

CUSTOMER SHOPPING TREND

SQL QUERY RESULT

Dataset Link : <https://www.kaggle.com/datasets/iamsouravbanerjee/customer-shopping-trends-dataset>

This dataset encompasses various features related to customer shopping preferences, gathering essential information for businesses seeking to enhance their understanding of their customer base. The features include customer age, gender, purchase amount, preferred payment methods, frequency of purchases, and feedback ratings. Additionally, data on the type of items purchased, shopping frequency, preferred shopping seasons, and interactions with promotional offers is included. With a collection of 3900 records

Overview of Table

Customer ID - Unique identifier for each customer

Age - Age of the customer

Gender - Gender of the customer (Male/Female)

Item Purchased - The item purchased by the customer

Category - Category of the item purchased

Purchase Amount (USD) - The amount of the purchase in USD

Location - Location where the purchase was made

Size - Size of the purchased item

Color - Color of the purchased item

Season - Season during which the purchase was made

Review Rating - Rating given by the customer for the purchased item

Subscription Status - Indicates if the customer has a subscription (Yes/No)

Shipping Type - Type of shipping chosen by the customer

Discount Applied - Indicates if a discount was applied to the purchase (Yes/No)

Promo Code Used - Indicates if a promo code was used for the purchase (Yes/No)

Previous Purchases - The total count of transactions concluded by the customer at the store, excluding the ongoing transaction

Payment Method - Customer's most preferred payment method

Frequency of Purchases - Frequency at which the customer makes purchases (e.g., Weekly,] Fortnightly, Monthly)

Customer Demographics

Average customer age.

```
SELECT ROUND(AVG(Age), 2) AS Customer_Average_Age FROM Shopping;
```

Result Grid		Filter Rows:
	Customer_Average_Age	
▶	44.07	

Gender distribution (percentage of male vs. female customers).

```
SELECT Gender, CONCAT(ROUND(COUNT(*) * 100.0 / (SELECT COUNT(*) FROM Shopping)), '%') AS  
Percentage  
FROM Shopping  
GROUP BY Gender;
```

Result Grid			Filter Rows
	Gender	Percentage	
▶	Male	68%	
	Female	32%	

Customer Loyalty

Percentage of repeat customers (based on Previous Purchases > 1)

```
SELECT  
CASE  
    WHEN Previous_Purchases > 1 THEN 'Repeat Customers'  
    ELSE 'First-Time Customers'  
END AS Customer_Type,  
CONCAT(ROUND(COUNT(*) * 100.0 / (SELECT COUNT(*) FROM Shopping), 2), '%') AS Percentage  
FROM Shopping  
GROUP BY Customer_Type;
```

Result Grid			Filter Rows:
	Customer_Type	Percentage	
▶	Repeat Customers	97.87%	
	First-Time Customers	2.13%	

Average number of previous purchases per customer

```
SELECT ROUND(AVG(Previous_Purchases), 2) AS  
Average_Previous_Purchases  
FROM Shopping;
```

Result Grid		Filter Rows:
	Average_Previous_Purchases	
▶	25.35	

Subscription Status

Percentage of customers subscribed vs. not subscribed

```
SELECT Subscription_Status, CONCAT(ROUND(COUNT(*) * 100.0 / (SELECT COUNT(*) FROM Shopping)), '%') AS Status_Percentage
FROM Shopping
GROUP BY Subscription_Status;
```

	Subscription_Status	Status_Percentage
▶	Yes	27%
	No	73%

Sales Performance

Total Sales

Total revenue (Purchase Amount (USD))

```
SELECT SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))) AS Total_Revenue
FROM Shopping;
```

