Superstore Dataset - Exploratory Data Analysis (EDA)

Objective: Extract meaningful insights through visual and statistical exploration. **Tools:** Python, Pandas, Matplotlib, Seaborn

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
sns.set(style='whitegrid')
import warnings
warnings.filterwarnings('ignore')
```

```
# Load the dataset
df = pd.read_csv('/content/superstore.csv.csv')
df.head()
```

	Row ID	Order ID	Order Date	Ship Date	Ship Mode	Customer ID	Customer Name	Segment	Country	City	...	Postal Code	Region	Product ID	Cat
0	1	CA-2017-152156	08/11/2017	11/11/2017	Second Class	CG-12520	Claire Gute	Consumer	Spain	Barcelona	...	42420.0	South	FUR-BO-10001798	Fu
1	2	CA-2017-152156	08/11/2017	11/11/2017	Second Class	CG-12520	Claire Gute	Consumer	United States	Henderson	...	42420.0	South	FUR-CH-10000454	Fu
2	3	CA-2017-138688	12/06/2017	16/06/2017	Second Class	DV-13045	Darrin Van Huff	Corporate	United States	Los Angeles	...	90036.0	West	OFF-LA-10000240	Su
3	4	US-2016-108966	11/10/2016	18/10/2016	Standard Class	SO-20335	Sean O'Donnell	Consumer	Germany	Munich	...	33311.0	South	FUR-TA-10000577	Fu
4	5	US-2016-108966	11/10/2016	18/10/2016	Standard Class	SO-20335	Sean O'Donnell	Consumer	United States	Fort Lauderdale	...	33311.0	South	OFF-ST-10000760	Su

5 rows × 21 columns

```
df.info()
df.describe()
df.isnull().sum()
df.duplicated().sum()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9024 entries, 0 to 9023
Data columns (total 21 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Row ID         9024 non-null   int64
 1   Order ID       9024 non-null   object
 2   Order Date     9024 non-null   object
 3   Ship Date      9024 non-null   object
 4   Ship Mode      9024 non-null   object
 5   Customer ID    9024 non-null   object
 6   Customer Name  9024 non-null   object
 7   Segment        9024 non-null   object
 8   Country        9024 non-null   object
 9   City           9024 non-null   object
 10  State          9024 non-null   object
 11  Postal Code    9021 non-null   float64
 12  Region         9024 non-null   object
 13  Product ID     9023 non-null   object
 14  Category       9023 non-null   object
 15  Sub-Category   9023 non-null   object
 16  Product Name   9023 non-null   object
 17  Sales          9023 non-null   float64
 18  Profit         9023 non-null   float64
 19  Quantity       9023 non-null   float64
 20  Discount       9023 non-null   float64
dtypes: float64(5), int64(1), object(15)
memory usage: 1.4+ MB
```

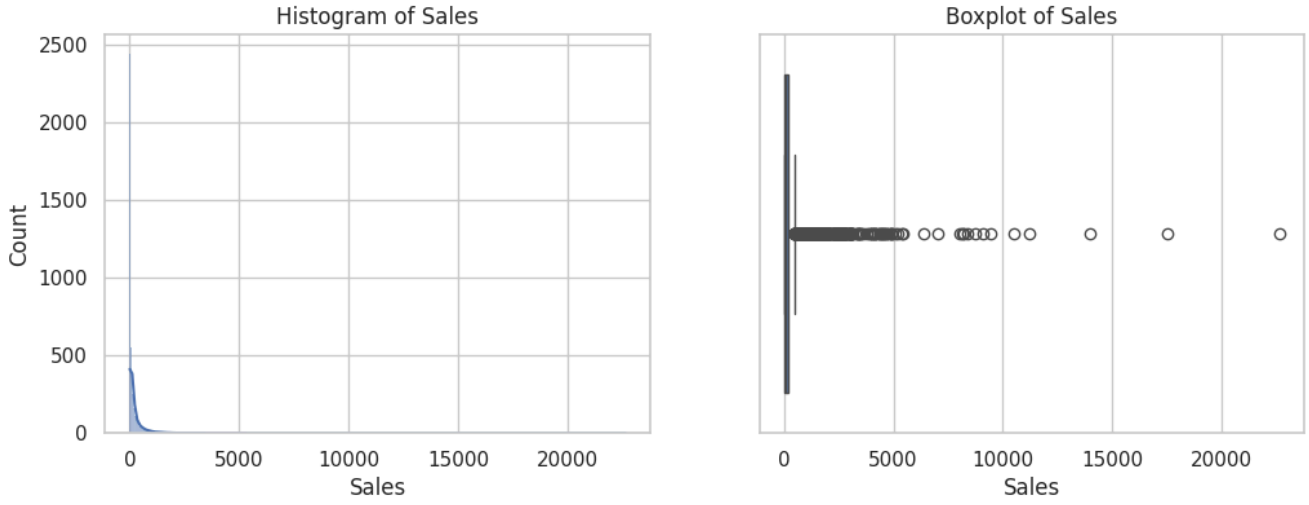
```
np.int64(0)
```

```
df['Order Date'] = pd.to_datetime(df['Order Date'], dayfirst=True)
df['Ship Date'] = pd.to_datetime(df['Ship Date'], dayfirst=True)
df.drop(columns=['Row ID'], inplace=True)
df.head()
```

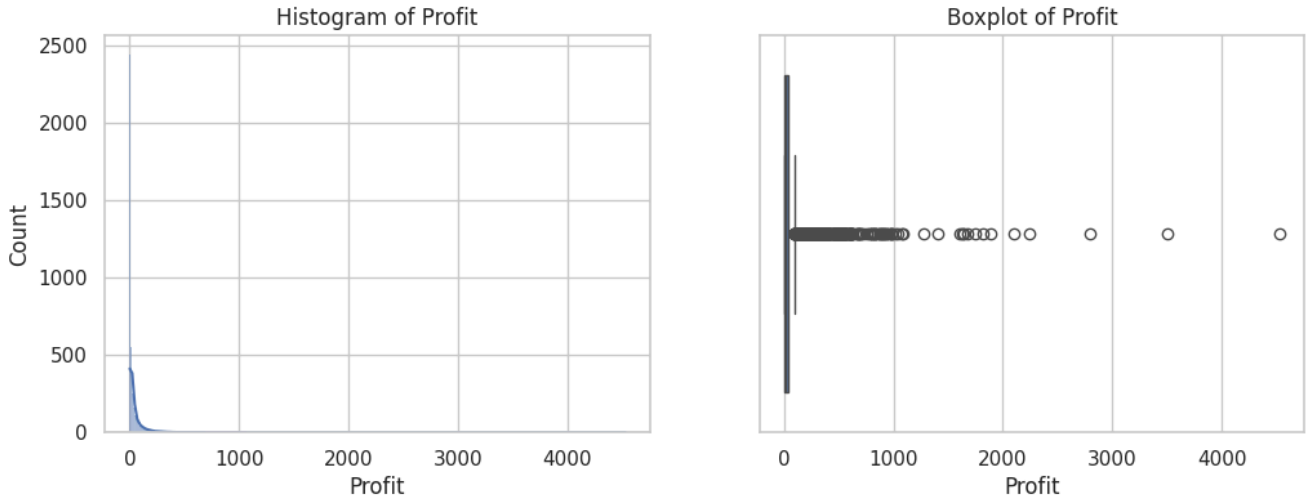
	Order ID	Order Date	Ship Date	Ship Mode	Customer ID	Customer Name	Segment	Country	City	State	Postal Code	Region	Product ID	Category	C
0	CA-2017-152156	2017-11-08	2017-11-11	Second Class	CG-12520	Claire Gute	Consumer	Spain	Barcelona	Catalonia	42420.0	South	FUR-BO-10001798	Furniture	Bc
1	CA-2017-152156	2017-11-08	2017-11-11	Second Class	CG-12520	Claire Gute	Consumer	United States	Henderson	Kentucky	42420.0	South	FUR-CH-10000454	Furniture	
2	CA-2017-138688	2017-06-12	2017-06-16	Second Class	DV-13045	Darrin Van Huff	Corporate	United States	Los Angeles	California	90036.0	West	OFF-LA-10000240	Office Supplies	
3	US-2016-108966	2016-10-11	2016-10-18	Standard Class	SO-20335	Sean O'Donnell	Consumer	Germany	Munich	Bavaria	33311.0	South	FUR-TA-10000577	Furniture	
4	US-2016-108966	2016-10-11	2016-10-18	Standard Class	SO-20335	Sean O'Donnell	Consumer	United States	Fort Lauderdale	Florida	33311.0	South	OFF-ST-10000760	Office Supplies	

Next steps: Generate code with df | View recommended plots | New interactive sheet

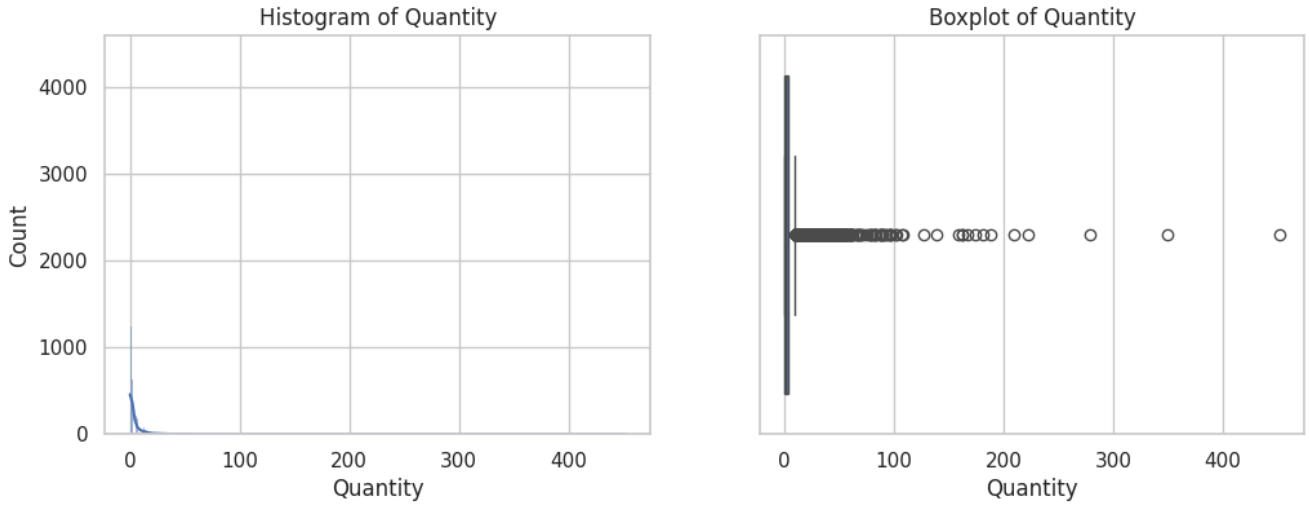
```
numerics = ['Sales', 'Profit', 'Quantity', 'Discount']
for col in numerics:
    plt.figure(figsize=(12,4))
    plt.subplot(1,2,1)
    sns.histplot(df[col], kde=True)
    plt.title(f'Histogram of {col}')

    plt.subplot(1,2,2)
    sns.boxplot(x=df[col])
    plt.title(f'Boxplot of {col}')
    plt.show()
```


Histogram of Sales
Count
Sales
0
500
1000
1500
2000
2500
0
5000
10000
15000
20000

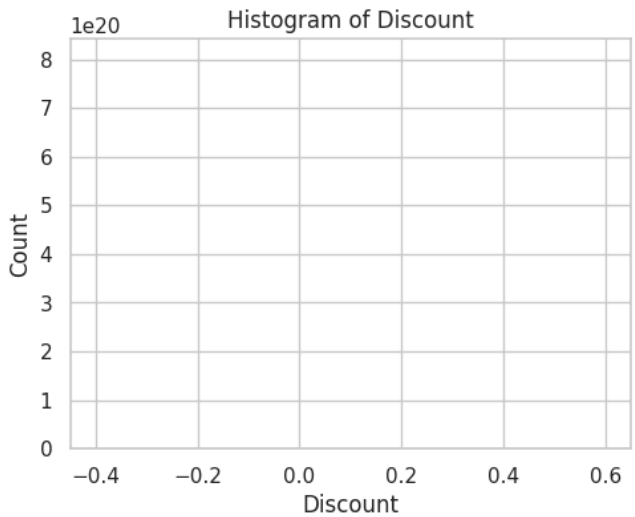
Boxplot of Sales
Sales
0
5000
10000
15000
20000

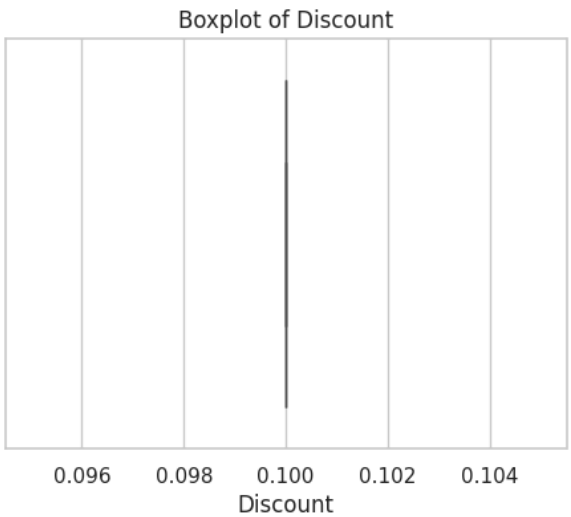

Histogram of Profit
Count
Profit
0
500
1000
1500
2000
2500
0
1000
2000
3000
4000

Boxplot of Profit
Profit
0
1000
2000
3000
4000


Histogram of Quantity
Count
Quantity
0
1000
2000
3000
4000
0
100
200
300
400

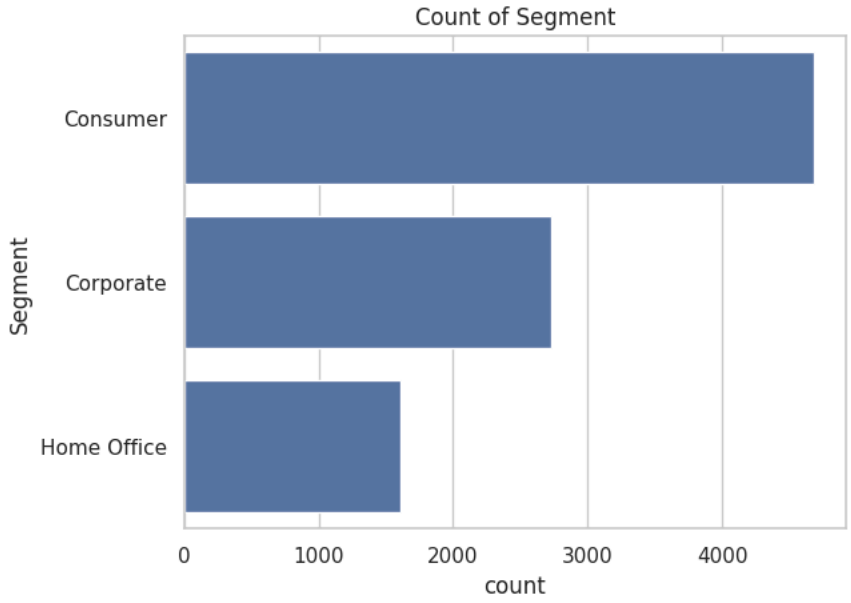
Boxplot of Quantity
Quantity
0
100
200
300
400


Histogram of Discount
1e20
Count
Discount
0
1
2
3
4
5
6
7
8
−0.4
−0.2
0.0
0.2
0.4
0.6

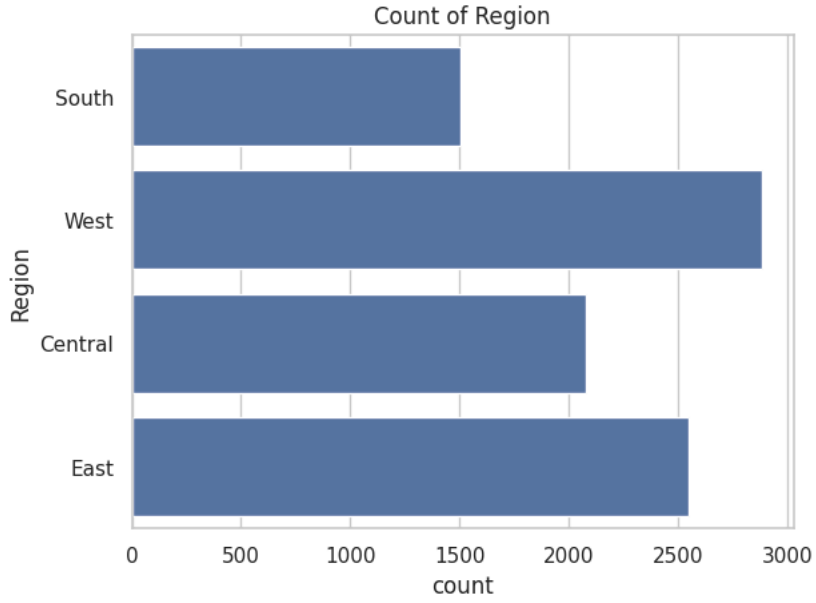

Boxplot of Discount
Discount
0.096
0.098
0.100
0.102
0.104

```
categorical = ['Segment', 'Region', 'Category', 'Sub-Category', 'Ship Mode']
for col in categorical:
    print(df[col].value_counts())
    sns.countplot(y=df[col])
    plt.title(f'Count of {col}')
    plt.show()
```

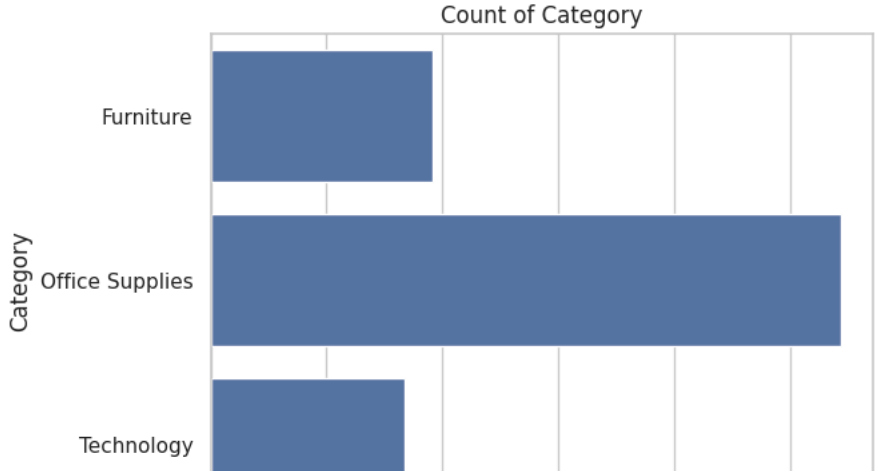
```
Segment
Consumer       4687
Corporate      2727
Home Office    1610
Name: count, dtype: int64
```

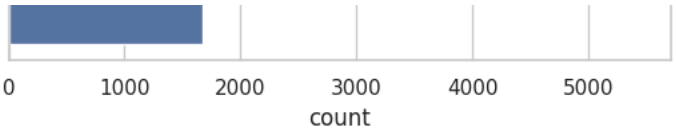



```
Region
West       2885
East       2551
Central    2081
South      1507
Name: count, dtype: int64
```

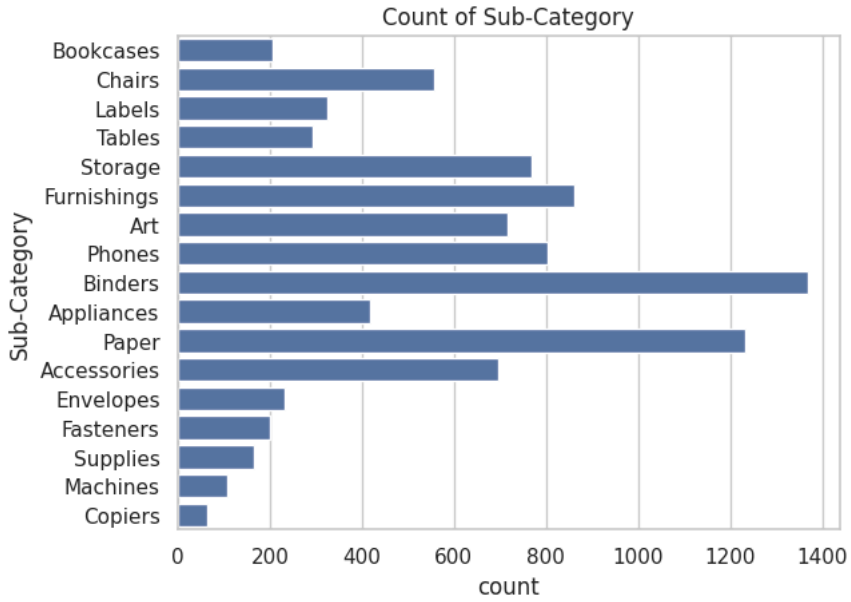



```
Category
Office Supplies    5434
Furniture          1915
Technology         1674
Name: count, dtype: int64
```

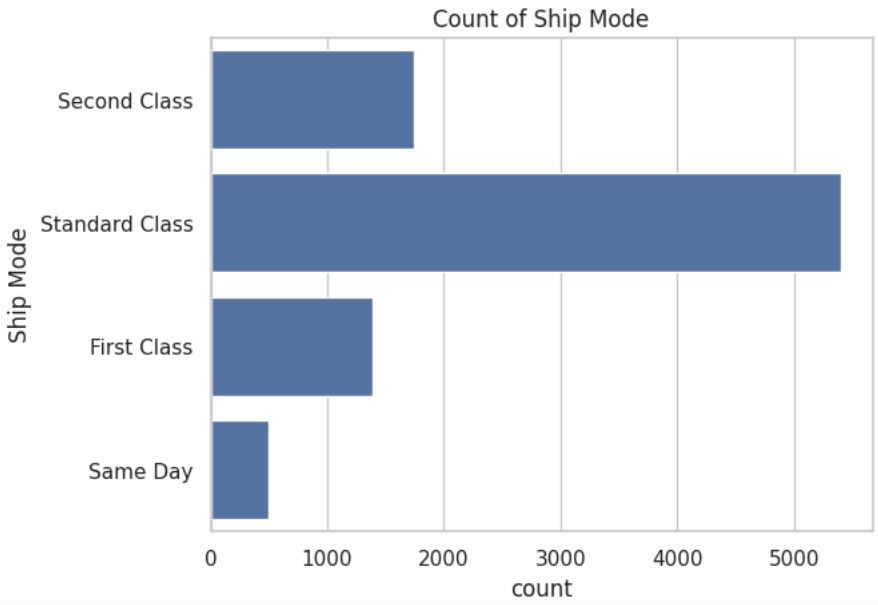





```
Sub-Category
Binders        1370
Paper          1234
Furnishings     861
Phones          805
Storage         770
Art             718
Accessories     695
Chairs          556
Appliances      417
Labels          326
Tables          293
Envelopes       231
Bookcases       205
Fasteners       201
Supplies        167
Machines        109
Copiers          65
Name: count, dtype: int64
```




```
Ship Mode
Standard Class    5404
Second Class      1743
First Class       1385
Same Day           492
Name: count, dtype: int64
```



```
sns.pairplot(df[numerics])
plt.show()
sns.heatmap(df[numerics].corr(), annot=True, cmap='coolwarm')
plt.title('Correlation Heatmap')
plt.show()
```