Task - 3 Exploratory Data Analysis - Retail

import liaberies

In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
import warnings
warnings.filterwarnings("ignore")
```

In [2]:

```
df = pd.read_csv("SampleSuperstore (1).csv")
df.head(15)
```

Out[2]:

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Cate
0	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Bookc
1	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	C
2	Second Class	Corporate	United States	Los Angeles	California	90036	West	Office Supplies	La
3	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Furniture	Ta
4	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Office Supplies	Sto
5	Standard Class	Consumer	United States	Los Angeles	California	90032	West	Furniture	Furnish
6	Standard Class	Consumer	United States	Los Angeles	California	90032	West	Office Supplies	
7	Standard Class	Consumer	United States	Los Angeles	California	90032	West	Technology	Ph
8	Standard Class	Consumer	United States	Los Angeles	California	90032	West	Office Supplies	Bin
9	Standard Class	Consumer	United States	Los Angeles	California	90032	West	Office Supplies	Applia
10	Standard Class	Consumer	United States	Los Angeles	California	90032	West	Furniture	Ta
11	Standard Class	Consumer	United States	Los Angeles	California	90032	West	Technology	Ph
12	Standard Class	Consumer	United States	Concord	North Carolina	28027	South	Office Supplies	P
13	Standard Class	Consumer	United States	Seattle	Washington	98103	West	Office Supplies	Bin
14	Standard Class	Home Office	United States	Fort Worth	Texas	76106	Central	Office Supplies	Applia

Data Exploration

In [13]:

```
df.head()                ##  first 5 rows of data set
```

Out[13]:

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Sub-Category
0	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Bookcases
1	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Chairs
2	Second Class	Corporate	United States	Los Angeles	California	90036	West	Office Supplies	Labels
3	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Furniture	Tables
4	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Office Supplies	Storage

In [14]:

```
df.tail()                ## last 5 rows of data set
```

Out[14]:

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Cat
9989	Second Class	Consumer	United States	Miami	Florida	33180	South	Furniture	Furni
9990	Standard Class	Consumer	United States	Costa Mesa	California	92627	West	Furniture	Furni
9991	Standard Class	Consumer	United States	Costa Mesa	California	92627	West	Technology	P
9992	Standard Class	Consumer	United States	Costa Mesa	California	92627	West	Office Supplies	
9993	Second Class	Consumer	United States	Westminster	California	92683	West	Office Supplies	Appli

In [15]:

```
df.shape
```

Out[15]:

```
(9994, 13)
```

In [16]:

```
df.info()                    #information on dataset
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9994 entries, 0 to 9993
Data columns (total 13 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Ship Mode     9994 non-null   object
 1   Segment       9994 non-null   object
 2   Country       9994 non-null   object
 3   City          9994 non-null   object
 4   State         9994 non-null   object
 5   Postal Code   9994 non-null   int64
 6   Region        9994 non-null   object
 7   Category      9994 non-null   object
 8   Sub-Category  9994 non-null   object
 9   Sales         9994 non-null   float64
 10  Quantity      9994 non-null   int64
 11  Discount      9994 non-null   float64
 12  Profit        9994 non-null   float64
dtypes: float64(3), int64(2), object(8)
memory usage: 1015.1+ KB
```

Data Cleaning

In [17]:

```
df.isnull().sum()
```

Out[17]:

```
Ship Mode       0
Segment         0
Country         0
City            0
State           0
Postal Code     0
Region          0
Category        0
Sub-Category    0
Sales           0
Quantity        0
Discount        0
Profit          0
dtype: int64
```

In [19]:

```
df.duplicated().sum()    #duplicate value
```

Out[19]:

```
17
```

In [20]:

```
df[df.duplicated(keep = 'last')]
```

Out[20]:

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	C
568	Standard Class	Corporate	United States	Seattle	Washington	98105	West	Office Supplies	
591	Standard Class	Consumer	United States	Salem	Oregon	97301	West	Office Supplies	
935	Standard Class	Home Office	United States	Philadelphia	Pennsylvania	19120	East	Office Supplies	
1186	Standard Class	Corporate	United States	Seattle	Washington	98103	West	Office Supplies	
1479	Standard Class	Consumer	United States	San Francisco	California	94122	West	Office Supplies	
2803	Standard Class	Consumer	United States	San Francisco	California	94122	West	Office Supplies	
2807	Second Class	Consumer	United States	Seattle	Washington	98115	West	Office Supplies	
2836	Standard Class	Consumer	United States	Los Angeles	California	90036	West	Office Supplies	
3127	Standard Class	Consumer	United States	New York City	New York	10011	East	Office Supplies	
3405	Standard Class	Home Office	United States	Columbus	Ohio	43229	East	Furniture	
3412	Standard Class	Corporate	United States	San Francisco	California	94122	West	Office Supplies	
5372	Standard Class	Corporate	United States	Houston	Texas	77041	Central	Office Supplies	
5493	Same Day	Home Office	United States	San Francisco	California	94122	West	Office Supplies	
6245	Standard Class	Home Office	United States	Seattle	Washington	98105	West	Furniture	Fur
6409	First Class	Consumer	United States	Houston	Texas	77041	Central	Office Supplies	
8457	Second Class	Corporate	United States	Chicago	Illinois	60653	Central	Office Supplies	
8533	Standard Class	Consumer	United States	Detroit	Michigan	48227	Central	Furniture	

In [21]:

```
df.drop_duplicates(inplace = True)
```

In [22]:

```
df.shape
```

Out[22]:

```
(9977, 13)
```

Calculated Field

In [23]:

```
df['Profit Margin %'] = (df.Profit / df.Sales) * 100
df.head(5)
```

Out[23]:

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Sub-Category
0	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Bookcases
1	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Chairs
2	Second Class	Corporate	United States	Los Angeles	California	90036	West	Office Supplies	Labels
3	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Furniture	Tables
4	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Office Supplies	Storage

Descriptive Statistics

In [24]:

```
df.describe(include = "all")
```

Out[24]:

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Cate
count	9977	9977	9977	9977	9977	9977.000000	9977	9977	
unique	4	3	1	531	49	NaN	4	3	
top	Standard Class	Consumer	United States	New York City	California	NaN	West	Office Supplies	Bi
freq	5955	5183	9977	914	1996	NaN	3193	6012	
mean	NaN	NaN	NaN	NaN	NaN	55154.964117	NaN	NaN	
std	NaN	NaN	NaN	NaN	NaN	32058.266816	NaN	NaN	
min	NaN	NaN	NaN	NaN	NaN	1040.000000	NaN	NaN	
25%	NaN	NaN	NaN	NaN	NaN	23223.000000	NaN	NaN	
50%	NaN	NaN	NaN	NaN	NaN	55901.000000	NaN	NaN	
75%	NaN	NaN	NaN	NaN	NaN	90008.000000	NaN	NaN	
max	NaN	NaN	NaN	NaN	NaN	99301.000000	NaN	NaN	

Exploratory Data Analysis

In [25]:

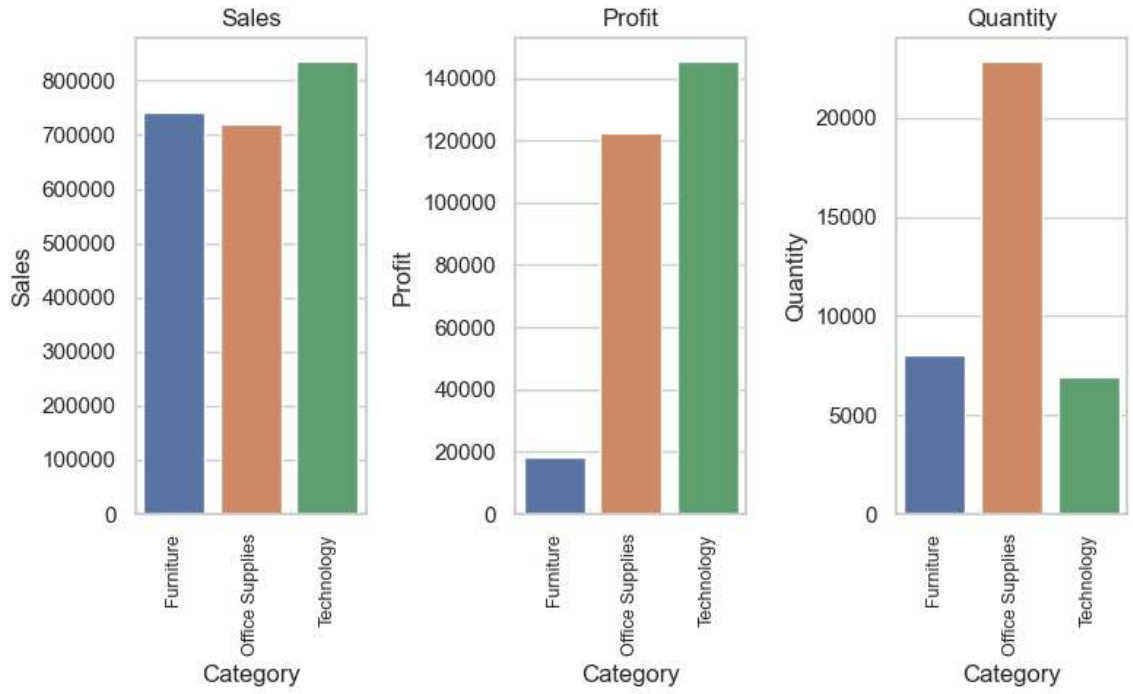
```
# Group sales, profit and quantity by category
category_analysis = pd.DataFrame(df.groupby(['Category'])[['Sales', 'Profit', 'Quantity'
category_analysis
```

Out[25]:

Category	Sales	Profit	Quantity
Furniture	741306.3133	18421.8137	8020
Office Supplies	718735.2440	122364.6608	22861
Technology	836154.0330	145454.9481	6939

In [26]:

```
# Set for grouped plots - figure with a 2x2 grid of Axes
sns.set_theme(style="whitegrid")
figure, axis = plt.subplots(1, 3, figsize=(8, 5))

# Plot barplots
cat1 = sns.barplot(x = category_analysis.index, y = category_analysis.Sales, ax=axis[0])
cat2 = sns.barplot(x = category_analysis.index, y = category_analysis.Profit, ax=axis[1]
cat3 = sns.barplot(x = category_analysis.index, y = category_analysis.Quantity, ax=axis[

# Set titles
cat1.set(title = 'Sales')
cat2.set(title = 'Profit')
cat3.set(title = 'Quantity')

# Rotate axis for x-axis
plt.setp(cat1.get_xticklabels(), rotation = 'vertical', size = 9)
plt.setp(cat2.get_xticklabels(), rotation = 'vertical', size = 9)
plt.setp(cat3.get_xticklabels(), rotation = 'vertical', size = 9)

# Set spacing between subplots
figure.tight_layout()
```



In [28]:

```
# Group by sub-category
subcat_analysis = pd.DataFrame(df.groupby(['Sub-Category'])[['Sales', 'Profit']].sum())
```

In [29]:

```
# Sort by descending order according to sales
subcat_sales = pd.DataFrame(subcat_analysis.sort_values('Sales', ascending = False))
subcat_sales
```

Out[29]:

Sub-Category	Sales	Profit
Phones	330007.0540	44515.7306
Chairs	327777.7610	26567.1278
Storage	223843.6080	21278.8264
Tables	206965.5320	-17725.4811
Binders	203409.1690	30228.0003
Machines	189238.6310	3384.7569
Accessories	167380.3180	41936.6357
Copiers	149528.0300	55617.8249
Bookcases	114879.9963	-3472.5560
Appliances	107532.1610	18138.0054
Furnishings	91683.0240	13052.7230
Paper	78224.1420	33944.2395
Supplies	46673.5380	-1189.0995
Art	27107.0320	6524.6118
Envelopes	16476.4020	6964.1767
Labels	12444.9120	5526.3820
Fasteners	3024.2800	949.5182

In [30]:

```
# Sort by descending order according to profit
subcat_profit = pd.DataFrame(subcat_analysis.sort_values('Profit', ascending = False))
subcat_profit
```

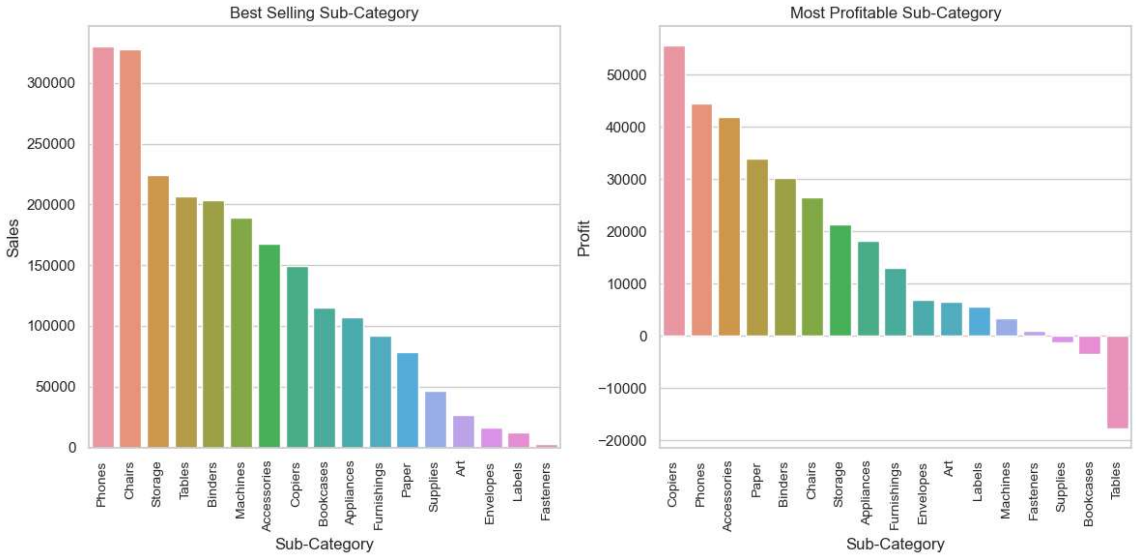
Out[30]:

Sub-Category	Sales	Profit
Copiers	149528.0300	55617.8249
Phones	330007.0540	44515.7306
Accessories	167380.3180	41936.6357
Paper	78224.1420	33944.2395
Binders	203409.1690	30228.0003
Chairs	327777.7610	26567.1278
Storage	223843.6080	21278.8264
Appliances	107532.1610	18138.0054
Furnishings	91683.0240	13052.7230
Envelopes	16476.4020	6964.1767
Art	27107.0320	6524.6118
Labels	12444.9120	5526.3820
Machines	189238.6310	3384.7569
Fasteners	3024.2800	949.5182
Supplies	46673.5380	-1189.0995
Bookcases	114879.9963	-3472.5560
Tables	206965.5320	-17725.4811

Plot Bar Plots

In [32]:

```
sns.set_theme(style="whitegrid")

# Set for grouped plots - figure with a 1x2 grid of Axes
figure, axis = plt.subplots(1, 2, figsize=(12, 6))

# Plot Bar Plot for Best Selling Sub-Category
subcat1 = sns.barplot(data = subcat_sales, x = subcat_sales.index, y = subcat_sales.Sale
subcat1.set(title="Best Selling Sub-Category")
subcat1.set_xticklabels(subcat1.get_xticklabels(),rotation = "vertical", size = 10)

# Plot Bar Plot for Most Profitable Sub-Category
subcat2 = sns.barplot(data = subcat_profit, x = subcat_profit.index, y = subcat_profit.P
subcat2.set(title = "Most Profitable Sub-Category")
subcat2.set_xticklabels(subcat2.get_xticklabels(),rotation = "vertical", size = 10)

# Set spacing between subplots
figure.tight_layout()

plt.show()
```



In [34]:

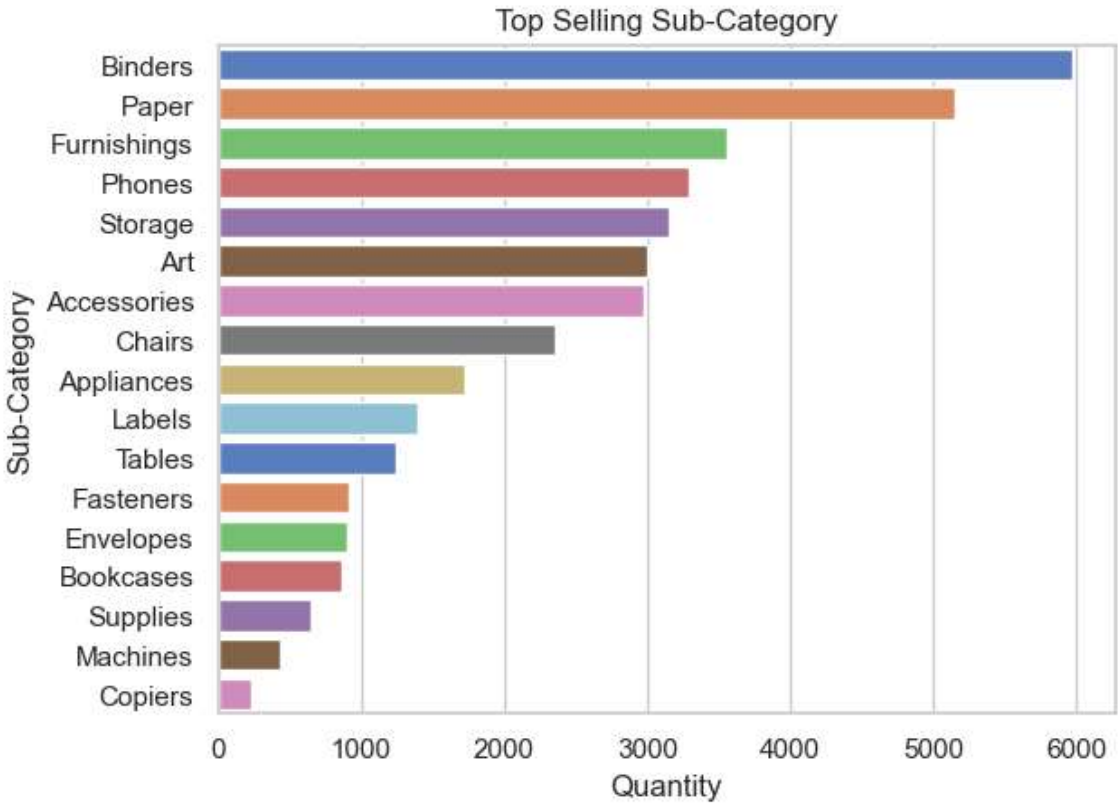
```
subcat_quantity = pd.DataFrame(df.groupby(['Sub-Category'])[['Quantity']].sum().sort_val
subcat_quantity
```

Out[34]:

Sub-Category	Quantity
Binders	5971
Paper	5144
Furnishings	3560
Phones	3289
Storage	3158
Art	2996
Accessories	2976
Chairs	2351
Appliances	1729
Labels	1396
Tables	1241
Fasteners	914
Envelopes	906
Bookcases	868
Supplies	647
Machines	440
Copiers	234

In [35]:

```
# Plot Bar Plot for Top Selling Sub-Category
sns.set_theme(style="whitegrid")
sns.barplot(data = subcat_quantity, y = subcat_quantity.index, x = subcat_quantity.Quant
plt.title("Top Selling Sub-Category")
plt.show()
```



In [37]:

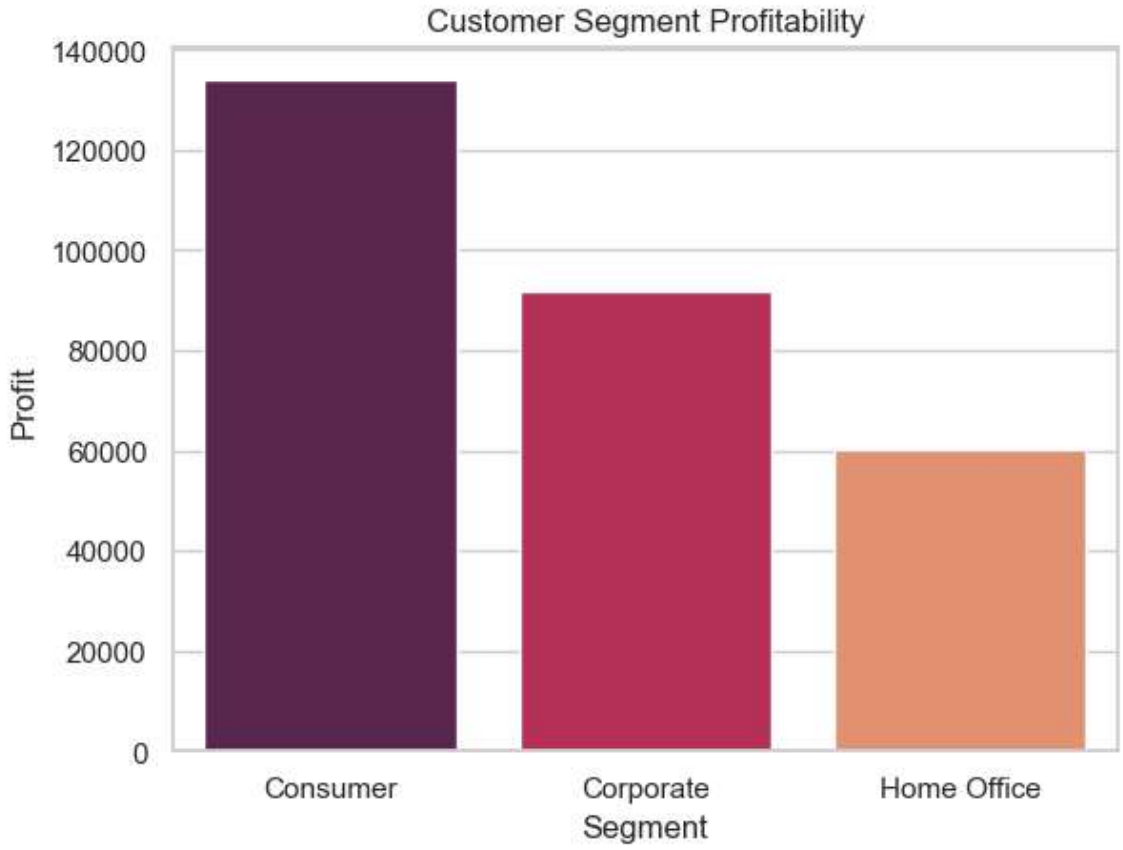
```
segment_analysis = pd.DataFrame(df.groupby(['Segment'])[['Profit']].sum())
segment_analysis
```

Out[37]:

Segment	Profit
Consumer	134007.4413
Corporate	91954.9798
Home Office	60279.0015

In [43]:

```
# Plot Bar Plot
sns.set_theme(style="whitegrid")
sns.barplot(data = segment_analysis, x = segment_analysis.index, y = segment_analysis.Pr
plt.title("Customer Segment Profitability")
plt.show()
```



In [ ]: