UNDER WATER TRIGGER

## Coding Challenge: Use a trigger for the water panel

In the lesson related to adding a water effect, we wrote code that toggled on a Panel GameObject when the camera was below the water and toggled off the Panel when the camera was above the water.

This solution has limitations. For instance, what if there is more than one body of water? Which water do we attach to the script to determine when the water effect should occur? If all bodies of water have the same y value, then any of them would work. However, our script would need to be modified to handle multiple water bodies with different y values.

Additionally, our current code seems inefficient in that it is constantly activating or deactivating the Panel each time through the Update method. However, we only need to activate or deactivate the Panel at the moment the player enters or exits the water.
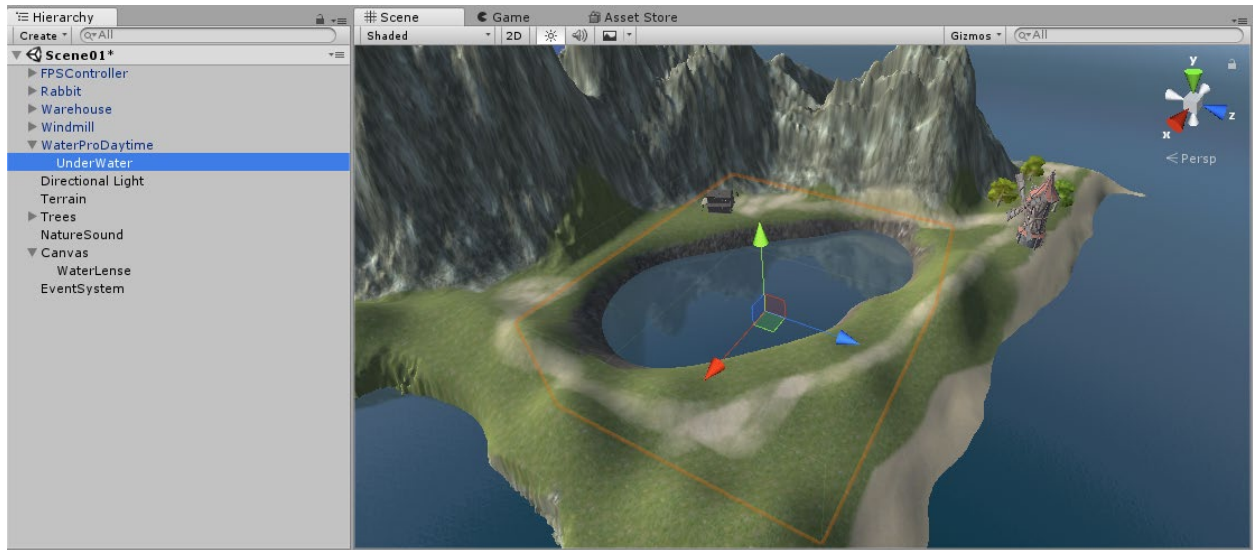
An alternate solution that addresses these concerns is to use a trigger.

First, create a Cube as a child of one of your bodies of water.
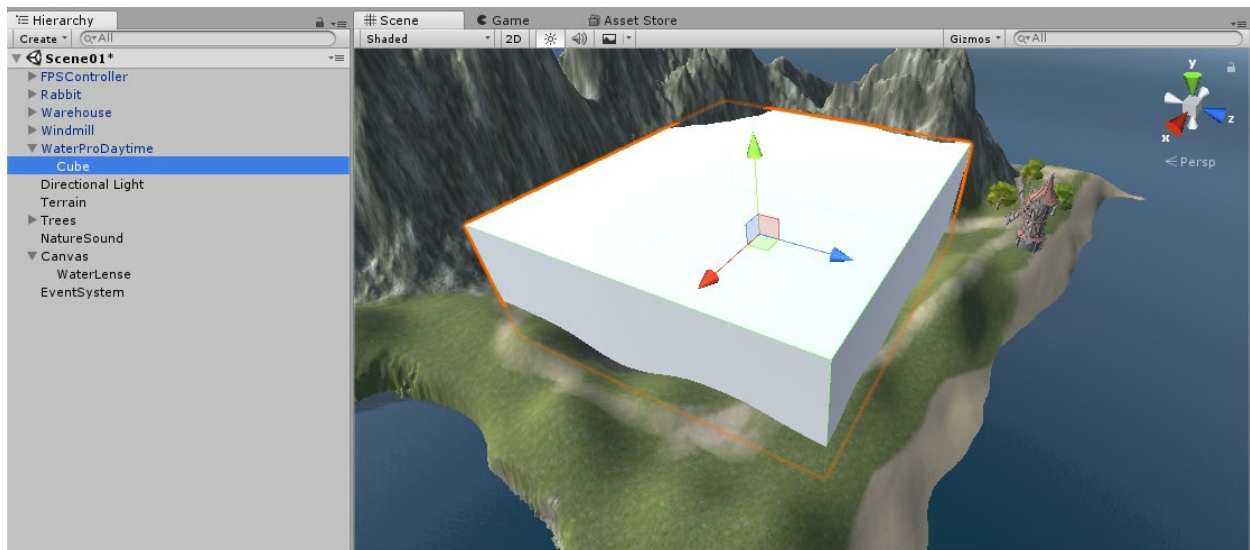
Give the Cube a meaningful name. This will be the area that is considered underwater.

Move the Cube downward and stop just when the water becomes visible.

Remove the Cube and Mesh Renderer components, leaving only the Transform and the Box Collider.

Check the "Is Trigger" box. Remember, this will allow you to use the OnTriggerEnter method to detect collisions while also allowing the colliding objects to pass through each other.

Use the Edit Collider button if you need to view or adjust the size of the collider, though be sure to keep the top side of the collider just under the level of the water.
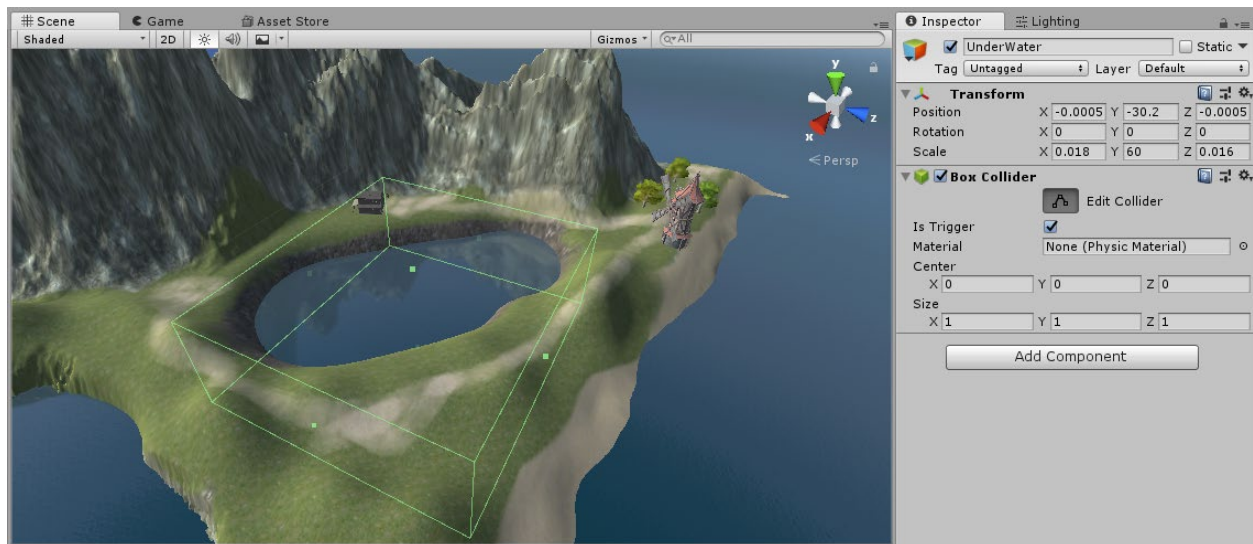
Remove the CameraBehavior script from the FirstPersonCharacter camera.

Add the CameraBehavior script to the GameObject you are using to detect collisions with the water.

In code, remove the public property that is being used to reference the Water object and delete the Update method.

Drag the Panel being used for the water effect to set the waterLense property of the script.


If you play the game at this point, the water effect should be immediate, before you enter the water.

Add a Start method, and in that method deactivate the WaterLense Panel.

Add an OnTriggerEnter method that activates the WaterLense Panel only when the object colliding with the UnderWater collider is the FPSController. You may want to refer to the LoadScene method where we created a similar collider as a door to load a new scene.
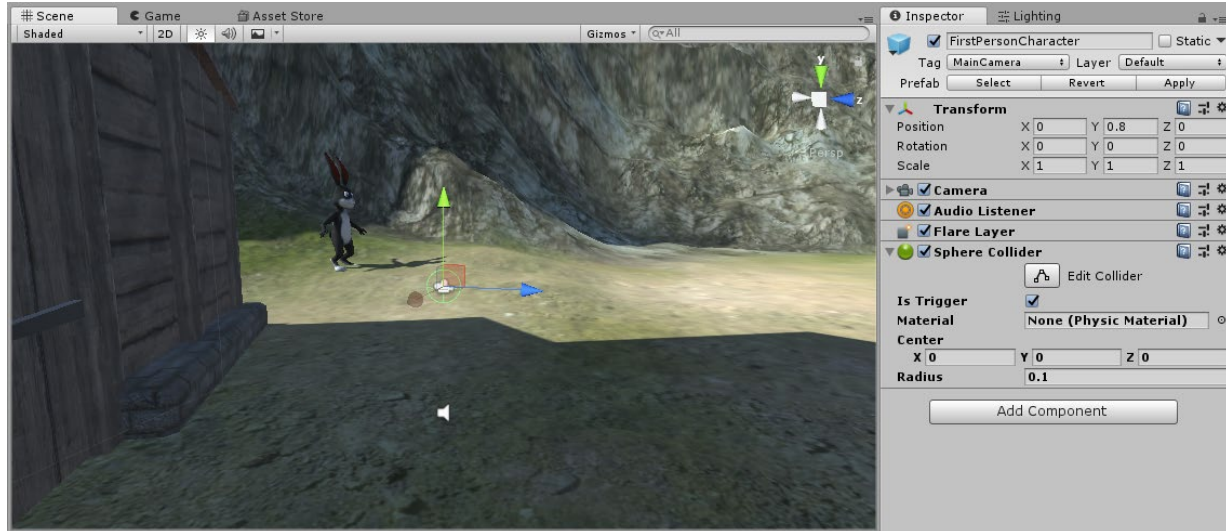
At this point when you play the game, the water effect should not be immediate, and should occur when your player enters the water. However, the collider of the player that is being used is larger than the camera, so the effect occurs as soon as the bottom of the player (the feet) enter the water; even though the camera itself is still above the water.
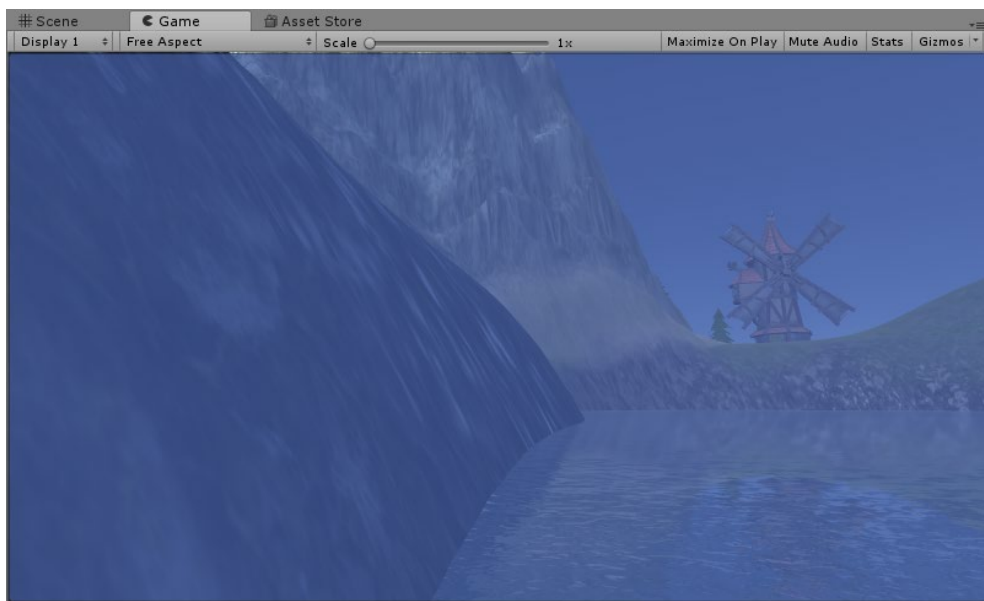
Add a Sphere Collider component to the FirstPersonCharacter camera.

Set the Radius to be very small. Consider the collider represents the head, or eyes, of the player.

Check the "Is Trigger" box to ensure the collider never causes any physical collisions with other objects.


Update the code in the OnTriggerEnter method to execute only when the collision occurs with the FirstPersonCharacter camera.

At this point when you play the game, the water effect should occur when your player's head enters the water. However, when you come up out of the water, the water effect is still in effect.

Add an OnTriggerExit method which deactivates the WaterLense Panel when the FirstPersonCharacter camera exits the UnderWater collider.

Expand the collider to include all space under the water it associated with. In my Scene, I only have one very large body of water, so my collider should be large enough to cover all underwater areas.
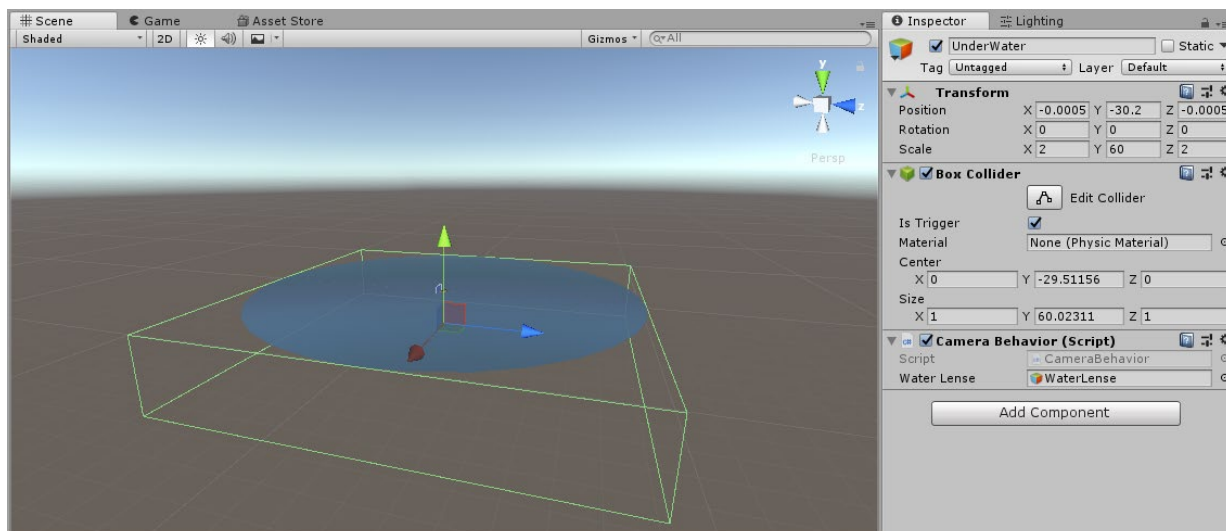
Audio Enhancements

After your water effect is working via the UnderWater collider, use the OnTriggerEnter and OnTriggerExit methods to:

- Play a sound effect when entering and exiting the water
- Turn the volume of ambiance sounds to zero while under the water
- Lower the volume of background music down while under the water