
WLAN_HOSTED_NETWORK_PEER_STATE 무선 호스트 네트워크의 피어에 대한 피어 상태에 대한 정보를 포함합니다.
WLAN_HOSTED_NETWORK_RADIO_STATE 무선 호스트 네트워크의 라디오 상태에 대한 정보를 포함합니다.
WLAN_HOSTED_NETWORK_SECURITY_SETTINGS 무선 호스트 네트워크의 보안 설정에 대한 정보를 포함합니다.
WLAN_HOSTED_NETWORK_STATE_CHANGE 무선 호스트 네트워크의 네트워크 상태 변경에 대한 정보를 포함합니다.
WLAN_HOSTED_NETWORK_STATUS 무선 호스트 네트워크의 상태 대한 정보를 포함합니다.
WLAN_INTERFACE_CAPABILITY 인터페이스의 기능에 대한 정보를 포함합니다.
WLAN_INTERFACE_INFO 무선 LAN 인터페이스에 대한 정보를 포함합니다.
WLAN_INTERFACE_INFO_LIST NIC 인터페이스 정보의 배열입니다.
WLAN_MAC_FRAME_STATISTICS 전송 및 수신된 MAC 프레임에 대한 정보를 포함합니다.
WLAN_MSM_NOTIFICATION_DATA

MSM(미디어 관련 모듈) 연결 관련 알림에 대한 정보를 포함합니다.
WLAN_PHY_FRAME_STATISTICS 전송 및 수신된 PHY 프레임에 대한 정보를 포함합니다.
WLAN_PHY_RADIO_STATE 라디오 상태를 지정합니다.
WLAN_PROFILE_INFO 프로필에 대한 기본 정보입니다.
WLAN_PROFILE_INFO_LIST 무선 프로필 정보 목록을 포함합니다.
WLAN_QOS_CAPABILITIES TBD
WLAN_QOS_INFO WFA의 Wi-Fi QoS 관리 사양에 설명되고 802.11 사양에 정의된 네 가지 기능에 대한 정보를 나타냅니다.
WLAN_RADIO_STATE PHY(물리적 계층) 형식 목록에서 라디오 상태를 지정합니다.
WLAN_RATE_SET 지원되는 데이터 속도 집합입니다.
WLAN_RAW_DATA 일부 네이티브 Wifi 함수에서 사용되는 Blob 형식의 원시 데이터를 포함합니다.
WLAN_RAW_DATA_LIST 일부 네이티브 Wifi 함수에서 사용되는 데이터 Blob 배열 형식의 원시 데이터를 포함합니다.
WLAN_REALTIME_CONNECTION_QUALITY 지정된 인터페이스에서 연결의 품질을 설명하는 특성을 나타냅니다.
WLAN_REALTIME_CONNECTION_QUALITY_LINK_INFO TBD

WLAN_SECURITY_ATTRIBUTES 무선 연결에 대한 보안 특성을 정의합니다.
WLAN_STATISTICS 인터페이스에 대한 다양한 통계입니다.

열거형

테이블 확장

DOT11_RADIO_STATE DOT11_RADIO_STATE 열거형은 802.11 라디오 상태를 지정합니다.
WL_DISPLAY_PAGES 무선 프로필 사용자 인터페이스 대화 상자가 나타날 때 활성 탭을 지정합니다.
WLAN_ADHOC_NETWORK_STATE WLAN_ADHOC_NETWORK_STATE 열거형은 임시 네트워크의 연결 상태를 지정합니다.
WLAN_AUTOCONF_OPCODE WLAN_AUTOCONF_OPCODE 열거형은 자동 구성 매개 변수를 지정합니다.
WLAN_CONNECTION_MODE 연결 모드를 정의합니다.
WLAN_FILTER_LIST_TYPE 필터 목록의 유형을 나타냅니다.
WLAN_HOSTED_NETWORK_NOTIFICATION_CODE 무선 Hosted Network에서 수신된 알림에 대한 NotificationCode 매개 변수의 가능한 값을 지정합니다.
WLAN_HOSTED_NETWORK_OPCODE 무선 Hosted Network에서 쿼리하거나 설정할 속성에 대한 작업 코드의 가능한 값을 지정합니다.

WLAN_HOSTED_NETWORK_PEER_AUTH_STATE 무선 Hosted Network에서 피어의 인증 상태에 사용할 수 있는 값을 지정합니다.
WLAN_HOSTED_NETWORK_REASON 무선 Hosted Network 함수 호출 결과에 사용할 수 있는 값을 지정합니다.
WLAN_HOSTED_NETWORK_STATE 무선 Hosted Network의 네트워크 상태에 사용할 수 있는 값을 지정합니다.
WLAN_IHV_CONTROL_TYPE WLAN_IHV_CONTROL_TYPE 열거형은 공급업체별 메서드에서 우회하는 소프트웨어 유형을 지정합니다.
WLAN_INTERFACE_STATE WLAN_INTERFACE_STATE 열거형은 인터페이스의 상태를 나타냅니다.
WLAN_INTERFACE_TYPE 무선 인터페이스 유형을 지정합니다.
WLAN_INTF_OPCODE 무선 인터페이스에서 매개 변수를 설정하고 쿼리하는 데 사용되는 다양한 opcode를 지정하는 상수를 정의합니다.
WLAN_NOTIFICATION_ACM WLAN_NOTIFICATION_ACM 열거형은 WLAN_NOTIFICATION_DATA 구조체의 NotificationCode 멤버의 가능한 값을 지정합니다.
WLAN_NOTIFICATION_MSM WLAN_NOTIFICATION_MSM 열거형은 WLAN_NOTIFICATION_DATA 구조체의 NotificationCode 멤버의 가능한 값을 지정합니다.
WLAN_OPCODE_VALUE_TYPE WLAN_OPCODE_VALUE_TYPE 열거형은 자동 구성(자동 구성) 설정의 원본을 지정합니다.
WLAN_POWER_SETTING WLAN_POWER_SETTING 열거형은 인터페이스의 전원 설정을 지정합니다.

WLAN_SECURABLE_OBJECT 네이티브 Wifi Functions에서 사용하는 보안 개체를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

wldp.h 헤더

아티클2024. 11. 20.

이 헤더는 보안 기능 영역의 일부입니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

wldp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

테이블 확장

WldpCanExecuteBuffer 실행 정책에서 제공된 버퍼에서 코드를 실행할 수 있는지 여부를 쿼리합니다.
WldpCanExecuteFile 실행 정책에서 제공된 파일에서 코드를 실행할 수 있는지 여부를 쿼리합니다.
WldpCanExecuteStream 실행 정책에서 제공된 스트림에서 코드를 실행할 수 있는지 여부를 쿼리합니다.
WldpGetApplicationSettingBoolean 제공된 파일의 모든 비즈니스용 애플리케이션 제어 정책에서 `boolean` 형식으로 정의된 애플리케이션 설정의 값을 쿼리합니다.
WldpGetApplicationSettingStringList 제공된 파일의 모든 비즈니스용 애플리케이션 제어 정책에서 `stringlist` 형식으로 정의된 애플리케이션 설정의 값을 쿼리합니다.
WldpGetApplicationSettingStringSet 제공된 파일의 모든 비즈니스용 애플리케이션 제어 정책에서 `stringset` 형식으로 정의된 애플리케이션 설정의 값을 쿼리합니다.
WldpGetLockdownPolicy 라이브러리를 호출하여 호스트에 상대적인 보안 상태를 가져와서 사용할 스크립트 또는 msi를 가져옵니다.

WldpIsClassInApprovedList 라이브러리를 호출하여 특정 CLSID를 호출해도 안전한지 확인합니다.
WldpIsDynamicCodePolicyEnabled .NET 동적 코드에 대한 Device Guard 정책 적용 상태를 설명하는 값을 검색합니다.
WldpQueryDynamicCodeTrust 지정된 메모리 내 또는 디스크 내 .NET CRL 동적 코드가 Device Guard 정책에서 신뢰할 수 있는지 여부를 결정하는 값을 검색합니다.
WldpQueryWindowsLockdownMode 현재 Windows 보안 모드를 검색합니다. Windows는 잠긴 모드, 잠금 해제된 표준 모드 또는 평가판 모드일 수 있습니다.
WldpSetDynamicCodeTrust .NET에 대해 신뢰할 수 있는 디스크 내 .NET CRL 동적 코드를 설정합니다.

구조

테이블 확장

WLDP_HOST_INFORMATION 평가할 호스트 및 원본 파일을 식별하는 구조입니다.

열거형

테이블 확장

WLDP_EXECUTION_EVALUATION_OPTIONS WldpCanExecuteBuffer, WldpCanExecuteFile 및 WldpCanExecuteStream을 사용하여 실행 권한 부여를 쿼리할 때 실행 옵션을 지정합니다.

WLDP_EXECUTION_POLICY
WLDP_HOST_ID WLDP 호출자의 호스트 유형을 식별합니다.
WLDP_WINDOWS_LOCKDOWN_MODE Windows 디바이스의 보안 모드(S 모드)에 대해 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

제품 사용자 의견 제공 | Microsoft Q&A에서 도움말 보기

wmcodecdsp.h 헤더

아티클2023. 08. 28.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- CIMWin32 WMI 공급자
- Directshow
- Microsoft Media Foundation
- Windows Media Format 11 SDK

wmcodecdsp.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IToc IToc 인터페이스는 개별 목차를 나타냅니다. 목차에 항목을 추가하고 항목을 제거하는 메서드를 제공합니다.
ITocCollection ITocCollection은 목차의 컬렉션을 나타냅니다. 컬렉션에서 목차를 추가, 검색 및 제거하는 메서드를 제공합니다.
ITocEntry ITocEntry 인터페이스는 목차의 개별 항목을 나타냅니다. 항목에 대한 설명 정보를 설정하고 검색하는 메서드를 제공합니다.
ITocEntryList ITocEntryList 인터페이스는 목차의 항목 목록을 나타냅니다. 목록에서 항목을 추가하고 항목을 제거하는 메서드를 제공합니다.
ITocParser ITocParser 인터페이스는 TOC 파서 개체를 나타냅니다. 비디오 파일에 목차를 저장하고 비디오 파일에서 목차를 검색하는 메서드를 제공합니다.
IWMCodecLeakyBucket 비디오 인코더에서 "새는 버킷" 매개 변수를 구성합니다.
IWMCodecOutputTimestamp

디코딩할 다음 비디오 프레임의 타임스탬프를 가져옵니다.
IWMCodecPrivateData 출력 미디어 형식에 추가해야 하는 프라이빗 코덱 데이터를 가져옵니다. 이 코덱 데이터는 Windows Media 비디오 콘텐츠를 제대로 디코딩하는 데 필요합니다.
IWMCodecProps 형식별 코덱 속성을 검색하는 메서드를 제공합니다.
IWMCodecStrings 코덱 및 형식에 대한 이름 및 설명 문자열을 검색합니다.
IWMColorConvProps 색 변환기 DSP의 속성을 설정합니다.
IWMResamplerProps 오디오 리샘플러 DSP의 속성을 설정합니다.
IWMResizerProps 비디오 리사이저 DSP의 속성을 설정합니다.
IWMSampleExtensionSupport 샘플 확장에 대한 코덱 지원을 구성합니다.
IWMVideoDecoderHurryup 비디오 디코더의 속도를 제어합니다.
IWMVideoDecoderReconBuffer 참고 이 인터페이스는 사용되지 않으므로 사용하면 안 됩니다. 재구성된 비디오 프레임을 관리합니다.
IWMVideoForceKeyFrame 인코더가 현재 프레임을 키 프레임으로 인코딩하도록 합니다.

구조체

AecQualityMetrics_Struct AEC(음향 에코 취소)에 대한 품질 메트릭을 포함합니다. 이 구조체는 MFPKEY_WMAAECMA_RETRIEVE_TS_STATS 속성과 함께 사용됩니다.
TOC_DESCRIPTOR TOC_DESCRIPTOR 구조체는 목차에 대한 설명 정보를 보유합니다.
TOC_ENTRY_DESCRIPTOR TOC_ENTRY_DESCRIPTOR 구조체는 목차의 항목에 대한 설명 정보를 보유합니다.

열거형

AEC_SYSTEM_MODE 음성 캡처 DSP의 처리 모드를 지정합니다. 이 열거형은 MFPKEY_WMAAECMA_SYSTEM_MODE 속성과 함께 사용됩니다.
AEC_VAD_MODE 음성 캡처 DSP에 대한 VAD(음성 활동 감지) 유형을 지정합니다. 이 열거형은 MFPKEY_WMAAECMA_FEATR_VAD 속성과 함께 사용됩니다.
MFVideoDSPMode 비디오 안정화 MFT의 처리 모드를 지정합니다.
MIC_ARRAY_MODE 음성 캡처 DSP가 마이크 배열 처리를 수행하는 방법을 지정합니다. 이 열거형은 MFPKEY_WMAAECMA_FEATR_MICARR_MODE 속성과 함께 사용됩니다.
TOC_POS_TYPE TOC_POS_TYPE 열거형에는 목차의 위치 유형을 지정하는 멤버가 포함됩니다.
WMT_PROP_DATATYPE IWMCodecProps 인터페이스의 메서드를 사용하여 액세스하는 코덱 및 DSP 속성에 사용되는 데이터 형식을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wmcontainer.h 헤더

아티클2023. 08. 28.

이 헤더는 Microsoft Media Foundation에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Microsoft Media Foundation

wmcontainer.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IMFASFContentInfo ASF(고급 시스템 형식) 사양을 준수하는 파일의 헤더 섹션을 작업하는 메서드를 제공합니다.
IMFASFIndexer ASF(시스템 형식) 파일에서 인덱스를 사용할 수 있는 메서드를 제공합니다.
IMFASFMultiplexer ASF(Advanced Systems Format) 데이터 패킷을 만드는 메서드를 제공합니다.
IMFASFMutualExclusion ASF 프로필에서 상호 배타적인 스트림 그룹에 대한 정보를 관리하는 ASF(Advanced Systems Format) 상호 제외 개체를 구성합니다.
IMFASFProfile ASF(고급 시스템 형식) 프로필을 관리합니다.
IMFASFSplitter ASF(Advanced Systems Format) 파일에서 데이터를 읽는 메서드를 제공합니다.
IMFASFStreamConfig ASF 파일에서 스트림의 설정을 구성합니다.
IMFASFStreamPrioritization 구현되지 않았습니다. (IMFASFStreamPrioritization)

IMFASFStreamSelector ASF 헤더의 상호 제외 정보에 따라 ASF(고급 시스템 형식) 파일에서 스트림을 선택합니다.
IMFDRMNetHelper 네트워크 싱크의 네트워크 디바이스에 대한 Windows Media DRM(디지털 권한 관리)을 구성합니다.

Functions

MFCreateASFContentInfo ASF Header 개체 개체를 만듭니다.
MFCreateASFIndexer ASF 인덱서 개체를 만듭니다.
MFCreateASFIndexerByteStream ASF 스트림의 인덱스 액세스에 대한 바이트 스트림을 만듭니다.
MFCreateASFMediaSink ASF 미디어 싱크를 만듭니다.
MFCreateASFMediaSinkActivate ASF 미디어 싱크를 만드는 데 사용할 수 있는 활성화 개체를 만듭니다.
MFCreateASFMultiplexer ASF 멀티플렉서를 만듭니다.
MFCreateASFProfile ASF 프로필 개체를 만듭니다.
MFCreateASFProfileFromPresentationDescriptor 프레젠테이션 설명자에서 ASF 프로필 개체를 만듭니다.
MFCreateASFSplitter

ASF 분할자를 만듭니다.
MFCreateASFStreamingMediaSink ASF 스트리밍 싱크에 대한 활성화 개체를 만듭니다. (MFCreateASFStreamingMediaSink)
MFCreateASFStreamingMediaSinkActivate ASF 스트리밍 싱크에 대한 활성화 개체를 만듭니다. (MFCreateASFStreamingMediaSinkActivate)
MFCreateASFStreamSelector ASF 스트림 선택기를 만듭니다.
MFCreatePresentationDescriptorFromASFProfile ASF 프로필 개체에서 프레젠테이션 설명자를 만듭니다.
MFCreateWMAEncoderActivate WMA(Windows Media Audio) 인코더를 만드는 데 사용할 수 있는 활성화 개체를 만듭니다.
MFCreateWMVEncoderActivate WMV(Windows Media Video) 인코더를 만드는 데 사용할 수 있는 활성화 개체를 만듭니다.

구조체

ASF_INDEX_DESCRIPTOR 스트림 및 인덱스 유형에 대한 인덱싱 구성을 설명합니다.
ASF_INDEX_IDENTIFIER ASF 인덱서 개체의 인덱스를 지정합니다.
ASF_MUX_STATISTICS ASF 멀티플렉서의 진행률에 대한 통계를 포함합니다.

열거형

ASF_SELECTION_STATUS ASF 스트림에 대한 선택 옵션을 정의합니다.
ASF_STATUSFLAGS IMFASFSplitter::GetNextSample 메서드에 대한 상태 조건을 정의합니다.
MFASF_INDEXER_FLAGS ASF 인덱서 옵션을 정의합니다.
MFASF_MULTIPLEXERFLAGS ASF 멀티플렉서 옵션을 정의합니다.
MFASF_SPLITTERFLAGS ASF 분할자 옵션을 정의합니다.
MFASF_STREAMSELECTOR_FLAGS ASF 스트림 선택기 옵션을 정의합니다.
MFSINK_WMDRMACTION ASF 파일 싱크가 Windows Media DRM을 적용하는 방법을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wmdmlog.h 헤더

아티클2023. 08. 28.

이 헤더는 Windows Media 디바이스 관리자 11 SDK에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Media 디바이스 관리자 11 SDK

wmdmlog.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWMDMLogger IWMDMLogger 인터페이스는 Windows Media 디바이스 관리자 애플리케이션 및 서비스 공급자가 공통 로그 파일에 항목을 기록하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wmiutils.h 헤더

아티클2023. 08. 28.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 레거시 Windows 환경 기능
- Windows Management Instrumentation

wmiutils.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWbemPath IWbemPath 인터페이스는 개체 경로 파서의 기본 인터페이스이며 표준 방식으로 프로그램에서 사용할 수 있는 경로를 구문 분석합니다. 이 인터페이스는 경로 정보를 설정하고 검색하기 위한 기본 인터페이스입니다.
IWbemPathKeyList 경로 키의 세부 정보에 액세스하는 데 사용됩니다.
IWbemQuery WQL(WMI Query Language) 쿼리를 구문 분석할 수 있는 진입점을 제공합니다.

구조체

SWbemAssocQueryInf WMIQ_ANALYSIS_ASSOC_QUERY 분석 유형을 사용할 때 IWbemQuery::GetAnalysis 메서드의 정보를 포함합니다.
SWbemQueryQualifiedName SWbemQueryQualifiedName 구조체는 IWbemQuery::GetAnalysis 메서드의 속성 이름을 저장합니다.
SWbemRpnEncodedQuery SWbemRpnEncodedQuery 구조체는 WMIQ_ANALYSIS_RPN_SEQUENCE 분석 유형을 사용할 때

IWbemQuery::GetAnalysis 메서드의 정보를 포함합니다. 구조체의 모든 필드가 적극적으로 사용되는 것은 아닙니다. 일부는 나중에 사용하도록 예약되어 있기 때문입니다.
SWbemRpnQueryToken SWbemRpnQueryToken 구조는 WMIQ_ANALYSIS_RPN_SEQUENCE 형식 쿼리의 쿼리 토큰을 나타냅니다. 쿼리 토큰의 예는 다음과 같습니다._j > 4.

열거형

tag_WBEM_GET_KEY_FLAGS 텍스트의 형식을 제어하는 플래그를 포함합니다.
tag_WBEM_GET_TEXT_FLAGS 텍스트가 반환되는 방법을 제어하는 플래그를 포함합니다.
tag_WBEM_PATH_CREATE_FLAG 허용되는 경로의 유형을 지정하는 플래그를 포함합니다.
tag_WBEM_PATH_STATUS_FLAG 경로를 설명하는 데 사용되는 플래그를 포함합니다.
WMIQ_ANALYSIS_TYPE GetAnalysis 메서드를 사용하여 수행할 분석 유형을 지정하는 데 사용되는 상수를 포함합니다.
WMIQ_ASSOCQ_FLAGS 쿼리의 기능을 나타내는 플래그를 포함합니다.
WMIQ_RPN_TOKEN_FLAGS GetAnalysis 메서드에 사용되는 쿼리 토큰을 설명하는 플래그를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wnvapi.h 헤더

아티클2023. 03. 08.

이 헤더는 Windows 네트워크 가상화에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 네트워크 가상화

wnvapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

함수

WnvOpen WNV 알림을 요청하고 수신하는 데 사용할 Windows WNV(네트워크 가상화) 드라이버 개체에 대한 핸들을 제공합니다.
WnvRequestNotification 특정 유형의 이벤트가 발생할 때마다 Windows WNV(네트워크 가상화) 드라이버에서 알림을 요청합니다.

구조체

WNV_IP_ADDRESS IP 주소 개체를 정의합니다.
WNV_NOTIFICATION_PARAM WnvRequestNotification 함수 호출의 버전, 알림 유형 및 버퍼 위치를 지정합니다.
WNV_OBJECT_CHANGE_PARAM Windows WNV(네트워크 가상화) 드라이버가 WnvObjectChangeType 유형의 알림을 생성하도록 하는 이벤트의 매개 변수를 지정합니다.
WNV_OBJECT_HEADER WnvRequestNotification 함수에 전달되는 WNV_NOTIFICATION_PARAM 구조체의 주 버전, 부 버전 및 버퍼 크기를 지정합니다.

