```
pip install pandas matplotlib seaborn
```

```
Requirement already satisfied: pandas in /usr/local/lib/python3.11/dist-packages (2.2.2)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.11/dist-packages (3.10.0)
Requirement already satisfied: seaborn in /usr/local/lib/python3.11/dist-packages (0.13.2)
Requirement already satisfied: numpy>=1.23.2 in /usr/local/lib/python3.11/dist-packages (from pandas) (2.0.2)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.11/dist-packages (from pandas) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-packages (from pandas) (2025.2)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (1.3.2)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (4.58.0)
Requirement already satisfied: kiwisolver>=1.3.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (1.4.8)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (24.2)
Requirement already satisfied: pillow>=8 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (11.2.1)
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (3.2.3)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-packages (from python-dateutil>=2.8.2->pandas) (1.17.0)
```

```
from google.colab import files
uploaded = files.upload()
```

Choose Files archive.zip

- **archive.zip**(application/x-zip-compressed) - 562846 bytes, last modified: 5/17/2025 - 100% done

```
Saving archive.zip to archive.zip
```

```
import zipfile

with zipfile.ZipFile('archive.zip', 'r') as zip_ref:
    zip_ref.extractall()  # Extracts to the current working directory

# Check what files were extracted
import os
os.listdir()
```

```
['.config', 'Sample - Superstore.csv', 'archive.zip', 'sample_data']
```

```
import os

os.listdir()
```

```
['.config', 'Sample - Superstore.csv', 'archive.zip', 'sample_data']
```

```
import pandas as pd

df = pd.read_csv('Sample - Superstore.csv', encoding='latin1')
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
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
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

```
# Summary info
df.info()

# Summary statistics
df.describe()
```

```
# Check first 5 rows
df.head()
```

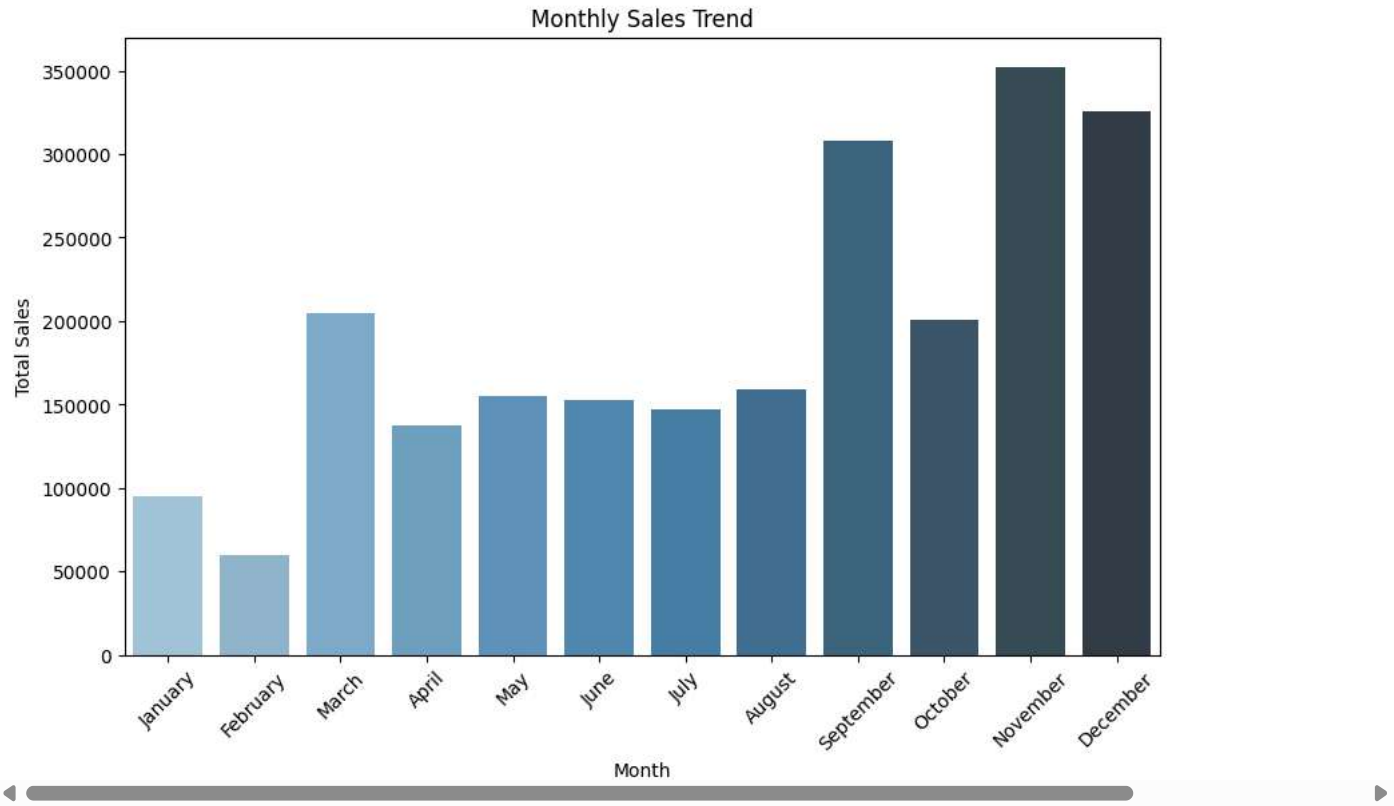
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9994 entries, 0 to 9993
Data columns (total 21 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Row ID         9994 non-null   int64
 1   Order ID       9994 non-null   object
 2   Order Date     9994 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
```

	Row ID	Order ID	Order Date	Ship Date	Ship Mode	Customer ID	Customer Name	Segment	Country	City	...	Postal Code	Region	Product ID	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
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

```
df['Order Date'] = pd.to_datetime(df['Order Date'], errors='coerce')
df['Month'] = df['Order Date'].dt.month_name()
df['Year'] = df['Order Date'].dt.year
```

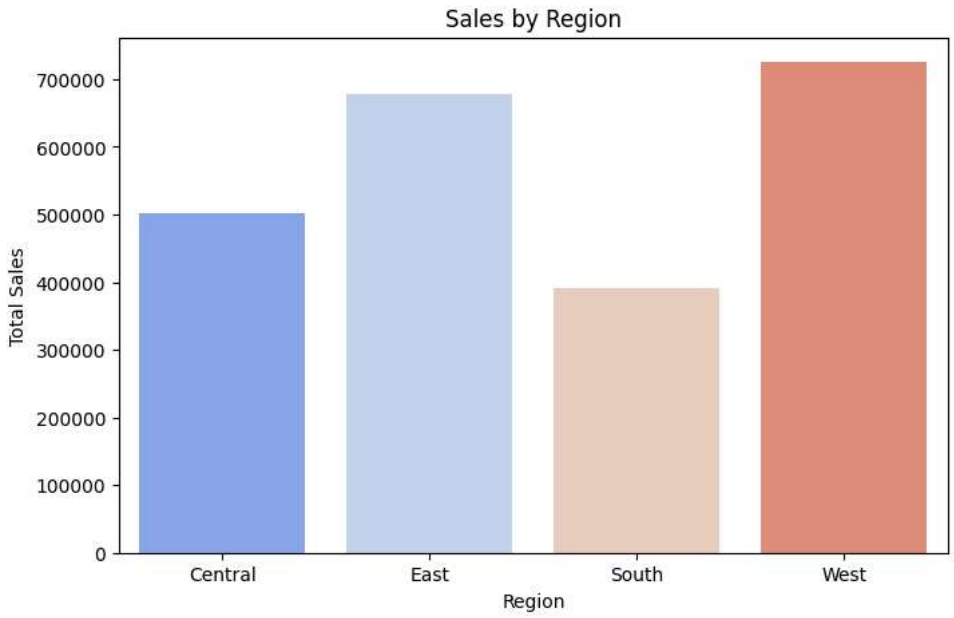
```
import matplotlib.pyplot as plt
import seaborn as sns

# Order months for proper plotting
months_order = ['January', 'February', 'March', 'April', 'May', 'June',
                'July', 'August', 'September', 'October', 'November', 'December']

monthly_sales = df.groupby('Month')['Sales'].sum().reindex(months_order)

plt.figure(figsize=(10,6))
sns.barplot(x=monthly_sales.index, y=monthly_sales.values, palette='Blues_d')
plt.xticks(rotation=45)
plt.title('Monthly Sales Trend')
plt.ylabel('Total Sales')
plt.xlabel('Month')
plt.show()
```

```
<ipython-input-12-30f46dd8a0ec>: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

  sns.barplot(x=monthly_sales.index, y=monthly_sales.values, palette='Blues_d')
```

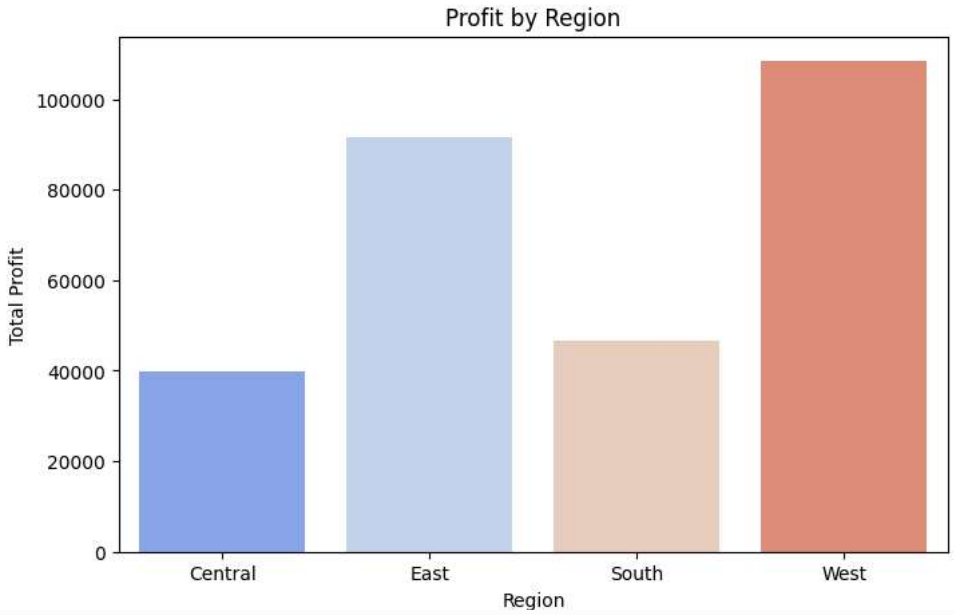



```
region_summary = df.groupby('Region')[['Sales', 'Profit']].sum().reset_index()

plt.figure(figsize=(8,5))
sns.barplot(x='Region', y='Sales', data=region_summary, palette='coolwarm')
plt.title('Sales by Region')
plt.ylabel('Total Sales')
plt.xlabel('Region')
plt.show()

plt.figure(figsize=(8,5))
sns.barplot(x='Region', y='Profit', data=region_summary, palette='coolwarm')
plt.title('Profit by Region')
plt.ylabel('Total Profit')
plt.xlabel('Region')
plt.show()
```

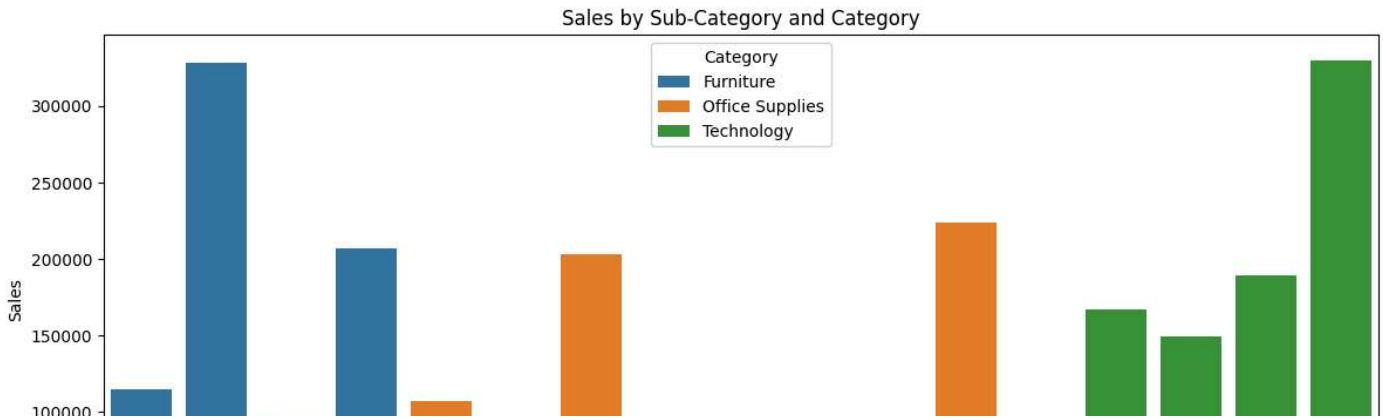
```
<ipython-input-13-3757b56c7592>:4: FutureWarning:

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

  sns.barplot(x='Region', y='Sales', data=region_summary, palette='coolwarm')
```




```
<ipython-input-13-3757b56c7592>: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 `leg

  sns.barplot(x='Region', y='Profit', data=region_summary, palette='coolwarm')
```




```
category_summary = df.groupby(['Category', 'Sub-Category'])[['Sales', 'Profit']].sum().reset_index()

plt.figure(figsize=(12,6))
sns.barplot(x='Sub-Category', y='Sales', hue='Category', data=category_summary)
plt.xticks(rotation=45)
plt.title('Sales by Sub-Category and Category')
plt.tight_layout()
plt.show()
```


Sales by Sub-Category and Category
Category
Furniture
Office Supplies
Technology
Sales
300000
250000
200000
150000
100000