

DWD AI Assistant Help

Release Notes V1-5-0

DWD AI Assistant: Revolutionize Your Design Workflow with AI-Powered Assistance

Transform your Autodesk Revit experience with DWD AI Assistant, the ultimate add-in that integrates AI directly into your Revit environment. With DWD AI Assistant, you can seamlessly chat with an intelligent assistant that understands and interacts with your Revit models using the Revit API.

Key Features:

- **Intuitive Chat Interface:** Easily access the chat box within Revit to ask questions, get design suggestions, and receive real-time assistance.
- **Enhanced Productivity:** Automate repetitive tasks, streamline your workflow, and focus on creativity while DWD AI Assistant handles the technical details.
- **Smart Model Interaction:** Use natural language commands to modify, analyse, and optimize your Revit models, making complex operations simpler and faster.
- **Seamless Integration:** Enjoy a smooth and intuitive integration with your existing Revit projects, with no need for extensive setup or training.

Why Choose DWD AI Assistant?

- **Boost Efficiency:** Save time and reduce errors by leveraging AI to perform tasks that would otherwise require manual effort.
- **Stay Ahead:** Keep up with the latest advancements in AI and design technology, ensuring your projects are cutting-edge.
- **User-Friendly:** Designed with architects, engineers, and designers in mind, DWD AI Assistant is easy to use and enhances your existing workflow without disruption.

Upgrade your Revit experience today with DWD AI Assistant and unlock the full potential of AI-driven design assistance. Download now from the Autodesk App Store and take your projects to the next level!

Prerequisite - Please Note: To use DWD AI Assistant, you will need an OpenAI API account. Please see the *Installation - OpenAI API setup* section below for details. The current version of the add-in routes your prompts to the OpenAI API. We plan to support connections to other AI providers in future versions so you can select which providers you wish to use.

Support

There is more help on the website at www.davidwooddesign.com, including examples and walkthroughs.

We want you to get the most out of the DWD AI Assistant, so please, if you need any help or advice in using it, get in touch with us at support@davidwooddesign.com.

Autodesk and Revit are registered trademarks or trademarks of Autodesk, Inc., and/or its subsidiaries and/or affiliates in the USA and/or other countries.

Contents

DWD AI Assistant Help	1
Support	1
Installation	3
Autodesk App Store download	3
OpenAI API setup	3
User Interface	4
Built-In Functions	5
Revit Version	5
Element Selection and Properties.....	5
Element Parameters	5
Element Modification	5
Element Creation	6
Geometry and Spatial Queries.....	6
Views.....	6
Filters.....	7
Sheets.....	7
Collectors	8
Collectors for Specific Categories	8
Family Operations.....	8
Annotation	9
Schedules	9
Document Management.....	9
Worksharing.....	9
Linked Models	10
Export and Output	10
File Operations	10
Model Health	10
User-Defined Functions - Dynamo.....	11
Dynamo Folders	11
Output.....	11
Sample Dynamo Functions.....	11
Model Context Protocol (MCP).....	12
Set up the MCP Server	12
Use the MCP Server	13
Release Notes (What's New)	14

Installation

Autodesk App Store download

Download the DWD AI Assistant installer from the Autodesk App Store and run it to install the Add-in. Revit needs to be closed while you install the Add-in.

When you restart Revit, it will show the 'Signed Add-In' dialog. Be sure to click the 'Always Run' to load the add-in.

