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

```
amazon_sales_data = pd.read_csv("/content/Amazon Sales data.csv")
```

```
amazon_sales_data.head()
```

	Region	Country	Item Type	Sales Channel	Order Priority	Order Date	Order ID	Ship Date	Units Sold
0	Australia and Oceania	Tuvalu	Baby Food	Offline	H	5/28/2010	669165933	6/27/2010	9925
1	Central America and the Caribbean	Grenada	Cereal	Online	C	8/22/2012	963881480	9/15/2012	2804
2	Europe	Russia	Office Supplies	Offline	L	5/2/2014	341417157	5/8/2014	1779
3	Sub-Saharan	Sao Tome and	Fruits	Online	C	6/20/2014	514321792	7/5/2014	8102

Next steps: Generate code with amazon_sales_data View recommended plots

```
# Convert 'Order Date' and 'Ship Date' to datetime format
amazon_sales_data['Order Date'] = pd.to_datetime(amazon_sales_data['Order Date'], format='%m/%d/%Y')
amazon_sales_data['Ship Date'] = pd.to_datetime(amazon_sales_data['Ship Date'], format='%m/%d/%Y')
```

```
# Extract month and year from 'Order Date'
amazon_sales_data['Order Month'] = amazon_sales_data['Order Date'].dt.month
amazon_sales_data['Order Year'] = amazon_sales_data['Order Date'].dt.year
```

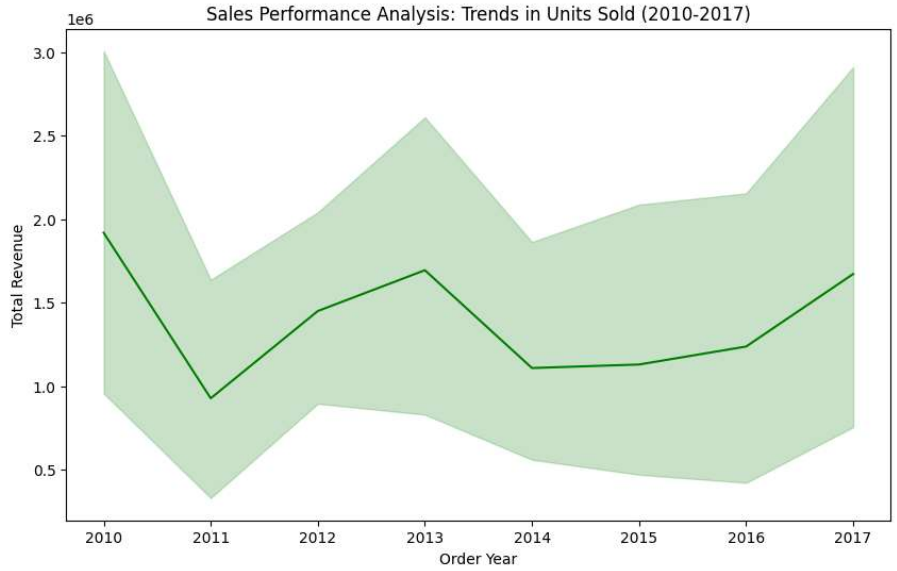
```
# 1. Monthly Sales Trend
monthly_sales_trend = amazon_sales_data.groupby('Order Month')['Total Revenue'].sum()
```

```
# 2. Yearly Sales Trend
yearly_sales_trend = amazon_sales_data.groupby('Order Year')['Total Revenue'].sum()
```

```
# 3. Yearly Month-wise Sales Trend
yearly_monthly_sales_trend = amazon_sales_data.groupby(['Order Year', 'Order Month'])['Total Revenue'].sum().unstack()
```

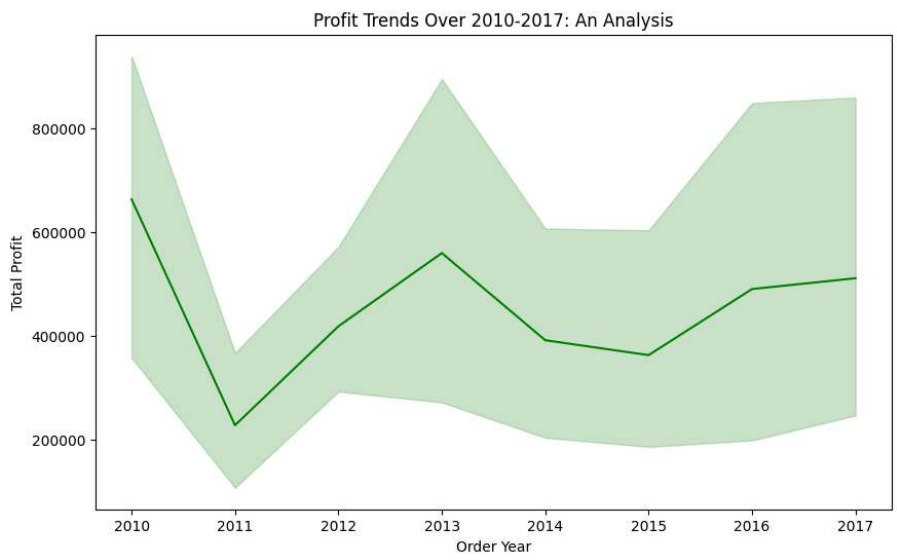
Sales Performance Analysis

```
plt.figure(figsize=(10,6))
sns.lineplot(data =amazon_sales_data, x = amazon_sales_data['Order Year'], y = amazon_sales_data['Total Revenue'], color = 'green')
plt.title("Sales Performance Analysis: Trends in Units Sold (2010-2017)")
plt.show()
```



Profit Trends Over 2010-17

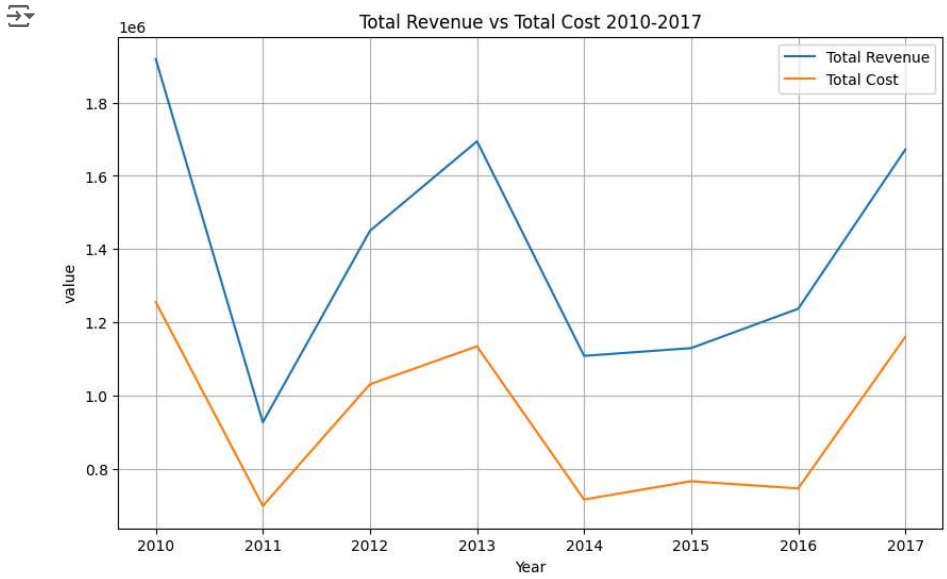
```
plt.figure(figsize=(10,6))
sns.lineplot(data = amazon_sales_data, x = amazon_sales_data['Order Year'], y = amazon_sales_data['Total Profit'], color='green')
plt.title('Profit Trends Over 2010-2017: An Analysis')
plt.show()
```



Total Revenue vs Total Cost 2010-2017

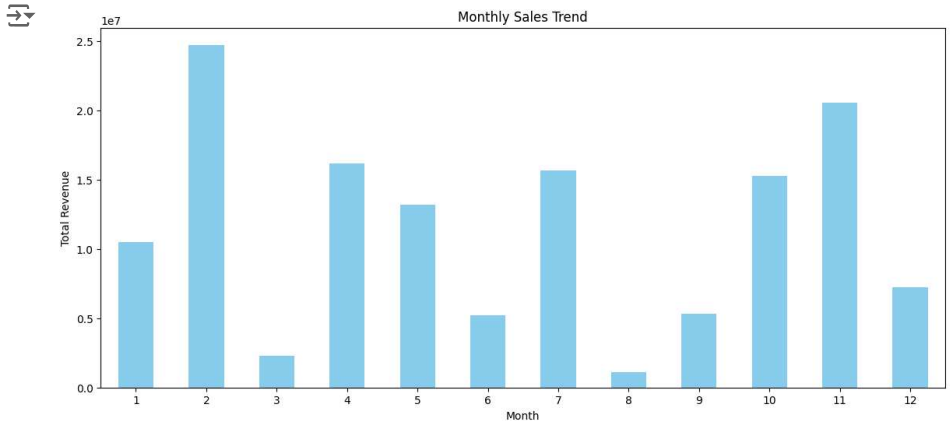
```
data_preproc = pd.DataFrame({
    'Year': amazon_sales_data['Order Year'],
    'Total Revenue': amazon_sales_data['Total Revenue'],
    'Total Cost' :amazon_sales_data['Total Cost']
    })
```

```
plt.figure(figsize=(10,6))
sns.lineplot(x='Year', y='value' , hue='variable', data=pd.melt(data_preproc, ['Year']), errorbar=None)
plt.legend()
plt.grid()
plt.title("Total Revenue vs Total Cost 2010-2017")
plt.show()
```



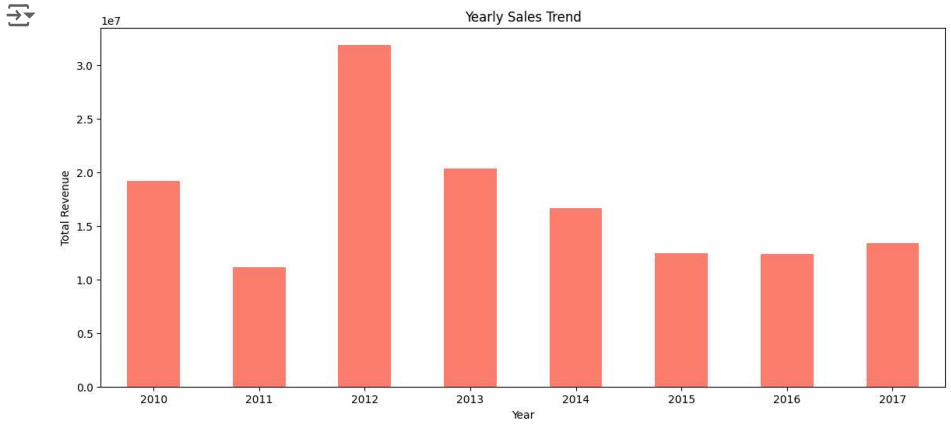
Monthly Sales

```
# Monthly Sales Trend
plt.figure(figsize=(14, 6))
monthly_sales_trend.plot(kind='bar', color='skyblue')
plt.title('Monthly Sales Trend')
plt.xlabel('Month')
plt.ylabel('Total Revenue')
plt.xticks(rotation=0)
plt.show()
```



Yearly Sales

```
# Yearly Sales Trend
plt.figure(figsize=(14, 6))
yearly_sales_trend.plot(kind='bar', color='salmon')
plt.title('Yearly Sales Trend')
plt.xlabel('Year')
plt.ylabel('Total Revenue')
plt.xticks(rotation=0)
plt.show()
```

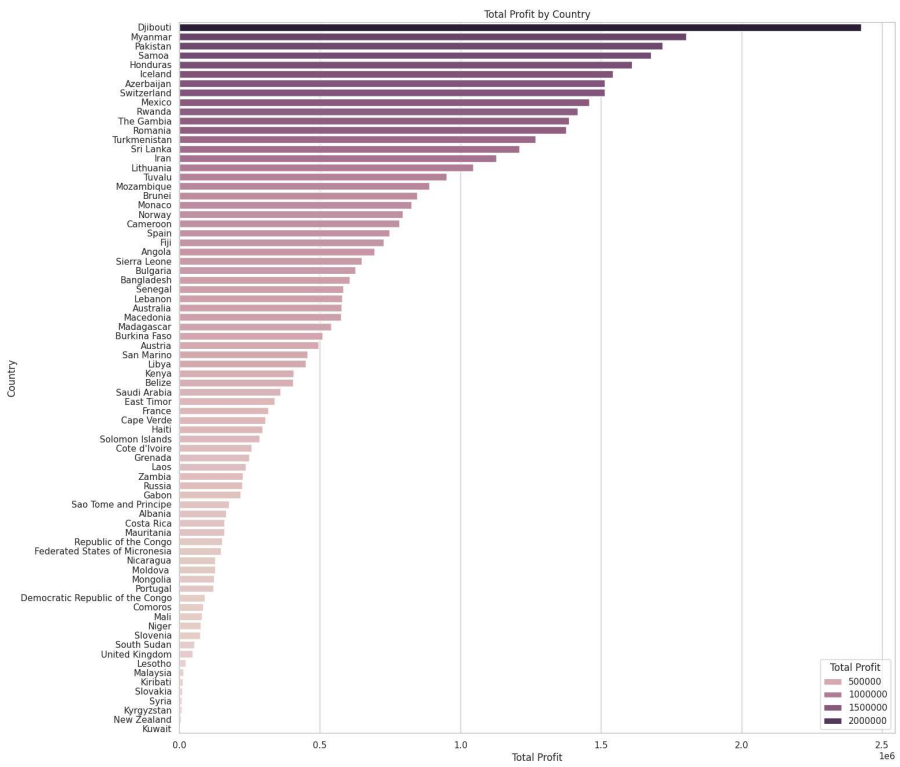



```
CountryProfit = amazon_sales_data.groupby('Country')['Total Profit'].sum().reset_index()
CountryProfit = CountryProfit.sort_values(by = 'Total Profit', ascending=False).reset_index()
CountryProfit.head()
```

	index	Country	Total Profit
0	16	Djibouti	2425317.87
1	46	Myanmar	1802771.70
2	51	Pakistan	1719922.04
3	57	Samoa	1678540.98
4	24	Honduras	1609947.52

Next steps: Generate code with CountryProfit View recommended plots

```
plt.figure(figsize=(16, 16))
sns.set_palette("pastel")
sns.set(style="whitegrid")
plt.title('Total Profit by Country')
sns.barplot(x='Total Profit', y='Country', data=CountryProfit, errorbar=None, hue = 'Total Profit')
plt.show()
```

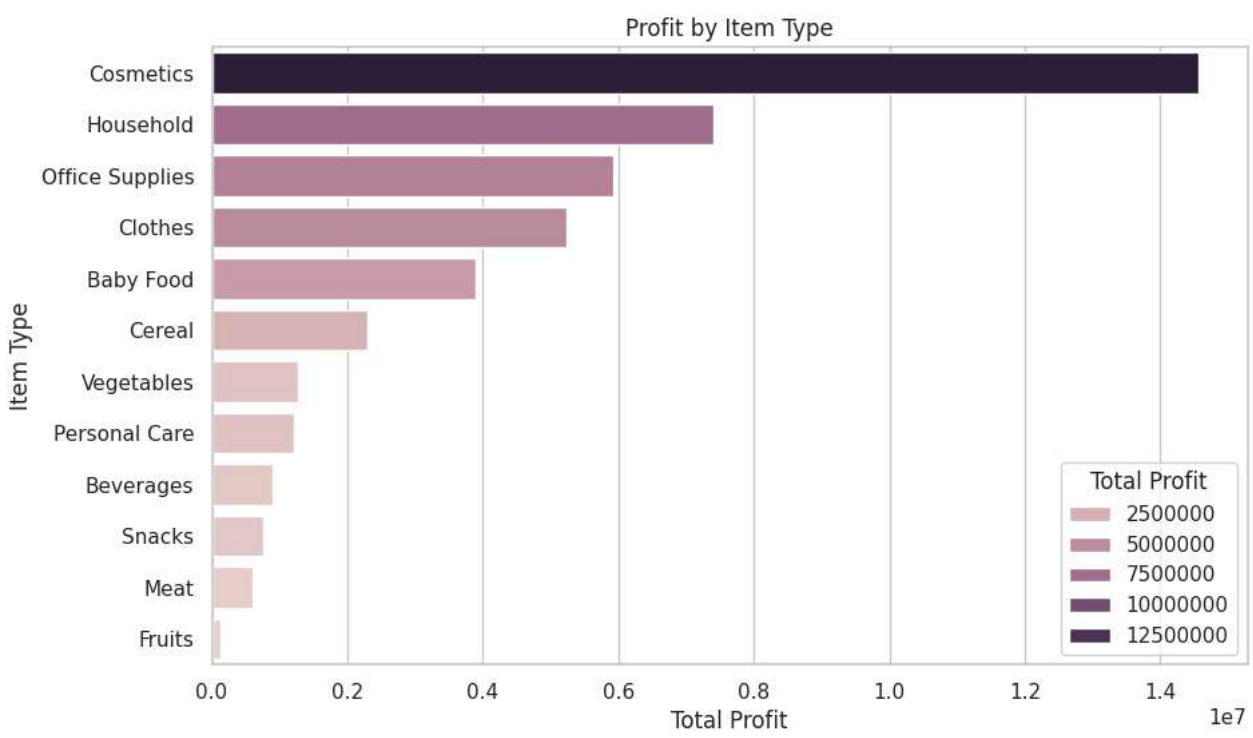


```
ProductProfit = amazon_sales_data.groupby('Item Type')['Total Profit'].sum().sort_values(ascending=False).reset_index()
ProductProfit.head()
```

	Item Type	Total Profit
0	Cosmetics	14556048.66
1	Household	7412605.71
2	Office Supplies	5929583.75
3	Clothes	5233334.40
4	Baby Food	3886643.70

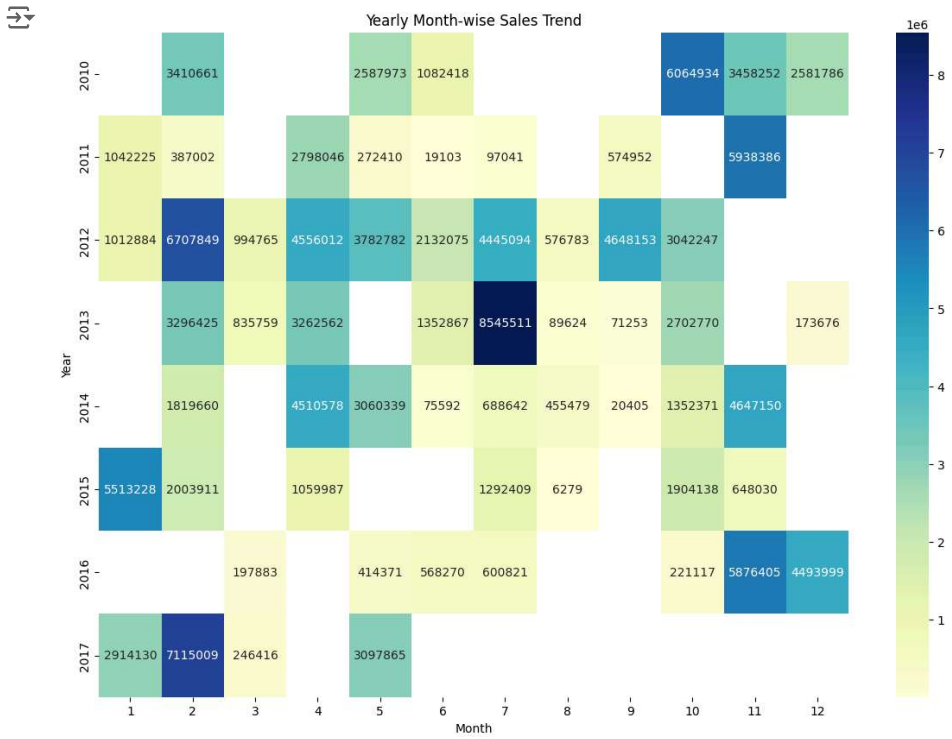
Next steps: Generate code with ProductProfit View recommended plots

```
plt.figure(figsize=(10,6))
sns.barplot(data = ProductProfit, x = 'Total Profit', y = 'Item Type', hue = 'Total Profit')
plt.title("Profit by Item Type")
plt.show()
```



Yearly Month-wise Sales

```
# Yearly Month-wise Sales Trend
plt.figure(figsize=(14, 10))
sns.heatmap(yearly_monthly_sales_trend, annot=True, fmt=".0f", cmap="YlGnBu")
plt.title('Yearly Month-wise Sales Trend')
plt.xlabel('Month')
plt.ylabel('Year')
plt.show()
```




```
# Calculate key metrics

# Total sales
total_sales = amazon_sales_data['Total Revenue'].sum()

# Average sales per order
average_sales_per_order = amazon_sales_data['Total Revenue'].mean()


# Select numeric columns for correlation matrix
numeric_columns = ['Units Sold', 'Unit Price', 'Unit Cost', 'Total Revenue', 'Total Cost', 'Total Profit']

# Correlation matrix to find relationships between numeric attributes
correlation_matrix = amazon_sales_data[numeric_columns].corr()


# Visualizing the correlation matrix
plt.figure(figsize=(14, 8))
sns.heatmap(correlation_matrix, annot=True, fmt=".2f", cmap="coolwarm")
plt.title('Correlation Matrix of Sales Data Attributes')
plt.show()
```