Name	Hatim Sawai
UID No.	2021300108
Course	Advanced Data Visualization

Experiment 1

Aim	Create basic charts using Tableau / Power BI / R / Python / D3.js to be performed on the dataset of Ecommerce field Complete all plots on practice dataset and reproduce on e-commerce dataset. Basic - Bar chart, Pie chart, Histogram, Timeline chart, Scatter plot, Bubble plot Calculate Product wise sales, region wise sales Write observations from each chart

1. Importing Libraries and Dataset

In [ ]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

Dataset You can view the dataset here

Description
This dataset contains the details of Amazon Sales. It provides detailed insights into Amazon sales data, including SKU Code, Design Number, Stock, Category, Size and Color, to help optimize product profitability. It has 23 columns:

Category: Type of product. (String)
Size: Size of the product. (String)
Date: Date of the sale. (Date)
Status: Status of the sale. (String)
Fulfilment: Method of fulfilment. (String)
Style: Style of the product. (String)
SKU: Stock Keeping Unit. (String)
ASIN: Amazon Standard Identification Number. (String)
Courier Status: Status of the courier. (String)
Qty: Quantity of the product. (Integer)
Amount: Amount of the sale. (Float)
B2B: Business to business sale. (Boolean)
Currency: The currency used for the sale. (String)

In [ ]:
```
data = pd.read_csv("../Datasets/Amazon Sale Report.csv", low_memory=False, index_co
print(data.shape)
data.head()
```

```
(128975, 23)
```

Out[ ]:

index	Order ID	Date	Status	Fulfilment	Sales Channel	ship-service-level	Style	SKU	Cat
0	405-8078784-5731545	04-30-22	Cancelled	Merchant	Amazon.in	Standard	SET389	SET389-KR-NP-S	
1	171-9198151-1101146	04-30-22	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	JNE3781	JNE3781-KR-XXXL	
2	404-0687676-7273146	04-30-22	Shipped	Amazon	Amazon.in	Expedited	JNE3371	JNE3371-KR-XL	
3	403-9615377-8133951	04-30-22	Cancelled	Merchant	Amazon.in	Standard	J0341	J0341-DR-L	W
4	407-1069790-7240320	04-30-22	Shipped	Amazon	Amazon.in	Expedited	JNE3671	JNE3671-TU-XXXL	

5 rows × 23 columns

2. Data Preprocessing

In [ ]:
```
# delete last column
data = data.iloc[:, :-1]
```

In [ ]:
```
# check for missing values
data.isnull().sum()
```

Out[ ]:
```
Order ID                0
Date                    0
Status                  0
Fulfilment              0
Sales Channel           0
ship-service-level      0
Style                   0
SKU                     0
Category                0
Size                    0
ASIN                    0
Courier Status          0
Qty                     0
currency             7793
Amount               7793
ship-city              28
ship-state             28
ship-postal-code       28
ship-country           28
B2B                     0
dtype: int64
```

In [ ]:
```
# delete columns: 'fulfilled-by' and 'promotion-ids'
data = data.drop(['fulfilled-by', 'promotion-ids'], axis=1)
```

In [ ]:
```
# delete rows with more than 20% missing values
data = data.dropna(thresh=data.shape[1]*0.80)

# replace nan in courier status with same value as in status
data['Courier Status'] = data['Courier Status'].fillna(data['Status'])
```

In [ ]:
```
print(data.shape)
data.head()
```

```
(128970, 20)
```

Out[ ]:

index	Order ID	Date	Status	Fulfilment	Sales Channel	ship-service-level	Style	SKU	Cat
0	405-8078784-5731545	04-30-22	Cancelled	Merchant	Amazon.in	Standard	SET389	SET389-KR-NP-S	
1	171-9198151-1101146	04-30-22	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	JNE3781	JNE3781-KR-XXXL	
2	404-0687676-7273146	04-30-22	Shipped	Amazon	Amazon.in	Expedited	JNE3371	JNE3371-KR-XL	
3	403-9615377-8133951	04-30-22	Cancelled	Merchant	Amazon.in	Standard	J0341	J0341-DR-L	W
4	407-1069790-7240320	04-30-22	Shipped	Amazon	Amazon.in	Expedited	JNE3671	JNE3671-TU-XXXL	

In [ ]:
```
# save cleaned data
data.to_csv("../Datasets/Amazon_Sales_cleaned.csv")
```

3. Charts & Plots

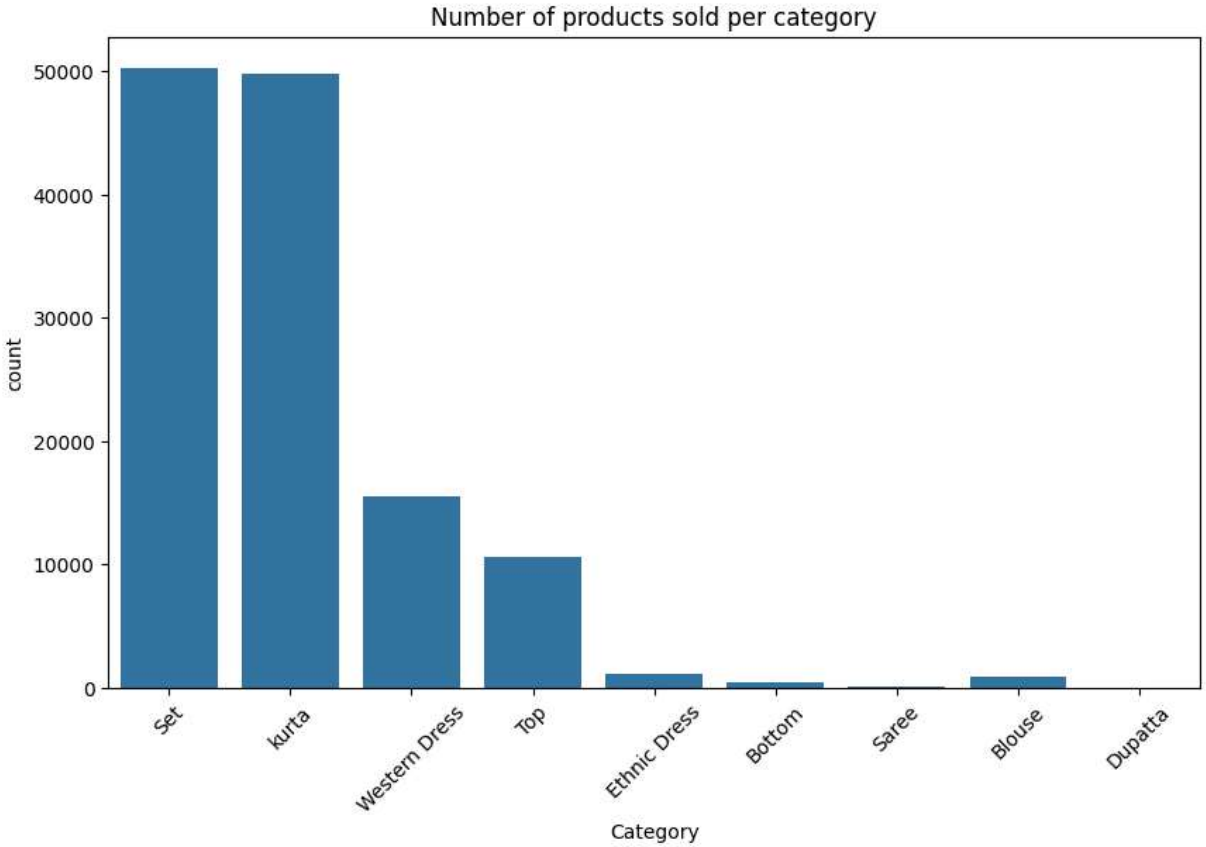
In [ ]:
```
df = pd.read_csv("../Datasets/Amazon_Sales_cleaned.csv", index_col=0)
df.head()
```

Out[ ]:

index	Order ID	Date	Status	Fulfilment	Sales Channel	ship-service-level	Style	SKU	Cat
0	405-8078784-5731545	04-30-22	Cancelled	Merchant	Amazon.in	Standard	SET389	SET389-KR-NP-S	
1	171-9198151-1101146	04-30-22	Shipped - Delivered to Buyer	Merchant	Amazon.in	Standard	JNE3781	JNE3781-KR-XXXL	
2	404-0687676-7273146	04-30-22	Shipped	Amazon	Amazon.in	Expedited	JNE3371	JNE3371-KR-XL	
3	403-9615377-8133951	04-30-22	Cancelled	Merchant	Amazon.in	Standard	J0341	J0341-DR-L	W
4	407-1069790-7240320	04-30-22	Shipped	Amazon	Amazon.in	Expedited	JNE3671	JNE3671-TU-XXXL	

3.1 Bar Chart

In [ ]:
```
# make a bar plot
plt.figure(figsize=(10, 6))
sns.countplot(data=df, x='Category')
plt.xticks(rotation=45)
plt.title('Number of products sold per category')
plt.show()
```



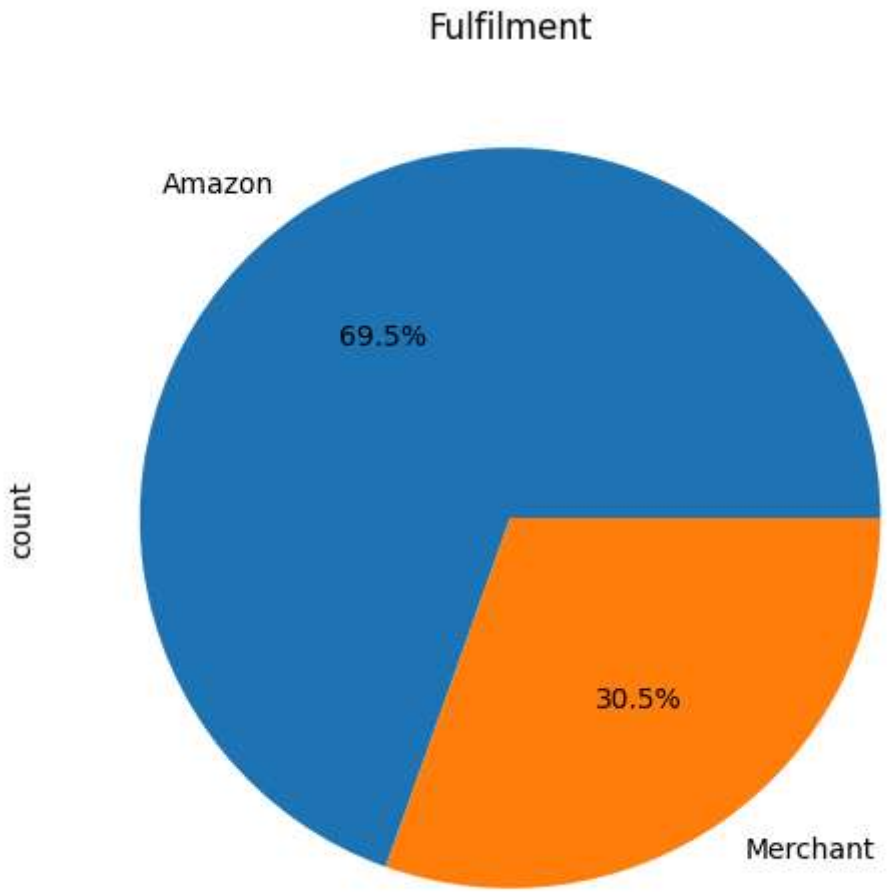
Oberservation:

- The bar chart shows the total sales of each product category.
- The category 'Set' has the highest sales, followed by 'Kurta' and 'Western Dress'.
- The category 'Dupatta' has the lowest sales.

This suggests that the 'Set' category is the most popular among customers and should be focused on more.

3.2 Pie Chart

In [ ]:
```
# make a pie chart for ship-service-level
plt.figure(figsize=(10, 6))
df['Fulfilment'].value_counts().plot.pie(autopct='%1.1f%%')
plt.title('Fulfilment')
plt.show()
```



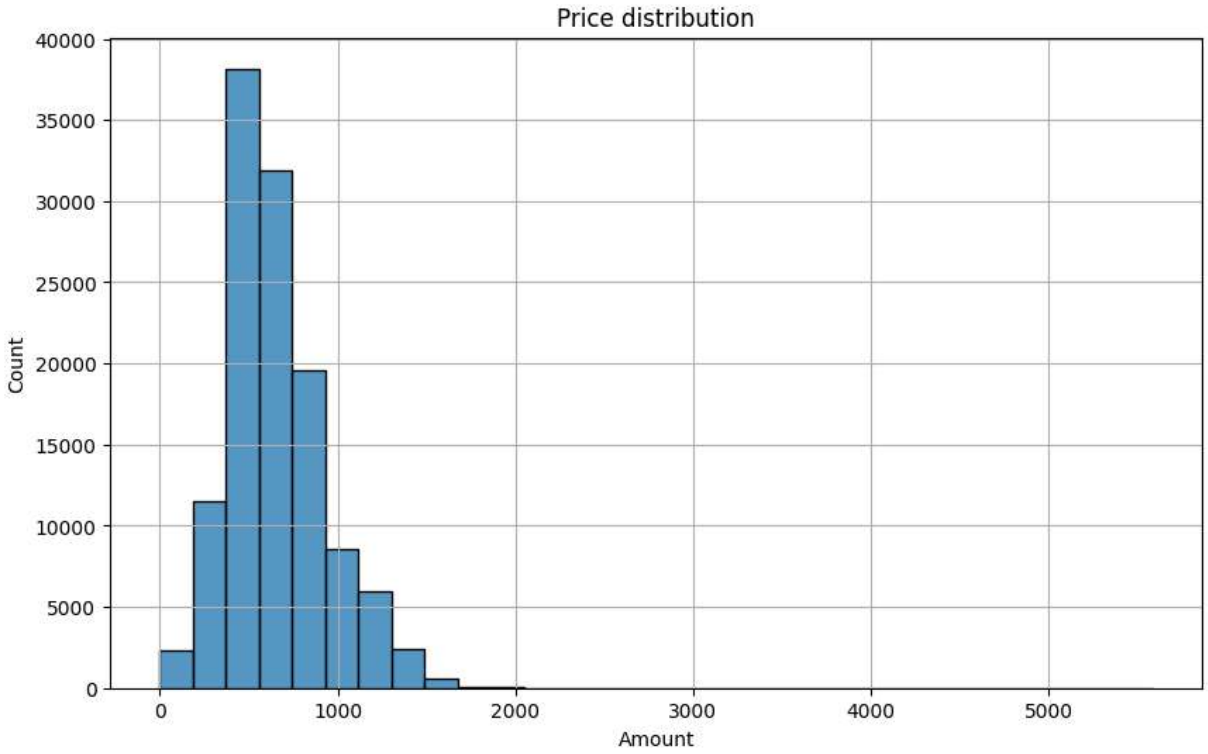
Oberservation:

- The pie chart shows how much percentage of orders are fulfilled by each method.
- Most Orders are Fullfilled by Amazon itself followed by others being fulfilled by the seller (Merchant).

This suggests that Amazon should focus on improving its own fulfilment services.

3.3 Histogram

In [ ]:
```
# make a histogram for price
plt.figure(figsize=(10, 6))
sns.histplot(df['Amount'], bins=30)
plt.title('Price distribution')
plt.grid()
plt.show()
```



Oberservation:

- The histogram shows the distribution of the price per product.
- Most products are priced between 0 and $1000.

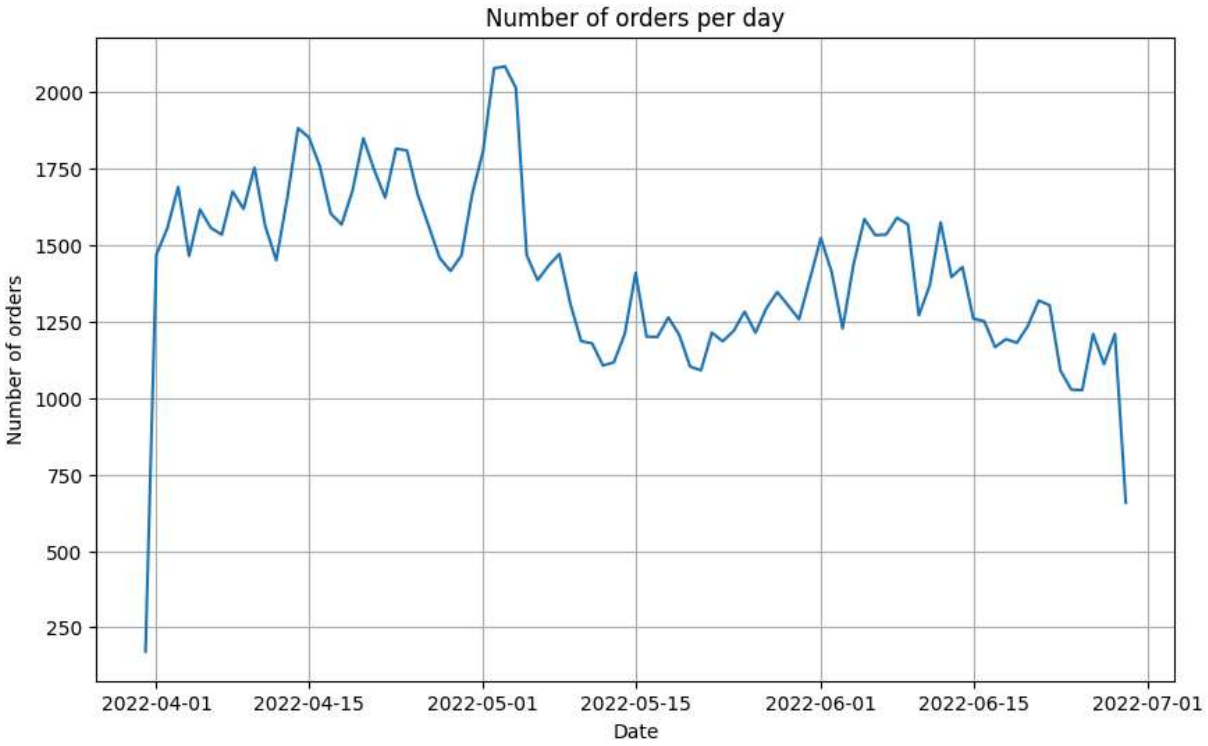
This suggests that most products are priced reasonably and are affordable to customers.

3.4 Timeline Chart

In [ ]:
```
# make a time series plot for order date
plt.figure(figsize=(10, 6))
df['Date'] = pd.to_datetime(df['Date'])

# group by date
df_grouped = df.groupby('Date').size().reset_index(name='Count')

plt.plot(df_grouped['Date'], df_grouped['Count'])
plt.title('Number of orders per day')
plt.xlabel('Date')
plt.ylabel('Number of orders')
plt.grid(True)
plt.show()
```



Oberservation:

- The timeline chart shows the sales trend over time.
- The sales peak in the month of May, out of the following months: April, May, June, July

This suggests that the month of May is the most profitable month for Amazon in the first quarter of the year owing to summer vacations.

3.5 Scatter Plot

In [ ]:
```
# make a scatter plot for price vs quantity
plt.figure(figsize=(10, 6))
sns.scatterplot(data=df, x='Qty', y='Amount')
plt.title('Price vs Quantity')
plt.grid()
plt.show()
```

Oberservation:

- The scatter plot shows the relationship between the quantity of products sold and the amount of the sale.
- There is a positive correlation between the quantity of products sold and the amount of the sale.

3.6 Bubble Plot

In [ ]:
```
# make a bubble plot for price vs quantity

plt.figure(figsize=(10, 6))
sns.scatterplot(data=df, x='Qty', y='Amount', size='Amount', sizes=(20, 200))
plt.title('Price vs Quantity')
plt.grid()
plt.show()
```

Oberservation:

- The bubble plot shows the relationship between the quantity of products sold and the amount of the sale, with the size of the bubble representing the price of the product.
- The bubble plot confirms the positive correlation between the quantity of products sold and the amount of the sale.

Conclusion

In this experiment, we learned how to create basic charts using Pandas and Seaborn in Python on a dataset of the Ecommerce field. We created the following plots on the practice dataset and reproduced them on the e-commerce dataset: bar chart, pie chart, histogram, timeline chart, scatter plot, and bubble plot. We wrote observations from each chart to gain insights into the data.