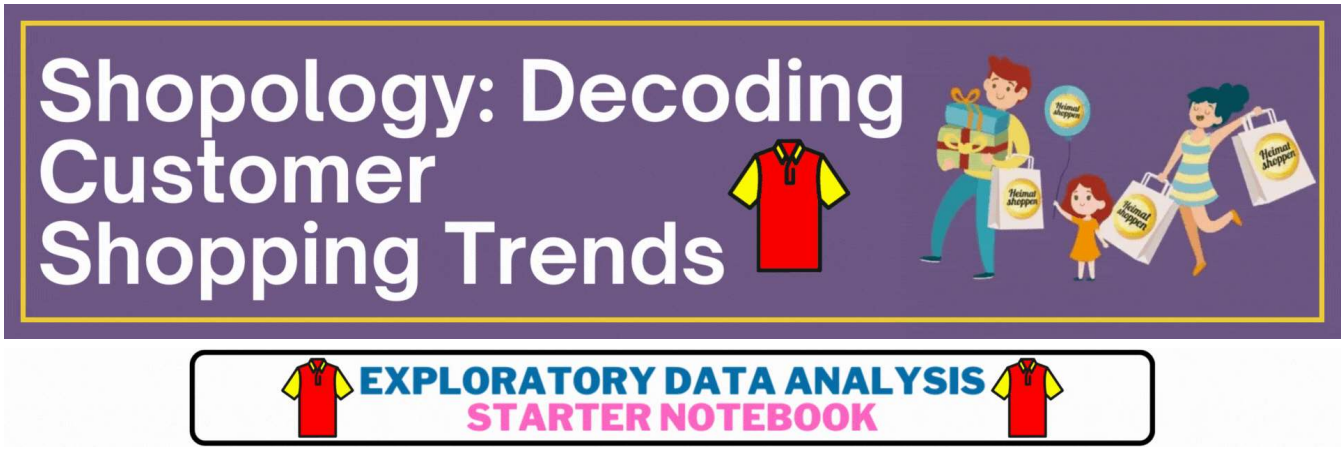


- 📌 **Customer ID** - Unique identifier for each customer.
- 📌 **Age** - Age of the customer.
- 📌 **Gender** - Gender of the customer (Male/Female).
- 📌 **Item Purchased** - The item purchased by the customer.
- 📌 **Category** - Category of the item purchased.
- 📌 **Purchase Amount (USD)** - The amount of the purchase in USD.
- 📌 **Location** - Location where the purchase was made.
- 📌 **Size** - Size of the purchased item.
- 📌 **Color** - Color of the purchased item.
- 📌 **Season** - Season during which the purchase was made.
- 📌 **Review Rating** - Rating given by the customer for the purchased item.
- 📌 **Subscription Status** - Indicates if the customer has a subscription (Yes/No).
- 📌 **Shipping Type** - Type of shipping chosen by the customer.
- 📌 **Discount Applied** - Indicates if a discount was applied to the purchase (Yes/No).
- 📌 **Promo Code Used** - Indicates if a promo code was used for the purchase (Yes/No).
- 📌 **Previous Purchases** - Number of previous purchases made by the customer.
- 📌 **Payment Method** - Customer's most preferred payment method.
- 📌 **Frequency of Purchases** - Frequency at which the customer makes purchases (e.g., Weekly, Fortnightly, Monthly).

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
```

```
!pip install WordCloud
```

```
Requirement already satisfied: WordCloud in /usr/local/lib/python3.11/dist-packages (1.9.4)
Requirement already satisfied: numpy>=1.6.1 in /usr/local/lib/python3.11/dist-packages (from WordCloud) (1.26.4)
Requirement already satisfied: pillow in /usr/local/lib/python3.11/dist-packages (from WordCloud) (11.1.0)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.11/dist-packages (from WordCloud) (3.10.0)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib->WordCloud) (1.3.1)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.11/dist-packages (from matplotlib->WordCloud) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.11/dist-packages (from matplotlib->WordCloud) (4.55.8)
Requirement already satisfied: kiwisolver>=1.3.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib->WordCloud) (1.4.8)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.11/dist-packages (from matplotlib->WordCloud) (24.2)
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib->WordCloud) (3.2.1)
Requirement already satisfied: python-dateutil>=2.7 in /usr/local/lib/python3.11/dist-packages (from matplotlib->WordCloud) (2.8.2)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-packages (from python-dateutil>=2.7->matplotlib->WordClouc
```

```
# importing libraries
import numpy as np # Importing the numpy library for array operations and mathematical functions
import pandas as pd # Use for exploring the data
import seaborn as sns # it has also plot
import matplotlib.pyplot as plt # for some extra plot functions
import plotly.express as px # this library can makes interactive plots
```

```
# reading the data set
shop = pd.read_csv('/content/drive/MyDrive/Colab Notebooks/shopping_trends_updated.csv')
```

```
shop.shape
```

```
(3900, 18)
```

```
shop.to_excel('/content/drive/MyDrive/Colab Notebooks/shopping_trends_updated.xlsx')
```

```
shop.head()
```

	Customer ID	Age	Gender	Item Purchased	Category	Purchase Amount (USD)	Location	Size	Color	Season	Review Rating	Subscription Status	Shipping Type	Discou Appli
0	1	55	Male	Blouse	Clothing	53	Kentucky	L	Gray	Winter	3.1	Yes	Express	
1	2	19	Male	Sweater	Clothing	64	Maine	L	Maroon	Winter	3.1	Yes	Express	
2	3	50	Male	Jeans	Clothing	73	Massachusetts	S	Maroon	Spring	3.1	Yes	Free Shipping	
													Next Day	

```
shop.dtypes
```

	0
Customer ID	int64
Age	int64
Gender	object
Item Purchased	object
Category	object
Purchase Amount (USD)	int64
Location	object
Size	object
Color	object
Season	object
Review Rating	float64
Subscription Status	object
Shipping Type	object
Discount Applied	object
Promo Code Used	object
Previous Purchases	int64
Payment Method	object
Frequency of Purchases	object

dtype: object

```
# it shows the names of the columns
shop.columns
```

```
Index(['Customer ID', 'Age', 'Gender', 'Item Purchased', 'Category',
       'Purchase Amount (USD)', 'Location', 'Size', 'Color', 'Season',
       'Review Rating', 'Subscription Status', 'Shipping Type',
       'Discount Applied', 'Promo Code Used', 'Previous Purchases',
       'Payment Method', 'Frequency of Purchases'],
      dtype='object')
```

```
shop.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3900 entries, 0 to 3899
Data columns (total 18 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   Customer ID             3900 non-null   int64
 1   Age                     3900 non-null   int64
 2   Gender                  3900 non-null   object
 3   Item Purchased          3900 non-null   object
 4   Category                3900 non-null   object
 5   Purchase Amount (USD)   3900 non-null   int64
 6   Location                3900 non-null   object
 7   Size                    3900 non-null   object
```

