

SQL SALES ANALYSIS



Overview

- Introduction
- Project Objectives
- Basic Queries
- Conditional Queries
- Aggregate Queries
- Group and Filter Queries
- Advance Queries
- Windows Function
- Conclusions



2

Introduction

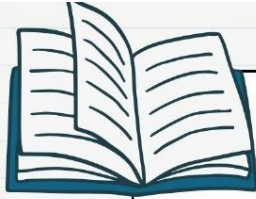
Customer Sale Data:

Here a table consist of 19 columns and 3900 rows . Here all 19 columns are Age,Gender,Item_Purchased,Category,Purchase_Amount,Location,Size,Color,Season,Review_Rating, Subscription_Status, Payment_Method, Shipping_Type,Discount_Applied,Promo_Code, Previous_Purchases,Preferred_Payment_Method, Frequency_of_Purchases

Details:

The name of DataBase is shop and Table name is shopping
Example In SQL Workbench : shop.shopping





Project Objectives



"The main objective of this project is to analyze customer sales data to identify key trends in purchase behavior, seasonal performance, category revenue contribution, and promotional effectiveness. The findings aim to support strategic decisions in marketing, inventory management, and customer engagement."



BASIC QUERIES

Basic Queries

1) Retrieve all unique items purchased:

```
-- Distinct and unique Item purchased by customer  
Select Distinct Item_Purchased from shop.shopping;
```

Output:

Item_Purchased
Blouse
Sweater
Jeans
Sandals
Sneakers
Shirt
Shorts
Coat
Handbag
Shoes
Dress
Skirt



Basic Queries

2) Find the total sales amount (USD):

```
-- Find the total sales amount (USD)  
select sum(Purchase_Amount) as Total_Amount From shop.shopping;
```

Output:

Total_Amount
233081

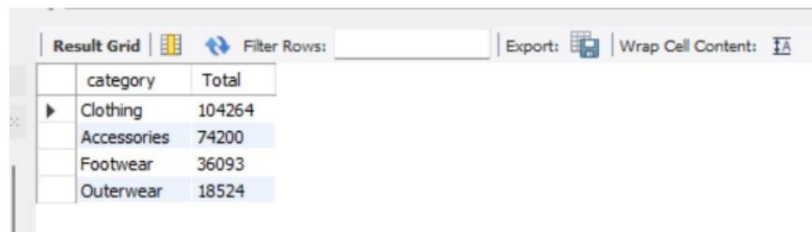


Basic Queries

3) List all unique categories of items purchased and total amount paid by customer for each categories

```
-- List all unique categories of items purchased and total amount paid by customer for each categories :  
Select distinct category, sum(Purchase_Amount) as Total from shop.shopping group by Category order by Total desc;
```

Output:



The screenshot shows a database query result grid with a toolbar at the top. The toolbar includes a 'Result Grid' icon, a 'Filter Rows' input field, an 'Export' button, and a 'Wrap Cell Content' button. The table below has two columns: 'category' and 'Total'. The data is sorted in descending order of total amount.

category	Total
Clothing	104264
Accessories	74200
Footwear	36093
Outerwear	18524



CONDITIONAL QUERIES

Conditional Queries

1) Find total purchases made by female customers:

```
-- Find total purchases made by female customers
Select gender,sum(purchase_amount) as Total_Amount_By_female from shop.shopping where gender="Female";
```

Output:

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
gender	Total_Amount_By_female			
Female	75191			



Conditional Queries

2) Identify purchases where the discount was applied:

```
5 -- Identify purchases where the discount was applied.
6 • Select Count(Purchase_Amount) As Total_Person_Applied_Discount, Discount_Applied from shop.shopping where Discount_Applied="Yes" group by Discount_Applied;
```

Output:

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
Total_Person_Applied_Discount	Discount_Applied			
1677	Yes			



Conditional Queries

3) Retrieve all records for customers who used promo codes:

```
-- Retrieve all records for customers who used promo codes.  
Select Count(Customer_ID) as Total_Person_Applied_Promo, Promo_Code from shop.shopping where Promo_Code="Yes" group by Promo_Code;
```

Output:

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
Total_Person_Applied_Promo	Promo_Code			
1677	Yes			



Conditional Queries

4) Find all purchases made in the "Winter" season

```
9 -- Find all purchases made in the "Winter" season.  
10 • Select Count(Customer_ID) as Total_Winter_Purchased, Season from shop.shopping where Season="Winter" group by Season;  
11
```

Output:

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
Total_Winter_Purchased	Season			
971	Winter			



AGGREGATE QUERIES

Aggregate Queries

1) Calculate the average purchase amount for each payment method.

```
2 -- Calculate the average purchase amount for each payment method.
3 • Select distinct Ceil(avg(Purchase_Amount)) As Average_Total_USD,Preferred_Payment_Method From shop.shopping group by Preferred_Payment_Method;
```

Output:

	Average_Total_USD	Preferred_Payment_Method
▶	59	Venmo
	60	Cash
	61	Credit Card
	60	PayPal
	60	Bank Transfer
	61	Debit Card



Aggregate Queries

2) Find the highest-rated item based on the review rating.

```
5 -- Find the highest-rated item based on the review rating.
6 • Select distinct Item_Purchased,Ceil(avg(Review_Rating)) as Avrage_rating from shop.shopping group by Item_Purchased limit 5;
7
```

Output:

	Item_Purchased	Avrage_rating
▶	Blouse	4
	Sweater	4
	Jeans	4
	Sandals	4
	Sneakers	4



Aggregate Queries

3) Count the number of purchases in each category.

-- Count the number of purchases in each category.

```
• Select Distinct Item_Purchased, Sum(Purchase_Amount) as Total_Amount from shop.shopping group by Item_Purchased order by Total_Amount desc limit 10
```

Output:

Item_Purchased	Total_Amount
Blouse	10410
Shirt	10332
Dress	10320
Pants	10090
Jewelry	10010
Sunglasses	9649
Belt	9635
Scarf	9561
Sweater	9462
Shorts	9433



GROUP AND FILTER QUERIES

Group and Filter Queries

1) Find the total sales (USD) for each location.

```
3 -- Find the total sales (USD) for each location.
4 • Select distinct Location , Sum(Purchase_Amount) as Total_Amount from shop.shopping group by Location order by Total_amount desc limit 10;
5
```

Output:

Location	Total_Amount
Montana	5784
Illinois	5617
California	5605
Idaho	5587
Nevada	5514
Alabama	5261
New York	5257
North Dakota	5220
West Virginia	5174
Nebraska	5172



Group and Filter Queries

2) List the top 3 most frequently purchased items.

```
6 -- List the top 3 most frequently purchased items in weekly.
7 • Select Distinct Item_Purchased,count(Frequency_of_Purchases) As Total from shop.shopping where Frequency_of_Purchases="weekly"
8 group by Item_Purchased order by Total desc limit 3;
```

Output:

Item_Purchased	Total
Sweater	31
Skirt	27
Coat	26



Group and Filter Queries

3) Find categories with total sales exceeding \$10,000

```
9
10 -- Find categories with total sales exceeding $10,000.
11 • select distinct Item_Purchased, Sum(Purchase_Amount) AS Amount_Over_5000 From shop.shopping group by Item_Purchased having Amount_Over_5000>10000
12 order by Amount_Over_5000 Desc limit 5;
```

Output:

Item_Purchased	Amount_Over_5000
Blouse	10410
Shirt	10332
Dress	10320
Pants	10090
Jewelry	10010



Group and Filter Queries

4) Which product categories perform better in each season?

```
-- Which product categories perform better in each season?
SELECT Season, Category, SUM(Purchase_Amount) AS Total_Sales FROM shop.shopping GROUP BY Season, Category
ORDER BY Season, Total_Sales DESC;
```

Output:

Season	Category	Total_Sales
Spring	Accessories	17007
Spring	Footwear	9555
Spring	Outerwear	4425
Summer	Clothing	23078
Summer	Accessories	19028
Summer	Footwear	9393
Summer	Outerwear	4278
Winter	Clothing	27274
Winter	Accessories	18291
Winter	Footwear	8480
Winter	Outerwear	4562



ADVANCE QUERIES

Advance Queries:

1) Determine Total Promo Code Usage and Associated Revenue.

```
2 -- Determine Total Promo Code Usage and Associated Revenue
3 • SELECT Promo_Code, COUNT(*) AS Total_Uses, SUM(Purchase_Amount) AS Total_Revenue FROM shop.shopping GROUP BY Promo_Code;
```

Output:

	Promo_Code	Total_Uses	Total_Revenue
▶	Yes	1677	99411
	No	2223	133670



Advance Queries:

2) Analyze Subscription Customers' Revenue Contribution by Payment Method.

```
6 -- Analyze Subscription Customers' Revenue Contribution by Payment Method
7 • Select Subscription_Status, count(*) as Total_Subscription , Preferred_Payment_Method , Sum(Purchase_amount) as Total_amount from shop.shopping
8 group by Subscription_Status, Preferred_Payment_Method Order by Total_amount desc;
```

Output:

	Subscription_Status	Total_Subscription	Preferred_Payment_Method	Total_amount
▶	No	492	Credit Card	29771
	No	497	Cash	29450
	No	497	PayPal	29352
	No	455	Bank Transfer	27459
	No	460	Venmo	27367
	No	446	Debit Card	27037
	Yes	190	Debit Card	11705
	Yes	180	PayPal	10757
	Yes	173	Cash	10552
	Yes	179	Credit Card	10539
	Yes	174	Venmo	10007



Advance Queries:

