In [5]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
#libraries were imported
```

In [6]:
```
Superstore=pd.read_csv('C:\\Users\\prath\\Superstore_sales.csv')
#data was read and stored in variable superstore as dataframe
```

In [7]:
```
#to get the information about the data
Superstore.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9977 entries, 0 to 9976
Data columns (total 13 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   Ship Mode     9977 non-null   object 
 1   Segment       9977 non-null   object 
 2   Country       9977 non-null   object 
 3   City          9977 non-null   object 
 4   State         9977 non-null   object 
 5   Postal Code   9977 non-null   int64  
 6   Region        9977 non-null   object 
 7   Category      9977 non-null   object 
 8   Sub-Category  9977 non-null   object 
 9   Sales         9977 non-null   float64
 10  Quantity      9977 non-null   int64  
 11  Discount      9977 non-null   float64
 12  Profit        9977 non-null   float64
dtypes: float64(3), int64(2), object(8)
memory usage: 1013.4+ KB
```

In [8]:
```
Superstore.isnull().sum()
```

Out[8]:
```
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 [9]:
```
Superstore.duplicated().sum()
```

Out[9]:
```
0
```

The data was cleaned using spreadsheet which included removing duplicates and missing values

In [10]:
```
Superstore.describe()
```

Out[10]:

	Postal Code	Sales	Quantity	Discount	Profit
count	9977.000000	9977.000000	9977.000000	9977.000000	9977.00000
mean	55154.964117	230.148902	3.790719	0.156278	28.69013
std	32058.266816	623.721409	2.226657	0.206455	234.45784
min	1040.000000	0.444000	1.000000	0.000000	-6599.97800
25%	23223.000000	17.300000	2.000000	0.000000	1.72620
50%	55901.000000	54.816000	3.000000	0.200000	8.67100
75%	90008.000000	209.970000	5.000000	0.200000	29.37200
max	99301.000000	22638.480000	14.000000	0.800000	8399.97600

In [11]:
```
Superstore['Postal Code']=Superstore['Postal Code'].astype(object)
```

In [12]:
```
Superstore.describe()
```

Out[12]:

	Sales	Quantity	Discount	Profit
count	9977.000000	9977.000000	9977.000000	9977.00000
mean	230.148902	3.790719	0.156278	28.69013
std	623.721409	2.226657	0.206455	234.45784
min	0.444000	1.000000	0.000000	-6599.97800
25%	17.300000	2.000000	0.000000	1.72620
50%	54.816000	3.000000	0.200000	8.67100
75%	209.970000	5.000000	0.200000	29.37200
max	22638.480000	14.000000	0.800000	8399.97600

In [13]:
```
Superstore.corr()
#to get the correlation of between different parameters of the data
```

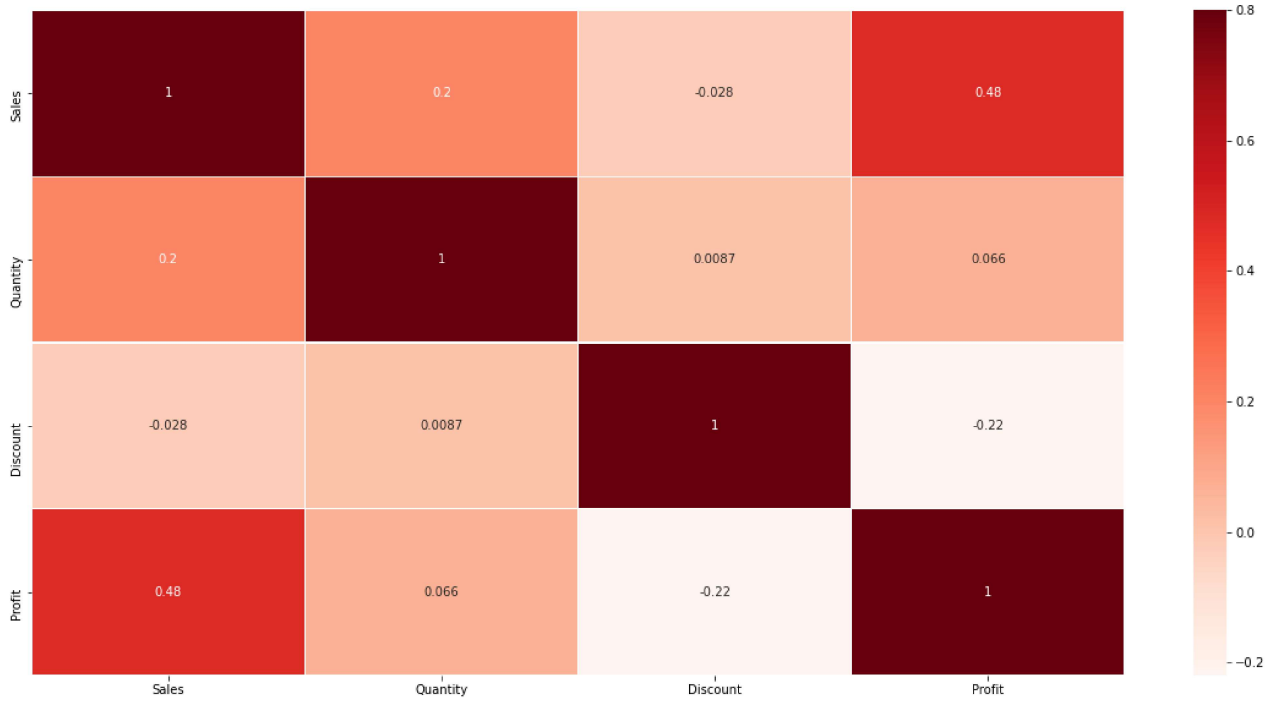
Out[13]:

