amazon-sales-analysis

January 16, 2024

[2]:
```
!pip install numpy
```

```
Requirement already satisfied: numpy in d:\users\ravi\anaconda3\lib\site-
packages (1.24.3)
```

[3]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

[27]:
```
df=pd.read_csv(r'C:\Users\RAVI\Downloads\python␣
↪project\Python_Amazon_Sales_Analysis-main\Amazon Sale Report.csv')
```

[28]:
```
df.shape
```

[28]:
```
(128976, 21)
```

[29]:
```
df.head(20)
```

[29]:

	index	Order ID	Date	Status	\
0	0	405-8078784-5731545	04-30-22	Cancelled	
1	1	171-9198151-1101146	04-30-22	Shipped - Delivered to Buyer	
2	2	404-0687676-7273146	04-30-22	Shipped	
3	3	403-9615377-8133951	04-30-22	Cancelled	
4	4	407-1069790-7240320	04-30-22	Shipped	
5	5	404-1490984-4578765	04-30-22	Shipped	
6	6	408-5748499-6859555	04-30-22	Shipped	
7	7	406-7807733-3785945	04-30-22	Shipped - Delivered to Buyer	
8	8	407-5443024-5233168	04-30-22	Cancelled	
9	9	402-4393761-0311520	04-30-22	Shipped	
10	10	407-5633625-6970741	04-30-22	Shipped	
11	11	171-4638481-6326716	04-30-22	Shipped	
12	12	405-5513694-8146768	04-30-22	Shipped - Delivered to Buyer	
13	13	408-7955685-3083534	04-30-22	Shipped	
14	14	408-1298370-1920302	04-30-22	Shipped - Delivered to Buyer	
15	15	403-4965581-9520319	04-30-22	Shipped - Delivered to Buyer	
16	16	406-9379318-6555504	04-30-22	Shipped	
17	17	405-9013803-8009918	04-30-22	Shipped	

18	18	402-4030358-5835511	04-30-22	Shipped - Delivered to Buyer
19	19	405-5957858-1051546	04-30-22	Shipped

	Fulfilment	Sales Channel	ship-service-level	Category	Size	Courier Status	\
0	Merchant	Amazon.in	Standard	T-shirt	S	On the Way	
1	Merchant	Amazon.in	Standard	Shirt	3XL	Shipped	
2	Amazon	Amazon.in	Expedited	Shirt	XL	Shipped	
3	Merchant	Amazon.in	Standard	Blazzer	L	On the Way	
4	Amazon	Amazon.in	Expedited	Trousers	3XL	Shipped	
5	Amazon	Amazon.in	Expedited	T-shirt	XL	Shipped	
6	Amazon	Amazon.in	Expedited	T-shirt	L	Shipped	
7	Merchant	Amazon.in	Standard	Shirt	S	Shipped	
8	Amazon	Amazon.in	Expedited	T-shirt	3XL	Cancelled	
9	Amazon	Amazon.in	Expedited	Shirt	XXL	Shipped	
10	Amazon	Amazon.in	Expedited	Shirt	S	Shipped	
11	Amazon	Amazon.in	Expedited	Shirt	XS	Shipped	
12	Merchant	Amazon.in	Standard	Shirt	XS	Shipped	
13	Amazon	Amazon.in	Expedited	T-shirt	XS	Shipped	
14	Merchant	Amazon.in	Standard	T-shirt	L	Shipped	
15	Merchant	Amazon.in	Standard	Shirt	6XL	Shipped	
16	Amazon	Amazon.in	Expedited	Shirt	XXL	Shipped	
17	Amazon	Amazon.in	Expedited	Shirt	XL	Shipped	
18	Merchant	Amazon.in	Standard	Shirt	XXL	Shipped	
19	Amazon	Amazon.in	Expedited	T-shirt	XS	Shipped	

	…	currency	Amount	ship-city	ship-state	ship-postal-code	\
0	…	INR	647.62	MUMBAI	MAHARASHTRA	400081.0	
1	…	INR	406.00	BENGALURU	KARNATAKA	560085.0	
2	…	INR	329.00	NAVI MUMBAI	MAHARASHTRA	410210.0	
3	…	INR	753.33	PUDUCHERRY	PUDUCHERRY	605008.0	
4	…	INR	574.00	CHENNAI	TAMIL NADU	600073.0	
5	…	INR	824.00	GHAZIABAD	UTTAR PRADESH	201102.0	
6	…	INR	653.00	CHANDIGARH	CHANDIGARH	160036.0	
7	…	INR	399.00	HYDERABAD	TELANGANA	500032.0	
8	…	NaN	NaN	HYDERABAD	TELANGANA	500008.0	
9	…	INR	363.00	Chennai	TAMIL NADU	600041.0	
10	…	INR	685.00	CHENNAI	TAMIL NADU	600073.0	
11	…	INR	364.00	NOIDA	UTTAR PRADESH	201303.0	
12	…	INR	399.00	Amravati.	MAHARASHTRA	444606.0	
13	…	INR	657.00	MUMBAI	MAHARASHTRA	400053.0	
14	…	INR	771.00	MUMBAI	MAHARASHTRA	400053.0	
15	…	INR	544.00	GUNTAKAL	ANDHRA PRADESH	515801.0	
16	…	INR	329.00	JAIPUR	RAJASTHAN	302020.0	
17	…	INR	399.00	NEW DELHI	DELHI	110074.0	
18	…	INR	458.00	Gurgaon	HARYANA	122004.0	
19	…	INR	886.00	BENGALURU	KARNATAKA	560017.0	

```
    ship-country    B2B  fulfilled-by New  PendingS
0             IN  False     Easy Ship NaN       NaN
1             IN  False     Easy Ship NaN       NaN
2             IN   True           NaN NaN       NaN
3             IN  False     Easy Ship NaN       NaN
4             IN  False           NaN NaN       NaN
5             IN  False           NaN NaN       NaN
6             IN  False           NaN NaN       NaN
7             IN  False     Easy Ship NaN       NaN
8             IN  False           NaN NaN       NaN
9             IN  False           NaN NaN       NaN
10            IN  False           NaN NaN       NaN
11            IN  False           NaN NaN       NaN
12            IN  False     Easy Ship NaN       NaN
13            IN  False           NaN NaN       NaN
14            IN  False     Easy Ship NaN       NaN
15            IN  False     Easy Ship NaN       NaN
16            IN  False           NaN NaN       NaN
17            IN  False           NaN NaN       NaN
18            IN  False     Easy Ship NaN       NaN
19            IN  False           NaN NaN       NaN

[20 rows x 21 columns]
```

[30]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 128976 entries, 0 to 128975
Data columns (total 21 columns):
 #   Column              Non-Null Count   Dtype
---  ------              --------------   -----
 0   index               128976 non-null  int64
 1   Order ID            128976 non-null  object
 2   Date                128976 non-null  object
 3   Status              128976 non-null  object
 4   Fulfilment          128976 non-null  object
 5   Sales Channel       128976 non-null  object
 6   ship-service-level  128976 non-null  object
 7   Category            128976 non-null  object
 8   Size                128976 non-null  object
 9   Courier Status      128976 non-null  object
 10  Qty                 128976 non-null  int64
 11  currency            121176 non-null  object
 12  Amount              121176 non-null  float64
 13  ship-city           128941 non-null  object
 14  ship-state          128941 non-null  object
 15  ship-postal-code    128941 non-null  float64
```

```
 16  ship-country        128941 non-null  object
 17  B2B                 128976 non-null  bool
 18  fulfilled-by        39263 non-null   object
 19  New                 0 non-null       float64
 20  PendingS            0 non-null       float64
dtypes: bool(1), float64(4), int64(2), object(14)
memory usage: 19.8+ MB
```

[31]:
```
#drop blank columns
df.drop(['New','PendingS'],axis=1,inplace=True)
```

[32]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 128976 entries, 0 to 128975
Data columns (total 19 columns):
 #   Column              Non-Null Count   Dtype
---  ------              --------------   -----
 0   index               128976 non-null  int64
 1   Order ID            128976 non-null  object
 2   Date                128976 non-null  object
 3   Status              128976 non-null  object
 4   Fulfilment          128976 non-null  object
 5   Sales Channel       128976 non-null  object
 6   ship-service-level  128976 non-null  object
 7   Category            128976 non-null  object
 8   Size                128976 non-null  object
 9   Courier Status      128976 non-null  object
 10  Qty                 128976 non-null  int64
 11  currency            121176 non-null  object
 12  Amount              121176 non-null  float64
 13  ship-city           128941 non-null  object
 14  ship-state          128941 non-null  object
 15  ship-postal-code    128941 non-null  float64
 16  ship-country        128941 non-null  object
 17  B2B                 128976 non-null  bool
 18  fulfilled-by        39263 non-null   object
dtypes: bool(1), float64(2), int64(2), object(14)
memory usage: 17.8+ MB
```

[33]:
```
#checking null value
pd.isnull(df)
```

[33]:

	index	Order ID	Date	Status	Fulfilment	Sales Channel	\
0	False	False	False	False	False	False	
1	False	False	False	False	False	False	
2	False	False	False	False	False	False	

```
3       False     False  False   False       False          False
4       False     False  False   False       False          False
…       …       …    …     …         …            …
128971  False     False  False   False       False          False
128972  False     False  False   False       False          False
128973  False     False  False   False       False          False
128974  False     False  False   False       False          False
128975  False     False  False   False       False          False

        ship-service-level  Category   Size  Courier Status    Qty  currency  \
0                    False     False  False           False  False     False
1                    False     False  False           False  False     False
2                    False     False  False           False  False     False
3                    False     False  False           False  False     False
4                    False     False  False           False  False     False
…                    …       …    …              …     …       …
128971               False     False  False           False  False     False
128972               False     False  False           False  False     False
128973               False     False  False           False  False     False
128974               False     False  False           False  False     False
128975               False     False  False           False  False     False

        Amount  ship-city  ship-state  ship-postal-code  ship-country    B2B  \
0        False      False       False             False         False  False
1        False      False       False             False         False  False
2        False      False       False             False         False  False
3        False      False       False             False         False  False
4        False      False       False             False         False  False
…        …        …         …               …           …     …
128971   False      False       False             False         False  False
128972   False      False       False             False         False  False
128973   False      False       False             False         False  False
128974   False      False       False             False         False  False
128975   False      False       False             False         False  False

        fulfilled-by
0              False
1              False
2               True
3              False
4               True
…              …
128971          True
128972          True
128973          True
128974          True
128975          True
```

```
[128976 rows x 19 columns]
```

[34]:
```
#sum will give total values of null values
pd.isnull(df).sum()
```

[34]:
```
index                    0
Order ID                 0
Date                     0
Status                   0
Fulfilment               0
Sales Channel            0
ship-service-level       0
Category                 0
Size                     0
Courier Status           0
Qty                      0
currency              7800
Amount                7800
ship-city               35
ship-state              35
ship-postal-code        35
ship-country            35
B2B                      0
fulfilled-by         89713
dtype: int64
```

[36]:
```
df.shape
```

[36]:
```
(128976, 19)
```

[37]:
```
#drop null values
df.dropna(inplace=True)
```

[38]:
```
df.shape
```

[38]:
```
(37514, 19)
```

[39]:
```
df.columns
```

[39]:
```
Index(['index', 'Order ID', 'Date', 'Status', 'Fulfilment', 'Sales Channel',
       'ship-service-level', 'Category', 'Size', 'Courier Status', 'Qty',
       'currency', 'Amount', 'ship-city', 'ship-state', 'ship-postal-code',
       'ship-country', 'B2B', 'fulfilled-by'],
      dtype='object')
```

[40]:
```
#change data type
df['ship-postal-code']=df['ship-postal-code'].astype('int')
```

[49]:
```
#checking whether the data type change or not
df['ship-postal-code'].dtype
```

[49]:
```
dtype('int32')
```

[42]:
```
df['Date']=pd.to_datetime(df['Date'])
```

```
C:\Users\RAVI\AppData\Local\Temp\ipykernel_15616\3023999556.py:1: UserWarning:
Could not infer format, so each element will be parsed individually, falling
back to `dateutil`. To ensure parsing is consistent and as-expected, please
specify a format.
  df['Date']=pd.to_datetime(df['Date'])
```

[44]:
```
df['Date'].dtype
```

[44]:
```
dtype('<M8[ns]')
```

[47]:
```
#rename columns
df.rename(columns={'Qty':'Quantity'},inplace=True)
```

[48]:
```
df.columns
```

[48]:
```
Index(['index', 'Order ID', 'Date', 'Status', 'Fulfilment', 'Sales Channel',
       'ship-service-level', 'Category', 'Size', 'Courier Status', 'Quantity',
       'currency', 'Amount', 'ship-city', 'ship-state', 'ship-postal-code',
       'ship-country', 'B2B', 'fulfilled-by'],
      dtype='object')
```

[50]:
```
df.describe()
```

[50]:

	index	Date	Quantity
count	37514.000000	37514	37514.000000
mean	60953.809858	2022-05-11 07:56:47.303939840	0.867383
min	0.000000	2022-03-31 00:00:00	0.000000
25%	27235.250000	2022-04-20 00:00:00	1.000000
50%	63470.500000	2022-05-09 00:00:00	1.000000
75%	91790.750000	2022-06-01 00:00:00	1.000000
max	128891.000000	2022-06-29 00:00:00	5.000000
std	36844.853039	NaN	0.354160

	Amount	ship-postal-code
count	37514.000000	37514.000000
mean	646.553960	463291.552754
min	0.000000	110001.000000

```
25%      458.000000     370465.000000
50%      629.000000     500019.000000
75%      771.000000     600042.000000
max     5495.000000     989898.000000
std      279.952414     194550.425637
```

[51]:
```
df.describe(include='object')
```

[51]:
```
                   Order ID                         Status Fulfilment  \
count                 37514                          37514      37514
unique                34664                             11          1
top     171-5057375-2831560  Shipped - Delivered to Buyer   Merchant
freq                     12                          28741      37514

       Sales Channel ship-service-level Category   Size Courier Status  \
count          37514              37514    37514  37514          37514
unique             1                  1        8     11              3
top        Amazon.in           Standard  T-shirt      M        Shipped
freq           37514              37514    14062   6806          31859

       currency  ship-city   ship-state ship-country fulfilled-by
count     37514      37514        37514        37514        37514
unique        1       4698           58            1            1
top         INR  BENGALURU  MAHARASHTRA           IN    Easy Ship
freq      37514       2839         6236        37514        37514
```

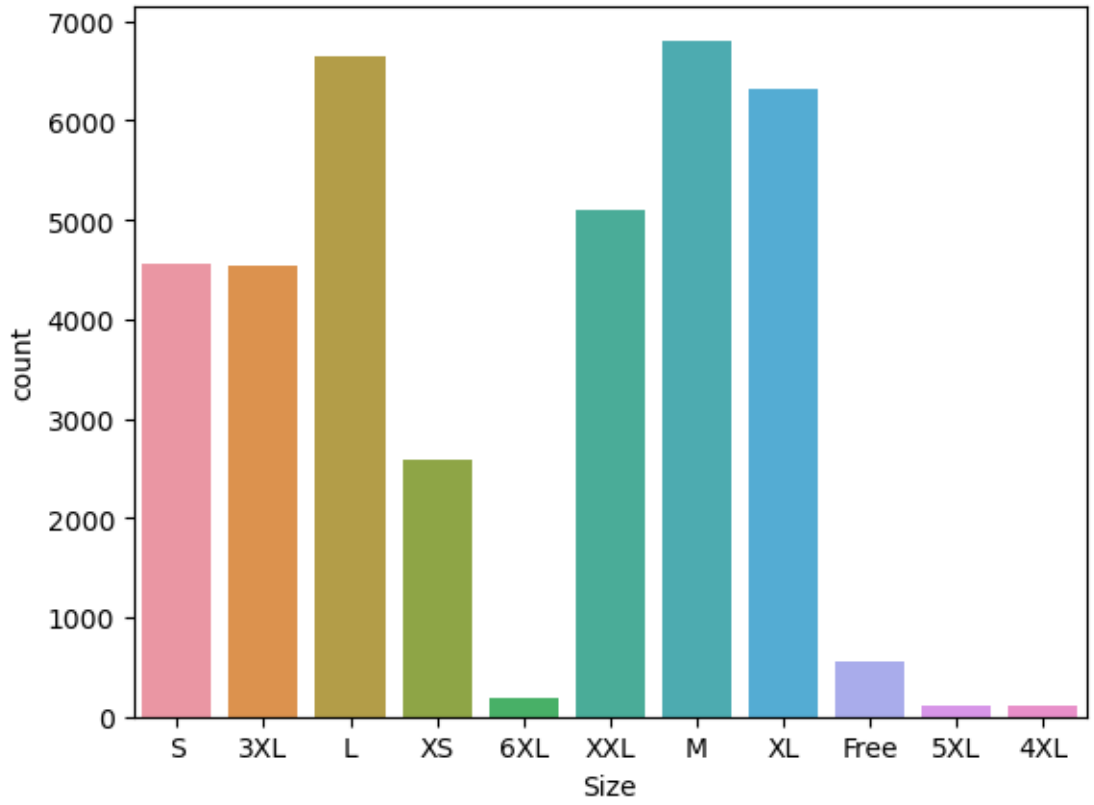
[52]:
```
#use describe() for specific columns
df[['Quantity','Amount']].describe()
```

[52]:
```
           Quantity        Amount
count  37514.000000  37514.000000
mean       0.867383    646.553960
std        0.354160    279.952414
min        0.000000      0.000000
25%        1.000000    458.000000
50%        1.000000    629.000000
75%        1.000000    771.000000
max        5.000000   5495.000000
```

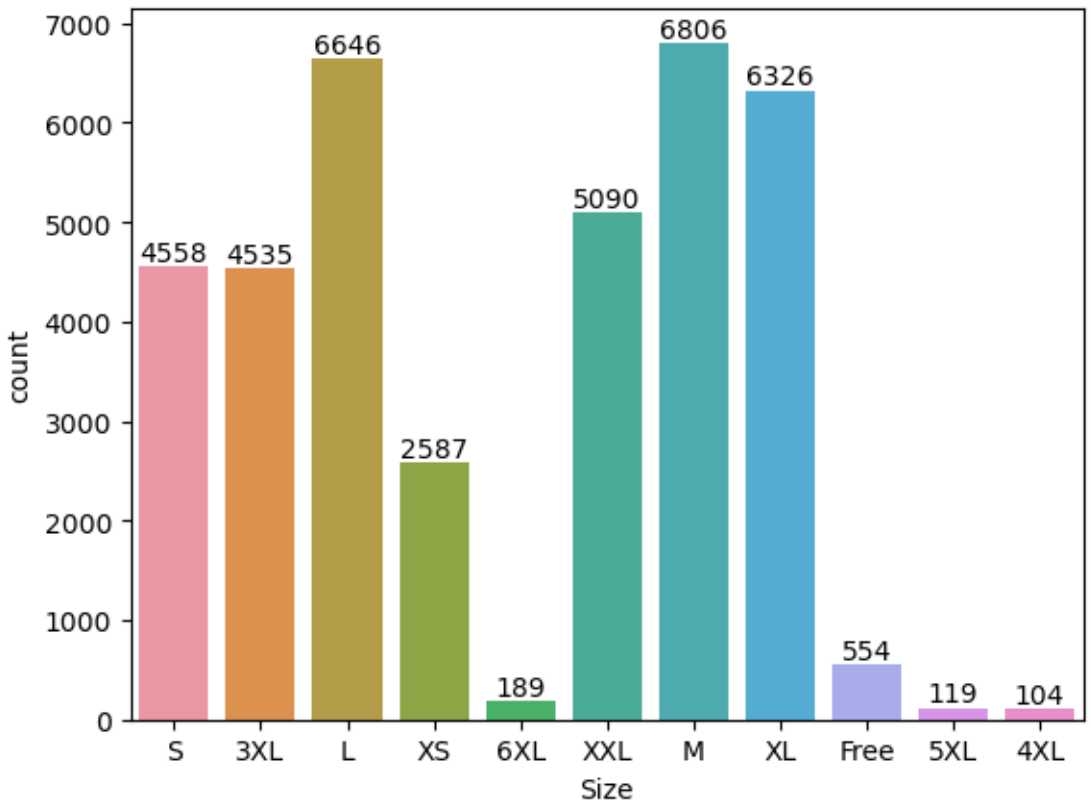
[53]:
```
df.columns
```

[53]:
```
Index(['index', 'Order ID', 'Date', 'Status', 'Fulfilment', 'Sales Channel',
       'ship-service-level', 'Category', 'Size', 'Courier Status', 'Quantity',
       'currency', 'Amount', 'ship-city', 'ship-state', 'ship-postal-code',
       'ship-country', 'B2B', 'fulfilled-by'],
      dtype='object')
```

[56]:
```
ax=sns.countplot(x='Size',data=df)
```



[57]:
```
ax=sns.countplot(x='Size',data=df)
for bars in ax.containers:
    ax.bar_label(bars)
```



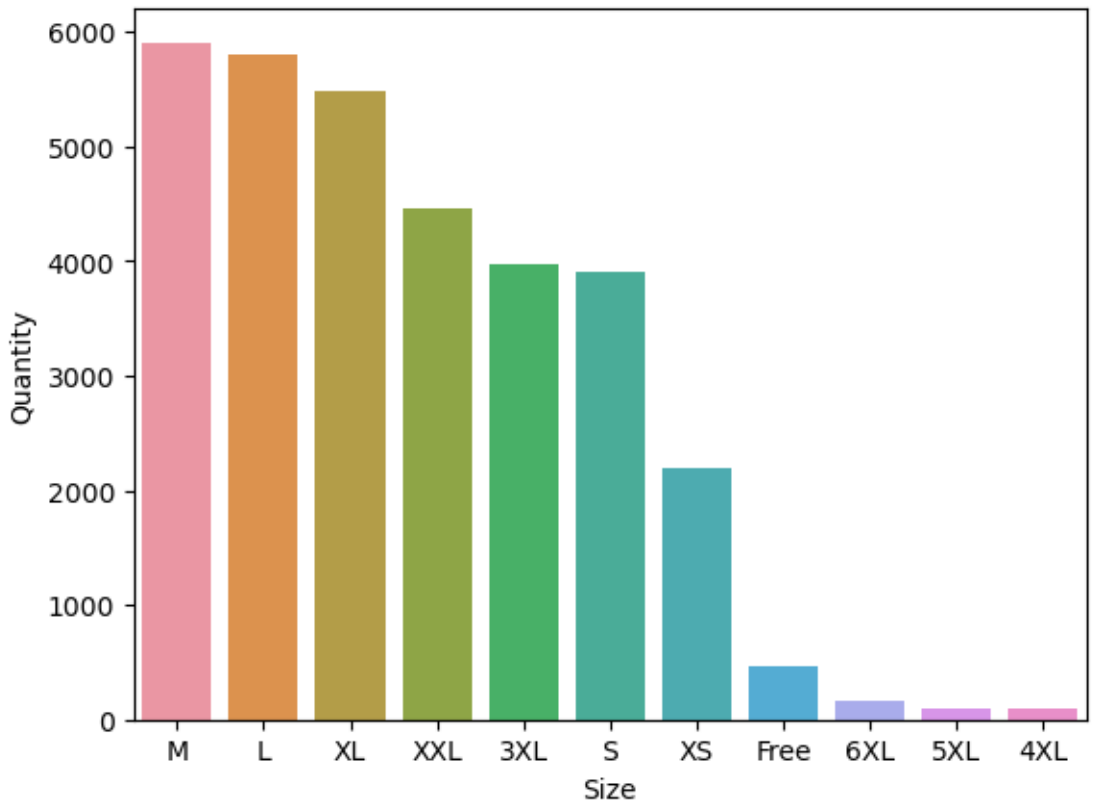
[58]:
```
df.groupby(['Size'],as_index=False)['Quantity'].sum().
  ↪sort_values(by='Quantity',ascending=False)
```

[58]:

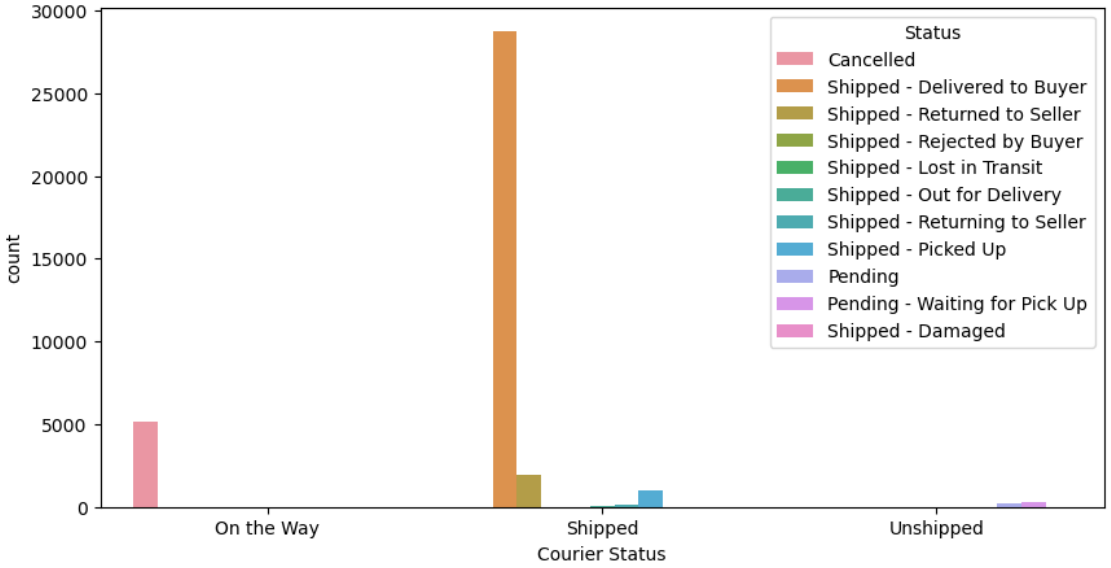
	Size	Quantity
6	M	5905
5	L	5795
8	XL	5481
10	XXL	4465
0	3XL	3972
7	S	3896
9	XS	2191
4	Free	467
3	6XL	170
2	5XL	104
1	4XL	93

[59]:
```
size_qty=df.groupby(['Size'],as_index=False)['Quantity'].sum().
  ↪sort_values(by='Quantity',ascending=False)
sns.barplot(x='Size',y='Quantity',data=size_qty)
```

[59]: <Axes: xlabel='Size', ylabel='Quantity'>

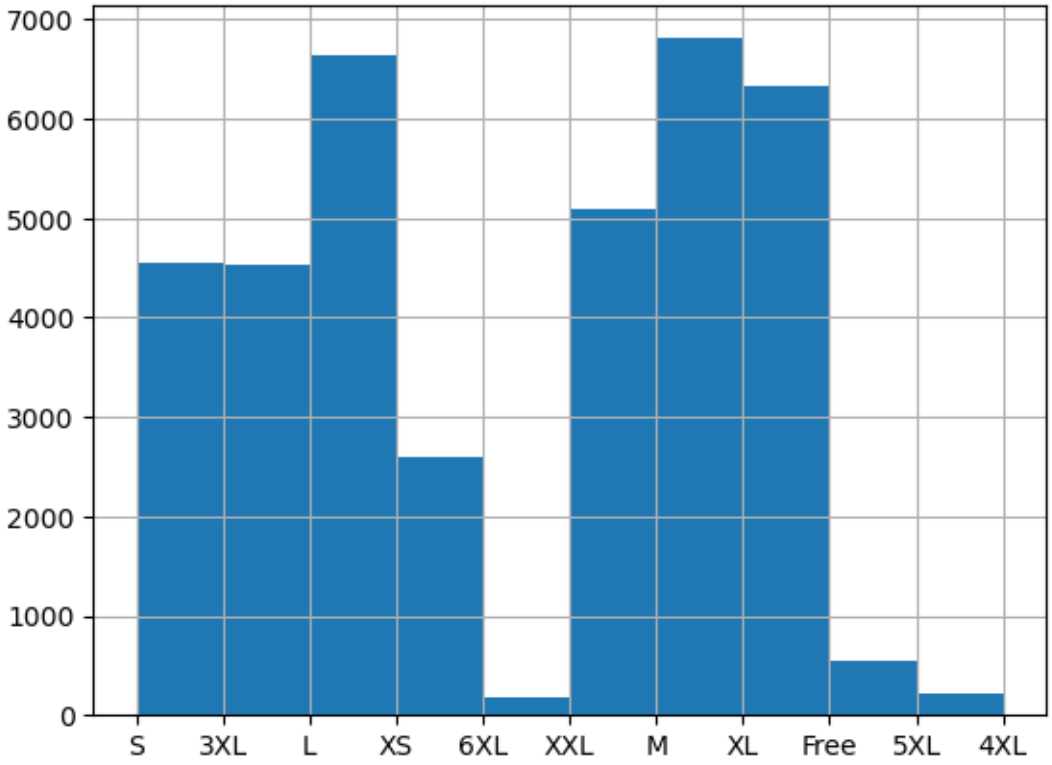


[61]:
```
plt.figure(figsize=(10,5))
ax=sns.countplot(data=df,x='Courier Status',hue='Status')
plt.show()
```

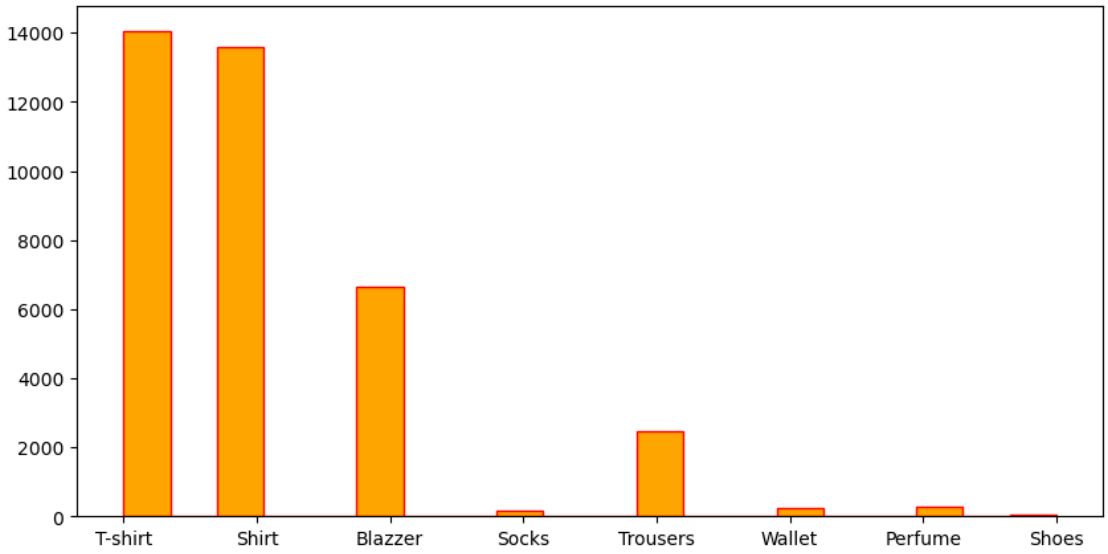


[62]:
```
#histogram
df['Size'].hist()
```

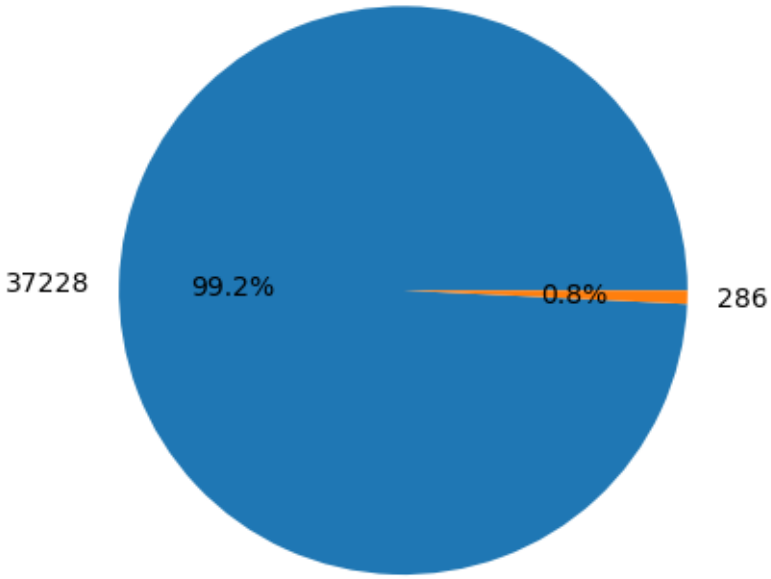
[62]: <Axes: >



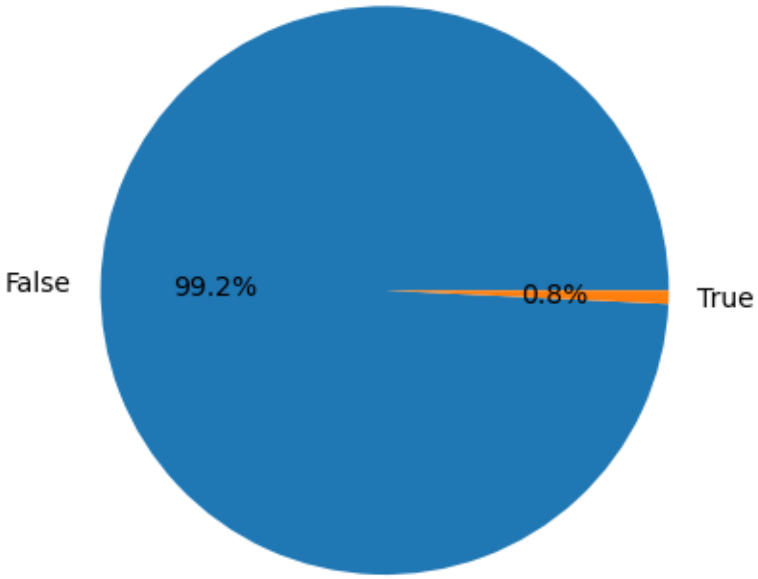
[70]:
```
df['Category']=df['Category'].astype(str)
column_data=df['Category']
plt.figure(figsize=(10,5))
plt.hist(column_data,bins=20,color= 'orange',edgecolor='red')
plt.xticks(rotation=360)
plt.show()
```



[73]:
```
#checking B2B data by using pie chart
B2B_check=df['B2B'].value_counts()
#plot the pie chart
plt.pie(B2B_check, labels=B2B_check,autopct='%1.1f%%')
#plt.axis('equal')
plt.show()
```

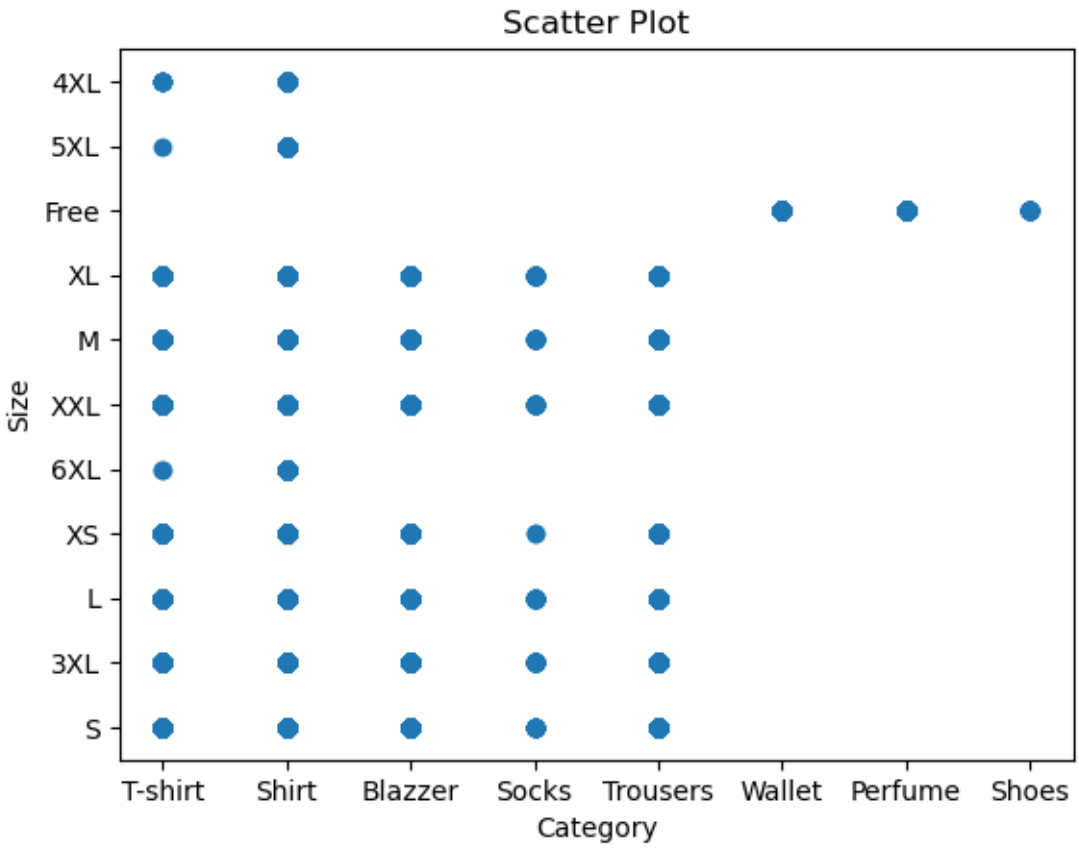


[74]:
```
#checking B2B data by using pie chart
B2B_check=df['B2B'].value_counts()
#plot the pie chart
plt.pie(B2B_check, labels=B2B_check.index,autopct='%1.1f%%')
#plt.axis('equal')
plt.show()
```



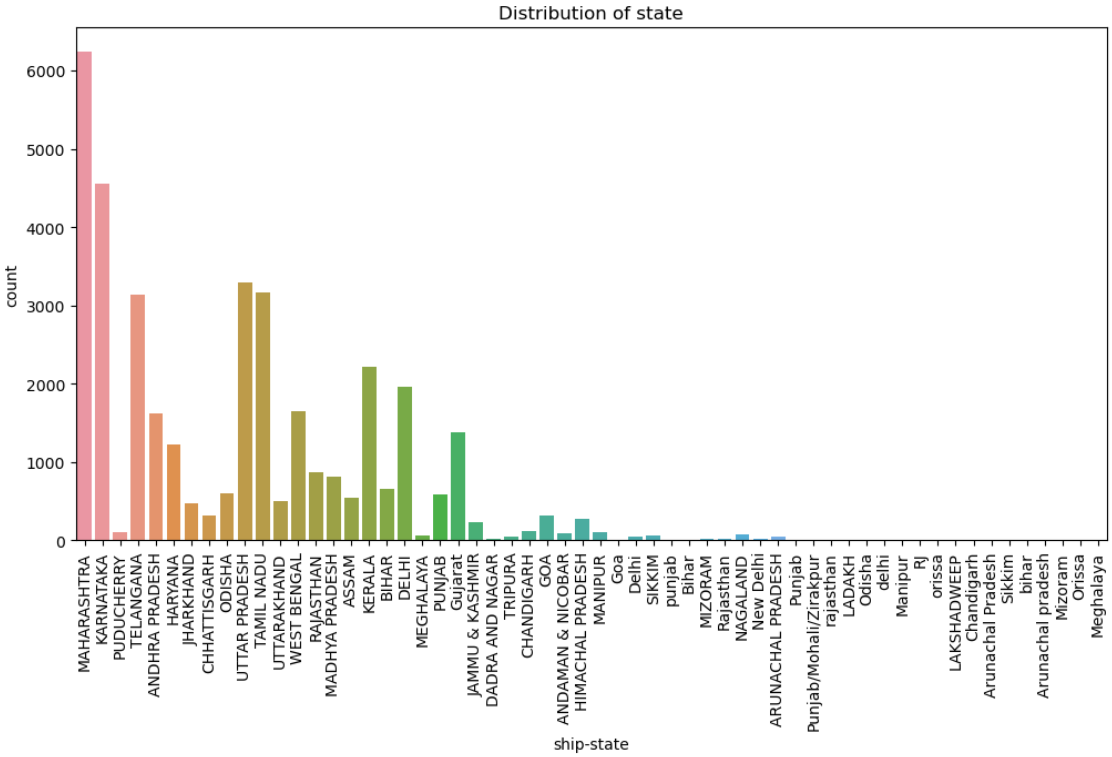
[75]:
```
#prepare data for scatter plot
x_data=df['Category']
y_data=df['Size']

#plot the scatterplot
plt.scatter(x_data,y_data)
plt.xlabel('Category')
plt.ylabel('Size')
plt.title('Scatter Plot')
plt.show()
```

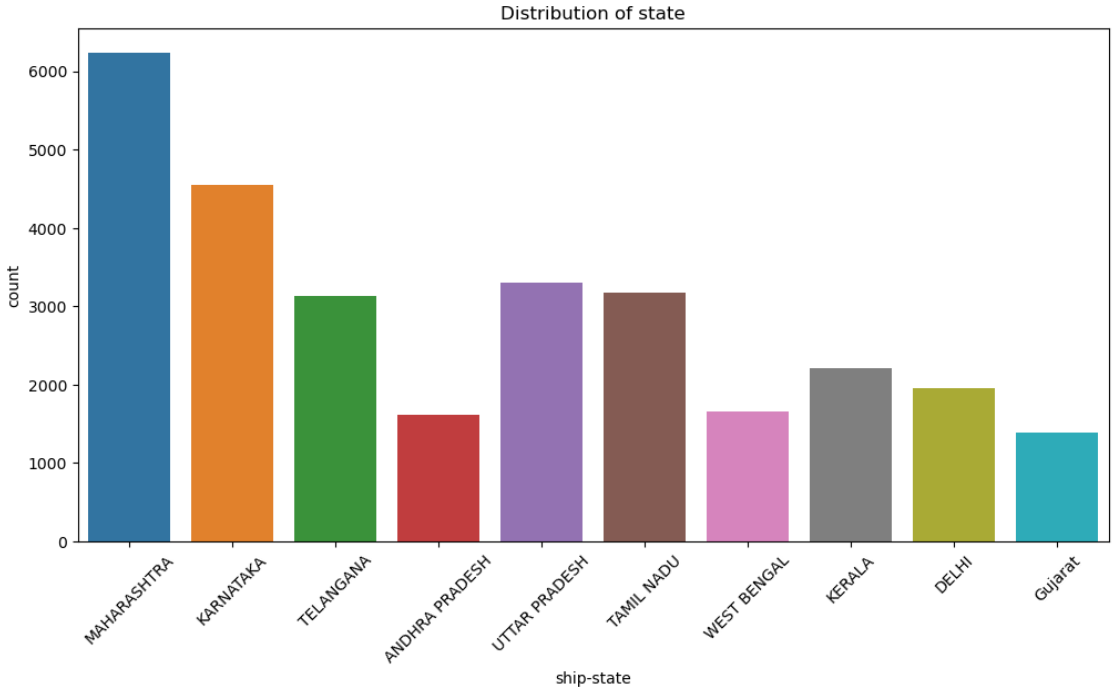


[76]:
```
#plot count of cities by state

plt.figure(figsize=(12,6))
sns.countplot(data=df,x='ship-state')
plt.xlabel('ship-state')
plt.ylabel('count')
plt.title('Distribution of state')
plt.xticks(rotation=90)
plt.show()
```



[79]:
```
#top_10_states
top_10_states =df['ship-state'].value_counts().head(10)
#plot count of cities by state
plt.figure(figsize=(12,6))
sns.countplot(data=df[df['ship-state'].isin(top_10_states.
  ↪index)],x='ship-state')
plt.xlabel('ship-state')
plt.ylabel('count')
plt.title('Distribution of state')
plt.xticks(rotation=45)
plt.show()
```


Distribution of state
count
6000
5000
4000
3000
2000
1000
0
MAHARASHTRA
KARNATAKA
TELANGANA
ANDHRA PRADESH
UTTAR PRADESH
TAMIL NADU
WEST BENGAL
KERALA
DELHI
Gujarat
ship-state

[ ]: