

**EX:No.1**

**DATE:25/01/25**

## **Implement Programs For Time Series Data Cleaning, Loading, And Handling Time Series Data And Pre-Processing Techniques**

### **AIM:**

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

### **OBJECTIVE:**

- Load, clean, and analyze sales data (2012-2021).
- Handle **missing values and outliers** for better accuracy.
- Identify sales trends over time.
- Visualize sales performance using graphs and time-series plots.
- Help businesses and decision-makers make informed sales strategies.

### **BACKGROUND:**

- Sales data analysis helps businesses understand performance and customer trends.
- Key factors include revenue, customer purchases, seasonal trends, and promotions.
- Analyzing historical sales data aids in demand forecasting and business growth strategies.

### **SCOPE OF THE PROGRAM:**

- **Data Processing & Cleaning**
- **Exploratory Data Analysis (EDA)**
- **Data Visualization**

### **CODE:**

```
i pdimport numpy as np

import matplotlib.pyplot as plt

df = pd.read_csv("r"C:\Users\exam\Downloads\archive (2)\supermarket_sales -
Sheet1.csv") df = df[~df.index.duplicated(keep='first')]

df.fillna(method="ffill", inplace=True)

df.fillna(method="bfill", inplace=True)
```

```

df = df[['pollution_today']]

Q1 = df.quantile(0.25)

Q3 = df.quantile(0.75)

IQR = Q3 - Q1

df = df[~((df < (Q1 - 1.5 * IQR)) | (df > (Q3 + 1.5 * IQR))).any(axis=1)]

df = df.asfreq('D')

df_weekly = df.resample('W').mean()

plt.figure(figsize=(12,5))

plt.plot(Total Sales Over Time (Cleaned Data) plt.xlabel("Date")

plt.ylabel("'Total Sales', fontsize=12")

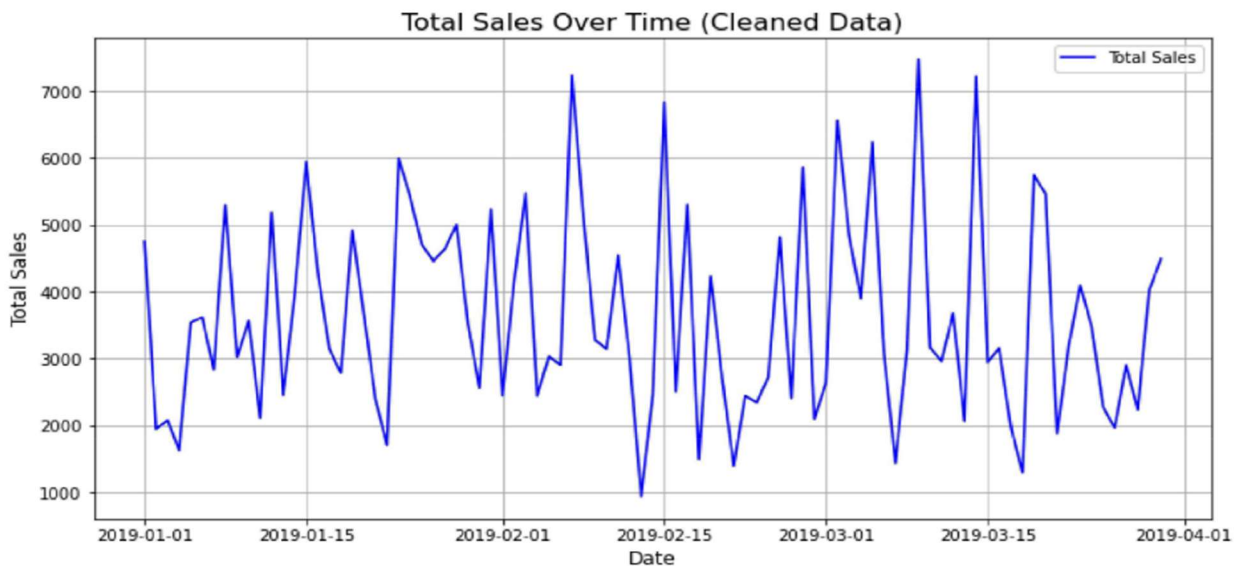
plt.title("Air Pollution Over Time (Cleaned)")

plt.legend()

plt.show()

```

## OUTPUT:



## RESULT:

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