

- **Name: Afreen Ahmed**
 - **Enrollment Number: Mern@003**
 - **Assignment: Weather Application using JavaScript**
 - **Date of Submission: 28/07/2025**
-

Problem Solving Activity: Weather Application using JavaScript

1. Program Statement:

This project is a weather application built using HTML, CSS, and JavaScript. It allows users to enter a city name and fetches real-time weather information from the OpenWeatherMap API. The application displays the city name, temperature, and weather condition. It also changes the background image based on the temperature range to show sunny, rainy, or cold weather visually.

2. Algorithm:

1. Start
 2. Get the city name input from the user
 3. Use the fetch() API to get weather data for the city from OpenWeatherMap
 4. Display the city name, temperature, and weather description
 5. Based on temperature:
 - If temp > 25°C → show sunny background
 - If temp between 15–25°C → show rainy background
 - If temp ≤ 15°C → show cold background
 6. Handle errors if the city is not found
 7. End
-

3. Pseudocode:

Function weather():

Get city from input field

Fetch weather data using API

Convert response to JSON

Display city name, temperature, and description

If temperature > 25:

Set background to sunny

Else if 15 < temperature <= 25:

Set background to rainy

Else:

Set background to cold

Catch error and alert "City not found"

Attach event listener to button click to call weather()

4. Program Code:

```
const weather = async () => {
  const city = document.getElementById('inputValue').value;
  const nameVal = document.getElementById('name');
  const temp = document.getElementById('temp');
  const desc = document.getElementById('desc');

  try {
    const response = await
    fetch('https://api.openweathermap.org/data/2.5/weather?q=${city}&units=metric&appid=c41e17e6d8e4ee1ffcb898fe0eddc1e');

    const data = await response.json();

    nameVal.innerText = data.name;

    temp.innerText = data.main.temp + "°C";
```

```

desc.innerText = data.weather[0].main;

const temperature = data.main.temp;

document.body.classList.remove("sunny", "rainy", "cold");

if (temperature > 25) {
    document.body.classList.add("sunny");
} else if (temperature <= 25 && temperature > 15) {
    document.body.classList.add("rainy");
} else {
    document.body.classList.add("cold");
}

} catch (error) {
    alert("City not found");
}

};

document.getElementById('button').addEventListener('click', weather);

```

5. Test Cases:

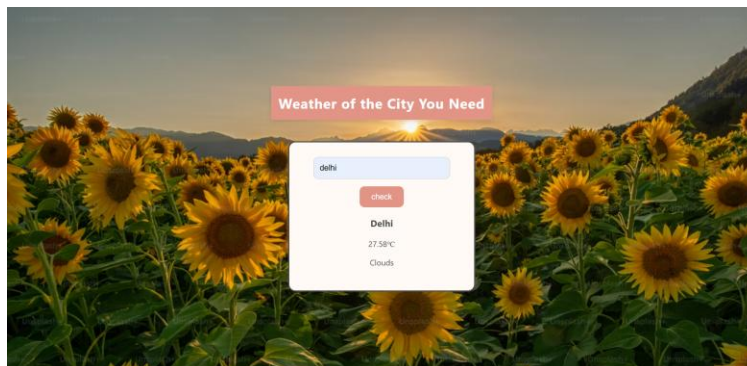
| Test Case No. | Input | Expected Output | Actual Output | Status |
|---------------|--------------|--|---------------|--------|
| 1 | Delhi | City: Delhi, Temp in °C, Weather Description, sunny bg | As expected | Pass |
| 2 | London | City: London, Temp in °C, Weather Description, cold bg | As expected | Pass |
| 3 | Invalid-City | Alert message "City not found" | Alert appears | Pass |

6. Screenshots of Output:

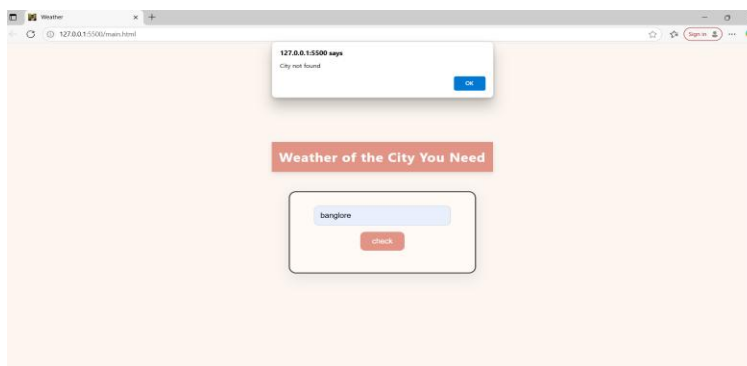
Case 1: Temperature is 14.54°C



Case 2: Temperature is 27.58°C



Case 3: Invalid City:



7. Observation / Reflection:

This project helped me understand how to work with APIs and dynamic DOM manipulation using JavaScript. It shows how real-time data can be fetched and used to create an interactive web application. The project also improved my skills in working with event listeners, error handling, and styling the UI based on logic. Overall, this activity made me more confident in using JavaScript for real-world web development.