Ex.No.2

31/1/25 Implementing various visualization techniques using Time series data

**AIM:**

To implement the various visualization techniques using Time series data

**Time series data:**

import matplotlib.pyplot as plt

import pandas as pd

data = {

'date': pd.date\_range(start='2022-01-01', periods=100, freq='D'),

'value': [i + (i \* 0.1) for i in range(100)] *# Simple increasing trend*

}

df = pd.DataFrame(data)

plt.figure(figsize=(10, 5))

plt.plot(df['date'], df['value'], label='Value', color='b')

plt.title('Time Series Plot')

plt.xlabel('Date')

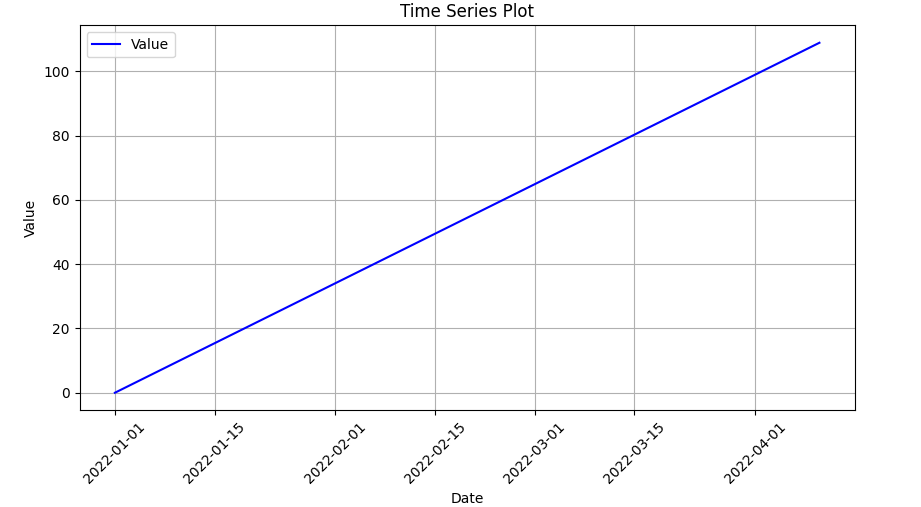
plt.ylabel('Value')

plt.xticks(rotation=45)

plt.legend()

plt.grid(True)

plt.show()



**Time series with rolling mean :**

df['rolling\_mean'] = df['value'].rolling(window=10).mean()

plt.figure(figsize=(10, 5))

plt.plot(df['date'], df['value'], label='Original Data', color='b', alpha=0.5)

plt.plot(df['date'], df['rolling\_mean'], label='Rolling Mean (10 days)', color='r', linestyle='--')

plt.title('Time Series with Rolling Mean')

plt.xlabel('Date')

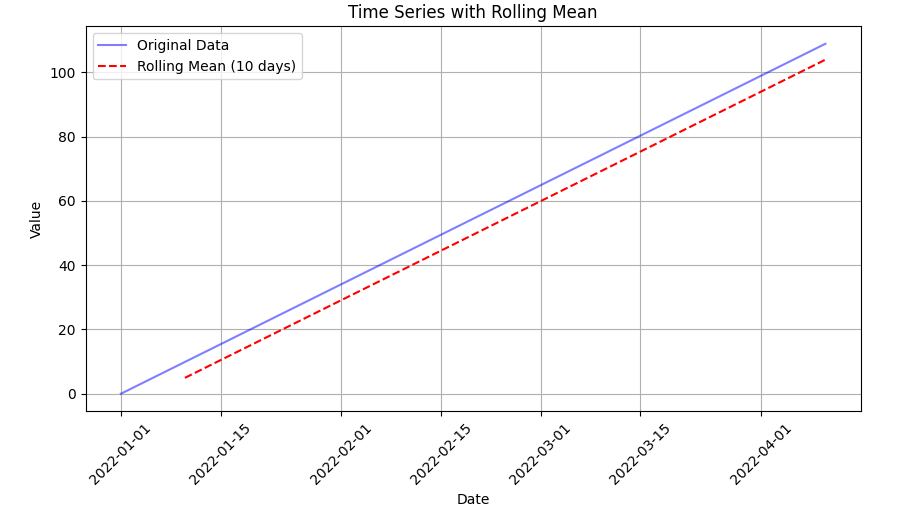
plt.ylabel('Value')

plt.xticks(rotation=45)

plt.legend()

plt.grid(True)

plt.show()



**Seaborn plot with time series data**:  
import seaborn as sns

plt.figure(figsize=(10, 5))

sns.lineplot(x=df['date'], y=df['value'], label='Value', color='b')

sns.lineplot(x=df['date'], y=df['rolling\_mean'], label='Rolling Mean', color='r', linestyle='--')

plt.title('Time Series with Seaborn')

plt.xlabel('Date')

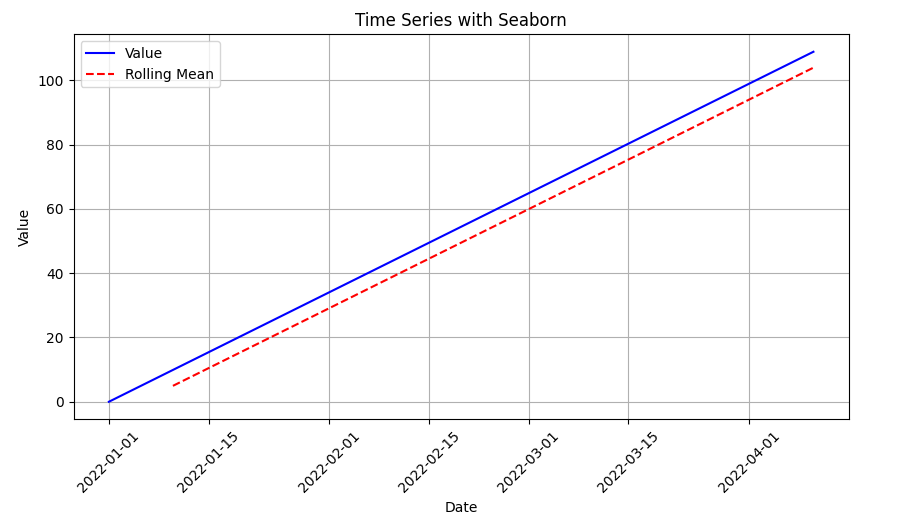
plt.ylabel('Value')

plt.xticks(rotation=45)

plt.legend()

plt.grid(True)

plt.show()



**Interactive time series plot:**

import plotly.express as px

fig = px.line(df, x='date', y='value', title='Interactive Time Series')

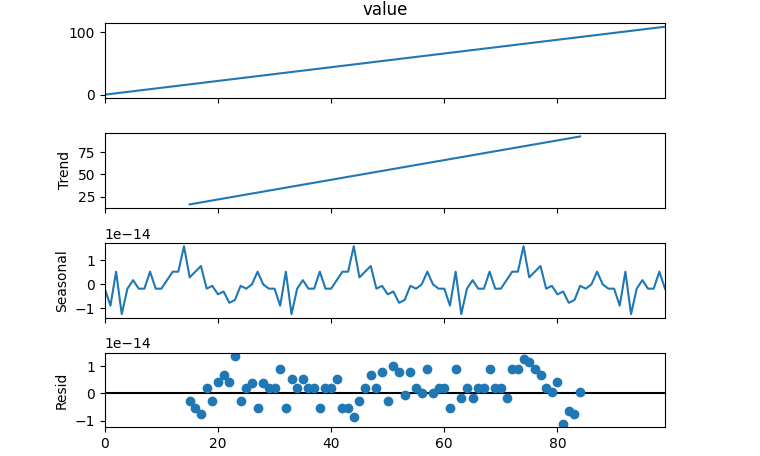
fig.add\_scatter(x=df['date'], y=df['rolling\_mean'], mode='lines', name='Rolling Mean', line=dict(dash='dash'))

fig.show()  
import statsmodels.api as sm

decomposition = sm.tsa.seasonal\_decompose(df['value'], model='additive', period=30)

decomposition.plot()

plt.show()



**RESULT:**

Thus the various visualization techniques were done successfully