

GAME PLAYERS REGISTRATION IN THE DATABASE



What is our GOAL for this MODULE?

We used our knowledge about OOPs programming to change the behavior of gameState when all the players are logged in.

What did we ACHIEVE in the class TODAY?

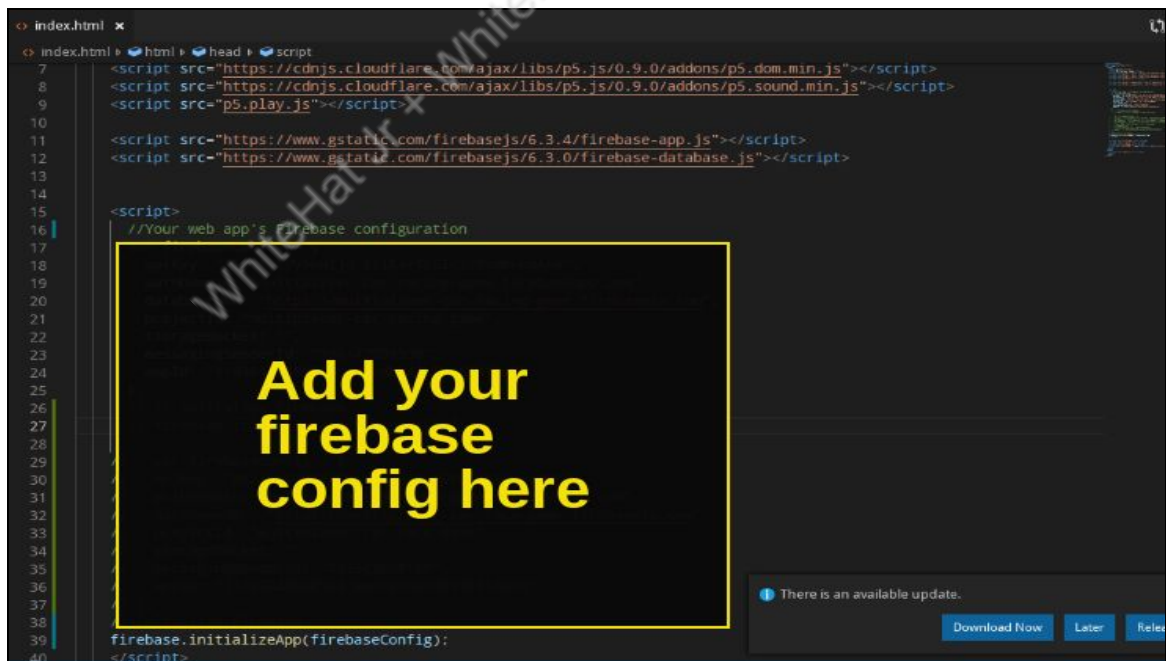
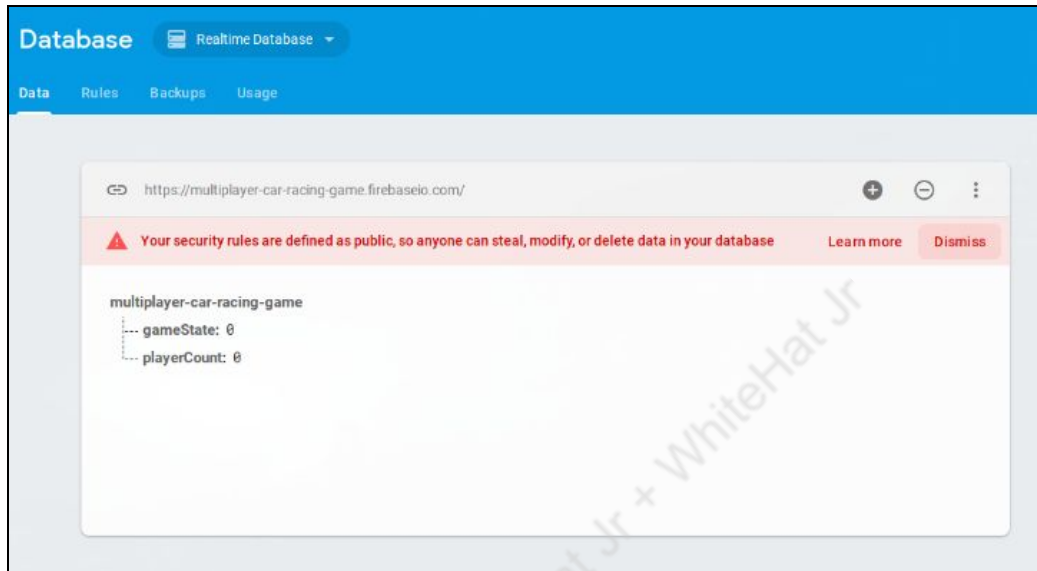
- Created the form which accepts the registration of only 4 players.
- Changed gameStart as soon the player count reaches four.
- Displayed the Game Start message when gameState is one.

Which CONCEPTS/CODING BLOCKS did we cover today?

- The concept of OOPs programming.
- Firebase database.
- Game states.

How did we DO the activities?

1. Change the existing database with reset value of playerCount and gameState.



2. Create a player class with **getCount()** and **updateCount()** function.

```

js > JS Player.js > ...
1  class Player {
2    constructor(){
3
4    }
5
6    getCount(){
7      var playerCountRef = database.ref('playerCount');
8      playerCountRef.on("value", (data)=>{
9        playerCount = data.val();
10     })
11   }
12
13   updateCount(count){
14     database.ref('/').update({
15       playerCount: count
16     });
17   }}
18
  
```

3. Include mousePressed event to the Form.

```

5 Form.js > ...
class Form {
  constructor() {
    this.input = createInput("Name");
    this.button = createButton('Play');
    this.greeting = createElement('h2');
  }
  hide(){
    this.greeting.hide();
    this.button.hide();
    this.input.hide();
  }
  display(){
    var title = createElement('h2')
    title.html("Car Racing Game");
    title.position(130, 0);

    this.input.position(130, 160);
    this.button.position(250, 200);

    this.button.mousePressed(()=>{
      this.input.hide();
      this.button.hide();
      player.name = this.input.value();

      this.greeting.html("Hello " + player.name)
      this.greeting.position(130, 100);
    });
  }
}
  
```

4. Increment the player count and update the database whenever the player registers the form.

```
constructor() {  
  this.input = createInput("Name");  
  this.button = createButton('Play');  
  this.greeting = createElement('h2');  
}  
  
hide(){  
  this.greeting.hide();  
  this.button.hide();  
  this.input.hide();  
}  
  
display(){  
  var title = createElement('h2')  
  title.html("Car Racing Game");  
  title.position(130, 0);  
  
  this.input.position(130, 160);  
  this.button.position(250, 200);  
  
  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(130, 100);  
  });  
}
```

5. Display the text message for game start inside the play function.

```
JS Game.js > ...
1  class Game {
2    constructor(){}
3
4    getState(){
5      var gameStateRef = database.ref('gameState');
6      gameStateRef.on("value",(data)=>{
7        gameState = data.val();
8      })
9    }
10
11    update(state){
12      database.ref('/').update({
13        gameState: state
14      });
15    }
16
17    start(){
18      if(gameState === 0){
19        //player = new Player();
20        //player.getCount();
21        form = new Form()
22        form.display();
23      }
24    }
25
26    play(){
27      form.hide();
28      textSize(30);
29      text("Game Start", 120, 100)
30    }
31  }
```

6. Include async/await function to wait for the promises inside the game class.

```
class Game {
  constructor(){}

  getState(){
    var gameStateRef = database.ref('gameState');
    gameStateRef.on("value",(data)=>{
      gameState = data.val();
    })
  }

  update(state){
    database.ref('/').update({
      gameState: state
    });
  }

  async start(){
    if(gameState === 0){
      player = new Player();
      var playerCountRef = await database.ref('playerCount').once("value");
      if(playerCountRef.exists()){
        playerCount = playerCountRef.val();
        player.getCount();
      }
      form = new Form()
      form.display();
    }
  }

  play(){
    form.hide();
    textSize(30);
    text("Game Start", 120, 100)
  }
}
```

7. Modify sketch.js to write conditions to change the game state and call **play()** function.

```
8
9  var form, player, game;
10
11
12  function setup(){
13    canvas = createCanvas(400,400);
14    database = firebase.database();
15    game = new Game();
16    game.getState();
17    game.start();
18  }
19
20
21  function draw(){
22    if(playerCount === 4){
23      game.update(1);
24    }
25    if(gameState === 1){
26      clear();
27      game.play();
28    }
29  }
30
```

Car Racing Game

Game Start

What's NEXT?

In the next class, you will be learning about adding the distance when the mouse key is pressed and highlight the name who covered the maximum distance.

EXTEND YOUR KNOWLEDGE:

1. You can learn about firebase from their official docs:
<https://firebase.google.com/docs/firestore>.

WhiteHat Jr + WhiteHat Jr + WhiteHat Jr