In [39]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import  warnings
warnings.filterwarnings('ignore')
```

# Retail Dataset

### Importing the dataset

In [41]:
```
df= pd.read_csv(r"C:\Users\91949\Downloads\archive\retail_sales_dataset.csv")
```

In [42]:
```
df.head()
```

Out[42]:

| | Transaction ID | Date | Customer ID | Gender | Age | Product Category | Quantity | Price per Unit | Total Amount |
|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 2023-11-24 | CUST001 | Male | 34 | Beauty | 3 | 50 | 150 |
| **1** | 2 | 2023-02-27 | CUST002 | Female | 26 | Clothing | 2 | 500 | 1000 |
| **2** | 3 | 2023-01-13 | CUST003 | Male | 50 | Electronics | 1 | 30 | 30 |
| **3** | 4 | 2023-05-21 | CUST004 | Male | 37 | Clothing | 1 | 500 | 500 |
| **4** | 5 | 2023-05-06 | CUST005 | Male | 30 | Beauty | 2 | 50 | 100 |

### Data Description

In [5]:
```
df.shape
```

Out[5]:
```
(1000, 9)
```

In [6]:
```
df.columns
```

Out[6]:
```
Index(['Transaction ID', 'Date', 'Customer ID', 'Gender', 'Age',
       'Product Category', 'Quantity', 'Price per Unit', 'Total Amount'],
      dtype='object')
```

In [7]:
```
df.dtypes
```

Out[7]:
```
Transaction ID       int64
Date                object
Customer ID         object
Gender              object
Age                  int64
Product Category    object
Quantity             int64
Price per Unit       int64
Total Amount         int64
dtype: object
```

In [7]:
```
for i in df.columns.to_list():

```

```
    print(f"No of unique values in {i}=>{df[i].nunique()}.")
```

```
No of unique values in Transaction ID=>1000.
No of unique values in Date=>345.
No of unique values in Customer ID=>1000.
No of unique values in Gender=>2.
No of unique values in Age=>47.
No of unique values in Product Category=>3.
No of unique values in Quantity=>4.
No of unique values in Price per Unit=>5.
No of unique values in Total Amount=>18.
```

In [8]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 9 columns):
 #   Column            Non-Null Count  Dtype 
---  ------            --------------  ----- 
 0   Transaction ID    1000 non-null   int64 
 1   Date              1000 non-null   object
 2   Customer ID       1000 non-null   object
 3   Gender            1000 non-null   object
 4   Age               1000 non-null   int64 
 5   Product Category  1000 non-null   object
 6   Quantity          1000 non-null   int64 
 7   Price per Unit    1000 non-null   int64 
 8   Total Amount      1000 non-null   int64 
dtypes: int64(5), object(4)
memory usage: 70.4+ KB
```

In [9]:
```
df.describe()
```

Out[9]:

| | Transaction ID | Age | Quantity | Price per Unit | Total Amount |
|---|---|---|---|---|---|
| **count** | 1000.000000 | 1000.00000 | 1000.000000 | 1000.000000 | 1000.000000 |
| **mean** | 500.500000 | 41.39200 | 2.514000 | 179.890000 | 456.000000 |
| **std** | 288.819436 | 13.68143 | 1.132734 | 189.681356 | 559.997632 |
| **min** | 1.000000 | 18.00000 | 1.000000 | 25.000000 | 25.000000 |
| **25%** | 250.750000 | 29.00000 | 1.000000 | 30.000000 | 60.000000 |
| **50%** | 500.500000 | 42.00000 | 3.000000 | 50.000000 | 135.000000 |
| **75%** | 750.250000 | 53.00000 | 4.000000 | 300.000000 | 900.000000 |
| **max** | 1000.000000 | 64.00000 | 4.000000 | 500.000000 | 2000.000000 |

# Data Cleaning

### Missing values

In [11]:
```
df.isna().sum()
```

Out[11]:
```
Transaction ID      0
Date                0
Customer ID         0
Gender              0
Age                 0
Product Category    0
Quantity            0
Price per Unit      0
Total Amount        0
dtype: int64
```

There are no missing values in the dataset

### Duplicates

In [12]:
```
df.duplicated().sum()
```

Out[12]:
```
0
```

There are no duplicates in the dataset

### Outliers

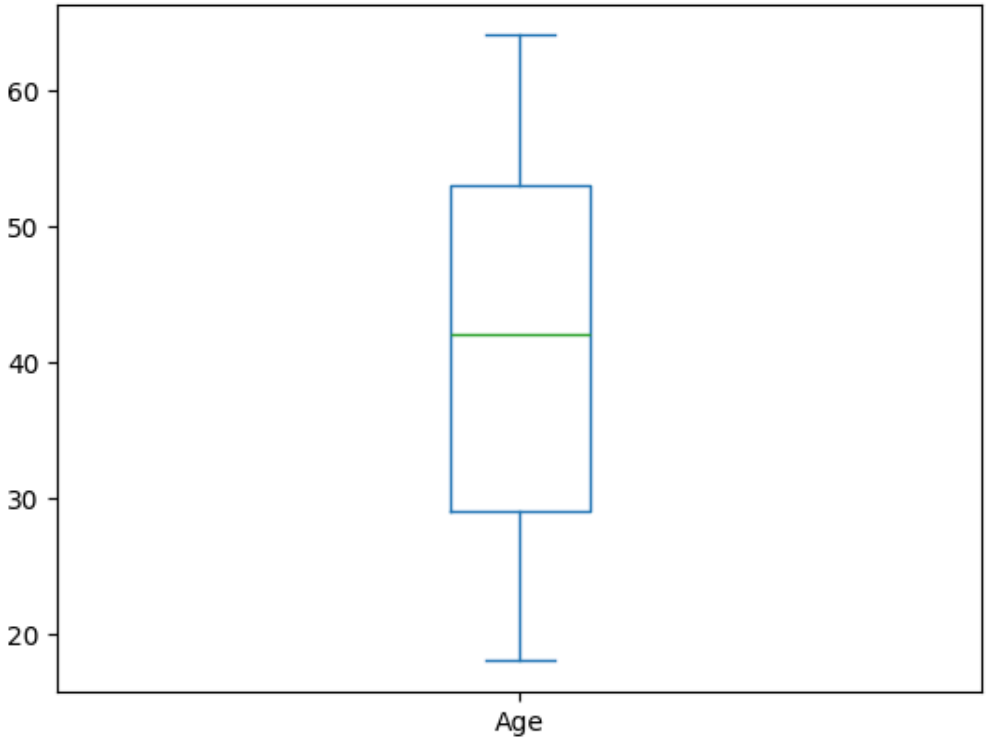
In [13]:
```
df.head()
```

Out[13]:

| | Transaction ID | Date | Customer ID | Gender | Age | Product Category | Quantity | Price per Unit | Total Amount |
|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 2023-11-24 | CUST001 | Male | 34 | Beauty | 3 | 50 | 150 |
| **1** | 2 | 2023-02-27 | CUST002 | Female | 26 | Clothing | 2 | 500 | 1000 |
| **2** | 3 | 2023-01-13 | CUST003 | Male | 50 | Electronics | 1 | 30 | 30 |
| **3** | 4 | 2023-05-21 | CUST004 | Male | 37 | Clothing | 1 | 500 | 500 |
| **4** | 5 | 2023-05-06 | CUST005 | Male | 30 | Beauty | 2 | 50 | 100 |

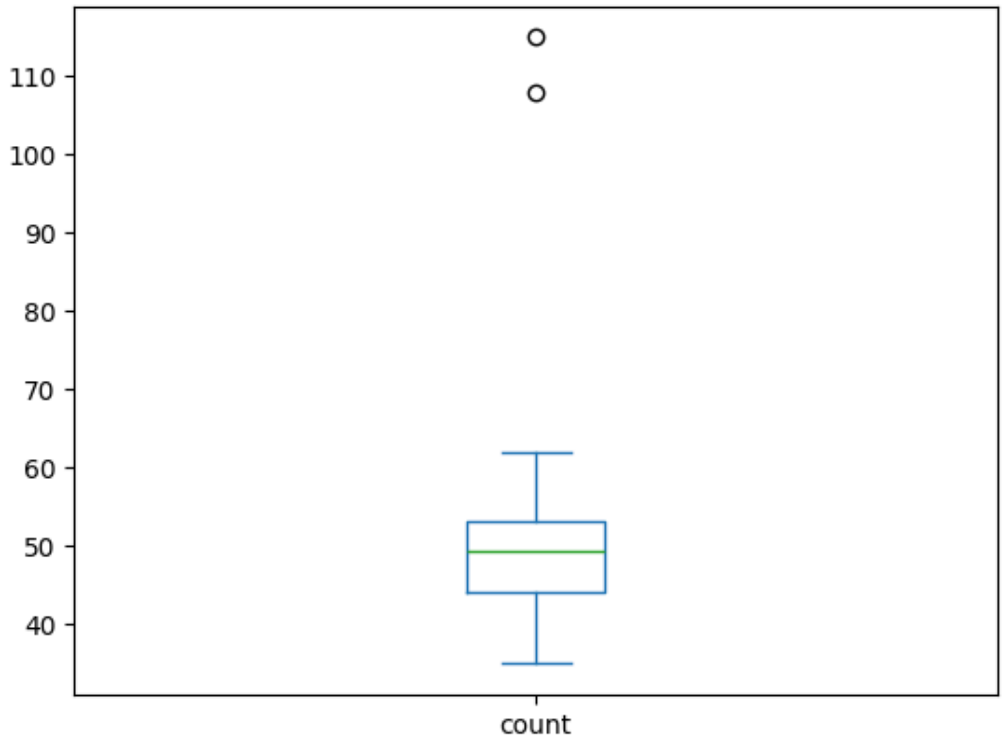
In [14]:
```
df['Age'].plot(kind='box')
```

Out[14]:
```
<Axes: >
```



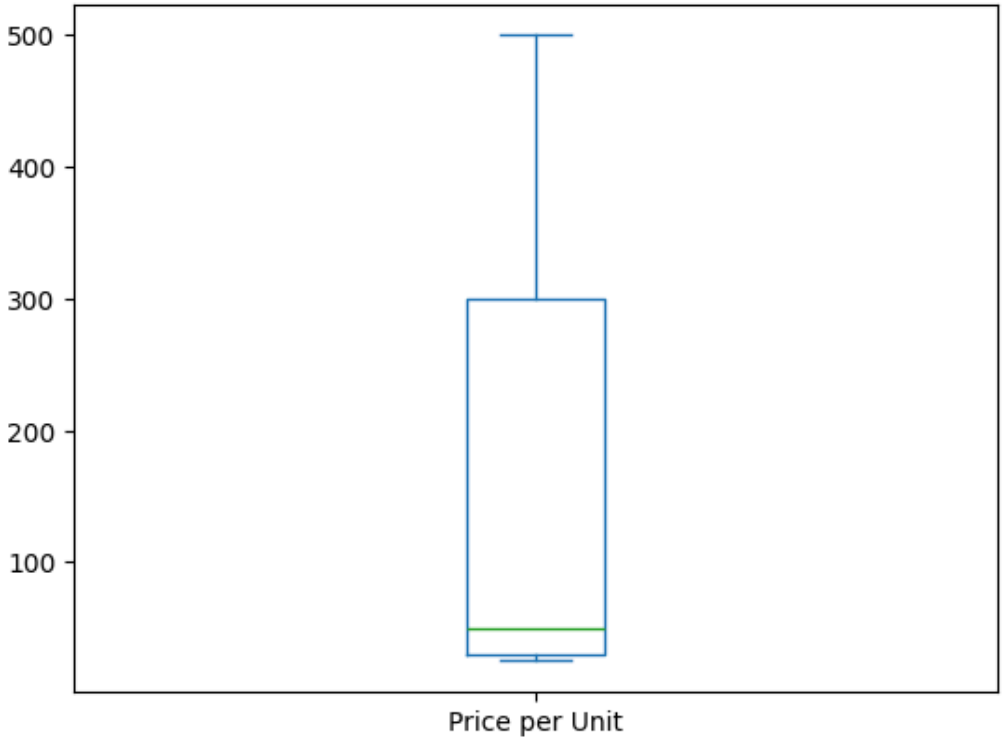

In [15]:
```
df['Total Amount'].value_counts().plot(kind='box')
```

Out[15]: <Axes: >




In [16]:
```
df['Price per Unit'].plot(kind='box')
```

Out[16]: <Axes: >




### Type Casting

In [17]:
```
df.dtypes
```

Out[17]:
```
Transaction ID       int64
Date                object
Customer ID         object
Gender              object
Age                  int64
Product Category    object
Quantity             int64
Price per Unit       int64
Total Amount         int64
dtype: object
```

In [19]:
```
def fun(n):
    return n.split("-")[-1]
```

In [20]:
```
df["day"]=df["Date"].apply(fun)
```

In [21]:
```
df["day"]
```

Out[21]:
```
0      24
1      27
2      13
3      21
4      06
       ..
995    16
996    17
997    29
998    05
999    12
Name: day, Length: 1000, dtype: object
```

In [22]:
```
df
```

Out[22]:

| | Transaction ID | Date | Customer ID | Gender | Age | Product Category | Quantity | Price per Unit | Total Amount | day |
|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 2023-11-24 | CUST001 | Male | 34 | Beauty | 3 | 50 | 150 | 24 |
| **1** | 2 | 2023-02-27 | CUST002 | Female | 26 | Clothing | 2 | 500 | 1000 | 27 |
| **2** | 3 | 2023-01-13 | CUST003 | Male | 50 | Electronics | 1 | 30 | 30 | 13 |
| **3** | 4 | 2023-05-21 | CUST004 | Male | 37 | Clothing | 1 | 500 | 500 | 21 |
| **4** | 5 | 2023-05-06 | CUST005 | Male | 30 | Beauty | 2 | 50 | 100 | 06 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **995** | 996 | 2023-05-16 | CUST996 | Male | 62 | Clothing | 1 | 50 | 50 | 16 |
| **996** | 997 | 2023-11-17 | CUST997 | Male | 52 | Beauty | 3 | 30 | 90 | 17 |
| **997** | 998 | 2023-10-29 | CUST998 | Female | 23 | Beauty | 4 | 25 | 100 | 29 |
| **998** | 999 | 2023-12-05 | CUST999 | Female | 36 | Electronics | 3 | 50 | 150 | 05 |
| **999** | 1000 | 2023-04-12 | CUST1000 | Male | 47 | Electronics | 4 | 30 | 120 | 12 |

1000 rows × 10 columns

In [23]:
```
def fun(n):
    return n.split("-")[-2]
```

In [24]:
```
df["month"]=df["Date"].apply(fun)
```

In [25]:
```
df["month"]
```

Out[25]:
```
0      11
1      02
2      01
3      05
4      05
       ..
995    05
996    11
997    10
998    12
999    04
Name: month, Length: 1000, dtype: object
```

In [26]:
```
df
```

Out[26]:

| | Transaction ID | Date | Customer ID | Gender | Age | Product Category | Quantity | Price per Unit | Total Amount | day | m |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 2023-11-24 | CUST001 | Male | 34 | Beauty | 3 | 50 | 150 | 24 | |
| **1** | 2 | 2023-02-27 | CUST002 | Female | 26 | Clothing | 2 | 500 | 1000 | 27 | |
| **2** | 3 | 2023-01-13 | CUST003 | Male | 50 | Electronics | 1 | 30 | 30 | 13 | |
| **3** | 4 | 2023-05-21 | CUST004 | Male | 37 | Clothing | 1 | 500 | 500 | 21 | |
| **4** | 5 | 2023-05-06 | CUST005 | Male | 30 | Beauty | 2 | 50 | 100 | 06 | |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | |
| **995** | 996 | 2023-05-16 | CUST996 | Male | 62 | Clothing | 1 | 50 | 50 | 16 | |
| **996** | 997 | 2023-11-17 | CUST997 | Male | 52 | Beauty | 3 | 30 | 90 | 17 | |
| **997** | 998 | 2023-10-29 | CUST998 | Female | 23 | Beauty | 4 | 25 | 100 | 29 | |
| **998** | 999 | 2023-12-05 | CUST999 | Female | 36 | Electronics | 3 | 50 | 150 | 05 | |
| **999** | 1000 | 2023-04-12 | CUST1000 | Male | 47 | Electronics | 4 | 30 | 120 | 12 | |

1000 rows × 11 columns

In [27]:
```
def fun(n):
    return n.split("-")[-3]
```

In [28]:
```
df["year"]=df["Date"].apply(fun)
```

In [29]:
```
df["year"]
```

Out[29]:
```
0      2023
1      2023
2      2023
3      2023
4      2023
       ...
995    2023
996    2023
997    2023
998    2023
999    2023
Name: year, Length: 1000, dtype: object
```

In [30]:
```
df
```

Out[30]:

| | Transaction ID | Date | Customer ID | Gender | Age | Product Category | Quantity | Price per Unit | Total Amount | day | m |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 2023-11-24 | CUST001 | Male | 34 | Beauty | 3 | 50 | 150 | 24 | |
| **1** | 2 | 2023-02-27 | CUST002 | Female | 26 | Clothing | 2 | 500 | 1000 | 27 | |
| **2** | 3 | 2023-01-13 | CUST003 | Male | 50 | Electronics | 1 | 30 | 30 | 13 | |
| **3** | 4 | 2023-05-21 | CUST004 | Male | 37 | Clothing | 1 | 500 | 500 | 21 | |
| **4** | 5 | 2023-05-06 | CUST005 | Male | 30 | Beauty | 2 | 50 | 100 | 06 | |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | |
| **995** | 996 | 2023-05-16 | CUST996 | Male | 62 | Clothing | 1 | 50 | 50 | 16 | |
| **996** | 997 | 2023-11-17 | CUST997 | Male | 52 | Beauty | 3 | 30 | 90 | 17 | |
| **997** | 998 | 2023-10-29 | CUST998 | Female | 23 | Beauty | 4 | 25 | 100 | 29 | |
| **998** | 999 | 2023-12-05 | CUST999 | Female | 36 | Electronics | 3 | 50 | 150 | 05 | |
| **999** | 1000 | 2023-04-12 | CUST1000 | Male | 47 | Electronics | 4 | 30 | 120 | 12 | |

1000 rows × 12 columns

In [31]:
```
df['day']=df['day'].astype('uint8')
```

In [32]:
```
df['month']=df['month'].astype('uint8')
```

In [33]:
```
df['year']=df['year'].astype('uint8')
```

In [34]:
```
df.dtypes
```

Out[34]:
```
Transaction ID       int64
Date                object
Customer ID         object
Gender              object
Age                  int64
Product Category    object
Quantity             int64
Price per Unit       int64
Total Amount         int64
day                  uint8
month                uint8
year                 uint8
dtype: object
```

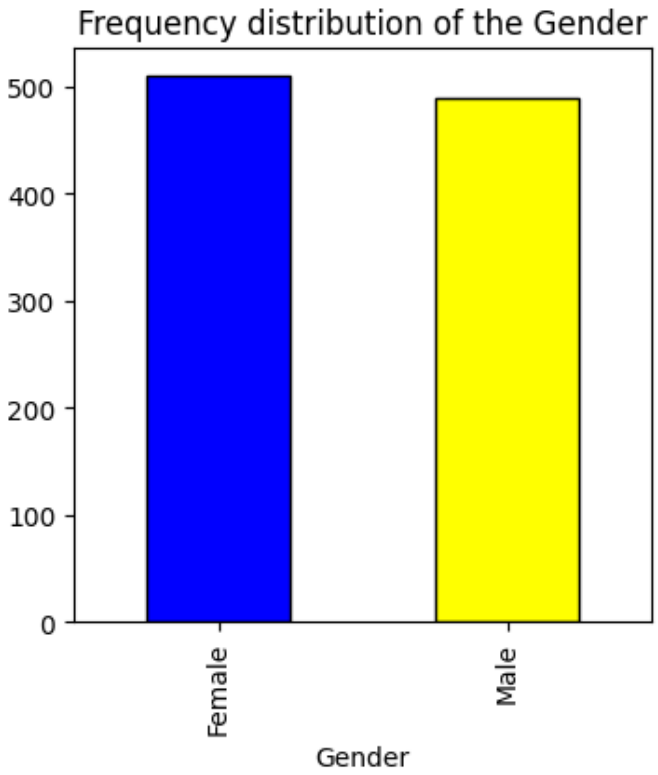
# Data Visualization

## Univariate Analysis

In [47]:
```
df.columns
```

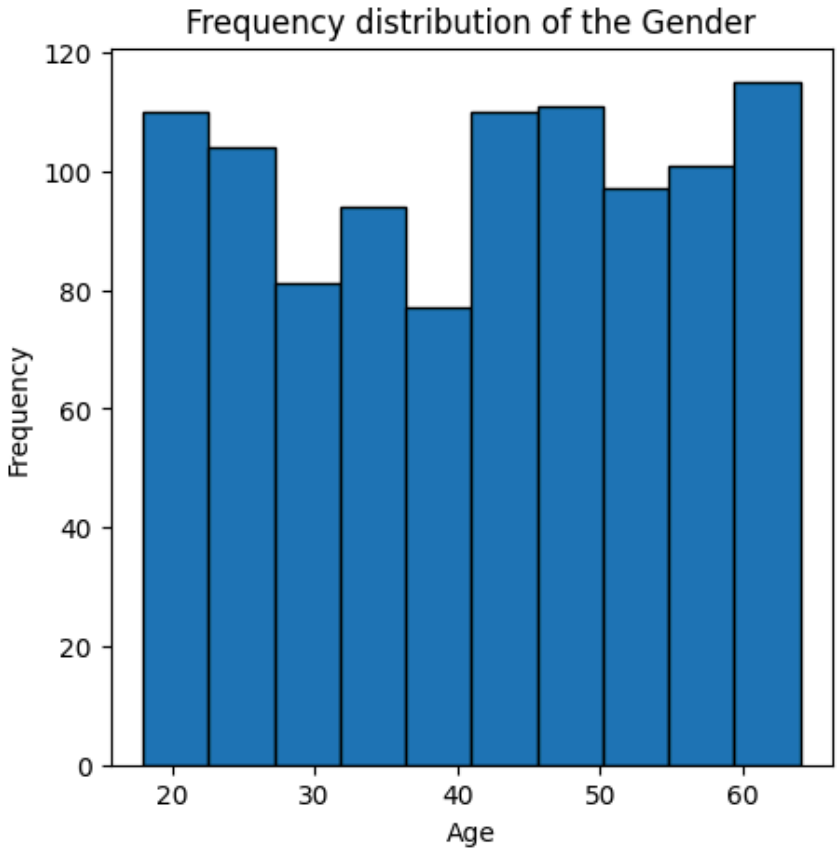
Out[47]:
```
Index(['Transaction ID', 'Date', 'Customer ID', 'Gender', 'Age',
       'Product Category', 'Quantity', 'Price per Unit', 'Total Amount', 'day',
       'month', 'year'],
      dtype='object')
```