	Sales	Quantity	Discount	Profit
Sales	1.000000	0.200722	-0.028311	0.479067
Quantity	0.200722	1.000000	0.008678	0.066211
Discount	-0.028311	0.008678	1.000000	-0.219662
Profit	0.479067	0.066211	-0.219662	1.000000

In [14]:
```
corrmat=Superstore.corr()
f,ax=plt.subplots(figsize=(20,10))
colormap=plt.cm.Reds
```

```
sns.heatmap(corrmat,linewidth=0.2,cmap=colormap,linecolor='white',vmax=0.8,annot=True)
#To visulaize the above correlations for better interpretation
```

Out[14]: <AxesSubplot:>



The interpretations are

1. The profit increases with increase in sales and the discount has negative impact on profits
2. Need for other strategy to increase profit as discount and sales are also negativel related

In [15]:
```
Superstore['Category'].unique()
```

Out[15]: array(['Furniture', 'Office Supplies', 'Technology'], dtype=object)

In [16]:
```
Superstore['Ship Mode'].unique()
```

Out[16]:
```
array(['Second Class', 'Standard Class', 'First Class', 'Same Day'],
      dtype=object)
```

In [17]:
```
Superstore['Region'].unique()
```

Out[17]: array(['South', 'West', 'Central', 'East'], dtype=object)

In [18]:
```
Superstore['Country'].unique()
```

Out[18]: array(['United States'], dtype=object)

In [19]:
```
Superstore['Segment'].unique()
```

Out[19]: array(['Consumer', 'Corporate', 'Home Office'], dtype=object)

In [20]:
```
Superstore['Sub-Category'].unique()
```

Out[20]:
```
array(['Bookcases', 'Chairs', 'Labels', 'Tables', 'Storage',
       'Furnishings', 'Art', 'Phones', 'Binders', 'Appliances', 'Paper',
       'Accessories', 'Envelopes', 'Fasteners', 'Supplies', 'Machines',
       'Copiers'], dtype=object)
```

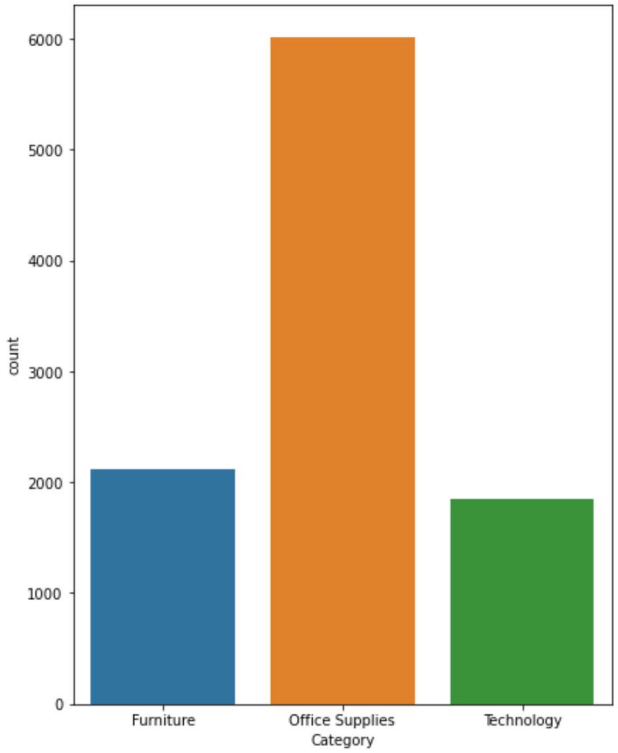
In [120…
```
Cat_Subcat=Superstore.groupby(['Category','Sub-Category'])[['Sales','Profit']].sum()[['
Cat_Subcat
#To know the distribution of the subcategories of product in different categories
```

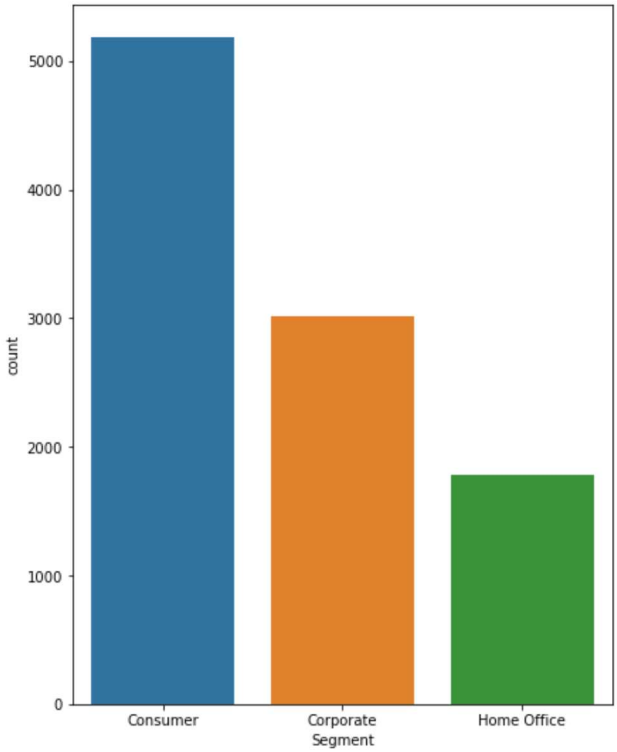
Out[120…

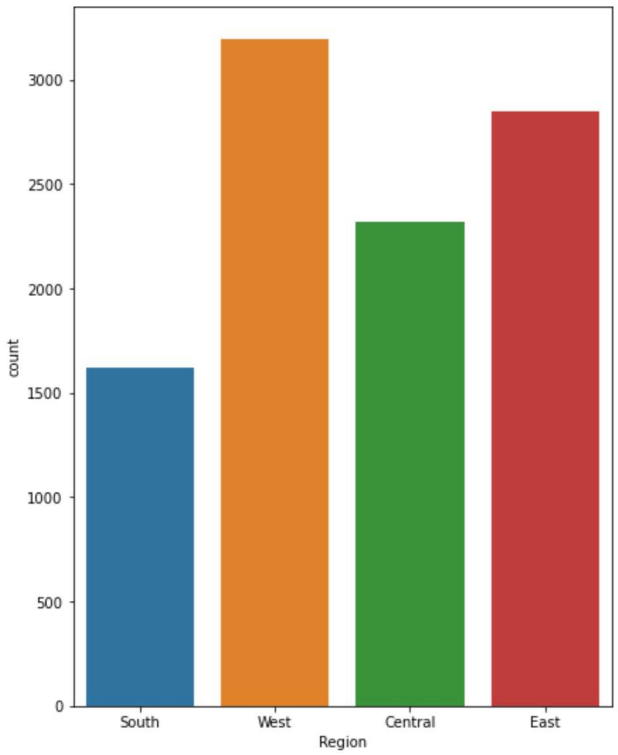
Category	Sub-Category	Sales	Profit
Furniture	Bookcases	114879.9963	-3472.5560
	Chairs	327777.7610	26567.1278
	Furnishings	91683.0240	13052.7230
	Tables	206965.5320	-17725.4811
Office Supplies	Appliances	107532.1610	18138.0054
	Art	27107.0320	6524.6118
	Binders	203409.1690	30228.0003
	Envelopes	16476.4020	6964.1767
	Fasteners	3024.2800	949.5182
	Labels	12444.9120	5526.3820
	Paper	78224.1420	33944.2395
	Storage	223843.6080	21278.8264
	Supplies	46673.5380	-1189.0995
Technology	Accessories	167380.3180	41936.6357
	Copiers	149528.0300	55617.8249
	Machines	189238.6310	3384.7569
	Phones	330007.0540	44515.7306

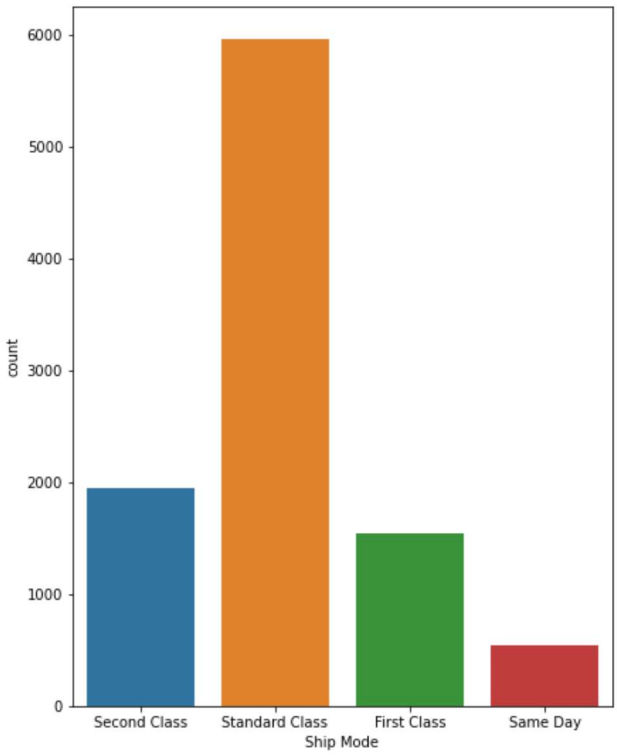
In [22]:
```
fig,ax1 =plt.subplots(nrows=2,ncols=2,figsize=(15,20))
sns.countplot(x=Superstore['Category'],data=Superstore,ax=ax1[0,0])
sns.countplot(x=Superstore['Segment'],data=Superstore,ax=ax1[0,1])
sns.countplot(x=Superstore['Region'],data=Superstore,ax=ax1[1,0])
sns.countplot(x=Superstore['Ship Mode'],data=Superstore,ax=ax1[1,1])
```

Out[22]:
```
<AxesSubplot:xlabel='Ship Mode', ylabel='count'>
```









In [99]:
```
Ship_mode_profits=pd.DataFrame(Superstore.groupby('Ship Mode')['Profit'].sum())
Ship_mode_profits.reset_index(inplace=True)
Ship_mode_profits
```

