```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

```python
df=pd.read_csv("SampleSuperstore.csv")
df.head()
```

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Sub-Category	Sales	Quan
0	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Bookcases	261.9600	
1	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Chairs	731.9400	
2	Second Class	Corporate	United States	Los Angeles	California	90036	West	Office Supplies	Labels	14.6200	
3	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Furniture	Tables	957.5775	
4	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Office Supplies	Storage	22.3680	

```python
df.drop(columns="Postal Code",inplace=True)
```

```python
df.head()
```

	Ship Mode	Segment	Country	City	State	Region	Category	Sub-Category	Sales	Quantity	Di:
0	Second Class	Consumer	United States	Henderson	Kentucky	South	Furniture	Bookcases	261.9600	2	
1	Second Class	Consumer	United States	Henderson	Kentucky	South	Furniture	Chairs	731.9400	3	
2	Second Class	Corporate	United States	Los Angeles	California	West	Office Supplies	Labels	14.6200	2	

```python
print(df["Ship Mode"].unique())
print(df["Segment"].unique())
print(df["Country"].unique())
print(df["Category"].unique())
print(df["Sub-Category"].unique())
print(df["Region"].unique())
```

```
['Second Class' 'Standard Class' 'First Class' 'Same Day']
['Consumer' 'Corporate' 'Home Office']
['United States']
['Furniture' 'Office Supplies' 'Technology']
['Bookcases' 'Chairs' 'Labels' 'Tables' 'Storage' 'Furnishings' 'Art'
 'Phones' 'Binders' 'Appliances' 'Paper' 'Accessories' 'Envelopes'
 'Fasteners' 'Supplies' 'Machines' 'Copiers']
['South' 'West' 'Central' 'East']
```

```python
df.describe()
```

	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

```
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
```

```
df.isna().sum()
```

```
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
```
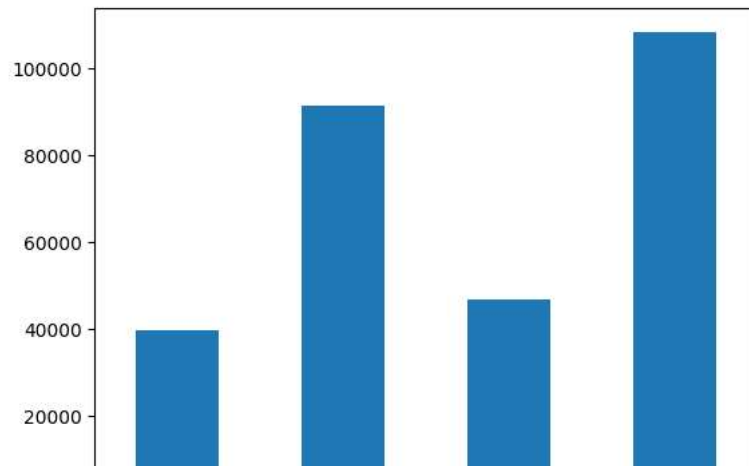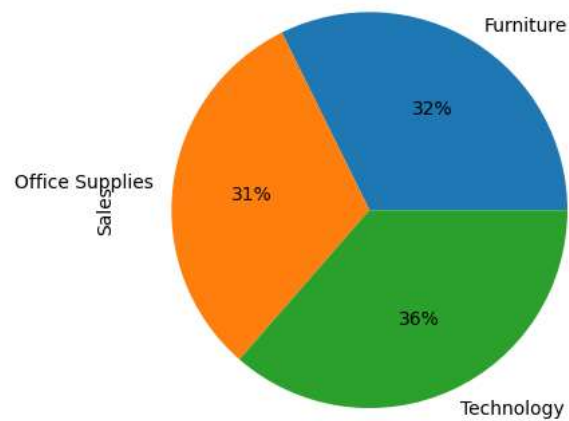
```
df.groupby("Region")["Sales"].sum().plot.bar()
```

```
<Axes: xlabel='Region'>
```
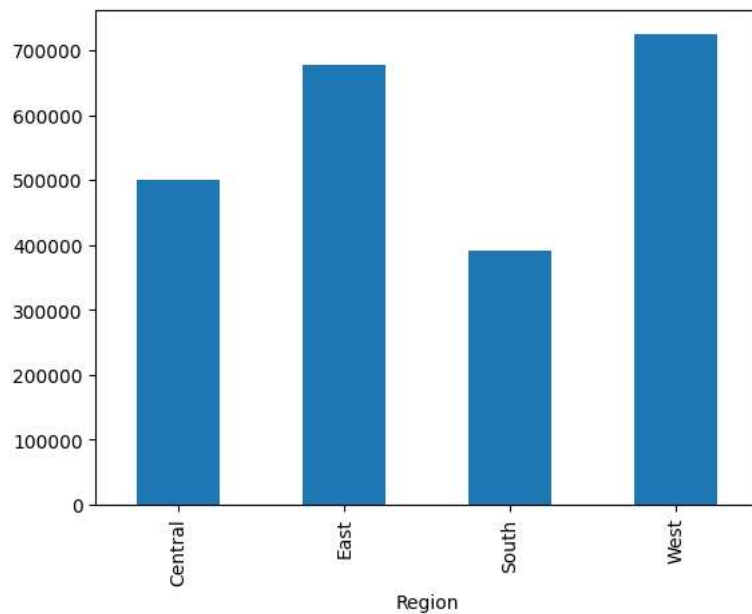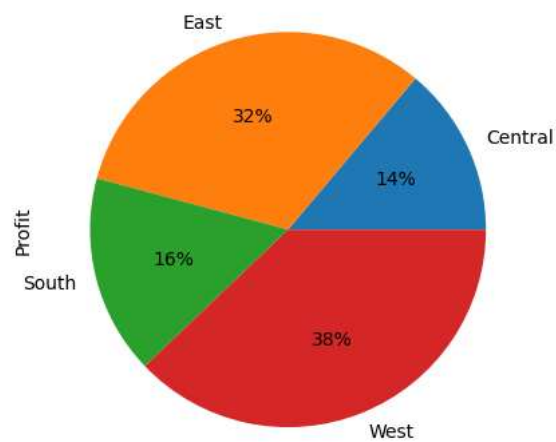


```
df.groupby("Region")["Profit"].sum().plot.bar()
```

```
<Axes: xlabel='Region'>
```



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

```
<Axes: ylabel='Sales'>
```



```python
df.groupby("Region")["Profit"].sum().plot.pie(autopct="%1.0f%%")
```
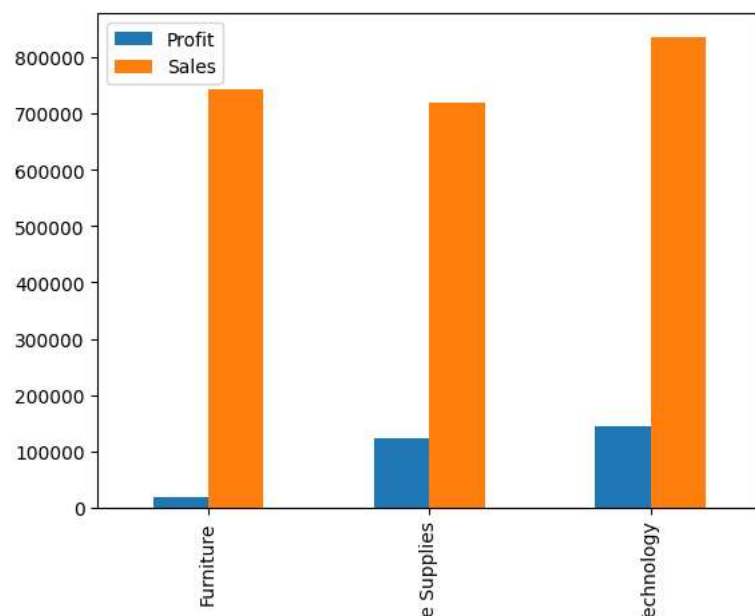
```
<Axes: ylabel='Profit'>
```
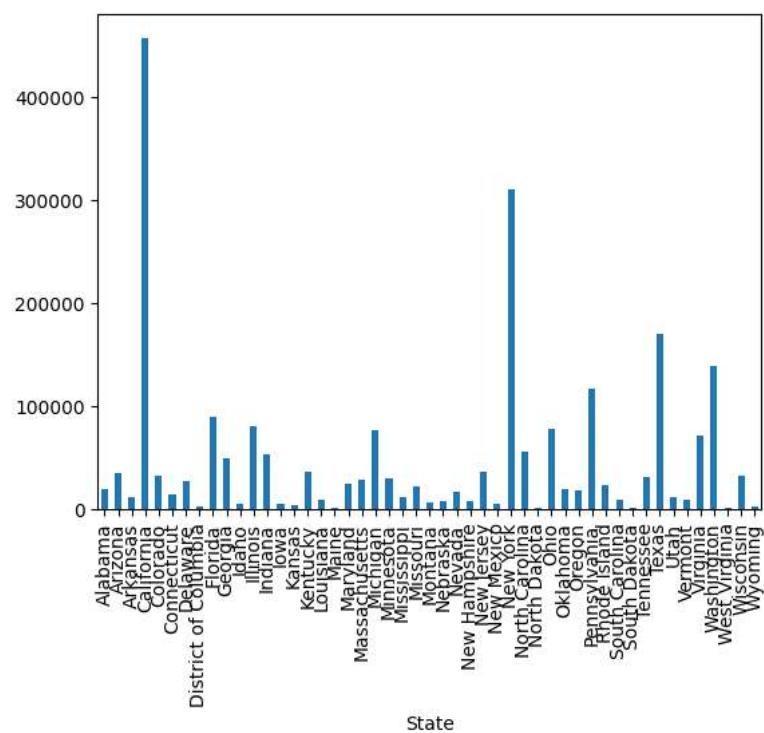


```python
df.groupby("Category")["Profit","Sales"].sum().plot.bar()
```

```
<ipython-input-38-06cd16d6bfdb>:1: FutureWarning: Indexing with multiple keys (implicitly converted to
  df.groupby("Category")["Profit","Sales"].sum().plot.bar()
<Axes: xlabel='Category'>
```



```
df.groupby("State")["Sales"].sum().plot.bar()
```
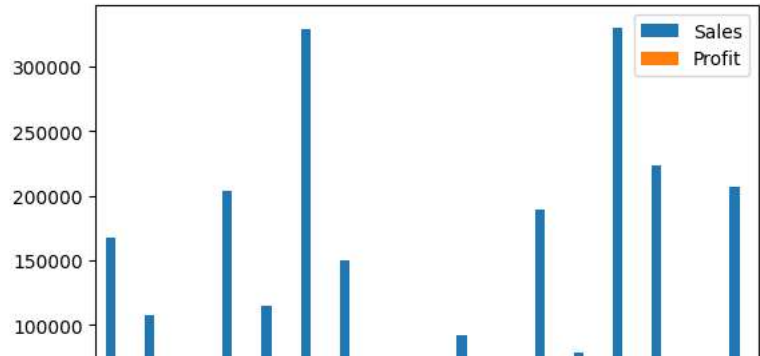
```
<Axes: xlabel='State'>
```



```
df.groupby("Sub-Category")["Sales","Profit"].sum().plot.bar()
```

```
<ipython-input-43-53028dbf61b1>:1: FutureWarning: Indexing with multiple keys (implicit
  df.groupby("Sub-Category")["Sales","Profit"].sum().plot.bar()
<Axes: xlabel='Sub-Category'>
```



```
df.groupby("Sub-Category")["Sales"].sum().plot.bar()
```

```
<Axes: xlabel='Sub-Category'>
```