Name: Ojeswi Subhash Ambekar

Task: Exploratory Data Analysis - Retail

In this task we will perform "Exploratory Data Analysis" on dataset "SampleSuperstore".As a business manager, we will try to find out the weak areas where we can work to make more profit. Also What other business problems we can derive by exploring the data.

In [69]:

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from plotnine import *
import warnings
warnings.filterwarnings('ignore')
```

In [70]:

```
#Reading the dataset

data=pd.read_csv("D:\SampleSuperstore.csv")
```

In [71]:

```
#to print first 5 rows of dataset
data.head()
```

Out[71]:

	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 [72]:

```
#to print last 5 rows of dataset
data.tail()
```

Out[72]:

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Su Categ
9989	Second Class	Consumer	United States	Miami	Florida	33180	South	Furniture	Furnishin
9990	Standard Class	Consumer	United States	Costa Mesa	California	92627	West	Furniture	Furnishin
9991	Standard Class	Consumer	United States	Costa Mesa	California	92627	West	Technology	Phon
9992	Standard Class	Consumer	United States	Costa Mesa	California	92627	West	Office Supplies	Pap
9993	Second Class	Consumer	United States	Westminster	California	92683	West	Office Supplies	Applianc

In [73]:

```
data.shape
```

Out[73]:

```
(9994, 13)
```

In [74]:

```
#to print full summary of dataset
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
```

In [75]:

```
#to print statistical data
data.describe()
```

Out[75]:

	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

In [76]:

```
#to print missing values
data.isnull()
```

Out[76]:

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Sub-Category	Sales	Quar
0	False	False	False	False	False	False	False	False	False	False	F
1	False	False	False	False	False	False	False	False	False	False	F
2	False	False	False	False	False	False	False	False	False	False	F
3	False	False	False	False	False	False	False	False	False	False	F
4	False	False	False	False	False	False	False	False	False	False	F
...	...	...	...	...	...	...	...	...	...	...	
9989	False	False	False	False	False	False	False	False	False	False	F
9990	False	False	False	False	False	False	False	False	False	False	F
9991	False	False	False	False	False	False	False	False	False	False	F
9992	False	False	False	False	False	False	False	False	False	False	F
9993	False	False	False	False	False	False	False	False	False	False	F

9994 rows × 13 columns

In [77]:

```
data.isnull().sum()
```

Out[77]:

```
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
```

In [78]:

```
#checking duplicate data
data.duplicated().sum()
```

Out[78]:

```
17
```

In [79]:

```
#droping duplicate data
data.drop_duplicates()
```

Out[79]:

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Su Categ
0	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Bookcas
1	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Cha
2	Second Class	Corporate	United States	Los Angeles	California	90036	West	Office Supplies	Lab
3	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Furniture	Tab
4	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Office Supplies	Stora
...	...	...	...	...	...	...	...	...	
9989	Second Class	Consumer	United States	Miami	Florida	33180	South	Furniture	Furnishin
9990	Standard Class	Consumer	United States	Costa Mesa	California	92627	West	Furniture	Furnishin
9991	Standard Class	Consumer	United States	Costa Mesa	California	92627	West	Technology	Phon
9992	Standard Class	Consumer	United States	Costa Mesa	California	92627	West	Office Supplies	Pap
9993	Second Class	Consumer	United States	Westminster	California	92683	West	Office Supplies	Applianc

9977 rows × 13 columns

In [80]:

```
data.nunique()
```

Out[80]:

```
Ship Mode          4
Segment            3
Country            1
City             531
State             49
Postal Code      631
Region             4
Category           3
Sub-Category      17
Sales           5825
Quantity          14
Discount          12
Profit          7287
dtype: int64
```

In [81]:

```
column=['Postal Code']
data1=data.drop(columns=column,axis=1)
```

In [82]:

```
#checking correlation between different variables
data1.corr()
```

Out[82]:

	Sales	Quantity	Discount	Profit
Sales	1.000000	0.200795	-0.028190	0.479064
Quantity	0.200795	1.000000	0.008623	0.066253
Discount	-0.028190	0.008623	1.000000	-0.219487
Profit	0.479064	0.066253	-0.219487	1.000000

In [83]:

```
data1.head()
```

Out[83]:

	Ship Mode	Segment	Country	City	State	Region	Category	Sub-Category	Sales
0	Second Class	Consumer	United States	Henderson	Kentucky	South	Furniture	Bookcases	261.9600
1	Second Class	Consumer	United States	Henderson	Kentucky	South	Furniture	Chairs	731.9400
2	Second Class	Corporate	United States	Los Angeles	California	West	Office Supplies	Labels	14.6200
3	Standard Class	Consumer	United States	Fort Lauderdale	Florida	South	Furniture	Tables	957.5775
4	Standard Class	Consumer	United States	Fort Lauderdale	Florida	South	Office Supplies	Storage	22.3680

Data Visualization

In [92]:

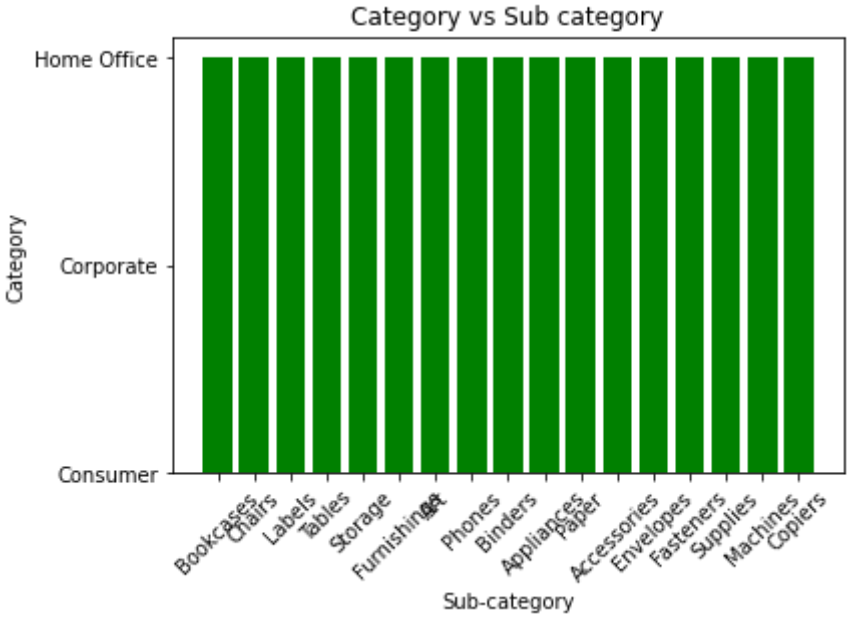
```
df = pd.DataFrame(data)
```

In [110]:

```
X = list(df.iloc[:, 8])
Y = list(df.iloc[:,1])
```

In [111]:

```
plt.bar(X, Y, color='g')
plt.title('Category vs Sub category')
plt.xlabel("Sub-category")
plt.ylabel("Category")
plt.xticks(rotation=45)
plt.show()
```



In [112]:

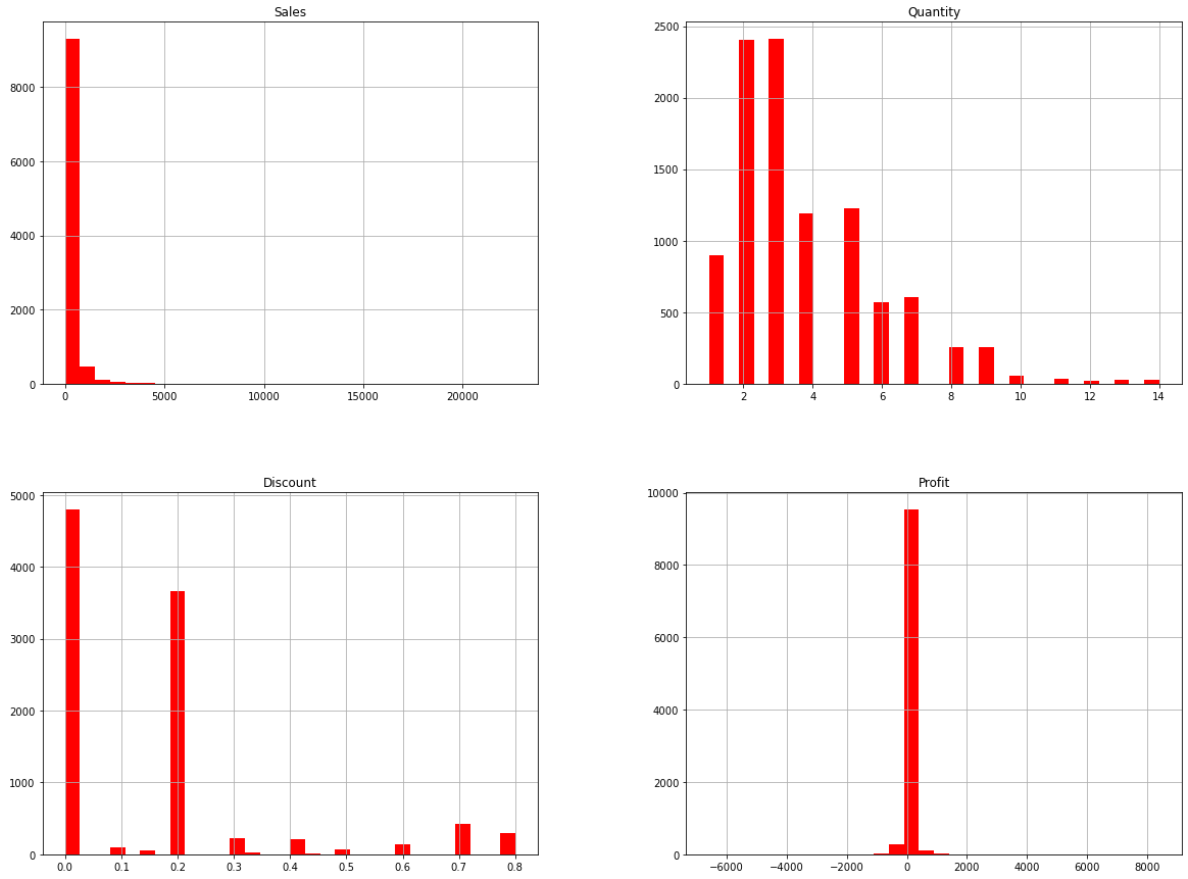
```
data1.corr()
```

Out[112]:

	Sales	Quantity	Discount	Profit
Sales	1.000000	0.200795	-0.028190	0.479064
Quantity	0.200795	1.000000	0.008623	0.066253
Discount	-0.028190	0.008623	1.000000	-0.219487
Profit	0.479064	0.066253	-0.219487	1.000000

In [114]:

