```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
plt.style.use('seaborn-v0_8-darkgrid')
df = pd.read_csv(r"C:\Users\dell\Downloads\superstore.csv")
df

      Row ID        Order ID  Order Date   Ship Date       Ship
Mode  \
0          1  CA-2013-152156   11/9/2013  11/12/2013    Second Class

1          2  CA-2013-152156   11/9/2013  11/12/2013    Second Class

2          3  CA-2013-138688   6/13/2013   6/17/2013    Second Class

3          4  US-2012-108966  10/11/2012  10/18/2012  Standard Class

4          5  US-2012-108966  10/11/2012  10/18/2012  Standard Class

...      ...             ...         ...         ...             ...

9989    9990  CA-2011-110422   1/22/2011   1/24/2011    Second Class

9990    9991  CA-2014-121258   2/27/2014    3/4/2014  Standard Class

9991    9992  CA-2014-121258   2/27/2014    3/4/2014  Standard Class

9992    9993  CA-2014-121258   2/27/2014    3/4/2014  Standard Class

9993    9994  CA-2014-119914    5/5/2014   5/10/2014    Second Class


     Customer ID     Customer Name    Segment        Country
City  \
0       CG-12520       Claire Gute   Consumer  United States
Henderson
1       CG-12520       Claire Gute   Consumer  United States
Henderson
2       DV-13045   Darrin Van Huff  Corporate  United States      Los
Angeles
3       SO-20335    Sean O'Donnell   Consumer  United States  Fort
Lauderdale
4       SO-20335    Sean O'Donnell   Consumer  United States  Fort
Lauderdale
...          ...               ...        ...            ...
...
9989    TB-21400  Tom Boeckenhauer   Consumer  United States
Miami
9990    DB-13060       Dave Brooks   Consumer  United States
Costa Mesa
```

```
9991    DB-13060       Dave Brooks   Consumer  United States
Costa Mesa
9992    DB-13060       Dave Brooks   Consumer  United States
Costa Mesa
9993    CC-12220      Chris Cortes   Consumer  United States
Westminster

      ... Postal Code  Region       Product ID         Category Sub-
Category  \
0     ...       42420   South  FUR-BO-10001798        Furniture
Bookcases
1     ...       42420   South  FUR-CH-10000454        Furniture
Chairs
2     ...       90036    West  OFF-LA-10000240  Office Supplies
Labels
3     ...       33311   South  FUR-TA-10000577        Furniture
Tables
4     ...       33311   South  OFF-ST-10000760  Office Supplies
Storage
...   ...         ...     ...              ...              ...
...
9989  ...       33180   South  FUR-FU-10001889        Furniture
Furnishings
9990  ...       92627    West  FUR-FU-10000747        Furniture
Furnishings
9991  ...       92627    West  TEC-PH-10003645       Technology
Phones
9992  ...       92627    West  OFF-PA-10004041  Office Supplies
Paper
9993  ...       92683    West  OFF-AP-10002684  Office Supplies
Appliances

                                            Product Name     Sales
Quantity  \
0                      Bush Somerset Collection Bookcase  261.9600
2
1     Hon Deluxe Fabric Upholstered Stacking Chairs,...  731.9400
3
2     Self-Adhesive Address Labels for Typewriters b...   14.6200
2
3         Bretford CR4500 Series Slim Rectangular Table  957.5775
5
4                         Eldon Fold 'N Roll Cart System   22.3680
2
...                                                  ...       ...
...
9989                            Ultra Door Pull Handle   25.2480
3
9990  Tenex B1-RE Series Chair Mats for Low Pile Car...   91.9600
```

```
2
9991                                 Aastra 57i VoIP phone  258.5760
2
9992  It's Hot Message Books with Stickers, 2 3/4" x 5"   29.6000
4
9993  Acco 7-Outlet Masterpiece Power Center, Wihtou...  243.1600
2

      Discount    Profit
0         0.00   41.9136
1         0.00  219.5820
2         0.00    6.8714
3         0.45 -383.0310
4         0.20    2.5164
...        ...       ...
9989      0.20    4.1028
9990      0.00   15.6332
9991      0.20   19.3932
9992      0.00   13.3200
9993      0.00   72.9480

[9994 rows x 21 columns]
```

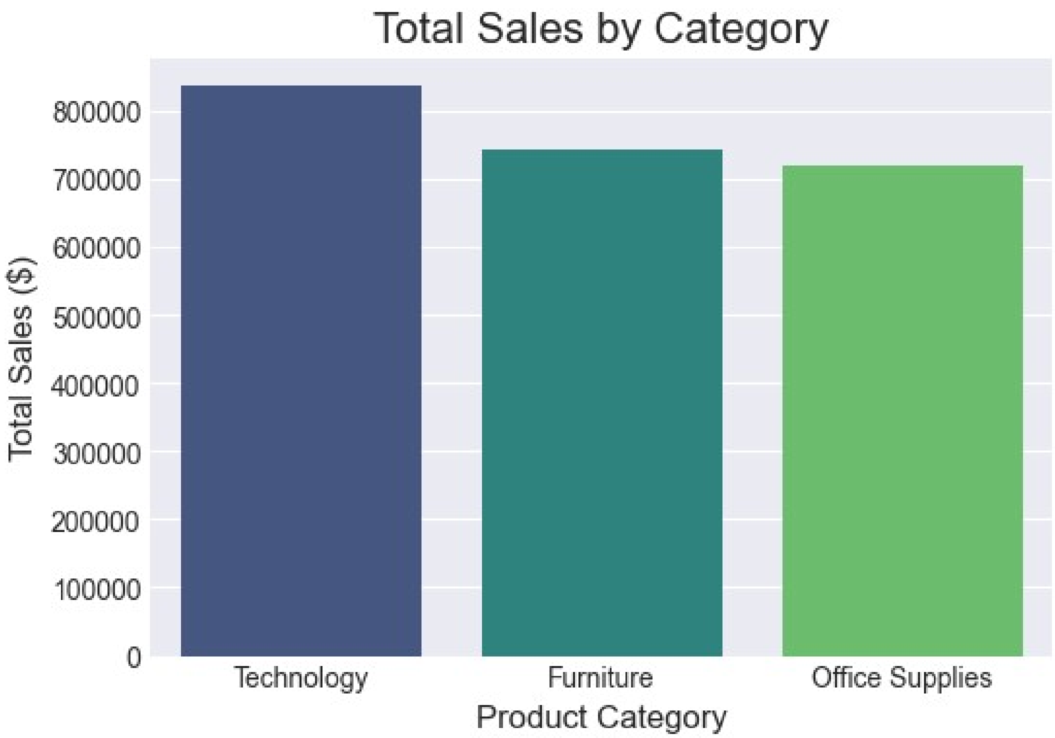
```python
def plot_sales_by_category():
    """Shows sales performance broken down by product category."""
    # Renaming columns and cleaning dates for this specific plot
    df.columns = df.columns.str.replace(' ', '_').str.replace('-',
'_')
    df['Order_Date'] = pd.to_datetime(df['Order_Date'], unit='D',
origin='1899-12-30')
plt.figure(figsize=(6, 4))
category_sales = df.groupby('Category')
['Sales'].sum().sort_values(ascending=False)
 # Using a simple bar plot to compare sales across categories.
category_sales = df.groupby('Category')
['Sales'].sum().sort_values(ascending=False)
sns.barplot(x=category_sales.index, y=category_sales.values,
palette='viridis')
plt.title('Total Sales by Category', fontsize=16)
plt.xlabel('Product Category', fontsize=12)
plt.ylabel('Total Sales ($)', fontsize=12)
plt.show()
```

```
C:\Users\dell\AppData\Local\Temp\ipykernel_14732\2324574620.py:10:
FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be
removed in v0.14.0. Assign the `x` variable to `hue` and set
```

```
`legend=False` for the same effect.

  sns.barplot(x=category_sales.index, y=category_sales.values,
palette='viridis')
```

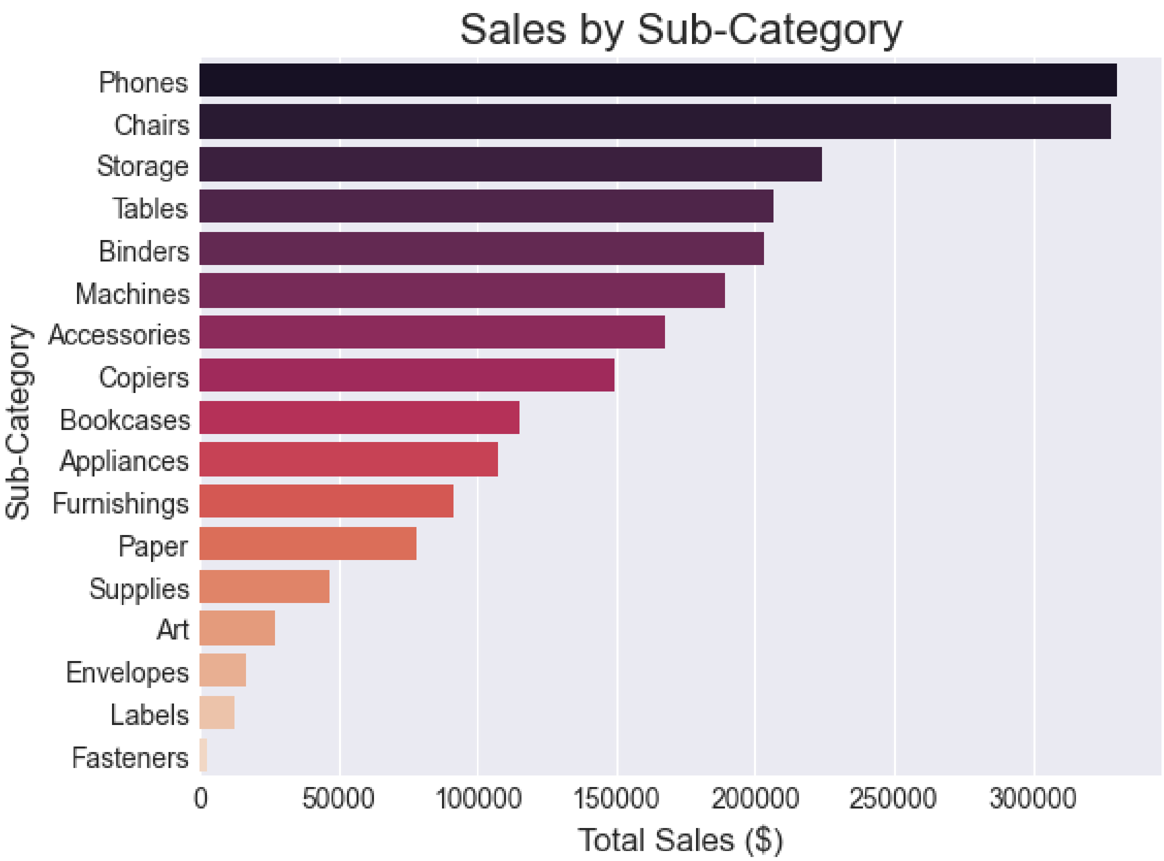


```
subcategory_sales = df.groupby('Sub-Category')
['Sales'].sum().sort_values(ascending=False)
sns.barplot(y=subcategory_sales.index, x=subcategory_sales.values,
palette='rocket')
plt.title('Sales by Sub-Category', fontsize=16)
plt.xlabel('Total Sales ($)', fontsize=12)
plt.ylabel('Sub-Category', fontsize=12)
plt.figure(figsize=(8, 6))
plt.tight_layout()
plt.show()

C:\Users\dell\AppData\Local\Temp\ipykernel_14732\275919304.py:2:
FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be
removed in v0.14.0. Assign the `y` variable to `hue` and set
`legend=False` for the same effect.

  sns.barplot(y=subcategory_sales.index, x=subcategory_sales.values,
palette='rocket')
```



```
<Figure size 800x600 with 0 Axes>
```

```
def plot_monthly_profit(df):

    df['order_month'] = df['order_date'].dt.to_period('M')
    monthly_profit = df.groupby('order_month')
['profit'].sum().reset_index()
    monthly_profit['order_month'] =
monthly_profit['order_month'].dt.to_timestamp()

    plt.figure(figsize=(16, 8))
    sns.lineplot(data= monthly_profit, x='order_month', y='profit',
marker='o')
    plt.title('Monthly Profit Over Time', fontsize=16)
    plt.xlabel('Date', fontsize=12)
    plt.ylabel('Total Profit ($)', fontsize=12)
    plt.xticks(rotation=45)
    plt.show()

region_summary = df.groupby('Region')[['Sales',
'Profit']].sum().reset_index()

    # We can use subplots to put both charts side-by-side.
```

```python
fig, axes = plt.subplots(1, 2, figsize=(16, 6))

    # Left plot: Sales
sns.barplot(ax=axes[0], data=region_summary, x='Region', y='Sales', 
palette='Blues_d')
axes[0].set_title('Sales by Region', fontsize=14)

    # Right plot: Profit
sns.barplot(ax=axes[1], data=region_summary, x='Region', y='Profit', 
palette='Greens_d')
axes[1].set_title('Profit by Region', fontsize=14)
plt.tight_layout()
plt.show()
```

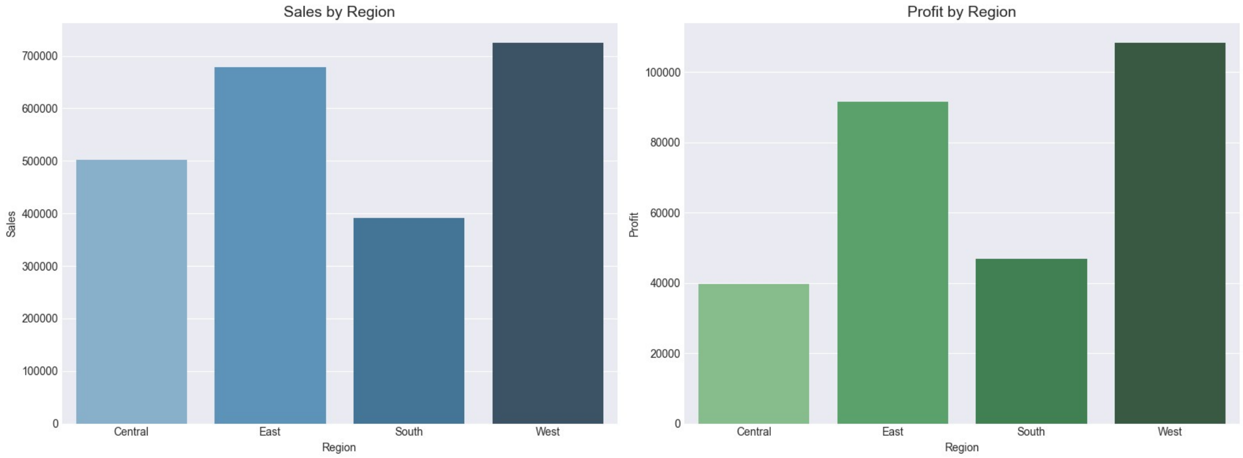
```
C:\Users\dell\AppData\Local\Temp\ipykernel_14716\3876426203.py:7: 
FutureWarning: 

Passing `palette` without assigning `hue` is deprecated and will be 
removed in v0.14.0. Assign the `x` variable to `hue` and set 
`legend=False` for the same effect.

  sns.barplot(ax=axes[0], data=region_summary, x='Region', y='Sales', 
palette='Blues_d')
C:\Users\dell\AppData\Local\Temp\ipykernel_14716\3876426203.py:11: 
FutureWarning: 

Passing `palette` without assigning `hue` is deprecated and will be 
removed in v0.14.0. Assign the `x` variable to `hue` and set 
`legend=False` for the same effect.

  sns.barplot(ax=axes[1], data=region_summary, x='Region', y='Profit', 
palette='Greens_d')
```


Sales by Region
Sales
Central
East
South
West
Region
Profit by Region
Profit
Central
East
South
West
Region