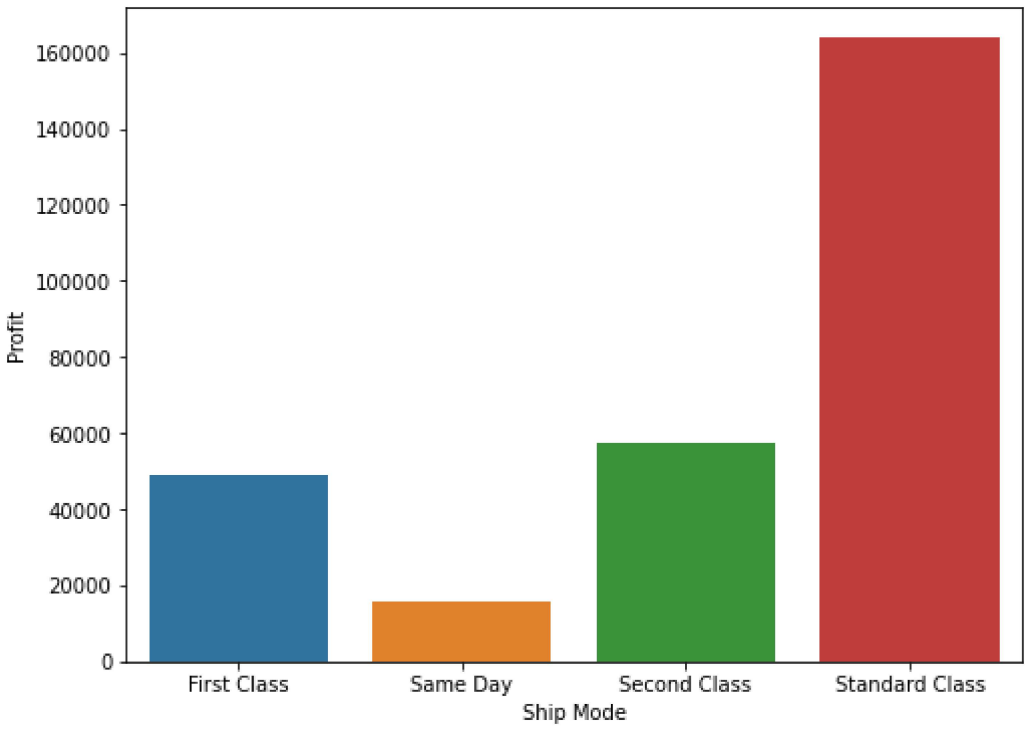
Out[99]:

	Ship Mode	Profit
0	First Class	48953.6561

	Ship Mode	Profit
1	Same Day	15871.8869
2	Second Class	57446.6516
3	Standard Class	163969.2280

In [100…
```
figure=plt.figure(figsize=(8,6))
sns.barplot(x=Ship_mode_profits['Ship Mode'],y=Ship_mode_profits['Profit'],data=Ship_mo
```

Out[100…
```
<AxesSubplot:xlabel='Ship Mode', ylabel='Profit'>
```



In [67]:
```
Superstore['Country'].unique()
```

Out[67]:
```
array(['United States'], dtype=object)
```

In [77]:
```
#As there is only one country in dataset we can remove the country column as it is unne
Superstore=Superstore.drop('Country',axis=1)
Superstore.head()
```

Out[77]:

	Ship Mode	Segment	City	State	Postal Code	Region	Category	Sub-Category	Sales	Quantity
0	Second Class	Consumer	Henderson	Kentucky	42420	South	Furniture	Bookcases	261.9600	2
1	Second Class	Consumer	Henderson	Kentucky	42420	South	Furniture	Chairs	731.9400	3
2	Second Class	Corporate	Los Angeles	California	90036	West	Office Supplies	Labels	14.6200	2

	Ship Mode	Segment	City	State	Postal Code	Region	Category	Sub-Category	Sales	Quantity
3	Standard Class	Consumer	Fort Lauderdale	Florida	33311	South	Furniture	Tables	957.5775	5
4	Standard Class	Consumer	Fort Lauderdale	Florida	33311	South	Office Supplies	Storage	22.3680	2

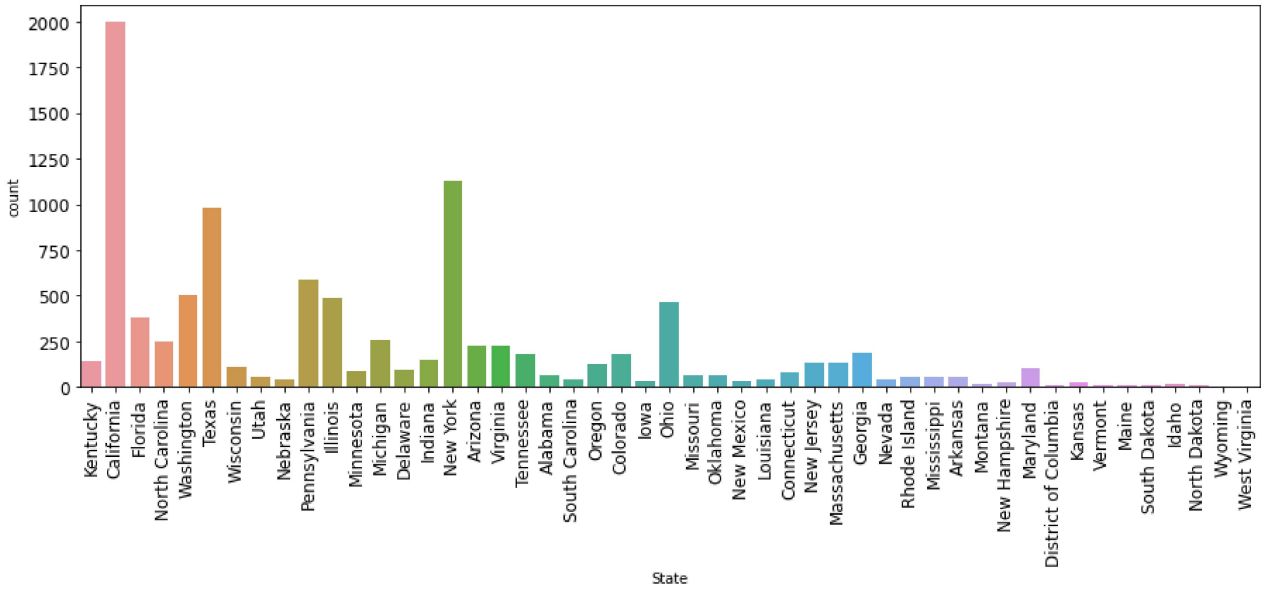
In [78]:
```
Superstore.tail()
```

Out[78]:

	Ship Mode	Segment	City	State	Postal Code	Region	Category	Sub-Category	Sales	Qua
9972	Second Class	Consumer	Miami	Florida	33180	South	Furniture	Furnishings	25.248	
9973	Standard Class	Consumer	Costa Mesa	California	92627	West	Furniture	Furnishings	91.960	
9974	Standard Class	Consumer	Costa Mesa	California	92627	West	Technology	Phones	258.576	
9975	Standard Class	Consumer	Costa Mesa	California	92627	West	Office Supplies	Paper	29.600	
9976	Second Class	Consumer	Westminster	California	92683	West	Office Supplies	Appliances	243.160	

In [79]:
```
#state wise Count plot

figure=plt.figure(figsize=(15,5))
sns.countplot(data=Superstore,x=Superstore['State'])
plt.xticks(rotation=90,fontsize=12)
plt.yticks(fontsize=12)
plt.show()
```



In [101…

```
region_profits=pd.DataFrame(Superstore.groupby('Region')['Sales','Profit'].sum())
region_profits.reset_index(inplace=True)
region_profits
```

```
<ipython-input-101-ce160cecc34c>:1: FutureWarning: Indexing with multiple keys (implicit
ly converted to a tuple of keys) will be deprecated, use a list instead.
  region_profits=pd.DataFrame(Superstore.groupby('Region')['Sales','Profit'].sum())
```

Out[101…

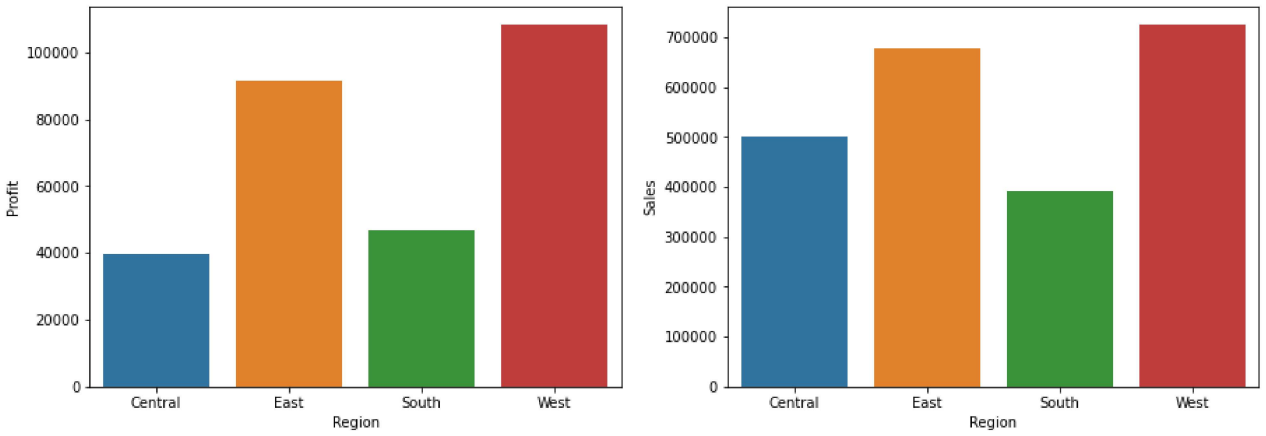
	Region	Sales	Profit
0	Central	500782.8528	39655.8752
1	East	678435.1960	91506.3092
2	South	391721.9050	46749.4303
3	West	725255.6365	108329.8079

In [102…

```
fig,ax2=plt.subplots(1,2,figsize=(15,5))
sns.barplot(x=region_profits['Region'],y=region_profits['Profit'],data=Superstore,ax=ax
sns.barplot(x=region_profits['Region'],y=region_profits['Sales'],data=Superstore,ax=ax2
```

Out[102…

```
<AxesSubplot:xlabel='Region', ylabel='Sales'>
```



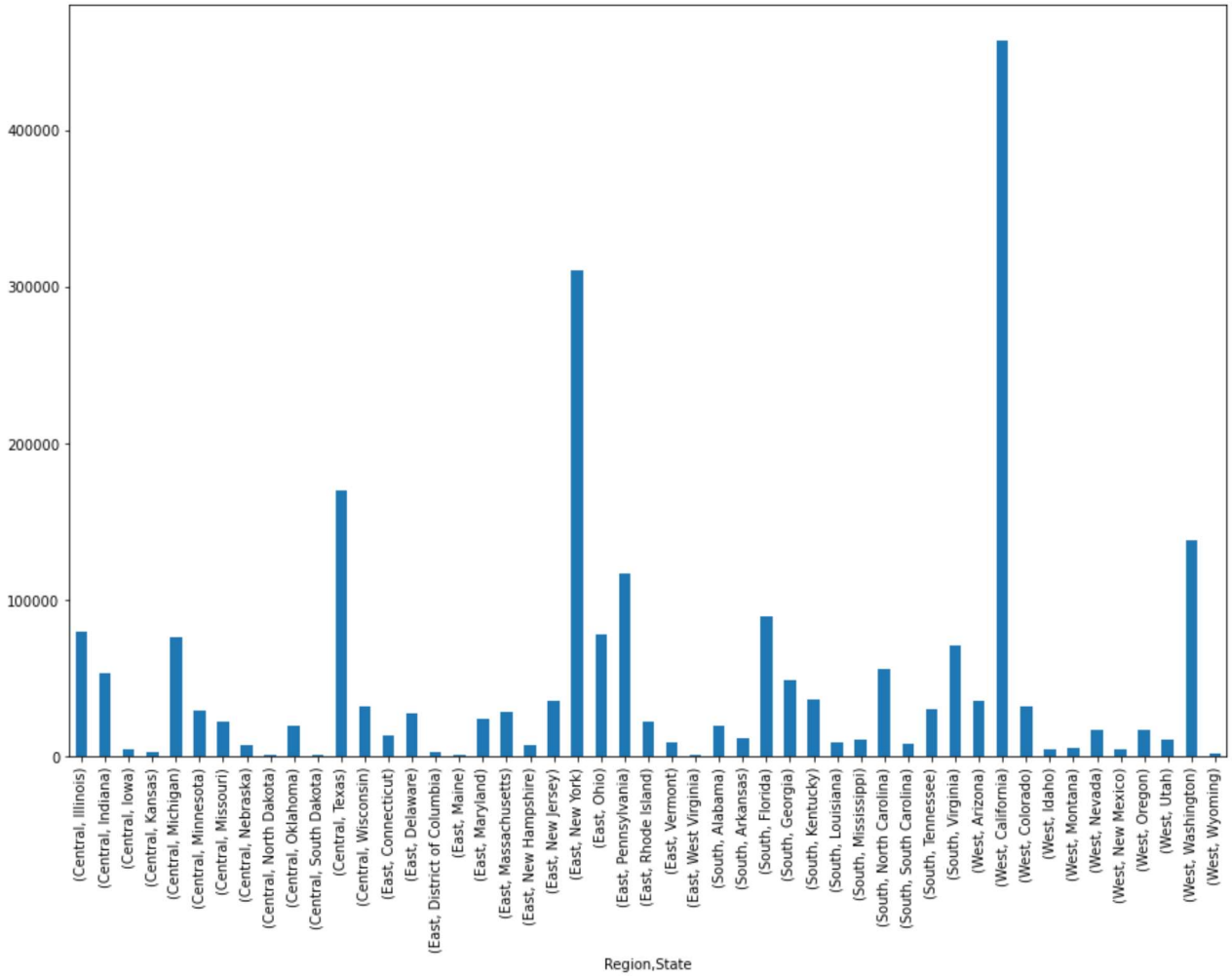
The sales and profit are the highest in West region

In [104…
```
reg_state_sales=Superstore.groupby(['Region','State'])['Sales'].sum()
reg_state_sales
```

Out[104…
```
Region   State
Central  Illinois                80162.5370
         Indiana                 53555.3600
         Iowa                     4579.7600
         Kansas                   2914.3100
         Michigan                75879.6440
         Minnesota               29863.1500
         Missouri                22205.1500
         Nebraska                 7464.9300
         North Dakota              919.9100
         Oklahoma                19683.3900
         South Dakota             1315.5600
         Texas                  170124.5418
         Wisconsin               32114.6100
East     Connecticut             13384.3570
         Delaware                27451.0690
         District of Columbia     2865.0200
         Maine                    1270.5300
         Maryland                23705.5230
         Massachusetts           28634.4340
         New Hampshire            7292.5240
         New Jersey              35764.3120
         New York               310827.1510
         Ohio                    77976.7640
         Pennsylvania           116496.3620
         Rhode Island            22627.9560
         Vermont                  8929.3700
         West Virginia            1209.8240
South    Alabama                 19510.6400
         Arkansas                11678.1300
         Florida                 89473.7080
         Georgia                 49095.8400
         Kentucky                36591.7500
         Louisiana                9217.0300
         Mississippi             10771.3400
         North Carolina          55603.1640
         South Carolina           8481.7100
         Tennessee               30661.8730
         Virginia                70636.7200
West     Arizona                 35282.0010
         California             457576.2715
         Colorado                32108.1180
         Idaho                    4382.4860
         Montana                  5589.3520
         Nevada                  16729.1020
         New Mexico               4783.5220
         Oregon                  17420.7820
         Utah                    11220.0560
         Washington             138560.8100
         Wyoming                  1603.1360
Name: Sales, dtype: float64
```

In [105…
```
reg_state_sales.plot(kind='bar',figsize=(15,10))
```

Out[105…
```
<AxesSubplot:xlabel='Region,State'>
```

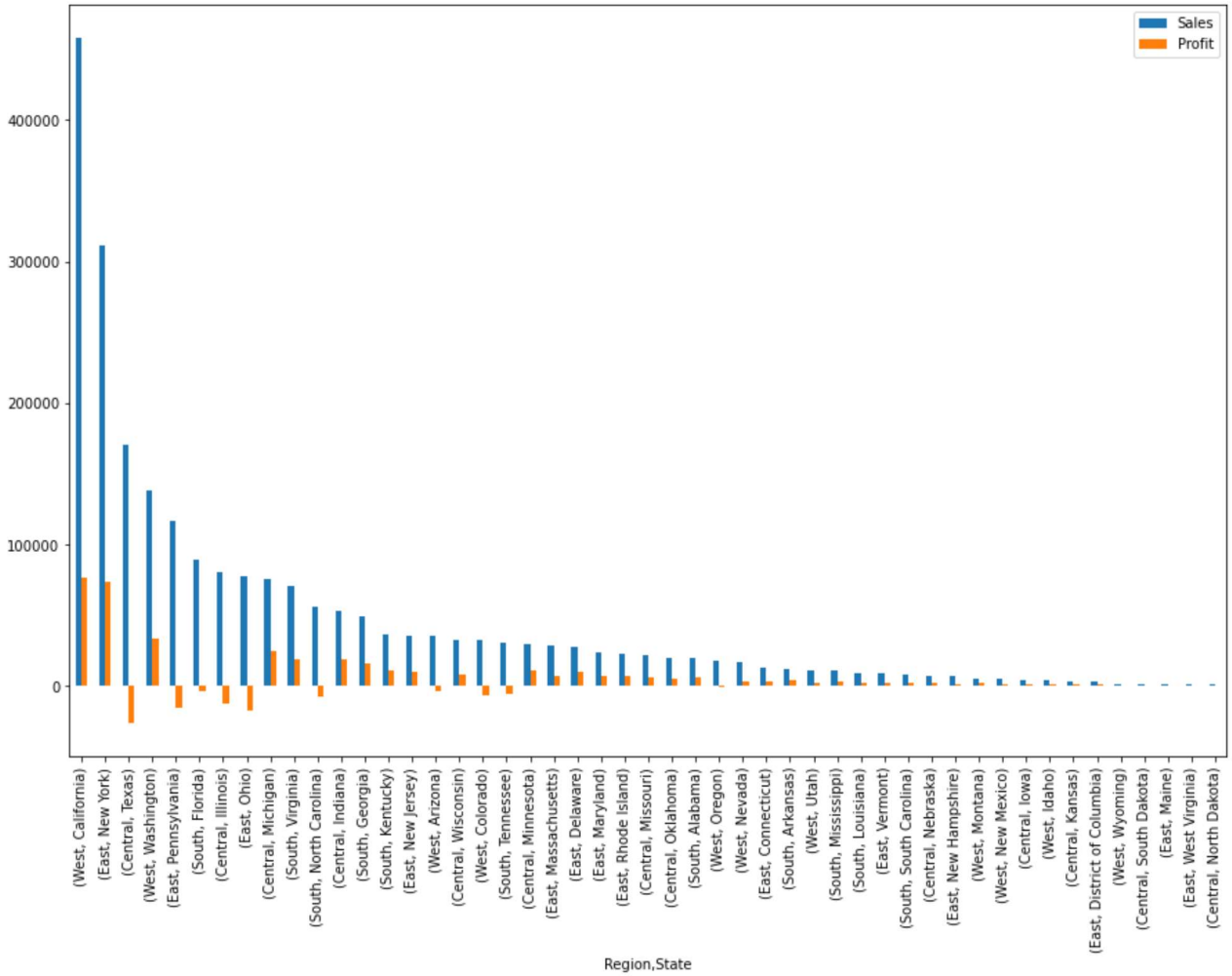


In [111…

```
reg_state_ps=Superstore.groupby(['Region','State'])[['Sales','Profit']].sum().sort_valu
reg_state_ps.plot(figsize=(15,10),kind='bar')
```

Out[111…

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



This shows that California which is in west region has highest profit and sales followed with New York from East region This also satisfies the positive corelation between sales and profit

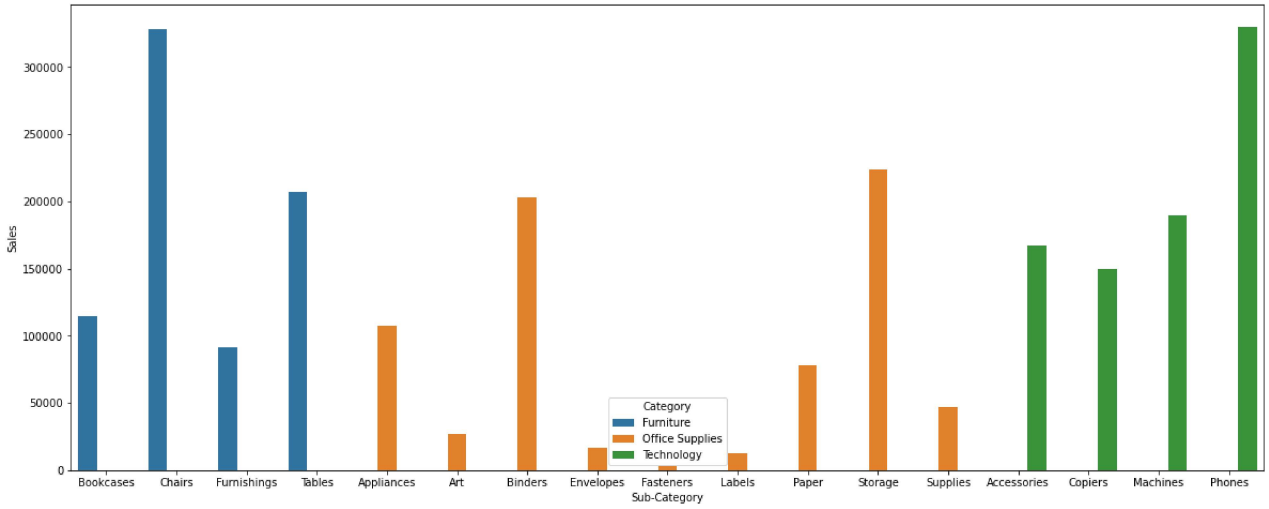
In [127…

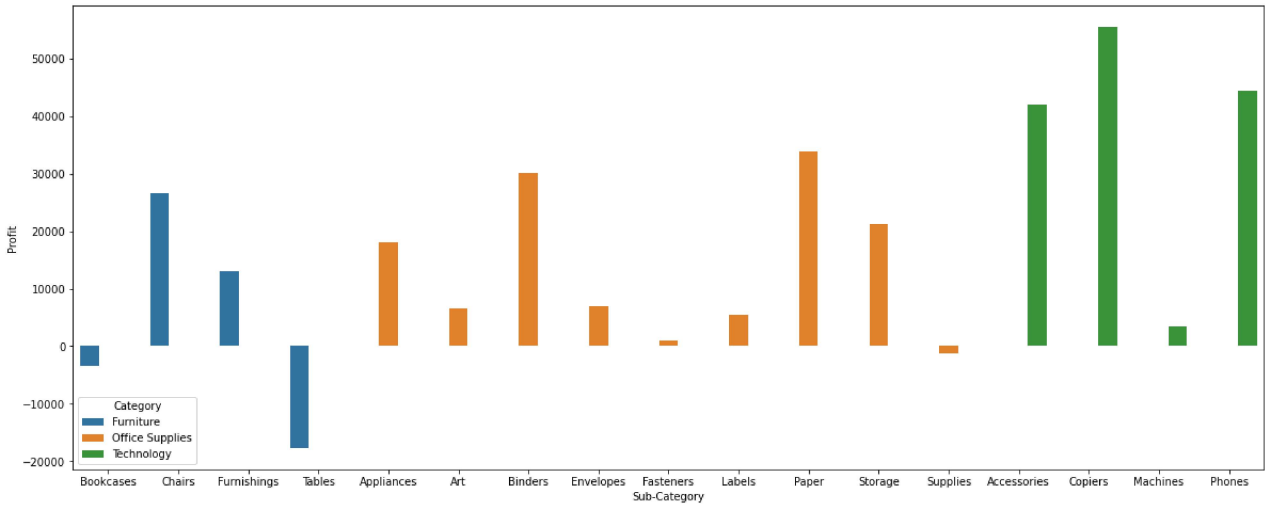
```
#Plotting the sales and profit graph for different subcategories in categories

Cat_Subcat=Superstore.groupby(['Category','Sub-Category'])[['Sales','Profit']].sum()[['
Cat_Subcat.reset_index(inplace=True)
fig1=plt.figure(figsize=(20,8))
sns.barplot(x=Cat_Subcat['Sub-Category'],y=Cat_Subcat['Sales'],hue=Cat_Subcat['Category
fig2=plt.figure(figsize=(20,8))
sns.barplot(x=Cat_Subcat['Sub-Category'],y=Cat_Subcat['Profit'],hue=Cat_Subcat['Categor
```

Out[127…

```
<AxesSubplot:xlabel='Sub-Category', ylabel='Profit'>
```

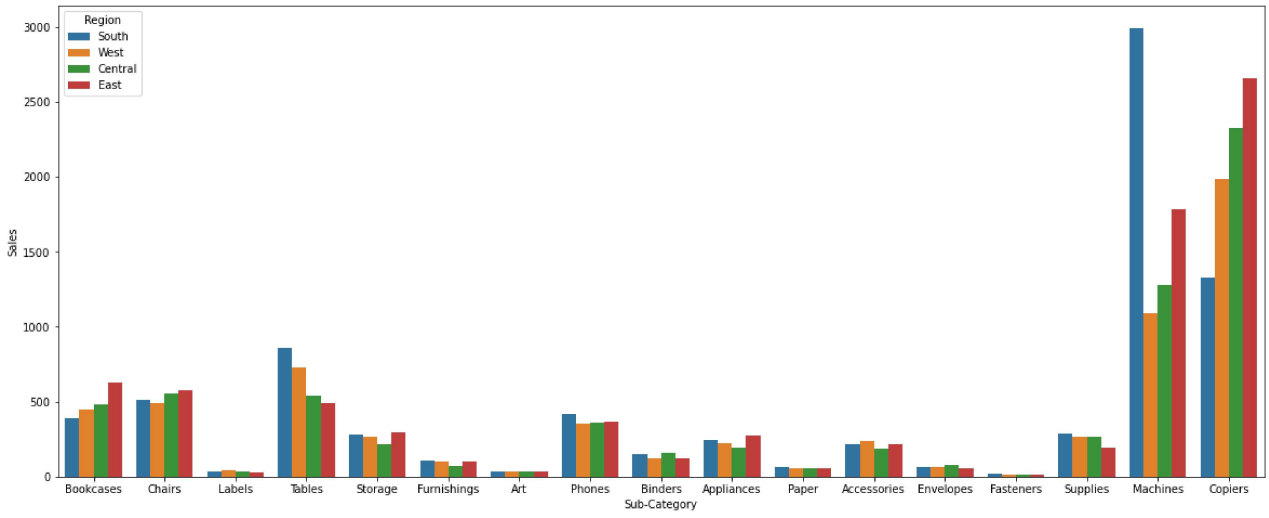




In [132…

```
fig2=plt.figure(figsize=(20,8))
sns.barplot(x=Superstore['Sub-Category'],y=Superstore['Sales'],hue=Superstore['Region']
```

Out[132… <AxesSubplot:xlabel='Sub-Category', ylabel='Sales'>



Observation:

1. The Office suppies category, West Region, Consumer Segment and Standard class Shipmode has highest number of sales and profit.

2. It is the positive corelation between sales and profit is strongly visible in all the ways except for the subcategories
3. Even if the highest sales were in phones subcategories in technology, The highest profits are in copiers 4.East and west region has highest profit and sales

Suggestions:

1. Increase the sales in south and central region
2. Focus on increasing sales in home supplies and Corporate segment
3. Focus on sales of category other than office supplies that will increase the overall profit.

Thank You!!!

In [ ]: