

```
In [3]: !pip install yfinance
        #!pip install pandas
        #!pip install requests
        !pip install bs4
        #!pip install plotly

Requirement already satisfied: yfinance in c:\python\lib\site-packages (0.1.67)
Requirement already satisfied: pandas>=0.24 in c:\python\lib\site-packages (from yfinance) (1.2.4)
Requirement already satisfied: numpy>=1.15 in c:\python\lib\site-packages (from yfinance) (1.20.1)
Requirement already satisfied: lxml>=4.5.1 in c:\python\lib\site-packages (from yfinance) (4.6.4)
Requirement already satisfied: multitasking>=0.0.7 in c:\python\lib\site-packages (from yfinance) (0.0.10)
Requirement already satisfied: requests>=2.20 in c:\python\lib\site-packages (from yfinance) (2.25.1)
Requirement already satisfied: python-dateutil>=2.7.3 in c:\python\lib\site-packages (from pandas>=0.24->yfinance) (2.8.1)
Requirement already satisfied: pytz>=2017.3 in c:\python\lib\site-packages (from pandas>=0.24->yfinance) (2021.1)
Requirement already satisfied: six>=1.5 in c:\python\lib\site-packages (from python-dateutil>=2.7.3->pandas>=0.24->yfinance) (1.15.0)
Requirement already satisfied: certifi>=2017.4.17 in c:\python\lib\site-packages (from requests>=2.20->yfinance) (2020.12.5)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\python\lib\site-packages (from requests>=2.20->yfinance) (1.26.4)
Requirement already satisfied: idna<3,>=2.5 in c:\python\lib\site-packages (from requests>=2.20->yfinance) (2.10)
Collecting bs4
  Downloading bs4-0.0.1.tar.gz (1.1 kB)
Requirement already satisfied: beautifulsoup4 in c:\python\lib\site-packages (from bs4) (4.9.3)
Requirement already satisfied: soupsieve>1.2 in c:\python\lib\site-packages (from beautifulsoup4->bs4) (2.2.1)
Building wheels for collected packages: bs4
  Building wheel for bs4 (setup.py): started
  Building wheel for bs4 (setup.py): finished with status 'done'
  Created wheel for bs4: filename=bs4-0.0.1-py3-none-any.whl size=1273 sha256=bc181c4032ddb97fce3236edda45c55811380219b64a9735aec4d35a58ba6c0b
Stored in directory: c:\users\anush\appdata\local\pip\cache\wheels\75\78\21\68b124549c9bdc94f822c02fb9aa3578a669843f9767776bca
Successfully built bs4
Installing collected packages: bs4
Successfully installed bs4-0.0.1
```

```
In [4]: 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
```

```
In [5]: tesla = yf.Ticker("TSLA")
```

```
In [6]: tesla_data = tesla.history(period="max")
```

```
In [7]: tesla_data.reset_index(inplace=True)
tesla_data.head()
```

	Date	Open	High	Low	Close	Volume	Dividends	Stock Splits
0	2010-06-29	3.800	5.000	3.508	4.778	93831500	0	0.0
1	2010-06-30	5.158	6.084	4.660	4.766	85935500	0	0.0
2	2010-07-01	5.000	5.184	4.054	4.392	41094000	0	0.0
3	2010-07-02	4.600	4.620	3.742	3.840	25699000	0	0.0
4	2010-07-06	4.000	4.000	3.166	3.222	34334500	0	0.0

```
In [8]: url = "https://www.macrotrends.net/stocks/charts/TSLA/tesla/revenue"
html_data = requests.get(url).text
```

```
In [9]: soup = BeautifulSoup(html_data, 'html5lib')
```

```
In [10]: tesla_revenue=pd.read_html(url, match="Tesla Quarterly Revenue", flavor='bs4')[0]
tesla_revenue.tail()
```

	Tesla Quarterly Revenue(Millions of US \$)	Tesla Quarterly Revenue(Millions of US \$).1
48	2010-06-30	\$28
49	2010-03-31	\$21
50	2009-12-31	NaN
51	2009-09-30	\$46
52	2009-06-30	\$27

```
In [11]: game_stop = yf.Ticker("GME")
```

```
In [12]: gme_data = game_stop.history(period="max")
```

```
In [13]: gme_data.reset_index(inplace=True)
gme_data.head()
```

	Date	Open	High	Low	Close	Volume	Dividends	Stock Splits
0	2002-02-13	1.620128	1.693350	1.603296	1.691666	76216000	0.0	0.0
1	2002-02-14	1.712707	1.716073	1.670626	1.683250	11021600	0.0	0.0
2	2002-02-15	1.683251	1.687459	1.658002	1.674834	8389600	0.0	0.0
3	2002-02-19	1.666418	1.666418	1.578047	1.607504	7410400	0.0	0.0
4	2002-02-20	1.615920	1.662209	1.603296	1.662209	6892800	0.0	0.0

```
In [14]: url = "https://www.macrotrends.net/stocks/charts/GME/gamestop/revenue"
html_data = requests.get(url).text
```

```
In [15]: soup = BeautifulSoup(html_data, 'html5lib')
```

```
In [16]: gme_revenue=pd.read_html(url,match="GameStop Quarterly Revenue", flavor='bs4')[0]
gme_revenue.tail()
```

	GameStop Quarterly Revenue(Millions of US \$)	GameStop Quarterly Revenue(Millions of US \$).1
49	2010-01-31	\$3,524
50	2009-10-31	\$1,835
51	2009-07-31	\$1,739
52	2009-04-30	\$1,981
53	2009-01-31	\$3,492

```
In [22]: import yfinance as yf
import pandas as pd

stock_data = yf.download("TSLA", start="2020-01-01", end="2021-09-30", progress=False)
revenue_data = yf.download("TSLA", start="2020-01-01", end="2021-09-30", progress=False)
stock_data.reset_index(inplace=True)
revenue_data.reset_index(inplace=True)

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)

    stock_data_specific = stock_data[stock_data.Date <= '2021-06-14']
    revenue_data_specific = revenue_data[revenue_data.Date <= '2021-04-30']

    fig.add_trace(go.Scatter(
        x=pd.to_datetime(stock_data_specific.Date, infer_datetime_format=True),
        y=stock_data_specific.Close.astype("float"), name="Share Price"), row=1, col=1)

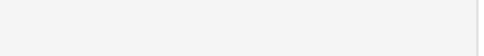
    fig.add_trace(go.Scatter(x=pd.to_datetime(revenue_data_specific.Date, infer_datetime_format=True),
        y=revenue_data_specific.Volume.astype("float"),
        name="Volume"), row=2, col=1)

    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()

make_graph(stock_data, revenue_data, 'TSLA')
```



```
In [33]: import yfinance as yf
import pandas as pd

stock_data = yf.download("GME", start="2020-01-01", end="2021-09-30", progress=False)
revenue_data = yf.download("GME", start="2020-01-01", end="2021-09-30", progress=False)
stock_data.reset_index(inplace=True)
revenue_data.reset_index(inplace=True)

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)

    stock_data_specific = stock_data[stock_data.Date <= '2021-06-14']
    revenue_data_specific = revenue_data[revenue_data.Date <= '2021-04-30']

    fig.add_trace(go.Scatter(
        x=pd.to_datetime(stock_data_specific.Date, infer_datetime_format=True),
        y=stock_data_specific.Close.astype("float"), name="Share Price"), row=1, col=1)

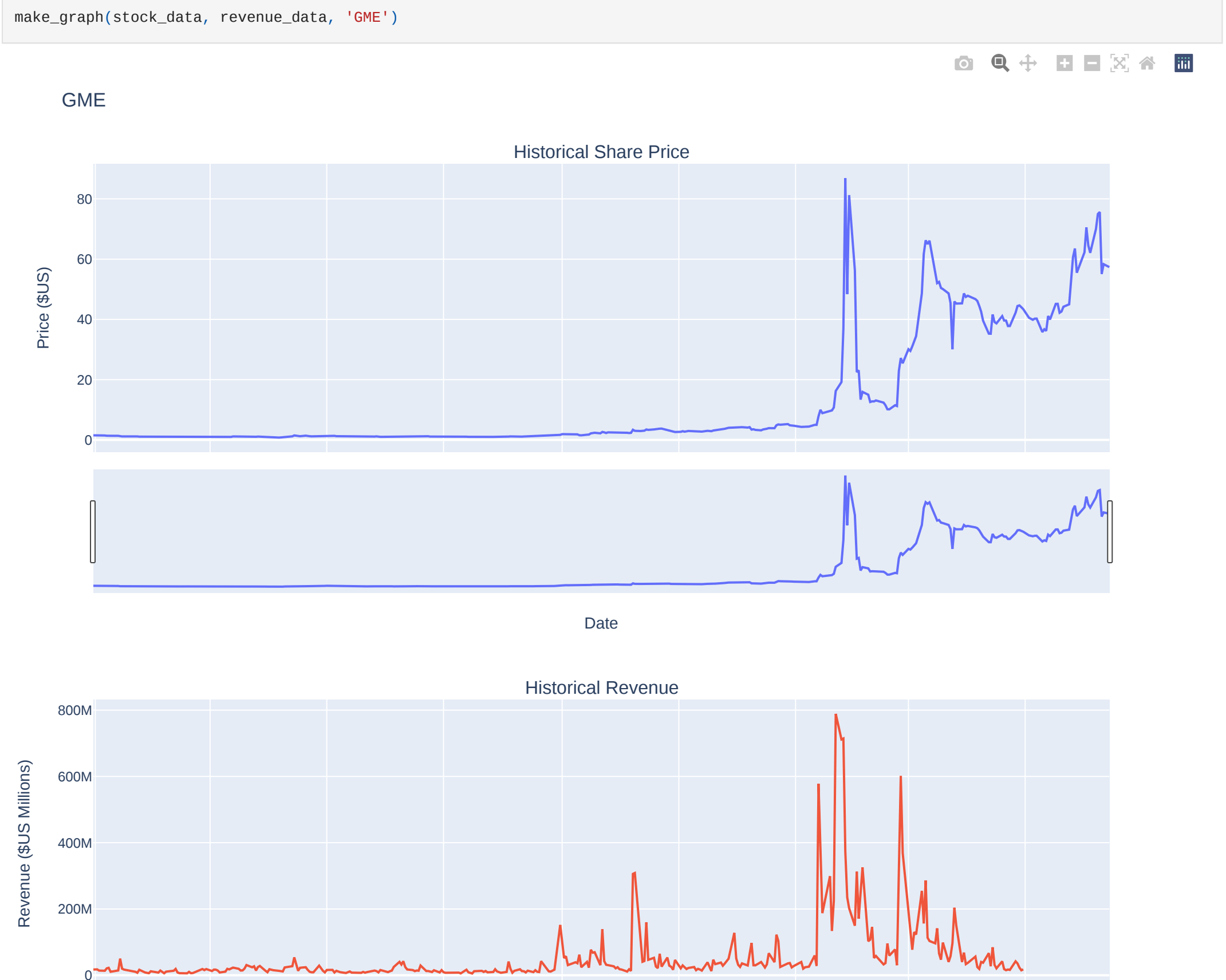
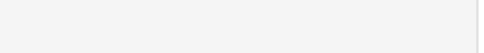
    fig.add_trace(go.Scatter(x=pd.to_datetime(revenue_data_specific.Date, infer_datetime_format=True),
        y=revenue_data_specific.Volume.astype("float"),
        name="Volume"), row=2, col=1)

    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()

make_graph(stock_data, revenue_data, 'GME')
```



```
In [ ]:
```