```
data1.hist(bins=30,figsize=(20,15),color='red')
plt.show()
```



In [115]:

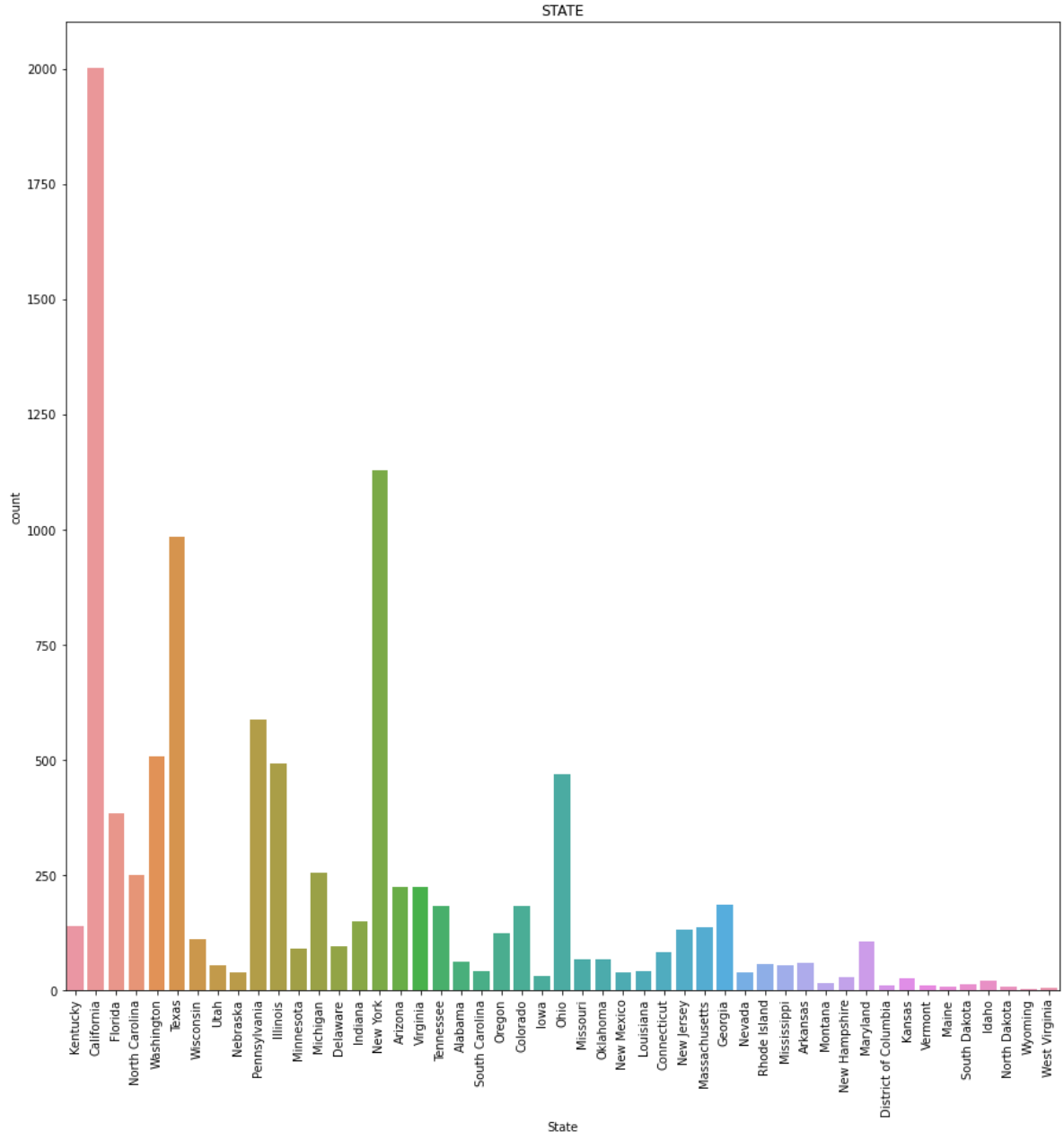
```
#printing total no. of states which are repeating
data1["State"].value_counts()
```

Out[115]:

```
California              2001
New York                1128
Texas                    985
Pennsylvania             587
Washington               506
Illinois                 492
Ohio                     469
Florida                  383
Michigan                 255
North Carolina           249
Arizona                  224
Virginia                 224
Georgia                  184
Tennessee                183
Colorado                 182
Indiana                  149
Kentucky                 139
Massachusetts            135
New Jersey               130
Oregon                   124
Wisconsin                110
Maryland                 105
Delaware                  96
Minnesota                 89
Connecticut               82
Oklahoma                  66
Missouri                  66
Alabama                   61
Arkansas                  60
Rhode Island              56
Utah                      53
Mississippi               53
Louisiana                 42
South Carolina            42
Nevada                    39
Nebraska                  38
New Mexico                37
Iowa                      30
New Hampshire             27
Kansas                    24
Idaho                     21
Montana                   15
South Dakota              12
Vermont                   11
District of Columbia      10
Maine                      8
North Dakota               7
West Virginia              4
Wyoming                    1
Name: State, dtype: int64
```

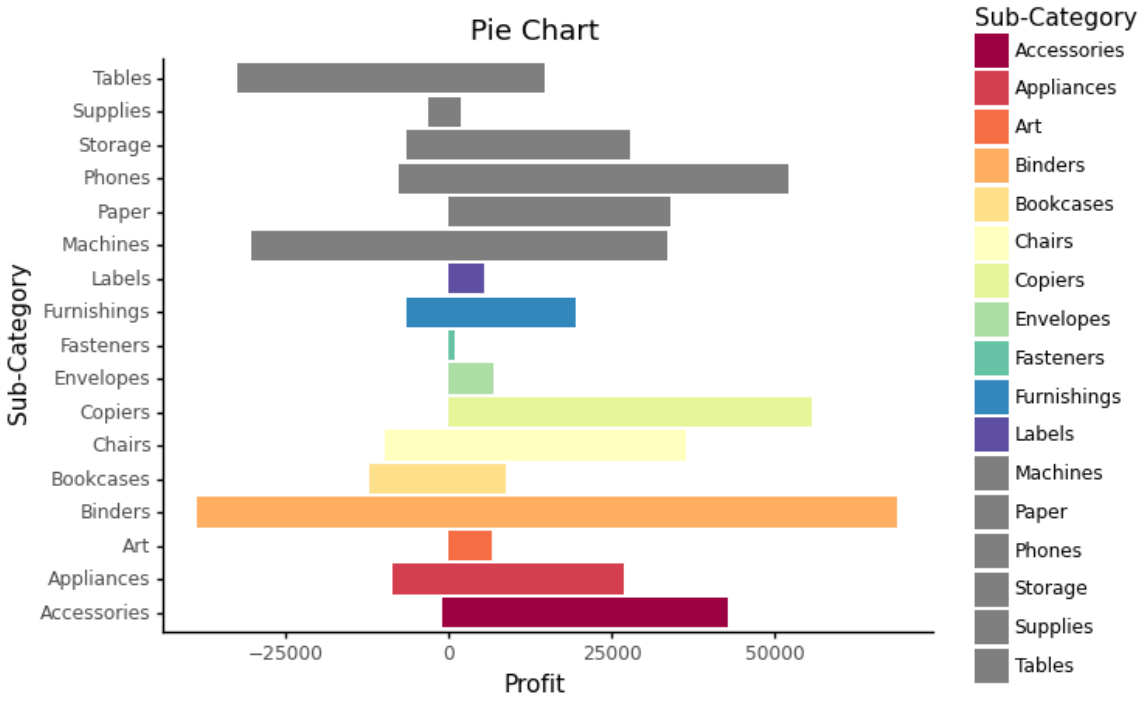
In [116]:

```
plt.figure(figsize=(15,15))
sns.countplot(x=data1['State'])
plt.xticks(rotation=90)
plt.title("STATE")
plt.show()
```



In [117]:

```
profit_plot=(ggplot(data, aes(x='Sub-Category', y='Profit',fill='Sub-Category')) + geom_col
             scale_fill_brewer(type='div', palette='Spectral') + theme_classic() + ggtitle(
display(profit_plot)
```

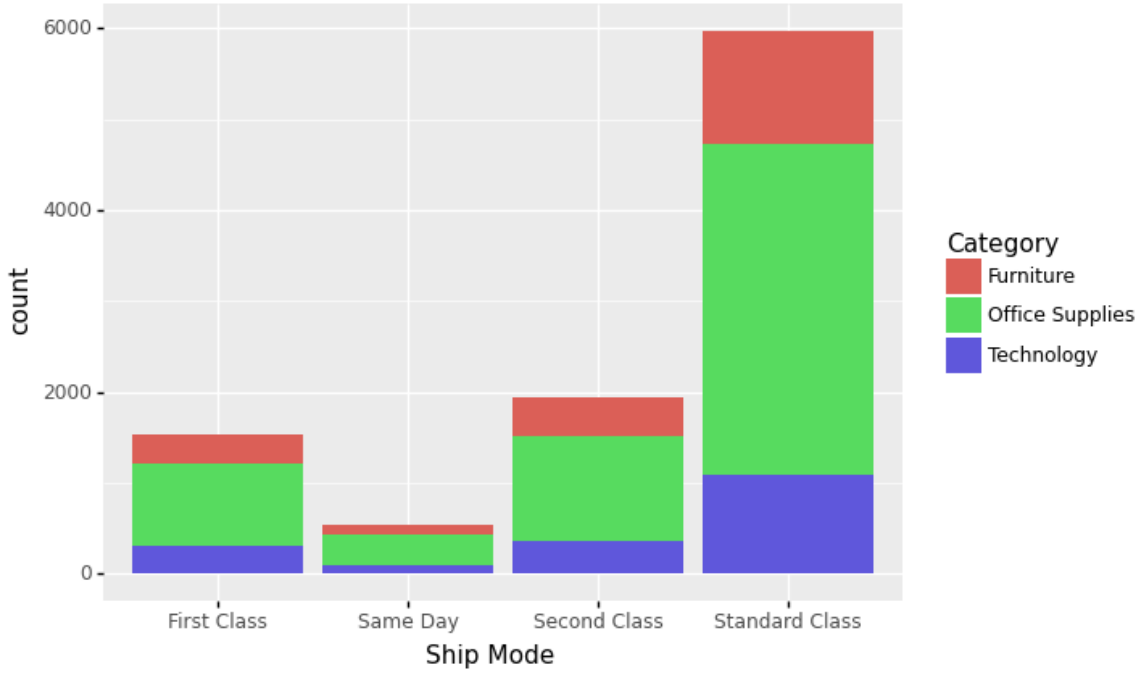


<ggplot: (142875796152)>

The above pie chart shows the profit and loss of each subcategories.
1.Sub category named "Copies" has gained highest profit among other subcategories with zero loss.Also "Accessories" subcategory has more profit with minimum loss.
2.Sub-category "Binders" has gained equal amount of loss and profit.

In [120]:

```
ggplot(data,aes(x='Ship Mode', fill='Category'))+ geom_bar(stat='count')
```



Out[120]:

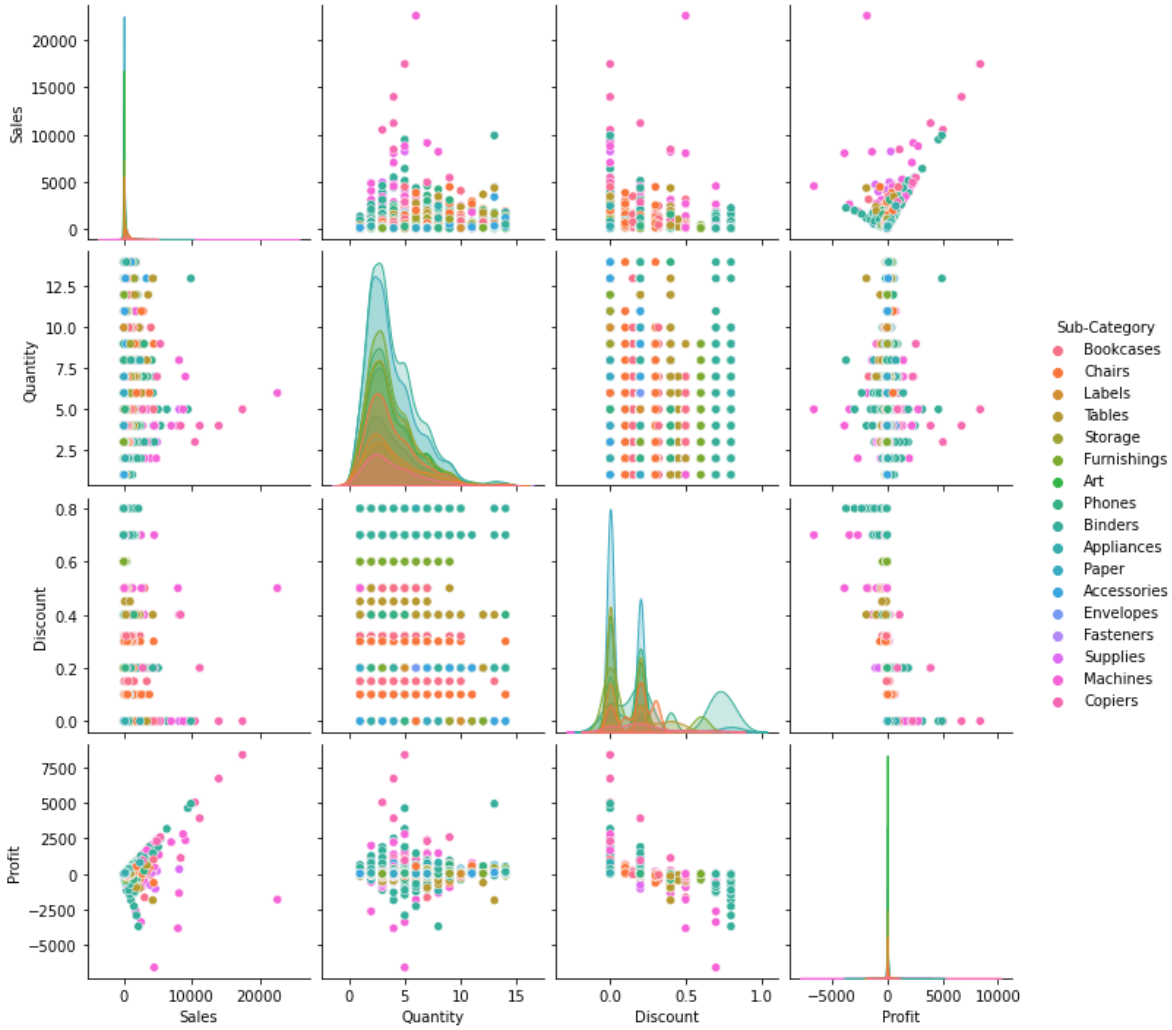
```
<ggplot: (142876757185)>
```

In [122]:

```
figsize=(14,8)
sns.pairplot(data1,hue='Sub-Category')
plt.show
```

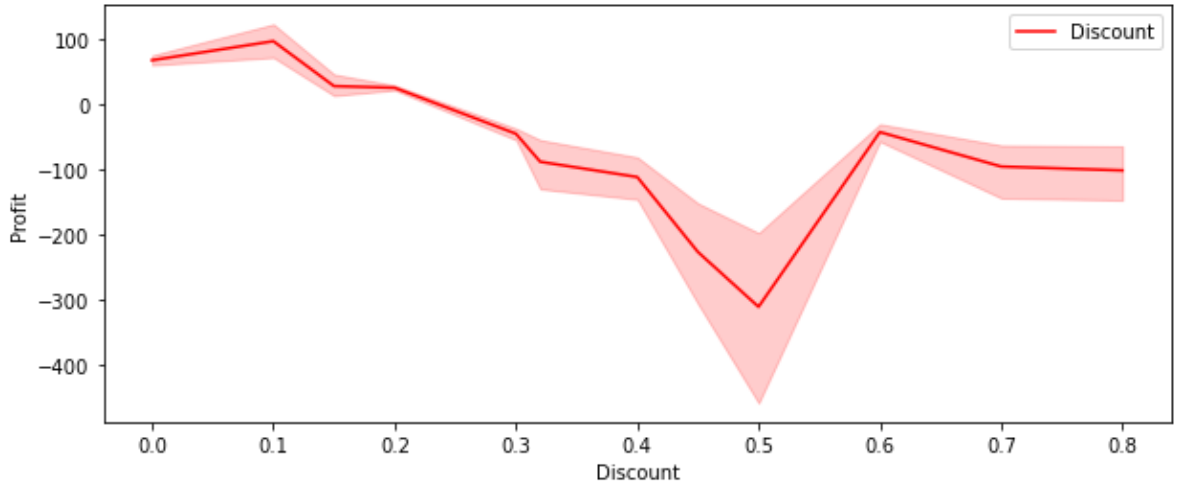
Out[122]:

```
<function matplotlib.pyplot.show(close=None, block=None)>
```



In [123]:

```
plt.figure(figsize=(10,4))
sns.lineplot("Discount","Profit", data=data1, color='r',label="Discount")
plt.legend()
plt.show()
```



In [138]:

```
data1=df[:]
```

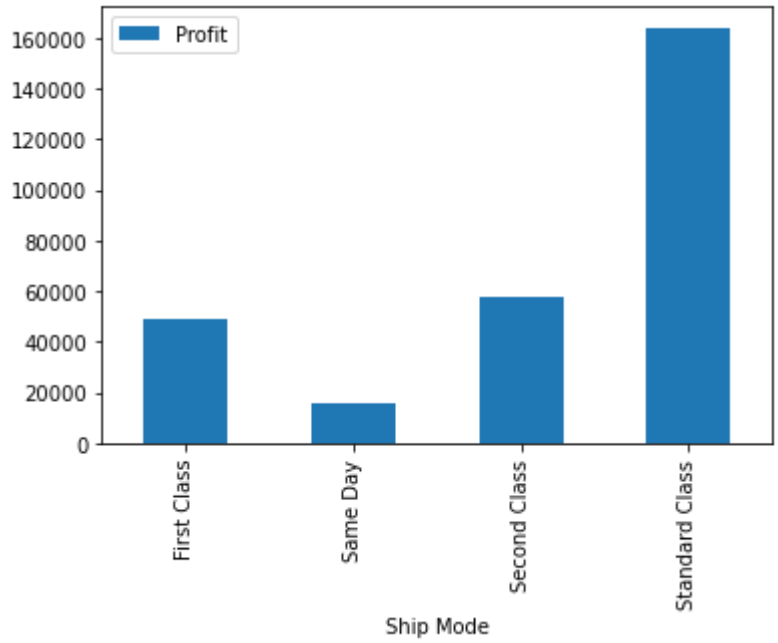
In [148]:

```
plot=pd.DataFrame(data1.groupby(['Ship Mode'])["Profit"].sum()).reset_index()
plot1=pd.DataFrame(data1.groupby(['Segment'])["Profit"].sum()).reset_index()
plot2=pd.DataFrame(data1.groupby(['Category'])["Profit"].sum()).reset_index()
plot3=pd.DataFrame(data1.groupby(['Discount'])["Profit"].sum()).reset_index()
```

Which Ship mode brings the Highest Profit?

In [141]:

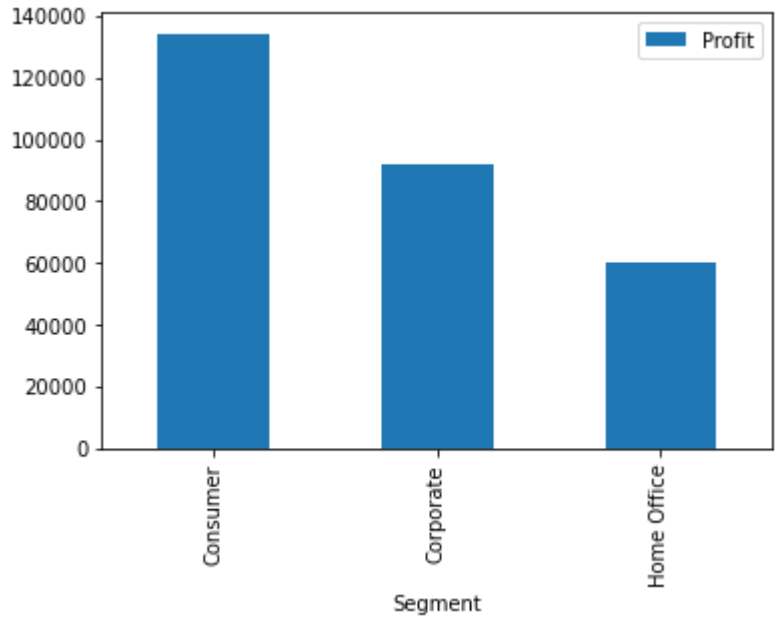
```
ax=plot.plot(kind='bar',x="Ship Mode",y='Profit')
```



Which segment brings the highest Profit?

In [144]:

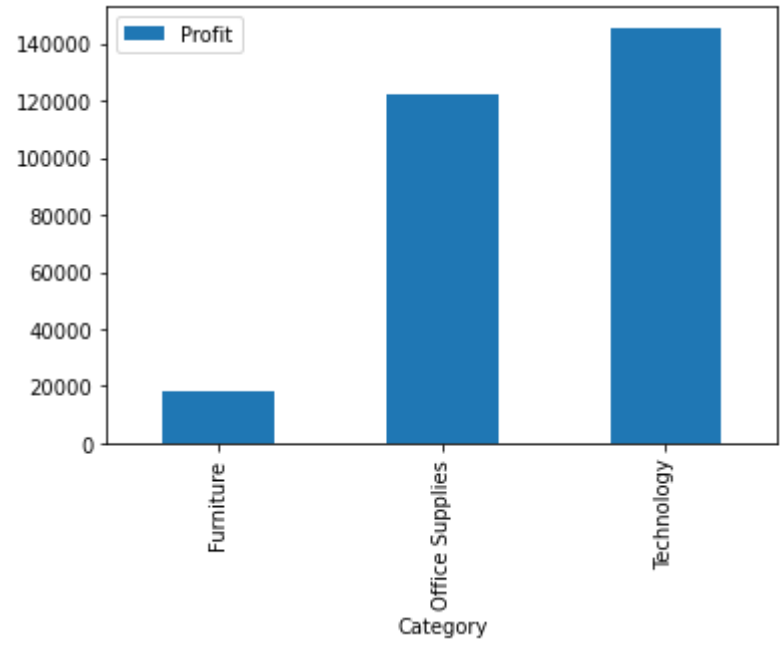
```
ax=plot1.plot(kind='bar',x='Segment',y="Profit")
```



Which Category brings the highest profit?

In [146]:

```
ax=plot2.plot(kind='bar',x='Category',y='Profit')
```



In [147]:

```
#Profit Analysis
pd.DataFrame(data1["Profit"]).describe()
```

Out[147]:

	Profit
count	9994.000000
mean	28.656896
std	234.260108
min	-6599.978000
25%	1.728750
50%	8.666500
75%	29.364000
max	8399.976000

Top 10 cities with maximum profit

In [149]:

```
city=data1[["City","Profit"]]
plot3=pd.DataFrame(city.groupby(["City"])["Profit"].aggregate("sum").reset_index().sort_val
plot3.head(10)
```

Out[149]:

	City	Profit
329	New York City	62036.9837
266	Los Angeles	30440.7579
452	Seattle	29156.0967
438	San Francisco	17507.3854
123	Detroit	13181.7908
233	Lafayette	10018.3876
215	Jackson	7581.6828
21	Atlanta	6993.6629
300	Minneapolis	6824.5846
437	San Diego	6377.1960

In [150]:

```
plot3.tail(10)
```

Out[150]:

	City	Profit
216	Jacksonville	-2323.8350
24	Aurora	-2691.7386
375	Phoenix	-2790.8832
109	Dallas	-2846.5257
60	Burlington	-3622.8772
80	Chicago	-6654.5688
241	Lancaster	-7239.0684
434	San Antonio	-7299.0502
207	Houston	-10153.5485
374	Philadelphia	-13837.7674

```
From the above data visualization,we can see the states and category where profits can be
high or low.We can improve in those states by providing discounts in prefered range so
that the company and consumer will both be in profit.We have found the top 10 highest as
well as lowest profit cities. In lowest cities we have to improve in strategies to
produce more profit.
```