In [1]:

```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

In [2]:

```python
df = pd.read_csv("C:/Users/sridh/Desktop/Supermart Grocery Sales - Retail Analytics Dataset.csv")
```

In [3]:

```python
# 3. Basic Info
print(df.shape)
print(df.columns)
df.info()
df.describe()
df.head()
```

```
(9994, 11)
Index(['Order ID', 'Customer Name', 'Category', 'Sub Category', 'City',
       'Order Date', 'Region', 'Sales', 'Discount', 'Profit', 'State'],
      dtype='object')
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9994 entries, 0 to 9993
Data columns (total 11 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   Order ID       9994 non-null   object 
 1   Customer Name  9994 non-null   object 
 2   Category       9994 non-null   object 
 3   Sub Category   9994 non-null   object 
 4   City           9994 non-null   object 
 5   Order Date     9994 non-null   object 
 6   Region         9994 non-null   object 
 7   Sales          9994 non-null   int64  
 8   Discount       9994 non-null   float64
 9   Profit         9994 non-null   float64
 10  State          9994 non-null   object 
dtypes: float64(2), int64(1), object(8)
memory usage: 859.0+ KB
```

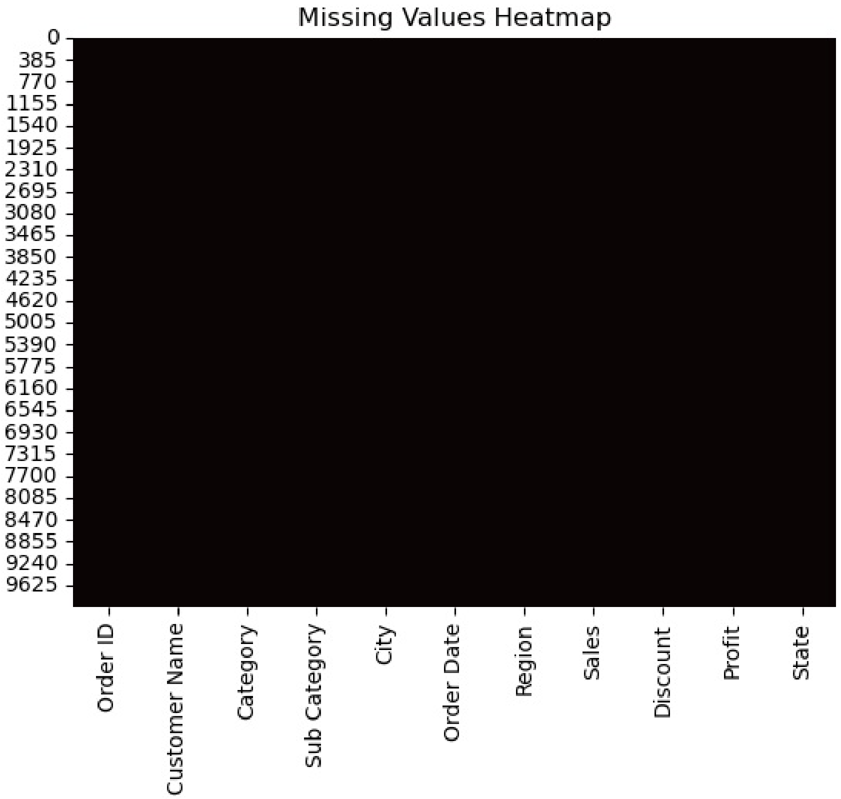
Out[3]:

	Order ID	Customer Name	Category	Sub Category	City	Order Date	Region	Sales	Discount	Profit	State
0	OD1	Harish	Oil & Masala	Masalas	Vellore	11-08-2017	North	1254	0.12	401.28	Tamil Nadu
1	OD2	Sudha	Beverages	Health Drinks	Krishnagiri	11-08-2017	South	749	0.18	149.80	Tamil Nadu
2	OD3	Hussain	Food Grains	Atta & Flour	Perambalur	06-12-2017	West	2360	0.21	165.20	Tamil Nadu
3	OD4	Jackson	Fruits & Veggies	Fresh Vegetables	Dharmapuri	10-11-2016	South	896	0.25	89.60	Tamil Nadu
4	OD5	Ridhesh	Food Grains	Organic Staples	Ooty	10-11-2016	South	2355	0.26	918.45	Tamil Nadu

In [5]:

```python
print(df.isnull().sum())
sns.heatmap(df.isnull(), cbar=False, cmap='mako')
plt.title("Missing Values Heatmap")
plt.show()
```

```
Order ID         0
Customer Name    0
Category         0
Sub Category     0
City             0
Order Date       0
Region           0
Sales            0
Discount         0
Profit           0
State            0
dtype: int64
```



In [6]:
```python
df = df.loc[:, ~df.columns.str.contains('^Unnamed')]
```

In [7]:
```python
print("Duplicates:", df.duplicated().sum())
df = df.drop_duplicates()
```

```
Duplicates: 0
```

In [8]:
```python
df = df.dropna()
```

In [9]:
```python
categorical_cols = df.select_dtypes(include='object').columns
for col in categorical_cols:
    print(f"\nValue counts for '{col}':")
    print(df[col].value_counts())
```

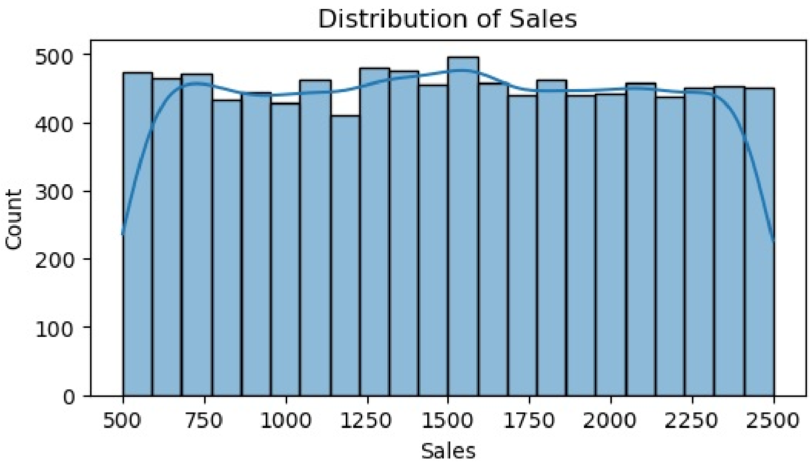
```
Value counts for 'Order ID':
Order ID
OD1       1
OD6666    1
OD6659    1
OD6660    1
OD6661    1
         ..
OD3333    1
OD3334    1
OD3335    1
OD3336    1
OD9994    1
Name: count, Length: 9994, dtype: int64

Value counts for 'Customer Name':
Customer Name
Amrish      227
Krithika    224
Verma       218
Arutra      218
Vidya       215
Shah        215
Suresh      212
Surya       209
Harish      208
```

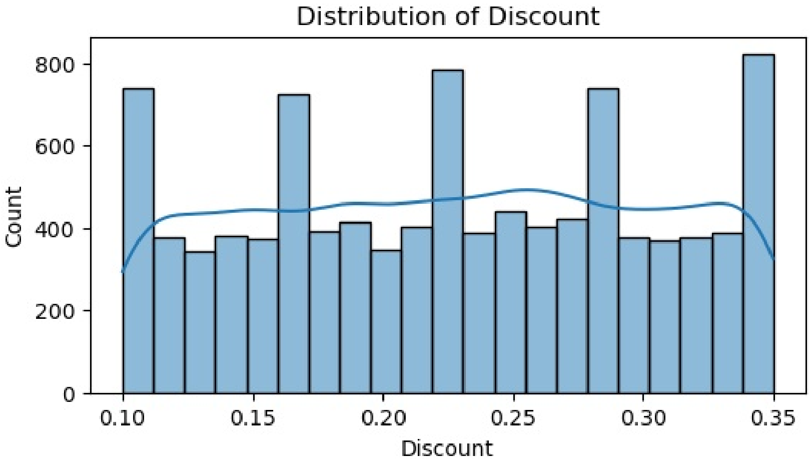
```
Hussain     208
Sudeep      207
Komal       206
Veena       205
Mathew      205
Adavan      205
Ridhesh     204
Muneer      204
Peer        204
Veronica    203
Arvind      203
Vinne       203
Sharon      202
Haseena     202
Malik       201
Yusuf       201
Roshan      201
Shree       200
Ravi        200
Jonas       198
Alan        198
James       197
Ram         197
Amy         196
Akash       196
Willams     195
Sheeba      195
Rumaiza     195
Ganesh      193
Esther      189
Sudha       189
Vince       188
Ramesh      188
Sabeela     188
Sundar      187
Aditi       187
Anu         186
Yadav       185
Jackson     182
Kumar       181
Hafiz       174
Name: count, dtype: int64

Value counts for 'Category':
Category
Snacks               1514
Eggs, Meat & Fish    1490
Fruits & Veggies     1418
Bakery               1413
Beverages            1400
Food Grains          1398
Oil & Masala         1361
Name: count, dtype: int64

Value counts for 'Sub Category':
Sub Category
Health Drinks         719
Soft Drinks           681
Cookies               520
Breads & Buns         502
Chocolates            499
Noodles               495
Masalas               463
Biscuits              459
Cakes                 452
Edible Oil & Ghee     451
Spices                447
Mutton                394
Eggs                  379
Organic Staples       372
Fresh Fruits          369
Fish                  369
Fresh Vegetables      354
Atta & Flour          353
Organic Fruits        348
Chicken               348
Organic Vegetables    347
Dals & Pulses         343
Rice                  330
Name: count, dtype: int64

Value counts for 'City':
City
```

```
Kanyakumari       459
Tirunelveli       446
Bodi              442
Krishnagiri       440
Vellore           435
Perambalur        434
Tenkasi           432
Chennai           432
Salem             431
Karur             430
Pudukottai        430
Coimbatore        428
Ramanadhapuram    421
Cumbum            417
Virudhunagar      416
Madurai           408
Ooty              404
Namakkal          403
Viluppuram        397
Dindigul          396
Theni             387
Dharmapuri        376
Nagercoil         373
Trichy            357
Name: count, dtype: int64

Value counts for 'Order Date':
Order Date
09-05-2017    38
09-02-2018    36
11-10-2017    35
12-02-2018    34
12-01-2018    34
              ..
7/19/2016      1
07-08-2016     1
10/17/2016     1
3/29/2018      1
04-06-2017     1
Name: count, Length: 1236, dtype: int64

Value counts for 'Region':
Region
West       3203
East       2848
Central    2323
South      1619
North         1
Name: count, dtype: int64

Value counts for 'State':
State
Tamil Nadu    9994
Name: count, dtype: int64
```

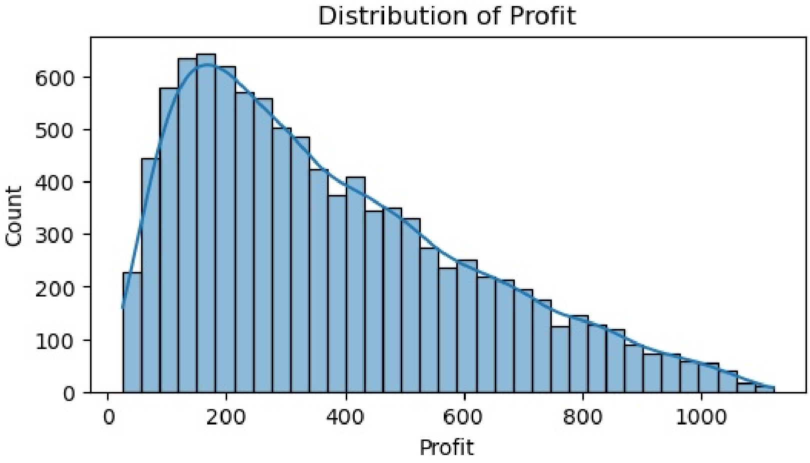
In [10]:

```
numerical_cols = df.select_dtypes(include=['int64', 'float64']).columns

for col in numerical_cols:
    plt.figure(figsize=(6, 3))
    sns.histplot(df[col], kde=True)
    plt.title(f"Distribution of {col}")
    plt.show()
```





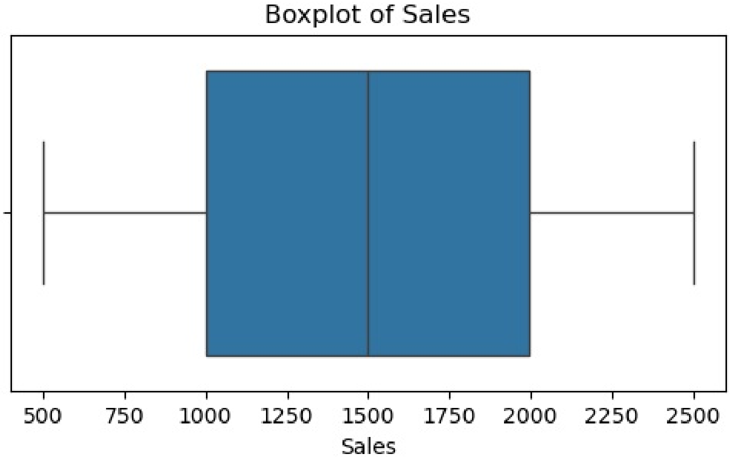
Distribution of Discount



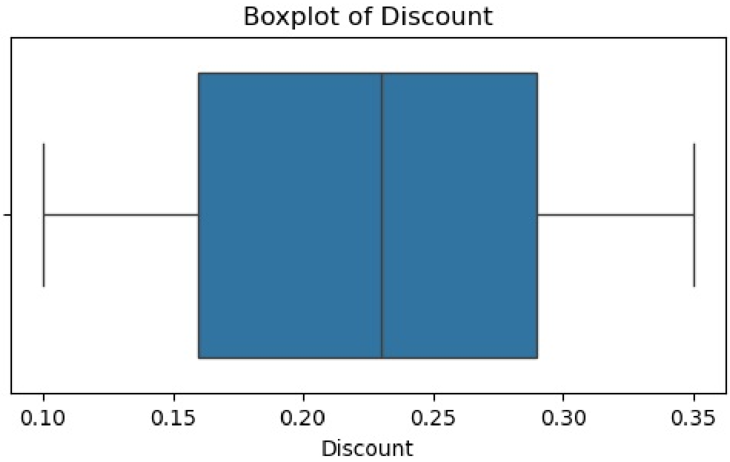
Distribution of Profit

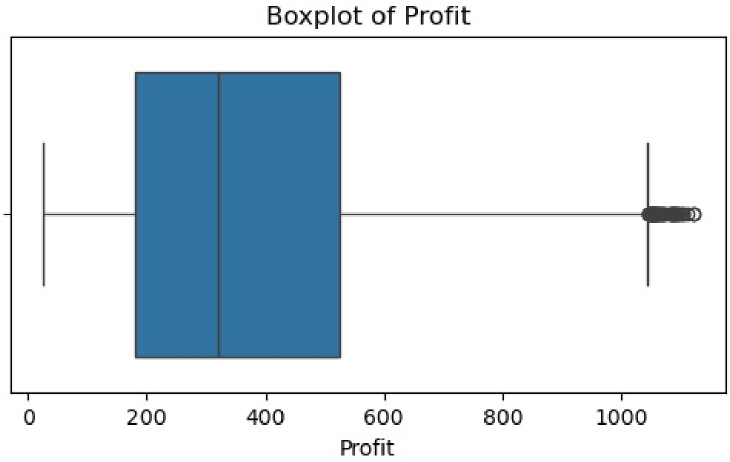
In [11]:

```
for col in numerical_cols:
    plt.figure(figsize=(6, 3))
    sns.boxplot(x=df[col])
    plt.title(f"Boxplot of {col}")
    plt.show()
```



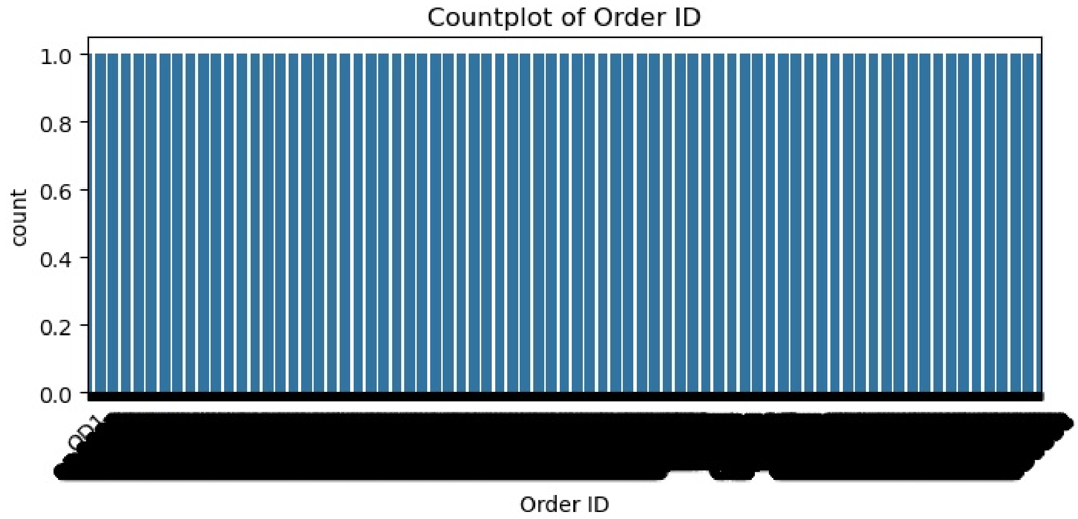
Boxplot of Sales

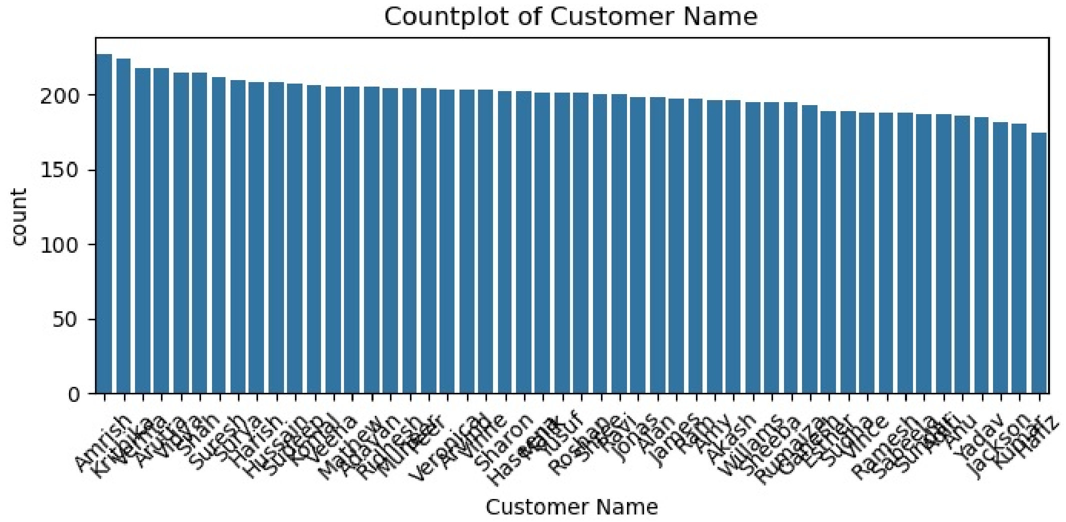


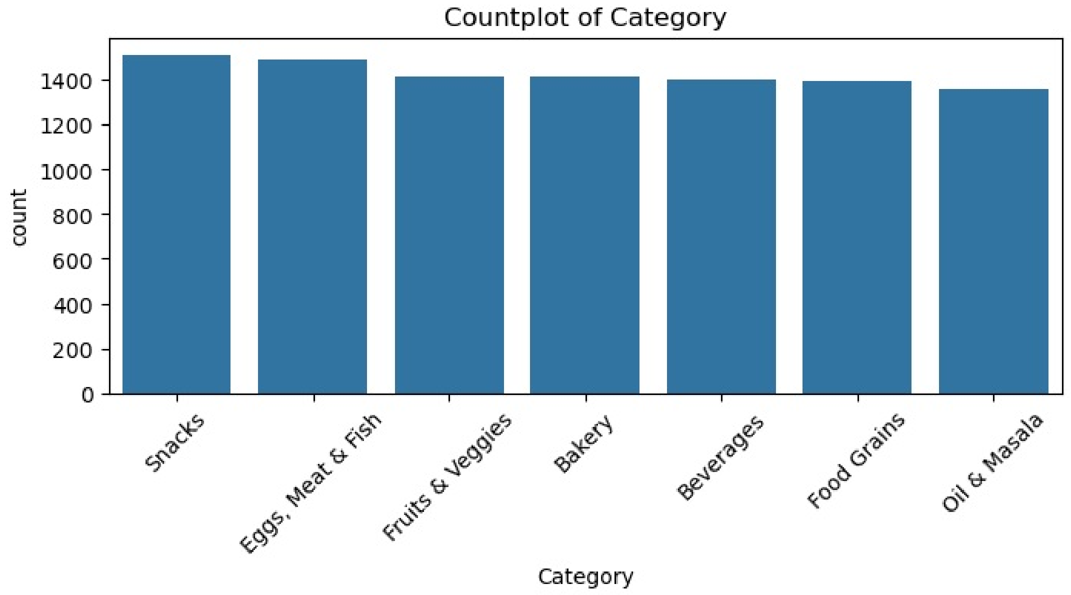


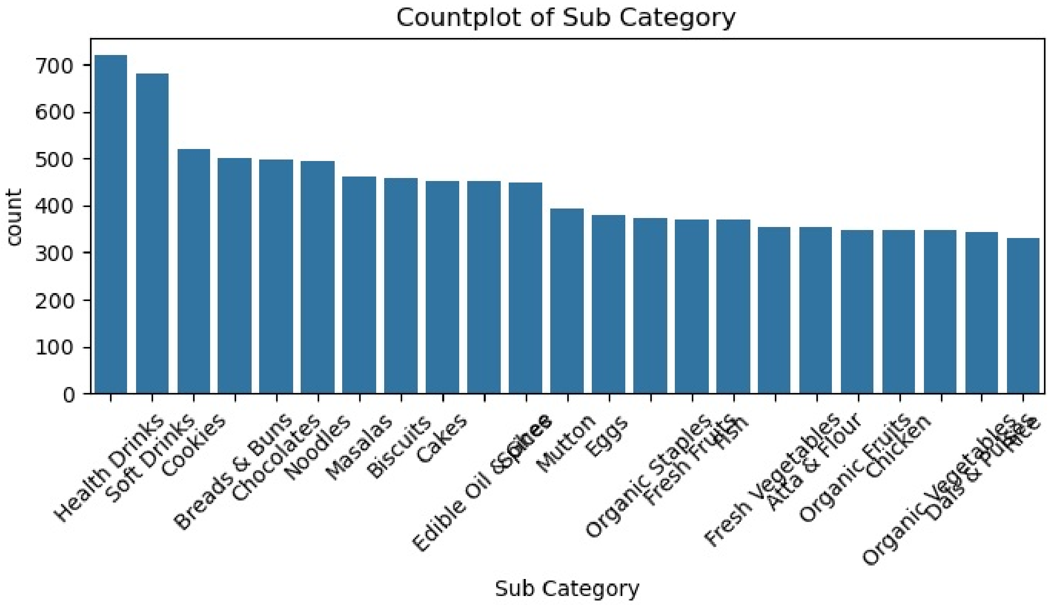
In [12]:

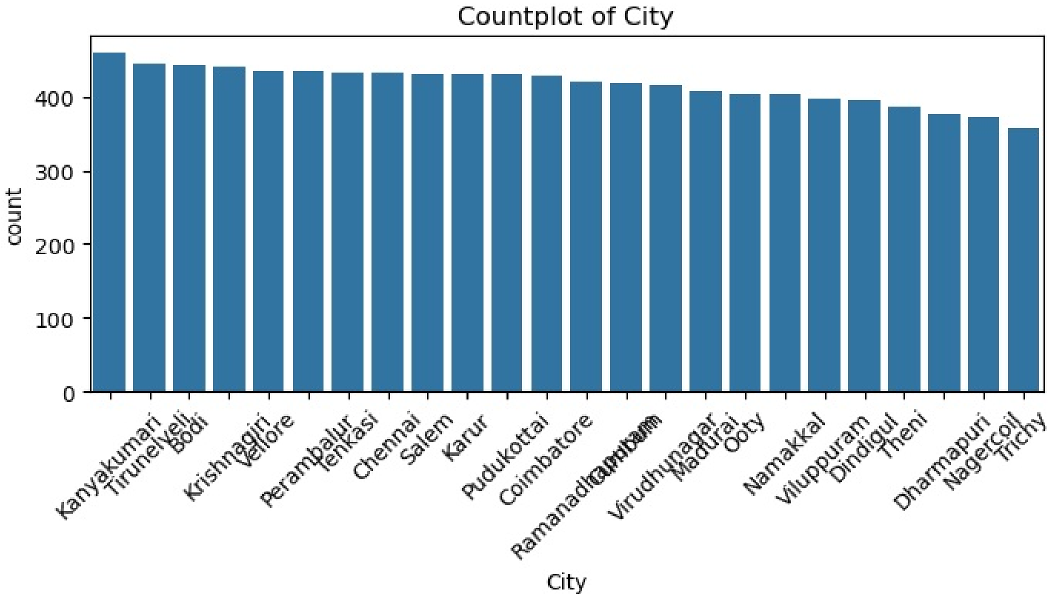
```
for col in categorical_cols:
    plt.figure(figsize=(8, 3))
    sns.countplot(data=df, x=col, order=df[col].value_counts().index)
    plt.title(f"Countplot of {col}")
    plt.xticks(rotation=45)
    plt.show()
```

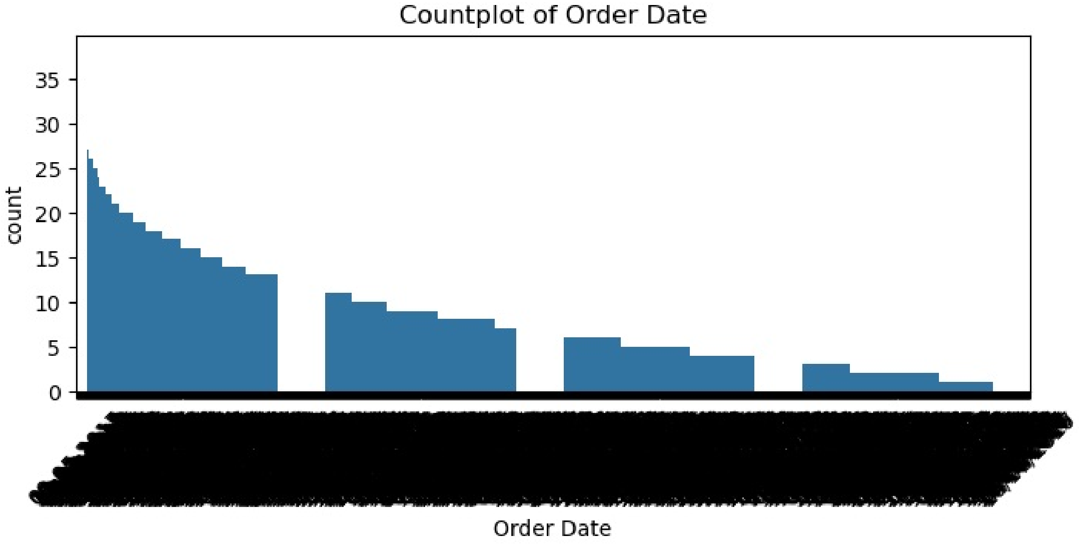

Countplot of Order ID
1.0
0.8
0.6
0.4
0.2
0.0
count
OD1
Order ID

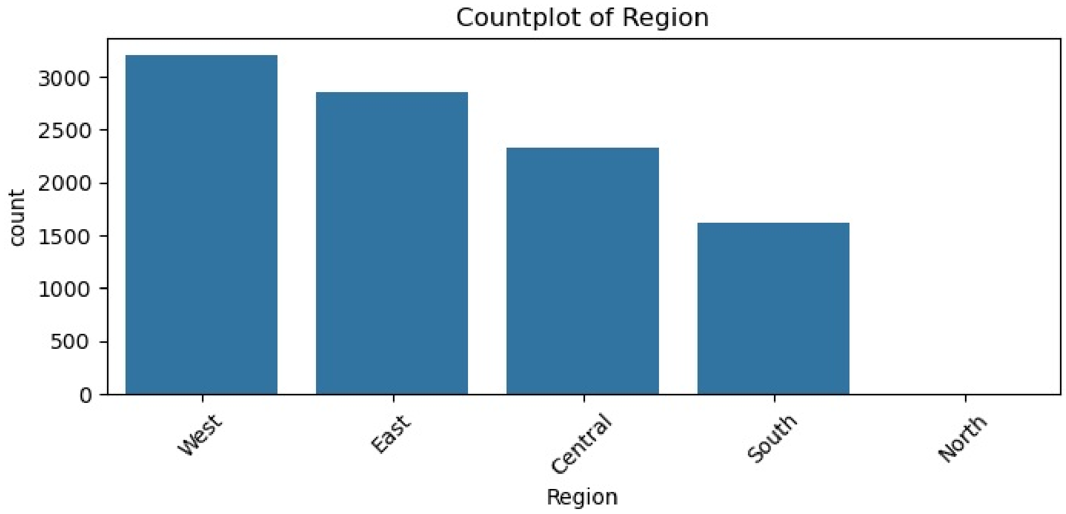

Countplot of Customer Name
200
150
100
50
0
count
Amrish
Verma
Vidya
Shah
Suresh
Surya
Harish
Hussain
Komal
Veena
Mathew
Adavan
Ridhesh
Muneer
Peer
Veronica
Arvind
Willie
Sharon
Yusuf
Roshan
Ravi
Jonas
Alan
James
Ram
Amy
Akash
Willams
Sheeba
Rumaiza
Esther
Sudha
Vince
Ramesh
Sabeela
Sundar
Anu
Yadav
Jackson
Kumar
Hafiz
Customer Name

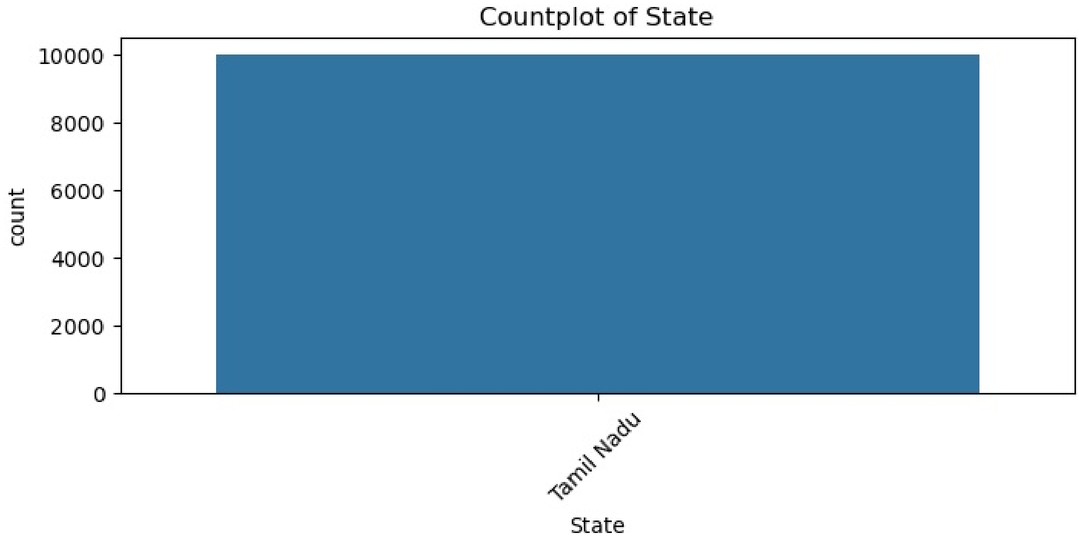

Countplot of Category
1400
1200
1000
800
600
400
200
0
count
Snacks
Eggs, Meat & Fish
Fruits & Veggies
Bakery
Beverages
Food Grains
Oil & Masala
Category


Countplot of Sub Category
count
700
600
500
400
300
200
100
0
Health Drinks
Soft Drinks
Cookies
Breads & Buns
Chocolates
Noodles
Masalas
Biscuits
Cakes
Edible Oil & Ghee
Spices
Mutton
Eggs
Organic Staples
Fresh Fruits
Fish
Fresh Vegetables
Atta & Flour
Organic Fruits
Chicken
Organic Vegetables
Dals & Pulses
Rice
Sub Category


Countplot of City
count
400
300
200
100
0
Kanyakumari
Tirunelveli
Bodi
Krishnagiri
Vellore
Perambalur
Tenkasi
Chennai
Salem
Karur
Pudukottai
Coimbatore
Ramanadhapuram
Cumbum
Virudhunagar
Madurai
Ooty
Namakkal
Viluppuram
Dindigul
Theni
Dharmapuri
Nagercoil
Trichy
City

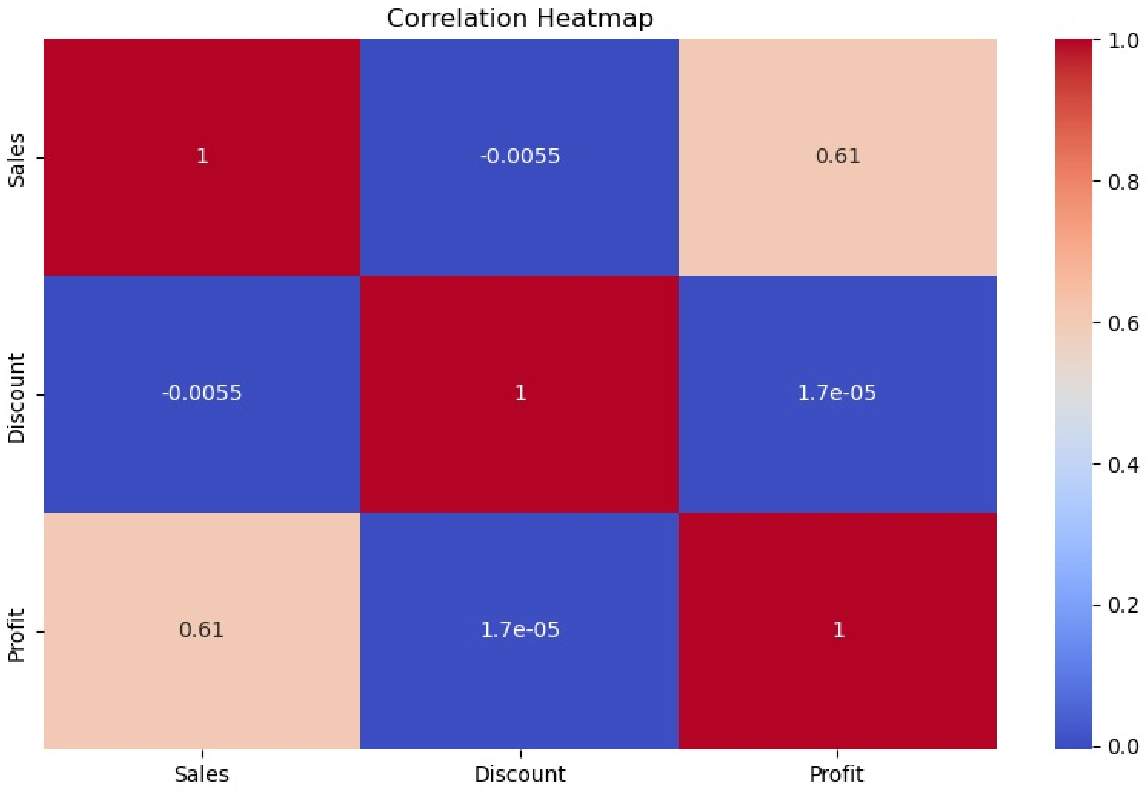

Countplot of Order Date
count
35
30
25
20
15
10
5
0
Order Date





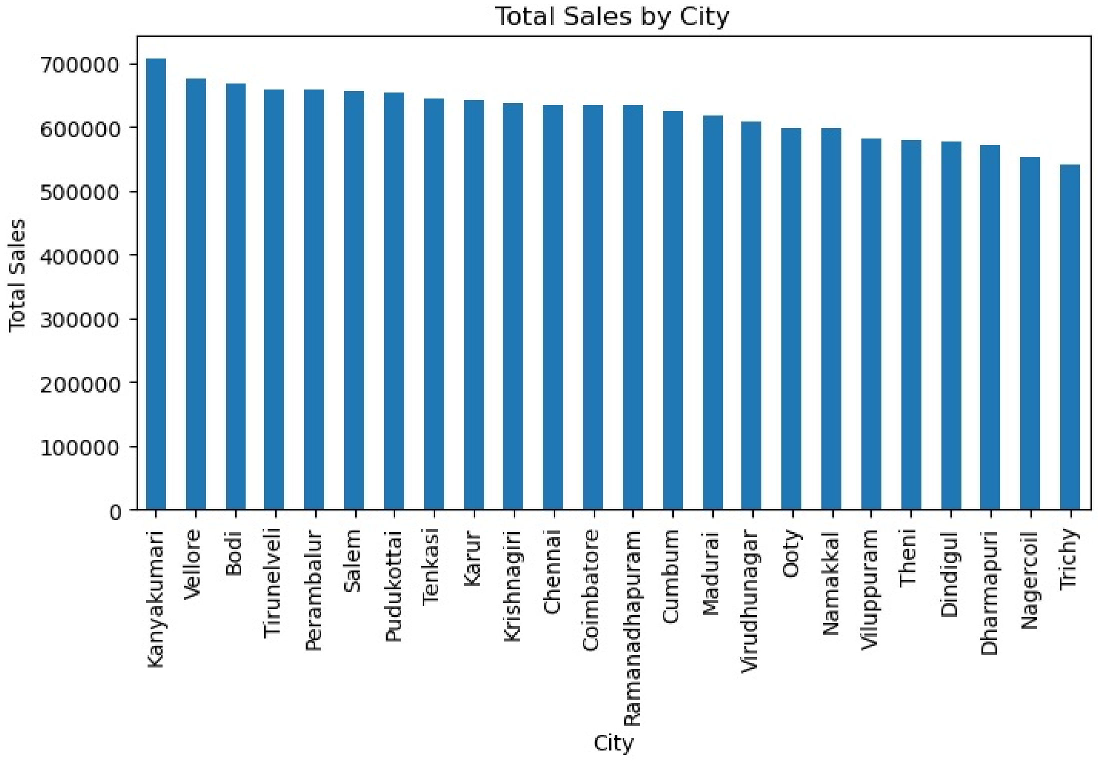
In [13]:

```
corr = df[numerical_cols].corr()
plt.figure(figsize=(10, 6))
sns.heatmap(corr, annot=True, cmap='coolwarm')
plt.title("Correlation Heatmap")
plt.show()
```



In [14]:

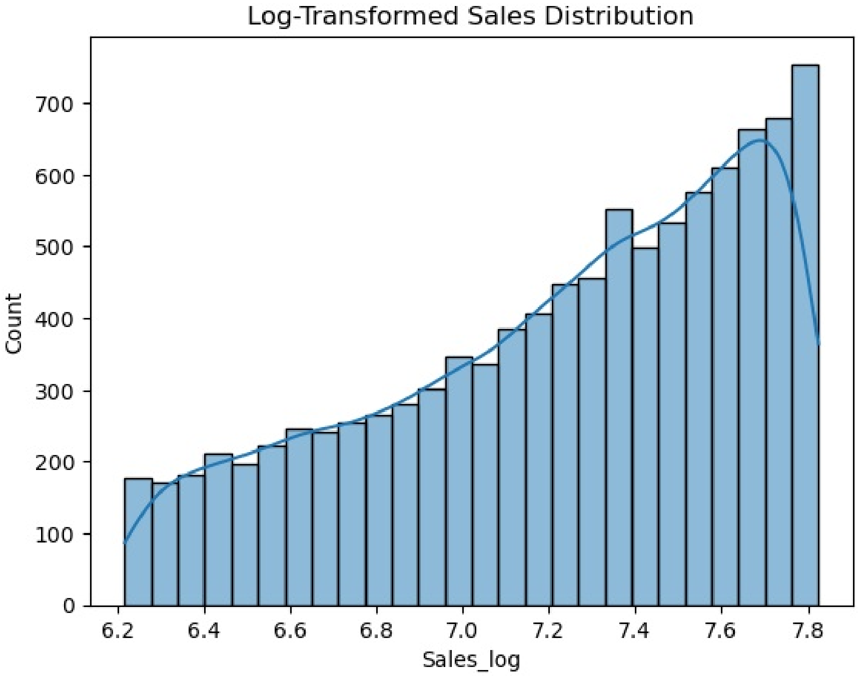
```
df.groupby("City")["Sales"].sum().sort_values(ascending=False).plot(kind="bar", figsize=(8,4), title="Total Sal
plt.ylabel("Total Sales")
plt.show()
```



In [15]:

```
print(df[numerical_cols].skew())

# Log transform highly skewed column
df['Sales_log'] = np.log1p(df['Sales'])
sns.histplot(df['Sales_log'], kde=True)
plt.title("Log-Transformed Sales Distribution")
plt.show()
```

```
Sales       0.000927
Discount   -0.026487
Profit      0.767397
dtype: float64
```



In [ ]:

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js