VOLVE STUDIOS

HOLLOW SHAPES 3D Documentation

V 1.0.0

Recommendations/Complaints:
Email us at official.adstudios@gmail.com

TABLE OF CONTENTS

DOCUMENTATION VERSION

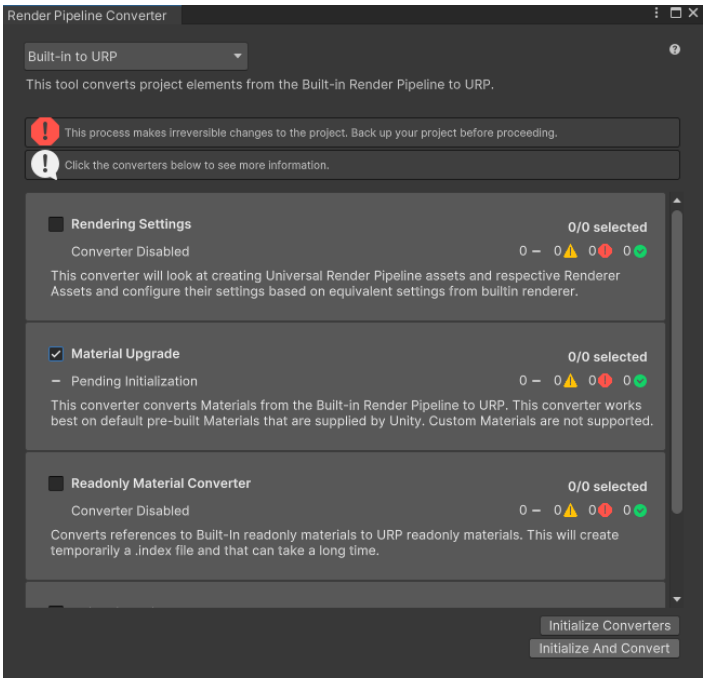
The latest documentation can be found in the link below:
VOLVE Hollow Shapes Documentation.pdf

LICENSE

SETUP

URP/HDRP Fix Materials

From the Unity toolbar, go to Window->Rendering->Render Pipeline Converter. Select Built-in to URP/HDRP in the dropdown menu. Tick the Material Upgrade Option, and at the bottom right corner, click the "Initialize and Convert" button. This will fix the pink material issue.



Asset Requirements

To make sure the asset is working properly, these are vital:

1. Make sure the Shapes3D folder is in “Assets/Plugins” folder.
2. Do not rename or remove any files or folders of the asset
3. Do not modify any prefabs in the asset

Contact us at official.adstudios@gmail.com if you encounter any problems.

Modifying the Default Material

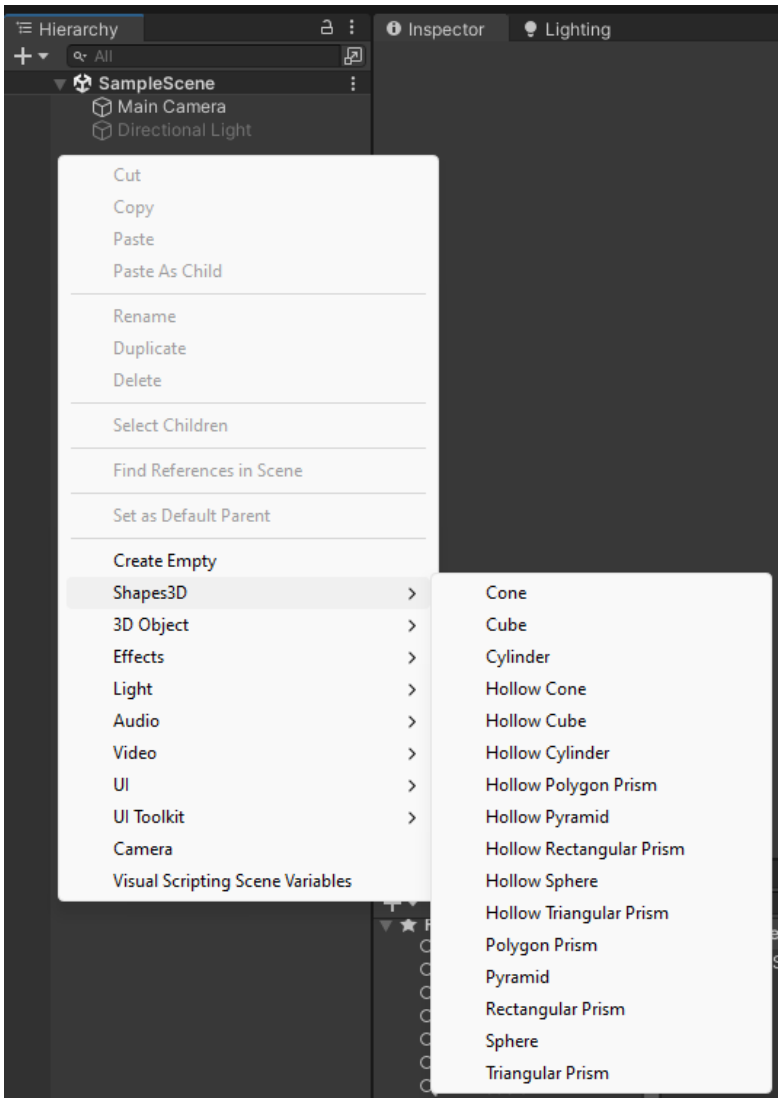
In the Plugins/Shapes3D folder, there is a material named “S3D Default Material”. Modify this material as you like and all the new and old shapes that use this material will be affected.

Creating a Shape

There are two different ways to create a shape:

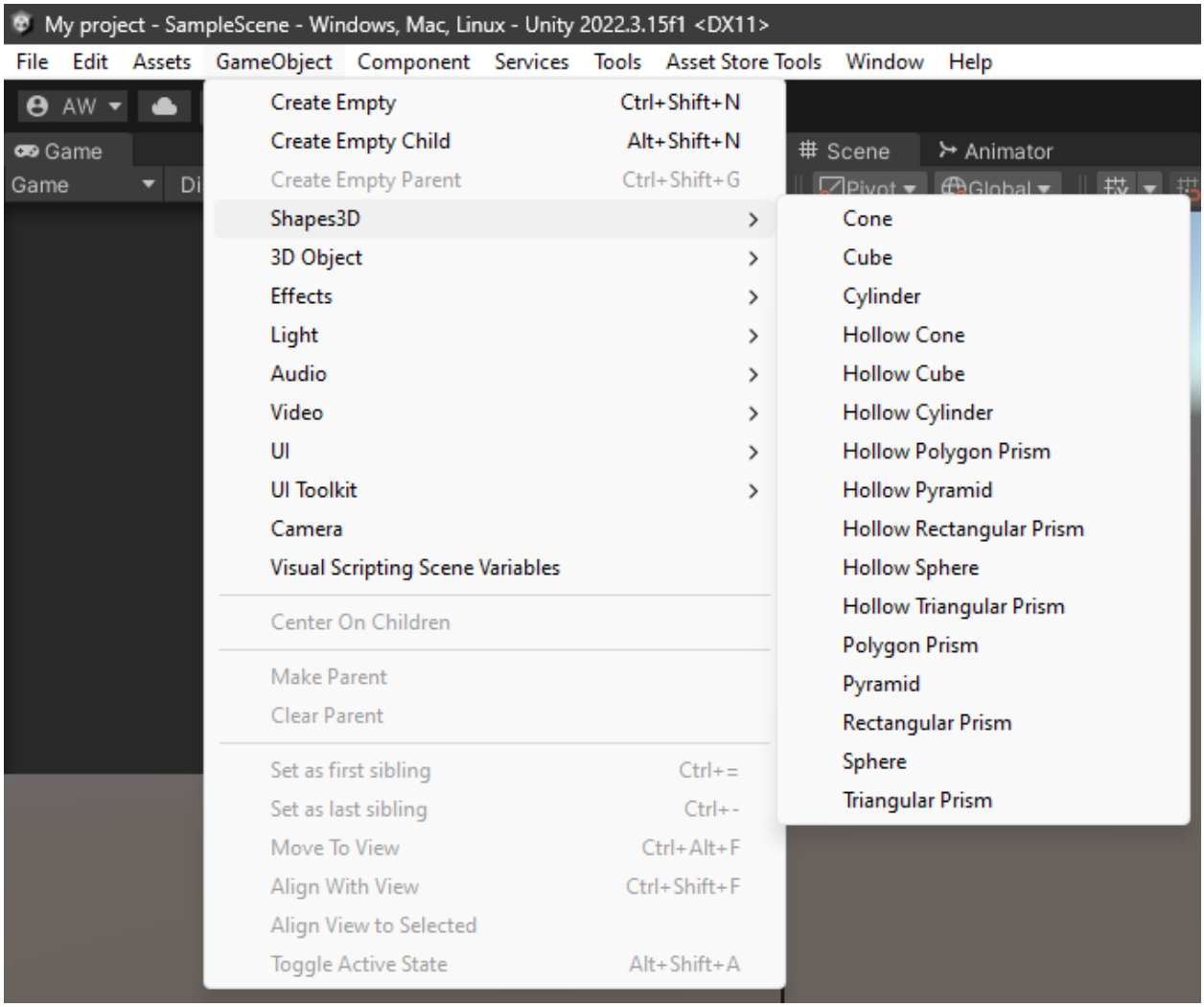
1. Through the hierarchy:

Right click anywhere on the hierarchy -> Shapes3D -> pick any shape



2. Using the toolbar

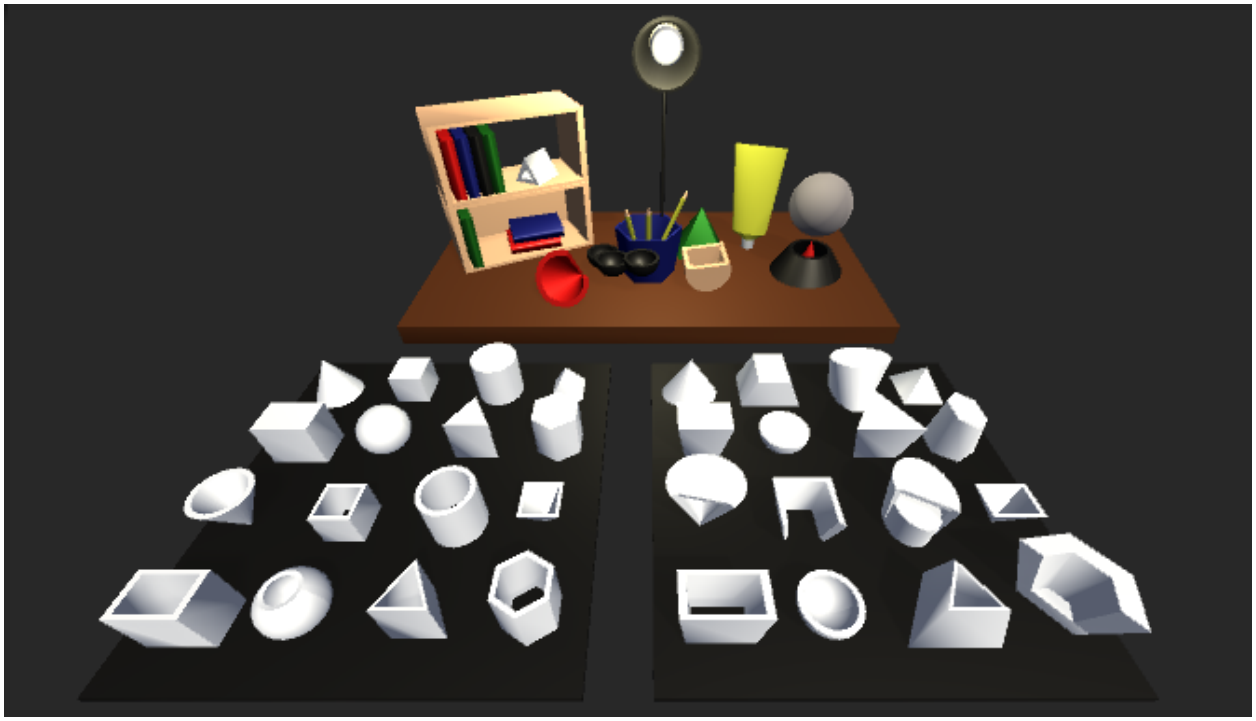
From the toolbar, select GameObject -> Shapes3D -> pick any shape



A new gameobject will be created:

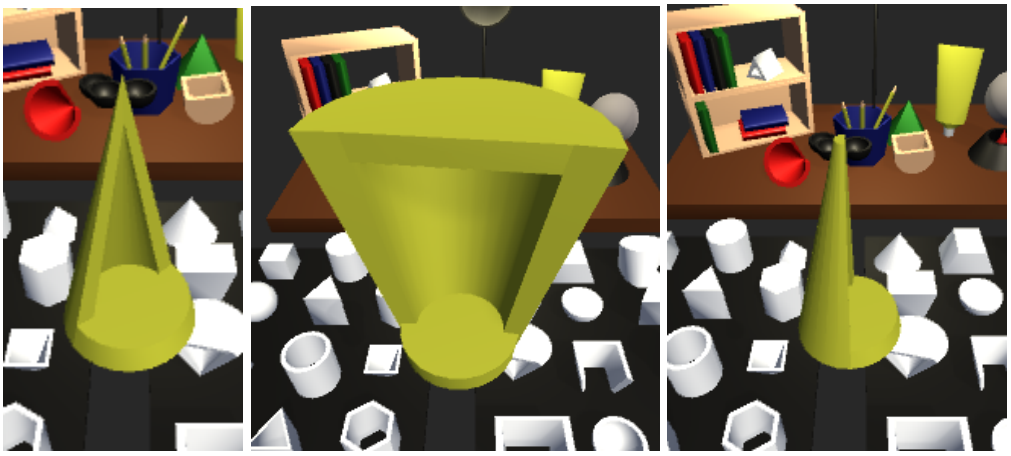
Demo Scenes

In the folder Assets/Plugins/Shapes3D/Demo, there are three demo scenes for you to check out. All of the shapes in the scene are made using this asset. Select any shape and see the details of the shape yourself.



Runtime Code Example

In the demo scene Demo_Runtime, there is the gameObject “RUNTIME SHAPE EXAMPLE”. Select it in the hierarchy, and open up its S3D Demo script to see how to create and modify shapes during runtime, from code. Press the play button to see the code in action.



S3D Shapes

Components that make up S3D Shapes:

1. Mesh Filter - Input of the mesh generated
2. Mesh Renderer - Display the mesh
3. Mesh Collider - Can be disabled if no collision is required
4. S3D Mesh - The main processor for S3DShapes. Require input of a S3D Complete component (either S3DPolygon or S3DSphere)
5. S3DPolygon/S3DSphere - To adjust the details of the shape

The basic details of what makes a shape lies on either the S3DPolygon script or S3DSphere script. Both scripts are mostly similar to each other.

Shapes that are governed by S3DPolygon:

1. Cone
2. Cube
3. Cylinder
4. Hollow Cone
5. Hollow Cube
6. Hollow Cylinder
7. Hollow Polygon Prism
8. Hollow Pyramid
9. Hollow Rectangular Prism
10. Hollow Triangular Prism
11. Polygon Prism
12. Pyramid
13. Rectangular Prism
14. Triangular Prism

Shapes that are governed by S3DSphere:

1. Sphere
2. Hollow Sphere

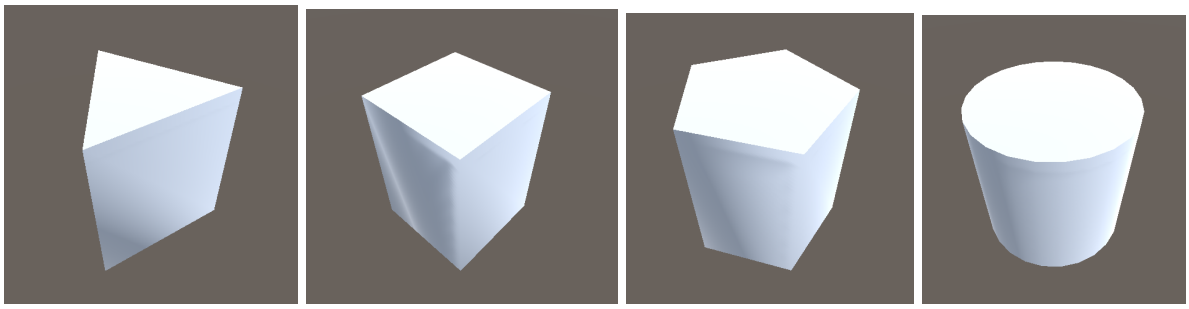
S3D Polygon

The inspector window of S3D Polygon script is as follows:

All of the 14 non-spherical shapes mentioned in the previous page can be derived by adjusting the parameters of S3D Polygon.

Vertex

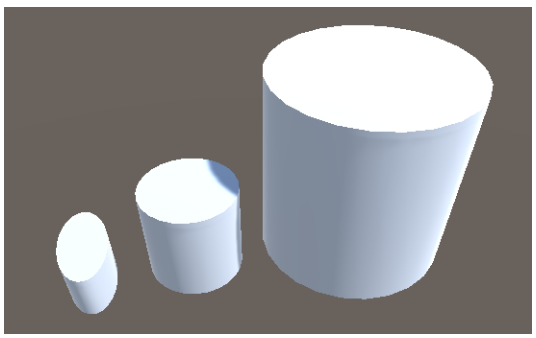
Controls the number of vertices the bottom and top faces will have. e.g. 3 for a triangle, 4 for a square, 5 for pentagon and more for a circle.



Vertex = 3, 4, 5, 25 respectively

Scale

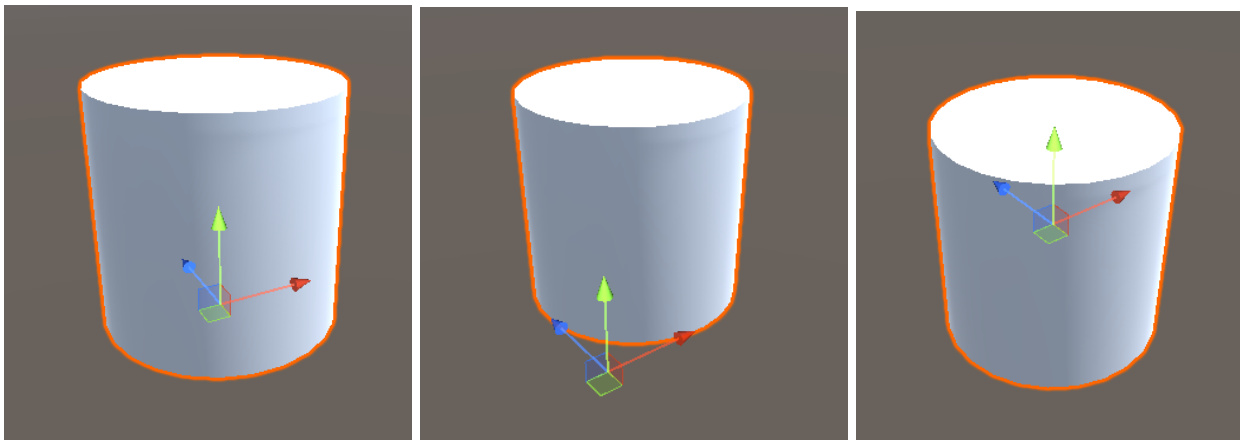
Adjusts the scale of the shape in the respective axes, respective of the pivot point.



[From left: (0.5,1,1), (1,1,1) & (2,2,2)] (Scale.x, Scale.y, Scale.z)

Pivot Point

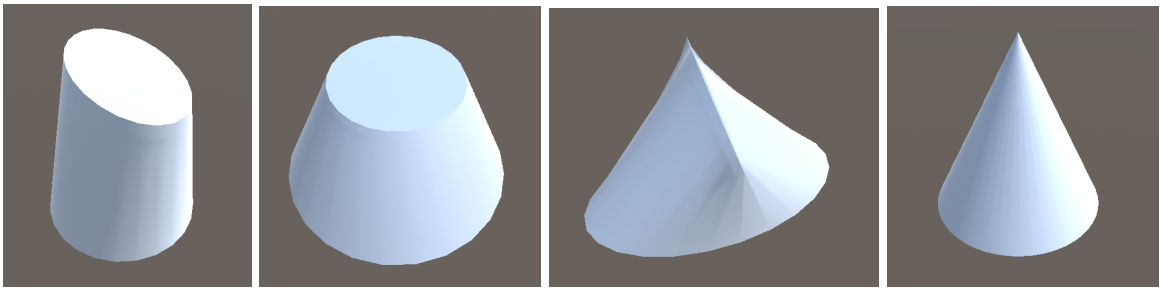
Ranges from -0.5 to 0.5 for each axis. Controls where the positioning, scaling and rotation will take place.



[From Left: (0,-0.5,0), (-0.5,-0.5,-0.5) & (0,0,0)]
(PivotPoint.x,PivotPoint.y,PivotPoint.z)

Taper

Ranges from 0 to infinity. Only applicable to x and z axes. If below than 0, the shape will taper inward. Greater than 0 will taper outward. Set taperX = 0 and taperZ = 0 at one side to get a cone shape.



From Left: Scenario 1, Scenario 2, Scenario 3 & Scenario 4

Scenario 1:

Taper for top: (0.5,1,1) Taper for Bottom: (1,1,1)

Scenario 2:

Taper for top: (1,1,1) Taper for Bottom: (2,2,2)

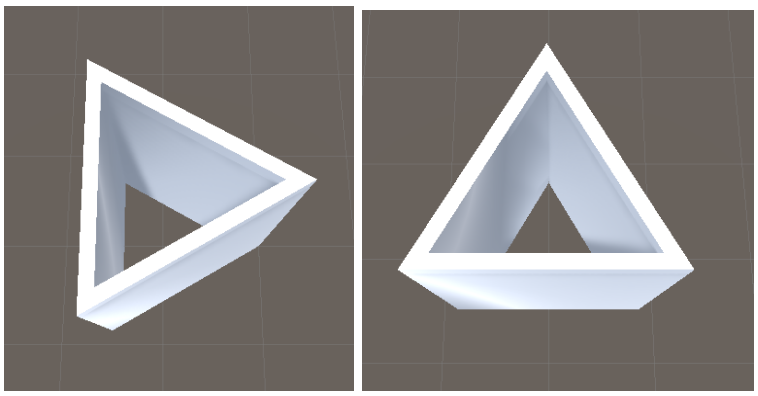
Scenario 3:

Taper for Top: (0,1,1) Taper for Bottom: (2,1,1)

Scenario 4:

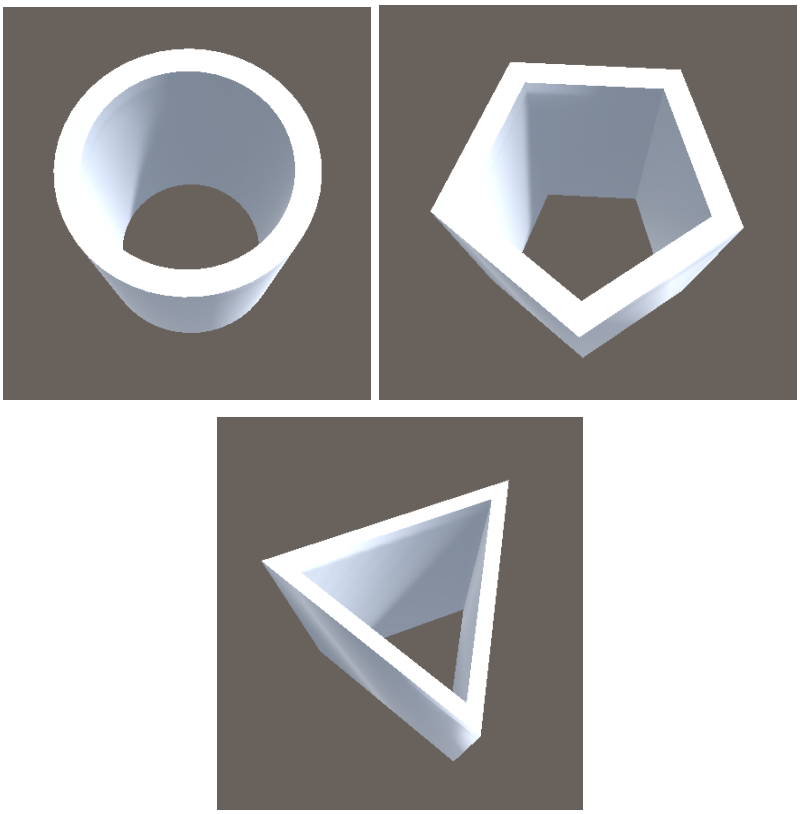
Taper for Top: (0,1,0) Taper for Bottom: (1,1,1)

Rotate by PI radians
Rotate the object by the (input value)*PI radians. For a triangle, set it to 1/6 to make it perfectly straight. For a square, set to 1/4.



Rotate by PI radians = 0 & 1/6 respectively

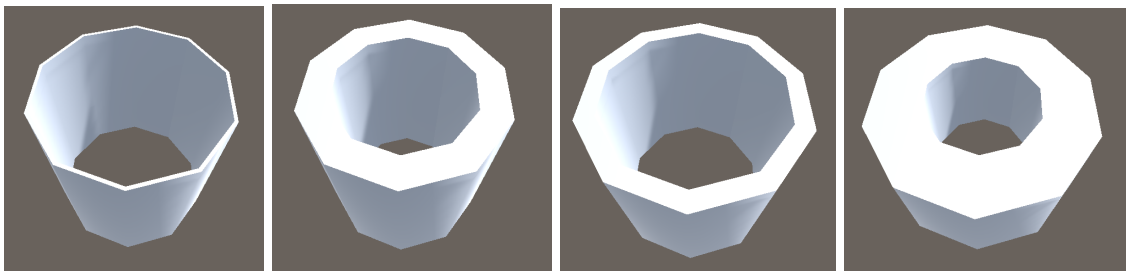
Hollow
Enable this to make the shape hollow.



Additional settings for Hollow

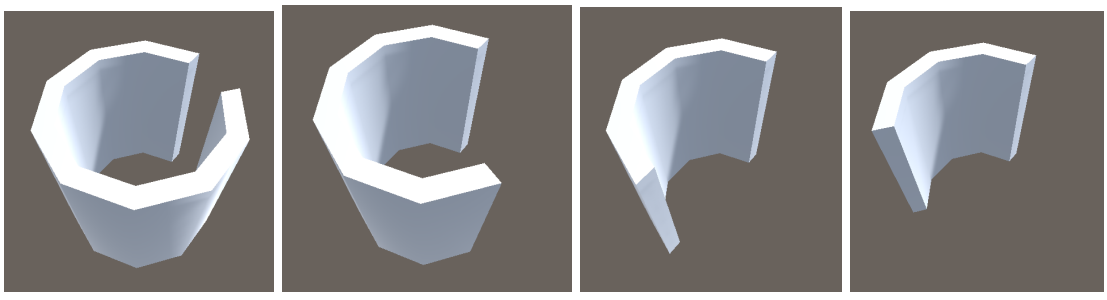
Once you enable hollow, the following settings will show up.

1. **Thickness**
   Adjusts the thickness of the walls of the object. Set above 0 to make inner walls, and below 0 to make outer walls.



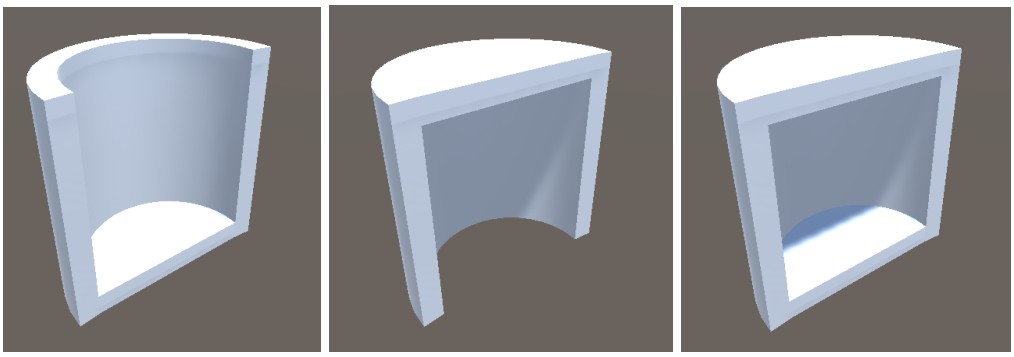
Thickness = 0.02, 0.15, -0.1 & -0.45 respectively

2. **Side Range**: Opens up the sides of the shape. Set equal to vertex/2 to perfectly cut the object in half.



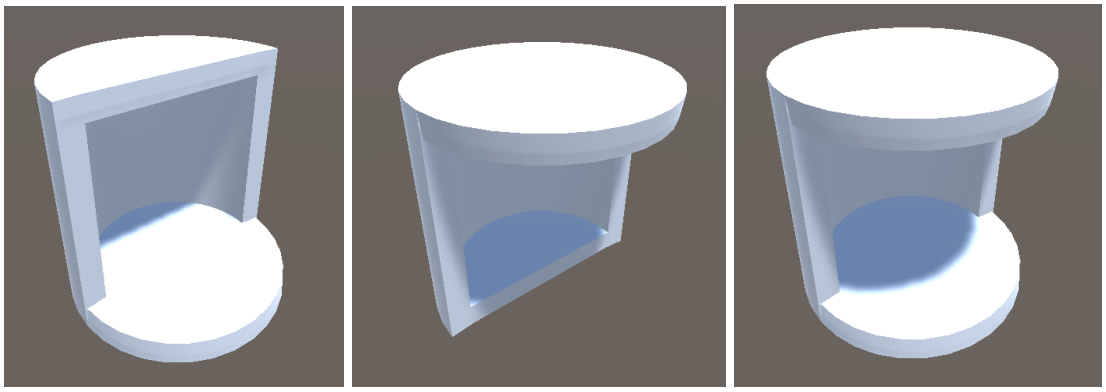
(Vertex = 10) Side Range = 9, 7, 5 & 4 respectively

3. **Cap Thickness**: Create a top/bottom cap to enclose the top/bottom part of the object.



From left: Bottom cap thickness = 0.1,
Top cap thickness = 0.1,
Bottom cap thickness = 0.1; Top cap thickness = 0.1

4. **Full Top/Bottom Cap**: Appear only when top/bottom cap > 0. Enable to get a full uncut top/bottom cap.



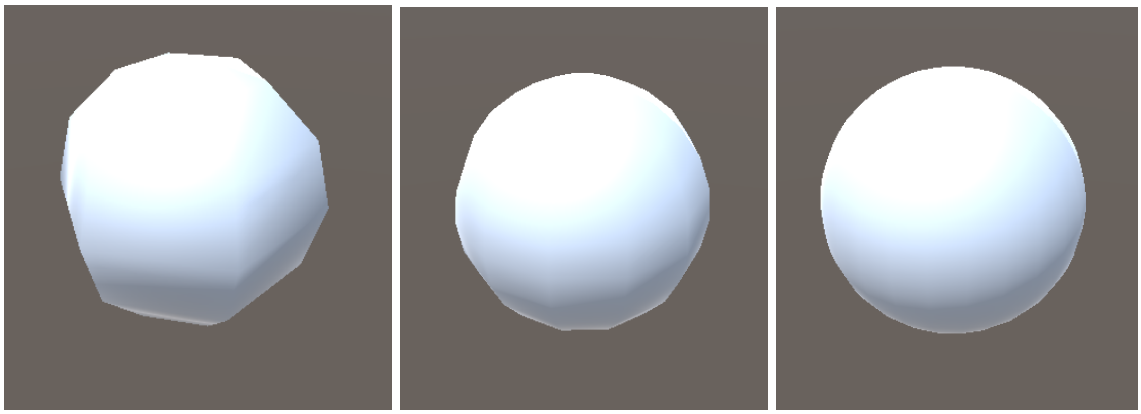
From left: Full bottom cap enabled,
Full top cap enabled,
Full bottom cap & Full top cap both enabled

S3D Sphere

The inspector window of S3D Polygon script is as follows:

Resolution

Determines how many vertices and triangles are created for the shape. The higher the number, the smoother the sphere.



Resolution = 6, 12 & 25 respectively

Scale

Adjusts the scale of the shape in the respective axes, respective of the pivot point.

Pivot Point

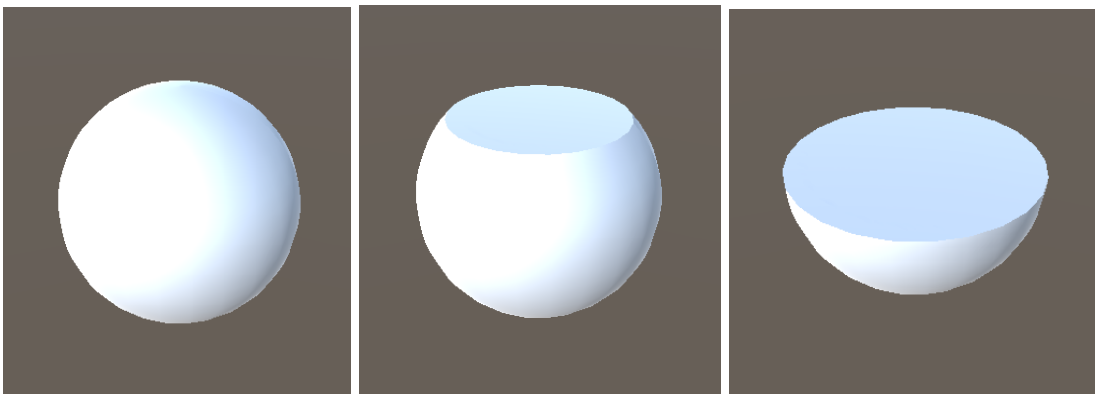
Ranges from -0.5 to 0.5 for each axis. Controls where the positioning, scaling and rotation will take place.

Radius

Controls the radius of the sphere.

Height

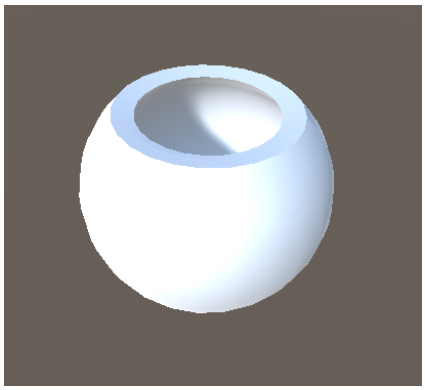
Cuts of the shape based on the height. Put 1 for a full sphere, 0.5 for a semi-sphere.



Height = 1, 0.75 & 0.5 respectively

Hollow

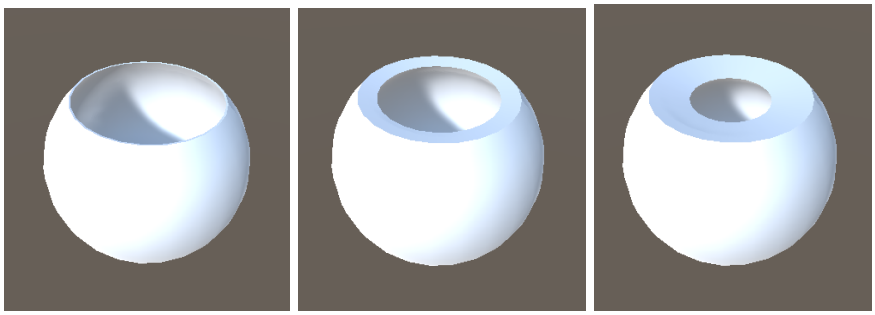
Enable to make the shape hollow. Adjust the height or thickness to see effect.



Hollow sphere with height = 0.75

Additional settings for hollow

1. **Thickness**: Adjusts the thickness of the walls of the object.



Thickness = 0.01, 0.1 & 0.25 respectively

Creating and Modifying from Script

Use the namespace S3DShapes to gain access to two of the important classes: S3DPolygon and S3DSphere.

```
using S3DShapes;
```

Creating the shape

You can create a new gameobject of type S3DPolygon/S3DSphere by Instantiating the prefab of a desired shape from Assets/Plugins/Shapes3D/Prefabs folder. Drag and drop the prefab into the inspector.

```
public GameObject shape;
void Create(){
        GameObject shapeGameObject = Instantiate(prefab);
}
```

Modifying the shape

You can set the parameters of the shape by using these methods:

S3DPolygon methods

- .SetVertex(int numberOfVertex)
- .SetScale(Vector3 scale)
- .SetPivotPoint(Vector3 pivotPoint)
- .SetTaperTop(Vector3 taperTop)
- .SetTaperBottom(Vector3 taperBottom)
- .SetHollow(bool hollow)
- .SetThickness(float thickness)
- .SetSideRange(int sideRange)
- .SetTopCapThickness(float topCapThickness)
- .SetBottomCapThickness(float bottomCapThickness)
- .SetFullTopCap(bool enableFullTopCap)
- .SetFullBottomCap(bool enableFullBottomCap)
- .SetRotateByPIRadians(float numPIRadians)

S3DSphere methods

- .SetResolution(int resolution)
- .SetScale(Vector3 scale)
- .SetPivotPoint(Vector3 pivotPoint)
- .SetRadius(float radius)
- .SetHeight(float height)
- .SetHollow(bool enableHollow)
- .SetThickness(float thickness)

After modifying any parameters, the shape will not be updated until the .Recreate() method is called. So, it is best practice to call the .Recreate() method at the very last after all wanted parameters are modified.

+ Calling .Recreate() inside Update() method or any other update methods are not advisable. Each time it is called, .Recreate() method will calculate and generate a new mesh based on the parameters set. Calling it in every frame is performance-intensive and not recommended.

Code Example:

```
void Modify(){
        //Get the S3DPolygon component
        S3DPolygon shape = shapeGameObject.GetComponent<S3DPolygon>();
        shape.SetPivotPoint(Vector3.zero);
        shape.SetTaperTop(2f * Vector3.one);
        shape.SetTaperBottom(0.5f * Vector3.one);
        shape.SetHollow(true);
        shape.SetThickness(0.15f);
        shape.SetSideRange(13);
        shape.SetTopCapThickness(0.1f);

        //you can also stack the method calls in one line like this:
        shape.SetBottomCapThickness(0.1f).SetFullBottomCap(true);

        //Apply changes by calling .Recreate
        shape.Recreate();
}
```

Demo Scene Code Example
In the demo scene Demo_Runtime, there is the gameObject “RUNTIME SHAPE EXAMPLE”. Select it in the hierarchy, and open up its S3D Demo script to see how to create and modify shapes during runtime, from code. Press the play button to see the code in action.

USEFUL TIPS

1. Put shapes next to each other, select them in the hierarchy, right click -> create an empty parent to combine them under a single parent.
2. Set pivot point Y to -0.5 for easier placement on the floor.
3. Rotation is in respect to the pivot point. So set the pivot point on where you would like the shape to rotate from.
4. All shapes come with a mesh collider. Disable the collider if you don't need it.
5. The Transform component in the inspector works as usual. You can use it to move, rotate, and scale as usual.
6. You can find generated meshes in the GeneratedMesh folder. Useful if you have the tools to convert it to .fbx or .obj files to edit externally.

ADDITIONAL NOTES

1. Do not remove any native folders in the Shapes3D folder, otherwise you will most likely get errors.
2. We would appreciate it if you could leave a kind review in the Unity Asset Store. It helps us out a lot.
3. Before leaving any bad reviews, we would appreciate it if you could email us at official.adstudios@gmail.com first for any problems or complaints. We will try to sort it out.
4. Email us at **official.adstudios@gmail.com** for any recommendations.

- DOCUMENTATION ENDS -

THANK YOU FOR USING HOLLOW SHAPES 3D