**Basic Weather App api using Html, Css and JavaScript for beginners.**

Presented by :

Carmona, Sem Gabrielle, S.

Ramos, KyleJoseph

Chavez, Jeremy, D.

# Overview

- This Weather API offers detailed insights into the weather conditions of specific locations, particularly those situated at high or low altitudes. Understanding the weather in your chosen destination is crucial for effective planning and preparation. This API provides quick access to weather data through the use of internet connection , ensuring immediate responses regarding the current and forecasted conditions of various locations.

# Objectives

- The project's objective is to create a weather application that uses web services and APIs from outside sources. The application retrieves current weather information. This project showcases integrative programming by utilizing outside services, such as the OpenWeather API, to improve the functionality of the application and provide users with local time and real-time weather information.

# Technologies Used

* Visual Studio Code

* Html – To structure and present content on the web.

* Css – To control presentation and layout of HTML elements on a web page.

* Javascript - Main components for functionalities

# System Features

1. Weather Data Fetching - After the user enters the city's name, the application retrieves weather data, including temperature, description, and an icon that indicates the current weather, via the OpenWeatherApp API.
2. Dynamic Timezone - Using the timezone information from the API, the app adapts to the chosen city's timezone. It updates every second to show the current local time and date.

1. Visual Weather Representation - The program makes use of video backgrounds that match the current weather description, such as rainy, cloudy, clear, and snowy seasons, This enhances the user experience by visually reflecting the weather conditions in real time.

**API Integration**

**1. OpenWeather API**

# - The application integrates with OpenWeatherApp API

**(https://openweathermap.org/api) to retrieve weather data.**

2. - To retrieve weather data for the provided city, the "get\_weather(city)" function sends a GET request. After that, it parses the JSON response to extract pertinent information like the icon, temperature, and weather description.

**Functionality**

* The user enters a city name in the input field.
* Clicking the "Search" button triggers the getWeather() function.
* getWeather() fetches data from the OpenWeatherMap API.
* displayWeather() updates the HTML with current weather information.
* displayHourlyForecast() updates the HTML with the hourly forecast.

**Future Improvements**

* Implement better error handling and user feedback.
* Store the API key securely.
* Improve the UI/UX.
* Add more features, such as a 5-day forecast.

# How to set up and run the application

* Install Vscode
* Download Extension pack

**2. Obtain an ExchangeRate API key and an OpenWeatherApp API key.**

-You must have an OpenWeatherApp API key in order to access data and run the application correctly. You can get one by registering on OpenWeather.

After obtaining your keys, use them to replace the code's placeholders:

--OpenWeather API key: Enter your OpenWeather API key in place of API\_key = "YOUR\_API\_KEY" in the code.

# Running the Application

* -Once you set up all requirements, you can run the application using vscode or other types of IDE.
* You can also run the application using other IDE as long it’s supported.

**Conclusion**

This weather application demonstrates a basic implementation of fetching and displaying weather data using web technologies. It can be further expanded and improved to provide a more comprehensive weather information service.