목차

[개요](#_Toc284490852)  [2](#_Toc284490852)

[Autodesk Revit API로 무엇을 할 수 있나요?](#_Toc284490853)  [2](#_Toc284490853)

[시작하기 위해 필요한 것](#_Toc284490854)  [2](#_Toc284490854)

[2 이해](#_Toc284490855)

[Autodesk Revit API](#_Toc284490856)  [3 설치](#_Toc284490856)

[개발 요구 사항](#_Toc284490857)  [3](#_Toc284490857)

[Autodesk Revit API](#_Toc284490858)  [3 사용](#_Toc284490858)

[Autodesk Revit SDK 및 온라인 도움말](#_Toc284490859)  [3](#_Toc284490859)

[문서화 규칙](#_Toc284490860)  [4](#_Toc284490860)

[Autodesk Revit API로 무엇을 할 수 있나요?](#_Toc284490861)  [4](#_Toc284490861)

[배포 옵션](#_Toc284490862)  [4](#_Toc284490862)

[1.](#_Toc284490863)  [외부 명령](#_Toc284490863)  [4](#_Toc284490863)

[2.](#_Toc284490864)  [외부 응용 프로그램](#_Toc284490864)  [4](#_Toc284490864)

[애드인 등록](#_Toc284490867)  [5](#_Toc284490867)

[외부 명령](#_Toc284490868)  [7](#_Toc284490868)

[IExternalCommand 인터페이스](#_Toc284490869)  [7](#_Toc284490869)

[매개변수](#_Toc284490870)  [7](#_Toc284490870)

[반환 값](#_Toc284490871)  [8](#_Toc284490871)

[비고](#_Toc284490872)  [8](#_Toc284490872)

[Autodesk Revit API 외부 명령](#_Toc284490873)  [8 사용](#_Toc284490873)

[외부 명령 객체 수명](#_Toc284490874)  [8](#_Toc284490874)

[외부 응용 프로그램](#_Toc284490875)  [9](#_Toc284490875)

[IExternal 애플리케이션 인터페이스](#_Toc284490876)  [9](#_Toc284490876)

[매개변수](#_Toc284490877)  [9](#_Toc284490877)

[반환 값](#_Toc284490878)  [9](#_Toc284490878)

[외부 응용 프로그램 개체 수명](#_Toc284490879)  [9](#_Toc284490879)

[64비트 Revit](#_Toc284490880)  [9 와 API 애플리케이션의 호환성](#_Toc284490880)

[9](#_Toc284490881)  [10 에서 애플리케이션 디버깅](#_Toc284490881)

[Revit 단위 시스템](#_Toc284490882)  [11](#_Toc284490882)

[11 에 대한 사용자 정의 데이터 저장 및 액세스](#_Toc284490883)

[부록 1 – Autodesk Revit 용어집](#_Toc284490884)  [12](#_Toc284490884)

[부록 2 - FAQ](#_Toc284490885)  [12](#_Toc284490885)

[일반:](#_Toc284490886)  [12](#_Toc284490886)

[Autodesk Revit 구조:](#_Toc284490887)  [14](#_Toc284490887)

# 개요

Autodesk Revit은 파워 유저와 외부 애플리케이션 개발자가 애플리케이션을 Autodesk Revit, Revit Structure, Revit Architecture 및 Revit MEP와 통합할 수 있도록 설계된 API를 제공합니다. 이러한 각 제품의 API는 매우 유사하며 Autodesk Revit 2025.3 API 또는 Autodesk Revit API로 통칭됩니다. Autodesk Revit API를 사용하기 전에 Autodesk Revit Architecture, Structure 또는 MEP와 해당 기능에 익숙해지는 것이 좋습니다. 교육은 Autodesk Developer Network(ADN)에서 찾을 수 있습니다.

# Autodesk Revit API로 무엇을 할 수 있나요?

API가 적합한 일반적인 분야는 다음과 같습니다.

* Autodesk Revit 사용자 인터페이스에서 반복적인 작업을 자동화하기 위한 애드인 만들기
* 오류를 자동으로 확인하여 프로젝트 설계 표준 적용
* 분석을 위한 프로젝트 데이터 추출 및 보고서 생성
* 새로운 요소 또는 매개변수 값을 생성하기 위해 외부 데이터 가져오기
* 분석 애플리케이션을 포함한 다른 애플리케이션을 Autodesk Revit 제품에 통합
* Autodesk Revit 프로젝트 문서를 자동으로 생성

# 시작하기 위해 필요한 것

* Autodesk Revit에 대한 실무 이해
* 소프트웨어 개발 키트를 포함한 Autodesk Revit 기반 제품 설치
* MS Visual Studio 2022 또는 MS Visual Studio 2022 Express Edition 또는 Microsoft Visual Studio 전체 설치.
* .NET 기반 개발 언어에 대한 경험(Autodesk Revit API 예제는 C# 및 Visual Basic.NET으로 제공됩니다.)
* 2일간의 자유시간

# Autodesk Revit 이해

모든 Autodesk Revit 기반제품은 매개변수적 건물 정보 모델링(BIM) 도구입니다. 이러한 도구는 개별 도면 파일 세트가 아닌 3D 모델을 구축하는 데 사용되는 CAD 프로그램으로 볼 수 있습니다. Autodesk Revit 모델링은 기둥, 벽, 문 및 창과 같은 실제 요소로 수행됩니다. 사용자는 평면도, 단면도 및 콜아웃을 포함하여 모델의 뷰를 만들 수 있습니다. 이러한 모든 뷰는 3D 물리적 모델에서 직접 생성되므로 한 뷰에서 변경한 내용은 다른 모든 뷰에 자동으로 전파됩니다. 이 프로세스는 모델에서 변경 사항이 있을 때 여러 도면과 세부 정보를 업데이트할 필요성을 사실상 없애줍니다.

Autodesk Revit API는 프로그램의 그래픽 사용자 인터페이스와 동일한 사용자 상호 작용 패러다임을 반영하도록 설계되었습니다. 따라서 API를 이해하기 위한 첫 번째 단계는 프로그램 사용 방법을 배우는 것입니다. Autodesk Revit 초보자라면 먼저 프로그램의 도움말 메뉴를 통해 액세스할 수 있는 튜토리얼을 살펴보는 것이 좋습니다. 지역 Autodesk 리셀러에서 교육 과정을 수강하는 것도 도움이 될 수 있습니다. 이렇게 하면 프로그램을 빠르게 익히는 데 도움이 됩니다.

Autodesk 리소스:

* http://www.autodesk.com/products/revit-family/overview
* http://www.autodesk.com/solutions/building-information-modeling/개요 <http://forums.autodesk.com/>

선택 = Autodesk Revit

그런 다음 = Autodesk Revit API를 선택하세요

외부 리소스:

* <http://www.augi.com/community/autodesk-revit>
* <http://www.revitcity.com>
* <http://revitinside.com>

# Autodesk Revit API 설치

Autodesk Revit API는 Autodesk Revit 기반 제품이 기본으로 설치될 때 자동으로 설치됩니다.

Autodesk Revit API 소프트웨어 개발 키트는 Autodesk Revit 설치 DVD의 도구 및 유틸리티 섹션에서 설치됩니다.

# 개발 요구 사항

Autodesk Revit API에는 Microsoft .NET 8이 필요합니다. API 애플리케이션을 편집하고 디버깅하려면 Microsoft Visual Studio 2022 또는 C# 또는 Visual Basic.NET용 MS Visual Studio Express Edition 중 하나와 같은 대화형 개발 환경이 필요합니다. Autodesk Revit API로 개발할 때는 프로젝트가 Autodesk Revit Program 디렉터리에 포함된 **RevitAPI.dll** 및 **RevitAPIUI.dll이라는 두 개의 DLL을 참조하는지 확인하십시오.**

API를 효과적으로 사용하려면 약간의 프로그래밍 기술이 필요합니다. 프로그래밍 초보자라면 Microsoft Visual Studio 2022와 C# 또는 Visual Basic.NET과 같은 호환 언어 중 하나를 배우는 것이 좋습니다. 시작하기에 좋은 책과 수업이 많이 있습니다.

**자원:**

온라인 리소스

* 무료 Visual Studio – <http://msdn.microsoft.com/VStudio/Express/>
* <http://www.codeguru.com/>
* <http://devx.com/>
* <http://www.msdn.microsoft.com/>

서적:

* 스티브 맥코넬의 Code Complete, 2판
* 소프트웨어 프로젝트 생존 가이드, 저자: Steve McConnell

# Autodesk Revit API 사용

## Autodesk Revit SDK 및 온라인 도움말

ADN 또는 설치 DVD에서 Autodesk Revit SDK를 받으세요. Autodesk Revit 2025.3 SDK는 설치 DVD의 도구 및 유틸리티 옵션에서 압축을 풀고 설치할 수 있습니다. SDK에는 API와 그 사용법을 더 잘 이해하는 데 도움이 되는 예제 파일이 있습니다. 각 예제 파일에는 Autodesk Revit Program 폴더에 있는 revit.ini 파일에 편집하여 넣어야 하는 정보가 있는 샘플 revit.ini 파일이 있으며, Autodesk Revit은 시작 시 이 파일에 액세스합니다.

* 로컬 드라이브에 SDK 압축 해제
* SDK에서 Read Me First.doc를 읽어보세요.

RevitAPI.chm 은 SDK 패키지의 \Revit 2025.3 SDK\ 폴더에 포함된 Autodesk Revit API 참조 문서 도움말 파일입니다.

**자원:**

[www.autodesk.com/adn](http://www.autodesk.com/adn) - Autodesk Developer Network 홈(ADN)

[www.autodesk.com/developrevit](http://www.autodesk.com/developrevit) - Autodesk Revit 개발 리소스

<http://discussion.autodesk.com>– Autodesk 제품 토론 그룹

선택 = Autodesk Revit Architecture

그런 다음 = Autodesk Revit API를 선택하세요

<http://thebuildingcoder.typepad.com/blog/>- Revit 코딩에 전념하는 AND 블로그인 The Building Coder

<http://bimapps.typepad.com/>- BIM 애플리케이션에 전념하는 블로그인 Bim Apps

### 문서화 규칙

**Autodesk.Revit.DB.Element** 와 같은 네임스페이스 형식의 이름이 포함됩니다.

## Autodesk Revit API로 무엇을 할 수 있나요?

### 배포 옵션

Autodesk Revit API는 진행 중 DLL만 지원합니다. 즉, API 애플리케이션은 Autodesk Revit 프로세스에 로드된 DLL로 컴파일됩니다.

Autodesk Revit API는 단일 스레드 액세스만 지원합니다. 즉, API 애플리케이션은 모든 Autodesk Revit API 호출을 메인 스레드(다양한 API 진입점에서 Autodesk Revit 프로세스가 호출)에서 수행해야 하며, API 애플리케이션은 다른 스레드에서 작업을 유지 관리하고 언제든지 Autodesk Revit에 호출할 수 있을 것으로 기대할 수 없습니다.

Autodesk Revit API를 사용하여 만들 수 있는 DLL에는 두 가지 유형이 있습니다.

### 외부 명령

Autodesk Revit API를 사용하면 Autodesk Revit의 사용자 인터페이스에 새 명령을 추가할 수 있습니다. 이러한 명령은 그림 1에서 볼 수 있듯이 '외부 도구' 풀다운 아래의 애드인 탭에 나타납니다. API를 통해 외부 도구 명령은 Autodesk Revit 데이터베이스와 현재 선택된 요소에 액세스할 수 있습니다.

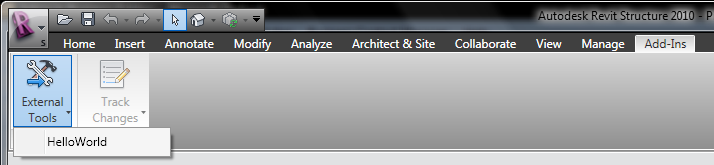


그림 1 - Revit에 추가된 외부 도구

### 외부 응용 프로그램

Autodesk Revit API를 사용하면 외부 애플리케이션을 추가할 수도 있습니다. 이러한 애플리케이션은 Autodesk Revit 시작 및 종료 중에 호출됩니다. 그림 2에서 볼 수 있듯이 Add-ins 탭에서 새 패널을 만들 수 있습니다. 또한 Autodesk Revit 사용자 인터페이스에서 발생하는 이벤트에 반응할 수 있는 핸들러를 등록할 수도 있습니다.

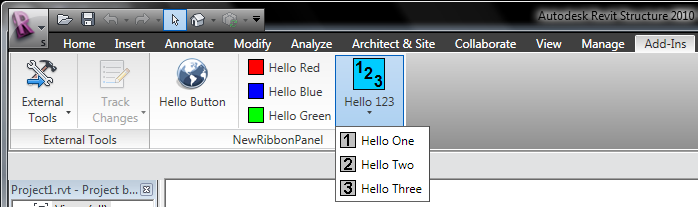


그림 2 - Revit에 추가된 새로운 패널 및 컨트롤

## 애드인 등록

.addin 매니페스트 파일을 통해 API 애플리케이션을 등록하는 기능을 제공합니다 .

매니페스트 파일은 사용자 시스템의 두 위치 중 하나에 배치되면 Revit에서 자동으로 읽힙니다.

* **"애플리케이션 데이터"의 사용자 특정 위치가 아닌 위치**
  + C:\ProgramData\Autodesk\Revit\Addins\2025\
* **"애플리케이션 데이터"의 사용자 특정 위치에서**
  + C:\Users\<사용자>\AppData\Roaming\Autodesk\Revit\Addins\2025\

.addin 이라는 이름의 모든 파일은 Revit이 시작될 때 읽고 처리합니다.

ExternalCommand 하나를 추가하는 기본 파일은 다음과 같습니다.

<? xml> 버전 = " 1.0 " 인코딩 = " utf-16 " standalone = " 아니오 " ?>

< RevitAddIns >

< 추가 기능 유형 = " 명령 " >

< 어셈블리 > c:\MyProgram\MyProgram.dll </ 어셈블리 >

< AddInId > 76eb700a-2c85-4888-a78d-31429ecae9ed </ AddInId >

<FullClassName> Revit.Samples.SampleCommand </FullClassName>​​​​

< 텍스트 > 샘플 명령 </ 텍스트 >

< VisibilityMode > NotVisibleInFamily </ VisibilityMode >

< 가시성 모드 > MEP에서 표시되지 않음 </ 가시성 모드 >

< AvailabilityClassName > Revit.Samples.SampleAccessibilityCheck </ AvailabilityClassName >

</ 추가 기능 >

</ RevitAddIns >

ExternalApplication을 추가하는 기본 파일은 다음과 같습니다.

<? xml> 버전 = " 1.0 " 인코딩 = " utf-16 " standalone = " 아니오 " ?>

< RevitAddIns >

< 추가 기능 유형 = " 응용 프로그램 " >

< 이름 > 내 샘플 애플리케이션 </ 이름 >

< 어셈블리 > c:\MyProgram\MyProgram.dll </ 어셈블리 >

< AddInId > 604B1052-F742-4951-8576-C261D1993107 </ AddInId >

<FullClassName> Revit.Samples.SampleApplication </FullClassName>​​​​

</ 추가 기능 >

</ RevitAddIns >

여러 개의 AddIn 요소를 하나의 매니페스트 파일에 제공할 수 있습니다.

새로운 메커니즘은 현재 다음과 같은 XML 태그를 제공합니다.

|  |  |
| --- | --- |
| **꼬리표** | **설명** |
| 집회 | 추가 기능 어셈블리 파일에 대한 전체 경로입니다. 모든 ExternalCommands 및 ExternalApplications 에 필요합니다 . |
| 전체 클래스 이름 | IExternalCommand 또는 IExternalApplication을 구현하는 어셈블리 파일의 클래스 전체 이름입니다 . 모든 ExternalCommands 및 ExternalApplications 에 필요합니다 . |
| 추가ID | 특정 애플리케이션 의 ID를 나타내는 GUID . AddInId는 Revit의 지정된 세션에 대해 고유해야 합니다. Autodesk는 등록된 각 애플리케이션 또는 명령에 대해 고유한 GUID를 생성할 것을 권장합니다. 모든 ExternalCommands 및 ExternalApplications 에 필요합니다. UIApplication.ActiveAddInId 속성은 필요한 경우 이 값에 대한 프로그래밍 방식 액세스를 제공합니다. |
| 이름 | 애플리케이션 이름. 필수. ExternalApplications 에만 해당. |
| 텍스트 | 버튼의 이름입니다. 선택 사항입니다. 이 태그는 ExternalCommands 에만 사용합니다 . 기본값은 "External Tool"입니다. |
| 설명 | 명령에 대한 간략한 설명은 버튼 툴팁으로 사용됩니다. 선택 사항입니다. 이 태그는 ExternalCommands 에만 사용합니다 . 기본값은 명령 텍스트만 있는 툴팁입니다. |
| 가시성모드 | 프로젝트 문서, 패밀리 문서 또는 전혀 문서에 표시되지 않는 명령을 지정할 수 있는 기능을 제공합니다. 또한 명령을 표시할 분야를 지정할 수 있는 기능도 제공합니다. 이 옵션에 여러 값을 설정할 수 있습니다. 선택 사항입니다. 이 태그는 ExternalCommands 에만 사용합니다 . 기본값은 **활성 문서가 없는 경우를 포함하여 모든 모드와 분야에서 명령을 표시하는 것입니다. 활성 문서에 대해 실행해야 하는 이전에 작성된 외부 명령은 활성** 문서가 없는 경우 명령 호출을 처리하도록 코드를 수정하거나 NotVisibleWhenNoActiveDocument 모드를 적용해야 합니다 . |
| 가용성클래스이름 | IExternalCommandAvailability를 구현한 어셈블리 파일의 클래스 전체 이름입니다 . 이 클래스를 사용하면 컨텍스트에 따라 명령 단추를 선택적으로 회색으로 표시할 수 있습니다. 선택 사항입니다. 이 태그는 ExternalCommands 에만 사용합니다 . 기본값은 표시될 때마다 사용할 수 있는 명령입니다. |
| 큰이미지 | 외부 도구 풀다운 메뉴에서 버튼에 사용할 아이콘 경로입니다. 최상의 결과를 얻으려면 아이콘이 32 x 32픽셀이어야 합니다. 선택 사항입니다. 이 태그는 ExternalCommands 에만 사용합니다 . 기본값은 아이콘 없이 버튼을 표시하는 것입니다. |
| 긴 설명 | 명령에 대한 긴 설명은 버튼의 확장된 툴팁의 일부로 사용됩니다. 이 툴팁은 마우스가 명령 위에 장시간 올려져 있을 때 표시됩니다. 각 문단 주위에 <p> 태그를 배치하여 이 옵션의 텍스트를 여러 문단으로 나눌 수 있습니다. 선택 사항입니다. 이 태그는 ExternalCommands 에만 사용합니다. 이 속성과 TooltipImage가 제공되지 않으면 버튼에 확장된 툴팁이 없습니다. |
| 툴팁이미지 | 마우스가 명령 위에 장시간 올려져 있을 때 표시되는 버튼 확장 툴팁의 일부로 표시할 이미지 파일의 경로입니다. 선택 사항입니다. 이 태그는 ExternalCommands 에만 사용합니다 . 이 속성과 TooltipImage가 제공되지 않으면 버튼에 확장 툴팁이 없습니다. |
| 언어 유형 | Text, Description, LargeImage , LongDescription , TooltipImage 에 대한 현지화 설정 입니다. Revit은 지정된 언어 리소스 dll 에서 리소스 값을 로드합니다 . 값은 Revit에서 지원하는 11개 언어 중 하나일 수 있습니다. LanguageType을 지정하지 않으면 Revit의 현재 세션에서 사용하는 언어 리소스가 자동으로 로드됩니다. |

# 외부 명령

* 이 경우 Autodesk.Revit.UI.IExternalCommand와 같은 특정 인터페이스에 대한 지원을 구현합니다 .
* 외부 명령을 설명하는 하나 이상의 항목을 Revit.ini 파일에 추가합니다.

Autodesk.Revit.UI.IExternalCommand 인터페이스를 지원하는 노출된 .NET 객체입니다 . 또한 Revit.ini 파일에는 Revit 이 명령을 "보고" 사용할 수 있도록 각 객체에 대한 항목이 하나씩 있어야 합니다 .

## IExternalCommand 인터페이스​

인터페이스의 선언(VB.NET)은 다음과 같습니다.

*함수 실행( ByVal) commandData Autodesk.Revit.UI.ExternalCommandData 와 같이*

*ByRef 메시지를 문자열로,*

*ByVal 요소 As Autodesk.Revit.DB.ElementSet )*

*결과적으로*

### 매개변수

* commandData : 이 매개변수에 전달된 객체는 실행 중인 명령에 중요한 정보를 포함합니다 . d . 이 데이터에는 Autodesk Revit Application 객체와 현재 활성화된 뷰가 포함됩니다.
* message : 메시지 문자열은 명령이 종료될 때 사용자에게 특정 메시지를 제공하도록 설정할 수 있습니다. 이 메시지가 표시되는 방식은 함수의 반환 값에 따라 달라집니다. 자세한 내용은 설명 섹션을 참조하세요.
* elements : 처음에는 Autodesk Revit 요소를 포함할 수 있는 빈 세트입니다. 명령이 종료되면 반환 값에 따라 이 세트 내의 요소가 표시될 수 있습니다. 자세한 내용은 설명 섹션을 참조하세요.

### 반환 값

결과 : 반환 값은 다음 중 하나일 수 있습니다.

* 성공 : 명령이 처리되지 않은 오류 조건 없이 예상대로 성공한 경우 반환됩니다. 외부 명령은 Autodesk Revit 사용자 인터페이스에서 실행 취소할 수 없는 작업으로 나타납니다.
* 취소됨 : 이 값은 사용자가 명령을 취소하도록 요청했음을 지정합니다. 외부 명령 실행 중에 Autodesk Revit 객체에 적용된 모든 변경 사항은 취소됩니다. 메시지를 게시할 수 있습니다. 설명 섹션을 참조하세요.
* 실패 : 실패는 외부 명령이 복구할 수 없는 방식으로 실패했음을 의미합니다. 외부 명령 실행 중에 Autodesk Revit 객체에 대한 모든 변경 사항은 취소됩니다. 메시지가 게시됩니다. 설명 섹션을 참조하세요.

### 비고

메시지 및 요소 매개변수는 명령이 취소되거나 실패한 경우에 사용됩니다.

* 취소됨 : 외부 명령이 취소되었고 메시지 매개변수가 외부 명령에 의해 설정된 경우 실행이 Autodesk Revit으로 다시 반환될 때 메시지가 표시됩니다 . 메시지 매개변수가 설정 되지 않은 경우 메시지가 표시되지 않고 명령이 자동으로 종료됩니다.
* 실패 : 외부 명령이 실패하면 메시지 매개변수의 내용이 표시됩니다. 요소 집합에 Autodesk Revit 요소가 포함되어 있으면 오류 메시지가 표시될 때 이러한 요소가 강조 표시되어 개발자가 사용자에게 문제 요소를 보여줄 수 있습니다.

## Autodesk Revit API 외부 명령 사용

1. 사용자가 Autodesk Revit에서 프로젝트를 열거나 만듭니다.
2. 사용자는 추가 기능 탭의 외부 도구 풀다운에서 외부 명령을 선택합니다.
3. 여러 Autodesk Revit 요소를 선택할 수 있는 옵션이 있었습니다 . 그렇게 하면 프로그램은 선택된 멤버에서만 기능을 수행하기로 결정할 수 있습니다.
4. API 프로그램은 Autodesk Revit에서 초점을 가져와 필요한 작업을 수행합니다. 종종 애플리케이션이 작업을 완료하기 전에 사용자 입력을 얻기 위해 대화 상자가 필요할 수 있습니다.
5. 추가 기능 도구가 기능을 완료하거나 사용자가 해제하면 프로그램은 필요에 따라 Autodesk Revit 모델을 업데이트하고 외부 명령에서 복귀하여 포커스를 다시 Autodesk Revit에 돌려줍니다.

|  |  |
| --- | --- |
|  |  |
|  |  |

## 외부 명령 객체 수명

Autodesk Revit에서 다른 명령이나 편집 모드가 활성화되지 않은 경우 등록된 외부 명령이 활성화됩니다. 선택하면 명령 객체가 생성되고 Execute 메서드가 호출됩니다. 이 메서드가 Autodesk Revit으로 돌아오면 명령 객체가 파괴됩니다. 이러한 파괴로 인해 명령 실행 사이에 객체 내에서 데이터가 유지될 수 없습니다. 데이터를 유지하려면 외부 파일이나 데이터베이스를 사용할 수 있습니다. Autodesk Revit 프로젝트 내에서 데이터를 유지하려면 공유 매개변수 메커니즘을 사용하여 이 데이터를 저장할 수 있습니다.

# 외부 응용 프로그램

* 이 경우 Autodesk.Revit.UI.IExternal Application 과 같은 특정 인터페이스에 대한 지원을 구현합니다 .
* 응용프로그램을 설명하는 하나 이상의 항목을 Revit.ini 파일에 추가합니다 .

기술적으로 외부 애플리케이션은 Autodesk.Revit.IExternal Application 인터페이스를 지원하는 노출된 .NET 객체입니다 . 또한 Revit.ini 파일에는 Autodesk Revit이 시작될 때 이러한 애플리케이션을 로드 할 수 있도록 각 객체에 대한 항목이 하나씩 있어야 합니다 .

## IExternal 애플리케이션 인터페이스​

인터페이스의 선언( C# )은 다음과 같습니다.

* Autodesk.Revit.UI.IExternalApplication.Result OnStartup ( Autodesk.Revit.ControlledApplication 응용 프로그램)
* Autodesk.Revit.UI.IExternalApplication.Result OnShutdown ( Autodesk.Revit.ControlledApplication 응용 프로그램)

### 매개변수

* 애플리케이션 : 이 매개변수에 전달된 객체에는 호출되는 명령 s OnStartup 및 OnShutdown 에 중요한 정보가 포함되어 있습니다 . 이 객체는 VersionName , VersionNumber와 같은 Autodesk Revit Application의 제한된 액세스 방법 과 OnDocumentOpened , OnDocumentSaved 와 같은 일부 이벤트에 대한 대리자를 제공합니다 .

### 반환 값

결과 : 반환 값은 다음 중 하나일 수 있습니다.

* 성공 : 외부 애플리케이션이 처리되지 않은 오류 조건 없이 예상대로 성공한 경우 반환됩니다 .
* 실패 : 실패는 외부 애플리케이션이 복구할 수 없는 방식으로 실패했음을 의미합니다.
* 취소됨 : 이 값은 외부 애플리케이션이 취소됨을 지정합니다.

## 외부 애플리케이션 객체 수명

Autodesk Revit이 시작 되면 외부 애플리케이션 객체 가 생성되고 OnStartup 메서드가 호출됩니다. 이 메서드가 다시 돌아 오면 성공적으로 Autodesk Revit에 외부 애플리케이션 객체가 Autodesk Revit 세션 전체 동안 유지 됩니다 . Autodesk Revit이 종료되면 OnShutdown 메서드가 호출 됩니다 .

# 64비트 Revit과 API 애플리케이션의 호환성

대부분의 API 애플리케이션은 32비트 Revit 또는 64비트 Revit에서 사용할 수 있는 설정으로 빌드할 수 있습니다. 다음 설정을 사용하여 이러한 방식으로 프로젝트를 빌드하세요.

* C# 프로젝트의 경우 프로젝트 속성->빌드 태그로 가서 플랫폼 대상을 "모든 CPU"로 선택합니다.
* VB.Net 프로젝트의 경우 프로젝트 속성->컴파일 태그로 가서 플랫폼을 "모든 CPU"로 선택합니다.
* C++/CLI 프로젝트의 경우 속성 페이지를 열고 구성 속성->일반->공용 언어 런타임 지원으로 이동한 후 / clr:safe를 선택합니다 .

일부 Microsoft 구성 요소(예: Microsoft Jet용 OleDB 공급자)가 있다는 점에 유의하세요. 모든 SDK 샘플은 지원되지 않는 Microsoft 구성 요소에 대한 종속성이 있는 샘플을 제외하고 32비트 또는 64비트 Revit에서 작동하도록 설정되어 있습니다.

**자세한 내용은 이 Microsoft 리소스를 참조하세요.**

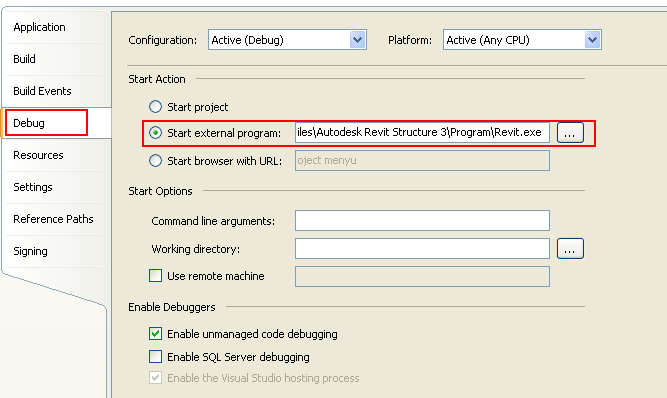
<http://msdn.microsoft.com/en-us/library/ms241064(VS.80).aspx>

# 1 9 에서 애플리케이션 디버깅

독립 실행형 애플리케이션(EXE)과 이를 실행하기 위해 다른 프로그램이 필요한 외부 애플리케이션(DLL)을 디버깅하는 데는 몇 가지 차이점이 있습니다. Autodesk Revit API를 사용하는 애플리케이션을 디버깅하려면 Autodesk Revit에서 활성화해야 합니다. 디버깅을 위해 개발자 환경에서 이를 수행하려면 다음이 필요합니다.

|  |  |
| --- | --- |
| 1. API 애플리케이션에 대한 Visual Studio 프로젝트를 엽니다 . ( 예 : 샘플 폴더의 AnalyticalSupportData\_Info.csproj ) 2. 프로젝트 메뉴에서 AnalyticalSupportData\_Info 속성을 선택합니다. |  |

1. 왼쪽의 디버그 탭을 선택하세요
2. "외부 프로그램 시작:" 라디오 버튼을 선택하세요
3. 찾아보기 버튼을 눌러 Revit.exe 파일을 찾은 후 열기를 누릅니다.



1. 소스 코드에 중단점을 설정하세요.
2. Visual Studio에서 "디버깅 시작"을 선택합니다. Autodesk Revit이 실행됩니다.
3. 중단 점을 치려면 외부 도구 메뉴에서 프로그램에 대한 옵션을 선택합니다. 컴파일러가 중단점 중 하나에 도달하면 프로그램을 디버깅할 수 있도록 중단됩니다.

# Revit 단위 시스템

Revit 단위 시스템은 다음과 같은 기본 단위를 사용합니다.

|  |  |  |
| --- | --- | --- |
| **기본 유닛** | **Revit에서의 단위** | **단위 시스템** |
| 길이 | 피트(ft) | 장엄한 |
| 각도 | 라디안 | 미터법 |
| 대량의 | 킬로그램(kg) | 미터법 |
| 시간 | 초(s) | 미터법 |
| 전류 | 암페어(A) | 미터법 |
| 온도 | 켈빈(K) | 미터법 |
| 광도 | 칸델라(cd) | 미터법 |

**참고** : Revit은 길이를 피트로 저장하고 다른 수량을 미터법으로 저장하기 때문에 길이를 포함하는 파생 단위는 영국식과 미터법을 사용하는 비표준 단위를 사용합니다. 예를 들어, 힘은 제곱 시간당 질량-길이로 측정되고 kg-ft/s 2 로 저장됩니다 .

# 애플리케이션을 위한 사용자 정의 데이터 저장 및 액세스

종종 Autodesk Revit에 연결된 프로그램은 Autodesk Revit 모델 데이터베이스에서 사용할 수 없는 정보를 요구합니다. 사용자가 이러한 추가 정보를 입력하는 방법에는 여러 가지 가 있습니다. 정보를 입력하는 방법과 위치는 용도에 따라 달라집니다.

* 정보가 일반적인 유형이고 사용자가 Autodesk Revit 내에서 해당 정보를 보고 편집하려는 경우, 해당 정보는 표시되는 프로젝트 또는 공유 매개변수로 저장해야 합니다.
* 정보가 Autodesk Revit 모델과 함께 유지되어야 하지만 표시할 필요는 없는 경우, 해당 정보를 Autodesk Revit 모델에 표시되지 않는 프로젝트나 공유 매개변수로 저장할 수 있습니다.
* 정보가 단일 애드온 프로그램에만 국한되고, 변경 대상인 다양한 건축 제품의 사양과 같이 Autodesk Revit 모델에 실제로 저장하기에는 너무 큰 경우, 최상의 솔루션은 프로그램별 정보를 저장하는 동시 모델 데이터베이스를 만드는 것일 수 있습니다. 이 경우 각 요소에 대한 요소 UniqueId 속성을 데이터베이스의 키로 사용하는 것이 유용할 수 있습니다. 요소의 UniqueId는 모델 내에서 안정적이기 때문입니다.

## 

## 부록 1 – Autodesk Revit 용어집

###### 가족

패밀리는 유형이라고 하는 객체의 모음입니다. 패밀리는 공통 매개변수 집합, 동일한 용도 및 유사한 그래픽 표현을 가진 요소를 그룹화합니다. 패밀리 내의 다른 유형은 일부 또는 모든 매개변수의 값이 다를 수 있지만 매개변수 집합(이름과 의미)은 동일합니다.

###### 매개변수

여러 가지 Autodesk Revit 매개변수 유형이 있습니다 . **공유 매개변수는** 사용자 정의 변수로 간주할 수 있습니다. **시스템 매개변수** 는 Autodesk Revit에 하드 코딩된 변수입니다. **패밀리 매개변수** 는 패밀리가 생성되거나 수정될 때 정의되는 변수입니다.

###### 유형

유형은 패밀리의 멤버입니다. 각 유형에는 모델에 존재하는 모든 유형의 인스턴스에 대해 일정한 특정 매개변수가 있습니다. 이를 유형 속성이라고 합니다. 유형에는 인스턴스 매개변수라는 다른 매개변수가 있으며, 이는 모델에서 이 유형의 다양한 인스턴스에 따라 달라질 수 있습니다.

## 부록 2 - FAQ

### 일반적인:

**질문:** Autodesk Revit에서 요소를 참조하려면 어떻게 해야 하나요?

**답변:** 각 요소에는 요소를 참조하는 데 사용할 수 있는 Id 및   
UniqueId 속성이 있습니다.

**질문:** 요소 ID는 보편적으로 고유한가요? 그리고 변경될 수 있나요?

**A:** 주어진 시간에 Revit 모델의 요소 ID는 고유합니다. 그러나 Revit에서 여러 사용자 상호 작용으로 인해 요소의 ID가 변경될 수 있습니다. 또한 다른 모델의 다른 요소에 동일한 ID가 있는 요소가 있을 수 있습니다. 나중에 요소를 찾기 위해 ID(특히 새로 생성된 요소의 ID)를 저장해야 하는 경우 UniqueId 를 저장합니다 .

**질문:** 모델은 공유 매개변수 파일을 하나만 사용할 수 있나요?

**A:** 공유 매개변수 파일은 매개변수 사양을 저장하는 데 사용됩니다. 이러한 매개변수 설명의 가장 중요한 부분은 단일 모델 파일과 여러 모델에서 매개변수의 고유성을 보장하는 데 사용되는 GUID입니다.

Autodesk Revit은 여러 공유 매개변수 파일을 사용할 수 있지만 한 번에 한 파일에서만 매개변수를 읽을 수 있습니다. 그런 다음 모든 모델에 대해 동일한 공유 매개변수 파일을 선택하거나 각 모델에 대해 다른 파일을 선택할지 여부는 사용자에게 달려 있습니다.

API 애플리케이션은 사용자의 매개변수 파일을 방해하지 않아야 합니다. 애플리케이션에는 필요한 매개변수가 포함된 매개변수 파일이 함께 제공되어야 합니다. 매개변수를 Autodesk Revit 파일에 로드하려면 애플리케이션은 다음을 수행해야 합니다.

* 사용자의 매개변수 파일 이름을 저장합니다.
* 응용프로그램 매개변수 파일을 최신 상태로 만듭니다.
* 필요한 매개변수를 로드하고,
* 사용자의 매개변수 파일을 최신으로 만듭니다.

**질문:** 다른 프로그램에서 공유 매개변수를 사용할 수 있도록 모델과 함께 공유 매개변수 파일을 배포해야 합니까?

**A:** 아니요, 공유 매개변수 파일을 배포할 필요가 없습니다. 공유 매개변수 파일은 공유 매개변수를 로드하는 데만 사용됩니다. 로드된 후에는 해당 모델에 파일이 더 이상 필요하지 않습니다.

**질문:** 해당 요소를 복사하면 공유된 매개변수 값도 복사됩니까?

**A:** 그렇습니다. 따라서 데이터베이스의 항목에 대한 고유 ID를 보유하는 공유 매개변수가 있는 경우 이 ID에 Autodesk Revit 요소의 UniqueId 속성을 포함하는 것이 좋습니다 . 이렇게 하면 애플리케이션의 기존 요소에서 복사된 요소를 식별할 수 있습니다.

**질문:** Autodesk Revit은 내 애플리케이션이 데이터를 모델로 다시 보낼 때 업데이트하는 데 시간이 오래 걸립니다. 속도를 높이려면 어떻게 해야 합니까?

**A:** API 프레임워크가 모든 모델 레벨 변경 후에 재생성되는 것을 방지하려면 Manual 재생성 모드를 사용해야 합니다. 그런 다음 재생성 API를 사용하여 변경 그룹 후에 문서를 강제로 업데이트합니다. Revit 문서의 여러 수정 성능은 RegenerationOption.Automatic 보다 빨라야 합니다 . 이 모드는 문서에 대한 모든 업데이트를 중단하기 때문에 애플리케이션은 문서가 재생성될 때까지 수정된 문서에서 데이터를 읽어서는 안 되며, 그렇지 않으면 오래된 데이터에 액세스할 위험이 있습니다.

**질문:** 공유 매개변수를 바인딩할 수 없는 요소에 공유 매개변수를 추가하려면 어떻게 해야 합니까? 예: 그리드 또는 머티리얼.

**A:** 요소 유형에 공유 매개변수를 포함할 수 있는 기능이 없는 경우 프로젝트 매개변수를 추가해야 합니다. 이렇게 하면 이러한 요소의 매개변수 값에 액세스하기 위해 다른 코드가 필요하므로 약간 더 복잡해집니다.

**질문:** 그룹이 포함된 모델을 외부 프로그램으로 내보낼 때 사용자는 내보내기가 끝날 때 오류를 받습니다.

"그룹 "그룹 1"에 대한 변경은 그룹 편집 모드에서만 허용됩니다. Edit Group 명령을 사용하여 그룹의 모든 인스턴스에 변경을 적용합니다. "Ungroup" 옵션을 사용하여 변경된 그룹 인스턴스를 그룹 해제하여 이 변경을 진행할 수 있습니다."

**A:** 현재 API는 그룹 멤버에 대한 변경을 허용하지 않습니다. 프로그래밍 방식으로 그룹을 해제하고, 변경하고, 다시 그룹화한 다음 이전 그룹의 다른 인스턴스를 새 그룹으로 바꿔서 동일한 효과를 얻을 수 있습니다.

**질문:** Autodesk Revit 문서에서 두 개의 서로 다른 FamilySymbol이 같은 이름을 가지고 있는 이유는 무엇입니까 ?

**에이:** Autodesk Revit API에서 기호는 형상을 나타냅니다.

동일한 유형의 두 창 인스턴스 가 서로 다른 유형의 벽 에 삽입 되었다고 가정해 보겠습니다 . 이제 이 두 창 인스턴스는 동일한 FamilySymbol을 참조 하고 동일한 유형으로 간주되지만, 두께가 다른 벽에 삽입하면 각 창의 지오메트리가 일치하도록 변경되므로 동일한 심볼의 지오메트리를 사용할 수 없다는 것은 분명합니다 . FamilySymbol 의 개별 인스턴스를 추적하기 위해 Autodesk Revit은 "기본" 및 "하위" 기능을 사용하며 "하위" FamilySymbol은 "기본" FamilySymbol 템플릿 에 따라 지오메트리를 주변 조건에 맞게 지역화합니다 .

모델에 문 인스턴스가 없는 경우 36"x84" 문에 대한 " 기본" 패밀리 심볼 은 하나뿐이지만 문이 삽입되면 Autodesk Revit은 새 36"x84" 문에 대한 "하위" 패밀리 심볼을 만듭니다 .

모델에 후속적으로 문을 배치하면 Autodesk Revit에서 추가적인 "하위" FamilySymbols가 생성됩니다 .

**큐:** Autodesk Revit 지오메트리 메시 객체 에서 정점 또는 삼각형의 순서가 시계 반대 방향 으로 안정적으로 정렬되어 있습니까 ?

**대답:** 네.

**질문:** 벽이나 문(외부와 내부 모두)을 뒤집은 후에 FromRoom 과 ToRoom 속성이 변경되지 않는 이유는 무엇입니까 ?

**에이:** 외부 문 및 외부 창에는 FromRoom 또는 ToRoom 만 있습니다 . 사용자는 문이나 창을 찾을 때 어떤 속성이 있는지 결정할 수 있습니다. 외부 문이나 창을 뒤집으면 FromRoom 또는 ToRoom 속성이 자동으로 변경되지 않지만 수동으로 변경할 수 있습니다.

### Autodesk Revit 구조:

**질문:** 때로는 구조 요소의 기본 끝 릴리스로 인해 모델이 불안정해지는 경우가 있습니다.

**A:** 어떤 경우 에는 사용자가 요소를 모델링할 때 요소 끝 릴리스에 주의를 기울이지 않았을 수 있습니다. 분석 모델 확인 기능은 이러한 문제 중 일부를 찾아냅니다. 분석 모델을 내보낼 때 사용자에게 RST에서 릴리스 조건을 유지할지 아니면 모든 보와 기둥을 고정 할 것인지 묻습니다. 모델을 RST로 다시 가져올 때는 항상 끝 릴리스를 업데이트하고 이후 분석 프로그램으로 내보낼 때 끝 릴리스를 덮어쓰지 마십시오 .

**질문:** 빔 의 방향을 회전시켜 약한 방향으로 회전시키고 있습니다. 즉, W14X30의 I가 90도 회전하여 H처럼 보이도록 회전합니다. API에서 이 회전 각도에 어떻게 액세스합니까?   
위치가 LocationPoint 가 아니라 LocationCurve 이기 때문에 Rotation 값에 액세스할 수 없으므로 무엇을 확인해야 합니까? 확인해야 할 FamilyInstance 요소 가 있는데 어떻게 해야 합니까?

**A:** SDK의 RotateFramingObject 예제 를 살펴보세요 . 여기에는 보, 브레이스 및 기둥의 회전 각도를 가져오고 변경하는 방법에 대한 예가 있습니다.

**질문:** 모델에 새로운 콘크리트 보와 기둥 크기를 추가하려면 어떻게 해야 하나요?

**답변:** SDK에서 FrameBuilder 샘플 코드 를 살펴보세요 .

|  |  |
| --- | --- |
| **질문:** 실제 데크 레이어를 보려면 어떻게 해야 하나요?  **A:** DeckProperties 라는 예제가 있는데 , 데크의 레이어 정보를 얻는 방법에 대한 정보를 제공합니다. 데크 정보는 UI와 정확히 같은 방식으로 보고됩니다. 데크 차원 매개변수는 다음 그림에 나와 있습니다. |  |

**질문:** 캔틸레버가 있는 보를 어떻게 알 수 있나요?

**A:** Autodesk Revit 데이터베이스에는 보에 캔틸레버가 있는지 여부를 알 수 있는 직접적인 방법이 없습니다. 그러나 아래 옵션 중 하나 이상을 사용하면 보가 캔틸레버인지 여부를 잘 추측할 수 있습니다.

옵션

1. Moment Connection Start와 Moment Connection End라는 두 가지 매개변수가 있습니다. 이 두 가지에 설정된 값이 None이 아니면 동일 선상에 있고 None이 아닌 다른 값으로 설정된 보가 있는지 확인해야 합니다 . 사용자에게 Moment Frame 옵션 대신 Cantilever Moment 옵션을 선택하도록 요청할 수도 있습니다.
2. 하나 또는 두 개의 요소를 넘어 연결성을 추적해 보세요.
3. 원소 방출 조건을 살펴보세요.

**질문:** 벽 아래에 돌출부가 있는 기초를 어떻게 모델링합니까?

**A:** Autodesk Revit Structure의 벽 기초 기능은 끝부분에 돌출부를 지원하지 않습니다. 가장 좋은 해결책은 GUI 또는 API에서 기초 슬래브 기능을 사용하는 것입니다. Autodesk.Revit.Creation.Document.NewFloor () 메서드를 사용하여 만들 수 있습니다. STRUCTURAL\_FLOOR\_ANALYZES\_AS의 경우. 기초에는 AA\_Mat 또는 AA\_SlabOnGrade를 사용하고 바닥인 경우 AA\_Slab을 사용해야 합니다 .