3) Calculate in which season the most Discount is applied

```
8
9  -- Calculate in which season the most Discount is applied
10 • Select Count(Discount_Applied) as Average,season from shop.shopping group by Season Order by Average desc ;
```

Output:

Result Grid			Filter Rows:	Export:	Wrap Cell Content:
	Average	season			
▶	999	Spring			
	975	Fall			
	971	Winter			
	955	Summer			



WINDOWS FUNCTION QUERIES

Windows Function Queries:

1) Give Top 10 Ranking of Item Purchased By Sum of Their Purchase Amount:

```
-- Give Top 10 Ranking of Item Purchased By Sum of Their Purchase Amount
Select Item_Purchased , Sum(Purchase_Amount) as Total , Rank() over (order by Sum(Purchase_Amount) Desc) as
Ranking From shop.shopping group by Item_Purchased limit 10;
```

Output:

Item_Purchased	Total	Ranking
Blouse	10410	1
Shirt	10332	2
Dress	10320	3
Pants	10090	4
Jewelry	10010	5
Sunglasses	9649	6
Belt	9635	7
Scarf	9561	8
Sweater	9462	9
Shorts	9433	10



Windows Function Queries:

2) Rank customers (Customer_ID) by their total Purchase_Amount within each Location using the ROW_RANK() function.

```
-- Rank customers (Customer_ID) by their total Purchase_Amount within each Location using the ROW_RANK() function.
Select Customer_ID, sum(Purchase_Amount) as total , row_number() over (order by Sum(Purchase_amount) Desc)
as Ranking From shop.shopping group by Customer_ID limit 10;
```

Output:

Customer_ID	total	Ranking
43	100	1
96	100	2
194	100	3
205	100	4
244	100	5
249	100	6
456	100	7
519	100	8
582	100	9
616	100	10



Windows Function Queries:

3) For each customer, calculate the difference in their Frequency_of_Purchases compared to the previous row using the LAG() function.

```
-- For each customer, calculate the difference in their Frequency_of_Purchases compared to the previous row using the LAG() function.  
SELECT Customer_ID, Purchase_Amount, Lag(Purchase_Amount) Over (Order by Customer_ID)  
as Previous , Purchase_Amount - Lag(Purchase_Amount) Over (Order by Customer_ID) as Difference from shop.shopping group by Customer_ID limit 10;
```

Output:

Customer_ID	Purchase_Amount	Previous	Difference
1	53	NULL	NULL
2	64	53	11
3	73	64	9
4	90	73	17
5	49	90	-41
6	20	49	-29
7	85	20	65
8	34	85	-51
9	97	34	63
10	31	97	-66



Windows Function Queries:

4) Include moving averages or percentiles for purchase patterns.

```
-- Include moving averages or percentiles for purchase patterns.  
SELECT Customer_ID,AVG(Purchase_Amount) OVER (PARTITION BY Location ORDER BY Customer_ID ROWS BETWEEN 2 PRECEDING AND CURRENT ROW)  
AS Moving_Average FROM shop.shopping;
```

Output:

Customer_ID	Moving_Average
17	36.0000
65	59.5000
71	49.3333
110	65.6667
183	70.0000
259	85.0000
409	78.6667
434	77.3333
465	70.0000
470	63.3333
544	65.3333
546	58.6667



CONCLUSION

Conclusions



In conclusion we come with following analysis:

- In Spring Season there is more sales and in Summer Season there is comparatively less.
- The Sum of total Purchase is \$233081 and from that there is \$75191 amount is spent by Female Gender and \$157890 amount is spent by Male Genders.
- There are Four Category In which items get purchased and among them the most revenue is generated by Clothing \$104264 and the least one is Outerwear Category with total revenue of \$18524.
- Among 3,900 customers, only 1,677 used promo codes, indicating a 43% adoption rate. This highlights an opportunity to improve promotional awareness. Promo code usage is low (43%). Improve awareness through targeted email campaigns or app notifications.
- Here Most Items purchased is at Montana and Illinois locations and least Item purchased is made by West Virginia and Nebraska .

26

To Be Continued....

Conclusions

In conclusion we come with following analysis:

- The Top 3 Most frequently Buy Item was Sweater, Skirt and Coat . This can be related because all locations present in table has average of 13°C temperature .
- Clothing generates the highest revenue \$104,264. Consider increasing inventory or promotions for this category during the Spring season, which sees peak sales.

Conclusions

Summary of your project's key metrics:

Metric	Value	Insight
Total Revenue	\$233,081	Optimize strategies for seasonal peaks
Top Category	Clothing (\$104,264)	Focus inventory and promotion on top items.
Promo Code Usage	43%	Boost awareness through targeted campaigns.
Top Location	Montana	Maintain stock and engage high value areas.
Low Location	West Virginia	Deploy location specific promotions.
Popular Items	Sweater, Skirt	Align stock with demand in colder seasons.
Peak Season	Spring	Maximize advertising and stock strategies.

**Thank's For
Watching**

