

```
File Edit View Insert Cell Kernel Help
```

```
xaxis_rangeslider_visible=True)
fig.show()
```

Question 1: Use yfinance to Extract Stock Data.

Using the Ticker function enter the ticker symbol of the stock we want to extract data on to create a ticker object. The stock is Tesla and its ticker symbol is TSLA.

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

Using the ticker object and the function history extract stock information and save it in a dataframe named tesla_data. Set the period parameter to max so we get information for the maximum amount of time.

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

Reset the index using the reset_index(inplace=True) function on the tesla_data DataFrame and display the first five rows of the tesla_data dataframe using the head function. Take a screenshot of the results and code from the beginning of Question 1 to the results below.

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

```
Out[7]:
```

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

Question 2: Use Webscraping to Extract Tesla Revenue Data

Activities Brave Web Browser Dec 12 14:22

Final_assignment - IBM Watson Studio Analyzing Historical Stock/Re

eu-de.dataplatform.cloud.ibm.com/analytics/notebooks/v2/43412194-e52f-48b1-8b8b-623f576f96f9?projectid=65ed1afe-ecc1-4347-8c70-6e5c91...

IBM Watson Studio Search in your workspaces Buy Prerana Patil's Account Frankfurt PP

Projects / Coursera_Notebook / Final_assignment

File Edit View Insert Cell Kernel Help Not Trusted Python 3.10

```
col = df[['Date', 'Revenue']]
date = col[0].text
revenue = col[1].text.replace(',', '').replace('$', '')

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

Execute the following line to remove the comma and dollar sign from the Revenue column.

```
In [13]: tesla_revenue["Revenue"] = tesla_revenue['Revenue'].str.replace(',', '\$', "")
```

```
/tmp/wsuser/ipykernel_601/349343550.py:1: FutureWarning: The default value of regex will change from True to False in a future version.
tesla_revenue["Revenue"] = tesla_revenue['Revenue'].str.replace(',', '\$', "")
```

Execute the following lines to remove an null or empty strings in the Revenue column.

```
In [14]: tesla_revenue.dropna(inplace=True)

tesla_revenue = tesla_revenue[tesla_revenue['Revenue'] != ""]
```

Display the last 5 row of the tesla_revenue dataframe using the tail function. Take a screenshot of the results.

```
In [15]: tesla_revenue.tail()
```

```
Out[15]:
```

	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

Question 3: Use yfinance to Extract Stock Data

Using the Ticker function enter the ticker symbol of the stock we want to extract data on to create a ticker object. The stock is GameStop and its ticker symbol is GME.

```
In [16]: GameStop = yf.Ticker("GME")
```

Using the ticker object and the function history extract stock information and save it in a dataframe named gme_data. Set the period parameter to max so we get information for the maximum amount of time.

Reset the index using the reset_index(inplace=True) function on the gme_data DataFrame and display the first five rows of the gme_data dataframe using the head function. Take a screenshot of the results and code from the beginning of Question 3 to the results below.

```
In [17]: gme_data = GameStop.history(period="max")
```

Reset the index using the reset_index(inplace=True) function on the gme_data DataFrame and display the first five rows of the gme_data dataframe using the head function. Take a screenshot of the results and code from the beginning of Question 3 to the results below.

```
In [18]: gme_data.reset_index(inplace=True)
gme_data.head()
```

```
Out[18]:
```

	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	1.683250	11021600	0.0	0.0
2	2002-02-15	1.683250	1.687458	1.658001	1.674834	8389600	0.0	0.0
3	2002-02-19	1.666417	1.666417	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

If you want to use the read_html function the table is located at index 1

```
In [21]: gme_revenue = pd.DataFrame(columns=['Date', 'Revenue'])

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

    if ('GameStop Quarterly Revenue' in table.find('th').text):
        rows = table.find_all('tr')

        for row in rows:
            col = row.find_all('td')

            if col != []:
                date = col[0].text
                revenue = col[1].text.replace(',','').replace('$','')

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

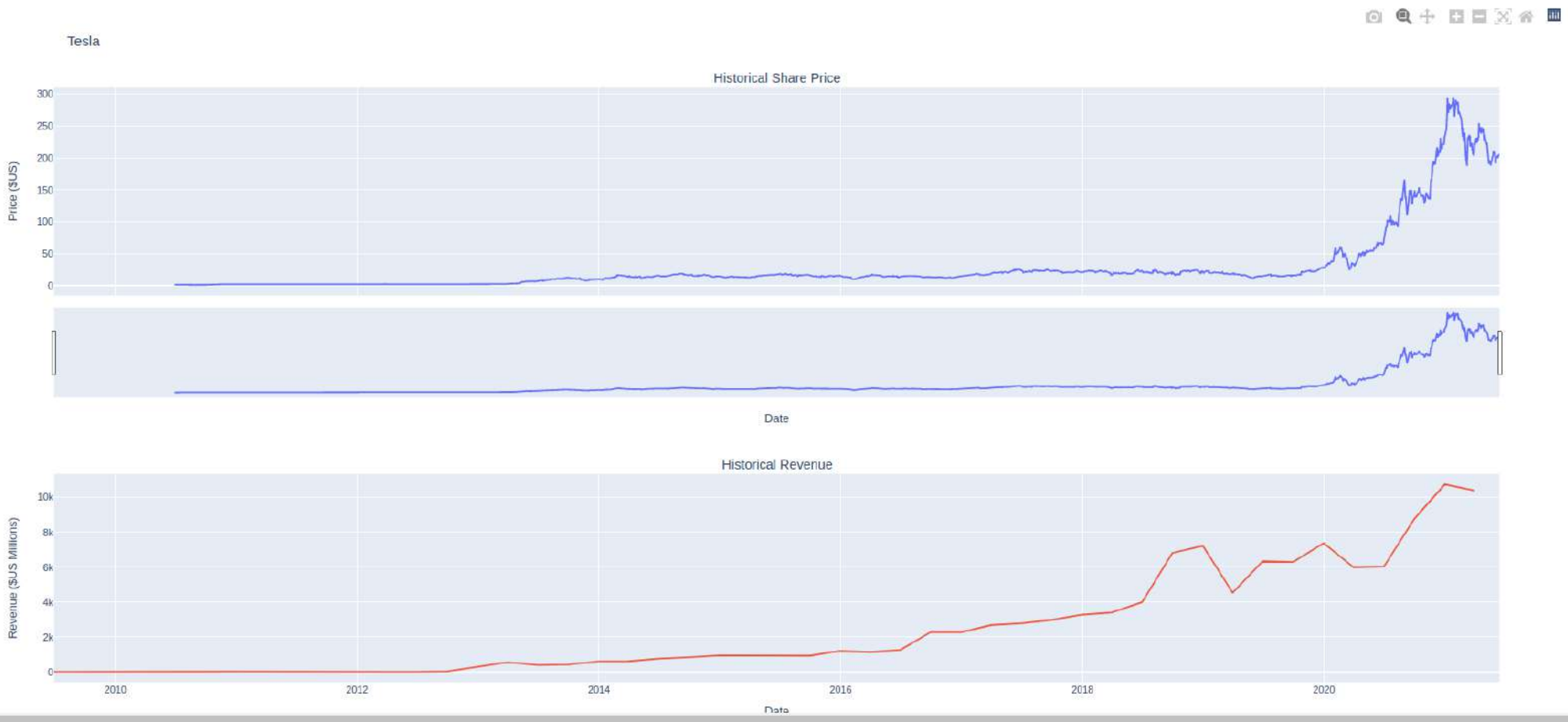
Display the last five rows of the gme_revenue dataframe using the tail function. Take a screenshot of the results.

```
In [22]: gme_revenue.tail()
```

```
Out[22]:
```

	Date	Revenue
57	2006-01-31	1667
58	2005-10-31	534
59	2005-07-31	416
60	2005-04-30	475
61	2005-01-31	709


```
In [23]: make_graph(tesla_data, tesla_revenue, 'Tesla')
```



In [24]: `make_graph(gme_data, gme_revenue, 'GameStop')`

