2025 Project

MySQL

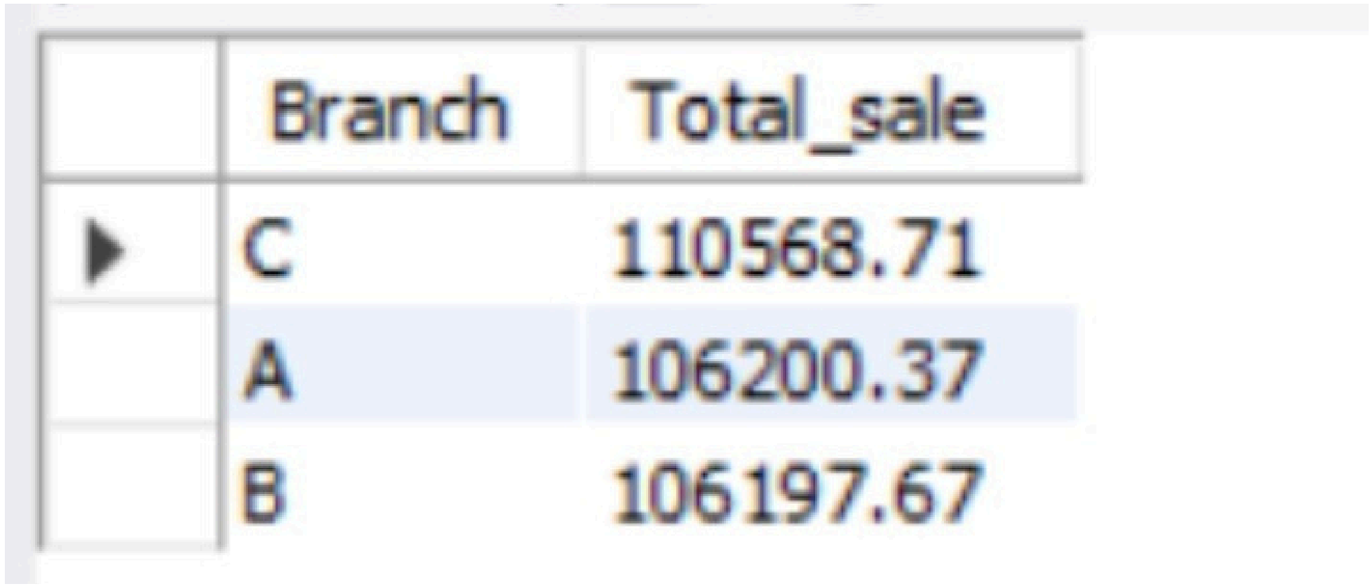
Walmart Sales Analysis

INTRODUCTION

This projects presents a comprehensive Data analysis of **Walmart's sales** performance using SQL . The goal is uncover sales trends , identify key factors influencing revenue and derive actionable business insights to support strategic decision making.

What are the total sales for each branch?

```
SELECT Branch, ROUND(SUM(Total)) AS Total_Sale
FROM walmartsaledata
GROUP BY Branch
ORDER BY Total_Sale DESC;
```



Branch	Total_sale
C	110568.71
A	106200.37
B	106197.67

List all sales from the 'Health and Beauty' product line that occurred in the month of march 2019.Show the Invoice id ,Date and Total.

```
SELECT InvoiceID, Date, Total
FROM walmartsaledata
WHERE ProductLine = 'Health and Beauty'
AND YEAR(Date) = 2019
AND MONTH(Date) = 3;
```

Invoice ID	Date	Total
829-34-3910	2019-03-29	749.49
656-95-9349	2019-03-11	506.6355
371-85-5789	2019-03-05	277.137
848-62-7243	2019-03-15	235.2105
595-11-5460	2019-03-15	202.818
333-73-7901	2019-03-23	461.328
617-15-4209	2019-03-16	32.277
877-22-3308	2019-03-13	166.635
152-08-9985	2019-03-12	608.202
766-85-7061	2019-03-29	922.635
716-39-1409	2019-03-19	223.0725
483-71-1164	2019-03-08	512.19
458-41-1477	2019-03-08	291.438
568-90-5112	2019-03-19	277.788
238-49-0436	2019-03-27	272.664
743-04-1105	2019-03-30	918.729
848-24-9445	2019-03-26	91.77
699-01-4164	2019-03-12	174.3
157-13-5295	2019-03-09	545.37
448-81-5016	2019-03-11	125.517
667-92-0055	2019-03-04	628.929
565-17-3836	2019-03-12	200.214
585-03-5943	2019-03-13	190.155
573-10-3877	2019-03-12	40.9605

How many unique customers made purchase each day

```
SELECT Date, COUNT(DISTINCT Invoice_ID) AS Unique_Customer
FROM walmartsaledata
GROUP BY Date
ORDER BY Date;
```

Date	Unique_customer
2019-01-01	12
2019-01-02	8
2019-01-03	8
2019-01-04	6
2019-01-05	12
2019-01-06	9
2019-01-07	9
2019-01-08	18
2019-01-09	8
2019-01-10	9
2019-01-11	8
2019-01-12	11
2019-01-13	10
2019-01-14	13
2019-01-15	13
2019-01-16	10
2019-01-17	11
2019-01-18	9
2019-01-19	16
2019-01-20	10
2019-01-21	8
2019-01-22	7
2019-01-23	17

What is the average income per transaction for each product line and which one has the highest?

```
SELECT  Product line , ROUND(AVG(Gross income) AS   Avg_gross_income
FROM walmartsaledata
GROUP BY Product line
ORDER BY  Avg_gross_income DESC
LIMIT 1
```

	Product line	Avg_gross_income
▶	Home and lifestyle	16

For each city. what is the most popular method?

```
WITH paymentCounts AS (
SELECT City, Payment, COUNT(*) AS Payment,
ROW_NUMBER() OVER (PARTITION BY City ORDER BY COUNT(*) DESC) AS rankNum
FROM walmartSaleData
GROUP BY  City, Payment )
SELECT City, Payment
FROM paymentCounts WHERE rankNum = 1;
```

City	Payment
Mandalay	Ewallet
Naypyitaw	Cash
Yangon	Ewallet

Calculate the total sales for each hour of the day across all branch.
what is the peak sales hour .

```
WITH HourlySales AS (
SELECT Branch,  HOUR(Time) AS Peak_Hour, ROUND(SUM(Total), 1) AS Total_Sale,
ROW_NUMBER() OVER (PARTITION BY Branch ORDER BY SUM(Total) DESC) AS Ranks
FROM WalmartSalesData
GROUP BY Branch, Time )
SELECT Branch, Peak_Hour, Total_Sale
FROM HourlySales  WHERE Ranks = 1
ORDER BY Branch;
```

Branch	Peak_Hour	Total_sale
A	15	1244.2
B	17	1453.2
C	13	1963.6

Categorize sales into low, Medium ,High .count the number of transactions in each category.

```
SELECT
 CASE
        WHEN Total < 100 THEN 'Low'
        WHEN Total >= 100 AND Total < 500 THEN  'Medium'
        ELSE  'High'
   END AS Sales_category,
    COUNT(*) AS Number_of_transactions
FROM  walmartsalesdata
 GROUP BY Sales_category
 ORDER BY Number_of_transactions;
```

Sales_category	Number_of_transactions
Low	208
High	227
Medium	565

What percentage of total sales does each branch contribute?

```
SELECT  Branch,
ROUND(SUM(Total), 2) AS Branch_sales,
ROUND((SUM(Total) / (SELECT SUM(Total)
FROM  walmartsalesdata)) * 100, 2) AS Sales_percentage
FROM walmartsalesdata
GROUP BY Branch
ORDER BY  Sales_percentage DESC ;
```

	Branch	Branch_sales	Sales_percentage
▶	C	110568.71	34.24
	A	106200.37	32.88
	B	106197.67	32.88

Calculate the running total of sales(month to date) for the entire year 2019.Ordered by date.

```
SELECT  Date,
ROUND(SUM(Total), 2) AS Daily_Sales,
ROUND(SUM(SUM(Total)) OVER (
PARTITION BY YEAR(Date), MONTH(Date)
ORDER BY Date
), 2) AS MTD_Running_Total
FROM walmartsalesdata
WHERE  YEAR(Date) = 2019
GROUP BY  Date
ORDER BY  Date;
```

Date	Daily_Sales	MTD_Running_Total
2019-01-01	4745.18	4745.18
2019-01-02	1945.5	6690.68
2019-01-03	2078.13	8768.81
2019-01-04	1623.69	10392.5
2019-01-05	3536.68	13929.18
2019-01-06	3614.2	17543.39
2019-01-07	2834.24	20377.63
2019-01-08	5293.73	25671.37
2019-01-09	3021.34	28692.71
2019-01-10	3560.95	32253.66
2019-01-11	2114.96	34368.62
2019-01-12	5184.76	39553.38
2019-01-13	2451.2	42004.59
2019-01-14	3966.62	45971.2
2019-01-15	5944.26	51915.46
2019-01-16	4289.08	56204.55
2019-01-17	3142.76	59347.3
2019-01-18	2780.47	62127.78
2019-01-19	4914.72	67042.5
2019-01-20	3655.45	70697.95
2019-01-21	2392.1	73090.05
2019-01-22	1704.77	74794.82

Identify the top spending customers for each product line.

```
WITH Customerspending AS (
SELECT  `Product line`, `Invoice ID`,
ROUND(SUM(Total), 1) AS Spending,
ROW_NUMBER() OVER (PARTITION BY `Product line` ORDER BY SUM(Total) DESC) AS rnk
FROM walmartsalesdata
GROUP BY `Product line`, `Invoice ID` )
SELECT `Product line`,`Invoice ID`, Spending
FROM Customerspending
WHERE rnk = 1
ORDER BY Spending DESC;
```

Product line	Invoice ID	Spending
Fashion accessories	860-79-0874	1042.6
Food and beverages	283-26-5248	1034.5
Home and lifestyle	751-41-9720	1023.8
Sports and travel	554-42-2417	1002.1
Health and beauty	280-17-4359	950.2
Electronic accessories	817-69-8206	942.4

Find the product lines that are consistently sold above their average unit. List them along with the difference from the average.

```
WITH ProductAvgPrice AS (
SELECT `Product line`,AVG(`Unit price`) AS Avg_Unit_Price  FROM walmartsalesdata
GROUP BY `Product line` ), ProductSales AS (
SELECT `Product line`,`Invoice ID`,`Unit price`,
( SELECT Avg_Unit_Price FROM ProductAvgPrice p WHERE p.`Product line` = w.`Product line`) AS Product_Avg_Price
FROM walmartsalesdata w ) SELECT `Product line`, COUNT(*) AS Transactions_Above_Avg,
ROUND(AVG(`Unit price` - Product_Avg_Price), 2) AS Avg_Price_Difference,
ROUND(MIN(`Unit price` - Product_Avg_Price), 2) AS Min_Price_Difference,
ROUND(MAX(`Unit price` - Product_Avg_Price), 2) AS Max_Price_Difference
FROM ProductSales WHERE `Unit price` > Product_Avg_Price GROUP BY `Product line`
HAVING COUNT(*) > 10
ORDER BY Avg_Price_Difference DESC;
```

Product line	Transactions_Above_Avg	Avg_Price_Difference	Min_Price_Difference	Max_Price_Difference
Electronic accessories	80	24.97	2.12	46.18
Food and beverages	81	24.25	0.55	43.78
Fashion accessories	85	23.93	0.12	42.74
Sports and travel	86	22.78	0.13	42.97
Home and lifestyle	81	22.49	0.25	44.6
Health and beauty	84	20.36	0.01	45.11

Compare each product line sales in the current month with the previous month. Calculate the month over month growth percent

```
WITH MonthlySales AS (
SELECT `Product line`,DATE_FORMAT(Date, '%Y-%m') AS
Sales_Month,
ROUND(SUM(Total), 2) AS Monthly_Sales
FROM walmartsalesdata
GROUP BY `Product line`, DATE_FORMAT(Date, '%Y-%m')
), SalesWithLag AS (
SELECT `Product line`,Sales_Month,Monthly_Sales,
LAG(Monthly_Sales) OVER (PARTITION BY `Product line`
ORDER BY Sales_Month
) AS Previous_Month_Sales
FROM MonthlySales
) SELECT `Product
line`,Sales_Month,Monthly_Sales,Previous_Month_Sales,
ROUND(((Monthly_Sales - Previous_Month_Sales) /
NULLIF(Previous_Month_Sales, 0)) * 100,2
) AS MoM_Growth_Percent from SalesWithLag order by
`Product line`,Sales_Month;
```

Product line	Sales_Month	Monthly_Sales	Previous_Month_Sales	MoM_Growth_Percent
Electronic accessories	2019-01	18831.29	NULL	NULL
Electronic accessories	2019-02	17362.9	18831.29	-7.8
Electronic accessories	2019-03	18143.34	17362.9	4.49
Fashion accessories	2019-01	19345.12	NULL	NULL
Fashion accessories	2019-02	19009.86	19345.12	-1.73
Fashion accessories	2019-03	15950.92	19009.86	-16.09
Food and beverages	2019-01	19570.53	NULL	NULL
Food and beverages	2019-02	20000.36	19570.53	2.2
Food and beverages	2019-03	16573.96	20000.36	-17.13
Health and beauty	2019-01	16383.17	NULL	NULL
Health and beauty	2019-02	14602.26	16383.17	-10.87
Health and beauty	2019-03	18208.31	14602.26	24.7
Home and lifestyle	2019-01	20494.74	NULL	NULL
Home and lifestyle	2019-02	12434.38	20494.74	-39.33
Home and lifestyle	2019-03	20932.79	12434.38	68.35
Sports and travel	2019-01	21667.02	NULL	NULL
Sports and travel	2019-02	13809.61	21667.02	-36.26
Sports and travel	2019-03	19646.19	13809.61	42.26

CONCLUSION

- Branch Performance :- Each branch contributed almost equally to total sales, with only a slight variation. This indicates balanced revenue distribution and efficient performance across locations.
- Sales Contribution & Categories :- Sales were classified into Low, Medium, and High categories, with the majority of transactions falling into the Medium range. This suggests that most customers prefer moderate spending patterns rather than extremes.
- Product Line Insights :- Product lines such as Food and Beverages, Fashion Accessories, and Health & Beauty consistently generated higher revenue. Certain product lines regularly sold above their average unit price, showing strong customer demand and brand value.
- Business Implications :- Walmart should leverage high-performing product lines while working on improving weaker ones. Customer retention strategies can focus on high-spending customers. Balanced branch sales suggest operational stability, but further optimization can increase efficiency.