OpenAI API setup

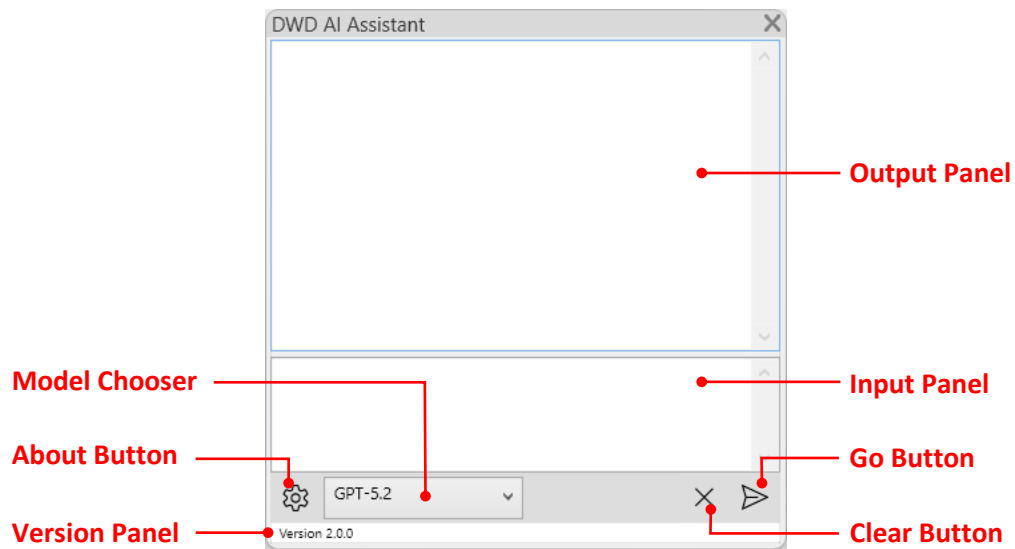
The Add-in uses the OpenAI API, so you will need an OpenAI API Key before you can use it. Go to the OpenAI website to create an API key and then set it as a Windows Environment Variable called 'OPENAI_API_KEY'.

<https://platform.openai.com/docs/quickstart>

You will also need to add a few dollars of credit to your OpenAI API account. Be sure to add the credit to your OpenAI API account not your ChatGPT account – they are different things.

We plan to support connections to other AI providers in future versions so you can select which providers you wish to use.

User Interface



Input Panel

This is where you type your prompt to the AI.

Go Button

Press this button or the Return key to send your Prompt to the AI.

Output Panel

This is where your chat with the AI will appear. Your prompt will be moved here, prefixed 'User:', and the AI's reply will appear below it, prefixed 'Assistant:'.

Model Chooser

Use this to pick which AI model to use

Clear Button

Press this button to clear the chat history and reset the AI to the initial state.

About Button

Press this button to open the 'About' dialog box.

Version Panel

This panel displays the current version of the add-in.

Built-In Functions

The add-in contains a set of functions that the AI will use to interact with Revit. We plan to add to this list in future versions. If you have ideas for extra functions that would be useful to you, please let us know.

Revit Version

- **GetRevitVersion**
Returns the version of Revit that the add-in is running in.
- **GetJournalFilePath**
Returns the path of the current Revit journal file.

Element Selection and Properties

- **CountSelectedElements**
Returns the number of elements in the current Revit selection.
- **GetIdsOfSelectedElements**
Returns the ElementIds of the elements in the current Revit selection.
- **SelectElementsById**
Selects the elements in the Revit UI that have the given IDs.

Element Parameters

- **GetParameterList**
Gets the list of parameters from a Revit element.
- **GetBasicElementInfo**
Retrieves basic information about a collection of Revit elements.
- **GetParameterValues**
Gets the value of a parameter for the Revit elements with the given IDs.
- **SetParameterValues**
Sets the value of a parameter for the Revit elements with the given IDs.
- **GetElementNameById**
Retrieves the name of the element that has the given ID.

Element Modification

- **DeleteElements**
Deletes one or more elements from the model.
- **MoveElements**
Moves elements by a given XYZ offset vector.
- **CopyElements**
Copies elements by a given XYZ offset vector and returns the new element IDs.
- **RotateElements**
Rotates elements around a vertical axis through the given origin point by the specified angle.
- **ChangeElementType**
Changes the type of one or more elements to the specified element type.
- **PinElements**
Pins one or more elements to prevent them from being moved accidentally.
- **UnpinElements**
Unpins one or more elements, allowing them to be moved.

- **MirrorElements**
Mirrors elements across a plane perpendicular to X or Y axis, optionally copying the original elements.

Element Creation

- **CreateWall**
Creates a wall between two points at a given level.
- **PlaceFamilyInstance**
Places a family instance at a given location in the model.
- **CreateLevel**
Creates a new level in the Revit model at the specified elevation.
- **CreateRoom**
Creates a new room at the specified point on a level. The point must be inside room-bounding elements.

Geometry and Spatial Queries

- **GetElementLocation**
Returns the location (point or curve endpoints) of one or more elements.
- **GetElementBoundingBox**
Returns the axis-aligned bounding box of one or more elements.
- **GetElementGeometrySummary**
Returns area, volume, perimeter/length values from element parameters where available.
- **GetDistanceBetweenElements**
Calculates the distance between two elements based on their locations.
- **GetRoomAtPoint**
Finds the room at a given XYZ coordinate point in the model.
- **GetElementsInRoom**
Gets elements spatially located inside a specific room.
- **FindUnplacedRooms**
Returns rooms that have been created but not yet placed in the model.
- **FindUnboundedRooms**
Returns rooms that are placed in the model but are not bounded (area = 0).

Views

- **GetIdOfCurrentView**
Returns the ElementId of the current Revit view.
- **ListAllViews**
Lists all views in the project with type, name, and ID.
- **SetActiveView**
Sets the active view in the Revit UI to the specified view.
- **GetViewProperties**
Returns detailed properties of a Revit view including type, scale, discipline, detail level, phase, crop box, and view range.
- **SetViewProperties**
Sets properties on a view such as scale, detail level, and discipline.
- **CreatePlanViews**
Creates a collection of plan views in Revit for the levels with the given IDs.

- **Create3DView**
Creates a new 3D view in the active document.
- **CreateCallout**
Creates a callout view within a parent view.
- **DuplicateView**
Duplicates an existing view.
- **GetViewFilters**
Gets the filters applied to a given view.
- **GetVisibilityGraphicsOverrides**
Gets the visibility/graphics overrides for a filter in a given view.
- **SetVisibilityGraphicsOverrides**
Sets visibility and graphic override settings for a category in a view.
- **GetViewTemplateSettings**
Gets the full settings of a view template.
- **ApplyViewTemplate**
Applies a view template to one or more views.

Filters

- **GetAllParameterFilters**
Lists all parameter filters defined in the active document with their names, IDs, and associated categories.
- **CreateParameterFilter**
Creates a new parameter filter with the specified rule that can be applied to views.
- **AddFilterToView**
Adds a parameter filter to a view and optionally sets visibility and projection line overrides.
- **RemoveFilterFromView**
Removes a parameter filter from a view.

Sheets

- **CreateSheets**
Creates a number of sheets in Revit.
- **ListSheets**
Lists all sheets in the active Revit document, sorted by sheet number.
- **RenameSheet**
Renames a sheet by changing its sheet number, name, or both.
- **PlaceViewsOnSheets**
Places a collection of views on a matching collection of sheets, one per sheet.
- **PlaceViewOnSheetAtPoint**
Places a view on a sheet at a specific point.
- **SetViewportLocation**
Moves an existing viewport on a sheet to a new center point.
- **GetViewportsOnSheet**
Lists all viewports placed on a sheet with their IDs, view names, and positions.
- **PlaceScheduleOnSheet**
Places a schedule view on a sheet at a specific point.

Collectors

- **CreateCollector**
Creates a FilteredElementCollector for the current Revit document.
- **Collector_OfCategory**
Collects elements of the given category.
- **Collector_OfClass**
Applies an ElementClassFilter using a .NET class name (e.g. Wall, FamilyInstance, Room).
- **Collector_WhereElementsIsNotElementType**
Collects only elements, not element types.
- **Collector_WhereParameterEquals**
Applies a parameter-equals filter to a FilteredElementCollector.
- **Collector_VisibleInView**
Creates a collector scoped to a specific view, returning only visible elements.
- **Collector_IntersectsBoundingBox**
Filters elements whose bounding box intersects a specified region.
- **Collector_OnLevel**
Filters elements associated with a specific level.
- **Collector_InLink**
Creates a collector scoped to a linked document.
- **CountCollector**
Counts the number of elements in the FilteredElementCollector.
- **GetElementIdsFromCollector**
Gets the ElementIDs of the elements in the FilteredElementCollector.

Collectors for Specific Categories

- **GetLevels**
Gets all the levels from the current Revit document.
- **GetScopeBoxes**
Gets all the scope boxes.
- **GetFloorPlanViewFamilyTypes**
Gets all the ViewFamilyTypes for floor plans.
- **GetViewTemplates**
Gets the view templates.
- **GetTitleblocks**
Gets the titleblocks.
- **GetBuiltInCategories**
Gets all the BuiltInCategories from Revit.

Family Operations

- **ListFamiliesAndTypes**
Lists all loadable families and their types, filtered by category name.
- **GetFamilyTypeParameters**
Lists all parameters of a family type with names, values, storage types, and read-only status.
- **LoadFamily**
Loads a family (.rfa file) into the current document.

- **DuplicateFamilyType**
Duplicates an existing family type and assigns it a new type name.

Annotation

- **CreateTextNote**
Creates a text note in a view at a specified position.
- **EditTextNote**
Modifies the text content of an existing text note element.
- **TagElement**
Creates an annotation tag for an element in a view.
- **GetTagsForElement**
Returns all annotation tags associated with a specific element.
- **GetAnnotationTypes**
Lists annotation types including text note types, dimension types, and spot dimension types.
- **GetDimensions**
Lists dimension elements in a given view, with optional filter by dimension type.

Schedules

- **ListSchedules**
Lists schedules in the active document, optionally filtered by category name.
- **CreateSchedule**
Creates a new schedule view for a specified category.
- **GetScheduleData**
Gets the data rows from a schedule view, including headers and cell values.

Document Management

- **CreateNewModel**
Creates a new Revit model from a template.
- **GetDefaultTemplate**
Gets the path of the default Revit template.
- **GetTemplateList**
Gets the list of available Revit templates.
- **OpenExistingModel**
Opens an existing Revit model file.
- **SaveModel**
Saves the current model, synchronising with central if the model is workshared.

Worksharing

- **EnableWorksharing**
Enables worksharing on the current model.
- **CreateWorksets**
Creates one or more worksets in the current model.
- **GetElementWorkset**
Returns the workset name for each specified element.
- **SetElementWorkset**
Moves elements to the specified workset.

Linked Models

- **GetLinkedModels**
Returns all Revit linked models in the active document, including file path and load status.
- **ReloadRevitLink**
Reloads a Revit link from its current path.
- **GetLinkTransform**
Returns the total transform (origin, basis vectors) of a Revit link instance.

Export and Output

- **ExportToIFC**
Exports the current Revit model to an IFC file.
- **ExportToDWG**
Exports specified views to DWG files.
- **ExportToPDF**
Exports specified sheets to PDF (requires Revit 2023 or later).
- **GetPrintSettings**
Lists the print settings defined in the current document.

File Operations

- **ReadTextFile**
Reads the contents of a text file.
- **WriteTextFile**
Writes or appends content to a text file.
- **ListDirectory**
Lists the files and folders in a directory.

Model Health

- **GetModelWarnings**
Returns the current warnings in the Revit model.

User-Defined Functions - Dynamo

You can add your own functions to the AI Assistant's skillset using Dynamo.

Create and test your Dynamo function, and then save it with a descriptive name in the AI Assistant's Dynamo folders. The AI Assistant will read the dyn files you place in these folders, and call them as it needs to alongside the built-in functions.

The AI Assistant will use the name of the dyn file to understand what your Dynamo function does. You can also set the Description field inside the dyn to give the AI Assistant more information. This image shows the Description field filled-in for one of the sample dyn files included with the AI Assistant:

```
{
  "Uuid": "47ac2ffd-a6df-47e0-bfa4-342a1a07205c",
  "IsCustomNode": false,
  "Description": "Create Sample Walls, one for each WallType in the model",
  "Name": "CreateSamplewalls",
  "ElementResolver": {
    "ResolutionMap": {}
  },
}
```

Dynamo Folders

There is a common Dynamo folder, where you can put dyn files that will work in all Revit versions; and a set of version-specific folders for use if you have version-specific dyn files.

