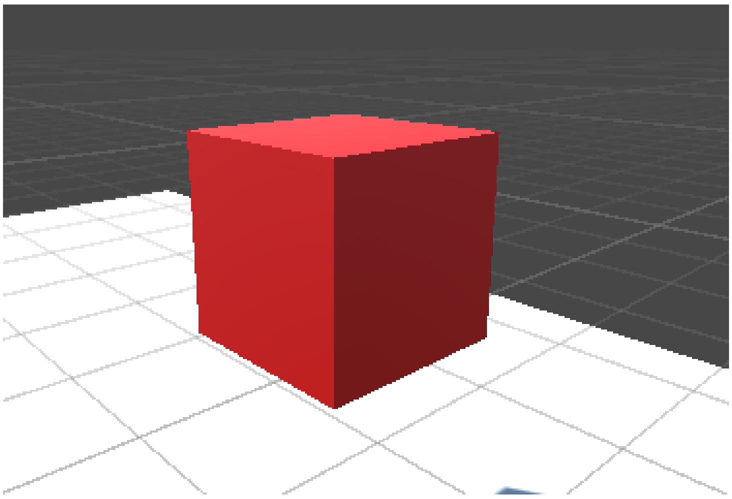
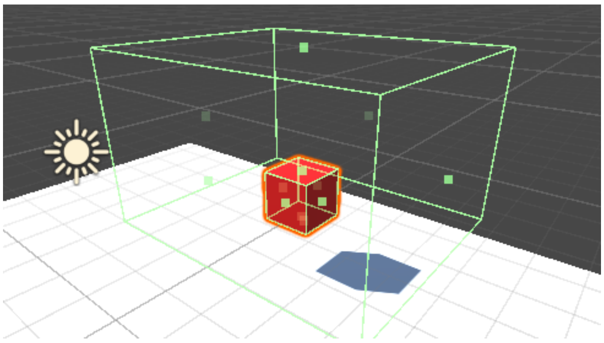
Unity Death plane tutorial

Making the death zone

1. Create a cube in your scene
2. (Optional) Use a bright red material to highlight the block so that when placing the death plane, it's easier to see where it is

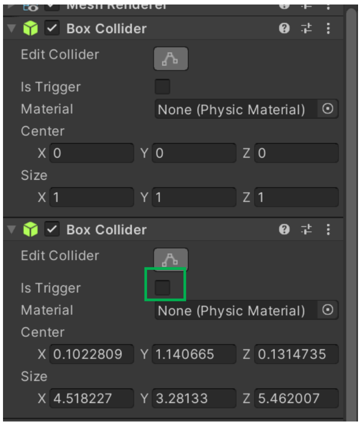


3. Go to the inspector for the cube and click add component
4. Attach a box collider (The cube will now have 2)
5. Stretch out the new box collider to as big as you want your death plane (later when the death plane is a prefab the size can be changed)



6. Click to set the new box collider as a trigger by ticking the Is Trigger box

Remember to select Is Trigger for the large box collider, you can tell which one you need to switch to a trigger by looking at the sizes of the box colliders. The box with numbers that are not 1, 1, 1 is the box you change.



Mesh Renderer

☑ Box Collider

Edit Collider

Is Trigger □

Material None (Physic Material)

Center X 0 Y 0 Z 0

Size X 1 Y 1 Z 1

☑ Box Collider

Edit Collider

Is Trigger □

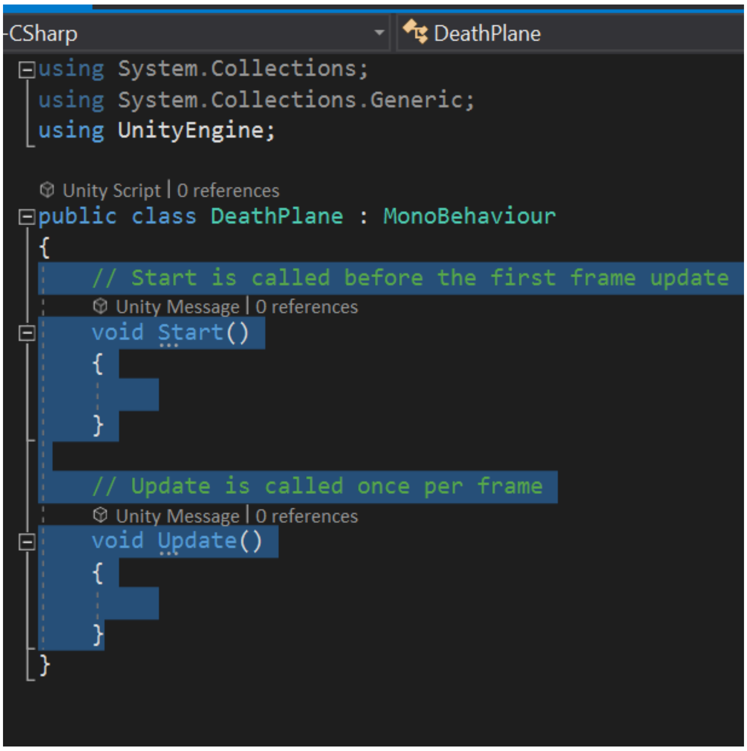
Material None (Physic Material)

Center X 0.1022809 Y 1.140665 Z 0.1314735

Size X 4.518227 Y 3.28133 Z 5.462007

Triggering the death zone

1. Go to the cube and in the inspector click add component
2. Type in and add a new script called DeathPlane
3. Remove this code

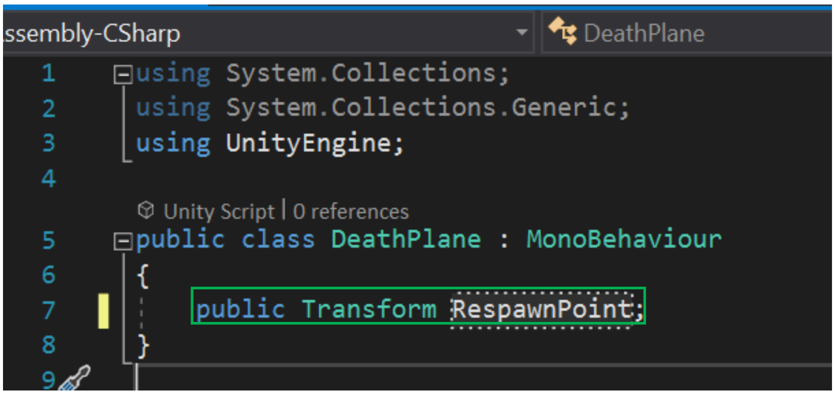


```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

Unity Script | 0 references
public class DeathPlane : MonoBehaviour
{
    // Start is called before the first frame update
    Unity Message | 0 references
    void Start()
    {
        
    }

    // Update is called once per frame
    Unity Message | 0 references
    void Update()
    {
        
    }
}
```

4. Copy this code

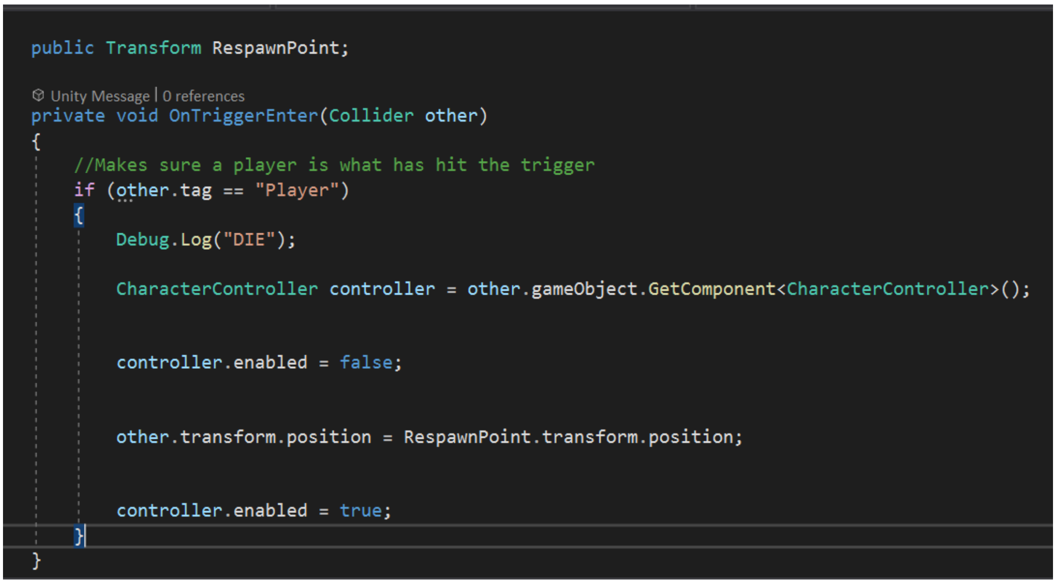


```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

Unity Script | 0 references
public class DeathPlane : MonoBehaviour
{
    public Transform RespawnPoint;
}
```

This respawn system will teleport the player to a gameobject that we set as our reset point therefore we need to write in code that we are going to set the spawn point

5. Copy this code



```
public Transform RespawnPoint;

Unity Message | 0 references
private void OnTriggerEnter(Collider other)
{
    //Makes sure a player is what has hit the trigger
    if (other.tag == "Player")
    {
        Debug.Log("DIE");

        CharacterController controller = other.gameObject.GetComponent<CharacterController>();


        controller.enabled = false;


        other.transform.position = RespawnPoint.transform.position;


        controller.enabled = true;
    }
}
```

We are starting with an IF statement that just checks if what entered the death plane is indeed the player. The first thing it will do is just send a message to the console saying Die so that we know the code is working and the trigger is detecting the player.

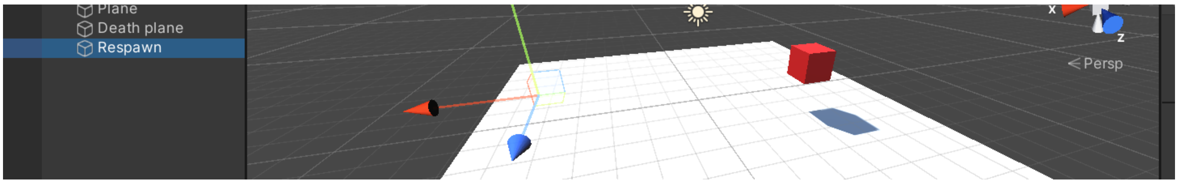
Afterwards we need to get the movement script from the player so that when we go to respawn the player they cannot move and mess up the respawn furthermore it just ensures the movement script wont get in the way of our death plane. Then afterwards we disable the character controller by setting it to false.

We are now able to teleport the player, here we say to teleport whatever triggered the IF statement (so only the player) to the location of where we have set our respawn point to be. At the moment we have not set our respawn point we will do that once we are done here.

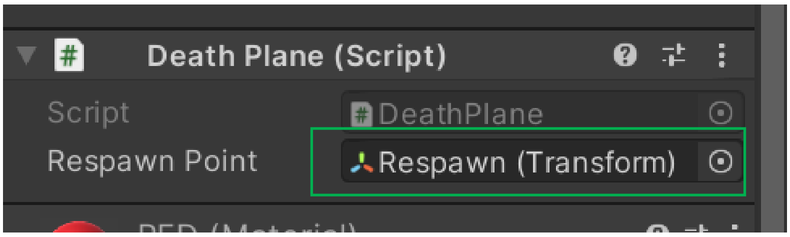
Finally, the controller is reenabled so that the player can move again and continue playing

Adding a respawn point

1. Save and exit out of the script
2. Create an empty gameobject or create your spawn point
3. Move the spawn point to where you want to respawn
4. Rename both your death plane and respawn point to appropriate names



5. In the inspector drag and drop the Respawn point into the script we put on the death plane



Death Plane (Script)

Script DeathPlane

Respawn Point Respawn (Transform)

RED (Material)

6. Place your character and test out the death plane