

# WWU GDC Godot Workshop Spring 2025

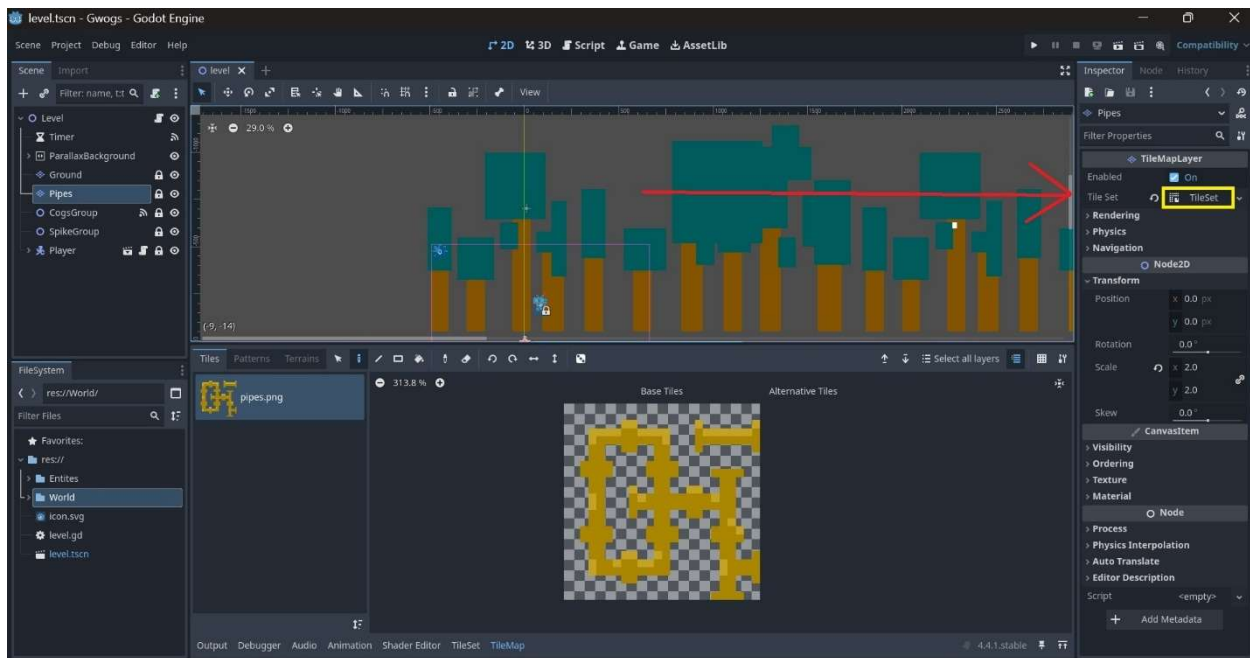
## The Little More Advanced Parts

### One Way Platform TileMap

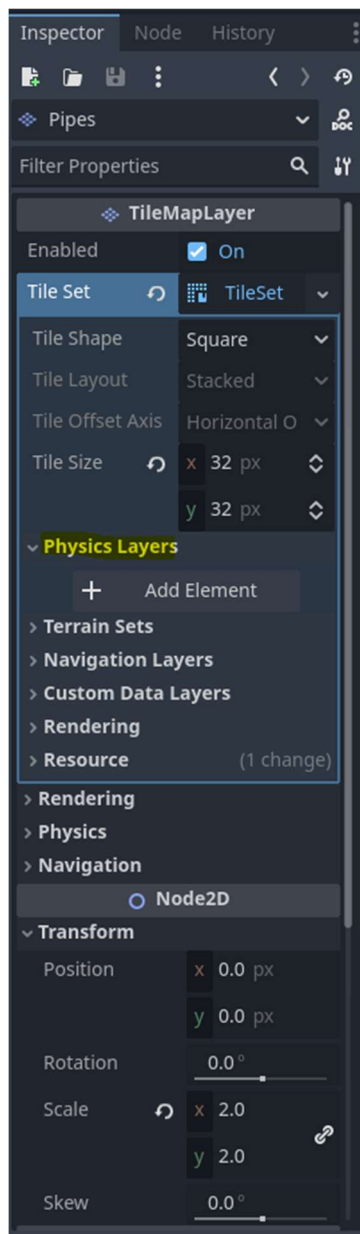
#### Steps:

In the “Level” Scene, select the TileMapLayer called “Pipes”

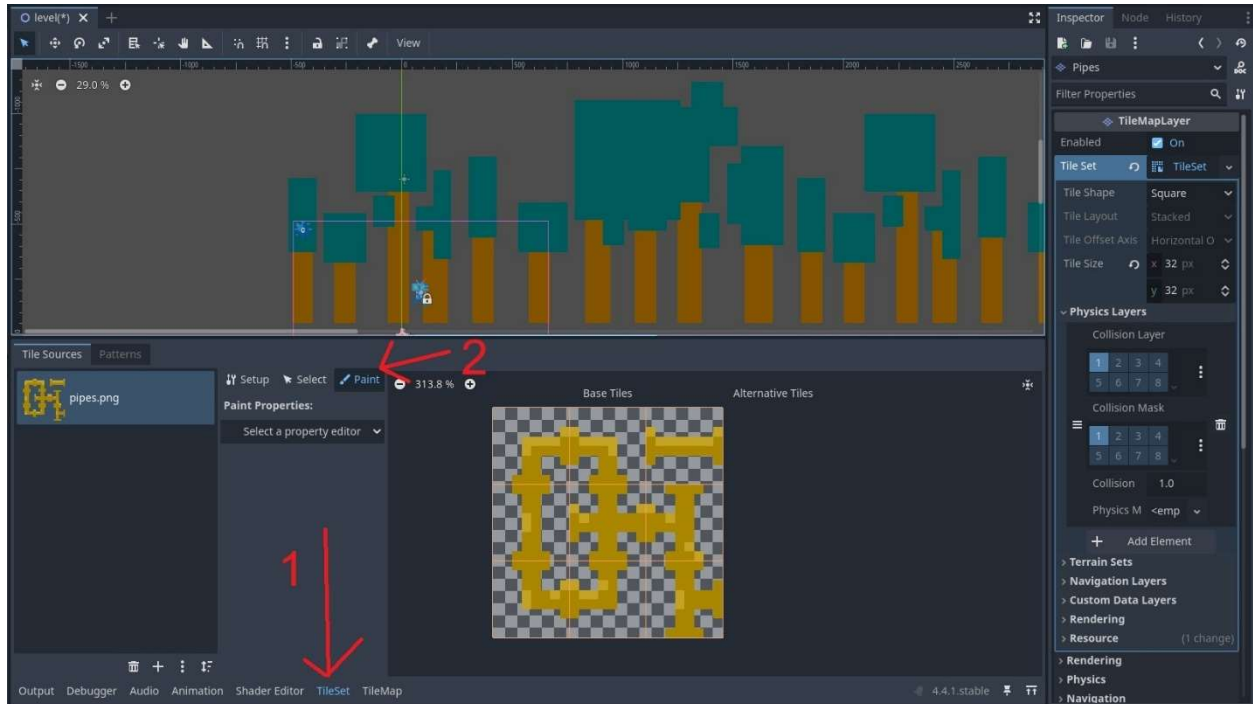
Click on the TileSet in the Inspector



Select “Add Element” under the “Physics Layers” tab

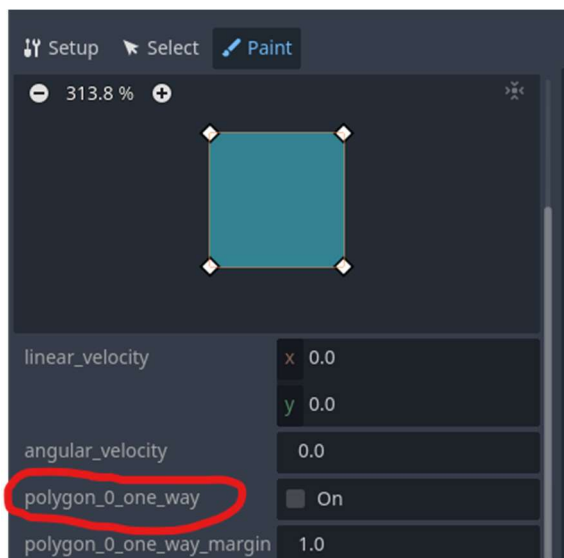


Now on the bottom click the “TileSet” tab and then “Paint”



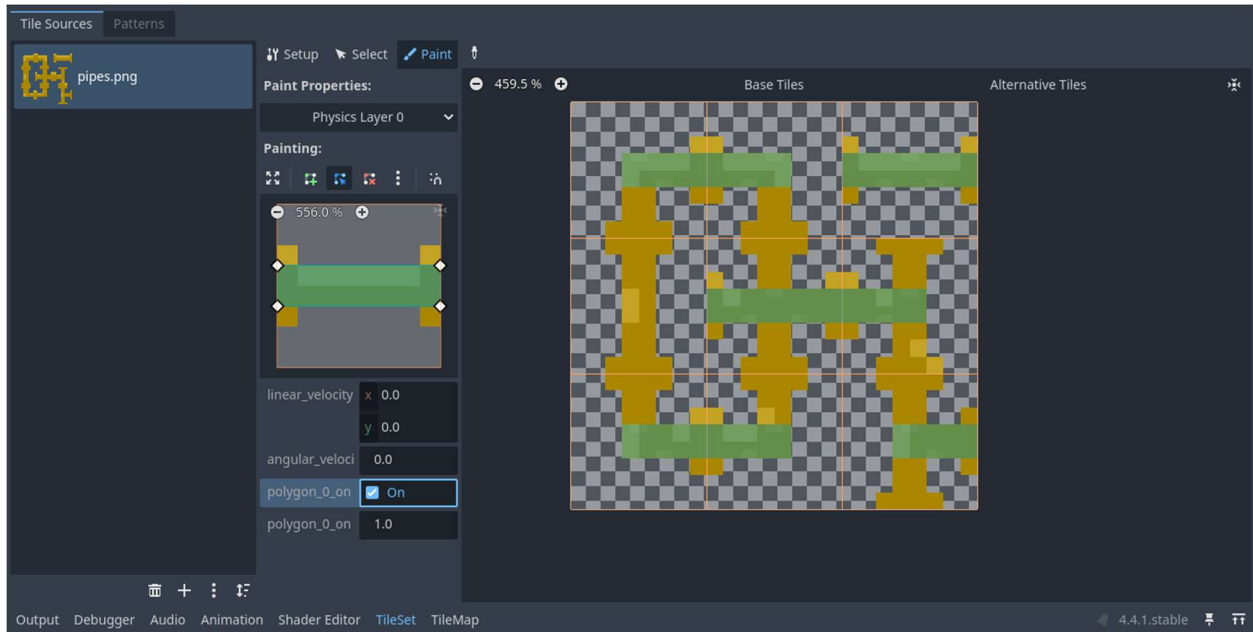
In the drop down menu of “Select a property editor” Select “Physics Layer 0”

Scroll down and check “Polygon\_0\_one\_way”



Now whenever you paint on the collision it will act as a one way collision

Paint the pipes accordingly by clicking on each tile and editing the collision shape in the small left window



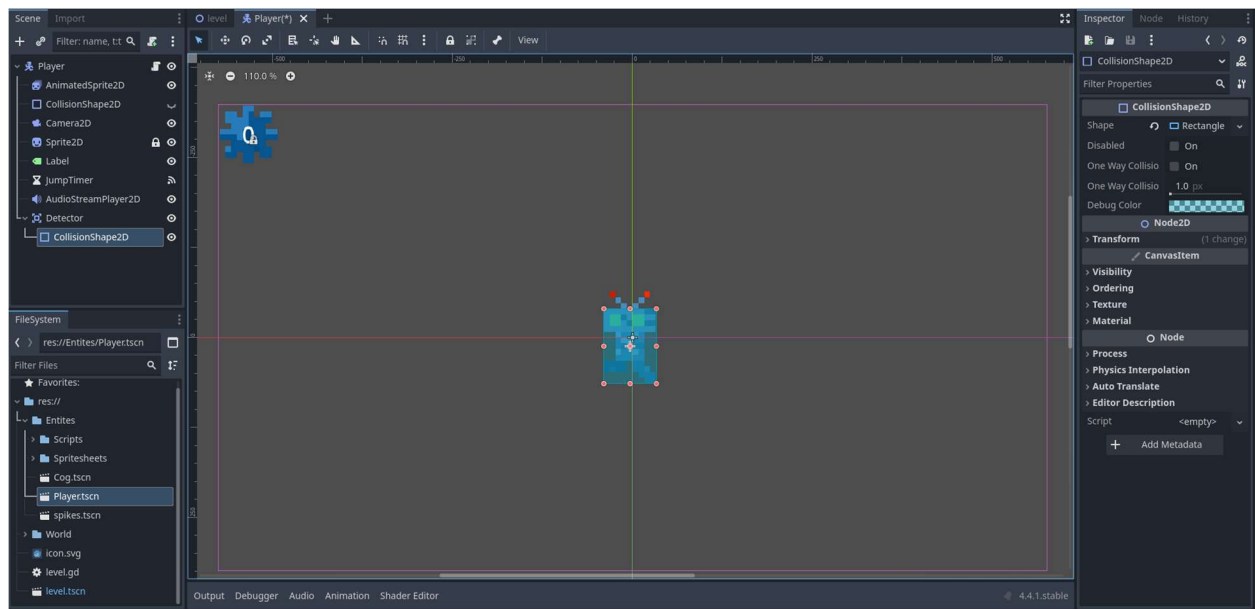
You now can place the pipes as one way platforms!

## Cog Pickups

### Steps:

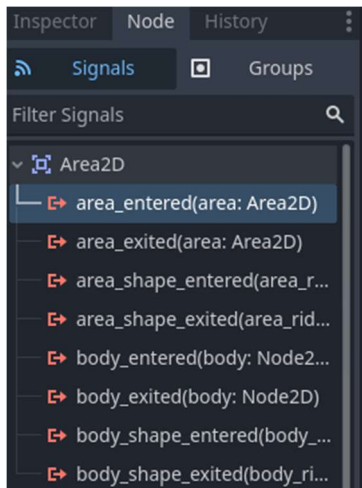
First Add an area 2D to the Player scene, call it “Detector” if you wish

Add a collision shape to the area2D, this defines the area that the area2D will detect

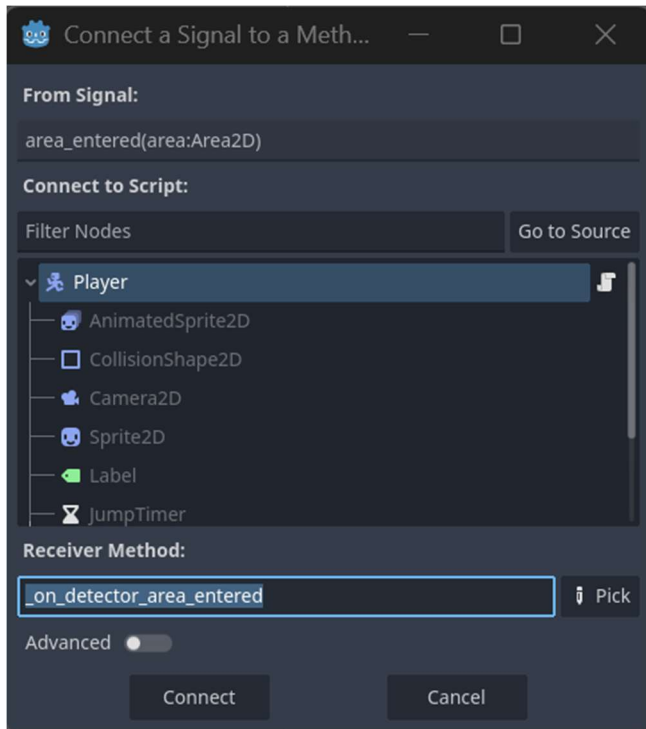


Select the area2D node and enter the “Node” Tab on the right side of the screen

Double click on the “area\_entered” signal



Connect to the player script



You should now see this function in the player script

```

51 >| # Checks if player is moving by comparing the player's velocity vector to the zero vector
52 >| if velocity == Vector2.ZERO:
53 >| >| $AnimatedSprite2D.play("default")
54 >| >| return
55 >|
56 >| # Set the direction the player is moving
57 >| if velocity.x < 0:
58 >| >| $AnimatedSprite2D.flip_h = true
59 >| elif velocity.x > 0:
60 >| >| $AnimatedSprite2D.flip_h = false
61 >|
62 >| # Checks if player is falling/Jumping
63 >| if abs(velocity.y) > 0:
64 >| >| if $AnimatedSprite2D.animation != "jump":
65 >| >| >| $AnimatedSprite2D.play("jump")
66 >| >| elif $AnimatedSprite2D.animation != "Walk":
67 >| >| $AnimatedSprite2D.play("Walk")
68
69
70
-> 71 >| func _on_jump_timer_timeout() -> void:
72 >| >| # When timer runs out coyote jump is no longer possible
73 >| >| coyote_jump = false
74
75
-> 76 >| func _on_detector_area_entered(area: Area2D) -> void:
77 >| >| pass # Replace with function body.
78

```

Implement this code:

```

func _on_area_2d_area_entered(area: Area2D) -> void:
>| var area_name = area.get_parent().name
>| if "cog" in area_name.to_lower():
>| >| points += 1

```

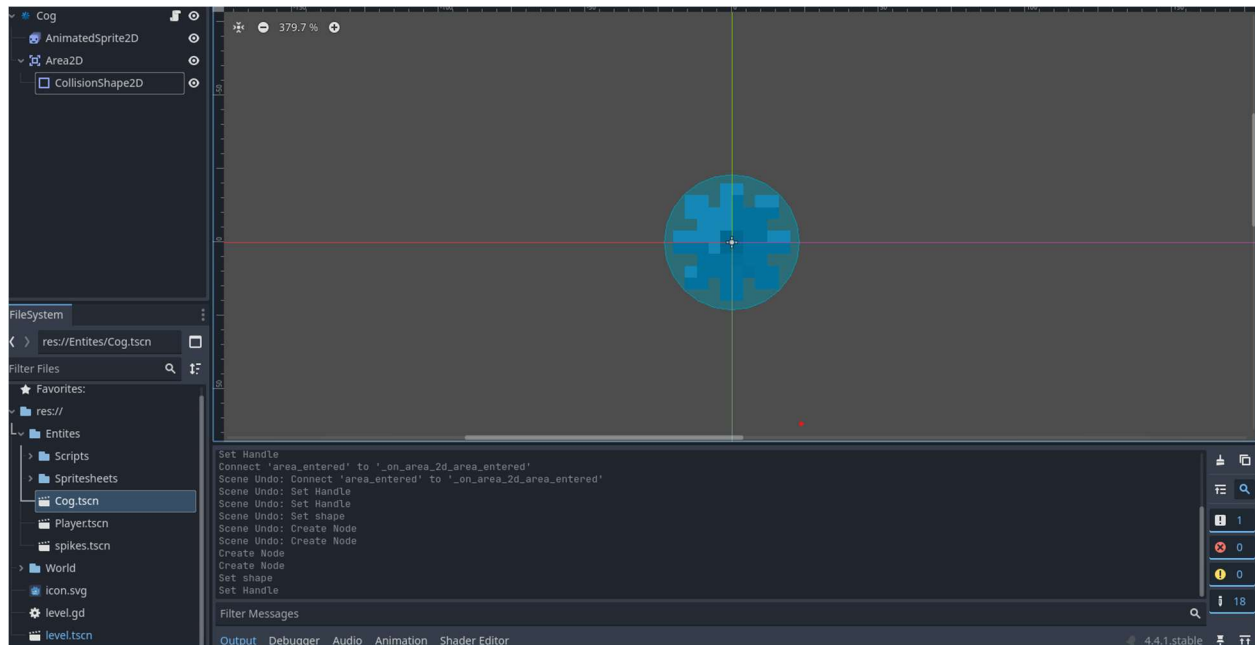
Explanation:

We will need to check whose area we are detecting, for this, we get the parent's name in the first line.

We check if the word "cog" is in the name, this wouldn't be best practices if you have other areas with the name "cog" in it but in this small game it works.

The reason we don't check if it's just called "cog" is because when you place multiple in the level, 3 for example, their names get changed to "Cog3".

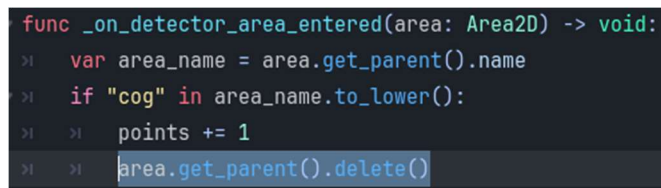
Next, we will need to add an area and collision shape to the Cog scene. Do not connect a signal to the script though



Go into the cog script and add this function:



Go back to the player script and have the player call that function when the area gets detected



This will get rid of the cog after it's "collected"



Now to display the points you have gotten to the label, add the last code:

```
func _on_detector_area_entered(area: Area2D) -> void:
    var area_name = area.get_parent().name
    if "cog" in area_name.to_lower():
        points += 1
        area.get_parent().delete()
        $Label.text = str(points)
```

And you're done!