Assignment Seventeen

Please read:

- This is your seventeen assignment which you need to submit on the LMS. This is an assignment on Redux.

- Work on the questions given below and be ready with your solution. You have to submit your codesandbox link below on this page.

- Late Submission: The deadline to submit this assignment is 19th February 2024, 6:00PM IST.

Important Instructions:

1. Make sure that you follow the rules as it will help you build your muscle.

2. Do not copy from someone else as that would be cheating.

challenge

1. Introduction:

The Fitness Tracker is a web application designed to help users track their fitness goals, including exercise routines, food intake, and fitness goals. It provides a dashboard with summarized data, allows users to add and remove exercises, record their food consumption, and set fitness goals.

2. Features:

2.1 Dashboard:

Display a summary of fitness metrics.

- Total Calories Burned
- Total Calories Consumed
- Total Calories Goal
- Remaining Calories to Goal

2.2 Exercise Tracking:

- Allow users to add exercises with details such as:

  - Exercise Name
  - Duration (in minutes)
  - Calories Burned (auto-calculated based on exercise type)
- Display a list of added exercises.
- Allow users to remove exercises from the list.

2.3 Food Tracking:

- Allow users to add food items with details such as:

  - Food Name
  - Calories
  - Protein (in grams)
  - Carbohydrates (in grams)
  - Fat (in grams)
- Display a list of added food items.
- Allow users to remove food items from the list.

2.4 Goal Tracking:

- Allow users to set fitness goals with details such as:

  - Goal Name
  - Goal Description
  - Target Date
  - Target Calories Value
  - Status (e.g., In Progress, Achieved, Abandoned)
- Display a list of set fitness goals.

- Allow users to remove fitness goals from the list.

2.5 Navigation:

- Provide a navigation menu to access different sections of the app (Dashboard, Exercises, Food, Goal Tracker).

3. User Interfaces:

- Dashboard:

  - Display fitness metrics.
  - Include navigation links to other sections.
- Exercises:

  - Include an input form for adding exercises.
  - Display a list of added exercises with the option to remove them.
- Food:

  - Include an input form for adding food items.
  - Display a list of added food items with the option to remove them.
- Goal Tracker:

  - Include an input form for adding fitness goals.
  - Display a list of set fitness goals with the option to remove them.

4. API Endpoints:

1. `/api/exercises`

- GET `/api/exercises`

  - Description: Fetches a list of exercises.
  - Response: JSON array of exercise objects, including exercise name, duration, and calories burned.
- POST `/api/exercises`

  - Description: Adds a new exercise to the list.
  - Request Body: JSON object with exercise details:

- - Exercise Name (string)
    - Duration (number, in minutes)
    - Calories Burned (number, auto-calculated based on exercise type)
  - Response: JSON object of the newly added exercise with an assigned unique ID.
- DELETE `/api/exercises/:exerciseId`

  - Description: Removes an exercise from the list by its unique ID.
  - Response: Status code 204 (No Content) on successful removal.

2. `/api/food`

- GET `/api/food`

  - Description: Fetches a list of food items.
  - Response: JSON array of food item objects, including food name, calories, protein, carbohydrates, and fat content.
- POST `/api/food`

  - Description: Adds a new food item to the list.
  - Request Body: JSON object with food item details:

    - Food Name (string)
    - Calories (number)
    - Protein (number, in grams)
    - Carbohydrates (number, in grams)
    - Fat (number, in grams)
  - Response: JSON object of the newly added food item with an assigned unique ID.
- DELETE `/api/food/:foodId`

  - Description: Removes a food item from the list by its unique ID.
  - Response: Status code 204 (No Content) on successful removal.

3. `/api/goals`

- GET `/api/goals`

  - Description: Fetches a list of fitness goals.
  - Response: JSON array of fitness goal objects, including goal name, description, target date, target calories value, and status.
- POST `/api/goals`

  - Description: Adds a new fitness goal to the list.
  - Request Body: JSON object with fitness goal details:

- Goal Name (string)
- Goal Description (string)
- Target Date (date)
- Target Calories Value (number)
- Status (string, e.g., "In Progress", "Achieved", "Abandoned")
  - Response: JSON object of the newly added fitness goal with an assigned unique ID.
- DELETE `/api/goals/:goalId`

  - Description: Removes a fitness goal from the list by its unique ID.
  - Response: Status code 204 (No Content) on successful removal.