Introduction

Easy Workspace allows you to save important files in a separate window for quick access to them, as well as create panels with custom content.

How to get started
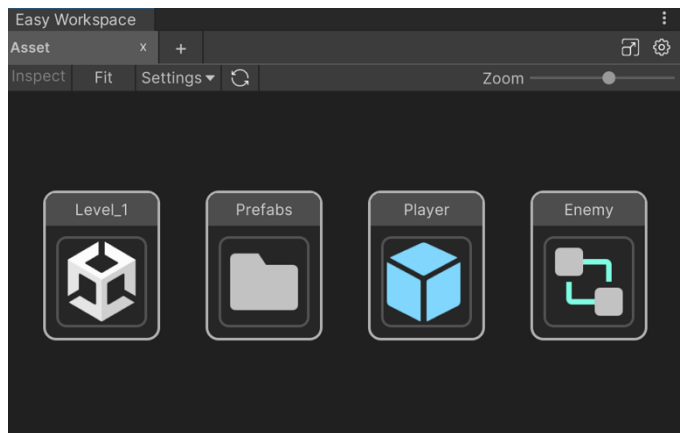
Open the tool via Window > Easy Workspace, afterwards you can drag and drop files or specify them using the RMB.

Default panels

Single Asset

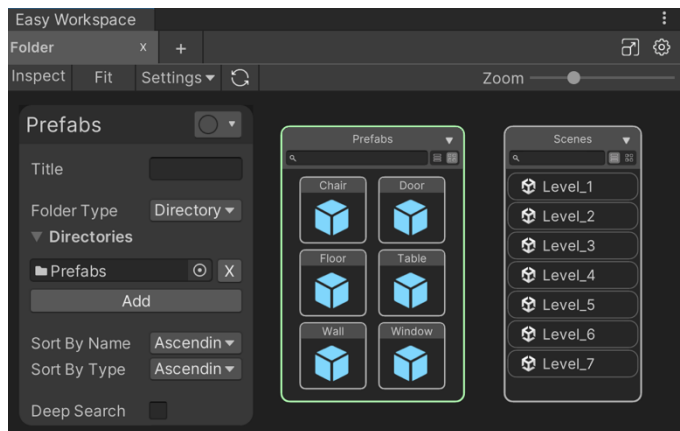Holds only a single file.

Can be created by dragging and dropping files or using the RMB.

Folders

Holds many files and has many settings in the inspector.

Can be created by dragging and dropping multiple files or using the RMB.

Advanced use

Basically, assets can be opened (LMB) and dragged like a normal file.

But you can also select an asset in Project: RMB or Shift + LMB
Or open in a separate window: MMB or Alt/Option + LMB.

Inspector

Each panel has its own settings window in the inspector. To open it, double-click on the panel or select and click the "Inspect" button in the upper left corner.

Custom panels

You can also create custom panels with your own features. You can create them through code, below are the instructions. It is advisable to have GUI experience for this purpose

```
1    using UnityEngine;
2    using UnityWorkspace;
3
4    [UWCustomFile("Example", 200f, 200f)]
5    public class PanelExample : UWCustom
6    {
7        protected override void DrawGUI()
8        {
9            GUILayout.Label("Example");
10       }
11   }
```

1.   Create a new script in the Editor folder.
2.   Inherit the script from EWCustom
3.   Add EWCustomFile attribute with name and size parameters
4.   Override the DrawGUI method with your GUI drawing logic
5.   The panel is ready to use. This is a minimally working script, below will be an example of additional features

Example: Note

This is a ready example of a note implementation. It can be created via RMB.

```
1   using UnityEditor;
2   using UnityEngine;
3
4   namespace EasyWorkspace
5   {
6       [EWCustom("Note", 200f, 200f)]
7       [EWResizable(100f, 100f, 700f, 700f)]
8       public class EWNote : EWCustom
9       {
10          [SerializeField] private string _text;
11          [SerializeField] private int _fontSize = 16;
12
13          protected override void DrawGUI()
14          {
15              GUIStyle style = new GUIStyle(GUI.skin.textArea) { wordWrap = true, fontSize = _fontSize };
16              _text = EditorGUILayout.TextArea(_text, style, GUILayout.ExpandHeight(true));
17          }
18
19          protected override void DrawInspectorGUI()
20          {
21              EditorGUILayout.LabelField("Font Size");
22              _fontSize = EditorGUILayout.IntSlider("", _fontSize, 8, 48);
23          }
24
25          protected override string GetCollapseInfo()
26          {
27              return _text;
28          }
29      }
30  }
```

- [EWFileResizable] - Adds the ability to resize the panel. Requires minimum and maximum size to be specified.
- DrawInspectorGUI() - Method used to draw settings in the inspector. Use serialized variables to store the settings.
- GetCollapsedInfo() - Display information while the panel is collapsed.

If you need to save the panel settings at some point, just call the Save() method.

If you have any questions or suggestions, please write to nutrolabgames@gmail.com