WNV_POLICY_MISMATCH_PARAM Windows WNV(네트워크 가상화) 드라이버가 WnvPolicyMismatchType 알림을 생성하도록 하는 이벤트(일반적으로 들어오는 패킷)의 매개 변수를 지정합니다.
WNV_PROVIDER_ADDRESS_CHANGE_PARAM 공급자 주소의 DAD(중복 주소 검색) 상태 변경을 지정하여 WNV(Windows 네트워크 가상화) 드라이버가 이 구조를 포함하는 WnvProviderAddressType 개체 형식을 지정하는 WnvObjectChangeType 알림을 생성합니다.
WNV_REDIRECT_PARAM WNV(Windows 네트워크 가상화) 드라이버가 WnvRedirectType 알림을 생성하도록 하는 이벤트의 매개 변수(들어오는 인터넷 제어 메시지 프로토콜 리디렉션 패킷 수신)를 지정합니다.

열거형

WNV_NOTIFICATION_TYPE 지정된 WNV(Windows 네트워크 가상화) 알림의 유형을 지정합니다.
WNV_OBJECT_TYPE WNV 알림 유형이 WnvObjectChangeType인 경우 지정된 WNV(Windows 네트워크 가상화) 알림의 개체 유형을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wofapi.h 헤더

아티클2023. 08. 28.

이 헤더는 데이터 액세스 및 스토리지에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지

wofapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WofEnumEntries 지정된 볼륨에 대해 지정된 공급자의 모든 데이터 원본을 열거합니다.
WofFileEnumFiles 지정된 볼륨에서 지정된 압축 알고리즘을 사용하여 압축되는 모든 파일을 열거합니다.
WofGetDriverVersion 특정 공급자를 지원하는 데 사용되는 드라이버 버전을 쿼리하는 데 사용됩니다.
WofIsExternalFile 실제 파일 또는 시스템 데이터 공급자가 파일을 백업하는지 확인하는 데 사용되며, 필요에 따라 파일에 대한 공급자 또는 추가 데이터를 나타냅니다.
WofSetFileDataLocation 파일을 물리적 파일 백업에서 시스템 데이터 공급자 백업으로 변경하는 데 사용됩니다.
WofShouldCompressBinaries 특정 볼륨에서 압축을 사용해야 하는지, 사용해야 한다면 어떤 압축 알고리즘을 사용해야 하는지 나타냅니다.
WofWimAddEntry WIM 내에 저장되는 볼륨에 파일을 만들 수 있도록 단일 WIM 데이터 원본을 볼륨에 추가합니다.
WofWimEnumFiles 지정된 볼륨의 지정된 WIM 데이터 원본이 백업하는 모든 파일을 열거합니다.

WofWimRemoveEntry 볼륨의 백업 파일에서 단일 WIM 데이터 원본을 제거합니다.
WofWimSuspendEntry 볼륨이 다시 탑재되거나 데이터 원본이 WofWimUpdateEntry로 업데이트될 때까지 볼륨의 백업 파일에서 WIM 데이터 원본을 일시적으로 제거합니다.
WofWimUpdateEntry 다른 WIM 파일 위치를 가리키도록 WIM 항목을 업데이트합니다.

콜백 함수

WofEnumEntryProc WofEnumEntries 호출에 대한 응답으로 각 데이터 원본에 대해 호출되는 콜백 함수입니다.
WofEnumFilesProc WIM 파일과 같은 외부 데이터 원본이 지원하는 각 파일에 대해 호출되는 콜백 함수입니다.

구조체

WIM_ENTRY_INFO 볼륨에서 호스트되는 각 WIM 데이터 원본과 관련된 메타데이터를 정의합니다.
WIM_EXTERNAL_FILE_INFO WOF_PROVIDER_WIM에서 제공하는 파일과 관련된 메타데이터를 정의합니다.
WOF_FILE_COMPRESSION_INFO_V1 WOF_PROVIDER_FILE에서 제공하는 파일과 관련된 메타데이터를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

workspaceax.h 헤더

아티클2023. 08. 28.

이 헤더는 원격 데스크톱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 데스크톱 서비스

workspaceax.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWorkspaceResTypeRegistry 플러그 인이 RemoteApp 및 데스크톱 연결 런타임에서 타사 파일 이름 확장명을 관리할 수 있도록 하는 메서드를 노출합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

workspaceruntime.h 헤더

아티클2023. 03. 08.

이 헤더는 원격 데스크톱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 데스크톱 서비스

workspaceruntime.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWorkspace RemoteApp 및 데스크톱 연결의 연결에 대한 정보를 제공하는 메서드를 노출합니다.
IWorkspace2 RemoteApp 및 데스크톱 연결의 연결에 대한 정보를 제공하는 추가 메서드를 노출합니다.
IWorkspace3 RemoteApp 및 데스크톱 연결에서 연결에 대한 정보를 제공하는 메서드를 노출하고 클레임 토큰을 검색하거나 설정하는 기능을 추가합니다.
IWorkspaceRegistration RemoteApp 및 데스크톱 연결에서 사용자 지정 클라이언트에 대한 참조를 추가하고 제거하는 메서드를 노출합니다. (IWorkspaceRegistration)
IWorkspaceRegistration2 RemoteApp 및 데스크톱 연결에서 사용자 지정 클라이언트에 대한 참조를 추가하고 제거하는 메서드를 노출합니다. (IWorkspaceRegistration2)
IWorkspaceReportMessage 원격 작업 영역에 대한 오류 메시지 처리를 지원하는 메서드를 노출합니다.
IWorkspaceScriptable RemoteApp 및 데스크톱 연결 자격 증명 및 연결을 관리하는 메서드를 노출합니다. (IWorkspaceScriptable)

IWorkspaceScriptable2 RemoteApp 및 데스크톱 연결 자격 증명 및 연결을 관리하는 메서드를 노출합니다. (IWorkspaceScriptable2)
IWorkspaceScriptable3 RemoteApp 및 데스크톱 연결 자격 증명 및 연결을 관리하는 메서드를 노출합니다. (IWorkspaceScriptable3)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

workspaceruntimeclientext.h 헤더

아티클2023. 03. 08.

이 헤더는 원격 데스크톱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 데스크톱 서비스

workspaceruntimeclientext.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWorkspaceClientExt 런타임이 RemoteApp 및 데스크톱 연결에서 사용자 지정 클라이언트의 연결을 끊을 수 있도록 하는 메서드를 노출합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wow64apiset.h 헤더

아티클2023. 03. 08.

이 헤더는 데이터 액세스 및 스토리지에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 데이터 액세스 및 스토리지

wow64apiset.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

GetSystemWow64Directory2A 지정된 이미지 파일 컴퓨터 형식을 사용하여 WOW64에서 사용하는 시스템 디렉터리의 경로를 검색합니다. (ANSI)
GetSystemWow64Directory2W 지정된 이미지 파일 컴퓨터 형식을 사용하여 WOW64에서 사용하는 시스템 디렉터리의 경로를 검색합니다. (유니코드)
GetSystemWow64DirectoryA WOW64에서 사용하는 시스템 디렉터리의 경로를 검색합니다. (ANSI)
GetSystemWow64DirectoryW WOW64에서 사용하는 시스템 디렉터리의 경로를 검색합니다. (유니코드)
IsWow64GuestMachineSupported 지정된 컴퓨터 아키텍처에서 지원되는 아키텍처(WOW64 아래)를 결정합니다.
IsWow64Process 지정된 프로세스가 WOW64 또는 x64 프로세서의 Intel64에서 실행되고 있는지 여부를 확인합니다.
IsWow64Process2 지정된 프로세스가 WOW64에서 실행 중인지 여부를 확인합니다. 또한 추가 머신 프로세스 및 아키텍처 정보를 반환합니다.

Wow64DisableWow64FsRedirection 호출 스레드에 대한 파일 시스템 리디렉션을 사용하지 않도록 설정합니다. 파일 시스템 리디렉션은 기본적으로 사용하도록 설정됩니다.
Wow64EnableWow64FsRedirection 호출 스레드에 대한 파일 시스템 리디렉션을 사용하거나 사용하지 않도록 설정합니다.
Wow64GetThreadContext 스레드 컨텍스트를 검색합니다.
Wow64RevertWow64FsRedirection 호출 스레드에 대한 파일 시스템 리디렉션을 복원합니다.
Wow64SetThreadContext 스레드 컨텍스트를 설정합니다.
Wow64SetThreadDefaultGuestMachine 스레드를 지정된 컴퓨터 아키텍처로 설정합니다.
Wow64SuspendThread 지정된 WOW64 스레드를 일시 중단합니다.

wpcapi.h 헤더

아티클2023. 08. 28.

이 헤더는 자녀 보호에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 자녀

wpcapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWindowsParentalControls 최소 준수 API의 모든 설정 인터페이스에 액세스할 수 있습니다.
IWindowsParentalControlsCore . (IWindowsParentalControlsCore)
IWPCGamesSettings 사용자의 게임 설정에 액세스합니다.
IWPCProviderConfig 타사에서 구현하는 구성 메서드를 노출합니다.
IWPCProviderState 타사에서 구현하는 공급자 상태 메서드를 노출합니다.
IWPCProviderSupport 타사 공급자가 현재 사용 가능한 공급자를 쿼리할 수 있도록 하는 메서드를 노출합니다.
IWPCSettings 사용자의 일반 설정에 액세스합니다.
IWPCWebSettings 사용자의 웹 제한 설정에 액세스합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wpcevent.h 헤더

아티클2023. 03. 08.

이 헤더는 자녀 보호에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 자녀

wpcevent.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

열거형

WPC_ARGS_CONVERSATIONINITEVENT 대화 시작에 대한 정보를 나타냅니다.
WPC_ARGS_CONVERSATIONJOINEVENT 기존 대화에 참가하는 방법에 대한 정보를 나타냅니다.
WPC_ARGS_CONVERSATIONLEAVEEVENT 대화 종료에 대한 정보를 나타냅니다.
WPC_ARGS_CUSTOMEVENT 일반 이벤트에서 다루지 않는 사용자 정의 이벤트에 대한 정보를 나타냅니다.
WPC_ARGS_EMAILCONTACTEVENT 전자 메일을 사용하여 다른 사람에게 연락하는 방법에 대한 정보를 나타냅니다.
WPC_ARGS_EMAILRECEIEVEDEVENT 받은 전자 메일 메시지에 대한 정보를 나타냅니다.
WPC_ARGS_EMAILSENTEVENT 전송된 전자 메일 메시지에 대한 정보를 나타냅니다.
WPC_ARGS_FILEDOWNLOADEVENT 다운로드된 파일에 대한 정보를 나타냅니다.

WPC_ARGS_GAMESTARTEVENT 컴퓨터 게임의 시작에 대한 정보를 나타냅니다.
WPC_ARGS_IMCONTACTEVENT 인스턴트 메시징 애플리케이션을 사용하여 다른 사람에게 연락하는 방법에 대한 정보를 나타냅니다.
WPC_ARGS_IMFEATUREEVENT 인스턴트 메시징 상호 작용의 기능에 대한 정보를 나타냅니다.
WPC_ARGS_MEDIADOWNLOADEVENT 미디어 파일 다운로드에 대한 정보를 나타냅니다.
WPC_ARGS_MEDIAPLAYBACKEVENT 미디어 파일의 재생에 대한 정보를 나타냅니다.
WPC_ARGS_SAFERAPPBLOCKED 차단되는 안전한 애플리케이션에 대한 정보를 나타냅니다.
WPC_ARGS_SETTINGSCHANGEEVENT 사용자가 변경한 설정에 대한 정보를 나타냅니다.
WPC_ARGS_URLVISITEVENT 본 웹 사이트의 주소 URL에 대한 정보를 나타냅니다.
WPC_MEDIA_EXPLICIT 미디어 파일의 명시적 등급에 대한 정보를 나타냅니다.
WPC_MEDIA_TYPE 액세스하는 미디어 파일의 형식에 대한 정보를 나타냅니다.
WPCFLAG_IM_FEATURE 인스턴트 메시징 상호 작용 중에 액세스하는 기능에 대한 정보를 나타냅니다.
WPCFLAG_IM_LEAVE 참가자가 인스턴트 메시징 상호 작용을 떠나는 시기에 대한 정보를 나타냅니다.

WPCFLAG_ISBLOCKED 사용이 차단된 이벤트 및 현재 위치에 있는 컨트롤에 대한 정보를 나타냅니다.
WPCFLAG_LOGOFF_TYPE 사용되는 로그오프 방법의 유형에 대한 정보를 나타냅니다.

wrdsgraphicschannels.h 헤더

아티클2023. 08. 28.

이 헤더는 원격 데스크톱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 데스크톱 서비스

wrdsgraphicschannels.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWRdsGraphicsChannel 이 인터페이스는 RemoteFX 그래픽 서비스에서 가상 그래픽 채널로 데이터를 보내고 받는 데 사용됩니다.
IWRdsGraphicsChannelEvents 이 인터페이스는 그래픽 가상 채널과 관련된 알림을 받습니다.
IWRdsGraphicsChannelManager 이 인터페이스는 RemoteFX 그래픽 서비스 API에서 그래픽 데이터를 원격화하는 데 필요한 그래픽 가상 채널을 만드는 데 사용됩니다.

열거형

WRdsGraphicsChannelType IWRdsGraphicsChannelManager::CreateChannel 메서드에서 만들 그래픽 가상 채널 유형을 지정하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ws2atm.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows 소켓 2에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 소켓 2

ws2atm.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

ATM_ADDRESS ATM_ADDRESS 구조체는 ATM 기반 소켓에 대한 ATM 주소 데이터를 보유합니다.
ATM_BHLI ATM_BHLI 구조체는 연결된 ATM 소켓에 대한 B-HLI 정보를 식별하는 데 사용됩니다.
ATM_BLLI ATM_BLLI 구조는 연결된 ATM 소켓에 대한 B-LLI 정보를 식별하는 데 사용됩니다.
sockaddr_atm Windows 소켓 sockaddr_atm 구조체는 ATM 소켓에 대한 소켓 주소 정보를 저장합니다.

ws2bth.h 헤더

아티클2023. 08. 28.

이 헤더는 Bluetooth에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Bluetooth

ws2bth.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

BTH_QUERY_DEVICE BTH_QUERY_DEVICE 구조는 Bluetooth 디바이스의 존재를 쿼리할 때 사용됩니다.
BTH_QUERY_SERVICE BTH_QUERY_SERVICE 구조는 Bluetooth 서비스를 쿼리하는 데 사용됩니다.
BTH_SET_SERVICE 지정된 Bluetooth 서비스에 대한 서비스 정보를 제공합니다.
SOCKADDR_BTH SOCKADDR_BTH 구조는 주소 패밀리 AF_BTH 정의된 Bluetooth 소켓 작업과 함께 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

ws2def.h 헤더

아티클2023. 03. 16.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- IP 도우미
- 네트워크 드라이버, Windows Vista 이상
- Windows 소켓 2

ws2def.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

ADDRINFO_DNS_SERVER Winsock API에 사용되는 사용자 지정 DNS(도메인 이름 시스템) 서버를 나타냅니다.
ADDRINFOA getaddrinfo 함수에서 호스트 주소 정보를 보유하는 데 사용됩니다.
ADDRINFOEX2A 정식 이름과 정규화된 도메인 이름이 모두 요청된 경우 GetAddrInfoEx 함수에서 호스트 주소 정보를 보유하는 데 사용됩니다. (ANSI)
ADDRINFOEX2W 정식 이름과 정규화된 도메인 이름이 모두 요청된 경우 GetAddrInfoEx 함수에서 호스트 주소 정보를 보유하는 데 사용됩니다. (유니코드)
ADDRINFOEX3 특정 네트워크 인터페이스가 요청되었을 때 GetAddrInfoEx 함수에서 호스트 주소 정보를 보유하는 데 사용됩니다. (addrinfoex3W)
ADDRINFOEX4 특정 네트워크 인터페이스가 요청되었을 때 GetAddrInfoEx 함수에서 호스트 주소 정보를 보유하는 데 사용됩니다. (addrinfoex4W)

ADDRINFOEX5 특정 네트워크 인터페이스가 요청되었을 때 GetAddrInfoExW 함수에서 호스트 주소 정보를 보유하는 데 사용됩니다. (ADDRINFOEX5)
ADDRINFOEX6 특정 네트워크 인터페이스가 요청되었을 때 GetAddrInfoExW 함수에서 호스트 주소 정보를 보유하는 데 사용됩니다. (ADDRINFOEX6)
ADDRINFOEXA GetAddrInfoEx 함수에서 호스트 주소 정보를 보유하는 데 사용됩니다. (ANSI)
ADDRINFOEXW GetAddrInfoEx 함수에서 호스트 주소 정보를 보유하는 데 사용됩니다. (유니코드)
ADDRINFOW GetAddrInfoW 함수에서 호스트 주소 정보를 보유하는 데 사용됩니다.
CSADDR_INFO CSADDR_INFO(ws2def.h) 구조체에는 소켓, 네트워크 서비스 또는 네임스페이스 공급자에 대한 Windows 소켓 주소 정보가 포함됩니다.
SOCKADDR SOCKADDR 구조체는 전송 주소를 지정하는 제네릭 구조체입니다.
SOCKADDR_IN SOCKADDR_IN 구조는 AF_INET 주소 패밀리의 전송 주소와 포트를 지정합니다.
SOCKADDR_STORAGE_LH SOCKADDR_STORAGE 구조체는 전송 주소를 지정하는 제네릭 구조체입니다. (SOCKADDR_STORAGE_LH)
SOCKADDR_STORAGE_XP SOCKADDR_STORAGE 구조체는 전송 주소를 지정하는 제네릭 구조체입니다. S
SOCKET_ADDRESS SOCKET_ADDRESS 구조는 프로토콜별 주소 정보를 저장합니다.

SOCKET_ADDRESS_LIST SOCKET_ADDRESS_LIST 구조체는 가변 크기의 전송 주소 목록을 정의합니다.
SOCKET_PROCESSOR_AFFINITY 소켓과 RSS 프로세서 코어 및 NUMA 노드 간의 연결을 포함합니다.
WSABUF WSABUF 구조를 사용하면 일부 Winsock 함수에서 사용하는 데이터 버퍼를 만들거나 조작할 수 있습니다.
WSACMSGHDR CMSGHDR 구조체는 데이터그램과 연결된 컨트롤 데이터 개체의 헤더를 정의합니다.
WSAMSG WSARecvMsg 및 WSASendMsg 함수와 함께 사용하여 메시지 데이터를 저장하는 데 사용되는 버퍼 배열뿐만 아니라 연결 및 연결되지 않은 소켓에 대한 주소 및 선택적 제어 정보를 저장합니다.

열거형

SCOPE_LEVEL SCOPE_LEVEL 열거형은 IP_ADAPTER_ADDRESSES 구조와 함께 IPv6 주소의 범위 수준을 식별하는 데 사용됩니다.

ws2ipdef.h 헤더

아티클2023. 11. 22.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- IP 도우미
- 네트워크 드라이버, Windows Vista 이상
- Windows 소켓 2

ws2ipdef.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

IN6_ADDR_EQUAL
IN6_IS_ADDR_ANYCAST
IN6_IS_ADDR_EUI64
IN6_IS_ADDR_GLOBAL
IN6_IS_ADDR_LINKLOCAL
IN6_IS_ADDR_LOOPBACK
IN6_IS_ADDR_MC_GLOBAL
IN6_IS_ADDR_MC_LINKLOCAL
IN6_IS_ADDR_MC_NODELOCAL
IN6_IS_ADDR_MC_ORGLOCAL
IN6_IS_ADDR_MC_SITELOCAL

IN6_IS_ADDR_MULTICAST
IN6_IS_ADDR_SITELOCAL
IN6_IS_ADDR_SUBNET_RESERVED_ANYCAST
IN6_IS_ADDR_SUBNET_ROUTER_ANYCAST
IN6_IS_ADDR_UNSPECIFIED
IN6_IS_ADDR_V4COMPAT
IN6_IS_ADDR_V4MAPPED
IN6_IS_ADDR_V4TRANSLATED
IN6_SET_ADDR_LOOPBACK
IN6_SET_ADDR_UNSPECIFIED
IN6ADDR_ISANY
IN6ADDR_ISEQUAL
IN6ADDR_ISLOOPBACK
IN6ADDR_ISUNSPECIFIED
IN6ADDR_SETANY
IN6ADDR_SETLOOPBACK

구조체

GROUP_FILTER 멀티캐스트 IPv6 또는 IPv4 주소에 대한 멀티캐스트 필터링 매개 변수를 제공합니다.
GROUP_REQ IPv6 또는 IPv4 주소에 대한 멀티캐스트 그룹 정보를 제공합니다.
GROUP_SOURCE_REQ 원본 IP 주소를 포함하는 IPv6 또는 IPv4 주소에 대한 멀티캐스트 그룹 정보를 제공합니다.
ICMP_ERROR_INFO 수신된 ICMP 오류 정보를 저장하는 데 사용됩니다.
IN_PKTINFO in_pktinfo 구조는 수신된 패킷 주소 정보를 저장하는 데 사용되며 Windows에서 수신된 패킷에 대한 정보를 반환하는 데 사용되며 패킷을 보내는 데 사용할 로컬 IPv4 주소를 지정할 수도 있습니다.
IN6_PKTINFO in6_pktinfo 구조는 수신된 IPv6 패킷 주소 정보를 저장하는 데 사용되며 Windows에서 수신된 패킷에 대한 정보를 반환하는 데 사용되며 패킷을 보내는 데 사용할 로컬 IPv6 주소를 지정할 수도 있습니다.
INTERFACE_INFO INTERFACE_INFO 구조체는 인터페이스 IP 주소에 대한 정보를 얻기 위해 SIO_GET_INTERFACE_LIST ioctl 명령과 함께 사용됩니다.
INTERFACE_INFO_EX INTERFACE_INFO_EX 구조는 인터페이스 IP 주소에 대한 정보를 얻기 위해 SIO_GET_INTERFACE_LIST IOCTL 명령과 함께 사용됩니다.
IP_MREQ ip_mreq 구조는 IPv4 주소에 대한 멀티캐스트 그룹 정보를 제공합니다.
IP_MREQ_SOURCE ip_mreq_source 구조는 IPv4 주소에 대한 멀티캐스트 그룹 정보를 제공합니다.
IP_MSFILTER

ip_msfilter 구조는 IPv4 주소에 대한 멀티캐스트 필터링 매개 변수를 제공합니다.
IPV6_MREQ ipv6_mreq 구조는 IPv6 주소에 대한 멀티캐스트 그룹 정보를 제공합니다.
sockaddr_gen 제네릭 소켓 주소 정보를 제공하며 INTERFACE_INFO 구조와 함께 사용됩니다.
SOCKADDR_IN6_LH SOCKADDR_IN6_LH(ws2ipdef.h) 구조체는 AF_INET6 주소 패밀리의 전송 주소와 포트를 지정합니다.
sockaddr_in6_old sockaddr_in6_old(ws2ipdef.h) 구조는 선택한 프로토콜에 따라 달라집니다.
SOCKADDR_IN6_PAIR 원본 및 대상 주소 쌍을 나타내는 IP 주소 쌍에 대한 포인터를 포함합니다.
SOCKADDR_IN6_W2KSP1 SOCKADDR_IN6_W2KSP1(ws2ipdef.h) 구조체는 AF_INET6 주소 패밀리의 전송 주소와 포트를 지정합니다.
SOCKADDR_INET IPv4, IPv6 주소 또는 주소 패밀리를 포함합니다.

열거형

MULTICAST_MODE_TYPE 멀티캐스트 그룹 주소에 대한 필터 모드를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

ws2spi.h 헤더

아티클2023. 03. 15.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Vista 이전의 네트워크 드라이버
- Windows 소켓 2

ws2spi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

NSPStartup DLL 진입점 목록과 같은 공급자에 대한 동적 정보를 검색합니다.
WPUCloseEvent WPUCloseEvent 함수는 열린 이벤트 개체 핸들을 닫습니다.
WPUCloseSocketHandle WPUCloseSocketHandle 함수는 기존 소켓 핸들을 닫습니다.
WPUCloseThread WPUCloseThread 함수는 WPUOpenCurrentThread 호출로 열린 스레드를 닫습니다.
WPUCompleteOverlappedRequest WPUCompleteOverlappedRequest 함수는 겹치는 I/O 작업에 대해 겹치는 I/O 완료 알림을 수행합니다.
WPUCreateEvent WPUCreateEvent 함수는 새 이벤트 개체를 만듭니다.
WPUCreateSocketHandle WPUCreateSocketHandle 함수는 새 소켓 핸들을 만듭니다.
WPUFDIsSet WPUFDIsSet 함수는 지정된 소켓 핸들의 멤버 자격을 확인합니다.

WPUGetProviderPath WPUGetProviderPath 함수는 지정된 공급자에 대한 DLL 경로를 검색합니다.
WPUModifyIFSHandle WPUModifyIFSHandle 함수는 Ws2_32.dll 수정된 IFS 핸들을 받습니다.
WPUOpenCurrentThread WPUOpenCurrentThread 함수는 계층화된 서비스 공급자에서 겹치는 함수와 함께 사용할 수 있는 현재 스레드에 대한 핸들을 엽니다.
WPUPostMessage WPUPostMessage 함수는 이전 버전의 WSOCK32.dll 이전 버전과의 호환성을 유지하는 방식으로 표준 Windows PostMessage 함수를 수행합니다.
WPUQueryBlockingCallback WPUQueryBlockingCallback 함수는 차단 작업을 서비스하는 동안 서비스 공급자가 주기적으로 호출해야 하는 콜백 함수에 대한 포인터를 반환합니다.
WPUQuerySocketHandleContext WPUQuerySocketHandleContext 함수는 지정된 소켓 핸들과 연결된 컨텍스트 값을 쿼리합니다.
WPUQueueApc WPUQueueApc 함수는 겹치는 I/O 완료 루틴의 호출을 용이하게 하기 위해 지정된 스레드에 사용자 모드 APC(비동기 프로시저 호출)를 큐에 대기합니다.
WPUResetEvent WPUResetEvent 함수는 지정된 이벤트 개체의 상태를 서명되지 않은 상태로 다시 설정합니다. 이 호출은 인터럽트 컨텍스트 내에서 사용하기에 안전합니다.
WPUSetEvent WPUSetEvent 함수는 지정된 이벤트 개체의 상태를 신호로 설정합니다. 이 호출은 인터럽트 컨텍스트 내에서 사용하기에 안전합니다.
WSAAdvertiseProvider 모든 적격 클라이언트에서 특정 네임스페이스 버전 2 공급자를 사용할 수 있게 합니다.
WSAProviderCompleteAsyncCall 네임스페이스 버전 2 공급자에 대한 비동기 호출이 완료되면 클라이언트에 알린다.

WSAUnadvertiseProvider 클라이언트에서 특정 네임스페이스 버전 2 공급자를 더 이상 사용할 수 없게 만듭니다.
WSCDeinstallProvider 시스템 구성 데이터베이스에서 지정된 전송 공급자를 제거합니다.
WSCDeinstallProvider32 시스템 구성 데이터베이스에서 지정된 32비트 전송 공급자를 제거합니다.
WSCEnableNSProvider 지정된 네임스페이스 공급자의 상태를 변경합니다.
WSCEnableNSProvider32 지정된 32비트 네임스페이스 공급자를 사용하거나 사용하지 않도록 설정합니다.
WSCEnumNameSpaceProviders32 사용 가능한 32비트 네임스페이스 공급자에 대한 정보를 반환합니다. 참고: 이 호출은 64비트 플랫폼에서 사용하기 위해 엄격하게 32비트 버전의 WSAEnumNameSpaceProviders입니다. 64비트 프로세스가 32비트 카탈로그에 액세스할 수 있도록 제공됩니다. .
WSCEnumNameSpaceProvidersEx32 사용 가능한 32비트 네임스페이스 공급자에 대한 정보를 검색합니다.
WSCEnumProtocols WSCEnumProtocols 함수는 사용 가능한 전송 프로토콜에 대한 정보를 검색합니다.
WSCEnumProtocols32 사용 가능한 전송 프로토콜에 대한 정보를 검색합니다. 참고: 이 호출은 64비트 플랫폼에서 사용하기 위해 엄격하게 32비트 버전의 WSCEnumProtocols입니다. 64비트 프로세스가 32비트 카탈로그에 액세스할 수 있도록 제공됩니다. .
WSCGetApplicationCategory 애플리케이션과 연결된 LSP(계층화된 서비스 공급자) 범주를 검색합니다.
WSCGetProviderInfo LSP(계층화된 서비스 공급자)에 대한 정보 클래스와 연결된 데이터를 검색합니다.

WSCGetProviderInfo32 32비트 LSP(계층화된 서비스 공급자)에 대한 정보 클래스와 연결된 데이터를 검색합니다. 참고: 이 호출은 64비트 플랫폼에서 사용하기 위해 엄격하게 32비트 버전의 WSCGetProviderInfo입니다.
WSCGetProviderPath WSCGetProviderPath 함수는 지정된 공급자에 대한 DLL 경로를 검색합니다.
WSCGetProviderPath32 지정된 32비트 공급자의 DLL 경로를 검색합니다. 참고: 이 호출은 64비트 플랫폼에서 사용하기 위해 엄격하게 32비트 버전의 WSCGetProviderPath입니다. 64비트 프로세스가 32비트 카탈로그에 액세스할 수 있도록 제공됩니다. .
WSCInstallNameSpace 네임스페이스 공급자를 설치합니다. (WSCInstallNameSpace)
WSCInstallNameSpace32 지정된 32비트 네임스페이스 공급자를 설치합니다. (WSCInstallNamespace32)
WSCInstallNameSpaceEx 네임스페이스 공급자를 설치합니다. (WSCInstallNameSpaceEx)
WSCInstallNameSpaceEx32 지정된 32비트 네임스페이스 공급자를 설치합니다. (WSCInstallNameSpaceEx32)
WSCInstallProvider 시스템 구성 데이터베이스에 지정된 전송 공급자를 설치합니다.
WSCInstallProvider64_32 지정된 전송 서비스 공급자를 64비트 컴퓨터의 32비트 및 64비트 시스템 구성 데이터베이스에 설치합니다.
WSCInstallProviderAndChains 지정된 32비트 전송 공급자와 특정 프로토콜 체인을 32비트 컴퓨터의 Winsock 2 시스템 구성 데이터베이스에 설치합니다.
WSCInstallProviderAndChains64_32 지정된 전송 공급자 및 해당 특정 프로토콜 체인을 64비트 컴퓨터의 32비트 및 64비트 Winsock 2 시스템 구성 데이터베이스에 모두 설치합니다.

WSCInstallQOSTemplate 시스템 구성 데이터베이스에 지정된 QoS 템플릿을 설치합니다.
WSCRemoveQOSTemplate 시스템 구성 데이터베이스에서 지정된 QoS 템플릿을 제거합니다.
WSCSetApplicationCategory 애플리케이션과 연결된 허용된 LSP(계층화된 서비스 공급자) 범주를 설정합니다.
WSCSetProviderInfo LSP(계층화된 서비스 공급자)에 대해 지정된 정보 클래스의 데이터 값을 설정합니다.
WSCSetProviderInfo32 LSP(계층화된 서비스 공급자)의 지정된 정보 클래스에 대한 데이터 값을 설정합니다.
WSCUnInstallNameSpace 표시된 이름 공간 공급자를 제거합니다.
WSCUnInstallNameSpace32 특정 32비트 네임스페이스 공급자를 제거합니다.
WSCUpdateProvider 시스템 구성 데이터베이스에서 지정된 전송 공급자를 수정합니다.
WSCUpdateProvider32 시스템 구성 데이터베이스에서 지정된 32비트 전송 공급자를 수정합니다. 참고: 이 호출은 64비트 플랫폼에서 사용하기 위해 엄격하게 32비트 버전의 WSCUpdateProvider입니다. 64비트 프로세스가 32비트 카탈로그에 액세스할 수 있도록 제공됩니다. .
WSPStartup WSPStartup 함수는 클라이언트에서 WINDOWS 소켓 SPI(서비스 공급자 인터페이스)의 사용을 시작합니다.

콜백 함수

LPNSPCLEANUP 특정 Windows 소켓 네임스페이스 서비스 공급자의 사용을 종료합니다.
LPNSPGETSERVICECLASSINFO 네임스페이스 공급자와 관련된 모든 관련 클래스 정보(스키마)를 검색합니다.
LPNSPINSTALLSERVICECLASS NSPInstallServiceClass 함수는 네임스페이스 공급자 내에서 서비스 클래스 스키마를 등록합니다.
LPNSPIOCTL IOCTL을 네임스페이스 서비스 공급자에게 보냅니다.
LPNSPLOOKUPSERVICEBEGIN WSAQUERYSET 구조에 포함된 정보로 제한되는 클라이언트 쿼리를 시작합니다.
LPNSPLOOKUPSERVICEEND NSPLookupServiceBegin 및 NSPLookupServiceNext에 대한 이전 호출 후 핸들을 해제하기 위해 호출됩니다.
LPNSPLOOKUPSERVICENEXT 요청된 서비스 정보를 검색하기 위해 NSPLookupServiceBegin에 대한 이전 호출에서 핸들을 가져온 후 호출됩니다.
LPNSPREMOVESERVICECLASS 네임스페이스에서 지정된 서비스 클래스를 영구적으로 제거합니다.
LPNSPSETSERVICE 네임스페이스 내에서 서비스 인스턴스를 등록하거나 등록 취소합니다.
LPNSPV2CLEANUP 네임스페이스 서비스 공급자 버전 2(NSPv2) 공급자에게 클라이언트 세션이 종료되었음을 알 수 있습니다.
LPNSPV2CLIENTSESSIONRUNDOWN 네임스페이스 서비스 공급자 버전 2(NSPv2) 공급자에게 클라이언트 세션이 종료되고 있음을 알 수 있습니다.

LPNSPV2LOOKUPSERVICEBEGIN WSAQUERYSET2 구조체에 포함된 정보로 제한되는 네임스페이스 버전 2 서비스 공급자의 클라이언트 쿼리를 시작합니다.
LPNSPV2LOOKUPSERVICEEND NSPv2LookupServiceBegin 및 NSPv2LookupServiceNextEx에 대한 이전 호출 후 핸들을 해제하기 위해 호출됩니다.
LPNSPV2LOOKUPSERVICENEXTEX 네임스페이스 버전 2 서비스 공급자에서 요청된 정보를 검색하기 위해 NSPv2LookupServiceBegin에 대한 이전 호출에서 핸들을 가져온 후 호출됩니다.
LPNSPV2SETSERVICEEX 네임스페이스 서비스 공급자 version-2(NSPv2) 공급자의 네임스페이스 내에서 이름 또는 서비스 인스턴스를 등록하거나 등록 취소합니다.
LPNSPV2STARTUP 네임스페이스 서비스 공급자 버전 2(NSPv2) 공급자에게 새 클라이언트 프로세스가 공급자 사용을 시작하도록 알 수 있습니다.
LPWSPACCEPT LPWSPAccept 함수는 조건부 함수의 반환 값에 따라 연결을 조건부로 허용합니다.
LPWSPADDRESSTOSTRING LPWSPAddressToString 함수는 sockaddr 구조체의 모든 구성 요소를 사람이 읽을 수 있는 주소의 숫자 문자열 표현으로 변환합니다. 이는 주로 표시 목적으로 사용됩니다.
LPWSPASYNCSELECT LPWSPAsyncSelect 함수는 소켓에 대한 네트워크 이벤트의 Windows 메시지 기반 이벤트 알림을 요청합니다.
LPWSPBIND LPWSPBind 함수는 로컬 주소(즉, 이름)를 소켓과 연결합니다.
LPWSPCANCELBLOCKINGCALL LPWSPCancelBlockingCall 함수는 현재 진행 중인 차단 호출을 취소합니다.

LPWSPCLEANUP LPWSPCleanup 함수는 Windows 소켓 서비스 공급자의 사용을 종료합니다.
LPWSPCLOSESOCKET LPWSPCloseSocket 함수는 소켓을 닫습니다.
LPWSPCONNECT LPWSPConnect 함수는 피어에 대한 연결을 설정하고, 연결 데이터를 교환하며, 제공된 흐름 사양에 따라 필요한 서비스 품질을 지정합니다.
LPWSPDUPLICATESOCKET LPWSPDuplicateSocket 함수는 공유 소켓에 대한 새 소켓 설명자를 만드는 데 사용할 수 있는 WSAPROTOCOL_INFO 구조를 반환합니다.
LPWSPENUMNETWORKEVENTS LPWSPEnumNetworkEvents 함수는 표시된 소켓에 대한 네트워크 이벤트의 발생을 보고합니다.
LPWSPEVENTSELECT **LPWSPEventSelect** 함수는 제공된 네트워크 이벤트 집합과 연결할 이벤트 개체를 지정합니다.
LPWSPGETOVERLAPPEDRESULT LPWSPGetOverlappedResult 함수는 지정된 소켓에서 겹치는 작업의 결과를 반환합니다.
LPWSPGETPEERNAME LPWSPGetPeerName 함수는 소켓이 연결된 피어의 주소를 가져옵니다.
LPWSPGETQOSBYNAME LPWSPGetQOSByName 함수는 명명된 템플릿을 기반으로 QOS 구조를 초기화하거나 사용 가능한 템플릿 이름의 열거형을 검색합니다.
LPWSPGETSOCKNAME LPWSPGetSockName 함수는 소켓의 로컬 이름을 가져옵니다.
LPWSPGETSOCKOPT LPWSPGetSockOpt 함수는 소켓 옵션을 검색합니다.

LPWSPIOCTL LPWSPIoctl 함수는 소켓의 모드를 제어합니다.
LPWSPJOINLEAF LPWSPJoinLeaf 함수는 리프 노드를 멀티포인트 세션에 조인하고, 연결 데이터를 교환하며, 제공된 흐름 사양에 따라 필요한 서비스 품질을 지정합니다.
LPWSPLISTEN LPWSPListen 함수는 들어오는 연결을 수신 대기하는 소켓을 설정합니다.
LPWSPRECV LPWSPRecv 함수는 소켓에서 데이터를 받습니다.
LPWSPRECVDISCONNECT LPWSPRecvDisconnect 함수는 소켓에서 수신을 종료하고 소켓이 연결 지향인 경우 연결 끊기 데이터를 검색합니다.
LPWSPRECVFROM LPWSPRecvFrom 함수는 데이터그램을 수신하고 원본 주소를 저장합니다.
LPWSPSELECT LPWSPSelect 함수는 하나 이상의 소켓 상태를 결정합니다.
LPWSPSEND LPWSPSend 함수는 연결된 소켓에 데이터를 보냅니다.
LPWSPSENDDISCONNECT LPWSPSendDisconnect 함수는 소켓에 대한 연결 종료를 시작하고 연결 끊기 데이터를 보냅니다.
LPWSPSENDTO WSPSendTo 함수는 겹치는 I/O를 사용하여 특정 대상으로 데이터를 보냅니다.
LPWSPSETSOCKOPT LPWSPSetSockOpt 함수는 소켓 옵션을 설정합니다.
LPWSPSHUTDOWN LPWSPShutdown 함수는 소켓에서 송신 및/또는 수신을 사용하지 않도록 설정합니다.

LPWSPSOCKET LPWSPSocket 함수는 소켓을 만듭니다.
LPWSPSTRINGTOADDRESS WSPStringToAddress 함수는 사람이 읽을 수 있는 숫자 문자열을 해당 구조를 사용하는 Windows 소켓 루틴에 전달하는 데 적합한 소켓 주소 구조(sockaddr)로 변환합니다.

구조체

NSP_ROUTINE 네임스페이스 서비스 공급자 버전 1(NSPv1) 공급자가 구현한 함수에 대한 정보를 포함합니다.
NSPV2_ROUTINE 네임스페이스 서비스 공급자 버전 2(NSPv2) 공급자가 구현한 함수에 대한 정보를 포함합니다.
WSATHREADID WSATHREADID 구조를 사용하면 공급자가 WPUQueueApc 함수를 사용하여 APC(비동기 프로시저 호출)를 큐에 대기할 수 있는 스레드를 식별할 수 있습니다.
WSC_PROVIDER_AUDIT_INFO Windows 소켓 2의 LSP(계층화된 서비스 공급자) 항목에 대한 감사 정보를 포함합니다.
WSPDATA WSPDATA 구조에는 서비스 공급자 정보가 포함됩니다.
WSPPROC_TABLE 서비스 공급자 함수에 대한 포인터 테이블을 포함합니다.
WSPUPCALLTABLE 서비스 공급자 업콜 함수에 대한 포인터 테이블을 포함합니다.

열거형

WSC_PROVIDER_INFO_TYPE 열거형 유형은 Windows 소켓 2에서 LSP(계층화된 서비스 프로토콜)의 정보 클래스를 지정하는 데 사용됩니다.

ws2tcpip.h 헤더

아티클2023. 08. 28.

이 헤더는 Windows 소켓 2에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 소켓 2

ws2tcpip.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

freeaddrinfo getaddrinfo 함수가 addrinfo 구조에서 동적으로 할당하는 주소 정보를 해제합니다.
FreeAddrInfoEx FreeAddrInfoEx 함수(ws2tcpip.h)는 GetAddrInfoEx 함수가 addrinfoex 구조에서 동적으로 할당하는 주소 정보를 해제합니다.
FreeAddrInfoExW FreeAddrInfoExW(유니코드) 함수(ws2tcpip.h)는 GetAddrInfoEx 함수가 addrinfoex 구조에 동적으로 할당하는 주소 정보를 해제합니다.
FreeAddrInfoW GetAddrInfoW 함수가 addrinfoW 구조에서 동적으로 할당하는 주소 정보를 해제합니다.
gai_strerrorA gai_strerror 함수는 getaddrinfo 함수에서 반환된 EAI_* 오류를 기반으로 오류 메시지를 인쇄하는 데 도움이 됩니다. (ANSI)
gai_strerrorW gai_strerror 함수는 getaddrinfo 함수에서 반환된 EAI_* 오류를 기반으로 오류 메시지를 인쇄하는 데 도움이 됩니다. (유니코드)
getaddrinfo ANSI 호스트 이름에서 주소로 프로토콜 독립적 변환을 제공합니다.
GetAddrInfoExA

요청을 처리해야 하는 네임스페이스 공급자를 한정하기 위한 추가 매개 변수를 사용하여 프로토콜 독립적 이름 확인을 제공합니다. (ANSI)
GetAddrInfoExCancel GetAddrInfoEx 함수에 의한 비동기 작업을 취소합니다.
GetAddrInfoExOverlappedResult GetAddrInfoEx 함수에 대한 비동기 작업에서 사용하는 OVERLAPPED 구조체의 반환 코드를 가져옵니다.
GetAddrInfoExW 요청을 처리해야 하는 네임스페이스 공급자를 한정하기 위한 추가 매개 변수를 사용하여 프로토콜 독립적 이름 확인을 제공합니다. (유니코드)
GetAddrInfoW 유니코드 호스트 이름에서 주소로 프로토콜 독립적 변환을 제공합니다.
getipv4sourcefilter IPv4 소켓에 대한 멀티캐스트 필터 상태를 검색합니다.
getnameinfo 주소에서 ANSI 호스트 이름으로, 포트 번호에서 ANSI 서비스 이름으로 프로토콜 독립적 이름 확인을 제공합니다.
GetNameInfoW 주소에서 유니코드 호스트 이름으로, 포트 번호에서 유니코드 서비스 이름으로 프로토콜 독립적 이름 확인을 제공합니다.
getsourcefilter IPv4 또는 IPv6 소켓의 멀티캐스트 필터 상태를 검색합니다.
inet_ntop InetNtop 함수는 IPv4 또는 IPv6 인터넷 네트워크 주소를 인터넷 표준 형식의 문자열로 변환합니다. 이 함수의 ANSI 버전은 inet_ntop. (InetNtopA 또는 inet_ntop)
inet_pton InetPton 함수는 표준 텍스트 프레젠테이션 형식의 IPv4 또는 IPv6 인터넷 네트워크 주소를 숫자 이진 형식으로 변환합니다. 이 함수의 ANSI 버전은 inet_pton. (InetPtonA 또는 inet_pton)

InetNtopW InetNtop 함수는 IPv4 또는 IPv6 인터넷 네트워크 주소를 인터넷 표준 형식의 문자열로 변환합니다. 이 함수의 ANSI 버전은 inet_ntop. (InetNtopW)
InetPtonW InetPton 함수는 표준 텍스트 프레젠테이션 형식의 IPv4 또는 IPv6 인터넷 네트워크 주소를 숫자 이진 형식으로 변환합니다. 이 함수의 ANSI 버전은 inet_pton. (InetPtonW)
SetAddrInfoExA 이름, 서비스 이름 및 연결된 주소를 특정 네임스페이스 공급자에 등록하거나 등록 취소합니다. (ANSI)
SetAddrInfoExW 이름, 서비스 이름 및 연결된 주소를 특정 네임스페이스 공급자에 등록하거나 등록 취소합니다. (유니코드)
setipv4sourcefilter IPv4 소켓의 멀티캐스트 필터 상태를 설정합니다.
setsourcefilter IPv4 또는 IPv6 소켓의 멀티캐스트 필터 상태를 설정합니다.
WSADeleteSocketPeerTargetName 피어 대상 이름과 소켓의 IP 주소 간의 연결을 제거합니다. 성공적으로 반환된 후에는 IP 주소와 대상 이름 간에 나중에 연결되지 않습니다.
WSAGetFailConnectOnIcmpError **TCP_FAIL_CONNECT_ON_ICMP_ERROR** 소켓 옵션의 상태를 쿼리합니다.
WSAGetIcmpErrorInfo 연결 설정 중 TCP 소켓에서 수신된 ICMP 오류에 대한 정보를 검색합니다.
WSAGetIPUserMtu 소켓에 대한 사용자 정의 IP 계층 MTU를 검색합니다.
WSAGetRecvIPEcn TBD(WSAGetRecvIPEcn)

WSAGetUdpRecvMaxCoalescedSize UDP 소켓에 대해 받은 병합된 메시지의 최대 크기를 검색합니다.
WSAGetUdpSendMessageSize UDP 소켓의 구분 메시지 크기를 검색합니다.
WSAImpersonateSocketPeer 애플리케이션 수준 권한 부여를 수행하기 위해 소켓 피어에 해당하는 보안 주체를 가장하는 데 사용됩니다.
WSAQuerySocketSecurity 소켓의 연결에 적용되는 보안에 대한 정보를 쿼리합니다.
WSARevertImpersonation 소켓 피어의 가장을 종료합니다. WSAImpersonateSocketPeer를 호출하고 액세스 검사를 완료한 후에 호출해야 합니다.
WSASetFailConnectOnIcmpError **TCP_FAIL_CONNECT_ON_ICMP_ERROR** 소켓 옵션의 상태를 설정합니다.
WSASetIPUserMtu 소켓에서 사용자 정의 IP 계층 MTU를 설정합니다.
WSASetRecvIPEcn IP 스택이 수신된 데이터그램에서 서비스 IPv4 헤더 필드(또는 트래픽 클래스 IPv6 헤더 필드)의 ECN(명시적 정체 알림) 코드포인트가 포함된 메시지로 컨트롤 버퍼를 채울지 여부를 지정합니다.
WSASetSocketPeerTargetName 피어 IP 주소에 해당하는 SPN(피어 대상 이름)을 지정하는 데 사용됩니다. 이 대상 이름은 인증해야 하는 피어를 안전하게 식별하기 위해 클라이언트 애플리케이션에서 지정해야 합니다.
WSASetSocketSecurity 소켓에 대한 보안을 사용하도록 설정하고 적용합니다.
WSASetUdpRecvMaxCoalescedSize UDP 소켓에 설정된 병합된 메시지의 최대 크기를 설정합니다.

WSASetUdpSendMessageSize UDP 소켓의 구분 메시지 크기를 설정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wsbapp.h 헤더

아티클2023. 08. 28.

이 헤더는 Windows Server Backup에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Server 백업

wsbapp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWsbApplicationAsync 비동기 작업의 진행률을 모니터링하고 제어하는 메서드를 정의합니다.
IWsbApplicationBackupSupport 애플리케이션의 VSS 기록기 구성 요소의 일관성을 확인하는 메서드를 정의합니다.
IWsbApplicationRestoreSupport 애플리케이션별 복원 작업을 수행하기 위한 메서드를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wsbonline.h 헤더

아티클2023. 08. 28.

이 헤더는 Windows Server Backup에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows Server 백업

wsbonline.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

DeregisterOnlineBackupFromWindowsServerBackup 이미 등록된 클라우드 백업 공급자를 등록 해제합니다.
RegisterOnlineBackupWithWindowsServerBackup Windows Server Backup에 클라우드 백업 공급자를 등록합니다.
UpdateOBStatusInWindowsServerBackup Windows Server Backup MMC 스냅인에서 클라우드 백업 공급자 상태 업데이트.

구조체

WSB_OB_REGISTRATION_INFO Windows Server Backup에 클라우드 백업 공급자를 등록하는 정보를 포함합니다.
WSB_OB_STATUS_ENTRY Windows Server Backup MMC 스냅인에 표시할 하나의 항목에 대한 상태 정보를 포함합니다.
WSB_OB_STATUS_ENTRY_VALUE_TYPE_PAIR 값 리소스 문자열을 확장하는 데 사용되는 매개 변수의 값 및 값 형식을 포함합니다.
WSB_OB_STATUS_INFO Windows Server Backup MMC 스냅인에서 클라우드 백업 공급자 상태 업데이트하는 정보를 포함합니다.

열거형

WSB_OB_STATUS_ENTRY_PAIR_TYPE 매개 변수 값의 형식을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wscapi.h 헤더

아티클2023. 08. 28.

이 헤더는 개발자 노트에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 정보

wscapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WscGetSecurityProviderHealth 지정된 WSC_SECURITY_PROVIDER 열거형 값으로 표시되는 보안 공급자 범주의 집계 상태를 가져옵니다.
WscRegisterForChanges WSC(Windows 보안 Center)가 보안 공급자 중 하나의 상태에 영향을 줄 수 있는 변경 사항을 검색할 때 실행할 콜백 함수를 등록합니다.
WscUnRegisterChanges WscRegisterForChanges 함수를 호출하여 만든 콜백 등록을 취소합니다.

열거형

WSC_SECURITY_PROVIDER WSC(Windows 보안 Center)에서 모니터링하는 모든 서비스를 정의합니다.
WSC_SECURITY_PROVIDER_HEALTH WSC(Windows 보안 Center)에서 모니터링하는 모든 서비스에 대해 가능한 상태를 정의합니다.

피드백

이 페이지가 도움이 되었나요?

Yes No

Microsoft Q&A에서 도움말 보기

wsdattachment.h 헤더

아티클2023. 08. 28.

이 헤더는 다른 디바이스 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 기타 디바이스 기술

wsdattachment.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWSDAttachment 다른 모든 첨부 파일 형식의 기본 인터페이스입니다.
IWSDInboundAttachment 애플리케이션이 들어오는 메시지에서 MIME로 인코딩된 첨부 파일 데이터를 읽을 수 있습니다.
IWSDOutboundAttachment 애플리케이션이 MIME 컨테이너를 사용하여 메시지의 첨부 파일 데이터를 보낼 수 있도록 합니다.

Functions

WSDCreateOutboundAttachment IWSDOutboundAttachment 개체를 만듭니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wsdbase.h 헤더

아티클2023. 08. 28.

이 헤더는 다른 디바이스 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 기타 디바이스 기술

wsdbase.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWSDAddress 전송 주소의 개별 구성 요소에 대한 액세스를 제공합니다.
IWSDHttpAddress HTTP 주소의 개별 구성 요소에 대한 액세스를 제공합니다.
IWSDHttpAuthParameters 이 인터페이스를 사용하여 클라이언트 인증 중에 사용되는 액세스 토큰 또는 권한 부여 체계를 검색합니다.
IWSDHttpMessageParameters SOAP over-HTTP를 통해 메시지를 전송할 때 사용되는 HTTP 헤더에 대한 액세스를 제공합니다.
IWSDMessageParameters 이 인터페이스를 사용하여 프로토콜 스택 위아래로 메시지 관련 정보를 전달합니다.
IWSDSignatureProperty 서명된 메시지의 속성을 제공합니다.
IWSDSSLClientCertificate 클라이언트 SSL 인증서를 검색합니다.
IWSDTransportAddress IP 기반 전송 주소를 나타냅니다.
IWSDUdpAddress

UDP 주소의 개별 구성 요소에 대한 액세스를 제공합니다.
IWSDUdpMessageParameters 이 인터페이스를 사용하여 WSD가 메시지 전송을 반복하는 빈도를 지정합니다.

Functions

WSDCreateHttpAddress IWSDHttpAddress 개체를 만듭니다.
WSDCreateHttpMessageParameters IWSDHttpMessageParameters 개체를 만듭니다.
WSDCreateUdpAddress IWSDUdpAddress 개체를 만듭니다.
WSDCreateUdpMessageParameters IWSDUdpMessageParameters 인터페이스에 대한 포인터를 검색합니다.

구조체

WSD_CONFIG_ADDRESSES 호스트가 수신 대기해야 하는 특정 주소에 대한 정보입니다.
WSD_CONFIG_PARAM WSDAPI 개체를 만들기 위한 구성 매개 변수를 나타냅니다.
WSD_SECURITY_CERT_VALIDATION HTTPS 서버와 클라이언트 인증서를 일치시키는 조건을 나타냅니다. (WSD_SECURITY_CERT_VALIDATION)
WSD_SECURITY_CERT_VALIDATION_V1

HTTPS 서버와 클라이언트 인증서를 일치시키는 조건을 나타냅니다. W
WSD_SECURITY_SIGNATURE_VALIDATION 메시지와 클라이언트 압축 서명을 일치시키는 조건을 나타냅니다.
WSDUdpRetransmitParams 메시지 전송을 반복하기 위한 매개 변수를 정의합니다.

열거형

WSD_CONFIG_PARAM_TYPE WSD_CONFIG_PARAM 구조에 저장된 데이터의 종류를 지정합니다.
WSDUdpMessageType UDP 메시지의 유형을 식별합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wsdclient.h 헤더

아티클2023. 08. 28.

이 헤더는 다른 디바이스 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 기타 디바이스 기술

wsdclient.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWSDAsyncCallback 비동기 작업 완료를 위한 콜백을 처리합니다.
IWSDAsyncResult 비동기 작업을 나타냅니다.
IWSDDeviceProxy 클라이언트 애플리케이션 및 미들웨어에 대한 DPWS(Web Services) 디바이스의 원격 디바이스 프로필을 나타냅니다.
IWSDEndpointProxy 디바이스 서비스 메시징 프록시를 구현합니다.
IWSDEventingStatus 이벤트 구독에서 상태 변경이 발생할 때 알림을 받도록 이 인터페이스를 구현합니다.
IWSDMetadataExchange 메타데이터에 액세스하는 다른 개체의 기본 클래스입니다.
IWSDServiceProxy 클라이언트 애플리케이션 및 미들웨어에 대한 원격 WSD 서비스를 나타냅니다.
IWSDServiceProxyEventing 클라이언트 애플리케이션 및 미들웨어에 대한 원격 WSD 서비스를 나타냅니다. 이 인터페이스를 사용하면 여러 비동기 작업을 구현할 수 있습니다.

Functions

WSDCreateDeviceProxy 디바이스 프록시를 만들고 IWSDDeviceProxy 인터페이스에 대한 포인터를 반환합니다. (WSDCreateDeviceProxy)
WSDCreateDeviceProxy2 서명된 메시지를 지원할 수 있는 디바이스 프록시를 만들고 IWSDDeviceProxy 인터페이스에 대한 포인터를 반환합니다.
WSDCreateDeviceProxyAdvanced 디바이스 프록시를 만들고 IWSDDeviceProxy 인터페이스에 대한 포인터를 반환합니다. (WSDCreateDeviceProxyAdvanced)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wsddisco.h 헤더

아티클2023. 08. 28.

이 헤더는 다른 디바이스 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 기타 디바이스 기술

wsddisco.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWSDiscoveredService 이 인터페이스는 원격으로 검색된 호스트를 나타냅니다.
IWSDiscoveryProvider WS-Discovery에서 보급한 네트워크에서 서비스를 검색하는 데 사용됩니다.
IWSDiscoveryProviderNotify IWSDiscoveryProvider에서 콜백 알림을 수신하기 위해 클라이언트 프로그램에 의해 구현됩니다.
IWSDiscoveryPublisher 호스트를 발표하고 호스트에 들어오는 쿼리를 관리하는 메서드를 제공합니다.
IWSDiscoveryPublisherNotify IWSDiscoveryPublisher에서 콜백 알림을 수신하기 위해 클라이언트 프로그램에 의해 구현됩니다.
IWSDScopeMatchingRule WS-Discovery에 정의된 표준 scope 일치 규칙을 확장하는 데 사용할 수 있는 사용자 지정 scope 일치 규칙을 제공하기 위해 클라이언트 프로그램에서 구현됩니다.

Functions

WSDCreateDiscoveryProvider IWSDiscoveryProvider 개체를 만듭니다.

WSDCreateDiscoveryProvider2 서명된 메시지를 지원하는 IWSDiscoveryProvider 개체를 만듭니다.
WSDCreateDiscoveryPublisher IWSDiscoveryPublisher 개체를 만듭니다.
WSDCreateDiscoveryPublisher2 서명된 메시지를 지원하는 IWSDiscoveryPublisher 개체를 만듭니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wsdevlicensing.h 헤더

아티클2023. 08. 28.

이 헤더는 개발자 라이선스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 개발자 라이선스

wsdevlicensing.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

AcquireDeveloperLicense 개발자 라이선스를 획득합니다.
CheckDeveloperLicense 개발자 라이선스가 있는지 확인합니다.
RemoveDeveloperLicense 개발자 라이선스를 제거합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wsdhost.h 헤더

아티클2023. 08. 28.

이 헤더는 다른 디바이스 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 기타 디바이스 기술

wsdhost.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWSDDeviceHost DPWS 규격 디바이스를 나타냅니다.
IWSDDeviceHostNotify IWSDDeviceHost 개체의 instance 디바이스 관련 알림을 제공합니다.
IWSDServiceMessaging 생성된 스텁 코드에서 들어오는 메시지에 오류 또는 응답을 보내는 데 사용됩니다.

Functions

WSDCreateDeviceHost 디바이스 호스트를 만들고 IWSDDeviceHost 인터페이스에 대한 포인터를 반환합니다. (WSDCreateDeviceHost)
WSDCreateDeviceHost2 서명된 메시지를 지원할 수 있는 디바이스 호스트를 만들고 IWSDDeviceHost 인터페이스에 대한 포인터를 반환합니다.
WSDCreateDeviceHostAdvanced 디바이스 호스트를 만들고 IWSDDeviceHost 인터페이스에 대한 포인터를 반환합니다. (WSDCreateDeviceHostAdvanced)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wsdtypes.h 헤더

아티클2023. 08. 28.

이 헤더는 다른 디바이스 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 기타 디바이스 기술

wsdtypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

콜백 함수

PWSD_SOAP_MESSAGE_HANDLER 들어오는 메시지에 대한 SOAP 메시지 처리기를 참조합니다.
WSD_STUB_FUNCTION 들어오는 메시지를 처리하는 데 사용되는 스텁 함수에 대해 설명합니다.

구조체

REQUESTBODY_GetStatus WS-Eventing GetStatus 요청 메시지를 나타냅니다.
REQUESTBODY_Renew WS-Eventing 갱신 요청 메시지를 나타냅니다.
REQUESTBODY_Subscribe WS-Eventing 구독 요청 메시지를 나타냅니다.
REQUESTBODY_Unsubscribe WS-Eventing 구독 취소 요청 메시지를 나타냅니다.
RESPONSEBODY_GetMetadata WS-MetadataExchange GetMetadata 응답 메시지를 나타냅니다.

RESPONSEBODY_GetStatus WS-Eventing GetStatus 응답 메시지를 나타냅니다.
RESPONSEBODY_Renew WS-Eventing 응답 갱신 메시지를 나타냅니다.
RESPONSEBODY_Subscribe WS-Eventing 구독 응답 메시지를 나타냅니다.
RESPONSEBODY_SubscriptionEnd WS-Eventing SubscriptionEnd 응답 메시지를 나타냅니다.
WSD_APP_SEQUENCE WS-Discovery 메시지와 관련된 애플리케이션 시퀀스 정보를 나타냅니다.
WSD_BYE Bye 메시지를 나타냅니다.
WSD_ENDPOINT_REFERENCE WS-Addressing 엔드포인트 참조를 나타냅니다.
WSD_ENDPOINT_REFERENCE_LIST WSD_ENDPOINT_REFERENCE 구조체의 단일 연결 목록에 있는 노드를 나타냅니다.
WSD_EVENT SOAP 메시지의 내부 표현을 제공합니다.
WSD_EVENTING_DELIVERY_MODE WS-Eventing 구독 메시지에 사용되는 배달 모드를 나타냅니다.
WSD_EVENTING_DELIVERY_MODE_PUSH WS-Eventing 구독 메시지에서 이벤트를 푸시 배달하는 데 사용되는 엔드포인트 참조를 나타냅니다.
WSD_EVENTING_EXPIRES WS-Eventing 메시지의 만료 시간을 나타냅니다.

WSD_EVENTING_FILTER WS-Eventing 구독 메시지에 사용되는 이벤트 필터를 나타냅니다.
WSD_EVENTING_FILTER_ACTION 이벤트 필터링에 사용되는 부울 식을 나타냅니다.
WSD_HANDLER_CONTEXT 들어오는 메시지를 처리하기 위한 컨텍스트를 지정합니다.
WSD_HEADER_RELATESTO WS-Addressing 사양에 지정된 RelatesTo SOAP 봉투 헤더 블록을 나타냅니다.
WSD_HELLO Hello 메시지를 나타냅니다.
WSD_HOST_METADATA 디바이스에서 호스트하는 모든 서비스에 대한 메타데이터를 제공합니다.
WSD_LOCALIZED_STRING 지역화된 단일 문자열을 나타냅니다.
WSD_LOCALIZED_STRING_LIST 지역화된 문자열의 단일 연결 목록에 있는 노드를 나타냅니다.
WSD_METADATA_SECTION 제네릭 형식의 메타데이터 섹션을 나타냅니다.
WSD_METADATA_SECTION_LIST 메타데이터 섹션의 단일 연결 목록에 있는 노드를 나타냅니다.
WSD_NAME_LIST 단일 연결된 XML 이름 구조 목록의 노드를 나타냅니다.
WSD_OPERATION WSDL에서 정의한 작업을 하나 또는 두 개의 메시지로 설명합니다.
WSD_PORT_TYPE

포트 형식에 대한 데이터를 제공합니다.
WSD_PROBE 프로브 메시지를 나타냅니다.
WSD_PROBE_MATCH ProbeMatch 메시지를 나타냅니다.
WSD_PROBE_MATCH_LIST ProbeMatch 메시지 구조의 단일 연결 목록에 있는 노드를 나타냅니다.
WSD_PROBE_MATCHES ProbeMatches 메시지를 나타냅니다.
WSD_REFERENCE_PARAMETERS 엔드포인트에서 사용되는 불투명 데이터를 지정합니다.
WSD_REFERENCE_PROPERTIES 엔드포인트를 고유하게 식별하는 데 사용되는 추가 데이터를 지정합니다.
WSD_RELATIONSHIP_METADATA 두 개 이상의 서비스 간의 관계에 대한 메타데이터를 제공합니다.
WSD_RESOLVE 확인 메시지를 나타냅니다.
WSD_RESOLVE_MATCH ResolveMatch 메시지를 나타냅니다.
WSD_RESOLVE_MATCHES ResolveMatches 메시지를 나타냅니다.
WSD_SCOPES WS-Discovery 메시징에 사용되는 범위의 컬렉션입니다.
WSD_SERVICE_METADATA 디바이스에서 호스트하는 서비스에 대한 메타데이터를 제공합니다.

WSD_SERVICE_METADATA_LIST 서비스 메타데이터 구조의 단일 연결 목록에 있는 노드를 나타냅니다.
WSD_SOAP_FAULT 생성된 SOAP 오류를 나타냅니다.
WSD_SOAP_FAULT_CODE 생성된 SOAP 오류 코드를 나타냅니다.
WSD_SOAP_FAULT_REASON WSD_SOAP_FAULT 연결된 이유 코드의 컬렉션입니다.
WSD_SOAP_FAULT_SUBCODE 생성된 SOAP 오류 하위 코드를 나타냅니다.
WSD_SOAP_HEADER WSD_SOAP_MESSAGE 구조체에 대한 SOAP 헤더 데이터를 제공합니다.
WSD_SOAP_MESSAGE WSD SOAP 메시지의 내용입니다.
WSD_SYNCHRONOUS_RESPONSE_CONTEXT 양방향 요청에 대한 응답을 처리하기 위한 컨텍스트를 제공합니다.
WSD_THIS_DEVICE_METADATA 특정 디바이스에 고유한 메타데이터를 지정합니다.
WSD_THIS_MODEL_METADATA 디바이스와 관련된 모델별 정보를 제공합니다.
WSD_UNKNOWN_LOOKUP 구문 분석할 수 없는 XML 요소를 나타냅니다.
WSD_URI_LIST 연결된 URI 목록의 노드를 나타냅니다.

열거형

WSD_PROTOCOL_TYPE 포트에서 지원하는 프로토콜 유형을 식별합니다.
WSDEventType 세션 계층에서 생성되는 이벤트의 유형을 식별합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wsdutil.h 헤더

아티클2023. 08. 28.

이 헤더는 다른 디바이스 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 기타 디바이스 기술

wsdutil.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WSDAllocateLinkedMemory 연결된 메모리 블록을 할당합니다.
WSDAttachLinkedMemory 자식 메모리 블록을 부모 메모리 블록에 연결합니다.
WSDDetachLinkedMemory 부모 메모리 블록에서 자식 메모리 블록을 분리합니다.
WSDFreeLinkedMemory WSDAllocateLinkedMemory를 사용하여 이전에 할당된 메모리 블록을 해제합니다.
WSDGenerateFault SOAP 오류를 생성합니다. (WSDGenerateFault)
WSDGenerateFaultEx SOAP 오류를 생성합니다. (WSDGenerateFaultEx)
WSDGetConfigurationOption WSDAPI 구성 옵션을 가져옵니다.
WSDSetConfigurationOption WSDAPI 구성 옵션을 설정합니다.
WSDUriDecode

RFC2396 규칙에 따라 URI를 디코딩합니다.
WSDUriEncode RFC2396 URI 인코딩 규칙에 따라 URI를 인코딩합니다.
WSDXMLAddChild 자식 요소를 추가합니다.
WSDXMLAddSibling 형제 요소를 추가합니다.
WSDXMLBuildAnyForSingleElement 지정된 이름과 값을 사용하여 XML 요소를 만듭니다.
WSDXMLCleanupElement XML 요소와 연결된 메모리를 해제합니다.
WSDXMLGetValueFromAny XML 모든 요소의 지정된 자식 요소에서 텍스트 값을 검색합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wsdxml.h 헤더

아티클2023. 08. 28.

이 헤더는 다른 디바이스 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 기타 디바이스 기술

wsdxml.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWSDXMLContext WSDAPI 스택에 사용되는 네임스페이스 및 형식의 컬렉션입니다.

Functions

WSDXMLCreateContext 새 IWSDXMLContext 개체를 만듭니다.
WSDXMLGetNameFromBuiltinNamespace 기본 제공 네임스페이스에서 지정된 이름을 가져옵니다.

구조체

WSD_DATETIME 타임스탬프를 나타냅니다.
WSD_DURATION 시간 길이를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wsdxmldom.h 헤더

아티클2023. 08. 28.

이 헤더는 다른 디바이스 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 기타 디바이스 기술

wsdxmldom.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

WSDXML_ATTRIBUTE XML 특성에 대해 설명합니다.
WSDXML_ELEMENT XML 요소를 설명합니다.
WSDXML_ELEMENT_LIST XML 요소의 연결된 목록에 있는 노드를 나타냅니다.
WSDXML_NAME XML 정규화된 이름을 지정합니다.
WSDXML_NAMESPACE XML 네임스페이스를 지정합니다.
WSDXML_NODE XML 노드에 대해 설명합니다.
WSDXML_PREFIX_MAPPING XML 네임스페이스 접두사를 설명합니다.
WSDXML_TEXT XML 노드의 텍스트를 설명합니다.
WSDXML_TYPE

XSD 형식에 대해 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wsipv6ok.h 헤더

아티클2023. 08. 28.

이 헤더는 Windows 소켓 2에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 소켓 2

wsipv6ok.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

gethostbyaddr gethostbyaddr 매크로 함수(wsipv6ok.h)는 네트워크 주소에 해당하는 호스트 정보를 검색합니다.
gethostbyname gethostbyname 매크로 함수(wsipv6ok.h)는 호스트 데이터베이스에서 호스트 이름에 해당하는 호스트 정보를 검색합니다.
inet_addr inet_addr 매크로 함수(wsipv6ok.h)는 IPv4 점선 소수점 주소가 포함된 문자열을 IN_ADDR 구조체에 적합한 주소로 변환합니다.
inet_ntoa inet_ntoa 매크로 함수(wsipv6ok.h)는 (Ipv4) 인터넷 네트워크 주소를 인터넷 표준 점선 10진수 형식의 ASCII 문자열로 변환합니다.
WSAAsyncGetHostByAddr WSAAsyncGetHostByAddr 매크로 함수(wsipv6ok.h)는 주소에 해당하는 호스트 정보를 비동기적으로 검색합니다.
WSAAsyncGetHostByName WSAAsyncGetHostByName 매크로 함수(wsipv6ok.h)는 호스트 이름에 해당하는 호스트 정보를 비동기적으로 검색합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wslapi.h 헤더

아티클2023. 08. 28.

이 헤더는 Linux용 Windows 하위 시스템 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Linux용 Windows 하위 시스템

wslapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WslConfigureDistribution WSL(Linux용 Windows 하위 시스템)에 등록된 배포의 동작을 수정합니다.
WslGetDistributionConfiguration WSL(Linux용 Windows 하위 시스템)에 등록된 배포의 현재 구성을 검색합니다.
WslIsDistributionRegistered 배포가 WSL(Linux용 Windows 하위 시스템)에 등록되어 있는지 확인합니다.
WslLaunch 특정 배포의 컨텍스트에서 WSL(Linux용 Windows 하위 시스템) 프로세스를 시작합니다.
WslLaunchInteractive 특정 배포의 컨텍스트에서 WSL(대화형 Linux용 Windows 하위 시스템) 프로세스를 시작합니다.
WslRegisterDistribution WSL(Linux용 Windows 하위 시스템)에 새 배포를 등록합니다.
WslUnregisterDistribution Linux용 Windows 하위 시스템(WSL)에서 배포를 등록 취소합니다.

열거형

WSL_DISTRIBUTION_FLAGS WSL_DISTRIBUTION_FLAGS 열거형은 Linux용 Windows 하위 시스템(WSL)의 분포 동작을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wsman.h 헤더

아티클2023. 08. 28.

이 헤더는 Windows 원격 관리에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 원격 관리

wsman.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WSManCloseCommand 명령을 삭제하고 연결된 리소스를 해제합니다.
WSManCloseOperation 비동기 작업을 취소하거나 닫습니다.
WSManCloseSession 세션 개체를 닫습니다.
WSManCloseShell 셸 개체를 삭제하고 셸과 연결된 리소스를 해제합니다.
WSManConnectShell 기존 서버 세션에 연결합니다.
WSManConnectShellCommand 셸에서 실행되는 기존 명령에 연결합니다.
WSManCreateSession 세션 개체를 만듭니다.
WSManCreateShell 셸 개체를 만듭니다.
WSManCreateShellEx

클라이언트 지정 셸 ID를 추가하여 WSManCreateShell 함수와 동일한 기능을 사용하여 셸 개체를 만듭니다.
WSManDeinitialize Windows 원격 관리 클라이언트 스택을 초기화합니다.
WSManDisconnectShell 활성 셸 및 관련 명령의 네트워크 연결을 끊습니다.
WSManGetErrorMessage 특정 오류 및 언어 코드와 관련된 오류 메시지를 검색합니다.
WSManGetSessionOptionAsDword 세션 옵션의 값을 가져옵니다. (WSManGetSessionOptionAsDword)
WSManGetSessionOptionAsString 세션 옵션의 값을 가져옵니다. (WSManGetSessionOptionAsString)
WSManInitialize Windows 원격 관리 클라이언트 API를 초기화합니다.
WSManPluginAuthzOperationComplete 사용자 작업에 대한 성공적인 권한 부여 또는 실패한 권한 부여를 보고합니다.
WSManPluginAuthzQueryQuotaComplete WSManPluginAuthzQueryQuota 플러그 인 진입점에서 호출되며 플러그 인이 요청을 수행할 수 있는지 여부를 호출해야 합니다.
WSManPluginAuthzUserComplete 성공적인 사용자 연결 권한 부여 또는 실패한 사용자 연결 권한 부여를 보고합니다.
WSManPluginFreeRequestDetails 작업 플러그 인 진입점에 전달되는 WSMAN_PLUGIN_REQUEST 구조에 할당된 메모리를 해제합니다.
WSManPluginGetOperationParameters 작업과 연결된 제한 시간 및 데이터 제한과 같은 항목에 대한 운영 정보를 가져옵니다.

WSManPluginOperationComplete WSManPluginStartup 및 WSManPluginShutdown 메서드를 제외한 모든 작업 진입점에 의한 작업 완료를 보고합니다.
WSManPluginReceiveResult WSMAN_PLUGIN_RECEIVE 플러그 인 호출에 대한 결과를 보고하고 결과를 반환하는 대부분의 셸 플러그 인에서 사용됩니다.
WSManPluginReportContext 셸 및/또는 명령에 대해 추가 작업을 수행할 수 있도록 셸 및 명령 컨텍스트를 WinRM(Windows 원격 관리) 인프라에 다시 보고합니다.
WSManReceiveShellOutput 실행 중인 명령 또는 셸에서 출력을 검색합니다.
WSManReconnectShell 이전에 연결이 끊긴 셸 세션을 다시 연결합니다. 셸 세션의 연결된 명령을 다시 연결하려면 WSManReconnectShellCommand를 사용합니다.
WSManReconnectShellCommand 이전에 연결이 끊긴 명령을 다시 연결합니다.
WSManRunShellCommand 기존 셸 내에서 명령 실행을 시작하고 명령이 완료될 때까지 기다리지 않습니다.
WSManRunShellCommandEx 명령 ID 옵션을 추가하여 WSManRunShellCommand 함수와 동일한 기능을 제공합니다.
WSManSendShellInput 입력 스트림을 실행 중인 명령 또는 셸에 Ipes합니다.
WSManSetSessionOption 세션에 대한 확장 옵션 집합을 설정합니다.
WSManSignalShell 기존 명령 또는 셸 자체에 컨트롤 코드를 보냅니다.

콜백 함수

WSMAN_PLUGIN_AUTHORIZE_OPERATION 특정 작업에 권한을 부여합니다.
WSMAN_PLUGIN_AUTHORIZE_QUERY_QUOTA 연결 권한이 부여된 후 사용자의 할당량 정보를 검색합니다.
WSMAN_PLUGIN_AUTHORIZE_RELEASE_CONTEXT 플러그 인이 WSManPluginAuthzUserComplete 또는 WSManPluginAuthzOperationComplete에서 보고하는 컨텍스트를 해제합니다.
WSMAN_PLUGIN_AUTHORIZE_USER 연결에 권한을 부여합니다.
WSMAN_PLUGIN_COMMAND 플러그 인에 대한 명령 콜백을 정의합니다.
WSMAN_PLUGIN_CONNECT 플러그 인에 대한 연결 콜백을 정의합니다.
WSMAN_PLUGIN_RECEIVE 플러그 인에 대한 수신 콜백을 정의합니다.
WSMAN_PLUGIN_RELEASE_COMMAND_CONTEXT 플러그 인에 대한 릴리스 명령 콜백을 정의합니다.
WSMAN_PLUGIN_RELEASE_SHELL_CONTEXT 플러그 인에 대한 릴리스 셸 콜백을 정의합니다.
WSMAN_PLUGIN_SEND 플러그 인에 대한 송신 콜백을 정의합니다.
WSMAN_PLUGIN_SHELL 플러그 인에 대한 셸 콜백을 정의합니다.
WSMAN_PLUGIN_SHUTDOWN

플러그 인에 대한 종료 콜백을 정의합니다.
WSMAN_PLUGIN_SIGNAL 플러그 인에 대한 신호 콜백을 정의합니다.
WSMAN_PLUGIN_STARTUP 플러그 인에 대한 시작 콜백을 정의합니다.
WSMAN_SHELL_COMPLETION_FUNCTION 셸 작업에 대해 호출되는 콜백 함수로, 원격 요청이 발생합니다.

구조체

WSMAN_AUTHENTICATION_CREDENTIALS 인증 방법 및 서버 또는 프록시 인증에 사용되는 자격 증명을 정의합니다.
WSMAN_AUTHZ_QUOTA 권한 부여 플러그 인에 대한 사용자 단위로 할당량 정보를 보고합니다.
WSMAN_CERTIFICATE_DETAILS 클라이언트 인증서와 함께 전송된 인바운드 요청에 대한 클라이언트 정보를 저장합니다.
WSMAN_COMMAND_ARG_SET 명령줄에 전달되는 인수 집합을 나타냅니다.
WSMAN_DATA WinRM(Windows 원격 관리) API에 사용되는 인바운드 및 아웃바운드 데이터를 포함합니다.
WSMAN_DATA_BINARY 다양한 Windows 원격 관리 함수에 사용할 이진 데이터를 보유합니다.
WSMAN_DATA_TEXT 다양한 Windows 원격 관리 함수와 함께 사용할 텍스트 데이터를 보유합니다.
WSMAN_ENVIRONMENT_VARIABLE

이름 및 값 쌍을 사용하여 개별 환경 변수를 정의합니다.
WSMAN_ENVIRONMENT_VARIABLE_SET 환경 변수의 배열을 정의합니다.
WSMAN_ERROR WinRM(Windows 원격 관리) 클라이언트에서 반환하는 오류 정보를 포함합니다.
WSMAN_FILTER WSMAN_FILTER 나중에 사용하도록 예약되어 있습니다.
WSMAN_FRAGMENT WSMAN_FRAGMENT 나중에 사용할 수 있습니다.
WSMAN_KEY 선택기 집합 내의 키 및 값 쌍을 나타내며 특정 리소스를 식별하는 데 사용됩니다.
WSMAN_OPERATION_INFO 플러그 인이 요청을 수행해야 하는 특정 리소스 엔드포인트를 나타냅니다.
WSMAN_OPTION 특정 옵션 이름 및 값 쌍을 나타냅니다.
WSMAN_OPTION_SET 옵션 집합을 나타냅니다.
WSMAN_PLUGIN_REQUEST 플러그 인 요청에 대한 정보를 지정합니다.
WSMAN_PROXY_INFO 프록시 정보를 지정합니다.
WSMAN_RECEIVE_DATA_RESULT WSManReceiveShellOutput 메서드에서 받은 출력 데이터를 나타냅니다.
WSMAN_RESPONSE_DATA WSMan 작업에서 받은 출력 데이터를 나타냅니다.

WSMAN_SELECTOR_SET 리소스의 ID를 나타내는 키 집합을 정의합니다.
WSMAN_SENDER_DETAILS 모든 인바운드 요청에 대한 클라이언트 세부 정보를 지정합니다.
WSMAN_SHELL_ASYNC 모든 셸 작업에 전달할 비동기 구조를 정의합니다.
WSMAN_SHELL_DISCONNECT_INFO 클라이언트의 연결이 끊긴 후에도 셸이 열린 상태로 유지되는 최대 기간(밀리초)을 지정합니다.
WSMAN_SHELL_STARTUP_INFO_V10 WSMAN_SHELL_STARTUP_INFO_V10(wsman.h) 구조체는 WSManCreateShell 함수와 함께 사용할 셸 시작 매개 변수를 정의합니다.
WSMAN_SHELL_STARTUP_INFO_V11 WSMAN_SHELL_STARTUP_INFO_V11(wsman.h) 구조체는 WSManCreateShell 함수와 함께 사용할 셸 시작 매개 변수를 정의합니다.
WSMAN_STREAM_ID_SET 셸 및 명령에 대한 입력 또는 출력에 사용되는 모든 스트림을 나열합니다.
WSMAN_USERNAME_PASSWORD_CREDS 인증에 사용되는 자격 증명을 정의합니다.

열거형

WSManAuthenticationFlags 작업에 대한 인증 방법을 결정합니다.
WSManCallbackFlags 모든 콜백 함수에서 사용하는 플래그 집합을 정의합니다.

WSManDataType WSMAN_DATA 구조체에서 공용 구조체의 현재 데이터 형식을 지정합니다.
WSManProxyAccessType 프록시 액세스 유형을 정의합니다.
WSManSessionOption 세션에 대한 확장 옵션 집합을 정의합니다. 이러한 옵션은 WSManSetSessionOption 메서드와 함께 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wsmandisp.h 헤더

아티클2023. 08. 28.

이 헤더는 Windows 원격 관리에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 원격 관리

wsmandisp.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWSMan Session 개체로 표시되는 세션을 만드는 데 사용되는 메서드 및 속성을 제공합니다.
IWSManConnectionOptions IWSManConnectionOptions 개체는 원격 컴퓨터의 로컬 계정과 연결된 사용자 이름과 암호를 제공하기 위해 IWSMan::CreateSession 메서드에 전달됩니다.
IWSManConnectionOptionsEx IWSManConnectionOptionsEx 개체는 인증에 사용되는 클라이언트 인증서의 지문을 제공하기 위해 IWSMan::CreateSession 메서드에 전달됩니다.
IWSManConnectionOptionsEx2 IWSManConnectionOptionsEx2 개체는 IWSMan::CreateSession 메서드에 전달되어 프록시 서버에 연결할 인증 메커니즘, 액세스 유형 및 자격 증명을 제공합니다.
IWSManEnumerator WS-Management 프로토콜 WS-Enumeration:Enumerate 작업과 같은 작업에서 반환된 결과 스트림을 나타냅니다.
IWSManEx IWSManResourceLocator 개체 만들기, 열거형 및 세션 플래그 값을 반환하는 메서드 및 확장된 오류 정보를 가져오는 메서드를 포함하도록 IWSMan 인터페이스의 메서드와 속성을 확장합니다.
IWSManEx2 클라이언트 인증서를 사용하여 인증과 관련된 세션 플래그 값을 반환하는 메서드를 포함하도록 IWSManEx 인터페이스의 메서드 및 속성을 확장합니다.

IWSManEx3 CredSSP(자격 증명 보안 지원 공급자)를 사용하여 인증과 관련된 세션 플래그 값을 반환하는 메서드를 포함하도록 IWSManEx 인터페이스의 메서드 및 속성을 확장합니다.
IWSManResourceLocator 리소스에 대한 경로를 제공합니다. IWSManSession.Get, IWSManSession.Put 또는 IWSManSession.Enumerate와 같은 IWSManSession 개체 작업의 리소스 URI 대신 IWSManResourceLocator 개체를 사용할 수 있습니다.
IWSManSession 작업 및 세션 설정을 정의합니다.

열거형

WSManProxyAccessTypeFlags 프록시 액세스 유형 플래그를 정의합니다.
WSManProxyAuthenticationFlags 프록시 인증 메커니즘을 결정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wsnwlink.h 헤더

아티클2023. 08. 28.

이 헤더는 Windows 소켓 2에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 소켓 2

wsnwlink.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

IPX_ADDRESS_DATA IPX_ADDRESS_DATA 구조는 IPX가 바인딩된 특정 어댑터에 대한 정보를 제공합니다. optname 매개 변수에서 IPX_ADDRESS 지정하는 getsockopt 함수 호출과 함께 사용됩니다.
IPX_NETNUM_DATA IPX_NETNUM_DATA 구조는 지정된 IPX 네트워크 번호에 대한 정보를 제공합니다. optname 매개 변수에서 IPX_GETNETINFO 지정하는 getsockopt 함수 호출과 함께 사용됩니다.
IPX_SPXCONNSTATUS_DATA IPX_SPXCONNSTATUS_DATA 구조체는 연결된 SPX 소켓에 대한 정보를 제공합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wsrm.h 헤더

아티클2023. 08. 28.

이 헤더는 Windows 소켓 2에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 소켓 2

wsrm.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

RM_FEC_INFO RM_FEC_INFO 구조는 신뢰할 수 있는 멀티캐스트에서 FEC(정방향 오류 수정)를 사용하기 위한 설정을 지정합니다. 이 구조체는 RM_USE_FEC 소켓 옵션과 함께 사용됩니다.
RM_RECEIVER_STATS 신뢰할 수 있는 멀티캐스트 수신기 세션에 대한 통계 정보를 제공합니다. 이 구조체는 RM_RECEIVER_STATISTICS 소켓 옵션과 함께 사용됩니다.
RM_SEND_WINDOW RM_SEND_WINDOW 구조는 신뢰할 수 있는 멀티캐스트 보내기 창을 지정합니다. 이 구조체는 RM_RATE_WINDOW_SIZE 소켓 옵션과 함께 사용됩니다.
RM_SENDER_STATS 신뢰할 수 있는 멀티캐스트 보낸 사람 세션에 대한 통계 정보를 제공합니다. 이 구조체는 RM_SENDER_STATISTICS 소켓 옵션과 함께 사용됩니다.

열거형

eWINDOW_ADVANCE_METHOD eWINDOW_ADVANCE_METHOD 열거형은 신뢰할 수 있는 멀티캐스트에 사용되는 창 사전 모드를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wtsapi32.h 헤더

아티클2023. 08. 28.

이 헤더는 원격 데스크톱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 데스크톱 서비스

wtsapi32.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WTSCloseServer 열린 핸들을 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에 닫습니다.
WTSConnectSessionA 원격 데스크톱 서비스 세션을 로컬 컴퓨터의 기존 세션에 연결합니다. (ANSI)
WTSConnectSessionW 원격 데스크톱 서비스 세션을 로컬 컴퓨터의 기존 세션에 연결합니다. (유니코드)
WTSCreateListenerA 새 원격 데스크톱 서비스 수신기를 만들거나 기존 수신기를 구성합니다. (ANSI)
WTSCreateListenerW 새 원격 데스크톱 서비스 수신기를 만들거나 기존 수신기를 구성합니다. (유니코드)
WTSDisconnectSession 세션을 닫지 않고 지정된 원격 데스크톱 서비스 세션에서 로그온한 사용자의 연결을 끊습니다.
WTSEnableChildSessions 자식 세션을 사용하거나 사용하지 않도록 설정합니다.
WTSEnumerateListenersA RD 세션 호스트(원격 데스크톱 세션 호스트) 서버의 모든 원격 데스크톱 서비스 수신기를 열거합니다. (ANSI)
WTSEnumerateListenersW

RD 세션 호스트(원격 데스크톱 세션 호스트) 서버의 모든 원격 데스크톱 서비스 수신기를 열거합니다. (유니코드)
WTSEnumerateProcessesA 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 활성 프로세스에 대한 정보를 검색합니다. (ANSI)
WTSEnumerateProcessesExA 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버 또는 RD 가상화 호스트(원격 데스크톱 가상화 호스트) 서버에서 활성 프로세스에 대한 정보를 검색합니다. (ANSI)
WTSEnumerateProcessesExW 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버 또는 RD 가상화 호스트(원격 데스크톱 가상화 호스트) 서버에서 활성 프로세스에 대한 정보를 검색합니다. (유니코드)
WTSEnumerateProcessesW 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 활성 프로세스에 대한 정보를 검색합니다. (유니코드)
WTSEnumerateServersA 지정된 도메인 내의 모든 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버 목록을 반환합니다. (ANSI)
WTSEnumerateServersW 지정된 도메인 내의 모든 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버 목록을 반환합니다. (유니코드)
WTSEnumerateSessionsA RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 세션 목록을 검색합니다. (ANSI)
WTSEnumerateSessionsExA 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버 또는 RD 가상화 호스트(원격 데스크톱 가상화 호스트) 서버에서 세션 목록을 검색합니다. (ANSI)
WTSEnumerateSessionsExW 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버 또는 RD 가상화 호스트(원격 데스크톱 가상화 호스트) 서버에서 세션 목록을 검색합니다. (유니코드)
WTSEnumerateSessionsW

RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 세션 목록을 검색합니다. (유니코드)
WTSFreeMemory 원격 데스크톱 서비스 함수에서 할당한 메모리를 해제합니다.
WTSFreeMemoryExA 원격 데스크톱 서비스 함수에서 할당한 WTS_PROCESS_INFO_EX 또는 WTS_SESSION_INFO_1 구조를 포함하는 메모리를 해제합니다. (ANSI)
WTSFreeMemoryExW 원격 데스크톱 서비스 함수에서 할당한 WTS_PROCESS_INFO_EX 또는 WTS_SESSION_INFO_1 구조를 포함하는 메모리를 해제합니다. (유니코드)
WTSGetChildSessionId 자식 세션 식별자(있는 경우)를 검색합니다.
WTSGetListenerSecurityA 원격 데스크톱 서비스 수신기의 보안 설명자를 검색합니다. (ANSI)
WTSGetListenerSecurityW 원격 데스크톱 서비스 수신기의 보안 설명자를 검색합니다. (유니코드)
WTSIsChildSessionsEnabled 자식 세션을 사용할 수 있는지 여부를 결정합니다.
WTSLogoffSession 지정된 원격 데스크톱 서비스 세션을 로그오프합니다.
WTSOpenServerA 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에 대한 핸들을 엽니다. (ANSI)
WTSOpenServerExA 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버 또는 RD 가상화 호스트(원격 데스크톱 가상화 호스트) 서버에 대한 핸들을 엽니다. (ANSI)
WTSOpenServerExW 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버 또는 RD 가상화 호스트(원격 데스크톱 가상화 호스트) 서버에 대한 핸들을 엽니다. (유니코드)

WTSOpenServerW 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에 대한 핸들을 엽니다. (유니코드)
WTSQueryListenerConfigA 원격 데스크톱 서비스 수신기에 대한 구성 정보를 검색합니다. (ANSI)
WTSQueryListenerConfigW 원격 데스크톱 서비스 수신기에 대한 구성 정보를 검색합니다. (유니코드)
WTSQuerySessionInformationA 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 지정된 세션에 대한 세션 정보를 검색합니다. (ANSI)
WTSQuerySessionInformationW 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 지정된 세션에 대한 세션 정보를 검색합니다. (유니코드)
WTSQueryUserConfigA 지정된 도메인 컨트롤러 또는 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 지정된 사용자의 구성 정보를 검색합니다. (ANSI)
WTSQueryUserConfigW 지정된 도메인 컨트롤러 또는 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 지정된 사용자의 구성 정보를 검색합니다. (유니코드)
WTSQueryUserToken 세션 ID로 지정된 로그온한 사용자의 기본 액세스 토큰을 가져옵니다.
WTSRegisterSessionNotification 세션 변경 알림을 받을 지정된 창을 등록합니다. (WTSRegisterSessionNotification)
WTSRegisterSessionNotificationEx 세션 변경 알림을 받을 지정된 창을 등록합니다. (WTSRegisterSessionNotificationEx)
WTSSendMessageA 지정된 원격 데스크톱 서비스 세션의 클라이언트 데스크톱에 메시지 상자를 표시합니다. (ANSI)

WTSSendMessageW 지정된 원격 데스크톱 서비스 세션의 클라이언트 데스크톱에 메시지 상자를 표시합니다. (유니코드)
WTSSetListenerSecurityA 원격 데스크톱 서비스 수신기의 보안 설명자를 구성합니다. (ANSI)
WTSSetListenerSecurityW 원격 데스크톱 서비스 수신기의 보안 설명자를 구성합니다. (유니코드)
WTSSetUserConfigA 지정된 도메인 컨트롤러 또는 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 지정된 사용자에 대한 구성 정보를 수정합니다. (ANSI)
WTSSetUserConfigW 지정된 도메인 컨트롤러 또는 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 지정된 사용자에 대한 구성 정보를 수정합니다. (유니코드)
WTSShutdownSystem 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버를 종료하고 필요에 따라 다시 시작합니다.
WTSStartRemoteControlSessionA 다른 원격 데스크톱 서비스 세션의 원격 제어를 시작합니다. 원격 세션에서 이 함수를 호출해야 합니다. (ANSI)
WTSStartRemoteControlSessionW 다른 원격 데스크톱 서비스 세션의 원격 제어를 시작합니다. 원격 세션에서 이 함수를 호출해야 합니다. (유니코드)
WTSStopRemoteControlSession 원격 제어 세션을 중지합니다.
WTSTerminateProcess 지정된 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 지정된 프로세스를 종료합니다.
WTSUnRegisterSessionNotification

더 이상 세션 변경 알림을 받지 않도록 지정된 창의 등록을 취소합니다. (WTSUnRegisterSessionNotification)
WTSUnRegisterSessionNotificationEx 더 이상 세션 변경 알림을 받지 않도록 지정된 창의 등록을 취소합니다. (WTSUnRegisterSessionNotificationEx)
WTSVirtualChannelClose 열린 가상 채널 핸들을 닫습니다.
WTSVirtualChannelOpen 지정된 가상 채널의 서버 끝에 대한 핸들을 엽니다.
WTSVirtualChannelOpenEx WTSVirtualChannelOpen과 유사한 방식으로 가상 채널을 만듭니다.
WTSVirtualChannelPurgeInput 클라이언트에서 지정된 가상 채널의 서버로 전송된 큐에 대기 중인 모든 입력 데이터를 삭제합니다.
WTSVirtualChannelPurgeOutput 서버에서 지정된 가상 채널의 클라이언트로 전송된 큐에 대기 중인 모든 출력 데이터를 삭제합니다.
WTSVirtualChannelQuery 지정된 가상 채널에 대한 정보를 반환합니다.
WTSVirtualChannelRead 가상 채널의 서버 끝에서 데이터를 읽습니다.
WTSVirtualChannelWrite 가상 채널의 서버 끝에 데이터를 씁니다.
WTSWaitSystemEvent 호출자에게 반환하기 전에 원격 데스크톱 서비스 이벤트를 기다립니다.

구조체

WTS_CLIENT_ADDRESS 원격 데스크톱 서비스 세션의 클라이언트 네트워크 주소를 포함합니다.
WTS_CLIENT_DISPLAY RDC(원격 데스크톱 연결) 클라이언트 표시에 대한 정보를 포함합니다. (WTS_CLIENT_DISPLAY)
WTS_PROCESS_INFO_EXA RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 실행되는 프로세스에 대한 확장된 정보를 포함합니다. (ANSI)
WTS_PROCESS_INFO_EXW RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 실행되는 프로세스에 대한 확장된 정보를 포함합니다. (유니코드)
WTS_PROCESS_INFOA RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 실행되는 프로세스에 대한 정보를 포함합니다. (ANSI)
WTS_PROCESS_INFOW RD 세션 호스트(원격 데스크톱 세션 호스트) 서버에서 실행되는 프로세스에 대한 정보를 포함합니다. (유니코드)
WTS_SERVER_INFOA 특정 원격 데스크톱 서비스 서버에 대한 정보를 포함합니다. (ANSI)
WTS_SERVER_INFOW 특정 원격 데스크톱 서비스 서버에 대한 정보를 포함합니다. (유니코드)
WTS_SESSION_ADDRESS 세션에 할당된 가상 IP 주소를 포함합니다.
WTS_SESSION_INFO_1A RD 세션 호스트(원격 데스크톱 세션 호스트) 서버 또는 RD 가상화 호스트(원격 데스크톱 가상화 호스트) 서버의 클라이언트 세션에 대한 확장된 정보를 포함합니다. (ANSI)
WTS_SESSION_INFO_1W RD 세션 호스트(원격 데스크톱 세션 호스트) 서버 또는 RD 가상화 호스트(원격 데스크톱 가상화 호스트) 서버의 클라이언트 세션에 대한 확장된 정보를 포함합니다. (유니코드)

WTS_SESSION_INFOA RD 세션 호스트(원격 데스크톱 세션 호스트) 서버의 클라이언트 세션에 대한 정보를 포함합니다. (ANSI)
WTS_SESSION_INFOW RD 세션 호스트(원격 데스크톱 세션 호스트) 서버의 클라이언트 세션에 대한 정보를 포함합니다. (유니코드)
WTSCLIENTA RDC(원격 데스크톱 연결) 클라이언트에 대한 정보를 포함합니다. (ANSI)
WTSCLIENTW RDC(원격 데스크톱 연결) 클라이언트에 대한 정보를 포함합니다. (유니코드)
WTSCONFIGINFOA 원격 데스크톱 서비스 세션에 대한 정보를 포함합니다. (WTSCONFIGINFOA)
WTSCONFIGINFOW 원격 데스크톱 서비스 세션에 대한 정보를 포함합니다. (WTSCONFIGINFOW)
WTSINFOA 원격 데스크톱 서비스 세션에 대한 정보를 포함합니다. (WTSINFOA)
WTSINFOEX_LEVEL_A 원격 데스크톱 서비스 세션에 대한 확장된 정보를 포함하는 WTSINFOEX_LEVEL1 구조를 포함합니다. (ANSI)
WTSINFOEX_LEVEL_W 원격 데스크톱 서비스 세션에 대한 확장된 정보를 포함하는 WTSINFOEX_LEVEL1 구조를 포함합니다. (유니코드)
WTSINFOEX_LEVEL1_A 원격 데스크톱 서비스 세션에 대한 확장된 정보를 포함합니다. (ANSI)
WTSINFOEX_LEVEL1_W 원격 데스크톱 서비스 세션에 대한 확장된 정보를 포함합니다. (유니코드)

WTSINFOEXA 원격 데스크톱 서비스 세션에 대한 확장된 정보를 포함하는 WTSINFOEX_LEVEL 공용 구조체를 포함합니다. (ANSI)
WTSINFOEXW 원격 데스크톱 서비스 세션에 대한 확장된 정보를 포함하는 WTSINFOEX_LEVEL 공용 구조체를 포함합니다. (유니코드)
WTSINFOW 원격 데스크톱 서비스 세션에 대한 정보를 포함합니다. (WTSINFOW)
WTSLISTENERCONFIGA 원격 데스크톱 서비스 수신기에 대한 정보를 포함합니다. (ANSI)
WTSLISTENERCONFIGW 원격 데스크톱 서비스 수신기에 대한 정보를 포함합니다. (유니코드)
WTSUSERCONFIGA 도메인 컨트롤러 또는 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버의 사용자에 대한 구성 정보를 포함합니다. (ANSI)
WTSUSERCONFIGW 도메인 컨트롤러 또는 RD 세션 호스트(원격 데스크톱 세션 호스트) 서버의 사용자에 대한 구성 정보를 포함합니다. (유니코드)

열거형

WTS_CONFIG_CLASS WTSQueryUserConfig 및 WTSSetUserConfig 함수에 대한 호출에서 설정하거나 검색할 사용자 구성 정보의 유형을 나타내는 값을 포함합니다.
WTS_CONFIG_SOURCE WTSQueryUserConfig 함수에서 반환된 구성 정보의 원본을 지정합니다.
WTS_CONNECTSTATE_CLASS

원격 데스크톱 서비스 세션의 연결 상태를 지정합니다.
WTS_INFO_CLASS WTSQuerySessionInformation 함수에 대한 호출에서 검색할 세션 정보의 유형을 나타내는 값을 포함합니다.
WTS_TYPE_CLASS 원격 데스크톱 서비스 함수가 버퍼에서 반환한 구조체의 형식을 지정합니다.
WTS_VIRTUAL_CLASS 검색할 가상 채널 정보의 유형을 나타내는 값을 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wtsdefs.h 헤더

아티클2023. 08. 28.

이 헤더는 원격 데스크톱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 데스크톱 서비스

wtsdefs.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

__WTS_PROPERTY_VALUE 프로토콜에서 검색할 속성 값에 대한 정보를 포함합니다.
_WTS_CACHE_STATS 프로토콜 캐시 통계를 포함합니다.
_WTS_CACHE_STATS_UN 캐시 통계를 포함합니다.
_WTS_CLIENT_DATA 클라이언트 연결에 대한 정보를 포함합니다.
_WTS_DISPLAY_IOCTL 클라이언트 디스플레이에 대한 정보를 포함합니다.
_WTS_LICENSE_CAPABILITIES 클라이언트의 라이선스 기능에 대한 정보를 포함합니다.
_WTS_PROTOCOL_CACHE 캐시 읽기 및 캐시 적중 횟수를 포함합니다.
_WTS_PROTOCOL_COUNTERS 프로토콜 성능 카운터를 포함합니다.
_WTS_PROTOCOL_STATUS

프로토콜의 상태 대한 정보를 포함합니다.
_WTS_SERVICE_STATE 원격 데스크톱 서비스 서비스의 상태 변경 내용에 대한 정보를 포함합니다.
_WTS_SESSION_ID 세션을 고유하게 식별하는 GUID를 포함합니다.
_WTS_SMALL_RECT 클라이언트 창 좌표를 포함합니다.
_WTS_SOCKADDR 소켓 주소를 포함합니다.
_WTS_SYSTEMTIME 표준 시간과 일광 절약 시간 사이의 전환 날짜 및 시간 정보를 지정합니다.
_WTS_TIME_ZONE_INFORMATION 클라이언트 표준 시간대 정보를 포함합니다.
_WTS_USER_CREDENTIAL 사용자에 대한 자격 증명 정보를 포함합니다.
WRDS_CONNECTION_SETTING 원격 데스크톱 연결에 대한 다양한 수준의 설정을 포함합니다.
WRDS_CONNECTION_SETTINGS 원격 세션에 대한 연결 설정 정보를 포함합니다. (WRDS_CONNECTION_SETTINGS)
WRDS_CONNECTION_SETTINGS_1 원격 세션에 대한 연결 설정 정보를 포함합니다. (WRDS_CONNECTION_SETTINGS_1)
WRDS_DYNAMIC_TIME_ZONE_INFORMATION 동적 표준 시간대 정보를 포함합니다.
WRDS_LISTENER_SETTING 원격 데스크톱 연결에 대한 다양한 수준의 수신기 설정을 포함합니다.

WRDS_LISTENER_SETTINGS 원격 세션에 대한 수신기 설정 정보를 포함합니다.
WRDS_LISTENER_SETTINGS_1 원격 세션에 대한 수신기 설정을 포함합니다.
WRDS_SETTING 원격 데스크톱 연결에 대한 다양한 수준의 정책 관련 설정을 포함합니다.
WRDS_SETTINGS 원격 세션에 대한 정책 관련 설정 정보를 포함합니다.
WRDS_SETTINGS_1 원격 세션에 대한 정책 관련 설정을 포함합니다.
WTS_POLICY_DATA 원격 데스크톱 서비스 서비스에서 프로토콜에 전달하는 정책 정보를 포함합니다.
WTS_USER_DATA 선택 클라이언트 속성 값을 포함합니다.

열거형

WRDS_CONNECTION_SETTING_LEVEL WRDS_CONNECTION_SETTINGS 구조체의 WRdsConnectionSetting 멤버에 포함된 구조체의 형식을 지정합니다.
WRDS_LISTENER_SETTING_LEVEL WRDS_LISTENER_SETTINGS 구조체의 WRdsListenerSetting 멤버에 포함된 구조체의 형식을 지정하는 데 사용됩니다.
WRDS_SETTING_LEVEL WRDS_SETTINGS 구조체의 WRdsSetting 멤버에 포함된 구조체의 형식을 지정합니다.

WRDS_SETTING_STATUS WRDS_SETTINGS_1 구조체의 다양한 멤버에 대한 정책 설정의 상태 지정합니다.
WRDS_SETTING_TYPE WRDS_SETTINGS 구조에 저장되는 설정의 범주를 지정합니다.
WTS_CERT_TYPE 라이선스를 획득하는 데 사용되는 인증서 유형을 지정하는 값을 포함합니다.
WTS_LOGON_ERROR_REDIRECTOR_RESPONSE 로그온 오류에 대한 프로토콜의 기본 응답을 지정하는 값을 포함합니다.
WTS_RCM_DRAIN_STATE RD 세션 호스트(원격 데스크톱 세션 호스트) 서버의 드레이닝 상태에 대한 정보를 포함합니다.
WTS_RCM_SERVICE_STATE 원격 데스크톱 서비스 서비스의 상태에 대한 정보를 포함합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wtshintapi.h 헤더

아티클2023. 08. 28.

이 헤더는 원격 데스크톱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 데스크톱 서비스

wtshintapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

WTSSetRenderHint 실제 콘텐츠인 창의 영역을 식별하기 위해 원격 세션에 표시하도록 최적화할 수 있는 콘텐츠를 표시하는 애플리케이션에서 사용됩니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wtsprotocol.h 헤더

아티클2023. 08. 28.

이 헤더는 원격 데스크톱 서비스에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 원격 데스크톱 서비스

wtsprotocol.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IWRdsEnhancedFastReconnectArbitrator 원격 데스크톱 서비스 서비스에서 호출한 메서드를 노출하여 향상된 빠른 다시 연결 시퀀스에서 다시 연결할 세션 ID를 가져옵니다.
IWRdsProtocolConnection 클라이언트 연결을 구성하기 위해 원격 데스크톱 서비스 서비스에서 호출한 메서드를 노출합니다.
IWRdsProtocolConnectionCallback 클라이언트 연결의 상태 대한 정보를 제공하고 클라이언트에 대한 작업을 수행하는 메서드를 노출합니다. 이 인터페이스는 원격 데스크톱 서비스 서비스에 의해 구현되고 프로토콜에 의해 호출됩니다.
IWRdsProtocolLicenseConnection 원격 데스크톱 서비스 서비스에서 연결 시퀀스 중에 라이선스 핸드셰이크를 수행하는 데 사용하는 메서드를 노출합니다.
IWRdsProtocolListener 프로토콜이 클라이언트 연결 요청의 수신 대기를 시작하고 중지할 것을 요청하는 메서드를 노출합니다.
IWRdsProtocolListenerCallback 클라이언트가 연결되었음을 원격 데스크톱 서비스 서비스에 알리는 메서드를 노출합니다.
IWRdsProtocolLogonErrorRedirector 원격 데스크톱 서비스 서비스에서 호출한 메서드를 노출하여 로그온 상태 업데이트하고 로그온 오류 메시지를 직접 전송하는 방법을 결정합니다.

IWRdsProtocolManager 원격 데스크톱 서비스 서비스에서 프로토콜 공급자와 통신하는 데 사용하는 메서드를 노출합니다.
IWRdsProtocolSettings 정책 관련 설정을 검색하고 추가하기 위한 메서드를 노출합니다.
IWRdsProtocolShadowCallback 원격 데스크톱 서비스 서비스에 그림자의 대상 쪽을 시작하거나 중지하도록 알리기 위해 프로토콜에서 호출한 메서드를 노출합니다.
IWRdsProtocolShadowConnection 프로토콜 공급자에게 세션 섀도의 상태 알리는 메서드를 노출합니다.
IWRdsWddmIddProps 이 인터페이스를 사용하면 사용자 지정 IDD 드라이버를 원격 세션에 로드할 수 있습니다.
IWTSProtocolConnection IWTSProtocolConnection을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnection을 사용합니다.
IWTSProtocolConnectionCallback IWTSProtocolConnectionCallback을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolConnectionCallback을 사용합니다.
IWTSProtocolLicenseConnection IWTSProtocolLicenseConnection을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolLicenseConnection을 사용합니다.
IWTSProtocolListener IWTSProtocolListener를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolListener를 사용합니다.
IWTSProtocolListenerCallback IWTSProtocolListenerCallback을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolListenerCallback을 사용합니다.
IWTSProtocolLogonErrorRedirector IWTSProtocolLogonErrorRedirector는 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolLogonErrorRedirector를 사용합니다.

IWTSProtocolManager IWTSProtocolManager를 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolManager를 사용합니다.
IWTSProtocolShadowCallback IWTSProtocolShadowCallback은 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolShadowCallback을 사용합니다.
IWTSProtocolShadowConnection IWTSProtocolShadowConnection을 더 이상 사용할 수 없습니다. 대신 IWRdsProtocolShadowConnection을 사용합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wtypes.h 헤더

아티클2023. 08. 28.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Automation
- COM(구성 요소 개체 모델)
- 개발자 정보
- 구조적 저장소
- Windows 속성 시스템

wtypes.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

구조체

CSPLATFORM 운영 체제 플랫폼 및 프로세서 아키텍처를 포함합니다.
Cy CY 구조는 돈과 관련된 계산이나 정확도가 특히 중요한 고정 소수점 계산에 유용합니다.
DECIMAL DECIMAL 구조체는 숫자에 대한 기호와 배율을 제공하는 10진수 데이터 형식을 나타냅니다.
PROPERTYKEY 프로그래밍 방식으로 속성을 식별하는 FMTID/PID 식별자를 지정합니다. SHCOLUMNID를 바꿉니다.
QUERYCONTEXT 클래스 구현을 조회하는 데 사용되는 특성 목록을 포함합니다.

열거형

DVASPECT

데이터를 그리거나 가져올 때 개체의 원하는 데이터 또는 뷰 모양을 지정합니다.
STATFLAG 메서드가 STATSTG 구조체의 pwcsName 멤버에서 이름을 반환해야 하는지 여부를 나타냅니다.
STGC IStorage::Commit 및 IStream::Commit 메서드에서 커밋 작업을 수행하기 위한 조건을 지정합니다.
STGMOVE 스토리지 요소를 이동하거나 복사할지 여부를 나타냅니다.
TYSPEC 클래스 ID에 대한 매핑을 지정합니다.
VARENUM 변형 형식을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wtypesbase.h 헤더

아티클2023. 08. 28.

이 헤더는 여러 기술에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- COM(구성 요소 개체 모델)
- Windows 소켓 2

wtypesbase.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

OLESTR 문자열 리터럴을 유니코드 문자열로 변환합니다.

구조체

BLOB Binary Large Object에서 파생된 BLOB(wtypesbase.h) 구조체에는 데이터 블록에 대한 정보가 포함됩니다.
COAUTHIDENTITY 사용자 이름 및 암호를 포함합니다.
COAUTHINFO 클라이언트 컴퓨터에서 서버 컴퓨터로 원격 활성화 요청을 수행하는 동안 사용되는 인증 설정을 포함합니다.
SECURITY_ATTRIBUTES SECURITY_ATTRIBUTES 구조체는 개체에 대한 보안 설명자를 포함하고 이 구조를 지정하여 검색된 핸들을 상속할 수 있는지 여부를 지정합니다.

열거형

CLSCTX 활성화 호출에 사용되는 값은 개체를 실행할 실행 컨텍스트를 나타냅니다.
MSHCTX unmarshaling을 수행할 프로세스인 대상 컨텍스트를 지정합니다.
MSHLFLAGS 마샬링을 수행하는 이유를 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

wuapi.h 헤더

아티클2023. 08. 28.

이 헤더는 Windows 업데이트 에이전트 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows 업데이트 에이전트 API

wuapi.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IAutomaticUpdates 자동 업데이트 기능을 포함합니다. (IAutomaticUpdates)
IAutomaticUpdates2 자동 업데이트 기능을 포함합니다. (IAutomaticUpdates2)
IAutomaticUpdatesResults 자동 업데이트 설명하는 읽기 전용 속성을 포함합니다.
IAutomaticUpdatesSettings 자동 업데이트 사용할 수 있는 설정을 포함합니다. (IAutomaticUpdatesSettings)
IAutomaticUpdatesSettings2 자동 업데이트 사용할 수 있는 설정을 포함합니다. (IAutomaticUpdatesSettings2)
IAutomaticUpdatesSettings3 자동 업데이트 사용할 수 있는 설정을 포함합니다. (IAutomaticUpdatesSettings3)
ICategory 업데이트가 속한 범주를 나타냅니다.
ICategoryCollection ICategory 인터페이스의 순서가 지정된 읽기 전용 목록을 나타냅니다.

IDownloadCompletedCallback 비동기 다운로드가 완료된 경우 사용되는 콜백을 제공합니다.
IDownloadCompletedCallbackArgs 다운로드 완료에 대한 정보를 포함합니다. 이 인터페이스는 IDownloadCompletedCallback 대리자에 대한 매개 변수 역할을 합니다. 업데이트의 다운로드 및 설치는 비동기적입니다.
IDownloadJob 다운로드 작업에 사용할 수 있는 속성 및 메서드를 포함합니다.
IDownloadProgress 비동기 다운로드 작업의 진행률을 나타냅니다.
IDownloadProgressChangedCallback 비동기 다운로드 작업의 진행 상황을 나타내는 알림을 처리합니다.
IDownloadProgressChangedCallbackArgs 비동기 다운로드 작업의 진행률 변경에 대한 정보를 포함합니다.
IDownloadResult 다운로드 작업의 결과를 나타냅니다.
IImageInformation 업데이트 또는 범주와 연결된 지역화된 이미지에 대한 정보를 포함합니다.
IInstallationAgent 업데이트 결과를 기록합니다.
IInstallationBehavior 업데이트의 설치 및 제거 옵션을 나타냅니다.
IInstallationCompletedCallback 비동기 설치 또는 제거가 완료되었음을 나타내는 알림을 처리합니다.
IInstallationCompletedCallbackArgs 설치 완료에 대한 정보를 포함하고 IInstallationCompletedCallback 대리자에 대한 매개 변수 역할을 합니다. 업데이트의 다운로드 및 설치는 비동기적입니다.

IInstallationJob 설치 또는 제거 작업에 사용할 수 있는 속성과 메서드를 포함합니다.
IInstallationProgress 비동기 설치 또는 제거의 진행률을 나타냅니다.
IInstallationProgressChangedCallback 비동기 설치 또는 제거의 진행 상황에 대한 알림을 처리하는 Invoke 메서드를 정의합니다.
IInstallationProgressChangedCallbackArgs 콜백이 수행되었을 때 비동기 설치 또는 제거 진행률의 변경 내용에 대한 정보를 포함합니다.
IInstallationResult 설치 또는 제거의 결과를 나타냅니다.
IInvalidProductLicenseException 제품에 대해 잘못된 라이선스가 검색될 때 throw되는 예외를 캡슐화합니다.
ISearchCompletedCallback 비동기 검색 작업의 완료에 대한 알림을 처리하는 메서드를 포함합니다.
ISearchCompletedCallbackArgs 비동기 검색 완료에 대한 정보를 포함합니다. SearchCompletedCallback 대리자에 대한 매개 변수로도 작동합니다.
ISearchJob 검색 작업에 사용할 수 있는 속성과 메서드를 포함합니다.
ISearchResult 검색 결과를 나타냅니다.
IStringCollection 정렬된 문자열 목록을 나타냅니다.
ISystemInformation 지정된 컴퓨터에 대한 정보를 포함합니다. 이 정보는 WUA(Windows 업데이트 에이전트)와 관련이 있습니다.

IUpdate 업데이트에 사용할 수 있는 속성 및 메서드를 포함합니다. (IUpdate)
IUpdate2 업데이트에 사용할 수 있는 속성 및 메서드를 포함합니다. (IUpdate2)
IUpdate3 업데이트에 사용할 수 있는 속성 및 메서드를 포함합니다. (IUpdate3)
IUpdate4 업데이트에 사용할 수 있는 속성 및 메서드를 포함합니다. (IUpdate4)
IUpdate5 업데이트에 사용할 수 있는 속성 및 메서드를 포함합니다. (IUpdate5)
IUpdateCollection 정렬된 업데이트 목록을 나타냅니다.
IUpdateDownloadContent 업데이트의 다운로드 콘텐츠를 나타냅니다. (IUpdateDownloadContent)
IUpdateDownloadContent2 업데이트의 다운로드 콘텐츠를 나타냅니다. (IUpdateDownloadContent2)
IUpdateDownloadContentCollection 업데이트에 대한 다운로드 콘텐츠의 컬렉션을 나타냅니다.
IUpdateDownloader 서버에서 업데이트를 다운로드합니다.
IUpdateDownloadResult 업데이트에 대한 다운로드 작업의 상태 나타내는 속성을 포함합니다.
IUpdateException 불완전한 ISearchResult 개체에서 반환된 검색 결과의 측면에 대한 정보를 나타냅니다.
IUpdateExceptionCollection

IUpdateException 인터페이스의 순서가 지정된 읽기 전용 목록을 나타냅니다.
IUpdateHistoryEntry 업데이트의 기록된 기록을 나타냅니다. (IUpdateHistoryEntry)
IUpdateHistoryEntry2 업데이트의 기록된 기록을 나타냅니다. (IUpdateHistoryEntry2)
IUpdateHistoryEntryCollection IUpdateHistoryEntry 인터페이스의 순서가 지정된 읽기 전용 목록을 나타냅니다.
IUpdateIdentity 업데이트의 고유 식별자를 나타냅니다.
IUpdateInstallationResult 업데이트 설치 또는 제거의 상태 사용할 수 있는 속성과 메서드를 포함합니다.
IUpdateInstaller 컴퓨터에서 또는 컴퓨터에 업데이트를 설치하거나 제거합니다.
IUpdateInstaller2 컴퓨터에 업데이트를 설치하거나 제거합니다. (IUpdateInstaller2)
IUpdateInstaller3 컴퓨터에 업데이트를 설치하거나 제거합니다. (IUpdateInstaller3)
IUpdateInstaller4 이전에 스테이징되거나 설치된 업데이트를 마무리하는 메서드를 제공합니다.
IUpdateLockdown 이 인터페이스의 메서드를 구현하는 개체의 메서드 및 속성에 대한 액세스를 제한합니다.
IUpdateSearcher 서버에서 업데이트를 검색합니다. (IUpdateSearcher)
IUpdateSearcher2 서버에서 업데이트를 검색합니다. (IUpdateSearcher2)

IUpdateSearcher3 서버에서 업데이트를 검색합니다. (IUpdateSearcher3)
IUpdateService WUA(Windows 업데이트 Agent) 또는 자동 업데이트 등록된 서비스에 대한 정보를 포함합니다. (IUpdateService)
IUpdateService2 WUA(Windows 업데이트 Agent) 또는 자동 업데이트 등록된 서비스에 대한 정보를 포함합니다. (IUpdateService2)
IUpdateServiceCollection IUpdateService 인터페이스 목록을 나타냅니다.
IUpdateServiceManager Windows 업데이트 에이전트 또는 자동 업데이트 업데이트 서비스의 등록을 추가하거나 제거합니다. (IUpdateServiceManager)
IUpdateServiceManager2 Windows 업데이트 에이전트 또는 자동 업데이트 업데이트 서비스의 등록을 추가하거나 제거합니다. (IUpdateServiceManager2)
IUpdateServiceRegistration 서비스의 등록 상태에 대한 정보를 포함합니다.
IUpdateSession 호출자가 업데이트를 포함하는 작업을 수행할 수 있는 세션을 나타냅니다. 예를 들어 이 인터페이스는 호출자가 검색, 다운로드, 설치 또는 제거 작업을 수행하는 세션을 나타냅니다. (IUpdateSession)
IUpdateSession2 호출자가 업데이트를 포함하는 작업을 수행할 수 있는 세션을 나타냅니다. 예를 들어 이 인터페이스는 호출자가 검색, 다운로드, 설치 또는 제거 작업을 수행하는 세션을 나타냅니다. (IUpdateSession2)
IUpdateSession3 호출자가 업데이트를 포함하는 작업을 수행할 수 있는 세션을 나타냅니다. 예를 들어 이 인터페이스는 호출자가 검색, 다운로드, 설치 또는 제거 작업을 수행하는 세션을 나타냅니다. (IUpdateSession3)

IWebProxy HTTP 프록시 설정을 포함합니다.
IWindowsDriverUpdate Windows 드라이버 업데이트에서만 사용할 수 있는 속성과 메서드를 포함합니다.
IWindowsDriverUpdate2 Windows 드라이버 업데이트에서만 사용할 수 있는 속성과 메서드를 포함합니다. (IWindowsDriverUpdate2)
IWindowsDriverUpdate3 Windows 드라이버 업데이트에서만 사용할 수 있는 속성과 메서드를 포함합니다. (IWindowsDriverUpdate3)
IWindowsDriverUpdate4 Windows 드라이버 업데이트에서만 사용할 수 있는 속성과 메서드를 포함합니다. (IWindowsDriverUpdate4)
IWindowsDriverUpdate5 Windows 드라이버 업데이트에서만 사용할 수 있는 속성과 메서드를 포함합니다. (IWindowsDriverUpdate5)
IWindowsDriverUpdateEntry Windows 드라이버 업데이트에서만 사용할 수 있는 속성을 포함합니다.
IWindowsDriverUpdateEntryCollection 드라이버 업데이트와 연결된 드라이버 업데이트 항목의 컬렉션을 포함합니다. 모든 속성에는 표준 컬렉션 의미 체계가 있습니다.
IWindowsUpdateAgentInfo WUA(Windows 업데이트 Agent) 버전에 대한 정보를 검색합니다.

열거형

AddServiceFlag

IUpdateServiceManager2 인터페이스가 서비스 등록 요청을 처리할 수 있는 가능한 방법을 정의합니다.
AutoDownloadMode 자동 업데이트 컴퓨터에 적용 가능한 것으로 확인되면 업데이트를 자동으로 다운로드할지 여부를 결정하는 데 사용되는 논리 유형을 정의합니다.
AutomaticUpdatesNotificationLevel 관리자 권한 사용자에게 자동 업데이트 이벤트에 대한 알림을 받을 수 있는 방법을 정의합니다.
AutomaticUpdatesPermissionType IAutomaticUpdatesSettings 인터페이스의 NotificationLevel 속성 또는 IAutomaticUpdatesSettings2 인터페이스의 IncludeRecommendedUpdates 속성을 설정하는 가능한 방법을 정의합니다.
AutomaticUpdatesScheduledInstallationDay 자동 업데이트 업데이트를 설치하거나 제거하는 요일을 정의합니다.
AutomaticUpdatesUserType 사용자 유형을 정의합니다.
AutoSelectionMode 사용자가 Windows 업데이트 사용자 인터페이스에서 사용 가능한 업데이트를 볼 때 특정 업데이트를 자동으로 선택할지 여부를 결정하는 데 사용되는 논리 유형을 정의합니다.
DeploymentAction 업데이트가 명시적으로 배포되는 작업을 정의합니다.
DownloadPhase IDownloadProgress 인터페이스의 CurrentUpdateDownloadPhase 속성에서 반환되는 현재 업데이트 다운로드 진행률을 정의합니다.
DownloadPriority 다운로드 작업의 가능한 우선 순위를 정의합니다.
InstallationImpact 업데이트를 설치하거나 제거하여 발생할 수 있는 가능한 영향 수준을 정의합니다.
InstallationRebootBehavior

업데이트에 대해 가능한 다시 시작 동작을 정의합니다.
OperationResultCode 업데이트에 대한 다운로드, 설치, 제거 또는 확인 작업의 가능한 결과를 정의합니다.
SearchScope search:_per-machine 업데이트, 사용자별 업데이트 또는 둘 다에서 반환해야 하는 다양한 업데이트를 정의합니다.
UpdateExceptionContext IUpdateException 개체를 제공할 수 있는 컨텍스트를 정의합니다.
UpdateLockdownOption WUA(Windows 업데이트 Agent) 개체가 Windows 업데이트 액세스할 수 있는 기능을 정의합니다.
UpdateOperation 업데이트에서 시도할 수 있는 작업을 정의합니다.
UpdateServiceOption 검사 패키지 서비스에 대한 서비스 등록을 제거하는 방법에 영향을 주는 옵션을 정의합니다.
UpdateServiceRegistrationState 업데이트 서비스에 사용할 수 있는 상태를 정의합니다.
UpdateType 드라이버 또는 소프트웨어 업데이트와 같은 업데이트 유형을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

x3daudio.h 헤더

아티클2023. 08. 28.

이 헤더는 XAudio2 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- XAudio2 API

x3daudio.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

X3DAudioCalculate 3D 매개 변수와 관련하여 DSP 설정을 계산합니다.
X3DAudioInitialize 모든 전역 3D 오디오 상수를 설정합니다.

구조체

X3DAUDIO_CONE 발광자의 방향과 관련하여 DSP 동작의 크기를 조정하여 단일 채널이 아닌 LFE 발광자의 방향성을 지정합니다.
X3DAUDIO_DISTANCE_CURVE 선형 세그먼트로 구성된 명시적 조각 곡선을 정의하여 정규화된 거리와 관련하여 DSP 동작을 직접 정의합니다.
X3DAUDIO_DISTANCE_CURVE_POINT 지정된 정규화된 거리에서 DSP 설정을 정의합니다.
X3DAUDIO_DSP_SETTINGS X3DAudioCalculate 호출에서 결과를 받습니다.
X3DAUDIO_EMITTER

임의의 수의 사운드 채널과 함께 사용되는 단일 지점 또는 다중 지점 3D 오디오 원본을 정의합니다.
X3DAUDIO_LISTENER 3D 오디오 수신 지점을 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

xamlom.h 헤더

아티클2023. 08. 28.

이 헤더는 XAML 진단에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- XAML 진단

xamlom.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IBitmapData 시각적 트리의 노드와 연결된 이미지를 나타냅니다.
IVisualTreeService XAML 시각적 트리를 관리하는 메서드를 제공합니다.
IVisualTreeService2 IVisualTreeService 개체의 추가 기능을 나타냅니다.
IVisualTreeService3 IVisualTreeService2 개체의 추가 기능을 나타냅니다.
IVisualTreeServiceCallback 시각적 트리의 상태를 전달합니다.
IVisualTreeServiceCallback2 IVisualTreeServiceCallback 개체의 추가 기능을 나타냅니다.
IXamlDiagnostics XAML 진단 세션을 나타냅니다.

Functions

InitializeXamlDiagnosticsEx Xaml Diagnostics 세션을 초기화합니다. XAML 진단 API를 사용하는 디버깅 도구의 진입점입니다.

구조체

BitmapDescription IBitmapData에 저장된 비트맵에 대한 정보를 나타냅니다.
CollectionElementValue 컬렉션의 요소를 나타냅니다.
EnumType XAML 런타임 열거형을 나타냅니다.
ParentChildRelation 부모 개체를 자식 개체와 연결합니다.
PropertyChainSource 대상 형식의 원본 개체(Style)를 나타냅니다.
PropertyChainValue 요소에 정의된 속성을 나타냅니다.
SourceInfo 개체의 XAML 원본 문서에 대한 정보를 나타냅니다.
VisualElement Microsoft Visual Studio의 라이브 시각적 트리에 있는 XAML 요소를 나타냅니다.

열거형

BaseValueSource 속성의 유효 값이 설정된 위치를 지정하는 상수를 정의합니다.
MetadataBit XAML 진단에서 반환된 PropertyChainValue를 정의하는 데 사용되는 상수를 정의합니다.
RenderTargetBitmapOptions 시각적 트리의 렌더링할 부분을 지정하는 상수를 정의합니다.
ResourceType 리소스 사전에서 리소스의 형식을 지정하는 상수를 정의합니다.
VisualElementState 시각적 트리에서 요소의 상태를 지정하는 상수를 정의합니다.
VisualMutationType 요소가 라이브 시각적 트리에 추가되거나 제거되었는지 여부를 지정하는 상수를 정의합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

xapo.h 헤더

아티클2023. 08. 28.

이 헤더는 XAudio2 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- XAudio2 API

xapo.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IXAPO XAudio2 효과 체인에서 사용되는 오디오 처리 개체의 인터페이스입니다.
IXAPOParameters XAPO가 효과별 매개 변수를 사용할 수 있도록 하는 선택적 인터페이스입니다.

Functions

XAPOAlloc 이후에 애플리케이션에 반환되는 임의 크기의 구조를 할당해야 하는 IXAPO 메서드에서 사용하는 메모리 할당 매크로입니다.
XAPOFree XAPOAlloc 매크로를 사용하여 할당된 메모리를 해제하는 데 사용되는 매크로입니다.

구조체

XAPO_LOCKFORPROCESS_PARAMETERS XAPO가 잠겨 있는 동안 일정하게 유지되는 스트림 버퍼 매개 변수를 정의합니다. IXAPO::LockForProcess 메서드와 함께 사용됩니다.

XAPO_PROCESS_BUFFER_PARAMETERS 한 호출에서 다음 호출로 변경할 수 있는 스트림 버퍼 매개 변수를 정의합니다. Process 메서드와 함께 사용됩니다.
XAPO_REGISTRATION_PROPERTIES XAPO의 일반적인 특징을 설명합니다. IXAPO::GetRegistrationProperties, CXAPOParametersBase::CXAPOParametersBase 및 CXAPOBase::CXAPOBase와 함께 사용됩니다.

열거형

XAPO_BUFFER_FLAGS 스트림 버퍼의 내용을 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

xapobase.h 헤더

아티클2023. 08. 28.

이 헤더는 XAudio2 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- XAudio2 API

xapobase.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

클래스

CXAPOBase IXAPO 인터페이스의 기본 구현입니다.
CXAPOParametersBase IXAPOParameters 인터페이스의 기본 구현입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

xapofx.h 헤더

아티클2023. 08. 28.

이 헤더는 XAudio2 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- XAudio2 API

xapofx.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

CreateFX 요청된 XAPOFX 효과의 instance 만듭니다.

구조체

FXECHO_INITDATA FXECHO XAPOFX와 함께 사용할 초기화 매개 변수입니다.
FXECHO_PARAMETERS FXECHO XAPOFX와 함께 사용할 매개 변수입니다.
FXEQ_PARAMETERS FXEQ XAPO에 사용할 매개 변수입니다.
FXMASTERINGLIMITER_PARAMETERS FXMasteringLimiter XAPO에 사용할 매개 변수입니다.
FXREVERB_PARAMETERS FXReverb XAPO와 함께 사용할 매개 변수입니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

xaudio2.h 헤더

아티클2023. 08. 28.

이 헤더는 XAudio2 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- XAudio2 API

xaudio2.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IXAudio2 IXAudio2는 모든 오디오 엔진 상태, 오디오 처리 스레드, 음성 그래프 등을 관리하는 XAudio2 개체의 인터페이스입니다.
IXAudio2EngineCallback IXAudio2EngineCallback 인터페이스에는 IXAudio2 엔진에서 특정 이벤트가 발생할 때 클라이언트에 알리는 메서드가 포함되어 있습니다.
IXAudio2MasteringVoice 마스터링 음성은 오디오 출력 디바이스를 나타내는 데 사용됩니다.
IXAudio2SourceVoice 원본 음성을 사용하여 오디오 데이터를 XAudio2 처리 파이프라인에 제출합니다.
IXAudio2SubmixVoice 서브믹스 음성은 주로 성능 향상 및 효과 처리에 사용됩니다.
IXAudio2Voice IXAudio2Voice는 IXAudio2SourceVoice, IXAudio2SubmixVoice 및 IXAudio2MasteringVoice가 파생되는 기본 인터페이스를 나타냅니다. 아래에 나열된 메서드는 모든 음성 서브클래스에 공통적으로 적용됩니다.
IXAudio2VoiceCallback IXAudio2VoiceCallback 인터페이스에는 지정된 IXAudio2SourceVoice에서 특정 이벤트가 발생할 때 클라이언트에 알리는 메서드가 포함되어 있습니다.

Functions

XAudio2AmplitudeRatioToDecibels 진폭 비율 값을 데시벨 값으로 변환하는 인라인 함수입니다.
XAudio2만들기 새 XAudio2 개체를 만들고 IXAudio2 인터페이스에 대한 포인터를 반환합니다.
XAudio2CutoffFrequencyToOnePoleCoefficient 헤르츠로 표현된 필터 차단 주파수에서 XAUDIO2_FILTER_PARAMETERS 구조체의 Frequency 멤버와 함께 사용되는 필터 계수로 변환하는 인라인 함수입니다.
XAudio2CutoffFrequencyToRadians 헤르츠로 표현된 필터 차단 주파수에서 XAUDIO2_FILTER_PARAMETERS 구조체의 Frequency 멤버에 사용되는 라디안 주파수 값으로 변환하는 인라인 함수입니다.
XAudio2DecibelsToAmplitudeRatio decibel 값을 진폭 비율 값으로 변환하는 인라인 함수입니다.
XAudio2FrequencyRatioToSemitones 빈도 비율 값을 세미톤 값으로 변환하는 인라인 함수입니다.
XAudio2RadiansToCutoffFrequency XAUDIO2_FILTER_PARAMETERS 사용되는 라디안 주파수에서 hertz의 절대 주파수로 다시 변환하는 인라인 함수입니다.
XAudio2SemitonesToFrequencyRatio 세미톤 값을 주파수 비율 값으로 변환하는 인라인 함수입니다.

구조체

XAUDIO2_BUFFER IXAudio2SourceVoice::SubmitSourceBuffer와 함께 사용되는 오디오 데이터 버퍼를 나타냅니다.

XAUDIO2_BUFFER_WMA xWMA 데이터를 제출할 때 IXAudio2SourceVoice::SubmitSourceBuffer와 함께 사용됩니다.
XAUDIO2_DEBUG_CONFIGURATION XAudio2에 대한 새 전역 디버그 구성을 포함합니다. SetDebugConfiguration 함수와 함께 사용됩니다.
XAUDIO2_EFFECT_CHAIN 효과 체인을 정의합니다.
XAUDIO2_EFFECT_DESCRIPTOR 효과 체인에 사용할 XAPO에 대한 정보를 포함합니다.
XAUDIO2_FILTER_PARAMETERS 원본 음성에 대한 필터 매개 변수를 정의합니다.
XAUDIO2_PERFORMANCE_DATA 성능 정보를 포함합니다. (XAUDIO2_PERFORMANCE_DATA)
XAUDIO2_SEND_DESCRIPTOR 다른 음성에서 보내는 대상인 대상 음성을 정의하고 필터를 사용할지 여부를 지정합니다.
XAUDIO2_VOICE_DETAILS 음성의 생성 플래그, 입력 채널 및 샘플 속도에 대한 정보를 포함합니다.
XAUDIO2_VOICE_SENDS 단일 출력 음성에서 데이터를 받을 음성 집합을 정의합니다.
XAUDIO2_VOICE_STATE 음성의 현재 상태 및 커서 위치 데이터를 반환합니다.

열거형

XAUDIO2_FILTER_TYPE

필터 유형을 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

xaudio2fx.h 헤더

아티클2023. 08. 28.

이 헤더는 XAudio2 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- XAudio2 API

xaudio2fx.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

ReverbConvertI3DL2ToNative I3DL2(대화형 3D 오디오 렌더링 지침 수준 2.0) 매개 변수를 네이티브 XAudio2 매개 변수로 변환하는 인라인 함수입니다.
XAudio2CreateReverb 새 반향 오디오 처리 개체(APO)를 만들고 포인터를 반환합니다.
XAudio2CreateVolumeMeter 새 볼륨 미터 APO(오디오 처리 개체)를 만들고 포인터를 반환합니다.

구조체

XAUDIO2FX_REVERB_I3DL2_PARAMETERS ReverbConvertI3DL2ToNative 함수에서 사용할 I3DL2(대화형 3D 오디오 렌더링 지침 수준 2.0) 매개 변수에 대해 설명합니다.
XAUDIO2FX_REVERB_PARAMETERS 반향 APO에서 사용할 매개 변수를 설명합니다.
XAUDIO2FX_VOLUMEMETER_LEVELS 볼륨 미터 APO와 함께 사용할 매개 변수에 대해 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

xblidpauthmanager.h 헤더

아티클2023. 08. 28.

이 헤더는 XblIdpAuthManager에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- XblIdpAuthManager

xblidpauthmanager.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IXblIdpAuthManager Microsoft용으로 예약되어 있습니다. (IXblIdpAuthManager)
IXblIdpAuthTokenResult Microsoft용으로 예약되어 있습니다. (IXblIdpAuthTokenResult)

열거형

XBL_IDP_AUTH_TOKEN_STATUS Microsoft용으로 예약되어 있습니다. (XBL_IDP_AUTH_TOKEN_STATUS)

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

xenroll.h 헤더

아티클2023. 08. 28.

이 헤더는 보안 및 ID에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- 보안 및 ID

xenroll.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

ICEnroll ICEnroll 인터페이스는 인증서 등록 제어를 나타내는 여러 인터페이스 중 하나입니다.
ICEnroll2 ICEnroll2 인터페이스는 인증서 등록 제어를 나타내는 여러 인터페이스 중 하나입니다.
ICEnroll3 인증서 등록 제어를 나타내는 여러 인터페이스 중 하나입니다.
ICEnroll4 ICEnroll4 인터페이스는 인증서 등록 제어를 나타내는 여러 인터페이스 중 하나입니다.
IEnroll 인증서 등록 제어를 나타내며 주로 인증서 요청을 생성하는 데 사용됩니다. (IEnroll)
IEnroll2 인증서 등록 제어를 나타내며 주로 인증서 요청을 생성하는 데 사용됩니다. (IEnroll2)
IEnroll4 IEnroll4 인터페이스는 인증서 등록 제어를 나타내며 주로 인증서 요청을 생성하는 데 사용됩니다.

피드백

이 페이지가 도움이 되었나요?

Yes No

Microsoft Q&A에서 도움말 보기

xinput.h 헤더

아티클2023. 10. 28.

이 헤더는 XInput 게임 컨트롤러 API에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- XInput 게임 컨트롤러 API

xinput.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

Functions

XInputEnable XInput의 보고 상태를 설정합니다.
XInputGetAudioDeviceIds 지정된 컨트롤러에 연결된 헤드셋과 연결된 사운드 렌더링 및 사운드 캡처 오디오 디바이스 ID를 검색합니다.
XInputGetBatteryInformation 무선 컨트롤러의 배터리 종류 및 충전 상태 검색합니다.
XInputGetCapabilities 연결된 컨트롤러의 기능 및 기능을 검색합니다.
XInputGetDSoundAudioDeviceGuids 지정된 컨트롤러에 연결된 헤드셋과 연결된 사운드 렌더링 및 사운드 캡처 디바이스 GUID를 가져옵니다.
XInputGetKeystroke 게임 패드 입력 이벤트를 검색합니다.
XInputGetState 지정된 컨트롤러의 현재 상태를 검색합니다.
XInputSetState 연결된 컨트롤러에 데이터를 보냅니다. 이 함수는 컨트롤러의 진동 함수를 활성화하는 데 사용됩니다.

구조체

XINPUT_BATTERY_INFORMATION 배터리 종류 및 충전 상태에 대한 정보를 포함합니다.
XINPUT_CAPABILITIES 연결된 컨트롤러의 기능을 설명합니다. XInputGetCapabilities 함수는 XINPUT_CAPABILITIES 반환합니다.
XINPUT_GAMEPAD 컨트롤러의 현재 상태를 설명합니다.
XINPUT_KEYSTROKE XInputGetKeystroke에서 반환된 키 입력 데이터를 지정합니다.
XINPUT_STATE 컨트롤러의 상태를 나타냅니다.
XINPUT_VIBRATION 컨트롤러의 진동 함수에 대한 모터 속도 수준을 지정합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

xpsdigitalsignature.h 헤더

아티클2023. 08. 28.

이 헤더는 XPS 문서에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- XPS 문서

xpsdigitalsignature.h에는 다음 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IXpsSignature 단일 디지털 서명을 나타냅니다.
IXpsSignatureBlock SignatureDefinitions 부분에 저장된 서명 요청 블록을 나타냅니다.
IXpsSignatureBlockCollection IXpsSignatureBlock 인터페이스의 컬렉션입니다.
IXpsSignatureCollection IXpsSignature 인터페이스 포인터의 컬렉션입니다.
IXpsSignatureManager XPS 문서의 디지털 서명 및 디지털 서명 요청을 관리합니다.
IXpsSignatureRequest 서명 요청의 구성 요소에 액세스합니다.
IXpsSignatureRequestCollection IXpsSignatureRequest 인터페이스의 컬렉션입니다.
IXpsSigningOptions 새 서명에 사용되는 개별 서명 옵션에 대한 액세스를 제공합니다.

열거형

XPS_SIGN_FLAGS 서명하기 전에 태그 호환성 검색을 실행해야 하는지 여부를 지정합니다.
XPS_SIGN_POLICY XPS 문서의 선택적 부분이 있는 경우 서명되는 부분을 나타내는 비트 열거자입니다.
XPS_SIGNATURE_STATUS 문서 디지털 서명의 상태 설명합니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기

xpsobjectmodel.h 헤더

아티클2023. 03. 16.

이 헤더는 XPS 문서에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- XPS 문서

xpsobjectmodel.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IXpsOMBrush 그래픽 개체를 그리는 데 사용되는 개체를 정의합니다. IXpsOMBrush에서 파생되는 클래스는 영역이 그려지는 방법을 설명합니다.
IXpsOMCanvas 시각적 요소 및 관련 속성의 그룹입니다.
IXpsOMColorProfileResource 색 프로필 리소스에 IStream 인터페이스를 제공합니다.
IXpsOMColorProfileResourceCollection IXpsOMColorProfileResource 인터페이스 포인터의 컬렉션입니다.
IXpsOMCoreProperties 이 인터페이스는 XPS 문서의 핵심 속성 부분에 저장된 메타데이터에 대한 액세스를 제공합니다.
IXpsOMDashCollection XPS_DASH 구조체의 컬렉션입니다.
IXpsOMDictionary 사전은 XPS 패키지에서 리소스를 공유하는 데 사용됩니다.
IXpsOMDocument 문서를 구성하는 고정 페이지 및 문서 수준 리소스의 순서가 지정된 시퀀스입니다.

IXpsOMDocumentCollection IXpsOMDocument 인터페이스 포인터의 컬렉션입니다.
IXpsOMDocumentSequence XPS 문서 콘텐츠가 있는 루트 개체입니다.
IXpsOMDocumentStructureResource DocumentStructure 파트의 리소스 스트림에 대한 XML 콘텐츠에 대한 액세스를 제공합니다.
IXpsOMFontResource 글꼴 리소스에 IStream 인터페이스를 제공합니다.
IXpsOMFontResourceCollection IXpsOMFontResource 인터페이스 포인터의 컬렉션입니다.
IXpsOMGeometry 경로 또는 클리핑 영역의 모양을 설명합니다.
IXpsOMGeometryFigure IXpsOMGeometry 인터페이스에 의해 지정된 경로 또는 클리핑 영역의 한 부분을 설명합니다.
IXpsOMGeometryFigureCollection IXpsOMGeometryFigure 인터페이스 포인터의 컬렉션입니다.
IXpsOMGlyphs 페이지에 표시되는 텍스트를 설명합니다.
IXpsOMGlyphsEditor IXpsOMGlyphs 인터페이스의 텍스트 콘텐츠에 영향을 주는 속성을 일괄 수정할 수 있습니다.
IXpsOMGradientBrush 이 인터페이스는 그라데이션 중지점으로 구성된 그라데이션을 설명합니다. IXpsOMGradientBrush에서 상속되는 클래스는 그라데이션 중지를 해석하는 다양한 방법을 지정합니다.
IXpsOMGradientStop 그라데이션 내의 단일 색과 위치를 나타냅니다.

IXpsOMGradientStopCollection IXpsOMGradientStop 인터페이스 포인터의 컬렉션입니다.
IXpsOMImageBrush 래스터 이미지를 소스로 사용하는 브러시입니다.
IXpsOMImageResource 이미지 리소스에 IStream 인터페이스를 제공합니다.
IXpsOMImageResourceCollection IXpsOMImageResource 인터페이스 포인터의 컬렉션입니다.
IXpsOMLinearGradientBrush 선형 그라데이션을 지정합니다. 이 그라데이션은 벡터를 따라 색 그라데이션입니다.
IXpsOMMatrixTransform 개체 모델의 다른 개체에 적용할 수 있는 아핀 행렬 변환을 지정합니다.
IXpsOMNameCollection 이름 문자열의 컬렉션입니다.
IXpsOMObjectFactory XPS 문서 개체 모델에 개체를 만듭니다.
IXpsOMPackage XPS 개체 모델 트리에 대한 최상위 항목을 제공합니다.
IXpsOMPackageTarget StartXpsPrintJob1 함수에서 만든 인쇄 작업에서 사용할 수 있는 IXpsOMPackageWriter를 만드는 메서드를 제공합니다.
IXpsOMPackageWriter XPS 문서의 일부를 패키지 파일에 증분 방식으로 씁니다.
IXpsOMPage 단일 페이지의 내용을 포함하는 개체 트리의 루트 노드를 제공합니다.

IXpsOMPageReference XPS 문서에서 페이지의 가상화를 사용하도록 설정합니다.
IXpsOMPageReferenceCollection IXpsOMPageReference 인터페이스 포인터의 컬렉션입니다.
IXpsOMPart 모든 XPS 문서 파트 인터페이스에 대한 기본 인터페이스입니다.
IXpsOMPartResources XPS 문서의 모든 공유 부분 기반 리소스에 대한 액세스를 제공합니다.
IXpsOMPartUriCollection IOpcPartUri 인터페이스 포인터의 컬렉션입니다.
IXpsOMPath 텍스트가 아닌 시각적 개체 항목을 설명합니다.
IXpsOMPrintTicketResource PrintTicket 리소스에 IStream 인터페이스를 제공합니다.
IXpsOMRadialGradientBrush 방사형 그라데이션을 지정합니다.
IXpsOMRemoteDictionaryResource XPS 패키지의 페이지가 리소스를 공유할 수 있도록 하는 인터페이스를 제공합니다.
IXpsOMRemoteDictionaryResourceCollection IXpsOMRemoteDictionaryResource 인터페이스 포인터의 컬렉션입니다.
IXpsOMResource XPS 개체 모델의 리소스 인터페이스에 대한 기본 인터페이스로 사용됩니다.
IXpsOMShareable 공유 가능한 인터페이스의 기본 인터페이스입니다.

IXpsOMSignatureBlockResource 서명 블록 리소스에 IStream 인터페이스를 제공합니다.
IXpsOMSignatureBlockResourceCollection IXpsOMSignatureBlockResource 인터페이스 포인터의 컬렉션입니다.
IXpsOMSolidColorBrush 단일 색 브러시입니다.
IXpsOMStoryFragmentsResource 페이지의 StoryFragments 부분의 리소스 스트림 콘텐츠에 대한 액세스를 제공합니다.
IXpsOMThumbnailGenerator 썸네일 이미지 리소스를 생성합니다.
IXpsOMTileBrush 타일 브러시는 시각적 이미지를 사용하여 이미지를 반복하여 영역을 그립니다.
IXpsOMVisual 경로, 캔버스 및 문자 모양 인터페이스에 대한 기본 인터페이스입니다.
IXpsOMVisualBrush 시각적 요소를 원본으로 사용하는 브러시입니다.
IXpsOMVisualCollection IXpsOMVisual 인터페이스 포인터의 컬렉션입니다.

구조체

XPS_COLOR colorType이 XPS_COLOR_TYPE_CONTEXT 경우 XPS_COLOR 구조체의 내용입니다.
XPS_DASH 이 구조체는 경로의 대시 요소를 설명합니다.

XPS_GLYPH_INDEX 문자 모양의 배치 및 위치를 설명합니다.
XPS_GLYPH_MAPPING 문자 모양-인덱스 매핑에 대해 설명합니다.
XPS_MATRIX 3-by-3 행렬의 왼쪽 두 열을 설명합니다.
XPS_POINT 2차원 공간에서 x 및 y 좌표 쌍을 나타냅니다. (XPS_POINT)
XPS_RECT 사각형의 너비, 높이 및 위치를 설명합니다. (XPS_RECT)
XPS_SIZE 개체의 크기를 설명합니다.

열거형

XPS_COLOR_INTERPOLATION 색 보간에 사용되는 감마 함수를 설명합니다.
XPS_COLOR_TYPE XPS_COLOR 구조체에서 사용하는 색 형식에 대해 설명합니다.
XPS_DASH_CAP 파선 스트로크의 대시 캡 스타일을 지정합니다.
XPS_FILL_RULE 지정된 점이 기하 도형의 일부인지 여부를 확인하기 위해 복합 셰이프에서 사용하는 규칙입니다.
XPS_FONT_EMBEDDING 글꼴을 포함하는 옵션을 설명합니다.

XPS_IMAGE_TYPE 이미지 형식을 설명합니다.
XPS_INTERLEAVING XPS OM의 내용을 파일 또는 스트림에 쓸 때 인터리브할지 여부를 지정합니다.
XPS_LINE_CAP 선 세그먼트 대문자의 모양을 지정합니다.
XPS_LINE_JOIN 두 개의 교차 선 세그먼트에 의해 만들어진 관절을 설명합니다.
XPS_OBJECT_TYPE IXpsOMShareable에서 파생된 개체의 형식을 설명합니다.
XPS_SEGMENT_STROKE_PATTERN 그림의 모든 세그먼트, 일부 또는 없음 세그먼트가 스트로크되는지 여부를 나타냅니다.
XPS_SEGMENT_TYPE 선 세그먼트를 설명합니다.
XPS_SPREAD_METHOD 스프레드 영역을 채우는 방법을 설명합니다.
XPS_STYLE_SIMULATION 글꼴 또는 문자 모양의 시뮬레이션 스타일을 설명합니다.
XPS_THUMBNAIL_SIZE 썸네일 이미지의 크기를 설명합니다.
XPS_TILE_MODE 타일 브러시의 바둑판식 배열 동작에 대해 설명합니다.

xpsobjectmodel_1.h 헤더

아티클2023. 03. 16.

이 헤더는 XPS 문서에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- XPS 문서

xpsobjectmodel_1.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IXpsDocumentPackageTarget IXpsDocumentPackageTarget 인터페이스에는 문서 인쇄 모델에서 XPS 콘텐츠를 인쇄하는 데 필요한 요소가 포함되어 있습니다.
IXpsOMObjectFactory1 IXpsOMObjectFactory에서 상속됩니다.
IXpsOMPackage1 IXpsOMPackage에서 상속됩니다.
IXpsOMPage1 IXpsOMPage에서 상속됩니다.
IXpsOMRemoteDictionaryResource1 IXpsOMRemoteDictionaryResource를 확장하여 OpenXPS 문서를 지원하는 메서드를 제공합니다.

열거형

XPS_DOCUMENT_TYPE 문서가 serialize된 형식을 나타냅니다.

xpsobjectmodel_2.h 헤더

아티클2023. 03. 16.

이 헤더는 XPS 문서에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- XPS 문서

xpsobjectmodel_2.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IXpsDocumentPackageTarget3D 인쇄를 위해 XPS에 3D 콘텐츠를 보내는 방법을 제공합니다.
IXpsOMPackageWriter3D 모델 텍스처 및 인쇄 티켓을 지원하는 메서드를 포함합니다.

xpsprint.h 헤더

아티클2023. 03. 16.

이 헤더는 Windows GDI에서 사용됩니다. 자세한 내용은 다음을 참조하세요.

- Windows GDI

xpsprint.h에는 다음과 같은 프로그래밍 인터페이스가 포함되어 있습니다.

인터페이스

IXpsPrintJob 현재 진행 중인 인쇄 작업에 대한 액세스를 제공합니다.
IXpsPrintJobStream 애플리케이션이 인쇄 작업 데이터를 쓰는 쓰기 전용 스트림 인터페이스입니다.

Functions

StartXpsPrintJob XPS 문서 스트림을 프린터로 인쇄하기 시작합니다.
StartXpsPrintJob1 XPS 문서 콘텐츠를 프린터로 보내기 위한 인쇄 작업을 만듭니다.

구조체

XPS_JOB_STATUS 작업 상태의 스냅샷을 포함합니다.

열거형

XPS_JOB_COMPLETION 인쇄 작업의 완료 상태를 나타냅니다.

피드백

이 페이지가 도움이 되었나요? Yes No

Microsoft Q&A에서 도움말 보기