

## Importing Libraries

```
In [51]: import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
import scipy.stats as st
```

## Uploading Data Set

```
In [22]: df = pd.read_excel("/Users/josedhernandez/Desktop/Stocks_Feb_2020.xlsx")

# Printing the first & last 5 rows of the data set
df
```

Out[22]:

	Name	Purchased Price	Shares Purchased	Date of Purchase	Purchase Total	Selling Price	Shares Sold	Sell Total	Profit Margin	Date of Sale
0	Apple	306.12	50	2020-02-03	15306.00	307.56	50	15378.00	72.00	2020-02-03
1	FaceBook	204.00	100	2020-02-03	20400.00	204.77	100	20477.00	77.00	2020-02-03
2	Amazon	2020.44	10	2020-02-03	20204.40	2033.13	10	20331.30	126.90	2020-02-03
3	Google	1465.00	15	2020-02-03	21975.00	1470.01	15	22050.15	75.15	2020-02-03
4	Microsoft	171.00	100	2020-02-03	17100.00	172.83	100	17283.00	183.00	2020-02-03
...	...	...	...	...	...	...	...	...	...	...
90	Apple	260.64	50	2028-02-28	13032.00	262.77	50	13138.50	106.50	2028-02-28
91	FaceBook	185.60	100	2028-02-28	18560.00	188.10	100	18810.00	250.00	2028-02-28
92	Amazon	1850.21	10	2028-02-28	18502.10	1842.00	10	18420.00	-82.10	2028-02-28
93	Google	1290.19	15	2028-02-28	19352.85	1312.40	15	19686.00	333.15	2028-02-28
94	Microsoft	154.68	100	2028-02-28	15468.00	159.64	100	15964.00	496.00	2028-02-28

95 rows × 10 columns

## Most & Least Profitable Trade

```
In [16]: # Most Profitable

df.loc[df['Profit Margin'].idxmax()]
```

Out[16]:

Name	Amazon
Purchased Price	2158.79
Shares Purchased	10
Date of Purchase	2020-02-18 00:00:00
Purchase Total	21587.9
Selling Price	2262.2
Shares Sold	10
Sell Total	22622
Profit Margin	1034.1
Date of Sale	2020-02-18 00:00:00

Name: 52, dtype: object

```
In [17]: #Least Profitable

df.loc[df['Profit Margin'].idxmin()]
```

Out[17]:

Name	Amazon
Purchased Price	2098.51
Shares Purchased	10
Date of Purchase	2020-02-10 00:00:00
Purchase Total	20985.1
Selling Price	2041.55
Shares Sold	10
Sell Total	20415.5
Profit Margin	-569.6
Date of Sale	2020-02-10 00:00:00

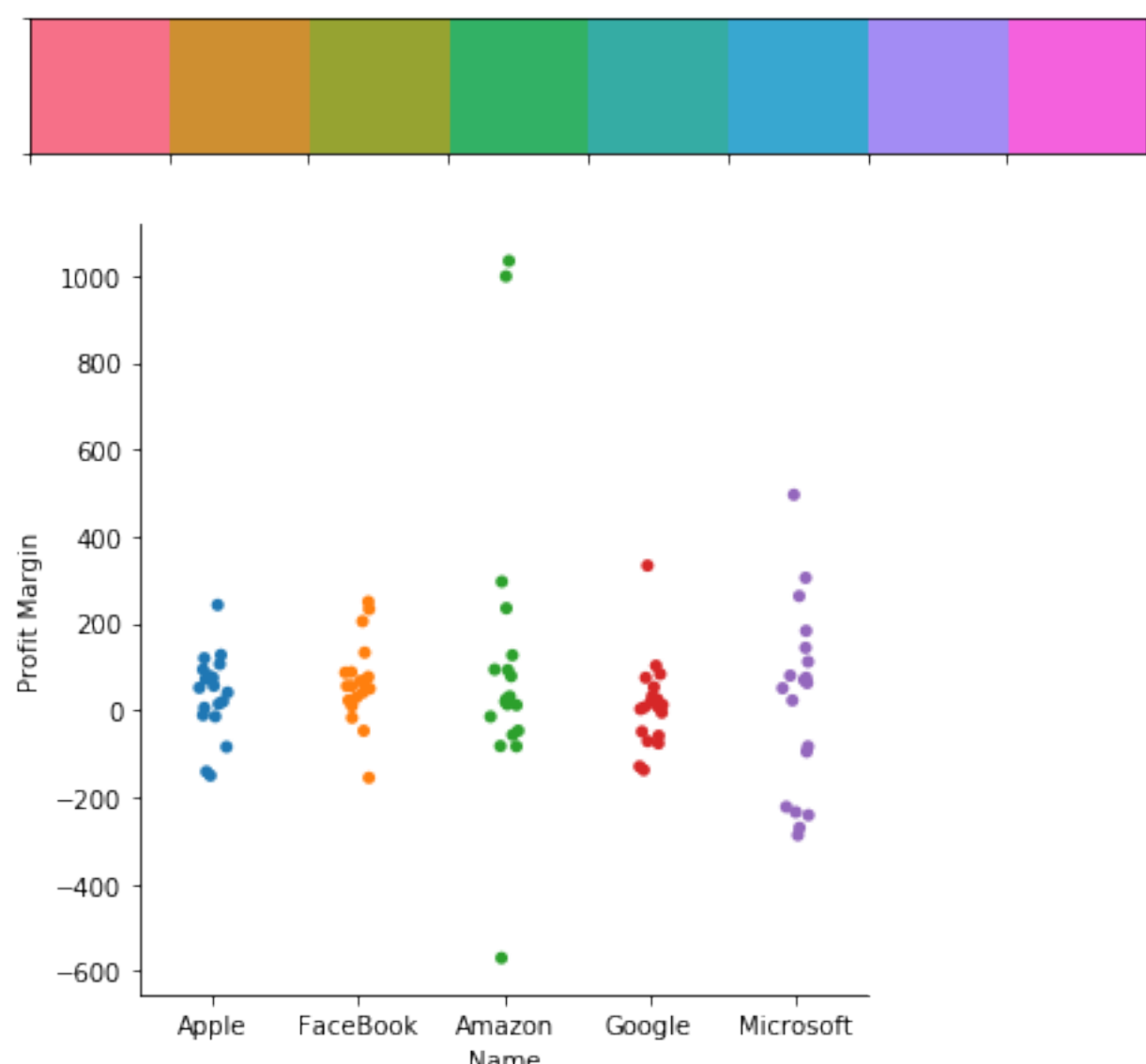
Name: 27, dtype: object

## Plot Showing Trades

```
In [21]: #This shows that Amazon had the most & least profitable transaction

Cpalet = sns.paletplot(sns.color_palette("husl", 8))
sns.catplot(x = "Name", y = "Profit Margin",hue = "Name",ci = None, color = Cpalet, data = df, aspect=1)
```

Out[21]: <seaborn.axisgrid.FacetGrid at 0x1a23fe9950>



## Total Profits for Month of February

```
In [27]: #Obtain total profits

Profits = df.groupby('Name')['Profit Margin'].sum()
Profits
```

Out[27]:

Name	
Amazon	2208.8
Apple	707.5
FaceBook	1241.0
Google	231.3
Microsoft	433.0

Name: Profit Margin, dtype: float64

```
In [25]: df[["Profit Margin","Date of Sale"]].head()
```

Out[25]:

	Profit Margin	Date of Sale
0	72.00	2020-02-03
1	77.00	2020-02-03
2	126.90	2020-02-03
3	75.15	2020-02-03
4	183.00	2020-02-03

```
In [28]: # Balance as in the account used to trade

Profits.sum() + 100000
```

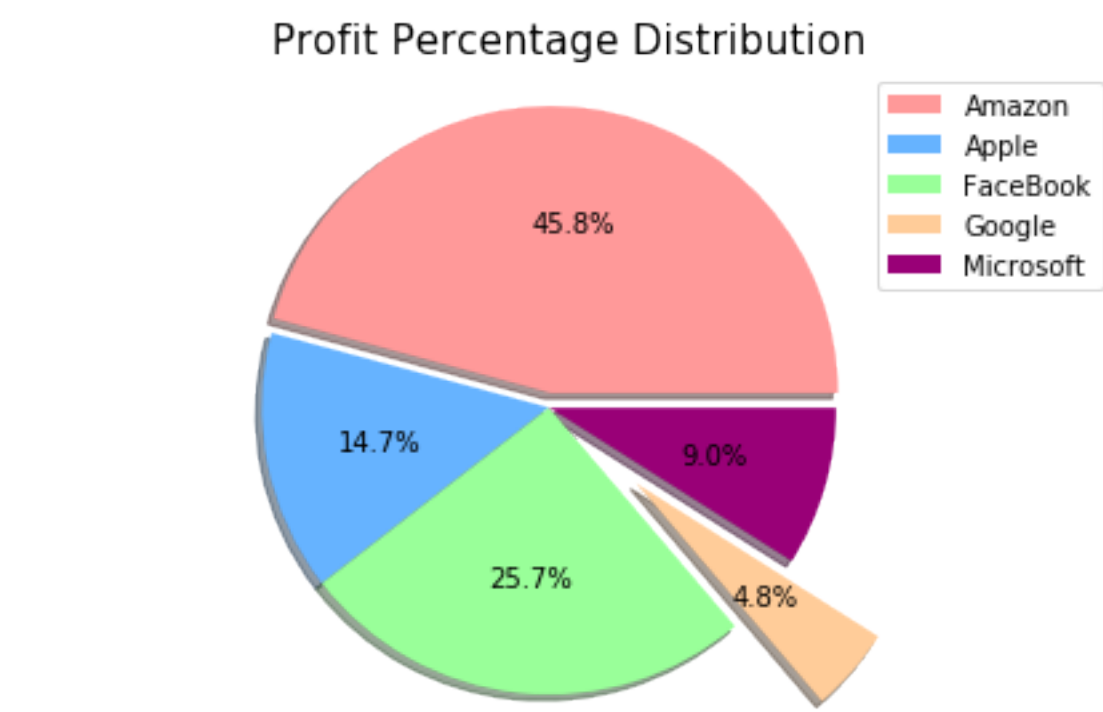
Out[28]: 104821.6

## Percentage Distribution

```
In [50]: labels = Profits.index
colors = colors = ['#ff9999','#66b3ff','#99ff99','#ffcc99','#990077']
explode = (0.05,0.0,0.,0.4,0.0)

plt.title('Profit Percentage Distribution', fontsize = 15)

plt.pie(Profits,autopct='%1.1f%%',explode = explode,colors = colors, shadow = True);
plt.axis('equal')
plt.legend(labels, loc = "upper right")
plt.tight_layout()
```

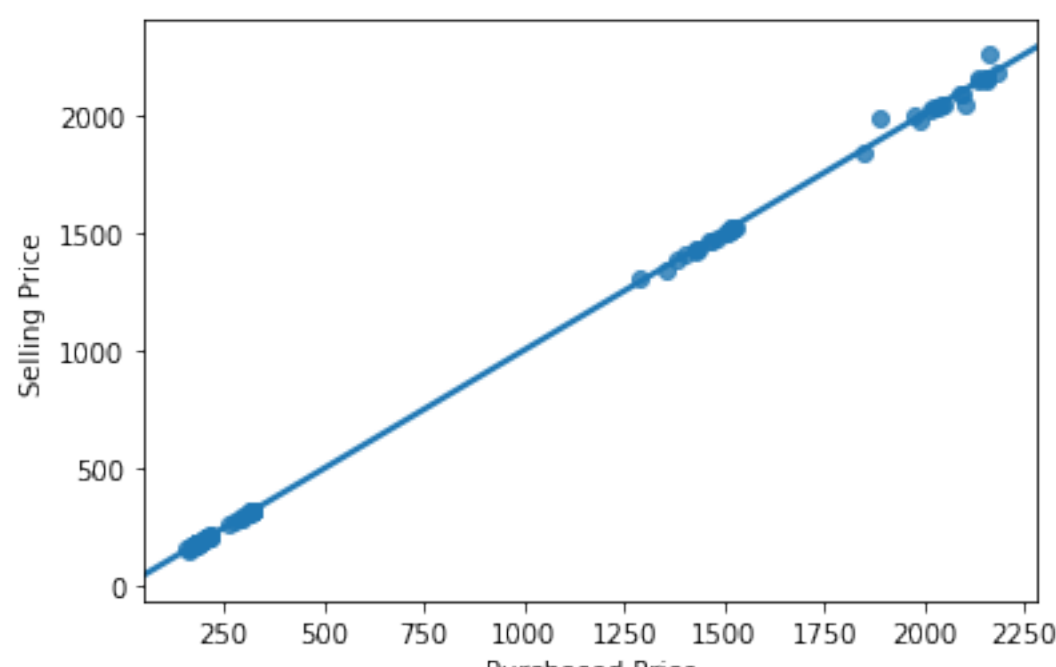


## Simple Linear Regression

```
In [72]: x = 'Purchased Price'
y = 'Selling Price'

sns.regplot(x,y, data = df, ci = None,fit_reg = True)
```

Out[72]: <matplotlib.axes.\_subplots.AxesSubplot at 0x1a28294310>



## Statistics

```
In [73]: df["Purchased Price"].mean()
```

Out[73]: 843.8557894736842

```
In [74]: df["Purchased Price"].std()
```

Out[74]: 780.9726520337547

```
In [75]: df["Purchased Price"].median()
```

Out[75]: 316.87

```
In [84]: df["Purchased Price"].var()
```

Out[84]: 609918.283224636

```
In [79]: df["Selling Price"].mean()
```

Out[79]: 846.6683157894737

```
In [80]: df["Selling Price"].std()
```

Out[80]: 784.5270568387218

```
In [81]: df["Selling Price"].median()
```

Out[81]: 319.1

```
In [83]: df["Selling Price"].var()
```

Out[83]: 615482.7029120271

```
In [ ]:
```