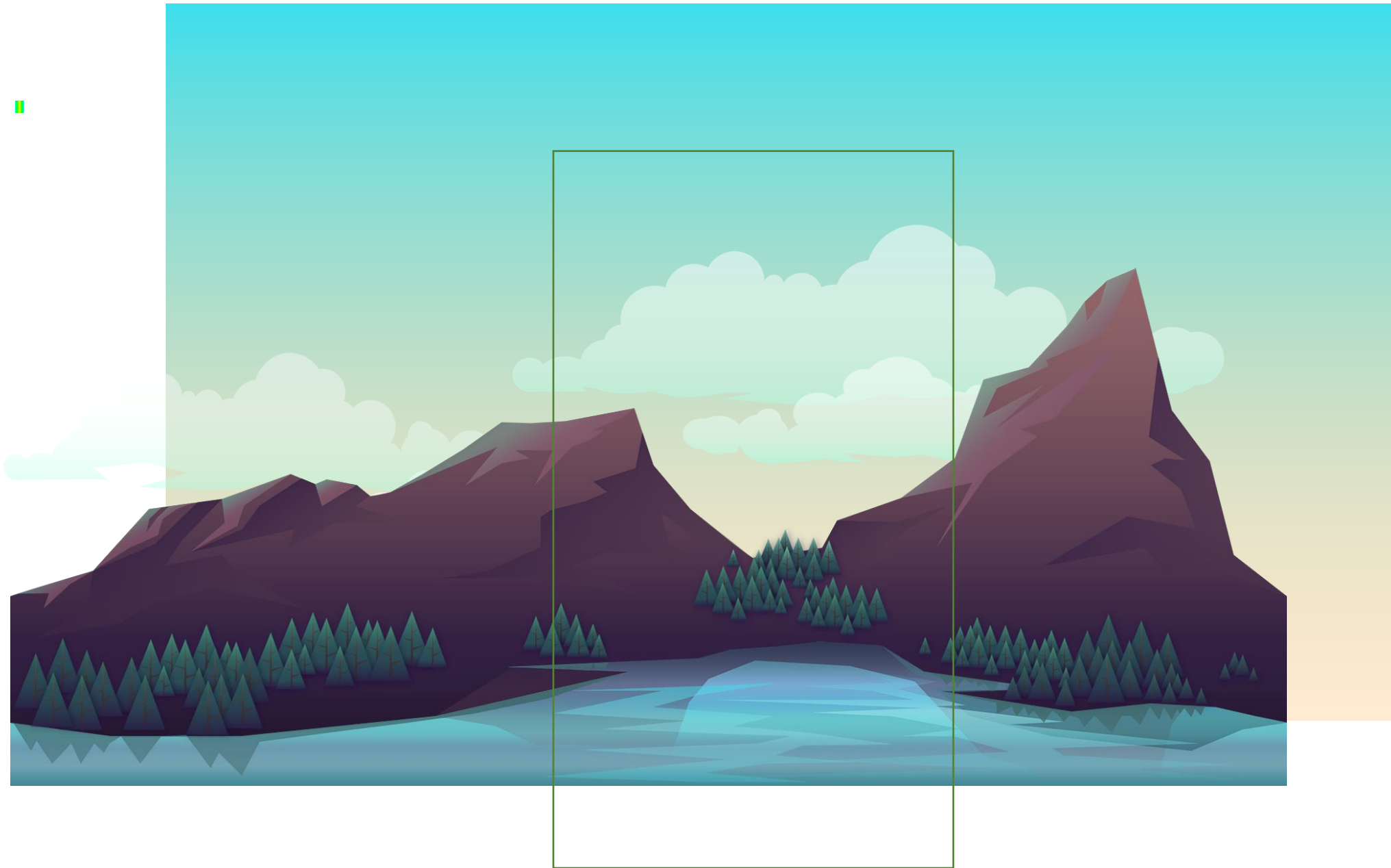
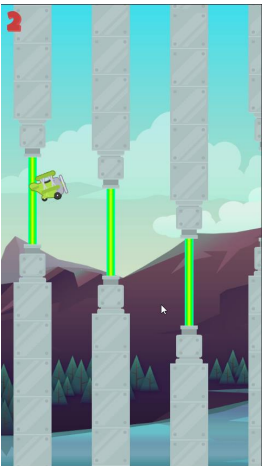
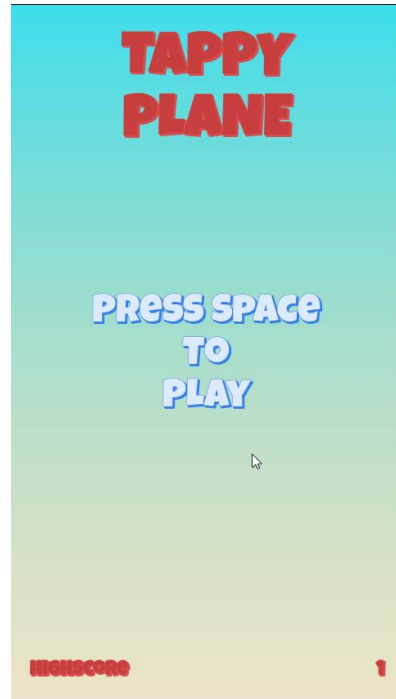


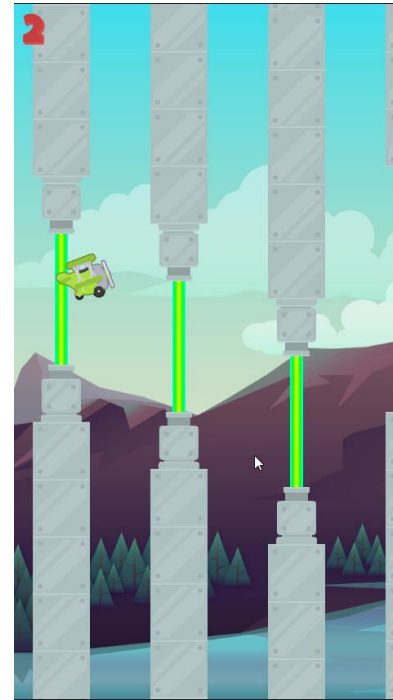
# Assets



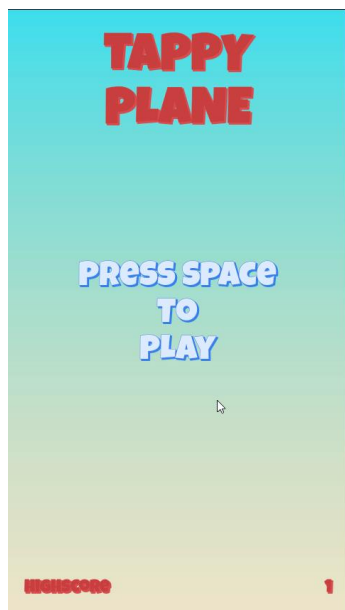
## Main Scene



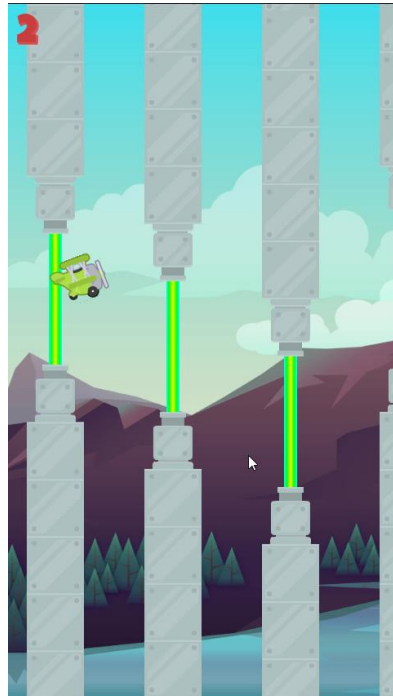
## Game Scene



Main Scene



Game Scene



Plane Scene



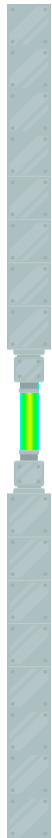
Pipe Scene



Laser Scene



Pipes Scene



# Plane Scene

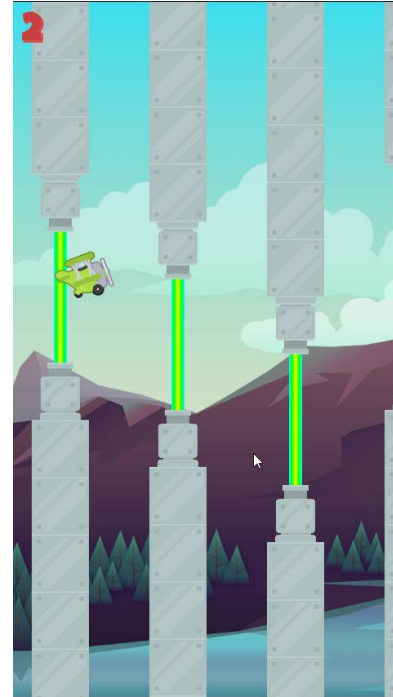


**Falls with gravity**

**Press a key to jump**

**Collides with laser, pipes, floor**

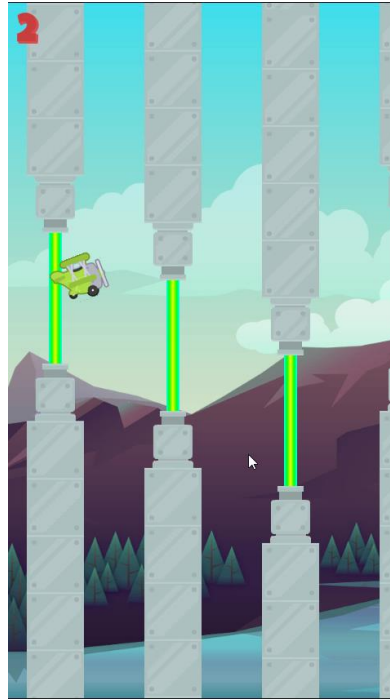
**Animated propellor**



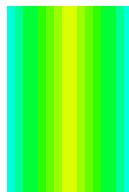
# Pipe Scene



**Collides with plane**

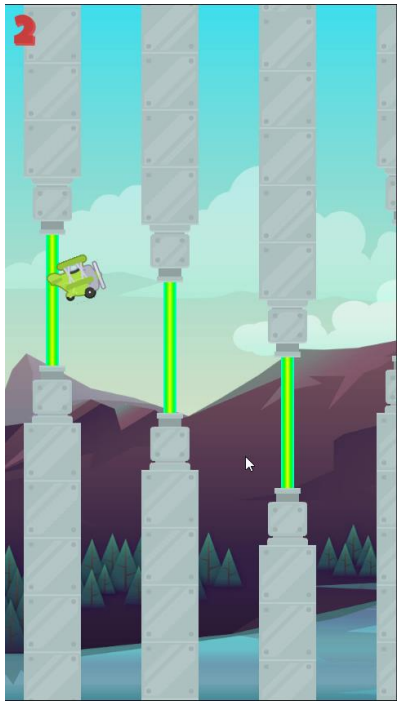


# Laser Scene



**Collides with plane**

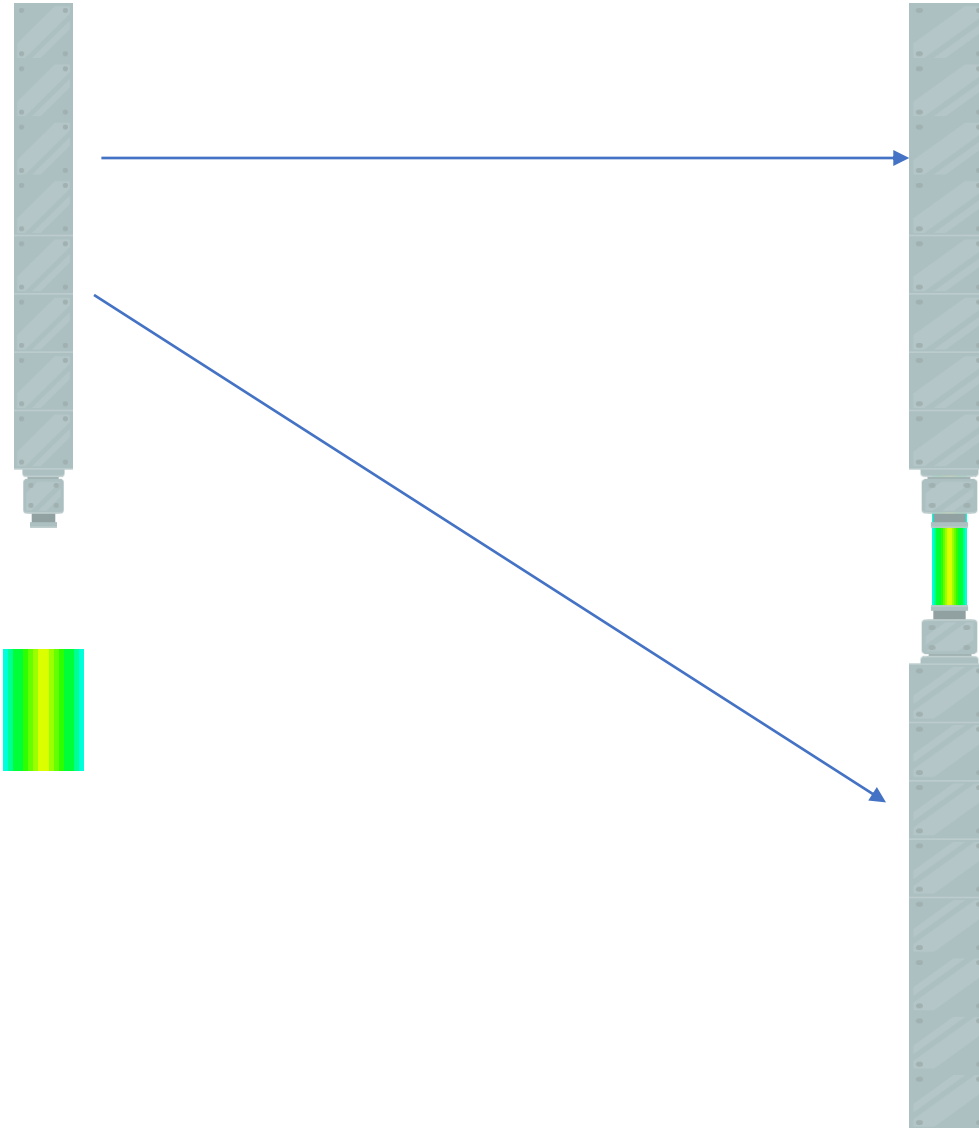
**Signals a score**



# Pipes Scene

Pipe Scene

Laser Scene



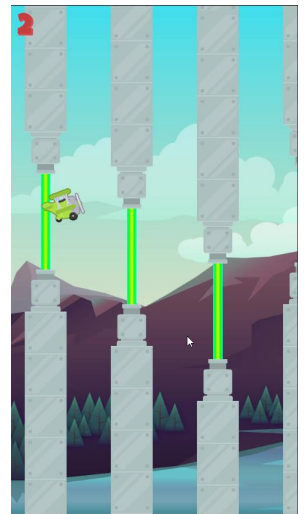
Two Pipe Scenes

One stretched Laser

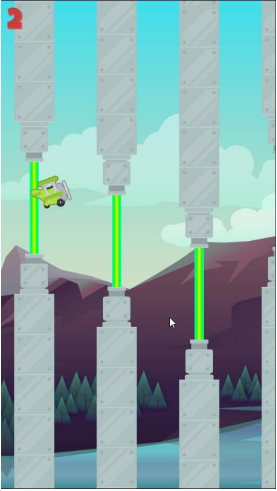
Start off the right of the screen

Scroll from Right to Left with a constant speed

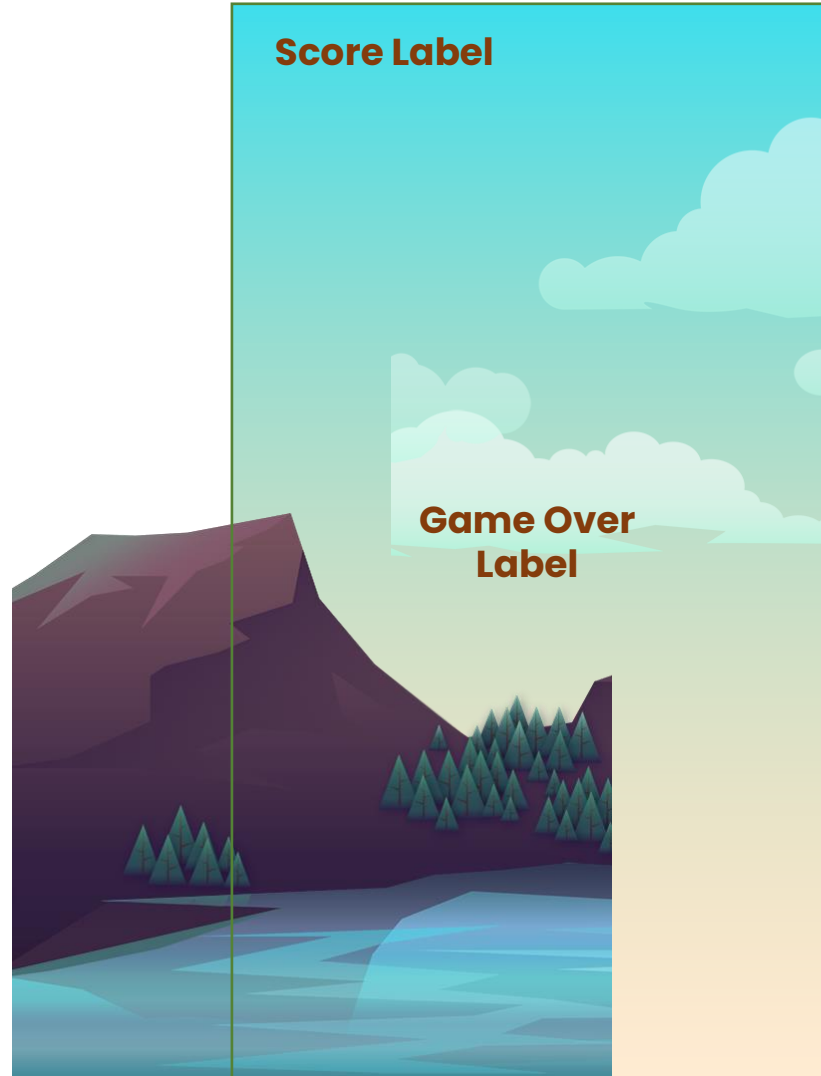
Deleted when going off screen



# Game Scene



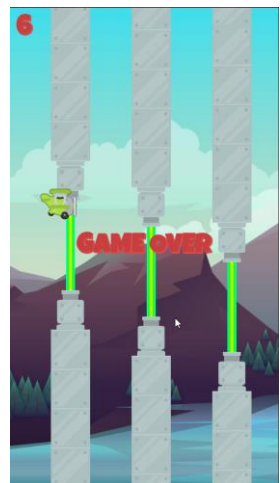
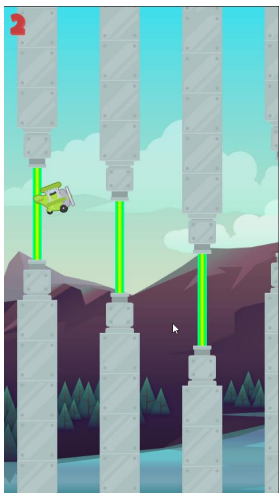
Parallax scrolling  
background



Random Pipes Scene  
Start position between  
points



# Game Scene



Parallax scrolling  
background

Pipes Self  
Destruct  
Off Screen

Score Label

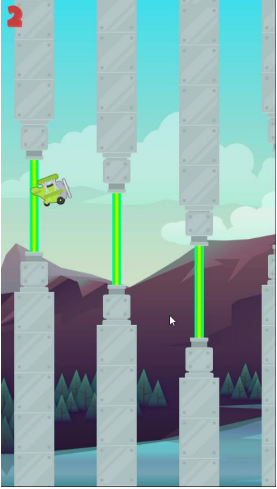
Plane Scene

Game Over  
Label

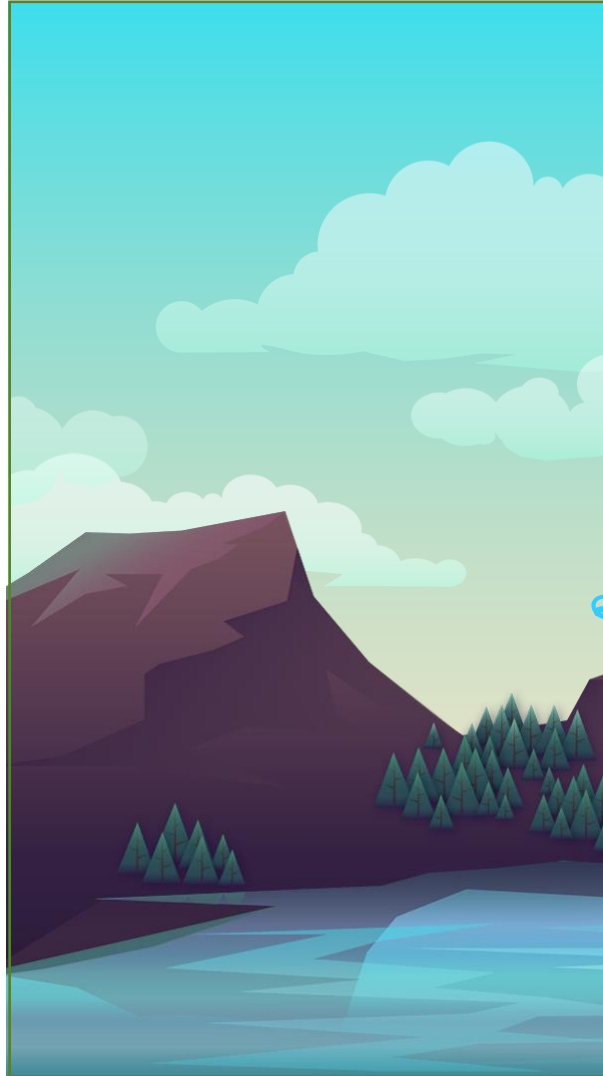
Pipes Scene

Random Pipes Scene  
Start position between  
points

# Game Scene



Node2D



Game Over  
Label

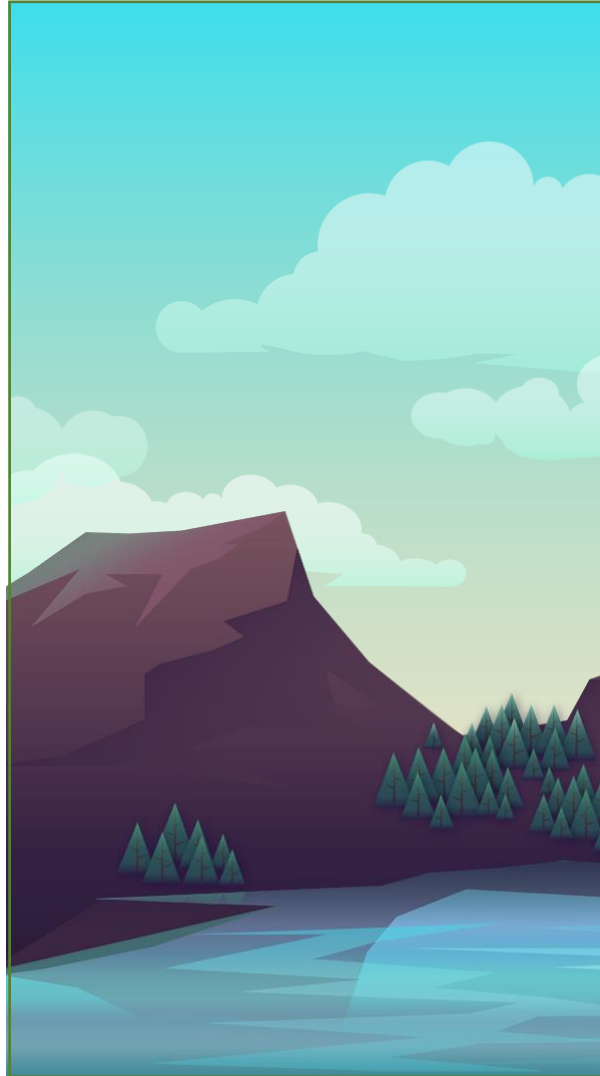
Canvas

Score Label

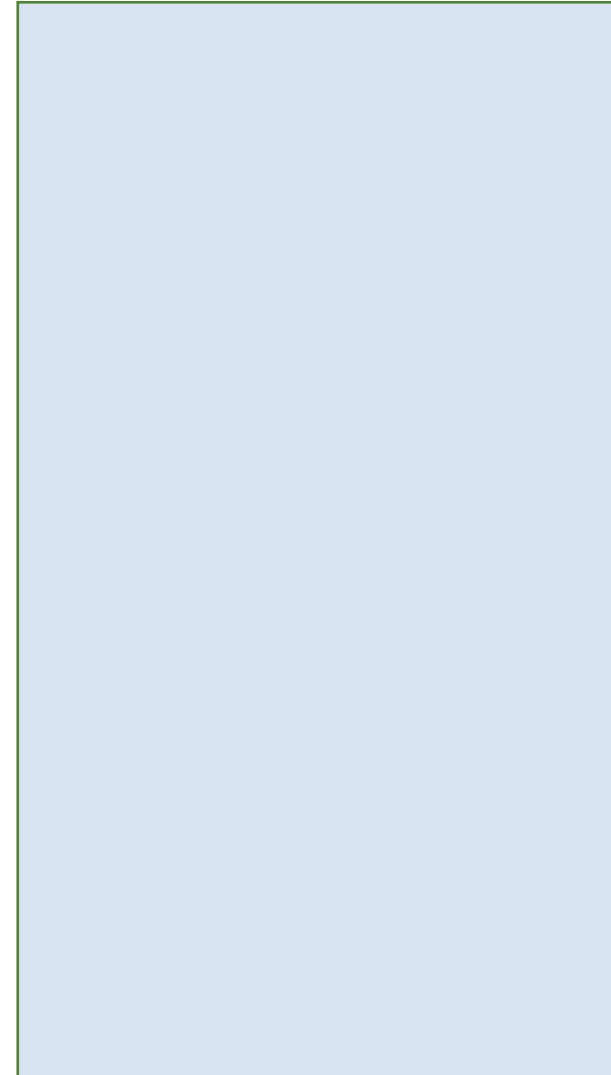
Press Space

# Game Scene

**Node2D**



**Canvas**

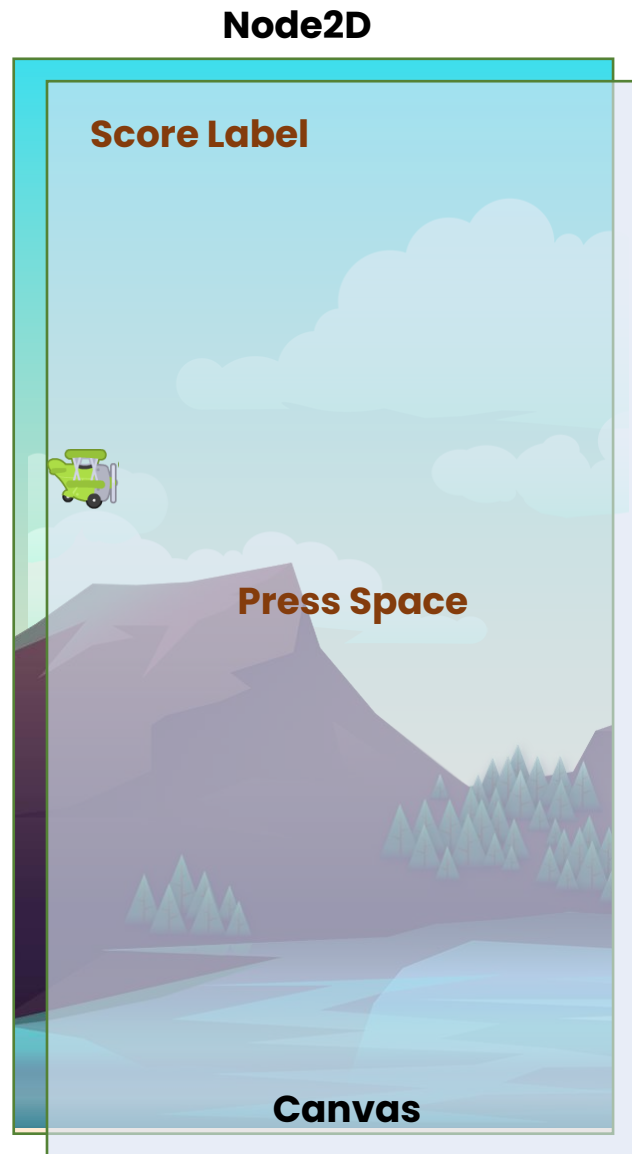


# UI canvas



**Canvas sits on top of the game, and does not „move“**

# UI canvas



**Canvas sits on top of  
the game, and does not  
„move“**

## Main Scene



Show highscore

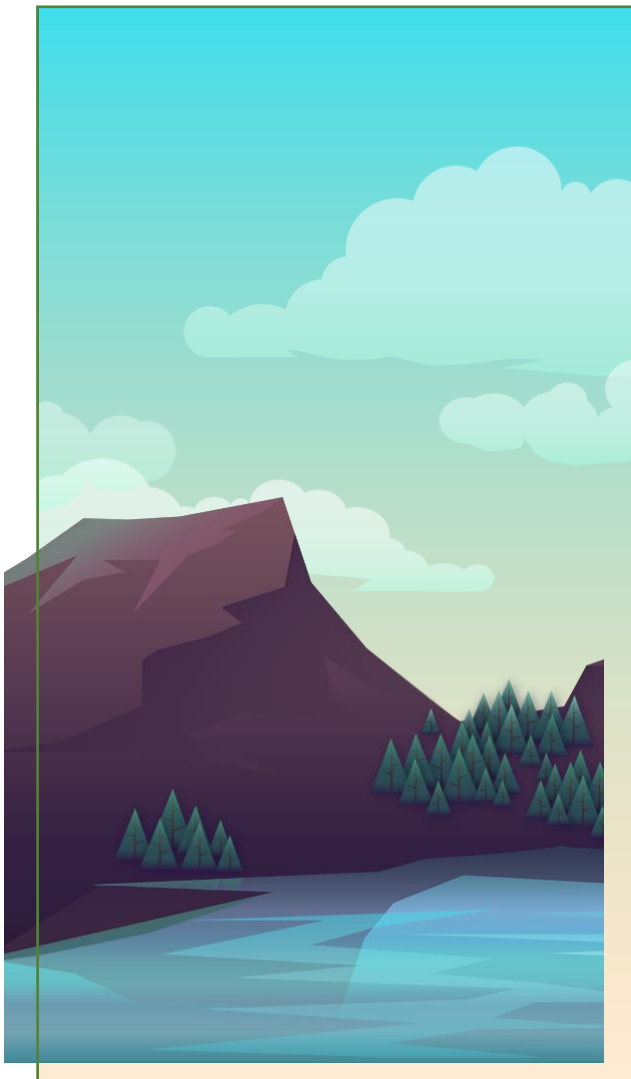
Flashing Press Space  
Label

Title Label

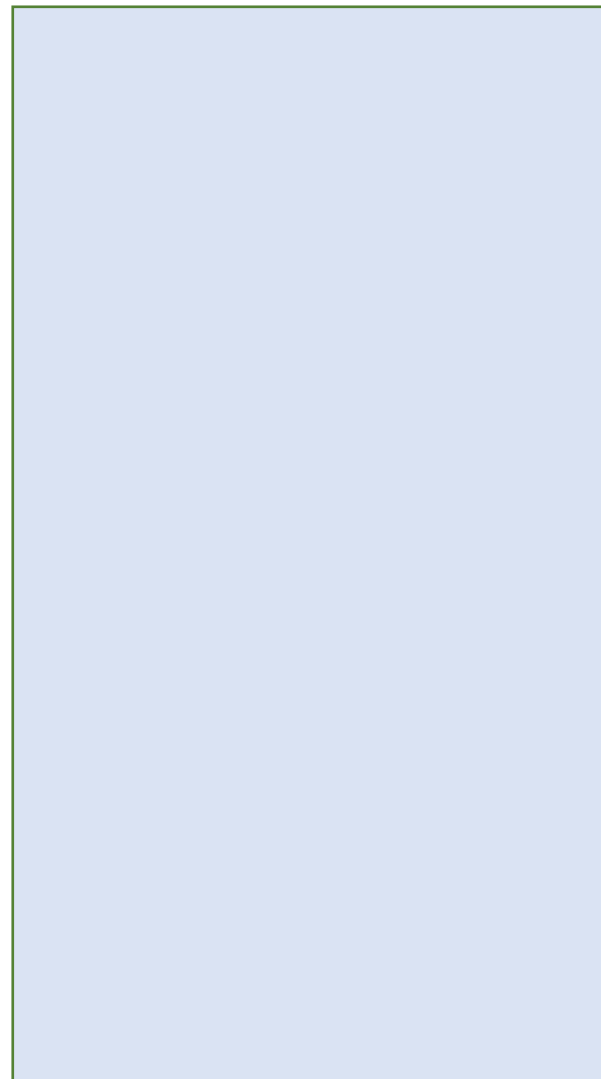
Music

# Canvas

**Node2D Canvas**



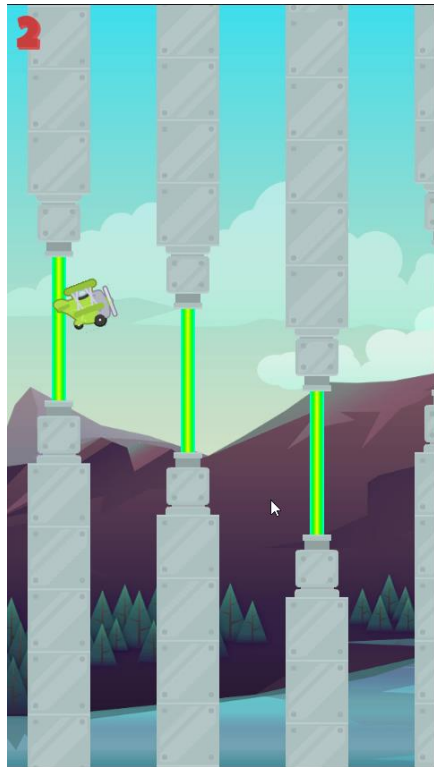
**UI Canvas(s)**



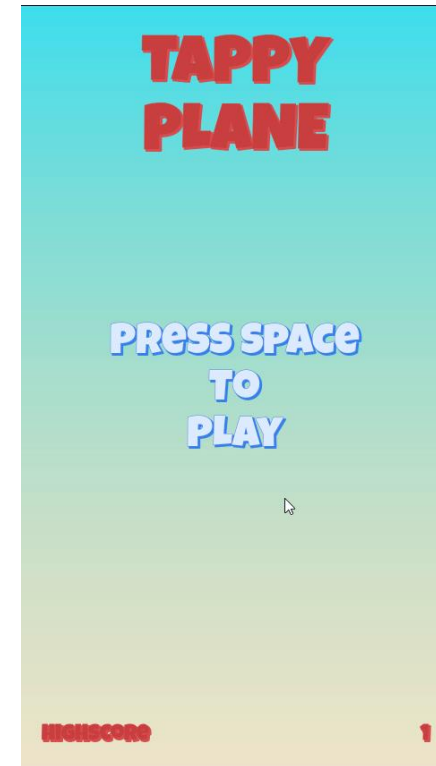
# Game Manager

Hold Score  
Switch Scenes

Game Scene



Main Scene

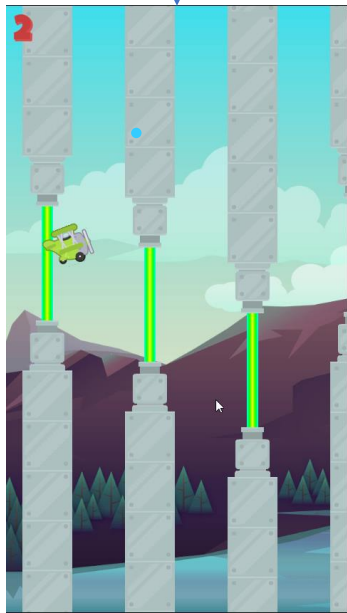




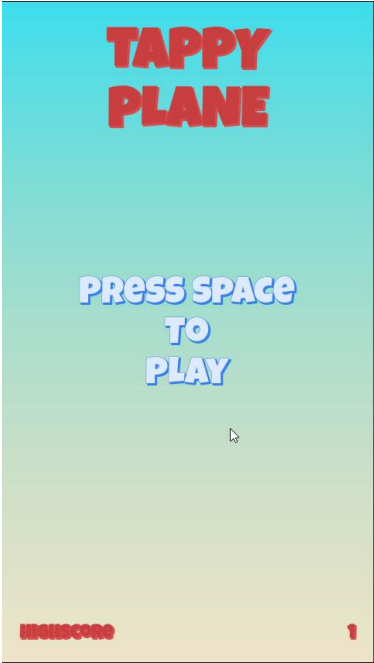
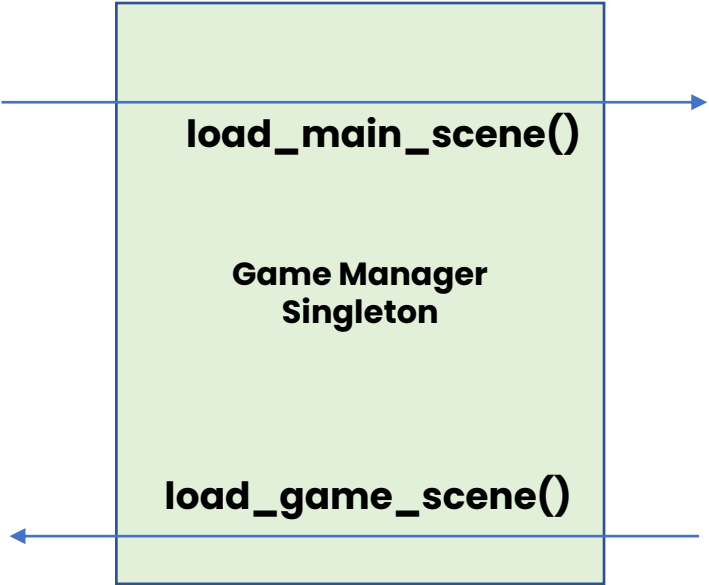
Plane hits floor



on\_plane\_died



Game Scene



Main Scene

Press Space

# Plane Scene

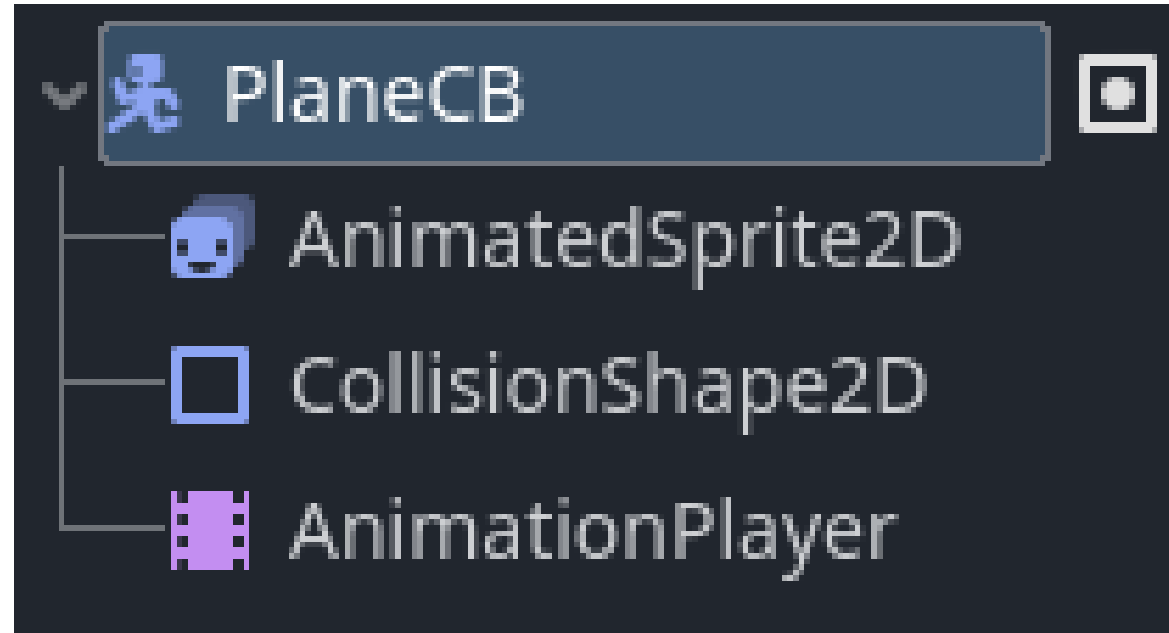


**This will be a `characterBody2D` node.**

**It will contain three other nodes.**

**These nodes are the tree.**

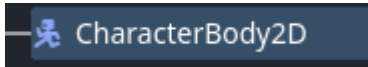
**All together they are the plane „Scene“**



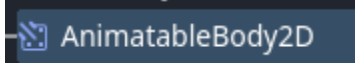
# Physics2D Nodes



Physics body for 2D physics which is static or moves only by script. Useful for floor and walls.



Specialized 2D physics body node for characters moved by script.

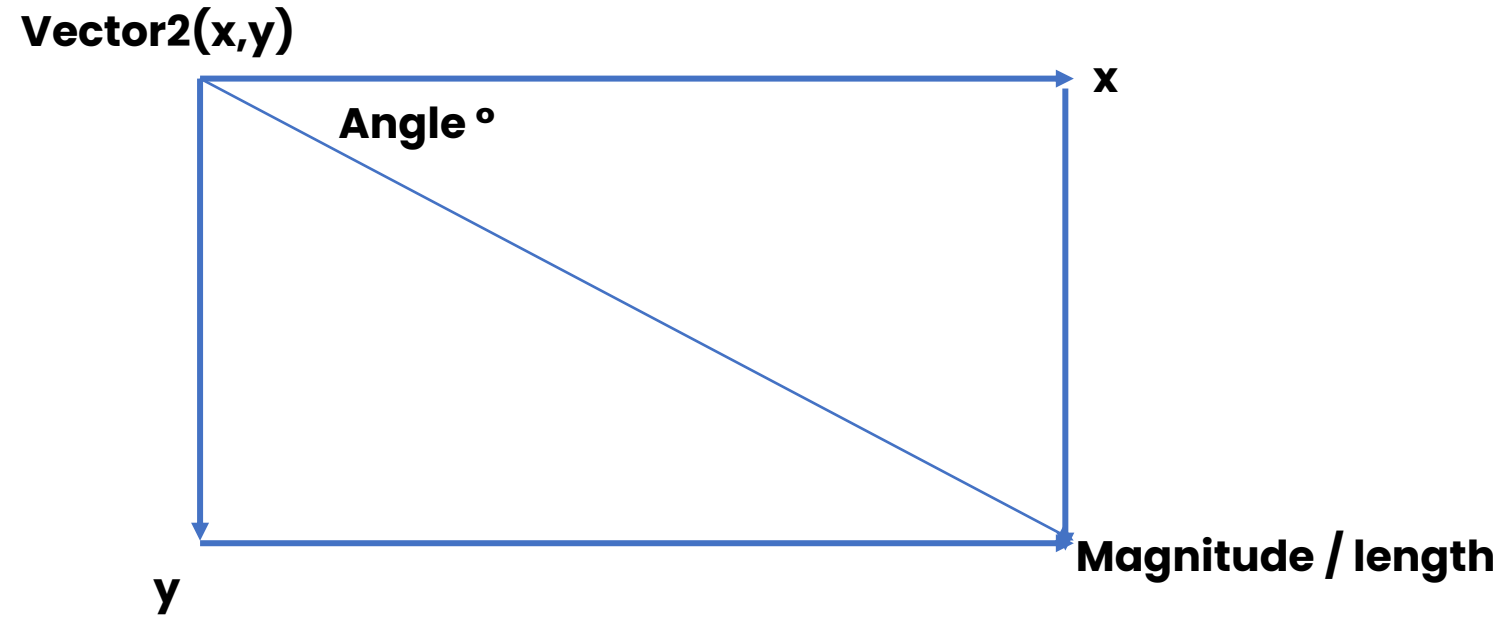


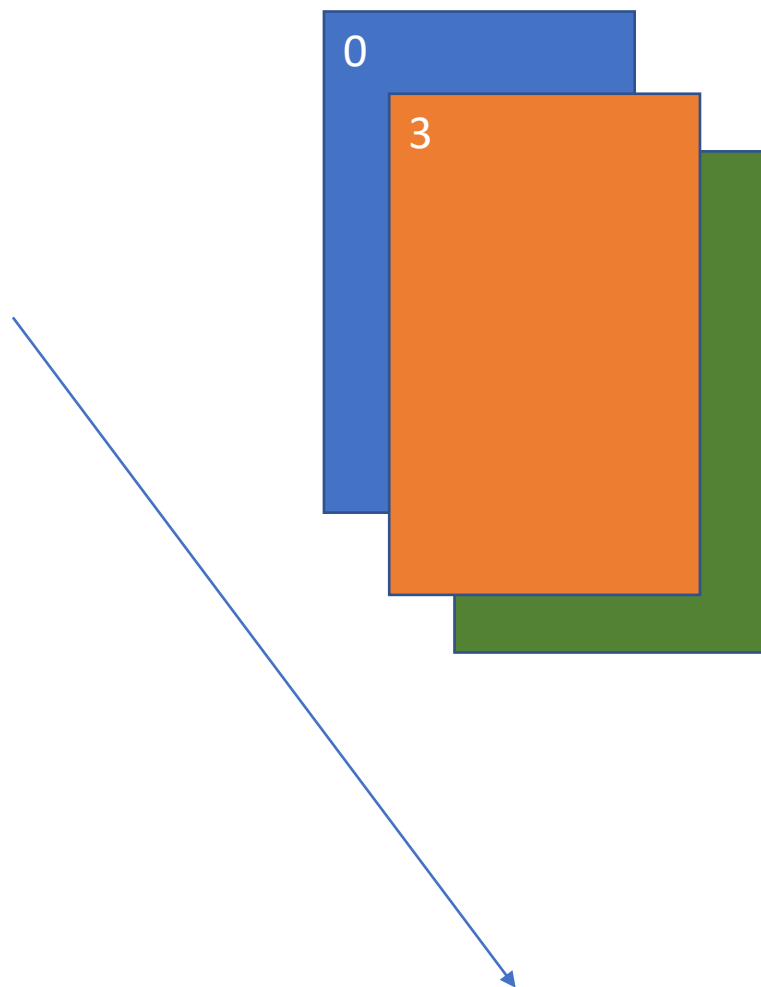
Physics body for 2D physics which moves only by script or animation. Useful for moving platforms and doors.



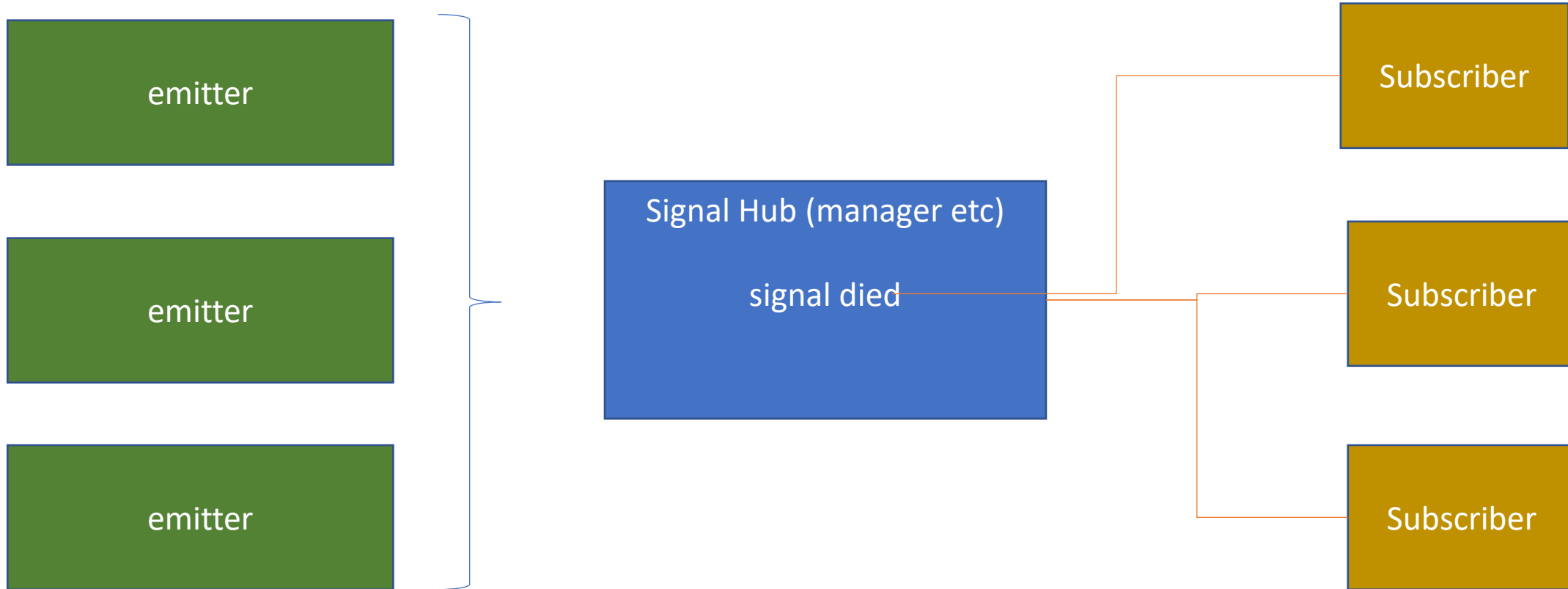
Physics Body which is moved by 2D physics simulation. Useful for objects that have gravity and can be pushed by other objects.

**velocity** is a  $\text{Vector2}(x,y)$

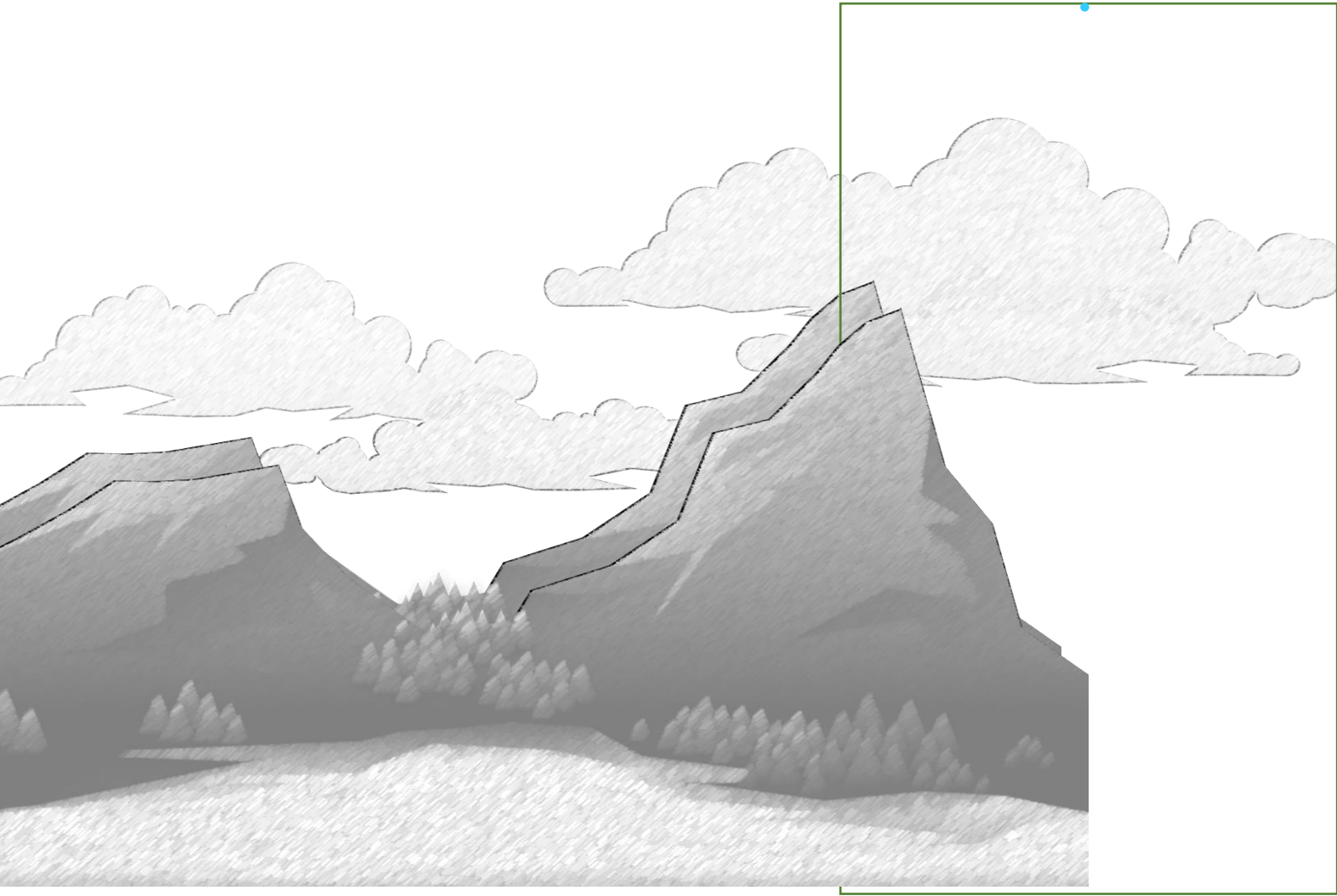




# Event / Signal Hub / bus



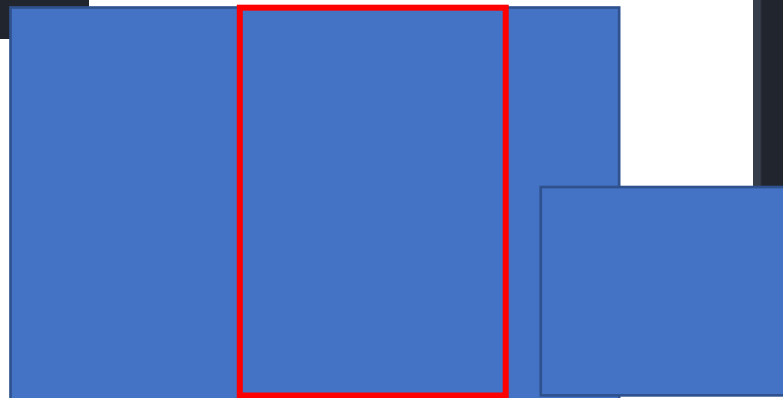
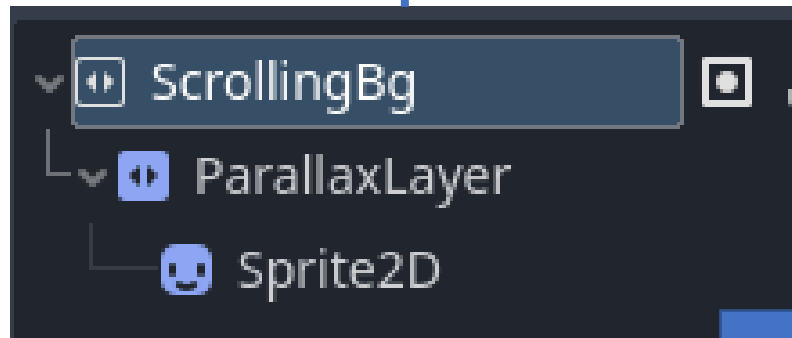
# Parallax



**Scroll\_offset**

- Vector2 scroll\_offset [default: Vector2(0, 0)]  
set\_scroll\_offset(value) setter  
get\_scroll\_offset() getter

The ParallaxBackground's scroll value. Calculated automatically when using a Camera2D, but can be used to manually manage scrolling when no camera is present.



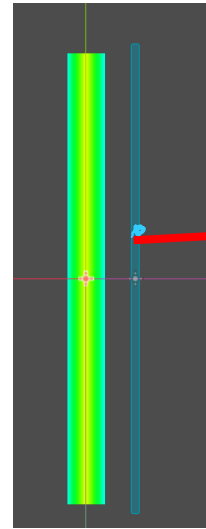
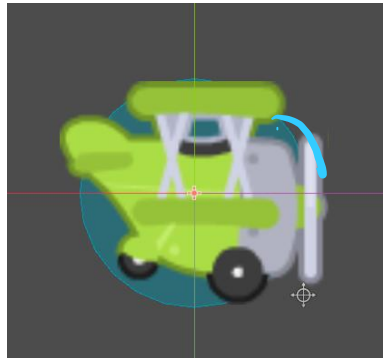


**Layer: What layer our collision object is on**

**Mask: What layer objects will invoke our signal**

Layer						
1	2	3	4	9	10	11
5	6	7	8	13	14	15
Mask						
1	2	3	4	9	10	11
5	6	7	8	13	14	15

# Laser Scene



Layer

1	2	3	4	9	10	11	12
5	6	7	8	13	14	15	16

Mask

1	2	3	4	9	10	11	12
5	6	7	8	13	14	15	16

**Layer: 3 (laser)**

**Mask: 1 (plane)**

```
42 func _on_laser_body_entered(body):  
43     if body.is_in_group(GameManager.GROUP_PLANE) == true:  
44         player_scored()  
45
```

# Layer Mask

Layer

1	2	3	4	9	10	11	12
5	6	7	8	13	14	15	16

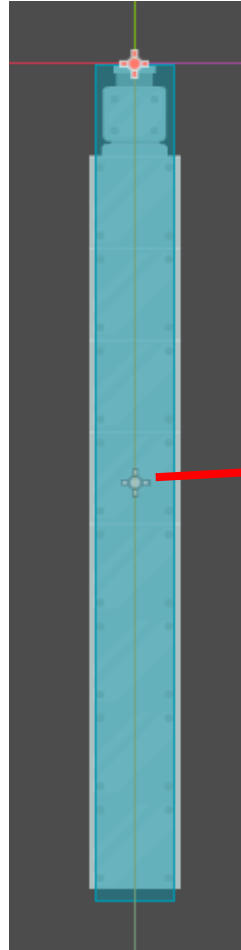
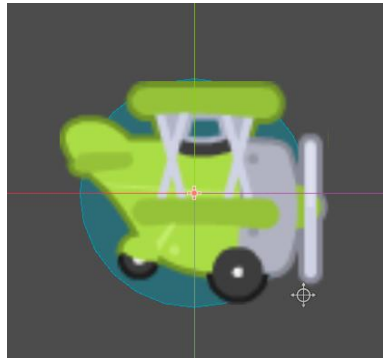
Mask

1	2	3	4	9	10	11	12
5	6	7	8	13	14	15	16

**Layer: What am I on**

**Mask: What layer will make me emit**

# Pipe Scene



Layer

1	2	3	4	9	10	11	12
5	6	7	8	13	14	15	16

Mask

1	2	3	4	9	10	11	12
5	6	7	8	13	14	15	16

**Layer: 2 (pipe)**

**Mask: 1 (plane)**

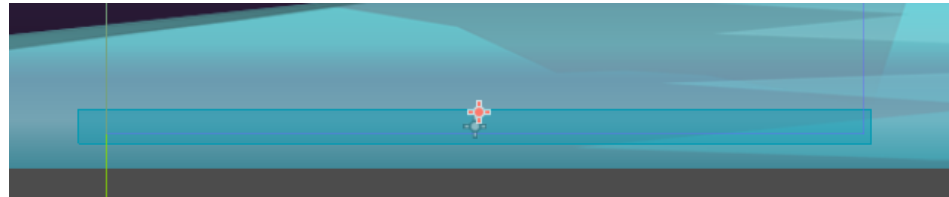
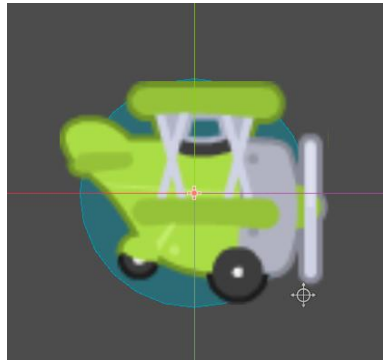
```
46  
47 ▼ func _on_pipe_body_entered(body):  
48   ▼ >| if body.is_in_group(GameManager.GROUP_PLANE) == true:  
49     >| >| body.die()  
50   >| >|
```

## Floor (Game Scene)

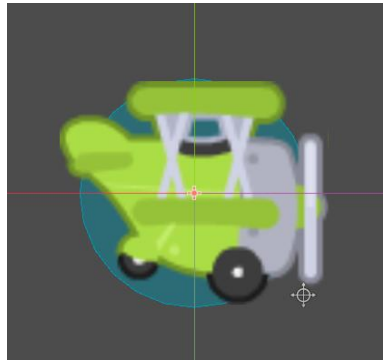
Layer									
1	2	3	4	9	10	11	12	17	18
5	6	7	8	13	14	15	16	21	22
Mask									
1	2	3	4	9	10	11	12	17	18
5	6	7	8	13	14	15	16	21	22

**Layer: 4 (floor)**

**Mask: None**



# Plane Scene



Layer

1	2	3	4	9	10	11	12
5	6	7	8	13	14	15	16

Mask

1	2	3	4	9	10	11	12
5	6	7	8	13	14	15	16

**Layer: 1 (plane)**

**Mask: 4 (floor)**

Is\_on\_floor() detects collisions with the floor