

Experiment - 6

AIM: To execute pandas program to create a scatter plot of the trading stock prices of Alphabet Inc.

Pseudo code:

1) Import necessary libraries: pandas & matplotlib.
Pyplot as pd, plt.

2) Load stock data:

Read the csv file into a Dataframe with date parsing.

3) Create scatter plot:

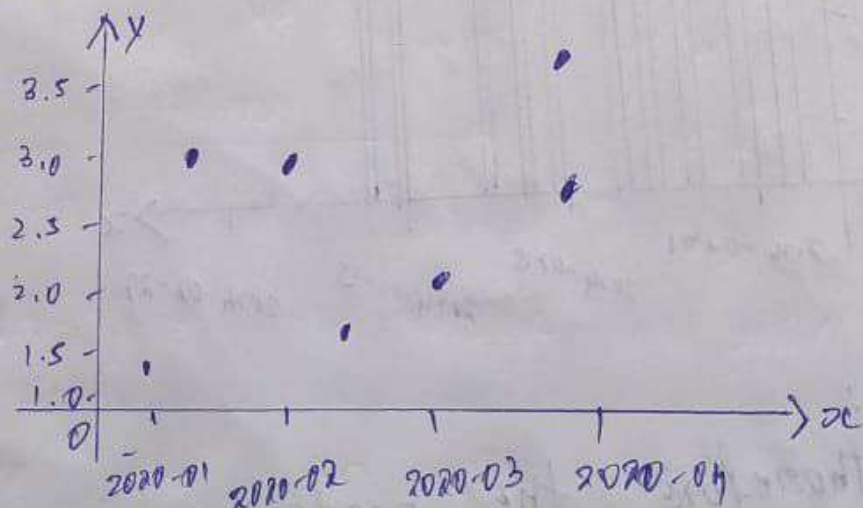
set figure (v,t) set plot title, label axis.

4) Display the plot.

Sample input:

Alphabet inc. database

Sample output:



Result:

Therefore pandas program execution for scatter plot successfully.

```

import pandas as pd
import matplotlib.pyplot as plt

# Load the CSV file containing stock data
alphabet_stock_data = pd.read_csv("C:/Users/abhip/OneDrive/Documents/DSA05 LAB/alphabet.csv")

# Convert 'Date' to datetime format
alphabet_stock_data['Date'] = pd.to_datetime(alphabet_stock_data['Date'], dayfirst=True)

# Filter data between specific dates
start_date = '2020-04-01'
end_date = '2020-05-01'
filtered_data = alphabet_stock_data[(alphabet_stock_data['Date'] >= start_date) & (alphabet_stock_data['Date'] <= end_date)]

# Create a scatter plot of trading volume vs stock price (Close)
plt.figure(figsize=(10, 6))
plt.scatter(filtered_data['Volume'], filtered_data['Close'], alpha=0.5)
plt.title('Alphabet Inc. Stock: Trading Volume vs Stock Price (April 2020)')
plt.xlabel('Trading Volume')
plt.ylabel('Stock Price (Close)')
plt.grid(True)
plt.show()

```

```

Date,Open,High,Low,Close,Adj Close,Volume
01-04-2020,1122,1129.69,1097.45,1105.62,1105.62,2343100
02-04-2020,1098.26,1122.86,1092.12,1117.89,1117.89,2154900
03-04-2020,1119.015,1120.86,1079.88,1097.88,1097.88,1983800
06-04-2020,1138,1194.66,1130.94,1186.92,1186.92,2664700
07-04-2020,1165.61,1185.21,1150.61,1186.58,1186.58,2847300
08-04-2020,1183,1193,1172,1182.11,1182.11,1975100
09-04-2020,1204.48,1225.57,1196.735,1211.45,1211.45,2175400
13-04-2020,1222,1225.57,1198.93,1217.56,1217.56,2174500
14-04-2020,1231.2,1254.41,1220.15,1250.41,1250.41,2360400
15-04-2020,1245.61,1280.46,1224.42,1262.47,1262.47,1671700
16-04-2020,1274,1280.46,1240.4,1246.27,1246.27,2515000
17-04-2020,1288.35,1287,1259.37,1266.61,1266.61,1955800
20-04-2020,1271,1278,1261.71,1266.61,1266.61,1600000

```

