Part I - (Analysis on Sales and Market Perfomance)

by (David Kipngeno Kiplangat)

Introduction

> The sales data is contains records of sales profits and items sold from different regions and segments and by different sales persons from a business environment.

Preliminary Wrangling

In [31]:
```
# import all packages and set plots to be embedded inline
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
# disabling warning
import warnings
warnings.filterwarnings('ignore')
```

> Load in your dataset and describe its properties through the questions below. Try and motivate your exploration goals through this section.

In [32]:
```python
# Loading the sales data from the csv file
data = pd.read_csv('SalesData.csv')
# Glance Understanding of the data
# getting the shape of the data
print(data.shape)
# understanding the data info and types
print(data.info())
# checking the head of the data
print(data.head())
```

```
(9976, 29)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9976 entries, 0 to 9975
Data columns (total 29 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   CustomerID       9976 non-null   object 
 1   CustomerName     9976 non-null   object 
 2   BusinessSegment  9976 non-null   object 
 3   Country          9976 non-null   object 
 4   Region           9976 non-null   object 
 5   State            9976 non-null   object 
 6   City             9976 non-null   object 
 7   PostalCode       9976 non-null   int64  
 8   Order ID         9976 non-null   object 
 9   Order Date       9976 non-null   object 
 10  ShipID           9976 non-null   float64
 11  ItemNum          9976 non-null   object 
 12  OrderQty         9976 non-null   float64
 13  Discount         9976 non-null   float64
 14  Ship Date        9976 non-null   object 
 15  Ship Mode        9976 non-null   object 
 16  Manufacture      9976 non-null   object 
 17  Category         9976 non-null   object 
 18  Sub-Category     9976 non-null   object 
 19  Product Name     9976 non-null   object 
 20  Price            9976 non-null   float64
 21  Cost             9976 non-null   float64
 22  Year             9976 non-null   float64
 23  Month            9976 non-null   float64
 24  Day              9976 non-null   float64
 25  MarkedPrice      9976 non-null   float64
 26  BuyingPrice      9976 non-null   float64
 27  SellingPrice     9976 non-null   float64
 28  Profit           9976 non-null   float64
dtypes: float64(12), int64(1), object(16)
```

```
memory usage: 2.2+ MB
None
    CustomerID   CustomerName BusinessSegment        Country   Region  \
0  A33717C73120  Aaron Bergman        Consumer  United States  Central
1  A33717C73120  Aaron Bergman        Consumer  United States  Central
2  A33717C76017  Aaron Bergman        Consumer  United States  Central
3  A33717W98103  Aaron Bergman        Consumer  United States     West
4  A33717W98103  Aaron Bergman        Consumer  United States     West

        State           City  PostalCode        Order ID  Order Date  ...  \
0    Oklahoma  Oklahoma City       73120  CA-2013-140935  2020-11-11  ...
1    Oklahoma  Oklahoma City       73120  CA-2013-140935  2020-11-11  ...
2       Texas      Arlington       76017  CA-2011-152905  2018-02-19  ...
3  Washington        Seattle       98103  CA-2011-156587  2018-03-07  ...
4  Washington        Seattle       98103  CA-2011-156587  2018-03-07  ...

                                       Product Name     Price       Cost  \
0  Sauder Facets Collection Library, Sky Alder Fi...  142.8000  74.764398
1                                  Samsung Convoy 3   76.4444  22.286997
2                                 Akro Stacking Bins    7.1538   4.041695
3            Carina 42"Hx23 3/4"W Media Storage Unit   74.3636  41.543911
4                                        Newell 330    5.4545   3.099148

     Year Month   Day MarkedPrice BuyingPrice SellingPrice     Profit
0  2020.0  11.0  13.0    142.8000   74.764398     142.8000  68.035602
1  2020.0  11.0  13.0     76.4444   22.286997      76.4444  54.157403
2  2018.0   2.0  25.0     14.3076    8.083390      14.1076   6.024210
3  2018.0   3.0   8.0    223.0908  124.631732     223.0908  98.459068
4  2018.0   3.0   8.0     16.3635    9.297443      16.3635   7.066057

[5 rows x 29 columns]
```

- **Data Descriptive Statistics**:
  - computing and understanding the data sammary statistics and its composition for numerical columns only.

In [33]:
```
data.describe()
```

Out[33]:

	PostalCode	ShipID	OrderQty	Discount	Price	Cost	Year	Month	Day	MarkedPrice	BuyingPrice
count	9976.000000	9976.000000	9976.000000	9976.000000	9976.000000	9976.000000	9976.000000	9976.000000	9976.000000	9976.000000	9976.000000
mean	55195.237670	556349.119186	3.705293	0.121227	60.736603	31.252270	2019.739074	7.737169	15.856756	223.906328	115.347387
std	32055.423413	259837.152240	2.337438	0.154716	137.097198	70.757570	1.128790	3.346413	8.807517	595.794296	302.165385
min	1040.000000	100030.000000	1.000000	0.000000	0.682900	0.414804	2018.000000	1.000000	1.000000	0.733300	0.414804
25%	23223.000000	331344.000000	2.000000	0.000000	5.666700	3.684211	2019.000000	5.000000	8.000000	17.268200	10.886013
50%	56560.000000	556714.000000	3.000000	0.000000	16.686700	9.277181	2020.000000	9.000000	16.000000	53.342200	30.045813
75%	90008.000000	784421.000000	5.000000	0.200000	61.225800	32.303371	2021.000000	11.000000	24.000000	201.333400	103.566485
max	99301.000000	999631.000000	18.000000	0.500000	3773.000000	1587.628866	2022.000000	12.000000	31.000000	26411.000000	10868.724279

What is the structure of your dataset?

> The data contains a total of 29 columns and 9976 records of data. Most of the variables are numeric in nature with a few of categorical ones; region : east, west, south profits is a continous variable

What is/are the main feature(s) of interest in your dataset?

> The main features in the data are the cost, price, region , segment and the net profit attracted from Sales

What features in the dataset do you think will help support your investigation into your feature(s) of interest?

> I expect that the growth of sales has a corresponding growth of profits so are the cost. I also expect that the profit margins grows progressively with time.

Univariate Exploration

In [34]:
```
data.columns
```

Out[34]:
```
Index(['CustomerID', 'CustomerName', 'BusinessSegment', 'Country', 'Region',
       'State', 'City', 'PostalCode', 'Order ID', 'Order Date', 'ShipID',
       'ItemNum', 'OrderQty', 'Discount', 'Ship Date', 'Ship Mode',
       'Manufacture', 'Category', 'Sub-Category', 'Product Name', 'Price',
       'Cost', 'Year', 'Month', 'Day', 'MarkedPrice', 'BuyingPrice',
       'SellingPrice', 'Profit'],
      dtype='object')
```

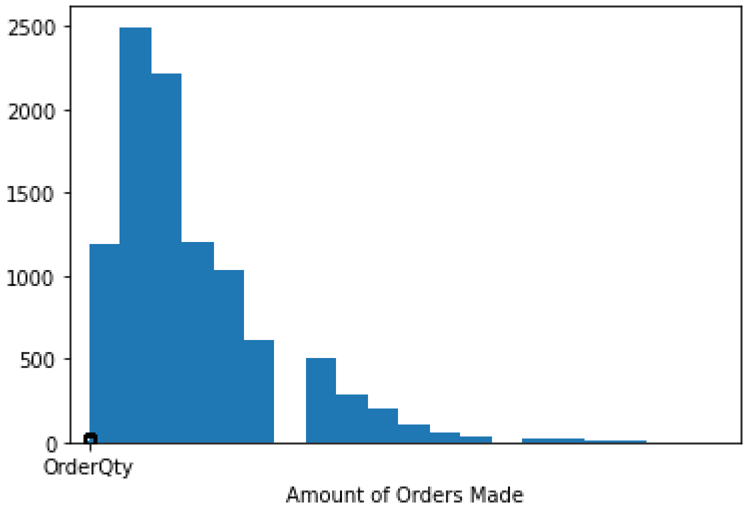
In [35]:
```
# creating numerical var holder
numeric_cols = ['OrderQty', 'Discount','Cost','Price','Profit']
# plotting univariate analysis
uni = data[numeric_cols]
# uni.plot(kind= 'hist')
```

