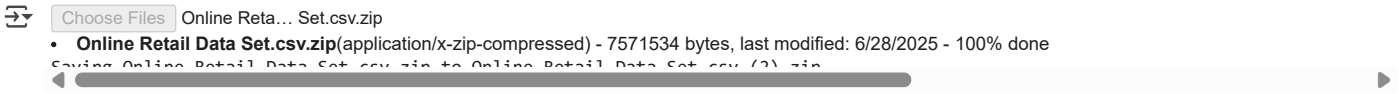
import libraries

```
import pandas as pd
```

upload your file which is dataset file here we used TATA's online retail data set file

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



Reading the file .before read ensure that you uploaded the dataset.

```
df = pd.read_csv("Online Retail Data Set.csv.zip", encoding='ISO-8859-1', compression='zip')
```

```
df = df.dropna(subset=['CustomerID', 'Description'])
```

```
df = df.dropna(subset=['CustomerID', 'Description'])
```

```
df['InvoiceDate'] = pd.to_datetime(df['InvoiceDate'], dayfirst=True)
```

```
df['TotalPrice'] = df['Quantity'] * df['UnitPrice']
```

```
df['Description'] = df['Description'].str.strip().str.lower()
```

```
df.to_csv("Online_Retail_Cleaned.csv", index=False)
```

```
files.download("Online_Retail_Cleaned.csv")
```

```
# Display the shape of the DataFrame
print("Shape of the DataFrame:", df.shape)

# Display data types of each column
print("\nData types:")
print(df.dtypes)

# Display descriptive statistics
print("\nDescriptive statistics:")
display(df.describe(include='all'))
```

```
Shape of the DataFrame: (406829, 9)

Data types:
InvoiceNo              object
StockCode              object
Description            object
Quantity                int64
InvoiceDate    datetime64[ns]
UnitPrice             float64
CustomerID            float64
Country                object
TotalPrice            float64
dtype: object

Descriptive statistics:
```

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country	TotalPrice
count	406829	406829	406829	406829.000000	406829	406829.000000	406829.000000	406829	406829.000000
unique	22190	3684	3885	NaN	NaN	NaN	NaN	37	NaN
top	576339	85123A	white hanging heart t-light holder	NaN	NaN	NaN	NaN	United Kingdom	NaN
freq	542	2077	2070	NaN	NaN	NaN	NaN	361878	NaN
mean	NaN	NaN	NaN	12.061303	2011-07-10 16:30:57.879207424	3.460471	15287.690570	NaN	20.401854
min	NaN	NaN	NaN	-80995.000000	2010-12-01 08:26:00	0.000000	12346.000000	NaN	-168469.600000
25%	NaN	NaN	NaN	2.000000	2011-04-06 15:02:00	1.250000	13953.000000	NaN	4.200000

EDA

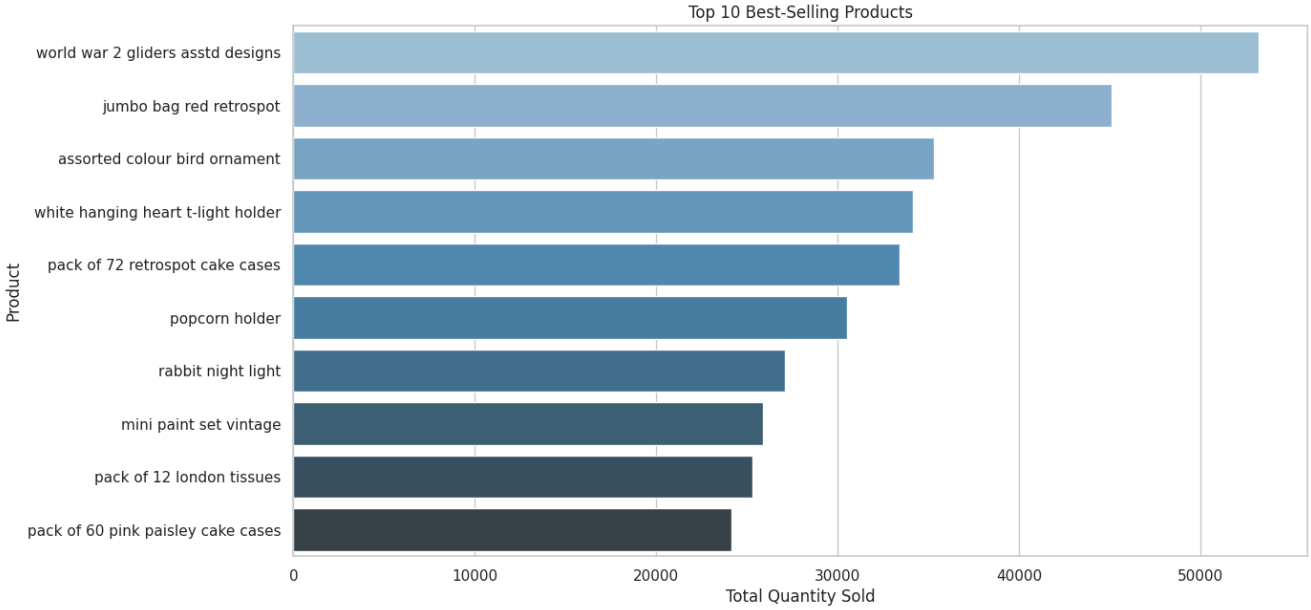
```
# 📦 STEP 1: Install necessary packages (if not already)
!pip install pandas matplotlib seaborn

# 📤 STEP 2: Upload the cleaned file
from google.colab import files
uploaded = files.upload()  # Upload 'Online_Retail_Cleaned.csv'

# 📄 STEP 3: Load the CSV
import pandas as pd
df = pd.read_csv("Online_Retail_Cleaned.csv")
df['InvoiceDate'] = pd.to_datetime(df['InvoiceDate'])  # ensure datetime format

# 📊 STEP 4: Import visualization libraries
import matplotlib.pyplot as plt
import seaborn as sns

sns.set(style='whitegrid')

# 1️⃣ Top-selling products
top_products = df.groupby('Description')['Quantity'].sum().sort_values(ascending=False).head(10)

# 2️⃣ Most active customers
top_customers = df['CustomerID'].value_counts().head(10)

# 3️⃣ Monthly sales trend
df['Month'] = df['InvoiceDate'].dt.to_period('M')
monthly_sales = df.groupby('Month')['TotalPrice'].sum()

# 📈 STEP 5: Plotting
fig, axs = plt.subplots(3, 1, figsize=(14, 20))

# Plot 1: Top Products
sns.barplot(x=top_products.values, y=top_products.index, ax=axs[0], palette="Blues_d")
axs[0].set_title('Top 10 Best-Selling Products')
axs[0].set_xlabel('Total Quantity Sold')
axs[0].set_ylabel('Product')

# Plot 2: Top Customers
sns.barplot(x=top_customers.index.astype(str), y=top_customers.values, ax=axs[1], palette="Greens_d")
axs[1].set_title('Top 10 Most Active Customers')
axs[1].set_xlabel('Customer ID')
axs[1].set_ylabel('Number of Transactions')

# Plot 3: Monthly Sales Trend
```

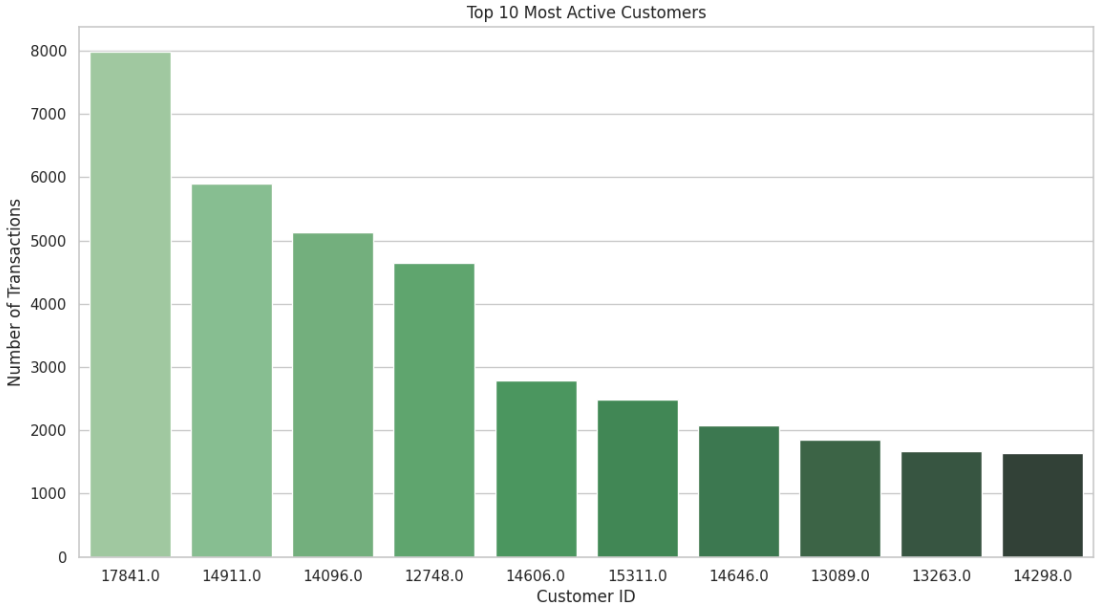
```
monthly_sales.index = monthly_sales.index.astype(str)
axs[2].plot(monthly_sales.index, monthly_sales.values, marker='o', linestyle='-', color='orange')
axs[2].set_title('Monthly Sales Trend')
axs[2].set_xlabel('Month')
axs[2].set_ylabel('Total Sales')
axs[2].tick_params(axis='x', rotation=45)

plt.tight_layout()
plt.show()
```

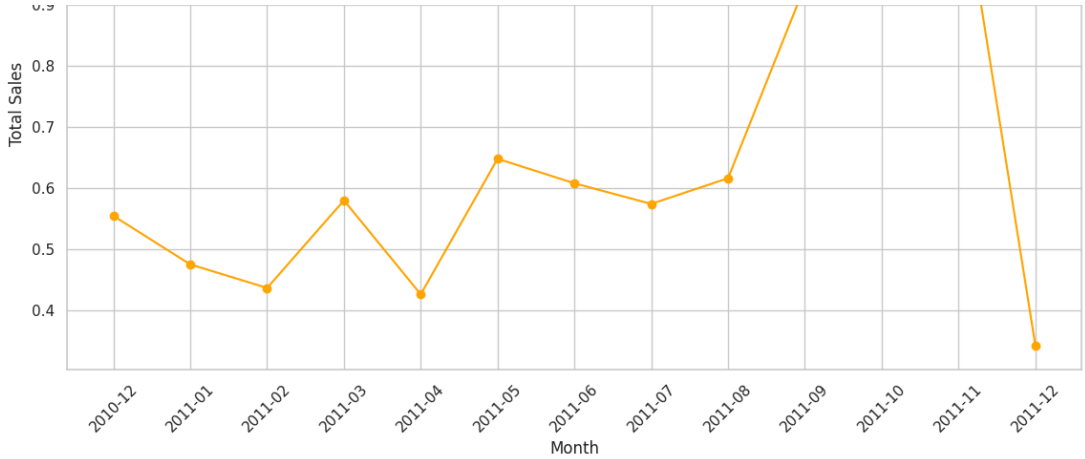
```
Requirement already satisfied: pandas in /usr/local/lib/python3.11/dist-packages (2.2.2)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.11/dist-packages (3.10.0)
Requirement already satisfied: seaborn in /usr/local/lib/python3.11/dist-packages (0.13.2)
Requirement already satisfied: 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) (4.58.4)
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)
```

Choose Files No file chosen Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

```
Saving Online_Retail_Cleaned (1).csv to Online_Retail_Cleaned (1) (1).csv
/tmp/ipython-input-25-4079157158.py:33: FutureWarning:

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

  sns.barplot(x=top_products.values, y=top_products.index, ax=axs[0], palette="Blues_d")
/tmp/ipython-input-25-4079157158.py:39: FutureWarning:

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

  sns.barplot(x=top_customers.index.astype(str), y=top_customers.values, ax=axs[1], palette="Greens_d")
```






Total Sales
0.8
0.7
0.6
0.5
0.4
2010-12
2011-01
2011-02
2011-03
2011-04
2011-05
2011-06
2011-07
2011-08
2011-09
2011-10
2011-11
2011-12
Month