

**\*\*STOCK MARKET PERFORMANCE ANALYSIS USING PYTHON \*\***

```
!pip install yfinance
```

```
#TO COLLECT REAL_TIME STOCK MARKET DATA USING "yfinance" API:
```

```
import pandas as pd
import yfinance as yf
from datetime import datetime
```

```
start_date=datetime.now() - pd.DateOffset(months=6)
end_date=datetime.now()
```

```
tickers=['AAPL','MSFT','NFLX','GOOG']
```

```
df_list=[]
```

```
for ticker in tickers:
    data=yf.download(ticker,start=start_date,end=end_date)
    df_list.append(data)
```

```
[*****100%*****] 1 of 1 completed
```

```
df=pd.concat(df_list,keys=tickers)
print(df.head())
```

		Open	High	Low	Close	Adj Close \
AAPL	2023-03-17	156.080002	156.740005	154.279999	155.000000	154.577164
	2023-03-20	155.070007	157.820007	154.149994	157.399994	156.970612
	2023-03-21	157.320007	159.399994	156.539993	159.279999	158.845490
	2023-03-22	159.300003	162.139999	157.809998	157.830002	157.399445
	2023-03-23	158.830002	161.550003	157.679993	158.929993	158.496445
Volume						
AAPL	2023-03-17	98944600				
	2023-03-20	73641400				
	2023-03-21	73938300				
	2023-03-22	75701800				
	2023-03-23	67622100				

```
# IN THE ABOVE DATASET, DATECOLUMN IS THE INDEX SO BEFORE MOVING FURTHER WE NEED TO CHANGE IT
```

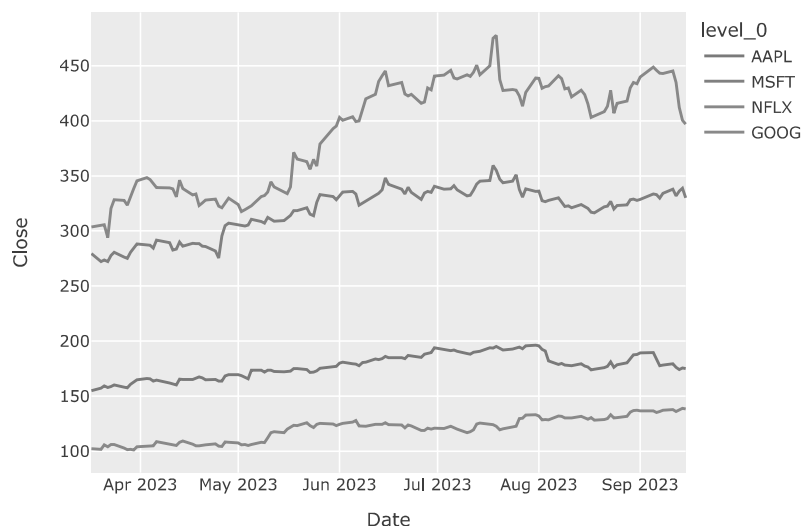
```
df=df.reset_index()
print(df.head())
```

	level_0	Date	Open	High	Low	Close \
0	AAPL	2023-03-17	156.080002	156.740005	154.279999	155.000000
1	AAPL	2023-03-20	155.070007	157.820007	154.149994	157.399994
2	AAPL	2023-03-21	157.320007	159.399994	156.539993	159.279999
3	AAPL	2023-03-22	159.300003	162.139999	157.809998	157.830002
4	AAPL	2023-03-23	158.830002	161.550003	157.679993	158.929993
Adj Close Volume						
0			154.577164	98944600		
1			156.970612	73641400		
2			158.845490	73938300		
3			157.399445	75701800		
4			158.496445	67622100		

```
#now lets look at the performance in the stock market
```

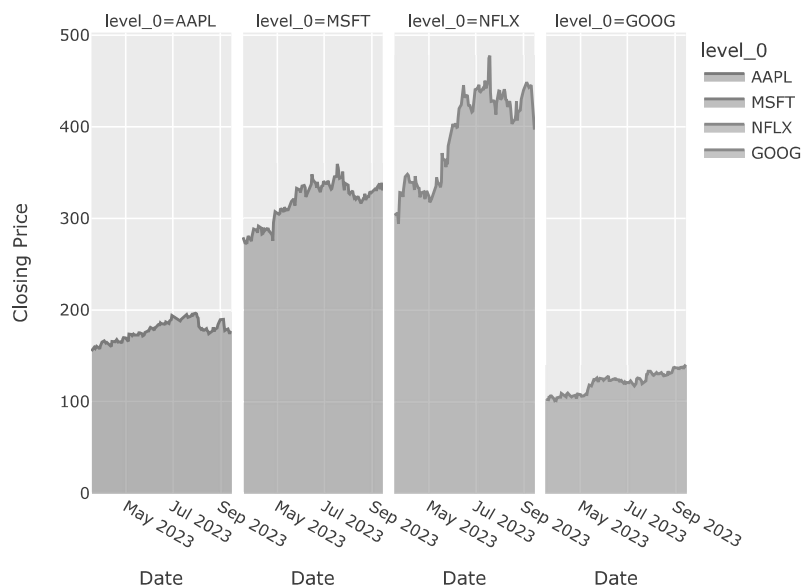
```
import plotly.express as px
fig=px.line(df,x='Date',y='Close',color='level_0',title="Stock Market Performance for the last six months")
fig.show()
```

Stock Market Performance for the last six months



```
fig=px.area(df,x='Date',y='Close',color='level_0',facet_col='level_0',labels={'Date':'Date','Close':'Closing Price','Ticker':'Company'},title
fig.show()
```

Stock Prices for Apple,Microsoft,Netflix, and Google



#NOW LET US ANALYZE THE MOVING AVERAGES TO IDENTIFY TRENDS AND PATTERNS IN EACH COMPANY'S STOCK PRICE MOVEMENTS OVER A PERIOD TIME

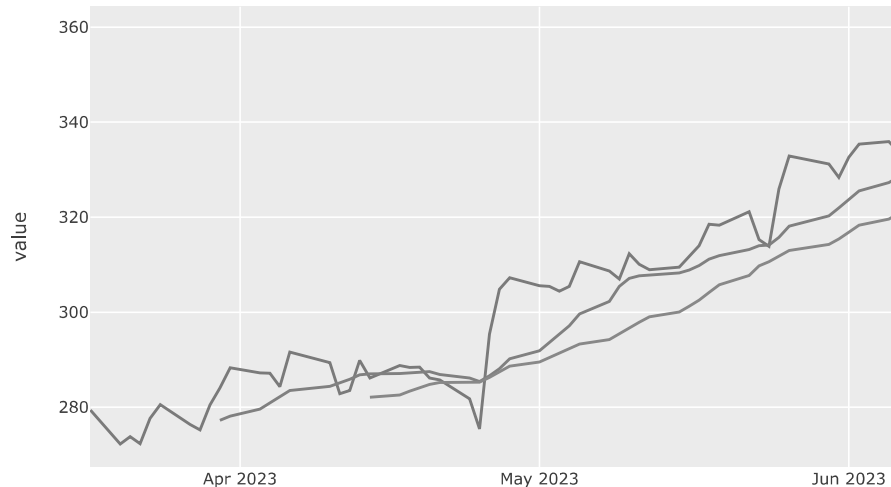
```
df['MA10']=df.groupby('level_0')['Close'].rolling(window=10).mean().reset_index(0,drop=True)
df['MA20']=df.groupby('level_0')['Close'].rolling(window=20).mean().reset_index(0,drop=True)
```

```
for level_0,group in df.groupby('level_0'):
    print(f'Moving Averages for {level_0}')
    print(group[['MA10','MA20']])
```

```
# NOW LET US VISUALIZE MOVING AVERAGES OF THE FOLLOWING COMPANIES
```

```
for level_0,group in df.groupby('level_0'):
    fig=px.line(group,x='Date',y=['Close','MA10','MA20'],title=f"{level_0}Moving Averages")
    fig.show()
```

MSFTMoving Averages



NFLXMoving Averages



#WHEN MA10 CROSSES ABOVE MA20----> (BULLISH SIGNAL INDICATING THAT STOCKS PRICE WILL CONTINUE TO RISE)

#WHEN MA10 CROSSES BELOW MA20----> (BEARISH SIGNAL STATES THAT STOCK PRICE WILL CONTINUE FALLING)

>

#NOW LET US UNDERSTAAND THE VOLATILITY OF ALL COMPANIES

# VOLATILITY---> MEASURE OF HOW MUCH AND HOW OFTEN THE STOCK PRICE OR MARKET FLUCTUATES OVER A GIVEN PERIOD OF TIME

#NOW LET US VISUALISE THE VOLATILITY OF ALL THE COMPANIES

```
df['Volatility']=df.groupby('level_0')['Close'].pct_change().rolling(window=10).std().reset_index(0,drop=True)
```

```
fig=px.line(df,x='Date',y='Volatility',color='level_0',title='Volatility of All Companies')
```

```
fig.show()
```

## Volatility of All Companies

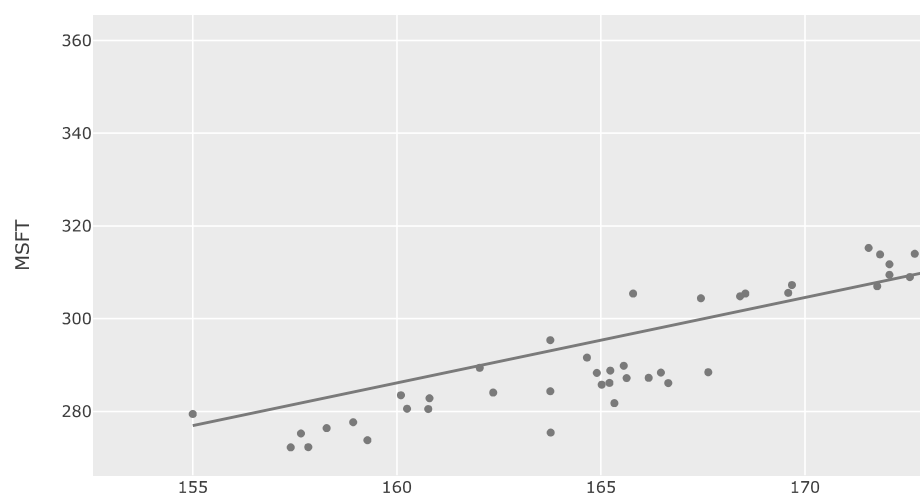
```
#HIGH VOLATILITY INDICATES THAT LARGE AND FREQUENT PRICE MOVEMENTS OF STOCK /MARKET EXPERIENCES
#LOW VOLATILITY INDICATES THAT MARKET EXPERIENCES SMALLER /LESS FREQUENT PRICE  MOVEMENTS
```

```
#NOW LET US ANALYZE THE CORRELATION BETWEEN STOCK PRICES OF APPLE AND MICROSOFT
```

```
#CREATE A DATAFRAME WITH STOCK PRICES OF APPLE AND MICROSOFT
apple=df.loc[df['level_0']=='AAPL',['Date','Close']].rename(columns={'Close': 'AAPL'})
microsoft=df.loc[df['level_0']=='MSFT',['Date','Close']].rename(columns={'Close': 'MSFT'})
df_corr=pd.merge(apple,microsoft,on='Date')
```

```
#CREATE A SCATTER PLOT TO VISUALIZE THE CORRELATION
fig=px.scatter(df_corr,x='AAPL',y='MSFT',trendline='ols',title='Correlation between Apple  and Microsoft')
fig.show()
```

## Correlation between Apple and Microsoft



```
#FROM THE ABOVE PLOT WE CAN SEE THAT THERE IS A STRONG RALATIONSHIP BETWEEN STOCK PRICES OF APPLE AND MICROSOFT(REASON MAY BE DUE TO FACTORS
```

```
#THIS IS HOW WE CAN PERFORM STOCK MARKET ANALYSIS USING PYTHON
```