

Ejercicio 2.1

Jonatan Valle Corales

1. Realiza una lista con los conceptos introducidos en este documento, indicar la página y explicar que son y para que se utilizan.

- **Interactuar con el protagonista.**

Primero accedemos al Object Inspector a si podremos ver las asociaciones que tiene Unity de las teclas por defecto. Unity almacena esas asociaciones en diferentes ejes. A parte esta configuración se puede modificar para a demás poder soportar otros dispositivos como un mando con joysticks.

Ahora lo que vamos a hacer es programar que el monigote se mueve hacia la izquierda o hacia la derecha. Para eso hacemos un Script le ponemos `PlayerStateController` y lo abrimos al abrirlo se puede abrir con el `MonoDevelop` como a mi no me funcionaba lo he abierto con `VisualStudio`.

Al poner el código que nos trae el PDF y leemos detenidamente el código podemos ver unos `if` y `else` que son los que se encargan de mover el personaje por ejes. Como ejemplo el `Input.GetAxis("Horizontal")` ese `getAxis` está asociado como antes he explicado por unity a las teclas de flecha direccionales o A y D.

NOTA: Siempre hay que compilar el `VisualStudio` para poder asociarlo las modificaciones al Unity.

Ahora para poder detectar los cambios de estado del monigote tendremos que crear otro Script que escuche estos cambios. Le llamamos `PlayerStateListener`.

Paso el código que nos dan al nuevo Script creado y me fijo que el movimiento se hace por la función `Translate` de la clase `transform`. Cuando utilizo las teclas de movimiento, este evento provoca que se llame al método `onStateChange`, donde se comprueba que la transición de un estado a otro tiene sentido y es correcto y cambiar la animación del monigote.

Finalmente, al iniciar el juego puedo ver que el monigote se mueve de izquierda a derecha.

Experimento.

Incorpora un movimiento hacia arriba del Player. Utiliza el estado jump. Conviene entender cómo funciona el código propuesto. No hace falta que el resultado sea perfecto. El objetivo es probar de modificar el código y tener contacto con MonoDevelop + Unity.

Para incorporar este movimiento de salto pensaba que se podría hacer igual que el otro a partir de ejes, pero claro me di cuenta que no funcionaba también no era viable ya que no podías estar todo el rato comprobando en el eje donde estaba el personaje constantemente para así poder hacer las condiciones para que saltara.

Finalmente buscando en internet y hablando con mis compañeros vi que haciendo un if donde le digo la condición de la tecla Space. Le puedo decir que ejecute un salto con el siguiente código.

```
if (Input.GetKey(KeyCode.Space))
{
    GetComponent<Rigidbody2D>().AddForce(new Vector2(0, 1),
ForceMode2D.Impulse);
}
```

Tengo que hacer un AddForce para poder impulsar el monigote hacia arriba. Al ir probando el new Vector me dado cuenta que eso es el numero para impulsar ahora mismo está en 0, 1 el 0 es el eje X y el 1 es el eje Y si lo aumento dará un salto más grande, pero si aumento el 0 y pongo el 1 a 0 podrá dar un impulso hace delante. A sí que puede ser también buena idea hacerlo de esta manera para juegos que al completar una barra de turbo le des y tenga como un turbo el monigote.