The Sparks Foundation

Data Science & Bussiness Analytics Intern (May-2022)

Aurthor: Preeti Hegde

Task 3: EXPLORATORY DATA ANALYSIS - RETAIL

Perform Exploratory Data Analysis on Dataset 'Samplesuperstore'

1. Business Problem

- As a business manager, where do you invest more to get more profit?
- Perform Exploratory data analysis to get the important insights about sales, profit.
- Creating a dashboard which tells the story of the data

1.1 Sub-questions needed to be answered:

- What are the optimal values for each feature to get maximum profit?
- Which city should I focus in each state to get maximum profit?
- PowerBI dashboard, which tells the story about the profit.

3. Data Overview

- **target : Profit (numerical)**
- Ship Mode : 9994 non-null object
- Segment : 9994 non-null object
- Country : 9994 non-null object
- City : 9994 non-null object
- State : 9994 non-null object
- Postal Code: 9994 non-null int64
- Region : 9994 non-null object
- Category : 9994 non-null object
- Sub-Category: 9994 non-null object
- Sales : 9994 non-null float64
- Quantity : 9994 non-null int64
- Discount : 9994 non-null float64
- Profit : 9994 non-null float64

2 Analysis

2.1 Data cleaning and Exploratory Data Analysis

Libraries and data...

In [1]:

```
#importing libraries...
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

import warnings
warnings.filterwarnings("ignore")
```

In [2]:

```
#Loading the data
data = pd.read_csv("SampleSuperstore (1).csv")
```

In [3]:

```
#data shape
print("Number of records present in the data: ",data.shape[0])
print("Number of columns present in the data: ",data.shape[1])
```

```
Number of records present in the data:  9994
Number of columns present in the data:  13
```

In [4]:

```
# Let's look at the first 5 rows of the dataframe
data.head()
```

Out[4]:

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Sub-Category	
0	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Bookcases	26
1	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Chairs	73
2	Second Class	Corporate	United States	Los Angeles	California	90036	West	Office Supplies	Labels	1
3	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Furniture	Tables	95
4	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Office Supplies	Storage	2

In [5]:

```
#Let us look at the high level info of the dataframe
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9994 entries, 0 to 9993
Data columns (total 13 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   Ship Mode     9994 non-null   object 
 1   Segment       9994 non-null   object 
 2   Country       9994 non-null   object 
 3   City          9994 non-null   object 
 4   State         9994 non-null   object 
 5   Postal Code   9994 non-null   int64  
 6   Region        9994 non-null   object 
 7   Category      9994 non-null   object 
 8   Sub-Category  9994 non-null   object 
 9   Sales         9994 non-null   float64
 10  Quantity      9994 non-null   int64  
 11  Discount      9994 non-null   float64
 12  Profit        9994 non-null   float64
dtypes: float64(3), int64(2), object(8)
memory usage: 1015.1+ KB
```

Insights:

- **There are 8 categorical variables, 1 numerical float variable and 1 numerical count variable, and 1 variable which is present as numerical but it is actually categorical (postal code).**
- **And we have 1 numerical target variable Profit**

In [6]:

```
#Let us look at the descriptive statistics
data.describe()
```

Out[6]:

	Postal Code	Sales	Quantity	Discount	Profit
count	9994.000000	9994.000000	9994.000000	9994.000000	9994.000000
mean	55190.379428	229.858001	3.789574	0.156203	28.656896
std	32063.693350	623.245101	2.225110	0.206452	234.260108
min	1040.000000	0.444000	1.000000	0.000000	-6599.978000
25%	23223.000000	17.280000	2.000000	0.000000	1.728750
50%	56430.500000	54.490000	3.000000	0.200000	8.666500
75%	90008.000000	209.940000	5.000000	0.200000	29.364000
max	99301.000000	22638.480000	14.000000	0.800000	8399.976000

Insights:

In [7]:

```
# Check for null values..
data.isnull().sum()
```

Out[7]:

```
Ship Mode       0
Segment         0
Country         0
City            0
State           0
Postal Code     0
Region          0
Category        0
Sub-Category    0
Sales           0
Quantity        0
Discount        0
Profit          0
dtype: int64
```

Insights:

- **There are no missing values in any of the column in this dataframe**

The dataframe seems to be clean. There is no much effort needed for cleaning. But by proceeding further we can identify if there are any outliers or categories repeated due to spelling mistakes. We can then clean the data.

Correlation analysis

In [8]:

```
a=data.corr()
a
```

Out[8]:

	Postal Code	Sales	Quantity	Discount	Profit
Postal Code	1.000000	-0.023854	0.012761	0.058443	-0.029961
Sales	-0.023854	1.000000	0.200795	-0.028190	0.479064
Quantity	0.012761	0.200795	1.000000	0.008623	0.066253
Discount	0.058443	-0.028190	0.008623	1.000000	-0.219487
Profit	-0.029961	0.479064	0.066253	-0.219487	1.000000

In [9]:

```
import seaborn as sns
sns.heatmap(a,annot=True,cmap='Blues')
plt.show()
```

- Sales and Profit are Moderately Correlated
- Discount and Profit are Negatively Correlated

Analysis of each features

In [10]:

```
def descriptive(feature):
    """Returns the descriptive statistics of the given feature"""

    descriptive = pd.DataFrame()
    descriptive["minimum"] = [data[feature].min()]
    descriptive["maximum"] = [data[feature].max()]
    descriptive["mean"] = [data[feature].mean()]
    descriptive["median"] = [data[feature].median()]
    descriptive["mode"] = [data[feature].mode()[0]]

    return descriptive
```

- **Target variable "Profit" Analysis**

In [11]:

```
descriptive("Profit")
```

Out[11]:

	minimum	maximum	mean	median	mode
0	-6599.978	8399.976	28.656896	8.6665	0.0

- **Profit has negative values also. So these can be considered as loss.**
- **And mean and median values of profit is very low compared to maximum value.**
- **Mode is 0, it seems for most of the products, there is no loss or profit.**

In [12]:

```
descriptive("Sales")
```

Out[12]:

	minimum	maximum	mean	median	mode
0	0.444	22638.48	229.858001	54.49	12.96

In [13]:

```
descriptive("Discount")
```

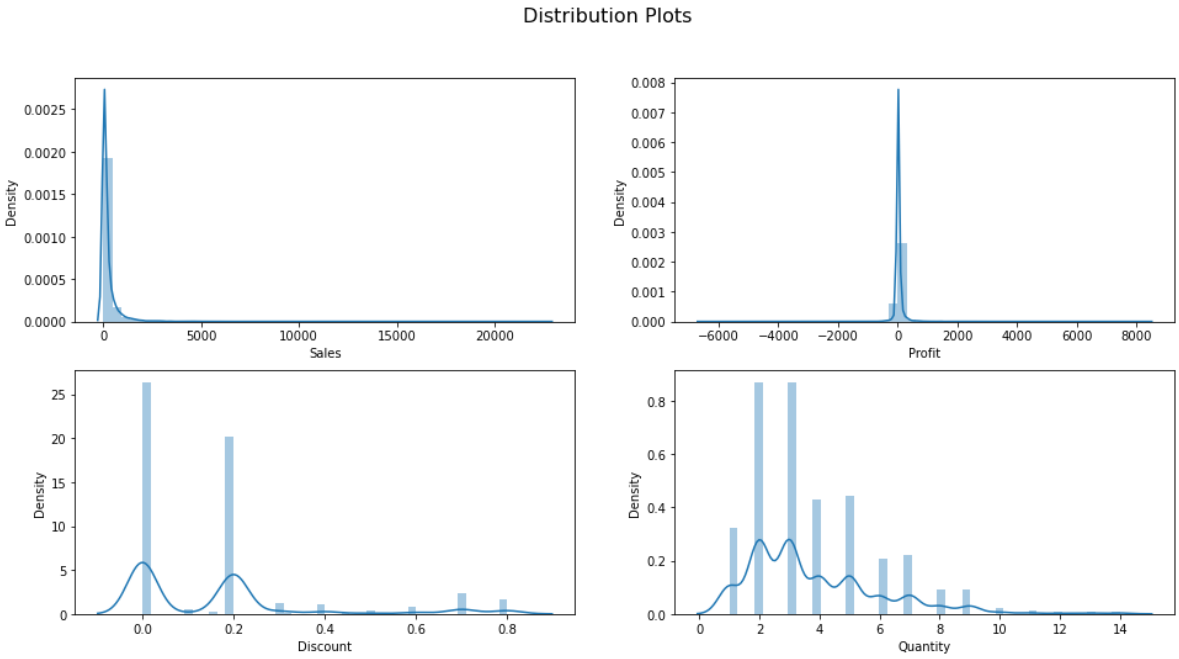
Out[13]:

	minimum	maximum	mean	median	mode
0	0.0	0.8	0.156203	0.2	0.0

PDF of sales, profit, discount and quantity.

In [14]:

```
fig, axes = plt.subplots(2,2, figsize = (16, 8))
fig.suptitle("Distribution Plots", fontsize = 16)
sns.distplot(data['Sales'], ax=axes[0,0])
sns.distplot(data['Profit'], ax=axes[0,1])
sns.distplot(data['Discount'], ax=axes[1,0])
sns.distplot(data['Quantity'], ax=axes[1,1])
plt.show()
```

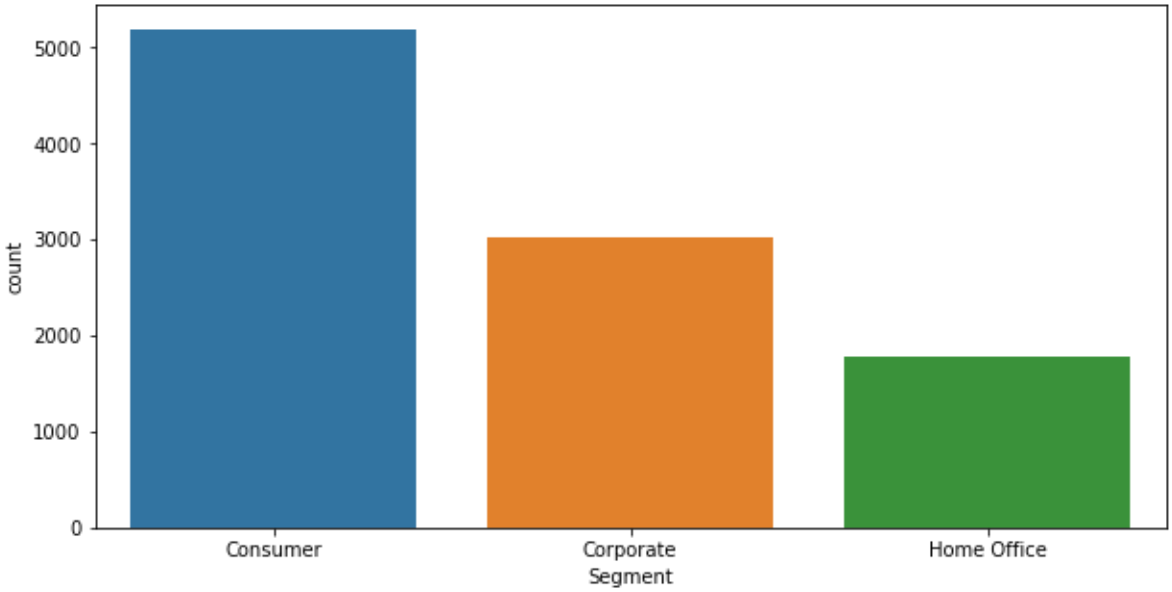


- Distribution plot of Sales rightly skewed (positively). There are very few values which are greater than 5000.
- From the pdf of Profit, we can see that there are very few products which has very high profit as well as very few products which has huge loss (negative profit).
- From pdf of discount, we can see that, there are 2 peaks near 0 and 0.2. And there are few values which has discount greater than 0.3
- Most of the product quantity is less than 8.

Count plot of segment

In [15]:

```
plt.rcParams['figure.figsize']=(10,5)
sns.countplot(x=data.Segment)
plt.show()
```

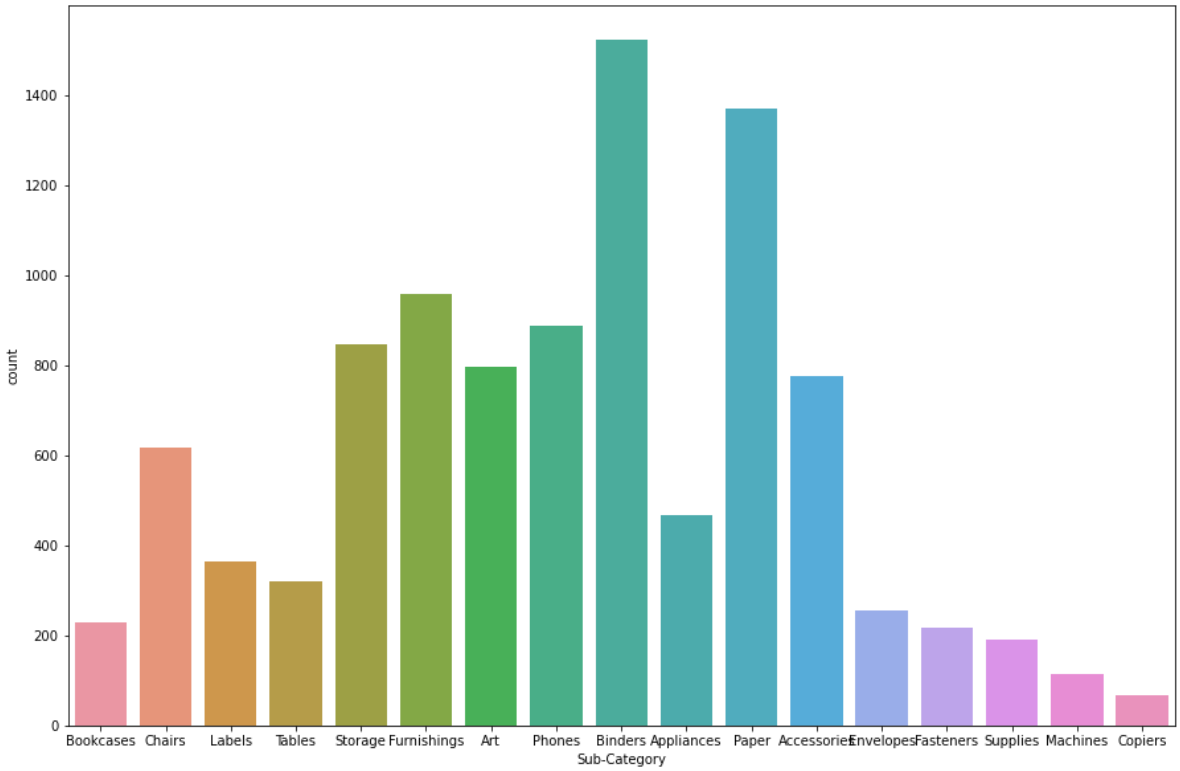


- Consumer segment products are more in this data and Home office segment products are less.

Bar chart of Sub-category

In [16]:

```
plt.figure(figsize=(15,10))
sns.countplot(data['Sub-Category'])
plt.show()
```

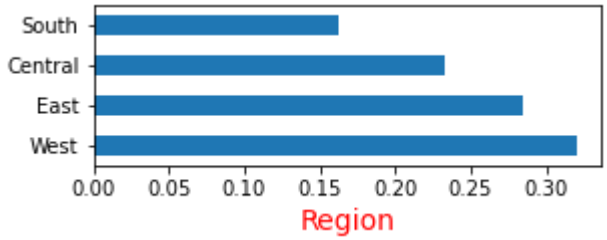


- Above plot shows Binders and Paper are more orderd by customers and Machines and Copiers are less ordered.

Region wise sale counts

In [17]:

```
plt.subplot(3,2,1)
data['Region'].value_counts(normalize=True).plot.barh()
plt.xlabel('Region', fontdict={'color':'red', 'size':14})
plt.show()
```

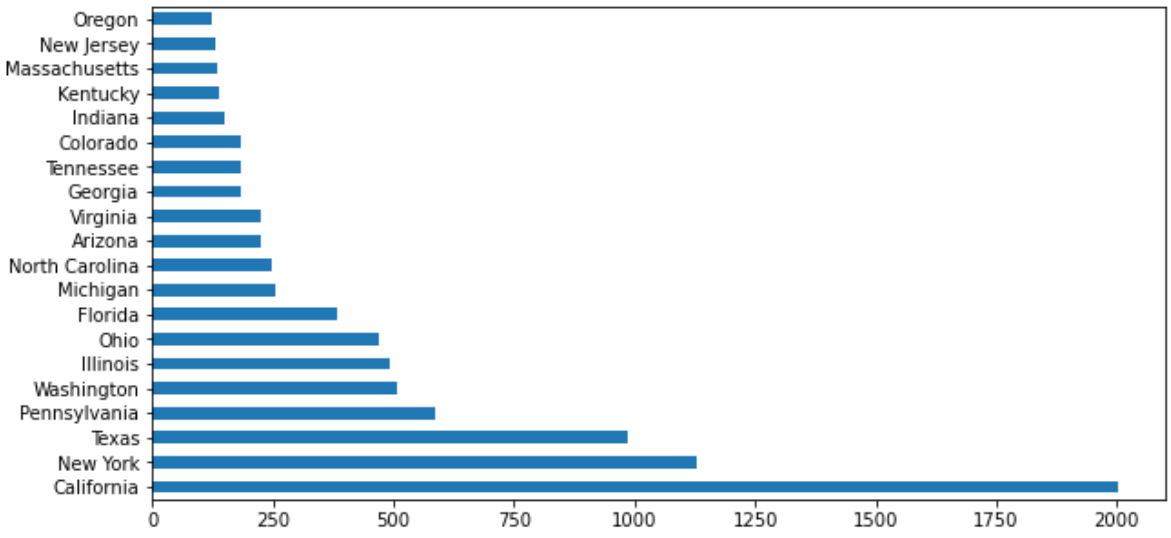


- East & West region is having large distribution.

State analysis

In [18]:

```
data['State'].value_counts()[:20].plot(kind='barh')
plt.show()
```



- Most of the products in this data belongs to california, followed by New York.
- Top 20 states can be seen in the above plot

Category distribution

In [19]:

```
data['Category'].value_counts()[:20]
```

Out[19]:

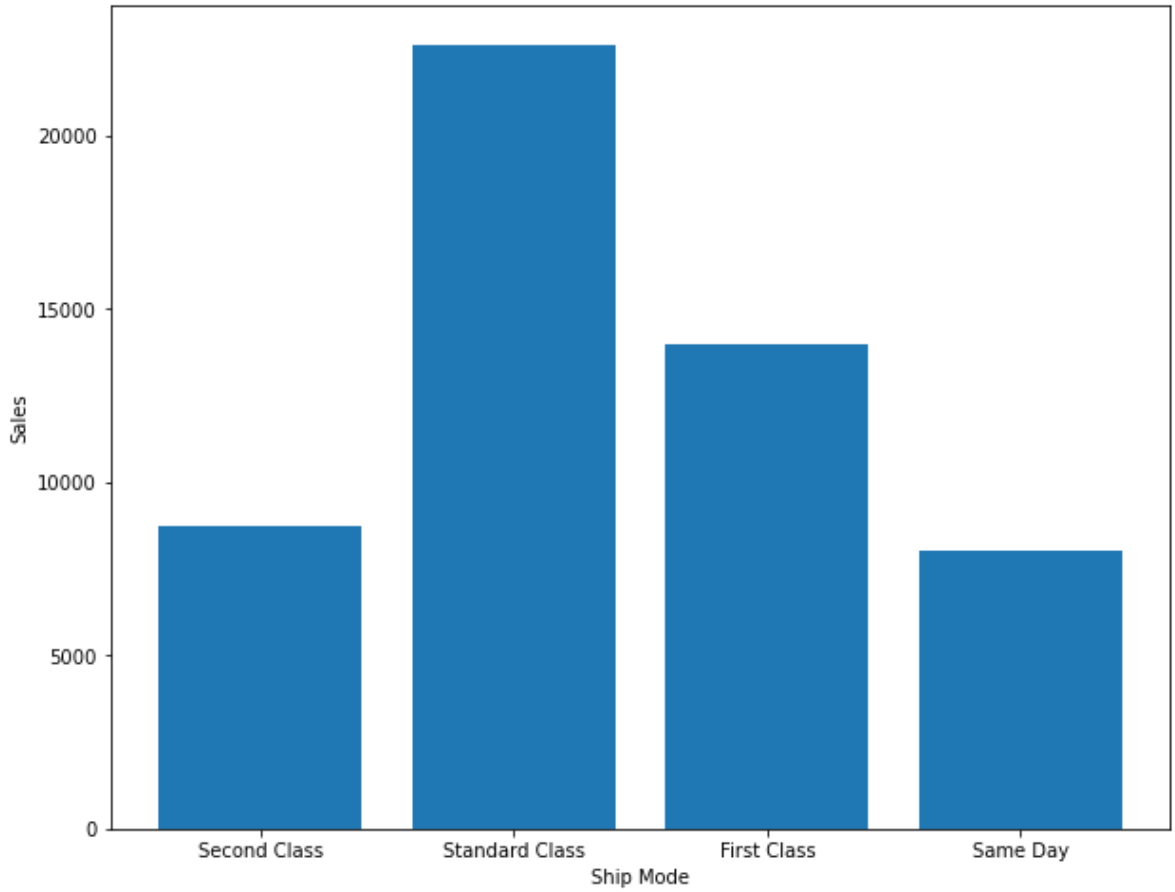
```
Office Supplies    6026
Furniture          2121
Technology         1847
Name: Category, dtype: int64
```

- Here Office Supplies have 6026 Categories, which is the highest.

