

A pixel art illustration of a 2D side-scroller level. The scene features several platforms of varying heights and widths, some with brick patterns and others with blue and white checkered patterns. Enemies, including small blue creatures and larger red creatures, are positioned on the platforms. Items such as a key, a heart, and a green gem are scattered throughout. The background consists of a blue sky with white clouds and a blue ground area. The text "2D SIDESCROLLER" is written in large, bold, white letters with a black outline, and "PIXEL ART KIT" is written in smaller, bold, white letters with a black outline below it.

# 2D SIDESCROLLER

## PIXEL ART KIT



# Thank you,

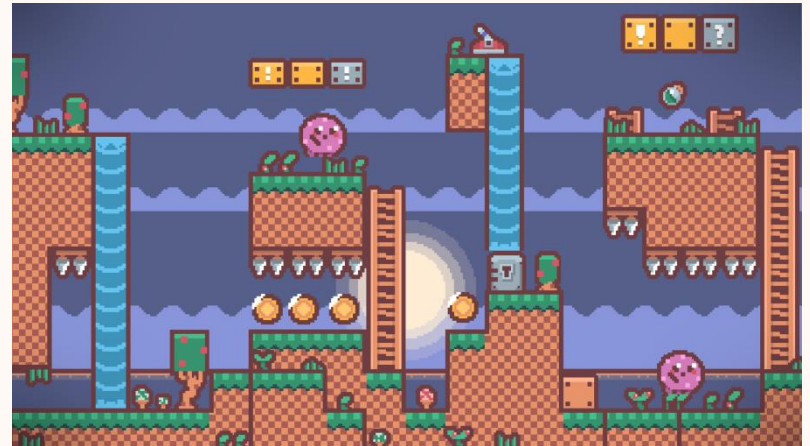
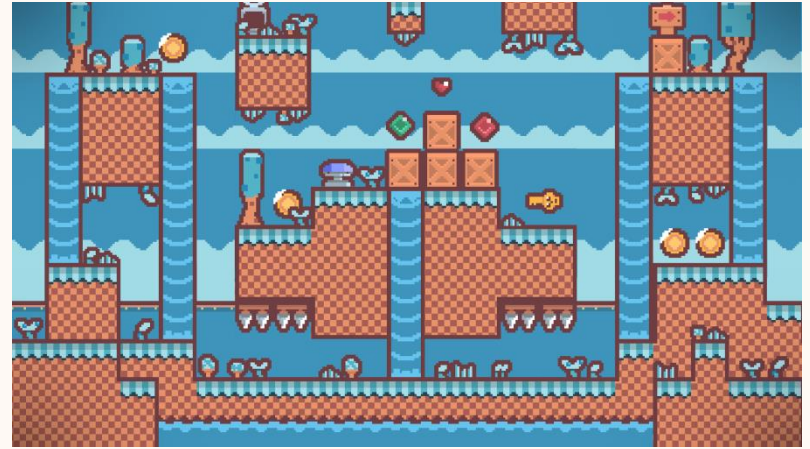
for downloading **2D SideScroller Pixel Art Kit** asset.  
In this following document you will find instructions to use some of the most complicated scripts.

If there is anything that is unclear, send me an email to [luisjoia.info@gmail.com](mailto:luisjoia.info@gmail.com)

**Note:** This is mainly an art asset, scripts are meant only for test purpose, to test movement and gameplay.

## Scripts

- Quick explanation:
  - Player
  - Enemies
  - Camera Follow
  - Audio Manager



# Player Script

## Player Script has 5 main features

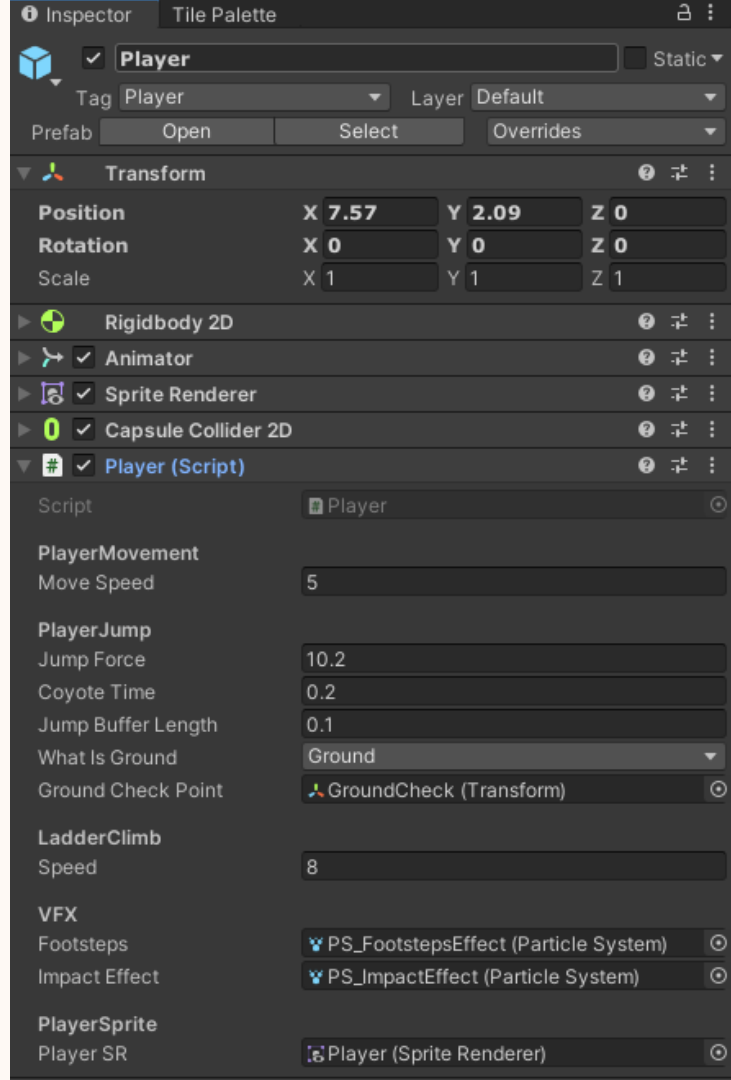
Walk, jump up and down, coyote jump, jump Buffer and a climb ladder component.

**Jump Buffer** prevents player to spam space button making it impossible to double jump when on the ground (ghost jumps).

**Note:** Every ladder component must have a ladder tag.

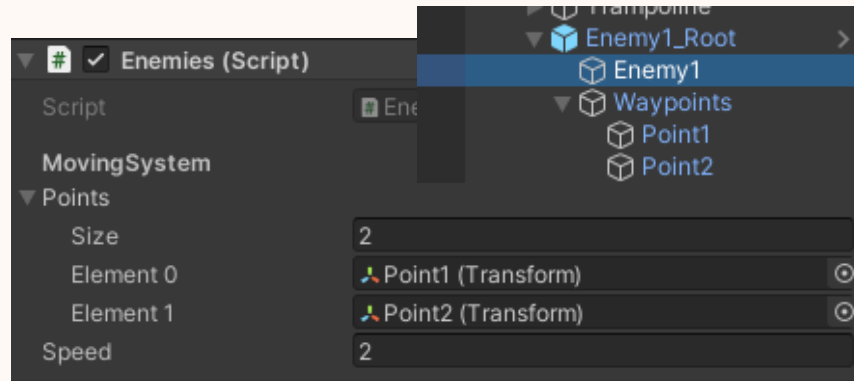
## Player Setup

- Main GameObject with its components
  - PS\_FootstepsEffect (it must be in character controller)
  - PS\_ImpactEffect (it must be in character controller)
  - GroundCheck (at the base of the character)



# Enemy Script

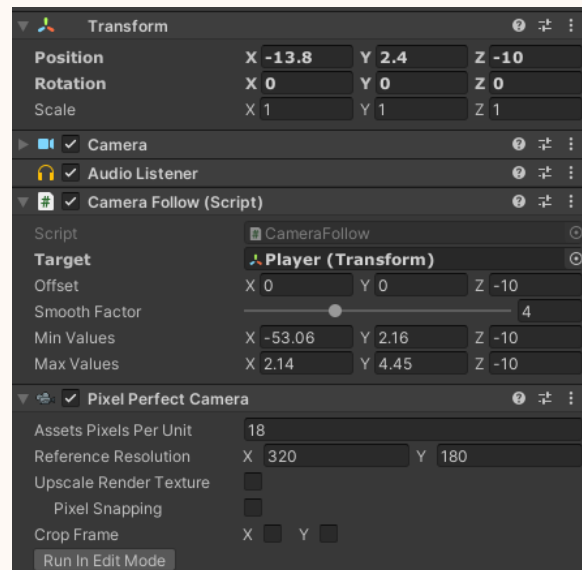
The most important thing to keep in mind is that every enemy object must be inside an empty game object called **Enemy1\_Root** (\_Root must be always after its name). Create Waypoints as enemy's path.



# Main Camera Script

Every main camera must have a -10 in the Z position, both in transform and in the script component. **Smooth Factor** is how fast the camera will follow the player.

Min Values: X-(Bottom left corner) Y- (top left corner) Z- (always -10)  
Max Values: X-(Bottom right corner) Y- (top right corner) Z- (always -10)



# Audio Manager Script

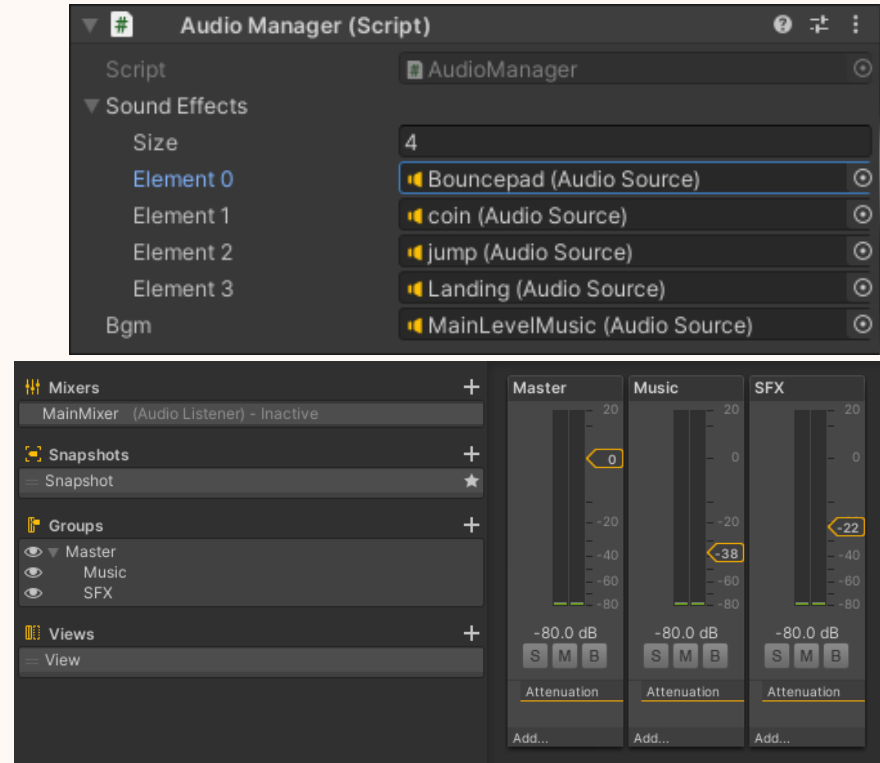
The main GameObject holds the script, its divided in two sections, basic SFX for the world's environment and the background music. Audio Manager holds every sound from SFX to music that is placed inside the script.

Both SFX and music components inside Audio Manager has its output from the main mixer, these outputs are located in Groups.

To add these sounds to the scrip type  
"AudioManager.instance.PlaySFX(2)"



AudioManager.instance.Play**NAMEOFTHEFOLDER**  
(**NAMEOFTHEELEMENT**)



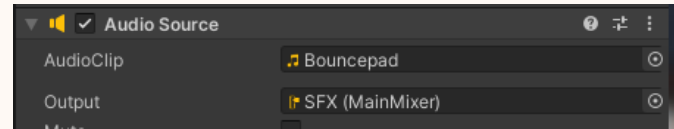
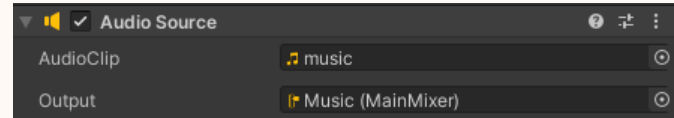
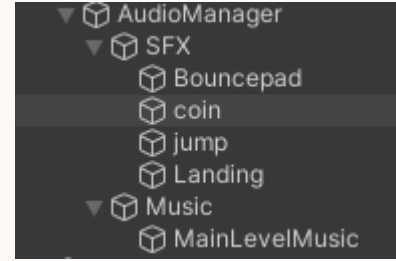
# Audio Manager Script

The main GameObject holds the script, its divided in two sections, basic SFX for the world's environment and the background music. Audio Manager holds every sound from SFX to music that is placed inside the script.

Both SFX and music components inside Audio Manager has its output from the main mixer, these outputs are located in Groups.

To add these sounds to the scrip type  
“AudioManager.instance.PlaySFX(2)”

AudioManager.instance.Play**NAMEOFTHEFOLDER**  
(**NAMEOFTHEELEMENT**)



```
// Jump in the air
if(jumpBufferCount >= 0 && CoyoteCounter > 0f) {
    anim.SetBool("Jump", true);
    rb.velocity = new Vector2(rb.velocity.x, jumpForce);
    jumpBufferCount = 0; // So we don't keep jumping in the air
    AudioManager.instance.PlaySFX(2);
}
```