REWARDS AND OBSTACLES

What is our GOAL for this MODULE?

In this class, we created sprites and added them to groups on a real-time basis. You also created random fuel & power coins as rewards for players to collect.

What did we ACHIEVE in the class TODAY?

- Created sprites on a real-time basis & add them at random positions
- Created sprites in real-time & displayed them in fixed positions
- Learned about the **overlap()** function to remove the fuel tank & coin which the player touches.

Which CONCEPTS/ CODING BLOCKS did we cover today?

- Sprite groups
- The **overlap()** function
- Initializing a Parameter "**positions=[]"** in function.

How did we DO the activities?

To add multiple sprites for the fuel tank and powerCoins at random positions in our game.

1. Create two variables for a group of sprites.

```
fuels = new Group();
powerCoins = new Group();
```

2. Create a function **addSprites()** to create sprites for fuel and powerCoins.
    - Give a random position to each sprite.
    - Add **fuels** & **powerCoins** sprites into their respective groups.

```
// Adding fuel sprite in the game
this.addSpirtes(fuels, 4, fuelImage, 0.02);

// Adding coin sprite in the game
this.addSpirtes(powerCoins, 18, powerCoinImage, 0.09);
```

3. Create an **overlap()** function to check when a player's car is touching any of the sprites (fuel/powerCoins).
    - Remove the touching sprite from the game.

```
handleFuel(index) {
  // Adding fuel
  cars[index - 1].overlap(fuels, function(collector, collected) {
    player.fuel = 185;
    //collected is the sprite in the group collectibles that triggered
    //the event
    collected.remove();
  });
}

handlePowerCoins(index) {
  cars[index - 1].overlap(powerCoins, function(collector, collected) {
    player.score += 21;
    player.update();
    //collected is the sprite in the group collectibles that triggered
    //the event
    collected.remove();
  });
}
```

Output:

Now, to add multiple sprites for obstacles at a fixed position in our game to make it interesting.

4. Declare global variables for images and obstacles in **sketch.js**.
    - Preload images in these variables in **preload()** function.

```
var canvas;
var backgroundImage, car1_img, car2_img, track;
var fuelImage, powerCoinImage, lifeImage;
var obstacle1Image, obstacle2Image;

var database, gameState;
var form, player, playerCount;
var allPlayers, car1, car2, fuels, powerCoins, obstacles;
var cars = [];

function preload() {
  backgroundImage = loadImage("./assets/background.png");
  car1_img = loadImage("../assets/car1.png");
  car2_img = loadImage("../assets/car2.png");
  track = loadImage("../assets/track.jpg");
  fuelImage = loadImage("./assets/fuel.png");
  powerCoinImage = loadImage("./assets/goldCoin.png");
  lifeImage = loadImage("./assets/life.png");
  obstacle1Image = loadImage("./assets/obstacle1.png");
  obstacle2Image = loadImage("./assets/obstacle2.png");

}
```

5. Create an **obstacles** group in the **start()** method of **Game.js.**

```
obstacles = new Group();
```

6. Use the predefined positions for each obstacle, sprite an array **obstaclesPositions.**

```
var obstaclesPositions = [
  { x: width / 2 + 250, y: height - 800, image: obstacle2Image },
  { x: width / 2 - 150, y: height - 1300, image: obstacle1Image },
  { x: width / 2 + 250, y: height - 1800, image: obstacle1Image },
  { x: width / 2 - 180, y: height - 2300, image: obstacle2Image },
  { x: width / 2, y: height - 2800, image: obstacle2Image },
  { x: width / 2 - 180, y: height - 3300, image: obstacle1Image },
  { x: width / 2 + 180, y: height - 3300, image: obstacle2Image },
  { x: width / 2 + 250, y: height - 3800, image: obstacle2Image },
  { x: width / 2 - 150, y: height - 4300, image: obstacle1Image },
  { x: width / 2 + 250, y: height - 4800, image: obstacle2Image },
  { x: width / 2, y: height - 5300, image: obstacle1Image },
  { x: width / 2 - 180, y: height - 5500, image: obstacle2Image }
];
```

7. Call the **addSprites()** function to create **obstacleSprites** and add to the group.
    - Modify **addSprites()** to pass positions of each obstacle sprites.
    - Do not modify previously made calls, as we are also initializing the default value to the new parameter.

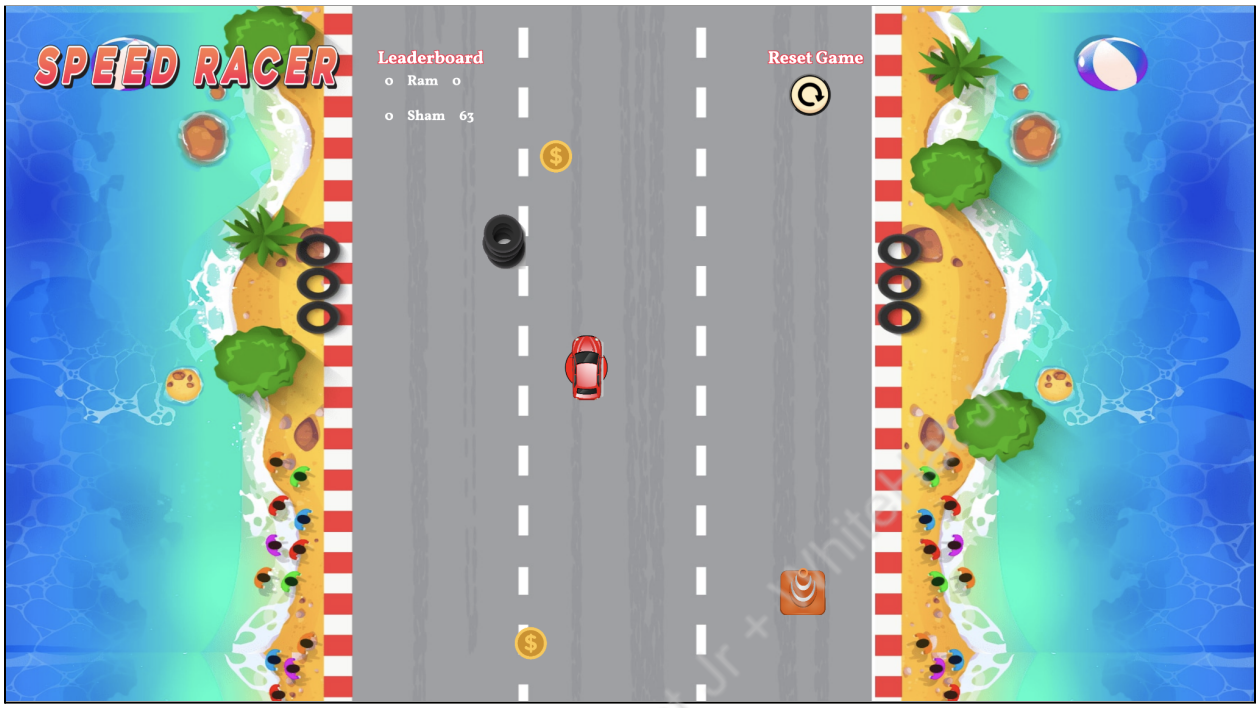
```
  //Adding obstacles sprite in the game
  this.addSpirtes( obstacles, obstaclesPositions.length, obstacle1Image, 0.04, obstaclesPositions );
}

addSpirtes(spriteGroup, numberOfSprites, spirteImage, scale, positions = []) {
  for (var i = 0; i < numberOfSprites; i++) {
    var x, y;
```

    - Modify **addSprite()** function to create obstacles at predefined positions.

```
addSpirtes(spriteGroup, numberOfSprites, spirteImage, scale, positions =
[]) {

  for (var i = 0; i < numberOfSprites; i++) {

        var x, y;

      if (positions.length > 0) {

        x = positions[i].x;

        y = positions[i].y;

        spirteImage = positions[i].image;
```

```
    } else {
      x = random(width / 2 + 150, width / 2 - 150);
      y = random(-height * 4.5, height - 400);
    }
    var spirte = createSprite(x, y);
    spirte.addImage("spirte", spirteImage);

    spirte.scale = scale;
    spriteGroup.add(spirte);
  }
}
```

Output:



What's next?

In the next class, you will build a ranking mechanism that ranks the player according to their performance in the car racing game. We will also build a Progress Bar for fuel and player's life property.

EXTEND YOUR KNOWLEDGE:

1. To know more about groups from the following link created by Mozilla and individual contributors:
   https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Regular_Expressions/Groups_and_Ranges

   About MDN by Mozilla Contributors is licensed under CC-BY-SA 2.5.