Weather analysis and visualisation using tableau and Python

Scope of the project:

Collecting the data from weather API website and manipulating it into the more organised format and doing various visualisation using Tableau and Python and creating database using PostgreSQL.

Weather API:

Weather APIs are **Application Programming Interfaces that provide access to current & historical weather data on a global scale**. There are several public weather APIs like Open Weather Map API, Dark Sky API, & Visual Crossing Weather API. In this project we are using **Open Weather Map** to get the data's.

Elements of open weather map:

- **City**
  - city.id - City ID
  - city.name - City name
  - city.coord
  - city.coord.lon - City geo location, longitude
  - city.coord.lat - City geo location, latitude
  - city.country - Country code (GB, JP etc.)
  - timezone -Shift in seconds from UTC
  - city.sun
  - city.sun.rise - Sunrise time
  - city.sun.set - Sunset time ➢ **Temperature**
  - temperature.value - Temperature
  - temperature.min - Minimum temperature at the moment of calculation. This is minimal currently observed temperature (within large megalopolises and urban areas), use this parameter optionally.
  - temperature.max - Maximum temperature at the moment of calculation. This is maximal currently observed temperature

(within large megalopolises and urban areas), use this parameter optionally.

temperature.unit - Unit of measurements. Possible value is C.

- feels_like.value - Temperature. This temperature parameter accounts for the human perception of weather.
- feels_like.unit - Unit of measurements. Possilbe valure is Celsius, Kelvin, and Fahrenheit. Unit Default: Kelvin

- **Humidity**
  - humidity.value - Humidity value
  - humidity.unit - Humidity units, %
- **pressure**
  - pressure.value Pressure value
  - pressure.unit Pressure units, hPa
- **wind**
  - wind.speed
  - wind.speed.value - Wind speed
  - wind.speed.unit - Wind speed units, m/s
  - wind.speed.name - Type of the wind
  - wind.direction
  - wind.direction.value - Wind direction, degrees (meteorological)
  - wind.direction.code - Code of the wind direction. Possilbe value is WSW, N, S etc.
  - wind.direction.name - Full name of the wind direction.
- **Clouds**
  - clouds.value - Cloudiness
  - clouds.name - Name of the cloudiness
- **Visibility**
  - visibility.value Visibility, meter. The maximum value of the visibility is 10km.

Software used:

1. **Python** - Python is commonly used for **developing websites and software, task automation, data analysis, and data visualization**. Since it's relatively easy to learn, Python has been adopted by many non-programmers such as accountants and scientists, for a variety of everyday tasks, like organizing finances.

2. **Tableau** - Tableau is an end-to-end data analytics platform that **allows you to prep, analyze, collaborate, and share your big data insights**. Tableau excels in self-service visual analysis, allowing people to ask new questions of governed big data and easily share those insights across the organization.

3. **PostgreSQL** - PostgreSQL is used **as the primary data store or data warehouse for many web, mobile, geospatial, and analytics applications**. It is a highly stable database management system, backed by more than 20 years of community development which has contributed to its high levels of resilience, integrity, and correctness.

Procedures:

1. The city name provided by the user is used as input.
2. The API link for the 'One Call API 3.0' is acquired from the OpenWeatherMap website.
3. An API key is generated from OpenWeatherMap.org after creating an account.
4. Using the obtained URL and API key, an API call is made to fetch the weather information for the specified city.
5. The received string data is converted into a DataFrame and then saved as a CSV file to facilitate visualization.
6. Tableau is utilized to import the CSV file and create diverse visualizations.
7. The Python programming language is used to normalize the data and divide it into four separate tables, which are subsequently uploaded to PostgreSQL.
8. PostgreSQL is employed to retrieve the data and execute various queries for analysis and reporting purposes.

Conclusion:

This project facilitates the comprehension of climate and weather variations, enabling the prediction of diverse climate changes worldwide. The provided visualizations aid in rapid information absorption, enhanced insights, and expedited decision-making processes.