Amazon Sales Report

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
```

In [2]:
```
df = pd.read_csv('Amazon Sale Report.csv',encoding = 'unicode_escape')
```

In [3]:
```
df.shape
```

Out[3]: (128976, 21)

In [4]:
```
1 df.head(10)
```

Out[4]:

	index	Order ID	Date	Status	Fulfilment	Sales Channel	ship-service-level	Category	Size	Courier Status	...	currency	Amount	ship-city	ship-
0	0	405-8078784-5731545	04-30-22	Cancelled	Merchant	Amazon.in	Standard	T-shirt	S	On the Way	...	INR	647.62	MUMBAI	MAHARASI
1	1	171-9198151-1101146	04-30-22	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	Shirt	3XL	Shipped	...	INR	406.00	BENGALURU	KARNA
2	2	404-0687676-7273146	04-30-22	Shipped	Amazon	Amazon.in	Expedited	Shirt	XL	Shipped	...	INR	329.00	NAVI MUMBAI	MAHARASI
3	3	403-9615377-8133951	04-30-22	Cancelled	Merchant	Amazon.in	Standard	Blazzer	L	On the Way	...	INR	753.33	PUDUCHERRY	PUDUCHE
4	4	407-1069790-7240320	04-30-22	Shipped	Amazon	Amazon.in	Expedited	Trousers	3XL	Shipped	...	INR	574.00	CHENNAI	TAMIL N
5	5	404-1490984-4578765	04-30-22	Shipped	Amazon	Amazon.in	Expedited	T-shirt	XL	Shipped	...	INR	824.00	GHAZIABAD	U PRAI
6	6	408-5748499-6859555	04-30-22	Shipped	Amazon	Amazon.in	Expedited	T-shirt	L	Shipped	...	INR	653.00	CHANDIGARH	CHANDIC
7	7	406-7807733-3785945	04-30-22	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	Shirt	S	Shipped	...	INR	399.00	HYDERABAD	TELANC
8	8	407-5443024-5233168	04-30-22	Cancelled	Amazon	Amazon.in	Expedited	T-shirt	3XL	Cancelled	...	NaN	NaN	HYDERABAD	TELANC
9	9	402-4393761-0311520	04-30-22	Shipped	Amazon	Amazon.in	Expedited	Shirt	XXL	Shipped	...	INR	363.00	Chennai	TAMIL N

10 rows × 21 columns

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

Out[5]:

	index	Order ID	Date	Status	Fulfilment	Sales Channel	ship-service-level	Category	Size	Courier Status	...	currency	Amount	ship-city	sh
128971	128970	406-6001380-7673107	05-31-22	Shipped	Amazon	Amazon.in	Expedited	Shirt	XL	Shipped	...	INR	517.0	HYDERABAD	TELA
128972	128971	402-9551604-7544318	05-31-22	Shipped	Amazon	Amazon.in	Expedited	T-shirt	M	Shipped	...	INR	999.0	GURUGRAM	HA
128973	128972	407-9547469-3152358	05-31-22	Shipped	Amazon	Amazon.in	Expedited	Blazzer	XXL	Shipped	...	INR	690.0	HYDERABAD	TELA
128974	128973	402-6184140-0545956	05-31-22	Shipped	Amazon	Amazon.in	Expedited	T-shirt	XS	Shipped	...	INR	1199.0	Halol	
128975	128974	408-7436540-8728312	05-31-22	Shipped	Amazon	Amazon.in	Expedited	T-shirt	S	Shipped	...	INR	696.0	Raipur	CHHATT

5 rows × 21 columns

In [6]:
```
1 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
```

In [7]:
```
1 #drop unrelated/blank columns
2 df.drop(['New','PendingS'], axis=1, inplace=True)
```

In [8]:
```
1 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
```

In [9]:
```
pd.isnull(df)
# checking null values
```

Out[9]:

	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	sh cour
0	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	Fa
1	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	Fa
2	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	Fa
3	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	Fa
4	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	Fa
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
128971	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	Fa
128972	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	Fa
128973	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	Fa
128974	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	Fa
128975	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	Fa

128976 rows × 19 columns

In [10]:
```
pd.isnull(df).sum()
# sum will give total values of null values
```

Out[10]:
```
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
```

In [11]:
```
df.shape
```

Out[11]:
```
(128976, 19)
```

In [12]:
```
#drop null values
df.dropna(inplace=True)
```

In [13]:
```
df.shape
```

Out[13]:
```
(37514, 19)
```

In [14]:
```
1 df.columns
```

Out[14]:
```
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')
```

In [15]:
```
1 #change data type
2 df['ship-postal-code']=df['ship-postal-code'].astype('int')
```

In [16]:
```
1 #checking whelther the data type change or not
2 df['ship-postal-code'].dtype
```

Out[16]:
```
dtype('int32')
```

In [17]:
```
1 df['Date']=pd.to_datetime(df['Date'])
```

In [18]:
```
1 df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 37514 entries, 0 to 128892
Data columns (total 19 columns):
 #   Column              Non-Null Count  Dtype         
---  ------              --------------  -----         
 0   index               37514 non-null  int64         
 1   Order ID            37514 non-null  object        
 2   Date                37514 non-null  datetime64[ns]
 3   Status              37514 non-null  object        
 4   Fulfilment          37514 non-null  object        
 5   Sales Channel       37514 non-null  object        
 6   ship-service-level  37514 non-null  object        
 7   Category            37514 non-null  object        
 8   Size                37514 non-null  object        
 9   Courier Status      37514 non-null  object        
 10  Qty                 37514 non-null  int64         
 11  currency            37514 non-null  object        
 12  Amount              37514 non-null  float64       
 13  ship-city           37514 non-null  object        
 14  ship-state          37514 non-null  object        
 15  ship-postal-code    37514 non-null  int32         
 16  ship-country        37514 non-null  object        
 17  B2B                 37514 non-null  bool          
 18  fulfilled-by        37514 non-null  object        
dtypes: bool(1), datetime64[ns](1), float64(1), int32(1), int64(2), object(13)
memory usage: 5.3+ MB
```

In [19]:
```
1 df.columns
```

Out[19]:
```
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')
```

In [20]:

```
#rename Columns
df.rename(columns={'Qty':'Quantity'})
```

Out[20]:

	index	Order ID	Date	Status	Fulfilment	Sales Channel	ship-service-level	Category	Size	Courier Status	Quantity	currency	Amount	ship-city
0	0	405-8078784-5731545	2022-04-30	Cancelled	Merchant	Amazon.in	Standard	T-shirt	S	On the Way	0	INR	647.62	MUMBAI
1	1	171-9198151-1101146	2022-04-30	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	Shirt	3XL	Shipped	1	INR	406.00	BENGALURU
3	3	403-9615377-8133951	2022-04-30	Cancelled	Merchant	Amazon.in	Standard	Blazzer	L	On the Way	0	INR	753.33	PUDUCHERRY
7	7	406-7807733-3785945	2022-04-30	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	Shirt	S	Shipped	1	INR	399.00	HYDERABAD
12	12	405-5513694-8146768	2022-04-30	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	Shirt	XS	Shipped	1	INR	399.00	Amravati.