```
 8   Color                   3900 non-null   object
 9   Season                  3900 non-null   object
 10  Review Rating           3900 non-null   float64
 11  Subscription Status     3900 non-null   object
 12  Shipping Type           3900 non-null   object
 13  Discount Applied        3900 non-null   object
 14  Promo Code Used         3900 non-null   object
 15  Previous Purchases      3900 non-null   int64
 16  Payment Method          3900 non-null   object
 17  Frequency of Purchases  3900 non-null   object
dtypes: float64(1), int64(4), object(13)
memory usage: 548.6+ KB
```

```
shop.shape
```

```
(3900, 18)
```

```
shop.isnull().sum()
```

	0
Customer ID	0
Age	0
Gender	0
Item Purchased	0
Category	0
Purchase Amount (USD)	0
Location	0
Size	0
Color	0
Season	0
Review Rating	0
Subscription Status	0
Shipping Type	0
Discount Applied	0
Promo Code Used	0
Previous Purchases	0
Payment Method	0
Frequency of Purchases	0

dtype: int64

```
print(f"The unique values of the 'Gender' column are: {shop['Gender'].unique()}")
print()# This will print a blank line
print(f"The unique values of the 'Category' column are: {shop['Category'].unique()}")
print()# This will print a blank line
print(f"The unique values of the 'Size' column are: {shop['Size'].unique()}")
print()# This will print a blank line
print(f"The unique values of the 'Subscription Status' column are: {shop['Subscription Status'].unique()}")
print()# This will print a blank line
print(f"The unique values of the 'Shipping Type' column are: {shop['Shipping Type'].unique()}")
print()# This will print a blank line
print(f"The unique values of the 'Discount Applied' column are: {shop['Discount Applied'].unique()}")
print()# This will print a blank line
print(f"The unique values of the 'Promo Code Used' column are: {shop['Promo Code Used'].unique()}")
print()# This will print a blank line
print(f"The unique values of the 'Payment Method' column are: {shop['Payment Method'].unique()}")
```

```
The unique values of the 'Gender' column are: ['Male' 'Female']

The unique values of the 'Category' column are: ['Clothing' 'Footwear' 'Outerwear' 'Accessories']

The unique values of the 'Size' column are: ['L' 'S' 'M' 'XL']

The unique values of the 'Subscription Status' column are: ['Yes' 'No']

The unique values of the 'Shipping Type' column are: ['Express' 'Free Shipping' 'Next Day Air' 'Standard' '2-Day Shipping'
 'Store Pickup']

The unique values of the 'Discount Applied' column are: ['Yes' 'No']
```

```
The unique values of the 'Promo Code Used' column are: ['Yes' 'No']

The unique values of the 'Payment Method' column are: ['Venmo' 'Cash' 'Credit Card' 'PayPal' 'Bank Transfer' 'Debit Card']
```

OBSERVATION:

Upon initial examination of the dataset, it is evident that we have a comprehensive and well-structured dataset with 3900 rows and 18 columns. The data is complete, with no missing values, which allows us to proceed confidently with our analysis.

Let's delve into the columns and their significance in understanding our custome

- **Customer ID:** This column serves as a unique identifier for each customer, enabling us to differentiate between individuals.
- **Age:** The age column provides insights into the age demographics of our customers, helping us understand their preferences and behaviors.
- **Gender:** This column showcases the gender of the customers, enabling us to analyze buying patterns based on gender.
- **Item Purchased:** Here, we can identify the specific products that customers have bought, allowing us to gain an understanding of popular choices.
- **Category:** The category column categorizes the products into different groups such as clothing, footwear, and more, aiding us in analyzing trends within specific product categories.
- **Purchase Amount (USD):** This column reveals the amount customers spent on their purchases, providing insights into their spending habits.
- **Location:** The location column indicates the geographical location of customers, which can help identify regional trends and preferences.
- **Size:** This column denotes the size of the purchased products, assisting in understanding size preferences across different categories.
- **Color:** Here, we can determine the color preferences of customers, aiding in analyzing color trends and their impact on purchasing decisions.
- **Season:** The season column allows us to identify the season during which customers made their purchases, enabling us to explore seasonal shopping trends.
- **Review Rating:** This column showcases the ratings given by customers, providing valuable feedback on product satisfaction and quality.
- **Subscription Status:** This column indicates whether customers have opted for a subscription status, which can help us understand customer loyalty and engagement.
- **Shipping Type:** Here, we can identify the different shipping methods used to deliver products to customers, shedding light on preferred shipping options.
- **Discount Applied:** This column indicates whether a discount was applied to the purchased products, enabling us to analyze the impact of discounts on customer behavior.
- **Promo Code Used:** Here, we can identify whether customers utilized promo codes during their purchases, helping us evaluate the effectiveness of promotional campaigns.
- **Previous Purchases:** This column reveals the number of previous purchases made by customers, aiding in understanding customer loyalty and repeat business.
- **Payment Method:** The payment method column showcases the various methods used by customers to make their purchases, allowing us to analyze preferred payment options.
- **Frequency of Purchases:** This column provides insights into the frequency at which customers make purchases, helping us identify patterns and customer buying habits.

ustomer buying habits. With this rich and diverse dataset, we are well-equipped to explore customer shopping trends, understand their preferences, and uncover valuable insights that can drive informed decision-making and enhance the overall customer experience. Let's embark on this exciting analysis journey!

1 What is the overall distribution of customer ages in the dataset?

```
shop['Age'].value_counts()
```

Age	count
69	88
57	87
41	86
25	85
49	84
50	83
54	83
27	83
62	83
32	82
19	81
58	81
42	80
43	79
28	79
31	79
37	77
46	76
29	76
68	75
59	75
63	75
56	74
36	74
55	73
52	73
64	73
35	72
51	72
65	72
40	72
45	72
47	71
66	71
30	71
23	71
38	70
53	70
18	69
21	69
26	69
34	68
48	68
24	68
39	68
70	67
22	66
61	65

60	65
33	63
20	62
67	54
44	51

dtype: int64

```
shop['Age'].mean()
```

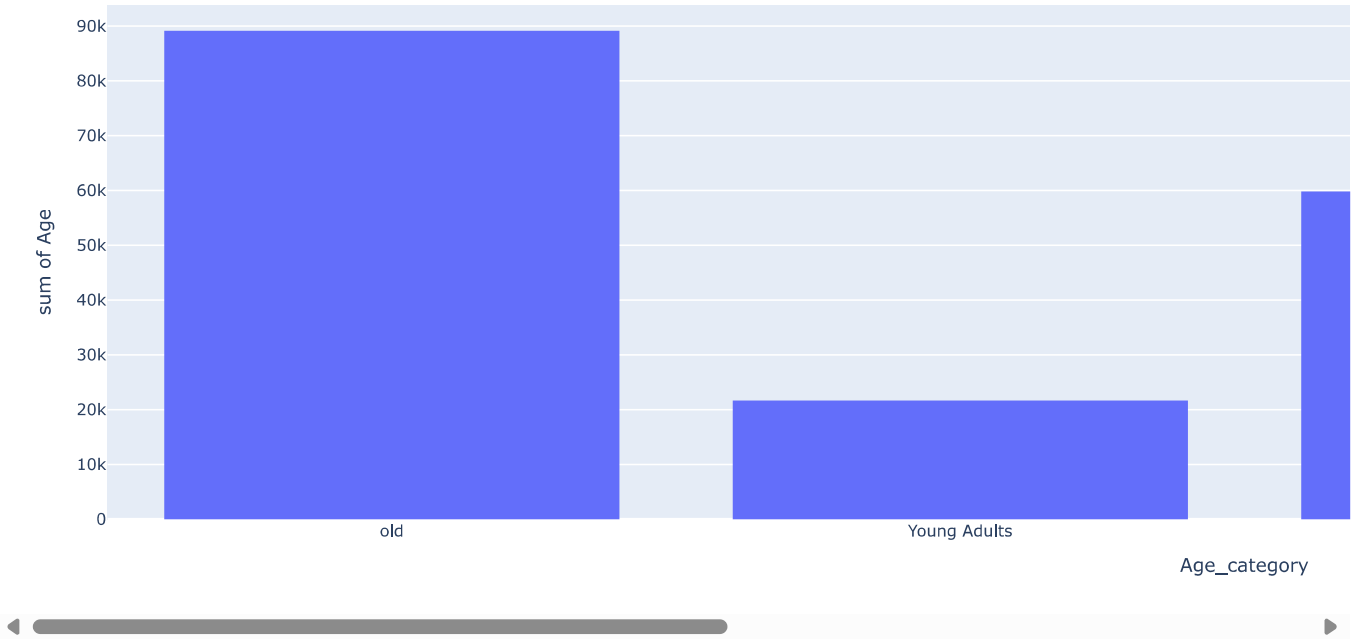
```
44.06846153846154
```

```
shop['Gender'].unique()
```

```
array(['Male', 'Female'], dtype=object)
```

```
shop['Age_category'] = pd.cut(shop['Age'], bins= [0,15, 18 , 30 , 50 , 70] , labels= ['child' , 'teen' , 'Young Adults' ,'Middle-Aged A
                                                                                                                                , 'old'] )
```

```
fig = px.histogram(shop , y = 'Age' , x = 'Age_category')
fig.show()
```



2 How does the average purchase amount vary across different product categories?

```
shop.columns
```

```
Index(['Customer ID', 'Age', 'Gender', 'Item Purchased', 'Category',
       'Purchase Amount (USD)', 'Location', 'Size', 'Color', 'Season',
       'Review Rating', 'Subscription Status', 'Shipping Type',
       'Discount Applied', 'Promo Code Used', 'Previous Purchases',
       'Payment Method', 'Frequency of Purchases', 'Age_category'],
      dtype='object')
```

```
shop['Category'].unique()
```

```
array(['Clothing', 'Footwear', 'Outerwear', 'Accessories'], dtype=object)
```

```
shop.groupby('Category')['Purchase Amount (USD)'].mean()
```

Category	Purchase Amount (USD)
Accessories	59.838710
Clothing	60.025331
Footwear	60.255426
Outerwear	57.172840

dtype: float64

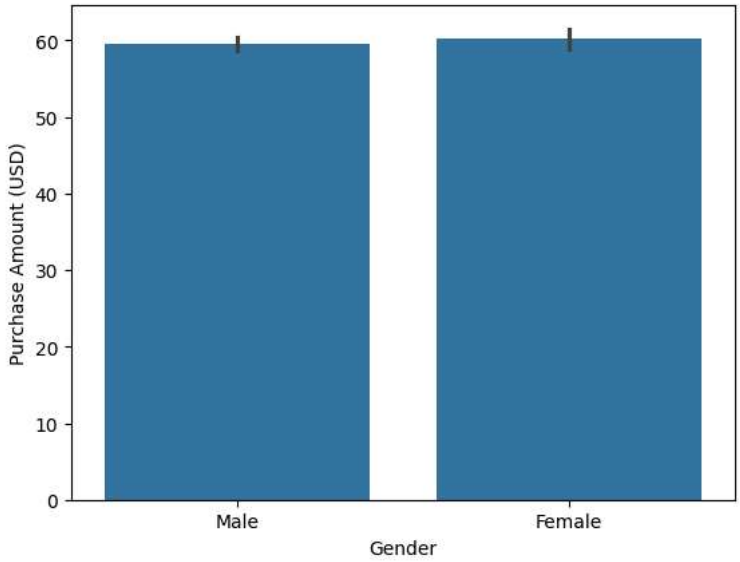
3 Which gender has the highest number of purchases?

```
shop.columns
```

```
Index(['Customer ID', 'Age', 'Gender', 'Item Purchased', 'Category',
       'Purchase Amount (USD)', 'Location', 'Size', 'Color', 'Season',
       'Review Rating', 'Subscription Status', 'Shipping Type',
       'Discount Applied', 'Promo Code Used', 'Previous Purchases',
       'Payment Method', 'Frequency of Purchases', 'Age_category'],
      dtype='object')
```

```
sns.barplot(shop , x = 'Gender' , y = 'Purchase Amount (USD)')
```

```
<Axes: xlabel='Gender', ylabel='Purchase Amount (USD)'>
```



4 What are the most commonly purchased items in each category?

```
shop.columns
```

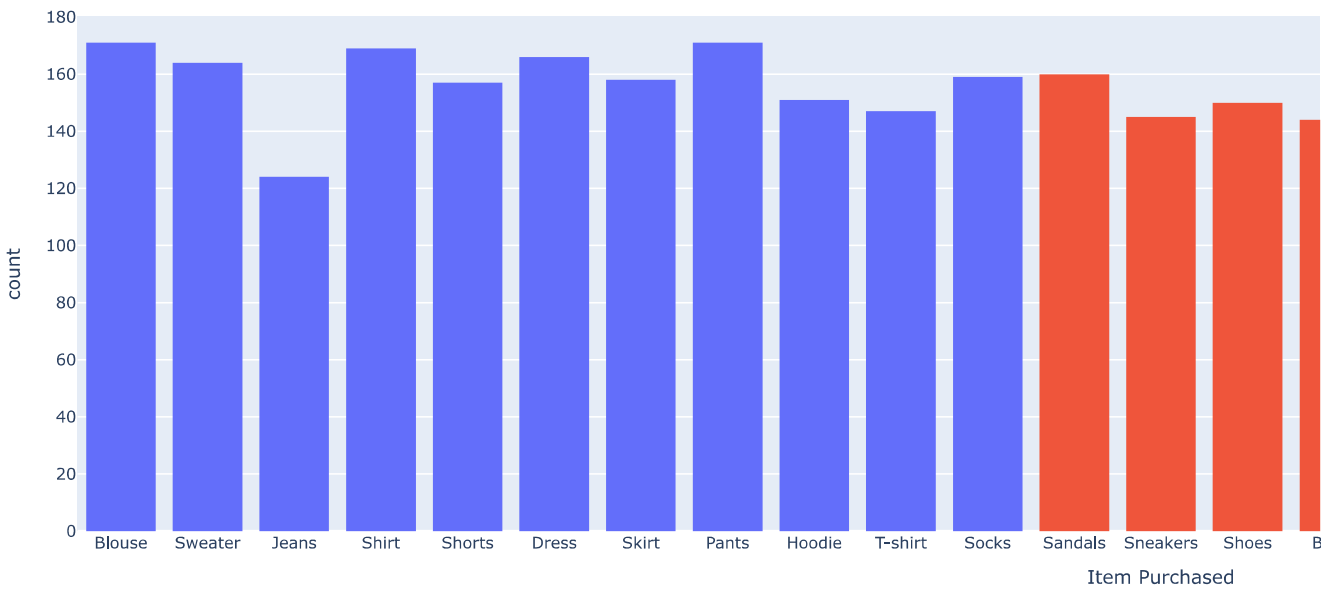
```
Index(['Customer ID', 'Age', 'Gender', 'Item Purchased', 'Category',
       'Purchase Amount (USD)', 'Location', 'Size', 'Color', 'Season',
       'Review Rating', 'Subscription Status', 'Shipping Type',
       'Discount Applied', 'Promo Code Used', 'Previous Purchases',
       'Payment Method', 'Frequency of Purchases', 'Age_category'],
      dtype='object')
```

