**Weather Dashboard**

A modern **Weather Dashboard** web application built with **HTML, CSS, and JavaScript**.  
It provides **real-time weather updates, 5-day forecasts, air quality index, favorite cities, and dark/light theme support**.

**Features**

* 🌍 **City Search** – Search weather information by city name.
* 📍 **Geolocation Support** – Fetch weather using the user’s current location.
* 🌡 **Current Weather** – Displays temperature, humidity, wind speed, pressure, and "feels like".
* 📆 **5-Day Forecast** – View upcoming weather trends with icons and temperatures.
* ❤️ **Favorites** – Save and quickly access favorite cities.
* 🎨 **Dark/Light Mode** – Toggle between light and dark themes with persistence (saved in local storage).
* 🔄 **Unit Toggle** – Switch between Celsius (°C) and Fahrenheit (°F).
* 🌫 **Air Quality Index (AQI)** – Displays air quality with color-coded status.
* 🗺 **Map Placeholder** – Integrated map container for future weather map API integration.
* ⚡ **Responsive Design** – Works seamlessly on desktop, tablet, and mobile.
* 🎬 **Smooth Animations** – Fade-in and transition effects for UI elements.

**Project Structure**

weather-dashboard/

├── index.html # Main HTML file with inline CSS & JS

└── README.md # Documentation

**Getting Started**

**1. Clone the Repository**

git clone https://github.com/yourusername/weather-dashboard.git

**2. Open in Browser**

Simply open the index.html file in your favorite browser.  
No build step or server required – it’s **pure client-side**.

**How It Works**

* **State Management**: Maintains city, favorites, temperature unit, and theme in a state object with persistence via localStorage.
* **Weather Data**: Currently uses **mock data** (setTimeout simulates API calls).
  + For real implementation, connect to APIs like OpenWeatherMap or WeatherAPI.
* **Geolocation**: Uses the **browser's navigator.geolocation API** to fetch latitude and longitude.
* **Dynamic Rendering**: Weather info, forecasts, favorites, and AQI are updated dynamically with JavaScript DOM manipulation.

**Customization**

* 🌐 **API Integration**: Replace mock weather data with a real API call.
* 🎨 **Theme**: Customize CSS variables in :root and .dark-mode for branding.
* 🗺 **Map**: Embed an actual interactive weather map using APIs like **Leaflet** or **Google Maps API**.

**Responsive Design**

* **Desktop**: Dashboard layout with sidebar and main content.
* **Tablet/Mobile**: Stacked layout with fluid resizing and optimized controls.

**Future Improvements**

* Real-time API integration.
* Hourly forecast.
* Interactive map with weather overlays.
* Drag-and-drop favorite city ordering.