1.Create sales_analysis.py

#Setup & **load data**

```
import pandas as pd
```

#Load CSV file into a DataFrame

```
df=pd.read_csv('sales_data.csv')
print(df)
```

```
          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
..         ...         ...       ...    ...         ...    ...          ...
95  2024-04-05      Tablet         8  20770     CUST096  North       166160
96  2024-04-06  Headphones         1   7647     CUST097   West         7647
97  2024-04-07      Tablet         5  27196     CUST098   East       135980
98  2024-04-08     Monitor         1  30717     CUST099  North        30717
99  2024-04-09  Headphones         5  23376     CUST100  South       116880

[100 rows x 7 columns]
```

#Quick check that data loaded

```
print("First 5 rows:")
print(df.head())
```

```
First 5 rows:
         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
```

2. Explore data

#Number of rows and columns

```
print("Shape of data (rows, columns):")
print(df.shape)
```

```
Shape of data (rows, columns):
(100, 7)
```

#List of all column names

```
print("\nColumn names:")
print(df.columns.tolist())
```

```

Column names:
['Date', 'Product', 'Quantity', 'Price', 'Customer_ID', 'Region', 'Total_Sales']
```

#Data type of each column

```
print("\nData types:")
print(df.dtypes)
```

```

Data types:
Date           object
Product        object
Quantity        int64
Price           int64
Customer_ID    object
Region         object
```

```
Total_Sales     int64
dtype: object
```

#Stats for numeric and non-numeric

```
print("\nBasic summary statistics:")
print(df.describe(include="all"))
```

```
Basic summary statistics:
              Date Product    Quantity         Price Customer_ID Region  \
count          100     100  100.000000    100.000000         100    100
unique         100       5         NaN           NaN         100      4
top     2024-01-01  Tablet         NaN           NaN     CUST001  North
freq             1      26         NaN           NaN           1     28
mean           NaN     NaN    4.780000  25808.510000         NaN    NaN
std            NaN     NaN    2.588163  13917.630242         NaN    NaN
min            NaN     NaN    1.000000   1308.000000         NaN    NaN
25%            NaN     NaN    2.750000  14965.250000         NaN    NaN
50%            NaN     NaN    5.000000  24192.000000         NaN    NaN
75%            NaN     NaN    7.000000  38682.250000         NaN    NaN
max            NaN     NaN    9.000000  49930.000000         NaN    NaN

          Total_Sales
count      100.000000
unique            NaN
top               NaN
freq              NaN
mean    123650.480000
std     100161.085275
min       6540.000000
25%      39517.500000
50%      97955.500000
75%     175792.500000
max     373932.000000
```

3.Clean data (missing values & duplicates)

#Drop completely duplicated rows

```
df = df.drop_duplicates()
```

#Handle missing product names: drop rows where product is missing

```
df = df.dropna(subset=["Product"])
```

#Handle missing quantities: fill with 0

```
df["Quantity"] = df["Quantity"].fillna(0)
```

#Ensure numeric types for quantity and price

```
df["Quantity"] = pd.to_numeric(df["Quantity"], errors="coerce").fillna(0)
df["Price"] = pd.to_numeric(df["Price"], errors="coerce")
```

```
print("Cleaned data preview:")
print(df.head())
```

```
Cleaned data preview:
         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
```

4. Analyze sales (compute metrics)

#Create revenue column

```
df["revenue"] = df["Quantity"] * df["Price"]
```

--- **Metrics** (**at least 3**) ---

#Total revenue (total sales)

```
total_revenue = df["revenue"].sum()
```

#Total number of orders after cleaning

```
total_orders = len(df)
```

#Revenue by product

```
revenue_by_product = df.groupby("Product")["revenue"].sum().sort_values(ascending=False)
```

#Quantity sold by product

```
quantity_by_product = df.groupby("Product")["Quantity"].sum().sort_values(ascending=False)
```

#Best-selling product by revenue

```
best_product_by_revenue = revenue_by_product.idxmax()
best_product_revenue = revenue_by_product.max()

print("Total revenue:", total_revenue)
print("Total orders:", total_orders)
print("\nRevenue by product:")
print(revenue_by_product)
print("\nQuantity by product:")
print(quantity_by_product)
print("\nBest product by revenue:", best_product_by_revenue, "->", best_product_revenue)
```

```
Total revenue: 12365048
Total orders: 100

Revenue by product:
Product
Laptop        3889210
Tablet        2884340
Phone         2859394
Headphones    1384033
Monitor       1348071
Name: revenue, dtype: int64

Quantity by product:
Product
Laptop        136
Tablet        127
Phone         101
Monitor        66
Headphones     48
Name: Quantity, dtype: int64

Best product by revenue: Laptop -> 3889210
```