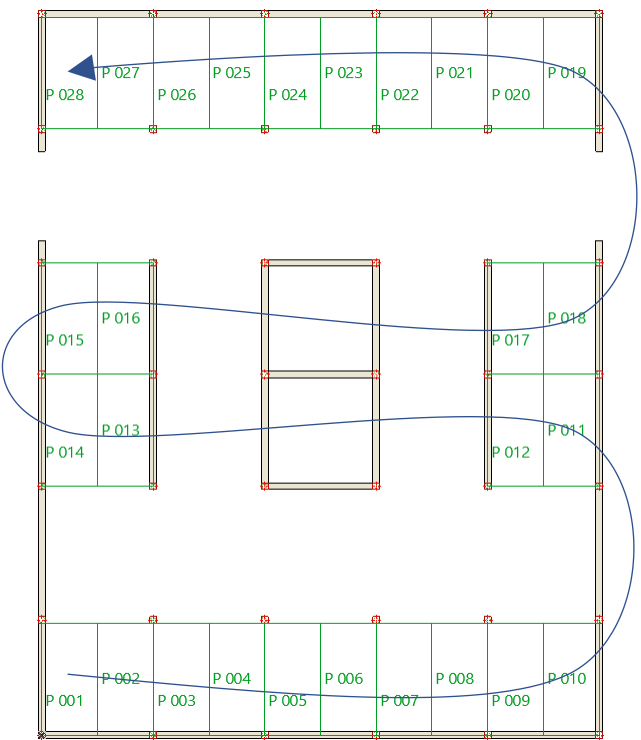


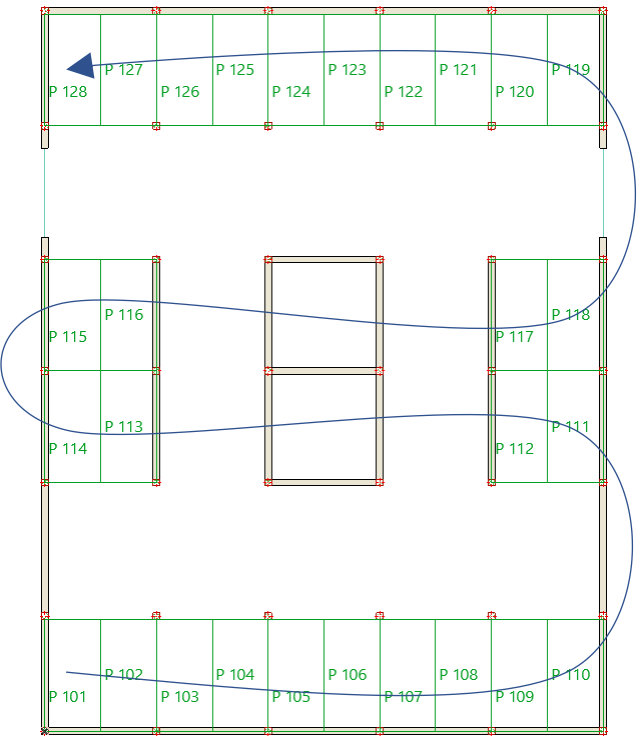
Step by Step Guide for Parking Space Numbering

Short Description

Generates unique ordered numbers for the element IDs of parking spaces, for all the elements classified as Parking Space in ARCHICAD Classification System. The script implements a zigzag ordering, see images below:



The ground floor of the parking house



The first floor of the parking house

Files to Download

Python Script

[https://graphisoft-my.sharepoint.com/:u:/g/personal/tlorantfy_graphisoft_com/EavqSGGkwJNKIZRSfyYS8zcB4rb - NYOT_IHJblZ41X_8w?e=1WXFwu](https://graphisoft-my.sharepoint.com/:u:/g/personal/tlorantfy_graphisoft_com/EavqSGGkwJNKIZRSfyYS8zcB4rb-NYOT_IHJblZ41X_8w?e=1WXFwu)

Example ARCHICAD Project

https://graphisoft-my.sharepoint.com/:u:/g/personal/tlorantfy_graphisoft_com/EaoCU1Xvl29PigKBbTaWAelBMxJHcRuQ_CdJ-6aTfxJvTA?e=ips5ml

Step by Step Tryout Guide

1. Download the Python script.
2. Download the example project file and open it in ARCHICAD.
3. Run the Python script (from Python Palette or in any external application).
4. The element IDs of the parking spaces will be automatically set on each floor.

The parking spaces in the example project display the value of element ID on the floor plan view.

Demo Video

https://tprivatenew.s3.amazonaws.com/pulse/tlorantfy-graphisoft/attachments/13785917/parking_spaces_demo.mp4

Details

Requirements

To execute the script successfully without any customization, the following requirements must be met. See the Customization section to change or eliminate these requirements.

- A classification system named ARCHICAD Classification
- A classification named Parking Space in the ARCHICAD Classification System
- ARCHICAD 24.2310 or above
- ARCHICAD-Python Connection (Archicad package) version 24.2310 or above

Element ID is a built-in property, which means all elements in all projects have this information.

Customization

Feel free to open the script in any text editor and customize it. Each example script has a well-separated section, named "CONFIGURATION". Change the values in that section for customization.

```
##### CONFIGURATION #####
propertyId = acu.GetBuiltInPropertyId('General_ElementID')
propertyValueStringPrefix = 'P '
classificationItem = acu.FindClassificationItemInSystem(
    'ARCHICAD Classification', 'Parking Space')
elements = acc.GetElementsByClassification(
    classificationItem.classificationItemId)

ROW_GROUPING_LIMIT = 0.25
STORY_GROUPING_LIMIT = 1

def GeneratePropertyValueString(storyIndex: int, elemIndex: int) -> str:
    return f"{propertyValueStringPrefix}{storyIndex:1d}{elemIndex:02d}"
#####
```

By default, this script sets element IDs of all project elements classified as Parking Space classification in ARCHICAD Classification System. To set IDs for elements by type (instead of Classification), change `GetElementsByClassification` to `GetAllElements` or `GetElementsByType`:

```
elements = acc.GetElementsByType("Zone")
```

This table explains the meaning of each variable and values in the configuration section to help customization.

Name of the variable	Description	Default value
propertyId	The identifier of the property to be modified.	By default, the script modifies the element ID property of the elements. Its identifier is <code>General_ElementID</code> .
propertyValueStringPrefix	The prefix that will be appended to the beginning of the modified element IDs.	P
classificationItem	The classification to query the elements.	Parking Space classification in ARCHICAD Classification System.
elements	The elements to be modified.	Elements with the given classification.
ROW_GROUPING_LIMIT	The maximum difference between the Y coordinates of the parking spaces in the same row. The implemented zigzag logic needs to detect rows.	0.25 meter
STORY_GROUPING_LIMIT	The maximum difference between the Z coordinates of the parking spaces in the same story. The story index will be added into the modified value.	1 meter
GeneratePropertyValueString	This function generates the string property value from the story index and element index parameters.	P 101

Possible Errors

If the script fails to execute and returns an error, please make sure the script's requirements have been met.

The following are possible errors and how to resolve them.

[StopIteration, return next\(system.classificationSystemId for system in ...](#)

The required classification system is not defined in the user's project file. It is possible that the user has modified the project's classification systems. Double check to see if the name of the classification system defined in the project matches that in the script configuration.

[AttributeError: 'NoneType' object has no attribute 'classificationItemId'](#)

Same error as above, but here, the required classification itself is not defined in the given classification system of the project file. It is possible that the user has modified the classifications. Double check to see if the name of the classification defined in the project matches that in the script configuration.