Task1 API INTEGRATION AND DATA VISUALIZATION

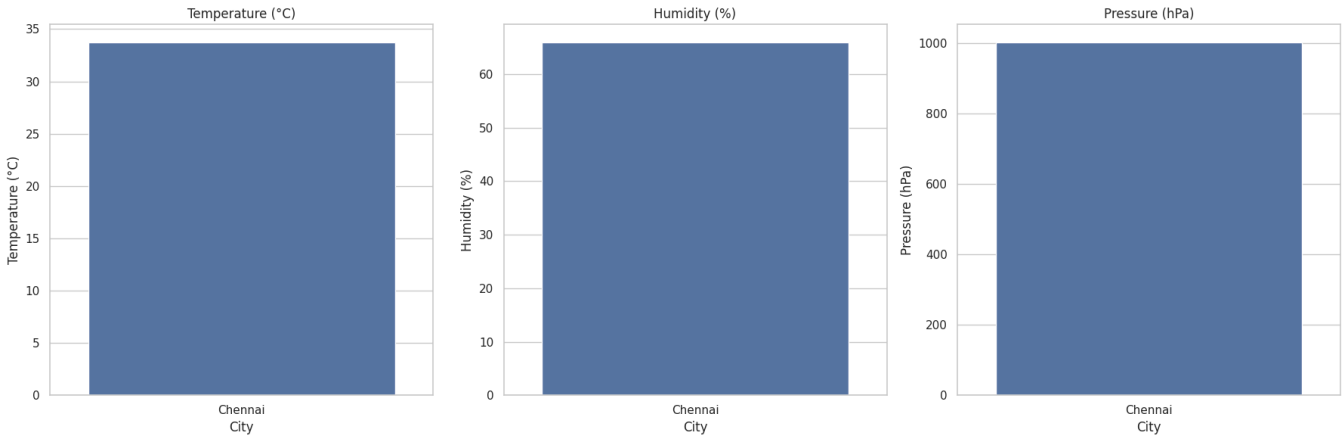
```
pip install requests pandas matplotlib seaborn
```

```
Requirement already satisfied: requests in /usr/local/lib/python3.11/dist-packages (2.32.3)
Requirement already satisfied: pandas in /usr/local/lib/python3.11/dist-packages (2.2.2)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.11/dist-packages (3.10.0)
Requirement already satisfied: seaborn in /usr/local/lib/python3.11/dist-packages (0.13.2)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.11/dist-packages (from requests) (3.4.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.11/dist-packages (from requests) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests) (2.4.0)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.11/dist-packages (from requests) (2025.4.26)
Requirement already satisfied: numpy>=1.23.2 in /usr/local/lib/python3.11/dist-packages (from pandas) (2.0.2)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.11/dist-packages (from pandas) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-packages (from pandas) (2025.2)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (1.3.2)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (4.58.0)
Requirement already satisfied: kiwisolver>=1.3.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (1.4.8)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (24.2)
Requirement already satisfied: pillow>=8 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (11.2.1)
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (3.2.3)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-packages (from python-dateutil>=2.8.2->pandas) (1.17.0)
```

```
import requests
import pandas as pd

# Replace with your OpenWeatherMap API key
API_KEY = '63e64d9566db363a2392a5f8fe453b3d'
CITY = 'Chennai'
URL = f'http://api.openweathermap.org/data/2.5/weather?q={CITY}&appid={API_KEY}&units=metric'

response = requests.get(URL)
data = response.json()

if response.status_code == 200:
    weather_data = {
        'City': CITY,
        'Temperature (°C)': data['main']['temp'],
        'Humidity (%)': data['main']['humidity'],
        'Pressure (hPa)': data['main']['pressure'],
        'Weather': data['weather'][0]['description']
    }
    df = pd.DataFrame([weather_data])
    print(df)
else:
    print(f"Error: {data.get('message', 'Unable to fetch data')}")
```

```
       City  Temperature (°C)  Humidity (%)  Pressure (hPa)           Weather
0  Chennai             33.75            66            1003  scattered clouds
```

```
import matplotlib.pyplot as plt
import seaborn as sns

# Assuming 'df' is the DataFrame from the previous script

# Set the style
sns.set(style="whitegrid")

# Create subplots
fig, axes = plt.subplots(1, 3, figsize=(18, 6))

# Temperature Plot
sns.barplot(x=df['City'], y=df['Temperature (°C)'], ax=axes[0])
axes[0].set_title('Temperature (°C)')

# Humidity Plot
sns.barplot(x=df['City'], y=df['Humidity (%)'], ax=axes[1])
axes[1].set_title('Humidity (%)')

# Pressure Plot
sns.barplot(x=df['City'], y=df['Pressure (hPa)'], ax=axes[2])
axes[2].set_title('Pressure (hPa)')

plt.tight_layout()
plt.show()
```



Start coding or generate with AI.