**PlayerMovement.CS**

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class PlayerMovement : MonoBehaviour

{

private CharacterController character\_Controller;

private Vector3 move\_Direction;//calculate xac dinh cai direction ma ta se di chuyen

public float speed = 5f;//bien cho toc do nhan vat

public float gravity = 20f;//khai bao trong luc(gravity) hay cac physic law cho vat the

public float jump\_Force = 10f;//khai bao luc nhay cua nhan vat theo chieu doc(y)

private float vertical\_Velocity;

void Awake()

{

character\_Controller = GetComponent<CharacterController>();//lay cac thong tin(component) cua character controller ben "player"

}

// Update is called once per frame

void Update()

{

MoveThePlayer();

}

void MoveThePlayer()

{

move\_Direction = new Vector3(Input.GetAxis(Axis.HORIZONTAL), 0f, Input.GetAxis(Axis.VERTICAL));//goi bien public tu phia ben class Axis qua player movement cho tien

//vertical up and down,horizontal is left and right

//print("HORIZONTAL: " + Input.GetAxis("Horizontal"));//neu an cac nut A D tren ban phim

move\_Direction=transform.TransformDirection(move\_Direction);//xac dinh local space

move\_Direction \*= speed\* Time.deltaTime;//giup tang toc do man hinh

ApplyGravity();

character\_Controller.Move(move\_Direction);//dung de nhan vat di chuyen

}//move player

void ApplyGravity()

{

if (character\_Controller.isGrounded)

{

vertical\_Velocity -= gravity \* Time.deltaTime;//giup giu cho nhan vat o duoi dat

//jump

PlayerJump();

}

else

{

vertical\_Velocity-=gravity\*Time.deltaTime;

}

//move\_Direction.y=vertical\_Velocity \* Time.deltaTime;

move\_Direction.y = vertical\_Velocity;

}//apply gravity

void PlayerJump()

{

if (character\_Controller.isGrounded && Input.GetKeyDown (KeyCode.Space))

{ //neu nhan vat o tren mat dat va nhan key "space" tren ban phim

vertical\_Velocity = jump\_Force;

}

}

}//class

Giải thích code:

character\_Controller = GetComponent<CharacterController>();//lay cac thong tin(component) cua character controller ben "player"

=>Lấy các thông tin của character controller bên player

void MoveThePlayer()

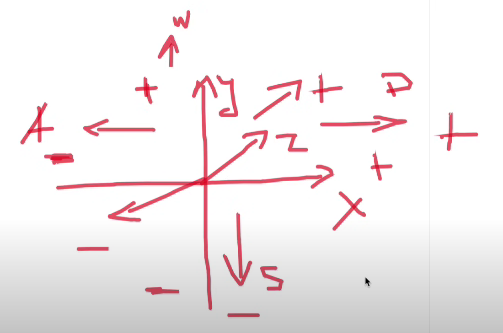
{

move\_Direction = new Vector3(Input.GetAxis(Axis.HORIZONTAL), 0f, Input.GetAxis(Axis.VERTICAL));//goi bien public tu phia ben class Axis qua player movement cho tien

}

Hàm trên được sử dụng để xác định mặt phẳng 3D cho nhân vật di chuyển tương đương với A S D W

A: sang trái;S:lùi;D:sang phải,W:tiến lên



**TagHolder.cs**

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class Axis

{

public const string HORIZONTAL = "Horizontal";//goi bien public tu phia ben class Axis qua player movement cho tien

public const string VERTICAL = "Vertical";

}

Tạo thêm một class là Axis ở **TagHolder.cs** giúp cho việc gọi hàm nó đơn giản hơn thay vì phải gọi chính xác Horizontal thì ta chỉ cần viết hoa HORIZONTAL tránh sai sót

move\_Direction=transform.TransformDirection(move\_Direction);

Transform local space to world space từ đó giúp cho khi nhân vật di chuyển (move\_Direction ) thì sẽ không bị lọt ra khỏi map

move\_Direction \*= speed\* Time.deltaTime;

move\_Direction: Đây thường là một vector (kiểu Vector3 hoặc Vector2) biểu thị hướng di chuyển.

speed: Một giá trị float đại diện cho tốc độ di chuyển của đối tượng.

Time.deltaTime: Một giá trị float đại diện cho thời gian đã trôi qua kể từ khung hình cuối cùng. Điều này đảm bảo rằng chuyển động không phụ thuộc vào tốc độ khung hình.

character\_Controller.Move(move\_Direction);//dung de nhan vat di chuyen

=>Áp dụng chuyển động cho đối tượng với phép tính toán trên move\_Direction \*= speed\* Time.deltaTime;

void ApplyGravity()

{

if (character\_Controller.isGrounded)

{

vertical\_Velocity -= gravity \* Time.deltaTime;//giup giu cho nhan vat o duoi dat

//jump

PlayerJump();

}

else

{

vertical\_Velocity-=gravity\*Time.deltaTime;

}

move\_Direction.y=vertical\_Velocity \* Time.deltaTime;

}//apply gravity

=>Hàm này giúp cho tạo trọng lực cho nhân vật

void PlayerJump()

{

if (character\_Controller.isGrounded && Input.GetKeyDown (KeyCode.Space))

{ //neu nhan vat o tren mat dat va nhan key "space" tren ban phim

vertical\_Velocity = jump\_Force;

}

🡺hàm này sẽ giúp cho Player có thể nhảy khi nhấn **space** và khi nhân vật đang ở trên bề mặt của map với jump\_Force =10f