In [2]:

```
!pip install yfinance
!pip install bs4
!pip install nbformat
```

```
Requirement already satisfied: yfinance in /opt/conda/lib/python3.11/site-package
s (0.2.43)
Requirement already satisfied: pandas>=1.3.0 in /opt/conda/lib/python3.11/site-pa
ckages (from yfinance) (2.2.3)
Requirement already satisfied: numpy>=1.16.5 in /opt/conda/lib/python3.11/site-pa
ckages (from yfinance) (2.1.1)
Requirement already satisfied: requests>=2.31 in /opt/conda/lib/python3.11/site-p
ackages (from yfinance) (2.31.0)
Requirement already satisfied: multitasking>=0.0.7 in /opt/conda/lib/python3.11/s
ite-packages (from yfinance) (0.0.11)
Requirement already satisfied: lxml>=4.9.1 in /opt/conda/lib/python3.11/site-pack
ages (from yfinance) (5.3.0)
Requirement already satisfied: platformdirs>=2.0.0 in /opt/conda/lib/python3.11/s
ite-packages (from yfinance) (4.2.1)
Requirement already satisfied: pytz>=2022.5 in /opt/conda/lib/python3.11/site-pac
kages (from yfinance) (2024.1)
Requirement already satisfied: frozendict>=2.3.4 in /opt/conda/lib/python3.11/sit
e-packages (from yfinance) (2.4.4)
Requirement already satisfied: peewee>=3.16.2 in /opt/conda/lib/python3.11/site-p
ackages (from yfinance) (3.17.6)
Requirement already satisfied: beautifulsoup4>=4.11.1 in /opt/conda/lib/python3.1
1/site-packages (from yfinance) (4.12.3)
Requirement already satisfied: html5lib>=1.1 in /opt/conda/lib/python3.11/site-pa
ckages (from yfinance) (1.1)
Requirement already satisfied: soupsieve>1.2 in /opt/conda/lib/python3.11/site-pa
ckages (from beautifulsoup4>=4.11.1->yfinance) (2.5)
Requirement already satisfied: six>=1.9 in /opt/conda/lib/python3.11/site-package
s (from html5lib>=1.1->yfinance) (1.16.0)
Requirement already satisfied: webencodings in /opt/conda/lib/python3.11/site-pac
kages (from html5lib>=1.1->yfinance) (0.5.1)
Requirement already satisfied: python-dateutil>=2.8.2 in /opt/conda/lib/python3.1
1/site-packages (from pandas>=1.3.0->yfinance) (2.9.0)
Requirement already satisfied: tzdata>=2022.7 in /opt/conda/lib/python3.11/site-p
ackages (from pandas>=1.3.0->yfinance) (2024.2)
Requirement already satisfied: charset-normalizer<4,>=2 in /opt/conda/lib/python
3.11/site-packages (from requests>=2.31->yfinance) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /opt/conda/lib/python3.11/site-pac
kages (from requests>=2.31->yfinance) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in /opt/conda/lib/python3.11/si
te-packages (from requests>=2.31->yfinance) (2.2.1)
Requirement already satisfied: certifi>=2017.4.17 in /opt/conda/lib/python3.11/si
te-packages (from requests>=2.31->yfinance) (2024.6.2)
Requirement already satisfied: bs4 in /opt/conda/lib/python3.11/site-packages (0.
0.2)
Requirement already satisfied: beautifulsoup4 in /opt/conda/lib/python3.11/site-p
ackages (from bs4) (4.12.3)
Requirement already satisfied: soupsieve>1.2 in /opt/conda/lib/python3.11/site-pa
ckages (from beautifulsoup4->bs4) (2.5)
Requirement already satisfied: nbformat in /opt/conda/lib/python3.11/site-package
s (5.10.4)
Requirement already satisfied: fastjsonschema>=2.15 in /opt/conda/lib/python3.11/
site-packages (from nbformat) (2.19.1)
Requirement already satisfied: jsonschema>=2.6 in /opt/conda/lib/python3.11/site-
packages (from nbformat) (4.22.0)
Requirement already satisfied: jupyter-core!=5.0.*,>=4.12 in /opt/conda/lib/pytho
n3.11/site-packages (from nbformat) (5.7.2)
Requirement already satisfied: traitlets>=5.1 in /opt/conda/lib/python3.11/site-p
ackages (from nbformat) (5.14.3)
Requirement already satisfied: attrs>=22.2.0 in /opt/conda/lib/python3.11/site-pa
ckages (from jsonschema>=2.6->nbformat) (23.2.0)
```

```
Requirement already satisfied: jsonschema-specifications>=2023.03.6 in /opt/cond
a/lib/python3.11/site-packages (from jsonschema>=2.6->nbformat) (2023.12.1)
Requirement already satisfied: referencing>=0.28.4 in /opt/conda/lib/python3.11/s
ite-packages (from jsonschema>=2.6->nbformat) (0.35.1)
Requirement already satisfied: rpds-py>=0.7.1 in /opt/conda/lib/python3.11/site-p
ackages (from jsonschema>=2.6->nbformat) (0.18.0)
Requirement already satisfied: platformdirs>=2.5 in /opt/conda/lib/python3.11/sit
e-packages (from jupyter-core!=5.0.*,>=4.12->nbformat) (4.2.1)
```

In [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
```

In [5]:

```
tesla = yf.Ticker('TLSA')
```

In [6]:

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

In [7]:

```
tesla_data.reset_index(inplace=True)
tesla_data.head()
```

Out[7]:

	Date	Open	High	Low	Close	Volume	Dividends	Stock Splits
0	2018-11-20 00:00:00-05:00	2.100840	2.100840	1.670668	1.670668	87465	0.0	0.0
1	2018-11-21 00:00:00-05:00	1.838735	2.434974	1.838735	2.118848	38984	0.0	0.0
2	2018-11-23 00:00:00-05:00	2.200880	2.200880	1.958784	1.960784	14994	0.0	0.0
3	2018-11-26 00:00:00-05:00	1.940776	1.940776	1.552621	1.676671	24990	0.0	0.0
4	2018-11-27 00:00:00-05:00	1.760704	1.800720	1.700680	1.792717	21991	0.0	0.0

In [8]:

```
import requests
```

In [11]:

```
url = "https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDev
```

In [12]:

```
response = requests.get(url)
```

In [13]:

```
html_data = response.text
```

In [14]:

```
print(html_data[:500])
```

```
<!DOCTYPE html>
<!--[if lt IE 7]>      <html class="no-js lt-ie9 lt-ie8 lt-ie7"> <![endif]-->
<!--[if IE 7]>         <html class="no-js lt-ie9 lt-ie8"> <![endif]-->
<!--[if IE 8]>         <html class="no-js lt-ie9"> <![endif]-->
<!--[if gt IE 8]><!--> <html class="no-js"> <!--<![endif]-->
    <head>
        <meta charset="utf-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
                <link rel="canonical" href="https://www.macrotrends.net/stocks/ch
arts/TSLA/tesla/revenue" />
```

In [15]:

```
pip install beautifulsoup4
```

```
Requirement already satisfied: beautifulsoup4 in /opt/conda/lib/python3.11/site-p
ackages (4.12.3)
Requirement already satisfied: soupsieve>1.2 in /opt/conda/lib/python3.11/site-pa
ckages (from beautifulsoup4) (2.5)
Note: you may need to restart the kernel to use updated packages.
```

In [16]:

```python
import requests
from bs4 import BeautifulSoup
```

In [17]:

```python
url = "https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDev
```

In [18]:

```python
response = requests.get(url)
```

In [19]:

```python
html_data = response.text
```

In [20]:

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

In [21]:

```python
print(soup.prettify()[:500])
```

```
<!DOCTYPE html>
<!--[if lt IE 7]>      <html class="no-js lt-ie9 lt-ie8 lt-ie7"> <![endif]-->
<!--[if IE 7]>         <html class="no-js lt-ie9 lt-ie8"> <![endif]-->
<!--[if IE 8]>         <html class="no-js lt-ie9"> <![endif]-->
<!--[if gt IE 8]><!-->
<html class="no-js">
 <!--<![endif]-->
 <head>
  <meta charset="utf-8"/>
  <meta content="IE=edge,chrome=1" http-equiv="X-UA-Compatible"/>
  <link href="https://www.macrotrends.net/stocks/charts/TSLA/tesla/revenue" rel
="canonical"/>
  <title>
   Te
```

In [22]:

```python
import requests
import pandas as pd
from bs4 import BeautifulSoup
```

In [23]:

```python
url = "https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDev
```

In [24]:

```python
response = requests.get(url)
```

In [25]:

```python
html_data = response.text
```

```
In [26]: soup = BeautifulSoup(html_data, 'html.parser')
```

```
In [27]: table = soup.find('table')
```

```
In [28]: dates = []
         revenues = []
```

```
In [29]: for row in table.find_all('tr')[1:]:  # Skip the header row
             cols = row.find_all('td')
             if len(cols) == 2:  # Make sure there are two columns
                 dates.append(cols[0].text.strip())
                 revenues.append(cols[1].text.strip().replace('$', '').replace(',', '').s
```

```
In [30]: tesla_revenue = pd.DataFrame({
             'Date': dates,
             'Revenue': revenues
         })
```

```
In [31]: tesla_revenue['Revenue'] = pd.to_numeric(tesla_revenue['Revenue'], errors='coerc

         print(tesla_revenue)
```

```
    Date  Revenue
0   2021    53823
1   2020    31536
2   2019    24578
3   2018    21461
4   2017    11759
5   2016     7000
6   2015     4046
7   2014     3198
8   2013     2013
9   2012      413
10  2011      204
11  2010      117
12  2009      112
```

```
In [32]: import yfinance as yf
```

```
In [33]: gme_ticker = yf.Ticker("GME")
```

```
In [34]: print(gme_ticker.info)
```

```
{'address1': '625 Westport Parkway', 'city': 'Grapevine', 'state': 'TX', 'zip':
'76051', 'country': 'United States', 'phone': '817 424 2000', 'website': 'http
s://www.gamestop.com', 'industry': 'Specialty Retail', 'industryKey': 'specialty-
retail', 'industryDisp': 'Specialty Retail', 'sector': 'Consumer Cyclical', 'sect
orKey': 'consumer-cyclical', 'sectorDisp': 'Consumer Cyclical', 'longBusinessSumm
ary': 'GameStop Corp., a specialty retailer, provides games and entertainment pro
ducts through its stores and ecommerce platforms in the United States, Canada, Au
stralia, and Europe. The company sells new and pre-owned gaming platforms; access
ories, such as controllers, gaming headsets, and virtual reality products; new an
d pre-owned gaming software; and in-game digital currency, digital downloadable c
ontent, and full-game downloads. It sells collectibles comprising apparel, toys,
trading cards, gadgets, and other retail products for pop culture and technology
enthusiasts, as well as engages in the digital asset wallet and NFT marketplace a
ctivities. The company operates stores and ecommerce sites under the GameStop, EB
Games, and Micromania brands; and pop culture themed stores that sell collectible
s, apparel, gadgets, electronics, toys, and other retail products under the Zing
Pop Culture brand, as well as offers Game Informer magazine, a print and digital
gaming publication. The company was formerly known as GSC Holdings Corp. GameStop
Corp. was founded in 1996 and is headquartered in Grapevine, Texas.', 'fullTimeEm
ployees': 8000, 'companyOfficers': [{'maxAge': 1, 'name': 'Mr. Ryan  Cohen', 'ag
e': 37, 'title': 'President, CEO & Executive Chairman', 'yearBorn': 1986, 'fiscal
Year': 2023, 'exercisedValue': 0, 'unexercisedValue': 0}, {'maxAge': 1, 'name':
'Mr. Daniel William Moore', 'age': 40, 'title': 'Principal Accounting Officer & P
rincipal Financial Officer', 'yearBorn': 1983, 'fiscalYear': 2023, 'totalPay': 27
7711, 'exercisedValue': 0, 'unexercisedValue': 0}, {'maxAge': 1, 'name': 'Mr. Mar
k Haymond Robinson', 'age': 45, 'title': 'General Counsel & Secretary', 'yearBor
n': 1978, 'fiscalYear': 2023, 'totalPay': 337657, 'exercisedValue': 0, 'unexercis
edValue': 0}], 'auditRisk': 8, 'boardRisk': 6, 'compensationRisk': 7, 'shareHolde
rRightsRisk': 3, 'overallRisk': 5, 'governanceEpochDate': 1726617600, 'compensati
onAsOfEpochDate': 1703980800, 'irWebsite': 'http://phx.corporate-ir.net/phoenix.z
html?c=130125&p=irol-irhome', 'maxAge': 86400, 'priceHint': 2, 'previousClose': 2
2.18, 'open': 22.4, 'dayLow': 21.8802, 'dayHigh': 22.63, 'regularMarketPreviousCl
ose': 22.18, 'regularMarketOpen': 22.4, 'regularMarketDayLow': 21.8802, 'regularM
arketDayHigh': 22.63, 'exDividendDate': 1552521600, 'fiveYearAvgDividendYield':
9.52, 'beta': -0.146, 'trailingPE': 159.17857, 'forwardPE': -2228.5, 'volume': 31
90725, 'regularMarketVolume': 3190725, 'averageVolume': 11900811, 'averageVolume1
0days': 14366240, 'averageDailyVolume10Day': 14366240, 'bid': 22.27, 'ask': 22.2
8, 'bidSize': 1800, 'askSize': 1300, 'marketCap': 9950475264, 'fiftyTwoWeekLow':
9.95, 'fiftyTwoWeekHigh': 64.83, 'priceToSalesTrailing12Months': 2.1859567, 'fift
yDayAverage': 22.3738, 'twoHundredDayAverage': 18.91745, 'currency': 'USD', 'ente
rpriseValue': 5789282816, 'profitMargins': 0.00934, 'floatShares': 388490527, 'sh
aresOutstanding': 446510016, 'sharesShort': 36551732, 'sharesShortPriorMonth': 35
971941, 'sharesShortPreviousMonthDate': 1723680000, 'dateShortInterest': 17261856
00, 'sharesPercentSharesOut': 0.0857, 'heldPercentInsiders': 0.085, 'heldPercentI
nstitutions': 0.22132999, 'shortRatio': 4.16, 'shortPercentOfFloat': 0.1037, 'imp
liedSharesOutstanding': 446510016, 'bookValue': 10.278, 'priceToBook': 2.1682234,
'lastFiscalYearEnd': 1706918400, 'nextFiscalYearEnd': 1738540800, 'mostRecentQuar
ter': 1722643200, 'netIncomeToCommon': 42500000, 'trailingEps': 0.14, 'forwardEp
s': -0.01, 'pegRatio': 15.13, 'lastSplitFactor': '4:1', 'lastSplitDate': 16584480
00, 'enterpriseToRevenue': 1.272, 'enterpriseToEbitda': 122.654, '52WeekChange':
0.3171022, 'SandP52WeekChange': 0.33085096, 'lastDividendValue': 0.095, 'lastDivi
dendDate': 1552521600, 'exchange': 'NYQ', 'quoteType': 'EQUITY', 'symbol': 'GME',
'underlyingSymbol': 'GME', 'shortName': 'GameStop Corporation', 'longName': 'Game
Stop Corp.', 'firstTradeDateEpochUtc': 1013610600, 'timeZoneFullName': 'America/N
ew_York', 'timeZoneShortName': 'EDT', 'uuid': '8ded85bd-8171-3e2e-afa6-c812722851
47', 'messageBoardId': 'finmb_1342560', 'gmtOffSetMilliseconds': -14400000, 'curr
entPrice': 22.285, 'targetHighPrice': 10.0, 'targetLowPrice': 5.75, 'targetMeanPr
ice': 7.88, 'targetMedianPrice': 7.88, 'recommendationMean': 4.5, 'recommendation
Key': 'underperform', 'numberOfAnalystOpinions': 2, 'totalCash': 4204199936, 'tot
alCashPerShare': 9.857, 'ebitda': 47200000, 'totalDebt': 533500000, 'quickRatio':
```

```
5.442, 'currentRatio': 6.233, 'totalRevenue': 4552000000, 'debtToEquity': 12.171,
'revenuePerShare': 13.97, 'returnOnAssets': 0.00043000001, 'returnOnEquity': 0.01
5039999, 'freeCashflow': -93387504, 'operatingCashflow': -33100000, 'revenueGrowt
h': -0.314, 'grossMargins': 0.26237, 'ebitdaMargins': 0.010369999, 'operatingMarg
ins': -0.03558, 'financialCurrency': 'USD', 'trailingPegRatio': None}
```

In [35]:
```python
import yfinance as yf
```

In [36]:
```python
gme_ticker = yf.Ticker("GME")
```

In [37]:
```python
gme_data = gme_ticker.history(period="max")
```

In [38]:
```python
print(gme_data.head())
```

```
                               Open      High       Low     Close    Volume  \
Date
2002-02-13 00:00:00-05:00  1.620129  1.693350  1.603296  1.691667  76216000
2002-02-14 00:00:00-05:00  1.712707  1.716074  1.670626  1.683250  11021600
2002-02-15 00:00:00-05:00  1.683251  1.687459  1.658002  1.674834   8389600
2002-02-19 00:00:00-05:00  1.666418  1.666418  1.578047  1.607504   7410400
2002-02-20 00:00:00-05:00  1.615920  1.662210  1.603296  1.662210   6892800

                           Dividends  Stock Splits
Date
2002-02-13 00:00:00-05:00        0.0           0.0
2002-02-14 00:00:00-05:00        0.0           0.0
2002-02-15 00:00:00-05:00        0.0           0.0
2002-02-19 00:00:00-05:00        0.0           0.0
2002-02-20 00:00:00-05:00        0.0           0.0
```

In [39]:
```python
import yfinance as yf
```

In [40]:
```python
gme_ticker = yf.Ticker("GME")
```

In [41]:
```python
gme_data = gme_ticker.history(period="max")
```

In [42]:
```python
gme_data.reset_index(inplace=True)
```

In [43]:
```python
print(gme_data.head())
```

```
                       Date      Open      High       Low     Close    Volume  \
0 2002-02-13 00:00:00-05:00  1.620128  1.693350  1.603296  1.691667  76216000
1 2002-02-14 00:00:00-05:00  1.712707  1.716074  1.670626  1.683250  11021600
2 2002-02-15 00:00:00-05:00  1.683251  1.687459  1.658002  1.674834   8389600
3 2002-02-19 00:00:00-05:00  1.666418  1.666418  1.578047  1.607504   7410400
4 2002-02-20 00:00:00-05:00  1.615920  1.662209  1.603296  1.662209   6892800

   Dividends  Stock Splits
0        0.0           0.0
1        0.0           0.0
2        0.0           0.0
3        0.0           0.0
4        0.0           0.0
```

In [44]:
```python
import requests
```

In [45]:
```python
url = "https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDev
```

In [46]:
```
response = requests.get(url)
```

In [47]:
```
html_data_2 = response.text
```

In [48]:
```
print(html_data_2[:500])
```

```
<!DOCTYPE html>
<!-- saved from url=(0105)https://web.archive.org/web/20200814131437/https://www.
macrotrends.net/stocks/charts/GME/gamestop/revenue -->
<html class=" js flexbox canvas canvastext webgl no-touch geolocation postmessage
websqldatabase indexeddb hashchange history draganddrop websockets rgba hsla mult
iplebgs backgroundsize borderimage borderradius boxshadow textshadow opacity cssa
nimations csscolumns cssgradients cssreflections csstransforms csstransforms3d cs
stransitions fontface g
```

In [49]:
```
import requests
from bs4 import BeautifulSoup
```

In [50]:
```
url = "https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDev
```

In [51]:
```
response = requests.get(url)
```

In [52]:
```
html_data_2 = response.text
```

In [53]:
```
soup = BeautifulSoup(html_data_2, 'html.parser')
```

In [54]:
```
print(soup.prettify()[:500])
```

```
<!DOCTYPE html>
<!-- saved from url=(0105)https://web.archive.org/web/20200814131437/https://www.
macrotrends.net/stocks/charts/GME/gamestop/revenue -->
<html class="js flexbox canvas canvastext webgl no-touch geolocation postmessage
websqldatabase indexeddb hashchange history draganddrop websockets rgba hsla mult
iplebgs backgroundsize borderimage borderradius boxshadow textshadow opacity cssa
nimations csscolumns cssgradients cssreflections csstransforms csstransforms3d cs
stransitions fontface ge
```

In [55]:
```
import requests
import pandas as pd
from bs4 import BeautifulSoup
```

In [56]:
```
url = "https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDev
```

In [57]:
```
response = requests.get(url)
```

In [58]:
```
html_data_2 = response.text
```

In [59]:
```
soup = BeautifulSoup(html_data_2, 'html.parser')
```

In [60]:
```
table = soup.find('table')
```

In [61]:
```
dates = []
revenues = []
```

In [62]:
```
for row in table.find_all('tr')[1:]:  # Skip the header row
    cols = row.find_all('td')
    if len(cols) == 2:  # Make sure there are two columns
        dates.append(cols[0].text.strip())
        revenues.append(cols[1].text.strip().replace('$', '').replace(',', '').s
```

In [63]:

```
gme_revenue = pd.DataFrame({
    'Date': dates,
    'Revenue': revenues
})
```

In [64]:

```
gme_revenue['Revenue'] = pd.to_numeric(gme_revenue['Revenue'], errors='coerce')
```

In [65]:

```
print(gme_revenue)
```

```
    Date  Revenue
0   2020     6466
1   2019     8285
2   2018     8547
3   2017     7965
4   2016     9364
5   2015     9296
6   2014     9040
7   2013     8887
8   2012     9551
9   2011     9474
10  2010     9078
11  2009     8806
12  2008     7094
13  2007     5319
14  2006     3092
15  2005     1843
```

In [67]:

```
pip install matplotlib
```

```
Collecting matplotlib
  Downloading matplotlib-3.9.2-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x8
6_64.whl.metadata (11 kB)
Collecting contourpy>=1.0.1 (from matplotlib)
  Downloading contourpy-1.3.0-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86
_64.whl.metadata (5.4 kB)
Collecting cycler>=0.10 (from matplotlib)
  Downloading cycler-0.12.1-py3-none-any.whl.metadata (3.8 kB)
Collecting fonttools>=4.22.0 (from matplotlib)
  Downloading fonttools-4.54.1-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x8
6_64.whl.metadata (163 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 163.7/163.7 kB 17.0 MB/s eta 0:00:00
Collecting kiwisolver>=1.3.1 (from matplotlib)
  Downloading kiwisolver-1.4.7-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x8
6_64.whl.metadata (6.3 kB)
Requirement already satisfied: numpy>=1.23 in /opt/conda/lib/python3.11/site-pack
ages (from matplotlib) (2.1.1)
Requirement already satisfied: packaging>=20.0 in /opt/conda/lib/python3.11/site-
packages (from matplotlib) (24.0)
Collecting pillow>=8 (from matplotlib)
  Downloading pillow-10.4.0-cp311-cp311-manylinux_2_28_x86_64.whl.metadata (9.2 k
B)
Collecting pyparsing>=2.3.1 (from matplotlib)
  Downloading pyparsing-3.1.4-py3-none-any.whl.metadata (5.1 kB)
Requirement already satisfied: python-dateutil>=2.7 in /opt/conda/lib/python3.11/
site-packages (from matplotlib) (2.9.0)
Requirement already satisfied: six>=1.5 in /opt/conda/lib/python3.11/site-package
s (from python-dateutil>=2.7->matplotlib) (1.16.0)
Downloading matplotlib-3.9.2-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_
64.whl (8.3 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 8.3/8.3 MB 99.9 MB/s eta 0:00:00:00:0
100:01
Downloading contourpy-1.3.0-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_6
4.whl (323 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 323.2/323.2 kB 37.1 MB/s eta 0:00:00
Downloading cycler-0.12.1-py3-none-any.whl (8.3 kB)
Downloading fonttools-4.54.1-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_
64.whl (4.9 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 4.9/4.9 MB 122.5 MB/s eta 0:00:0000:0
1
Downloading kiwisolver-1.4.7-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_
64.whl (1.4 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1.4/1.4 MB 85.5 MB/s eta 0:00:00
Downloading pillow-10.4.0-cp311-cp311-manylinux_2_28_x86_64.whl (4.5 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 4.5/4.5 MB 107.9 MB/s eta 0:00:0000:0
1
Downloading pyparsing-3.1.4-py3-none-any.whl (104 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 104.1/104.1 kB 12.7 MB/s eta 0:00:00
Installing collected packages: pyparsing, pillow, kiwisolver, fonttools, cycler,
contourpy, matplotlib
Successfully installed contourpy-1.3.0 cycler-0.12.1 fonttools-4.54.1 kiwisolver-
1.4.7 matplotlib-3.9.2 pillow-10.4.0 pyparsing-3.1.4
Note: you may need to restart the kernel to use updated packages.
```

In [68]:

```python
import yfinance as yf
import pandas as pd
import matplotlib.pyplot as plt
```

In [69]:

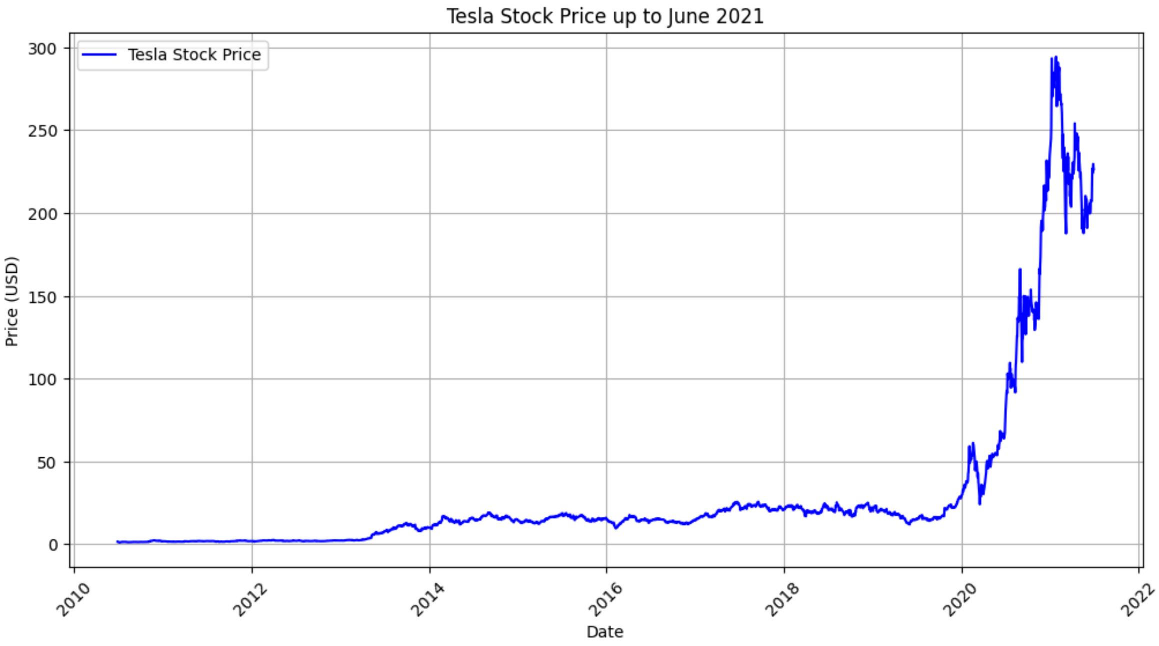
```python
tesla_ticker = yf.Ticker("TSLA")
tesla_data = tesla_ticker.history(period="max")
```

```
tesla_data.reset_index(inplace=True)
```

In [70]:
```
def make_graph(data, title):
    plt.figure(figsize=(12, 6))
    plt.plot(data['Date'], data['Close'], label='Tesla Stock Price', color='blue
    plt.title(title)
    plt.xlabel('Date')
    plt.ylabel('Price (USD)')
    plt.xticks(rotation=45)
    plt.grid()
    plt.legend()
    plt.show()
```

In [71]:
```
tesla_data_filtered = tesla_data[tesla_data['Date'] <= '2021-06-30']
```

In [72]:
```
make_graph(tesla_data_filtered, title="Tesla Stock Price up to June 2021")
```

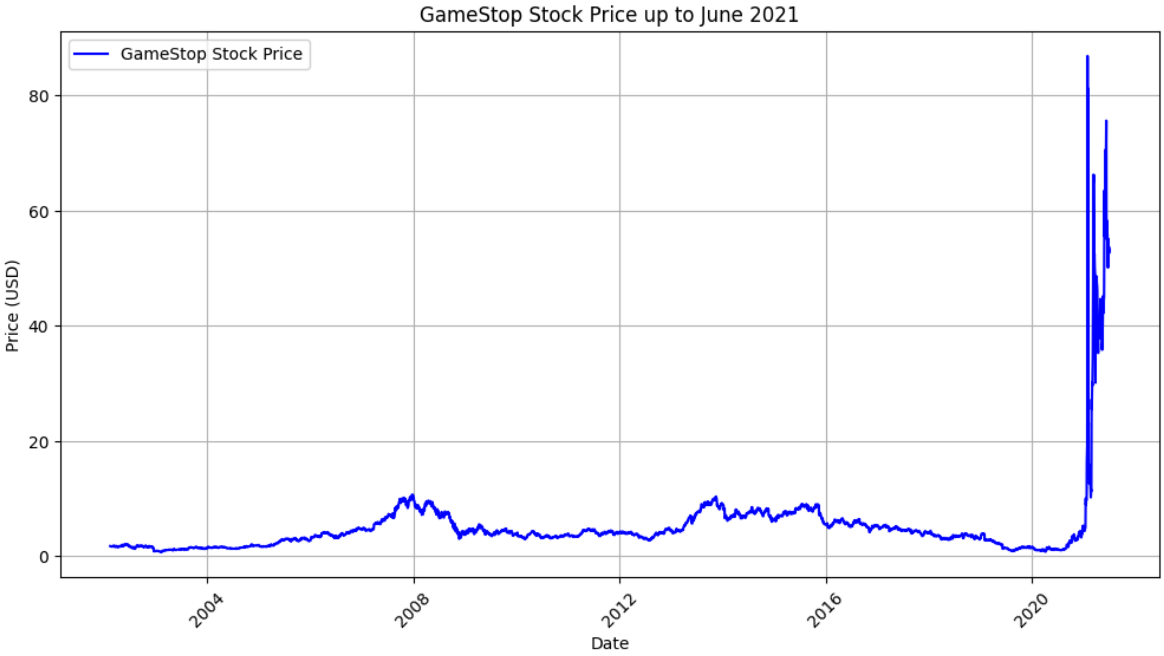


In [78]:
```
gme_ticker = yf.Ticker("GME")
gme_data = gme_ticker.history(period="max")
gme_data.reset_index(inplace=True)
```

In [79]:
```
def make_graph(data, title):
    plt.figure(figsize=(12, 6))
    plt.plot(data['Date'], data['Close'], label='GameStop Stock Price', color='b
    plt.title(title)
    plt.xlabel('Date')
    plt.ylabel('Price (USD)')
    plt.xticks(rotation=45)
    plt.grid()
    plt.legend()
    plt.show()
```

In [80]:
```
gme_data_filtered = gme_data[gme_data['Date'] <= '2021-06-30']
```

In [81]:
```
make_graph(gme_data_filtered, title="GameStop Stock Price up to June 2021")
```


GameStop Stock Price up to June 2021
GameStop Stock Price
Price (USD)
0
20
40
60
80
2004
2008
2012
2016
2020
Date