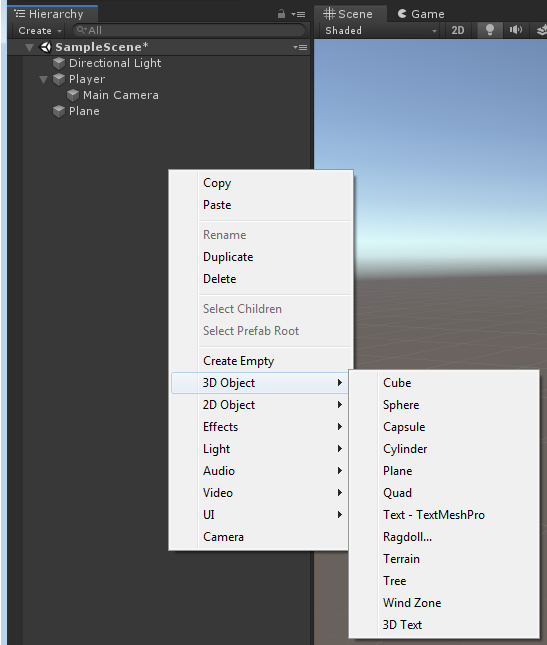
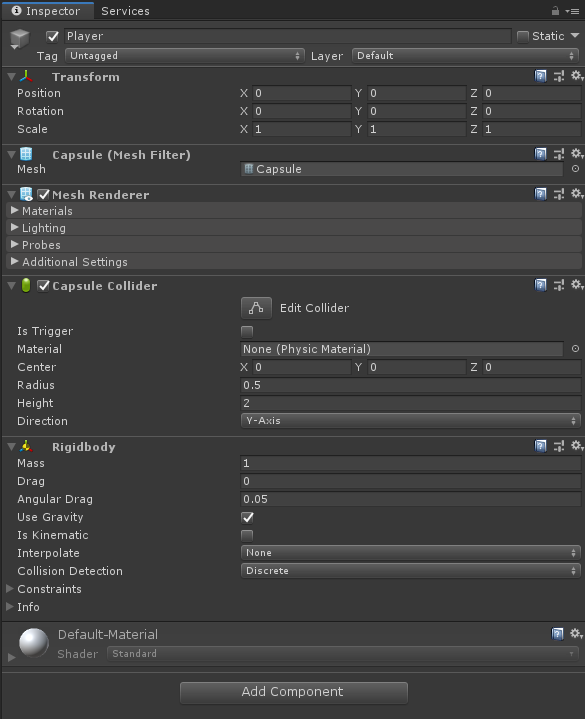
Tutorial 1 – Player Movement

In this tutorial I will demonstrate how to create a player movement script in which the player can run and walk around a scene.

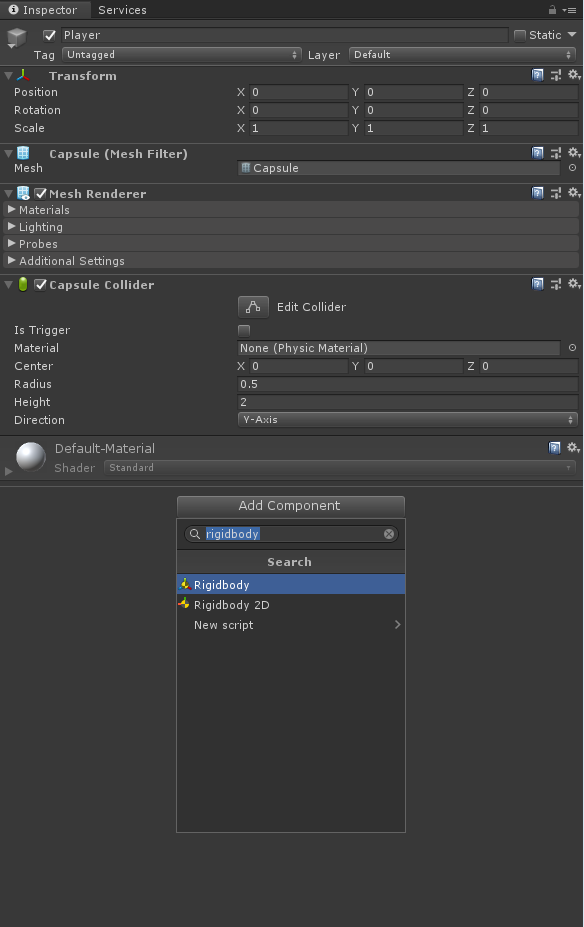
1. Create the player by right clicking on the hierarchy going to 3D objects and spawning a capsule, also spawn a plane to use as the ground from the same location. Position the ground below the player by pressing W and dragging the green arrow downwards.



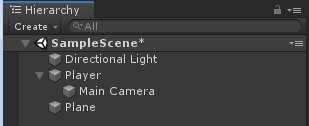
1. Clicking on the Capsule let’s start by changing the name from capsule for clarity, by clicking on the capsule name within the hierarchy.



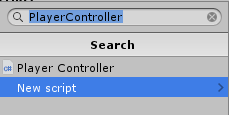
1. Whilst still inspecting the Player click add component at the bottom of the inspector. In the search bar type Rigidbody, this will automatically add mass, gravity and drag to your player.



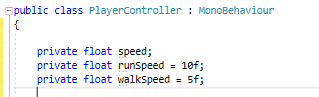
1. Drag the Main Camera into the player capsule within the hierarchy and reposition it using the move tool (W key) so that it’s at an appropriate eye level.



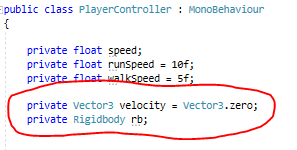
1. Now on the player inspector click add component and create a script called PlayerController. Then open the script by double clicking it in the inspector.



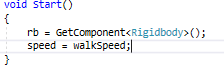
1. In the script we will start by creating three private floats for speed, runSpeed and walkSpeed. The speed float will hold the players current speed using the speeds stored in both run and walk speed floats. We will also set the runSpeed to 10f and walkSpeed to about half of it (feel free to play around with these to find what feels best).



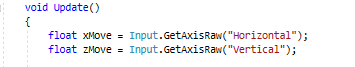
1. We will also need to make two other private variables this being a Vector3 variable for velocity and a rigidbody reference to apply velocity to we will name it rb for short. For the velocity variable we will also make sure the velocity starts at zero so we will make it equal to Vector3.zero



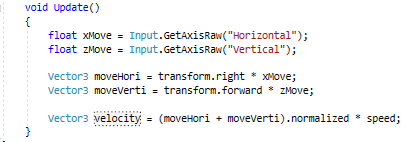
1. Now within the Start void function (the start function runs as soon as the script is activated) we will use the GetComponent to find the rigidbody component on the player gameobject. We shall also set the players move speed to be equal to walkSpeed to begin with, so the player doesn’t start off running.



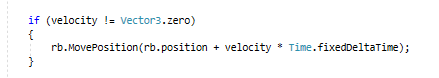
1. Next, we will work on the Update void function (the update function runs every frame) in here we will start by collecting the players key input. For this we will use two new floats called zMove and xMove (z for forward/backward and x for left/right) and set them to record input by using Input.GetAxisRaw and then in brackets “Horizontal” for xMove and “Vertical” for zMove.



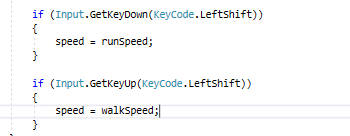
1. From this we will now create three new Vector3 variables called moveHori and moveVerti which will use the players x/yMove floats as well as either transform.right or forward to set the direction the player shall move in. We will then take the moveHori and moveVerti, normalise them and then multiply it by the current speed of the player, this will all be a Vector3 called velocity.



1. Now that we have the players direction and velocity determined we just need to apply it to the characters rigidbody. To do this we will create an if statement at the bottom of the update void that will check if the velocity is not equal to zero and if it is not it will add the velocity to the rb.position and then multiply it over time.



1. To finish off we will use two if statements inside the void update that will track the left shift key being pressed down and released to apply the running and walking movement speeds. We will take Input.GetKeyDown for when the player presses the shift key down to run and then the Input.GetKeyUp for when the player wishes to walk again.



Congratulations you should now be able to walk and run around your scene freely using the wasd or arrow keys. For camera rotation please read my next tutorial.