EX:No.1 221501518

25/01/26

**1.Program to implement time series data for import library, load data,Preprosessing and visualizing**

**Aim:**

Write a program to implement time series data for import library, load data, Preprocessing and visualising.

**Algorithm :**

Step 1: Install required libraries (if not already installed).

Step 2: Import necessary libraries (pandas, numpy, matplotlib).

Step 3: Load air pollution data, parse dates, and set 'date' as the index.

Step 4: Remove duplicate timestamps and fill missing values.

Step 5: Select the 'pollution\_today' column.

Step 6: Remove outliers using the IQR method.

Step 7: Ensure daily data frequency.

Step 8: Resample to weekly average (optional, not used in the plot).

Step 9: Create a figure and plot daily pollution levels as a line graph.

Step 10: Set labels, title, and legend for the plot.

Step 11: Show the plot.

**Program Code :**

import numpy as np

import pandas as pd

import seaborn as sns

import matplotlib.pyplot as plt

# Load the dataset

df = pd.read\_csv(r"C:\Users\HDC0422272\Downloads\GlobalLandTemperaturesByCity.csv",

parse\_dates=True, index\_col=0).dropna()

# Drop unnecessary columns if they exist

columns\_to\_drop = ['LandAverageTemperatureUncertainty', 'LandMaxTemperature',

'LandMaxTemperatureUncertainty', 'LandMinTemperature', 'LandMinTemperatureUncertainty',

'LandAndOceanAverageTemperature',

'LandAndOceanAverageTemperatureUncertainty']

df.drop(columns=[col for col in columns\_to\_drop if col in df.columns], inplace=True)

# Rename to 'Temp' if only one column remains

if df.shape[1] == 1:

df.columns = ['Temp']

df.index.name = "Date"

# Debugging: Check available columns

print("Available columns:", df.columns)

# Find the temperature column (case-insensitive)

temp\_col = next((col for col in df.columns if 'temp' in col.lower()), None)

if temp\_col is None:

raise KeyError("No 'Temp' column found!")

# Plot temperature variation

plt.figure(figsize=(22, 6))

sns.lineplot(x=df.index, y=df[temp\_col])

plt.title('Temperature Variation')

plt.xlabel('Date')

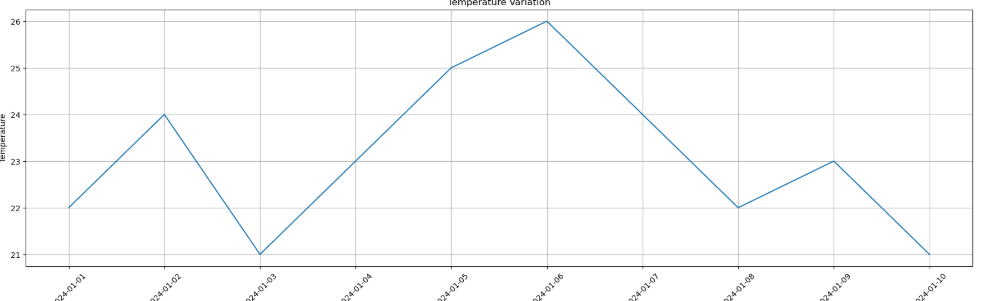
plt.ylabel('Temperature')

plt.xticks(rotation=45)

plt.grid(True)

plt.show()

**Output :**



**Result:**

Thus, the program using the time series data implementation has been done successfully.