In [2]:

```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import datetime as dt
```

In [3]:

```python
var = pd.read_csv('Adidas Sales.csv')
```

In [4]:

```python
df = pd.DataFrame(var)
```

In [5]:

```python
df.head()
```

Out[5]:

| | Retailer | Retailer ID | Invoice Date | Region | State | City | Product | Price per Unit | Units Sold | Total Sales | Operating Profit | Operating Margin | Sales Method |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | Foot Locker | 1185732 | 01-01-2020 | Northeast | New York | New York | Men's Street Footwear | $50.00 | 1,200 | $6,00,000 | $3,00,000 | 50% | In-store |
| **1** | Foot Locker | 1185732 | 02-01-2020 | Northeast | New York | New York | Men's Athletic Footwear | $50.00 | 1,000 | $5,00,000 | $1,50,000 | 30% | In-store |
| **2** | Foot Locker | 1185732 | 03-01-2020 | Northeast | New York | New York | Women's Street Footwear | $40.00 | 1,000 | $4,00,000 | $1,40,000 | 35% | In-store |
| **3** | Foot Locker | 1185732 | 04-01-2020 | Northeast | New York | New York | Women's Athletic Footwear | $45.00 | 850 | $3,82,500 | $1,33,875 | 35% | In-store |
| **4** | Foot Locker | 1185732 | 05-01-2020 | Northeast | New York | New York | Men's Apparel | $60.00 | 900 | $5,40,000 | $1,62,000 | 30% | In-store |

In [6]:

```python
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9648 entries, 0 to 9647
Data columns (total 13 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   object
 3   Region            9648 non-null   object
 4   State             9648 non-null   object
 5   City              9648 non-null   object
 6   Product           9648 non-null   object
 7   Price per Unit    9648 non-null   object
 8   Units Sold        9648 non-null   object
 9   Total Sales       9648 non-null   object
 10  Operating Profit  9648 non-null   object
 11  Operating Margin  9648 non-null   object
 12  Sales Method      9648 non-null   object
dtypes: int64(1), object(12)
memory usage: 980.0+ KB
```

In [8]:

```
df.dtypes
```

Out[8]:

```
Retailer            object
Retailer ID          int64
Invoice Date        object
Region              object
State               object
City                object
Product             object
Price per Unit      object
Units Sold          object
Total Sales         object
Operating Profit    object
Operating Margin    object
Sales Method        object
dtype: object
```

In [10]:

```
df.isnull().sum()
```

Out[10]:

```
Retailer            0
Retailer ID         0
Invoice Date        0
Region              0
State               0
City                0
Product             0
Price per Unit      0
Units Sold          0
Total Sales         0
Operating Profit    0
Operating Margin    0
Sales Method        0
dtype: int64
```

In [12]:

```
df.dropna()
```

Out[12]:

| | Retailer | Retailer ID | Invoice Date | Region | State | City | Product | Price per Unit | Units Sold | Total Sales | Operating Profit | Operating Margin | Sales Method |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | Foot Locker | 1185732 | 01-01-2020 | Northeast | New York | New York | Men's Street Footwear | $50.00 | 1,200 | $6,00,000 | $3,00,000 | 50% | In-store |
| **1** | Foot Locker | 1185732 | 02-01-2020 | Northeast | New York | New York | Men's Athletic Footwear | $50.00 | 1,000 | $5,00,000 | $1,50,000 | 30% | In-store |
| **2** | Foot Locker | 1185732 | 03-01-2020 | Northeast | New York | New York | Women's Street Footwear | $40.00 | 1,000 | $4,00,000 | $1,40,000 | 35% | In-store |
| **3** | Foot Locker | 1185732 | 04-01-2020 | Northeast | New York | New York | Women's Athletic Footwear | $45.00 | 850 | $3,82,500 | $1,33,875 | 35% | In-store |
| **4** | Foot Locker | 1185732 | 05-01-2020 | Northeast | New York | New York | Men's Apparel | $60.00 | 900 | $5,40,000 | $1,62,000 | 30% | In-store |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **9643** | Foot Locker | 1185732 | 24-01-2021 | Northeast | New Hampshire | Manchester | Men's Apparel | $50.00 | 64 | $3,200 | $896 | 28% | Outlet |
| **9644** | Foot Locker | 1185732 | 24-01-2021 | Northeast | New Hampshire | Manchester | Women's Apparel | $41.00 | 105 | $4,305 | $1,378 | 32% | Outlet |
| **9645** | Foot Locker | 1185732 | 22-02-2021 | Northeast | New Hampshire | Manchester | Men's Street Footwear | $41.00 | 184 | $7,544 | $2,791 | 37% | Outlet |
| **9646** | Foot Locker | 1185732 | 22-02-2021 | Northeast | New Hampshire | Manchester | Men's Athletic Footwear | $42.00 | 70 | $2,940 | $1,235 | 42% | Outlet |
| **9647** | Foot Locker | 1185732 | 22-02-2021 | Northeast | New Hampshire | Manchester | Women's Street Footwear | $29.00 | 83 | $2,407 | $650 | 27% | Outlet |

9648 rows × 13 columns

In [13]:

```
df1 = df.iloc[[1,2,3,4,5,6,7,8,9]]
```

In [15]:

```
df1.head(10)
```

Out[15]:

| | Retailer | Retailer ID | Invoice Date | Region | State | City | Product | Price per Unit | Units Sold | Total Sales | Operating Profit | Operating Margin | Sales Method |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **1** | Foot Locker | 1185732 | 02-01-2020 | Northeast | New York | New York | Men's Athletic Footwear | $50.00 | 1,000 | $5,00,000 | $1,50,000 | 30% | In-store |
| **2** | Foot Locker | 1185732 | 03-01-2020 | Northeast | New York | New York | Women's Street Footwear | $40.00 | 1,000 | $4,00,000 | $1,40,000 | 35% | In-store |
| **3** | Foot Locker | 1185732 | 04-01-2020 | Northeast | New York | New York | Women's Athletic Footwear | $45.00 | 850 | $3,82,500 | $1,33,875 | 35% | In-store |
| **4** | Foot Locker | 1185732 | 05-01-2020 | Northeast | New York | New York | Men's Apparel | $60.00 | 900 | $5,40,000 | $1,62,000 | 30% | In-store |
| **5** | Foot Locker | 1185732 | 06-01-2020 | Northeast | New York | New York | Women's Apparel | $50.00 | 1,000 | $5,00,000 | $1,25,000 | 25% | In-store |
| **6** | Foot Locker | 1185732 | 07-01-2020 | Northeast | New York | New York | Men's Street Footwear | $50.00 | 1,250 | $6,25,000 | $3,12,500 | 50% | In-store |
| **7** | Foot Locker | 1185732 | 08-01-2020 | Northeast | New York | New York | Men's Athletic Footwear | $50.00 | 900 | $4,50,000 | $1,35,000 | 30% | Outlet |
| **8** | Foot Locker | 1185732 | 21-01-2020 | Northeast | New York | New York | Women's Street Footwear | $40.00 | 950 | $3,80,000 | $1,33,000 | 35% | Outlet |
| **9** | Foot Locker | 1185732 | 22-01-2020 | Northeast | New York | New York | Women's Athletic Footwear | $45.00 | 825 | $3,71,250 | $1,29,938 | 35% | Outlet |

In [16]:

```
df.head()
```

Out[16]:

| | Retailer | Retailer ID | Invoice Date | Region | State | City | Product | Price per Unit | Units Sold | Total Sales | Operating Profit | Operating Margin | Sales Method |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | Foot Locker | 1185732 | 01-01-2020 | Northeast | New York | New York | Men's Street Footwear | $50.00 | 1,200 | $6,00,000 | $3,00,000 | 50% | In-store |
| **1** | Foot Locker | 1185732 | 02-01-2020 | Northeast | New York | New York | Men's Athletic Footwear | $50.00 | 1,000 | $5,00,000 | $1,50,000 | 30% | In-store |
| **2** | Foot Locker | 1185732 | 03-01-2020 | Northeast | New York | New York | Women's Street Footwear | $40.00 | 1,000 | $4,00,000 | $1,40,000 | 35% | In-store |
| **3** | Foot Locker | 1185732 | 04-01-2020 | Northeast | New York | New York | Women's Athletic Footwear | $45.00 | 850 | $3,82,500 | $1,33,875 | 35% | In-store |
| **4** | Foot Locker | 1185732 | 05-01-2020 | Northeast | New York | New York | Men's Apparel | $60.00 | 900 | $5,40,000 | $1,62,000 | 30% | In-store |

In [20]:

```
df['Price per Unit'] = df['Price per Unit'].str.replace('$',"")
df['Units Sold']= df['Units Sold'].str.replace(',',"")
df['Total Sales']= df['Total Sales'].str.replace('$',"").str.replace(',','')
df['Operating Profit']= df['Operating Profit'].str.replace('$',"").str.replace(',',"")
df['Operating Margin']= df['Operating Margin'].str.replace('%',"")
```

In [21]:

```
df.head()
```

Out[21]:

| | Retailer | Retailer ID | Invoice Date | Region | State | City | Product | Price per Unit | Units Sold | Total Sales | Operating Profit | Operating Margin | Sales Method |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | Foot Locker | 1185732 | 01-01-2020 | Northeast | New York | New York | Men's Street Footwear | 50.00 | 1200 | 600000 | 300000 | 50 | In-store |
| **1** | Foot Locker | 1185732 | 02-01-2020 | Northeast | New York | New York | Men's Athletic Footwear | 50.00 | 1000 | 500000 | 150000 | 30 | In-store |
| **2** | Foot Locker | 1185732 | 03-01-2020 | Northeast | New York | New York | Women's Street Footwear | 40.00 | 1000 | 400000 | 140000 | 35 | In-store |
| **3** | Foot Locker | 1185732 | 04-01-2020 | Northeast | New York | New York | Women's Athletic Footwear | 45.00 | 850 | 382500 | 133875 | 35 | In-store |
| **4** | Foot Locker | 1185732 | 05-01-2020 | Northeast | New York | New York | Men's Apparel | 60.00 | 900 | 540000 | 162000 | 30 | In-store |

In [22]:

```
df['Price per Unit'] = df['Price per Unit'].astype('float64').astype('int64')
df['Units Sold']= df['Units Sold'].astype('int64')
df['Total Sales']= df['Total Sales'].astype('int64')
df['Operating Profit']= df['Operating Profit'].astype('int64')
df['Operating Margin']= df['Operating Margin'].astype('int64')
```

In [23]:

```
df.head()
```

Out[23]:

| | Retailer | Retailer ID | Invoice Date | Region | State | City | Product | Price per Unit | Units Sold | Total Sales | Operating Profit | Operating Margin | Sales Method |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | Foot Locker | 1185732 | 01-01-2020 | Northeast | New York | New York | Men's Street Footwear | 50 | 1200 | 600000 | 300000 | 50 | In-store |
| **1** | Foot Locker | 1185732 | 02-01-2020 | Northeast | New York | New York | Men's Athletic Footwear | 50 | 1000 | 500000 | 150000 | 30 | In-store |
| **2** | Foot Locker | 1185732 | 03-01-2020 | Northeast | New York | New York | Women's Street Footwear | 40 | 1000 | 400000 | 140000 | 35 | In-store |
| **3** | Foot Locker | 1185732 | 04-01-2020 | Northeast | New York | New York | Women's Athletic Footwear | 45 | 850 | 382500 | 133875 | 35 | In-store |
| **4** | Foot Locker | 1185732 | 05-01-2020 | Northeast | New York | New York | Men's Apparel | 60 | 900 | 540000 | 162000 | 30 | In-store |

In [24]:

```
df.loc[df['Units Sold'] >= 1000 , 'Units Segregation'] = 'High Units'
df.loc[df['Units Sold']< 1000 , 'Units Segregation'] = 'Medium Units'
df.loc[df['Units Sold'] < 500 , 'Units Segregation'] = 'Low Units'
```

In [26]:

```
df['Units Segregation'].unique()
```

Out[26]:

```
array(['High Units', 'Medium Units', 'Low Units'], dtype=object)
```

In [25]:

```
df.head()
```

Out[25]:

| | Retailer | Retailer ID | Invoice Date | Region | State | City | Product | Price per Unit | Units Sold | Total Sales | Operating Profit | Operating Margin | Sales Method | Units Segregation |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | Foot Locker | 1185732 | 01-01-2020 | Northeast | New York | New York | Men's Street Footwear | 50 | 1200 | 600000 | 300000 | 50 | In-store | High Units |
| **1** | Foot Locker | 1185732 | 02-01-2020 | Northeast | New York | New York | Men's Athletic Footwear | 50 | 1000 | 500000 | 150000 | 30 | In-store | High Units |
| **2** | Foot Locker | 1185732 | 03-01-2020 | Northeast | New York | New York | Women's Street Footwear | 40 | 1000 | 400000 | 140000 | 35 | In-store | High Units |
| **3** | Foot Locker | 1185732 | 04-01-2020 | Northeast | New York | New York | Women's Athletic Footwear | 45 | 850 | 382500 | 133875 | 35 | In-store | Medium Units |
| **4** | Foot Locker | 1185732 | 05-01-2020 | Northeast | New York | New York | Men's Apparel | 60 | 900 | 540000 | 162000 | 30 | In-store | Medium Units |

In [27]:

```
df['Invoice Date'] = pd.to_datetime(df['Invoice Date'],format = "%d-%m-%Y")
```

In [28]:

```
df['Month'] = df['Invoice Date'].dt.strftime('%B')
```

In [29]:

```
df['Weekday'] = df['Invoice Date'].dt.strftime('%A')
```

In [30]:

```
df.head()
```

Out[30]:

| | Retailer | Retailer ID | Invoice Date | Region | State | City | Product | Price per Unit | Units Sold | Total Sales | Operating Profit | Operating Margin | Sales Method | Units Segregation | Month | Weekday |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | Foot Locker | 1185732 | 2020-01-01 | Northeast | New York | New York | Men's Street Footwear | 50 | 1200 | 600000 | 300000 | 50 | In-store | High Units | January | Wednesday |
| **1** | Foot Locker | 1185732 | 2020-01-02 | Northeast | New York | New York | Men's Athletic Footwear | 50 | 1000 | 500000 | 150000 | 30 | In-store | High Units | January | Thursday |
| **2** | Foot Locker | 1185732 | 2020-01-03 | Northeast | New York | New York | Women's Street Footwear | 40 | 1000 | 400000 | 140000 | 35 | In-store | High Units | January | Friday |
| **3** | Foot Locker | 1185732 | 2020-01-04 | Northeast | New York | New York | Women's Athletic Footwear | 45 | 850 | 382500 | 133875 | 35 | In-store | Medium Units | January | Saturday |
| **4** | Foot Locker | 1185732 | 2020-01-05 | Northeast | New York | New York | Men's Apparel | 60 | 900 | 540000 | 162000 | 30 | In-store | Medium Units | January | Sunday |

# Matplotlib

In [32]:

```
df2 = df.groupby('Product').agg({'Units Sold':'sum','Total Sales':'sum','Operating Profit':'sum'}).reset_index()
```

In [33]:

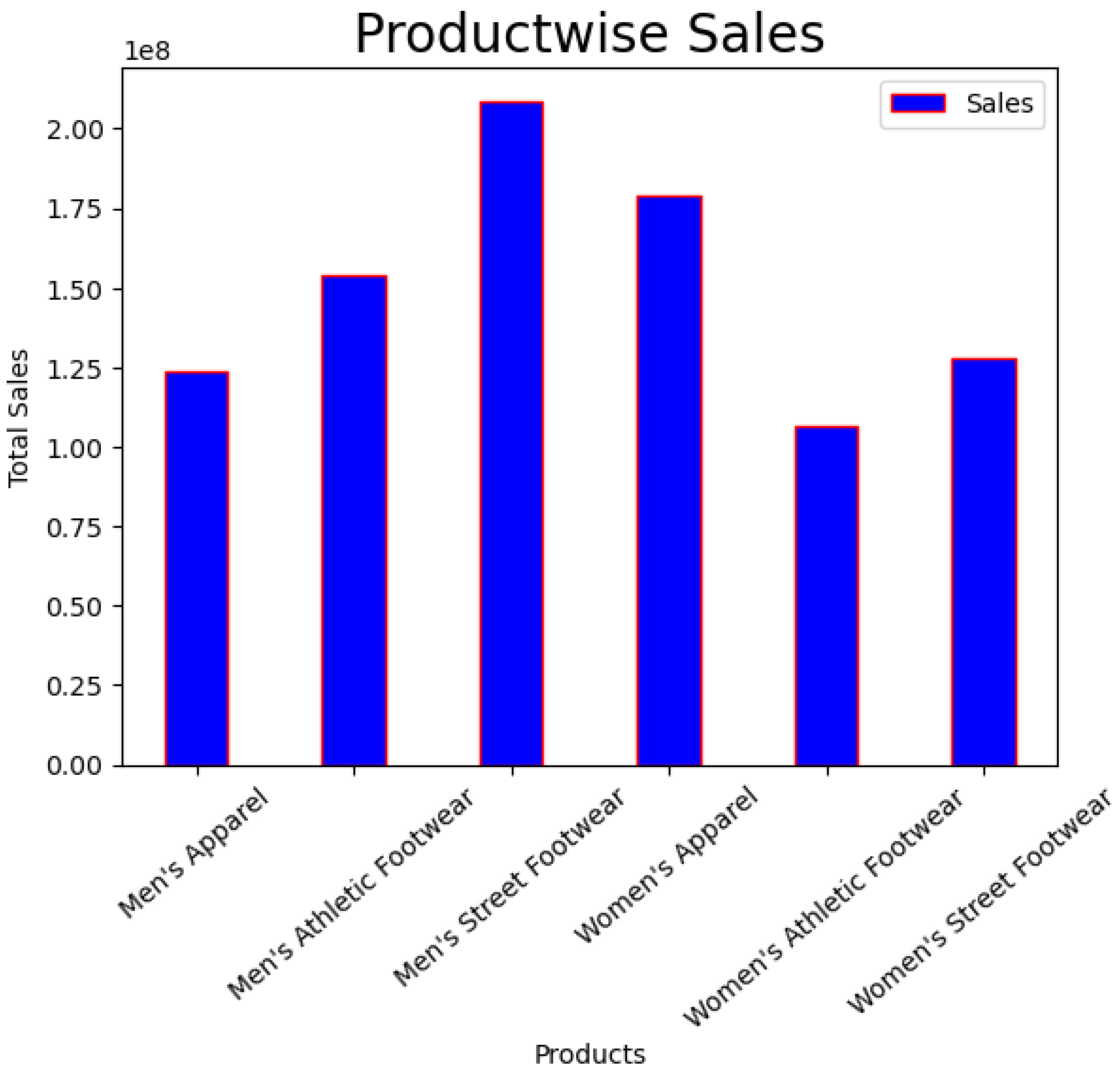
```
df2.head()
```

Out[33]:

| | Product | Units Sold | Total Sales | Operating Profit |
|---|---|---|---|---|
| **0** | Men's Apparel | 306683 | 123728632 | 44763099 |
| **1** | Men's Athletic Footwear | 435526 | 153673680 | 51846964 |
| **2** | Men's Street Footwear | 593320 | 208826244 | 82802323 |
| **3** | Women's Apparel | 433827 | 179038860 | 68650996 |
| **4** | Women's Athletic Footwear | 317236 | 106631896 | 38975843 |

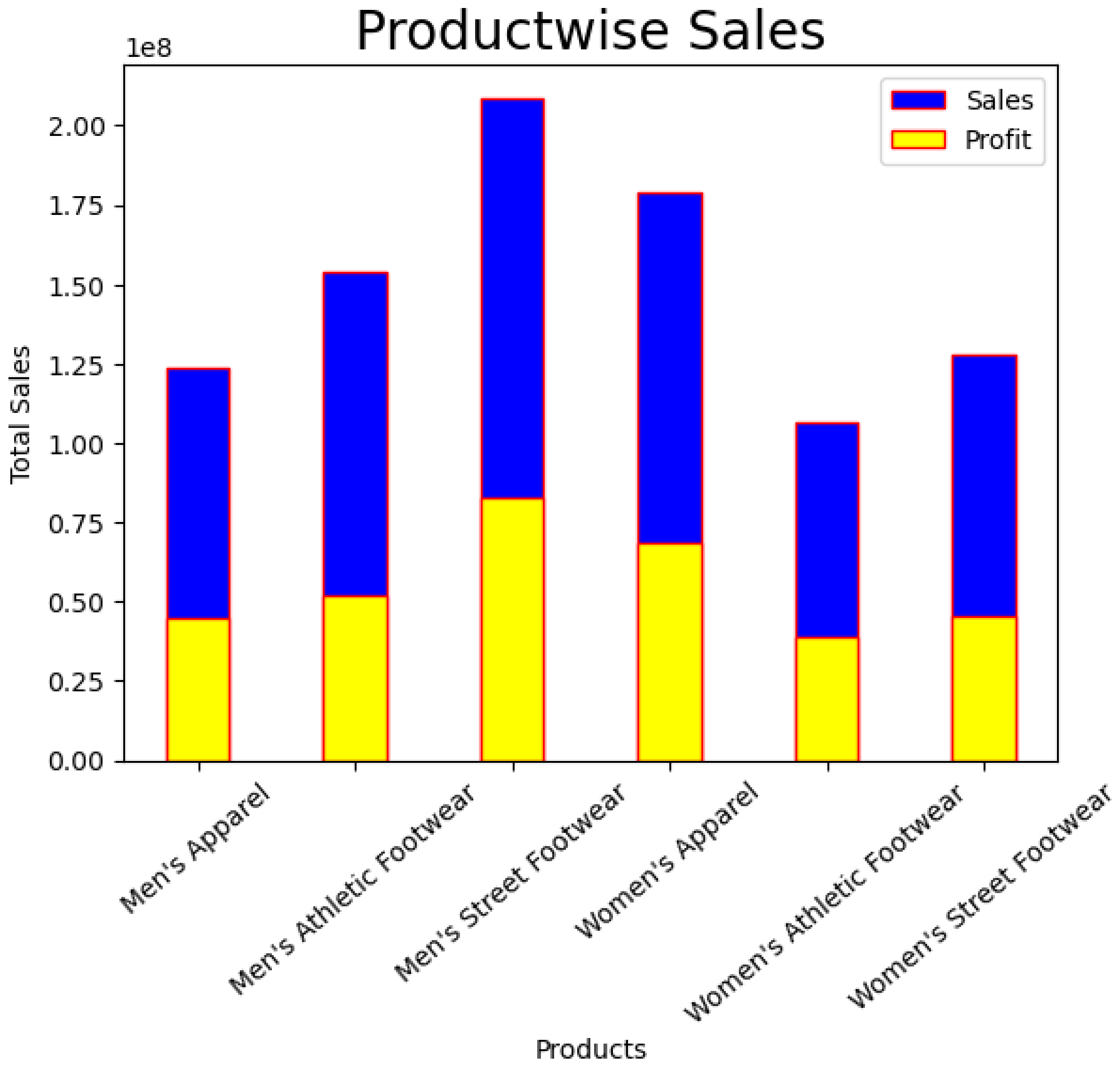
In [34]:

```
plt.title('Productwise Sales',fontsize = 20)
plt.xlabel('Products',fontsize = 10)
plt.ylabel('Total Sales',fontsize = 10)
plt.bar(df2['Product'],df2['Total Sales'],label = 'Sales',edgecolor = 'Red',color = 'Blue',width = 0.4)
plt.legend()
plt.xticks(rotation = 40)
plt.show()
```



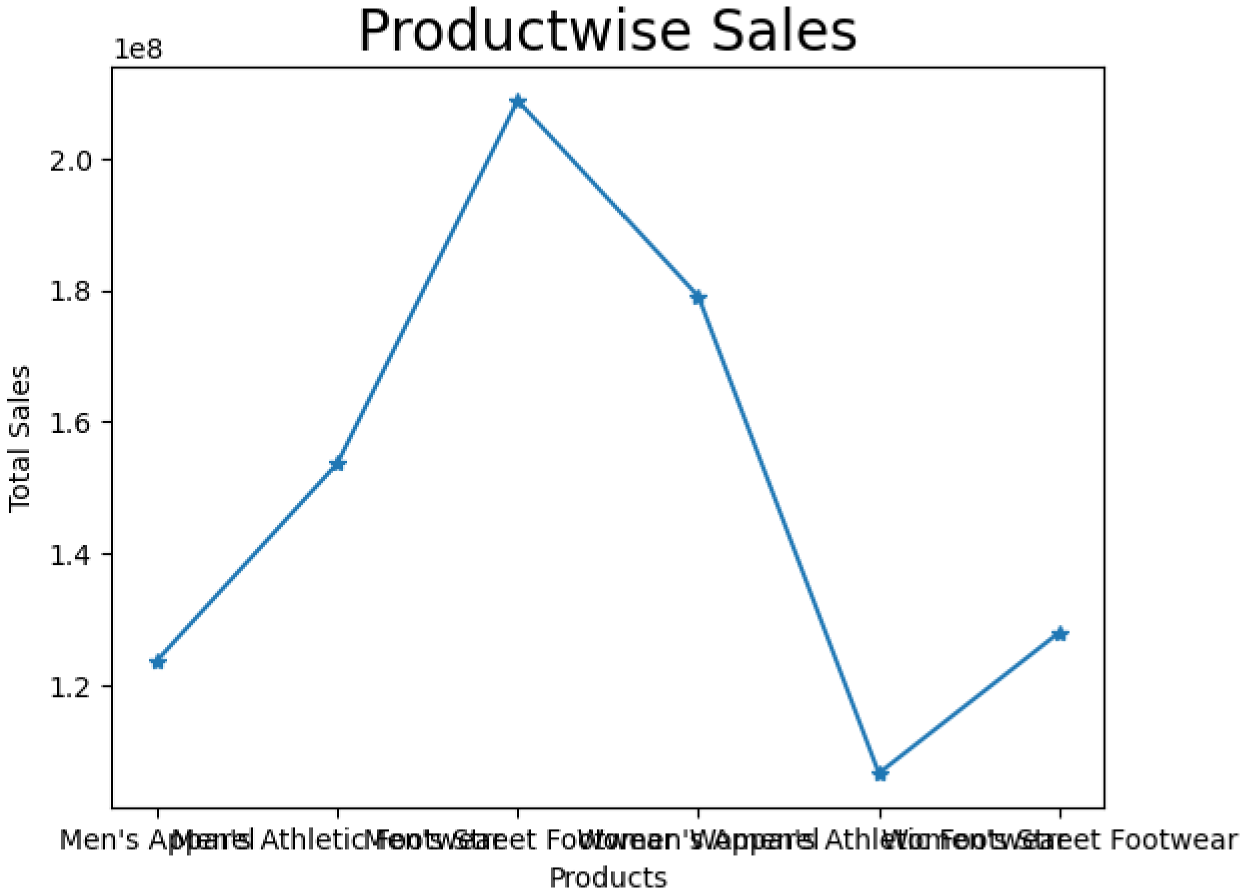
In [37]:

```
plt.title('Productwise Sales',fontsize = 20)
plt.xlabel('Products',fontsize = 10)
plt.ylabel('Total Sales',fontsize = 10)
plt.bar(df2['Product'],df2['Total Sales'],label = 'Sales',edgecolor = 'Red',color = 'Blue',width = 0.4)
plt.bar(df2['Product'],df2['Operating Profit'],label = 'Profit',edgecolor = 'Red',color = 'yellow',width = 0.4)
plt.legend()
plt.xticks(rotation = 40)
plt.show()
```



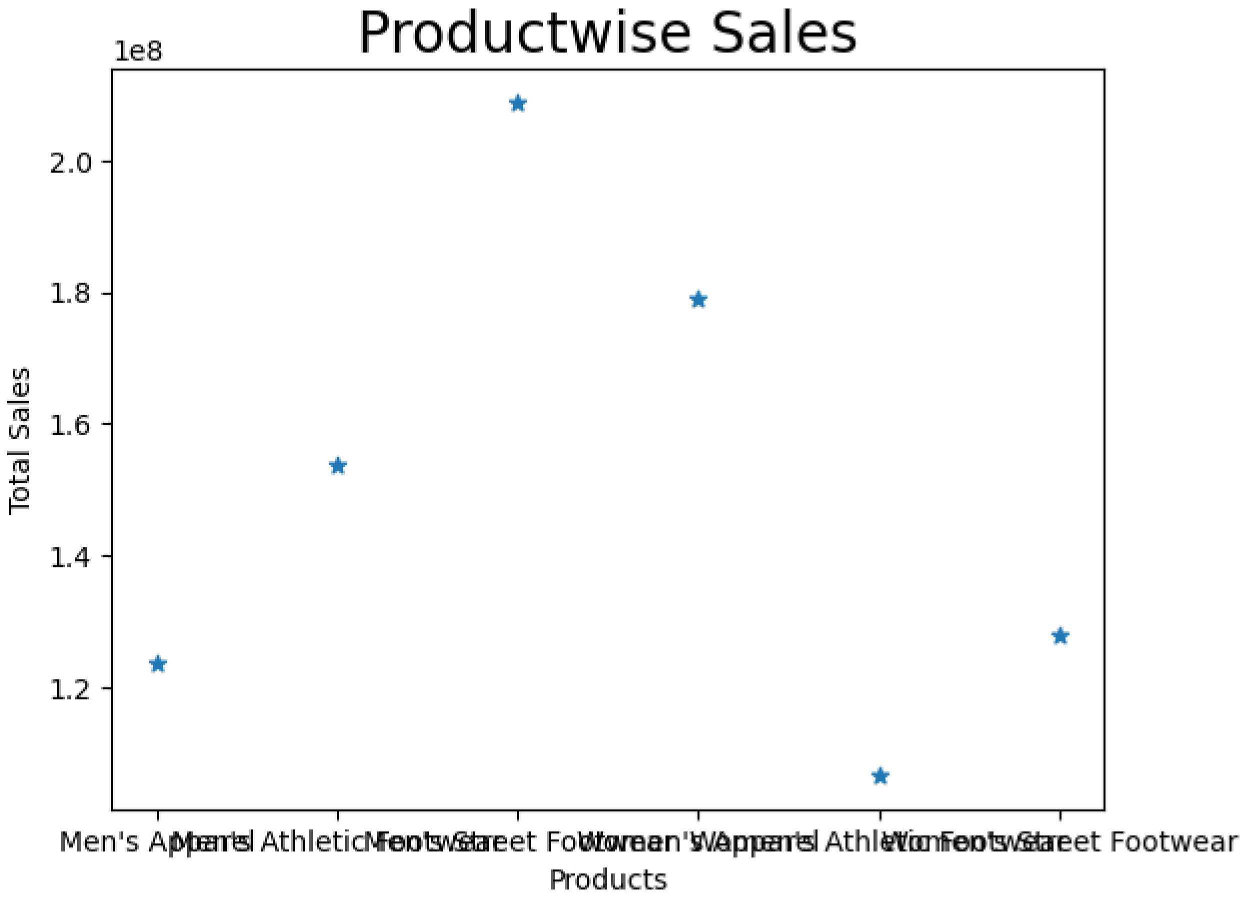
In [40]:

```
plt.title('Productwise Sales',fontsize = 20)
plt.xlabel('Products',fontsize = 10)
plt.ylabel('Total Sales',fontsize = 10)
plt.plot(df2['Product'],df2['Total Sales'],marker = "*")
plt.show()
```

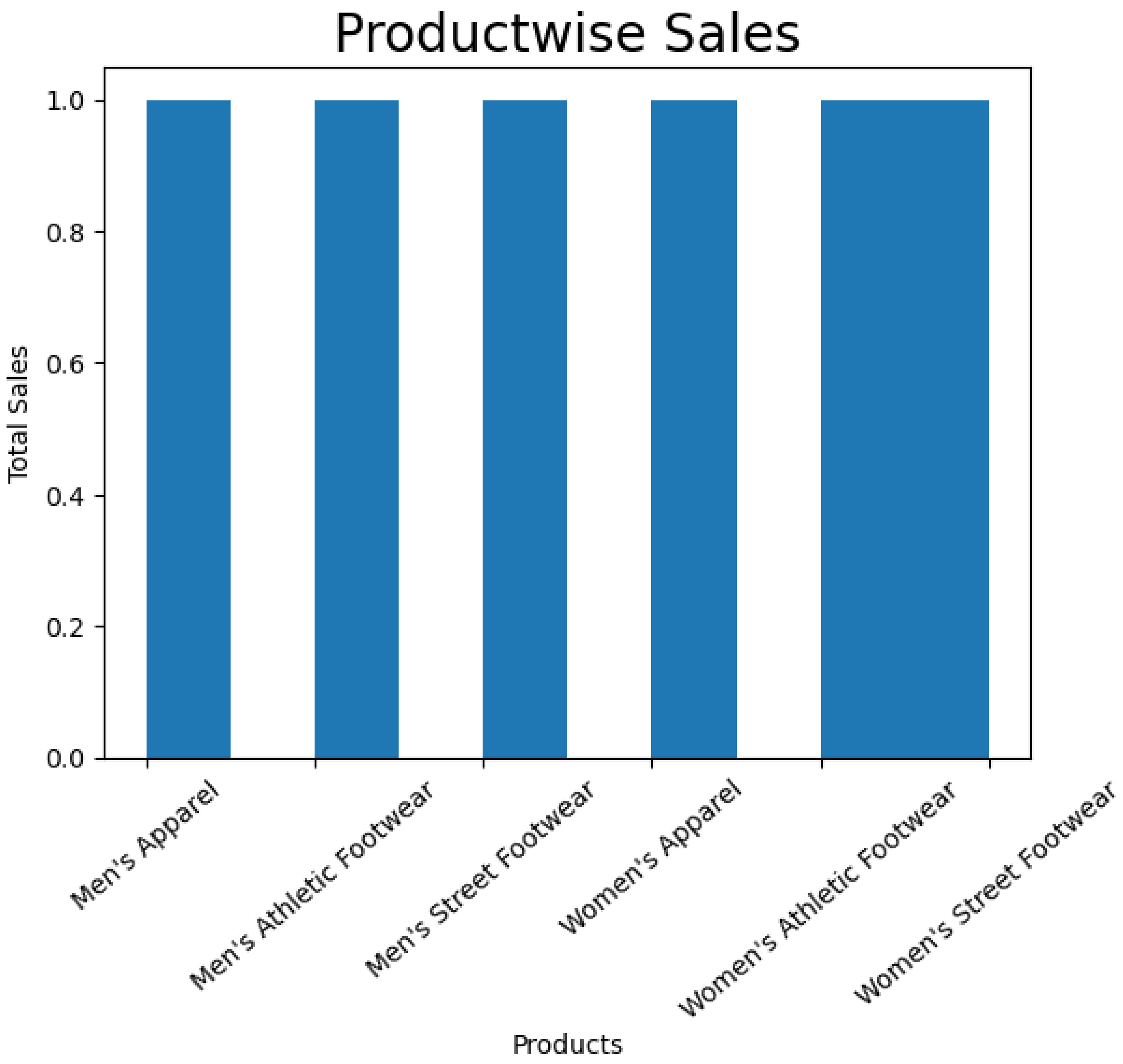


In [43]:

```
plt.title('Productwise Sales',fontsize = 20)
plt.xlabel('Products',fontsize = 10)
plt.ylabel('Total Sales',fontsize = 10)
plt.scatter(df2['Product'],df2['Total Sales'],marker = "*")
plt.show()
```

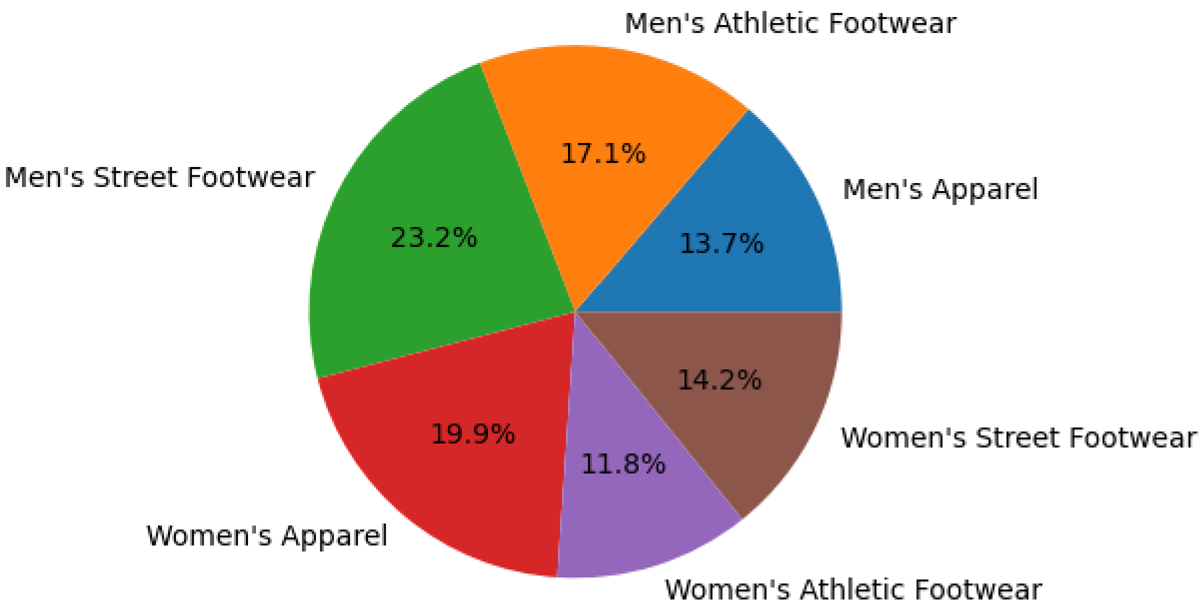


```
In [51]: plt.title('Productwise Sales',fontsize = 20)
         plt.xlabel('Products',fontsize = 10)
         plt.ylabel('Total Sales',fontsize = 10)
         plt.hist(df2['Product'])
         plt.xticks(rotation = 40)
         plt.show()
```



In [49]:

```
plt.pie(df2['Total Sales'],labels = df2['Product'],autopct = '%0.1f%%',radius = 0.9)
plt.show()
```



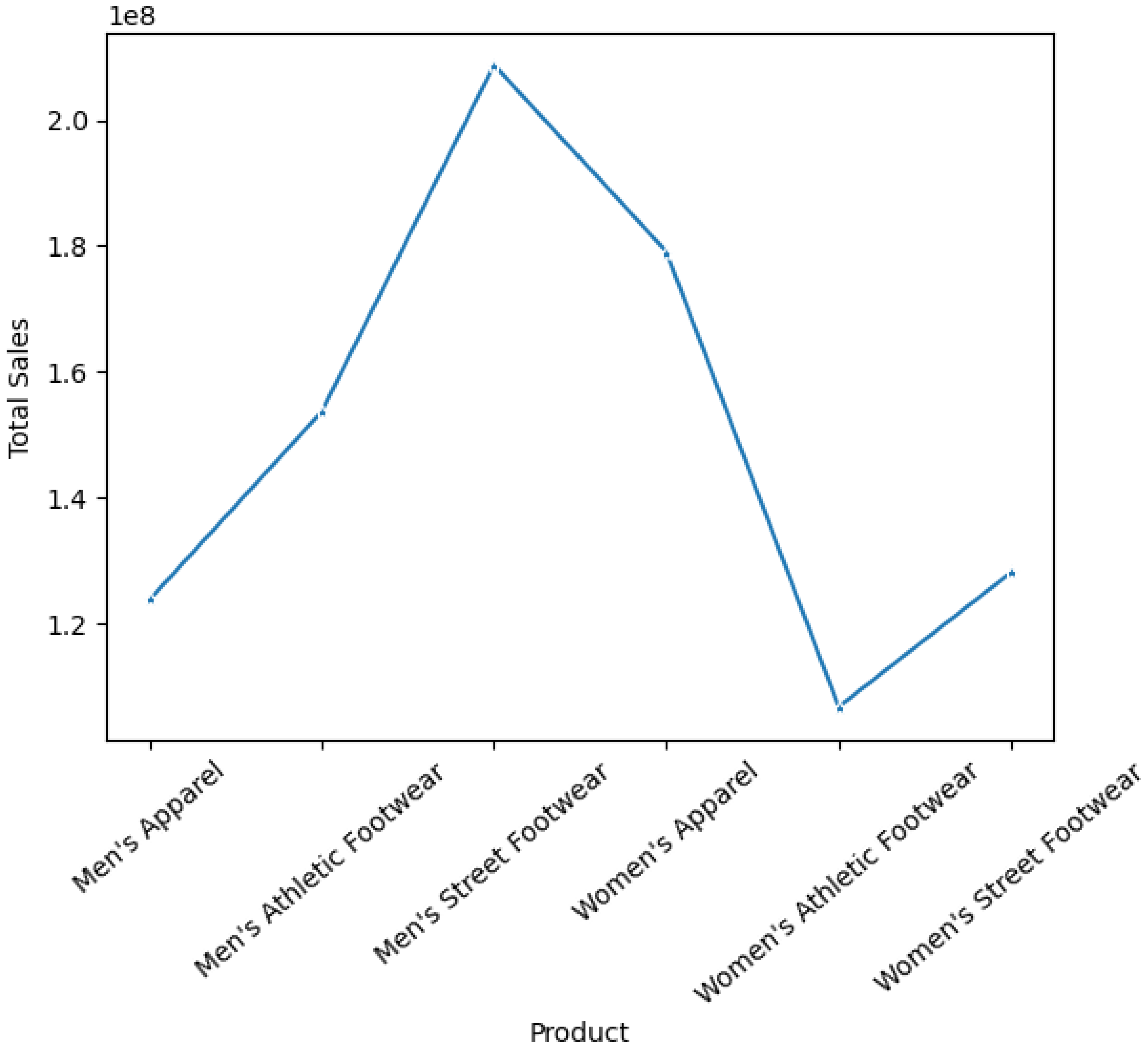
# Seaborn

In [58]:

```
sns.lineplot(x = 'Product',y = 'Total Sales',data = df2,marker = '*')
plt.xticks(rotation = 40)
```

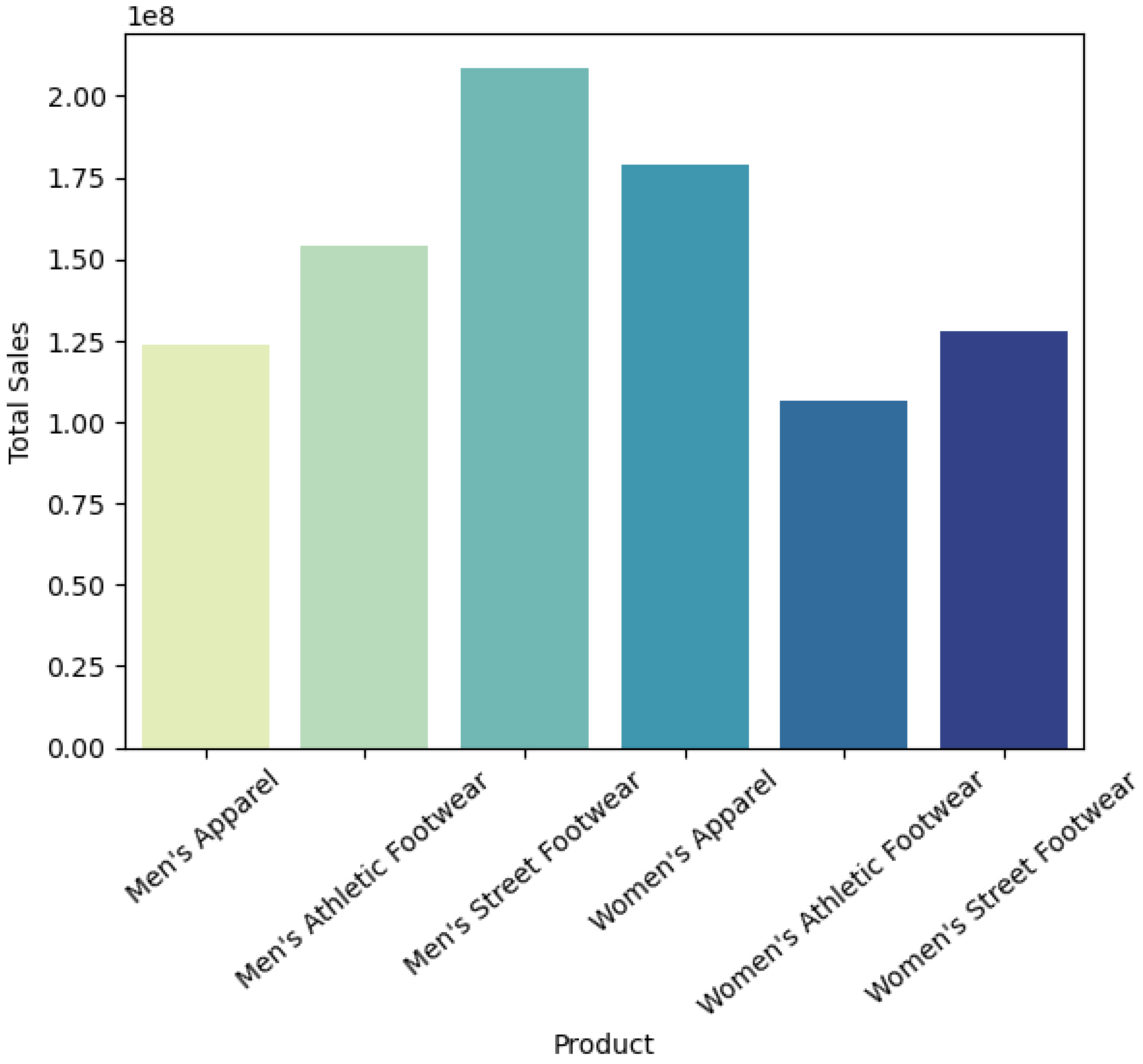
Out[58]:

```
([0, 1, 2, 3, 4, 5],
 [Text(0, 0, "Men's Apparel"),
  Text(1, 0, "Men's Athletic Footwear"),
  Text(2, 0, "Men's Street Footwear"),
  Text(3, 0, "Women's Apparel"),
  Text(4, 0, "Women's Athletic Footwear"),
  Text(5, 0, "Women's Street Footwear")])
```



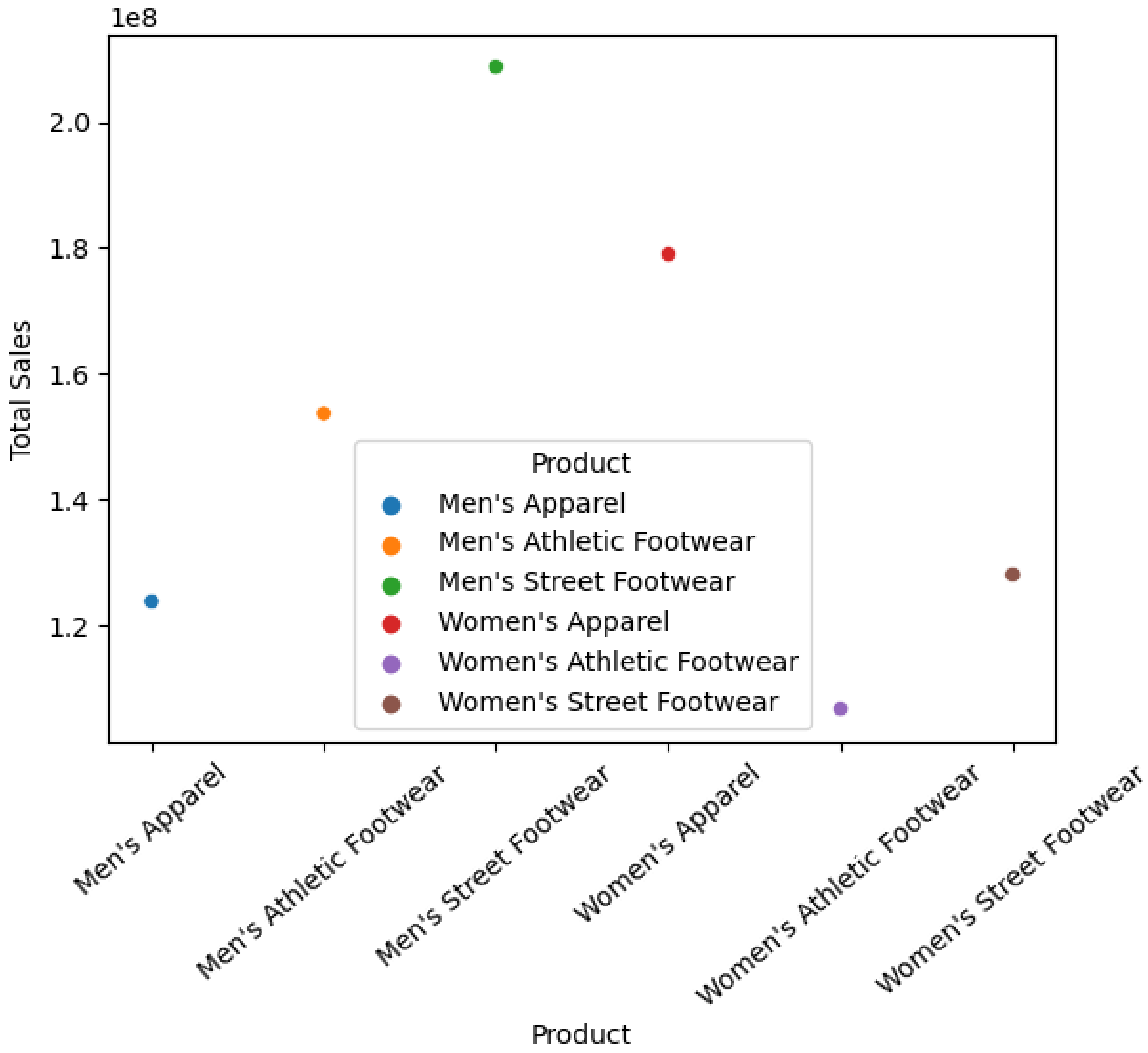
In [55]:
```
sns.barplot(x = 'Product',y = 'Total Sales',data = df2,palette = 'YlGnBu')
plt.xticks(rotation = 40)
```

```
Out[55]: (array([0, 1, 2, 3, 4, 5]),
          [Text(0, 0, "Men's Apparel"),
           Text(1, 0, "Men's Athletic Footwear"),
           Text(2, 0, "Men's Street Footwear"),
           Text(3, 0, "Women's Apparel"),
           Text(4, 0, "Women's Athletic Footwear"),
           Text(5, 0, "Women's Street Footwear")])
```



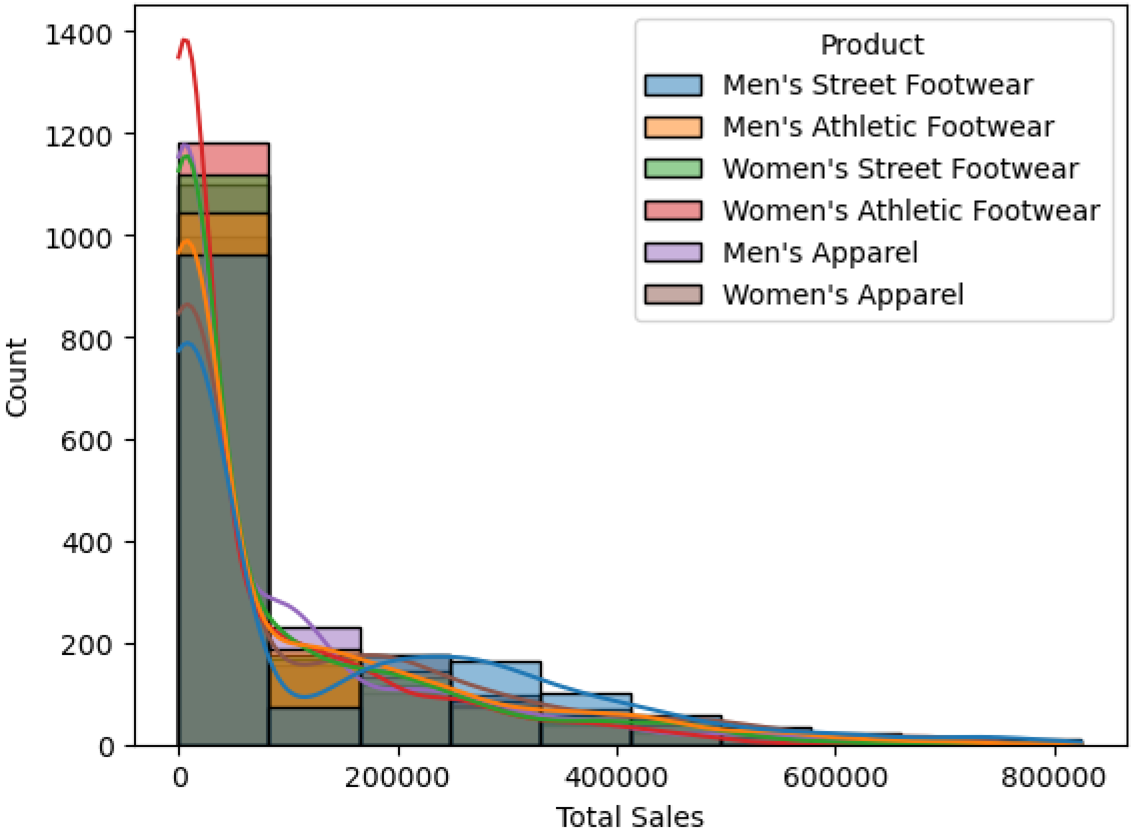
```
In [60]: sns.scatterplot(x = 'Product',y = 'Total Sales', data = df2,hue = 'Product')
         plt.xticks(rotation = 40)
```

```
Out[60]: ([0, 1, 2, 3, 4, 5],
          [Text(0, 0, "Men's Apparel"),
           Text(1, 0, "Men's Athletic Footwear"),
           Text(2, 0, "Men's Street Footwear"),
           Text(3, 0, "Women's Apparel"),
           Text(4, 0, "Women's Athletic Footwear"),
           Text(5, 0, "Women's Street Footwear")])
```



```
In [62]: sns.histplot(x = 'Total Sales',data = df,hue = 'Product',kde = True,bins = 10)
```

```
Out[62]: <Axes: xlabel='Total Sales', ylabel='Count'>
```



In [88]:

```
sns.jointplot(x = 'Total Sales',y = 'Operating Profit', data = df2,kind = 'kde',shade = True)
```

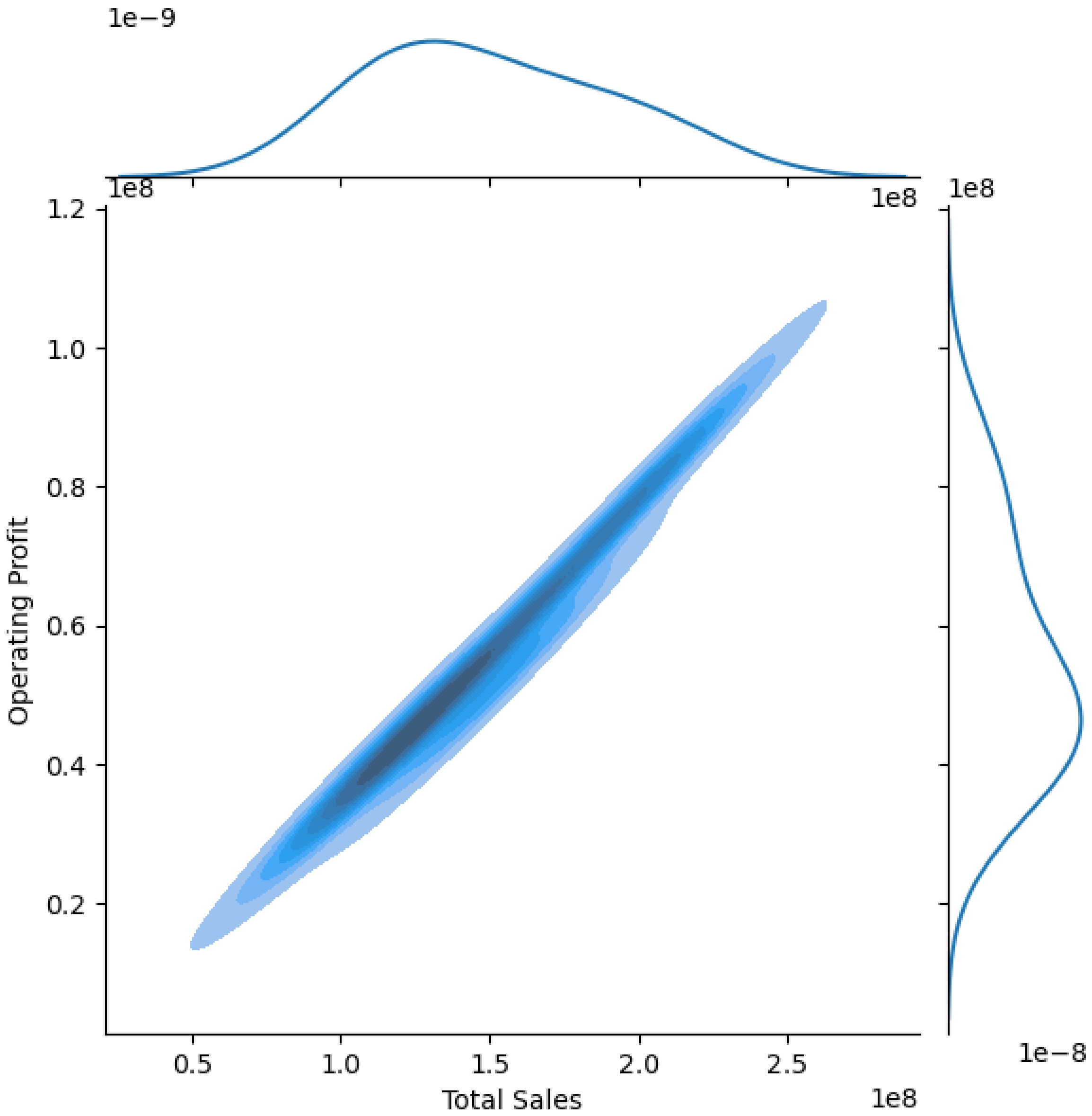
```
D:\Anaconda\Lib\site-packages\seaborn\axisgrid.py:1826: FutureWarning:

`shade` is now deprecated in favor of `fill`; setting `fill=True`.
This will become an error in seaborn v0.14.0; please update your code.

  func(x=self.x, y=self.y, **kwargs)
```

Out[88]:

```
<seaborn.axisgrid.JointGrid at 0x13479386f10>
```

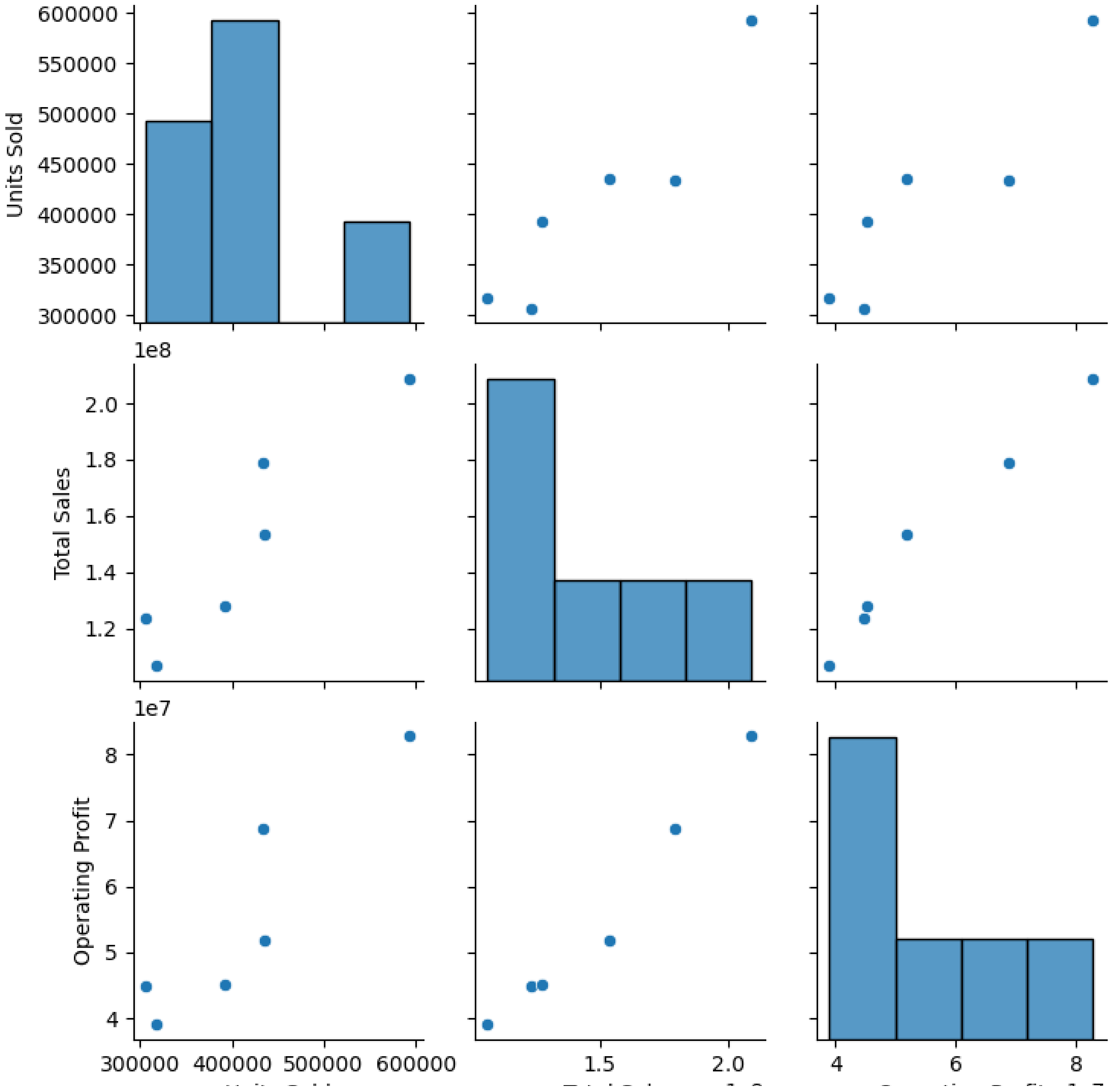


In [73]:

```
sns.pairplot(df2)
```

```
D:\Anaconda\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The figure layout has changed to tight
  self._figure.tight_layout(*args, **kwargs)
```

```
Out[73]: <seaborn.axisgrid.PairGrid at 0x134079c6e10>
```


Units Sold
600000
550000
500000
450000
400000
350000
300000
1e8
Total Sales
2.0
1.8
1.6
1.4
1.2
1e7
Operating Profit
8
7
6
5
4
300000
400000
500000
600000
1.5
2.0
4
6
8

In [74]:
```
df2.head()
```

Out[74]:

| | Product | Units Sold | Total Sales | Operating Profit |
|---|---|---|---|---|
| **0** | Men's Apparel | 306683 | 123728632 | 44763099 |
| **1** | Men's Athletic Footwear | 435526 | 153673680 | 51846964 |
| **2** | Men's Street Footwear | 593320 | 208826244 | 82802323 |
| **3** | Women's Apparel | 433827 | 179038860 | 68650996 |
| **4** | Women's Athletic Footwear | 317236 | 106631896 | 38975843 |

In [76]:
```
df2.drop(columns = ['Product'],inplace = True)
```

In [77]:
```
df2.head()
```

Out[77]:

| | Units Sold | Total Sales | Operating Profit |
|---|---|---|---|
| **0** | 306683 | 123728632 | 44763099 |
| **1** | 435526 | 153673680 | 51846964 |
| **2** | 593320 | 208826244 | 82802323 |
| **3** | 433827 | 179038860 | 68650996 |
| **4** | 317236 | 106631896 | 38975843 |

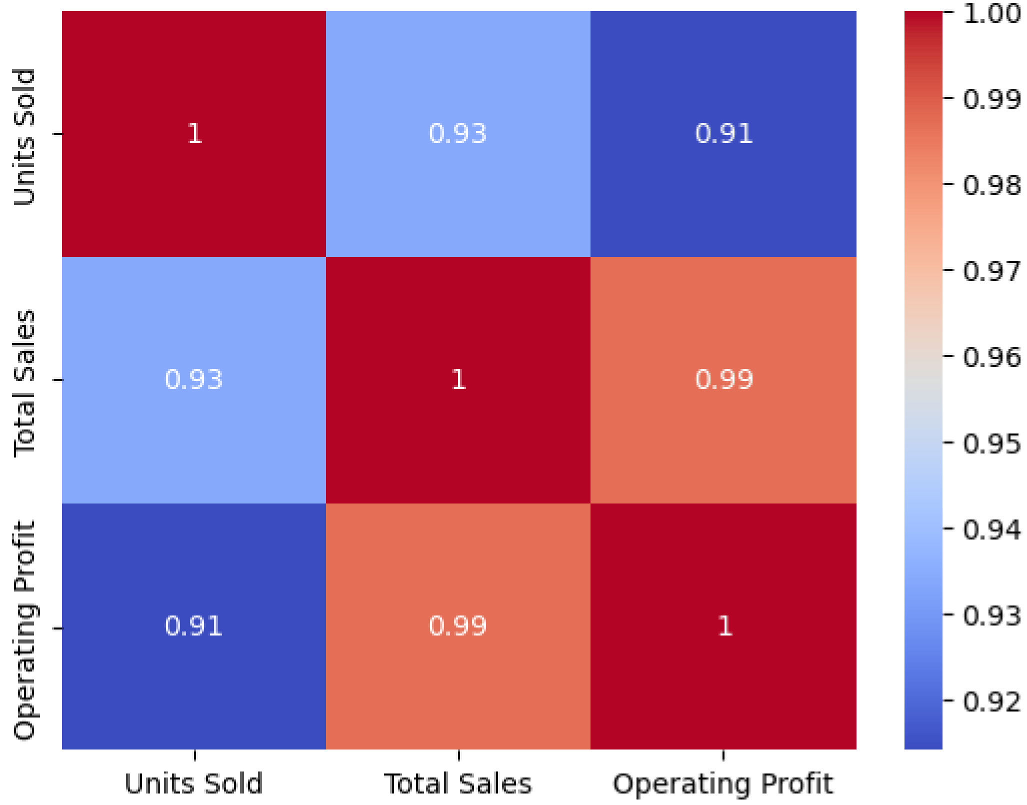
In [78]:
```
df2.corr()
```

Out[78]:

| | Units Sold | Total Sales | Operating Profit |
|---|---|---|---|
| **Units Sold** | 1.000000 | 0.933991 | 0.914135 |
| **Total Sales** | 0.933991 | 1.000000 | 0.986849 |
| **Operating Profit** | 0.914135 | 0.986849 | 1.000000 |

```
In [86]: sns.heatmap(df2.corr(),annot = True, cmap = 'coolwarm')
```

```
Out[86]: <Axes: >
```



```
In [ ]:
```