1. IMPLEMENT PROGRAMS FOR TIME SERIES DATA CLEANING, LOADING AND HANDLING TIMES SERIES DATA AND PRE-PROCESSING TECHNIQUES.

EX.N0:01

DATE: 20/01/2025

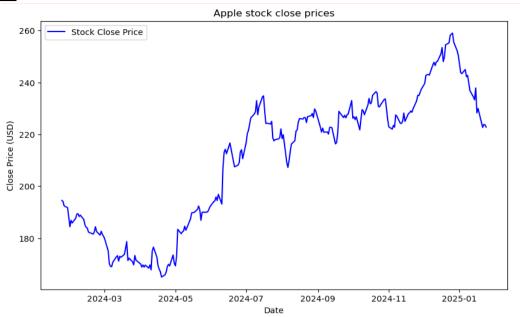
Implement programs for time series data cleaning, loading and handling times series data and preprocessing techniques.

<u>**AIM:**</u> To Implement programs for time series data cleaning, loading and handling times series data and preprocessing techniques.

PROGRAM:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
def load data(file path):
  df = pd.read csv(file path, parse dates=['Date'])
  df.set index('Date', inplace=True)
  return df
def clean_data(df):
  df.replace(',', ", regex=True, inplace=True) # Remove commas from numeric values
  df = df.astype(float) # Convert all values to float
  df = df.dropna() # Drop missing values
  df = df[df > 0] # Remove negative or zero values if present
  return df
def handle missing data(df):
  df.fillna(method='ffill', inplace=True) # Forward fill
  df.fillna(method='bfill', inplace=True) # Backward fill
  return df
def preprocess_data(df):
  df['Return'] = df['Close'].pct change() # Calculate daily returns
  df['Volatility'] = df['Return'].rolling(window=5).std() # Rolling volatility
  return df
def visualize data(df):
  plt.figure(figsize=(10, 6))
  plt.plot(df['Close'], label='Stock Close Price', color='blue')
  plt.title('Stock Close Price Over Time')
  plt.xlabel('Date')
  plt.ylabel('Close Price (USD)')
  plt.legend()
  plt.show()
file_path = "D:/221501511/Download Data - STOCK_US_XNAS_AAPL (1).csv"
df = load_data(file_path)
df = clean data(df)
df = handle missing data(df)
df = preprocess data(df)
visualize data(df)
print(df.head())
```

OUTPUT:



RESULT:

Thus, the program for Implement programs for time series data cleaning, loading and handling times series data and preprocessing techniques is executed successfully.