The common Dynamo folder is:

```
%AppData%\Autodesk\ApplicationPlugins\DavidWoodDesign_DWDAIAssistant.bundle\Contents\Source\Dynamo
```

The version-specific Dynamo folders are:

```
%AppData%\Autodesk\ApplicationPlugins\DavidWoodDesign_DWDAIAssistant.bundle\Contents\Source\
```

Output

In this version, your Dynamo function can have a text output. Set up a Watch Node at the end of your graph, rename it as 'Output' and set it to 'Is Output'. The AI Assistant will read the output from your graph so it knows the result of the run.

The sample graphs output the number of walls and floors created.

Sample Dynamo Functions

The AI Assistant installation includes two sample Dynamo functions: Create Sample Walls and Create Sample Floors.

Both the sample functions are set up in millimetres. If your Revit is set up to use other length units, you will need to adjust the functions before using them.

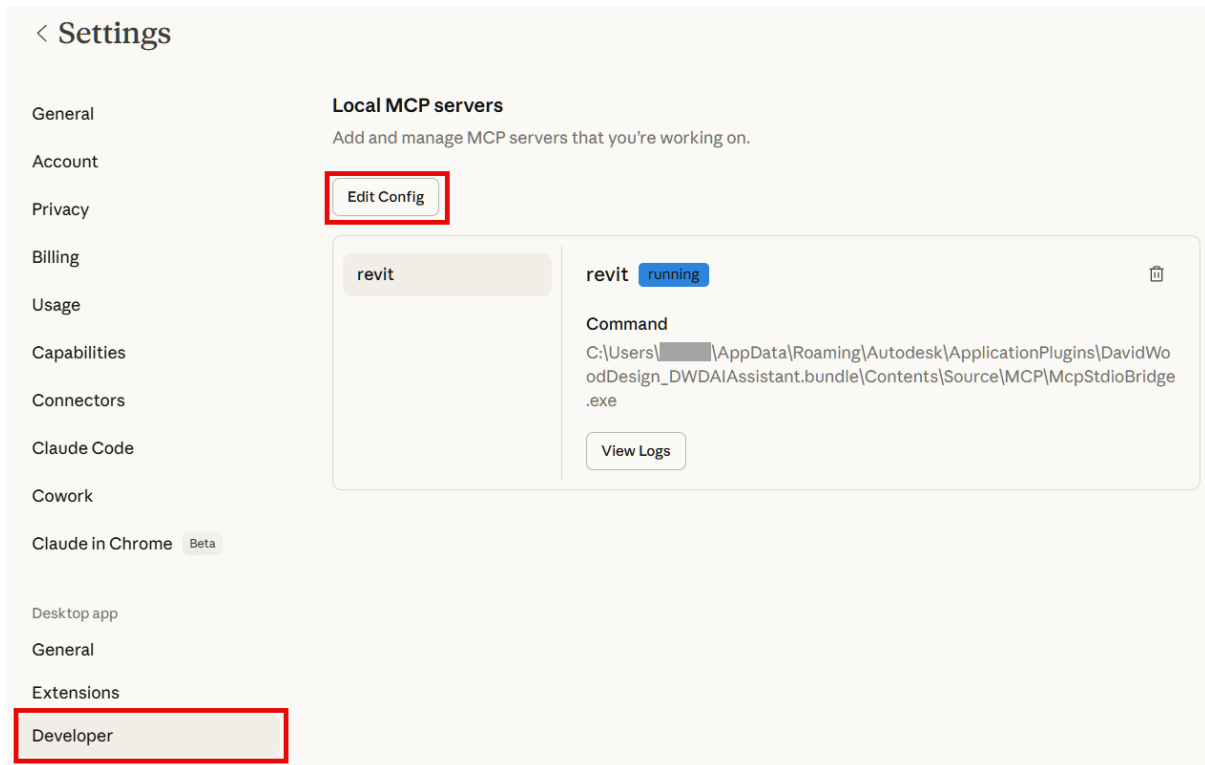
Model Context Protocol (MCP)

DWD AI Assistant includes a Model Context Protocol (MCP) Server, so you can control your Revit sessions from external MCP Clients like Claude Desktop.

Set up the MCP Server

Add DWD AI Assistant as a Local MCP Server.

In Claude Desktop, click your account name (bottom left) and choose 'Settings'. Then navigate to the 'Developer' tab and click the 'Edit Config' button.



That will take you to the folder containing `claude_desktop_config.json`. Open the file in a text editor and add or edit the `mcpServers` section to include the DWD AI Assistant's `McpStdioBridge.exe`:

```
"mcpServers": {
  "DWD_AI_Assistant_for_Revit": {
    "command":
"C:\\Users\\<your windows id>\\AppData\\Roaming\\Autodesk\\ApplicationPlugins\\DavidWoodDesign_DWDIAssistant.bundle\\Contents\\Source\\MCP\\McpStdioBridge.exe"
  }
}
```

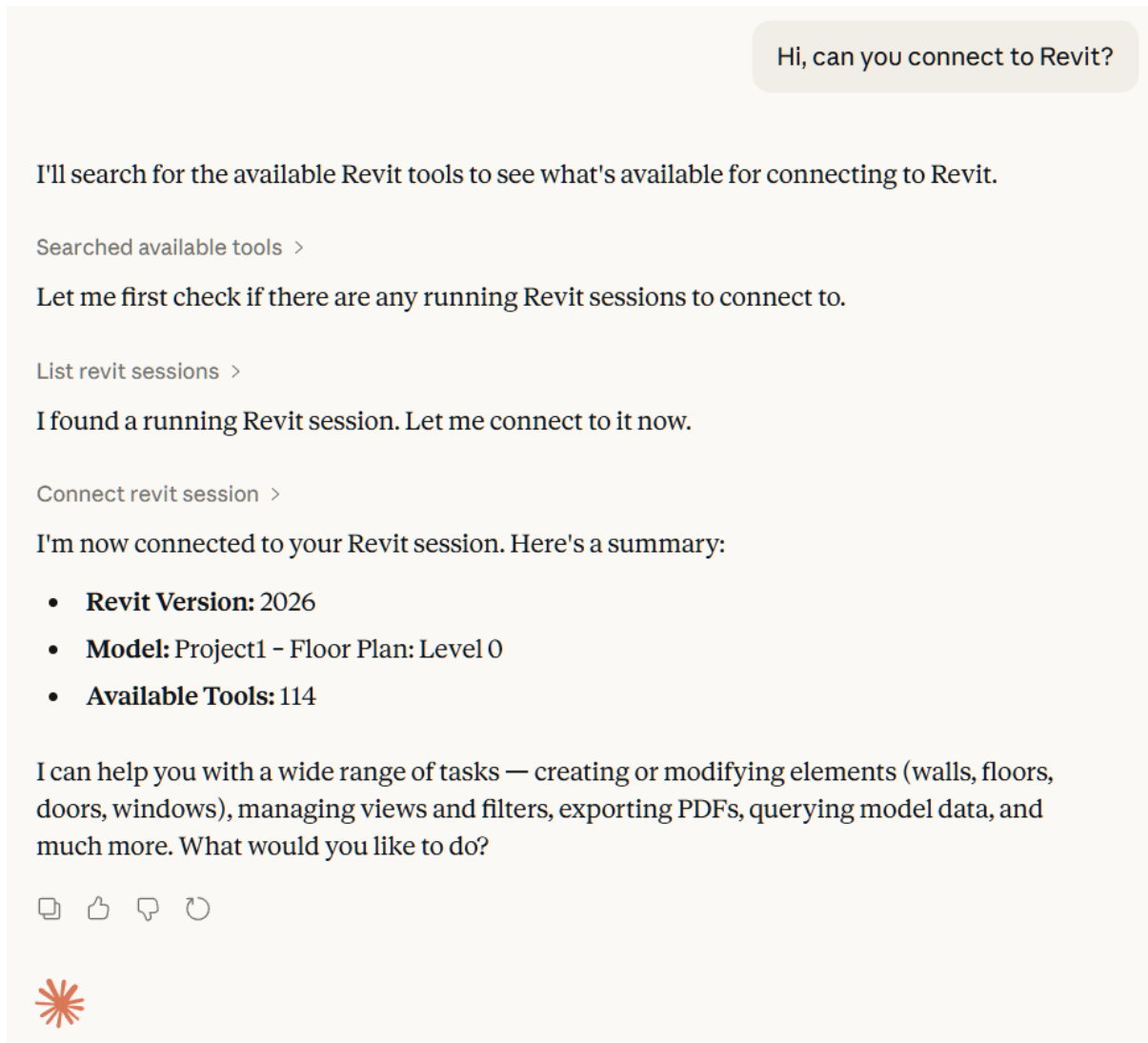
Replace `<your windows id>` with the correct folder name for your user folder.

