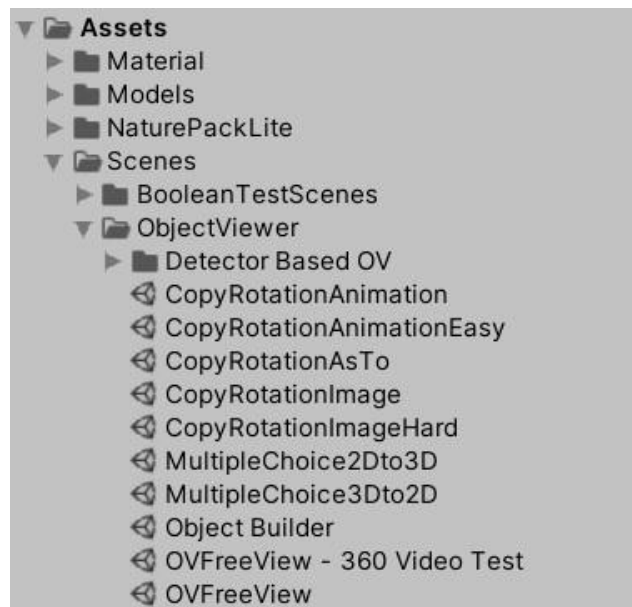


Creating New Object Using Template

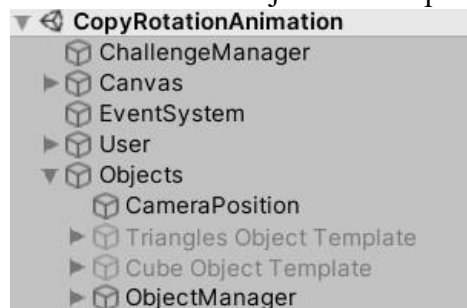
We will use the Object Template in order to easily create a new shape.

File Navigation

In order to move into the directory of the Object Template, go into Assets/Scenes/ObjectViewer and click on “CopyRotationAnimation” while in Unity. It should look like the following picture:



In CopyRotationAnimation, we will choose the Objects. This opens up 4 objects to choose from.

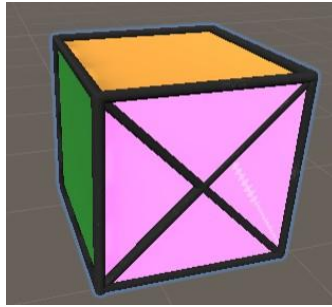
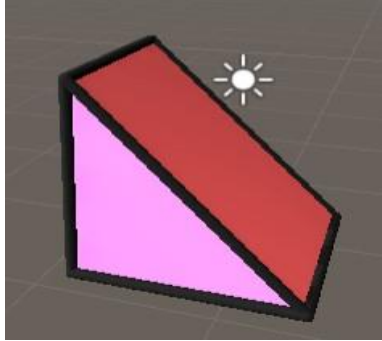


Templates

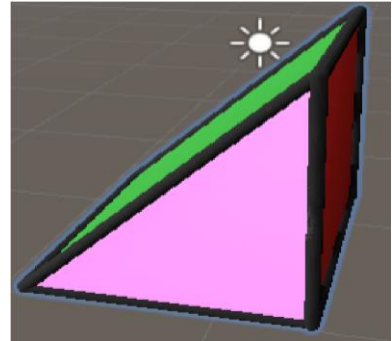
Triangles Object Template - This will be the template to use for creating slants in an object. Users must change the position of the triangle object to merge into their new object. Each triangle represents the different possible positions for it to be implemented into a new shape.

Each 4 of the triangles make up a “cube”

Triangle 1



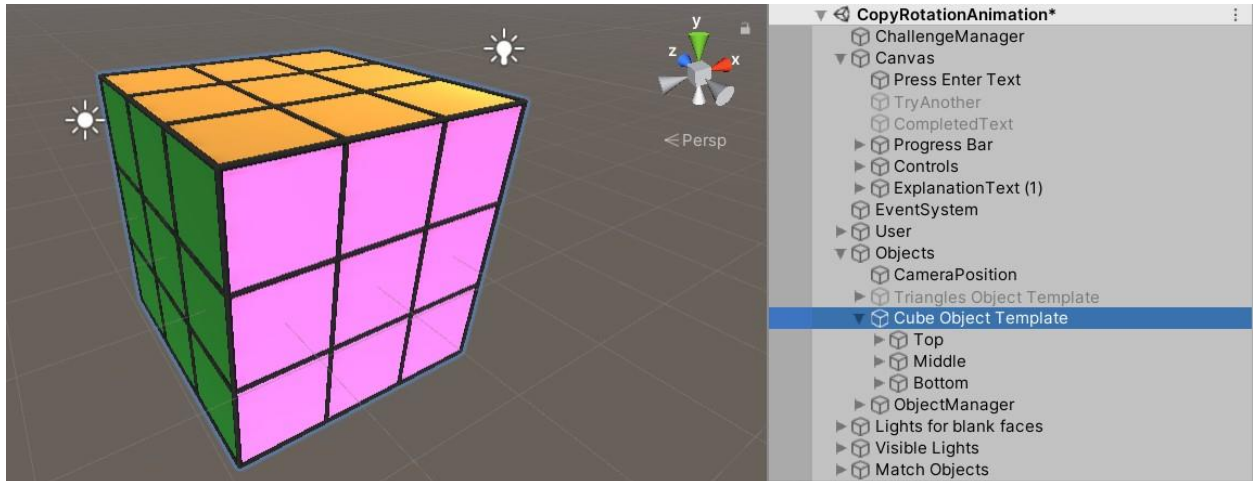
Triangle 2



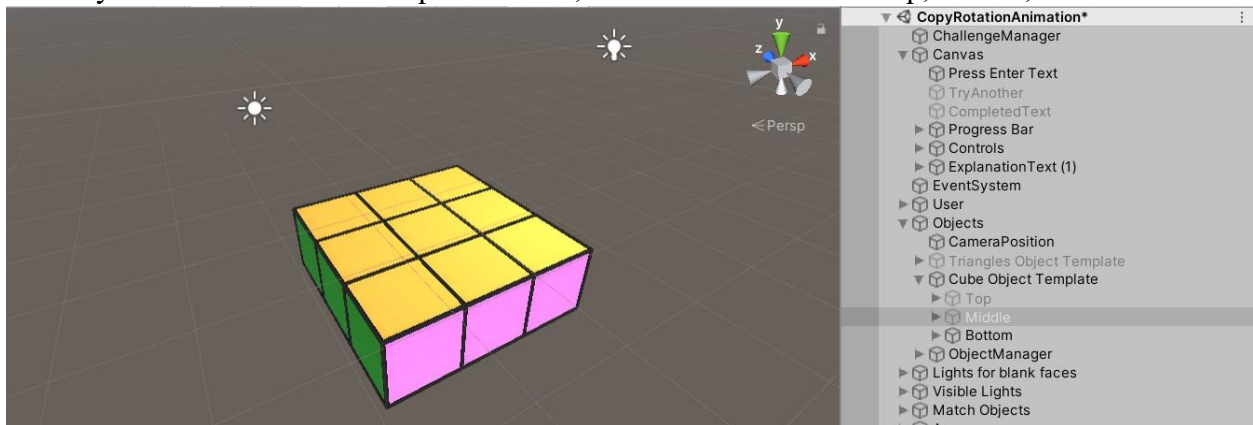
Each triangle piece contains its own edges and faces. This means when merging these pieces into the new shapes, users **must remove** the redundant edges or hidden faces in order to make the shape run smoothly in the program.

Cube Object Template - This will be the template for creating general cube like shapes. To create a new shape, it would be the same as “sculpting” this cube object to get the desired new shape.

Triangles from the previous triangle object template can be added into this template by changing the triangle positions to the replaced cube’s positions.



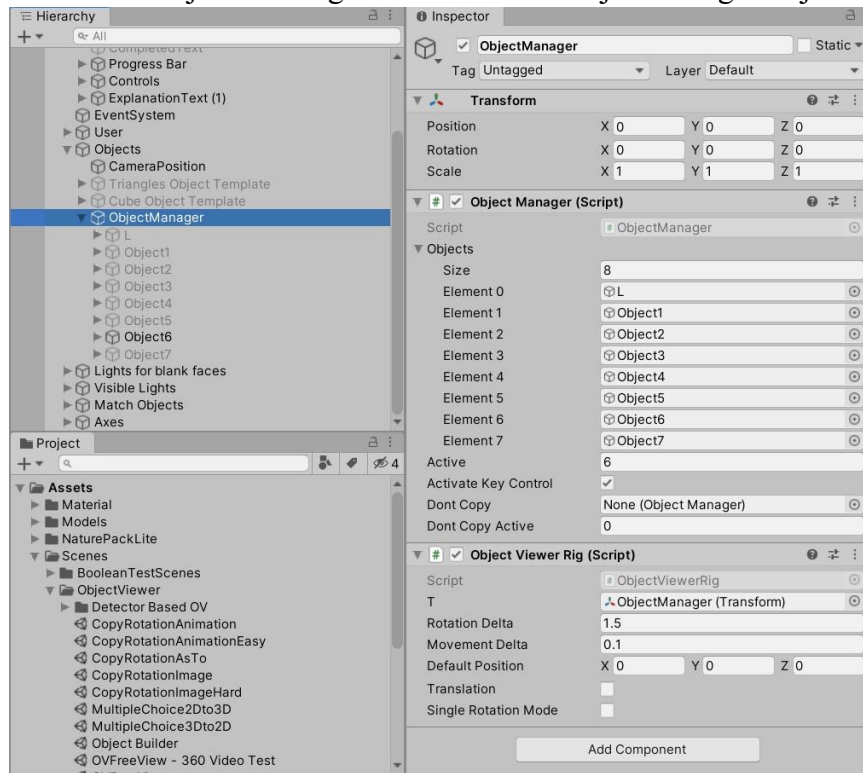
Each layer of the cube is made up of 9 cubes, stacked into 3 rows as top, middle, and bottom.



Each cube piece, like the triangle template, contains its own edges and faces. This means when merging these pieces into the new shapes, users **must remove** the redundant edges or hidden faces in order to make the shape run smoothly in the program.

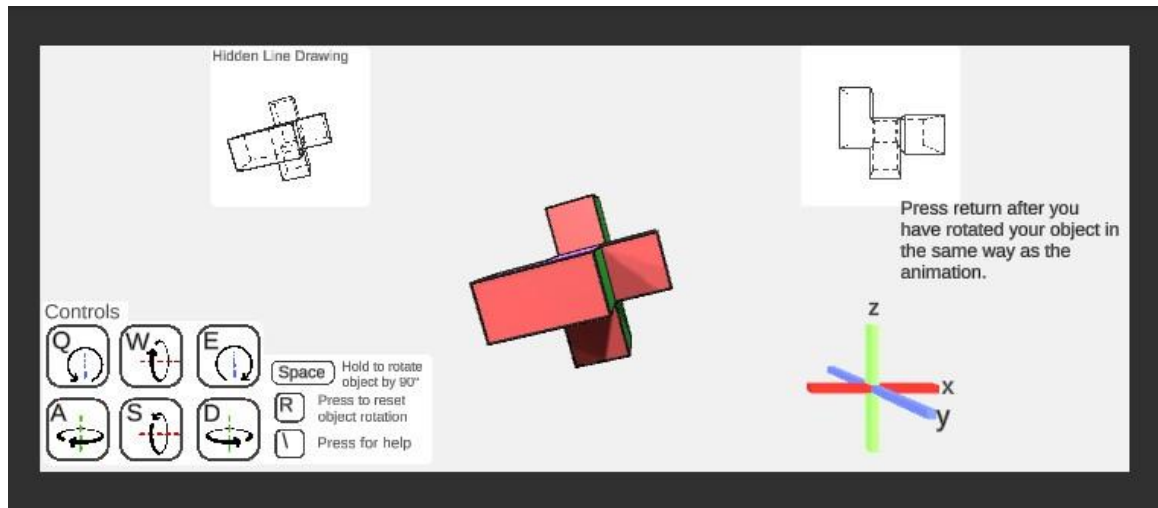
Structure of Program

The main object viewing is located in the ObjectManager object.



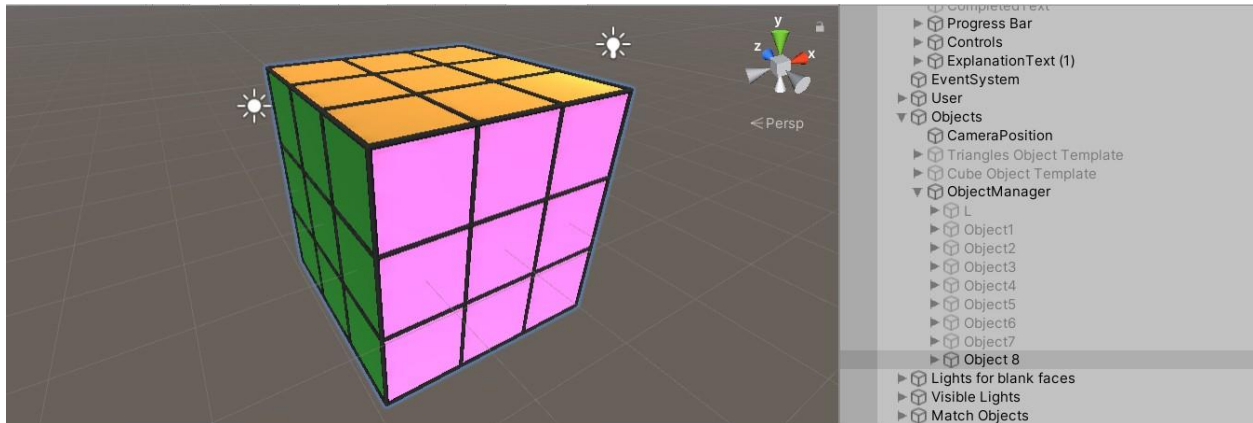
ObjectManager has an ObjectManager Script in the Inspector tab. This contains a global array of elements the script stores. Users will need to **increment** the size of this array to add a new object into the scene. The previous objects are children of the ObjectManger object. In order to add to the array, simply drag the shape you desire into one of the element # array. To test and view the new object, run the game. Choose the number correlated to the shape, in this case 4 was chosen.

This corresponds to Element 4, which can then be rotated using WASD.

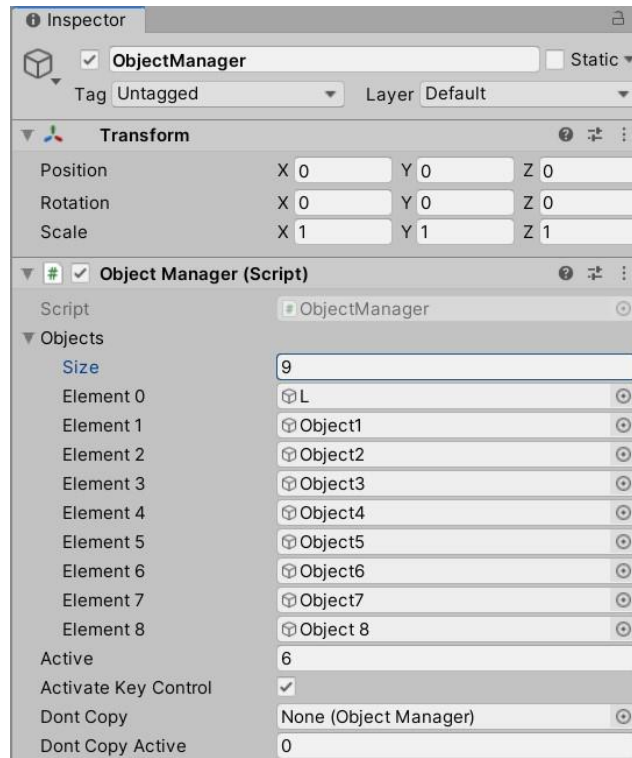


Creating A New Shape

First, make everything invisible and clone the CubeTemplate object into the ObjectManager's child objects. Here it is named Object 8.

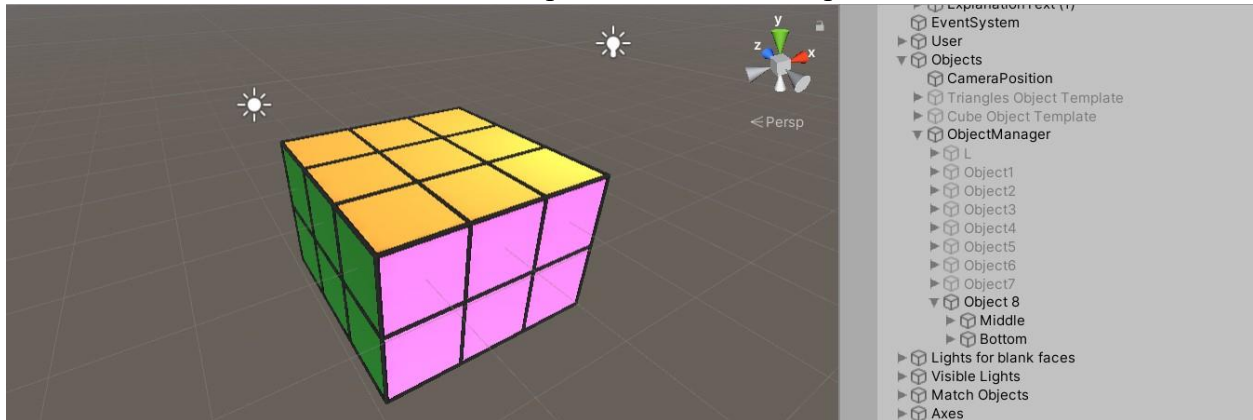


We will go to ObjectManager and increment the array size to 9, and drag Object 8 into the Element 8 slot.

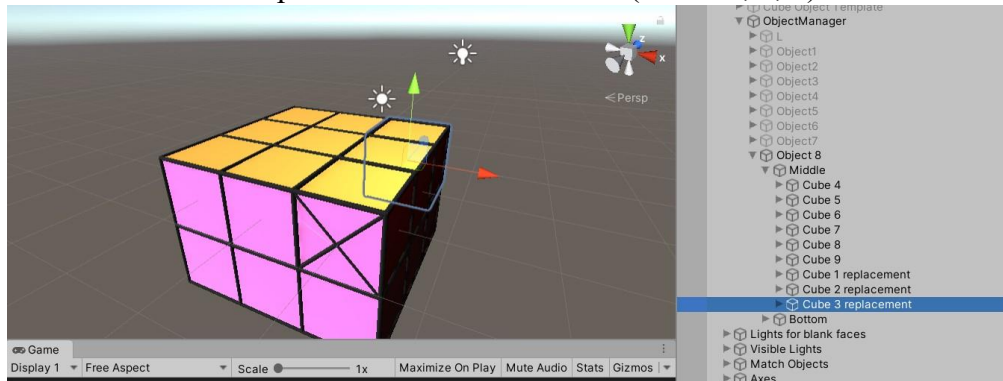


Now sculpting the new object:

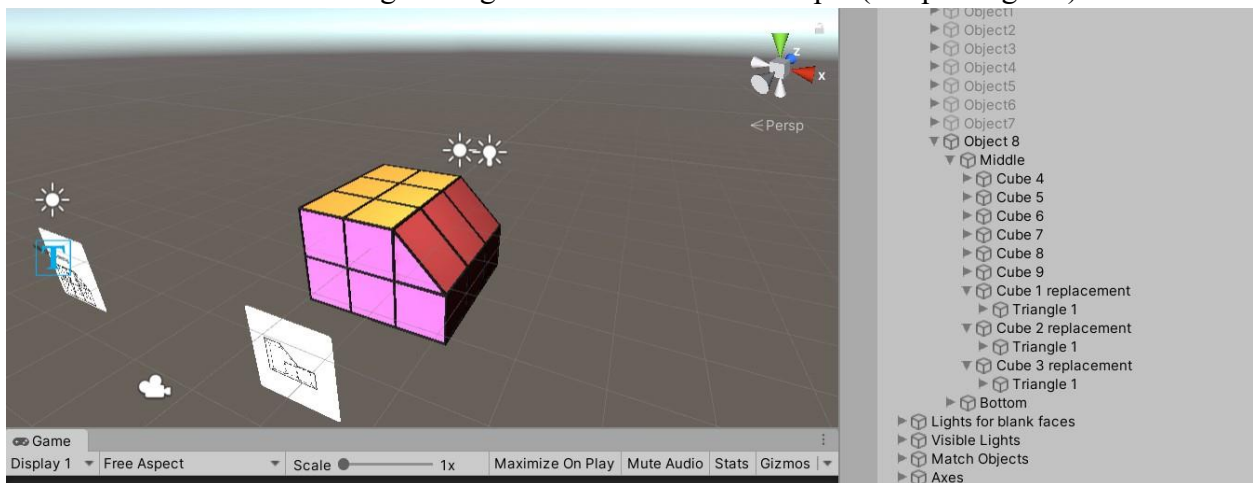
Remove the top row of the cube template.



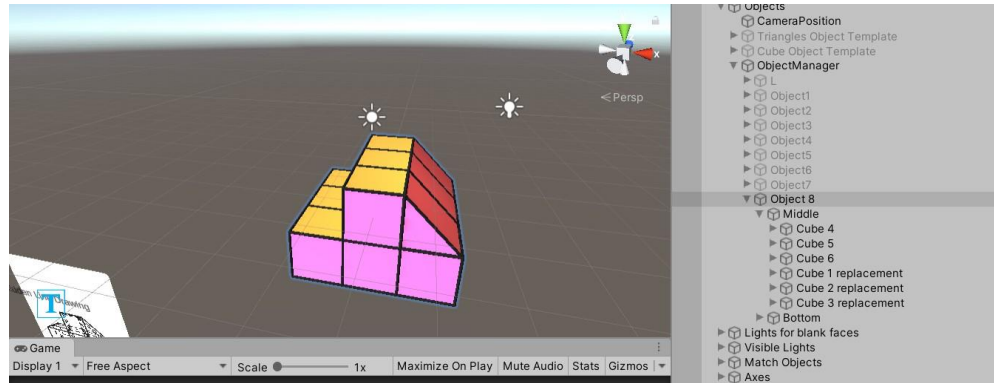
Turn the three pink sided cubes in the middle row into wedges by copying the triangle template and move its position to the three cubes. (Cubes 1, 2, 3)



Choose the triangle/wedge best fit for the new shape. (Keep triangle 1)

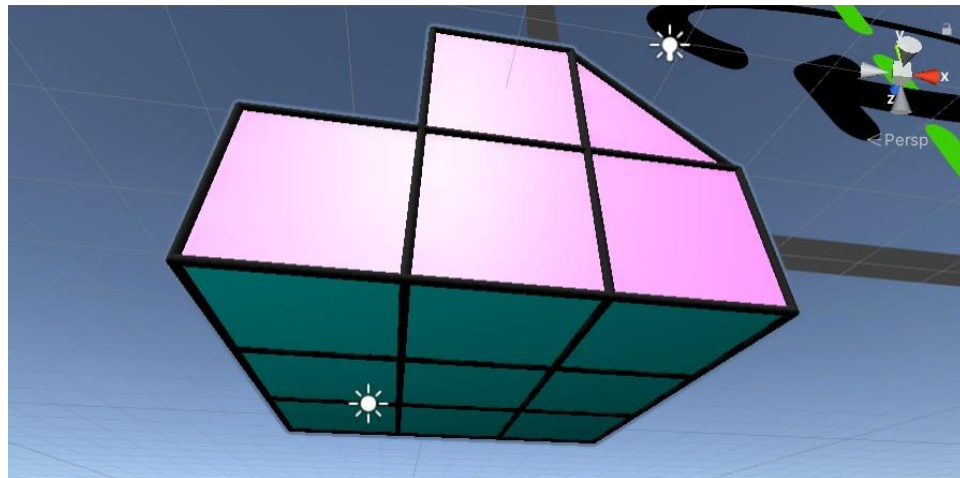


Remove Cubes 7, 8, 9 in the middle row to get the new desired shape

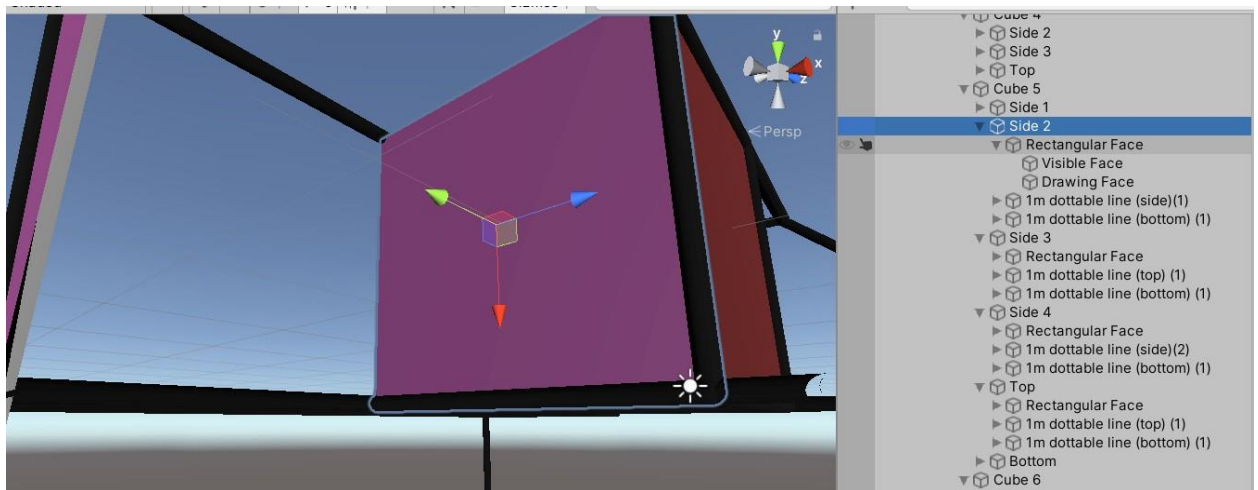
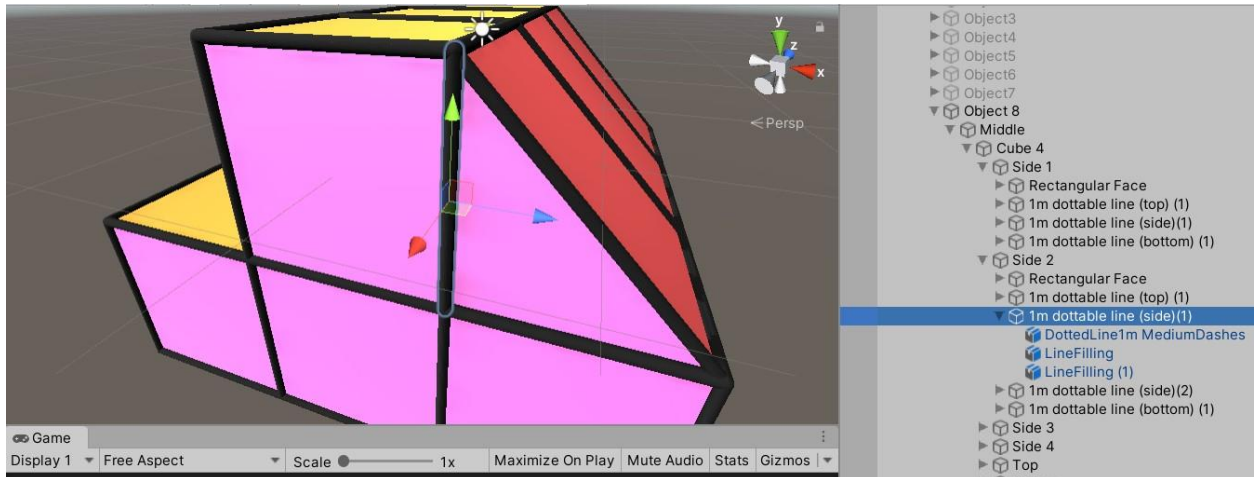


If need be, remove more of the cubes to create a more unique shape. Add in more triangles/edges into the shape for more uniqueness.

Now the last step is removing some of the faces. As mentioned above, every single cube has its own edges. This means in any given edge we see in the object, there will be multiple edges.

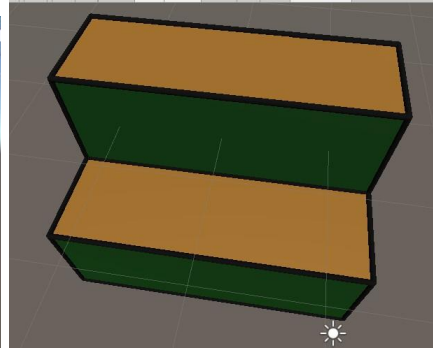
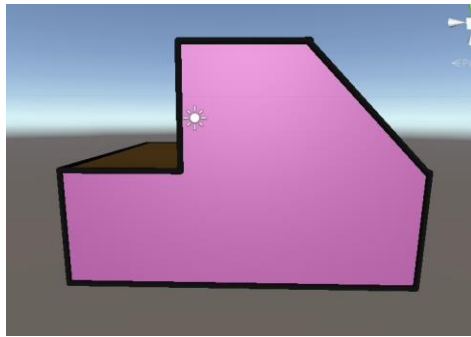


We can remove the edge by deleting the face holding the edges, usually denoted by the parent class “Side”. Delete any of the sides that are not visible to the player. If there are still performance issues, remove the visible edges so only one edge is available per edge. Any connected surfaces should not have an edge, so remove those.



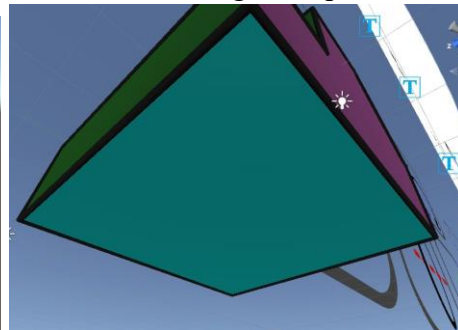
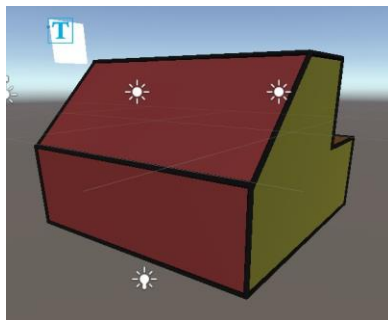
One way is to select the edge on the screen, then move to the parent object of that edge to remove as shown above.

Note: Make sure to move the faces of the sides in order to look merged after removing the edges if necessary, or else you will see unconnected spaces between the sides.



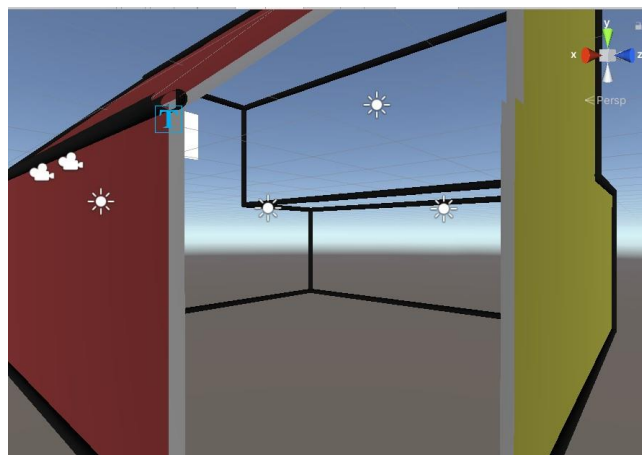
Pink side completed

Green/Orange complete.



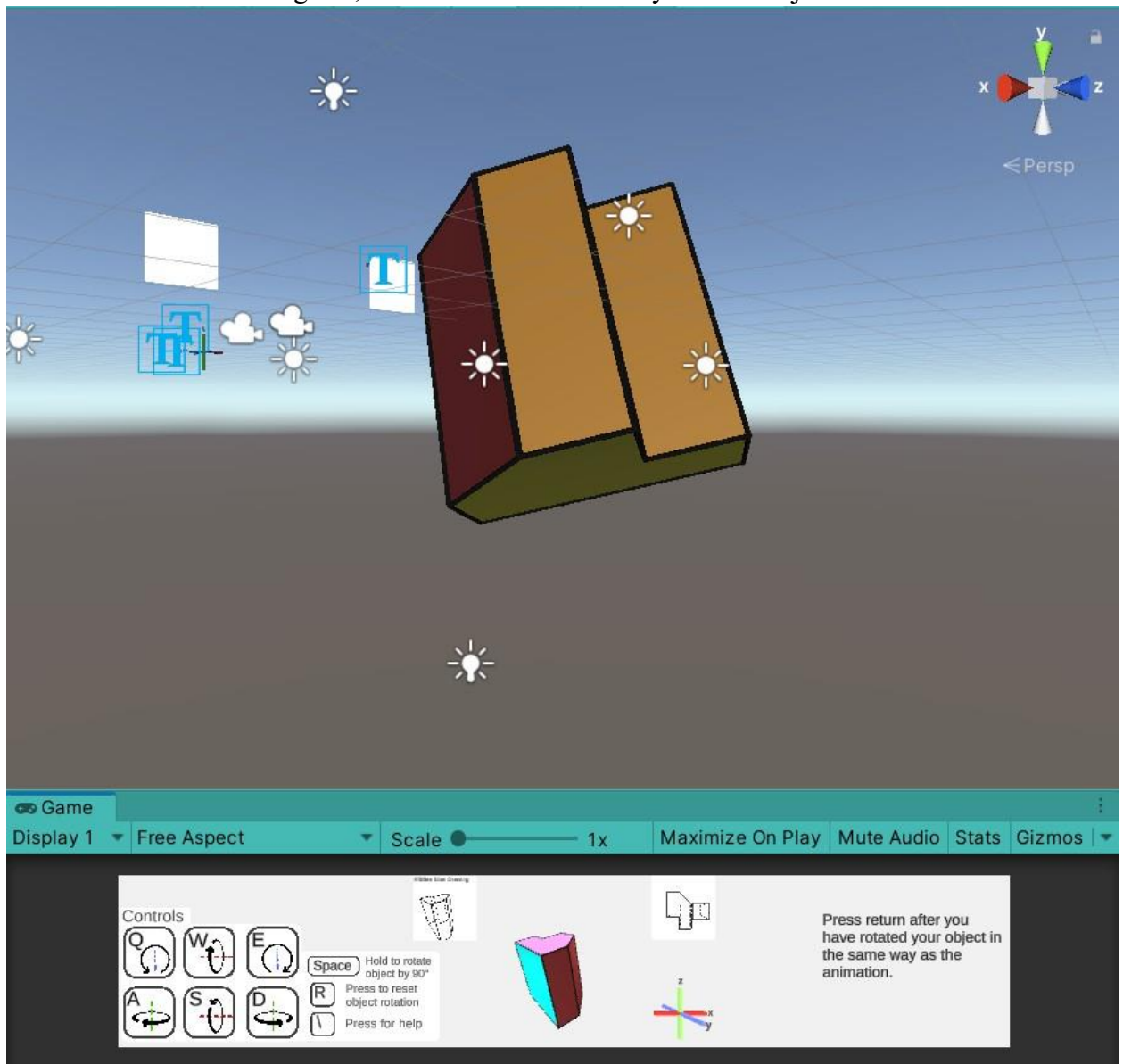
Red/Yellow completed

Bottom side completed

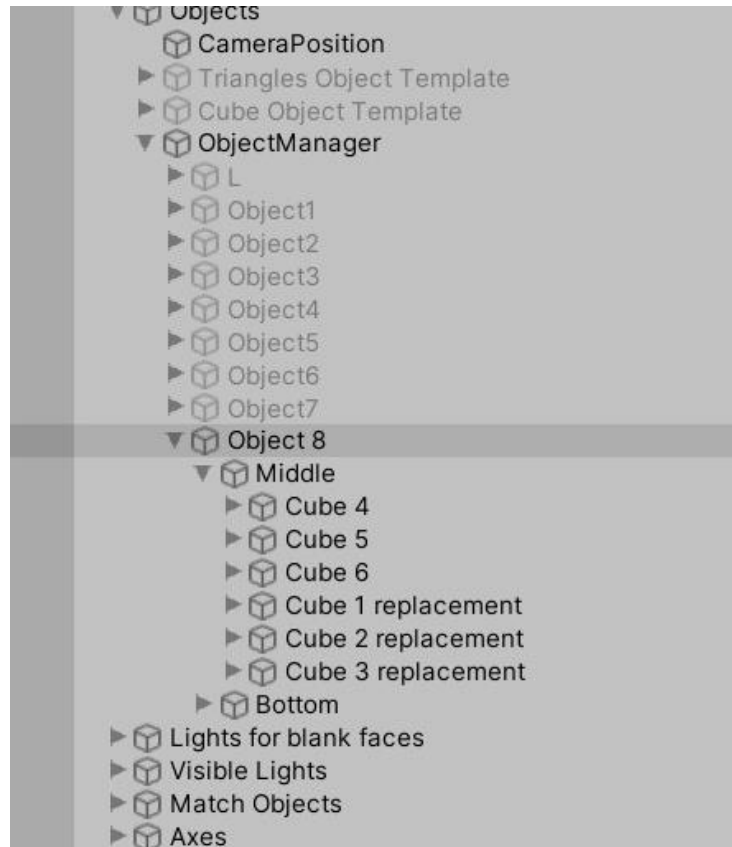


Make sure no invisible sides are left by checking inside the shape

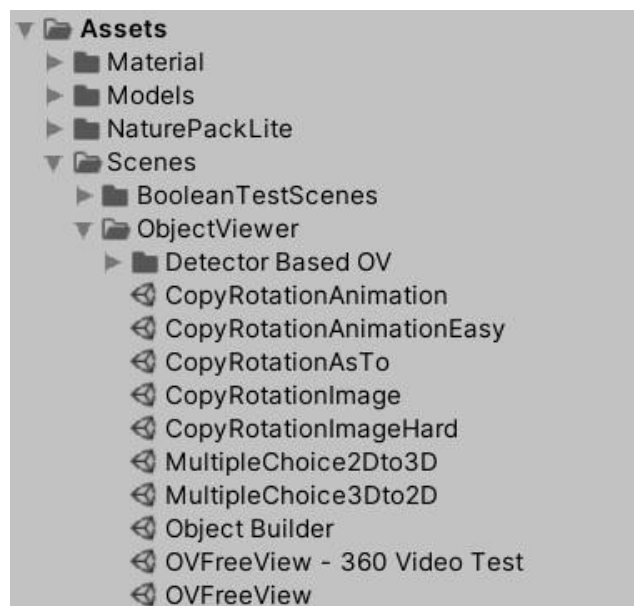
Now run the game, select 9 to view the newly created object!



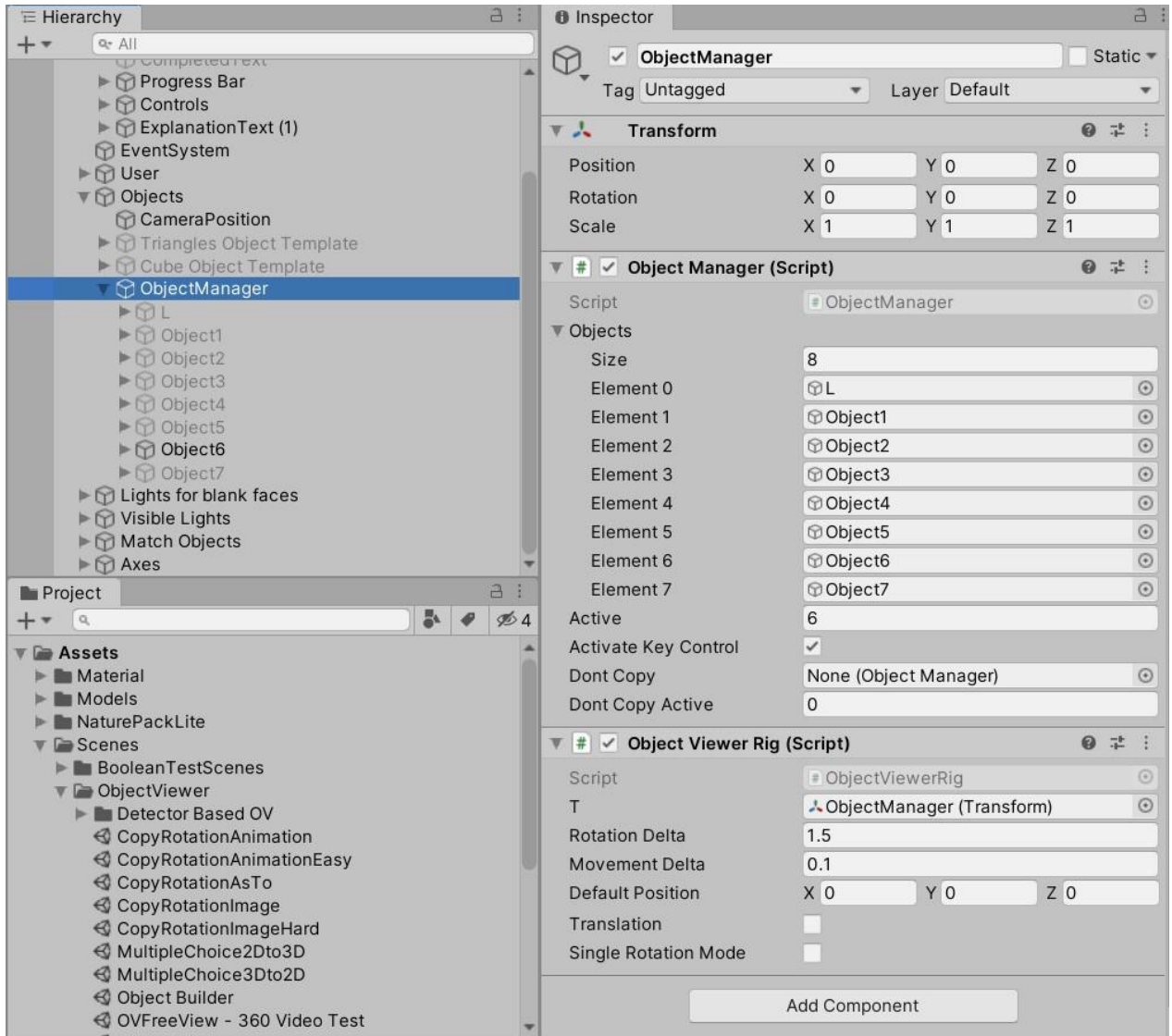
Finally Adding the Object to all Scenes:



Select your newly created object under the hierarchy. On your keyboard press **ctrl + c** on Windows or **cmd + c** on Mac



Under Assets → Scenes → ObjectViewer. Open a new scene.



With the ObjectManager selected in the newly opened scene press **ctrl + v** on Windows or **cmd + v** on Mac to paste your new object into the new scene. Remember to rename your object so that the number in the name is incremented up based on the second most recently added object. For example, if the scene has Object6 as its newest added object please name yours Object7. Repeat the steps found under [Structure of Program](#) to add your object to the ObjectManager for gameplay.