Ship mode distribution

In [20]:

```
plt.rcParams['figure.figsize']=(10,8)
plt.bar(data['Ship Mode'],data['Sales']);
plt.rcParams.update({'font.size':14});
plt.xlabel('Ship Mode');
plt.ylabel('Sales');
```

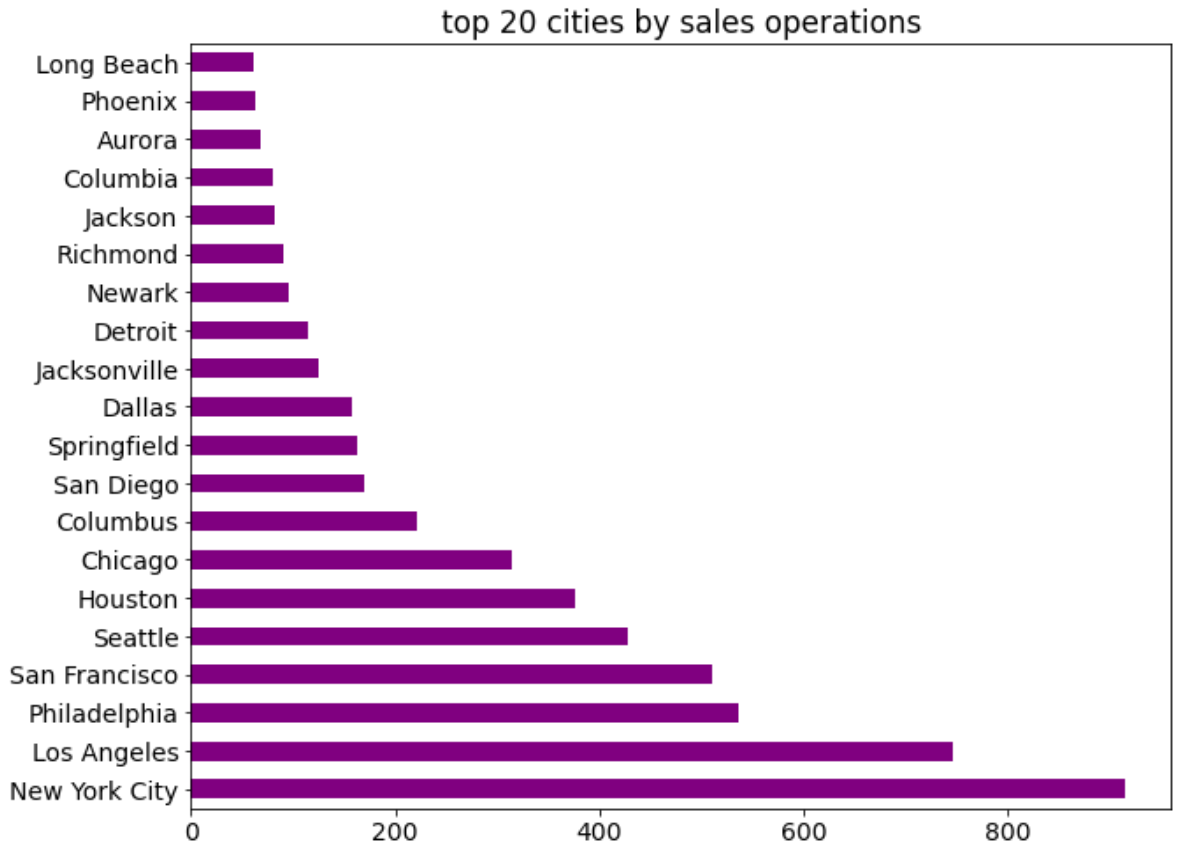


- We can see that, more products belongs to standard class ship mode, followed by rhe First class.

Top 20 cities

In [21]:

```
data.City.value_counts()[:20].plot(kind='barh',color='Purple')
plt.title('top 20 cities by sales operations')
plt.show()
```

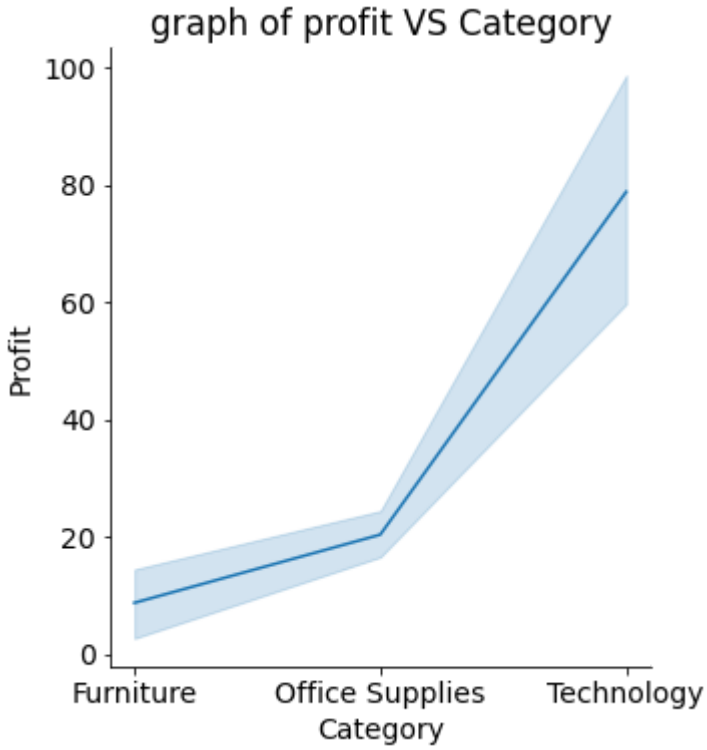


- New York city has most sales operations

Category vs Profit

In [22]:

```
sns.relplot(x="Category",y="Profit",data=data,kind='line')
plt.title('graph of profit VS Category')
plt.show()
```



- Here technology Category get highest Profit

Discount vs Profit

In [23]:

```
plt.figure(figsize=(10,5))
plt.title('Discount vs Profit\n', fontdict={'color':'red','size':15})
sns.lineplot(x='Discount',y='Profit', data=data , color='r')
plt.show()
```

For minimum discount, Profit was good but as discount increases, profit goes down.

this graph shows the negative relationship between discount and profit it means that generally if discount is increasing than profit is decreasing

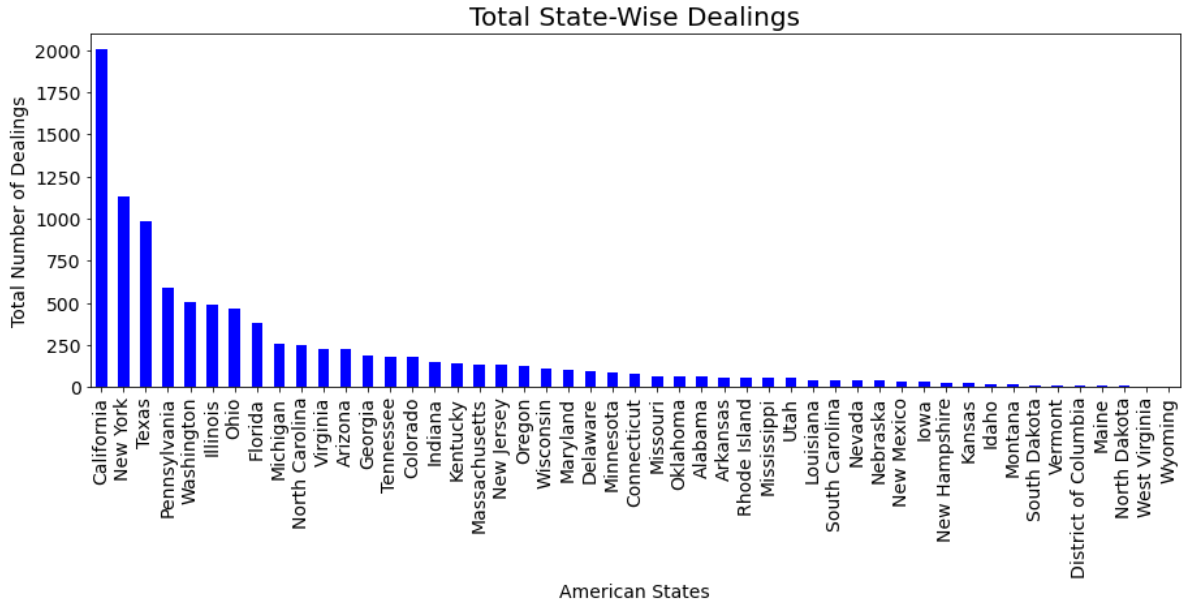
We can observe at 50% discount, there is the lowest profit or highest loss.

Total State-Wise Dealings

In [24]:

```
# total dealings for each State
data_state_dealings = data.groupby('State')['Quantity'].count().sort_values(ascending = Fal

plt.ylabel('Total Number of Dealings')
plt.xlabel('American States')
plt.title('Total State-Wise Dealings', fontsize = 20)
plt.show()
```



California is having higher transaction whereas New York and Texas is moderate one .

Subcategory wise analysis

In [25]:

```
data.groupby(by='Sub-Category').sum().sort_values('Profit',ascending=True).head(10)
```

Out[25]:

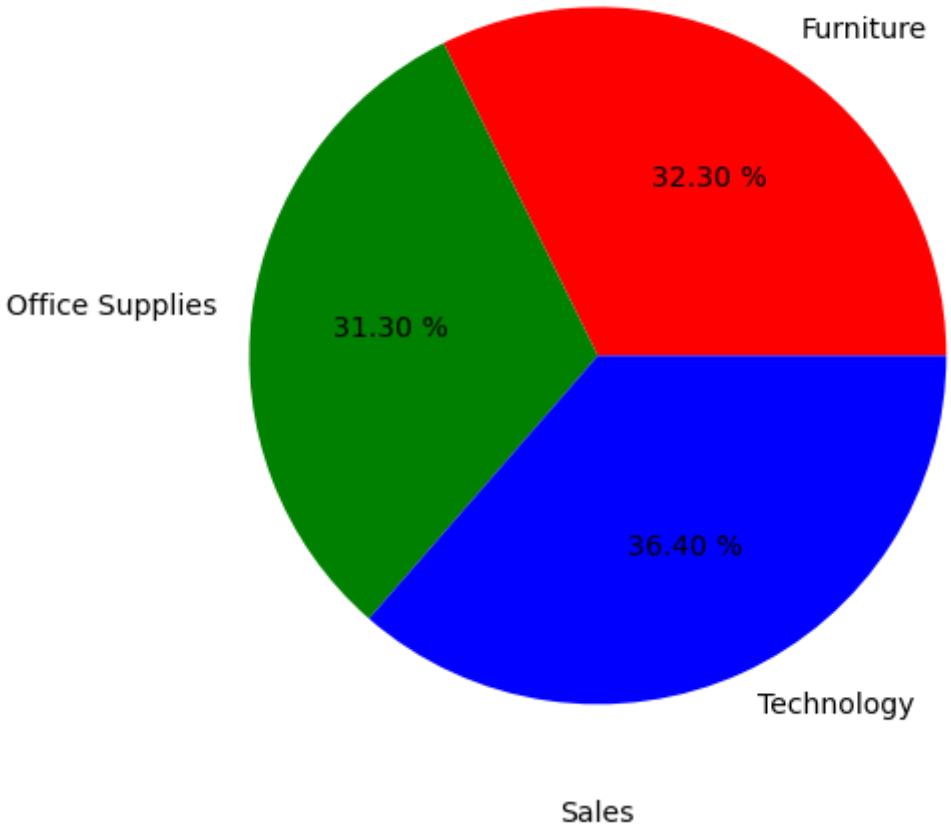
Sub-Category	Postal Code	Sales	Quantity	Discount	Profit
Tables	18607828	206965.5320	1241	83.35	-17725.4811
Bookcases	12771539	114879.9963	868	48.14	-3472.5560
Supplies	10633558	46673.5380	647	14.60	-1189.0995
Fasteners	12506063	3024.2800	914	17.80	949.5182
Machines	6364668	189238.6310	440	35.20	3384.7569
Labels	19552985	12486.3120	1400	25.00	5546.2540
Art	43329658	27118.7920	3000	59.60	6527.7870
Envelopes	13325731	16476.4020	906	20.40	6964.1767
Furnishings	51880430	91705.1640	3563	132.40	13059.1436
Appliances	25250538	107532.1610	1729	77.60	18138.0054

- Tables and bookcases are having major transctions.
- Tables sales found to be loss. Because, after summing up all the profits of tables category we are getting -17725.4811, which is negative.
- Booscases and supplies categories also incurred loss.
- Rest of the products are having profits.

Category wise sales

In [26]:

```
c=data.groupby(by="Category")[["Sales"]].sum()
res=c.reset_index()
plt.pie(x="Sales",labels="Category", data=res, autopct="%.2f %%",colors=["r","g","b"])
plt.xlabel("Sales")
plt.show()
```



Here Technology Sales are 36.40% , Office supplies sales are 31.30% and Furniture Sales are 32.30%

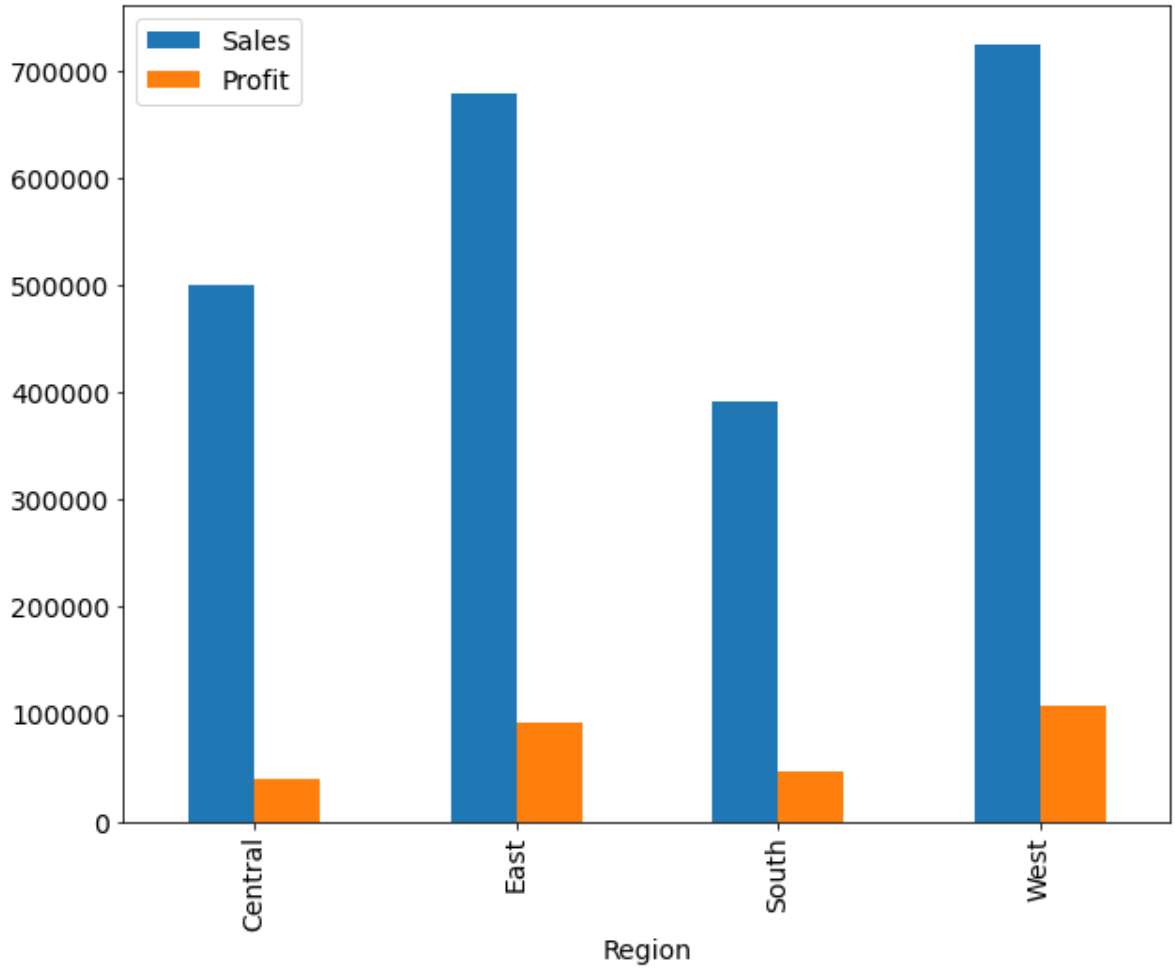
here Technology Category is major sale

In [27]:

```
pd.DataFrame(data.groupby('Region').sum()[['Sales','Profit']]).plot(kind='bar')
```

Out[27]:

```
<AxesSubplot:xlabel='Region'>
```



Western & Eastern regions have shown higher sales and profits as compared to the Southern and central regions.

Central region has higher sales than the Southern region

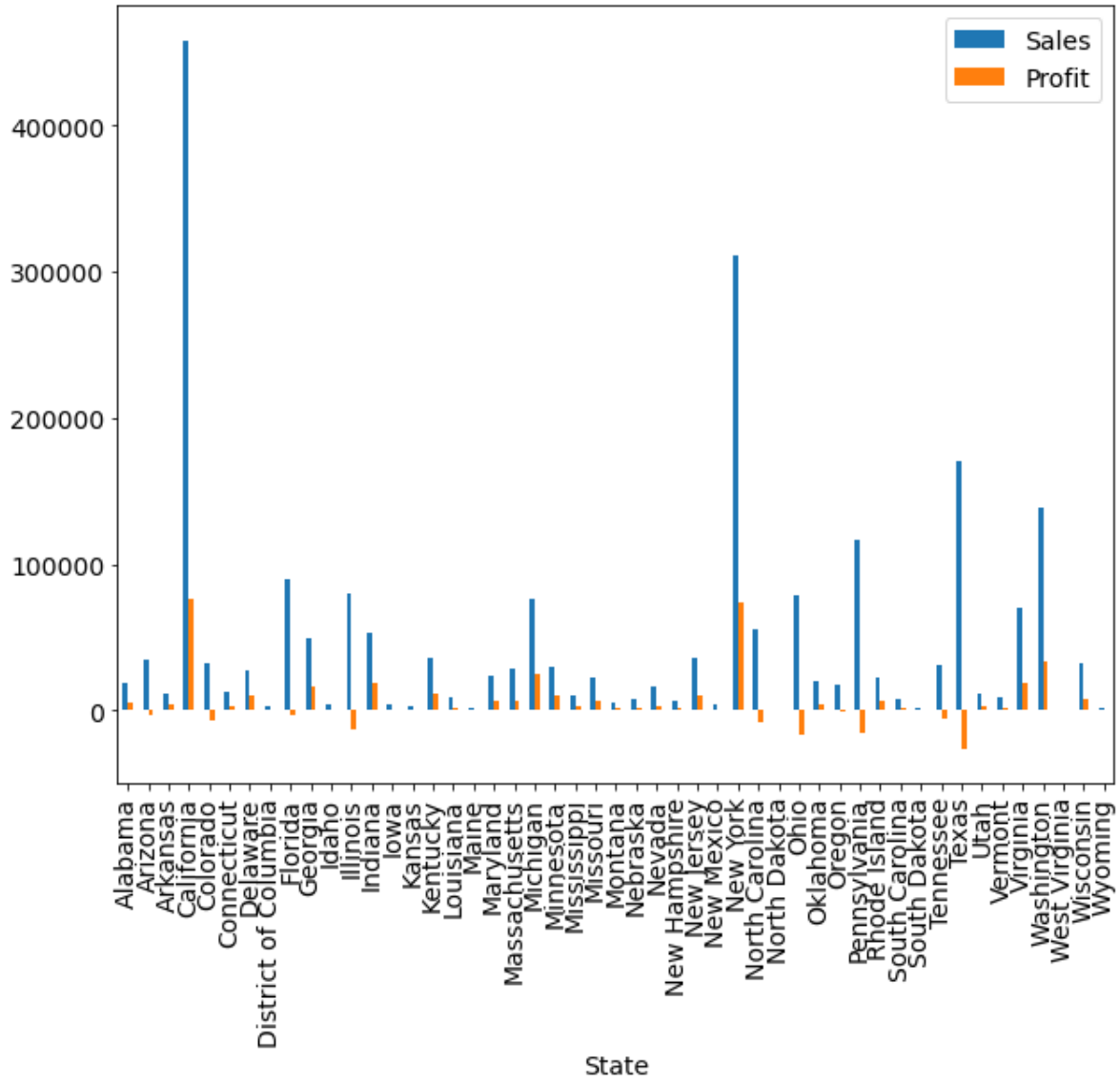
west and east region has highest profit compared to central and south region

In [28]:

```
pd.DataFrame(data.groupby('State').sum()[['Sales','Profit']]).plot(kind='bar')
```

Out[28]:

```
<AxesSubplot:xlabel='State'>
```



The company has highest sales in the state of California

The company has highest profit in the state of California

newyork and The company has lowest sales in the state of wyoming, south dakota, maine.

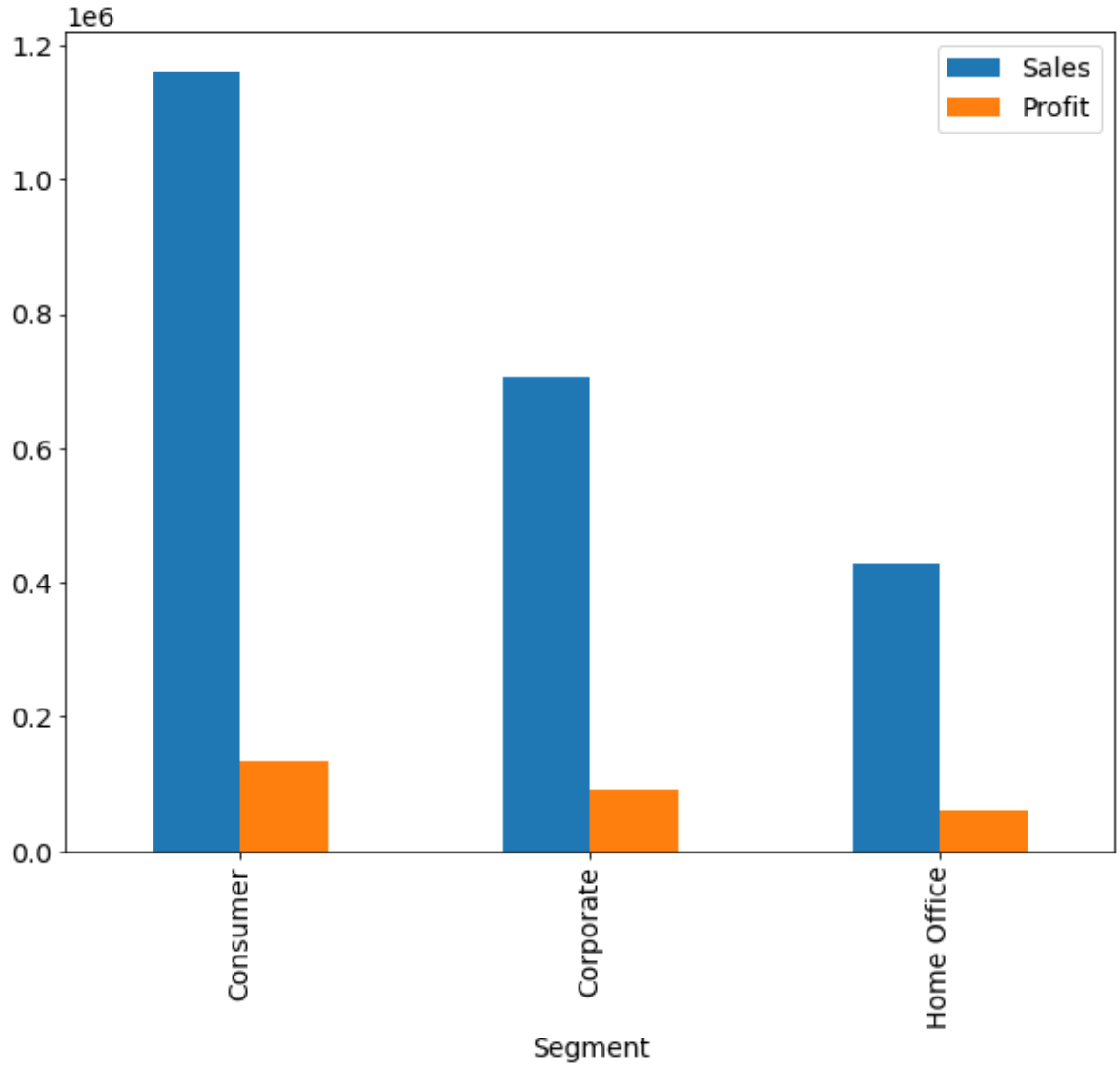
The company has lowest profit in the state of texas,ohio

In [29]:

```
pd.DataFrame(data.groupby('Segment').sum()[['Sales','Profit']]).plot(kind='bar')
```

Out[29]:

```
<AxesSubplot:xlabel='Segment'>
```



Consumer has highest profit and sales

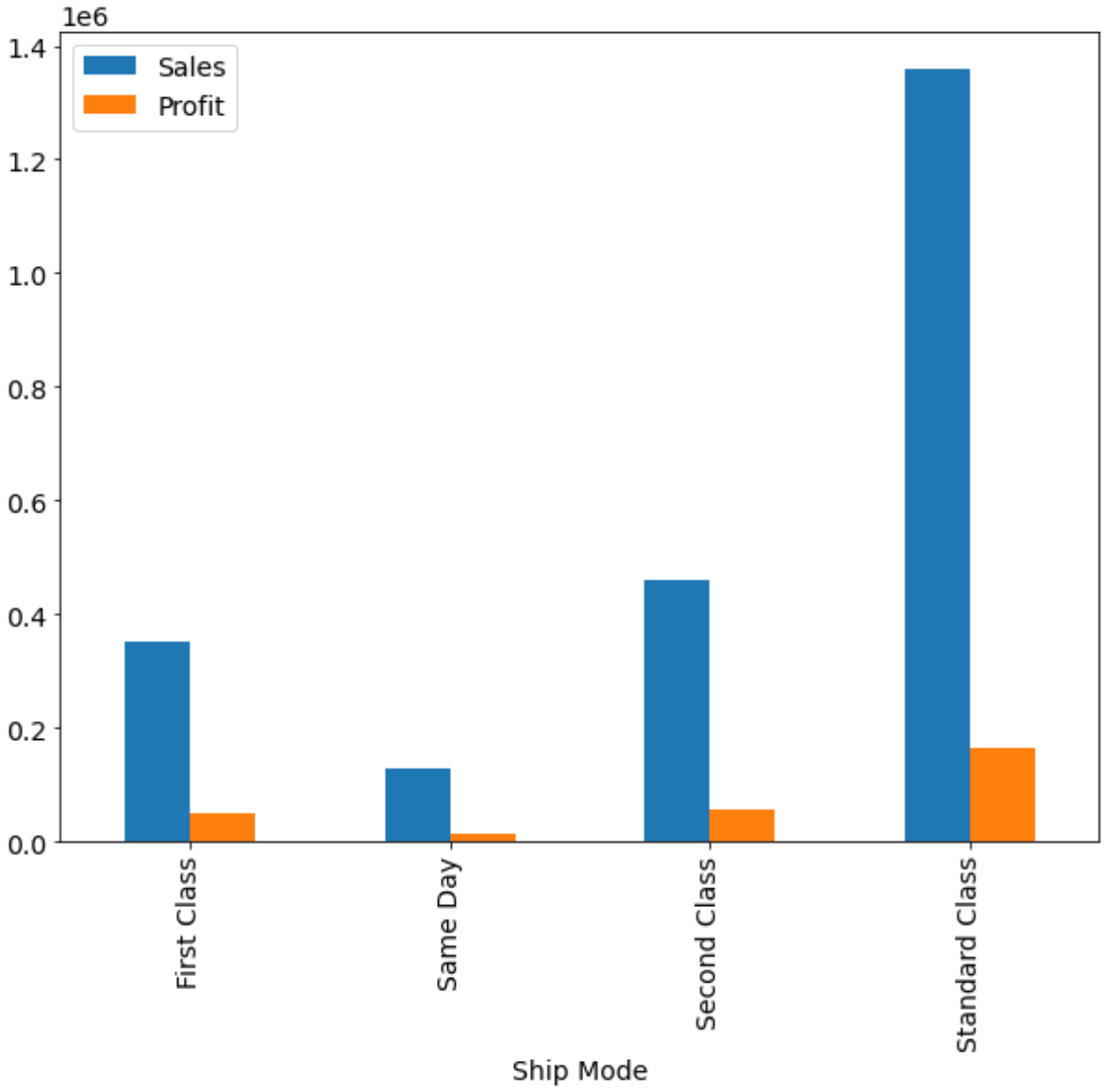
Home office has lowest profit and sales

In [30]:

```
pd.DataFrame(data.groupby('Ship Mode').sum()[['Sales','Profit']]).plot(kind='bar')
```

Out[30]:

```
<AxesSubplot:xlabel='Ship Mode'>
```



standard class has highest profit and sales

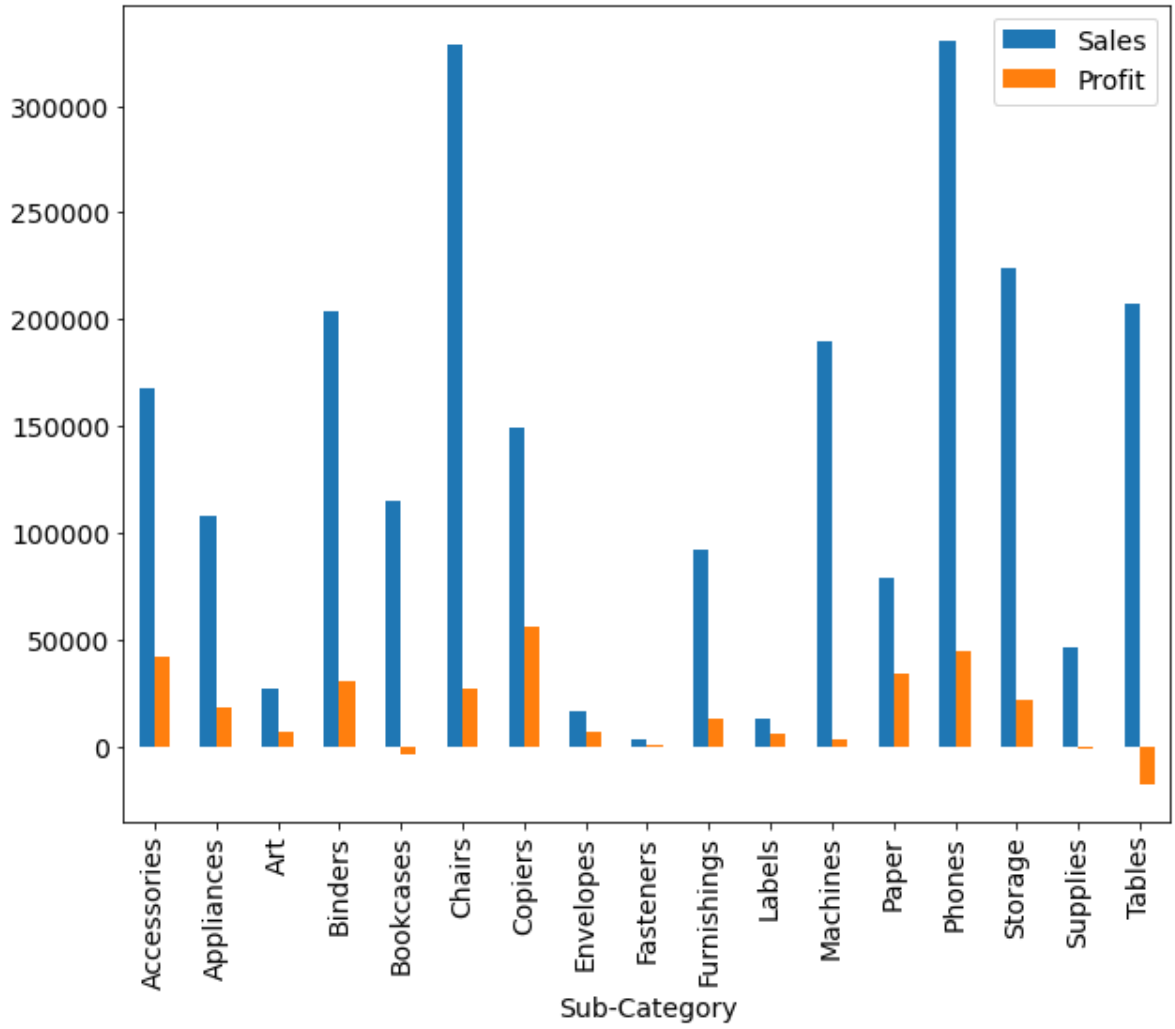
same day has lowest profit and sales

In [31]:

```
pd.DataFrame(data.groupby('Sub-Category').sum()[['Sales','Profit']]).plot(kind='bar')
```

Out[31]:

```
<AxesSubplot:xlabel='Sub-Category'>
```



phones and chairs has high sales

fastners and labels has low sales

tables and book cases has low profit

copiers and phones has heigh profit

Conclusion:

- The main reason for loss is Discount. some areas lead to loss due to more discounts, and some areas lead to fewer sales due to fewer discounts, hence it needs to be improved.
- The Home office segment needs better improvement.

- Some cities have fewer sales, so it needs better improvement
- As Sales increase profit is increase. Hence increase rate of Sales.
- Majority is the profit took place in Standard Shipment mode. Hence Same day and Second Class mode needs improvement.

In [ ]: