**Experiment 5: JavaScript Function to Fetch Weather Data**

**Aim:**

To create a JavaScript function that fetches and displays the current weather data for a city entered by the user using a weather API.

**Objectives:**

* Implement and call a JavaScript function to make HTTP requests.
* Handle JSON responses and dynamically update the DOM with fetched data.
* Debug and handle errors during the data fetching process.
* Display the fetched weather data on the webpage dynamically.

**Prerequisites:**

* Completion of Lab 4 (City input field).
* Basic understanding of HTML, CSS, and JavaScript.
* Familiarity with asynchronous JavaScript and API usage.
* A system with Git and an IDE installed (preferably VS Code).

**Procedure:**

1. Open the project folder weather-app in VS Code.
2. Sign up on OpenWeatherMap to obtain an API key.
3. Write a JavaScript function fetchWeatherData(city) in js/script.js to fetch weather data using the Fetch API.
4. Add an event listener to the search button to call the fetch function with the entered city.
5. Test the function by entering city names and checking the displayed data and console logs.
6. Commit changes to Git using meaningful commit messages.

**Code:**

**1. index.html**

<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1.0">  
 <title>Weather App - Experiment 5</title>  
 <link rel="stylesheet" href="css/style.css">  
</head>  
<body>  
 <div class="container">  
 <h2>Weather Application 🌦️</h2>  
 <input type="text" id="city-input" placeholder="Enter city name">  
 <button id="search-btn">Search</button>  
 <div id="weather-result"></div>  
 </div>  
 <script src="js/script.js"></script>  
</body>  
</html>

**2. css/style.css**

body {  
 font-family: Arial, sans-serif;  
 background: linear-gradient(135deg, #a2c2e0, #d1e8ff);  
 display: flex;  
 justify-content: center;  
 align-items: center;  
 height: 100vh;  
 margin: 0;  
}  
.container {  
 background-color: white;  
 padding: 30px;  
 border-radius: 15px;  
 box-shadow: 0 0 15px rgba(0,0,0,0.1);  
 text-align: center;  
 width: 320px;  
}  
h2 { color: #333; }  
input {  
 width: 80%;  
 padding: 10px;  
 margin-top: 10px;  
 border-radius: 8px;  
 border: 1px solid #ccc;  
 outline: none;  
}  
button {  
 margin-top: 10px;  
 padding: 10px 15px;  
 background-color: #0078d7;  
 color: white;  
 border: none;  
 border-radius: 8px;  
 cursor: pointer;  
}  
button:hover { background-color: #005fa3; }  
#weather-result { margin-top: 20px; text-align: left; }  
#weather-result h3 { color: #0078d7; }

**3. js/script.js**

async function fetchWeatherData(city) {  
 const apiKey = 'YOUR\_API\_KEY'; // Replace with your actual API key  
 const apiUrl = `https://api.openweathermap.org/data/2.5/weather?q=${city}&appid=${apiKey}&units=metric`;  
 try {  
 const response = await fetch(apiUrl);  
 const data = await response.json();  
 if (data.cod === "404") {  
 document.getElementById('weather-result').innerHTML =  
 `<p style="color:red;">City not found. Please enter a valid city name.</p>`;  
 return;  
 }  
 console.log("Weather data:", data);  
 document.getElementById('weather-result').innerHTML = `  
 <h3>Weather in ${data.name}</h3>  
 <p><strong>Temperature:</strong> ${data.main.temp} °C</p>  
 <p><strong>Condition:</strong> ${data.weather[0].description}</p>  
 <p><strong>Humidity:</strong> ${data.main.humidity}%</p>  
 <p><strong>Wind Speed:</strong> ${data.wind.speed} m/s</p>  
 `;  
 } catch (error) {  
 console.error("Error fetching weather data:", error);  
 document.getElementById('weather-result').innerHTML =  
 `<p style="color:red;">Error fetching weather data. Please try again later.</p>`;  
 }  
}  
document.getElementById('search-btn').addEventListener('click', function() {  
 const city = document.getElementById('city-input').value.trim();  
 if (city === "") {  
 document.getElementById('weather-result').innerHTML =  
 `<p style="color:red;">Please enter a city name.</p>`;  
 return;  
 }  
 fetchWeatherData(city);  
});

**Output / Testing:**

* Open index.html in the browser.
* Enter a city name and click **Search**.
* Check the console (F12 → Console) for weather data.
* The weather information should appear dynamically below the search button.

**Screenshot placeholders:**

1. Screenshot: Output in browser showing weather data.

**Conclusion:**

* Successfully implemented a JavaScript function to fetch and display weather data.
* Learned to use Fetch API and async/await for API calls.
* Practiced dynamically updating the DOM and handling errors gracefully.

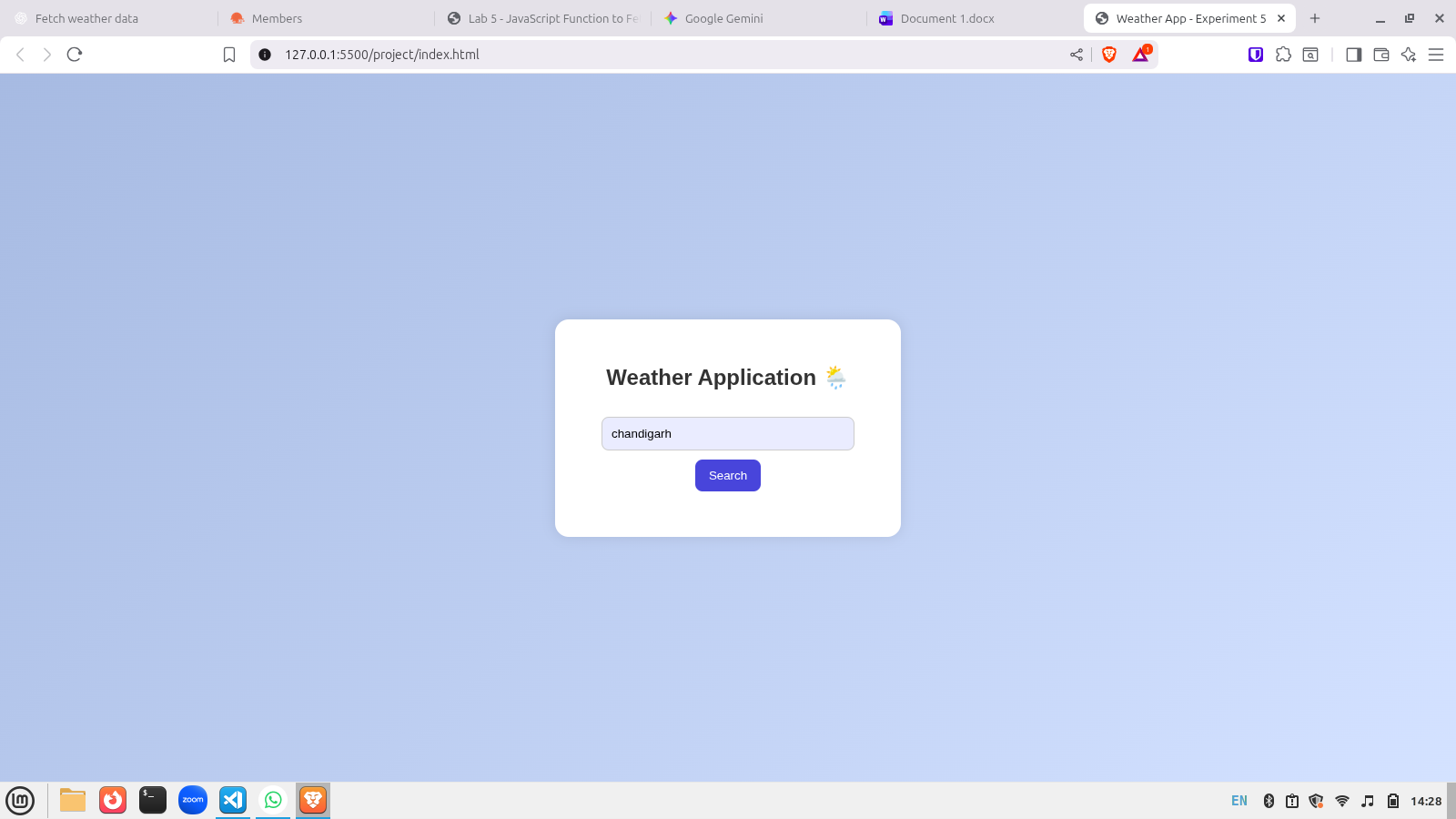
**Git Commit:**

git add .  
git commit -m "Implemented function to fetch current weather data"  
git push

**API Key Reminder:**

Replace 'YOUR\_API\_KEY' in script.js with your own OpenWeatherMap API key for the application to work.

***Screenshots:***



***Project Link:***

***https://github.com/manjot11singh/Weather-App***