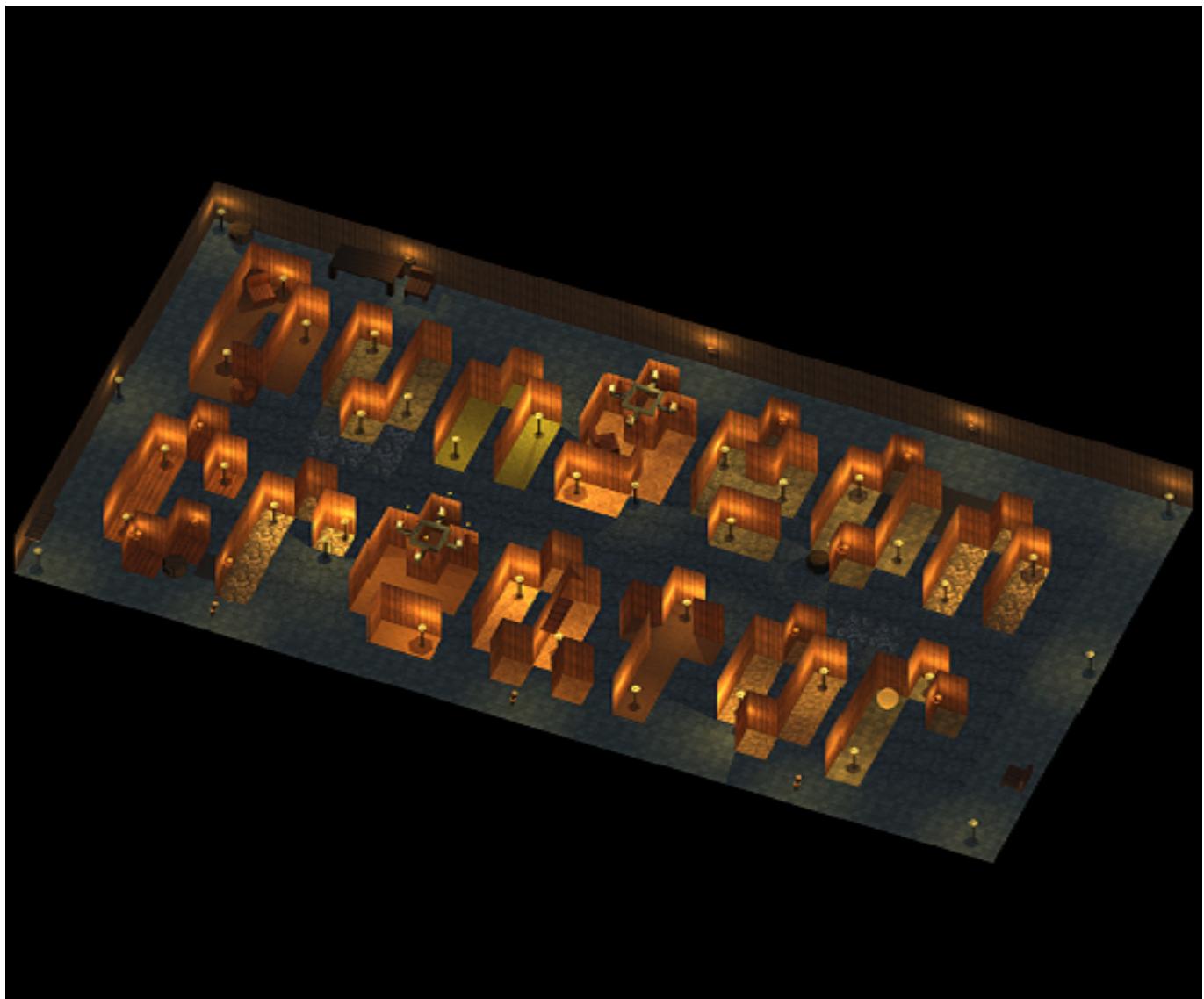


Dungeon Creator



Dungeon Creator is a Unity Editor Package that helps build and edit your dungeon floors and walls with ease.

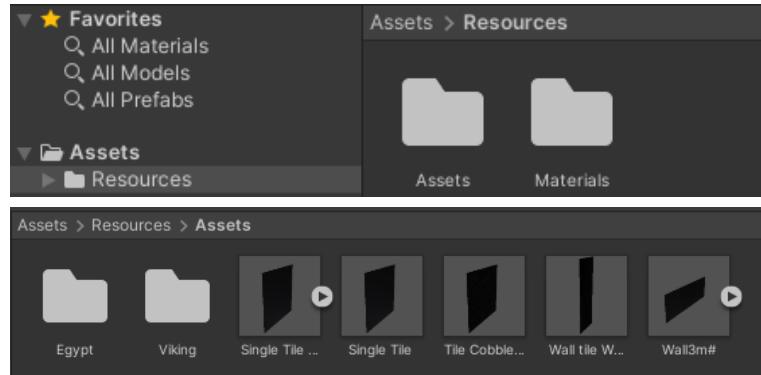
The package contains a drawing window which will generate the appropriate floors and walls for every room in your dungeon. Create a simple dungeon in less than 5 minutes with 9 included materials and over 20 themed assets.

Contents

Included Assets and Materials	3
Tiles	3
Materials	3
Assets	4
Viking	4
Ancient Egypt	4
Start up	5
Installation	5
Creator	6
Draw new room	6
Drawing and editing rooms	6
Remover	8
Material replacements (Optional Features)	9
Additional material spawning	9
Wall Material changes	11

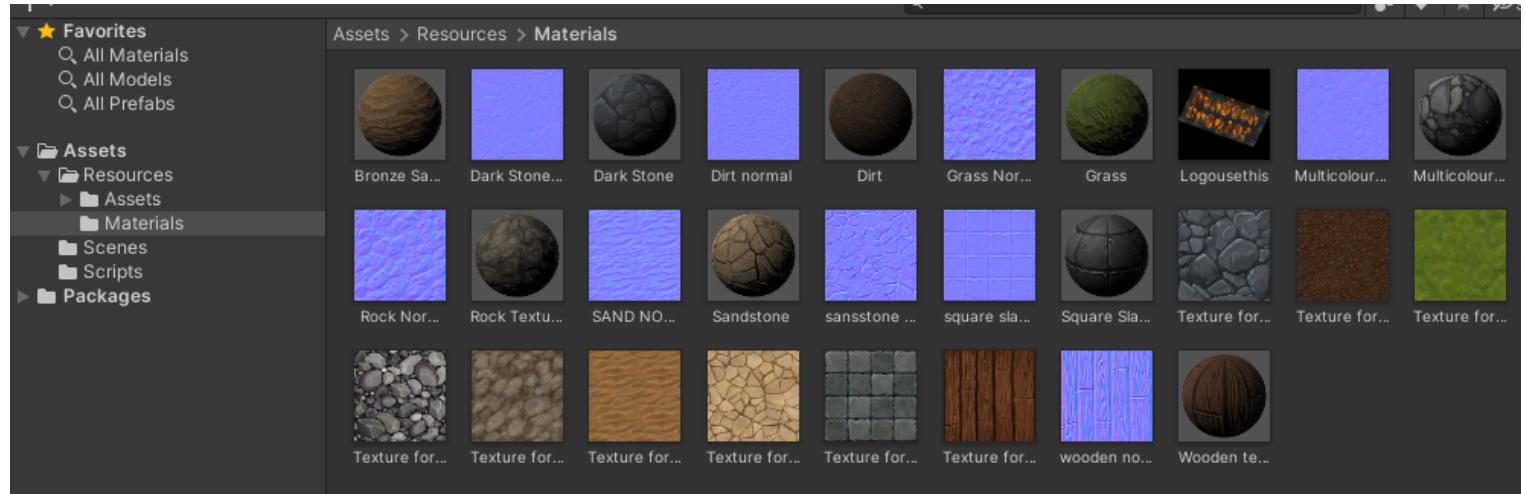
Included Assets and Materials

Tiles



Tiles are stored in the Assets folder within the Resources Folder. Here you will find prefabs and models for wall tiles and floor tiles. These will generate into the scene using the drawing window.

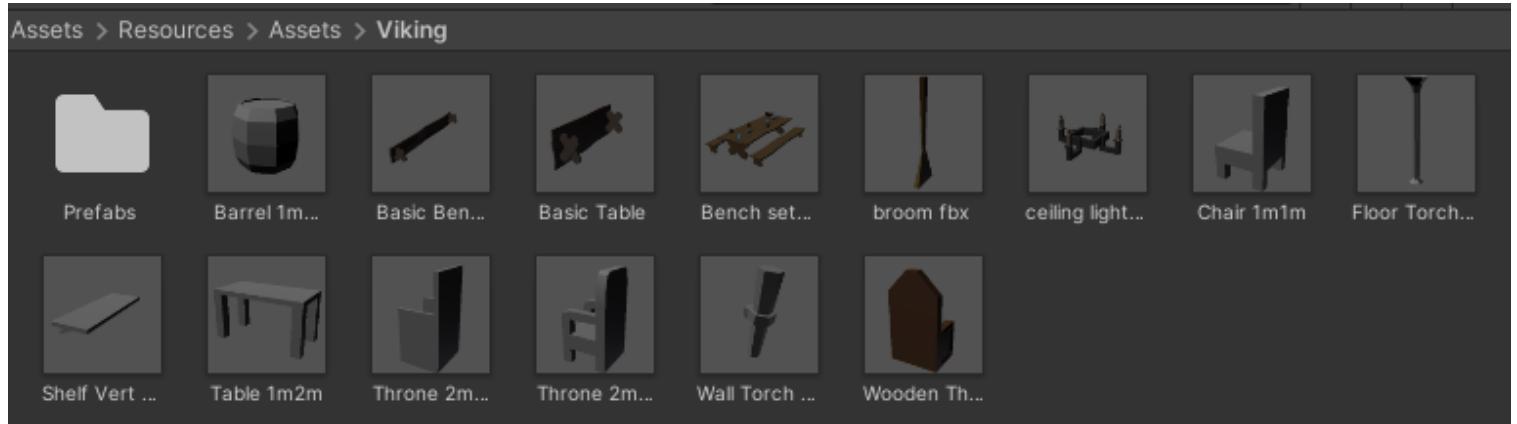
Materials



Materials are stored in the Materials folder within the Resources Folder. Here you will find materials with their respective normal maps and textures. When a tile is generated, all parts of the material will be generated also. Material changes happen through the material selection buttons on the drawing window.

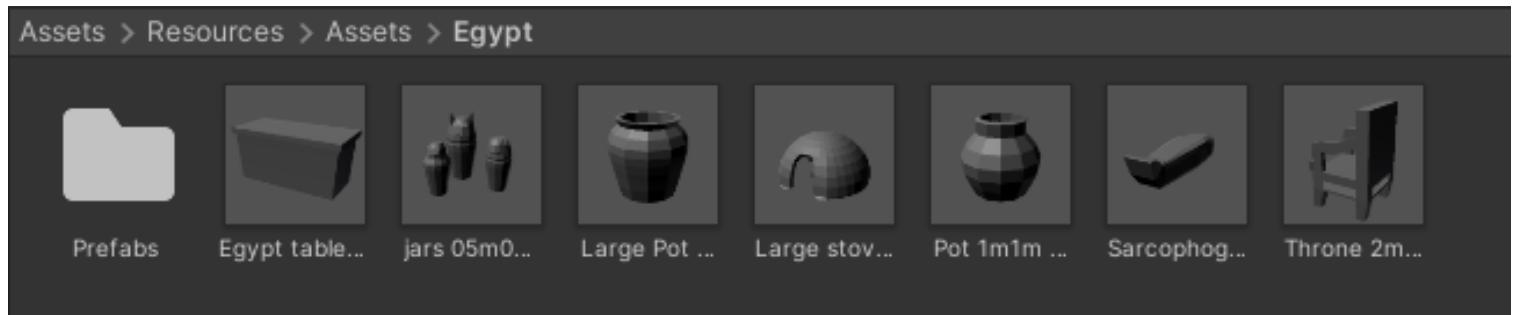
Assets

Viking



Viking assets are stored in the Viking folder in the Assets folder within the Resources folder. Here you will find assets themed towards the Vikings. These assets can be directly dragged from Unity into the scene once rooms are created. I have included a range of furniture type assets and some lighting asset options. The Prefabs are also included in their own folder.

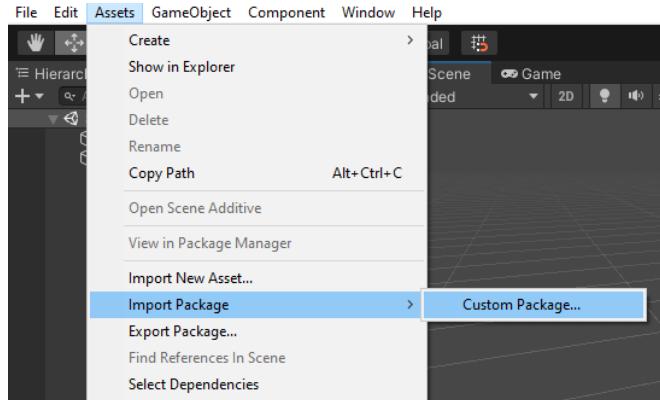
Ancient Egypt



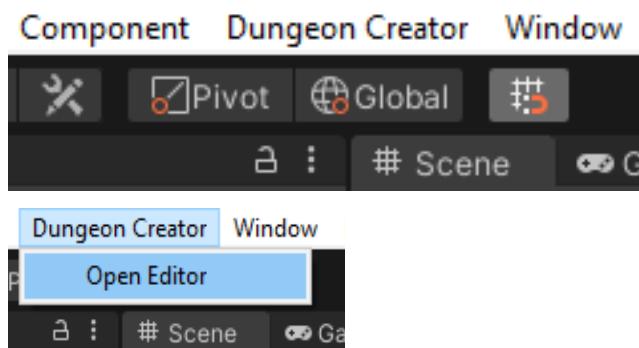
Ancient Egypt assets are stored in the Egypt folder in the Assets folder within the Resources folder. Here you will find assets themed towards the Ancient Egyptians. Similar to other included assets, these can be dragged into the scene easily.

Start up

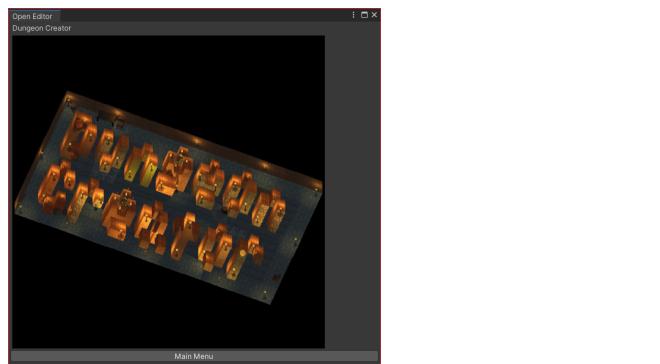
Installation



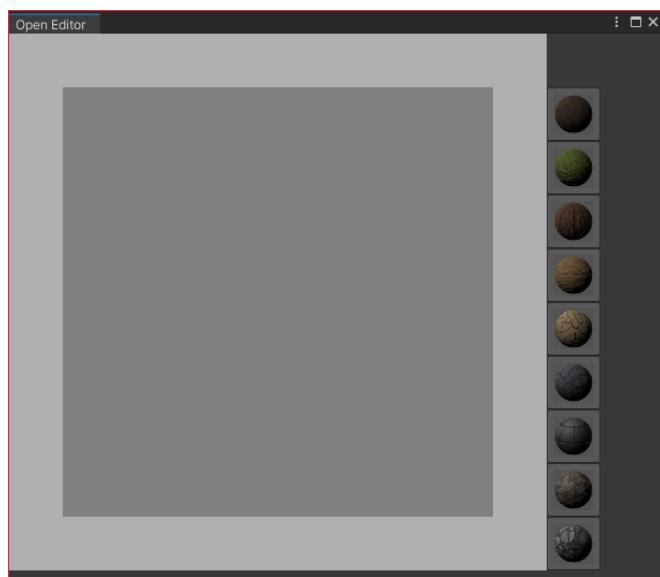
Once the Package is downloaded and you have a Unity project open you can open the Assets tab and select Import Package. This will load the package and open a window with the contents of the package. Once imported new folders will appear in your project containing the assets, materials and scripts.



The Tool Handle Position should be changed to the 'Pivot' option and the Tool Handle Rotation should be changed to the 'Global' option so that 'Grid snapping' can be turned on to aid the moving of rooms later in production.



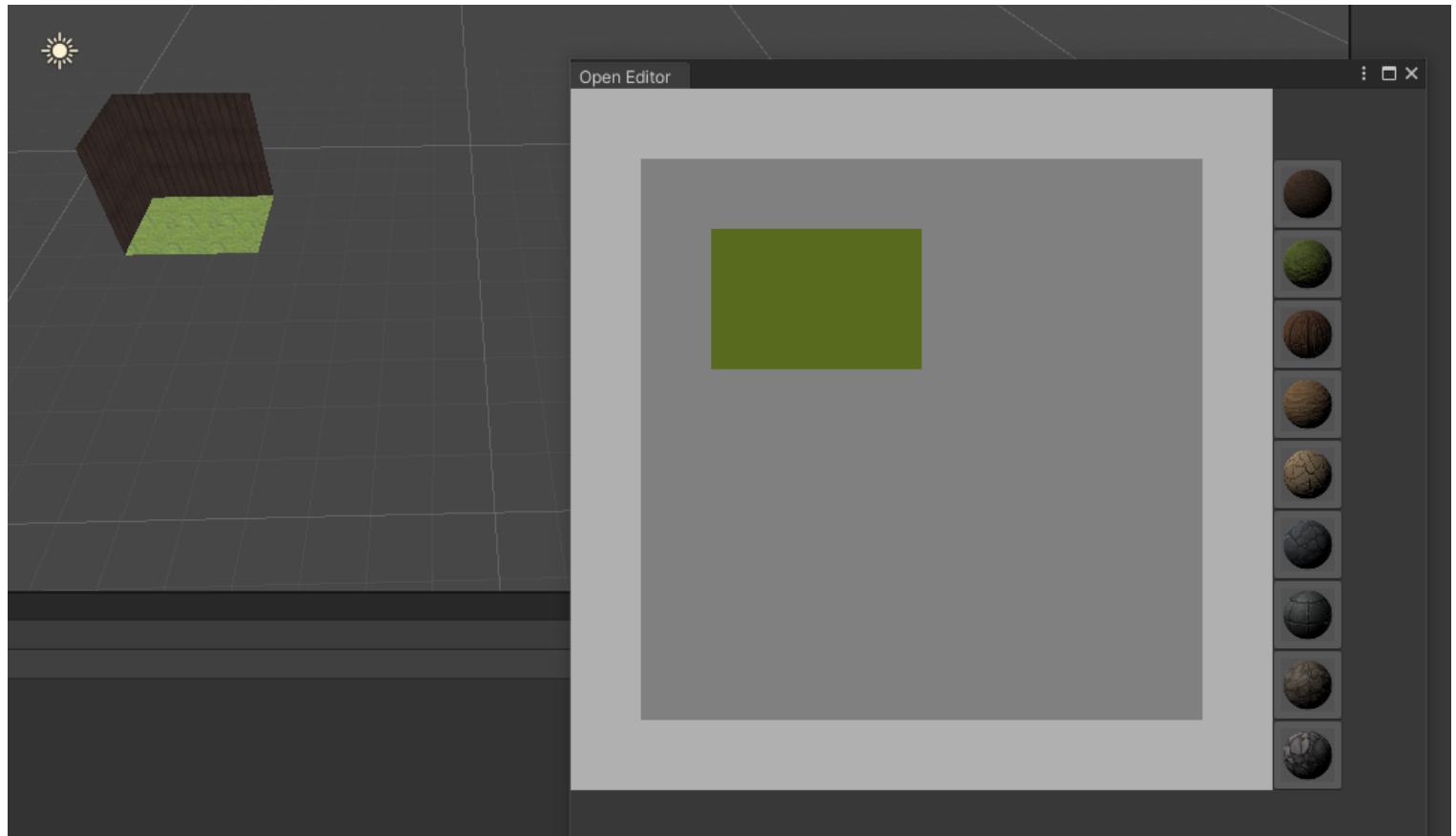
To start creating a dungeon room, find the Dungeon Creator tab at the top of the screen and open the drop down option, 'Open Editor'. This will open a window with the Logo for Dungeon Creator and a Main menu Option.



The next screen will have a button for creating a room. Here it will load the material buttons to choose your material type for the tile you want to place. Hovering over each material type will tell you its name.

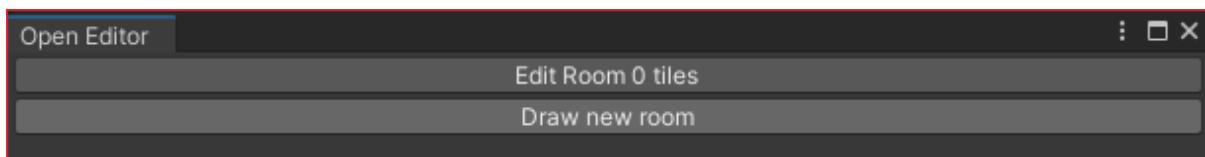
Creator

Draw new room

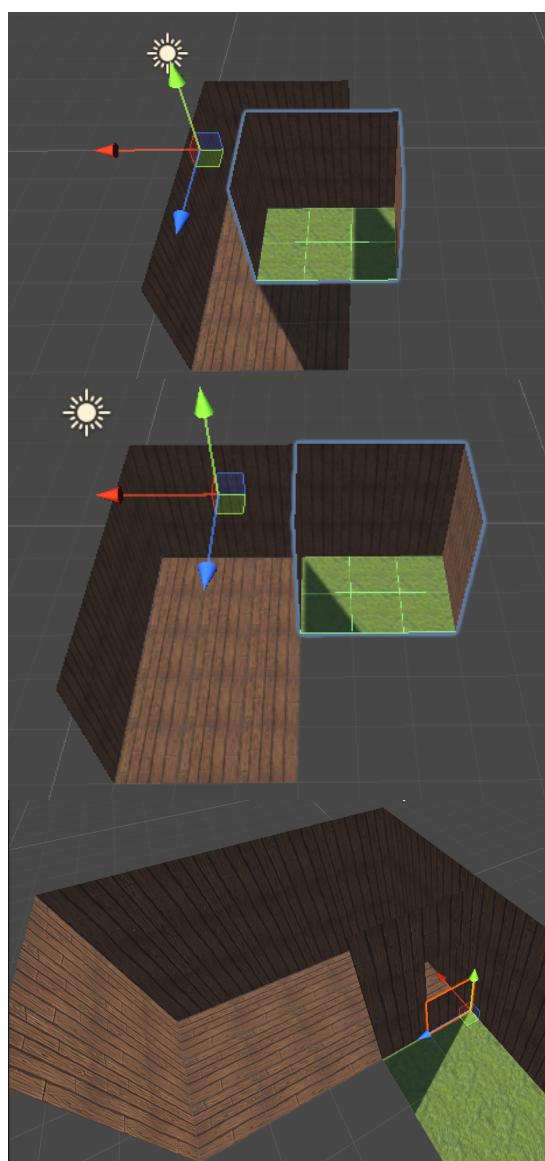


With the drawing window open and a material selected, you can Left-click on squares in the drawing texture to generate them in the Unity scene. The tiles that spawn already have scripts that know when to spawn and despawn each wall. The maximum drawing size here is 9x9 tiles however you can join multiple rooms up and delete walls in between to form much larger rooms.

Drawing and editing rooms



Reloading the window will allow you to see how many rooms you have created. You can rename these in the inspector. Here you can draw new rooms however the drawing space in Unity will be the same.

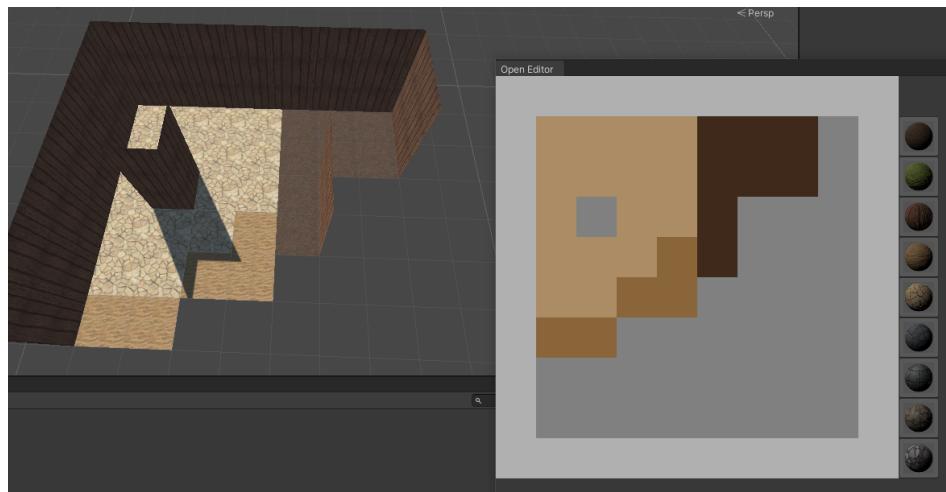


Rooms can be moved by selecting the room manager in Unity's Hierarchy by name, for example "Room 0". The settings for grid lock will apply here when moving rooms around. They can snap to grids allowing rooms to align perfectly.

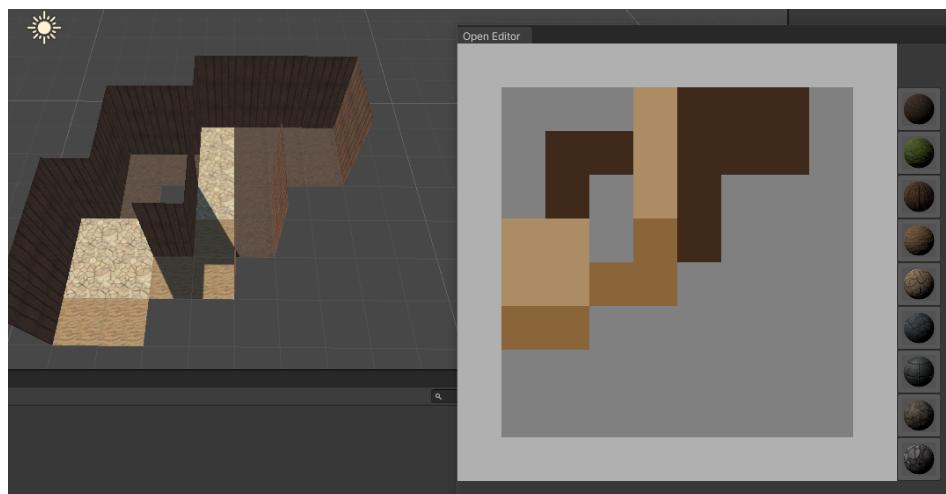
Joining rooms will have 2 sets of walls. In order to form a doorway, you can select which walls to delete in the hierarchy.

Rooms can be edited at any time using the drawing window and selecting the room you wish to change.

Remover



With the drawing window open you are able to Right-Click on tiles to remove them from the Unity scene. This will use maths in my script to generate walls on the edges of tiles that have been removed and calculate corner piece walls too.



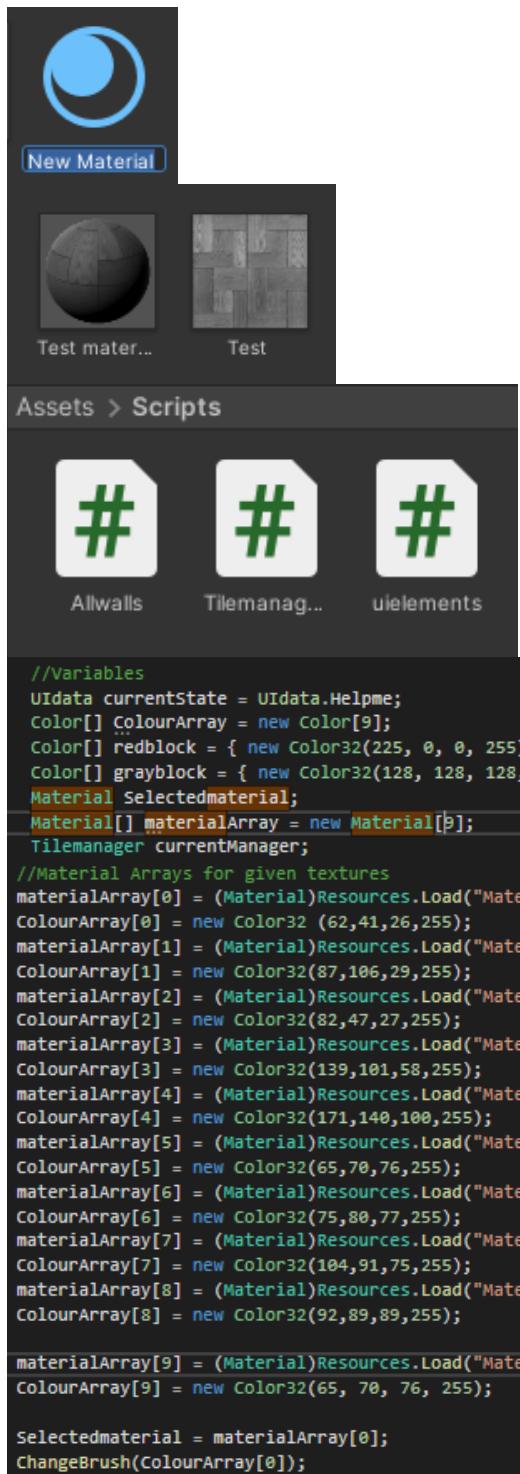
Here you are also able to replace tiles and their materials. Simply Left-Click on the desired material and Left-Click on the tile you wish to replace. This will generate the new tiles in real time to ensure you have full visualisation

of your dungeon whilst designing it.

Material replacements (Optional Features)

These optional features are not yet fully developed.

Additional material spawning



Additional materials will need to be added into the Materials folder within the Resource Folder. These can be renamed also. Here you will need to upload a texture and apply that to a new material. You also have the option of including other image maps which should be added into the materials before moving on.

Using the scripts folder you are able to open the script named 'uielements' which includes the spawning code for each floor and wall tile.

Locate the Variables at the top of the script and find the lines:

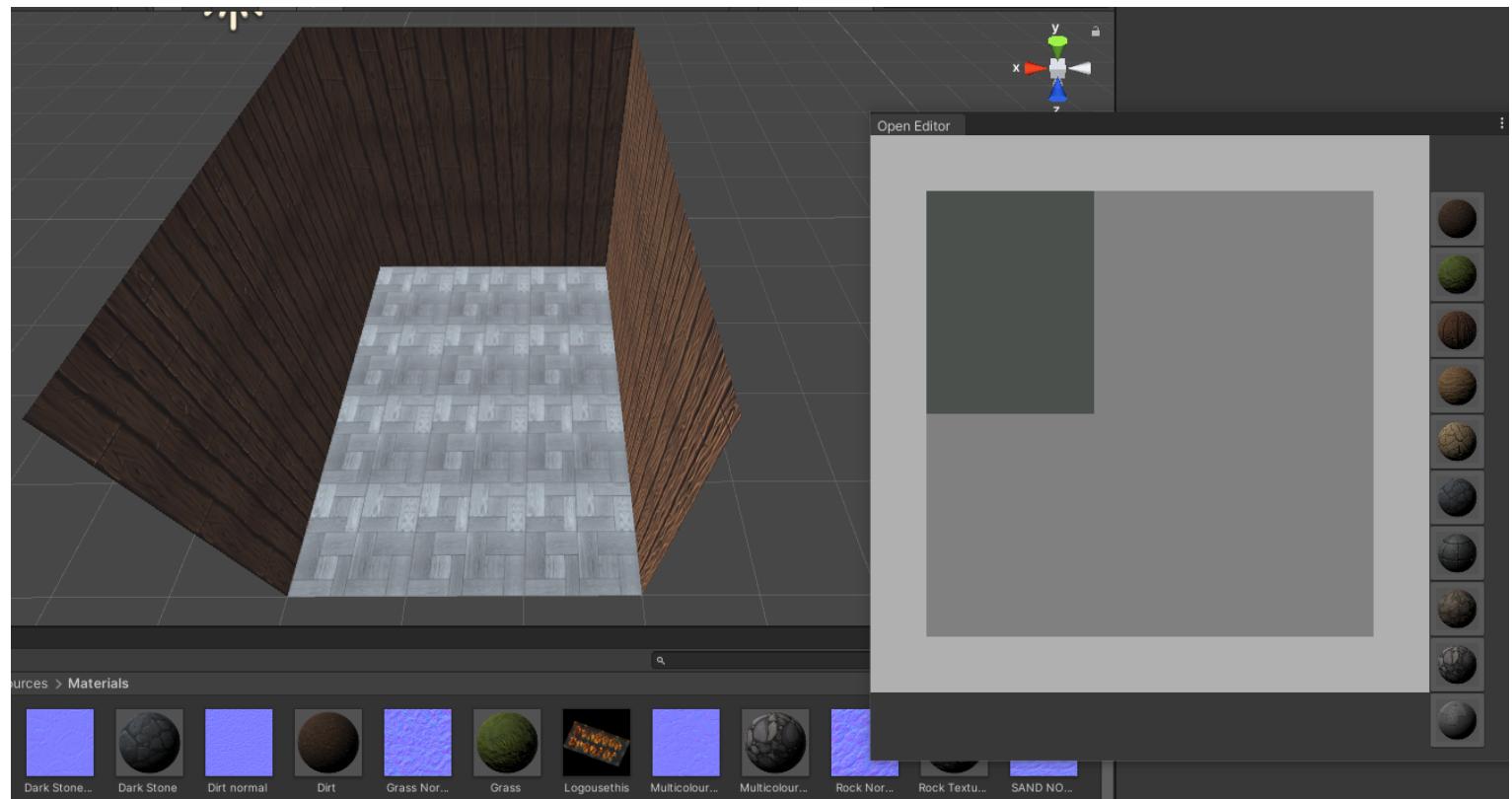
```
Color[] ColourArray = new Color[9];  
Material[] materialArray = new Material[9];
```

Changing the number will change how many materials you wish to choose from. Since we are testing one additional, this number will change to 10. Make sure to change them both.

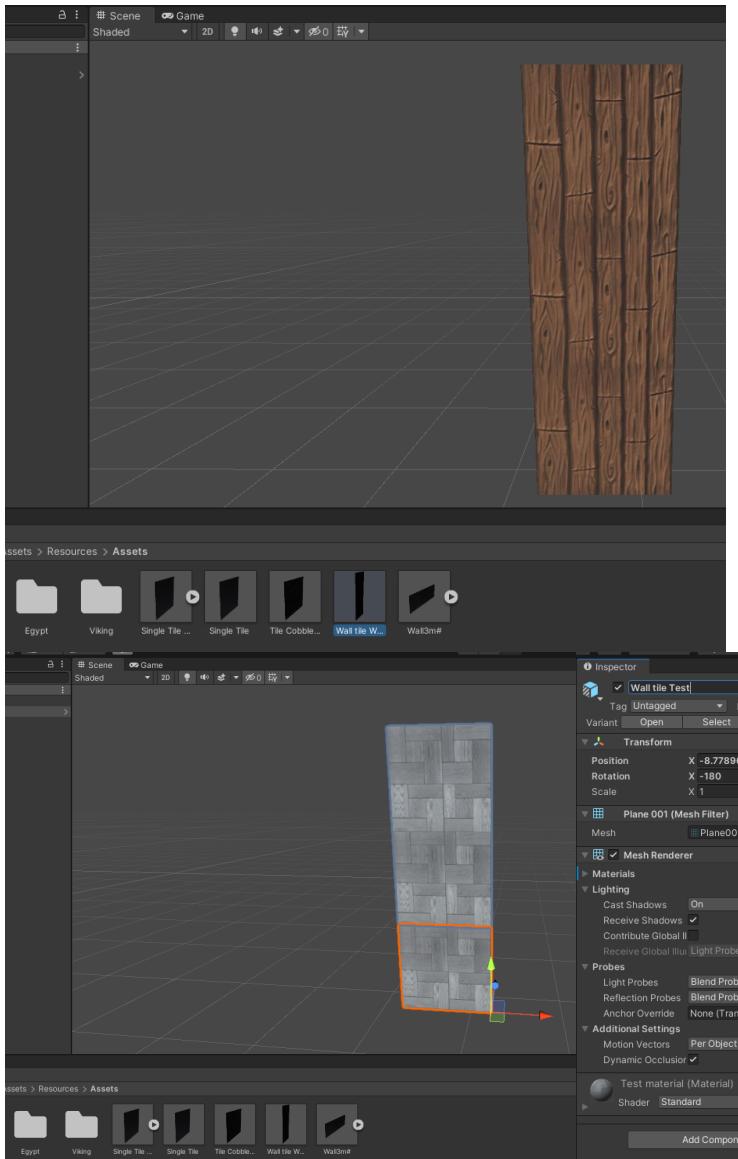
Next, locate the OnEnable() function to find the Material arrays. Here, we assign each material to an array. Adding a material will need the following lines:

```
materialArray[9] = (material)Resources.Load("Materials/Test material");
ColourArray[9] = new Color32(75, 80, 78, 255);
```

The first line will add the material into the array and allow the button to change materials for the drawn tile. The second is to give the drawing window a similar colour to the material when a tile is placed.



Wall Material changes



Wall material changes can happen through Unity easily, or through Unity and changes to code with little effort. Locate the 'Wall tile Wooden' in the Assets folder within the Resources Folder.

Drag the asset into the scene. Here you can click on the Materials folder within the Resources folder and drag a new material onto the asset tiles.

Rename the asset. For example, I have named this one 'Wall tile Test'. In the hierarchy, drag the object and its children into the Assets folder alongside other tiles. A pop up will appear asking you to choose the original prefab or this new variant. Save this as a new variant.

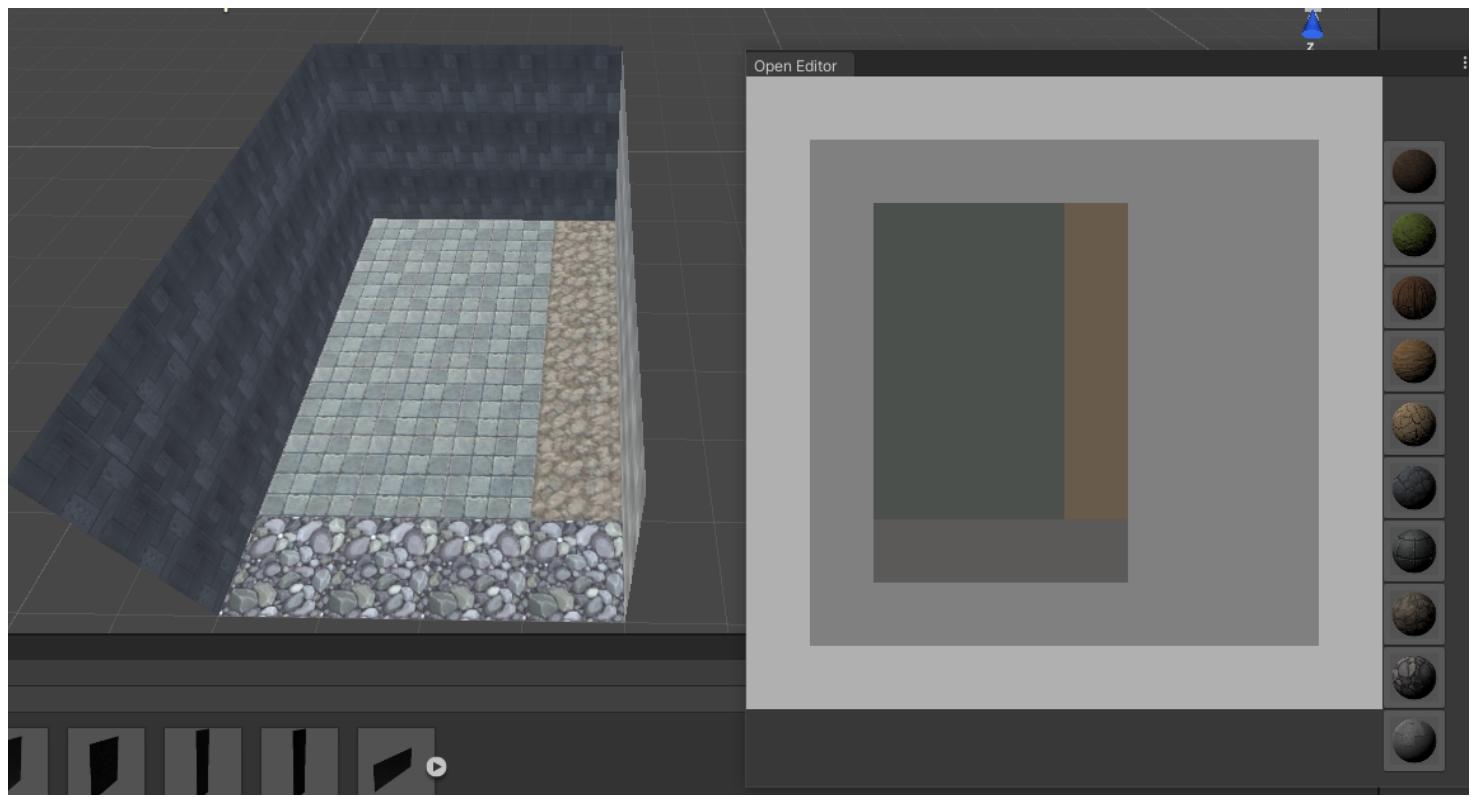
You will notice the name has changed to 'Wall tile Test Variant', you will need this for the script.

Next go back into the script and use Control-F to use the find function. Replace instances of 'Wall tile wooden' with 'Wall tile Test Variant'.

```
if (currentManager.tilesInThisObject[unityxvalue, unityyvalue].GetComponent<Allwalls>().walltiles[1] == null)
{
    GameObject walltile = Instantiate(Resources.Load("Assets/Wall tile Wooden")) as GameObject;

if (currentManager.tilesInThisObject[unityxvalue, unityyvalue].GetComponent<Allwalls>().walltiles[0] == null)
{
    GameObject walltile = Instantiate(Resources.Load("Assets/Wall tile Test Variant")) as GameObject;
```

All walls spawned will now contain the new material.



Jenny Anderson
youtube.com/c/jennyandersoncasualvlogs/videos