Project Development Phase

Sprint – 1 (Understanding the dataset)

Date	3 November 2022
Team ID	PNT2022TMID21528
Project Name	Project – Global Sales Data Analytics

1. Importing the important packages and importing dataset

```
In [2]: import numpy as np
        import pandas as pd
        import matplotlib.pyplot as plt
        import seaborn as sns
```

```
In [3]: df = pd.read_csv('Global_Superstore2.csv',encoding='latin-1')
```

2. Understanding the dataset

```
In [5]: df.head()
```

Out[5]:

	Row ID	Order ID	Order Date	Ship Date	Ship Mode	Customer ID	Customer Name	Segment	City	State	...	Product ID	Category	Sub-Category	Product Name	Sales
0	32298	CA-2012-124891	31-07-2012	31-07-2012	Same Day	RH-19495	Rick Hansen	Consumer	New York City	New York	...	TEC-AC-10003033	Technology	Accessories	Plantronics CS510 - Over-the-Head monaural Wir...	2309.650
1	26341	IN-2013-77878	05-02-2013	07-02-2013	Second Class	JR-16210	Justin Ritter	Corporate	Wollongong	New South Wales	...	FUR-CH-10003950	Furniture	Chairs	Novimex Executive Leather Armchair, Black	3709.395
2	25330	IN-2013-71249	17-10-2013	18-10-2013	First Class	CR-12730	Craig Reiter	Consumer	Brisbane	Queensland	...	TEC-PH-10004664	Technology	Phones	Nokia Smart Phone, with Caller ID	5175.171
3	13524	ES-2013-1579342	28-01-2013	30-01-2013	First Class	KM-16375	Katherine Murray	Home Office	Berlin	Berlin	...	TEC-PH-10004583	Technology	Phones	Motorola Smart Phone, Cordless	2892.510
4	47221	SG-2013-4320	05-11-2013	06-11-2013	Same Day	RH-9495	Rick Hansen	Consumer	Dakar	Dakar	...	TEC-SHA-10000501	Technology	Copiers	Sharp Wireless Fax, High-Speed	2832.960

5 rows × 24 columns

In [6]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 51290 entries, 0 to 51289
Data columns (total 24 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   Row ID          51290 non-null  int64  
 1   Order ID        51290 non-null  object 
 2   Order Date      51290 non-null  object 
 3   Ship Date       51290 non-null  object 
 4   Ship Mode       51290 non-null  object 
 5   Customer ID     51290 non-null  object 
 6   Customer Name   51290 non-null  object 
 7   Segment         51290 non-null  object 
 8   City            51290 non-null  object 
 9   State           51290 non-null  object 
 10  Country         51290 non-null  object 
 11  Postal Code     9994 non-null   float64
 12  Market          51290 non-null  object 
 13  Region          51290 non-null  object 
 14  Product ID      51290 non-null  object 
 15  Category        51290 non-null  object 
 16  Sub-Category    51290 non-null  object 
 17  Product Name    51290 non-null  object 
 18  Sales           51290 non-null  float64
 19  Quantity        51290 non-null  int64  
 20  Discount        51290 non-null  float64
 21  Profit          51290 non-null  float64
 22  Shipping Cost   51290 non-null  float64
 23  Order Priority  51290 non-null  object 
dtypes: float64(5), int64(2), object(17)
memory usage: 9.4+ MB
```

In [7]:
```
df.shape
```

Out[7]: (51290, 24)

In [8]:
```
df.describe()
```

Out[8]:

	Row ID	Postal Code	Sales	Quantity	Discount	Profit	Shipping Cost
count	51290.00000	9994.000000	51290.000000	51290.000000	51290.000000	51290.000000	51290.000000
mean	25645.50000	55190.379428	246.490581	3.476545	0.142908	28.610982	26.375915
std	14806.29199	32063.693350	487.565361	2.278766	0.212280	174.340972	57.296804
min	1.00000	1040.000000	0.444000	1.000000	0.000000	-6599.978000	0.000000
25%	12823.25000	23223.000000	30.758625	2.000000	0.000000	0.000000	2.610000
50%	25645.50000	56430.500000	85.053000	3.000000	0.000000	9.240000	7.790000
75%	38467.75000	90008.000000	251.053200	5.000000	0.200000	36.810000	24.450000
max	51290.00000	99301.000000	22638.480000	14.000000	0.850000	8399.976000	933.570000

3. Finding Missing values in dataset

```
In [9]: df.isnull().sum()
```

```
Out[9]: Row ID              0
        Order ID            0
        Order Date          0
        Ship Date           0
        Ship Mode           0
        Customer ID         0
        Customer Name       0
        Segment             0
        City                0
        State               0
        Country             0
        Postal Code     41296
        Market              0
        Region              0
        Product ID          0
        Category            0
        Sub-Category        0
        Product Name        0
        Sales               0
        Quantity            0
        Discount            0
        Profit              0
        Shipping Cost       0
        Order Priority      0
        dtype: int64
```

4. Working with the data

```
In [11]: df.nunique()
```

```
Out[11]: Row ID          51290
         Order ID        25035
         Order Date       1430
         Ship Date        1464
         Ship Mode           4
         Customer ID      1590
         Customer Name     795
         Segment             3
         City             3636
         State            1094
         Country           147
         Postal Code       631
         Market              7
         Region             13
         Product ID       10292
         Category             3
         Sub-Category        17
         Product Name      3788
         Sales            22995
         Quantity            14
         Discount            27
         Profit           24575
         Shipping Cost    10037
         Order Priority       4
         dtype: int64
```

```
In [12]: df_customer = df[['Customer ID','Order ID','Order Date', 'Ship Date', 'Ship Mode','Country']]
         df_customer.count()
```

```
Out[12]: Customer ID    51290
         Order ID       51290
         Order Date     51290
         Ship Date      51290
         Ship Mode      51290
         Country        51290
         dtype: int64
```

5. Correlating the Integer data

```
In [16]: df.drop(['Row ID','Postal Code'],axis=1,inplace=True)
```

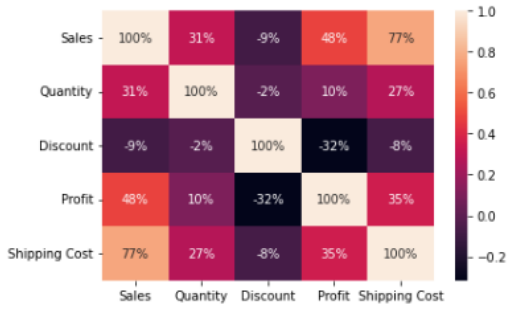
```
In [17]: df.corr()
```

Out[17]:

	Sales	Quantity	Discount	Profit	Shipping Cost
Sales	1.000000	0.313577	-0.086722	0.484918	0.768073
Quantity	0.313577	1.000000	-0.019875	0.104365	0.272649
Discount	-0.086722	-0.019875	1.000000	-0.316490	-0.079056
Profit	0.484918	0.104365	-0.316490	1.000000	0.354441
Shipping Cost	0.768073	0.272649	-0.079056	0.354441	1.000000

```
In [18]: sns.heatmap(df.corr(),annot= True,fmt='.0%')
```

Out[18]: <AxesSubplot:>



Conclusion

- There is no column with missing data except postal code which can be ignored because it doesn't make big difference for visualization
- Some of the categorical column in dataset are ship mode, segment, market, region, category and sub-category
- As the result of correlation, we can conclude that as the discount increases the profit decreases
- As the result of correlation, sales and profit are positively related