



E-Commerce Data Analysis: Unlocking Business Insights!

A Data-Driven Approach Using Python,
Seaborn & Matplotlib

Projects's agenda

Problem Statement:

How can data help
businesses optimize sales
& profitability?



Objective: Analyze sales
trends, customer
behavior & profitability



Dataset Overview & Cleaning Process

03

Data Source: "Superstore sales data"

```
[1]: import pandas as pd  
import seaborn as sns  
import matplotlib.pyplot as plt  
  
[2]: df = pd.read_csv("Sample _Superstore.csv",encoding = 'latin1')  
  
[3]: df.head()
```

	Row ID	Order ID	Order Date	Ship Date	Ship Mode	Customer ID	Customer Name	Segment	Country	City	...	Postal Code	Region	Product ID	Category	Sub-Category
0	1	CA-2016-152156	11/8/2016	11/11/2016	Second Class	CG-12520	Claire Gute	Consumer	United States	Henderson	...	42420	South	FUR-BO-10001798	Furniture	Bookcases
1	2	CA-2016-152156	11/8/2016	11/11/2016	Second Class	CG-12520	Claire Gute	Consumer	United States	Henderson	...	42420	South	FUR-CH-10000454	Furniture	Chairs

Dataset Overview & Cleaning Process

04

```
[4]: df.info()

<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
```

Data Cleaning Steps:

- Handled missing values
- Converted dates
- Removed duplicates

```
[6]: df['Order Date'] = pd.to_datetime(df['Order Date'])
```

```
[7]: df['Ship Date'] = pd.to_datetime(df['Ship Date'])
```

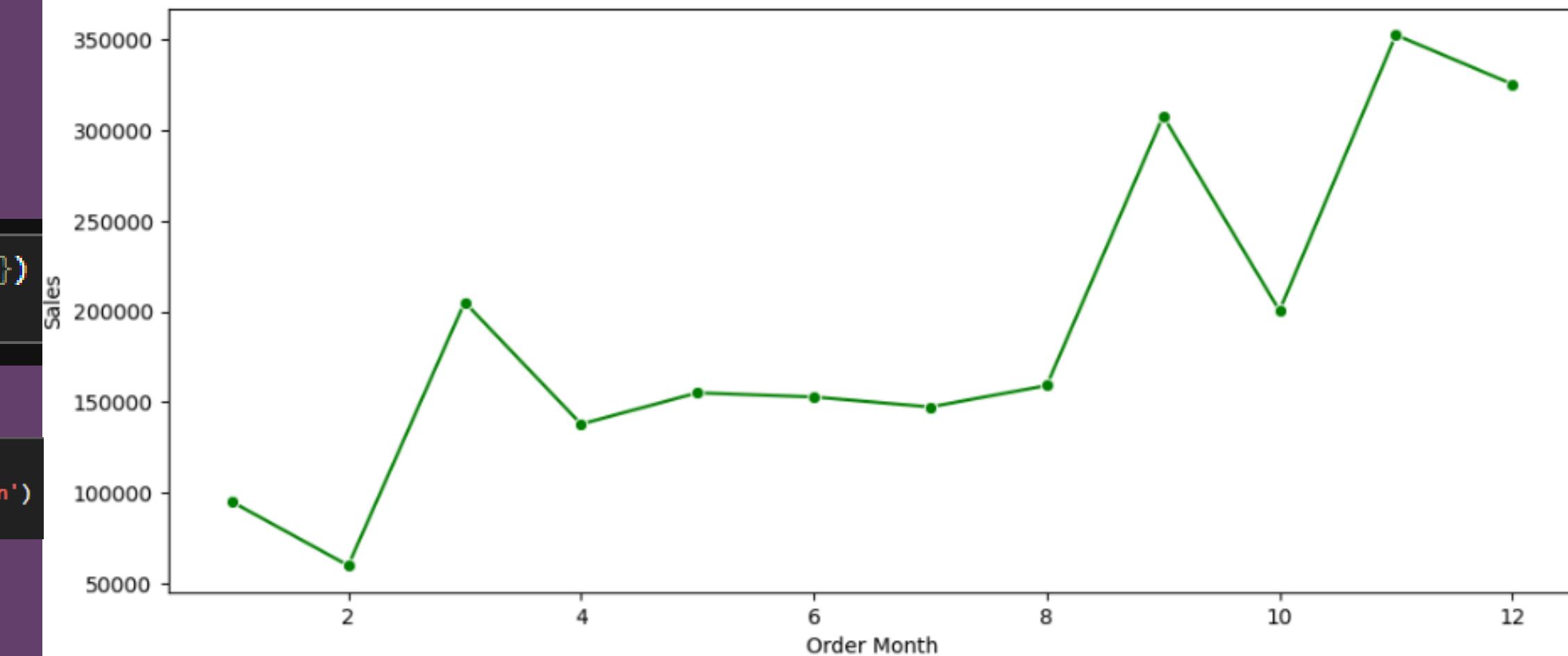
Sales Trends Analysis (Visualizations)

05

Show trends over time

```
salesByMonth = df.groupby("Order Month").agg({"Sales":"sum"})
salesByMonth
```

```
plt.figure(figsize=(12, 5))
sns.lineplot(data=salesByMonth,x = 'Order Month',y = 'Sales', marker='o', color='green')
plt.show()
```

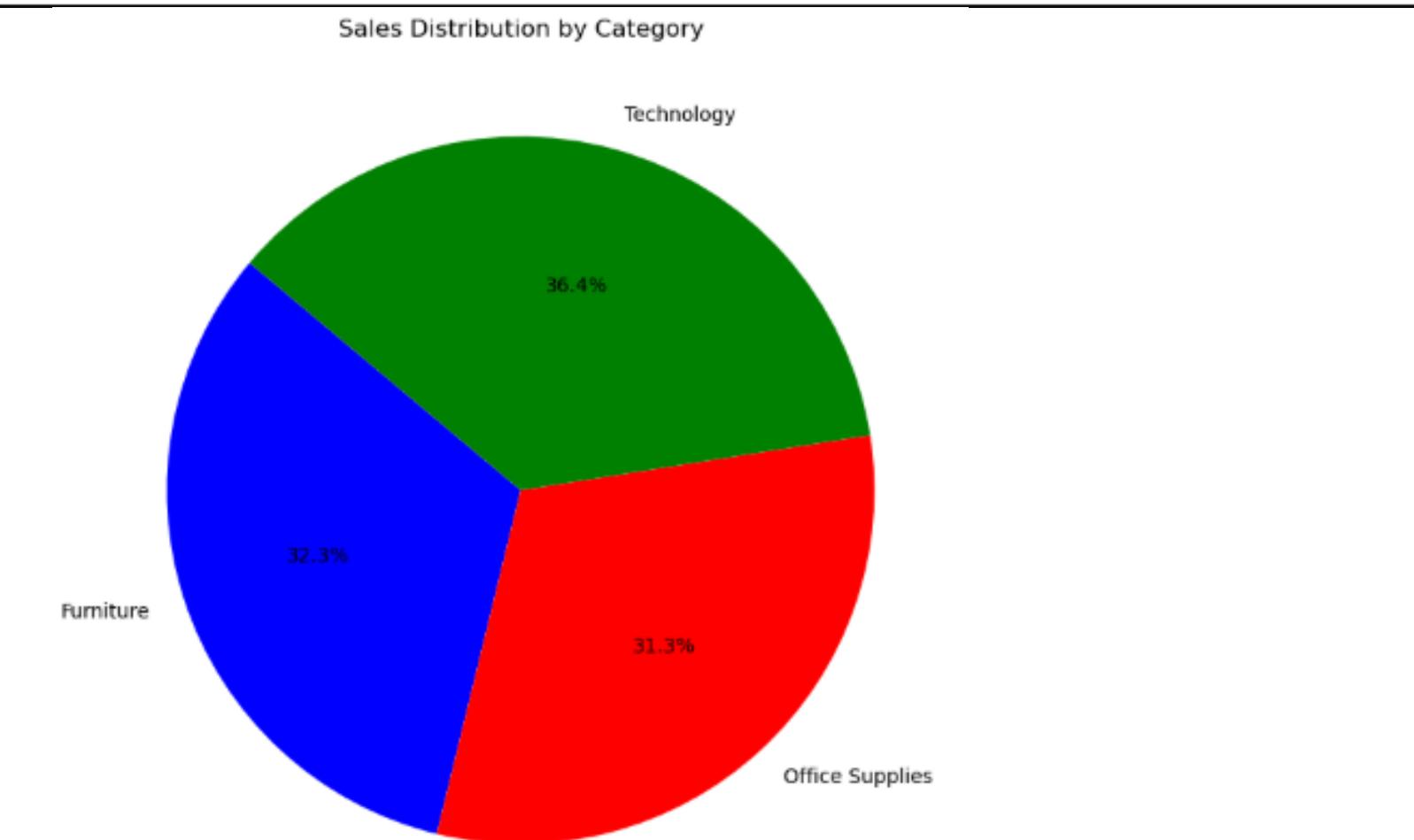


Category wise sales analysis

```
SalesByCategory = df.groupby("Category").agg({"Sales":"sum"})
SalesByCategory
```

```
plt.figure(figsize=(8, 8))
plt.pie(SalesByCategory['Sales'], labels=SalesByCategory.index, autopct='%1.1f%%', startangle=140, colors=['blue', 'red', 'green'])

plt.title("Sales Distribution by Category")
plt.show()
```

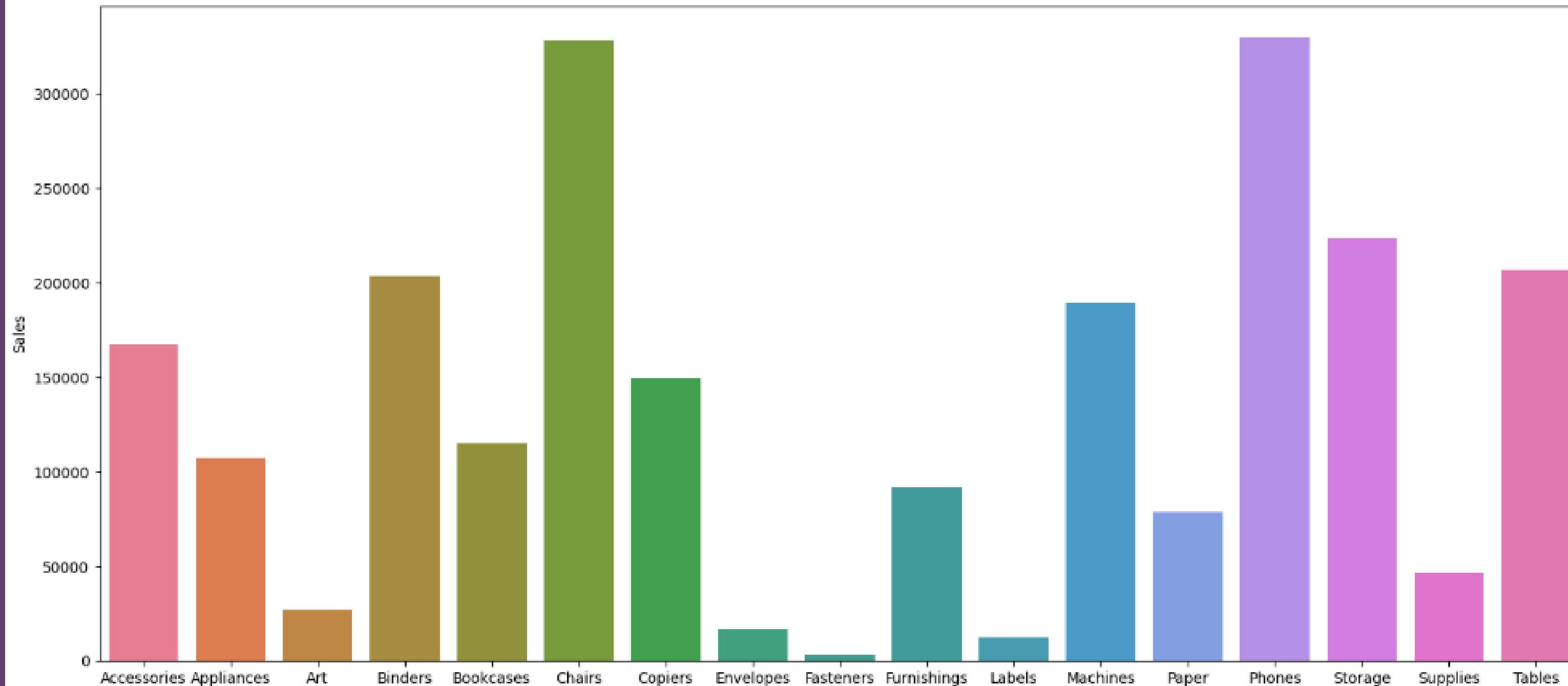


sales analysis by sub category

06

```
salesBySubcategory = df.groupby("Sub-Category").agg({"Sales":"sum"})  
salesBySubcategory
```

```
plt.figure(figsize=(18, 8))  
sns.barplot(data =salesBySubcategory,x ="Sub-Category",y="Sales",hue = "Sub-Category")  
plt.show()
```

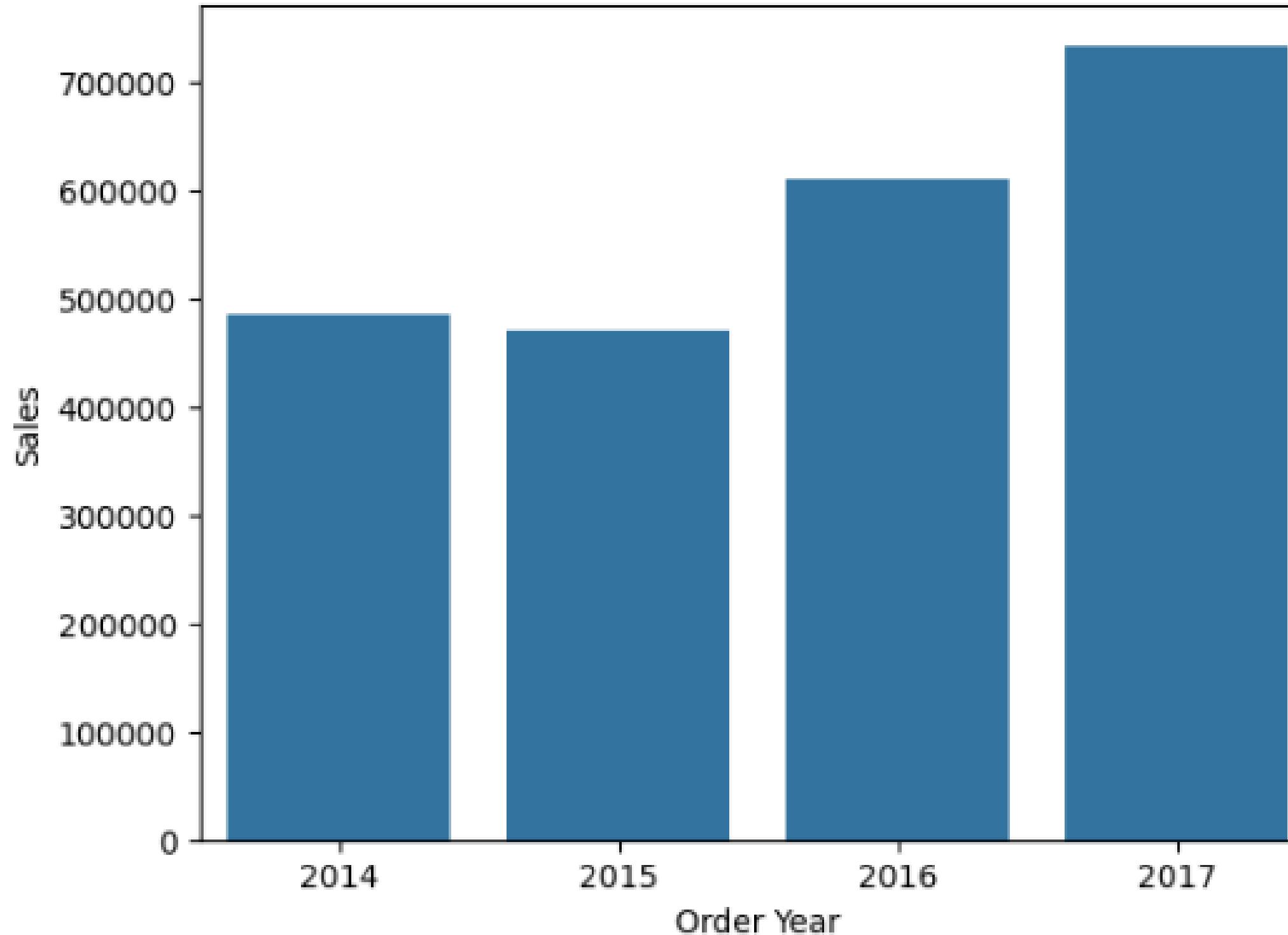


compare yearly sales

07

```
yearly_sales = df.groupby("Order Year").agg({"Sales":"sum"})  
yearly_sales
```

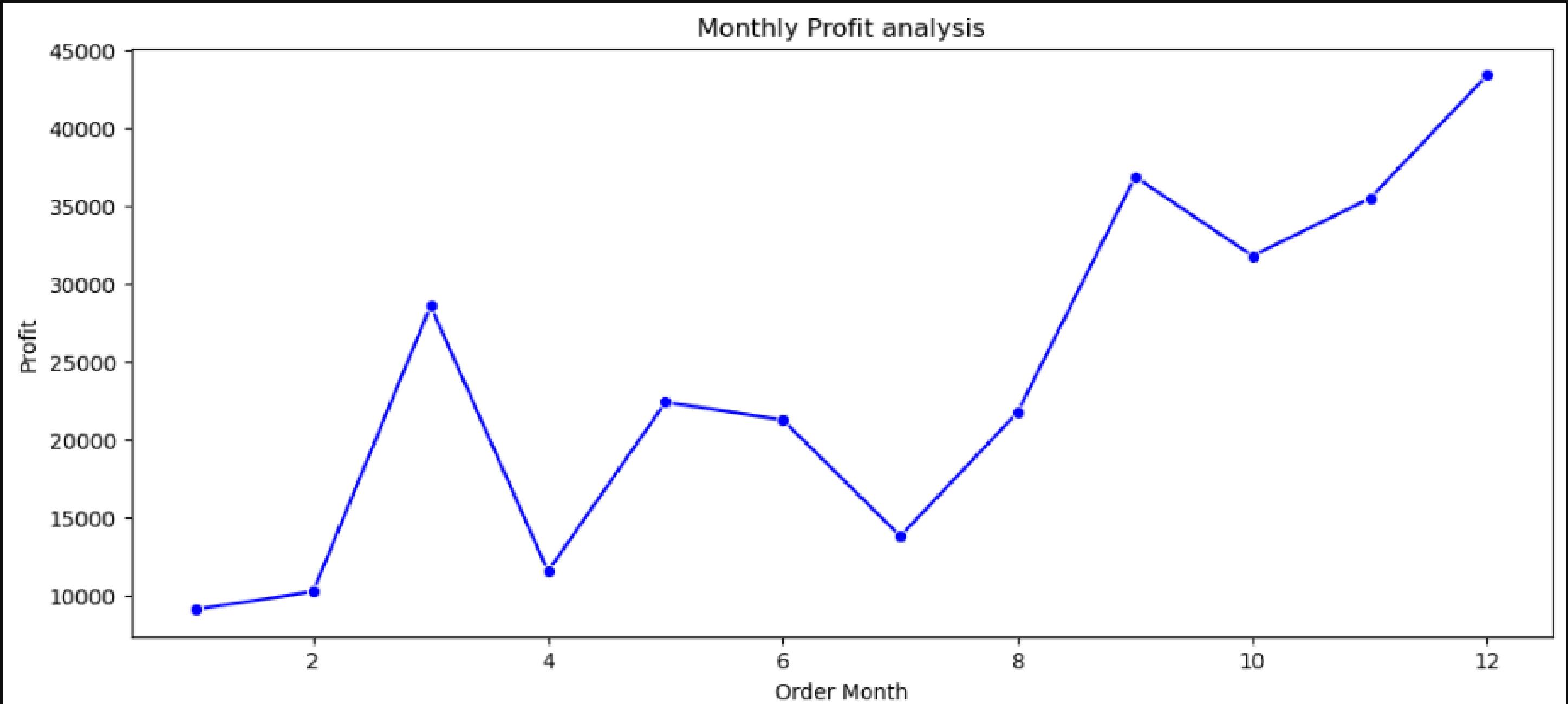
```
sns.barplot(data = yearly_sales,x = "Order Year",y= "Sales")  
plt.show()
```



monthly_profit

```
monthly_profit = df.groupby("Order Month").agg({"Profit": "sum"})
monthly_profit
```

```
plt.figure(figsize=(12, 5))
sns.lineplot(data=monthly_profit, x = 'Order Month', y = 'Profit', marker='o', color='blue')
plt.title("Monthly Profit analysis")
plt.show()
```

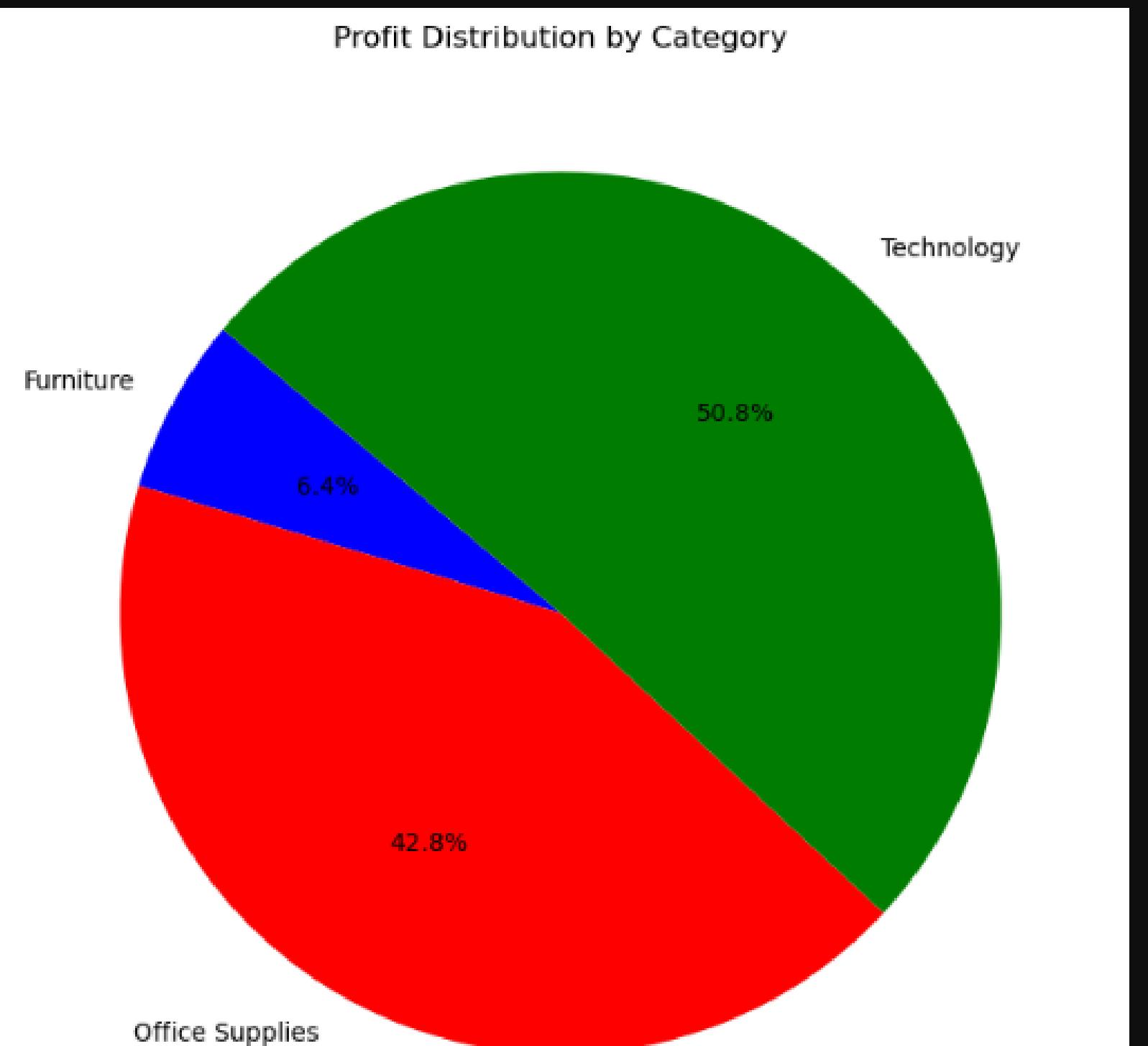


CategoryWise profit

09

```
categoryBy_profit = df.groupby("Category").agg({"Profit": "sum"})
categoryBy_profit
```

```
plt.figure(figsize=(8, 8))
plt.pie(categoryBy_profit['Profit'], labels=categoryBy_profit.index, autopct='%1.1f%%', startangle=140, colors=['blue', 'red', 'green'])
plt.title("Profit Distribution by Category")
plt.show()
```

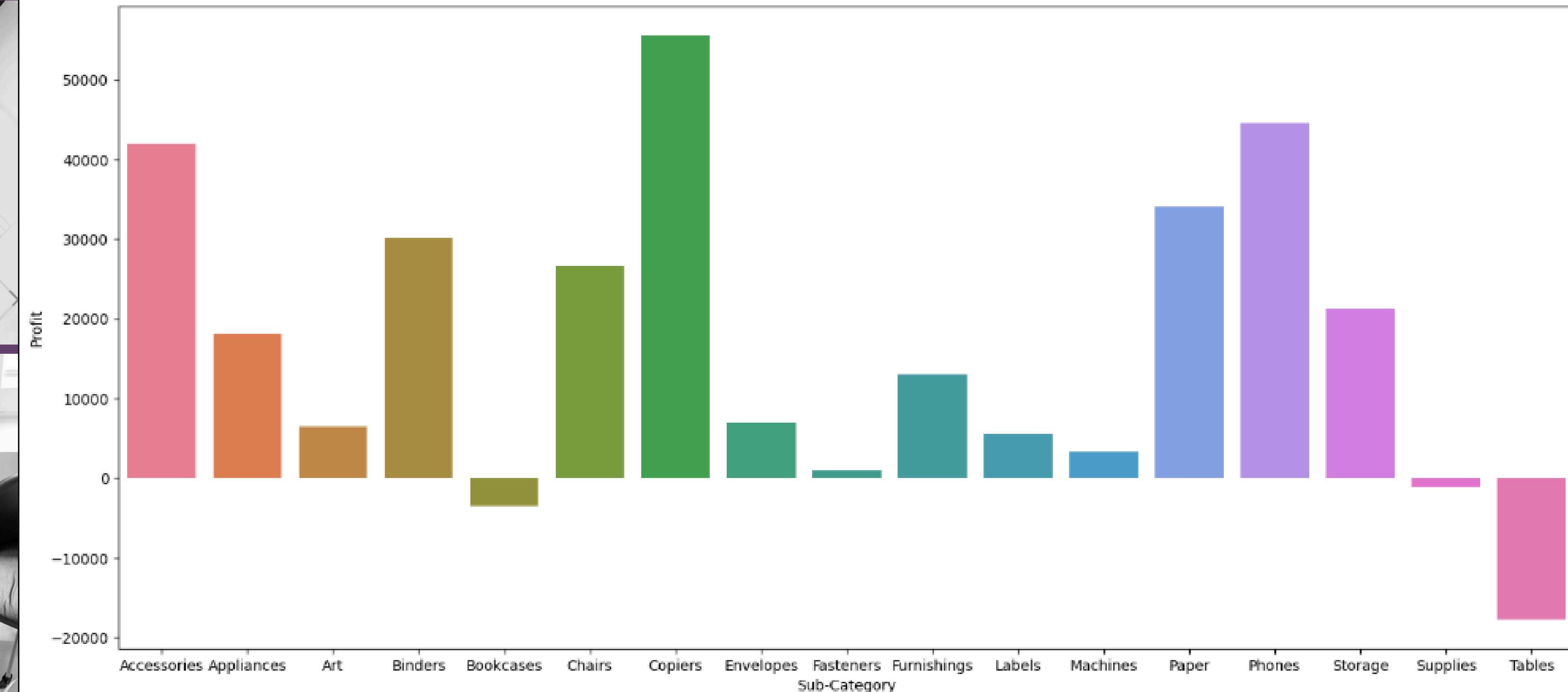


profit by Sub-category

10

```
SubcategoryBy_profit = df.groupby("Sub-Category").agg({"Profit":"sum"})  
SubcategoryBy_profit
```

```
plt.figure(figsize=(18, 8))  
sns.barplot(data =SubcategoryBy_profit,x ="Sub-Category",y="Profit",hue = "Sub-Category")  
plt.show()
```



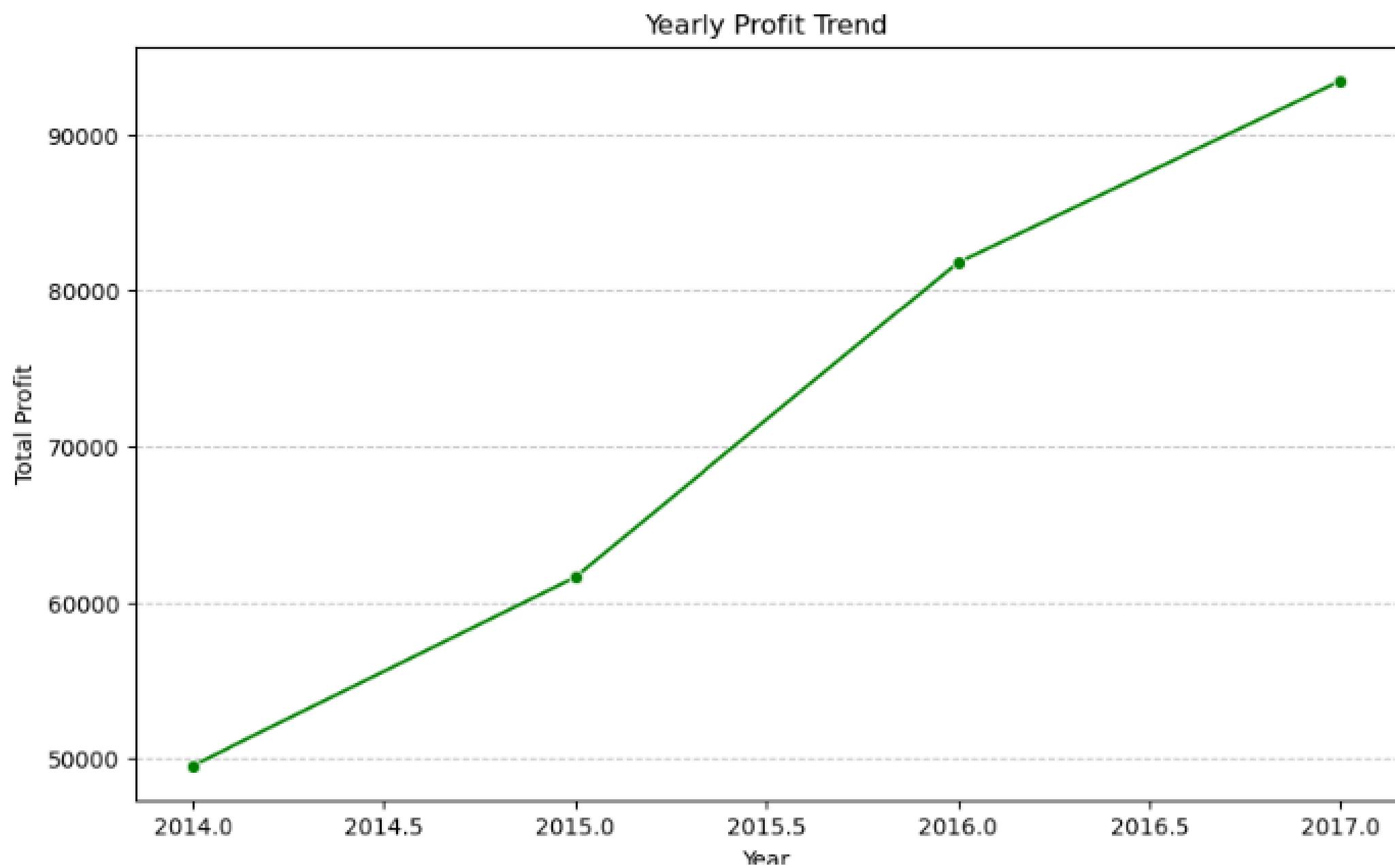
compare_yearly_profit

11

```
yearly_profit = df.groupby("Order Year").agg({"Profit":"sum"})
```

```
yearly_profit
```

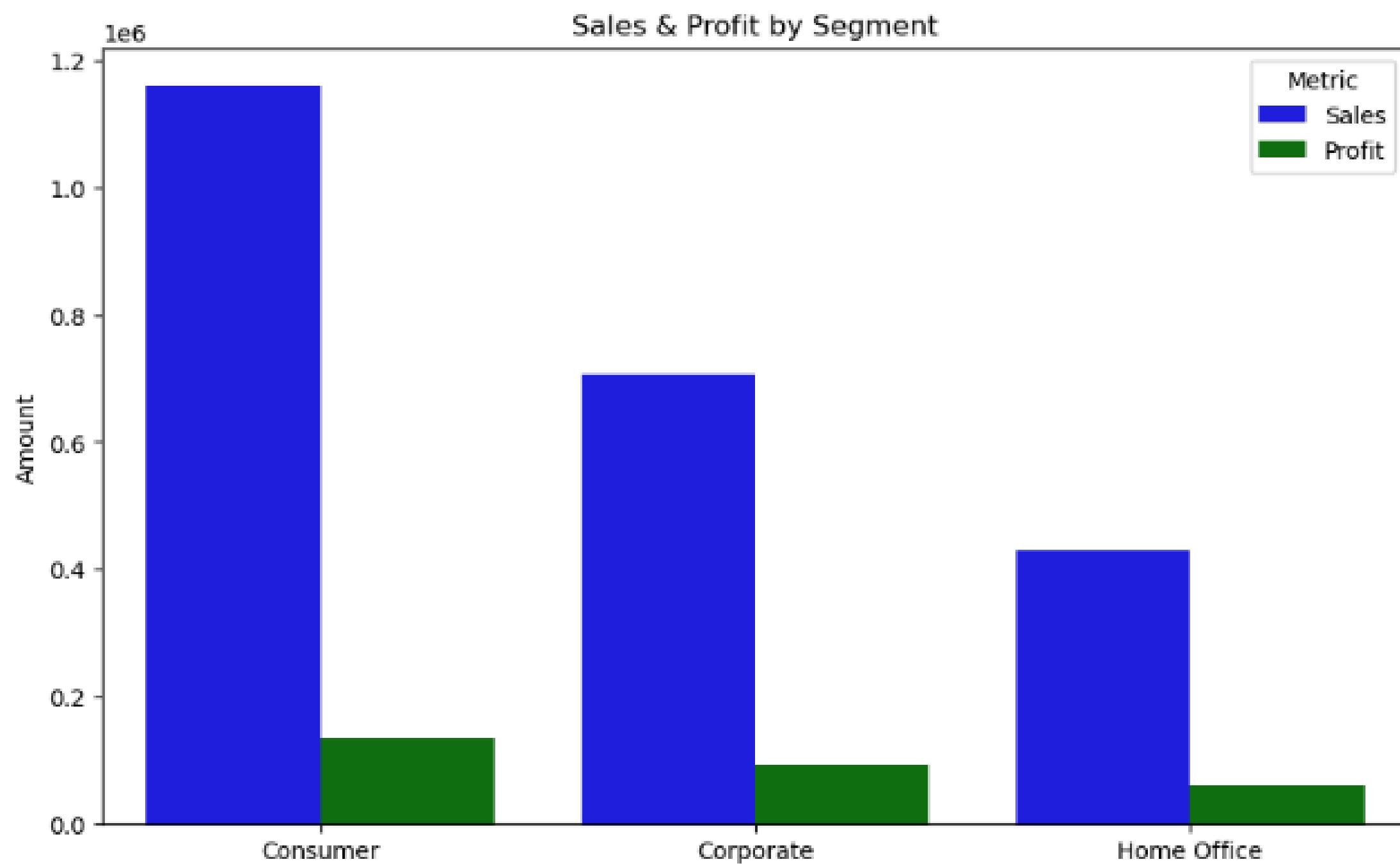
```
plt.figure(figsize=(10, 6))
sns.lineplot(x="Order Year", y="Profit", data=yearly_profit, marker="o", color="green")
plt.xlabel("Year")
plt.ylabel("Total Profit")
plt.title("Yearly Profit Trend")
plt.grid(axis="y", linestyle="--", alpha=0.7)
plt.show()
```



Analyze the sales and profit by customer segment

12

```
df_melted = sales_profit_by_segment.melt(id_vars="Segment", var_name="Metric", value_name="Amount")
plt.figure(figsize=(10, 6))
sns.barplot(x="Segment", y="Amount", hue="Metric", data=df_melted, palette=["blue", "green"])
plt.xlabel("Segment")
plt.ylabel("Amount")
plt.title("Sales & Profit by Segment")
plt.legend(title="Metric")
plt.show()
```



```
sales_profit_by_segment= df.groupby("Segment").agg({"Sales": "sum", "Profit": "sum"})
sales_profit_by_segment
```

Segment	Sales	Profit
0 Consumer	1.161401e+06	134119.2092
1 Corporate	7.061464e+05	91979.1340
2 Home Office	4.296531e+05	60298.6785



Analyze the sales to profit ratio

13

```
sales_profit_by_segment= df.groupby("Segment").agg({"Sales":"sum","Profit":"sum"}).reset_index()
sales_profit_by_segment["Sales Profit ratio"] = sales_profit_by_segment['Sales'] / sales_profit_by_segment['Profit']
sales_profit_by_segment[ ["Sales Profit ratio","Segment"]]
```

	Sales Profit ratio	Segment
0	8.659471	Consumer
1	7.677245	Corporate
2	7.125416	Home Office



Conclusion: Key Insights from E-Commerce Sales Analysis

14

Our analysis of the E-Commerce sales data reveals several significant trends that can help businesses optimize their strategies for maximizing revenue and profitability.

Monthly & Yearly Trends

- **December** consistently records the highest sales and profitability, likely due to holiday shopping and year-end promotions.
- **2017** stands out as the most successful year in both sales and profit, indicating strong market demand and business performance.

Category-Wise Sales & Profitability

- The best-selling category is **Technology (Phones)**, followed by Furniture (Chairs), and Office Supplies (Storage) in last place.
- In terms of profitability, **Technology (Phones)** remains the top contributor, followed by Office Supplies (Copiers), while Chairs rank the lowest.

Customer Segment Insights

- **Consumer Segment** drives the highest sales (\$1.16M) and profit (\$134K), making it the most valuable target group.
- Corporate Segment follows with \$706K in sales and \$91K in profit, indicating a stable B2B market.
- Home Office Segment ranks last, generating \$429K in sales and \$60K in profit, suggesting a niche but profitable audience.