Q1:Query

select tmphigh.company, tmphigh.HighDate, tmphigh.High, tmplow.LowDate, Round(tmplow.Low, 2) as Low

from ((select top 1'AMZN' as company, max(High) as High, Date as HighDate from amazon group by Date order by max(High) DESC) as tmphigh

join

(select top 1'AMZN' as company, min(Low) as Low, Date as LowDate from amazon group by Date order by min(Low) ASC) as tmplow

on

tmphigh.company = tmplow.company)

union

select tmphigh.company, tmphigh.HighDate, tmphigh.High, tmplow.LowDate, Round(tmplow.Low, 2) as Low

from ((select top 1'FB' as company, max(High) as High, Date as HighDate from fb group by Date order by max(High) DESC) as tmphigh

join

(select top 1'FB' as company, min(Low) as Low, Date as LowDate from fb group by Date order by min(Low) ASC) as tmplow

on

tmphigh.company = tmplow.company)

Table

Description automatically generated

Q2: Query

select 'AMZN' as company, Format(sum(Volume), '###,###,###') as 'Total Volume(2015)' from amazon

where year(Date) = 2015

UNION

select 'FB' as company, Format(sum(Volume), '#,###,###,###') as 'Total Volume(2015)' from fb

where year(Date) = 2015

Output:

Table

Description automatically generated

Q3: Query

select tmp.company, tmp.DOW, tmp.[Volume(Billions)] from

(select Top 1 'AMZN' as company, Datename(WEEKDAY, Date) as DOW, Round(Sum(Volume)/1000000000, 2) as 'Volume(Billions)' from amazon

where year(Date) >=2012 and year(Date)<=2015

group by Datename(WEEKDAY, Date)

order by Round(Sum(Volume)/1000000000, 2) desc) as tmp

UNION

select tmp.company, tmp.DOW, tmp.[Volume(Billions)] from

(select Top 1 'FB' as company, Datename(WEEKDAY, Date) as DOW, Round(Sum(Volume)/1000000000, 2) as 'Volume(Billions)' from fb

where year(Date) >=2012 and year(Date)<=2015

group by Datename(WEEKDAY, Date)

order by Round(Sum(Volume)/1000000000, 2) desc) as tmp

Output:

Table

Description automatically generated

Q4: 1) Introduction:

We have been provided with two datasets on the Facebook and Amazon stocks. The datasets list daily details of the two stocks from August 31, 2012 to June 30, 2017. The variables of interest include the opening price of the stock, the highest value during the day, the lowest value during the day, the closing price of the stock for the day and the daily volume of the stocks traded. The dates exclude weekends and national holidays during which the markets are closed.

For both the datasets, we have two variables describing the SplitRatio and the ExDividend. On a first glance, the value of these variables does not change at all throughout the tables. This means that Facebook and Amazon do not offer dividend to their shareholders and their shares have never been split (why shares are split is a different topic and not relevant to our analysis). Running a few simple queries on the columns like High and AdjHigh or Low and AdjLow tells us that the respective column pairs are not very different. Therefore, we shall use columns like High, Low etc and ignore the adjusted values (since, they are the same). Finally, we observe two key differences between the two stocks. The value of each share is much higher for Amazon than for Facebook. This makes sense since Amazon started in 1994 while facebook started 10 years later in 2004. What is interesting is that the volume of shares traded is way higher for Facebook in comparison to Amazon.

We will now proceed to analyze the two datasets and try to find some insights.

2) Queries, Tables and Graphs

Query 1:

select tmp.company, tmp.DOM, tmp.average\_lowest\_price from

(select 'AMZN' as company, Datepart(MONTH, Date) as DOM, Avg(Low) as average\_lowest\_price from amazon

where year(Date) >=2012 and year(Date)<=2017

group by Datepart(MONTH, Date)) as tmp

UNION

select tmp.company, tmp.DOM, tmp.average\_lowest\_price from

(select 'FB' as company, Datepart(MONTH, Date) as DOM, Avg(Low) as average\_lowest\_price from fb

where year(Date) >=2012 and year(Date)<=2017

group by Datepart(MONTH, Date)) as tmp

Chart 1:

Query 2:

Select 'AMZN' as company, DATEPART(Month, Date) as month, count(Date) as positive\_days, neg.negative\_days

from amazon a, (Select DATEPART(Month, Date) as month, count(Date) as negative\_days from amazon where AdjClose < AdjOpen group by DATEPART(Month, Date)) as neg

where AdjClose > AdjOpen and neg.month = DATEPART(Month, Date)

group by DATEPART(Month, Date), neg.negative\_days

union

Select 'FB' as company, DATEPART(Month, Date) as month, count(Date) as positive\_days, neg.negative\_days

from fb a, (Select DATEPART(Month, Date) as month, count(Date) as negative\_days from fb where AdjClose < AdjOpen group by DATEPART(Month, Date)) as neg

where AdjClose > AdjOpen and neg.month = DATEPART(Month, Date)

group by DATEPART(Month, Date), neg.negative\_days

Chart 2:

**Chart 3:**

3) Conclusion:

Based on my analysis from chart 1, I found that on average both Facebook and Amazon stocks hit the largest low prices in June and the smallest low prices in August. This means that an investor should wait till August if he/she would like to buy either of these two stocks to get them at a low price. The chart also tells us that they should avoid buying in June. Investors who are already invested should avoid selling during August. We can do a similar analysis to determine which months are good to sell these stocks when the average high prices are the highest but due to the 6 page constraint on this report we will not be doing that.

From chart 2 and 3, we can see which month has the maximum number of positive days and which month has the maximum number of negative days. This chart is especially beneficial to day traders.

Day traders can focus on the Amazon stock during the month of May as the number of positive days in June is the largest. This means that the day trader has maximum chances of making a profit during June. They should also try to stay away from it during the month of May since the stock sees the maximum number of negative days in that month. Similarly, they can focus of Facebook stock during the months of June and September and they should avoid it during May.