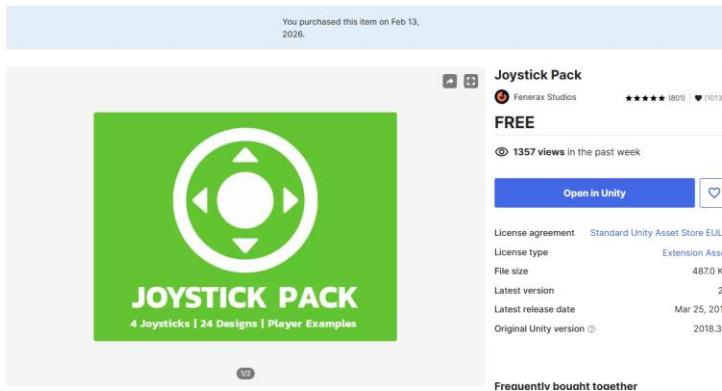
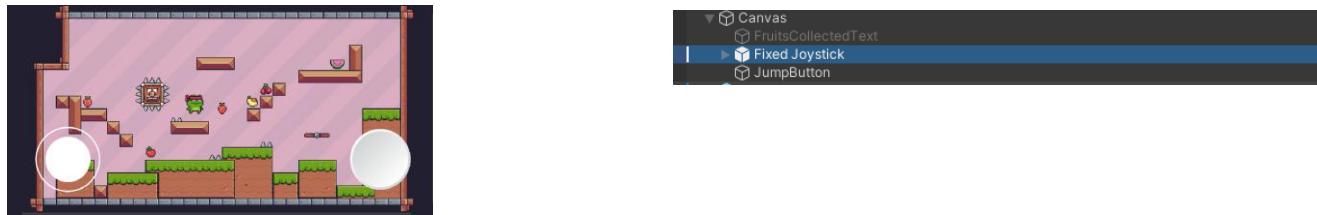


Lección 13:

Implementaremos ahora un Joystick para lo cual importaremos un nuevo asset de assets store.



Y agregaremos el joystick de los prefab de estos assets y un botón para dar el salto todo dentro de un canvas para que se adapte a la resolución de la pantalla en que se corra nuestro juego



Teniendo esto listo es hora de crear un nuevo script para mapear estos controles que deberá llevar casi el mismo que nuestros controles pero mapeado al joysticks este se lo añadimos a nuestro player y referenciamos los componentes y objetos que utilizaremos y solo quedara probar

The image shows the Unity Editor interface. On the left, the code editor displays the 'PlayerMoveJoystick.cs' script. The script uses Unity's C# syntax to control a player character using joysticks. It includes logic for running, jumping, and double-jumping. On the right, the 'Inspector' window shows the component settings for 'Player Move Joystick (Script)'. It references a 'PlayerMoveJoystick' script, a 'Fixed Joystick' component, and sets values for Run Speed Horizontal (2), Run Speed (1.25), Jump Speed (3), Double Jump Speed (2.5), and points to a 'Frog' sprite renderer and animator. This setup maps the joystick inputs to the player's movement and jump actions.

```
public class PlayerMoveJoystick : MonoBehaviour
{
    public SpriteRenderer spriteRenderer;
    public Animator animator;

    void Start()
    {
        rb2D = GetComponent< Rigidbody2D>();
    }

    private void Update()
    {
        if (horizontalMove > 0)
        {
            spriteRenderer.flipX = false;
            animator.SetBool("Run", true);
        }
        else if (horizontalMove < 0)
        {
            spriteRenderer.flipX = true;
            animator.SetBool("Run", true);
        }
        else
        {
            animator.SetBool("Run", false);
        }

        if (CheckGround.isGrounded == false)
        {
            animator.SetBool("Jump", !animator.GetBool("DoubleJump"));
            animator.SetBool("Run", false);
        }
        if (CheckGround.isGrounded == true)
        {
            animator.SetBool("Jump", false);
            animator.SetBool("DoubleJump", false);
            animator.SetBool("Falling", false);
        }

        if (rb2D.velocity.y < 0)
        {
            animator.SetBool("Falling", true);
        }
        else if (rb2D.velocity.y > 0)
        {
            animator.SetBool("Falling", false);
        }
    }
}
```