Data Analysis Portfolio Project:

Marketing Insights for E-Commerce Company

Business Problem Statement:

- A rapidly growing e-commerce company aims to transition from intuition-based marketing to a data-driven approach.
- By analyzing customer demographics, transaction data, marketing spend, and discount details from 2019, the company seeks to gain a comprehensive understanding of customer behavior.
- The objectives are to optimize marketing campaigns across various channels, leverage data insights to enhance customer retention, predict customer lifetime value, and ultimately drive sustainable revenue growth.

Data description:

https://drive.google.com/drive/folders/1VXaZSDFqN_Zi3FxucRlthz97Etl1CYfJ?usp=sharing
(https://drive.google.com/drive/folders/1VXaZSDFqN_Zi3FxucRlthz97Etl1CYfJ?usp=sharing)

Online_Sales.csv:

This file contains actual orders data (point of Sales data) at transaction level with the below variables.

1. CustomerID: Customer unique ID
2. Transaction_ID: Transaction Unique ID
3. Transaction_Date: Date of Transaction
4. Product_SKU: SKU ID – Unique Id for product
5. Product_Description: Product Description
6. Product_Cateogry: Product Category
7. Quantity: Number of items ordered
8. Avg_Price: Price per one quantity
9. Delivery_Charges: Charges for delivery
10. Coupon_Status: Any discount coupon applied

Customers_Data.csv:

This file contains customer's demographics.

1. CustomerID: Customer Unique ID
2. Gender: Gender of customer
3. Location: Location of Customer
4. Tenure_Months: Tenure in Months

Discount_Coupon.csv:

Discount coupons have been given for different categories in different months

1. Month: Discount coupon applied in that month
2. Product_Category: Product category
3. Coupon_Code: Coupon Code for given Category and given month
4. Discount_pct: Discount Percentage for given coupon

Marketing_Spend.csv:

Marketing spend on both offline & online channels on day wise.

1. Date: Date
2. Offline_Spend: Marketing spend on offline channels like TV, Radio, NewsPapers, hoardings etc.
3. Online_Spend: Marketing spend on online channels like Google keywords, facebook etc.

Tax_Amount.csv:

GST Details for given category

1. Product_Category: Product Category
2. GST: Percentage of GST

Approach:

Through this project, we expect to leverage a range of data analysis techniques to uncover actionable insights that propel the e-commerce company towards significant customer retention and revenue growth.

This includes:

- **Identifying key customer segments and behaviors:** Utilizing descriptive statistics and segmentation techniques to understand what drives customer acquisition and churn.
- **Evaluating marketing campaign effectiveness:** Employing hypothesis testing to assess the impact of online and offline marketing efforts on customer behavior and revenue.
- **Optimizing discount strategies:** Analyzing the influence of discounts and promotions on revenue and customer engagement to identify optimal pricing strategies.
- **Predicting customer lifetime value:** Implementing data-driven models to anticipate future customer value and prioritize retention efforts.
- **Unveiling cross-selling opportunities:** Performing market basket analysis to discover frequently co-purchased products and inform product placement strategies.
- **Formulating data-driven recommendations:** Presenting clear and compelling visualizations and reports that translate insights into actionable marketing strategies for maximizing customer retention and revenue growth.

Solution Methodology:

1. Data Cleaning and Preprocessing:

- **Data Cleaning:** Ensure data quality by identifying and handling missing values, inconsistencies, and outliers in each dataset.
- It is crucial to begin by calculating the invoice amount or revenue for each transaction using this formula, This establishes the foundation for revenue analysis. **Invoice Value = ((Quantity Avg_price) (1 - Discount_pct) * (1 + GST)) + Delivery_Charges.**

2. Exploratory Data Analysis (EDA):

- **Customer Acquisition & Retention:** Analyze trends in customer acquisition and churn across different customer demographics (gender, location, tenure) and timeframes (monthly). Tools like time series analysis and segmentation can be helpful here.
- **Marketing Campaign Impact:** Explore the relationship between marketing spend (online & offline) and customer behavior (orders, revenue) to assess campaign effectiveness. Utilize techniques like hypothesis testing to validate your findings.
- **Discount Analysis:** Investigate how discounts and promotions affect revenue and customer engagement. Analyze KPIs like average order value and customer acquisition cost across different discount structures.

3. Deeper Analysis:

- **Seasonality & Trends:** Identify seasonal trends and patterns in sales data across different timeframes (month, week, day) to inform future marketing strategies.
- **Calculate key performance indicators (KPIs)** KPIs like revenue, number of orders, and average order value across various dimensions (category, month, week, day).
- **Marketing Spend & Revenue:** Calculate revenue, marketing spend, and delivery charges by month to understand their correlation. This can reveal areas for optimization.
- **Product & Customer Relationships:** Analyze co-purchased products through market basket analysis. This will uncover cross-selling opportunities and inform product placement strategies. **(Optional)**
- **Customer Lifetime Value (CLTV):** Implement predictive models to estimate the future value of each customer. This helps prioritize retention efforts for high-value customers. **(Optional)**

4. Cohort Analysis:

- **Create customer cohorts based on their acquisition month:** Track their behavior (orders, revenue) over time to identify the cohort with the highest retention rate. This reveals valuable customer acquisition trends.

5. Actionable Insights & Recommendations:

- **Insights:** Translate your findings into clear and compelling visualizations and reports.
- **Recommendations:** Formulate data-driven recommendations for optimizing marketing strategies, improving customer retention, and maximizing revenue growth for the e-commerce company.

Importing Libraries:

In [1]:
```
import numpy as np  # linear algebra

import pandas as pd  # data processing

import seaborn as sns  # data visualization

import matplotlib.pyplot as plt  # data visualization

import plotly.graph_objs as go  # data visualization

import plotly.express as px  # data visualization

from plotly.subplots import make_subplots  # data visualization

from datetime import datetime  # datetime

from datetime import date  # date

import calendar  # calendar

import warnings  # control how warnings are handled
warnings.filterwarnings('ignore')
```

Reading the dataset:

In [2]:
```
customers_data = pd.read_csv('Customers_Data.csv')
discount_coupon = pd.read_csv('Discount_Coupon.csv')
marketing_spend = pd.read_csv('Marketing_Spend.csv')
online_sales = pd.read_csv('Online_Sales.csv')
tax_amount = pd.read_csv('Tax_amount.csv')
```

Looking at the dataset:

In [3]:
```
# ANSI escape code for bold text
bold_start = "\033[1m"
bold_end = "\033[0m"

print(f"{bold_start}Customers Data Dataframe Head:{bold_end}")
print(customers_data.head())

print(f"{bold_start}\nDiscount Coupon Dataframe Head:{bold_end}")
print(discount_coupon.head())

print(f"{bold_start}\nMarketing Spend Dataframe Head:{bold_end}")
print(marketing_spend.head())

print(f"{bold_start}\nTax Amount Dataframe Head:{bold_end}")
print(tax_amount.head())
```

```
Customers Data Dataframe Head:
   CustomerID Gender    Location  Tenure_Months
0       17850      M     Chicago             12
1       13047      M  California             43
2       12583      M     Chicago             33
3       13748      F  California             30
4       15100      M  California             49

Discount Coupon Dataframe Head:
  Month Product_Category Coupon_Code  Discount_pct
0   Jan          Apparel      SALE10            10
1   Feb          Apparel      SALE20            20
2   Mar          Apparel      SALE30            30
3   Jan         Nest-USA      ELEC10            10
4   Feb         Nest-USA      ELEC20            20

Marketing Spend Dataframe Head:
       Date  Offline_Spend  Online_Spend
0  1/1/2019           4500       2424.50
1  1/2/2019           4500       3480.36
2  1/3/2019           4500       1576.38
3  1/4/2019           4500       2928.55
4  1/5/2019           4500       4055.30

Tax Amount Dataframe Head:
  Product_Category  GST
0         Nest-USA  10%
1           Office  10%
2          Apparel  18%
3             Bags  18%
4        Drinkware  18%
```

In [4]:
```
# ANSI escape code for bold text
bold_start = "\033[1m"
bold_end = "\033[0m"

print(f"{bold_start}Customers Data Dataframe Tail:{bold_end}")
print(customers_data.tail())

print(f"{bold_start}\nDiscount Coupon Dataframe Tail:{bold_end}")
print(discount_coupon.tail())

print(f"{bold_start}\nMarketing Spend Dataframe Tail:{bold_end}")
print(marketing_spend.tail())

print(f"{bold_start}\nTax Amount Dataframe Tail:{bold_end}")
print(tax_amount.tail())
```

```
Customers Data Dataframe Tail:
      CustomerID Gender    Location  Tenure_Months
1463       14438      F    New York             41
1464       12956      F     Chicago             48
1465       15781      M  New Jersey             19
1466       14410      F    New York             45
1467       14600      F  California              7

Discount Coupon Dataframe Tail:
    Month      Product_Category Coupon_Code  Discount_pct
199   Nov  Notebooks & Journals        NJ20            20
200   Dec  Notebooks & Journals        NJ30            30
201   Oct               Android       AND10            10
202   Nov               Android       AND20            20
203   Dec               Android       AND30            30

Marketing Spend Dataframe Tail:
           Date  Offline_Spend  Online_Spend
360  12/27/2019           4000       3396.87
361  12/28/2019           4000       3246.84
362  12/29/2019           4000       2546.58
363  12/30/2019           4000        674.31
364  12/31/2019           4000       2058.75

Tax Amount Dataframe Tail:
   Product_Category  GST
15        More Bags  18%
16       Housewares  12%
17          Android  10%
18      Accessories  10%
19             Nest   5%
```

In [5]:
```
online_sales.head()
```

Out[5]:

	CustomerID	Transaction_ID	Transaction_Date	Product_SKU	Product_Description	Product_Category	Quantity	Avg_Price	Delivery_Charges	Coup
0	17850	16679	1/1/2019	GGOENEBJ079499	Nest Learning Thermostat 3rd Gen-USA - Stainle...	Nest-USA	1	153.71	6.5	
1	17850	16680	1/1/2019	GGOENEBJ079499	Nest Learning Thermostat 3rd Gen-USA - Stainle...	Nest-USA	1	153.71	6.5	
2	17850	16681	1/1/2019	GGOEGFKQ020399	Google Laptop and Cell Phone Stickers	Office	1	2.05	6.5	
3	17850	16682	1/1/2019	GGOEGAAB010516	Google Men's 100% Cotton Short Sleeve Hero Tee...	Apparel	5	17.53	6.5	
4	17850	16682	1/1/2019	GGOEGBJL013999	Google Canvas Tote Natural/Navy	Bags	1	16.50	6.5	

In [6]:
```
online_sales.tail()
```

Out[6]:

	CustomerID	Transaction_ID	Transaction_Date	Product_SKU	Product_Description	Product_Category	Quantity	Avg_Price	Delivery_Charges	C
52919	14410	48493	12/31/2019	GGOENEBB078899	Nest Cam Indoor Security Camera - USA	Nest-USA	1	121.30	6.50	
52920	14410	48494	12/31/2019	GGOEGAEB091117	Google Zip Hoodie Black	Apparel	1	48.92	6.50	
52921	14410	48495	12/31/2019	GGOENEBQ084699	Nest Learning Thermostat 3rd Gen-USA - White	Nest-USA	1	151.88	6.50	
52922	14600	48496	12/31/2019	GGOENEBQ079199	Nest Protect Smoke + CO White Wired Alarm-USA	Nest-USA	5	80.52	6.50	
52923	14600	48497	12/31/2019	GGOENEBQ079099	Nest Protect Smoke + CO White Battery Alarm-USA	Nest-USA	4	80.52	19.99	

Shape of the dataset:

In [7]:
```
# ANSI escape code for bold text
bold_start = "\033[1m"
bold_end = "\033[0m"

print(f"{bold_start}Customers Dataframe shape is:{bold_end}", customers_data.shape)
print(f"{bold_start}\nDiscount Coupon Dataframe shape is:{bold_end}", discount_coupon.shape)
print(f"{bold_start}\nMarketing Spend Dataframe shape is:{bold_end}", marketing_spend.shape)
print(f"{bold_start}\nOnline Sales Dataframe shape is:{bold_end}", online_sales.shape)
print(f"{bold_start}\nTax Amount Dataframe shape is:{bold_end}", tax_amount.shape)
```

Customers Dataframe shape is: (1468, 4)

Discount Coupon Dataframe shape is: (204, 4)

Marketing Spend Dataframe shape is: (365, 3)

Online Sales Dataframe shape is: (52924, 10)

Tax Amount Dataframe shape is: (20, 2)

In [8]:
```
# ANSI escape code for bold text
bold_start = "\033[1m"
bold_end = "\033[0m"

print(f"{bold_start}Customers:{bold_end} \n# rows: {customers_data.shape[0]}, # columns: {customers_data.shape[1]}")
print(f"{bold_start}\nDiscount Coupon:{bold_end} \n# rows: {discount_coupon.shape[0]}, # columns: {discount_coupon.shape[1]}
print(f"{bold_start}\nMarketing Spend:{bold_end} \n# rows: {marketing_spend.shape[0]}, # columns: {marketing_spend.shape[1]}
print(f"{bold_start}\nOnline Sales:{bold_end} \n# rows: {online_sales.shape[0]}, # columns: {online_sales.shape[1]}")
print(f"{bold_start}\nTax Amount:{bold_end} \n# rows: {tax_amount.shape[0]}, # columns: {tax_amount.shape[1]}")
```

Customers:
rows: 1468, # columns: 4

Discount Coupon:
rows: 204, # columns: 4

Marketing Spend:
rows: 365, # columns: 3

Online Sales:
rows: 52924, # columns: 10

Tax Amount:
rows: 20, # columns: 2

Columns in the Dataset:

In [9]:
```
# ANSI escape code for bold text
bold_start = "\033[1m"
bold_end = "\033[0m"

print(f"{bold_start}Customers Columns are:{bold_end}", customers_data.columns)
print(f"{bold_start}\nDiscount Coupon Columns are:{bold_end}", discount_coupon.columns)
print(f"{bold_start}\nMarketing Spend Columns are:{bold_end}", marketing_spend.columns)
print(f"{bold_start}\nOnline Sales Columns are:{bold_end}", online_sales.columns)
print(f"{bold_start}\nTax Amount Columns are:{bold_end}", tax_amount.columns)
```

Customers Columns are: Index(['CustomerID', 'Gender', 'Location', 'Tenure_Months'], dtype='object')

Discount Coupon Columns are: Index(['Month', 'Product_Category', 'Coupon_Code', 'Discount_pct'], dtype='object')

Marketing Spend Columns are: Index(['Date', 'Offline_Spend', 'Online_Spend'], dtype='object')

Online Sales Columns are: Index(['CustomerID', 'Transaction_ID', 'Transaction_Date', 'Product_SKU',
'Product_Description', 'Product_Category', 'Quantity', 'Avg_Price',
'Delivery_Charges', 'Coupon_Status'],
dtype='object')

Tax Amount Columns are: Index(['Product_Category', 'GST'], dtype='object')

Datatype of the columns:

In [10]:
```
bold_start = "\033[1m"
bold_end = "\033[0m"

print(f"{bold_start}Customers Columns Data Types are:\n{bold_end}", customers_data.dtypes)
print(f"{bold_start}\nDiscount Coupon Columns Data Types are:\n{bold_end}", discount_coupon.dtypes)
print(f"{bold_start}\nMarketing Spend Columns Data Types are:\n{bold_end}", marketing_spend.dtypes)
print(f"{bold_start}\nOnline Sales Columns Data Types are:\n{bold_end}", online_sales.dtypes)
print(f"{bold_start}\nTax Amount Columns Data Types are:\n{bold_end}", tax_amount.dtypes)
```

```
Customers Columns Data Types are:
 CustomerID        int64
Gender            object
Location          object
Tenure_Months      int64
dtype: object

Discount Coupon Columns Data Types are:
 Month                object
Product_Category     object
Coupon_Code          object
Discount_pct          int64
dtype: object

Marketing Spend Columns Data Types are:
 Date              object
Offline_Spend      int64
Online_Spend     float64
dtype: object

Online Sales Columns Data Types are:
 CustomerID               int64
Transaction_ID            int64
Transaction_Date         object
Product_SKU              object
Product_Description      object
Product_Category         object
Quantity                  int64
Avg_Price               float64
Delivery_Charges        float64
Coupon_Status            object
dtype: object

Tax Amount Columns Data Types are:
 Product_Category    object
GST                 object
dtype: object
```

Basic information about the dataset:

In [11]:
```
# ANSI escape code for bold text
bold_start = "\033[1m"
bold_end = "\033[0m"

print(f"{bold_start}Customers Dataframe Info:{bold_end}")
print(customers_data.info())

print(f"{bold_start}\nDiscount Coupon Dataframe Info:{bold_end}")
print(discount_coupon.info())

print(f"{bold_start}\nMarketing Spend Dataframe Info:{bold_end}")
print(marketing_spend.info())

print(f"{bold_start}\nOnline Sales Dataframe Info:{bold_end}")
print(online_sales.info())

print(f"{bold_start}\nTax Amount Dataframe Info:{bold_end}")
print(tax_amount.info())
```

Customers Dataframe Info:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1468 entries, 0 to 1467
Data columns (total 4 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   CustomerID     1468 non-null   int64
 1   Gender         1468 non-null   object
 2   Location       1468 non-null   object
 3   Tenure_Months  1468 non-null   int64
dtypes: int64(2), object(2)
memory usage: 46.0+ KB
None
```

Discount Coupon Dataframe Info:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 204 entries, 0 to 203
Data columns (total 4 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   Month             204 non-null    object
 1   Product_Category  204 non-null    object
 2   Coupon_Code       204 non-null    object
 3   Discount_pct      204 non-null    int64
dtypes: int64(1), object(3)
memory usage: 6.5+ KB
None
```

Marketing Spend Dataframe Info:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 365 entries, 0 to 364
Data columns (total 3 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Date           365 non-null    object
 1   Offline_Spend  365 non-null    int64
 2   Online_Spend   365 non-null    float64
dtypes: float64(1), int64(1), object(1)
memory usage: 8.7+ KB
None
```

Online Sales Dataframe Info:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 52924 entries, 0 to 52923
Data columns (total 10 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   CustomerID           52924 non-null  int64
 1   Transaction_ID       52924 non-null  int64
 2   Transaction_Date     52924 non-null  object
 3   Product_SKU          52924 non-null  object
 4   Product_Description  52924 non-null  object
 5   Product_Category     52924 non-null  object
 6   Quantity             52924 non-null  int64
 7   Avg_Price            52924 non-null  float64
 8   Delivery_Charges     52924 non-null  float64
 9   Coupon_Status        52924 non-null  object
dtypes: float64(2), int64(3), object(5)
memory usage: 4.0+ MB
None
```

