

Unity tutorial 1

Player movement

First I made a physics material 2D with 0 friction and 0 bounciness so if you don't stick to walls. I gave the character the tag of Player and a rigid body. I also gave the floor a layer of ground.

```
using UnityEngine;

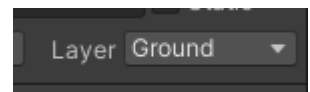
public class PlayerMovement : MonoBehaviour
{
    private float inputX;
    private float speed = 5f;

    private float jumpForce = 10f;
    private bool isJumping = false;
```

I set up a script named PlayerMovement. I created 2 private floats named inputX which is the input along the x axis and speed which is how fast the player moves

JumpForce is how high the player can jump and the bool is a true or false of whether he is jumping or not.

The Rigidbody gives the player physics. We're storing ground layers in variables and giving a ground check to the player in unity.



```
private Rigidbody2D rb;

[SerializeField] private LayerMask groundLayer;
[SerializeField] private Transform groundCheck;
```

```
void Start()
{
    rb = GetComponent<Rigidbody2D>();
}
```

Referencing and storing the character's rigidbody

Updates it's horizontal value every frame, then if the jump input is pressed then set the value IsJumping to true

```
void Update()  
{  
    inputX = Input.GetAxisRaw("Horizontal");  
    if (Input.GetButtonDown("Jump")) isJumping = true;  
}
```

This is for physics updates, it updates the player's movement. If the jump key is pressed, it will add jump velocity and then update isJumping to false so the player doesn't go up indefinitely.

```
private void FixedUpdate()  
{  
    rb.velocity = new Vector2(inputX * speed, rb.velocity.y);  
  
    if (isJumping && IsGrounded())  
    {  
        rb.velocity = new Vector2(rb.velocity.x, jumpForce);  
        isJumping = false;  
    }  
    else  
    {  
        isJumping = false;  
    }  
}
```

Make an else statement stating if the player is not grounded, set isJumping to false to prevent queue jump.