Highlighter

Highlighter is a post processing effect. It has some predefined effects to highlight game objects directly in your game view.

Getting Started:
You have to follow only three major steps listed below to complete the setup process of “Highlighter” in your project.

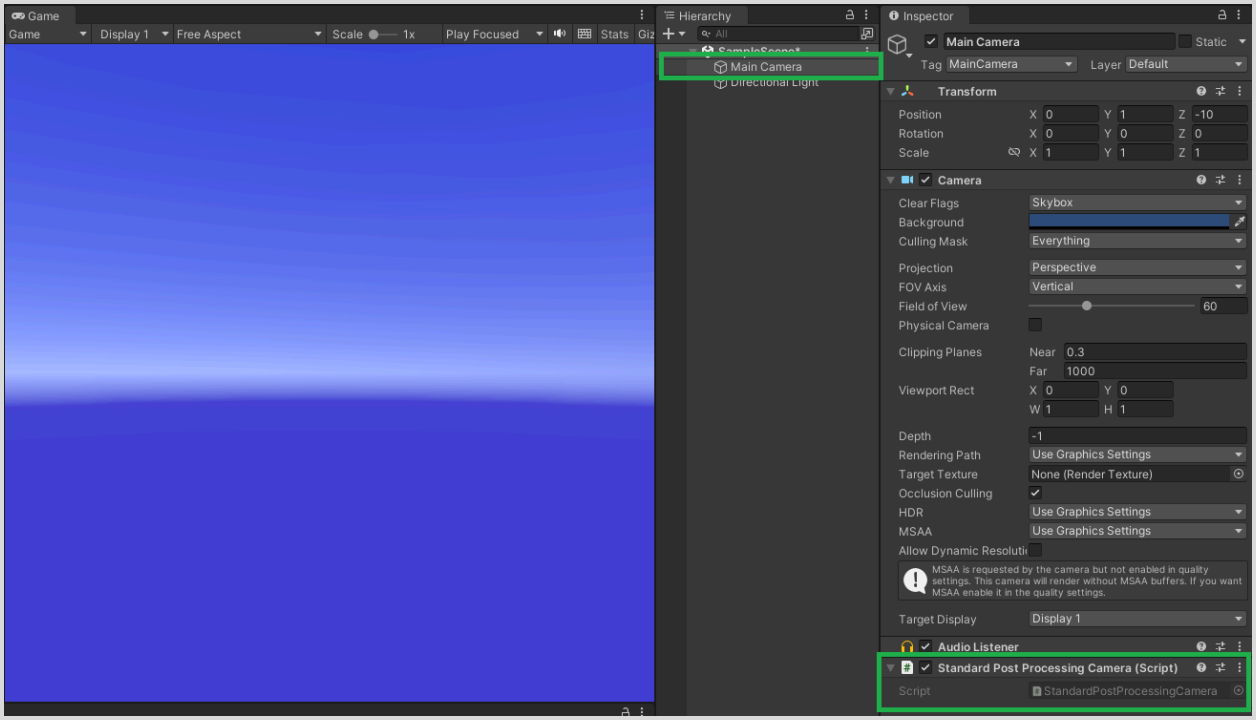
Step 1: Setup Highlighter with Render Pipeline.
Step 2: Add Highlighter Manager Prefab in the Scene.
Step 3: Add Highlight effect Component on gameobject.

First step setup the “**Highlighter**” depends on your projects render pipeline and second step is add the **Highlighter Manager** prefab in the scene to change the background fill color from the inspector and third step is add the Highlight effect component with the game object.

Step 1: Setup Highlighter with Render Pipeline

Step 1.1: Built-in Render Pipeline Setup

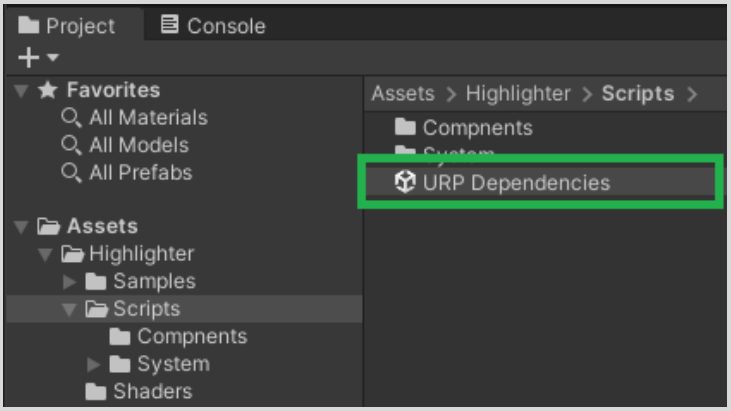
Select the **Main Camera** from the scene and add “**StandardPostProcessingCamera**” Component then you will be able to see some background color automatically applied in your game view.
*Note: The “**StandardPostProcessingCamera**” Component works only with the Built-in render pipeline. If you are using only the Built-in Render Pipeline in your project please move to **Step 2** otherwise if you are using Universal Render Pipeline please follow **step 1.2**.*



