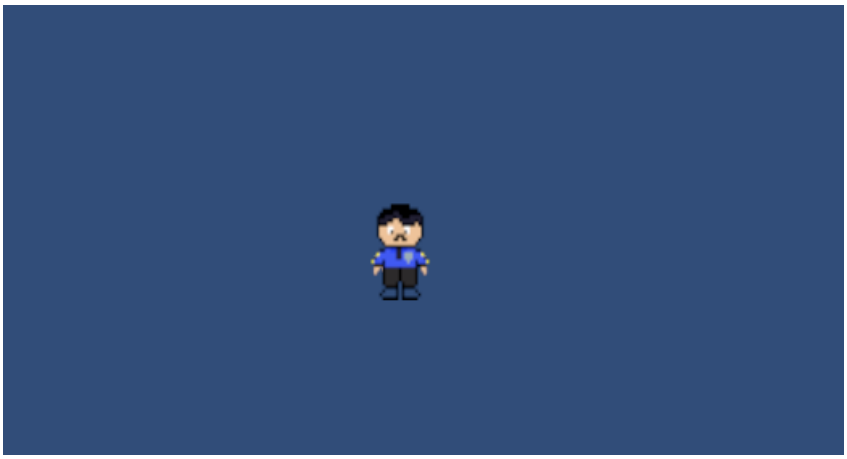
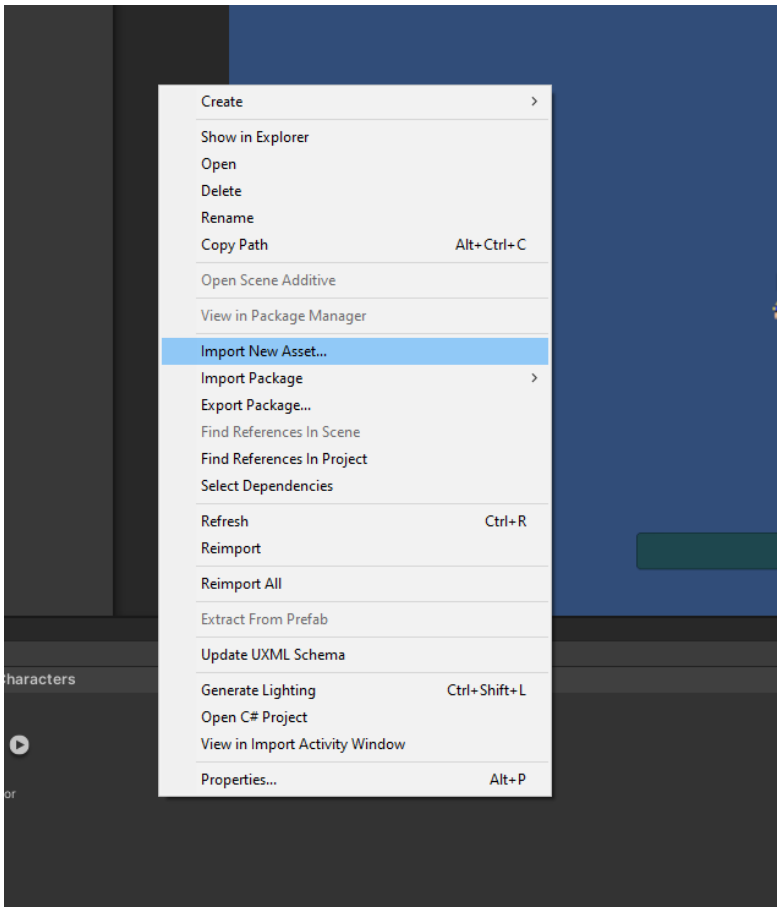
Unity tutorial 2: Player Movement

Step 1: add an object or alternatively import an asset to your project

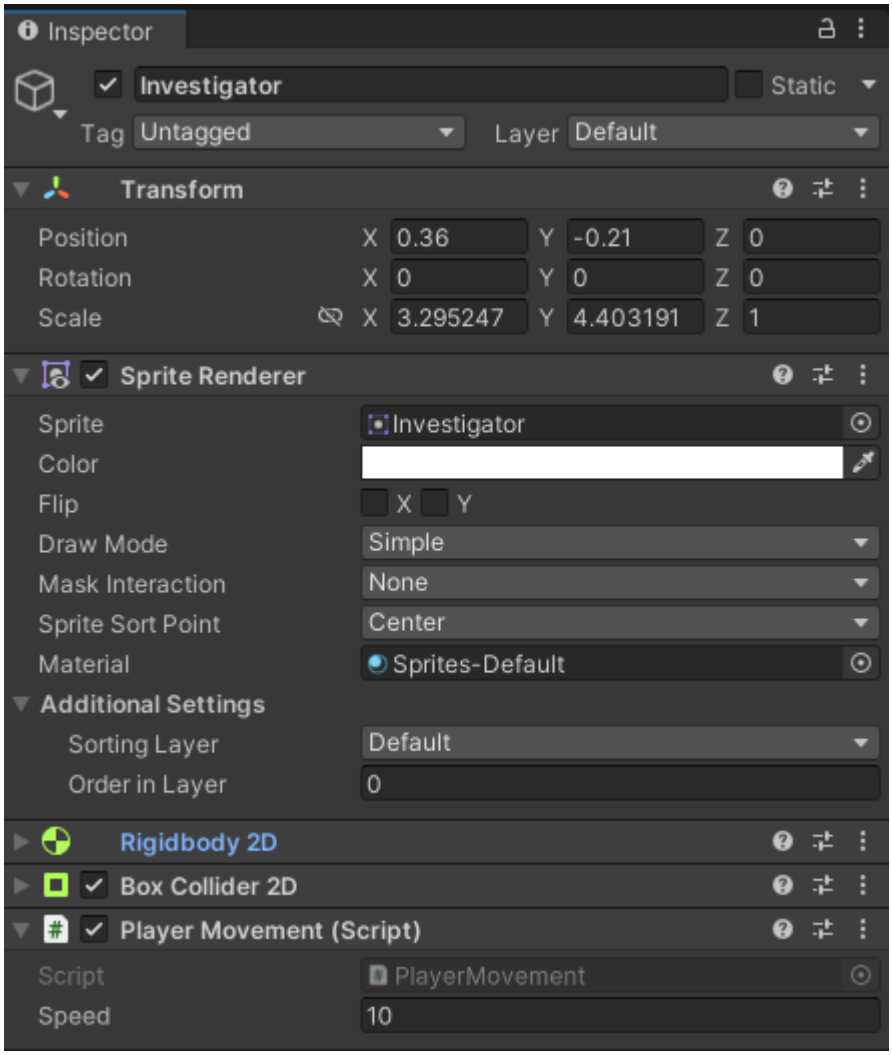


The character can be anything. Even a circle or square works but for this tutorial I will be using a pre-made asset.



Step 2: add a movement script to your character

Now that you have a player character to mess around with in your game you can now add a script to your asset. Remember to give it a nice name, preferably something you can easily remember. After that, you can add a C# script and name it “player movement”



Step 3: Programming the player's movement

When you first open up visual studio you add a variable and name it "speed" and assign a value that will be the player's movement speed. You will need this variable later on when you start working on the movement script.

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

Unity Script (1 asset reference) | 0 references
public class PlayerMovement : MonoBehaviour
{
    public float speed = 10f;
    // Start is called before the first frame update
    Unity Message | 0 references
    void Start()
    {

    }
```

After adding the variable you add a condition to your player which allows you to assign the player's movement along with the key binds.

```
// Update is called once per frame
Unity Message | 0 references
void Update()
{
    if(Input.GetKey(KeyCode.A))
    {
        transform.position += new Vector3(-1, 0, 0) * speed * Time.deltaTime;
    }
    if(Input.GetKey(KeyCode.D))
    {
        transform.position += new Vector3(1, 0, 0) * speed * Time.deltaTime;
    }
    if(Input.GetKey(KeyCode.W))
    {
        transform.position += new Vector3(0, 1, 0) * speed * Time.deltaTime;
    }
    if(Input.GetKey(KeyCode.S))
    {
        transform.position += new Vector3(0, -1, 0) * speed * Time.deltaTime;
    }
}
```

For reference:

- “Transform.position” changes the player's position, basically moving the player when the corresponding key bind is pressed. The movement is then multiplied by “speed” which is the variable you assigned before starting the conditionals.
- “deltaTime” makes it so your player movement runs independently from your frame rate.