Separate the `mcpServers` section from other sections with a comma, and (if you have more than one `mcp` server), separate them with commas.

Save your changes and restart Claude (you will need to Quit Claude from the System Tray, or restart your computer).

Use the MCP Server

Start Revit and Claude Desktop. Ask Claude to connect to Revit:



Hi, can you connect to Revit?

I'll search for the available Revit tools to see what's available for connecting to Revit.

Searched available tools >

Let me first check if there are any running Revit sessions to connect to.

List revit sessions >

I found a running Revit session. Let me connect to it now.


Connect revit session >

I'm now connected to your Revit session. Here's a summary:

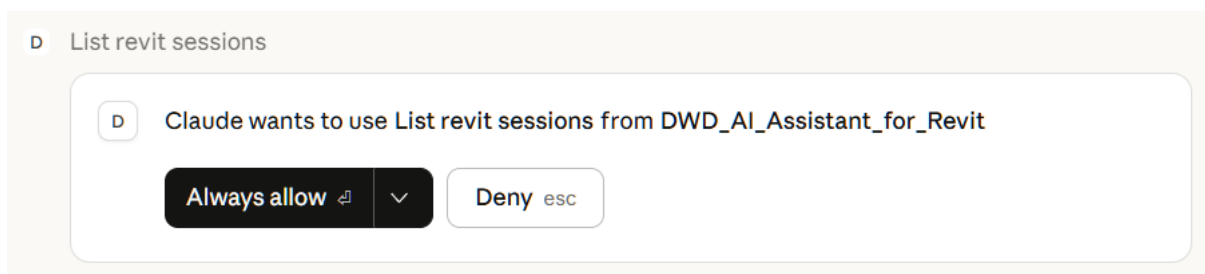
- **Revit Version:** 2026
- **Model:** Project1 - Floor Plan: Level 0
- **Available Tools:** 114

I can help you with a wide range of tasks — creating or modifying elements (walls, floors, doors, windows), managing views and filters, exporting PDFs, querying model data, and much more. What would you like to do?

📄 👍 💬 ↻



Claude will ask for permission to use each of the tools in the MCP Server, and then will use them to access the Revit sessions and use the tools in the DWD AI Assistant.



D List revit sessions

D Claude wants to use List revit sessions from DWD_AI_Assistant_for_Revit

Always allow ⌨️ ▼ Deny esc

Release Notes (What's New)

V1-5-0 New Features:

- Extra functions
- New models
- MCP interface
- Bugfixes and stability enhancements.

V1-4-1 New Features:

- Bugfixes and stability enhancements.

V1-4-0 New Features:

- Revit 2026 version.
- Bugfixes and stability enhancements.

V1-2-0 New Features:

- User-defined functions in Dynamo.
- Bugfixes and stability enhancements.

V1-1-0 New Features:

- Additional Functions.
- Bugfixes and stability enhancements.

V1-0-0 New Features:

- First version for the Autodesk App Store.