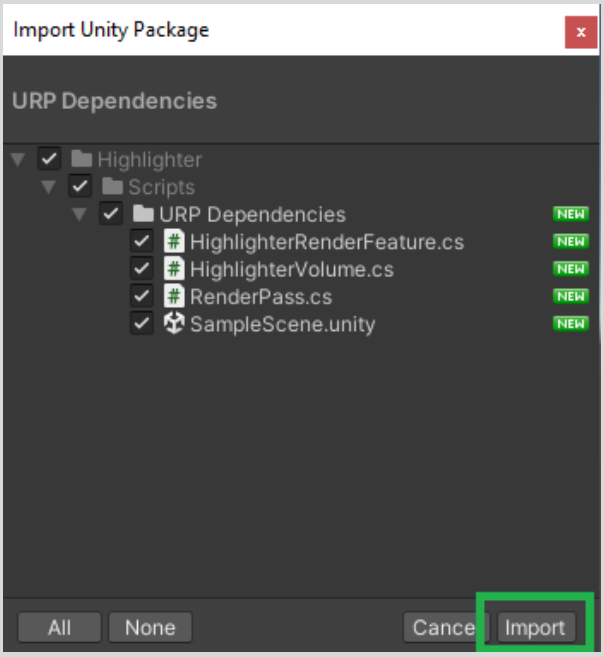
Step 1.2: Universal Render Pipeline Setup

If you are using Universal Render Pipeline Please make sure the Unity **Universal Render Pipeline** package is already imported in your project.

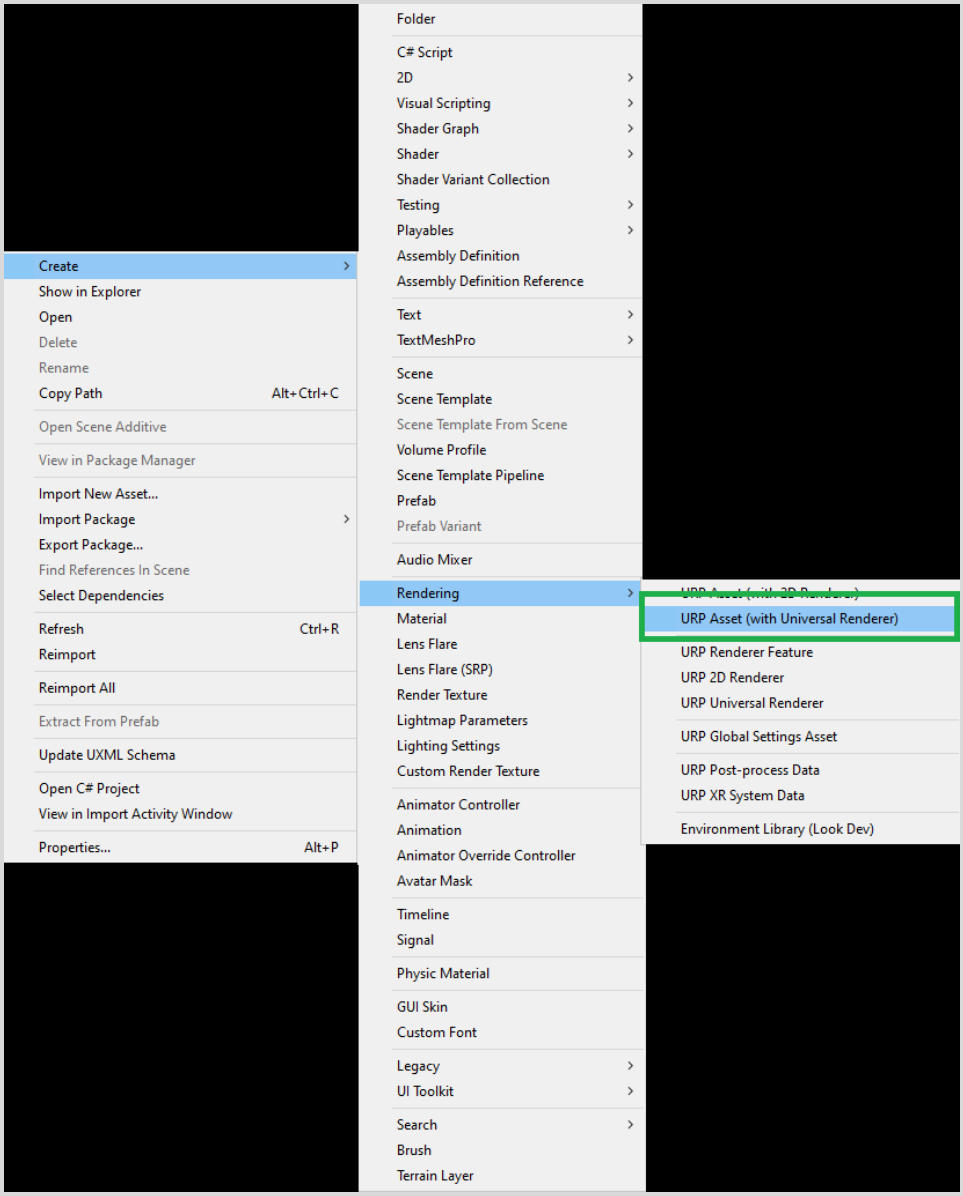
Import the "**URP Dependencies.unitypackage**" file from the following directory.
Highlighter>Scripts>URP Dependencies.unitypackage



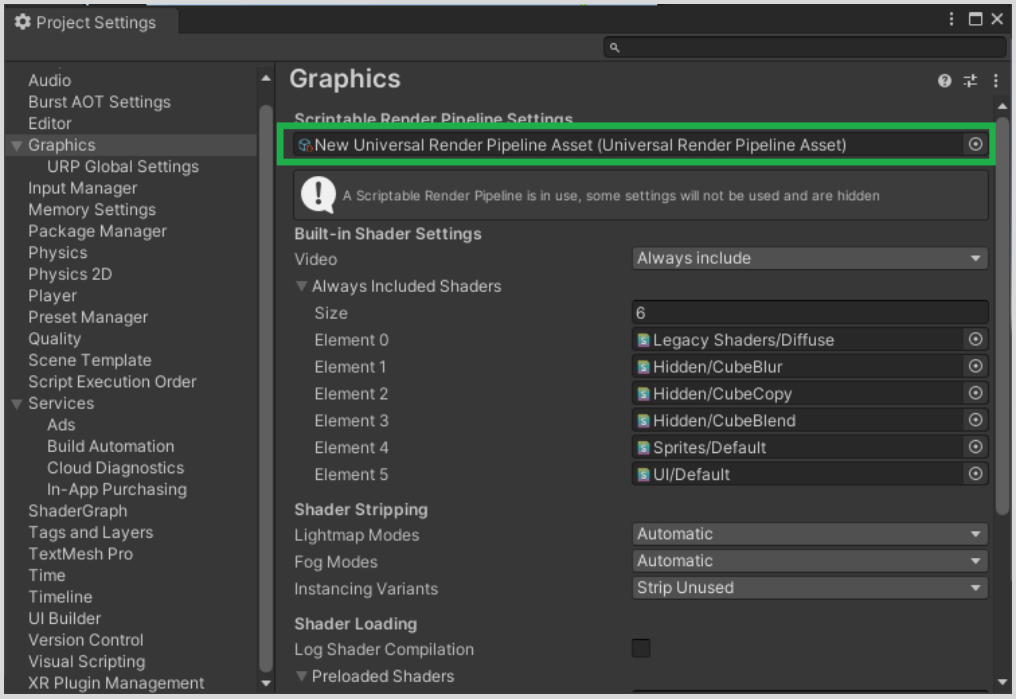
Double Click on "**URP Dependencies.unitypackage**" then click on the "**Import**" button.
*Note: Make sure to check all of the given files to import into your project also you should make sure Unity **Universal Render Pipeline** Package exists in your project otherwise you will get some errors.*



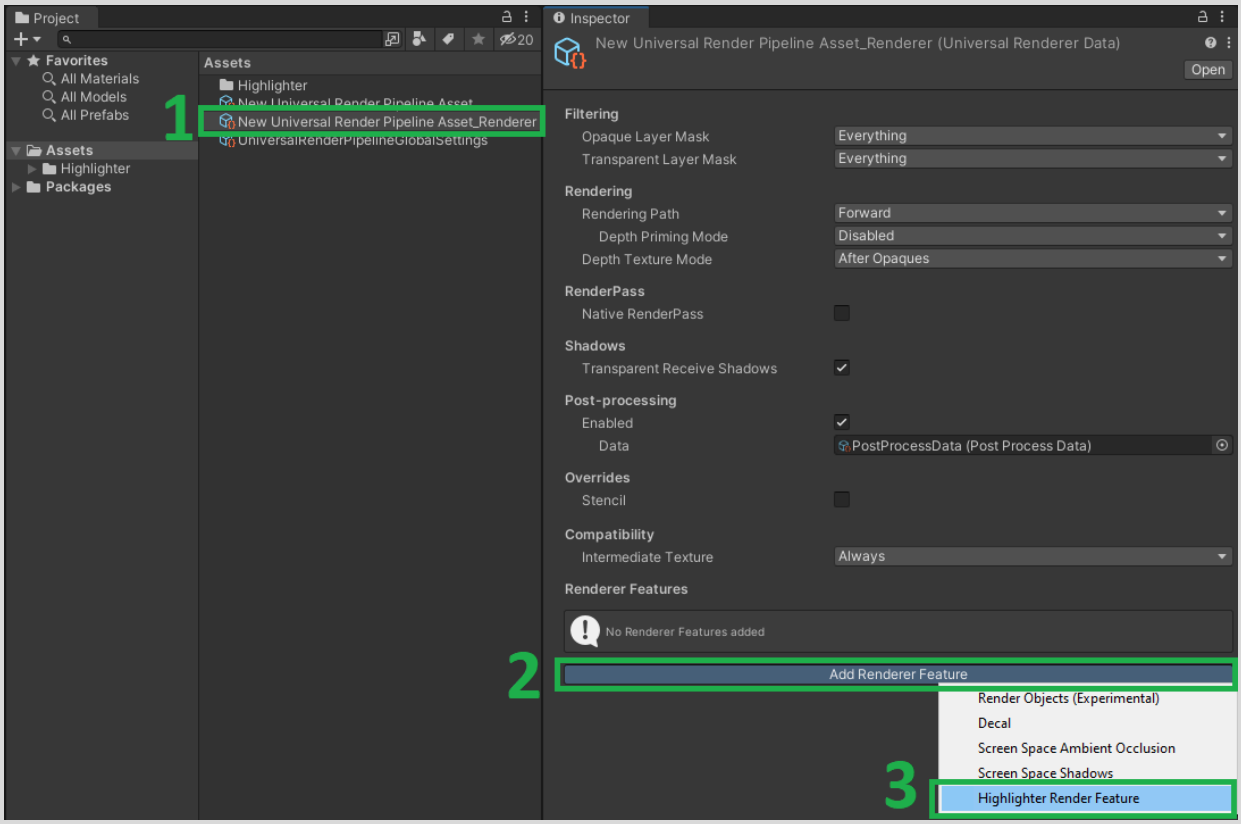
If you have not create any URP Assets, Please right click on empty place of “**Assets**” folder and navigate to **Create>Rendering>URP Asset (with Universal Rendering)** to create new URP assets



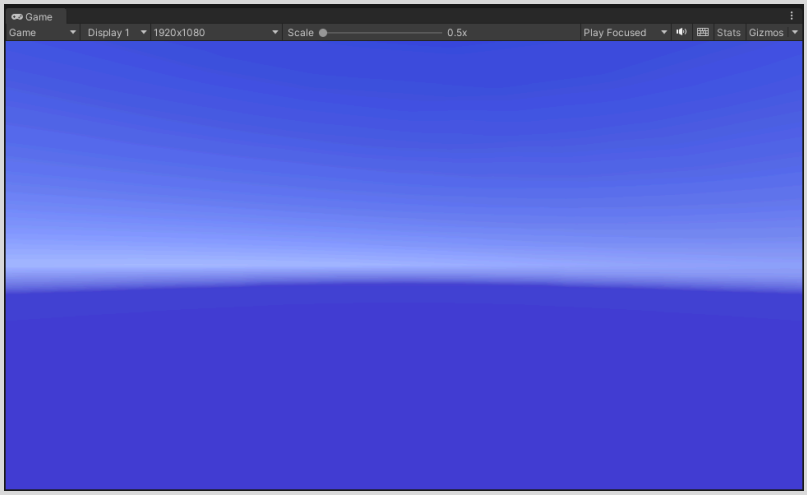
Open **Project Settings** from **Edit>Project Settings** then select “**Graphics**” and assign **Render Pipeline Asset** into the “**Scriptable Render Pipeline Settings**” field.



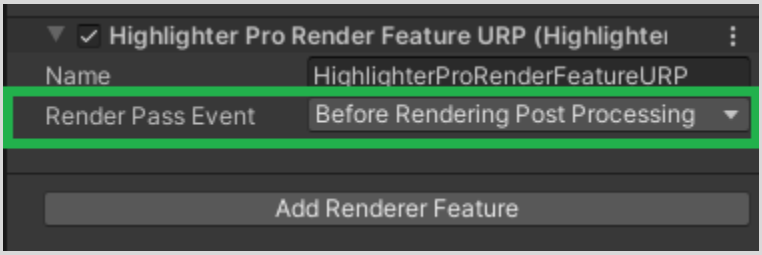
Please select the **Universal Render Pipeline Asset_Renderer** scriptable object from the asset folder then click on the "**Add Render Feature**" button from the inspector and choose "**Highlighter Render Feature**".



If you completed successfully the above steps you will be able to see some background color in your game view.

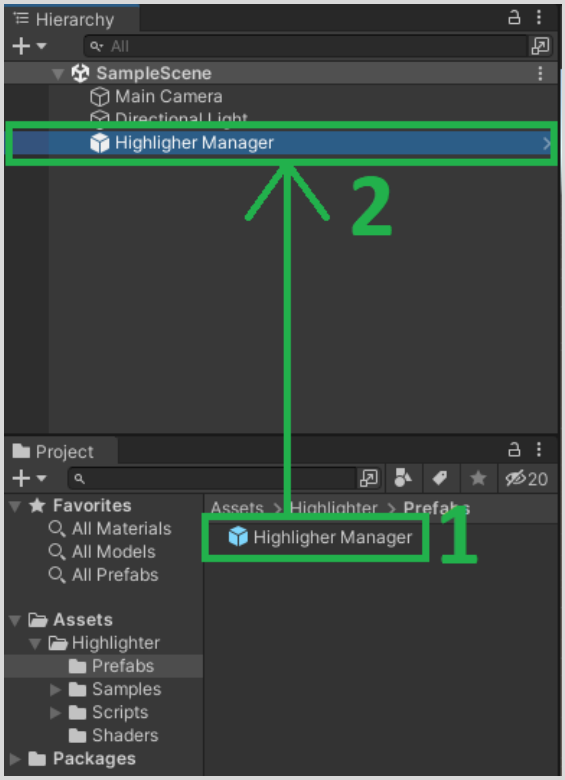


After adding the **Highlighter** render feature, you will be able to change "**Render Pass Event**" from the inspector.



Step 2: Add Highlighter Manager Prefab in the Scene

Navigate the "**Highlighter Manager"** prefab and drag into the scene from following directory **Highlighter>Prefabs>Highlighter Manager.prefab.** Now you can change the background color with the Highlighter Manager component from the inspector.



Step 3: Add Highlight effect Component on gameobject

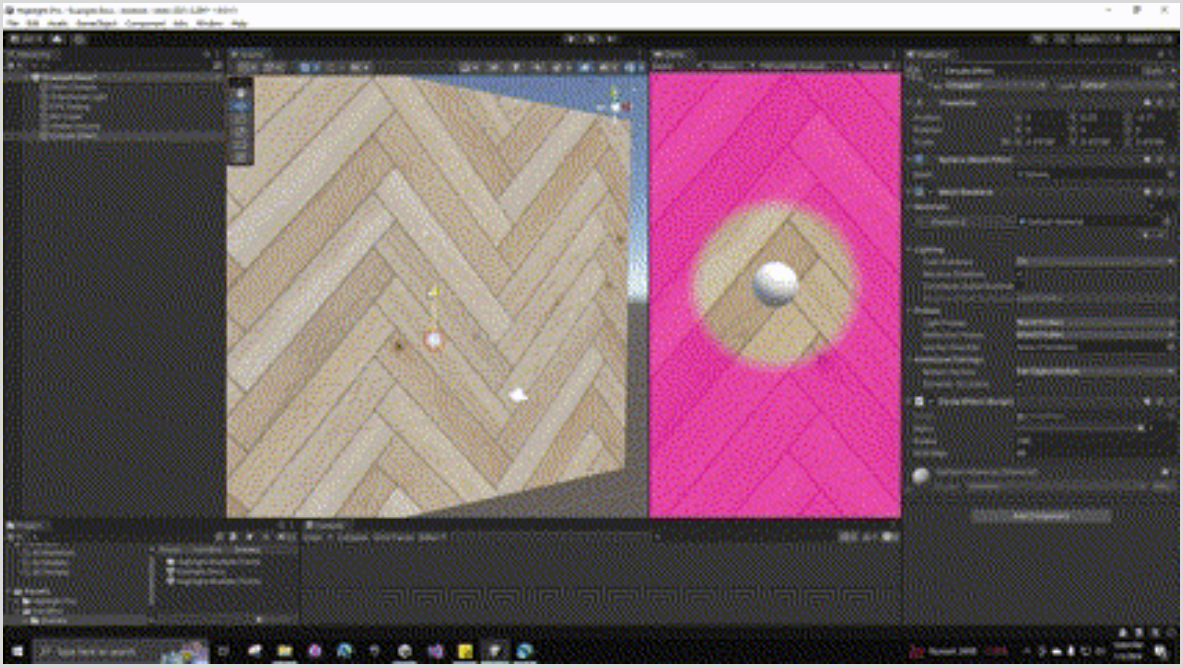
You have to add at least one component of highlight effects on a Game Object which one you want to highlight. You can also add multiple effects on a single Game Object.

Select a game object and add any components listed below which highlight effects you want to apply on the selected gameobject.
*Note: The **highlight effect** will be only visible when the **gameobject** is placed in front of the **Camera**. If the **gameobject** placed behind the **Camera** you will be unable to see the effect.*

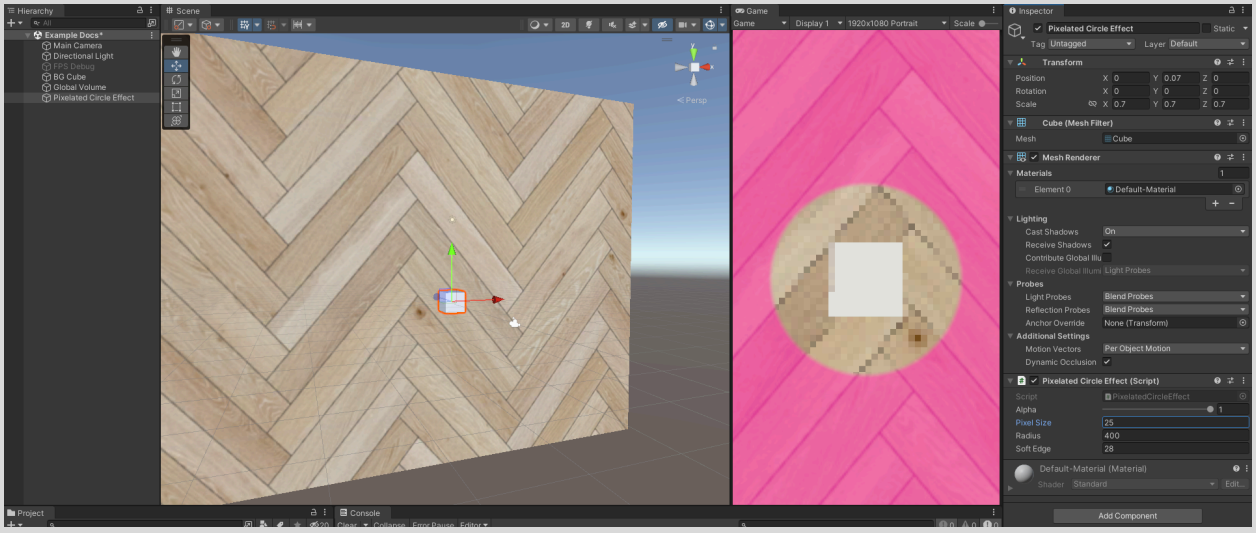
Highlighter allows you to apply highlight effects listed below.

- **Circle Effect**
- **Pixelated Circle Effect**
- **Rectangle Effect**
- **Pixelated Rectangle Effect**
- **Radial Wave Effect**

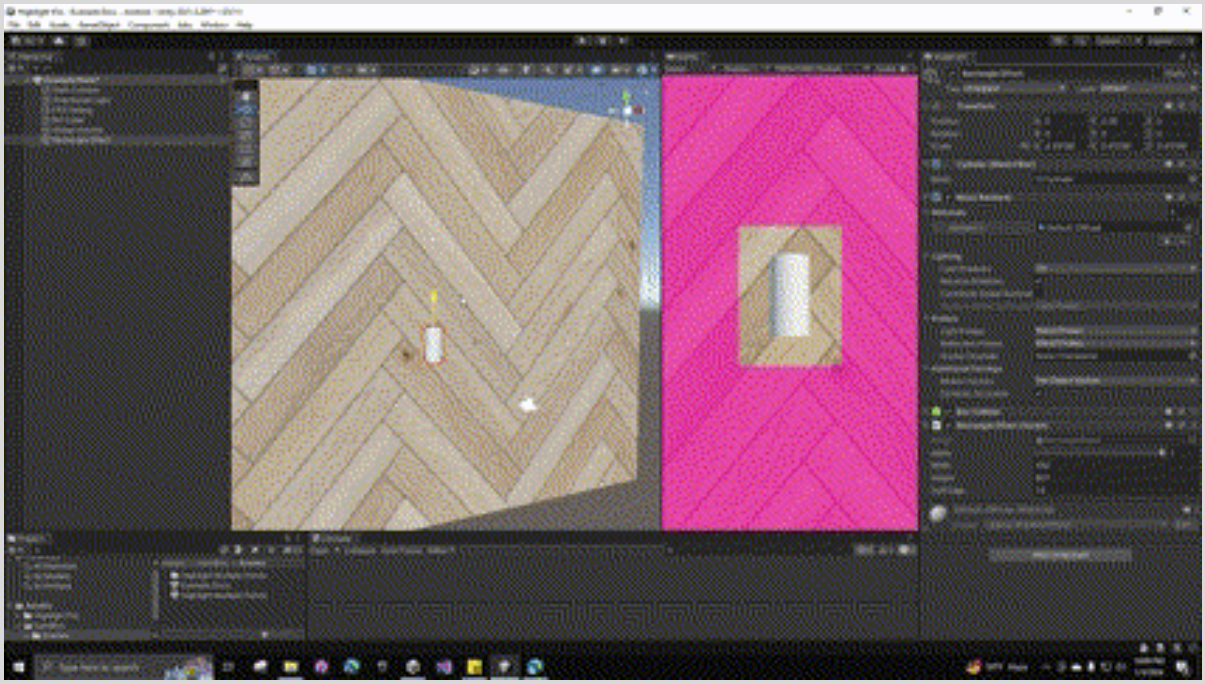
Circle Effect: It is a component that allows you to highlight a gameobject that contains this Component with circular shape.



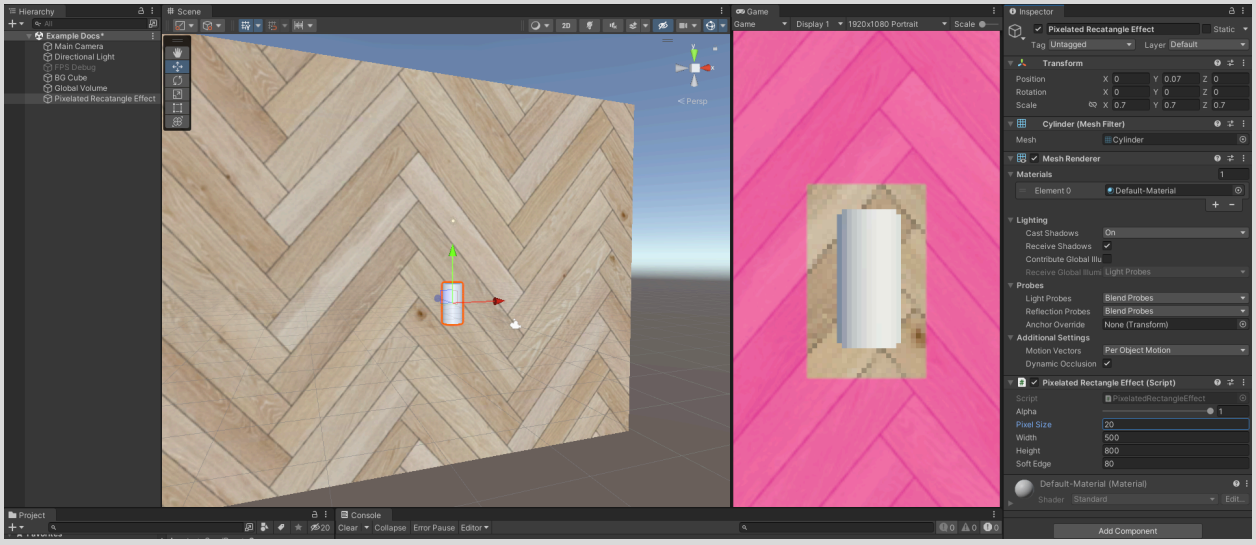
Pixelated Circle Effect: It is a component that allows you to highlight a gameobject that contains this Component with pixelated circular shape. The highlighted area will render with pixelated effect.



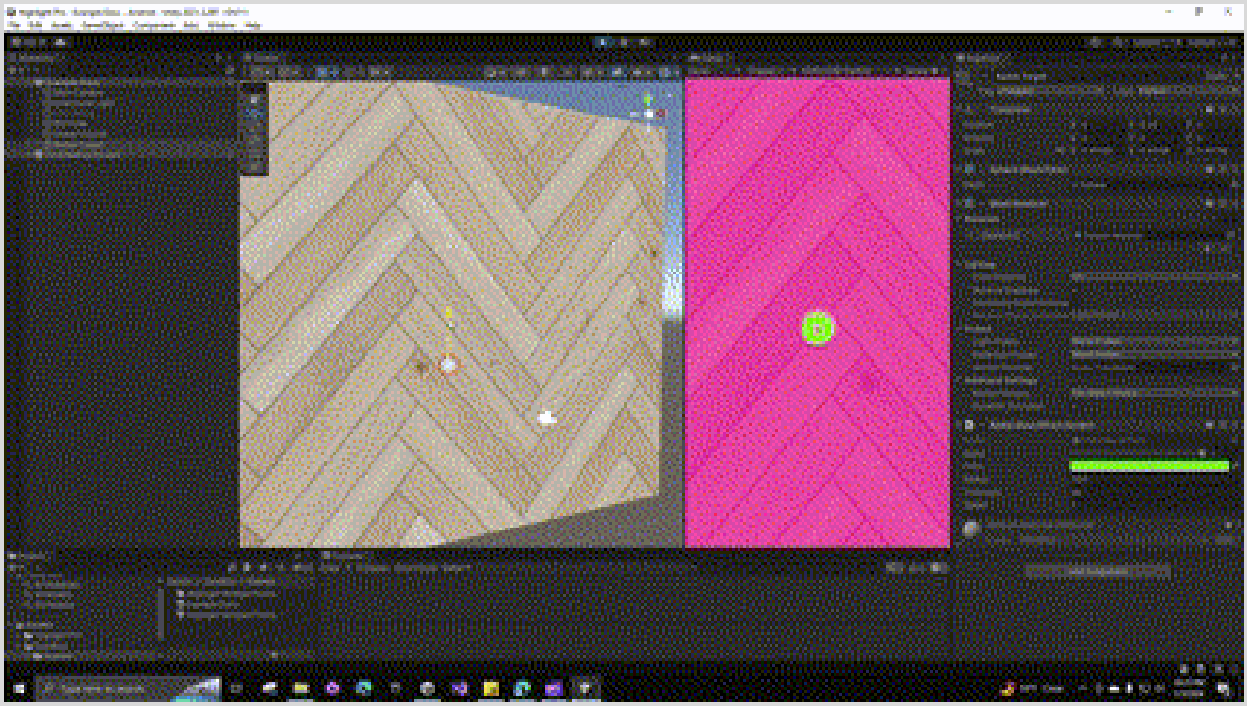
Rectangle Effect: It is a component that allows you to highlight a gameobject that contains a this Component with rectangle shape.



Pixelated Rectangle Effect: It is a component that allows you to highlight a gameobject that contains a this Component with pixelated rectangle shape. The highlighted area will render with pixelated effect.

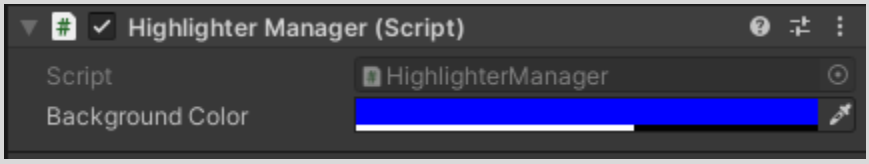


Radial Wave Effect: It is a component that allows you to highlight a gameobject that contains a this Component with radial wave effect.



API Documentation

HighlighterManager

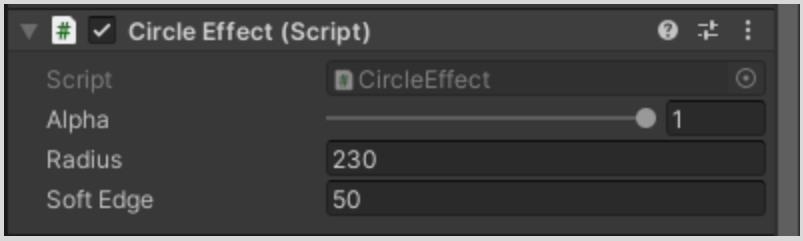


Public Property:

static Color BackgroundColor {get; set;}

Change the background fill color of the game view

Circle Effect



Public Property

uint radius :

Resize the radius of a highlighted circle.

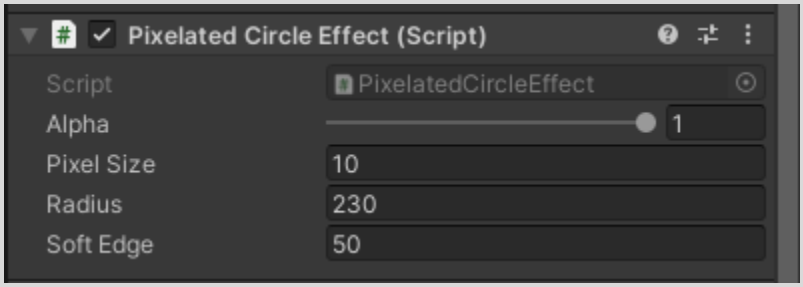
uint softEdge :

Edge softness of the highlighted circle. You can change the circle edge from soft to hard by changing this value.

float alpha :

It's an inherited property, The alpha value shows how transparent this effect is to render on the screen.

Pixelated Circle Effect:



Public Property

uint pixelSize :

Pixel size of highlighted area. You can change the pixel size of the highlighted area.

uint radius :

Resize the radius of a highlightable circle.

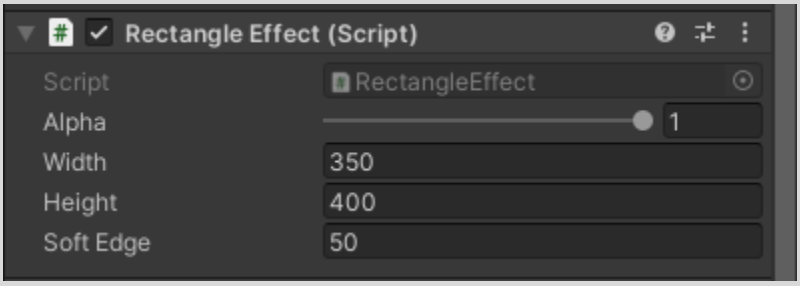
uint softEdge :

Edge softness of the highlighted circle. You can change the circle edge from soft to hard by changing this value.

float alpha :

It's an inherited property, The alpha value shows how transparent this effect is to render on the screen.

Rectangle Effect



Public Property

uint width :

Width of a highlighted rectangle. You can resize the width of the "Rectangle Effect" component by changing this value.

uint height :

Height of a highlighted rectangle. You can resize the height of the "Rectangle Effect" component by changing this value.

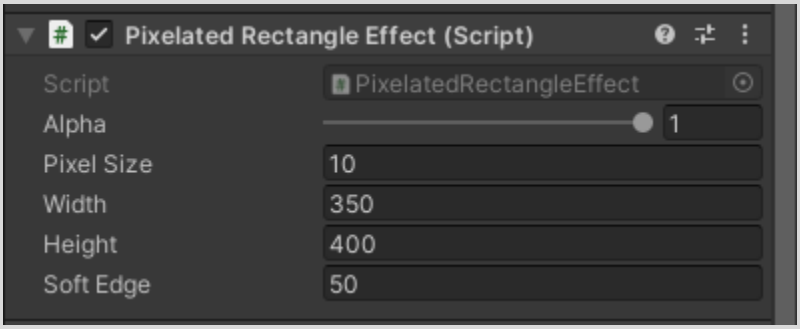
uint softEdge :

Edge softness of the highlighted rectangle.

float alpha :

It's an inherited property, The alpha value shows how transparent this effect is to render on the screen.

Pixelated Rectangle Effect



Public Property

uint pixelSize :

Pixel size of highlighted area. You can change the pixel size of the highlighted area.

uint width :

Width of a highlighted rectangle. You can resize the width of the "Pixelated Rectangle Effect" component by changing this value.

uint height :

Height of a highlighted rectangle. You can resize the height of the "Pixelated Rectangle Effect" component by changing this value.

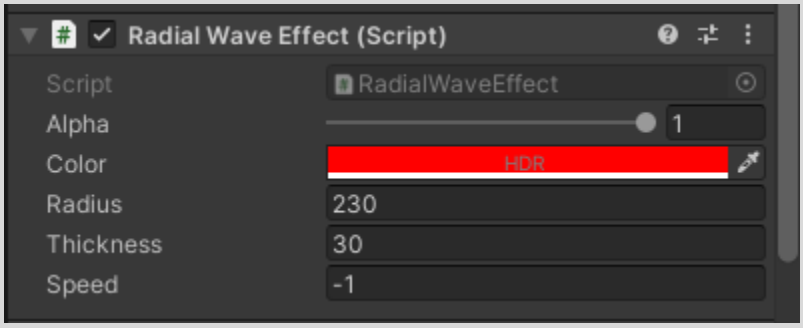
uint softEdge :

Edge softness of the highlighted rectangle.

float alpha :

It's an inherited property, The alpha value shows how transparent this effect is to render on the screen.

Radial Wave Effect



Public Property

Color color :

Color of the effect.

uint radius :

Radius of the radial wave, How much radius covers this "RadialWaveEffect".

uint thickness :

The thickness of the wave, how thick it will fill the wave edge to render.

float speed :

Speed of the wave, how fast the wave travels. You can change the wave direction by changing the sign of the speed value from negative to positive.

float alpha :

It's an inherited property, The alpha value shows how transparent this effect is to render on the screen.

.

Thanks For Reading