

CUSTOMER SHOPPING BEHAVIOR ANALYSIS



Project By Hernandia Zanua Leras

BACKGROUND



The retail company aims to better understand customer shopping behavior to improve sales performance, customer satisfaction, and long-term loyalty. Changes in purchasing patterns across customers and products highlight the need to identify factors that influence buying decisions, such as product preferences, pricing, and shopping trends. Analyzing consumer shopping data helps uncover insights that support more effective marketing strategies and product optimization.

PROJECT OVERVIEW

OBJECTIVE

- Analyze customer shopping behavior to identify purchasing patterns and key factors influencing sales.
- Evaluate product and customer performance to support data-driven marketing and business decisions.

TOOLS

- **Python** – used for data cleaning and data preparation.
- **MySQL** – used for data analysis and querying insights.



VISUALIZATION

- **Power BI** – used for data visualization and dashboard creation.



Dataset Overview

| A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R |
|-------------|-----|--------|----------------|-------------|-----------------|----------------|------|-----------|--------|---------------|---------------------|----------------|------------------|-----------------|--------------------|----------------|------------------------|
| customer_id | age | gender | item_purchased | category | purchase_amount | location | size | color | season | review_rating | subscription_status | shipping_type | discount_applied | promo_code_used | previous_purchases | payment_method | frequency_of_purchases |
| 1 | 55 | Male | Blouse | Clothing | 53 | Kentucky | L | Gray | Winter | 3.1 | Yes | Express | Yes | Yes | 14 | Venmo | Fortnightly |
| 2 | 19 | Male | Sweater | Clothing | 64 | Maine | L | Maroon | Winter | 3.1 | Yes | Express | Yes | Yes | 2 | Cash | Fortnightly |
| 3 | 50 | Male | Jeans | Clothing | 73 | Massachusetts | S | Maroon | Spring | 3.1 | Yes | Free Shipping | Yes | Yes | 23 | Credit Card | Weekly |
| 4 | 21 | Male | Sandals | Footwear | 90 | Rhode Island | M | Maroon | Spring | 3.5 | Yes | Next Day Air | Yes | Yes | 49 | PayPal | Weekly |
| 5 | 45 | Male | Blouse | Clothing | 49 | Oregon | M | Turquoise | Spring | 2.7 | Yes | Free Shipping | Yes | Yes | 31 | PayPal | Annually |
| 6 | 46 | Male | Sneakers | Footwear | 20 | Wyoming | M | White | Summer | 2.9 | Yes | Standard | Yes | Yes | 14 | Venmo | Weekly |
| 7 | 63 | Male | Shirt | Clothing | 85 | Montana | M | Gray | Fall | 3.2 | Yes | Free Shipping | Yes | Yes | 49 | Cash | Quarterly |
| 8 | 27 | Male | Shorts | Clothing | 34 | Louisiana | L | Charcoal | Winter | 3.2 | Yes | Free Shipping | Yes | Yes | 19 | Credit Card | Weekly |
| 9 | 26 | Male | Coat | Outerwear | 97 | West Virginia | L | Silver | Summer | 2.6 | Yes | Express | Yes | Yes | 8 | Venmo | Annually |
| 10 | 57 | Male | Handbag | Accessories | 31 | Missouri | M | Pink | Spring | 4.8 | Yes | 2-Day Shipping | Yes | Yes | 4 | Cash | Quarterly |
| 11 | 53 | Male | Shoes | Footwear | 34 | Arkansas | L | Purple | Fall | 4.1 | Yes | Store Pickup | Yes | Yes | 26 | Bank Transfer | Bi-Weekly |
| 12 | 30 | Male | Shorts | Clothing | 68 | Hawaii | S | Olive | Winter | 4.9 | Yes | Store Pickup | Yes | Yes | 10 | Bank Transfer | Fortnightly |
| 13 | 61 | Male | Coat | Outerwear | 72 | Delaware | M | Gold | Winter | 4.5 | Yes | Express | Yes | Yes | 37 | Venmo | Fortnightly |
| 14 | 65 | Male | Dress | Clothing | 51 | New Hampshire | M | Violet | Spring | 4.7 | Yes | Express | Yes | Yes | 31 | PayPal | Weekly |
| 15 | 64 | Male | Coat | Outerwear | 53 | New York | L | Teal | Winter | 4.7 | Yes | Free Shipping | Yes | Yes | 34 | Debit Card | Weekly |
| 16 | 64 | Male | Skirt | Clothing | 81 | Rhode Island | M | Teal | Winter | 2.8 | Yes | Store Pickup | Yes | Yes | 8 | PayPal | Monthly |
| 17 | 25 | Male | Sunglasses | Accessories | 36 | Alabama | S | Gray | Spring | 4.1 | Yes | Next Day Air | Yes | Yes | 44 | Debit Card | Bi-Weekly |
| 18 | 53 | Male | Dress | Clothing | 38 | Mississippi | XL | Lavender | Winter | 4.7 | Yes | 2-Day Shipping | Yes | Yes | 36 | Venmo | Quarterly |
| 19 | 52 | Male | Sweater | Clothing | 48 | Montana | S | Black | Summer | 4.6 | Yes | Free Shipping | Yes | Yes | 17 | Cash | Weekly |
| 20 | 66 | Male | Pants | Clothing | 90 | Rhode Island | M | Green | Summer | 3.3 | Yes | Standard | Yes | Yes | 46 | Debit Card | Bi-Weekly |
| 21 | 21 | Male | Pants | Clothing | 51 | Louisiana | M | Black | Winter | 2.8 | Yes | Express | Yes | Yes | 50 | Cash | Every 3 Months |
| 22 | 31 | Male | Pants | Clothing | 62 | North Carolina | M | Charcoal | Winter | 4.1 | Yes | Store Pickup | Yes | Yes | 22 | Debit Card | Quarterly |
| 23 | 56 | Male | Pants | Clothing | 37 | California | M | Peach | Summer | 3.2 | Yes | Store Pickup | Yes | Yes | 32 | Debit Card | Annually |
| 24 | 31 | Male | Pants | Clothing | 88 | Oklahoma | XL | White | Winter | 4.4 | Yes | Express | Yes | Yes | 40 | Credit Card | Weekly |
| 25 | 18 | Male | Jacket | Outerwear | 22 | Florida | M | Green | Fall | 2.9 | Yes | Store Pickup | Yes | Yes | 16 | Debit Card | Weekly |
| 26 | 18 | Male | Hoodie | Clothing | 25 | Texas | M | Silver | Summer | 3.6 | Yes | Express | Yes | Yes | 14 | PayPal | Annually |
| 27 | 38 | Male | Jewelry | Accessories | 20 | Nevada | M | Red | Spring | 3.6 | Yes | Next Day Air | Yes | Yes | 13 | Credit Card | Annually |
| 28 | 56 | Male | Shorts | Clothing | 56 | Kentucky | L | Cyan | Summer | 5 | Yes | Next Day Air | Yes | Yes | 7 | Bank Transfer | Every 3 Months |
| 29 | 54 | Male | Handbag | Accessories | 94 | North Carolina | M | Gray | Fall | 4.4 | Yes | Free Shipping | Yes | Yes | 41 | PayPal | Every 3 Months |
| 30 | 31 | Male | Dress | Clothing | 48 | Wyoming | S | Black | Fall | 4.1 | Yes | Store Pickup | Yes | Yes | 14 | Credit Card | Weekly |
| 31 | 57 | Male | Jewelry | Accessories | 31 | North Carolina | L | Black | Winter | 4.7 | Yes | Standard | Yes | Yes | 16 | Credit Card | Monthly |
| 32 | 33 | Male | Dress | Clothing | 79 | West Virginia | L | Brown | Winter | 4.7 | Yes | Store Pickup | Yes | Yes | 45 | Venmo | Monthly |
| 33 | 36 | Male | Jacket | Outerwear | 67 | Kansas | M | Silver | Summer | 4.9 | Yes | Free Shipping | Yes | Yes | 37 | Venmo | Annually |
| 34 | 54 | Male | Pants | Clothing | 38 | Colorado | L | Green | Summer | 3.3 | Yes | Store Pickup | Yes | Yes | 45 | Cash | Quarterly |

DATASET DESCRIPTION

01

- **Dataset**

Customer Shopping Behavior

02

- **Dataset Size**

Rows: 3,900

Columns: 18

03

- **Goals**

The goal of this project is to gain actionable insights from consumer shopping data that support better marketing strategies, product optimization, and overall business decision making.

DATASET DESCRIPTION

Key Features:

- Customer profile: age, gender, location
- Product detail: item_purchased, category, size, color, season
- Transaction: purchase_amount, shipping_type, payment_method
- Behavior: frequency_of_purchases, previous_purchases, subscription_status
- Marketing: discount_applied, promo_code_used
- Experience: review_rating

EDA USING PYTHON

1. Import Data

```
import pandas as pd

df = pd.read_csv('customer_shopping_behavior.csv')
df.head()
```

| | customer_id | age | gender | item_purchased | category | purchase_amount | location | size | color | season | review_rating | subscription_status |
|---|-------------|-----|--------|----------------|----------|-----------------|---------------|------|-----------|--------|---------------|---------------------|
| 0 | 1 | 55 | Male | Blouse | Clothing | 53 | Kentucky | L | Gray | Winter | 3.1 | Yes |
| 1 | 2 | 19 | Male | Sweater | Clothing | 64 | Maine | L | Maroon | Winter | 3.1 | Yes |
| 2 | 3 | 50 | Male | Jeans | Clothing | 73 | Massachusetts | S | Maroon | Spring | 3.1 | Yes |
| 3 | 4 | 21 | Male | Sandals | Footwear | 90 | Rhode Island | M | Maroon | Spring | 3.5 | Yes |
| 4 | 5 | 45 | Male | Blouse | Clothing | 49 | Oregon | M | Turquoise | Spring | 2.7 | Yes |

| shipping_type | discount_applied | promo_code_used | previous_purchases | payment_method | frequency_of_purchases |
|---------------|------------------|-----------------|--------------------|----------------|------------------------|
| Express | Yes | Yes | 14 | Venmo | Fortnightly |
| Express | Yes | Yes | 2 | Cash | Fortnightly |
| Free Shipping | Yes | Yes | 23 | Credit Card | Weekly |
| Next Day Air | Yes | Yes | 49 | PayPal | Weekly |
| Free Shipping | Yes | Yes | 31 | PayPal | Annually |

This code imports the pandas library, reads a CSV file named 'customer_shopping_behavior.csv' into a DataFrame, and displays the first 5 rows of the dataset using the head() function.

EDA USING PYTHON

2. Initial Exploration

| df.describe(include='all') | | | | | | | | | | | | | |
|----------------------------|-------------|-------------|--------|----------------|----------|-----------------|----------|------|-------|--------|---------------|---------------------|--|
| | customer_id | age | gender | item_purchased | category | purchase_amount | location | size | color | season | review_rating | subscription_status | |
| count | 3900.000000 | 3900.000000 | 3900 | 3900 | 3900 | 3900.000000 | 3900 | 3900 | 3900 | 3900 | 3863.000000 | 3900 | |
| unique | Nan | Nan | 2 | 25 | 4 | Nan | 50 | 4 | 25 | 4 | Nan | 2 | |
| top | Nan | Nan | Male | Blouse | Clothing | Nan | Montana | M | Olive | Spring | Nan | No | |
| freq | Nan | Nan | 2652 | 171 | 1737 | Nan | 96 | 1755 | 177 | 999 | Nan | 2847 | |
| mean | 1950.500000 | 44.068462 | Nan | Nan | Nan | 59.764359 | Nan | Nan | Nan | Nan | 3.750065 | Nan | |
| std | 1125.977353 | 15.207589 | Nan | Nan | Nan | 23.685392 | Nan | Nan | Nan | Nan | 0.716983 | Nan | |
| min | 1.000000 | 18.000000 | Nan | Nan | Nan | 20.000000 | Nan | Nan | Nan | Nan | 2.500000 | Nan | |
| 25% | 975.750000 | 31.000000 | Nan | Nan | Nan | 39.000000 | Nan | Nan | Nan | Nan | 3.100000 | Nan | |
| 50% | 1950.500000 | 44.000000 | Nan | Nan | Nan | 60.000000 | Nan | Nan | Nan | Nan | 3.800000 | Nan | |
| 75% | 2925.250000 | 57.000000 | Nan | Nan | Nan | 81.000000 | Nan | Nan | Nan | Nan | 4.400000 | Nan | |
| max | 3900.000000 | 70.000000 | Nan | Nan | Nan | 100.000000 | Nan | Nan | Nan | Nan | 5.000000 | Nan | |

Checked data structure using df.info() and generated descriptive statistics with df.describe()

| shipping_type | discount_applied | promo_code_used | previous_purchases | payment_method | frequency_of_purchases | |
|---------------|------------------|-----------------|--------------------|----------------|------------------------|--|
| 3900 | 3900 | 3900 | 3900.000000 | 3900 | 3900 | |
| 6 | 2 | 2 | Nan | 6 | 7 | |
| Free Shipping | No | No | Nan | PayPal | Every 3 Months | |
| 675 | 2223 | 2223 | Nan | 677 | 584 | |
| Nan | Nan | Nan | 25.351538 | Nan | Nan | |
| Nan | Nan | Nan | 14.447125 | Nan | Nan | |
| Nan | Nan | Nan | 1.000000 | Nan | Nan | |
| Nan | Nan | Nan | 13.000000 | Nan | Nan | |
| Nan | Nan | Nan | 25.000000 | Nan | Nan | |
| Nan | Nan | Nan | 38.000000 | Nan | Nan | |
| Nan | Nan | Nan | 50.000000 | Nan | Nan | |

EDA USING PYTHON

3. Missing Data Handling

```
df.isnull().sum()  
  
customer_id      0  
age              0  
gender           0  
item_purchased   0  
category          0  
purchase_amount   0  
location          0  
size              0  
color              0  
season             0  
review_rating     37  
subscription_status 0  
shipping_type     0  
discount_applied   0  
promo_code_used    0  
previous_purchases 0  
payment_method     0  
frequency_of_purchases 0  
dtype: int64
```

Identified 37 missing values in the review_rating column that require handling before analysis.



```
df['review_rating'] = (  
    df.groupby('category')['review_rating']  
    .transform(lambda x: x.fillna(x.median()))  
)  
df.isnull().sum()  
✓ 0.0s  
  
customer_id      0  
age              0  
gender           0  
item_purchased   0  
category          0  
purchase_amount   0  
location          0  
size              0  
color              0  
season             0  
review_rating     0  
subscription_status 0  
shipping_type     0  
discount_applied   0  
promo_code_used    0  
previous_purchases 0  
payment_method     0  
frequency_of_purchases 0  
dtype: int64
```

Filled missing Review Rating values with the median rating of each product category using groupby transformation, verified no missing values remain.



EDA USING PYTHON

4. Feature Engineering

```
#create a column age_group
labels = ['Young Adult', 'Adult', 'Middle-aged', 'Senior']
df['age_group'] = pd.qcut(df['age'], q=4, labels=labels)

df[['age', 'age_group']].head(10)
```

| | age | age_group |
|---|-----|-------------|
| 0 | 55 | Middle-aged |
| 1 | 19 | Young Adult |
| 2 | 50 | Middle-aged |
| 3 | 21 | Young Adult |
| 4 | 45 | Middle-aged |
| 5 | 46 | Middle-aged |
| 6 | 63 | Senior |
| 7 | 27 | Young Adult |
| 8 | 26 | Young Adult |
| 9 | 57 | Middle-aged |



Created an `age_group` column by categorizing customer ages into four segments (Young Adult, Adult, Middle-aged, Senior) using quartile-based binning with `pd.qcut()`.

```
# create column purchase_frequency_days
frequency_mapping = {
    'Fortnightly': 14,
    'Weekly': 7,
    'Monthly': 30,
    'Quarterly': 90,
    'Bi-Weekly': 14,
    'Annually': 365,
    'Every 3 Months': 90
}
df['purchase_frequency_days'] = df['frequency_of_purchases'].map(frequency_mapping)

df[['purchase_frequency_days', 'frequency_of_purchases']].head(10)
```

| | purchase_frequency_days | frequency_of_purchases |
|---|-------------------------|------------------------|
| 0 | 14 | Fortnightly |
| 1 | 14 | Fortnightly |
| 2 | 7 | Weekly |
| 3 | 7 | Weekly |
| 4 | 365 | Annually |
| 5 | 7 | Weekly |
| 6 | 90 | Quarterly |
| 7 | 7 | Weekly |
| 8 | 365 | Annually |
| 9 | 90 | Quarterly |



Converted purchase frequency categories into numerical values (days) by mapping each frequency type (Fortnightly, Weekly, Monthly, etc.) to its corresponding day interval using a dictionary.

EDA USING PYTHON

5. Data Consistency Check

```
df[['discount_applied', 'promo_code_used']].head(10)

discount_applied  promo_code_used
0      Yes          Yes
1      Yes          Yes
2      Yes          Yes
3      Yes          Yes
4      Yes          Yes
5      Yes          Yes
6      Yes          Yes
7      Yes          Yes
8      Yes          Yes
9      Yes          Yes

(df['discount_applied'] == df['promo_code_used']).all()

True

df = df.drop('promo_code_used', axis=1)
```

```
print(df.columns)

Index(['customer_id', 'age', 'gender', 'item_purchased', 'category',
       'purchase_amount', 'location', 'size', 'color', 'season',
       'review_rating', 'subscription_status', 'shipping_type',
       'discount_applied', 'previous_purchases', 'payment_method',
       'frequency_of_purchases', 'age_group', 'purchase_frequency_days'],
       dtype='object')
```

Verified that discount_applied and promo_code_used columns contain identical values, then removed the redundant promo_code_used column to eliminate data duplication.

EDA USING PYTHON

6. Database Connection

```
import mysql.connector
import pandas as pd

# 1. Create a connection
conn = mysql.connector.connect(
    host="localhost",
    user="root",
    password="Bismillahsukses01",
    database="project"
)

# 2. check connection
print("Connected:", conn.is_connected())

Connected: True

from sqlalchemy import create_engine
engine = create_engine(
    "mysql+pymysql://root:Bismillahsukses01@localhost/project"
)

df.to_sql(
    name="customer",
    con=engine,
    if_exists="replace", # replace / append
    index=False
)
```

- **MySQL Connection Established** – Connected to the local MySQL database using mysql.connector.
- **Connection Verified** – Ensured the database connection was active using conn.is_connected().
- **SQLAlchemy Engine Created** – Built a SQLAlchemy engine to integrate MySQL with pandas.
- **Data Exported to MySQL** – Loaded the cleaned DataFrame into the customer table using to_sql() with overwrite enabled.

DATA ANALYSIS USING SQL

1. What is the total revenue generated by male vs female customers?

Query

```
SELECT  
    gender,  
    SUM(purchase_amount) AS revenue  
FROM customer  
GROUP BY gender;
```

Result

| | gender | revenue |
|---|--------|---------|
| ▶ | Male | 157890 |
| | Female | 75191 |

- The query groups data by gender.
- It calculates the total purchase amount for each gender using `SUM(purchase_amount)`.
- The result shows:
 - Male customers → revenue 157,890
 - Female customers → revenue 75,191

The query summarizes purchase behavior by gender, and the output reveals that male customers generated more revenue than female customers.

DATA ANALYSIS USING SQL

2. Which customers used a discount but still spent more than the average purchase amount?

Query

```
SELECT  
    customer_id,  
    purchase_amount  
FROM customer  
WHERE discount_applied = 'Yes'  
AND purchase_amount > (  
    SELECT AVG(purchase_amount)  
    FROM customer  
)
```

Result

| | customer_id | purchase_amount |
|---|-------------|-----------------|
| ▶ | 2 | 64 |
| | 3 | 73 |
| | 4 | 90 |
| | 7 | 85 |
| | 9 | 97 |
| | 12 | 68 |
| | 13 | 72 |
| | 16 | 81 |
| | 20 | 90 |
| | 22 | 62 |
| | 24 | 88 |
| | 29 | 94 |

- It selects customers who used a discount (discount_applied = 'Yes').
- Among them, it only shows those whose purchase amount is greater than the overall average purchase amount.
- The output lists their customer_id and purchase_amount.

This condition actually returns 839 rows, meaning many customers with discounts still spent above the average purchase amount.

DATA ANALYSIS USING SQL

3. Which are the top 5 products with the highest average review rating?

Query

```
SELECT  
    item_purchased,  
    ROUND(AVG(review_rating), 2) AS average_product_rating  
FROM customer  
GROUP BY item_purchased  
ORDER BY average_product_rating DESC  
LIMIT 5;
```

- It calculates the average review rating for each product.
- Results are ordered from highest to lowest rating.
- Only the top 5 products are displayed, namely Gloves, Sandals, Boots, Hat, Handbag

Result

| | item_purchased | average_product_rating |
|---|----------------|------------------------|
| ▶ | Gloves | 3.86 |
| | Sandals | 3.84 |
| | Boots | 3.82 |
| | Hat | 3.8 |
| | Handbag | 3.78 |

DATA ANALYSIS USING SQL

4. Compare the average Purchase Amounts between Standard and Express Shipping Query

SELECT

```
shipping_type,  
ROUND(AVG(purchase_amount), 2) AS avg_purchase  
FROM customer  
WHERE shipping_type IN ('Standard', 'Express')  
GROUP BY shipping_type;
```

Result

| | shipping_type | avg_purchase |
|---|---------------|--------------|
| ▶ | Express | 60.48 |
| | Standard | 58.46 |

- It calculates the average purchase amount for each shipping type (Standard vs Express).
- The results show:
 - Express Shipping → 60.48
 - Standard Shipping → 58.46

Customers using Express Shipping spend slightly more on average compared to those using Standard Shipping.

DATA ANALYSIS USING SQL

5. Do subscribed customers spend more? Compare average spend and total revenue between subscribers and non-subscribers

Query

```
SELECT
    subscription_status,
    COUNT(customer_id) AS total_customer,
    ROUND(SUM(purchase_amount), 2) AS total_revenue,
    ROUND(AVG(purchase_amount), 2) AS avg_spend
FROM customer
GROUP BY subscription_status
ORDER BY total_revenue DESC;
```

Result

| | subscription_status | total_customer | total_revenue | avg_spend |
|---|---------------------|----------------|---------------|-----------|
| ▶ | No | 2847 | 170436 | 59.87 |
| | Yes | 1053 | 62645 | 59.49 |

- It compares subscribed vs non-subscribed customers.

- Average spend:

Subscribers → 59.49

Non-subscribers → 59.87

- Total revenue:

Subscribers → 62,645

Non-subscribers → 170,436

Subscribed customers do not spend more. Both average spend and total revenue are higher for non-subscribers.

DATA ANALYSIS USING SQL

6. Which 5 products have the highest percentage of purchases with discounts applied?

Query

```
SELECT
    item_purchased,
    ROUND(
        SUM(CASE WHEN discount_applied = 'Yes' THEN 1 ELSE 0 END)
        / COUNT(*) * 100, 2
    ) AS discount_rate
FROM customer
GROUP BY item_purchased
ORDER BY discount_rate DESC
LIMIT 5;
```

Result

| | item_purchased | discount_rate |
|---|----------------|---------------|
| ▶ | Hat | 50.00 |
| | Sneakers | 49.66 |
| | Coat | 49.07 |
| | Sweater | 48.17 |
| | Pants | 47.37 |

- It calculates the percentage of purchases with discounts applied for each product.
- Results are ordered from highest discount rate to lowest.
- The top 5 products are:

Hat → 50.00%

Sneakers → 49.66%

Coat → 49.07%

Sweater → 48.17%

Pants → 47.37%

These are the items most frequently bought with discounts.

DATA ANALYSIS USING SQL

7. How many customers fall into each segment (New, Returning, and Loyal) based on their number of previous purchases?

Query

```
WITH customer_type AS (
    SELECT
        customer_id,
        previous_purchases,
        CASE
            WHEN previous_purchases = 1 THEN 'New'
            WHEN previous_purchases BETWEEN 2 AND 10 THEN 'Returning'
            ELSE 'Loyal'
        END AS customer_segment
    FROM customer
)
SELECT
    customer_segment,
    COUNT(*) AS number_of_customers
FROM customer_type
GROUP BY customer_segment;
```

Result

| | customer_segment | number_of_customers |
|---|------------------|---------------------|
| ▶ | Loyal | 3116 |
| | Returning | 701 |
| | New | 83 |

CTE (Common Table Expression) - Creates a temporary result set named customer_type to organize customer segmentation logic

- Customers are segmented by their previous purchases:
 - New → 1 purchase
 - Returning → 2-10 purchases
 - Loyal → more than 10 purchases
 - The result shows:
 - Loyal → 3,116 customers
 - Returning → 701 customers
 - New → 83 customers
- Most customers fall into the Loyal segment.

DATA ANALYSIS USING SQL

8. What are the top 3 most purchased products within each category?

Query

```
WITH item_counts AS (
    SELECT
        category,
        item_purchased,
        COUNT(*) AS total_orders,
        ROW_NUMBER() OVER (
            PARTITION BY category
            ORDER BY COUNT(*) DESC
        ) AS item_rank
    FROM customer
    GROUP BY category, item_purchased
)

SELECT
    category,
    item_purchased,
    total_orders
FROM item_counts
WHERE item_rank <= 3;
```

Result

| | category | item_purchased | total_orders |
|---|-------------|----------------|--------------|
| ▶ | Accessories | Jewelry | 171 |
| | Accessories | Sunglasses | 161 |
| | Accessories | Belt | 161 |
| | Clothing | Blouse | 171 |
| | Clothing | Pants | 171 |
| | Clothing | Shirt | 169 |
| | Footwear | Sandals | 160 |
| | Footwear | Shoes | 150 |
| | Footwear | Sneakers | 145 |
| | Outerwear | Jacket | 163 |
| | Outerwear | Coat | 161 |

- It counts total orders for each product within its category.
- Then ranks items by order volume using ROW_NUMBER().
- Finally, it selects the top 3 items per category.

Output:

- Accessories → Jewelry (171), Sunglasses (161), Belt (161)
- Clothing → Blouse (171), Pants (171), Shirt (169)
- Footwear → Sandals (160), Shoes (150), Sneakers (145)
- Outerwear → Jacket (163), Coat (161)

These are the top 3 most purchased products in each category.

DATA ANALYSIS USING SQL

9. Are repeat buyers (customers with more than five previous purchases) more likely to be subscribed?

Query

```
SELECT  
    subscription_status,  
    COUNT(customer_id) AS repeat_buyers  
FROM customer  
WHERE previous_purchases > 5  
GROUP BY subscription_status;
```

- It counts repeat buyers (customers with more than 5 previous purchases).
- Results:
 - Subscribed → 958 repeat buyers
 - Not subscribed → 2,518 repeat buyers

Result

| | subscription_status | repeat_buyers |
|---|---------------------|---------------|
| ▶ | Yes | 958 |
| | No | 2518 |

Repeat buyers are more likely to be non-subscribers, since their count is much higher compared to subscribed customers.

DATA ANALYSIS USING SQL

10. What is the revenue contribution of each age group?

Query

```
SELECT  
    age_group,  
    SUM(purchase_amount) AS total_revenue  
FROM customer  
GROUP BY age_group  
ORDER BY total_revenue DESC;
```

Result

| | age_group | total_revenue |
|---|-------------|---------------|
| ▶ | Young Adult | 62143 |
| | Middle-aged | 59197 |
| | Adult | 55978 |
| | Senior | 55763 |

- It calculates the total revenue contributed by each age group.

- Results:

Young Adult → 62,143

Middle-aged → 59,197

Adult → 55,978

Senior → 55,763

Young Adults contribute the highest revenue, followed closely by Middle-aged, while Adults and Seniors contribute slightly less.

DASHBOARD

Customer Behavior Dashboard

| | |
|--------------------------------------|------|
| Subscription Status | |
| No | Yes |
| Gender | |
| Female | Male |
| Category | |
| Accessories | |
| Clothing | |
| Footwear | |
| Outerwear | |
| Shipping Type | |
| <input type="radio"/> 2-Day Shipping | |
| <input type="radio"/> Express | |
| <input type="radio"/> Free Shipping | |
| <input type="radio"/> Next Day Air | |
| <input type="radio"/> Standard | |
| <input type="radio"/> Store Pickup | |

3.9K

Number of Customer

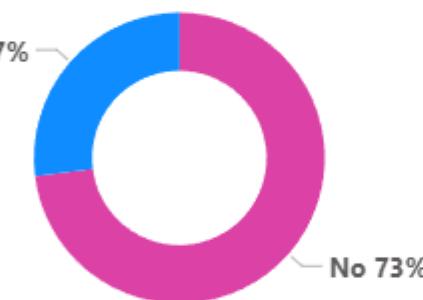
\$59.76

Average Purchase Amount

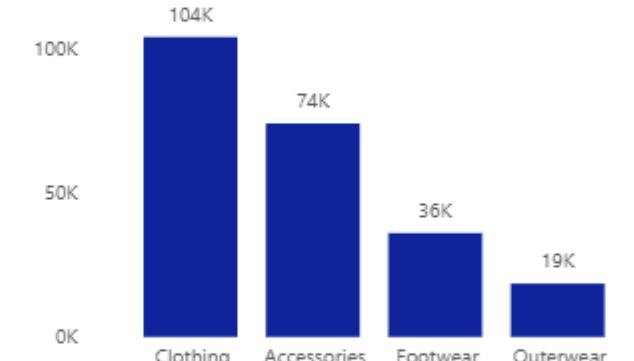
3.75

Average Review Rating

% of Customers by Subscription Status



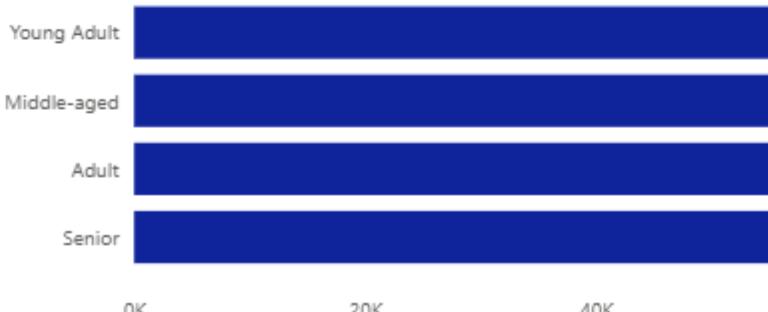
Revenue by Category



Sales by Category



Revenue by Age Group



Sales by Age Group



CONCLUSION

- **Revenue Drivers**

Male customers, Young Adults, and Middle-aged groups contribute the largest share of revenue.

- **Customer Segments**

Most customers are Loyal, while New and Returning segments are smaller but still important.

- **Subscription Impact**

Non-subscribers generate more revenue and include more repeat buyers than subscribers.

- **Product & Discount Trends**

Certain products rely heavily on discounts, while top-rated items stand out in quality and appeal.

SUGGESTIONS

- **Drive Subscription Growth**

Offer exclusive perks and tailored benefits to convert repeat buyers into subscribers.

- **Strengthen Loyalty Programs**

Reward and recognize repeat buyers to reinforce engagement and expand the Loyal segment.

- **Optimize Discount Policies**

Refine discount strategies to maintain sales momentum while protecting profit margins.

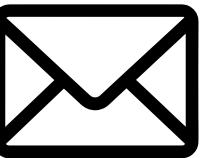
- **Elevate Product & Marketing Focus**

Highlight top-rated and best-selling products, and target high-revenue age groups plus Express shipping users.

LET'S CONNECT



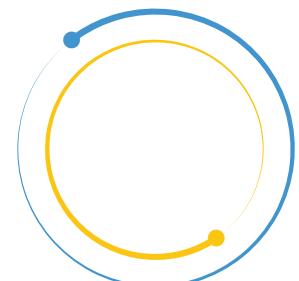
083109376438



hernandiazanua@gmail.com



Hernandia Zanua Leras



THANK YOU