DIWALI SALES REPORT

DATA ANALYSIS PROJECT WITH PYTHON

BY

ROHIT KUMAR MAURYA

SHWETA VISHWAKARMA

SHREYA SINGH

SATAKSHI YADAV

SANJANA MAURYA

SRISHTI BANSAL

Accenture Training DA Phase 2 Batch2 (Group-8)

Challenge Statement

1. Which age group spend maximum amount.

2. Which gender spend maximum amount

3. Which occupation does the customer spend maximum amount?

4. Which state spends the maximum amount on purchasing?

5. Which type of product is sold most?

6. In India which zone has spent highest in Diwali sales?

7. How does the distribution of product categories purchased change as people age?

8. Are married people buying more?

Importing Libraries:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
print("import sucessfully")
```

```
import sucessfully
```

Loading the dataset:

```
import pandas as pd
#df=pd.read_csv("C:/Users/OM/Documents/Diwali_Sales_data.csv")
df = pd.read_csv("C:/Users/Rohit/Documents/Python/Projects/Diwali_Sales_Data.csv", encoding='ISO-8859-1')
df
```

[84]:

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	State	Zone	Occupation	Product_Category	Orders	Amount
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Maharashtra	Western	Healthcare	Auto	1	23952.0
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra Pradesh	Southern	Govt	Auto	3	23934.0
2	1001990	Bindu	P00118542	F	26-35	35	1	Uttar Pradesh	Central	Automobile	Auto	3	23924.0
3	1001425	Sudevi	P00237842	M	0-17	16	0	Karnataka	Southern	Construction	Auto	2	23912.0
4	1000588	Joni	P00057942	M	26-35	28	1	Gujarat	Western	Food Processing	Auto	2	23877.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
11246	1000695	Manning	P00296942	M	18-25	19	1	Maharashtra	Western	Chemical	Office	4	370.0
11247	1004089	Reichenbach	P00171342	M	26-35	33	0	Haryana	Northern	Healthcare	Veterinary	3	367.0
11248	1001209	Oshin	P00201342	F	36-45	40	0	Madhya Pradesh	Central	Textile	Office	4	213.0
11249	1004023	Noonan	P00059442	M	36-45	37	0	Karnataka	Southern	Agriculture	Office	3	206.0
11250	1002744	Brumley	P00281742	F	18-25	19	0	Maharashtra	Western	Healthcare	Office	3	188.0

11251 rows × 13 columns

Accenture Training DA Phase 2 Batch2 (Group-8)

Exploratory Data Analysis and Data Cleaning:

Top-5 Records of Dataset:

```
# view first five row/records
df.head()
```

	User_ID	Cust_name	Product_ID	Gender	Age_Group	Age	Marital_Status	State	Zone	Occupation	Product_Category	Orders	Amount
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Maharashtra	Western	Healthcare	Auto	1	23952.0
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra Pradesh	Southern	Govt	Auto	3	23934.0
2	1001990	Bindu	P00118542	F	26-35	35	1	Uttar Pradesh	Central	Automobile	Auto	3	23924.0
3	1001425	Sudevi	P00237842	M	0-17	16	0	Karnataka	Southern	Construction	Auto	2	23912.0
4	1000588	Joni	P00057942	M	26-35	28	1	Gujarat	Western	Food Processing	Auto	2	23877.0

Last-5 Records of Dataset:

```
# view last five row/records
df.tail()
```

	User_ID	Cust_name	Product_ID	Gender	Age_Group	Age	Marital_Status	State	Zone	Occupation	Product_Category	Orders	Amount
11246	1000695	Manning	P00296942	M	18-25	19	1	Maharashtra	Western	Chemical	Office	4	370.0
11247	1004089	Reichenbach	P00171342	M	26-35	33	0	Haryana	Northern	Healthcare	Veterinary	3	367.0
11248	1001209	Oshin	P00201342	F	36-45	40	0	Madhya Pradesh	Central	Textile	Office	4	213.0
11249	1004023	Noonan	P00059442	M	36-45	37	0	Karnataka	Southern	Agriculture	Office	3	206.0
11250	1002744	Brumley	P00281742	F	18-25	19	0	Maharashtra	Western	Healthcare	Office	3	188.0

Total number of Rows and Columns in the Dataset:

```
#total no. of rows & columns
print("No of rows and columns in the dataset:")
df.shape
```

```
No of rows and columns in the dataset:
(11251, 13)
```

Information about the Dataset:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11251 entries, 0 to 11250
Data columns (total 13 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   User_ID           11251 non-null  int64
 1   Cust_name         11251 non-null  object
 2   Product_ID        11251 non-null  object
 3   Gender            11251 non-null  object
 4   Age Group         11251 non-null  object
 5   Age               11251 non-null  int64
 6   Marital_Status    11251 non-null  int64
 7   State             11251 non-null  object
 8   Zone              11251 non-null  object
 9   Occupation        11251 non-null  object
 10  Product_Category  11251 non-null  object
 11  Orders            11251 non-null  int64
 12  Amount            11239 non-null  float64
dtypes: float64(1), int64(4), object(8)
memory usage: 1.1+ MB
```

Summary of the Dataset:

```
df.describe()
```

	User_ID	Age	Marital_Status	Orders	Amount
count	1.125100e+04	11251.000000	11251.000000	11251.000000	11239.000000
mean	1.003004e+06	35.421207	0.420318	2.489290	9453.610858
std	1.716125e+03	12.754122	0.493632	1.115047	5222.355869
min	1.000001e+06	12.000000	0.000000	1.000000	188.000000
25%	1.001492e+06	27.000000	0.000000	1.500000	5443.000000
50%	1.003065e+06	33.000000	0.000000	2.000000	8109.000000
75%	1.004430e+06	43.000000	1.000000	3.000000	12675.000000
max	1.006040e+06	92.000000	1.000000	4.000000	23952.000000

```
df.describe(include='object')
```

	Cust_name	Product_ID	Gender	Age Group	State	Zone	Occupation	Product_Category
count	11251	11251	11251	11251	11251	11251	11251	11251
unique	1250	2351	2	7	16	5	15	18
top	Vishakha	P00265242	F	26-35	Uttar Pradesh	Central	IT Sector	Clothing & Apparel
freq	42	53	7842	4543	1946	4296	1588	2655

Checking Null Values in Dataset:

```
df.isna().sum()
```

```
User_ID             0
Cust_name           0
Product_ID          0
Gender              0
Age Group           0
Age                 0
Marital_Status      0
State               0
Zone                0
Occupation          0
Product_Category    0
Orders              0
Amount             12
dtype: int64
```

Removing Null Values:

```
df = df.dropna(subset=['Amount'])
# to delete the null value from the  amount  column
```

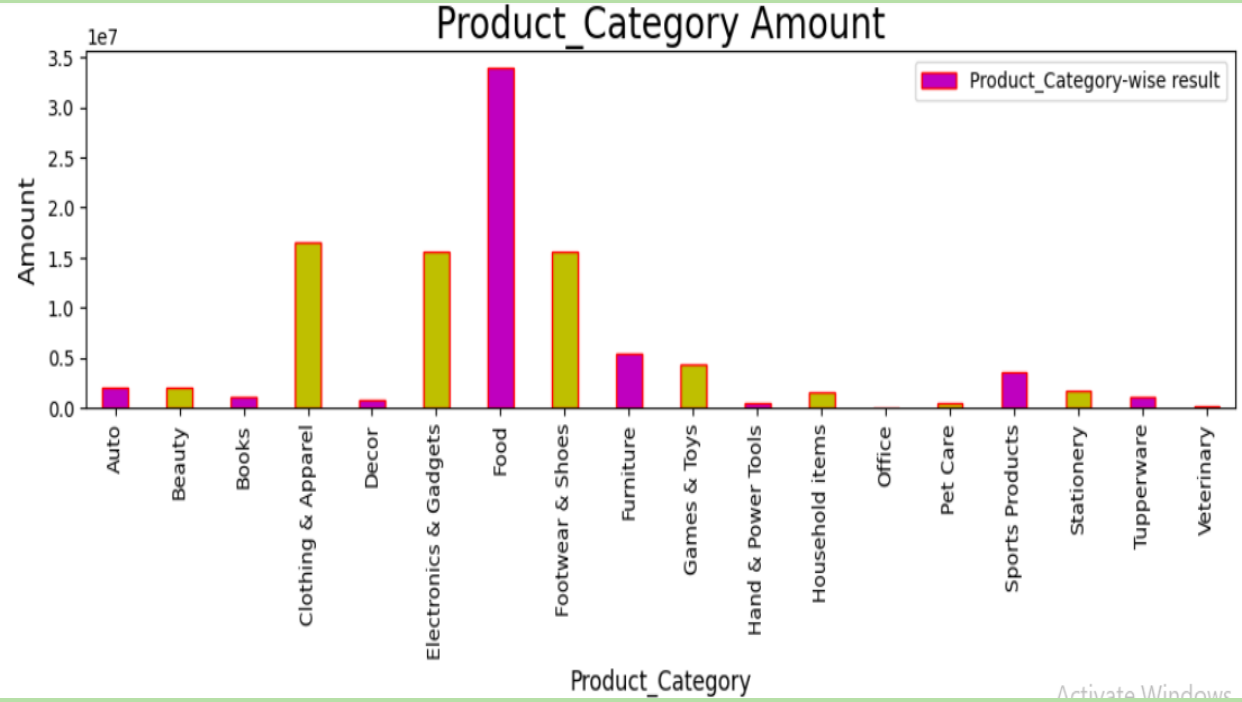
DATA ANALYSIS AND VISUALIZATION:

Product Category Wise Spent Amount:

```
Product_Category_total = df.groupby('Product_Category')['Amount'].sum()
plt.figure(figsize=(12,4))
plt.xlabel("Product_Category", fontsize=12)
plt.ylabel("Amount", fontsize=13)
plt.title("Product_Category Amount", fontsize=20)

Product_Category_total.plot(kind='bar', color=['m', 'y'], width=0.4, align='center', edgecolor='red', label='Product_Category-wise result')

plt.legend()
plt.show()
```

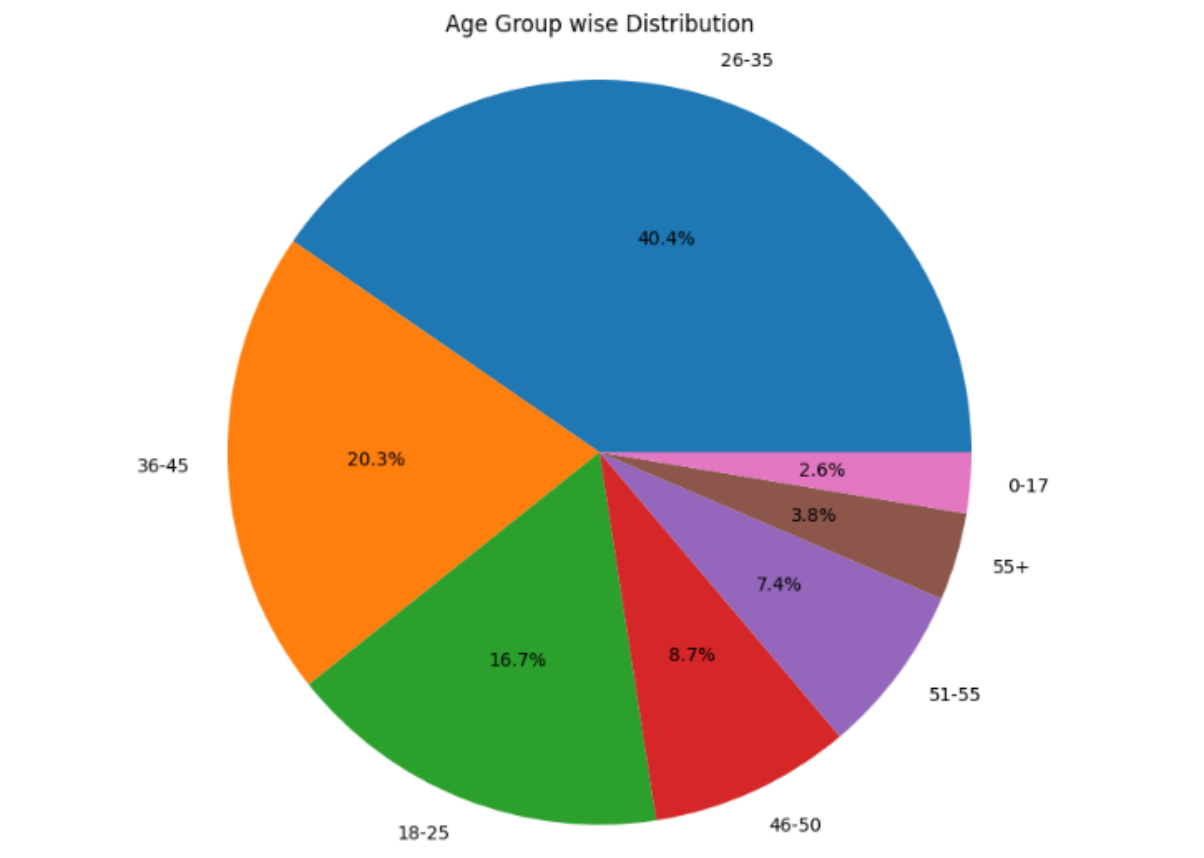


Age Group Wise Distribution:

```
Age_Group_counts = df['Age Group'].value_counts()
plt.figure(figsize=(12,8))
# Create a pie chart of the moon phase counts
plt.pie(Age_Group_counts, labels=Age_Group_counts.index, autopct='%1.1f%%')

# Customize the pie chart
plt.title('Age Group wise Distribution')
plt.axis('equal')

# Show the plot
plt.show()
```

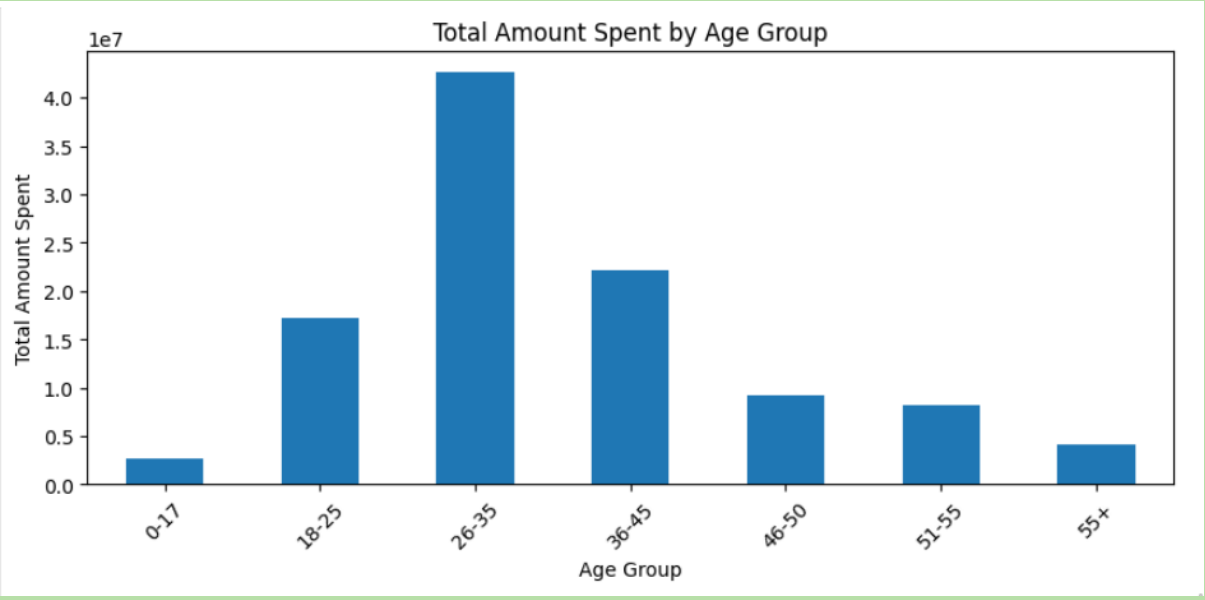


Renaming Columns:

```
# Renaming columns
df=df.rename(columns={'Age Group':'Age_Group'})
df.info()
```

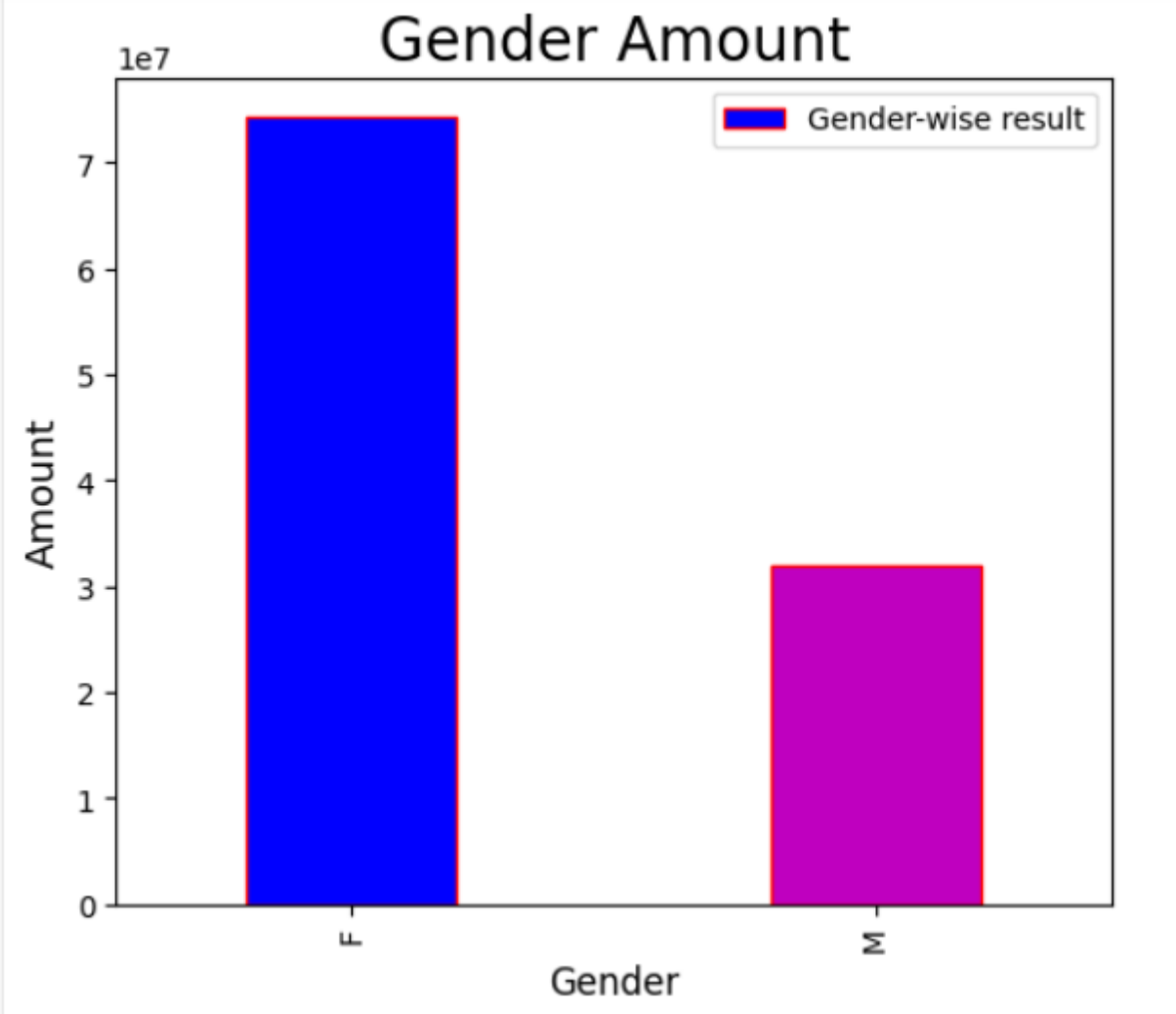
Total Amount Spend by Age_Group:

```
age_group_amount = df.groupby('Age_Group')['Amount'].sum()
plt.figure(figsize=(10, 6))
age_group_amount.plot(kind='bar')
plt.xlabel('Age Group')
plt.ylabel('Total Amount Spent')
plt.title('Total Amount Spent by Age Group')
plt.xticks(rotation=45)
plt.show()
```



Gender Amount Plotting:

```
plt.figure(figsize=(6,5))
gender_amount = df.groupby('Gender')['Amount'].sum()
plt.xlabel("Gender", fontsize=12)
plt.ylabel("Amount", fontsize=13)
plt.title("Gender Amount", fontsize=20)
gender_amount.plot(kind='bar', color=['b', 'm'], width=0.4, align='center', edgecolor='red', label='Gender-wise result')
plt.legend()
plt.show()
```

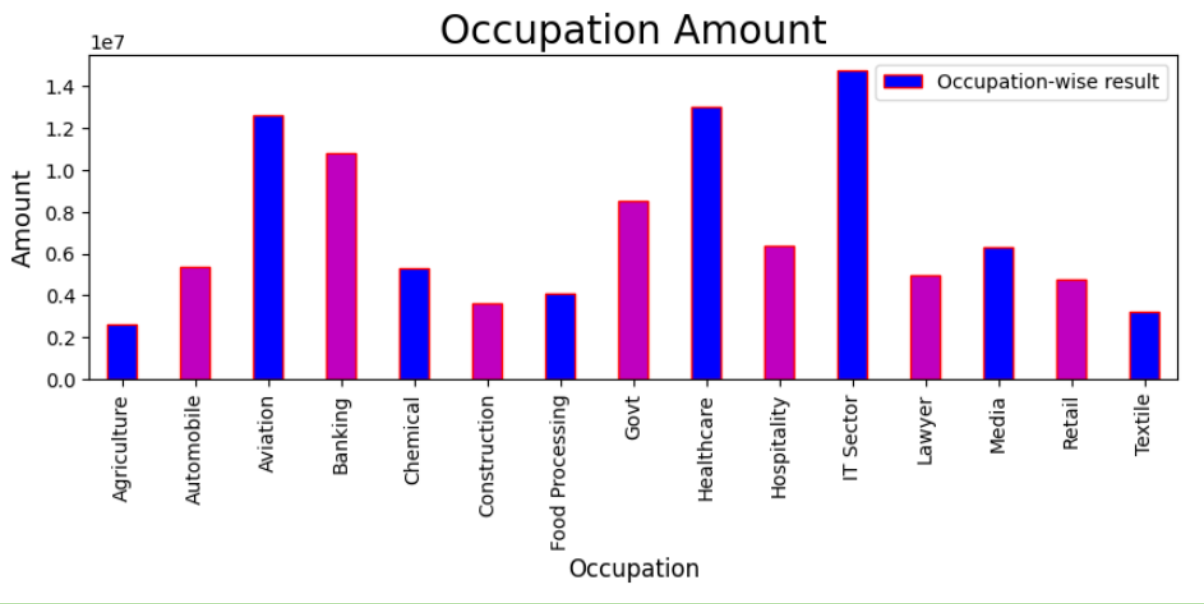


Occupation Amount Wise Plotting:

```
Occupation_total = df.groupby('Occupation')['Amount'].sum()
plt.figure(figsize=(10,3))
plt.xlabel("Occupation", fontsize=12)
plt.ylabel("Amount", fontsize=13)
plt.title("Occupation Amount", fontsize=20)

Occupation_total.plot(kind='bar', color=['b', 'm'], width=0.4, align='center', edgecolor='red', label='Occupation-wise result')

plt.legend()
plt.show()
```

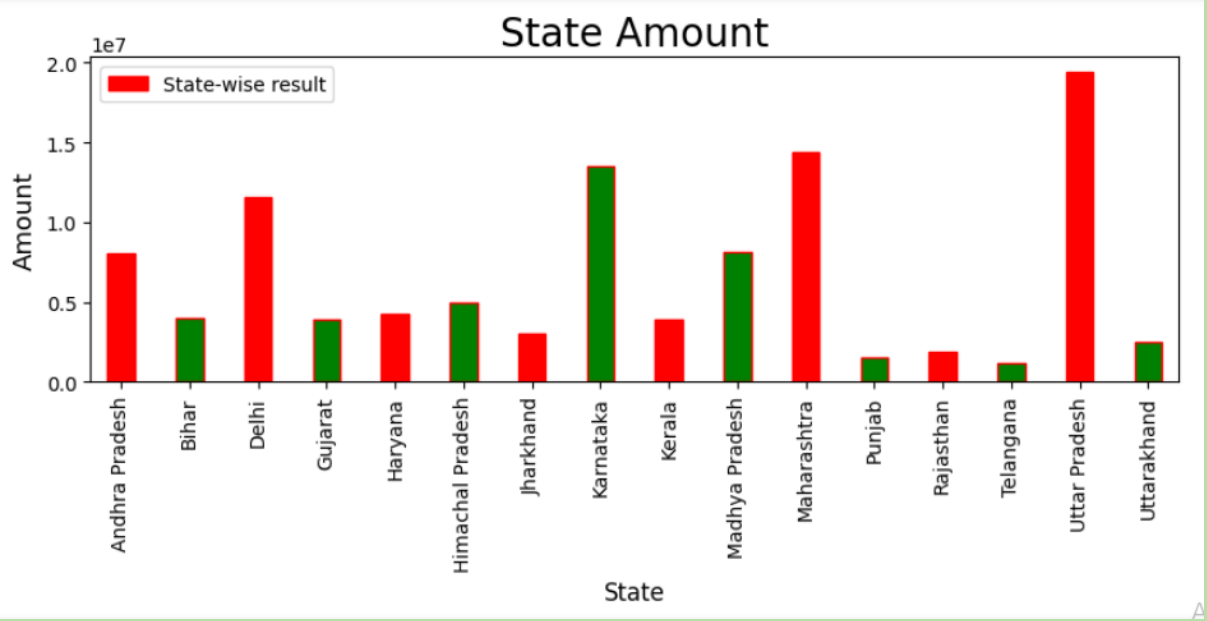


State vs Amount Plotting:

```
State_total = df.groupby('State')['Amount'].sum()
plt.figure(figsize=(10,3))
plt.xlabel("State", fontsize=12)
plt.ylabel("Amount", fontsize=13)
plt.title("State Amount", fontsize=20)

State_total.plot(kind='bar', color=['r', 'g'], width=0.4, align='center', edgecolor='red', label='State-wise result')

plt.legend()
plt.show()
```



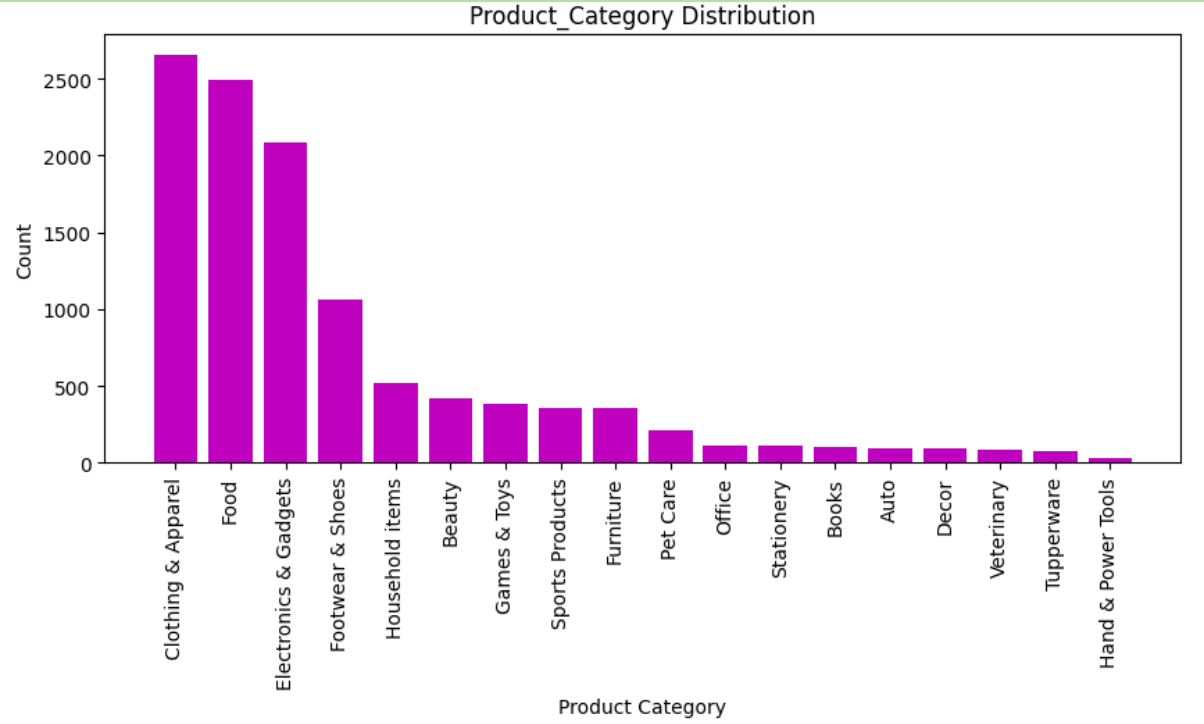
Product_Category Distribution Wise Plotting:

```
# Create a bar plot of the Product_Category counts
Product_Category_counts = df['Product_Category'].value_counts()
plt.figure(figsize=(10, 4))
plt.bar(Product_Category_counts.index, Product_Category_counts, color="m")

# Customize the bar plot
plt.title('Product_Category Distribution')
plt.xlabel('Product Category')
plt.ylabel('Count')

# Rotate x-axis labels for better readability (optional)
plt.xticks(rotation=90)

# Show the bar plot
plt.show()
```

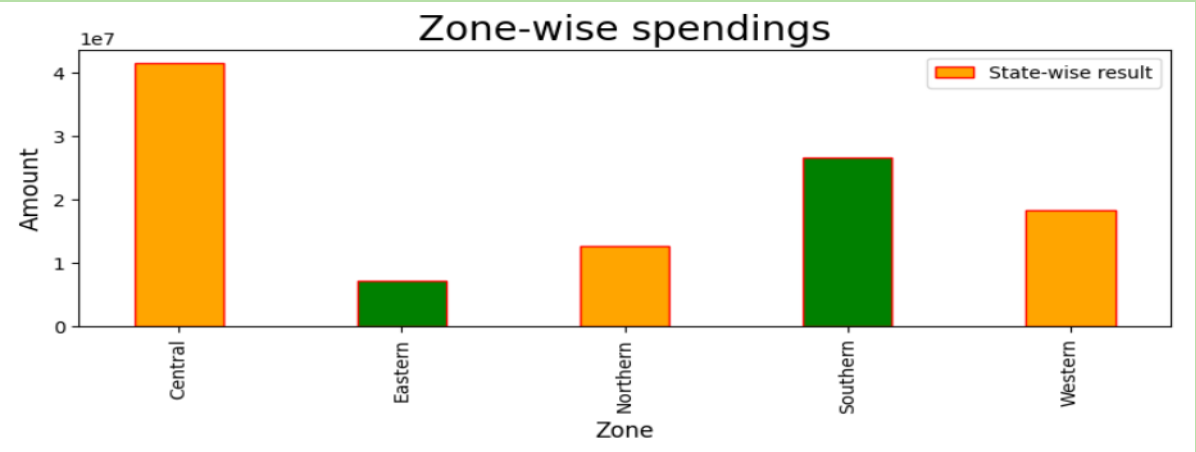


Zone-Wise Spendings Plotting:

```
State_total = df.groupby('Zone')['Amount'].sum()
plt.figure(figsize=(10,3))
plt.xlabel("Zone", fontsize=12)
plt.ylabel("Amount", fontsize=13)
plt.title("Zone-wise spendings", fontsize=20)

State_total.plot(kind='bar', color=['orange', 'g'], width=0.4, align='center', edgecolor='red', label='State-wise result')

plt.legend()
plt.show()
```



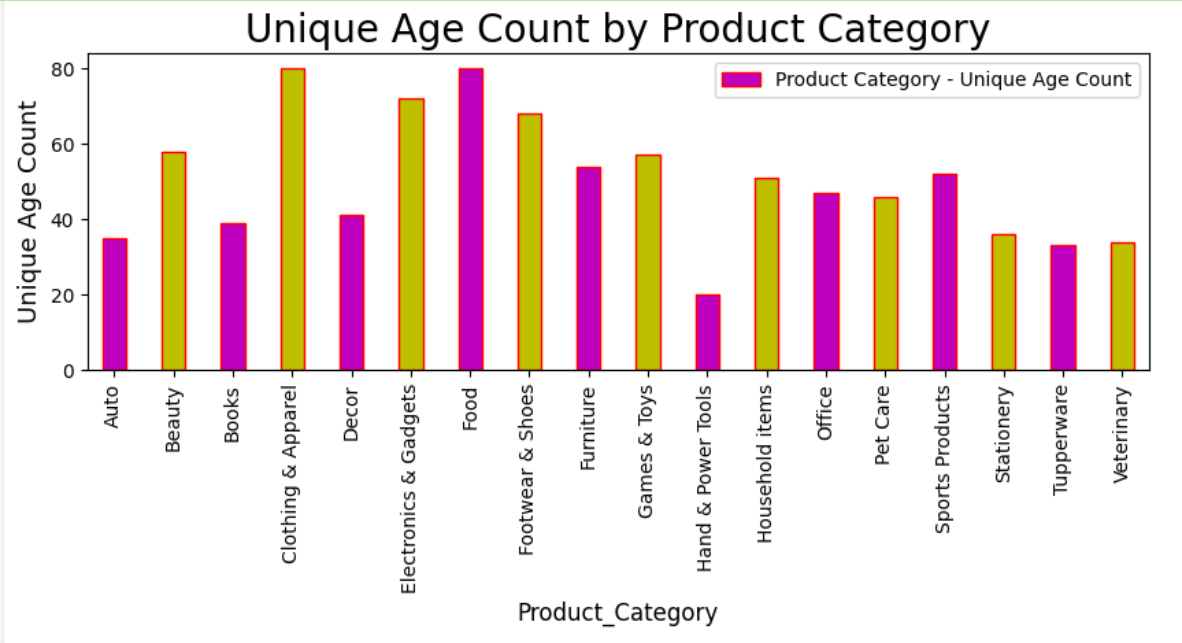
Age vs Product_Category Plotting:

```
# Grouping the DataFrame by 'Product_Category' and counting the unique 'Age' values for each category.
Product_Category_total = df.groupby('Product_Category')['Age'].nunique()

plt.figure(figsize=(10, 3))  # Adjust the figure size as needed
plt.xlabel("Product_Category", fontsize=12)
plt.ylabel("Unique Age Count", fontsize=13)
plt.title("Unique Age Count by Product Category", fontsize=20)

# Creating a bar chart with 'Product_Category' on the x-axis and 'Unique Age Count' on the y-axis.
Product_Category_total.plot(kind='bar', color=['m', 'y'], width=0.4, align='center', edgecolor='red', label='Product Category - Unique Age Count')

plt.legend()
plt.show()
```

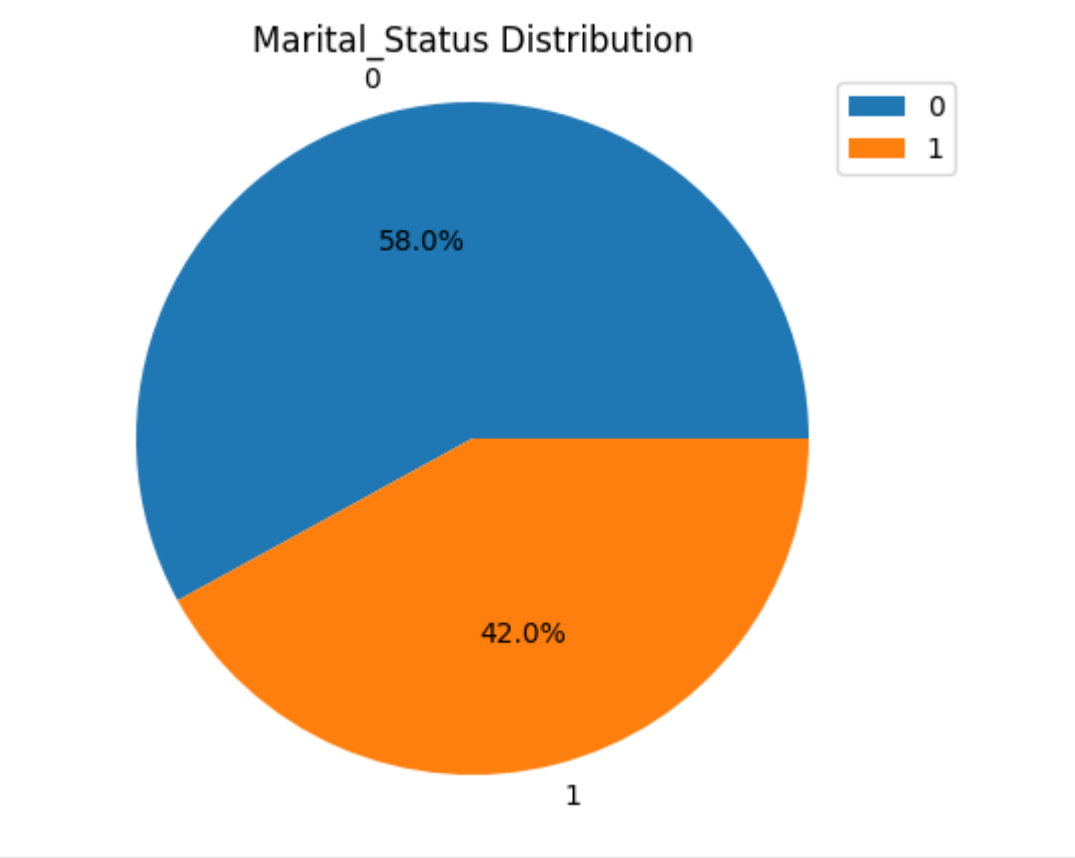


Marital_Status Distribution-Wise Plotting:

```
Marital_Status_counts = df['Marital_Status'].value_counts()

plt.pie(Marital_Status_counts, labels=Marital_Status_counts.index, autopct='%1.1f%%')

# Customize the pie chart
plt.title('Marital_Status Distribution')
plt.axis('equal')
plt.legend()
# Show the plot
plt.show()
```

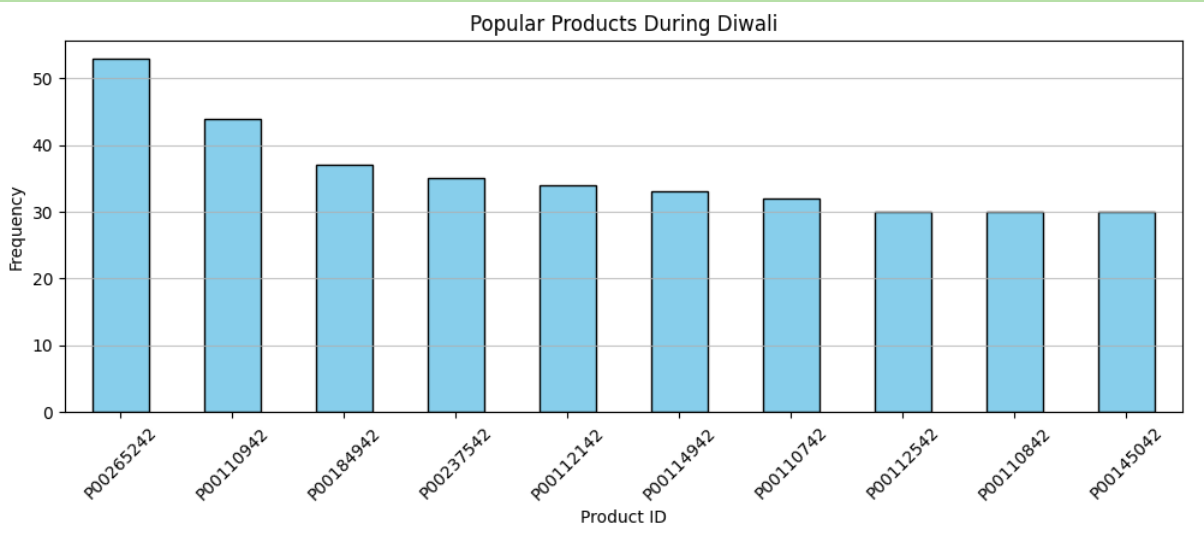


Order Distribution-Wise Plotting:

```
plt.figure(figsize=(10, 6))
plt.hist(df['Orders'], bins=20, color='skyblue', edgecolor='black')
plt.xlabel('Number of Orders')
plt.ylabel('Frequency')
plt.title('Distribution of Number of Orders during Diwali')
plt.grid(axis='y', alpha=0.75)
plt.show()
```

Product ID Wise Plotting:

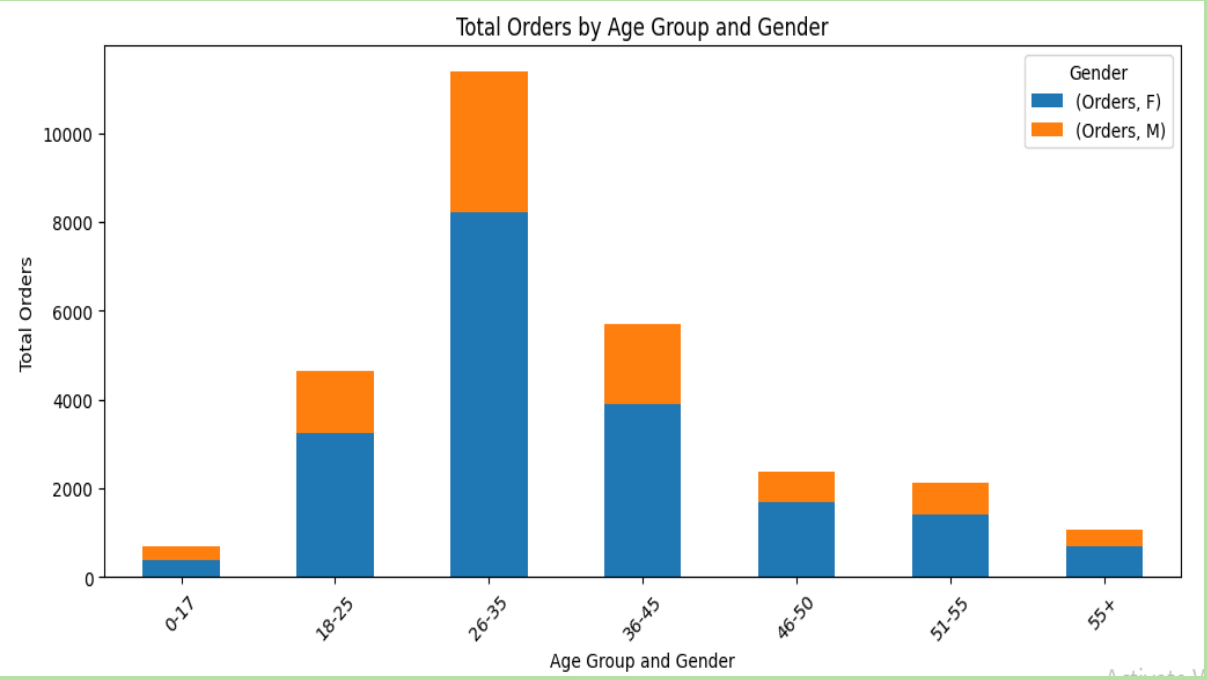
```
# Count the occurrences of each Product ID
popular_products = df['Product_ID'].value_counts().head(10)
plt.figure(figsize=(10, 6))
popular_products.plot(kind='bar', color='skyblue', edgecolor='black')
plt.xlabel('Product ID')
plt.ylabel('Frequency')
plt.title('Popular Products During Diwali')
plt.xticks(rotation=45)
plt.grid(axis='y', alpha=0.75)
plt.show()
```



Total Orders by Age Group and Gender Wise Plotting:

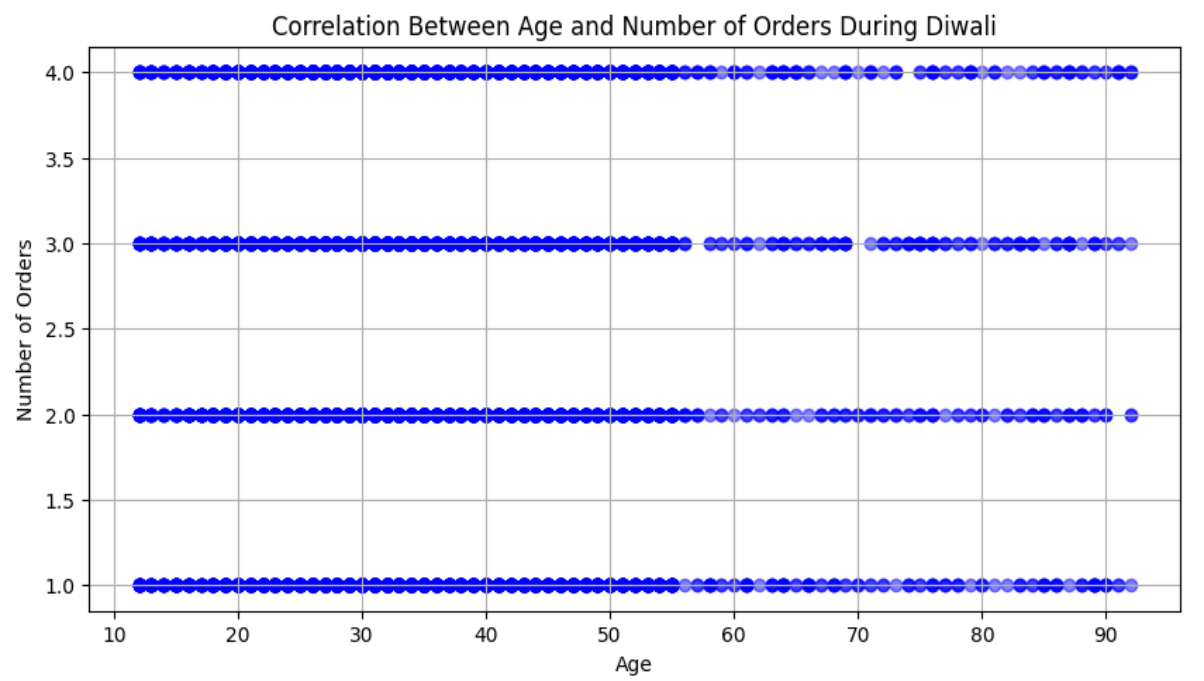
```
# Pivot the data to create a table of total Orders by Age Group and Gender
pivot_table = pd.pivot_table(df, values='Orders', index=['Age_Group', 'Gender'], aggfunc='sum')

# Create a stacked bar chart
pivot_table.unstack().plot(kind='bar', stacked=True, figsize=(12, 6))
plt.xlabel('Age Group and Gender')
plt.ylabel('Total Orders')
plt.title('Total Orders by Age Group and Gender')
plt.xticks(rotation=45)
plt.legend(title='Gender')
plt.show()
```



Age vs Orders Plotting:

```
plt.figure(figsize=(10, 6))
plt.scatter(df['Age'], df['Orders'], alpha=0.5, color='blue')
plt.xlabel('Age')
plt.ylabel('Number of Orders')
plt.title('Correlation Between Age and Number of Orders During Diwali')
plt.grid(True)
plt.show()
```



Findings of Challenge Statements

1. 26-35 Age Group people spend maximum amount in Diwali Sale.

2. Females spend maximum amount in Diwali Sale.

3. IT Sector Occupation people spend maximum amount in Diwali Sale.

4. Uttar Pradesh spends maximum amount in purchasing in Diwali Sale.

5. Clothing & Apparel products in Product_Category are sold highest in the Sale.

6. Central zone in India has spent highest in Diwali sales.

7. i) Younger people are more likely to purchase fashion and beauty products.

   ii) Middle-aged people are more likely to purchase products for their homes and families.

   iii) Older people are more likely to purchase products for their health and well-being.

8. Unmarried people are buying more products than married people.

- Overall, the number of orders during Diwali is highest on the first and sixth days, and lowest on the third and fifth days.
  This is likely because people are more likely to shop on the first and sixth days of Diwali, which are the most important days of the festival.

- Popular product in Diwali is Product_ID P00265242.

- Overall, male customers placed more orders than female customers.

- The age group with the most orders was 26-35, followed by 18-25 and 36-45.

- The age group with the least orders was 55+, followed by 51-55 and 10-17.
- The gender gap in orders was highest in the 26-35 age group, where male customers placed twice as many orders as female customers.

- The age group with the most orders was 26-35, but there was no significant difference in the number of orders placed by other age groups.

Conclusion

Here are some specific takeaways for businesses:

- Target the 26-35 age group, as they are the biggest spenders during Diwali.
- Focus on female customers, as they also spend more money than males on Diwali shopping.
- Consider targeting people in the IT sector, as they are another group that spends heavily during Diwali.
- Focus on Uttar Pradesh, as it is the state with the highest spending on Diwali shopping.
- Feature clothing and apparel products prominently in your marketing campaigns, as they are the most popular category of products purchased during Diwali.
- Consider targeting the central zone of India, as it is the region with the highest spending on Diwali shopping.

Businesses should also be aware of the following trends:

- Younger people are more likely to purchase fashion and beauty products.
- Middle-aged people are more likely to purchase products for their homes and families.
- Older people are more likely to purchase products for their health and well-being.
- Unmarried people are buying more products than married people.
- The number of orders during Diwali is highest on the first and sixth days, and lowest on the third and fifth days.