Tax Amount Dataframe Info:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20 entries, 0 to 19
Data columns (total 2 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   Product_Category  20 non-null     object
 1   GST               20 non-null     object
dtypes: object(2)
memory usage: 452.0+ bytes
None
```

Basic statistical information about the dataset:

In [12]:
```
# ANSI escape code for bold text
bold_start = "\033[1m"
bold_end = "\033[0m"

print(f"{bold_start}Customers Dataframe Statistical Info:{bold_end}")
print(customers_data.describe())

print(f"{bold_start}\nDiscount Coupon Dataframe Statistical Info:{bold_end}")
print(discount_coupon.describe())

print(f"{bold_start}\nMarketing Spend Dataframe Statistical Info:{bold_end}")
print(marketing_spend.describe())

print(f"{bold_start}\nOnline Sales Dataframe Statistical Info:{bold_end}")
print(online_sales.describe())

print(f"{bold_start}\nTax Amount Dataframe Statistical Info:{bold_end}")
print(tax_amount.describe())
```

Customers Dataframe Statistical Info:

	CustomerID	Tenure_Months
count	1468.000000	1468.000000
mean	15314.386240	25.912125
std	1744.000367	13.959667
min	12346.000000	2.000000
25%	13830.500000	14.000000
50%	15300.000000	26.000000
75%	16882.250000	38.000000
max	18283.000000	50.000000

Discount Coupon Dataframe Statistical Info:

	Discount_pct
count	204.000000
mean	20.000000
std	8.185052
min	10.000000
25%	10.000000
50%	20.000000
75%	30.000000
max	30.000000

Marketing Spend Dataframe Statistical Info:

	Offline_Spend	Online_Spend
count	365.000000	365.000000
mean	2843.561644	1905.880740
std	952.292448	808.856853
min	500.000000	320.250000
25%	2500.000000	1258.600000
50%	3000.000000	1881.940000
75%	3500.000000	2435.120000
max	5000.000000	4556.930000

Online Sales Dataframe Statistical Info:

	CustomerID	Transaction_ID	Quantity	Avg_Price	\
count	52924.00000	52924.000000	52924.000000	52924.000000	
mean	15346.70981	32409.825675	4.497638	52.237646	
std	1766.55602	8648.668977	20.104711	64.006882	
min	12346.00000	16679.000000	1.000000	0.390000	
25%	13869.00000	25384.000000	1.000000	5.700000	
50%	15311.00000	32625.500000	1.000000	16.990000	
75%	16996.25000	39126.250000	2.000000	102.130000	
max	18283.00000	48497.000000	900.000000	355.740000	

	Delivery_Charges
count	52924.000000
mean	10.517630
std	19.475613
min	0.000000
25%	6.000000
50%	6.000000
75%	6.500000
max	521.360000

Tax Amount Dataframe Statistical Info:

	Product_Category	GST
count	20	20
unique	20	4
top	Nest-USA	10%
freq	1	7

In [13]:
```
# ANSI escape code for bold text
bold_start = "\033[1m"
bold_end = "\033[0m"

print(f"{bold_start}Customers Dataframe Statistical Info:{bold_end}")
print(customers_data.describe(include = 'object'))

print(f"{bold_start}\nDiscount Coupon Dataframe Statistical Info:{bold_end}")
print(discount_coupon.describe(include = 'object'))

print(f"{bold_start}\nMarketing Spend Dataframe Statistical Info:{bold_end}")
print(marketing_spend.describe(include = 'object'))

print(f"{bold_start}\nOnline Sales Dataframe Statistical Info:{bold_end}")
print(online_sales.describe(include = 'object'))

print(f"{bold_start}\nTax Amount Dataframe Statistical Info:{bold_end}")
print(tax_amount.describe(include = 'object'))
```

Customers Dataframe Statistical Info:

	Gender	Location
count	1468	1468
unique	2	5
top	F	California
freq	934	464

Discount Coupon Dataframe Statistical Info:

	Month	Product_Category	Coupon_Code
count	204	204	204
unique	12	17	48
top	Jan	Apparel	EXTRA10
freq	17	12	8

Marketing Spend Dataframe Statistical Info:

	Date
count	365
unique	365
top	1/1/2019
freq	1

Online Sales Dataframe Statistical Info:

	Transaction_Date	Product_SKU	Product_Description	Product_Category	Coupon_Status
count	52924	52924	52924	52924	52924
unique	365	1145	404	20	3
top	11/27/2019	GGOENEBJ079499	Nest Learning Thermostat 3rd Gen-USA - Stainle...	Apparel	Clicked
freq	335	3511	3511	18126	26926

Tax Amount Dataframe Statistical Info:

	Product_Category	GST
count	20	20
unique	20	4
top	Nest-USA	10%
freq	1	7

1. Data Cleaning and Preprocessing:

1A. Missing value detection and perform imputation:

In [14]:
```
# ANSI escape code for bold text
bold_start = "\033[1m"
bold_end = "\033[0m"

print(f"{bold_start}Customers DataFrame Missing Value Info:{bold_end}")
print(customers_data.isnull().sum())

print(f"{bold_start}\nDiscount Coupon DataFrame Missing Value Info:{bold_end}")
print(discount_coupon.isnull().sum())

print(f"{bold_start}\nMarketing Spend DataFrame Missing Value Info:{bold_end}")
print(marketing_spend.isnull().sum())

print(f"{bold_start}\nOnline Sales DataFrame Missing Value Info:{bold_end}")
print(online_sales.isnull().sum())

print(f"{bold_start}\nTax Amount DataFrame Missing Value Info:{bold_end}")
print(tax_amount.isnull().sum())
```

```
Customers DataFrame Missing Value Info:
CustomerID       0
Gender           0
Location         0
Tenure_Months    0
dtype: int64

Discount Coupon DataFrame Missing Value Info:
Month               0
Product_Category    0
Coupon_Code         0
Discount_pct        0
dtype: int64

Marketing Spend DataFrame Missing Value Info:
Date            0
Offline_Spend   0
Online_Spend    0
dtype: int64

Online Sales DataFrame Missing Value Info:
CustomerID             0
Transaction_ID         0
Transaction_Date       0
Product_SKU            0
Product_Description    0
Product_Category       0
Quantity               0
Avg_Price              0
Delivery_Charges       0
Coupon_Status          0
dtype: int64

Tax Amount DataFrame Missing Value Info:
Product_Category    0
GST                 0
dtype: int64
```

5A. Observations:

- There are no missing values in the dataset.

1B. Identify and remove duplicate records:

In [15]:
```
# ANSI escape code for bold text
bold_start = "\033[1m"
bold_end = "\033[0m"

print(f"{bold_start}Customers DataFrame Duplicates Info:{bold_end}")
print(customers_data.duplicated())

print(f"{bold_start}\nDiscount Coupon DataFrame Duplicates Info:{bold_end}")
print(discount_coupon.duplicated())

print(f"{bold_start}\nMarketing Spend DataFrame Duplicates Info:{bold_end}")
print(marketing_spend.duplicated())

print(f"{bold_start}\nOnline Sales DataFrame Duplicates Info:{bold_end}")
print(online_sales.duplicated())

print(f"{bold_start}\nTax Amount DataFrame Duplicates Info:{bold_end}")
print(tax_amount.duplicated())
```

Customers DataFrame Duplicates Info:
```
0       False
1       False
2       False
3       False
4       False
        ...
1463    False
1464    False
1465    False
1466    False
1467    False
Length: 1468, dtype: bool
```

Discount Coupon DataFrame Duplicates Info:
```
0      False
1      False
2      False
3      False
4      False
       ...
199    False
200    False
201    False
202    False
203    False
Length: 204, dtype: bool
```

Marketing Spend DataFrame Duplicates Info:
```
0      False
1      False
2      False
3      False
4      False
       ...
360    False
361    False
362    False
363    False
364    False
Length: 365, dtype: bool
```

Online Sales DataFrame Duplicates Info:
```
0        False
1        False
2        False
3        False
4        False
         ...
52919    False
52920    False
52921    False
52922    False
52923    False
Length: 52924, dtype: bool
```

Tax Amount DataFrame Duplicates Info:
```
0     False
1     False
2     False
3     False
4     False
5     False
6     False
7     False
8     False
9     False
10    False
11    False
12    False
13    False
14    False
15    False
16    False
17    False
18    False
19    False
dtype: bool
```

In [16]:
```
# ANSI escape code for bold text
bold_start = "\033[1m"
bold_end = "\033[0m"

print(f"{bold_start}Are there any duplicates in Customers DataFrame:{bold_end}")
print(np.any(customers_data.duplicated()))

print(f"{bold_start}\nAre there any duplicates in Discount Coupon DataFrame:{bold_end}")
print(np.any(discount_coupon.duplicated()))

print(f"{bold_start}\nAre there any duplicates in Marketing Spend DataFrame:{bold_end}")
print(np.any(marketing_spend.duplicated()))

print(f"{bold_start}\nAre there any duplicates in Online Sales DataFrame:{bold_end}")
print(np.any(online_sales.duplicated()))

print(f"{bold_start}\nAre there any duplicates in Tax Amount DataFrame:{bold_end}")
print(np.any(tax_amount.duplicated()))
```

Are there any duplicates in Customers DataFrame:
False

Are there any duplicates in Discount Coupon DataFrame:
False

Are there any duplicates in Marketing Spend DataFrame:
False

Are there any duplicates in Online Sales DataFrame:
False

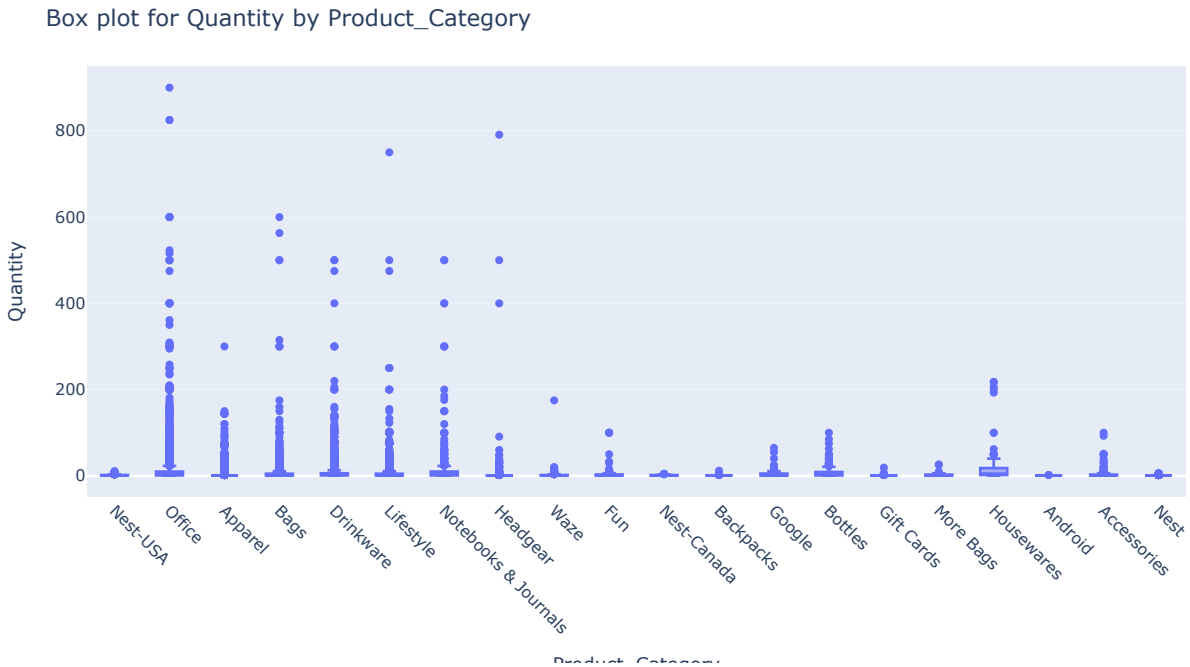
Are there any duplicates in Tax Amount DataFrame:
False

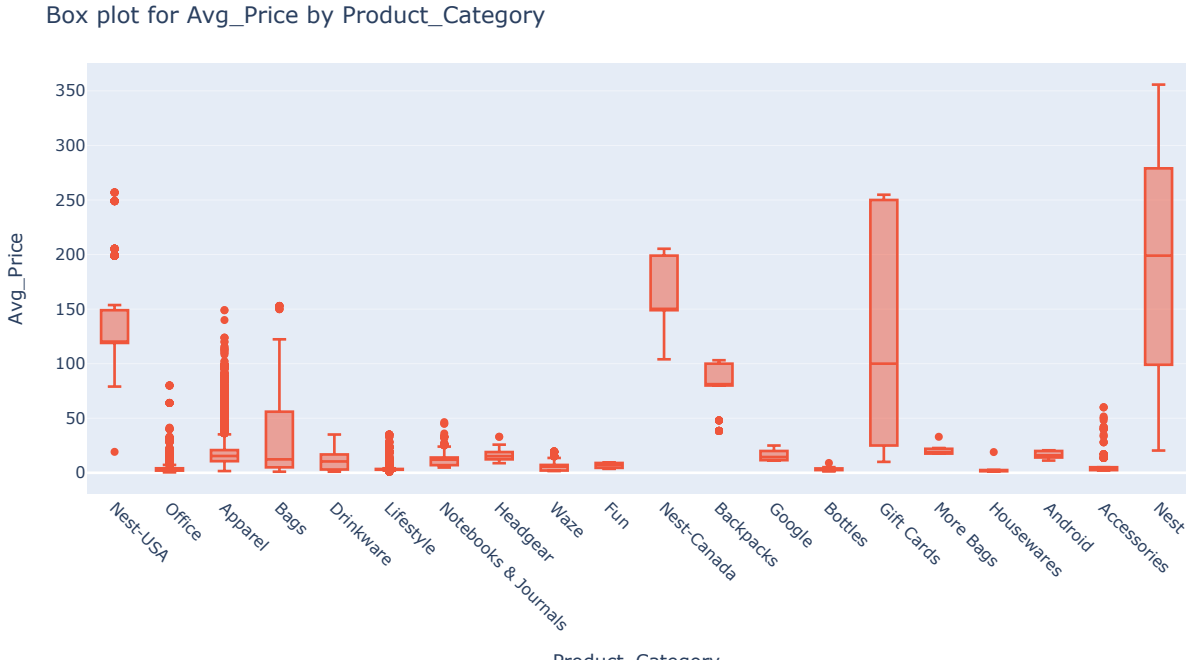
5A. Observations:

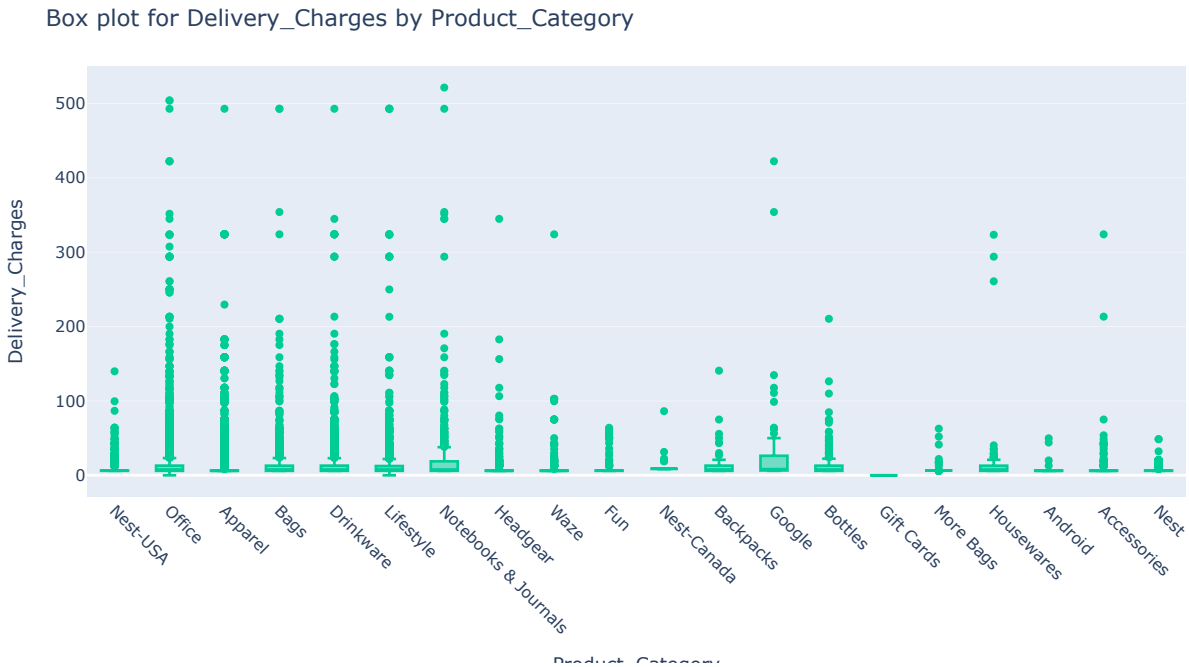
- There are no duplicate values in the dataset.

1B. Plot Box Plots and identify outliers in each dataset:

In [17]:
```
# Box plot for Quantity with custom color and rotated x-axis labels
fig = px.box(online_sales,
             x = 'Product_Category',
             y = 'Quantity',
             color_discrete_sequence = ['#636EFA'])

fig.update_layout(title = 'Box plot for Quantity by Product_Category',
                  xaxis_title = 'Product_Category',
                  yaxis_title = 'Quantity')

fig.update_xaxes(tickangle = 45)
fig.show()

# Box plot for Avg_Price with custom color and rotated x-axis labels
fig = px.box(online_sales,
             x = 'Product_Category',
             y = 'Avg_Price',
             color_discrete_sequence = ['#EF553B'])

fig.update_layout(title = 'Box plot for Avg_Price by Product_Category',
                  xaxis_title = 'Product_Category',
                  yaxis_title = 'Avg_Price')

fig.update_xaxes(tickangle = 45)
fig.show()

# Box plot for Delivery_Charges with custom color and rotated x-axis labels
fig = px.box(online_sales,
             x = 'Product_Category',
             y = 'Delivery_Charges',
             color_discrete_sequence = ['#00CC96'])

fig.update_layout(title = 'Box plot for Delivery_Charges by Product_Category',
                  xaxis_title = 'Product_Category',
                  yaxis_title = 'Delivery_Charges')

fig.update_xaxes(tickangle = 45)
fig.show()
```







5A. Observations:

- **Delivery Charges by Product Category-**
  - **High Variability in Delivery Charges:** Delivery charges exhibit a high degree of variability across product categories. Categories like 'Nest-USA', 'Office', and 'Apparel' have many outliers with delivery charges going up to 500.
  - **Low to Moderate Delivery Charges:** Most categories have lower delivery charges with median values close to zero. For example, 'Bottles', 'Gift Cards', and 'More Bags' have relatively low median delivery charges.
  - **Outliers:** Significant outliers are present in several categories, especially in 'Nest-USA', 'Office', and 'Apparel', indicating some instances of very high delivery charges.
- **Average Price by Product Category-**
  - **Wide Range of Average Prices:** The average prices vary widely across product categories. Categories like 'Office', 'Nest-USA', and 'Apparel' have higher average prices with a considerable spread, whereas 'Drinkware', 'Headgear', and 'Fun' have lower and more consistent average prices.
  - **High Priced Categories:** 'Nest' and 'Accessories' show higher average prices with significant spread, indicating a variety of premium products in these categories.
  - **Low Priced Categories:** Categories such as 'Gift Cards' and 'More Bags' have lower average prices with relatively narrow ranges, suggesting standardized pricing within these categories.
- **Quantity by Product Category-**
  - **High Quantity Orders in Specific Categories:** Categories like 'Office', 'Apparel', and 'Drinkware' have higher quantities ordered, with several outliers indicating bulk purchases.
  - **Low Quantity Orders in Other Categories:** Categories like 'Gift Cards', 'More Bags', and 'Accessories' have lower quantities ordered, suggesting these items are typically bought in smaller quantities.
  - **Outliers:** Significant outliers exist in categories such as 'Office' and 'Apparel', showing some instances of very large orders.

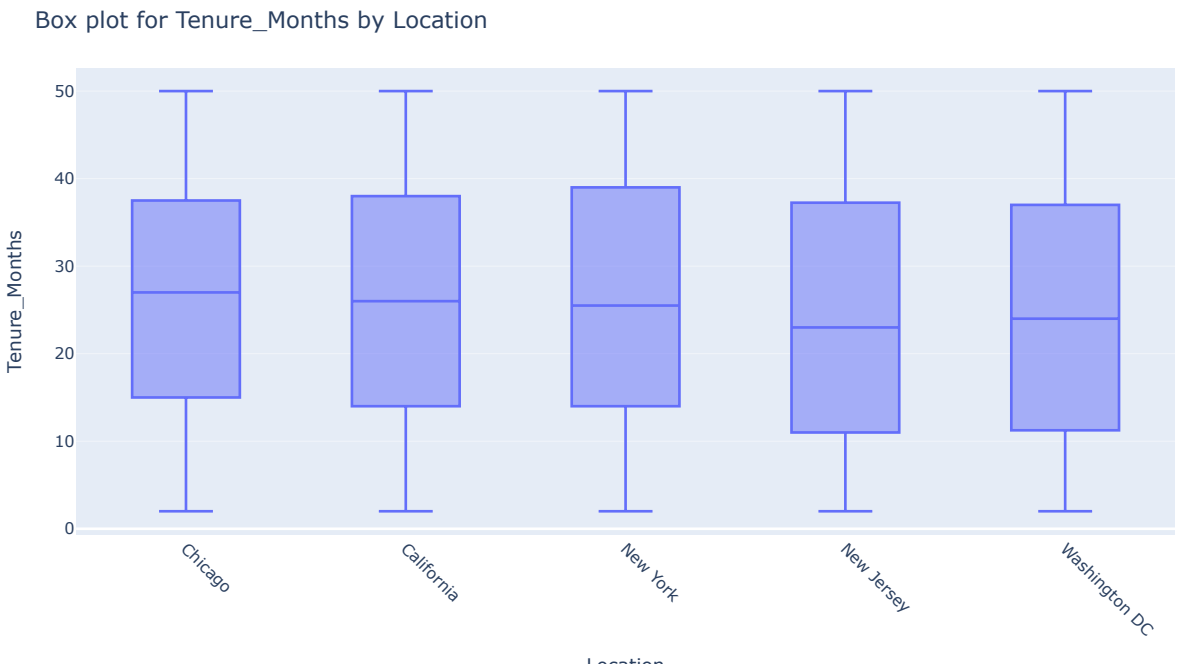
5B. Insights:

- **Delivery Charges Optimization:** High variability and outliers in delivery charges suggest potential inefficiencies. Optimizing delivery logistics could reduce costs and standardize delivery charges across categories.
- **Product Pricing Strategy:** Wide range of average prices indicates a diverse product mix. Pricing strategies should consider category-specific factors such as competition, demand elasticity, and perceived value.
- **Bulk Purchase Opportunities:** High quantities ordered in categories like 'Office' and 'Apparel' suggest opportunities for promoting bulk purchases through volume discounts or special offers.

5B. Recommendations:

- **Standardize Delivery Charges:** Analyze the factors leading to high delivery charges in specific categories and work on standardizing these charges. Consider implementing flat-rate shipping or free shipping thresholds to simplify the customer experience.
- **Review Pricing Strategy:** Conduct market research to ensure that pricing aligns with customer expectations and competitor offerings. Adjust pricing for categories with high price variability to improve competitiveness and sales.
- **Promote Bulk Purchases:** Introduce promotions and discounts for bulk purchases in categories with high order quantities. This can help increase sales volumes and improve inventory turnover.
- **Address Outliers:** Investigate the reasons behind significant outliers in delivery charges and quantities. Address any operational inefficiencies or unusual order patterns to streamline processes and reduce anomalies.
- **Customer Segmentation:** Use the insights on average prices and quantities to segment customers and tailor marketing strategies. High-value segments might be targeted with premium offerings, while price-sensitive segments could benefit from special discounts and offers.

In [18]:
```
# Box plot for Tenure_Months with custom color and rotated x-axis labels
fig = px.box(customers_data,
             x = 'Location',
             y = 'Tenure_Months',
             color_discrete_sequence = ['#636EFA'])