In [14]:
```
colors = ['blue','yellow']
plt.figure(figsize = (4,4))
df['Gender'].value_counts().plot(kind='bar',edgecolor='black',color = colors)
plt.title('Frequency distribution of the Gender')
plt.xlabel("Gender")
plt.show()
```



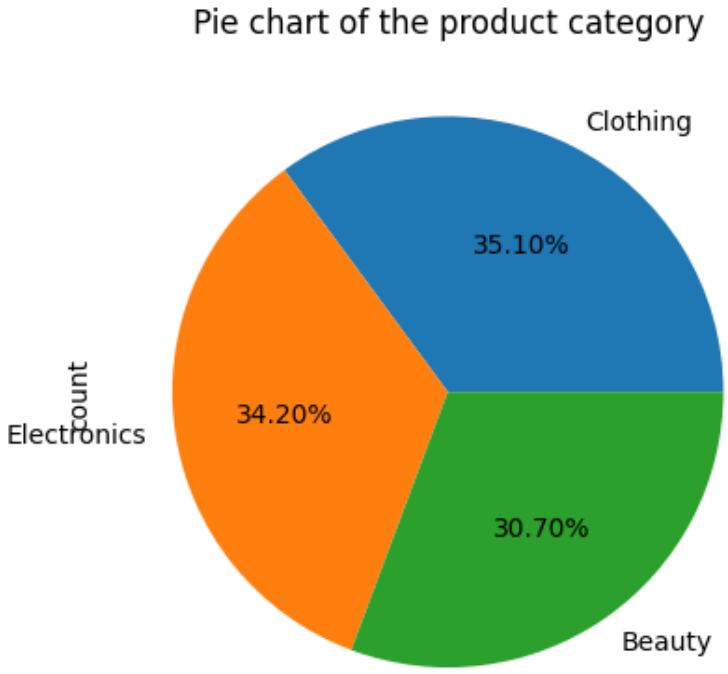

- Female customers are doing more retail shopping compare to male customers.

In [15]:
```
plt.figure(figsize = (5,5))
df['Age'].plot(kind='hist',edgecolor='black')
plt.title('Frequency distribution of the Gender')
plt.xlabel("Age")
plt.show()
```



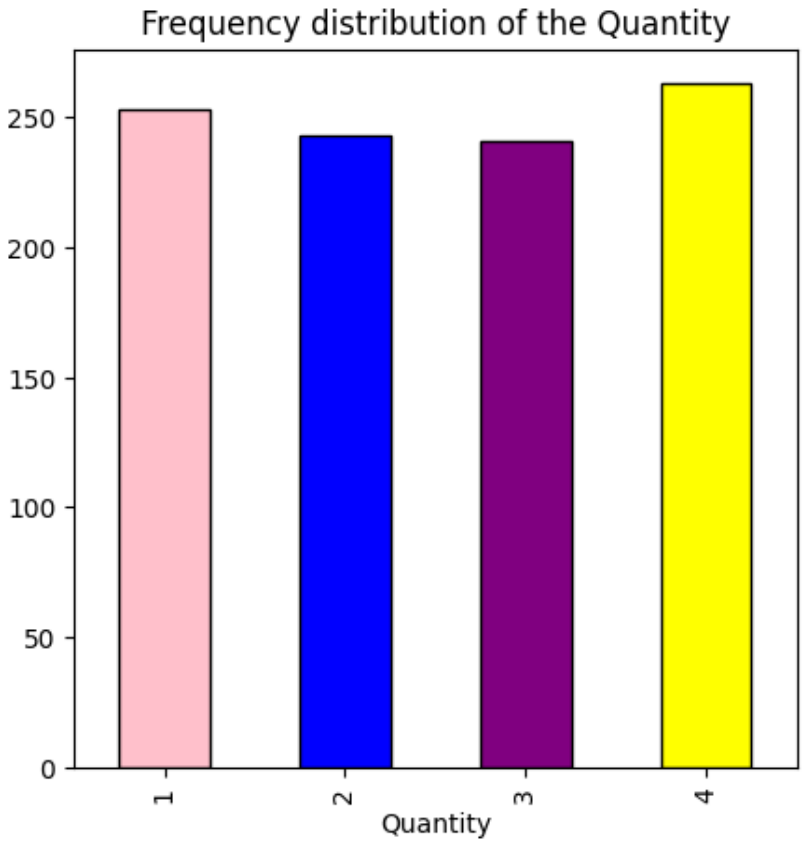

- From the above data the customers above 60 age are doing more retail shopping.
- customers of age 40 are doing less shopping.
- Age 20 & 41-50 customers are doing more Moderate shopping.

In [17]:
```
df['Product Category'].value_counts().plot(kind = 'pie',autopct = '%.2f%%')
plt.title('Pie chart of the product category')
plt.show()
```



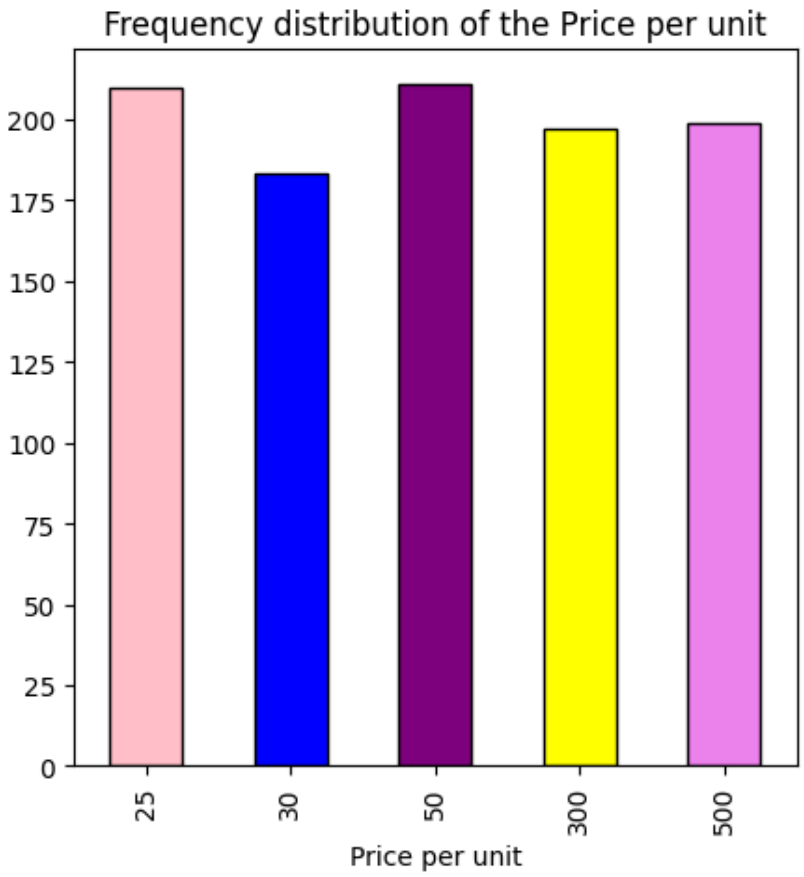
- 35% of the people are doing the clothing shopping
- 34% of people are doing electronics shopping
- The beauty sales is 30%
- According to product category clothing sales are more in the market.

In [19]:
```
colors = ['pink', 'blue','purple','yellow']
plt.figure(figsize = (5,5))
df['Quantity'].value_counts().sort_index().plot(kind="bar",color = colors,edgecolor = '
plt.title('Frequency distribution of the Quantity')
plt.xlabel("Quantity")
plt.show()
```



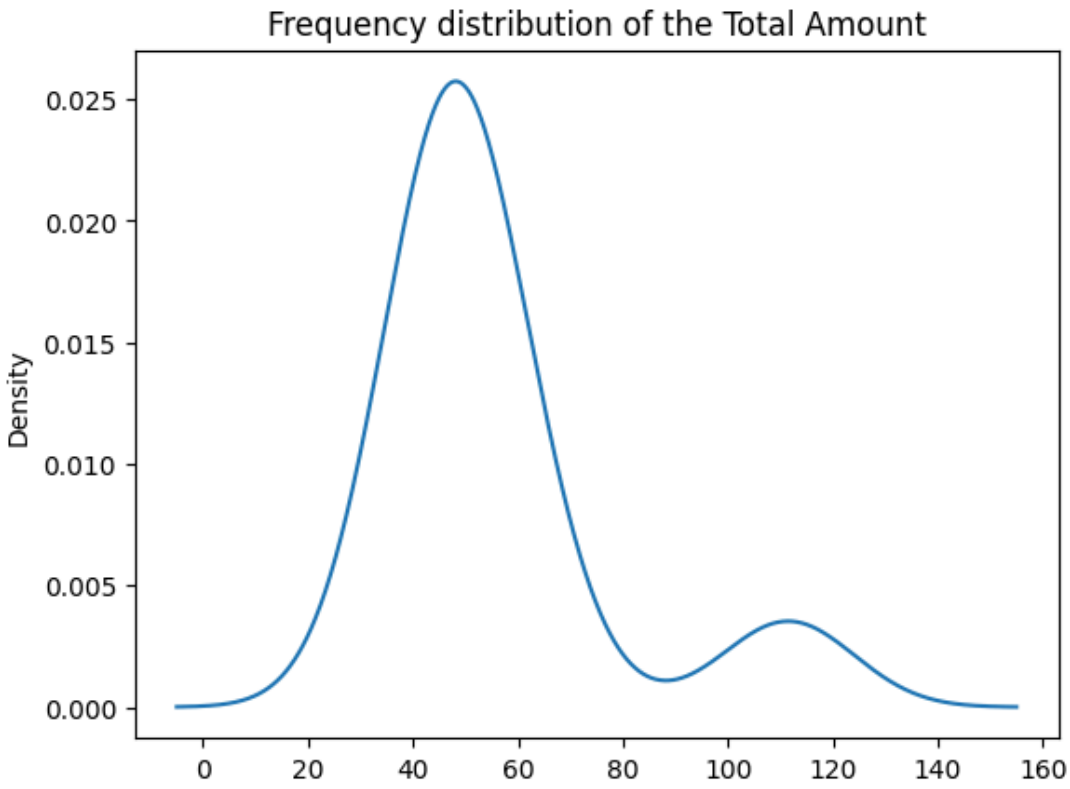

- Most of the people are buying 4 quantities.
- some customers prefer only 1 quantity
- 2 & 3 quantity customers having more sales.

In [21]:
```
colors = ['pink', 'blue','purple','yellow','violet']
plt.figure(figsize = (5,5))
df['Price per Unit'].value_counts().sort_index().plot(kind='bar',color = colors,edgecol
plt.title('Frequency distribution of the Price per unit')
plt.xlabel("Price per unit")
plt.show()
```




- Price per unit 50 and 25 are selling more.
- More sales from 300 to 500 Price per unit.
- According to sales people prefer 25 and 50 price per unit

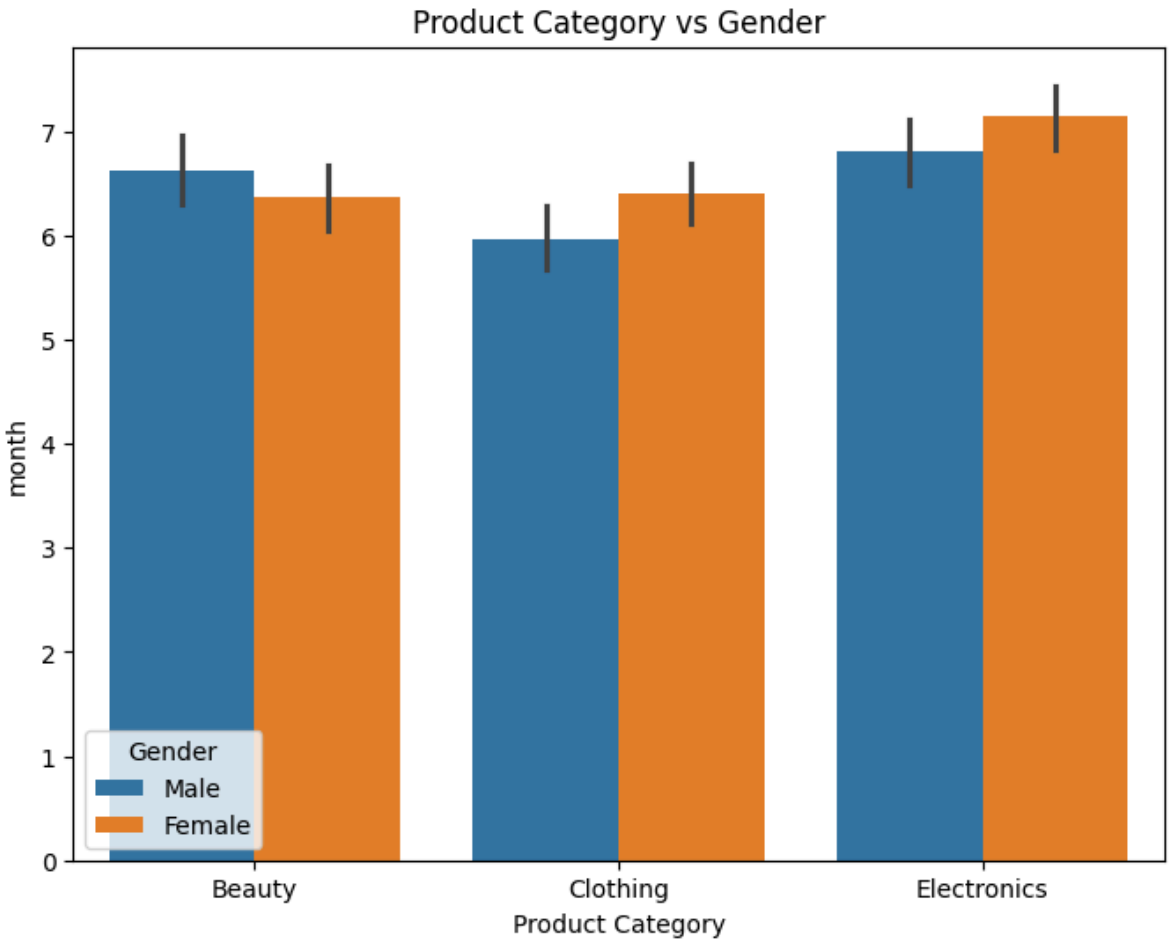
In [23]:
```
df['Total Amount'].value_counts().plot(kind='kde')
plt.title('Frequency distribution of the Total Amount')
plt.show()
```




- From the given data the customers are buying the products with the total amount 50.
- The total amount which lies between 80 to 160 less sales.

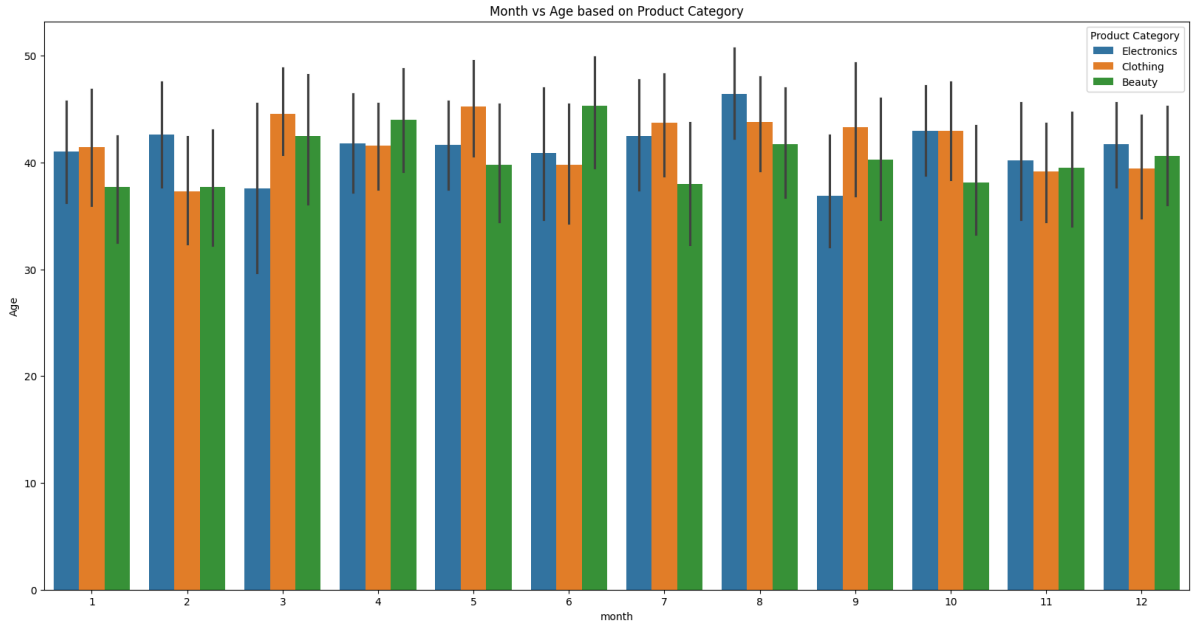
## Bivariate Analysis

In [53]:
```
plt.figure(figsize = (8,6))
plt.title("Product Category vs Gender")
sns.barplot(data = df,x='Product Category',y = 'month',hue = 'Gender',errorbar=('ci',75
plt.show()
```




- Beauty: From the given data ,in the sixth to seventh month male customers bought more beauty products compare to females
- Clothing: From the given data ,in the sixth to seventh month female customers are shopping clothes when compare to male customers
- Electronics: From the given data in the seventh month the electronics products are sold more.

In [54]:
```
plt.figure(figsize = (20,10))
plt.title("Month vs Age based on Product Category")
sns.barplot(data = df,x='month',y = 'Age',hue = 'Product Category')
plt.show()
```

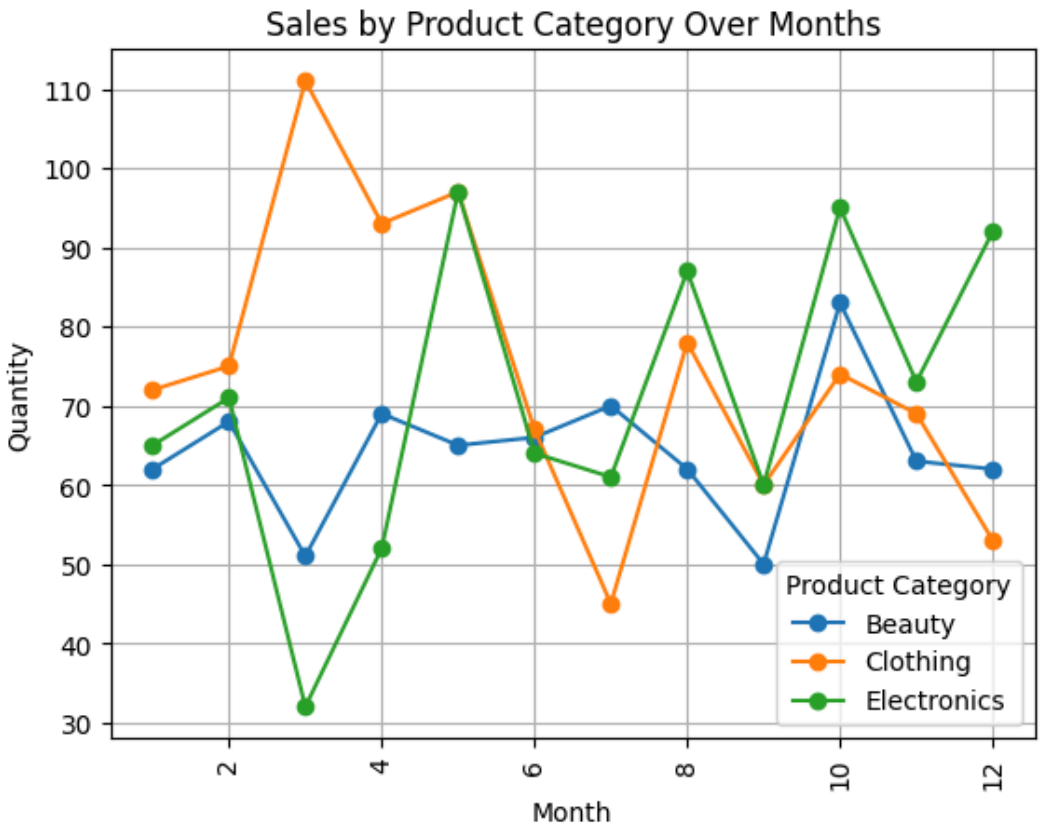



- From the above graph, electronics are more sold in the month of 8 by the age group of 45.
- From the above graph , clothing are more sold in the month of 5 by the age group of 45.
- From the above graph , beauty products are more sold in the month of 6 by the age group of 47.
- More shopping related to beauty , electronics , and clothing more than 45 age group people are shopping when compare to other age group people.

In [55]:
```
pivot = pd.pivot_table(df,values='Quantity',index='month',columns='Product Category',ag
print(pivot)
```

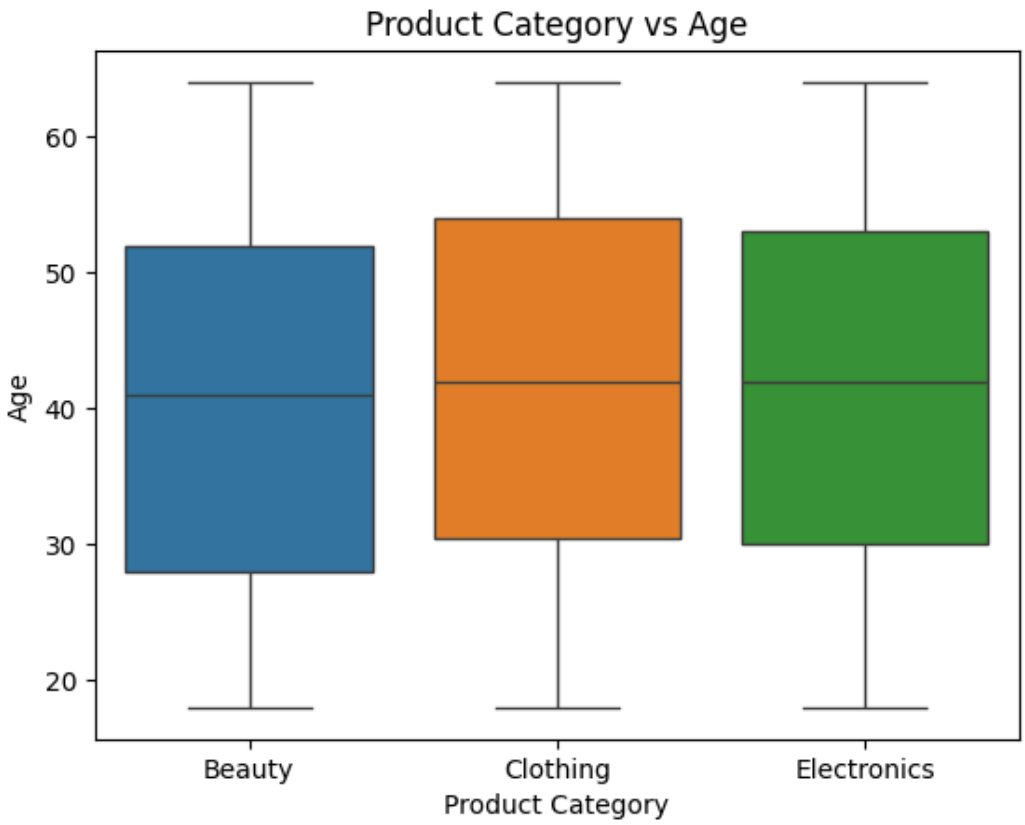
```
Product Category  Beauty  Clothing  Electronics
month
1                     62        72           65
2                     68        75           71
3                     51       111           32
4                     69        93           52
5                     65        97           97
6                     66        67           64
7                     70        45           61
8                     62        78           87
9                     50        60           60
10                    83        74           95
11                    63        69           73
12                    62        53           92
```

In [56]:
```
pivot.plot(kind='line', marker='o')  # Line plot with markers
plt.title("Sales by Product Category Over Months")
plt.ylabel("Quantity")
plt.xlabel("Month")
plt.xticks(rotation=90)  # Rotate x-axis labels for better readability
plt.legend(title='Product Category')
plt.grid(True)
plt.show()
```




- Beauty: From the given data set the beauty products got a sudden decline in the month of 3 and 9 and in the remaining months it is varying constantly.
- Clothing: From the given dataset the clothing products where bought in more quantity when compare to beauty and electronics . In the month of 3 the sales has increased very much and in the month of 7 there is sudden decline in the quantity bought and slowly inclined in the month of 8.
- Electronics: From the given dataset the electronics product has been declined in the month of 3 and there is a sudden increase in the month of 5 and constantly increasing further upto 12 month.

In [35]:
```
plt.title("Product Category vs Age")
sns.boxplot(data = df,x ='Product Category',y = 'Age',palette='tab10')
plt.show()
```

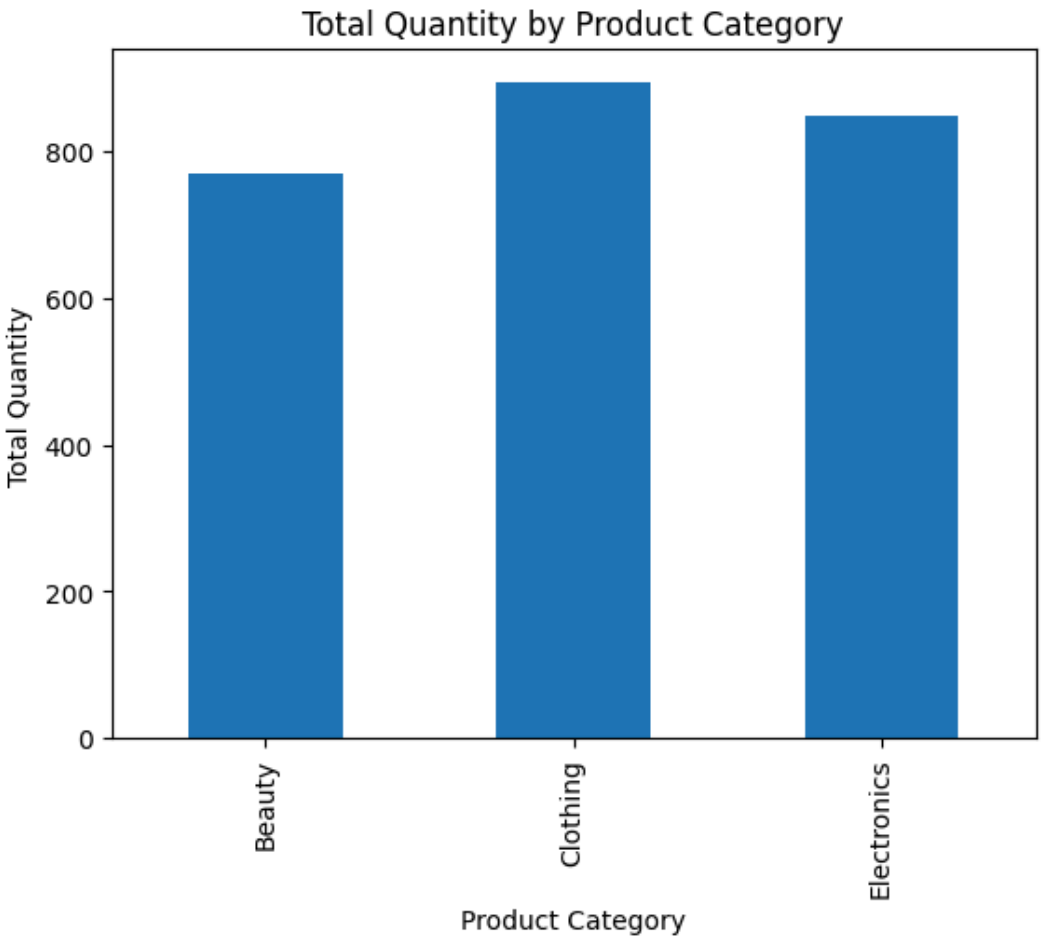



- The data suggests there is no distinct age preference for one product category over another.
- Marketing campaigns should focus on people aged 30 to 50, as the IQR suggests that most customers fall within this range.
- The age range for all categories is consistent, from approximately 20 to 60 years.
- The interquartile range (middle 50% of the data) for all three categories is very similar.

In [58]:
```
pivot = pd.pivot_table(df, values='Quantity', index='Product Category', aggfunc='sum')
print(pivot)
```

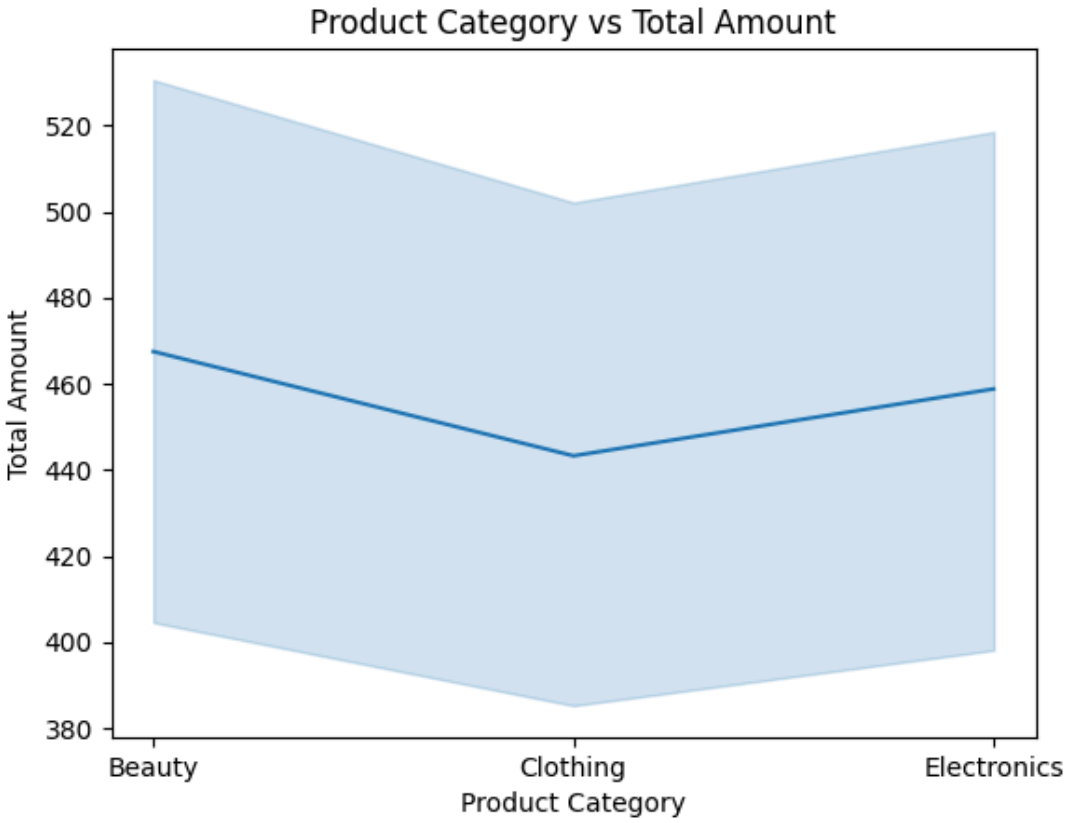
```
                 Quantity
Product Category
Beauty                771
Clothing              894
Electronics           849
```

In [59]:
```
pivot.plot(kind='bar', legend=False)
plt.title('Total Quantity by Product Category')
plt.ylabel('Total Quantity')
plt.xlabel('Product Category')
plt.show()
```



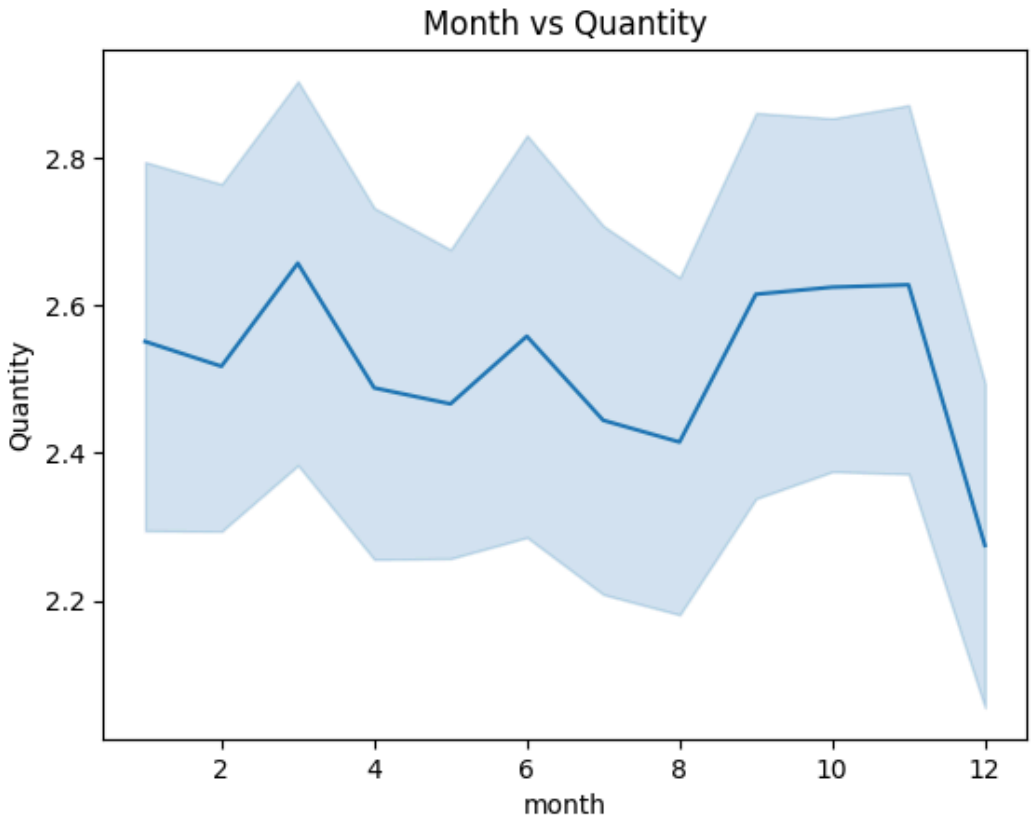

- The total quantity based on the products from the given data is :
  - Beauty products are sold above 700
  - Clothing products are sold above 850
  - Electronics are sold above 800

In [61]:
```
plt.title("Product Category vs Total Amount")
sns.lineplot(data = df,x ='Product Category',y = 'Total Amount')
plt.show()
```



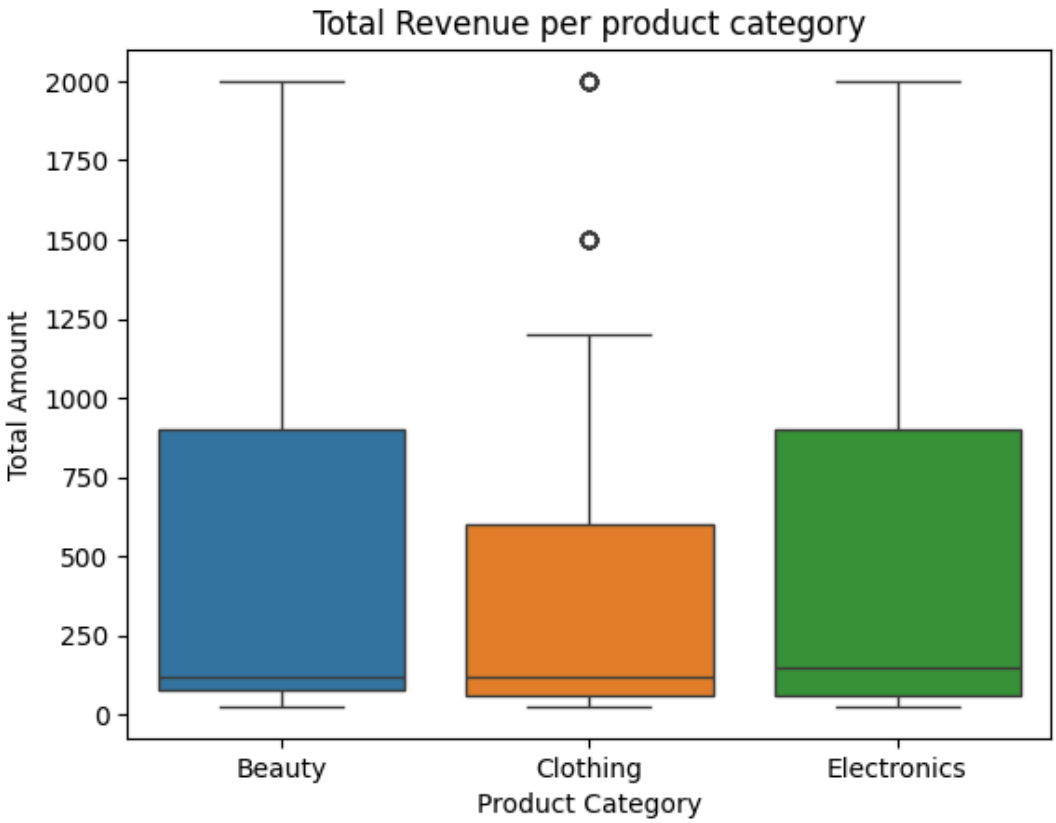

- From the given data set the beauty products has more price and coming to the clothing the total amount is declined and for electronics it has increased gradually.

In [62]:
```
plt.title("Month vs Quantity")
sns.lineplot(data = df, x = 'month',y = 'Quantity')
plt.show()
```



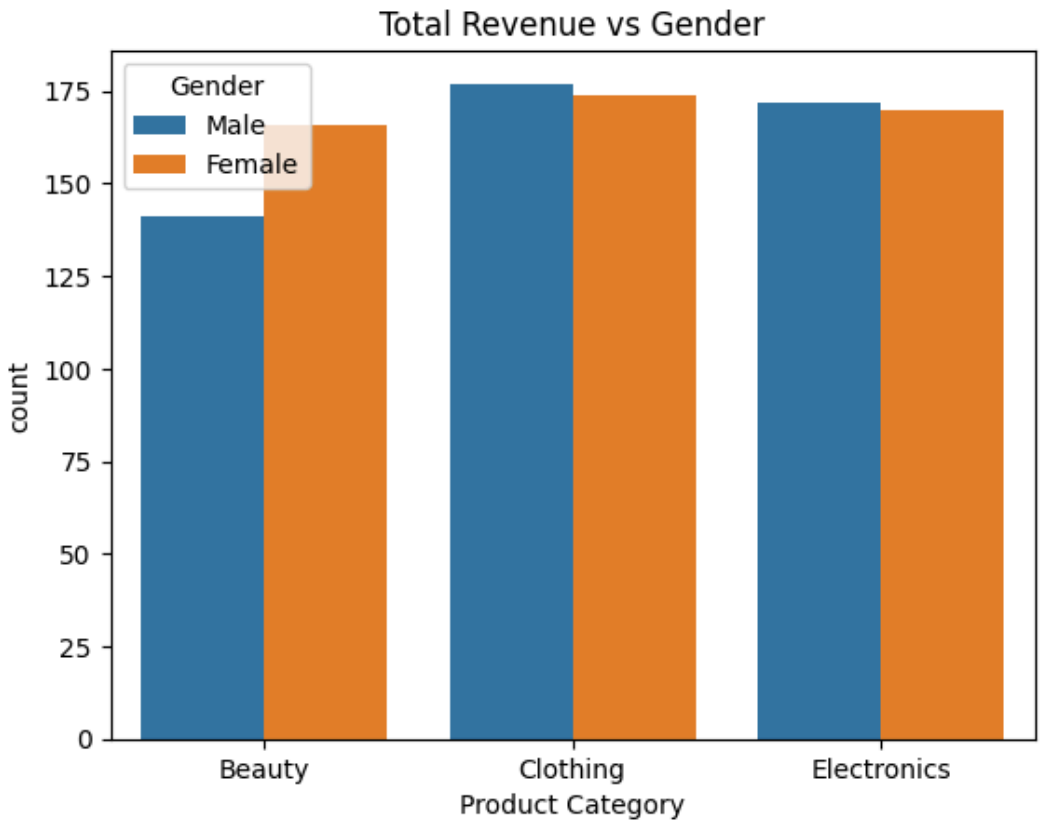
- The focus on mid-year (around months 6-8) might indicate a seasonal trend, where activity increases or decreases noticeably during these months.
- The presence of various peaks and plateaus suggests fluctuations across different months, likely indicating varying performance, sales, or expenses across the year.

In [36]:
```
plt.title("Total Revenue per product category")
sns.boxplot(data = df, x = 'Product Category',y = 'Total Amount',palette = 'tab10')
plt.show()
```



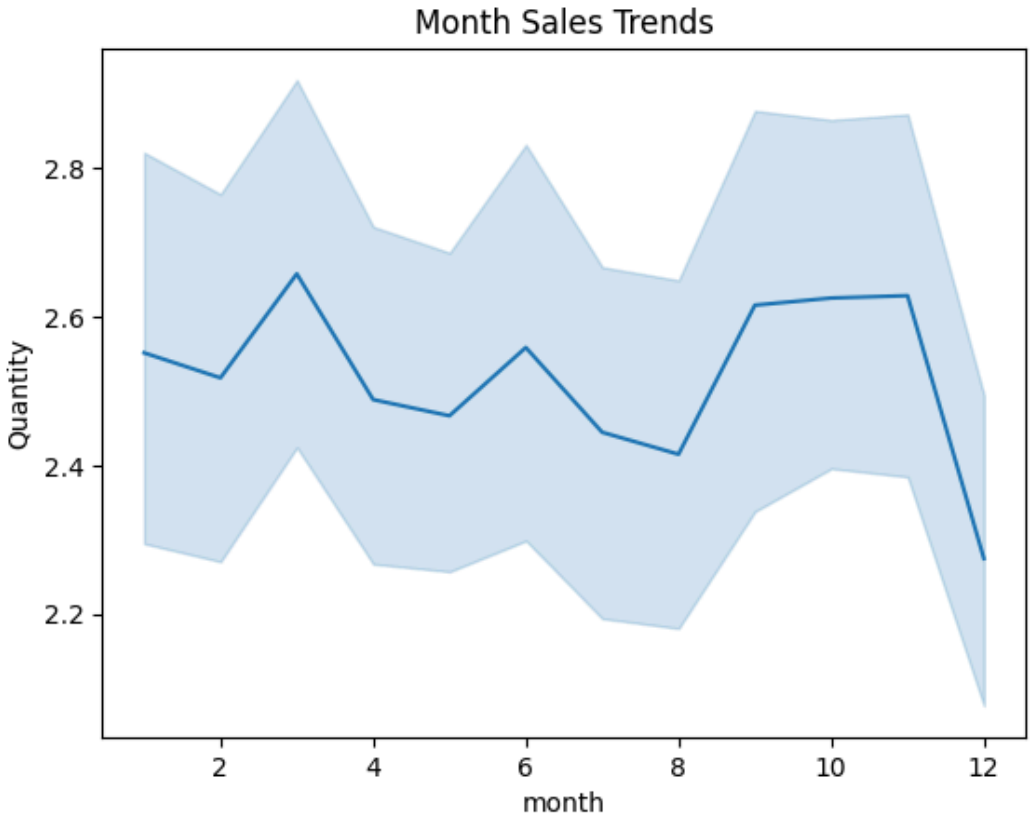

- From the given data, the Beauty and Electronics maximum amount of products lies between 90 to 900.
- The Clothing lies between 90 to 600 Total Amount.
- The interquartile range (middle 50% of the data) for all three categories is very similar.

In [38]:
```
plt.title("Total Revenue vs Gender")
sns.countplot(data = df,x = 'Product Category',hue = 'Gender')
plt.show()
```



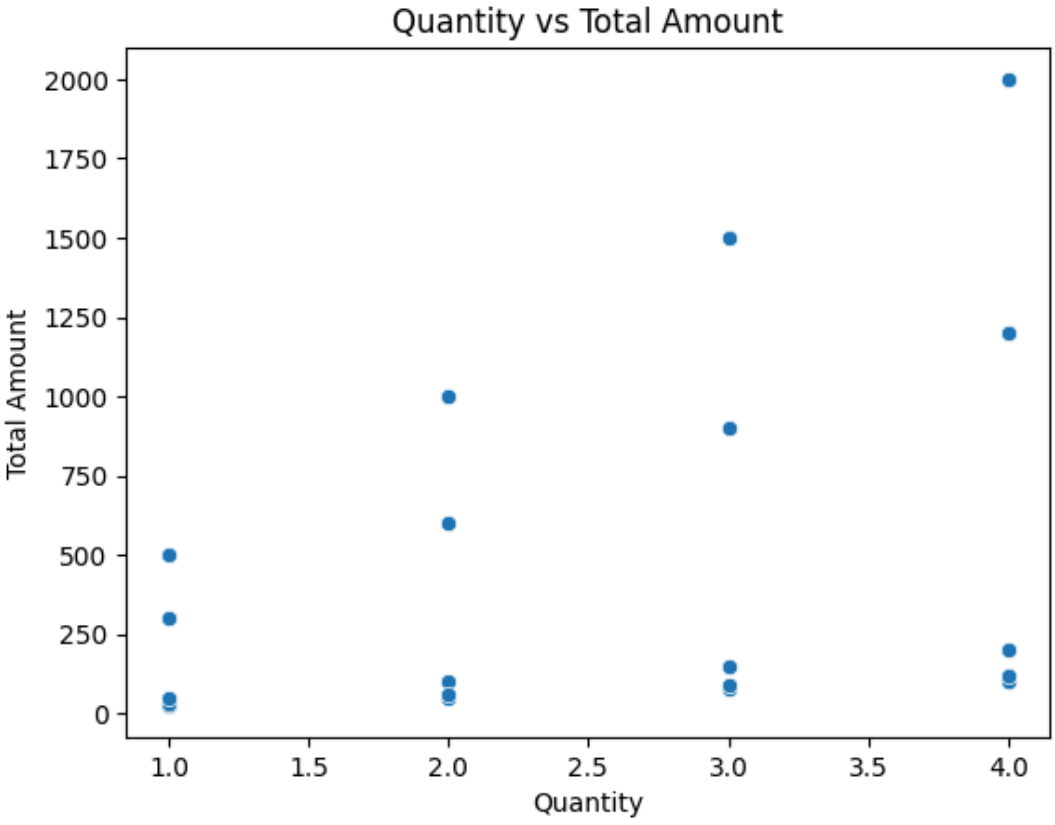

- From the given data, Males are doing more shopping on clothes than other products.
- Beauty- Females are shopping more Beauty products.
- Clothing- Males are shopping more on Clothing.
- Electronics- Males are buying more electronics ,Females are buying somewhat less than males.

In [67]:
```
plt.title("Month Sales Trends")
sns.lineplot(data = df, x= 'month',y='Quantity')
plt.show()
```




- In the month of 3,9,10,11 months customers are purchasing more quantity.
- In the month of 2,5,8,12 very less sales comparing to other months.

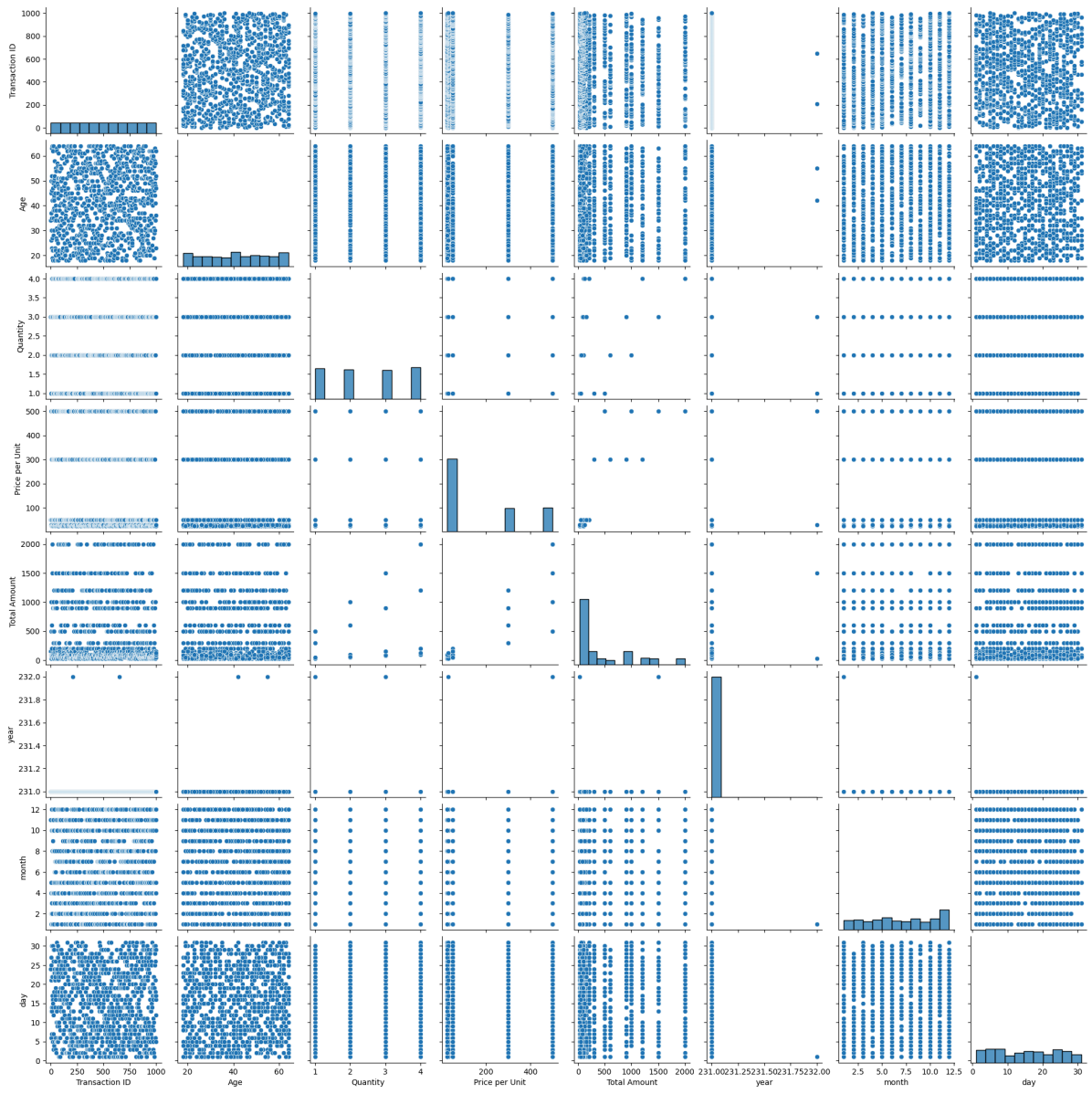
In [69]:
```
plt.title("Quantity vs Total Amount")
sns.scatterplot(data = df,x = 'Quantity',y = 'Total Amount')
plt.show()
```



- The highest amount is 500 for the quantity 1.
- The highest amount is 1000 for the 2 quantities.
- The highest amount is 1500 for the 3 quantities.
- The highest amount is 2000 for the 4 quantities.

  As the quantity is increasing, the amount is increasing.

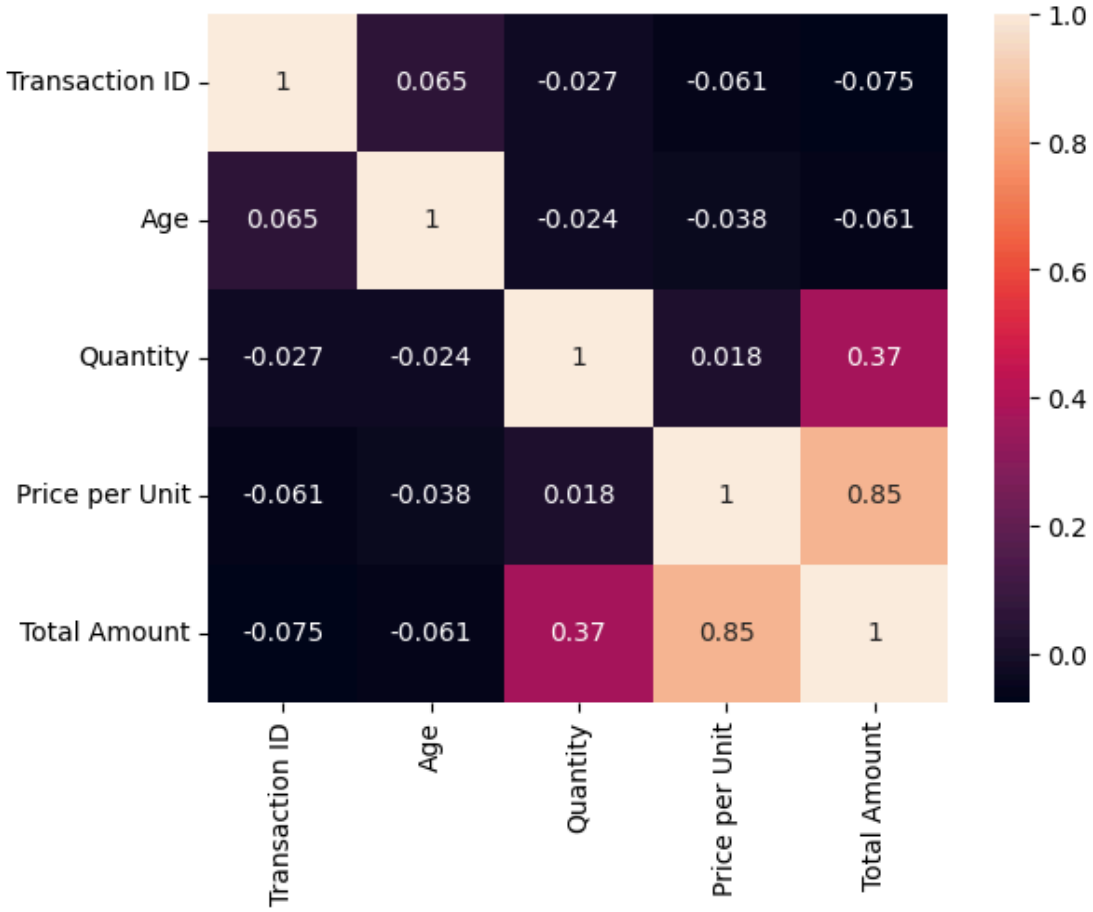
In [71]:
```
sns.pairplot(data = df)
plt.show()
```




Multivariate Analysis

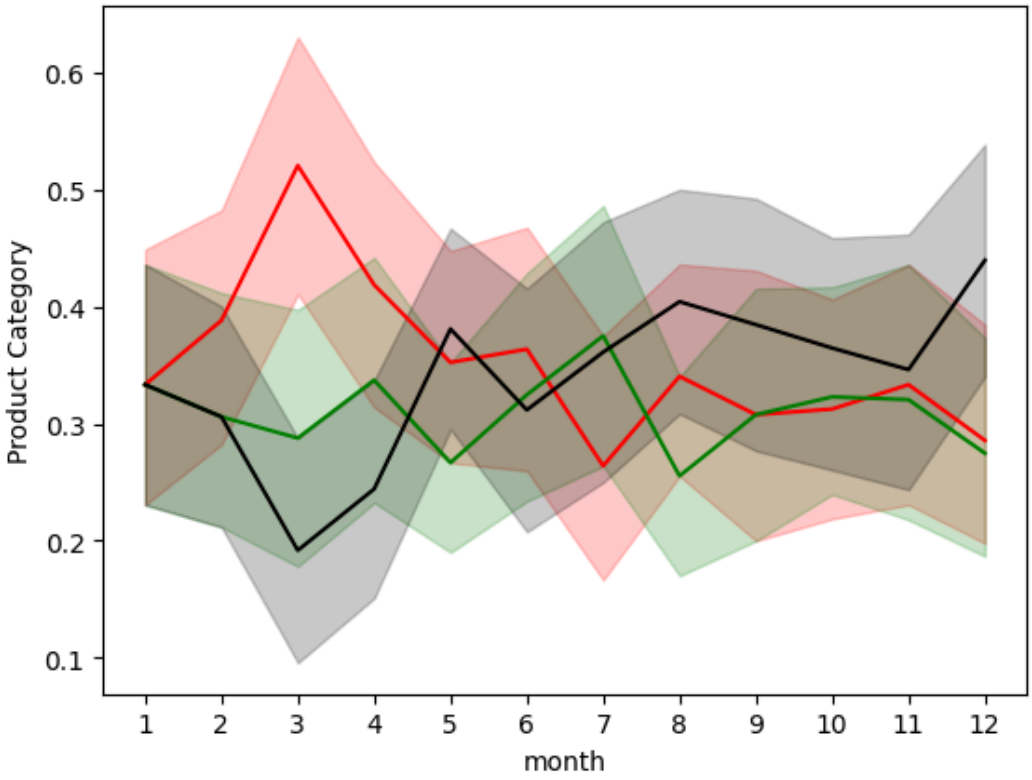
In [61]:
```
sns.heatmap(df.corr(numeric_only = True),annot=True)
```

Out[61]: <Axes: >




- Strong Positive :- Total Amount and Price per Unit
- Weak Positive:- Quantity and Total Amount
- Weak Negative:- Quantity and Transaction Id

In [73]:
```
sns.lineplot(data=df,y=df['Product Category']=='Clothing',x='month',color='red')
sns.lineplot(data=df,y=df['Product Category']=='Beauty',x='month',color='green')
sns.lineplot(data=df,y=df['Product Category']=='Electronics',x='month',color='black')
plt.xticks([1,2,3,4,5,6,7,8,9,10,11,12])
plt.show()
```




- In the month of 3 clothing sales are more and electronics sales is less and the month of 5 electronics sales is more and beauty sales is less.
- In the month of 7 clothing sales is less and beauty sales is increased.
- In the month of 8 electronics sales is more and beauty,clothing sales are less.

# Summary

**There is no missing values,Duplicates,Outliers in the dataset.**

## Univariate Analysis

**- Female customers are doing more retail shopping compare to male customers**

**- From the above data the customers above 60 age are doing more retail shopping.**

**- customers of age 40 are doing less shopping.**

**- Age 20 & 41-50 customers are doing more Moderate shopping.**

**- 35% of the people are doing the clothing shopping**

**- 34% of people are doing electronics shopping**

**- The beauty sales is 30%.**

**- According to product category clothing sales are more in the market.**

**- Most of the people are buying 4 quantities.**

**- some customers prefer only 1 quantity**

**- 2 & 3 quantity customers having more sales.**

**- Price per unit 50 and 25 are selling more.**

**- More sales from 300 to 500 Price per unit.**

**- According to sales people prefer 25 and 50 price per unit.**

**- From the given data the customers are buying the products with the total amount 50.**

**- The total amount which lies between 80 to 160 less sales.**

## Bivariate Analysis

**-From the above product category vs Gender Graph**

- Beauty: From the given data ,in the sixth to seventh month male customers bought more beauty products compare to females.
- Clothing: From the given data ,in the sixth to seventh month female customers are shopping clothes when compare to male customers.
- Electronics: From the given data in the seventh month the electronics products are sold more.

**-Month vs Age based on Product Category**

- From the above graph, electronics are more sold in the month of 8 by the age group of 45.
- From the above graph , clothing are more sold in the month of 5 by the age group of 45.
- From the above graph , beauty products are more sold in the month of 6 by the age group of 47.
- More shopping related to beauty , electronics , and clothing more than 45 age group people are shopping when compare to other age group people.

**-Sales by Product Category Over Months**

- Beauty: From the given data set the beauty products got a sudden decline in the month of 3 and 9 and in the remaining months it is varying constantly.
- Clothing: From the given dataset the clothing products where bought in more quantity when compare to beauty and electronics . In the month of 3 the sales has increased very much and in the month of 7 there is sudden decline in the quantity bought and slowly inclined in the month of 8.
- Electronics: From the given dataset the electronics product has been declined in the month of 3 and there is a sudden increase in the month of 5 and constantly increasing further upto 12 months.

**-Product Category vs Age**

- The data suggests there is no distinct age preference for one product category over another.
- Marketing campaigns should focus on people aged 30 to 50, as the IQR suggests that most customers fall within this range.
- The age range for all categories is consistent, from approximately 20 to 60 years.
- The interquartile range (middle 50% of the data) for all three categories is very similar.

**-Total Quantity vs Product category**

- The total quantity based on the products from the given data is:
- Beauty products are sold above 700.
- Clothing products are sold above 850.
- Electronics are sold above 800.

**-Product Category vs Total Amount**

- From the given data set the beauty products has more price and coming to the clothing the total amount is declined and for electronics it has increased gradually.

**-Month vs Quantity**

- The focus on mid-year (around months 6-8) might indicate a seasonal trend, where activity increases or decreases noticeably during these months.
- The presence of various peaks and plateaus suggests fluctuations across different months, likely indicating varying performance, sales, or expenses across the year.

**-Total Revenue per product category**

- From the given data, the Beauty and Electronics maximum amount of products lies between 90 to 900.
- The Clothing lies between 90 to 600 Total Amount.
- The interquartile range (middle 50% of the data) for all three categories is very similar.

**-Total Revenue vs Gender**

- From the given data, Males are doing more shopping on clothes than other products.
- Beauty- Females are shopping more Beauty products.
- Clothing- Males are shopping more on Clothing.
- Electronics- Males are buying more electronics ,Females are buying somewhat less than males.

**-Month Sales Trends**

- In the month of 3,9,10,11 months customers are purchasing more quantity.
- In the month of 2,5,8,12 very less sales comparing to other months.

**-Quantity vs Total Amount**

- The highest amount is 500 for the quantity 1.
- The highest amount is 1000 for the 2 quantities.
- The highest amount is 1500 for the 3 quantities.
- The highest amount is 2000 for the 4 quantities.

  As the quantity is increasing, the amount is increasing.

## Multivariate Analysis

- Strong Positive :- Total Amount and Price per Unit.
- Weak Positive:- Quantity and Total Amount.

- Weak Negative:- Quantity and Transaction Id.
- In the month of 3 clothing sales are more and electronics sales is less and the month of 5 electronics sales is more and beauty sales is less.
- In the month of 7 clothing sales is less and beauty sales is increased.
- In the month of 8 electronics sales is more and beauty,clothing sales are less.

In [ ]: