In this first lesson, we'll use JavaScript to code a single round of Robot Gladiators. When we're done, the game will follow this sequence:

1. The game will prompt the player to name their robot.
2. The player's robot will be initialized with the following properties:
   * 100 health points
   * 10 attack points
   * 10 money points
3. The player's opponent, Roborto, will be initialized with the following properties:
   * 50 health points
   * 12 attack points
4. The game will display "Welcome to Robot Gladiators!"
5. The game will prompt the player to either fight the round or skip it.
6. If the player chooses to skip:
   * A penalty of 10 money points will be deducted from the player's robot.
   * The game will end.
7. If the player chooses to fight:
   * The player's robot will attack Roborto, and the player-robot's attack points will be deducted from Roborto's health points.
   * The game will display Roborto's remaining health points.
   * Roborto will attack the player's robot, and Roberto's attack points will be deducted from the player's robot's health points.
   * The game will display the player-robot's remaining health points.
8. The game will end.

**Title: Initial game functionality - MVP**

\*\* Description \*\*

\*\* Must Haves \*\*

* Build a game where a player's robot can fight another robot until one of them loses.
* If the enemy-robot loses first, the player's robot will move on to fight another enemy-robot.

\*\* Features \*\*

* The player's robot's name can be dynamically created by the player through the browser.
* The player is given the option to skip the fight by paying a penalty fee, or continue with the fight.

**Title: Add shop functionality**

\*\*Description\*\*

* After defeating an enemy, ask the player if they would like to purchase an item from a store.
* The player can purchase health or attack points if they can afford it.

**Add randomness to health and damage values**

\*\*Description\*\*

* Start enemies at a random health value between 40 and 60.
* Start enemies with a random attack value between 10 and 14.
* Attack damage is random, using the robot's attack value as an upper limit (for example, if the player's attack is 10, their damage range is 7-10).

**Restructure data with JS objects**

\*\*Description\*\*

* Create and use JavaScript objects to organize data related to the player and data related to the enemies.
* Create object methods to perform player actions.