superstore-analysis

November 16, 2024

1 Import Necessary Libraries

[50]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

1.1 Load and Check the DataSet

[27]:
```
df = pd.read_csv("SampleSuperstore.csv")
df.head()
```

[27]:

	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

1.2 Drop a Column Temporarily

[29]:
```
df.drop(columns='Postal Code')
```

[29]:

	Ship Mode	Segment	Country	City	State	Region	Category	Sub-Category	Sales	Quantity	Discount	Profit
0	Second Class	Consumer	United States	Henderson	Kentucky	South	Furniture	Bookcases	261.9600	2	0.00	41.9136
1	Second Class	Consumer	United States	Henderson	Kentucky	South	Furniture	Chairs	731.9400	3	0.00	219.5820
2	Second Class	Corporate	United States	Los Angeles	California	West	Office Supplies	Labels	14.6200	2	0.00	6.8714
3	Standard Class	Consumer	United States	Fort Lauderdale	Florida	South	Furniture	Tables	957.5775	5	0.45	-383.0310
4	Standard Class	Consumer	United States	Fort Lauderdale	Florida	South	Office Supplies	Storage	22.3680	2	0.20	2.5164
…	…	…	…	…	…	…	…	…	…	…	…	…
9989	Second Class	Consumer	United States	Miami	Florida	South	Furniture	Furnishings	25.2480	3	0.20	4.1028
9990	Standard Class	Consumer	United States	Costa Mesa	California	West	Furniture	Furnishings	91.9600	2	0.00	15.6332
9991	Standard Class	Consumer	United States	Costa Mesa	California	West	Technology	Phones	258.5760	2	0.20	19.3932
9992	Standard Class	Consumer	United States	Costa Mesa	California	West	Office Supplies	Paper	29.6000	4	0.00	13.3200
9993	Second Class	Consumer	United States	Westminster	California	West	Office Supplies	Appliances	243.1600	2	0.00	72.9480

[9994 rows x 12 columns]

1.3 Drop a Column Permanently

[33]:
```
df.drop(columns='Postal Code', inplace=True)
df.head()
```

[33]:

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

1.4 Get Unique Values of Ship Mode, Segment, Country and Category

[45]:
```
print(df['Ship Mode'].unique())
print(df['Segment'].unique())
print(df['Country'].unique())
print(df['Category'].unique())
```

```
['Second Class' 'Standard Class' 'First Class' 'Same Day']
['Consumer' 'Corporate' 'Home Office']
['United States']
['Furniture' 'Office Supplies' 'Technology']
```

1.5 Get Statistical information of only numeric columns

[56]:
```
df.describe()
```

[56]:

	Sales	Quantity	Discount	Profit
count	9994.000000	9994.000000	9994.000000	9994.000000
mean	229.858001	3.789574	0.156203	28.656896
std	623.245101	2.225110	0.206452	234.260108
min	0.444000	1.000000	0.000000	-6599.978000
25%	17.280000	2.000000	0.000000	1.728750
50%	54.490000	3.000000	0.200000	8.666500
75%	209.940000	5.000000	0.200000	29.364000
max	22638.480000	14.000000	0.800000	8399.976000

1.6 Get more Information about the dataset

[59]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9994 entries, 0 to 9993
Data columns (total 12 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   Region        9994 non-null   object
 6   Category      9994 non-null   object
 7   Sub-Category  9994 non-null   object
 8   Sales         9994 non-null   float64
 9   Quantity      9994 non-null   int64
 10  Discount      9994 non-null   float64
 11  Profit        9994 non-null   float64
dtypes: float64(3), int64(1), object(8)
memory usage: 937.1+ KB
```

1.7 Check for any missing Values

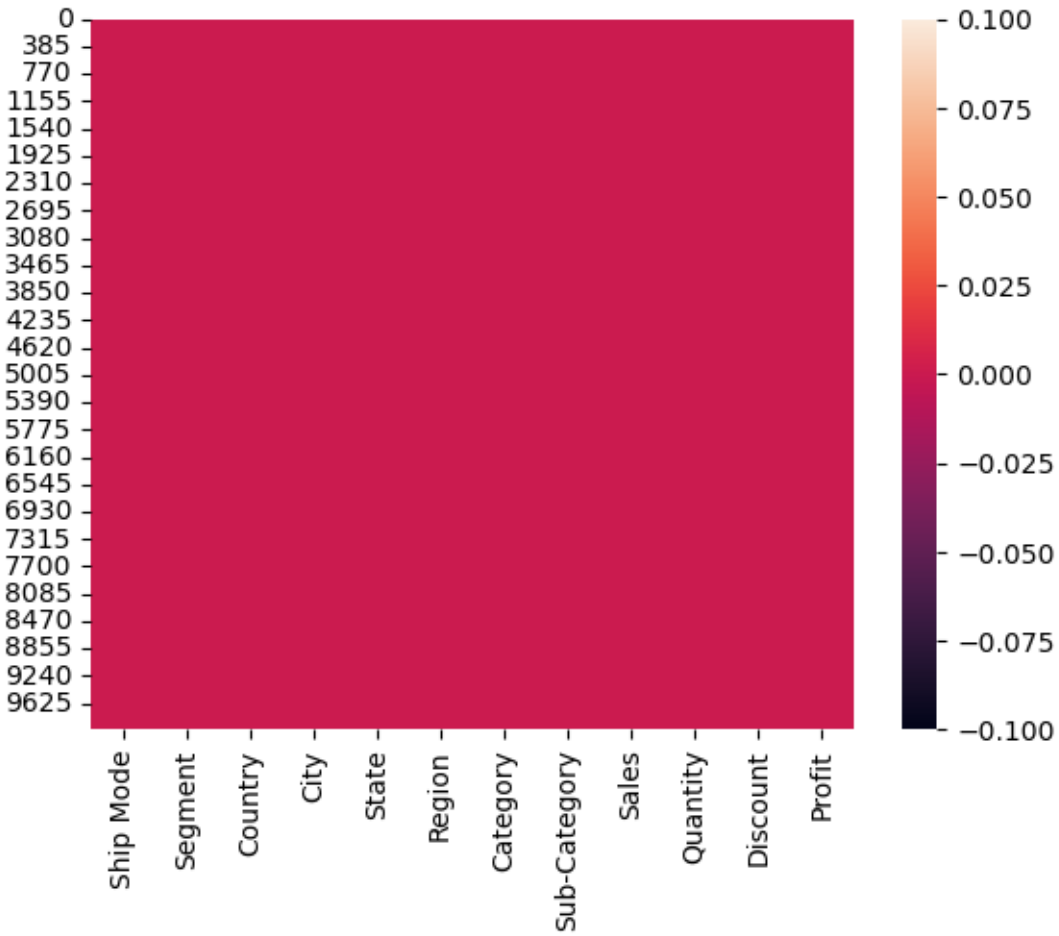
[67]:
```
df.isna().sum()
```

[67]:
```
Ship Mode       0
Segment         0
Country         0
City            0
State           0
Region          0
Category        0
Sub-Category    0
Sales           0
Quantity        0
Discount        0
Profit          0
dtype: int64
```

1.8 Check For Null values using Visualization

[69]:
```
sns.heatmap(df.isnull())
```

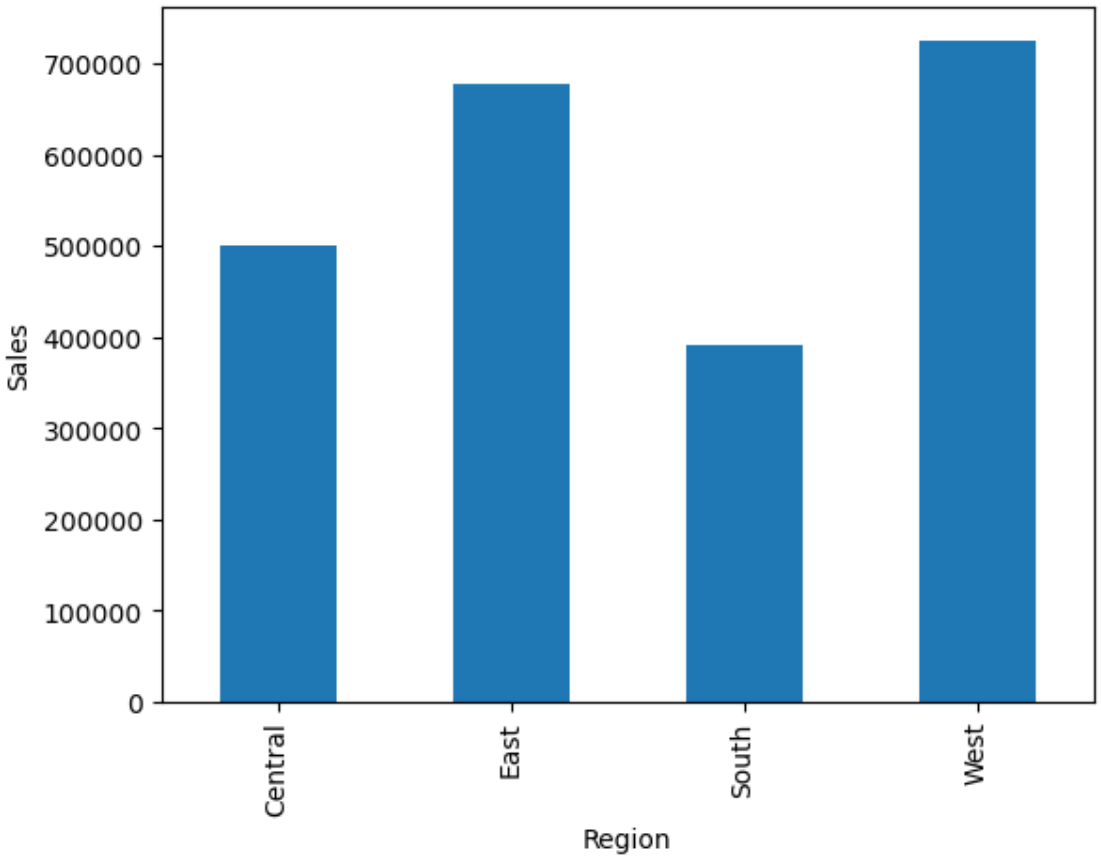
[69]:
```
<Axes: >
```



1.9 Sales Analysis based on Regions

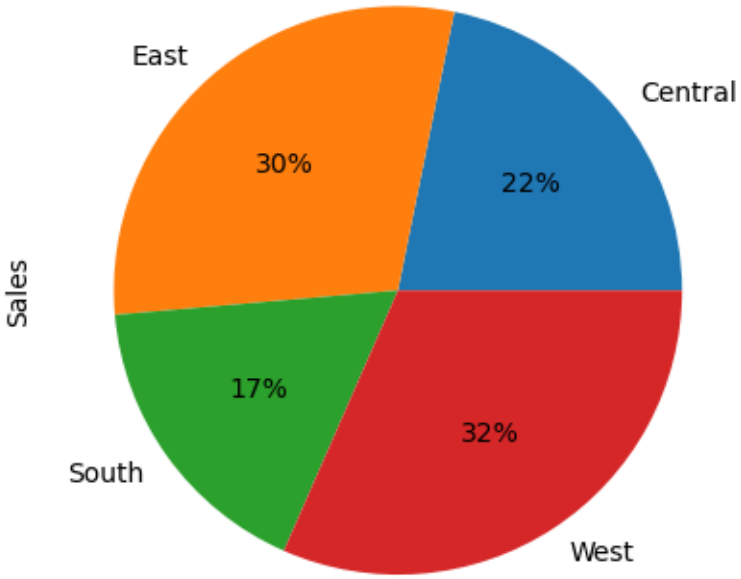
[76]:
```
df.groupby("Region")['Sales'].sum().plot.bar()
plt.xlabel("Region")
plt.ylabel("Sales")
```

[76]:
```
Text(0, 0.5, 'Sales')
```



[81]:
```
df.groupby("Region")['Sales'].sum().plot.pie(autopct="%1.0f%%")
```

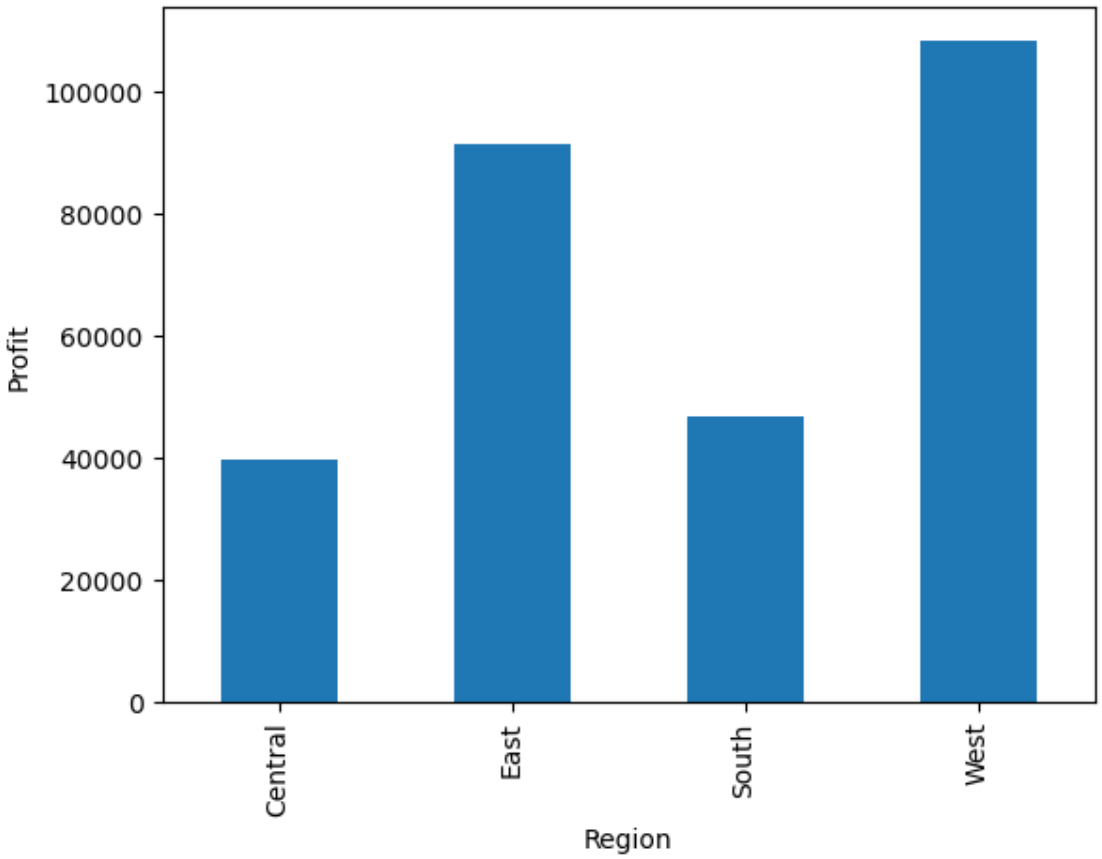
[81]: <Axes: ylabel='Sales'>



1.10 Profit Analysis based on Regions

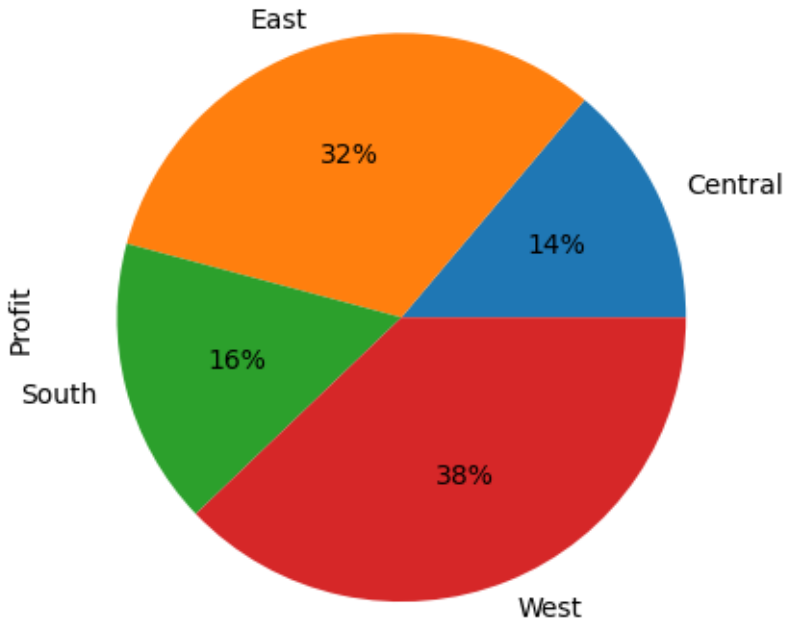
[79]:
```
df.groupby("Region")['Profit'].sum().plot.bar()
plt.xlabel("Region")
plt.ylabel("Profit")
```

[79]:
```
Text(0, 0.5, 'Profit')
```



[83]:
```
df.groupby("Region")['Profit'].sum().plot.pie(autopct="%1.0f%%")
```

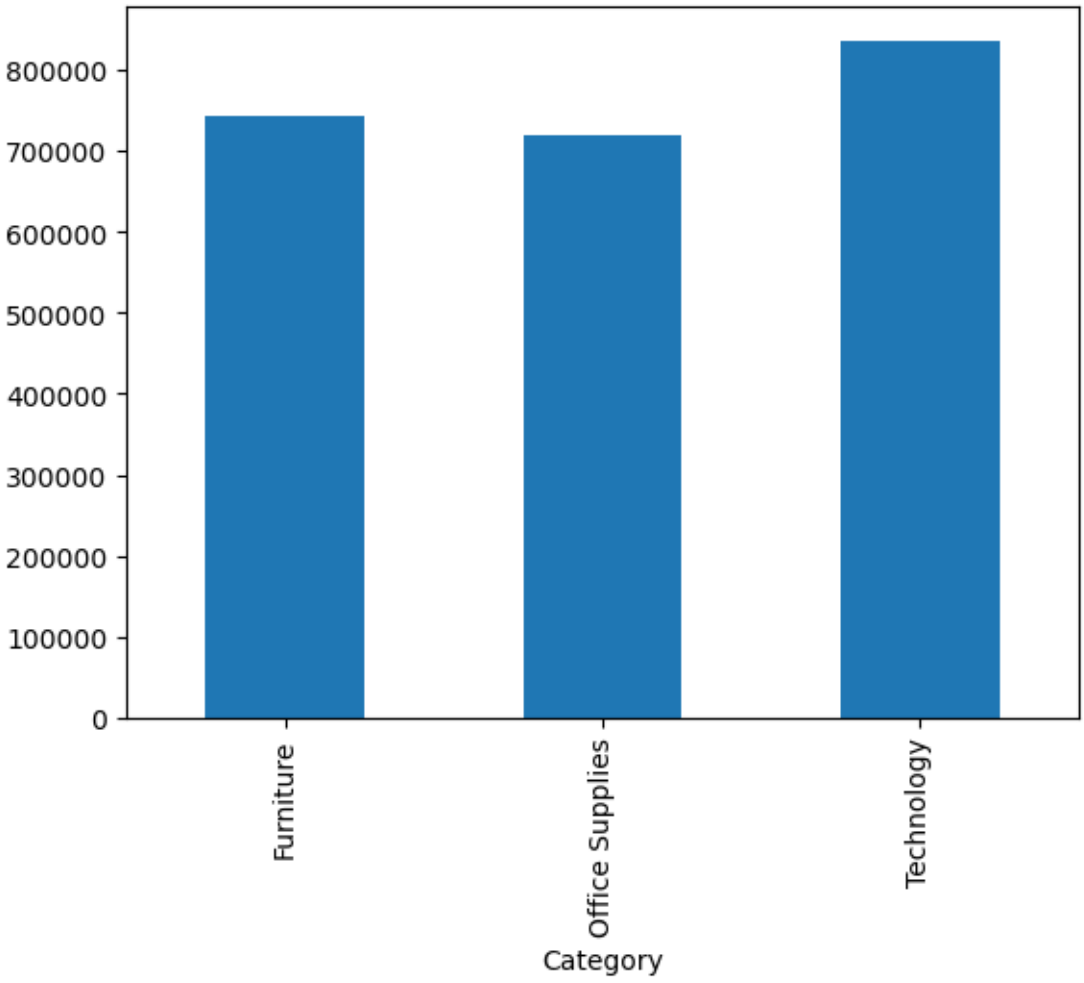
[83]: <Axes: ylabel='Profit'>



1.11 Sales analysis based on Category

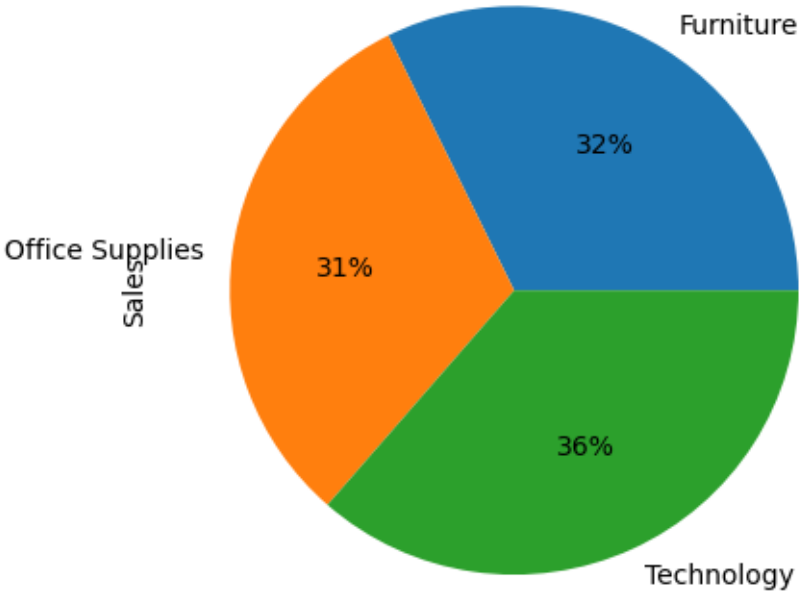
[93]:
```
df.groupby("Category")['Sales'].sum().plot.bar()
```

[93]:
```
<Axes: xlabel='Category'>
```



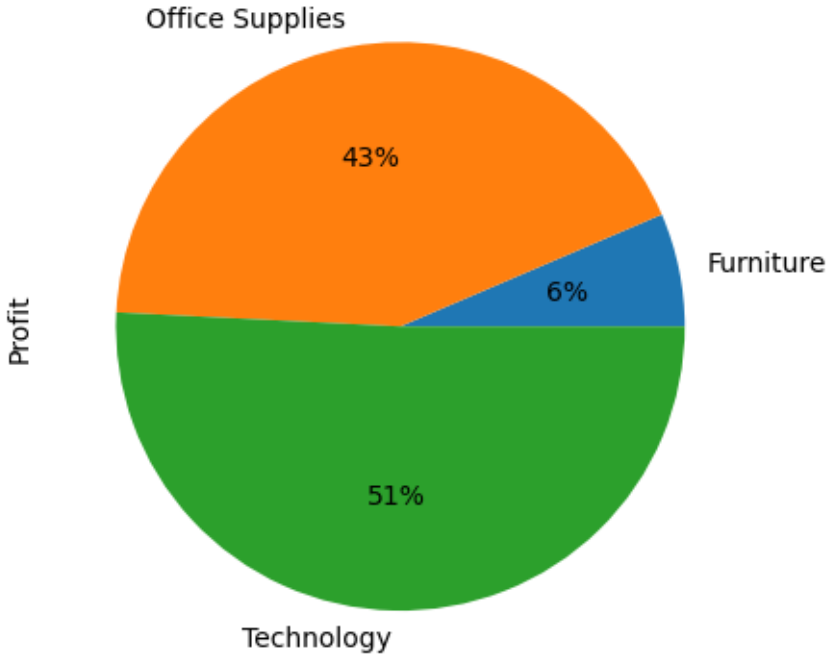
[95]:
```
df.groupby("Category")['Sales'].sum().plot.pie(autopct="%1.0f%%")
```

[95]: <Axes: ylabel='Sales'>



[97]:
```
df.groupby("Category")['Profit'].sum().plot.pie(autopct="%1.0f%%")
```

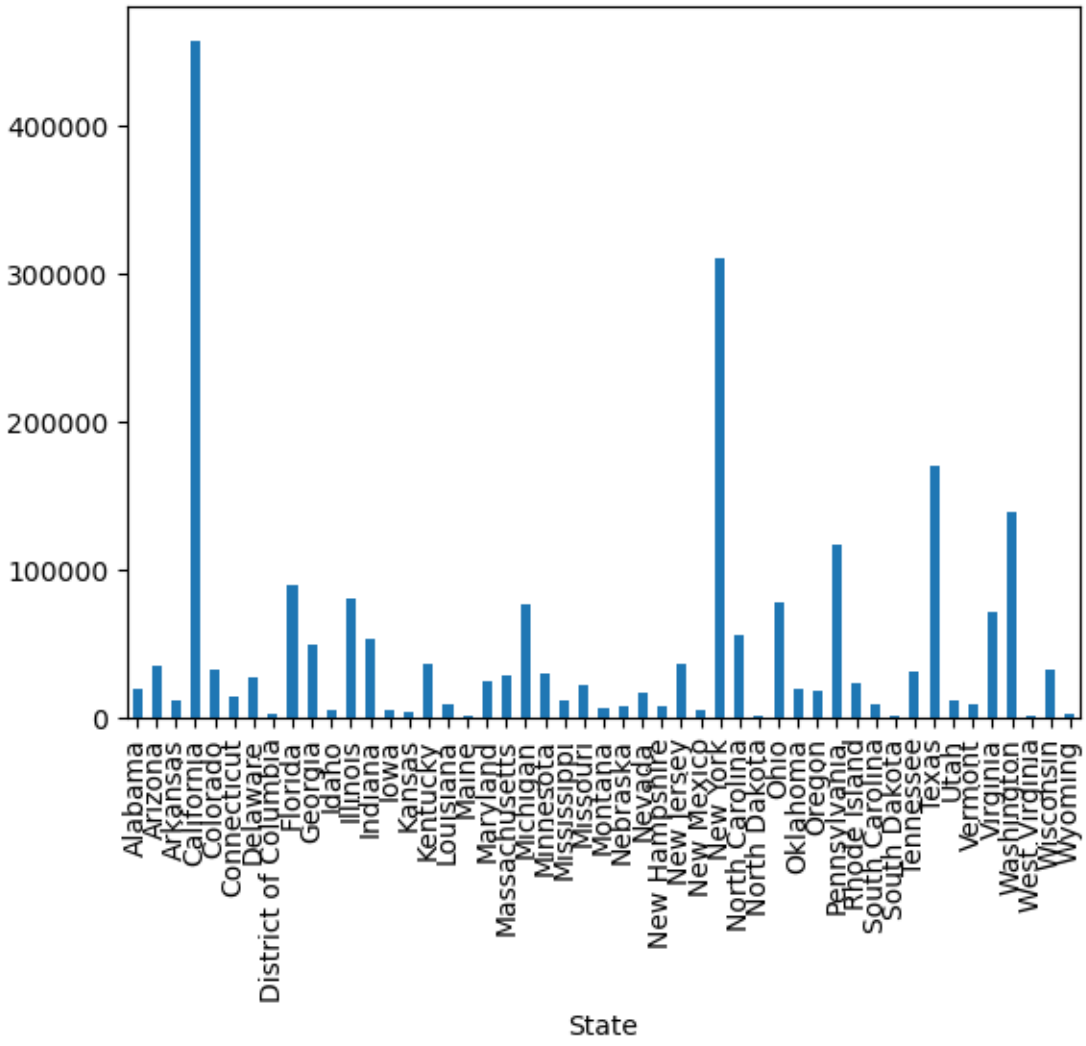
[97]: <Axes: ylabel='Profit'>



1.12 Analyse the Sales Statewise

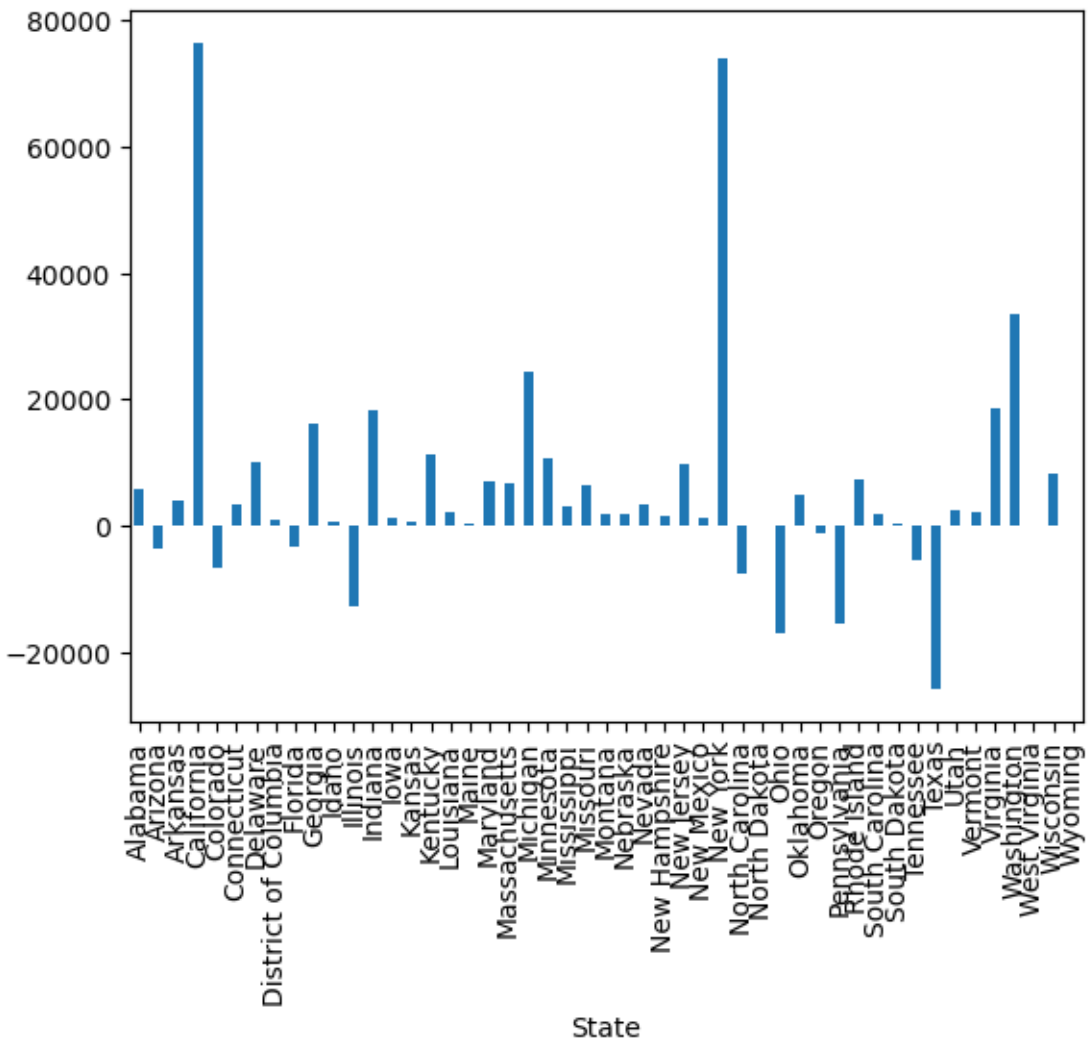
[99]:
```
df.groupby("State")['Sales'].sum().plot.bar()
```

[99]:
```
<Axes: xlabel='State'>
```



[102]:
```
df.groupby("State")['Profit'].sum().plot.bar()
```

[102]: <Axes: xlabel='State'>


80000
60000
40000
20000
0
−20000
Alabama
Arizona
Arkansas
California
Colorado
Connecticut
Delaware
District of Columbia
Florida
Georgia
Idaho
Illinois
Indiana
Iowa
Kansas
Kentucky
Louisiana
Maine
Maryland
Massachusetts
Michigan
Minnesota
Mississippi
Missouri
Montana
Nebraska
Nevada
New Hampshire
New Jersey
New Mexico
New York
North Carolina
North Dakota
Ohio
Oklahoma
Oregon
Pennsylvania
Rhode Island
South Carolina
South Dakota
Tennessee
Texas
Utah
Vermont
Virginia
Washington
West Virginia
Wisconsin
Wyoming
State