***weather app***

**🌤️ Real-Time Weather Website Using WeatherAPI**

**📌 Overview**

This project is a fully responsive web application that provides users with real-time weather information based on the city name they input. It integrates with the [WeatherAPI](https://www.weatherapi.com/) to fetch current weather data such as temperature, condition description, location name, and air quality information. The goal was to build a fast, functional, and easy-to-use frontend-only website using pure **HTML**, **CSS**, and **JavaScript**, while also gaining hands-on experience with **REST APIs** and asynchronous programming.

**🎯 Objectives**

The main objectives of the project were:

* To develop a simple yet effective weather web application using only client-side technologies.
* To understand and implement API integration in a real-world context.
* To apply modern frontend principles including responsive design and user-centric development.
* To explore asynchronous JavaScript with fetch, async/await, and error handling mechanisms.
* To design a clean and user-friendly interface using only HTML and CSS.

**🧠 Key Features**

* **City Search**: Users can type in the name of any city globally to get current weather information.
* **Real-Time Weather Data**: On search, the app fetches live weather data including temperature (in Celsius), location (city and country), and weather condition.
* **API Integration**: Utilizes the WeatherAPI’s /current.json endpoint with a dynamic query string based on user input.
* **Error Handling**: Displays user-friendly error messages if the city is not found or if the API fails to respond.
* **Responsive Design**: Fully optimized for both desktop and mobile users with simple, clean layout and intuitive interaction.
* **Clean UI/UX**: Focuses on accessibility and ease of use, with a minimalistic design that ensures content is clear and visually appealing.

**🔧 Technologies Used**

| **Area** | **Tools / Languages** |
| --- | --- |
| **Frontend** | HTML5, CSS3, JavaScript (ES6) |
| **API** | [WeatherAPI](https://www.weatherapi.com/) |
| **Versioning** | Git & GitHub |
| **Deployment** | (Optional) GitHub Pages / Netlify / Vercel |

**🔎 API Used**

**Endpoint**:  
http://api.weatherapi.com/v1/current.json

**Example Request**:  
http://api.weatherapi.com/v1/current.json?key=YOUR\_API\_KEY&q=London&aqi=yes

**Response Data Includes**:

* location.name, location.country
* current.temp\_c
* current.condition.text
* current.air\_quality.pm2\_5 (optional for AQI)

**🖼️ User Interface Highlights**

* **Input Field**: Allows user to type a location (city name)
* **Button**: Triggers fetch call to API
* **Results Section**: Displays weather information dynamically using DOM manipulation
* **Styling**: Modern look with responsive behavior using CSS media queries and flexbox

**📦 Folder Structure**

plaintext

CopyEdit

weather-app/

│

├── index.html # Main HTML file

├── style.css # Styling for layout and design

└── script.js # JavaScript for API integration and DOM logic

**🚀 Deployment**

You can host this project using:

* **GitHub Pages**: Free hosting directly from your GitHub repo
* **Netlify**: One-click deploy and live preview
* **Vercel**: Optimized for frontend apps with CI/CD integration

**✅ Learning Outcomes**

* Gained a deep understanding of how to consume third-party REST APIs using fetch and async/await.
* Learned how to validate and handle user input on the frontend.
* Applied core JavaScript concepts such as DOM manipulation, conditionals, and error handling in a real-world app.
* Practiced responsive web design and mobile-first development using pure CSS.
* Developed a maintainable codebase using separation of concerns between HTML, CSS, and JavaScript.