```
shop.groupby('Category')['Item Purchased'].value_counts()
```

Category	Item Purchased	count
Accessories	Jewelry	171
	Belt	161
	Sunglasses	161
	Scarf	157
	Hat	154
	Handbag	153
	Backpack	143
	Gloves	140
Clothing	Blouse	171
	Pants	171
	Shirt	169
	Dress	166
	Sweater	164
	Socks	159
	Skirt	158
	Shorts	157
	Hoodie	151
	T-shirt	147
	Jeans	124
Footwear	Sandals	160
	Shoes	150
	Sneakers	145
	Boots	144
Outerwear	Jacket	163
	Coat	161

dtype: int64

```
fig = px.histogram(shop , x = 'Item Purchased' , color = 'Category')
fig.show()
```



5 Are there any specific seasons or months where customer spending is significantly higher?

```
shop['Season'].unique()
```

```
array(['Winter', 'Spring', 'Summer', 'Fall'], dtype=object)
```

```
shop[shop['Season'] == 'Summer'].value_counts().sum()
```

```
955
```

```
shop[shop['Season'] == 'Winter'].value_counts().sum()
```

```
971
```

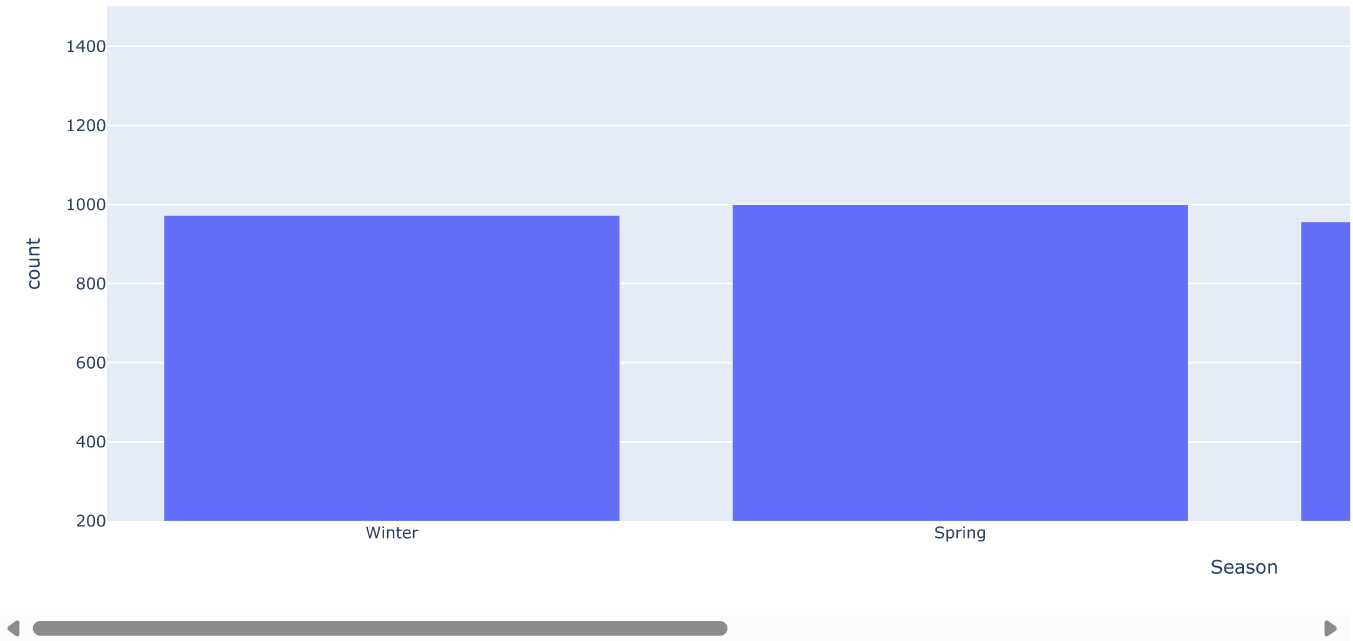
```
shop[shop['Season'] == 'Spring'].value_counts().sum()
```

```
999
```

```
shop[shop['Season'] == 'Fall'].value_counts().sum()
```

```
975
```

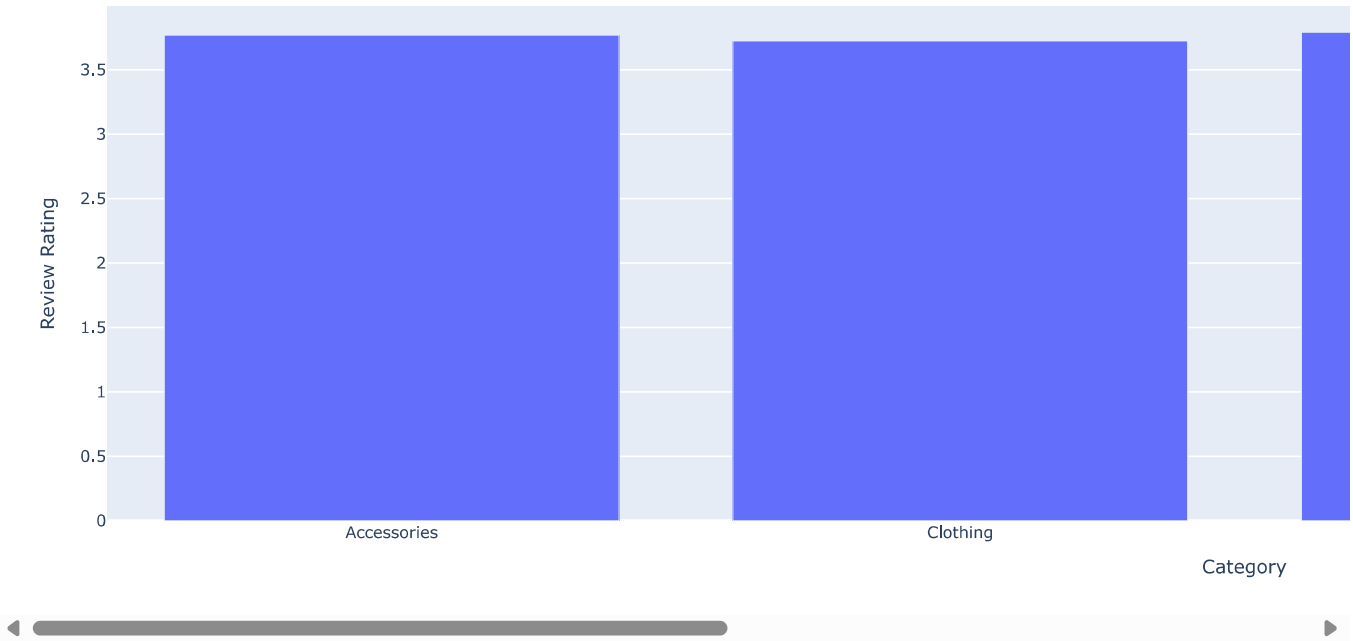
```
fig = px.histogram(shop , x = 'Season' , range_y= [200 , 1500] )

fig.show()
```



6 What is the average rating given by customers for each product category?

```
shop_groupby = shop.groupby('Category')['Review Rating'].mean().reset_index()
```

```
fig = px.bar(shop_groupby ,x= 'Category' , y = 'Review Rating' )
fig.show()
```



7 Are there any notable differences in purchase behavior between subscribed and non-subscribed customers?

```
shop.columns
```

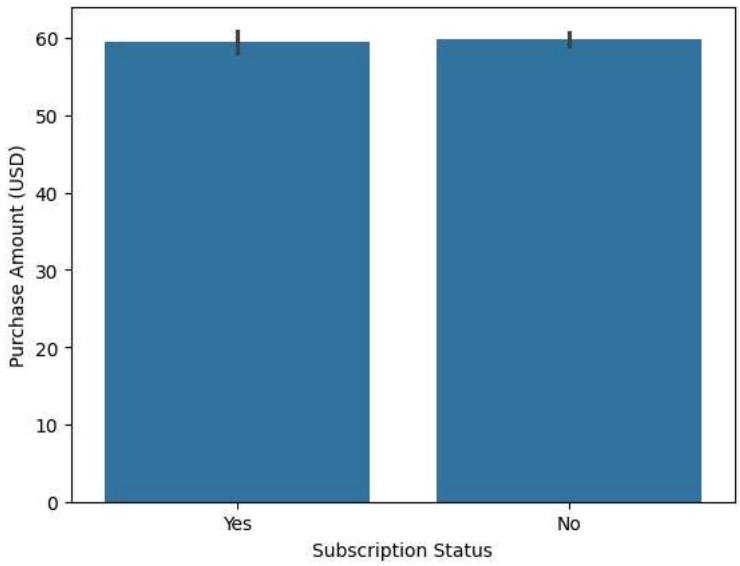
```
Index(['Customer ID', 'Age', 'Gender', 'Item Purchased', 'Category',
       'Purchase Amount (USD)', 'Location', 'Size', 'Color', 'Season',
       'Review Rating', 'Subscription Status', 'Shipping Type',
       'Discount Applied', 'Promo Code Used', 'Previous Purchases',
       'Payment Method', 'Frequency of Purchases', 'Age_category'],
      dtype='object')
```

```
shop['Subscription Status'].unique()
```

```
array(['Yes', 'No'], dtype=object)
```

```
sns.barplot(shop  , x = 'Subscription Status' , y = 'Purchase Amount (USD)')
```

```
<Axes: xlabel='Subscription Status', ylabel='Purchase Amount (USD)'>
```




```
shop['Purchase Amount (USD)'].sum()
```

```
233081
```

```
shop.groupby('Subscription Status')['Purchase Amount (USD)'].mean()
```

Subscription Status	Purchase Amount (USD)
No	59.865121
Yes	59.491928

dtype: float64

8 Which payment method is the most popular among customers?

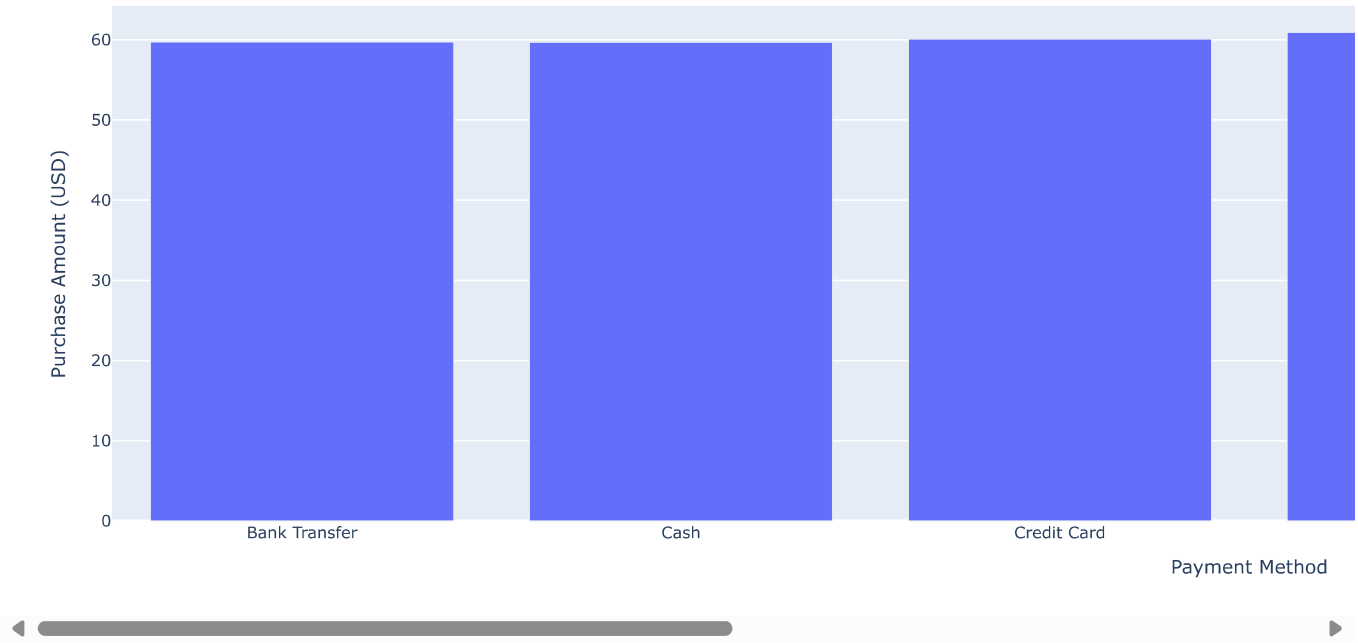
```
shop.groupby('Payment Method')['Purchase Amount (USD)'].mean().sort_values(ascending= False)
```

Payment Method	Purchase Amount (USD)
Debit Card	60.915094
Credit Card	60.074516
Bank Transfer	59.712418
Cash	59.704478
PayPal	59.245199
Venmo	58.949527

dtype: float64

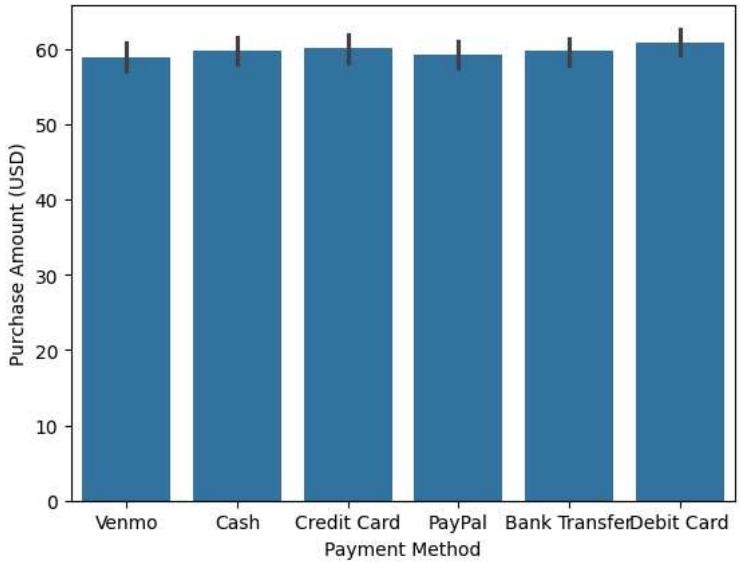
```
shop_groupby = shop.groupby('Payment Method')['Purchase Amount (USD)'].mean().reset_index()
```

```
fig = px.bar(shop_groupby , x = 'Payment Method' , y = 'Purchase Amount (USD)')
fig.show()
```




```
sns.barplot(shop ,x='Payment Method' , y = 'Purchase Amount (USD)')
```

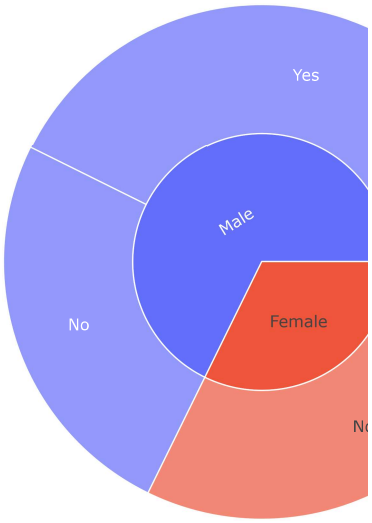
```
<Axes: xlabel='Payment Method', ylabel='Purchase Amount (USD)'>
```



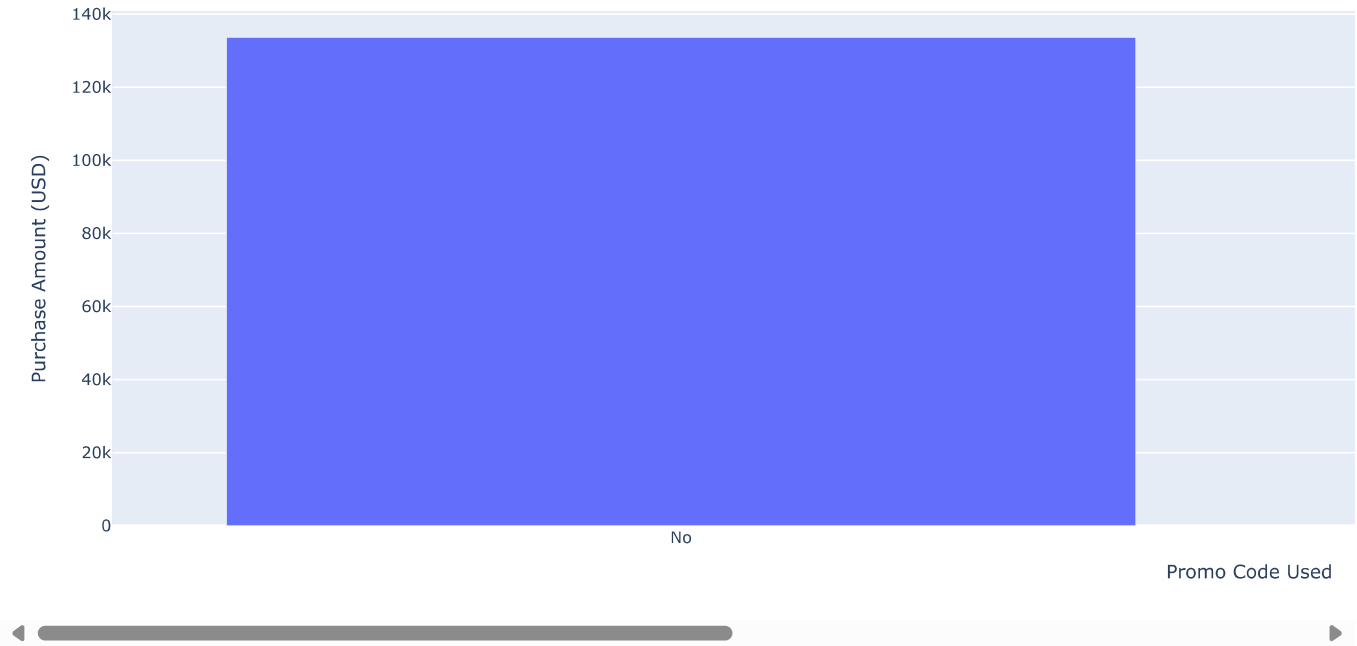
9 Do customers who use promo codes tend to spend more than those who don't?

```
shop_groupby  = shop.groupby('Promo Code Used')['Purchase Amount (USD)'].sum().reset_index()
```

```
fig = px.sunburst(shop , path=['Gender' , 'Promo Code Used'] , values='Purchase Amount (USD)')
fig.show()
```




```
fig  =  px.bar(shop_groupby , x= 'Promo Code Used' , y = 'Purchase Amount (USD)')
fig.show()
```



10 How does the frequency of purchases vary across different age groups?

```
shop[['Age' , 'Age_category']]
```

	Age	Age_category
0	55	old
1	19	Young Adults
2	50	Middle-Aged Adults
3	21	Young Adults
4	45	Middle-Aged Adults
...	...	...
3895	40	Middle-Aged Adults
3896	52	old
3897	46	Middle-Aged Adults
3898	44	Middle-Aged Adults
3899	52	old

3900 rows × 2 columns

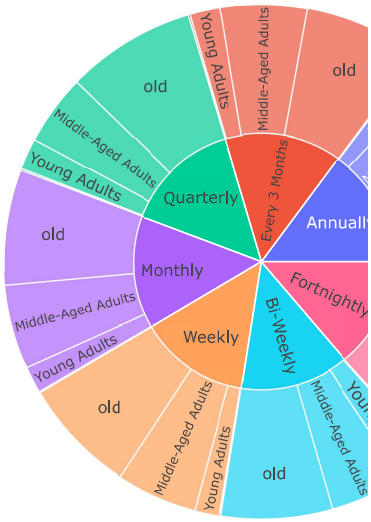
```
shop['Age_category'].unique()
```

```
['old', 'Young Adults', 'Middle-Aged Adults', 'teen']
Categories (5, object): ['child' < 'teen' < 'Young Adults' < 'Middle-Aged Adults' < 'old']
```

```
shop_group = shop.groupby('Frequency of Purchases')['Age'].sum()
```

```
px.sunburst(shop , path=['Frequency of Purchases','Age_category'] , values='Age')
```

```
/usr/local/lib/python3.11/dist-packages/plotly/express/_core.py:1727: FutureWarning:

The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retai
```



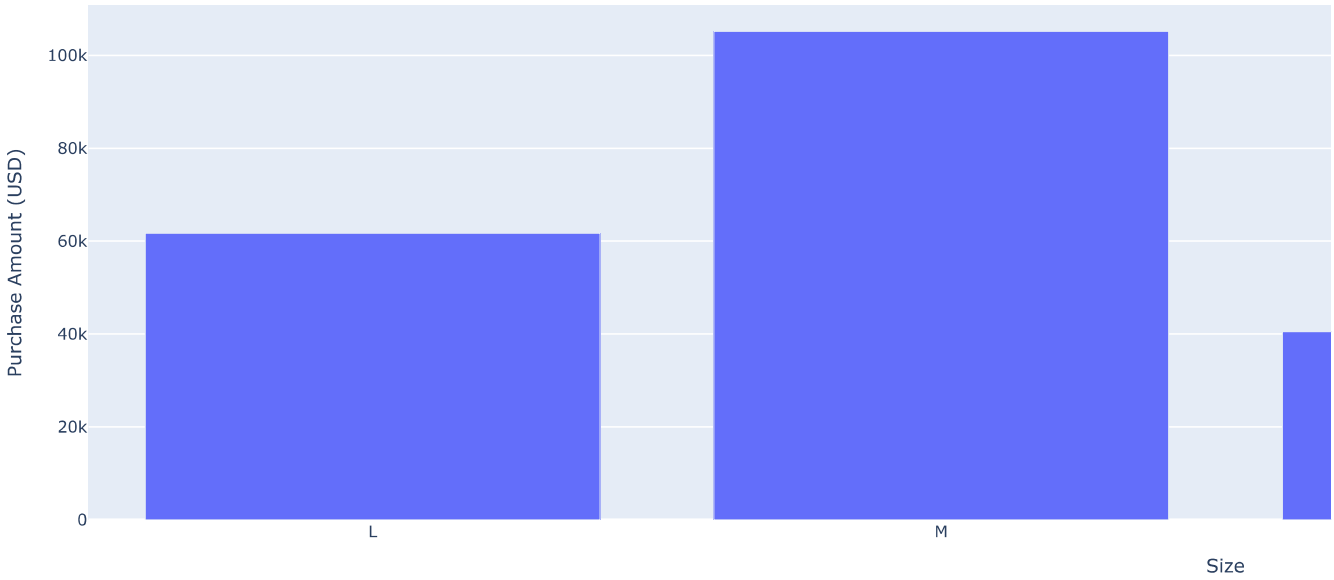
11 Are there any correlations between the size of the product and the purchase amount?

```
shop.columns
```

```
Index(['Customer ID', 'Age', 'Gender', 'Item Purchased', 'Category',
       'Purchase Amount (USD)', 'Location', 'Size', 'Color', 'Season',
       'Review Rating', 'Subscription Status', 'Shipping Type',
       'Discount Applied', 'Promo Code Used', 'Previous Purchases',
       'Payment Method', 'Frequency of Purchases', 'Age_category'],
      dtype='object')
```

```
shop_group = shop.groupby('Size')['Purchase Amount (USD)'].sum().reset_index()
```

```
fig  = px.bar(shop_group , x = 'Size' , y ='Purchase Amount (USD)'  )
fig.show()
```



12 Which shipping type is preferred by customers for different product categories?

```
shop.groupby('Category')['Shipping Type'].value_counts().sort_values(ascending= False)
```

Category	Shipping Type	count
Clothing	Standard	297
	Free Shipping	294
	Next Day Air	293
	Express	290
	Store Pickup	282
	2-Day Shipping	281
Accessories	Store Pickup	217
	Next Day Air	211
	Standard	208
	2-Day Shipping	206
	Express	203
	Free Shipping	195
Footwear	Free Shipping	122
	Standard	100
	Store Pickup	98
	Express	96
	Next Day Air	93
	2-Day Shipping	90
Outerwear	Free Shipping	64
	Express	57
	Store Pickup	53
	Next Day Air	51
	2-Day Shipping	50
	Standard	49

dtype: int64

```
shop['Shipping_Category'] =shop['Shipping Type'].map({'Express': 0, 'Free Shipping': 1, 'Next Day Air': 2,
                                                      'Standard': 3, '2-Day Shipping': 4, 'Store Pickup': 5})
```

```
shop['Category'].unique()
```

```
array(['Clothing', 'Footwear', 'Outerwear', 'Accessories'], dtype=object)
```

```
shop['Category_num'] =shop['Category'].map({'Clothing':1, 'Footwear':2, 'Outerwear':3, 'Accessories':4})
```

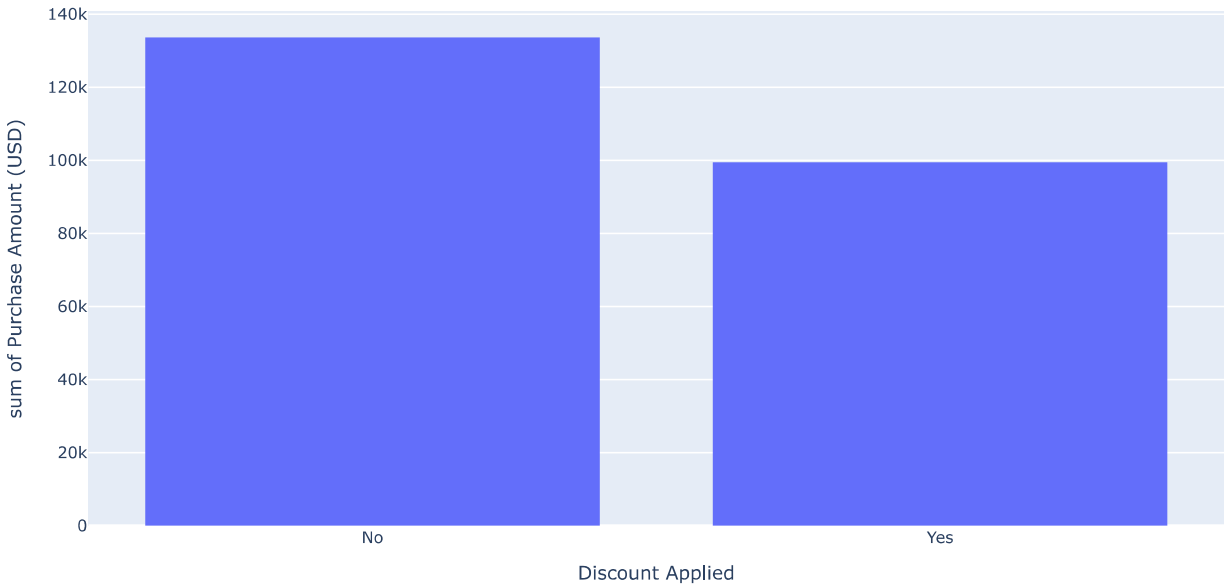
13 How does the presence of a discount affect the purchase decision of customers?

```
shop.columns
```

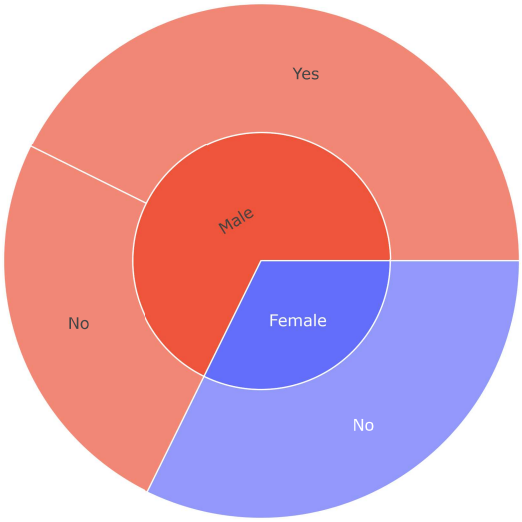
```
Index(['Customer ID', 'Age', 'Gender', 'Item Purchased', 'Category',
       'Purchase Amount (USD)', 'Location', 'Size', 'Color', 'Season',
       'Review Rating', 'Subscription Status', 'Shipping Type',
       'Discount Applied', 'Promo Code Used', 'Previous Purchases',
       'Payment Method', 'Frequency of Purchases', 'Age_category',
       'Shipping_Category', 'Category_num'],
      dtype='object')
```

```
shop_group = shop.groupby('Discount Applied')['Purchase Amount (USD)'].sum().reset_index()
```

```
px.histogram(shop_group , x = 'Discount Applied' , y = 'Purchase Amount (USD)')
```

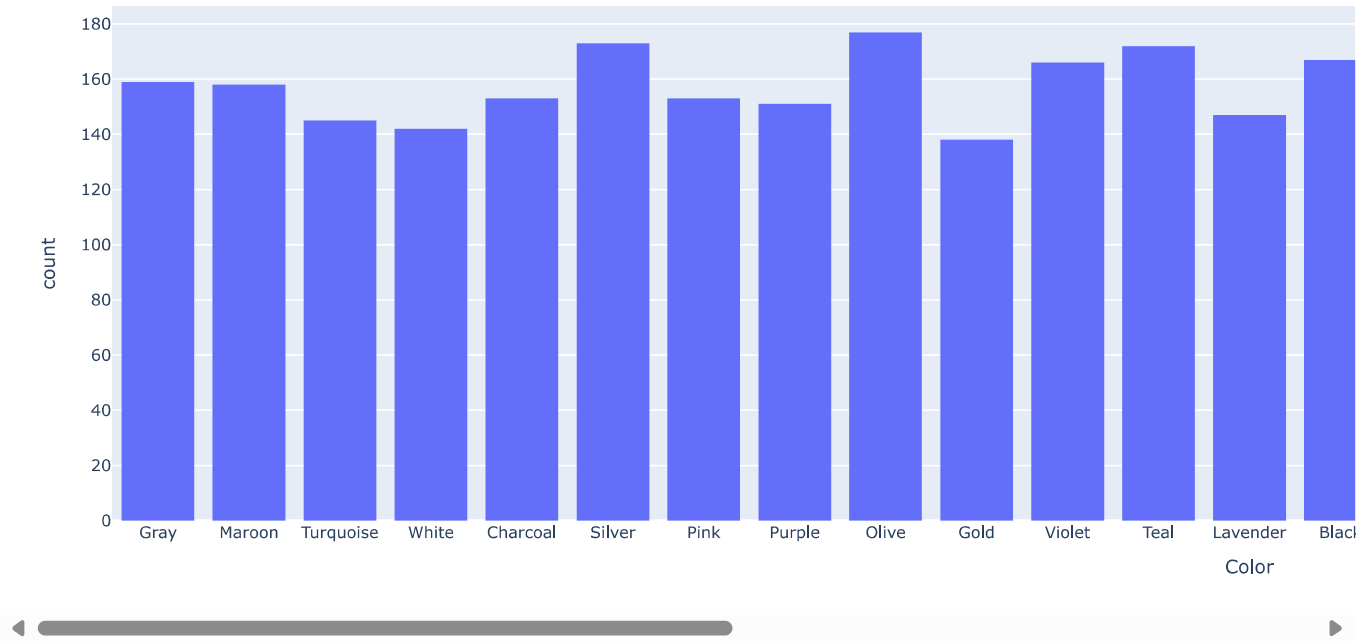



```
fig = px.sunburst(shop , path = ['Gender' , 'Discount Applied'], values='Purchase Amount (USD)' , color= 'Gender')

fig.show()
```



14 Are there any specific colors that are more popular among customers?

```
px.histogram(shop , x = 'Color')
```




```
shop['Color'].value_counts().nlargest(5)
```

	count
Color	
Olive	177
Yellow	174
Silver	173
Teal	172
Green	169

dtype: int64

15 What is the average number of previous purchases made by customers?

```
shop['Previous Purchases'].mean()
```

```
25.35153846153846
```

16 Are there any noticeable differences in purchase behavior between different locations?

```
shop.groupby('Location')['Purchase Amount (USD)'].mean().sort_values(ascending = False)
```

Location	Purchase Amount (USD)
Alaska	67.597222
Pennsylvania	66.567568
Arizona	66.553846
West Virginia	63.876543
Nevada	63.379310
Washington	63.328767
North Dakota	62.891566
Virginia	62.883117
Utah	62.577465
Michigan	62.095890
Tennessee	61.974026
New Mexico	61.901235
Rhode Island	61.444444
Texas	61.194805
Arkansas	61.113924
Illinois	61.054348
Mississippi	61.037500
Massachusetts	60.888889
Iowa	60.884058
North Carolina	60.794872
Wyoming	60.690141
South Dakota	60.514286
New York	60.425287
Ohio	60.376623
Montana	60.250000
Idaho	60.075269
Nebraska	59.448276
New Hampshire	59.422535
Alabama	59.112360
California	59.000000
Indiana	58.924051
Georgia	58.797468
South Carolina	58.407895
Oklahoma	58.346667
Missouri	57.913580
Hawaii	57.723077
Louisiana	57.714286
Oregon	57.337838
Vermont	57.176471
Maine	56.987013
New Jersey	56.746269
Minnesota	56.556818
Colorado	56.293333
Wisconsin	55.946667
Florida	55.852941
Maryland	55.755814
Kentucky	55.721519
Delaware	55.325581