In [14]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as pt
%matplotlib inline
import seaborn as sns
```

```
Matplotlib is building the font cache; this may take a moment.
```

In [18]:
```
df = pd.read_csv(r'D:\Data Analysis\Python\Amazon Sales Report Project\Dataset\Amaz
```

In [19]:
```
df.shape
```

Out[19]:
```
(128976, 21)
```

In [21]:
```
df.head(10)
```

Out[21]:

	index	Order ID	Date	Status	Fulfilment	Sales Channel	ship-service-level	Category	Size	Co s
0	0	405-8078784-5731545	04-30-22	Cancelled	Merchant	Amazon.in	Standard	T-shirt	S	C
1	1	171-9198151-1101146	04-30-22	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	Shirt	3XL	Sh
2	2	404-0687676-7273146	04-30-22	Shipped	Amazon	Amazon.in	Expedited	Shirt	XL	Sh
3	3	403-9615377-8133951	04-30-22	Cancelled	Merchant	Amazon.in	Standard	Blazzer	L	C
4	4	407-1069790-7240320	04-30-22	Shipped	Amazon	Amazon.in	Expedited	Trousers	3XL	Sh
5	5	404-1490984-4578765	04-30-22	Shipped	Amazon	Amazon.in	Expedited	T-shirt	XL	Sh
6	6	408-5748499-6859555	04-30-22	Shipped	Amazon	Amazon.in	Expedited	T-shirt	L	Sh
7	7	406-7807733-3785945	04-30-22	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	Shirt	S	Sh
8	8	407-5443024-5233168	04-30-22	Cancelled	Amazon	Amazon.in	Expedited	T-shirt	3XL	Can
9	9	402-4393761-0311520	04-30-22	Shipped	Amazon	Amazon.in	Expedited	Shirt	XXL	Sh

10 rows × 21 columns

In [23]:
```
df.tail(10)
```

Out[23]:

	index	Order ID	Date	Status	Fulfilment	Sales Channel	ship-service-level	Category	Si
128966	128965	408-5154281-4593912	05-31-22	Cancelled	Amazon	Amazon.in	Expedited	Trousers	3X
128967	128966	406-9812666-2474761	05-31-22	Shipped	Amazon	Amazon.in	Expedited	T-shirt	X
128968	128967	404-5182288-1653947	05-31-22	Cancelled	Amazon	Amazon.in	Expedited	Shirt	X
128969	128968	403-7059995-7618722	05-31-22	Shipped	Amazon	Amazon.in	Expedited	T-shirt	X
128970	128969	404-3802633-7250760	05-31-22	Cancelled	Amazon	Amazon.in	Expedited	T-shirt	
128971	128970	406-6001380-7673107	05-31-22	Shipped	Amazon	Amazon.in	Expedited	Shirt	X
128972	128971	402-9551604-7544318	05-31-22	Shipped	Amazon	Amazon.in	Expedited	T-shirt	
128973	128972	407-9547469-3152358	05-31-22	Shipped	Amazon	Amazon.in	Expedited	Blazzer	XX
128974	128973	402-6184140-0545956	05-31-22	Shipped	Amazon	Amazon.in	Expedited	T-shirt	X
128975	128974	408-7436540-8728312	05-31-22	Shipped	Amazon	Amazon.in	Expedited	T-shirt	

10 rows × 21 columns

In [24]:
```
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
```

Data Cleaning

Drop Null Columns

In [29]:
```
df.drop(['New', 'PendingS'], axis = 1, inplace = True)
```

In [30]:
```
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 [31]:
```
null_rows = df[df.isnull().any(axis=1)]
print(null_rows)
```

```
         index             Order ID      Date     Status Fulfilment  \
2            2  404-0687676-7273146  04-30-22    Shipped     Amazon
4            4  407-1069790-7240320  04-30-22    Shipped     Amazon
5            5  404-1490984-4578765  04-30-22    Shipped     Amazon
6            6  408-5748499-6859555  04-30-22    Shipped     Amazon
8            8  407-5443024-5233168  04-30-22  Cancelled     Amazon
...        ...                  ...       ...        ...        ...
128971  128970  406-6001380-7673107  05-31-22    Shipped     Amazon
128972  128971  402-9551604-7544318  05-31-22    Shipped     Amazon
128973  128972  407-9547469-3152358  05-31-22    Shipped     Amazon
128974  128973  402-6184140-0545956  05-31-22    Shipped     Amazon
128975  128974  408-7436540-8728312  05-31-22    Shipped     Amazon

       Sales Channel ship-service-level  Category Size Courier Status  Qty  \
2          Amazon.in          Expedited     Shirt   XL        Shipped    1
4          Amazon.in          Expedited  Trousers  3XL        Shipped    1
5          Amazon.in          Expedited   T-shirt   XL        Shipped    1
6          Amazon.in          Expedited   T-shirt    L        Shipped    1
8          Amazon.in          Expedited   T-shirt  3XL      Cancelled    0
...              ...                ...       ...  ...            ...  ...
128971     Amazon.in          Expedited     Shirt   XL        Shipped    1
128972     Amazon.in          Expedited   T-shirt    M        Shipped    1
128973     Amazon.in          Expedited   Blazzer  XXL        Shipped    1
128974     Amazon.in          Expedited   T-shirt   XS        Shipped    1
128975     Amazon.in          Expedited   T-shirt    S        Shipped    1

       currency  Amount    ship-city     ship-state  ship-postal-code  \
2           INR   329.0  NAVI MUMBAI    MAHARASHTRA          410210.0
4           INR   574.0      CHENNAI     TAMIL NADU          600073.0
5           INR   824.0    GHAZIABAD  UTTAR PRADESH          201102.0
6           INR   653.0   CHANDIGARH     CHANDIGARH          160036.0
8           NaN     NaN    HYDERABAD      TELANGANA          500008.0
...         ...     ...          ...            ...               ...
128971      INR   517.0    HYDERABAD      TELANGANA          500013.0
128972      INR   999.0     GURUGRAM        HARYANA          122004.0
128973      INR   690.0    HYDERABAD      TELANGANA          500049.0
128974      INR  1199.0        Halol        Gujarat          389350.0
128975      INR   696.0       Raipur   CHHATTISGARH          492014.0

       ship-country    B2B fulfilled-by
2                IN   True          NaN
4                IN  False          NaN
5                IN  False          NaN
6                IN  False          NaN
8                IN  False          NaN
...             ...    ...          ...
128971           IN  False          NaN
128972           IN  False          NaN
128973           IN  False          NaN
128974           IN  False          NaN
128975           IN  False          NaN

[91462 rows x 19 columns]
```

To Check Null Values

In [32]:
```
pd.isnull(df)
```

Out[32]:

	index	Order ID	Date	Status	Fulfilment	Sales Channel	ship-service-level	Category	Size	Courie Statu
0	False	False	False	False	False	False	False	False	False	Fals
1	False	False	False	False	False	False	False	False	False	Fals
2	False	False	False	False	False	False	False	False	False	Fals
3	False	False	False	False	False	False	False	False	False	Fals
4	False	False	False	False	False	False	False	False	False	Fals
...	...	...	...	...	...	...	...	...	...	
128971	False	False	False	False	False	False	False	False	False	Fals
128972	False	False	False	False	False	False	False	False	False	Fals
128973	False	False	False	False	False	False	False	False	False	Fals
128974	False	False	False	False	False	False	False	False	False	Fals
128975	False	False	False	False	False	False	False	False	False	Fals

128976 rows × 19 columns

Total Num Values

In [33]:
```
pd.isnull(df).sum()
```

Out[33]:
```
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 [34]:
```
df.shape
```

Out[34]:
```
(128976, 19)
```

Drop Null Values

In [35]:
```
df.dropna(inplace = True)
```

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

Out[36]:
```
(37514, 19)
```

In [37]:
```
df.columns
```

Out[37]:
```
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')
```

Change Data Types

In [38]:
```
df['ship-postal-code']=df['ship-postal-code']. astype('int')
```

In [39]:
```
df['ship-postal-code'].dtype
```

Out[39]:
```
dtype('int64')
```

In [43]:
```
df['Date']= pd.to_datetime(df['Date'])
```

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

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

In [45]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 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  int64         
 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), int64(3), object(13)
memory usage: 5.5+ MB
```

In [46]:
```
df.columns
```

Out[46]:
```
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')
```

Change Column Name

In [47]:
```
df.rename(columns={'Qty':'Quantity'})
```

Out[47]:

	index	Order ID	Date	Status	Fulfilment	Sales Channel	ship-service-level	Category	Siz
0	0	405-8078784-5731545	2022-04-30	Cancelled	Merchant	Amazon.in	Standard	T-shirt	
1	1	171-9198151-1101146	2022-04-30	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	Shirt	3X
3	3	403-9615377-8133951	2022-04-30	Cancelled	Merchant	Amazon.in	Standard	Blazzer	
7	7	406-7807733-3785945	2022-04-30	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	Shirt	
12	12	405-5513694-8146768	2022-04-30	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	Shirt	X
...	...	...	...	...	...	...	...	...	
128875	128874	405-4724097-1016369	2022-06-01	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	T-shirt	
128876	128875	403-9524128-9243508	2022-06-01	Cancelled	Merchant	Amazon.in	Standard	Blazzer	X
128888	128887	405-6493630-8542756	2022-05-31	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	Trousers	L
128891	128890	407-0116398-1810752	2022-05-31	Cancelled	Merchant	Amazon.in	Standard	Wallet	Fre
128892	128891	403-0317423-9322704	2022-05-31	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	Blazzer	L

37514 rows × 19 columns

In [48]:
```
df.describe()
```

Out[48]:

	index	Date	Qty	Amount	ship-postal-code
count	37514.000000	37514	37514.000000	37514.000000	37514.000000
mean	60953.809858	2022-05-11 07:56:47.303939840	0.867383	646.553960	463291.552754
min	0.000000	2022-03-31 00:00:00	0.000000	0.000000	110001.000000
25%	27235.250000	2022-04-20 00:00:00	1.000000	458.000000	370465.000000
50%	63470.500000	2022-05-09 00:00:00	1.000000	629.000000	500019.000000
75%	91790.750000	2022-06-01 00:00:00	1.000000	771.000000	600042.000000
max	128891.000000	2022-06-29 00:00:00	5.000000	5495.000000	989898.000000
std	36844.853039	NaN	0.354160	279.952414	194550.425637

In [54]:
```
df.describe(include = 'object')
```

Out[54]:

	Order ID	Status	Fulfilment	Sales Channel	ship-service-level	Category	Size	Courier Status	cu
count	37514	37514	37514	37514	37514	37514	37514	37514	
unique	34664	11	1	1	1	8	11	3	
top	171-5057375-2831560	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	T-shirt	M	Shipped	
freq	12	28741	37514	37514	37514	14062	6806	31859	

In [64]:
```
df[['Qty','Amount']].describe()
```

Out[64]:

