

# Analyzing Tesla and Gamestop's Historical Stock and Revenue Data and Building a Dashboard

## ***Tesla:***

```
!pip install yfinance==0.1.67
```

```
!mamba install bs4==4.10.0 -y
```

```
!pip install nbformat==4.2.0
```

```
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
```

```
# Define the function make_graph
```

```
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.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_rangeflider_visible=True)
```

```
fig.show()
```

```
# Use Yfinance to extract Tesla's stock data
```

```
# Create a ticker object, "tesla", that takes in the stock data using the key "TSLA"
```

```
tesla = yf.Ticker("TSLA")
```

```
# Extract stock information and save it in a dataframe named tesla_data. The period parameter is set to max so we get information for the maximum amount of time.
```

```
tesla_data = tesla.history(period="max")
```

```
tesla_data
```

```
# Output:
```

	Open	High	Low	Close	Volume	Dividends	Stock Splits
Date							
2010-06-29	1.266667	1.666667	1.169333	1.592667	281494500	0	0.0
2010-06-30	1.719333	2.028000	1.553333	1.588667	257806500	0	0.0
2010-07-01	1.666667	1.728000	1.351333	1.464000	123282000	0	0.0
2010-07-02	1.533333	1.540000	1.247333	1.280000	77097000	0	0.0
2010-07-06	1.333333	1.333333	1.055333	1.074000	103003500	0	0.0
...	...	...	...	...	...	...	...
2023-02-28	210.589996	211.229996	203.750000	205.710007	153144900	0	0.0
2023-03-01	206.210007	207.199997	198.520004	202.770004	156852800	0	0.0
2023-03-02	186.740005	193.750000	186.009995	190.899994	181500700	0	0.0
2023-03-03	194.800003	200.479996	192.880005	197.789993	153800400	0	0.0
2023-03-06	198.539993	198.567993	192.309998	193.809998	127504923	0	0.0

2103 rows x 7 columns

```
# Reset the index and display the first 5 rows
```

	index	Date	Open	High	Low	Close	Volume	Dividends	Stock Splits
0	0	2010-06-29	1.266667	1.666667	1.169333	1.592667	281494500	0	0.0
1	1	2010-06-30	1.719333	2.028000	1.553333	1.588667	257806500	0	0.0
2	2	2010-07-01	1.666667	1.728000	1.351333	1.464000	123282000	0	0.0
3	3	2010-07-02	1.533333	1.540000	1.247333	1.280000	77097000	0	0.0
4	4	2010-07-06	1.333333	1.333333	1.055333	1.074000	103003500	0	0.0

```
# Use webscraping to extract Tesla Revenue Data
```

```
# Download the webpage
```

```
https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDeveloperSkillsNetwork-PY0220EN-SkillsNetwork/labs/project/revenue.html and save the text as a variable, html_data
```

```
import requests
```

```
from bs4 import BeautifulSoup
```

```
url =
```

```
"https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDeveloperSkillsNetwork-PY0220EN-SkillsNetwork/labs/project/revenue.htm"
```

```
html_data = requests.get(url).text
```

```
# Parse the html data using BeautifulSoup
soup = BeautifulSoup(html_data,'html.parser')
tag_object=soup.title
print("tag object:",tag_object)
```

```
# Output:
tag object: <title>Tesla Revenue 2010-2022 | TSLA | MacroTrends</title>
```

```
# Use BeautifulSoup to extract the table with Tesla Quarterly Revenue and store it into a
dataframe named tesla_revenue, with columns Date and Revenue
```

```
tesla_revenue = pd.DataFrame(columns=["Date" , "Revenue"])
for table in soup.find_all('table'):
    if table.find('th').getText().startswith("Tesla Quarterly Revenue"):
        for row in table.find("tbody").find_all("tr"):
            col = row.find_all("td")
            date = col[0].text
            revenue = col[1].text
            tesla_revenue = tesla_revenue.append({"Date":date, "Revenue":revenue},
            ignore_index=True)
tesla_revenue.head()
```

```
# Output:
```

```
:      Date  Revenue
0  2022-09-30  $21,454
1  2022-06-30  $16,934
2  2022-03-31  $18,756
3  2021-12-31  $17,719
4  2021-09-30  $13,757
```

```
# Remove the comma and dollar sign from the Revenue column
tesla_revenue["Revenue"] = tesla_revenue['Revenue'].str.replace(',|\$','')
```

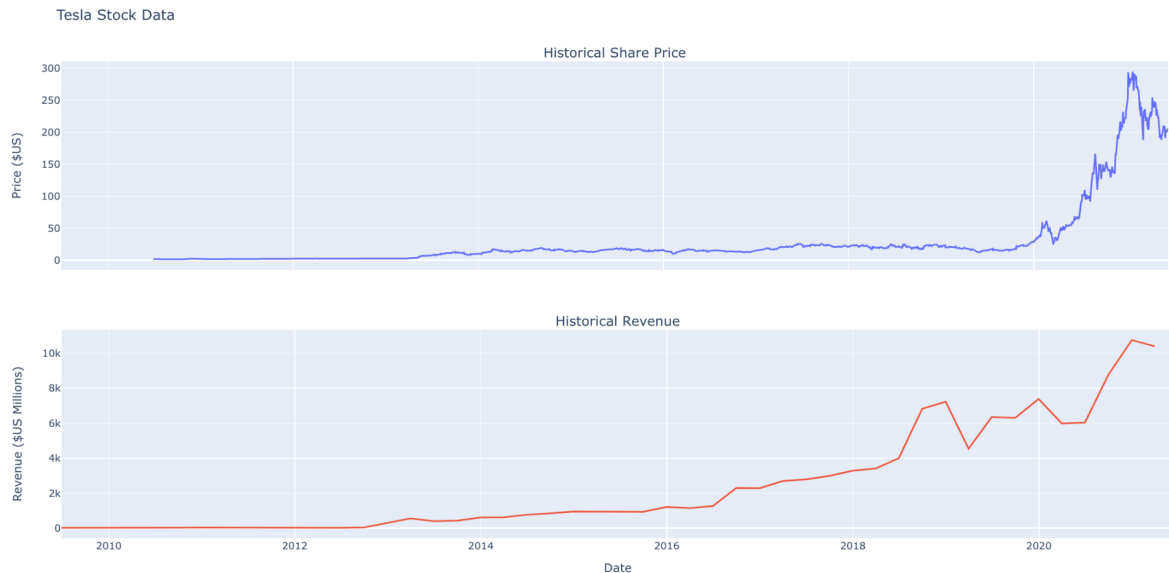
```
# Remove an null or empty strings in the Revenue column
tesla_revenue.dropna(inplace=True)
tesla_revenue = tesla_revenue[tesla_revenue['Revenue'] != ""]
tesla_revenue.tail(5)
```

```
# Output:
```

```
      Date  Revenue
48  2010-09-30      31
49  2010-06-30      28
50  2010-03-31      21
52  2009-09-30      46
53  2009-06-30      27
```

# Graph the Tesla Stock Data. The graph will only show data up to June 2021  
make\_graph(tesla\_data, tesla\_revenue, 'Tesla Stock Data')

# Output:



# Analysis:

#### Tesla's Historical Share Price:

During the lockdown in the beginning of 2020, all stock prices decreased, including Teslas, since people had no desire to buy cars at the time. Shortly after the lockdown was lifted, people began to buy cars once again. However, production took a long time to recover, thereby increasing stock prices. In 2021, people started to realize that Tesla's stock price was exaggerated and overhyped, which likely caused the decline.

#### Tesla's Historical Revenue:

Tesla started to deliver the new model 3 in 2017. However, they struggled to produce enough cars to meet the market demand. They began to overcome their production issues in 2018, allowing them to sell many more cars to cover the demand, which is likely the reason for the spike shown in late 2018. The government federal tax incentives ended during the 3rd and 4th quarter of 2019, which is likely the cause for the decline in revenue. However, Tesla continued to gain in popularity, which is likely the reason their revenue continued to increase following the decline of late 2019. The sharp revenue increase in mid-2020 is likely due to the end of lockdown, in which the demand came back in large, but slowed production during the lockdown caused a shortage, thereby causing prices to spike by tens of percentages.

## Gamestop:

# Use Yfinance to extract Gamestop's stock data

# Create a ticker object, "gamestop", that takes in the stock data using the key "GME"  
gamestop = yf.Ticker("GME")

# Extract stock information and save it in a dataframe named **gme\_data**. The **period** parameter is set to **max** so we get information for the maximum amount of time.

gme\_data = gamestop.history(period="max")

Gme\_data

# Output:

	Open	High	Low	Close	Volume	Dividends	Stock Splits
Date							
2002-02-13	1.620128	1.693350	1.603296	1.691666	76216000	0.0	0.0
2002-02-14	1.712707	1.716074	1.670626	1.683251	11021600	0.0	0.0
2002-02-15	1.683250	1.687458	1.658002	1.674834	8389600	0.0	0.0
2002-02-19	1.666418	1.666418	1.578047	1.607504	7410400	0.0	0.0
2002-02-20	1.615920	1.662209	1.603296	1.662209	6892800	0.0	0.0
...	...	...	...	...	...	...	...
2023-04-24	20.150000	20.410000	19.809999	19.930000	2742400	0.0	0.0
2023-04-25	19.740000	20.040001	18.910000	19.000000	2560800	0.0	0.0
2023-04-26	18.980000	19.230000	18.650000	18.650000	2287600	0.0	0.0
2023-04-27	18.750000	19.340000	18.719999	18.940001	2238900	0.0	0.0
2023-04-28	18.920000	19.719999	18.879999	19.290001	2835200	0.0	0.0

# Reset the index and display the first 5 rows

gme\_data.reset\_index(inplace=True)

gme\_data.head(5)

# Output:

	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	1.716074	1.670626	1.683251	11021600	0.0	0.0
2	2002-02-15	1.683250	1.687458	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

```
# Use webscraping to extract Gamestop Revenue Data
```

```
# Download the webpage
```

```
https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDeveloperSkillsNetwork-PY0220EN-SkillsNetwork/labs/project/stock.html and save the text as a variable named html_data
```

```
url =
```

```
"https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDeveloperSkillsNetwork-PY0220EN-SkillsNetwork/labs/project/stock.html"
```

```
html_data = requests.get(url).text
```

```
soup = BeautifulSoup(html_data, 'html.parser')
```

```
# Use BeautifulSoup to extract the table with Gamestop Quarterly Revenue and store it into a dataframe named gme_revenue, with columns Date and Revenue
```

```
gme_revenue = pd.DataFrame(columns=["Date", "Revenue"])
```

```
for table in soup.find_all('table'):
```

```
    if table.find('th').getText().startswith("GameStop Quarterly Revenue"):
```

```
        for row in table.find("tbody").find_all("tr"):
```

```
            col = row.find_all("td")
```

```
            date = col[0].text
```

```
            revenue = col[1].text
```

```
            gme_revenue = gme_revenue.append({"Date":date, "Revenue":revenue},
```

```
ignore_index=True)
```

```
gme_revenue["Revenue"] = gme_revenue["Revenue"].str.replace(',|\$', '')
```

```
gme_revenue.dropna(inplace=True)
```

```
gme_revenue = gme_revenue[gme_revenue["Revenue"] != ""]
```

```
gme_revenue.head(5)
```

```
gme_revenue.tail(5)
```

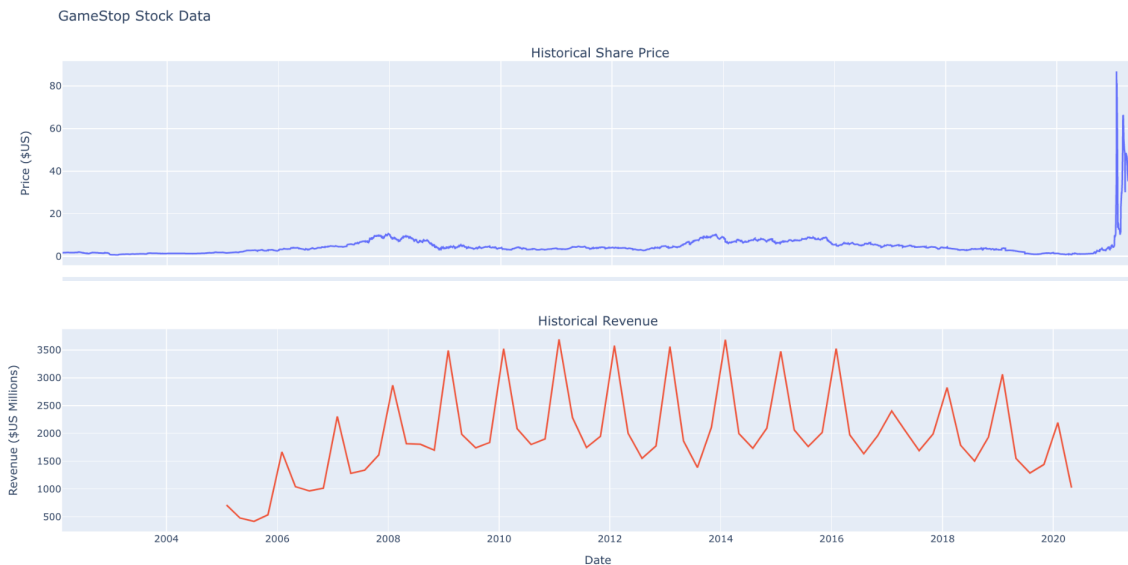
```
# Output:
```

	Date	Revenue		Date	Revenue
0	2020-04-30	1021	57	2006-01-31	1667
1	2020-01-31	2194	58	2005-10-31	534
2	2019-10-31	1439	59	2005-07-31	416
3	2019-07-31	1286	60	2005-04-30	475
4	2019-04-30	1548	61	2005-01-31	709

```
# Graph the Gamestop Stock Data. The graph will only show data up to June 2021
make_graph(gme_data, gme_revenue, 'GameStop Stock Data')
```

```
# Output:
```

Gamestop Stock Data



## # Analysis:

### Gamestop's Historical Share Price:

Gamestop's historical share price shows a mostly constant slope hovering around stock prices of \$2-\$5 with a couple of minor increases in 2008 and between 2014-2016, showing prices rise to about \$10. It wasn't until 2021 that Gamestop experienced an immense spike. In 2021, many people short-sold the stock under the assumption that the stock price would go down. Thanks in part to a subreddit group and a number of hedge funds who decided to make several posts and articles persuading people to purchase the Gamestop shares, the number of short-selling increased dramatically and triggered a dramatic increase in stock price. This daunting price increase caused short sellers to attempt to cover their positions by buying the stock, causing the demand to increase so much that it was far higher than Gamestop's supply. A short-squeeze took place.

### Gamestop's Historical Revenue:

Gamestop's up & down historical revenue is likely due to two things: seasonal market changes and the release of new gaming consoles and games. It seems that a single increase in revenue happens around the beginning of the holiday season and then lowers down in a pretty consistent manner each year. This suggests that more products are bought during the holiday season and tend to be at a low throughout the rest of the year. Additionally, when new consoles or popular games are released, there is a surge in demand, leading to higher revenues. Games are typically released at least once a year, and consoles- every few years, often during holiday season.