```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
cd drive/
```

```
/content/drive
```

```
cd MyDrive/
```

```
/content/drive/MyDrive
```

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

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

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Sub-Category	
0	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Bookcases	26
1	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Chairs	73
2	Second Class	Corporate	United States	Los Angeles	California	90036	West	Office Supplies	Labels	
3	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Furniture	Tables	95
4	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Office Supplies	Storage	2

```
df.drop(columns="Postal Code")
```

	Ship Mode	Segment	Country	City	State	Region	Category	Sub-Category	S
0	Second Class	Consumer	United States	Henderson	Kentucky	South	Furniture	Bookcases	261.
1	Second Class	Consumer	United States	Henderson	Kentucky	South	Furniture	Chairs	731.
2	Second Class	Corporate	United States	Los Angeles	California	West	Office Supplies	Labels	14.
3	Standard Class	Consumer	United States	Fort Lauderdale	Florida	South	Furniture	Tables	957.
4	Standard Class	Consumer	United States	Fort Lauderdale	Florida	South	Office Supplies	Storage	22.
...	...	...	...	...	...	...	...	...	
9989	Second Class	Consumer	United States	Miami	Florida	South	Furniture	Furnishings	25.
9990	Standard Class	Consumer	United States	Costa Mesa	California	West	Furniture	Furnishings	91.
9991	Standard Class	Consumer	United States	Costa Mesa	California	West	Technology	Phones	258.
9992	Standard Class	Consumer	United States	Costa Mesa	California	West	Office Supplies	Paper	29.
9993	Second Class	Consumer	United States	Westminster	California	West	Office Supplies	Appliances	243.

9994 rows × 12 columns

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

```
df.head()
```

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

```
print(df["Ship Mode"].unique())
print(df["Segment"].unique())
print(df["Country"].unique())
print(df["Category"].unique())
print(df["City"].unique())
print(df["State"].unique())
print(df["Region"].unique())
print(df["Sub-Category"].unique())
print(df["Sales"].unique())
print(df["Quantity"].unique())
print(df["Discount"].unique())
print(df["Profit"].unique())
```

```
 Arkansas   Montana   New Hampshire   Maryland   District of Columbia
 'Kansas' 'Vermont' 'Maine' 'South Dakota' 'Idaho' 'North Dakota'
 'Wyoming' 'West Virginia']
['South' 'West' 'Central' 'East']
['Bookcases' 'Chairs' 'Labels' 'Tables' 'Storage' 'Furnishings' 'Art'
 'Phones' 'Binders' 'Appliances' 'Paper' 'Accessories' 'Envelopes'
 'Fasteners' 'Supplies' 'Machines' 'Copiers']
[261.96  731.94   14.62  ... 437.472  97.98  243.16 ]
[ 2  3  5  7  4  6  9  1  8 14 11 13 10 12]
[0.   0.45 0.2  0.8  0.3  0.5  0.7  0.6  0.32 0.1  0.4  0.15]
[ 41.9136 219.582    6.8714 ...  16.124    4.1028  72.948 ]
```

```
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()
```

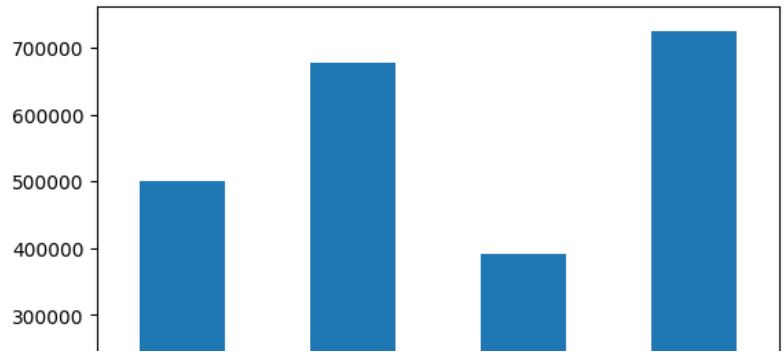
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9994 entries, 0 to 9993
Data columns (total 12 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   Ship Mode     9994 non-null   object 
 1   Segment       9994 non-null   object 
 2   Country       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
```

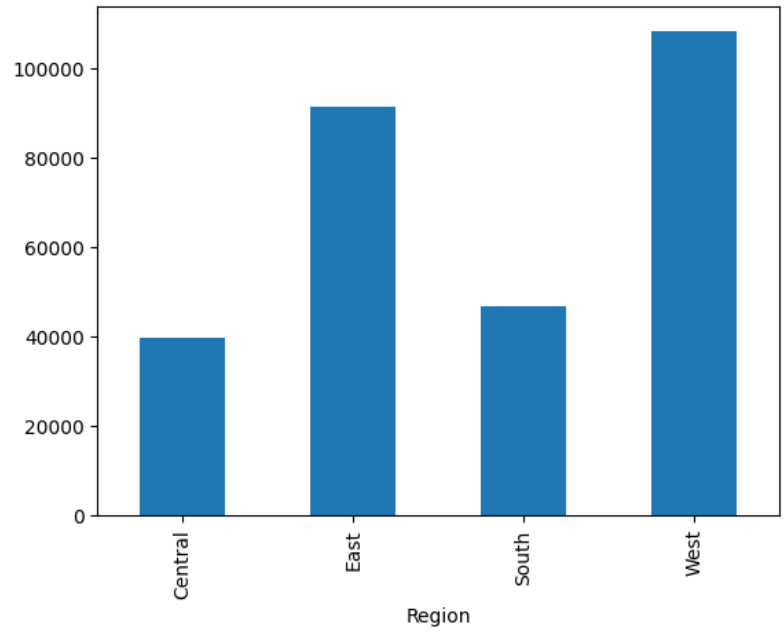
```
df.groupby("Region")["Sales"].sum().plot.bar()
```

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




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

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

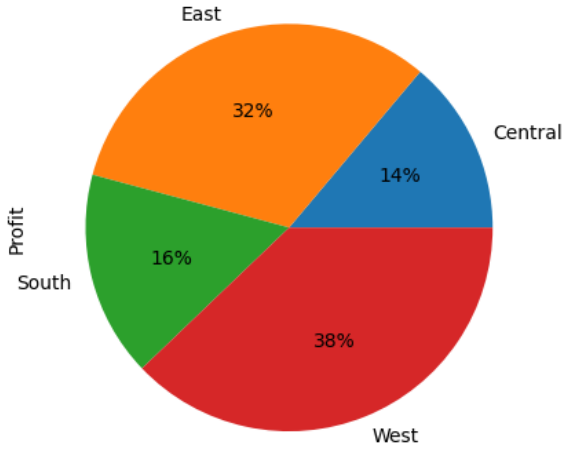



```
df.groupby("Region")["Sales"].sum().plot.pie(autopct="%1.0f%%")
```

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

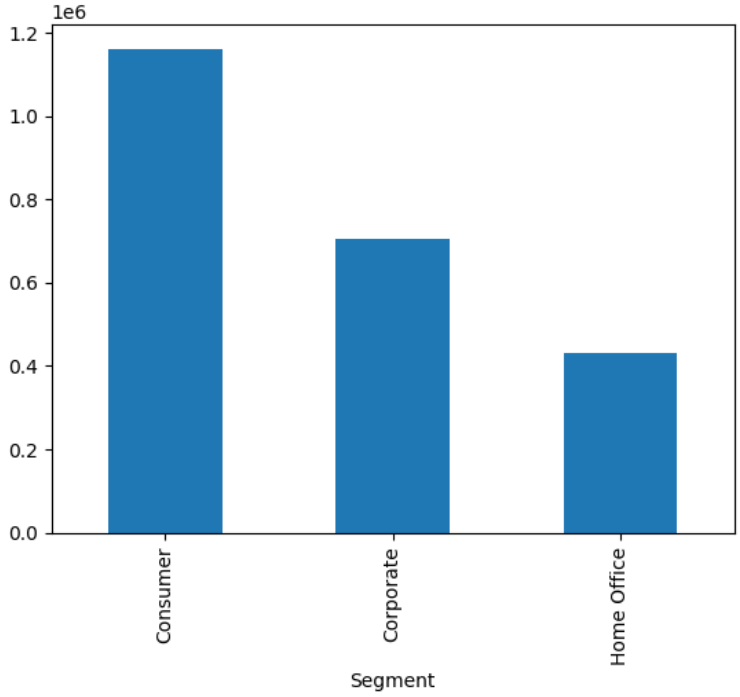
```
df.groupby("Region")["Profit"].sum().plot.pie(autopct="%1.0f%%")
```

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



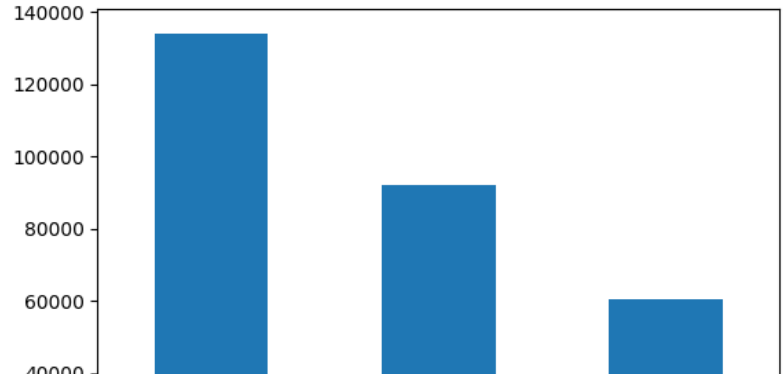

```
df.groupby("Segment")["Sales"].sum().plot.bar()
```

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



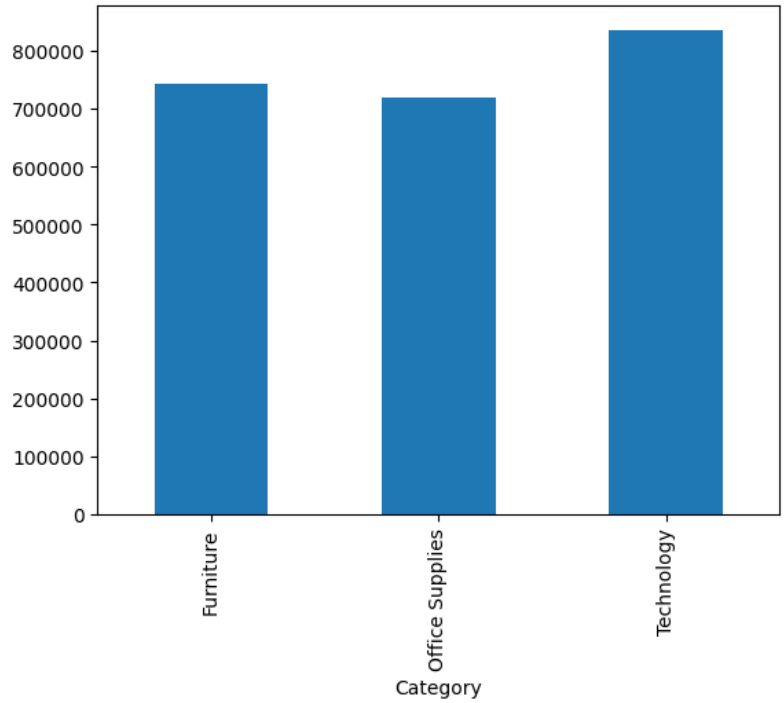

```
df.groupby("Segment")["Profit"].sum().plot.bar()
```

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



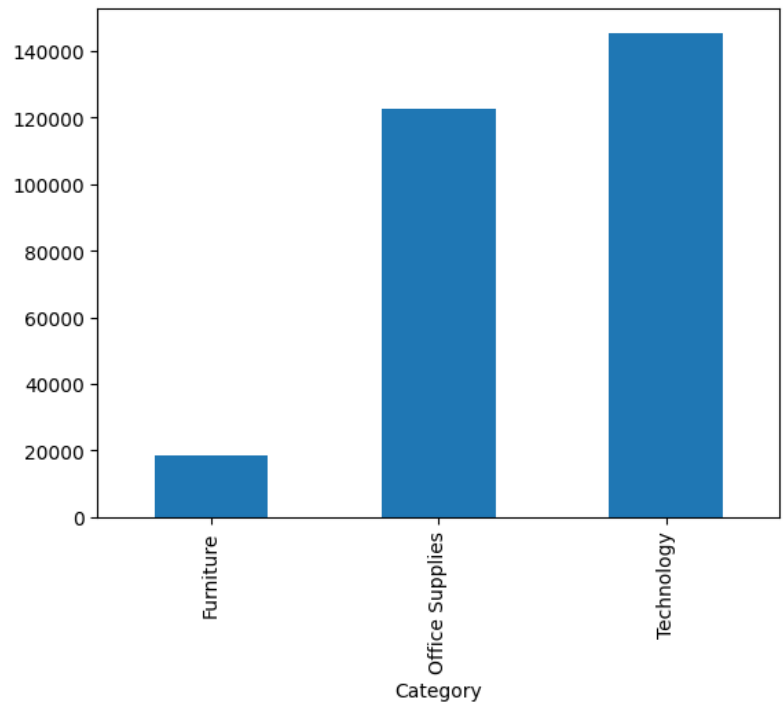

```
df.groupby("Category")["Sales"].sum().plot.bar()
```

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



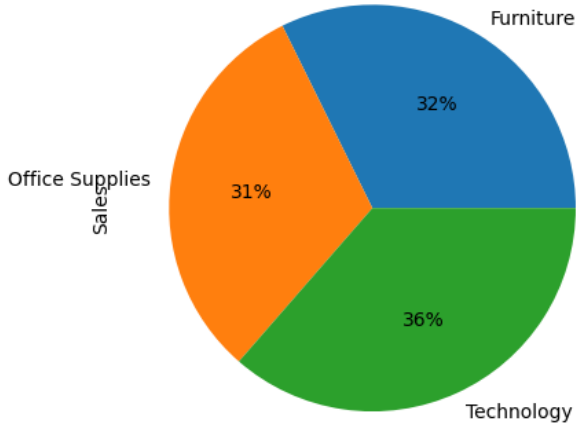

```
df.groupby("Category")["Profit"].sum().plot.bar()
```

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



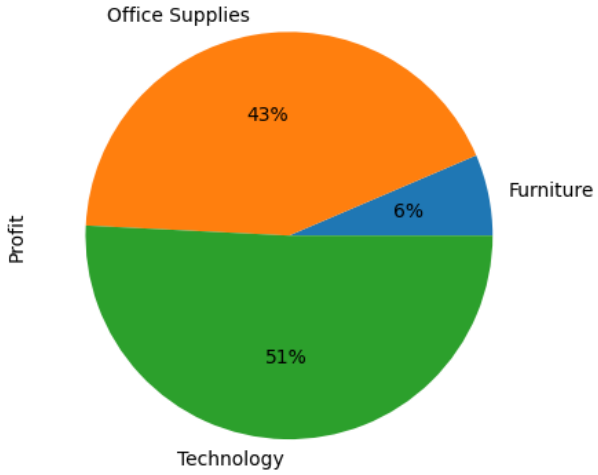
```
df.groupby("Category")["Sales"].sum().plot.pie(autopct="%1.0f%%")
```

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



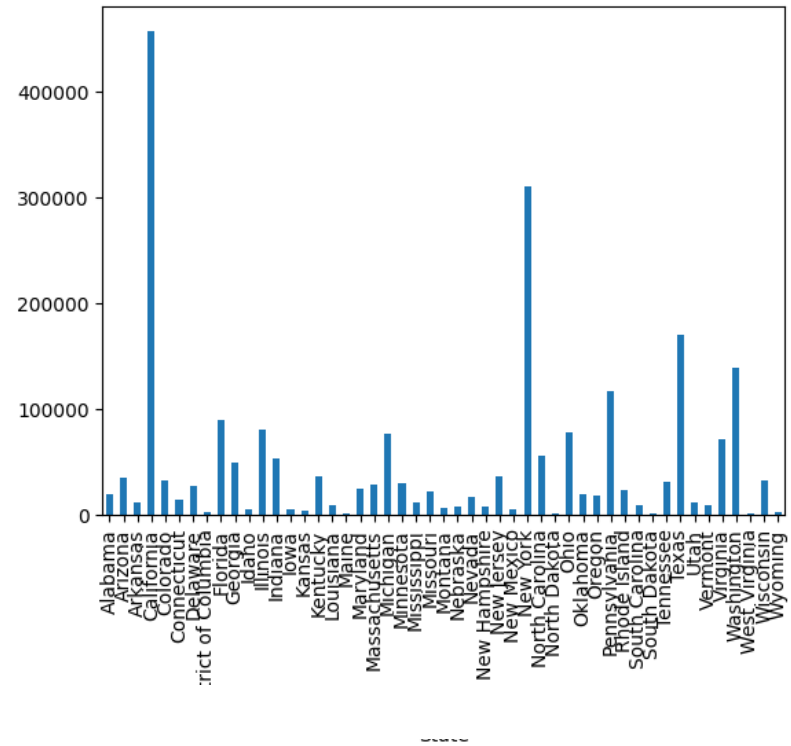

```
df.groupby("Category")["Profit"].sum().plot.pie(autopct="%1.0f%%")
```

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




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

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




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

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