# WeatherNow — A Responsive Weather Web Application

## Introduction

WeatherNow is a responsive, multi-page weather application developed using HTML, CSS, and JavaScript. The application allows users to search for real-time weather information by city, offering an intuitive and accessible way to stay informed about current weather conditions worldwide.

The project is centered around practical implementation of essential web development concepts, such as:

- Multi-page navigation and layout structuring
- Client-side scripting for API interaction
- Real-time data rendering
- Responsive design for various screen sizes

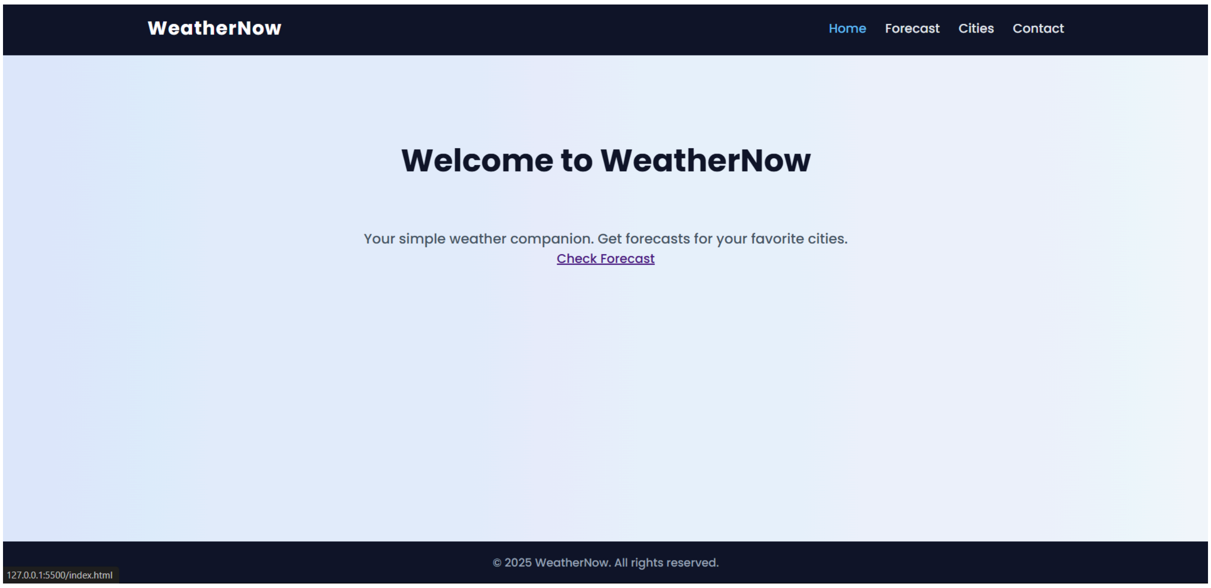
One of the core features of WeatherNow is its integration with the OpenWeatherMap API, which provides live weather data including temperature, conditions (like clear sky, rain, etc.), humidity, and wind speed. The app ensures users can quickly search for any city and receive accurate, up-to-date information through a simple, user-friendly interface.

In addition to the forecast functionality, the project includes structured pages for:

- A Home page that introduces the app and allows navigation to other sections
- A Forecast page which displays the weather and 24hr forecast of the desired city you wish to search for
- A Cities page that offers weather previews for commonly searched cities with a single click
- A Contact page for user feedback or inquiries

This project serves as a demonstration of how static web technologies can be combined with real-time API data to build meaningful, interactive applications.

## Home Page



Welcome to WeatherNow

Your simple weather companion. Get forecasts for your favorite cities.

© 2025 WeatherNow. All rights reserved.

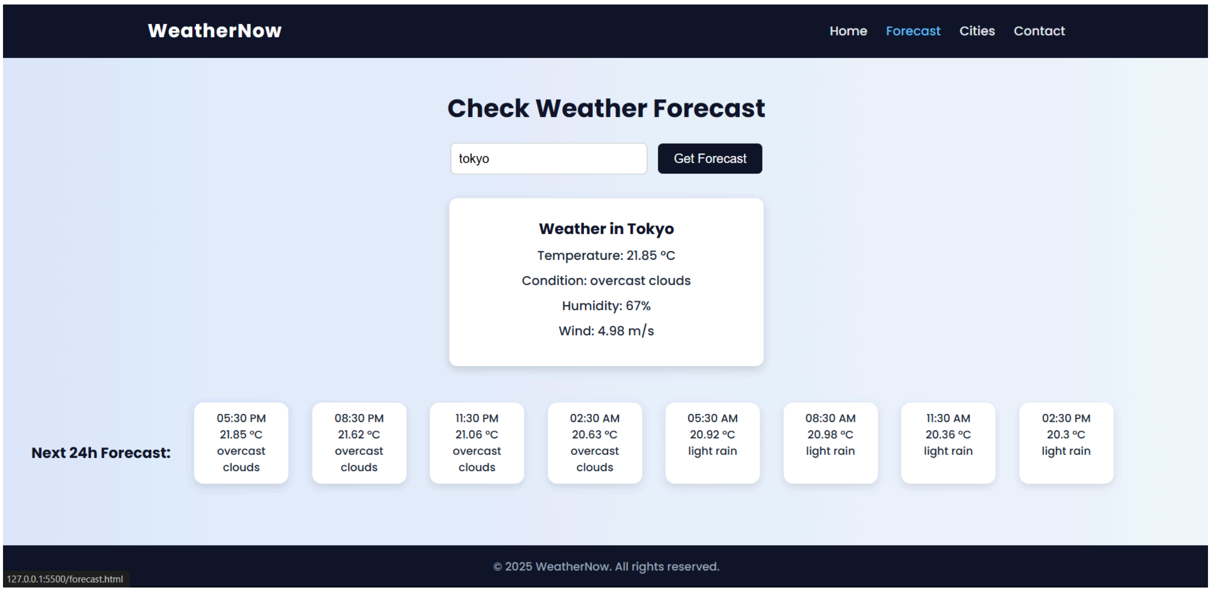
The Home page serves as the introductory interface of the WeatherNow application. It is designed to provide users with a clear and immediate understanding of the app's purpose and how to navigate through it.

Key Features:

- Introductory Message:
  A concise tagline or description is displayed prominently, letting users know that the app provides live weather updates by city.
- Navigation Bar:
  A simple, fixed navigation menu links to the core sections of the application:
  - Forecast — Search for weather by city
  - Cities — View weather for popular preset cities
  - Contact — Send feedback or inquiries
- Call to Action:
  The home page encourages users to head over to the Forecast page to begin interacting with the application.

This page sets the tone for the application by being visually straightforward, easy to use, and functional. It also reinforces the app's overall goal: helping users quickly and easily access accurate weather information.

## Forecast Page



Check Weather Forecast

tokyo

Get Forecast

Weather in Tokyo

Temperature: 21.85 °C

Condition: overcast clouds

Humidity: 67%

Wind: 4.98 m/s

Next 24h Forecast:

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 05:30 PM | 08:30 PM | 11:30 PM | 02:30 AM | 05:30 AM | 08:30 AM | 11:30 AM | 02:30 PM |
| 21.85 °C | 21.62 °C | 21.06 °C | 20.63 °C | 20.92 °C | 20.98 °C | 20.36 °C | 20.3 °C |
| overcast clouds | overcast clouds | overcast clouds | overcast clouds | light rain | light rain | light rain | light rain |

© 2025 WeatherNow. All rights reserved.

127.0.0.1:5500/forecast.html

The Forecast page is the functional core of the WeatherNow application. It allows users to enter the name of any city and retrieve real-time weather data using the OpenWeatherMap API. This page demonstrates how client-side JavaScript can dynamically fetch and display data based on user input.

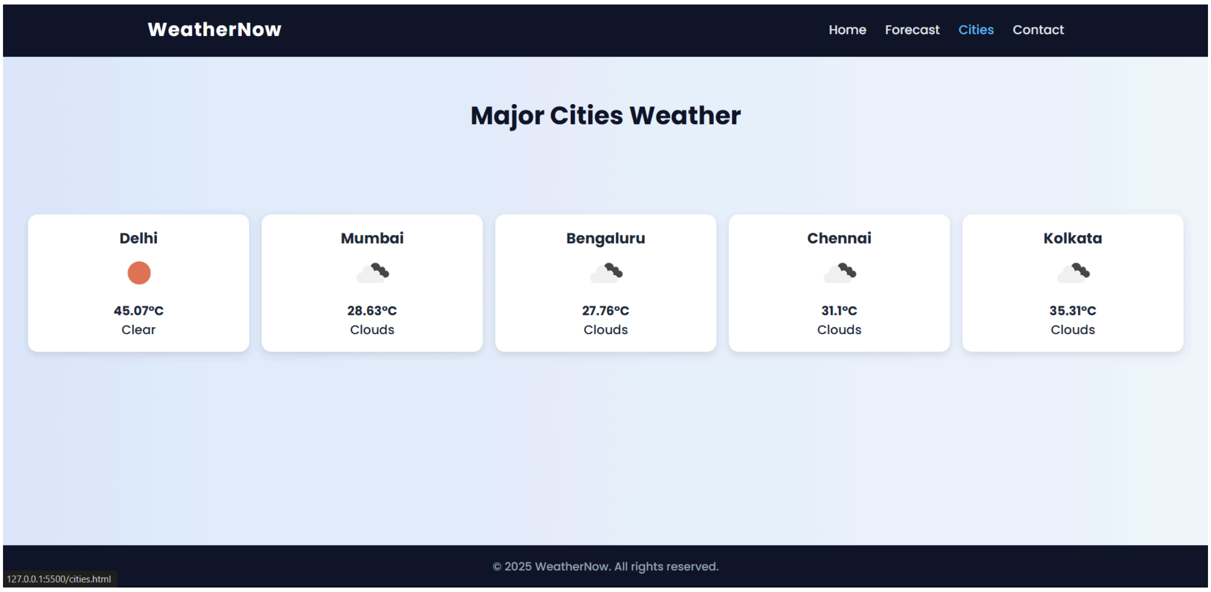
Key Features:

- Search Functionality:
  Users can input the name of any city in a search bar. Upon clicking the “Get Weather” button, the application fetches weather information specific to that location.
- API Integration:
  The application uses the OpenWeatherMap `/weather` endpoint to retrieve live data such as temperature, weather condition, humidity, and wind speed.
- Weather Display Card:
  Fetched data is presented in a styled weather card, providing users with:
  - Temperature (in Celsius)
  - Weather description (e.g., “clear sky”, “light rain”)
  - Humidity percentage

    - Wind speed in meters per second
- Error Handling:
  If a user enters an invalid city name or if there’s a network/API error, the application displays a friendly error message guiding the user to try again.
- Dynamic Rendering:
  JavaScript is used to dynamically insert the weather data into the HTML page, updating the content without needing a page reload.

This page is designed to be functional, interactive, and user-friendly. It ensures that users can quickly access accurate and meaningful weather information with minimal effort.

## Cities Page



Major Cities Weather

| Delhi | Mumbai | Bengaluru | Chennai | Kolkata |
|---|---|---|---|---|
| 45.07°C | 28.63°C | 27.76°C | 31.1°C | 35.31°C |
| Clear | Clouds | Clouds | Clouds | Clouds |

© 2025 WeatherNow. All rights reserved.

The Cities page serves as a curated shortcut for users to quickly view the weather of a set of predefined cities without typing them manually. It enhances usability by offering one-click access to weather reports of popular or frequently searched locations.

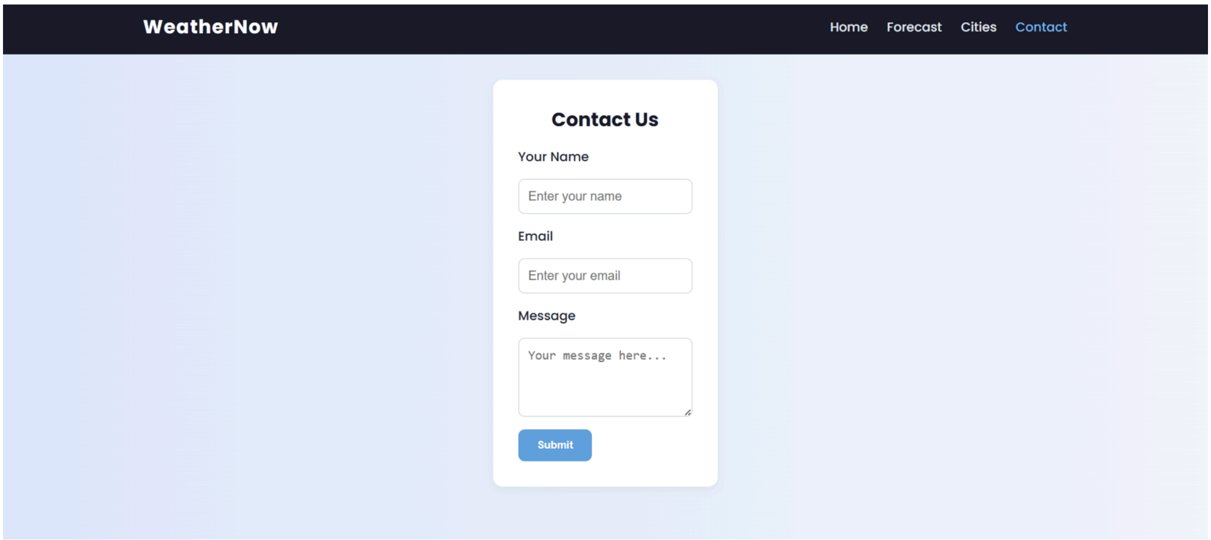
Key Features:

- Displays a grid/list of popular city names (e.g., Chennai, Delhi, Mumbai, Kolkata, Bengluru)
- Clicking a city triggers a weather fetch for that city using the same API used on the Forecast page
- Weather data is dynamically inserted into a container on the page (using JavaScript)
- Reuses core logic from forecast page but applies it to multiple preset cities

Technical Details:

- Each city button or card is hardcoded into the HTML with an one-click event
- Uses the getWeather(cityName) function to fetch and display current weather for the selected city
- Weather display includes:
  - Temperature in Celsius
  - Weather description (clear sky, rain, etc.)

## Contact Page



This page contains a simple contact form for feedback or queries.

Fields:

- Name
- Email
- Message textarea

While the form does not currently submit data to a backend, it serves as a well-structured, styled placeholder for future expansion.

## Styling

The CSS file ensures consistent styling across all pages. It includes:

- Flex and grid-based layout
- Centered content
- Styled input boxes and buttons
- Responsive formatting for various screen sizes

## Technologies Used

- **HTML5** — page structure and layout
- **CSS3** — styling and responsive design
- **JavaScript (Vanilla)** — dynamic DOM updates and API handling
- **OpenWeatherMap API** — fetches current weather by city name

## Conclusion

This project successfully demonstrates how to:

- Build a static multi-page application
- Integrate third-party APIs for real-time data
- Dynamically update UI content with JavaScript
- Provide a smooth user experience with responsive design

**WeatherNow** serves as a great foundation for more complex frontend projects in the future.