Untitled1

March 15, 2023

FATIMA

[1]:
```
!pip install yfinance
!pip install pandas
!pip install requests
!pip install bs4
!pip install plotly
```

```
Collecting yfinance
  Downloading yfinance-0.2.12-py2.py3-none-any.whl (59 kB)
                             59.2/59.2 kB
8.4 MB/s eta 0:00:00
Requirement already satisfied: cryptography>=3.3.2 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from yfinance)
(38.0.2)
Requirement already satisfied: pytz>=2022.5 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from yfinance)
(2022.6)
Collecting appdirs>=1.4.4
  Downloading appdirs-1.4.4-py2.py3-none-any.whl (9.6 kB)
Collecting html5lib>=1.1
  Downloading html5lib-1.1-py2.py3-none-any.whl (112 kB)
                            112.2/112.2 kB
15.4 MB/s eta 0:00:00
Collecting frozendict>=2.3.4
  Downloading
frozendict-2.3.5-cp37-cp37m-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (99
kB)
                             99.8/99.8 kB
12.6 MB/s eta 0:00:00
Collecting multitasking>=0.0.7
  Downloading multitasking-0.0.11-py3-none-any.whl (8.5 kB)
Requirement already satisfied: lxml>=4.9.1 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from yfinance)
(4.9.1)
Requirement already satisfied: numpy>=1.16.5 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from yfinance)
(1.21.6)
Requirement already satisfied: pandas>=1.3.0 in
```

```
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from yfinance)
(1.3.5)
Requirement already satisfied: requests>=2.26 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from yfinance)
(2.28.1)
Requirement already satisfied: beautifulsoup4>=4.11.1 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from yfinance)
(4.11.1)
Requirement already satisfied: soupsieve>1.2 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from
beautifulsoup4>=4.11.1->yfinance) (2.3.2.post1)
Requirement already satisfied: cffi>=1.12 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from
cryptography>=3.3.2->yfinance) (1.15.1)
Requirement already satisfied: webencodings in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from
html5lib>=1.1->yfinance) (0.5.1)
Requirement already satisfied: six>=1.9 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from
html5lib>=1.1->yfinance) (1.16.0)
Requirement already satisfied: python-dateutil>=2.7.3 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from
pandas>=1.3.0->yfinance) (2.8.2)
Requirement already satisfied: charset-normalizer<3,>=2 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from
requests>=2.26->yfinance) (2.1.1)
Requirement already satisfied: certifi>=2017.4.17 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from
requests>=2.26->yfinance) (2022.9.24)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from
requests>=2.26->yfinance) (1.26.13)
Requirement already satisfied: idna<4,>=2.5 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from
requests>=2.26->yfinance) (3.4)
Requirement already satisfied: pycparser in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from
cffi>=1.12->cryptography>=3.3.2->yfinance) (2.21)
Installing collected packages: multitasking, appdirs, html5lib, frozendict,
yfinance
Successfully installed appdirs-1.4.4 frozendict-2.3.5 html5lib-1.1
multitasking-0.0.11 yfinance-0.2.12
Requirement already satisfied: pandas in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (1.3.5)
Requirement already satisfied: python-dateutil>=2.7.3 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from pandas)
(2.8.2)
Requirement already satisfied: pytz>=2017.3 in
```

```
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from pandas)
(2022.6)
Requirement already satisfied: numpy>=1.17.3 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from pandas)
(1.21.6)
Requirement already satisfied: six>=1.5 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from python-
dateutil>=2.7.3->pandas) (1.16.0)
Requirement already satisfied: requests in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (2.28.1)
Requirement already satisfied: charset-normalizer<3,>=2 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from requests)
(2.1.1)
Requirement already satisfied: certifi>=2017.4.17 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from requests)
(2022.9.24)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from requests)
(1.26.13)
Requirement already satisfied: idna<4,>=2.5 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from requests)
(3.4)
Collecting bs4
  Downloading bs4-0.0.1.tar.gz (1.1 kB)
  Preparing metadata (setup.py) … done
Requirement already satisfied: beautifulsoup4 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from bs4)
(4.11.1)
Requirement already satisfied: soupsieve>1.2 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from
beautifulsoup4->bs4) (2.3.2.post1)
Building wheels for collected packages: bs4
  Building wheel for bs4 (setup.py) … done
  Created wheel for bs4: filename=bs4-0.0.1-py3-none-any.whl size=1256
sha256=2090ab219010b21a8af986d0440442f0a6e1a57325cf01ac53aa48b16e602ee5
  Stored in directory: /home/jupyterlab/.cache/pip/wheels/77/8a/04/7b1a8ce5de655
5a18e09370d3d4fde48be9571ac07a623071e
Successfully built bs4
Installing collected packages: bs4
Successfully installed bs4-0.0.1
Requirement already satisfied: plotly in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (5.11.0)
Requirement already satisfied: tenacity>=6.2.0 in
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from plotly)
(8.1.0)
```

[2]:
```
import yfinance as yf
import pandas as pd
import requests
from bs4 import BeautifulSoup
import plotly.graph_objects as go
from plotly.subplots import make_subplots
```

[3]:
```
def make_graph(stock_data, revenue_data, stock):
    fig = make_subplots(rows=2, cols=1, shared_xaxes=True,␣
 ↪subplot_titles=("Historical Share Price", "Historical Revenue"),␣
 ↪vertical_spacing = .3)
    fig.add_trace(go.Scatter(x=pd.to_datetime(stock_data.Date,␣
 ↪infer_datetime_format=True), y=stock_data.Close.astype("float"), name="Share␣
 ↪Price"), row=1, col=1)
    fig.add_trace(go.Scatter(x=pd.to_datetime(revenue_data.Date,␣
 ↪infer_datetime_format=True), y=revenue_data.Revenue.astype("float"),␣
 ↪name="Revenue"), row=2, col=1)
    fig.update_xaxes(title_text="Date", row=1, col=1)
    fig.update_xaxes(title_text="Date", row=2, col=1)
    fig.update_yaxes(title_text="Price ($US)", row=1, col=1)
    fig.update_yaxes(title_text="Revenue ($US Millions)", row=2, col=1)
    fig.update_layout(showlegend=False,
    height=900,
    title=stock,
    xaxis_rangeslider_visible=True)
    fig.show()
```

Question 1: Use yfinance to Extract Stock Data

[4]:
```
Tesla = yf.Ticker('TSLA')
```

[5]:
```
tesla_data = Tesla.history(period = "max")
```

[6]:
```
tesla_data.reset_index(inplace = True)
tesla_data.head()
```

[6]:

	Date	Open	High	Low	Close	Volume	Dividends	Stock Splits
0	2010-06-29 00:00:00-04:00	1.266667	1.666667	1.169333	1.592667	281494500	0.0	0.0
1	2010-06-30 00:00:00-04:00	1.719333	2.028000	1.553333	1.588667	257806500	0.0	0.0
2	2010-07-01 00:00:00-04:00	1.666667	1.728000	1.351333	1.464000	123282000	0.0	0.0
3	2010-07-02 00:00:00-04:00	1.533333	1.540000	1.247333	1.280000			
4	2010-07-06 00:00:00-04:00	1.333333	1.333333	1.055333	1.074000			

```
3   77097000        0.0           0.0
4  103003500        0.0           0.0
```

Question 2: Use Webscraping to Extract Tesla Revenue Data

[7]:
```
url = "https://www.macrotrends.net/stocks/charts/TSLA/tesla/revenue"
html_data = requests.get(url).text
```

[8]:
```
soup = BeautifulSoup(html_data, "html.parser")
soup.find_all('title')
```

[8]:
```
[<title>Tesla Revenue 2010-2022 | TSLA | MacroTrends</title>]
```

[9]:
```
tesla_revenue = pd.DataFrame(columns = ['Date', 'Revenue'])

for row in soup.find_all("tbody")[1].find_all("tr"):
    col = row.find_all("td")
    date = col[0].text
    revenue = col[1].text.replace("$", "").replace(",", "")

    tesla_revenue = tesla_revenue.append({"Date": date, "Revenue": revenue},␣
 ↪ignore_index = True)
```

[10]:
```
tesla_revenue.dropna(inplace=True)
tesla_revenue = tesla_revenue[tesla_revenue['Revenue'] != ""]
```

[11]:
```
tesla_revenue.tail()
```

[11]:
```
          Date Revenue
49  2010-09-30      31
50  2010-06-30      28
51  2010-03-31      21
53  2009-09-30      46
54  2009-06-30      27
```

Question 3: Use yfinance to Extract Stock Data

[12]:
```
GameStop = yf.Ticker("GME")
```

[13]:
```
gme_data = GameStop.history(period = 'max')
```

[14]:
```
gme_data.reset_index(inplace = True)
gme_data.head()
```

[14]:
```
                       Date      Open      High       Low     Close    Volume  \
0 2002-02-13 00:00:00-05:00  1.620129  1.693350  1.603296  1.691667  76216000
1 2002-02-14 00:00:00-05:00  1.712707  1.716074  1.670626  1.683250  11021600
2 2002-02-15 00:00:00-05:00  1.683250  1.687458  1.658002  1.674834   8389600
```

```
3 2002-02-19 00:00:00-05:00  1.666418  1.666418  1.578047  1.607504   7410400
4 2002-02-20 00:00:00-05:00  1.615921  1.662210  1.603296  1.662210   6892800

   Dividends  Stock Splits
0        0.0           0.0
1        0.0           0.0
2        0.0           0.0
3        0.0           0.0
4        0.0           0.0
```

Question 4: Use Webscraping to Extract GME Revenue Data

[15]:
```
url = "https://www.macrotrends.net/stocks/charts/GME/gamestop/revenue"
html_data = requests.get(url).text
```

[16]:
```
soup = BeautifulSoup(html_data, "html.parser")
soup.find_all('title')
```

[16]:
```
[<title>GameStop Revenue 2010-2022 | GME | MacroTrends</title>]
```

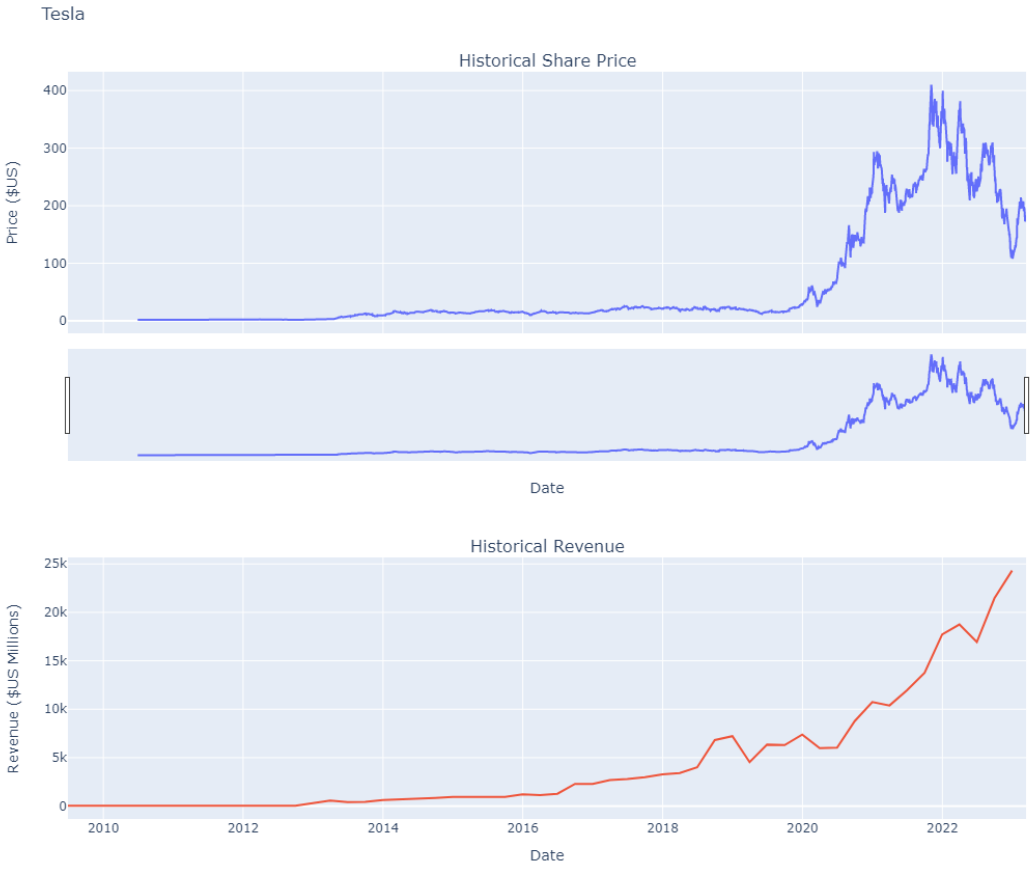
[17]:
```
gme_revenue = pd.DataFrame(columns = ['Date', 'Revenue'])

for row in soup.find_all("tbody")[1].find_all("tr"):
    col = row.find_all("td")
    date = col[0].text
    revenue = col[1].text.replace("$", "").replace(",", "")

    gme_revenue = gme_revenue.append({"Date": date, "Revenue": revenue},␣
 ↪ignore_index = True)
```

[18]:
```
tesla_revenue.dropna(inplace=True)
tesla_revenue = tesla_revenue[tesla_revenue['Revenue'] != ""]
gme_revenue.tail()
```

[18]:
```
          Date Revenue
51  2010-01-31    3524
52  2009-10-31    1835
53  2009-07-31    1739
54  2009-04-30    1981
55  2009-01-31    3492
```

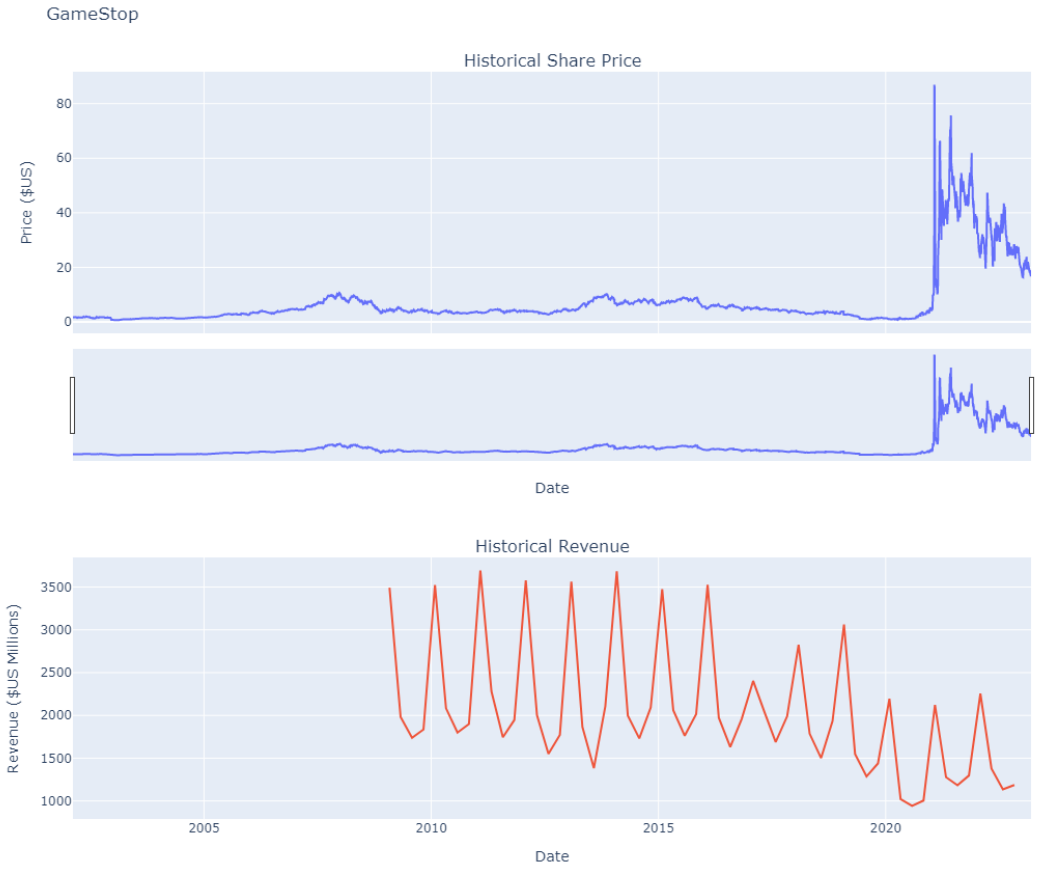
Question 5: Plot Tesla Stock Graph

[19]:
```
make_graph(tesla_data, tesla_revenue, 'Tesla')
```



Question 6: Plot GameStop Stock Graph

[20]:
```
make_graph(gme_data, gme_revenue, 'GameStop')
```


GameStop
Historical Share Price
Price ($US)
80
60
40
20
0
Date
Historical Revenue
Revenue ($US Millions)
3500
3000
2500
2000
1500
1000
2005
2010
2015
2020
Date

[ ]: