HARSHIT GUPTA

DATA SCIENCE INTERN @THE SPARK FOUNDATION

DATASET : SAMPLESUPERSTORE.CSV (https://bit.ly/3i4rbWl)

EXPLORATORY DATA ANALYSIS - RETAIL

In [1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

In [2]:
```
df=pd.read_csv('Downloads\\SampleSuperstore.csv')
```

In [3]:
```
df.head()
```

Out[3]:

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Sub-Category	Sal
0	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Bookcases	261.96(
1	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Chairs	731.94(
2	Second Class	Corporate	United States	Los Angeles	California	90036	West	Office Supplies	Labels	14.62(
3	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Furniture	Tables	957.57
4	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Office Supplies	Storage	22.36(

In [4]:
```
df.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 [5]:
```
df.isnull().sum()
```

Out[5]:
```
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 [6]:
```
df.shape
```

Out[6]:
```
(9994, 13)
```

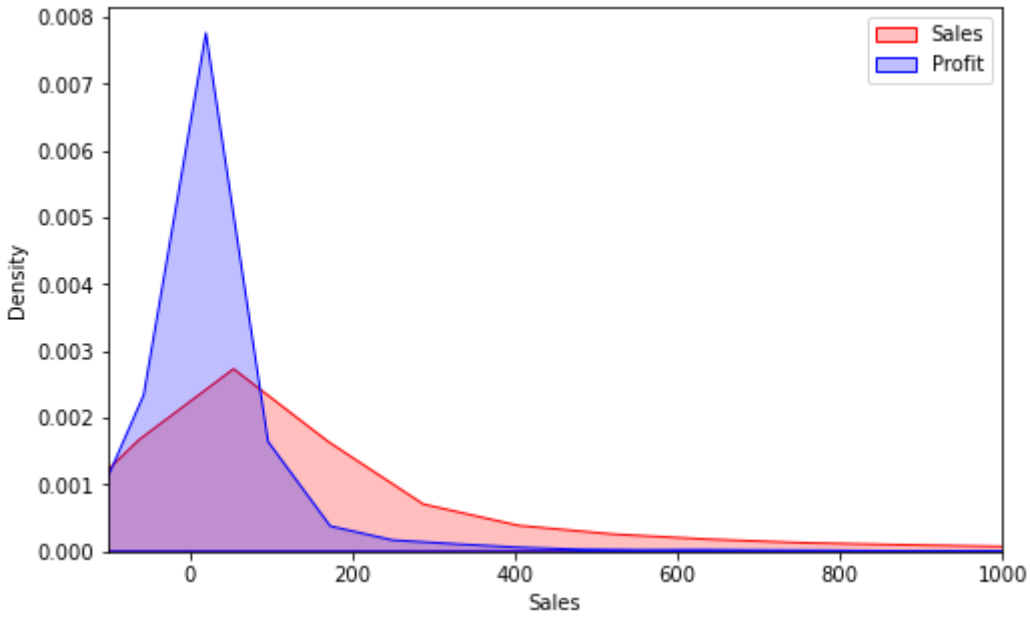
In [7]:
```
df.columns
```

Out[7]:
```
Index(['Ship Mode', 'Segment', 'Country', 'City', 'State', 'Postal Code',
       'Region', 'Category', 'Sub-Category', 'Sales', 'Quantity', 'Discount',
       'Profit'],
      dtype='object')
```

Exploratory Data Analysis

In [8]:
```
plt.figure(figsize=(8,5))
sns.kdeplot(df['Sales'],color='red',label='Sales',shade=True)
sns.kdeplot(df['Profit'],color='Blue',label='Profit',shade=True)
plt.xlim([-100,1000])
plt.legend()
```

Out[8]:
```
<matplotlib.legend.Legend at 0x1e2c09a6df0>
```

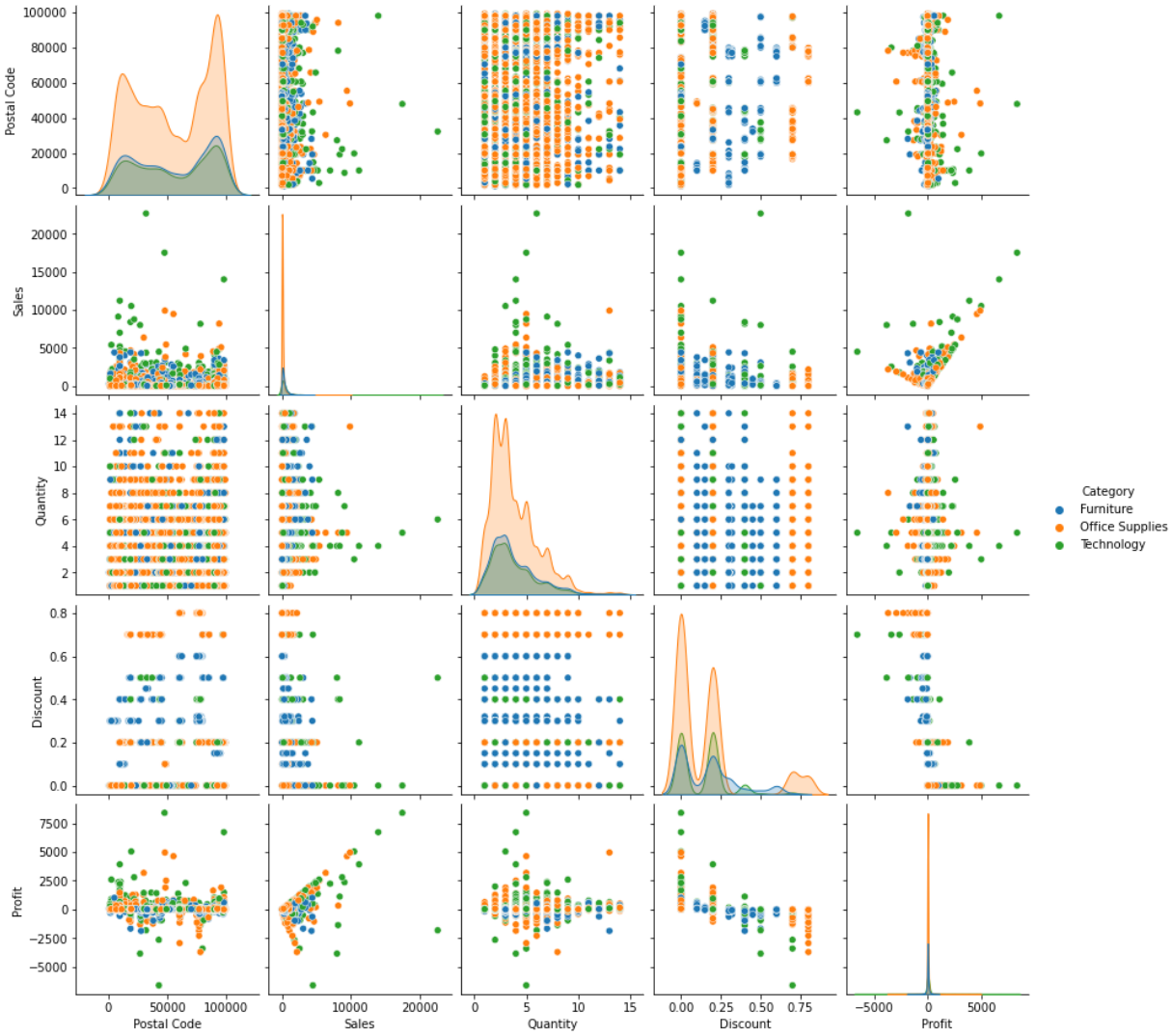


In [9]:
```
sns.pairplot(df,hue='Region')
```

Out[9]:
```
<seaborn.axisgrid.PairGrid at 0x1e2c06744f0>
```

In [10]:
```
sns.pairplot(df,hue='Category')
```

Out[10]:
```
<seaborn.axisgrid.PairGrid at 0x1e2c26a8b80>
```



In [11]:
```
df.corr
```

Out[11]:
```
<bound method DataFrame.corr of            Ship Mode    Segment        Country
City       State  \
0       Second Class   Consumer  United States        Henderson    Kentucky
1       Second Class   Consumer  United States        Henderson    Kentucky
2       Second Class  Corporate  United States      Los Angeles  California
3     Standard Class   Consumer  United States  Fort Lauderdale     Florida
4     Standard Class   Consumer  United States  Fort Lauderdale     Florida
...              ...        ...            ...              ...         ...
9989    Second Class   Consumer  United States            Miami     Florida
9990  Standard Class   Consumer  United States       Costa Mesa  California
9991  Standard Class   Consumer  United States       Costa Mesa  California
9992  Standard Class   Consumer  United States       Costa Mesa  California
9993    Second Class   Consumer  United States      Westminster  California

      Postal Code Region         Category Sub-Category     Sales  Quantity  \
0           42420  South        Furniture    Bookcases  261.9600         2
1           42420  South        Furniture       Chairs  731.9400         3
2           90036   West  Office Supplies       Labels   14.6200         2
3           33311  South        Furniture       Tables  957.5775         5
4           33311  South  Office Supplies      Storage   22.3680         2
...           ...    ...              ...          ...       ...       ...
9989        33180  South        Furniture  Furnishings   25.2480         3
9990        92627   West        Furniture  Furnishings   91.9600         2
9991        92627   West       Technology       Phones  258.5760         2
9992        92627   West  Office Supplies        Paper   29.6000         4
9993        92683   West  Office Supplies   Appliances  243.1600         2

      Discount    Profit
0         0.00   41.9136
1         0.00  219.5820
```

```
2          0.00    6.8714
3          0.45 -383.0310
4          0.20    2.5164
...         ...       ...
9989       0.20    4.1028
9990       0.00   15.6332
9991       0.20   19.3932
9992       0.00   13.3200
9993       0.00   72.9480

[9994 rows x 13 columns]>
```

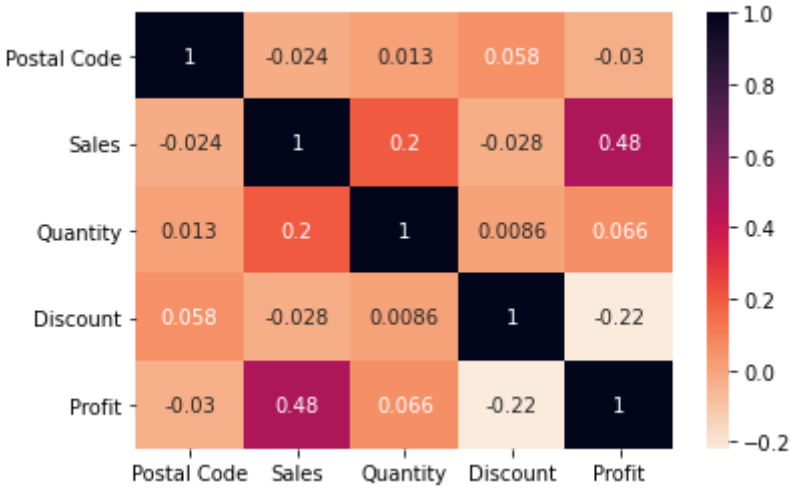
In [12]:
```
df.corr()
```

Out[12]:

	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 [13]:
```
sns.heatmap(df.corr(),cmap='rocket_r',annot=True)
```

Out[13]: <AxesSubplot:>



From above Heatmap:

Sales and Profit are Moderately Correlated.

Discount and Profit are Negatively Correlated.

Quantity and Profit are less Moderately Correlated.

COUNTPLOT FOR EACH COLUMNS

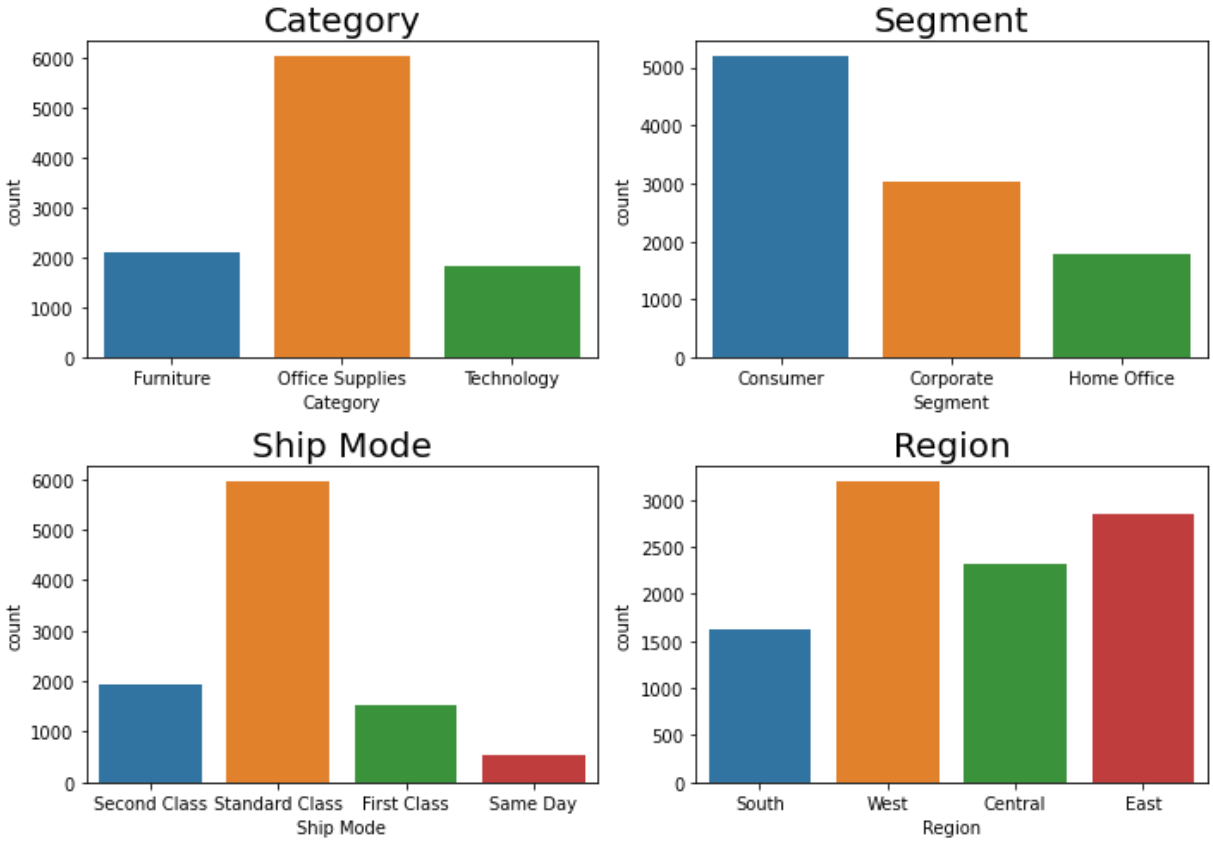
In [14]:
```
import warnings
warnings.filterwarnings('ignore')
```

In [15]:
```
fig,axs=plt.subplots(nrows=2,ncols=2,figsize=(10,7));

sns.countplot(df['Category'],ax=axs[0][0])
sns.countplot(df['Segment'],ax=axs[0][1])
```

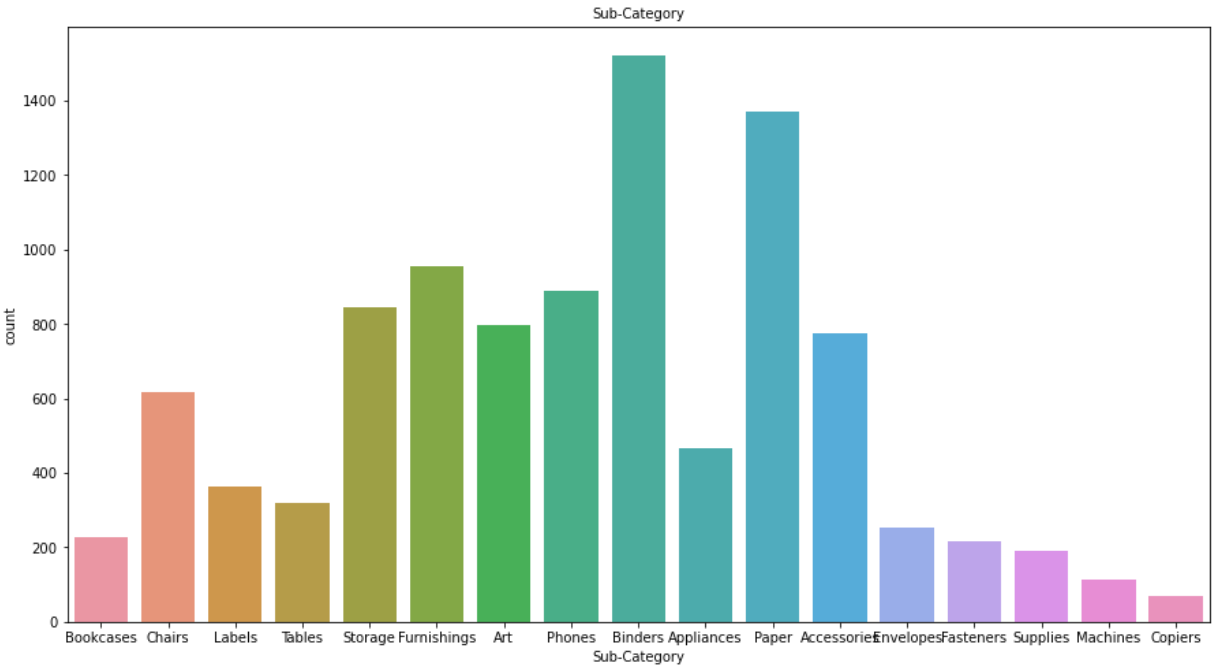
```
sns.countplot(df['Ship Mode'],ax=axs[1][0])
sns.countplot(df['Region'],ax=axs[1][1])
axs[0][0].set_title('Category',fontsize=20)
axs[0][1].set_title('Segment',fontsize=20)
axs[1][0].set_title('Ship Mode',fontsize=20)
axs[1][1].set_title('Region',fontsize=20)

plt.tight_layout()
```



In [16]:
```
plt.figure(figsize=(15,8))
sns.countplot(df['Sub-Category'])
plt.title('Sub-Category',fontsize=10)
```

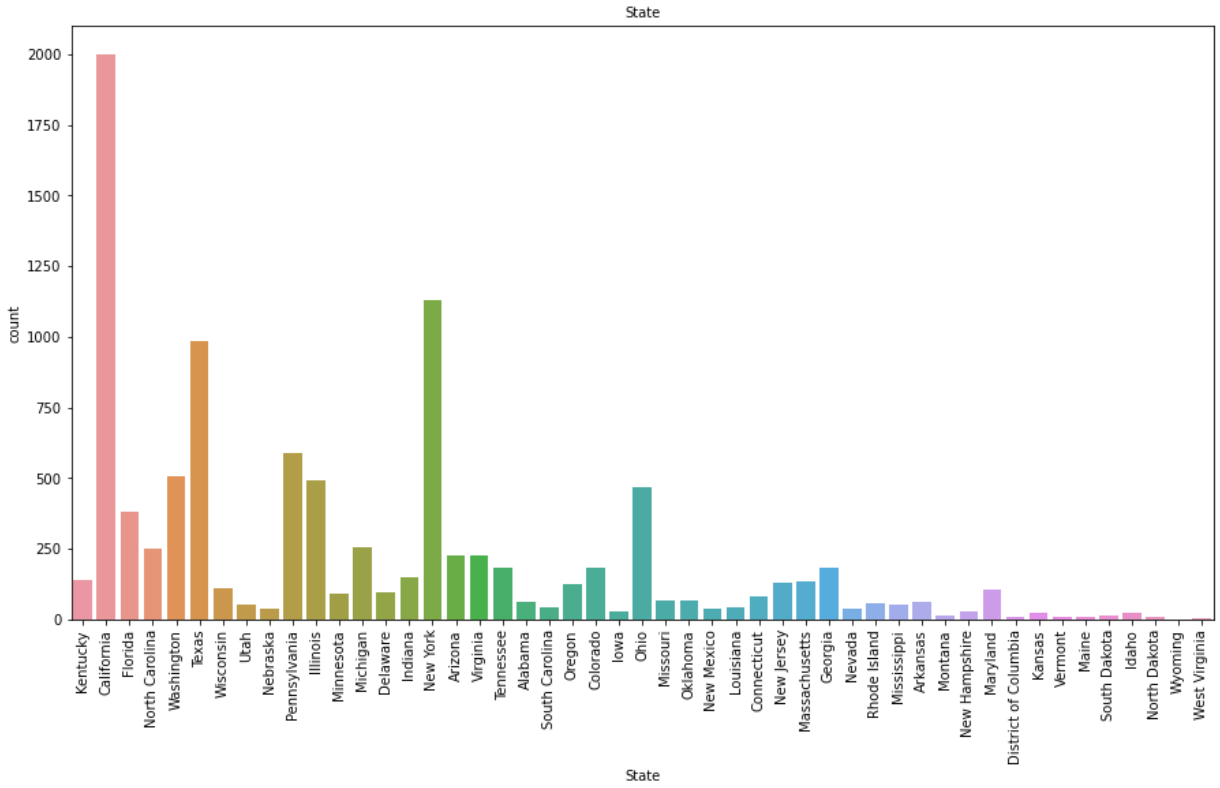
Out[16]: Text(0.5, 1.0, 'Sub-Category')




```
plt.figure(figsize=(15,8))
```

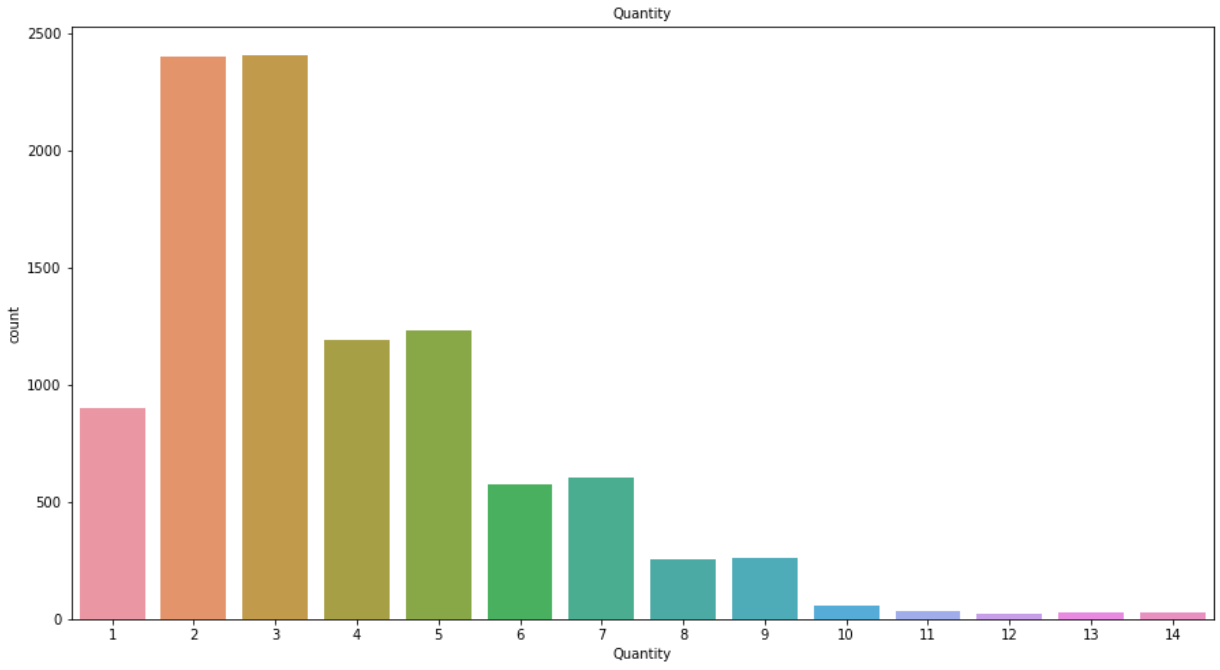
In [17]:
```
sns.countplot(df['State'])
plt.xticks(rotation=90)
plt.title('State',fontsize=10)
```

Out[17]:
```
Text(0.5, 1.0, 'State')
```



In [18]:
```
plt.figure(figsize=(15,8))
sns.countplot(df['Quantity'])
plt.title('Quantity',fontsize=10)
```

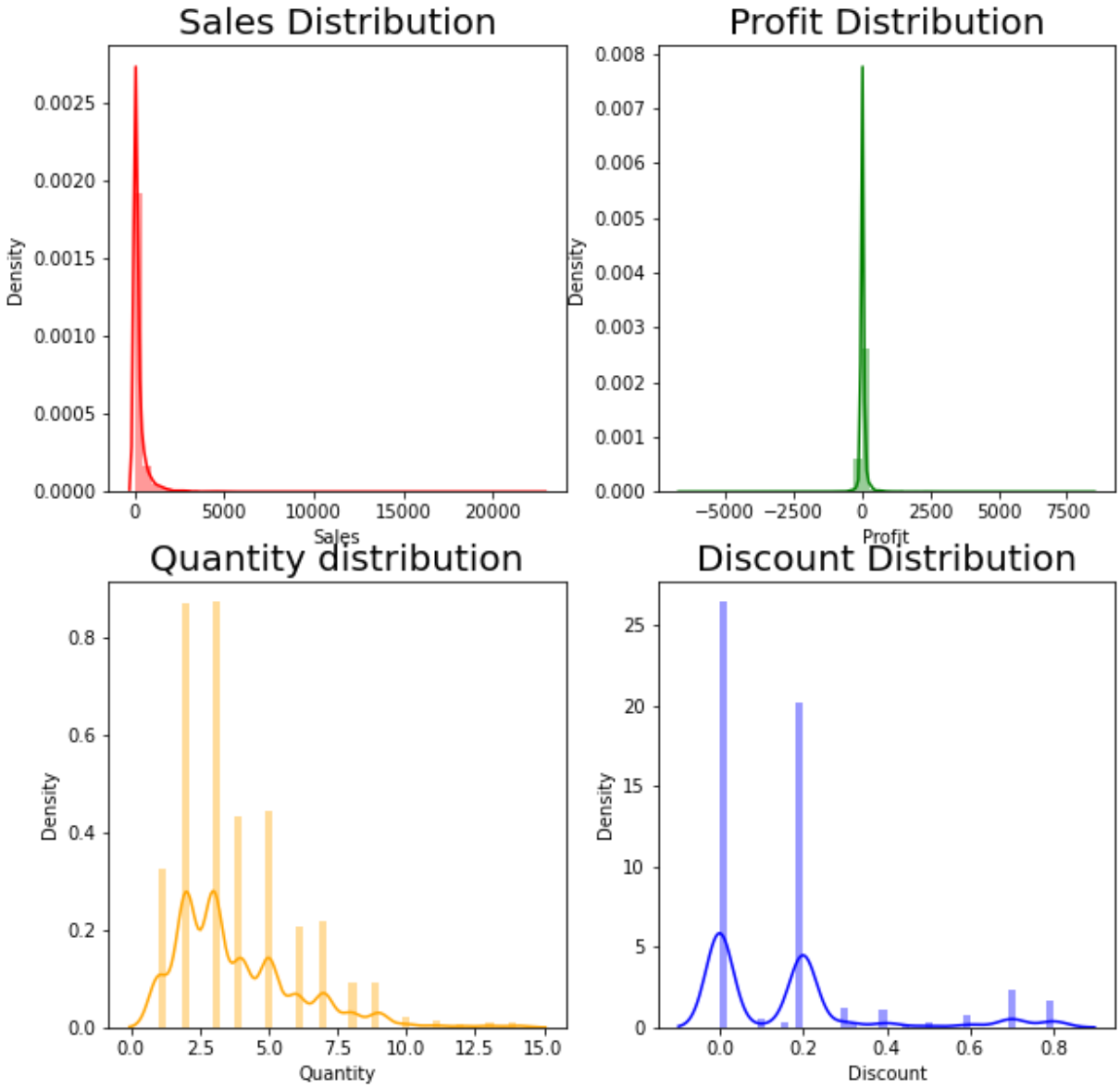
Out[18]:
```
Text(0.5, 1.0, 'Quantity')
```



Distribution of the data using the plot

In [19]:
```
fig, axs = plt.subplots(ncols=2, nrows = 2, figsize = (10,10))
sns.distplot(df['Sales'], color = 'red',  ax = axs[0][0])
```

```
sns.distplot(df['Profit'], color = 'green',  ax = axs[0][1])
sns.distplot(df['Quantity'], color = 'orange',  ax = axs[1][0])
sns.distplot(df['Discount'], color = 'blue',  ax = axs[1][1])
axs[0][0].set_title('Sales Distribution', fontsize = 20)
axs[0][1].set_title('Profit Distribution', fontsize = 20)
axs[1][0].set_title('Quantity distribution', fontsize = 20)
axs[1][1].set_title('Discount Distribution', fontsize = 20)
plt.show()
```



STATE WISE DEAL

In [20]:
```
df['Country'].value_counts()
```

Out[20]:
```
United States    9994
Name: Country, dtype: int64
```

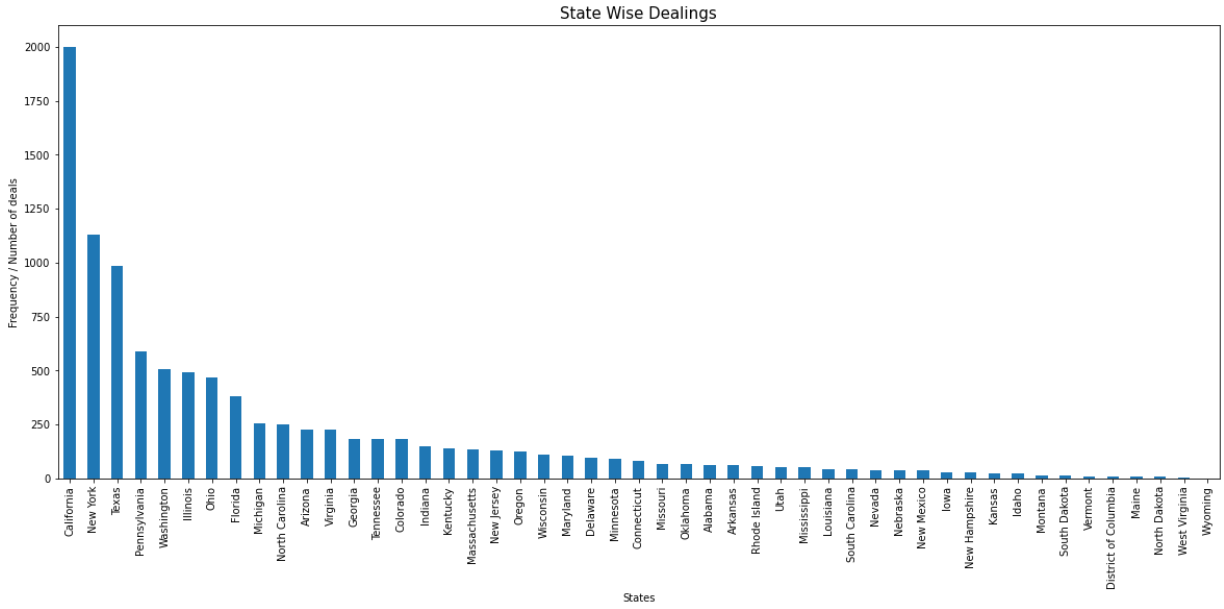
In [24]:
```
data=df['State'].value_counts()
data.head(10)
```

Out[24]:
```
California        2001
New York          1128
Texas              985
Pennsylvania       587
Washington         506
Illinois           492
Ohio               469
```

```
Florida            383
Michigan           255
North Carolina     249
Name: State, dtype: int64
```

In [26]:
```
data.plot(kind='bar',figsize=(20,8))
plt.ylabel('Frequency / Number of deals')
plt.xlabel('States')

plt.title('State Wise Dealings', fontsize = 15)
plt.show()
```



Califonia,New York,Texas:-These are the top three states where deals are high.

Wyoming has the Lowest Number of deal.

In [27]:
```
df['State'].value_counts().mean()
```

Out[27]:
```
203.9591836734694
```

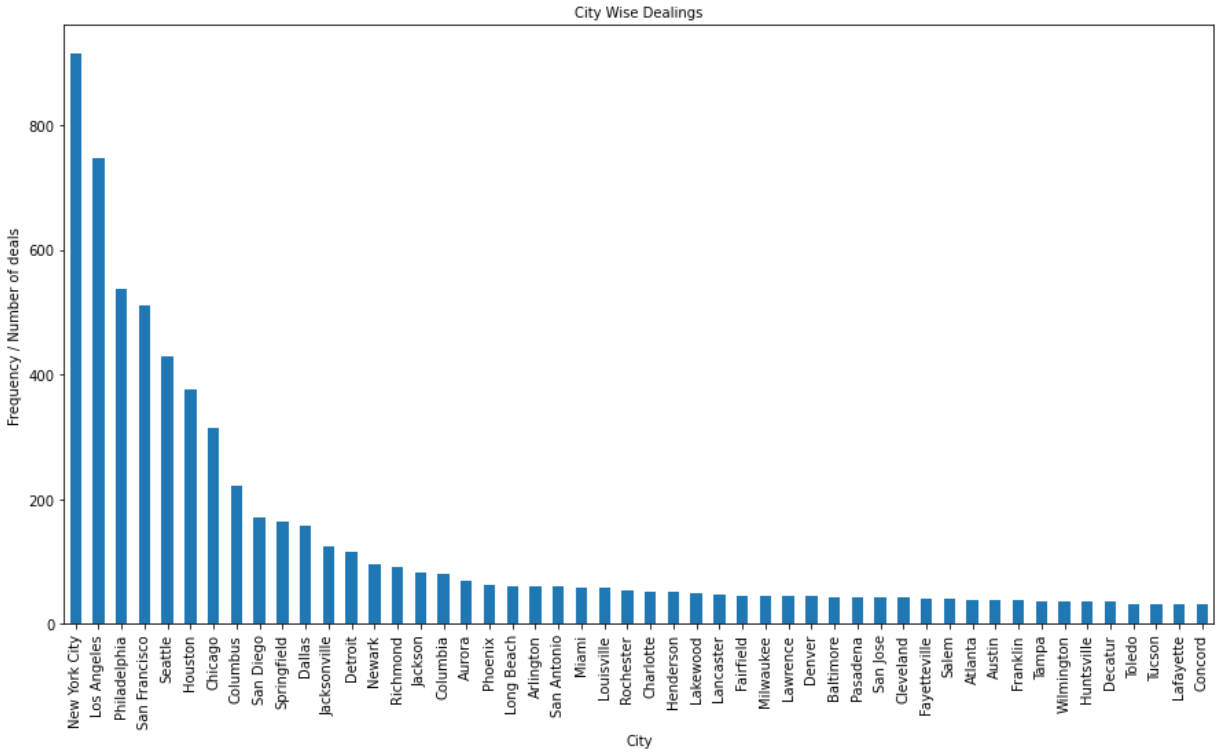
Above is the average number of deals per state.

City Wise analysis of the dealing

In [29]:
```
data1 = df['City'].value_counts()
data1=data1.head(50)
```

In [33]:
```
data1.plot(kind='bar',figsize=(15,8))
plt.ylabel('Frequency / Number of deals')
plt.xlabel('City')

plt.title('City Wise Dealings', fontsize = 10)
plt.show()
```



Top 3 city where deals are Highest:-

1. New York City
2. Los Angeles
3. Philadelphia

In [34]:
```
df['City'].value_counts().mean()
```

Out[34]:
```
18.821092278719398
```

Above is the average deal per city.