	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

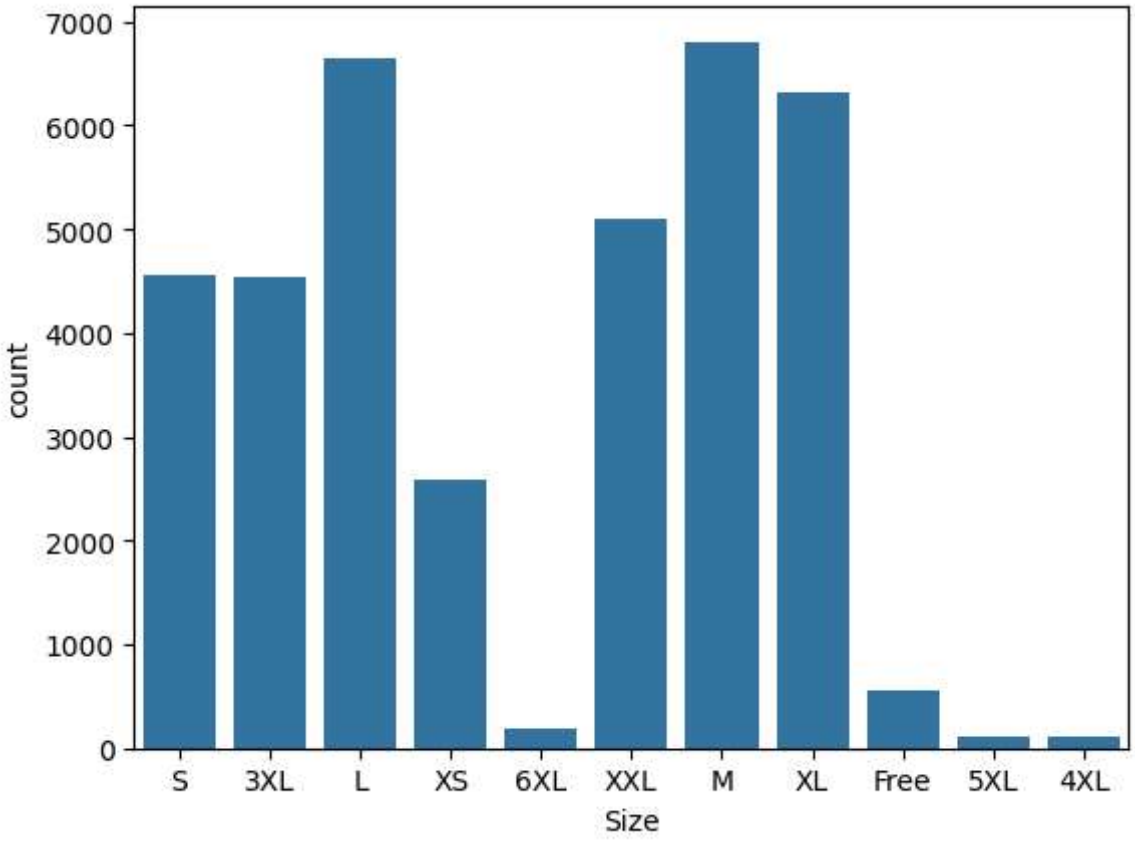
Data Analysis and Visualization

In [65]:
```
df.columns
```

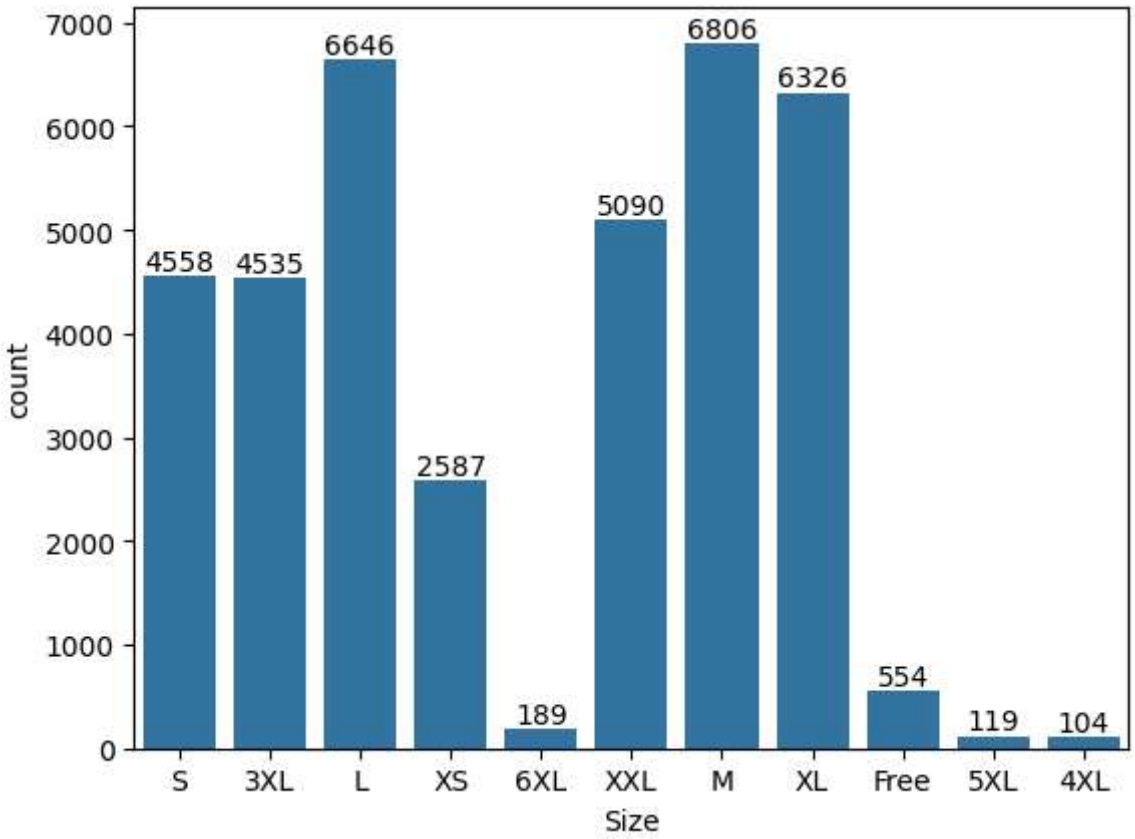
Out[65]:
```
Index(['index', 'Order ID', 'Date', 'Status', 'Fulfilment', 'Sales Channel',
       'ship-service-level', 'Category', 'Size', 'Courier Status', 'Qty',
       'currency', 'Amount', 'ship-city', 'ship-state', 'ship-postal-code',
       'ship-country', 'B2B', 'fulfilled-by'],
      dtype='object')
```

Customer Count for Sizes

In [66]:
```
ax = sns.countplot(x='Size', data = df)
```



In [71]:
```
ax = sns.countplot(x = 'Size', data = df)
for bars in ax.containers:
    ax.bar_label(bars)
```



Note:

Under to the above graph, Most of the people buy M size and least number of people buy 4XL size

Group By and Sort Values

In [85]:
```
df.groupby(['Size'], as_index = False)['Qty'].sum().sort_values(by = 'Qty', ascendi
```

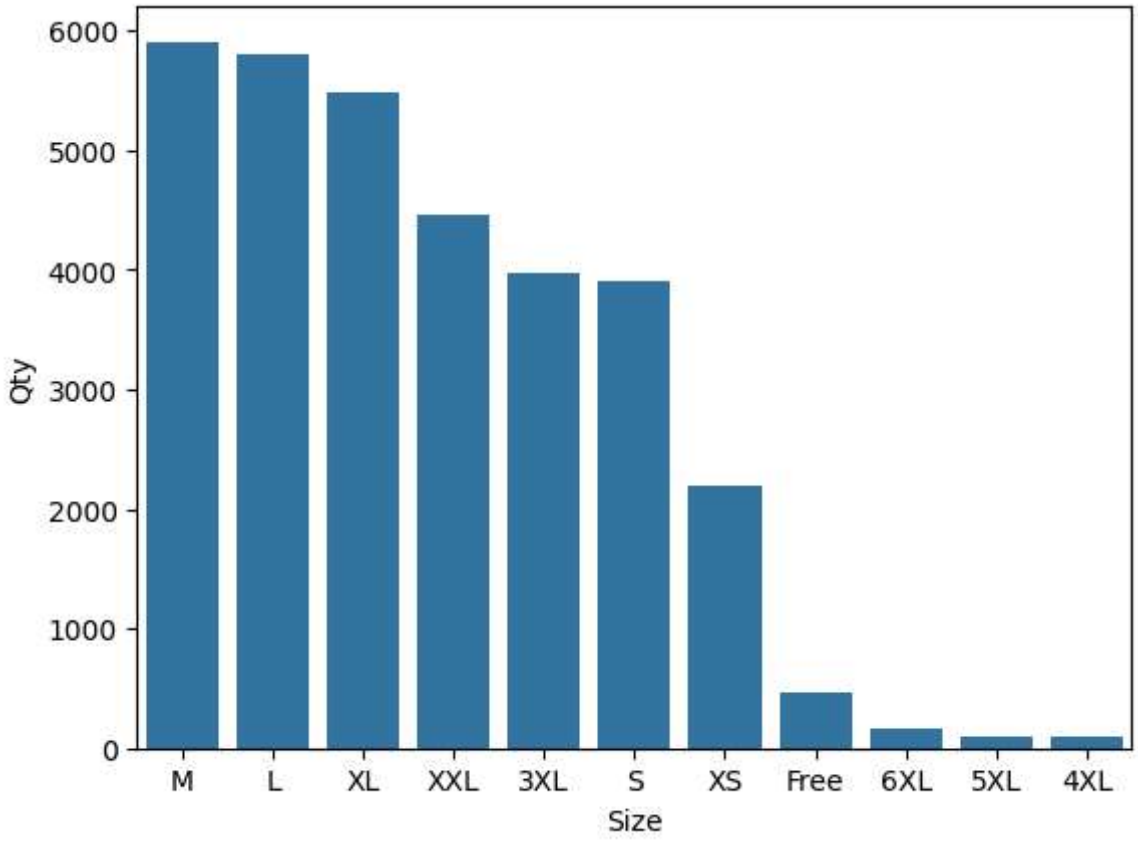
Out[85]:

	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

Quantity of Sizes

In [90]:
```
Size_Qty = df.groupby(['Size'],as_index = False)['Qty'].sum().sort_values(by='Qty',
sns.barplot(x = 'Size', y = 'Qty', data = Size_Qty)
```

Out[90]:
```
<Axes: xlabel='Size', ylabel='Qty'>
```



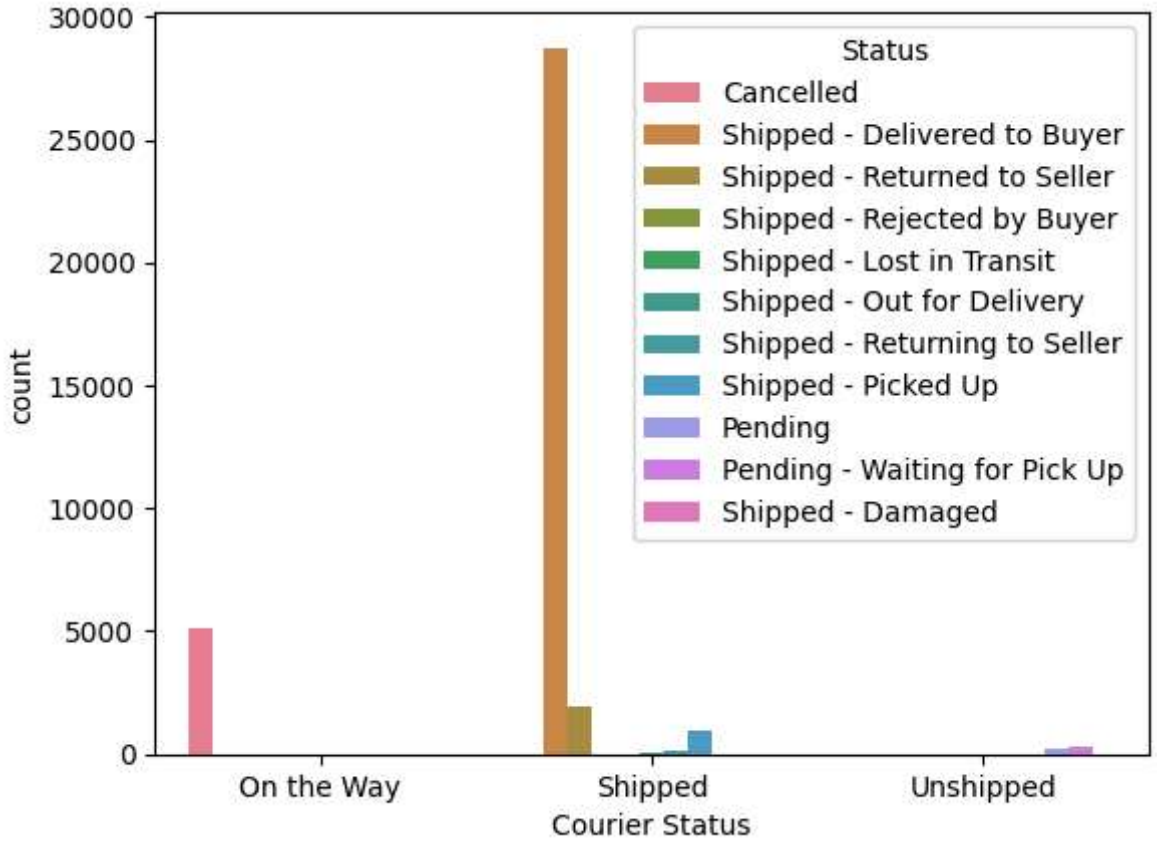
Note:

Under to the above graph, Most of the Qty buy M size and least Qty buy 4XL size

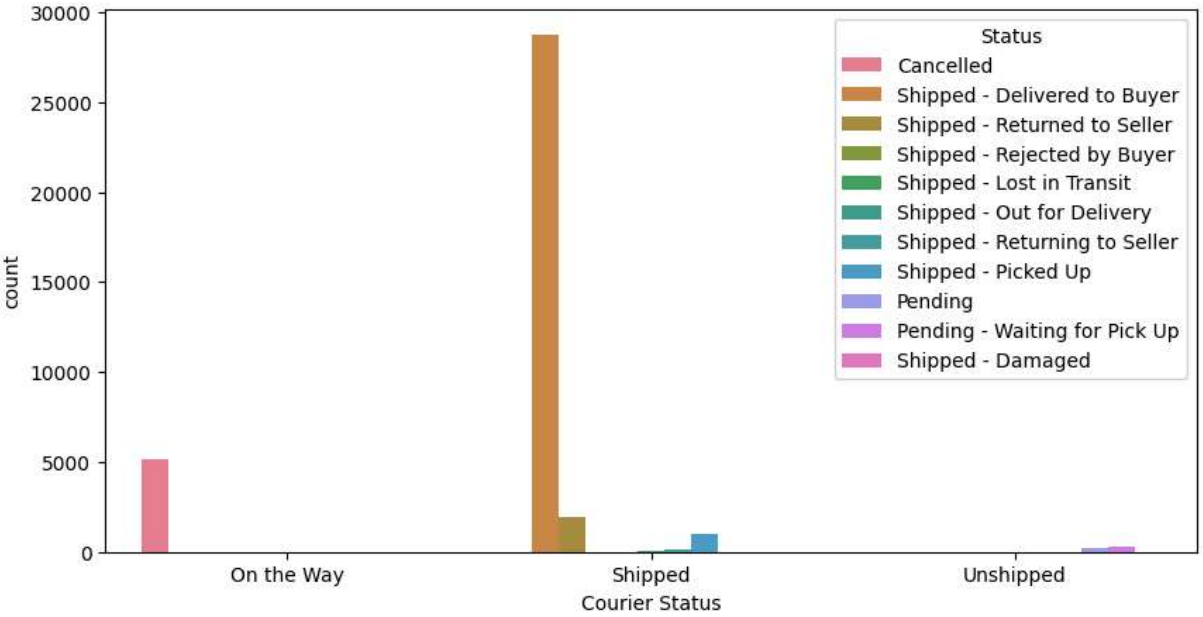
Courier Status

In [93]:
```
sns.countplot(x = 'Courier Status', hue = 'Status', data = df)
```

Out[93]:
```
<Axes: xlabel='Courier Status', ylabel='count'>
```



In [98]:
```
pt.figure(figsize=(10,5))
ax = sns.countplot(x = 'Courier Status', hue = 'Status', data = df)
pt.show()
```

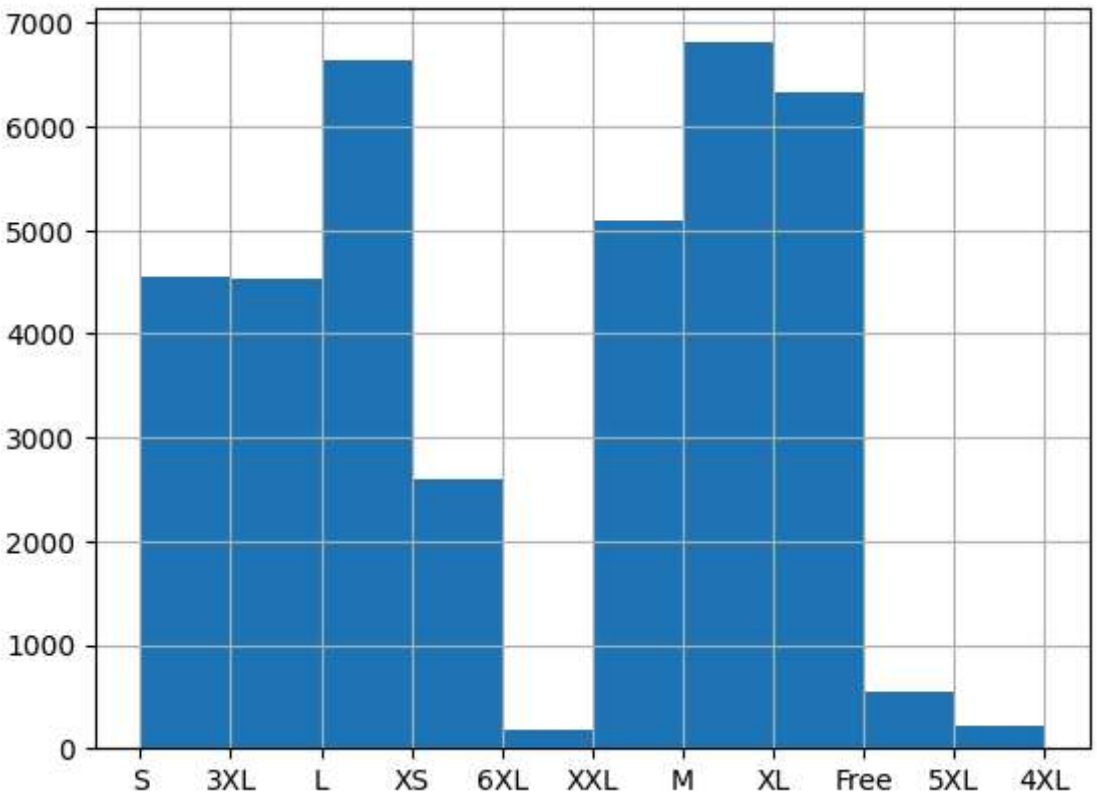


Note:

Under to the above graph, Majority of the orders are shipped through the courier.

In [99]:
```
df['Size'].hist()
```

Out[99]: <Axes: >



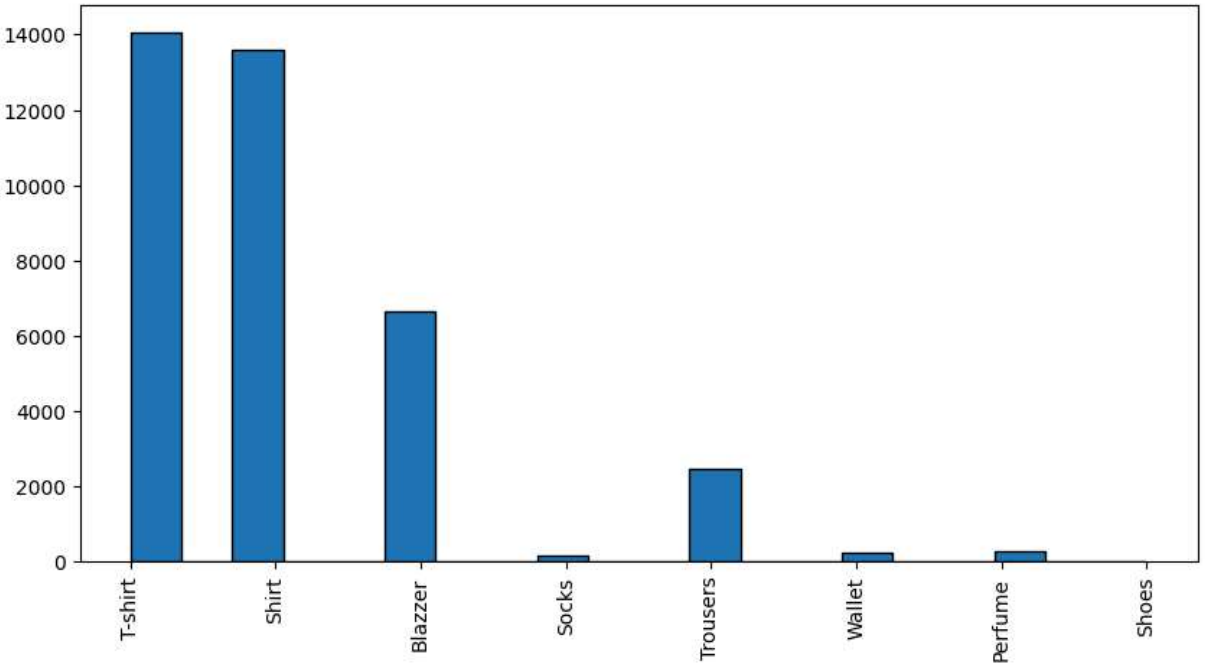
In [100…
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 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  int64
 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), int64(3), object(13)
memory usage: 5.5+ MB
```

Customer Categories

In [109…

```
df['Category'] = df['Category'].astype(str)
column_data = df['Category']
pt.figure(figsize=(10,5))
pt.hist(column_data, bins=20, edgecolor = 'Black')
pt.xticks(rotation = 90)
pt.show()
```



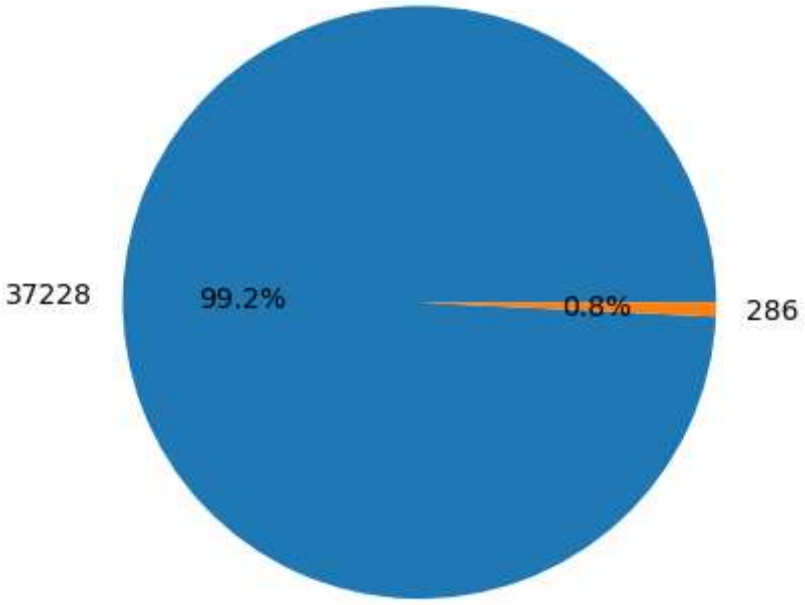
Note:

Under to the above graph, Most of the customers bought T-shirts.

B2B or Retailer

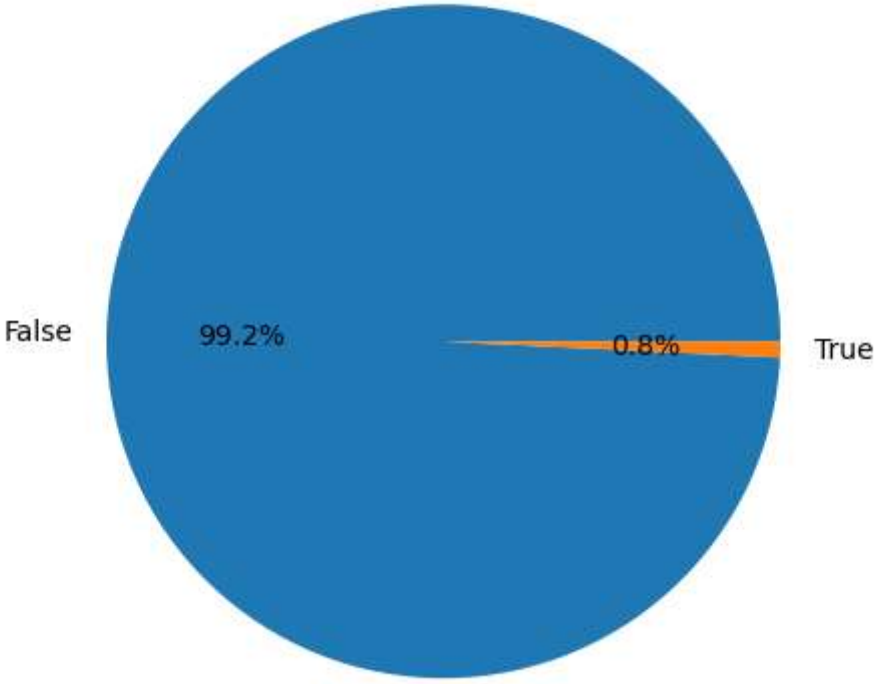
In [122…

```
##Prepare data for pie chart
B2B_check = df['B2B'].value_counts()

pt.pie(B2B_check, labels=B2B_check, autopct = '%1.1f%%')
pt.show()
```



In [125…

```
##Prepare data for pie chart
B2B_check = df['B2B'].value_counts()

pt.pie(B2B_check, labels = B2B_check.index, autopct ='%1.1f%%')
pt.axis('equal')
pt.show()
```



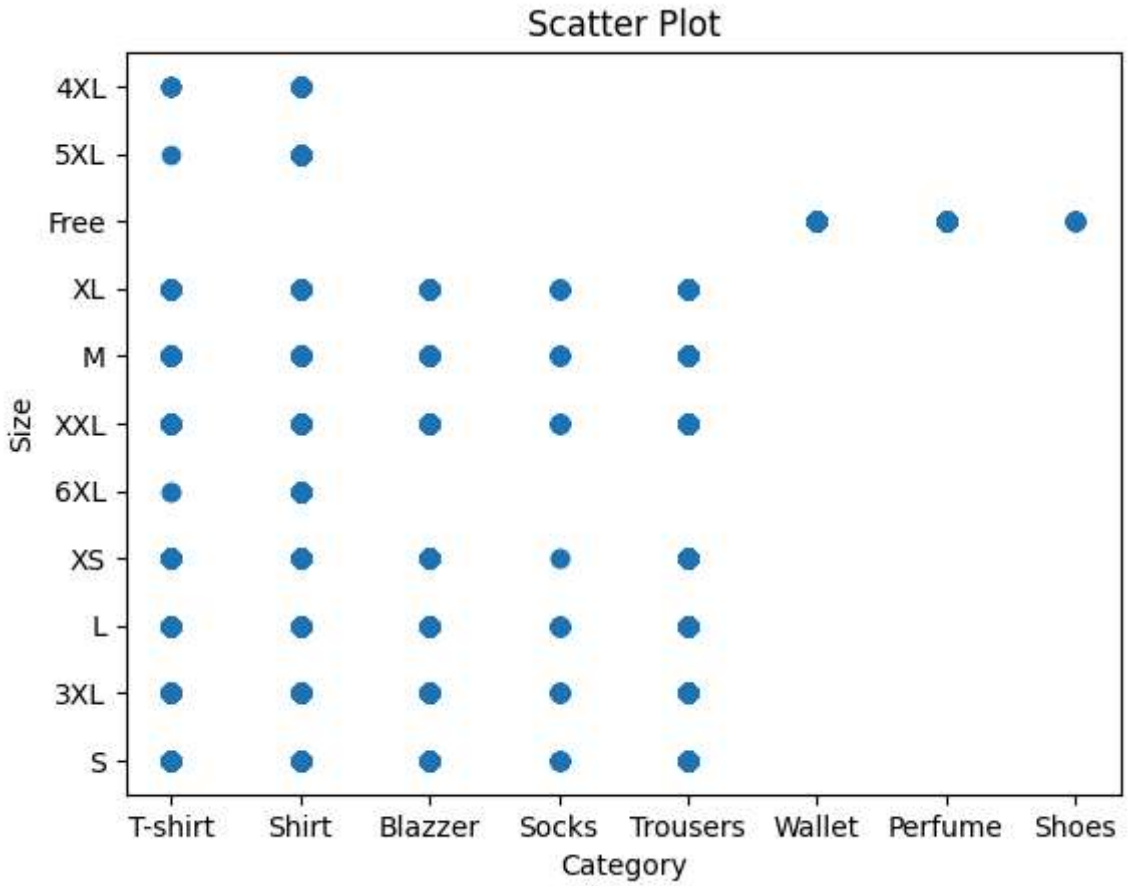
Note:

Under to the above graph, 99.2% buyers are retailers while 0.8% are B2B buyers.

Categories Over Sizes

In [131…

```
##Prepare data for Scatter Plot
x_label = df['Category']
y_label = df['Size']

pt.scatter(x_label,y_label)
pt.xlabel('Category')
pt.ylabel('Size')
pt.title('Scatter Plot')
pt.show()
```



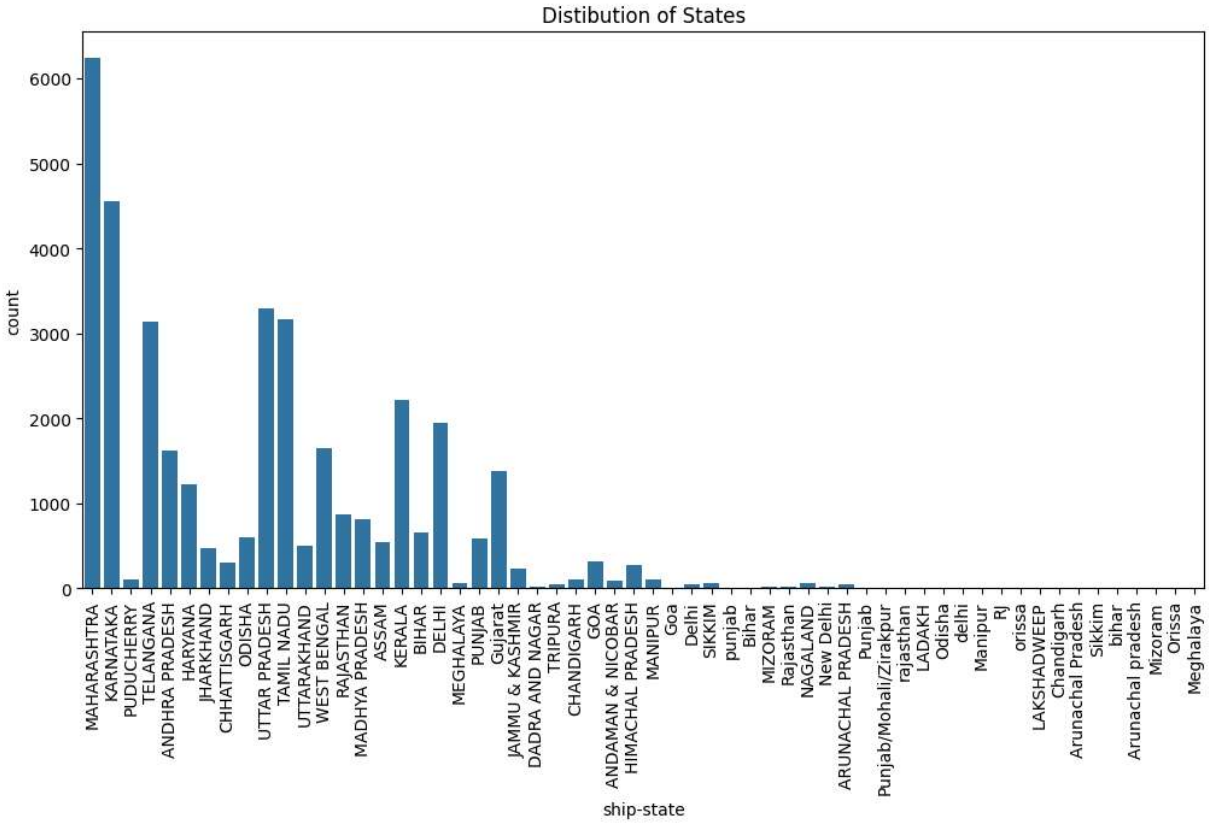
Note:

Under to the above graph, T-shirts available the highest sizes.

In [145…

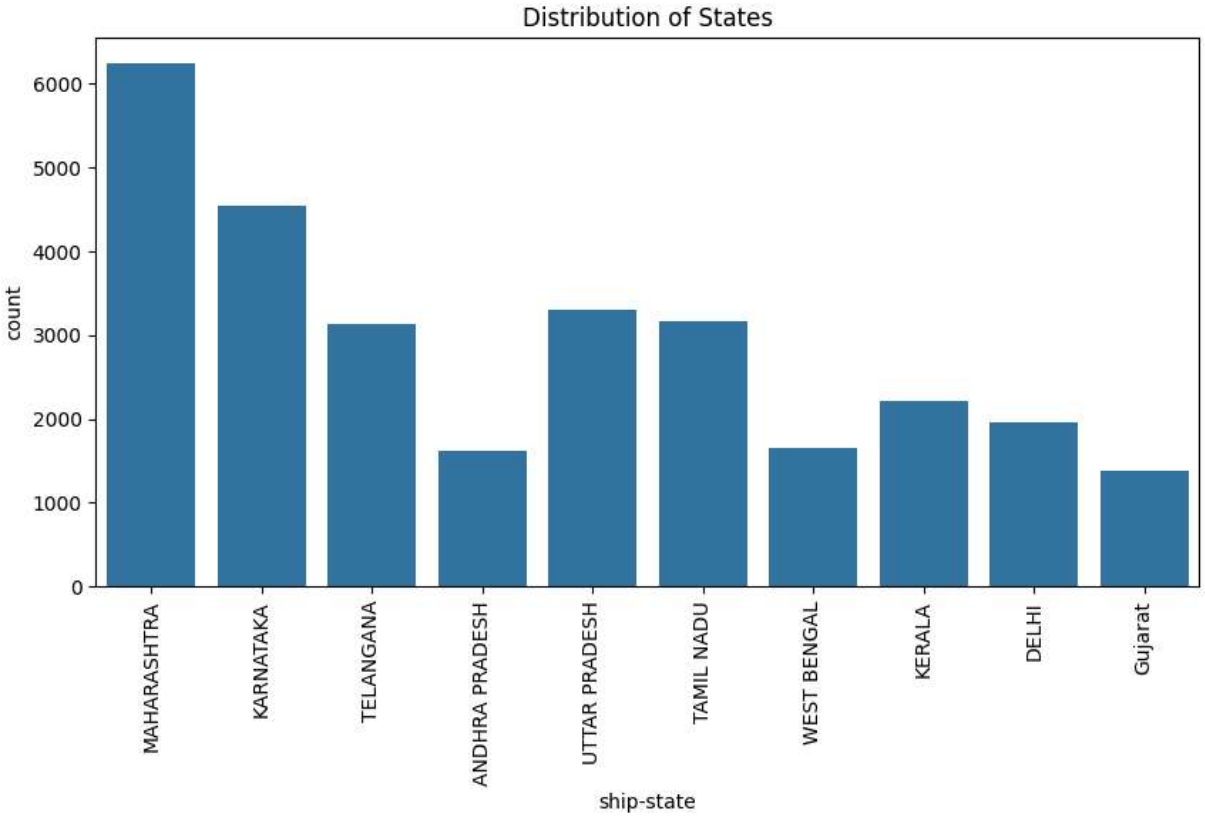
```
pt.figure(figsize=(12,6))
sns.countplot(x= 'ship-state', data=df)
pt.xlabel('ship-state')
pt.ylabel('count')
pt.title('Distibution of States')
```

```
pt.xticks(rotation = 90)
pt.show()
```



In [149…

```
##Top 10 states
top_10_states = df['ship-state'].value_counts().head(10)
pt.figure(figsize=(10,5))
sns.countplot(data=df[df['ship-state'].isin(top_10_states.index)],x = 'ship-state')
pt.xlabel('ship-state')
pt.ylabel('count')
pt.xticks(rotation = 90)
pt.title('Distribution of States')
pt.show()
```



Note:

Under to the above graph, Most of buyers are in Maharasthra

Conclusion:

When considering about these details we can see that the business has a significant customer base in Mahasthra state and mainly serves retailers through Amazon.Compairing to others the highest demand is for T-shirts with M sizes among buyers.

In [ ]: