



Analyzing Historical Stock/Revenue Data and Building a Dashboard

KUNAL SACHDEV

Tesla Stock Data From yFinance

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	3.800	5.000	3.508	4.778	93831500	0	0.0
1	2010-06-30	5.158	6.084	4.860	4.786	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



Tesla Stock Graph

Question 5: Plot Tesla Stock Graph

Use the `make_graph` function to graph the Tesla Stock Data, also provide a title for the graph. The structure to call the `make_graph` function is `make_graph(tesla_data, tesla_revenue, "Tesla")`. Note the graph will only show data upto June 2021.

```
In [70]: make_graph(tesla_data, tesla_revenue, "Tesla Stock Data")
```



GameStop Stock Data Using yFinance

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 [14]: gamestock = 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.

```
In [15]: gme_data = gamestock.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 [16]: gme_data.reset_index(inplace = True)
gme_data.head()
```

```
Out[16]:
```

	Date	Open	High	Low	Close	Volume	Dividends	Stock Splits
0	2002-02-13	6.480513	6.773399	6.413183	6.766665	19054000	0.0	0.0
1	2002-02-14	6.850828	6.864294	6.682503	6.733000	2755400	0.0	0.0
2	2002-02-15	6.733002	6.749834	6.632007	6.699337	2097400	0.0	0.0
3	2002-02-19	6.665671	6.665671	6.312189	6.430017	1852600	0.0	0.0
4	2002-02-20	6.463682	6.648839	6.413184	6.648839	1723200	0.0	0.0

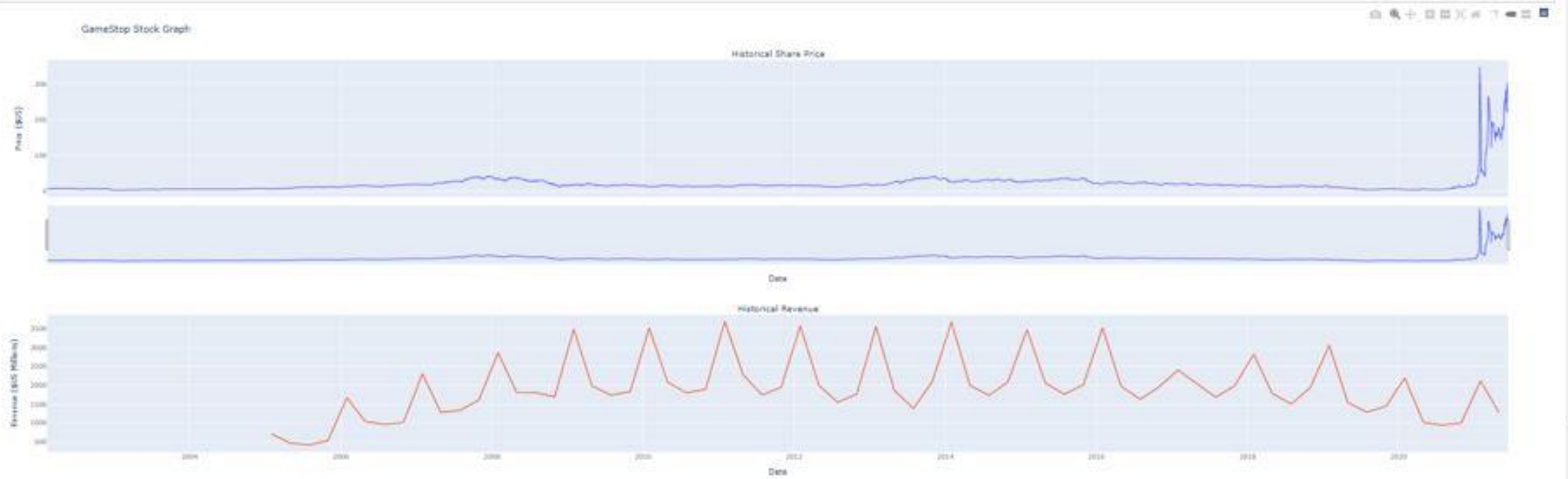


GameStop Stock Graph

Question 6: Plot GameStop Stock Graph

Use the `make_graph` function to graph the GameStop Stock Data, also provide a title for the graph. The structure to call the `make_graph` function is `make_graph(game_data, game_revenue, "GameStop")`. Note the graph will only show data upto June 2021.

```
In [11]: make_graph(game_data, game_revenue, "GameStop Stock Graph")
```





Thank You
