

3.2.8 Improve the Game Experience

For this last part of game improvements, let's add the following messages to notify users about the state of the game:

- The round number
- Game over

Adding these messages will improve the player's experience, which is always a bonus with game jam judges. Earlier in this lesson we removed the welcome message, `alert("Welcome to Robot Gladiators!")`, from the `fight()` function. Let's place it back in the game and add the round number.

Fix the Round Number Notification

Before we can modify the round notification, we must first determine precisely when the new round will begin.

Hypothetically a new round should start when the player robot starts fighting with a new enemy robot.

The game will introduce new enemy robots by iterating through the `enemyNames` array in the `for` loop.

Can you think of a scenario when a new enemy robot is presented, but a new round of battle should **NOT** begin?

In one scenario, an enemy robot is introduced but the player robot has no health, so is unable to fight. Currently the round alert message triggers even when the player's robot has lost. Let's fix this situation before proceeding with the round number.

If a player robot has been defeated, the health condition in the `while` loop, `playerHealth > 0`, will exit back to the `for` loop. The `for` loop will then continue to iterate through the `enemyNames` array until there are no enemy robots left. This means we should also check in the `for` loop if the player robot has been defeated.

Add the following statement to the top of the `for` loop code block to check on the state of the player robot's health:

```
if (playerHealth > 0) {  
  window.alert("Welcome to Robot Gladiators!");  
}
```

Now its time to add the round number to the round notification.

PAUSE

What's the connection between the iterator in the `for` loop and the round number?

Each new robot initiates a new round and the iterator increments at the beginning of each round, so these two counters map

together well. However, the round number starts at one while the iterator starts at zero.

[Hide Answer](#)

Increasing the iterator by one allows the round number to be calculated as follows. Edit the `window.alert` to this:

```
window.alert("Welcome to Robot Gladiators! Round " + ( i + 1 ) );
```

Notice how the `(i + 1)` was wrapped in parentheses. What happens if we remove the parentheses?

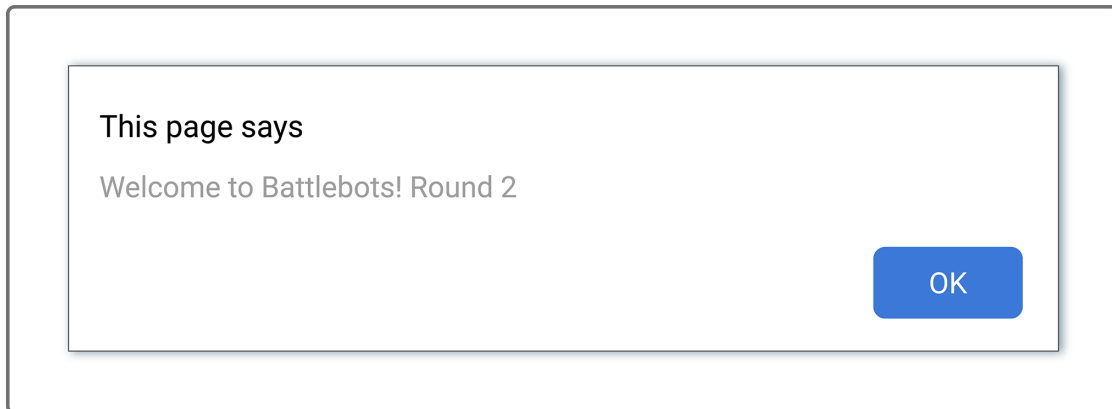
The parentheses allow the arithmetic operation to perform and then concatenate this sum to the string message. Without the parentheses, the `i` does not add the one but treats it like a string and concatenates the string "1", which isn't what we're looking for.

Let's place all the code in the `for` loop inside the `playerHealth` condition so that loop iterates only when the player robot is alive.

```
if (playerHealth > 0) {  
  // let user know what round they are in, remember that arrays start at 0  
  window.alert("Welcome to Robot Gladiators! Round " + (i + 1));  
  
  // pick new enemy to fight based on the index of the enemyNames array  
  var pickedEnemyName = enemyNames[i];  
  
  // reset enemyHealth before starting new fight  
  enemyHealth = 50;  
  
  // use debugger to pause script from running and check what's going on  
  // debugger;  
  
  // pass the pickedEnemyName variable's value into the fight function
```

```
fight(pickedEnemyName);  
}
```

Now when the player robot survives to a new round, the notification should now look like this:



Next we should add a message that notifies the player when the player robot has been defeated and the game has ended. We can use the `else` conditional statement because this occurs only when the player robot has no health or has been defeated.

Let's add this statement and a `break` below the `if` statement:

```
else {  
    window.alert("You have lost your robot in battle! Game Over!");  
    break;  
}
```

To execute this statement, what is the current state of the player robot?
The player robot has been defeated.

Nice work! That's a lot of information to take in. Take a quick assessment to test your knowledge.

Which is NOT the proper way to iterate through this `cars` array to display each element?

- ☐

```
for (var i = 0; i < cars.length; i++) {  
  console.log(cars[i]);  
}
```
- ☐

```
for (var i = 0; i < cars.length; i++) { console.log(i, cars[i]); }
```
- ☐

```
for (var i = 0, i > cars.length, i++) {  
  console.log(cars[i]);  
}
```
- ☐

```
var i = cars.length-1;  
while (i >= 0) {  
  console.log(cars[i]);  
  i--;  
}
```

Check Answer

Using the same `cars` array from the previous question, will the following code ever display the "Stop" message?

```
for (var i = 0; i < cars.length; i++) {  
  if (cars[i]) {  
    console.log("Vroom");  
  } else {  
    console.log("Stop");  
  }  
}
```

- ☐ Yes, at least once.
- ☐ No, never.
- ☐ It depends.

Check Answer

Finish ►