In [2]:
```
import pandas as pd
import numpy as np
import seaborn as sns
from matplotlib import pyplot as plt
from pandas.core.computation.check import NUMEXPR_INSTALLED
%matplotlib inline
```

In [3]:
```
import warnings
warnings.filterwarnings('ignore')
```

In [4]:
```
df = pd.read_csv('C:/Users/DELL/Downloads/SampleSuperstore.csv')  #loading dat
df.head()    #display top 5 rows
```

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	261
1	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Chairs	731
2	Second Class	Corporate	United States	Los Angeles	California	90036	West	Office Supplies	Labels	14
3	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Furniture	Tables	957
4	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Office Supplies	Storage	22

In [5]:
```
df.tail()
```

Out[5]:

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Sub Categor
9989	Second Class	Consumer	United States	Miami	Florida	33180	South	Furniture	Furnishing
9990	Standard Class	Consumer	United States	Costa Mesa	California	92627	West	Furniture	Furnishing
9991	Standard Class	Consumer	United States	Costa Mesa	California	92627	West	Technology	Phone
9992	Standard Class	Consumer	United States	Costa Mesa	California	92627	West	Office Supplies	Pape
9993	Second Class	Consumer	United States	Westminster	California	92683	West	Office Supplies	Appliance

In [6]:
```
df.shape
```

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

In [7]:
```
df.describe()
```

Out[7]:

	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 [8]:
```
df.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]:
```
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 [11]:
```
df.columns
```

Out[11]:
```
Index(['Ship Mode', 'Segment', 'Country', 'City', 'State', 'Postal Code',
       'Region', 'Category', 'Sub-Category', 'Sales', 'Quantity', 'Discount',
       'Profit'],
      dtype='object')
```

In [12]:
```
df.duplicated().sum()
```

Out[12]:
```
17
```

In [13]:
```
df.nunique()
```

Out[13]:
```
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 [14]:
```
df['Postal Code'] = df['Postal Code'].astype('object')
```

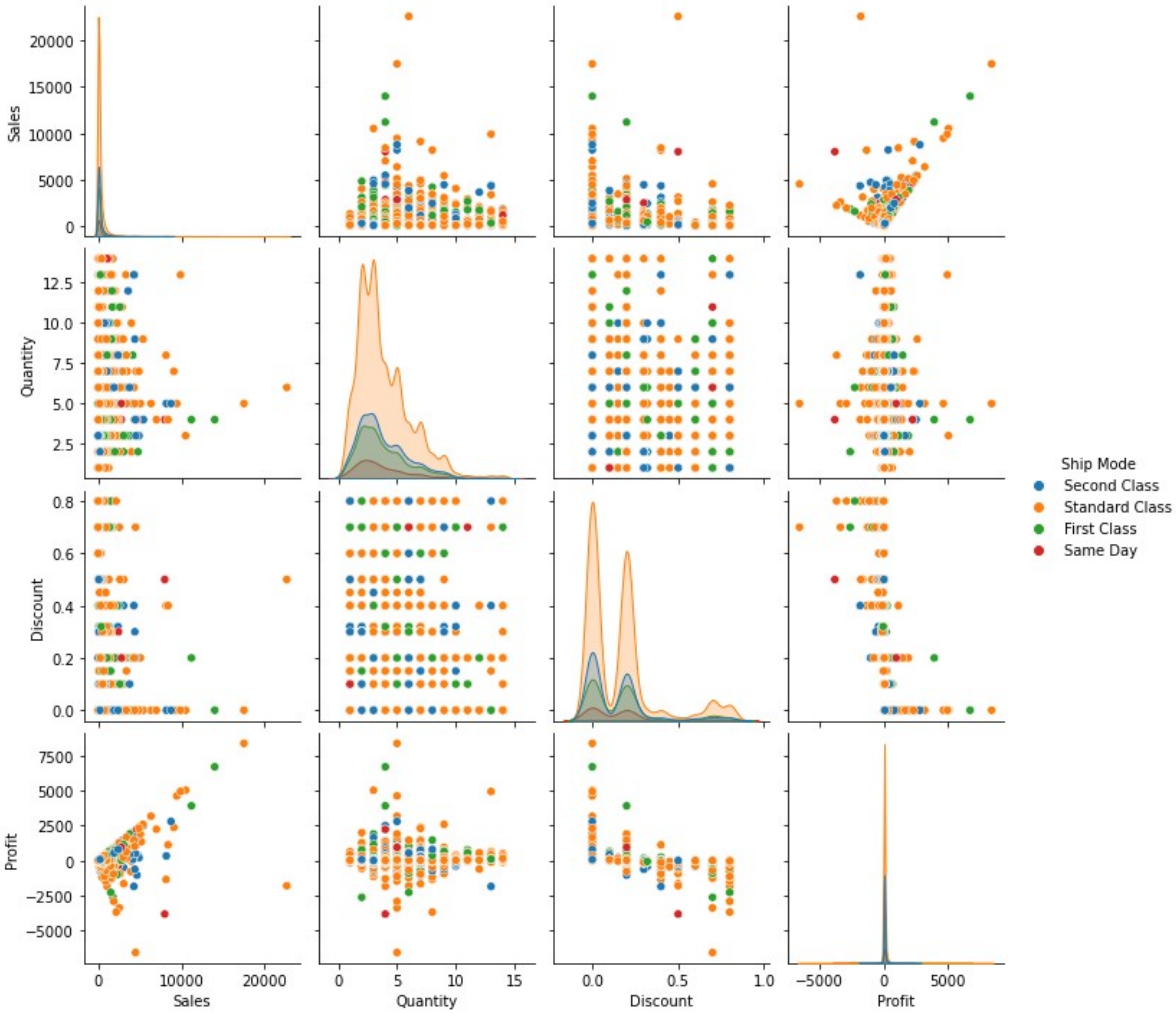
In [15]:
```
df.drop_duplicates(subset=None,keep='first',inplace=True)
df.duplicated().sum()
```

Out[15]:
```
0
```

In [18]:
```
df = df.drop(['Postal Code'],axis = 1)
```

In [19]:
```
sns.pairplot(df, hue = 'Ship Mode')
```

Out[19]:
```
<seaborn.axisgrid.PairGrid at 0x29b5b7773d0>
```

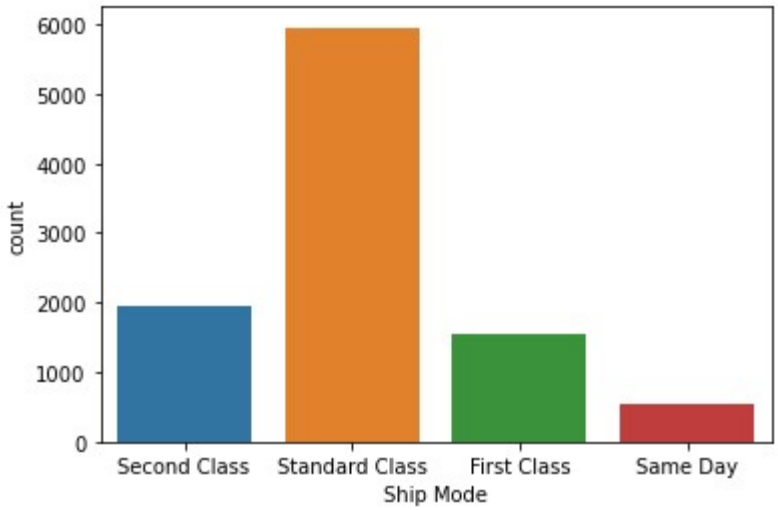


In [20]:
```
df['Ship Mode'].value_counts()
```

Out[20]:
```
Ship Mode
Standard Class    5955
Second Class      1943
First Class       1537
Same Day           542
Name: count, dtype: int64
```

In [21]:
```
sns.countplot(x=df['Ship Mode'])
```

Out[21]:
```
<AxesSubplot:xlabel='Ship Mode', ylabel='count'>
```

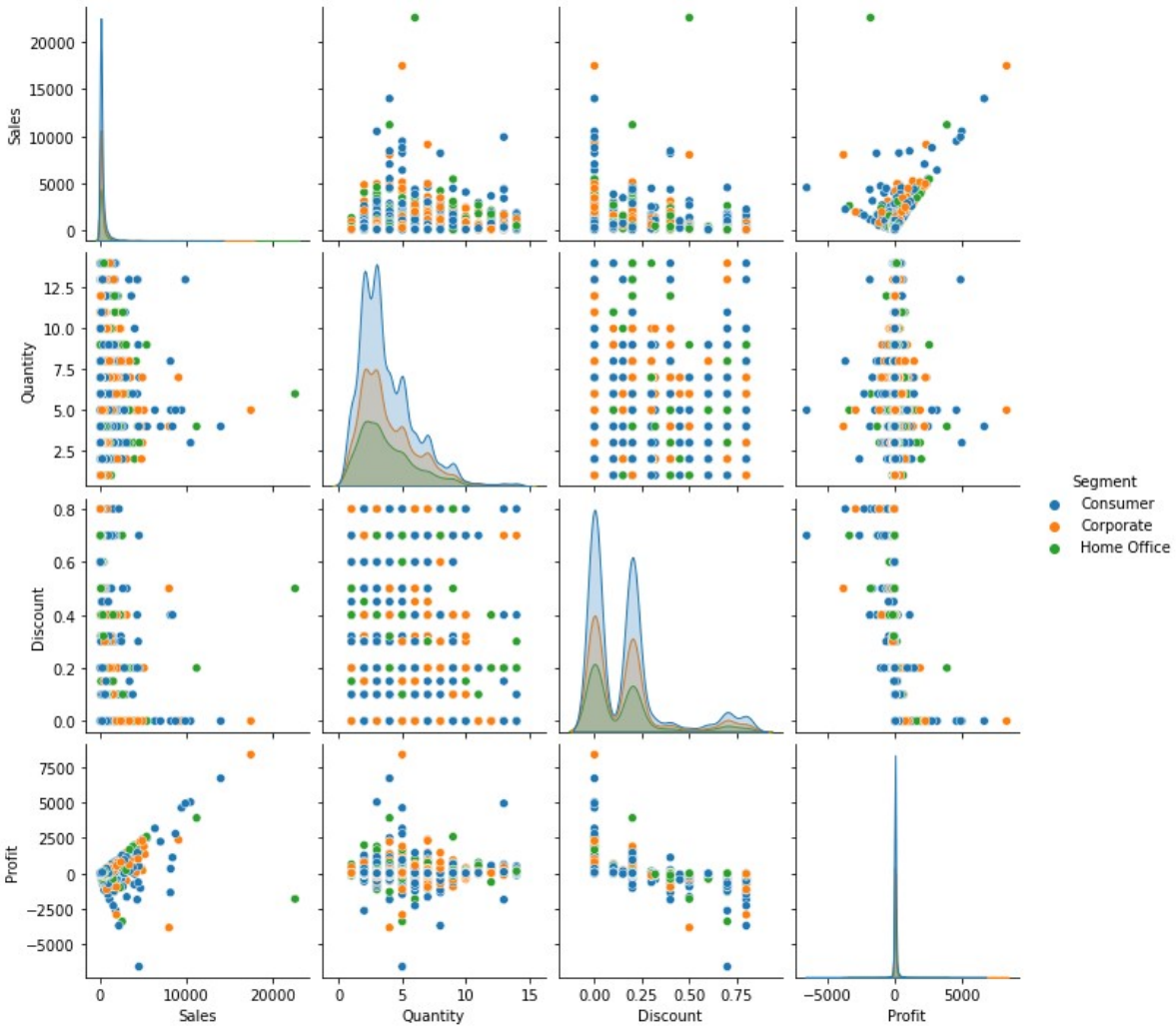


In [22]:
```
df['Segment'].value_counts()
```

Out[22]:
```
Segment
Consumer       5183
Corporate      3015
Home Office    1779
Name: count, dtype: int64
```

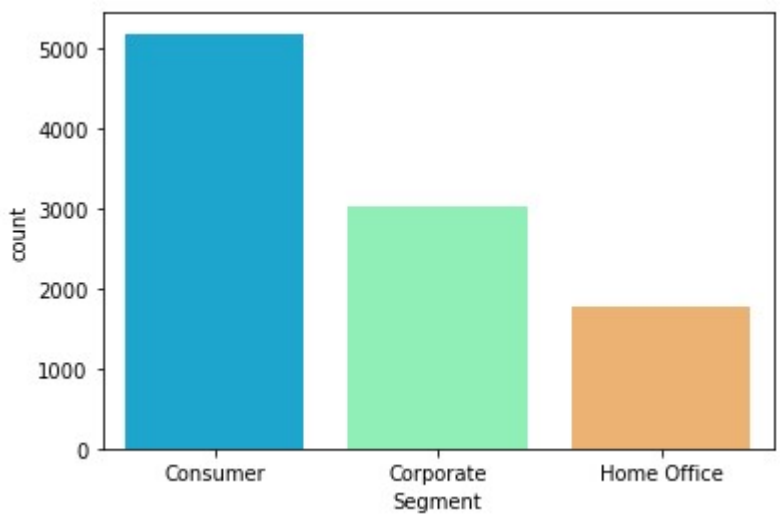
In [23]:
```
sns.pairplot(df,hue = 'Segment')
```

Out[23]:
```
<seaborn.axisgrid.PairGrid at 0x29b726dee20>
```



In [24]:
```
sns.countplot(x = 'Segment',data = df, palette = 'rainbow')
```

Out[24]:
```
<AxesSubplot:xlabel='Segment', ylabel='count'>
```

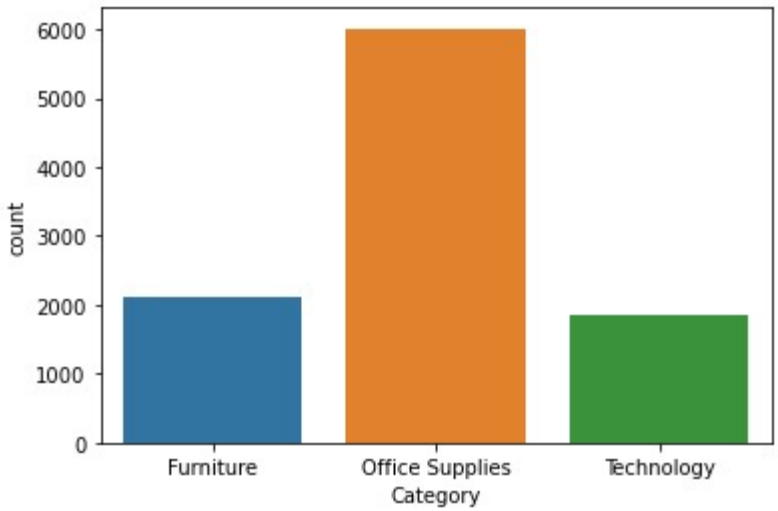


In [25]:
```
df['Category'].value_counts()
```

Out[25]:
```
Category
Office Supplies    6012
Furniture          2118
Technology         1847
Name: count, dtype: int64
```

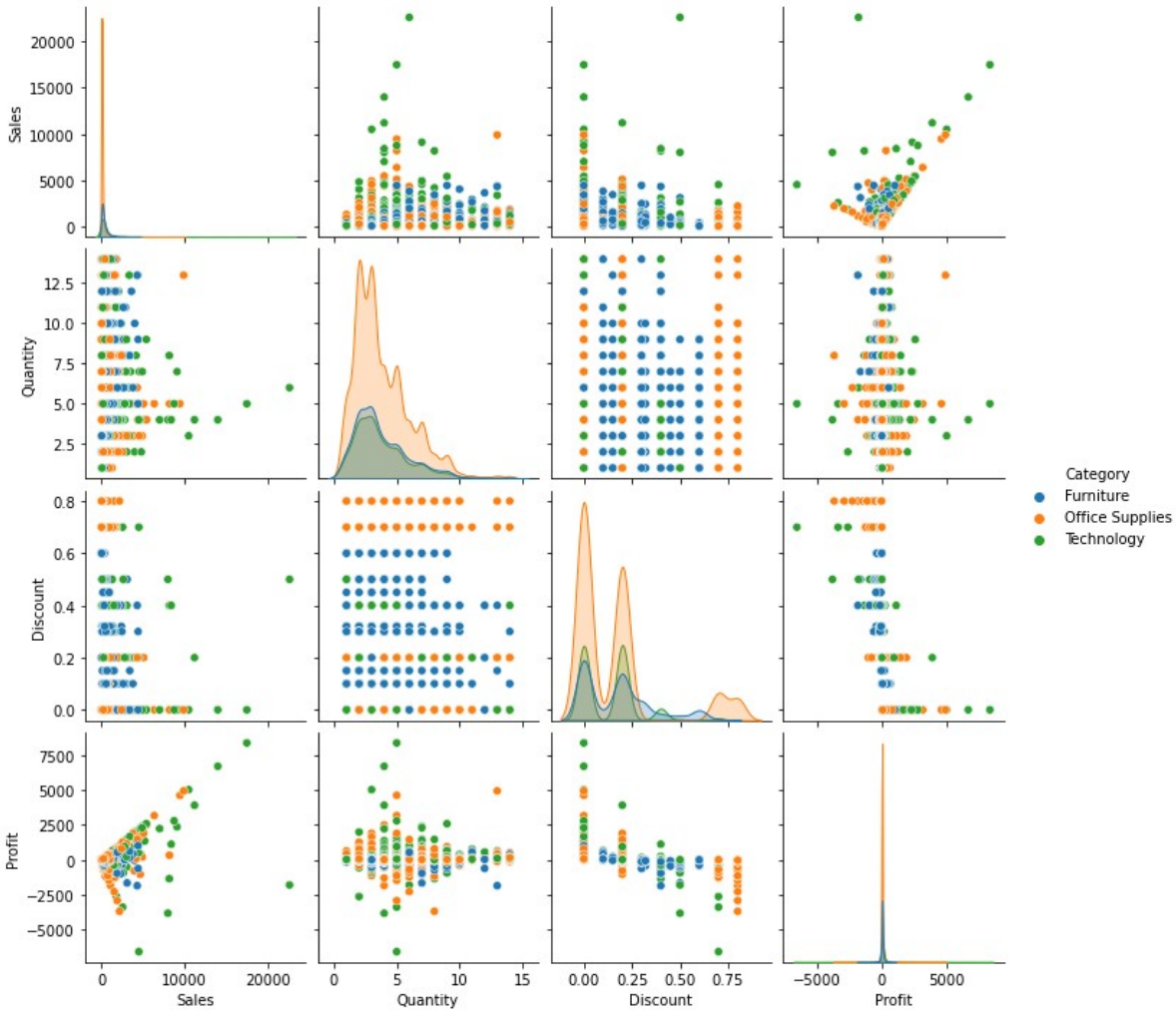
In [26]:
```
sns.countplot(x='Category',data=df,palette='tab10')
```

Out[26]:
```
<AxesSubplot:xlabel='Category', ylabel='count'>
```



In [27]:
```
sns.pairplot(df,hue='Category')
```

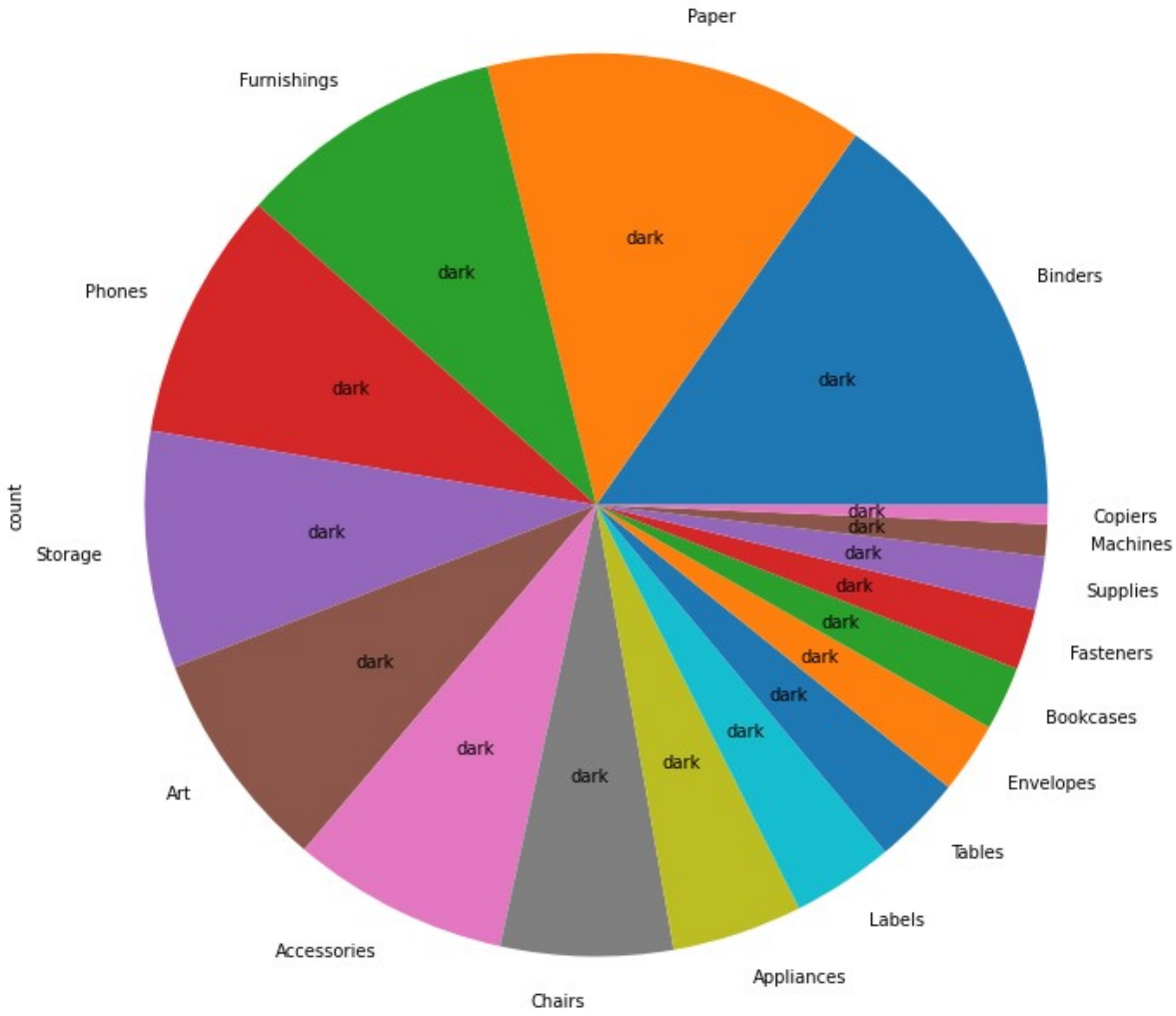
Out[27]:
```
<seaborn.axisgrid.PairGrid at 0x29b741c6100>
```



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

Out[28]:
```
Sub-Category
Binders        1522
Paper          1359
Furnishings     956
Phones          889
Storage         846
Art             795
Accessories     775
Chairs          615
Appliances      466
Labels          363
Tables          319
Envelopes       254
Bookcases       228
Fasteners       217
Supplies        190
Machines        115
Copiers          68
Name: count, dtype: int64
```

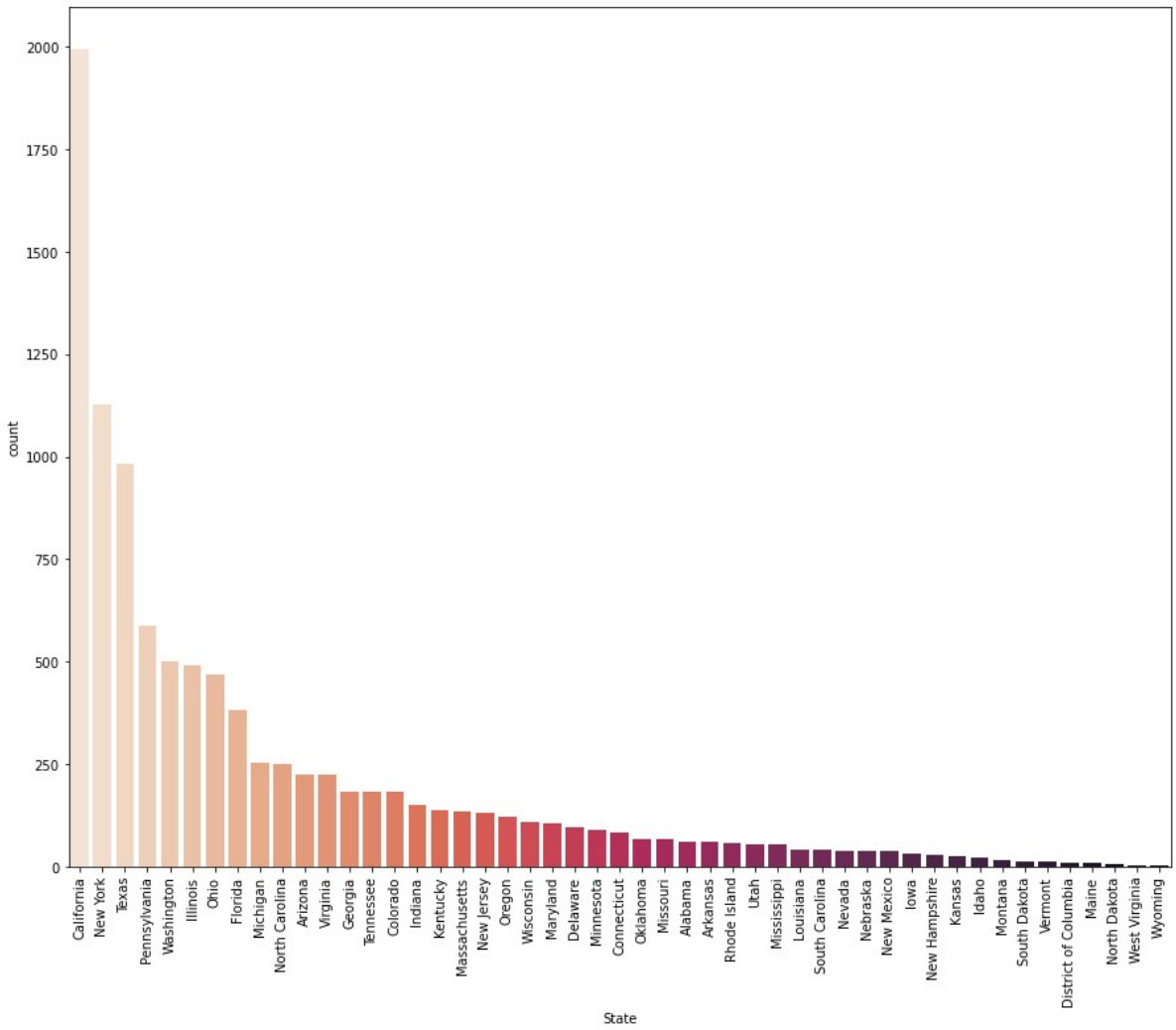
In [29]:
```
plt.figure(figsize=(15,12))
df['Sub-Category'].value_counts().plot.pie(autopct='dark')
plt.show()
```



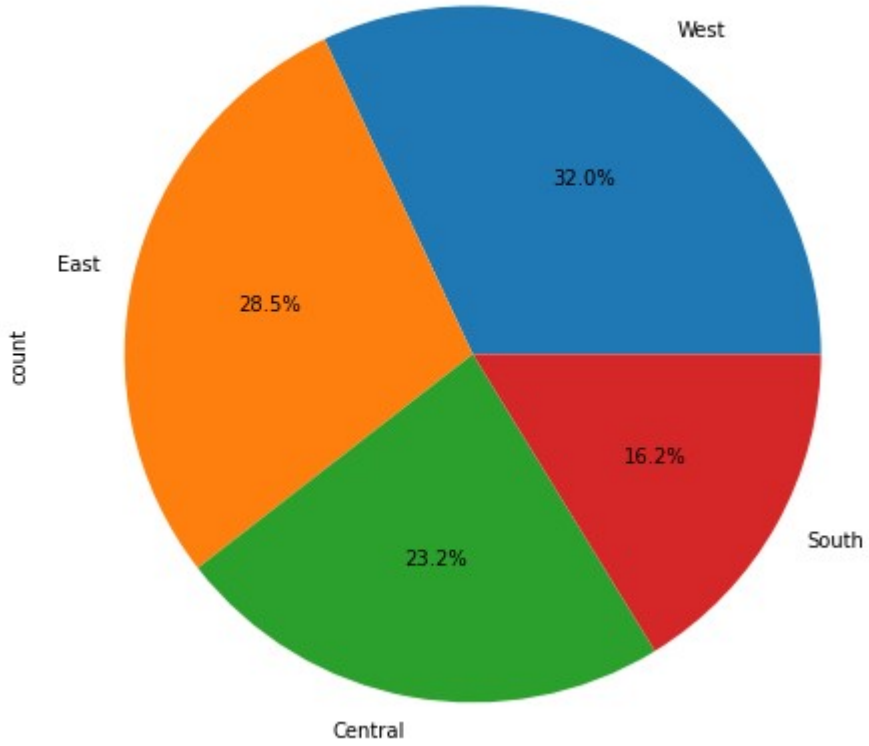
In [30]:
```
df['State'].value_counts()
```

Out[30]:
```
State
California              1996
New York                1127
Texas                    983
Pennsylvania             586
Washington               502
Illinois                 491
Ohio                     468
Florida                  383
Michigan                 254
North Carolina           249
Arizona                  224
Virginia                 224
Georgia                  184
Tennessee                183
Colorado                 182
Indiana                  149
Kentucky                 139
Massachusetts            135
New Jersey               130
Oregon                   123
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: count, dtype: int64
```

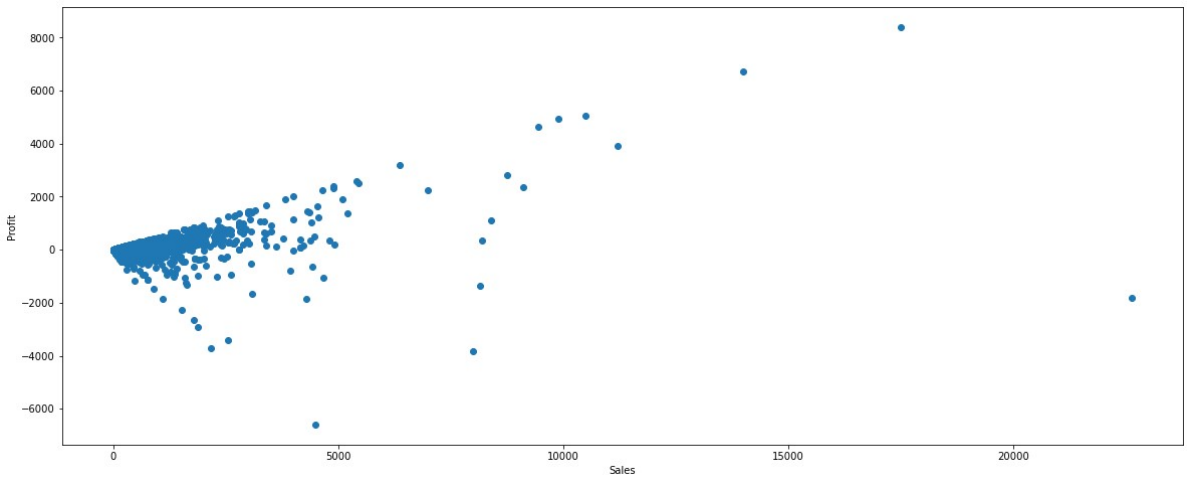
In [31]:
```
plt.figure(figsize=(15,12))
sns.countplot(x='State',data=df,palette='rocket_r',order=df['State'].value_cou
plt.xticks(rotation=90)
plt.show()
```



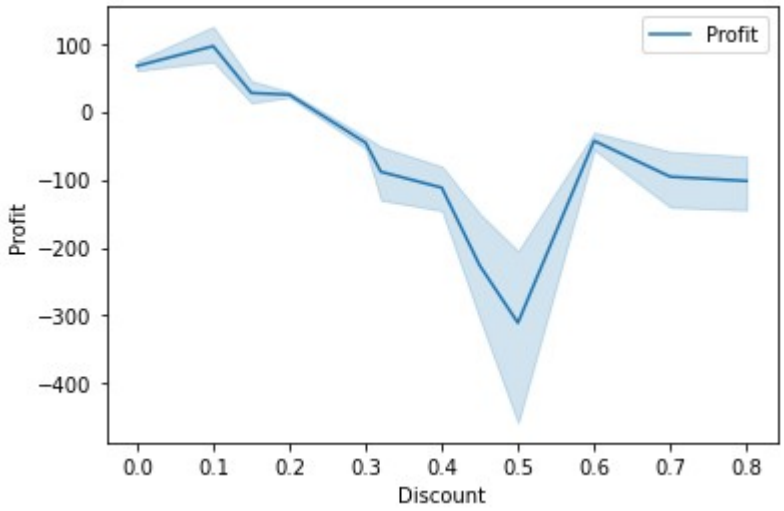
In [33]:
```
plt.figure(figsize=(10,8))
df['Region'].value_counts().plot.pie(autopct = '%1.1f%%')
plt.show()
```



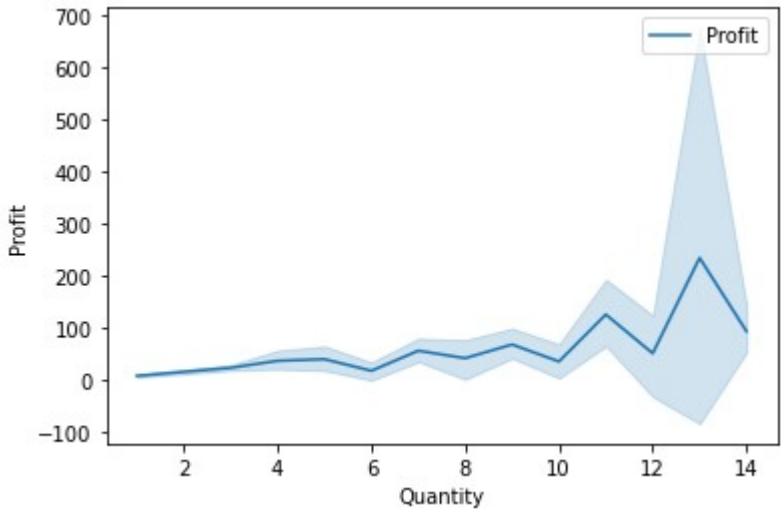
In [34]:
```
fig,ax=plt.subplots(figsize=(20,8))
ax.scatter(df['Sales'],df['Profit'])
ax.set_xlabel('Sales')
ax.set_ylabel('Profit')
plt.show()
```



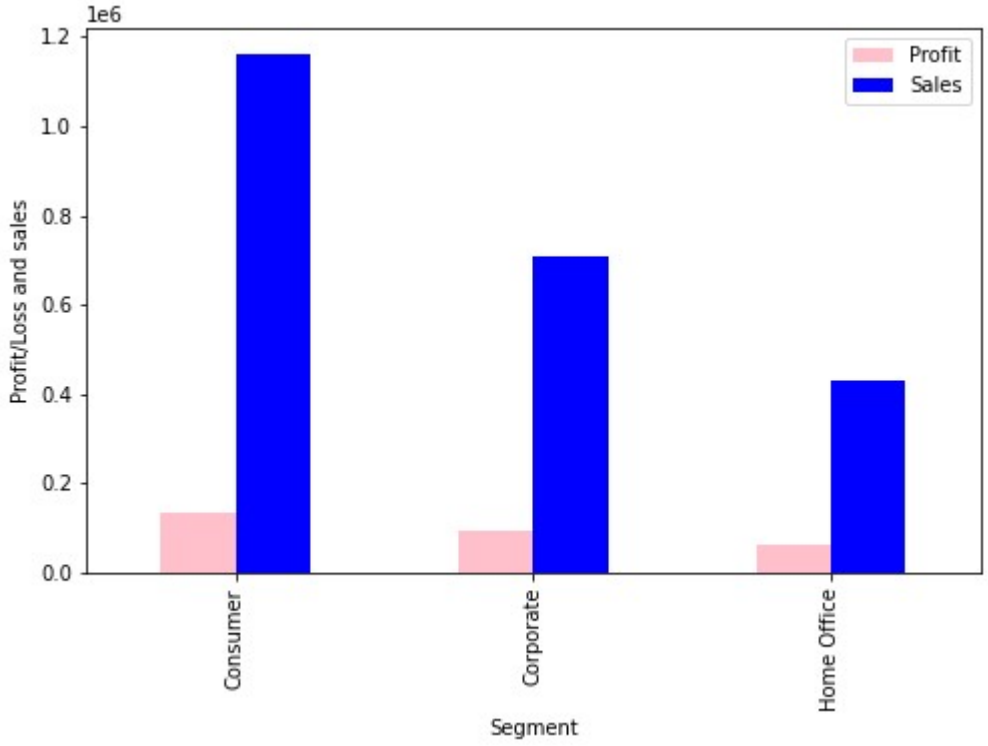
In [35]:
```
sns.lineplot(x='Discount',y='Profit',label='Profit',data=df)
plt.legend()
plt.show()
```



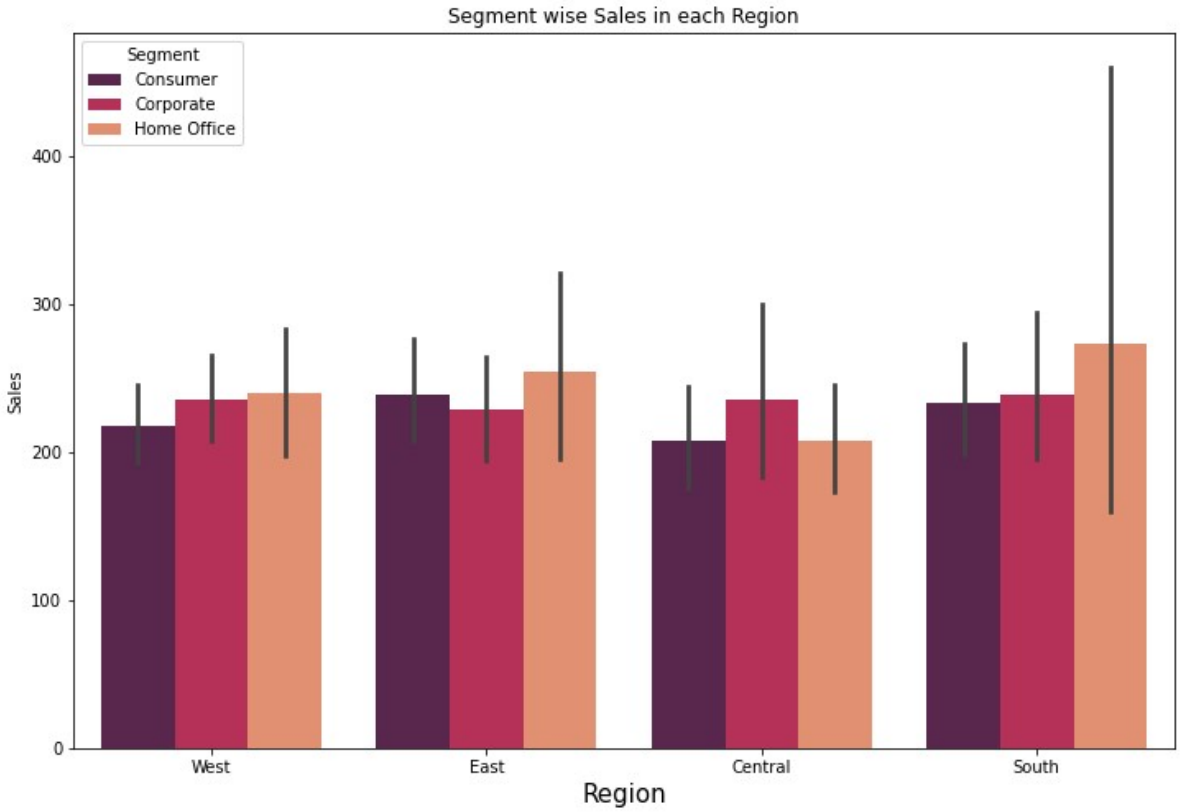
In [36]:
```
sns.lineplot(x='Quantity',y='Profit',label='Profit',data=df)
plt.legend()
plt.show()
```



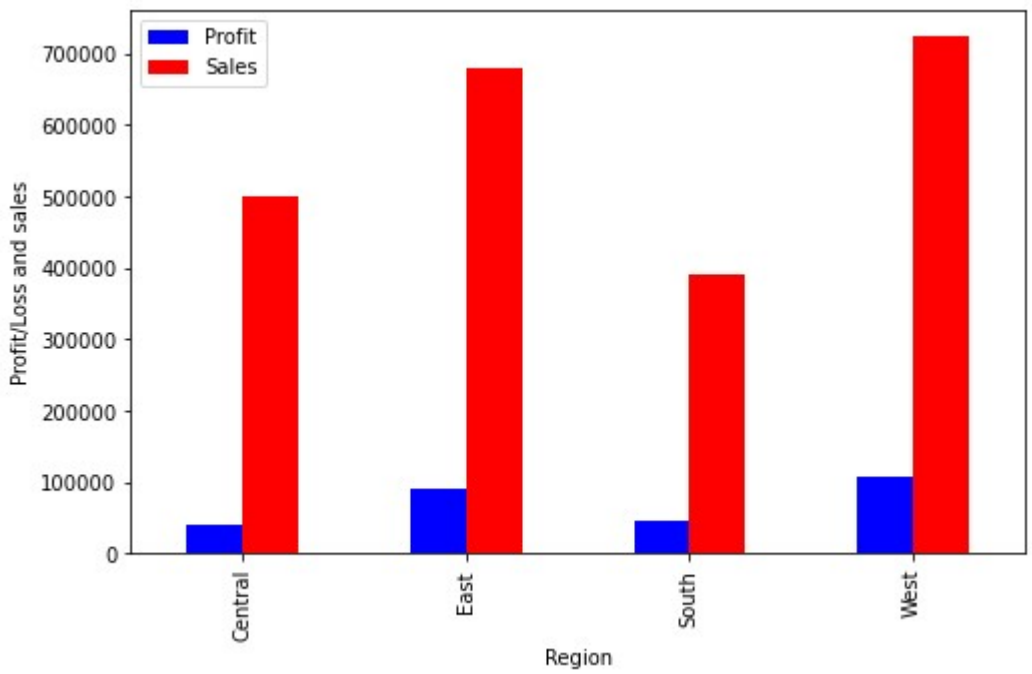
In [37]:
```
df.groupby('Segment')[['Profit','Sales']].sum().plot.bar(color=['pink','blue']
plt.ylabel('Profit/Loss and sales')
plt.show()
```



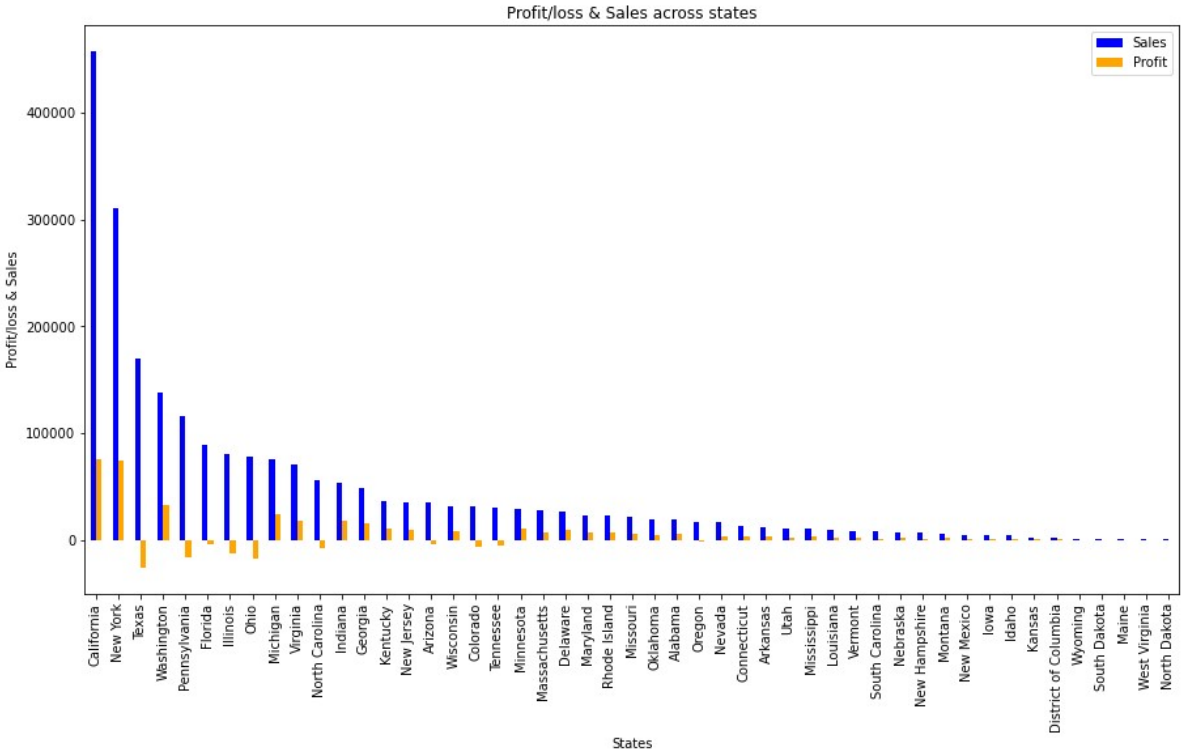
In [38]:
```
plt.figure(figsize=(12,8))
plt.title('Segment wise Sales in each Region')
sns.barplot(x='Region',y='Sales',data=df,hue='Segment',order=df['Region'].valu
plt.xlabel('Region',fontsize=15)
plt.show()
```



In [39]:
```
df.groupby('Region')[['Profit','Sales']].sum().plot.bar(color=['blue','red'],f
plt.ylabel('Profit/Loss and sales')
plt.show()
```



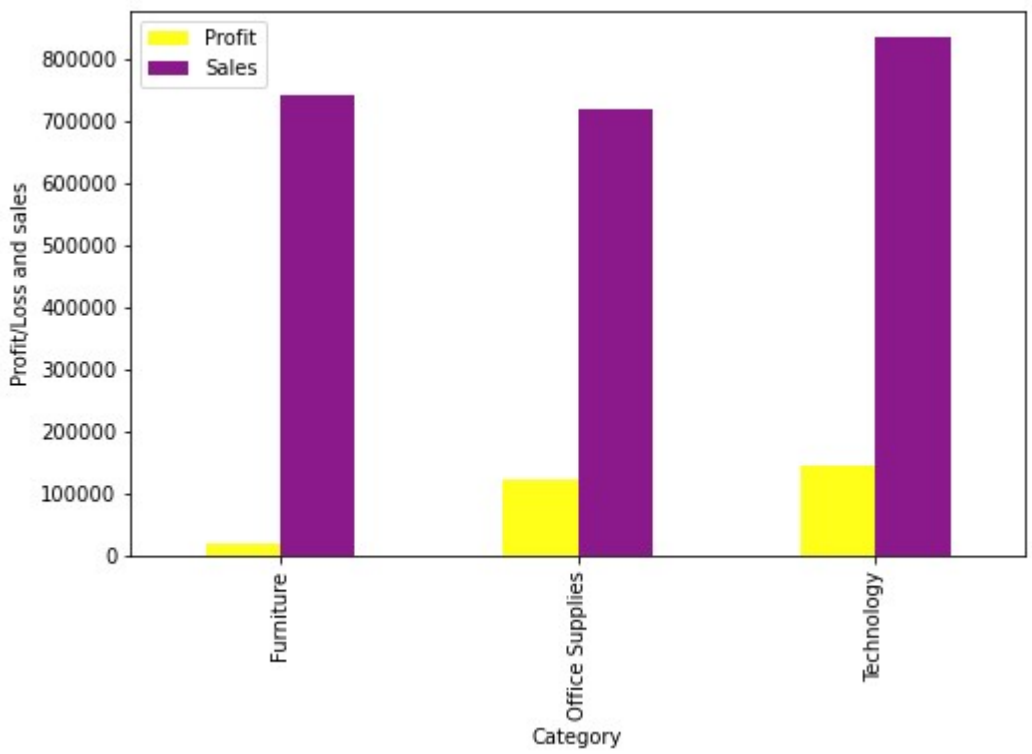
In [40]:
```
ps = df.groupby('State')[['Sales','Profit']].sum().sort_values(by='Sales',asce
ps[:].plot.bar(color=['blue','orange'],figsize=(15,8))
plt.title('Profit/loss & Sales across states')
plt.xlabel('States')
plt.ylabel('Profit/loss & Sales')
plt.show()
```



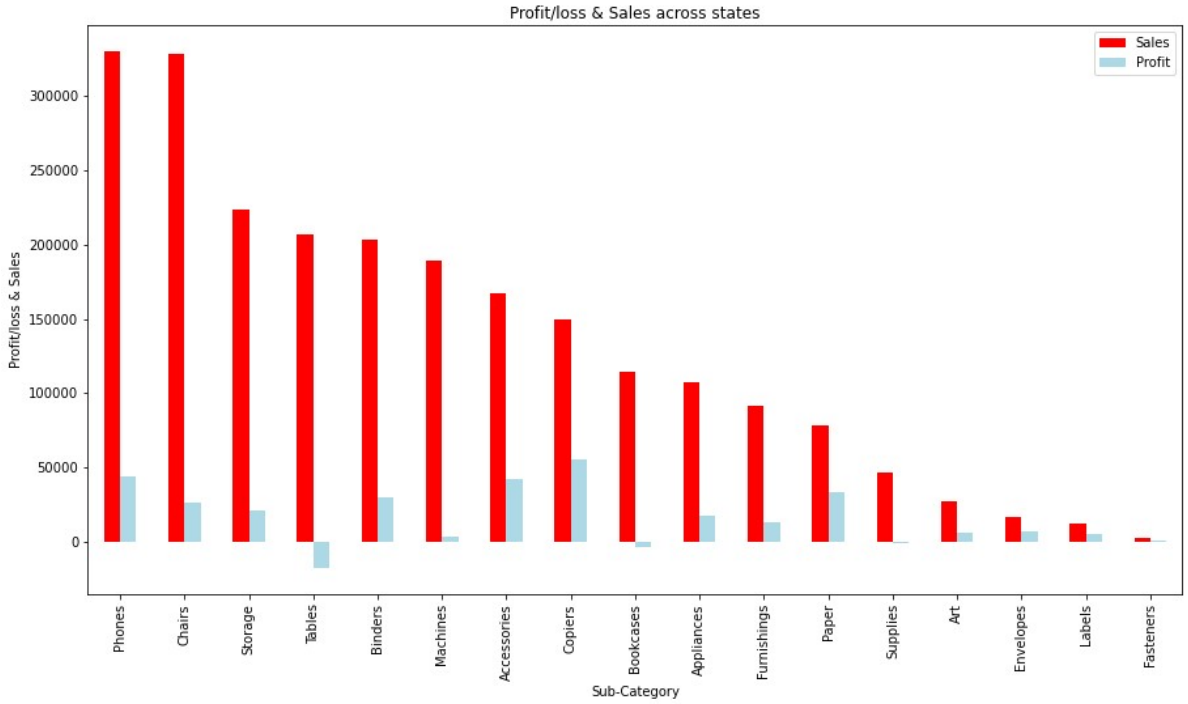
In [41]:
```
t_states = df['State'].value_counts().nlargest(10)
t_states
```

Out[41]:
```
State
California        1996
New York          1127
Texas              983
Pennsylvania       586
Washington         502
Illinois           491
Ohio               468
Florida            383
Michigan           254
North Carolina     249
Name: count, dtype: int64
```

In [42]:
```
df.groupby('Category')[['Profit','Sales']].sum().plot.bar(color=['yellow','pur
plt.ylabel('Profit/Loss and sales')
plt.show()
```



In [43]:
```
ps = df.groupby('Sub-Category')[['Sales','Profit']].sum().sort_values(by='Sale
ps[:].plot.bar(color=['red','lightblue'],figsize=(15,8))
plt.title('Profit/loss & Sales across states')
plt.xlabel('Sub-Category')
plt.ylabel('Profit/loss & Sales')
plt.show()
```



In [ ]: