Name: Shubham Jha
Class: D15C
Roll No.: 19

Exp 2

Aim: Data Visualization/ Exploratory data Analysis using Matplotlib and Seaborn.

Introduction:

Exploratory Data Analysis

Exploratory Data Analysis (EDA) is the first step in your data analysis process developed by “John Tukey” in the 1970s. In statistics, exploratory data analysis is an approach to analyzing data sets to summarize their main characteristics, often with visual methods. By the name itself, we can get to know that it is a step in which we need to explore the data set.

When you are trying to build a machine learning model you need to be pretty sure whether your data is making sense or not. The main aim of exploratory data analysis is to obtain confidence in your data to an extent where you’re ready to engage a machine learning algorithm.

Why do we do EDA?

Exploratory Data Analysis is a crucial step before you jump to machine learning or modeling your data. By doing this you can get to know whether the selected features are good enough to model, are all the features required, are there any correlations based on which we can either go back to the Data Preprocessing step or move on to modeling.

Once EDA is complete and insights are drawn, its feature can be used for supervised and unsupervised machine learning modeling.

In every machine learning workflow, the last step is Reporting or Providing the insights to the Stakeholders and as a Data Scientist you can explain every bit of code but you need to keep in mind the audience. By completing the EDA you will have many plots,heat-maps, frequency distribution, graphs, correlation matrix along with the hypothesis by which any individual can understand what your data is all about and what insights you got from exploring your data set.

Data visualization is very critical to market research where both numerical and categorical data can be visualized, which helps in an increase in the impact of insights and also helps in reducing the risk of analysis paralysis

Advantages of Data visualization:

1. Better Agreement:

In business, for numerous periods, it happens that we need to look at the exhibitions of two components or two situations. A conventional methodology is to experience the massive information of both the circumstances and afterward examine it. This will clearly take a great deal of time.

2. A Superior Method:

It can tackle the difficulty of placing the information of both perspectives into the pictorial structure. This will unquestionably give a superior comprehension of the circumstances. For instance, Google patterns assist us with understanding information identified with top ventures or inquiries in pictorial or graphical structures.

3. Simple Sharing of Data:

With the representation of the information, organizations present another arrangement of correspondence. Rather than sharing the cumbersome information, sharing the visual data will draw in and pass on across the data which is more absorbable.

4. Deals Investigation:

With the assistance of information representation, a salesman can, without much of a stretch, comprehend the business chart of items. With information

perception instruments like warmth maps, he will have the option to comprehend the causes that are pushing the business numbers up just as the reasons that are debasing the business numbers. Information representation helps in understanding the patterns and furthermore, different variables like sorts of clients keen on purchasing, rehashing clients, the impact of topography, and so forth.

5. Discovering Relations Between Occasions:

A business is influenced by a lot of elements. Finding a relationship between these elements or occasions encourages chiefs to comprehend the issues identified with their business. For instance, the online business market is anything but another thing today. Each time during certain happy seasons, like Christmas or Thanksgiving, the diagrams of online organizations go up. Along these lines, state if an online organization is doing a normal $1 million business in a specific quarter and the business ascends straightaway, at that point they can rapidly discover the occasions compared to it.

6. Investigating Openings and Patterns:

With the huge loads of information present, business chiefs can discover the profundity of information in regard to the patterns and openings around them. Utilizing information representation, the specialists can discover examples of the conduct of their clients, subsequently preparing for them to investigate patterns and open doors for business.

Introduction to Technologies Used:

Matplotlib

Matplotlib is a plotting library in Python used for creating static, animated, and interactive visualizations. It is highly customizable and supports a wide range of graphs, including bar graphs, histograms, scatter plots, and more.

Seaborn

Seaborn is a Python visualization library built on top of Matplotlib. It provides a high-level interface for creating attractive statistical graphics, such as heatmaps, box plots, and scatter plots.

General Syntax in Python for Data Visualization

Python libraries like Matplotlib and Seaborn follow a general syntax for creating visualizations:

1. **Import the library**: Import the required libraries (e.g., `import matplotlib.pyplot as plt`).
2. **Prepare the data**: Use Pandas to manipulate and prepare the data for visualization.
3. **Create the plot**: Use functions like `plot()`, `scatter()`, `boxplot()`, etc., to create the visualization.
4. **Customize the plot**: Add titles, labels, legends, and other customizations.
5. **Display the plot**: Use `plt.show()` to display the visualization.

<------This doc is using up on the cleaned data of previous experiment.------->

1. Bar Graph and Contingency Table

Theory

- **Bar Graph**: A bar graph is used to represent categorical data with rectangular bars. The length of each bar corresponds to the value it represents. It is useful for comparing categories or showing distributions.
- **Contingency Table**: A contingency table (also called a cross-tabulation) is a table that displays the frequency distribution of two categorical variables. It helps in understanding the relationship between the variables.

Terms

- **Categorical Data**: Data that can be divided into groups or categories (e.g., `Product line`, `Payment`).
- **Frequency**: The number of times a value occurs in a dataset.

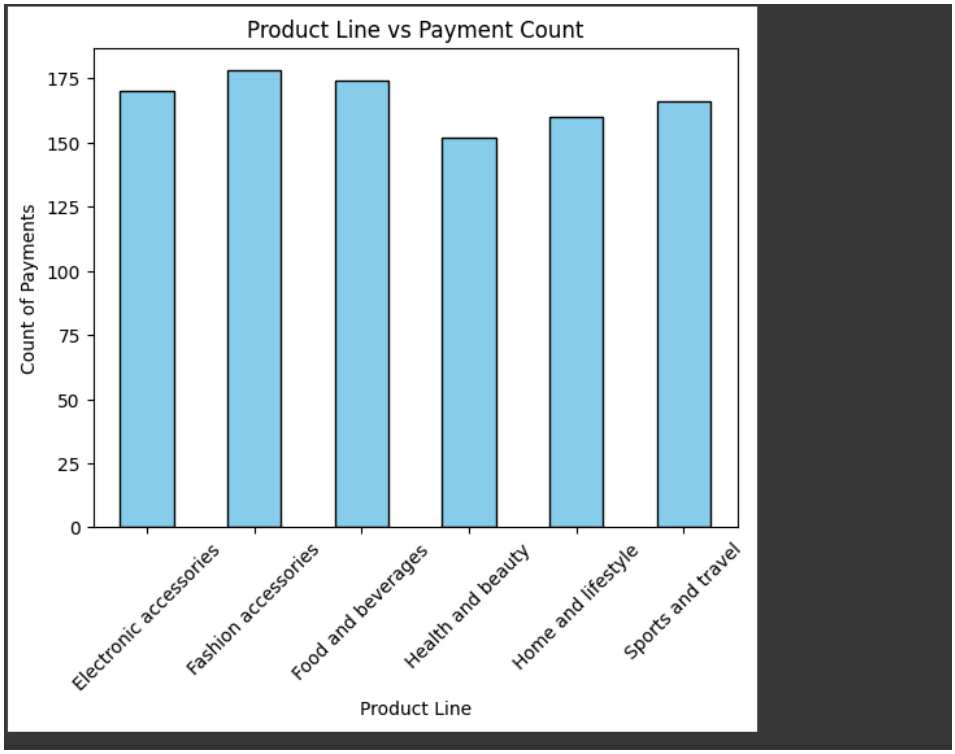
```
# Bar plot for product line and payment method
df.groupby('Product line')['Payment'].count().plot(kind='bar', color='skyblue', edgecolor='blac
k')
plt.title('Product Line vs Payment Count')
plt.xlabel('Product Line')
plt.ylabel('Count of Payments')
plt.xticks(rotation=45)
plt.show()

# Contingency table
contingency_table = pd.crosstab(df['Product line'], df['Payment'])
print(contingency_table)
```

Explanation

- **Bar Graph**:
  - `df.groupby('Product line')['Payment'].count()` groups the data by `Product line` and counts the occurrences of each `Payment` method.
  - `.plot(kind='bar')` creates a bar graph.
  - `plt.title()`, `plt.xlabel()`, and `plt.ylabel()` add titles and labels to the graph.
  - `plt.xticks(rotation=45)` rotates the x-axis labels for better readability.
- **Contingency Table**:
  - `pd.crosstab(df['Product line'], df['Payment'])` creates a table showing the frequency distribution of `Payment` methods for each `Product line`.

Output:



Payment	Cash	Credit card	Ewallet
Product line			
Electronic accessories	71	46	53
Fashion accessories	57	56	65
Food and beverages	57	61	56
Health and beauty	49	50	53
Home and lifestyle	51	45	64
Sports and travel	59	53	54

2. Scatter Plot, Box Plot, and Heatmap

Theory

- **Scatter Plot**: A scatter plot is used to visualize the relationship between two numerical variables. Each point represents an observation.
- **Box Plot**: A box plot (or whisker plot) is used to display the distribution of numerical data through quartiles. It helps identify outliers and compare distributions across categories.
- **Heatmap**: A heatmap is a graphical representation of data where values are represented as colors. It is often used to visualize correlation matrices.

Terms

- **Numerical Data**: Data that represents quantities (e.g., `Unit price`, `Total`).
- **Quartiles**: Values that divide a dataset into four equal parts.
- **Correlation**: A measure of the relationship between two variables.

```
# Scatter plot
sns.scatterplot(data=df, x='Unit price', y='Total', hue='Gender_Male')
plt.title('Unit Price vs Total with Gender (Male=1)')
plt.show()

# Box plot
sns.boxplot(data=df, x='Product line', y='Total')
plt.title('Box Plot of Total by Product Line')
plt.xticks(rotation=45)
plt.show()

# Heatmap
numeric_df = df.select_dtypes(include=['float64', 'int64'])
sns.heatmap(numeric_df.corr(), annot=True, cmap='coolwarm')
plt.title('Heatmap of Numerical Features Correlation')
plt.show()
```

Explanation

- **Scatter Plot**:
    - `sns.scatterplot()` creates a scatter plot with `Unit price` on the x-axis and `Total` on the y-axis.
    - `hue='Gender_Male'` adds a color dimension to differentiate between genders.

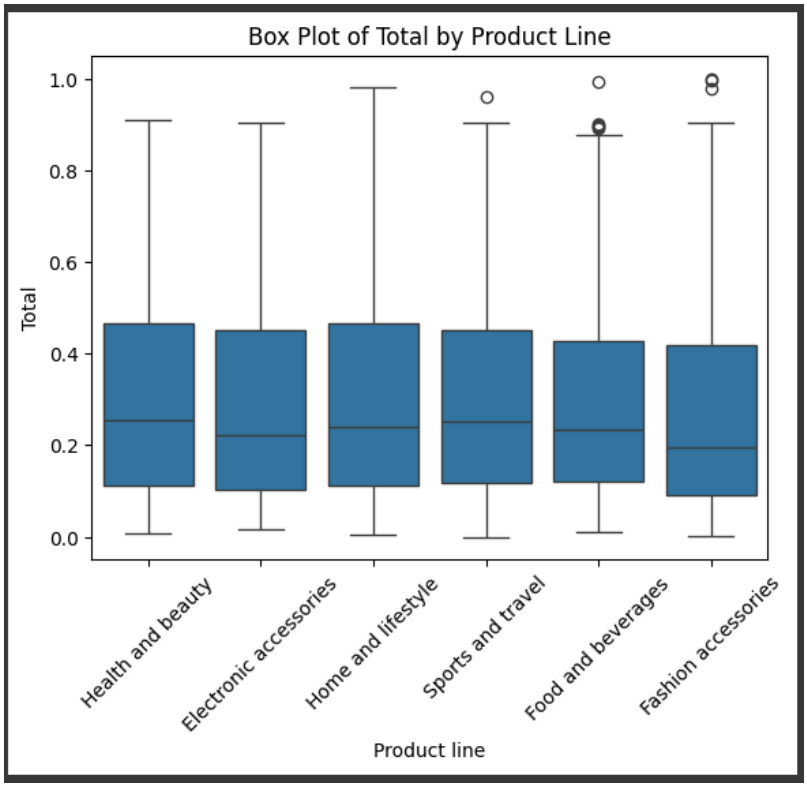
- **Box Plot**:
  - `sns.boxplot()` creates a box plot to show the distribution of `Total` sales across `Product line`.
  - `plt.xticks(rotation=45)` rotates the x-axis labels for better readability.
- **Heatmap**:
  - `numeric_df.corr()` calculates the correlation matrix for numerical features.
  - `sns.heatmap()` visualizes the correlation matrix with colors.

Output:

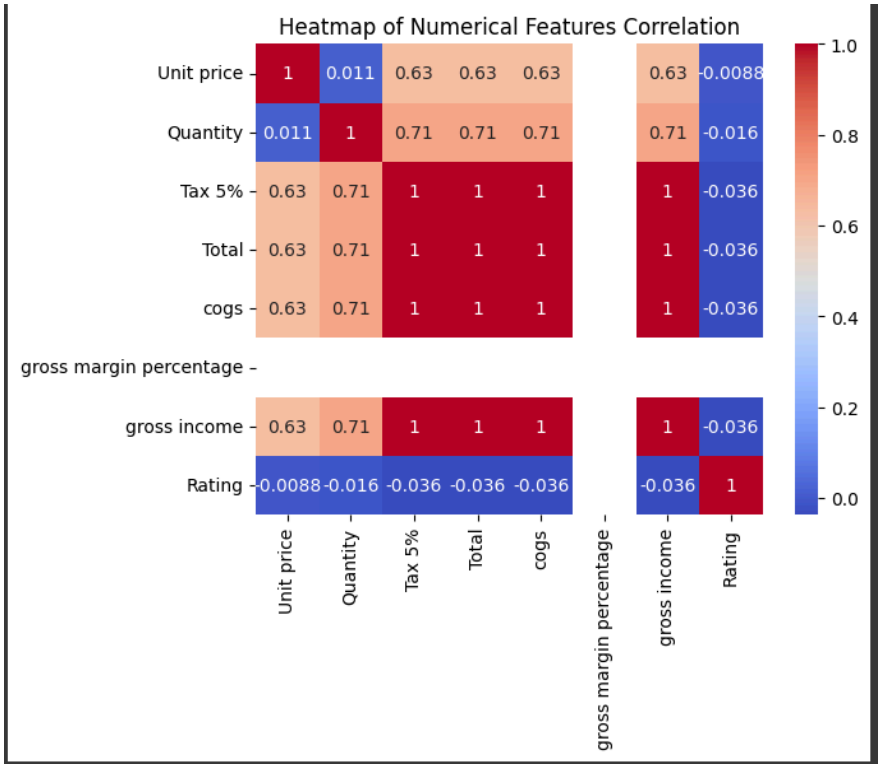
Scatter Plot

Inference

- If the points in the scatter plot show an upward trend (from bottom-left to top-right), it indicates a **positive correlation** between `Unit price` and `Total`. This means that as the unit price increases, the total sales amount also tends to increase.
- If the points are scattered randomly, it suggests **no strong correlation** between the two variables.



Box Plot



Heat Map

Key Observations:

- **Total vs Quantity (High Positive Correlation)**
  - A high positive correlation (close to 1) suggests that the total sales amount increases as the number of purchased items (Quantity) increases. This is expected in sales data.
- **Gross Income vs Total (Strong Positive Correlation)**
  - This indicates that a higher total amount is strongly associated with higher gross income. This is intuitive as gross income is often derived from total sales.
- **Weak Correlations:**
  - Some features, like *Unit Price* and *Quantity*, may show weak or no correlation, suggesting that the number of items purchased doesn't necessarily depend on unit prices.
- **No Negative Correlations:**
  - Since this is a sales dataset, most numerical features are likely positively related.

3. Histogram and Normalized Histogram

Theory

- **Histogram**: A histogram is used to represent the distribution of numerical data. It divides the data into bins and shows the frequency of observations in each bin.
- **Normalized Histogram**: A normalized histogram represents the probability distribution of the data, where the area under the histogram sums to 1.

Terms

- **Bins**: Intervals into which the data is divided.
- **Density**: The probability density of the data.

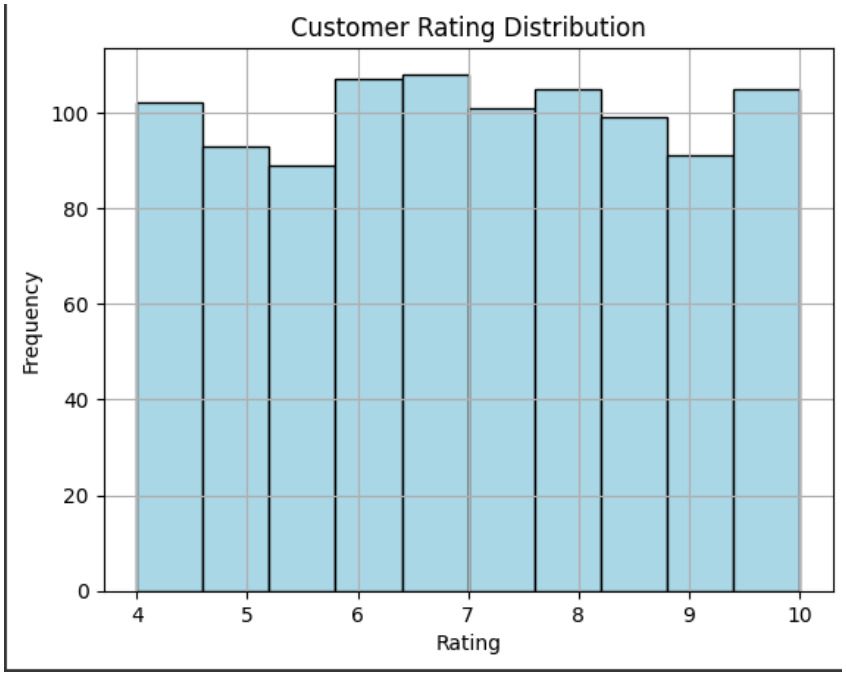
```
# Histogram
df['Rating'].hist(bins=10, color='lightblue', edgecolor='black')
plt.title('Customer Rating Distribution')
plt.xlabel('Rating')
plt.ylabel('Frequency')
plt.show()

# Normalized Histogram
df['Rating'].hist(bins=10, density=True, color='lightgreen', edgecolor='black')
plt.title('Normalized Customer Rating Distribution')
plt.xlabel('Rating')
plt.ylabel('Density')
plt.show()
```

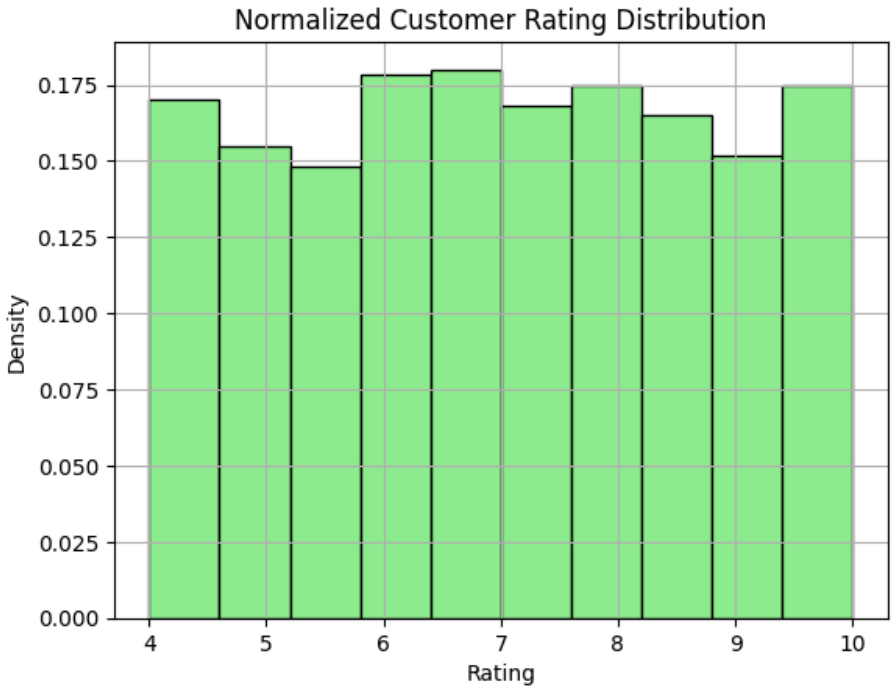
Explanation

- **Histogram**:
  - `df['Rating'].hist()` creates a histogram for the `Rating` column.
  - `bins=10` divides the data into 10 intervals.
  - `color` and `edgecolor` customize the appearance of the bars.
- **Normalized Histogram**:
  - `density=True` normalizes the histogram so that the area under the curve sums to 1.

Output:



Histogram



Normalized Histogram

Inference: Customer Rating Distribution Histogram

1. **Rating Spread:**
   The histogram shows how customer ratings are distributed across different ranges, with the bins dividing ratings from low to high.
2. **Most Common Ratings:**
   If there's a peak near higher ratings (like 8-10), it indicates customer satisfaction, whereas peaks at lower ratings suggest dissatisfaction trends.
3. **Skewness of Ratings:**If the distribution leans towards higher ratings, it suggests overall positive feedback from customers; if it's more balanced, opinions are mixed

4. Handling Outliers Using Box Plot and IQR

Theory

- **Outliers**: Data points that are significantly different from other observations.
- **Box Plot**: A box plot helps visualize outliers using the interquartile range (IQR).
- **IQR Method**: A statistical method to identify and remove outliers. Outliers are defined as observations below `Q1 - 1.5 * IQR` or above `Q3 + 1.5 * IQR`.

Terms

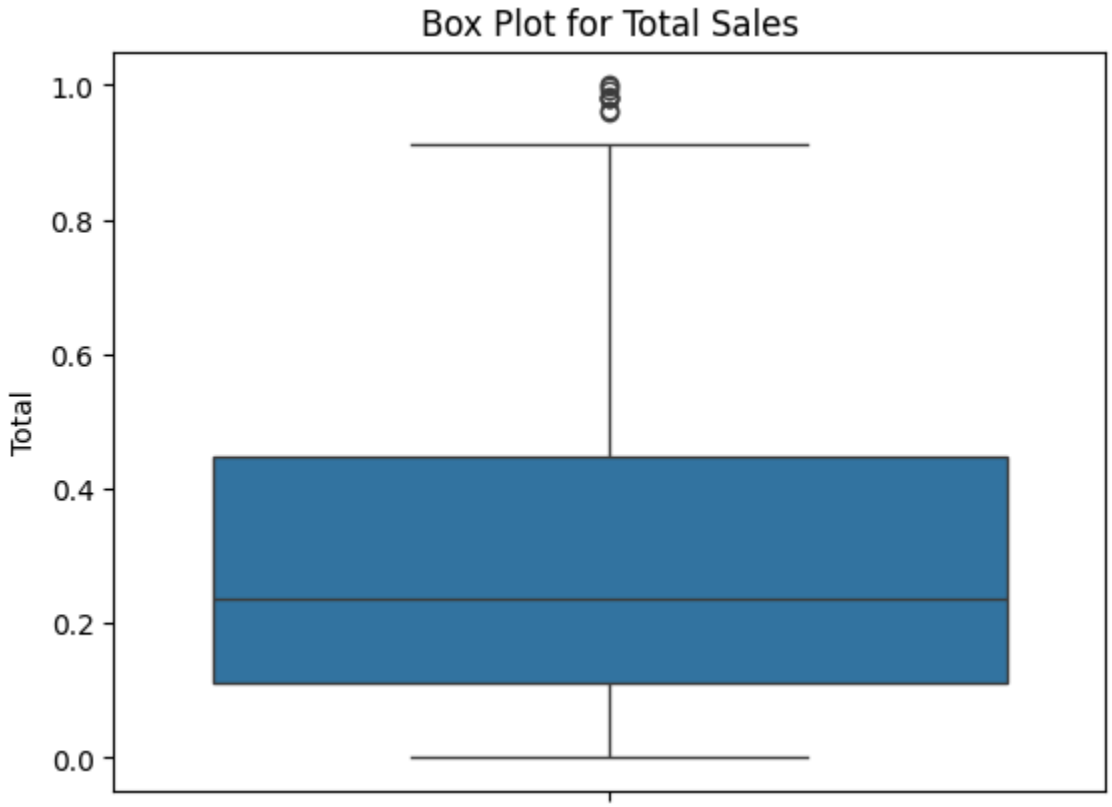
- **Quartiles (Q1, Q3)**: The 25th and 75th percentiles of the data.
- **IQR**: The range between Q1 and Q3.

Code:

```
# Box Plot to Visualize Outliers
sns.boxplot(data=df, y='Total')
plt.title('Box Plot for Total Sales')
plt.show()

# Handle Outliers with IQR
Q1 = df['Total'].quantile(0.25)
Q3 = df['Total'].quantile(0.75)
IQR = Q3 - Q1
lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR
cleaned_df = df[(df['Total'] >= lower_bound) & (df['Total'] <= upper_bound)]
print(f"Rows before outlier removal: {len(df)}")
print(f"Rows after outlier removal: {len(cleaned_df)}")
```

Output:



Inference: Box Plot for Total Sales

1. Identifying Outliers:
    - Any data points outside the whiskers of the box plot are considered outliers. These points represent unusually high total sales amounts.
2. Sales Variability:
    - The spread of the box shows the range of typical sales values, while the whiskers indicate the overall variability.
3. Business Insight:
    - Outliers may indicate rare high-value transactions or potential data entry errors that require investigation.
    - Understanding these outliers can help identify key trends, such as promotional events leading to significant sales.

Outliers detected in *Total* or *Gross Income* columns suggest extreme sales figures, possibly due to special promotions or data entry errors.
Handling these outliers ensures more accurate statistical analysis.

Conclusion:

In this experiment, we conducted an in-depth **Exploratory Data Analysis (EDA)** to uncover patterns and insights within the dataset. We used various visualizations, including bar graphs, scatter plots, box plots, histograms, and heatmaps, to analyze product line performance, payment preferences, customer spending behavior, and rating distributions. Key findings revealed that certain product lines, like "Fashion accessories," had higher transaction counts, cash was the most common payment method, and there was a positive correlation between unit price and total sales. Additionally, most customer ratings clustered around 9, indicating overall satisfaction. Outliers in sales data were identified and removed using the IQR method to improve analysis accuracy.

This experiment reinforced the importance of **EDA** in data-driven decision-making. By leveraging visualization techniques and statistical methods, we gained actionable insights that could optimize inventory management, refine marketing strategies, and enhance customer satisfaction. The process also emphasized the necessity of data cleaning, particularly in handling outliers, to ensure reliable and meaningful analysis.