EXPERIMENT NO: 2

Upload and Analyze the data set given in csv format and perform data preprocessing and visualization

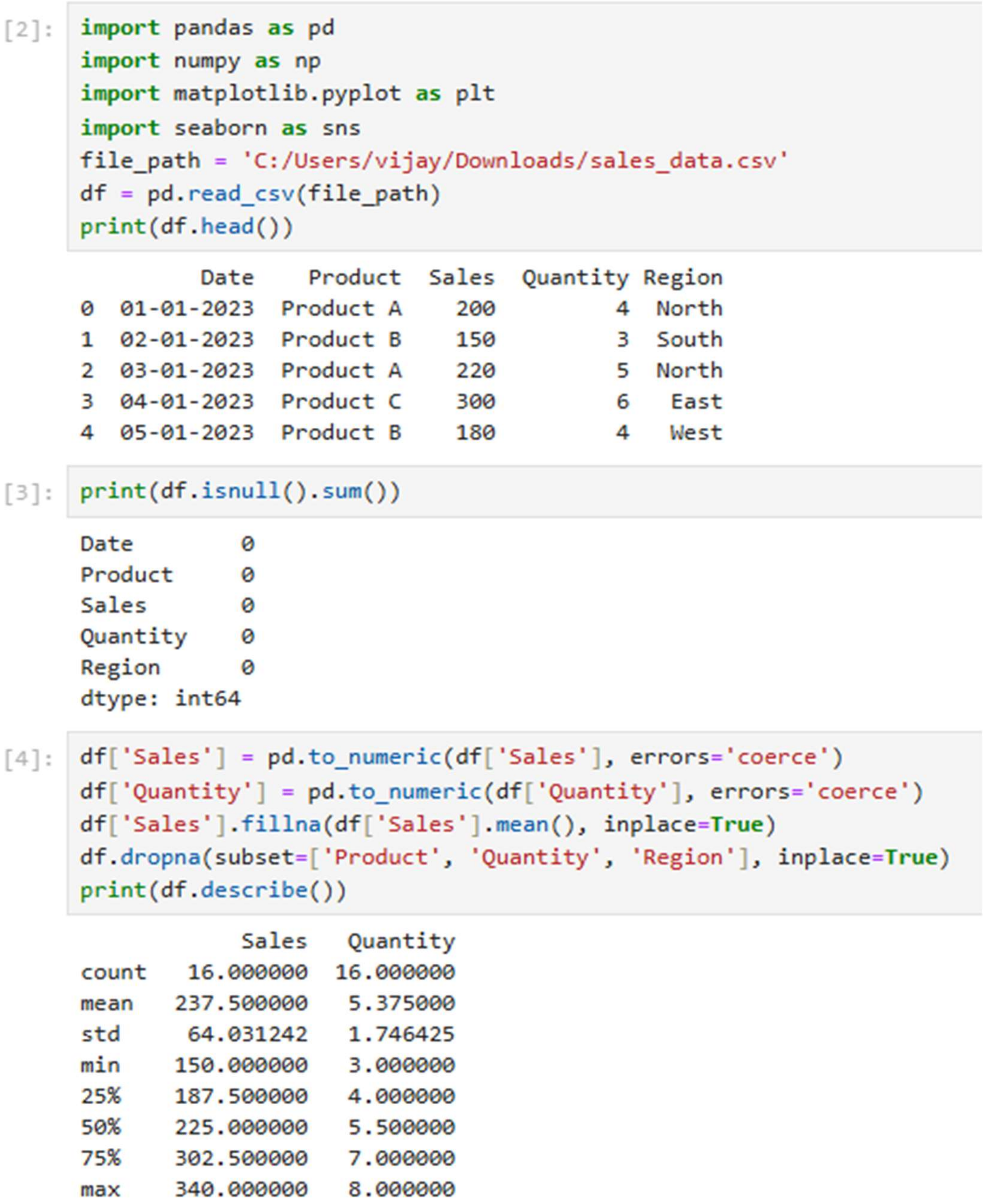
Aim:

To analyze and visualize sales data, clean missing values, summarize total sales and quantities per product, and examine correlations between numeric variables.

Algorithm:

1. **Import libraries: pandas, numpy, matplotlib, seaborn.**
2. **Load the CSV file into a DataFrame.**
3. **Clean data: convert columns to numeric, fill or drop missing values.**
4. **Group data by product to calculate total sales and quantity.**
5. **Plot a bar chart of total sales per product.**
6. **Create a pivot table showing sales by region and product.**
7. **Compute correlation matrix and plot a heatmap.**

Program:



```
[2]: import pandas as pd
     import numpy as np
     import matplotlib.pyplot as plt
     import seaborn as sns
     file_path = 'C:/Users/vijay/Downloads/sales_data.csv'
     df = pd.read_csv(file_path)
     print(df.head())
```

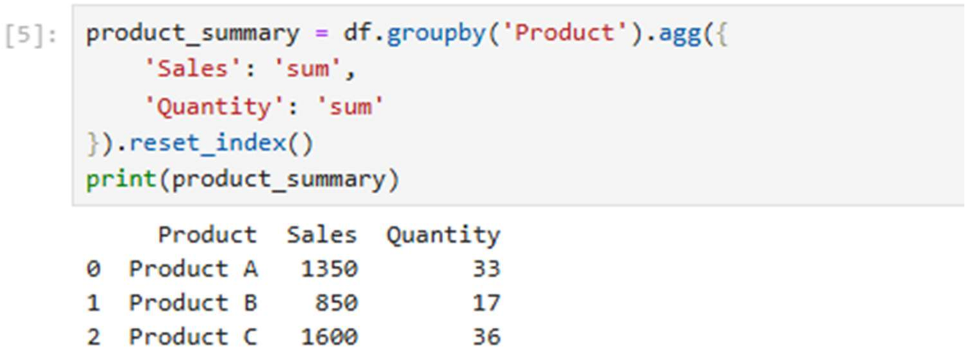
```
         Date    Product  Sales  Quantity Region
0  01-01-2023  Product A    200         4  North
1  02-01-2023  Product B    150         3  South
2  03-01-2023  Product A    220         5  North
3  04-01-2023  Product C    300         6   East
4  05-01-2023  Product B    180         4   West
```

```
[3]: print(df.isnull().sum())
```

```
Date        0
Product     0
Sales       0
Quantity    0
Region      0
dtype: int64
```

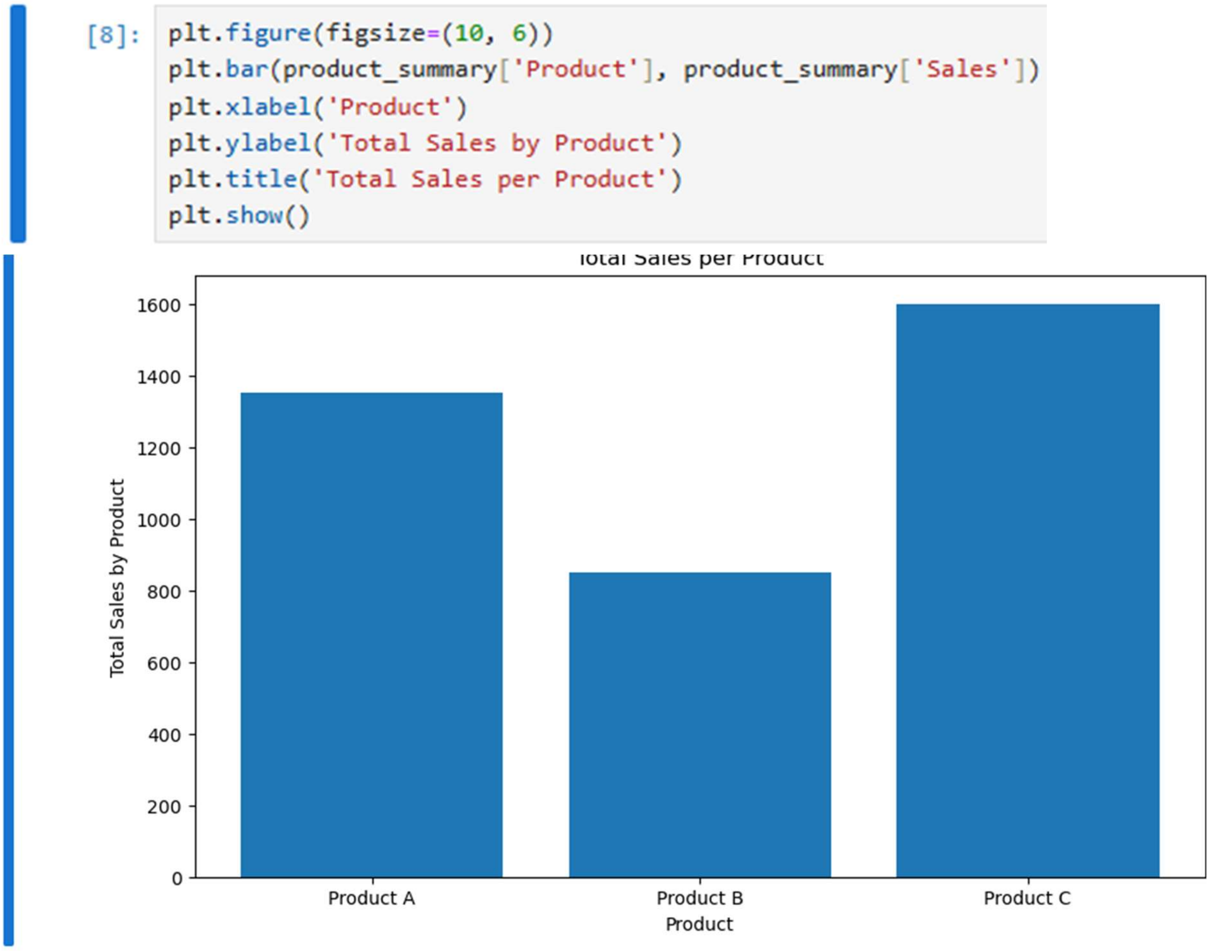
```
[4]: df['Sales'] = pd.to_numeric(df['Sales'], errors='coerce')
     df['Quantity'] = pd.to_numeric(df['Quantity'], errors='coerce')
     df['Sales'].fillna(df['Sales'].mean(), inplace=True)
     df.dropna(subset=['Product', 'Quantity', 'Region'], inplace=True)
     print(df.describe())
```

```
            Sales   Quantity
count   16.000000  16.000000
mean   237.500000   5.375000
std     64.031242   1.746425
min    150.000000   3.000000
25%    187.500000   4.000000
50%    225.000000   5.500000
75%    302.500000   7.000000
max    340.000000   8.000000
```

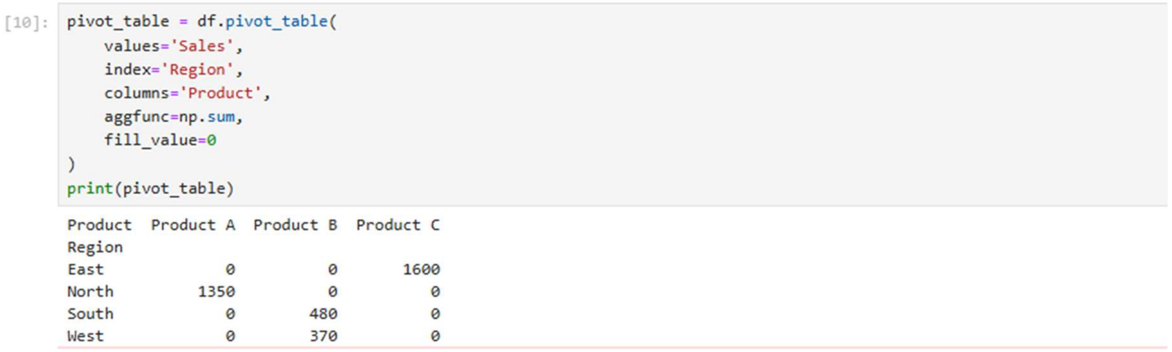


```
[5]: product_summary = df.groupby('Product').agg({
         'Sales': 'sum',
         'Quantity': 'sum'
     }).reset_index()
     print(product_summary)
```

```
     Product  Sales  Quantity
0  Product A   1350        33
1  Product B    850        17
2  Product C   1600        36
```

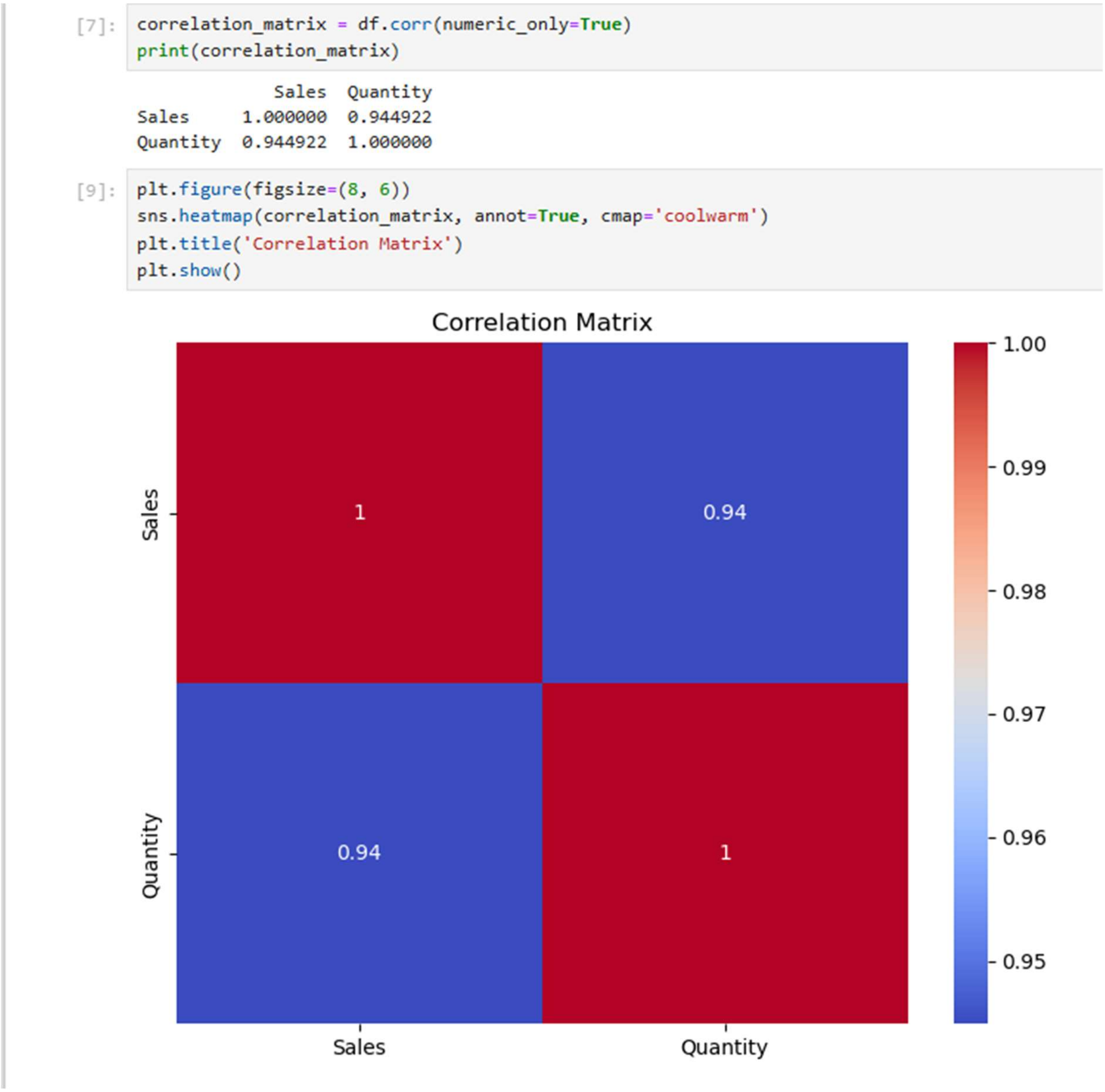


```
[8]: plt.figure(figsize=(10, 6))
     plt.bar(product_summary['Product'], product_summary['Sales'])
     plt.xlabel('Product')
     plt.ylabel('Total Sales by Product')
     plt.title('Total Sales per Product')
     plt.show()
```



```
[10]: pivot_table = df.pivot_table(
          values='Sales',
          index='Region',
          columns='Product',
          aggfunc=np.sum,
          fill_value=0
      )
      print(pivot_table)
```

```
Product  Product A  Product B  Product C
Region
East             0          0       1600
North         1350          0          0
South            0        480          0
West             0        370          0
```



```
[7]: correlation_matrix = df.corr(numeric_only=True)
     print(correlation_matrix)
```

```
              Sales  Quantity
Sales     1.000000  0.944922
Quantity  0.944922  1.000000
```

```
[9]: plt.figure(figsize=(8, 6))
     sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm')
     plt.title('Correlation Matrix')
     plt.show()
```

Result:

The dataset was cleaned, total sales and quantities per product were calculated, the bar chart highlighted top-selling products, the pivot table showed regional sales distribution, and the correlation matrix revealed a strong positive relationship between quantity and sales.