Result Grid	
	Total_Revenue
▶	233081

Average order value

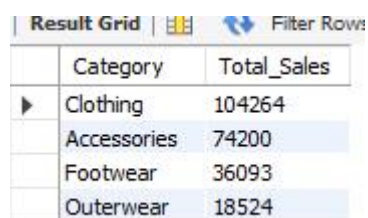
```
SELECT SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))) AS Total_Sales,
COUNT(*) AS Total_Orders,
ROUND(SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))) / COUNT(*), 2) AS
Average_Order_Value
FROM Shopping;
```

Result Grid			
Filter Rows: <input type="text"/>			
	Total_Sales	Total_Orders	Average_Order_Value
▶	233081	3900	59.76

Sales by Category

Revenue contribution by category

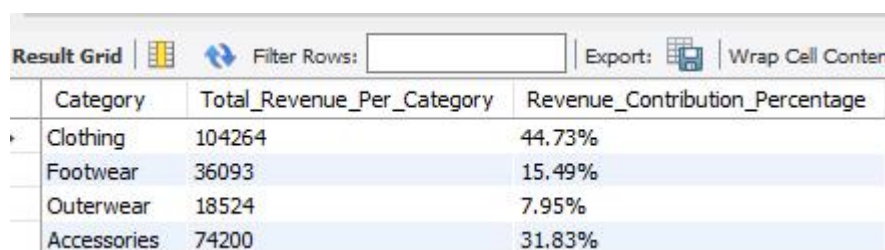
```
SELECT Category, SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))) AS Total_Sales
FROM Shopping
GROUP BY Category
ORDER BY Total_Sales DESC;
```



The screenshot shows a database interface with a 'Result Grid' tab. It contains a table with two columns: 'Category' and 'Total_Sales'. The data is sorted in descending order of total sales.

Category	Total_Sales
Clothing	104264
Accessories	74200
Footwear	36093
Outerwear	18524

```
SELECT Category,
ROUND(SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10)))) AS Total_Revenue_Per_Category,
CONCAT(ROUND(SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))) * 100.0 /
(SELECT SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))) FROM Shopping), 2), '%') AS
Revenue_Contribution_Percentage
FROM Shopping
GROUP BY Category;
```



The screenshot shows a database interface with a 'Result Grid' tab. It contains a table with three columns: 'Category', 'Total_Revenue_Per_Category', and 'Revenue_Contribution_Percentage'. The data is sorted in descending order of total revenue per category.

Category	Total_Revenue_Per_Category	Revenue_Contribution_Percentage
Clothing	104264	44.73%
Footwear	36093	15.49%
Outerwear	18524	7.95%
Accessories	74200	31.83%

Top-performing categories based on revenue.

```
SELECT Category, ROUND(SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))), 2) AS
Total_Revenue_Per_Category
FROM Shopping
GROUP BY Category
ORDER BY Total_Revenue_Per_Category DESC
LIMIT 10; -- You can adjust the number to show more or fewer top categories
```

Category	Total_Revenue_Per_Category
Clothing	104264
Accessories	74200
Footwear	36093
Outerwear	18524

Regional Performance

Sales distribution by location

```
SELECT Location, SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))) AS Total_Sales
FROM Shopping
GROUP BY Location
ORDER BY Total_Sales DESC;
```

Location	Total_Sales	Location	Total_Sales	Location	Total_Sales
Montana	5784	Pennsylvania	4926	Missouri	4691
Illinois	5617	Mississippi	4883	Indiana	4655
California	5605	Alaska	4867	Ohio	4649
Idaho	5587	Vermont	4860	Georgia	4645
Nevada	5514	Louisiana	4848	Washington	4623
Alabama	5261	Virginia	4842	Michigan	4533
New York	5257	Arkansas	4828	Utah	4443
North Dakota	5220	Maryland	4795	South Carolina	4439
West Virginia	5174	Tennessee	4772	Kentucky	4402
Nebraska	5172	Delaware	4758	Maine	4388
New Mexico	5014	North Carolina	4742	Massachusetts	4384
Minnesota	4977	Texas	4712	Oklahoma	4376

```
SELECT Location,
ROUND(SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))), 2) AS
Total_Revenue_Per_Location,
CONCAT(ROUND(SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))) * 100.0 /
(SELECT SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))) FROM Shopping), 2), '%') AS
Revenue_Distribution_Percentage
FROM Shopping
GROUP BY Location
ORDER BY Total_Revenue_Per_Location DESC;
```

Location	Total_Revenue_Per_Location	Revenue_Distribution_Percentage	Location	Total_Revenue_Per_Location	Revenue_Distribution_Percentage	Location	Total_Revenue_Per_Location	Revenue_Distribution_Percentage
Montana	5784	2.48%	Pennsylvania	4926	2.11%	Missouri	4691	2.01%
Illinois	5617	2.41%	Mississippi	4883	2.09%	Indiana	4655	2.00%
California	5605	2.40%	Alaska	4867	2.09%	Ohio	4649	1.99%
Idaho	5587	2.40%	Vermont	4860	2.09%	Georgia	4645	1.99%
Nevada	5514	2.37%	Louisiana	4848	2.08%	Washington	4623	1.98%
Alabama	5261	2.26%	Virginia	4842	2.08%	Michigan	4533	1.94%
New York	5257	2.26%	Arkansas	4828	2.07%	Utah	4443	1.91%
North Dakota	5220	2.24%	Maryland	4795	2.06%	South Carolina	4439	1.90%
West Virginia	5174	2.22%	Tennessee	4772	2.05%	Kentucky	4402	1.89%
Nebraska	5172	2.22%	Delaware	4758	2.04%	Maine	4388	1.88%
New Mexico	5014	2.15%	North Carolina	4742	2.03%	Massachusetts	4384	1.88%
Minnesota	4977	2.14%	Texas	4712	2.02%	Oklahoma	4376	1.88%

Regions with the highest/lowest revenue.

```
(SELECT Location, SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))) AS Total_Sales
FROM Shopping
GROUP BY Location
ORDER BY Total_Sales DESC
LIMIT 1)
```

UNION ALL

```
(SELECT Location, SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))) AS Total_Sales
FROM Shopping
GROUP BY Location
ORDER BY Total_Sales ASC
LIMIT 1);
```

Location	Total_Sales
Montana	5784
Kansas	3437

Seasonal Trends

Revenue distribution by season.

```
SELECT Season,
SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))) AS Revenue_Per_Season,
CONCAT(ROUND(SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))) * 100.0 /
(SELECT SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))) FROM Shopping), 2), '%') AS
Revenue_Distribution_Percentage
FROM Shopping
GROUP BY Season
```

ORDER BY Revenue_Per_Season, Revenue_Distribution_Percentage DESC;

Season	Revenue_Per_Season	Revenue_Distribution_Percentage
Summer	55777	23.93%
Winter	58607	25.14%
Spring	58679	25.18%
Fall	60018	25.75%

Product Trends

Most purchased items

```
SELECT Item_Purchased, COUNT(*) AS Purchase_Count
FROM Shopping
GROUP BY Item_Purchased
ORDER BY Purchase_Count DESC
LIMIT 10;
```

Item_Purchased	Purchase_Count
Blouse	171
Pants	171
Jewelry	171
Shirt	169
Dress	166
Sweater	164
Jacket	163
Coat	161
Sunglasses	161
Belt	161

Items with the highest revenue contribution.

```
SELECT Item_Purchased,
SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))) AS Purchase_Revenue,
CONCAT(ROUND(SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))) * 100 /
(SELECT(SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))) FROM Shopping), 2), '%') AS
Revenue_Distribution_Percentage
FROM Shopping
GROUP BY Item_Purchased
ORDER BY Purchase_Revenue DESC
LIMIT 10;
```

Item_Purchased	Purchase_Revenue	Revenue_Distribution_Percentage
Blouse	10410	4.47%
Shirt	10332	4.43%
Dress	10320	4.43%
Pants	10090	4.33%
Jewelry	10010	4.29%
Sunglasses	9649	4.14%
Belt	9635	4.13%
Scarf	9561	4.10%
Sweater	9462	4.06%
Shorts	9433	4.05%

Product Attributes

Sales by item size and color

```
SELECT Size, Color, COUNT(*) AS Count_Item,
SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))) AS Revenue
FROM Shopping
GROUP BY Size, Color
ORDER BY Revenue DESC
LIMIT 10;
```

Size	Color	Count_Item	Revenue
M	Violet	87	5449
M	Olive	86	5121
M	Gray	78	5090
M	Charcoal	77	4661
M	White	71	4585
M	Orange	74	4513
M	Green	70	4466
M	Silver	79	4417
M	Maroon	76	4295
M	Peach	70	4249

Customer preferences for item attributes.

```
SELECT Category, COUNT(*) AS Customer_Preference,
CONCAT(ROUND(COUNT(*) * 100.0 / (SELECT COUNT(*) FROM Shopping), 2), '%') AS
Preference_Percentage
FROM Shopping
GROUP BY Category
ORDER BY Customer_Preference DESC;
```


Category	Customer_Preference	Preference_Percentage
Clothing	1737	44.54%
Accessories	1240	31.79%
Footwear	599	15.36%
Outerwear	324	8.31%

Discount and Promo Code Usage

Percentage of transactions with discounts applied.

```
SELECT Discount_Applied AS Discount,
CONCAT(ROUND(COUNT(*) * 100.0 / (SELECT COUNT(*) FROM Shopping), 2), '%') AS
Discount_Percentage
FROM Shopping
GROUP BY Discount
ORDER BY Discount_Percentage DESC;
```

Discount	Discount_Percentage
No	57.00%
Yes	43.00%

Percentage of transactions using promo codes.

```
SELECT Promo_Code_Used AS Promo_Code,
CONCAT(ROUND(COUNT(*) * 100.0 / (SELECT COUNT(*) FROM Shopping), 2), '%') AS
Promo_Code_Percentage
FROM Shopping
GROUP BY Promo_Code
ORDER BY Promo_Code_Percentage DESC;
```

Promo_Code	Promo_Code_Percentage
No	57.00%
Yes	43.00%

Customer Behavior

Distribution of payment methods used

```
SELECT Preferred_Payment_Method AS Payment_Method,  
SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))) AS Revenue,  
CONCAT(ROUND(COUNT(*) * 100.0 / (SELECT COUNT(*) FROM Shopping), 2), '%') AS  
Payment_Method_Percentage  
FROM Shopping  
GROUP BY Preferred_Payment_Method  
ORDER BY Revenue DESC;
```

Payment_Method	Revenue	Payment_Method_Percentage
Credit Card	40310	17.21%
PayPal	40109	17.36%
Cash	40002	17.18%
Debit Card	38742	16.31%
Venmo	37374	16.26%
Bank Transfer	36544	15.69%

Purchase Frequency

Breakdown of customers by Frequency of Purchases

```
SELECT Frequency_of_Purchases AS Frequency_Purchase, COUNT(*) AS Purchase_Count,  
CONCAT(ROUND(COUNT(*) * 100 / (SELECT COUNT(*) FROM Shopping), 2), '%') AS  
Count_Percentage  
FROM Shopping  
GROUP BY Frequency_Purchase  
ORDER BY Purchase_Count DESC;
```

Frequency_Purchase	Purchase_Count	Count_Percentage
Every 3 Months	584	14.97%
Annually	572	14.67%
Quarterly	563	14.44%
Monthly	553	14.18%
Bi-Weekly	547	14.03%
Fortnightly	542	13.90%
Weekly	539	13.82%

Percentage of customers choosing each Shipping Type

```
SELECT Shipping_Type AS Ship_Time, COUNT(*) AS Ship_Time_Count,
CONCAT(ROUND(COUNT(*) * 100 / (SELECT COUNT(*) FROM Shopping), 2), '%') AS
Count_Percentage
FROM Shopping
GROUP BY Ship_Time
ORDER BY Ship_Time_Count DESC;
```

Ship_Time	Ship_Time_Count	Count_Percentage
Free Shipping	675	17.31%
Standard	654	16.77%
Store Pickup	650	16.67%
Next Day Air	648	16.62%
Express	646	16.56%
2-Day Shipping	627	16.08%

Operational Efficiency

Average Review Rating by category and item.

```
SELECT Category, Item_Purchased, ROUND(AVG(Review_Rating), 2) AVG_Rating
FROM Shopping
GROUP BY Category, Item_Purchased
ORDER BY Category ASC, AVG_Rating DESC;
```

Category	Item_Purchased	AVG_Rating	Category	Item_Purchased	AVG_Rating
Accessories	Gloves	3.86	Clothing	Dress	3.75
Accessories	Hat	3.81	Clothing	Socks	3.75
Accessories	Handbag	3.78	Clothing	Pants	3.72
Accessories	Jewelry	3.76	Clothing	Hoodie	3.72
Accessories	Belt	3.76	Clothing	Shorts	3.71
Accessories	Backpack	3.75	Clothing	Blouse	3.68
Accessories	Sunglasses	3.74	Clothing	Jeans	3.65
Accessories	Scarf	3.7	Clothing	Shirt	3.63
Clothing	Skirt	3.78	Footwear	Sandals	3.84
Clothing	T-shirt	3.78	Footwear	Boots	3.81
Clothing	Sweater	3.77	Footwear	Sneakers	3.76
Clothing	Dress	3.75	Footwear	Shoes	3.75

Outerwear	Jacket	3.76
Outerwear	Coat	3.73

Items with the highest and lowest average ratings.

```
(SELECT Category, Item_Purchased, ROUND(AVG(Review_Rating), 2) AVG_Rating
FROM Shopping
GROUP BY Category, Item_Purchased
ORDER BY AVG_Rating DESC
LIMIT 1)
```

UNION

```
(SELECT Category, Item_Purchased, ROUND(AVG(Review_Rating), 2) AVG_Rating
FROM Shopping
GROUP BY Category, Item_Purchased
ORDER BY AVG_Rating ASC
LIMIT 1);
```

Category	Item_Purchased	AVG_Rating
Accessories	Gloves	3.86
Clothing	Shirt	3.63

Discount Impact

Revenue generated from discounted vs. non-discounted sales.

```
SELECT
CASE
    WHEN Discount_Applied > 0 THEN 'Discounted Sales'
    ELSE 'Non-Discounted Sales'
END AS Sales_Type,
SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))) AS Total_Revenue
FROM Shopping
```

GROUP BY Sales_Type;

Sales_Type	Total_Revenue
Non-Discounted Sales	233081

Promo Code Effectiveness

Average order value for transactions using promo codes vs. those without.

```
SELECT
  CASE
    WHEN Promo_Code_Used IS NOT NULL THEN 'With Promo Code'
    ELSE 'Without Promo Code'
  END AS Promo_Code_Usage,
  SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))) AS Total_Revenue
FROM Shopping
GROUP BY Promo_Code_Usage;
```

Promo_Code_Usage	Total_Revenue
With Promo Code	233081

Lifetime Value (LTV)

Average revenue per customer

```
SELECT ROUND(SUM(CAST(`Purchase_Amount_(USD)` AS DECIMAL(10))) /
  COUNT(DISTINCT(Customer_ID)), 2) AS Per_Customer_Avg_Revenue
FROM Shopping;
```

Per_Customer_Avg_Revenue
59.76

Churn Rate

Identify inactive customers

```
SELECT COUNT(Customer_ID) AS Inactive_Customer_Count  
FROM Shopping  
WHERE Previous_Purchases <= 1;
```

Inactive_Customer_Count

83
