[](https://skills.network/?utm_medium=Exinfluencer&utm_source=Exinfluencer&utm_content=000026UJ&utm_term=10006555&utm_id=NA-SkillsNetwork-Channel-SkillsNetworkCoursesIBMDeveloperSkillsNetworkPY0220ENSkillsNetwork900-2022-01-01)

**Extracting and Visualizing Stock Data**

**Description**

Extracting essential data from a dataset and displaying it is a necessary part of data science; therefore individuals can make correct decisions based on the data. In this assignment, you will extract some stock data, you will then display this data in a graph.

**Table of Contents**

* Define a Function that Makes a Graph
* Question 1: Use yfinance to Extract Stock Data
* Question 2: Use Webscraping to Extract Tesla Revenue Data
* Question 3: Use yfinance to Extract Stock Data
* Question 4: Use Webscraping to Extract GME Revenue Data
* Question 5: Plot Tesla Stock Graph
* Question 6: Plot GameStop Stock Graph

Estimated Time Needed: **30 min**

[2]:



**!**pip install yfinance**==**0.1.67

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

**!**pip install nbformat**==**4.2.0

Collecting yfinance==0.1.67

Downloading yfinance-0.1.67-py2.py3-none-any.whl (25 kB)

Requirement already satisfied: pandas>=0.24 in /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from yfinance==0.1.67) (1.3.5)

Requirement already satisfied: numpy>=1.15 in /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from yfinance==0.1.67) (1.21.6)

Requirement already satisfied: requests>=2.20 in /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from yfinance==0.1.67) (2.29.0)

Collecting multitasking>=0.0.7 (from yfinance==0.1.67)

Downloading multitasking-0.0.11-py3-none-any.whl (8.5 kB)

Requirement already satisfied: lxml>=4.5.1 in /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from yfinance==0.1.67) (4.9.2)

Requirement already satisfied: python-dateutil>=2.7.3 in /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from pandas>=0.24->yfinance==0.1.67) (2.8.2)

Requirement already satisfied: pytz>=2017.3 in /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from pandas>=0.24->yfinance==0.1.67) (2023.3)

Requirement already satisfied: charset-normalizer<4,>=2 in /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from requests>=2.20->yfinance==0.1.67) (3.1.0)

Requirement already satisfied: idna<4,>=2.5 in /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from requests>=2.20->yfinance==0.1.67) (3.4)

Requirement already satisfied: urllib3<1.27,>=1.21.1 in /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from requests>=2.20->yfinance==0.1.67) (1.26.15)

Requirement already satisfied: certifi>=2017.4.17 in /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from requests>=2.20->yfinance==0.1.67) (2023.5.7)

Requirement already satisfied: six>=1.5 in /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from python-dateutil>=2.7.3->pandas>=0.24->yfinance==0.1.67) (1.16.0)

Installing collected packages: multitasking, yfinance

Successfully installed multitasking-0.0.11 yfinance-0.1.67

\_\_ \_\_ \_\_ \_\_

/ \ / \ / \ / \

/ \/ \/ \/ \

███████████████/ /██/ /██/ /██/ /████████████████████████

/ / \ / \ / \ / \ \\_\_\_\_

/ / \\_/ \\_/ \\_/ \ o \\_\_,

/ \_/ \\_\_\_\_\_/ `

|/

███╗ ███╗ █████╗ ███╗ ███╗██████╗ █████╗

████╗ ████║██╔══██╗████╗ ████║██╔══██╗██╔══██╗

██╔████╔██║███████║██╔████╔██║██████╔╝███████║

██║╚██╔╝██║██╔══██║██║╚██╔╝██║██╔══██╗██╔══██║

██║ ╚═╝ ██║██║ ██║██║ ╚═╝ ██║██████╔╝██║ ██║

╚═╝ ╚═╝╚═╝ ╚═╝╚═╝ ╚═╝╚═════╝ ╚═╝ ╚═╝

mamba (1.4.2) supported by @QuantStack

GitHub: <https://github.com/mamba-org/mamba>

Twitter: <https://twitter.com/QuantStack>

█████████████████████████████████████████████████████████████

Looking for: ['bs4==4.10.0']

[+] 0.0s

[+] 0.1s

pkgs/main/linux-64 ━━━━━━━━╸━━━━━━━━━━━━━━━━ 0.0 B / ??.?MB @ ??.?MB/s 0.1s

pkgs/main/noarch ━━━━━━━━━━━╸━━━━━━━━━━━━━ 0.0 B / ??.?MB @ ??.?MB/s 0.1s

pkgs/r/linux-64 ━━╸━━━━━━━━━━━━━━━╸━━━━━━ 0.0 B / ??.?MB @ ??.?MB/s 0.1s

pkgs/r/noarch ━━━━━━━━━╸━━━━━━━━━━━━━━━ 0.0 B / ??.?MB @ ??.?MB/s 0.1s[+] 0.2s

pkgs/main/linux-64 ━━━━━━━━━━━╸━━━━━━━━━━━━━ 73.7kB / ??.?MB @ 479.2kB/s 0.2s

pkgs/main/noarch ━━━━━━━━━━━━━━╸━━━━━━━━━━ 57.4kB / ??.?MB @ 373.2kB/s 0.2s

pkgs/r/linux-64 ━━━━╸━━━━━━━━━━━━━━━╸━━━━ 57.4kB / ??.?MB @ 373.4kB/s 0.2s

pkgs/r/noarch ━━━━━━━━━━━━╸━━━━━━━━━━━━ 41.0kB / ??.?MB @ 266.9kB/s 0.2s[+] 0.3s

pkgs/main/linux-64 ━━━━━━━━━━━━━╸━━━━━━━━━━━ 598.0kB / ??.?MB @ 2.3MB/s 0.3s

pkgs/main/noarch ━━━━━━━━━━━━━━━━━━━━━━━━━ 614.4kB / ??.?MB @ 2.4MB/s Downloaded 0.3s

pkgs/r/linux-64 ━━━━━━━╸━━━━━━━━━━━━━━━╸━ 548.9kB / ??.?MB @ 2.2MB/s 0.3s

pkgs/r/noarch ━━━━━━━━━━━━━━╸━━━━━━━━━━ 593.9kB / ??.?MB @ 2.3MB/s 0.3spkgs/main/noarch 850.6kB @ 2.9MB/s 0.3s

[+] 0.4s

pkgs/main/linux-64 ━━━━━━━━━━━━━━╸━━━━━━━━━━ 1.1MB / ??.?MB @ 3.2MB/s 0.4s

pkgs/r/linux-64 ━━━━━━━━╸━━━━━━━━━━━━━━━━ 999.4kB / ??.?MB @ 2.9MB/s 0.4spkgs/r/noarch 1.3MB @ 3.4MB/s 0.4s

pkgs/r/linux-64 1.4MB @ 3.2MB/s 0.5s

[+] 0.5s

pkgs/main/linux-64 ━╸━━━━━━━━━━━━━━━╸━━━━━━━ 2.0MB / ??.?MB @ 4.0MB/s 0.5s[+] 0.6s

pkgs/main/linux-64 ━━━╸━━━━━━━━━━━━━━━╸━━━━━ 2.6MB / ??.?MB @ 4.4MB/s 0.6s[+] 0.7s

pkgs/main/linux-64 ━━━━━╸━━━━━━━━━━━━━━━╸━━━ 3.1MB / ??.?MB @ 4.5MB/s 0.7s[+] 0.8s

pkgs/main/linux-64 ━━━━━━━╸━━━━━━━━━━━━━━━╸━ 3.4MB / ??.?MB @ 4.5MB/s 0.8s[+] 0.9s

pkgs/main/linux-64 ━━━━━━━━━╸━━━━━━━━━━━━━━━ 4.0MB / ??.?MB @ 4.7MB/s 0.9s[+] 1.0s

pkgs/main/linux-64 ━━━━━━━━━━━╸━━━━━━━━━━━━━ 4.6MB / ??.?MB @ 4.8MB/s 1.0s[+] 1.1s

pkgs/main/linux-64 ━━━━━━━━━━━━━━╸━━━━━━━━━━ 5.1MB / ??.?MB @ 4.8MB/s 1.1s[+] 1.2s

pkgs/main/linux-64 ━━━━━━━━━━━━━━╸━━━━━━━━━━ 5.1MB / ??.?MB @ 4.8MB/s 1.2s[+] 1.3s

pkgs/main/linux-64 ━━━━━━━━╸━━━━━━━━━━━━━━━━ 5.5MB / ??.?MB @ 4.3MB/s 1.3s[+] 1.4s

pkgs/main/linux-64 ━━━━━━━━━╸━━━━━━━━━━━━━━━ 5.8MB / ??.?MB @ 4.3MB/s 1.4s[+] 1.5s

pkgs/main/linux-64 ━━━━━━━━━━━╸━━━━━━━━━━━━━ 5.9MB / ??.?MB @ 4.1MB/s 1.5spkgs/main/linux-64 5.9MB @ 4.1MB/s 1.6s

[+] 1.6s

Pinned packages:

- python 3.7.\*

Transaction

Prefix: /home/jupyterlab/conda/envs/python

Updating specs:

- bs4==4.10.0

- ca-certificates

- certifi

- openssl

Package Version Build Channel Size

─────────────────────────────────────────────────────────────────────────────

Install:

─────────────────────────────────────────────────────────────────────────────

+ bs4 4.10.0 hd3eb1b0\_0 pkgs/main/noarch 10kB

Upgrade:

─────────────────────────────────────────────────────────────────────────────

- ca-certificates 2023.5.7 hbcca054\_0 conda-forge

+ ca-certificates 2023.05.30 h06a4308\_0 pkgs/main/linux-64 123kB

- openssl 1.1.1t h0b41bf4\_0 conda-forge

+ openssl 1.1.1u h7f8727e\_0 pkgs/main/linux-64 4MB

Downgrade:

─────────────────────────────────────────────────────────────────────────────

- beautifulsoup4 4.11.1 pyha770c72\_0 conda-forge

+ beautifulsoup4 4.10.0 pyh06a4308\_0 pkgs/main/noarch 87kB

Summary:

Install: 1 packages

Upgrade: 2 packages

Downgrade: 1 packages

Total download: 4MB

─────────────────────────────────────────────────────────────────────────────

[+] 0.0s

Downloading ━━━━━━━━━━━━━━━━━━━━━━━ 0.0 B 0.0s

Extracting ━━━━━━━━━━━━━━━━━━━━━━━ 0 0.0s[+] 0.1s

Downloading (4) ━━━━━━━━━━━━━━━━━━━━━━━ 0.0 B beautifulsoup4 0.0s

Extracting ━━━━━━━━━━━━━━━━━━━━━━━ 0 0.0sbeautifulsoup4 86.6kB @ 619.1kB/s 0.1s

ca-certificates 122.6kB @ 845.4kB/s 0.1s

bs4 10.2kB @ 69.9kB/s 0.2s

[+] 0.2s

Downloading ━━━━━━━━━━━━━━━━━━━━━━━ 4.1MB 0.1s

Extracting (4) ━━━━━━━━━━━╸━━━━━━━━━━━ 0 beautifulsoup4 0.0sopenssl 3.9MB @ 19.9MB/s 0.2s

[+] 0.3s

Downloading ━━━━━━━━━━━━━━━━━━━━━━━ 4.1MB 0.1s

Extracting (4) ━━━━━━━━━━━━╸━━━━━━━━━━ 0 beautifulsoup4 0.1s[+] 0.4s

Downloading ━━━━━━━━━━━━━━━━━━━━━━━ 4.1MB 0.1s

Extracting (4) ━━━━━━━╸━━━━━━━━━━━━━━━ 0 beautifulsoup4 0.2s[+] 0.5s

Downloading ━━━━━━━━━━━━━━━━━━━━━━━ 4.1MB 0.1s

Extracting (4) ━━━━━━━━━╸━━━━━━━━━━━━━ 0 beautifulsoup4 0.3s[+] 0.6s

Downloading ━━━━━━━━━━━━━━━━━━━━━━━ 4.1MB 0.1s

Extracting (4) ━━━━━━━━━━╸━━━━━━━━━━━━ 0 bs4 0.4s[+] 0.7s

Downloading ━━━━━━━━━━━━━━━━━━━━━━━ 4.1MB 0.1s

Extracting (4) ━━━━━━━━━━━╸━━━━━━━━━━━ 0 bs4 0.5s[+] 0.8s

Downloading ━━━━━━━━━━━━━━━━━━━━━━━ 4.1MB 0.1s

Extracting (4) ━━━━━━━━━━━━╸━━━━━━━━━━ 0 bs4 0.6s[+] 0.9s

Downloading ━━━━━━━━━━━━━━━━━━━━━━━ 4.1MB 0.1s

Extracting (4) ━━━━━━━━━━━━━╸━━━━━━━━━ 0 bs4 0.7s[+] 1.0s

Downloading ━━━━━━━━━━━━━━━━━━━━━━━ 4.1MB 0.1s

Extracting (4) ━━━━━━━━━━━━━━╸━━━━━━━━ 0 ca-certificates 0.8s[+] 1.1s

Downloading ━━━━━━━━━━━━━━━━━━━━━━━ 4.1MB 0.1s

Extracting (4) ━━━━━━━━━━━━━━━╸━━━━━━━ 0 ca-certificates 0.9s[+] 1.2s

Downloading ━━━━━━━━━━━━━━━━━━━━━━━ 4.1MB 0.1s

Extracting (4) ╸━━━━━━━━━━━━━━━╸━━━━━━ 0 ca-certificates 1.0s[+] 1.3s

Downloading ━━━━━━━━━━━━━━━━━━━━━━━ 4.1MB 0.1s

Extracting (4) ━╸━━━━━━━━━━━━━━━╸━━━━━ 0 ca-certificates 1.1s[+] 1.4s

Downloading ━━━━━━━━━━━━━━━━━━━━━━━ 4.1MB 0.1s

Extracting (4) ━━╸━━━━━━━━━━━━━━━╸━━━━ 0 openssl 1.2s[+] 1.5s

Downloading ━━━━━━━━━━━━━━━━━━━━━━━ 4.1MB 0.1s

Extracting (4) ━━━━╸━━━━━━━━━━━━━━━╸━━ 0 openssl 1.3s[+] 1.6s

Downloading ━━━━━━━━━━━━━━━━━━━━━━━ 4.1MB 0.1s

Extracting (4) ━━━━━╸━━━━━━━━━━━━━━━╸━ 0 openssl 1.4s[+] 1.7s

Downloading ━━━━━━━━━━━━━━━━━━━━━━━ 4.1MB 0.1s

Extracting (4) ━━━━━━╸━━━━━━━━━━━━━━━━ 0 openssl 1.5s[+] 1.8s

Downloading ━━━━━━━━━━━━━━━━━━━━━━━ 4.1MB 0.1s

Extracting (4) ━━━━━━━╸━━━━━━━━━━━━━━━ 0 beautifulsoup4 1.6s[+] 1.9s

Downloading ━━━━━━━━━━━━━━━━━━━━━━━ 4.1MB 0.1s

Extracting (3) ━━━━╸━━━━━━━━━━━━━━━━━━ 1 beautifulsoup4 1.7s[+] 2.0s

Downloading ━━━━━━━━━━━━━━━━━━━━━━━ 4.1MB 0.1s

Extracting (3) ━━━━╸━━━━━━━━━━━━━━━━━━ 1 beautifulsoup4 1.8s[+] 2.1s

Downloading ━━━━━━━━━━━━━━━━━━━━━━━ 4.1MB 0.1s

Extracting (2) ━━━━━━━━━━╸━━━━━━━━━━━━ 2 ca-certificates 1.9s[+] 2.2s

Downloading ━━━━━━━━━━━━━━━━━━━━━━━ 4.1MB 0.1s

Extracting ━━━━━━━━━━━━━━━━━━━━━━━ 4 2.0s

Downloading and Extracting Packages

Preparing transaction: done

Verifying transaction: done

Executing transaction: done

Collecting nbformat==4.2.0

Downloading nbformat-4.2.0-py2.py3-none-any.whl (153 kB)

━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 153.3/153.3 kB 23.9 MB/s eta 0:00:00

Requirement already satisfied: ipython-genutils in /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from nbformat==4.2.0) (0.2.0)

Requirement already satisfied: jsonschema!=2.5.0,>=2.4 in /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from nbformat==4.2.0) (4.17.3)

Requirement already satisfied: jupyter-core in /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from nbformat==4.2.0) (4.12.0)

Requirement already satisfied: traitlets>=4.1 in /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from nbformat==4.2.0) (5.9.0)

Requirement already satisfied: attrs>=17.4.0 in /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from jsonschema!=2.5.0,>=2.4->nbformat==4.2.0) (23.1.0)

Requirement already satisfied: importlib-metadata in /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from jsonschema!=2.5.0,>=2.4->nbformat==4.2.0) (4.11.4)

Requirement already satisfied: importlib-resources>=1.4.0 in /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from jsonschema!=2.5.0,>=2.4->nbformat==4.2.0) (5.12.0)

Requirement already satisfied: pkgutil-resolve-name>=1.3.10 in /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from jsonschema!=2.5.0,>=2.4->nbformat==4.2.0) (1.3.10)

Requirement already satisfied: pyrsistent!=0.17.0,!=0.17.1,!=0.17.2,>=0.14.0 in /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from jsonschema!=2.5.0,>=2.4->nbformat==4.2.0) (0.19.3)

Requirement already satisfied: typing-extensions in /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from jsonschema!=2.5.0,>=2.4->nbformat==4.2.0) (4.5.0)

Requirement already satisfied: zipp>=3.1.0 in /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages (from importlib-resources>=1.4.0->jsonschema!=2.5.0,>=2.4->nbformat==4.2.0) (3.15.0)

Installing collected packages: nbformat

Attempting uninstall: nbformat

Found existing installation: nbformat 5.8.0

Uninstalling nbformat-5.8.0:

Successfully uninstalled nbformat-5.8.0

ERROR: pip's dependency resolver does not currently take into account all the packages that are installed. This behaviour is the source of the following dependency conflicts.

jupyter-server 1.24.0 requires nbformat>=5.2.0, but you have nbformat 4.2.0 which is incompatible.

nbclient 0.7.4 requires nbformat>=5.1, but you have nbformat 4.2.0 which is incompatible.

nbconvert 7.4.0 requires nbformat>=5.1, but you have nbformat 4.2.0 which is incompatible.

Successfully installed nbformat-4.2.0

[3]:



**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 Graphing Function**

In this section, we define the function make\_graph. You don't have to know how the function works, you should only care about the inputs. It takes a dataframe with stock data (dataframe must contain Date and Close columns), a dataframe with revenue data (dataframe must contain Date and Revenue columns), and the name of the stock.

[4]:



**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\_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.

[7]:



*# Create a ticker object for Tesla*

ticker **=** 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.

[8]:



*# Extract stock information for the maximum amount of time*

tesla\_data **=** ticker.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.

[9]:



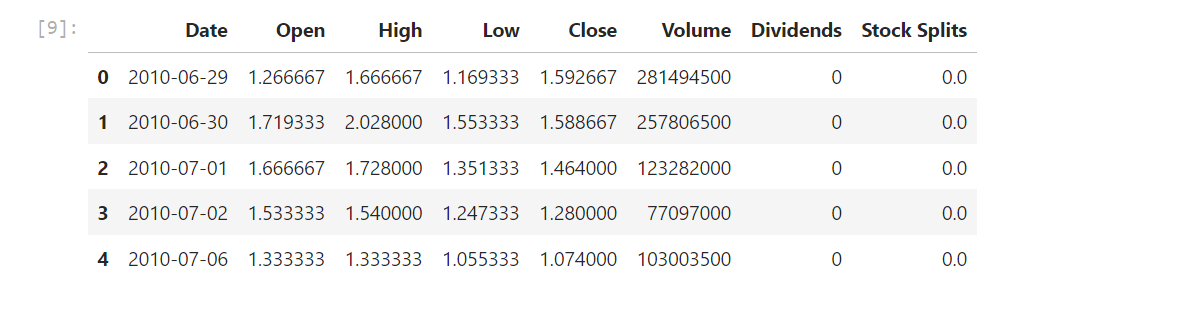
*# Reset the index of the DataFrame*

tesla\_data.reset\_index(inplace**=True**)

​

*# Display the first five rows of the DataFrame*

tesla\_data.head()



**Question 2: Use Webscraping to Extract Tesla Revenue Data**

Use the requests library to download the webpage <https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDeveloperSkillsNetwork-PY0220EN-SkillsNetwork/labs/project/revenue.htm> Save the text of the response as a variable named html\_data.

[18]:



html\_data **=** requests.get("https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDeveloperSkillsNetwork-PY0220EN-SkillsNetwork/labs/project/revenue.htm").text

Parse the html data using beautiful\_soup.

[19]:



soup **=** BeautifulSoup(html\_data)

Using BeautifulSoup or the read\_html function extract the table with Tesla Quarterly Revenue and store it into a dataframe named tesla\_revenue. The dataframe should have columns Date and Revenue.

Click here if you need help locating the table

Below is the code to isolate the table, you will now need to loop through the rows and columns like in the previous lab

soup.find\_all("tbody")[1]

If you want to use the read\_html function the table is located at index 1

[27]:



*# Find the table with Tesla Quarterly Revenue*

table **=** soup.find\_all('table')[1]

​

*# Use the read\_html function to extract the table into a DataFrame*

tesla\_revenue **=** pd.read\_html(str(table))[0]

​

*# Set the column names*

tesla\_revenue.columns **=** ['Date', 'Revenue']

​

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

[28]:



tesla\_revenue['Revenue'] **=** tesla\_revenue['Revenue'].replace({',': '', '\$': ''}, regex**=True**)

​

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

​

[29]:



tesla\_revenue.dropna(inplace**=True**)

​

tesla\_revenue **=** tesla\_revenue[tesla\_revenue['Revenue'] **!=** ""]

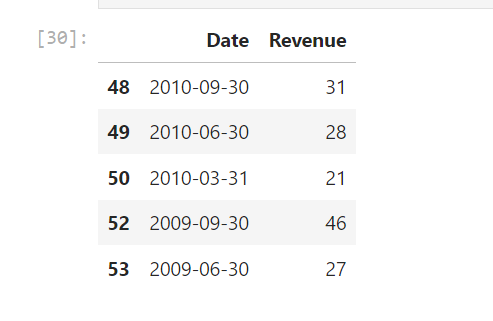
Display the last 5 row of the tesla\_revenue dataframe using the tail function. Take a screenshot of the results.

[30]:



tesla\_revenue.tail(5)

[30]:

****

**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.

[31]:



*# Create a ticker object for Tesla*

ticker **=** 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.

[32]:



*# Extract stock information for the maximum amount of time*

gme\_data **=** ticker.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.

[33]:



*# Reset the index of the DataFrame*

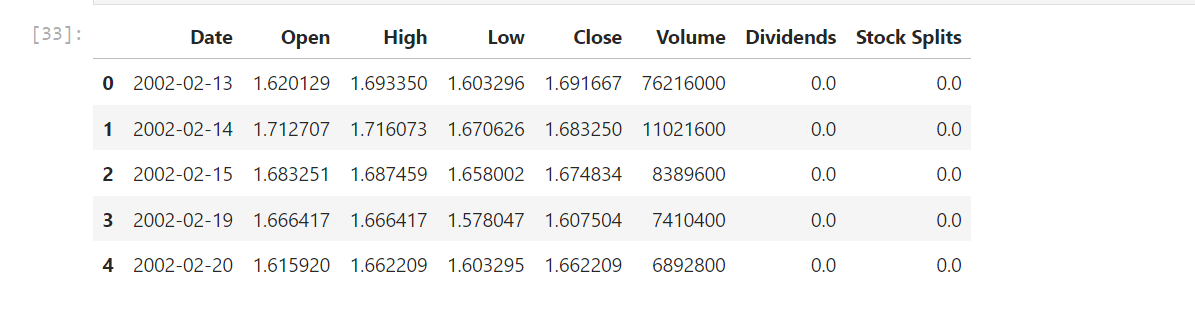
gme\_data.reset\_index(inplace**=True**)

​

*# Display the first five rows of the DataFrame*

gme\_data.head()

[33]:

****

**Question 4: Use Webscraping to Extract GME Revenue Data**

Use the requests library to download the webpage <https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDeveloperSkillsNetwork-PY0220EN-SkillsNetwork/labs/project/stock.html>. Save the text of the response as a variable named html\_data.

[34]:



html\_data **=** requests.get("https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDeveloperSkillsNetwork-PY0220EN-SkillsNetwork/labs/project/stock.html").text

​

Parse the html data using beautiful\_soup.

[35]:



soup **=** BeautifulSoup(html\_data)

Using BeautifulSoup or the read\_html function extract the table with GameStop Quarterly Revenue and store it into a dataframe named gme\_revenue. The dataframe should have columns Date and Revenue. Make sure the comma and dollar sign is removed from the Revenue column using a method similar to what you did in Question 2.

Click here if you need help locating the table

Below is the code to isolate the table, you will now need to loop through the rows and columns like in the previous lab

soup.find\_all("tbody")[1]

If you want to use the read\_html function the table is located at index 1

[57]:



*# Find the table with Tesla Quarterly Revenue*

table **=** soup.find\_all('table')[1]

​

*# Use the read\_html function to extract the table into a DataFrame*

gme\_revenue **=** pd.read\_html(str(table))[0]

​

*# Set the column names*

gme\_revenue.columns **=** ['Date', 'Revenue']

​

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

gme\_revenue['Revenue'] **=** gme\_revenue['Revenue'].replace({',': '', '\$': ''}, regex**=True**)

​

Using BeautifulSoup or the read\_html function extract the table with GameStop Quarterly Revenue and store it into a dataframe named gme\_revenue. The dataframe should have columns Date and Revenue. Make sure the comma and dollar sign is removed from the Revenue column using a method similar to what you did in Question 2.

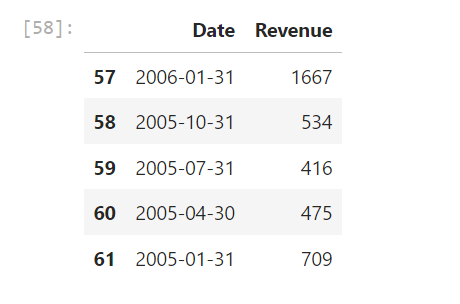
Display the last five rows of the gme\_revenue dataframe using the tail function. Take a screenshot of the results.

[58]:



gme\_revenue.tail(5)

[58]:

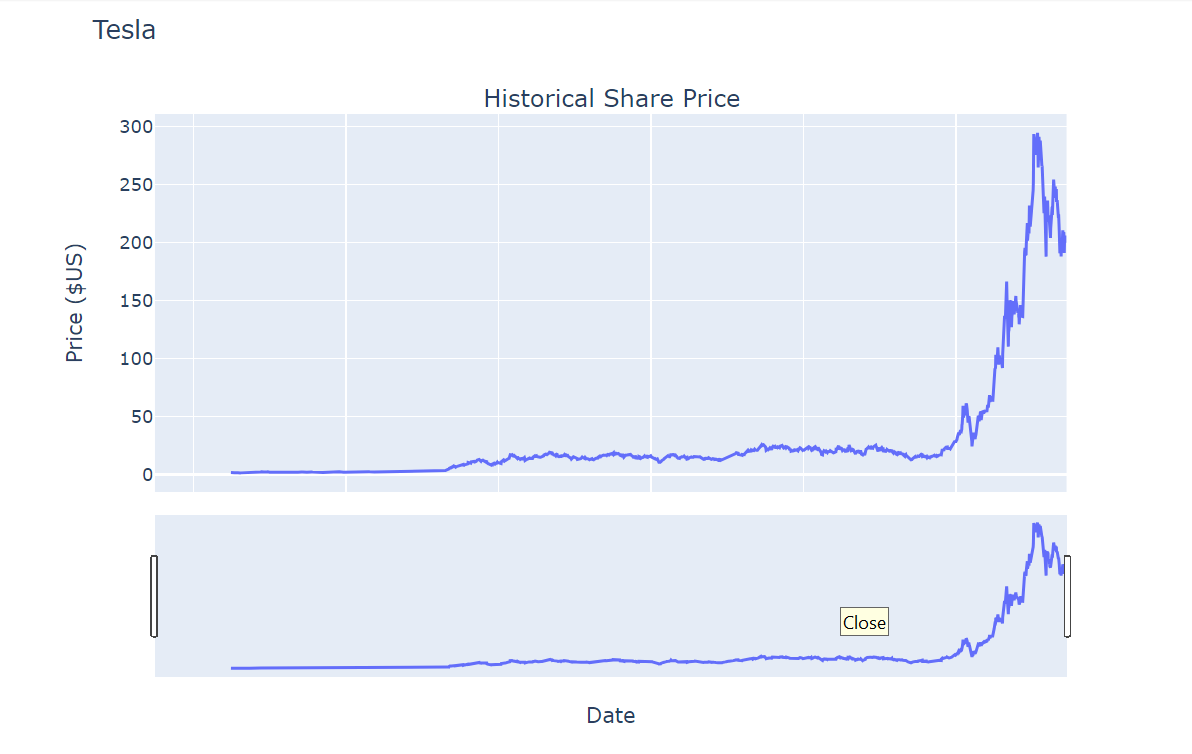


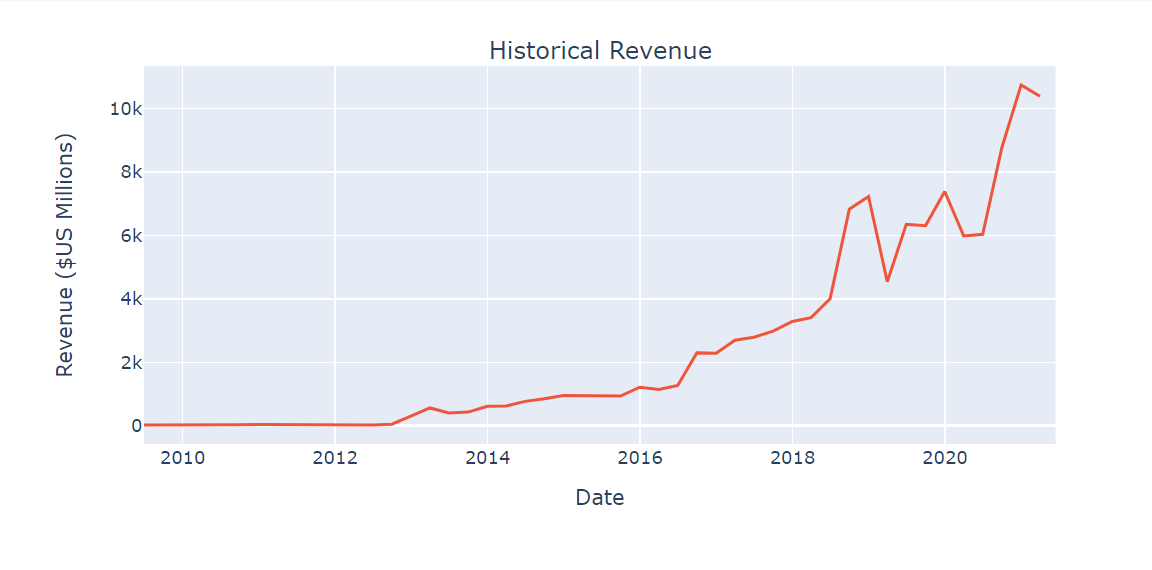
**Question 5: Plot Tesla Stock Graph**

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.

[59]:

make\_graph(tesla\_data, tesla\_revenue, 'Tesla')



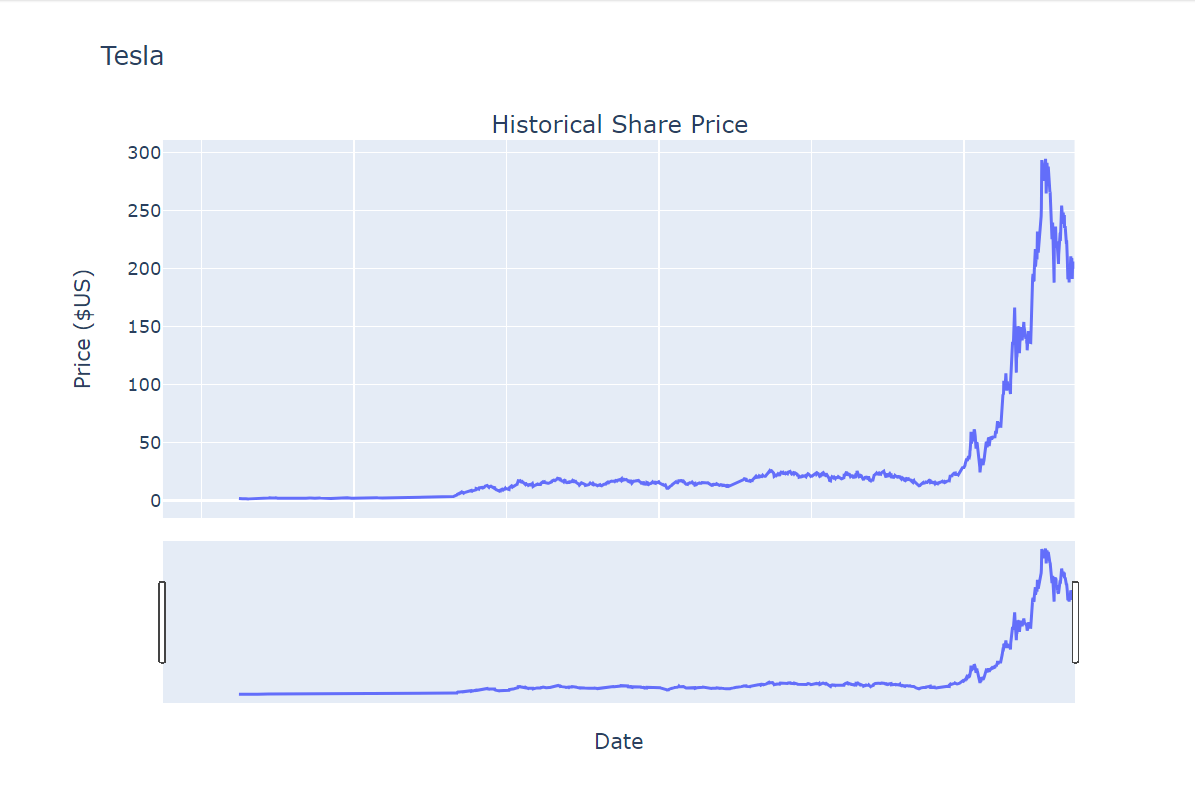


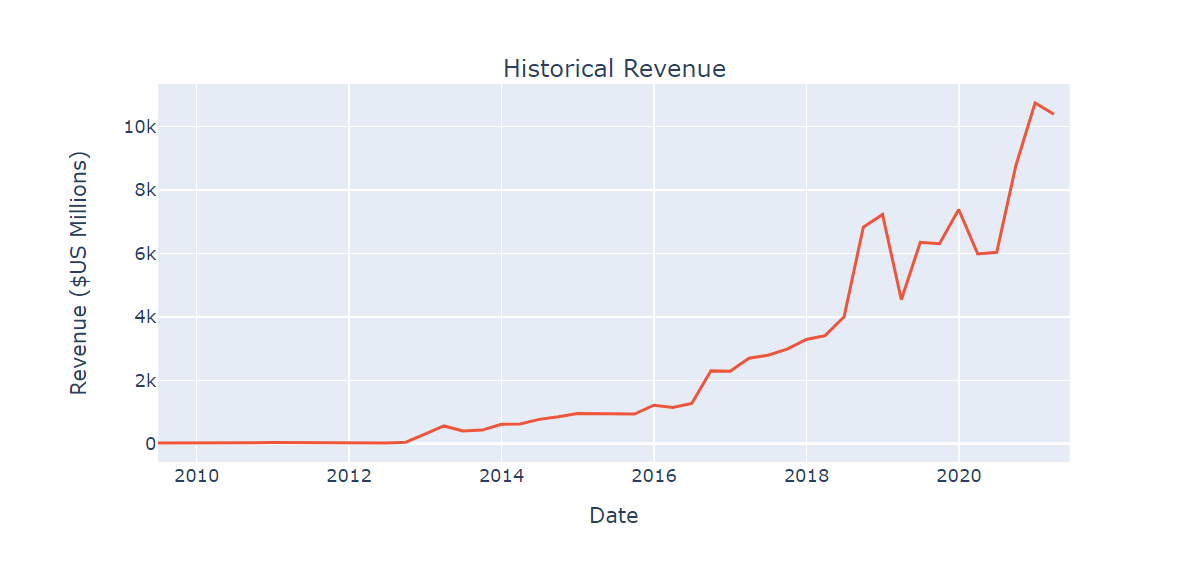
**Question 6: Plot GameStop Stock Graph**

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(gme\_data, gme\_revenue, 'GameStop'). Note the graph will only show data upto June 2021.

[60]:

make\_graph(gme\_data, gme\_revenue, 'GameStop')





**About the Authors:**

[Joseph Santarcangelo](https://www.linkedin.com/in/joseph-s-50398b136/?utm_medium=Exinfluencer&utm_source=Exinfluencer&utm_content=000026UJ&utm_term=10006555&utm_id=NA-SkillsNetwork-Channel-SkillsNetworkCoursesIBMDeveloperSkillsNetworkPY0220ENSkillsNetwork900-2022-01-01) has a PhD in Electrical Engineering, his research focused on using machine learning, signal processing, and computer vision to determine how videos impact human cognition. Joseph has been working for IBM since he completed his PhD.

Azim Hirjani



**## Change Log**

​

| Date (YYYY-MM-DD) | Version | Changed By   | Change Description       |

| ----------------- | ------- | ------------- | ------------------------- |

| 2022-02-28       | 1.2     | Lakshmi Holla | Changed the URL of GameStop |

| 2020-11-10       | 1.1     | Malika Singla | Deleted the Optional part |

| 2020-08-27       | 1.0     | Malika Singla | Added lab to GitLab       |

​

<hr>

​

**##** <h3 align="center"> © IBM Corporation 2020. All rights reserved. <h3/>

​

<p>

​