**EX:No.1** **221501105**

**21/01/25**

# PROGRAM TO IMPLEMENT TIME SERIES DATA FOR IMPORT LIBRARY, LOAD DATA, PREPROCESSING AND VISUALISING

**AIM:**

To write Program to implement time series data for import library, load data, Preprocessing and visualising.

# PROCESS:

## #Importing libraries

import pandas as pd

## # Load the stock data

file\_path = r'AAPL\_short\_volume.csv' data = pd.read\_csv(file\_path) close\_prices\_AAPL = data['Close']

## # Reverse the order of the data

close\_prices\_AAPL\_reverse = close\_prices\_AAPL.iloc[::-1]

## # Reset index to maintain the correct time series order in the plot

close\_prices\_AAPL\_reverse.reset\_index(drop=True, inplace=True)

## # 1. Handling Missing Values:

**# Check for missing values in each column**

print(data.isnull().sum())

## # Drop rows with missing values (if not too many)

data.dropna(inplace=True)

## # Fill missing values in 'Close' with the mean - Moved before outlier handling

data['Close'].fillna(data['Close'].mean(), inplace=True) # Fill NaNs in 'Close' column

**# 2. Handling Outliers:**

**# (a) Visualization: Create box plots or scatter plots to visually identify outliers. # (b) Using IQR (Interquartile Range):**

**# Calculate IQR for relevant numerical columns, e.g., 'Close'**

Q1 = data['Close'].quantile(0.25) Q3 = data['Close'].quantile(0.75)

IQR = Q3 - Q1

lower\_bound = Q1 - 1.5 \* IQR upper\_bound = Q3 + 1.5 \* IQR

## # Filter data to remove outliers

data = data[(data['Close'] >= lower\_bound) & (data['Close'] <= upper\_bound)]

## # Data preprocessing

import numpy as np

data = close\_prices\_AAPL\_reverse.values.reshape(-1, 1) # Reshape the data data\_normalized = data / np.max(data) # Normalize the data

# Split the data into training and testing sets train\_size = int(len(data\_normalized) \* 0.8) train\_data = data\_normalized[:train\_size] test\_data = data\_normalized[train\_size:]

## # Plot the line chart

import matplotlib.pyplot as plt plt.figure(figsize=(10, 6)) plt.plot(close\_prices\_AAPL\_reverse) plt.xlabel('Time')

plt.ylabel('Close Prices') plt.title('AAPL Stock Close Prices') plt.grid(True)

plt.show()

# OUTPUT:

**\**

# RESULT:

The program To write Program to implement time series data for import library, load data, Preprocessing and visualising is created and executed successfully.