Adidas Sales Analysis

In [5]:
```
import pandas as pd
```

In [2]:
```
df=pd.read_csv(r'C:\\Users\user\Downloads\Adidas_sales.csv')
print(df)
```

```
         Retailer  Retailer ID                 Invoice Date     Region  \
0     Foot Locker      1185732    Tuesday, October 26, 2021  Northeast
1     Foot Locker      1185732  Wednesday, October 27, 2021  Northeast
2     Foot Locker      1185732   Thursday, October 28, 2021  Northeast
3     Foot Locker      1185732     Friday, October 29, 2021  Northeast
4     Foot Locker      1185732   Saturday, October 30, 2021  Northeast
...           ...          ...                          ...        ...
9643    West Gear      1128299     Saturday, March 14, 2020       West
9644    West Gear      1128299       Sunday, March 15, 2020       West
9645    West Gear      1128299       Monday, March 16, 2020       West
9646    West Gear      1128299      Tuesday, March 17, 2020       West
9647    West Gear      1128299    Wednesday, March 18, 2020       West

             State          City Gender Type   Product Category  \
0     Pennsylvania  Philadelphia         Men            Apparel
1     Pennsylvania  Philadelphia       Women            Apparel
2     Pennsylvania  Philadelphia         Men    Street Footwear
3     Pennsylvania  Philadelphia         Men  Athletic Footwear
4     Pennsylvania  Philadelphia       Women    Street Footwear
...            ...           ...         ...                ...
9643       Nevada     Las Vegas       Women            Apparel
9644       Nevada     Las Vegas         Men    Street Footwear
9645       Nevada     Las Vegas         Men  Athletic Footwear
9646       Nevada     Las Vegas       Women    Street Footwear
9647       Nevada     Las Vegas       Women  Athletic Footwear

      Price per Unit  Units Sold  Operating Profit  Operating Margin  \
0                 55         125          24062.50              0.35
1                 45         225          30375.00              0.30
2                 45         475         117562.50              0.55
3                 45         125          19687.50              0.35
4                 35         175          24500.00              0.40
...              ...         ...               ...               ...
9643              56         170           1713.60              0.18
9644              20         149           1192.00              0.40
9645              31         145           1123.75              0.25
9646              26         128           1397.76              0.42
9647              26          96            848.64              0.34

     Sales Method
0          Outlet
1          Outlet
2          Outlet
3          Outlet
4          Outlet
...           ...
9643       Outlet
9644       Outlet
9645       Outlet
9646       Outlet
9647       Outlet
```

```
[9648 rows x 13 columns]
```

In [3]:
```
df.head()
```

Out[3]:

	Retailer	Retailer ID	Invoice Date	Region	State	City	Gender Type	Product Category	Price per Unit	Unit Sol
0	Foot Locker	1185732	Tuesday, October 26, 2021	Northeast	Pennsylvania	Philadelphia	Men	Apparel	55	12
1	Foot Locker	1185732	Wednesday, October 27, 2021	Northeast	Pennsylvania	Philadelphia	Women	Apparel	45	22
2	Foot Locker	1185732	Thursday, October 28, 2021	Northeast	Pennsylvania	Philadelphia	Men	Street Footwear	45	47
3	Foot Locker	1185732	Friday, October 29, 2021	Northeast	Pennsylvania	Philadelphia	Men	Athletic Footwear	45	12
4	Foot Locker	1185732	Saturday, October 30, 2021	Northeast	Pennsylvania	Philadelphia	Women	Street Footwear	35	17

In [63]:
```
df.tail()
```

Out[63]:

	Retailer	Retailer ID	Invoice Date	Region	State	City	Gender Type	Product Category	Price per Unit	Units Sold	Operating Profit
9643	West Gear	1128299	2020-03-14	West	Nevada	Las Vegas	Women	Apparel	56.0	170	1713.60
9644	West Gear	1128299	2020-03-15	West	Nevada	Las Vegas	Men	Street Footwear	20.0	149	1192.00
9645	West Gear	1128299	2020-03-16	West	Nevada	Las Vegas	Men	Athletic Footwear	31.0	145	1123.75
9646	West Gear	1128299	2020-03-17	West	Nevada	Las Vegas	Women	Street Footwear	26.0	128	1397.76
9647	West Gear	1128299	2020-03-18	West	Nevada	Las Vegas	Women	Athletic Footwear	26.0	96	848.64

In [69]:
```
#changing to proper date time format
df['Invoice Date'] = pd.to_datetime(df['Invoice Date'])
```

In [71]:
```
#extracting year and month from date
df['Year'] = df['Invoice Date'].dt.year
df['Month'] = df['Invoice Date'].dt.month
```

In [73]:
```
#calculating profit per unit
df['Profit_per_Unit'] = df['Operating Profit'] / df['Units Sold']
```

In [74]:
```
#calculating total sales
df['Total_Sales'] = df['Price per Unit'] * df['Units Sold']
```

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

Out[9]:

	Retailer	Retailer ID	Invoice Date	Region	State	City	Gender Type	Product Category	Price per Unit	Units Sold
0	Foot Locker	1185732	2021-10-26	Northeast	Pennsylvania	Philadelphia	Men	Apparel	55	125
1	Foot Locker	1185732	2021-10-27	Northeast	Pennsylvania	Philadelphia	Women	Apparel	45	225
2	Foot Locker	1185732	2021-10-28	Northeast	Pennsylvania	Philadelphia	Men	Street Footwear	45	475
3	Foot Locker	1185732	2021-10-29	Northeast	Pennsylvania	Philadelphia	Men	Athletic Footwear	45	125
4	Foot Locker	1185732	2021-10-30	Northeast	Pennsylvania	Philadelphia	Women	Street Footwear	35	175

In [64]:
```
df.shape
```

Out[64]:
```
(9648, 17)
```

In [65]:
```
df.columns
```

Out[65]:
```
Index(['Retailer', 'Retailer ID', 'Invoice Date', 'Region', 'State', 'City',
       'Gender Type', 'Product Category', 'Price per Unit', 'Units Sold',
       'Operating Profit', 'Operating Margin', 'Sales Method', 'Year', 'Month',
       'Profit_per_Unit', 'Total_Sales'],
      dtype='object')
```

In [66]:
```
df.duplicated().sum()
```

Out[66]:
```
0
```

In [67]:
```
df.isnull().sum()
```

Out[67]:
```
Retailer            0
Retailer ID         0
Invoice Date        0
Region              0
State               0
City                0
Gender Type         0
Product Category    0
Price per Unit      0
```

```
Units Sold          0
Operating Profit    0
Operating Margin    0
Sales Method        0
Year                0
Month               0
Profit_per_Unit     4
Total_Sales         0
dtype: int64
```

In [68]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9648 entries, 0 to 9647
Data columns (total 17 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   Retailer          9648 non-null   object
 1   Retailer ID       9648 non-null   int64
 2   Invoice Date      9648 non-null   datetime64[ns]
 3   Region            9648 non-null   object
 4   State             9648 non-null   object
 5   City              9648 non-null   object
 6   Gender Type       9648 non-null   object
 7   Product Category  9648 non-null   object
 8   Price per Unit    9648 non-null   float64
 9   Units Sold        9648 non-null   int64
 10  Operating Profit  9648 non-null   float64
 11  Operating Margin  9648 non-null   float64
 12  Sales Method      9648 non-null   object
 13  Year              9648 non-null   int64
 14  Month             9648 non-null   int64
 15  Profit_per_Unit   9644 non-null   float64
 16  Total_Sales       9648 non-null   int64
dtypes: datetime64[ns](1), float64(4), int64(5), object(7)
memory usage: 1.3+ MB
```

In [76]:

```
#top 5 retailers based on sales
top_retailers_by_sales = df.groupby('Retailer')['Total_Sales'].sum().nlargest(5)
print(top_retailers_by_sales)
```

```
Retailer
West Gear        32409558.0
Foot Locker      29024945.0
Sports Direct    24616622.0
Kohl's           13512453.0
Walmart          10506085.0
Name: Total_Sales, dtype: float64
```

In [77]:

```
#popular product category
popular_product_categories = df['Product Category'].value_counts()
print(popular_product_categories)
```

```
Street Footwear      3218
Athletic Footwear    3216
Apparel              3214
Name: Product Category, dtype: int64
```

In [78]:

```
#sales trend over time
sales_trend_over_time = df.groupby('Invoice Date')['Total_Sales'].sum()
print(sales_trend_over_time)
```

```
Invoice Date
2020-01-01    119516.0
2020-01-02     97660.0
2020-01-03     90323.0
2020-01-04     88580.0
2020-01-05    100329.0
                 ...
2021-12-27     80353.0
2021-12-28     40010.0
2021-12-29     28859.0
2021-12-30     22778.0
2021-12-31     26534.0
Name: Total_Sales, Length: 724, dtype: float64
```

In [79]:

```
#sales trend over year
sales_trend_over_Year = df.groupby('Year')['Total_Sales'].sum()
print(sales_trend_over_Year)
```

```
Year
2020    24237325.0
2021    95929325.0
Name: Total_Sales, dtype: float64
```

In [80]:

```
#profit based on product category
profit_per_category = df.groupby('Product Category')['Profit_per_Unit'].mean()
print(profit_per_category)
```

```
Product Category
Apparel              93.909885
Athletic Footwear    74.850205
Street Footwear      78.781386
Name: Profit_per_Unit, dtype: float64
```

In [17]:

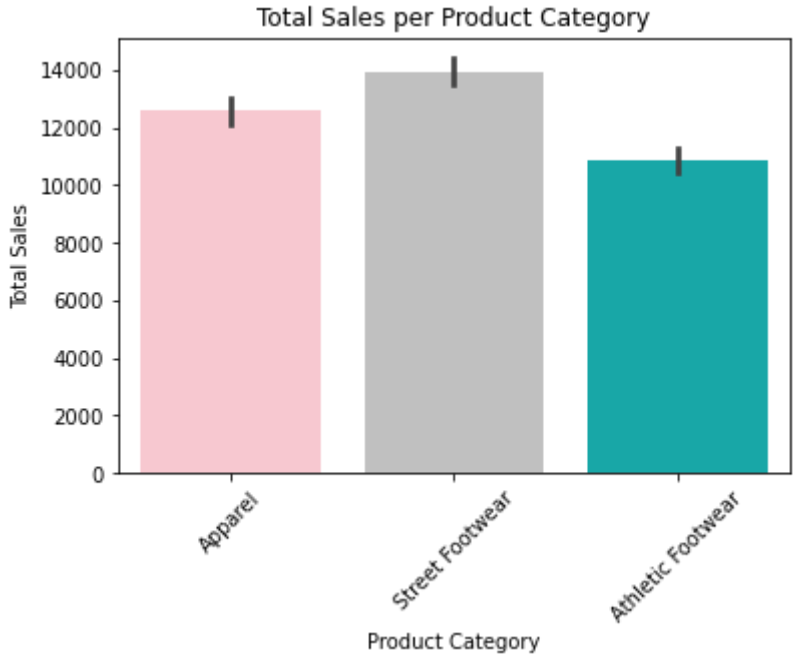
```
import seaborn as sns
import matplotlib.pyplot as plt
```

Visualization

Total Sales per Product Category

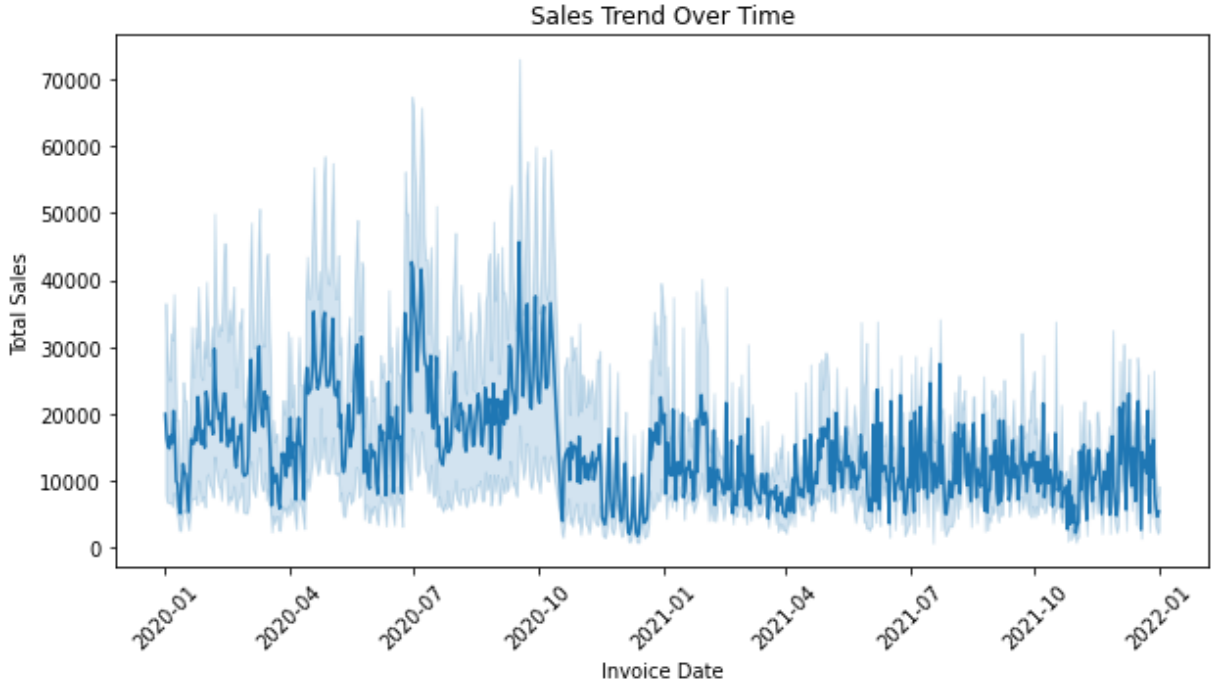
In [32]:

```
color=['pink','silver','c']
sns.barplot(x='Product Category', y='Total_Sales', data=df, palette=color)
plt.title('Total Sales per Product Category')
plt.xlabel('Product Category')
plt.ylabel('Total Sales')
plt.xticks(rotation=45)
plt.show()
```



Sales Trend Over Time

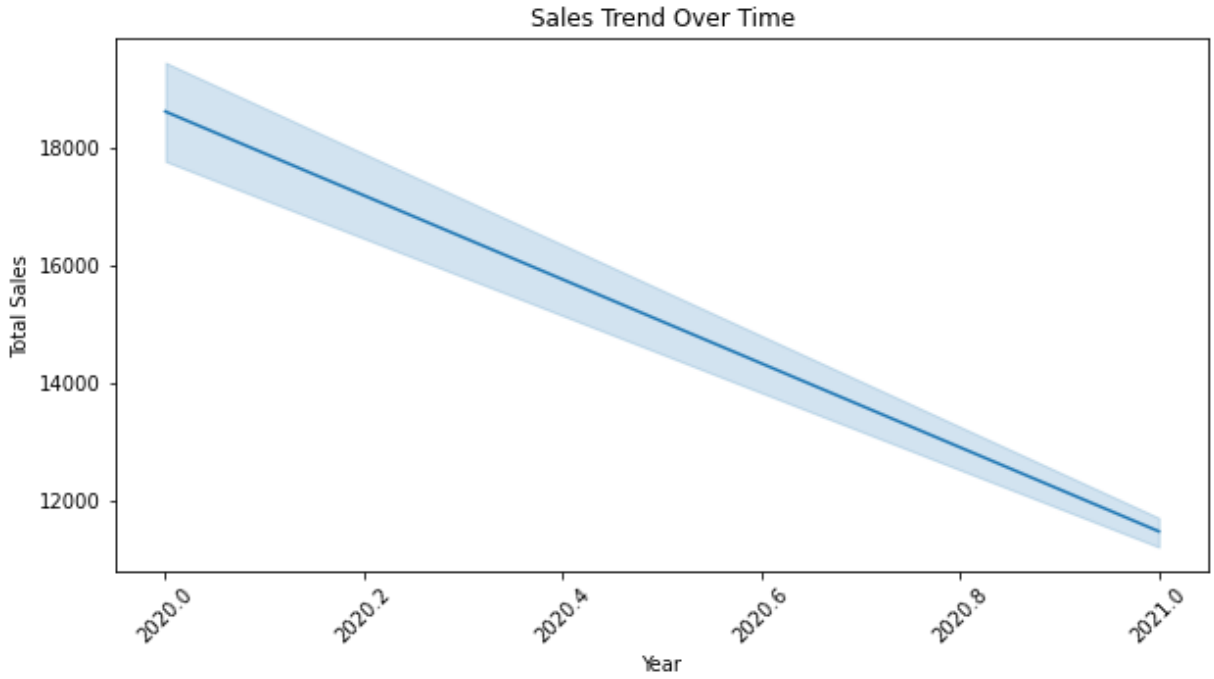
In [26]:
```
plt.figure(figsize=(10, 5))
sns.lineplot(x='Invoice Date', y='Total_Sales', data=df)
plt.title('Sales Trend Over Time')
plt.xlabel('Invoice Date')
plt.ylabel('Total Sales')
plt.xticks(rotation=45)
plt.show()
```



Sales Trend Over Year

In [35]:
```
plt.figure(figsize=(10, 5))
sns.lineplot(x='Year', y='Total_Sales', data=df)
plt.title('Sales Trend Over Year')
```

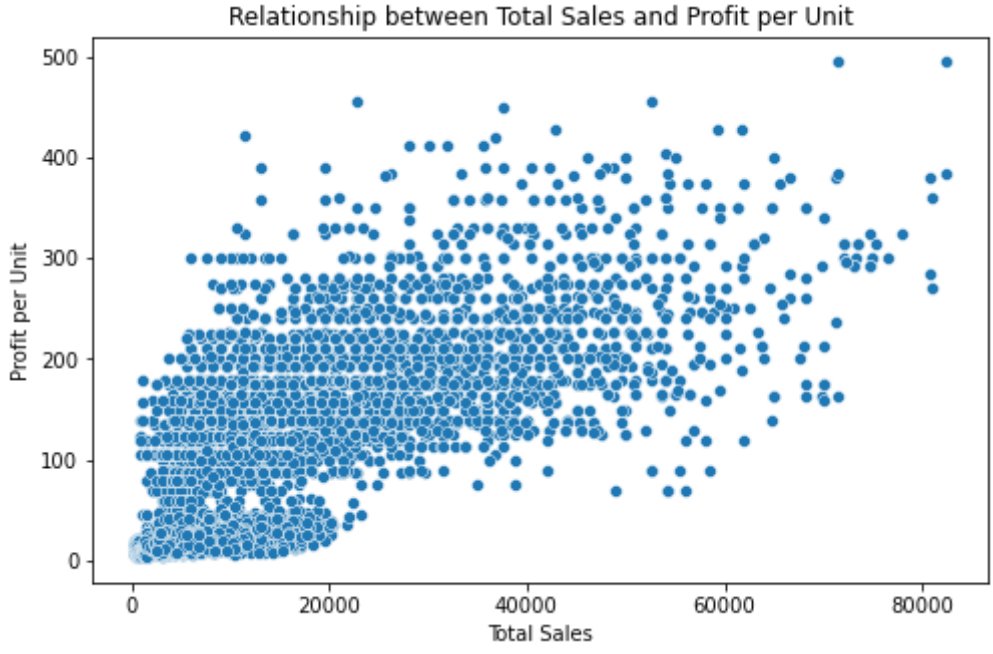
```
plt.xlabel('Year')
plt.ylabel('Total Sales')
plt.xticks(rotation=45)
plt.show()
```



Relationship between Total Sales and Profit per Unit

In [36]:

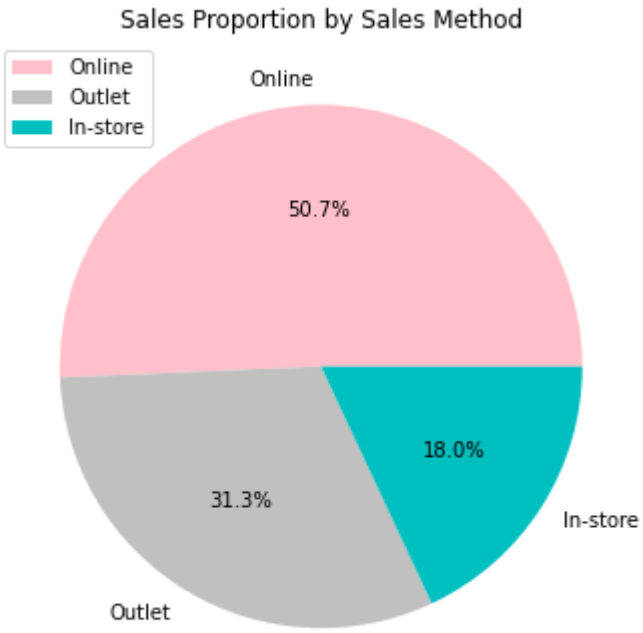
```
plt.figure(figsize=(8, 5))
sns.scatterplot(x='Total_Sales', y='Profit_per_Unit', data=df)
plt.title('Relationship between Total Sales and Profit per Unit')
plt.xlabel('Total Sales')
plt.ylabel('Profit per Unit')
plt.show()
```



Sales Proportion by Sales Method

In [42]:

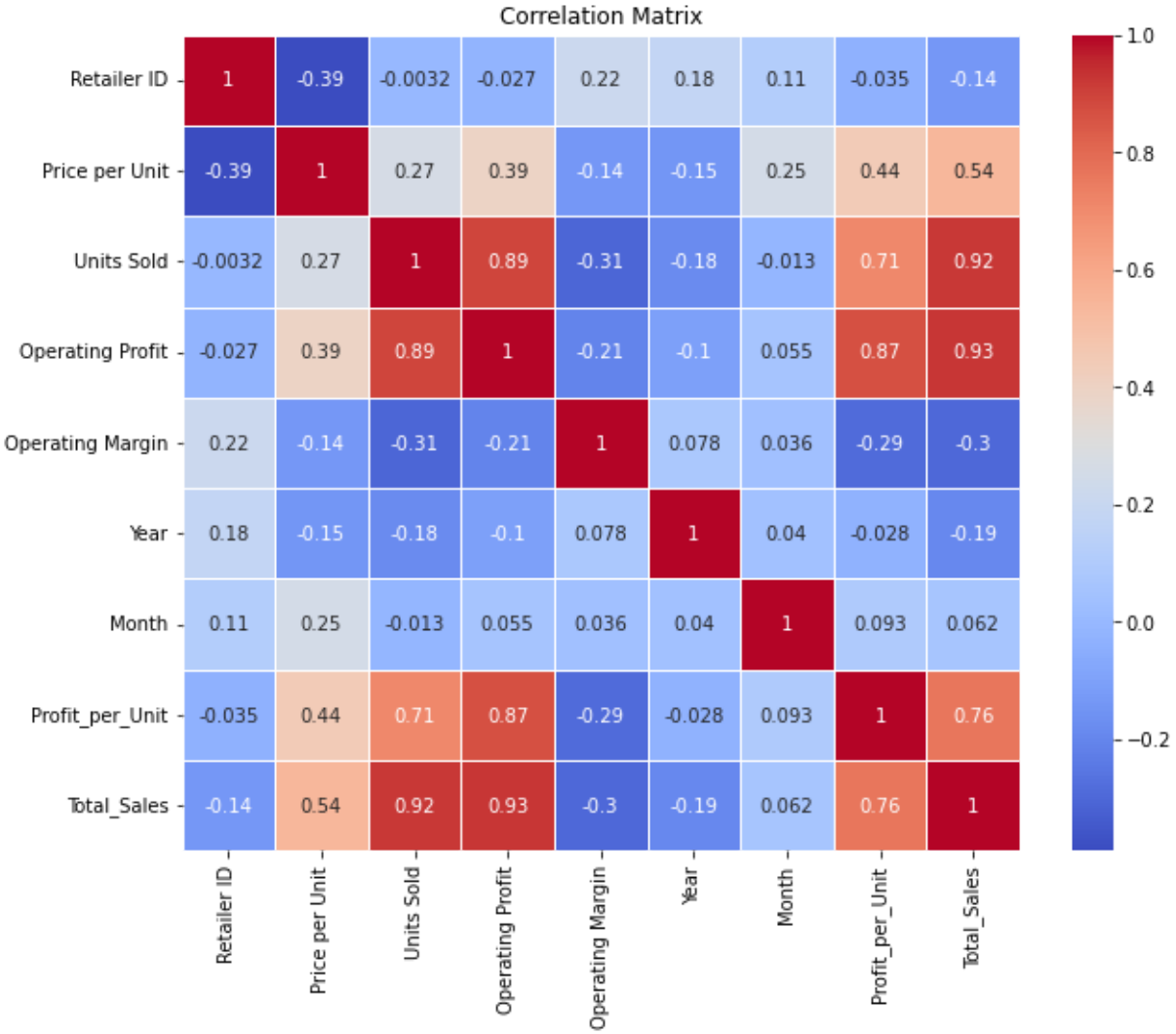
```
plt.figure(figsize=(6, 6))
color=['pink','silver','c']
df['Sales Method'].value_counts().plot(kind='pie', autopct='%1.1f%%',radius = 1,colo
plt.title('Sales Proportion by Sales Method')
plt.ylabel('')
plt.legend(loc='upper left')
plt.show()
```



Correlation Matrix

In [43]:

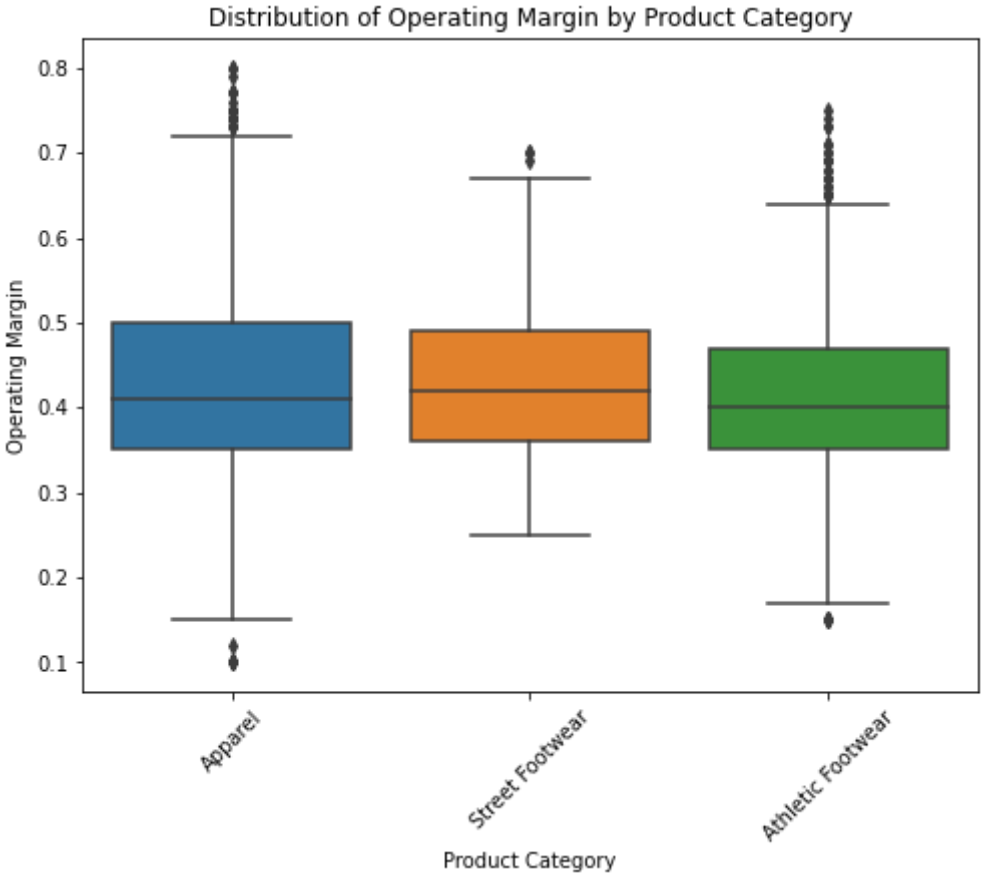
```
correlation_matrix = df.corr()
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', linewidths=0.5)
plt.title('Correlation Matrix')
plt.show()
```



Distribution of Operating Margin by Product Category

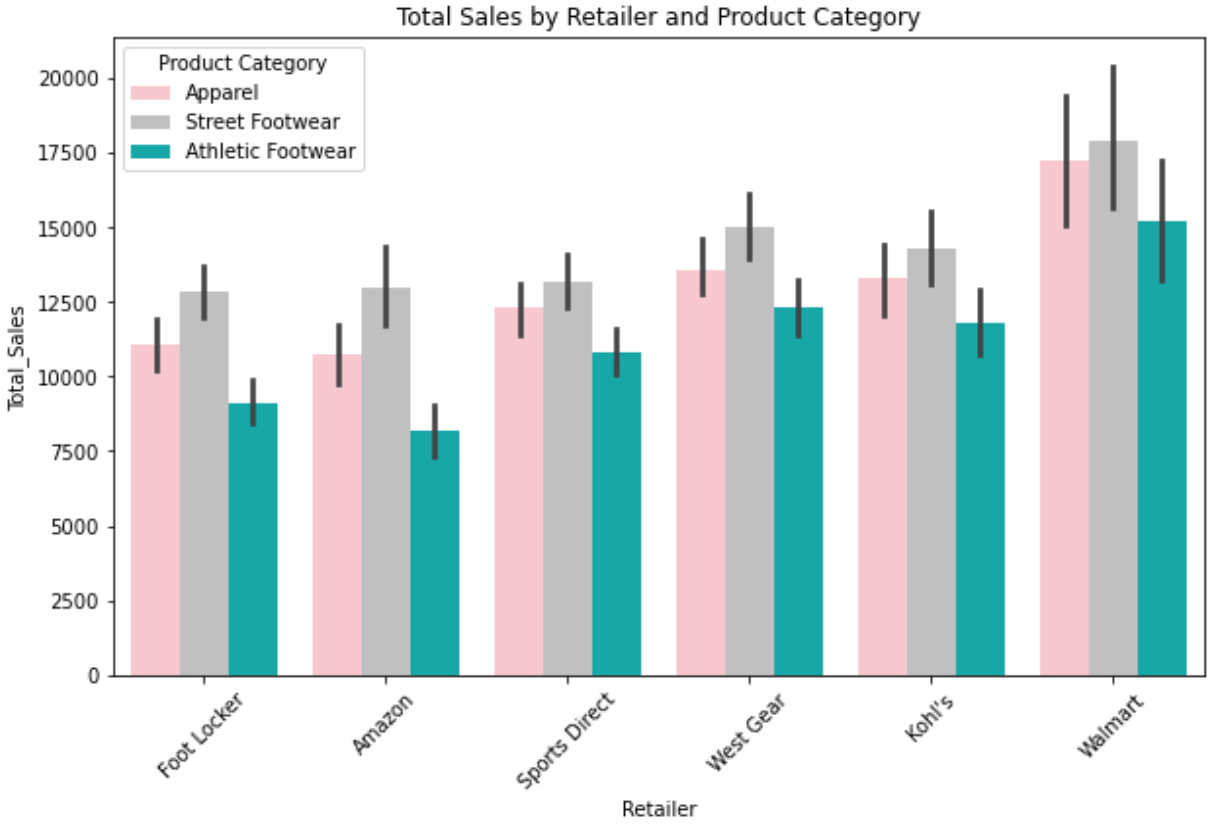
In [45]:

```
plt.figure(figsize=(8, 6))
sns.boxplot(x='Product Category', y='Operating Margin', data=df)
plt.title('Distribution of Operating Margin by Product Category')
plt.xticks(rotation=45)
plt.show()
```



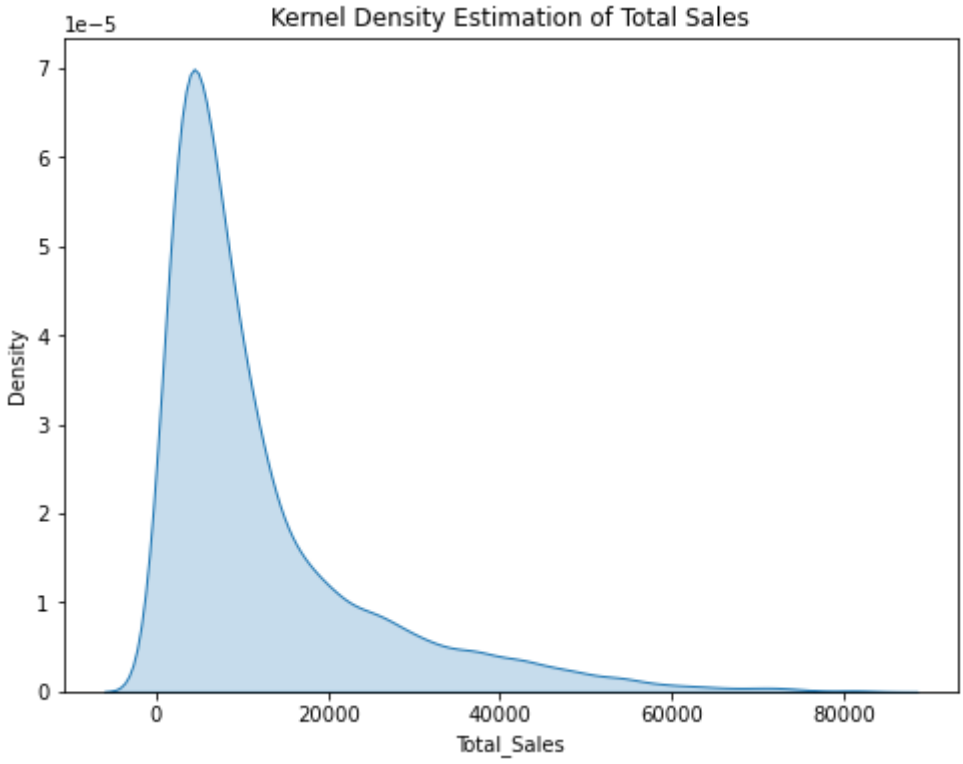
Total Sales by Retailer and Product Category

In [49]:
```
plt.figure(figsize=(10, 6))
color = {'Apparel': 'pink', 'Street Footwear': 'silver', 'Athletic Footwear': 'c'}
sns.barplot(x='Retailer', y='Total_Sales', hue='Product Category', data=df, palette=
plt.title('Total Sales by Retailer and Product Category')
plt.xticks(rotation=45)
plt.show()
```



Kernel Density Estimation

In [53]:
```
plt.figure(figsize=(8, 6))
sns.kdeplot(data=df['Total_Sales'], fill=True)
plt.title('Kernel Density Estimation of Total Sales')
plt.show()
```



In [54]:
```
import pandas as pd
from scipy import stats
```

In [55]:
```
summary_statistics = df.describe()
print(summary_statistics)
```

```
        Retailer ID  Price per Unit   Units Sold  Operating Profit  \
count  9.648000e+03     9648.000000  9648.000000       9648.000000
mean   1.173850e+06       45.216625   256.930037      34425.244761
std    2.636038e+04       14.705397   214.252030      54193.113713
min    1.128299e+06        7.000000     0.000000          0.000000
25%    1.185732e+06       35.000000   106.000000       1921.752500
50%    1.185732e+06       45.000000   176.000000       4371.420000
75%    1.185732e+06       55.000000   350.000000      52062.500000
max    1.197831e+06      110.000000  1275.000000     390000.000000

       Operating Margin         Year        Month  Profit_per_Unit  \
count       9648.000000  9648.000000  9648.000000      9644.000000
mean           0.422991  2020.865050     6.458126        82.513867
std            0.097197     0.341688     3.454799        86.810100
min            0.100000  2020.000000     1.000000         2.940000
25%            0.350000  2021.000000     3.000000        17.550000
50%            0.410000  2021.000000     6.000000        27.000000
75%            0.490000  2021.000000     9.000000       150.000000
max            0.800000  2021.000000    12.000000       495.000000

        Total_Sales
count   9648.000000
mean   12455.083955
std    12716.392111
min        0.000000
25%     4065.250000
50%     7803.500000
75%    15864.500000
max    82500.000000
```

T-statistics and P value

In [57]:
```
men_sales = df[df['Gender Type'] == 'Men']['Total_Sales']
women_sales = df[df['Gender Type'] == 'Women']['Total_Sales']
t_statistic, p_value = stats.ttest_ind(men_sales, women_sales)
print("T-statistic:", t_statistic)
print("P-value:", p_value)
```

```
T-statistic: 7.499571719264699
P-value: 6.967577148164075e-14
```

F statistics and ANOVA

In [59]:
```
regions = df['Region'].unique()
grouped_data = [df[df['Region'] == region]['Total_Sales'] for region in regions]

f_statistic, p_value_anova = stats.f_oneway(*grouped_data)
print("F-statistic:", f_statistic)
print("P-value (ANOVA):", p_value_anova)
```

```
F-statistic: 126.4861809961773
P-value (ANOVA): 2.0105836721943673e-105
```

Overall, based on the statistical results obtained from the T-test and ANOVA, we can conclude that :

1. T-test indicates a significant difference in a variable (e.g., sales or profit) between Men and Women in the dataset.
2. ANOVA reveals significant differences in a numerical variable (e.g., sales or profit) among various 'Product Category' groups.
3. Gender and 'Product Category' have a notable impact on key performance indicators (KPIs) such as sales and profitability in the Adidas sales dataset

In [ ]: