

```
In [13]: import os
os.getcwd()
```

```
Out[13]: '/Users/rishikasangolli/Desktop/Rishika'
```

Loading the Data

```
In [15]: import pandas as pd

# Load the dataset from the same directory
df = pd.read_csv('SampleSuperstore.csv', encoding='ISO-8859-1')

# Preview the data
df.head()
```

```
Out[15]:
```

	Row ID	Order ID	Order Date	Ship Date	Ship Mode	Customer ID	Customer Name	Segment	Country	
0	1	CA-2016-152156	11/8/2016	11/11/2016	Second Class	CG-12520	Claire Gute	Consumer	United States	Her
1	2	CA-2016-152156	11/8/2016	11/11/2016	Second Class	CG-12520	Claire Gute	Consumer	United States	Her
2	3	CA-2016-138688	6/12/2016	6/16/2016	Second Class	DV-13045	Darrin Van Huff	Corporate	United States	,
3	4	US-2015-108966	10/11/2015	10/18/2015	Standard Class	SO-20335	Sean O'Donnell	Consumer	United States	Lau
4	5	US-2015-108966	10/11/2015	10/18/2015	Standard Class	SO-20335	Sean O'Donnell	Consumer	United States	Lau

5 rows × 21 columns

BASIC INFOMATION

```
In [16]: # Get a summary of the dataset
df.info()

# Check for missing values
df.isnull().sum()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9994 entries, 0 to 9993
Data columns (total 21 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Row ID                 9994 non-null  int64
1   Order ID               9994 non-null  object
2   Order Date             9994 non-null  object
3   Ship Date              9994 non-null  object
4   Ship Mode              9994 non-null  object
5   Customer ID            9994 non-null  object
6   Customer Name          9994 non-null  object
7   Segment                9994 non-null  object
8   Country                9994 non-null  object
9   City                   9994 non-null  object
10  State                  9994 non-null  object
11  Postal Code            9994 non-null  int64
12  Region                 9994 non-null  object
13  Product ID             9994 non-null  object
14  Category               9994 non-null  object
15  Sub-Category           9994 non-null  object
16  Product Name           9994 non-null  object
17  Sales                  9994 non-null  float64
18  Quantity               9994 non-null  int64
19  Discount               9994 non-null  float64
20  Profit                 9994 non-null  float64
```

```
dtypes: float64(3), int64(3), object(15)
```

```
memory usage: 1.6+ MB
```

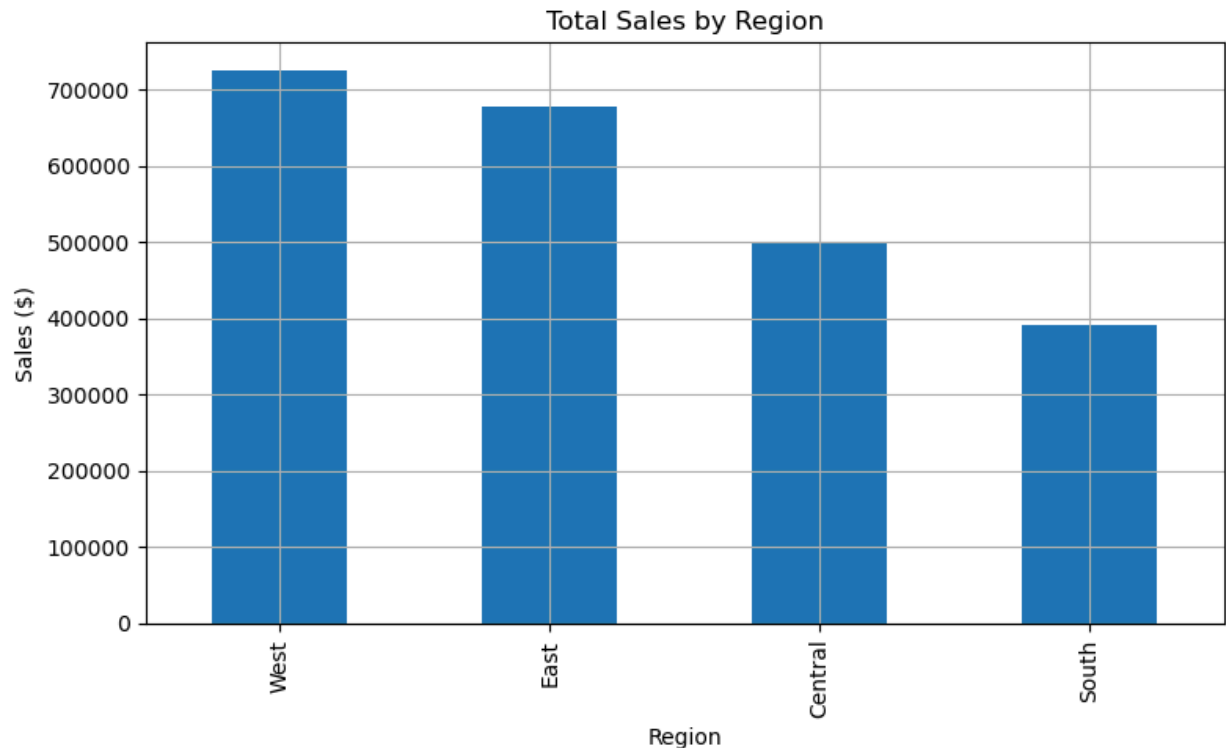
```
Out[16]: Row ID                0
Order ID                0
Order Date              0
Ship Date               0
Ship Mode               0
Customer ID             0
Customer Name           0
Segment                 0
Country                 0
City                    0
State                   0
Postal Code             0
Region                  0
Product ID              0
Category                0
Sub-Category            0
Product Name            0
Sales                   0
Quantity                0
Discount                0
Profit                  0
dtype: int64
```

Sales by Region

```
In [17]: import matplotlib.pyplot as plt
import seaborn as sns

# Group by Region
region_sales = df.groupby('Region')['Sales'].sum().sort_values(ascending=False)
```

```
# Plot
region_sales.plot(kind='bar', title='Total Sales by Region', figsize=(8,5))
plt.ylabel('Sales ($)')
plt.xlabel('Region')
plt.grid(True)
plt.tight_layout()
plt.show()
```



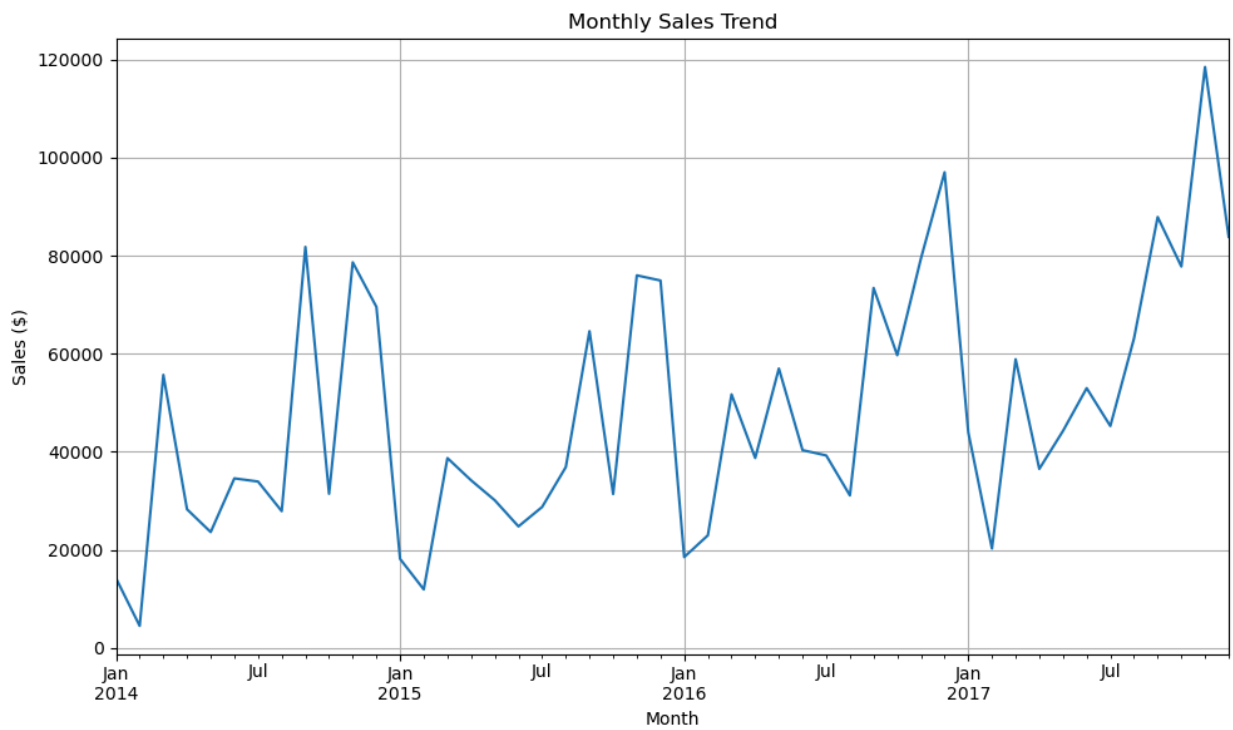
Monthly Sales Trend - Converting Dates and Resample

```
In [18]: # Convert 'Order Date' to datetime
df['Order Date'] = pd.to_datetime(df['Order Date'])

# Set it as index
df.set_index('Order Date', inplace=True)

# Resample monthly and sum sales
monthly_sales = df.resample('M')['Sales'].sum()

# Plot
monthly_sales.plot(figsize=(10,6), title='Monthly Sales Trend')
plt.ylabel('Sales ($)')
plt.xlabel('Month')
plt.grid(True)
plt.tight_layout()
plt.show()
```

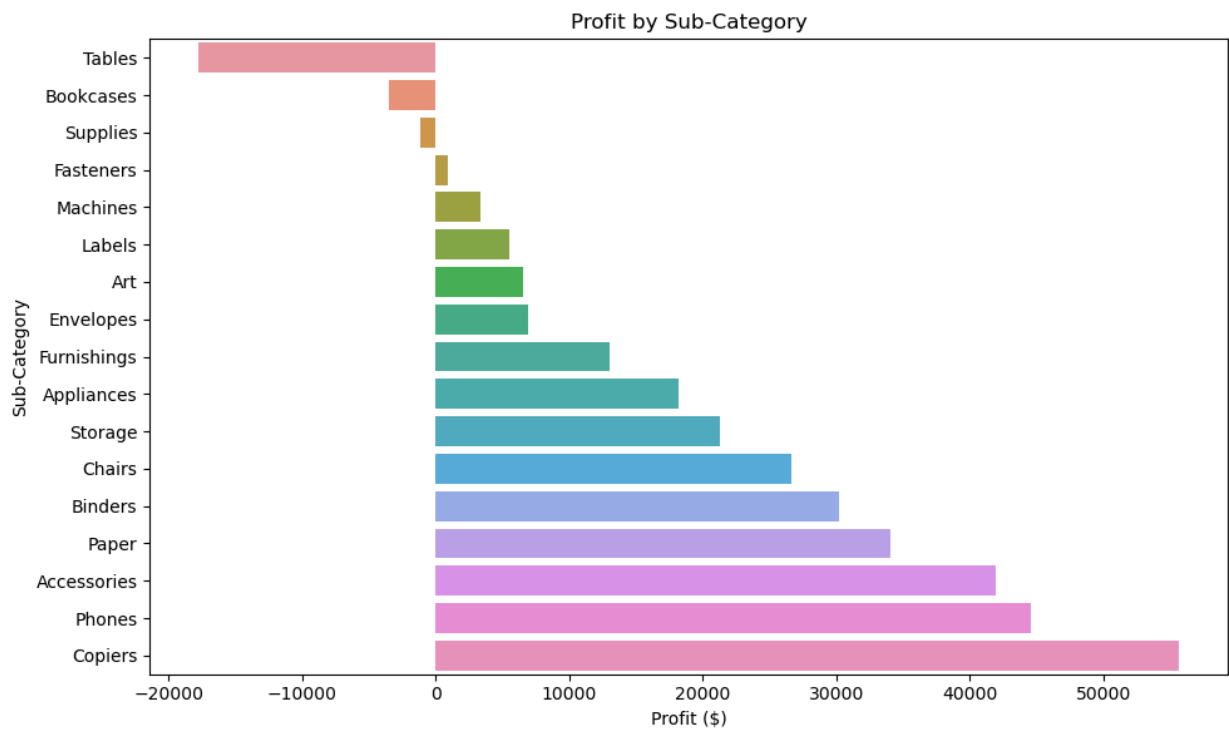


Profit by Sub-Category

```
In [19]: # Reset index to access Sub-Category
df.reset_index(inplace=True)

# Profit by Sub-Category
subcat_profit = df.groupby('Sub-Category')['Profit'].sum().sort_values()

# Plot
plt.figure(figsize=(10,6))
sns.barplot(x=subcat_profit.values, y=subcat_profit.index)
plt.title('Profit by Sub-Category')
plt.xlabel('Profit ($)')
plt.tight_layout()
plt.show()
```



In []:

