**Ex : 01**

**Implement programs for time series data cleaning, loading and handling times series data and pre-processing techniques**

**AIM**

To write a program for time series data cleaning, loading and handling times series data and pre-processing techniques

**PROCEDURE**

 **Load the Car Sales Data**: Import the dataset that contains the car sales information (with a date column and the number of cars sold) from an external source (such as an Excel or CSV file).

 **Clean the Data**:

* Standardize the column names for easier access and manipulation by converting them to lowercase and stripping any leading or trailing spaces.
* Ensure that the date column is in a proper datetime format so it can be used for time-based operations and plotting.

 **Preprocess the Data**:

* Aggregate the sales data by date, allowing for a clearer understanding of the number of cars sold on each day.
* Handle any missing data by using imputation techniques (e.g., forward filling), which ensures that the dataset is complete and accurate for further analysis.

 **Visualize the Data**:

* Create a time series line plot that shows the trend of car sales over time. This visualization helps in identifying patterns, trends, and seasonality in the data, and can also reveal outliers or irregularities.

 **Make the Data Ready for Further Analysis or Modeling**:

* Cleaned, aggregated, and visualized data is now ready for further statistical analysis or time series forecasting models.
* The procedure also lays the groundwork for exploring advanced techniques such as anomaly detection, trend analysis, or building predictive models like ARIMA or machine learning-based forecasting

**PROGRAM**

import pandas as pd

import seaborn as sns

import matplotlib.pyplot as plt

df = pd.read\_csv("C:/Users/prave/Downloads/Car Sales.xlsx - car\_data.csv/Car Sales.xlsx - car\_data.csv")

df.columns = df.columns.str.strip().str.lower()

print("Column Names:", df.columns)

df['date'] = pd.to\_datetime(df['date'])

sales\_per\_day = df.groupby('date').size().reset\_index(name='sales')

sns.set(style="whitegrid")

plt.figure(figsize=(12, 6))

sns.lineplot(data=sales\_per\_day, x='date', y='sales', label='Car Sales', color='blue')

plt.xlabel('Date')

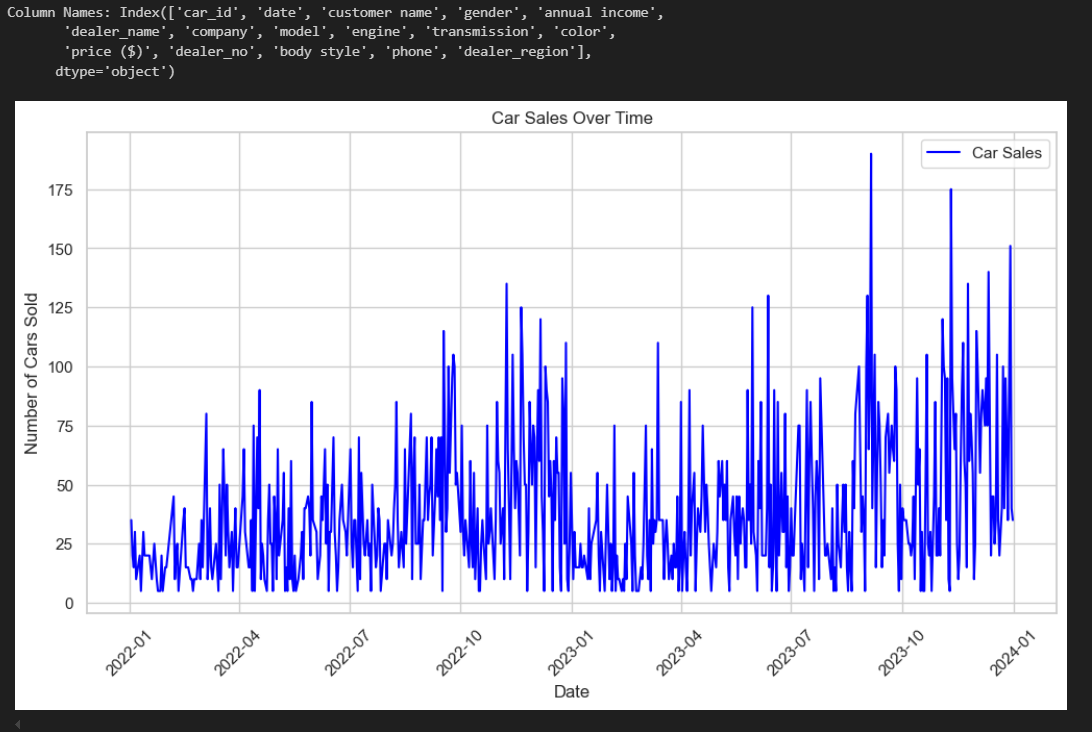
plt.ylabel('Number of Cars Sold')

plt.title('Car Sales Over Time')

plt.xticks(rotation=45)

plt.show()

**OUTPUT**



**RESULT**

Thus the program has been implemented successfully.