Assignment No: 02

Web Engineering

Name: **Sania Zaheer**
Roll Number: **22i 2674**
Section Name: **SE C**
Instruction: **Miss Laiba Imran**

Project: Weather Dashboard with Table and Chatbot Assistant

1. Overview:

This project is a web based weather dashboard that allows users to search for current weather conditions and view forecasts for various cities. The application consists of two pages:
1. **Dashboard**: Displays weather data using charts and widgets.
2. **Table Page**: Displays the forecast in tabular format, with sorting, filtering, and chatbot functionality.

2. Features

2.1. Common Features:

1. **City Search:** Users can search for weather information by entering a city name.
2. **Weather Data:** Displays current temperature, humidity, wind speed, and weather conditions.
3. **Temperature Units:** Toggle between Celsius and Fahrenheit for both the current weather and forecast.
4. Integration with OpenWeatherMap API for real time weather data.
5. **Chart.js:** Used to display weather conditions in visual format such as bar charts, line charts, and doughnut charts.

2.2. Dashboard Page (Dashboard.html):

1. **Charts:** Temperature Bar Chart, Weather Conditions Doughnut Chart, Temperature Line Chart
2. **5 Day Forecast:** Displays weather forecasts for five days with icons and descriptions.

2.3. Table Page (Table.html):

1. **Weather Forecast Table :** Displays forecast data such as date, time, temperature, humidity, and wind speed.
2. **Sorting and Filtering :** Sort data by temperature and filter based on weather conditions like rain.
3. **Pagination :** Paginate through weather data to view more entries.

4. **Chatbot Assistant :** The chatbot can answer weather related queries and provide general assistance using Google Generative AI.

3. Project Structure

1. **Dashboard.html:** The main HTML file for the weather dashboard, featuring the charts and layout.
2. **Dashboard.css:** The stylesheet for styling the dashboard, including widgets and responsiveness.
3. **Dashboard.js:** JavaScript file to fetch weather data, handle UI logic, and generate charts.
4. **Table.html :** The HTML file for the table layout, including the weather forecast table and chatbot interface.
5. **Table.css :** The CSS file for the table and chatbot styling, including pagination and filters.
6. **Table.js :** The JavaScript file handling API requests, table operations, and chatbot logic.

4. Prerequisites

To run the project locally, ensure you have:
A web browser (e.g., Chrome, Firefox). Internet access to fetch data from APIs.

Instructions to Run Locally

4.1. Clone or Download the Project:

```bash
git clone https://github.com/your username/weather dashboard.git
```

Or simply download the files as a ZIP and extract them.

4.2. Set up API Keys:

The project uses OpenWeatherMap API and Google Generative AI. Replace the placeholder keys with your actual API keys.

In Dashboard.js and Table.js , update the following lines:

```javascript
const apiKey = 'YOUR_OPENWEATHER_API_KEY';
```

```
const GEMINI_API_KEY = 'YOUR_GOOGLE_GENERATIVE_AI_KEY';
```

4.3. Run the Project:

Open either Dashboard.html or Table.html in a web browser.
You can use [Live Server](https://marketplace.visualstudio.com/items?itemName=ritwickdey.LiveServer) for VSCode or a simple HTTP server like Python:

```bash
python3 -m http.server
```

5. Usage

5.1. Dashboard Page:

1. Open Dashboard.html in the browser.
2. Enter a city name in the search box.
3. Toggle between °C or °F for temperature units.
4. Click Get Weather to view the current weather and a 5 day forecast in chart form.

5.2. Table Page:

1. Open Table.html in the browser.
2. Enter a city name in the search box and select the temperature unit.
3. Click Get Weather to view the forecast in a table.
4. Use the sorting and filtering buttons to manipulate the table.
5. Interact with the Chatbot Assistant by clicking the chatbot icon and typing weather related questions (e.g., "What is the current weather in London?").

5.3. External Dependencies

Chart.js : Used for rendering charts (loaded via CDN).
OpenWeatherMap API : Fetches weather data.
Google Generative AI : Powers the chatbot for general and weather related queries.

5.4. Troubleshooting

City Not Found : Ensure the city name is spelled correctly or verify the API key.

API Limits : The free version of OpenWeatherMap has request limits. If you reach the limit, consider upgrading to a higher plan.

Console Errors : Use the browser's developer console (`F12` or `Ctrl+Shift+I`) for error logs.

With this setup, you should be able to explore the Weather Dashboard and Table functionalities locally. Enjoy using the Weather Dashboard !