Name : Devkumar Biswas

Class: BE(AI&DS)

Div: B

Subject : DMV(CL-I Lab)

Roll no. : BEAD21267

**Assignment No. - 1**

**Problem Statement** : Interacting with Web APIs Problem Statement: Analyzing Weather Data from OpenWeatherMap API Dataset: Weather data retrieved from OpenWeatherMap API Description: The goal is to interact with the OpenWeatherMap API to retrieve weather data for a specific location and perform data modeling and visualization to analyze weather patterns over time.

Tasks to Perform: 1. Register and obtain API key from OpenWeatherMap. 2. Interact with the OpenWeatherMap API using the API key to retrieve weather data for a specific location. 3. Extract relevant weather attributes such as temperature, humidity, wind speed, and precipitation from the API response. 4. Clean and preprocess the retrieved data, handling missing values or inconsistent formats. 5. Perform data modeling to analyze weather patterns, such as calculating average temperature, maximum/minimum values, or trends over time. 6. Visualize the weather data using appropriate plots, such as line charts, bar plots, or scatter plots, to represent temperature changes, precipitation levels, or wind speed variations. 7. Apply data aggregation techniques to summarize weather statistics by specific time periods (e.g., daily, monthly, seasonal). 8. Incorporate geographical information, if available, to create maps or geospatial visualizations representing weather patterns across different locations. 9. Explore and visualize relationships between weather attributes, such as temperature and humidity, using correlation plots or heatmaps.

Code:

import requests

import pandas as pd

import datetime

# Set your OpenWeatherMap API key

api\_key = 'fb365aa6104829b44455572365ff3b4e'

lat = 18.184135

lon = 74.610764

#https://openweathermap.org/api/one-call-3#how How to use api call

# Construct the API URL

api\_url = f"http://api.openweathermap.org/data/2.5/forecast?lat={lat}&lon={lon}&appid={api\_key}"

# Send a GET request to the API

response = requests.get(api\_url)

weather\_data = response.json() #pass response to weather\_data object(dictionary)

weather\_data.keys()

dict\_keys=(['cod', 'message', 'cnt', 'list', 'city'])

weather\_data['list'][0]

{'dt': 1690189200,

'main': {'temp': 298.21,

'feels\_like': 298.81,

'temp\_min': 298.1,

'temp\_max': 298.21,

'pressure': 1006,

'sea\_level': 1006,

'grnd\_level': 942,

'humidity': 78,

'temp\_kf': 0.11},

'weather': [{'id': 804,

'main': 'Clouds',

'description': 'overcast clouds',

'icon': '04d'}],

'clouds': {'all': 100},

'wind': {'speed': 6.85, 'deg': 258, 'gust': 12.9},

'visibility': 10000,

'pop': 0.59,

'sys': {'pod': 'd'},

'dt\_txt': '2023-07-24 09:00:00'}

len(weather\_data['list'])

weather\_data['list'][0]['weather'][0]['description']

temperatures = [item['main']['temp'] for item in weather\_data['list']]

timestamps = [pd.to\_datetime(item['dt'], unit='s') for item in weather\_data['list']]

temperature = [item['main']['temp'] for item in weather\_data['list']]

humidity = [item['main']['humidity'] for item in weather\_data['list']]

wind\_speed = [item['wind']['speed'] for item in weather\_data['list']]

weather\_description = [item['weather'][0]['description'] for item in weather\_data['list']]

# Create a pandas DataFrame with the extracted weather data

weather\_df = pd.DataFrame({

'Timestamp': timestamps,

'Temperature': temperatures,

'humidity': humidity,

'wind\_speed': wind\_speed,

'weather\_description': weather\_description,

})

# Set the Timestamp column as the DataFrame's index

weather\_df.set\_index('Timestamp', inplace=True)

max\_temp = weather\_df['Temperature'].max()

max\_temp

min\_temp = weather\_df['Temperature'].min()

min\_temp

# Handling missing values

weather\_df.fillna(0, inplace=True) # Replace missing values with 0 or appropriate value

# Handling inconsistent format (if applicable)

weather\_df['Temperature'] = weather\_df['Temperature'].apply(lambda x: x - 273.15 if isinstance(x, float)else x)

print(weather\_df)

OUTPUT:-





