# Player facing movement’s direction

* Create­ a **3D scene;**
* Create a **cube** called Ground and scale it properly;
* Create a **cube** and call it Player;
* Add a **Rigidbody** to the player and freeze its rotation on the **X** and **Z** axises;
* Create a **script** called Player\_Movement and add it to the player;
* Add this lines of code in order to **move the player**:

public float speed;

void Start ()

{

sound = GetComponent<AudioSource>();

rb = GetComponent<Rigidbody>();

}

void Update()

{

rb.velocity = new Vector3(Input.GetAxis("Horizontal")\* speed, rb.velocity.y, Input.GetAxis("Vertical")\* speed);

}

Give a **value** to the **speed** inside the inspector;

* Now to be able to **rotate** **the player** we need to call inside the update method the following void:

void Update()

{

ControlPlayer():

rb.velocity = new Vector3(Input.GetAxis("Horizontal")\* speed, rb.velocity.y, Input.GetAxis("Vertical")\* speed);

}

* Then we have to write what the **ControlPlayer** stands for, so:

void ControlPlayer()

{

float moveHorizontal = Input.GetAxisRaw("Horizontal");

float moveVertical = Input.GetAxisRaw("Vertical");

Vector3 movement = new Vector3(moveHorizontal, 0f, moveVertical);

if (movement != Vector3.zero)

{

transform.rotation = Quaternion.Slerp(transform.rotation, Quaternion.LookRotation(movement), 0.15f);

*//Slerp makes your rotation smoother based on a value from 0 to 1*

}

}

Now you will be able to move your character who will face the direction in which it’s moving.