

React Weather App - Step-by-Step Instructions

1. Set up the project

- Open terminal
- Run: `npm create vite@latest weather-app --template react`
- `cd weather-app`
- `npm install`

2. Install and import Bootstrap

- Run: `npm install bootstrap`
- In `main.jsx`, add: `import 'bootstrap/dist/css/bootstrap.min.css';`

3. Get a free OpenWeatherMap API Key

- Go to https://home.openweathermap.org/api_keys
- Sign up or log in and generate a key
- In the project root, create a file named `.env`
- Add: `VITE_WEATHER_API_KEY=your_api_key_here`

4. Restart the dev server

- Vite only loads environment variables on startup
- Run: `npm run dev`

5. Create context file (WeatherContext.js)

- In `src/`, create `WeatherContext.js`
- Use `createContext`, `useState`, and `axios`
- Store: `city`, `weather`, `error`, `unit (C/F)`, `fetchWeather`, `toggleUnit`, and `conversion function`

6. Wrap App.jsx with WeatherProvider

- Import and wrap all components inside `<WeatherProvider>` in `App.jsx`

7. Create WeatherInput.jsx

- Create a text input for city name
- On button click, trigger `fetchWeather()`

8. Create WeatherDisplay.jsx

React Weather App - Step-by-Step Instructions

- Display city name, temperature, condition
- Add button to toggle between Celsius and Fahrenheit

9. Handle errors

- If city is not found, show a clear error message (e.g., `setError('City not found')`)

10. Test API response

- Log the full URL and paste in browser if needed
- Example: `https://api.openweathermap.org/data/2.5/weather?q=London&appid=YOUR_API_KEY&units=metric`

11. Search by location (optional)

- Use `q=City,CountryCode` or `lat/lon` to search more accurately

12. Run and test

- Start dev server: `npm run dev`
- Open `http://localhost:5173`
- Try valid and invalid city names