2024

MyInteractionKit Documentation

myintera
Syazmedia PLT
9/5/2024

Welcome to the MyInteractionKit PDF Manual.

Thank you for purchasing MyInteractionKit. We hope that it accomplishes what you are trying to build.

For up-to-date tutorials, visit MyInteractionKit website at https://myinteractionkit.com/index.php/tutorials/.

To view upcoming and latest videos as they are developed, please check out https://myinteractionkit.com/index.php/videos/.

For detailed, script-by-script documentation, go to https://myinteractionkit.com/index.php/docs/.

If you need any support, feel free to reach out at syazmediaplt@gmail.com.

Contents

MyInteractionKit Documentation

Introduction

MyInteractionKit is a comprehensive Unity asset designed for simple yet flexible interactions without needing to write code. Each interaction is driven by prefabs, making it easy for developers to integrate advanced mechanics into their projects quickly.

Features

Basic Trigger Interactions

Easily detect when objects enter or exit trigger areas and perform actions such as highlighting or showing on-screen prompts. You can define how each object behaves when interacted with using the provided prefab system.

Emissive Object Highlighting

Automatically highlight objects when a player is nearby, using customizable emissive materials. This is useful for visual feedback, helping players identify which objects are interactive.

Object Interactions

Trigger various UnityEvents, like playing animations, opening doors, or activating UI elements when players press the correct key. The key used for interaction is customizable for each object.

Customizable Key Bindings

Each object interaction can be configured to respond to different key presses, allowing for flexibility in control schemes. This is especially useful for developers who want to offer customizable controls to their players.

Tweening System

SimpleTween allows smooth object animations between two points with optional reversibility. This feature is great for creating smooth object transitions, such as doors opening or platforms moving.

Rotator Control

Easily start and stop object rotation with simple controls. This feature can be useful for rotating objects such as collectible items, animated environments, or mechanisms.

Screenshot Tool

Capture screenshots directly within the Unity Editor. The screenshot tool offers two options:

1. **Take Screenshot**: Press Ctrl+T to take an immediate screenshot.
2. **Take Delayed Screenshot**: Press Ctrl+Shift+T to take a delayed screenshot with a default 5-second delay.
The tool also offers preset resolution sizes for various platforms, including iOS App Store and Unity Asset Store formats.

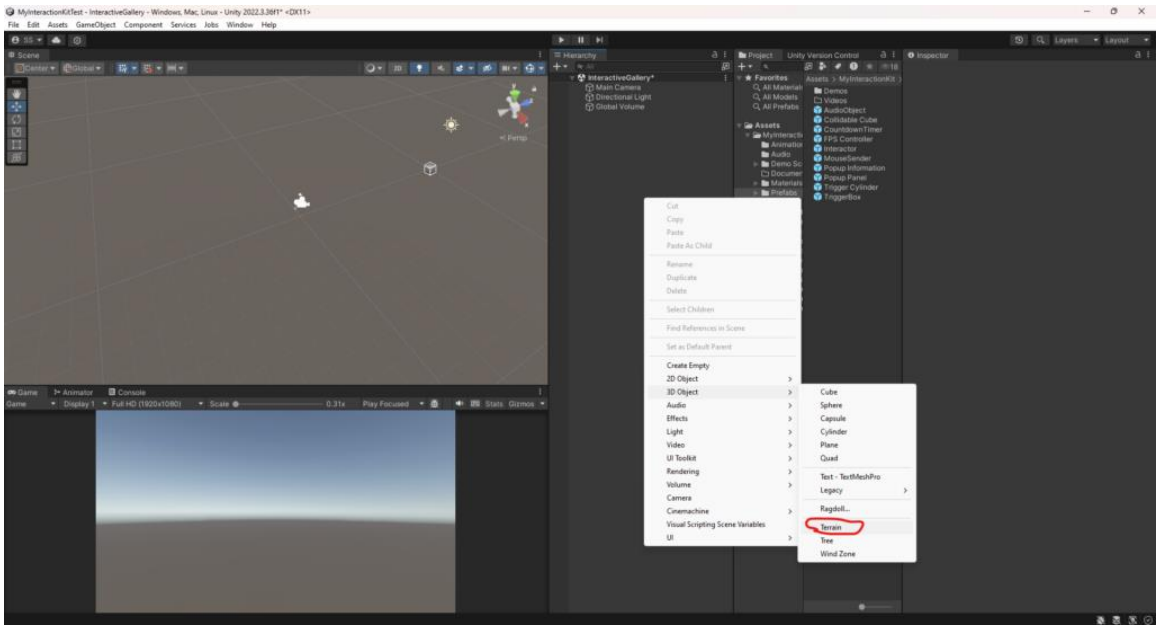
Conclusion

MyInteractionKit provides a no-code solution for adding interactivity to your Unity projects. With customizable key bindings, emissive highlights, and smooth object animations, it's perfect for rapid prototyping and non-programmers.

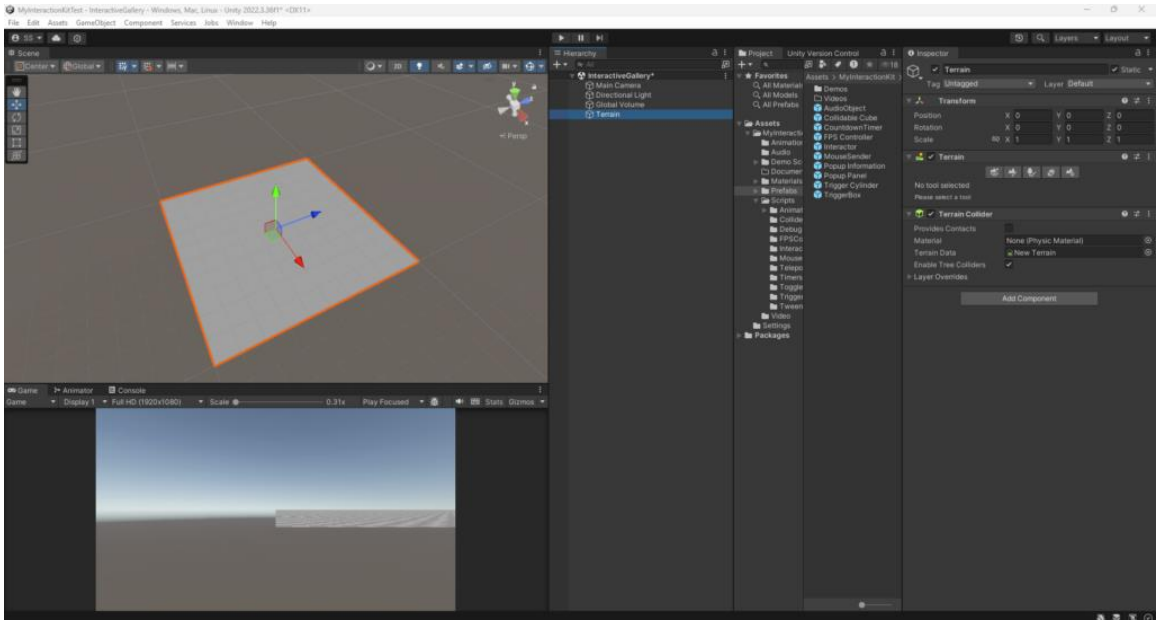
Setting Up The First Person Controller

In this tutorial, we will setup MyInteractionKit and build an Interactive Galllery. Let's start.

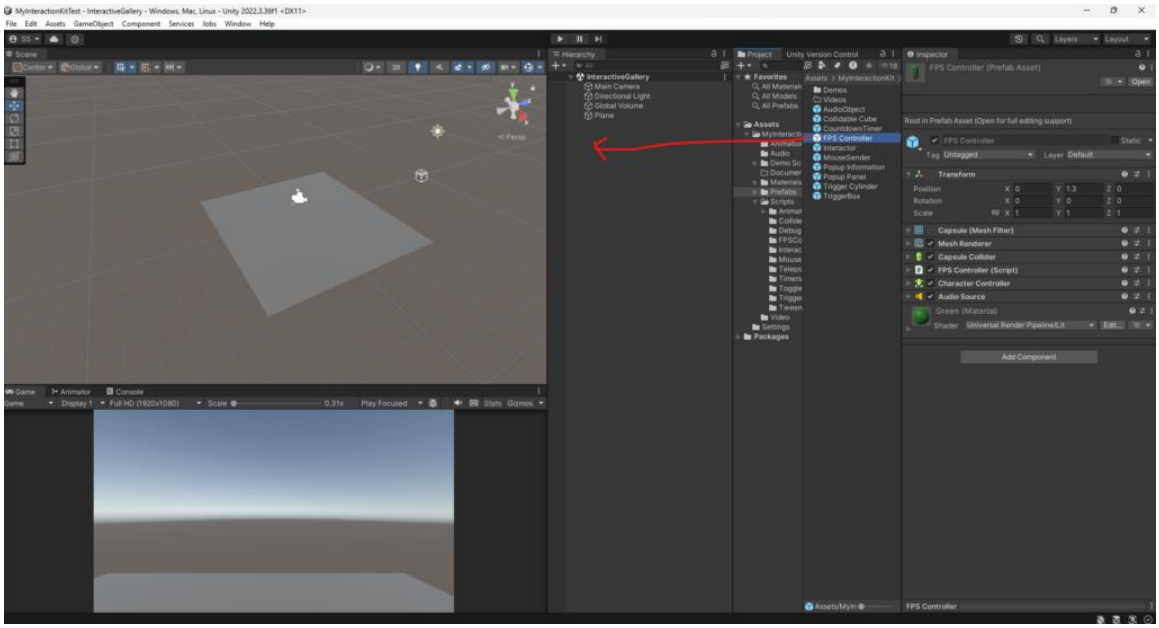
Firstly, we will create a Terrain so that we can have a large surface to walk on. In the **Hierarchy** window, **Right-click > 3D Object > Terrain**



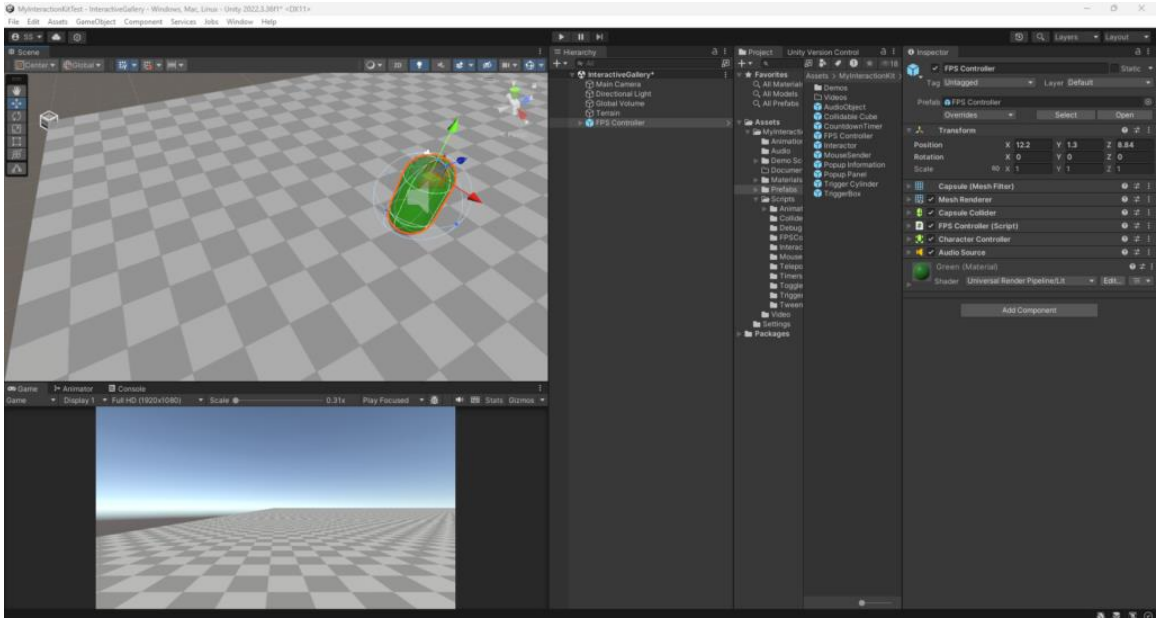
Select the Terrain, and press F on the keyboard. The terrain will center on the Scene window



Inside the prefabs folder of MyInteractionKit, drag and drop a **First Person Controller**.



Next, select the FPS Controller, press **F** on the keyboard to focus on the FPS Controller. Lets move and position the FPS controller above the terrain so that we can walk on top of it. Something like below



Press Play, you should now be able to walk on top of our Terrain now. This is the end of our first tutorial.

Note: If you fall down endlessly, it means that your FPS controller is not above the terrain.

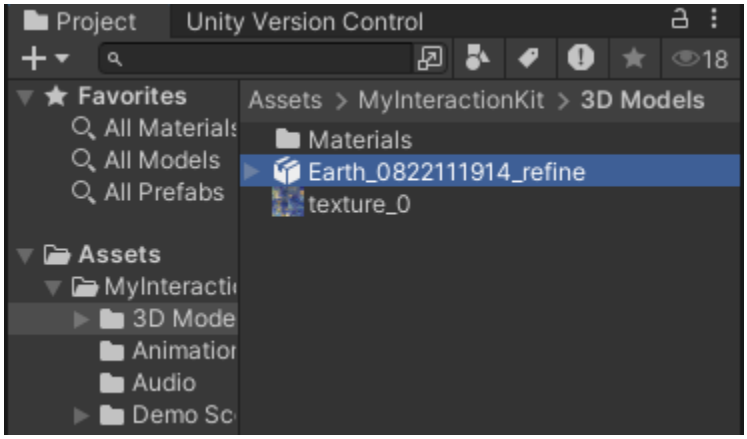
Let's Start Building Interactions

Now that we have an FPS Controller, the next thing we want to do is to put in our first artwork for this gallery. It is an interactive Earth 3D model.

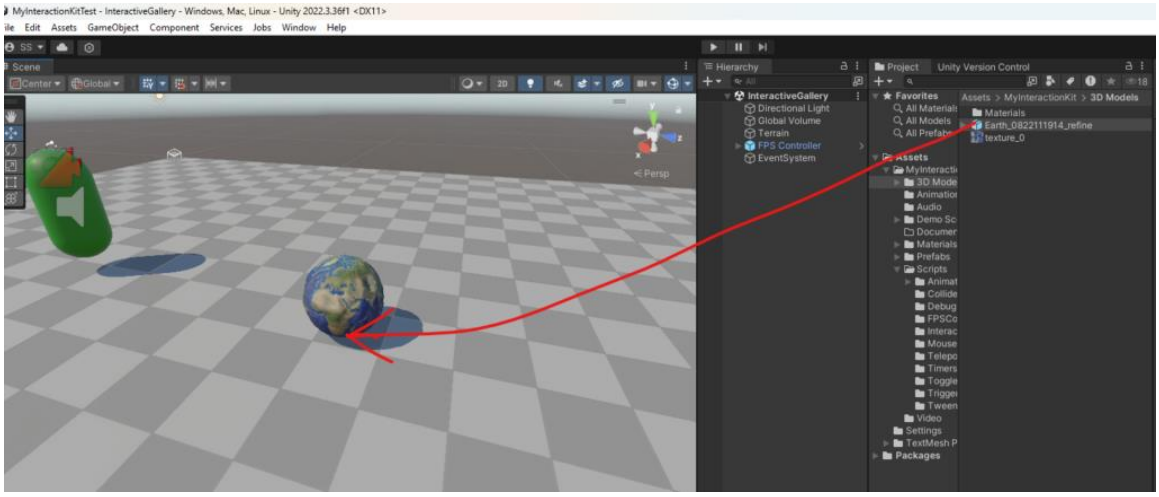
At the end of this tutorial, it will look like this.

When you go near it, an interaction option will appear, and if you press F, the Earth rotates, music plays, lights come on and particles starts to appear. Let's start.

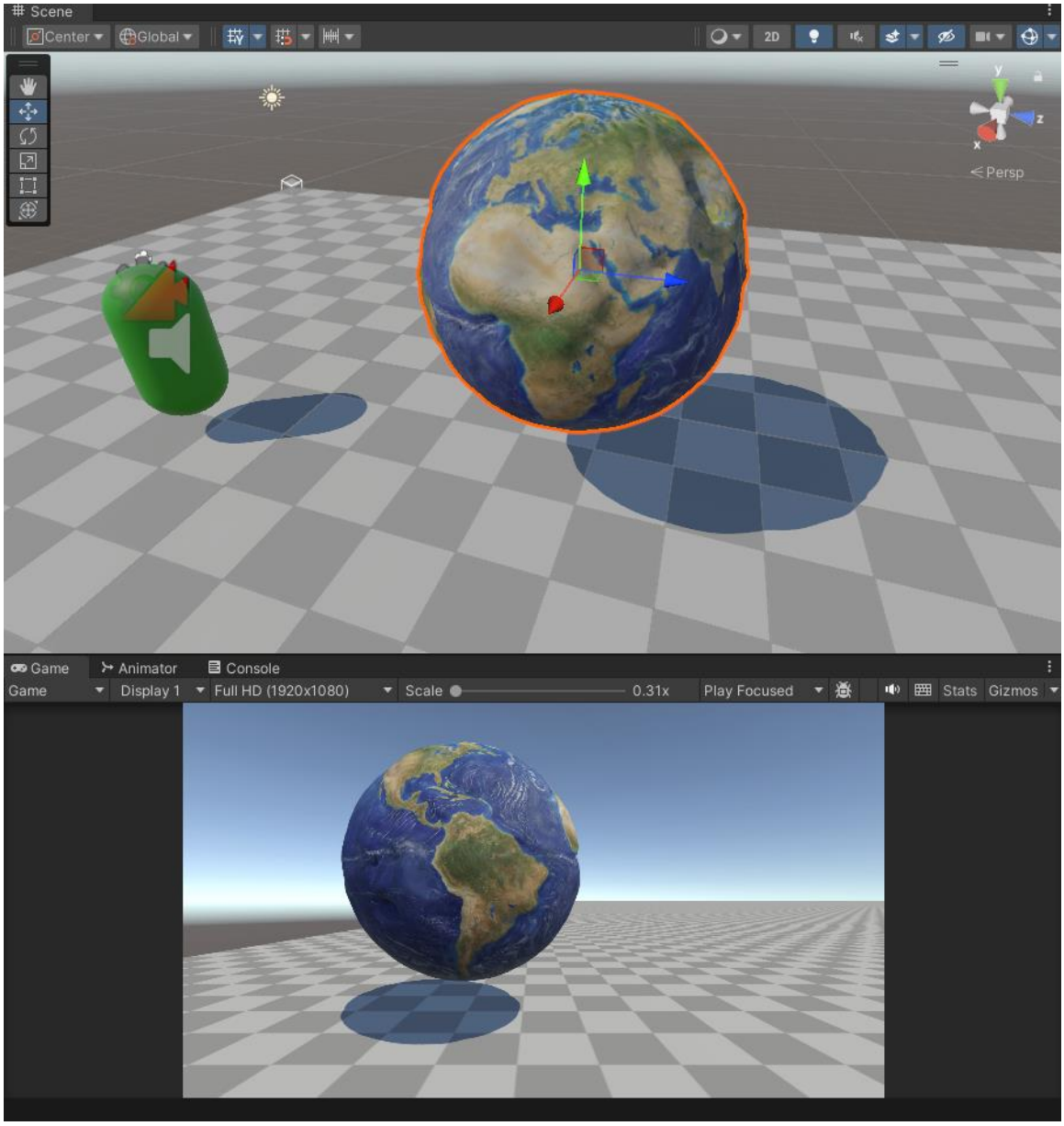
Firstly, lets drag and drop the Earth 3D model. You can find it in the **MyInteractionKit > 3D Models** folder.



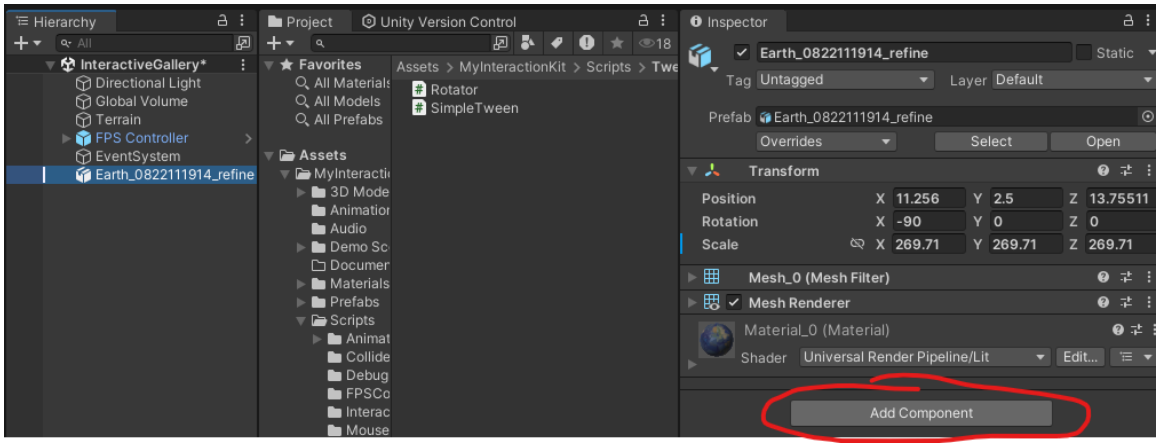
Drag and Drop the Earth into our Scene like this.



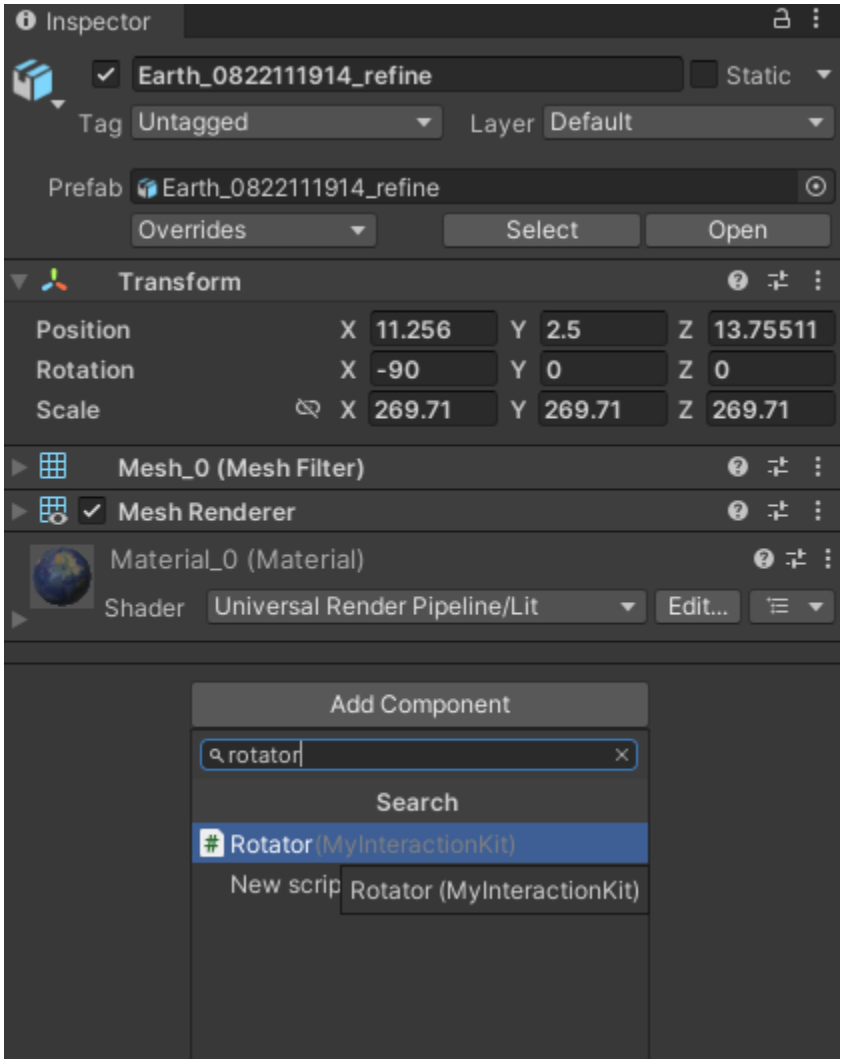
To make an interaction impactful, people need to see it. So let's make the Earth bigger and lift it above the floor so that it looks like its floating. Adjust as you see fit.


Scene
Center
Global
2D
Persp
Game
Animator
Console
Game
Display 1
Full HD (1920x1080)
Scale
0.31x
Play Focused
Stats
Gizmos

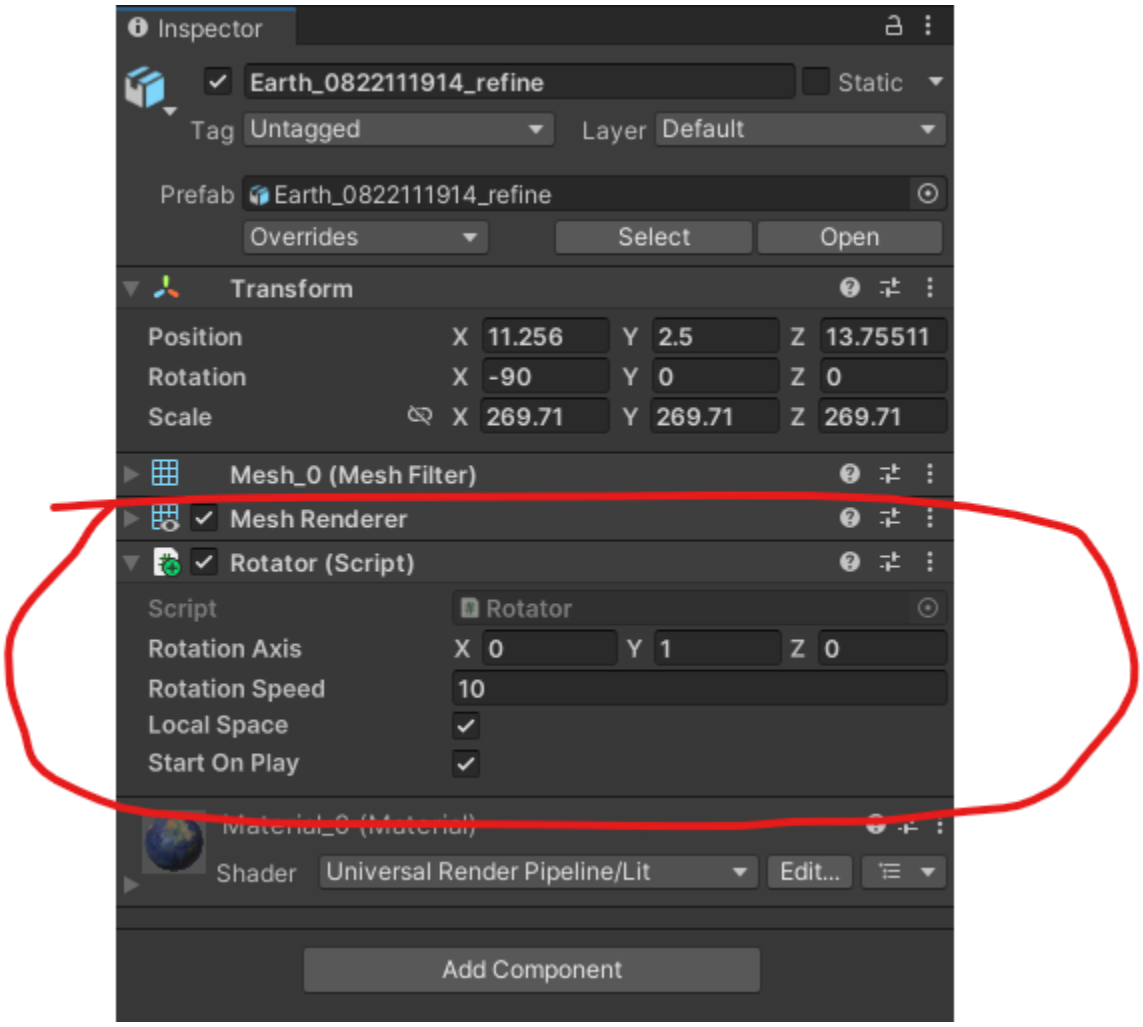
Now that we have the Earth, let's make it Rotate. Select the Earth, and press **Add Component** on the **Inspector window**



A search window thingy will appear, search for **Rotator** and select the **Rotator** script.

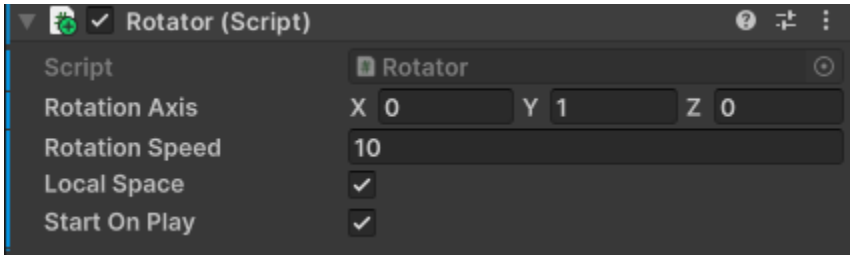


The Rotator component will appear in the Inspector



Press **Play** in **Unity**, you will see that the Earth is now rotating.

The Rotator Component



Lets get to know the Rotator settings

Rotation Axis – Changing these values impact the direction and magnitude of the rotation.

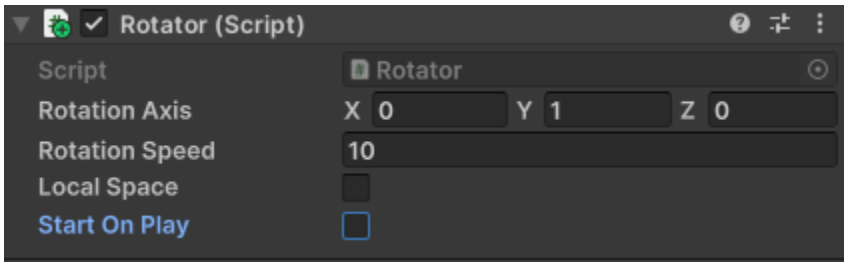
Rotation Speed – The Speed of the rotation.

Local Space – If this is checked, the object will rotate on *Local Space* instead of *World Space*. Here is an explainer post.

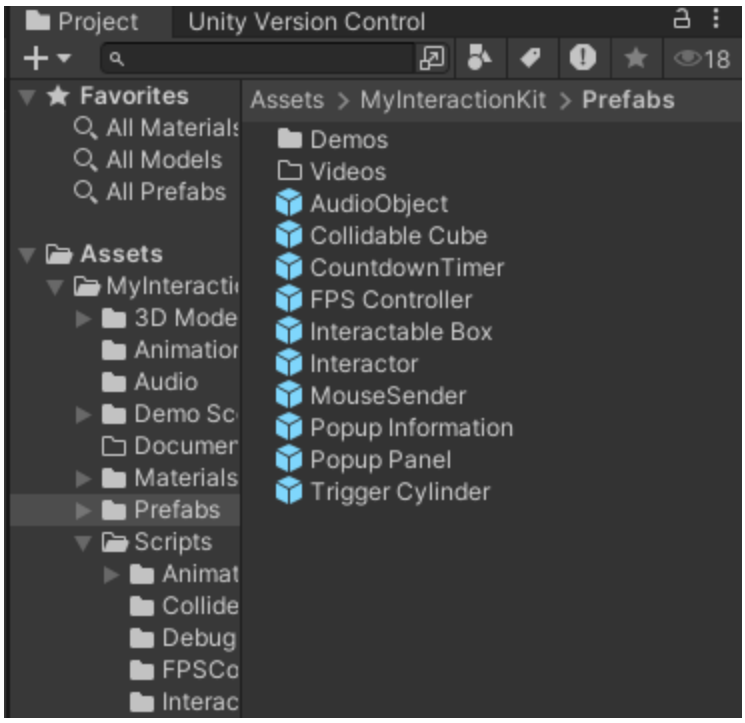
Start On Play – If this is checked, the rotation will automatically start every time the game runs

Making Earth rotate on command

Currently, the Earth is rotating immediately. That's not so Interactive Artwork-ish isn't it? Let's make it rotate when user interacts on it. **Uncheck** the **Start On Play** checkbox. This will make it stop playing automatically.



Next, let's put in an **Interactable Box**. You can find it in **MyInteractionKit > Prefab** folder.

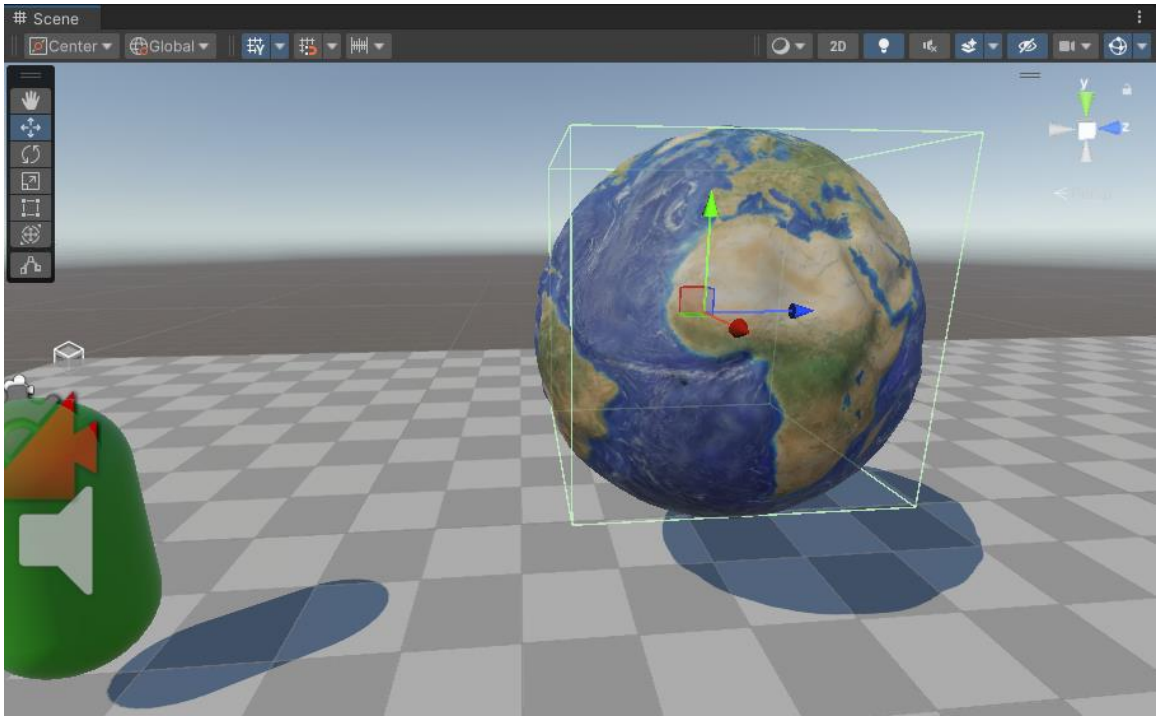


Drag and Drop Interactable Box into the Scene, just like you dragged the Earth. You will see a **Green Transparent Box**. Players can't see this, only you can.

That is the **Interactable Box**.

Setting up the Interactable Box

First, lets move the box so that it nicely covers the Earth, like this.



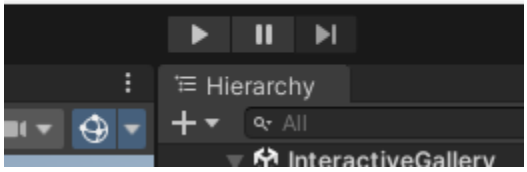
Press **Play**, and move closer to the **Earth**. A popup appears like below.

When the FPS Controller gets near an Interactable, it will cause this popup to appear. Its designed that way.

Note:

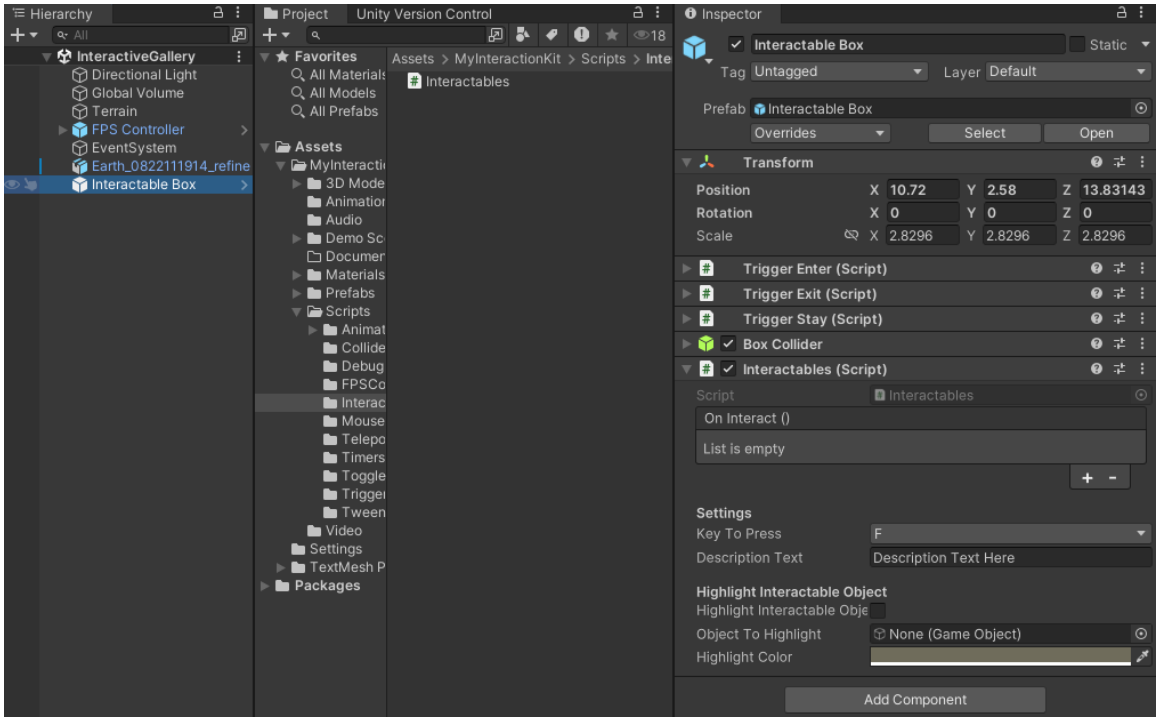
This popup is common in games. You see them almost everywhere. But just in case you never saw it before, the letter highlighted in red is the key you are supposed to press. The description, well.. explains what will happen.

Press **Play** again to stop playing. Make sure the Play button is not highlighted in blue.



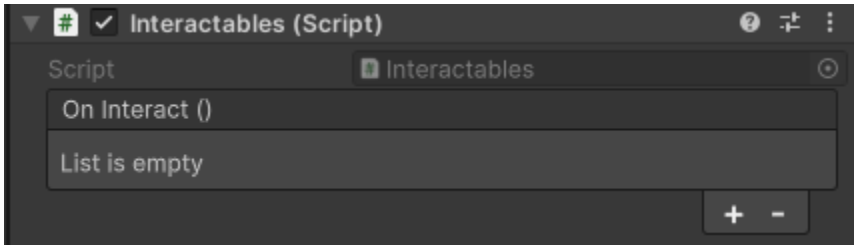
Why? if you make changes during play, when you stop playing the changes disappear.

Next, select our **Interactable Box**, and the Inspector Window will show the **Interactables** component. We will get to know this component in another post. Let's move on and you will see how this works.

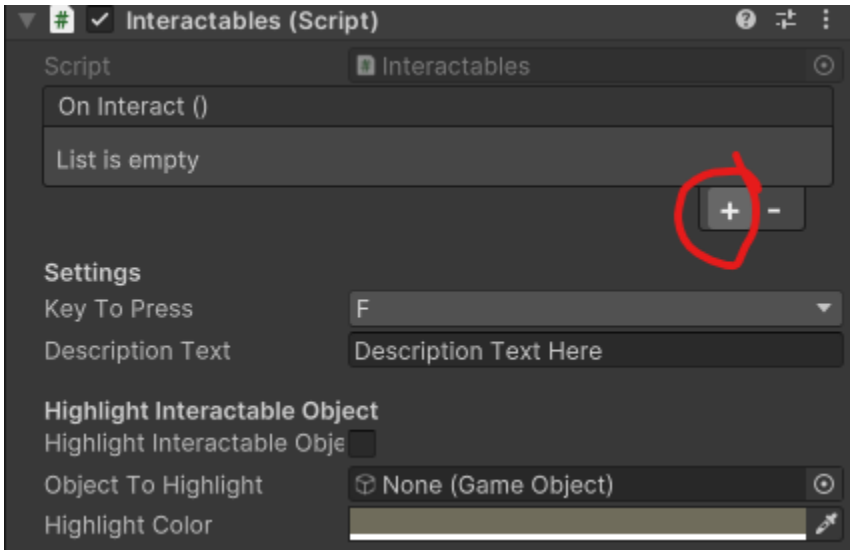


Telling the Earth to Rotate

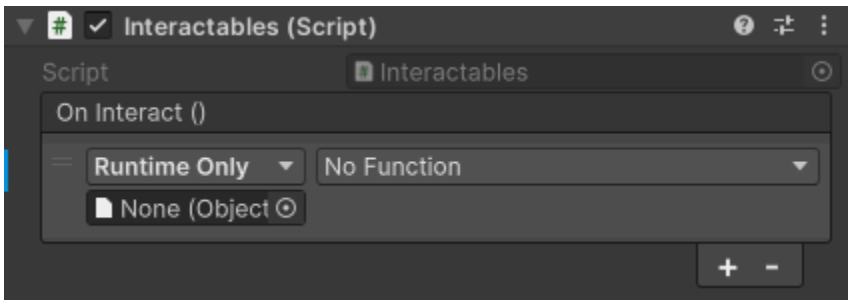
On the Interactables component, find the On Interact() list.



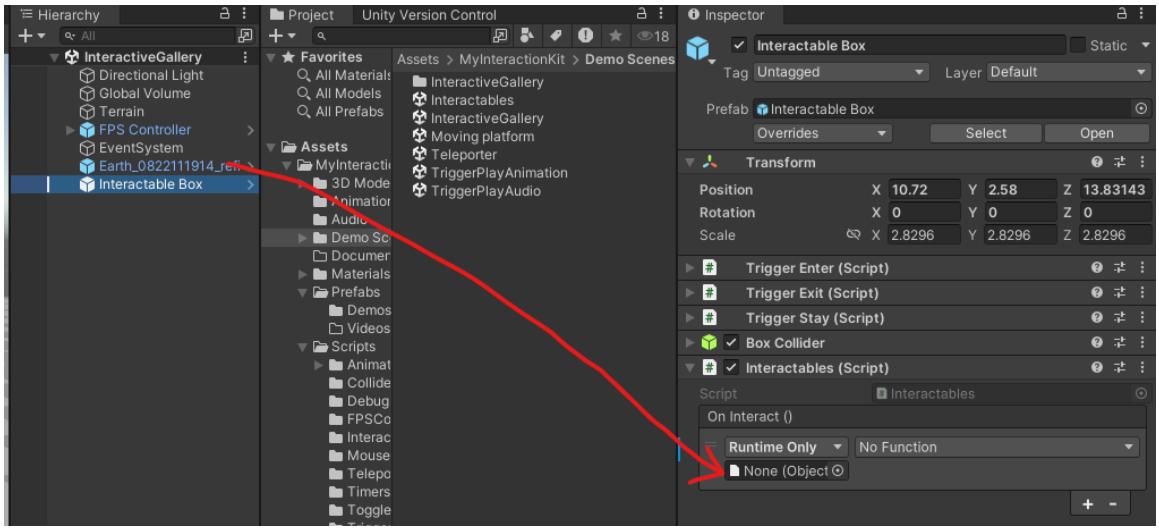
In the **OnInteract**, press the +;



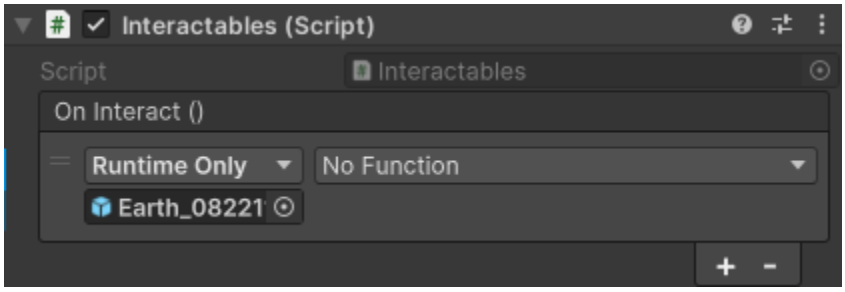
A new item appears on the OnInteract() event list, like below.



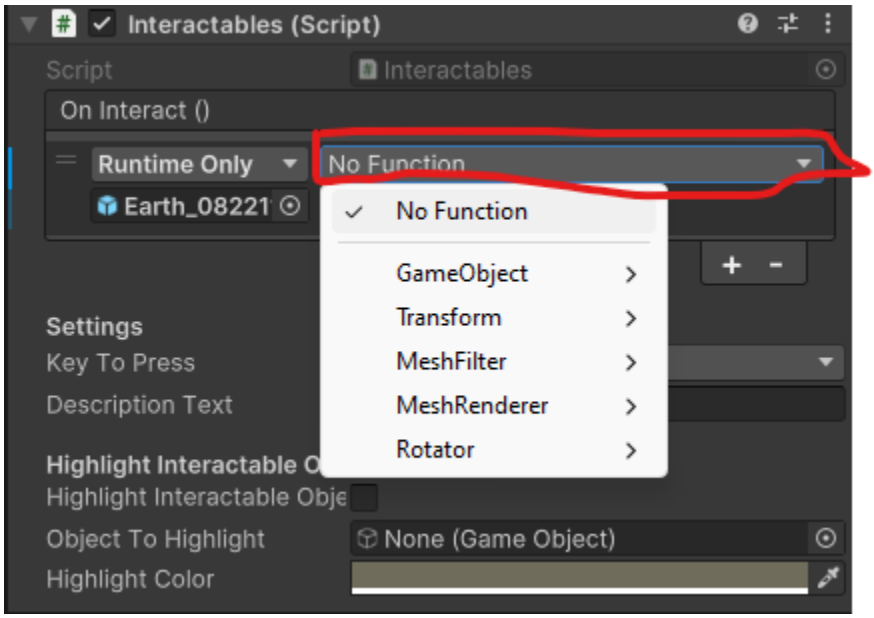
Drag and Drop Earth to the Object area.



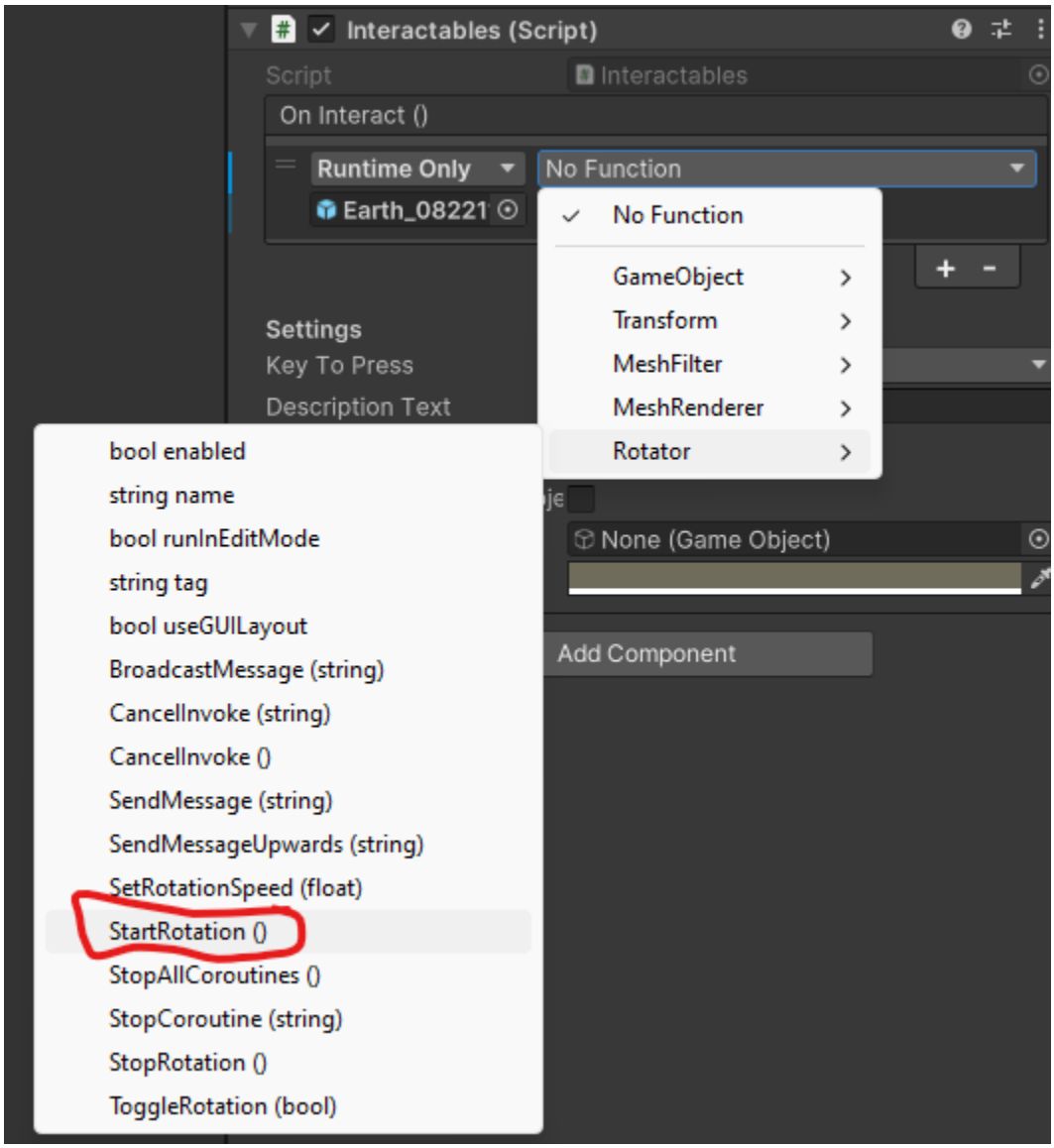
It will look like this now.



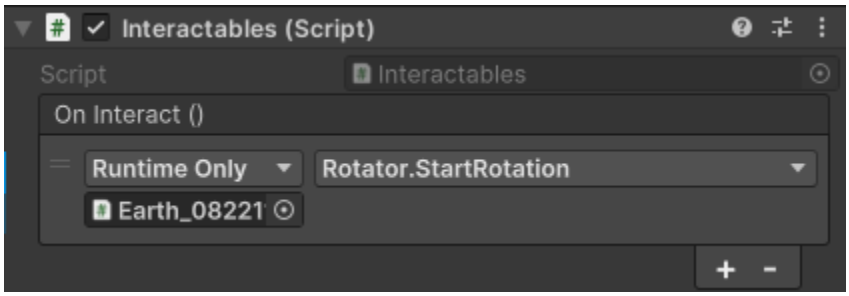
Press the **Function** dropdown. A list will appear. This are the list of all commands available for all Components. Upon interaction, we can call these commands so that it gets executed.



Select **Rotator > Start Rotation**



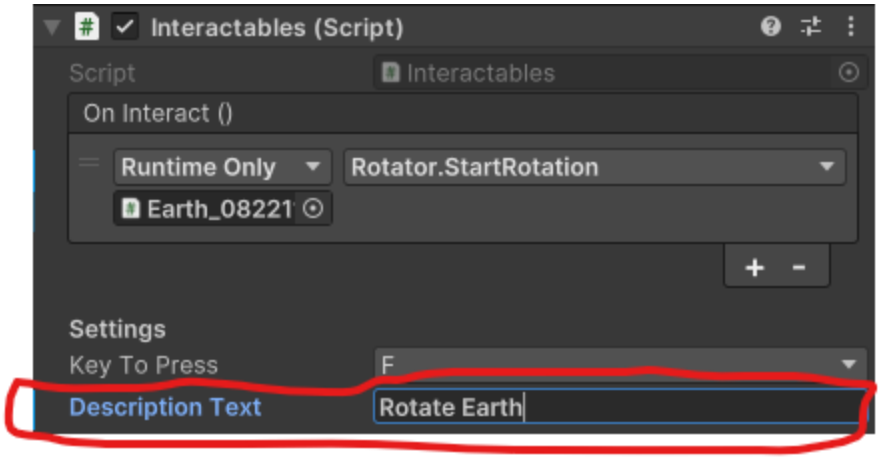
You will have something like this.



Press **Play** and test it out. When you go near Earth, the popup appears, when you press F key, the Earth rotates!

Customizing The Popup

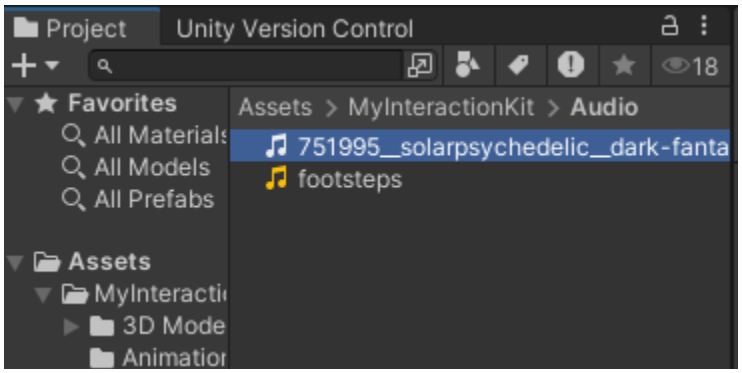
So now lets customize the popup. To do that, lets change the **description** to **"Rotate Earth"** on the **Interactables** component.



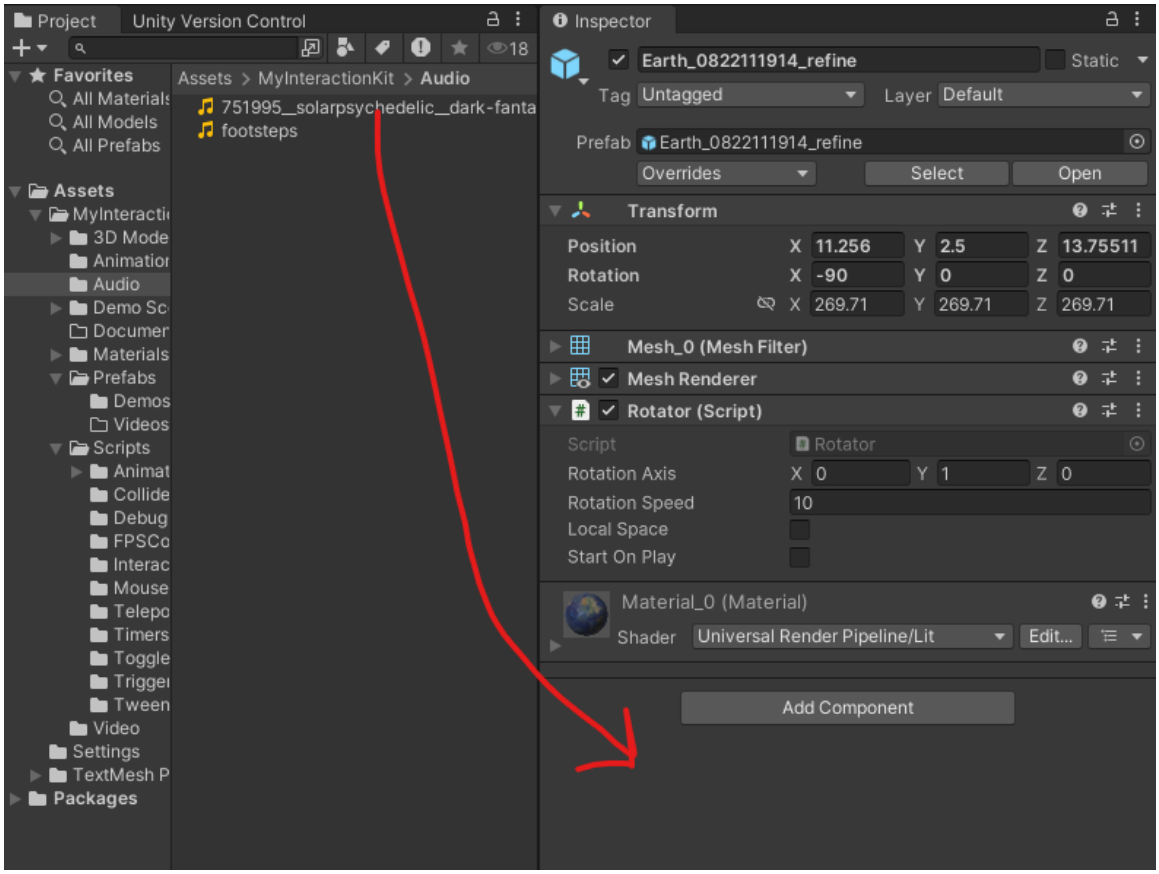
Press **Play**, and view your changes.

Adding Some Music

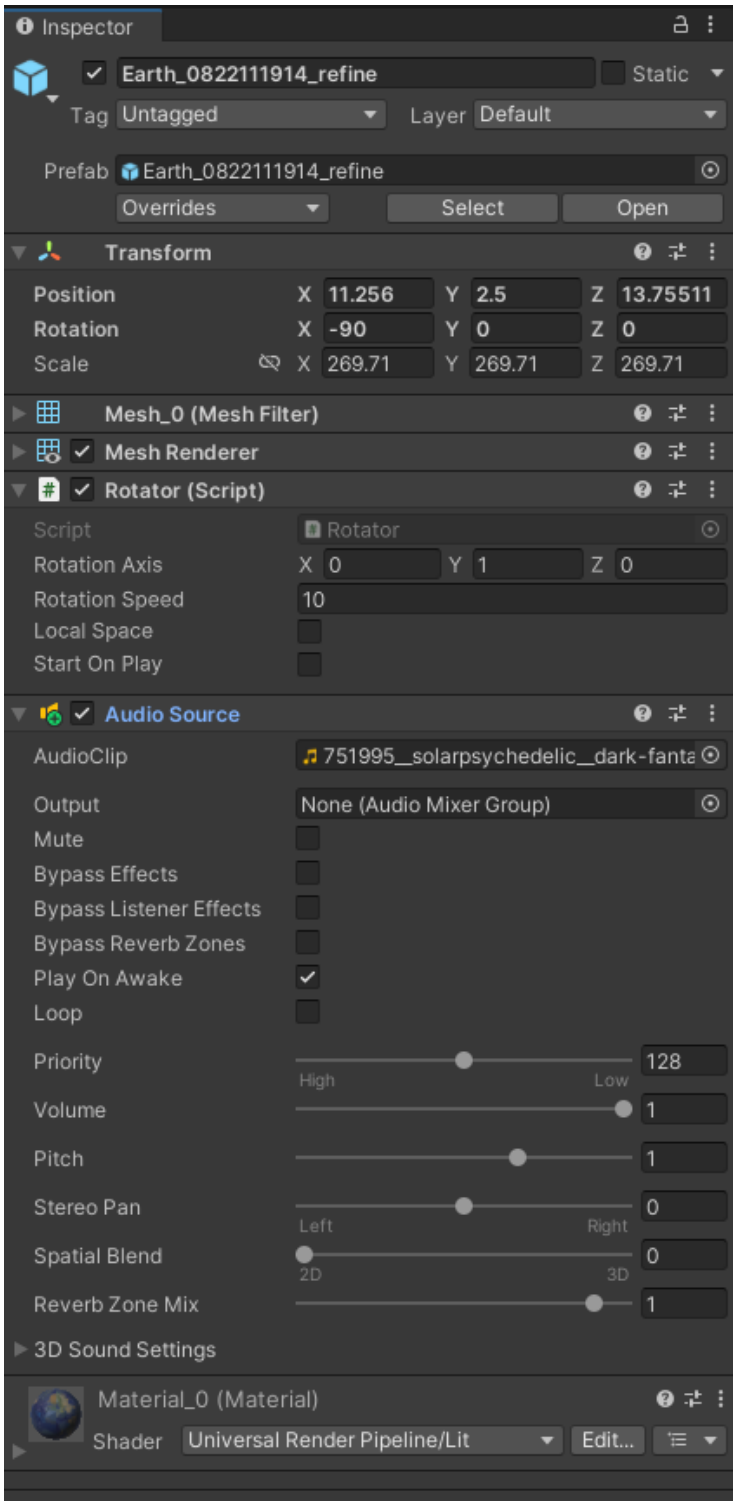
But this is hardly art right now, let's give it some more depth. We need music. In the **MyInteractionKit > Audio folder**, find the music named **751995__solarpsychedelic__dark-fantasy-theme.**



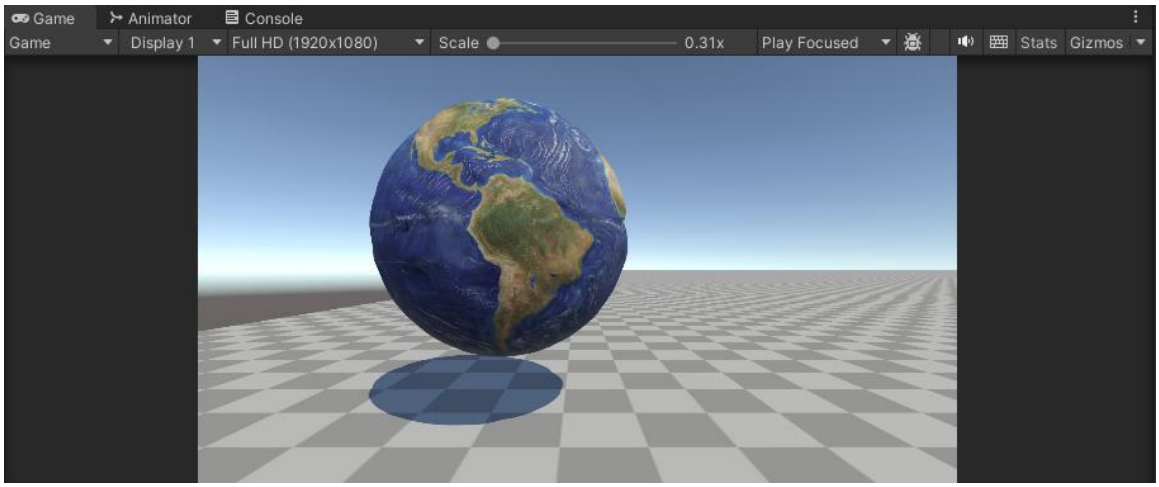
Select **Earth**. Drag and Drop the Music to the Inspector window.



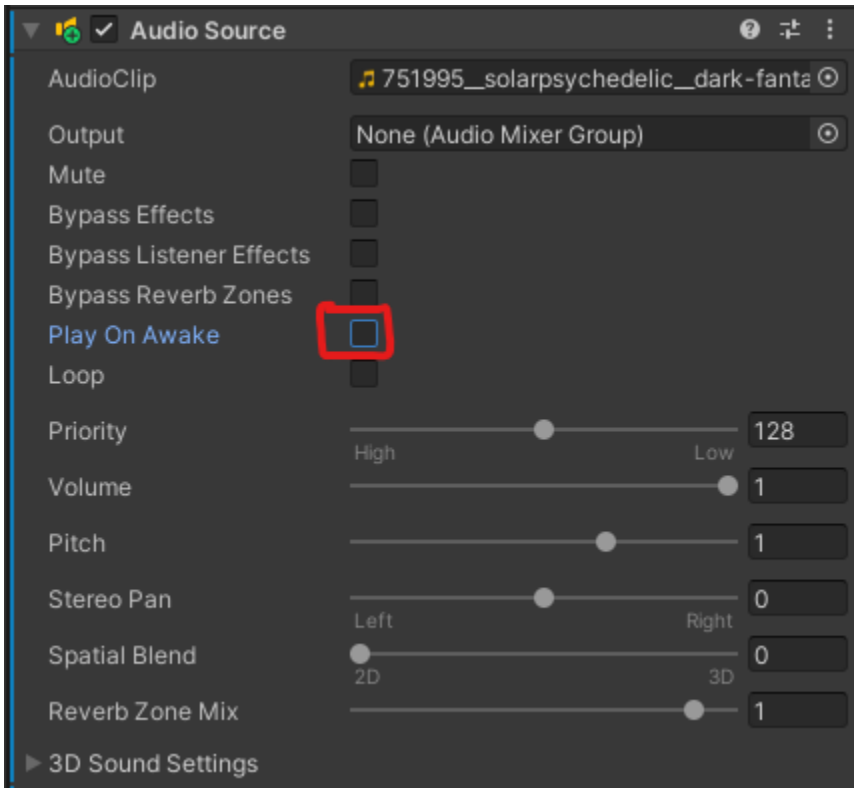
An Audio Source will appear in the Inspector. It will look like this.



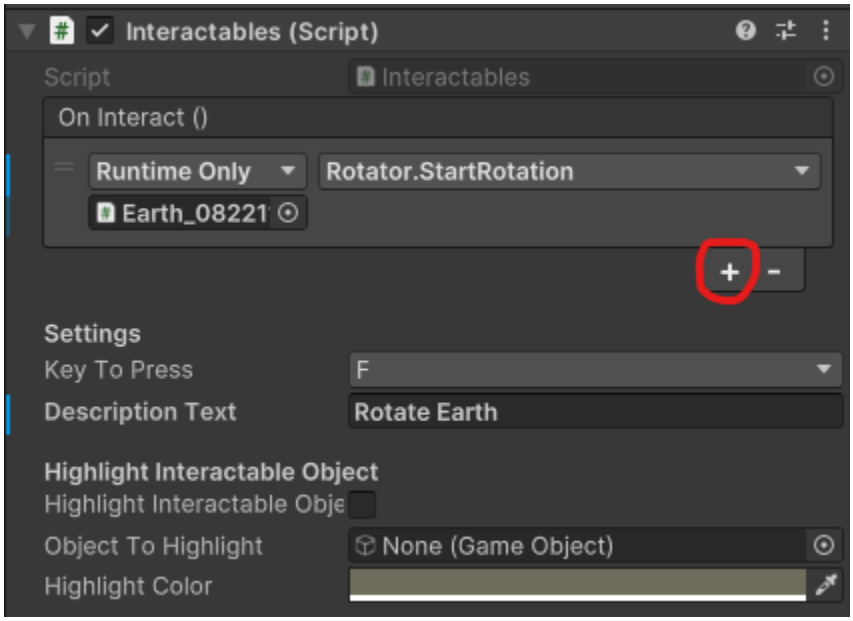
Press **Play**, you would hear music playing.



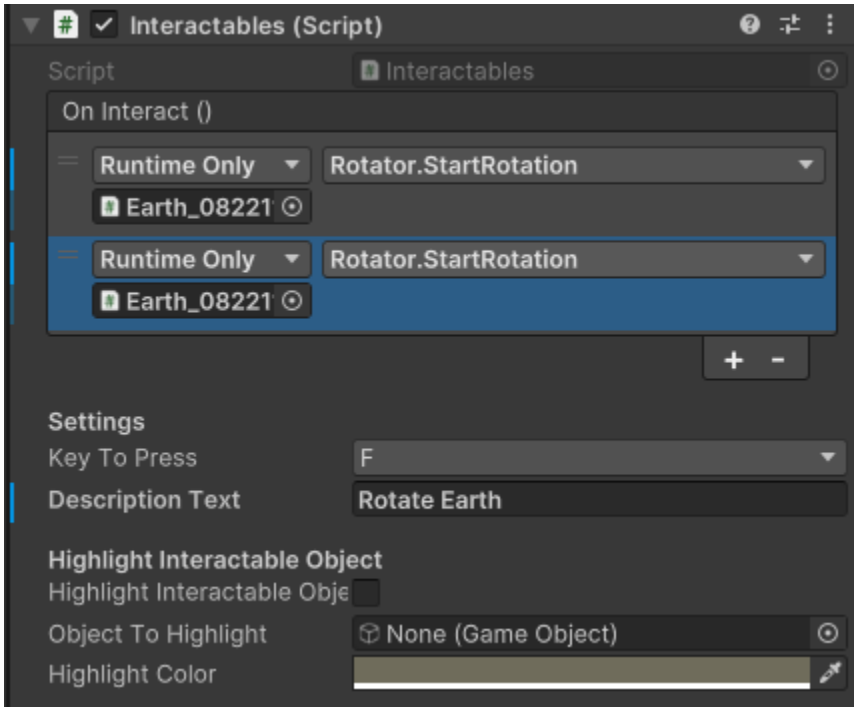
Okay, but we are not attempting to do background music. We want it to play together with the Earth spinning. **Uncheck** the **Play on Awake**.



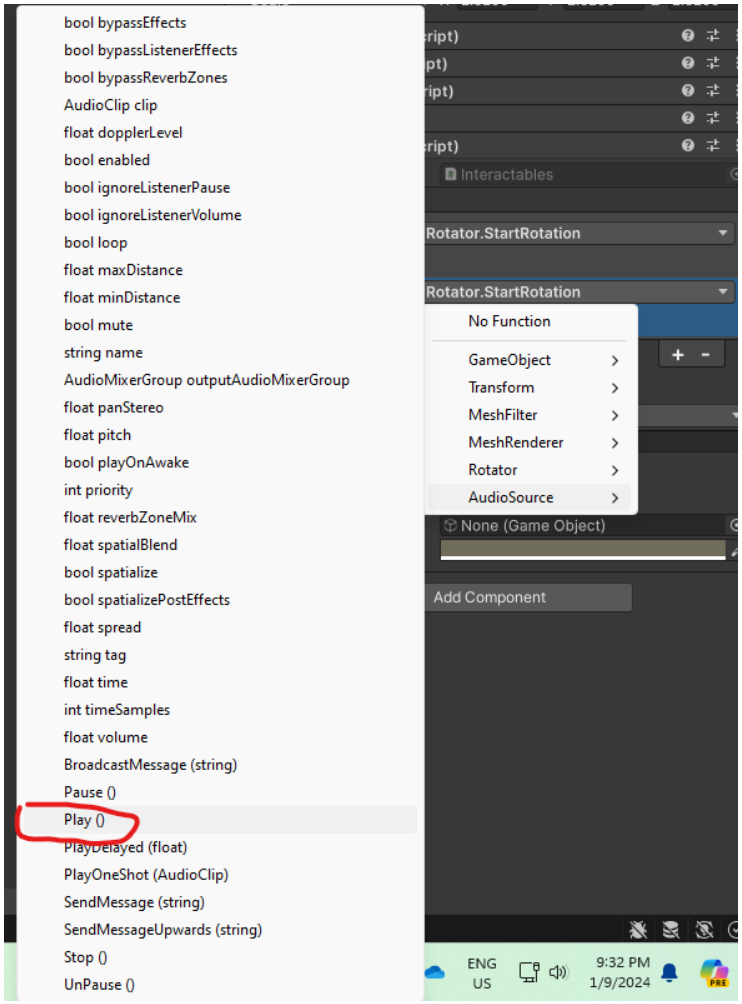
Select our **Interactable Box**, find in the **Interactables** component in the Inspector window. Press **+** to add another action into the List.



Another item has been added to the List.



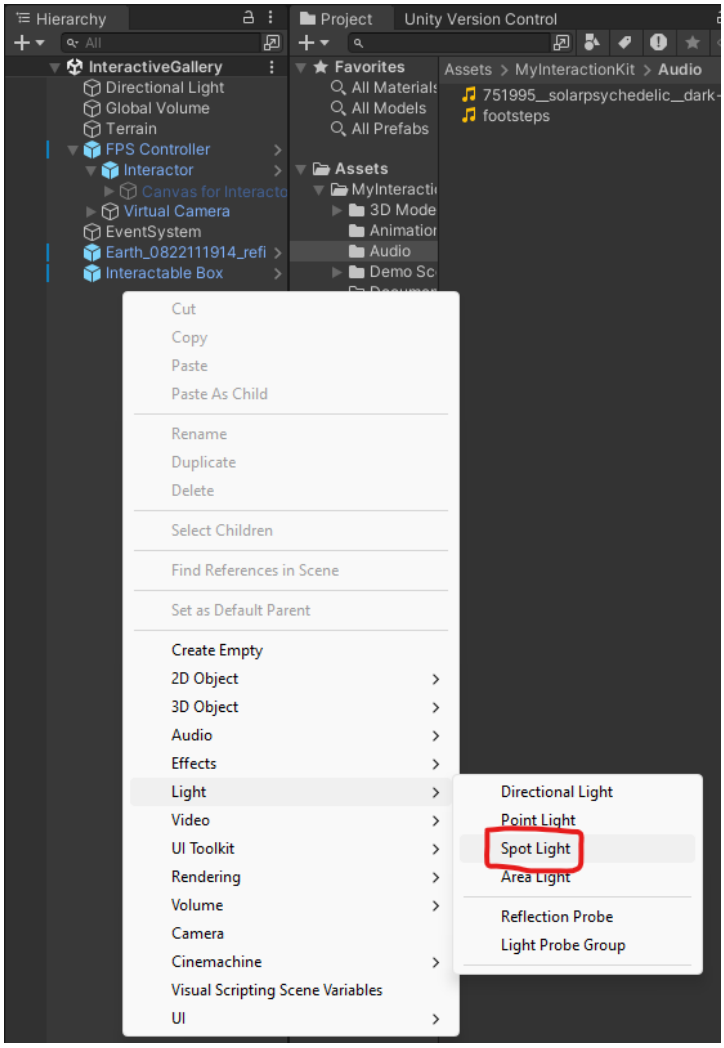
Press the **Function** dropdown. Just like before, a list will appear but this time there is Audio Source. Select **Audio Source > Play**()



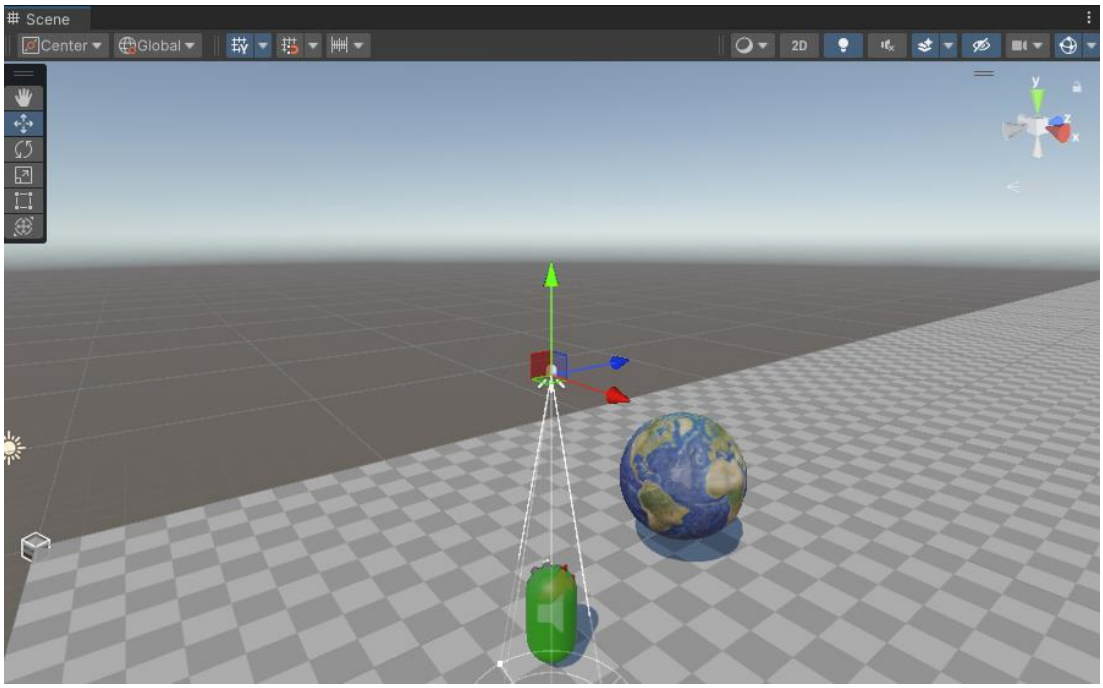
Press **Play**, when you interact with the Earth, the Earth starts to rotate, and the Music will play.

Light to set mood

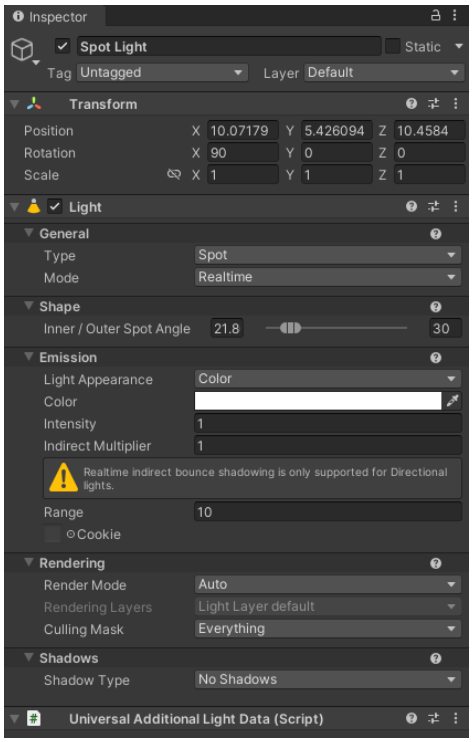
Now that we have such sombre music, let's add some light to get some sombre mood on it. Right-click on the Hierarchy window, select **Light > Spot Light**.



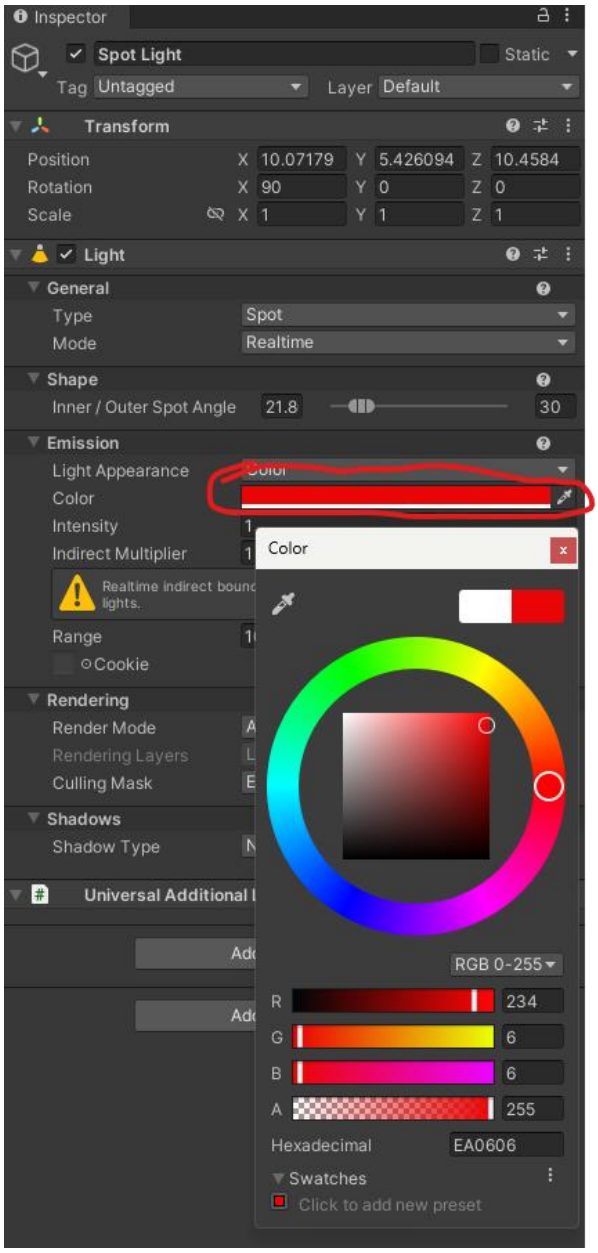
A spotlight will appear.



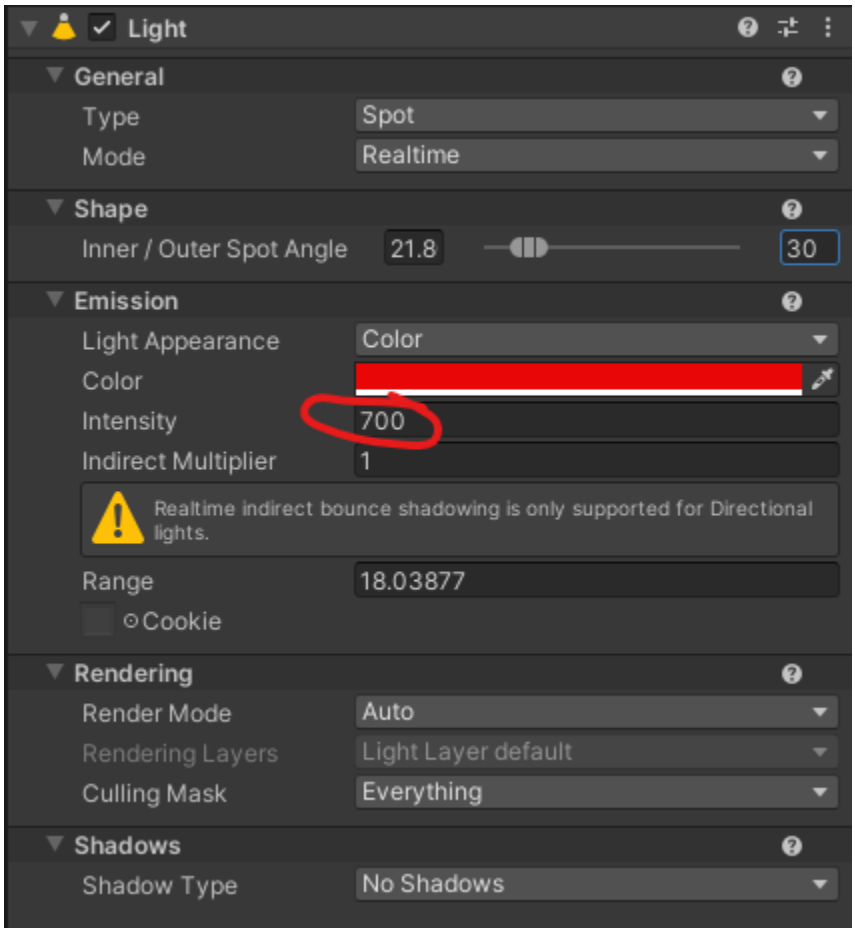
But it is not too visible. We will change the light to Red nd amp up its brightness. Select the **Spot Light**, in the **Inspector**, find the **Light component**.



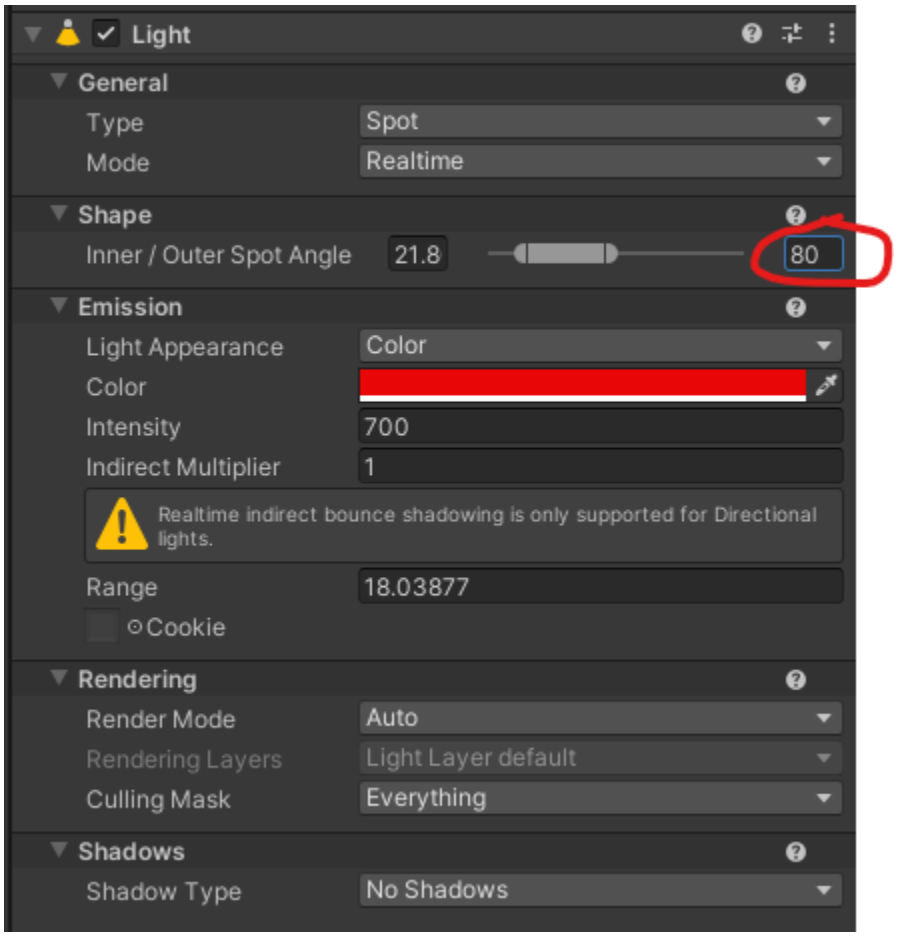
Press the Color, a color selector will appear, set it to Red



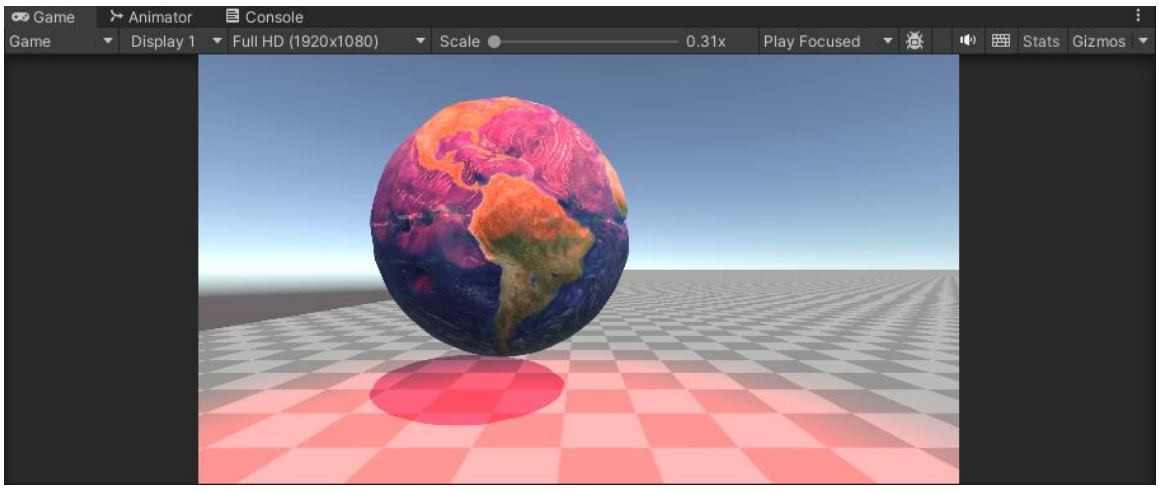
Set the **Intensity** to **700**



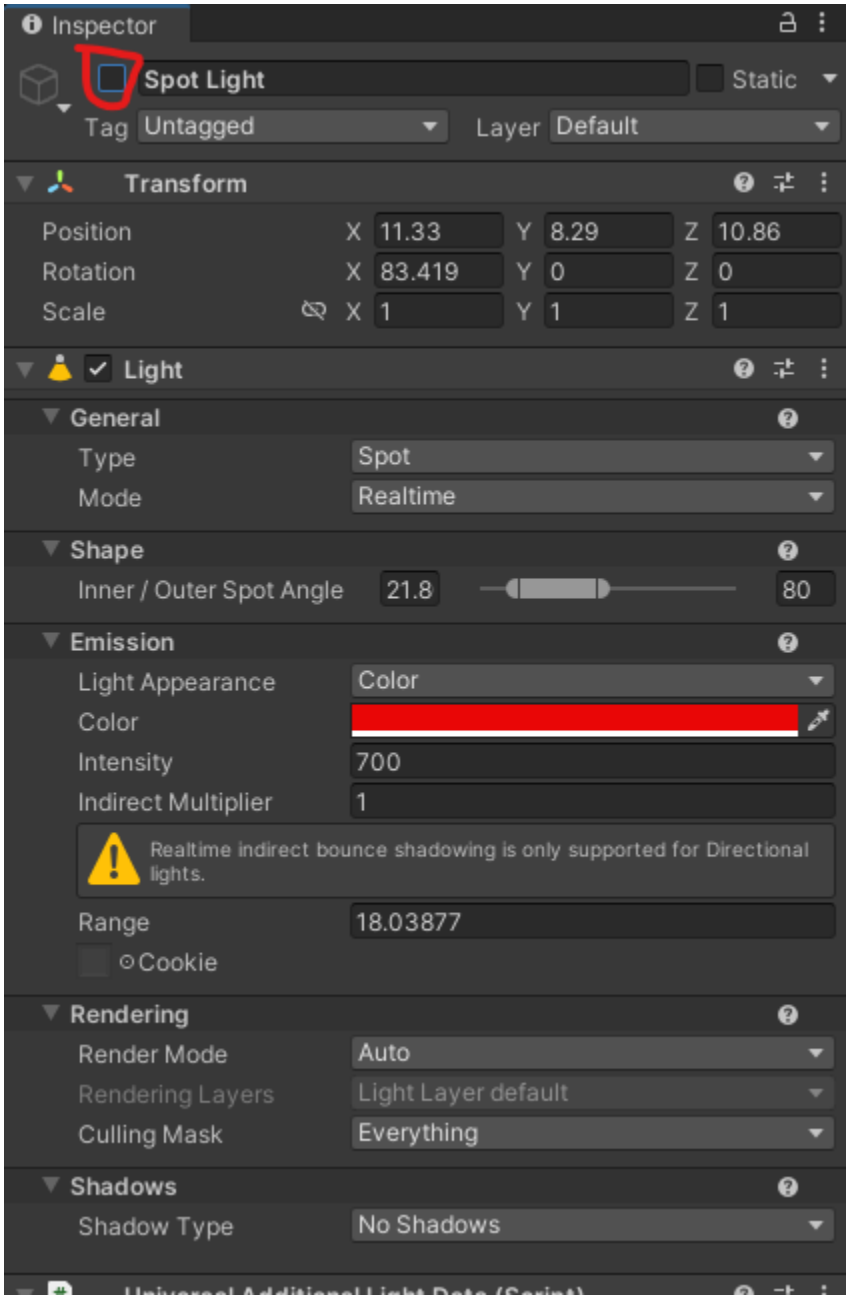
Change the **Outer Spot Angle** to **80**



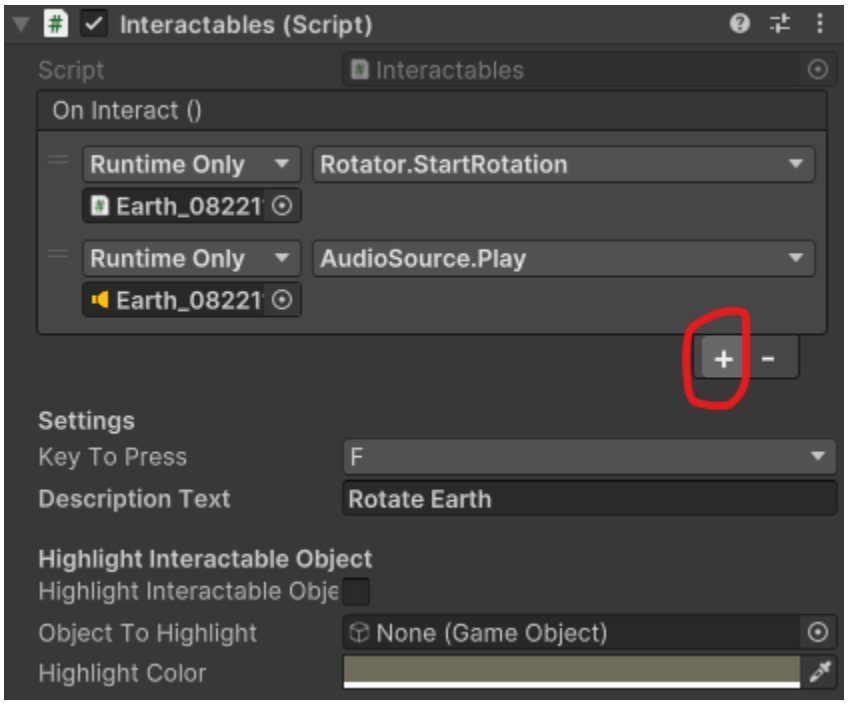
Now, its more obvious. **Move** and **Rotate** the light it so that the Earth so that it shines on Earth. It should look something like this.



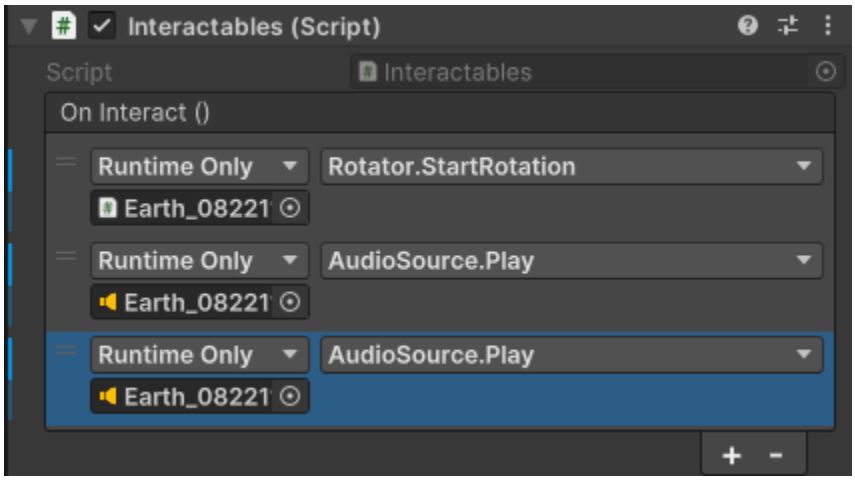
To make it only enabled when we Interact, select the **Spot Light**, and **uncheck** the **checkbox** besides its name. This will disable it.



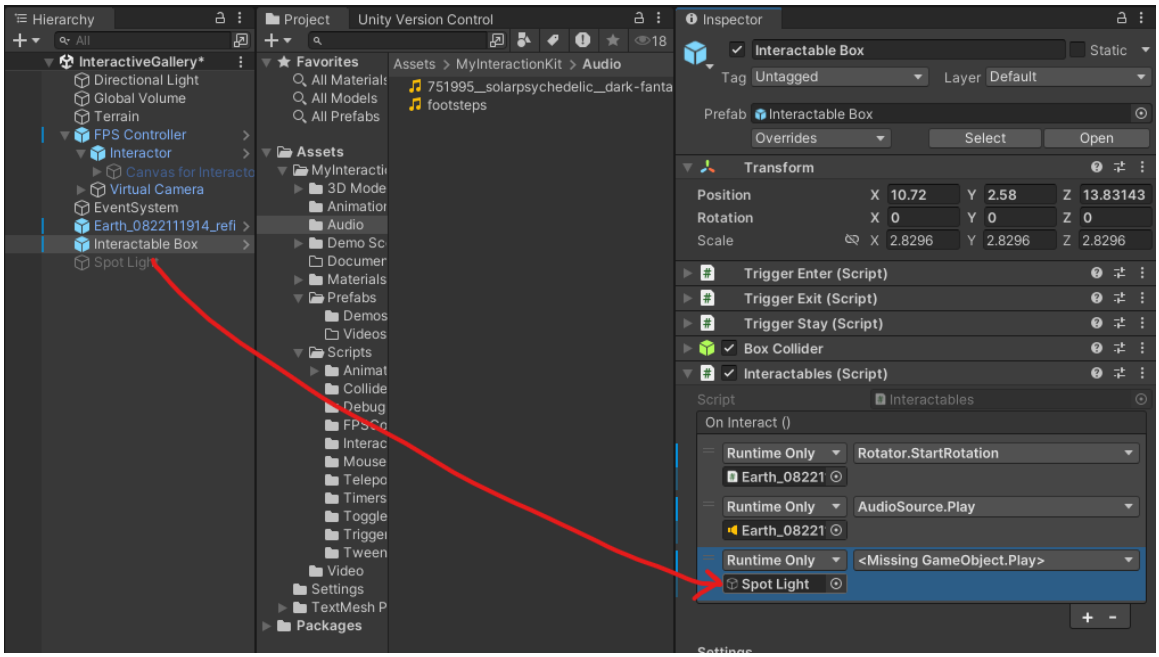
Select our **Interactable Box**, again, press **+** to add another action into the List.



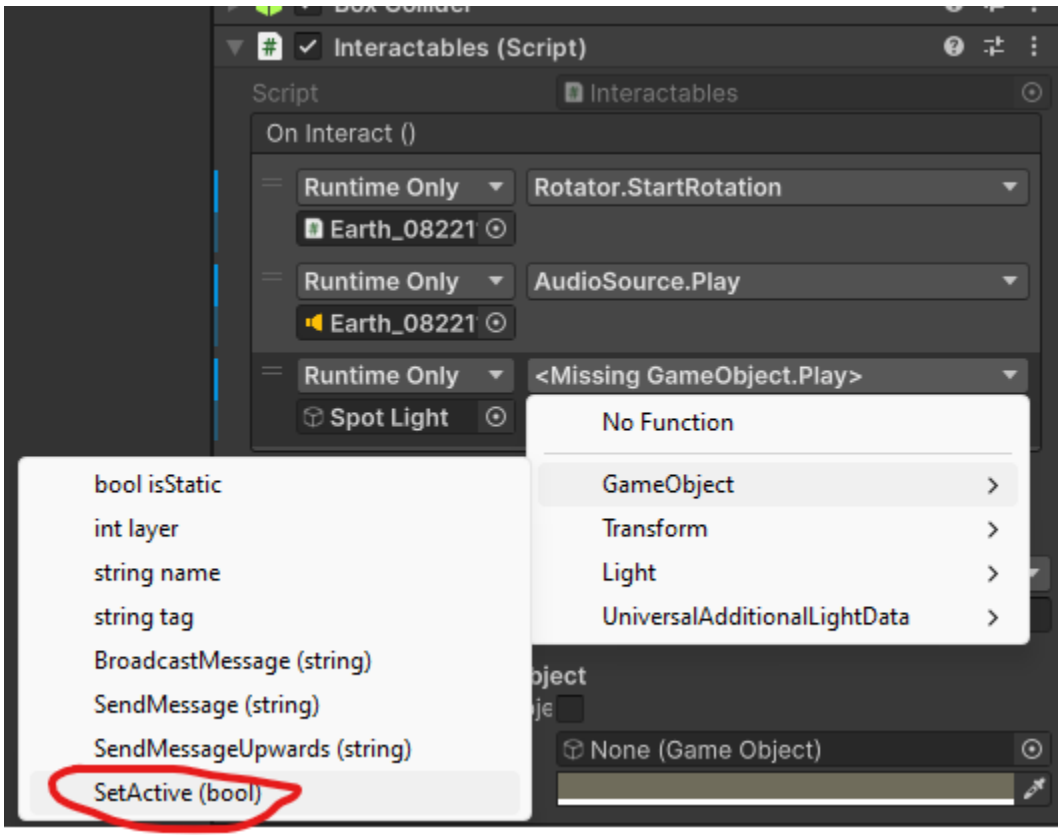
It will look like this.



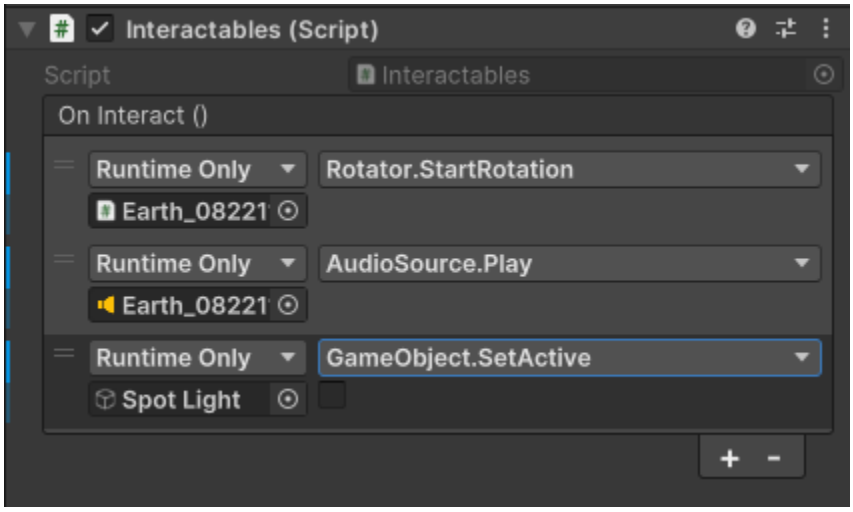
Drag and Drop the Spot Light into the Object area of the newly added item



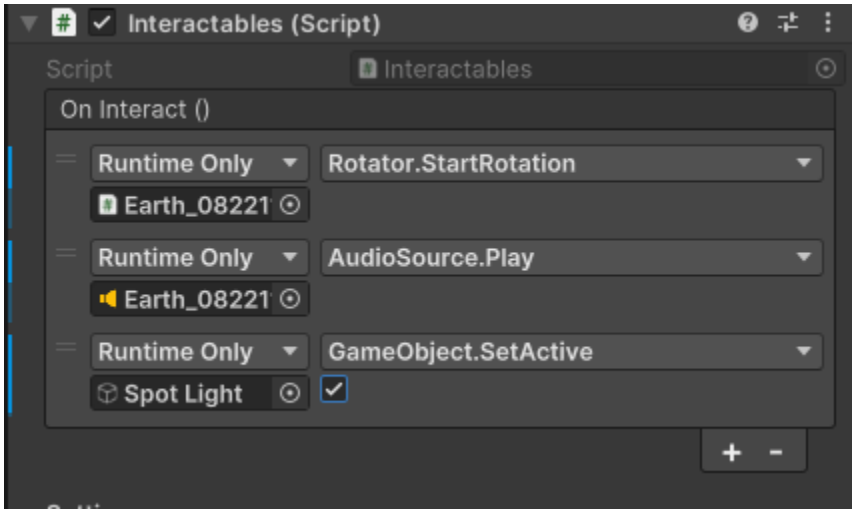
Press the Function dropdown menu. Select **GameObject > SetActive(bool)**



You will have something like this. SetActive is a Unity function to activate and deactivate a GameObject.



Check the **checkbox** in **SetActive**. You will have something like this. This means, that the **Spot Light GameObject** will be activated when we Interact with Earth.

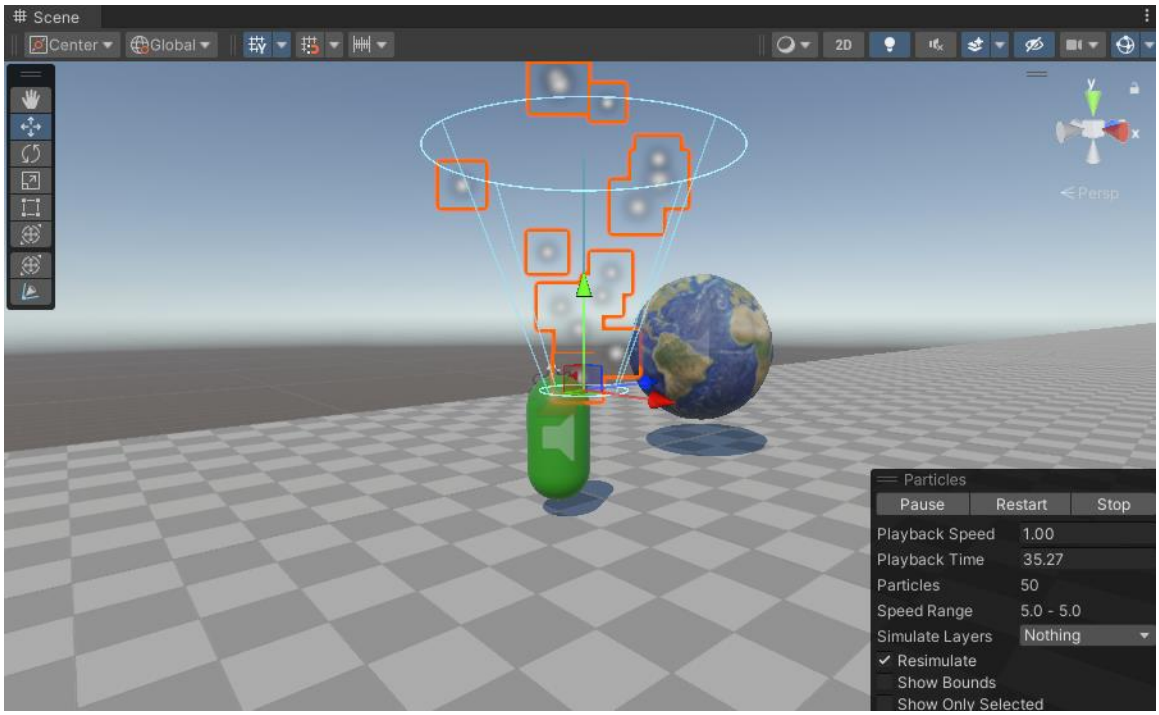


Press **Play**, when we Interact with Earth, the Earth starts spinning, the audio plays, and the red light activates.

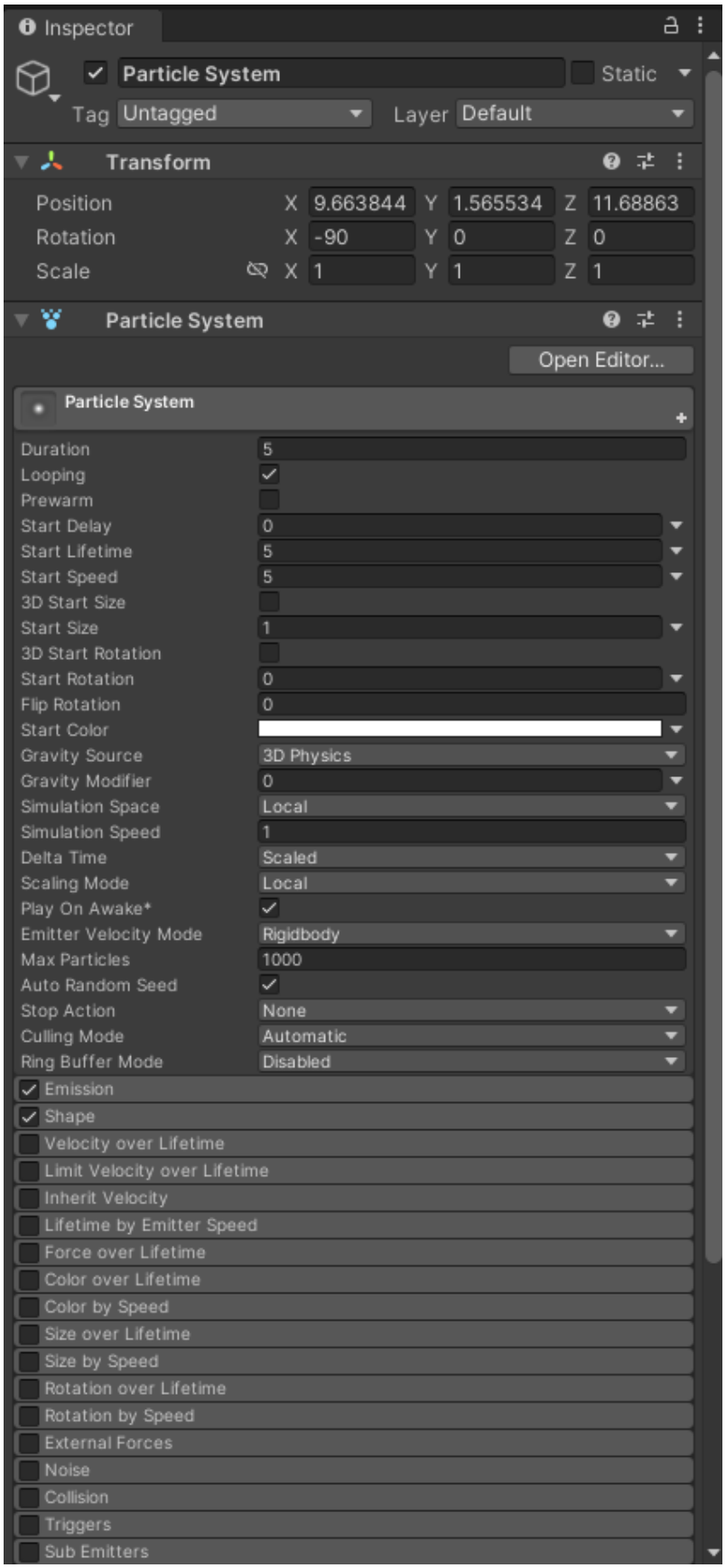
Particles for some more mood

Finally, lets add a particle to actually make it look magical. Hopefully. On the **Hierarchy**, **Right Click** and **Select Effects > Particles System**.

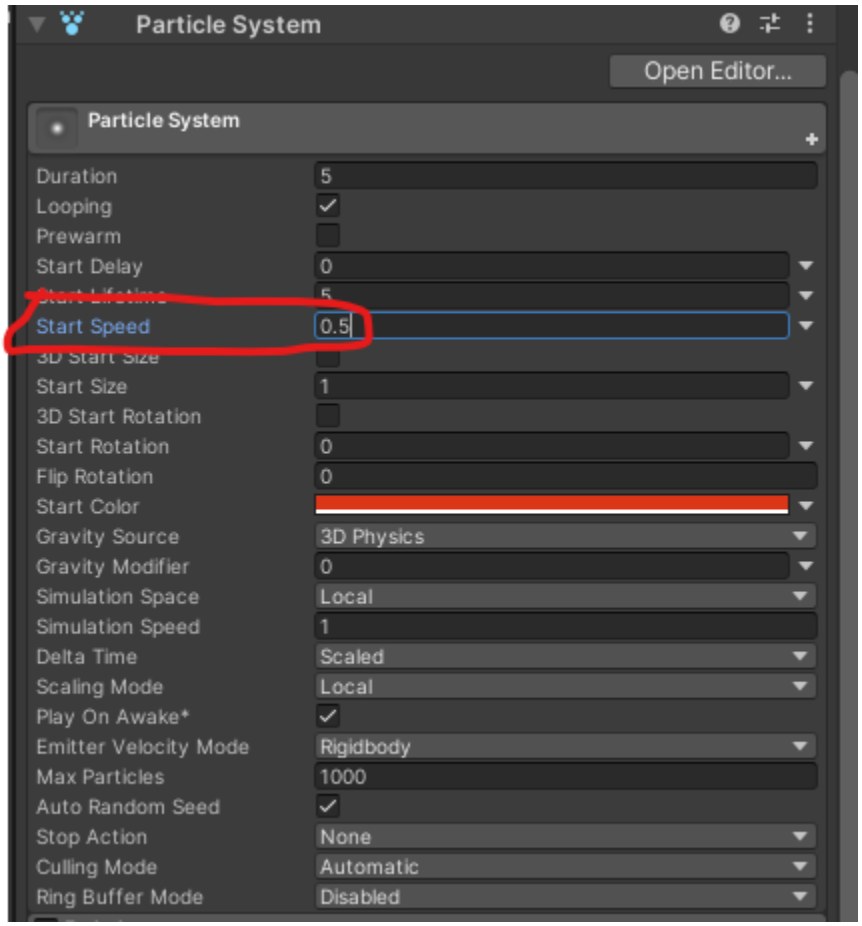
You will get something like this.



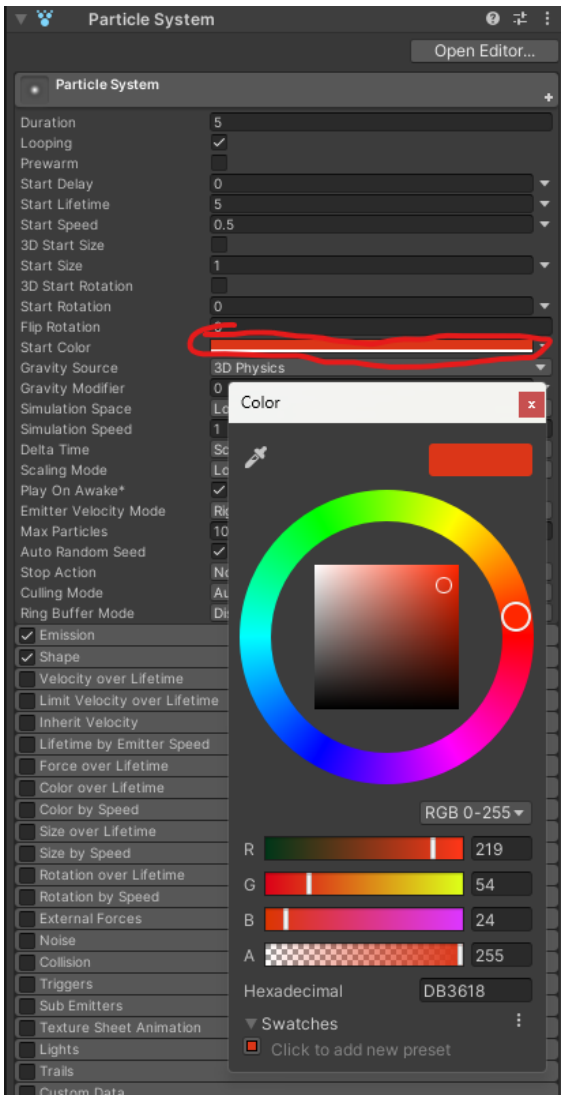
Select the **Particle System** if not yet selected, in the Inspector, find the **Particle System** component.


Inspector
Particle System
Static
Tag Untagged
Layer Default
Transform
Position X 9.663844 Y 1.565534 Z 11.68863
Rotation X -90 Y 0 Z 0
Scale X 1 Y 1 Z 1
Particle System
Open Editor...
Particle System
Duration 5
Looping
Prewarm
Start Delay 0
Start Lifetime 5
Start Speed 5
3D Start Size
Start Size 1
3D Start Rotation
Start Rotation 0
Flip Rotation 0
Start Color
Gravity Source 3D Physics
Gravity Modifier 0
Simulation Space Local
Simulation Speed 1
Delta Time Scaled
Scaling Mode Local
Play On Awake*
Emitter Velocity Mode Rigidbody
Max Particles 1000
Auto Random Seed
Stop Action None
Culling Mode Automatic
Ring Buffer Mode Disabled
Emission
Shape
Velocity over Lifetime
Limit Velocity over Lifetime
Inherit Velocity
Lifetime by Emitter Speed
Force over Lifetime
Color over Lifetime
Color by Speed
Size over Lifetime
Size by Speed
Rotation over Lifetime
Rotation by Speed
External Forces
Noise
Collision
Triggers
Sub Emitters

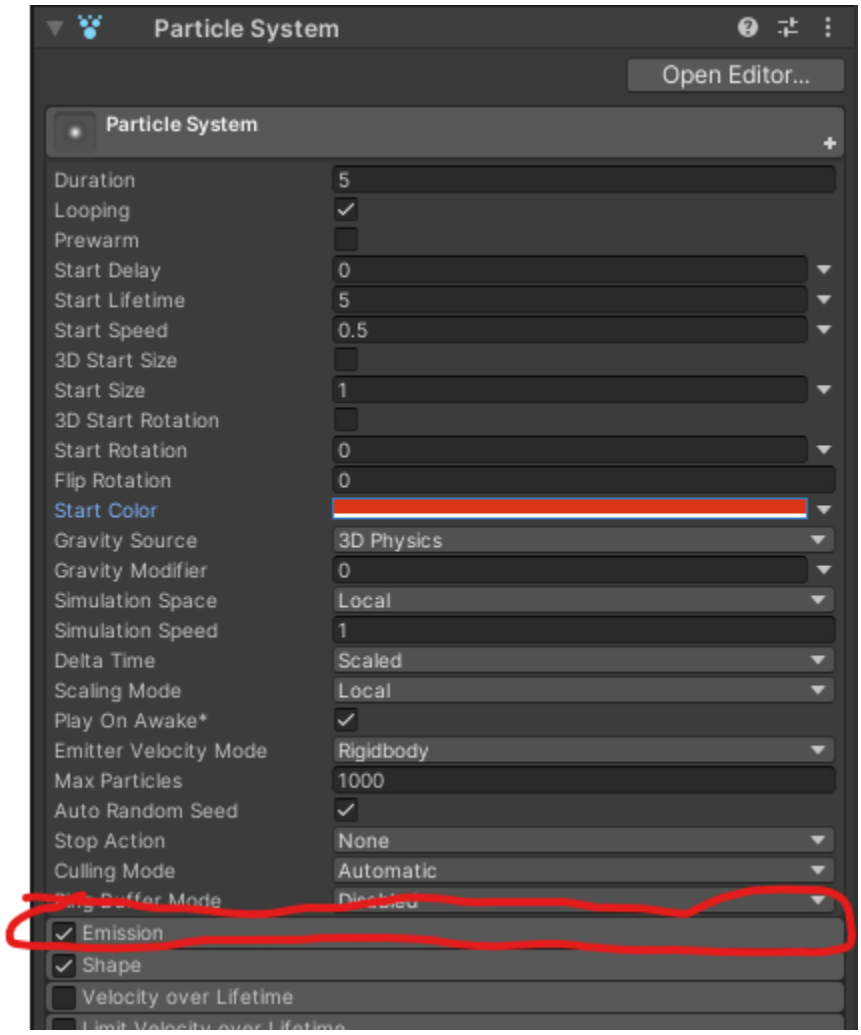
The particles are too fast so lets slow it down. Change the **Start Speed** to **0.5**.



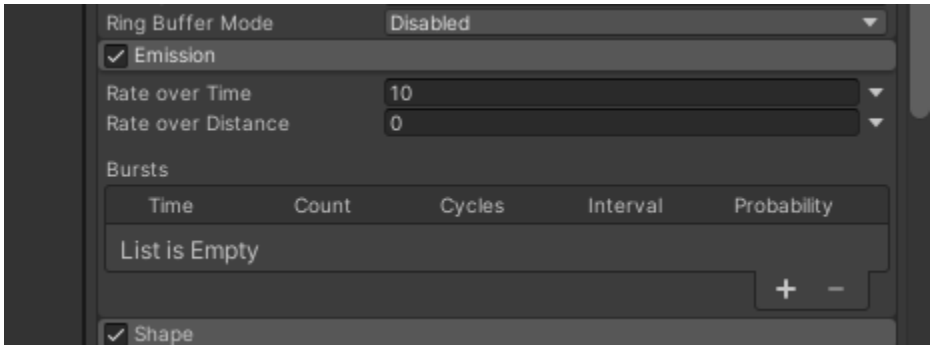
Lets change the color to **Red-Orange**. Press the color at **Start Color**. A color selector appears similar to before, set the color to **Red-Orange**.



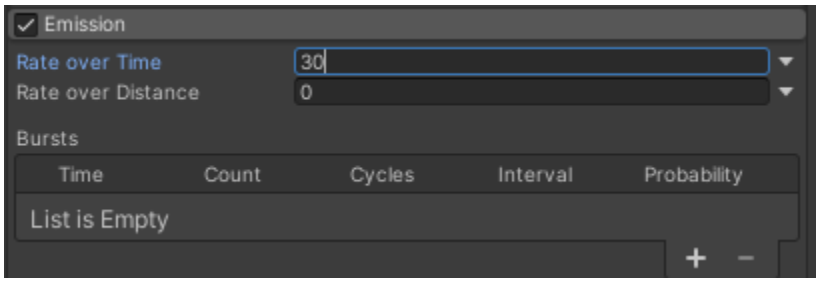
Press the Emission category.



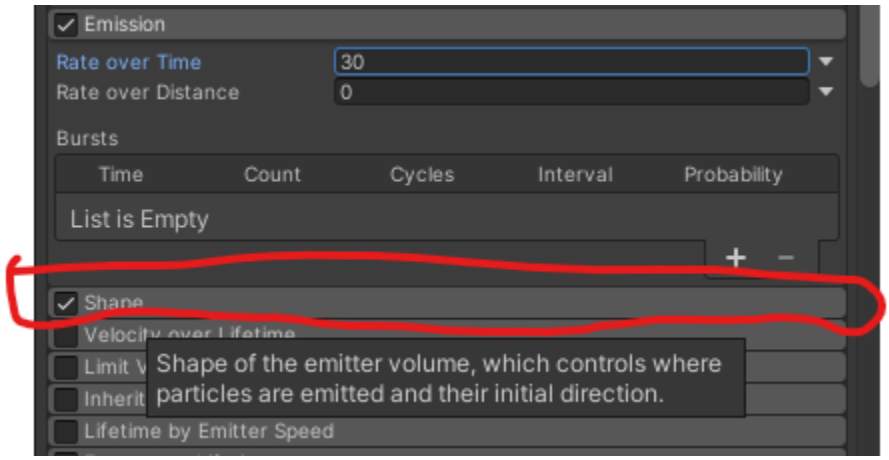
The Emissions settings will be expanded like this.



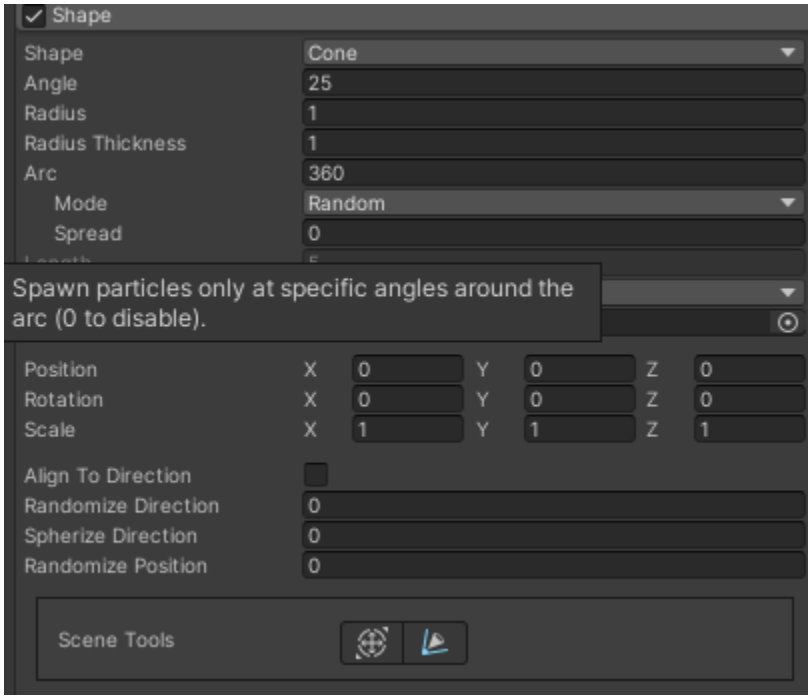
We need more particles, change the **Rate over Time** to **30**.



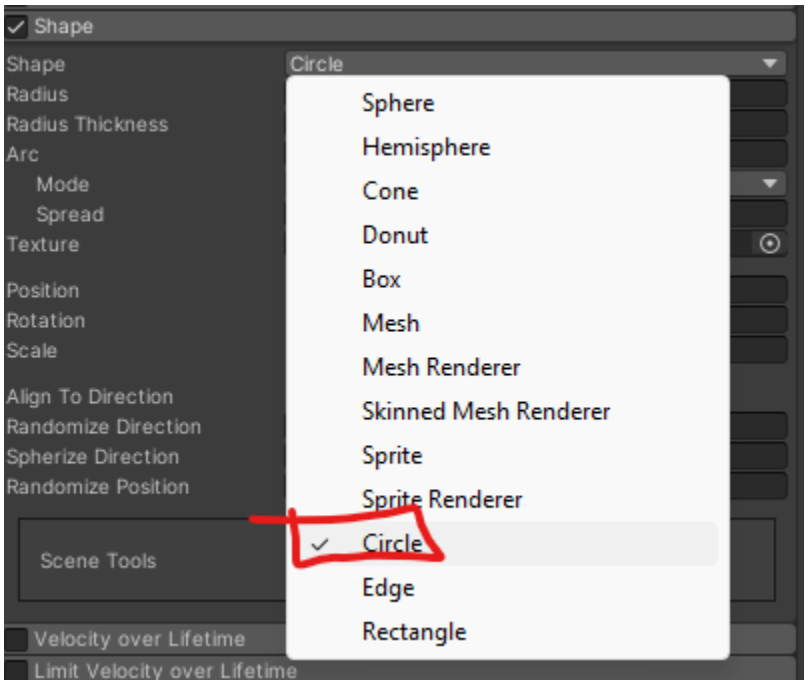
Then, click on Shape category.



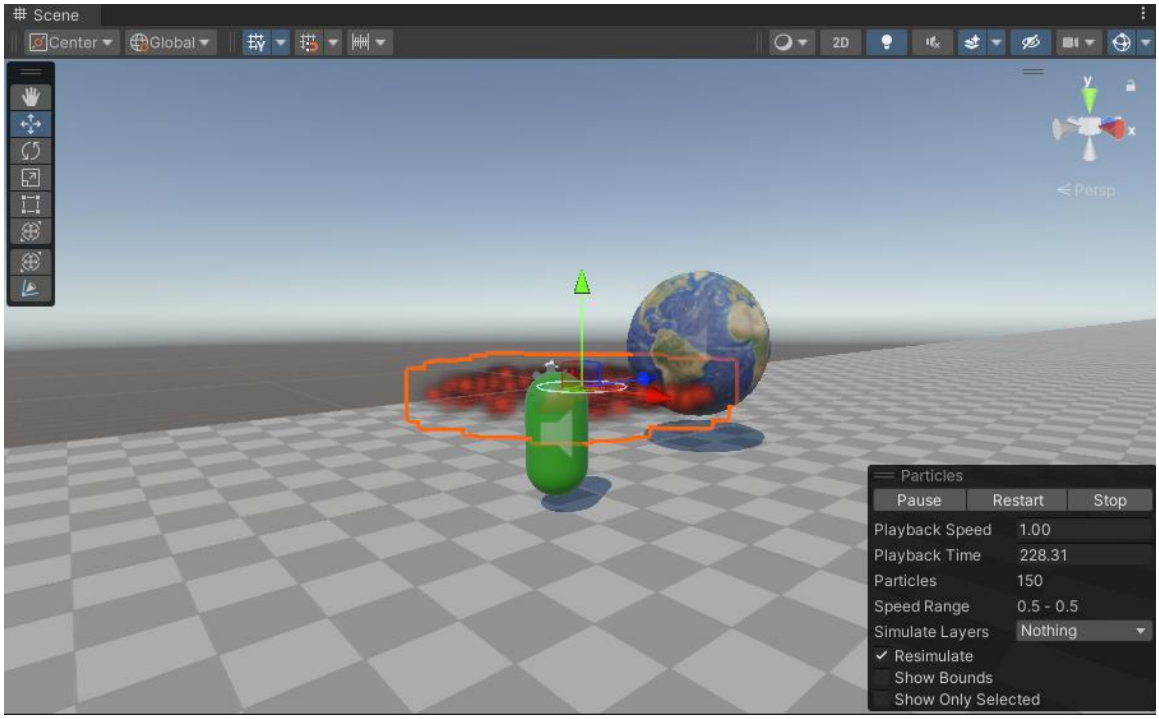
The options will expand and look like this.



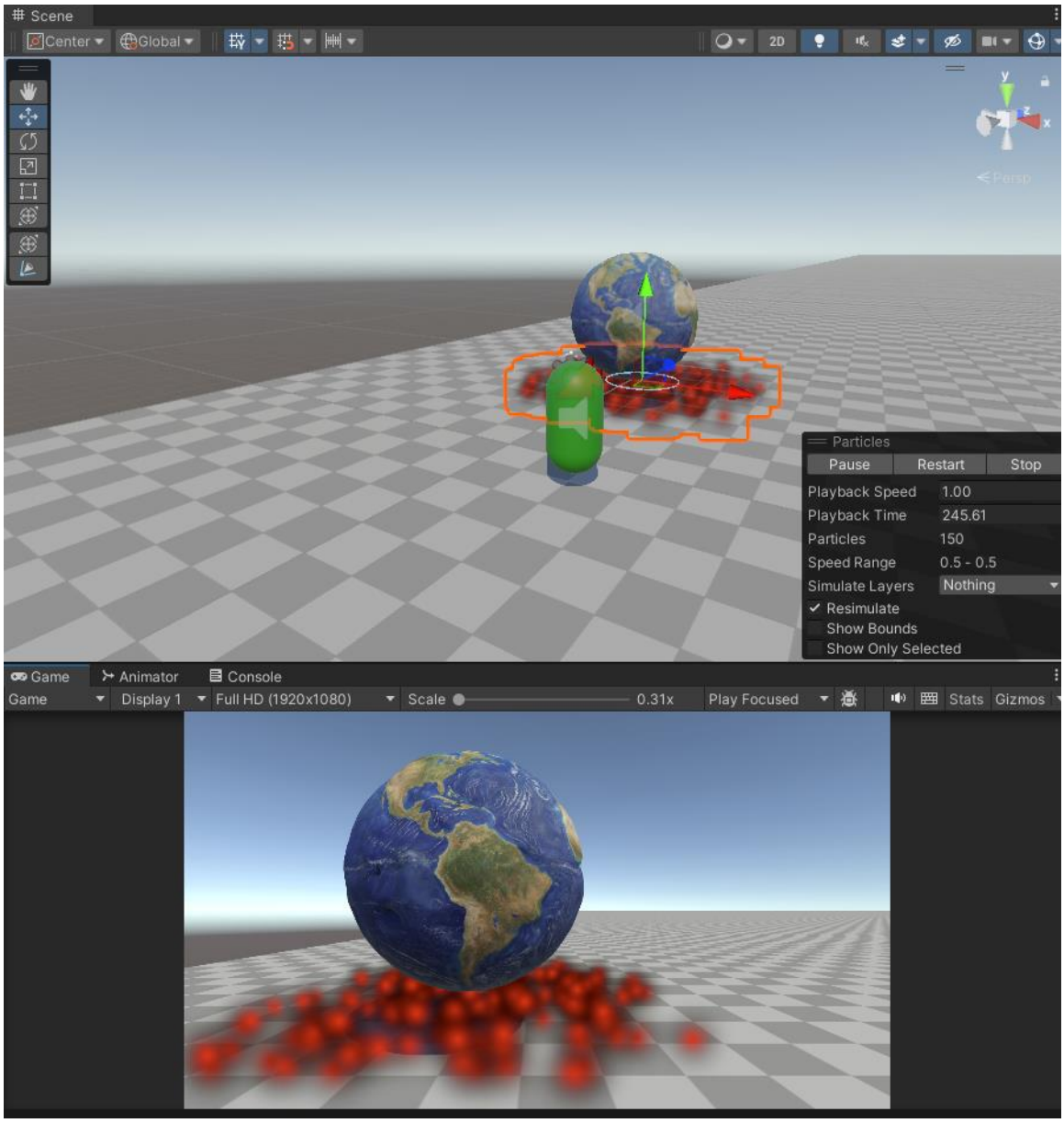
Click the Shape dropdown and select **Circle**



At the end, we have something like this.



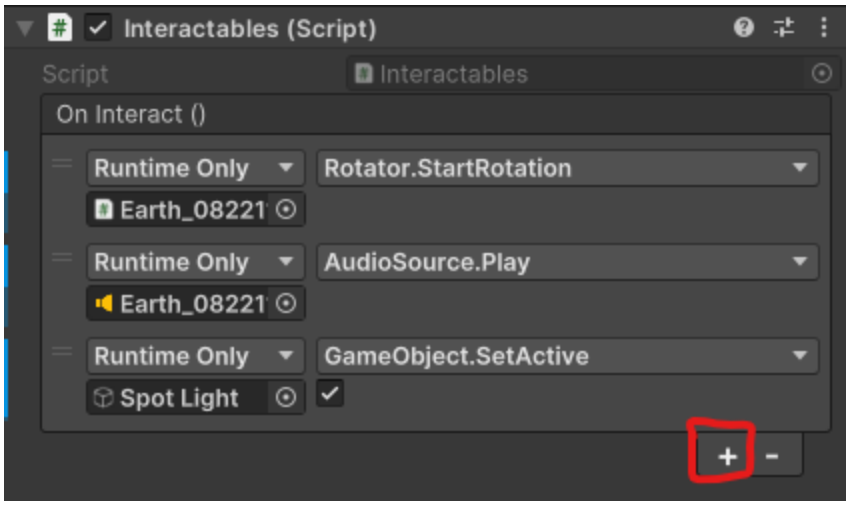
Move the Particle System and put it under Earth. It will look something like this.



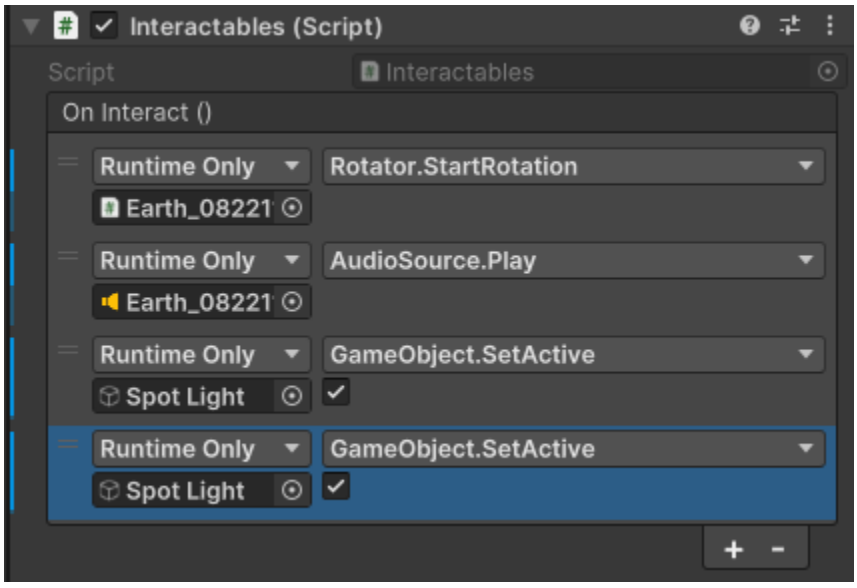
Making it work with the others

Next, we need to stop it from starting automatically. In the Particle System component, uncheck **Play on Awake**.

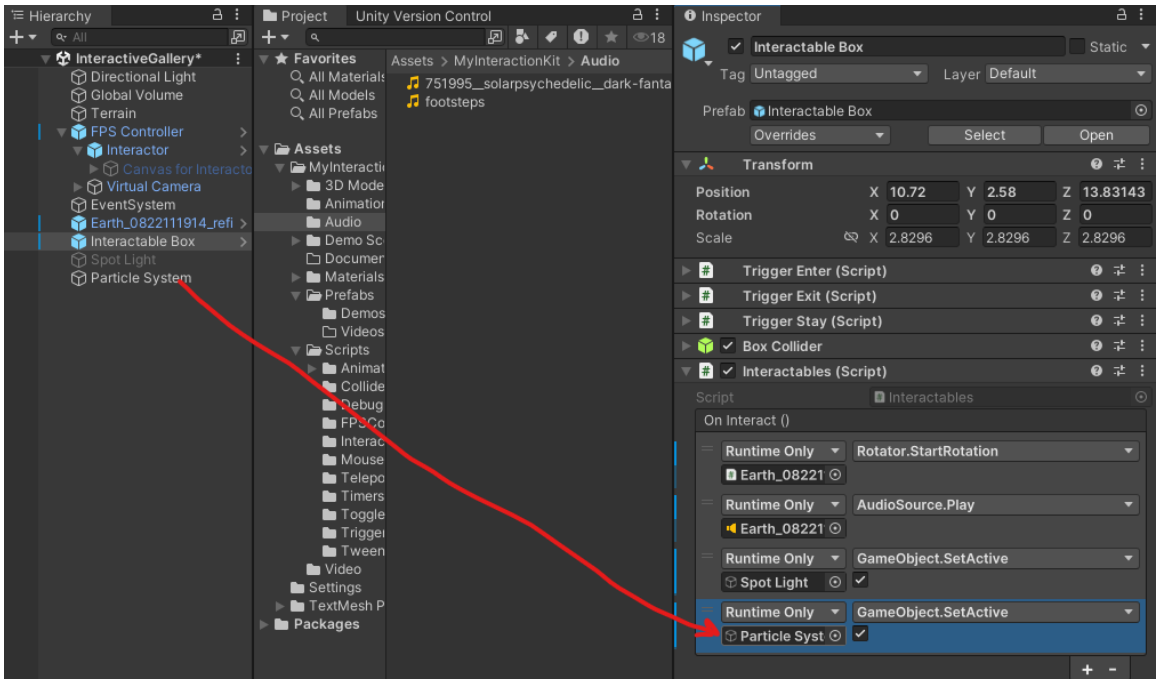
Select our **Interactable Box**, in the **Inspector**, find the **Interactables** component and press **+** to add another action to the list.



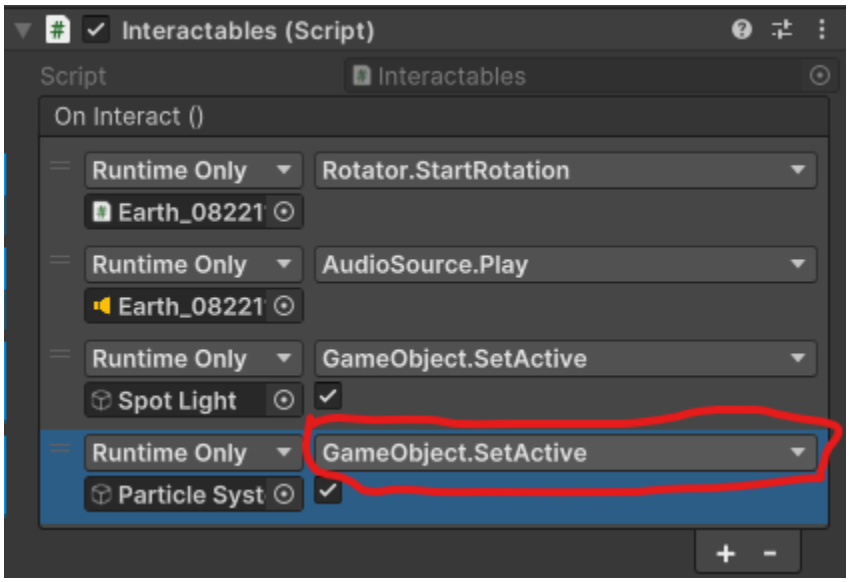
It will look like this.



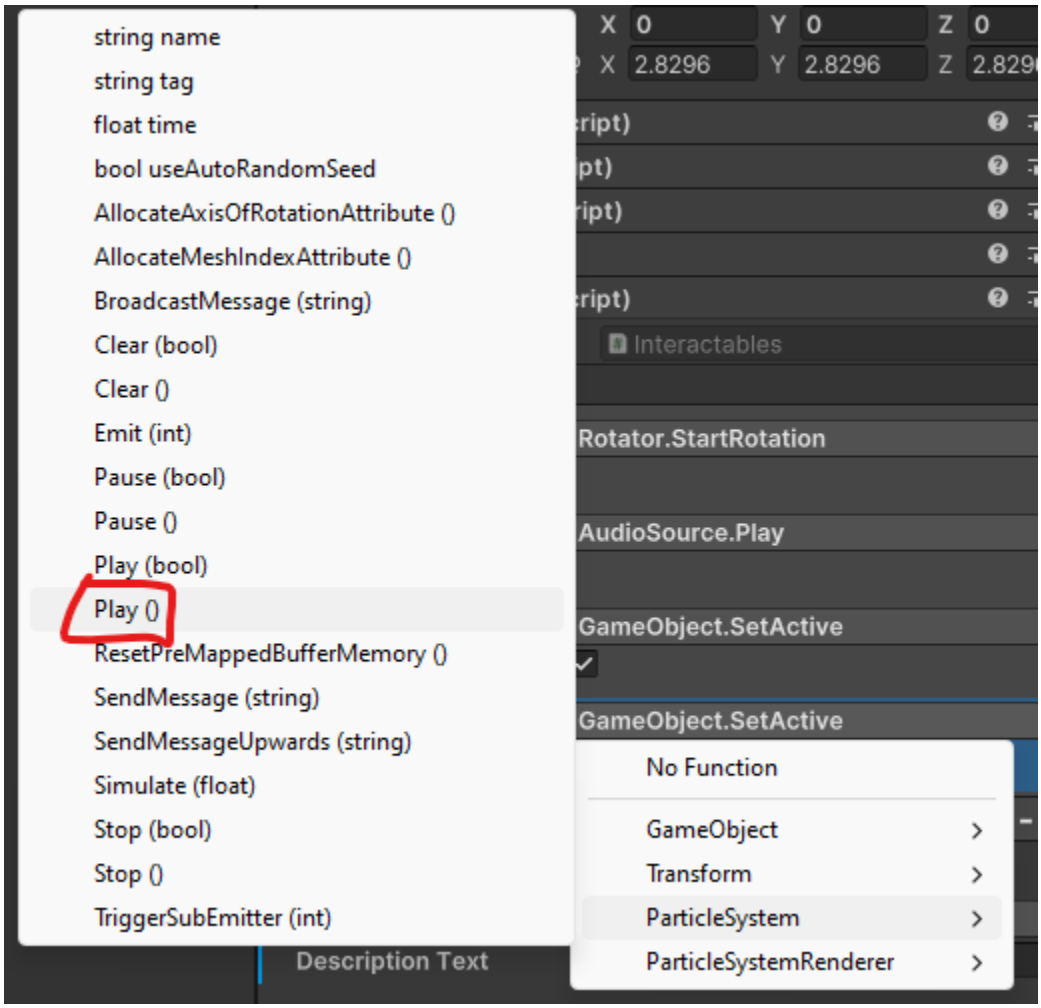
Drag and Drop the **Particle System** to the Object area of the new item.



Press the Function dropdown of the new item.



Select **Particle System > Play()**



Finally, press **Play** to test. When you approach Earth and Interact with it:

1. The Earth rotates
2. Music plays
3. Light turns on
4. Particles starts appearing

This is the end of this tutorial. I hope you have fun!

Screenshot System

Taking A Screenshot

1. In the Unity Editor menu bar, go to Tools > Screenshot! > Take Screenshot

2. If the capture folder is not set, the screenshot will be saved in My Documents/Screenshot!

folder

Setting Screenshot Folder

1. In the Unity Editor menu bar, go to Tools > Screenshot! > Set Screenshot Folder

2. An open folder dialog appears, select the folder that you want to save the screenshots to and

press OK

Reset To Default

This will reset the screenshot folder back to default settings.

1. In the Unity Editor menu bar, go to Tools > Screenshot! > Reset To Default

Add iOS AppStore Screenshot Sizes

Adds iOS App Store Screenshot Sizes to the shortcuts in game view.

1. In the Unity Editor menu bar, go to Tools > Screenshot! > Add iOS App Store Screenshot Sizes

2. An conformation dialog appears, press OK

Warning: Using this button more than once could result in duplicate shortcuts in GameView. If that happens, Right-click the duplicate shortcut and press Delete

Add Unity Asset Store Screenshot Sizes

Adds iOS App Store Screenshot Sizes to the shortcuts in game view.

1. In the Unity Editor menu bar, go to Tools > Screenshot! > Add Unity Asset Store Screenshot Sizes

2. An conformation dialog appears, press OK

Warning: Using this button more than once could result in duplicate shortcuts in GameView. If that happens, Right-click the duplicate shortcut and press Delete

How do I Removing MyInteractionKit?

Delete the extension in the Assets/MyInteractionKit folder