Author: ATANU DAS

The Sparks Foundation

Task #3 : Exploratory data analysis on the dataset 'SampleSuperstore'

In [1]:
```
#Importing modules
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

In [3]:
```
#reading the csv file provided
data = pd.read_csv(r"C:\Users\ATANU\Desktop\spark foundation\SampleSuperstore.csv")
```

In [4]:
```
#getting basic statistical overview about the data provided
data.describe()
```

Out[4]:

	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

	Postal Code	Sales	Quantity	Discount	Profit
max	99301.000000	22638.480000	14.000000	0.800000	8399.976000

In [5]:
```
#first 5 rows
data.head()
```

Out[5]:

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Sub-Category	Sales	Quantity	Discount	Profit
0	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Bookcases	261.9600	2	0.00	41.9136
1	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Chairs	731.9400	3	0.00	219.5820
2	Second Class	Corporate	United States	Los Angeles	California	90036	West	Office Supplies	Labels	14.6200	2	0.00	6.8714
3	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Furniture	Tables	957.5775	5	0.45	-383.0310
4	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Office Supplies	Storage	22.3680	2	0.20	2.5164

In [6]:
```
#getting the number of rows and columns in the provided dataset
data.shape
```

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

In [7]:
```
#comlumns in dataset
data.columns
```

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

In [8]:
```
#information about the data
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 [9]:
```
#checking for null values
data.isnull().sum()
```

Out[9]:
```
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 [10]:
```
# calculating the investment
data["investment"]=data["Sales"]-data["Profit"]
```

In [11]:
```
data.head()
```

Out[11]:

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Sub-Category	Sales	Quantity	Discount	Profit	investme
0	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Bookcases	261.9600	2	0.00	41.9136	220.04
1	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Chairs	731.9400	3	0.00	219.5820	512.35
2	Second Class	Corporate	United States	Los Angeles	California	90036	West	Office Supplies	Labels	14.6200	2	0.00	6.8714	7.74
3	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Furniture	Tables	957.5775	5	0.45	-383.0310	1340.60
4	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Office Supplies	Storage	22.3680	2	0.20	2.5164	19.85

In [12]:

```
#Calculating Profit Percentage
data["Profpercent"]= (data["Profit"]/data["investment"])*100
```

In [13]:

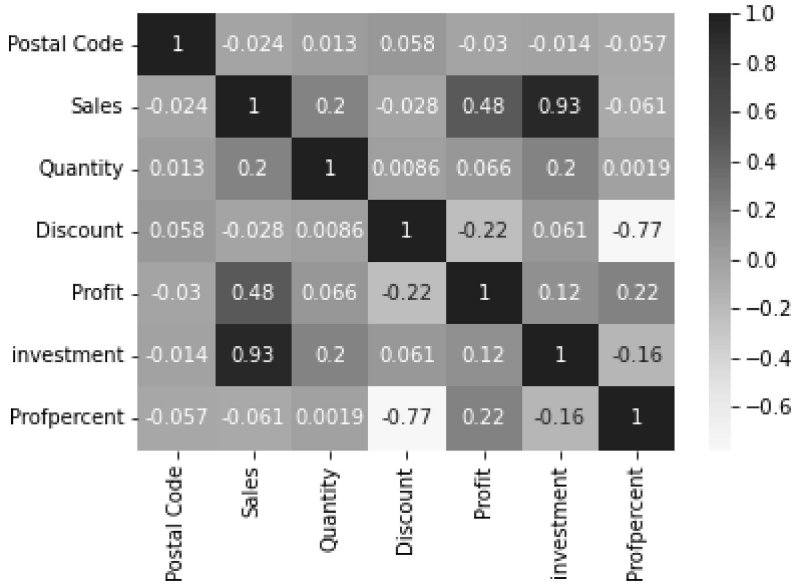
```
#checking first 5 rows again
data.head()
```

Out[13]:

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Sub-Category	Sales	Quantity	Discount	Profit	investme
0	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Bookcases	261.9600	2	0.00	41.9136	220.04
1	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Chairs	731.9400	3	0.00	219.5820	512.35
2	Second Class	Corporate	United States	Los Angeles	California	90036	West	Office Supplies	Labels	14.6200	2	0.00	6.8714	7.74
3	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Furniture	Tables	957.5775	5	0.45	-383.0310	1340.60
4	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Office Supplies	Storage	22.3680	2	0.20	2.5164	19.85

In [14]:
```
#to find correlation between entities
corr = data.corr()
sns.heatmap(corr,annot=True,cmap='Reds')
```

Out[14]: <AxesSubplot:>

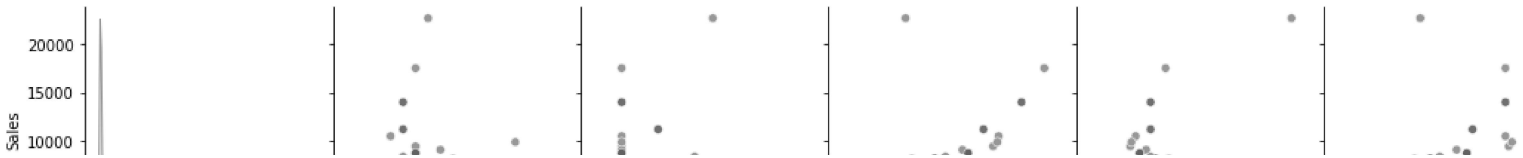


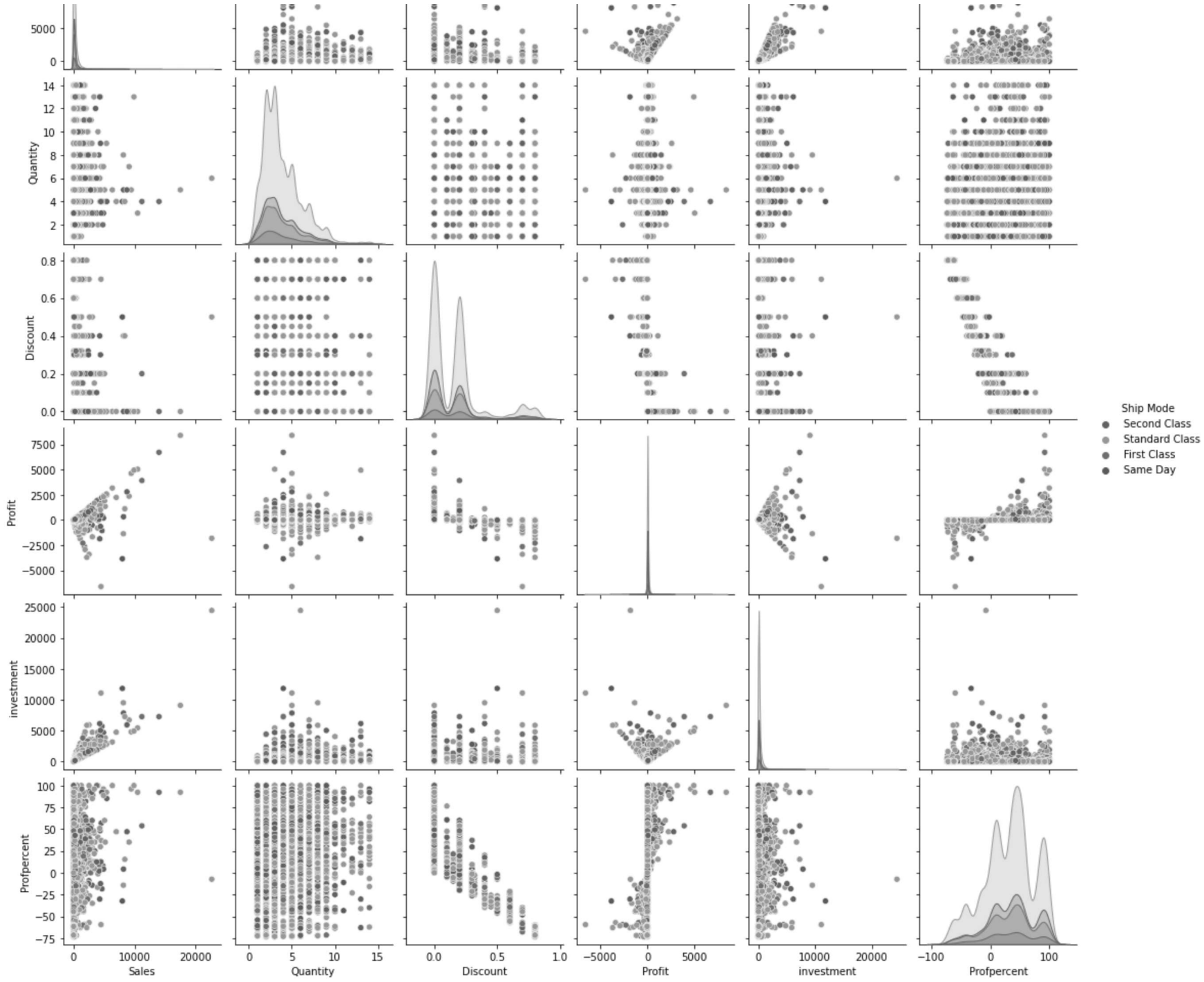
In [15]:
```
#dropping postal code columns as not useful
data = data.drop(['Postal Code'],axis = 1)
```

Shipping mode

In [16]:
```
sns.pairplot(data, hue = 'Ship Mode')
```

Out[16]: <seaborn.axisgrid.PairGrid at 0x14f312b7e80>



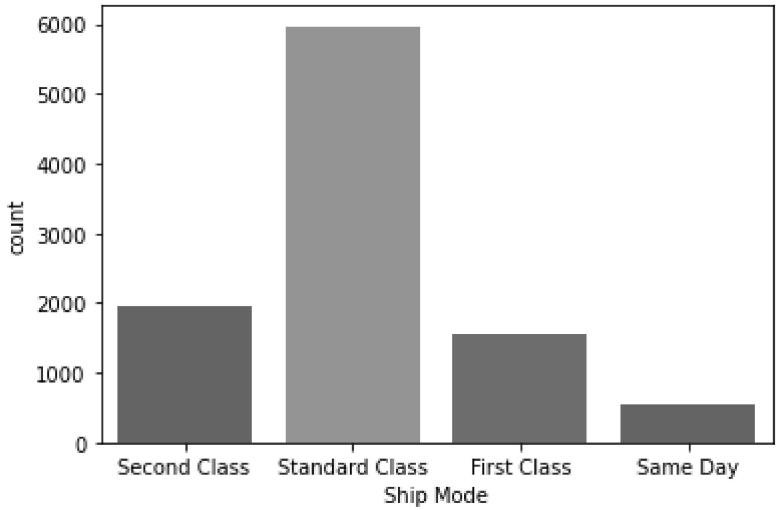

Sales
Quantity
Discount
Profit
investment
Profpercent
Ship Mode
Second Class
Standard Class
First Class
Same Day

In [17]:
```
data['Ship Mode'].value_counts()
```

Out[17]:
```
Standard Class    5968
Second Class      1945
First Class       1538
Same Day           543
Name: Ship Mode, dtype: int64
```

In [18]:
```
sns.countplot(x=data['Ship Mode'])
```

Out[18]:
```
<AxesSubplot:xlabel='Ship Mode', ylabel='count'>
```



Segments of customer

In [19]:
```
segmenttype=data.groupby("Segment")
for i,df in segmenttype:
    print(i)
```

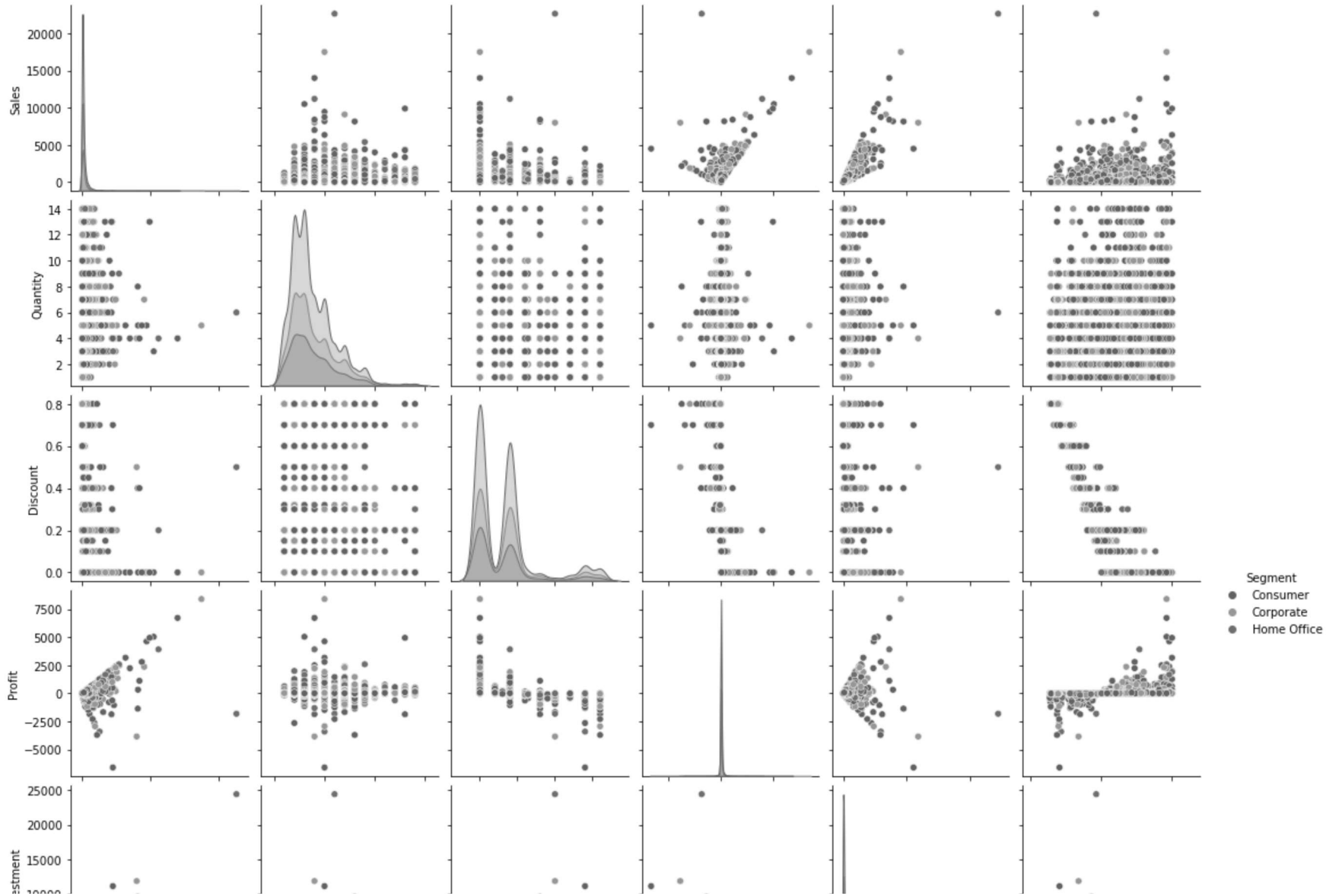
```
Consumer
Corporate
Home Office
```

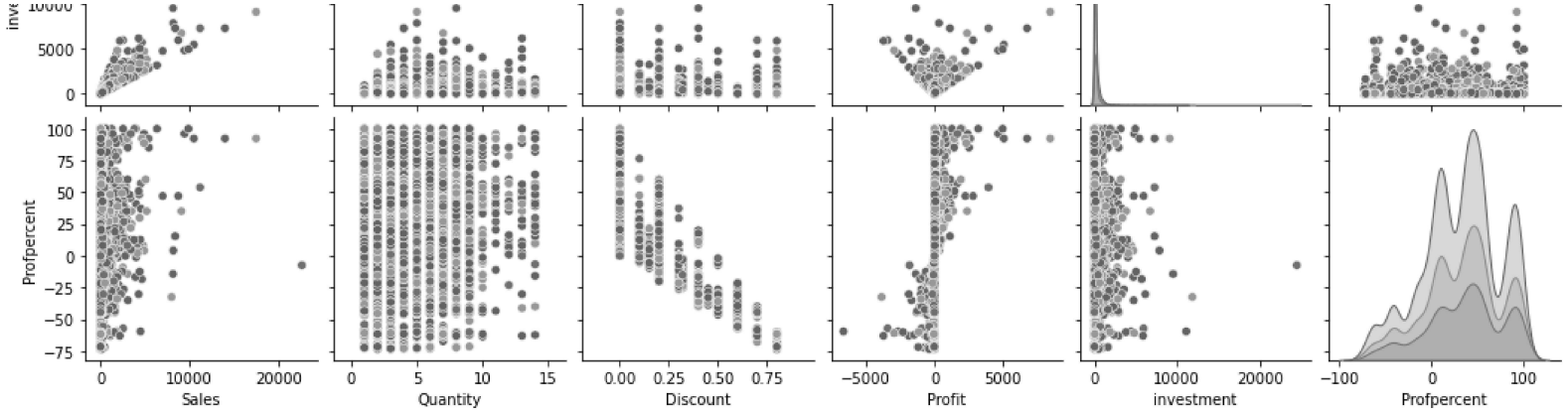
In [20]:
```
data["Segment"].value_counts()
```

Out[20]:
```
Consumer       5191
Corporate      3020
Home Office    1783
Name: Segment, dtype: int64
```

In [21]:
```
sns.pairplot(data,hue = 'Segment')
```

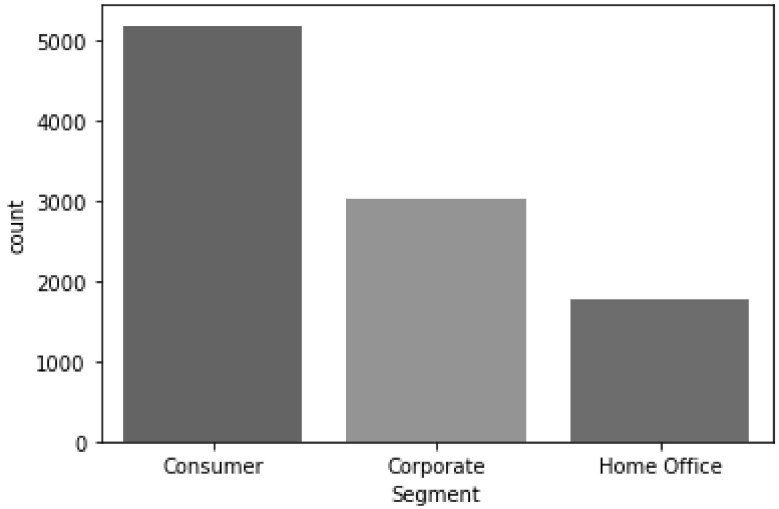
Out[21]:
```
<seaborn.axisgrid.PairGrid at 0x14f31aa75b0>
```





In [22]:
```
sns.countplot(x = 'Segment',data = data)
```

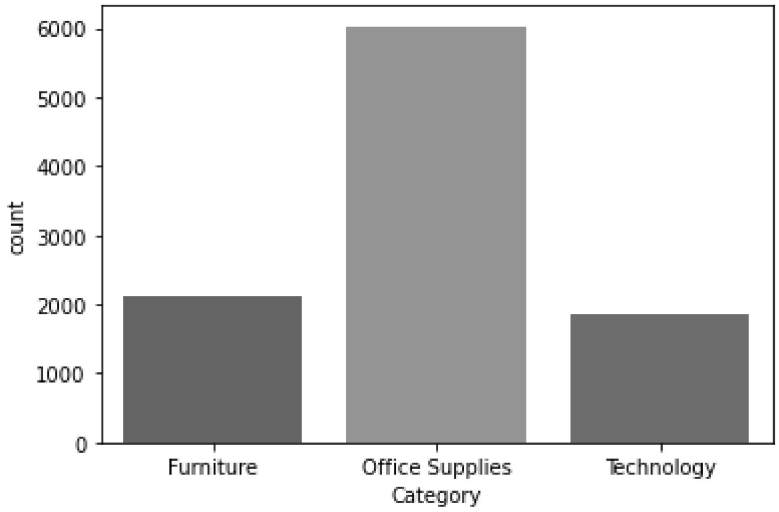
Out[22]:
```
<AxesSubplot:xlabel='Segment', ylabel='count'>
```



Category analysis

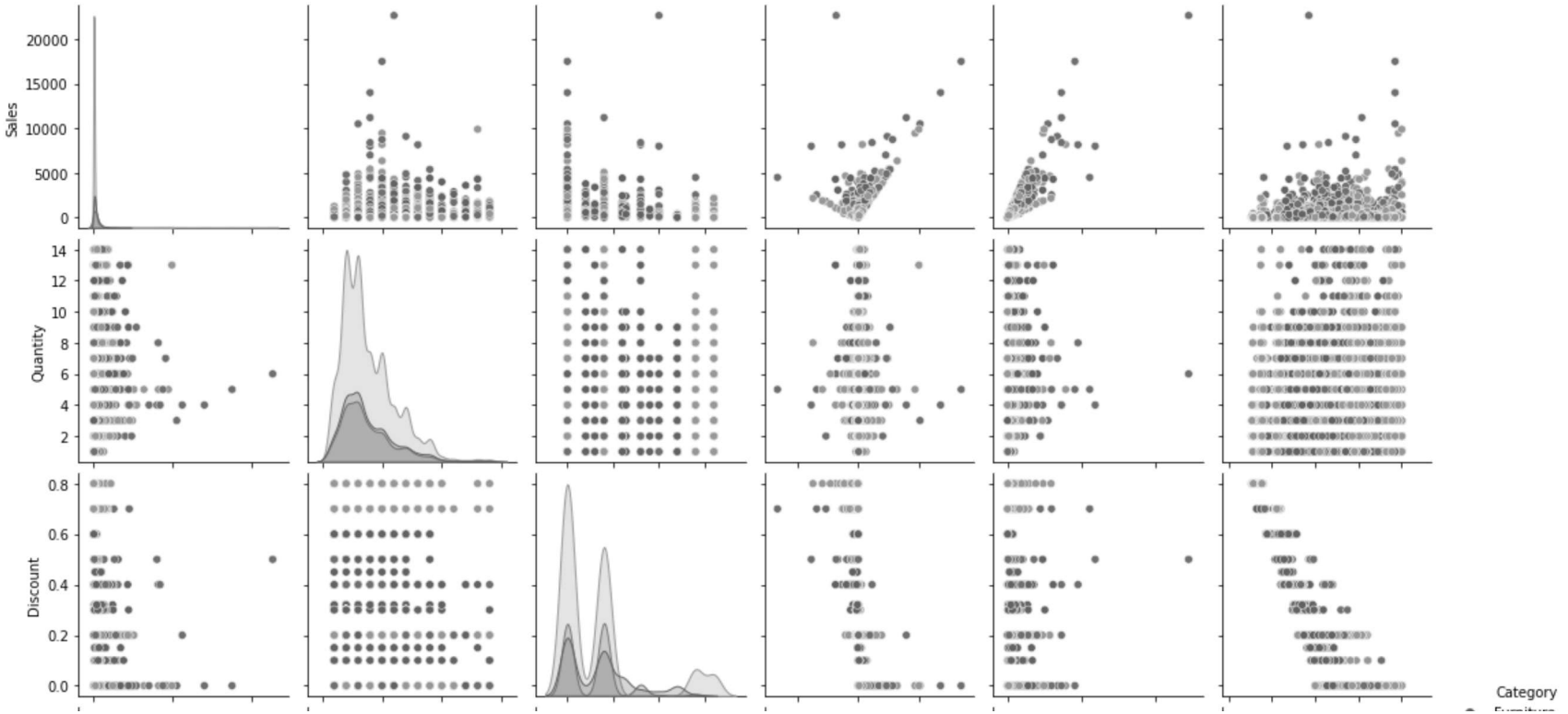
In [23]:
```
sns.countplot(x='Category',data=data)
```

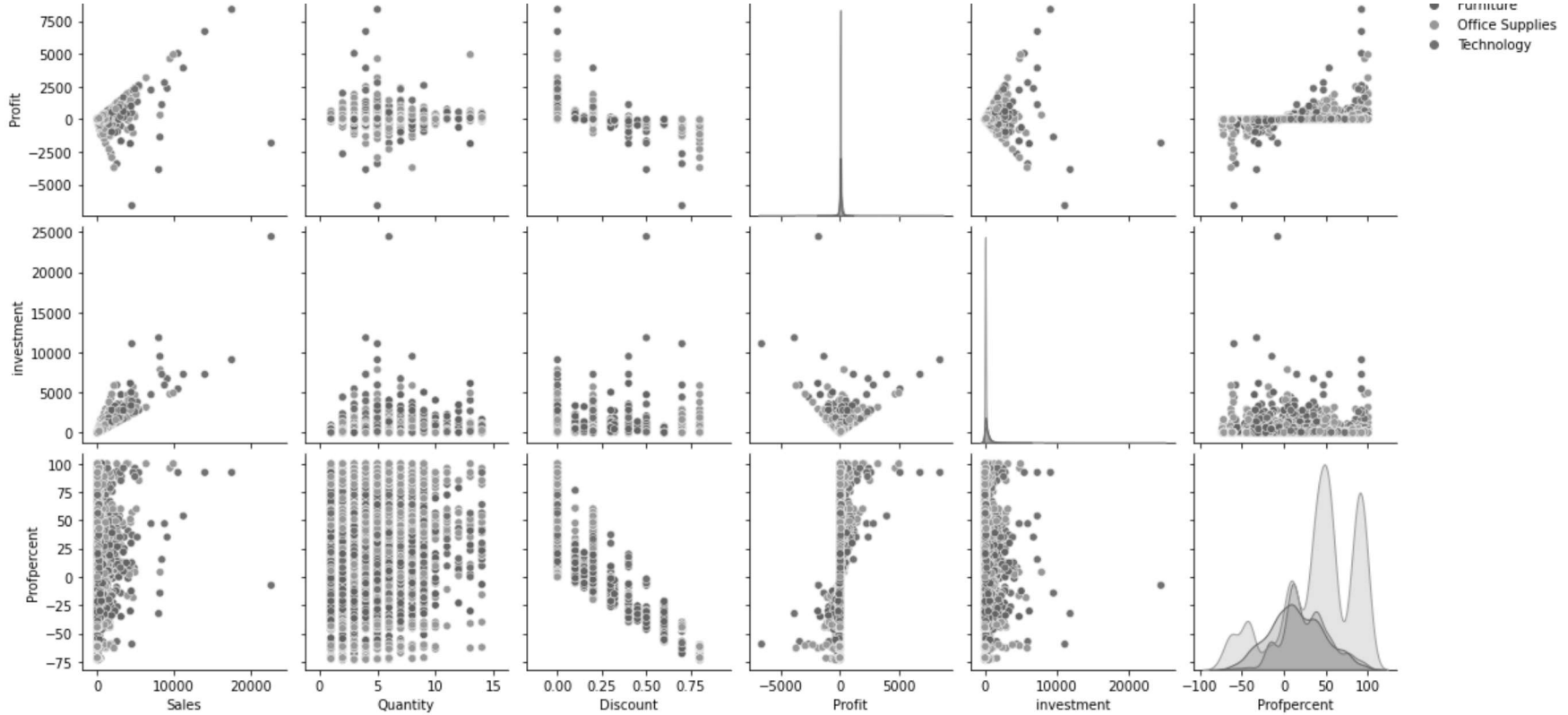
Out[23]:
```
<AxesSubplot:xlabel='Category', ylabel='count'>
```



In [24]:
```
sns.pairplot(data,hue='Category')
```

Out[24]:
```
<seaborn.axisgrid.PairGrid at 0x14f397e2f10>
```

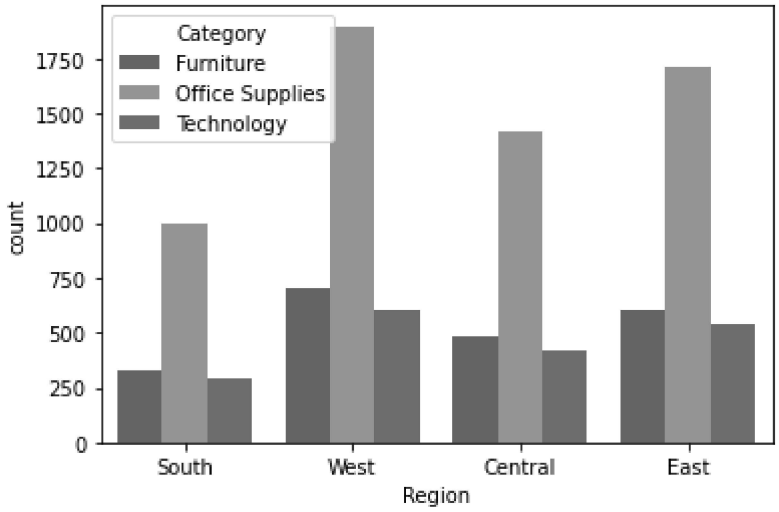




Regional Analysis

In [25]:
```
sns.countplot(x=data['Region'], hue=data['Category'])
```

Out[25]:
```
<AxesSubplot:xlabel='Region', ylabel='count'>
```

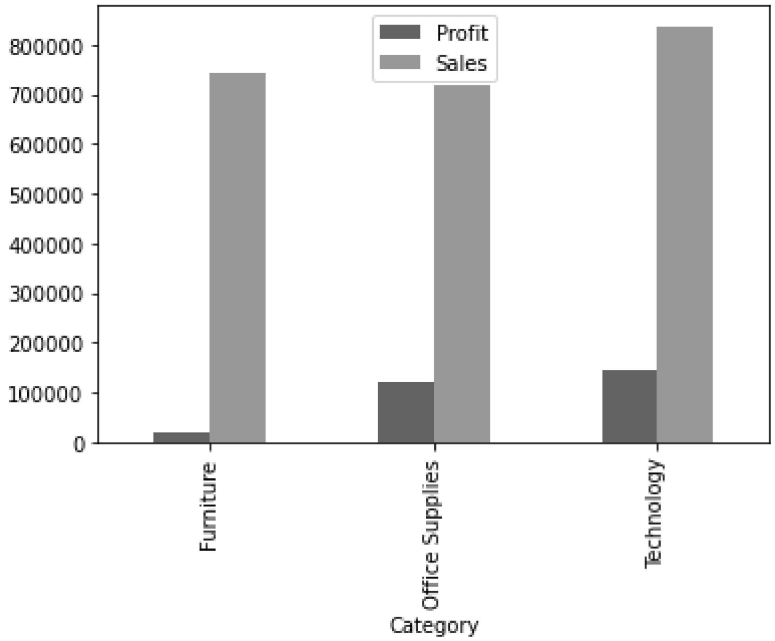


Category profit and sales

In [26]:
```
ds=data.groupby('Category')['Profit','Sales'].agg('sum')
print(ds)
ds.plot.bar()
```

```
                    Profit        Sales
Category
Furniture         18451.2728  741999.7953
Office Supplies  122490.8008  719047.0320
Technology       145454.9481  836154.0330
C:\Users\ATANU\AppData\Local\Temp/ipykernel_6792/1756076298.py:1: FutureWarning: Indexing with multiple keys (implicitly converted to a tuple of keys) will be deprecated, use a list instead.
  ds=data.groupby('Category')['Profit','Sales'].agg('sum')
```

Out[26]: <AxesSubplot:xlabel='Category'>



Subcategory Analysis

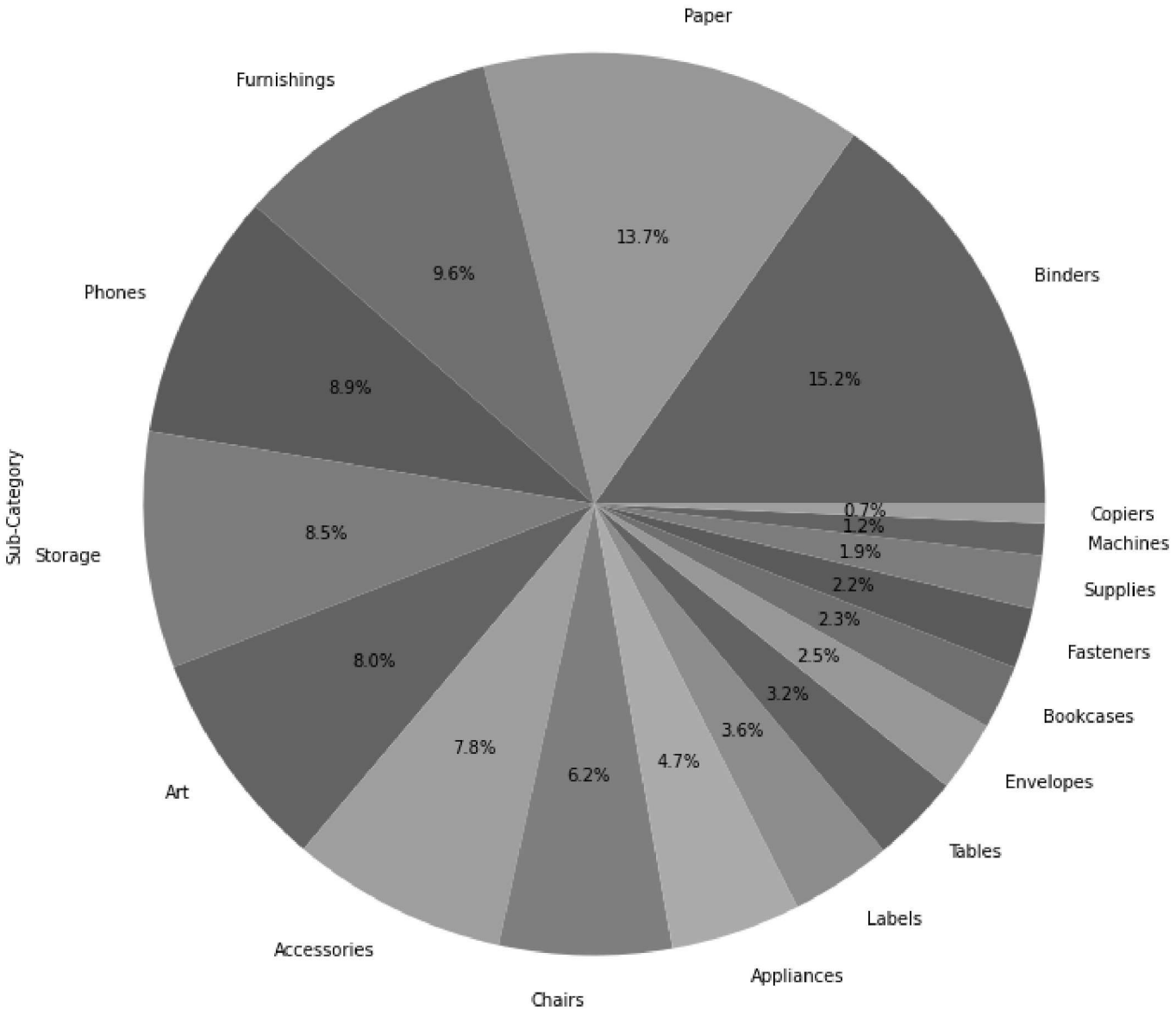
In [28]:
```
data['Sub-Category'].value_counts()
```

Out[28]:
```
Binders        1523
Paper          1370
Furnishings     957
Phones          889
Storage         846
Art             796
Accessories     775
Chairs          617
Appliances      466
Labels          364
Tables          319
Envelopes       254
Bookcases       228
Fasteners       217
Supplies        190
Machines        115
```

```
Copiers          68
Name: Sub-Category, dtype: int64
```

In [29]:

```
plt.figure(figsize=(15,12))
data['Sub-Category'].value_counts().plot.pie(autopct='%1.1f%%')
plt.show()
```

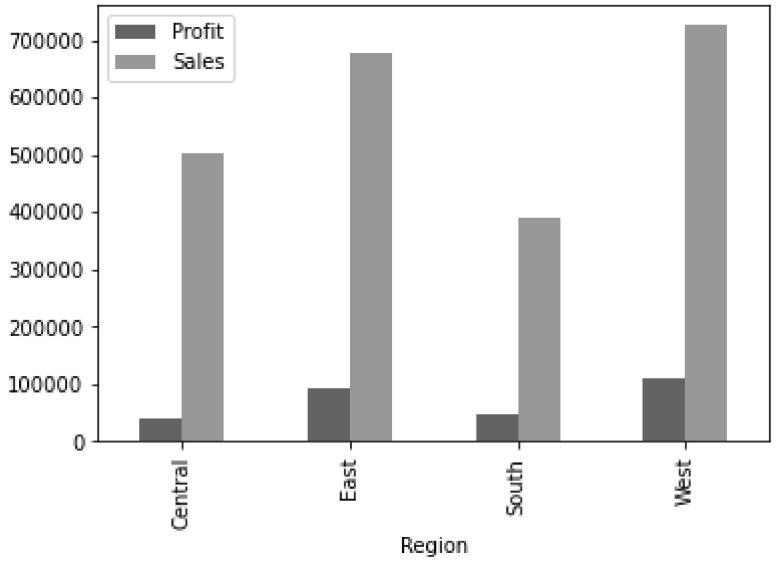


Regional Analysis

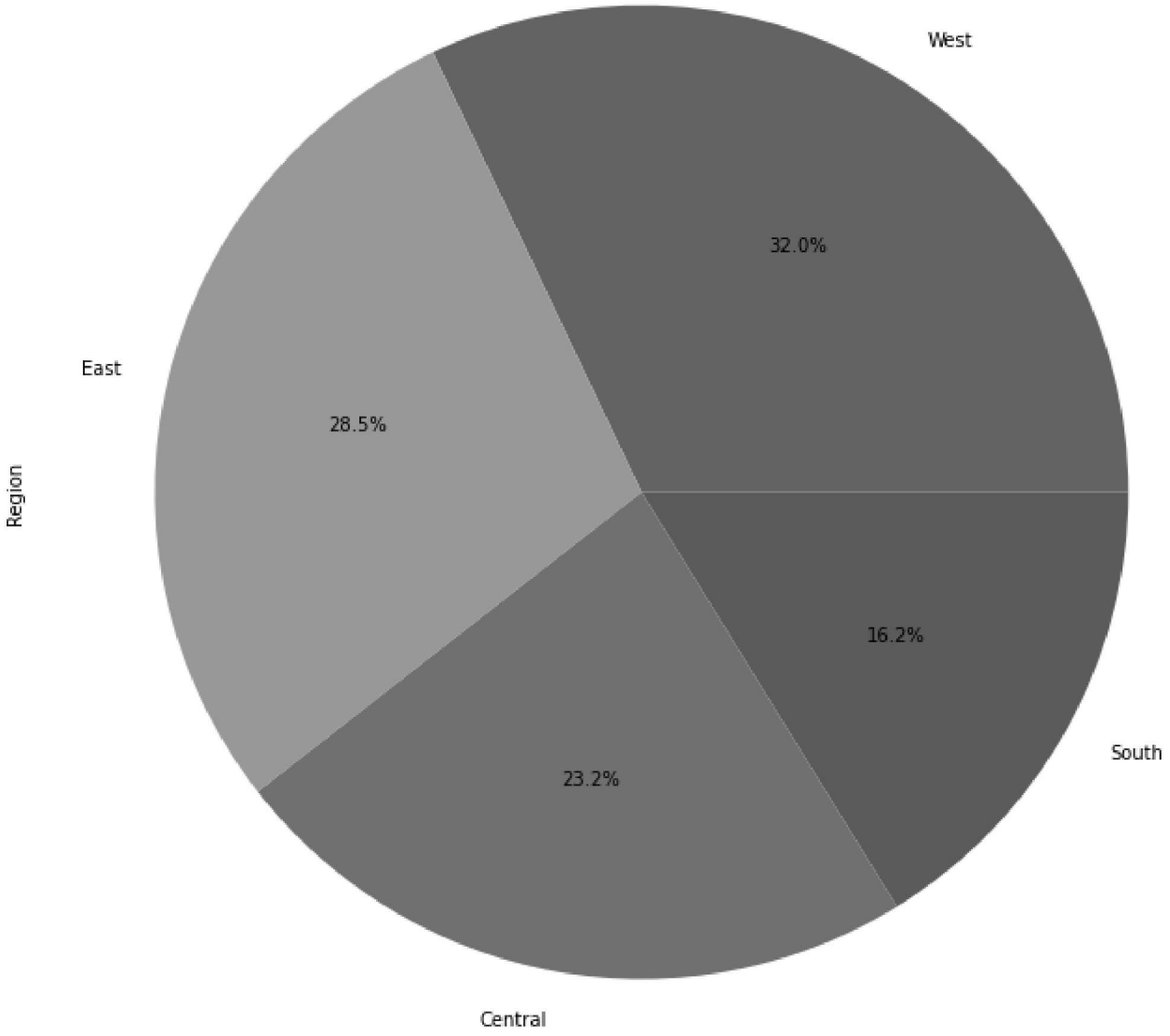
In [30]:
```
rg=data.groupby('Region')['Profit','Sales'].agg('sum')
rg.plot.bar()
```

```
C:\Users\ATANU\AppData\Local\Temp/ipykernel_6792/1781158339.py:1: FutureWarning: Indexing with multiple keys (implicitly
converted to a tuple of keys) will be deprecated, use a list instead.
  rg=data.groupby('Region')['Profit','Sales'].agg('sum')
```

Out[30]:
```
<AxesSubplot:xlabel='Region'>
```



In [31]:
```
plt.figure(figsize=(15,12))
data['Region'].value_counts().plot.pie(autopct='%1.1f%%')
plt.show()
```



State wise Sales

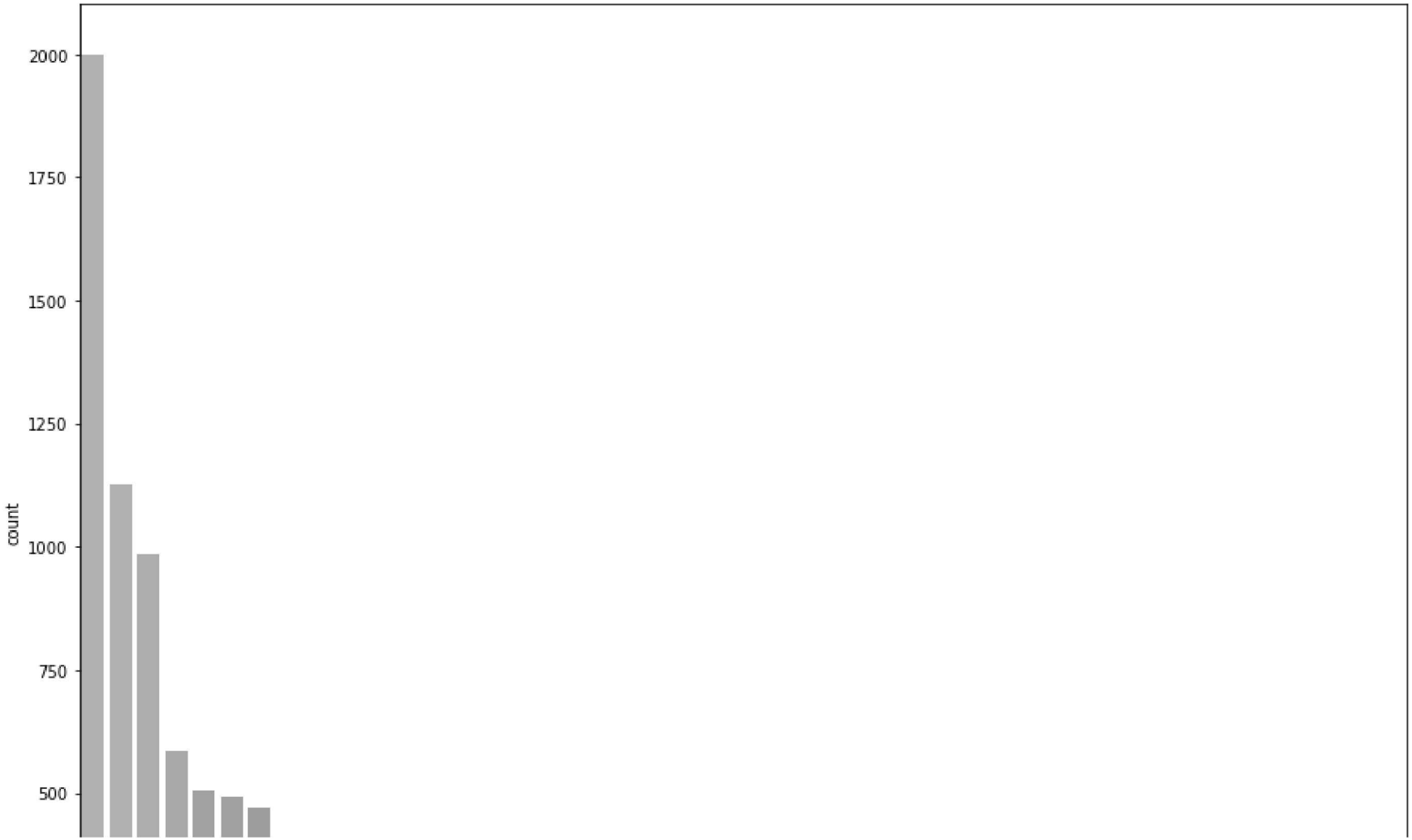
In [33]:
```
data['State'].value_counts()
```

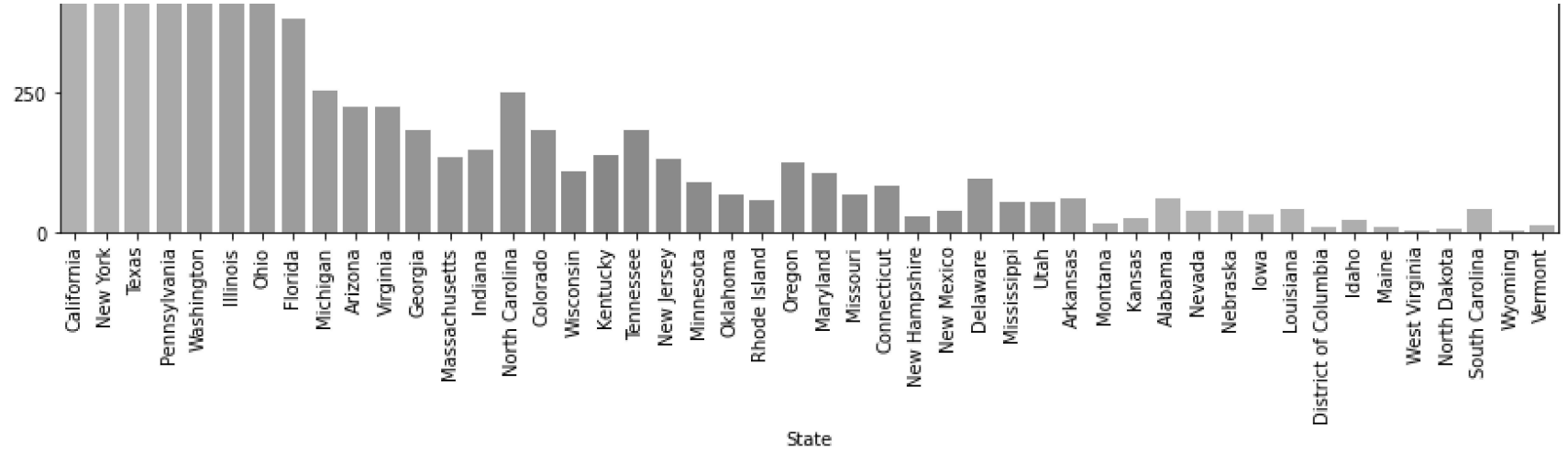
Out[33]:
```
California              2001
New York                1128
Texas                    985
Pennsylvania             587
Washington               506
Illinois                 492
Ohio                     469
Florida                  383
Michigan                 255
North Carolina           249
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 [34]:

```
plt.figure(figsize=(15,12))
sns.countplot(x='State',data=data,order=df['State'].value_counts().index)
plt.xticks(rotation=90)
plt.show()
```





Observations and Conclusions 1.) Maximum Sales are from selling Binders (15.2%), Paper(13.7%), Furnishing(9.6%) and Minimum are from Copiers(0.7%), Machines(1.2%) etc.

2.) California, Newyork and Texas provide most sales out of all states.

3.) West Regions provides the most sales (32%)

4.) Profit Percentage and Discount have a negative Correlation.

5.) Maximum profits are derived from west and east branch.

6.) Category wise Technology provides the most sales and profit.

7.) Despite having more sales than office supplies Furniture is lacking in profit due to lesser margins thus the profit margin on furniture should be increased.

In [ ]: