MARKING PLAYER & GAME CAMERA

What is our GOAL for this MODULE?

In this class, we learned the concept of the game cameras to focus the game on the active player in the game. We moved the cars forwards with an arrow key and updated the distance to the database.

What did we ACHIEVE in the class TODAY?

- Created a For-In loop to display the cars.
- Created an **ellipse()** below the car as an identifier.
- Used a Game Camera to focus on the active player in the game.
- Created **handlePlayerControls()** to move the car forward.
- Created methods to update and retrieve distance in the database.

Which CONCEPTS/ CODING BLOCKS did we cover today?

- Game Camera
- Array index to mark a player
- For-In loop

How did we DO the activities?

1. Create a for-in loop to give the position to each car inside play() in game.js.
    - Create a variable index as **var index = 0** to extract the car index from the 'cars' array.
    - Write **for(var plr in allPlayers)** loop to place the cars at their starting positions.

```
play() {
  this.handleElements();
  Player.getPlayersInfo();
  if (allPlayers !== undefined) {
    image(track, 0, -height * 5, width, height * 6);

    var index = 0;
    for (var plr in allPlayers) {
      index = index + 1;
      var x = allPlayers[plr].positionX;
      var y = height - allPlayers[plr].positionY;

      cars[index-1].position.x= x;
      cars[index-1].position.y = y;
    }
```

2. Call **this.handlePlayerControls()** after for-in loop of **play()**.

```
for (var plr in allPlayers) {
    //add 1 to the index for every loop
    index = index + 1;

    //use data form the database to display the cars in x and
    var x = allPlayers[plr].positionX;
    var y = height - allPlayers[plr].positionY;

    cars[index - 1].position.x = x;
    cars[index - 1].position.y = y;
}

this.handlePlayerControls();

drawSprites();
```

3. Create a condition to move the car upwards inside **handlePlayerControls()** of **Game.js**.

```
handlePlayerControls() {
// handling keyboard events
    if (keyIsDown(UP_ARROW)) {
        player.positionY += 10;
        player.update();
    }
}
```

4. Create an update() method in player.js to update the distance traveled by car to the database.

```
update() {
  var playerIndex = "players/player" + this.index;
  database.ref(playerIndex).update({
    positionX: this.positionX,
    positionY: this.positionY,
  });
}
```

5. Create a getDistance**()** method to retrieve the distance from the database. This allows for one player to see the position of the other player.

    - Call this method as **player.getDistance()** in **handleMousePressed()** method of **Form.js**

```
getDistance() {
   var playerDistanceRef = database.ref("players/player" + this.index);
   playerDistanceRef.on("value", data => {
     var data = data.val();
     this.positionX = data.positionX;
     this.positionY = data.positionY;
   });
 }
```

6. Add a small red circle below the car to identify the player's own car.

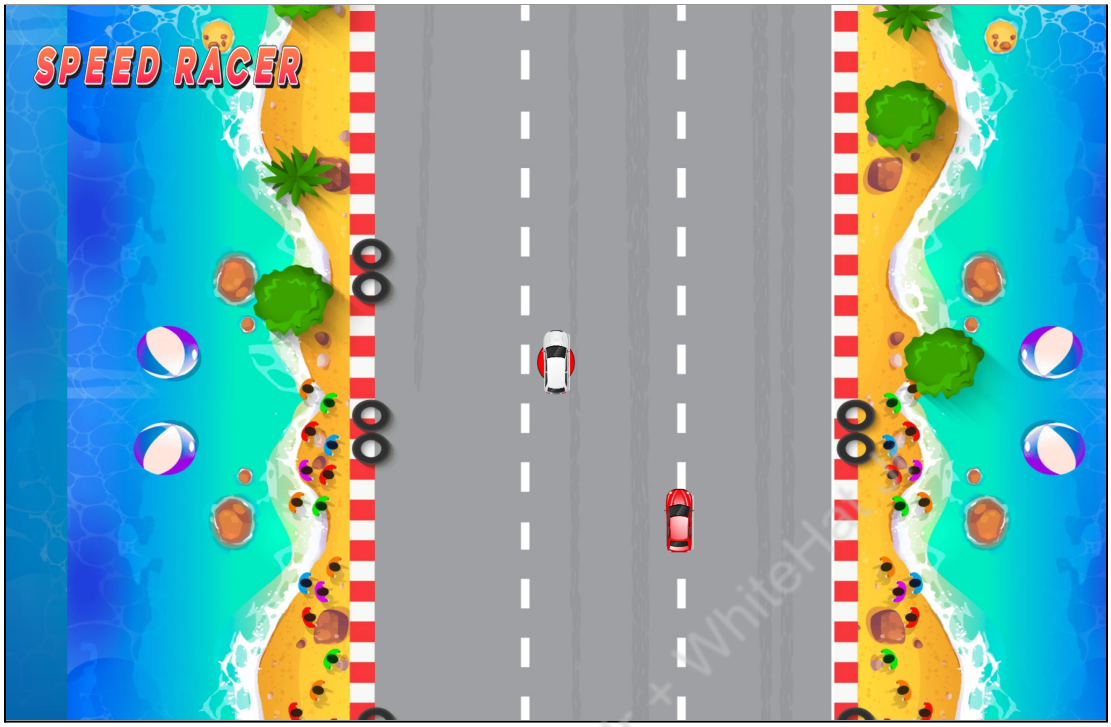
- Identify the current player by matching **index** with **player.index** inside the condition - **if(index===player.index).**

```
if (index === player.index) {
  stroke(10);
  fill("red");
  ellipse(x, y, 60, 60);
```

7. Use the game camera on active players.
   - Without a game camera, when the cars move beyond the screen, they can't be seen.
   - Secondly, all the players see the same thing in the game.
   - The game camera allows us to change how and from where we are viewing the game.
   - Focus the game camera on each player's car.
   - Set the camera position in the game differently for each player.

```
    //use data form the database to display the cars in x and y direct
    var x = allPlayers[plr].positionX;
    var y = height - allPlayers[plr].positionY;

    cars[index - 1].position.x = x;
    cars[index - 1].position.y = y;

    if (index === player.index) {
      stroke(10);
      fill("red");
      ellipse(x, y, 60, 60);

      this.handleFuel(index);
      this.handlePowerCoins(index);

      // Changing camera position in y direction
      camera.position.x = width/2
      camera.position.y = cars[index - 1].position.y;
    }
  }
```

OUTPUT:



What's next?

In the next class, you will create a reset button to reset the values in the database. You will also move the car left & right.

EXTEND YOUR KNOWLEDGE:

1. Know more about for-in loop from the following link created by Mozilla and individual contributors:
   https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/for...in