| **Content Type** | **Tutorial** |
| --- | --- |
| **Title** | **Lesson 4.4 - For-Loops For Waves** |
| **Summary** | **Overview:**  We have all the makings of a great game; A player that rolls around and rotates the camera, a powerup that grants super strength, and an enemy that chases the player until the bitter end. In this lesson we will wrap things up by putting these pieces together!  First we will enhance the enemy spawn manager, allowing it to spawn multiple enemies and increase their number every time a wave is defeated. Lastly we will spawn the powerup with every wave, giving the player a chance to fight back against the ever-increasing horde of enemies.  **Project Outcome:**  The Spawn Manager will operate in waves, spawning multiple enemies and a new powerup with each iteration. Every time the enemies drop to zero, a new wave is spawned and the enemy count increases. |
| **Overview Video** | [Lesson 4.4 - For-Loops For Waves](https://youtu.be/Rpmz-XKCzX8) |
| **Cover Image** |  |
| **Unity Version** | 2018.4 - 2020.3 |
| **Difficulty Level** | Beginner |
| **Estimated Time** | 0h 50min |
| **Skills** | Use common logic structures to control the execution of code.  Write code that utilizes the various Unity APIs  Write code that integrates into an existing system |
| **XP Categories and Values** | Unity Essentials - 5 xp  Programming - 5 xp |
| **Unity editor packages** | - |
| **Tags** | - |
| **Topics** | For Educators  Scripting |
| **Industries** | - |
| **Video transcripts** | - |
| **Files and assets** | - |
| **Unity docs links** | - |
| **Asset store links** | - |
| **Learn group link** | - |
| **Hidden / Visible** | Hidden |

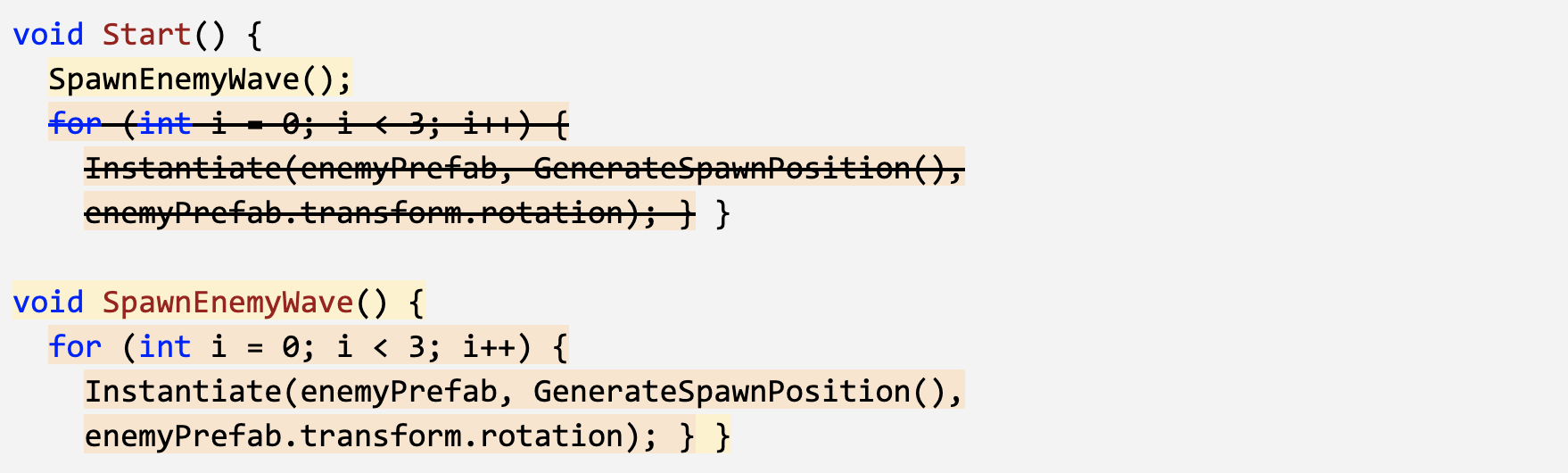
## Step 1: Write a for-loop to spawn 3 enemies

*We should challenge the player by spawning more than one enemy. In order to do so, we will repeat enemy instantiation with a loop.*

**CWC\_U4.L4.S1\_T1\_v01 -** [**https://youtu.be/vHVbsAeAndg**](https://youtu.be/vHVbsAeAndg)

***CwC 4.4.1 Write a for-loop to spawn 3 enemies***

1. In SpawnManager.cs, in ***Start()***, replace single **Instantiation** with a **for-loop** that spawns 3 enemies
2. Move the for-loop to a new ***void SpawnEnemyWave()*** function, then call that function from ***Start()***

******

**Alt text:** *-*

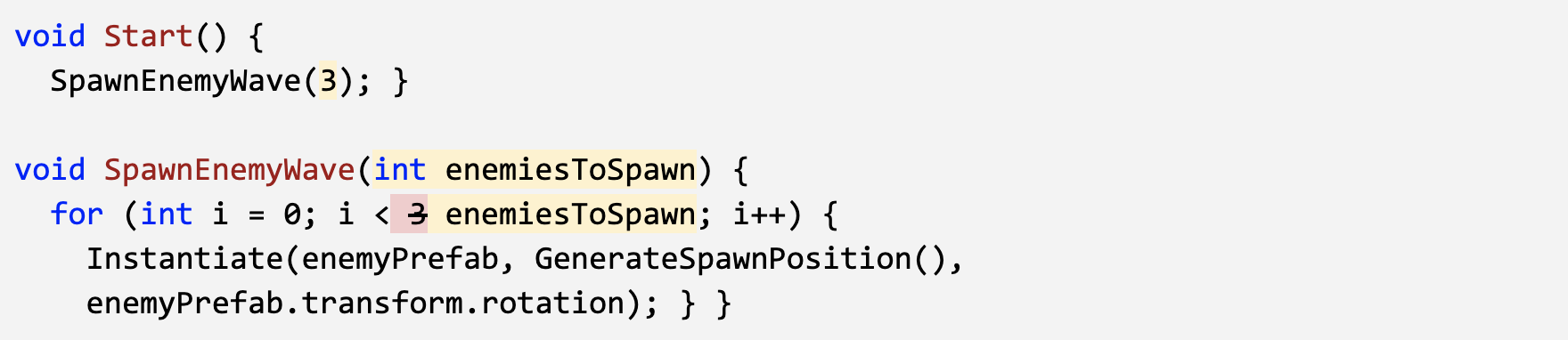
## Step 2: Give the for-loop a parameter

*Right now, SpawnEnemyWave spawns exactly 3 enemies, but if we’re going to dynamically increase the number of enemies that spawn during gameplay, we need to be able to pass information to that method.*

**CWC\_U4.L4.S2\_T1\_v01 -** [**https://youtu.be/ihpAJB-usWA**](https://youtu.be/ihpAJB-usWA)

***CwC 4.4.2 Give the for-loop a parameter***

1. Add a parameter ***int enemiesToSpawn*** to the ***SpawnEnemyWave*** function
2. Replace ***i < \_\_*** with ***i < enemiesToSpawn***
3. Add this new variable to the function call in ***Start()***: ***SpawnEnemyWave(\_\_\_);***

******

**Alt text:** *-*

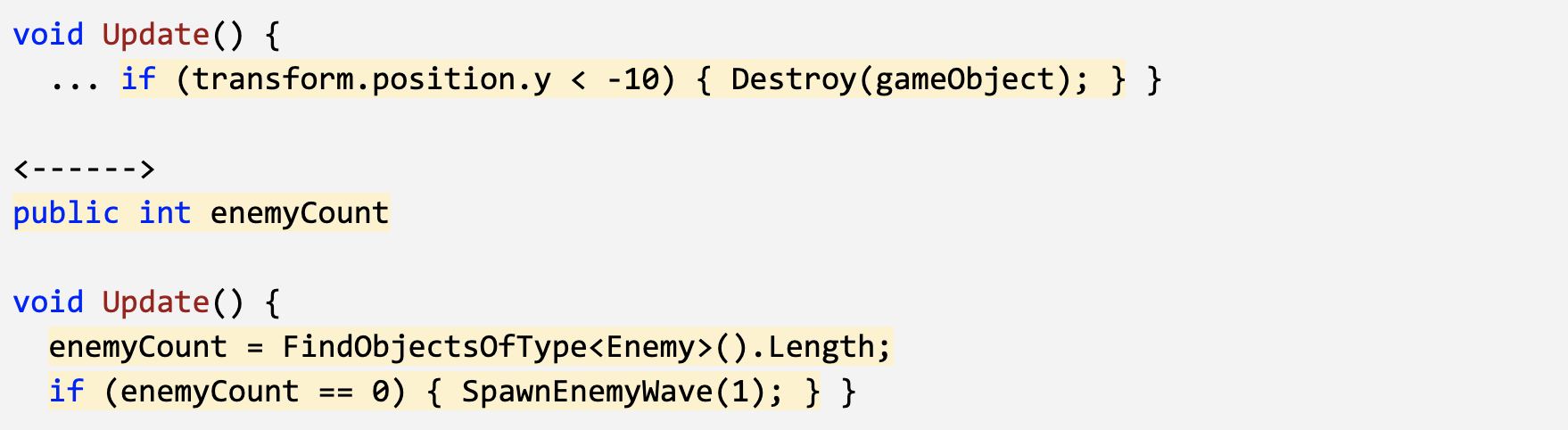
## Step 3: Destroy enemies if they fall off

*Once the player gets rid of all the enemies, they’re left feeling a bit lonely. We need to destroy enemies that fall, and spawn a new enemy wave once the last one is vanquished!*

**CWC\_U4.L4.S3\_T1\_v01 -** [**https://youtu.be/QDr-cvFUUs8**](https://youtu.be/QDr-cvFUUs8)

***CwC 4.4.3 Destroy enemies if they fall off***

1. In Enemy.cs, **destroy** the enemies if their position is less than a **-Y value**
2. In SpawnManager.cs, declare a new ***public int enemyCount*** variable
3. In ***Update()***, set ***enemyCount = FindObjectsOfType<Enemy>().Length;***
4. Write the **if-statement** that if ***enemyCount == 0*** then ***SpawnEnemyWave***, then delete it from ***Start()***

******

**Alt text:** *-*

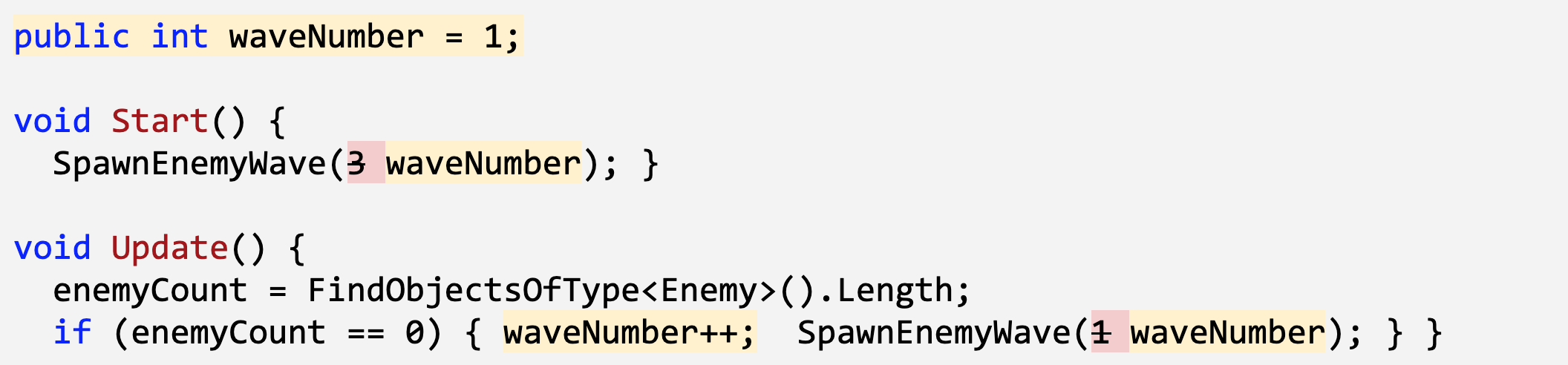
## Step 4: Increase enemyCount with waves

*Now that we control the amount of enemies that spawn, we should increase their number in waves. Every time the player defeats a wave of enemies, more should rise to take their place.*

**CWC\_U4.L4.S4\_T1\_v01 -** [**https://youtu.be/-UP81vsK0Ew**](https://youtu.be/-UP81vsK0Ew)

***CwC 4.4.4 Increase enemyCount with waves***

1. Declare a new ***public int waveNumber = 1;***, then implement it in ***SpawnEnemyWave(waveNumber);***
2. In the if-statement that tests if there are 0 enemies left, **increment waveCount** by 1



**Alt text:** *-*

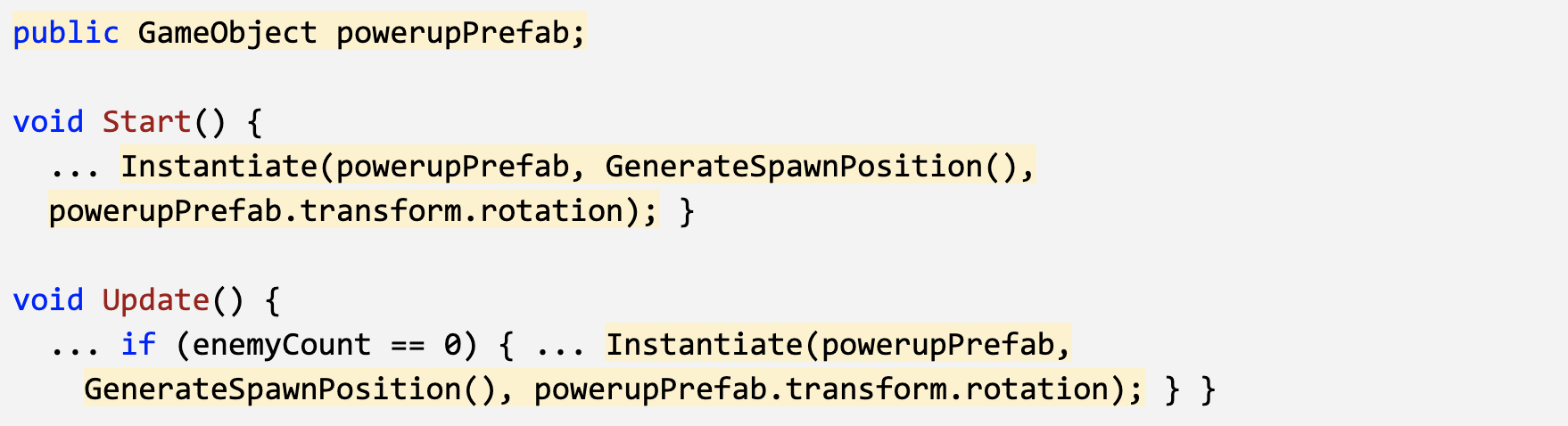
## Step 5: Spawn Powerups with new waves

*Our game is almost complete, but we’re missing something. Enemies continue to spawn with every wave, but the powerup gets used once and disappears forever, leaving the player vulnerable. We need to spawn the powerup in a random position with every wave, so the player has a chance to fight back.*

**CWC\_U4.L4.S5\_T1\_v01 -** [**https://youtu.be/6FLumxl1si4**](https://youtu.be/6FLumxl1si4)

***CwC 4.4.5 Spawn Powerups with new waves***

1. In SpawnManager.cs, declare a new ***public GameObject powerupPrefab*** variable, assign the **prefab** in the inspector and **delete** it from the scene
2. In **Start()**, **Instantiate** a new Powerup
3. Before the ***SpawnEnemyWave()*** call, **Instantiate** a new Powerup



**Alt text:** *-*

## Step 6: Lesson Recap

**CWC\_U4.L4.SRecap\_T1\_v02 -** [**https://youtu.be/rzxnF7cKZW4**](https://youtu.be/rzxnF7cKZW4)

***CwC 4.4.6 Lesson Recap***

New Functionality:

* Enemies spawn in waves
* The number of enemies spawned increases after every wave is defeated
* A new power up spawns with every wave

New Concepts and Skills**:**

* For-loops
* Increment (++) operator
* Custom methods with parameters
* FindObjectsOfType