

22/01/25

**Program to Implement programs to check stationary of a time series data****Aim:**

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

**Algorithm:****Step 1: Import Required Libraries**

Import **pandas**, **matplotlib.pyplot**, **seaborn** for visualization, and **adfuller** from **statsmodels.tsa.stattools** for stationarity test.

**Step 2: Load the Dataset**

Load the CSV file using **pandas.read\_csv()** and convert it into a time series format using **pd.date\_range()**.

**Step 3: Preprocess Data**

Clean and rename necessary columns. Convert the date column to datetime index. Handle missing values.

**Step 4: Perform ADF Test**

Use the **adfuller()** method on the price column to get the ADF statistic, p-value, and critical values.

**Step 5: Visualize Rolling Statistics**

Plot rolling mean and standard deviation to visually assess stationarity.

**Code:**

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from statsmodels.tsa.stattools import adfuller

# Load dataset (assumes 'df' is already loaded)
df['sale_date'] = pd.date_range(start="2023-01-01", periods=len(df), freq="D")
df.set_index('sale_date', inplace=True)

# Extract the price series
price_series = df['price'].dropna()

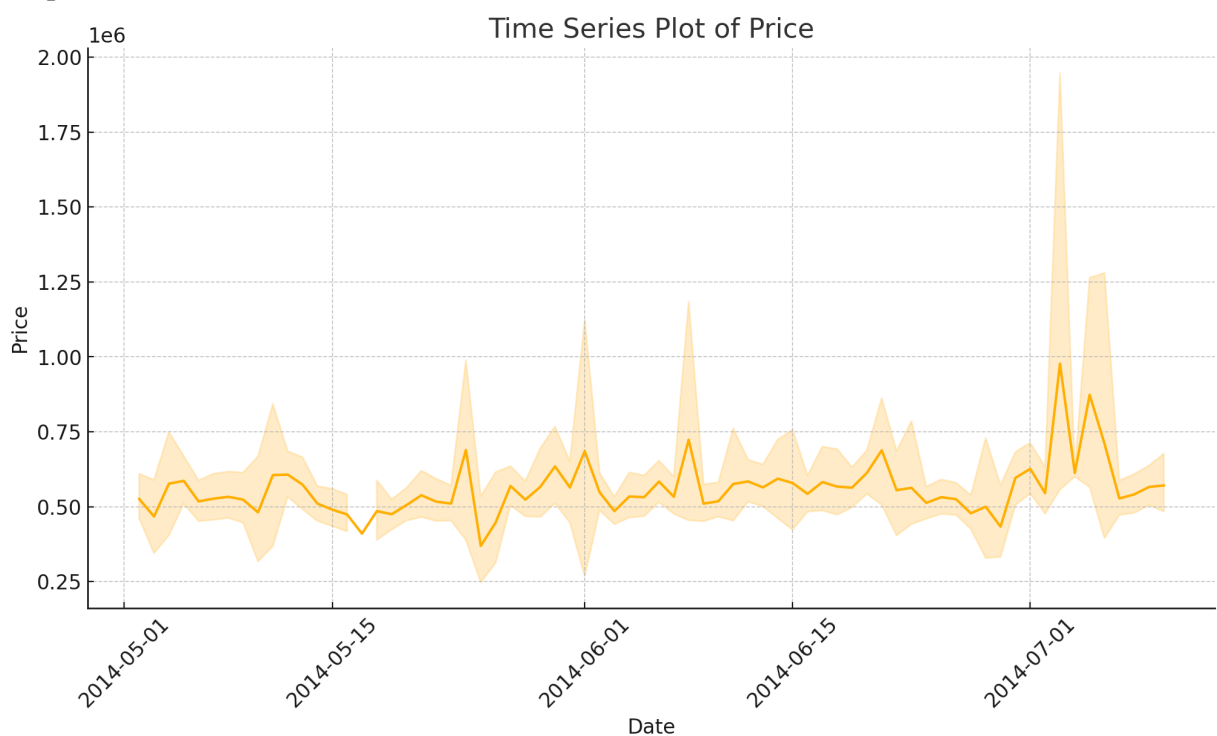
# Perform Augmented Dickey-Fuller Test
```

```

adf_result = adfuller(price_series)
print("ADF Statistic:", adf_result[0])
print("p-value:", adf_result[1])
print("Critical Values:", adf_result[4])
print("Is Stationary:", "Yes" if adf_result[1] < 0.05 else "No")
# Calculate rolling mean and std
rolling_mean = price_series.rolling(window=12).mean()
rolling_std = price_series.rolling(window=12).std()
# Plot rolling statistics
plt.figure(figsize=(12,6))
plt.plot(price_series, color='blue', label='Original')
plt.plot(rolling_mean, color='red', label='Rolling Mean')
plt.plot(rolling_std, color='green', label='Rolling Std Dev')
plt.legend(loc='best')
plt.title('Rolling Mean & Standard Deviation')
plt.grid(True, linestyle='--', alpha=0.6)
plt.tight_layout()
plt.savefig("/mnt/data/rolling_statistics.png")
plt.show()

```

### Output:



### Result:

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