I would like to understand the order quantity perfomance

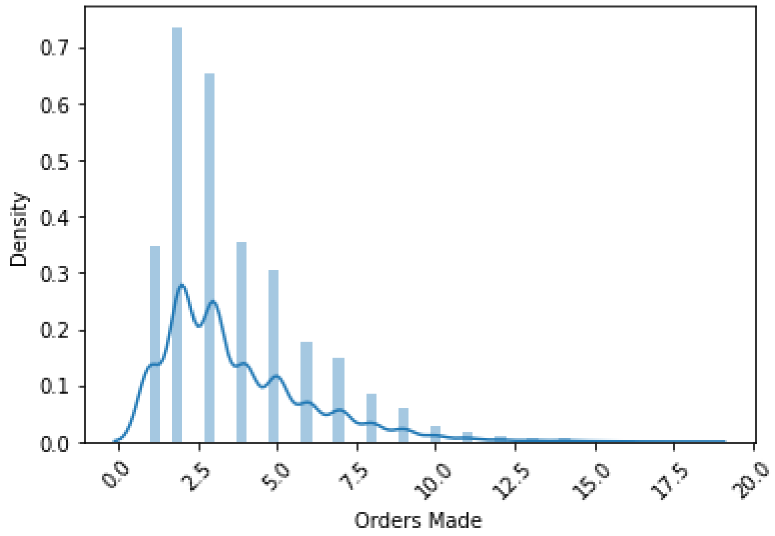
In [36]:
```
print(data['OrderQty'].describe())
# understanding the data distribution
data['OrderQty'].plot(kind='box')
# plotting a histogram for the quantity column
plt.hist(data=data,x='OrderQty',bins = 20)
plt.xlabel('Amount of Orders Made')
```

```
count    9976.000000
mean        3.705293
std         2.337438
min         1.000000
25%         2.000000
50%         3.000000
75%         5.000000
max        18.000000
Name: OrderQty, dtype: float64
```

Out[36]:
```
Text(0.5, 0, 'Amount of Orders Made')
```



In [37]:
```
# plotting a distribution of the orders sales.
sns.distplot(data['OrderQty'])
plt.xticks(rotation=45)
plt.xlabel('Orders Made')
plt.show()
```



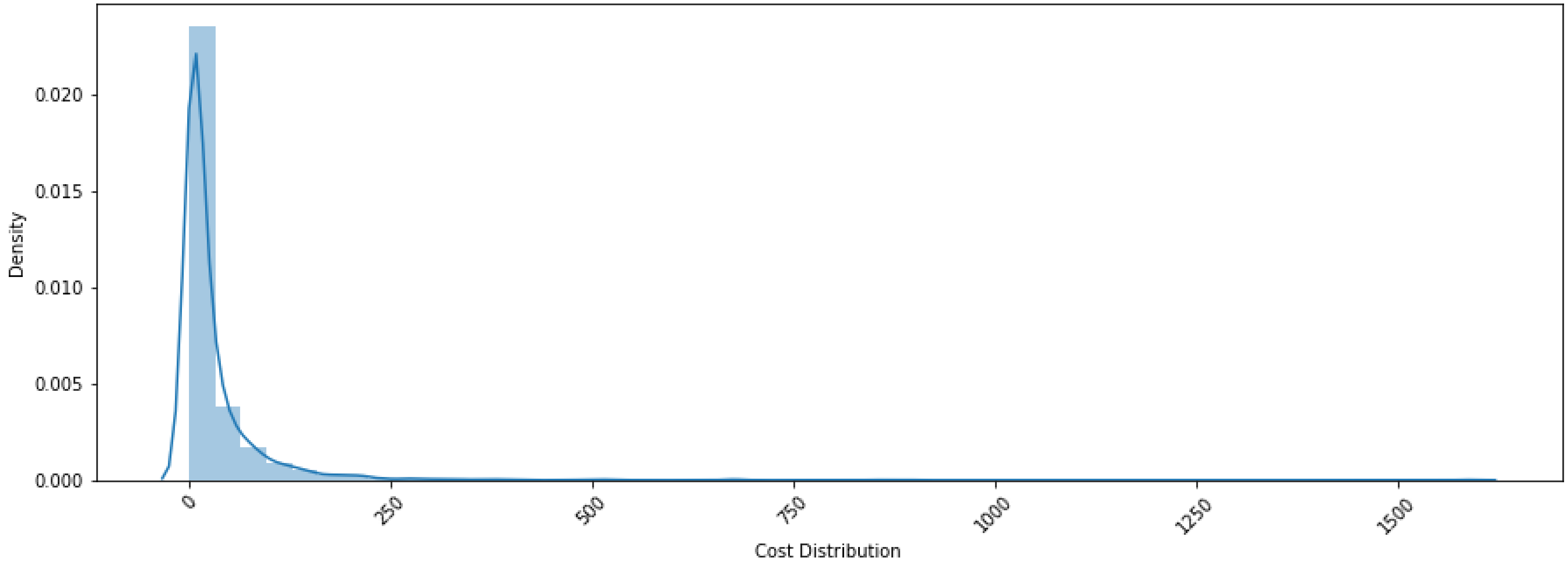
Observations It can be observed that the distribution of orders started at a good foot, with progressing time, it lowered progressivley, the distributioni therefore is said to be skewed towards left.

In [38]:
```
# understanding the cost distribution
pd.DataFrame(data.Cost.describe()).T
```

Out[38]:

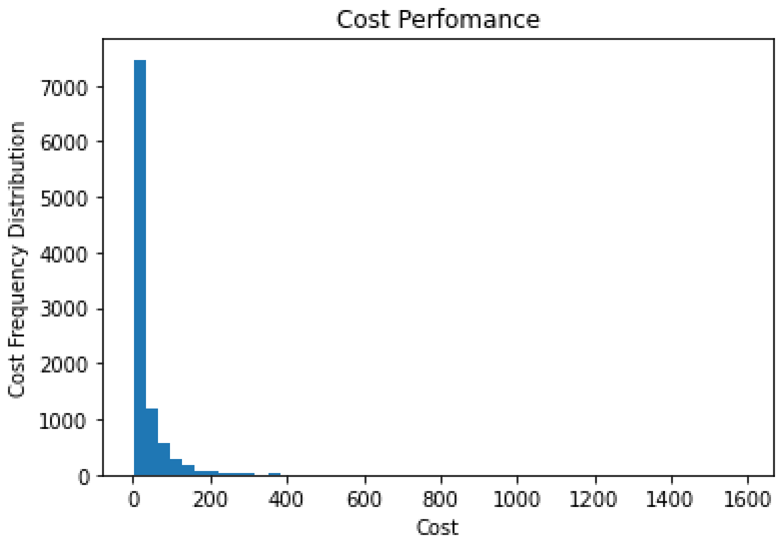
	count	mean	std	min	25%	50%	75%	max
Cost	9976.0	31.25227	70.75757	0.414804	3.684211	9.277181	32.303371	1587.628866

In [39]:
```
# understanding the cost in deeper view.
def plot_cost(data):
    plt.figure(figsize=[15,5])
    sns.distplot(data.Cost,bins = 50)
    plt.xticks(rotation=45)
    plt.xlabel('Cost Distribution')
    plt.show()
# calling the function
plot_cost(data)
```



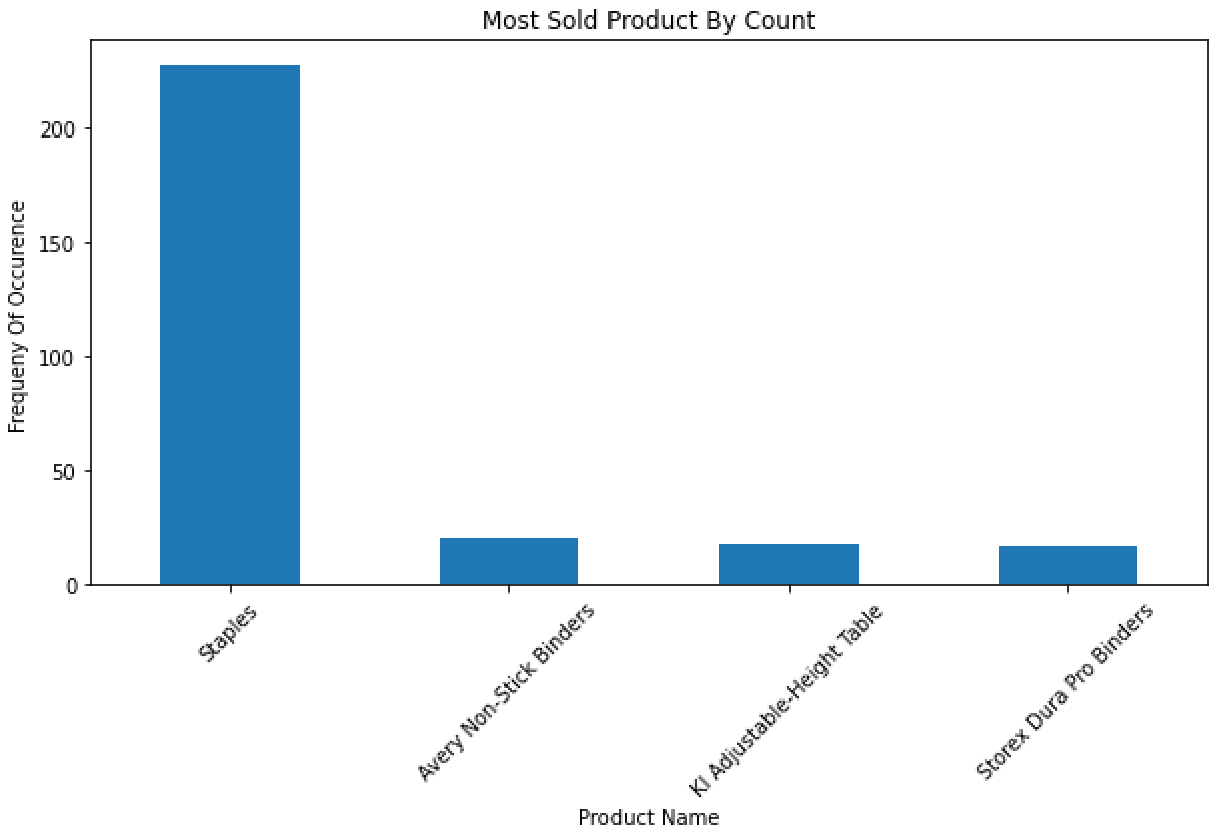
The minimum cost of a product is 31 on approximate while the maximum is approximately 1588. The cost is observed to be skewed towards left and the minority is towards the right tail of the distribution.

In [40]:
```
def plot_cost(data):
    data.Cost.plot(kind='hist',bins=50)
    plt.title("Cost Perfomance")
    plt.xlabel("Cost")
    plt.ylabel("Cost Frequency Distribution")
    plt.show()
plot_cost(data)
```



For the cost variable distribution, the distributioin is skewed towards left. Most orders lies between the price within the range of 1 to 50.

In [41]:
```
# plotting the most sold product.
def plot_most_sold_product(data):
    most_sold_product = data['Product Name'].value_counts().head(4)
    plt.figure(figsize=[10,5])
    most_sold_product.plot(kind='bar')
    plt.xlabel('Product Name')
    plt.ylabel('Frequeny Of Occurence')
    plt.title("Most Sold Product By Count")
    plt.xticks(rotation=45)
    plt.show()
# calling the function
plot_most_sold_product(data)
```



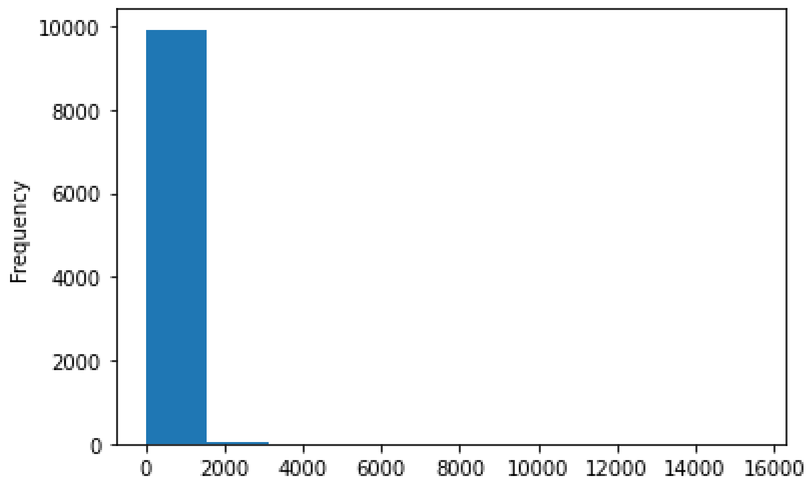
The product staples commanded the most sales ny count as can be observed, the storex dura pro binders come last

- **Understanding the Profit Perfomance**

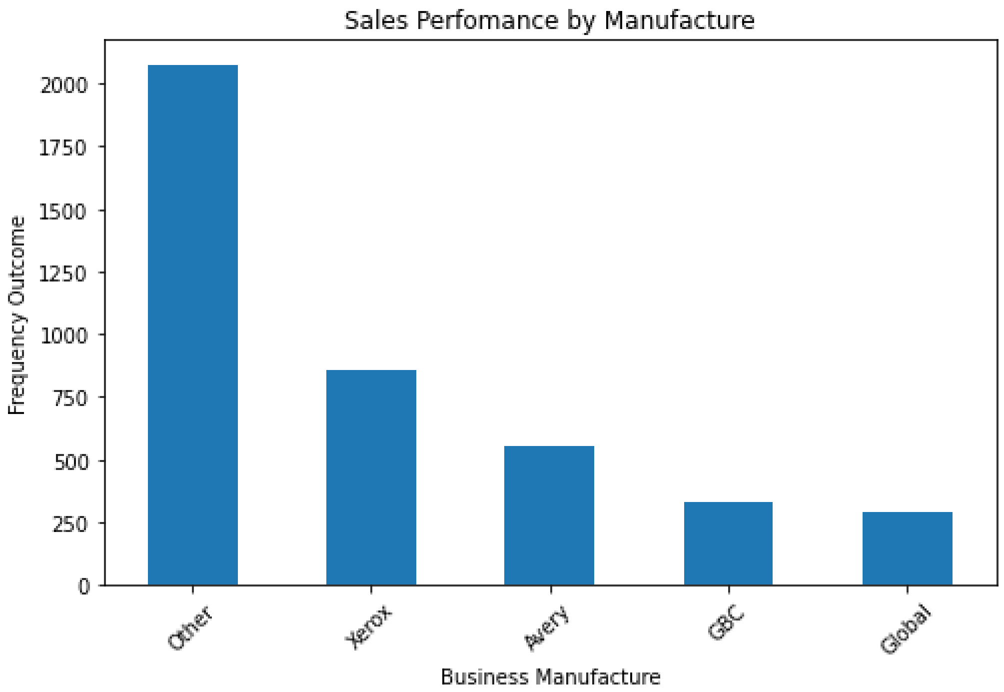
```
- Profit Distribution outcome
```

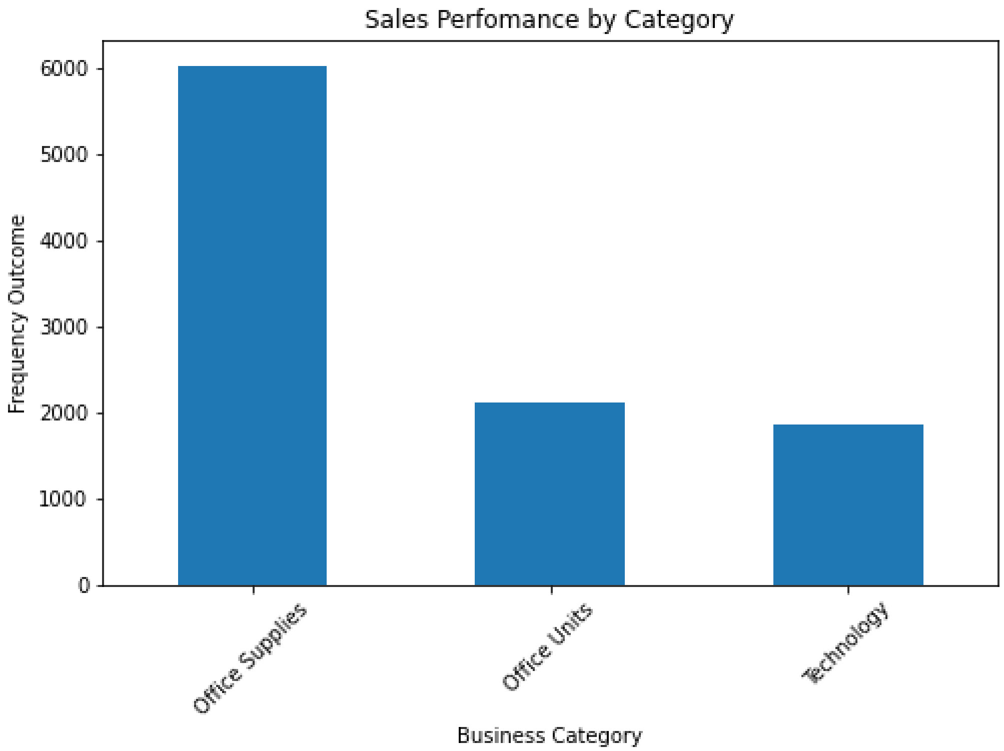
In [42]:
```
data.Profit.plot(kind='hist', bins = 10)
```

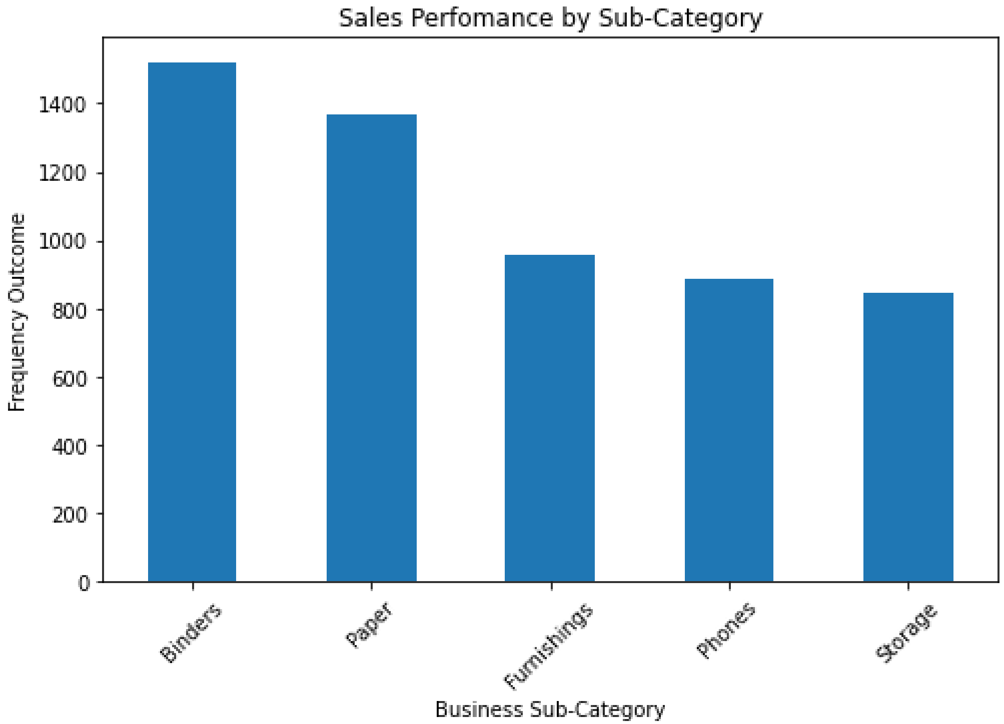
Out[42]:
```
<AxesSubplot:ylabel='Frequency'>
```

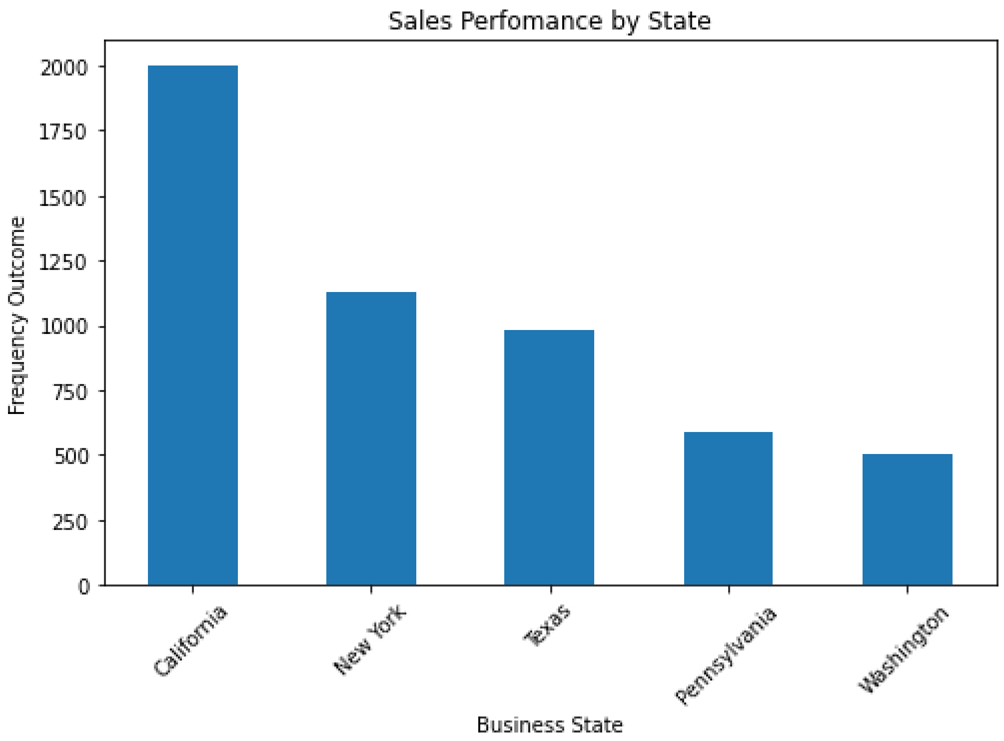


In [43]:
```
Categorical_columns = ['Manufacture','Category','Sub-Category','State']
for col in Categorical_columns:
#     creating the plotting data
    dataplot = data[col].value_counts().head()
#     setting the figure size
    plt.figure(figsize=[8,5])
    dataplot.plot(kind='bar')
    plt.xlabel(f'Business {col}')
    plt.ylabel('Frequency Outcome')
    plt.title(f'Sales Perfomance by {col}')
    plt.xticks(rotation=45)
    plt.show()
```




Sales Perfomance by Category
Frequency Outcome
6000
5000
4000
3000
2000
1000
0
Office Supplies
Office Units
Technology
Business Category


Sales Perfomance by Sub-Category
Frequency Outcome
0
200
400
600
800
1000
1200
1400
Binders
Paper
Furnishings
Phones
Storage
Business Sub-Category

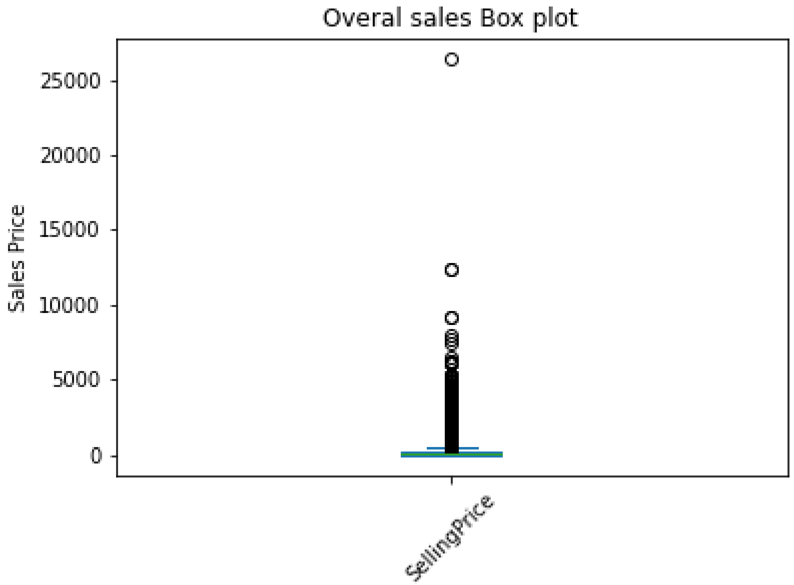


manufacturer: The plot of manufacturer is as shown in the first plot above in the categorical plots. The manufacturer type of other was observed to have the majority count. Following was the manufacturer of xerox then was the very. Global come last. It can be concluded that other manufucturer took the market by far off as compared to other manufacturers.

supplies: As for the sapllies of the items, the suplie of office supplie was the predominant , followed by the office units and lastly was the supplier of the technology.

Sub Category : The subcategory would help to understand the distribution of sales by the subcategory. As for this, the subcategory of binders was the leading followed by the category of paper, furnishing and lastly was the subcategory of storage.

In [44]:
```
def plot_sales(data):
    selling=data.SellingPrice
    selling.plot(kind='box')
    plt.xticks(rotation=45)
    plt.title('Overal sales Box plot')
    plt.ylabel("Sales Price")
plot_sales(data)
```

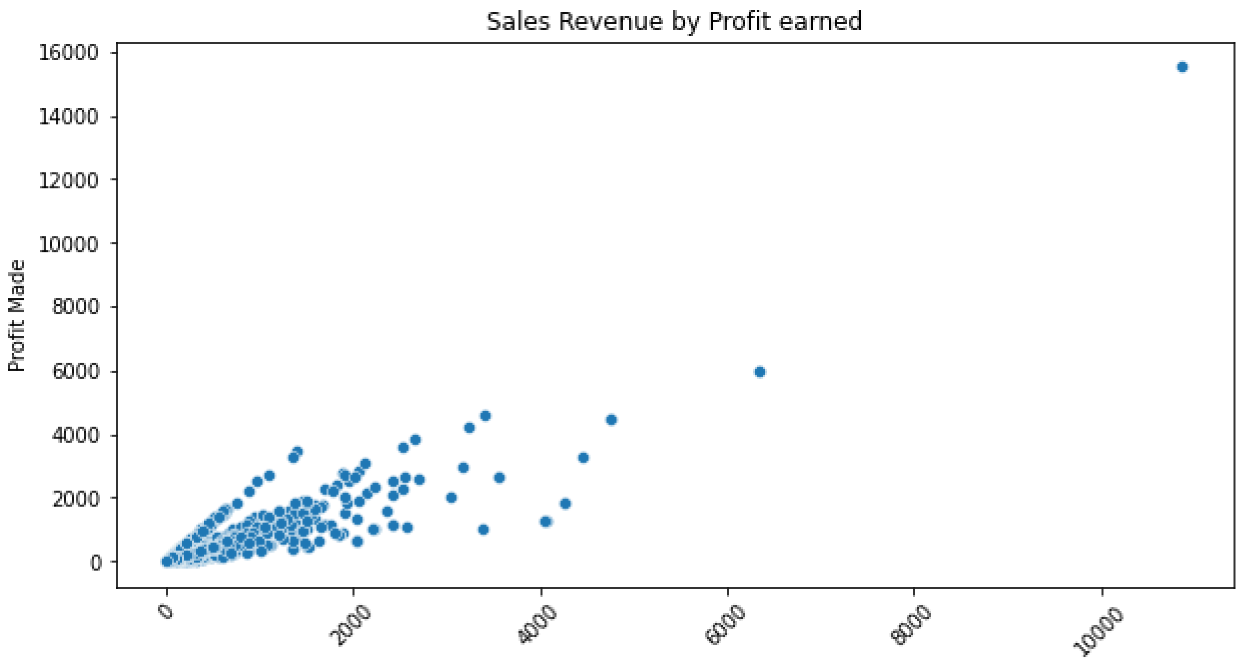


> **Overal Sales** : There was an outlier in the overal sales made, it was an observation made that there was a product sold at a very high price. This was conclusion was reached from the visualization of the box plot above. This was the far i went with univariate data analysis is concerned.

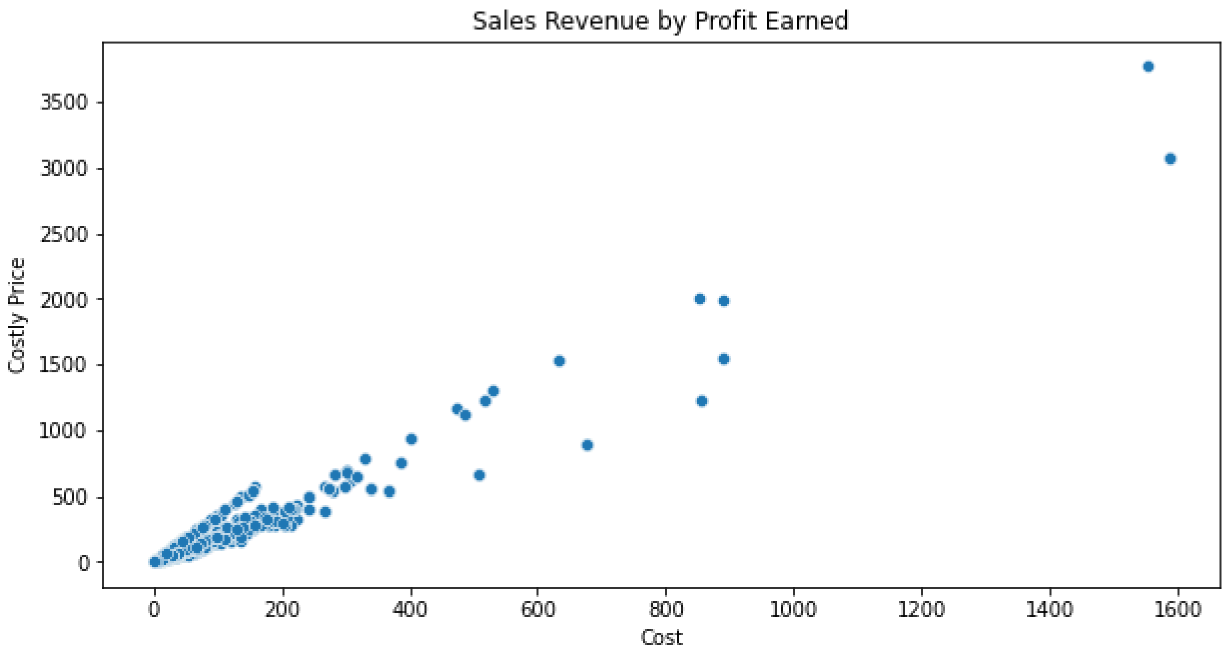
Bivariate Exploration

> My focus in with bivariate analysis was to find the correlation of variables in the dataset, I compute the correlation and visualize the result. The strongly correlated features would greatly helped to predict the outcome and likely project the future outcome of the business.

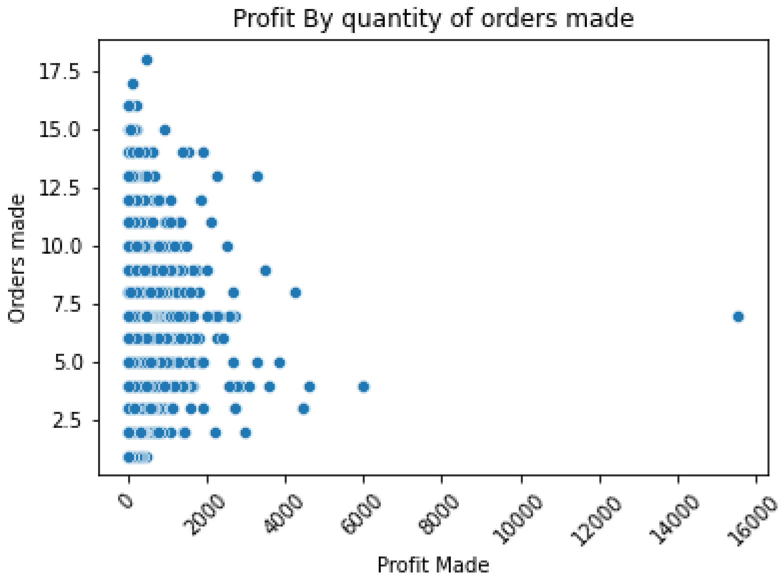
In [45]:
```python
def plot_Profit_by_sales(data):
    # setting the figure size
    plt.figure(figsize=[10,5])
    # plotting the data
    sns.scatterplot(data=data,x='BuyingPrice',y='Profit')
    # labelling the figure
    plt.xlabel('Buying Price')
    plt.ylabel('Profit Made')
    plt.title('Sales Revenue by Profit earned')
    plt.xticks(rotation=45)
    plt.show()
plot_Profit_by_sales(data)
```



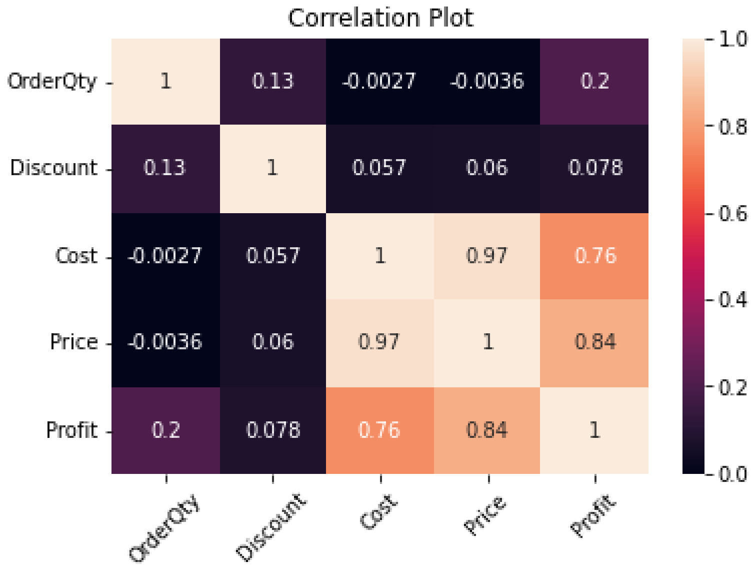
In [46]:
```
def plot_Profit_by_sales(data):
    # setting the figure size
    plt.figure(figsize=[10,5])
    # plotting the data
    sns.scatterplot(data=data,x='Cost',y='Price')
    # labelling the figure
    plt.xlabel('Cost')
    plt.ylabel('Costly Price')
    plt.title('Sales Revenue by Profit Earned')
    plt.show()
plot_Profit_by_sales(data)
```



In [47]:
```
def plot_quantity_by_order(data):
    # plotting the data
    sns.scatterplot(data=data,x='Profit',y='OrderQty')
    # labelling the figure
    plt.ylabel('Orders made')
    plt.xlabel('Profit Made')
    plt.title('Profit By quantity of orders made')
    plt.xticks(rotation=45)
    plt.show()
plot_quantity_by_order(data)
```



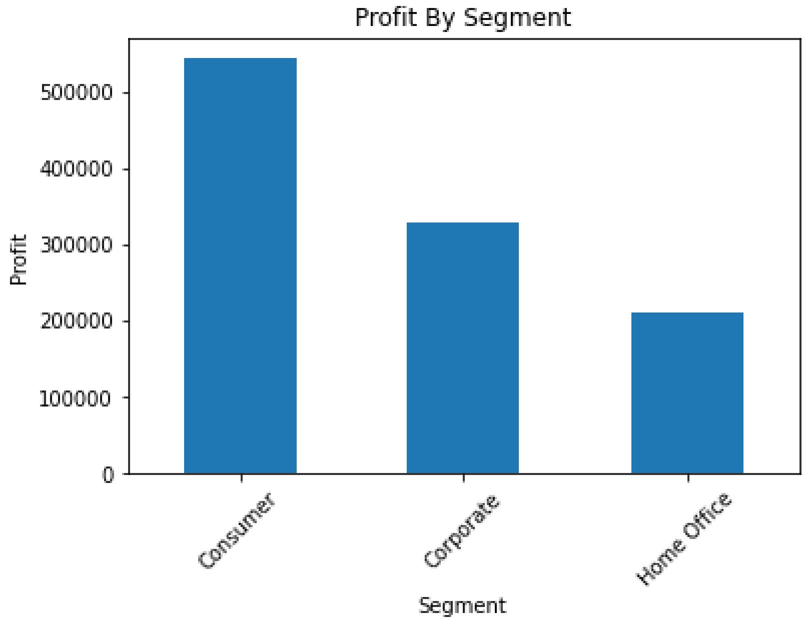
In [48]:
```
def compute_and_plot_correlation(data):
    # computing the data corelation
    # datacorr = data.corr()
    numeric_cols = ['OrderQty', 'Discount','Cost','Price','Profit']
    data_numerical = data[numeric_cols]
    numerical_corr = data_numerical.corr()
    sns.heatmap(data=numerical_corr,annot=True)
    plt.xticks(rotation=45)
    plt.title('Correlation Plot')
compute_and_plot_correlation(data)
```



Segment Perfomance by Profit

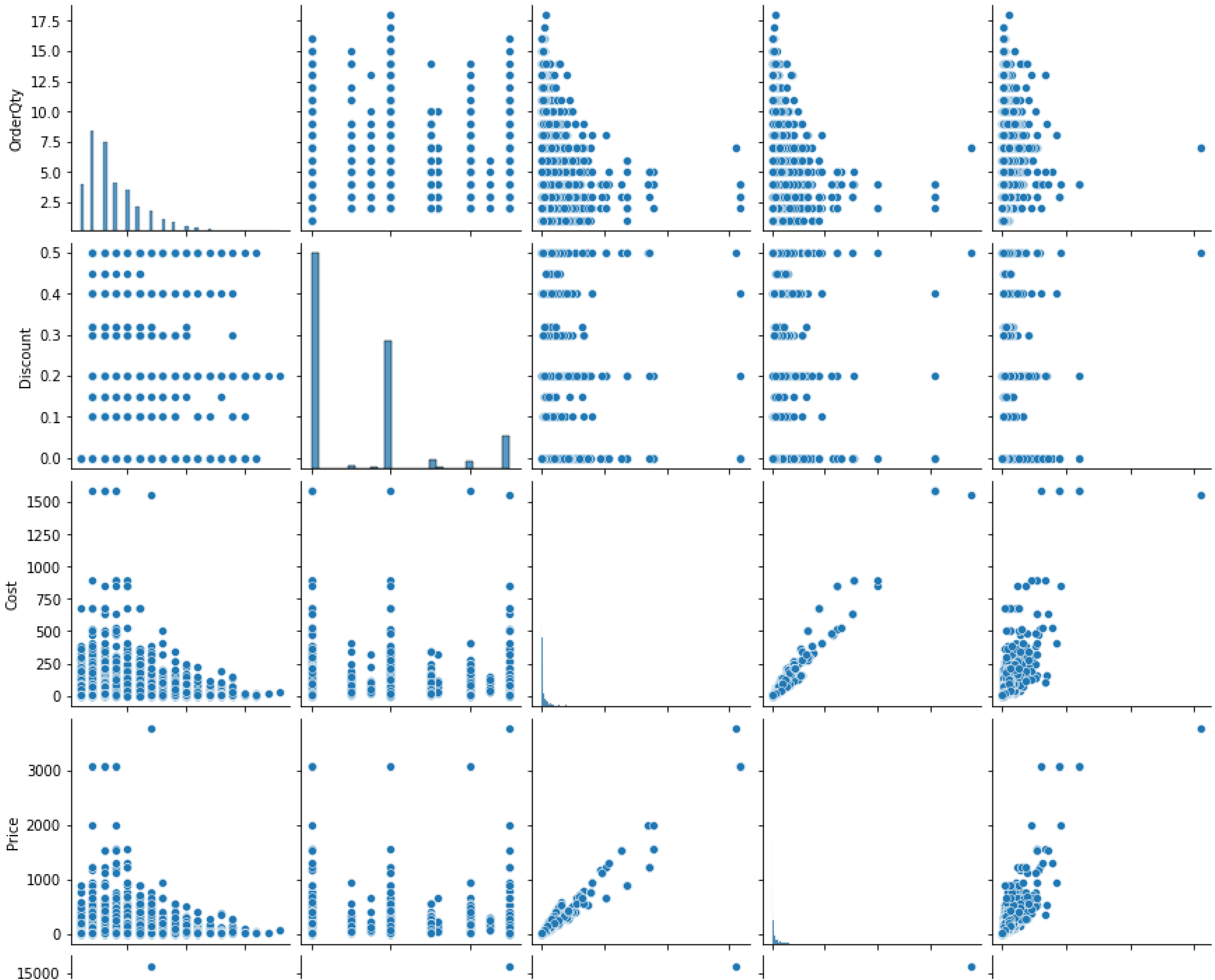
In [49]:

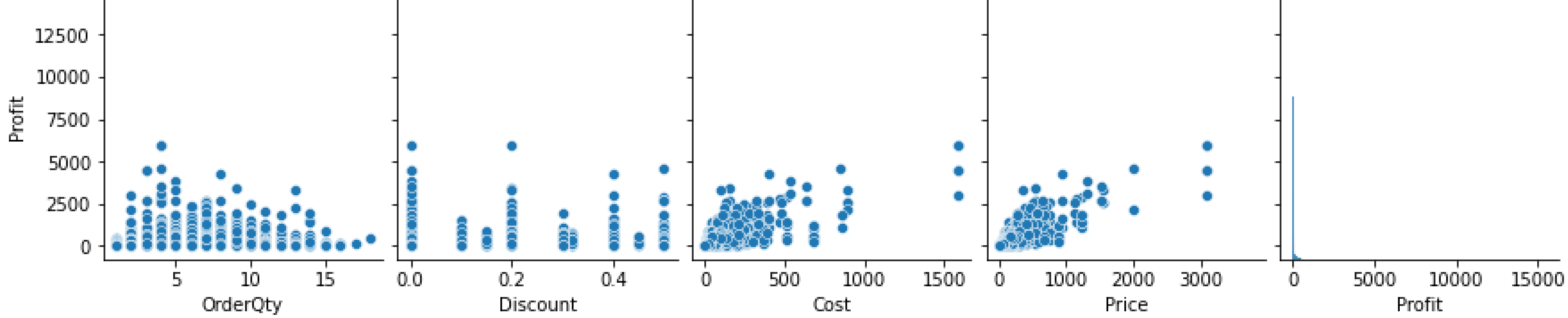
```
# creating a function to plot the perfomance by segment.
def plot_profit_by_segment(data):
    profit_by_segment = data.groupby('BusinessSegment')['Profit'].sum()
    profit_by_segment.plot(kind = 'bar')
    plt.xlabel('Segment')
    plt.ylabel('Profit')
    plt.title('Profit By Segment')
    plt.xticks(rotation=45)
    plt.show()
# calling the function.
plot_profit_by_segment(data)
```



In [50]:
```
data_numerical= data[numeric_cols]
sns.pairplot(data_numerical)
```

Out[50]: <seaborn.axisgrid.PairGrid at 0x20af896aaf0>





Talk about some of the relationships you observed in this part of the investigation. How did the feature(s) of interest vary with other features in the dataset?

> It was my observations that the profit was strongly correlated to the cost of commodity, also the price had a linear relationship with a strong positve correlations. I can deduce therefore that the price could be a predictor variable to the cost as it depicted a linear function.

Did you observe any interesting relationships between the other features (not the main feature(s) of interest)?

> It was interesting to me that the overall profit as compared to the variables of cost and price appeared to be constant. There was an extreme profit made which appeared to be an outlier to me. This profit data point was as onbserved from the data.

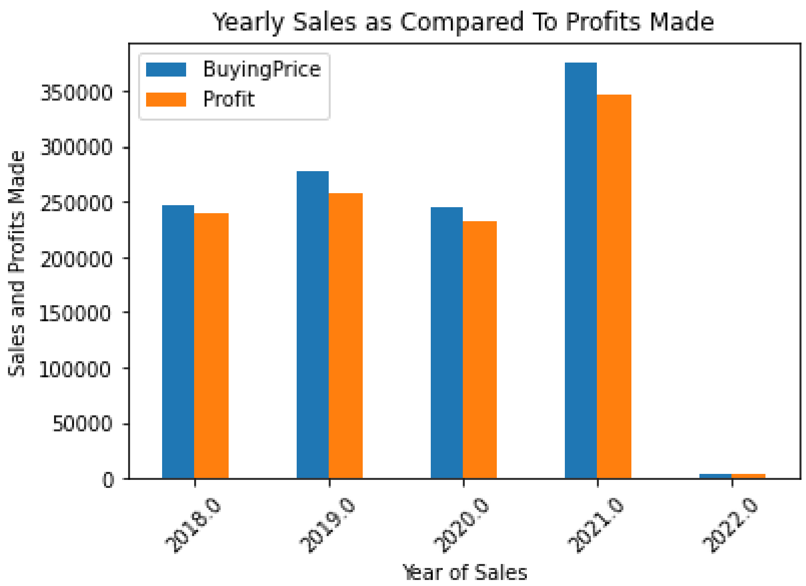
Multivariate Exploration

> I intend to understand the relationship between multiple variables in this phase. I will investigate the behavior and make visualization with coresponding observations from the data. I have structured this section to answer specific questions mentioned below.

Question 1 : What was the yearly sales perfomance?

In [51]:

```
# creating a function
def yearly_sales(data):
    yearlySales = data.groupby('Year')[['BuyingPrice','Profit']].sum()
    yearlySales = pd.DataFrame(yearlySales)
    yearlySales.plot(kind = 'bar')
    plt.xlabel('Year of Sales')
    plt.ylabel('Sales and Profits Made')
    plt.title('Yearly Sales as Compared To Profits Made')
    plt.xticks(rotation=45)
    plt.show()
# calling the function
yearly_sales(data)
```

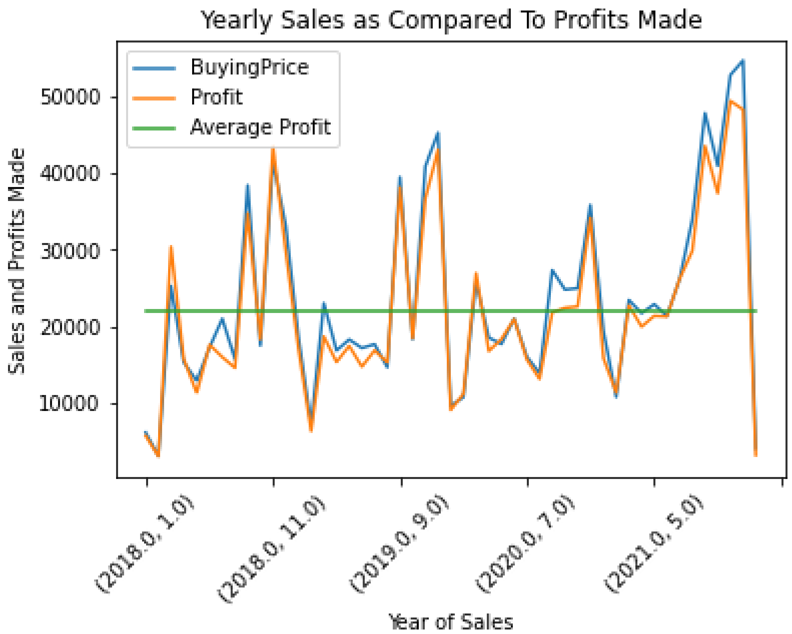


Findings the year of 2021 appeared to have a good perfomance of both profit and sales made from it. The year can therefore be considered the best month for the business.

Intrestingly, there was bearly a harvest in the year of 2022.

Question 2 what was the Yearly Perfomance by Profits?

In [52]:
```
# creating a function to plot the yearly perfomance by profits.
def average_monthly_sales(data):
    yearlySalesmonthly = data.groupby(['Year','Month'])[['BuyingPrice','Profit']].sum()
    yearlySalesmonthly = pd.DataFrame(yearlySalesmonthly)
    yearlySalesmonthly['Average Profit'] = yearlySalesmonthly.Profit.mean()
    yearlySalesmonthly.plot()
    plt.xlabel('Year of Sales')
    plt.ylabel('Sales and Profits Made')
    plt.title('Yearly Sales as Compared To Profits Made')
    plt.xticks(rotation=45)
    plt.show()
# calling the functions.
average_monthly_sales(data)
```

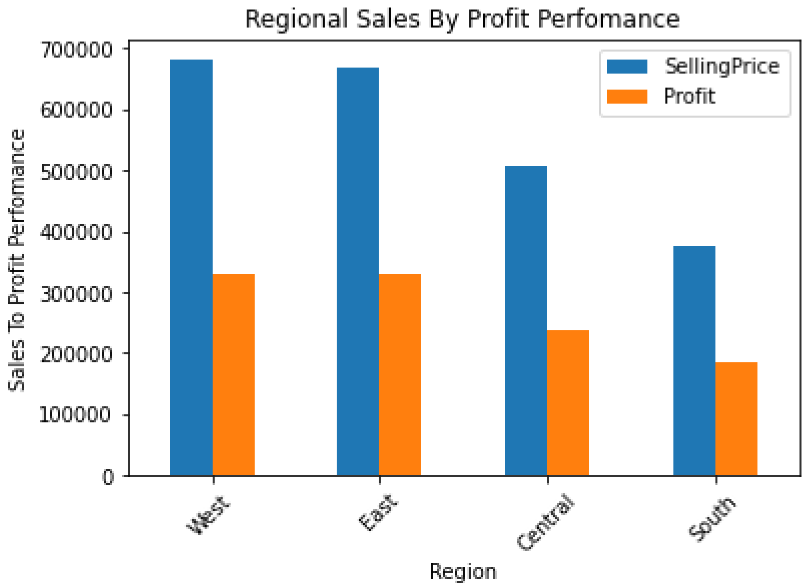


Findings From the figure above shows the perfomance of the sales and profits over time through out the years of sales. The line of average profit perfomance was plotted to show the threshold.

It was observed that there were years where the profit gain was below the average. This instances can be flagged as danger areas and more market compaigns is required on the products made to make the business live. This consclusions were reached after a careful observations of the figure.

Question 3 What was the Regional Sales Perfomance

In [53]:
```
# creating the function to explore the question.
def plot_regional_perfomance_bar(data):
    regional_sales_profit = data.groupby('Region')[['SellingPrice','Profit']].sum().sort_values('Profit',ascending=False)
    regional_sales_profit.plot(kind = 'bar')
    plt.xticks(rotation=45)
    plt.title("Regional Sales By Profit Perfomance")
    plt.xlabel("Region")
    plt.ylabel("Sales To Profit Perfomance")
    plt.show()
plot_regional_perfomance_bar(data)
```

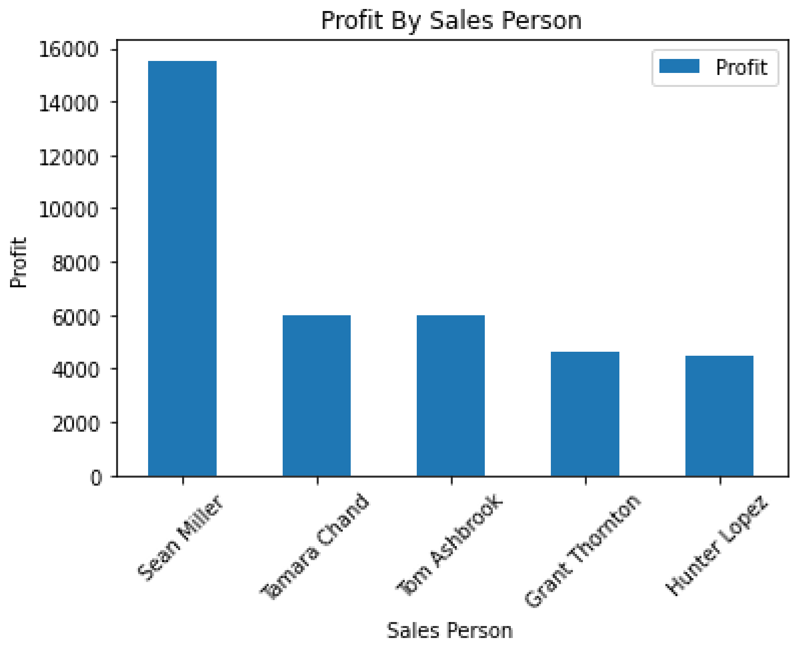


In [54]:
```
def plot_regional_perfomance_pie(data):
    regional_sales_profit = data.groupby('Region')['Profit'].sum()
    plt.title("Regional Sales By Profit Perfomance")
    regional_sales_profit.plot(kind='pie')
plot_regional_perfomance_pie(data)
```

Question 4: Who [top 5] brought the most profits and from which business segment and by what amount of Profit

In [55]:

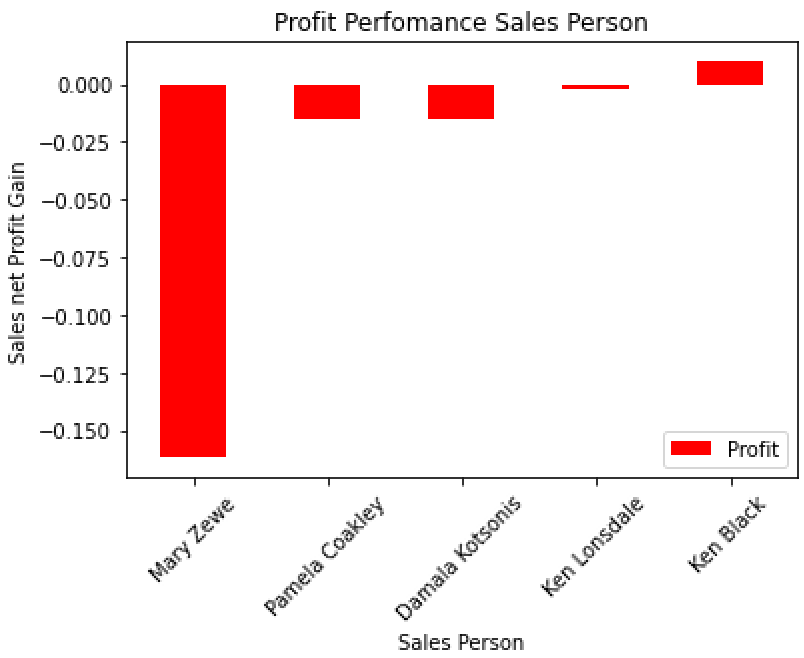
```
# creating a fucntion to answer the question.
def plot_best_sellers(data):
    best_seller = data.sort_values(['Profit'],ascending= False)
    best_seller = best_seller[['CustomerName','Profit','BusinessSegment','Region','State','City']]
    best_seller = best_seller.set_index('CustomerName')
    best_seller.head().plot(kind='bar')
    plt.xlabel('Sales Person')
    plt.ylabel('Profit')
    plt.title('Profit By Sales Person')
    plt.xticks(rotation=45)
    plt.show()
# calling the function.
plot_best_sellers(data)
```



Question 5: Who [Bottom 5] brought the most profits and from which business segment and by what amount of Profit

```
In [56]: def plot_least_perfomming_seller(data):
             least_seller = data.sort_values(['Profit'])
             least_seller = least_seller[['CustomerName','Profit','BusinessSegment','Region','State','City']]
             least_seller = pd.DataFrame(least_seller)
             least_seller = least_seller.set_index('CustomerName')
             plt.figure(figsize=[8,8])
             least_seller.head().plot(kind='bar',color = 'r')
             plt.xticks(rotation=45)
             plt.xlabel("Sales Person")
             plt.ylabel("Sales net Profit Gain")
             plt.title("Profit Perfomance Sales Person")
             plt.show()
             print(least_seller.head())
         plot_least_perfomming_seller(data)
```

```
<Figure size 576x576 with 0 Axes>
```




```
                   Profit BusinessSegment   Region        State         City
CustomerName
Mary Zewe       -0.161554       Corporate  Central        Texas    Arlington
Pamela Coakley  -0.015361       Corporate     West     Colorado     Loveland
Damala Kotsonis -0.015361       Corporate     East Pennsylvania Philadelphia
Ken Lonsdale    -0.001802        Consumer  Central        Texas      Houston
Ken Black        0.009782       Corporate    South      Florida      Hialeah
```

Talk about some of the relationships you observed in this part of the investigation. Were there features that strengthened each other in terms of looking at your feature(s) of interest?

> There was a singificantly huge relationship between the sales made and the profit attracted,a significnat trend of both sales and profits was also noted with a significant imporvement each year, however the was drop largy in the year of 2020.

Were there any interesting or surprising interactions between features?

> **poor perfoming Sales person** : Mary bought an insignificant profit from her sales , he was followed by pamela coakley. they can be said to have been prbably employing the wrong strategies of sales.

> I observed that the profit for west and east was almost similar, it was however intresting how much the sales were higher for the west region. It occured to me that, more sales had to be made on west region to attract a significant profit as compared to the region of east.

Conclusions

> The sales or profit of a given item is strongly influenced by the numer of orders of item sold with a corresponding cost of each. This analysis has paved way for an exploration of sales made and the profit respectivily. This analysis is important in the direction that it allows the project of sales and the likely profits to be realized from the sales. This just a few of the findings from analysis. The outcome of this analysis is used for making appropriate informaed decision making and allow the decision making faculty utilize vailable opportunities and resource to better strategies on its sales to reaize the maximum profits possible.

In [60]:
```
# !jupyter nbconvert Part_I_david_sales_analysis.ipynb --to slides --post serve --no-input --no-prompt
```

In [ ]: