Godot 2D Game
Lesson 11: Scoring Systems

Having a game where you never actually win or lose is usually not very entertaining for the player. So in this lesson, we will add a simple scoring system. We will give the player a time limit and award points based on how many bubbles they pop. Let's start by adding 2 new variables to our "Main" scene:

```
export (PackedScene) var Bubble
export var max_bubble_cnt = 10

var screen_rect
var bubble_cnt = 0
var score
var time
```

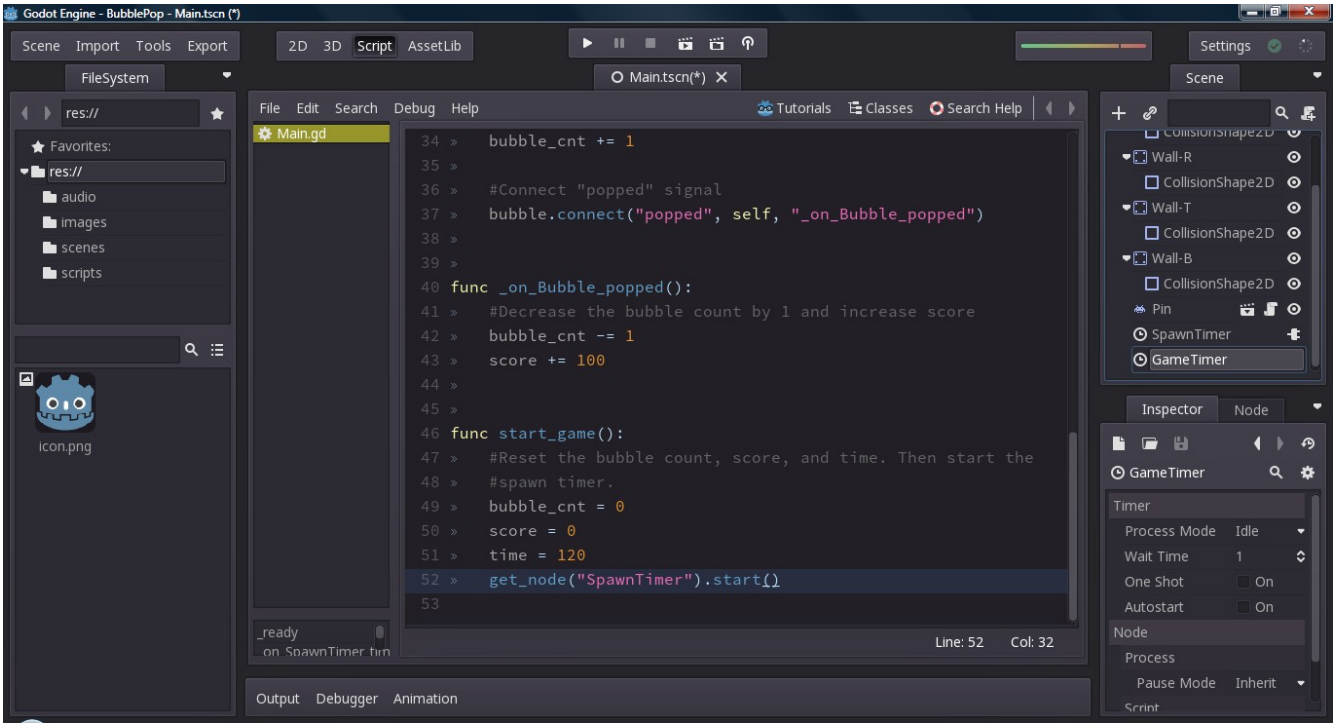
Our new "score" variable will be used to keep track of the player's current score and our new "time" variable will be used to keep track of the remaining time. Now let's update our "start_game" function so that it resets the score and time:

```
func start_game():
	#Reset the bubble count, score, and time. Then start the
	#spawn timer.
	bubble_cnt = 0
	score = 0
	time = 120
	get_node("SpawnTimer").start()
```

To update the player's score, we will simply add 100 for each bubble they pop:

```
func _on_Bubble_popped():
	#Decrease the bubble count by 1 and increase score
	bubble_cnt -= 1
	score += 100
```

To update the timer, we will need to create a new Timer node. We will call this one "GameTimer":



Now we need to attach a new function to the "timeout" signal of the new timer:

```
func _on_GameTimer_timeout():
	#Update the timer variable
	time -= 1
```

Updating the timer variable is as simple as subtracting one each time the game timer fires. We also need to remember to start the game timer:

```
func start_game():
	#Reset the bubble count, score, and time. Then start the
	#spawn and game timers.
	bubble_cnt = 0
	score = 0
	time = 120
	get_node("SpawnTimer").start()
	get_node("GameTimer").start()
```

We now have the basis for our scoring system. However, we currently have no way to actually see the current score and time. In the next lesson we will create a HUD to display them.