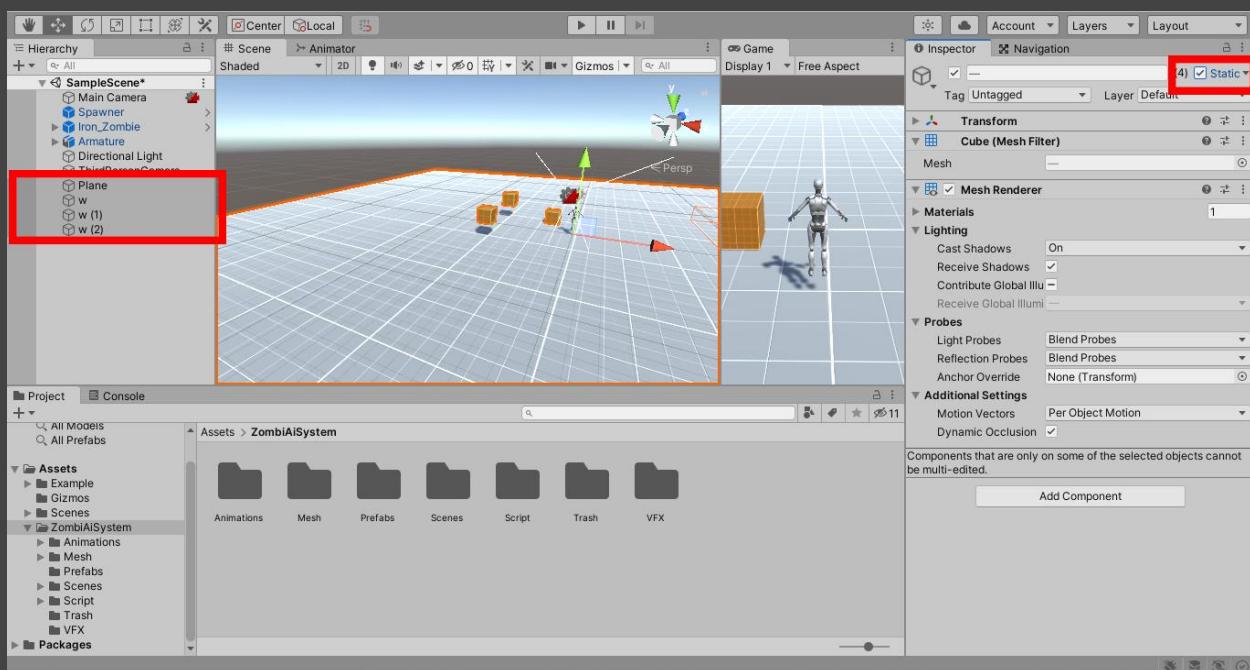


ZOMBIE AI SYSTEM

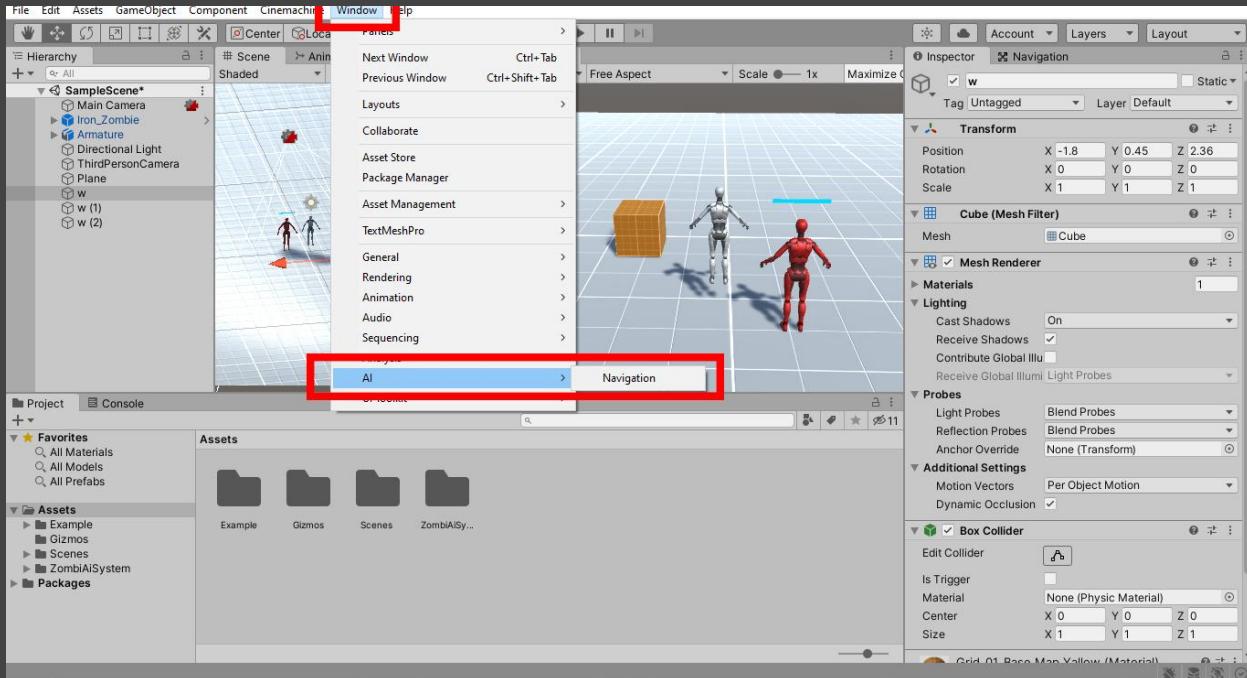
Click link below to watch the full step-by-step tutorial

https://www.youtube.com/watch?v=iTsBn99JIOo&ab_channel=RemeshGames

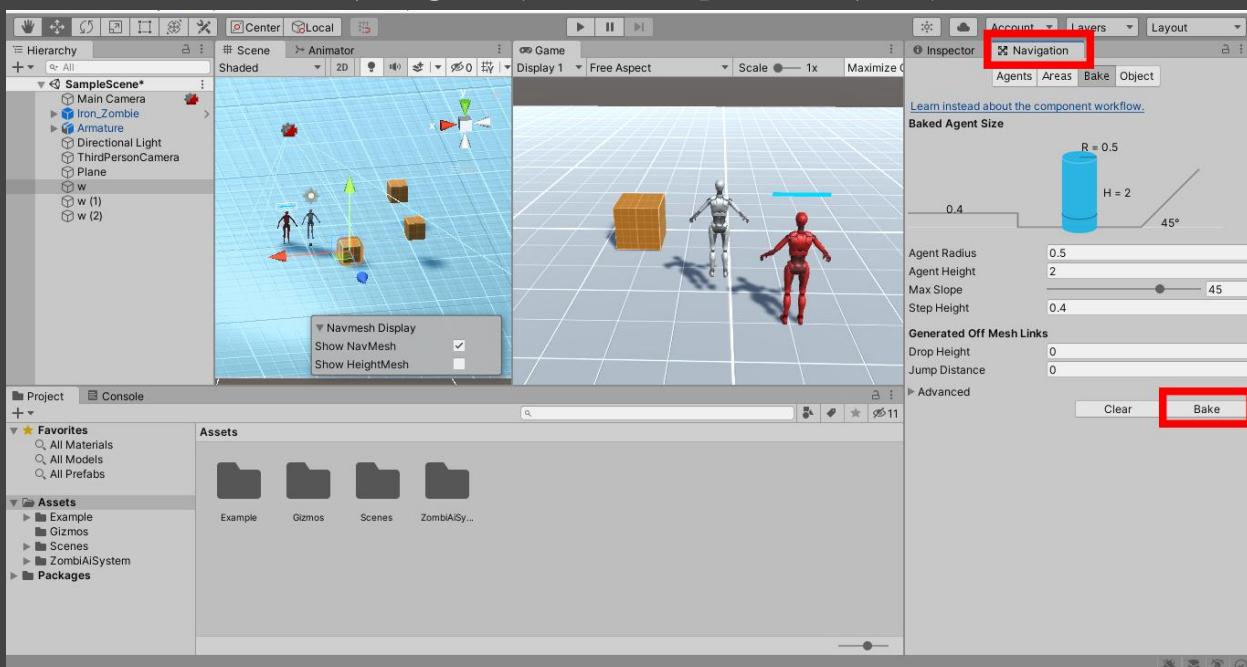
Select all the objects in the level and set them to (Static)



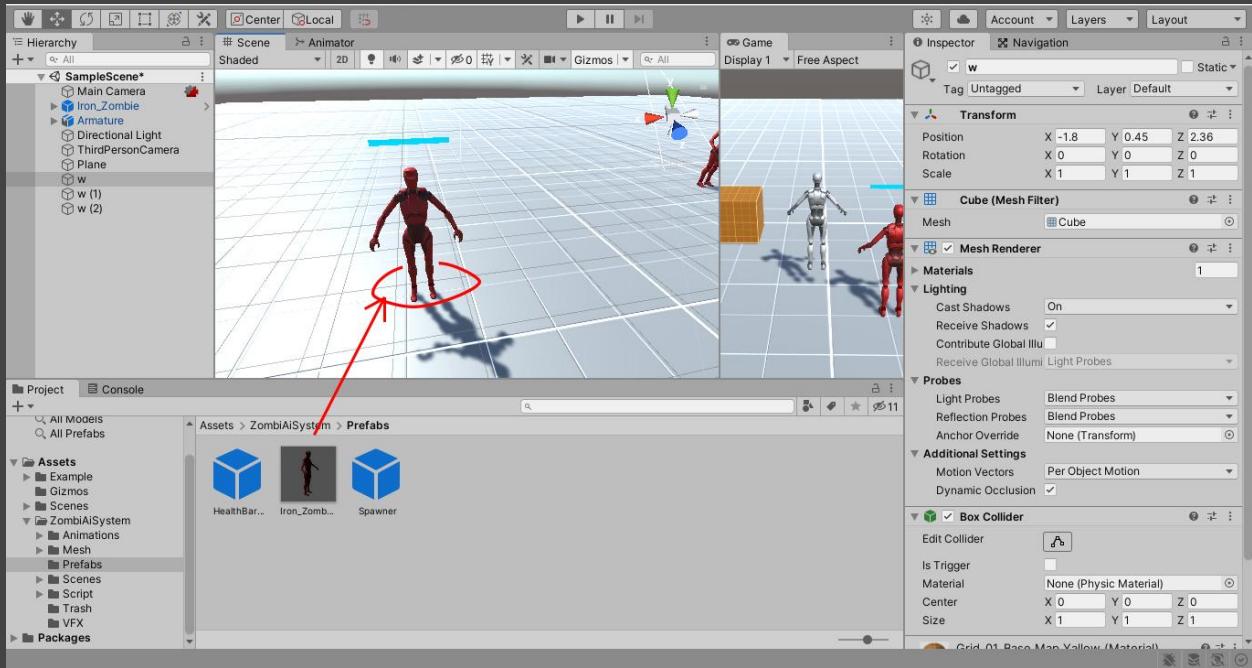
Go to (window) and Choose (Navigation) from (AI)



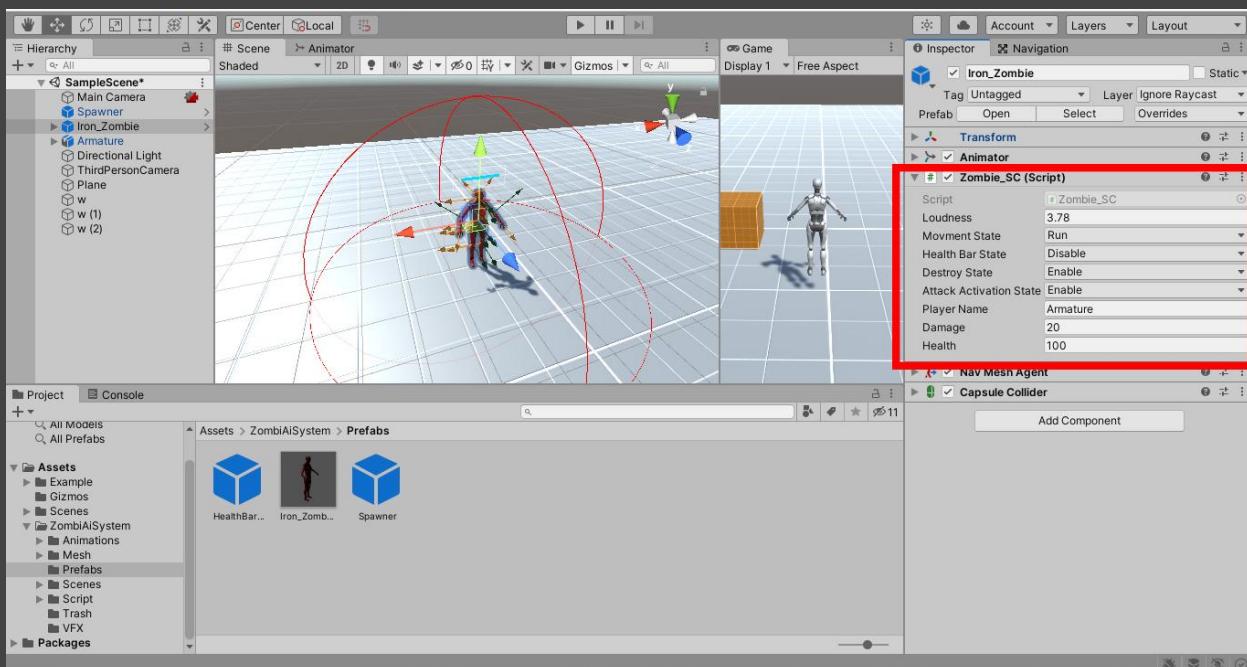
Go to the (Navigation) menu and press the (Bake) button



Go to **prefabs folder then Drag and drop the zombie in the level**

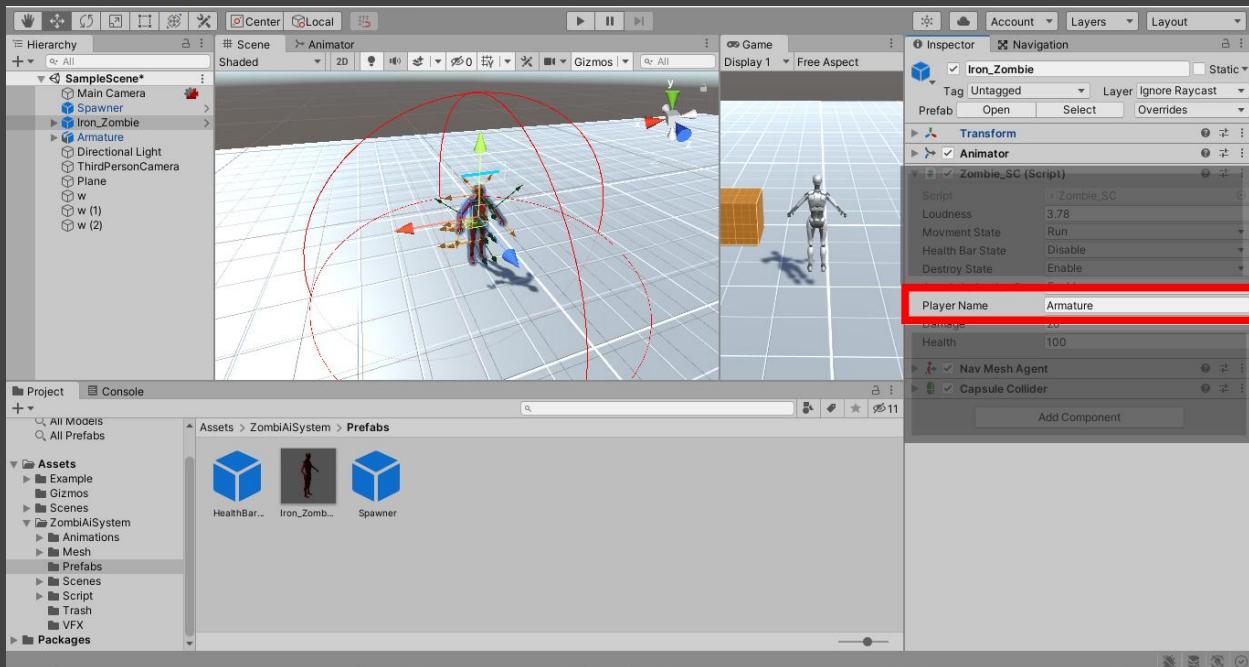


Zombie Properties
Loudness
Movement State [Walk, Run]
HealthBar State [Enable, Disable]
Destroy State [Enable, Disable]
Attack Activation State [Enable, Disable]
Player name, Damage and Health



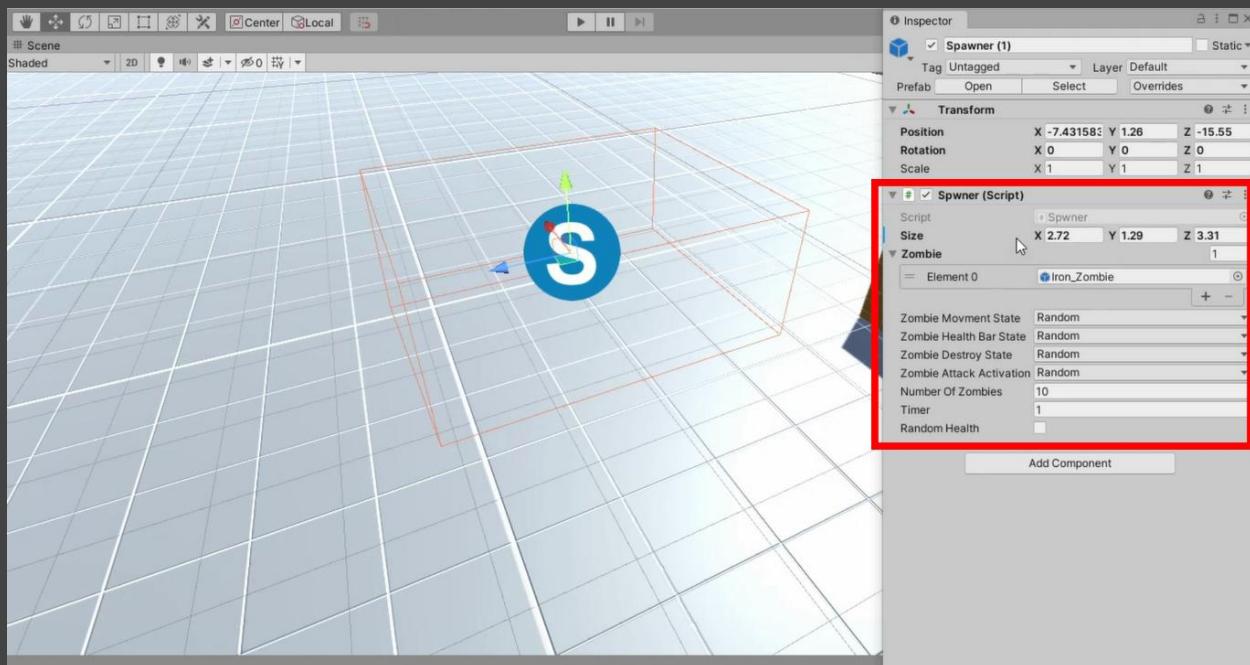
Player Name

Important Note: You have to Copy the name of your own Player Character and paste it in the **(Player Name)** in order to make the zombie recognize the player character.



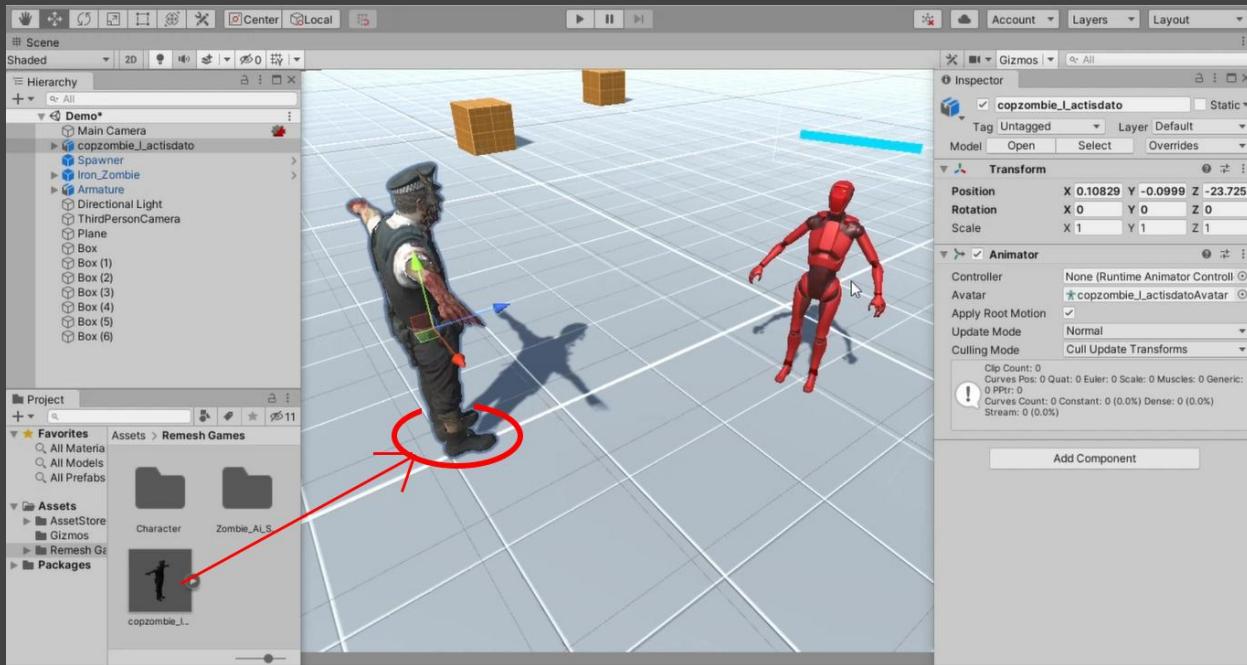
Spawner Properties

Box Extent
Zombie List
Zombie Movement State [Walk, Run]
Zombie HealthBar State [Enable, Disable]
Zombie Destroy State [Enable, Disable]
Zombie Attack Activation State [Enable, Disable]
Number of Zombie,
Timer and Random Health

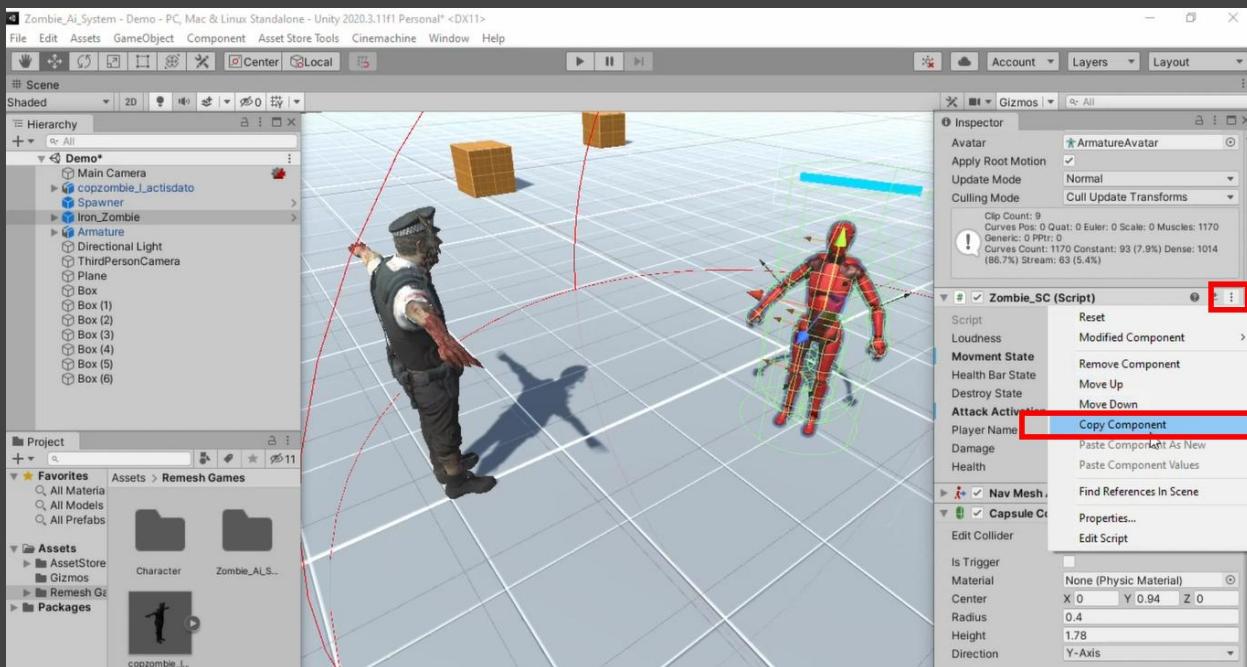


CHANGE ZOMBIE CHARACTER

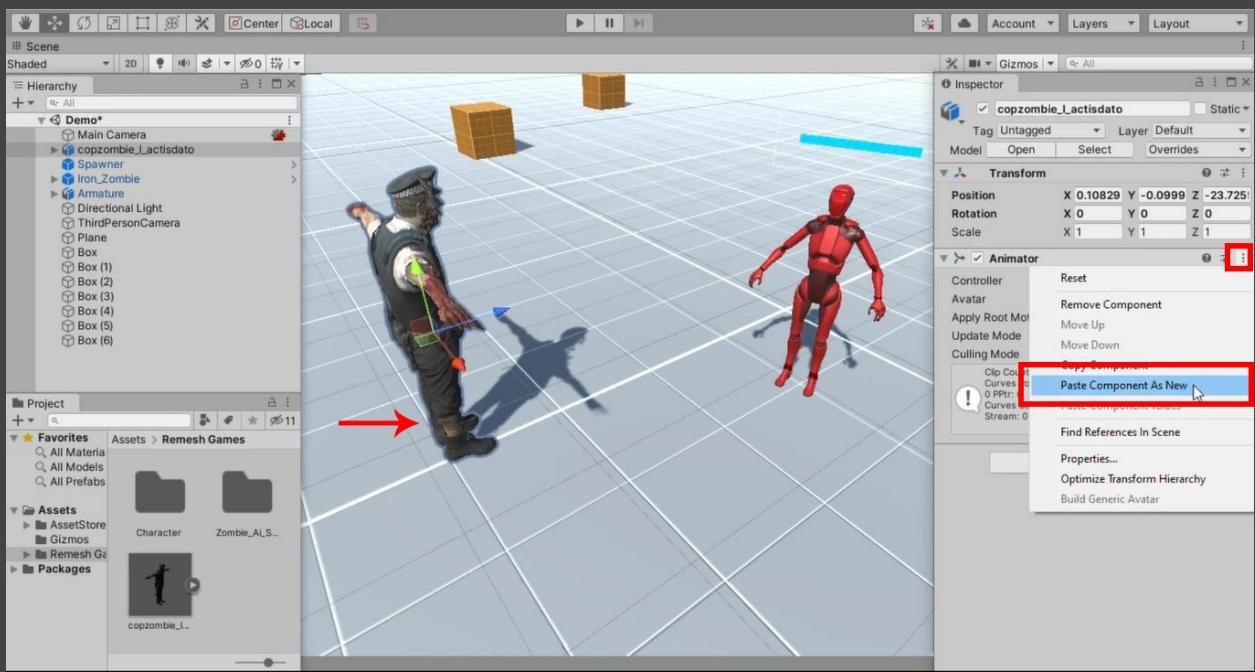
Drag your character to the level



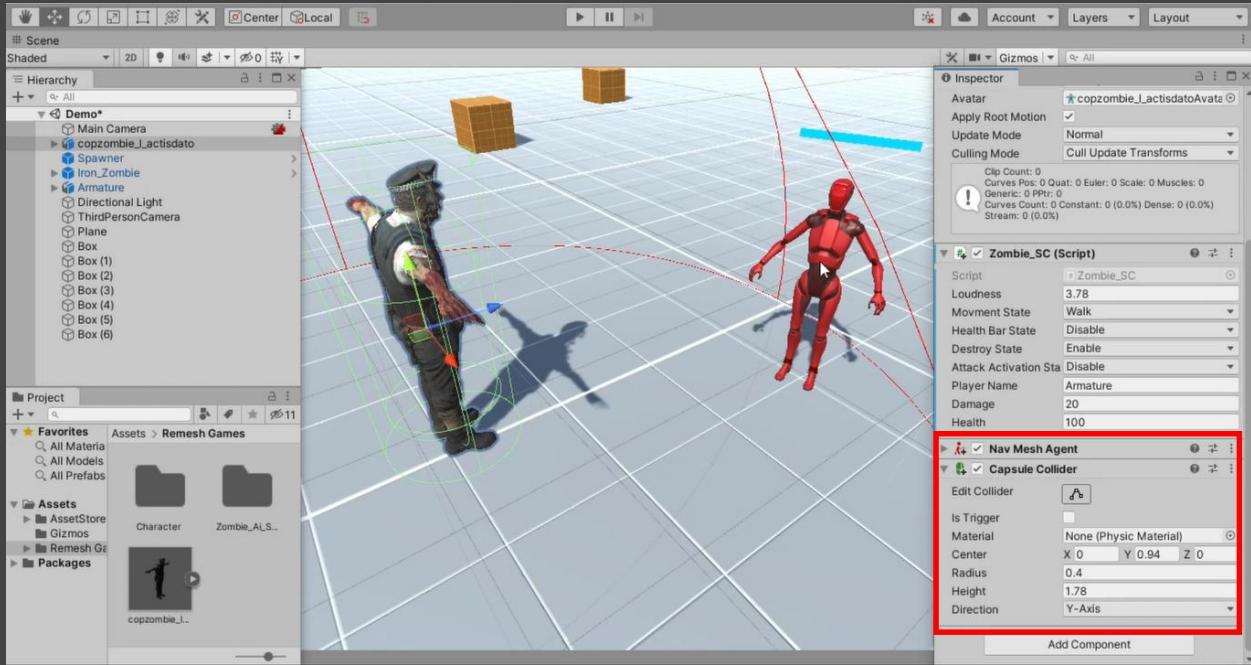
Select (Iron Zombie) character then Click (Copy Component) on the script



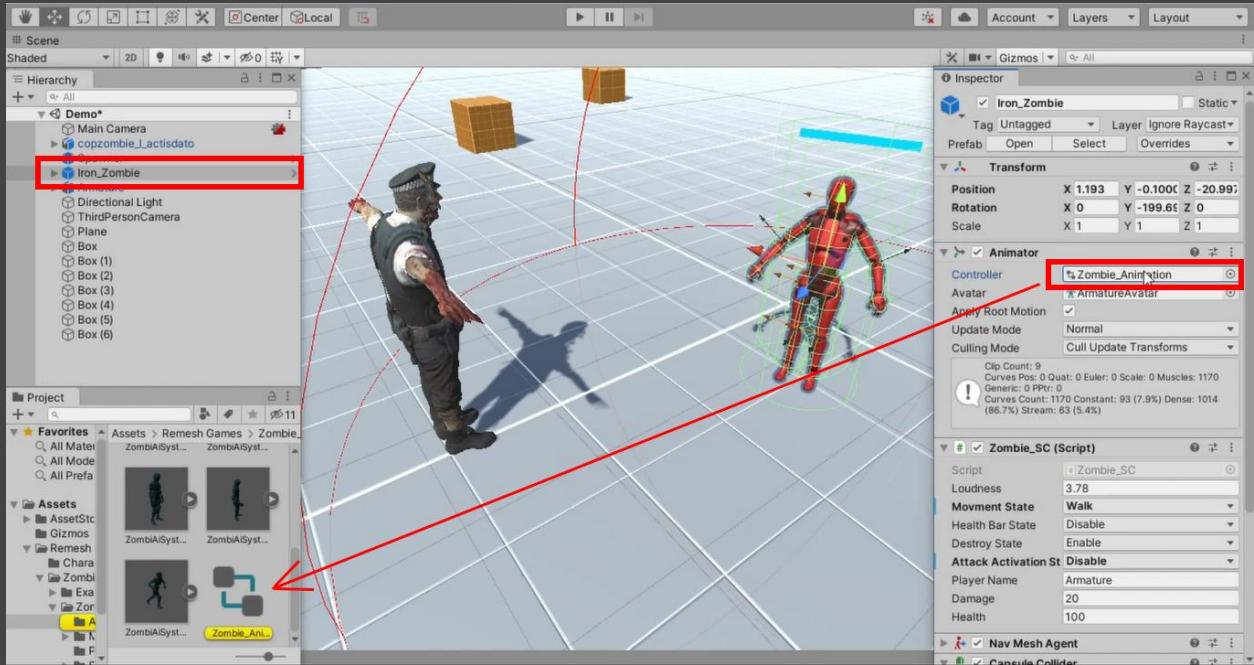
Then select your zombie character and (paste component as new)



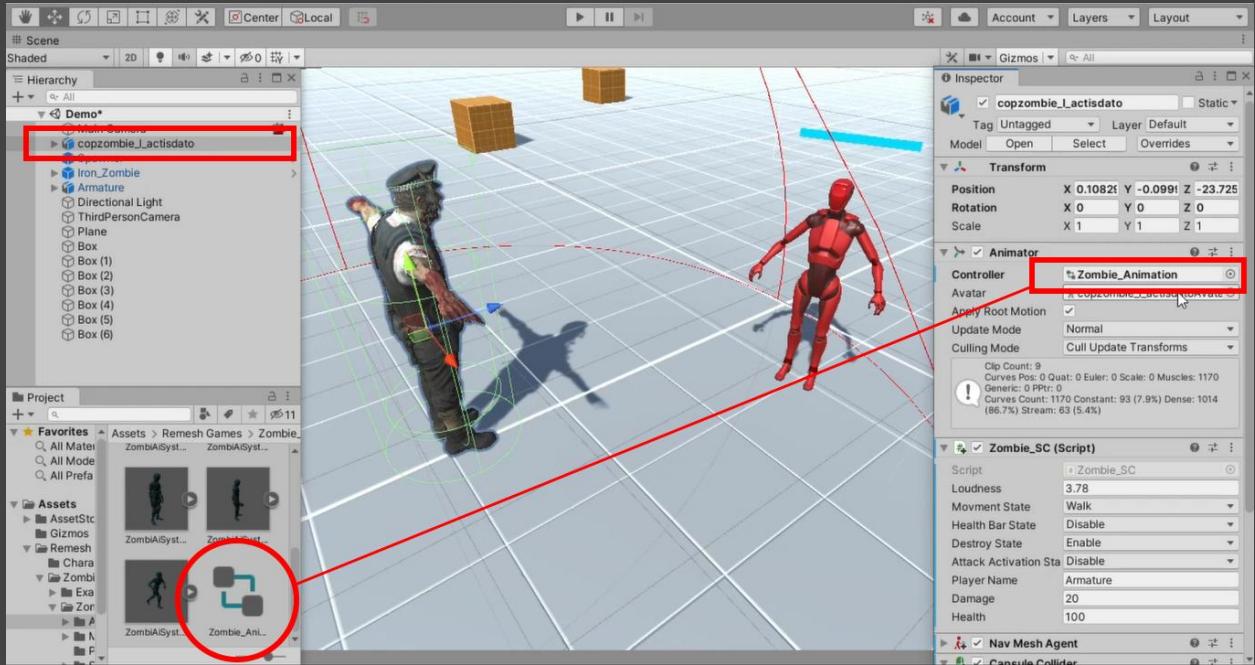
Again select (Iron Zombie) character then Copy all the components of the (Iron Zombie), then select your zombie character and paste the components on your zombie character



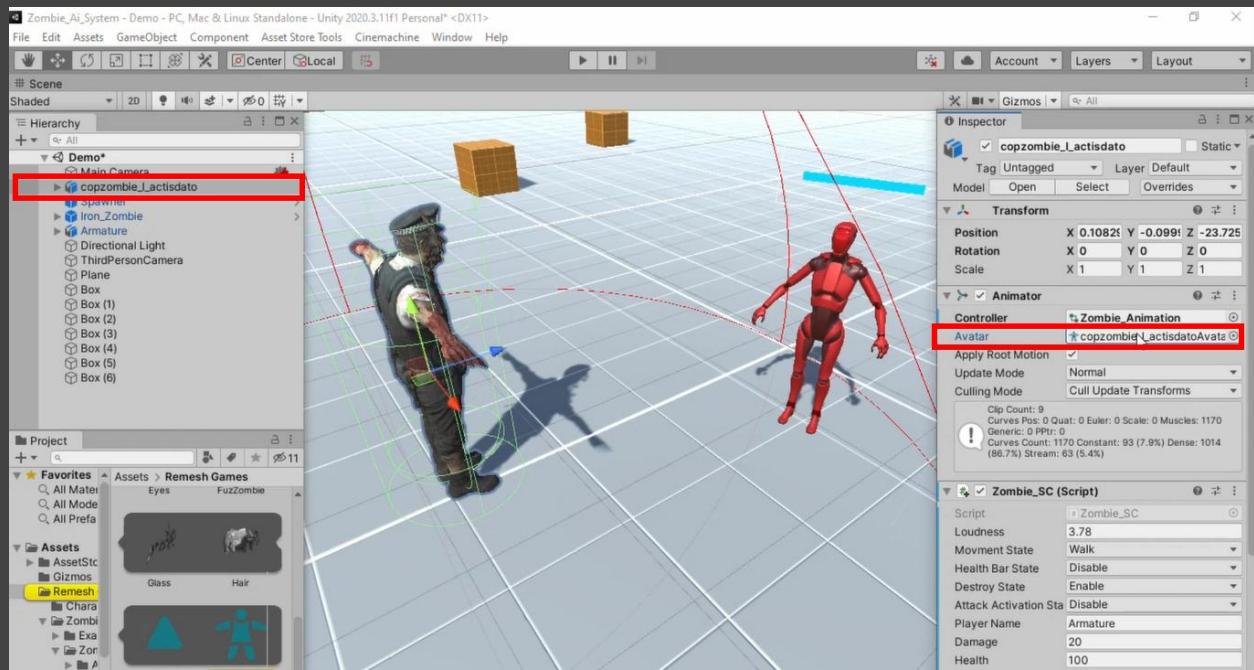
In order to go to (Zombie animation) folder select the (Iron Zombie) character, then from (Animator) select (Zombie animation) from the controller slot



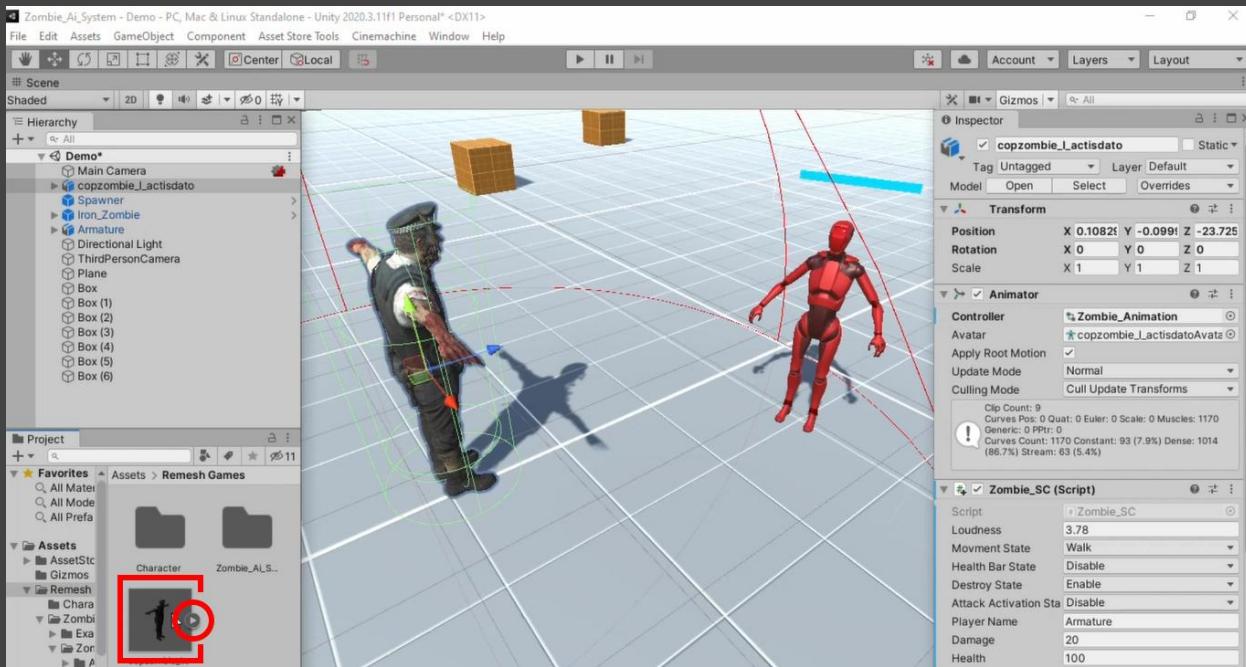
After that select your zombie character and give it the (Zombie animation)



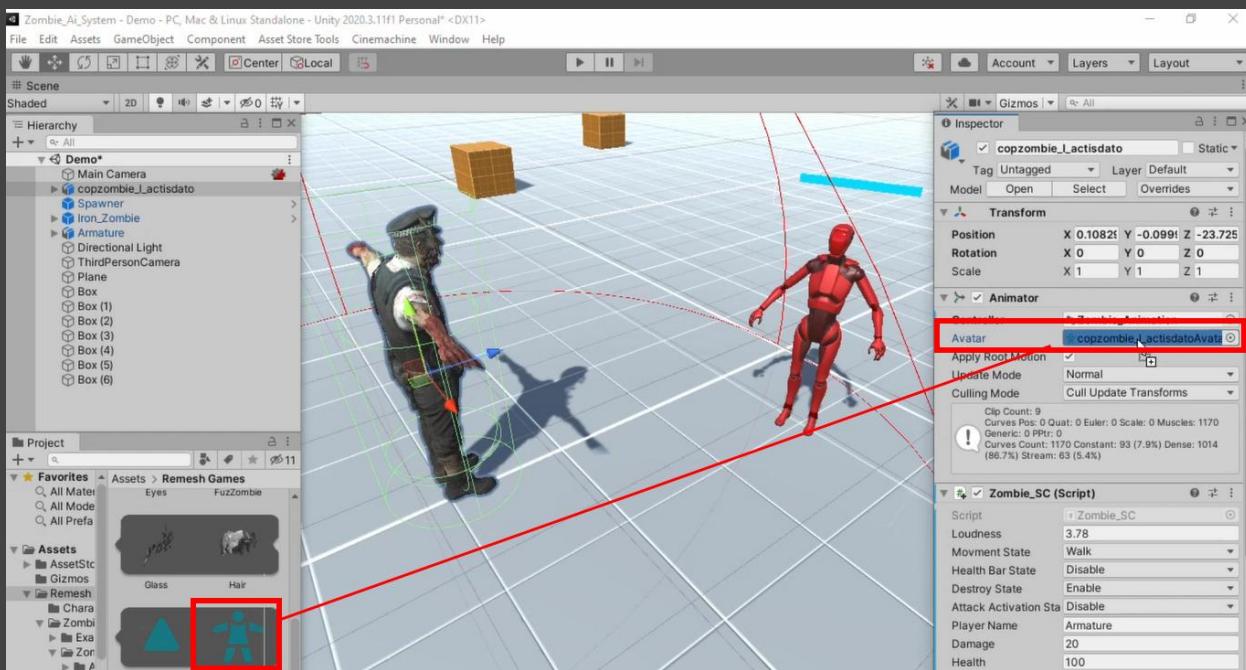
Make sure your character's avatar is valid if the avatar is not valid the animation will not function properly



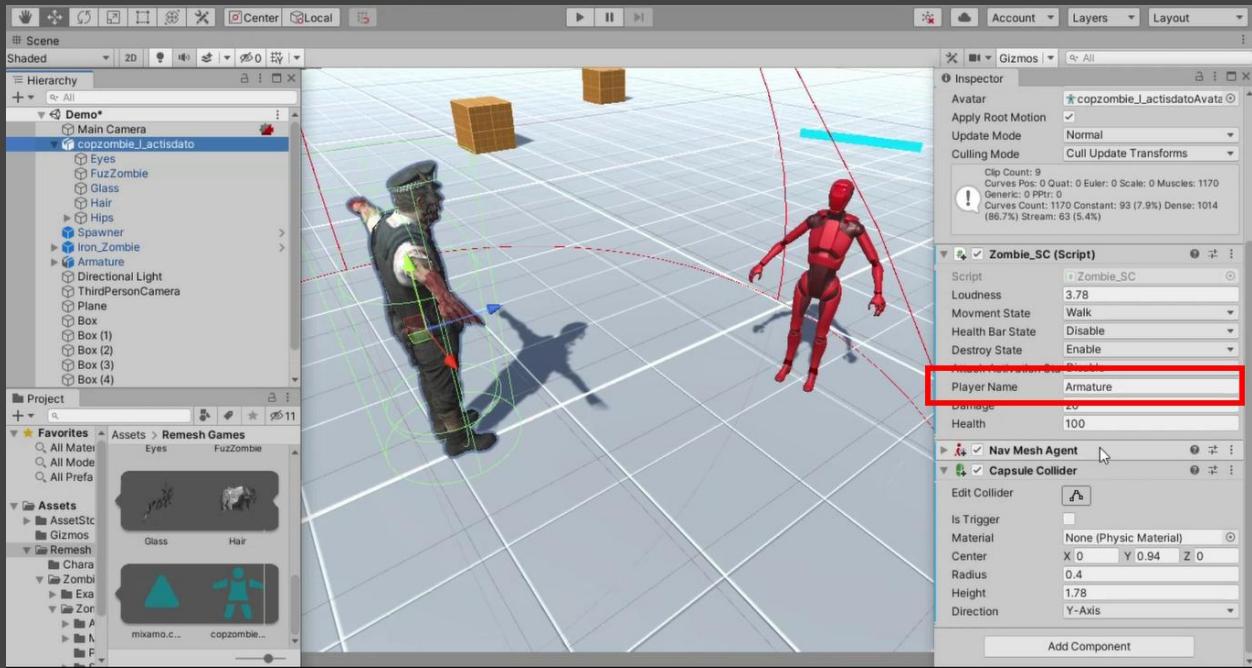
In order to find your character's avatar go to your character's model, click on the small arrow to display everything within the model



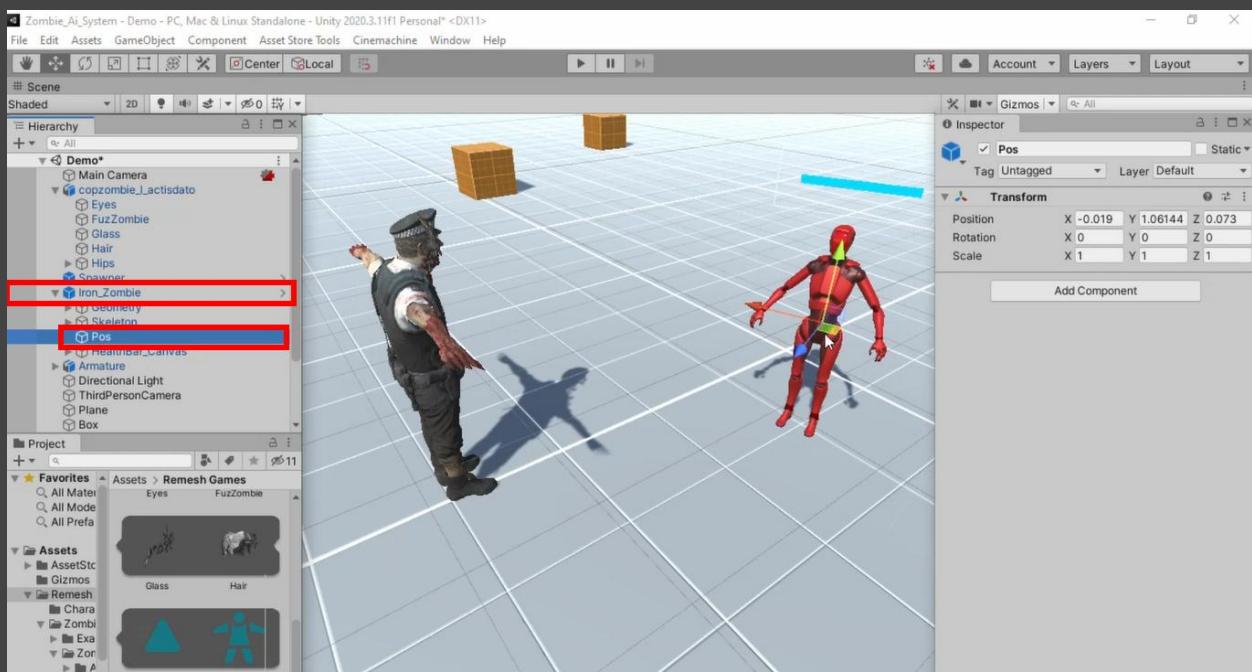
Then drag and drop it on the avatar slot



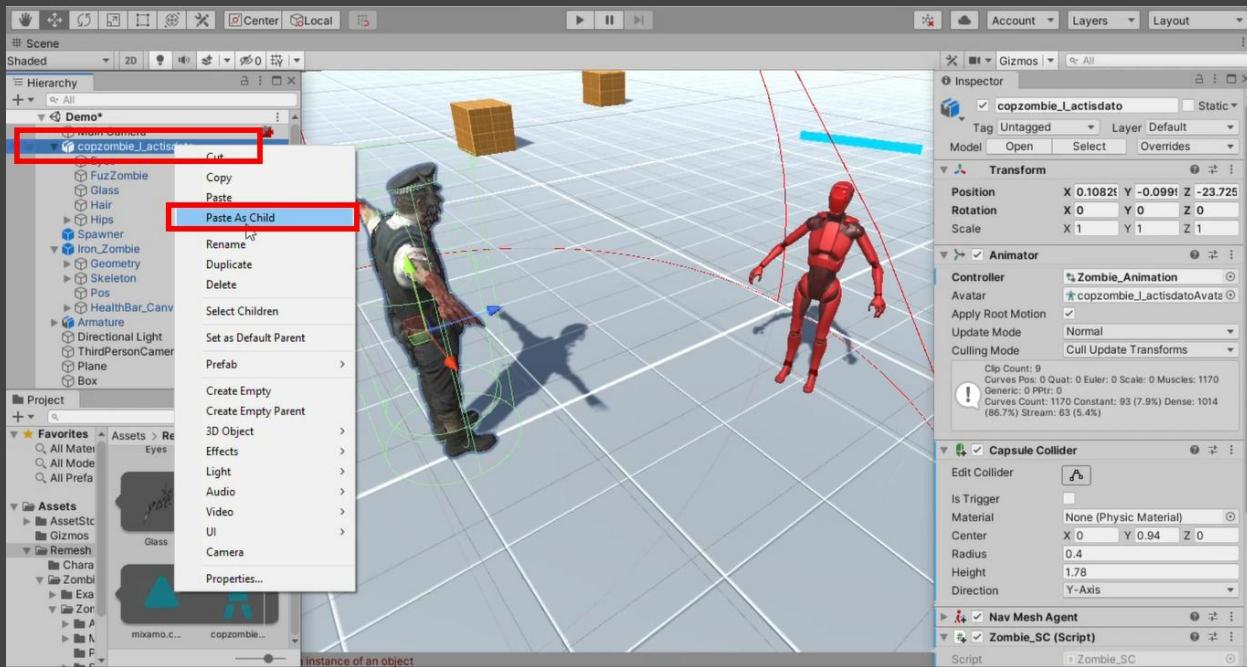
Also, Make sure the player name is right



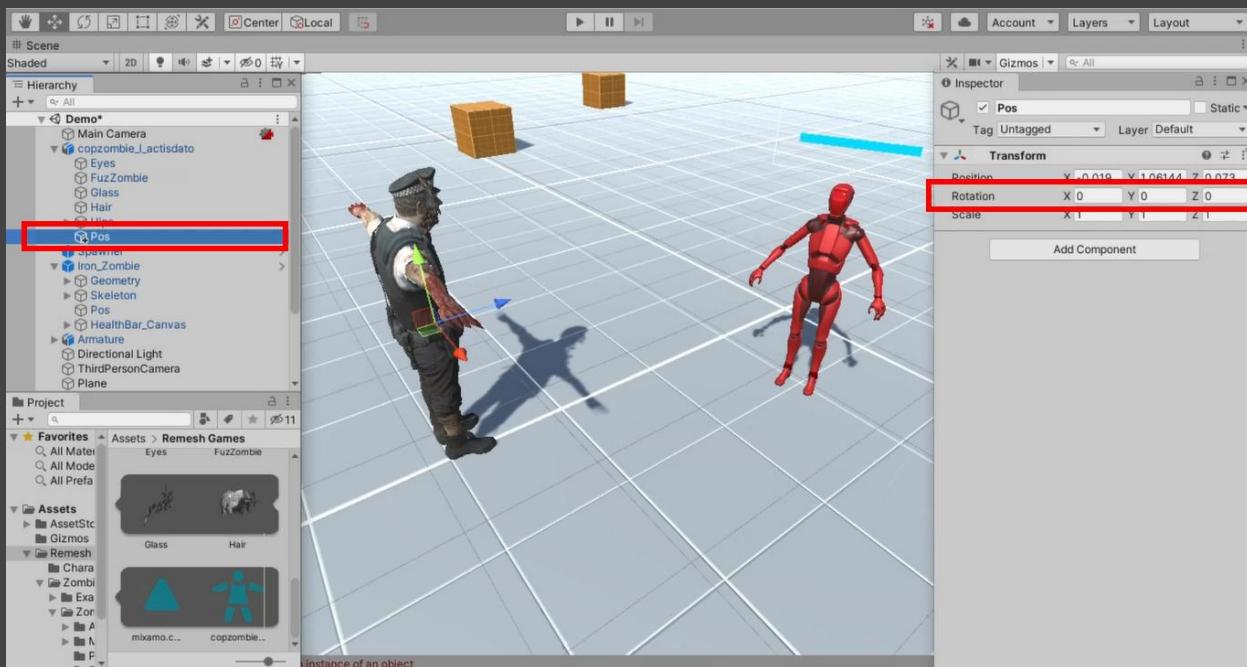
From Hierarchy go to (Iron Zombie) character, click on the small arrow and copy (Pos)



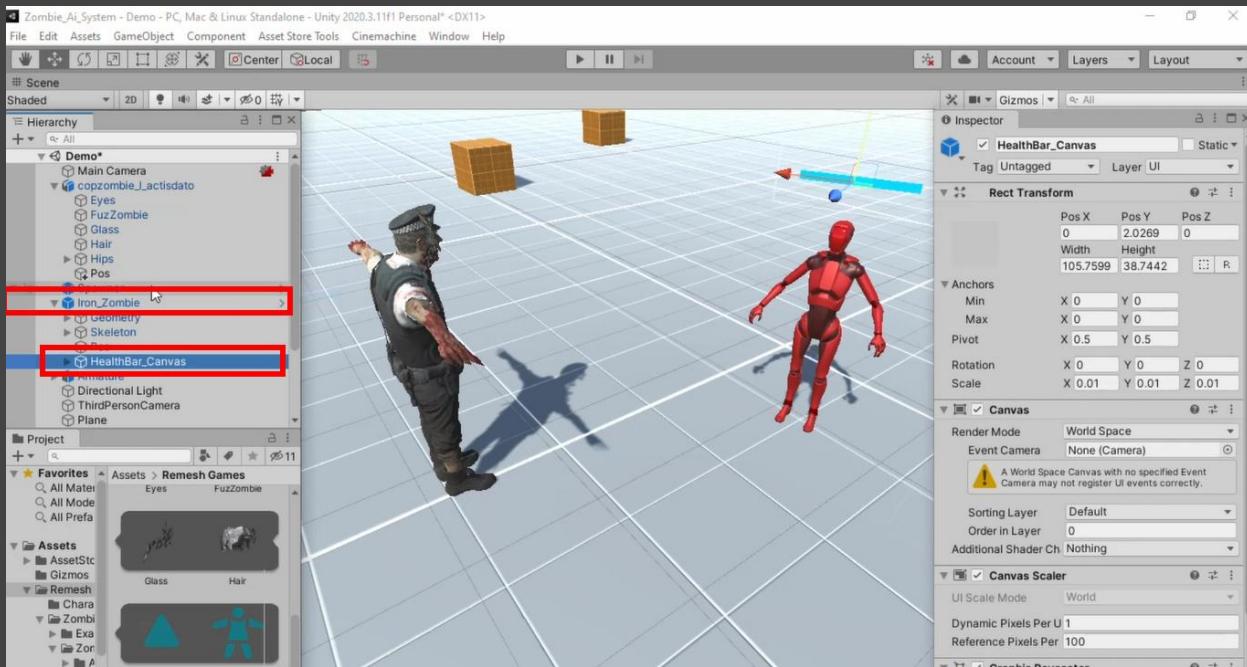
Then go to your character and (Paste As Child)



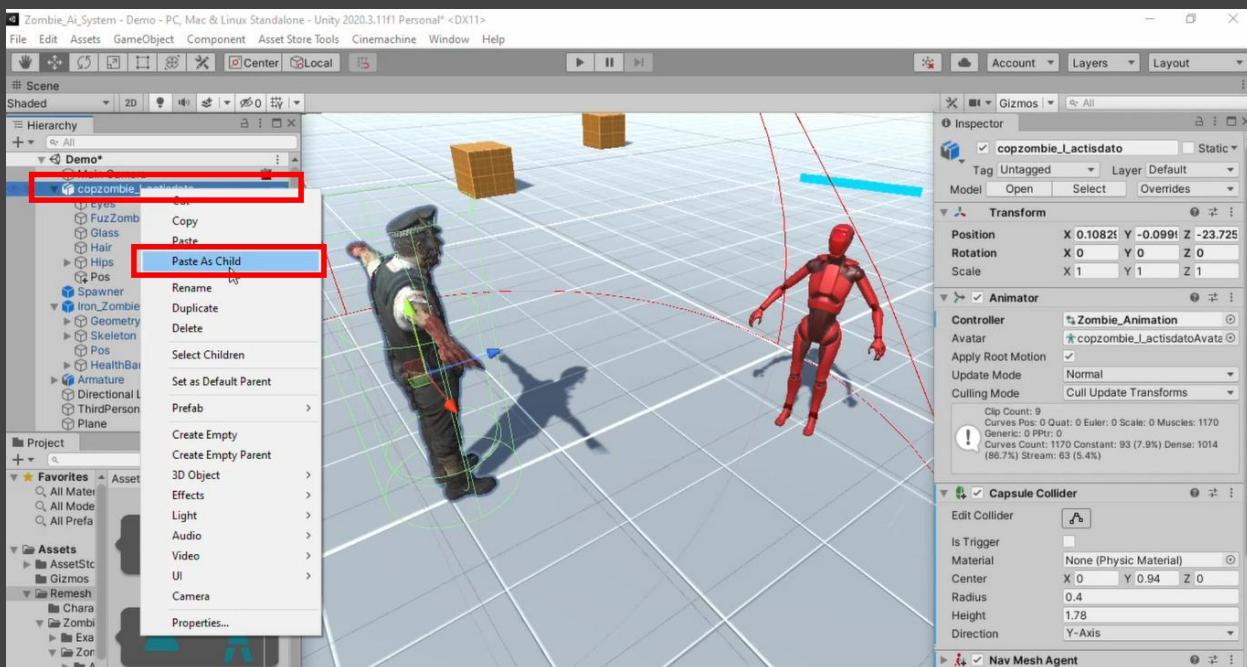
Make sure (Pos) Rotation is equal to zero



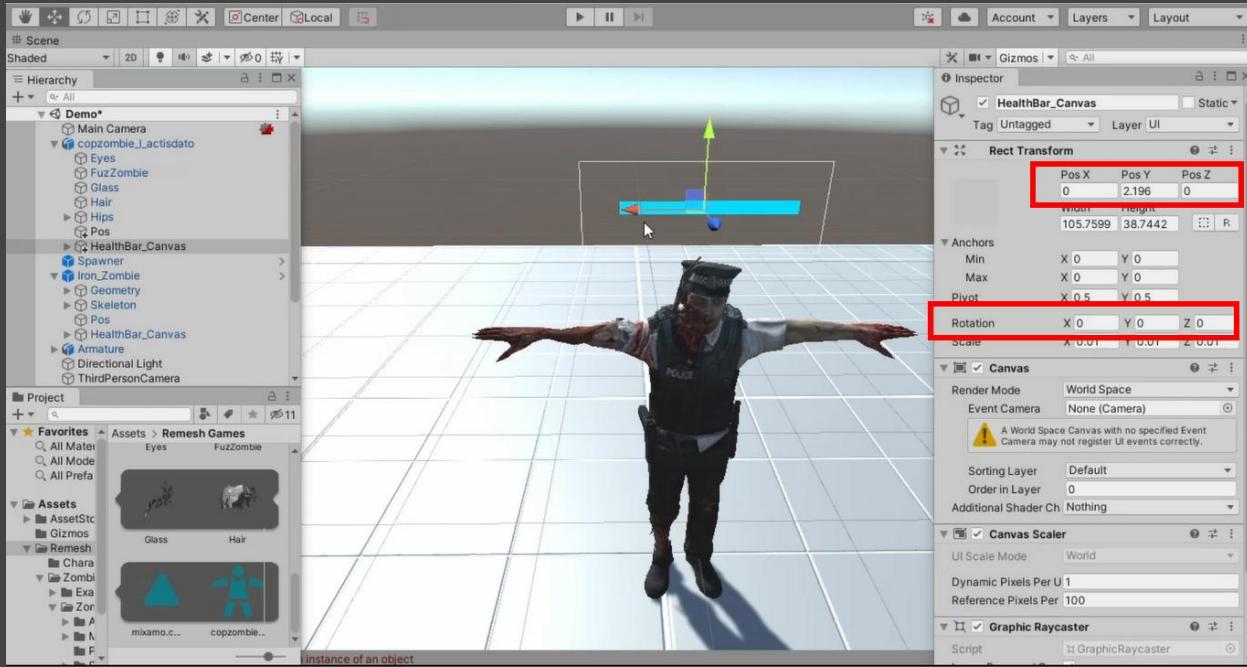
Go to (Iron Zombie) character and copy (HealthBar_Canvases)



Then go to your character and (Paste As Child)

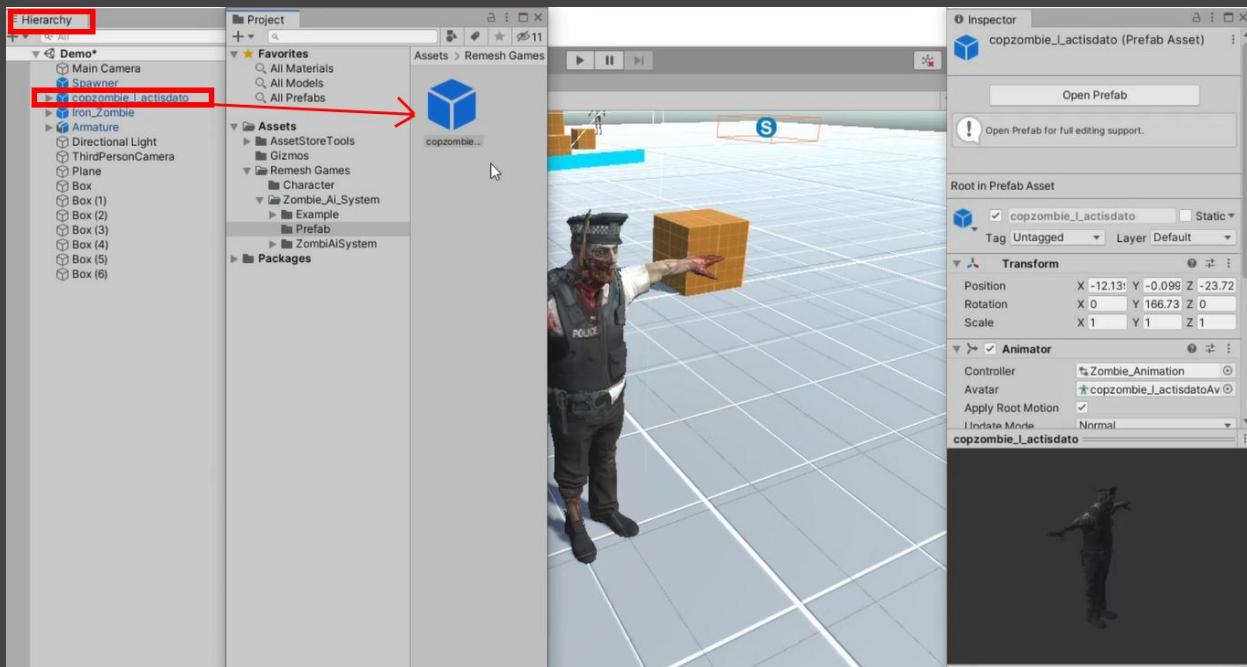


Make sure (HealthBar_Canvas) Rotation is equal to zero and its position is set in the right place

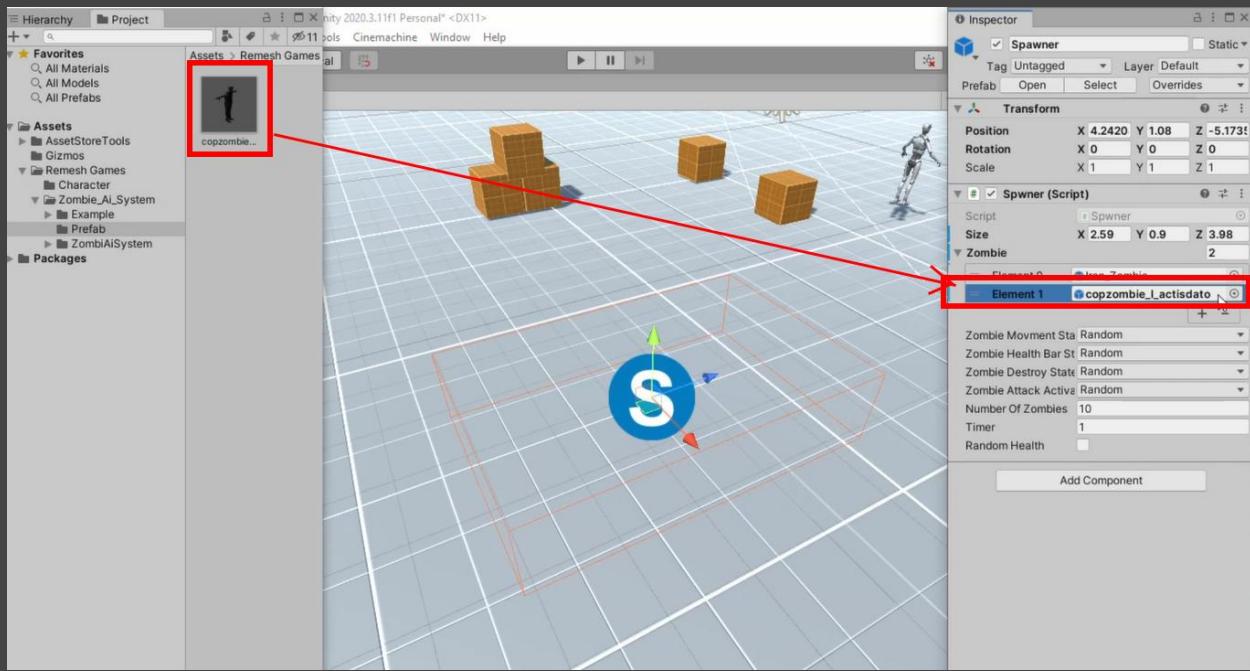


SET YOUR CHARACTER AS PREFABS

To create your zombie character as a (prefab) just drag your character from the hierarchy and drop it into your prefab folder or whatever folder you want

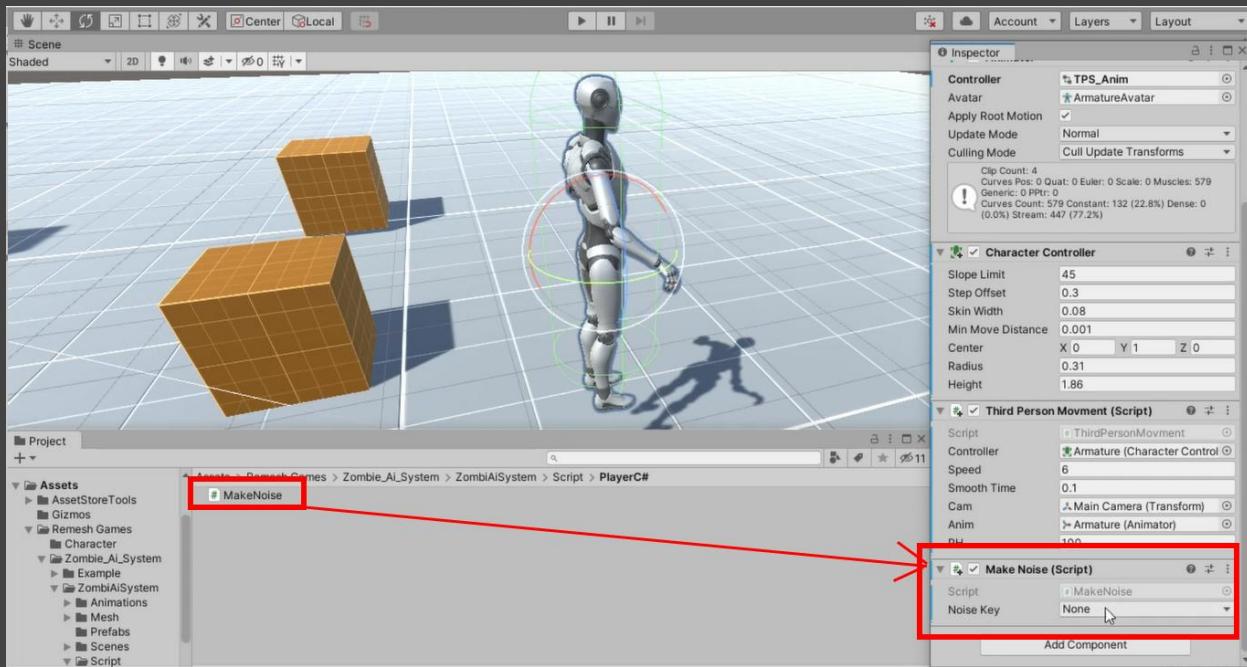


Now you can add your character prefab to (Zombie List) in the spawner

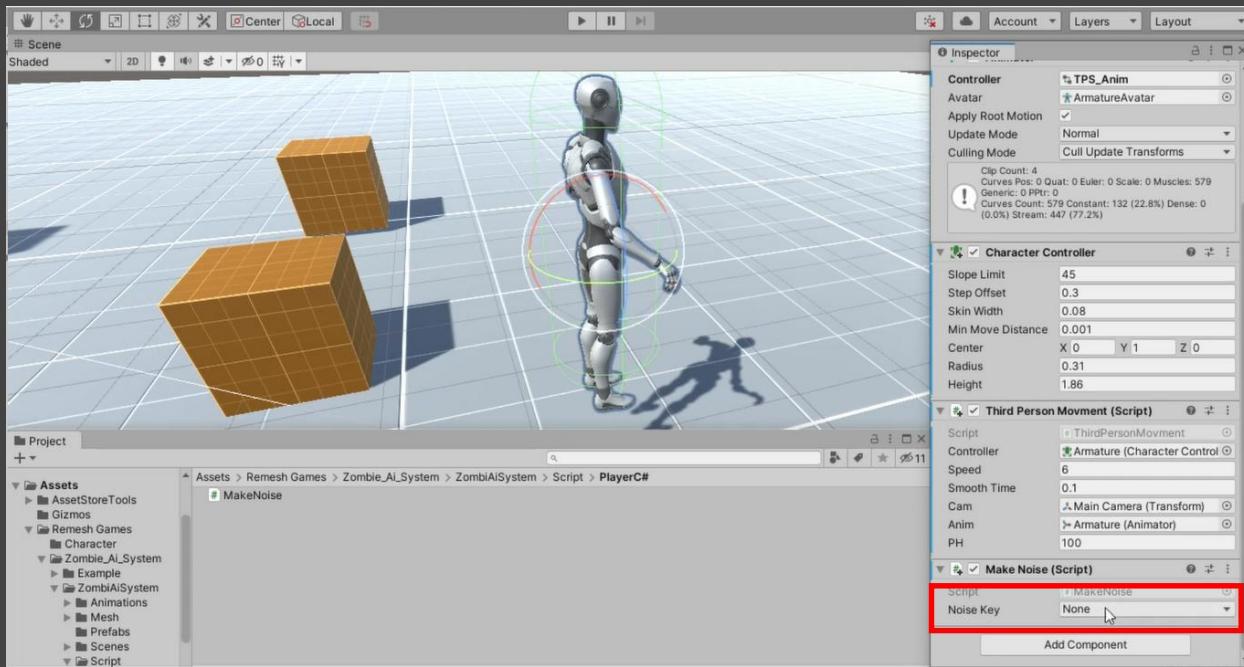


AI HEARING USE (MAKE NOISE) SCRIPT

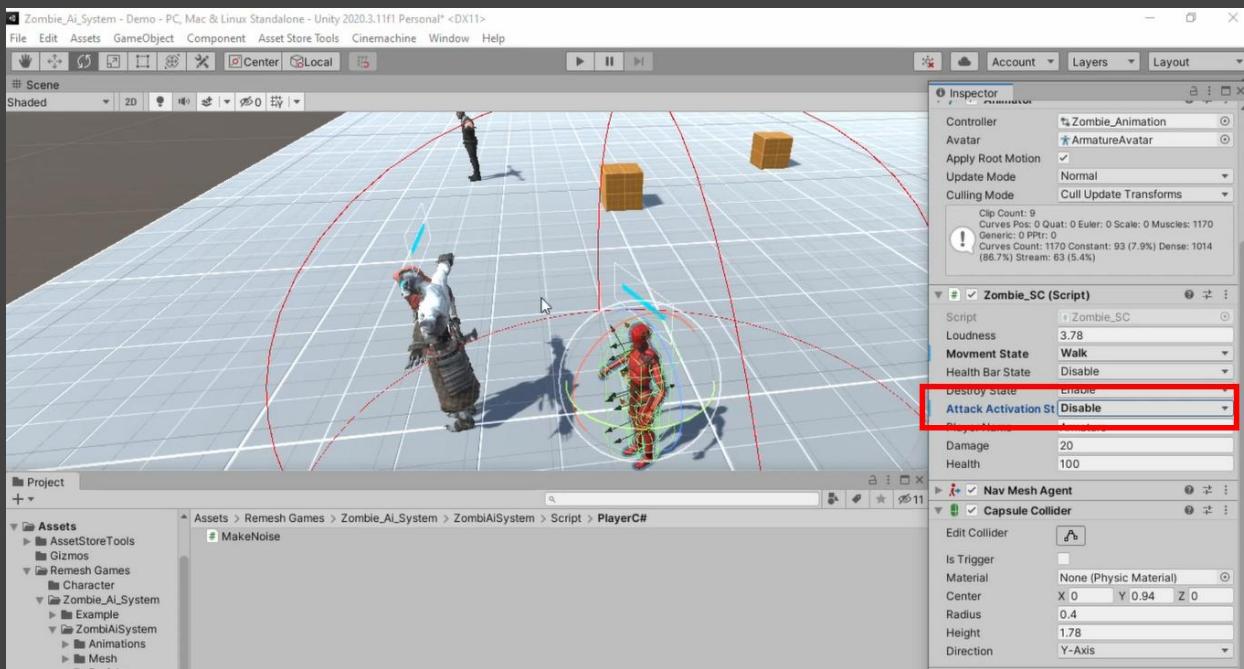
Add (Make Noise) script to your player character



And select the key you want



Make sure the (Attack Activation State) feature is Disabled so the zombies will not attack immediately rather they will rely on their hearing



Getting Damage

USE (APPLY DAMAGE)

Copy (Apply damage) function from the document



Here is the code

```
public void ApplyDamage(float Damage)
{
    PlayerHealth -= Damage;
}
```

And paste it in your player health script then replace the (Player Health) with your player variable health

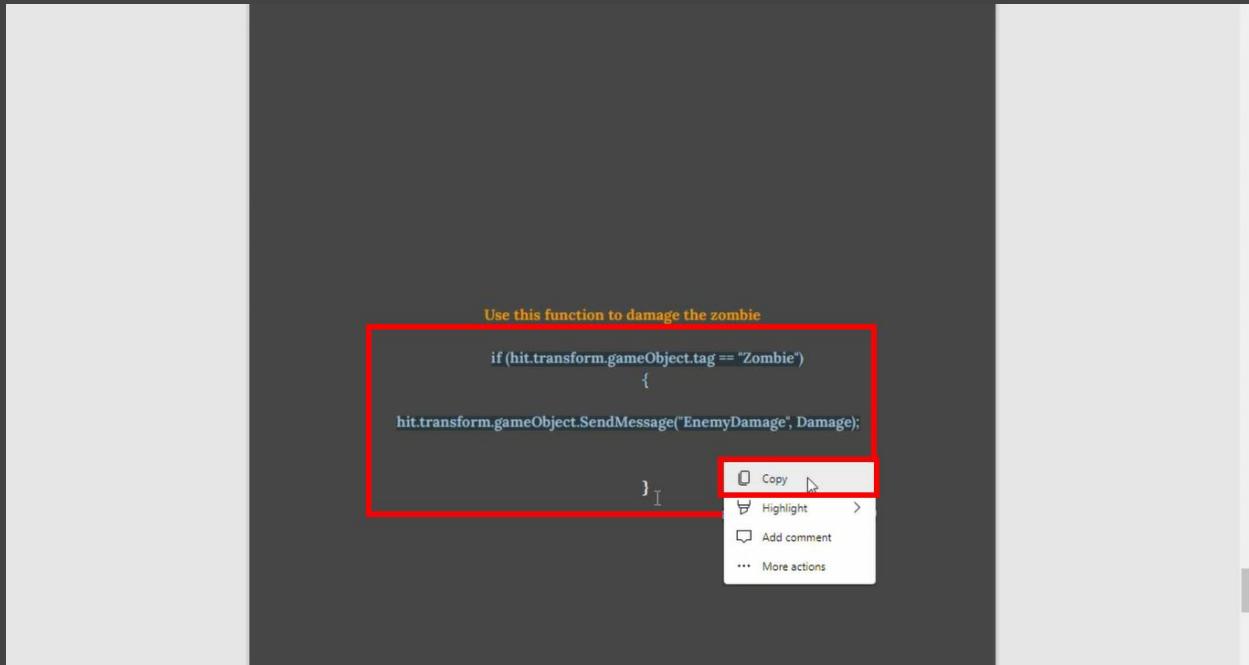
The screenshot shows the Unity Editor's code editor with the file `PlayerHealth_CS.cs` open. The code defines a class `PlayerHealth_CS` that inherits from `MonoBehaviour`. It has a public float variable `Health` initialized to 100.0f. The `Start()` method is empty. The `Update()` method is also empty. A method `applyDamage(float Damage)` is defined, which subtracts the damage from the `Health` variable. The `Health` variable and the `applyDamage` method are both highlighted with red boxes.

```
4
5     public class PlayerHealth_CS : MonoBehaviour
6     {
7
8         public float Health = 100.0f;
9
10        // Start is called before the first frame update
11        void Start()
12        {
13        }
14
15        // Update is called once per frame
16        void Update()
17        {
18        }
19
20
21
22
23        0 references
24        public void applyDamage(float Damage)
25        {
26            Health -= Damage;
27        }
28
29
30
```

Player Attack

USE (Send Message)

Copy this function from the document and paste it into your bullet script, this function is Used to damage the zombie when the player shoots at it



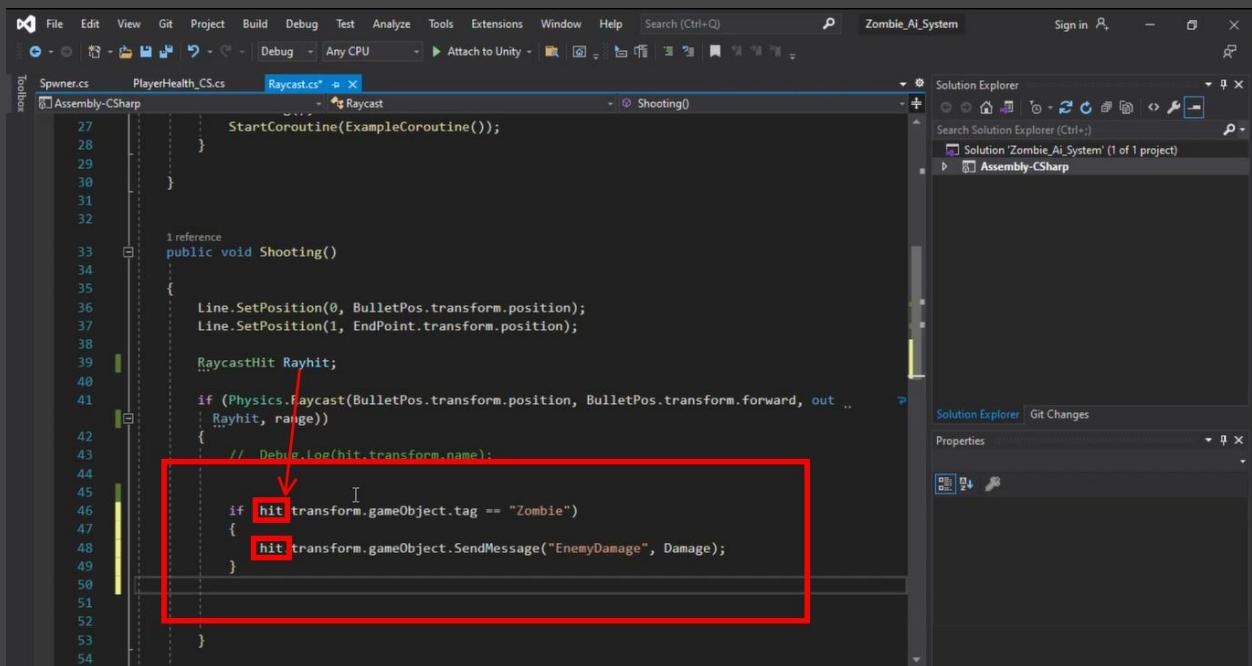
Here is the code

```
if (hit.transform.gameObject.tag == "Zombie")
{
    hit.transform.gameObject.SendMessage("EnemyDamage", Damage);
```



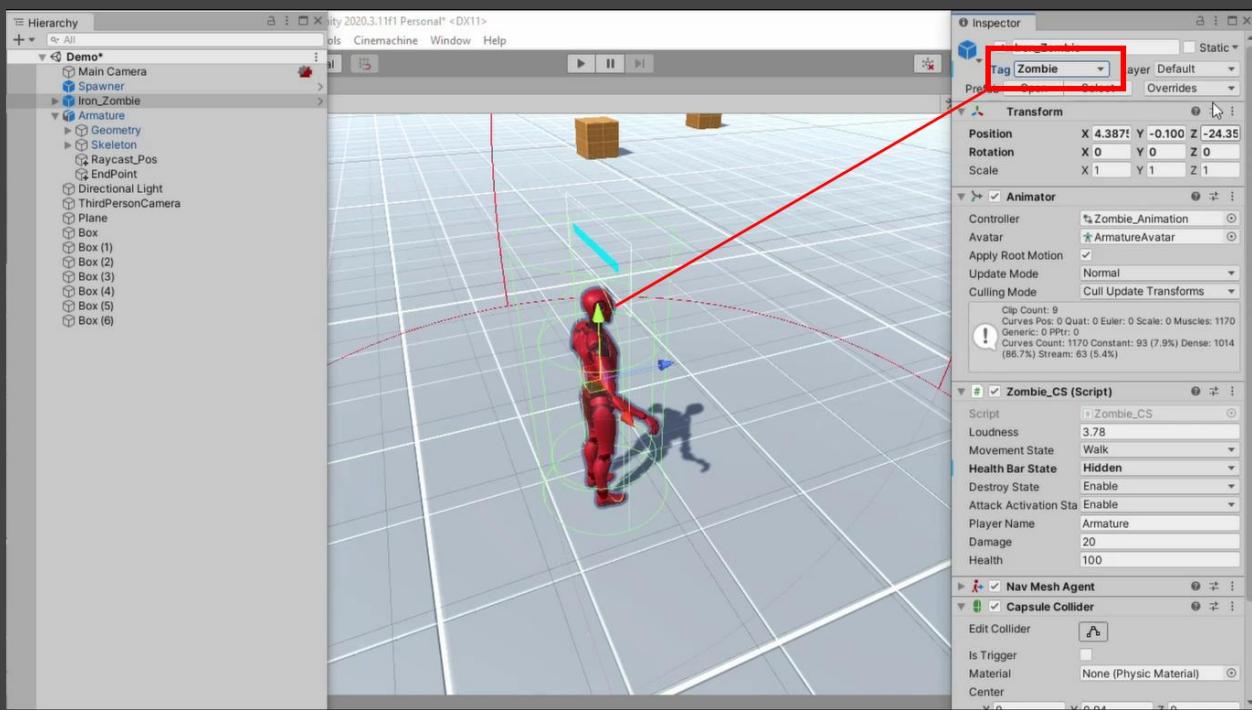
```
}
```

Paste the code here and make sure to replace the **(RaycastHit)** variable with your own variable for instance we replaced the **(hit)** with **(RayHit)**

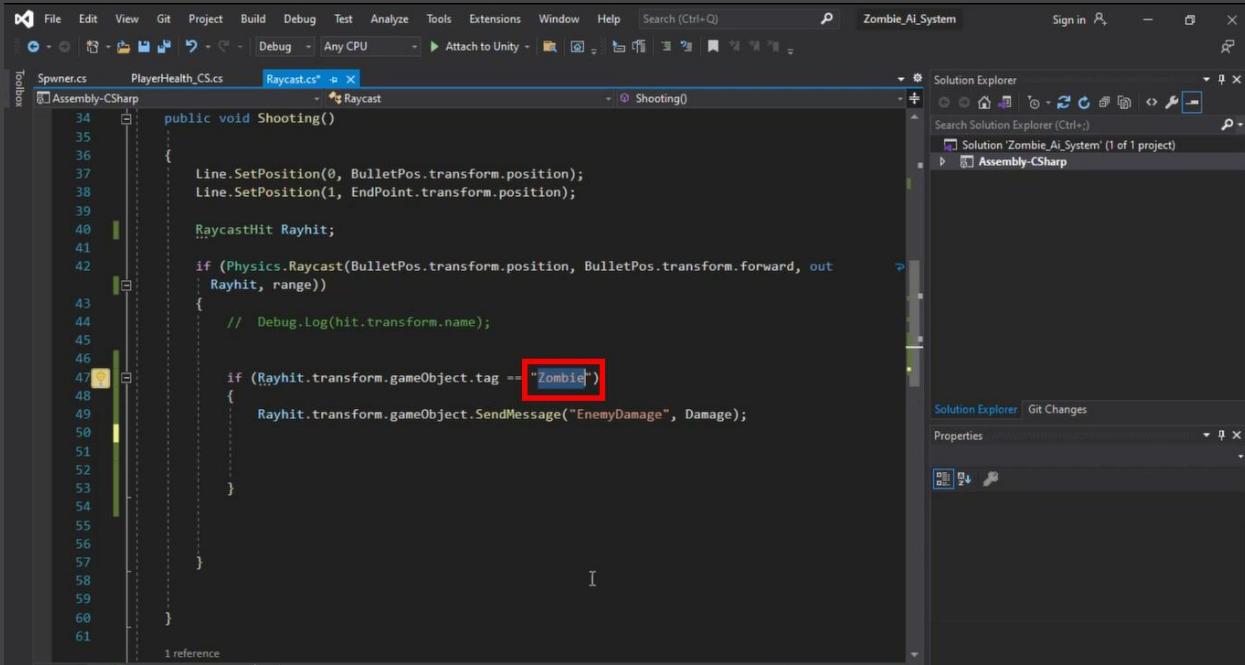


```
27     StartCoroutine(ExampleCoroutine());
28 
29 }
30 
31 }
32 
33 public void Shooting()
34 {
35     Line.SetPosition(0, BulletPos.transform.position);
36     Line.SetPosition(1, EndPoint.transform.position);
37 
38     RaycastHit Rayhit;
39 
40     if (Physics.Raycast(BulletPos.transform.position, BulletPos.transform.forward, out Rayhit, range))
41     {
42         // Debug.Log(hit.transform.name);
43 
44         if (hit.transform.gameObject.tag == "Zombie")
45         {
46             hit.transform.gameObject.SendMessage("EnemyDamage", Damage);
47         }
48 
49     }
50 
51 }
52 
53 }
54 }
```

Make sure your zombie character has the (Zombie) tag if you don't have the (Zombie) tag you will need to create one



Just copy the (Zombie) text from the script



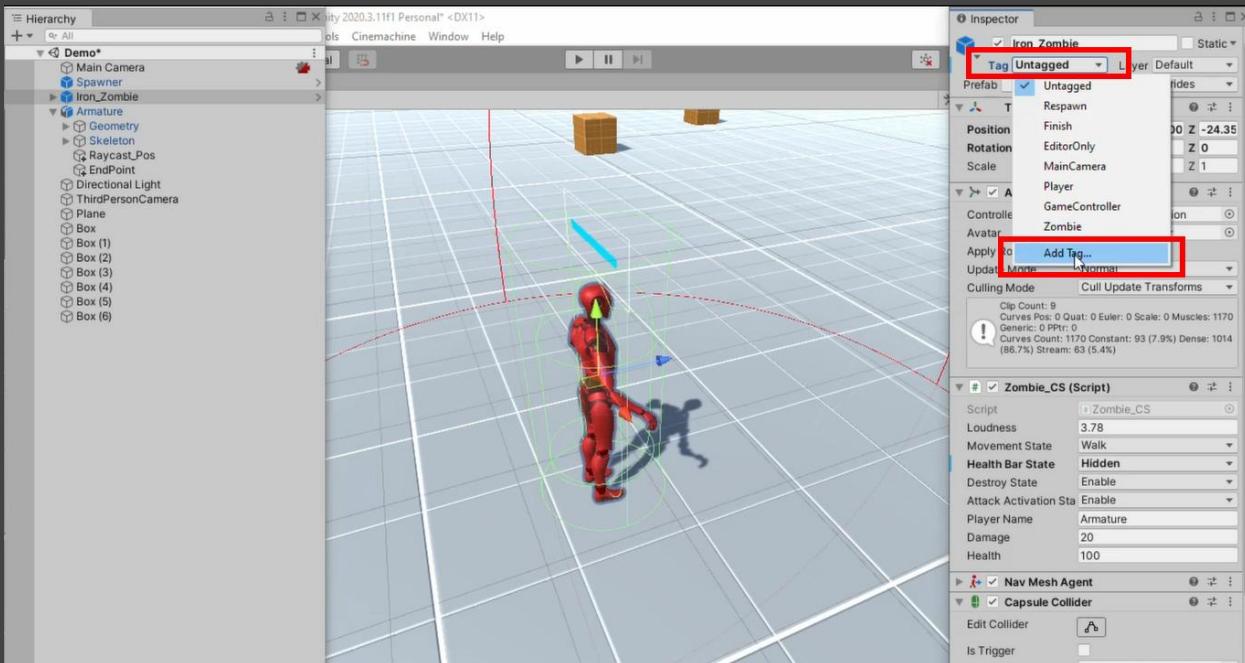
```
public void Shooting()
{
    Line.SetPosition(0, BulletPos.transform.position);
    Line.SetPosition(1, EndPoint.transform.position);

    RaycastHit Rayhit;

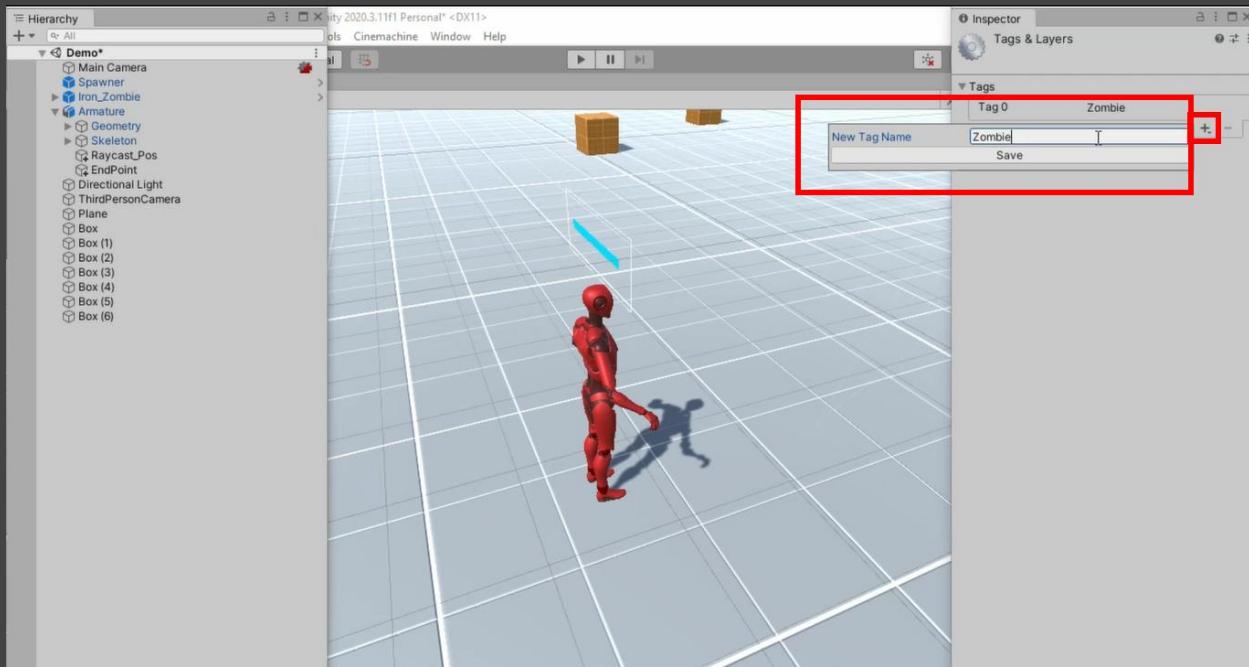
    if (Physics.Raycast(BulletPos.transform.position, BulletPos.transform.forward, out Rayhit, range))
    {
        // Debug.Log(hit.transform.name);

        if (Rayhit.transform.gameObject.tag == "Zombie")
        {
            Rayhit.transform.gameObject.SendMessage("EnemyDamage", Damage);
        }
    }
}
```

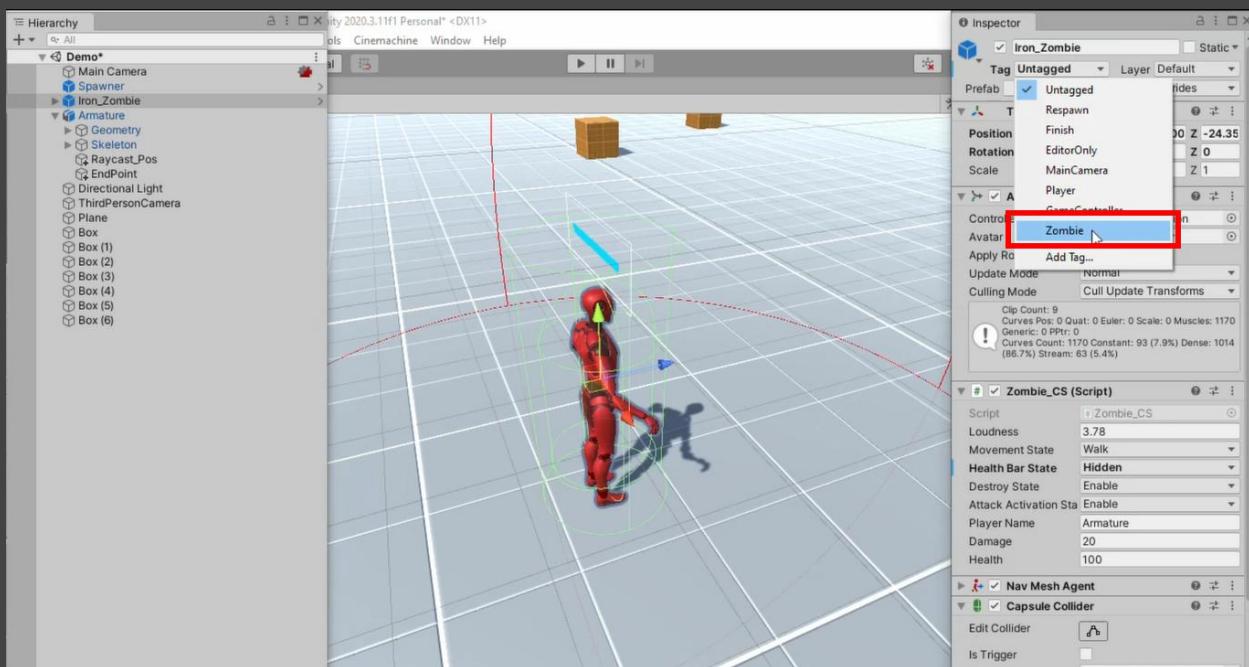
After that go to the tag in the (inspector) menu and click (Add tag)



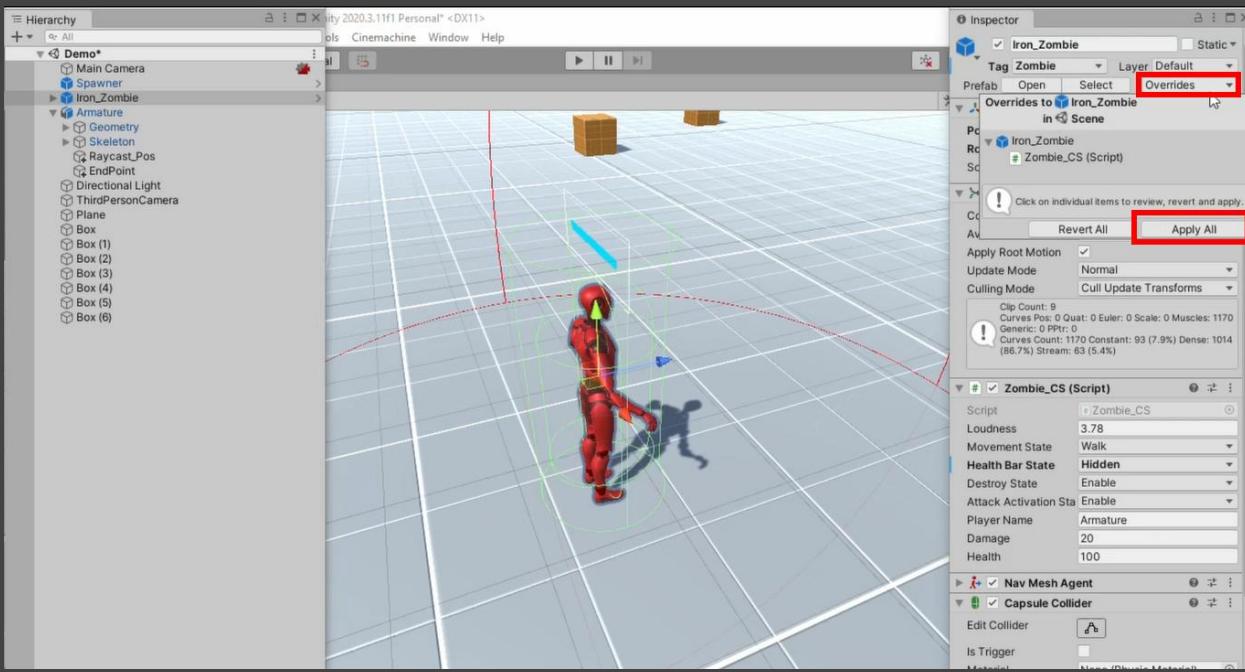
Then go to (tags) and click the plus sign then paste the (Zombie) text and save it



After you create the (Zombie) tag select your zombie character then go to the (tag) in the inspector menu and choose (Zombie)



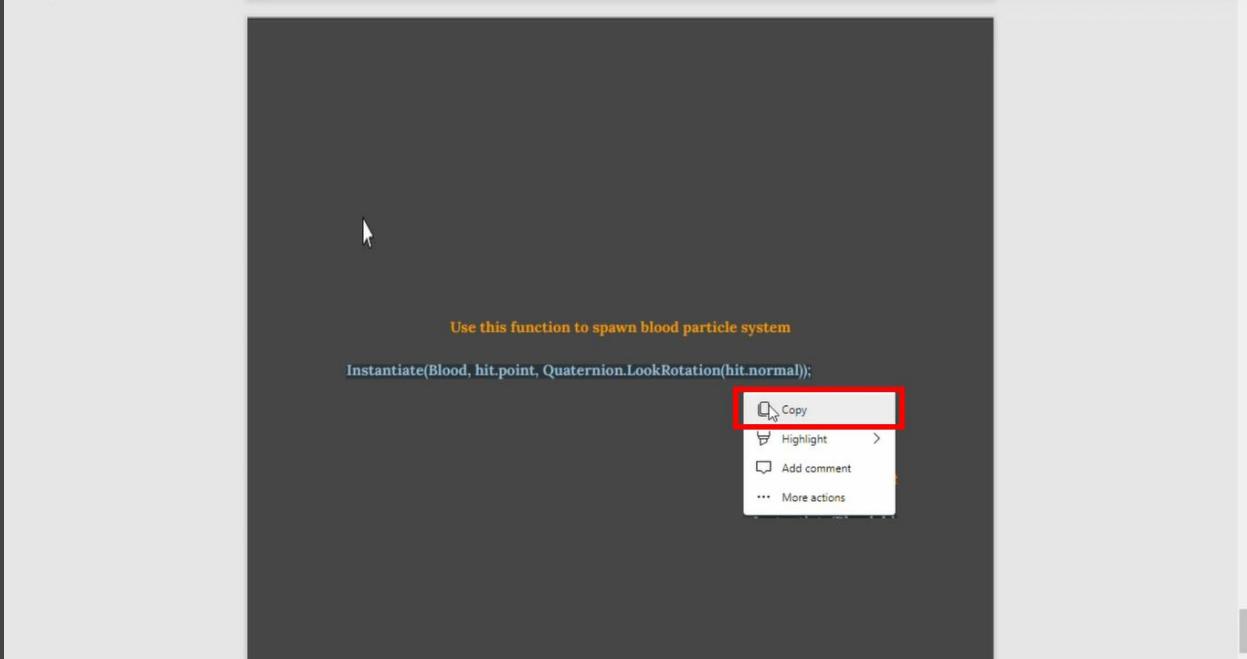
Then click (Apply All) to save everything to the zombie prefab



Blood VFX

USE (Instantiate Object)

Use this function to spawn a blood VFX from the zombie, just copy the code from the document and paste it into your bullet script



Here is the code

```
Instantiate(Blood, hit.point, Quaternion.LookRotation(hit.normal));
```

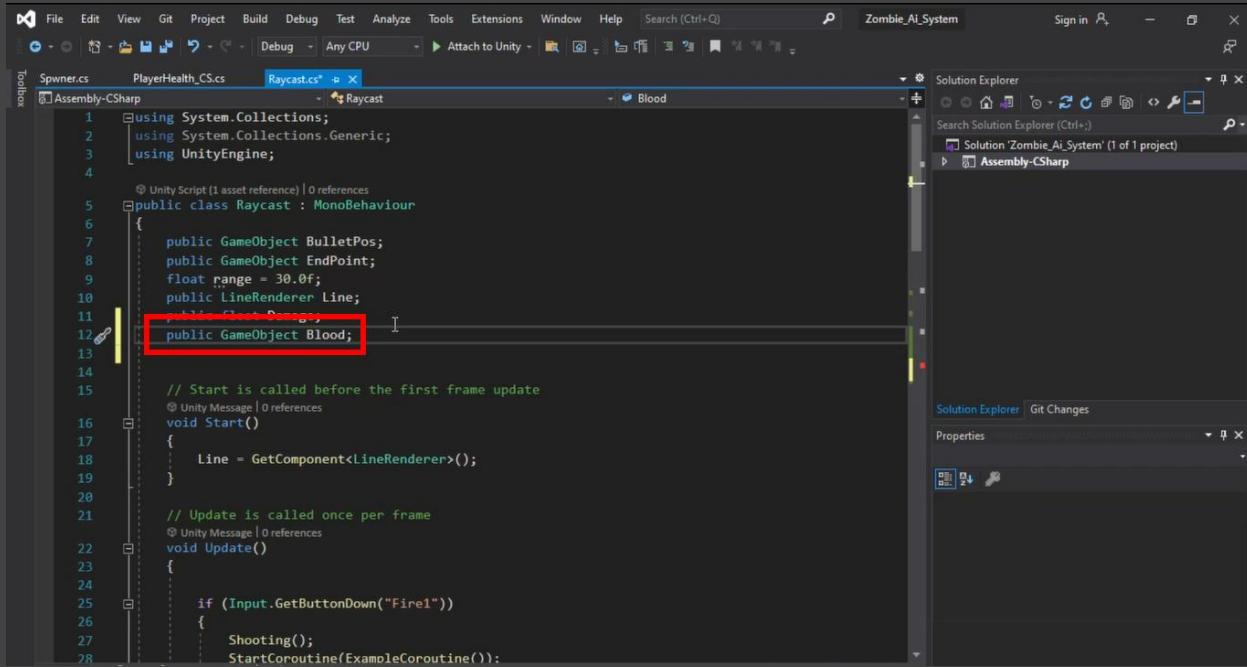
Paste the code here and make sure to replace the (RaycastHit) variable with your own variable for instance we replaced the (hit) with (RayHit)

The screenshot shows the Unity Editor's code editor with the file "Raycast.cs" open. The code is as follows:

```
39     RaycastHit Rayhit;
40
41     if (Physics.Raycast(BulletPos.transform.position, BulletPos.transform.forward, out
42         Rayhit, range))
43     {
44         // Debug.Log(hit.transform.name);
45
46         if (Rayhit.transform.gameObject.tag == "Zombie")
47         {
48             Rayhit.transform.gameObject.SendMessage("EnemyDamage", Damage);
49             Instantiate(Blood, hit.point, Quaternion.LookRotation(hit.normal));
50
51         }
52     }
53
54 }
55
56 }
57
58 }
59
60 }
61
62 IEnumerator ExampleCoroutine()
63 {
64     yield return new WaitForSeconds(0.1f);
65     Line.SetPosition(0, BulletPos.transform.position);
66     Line.SetPosition(1, BulletPos.transform.position);
```

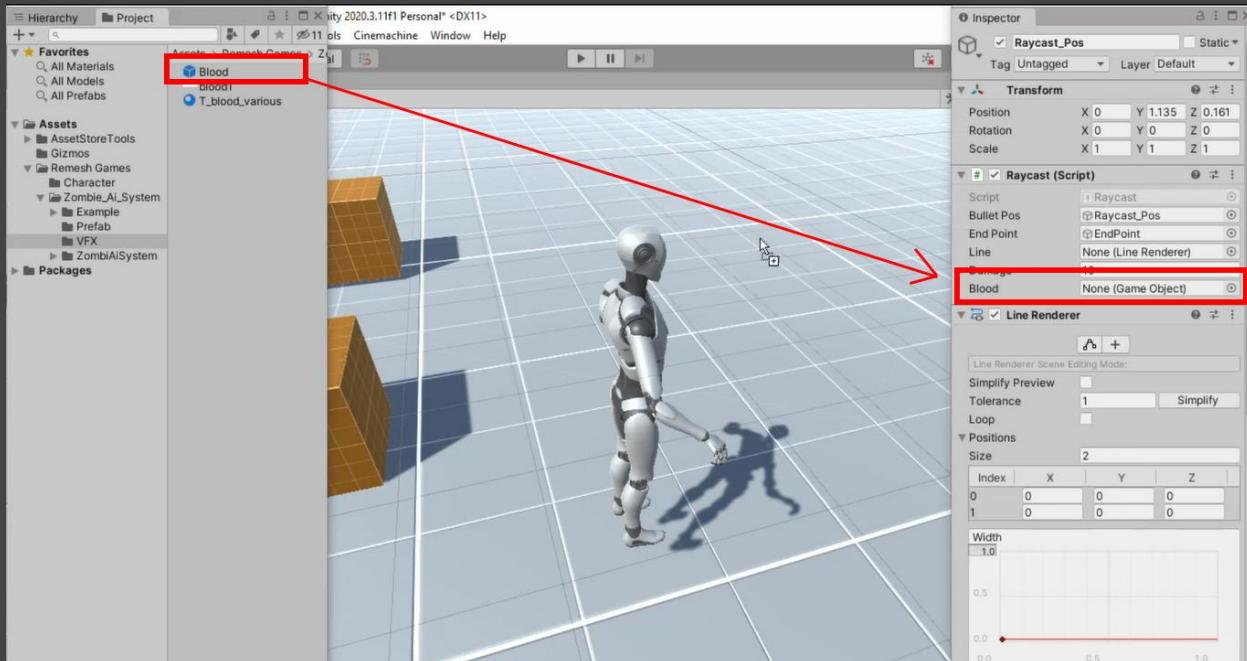
Red boxes highlight the variable names "RaycastHit" and "hit". Red arrows point from these highlighted words to the corresponding words "RayHit" and "RayHit" in the original image.

Next step we need to create a variable for the blood and make sure it is public



```
Spawner.cs PlayerHealth_CS.cs Raycast.cs
Assembly-CSharp
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5     @ Unity Script (1 asset reference) | 0 references
6     public class Raycast : MonoBehaviour
7     {
8         public GameObject BulletPos;
9         public GameObject EndPoint;
10        float range = 30.0f;
11        public LineRenderer Line;
12        public GameObject Blood;
13
14
15        // Start is called before the first frame update
16        void Start()
17        {
18            Line = GetComponent<LineRenderer>();
19        }
20
21        // Update is called once per frame
22        void Update()
23        {
24
25            if (Input.GetButtonDown("Fire1"))
26            {
27                Shooting();
28                StartCoroutine(ExampleCoroutine());
29            }
30        }
31    }
```

Now you can drag your Blood VFX and drop it into the (Blood) variable slot that we created previously



Contact Us

remeshgames@gmail.com