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
128875	128874	405-4724097-1016369	2022-06-01	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	T-shirt	S	Shipped	1	INR	854.00	ALLUR
128876	128875	403-9524128-9243508	2022-06-01	Cancelled	Merchant	Amazon.in	Standard	Blazzer	XL	On the Way	0	INR	734.29	Barabanki
128888	128887	405-6493630-8542756	2022-05-31	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	Trousers	M	Shipped	1	INR	518.00	NOIDA
128891	128890	407-0116398-1810752	2022-05-31	Cancelled	Merchant	Amazon.in	Standard	Wallet	Free	On the Way	0	INR	398.10	MADURAI
128892	128891	403-0317423-9322704	2022-05-31	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	Blazzer	M	Shipped	1	INR	721.00	UTTAR BAGDOGRA

37514 rows × 19 columns

In [21]:
```
#describe() method return description of the data in the Dataframe(i.e count,mean,std,min.etc)
df.describe()
```

Out[21]:

	index	Qty	Amount	ship-postal-code
count	37514.000000	37514.000000	37514.000000	37514.000000
mean	60953.809858	0.867383	646.553960	463291.552754
std	36844.853039	0.354160	279.952414	194550.425637
min	0.000000	0.000000	0.000000	110001.000000
25%	27235.250000	1.000000	458.000000	370465.000000
50%	63470.500000	1.000000	629.000000	500019.000000
75%	91790.750000	1.000000	771.000000	600042.000000
max	128891.000000	5.000000	5495.000000	989898.000000

In [22]:
```
df.describe(include='object')
```

Out[22]:

	Order ID	Status	Fulfilment	Sales Channel	ship-service-level	Category	Size	Courier Status	currency	ship-city	ship-state	ship-country	fulfilled-by
count	37514	37514	37514	37514	37514	37514	37514	37514	37514	37514	37514	37514	37514
unique	34664	11	1	1	1	8	11	3	1	4698	58	1	1
top	171-5057375-2831560	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	T-shirt	M	Shipped	INR	BENGALURU	MAHARASHTRA	IN	Easy Ship
freq	12	28741	37514	37514	37514	14062	6806	31859	37514	2839	6236	37514	37514

In [23]:
```
#use describe() for specific columns
df[['Qty','Amount']].describe()
```

Out[23]:

	Qty	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

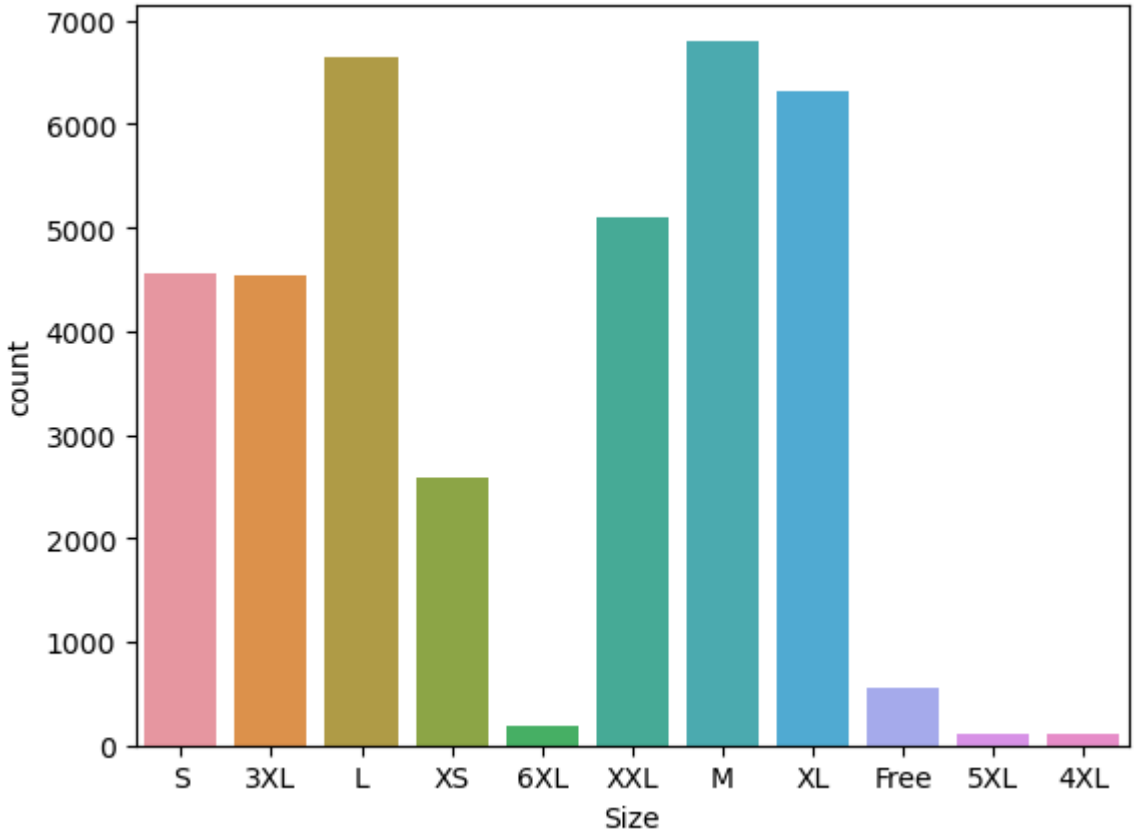
Exploratory Data Analysis

In [24]:
```
df.columns
```

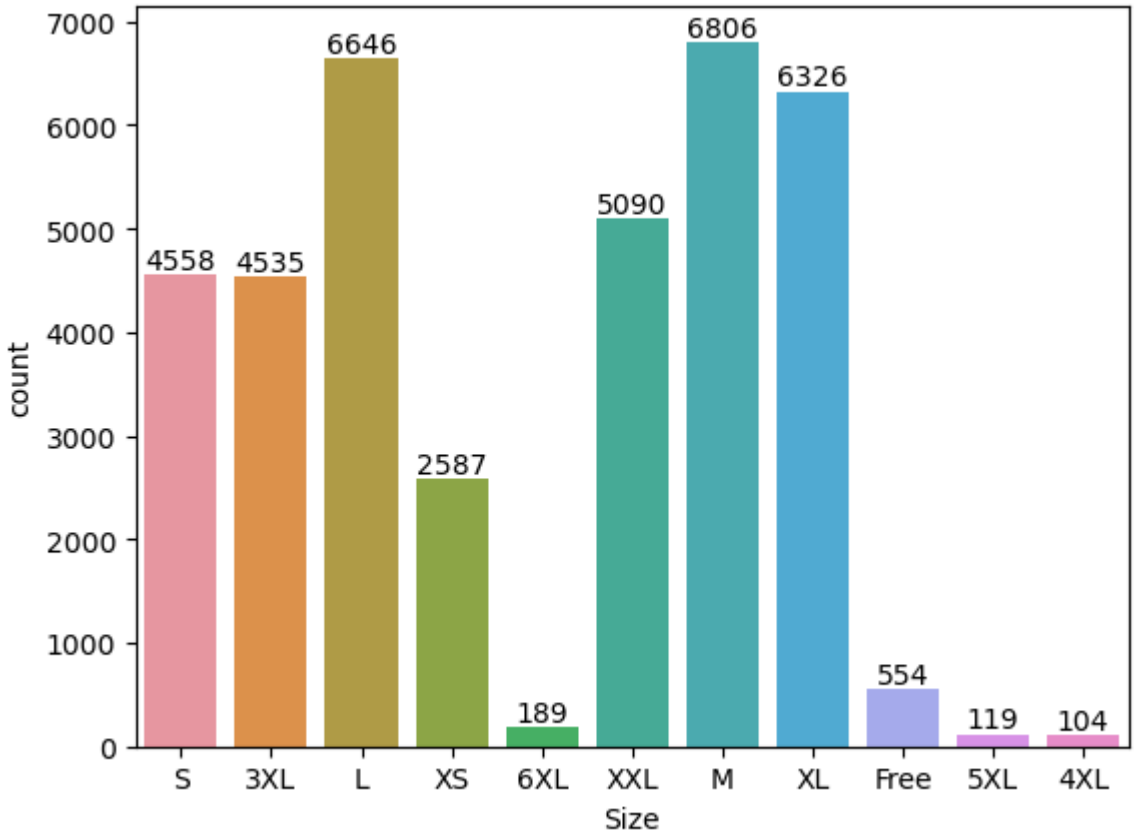
Out[24]:
```
Index(['index', 'Order ID', 'Date', 'Status', 'Fulfilment', 'Sales Channel',
       'ship-service-level', 'Category', 'Size', 'Courier Status', 'Qty',
       'currency', 'Amount', 'ship-city', 'ship-state', 'ship-postal-code',
       'ship-country', 'B2B', 'fulfilled-by'],
      dtype='object')
```

Size

In [25]:
```
1 ax=sns.countplot(x='Size' , data=df)
```



In [26]:
```
ax=sns.countplot(x='Size' , data=df)

for bars in ax.containers:
    ax.bar_label(bars)
```



Note: From above Graph you can see that most of the people buys M-size

Graph By:-

The groupby() function in pandas is used to group data based on one or more columns in a DataFrame

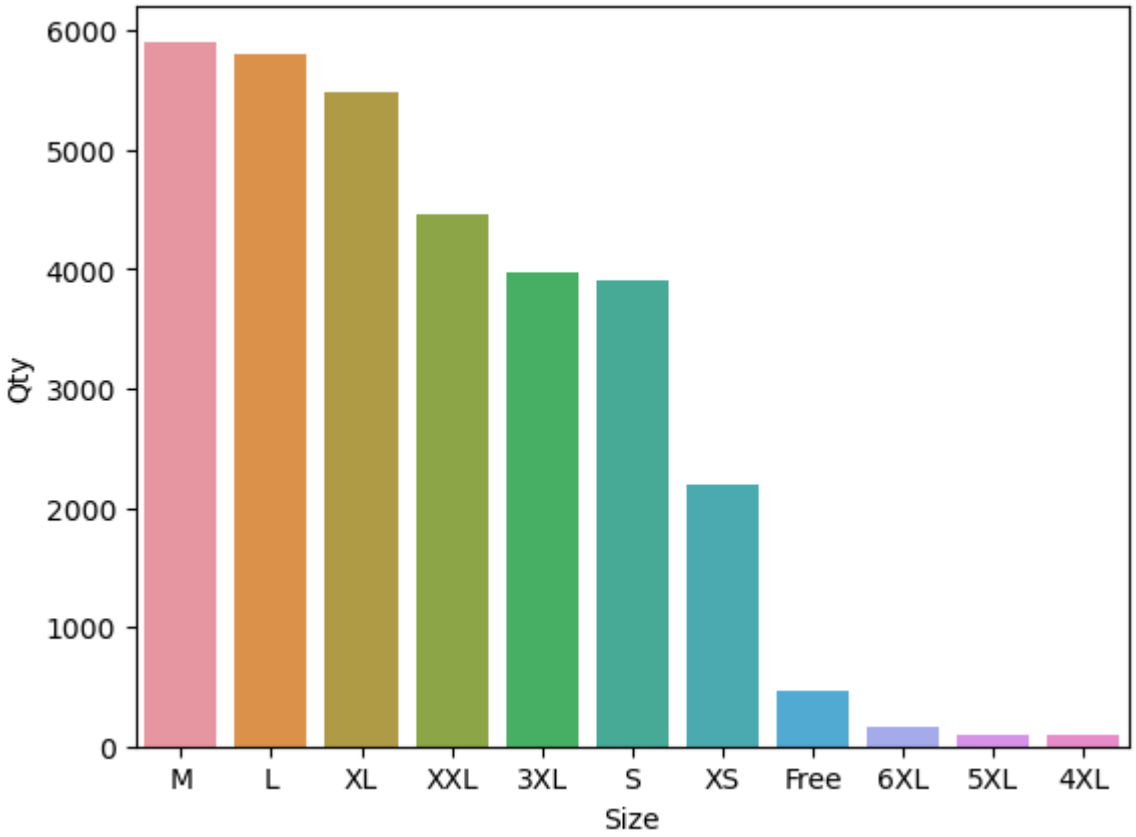
In [27]:
```
1 df.groupby(['Size'], as_index=False)['Qty'].sum().sort_values(by='Qty',ascending=False)
```

Out[27]:

	Size	Qty
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

In [28]:
```
S_Qty=df.groupby(['Size'], as_index=False)['Qty'].sum().sort_values(by='Qty',ascending=False)

sns.barplot(x='Size',y='Qty', data=S_Qty)
```

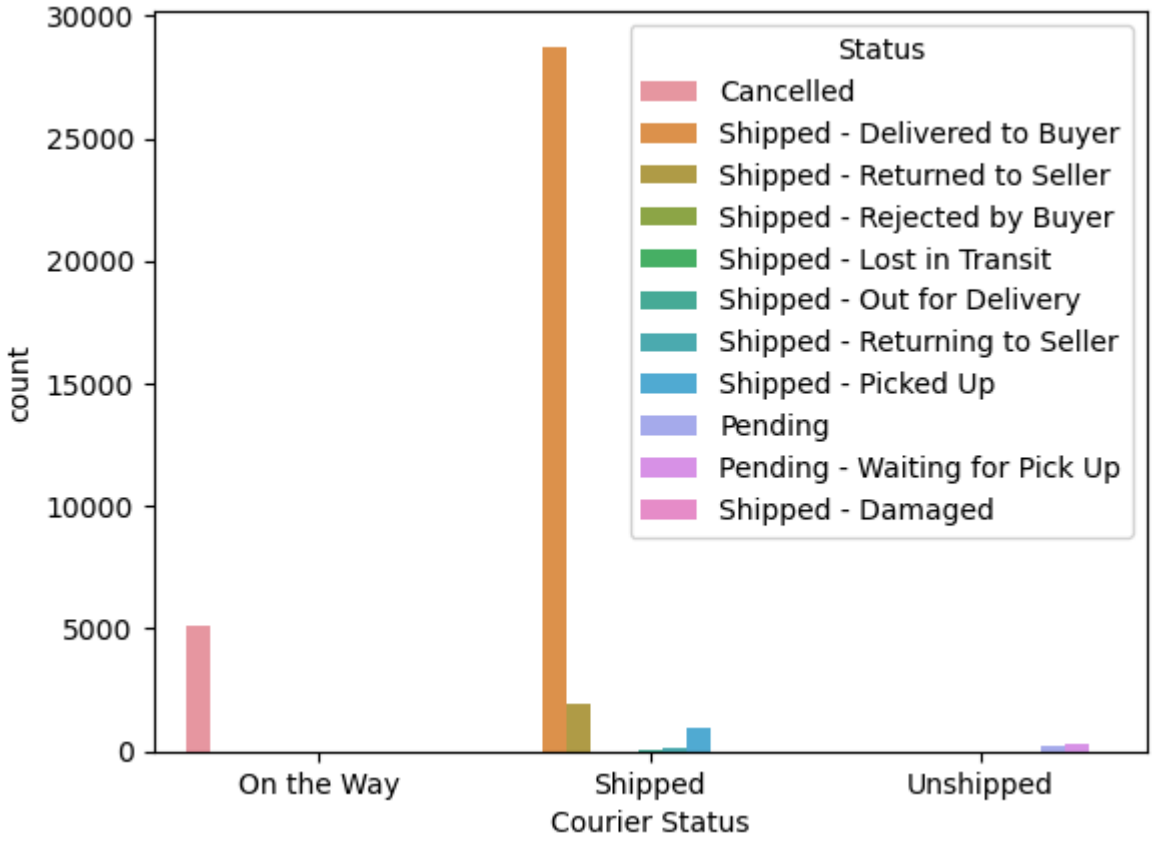
Out[28]: <Axes: xlabel='Size', ylabel='Qty'>



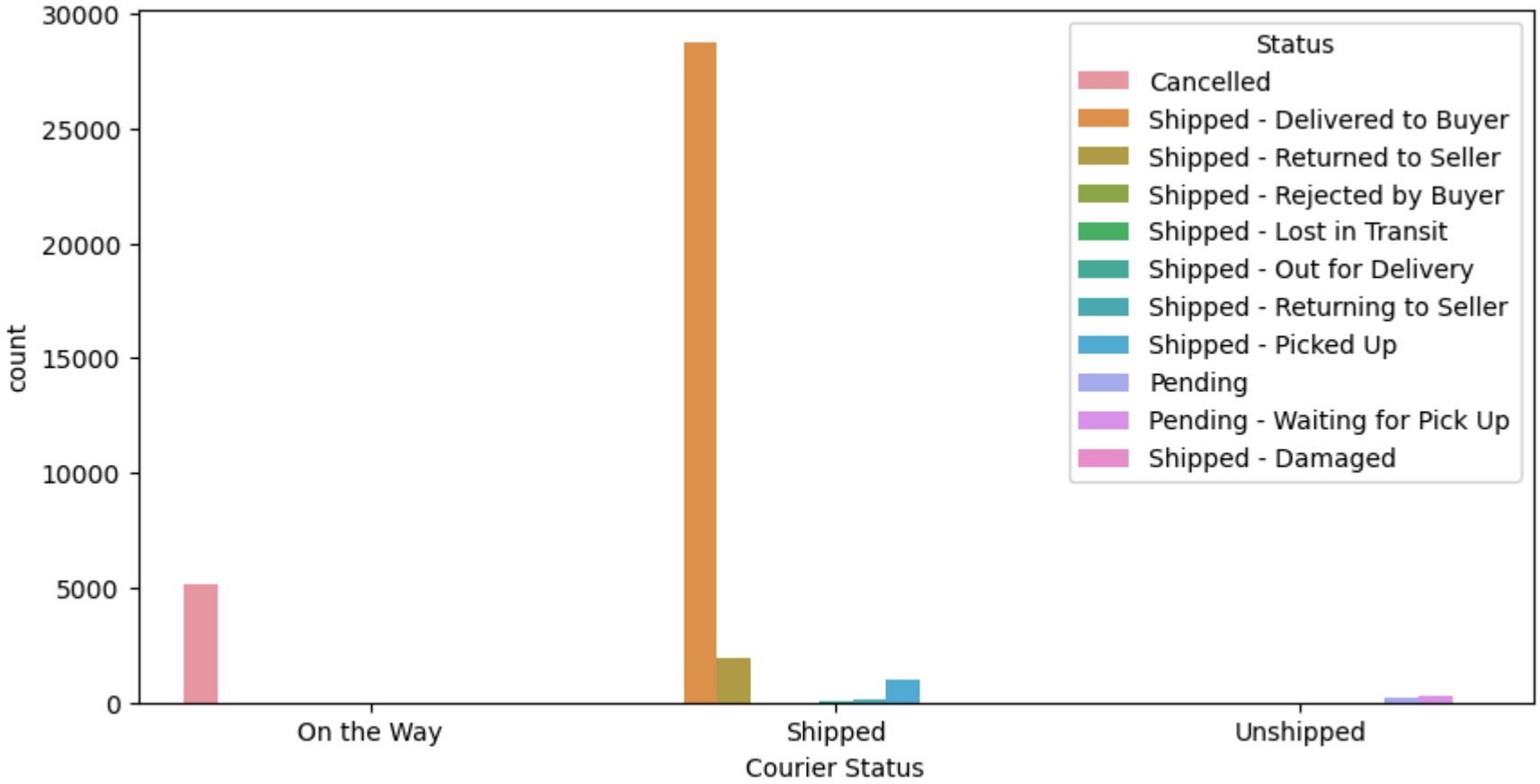
Note: From above Graph you can see that most of the Qty buys M-Size in the sales

In [29]:
```
1 sns.countplot(data=df, x='Courier Status', hue='Status')
```

Out[29]: <Axes: xlabel='Courier Status', ylabel='count'>



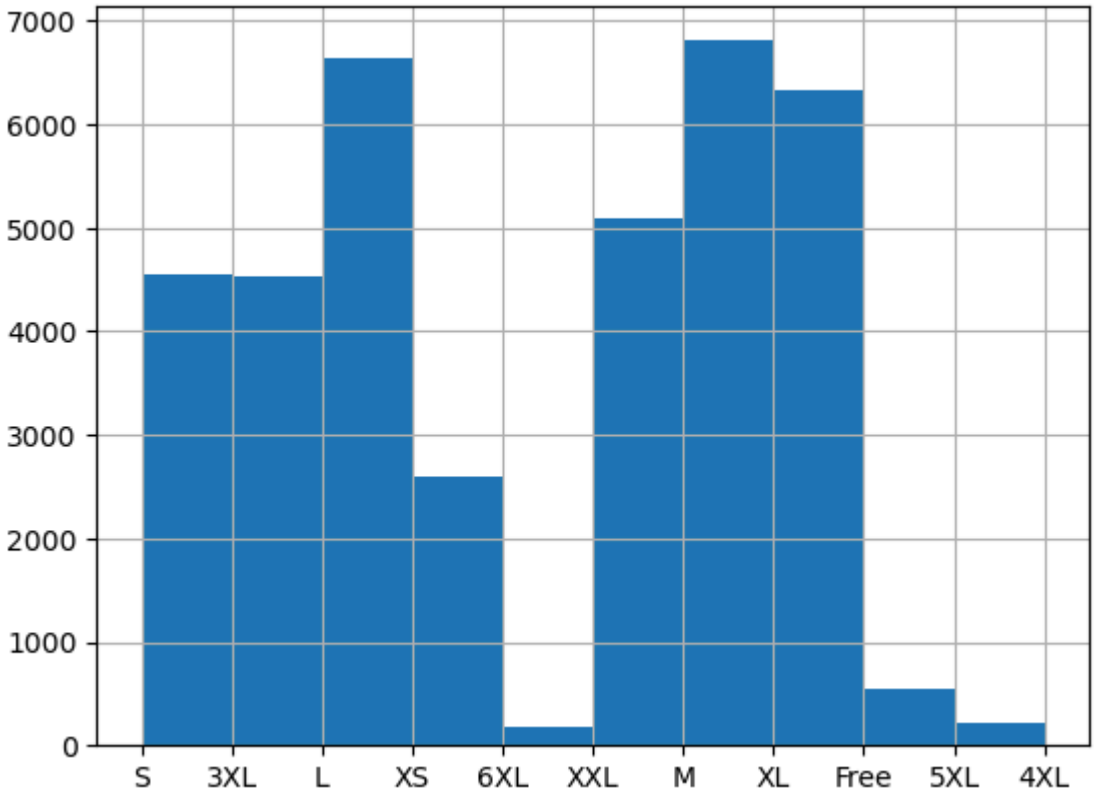
In [30]:
```
plt.figure(figsize=(10,5))

ax=sns.countplot(data=df, x='Courier Status',hue= 'Status')

plt.show()
```



Note: From above Graph the majority of the orders are shipped through the courier

In [31]:
```
#histogram
df['Size'].hist()
```

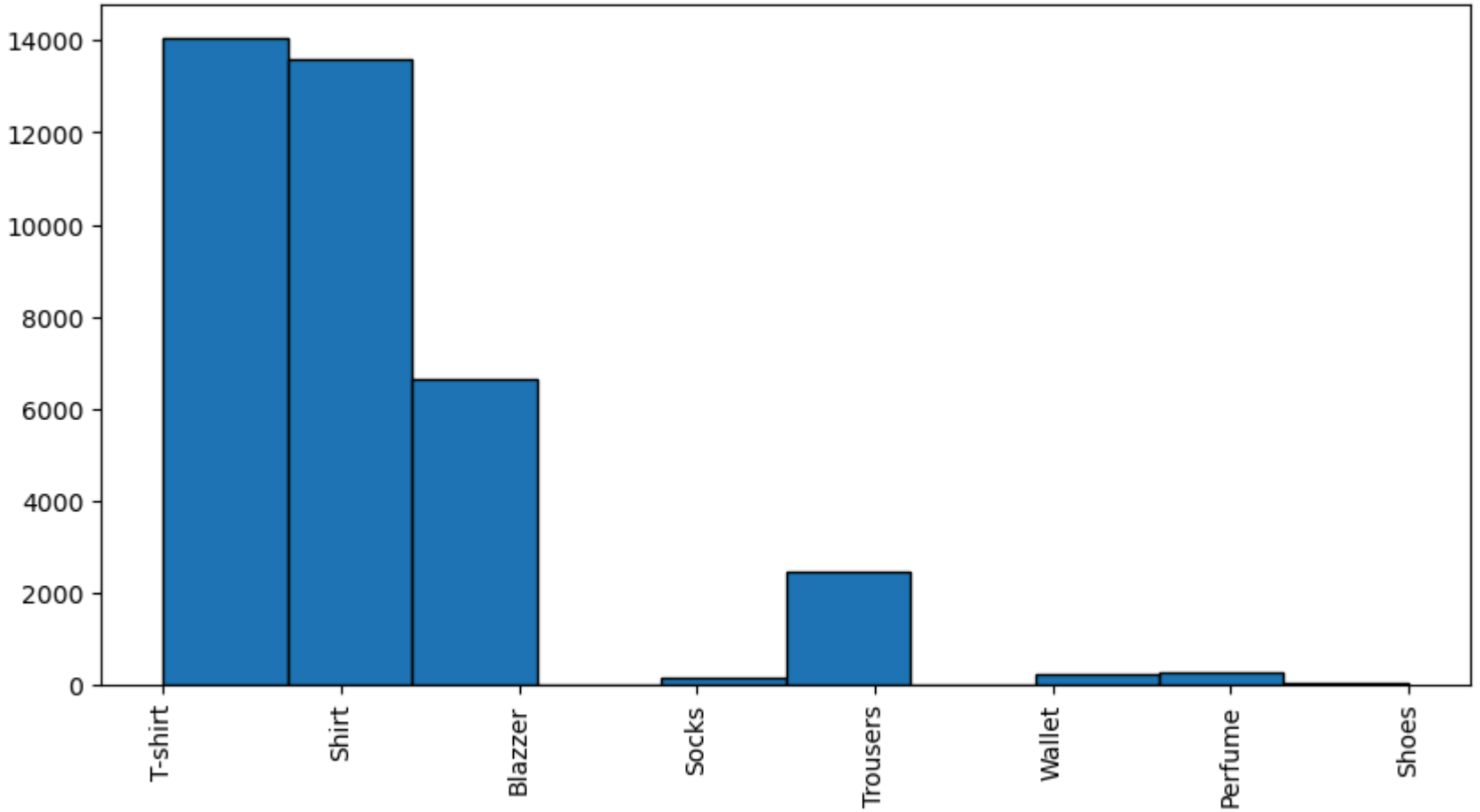
Out[31]: <Axes: >



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

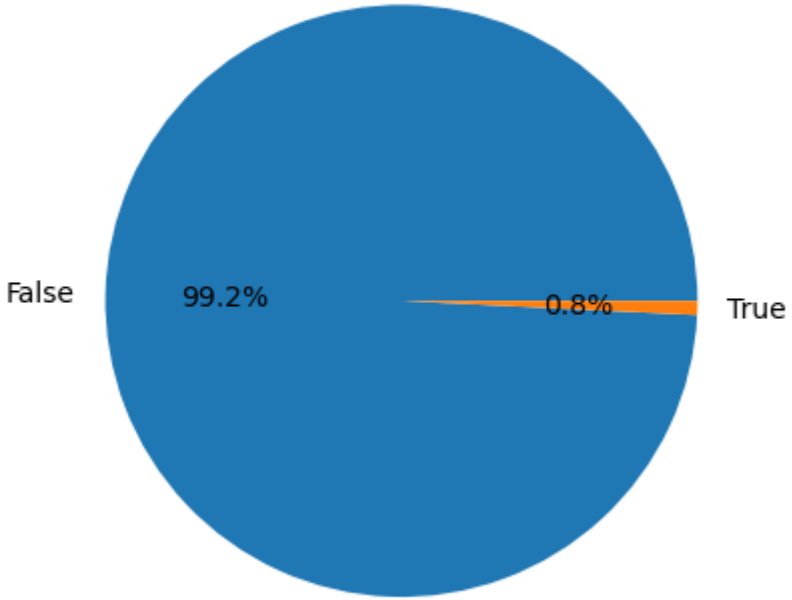
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 37514 entries, 0 to 128892
Data columns (total 19 columns):
 #   Column              Non-Null Count  Dtype         
---  ------              --------------  -----         
 0   index               37514 non-null  int64         
 1   Order ID            37514 non-null  object        
 2   Date                37514 non-null  datetime64[ns]
 3   Status              37514 non-null  object        
 4   Fulfilment          37514 non-null  object        
 5   Sales Channel       37514 non-null  object        
 6   ship-service-level  37514 non-null  object        
 7   Category            37514 non-null  object        
 8   Size                37514 non-null  object        
 9   Courier Status      37514 non-null  object        
 10  Qty                 37514 non-null  int64         
 11  currency            37514 non-null  object        
 12  Amount              37514 non-null  float64       
 13  ship-city           37514 non-null  object        
 14  ship-state          37514 non-null  object        
 15  ship-postal-code    37514 non-null  int32         
 16  ship-country        37514 non-null  object        
 17  B2B                 37514 non-null  bool          
 18  fulfilled-by        37514 non-null  object        
dtypes: bool(1), datetime64[ns](1), float64(1), int32(1), int64(2), object(13)
memory usage: 5.3+ MB
```

In [33]:
```
df['Category'] = df['Category'].astype(str)
column_data = df['Category']
plt.figure(figsize=(10,5))
plt.hist(column_data, bins=10, edgecolor='Black')
plt.xticks(rotation=90)
plt.show()
```



Note: From above Graph you can see that most of the buyers are T-shirt

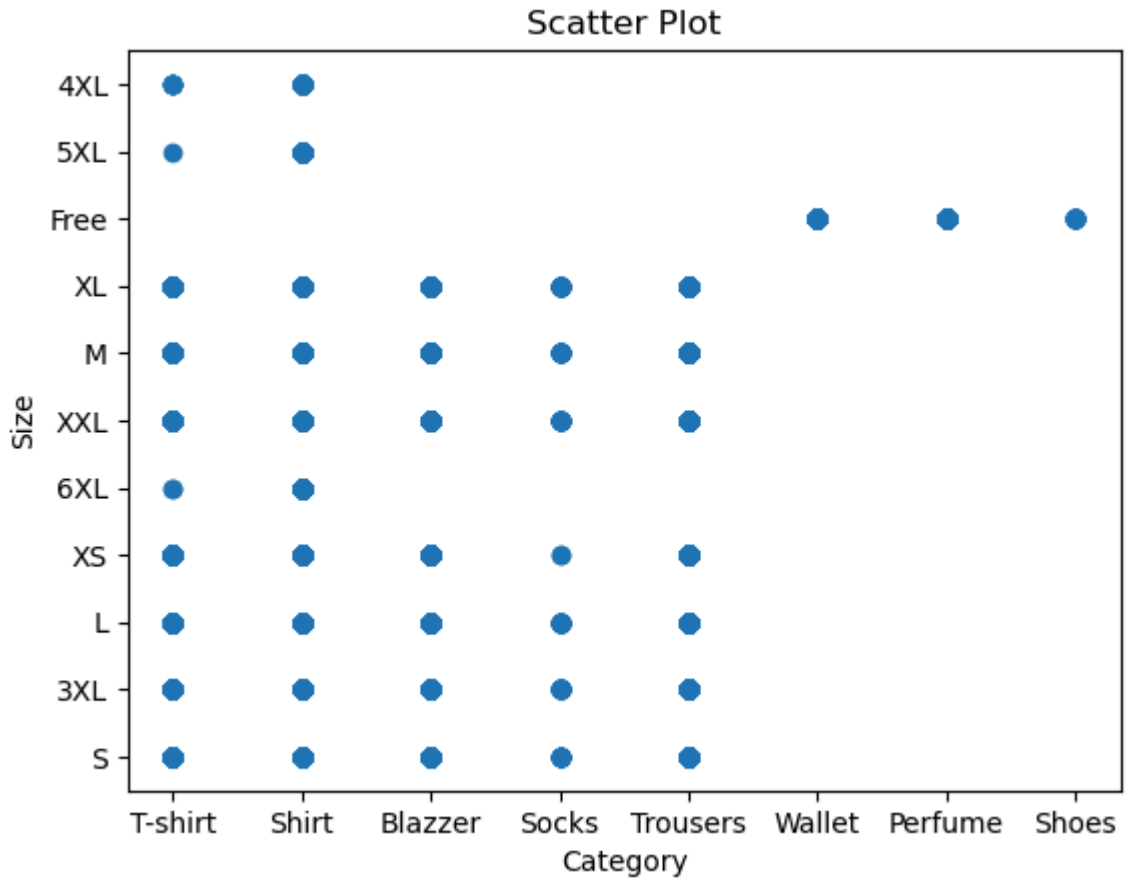
In [34]:
```
#Checking B2B Data by using pie chart
B2B_Check = df['B2B'].value_counts()

# Plot the Pie chart
plt.pie(B2B_Check, labels=B2B_Check.index ,autopct='%1.1f%%')
#plt.axis('equals')
plt.show()
```



Note: From above chart we can see that maximum i.e. 99.3% of buyers are retailers and 0.8% are B2B buyers

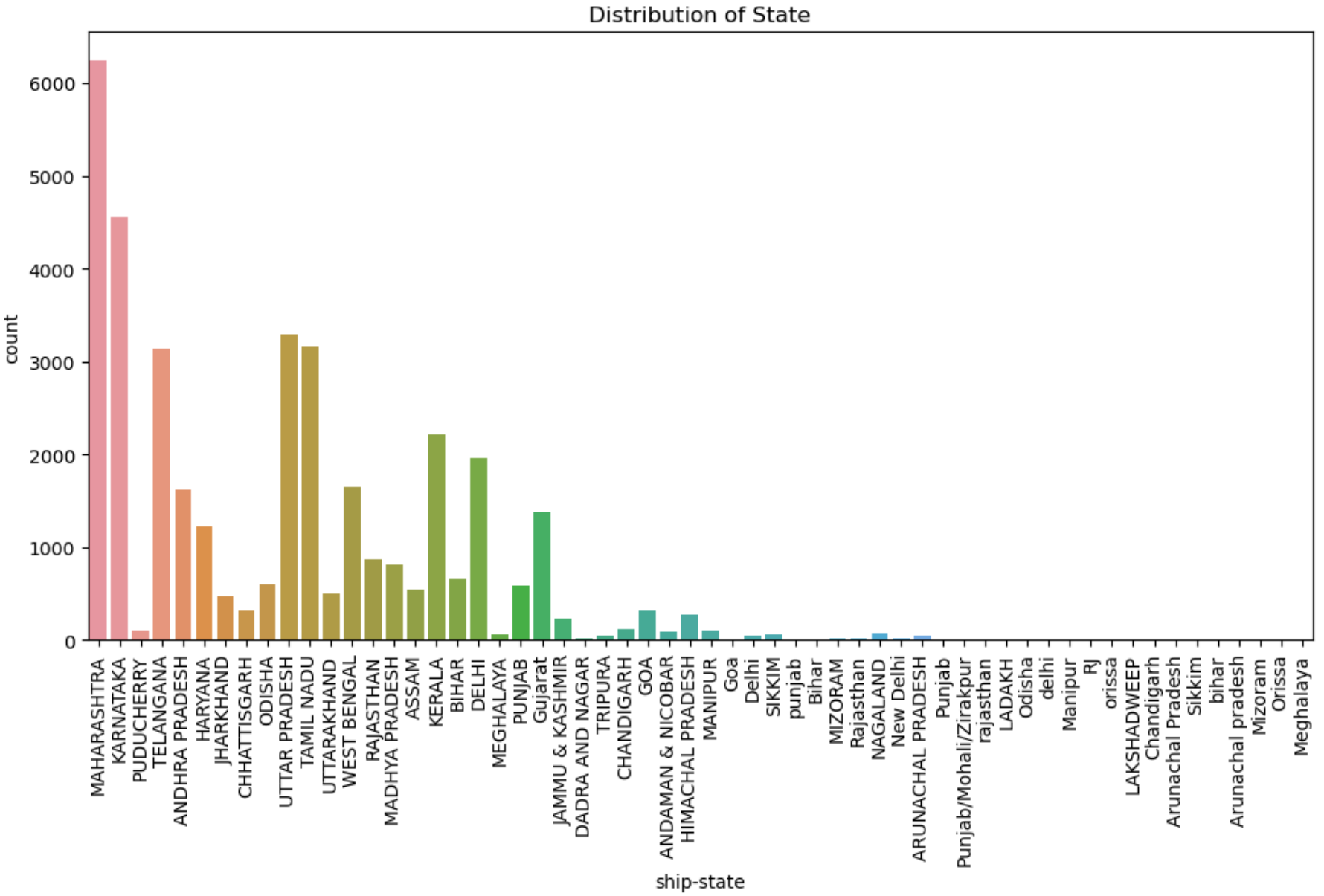
In [35]:

```
# Prepare data for scatter plot
x_data = df['Category']
y_data = df['Size']

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



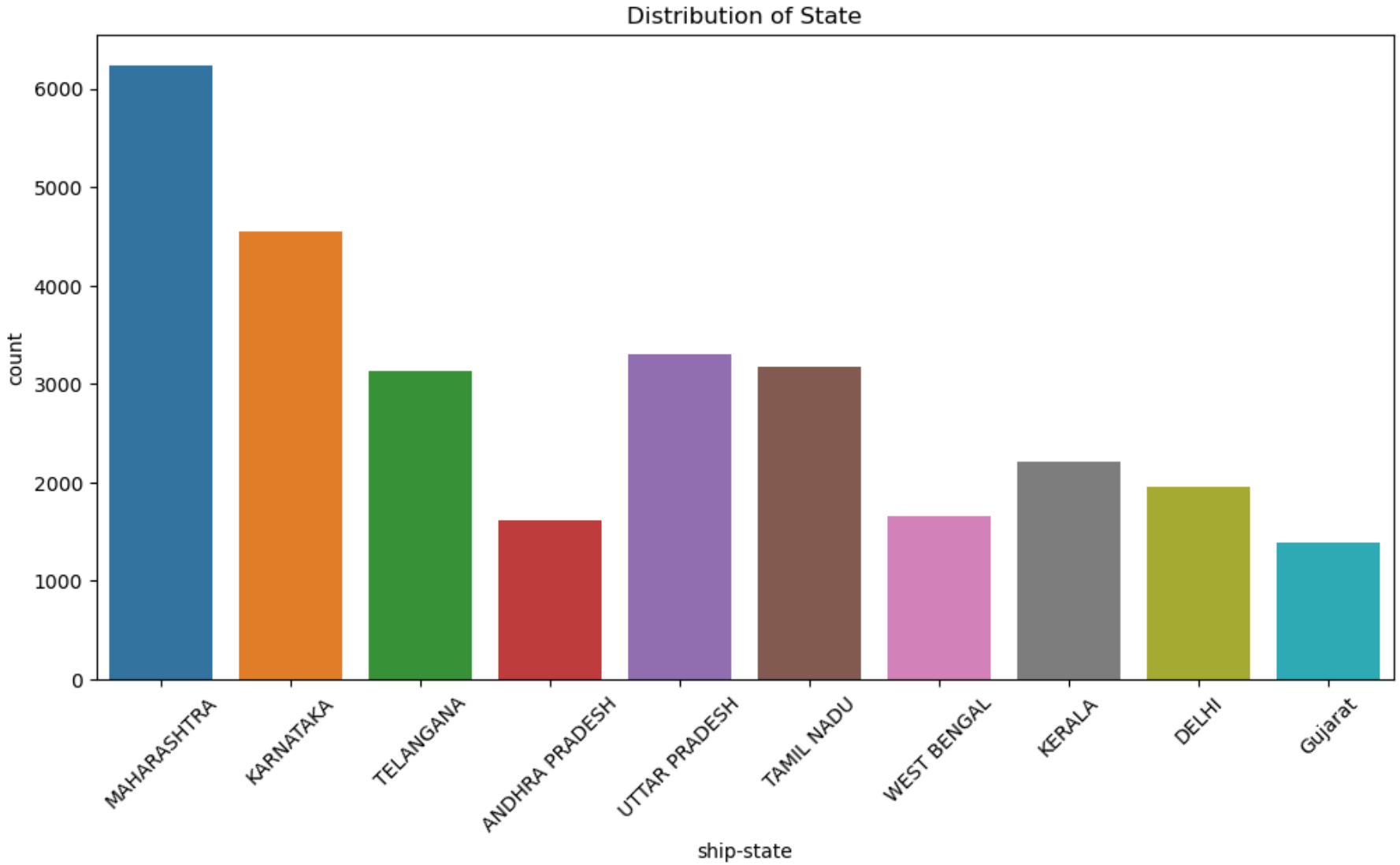
In [36]:

```
#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()
```


Distribution of State
count
0
1000
2000
3000
4000
5000
6000
MAHARASHTRA
KARNATAKA
PUDUCHERRY
TELANGANA
ANDHRA PRADESH
HARYANA
JHARKHAND
CHHATTISGARH
ODISHA
UTTAR PRADESH
TAMIL NADU
UTTARAKHAND
WEST BENGAL
RAJASTHAN
MADHYA PRADESH
ASSAM
KERALA
BIHAR
DELHI
MEGHALAYA
PUNJAB
Gujarat
JAMMU & KASHMIR
DADRA AND NAGAR
TRIPURA
CHANDIGARH
GOA
ANDAMAN & NICOBAR
HIMACHAL PRADESH
MANIPUR
Goa
Delhi
SIKKIM
punjab
Bihar
MIZORAM
Rajasthan
NAGALAND
New Delhi
ARUNACHAL PRADESH
Punjab
Punjab/Mohali/Zirakpur
rajasthan
LADAKH
Odisha
delhi
Manipur
RJ
orissa
LAKSHADWEEP
Chandigarh
Arunachal Pradesh
Sikkim
bihar
Arunachal pradesh
Mizoram
Orissa
Meghalaya
ship-state

In [37]:

```
#top_10_States
top_10_state = 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_state.index)],x='ship-state')
plt.xlabel('ship-state')
plt.ylabel('count')
plt.title('Distribution of State')
plt.xticks(rotation=45)
plt.show()
```



Note: From above you can see that most of the buyers are Maharashtra state

Conclusion:-

The Data Analysis reveals that the business has a significant customer base in Maharashtra state,mainly serves retailers, fulfills orders through Amazon, experience high demand for T-shirts, and sees M-Size as the preferred choice among buyers.

Thank you