COMSATS University, Islamabad Campus
Department of Computer Science
Lab Midterm Examination - Fall 2024

Subject: -	Web Technologies	Teacher: -	Ms. Saadia Maqbool
Class & Sec.	BCS 5 A&B	Date: -	11-11-2024
Marks: -	25	Duration: -	180 minutes

CLO4 - Apply the concepts of markup & scripting languages and client-side technologies.

Task: Build a Weather Dashboard Application

Instructions:

1. Complete the following requirements to create a weather dashboard that fetches data from a weather API.
2. Maximum time required to complete this task is **3 hours**.
3. Use only HTML, CSS, and JavaScript (no frameworks or libraries).
4. Comment your code and ensure it's well-organized.

Scenario:

Create an interactive weather dashboard where users can search for a city and see the current weather along with a 5-day forecast. Use the OpenWeatherMap API (or any other free weather API) to retrieve data. (You will need to create a free account on the chosen API platform to get an API key).

Requirements

Part 1: HTML Structure (5 points)

1. **Create an HTML file** with a title (e.g., "Weather Dashboard").
2. In the `<body>`, add:
    - An input field for entering the city name.
    - A "Search" button.
    - A container to display the current weather information.
    - A container for the 5-day forecast.

Part 2: CSS Styling (5 points)

1. Centre the application and add padding and a light background colour.
2. Style the input field, buttons, and containers for readability.

3. Add hover effects to buttons and visual elements like borders and rounded corners for the forecast cards.

Part 3: JavaScript Functionality (15 points)

1. **Fetch Current Weather Data** (2 points):
    - Capture the city name entered in the input field when the "Search" button is clicked.
    - Fetch the current weather data for that city from the weather API.
    - Display:
        - City name, temperature, humidity, wind speed, and weather icon.
2. **Fetch 5-Day Weather Forecast** (2 points):
    - Fetch and display the 5-day weather forecast, showing the date, weather icon, temperature, and humidity for each day.
3. **Display Weather Data in a User-Friendly Format** (5 points):
    - Format the weather data so that each day's forecast is displayed in a card.
    - Ensure that each card displays the date and weather details clearly.
4. **Error Handling and Input Validation** (2 points):
    - If the city name is invalid or not found, display an error message (e.g., “City not found. Please try again.”).
    - Prevent blank searches from triggering an API call.
5. **Save Search History in Local Storage** (2 points):
    - Save each searched city name in local storage.
    - Display the search history as a list below the search bar.
    - Allow users to click on a past city to quickly fetch weather data for that city again.
6. **Clear Search History Button** (2 points):
    - Add a "Clear Search History" button to clear the saved search history from both local storage and the displayed list.

Bonus Requirements

1. **Temperature Unit Toggle**:
    - Add a toggle button to switch between Celsius and Fahrenheit.
2. **Weather Icons and Color Coding**:
    - Use color coding or different background images/icons to represent various weather conditions (e.g., blue background for rain, yellow for sunny).
3. Any other customizations, JavaScript usage is welcomed and shall get you additional marks within marks range.

Submission

Create a word/pdf document that includes your code chunks explained, all important output screenshots with proper report formatting. Upload on MS Teams.

Keep the code with yourself, in case, vivas are called.

Copied submissions shall lead to exam failure, basic AI generated structures will also get you basic minimum marks.