Segment Wise Analysis on Profit , Sales and Discounts:

In [35]:
```
df['Segment'].value_counts()
```

Out[35]:
```
Consumer       5191
Corporate      3020
Home Office    1783
Name: Segment, dtype: int64
```

In [36]:
```
df_segment= df.groupby(['Segment'])[['Sales', 'Discount', 'Profit']].mean()
df_segment
```

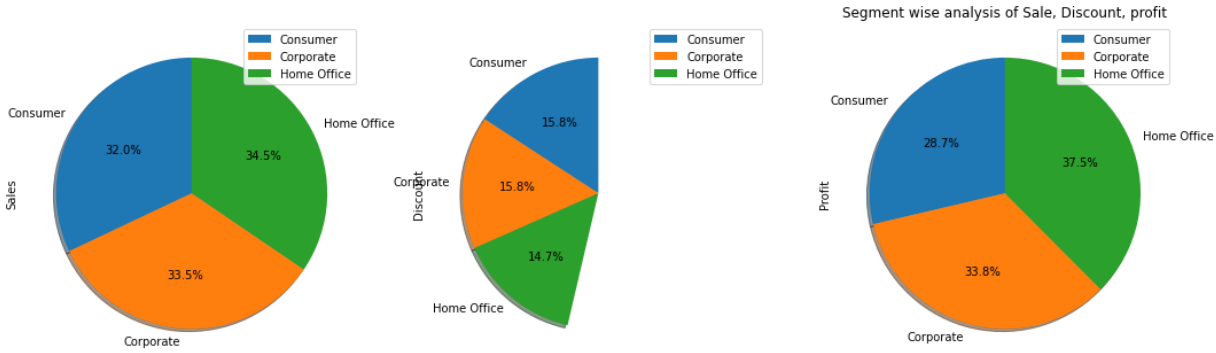
Out[36]:

Segment	Sales	Discount	Profit
Consumer	223.733644	0.158141	25.836873
Corporate	233.823300	0.158228	30.456667
Home Office	240.972041	0.147128	33.818664

In [37]:
```
df_segment.plot.pie(subplots=True,
                    autopct='%1.1f%%',
                    figsize=(18, 20),
```

```
                    startangle=90,
                    shadow=True,
                    labels = df_segment.index)
plt.title('Segment wise analysis of Sale, Discount, profit')
```

Out[37]: Text(0.5, 1.0, 'Segment wise analysis of Sale, Discount, profit')



Sales:

Consumer : 32%

Corporate - 33.5%

Home Office : 34.5%

Discount :

Consumer : 15.8%

Corporate : 15.8%

Home Office : 14.7%

Profit :

Consumer : 15.8%

Corporate : 15.8%

Home Office : 14.7%

Statewise analysis of Profit Discount and sell

In [38]:
```
df['State'].value_counts().head(10)
```

Out[38]:
```
California        2001
New York          1128
Texas              985
Pennsylvania       587
Washington         506
Illinois           492
Ohio               469
Florida            383
Michigan           255
North Carolina     249
Name: State, dtype: int64
```

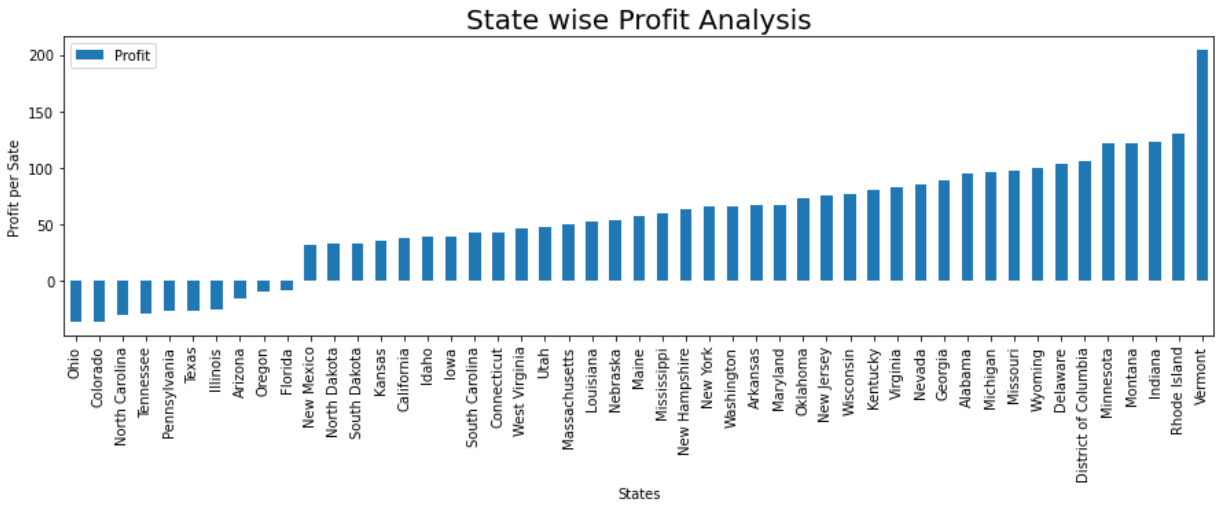
In [39]:
```
df_state= df.groupby(['State'])[['Sales', 'Discount', 'Profit']].mean()
df_state.head(10)
```

Out[39]:

State	Sales	Discount	Profit
Alabama	319.846557	0.000000	94.865989
Arizona	157.508933	0.303571	-15.303235
Arkansas	194.635500	0.000000	66.811452
California	228.729451	0.072764	38.171608
Colorado	176.418231	0.316484	-35.867351
Connecticut	163.223866	0.007317	42.823071
Delaware	285.948635	0.006250	103.930988
District of Columbia	286.502000	0.000000	105.958930
Florida	233.612815	0.299347	-8.875461
Georgia	266.825217	0.000000	88.315453

In [40]:
```
df_state1=df_state.sort_values('Profit')

df_state1[['Profit']].plot(kind = 'bar', figsize = (15,4))
plt.title('State wise Profit Analysis', fontsize = 20)
plt.ylabel('Profit per Sate')
plt.xlabel('States')
plt.show()
```

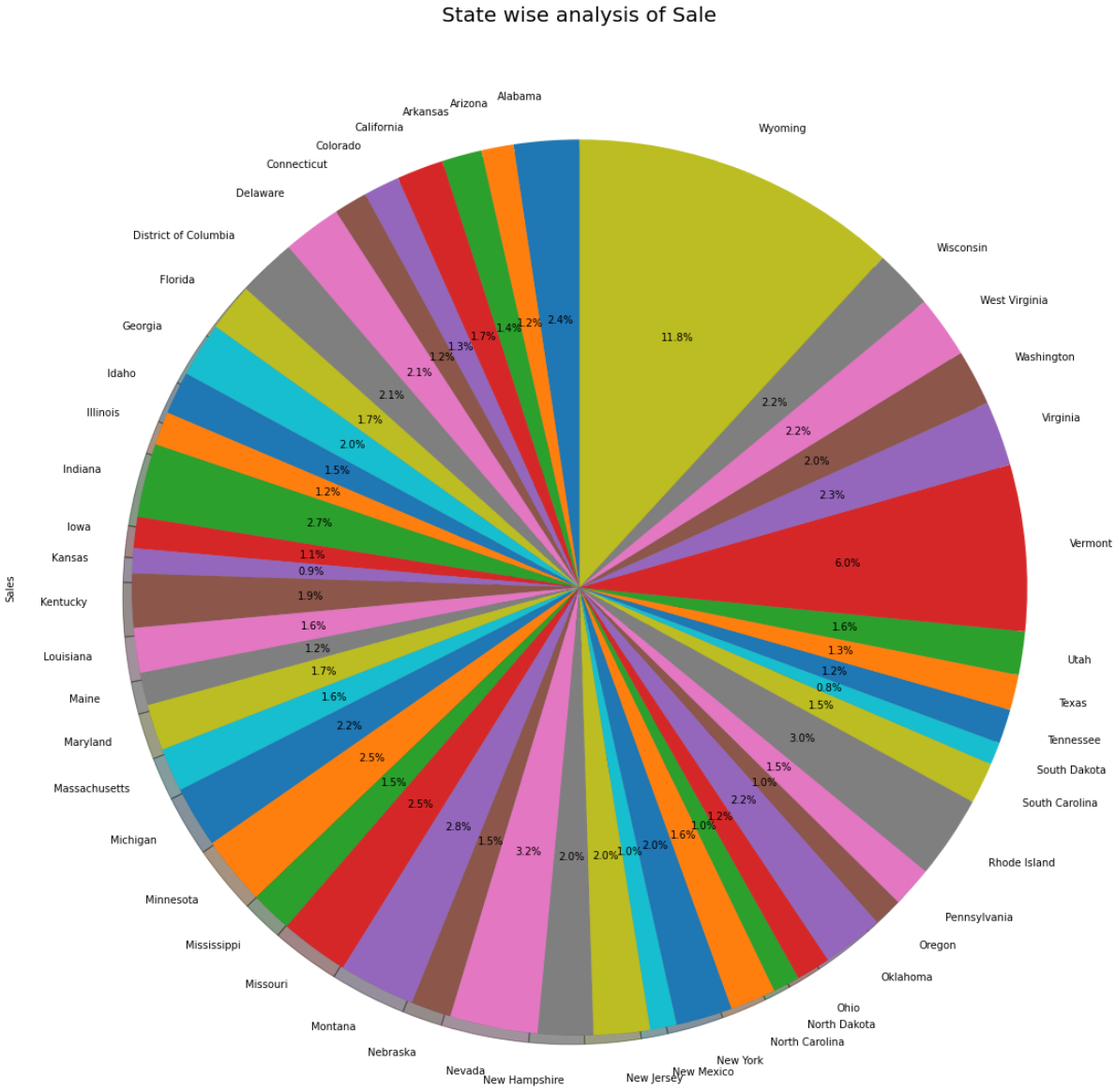


Vermont: Highest Profit

Ohio: Lowest Profit

In [41]:
```
df_state['Sales'].plot(kind='pie',
                       figsize = (20,20),
                       autopct='%1.1f%%',
                       startangle=90,     # start angle 90° (Africa)
                       shadow=True)
plt.title('State wise analysis of Sale',fontsize=20)
```

Out[41]:
```
Text(0.5, 1.0, 'State wise analysis of Sale')
```



Highest amount of sales= Wyoming(11.8%)

Lowest amount of sales= South Dakota(0.8%)

CityWise Analysis of Profit

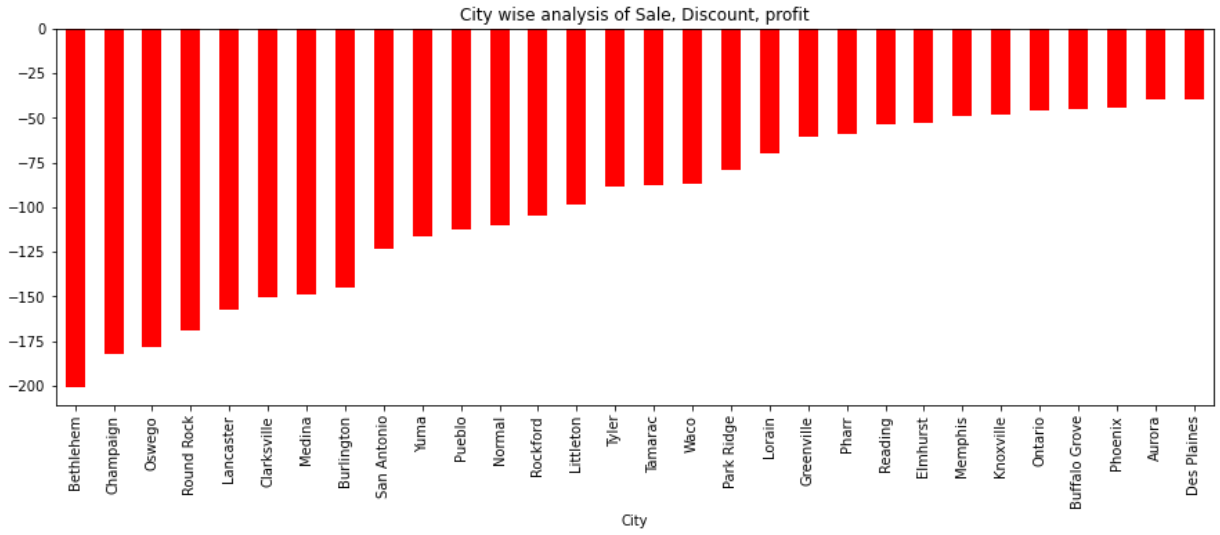
In [42]:
```
df_city= df.groupby(['City'])[['Sales', 'Discount', 'Profit']].mean()
df_city = df_city.sort_values('Profit')
df_city.head()
```

Out[42]:

City	Sales	Discount	Profit
Bethlehem	337.926800	0.380000	-200.619160
Champaign	151.960000	0.600000	-182.352000
Oswego	107.326000	0.600000	-178.709200
Round Rock	693.436114	0.274286	-169.061614
Lancaster	215.031826	0.315217	-157.371052

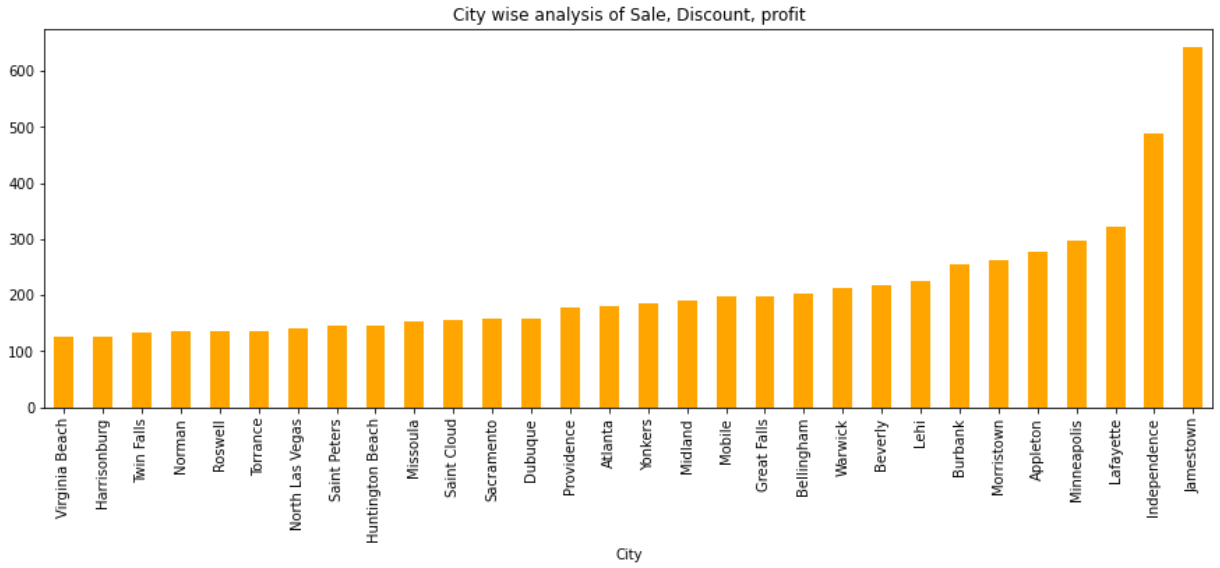
In [43]:
```
#1.Low Profit
df_city['Profit'].head(30).plot(kind='bar',figsize=(15,5),color = 'Red')
plt.title('City wise analysis of Sale, Discount, profit')
```

Out[43]: Text(0.5, 1.0, 'City wise analysis of Sale, Discount, profit')



In [44]:
```
#2. High Profit
df_city['Profit'].tail(30).plot(kind='bar',figsize=(15,5),color = 'Orange')
plt.title('City wise analysis of Sale, Discount, profit')
```

Out[44]: Text(0.5, 1.0, 'City wise analysis of Sale, Discount, profit')



QUANTITY WISE SALES, PROFIT AND DISCOUNT ANALYSIS

In [45]:
```
df_quantity = df.groupby(['Quantity'])[['Sales', 'Discount', 'Profit']].mean()
df_quantity.head(10)
```

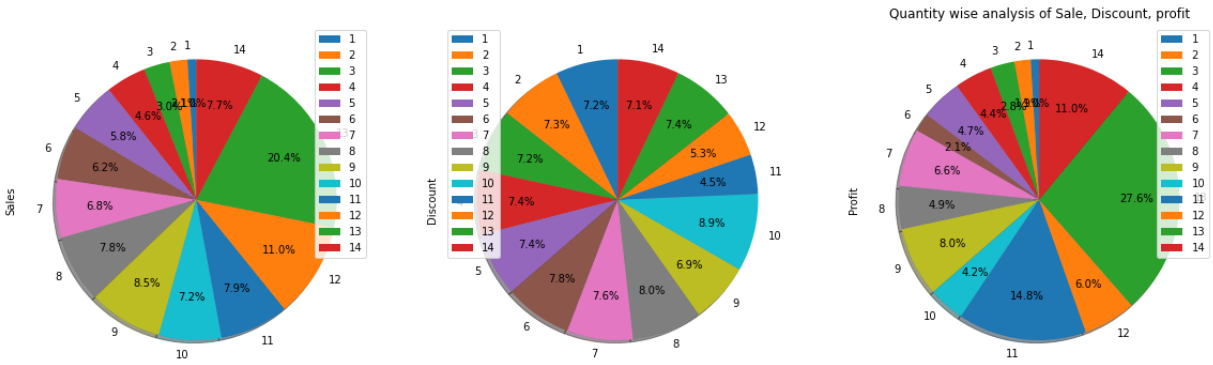
Out[45]:

Quantity	Sales	Discount	Profit
1	59.234632	0.152959	8.276396
2	120.354488	0.154858	16.006831
3	175.201578	0.153329	23.667715
4	271.764059	0.157708	37.131310

Quantity	Sales	Discount	Profit
5	337.936339	0.157146	40.257394
6	362.101960	0.166556	18.051517
7	395.888393	0.161980	56.579163
8	458.210802	0.171595	42.244342
9	498.083683	0.147946	68.557716
10	422.046737	0.190702	35.862404

In [46]:
```
df_quantity.plot.pie(subplots=True,
                     autopct='%1.1f%%',
                     figsize=(20, 20),
                      pctdistance=0.69,
                     startangle=90,
                     shadow=True,
                     labels = df_quantity.index)
plt.title('Quantity wise analysis of Sale, Discount, profit')
```

Out[46]:
```
Text(0.5, 1.0, 'Quantity wise analysis of Sale, Discount, profit')
```



13(green) Number of Quantity is high for sales and Profit.

CATAGORY WISE SALES DISCOUNT AND PROFIT

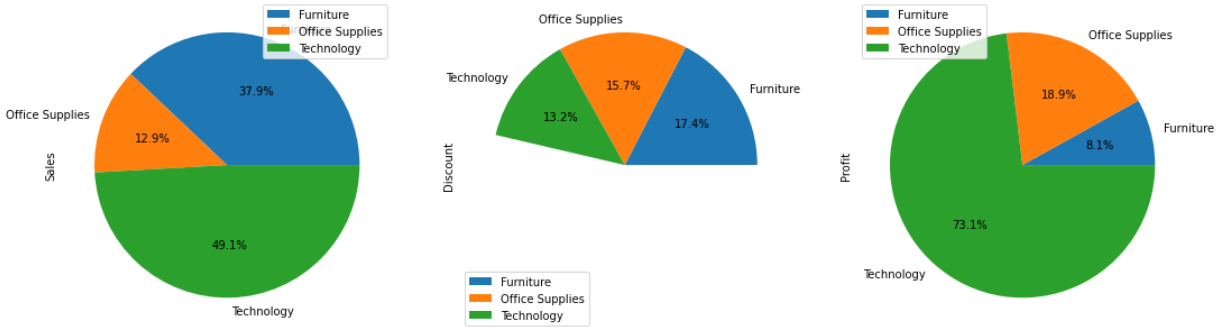
In [47]:
```
df_category = df.groupby(['Category'])[['Sales', 'Discount', 'Profit']].mean()
df_category
```

Out[47]:

Category	Sales	Discount	Profit
Furniture	349.834887	0.173923	8.699327
Office Supplies	119.324101	0.157285	20.327050
Technology	452.709276	0.132323	78.752002

In [48]:
```
df_category.plot.pie(subplots=True,
                     figsize=(18, 20),
                     autopct='%1.1f%%',
                     labels = df_category.index)
```

Out[48]:
```
array([<AxesSubplot:ylabel='Sales'>, <AxesSubplot:ylabel='Discount'>,
       <AxesSubplot:ylabel='Profit'>], dtype=object)
```



Maximun sales and Profit obtain in Technology.

Minimun profit obtain in Furniture

Sub-Category wise Sales, Profit and Discount

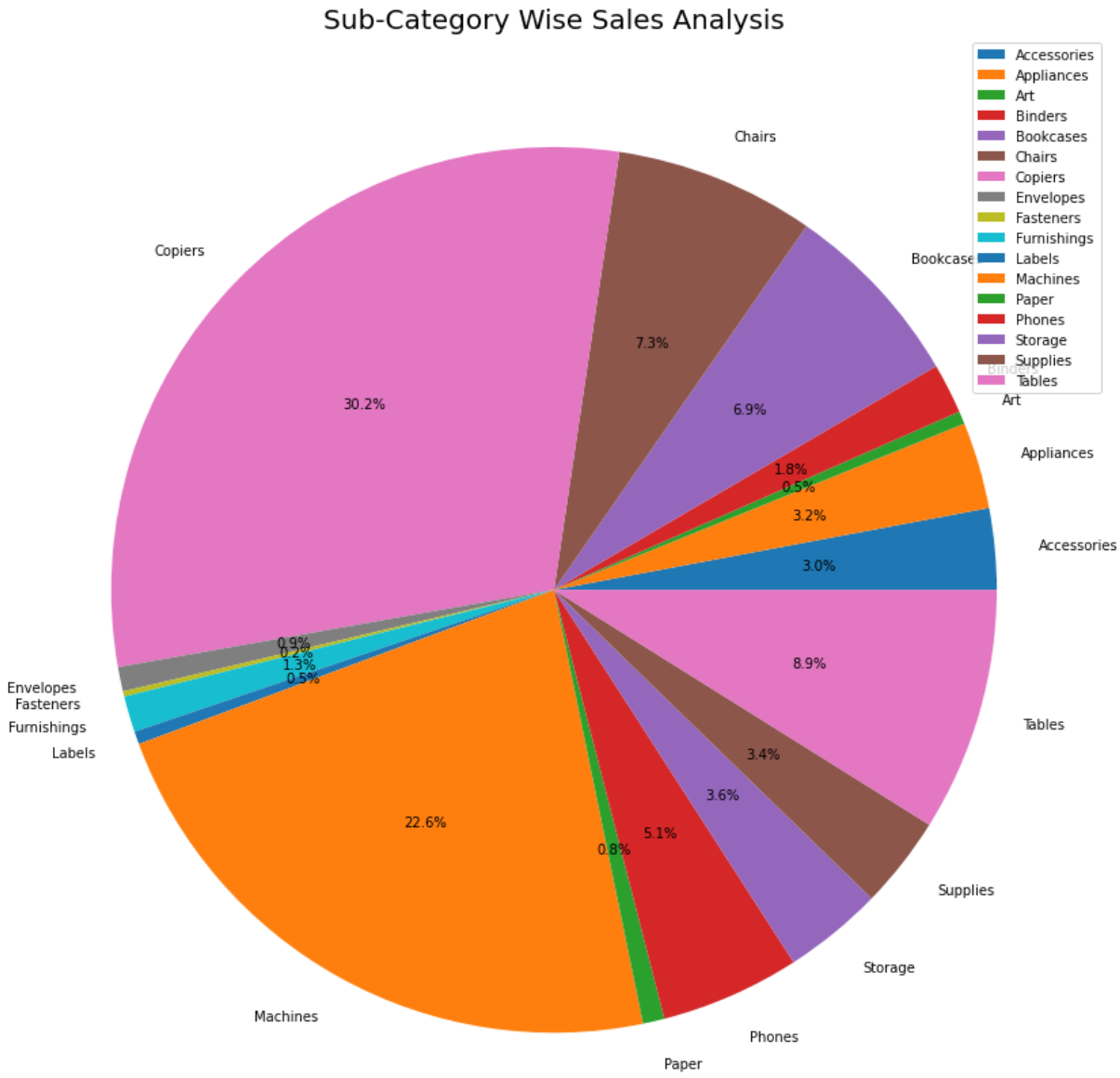
In [49]:
```
df_sub_category = df.groupby(['Sub-Category'])[['Sales', 'Discount', 'Profit']].mean
df_sub_category.head(10)
```

Out[49]:

Sub-Category	Sales	Discount	Profit
Accessories	215.974604	0.078452	54.111788
Appliances	230.755710	0.166524	38.922758
Art	34.068834	0.074874	8.200737
Binders	133.560560	0.372292	19.843574
Bookcases	503.859633	0.211140	-15.230509
Chairs	532.332420	0.170178	43.095894
Copiers	2198.941618	0.161765	817.909190
Envelopes	64.867724	0.080315	27.418019
Fasteners	13.936774	0.082028	4.375660
Furnishings	95.825668	0.138349	13.645918

[1] BASED ON THE SALES

In [50]:
```
plt.figure(figsize = (15,15))
plt.pie(df_sub_category['Sales'], labels = df_sub_category.index, autopct = '%1.1f%%
plt.title('Sub-Category Wise Sales Analysis', fontsize = 20)
plt.legend()
plt.xticks(rotation = 90)
plt.show()
```

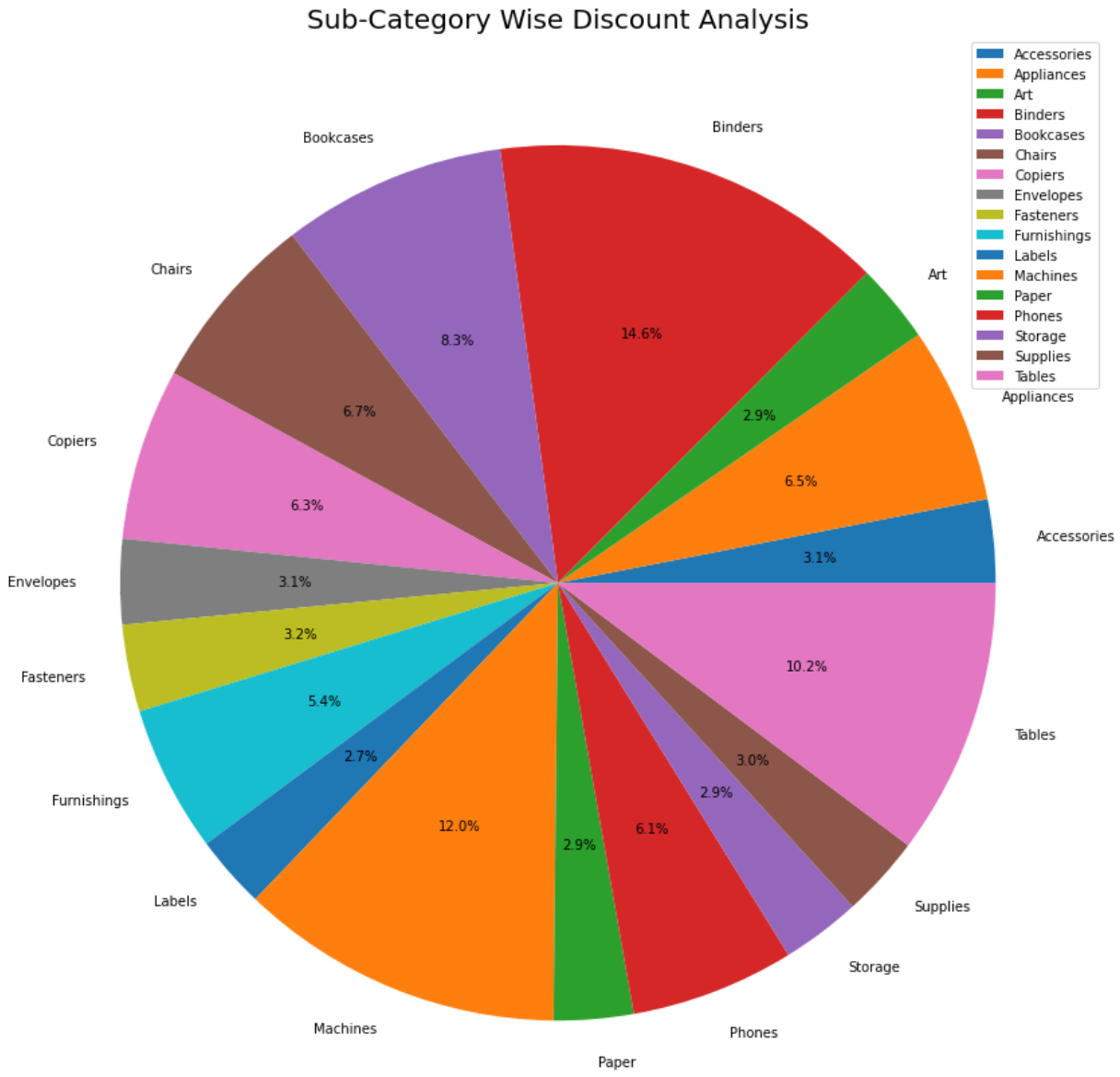


Copier and Machine have High sales.

---

[2] BASED ON THE DISCOUNT

In [51]:
```
plt.figure(figsize = (15,15))
plt.pie(df_sub_category['Discount'], labels = df_sub_category.index, autopct = '%1.1
plt.title('Sub-Category Wise Discount Analysis', fontsize = 20)
plt.legend()
plt.xticks(rotation = 90)
plt.show()
```



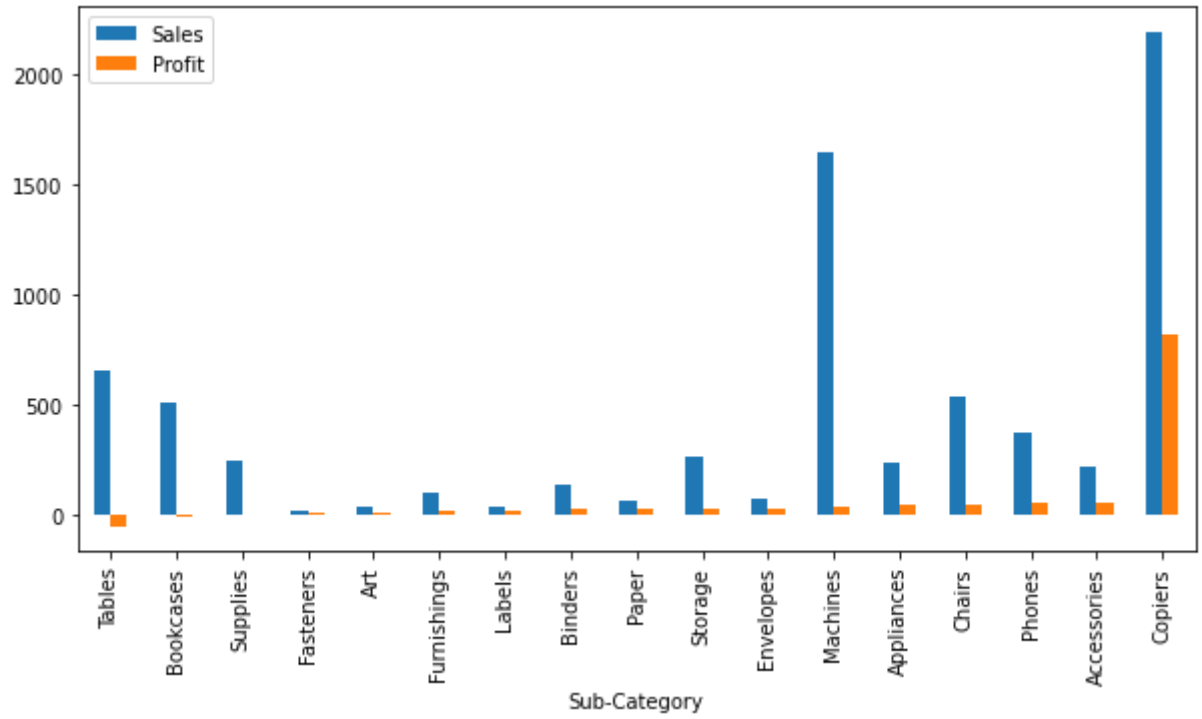
Binders, Machines and Tables have high Discount

---

[3] BASED ON THE PROFIT

In [52]:
```
df_sub_category.sort_values('Profit')[['Sales','Profit']].plot(kind='bar',
                                                               figsize= (10,5),
                                                               label=['Avg Sales Pric
```

Out[52]: <AxesSubplot:xlabel='Sub-Category'>



COPIER has the Highest Profit as Well as Sell

REGION WISE ANALYSIS

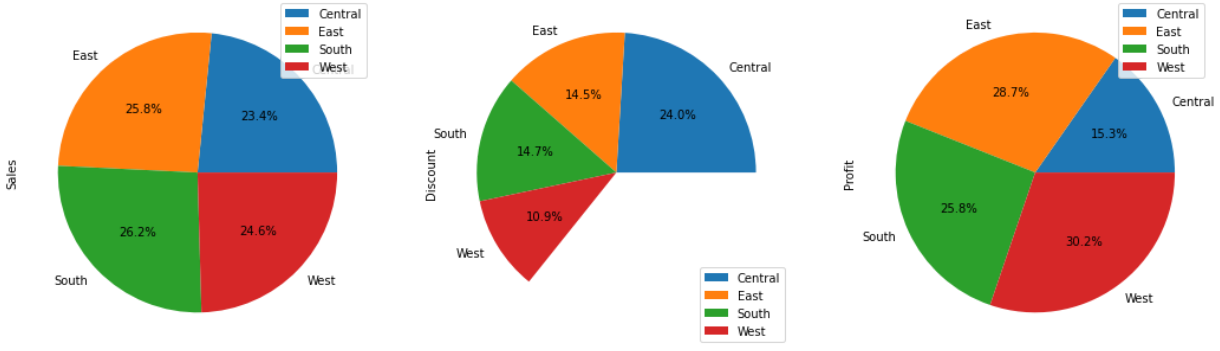
In [53]:
```
df_region = df.groupby(['Region'])[['Sales', 'Discount', 'Profit']].mean()
df_region
```

Out[53]:

Region	Sales	Discount	Profit
Central	215.772661	0.240353	17.092709
East	238.336110	0.145365	32.135808
South	241.803645	0.147253	28.857673
West	226.493233	0.109335	33.849032

In [54]:
```
df_region.plot.pie(subplots=True,
                   figsize=(18, 20),
                   autopct='%1.1f%%',
                   labels = df_region.index)
```

Out[54]:
```
array([<AxesSubplot:ylabel='Sales'>, <AxesSubplot:ylabel='Discount'>,
       <AxesSubplot:ylabel='Profit'>], dtype=object)
```



WEST has High Profit

############### RESULT AND CONCLUSION

Profit is more than that of sale but there are some areas where profit could be increased.

Profit and Discount is high in First Class

Sales is high for Same day ship

Sub-category: Copier: High Profit & sales

Sub-category: Binders , Machines and then tables have high Discount.

Category: Maximun sales and Profit obtain in Technology.

Category: Minimun profit obtain in Furniture

State: Vermont: Highest Profit

State: Ohio: Lowest Profit

Segment: Home-office: High Profit & sales

Here is top 3 city where deals are Highest.

[1] York City

[2] Los Angeles

[3] Philadelphia

Sales and Profit are Moderately Correlated.

Quantity and Profit are less Moderately Correlated.

Discount and Profit are Negatively Correlated

Here is top 3 state where deals are Highest.

[1] Califonia

[2] New York

[3] Texas

Wyoming : Lowest Number of deal,Highest amount of sales= Wyoming(11.8%)

Lowest amount of sales= South Dakota(0.8%)

########################## THANK YOU ##########################