**Player is Reset Tutorial**

Open up a new Unity 3D scene. Once open, create a 3D plane and set the scale to 2 in X, Y and Z. Make sure the position is 0 for X, Y and Z. Rename this to “Ground”.

Now create a 3D Capsule and set the position for X and Z to 0 with Y at 1. Rename this to “Enemy”.

(Give the two 3D objects two different materials if desired).

(When a player model is added, make sure the player has ‘is trigger’ ticked in the inspector)

On the capsule collider for the enemy, make sure ‘is Trigger’ is ticked.

On the Enemy inspector, add a Rigidbody and under constraints; freeze the rotation on the Y axis for Position and Rotation.

Make sure the ‘Collision Detection’ is on Continuous.

Create a script in the assets folder called; “PlayerRespawn”. Attach the script to the Enemy object.

Open up the PlayerRespawn script in your script writing software and type out the following **bold** code (before writing anything, delete the ‘void Start’ and ‘void Update’ code):

{

**[SerializeField] private Transform Player;**

**[SerializeField] private Transform RespawnPoint;**

These determine the position of the player and respawn point and where the x, y and z axis of the respawn point where the player will return to.

After you have written out that code, type out the following **bold** code:

**Private void OnTriggerEnter(collider player)**

This is going to incase the code which means when the enemy and player collide the player will be sent back to the respawn point. The trigger has to be on enter so when the rigid bodies collide the script runs.

After you have written out that code, type out the following **bold** code:

Private void OnTriggerEnter(collider player)

**{**

**Player.transform.position = RespawnPoint.transform.position;**

**}**

What the code above does is when that trigger happens the players position will be transformed to the respawn points position.