PROFESSIONAL

MYQUIZ GAME

INSTRUCTIONS:

Goal of the Project:

In Class 37, you have structured a car racing game, created more properties and functions in each class, given game states, and displayed all the players with their distance scores.

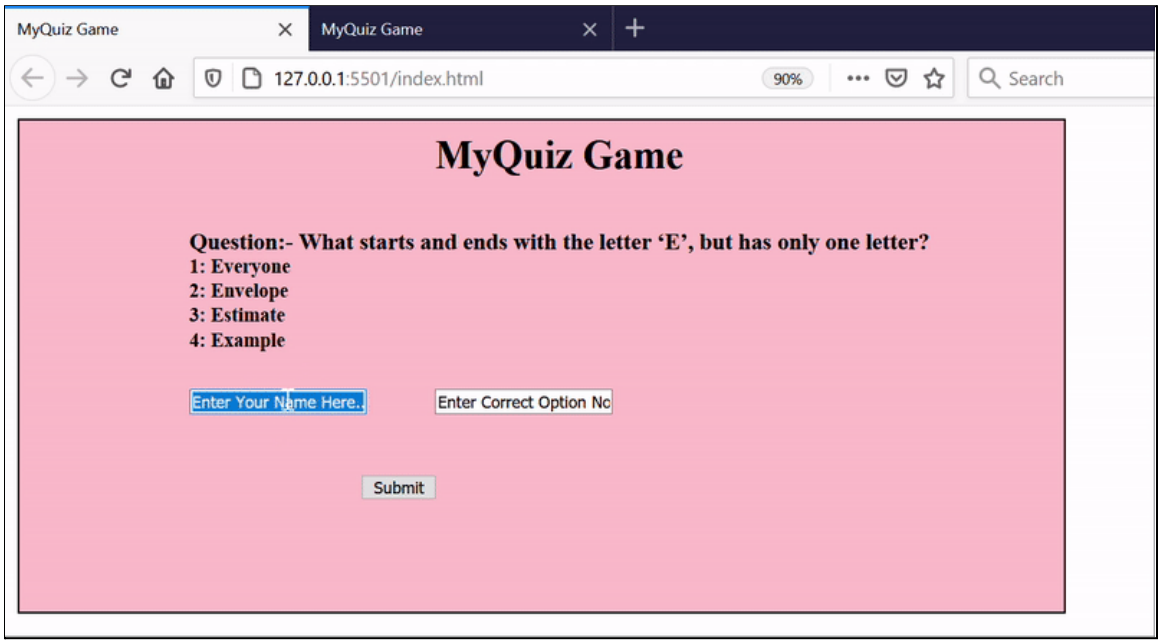
In this project, you will apply what you have learned in the class to create a two player quiz game and store their response for the quiz question in the database.

Story:

Prakriti loves asking quizzes. And she always tries to find unique questions and then ask different people. But now she is thinking of creating her own multiplayer quiz game where she can ask a quiz question to different people at the same time.

Can you help her in creating the game?

See a video of this in action.



PROFESSIONAL

MYQUIZ GAME

***This is just for your reference. We expect you to apply your own creativity in the project.**

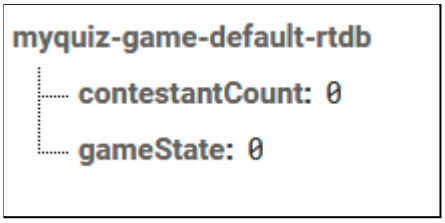
Getting Started:

1. Download a project template here: Project Template.
2. **Unzip** this folder.
3. Rename the unzipped folder as **Project 37.**
4. **Import** this folder **into VS Code**.
5. Start editing your code in **sketch.js**.

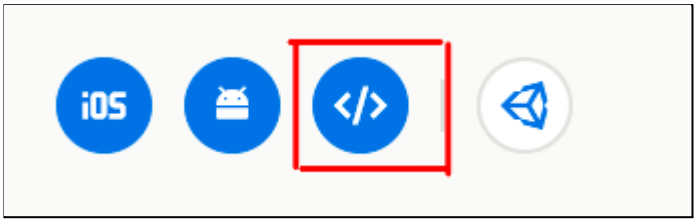
Specific Tasks to complete the Project:

1. Setup **Firebase** for the Project.
    - Go to your firebase console and click on **Create a Project**.
    - Enter the name of the Project as **MyQuiz Game**.
    - Accept terms and click on Continue.
    - **Disable the Google Analytics** option.
    - Click on Create Project.
    - In the left-hand side panel, click on **Database**.
    - Under **Realtime Database**, click on **Create Database**.
    - To create a database in test mode, click on **start in test mode** and click on **Enable**.

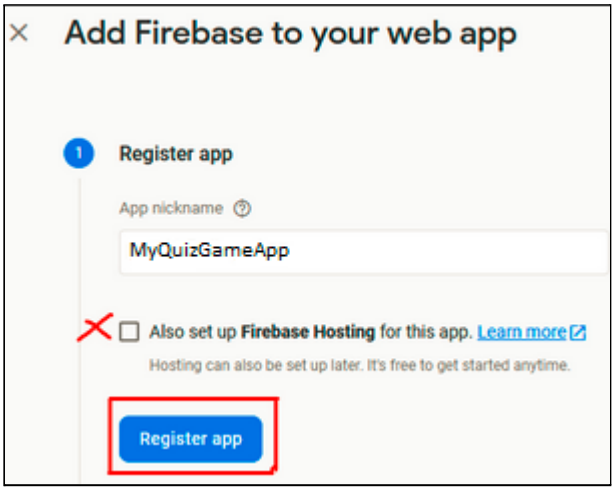
- Create a node in the Database as a **gameState** and **contestantCount**. Set 0 as the value for both the nodes.



- Click on **Project Overview** and select the **Web** option.



- Register the app and don't check the Firebase hosting option.



- Add **Firebase SDK**.
  - Copy the content by clicking on the icon to the bottom right and paste in the **index.html** file along with an **src library** for the Firebase database in **VS**.

```
<!-- The core Firebase JS SDK is always required and must be listed
<script src="https://www.gstatic.com/firebasejs/8.2.2/firebase-app.j

<!-- TODO: Add SDKs for Firebase products that you want to use
     https://firebase.google.com/docs/web/setup#available-libraries

<script>
  // Your web app's Firebase configuration
  var firebaseConfig = {
    apiKey: "AIzaSyAlYTDtmHd5ebKMonZUo7lnweOzDdBo40E",
    authDomain: "myquiz-game.firebaseapp.com",
    databaseURL: "https://myquiz-game-default-rtdb.firebaseio.com",
    projectId: "myquiz-game",
    storageBucket: "myquiz-game.appspot.com",
    messagingSenderId: "135491178213",
    appId: "1:135491178213:web:7fd97cb4bc6b0b0b4c0e0c"
  };
  // Initialize Firebase
  firebase.initializeApp(firebaseConfig);
</script>
```

2. In **Quiz.js** file:
    - Inside **play()** write code for the following instructions:
        1. Inside this function, hide the question class elements like inputs, button, and title.

```
play(){
  //write code here to hide question elements

}
```

        2. Change the background color to yellow or the color of your choice.

```
play(){
  //write code here to hide question elements

  //write code to change the background color here

}
```

        3. Add a heading for showing the result of the Quiz and give some size to the text.

```
play(){
  //write code here to hide question elements

  //write code to change the background color here

  //write code to show a heading for showing the result of Quiz


}
```

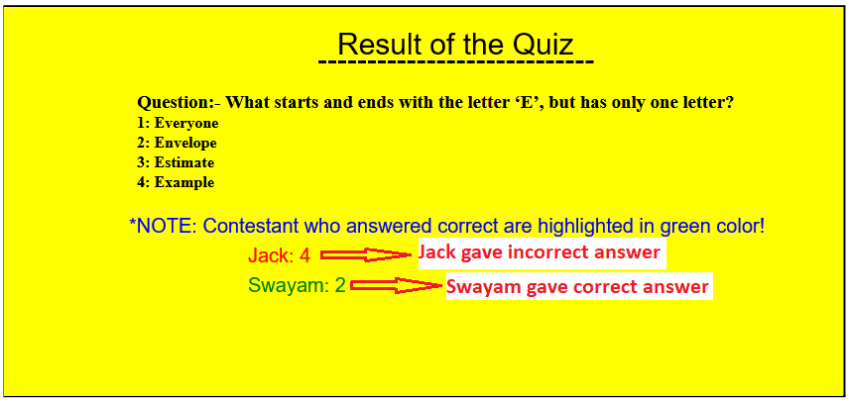
4. Call the **getContestantInfo( )** of Contestant class.
5. Write a condition to check if allContestants data is not equal to **undefined**.

```
play(){
  //write code here to hide question elements

  //write code to change the background color here

  //write code to show a heading for showing the result of Quiz

  //call getContestantInfo( ) here

  if(allContestants !== undefined){

  }
}
```

6. Then add a Note to help contestants understand the result. As shown below:

```
if(allContestants !== undefined){
  fill("Blue");
  textSize(20);
  text("*NOTE: Contestant who answered correct are highlighted in green color!",130,230);
}
```

7. Write a loop statement and check the condition if the correct answer is equal to the contestant's answer.
8. If the answer is correct, then show the contestant's name and his/her answer in **green** color and if the answer is wrong, then highlight it with **red** color.



3. Make sure the project works before you submit it.

*Refer to the images given above for reference.

Submitting the Project:

1. **Upload** your completed project to your own GitHub account.
2. Enable **GitHub** pages for the repository.
3. Copy and paste the link to the GitHub pages in the Student Dashboard against the correct class number.

Hints for the project:

1. You can use the code given below as a reference to check the answers and highlight the contestant:

```
for(var plr in allContestants){
  var correctAns = "2";
  if (correctAns === allContestants[plr].answer)
    fill("Green")
  else
    fill("red");
```

REMEMBER... Try your best, that's more important than being correct.
After submitting your project your teacher will send you feedback on your work.

——————— xxx ——————— xxx ——————— xxx ——————— xxx ——————— xxx ———————