

Sistema de movimiento del Player

Este script fue creado con el objetivo de hacer que el Player pueda moverse dentro del juego y de esa manera activar los sistemas de dialogo.

1. Primero se agregaron las referencias para poder programarlo.

```
//Movimiento
public float moveSpeed;

//Salto
private bool canDoubleJump;
public float jumpForce;

//Componentes
public Rigidbody2D rgb;

//Animaciones+
private Animator anim;
private SpriteRenderer theSR;

//Piso
private bool isGrounded;
public Transform groundCheckPoint;
public LayerMask whatIsGround;
```

2. Aquí le decimos que el jugador va a poder moverse en los planos (x, y) a la velocidad deseada en el unity.

Le decimos que detecte el suelo todo objeto que tenga la etiqueta "isGrounded" para así hacer que solo pueda saltar cuando lo este tocando y no en cualquier área.

También se dio la opción de tener un doble salto, pero no se puede reactivar hasta que vuelva a tocar suelo.

Y se activa una animación en el jugador depende la dirección en la cual se mueva.

```
void Update()
{
    rgb.velocity = new Vector2(moveSpeed * Input.GetAxis("Horizontal"), rgb.velocity.y);
```

```

isGrounded = Physics2D.OverlapCircle(groundCheckPoint.position, 0.2f, whatIsGround);

if(isGrounded)
{
    canDoubleJump = true;
}

if(Input.GetButtonDown("Jump"))
{
    if (isGrounded)
    {
        rgb.velocity = new Vector2(rgb.velocity.x, jumpForce);

    }
    else
    {
        if(canDoubleJump)
        {
            rgb.velocity = new Vector2(rgb.velocity.x, jumpForce);
            canDoubleJump = false;
        }
    }
}

if(rgb.velocity.x < 0)
{
    theSR.flipX = true;
}
else
    if(rgb.velocity.x > 0)
    {
        theSR.flipX = false;
    }
anim.SetFloat("moveSpeed", Mathf.Abs(rgb.velocity.x));
anim.SetBool("isGrounded", isGrounded);
}
}

```