

FastAPI Formula1 App Documentation

Def.validateFirebaseToken(id_token)

Validates a Firebase authentication token to determine if the user is logged in. If the token is missing or invalid, the function returns a failure response. If valid, it confirms authentication and allows the user to proceed.

Def.root(request)

Retrieves the Firebase token from cookies, validates it, and renders the index template while passing the login status to the frontend.

Def.login(request)

Renders and returns the login.html template

Def.view_driver(request)

Fetches a list of all drivers stored in the Firestore database and returns the data for display in the application.

Def.delete_driver(request)

Deletes a driver from Firestore based on the provided driver ID. Before deletion, it checks user authentication to ensure only authorized users can perform this action.

Def.add_driver(request)

Loads a form where users can enter driver details and select a team from available options before adding the driver to the Firestore database.

Def.add_driver_post(request)

Processes a submitted form containing driver details. If the driver does not already exist in the database, they are added with the provided attributes, including team and performance statistics.

Def.driver_details(request)

Retrieves and returns details of a specific driver, including their associated team. This function ensures all relevant driver information is displayed accurately.

Def.edit_driver(request)

Loads a form pre-filled with driver details, allowing the user to modify attributes such as name, age, and team before updating the database.

Def.edit_driver_post(request)

Processes submitted edits for a driver. The updated details are saved in Firestore, ensuring all changes are reflected accurately in the system.

Def.filter_driver(request)

Displays options for filtering drivers based on specific attributes, such as race wins, pole positions, or championship titles.

Def.filter_driver_post(request)

Processes user-defined filtering criteria to retrieve and display drivers who meet the specified conditions.

Def.view_team(request)

Retrieves and displays all racing teams stored in Firestore, allowing users to browse through the available teams.

Def.team_details(request)

Fetches and returns detailed information about a specific team, such as year founded, total race wins, total pole positions.

Def.add_team(request)

Processes a submitted form containing team details. If the team does not already exist, it is added to Firestore with attributes such as founding year and championship titles.

Def.edit_team(request)

Loads a form pre-filled with team details, allowing the user to update attributes such as name, founding year, and past achievements.

Def.edit_team_post(request)

Processes submitted edits for a team, updating the relevant attributes in Firestore to reflect the latest changes.

Def.delete_team(request)

Deletes a team from Firestore based on the provided team ID. The function verifies user authentication before proceeding with deletion.

Def.filter_team(request)

Displays options for filtering teams based on attributes such as championship titles, race wins, and pole positions.

Def.filter_team_post(request)

Processes user-defined filtering criteria to retrieve and display teams that meet the specified conditions.

Def.compare_drivers(request)

Displays a selection page where users can choose two drivers to compare based on their statistics.

Def.compare_drivers_post(request)

Retrieves the statistics of two selected drivers and compares attributes such as race wins, points, and championship titles.

Def.compare_teams(request)

Displays a selection page where users can choose two teams to compare their performance history.

Def.compare_teams_post(request)

Retrieves the statistics of two selected teams and compares attributes such as constructor championships, pole positions, and race victories.

Database Models:

Drivers Collection:

- **name:** Stores the driver's name.
- **age:** Stores the age of the driver.
- **total_pole_positions:** Stores the total number of pole positions.
- **total_race_wins:** Stores the total number of race wins.
- **total_points:** Stores the total number of championship points.
- **total_world_titles:** Stores the total number of world titles.
- **total_fastest_laps:** Stores the total number of fastest laps.
- **team:** Stores the team ID the driver belongs to.

Teams Collection:

- **team_name:** Stores the name of the team.
- **year_founded:** Stores the founding year of the team.
- **total_pole_positions:** Stores the total number of pole positions earned by the team.
- **total_race_wins:** Stores the total number of races won by the team.
- **total_constructor_titles:** Stores the total number of constructor championships won by the team.
- **finishing_position:** Stores the team's last season finishing position.

