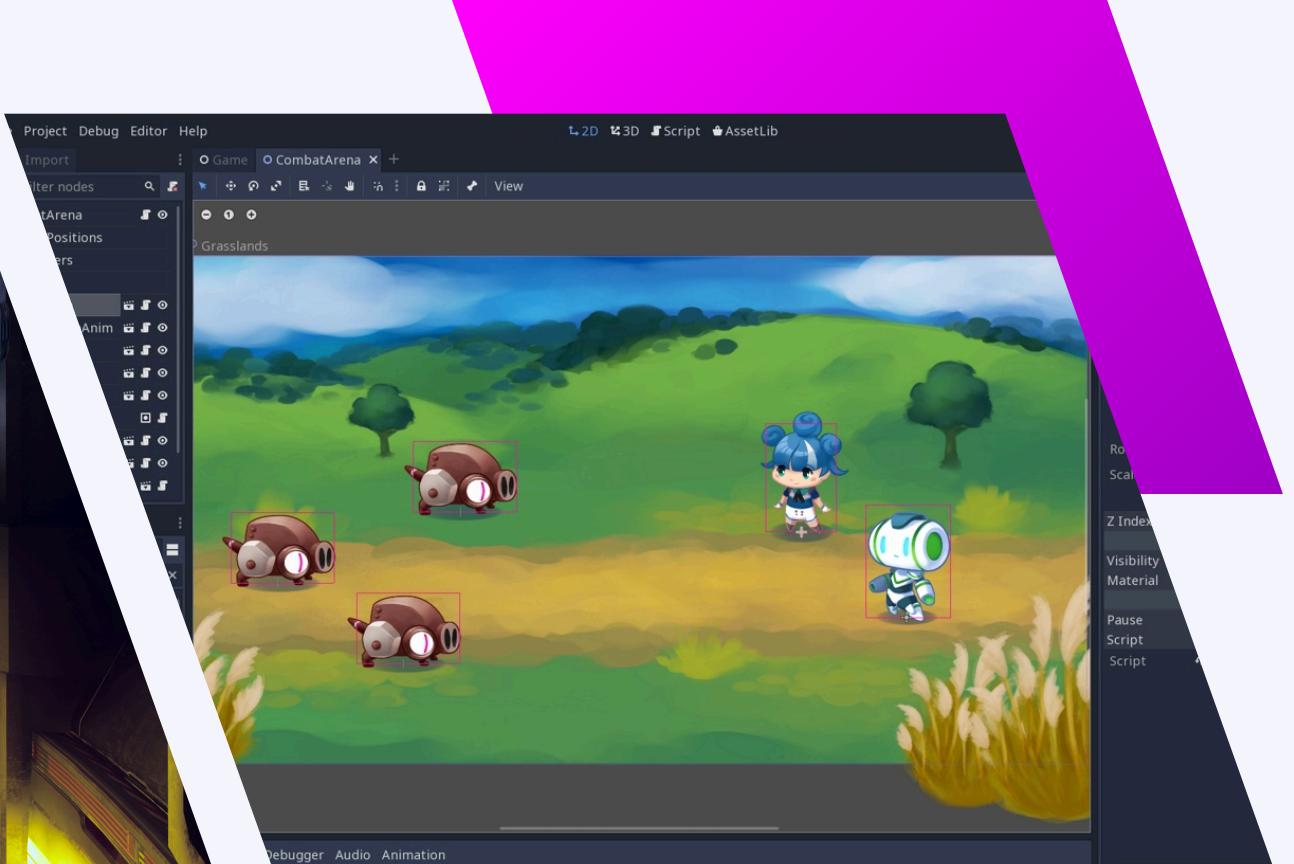


# GAME DEVELOPER

College of Creative Design and Entertainment Technology



การพัฒนาเกมด้วยโปรแกรม Godot Engine  
IG422 หัวข้อพิเศษด้านเทคโนโลยี

## ขั้นตอนที่ 1 การออกแบบระบบ Spawn Enemy

### SpawnEnemy คืออะไร?

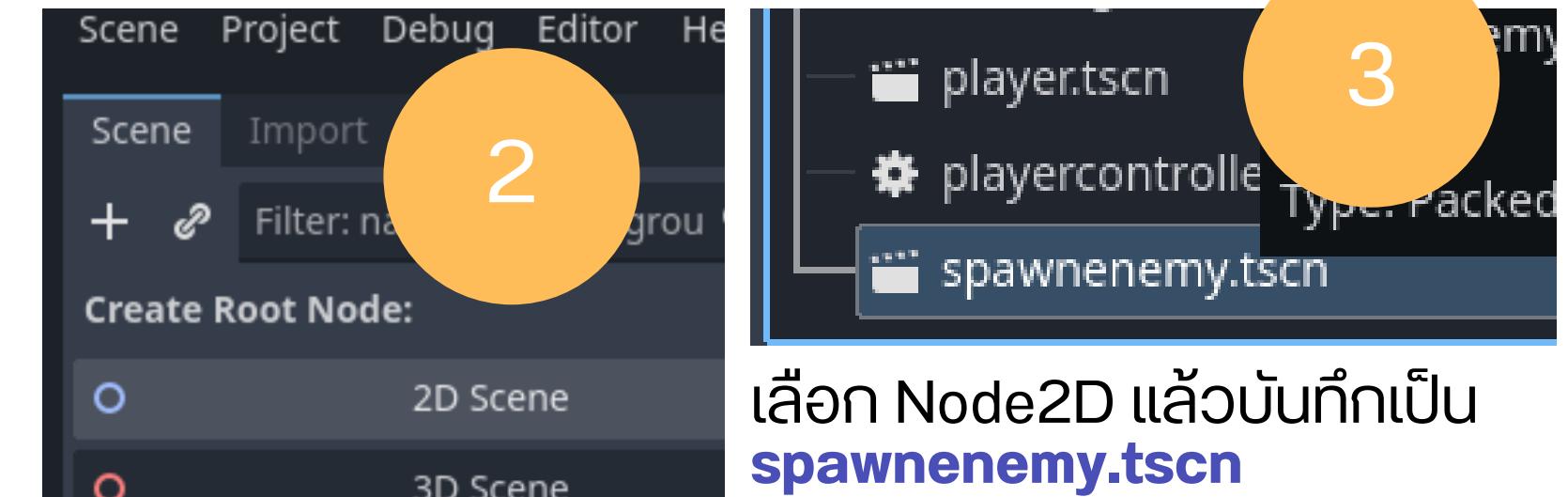
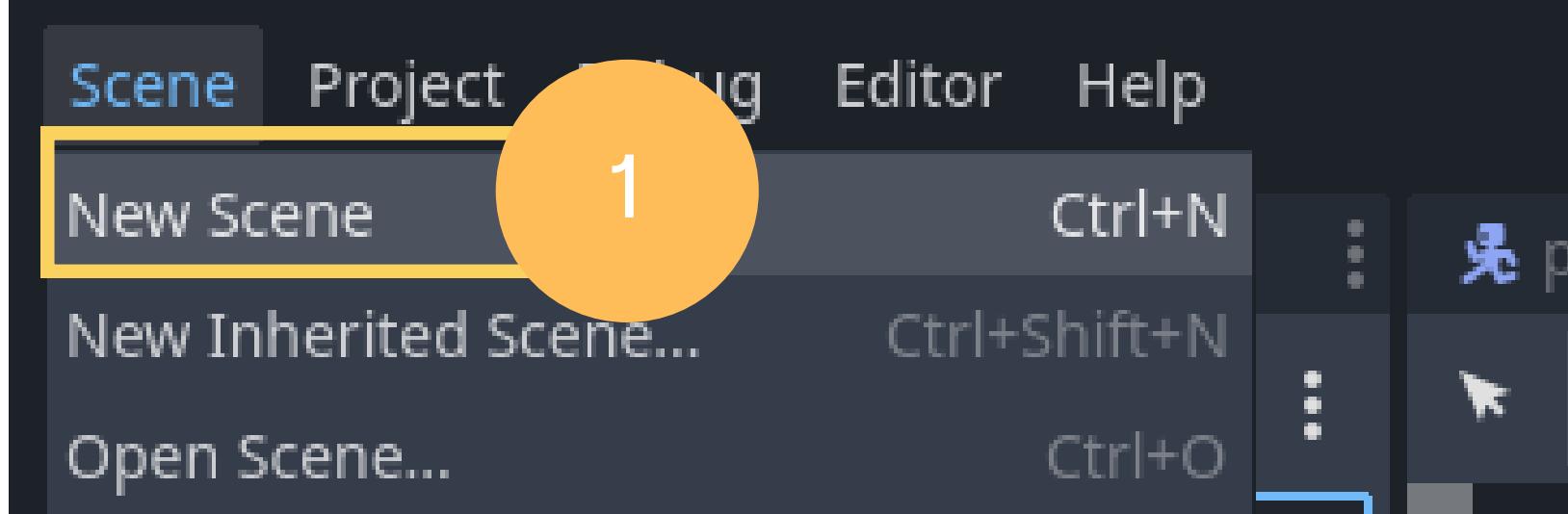
SpawnEnemy หมายถึงการ “สร้างศัตรูใหม่ขึ้นในเกม”

ระหว่างการเล่น เช่น:

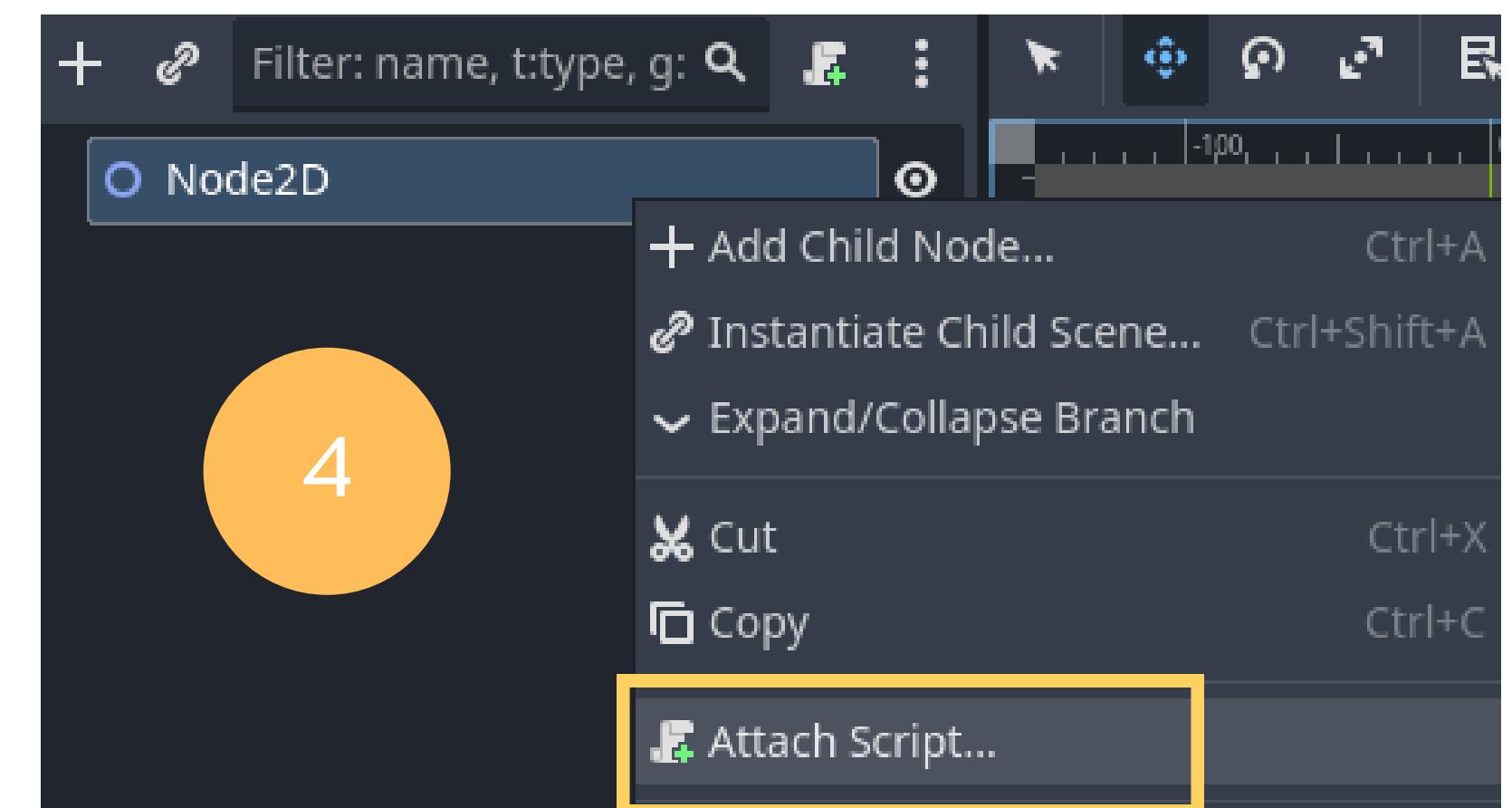
- สุ่มศัตรูเกิดตามเวลา (เช่น ทุก 3 วินาที)
- ศัตรูเกิดจากจุดที่กำหนดไว้ (เช่น ประตู, พื้นที่เกิด)
- สร้างศัตรูเมื่อผู้เล่นเดินผ่านพื้นที่หนึ่ง (Trigger Zone)

เป็นต้น เราสามารถเริ่มต้นพัฒนาได้กับที่โดยการสร้าง Scene ใหม่ขึ้นมาเป็น Node2D และบันทึกว่า **spawnenemy.tscn**

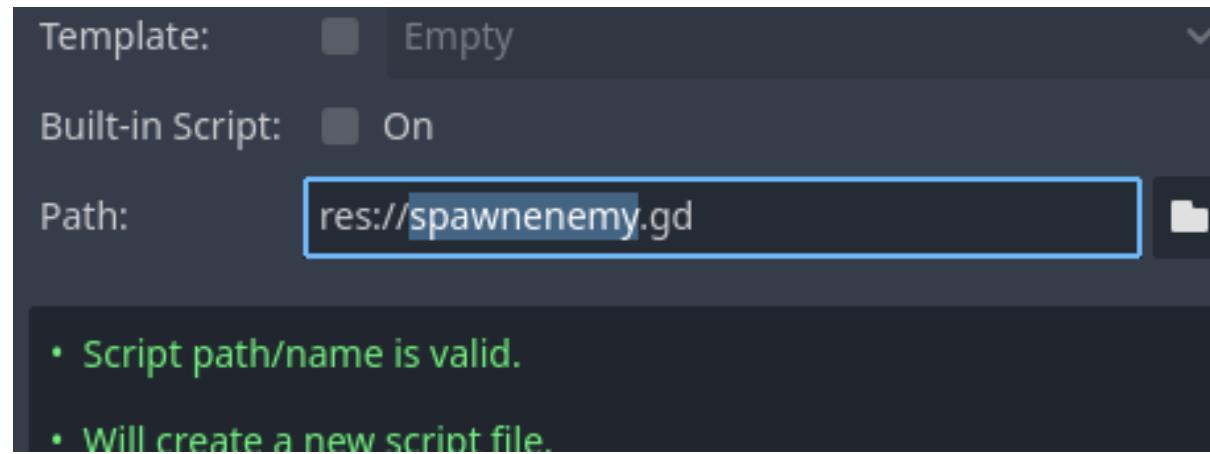
(\*) game.tscn - Game - Godot Engine



เลือก Node2D และบันทึกเป็น  
**spawnenemy.tscn**



คลิกที่ Node2D และเลือก **Attach Script...**



## สร้าง Script ว่า **spawnenemy.gd**

```
extends Node2D

@export var isSpawn := false
var enemy_prefab = preload("res://enemy.tscn")

func _ready():
    set_process(true)
    pass
```

สร้างตัวแปร Boolean ชื่อ `isSpawn` ที่สามารถเปิด/ปิดได้จาก Inspector ใช้สำหรับ “กำหนดว่าจะให้ Spawn คัตติวใหม่” ( เช่น เป็นจุดเกิดหรือไม่ )

โหลดไฟล์คัตติวที่เป็น Scene ไว้ล่วงหน้า (`enemy.tscn`) เพื่อใช้ในภายหลังตอนเรียก `instantiate()`

```
func _process(delta):
    if !isSpawn:
        isSpawn = true
        var timer_count = randf_range(1.0,5.0)
        await get_tree().create_timer(timer_count).timeout
        spawning_enemy()
        isSpawn = false
    pass
```

### การทำงานในฟังก์ชัน **\_process()** คือ

ถ้า `isSpawn` ยังเป็น `false` (ยังไม่ได้กำลัง `Spawn`) ป้องกันไปให้ฟังก์ชัน `spawn` ทำงาน หลายครั้งซ้อนกัน

ตั้งสถานะว่าตอนนี้ “กำลัง `Spawn` อยู่” เพื่อไม่ให้เข้าเงื่อนไขนี้ซ้ำๆ ระหว่างรอ

สุ่มตัวเลข float ระหว่าง 1.0 ถึง 5.0 วินาที ใช้เป็นเวลาอ包包สุ่มก่อนที่จะ `Spawn` คัตติว

รอเวลา `timer_count` วินาที ทำให้ฟังก์ชันหยุดชั่วคราวก่อนจะไปบรรทัดถัดไป (non-blocking)

```

1  extends Node2D
2
3  @export var isSpawn := false
4  var enemy_prefab = preload("res://enemy.tscn")
5
6  func _ready():
7      set_process(true)
8      pass
9
10 func _process(delta):
11     if !isSpawn:
12         isSpawn = true
13         var timer_count = randf_range(1.0, 5.0)
14         await get_tree().create_timer(timer_count).timeout
15         spawning_enemy()
16         isSpawn = false
17     pass

```

เอาไว้ตรวจสอบการเยื่องบรรกัด และการพิมพ์ฟังก์ชัน ต่อไปเราจะทำการ Spawn ศัตรูด้วยฟังก์ชันหลักคือ:

```

func spawning_enemy():
var enemy_instance = enemy_prefab.instantiate()
var player_node = get_node("../Player")
if player_node:
    enemy_instance.player_path = player_node.get_path()
    enemy_instance.global_position = get_random_position()
    get_parent().add_child(enemy_instance)

```

ฟังก์ชัน spawning\_enemy() ใช้สำหรับสร้างศัตรูใหม่จาก enemy\_prefab แล้วกำหนดตำแหน่งแบบสุ่มด้วย get\_random\_position() จากนั้นเพิ่มเข้าสู่จากโดยใช้ add\_child() ใต้ Node แม่. ทำให้ศัตรูเกิดในตำแหน่งที่แตกต่างกันได้ในแต่ละครั้ง

```

func get_random_position():
var random_x = randf_range(-20, 100)
var random_y = randf_range(-12, 80)
return Vector2(random_x, random_y)

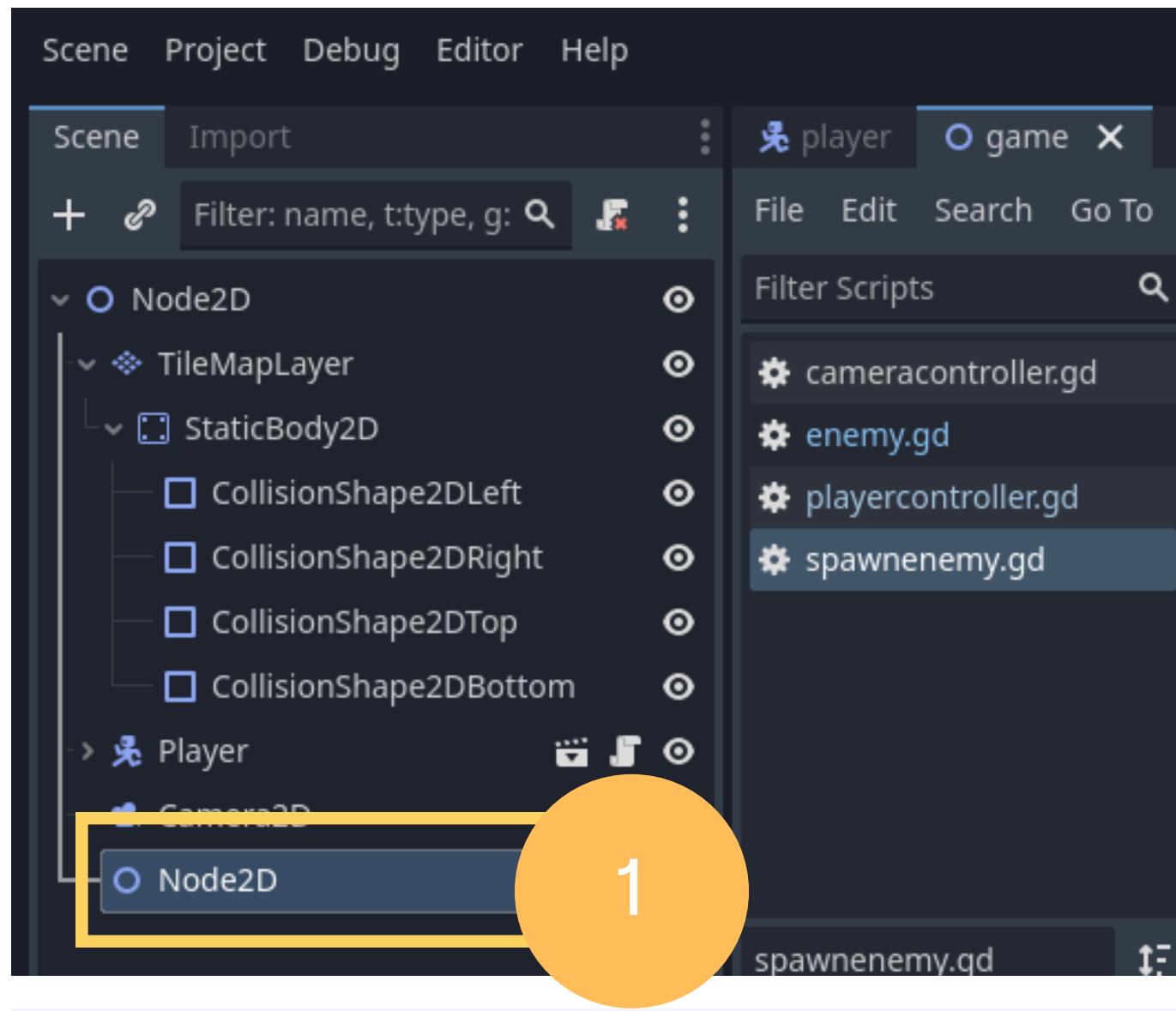
```

ฟังก์ชัน spawning\_enemy() ใช้สำหรับสร้างศัตรูใหม่จาก enemy\_prefab แล้วกำหนดตำแหน่งแบบสุ่มด้วย get\_random\_position() จากนั้นเพิ่มเข้าสู่จากโดยใช้ add\_child() ใต้ Node แม่. ทำให้ศัตรูเกิดในตำแหน่งที่แตกต่างกันได้ในแต่ละครั้ง

```

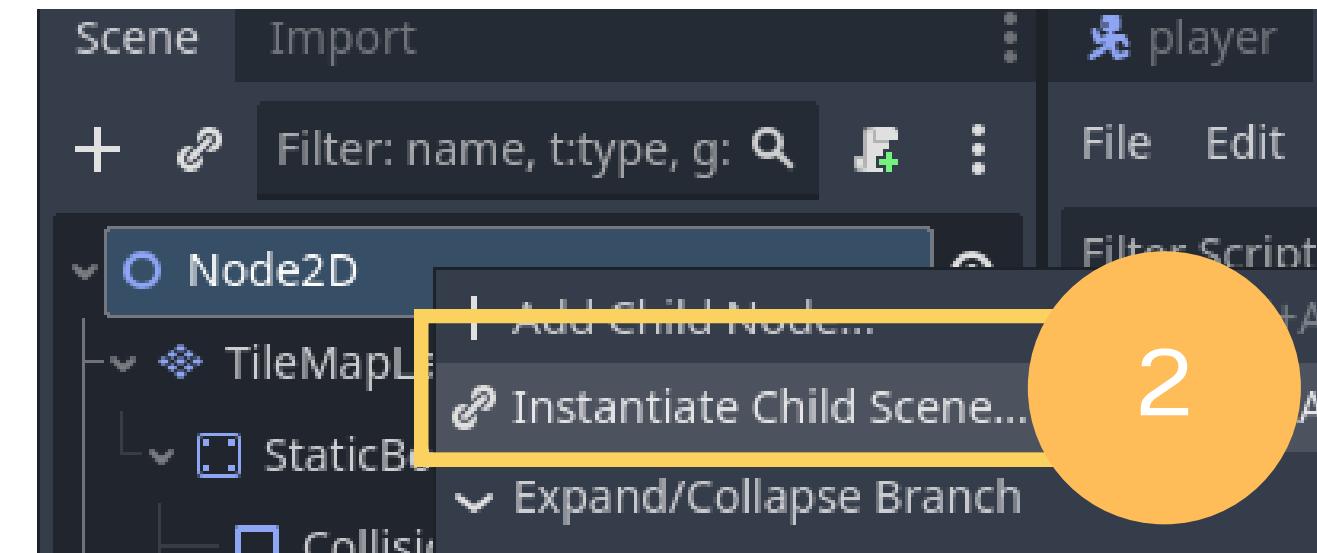
18
19  func spawning_enemy():
20      var enemy_instance = enemy_prefab.instantiate()
21      var player_node = get_node("../Player")
22      if player_node:
23          enemy_instance.player_path = player_node.get_path()
24          enemy_instance.global_position = get_random_position()
25          get_parent().add_child(enemy_instance)
26
27  func get_random_position():
28      var random_x = randf_range(-20, 100)
29      var random_y = randf_range(-12, 80)
30      return Vector2(random_x, random_y)
31

```

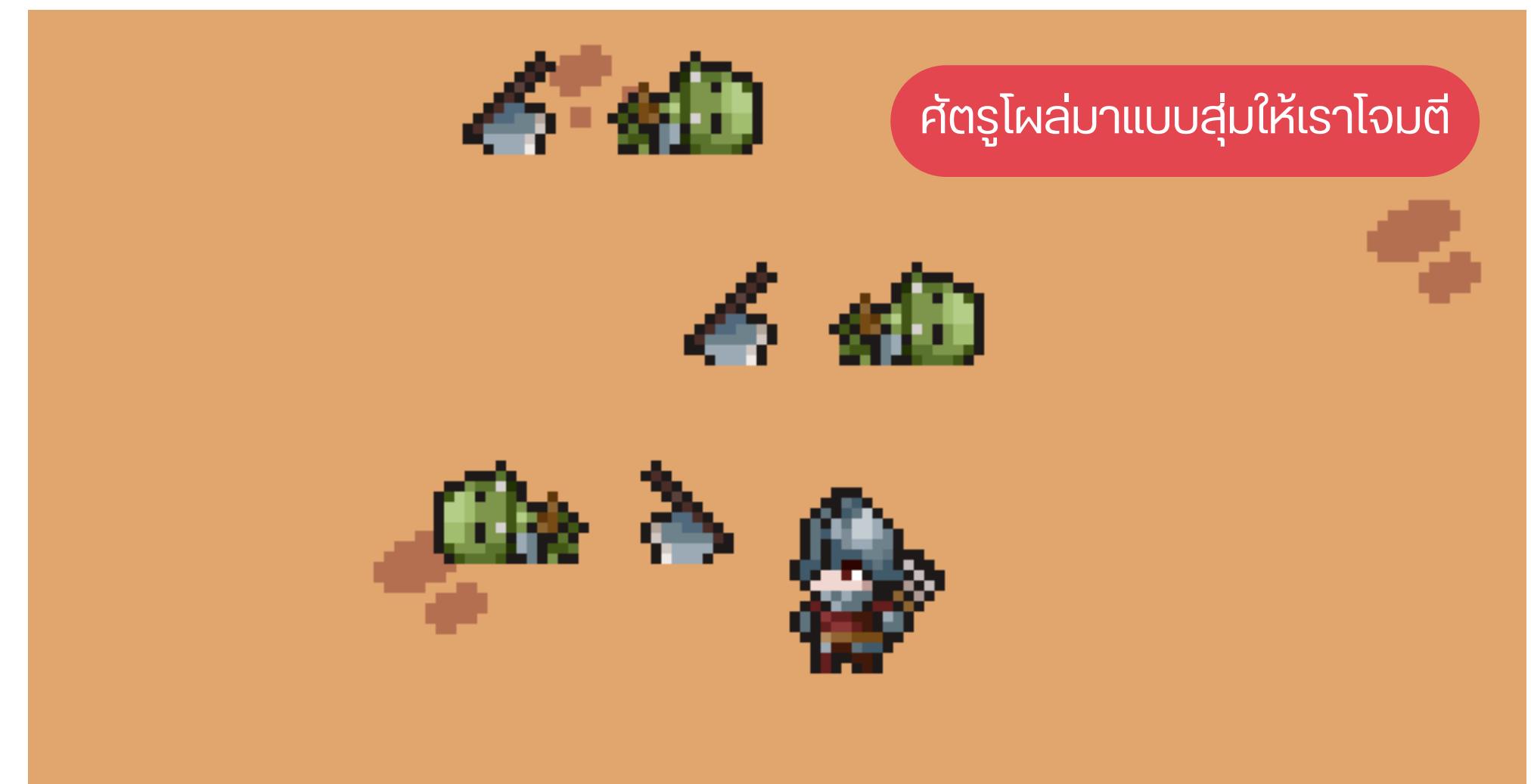


## ขั้นตอนที่ 2 เพิ่ม Spawn System ใน scene

ทำการลบ enemy ออกจาก Scene game.tscn แล้ว  
ให้เพิ่ม Instantiate Child Scene เลือก  
spawnenemy.tscn เข้าไปแทน เราจะได้ Node2D มา  
ปรากฏขึ้น



ทำการเพิ่ม Scene ชื่อ  
spawnenemy.tscn เข้าไป  
ใน Scene game.tscn แล้ว  
ทำการลบ Enemy ออกไป  
จาก Scene

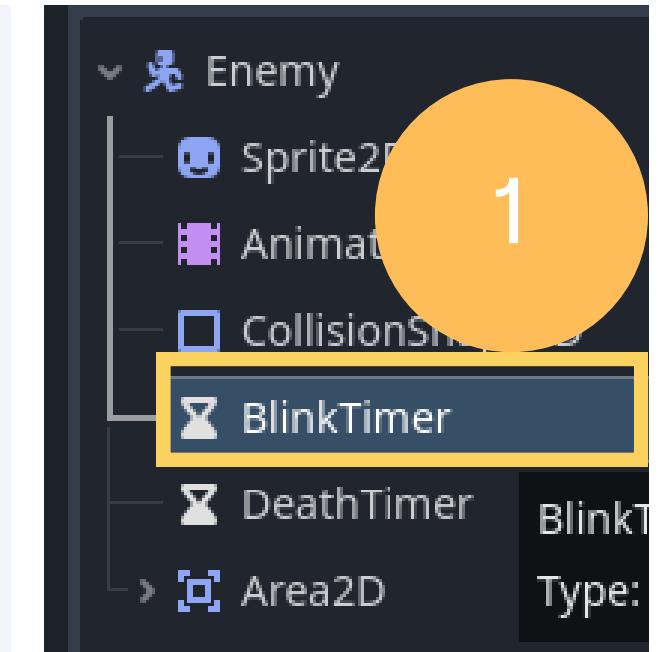


ปรับฟังก์ชันการตาย die() ใหม่ดังนี้

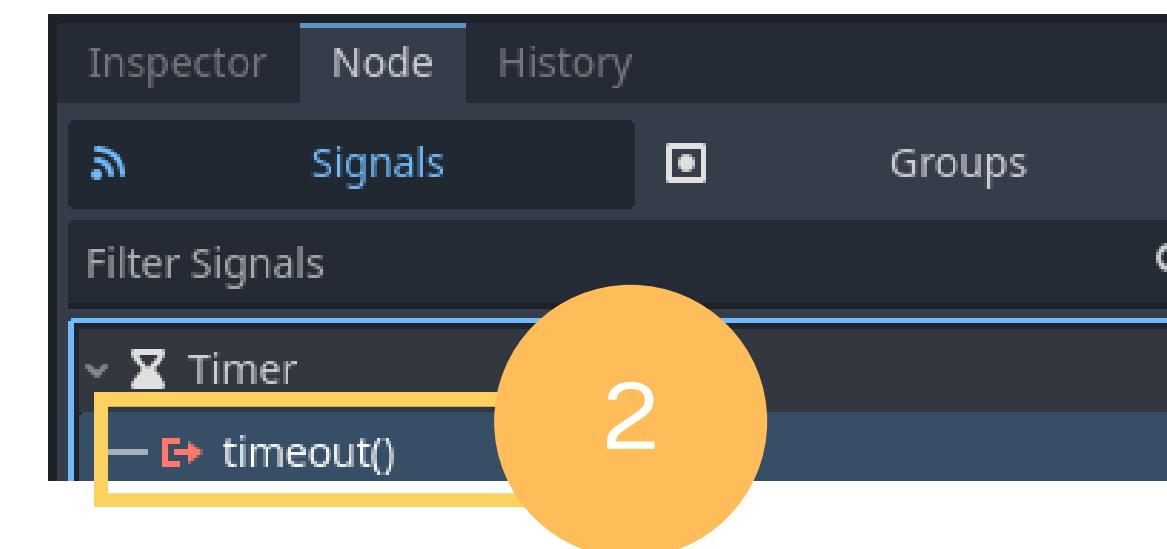
```
func die():
    is_dead = true
    velocity = Vector2.ZERO
    animation_player.play("Death")

    # ปิดการชนทั้งหมด
    collisionShape.disabled = true
    collisionShapeArea.disabled = true
    collision_layer = 0
    collision_mask = 0

    death_timer.start(0.8)
```



เลือก Node BlinkTimer  
ไปที่แบบ "Node" > "Signals"  
ดับเบิลคลิกที่ timeout() → เชื่อมไปที่ Node Enemy  
เลือกฟังก์ชัน: \_on\_BlinkTimer\_timeout

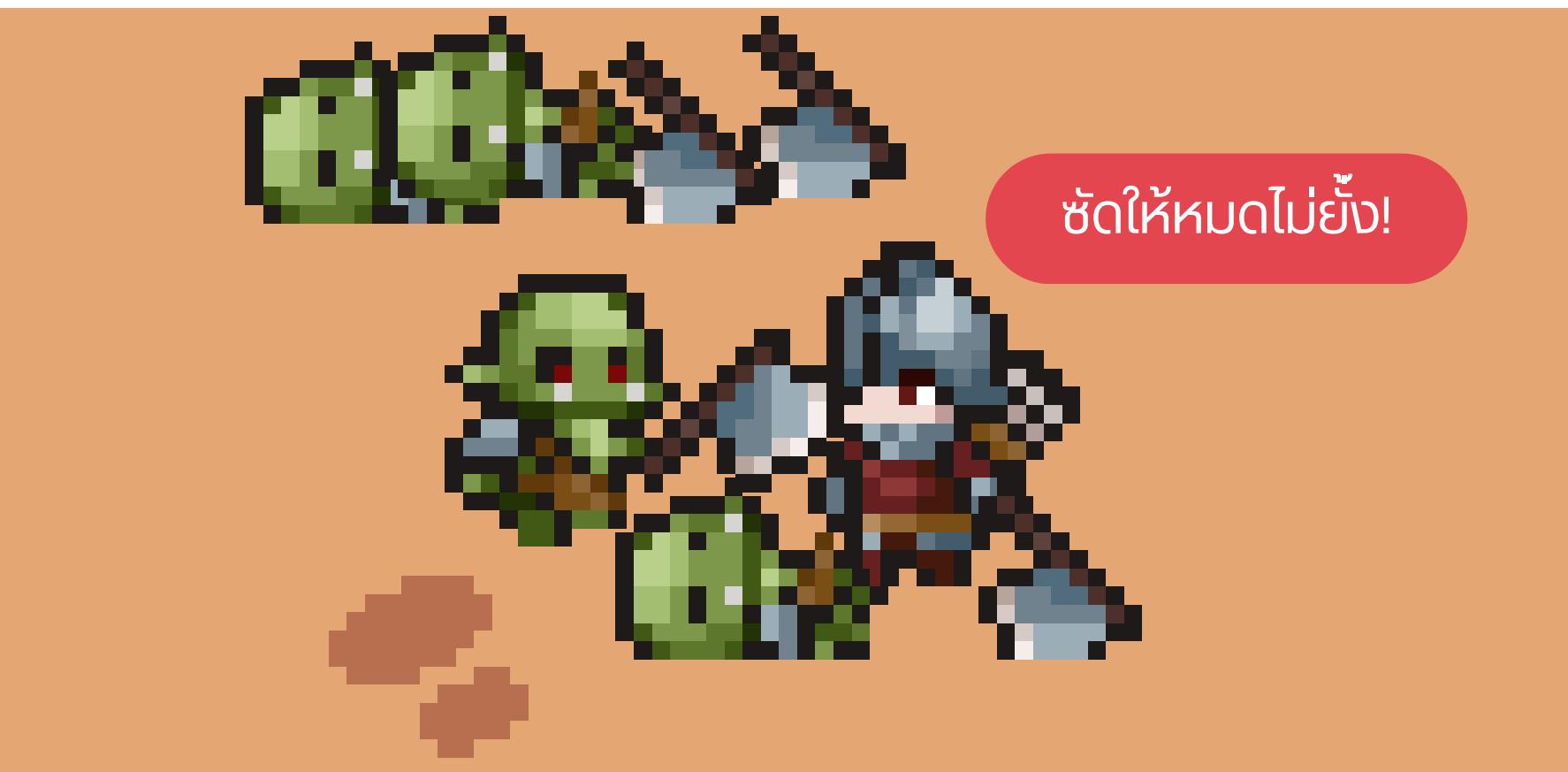
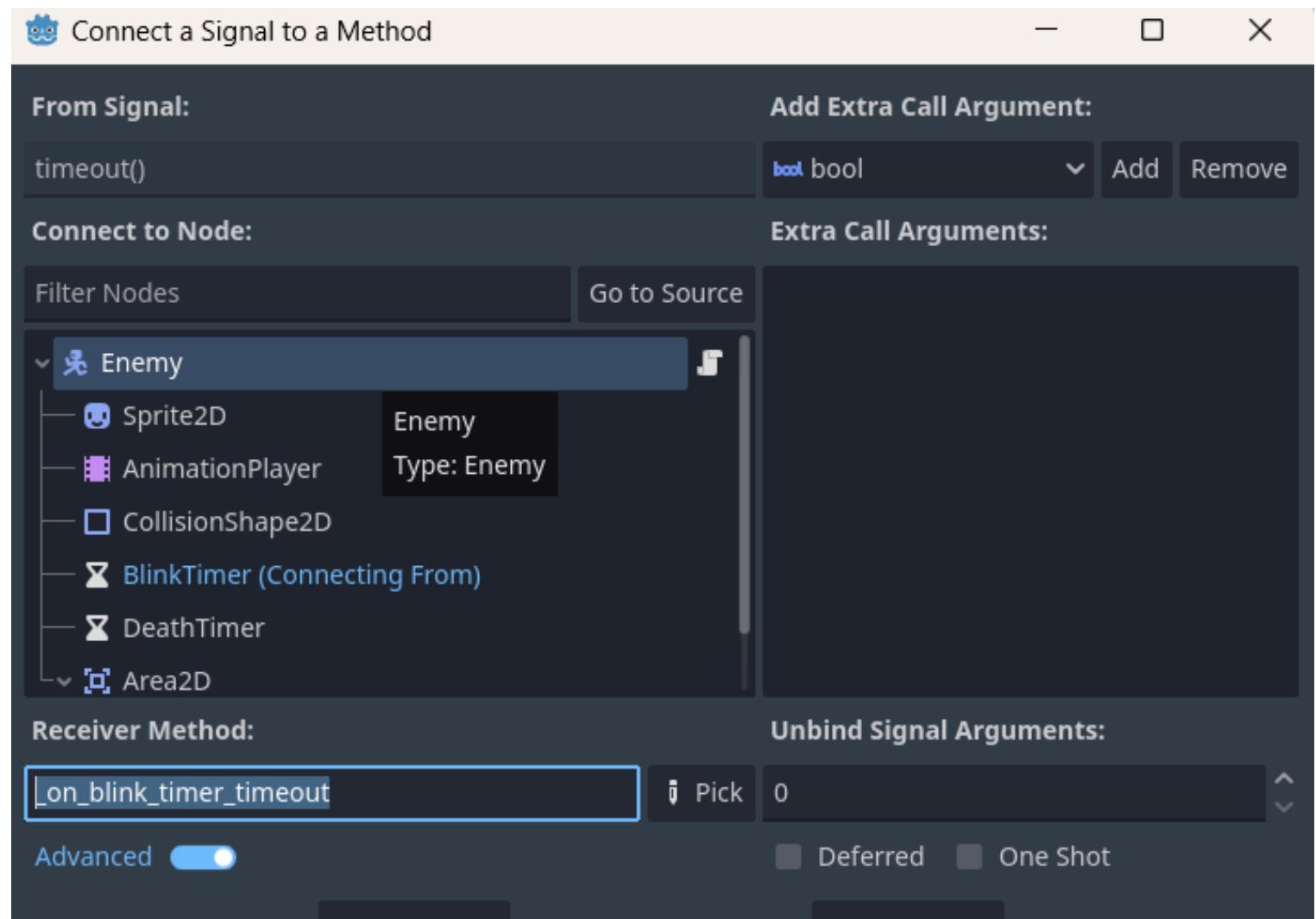


### ขั้นตอนที่ 3 ปรับ CollisionShape2D ของ Enemy เมื่อโดนผู้เล่นโจมตี

เมื่อผู้เล่นโจมตีศัตรู แล้วยังมี CollisionShape2D อยู่ทำให้ผู้เล่นเดินชน ศพของศัตรู ดังนั้นให้เราปรับเพิ่มเพื่อให้ผู้เล่นเดินผ่านศพของศัตรูได้ตั้งปรับ **enemy.gd**

```
@onready var collisionShape: CollisionShape2D = $CollisionShape2D
@onready var collisionShapeArea: CollisionShape2D = $Area2D/CollisionShape2D
```

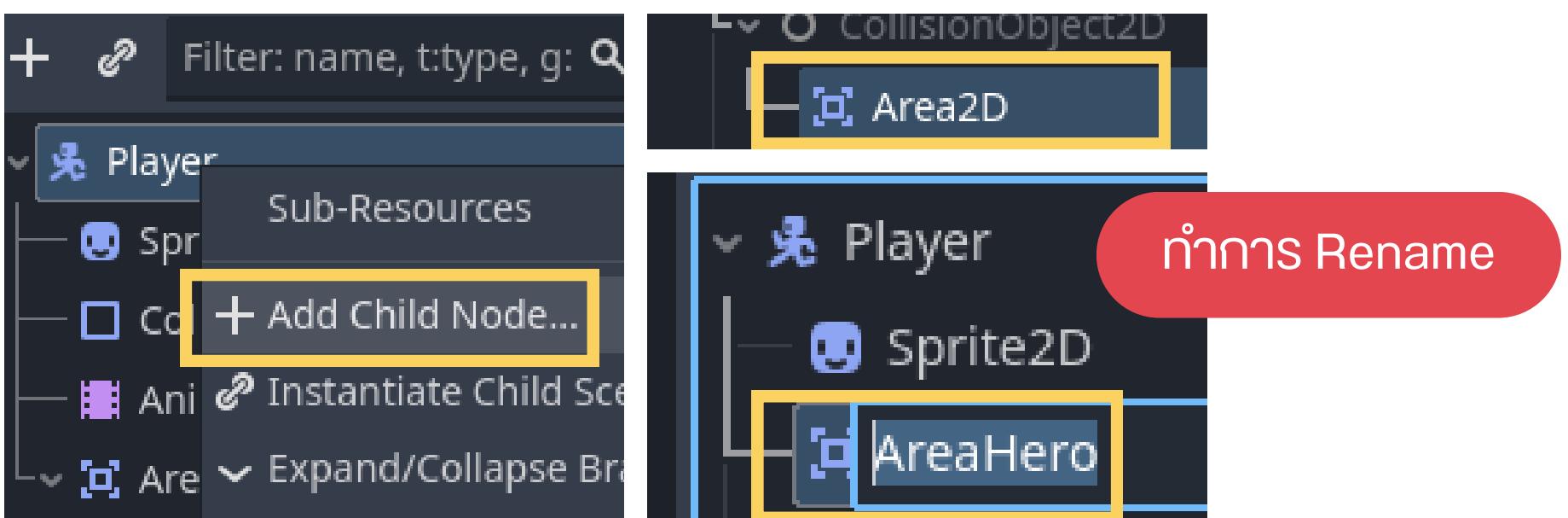
```
8  @export var knockback_distance: float = 5.0
9  @onready var sprite: Sprite2D = $Sprite2D
10 @onready var animation_player: AnimationPlayer = $AnimationPlayer
11 @onready var blink_timer: Timer = $BlinkTimer
12 @onready var death_timer: Timer = $DeathTimer
13
14 @onready var collisionShape: CollisionShape2D = $CollisionShape2D
15 @onready var collisionShapeArea: CollisionShape2D = $Area2D/CollisionShape2D
16
```

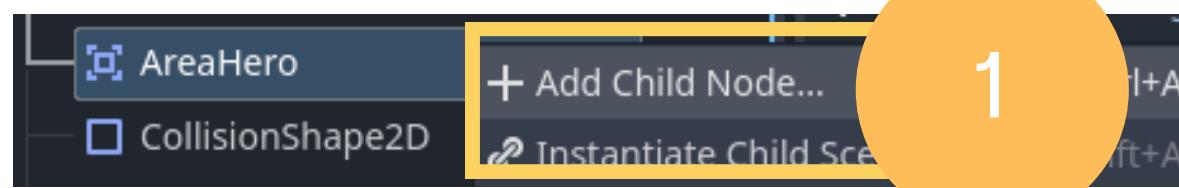


ตอนนี้ศัตรูของเรามีอุปกรณ์เป็นชิพ Player สามารถเดินผ่าน และเดินซ้อนได้แล้วทำให้แผนที่สามารถเดินไปมาได้ปกติ จากตอนแรกที่เดินไปมาไม่ได้

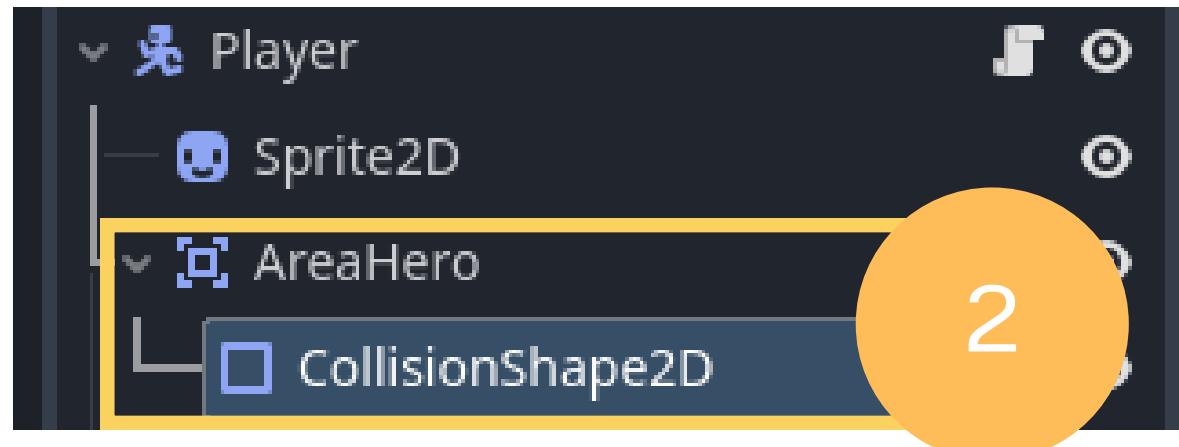
## ขั้นตอนที่ 4 เพิ่มเงื่อนไขพลังให้ผู้เล่น

ผู้เล่นเป็นออมต์เกมจะไม่มีความน่าสนใจและน่าก้าว่าย ดังนั้นต้องมีการเพิ่มเงื่อนไขให้ Player ของเรามา เพิ่ม **Area2D** ของ Player บน Scene Panel ให้มีขึ้นมาเป็นอีกตัวไม่ใช่ของ Hitbox และทำการ Rename Area2D เป็น **AreaHero**

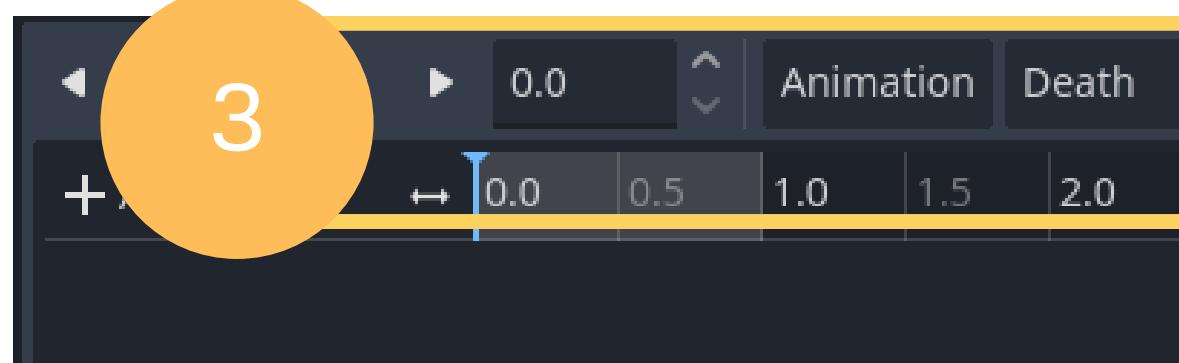




เพิ่ม **+Add Child Node** ส่วนของ **AreaHero**  
เพิ่ม CollisionShape2D เข้าไป เป็น Rectangle

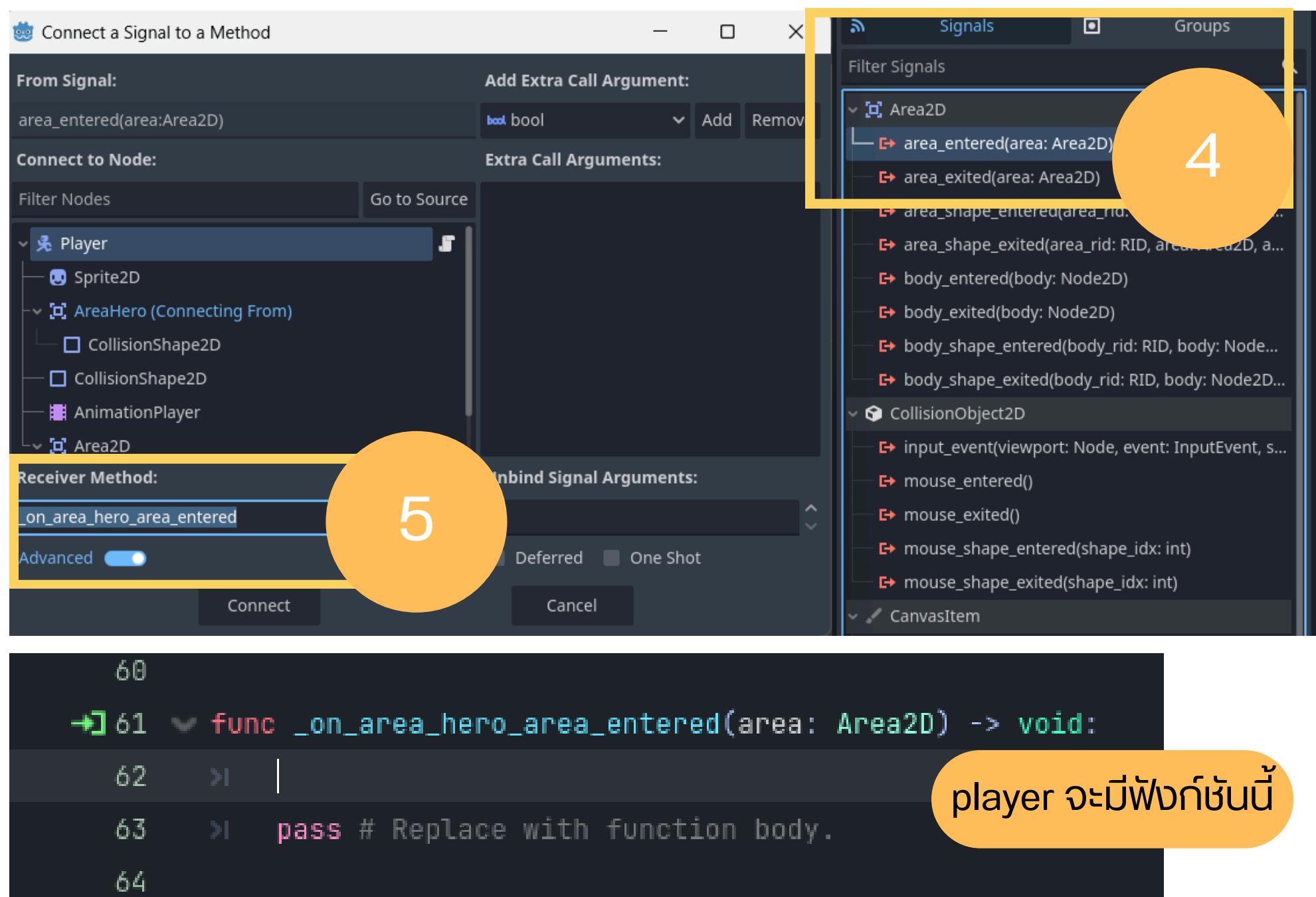


เปิด Player ขึ้นมาหลังจากนั้นคลิกที่  
**AnimationPlayer** สร้าง Animation ใหม่ว่า  
“Death” ของ Player



เปิดไฟล์ **playercontroller.gd** ขึ้นมาหลังจากนั้นประกาศตัวแปร

```
@export var hp = 5
var is_dead = false
```



```

if is_dead:
    return

if area.is_in_group("Enemy"):
    hp -= 1
    if hp <= 0:
        die()
    pass # Replace with function body.

```

เช็คให้ Area ของ player ว่าดีไซบ์ group ส่วน **Enemy** จะให้ลดพลังด้า **hp <= 0** ให้ไปฟังก์ชัน **die()**

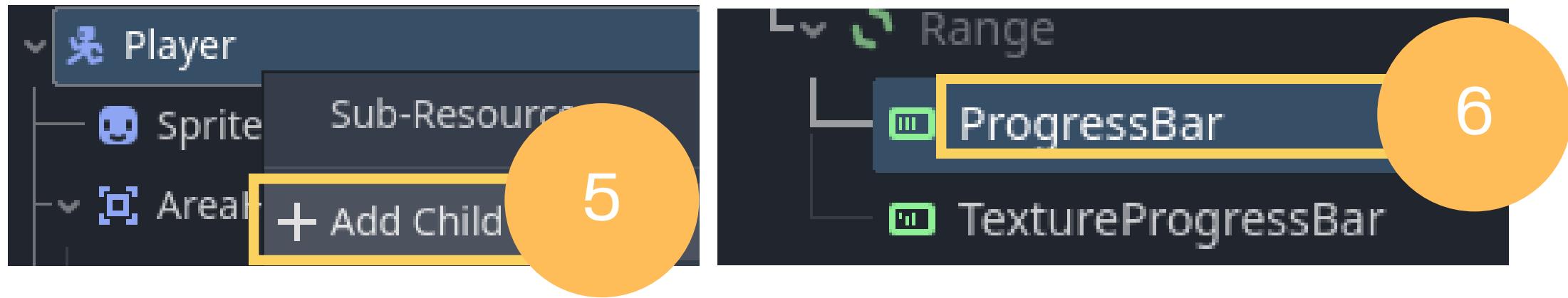
```

func die():
    is_dead = true
    animation_player.play("Death")
    velocity = Vector2.ZERO
    set_physics_process(false)
    # หยุดการประมวลผลการเคลื่อนไหว
    hitbox.disabled = true

```

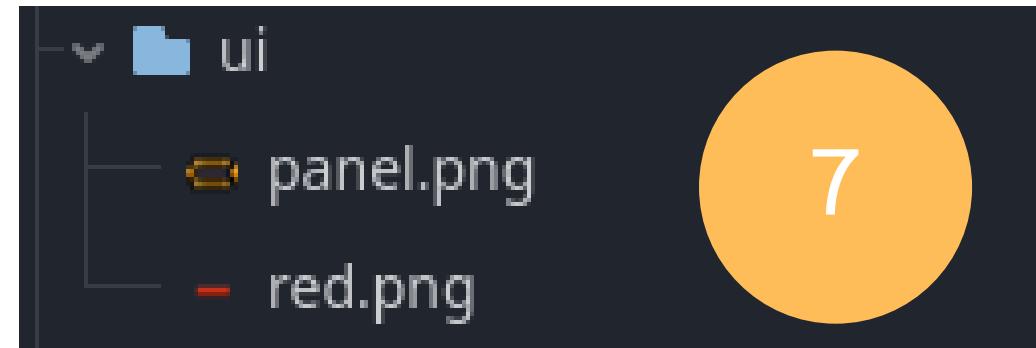
## ขั้นตอนที่ 5 เพิ่มส่วนของ UI

ผู้เล่นเราจะมี **ProgressBar** มาวางแผนเพื่อบำนวน hp ของเรามาเป็น Child ของ Player ให้ปรากฏ

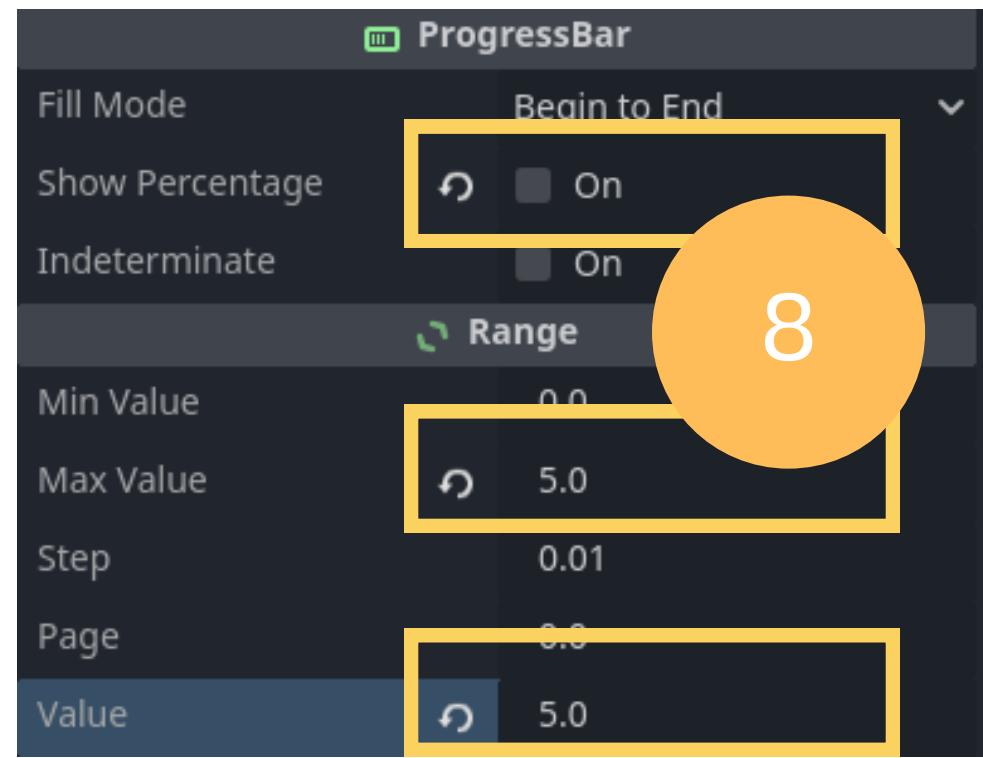


ดาวน์โหลด **Assets** ส่วนของ UI ได้ที่นี่:

[https://drive.google.com/file/d/1EHBj5\\_oRXusHBmrqvj4hoghySiBL\\_u0w/view?usp=drive\\_link](https://drive.google.com/file/d/1EHBj5_oRXusHBmrqvj4hoghySiBL_u0w/view?usp=drive_link)

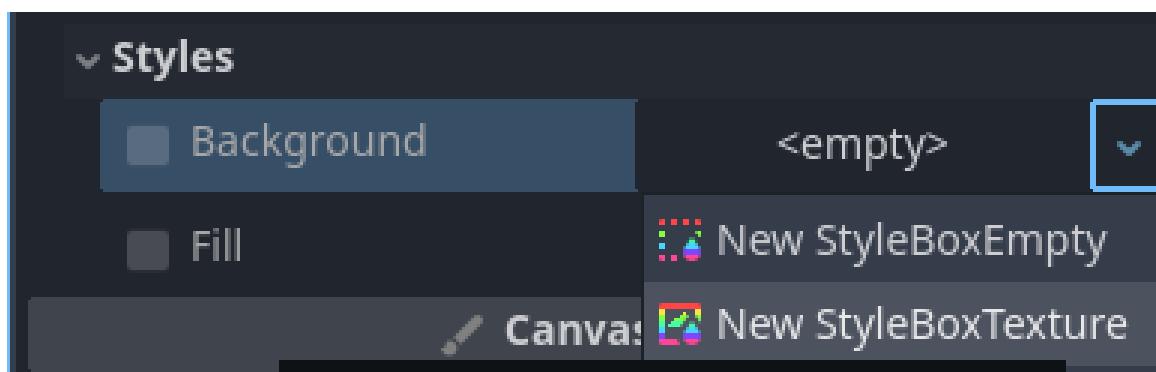
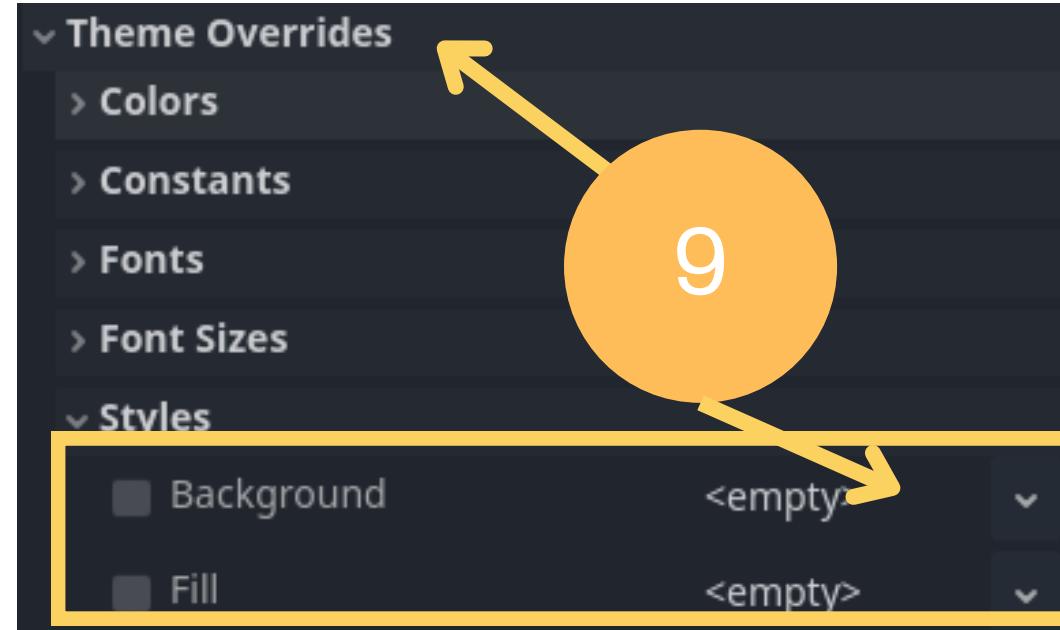


เอา **panel.png** และ **red.png** เป็นไฟล์ png ใส่ลงไปใน Folder ui (หากไม่มีให้สร้างไฟล์ผ่าน **FileSystem**)

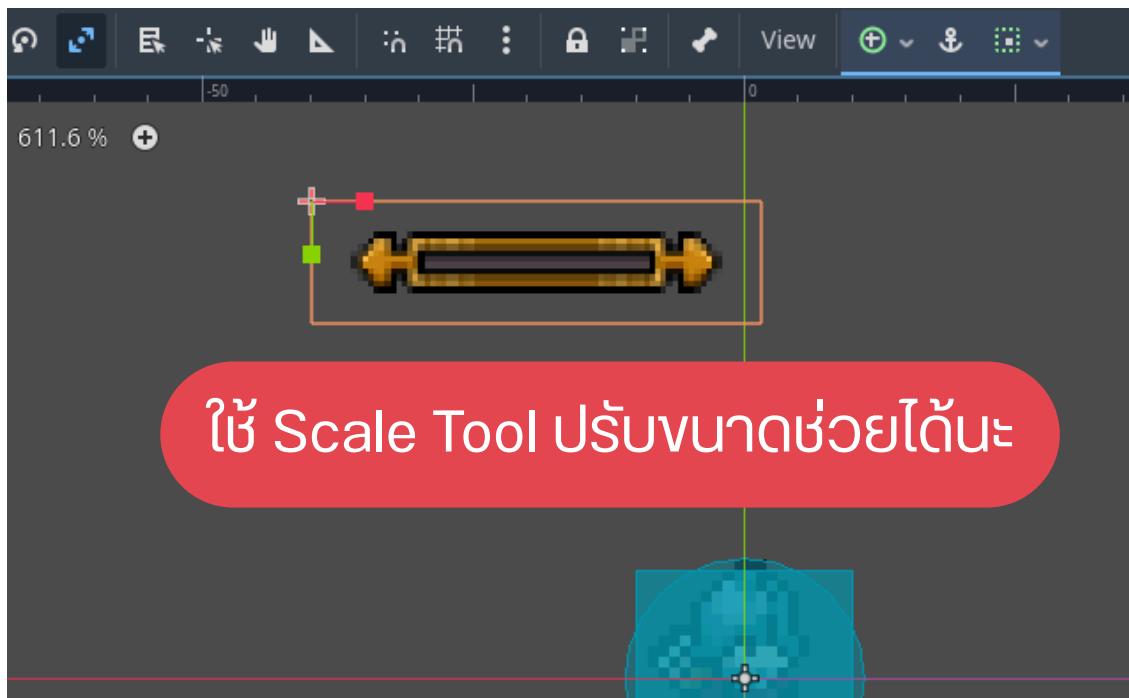
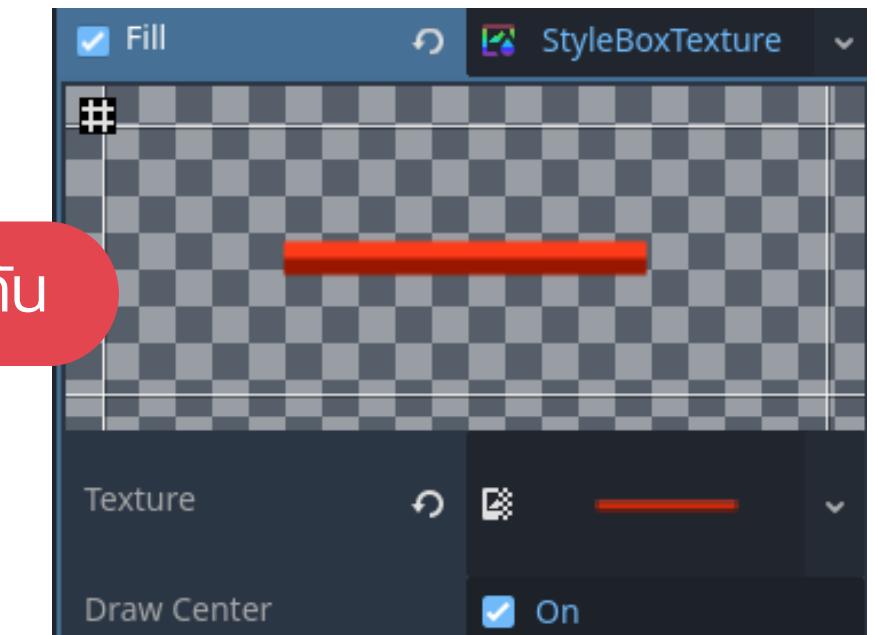
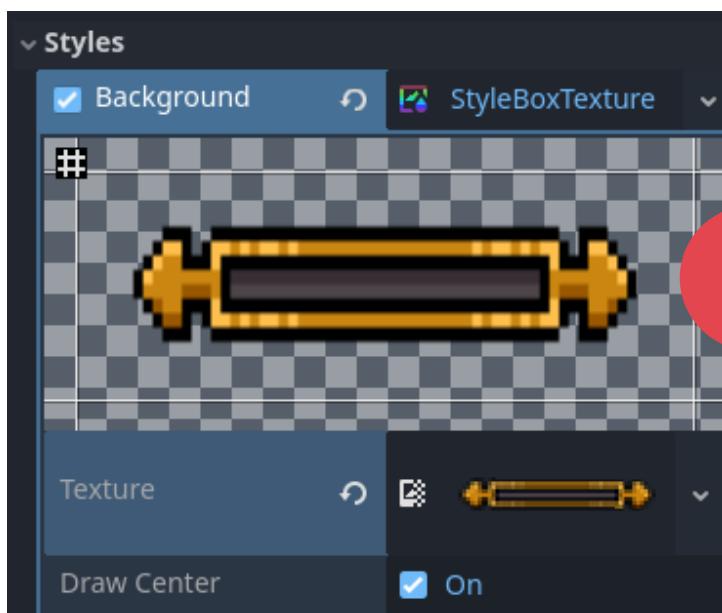


เอาเครื่องหมายถูกที่ Show Percentageออก และใส่ max value เป็น 5 และ value เป็น 5

ไปที่ Theme Overrides เปิด Styles



คลิก Drop Down ที่ Background และ Fill (กำลังจะคลิก) ให้เลือก New StyleBoxTexture หลังจากนั้นคลิก Load เอาภาพ Panel เข้าไป



ปรับขนาดของกลุ่ม UI ส่วนนี้หลังจากนั้นให้ทำการลากไปยังตำแหน่งที่เหมาะสมของตัวละคร

**ช่องหลักที่ดีที่สุดคือ ส่วนของศรีษะของ Player**



## ขั้นตอนที่ 6 เขียนคำสั่งไปควบคุมแสดงผล HP Bar

ผู้เล่นเราจะมี **ProgressBar** แล้วขั้นตอนต่อไปคือการเปิดไฟล์ `playercontroller.gd` ขึ้นมาหลังจากนั้นเขียนโปรแกรมคำสั่งดังนี้:

```
@onready var hp_bar: ProgressBar = $ProgressBar  
@export var hp = 5
```

ประกาศตัวแปร รับค่า `ProgressBar` และค่า `hp` ของผู้เล่น กำหนดมีค่าเป็น 5 ตามที่เราออกแบบไว้

เพิ่มฟังก์ชัน `_ready()` ขึ้นมา กำหนด `max_value` ของ `hp_bar` เป็น 5 และเชื่อมกับ `hp`

```
func _ready():  
    hp_bar.max_value = 5
```

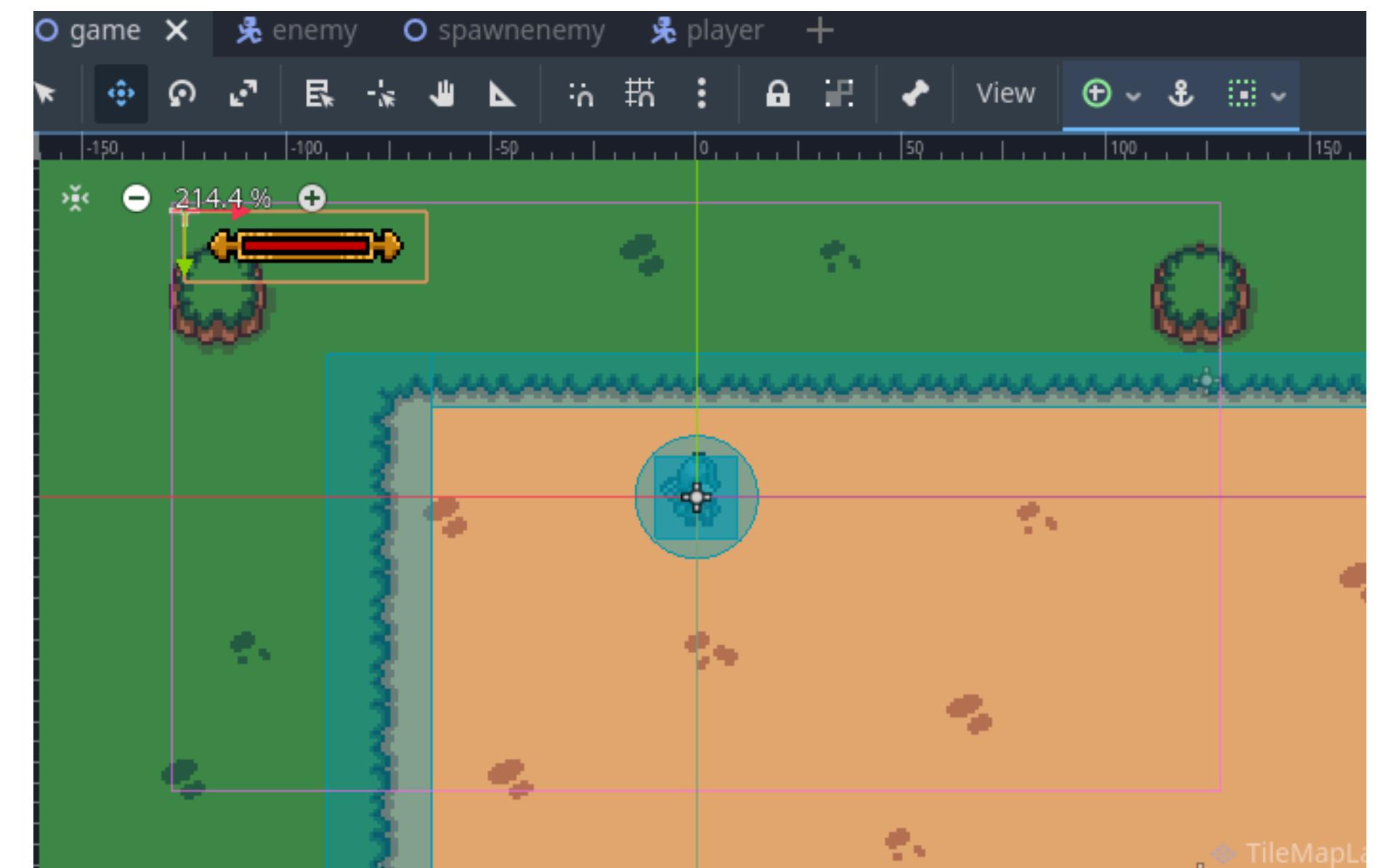
เพิ่บคำสั่งให้ `hp_bar` เชื่อมกับ `hp` ที่เป็นตัวแปรพลังงานเรา

```
hp_bar.value = hp
```

```
15 func _ready():  
16     hp_bar.max_value = 5  
17     hp_bar.value = hp  
18  
19 func _physics_process(delta: float) -> void:  
20     var direction = Vector2.ZERO  
21     hp_bar.value = hp  
22     # ถ้าโจมตีอยู่ ไม่ให้เคลื่อนที่หรือเปลี่ยนอะนิเมชัน
```



ໃດນໂຈບຕີກົຈະຕາຍ



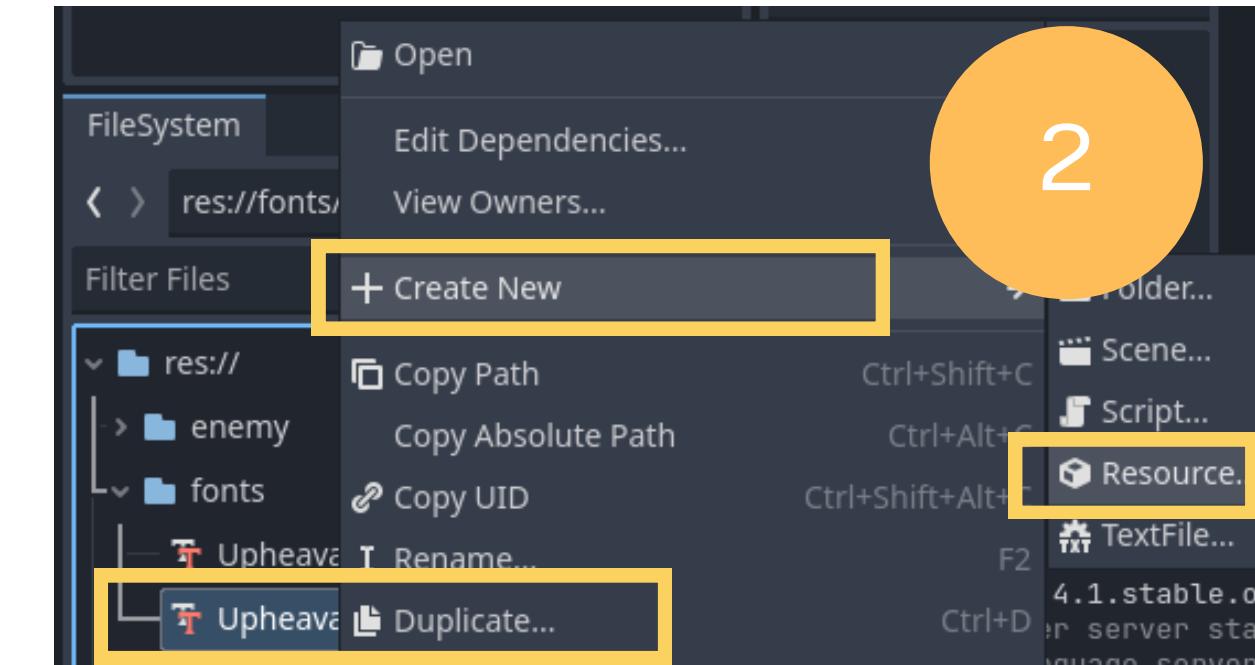
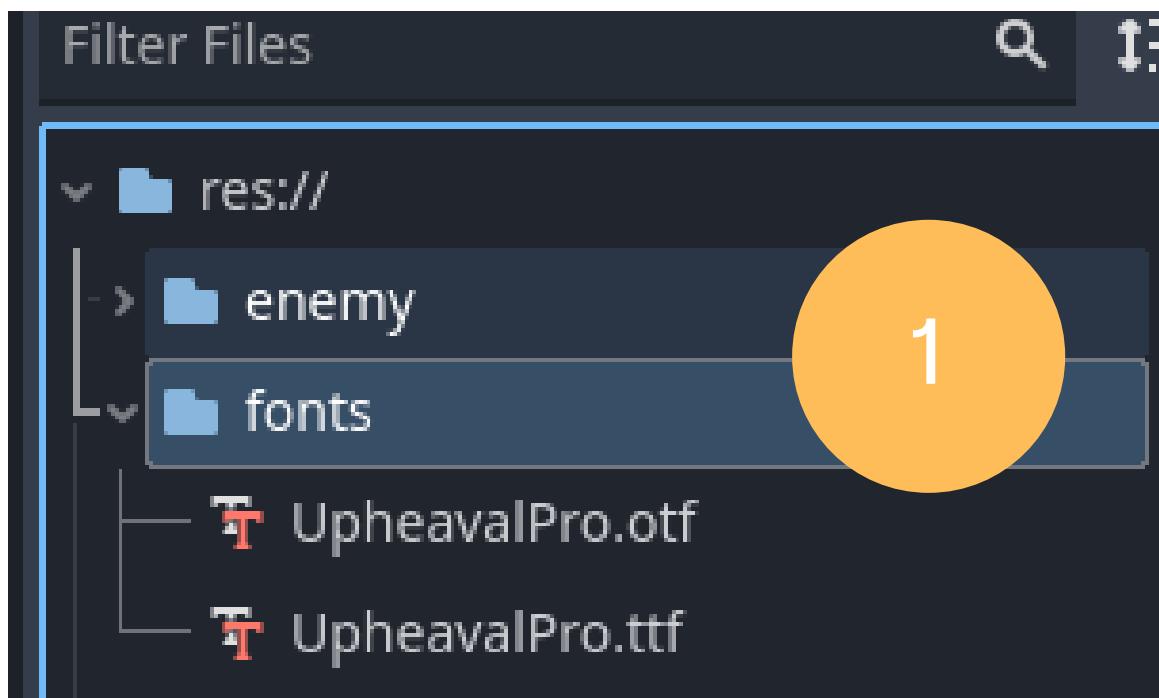
ວາງຕຳແໜ່ນ ProgressBar ໃຫມໍໃຫ້ເຮັດວຽກຕາມຄວາມແນະສນ

# FORGOTTEN STEEL

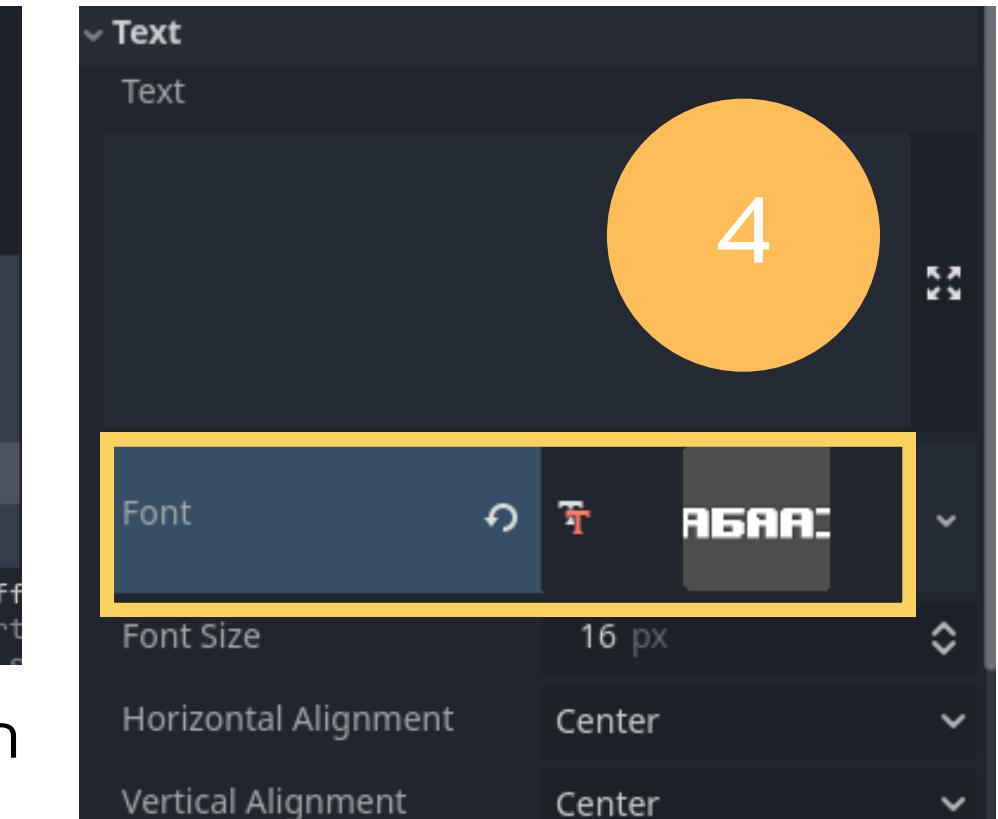
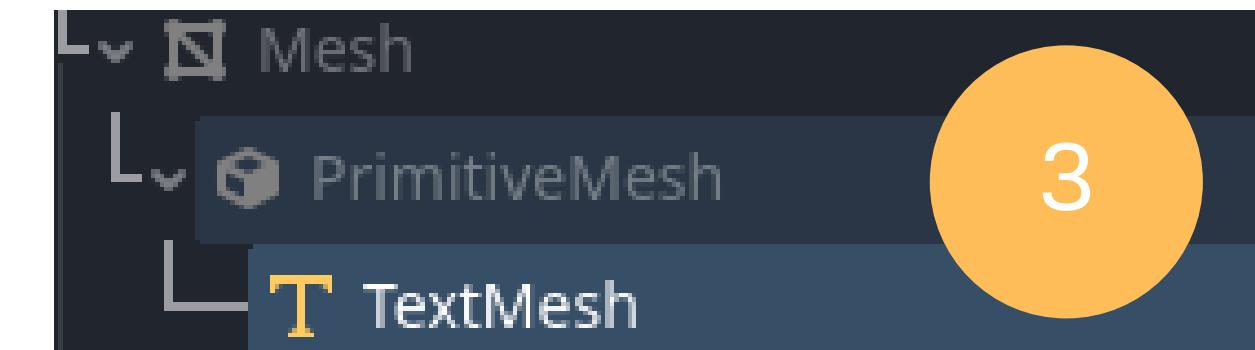
## FORGOTTEN STEEL

### ขั้นตอนที่ 7 การนำเข้าตัวอักษร

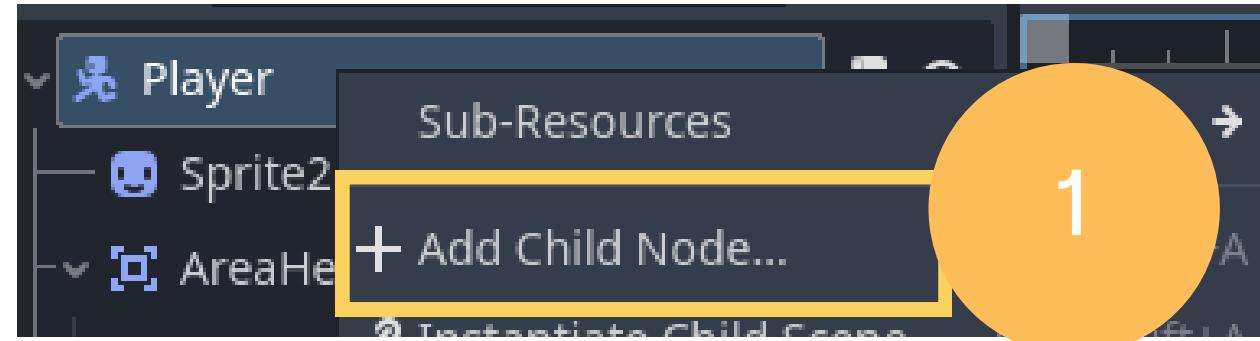
เรารสามารถดาวน์โหลด **Font** มาใช้งานกับ Project ของเราได้ โดยการสร้าง Folder `res://` ว่า **Fonts** และลากไฟล์ **.ttf, .otf** ไปไว้ในนั้น



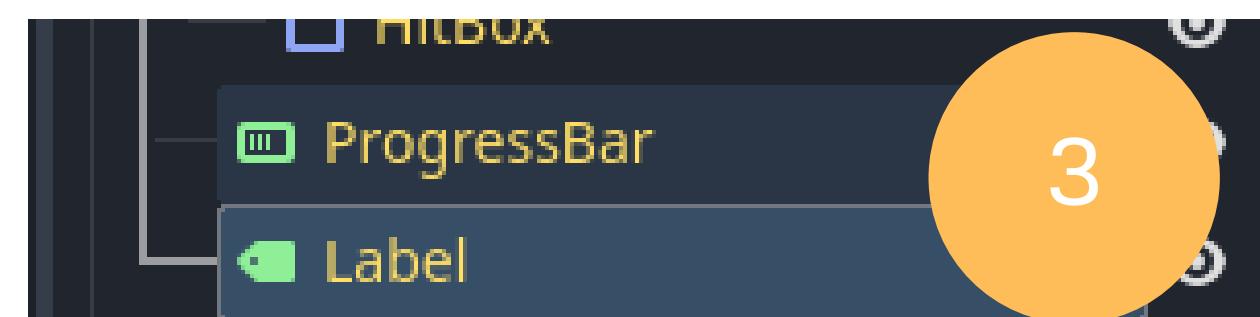
คลิกขวาที่ไฟล์ **.ttf** (ตัวอักษร) ใน `res://` หลังจากนั้นเลือก **+ Create New** และเลือก **Resource...**  
เลือก **TextMesh**



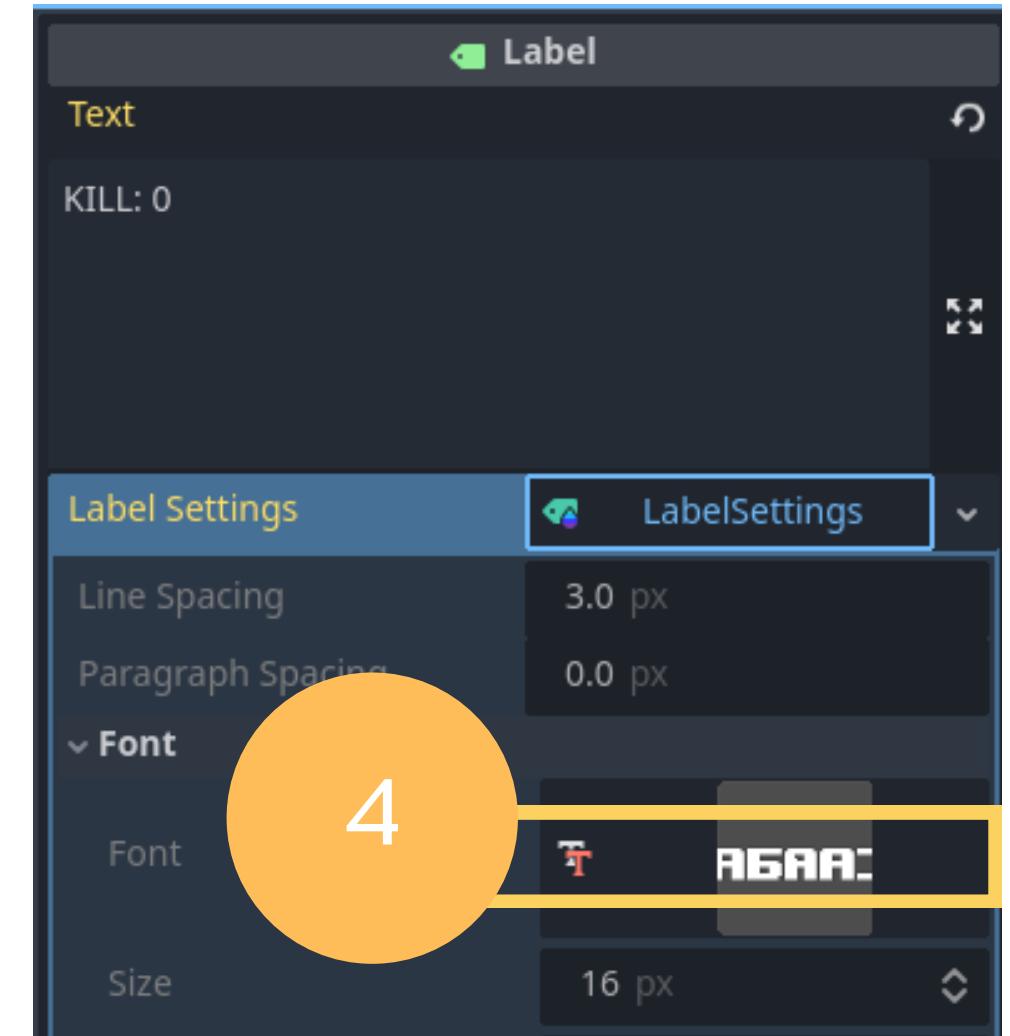
ไปที่ Inspector ลากไฟล์ Font **.ttf** ของเรามาเข้าไปใส่ที่ส่วนของ Font  
ใน TextMesh Resource



ทำการ เพิ่ม Child ของ Player เลือก **Label**



วาง Label ไว้ตำแหน่งไหนก็ได้ไปที่ Inspector



กำหนดการสร้าง LabelSettings แล้วใส่ Font ด้วย ตัวอักษรของเราก็ เราสร้าง หรือดาวน์โหลดมา .ttf

```
@onready var kill_label: Label = $Label
var kill_count = 0
```

ประกาศตัวแปรที่ **playercontroller.gd**  
ให้เชื่อมกับ Label ของเราที่ kill\_label

```
func add_kill():
    kill_count += 1
    kill_label.text = "Kill: %d" % kill_count
```

เพิ่มฟังก์ชันใหม่ว่า **add\_kill()**: เข้าไปเพื่อ  
มีการโจมตีศัตรูแล้วศัตรูตายให้ส่งมาก็นี่  
ไปเพิ่มคำสั่งใน **enemy.gd** กี่ฟังก์ชัน die()

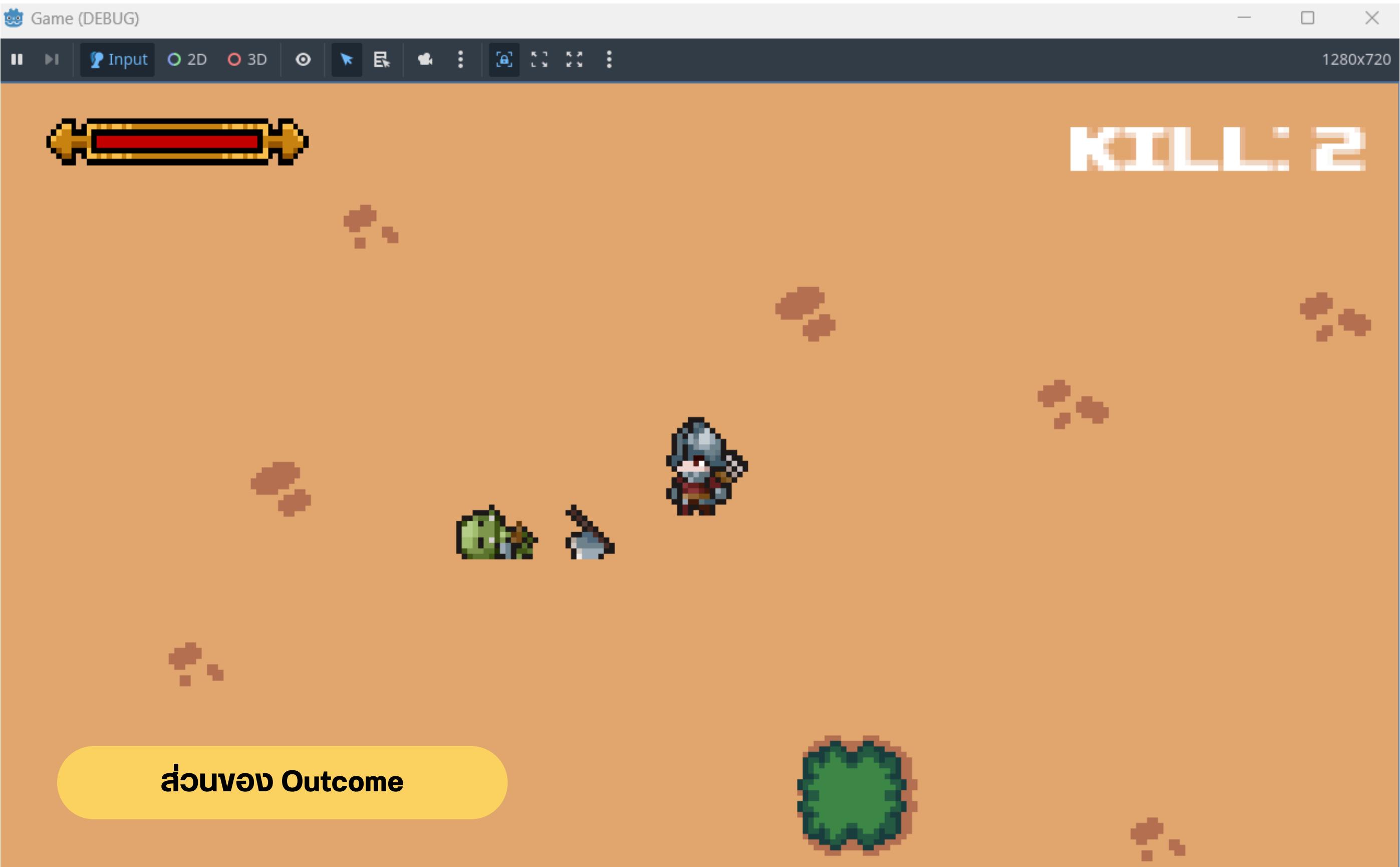
```
collisionShape.disabled = true
collisionShapeArea.disabled = true
player.add_kill()
```

## ໄວ້ playercontroller.gd

```
15  @onready var kill_label: Label = $Label  
16  var kill_count = 0  
17  
61  func add_kill():  
62    >| kill_count += 1  
63    >| kill_label.text = "Kill: %d" % kill_count  
64
```

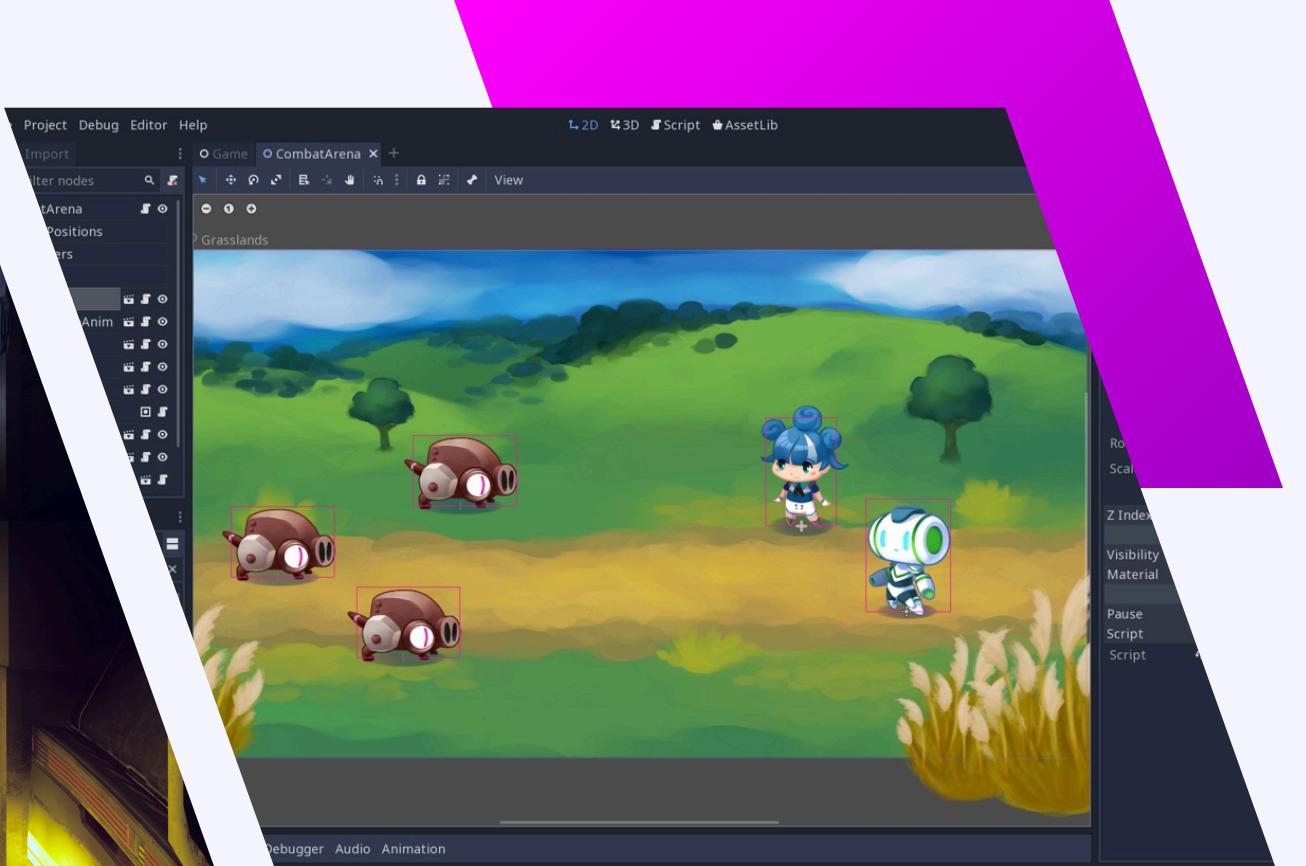
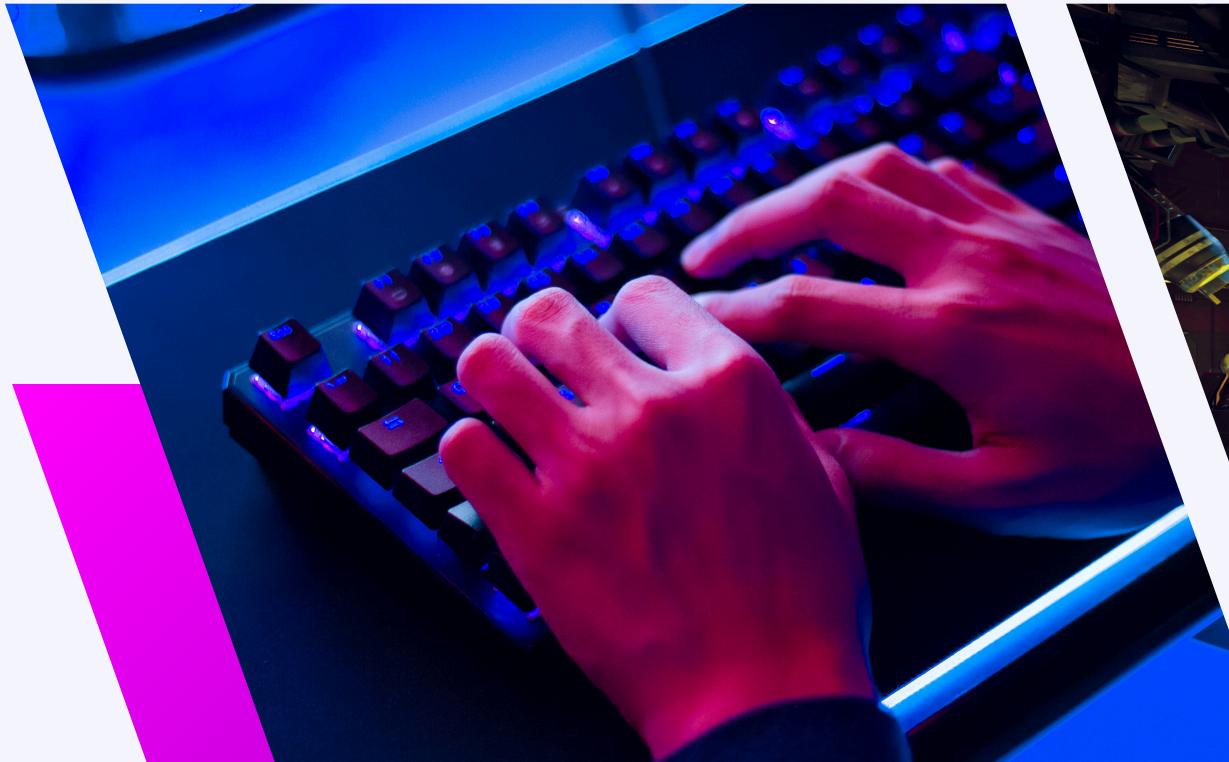
## ໄວ້ enemy.gd

```
48  func die():  
49    >| is_dead = true  
50    >| velocity = Vector2.ZERO  
51    >| animation_player.play("Death")  
52    >|  
53    >| # ໂຄນສະໜັກຂອງວາ  
54    >| collision_layer = 0  
55    >| collision_mask = 0  
56    >| collisionShape.disabled = true  
57    >| collisionShapeArea.disabled = true  
58    >| player.add_kill() # ເຮັດ player ມີເລີຍ ດັກ  
59  >| if is_in_group("Enemy"):  
60    >| >| remove_from_group("Enemy")  
61    >| death_timer.start(0.8)>|
```



# GAME DEVELOPER

College of Creative Design and Entertainment Technology



การพัฒนาเกมด้วยโปรแกรม Godot Engine