Setup & Install

```
!pip install pandas matplotlib
```

```
Requirement already satisfied: pandas in /usr/local/lib/python3.12/dis
Requirement already satisfied: matplotlib in /usr/local/lib/python3.12
Requirement already satisfied: numpy>=1.26.0 in /usr/local/lib/python3
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/li
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/pyth
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/pyt
Requirement already satisfied: kiwisolver>=1.3.1 in /usr/local/lib/pyt
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/pytho
Requirement already satisfied: pillow>=8 in /usr/local/lib/python3.12/
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/pyth
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.12/d
```

```
import pandas as pd
import matplotlib.pyplot as plt
```

#Show plots inline

```
%matplotlib inline
```

Upload CSV

```
from google.colab import files
uploaded = files.upload()  # Select sales_data.csv
df_raw = pd.read_csv("sales_data.csv")
df_raw.head()
```

Choose files sales_data.csv

sales_data.csv(text/csv) - 4798 bytes, last modified: 04/01/2026 - 100% done

```
Saving sales_data.csv to sales_data (1).csv
```

	Date	Product	Quantity	Price	Customer_ID	Region	Total_Sales
0	2024-01-01	Phone	7	37300	CUST001	East	261100
1	2024-01-02	Headphones	4	15406	CUST002	North	61624
2	2024-01-03	Phone	2	21746	CUST003	West	43492
3	2024-01-04	Headphones	1	30895	CUST004	East	30895
4	2024-01-05	Laptop	8	39835	CUST005	North	318680

Next steps: Generate code with df_raw New interactive sheet

Clean and **preprocess**

#Convert Date to **datetime**

```
df = df_raw.copy()
df["Date"] = pd.to_datetime(df["Date"], errors="coerce")
```

#Ensure numeric types

```
for col in ["Quantity", "Price", "Total_Sales"]:
    df[col] = pd.to_numeric(df[col], errors="coerce")
```

#Drop invalid rows

```
df = df.dropna(subset=["Date", "Quantity", "Price", "Total_Sales"])
```

#Recalculate Total_Sales and add Month

```
df["Total_Sales"] = df["Quantity"] * df["Price"]
df["Month"] = df["Date"].dt.to_period("M").astype(str)
df.head()
```

	Date	Product	Quantity	Price	Customer_ID	Region	Total_Sales
0	2024-01-01	Phone	7	37300	CUST001	East	261100
1	2024-01-02	Headphones	4	15406	CUST002	North	61624
2	2024-01-03	Phone	2	21746	CUST003	West	43492
3	2024-01-04	Headphones	1	30895	CUST004	East	30895
4	2024-01-05	Laptop	8	39835	CUST005	North	318680

Next steps: Generate code with df New interactive sheet

Basic analysis

```
sales_by_product = (
    df.groupby("Product")["Total_Sales"]
      .sum()
      .sort_values(ascending=False)
)

sales_by_region = (
    df.groupby("Region")["Total_Sales"]
      .sum()
      .sort_values(ascending=False)
)

monthly_sales = (
    df.groupby("Month")["Total_Sales"]
      .sum()
      .sort_values()
)

print("Sales by product:\n", sales_by_product, "\n")
print("Sales by region:\n", sales_by_region, "\n")
print("Monthly sales:\n", monthly_sales)
```

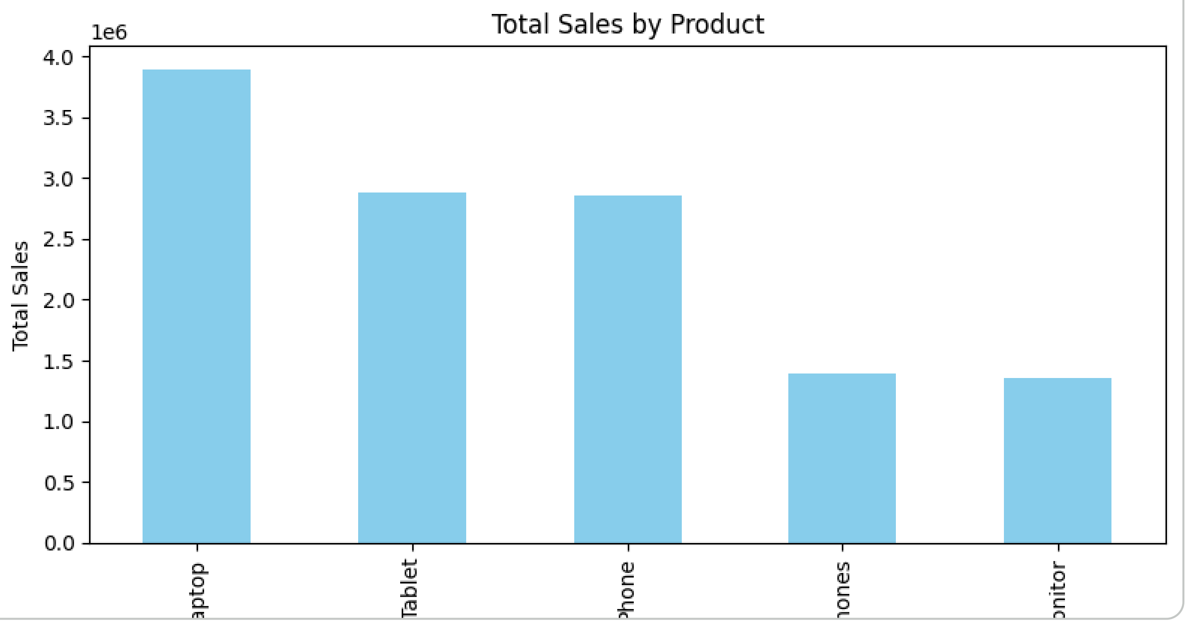
```
Sales by product:
 Product
Laptop        3889210
Tablet        2884340
Phone         2859394
Headphones    1384033
Monitor       1348071
Name: Total_Sales, dtype: int64
```

```
Sales by region:
 Region
North    3983635
South    3737852
East     2519639
West     2123922
Name: Total_Sales, dtype: int64

Monthly sales:
 Month
2024-04    1103468
2024-02    2656050
2024-01    4120524
2024-03    4485006
Name: Total_Sales, dtype: int64
```

Bar chart: sales by product

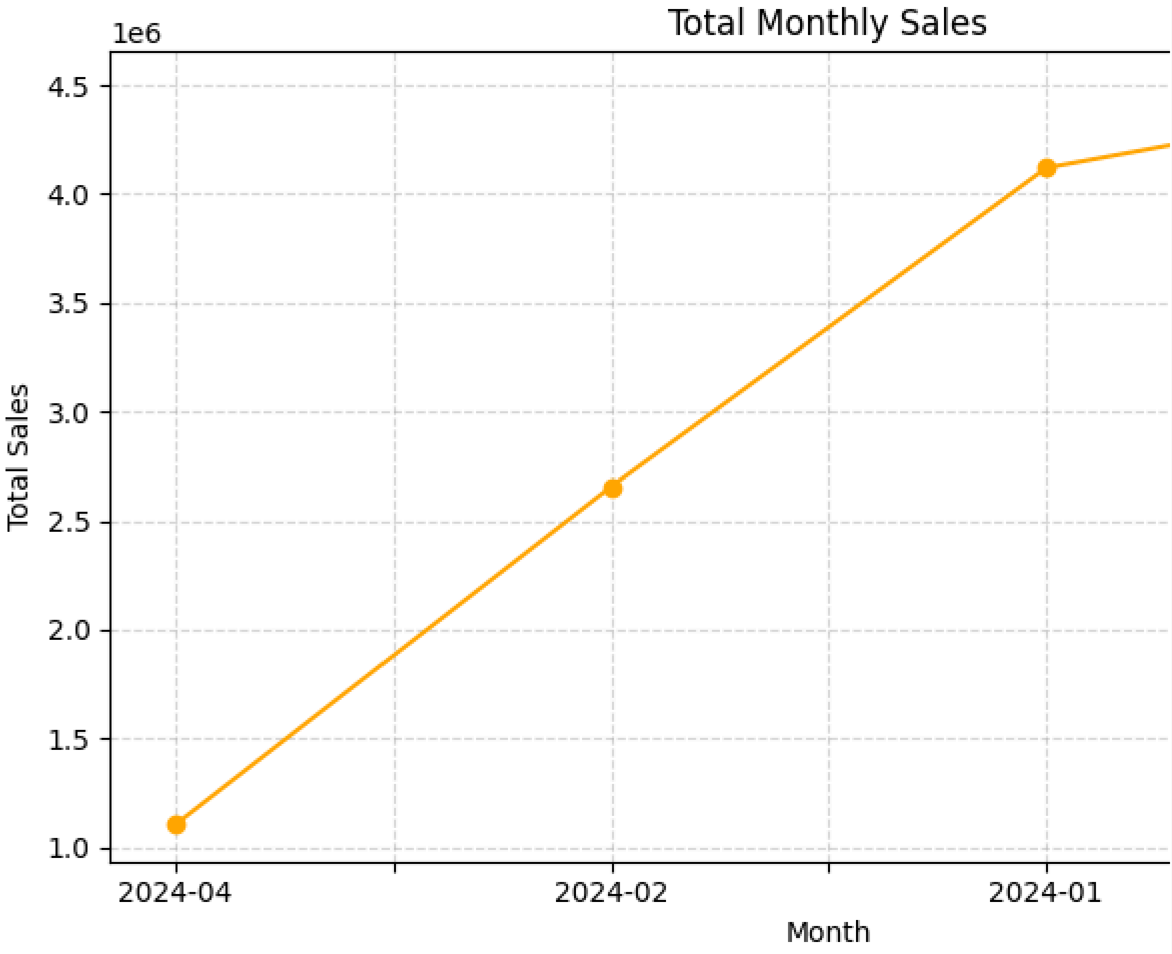
```
plt.figure(figsize=(8, 5))
sales_by_product.plot(kind="bar", color="skyblue")
plt.title("Total Sales by Product")
plt.xlabel("Product")
plt.ylabel("Total Sales")
plt.tight_layout()
plt.show()
```



Line chart: monthly sales

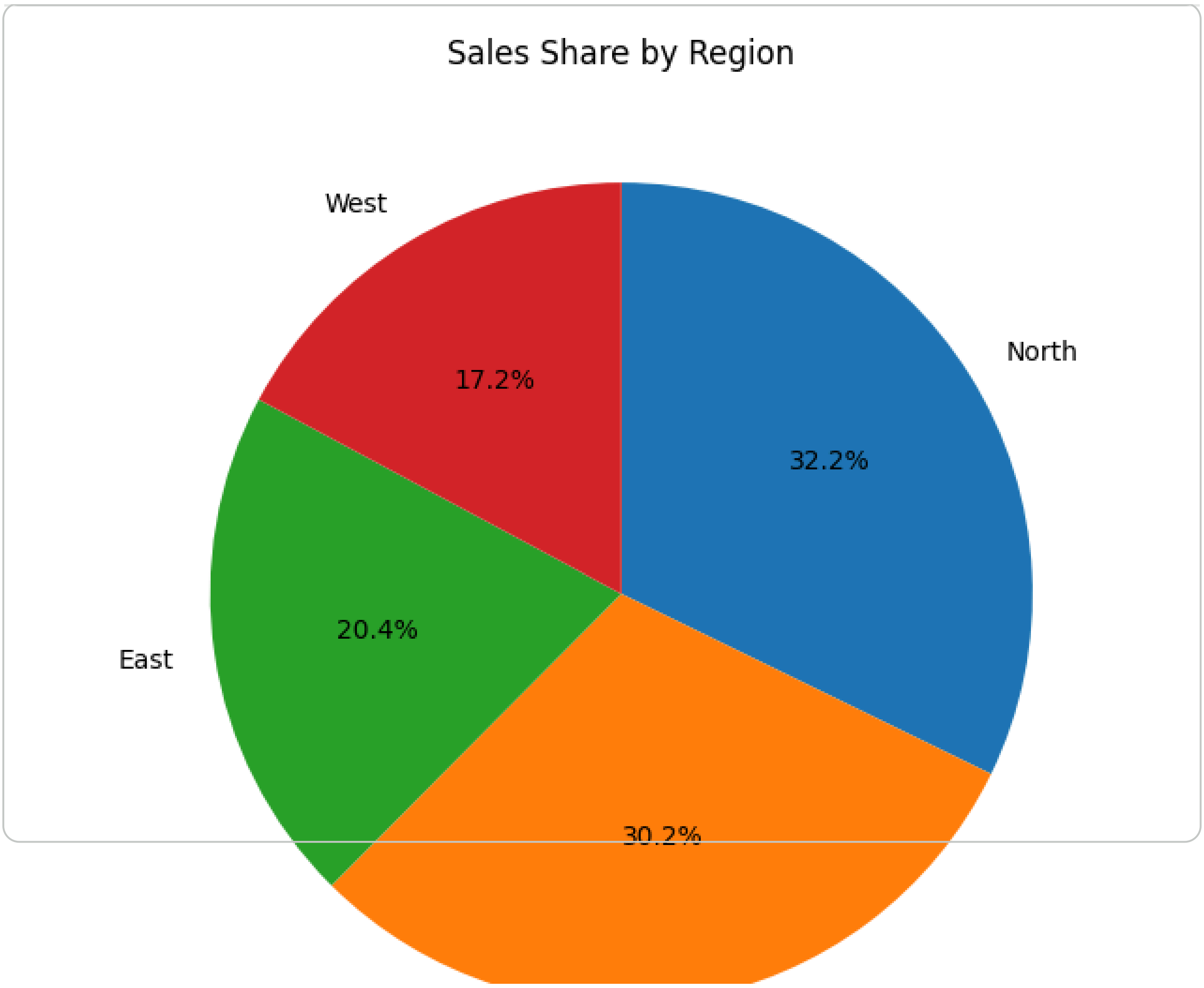
```
plt.figure(figsize=(8, 5))
monthly_sales.plot(kind="line", marker="o", color="orange")
plt.title("Total Monthly Sales")
plt.xlabel("Month")
```

```
    plt.ylabel("Total Sales")
    plt.grid(True, linestyle="--", alpha=0.5)
    plt.tight_layout()
    plt.show()
```



Pie chart: sales by region

```
    plt.figure(figsize=(6, 6))
    sales_by_region.plot(
        kind="pie",
        autopct="%1.1f%%",
        startangle=90,
        counterclock=False
    )
    plt.title("Sales Share by Region")
    plt.ylabel("")
    plt.tight_layout()
    plt.show()
```


Sales Share by Region
West
17.2%
North
32.2%
East
20.4%
30.2%