fig.update_layout(title = 'Box plot for Tenure_Months by Location',
                  xaxis_title = 'Location',
                  yaxis_title = 'Tenure_Months')

fig.update_xaxes(tickangle = 45)
fig.show()
```



5A. Observations:

- **Tenure Distribution:** The median tenure (50th percentile) for customers is around 25 to 30 months across all locations. The interquartile range (IQR), which represents the middle 50% of the data, varies slightly between locations but generally falls between 15 to 40 months.
- **Variation Across Locations:** Chicago and New York have slightly higher median tenures compared to California, New Jersey, and Washington DC. The spread of tenure is quite similar across all locations, with each location having a range from approximately 0 to 50 months.
- **Outliers and Extremes:** There are no significant outliers, as the whiskers extend to the minimum and maximum values without any points outside this range. All locations show a similar maximum tenure close to 50 months and minimum tenure near 0 months.

5B. Insights:

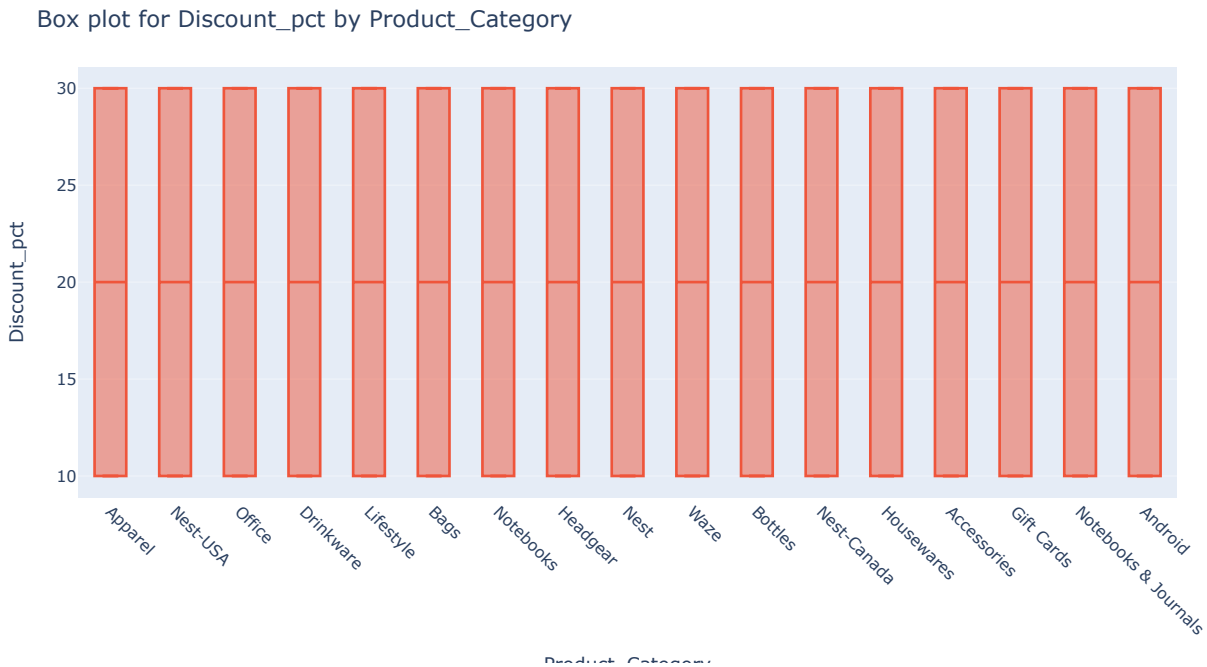
- **Consistent Customer Retention:** The consistency in median and IQR across different locations suggests that the company has a similar customer retention rate in different geographic areas.
- **Potential for Improvement:** Given the similarity in tenure distribution across locations, targeted efforts to increase tenure might be applied uniformly rather than requiring location-specific strategies.

- **Customer Loyalty:** The data indicates a moderate level of customer loyalty with a substantial proportion of customers remaining with the company for over 25 months.

5B. Recommendations:

- **Retention Programs:** Implement retention programs that reward long-term customers. Loyalty programs, special discounts, or exclusive offers for customers with higher tenure can help boost retention rates further.
- **Customer Experience Enhancement:** Enhance the overall customer experience by improving product quality, customer service, and engagement activities to encourage customers to remain with the company longer.
- **Analyze Customer Feedback:** Collect and analyze customer feedback across different locations to identify any specific issues that may be impacting customer tenure. Addressing these issues can help improve retention uniformly.
- **Personalized Marketing:** Use customer tenure data to create personalized marketing campaigns. Long-tenured customers can be targeted with premium offers, while newer customers can be engaged with introductory promotions to encourage loyalty.
- **Regular Engagement:** Maintain regular engagement with customers through newsletters, product updates, and personalized communications. Keeping customers informed and engaged can lead to higher retention.
- **Monitor and Adjust:** Continuously monitor customer tenure data and adjust strategies as needed. Use data analytics to identify trends and patterns

In [19]:
```
# Box plot for Discount_pct with custom color and rotated x-axis labels
fig = px.box(discount_coupon,
             x = 'Product_Category',
             y = 'Discount_pct',
             color_discrete_sequence = ['#EF553B'])

fig.update_layout(title = 'Box plot for Discount_pct by Product_Category',
                  xaxis_title = 'Product_Category',
                  yaxis_title = 'Discount_pct')

fig.update_xaxes(tickangle = 45)
fig.show()
```



5A. Observations:

- **Consistency in Discount Percentages:** All product categories have very similar discount percentages, centered around 20%. The spread of the discount percentages is narrow, indicating a consistent discount strategy across all product categories.
- **Lack of Outliers:** There are no visible outliers in the data, suggesting that the discount percentages are uniformly applied within the range of 10% to 30%.

5B. Insights:

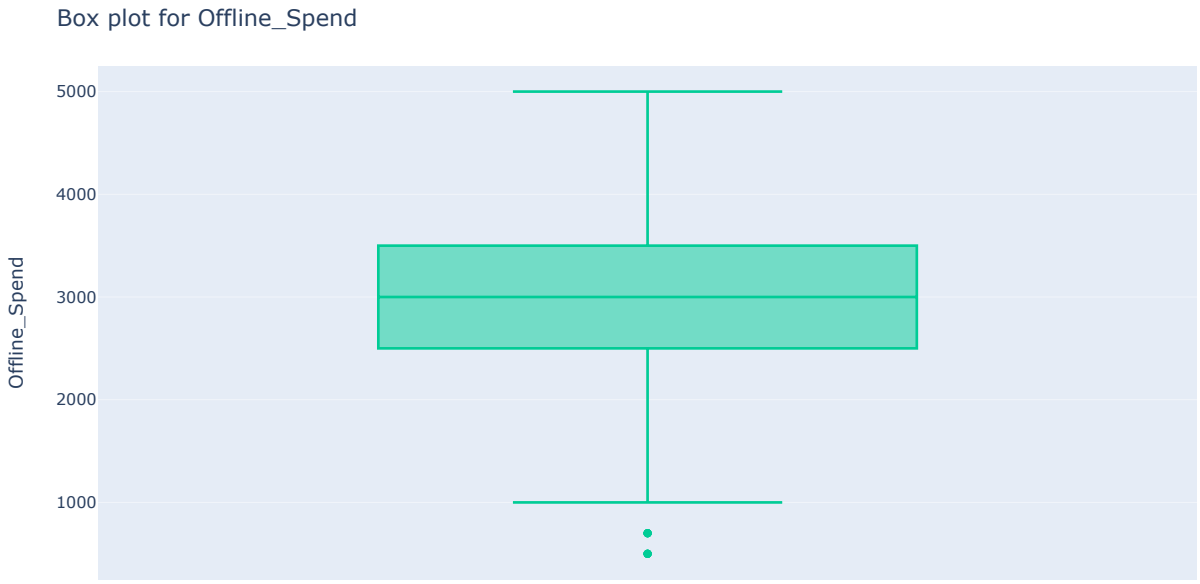
- **Uniform Discount Strategy:** The ecommerce company appears to apply a uniform discount strategy across all product categories. This could indicate a policy to simplify discounting for easier management and customer understanding.
- **Potential for Differentiation:** Given the uniformity, there might be an opportunity to differentiate discounts based on product category performance, customer demand, or seasonality to optimize sales and margins.
- **Customer Perception:** Consistent discounting may positively influence customer perception of fairness and predictability in pricing, potentially improving customer loyalty.

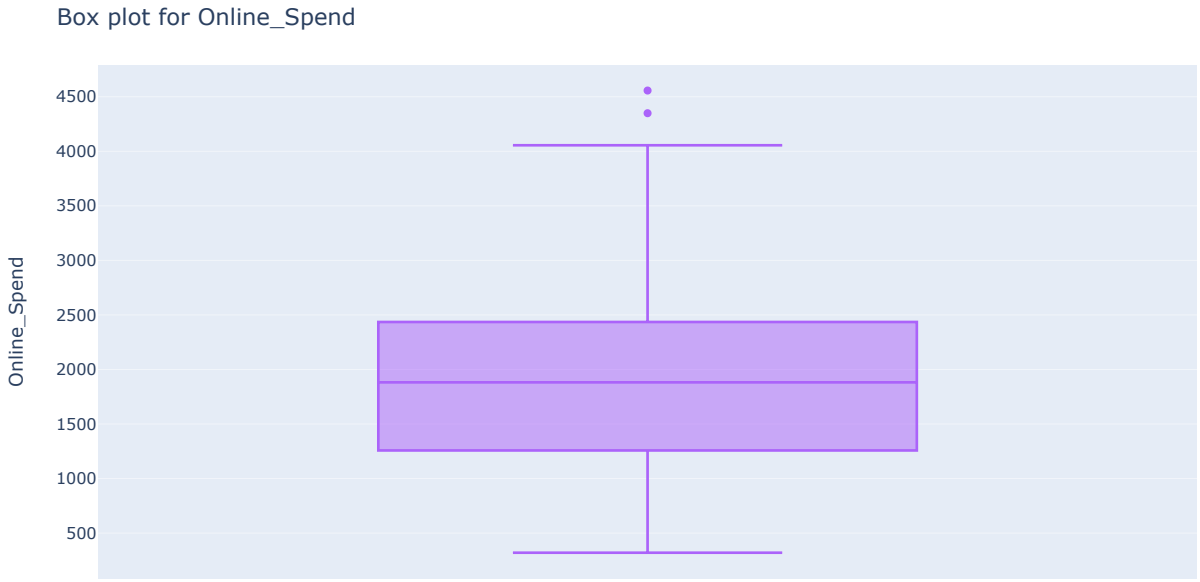
5B. Recommendations:

- **Analyze Sales Performance by Category:** Perform a detailed analysis of sales performance for each product category to determine if the uniform discount strategy is optimal. Adjust discounts where necessary to maximize revenue and profit.
- **Implement Targeted Promotions:** Consider implementing targeted promotions for specific categories, especially those with higher margins or those needing a sales boost, to drive more sales and better inventory turnover.

- **Seasonal and Demand-based Discounts:** Introduce seasonal or demand-based discounts to capitalize on peak shopping periods and high-demand products, which can drive higher sales volume and improve customer satisfaction.
- **Customer Segmentation:** Utilize customer demographics and purchase history to offer personalized discounts, enhancing customer experience and increasing repeat purchases.
- **A/B Testing:** Conduct A/B testing on different discount strategies to identify the most effective discount rates and promotions for various categories. Use the insights gained to refine the overall discounting strategy.
- **Monitor Competitor Pricing:** Keep track of competitors' pricing and discount strategies to ensure the company remains competitive. Adjust discount rates as needed to attract and retain customers.

In [20]:
```
# Box plot for Offline_Spend with custom color and rotated x-axis labels
fig = px.box(marketing_spend,
             y = 'Offline_Spend',
             color_discrete_sequence = ['#00CC96'])

fig.update_layout(title = 'Box plot for Offline_Spend',
                  yaxis_title = 'Offline_Spend')

fig.show()

# Box plot for Online_Spend with custom color and rotated x-axis labels
fig = px.box(marketing_spend,
             y = 'Online_Spend',
             color_discrete_sequence = ['#AB63FA'])

fig.update_layout(title = 'Box plot for Online_Spend',
                  yaxis_title = 'Online_Spend')

fig.show()
```





5A. Observations:

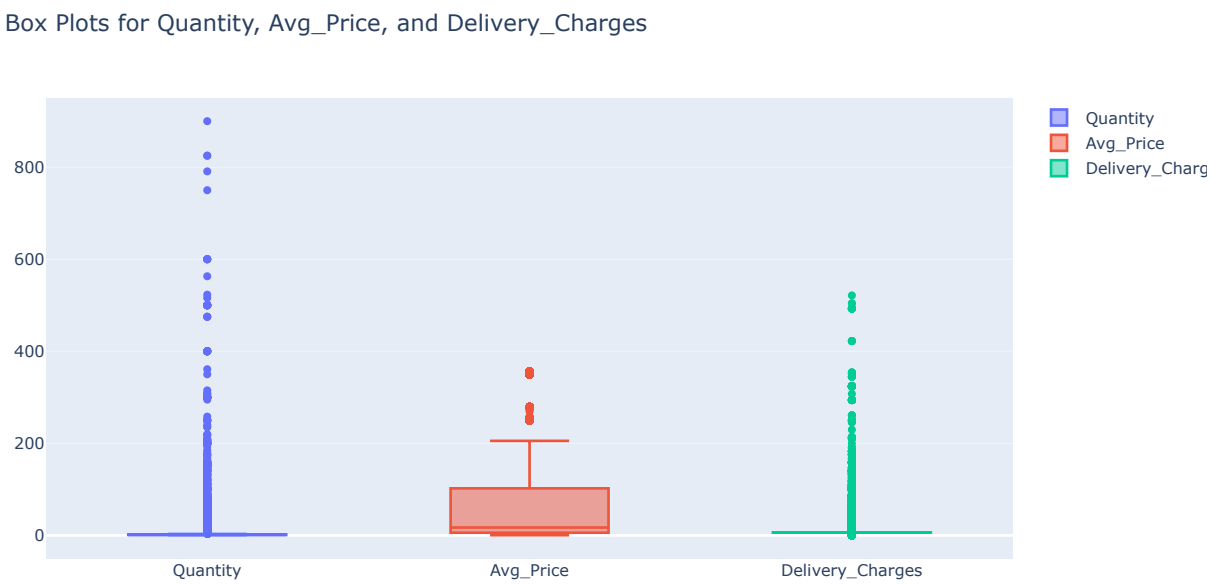
- **Online Spend:**
  - **Distribution and Range:**
    - The online spend varies from around 500 to just over 4000.
    - The median online spend is approximately 2000.

    - The interquartile range (IQR) spans from around 1250 to 2750, indicating that 50% of the data points lie within this range.
  - **Outliers:**
    - There are a few data points above 4000, indicating occasional higher spends on online marketing.
    - No significant lower outliers are present, suggesting a relatively consistent minimum spend.
- **Offline Spend:**
  - **Distribution and Range:
    - The offline spend ranges from around 1000 to 5000.
    - The median offline spend is slightly above 3000.
    - The IQR spans from approximately 2500 to 4000, meaning that 50% of the data points fall within this range.
  - **Outliers:**
    - There are a few lower outliers below 1000, suggesting occasional lower investment in offline marketing.
    - No significant higher outliers are present, indicating a consistent upper spend limit.

5B and 5C. Insights and Recommendations:

- **Marketing Spend Balance:** The company seems to invest more consistently in offline marketing compared to online marketing. The higher median and upper range in offline spend might reflect a strategy focused on traditional media channels.
- **Optimizing Marketing Strategy:**
  - **Online Marketing:** There is potential to explore periods of higher online spend to analyze their impact on sales and customer acquisition. If higher online spends correlate with increased revenue or customer engagement, consider reallocating budget towards online marketing during strategic periods.
  - **Offline Marketing:** The consistent high spend in offline channels suggests this is a trusted and effective medium for the company. Evaluate the return on investment (ROI) for offline campaigns to ensure sustained or increased budget allocation is justified.
- **Strategic Investments:** Investigate the specific campaigns or time periods that resulted in outliers in both online and offline spends. Optimize the marketing budget by focusing on high-ROI campaigns, whether online or offline.
- **Seasonal and Campaign Analysis:** Conduct a seasonal analysis of both online and offline spends to identify trends or patterns. Align marketing spend with peak sales periods, ensuring that high-investment campaigns coincide with times of high customer engagement.

In [21]:
```
# Box plots for Quantity, Avg_Price, and Delivery_Charges
fig = go.Figure()

fig.add_trace(go.Box(y = online_sales['Quantity'],
                     name = 'Quantity'))

fig.add_trace(go.Box(y = online_sales['Avg_Price'],
                     name = 'Avg_Price'))

fig.add_trace(go.Box(y = online_sales['Delivery_Charges'],
                     name = 'Delivery_Charges'))

fig.update_layout(title = 'Box Plots for Quantity, Avg_Price, and Delivery_Charges')
fig.show()
```



5A. Observations:

1. Quantity:

- The quantity plot shows significant outliers, with some transactions having quantities as high as 800.
- Most transactions have quantities concentrated near the lower end, indicating that a large number of orders consist of few items.
- There is a high variability in the quantity ordered, suggesting that while most orders are small, a few are bulk purchases.

2. Avg_Price:

- The average price per item has a wide range with many outliers.
- Most of the data points are clustered near the lower end of the price range, indicating that the majority of products sold are relatively inexpensive.
- The median price appears to be around 20-30, with few products priced much higher.

3. Delivery_Charges:

- Delivery charges also show outliers, though not as extreme as quantity or average price.
- The majority of delivery charges are low, but there are some transactions with significantly higher charges.
- This might suggest that delivery charges vary greatly depending on the product category, customer location, or delivery service used.

5B. Insights:

1. Customer Behavior:

- The bulk of transactions involve purchasing a small number of items, which might indicate that customers frequently make smaller, more frequent purchases rather than large bulk buys.
- There are a few high-volume purchasers who could be either business clients or loyal customers making bulk orders.

2. Product Pricing:

- The presence of many low-priced items suggests that the company offers a wide range of affordable products.
- The high variability in average price per item indicates a diverse product range, catering to both budget-conscious and premium customers.

3. Delivery Strategy:

- The variation in delivery charges suggests that there might be room for optimization in the shipping strategy. It might be beneficial to analyze which factors contribute to higher delivery charges and see if there are opportunities to reduce costs.
- High delivery charges could potentially deter customers, especially if they are ordering low-priced items.

5C. Recommendations:

1. Targeted Marketing:

- Use segmentation to target frequent low-volume purchasers with personalized offers to encourage larger orders.
- Develop marketing strategies to retain and grow the high-volume purchaser segment, possibly through loyalty programs or bulk purchase discounts.

2. Pricing Strategy:

- Ensure that pricing strategies reflect the value proposition for different customer segments.
- Evaluate if there are opportunities to introduce mid-tier pricing products if there is a large gap between low-priced and high-priced items.

3. Delivery Optimization:

- Investigate the reasons behind high delivery charges and explore partnerships with more cost-effective logistics providers.
- Consider offering free or reduced shipping for orders above a certain threshold to incentivize higher order values.

4. Product Offering Analysis:

- Regularly review the product mix to ensure it meets the evolving needs and preferences of customers.
- Identify and promote high-margin products more effectively to boost overall profitability.

1B. Invoice Amount:

It is crucial to begin by calculating the invoice amount or revenue for each transaction using this formula, This establishes the foundation for revenue analysis.

Invoice Value = ((Quantity Avg_price) (1 - Discount_pct) * (1 + GST)) + Delivery_Charges.

In [22]:
```
online_sales['Transaction_Date'] = pd.to_datetime(online_sales['Transaction_Date'], format='%m/%d/%Y')

online_sales['Month'] = online_sales['Transaction_Date'].dt.strftime('%B')

# Merge online_sales with discount_coupon on Month and Product_Category
result = pd.merge(online_sales,
                  discount_coupon,
                  on = ['Month', 'Product_Category'],
                  how = 'left')

# Merge result with tax_amount on Product_Category
merged_df = pd.merge(result,
                     tax_amount,
                     on = ['Product_Category'],
                     how = 'left')

# Fill 'Not Available' in 'Coupon_Code' and 0 in 'Discount_pct' if no matching coupon_code or discount_pct found
merged_df['Coupon_Code'].fillna('Not Available', inplace = True)
merged_df['Discount_pct'].fillna(0, inplace = True)

# Convert columns to numeric where applicable
cols_to_numeric = ['Quantity', 'Avg_Price', 'Delivery_Charges', 'Discount_pct', 'GST']
merged_df[cols_to_numeric] = merged_df[cols_to_numeric].apply(pd.to_numeric, errors='coerce')

# Handle NaN values if any
merged_df.fillna(0, inplace = True)

# Calculate the invoice value
merged_df['Invoice'] = ((merged_df['Quantity'] * merged_df['Avg_Price']) *
                                        (1 - merged_df['Discount_pct'] / 100) *
                                        (1 + merged_df['GST'])) + merged_df['Delivery_Charges']

# Display the first few rows of the dataframe with the new 'Invoice_Value' column
print(merged_df[['CustomerID', 'Transaction_ID', 'Invoice']].head())
```

```
   CustomerID  Transaction_ID  Invoice
0       17850           16679   160.21
1       17850           16680   160.21
2       17850           16681     8.55
3       17850           16682    94.15
4       17850           16682    23.00
```

In [23]:
```
merged_df.head()
```

Out[23]:

	CustomerID	Transaction_ID	Transaction_Date	Product_SKU	Product_Description	Product_Category	Quantity	Avg_Price	Delivery_Charges	Coup
0	17850	16679	2019-01-01	GGOENEBJ079499	Nest Learning Thermostat 3rd Gen-USA - Stainle...	Nest-USA	1	153.71	6.5	
1	17850	16680	2019-01-01	GGOENEBJ079499	Nest Learning Thermostat 3rd Gen-USA - Stainle...	Nest-USA	1	153.71	6.5	
2	17850	16681	2019-01-01	GGOEGFKQ020399	Google Laptop and Cell Phone Stickers	Office	1	2.05	6.5	
3	17850	16682	2019-01-01	GGOEGAAB010516	Google Men's 100% Cotton Short Sleeve Hero Tee...	Apparel	5	17.53	6.5	
4	17850	16682	2019-01-01	GGOEGBJL013999	Google Canvas Tote Natural/Navy	Bags	1	16.50	6.5	

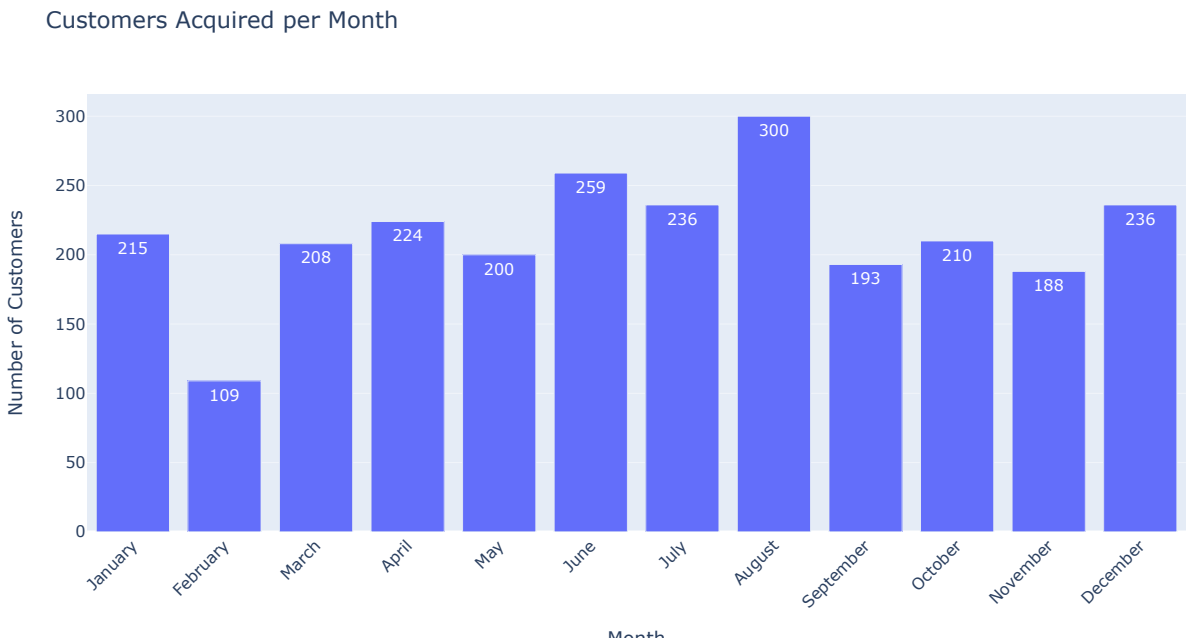
---

2. Exploratory Data Analysis (EDA):

2A. Customer Acquisition:

Analyzing trends in customer acquisition month over month:

In [24]:
```
# Create a new column for the transaction month
merged_df['Transaction_Month'] = merged_df['Transaction_Date'].dt.to_period('M')

# Calculate the number of unique customers acquired each month
monthly_cust = merged_df.groupby(['Transaction_Month'])['CustomerID'].nunique()

# Map the periods to month names
monthly_cust.index = monthly_cust.index.map(lambda x: x.strftime('%B'))

# Convert to DataFrame for Plotly
monthly_cust_df = monthly_cust.reset_index()
monthly_cust_df.columns = ['Month', 'Number of Customers']

# Plot the bar chart using Plotly
fig = px.bar(monthly_cust_df,
             x = 'Month',
             y = 'Number of Customers',
             title = 'Customers Acquired per Month',
             labels = {'Month': 'Month', 'Number of Customers': 'Number of Customers'},
             text = 'Number of Customers')

# Update layout for better presentation
fig.update_layout(xaxis_title = 'Month',
                  yaxis_title = 'Number of Customers',
                  xaxis = dict(tickangle=-45))

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



5A. Observations:

1. Monthly Customer Acquisition Trends:

- The number of customers acquired per month shows significant fluctuations.
- The highest number of customers were acquired in August (300), followed by June (259), and December (236).
- The lowest number of customers were acquired in February (109).

2. Seasonal Patterns:

- There appears to be a peak in customer acquisition during the summer months (June and August) and around the holiday season in December.
- There is a noticeable dip in customer acquisition in February and September.

5B. Insights:

1. Effectiveness of Marketing Campaigns:

- The peak in August could be a result of effective marketing campaigns, seasonal sales, or promotions during this period.
- The increase in December suggests successful holiday season promotions.

2. Customer Acquisition Challenges:

- The low numbers in February indicate potential challenges in customer acquisition during this month. This could be due to seasonal factors or less effective marketing strategies during this period.
- The drop in September might indicate a post-summer lull or less impactful marketing efforts.

3. Promotion and Discount Effectiveness:

- It would be useful to correlate the number of customers acquired with the marketing spend and discount coupons applied during these months to determine their effectiveness.

5C. Recommendations:

1. Enhanced Marketing During Low Months:

- Increase marketing efforts and promotions during February and September to boost customer acquisition.
- Consider special campaigns or incentives to attract more customers during these slower months.

2. Leverage Peak Seasons:

- Continue to capitalize on the peak acquisition periods (June, August, and December) by planning significant marketing campaigns and discounts during these months.
- Ensure that inventory levels and customer support are adequately prepared to handle increased demand during these periods.

3. Analyze Marketing Spend Efficiency:

- Conduct a detailed analysis to understand the correlation between marketing spend (both online and offline) and customer acquisition.
- Optimize marketing budgets by focusing more on channels and strategies that yield the highest customer acquisition rates.

4. Utilize Customer Demographics:

- Use customer demographic data to tailor marketing campaigns more effectively. For example, targeting specific locations or demographics that show higher engagement and acquisition rates.

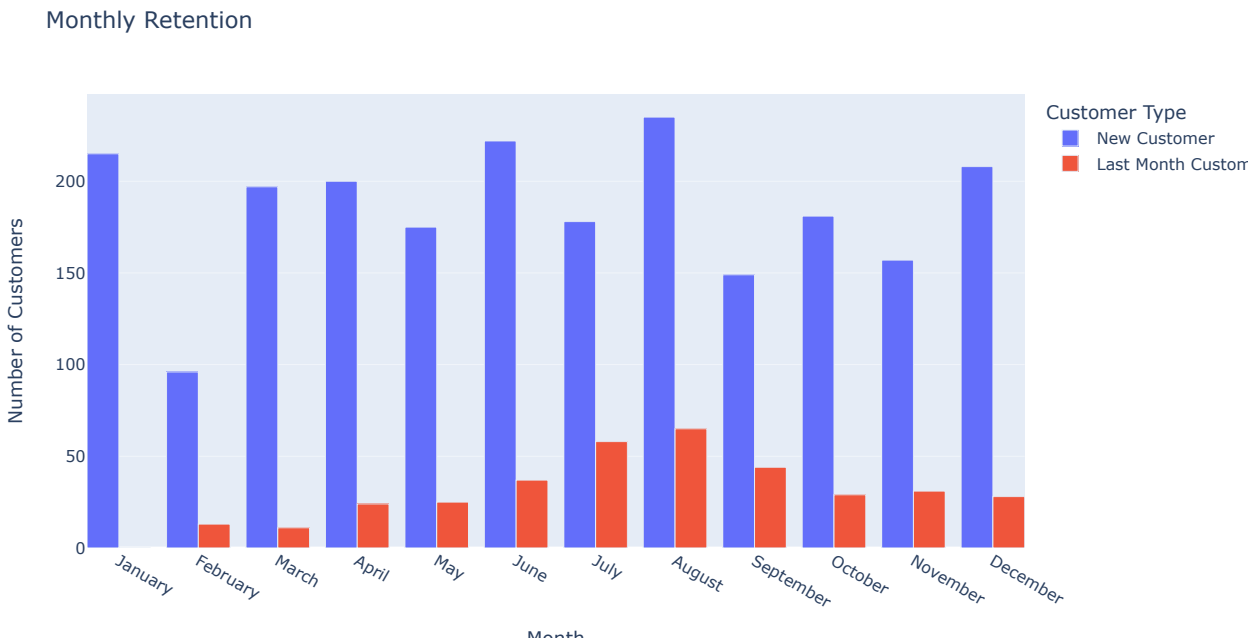
5. Promotion Strategies:

- Review the effectiveness of discount coupons and their impact on customer acquisition. Ensure that the discounts are attractive enough to incentivize purchases but also sustainable for the business.
- Experiment with different types of promotions (e.g., bundle deals, limited-time offers) to see which ones drive the most customer acquisition.

2A. Customer Retention:

Analyzing trends in customer retention month over month:

In [25]:
```
merged_df['Transaction_Month'] = merged_df['Transaction_Date'].dt.month

non_ret_customer = {}
ret_customer = {}
temp = set()  # Use a set to track unique customer IDs
new_customer_id = {}
exist_customer_id = {}
new_customer = {}
exist_customer = {}

for i in range(1, 13):
    current_month = set(merged_df[merged_df['Transaction_Month'] == i]['CustomerID'].unique())
    prev_month = set(merged_df[merged_df['Transaction_Month'] == i-1]['CustomerID'].unique()) if i > 1 else set()

    non_ret_cust = current_month - prev_month
    non_ret_customer[i] = len(non_ret_cust)

    ret_cust = current_month & prev_month
    ret_customer[i] = len(ret_cust)

    new_cust = current_month - temp
    exist_cust = current_month & temp
    temp.update(current_month)

    new_customer_id[i] = list(new_cust)
    exist_customer_id[i] = list(exist_cust)
    new_customer[i] = len(new_cust)
    exist_customer[i] = len(exist_cust)

customer_df = pd.DataFrame({
    'Month': list(non_ret_customer.keys()),
    'Non_Ret_Cust': list(non_ret_customer.values()),
    'Ret_Cust': list(ret_customer.values()),
    'New_Cust': list(new_customer.values()),
    'Exist_Cust': list(exist_customer.values())
})

# Plotting with Plotly
x = range(1, 13)
width = 0.4

fig = go.Figure()

# Add bar for non-retained customers
fig.add_trace(go.Bar(x = [pos - width/2 for pos in x],
                     y = customer_df['Non_Ret_Cust'],
                     width = width,
                     name = 'New Customer'))

# Add bar for retained customers
fig.add_trace(go.Bar(x = [pos + width/2 for pos in x],
                     y = customer_df['Ret_Cust'],
                     width = width,
                     name = 'Last Month Customer'))

# Update layout
fig.update_layout(title = 'Monthly Retention',
                  xaxis_title = 'Month',
                  yaxis_title = 'Number of Customers',
                  xaxis = dict(tickmode = 'array',
                               tickvals = list(x),
                               ticktext = [calendar.month_name[i] for i in x]),
                  legend_title_text = 'Customer Type',
                  barmode = 'group')

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



5A. Observations:

1. New vs. Last Month Customer Trends:

- The number of new customers acquired each month generally exceeds the number of returning customers from the previous month.
- There is a noticeable dip in the number of returning customers in February, April, and September.

2. Monthly Retention:

- Retention rates appear to be lower overall, as indicated by the smaller number of returning customers (in red) compared to new customers (in blue).

3. High Retention Periods:

- Retention is relatively higher in the months of June, July, and August compared to other months.

5B. Insights:

1. Customer Acquisition and Retention Imbalance:

- The company is successful in acquiring new customers each month, but struggles to retain them.
- This imbalance suggests that while initial marketing efforts are effective, there may be issues with customer satisfaction, product quality, or post-purchase engagement that need to be addressed.

2. Seasonal Retention Patterns:

- The higher retention rates in the summer months (June, July, August) could be due to seasonal products, effective marketing campaigns, or customer loyalty programs during this period.
- Lower retention rates in February, April, and September could indicate potential dissatisfaction or lack of engagement during these months.

5C. Recommendations:

1. Improving Customer Retention:

- Implement customer loyalty programs to incentivize repeat purchases. This could include rewards points, exclusive discounts for returning customers, and personalized offers.
- Improve post-purchase engagement through follow-up emails, feedback requests, and personalized recommendations to keep customers engaged and satisfied.

2. Analyze Customer Feedback:

- Collect and analyze customer feedback to identify common pain points and areas for improvement. Addressing these issues promptly can help improve overall customer satisfaction and retention.
- Conduct surveys or use customer reviews to gain insights into why customers might not be returning.

3. Enhance Product Quality and Service:

- Ensure that product quality and delivery services meet customer expectations. Any issues in these areas can lead to customer dissatisfaction and lower retention rates.
- Offer excellent customer service to handle any complaints or issues promptly and effectively.

4. Seasonal Campaigns:

- During low retention months (February, April, September), run targeted campaigns to re-engage previous customers. This could include special offers, reminders of past purchases, or introducing new products.
- Leverage the high retention periods by reinforcing marketing efforts and ensuring that the customer experience during these times encourages future retention.

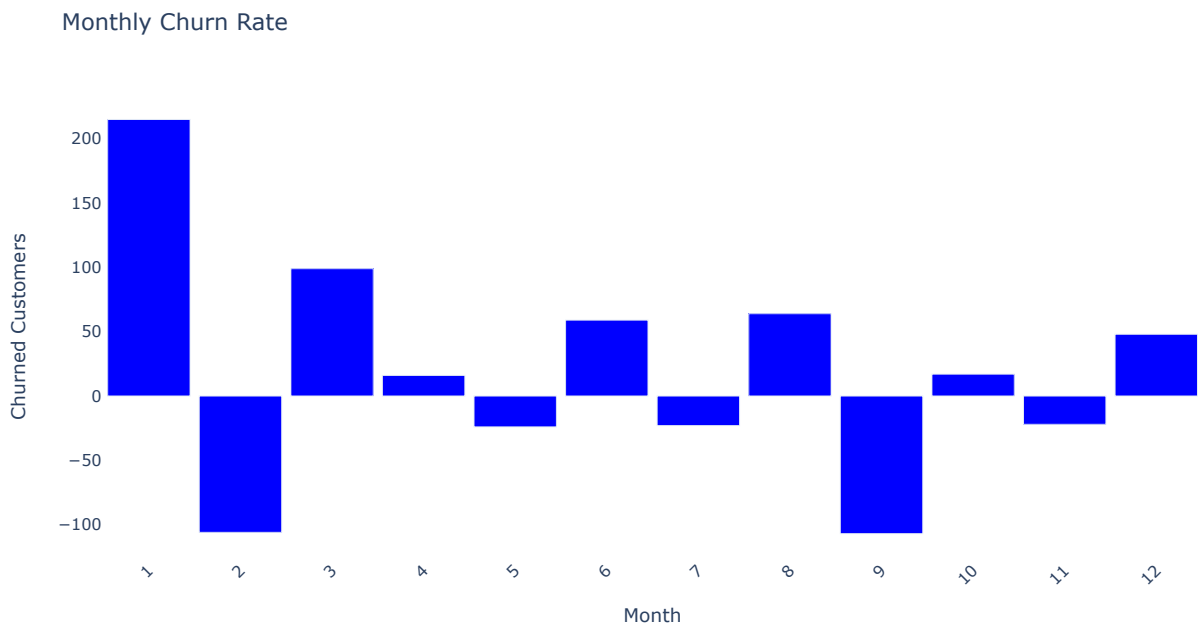
5. Customer Segmentation:

- Segment customers based on their purchasing behavior, demographics, and preferences to create targeted marketing strategies. Personalizing the customer experience can lead to higher retention rates.
- Use data analytics to identify which segments are more likely to return and focus on enhancing their experience.

2A. Customer Churn:

Calculating churn across different customer demographics (gender, location, tenure) and timeframes (monthly).

In [26]:
```
# Calculate monthly churn
monthly_active_customers = merged_df.groupby(['Transaction_Month'])['CustomerID'].nunique()
monthly_churn = monthly_active_customers.diff().fillna(monthly_active_customers.iloc[0])

# Create Plotly bar chart
fig = go.Figure()

fig.add_trace(go.Bar(x = monthly_churn.index.astype(str),
                     y = monthly_churn.values,
                     marker_color = 'blue'))

# Update layout
fig.update_layout(title = 'Monthly Churn Rate',
                  xaxis_title = 'Month',
                  yaxis_title = 'Churned Customers',
                  xaxis_tickangle = -45,
                  bargap = 0.1,
                  plot_bgcolor = 'rgba(0,0,0,0)')

# Show plot
fig.show()
```



5A. Observations:

1. Monthly Churn Trends:

- January has the highest churn rate with over 200 customers lost.
- February has a negative churn rate, indicating a net gain in customers.
- Significant churn is observed in March, June, and September.
- Other months show relatively lower and more stable churn rates.

2. Churn Patterns:

- High churn in January might suggest post-holiday season attrition where customers do not return after the holiday shopping.
- The net gain in February indicates successful customer retention strategies or acquisition efforts that outweigh the churn.

5B. Insights:

1. Seasonal Effects on Churn:

- The high churn in January is likely a result of post-holiday drop-off, where customers who made purchases in December do not continue into the new year.
- The spike in churn during March, June, and September could be related to specific events, promotions, or changes in customer behavior.

2. Retention Success in February:

- The negative churn in February suggests effective retention strategies or successful marketing campaigns that not only retained existing customers but also attracted new ones.

5C. Recommendations:

1. Address Post-Holiday Churn:

- Implement targeted retention strategies in January to mitigate the high churn rate. This could include follow-up marketing campaigns, special offers, or loyalty programs to encourage repeat purchases.
- Analyze the reasons for the high churn in January by collecting customer feedback and identifying common factors.

2. Improve Retention Strategies:

- Focus on the months with higher churn rates (March, June, September) by enhancing customer engagement and satisfaction. Personalized communication, targeted promotions, and quality customer service can help reduce churn.
- Develop a customer retention plan that includes incentives for repeat purchases and loyalty rewards.

3. Leverage Successful Strategies:

- Identify and replicate the successful strategies used in February that resulted in a net gain in customers. This could include specific marketing campaigns, promotions, or engagement tactics.
- Apply these strategies in other months to achieve more consistent retention and reduce churn.

4. Monitor Customer Behavior:

- Continuously monitor customer behavior and feedback to identify potential issues early and address them promptly. This proactive approach can help reduce churn rates.
- Use data analytics to segment customers based on their behavior and tailor retention strategies accordingly.

5. Enhance Customer Experience:

- Focus on providing an exceptional customer experience throughout the year. Ensure that product quality, delivery, and customer service meet or exceed customer expectations.
- Offer personalized recommendations and promotions based on customer preferences and past purchases to enhance their shopping experience.

---

2. Exploratory Data Analysis (EDA):

2B. Marketing Campaign Impact:

1. Marketing Spends:

In [27]:

```
# Convert 'Date' column to datetime if necessary
merged_df['Transaction_Date'] = pd.to_datetime(merged_df['Transaction_Date'])
marketing_spend['Date'] = pd.to_datetime(marketing_spend['Date'])

# Aggregate data
sales_data = merged_df.groupby(['Transaction_Date'])['Invoice'].sum().reset_index()
offline_spend_data = marketing_spend.set_index('Date')['Offline_Spend']
online_spend_data = marketing_spend.set_index('Date')['Online_Spend']

# Create a subplot figure
fig = make_subplots(rows = 3,
                    cols = 1,
                    shared_xaxes = True,
                    vertical_spacing = 0.1)

# Add sales plot
fig.add_trace(go.Scatter(x = sales_data['Transaction_Date'],
                         y = sales_data['Invoice'],
                         name = 'Sales'),
              row = 1,
              col = 1)

# Add offline spend plot
fig.add_trace(
    go.Scatter(x=offline_spend_data.index, y=offline_spend_data, name='Offline Spend', line=dict(color='orange')),
    row=2, col=1
)

# Add online spend plot
fig.add_trace(go.Scatter(x = online_spend_data.index,
                         y = online_spend_data,
                         name = 'Online Spend',
                         line = dict(color = 'green')),
              row = 3,
              col = 1)

# Update layout
fig.update_layout(height = 900,
                  width = 1600,
                  title_text = 'Sales and Marketing Spend Over Time',
                  xaxis_title = 'Date',
                  showlegend = False)

# Update y-axes titles
fig.update_yaxes(title_text = "Sales",
                 row = 1,
                 col = 1)
fig.update_yaxes(title_text = "Offline Spend",
                 row = 2,
                 col = 1)
fig.update_yaxes(title_text = "Online Spend",
                 row = 3,
                 col = 1)

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

5A. Observations:

1. Sales Trends (Top Plot - Blue Line)

- There is significant variability in daily sales, with occasional sharp spikes.
- Sales appear to have a gradual upward trend over the year, with noticeable peaks around certain times.
- The largest sales spikes occur sporadically, which may be tied to specific events or promotions.

2. Offline Marketing Spend (Middle Plot - Orange Line)

- Offline marketing spend shows a step-like pattern, with periods of constant spend followed by abrupt changes.
- There are several distinct periods where offline spend was either increased or decreased, suggesting a strategic adjustment of budget allocation.
- The offline marketing spend is relatively higher during the initial months and becomes more varied throughout the year.

3. Online Marketing Spend (Bottom Plot - Green Line)

- Online marketing spend shows a high frequency of fluctuations, with daily variations.
- There are frequent peaks in online spend, indicating a dynamic approach to online marketing campaigns.
- The overall online spend shows more consistent peaks throughout the year compared to offline spend.

5B. Insights:

1. Correlation between Marketing Spend and Sales

- There appears to be a correlation between peaks in online spend and sales spikes, suggesting that online marketing efforts may have a direct impact on sales performance.
- The step changes in offline spend might be contributing to maintaining baseline sales levels but don't seem to correspond as directly to the spikes seen in sales.

2. Seasonality and Promotions

- Sales spikes may align with specific promotions, holidays, or seasonal campaigns. Identifying the exact dates of these spikes could help correlate with specific marketing or promotional activities.

3. Marketing Spend Efficiency

- The high variability in online spend suggests a responsive strategy to market conditions or campaign performance. This could indicate a higher efficiency and adaptability in online marketing.
- Offline spend, with its less frequent adjustments, may reflect longer-term planning or less immediate feedback on performance.

5C. Recommendations:

1. Enhanced Attribution Analysis

- Conduct a detailed attribution analysis to better understand the direct impact of online and offline marketing spend on sales. This can help in optimizing budget allocation.

2. Promotion and Campaign Alignment

- Align major online and offline marketing campaigns with identified high-sales periods to maximize their impact. Use historical data to forecast and prepare for these peaks.

3. Optimize Online Marketing Spend

- Given the high variability and apparent effectiveness of online marketing, further investment in real-time analytics and automated bid adjustments could improve return on investment (ROI).

4. Refine Offline Marketing Strategy

- Evaluate the periods of high and low offline spend for their effectiveness in driving sales. Consider more frequent adjustments to offline spend to better match market conditions and sales patterns.

5. Customer Segmentation and Targeting

- Utilize customer demographic data to segment and target marketing efforts more effectively. Tailoring campaigns based on gender, location, and tenure can lead to more personalized and impactful marketing strategies.

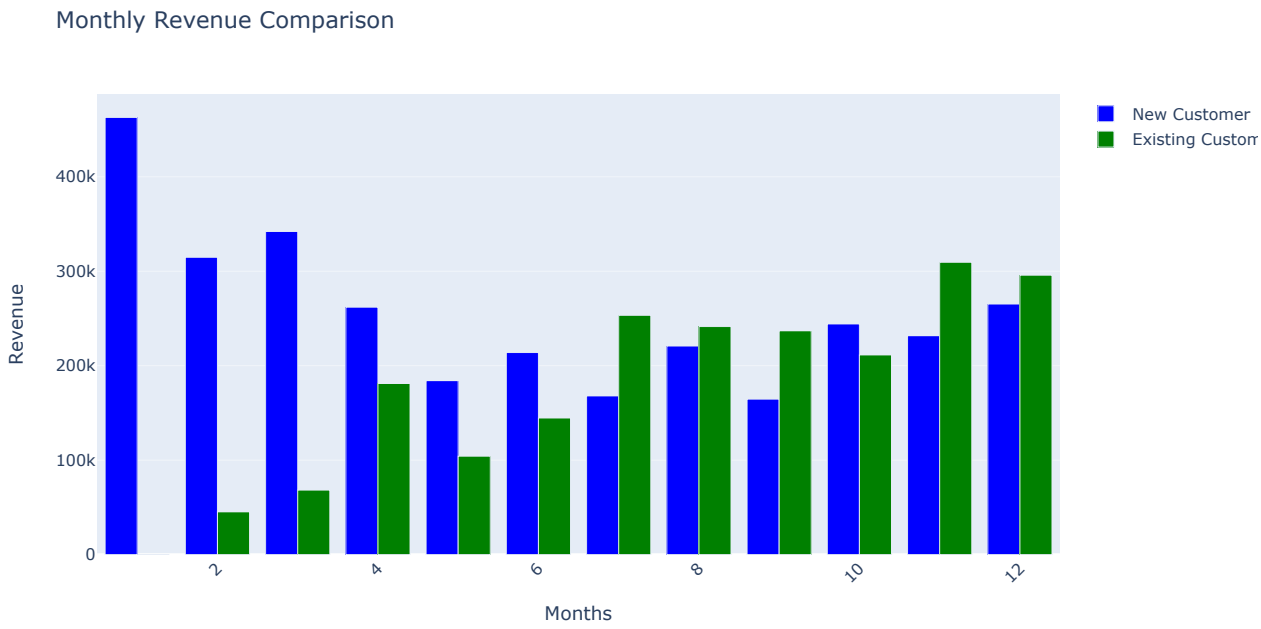
6. Cross-channel Marketing Integration

- Develop integrated marketing campaigns that leverage both online and offline channels to create a cohesive customer journey. Consistency across channels can amplify the overall marketing impact.

2. Exploring the relationship between marketing spend (online & offline) and customer behavior (orders, revenue) to assess campaign effectiveness:

In [28]:

```
temp = []
new_cust_each_month = {}
existing_cust_each_month = {}

no_of_new_cust_each_month = {}
no_of_existing_cust_each_month = {}

# Calculate new and existing customers for each month
for i in merged_df['Transaction_Month'].unique():
    x = merged_df[merged_df['Transaction_Month'] == i]['CustomerID'].unique().tolist()
    new_cust = [value for value in x if value not in temp]
    existing_cust = [value for value in x if value in temp]
    temp.extend(x)
    temp = list(set(temp))
    new_cust_each_month[i] = new_cust
    existing_cust_each_month[i] = existing_cust
    no_of_new_cust_each_month[i] = len(new_cust)
    no_of_existing_cust_each_month[i] = len(existing_cust)

# Calculate revenue from new and existing customers for each month
new_cust_each_month_revenue = {}
existing_cust_each_month_revenue = {}

for month, ids in new_cust_each_month.items():
    new_cust_each_month_revenue[month] = merged_df[(merged_df['Transaction_Month'] == month) &
                                                   (merged_df['CustomerID'].isin(ids))]['Invoice'].sum()

for month, ids in existing_cust_each_month.items():
    existing_cust_each_month_revenue[month] = merged_df[(merged_df['Transaction_Month'] == month) &
                                                        (merged_df['CustomerID'].isin(ids))]['Invoice'].sum()

# Prepare data for plotting
months = list(existing_cust_each_month_revenue.keys())
new_cust = list(new_cust_each_month_revenue.values())
existing_cust = list(existing_cust_each_month_revenue.values())

# Create a Plotly bar chart
fig = go.Figure()

# Add bars for new customer revenue
fig.add_trace(go.Bar(x = months,
                     y = new_cust,
                     name = 'New Customer',
                     marker_color = 'blue'))

# Add bars for existing customer revenue
fig.add_trace(go.Bar(x = months,
                     y = existing_cust,
                     name = 'Existing Customer',
                     marker_color = 'green'))

# Update layout for better presentation
fig.update_layout(title = 'Monthly Revenue Comparison',
                  xaxis_title = 'Months',
                  yaxis_title = 'Revenue',
                  barmode = 'group',
                  xaxis_tickangle = -45)

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



5A. Observations:

1. Revenue from New Customers (Blue Bars)

- January has the highest revenue from new customers, exceeding 400k.
- There is a sharp decline in revenue from new customers after January, with a gradual decrease until a slight rise in June.
- In the later months, from August to December, the revenue from new customers shows some recovery but remains lower than the early months.

2. Revenue from Existing Customers (Green Bars)

- Revenue from existing customers is much lower in the initial months (January to April).
- From May onwards, there is a noticeable increase in revenue from existing customers, surpassing new customers' revenue in several months (e.g., September, October, and December).
- December shows a strong finish for both new and existing customers, with existing customer revenue slightly higher.

3. Monthly Comparison

- The first quarter (January to March) is dominated by revenue from new customers, while existing customer revenue starts to pick up from the second quarter onwards.
- The second half of the year shows a more balanced revenue contribution from both new and existing customers, with existing customers contributing significantly towards the end of the year.

5B. Insights:

1. Customer Acquisition vs. Retention

- The high revenue from new customers in January suggests effective customer acquisition strategies or promotions at the start of the year.
- The increasing trend in revenue from existing customers indicates successful customer retention efforts and possibly the effect of loyalty programs or repeat purchases.

2. Seasonality and Promotions

- The sharp decline in new customer revenue after January suggests that the initial promotions or campaigns were very effective but not sustained throughout the year.
- The steady growth in existing customer revenue in the latter half of the year might be due to targeted retention strategies, holiday season promotions, or improved customer experience leading to repeat purchases.

3. Strategic Shifts

The data indicates a possible strategic shift from focusing on acquiring new customers to retaining existing ones as the year progresses. This balance is crucial for sustainable growth, as acquiring new customers generally costs more than retaining existing ones.

5C. Recommendations:

1. Sustain Customer Acquisition Efforts

- To avoid the sharp decline in new customer revenue after January, implement sustained customer acquisition efforts throughout the year. This can include periodic promotions, referral programs, and targeted advertising campaigns.

2. Enhance Customer Retention Strategies

- Continue to focus on retaining existing customers by offering loyalty programs, personalized offers, and excellent customer service. The data shows these efforts are paying off in the latter half of the year.

3. Analyze Promotional Effectiveness

- Conduct a detailed analysis of the promotions and campaigns run in January to understand what drove the high revenue from new customers. Apply these learnings to future campaigns.

4. Balance Marketing Spend

- Allocate marketing spend strategically to balance between acquiring new customers and retaining existing ones. This can be informed by the revenue patterns observed in the data.

5. Leverage Peak Seasons

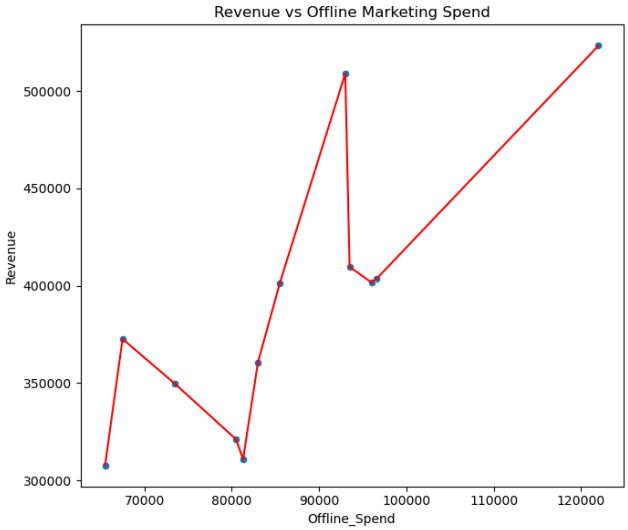
- Utilize the insights from peak revenue periods to plan marketing and promotional activities around key seasons. For example, focus on ramping up efforts before the holiday season when existing customer revenue tends to increase.

6. Customer Feedback and Improvement

- Collect and analyze customer feedback to continuously improve the shopping experience. This can lead to higher customer satisfaction, loyalty, and ultimately, increased revenue from existing customers.

In [29]:
```
# Load datasets
online_sales = pd.read_csv('Online_Sales.csv')
marketing_spend = pd.read_csv('Marketing_Spend.csv')

# Convert Transaction_Date and Date to datetime
online_sales['Transaction_Date'] = pd.to_datetime(online_sales['Transaction_Date'])
marketing_spend['Date'] = pd.to_datetime(marketing_spend['Date'])

# Calculate Revenue if not already present
online_sales['Revenue'] = online_sales['Quantity'] * online_sales['Avg_Price']

# Extract Year-Month from Transaction_Date and Date
online_sales['YearMonth'] = online_sales['Transaction_Date'].dt.to_period('M')
marketing_spend['YearMonth'] = marketing_spend['Date'].dt.to_period('M')

# Aggregate customer behavior metrics by month
monthly_metrics = online_sales.groupby('YearMonth').agg({
    'Transaction_ID': 'count',  # Number of orders
    'Revenue': 'sum'  # Total revenue
}).reset_index()

# Aggregate marketing spend by month
monthly_spend = marketing_spend.groupby('YearMonth').agg({
    'Offline_Spend': 'sum',
    'Online_Spend': 'sum'
}).reset_index()

# Merge the aggregated dataframes
merged_data = pd.merge(monthly_metrics,
                       monthly_spend,
                       on = 'YearMonth')

# Rename columns for clarity
merged_data.columns = ['Month', 'Orders', 'Revenue', 'Offline_Spend', 'Online_Spend']

# Correlation analysis
correlation_matrix = merged_data.corr()
print(correlation_matrix)

# Visualize relationships
plt.figure(figsize = (14, 6))

# Plot Revenue vs Marketing Spend
plt.subplot(1, 2, 1)
sns.scatterplot(data = merged_data,
                x = 'Offline_Spend',
                y = 'Revenue')
sns.lineplot(data = merged_data,
             x = 'Offline_Spend',
             y = 'Revenue',
             color = 'red')
plt.title('Revenue vs Offline Marketing Spend')

plt.subplot(1, 2, 2)
sns.scatterplot(data = merged_data,
                x = 'Online_Spend',
                y = 'Revenue')
sns.lineplot(data = merged_data,
             x = 'Online_Spend',
             y = 'Revenue',
             color = 'red')
plt.title('Revenue vs Online Marketing Spend')

plt.tight_layout()
plt.show()
```

```
                Orders   Revenue  Offline_Spend  Online_Spend
Orders        1.000000  0.088470      -0.167176     -0.089020
Revenue       0.088470  1.000000       0.798801      0.887384
Offline_Spend -0.167176  0.798801       1.000000      0.879841
Online_Spend  -0.089020  0.887384       0.879841      1.000000
```



5A. Observations:

1. Revenue vs Offline Marketing Spend:

- The plot shows a general upward trend in revenue with increasing offline marketing spend.
- There are fluctuations, particularly when offline spend is between 70,000 and 90,000, where revenue dips before increasing again.
- The highest revenue is achieved at the highest offline spend of approximately 120,000.

2. Revenue vs Online Marketing Spend:

- The plot indicates a more erratic pattern compared to offline spend.
- Despite fluctuations, there is an overall upward trend in revenue with increasing online marketing spend.
- The highest revenue is achieved at the highest online spend of approximately 75,000.

5B. Insights:

1. Offline Marketing Spend Effectiveness:

- Offline marketing spend shows a clearer and more consistent positive impact on revenue compared to online spend.
- The significant revenue increase at higher offline spend levels suggests that traditional marketing channels like TV, Radio, and Newspapers are effective for this ecommerce company.

2. Online Marketing Spend Variability:

- Online marketing spend results in a more variable revenue pattern, indicating that online marketing efforts may be more unpredictable.
- Despite the variability, there is a positive correlation between online marketing spend and revenue, especially at higher spend levels.

3. Optimal Marketing Spend:

- Both offline and online marketing spends contribute to revenue growth, but offline marketing appears to provide a more predictable return on investment.
- Combining both marketing strategies can help balance the unpredictability of online marketing with the steadier returns from offline marketing.

5C. Recommendations:

1. Increase Offline Marketing Spend:

- Given the clear positive correlation between offline marketing spend and revenue, the company should consider allocating more budget to offline marketing channels.
- Continuously monitor the performance to identify if there is a plateau where additional spend does not translate into proportionate revenue increases.

2. Optimize Online Marketing Strategies:

- Focus on optimizing online marketing strategies to reduce variability. This could include better targeting, using data analytics to refine campaigns, and investing in high-conversion channels.
- Consider testing different online marketing approaches (e.g., A/B testing) to identify the most effective strategies.

3. Balanced Marketing Mix:

- Maintain a balanced marketing mix, leveraging the strengths of both offline and online marketing channels.
- Allocate budgets dynamically based on performance metrics and seasonal trends to maximize overall revenue.

4. Regular Performance Analysis:

- Conduct regular analysis of marketing spend effectiveness to ensure that funds are being utilized efficiently.
- Use performance data to make informed decisions on adjusting marketing strategies and spend allocation.

5. Customer Segmentation:

- Segment customers based on demographics, purchase behavior, and response to marketing channels. Tailor marketing strategies to target each segment effectively.
- Personalized marketing can lead to higher conversion rates and better ROI on marketing spend.

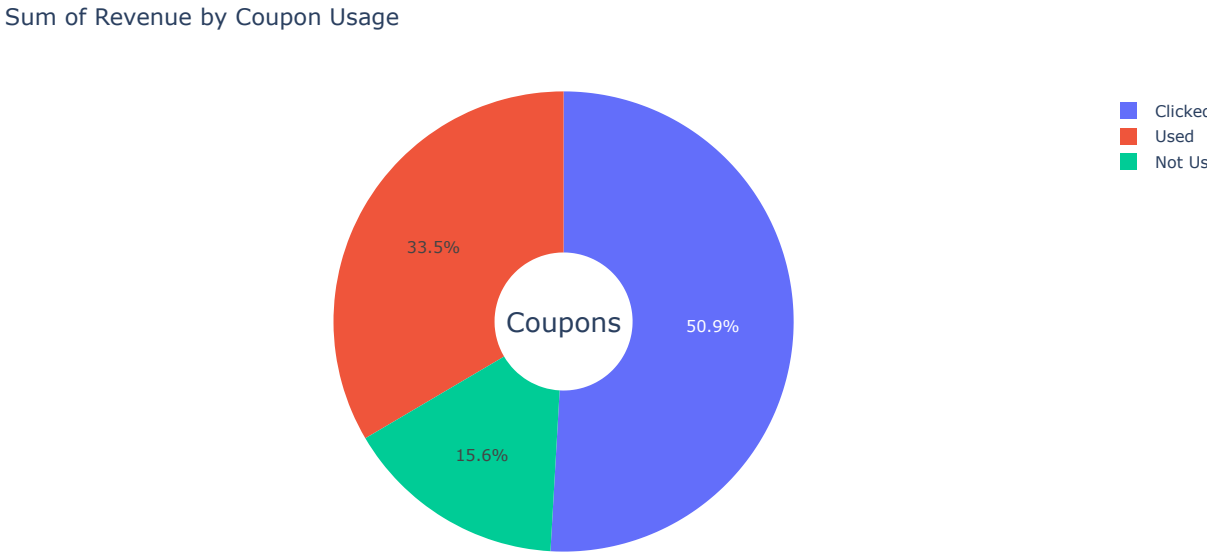
---

2. Exploratory Data Analysis (EDA):

2C. Discount Analysis:

Investigating how discounts and promotions affect revenue and customer engagement:

In [30]:
```
# Group the data by Coupon_Status and sum the 'Invoice' values
grouped = merged_df.groupby('Coupon_Status')['Invoice'].sum()

# Create a Plotly pie chart
fig = go.Figure(data=[go.Pie(labels = grouped.index,
                             values = grouped,
                             hole = .3)])

# Update the layout for better presentation
fig.update_layout(title_text = 'Sum of Revenue by Coupon Usage',
                  annotations = [dict(text = 'Coupons',
                                      x = 0.5,
                                      y = 0.5,
                                      font_size = 20,
                                      showarrow = False)])

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



5A. Observations:

1. Revenue Distribution by Coupon Usage:

- Clicked: 50.9% of the revenue is generated from transactions where customers clicked on a coupon.
- Used: 33.5% of the revenue comes from transactions where customers used a coupon.
- Not Used: 15.6% of the revenue is from transactions where no coupon was used.

5B. Insights:

1. **Effectiveness of Coupons:** The fact that over 50% of the revenue is generated from customers who clicked on a coupon indicates that coupon visibility significantly drives engagement and potential purchases. However, only 33.5% of the revenue comes from actual coupon usage, suggesting a gap between customers clicking on coupons and actually using them.
2. **Customer Behavior:** The 15.6% revenue from transactions without any coupon indicates a segment of customers who might be less price-sensitive or unaware of the coupons available.
3. **Potential Barriers to Coupon Usage:** The difference between clicks (50.9%) and usage (33.5%) might indicate that customers face barriers in applying coupons, such as complex redemption processes, restrictive terms, or irrelevant offers.

5C. Recommendations:

1. **Simplify Coupon Redemption:** Ensure that the process to apply coupons at checkout is straightforward and user-friendly. This could involve minimizing steps or clearly indicating how and where to apply the coupon.
2. **Targeted Marketing:** Use the data from Marketing_Spend.csv and Discount_Coupon.csv to analyze the effectiveness of different marketing channels and coupon campaigns. Focus marketing efforts on channels that yield higher engagement and coupon usage.
3. **Customized Offers:** Leverage customer demographics from Customers_Data.csv to create targeted and personalized coupon offers. For instance, different discounts can be tailored based on customer location, tenure, or purchase history.
4. **Promote Coupon Awareness:** Enhance visibility and awareness of coupons through marketing campaigns. Utilize Marketing_Spend.csv to allocate budget effectively between offline and online channels to reach a broader audience.
5. **Analyze Product Categories:** Use data from Online_Sales.csv and Discount_Coupon.csv to identify which product categories have higher coupon engagement and usage. Focus on promoting these categories with attractive coupon offers.
6. **Feedback and Improvement:** Collect customer feedback regarding coupon usage experiences. Identify pain points and improve the overall coupon offering and redemption experience.

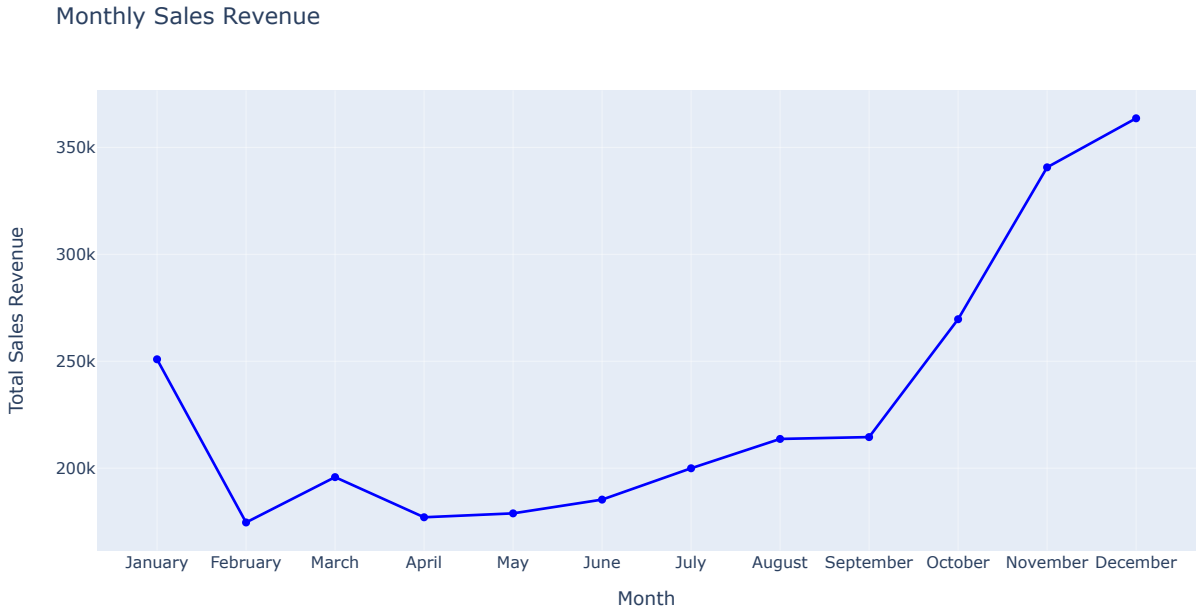
---

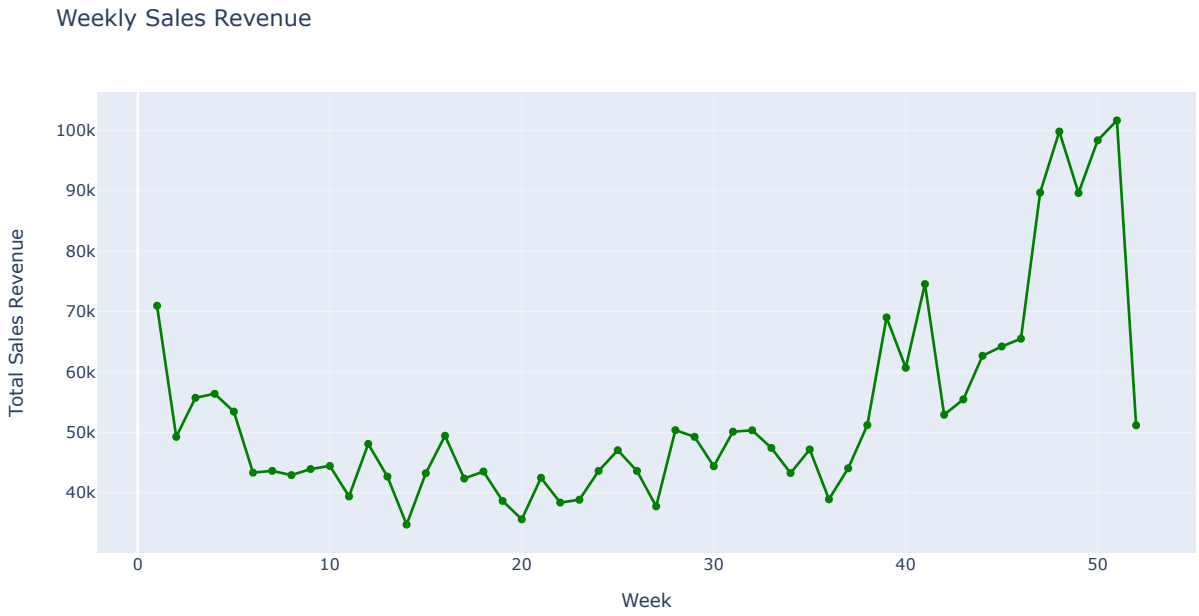
3. Deeper Analysis:

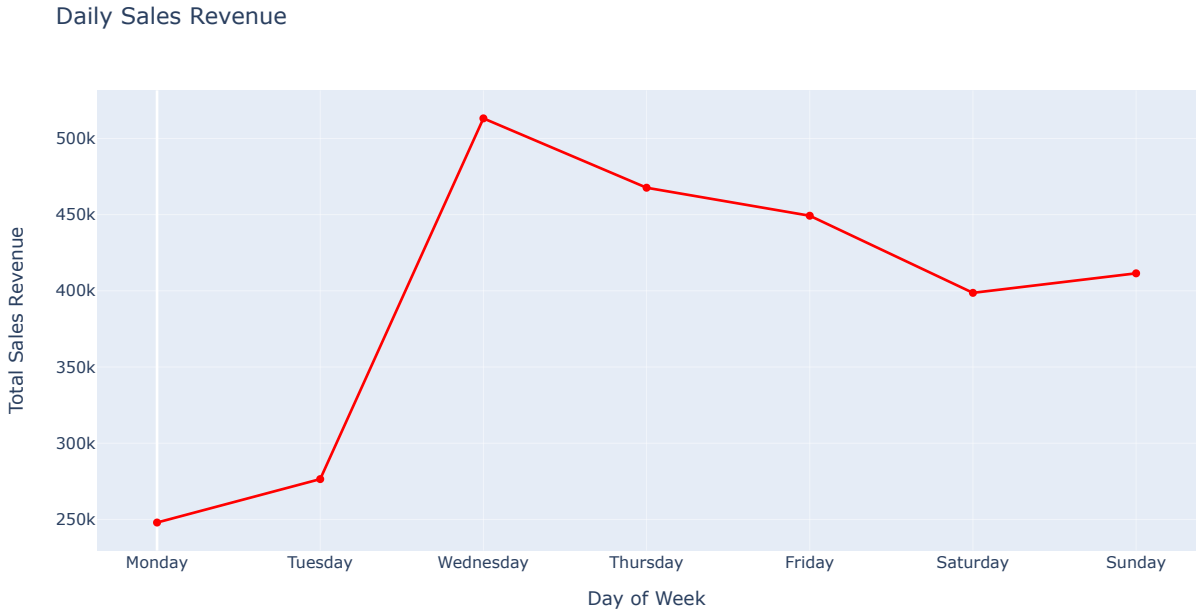
3A. Seasonality and Trends:

Identifying seasonal trends and patterns in sales data across different timeframes (month, week, day) to inform future marketing strategies:

In [31]:
```
# Convert Transaction_Date to datetime if not already done
merged_df['Transaction_Date'] = pd.to_datetime(merged_df['Transaction_Date'])

# Extract month, week, and day of week
merged_df['Transaction_Month'] = merged_df['Transaction_Date'].dt.month
merged_df['Transaction_Week'] = merged_df['Transaction_Date'].dt.isocalendar().week
merged_df['Transaction_Day'] = merged_df['Transaction_Date'].dt.dayofweek

# Group by Transaction_Month to calculate monthly sales revenue
monthly_sales = merged_df.groupby('Transaction_Month')['Avg_Price'].sum().reset_index()

# Group by Transaction_Week to calculate weekly sales revenue
weekly_sales = merged_df.groupby('Transaction_Week')['Avg_Price'].sum().reset_index()

# Group by Transaction_Day to calculate daily sales revenue
daily_sales = merged_df.groupby('Transaction_Day')['Avg_Price'].sum().reset_index()

# Monthly Sales Revenue Plot
fig_monthly = go.Figure()
fig_monthly.add_trace(go.Scatter(x = monthly_sales['Transaction_Month'],
                                 y = monthly_sales['Avg_Price'],
                                 mode = 'lines+markers',
                                 name = 'Monthly Sales',
                                 line = dict(color = 'blue')))
fig_monthly.update_layout(title = 'Monthly Sales Revenue',
                          xaxis_title = 'Month',
                          yaxis_title = 'Total Sales Revenue',
                          xaxis = dict(tickmode = 'array',
                                       tickvals = list(range(1, 13)),
                                       ticktext = calendar.month_name[1:13]))
fig_monthly.show()

# Weekly Sales Revenue Plot
fig_weekly = go.Figure()
fig_weekly.add_trace(go.Scatter(x = weekly_sales['Transaction_Week'],
                                y = weekly_sales['Avg_Price'],
                                mode = 'lines+markers',
                                name = 'Weekly Sales',
                                line = dict(color = 'green')))
fig_weekly.update_layout(title = 'Weekly Sales Revenue',
                         xaxis_title = 'Week',
                         yaxis_title = 'Total Sales Revenue')
fig_weekly.show()

# Daily Sales Revenue Plot
day_names = ['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday']
fig_daily = go.Figure()
fig_daily.add_trace(go.Scatter(x = daily_sales['Transaction_Day'],
                               y = daily_sales['Avg_Price'],
                               mode = 'lines+markers',
                               name = 'Daily Sales',
                               line = dict(color = 'red')))
fig_daily.update_layout(title = 'Daily Sales Revenue',
                        xaxis_title = 'Day of Week',
                        yaxis_title = 'Total Sales Revenue',
                        xaxis = dict(tickmode = 'array',
                                     tickvals = list(range(7)),
                                     ticktext = day_names))
fig_daily.show()
```







5A. Observations:

1. **Daily Sales Revenue:** Sales peak on Wednesday and then gradually decline towards the weekend, with a slight rise on Sunday. The lowest sales occur on Monday, with a steady increase until Wednesday.
2. **Weekly Sales Revenue:** There is a notable spike in sales around week 1 and then a decrease, with another significant increase starting around week 40. The sales trend shows fluctuations throughout the year, with higher consistency in the latter weeks.
3. **Monthly Sales Revenue:** Sales are highest in December, showing a significant upward trend starting from September. February has the lowest sales, followed by a gradual increase over the subsequent months.

5B. Insights:

1. **Daily Patterns:** Mid-week (Wednesday) appears to be the most lucrative day for sales, potentially due to mid-week promotions or consumer purchasing behavior. The dip in sales towards the weekend suggests a possible decline in consumer interest or other external factors affecting sales.
2. **Weekly Trends:** The spike in week 1 could be attributed to New Year's promotions or residual holiday shopping. The increase from week 40 onward indicates a build-up towards the holiday season, with consumers starting their holiday shopping early.
3. **Monthly Trends:** The significant increase from September to December suggests a strong correlation with holiday shopping seasons like Black Friday, Cyber Monday, and Christmas. Lower sales in February might be due to post-holiday spending fatigue or fewer promotions.

5C. Recommendations:

1. **Marketing and Promotions:** Mid-Week Promotions: Since Wednesday shows the highest sales, running special mid-week promotions could capitalize on this trend. Weekend Engagement: Investigate reasons for lower weekend sales and consider targeted campaigns or promotions to boost weekend sales.

2. **Seasonal Strategies:** Early Holiday Promotions: Given the spike starting in week 40, initiating holiday promotions early can capture early shoppers and maximize revenue. Post-Holiday Campaigns: Implement strategies to boost sales in February, such as Valentine's Day promotions or post-holiday discounts.
3. **Customer Retention and Engagement:** Customer Insights: Utilize demographic data from the Customers_Data.csv to personalize marketing efforts based on gender, location, and tenure. Discount Utilization: Analyze the effectiveness of discount coupons from the Discount_Coupon.csv and optimize future discount strategies.
4. **Operational Adjustments:** Align inventory levels with the observed sales trends to ensure high-demand products are well-stocked, especially during peak periods. Adjust staffing and delivery logistics to handle the higher volume during peak sales periods efficiently.

---

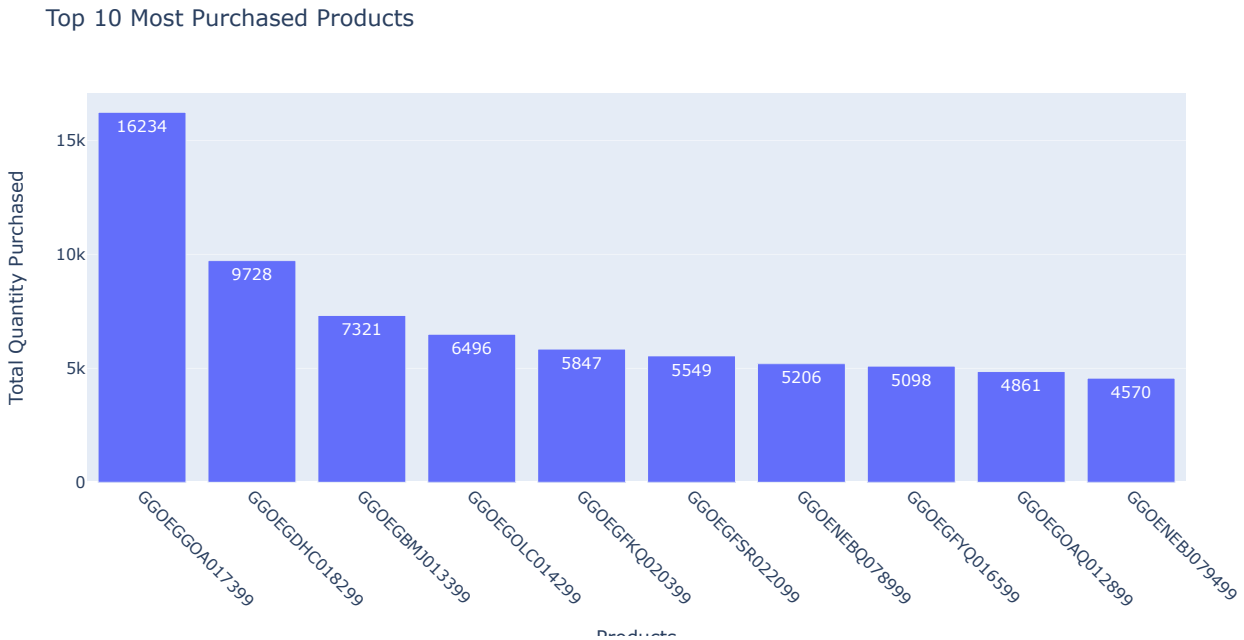
3. Deeper Analysis:

3B. Calculate key performance indicators (KPIs):

KPIs like revenue, number of orders, and average order value across various dimensions (category, month, week, day):

Which are top 10 products purchased?

In [32]:
```
# Calculate total quantity purchased per Product_SKU
category_quantity = merged_df.groupby('Product_SKU')['Quantity'].sum()

# Find the top 10 products with the highest total quantity
top_10_products = category_quantity.sort_values(ascending = False).head(10).reset_index()

# Create a bar chart for the top 10 products
fig = px.bar(top_10_products,
             x = 'Product_SKU',
             y = 'Quantity',
             title = 'Top 10 Most Purchased Products',
             labels = {'Product_SKU': 'Products', 'Quantity': 'Total Quantity Purchased'},
             text = 'Quantity')

# Update layout for better presentation
fig.update_layout(xaxis = dict(tickangle = 45),
                  yaxis = dict(title = 'Total Quantity Purchased'),
                  showlegend = False)

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



5A. Observations:

- **Top Purchased Product:** The product with SKU GGGEGGA017399 is the most purchased product with a total quantity of 16,234 units, significantly higher than the others.
- **Second Most Purchased Product:** The second highest is GGOEGDHC018299 with 9,728 units, followed by GGOEGBM013399 with 7,321 units.
- **Other Products:** The quantities for other products in the top 10 range from around 4,570 to 6,496 units.
- **Drop in Quantity:** There is a noticeable drop in quantity from the first product to the second and a gradual decline from the second to the tenth product.

5B. Insights:

- **Popular Products:** The top product (GGGEGGA017399) is highly popular among customers, indicating a strong preference or higher demand.
- **Demand Variation:** There is a significant variation in demand among the top products. The top product's quantity is almost double that of the second most purchased product.
- **Market Focus:** Focusing on the top products can provide higher returns since they contribute significantly to overall sales.

5C. Recommendations:

- **Inventory Management:** Ensure ample stock availability for the top-selling products, especially GGGEGGA017399, to avoid stockouts and lost sales.
- **Marketing Strategies:** Invest in marketing campaigns for the top products to further boost their sales and capitalize on their popularity. Consider promoting products with moderate sales to elevate their market presence.
- **Analyze Customer Preferences:** Perform deeper analysis on why these products are more popular (e.g., customer reviews, product features) and use these insights to improve or innovate other products.
- **Price Optimization:** Evaluate the pricing strategy for these products. If the demand is highly inelastic, there might be room for price adjustments to maximize revenue.
- **Cross-Selling Opportunities:** Develop cross-selling strategies by recommending complementary products when customers purchase these top items.
- **Product Category Analysis:** Examine the categories of these top products to identify if certain categories are generally more popular and consider expanding those categories.
- **Coupon and Discount Strategies:** Assess the impact of discount coupons on the sales of these products. Optimize coupon strategies to drive more

Which are top products purchased by quantity and product categories?

In [33]:
```
# Calculate total quantity purchased per Product_SKU
category_quantity = merged_df.groupby('Product_SKU')['Quantity'].sum()

# Find the category with the highest total quantity
top_10_categories = category_quantity.sort_values(ascending=False).head(10)

# Reset index for top 10 categories
top10 = top_10_categories.reset_index()

# Filter the original DataFrame for the top 10 products
filtered_df = merged_df[merged_df['Product_SKU'].isin(top10['Product_SKU'])]

# Group by 'Product_SKU' and calculate summary statistics
summary_stats = filtered_df.groupby('Product_SKU').agg({
    'Product_Description': 'first',
    'Product_Category': 'first',
    'Quantity': 'sum',
    'Invoice': 'sum'
}).reset_index()

# Rename columns for better readability
summary_stats.columns = ['Product ID', 'Product Description', 'Product Category', 'Quantity', 'Revenue']

# Sort the DataFrame by Quantity in descending order
summary_stats_sorted = summary_stats.sort_values(by = 'Quantity', ascending = False)

# Plotting with Plotly
fig = px.bar(summary_stats_sorted,
             x = 'Product Description',
             y = 'Quantity',
             color = 'Product Category',
             labels = {'Quantity': 'Total Quantity', 'Product Description': 'Product Description'},
             title = 'Top Products by Quantity (Descending Order)')

# Add annotations to each bar
for i in range(len(summary_stats_sorted)):
    fig.add_annotation(x = summary_stats_sorted['Product Description'].iloc[i],  # X coordinate (product description)
                       y = summary_stats_sorted['Quantity'].iloc[i],  # Y coordinate (quantity)
                       text = f"{summary_stats_sorted['Quantity'].iloc[i]:,.0f}",  # Text for annotation (formatted quantity)
                       showarrow = True,
                       arrowhead = 1,
                       ax = 0,
                       ay = -30)

# Customize layout
fig.update_layout(xaxis_title = 'Product Description',
                  yaxis_title = 'Total Quantity',
                  xaxis_tickangle = -45,
                  plot_bgcolor = 'rgba(0,0,0,0)',
                  showlegend = True)

# Show plot
fig.show()
```

5A. Observations:

- **Top Purchased Product:** The Maze Pen is the most purchased product with a total quantity of 16,234 units.
- **Second Most Purchased Product:** The Google 22 oz Water Bottle follows with 9,728 units, and the Sport Bag is the third with 7,321 units.
- **Product Categories:**
  - **Office:** Includes products like Maze Pen, Google Metallic Notebook Set, Google Laptop and Cell Phone Stickers, and Ballpoint LED Light Pen.
  - **Drinkware:** Google 22 oz Water Bottle.
  - **Bags:** Sport Bag.
  - **Lifestyle:** Google Kick Ball.
  - **Nest-USA:** Nest Cam Outdoor Security Camera - USA and Nest Learning Thermostat 3rd Gen USA - Stainless Steel.
- **Quantity Variation:** The quantities range from 4,570 to 16,234, with the top product having a significantly higher quantity than the others.

5B. Insights:

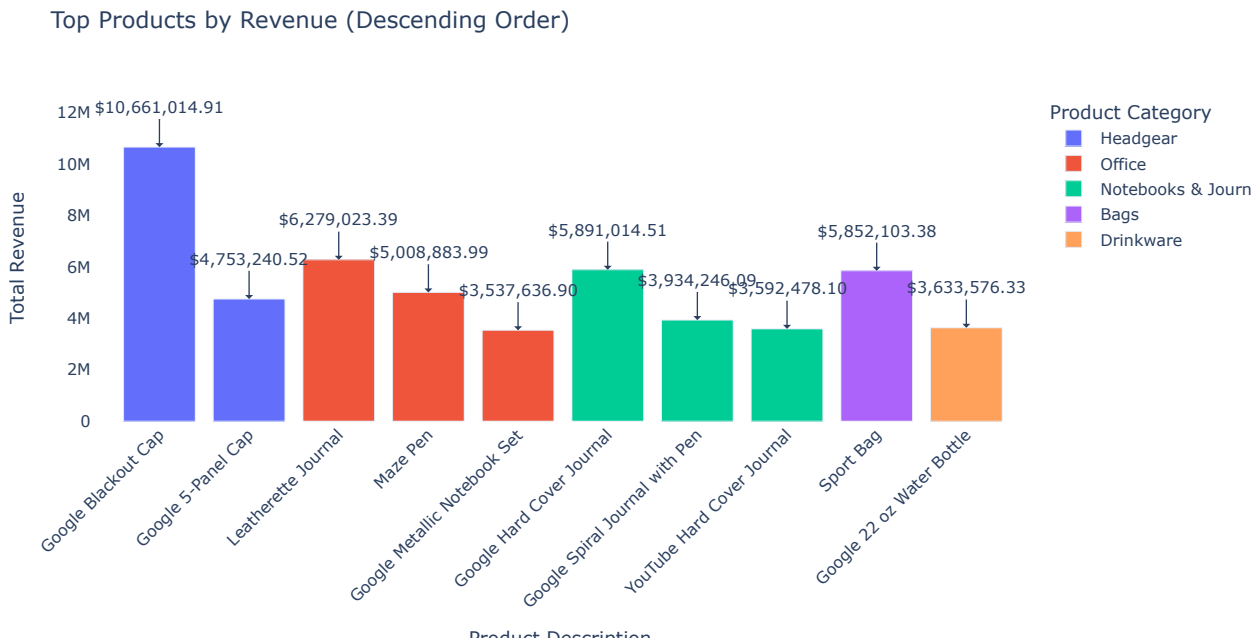
- **Product Popularity:** The Maze Pen is extremely popular, indicating a strong demand. Products in the Office category dominate the top purchased products, showing a higher preference for office-related items.
- **Category Analysis:** Office products have a higher representation among the top products, suggesting that customers have a strong preference for office supplies.
- **Balanced Demand:** The presence of products from different categories (Office, Drinkware, Bags, Lifestyle, Nest-USA) in the top 10 indicates a diverse demand across various product types.

5C. Recommendations:

- **Inventory Management:** Focus on maintaining a robust inventory for the top-selling products, especially the Maze Pen, to meet customer demand without facing stockouts.
- **Targeted Marketing:** Develop marketing campaigns targeting the Office category, as it shows higher customer preference. Additionally, leverage the popularity of specific products in other categories (e.g., Google 22 oz Water Bottle in Drinkware) to drive sales.
- **Product Expansion:** Consider expanding the product lines within the Office category and introducing new variations to capitalize on the existing demand.
- **Promotional Strategies:** Implement promotional strategies for moderately popular products like Google Metallic Notebook Set and Sport Bag to enhance their visibility and sales.
- **Cross-Selling and Bundling:** Create cross-selling opportunities by bundling complementary products, such as offering a Maze Pen with a Google Metallic Notebook Set at a discounted price.
- **Customer Feedback:** Gather and analyze customer feedback on top products to understand their preferences and improve product features or introduce new items that align with their interests.
- **Coupon and Discount Optimization:** Evaluate the effectiveness of discount coupons on the sales of these top products and optimize coupon strategies to maximize sales while maintaining profitability.
- **Diverse Product Offerings:** Ensure a diverse product portfolio to cater to different customer needs, as evidenced by the demand for products across various categories.

Which are top products by revenue and product categories?

In [34]:
```
# Calculate total revenue per Product_SKU
merged_df['Revenue'] = merged_df['Quantity'] * merged_df['Invoice']

# Group by 'Product_SKU' and calculate total revenue
category_revenue = merged_df.groupby('Product_SKU')['Revenue'].sum()

# Find the top 10 products by revenue
top_10_categories_revenue = category_revenue.sort_values(ascending = False).head(10)

# Reset index for top 10 categories
top10_revenue = top_10_categories_revenue.reset_index()

# Filter the original DataFrame for the top 10 products by revenue
filtered_df_revenue = merged_df[merged_df['Product_SKU'].isin(top10_revenue['Product_SKU'])]

# Group by 'Product_SKU' and calculate summary statistics
summary_stats_revenue = filtered_df_revenue.groupby('Product_SKU').agg({
    'Product_Description': 'first',
    'Product_Category': 'first',
    'Quantity': 'sum',
    'Revenue': 'sum'
}).reset_index()

# Rename columns for better readability
summary_stats_revenue.columns = ['Product ID', 'Product Description', 'Product Category', 'Quantity', 'Revenue']

# Sort the DataFrame by Revenue in descending order
summary_stats_sorted_revenue = summary_stats_revenue.sort_values(by = 'Revenue', ascending = False).reset_index(drop = True)

# Plot using Plotly
fig_revenue = px.bar(summary_stats_sorted_revenue,
                     x = 'Product Description',
                     y = 'Revenue',
                     color = 'Product Category',
                     labels = {'Revenue': 'Total Revenue', 'Product Description': 'Product Description'},
                     title = 'Top Products by Revenue (Descending Order)')

# Add annotations to each bar
for i in range(len(summary_stats_sorted_revenue)):
    fig_revenue.add_annotation(x = summary_stats_sorted_revenue['Product Description'].iloc[i],  # X coordinate (product des
                               y = summary_stats_sorted_revenue['Revenue'].iloc[i],  # Y coordinate (revenue)
                               text = f"${summary_stats_sorted_revenue['Revenue'].iloc[i]:,.2f}",  # Text for annotation (fo
                               showarrow = True,
                               arrowhead = 1,
                               ax = 0,
                               ay = -30)

# Customize layout
fig_revenue.update_layout(xaxis_title = 'Product Description',
                          yaxis_title = 'Total Revenue',
                          xaxis_tickangle = -45,
                          plot_bgcolor = 'rgba(0,0,0,0)',
                          showlegend = True)

# Show plot
fig_revenue.show()
```



5A. Observations:

- The plot shows the top products by revenue generated in 2019 for an ecommerce company.
- The most profitable product was a Google Blackout Cap (GGOOGBCAP), which generated over USD 10 million in revenue.
- Other high-revenue products include Leatherette Journal, Google 5-Panel Cap, and Google 22 oz Water Bottle.
- Interestingly, the top revenue-generating product (Google Blackout Cap) is not the same as the top-selling product (Google Maze Pen) by quantity (shown in your previous query).

5B. Insights:

- This discrepancy suggests that the Google Blackout Cap has a higher average selling price than the Google Maze Pen. This could be due to the product itself, or it could be due to different marketing strategies for the two products.
- It is also worth noting that three out of the top five revenue-generating products are from the Headgear category, while none of the top-selling products by quantity were in that category. This suggests that Headgear products tend to be more expensive than Office products, even though Office products may sell in higher volumes.

5C. Recommendations:

- Investigate why the Google Blackout Cap is so profitable. Is it due to a high price point, superior quality, or effective marketing? Understanding the reasons for its success can help the company develop similar products or marketing strategies for other products.
- Consider analyzing the profitability of different product categories. You may find that some categories are more profitable than others, even if they do not sell in as high volumes. This could help you to focus your marketing efforts on the most profitable categories.
- You could also consider developing more high-end products in categories that are typically dominated by lower-priced items. This could be a way to increase your profit margins.

Which are top 5 categories by quantity?

In [35]:
```
# Group by 'Product_Category' and calculate the sum of 'Invoice' and 'Quantity'
category_summary = merged_df.groupby('Product_Category').agg({
    'Invoice': 'sum',
    'Quantity': 'sum'
})

# Get the top 5 categories by revenue (Invoice)
top_categories_Quantity = category_summary.nlargest(5, 'Quantity')

# Display the top 5 categories by revenue
top_categories_Quantity
```

Out[35]:

Product_Category	Invoice	Quantity
Office	370637.826	88383
Apparel	756240.060	32438
Drinkware	247551.160	30501
Lifestyle	116471.508	24881
Nest-USA	2619135.650	21430

5A and 5B. Observations and Insights:

1. **High Revenue from Apparel:** The Apparel category has the highest invoice value (USD 756,240.060) among the listed categories, indicating that this category is a major revenue driver.
2. **High Volume from Office Products:** The Office category has the highest quantity sold (88,383 units). This suggests a high demand for office products despite the lower total invoice value compared to Apparel.
3. **Low Volume but High Invoice from Nest-USA:** The Nest-USA category shows a high invoice value (USD 2,619,135.650) with a relatively low quantity sold (21,430 units). This implies that products in this category are high-value items.
4. **Moderate Performance of Drinkware and Lifestyle:** Drinkware and Lifestyle categories have moderate invoice values (USD 247,551.160 and USD 116,471.508 respectively) and quantities sold (30,501 and 24,881 units respectively). These categories may represent stable, consistent sales but do not dominate the sales figures.

5C. Recommendations:

1. **Focus on High Revenue Categories:** Given the significant revenue from Apparel and high-value items in Nest-USA, focus marketing and promotional efforts on these categories to maximize revenue. Consider offering special discounts or bundles to attract more customers to these high-revenue categories.
2. **Promote High Volume Categories:** Since Office products have high sales volume, there is an opportunity to increase profitability by optimizing the pricing strategy or reducing costs. Additionally, consider loyalty programs or bulk purchase discounts for office supplies to further increase sales volume.
3. **Explore Upselling and Cross-Selling:** Implement upselling and cross-selling strategies, especially for high-value items in the Nest-USA category. For example, recommend complementary products or higher-end versions of the products to customers.
4. **Marketing Spend Allocation:** Allocate a higher portion of the marketing budget to online channels targeting high-revenue and high-volume categories. Use data-driven marketing strategies to target specific customer segments who are likely to purchase these products.
5. **Customer Segmentation:** Analyze customer demographics to identify key customer segments for each product category. Tailor marketing campaigns to these segments to increase customer engagement and sales.
6. **Monitor Discount Effectiveness:** Use data from Discount_Coupon.csv to analyze the effectiveness of discount campaigns. Focus on categories and months where discounts lead to significant sales increases. Adjust future discount strategies based on these insights.

Which are top 5 product categories by revenue?

In [36]:
```
# Group by 'Product_Category' and calculate the sum of 'Invoice' and 'Quantity'
category_summary = merged_df.groupby('Product_Category').agg({
    'Invoice': 'sum',
    'Quantity': 'sum'
})

# Get the top 5 categories by revenue (Invoice)
top_categories_Invoice = category_summary.nlargest(5, 'Invoice')

# Display the top 5 categories by revenue
top_categories_Invoice
```

Out[36]:

Product_Category	Invoice	Quantity
Nest-USA	2619135.650	21430
Apparel	756240.060	32438
Nest	534276.300	2837
Office	370637.826	88383
Drinkware	247551.160	30501

5A and 5B. Observations and Insights:

1. **Dominant Revenue from Nest-USA:** The Nest-USA category leads with an invoice value of USD 2,619,135.650, suggesting it is a significant revenue contributor despite having a lower quantity sold (21,430 units). This indicates high-priced items in this category.
2. **High Volume from Office Products:** Office products have the highest quantity sold (88,383 units), but with a moderate invoice value of USD 370,637.826. This points to a high demand for lower-priced items in this category.
3. **Strong Performance of Apparel:** The Apparel category shows strong performance with an invoice value of USD 756,240.060 and a relatively high quantity sold (32,438 units), indicating both high demand and significant revenue generation.
4. **High Value from Nest:** The Nest category, with an invoice value of USD 534,276.300 and only 2,837 units sold, indicates very high-value items.
5. **Moderate Sales for Drinkware:** Drinkware has a moderate invoice value of USD 247,551.160 and quantity sold (30,501 units), suggesting stable but not outstanding sales.

5C. Recommendations:

1. **Focus on High Revenue and High-Value Categories:** Prioritize marketing efforts for Nest-USA and Nest categories due to their high revenue and high-value items. Use targeted advertising and personalized promotions to attract high-value customers.
2. **Increase Profitability in High Volume Categories:** For Office products, which have the highest sales volume, consider strategies to increase profitability, such as negotiating better supplier prices, reducing delivery costs, or increasing prices slightly without affecting demand. Promotions like bulk purchase discounts can also boost sales volume further.
3. **Leverage Apparel Popularity:** Given the strong performance of the Apparel category, continue to promote this category heavily. Consider seasonal sales, new product launches, and collaborations with influencers to maintain and grow customer interest.
4. **Optimize Pricing and Promotions:** For categories with moderate performance like Drinkware, review pricing strategies and explore promotions to increase their appeal. Offering limited-time discounts or bundling products can stimulate sales.

5. **Enhanced Marketing Spend Allocation:** Allocate more marketing spend to online channels, specifically targeting high-revenue categories like Nest-USA and Apparel. Use data analytics to identify the most effective channels and customer segments for these categories.
6. **Customer Segmentation and Personalization:** Utilize customer demographic data (from Customers_Data.csv) to create targeted marketing campaigns for different segments. Personalize offers and communications to match customer preferences and buying behaviors.
7. **Evaluate Discount Campaigns:** Assess the impact of discount campaigns using data from Discount_Coupon.csv. Identify the most effective discount strategies and apply them to boost sales in slower-moving categories.

Which are top 5 revenue days?

In [37]:
```
# ANSI escape code for bold text
bold_start = "\033[1m"
bold_end = "\033[0m"


merged_df['Date'] = merged_df['Transaction_Date'].dt.date
top_days = merged_df.groupby('Date')['Invoice'].sum().nlargest(5)

print(f"{bold_start}\nTop 5 revenue days:{bold_end}")
top_days
```

Top 5 revenue days:

Out[37]:
```
Date
2019-04-05    56590.93
2019-11-27    56113.39
2019-04-18    48930.24
2019-08-02    41864.69
2019-07-18    40022.75
Name: Invoice, dtype: float64
```

5A and 5B. Observations and Insights:

1. **Peak Revenue Day - April 5, 2019:** The highest revenue day is April 5, 2019, with USD 56,590.93. This indicates a significant sales event or promotion on this day.
2. **Strong Performance in Late November:** November 27, 2019, also shows high revenue with USD 56,113.39. This coincides with Black Friday sales, suggesting effective marketing and promotional strategies around this period.
3. **High Sales in Mid-April:** April 18, 2019, generated USD 48,930.24, making it another high revenue day in April. This could be due to a mid-month promotion or seasonal factors.
4. **Consistent Performance in Summer:** August 2, 2019, and July 18, 2019, are notable with revenues of USD 41,864.69 and USD 40,022.75, respectively. This indicates strong summer sales, potentially due to back-to-school promotions or summer clearance sales.

5C. Recommendations:

1. **Replicate Successful Promotions:** Analyze the specific promotions, marketing campaigns, or events that led to high revenues on these top days. Replicate these strategies during similar periods in the future to boost sales.
2. **Seasonal and Event-Based Marketing:** Plan and execute targeted marketing campaigns around significant shopping events such as Black Friday (November) and mid-year sales (April). Utilize customer data to tailor promotions to maximize impact.
3. **Focus on High-Performance Months:** Given the strong sales performance in April, July, and August, concentrate marketing efforts and promotional budgets during these months. Consider introducing special events or product launches during these high-potential periods.
4. **Leverage Historical Data:** Use historical sales data to predict and prepare for potential high-revenue days. Ensure adequate inventory and staffing to handle increased demand during these peak periods.
5. **Customer Engagement and Retention:** Implement loyalty programs and personalized offers to engage customers leading up to and during high-revenue periods. Encourage repeat purchases and customer retention through exclusive deals for loyal customers.
6. **Monitor and Adjust Strategies:** Continuously monitor the effectiveness of promotions and adjust strategies based on real-time data and customer feedback. Be flexible in adapting marketing tactics to maximize revenue.

Which are top 5 revenue weeks?

In [38]:
```
# ANSI escape code for bold text
bold_start = "\033[1m"
bold_end = "\033[0m"


merged_df['Week'] = merged_df['Transaction_Date'].dt.strftime('%Y-%U')
top_weeks = merged_df.groupby('Week')['Invoice'].sum().nlargest(5)

print(f"{bold_start}\nTop 5 revenue weeks:{bold_end}")
print(top_weeks)
```

Top 5 revenue weeks:
```
Week
2019-47    169313.66
2019-50    160696.02
2019-49    152965.89
2019-48    136491.21
2019-30    127810.18
Name: Invoice, dtype: float64
```

5A and 5B. Observations and Insights:

1. **High Revenue in Late November:** Weeks 47 (November) to 50 (December) are the highest revenue weeks, with Week 47 leading at USD 169,313.66. This period likely benefits from Black Friday, Cyber Monday, and early holiday shopping.
2. **Consistent Peak in Late November to Early December:** Weeks 48 and 49 also show strong revenues of USD 136,491.21 and USD 152,965.89, respectively. This indicates a sustained high sales period around the end of November to early December.
3. **Notable Mid-Summer Sales:** Week 30 (July) has a high revenue of USD 127,810.18, indicating a successful mid-summer sales period, potentially due to back-to-school promotions or mid-year sales events.

5C. Recommendations:

1. **Capitalize on Holiday Shopping Period:** Strengthen marketing and promotional efforts during the late November to early December period to capitalize on the holiday shopping season. Offer special discounts, limited-time offers, and holiday bundles to attract shoppers.
2. **Optimize Mid-Summer Promotions:** Leverage the strong sales performance in Week 30 by planning significant promotions or sales events in mid-summer. Back-to-school promotions, summer clearance sales, and exclusive deals can drive revenue during this period.
3. **Increase Inventory and Staffing:** Ensure adequate inventory and staffing levels during peak revenue weeks to meet increased demand. Avoid stockouts and delays by preparing in advance for high sales volumes.
4. **Engage Customers with Targeted Campaigns:** Use targeted marketing campaigns to engage customers during these high-revenue weeks. Personalize offers based on customer preferences and purchasing history to increase conversion rates.
5. **Analyze and Replicate Successful Strategies:** Analyze the specific strategies and campaigns that led to high revenues during these weeks. Replicate and refine these strategies for future use, ensuring continued success during peak periods.
6. **Plan for Early Promotions:** Start holiday promotions early in November to build momentum leading into the peak weeks. Early bird discounts and pre-holiday sales can attract early shoppers and spread out the demand.
7. **Monitor and Adapt:** Continuously monitor sales data during these high-revenue weeks and be prepared to adapt strategies based on real-time performance. Flexibility in marketing tactics can help maximize revenue.

Which are top 2 months by revenue?

In [39]:
```
# ANSI escape code for bold text
bold_start = "\033[1m"
bold_end = "\033[0m"


top_months = merged_df.groupby('Month')['Invoice'].sum().nlargest(2)

print(f"{bold_start}\nTop 2 months by revenue:{bold_end}")
print(top_months)
```

```
Top 2 months by revenue:
Month
December    561140.18
November    541254.55
Name: Invoice, dtype: float64
```

5A and 5B. Observations and Insights:

1. **Highest Revenue in December:** December is the top revenue month with USD 561,140.18, indicating a significant boost from holiday shopping, end-of-year sales, and potentially gift purchases.
2. **Strong Performance in November:** November follows closely with USD 541,254.55, driven by major shopping events like Black Friday and Cyber Monday, as well as early holiday shopping.

5C. Recommendations:

1. **Maximize Holiday Sales:** Focus heavily on marketing and promotional efforts in November and December to capitalize on the holiday shopping season. Implement holiday-themed campaigns, gift guides, and special discounts to attract customers.
2. **Early and Extended Promotions:** Start promotions early in November and extend them through December to maintain momentum and capture early and late holiday shoppers. Early bird specials and extended sales can spread out demand and boost overall revenue.
3. **Holiday Bundles and Gift Sets:** Create holiday bundles and gift sets to increase average order value. Promote these as convenient gift options to attract holiday shoppers looking for quick and easy gift ideas.
4. **Enhanced Customer Engagement:** Engage customers with personalized email campaigns, retargeting ads, and social media promotions. Use customer data to tailor offers and recommendations, enhancing the shopping experience and increasing conversions.
5. **Optimize Website and Mobile Experience:** Ensure the website and mobile app are optimized for high traffic and provide a seamless shopping experience. Fast loading times, easy navigation, and a smooth checkout process can reduce cart abandonment and increase sales.
6. **Inventory Management:** Prepare for high demand by managing inventory effectively. Ensure popular items are well-stocked and consider offering expedited shipping options to meet customer expectations for holiday delivery.
7. **Customer Support and Service:** Provide excellent customer support during these peak months. Offer live chat, extended support hours, and easy returns to enhance customer satisfaction and build loyalty.

Which are top 2 months by order count?

In [40]:
```
# ANSI escape code for bold text
bold_start = "\033[1m"
bold_end = "\033[0m"


top_months = merged_df.groupby('Month')['Transaction_ID'].nunique().nlargest(2)

print(f"{bold_start}\nTop 2 months by order count:{bold_end}")
print(top_months)
```

Top 2 months by order count:
```
Month
December    2684
August      2414
Name: Transaction_ID, dtype: int64
```

5A and 5B. Observations and Insights:

1. **Highest Order Count in December:** December has the highest order count with 2,684 orders, aligning with the holiday shopping season and indicating high customer activity and engagement.
2. **Strong Order Count in August:** August follows with 2,414 orders, suggesting a significant shopping period, possibly driven by back-to-school promotions and summer sales.

5C. Recommendations:

1. **Enhance Holiday Campaigns in December:** Focus on creating compelling holiday marketing campaigns in December to maintain high order volumes. Use festive themes, special discounts, and gift recommendations to attract and engage customers.
2. **Leverage Back-to-School Promotions in August:** Capitalize on the high order volume in August by promoting back-to-school sales and summer clearance events. Offer discounts on relevant products and bundle deals to increase order counts.
3. **Personalized Marketing:** Utilize customer data to send personalized marketing messages and recommendations during these peak months. Tailored offers can improve customer engagement and drive repeat purchases.
4. **Optimize Inventory and Supply Chain:** Ensure that inventory levels are sufficient to meet the high demand in December and August. Optimize the supply chain to prevent stockouts and ensure timely deliveries, especially for popular items.
5. **Improve User Experience:** Enhance the user experience on the website and mobile app to handle increased traffic smoothly. Ensure fast load times, easy navigation, and a seamless checkout process to reduce cart abandonment rates.
6. **Offer Incentives for Larger Orders:** Encourage customers to place larger orders by offering incentives such as free shipping, bulk discounts, or loyalty points. This can help boost the average order value and overall sales.
7. **Extend Customer Support:** Provide extended customer support hours during these peak months to assist with order inquiries, returns, and other customer service issues. Excellent support can enhance customer satisfaction and loyalty.
8. **Analyze and Adapt:** Continuously analyze sales data and customer feedback from these peak months to understand what drives high order volumes. Adapt marketing and operational strategies based on these insights to improve performance in future high-demand periods.

---

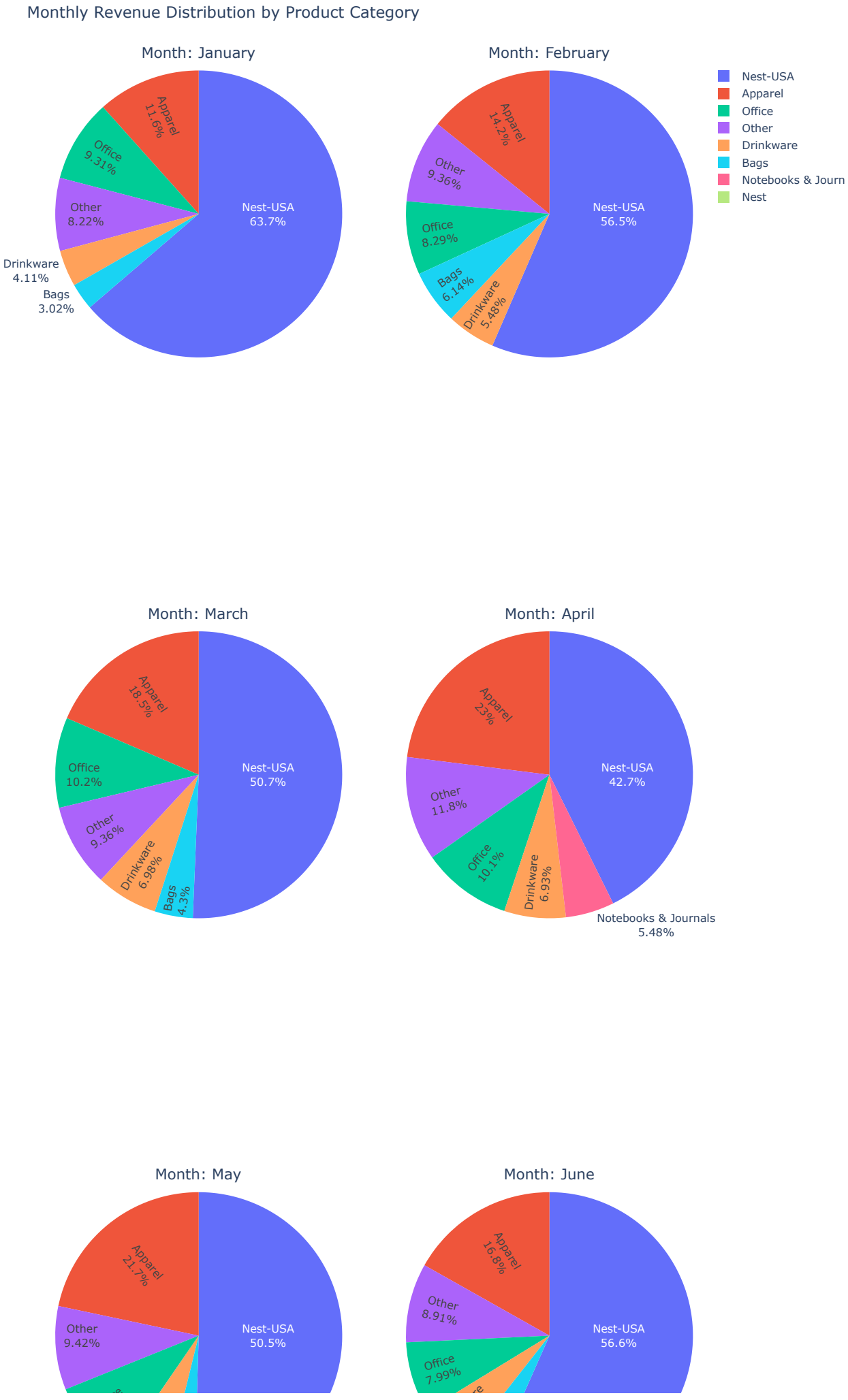
3. Deeper Analysis:

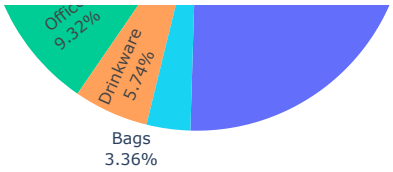
3C. Marketing Spend and Revenue:

Calculating revenue, marketing spend, and delivery charges by month to understand their correlation. This can reveal areas for optimization:

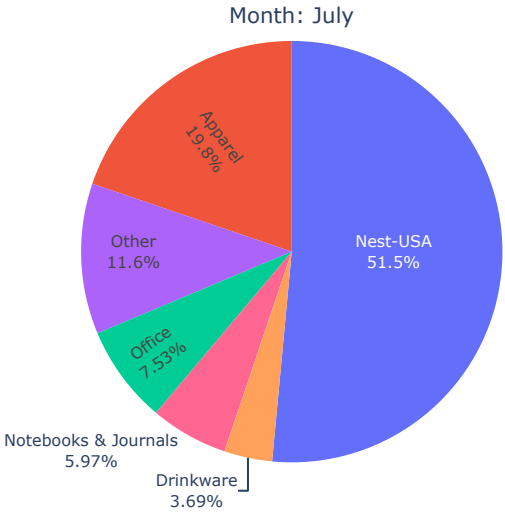
In [41]:

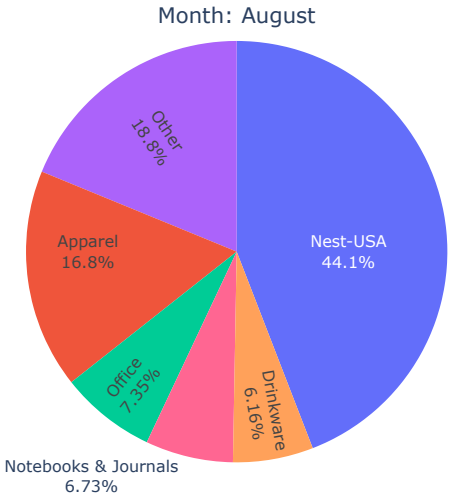
```
# Prepare the data
months = merged_df['Month'].unique()
categories_to_show = 5

# Create subplots
num_rows = int(len(months) / 2) + (len(months) % 2)
num_cols = 2  # Adjust as needed
fig = make_subplots(rows = num_rows,
                    cols = num_cols,
                    specs = [[{'type': 'pie'} for _ in range(num_cols)] for _ in range(num_rows)],
                    subplot_titles = [f'Month: {month}' for month in months])

for i, month in enumerate(months):
    month_data = merged_df[merged_df['Month'] == month]
    total_invoice = month_data.groupby('Product_Category')['Invoice'].sum()
    total_invoice = total_invoice.sort_values(ascending = False)  # Sort the total_invoice

    if len(total_invoice) > categories_to_show:
        other_sum = total_invoice.iloc[categories_to_show:].sum()
        total_invoice = total_invoice.iloc[:categories_to_show]
        total_invoice['Other'] = other_sum

    row = (i // num_cols) + 1
    col = (i % num_cols) + 1

    fig.add_trace(go.Pie(labels = total_invoice.index,
                         values = total_invoice,
                         textinfo = 'label+percent',
                         insidetextorientation = 'radial'),
                  row = row,
                  col = col)

# Update layout
fig.update_layout(height = 600 * num_rows,
                  title_text = "Monthly Revenue Distribution by Product Category")

fig.show()
```

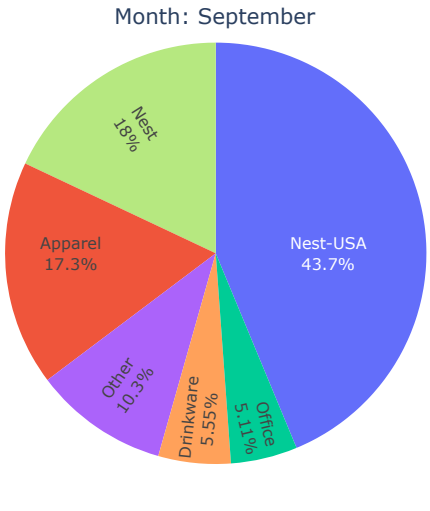

Monthly Revenue Distribution by Product Category
Month: January
Nest-USA 63.7%
Apparel 11.6%
Office 9.31%
Other 8.22%
Drinkware 4.11%
Bags 3.02%
Month: February
Nest-USA 56.5%
Apparel 14.2%
Other 9.36%
Office 8.29%
Bags 6.14%
Drinkware 5.48%
Nest-USA
Apparel
Office
Other
Drinkware
Bags
Notebooks & Journ
Nest
Month: March
Nest-USA 50.7%
Apparel 18.5%
Office 10.2%
Other 9.36%
Drinkware 6.98%
Bags 4.3%
Month: April
Nest-USA 42.7%
Apparel 23%
Other 11.8%
Office 10.1%
Drinkware 6.93%
Notebooks & Journals 5.48%
Month: May
Nest-USA 50.5%
Apparel 21.7%
Other 9.42%
Month: June
Nest-USA 56.6%
Apparel 16.8%
Other 8.91%
Office 7.99%

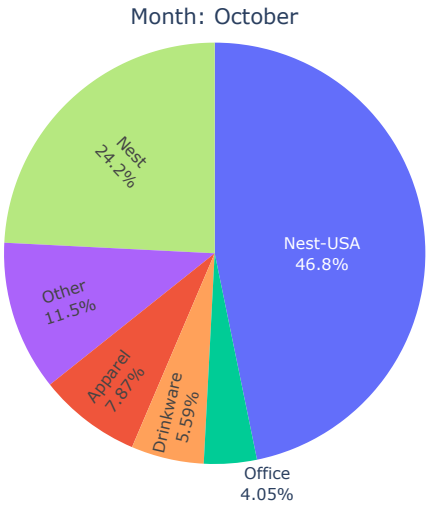

Drinkware
5.74%
Bags
3.36%

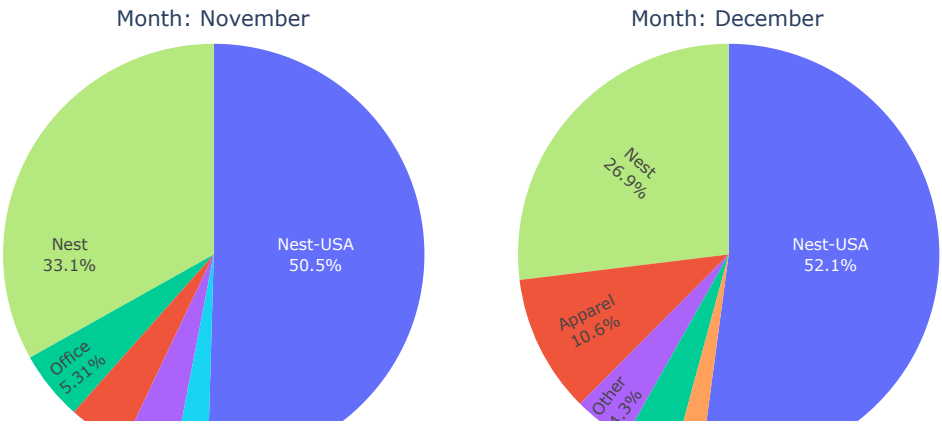

Drinkwa
5.66%
Bags
3.98%


Month: July
Nest-USA
51.5%
Apparel
19.8%
Other
11.6%
Office
7.53%
Notebooks & Journals
5.97%
Drinkware
3.69%


Month: August
Nest-USA
44.1%
Other
18.8%
Apparel
16.8%
Office
7.35%
Drinkware
6.16%
Notebooks & Journals
6.73%


Month: September
Nest-USA
43.7%
Nest
18%
Apparel
17.3%
Other
10.3%
Drinkware
5.55%
Office
5.11%


Month: October
Nest-USA
46.8%
Nest
24.2%
Other
11.5%
Apparel
7.87%
Drinkware
5.59%
Office
4.05%



5A and 5B. Observations and Insights:

1. **Dominance of Nest-USA Across All Months:** The Nest-USA category consistently generates the highest revenue each month, ranging from 30.1% in November to 63.7% in January. This indicates a strong and steady demand for Nest-USA products throughout the year.
2. **Significant Revenue from Apparel in Specific Months:** Apparel shows notable revenue contributions in August (16.8%), October (17.3%), and December (10.9%). This suggests seasonal trends, likely related to back-to-school shopping and holiday seasons.
3. **Consistent Revenue from Office Products:** Office products contribute significantly to revenue in several months, with peaks in October (11.7%) and December (9.5%). This could be due to business procurement cycles or promotions targeting office supplies.
4. **Other Categories with Variable Contributions:** Other categories, such as Drinkware and Notebooks & Journals, have variable revenue contributions, often peaking in specific months. For example, Notebooks & Journals contribute 5.48% in April and 5.97% in July.

5C. Recommendations:

1. **Focus on Nest-USA Product Promotions:** Given the consistent high revenue from Nest-USA products, continue to invest in marketing and promotional activities for this category. Consider introducing new products or variants to maintain customer interest and drive sales.
2. **Seasonal Campaigns for Apparel:** Leverage the peak months for apparel (August, October, and December) by planning targeted marketing campaigns. Offer back-to-school promotions in August and holiday deals in December to maximize revenue from this category.
3. **Boost Office Product Sales:** Increase promotional efforts for office products during months with historically high sales, such as October and December. Consider business-to-business marketing strategies to attract corporate clients and bulk purchases.
4. **Enhance Visibility for Other Categories:** Identify months where other categories, such as Drinkware and Notebooks & Journals, perform well and increase their visibility during these periods. Use targeted ads, special offers, and bundling strategies to boost their sales.
5. **Inventory and Supply Chain Optimization:** Ensure adequate inventory levels for high-demand categories, especially Nest-USA products, to prevent stockouts and meet customer demand. Optimize the supply chain to handle peak sales periods efficiently.
6. **Customer Segmentation and Personalization:** Use customer segmentation to understand purchasing patterns and preferences for different product categories. Tailor marketing messages and offers to different customer segments to increase engagement and conversion rates.
7. **Monitor and Adapt Strategies:** Continuously monitor sales data and customer feedback to identify emerging trends and adjust marketing strategies accordingly. Be flexible in adapting to changes in customer behavior and market conditions.

---

4. Cohort Analysis

4A. Create customer cohorts based on their acquisition month:

Tracking their behavior (orders, revenue) over time to identify the cohort with the highest retention rate. This reveals valuable customer acquisition trends:

In [42]:
```
cohorts = merged_df.groupby('Month')

# Calculate metrics for each cohort
cohort_metrics = cohorts.agg({
    'CustomerID': 'nunique',  # Count unique customers
    'Invoice': ['count','sum']  # Count total invoices
})

# # Rename columns for clarity
cohort_metrics.columns = cohort_metrics.columns.to_flat_index()
cohort_metrics.columns = ['Unique Customers', 'Total Invoices', 'Total Invoices Amount']

# Calculate cohort retention rates
cohort_size = cohort_metrics.iloc[:, 0]
retention = cohort_metrics.divide(cohort_size, axis=0)

# Find the month cohort with maximum retention
max_retention_month = cohort_metrics['Unique Customers'].idxmax()

# Display the cohort analysis results
print("Cohort Metrics:")
print(cohort_metrics)
print("\nCohort Retention Rates:")
print(retention)
print("\nMonth cohort with maximum retention:", max_retention_month)
```

```
Cohort Metrics:
           Unique Customers  Total Invoices  Total Invoices Amount
Month
April                   224            4150             443100.160
August                  300            6150             462309.940
December                236            4502             561140.180
February                109            3284             360036.400
January                 215            4063             462866.900
July                    236            5251             421362.000
June                    259            4193             358594.960
March                   208            4346             410408.030
May                     200            4572             288368.234
November                188            3961             541254.550
October                 210            4164             455643.160
September               193            4288             401553.820

Cohort Retention Rates:
           Unique Customers  Total Invoices  Total Invoices Amount
Month
April                   1.0       18.526786            1978.125714
August                  1.0       20.500000            1541.033133
December                1.0       19.076271            2377.712627
February                1.0       30.128440            3303.086239
January                 1.0       18.897674            2152.869302
July                    1.0       22.250000            1785.432203
June                    1.0       16.189189            1384.536525
March                   1.0       20.894231            1973.115529
May                     1.0       22.860000            1441.841170
November                1.0       21.069149            2879.013564
October                 1.0       19.828571            2169.729333
September               1.0       22.217617            2080.589741

Month cohort with maximum retention: August
```

---

5. Recommendations:

I have already provided insights and recommendations in each cell and after almost each chart. However, I am also providing a summarized, more clear and consise list of recommendations:

1. Marketing and Promotions:

- **Enhanced Marketing During Low Months:** Increase marketing efforts in February and September to boost customer acquisition with special campaigns or incentives.
- **Leverage Peak Seasons:** Plan significant marketing campaigns and discounts during peak acquisition periods (June, August, December) and ensure inventory and customer support are prepared for increased demand.
- **Promotion Strategies:** Review and optimize discount coupons and experiment with different types of promotions (e.g., bundle deals, limited-time offers).
- **Seasonal Strategies:** Run targeted campaigns during low retention months (February, April, September) and reinforce marketing efforts during high retention periods.
- **Mid-Week and Seasonal Promotions:** Implement special mid-week promotions on Wednesdays and early holiday promotions starting in week 40, leveraging strong sales periods.

2. Customer Acquisition and Retention:

- **Sustain Customer Acquisition Efforts:** Implement periodic promotions, referral programs, and targeted advertising campaigns to avoid sharp declines in new customer revenue after January.

- **Improving Customer Retention:** Implement loyalty programs, improve post-purchase engagement, and use customer feedback to identify and address pain points.
- **Address Post-Holiday Churn:** Implement targeted retention strategies in January and analyze high churn rates to identify common factors.

3. Customer Segmentation and Personalization:

- **Utilize Customer Demographics:** Tailor marketing campaigns based on customer demographics, targeting locations and segments with higher engagement and acquisition rates.
- **Customer Segmentation:** Segment customers based on purchasing behavior and demographics to create targeted marketing strategies and improve personalization.

4. Coupon Strategy:

- Investigate customer behavior to understand why some customers abandon their carts after clicking on a coupon.
- Conduct surveys or focus groups and offer different types of coupons (e.g., free shipping or percentage discounts).
- A/B test coupon placements on the website to optimize visibility and usage, and streamline the checkout process to reduce cart abandonment.

5. Inventory and Supply Chain Management:

- **Inventory Optimization:** Ensure high-demand products are well-stocked, especially during peak sales periods, and align inventory levels with observed sales trends.
- **Enhance Supply Chain Efficiency:** Optimize the supply chain to handle peak sales periods efficiently and prevent stockouts, particularly for high-demand categories.

6. Marketing Spend Analysis and Optimization:

- **Analyze Marketing Spend Efficiency:** Conduct detailed analysis to understand the correlation between marketing spend and customer acquisition, and optimize budgets accordingly.
- **Enhanced Attribution Analysis:** Conduct detailed attribution analysis to better understand the impact of online and offline marketing spend on sales.

7. Product and Category Focus:

- **Focus on High Revenue and High-Value Categories:** Prioritize marketing for high-revenue categories like Nest-USA and Apparel, and explore upselling and cross-selling opportunities.
- **Promote High Volume Categories:** Increase profitability of high-volume categories (Office products) by optimizing pricing strategy and exploring bulk purchase discounts.
- **Product Expansion and Optimization:** Consider expanding product lines within high-demand categories and promoting products with moderate sales to elevate their market presence.

8. Customer Feedback and Continuous Improvement:

- **Analyze Customer Feedback:** Collect and analyze customer feedback to identify common pain points and areas for improvement, and address these promptly.
- **Monitor and Adapt Strategies:** Continuously monitor sales data and customer feedback to identify trends and adjust marketing strategies accordingly.

9. Operational Adjustments:

- **Optimize Staffing and Logistics:** Adjust staffing and delivery logistics to handle higher volume during peak sales periods efficiently.

10. Enhanced Customer Experience:

- **Improve User Experience:** Ensure the website and mobile app provide a seamless shopping experience, particularly during peak periods, to reduce cart abandonment rates.
- **Customer Support and Service:** Provide excellent customer support, especially during peak months, to enhance customer satisfaction and loyalty.

By implementing these recommendations, you can improve your marketing efficiency, enhance customer acquisition and retention, and optimize overall