Game Camera and Display Size

What is our GOAL for this MODULE?

The goal of this module is to learn to adjust the game camera to focus on the current player.

What did we ACHIEVE in the class TODAY?

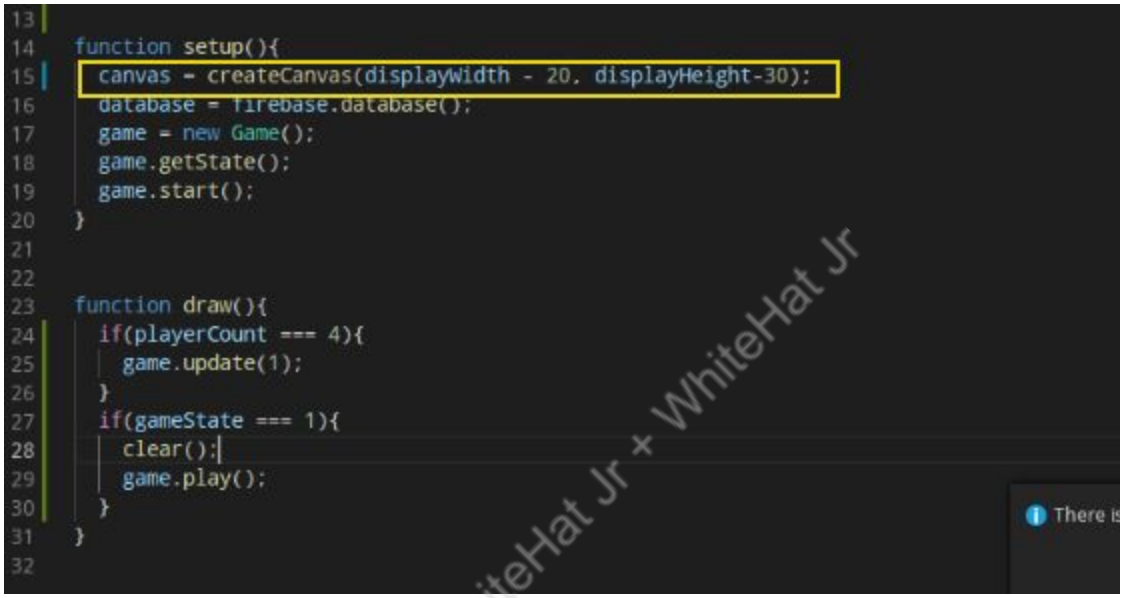
- We used the data from the database to design the car racing game when the game is in play state.
- We used Game Camera to focus the game on the active player in the game.
- We were able to adjust the game canvas to the size of the display device.

Which CONCEPTS/CODING BLOCKS did we cover today?

- Camera positions
- Adjusting the camera size

How did we DO the activities?

Used the data from the database and sprites to design the racing game in action.
Designed the game in a given display size using displayWidth and displayHeight. In P5, using 'displayWidth' and 'displayHeight' will automatically capture the device size on the device in which the program is running.
Used 'displayWidth' and 'displayHeight' in the code to create the canvas to fill the browser.



```
13
14  function setup(){
15    canvas = createCanvas(displayWidth - 20, displayHeight-30);
16    database = firebase.database();
17    game = new Game();
18    game.getState();
19    game.start();
20  }
21
22
23  function draw(){
24    if(playerCount === 4){
25      game.update(1);
26    }
27    if(gameState === 1){
28      clear();
29      game.play();
30    }
31  }
32
```

The positions of the form elements also needed to be adjusted so that the buttons, input box etc. were visible in the correct places.

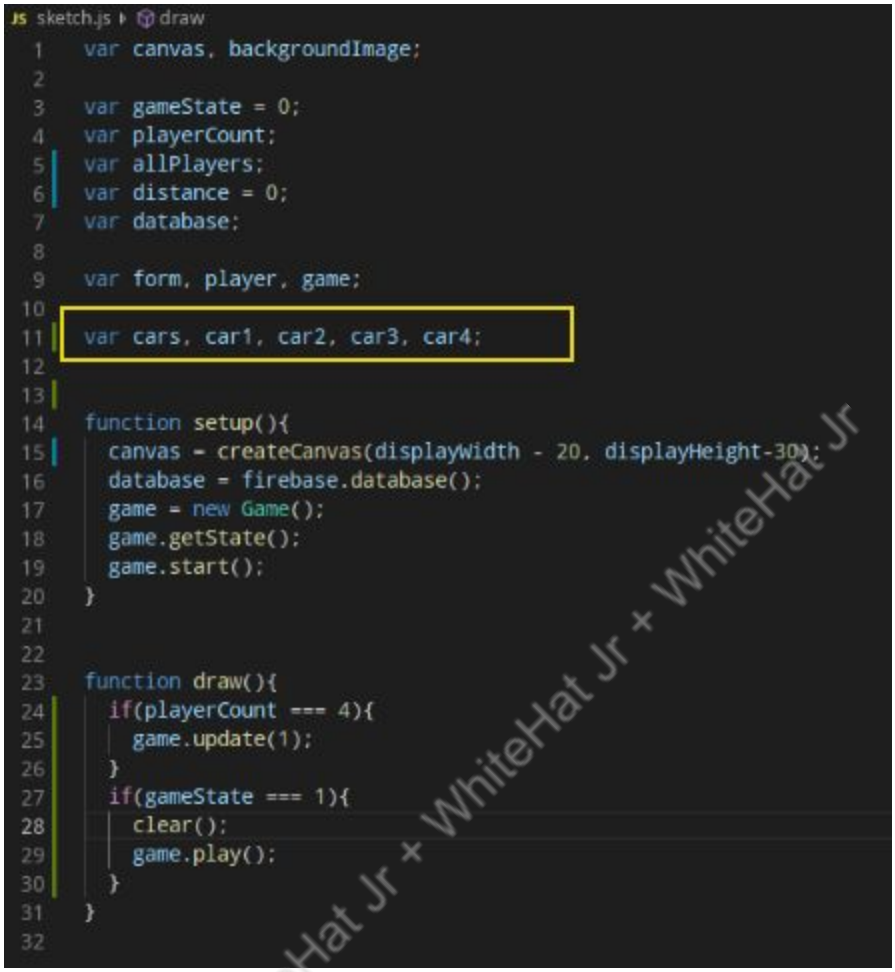
```
    this.input = createInput("Name");
    this.button = createButton('Play');
    this.greeting = createElement('h2');
    this.title = createElement('h2');
  }
  hide(){
    this.greeting.hide();
    this.button.hide();
    this.input.hide();
    this.title.hide();
  }

  display(){
    this.title.html("Car Racing Game");
    this.title.position(displayWidth/2 - 50, 0);

    this.input.position(displayWidth/2 - 40 , displayHeight/2 - 80);
    this.button.position(displayWidth/2 + 30, displayHeight/2);

    this.button.mousePressed(()=>{
      this.input.hide();
      this.button.hide();
      player.name = this.input.value();
      playerCount+=1;
      player.index = playerCount;
      player.update();
      player.updateCount(playerCount);
      this.greeting.html("Hello " + player.name)
      this.greeting.position(displayWidth/2 - 70, displayHeight/4);
    });

  }
}
```

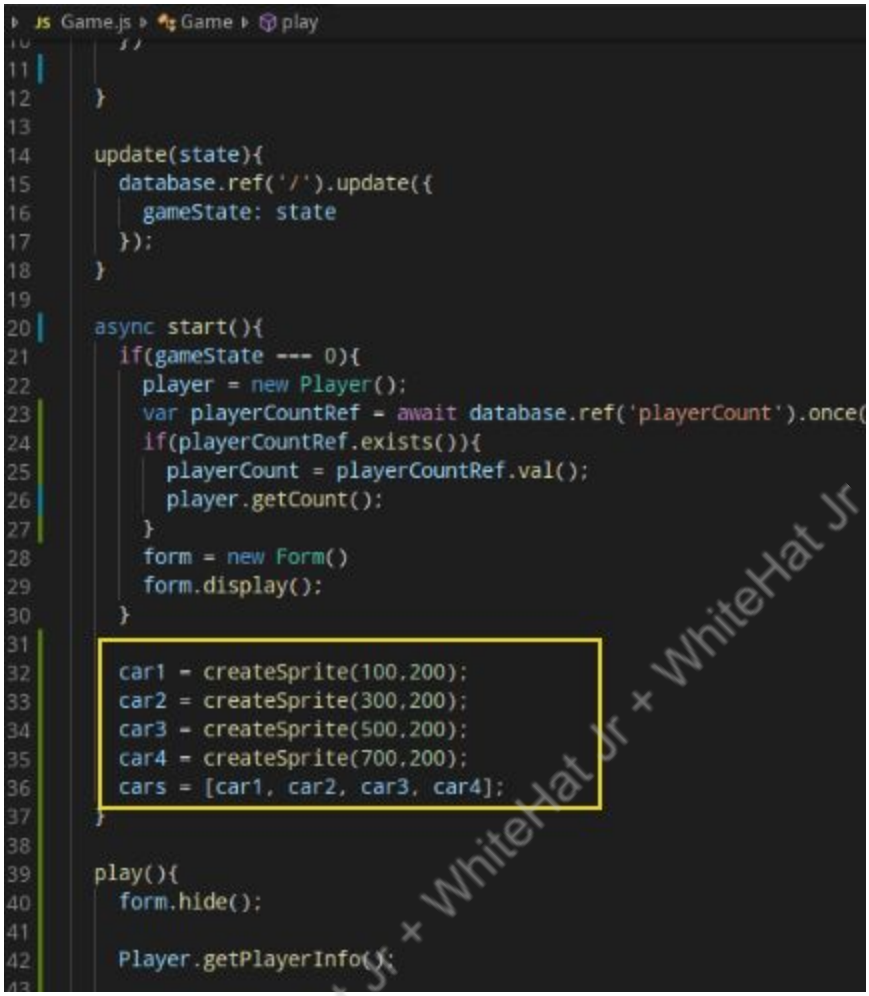
There is an available update.

Download Now

- Used data from the database to design the car racing game.
- Used Game Camera to focus on the player.

To create a simple car racing game inside the play() function in Game.js:





The code to draw the 4 rectangular car sprites at the bottom of the screen:

```
    cars = [car1, car2, car3, car4];
}

play(){
  form.hide();

  Player.getPlayerInfo();

  if(allPlayers !== undefined){
    //var display_position = 100;

    //index of the array
    var index = 0;

    //x and y position of the cars
    var x = 0;
    var y;

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

      //position the cars a little away from each other in x direction
      x = x + 200;
      //use data form the database to display the cars in y direction
      y = displayHeight - allPlayers[plr].distance;
      cars[index-1].x = x;
      cars[index-1].y = y;

      if (index === player.index){
        cars[index - 1].shapeColor = red;

      }
```

To give different color to the player active in the browser:

```
36    cars = [car1, car2, car3, car4];
37  }
38
39  play(){
40    form.hide();
41
42    Player.getPlayerInfo();
43
44    if(allPlayers !== undefined){
45      //var display_position = 100;
46
47      //index of the array
48      var index = 0;
49
50      //x and y position of the cars
51      var x = 0;
52      var y;
53
54      for(var plr in allPlayers){
55        //add 1 to the index for every loop
56        index = index + 1 ;
57
58        //position the cars a little away from each other in x direction
59        x = x + 200;
60        //use data form the database to display the cars in y direction
61        y = displayHeight - allPlayers[plr].distance;
62        cars[index-1].x = x;
63        cars[index-1].y = y;
64
65        if (index === player.index){
66          cars[index - 1].shapeColor = red;
67
68        }
```

To set camera position for each player in the game:

```
38
39  play(){
40    form.hide();
41
42    Player.getPlayerInfo();
43
44    if(allPlayers !== undefined){
45      //var display_position = 100;
46
47      //index of the array
48      var index = 0;
49
50      //x and y position of the cars
51      var x = 0;
52      var y;
53
54      for(var plr in allPlayers){
55        //add 1 to the index for every loop
56        index = index + 1 ;
57
58        //position the cars a little away from each other in x direction
59        x = x + 200;
60        //use data form the database to display the cars in y direction
61        y = displayHeight - allPlayers[plr].distance;
62        cars[index-1].x = x;
63        cars[index-1].y = y;
64
65        if (index === player.index){
66          cars[index - 1].shapeColor = red;
67          camera.position.x = displayWidth/2;
68          camera.position.y = cars[index-1].y
69        }
```


What's NEXT?

In the next class, you will be learning about replacing the sprites with real cars of their choice.

EXTEND YOUR KNOWLEDGE:

You can try changing the positions to know the difference between the camera angles in a game.