

Learner Assignment Submission Format

Learner Details

- **Name:**H O Akash
 - **Enrollment Number:**SU625MR015
 - **Batch / Class:**MERN Stack
 - **Assignment:** Weather Application using JavaScript
 - **Date of Submission:**28/07/2025
-

Problem Solving Activity 1.1

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. Get the city name from the input field.
 2. Fetch the weather data from the API using the city name.
 3. Check if the data is valid.
 4. If the data is valid, display the city name, temperature, and weather condition.
 5. Update the body class based on the weather condition.
 6. If the data is not valid, display an error message.
-

3. Pseudocode

BEGIN

 GET city name from input field

 FETCH weather data from API using city name

 IF data is valid

 DISPLAY city name

 DISPLAY temperature

 DISPLAY weather condition

 UPDATE body class based on weather condition

 ELSE

 DISPLAY error message

 END IF

END

4. Program Code(HTML)

```
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <link rel="stylesheet" href="weather.css">
```

```
</head>

<body>

  <center> <h1>WEATHER </h1></center>

  <center> <div class="container">

    <input id="input" type="text" placeholder="enter the city name">

    <button id="button"><img alt="search icon" data-bbox="345 380 360 395"/></button>

    <div class="display">
<h2 id="name"></h2>
<p id="temp"></p>
<p id="desc"></p>

    </div>

  </div></center>

<script src="weather.js"></script>

</body>

</html>
```

5. Program Code(JavaScript)

```
const getWeather = async () => {

    const city = document.getElementById('input').value;

    const nameval = document.getElementById('name');

    const temp = document.getElementById('temp');

    const desc = document.getElementById('desc');

    const body = document.body;

    try {

        const response = await
fetch(`https://api.openweathermap.org/data/2.5/weather?q=${city}&APPID=cb7
f1c0dd1a0536c059f2315435439e8`);

        const data = await response.json()

        const weather = data.weather[0].main;

        nameval.innerText = data.name;

        temp.innerText = data.main.temp + "F";

        const weatherCondition = data.weather[0].main;

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

        if (weatherCondition.includes('Cloud')) {

            body.classList.add('cloudy');

        } else if (weatherCondition.includes('Rain')){

            body.classList.add('rainy');

        } else if ((data.main.temp) >= 30) {

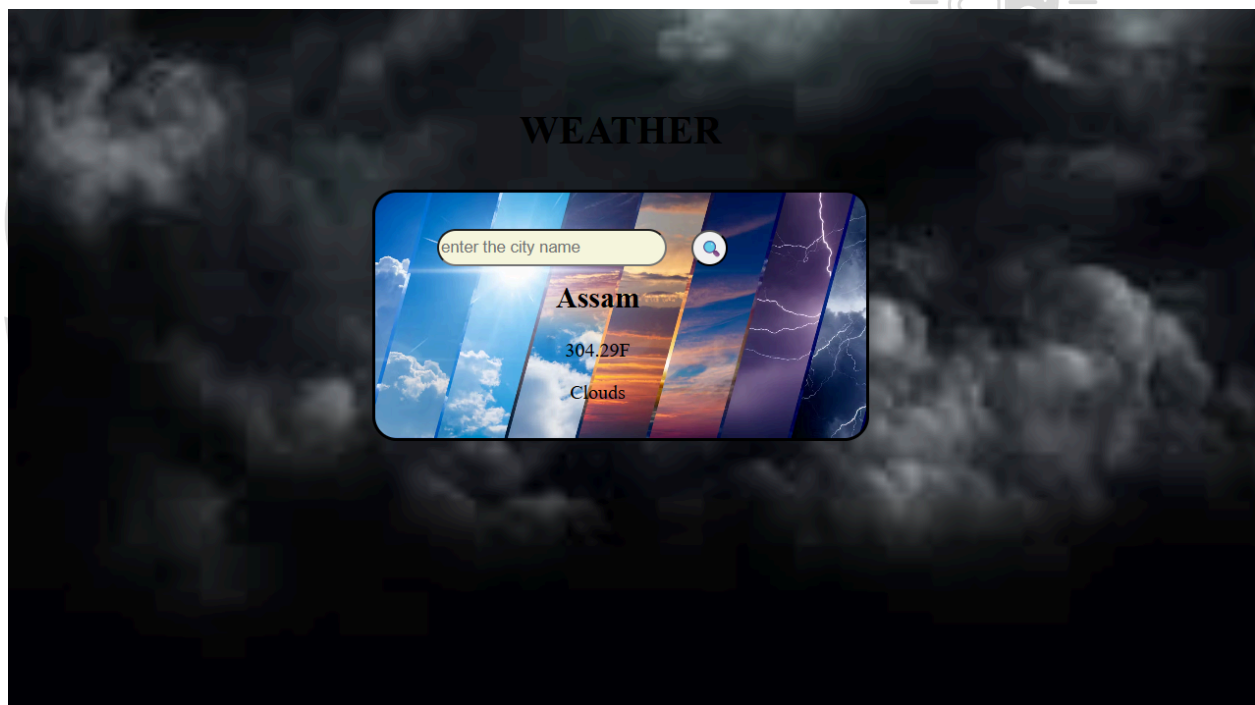
            body.classList.add('clear');

        } else {

            body.classList.add('default');
```

```
}  
  
} catch (error) {  
    alert("city is not found");  
};  
}  
  
document.getElementById('button').addEventListener('click', getWeather)
```

6. Screenshots of Output



7. Observation / Reflection

- The combined program is a simple and effective way to display the current weather conditions for a given city. The use of the OpenWeatherMap API makes it easy to fetch the weather data, and the pseudocode and algorithm provide a clear outline of the program

Problem Solving Activity 1.2

Follow the same Structure as problem Solving Activity 1.1.

