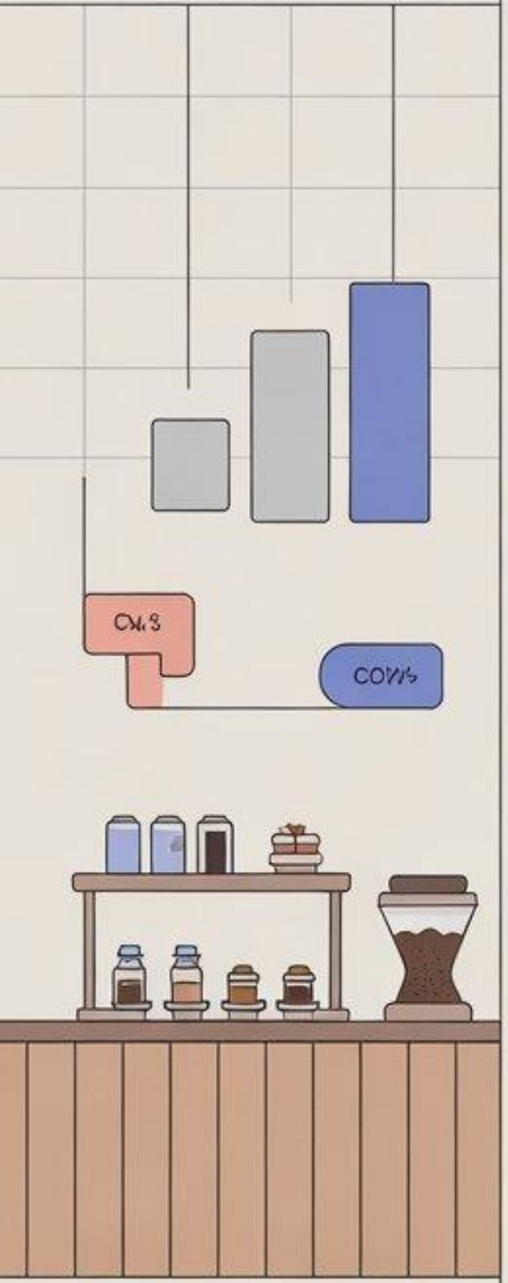
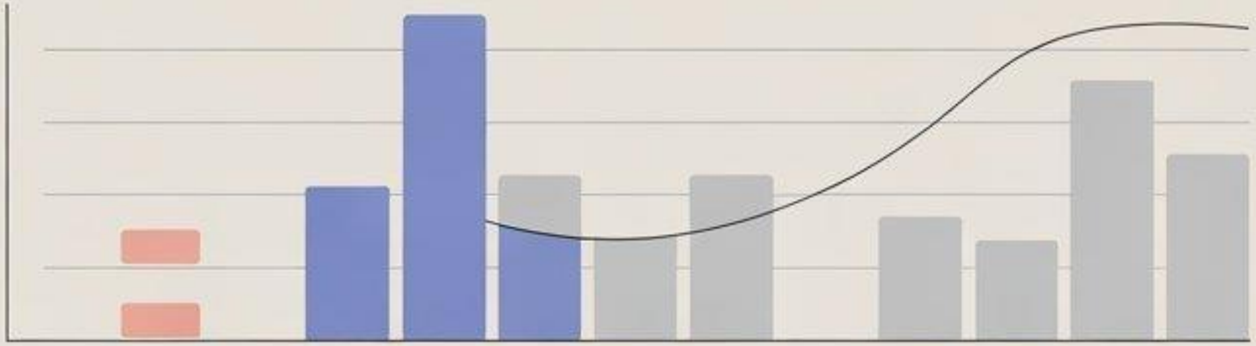
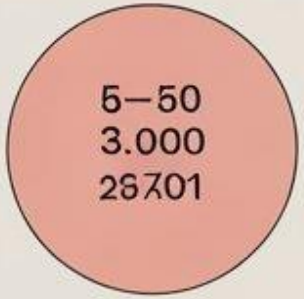


Github

Kap's Café

SQL Analytics Project





Data Insights

Kap's Café SQL Project

Objective

1. Analyze Revenue, Orders, and Quantities with Precision

- Calculate monthly sales revenue by combining unit prices and transaction quantities.
- Track order volumes and product demand to understand customer purchasing behavior.
- Compare absolute and relative changes month-over-month (MoM) to detect growth or decline.

2. Identify Monthly Trends and Demand Shifts

- Use time-series aggregation to uncover seasonal patterns and consumption peaks.
- Highlight high-variance months where performance significantly deviates from the norm.
- Distinguish between revenue-driven growth (higher spending per order) vs. volume-driven growth (more orders/quantities).

3. Evaluate Business Performance with Diagnostic Metrics

- Measure stability and volatility by calculating absolute differences in sales, orders, and quantities across months.
- Pinpoint months with declining orders but stable or rising revenue, signaling a possible premium-pricing effect.
- Isolate anomalies (e.g., sudden sales drops or spikes) that may indicate promotions, stockouts, or reporting gaps.

3. Provide Actionable, Data-Driven Recommendations for Growth

- Recommend promotions in low-performing months based on historical trends.
- Suggest inventory adjustments aligned with demand seasonality to reduce waste and stockouts.
- Propose loyalty and retention strategies to stabilize fluctuating order volumes.
- Refine pricing strategies by identifying periods where revenue growth outpaces order growth.

Data
Insights

Kap’s Café SQL Project

Problem Statement

Kap’s Café management struggled with making data-driven decisions because their transactional data was stored but not actively analyzed. They relied on spreadsheets and intuition, which led to blind spots in revenue tracking, customer demand, and operational efficiency.

To move beyond guesswork, management posed the following key business questions:

1. How much sales revenue is generated each month?
2. Are sales increasing or decreasing month-over-month (MoM)?
3. What is the difference in sales and orders compared to the previous month?
4. How many unique customer orders are received each month?
5. How much product quantity is sold, and how does it change monthly?

Without structured SQL-based analysis, these questions could not be answered reliably, leaving management without the insights needed for strategic planning.

Data Insights

Kap's Café SQL Project

Data Description

Source Table: KAPS

The primary dataset is a transactional sales table containing order-level data for Kap's Café. Each row represents a line item within a customer transaction.

Columns Used in Analysis

1. transaction_id → Unique identifier for each transaction. Used to count distinct customer orders.
2. transaction_date → Date of the transaction. Enables monthly aggregations and time-series analysis.
3. unit_price → Price of a single product unit. When multiplied with transaction_qty, forms the basis of revenue.
4. transaction_qty → Quantity of items sold in a transaction line. Summed to calculate total demand.

Derived Metrics Revenue = unit_price * transaction_qty

- Captures sales value at the most granular level.
- Aggregated by month to calculate total sales and monitor trends.

Why These Fields Matter

- These four fields cover the core business levers: how many items were sold, at what price, when, and in how many distinct orders.
- All 10 SQL queries in this project derive directly from this simple but powerful schema.

Data Insights

Kap’s Café SQL Project

Approach / Methodology

To transform raw transaction data into actionable business insights, the following structured methodology was applied:

1. Data Extraction with SQL

- Pulled data directly from the KAPS table.
- Focused on key columns: transaction_id, transaction_date, unit_price, transaction_qty.

2. Monthly Aggregation

- Grouped transactions into monthly buckets (yyyy-MM).
- Computed total revenue, number of unique orders, and total quantity sold for each month.
- Established the baseline trend analysis.

3. Window Functions (LAG) for MoM Change

- Applied the LAG() function to compare current month metrics against the previous month.
- Measured Month-over-Month (MoM) growth or decline in sales, orders, and quantities.

4. Absolute Differences for Stability Checks

- Calculated the absolute delta between months.
- Helped identify volatility (sharp spikes or drops), not just directional change.

5. Business Interpretation → Recommendations

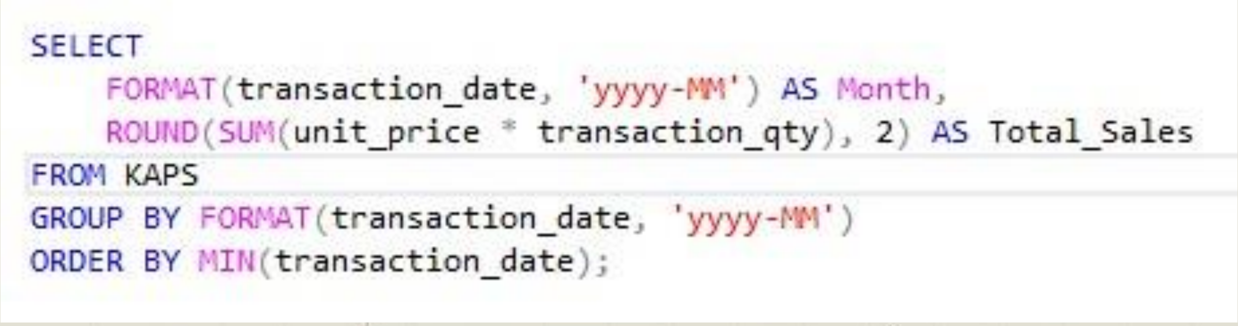
- Translated numeric outputs into actionable insights for management.
- Linked findings to potential strategies (promotions, pricing, inventory, loyalty programs).

Data Insights

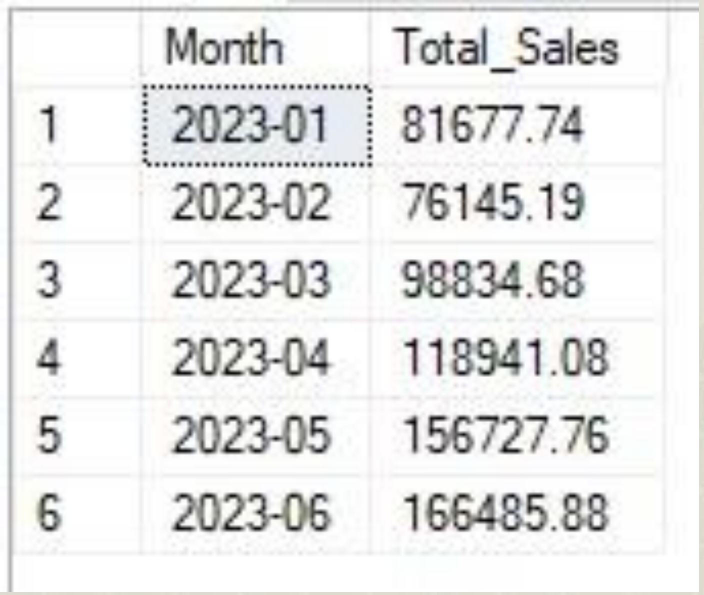
Kap’s Café SQL Project

Business Questions

1. What is the total sales value (₹) for each month?



```
SELECT
    FORMAT(transaction_date, 'yyyy-MM') AS Month,
    ROUND(SUM(unit_price * transaction_qty), 2) AS Total_Sales
FROM KAPS
GROUP BY FORMAT(transaction_date, 'yyyy-MM')
ORDER BY MIN(transaction_date);
```



	Month	Total_Sales
1	2023-01	81677.74
2	2023-02	76145.19
3	2023-03	98834.68
4	2023-04	118941.08
5	2023-05	156727.76
6	2023-06	166485.88

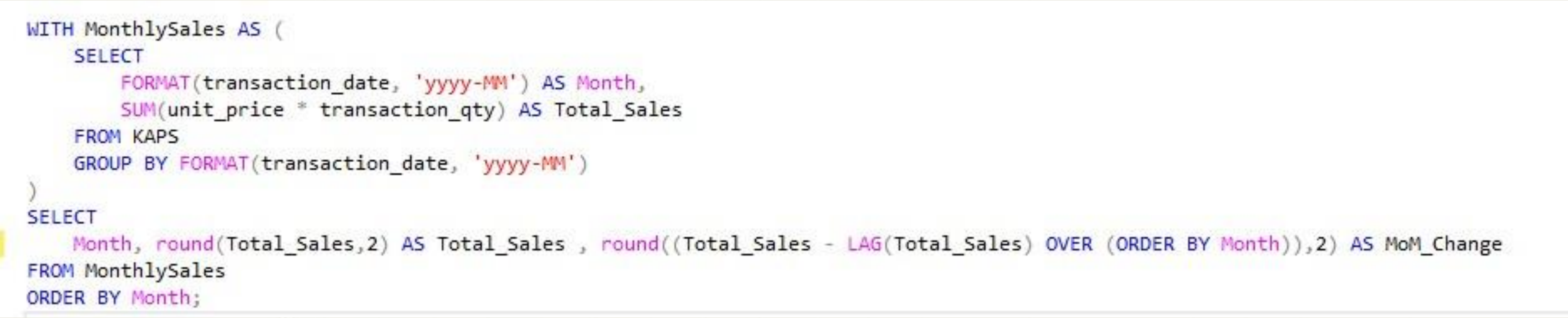
Key Insights :

- Sales revenue trends showed **peaks and troughs across months.**
- Identified high-revenue months (possibly seasonal or promotion-driven).
- Low-performing months revealed **opportunities for targeted campaigns.**

Data Insights

Kap’s Café SQL Project

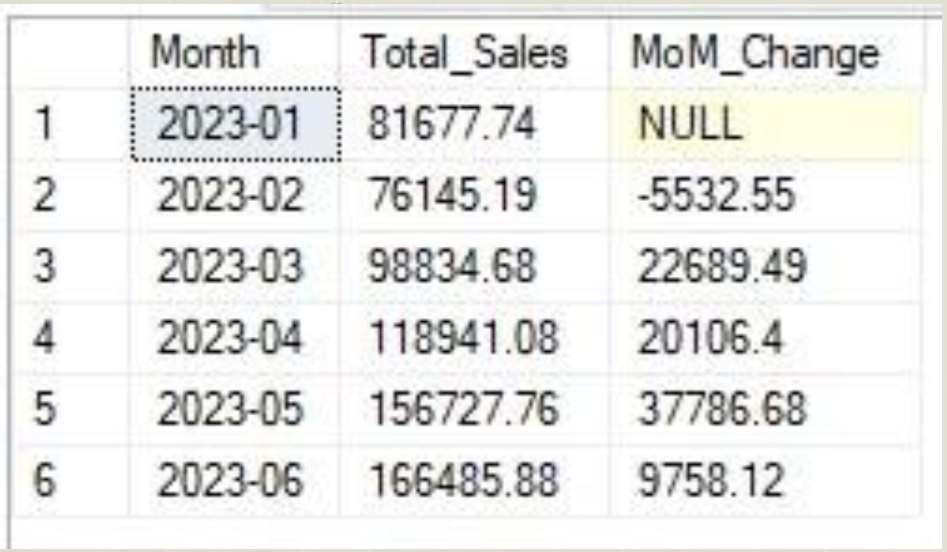
2. What is the month-on-month (MoM) increase or decrease in total sales (₹)?



```
WITH MonthlySales AS (
    SELECT
        FORMAT(transaction_date, 'yyyy-MM') AS Month,
        SUM(unit_price * transaction_qty) AS Total_Sales
    FROM KAPS
    GROUP BY FORMAT(transaction_date, 'yyyy-MM')
)
SELECT
    Month, round(Total_Sales,2) AS Total_Sales , round((Total_Sales - LAG(Total_Sales) OVER (ORDER BY Month)),2) AS MoM_Change
FROM MonthlySales
ORDER BY Month;
```

Key Insights :

- Some months experienced steady growth, while others had sharp declines.
- Positive MoM changes indicated effective periods (potential promotions or customer inflow).
- Negative MoM highlighted business risk periods, useful for management reviews.

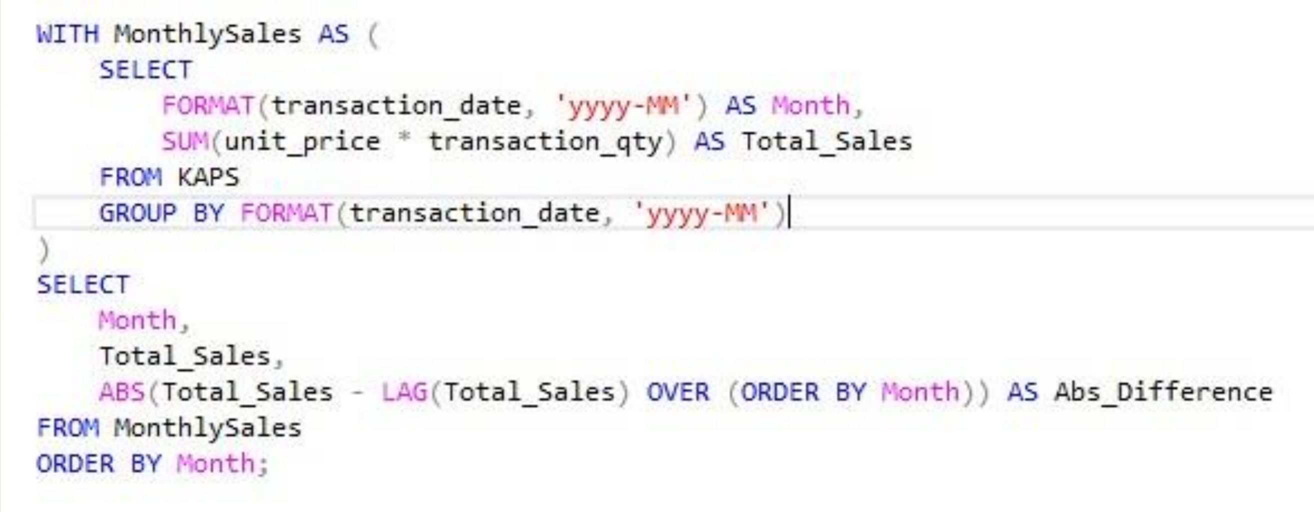


	Month	Total_Sales	MoM_Change
1	2023-01	81677.74	NULL
2	2023-02	76145.19	-5532.55
3	2023-03	98834.68	22689.49
4	2023-04	118941.08	20106.4
5	2023-05	156727.76	37786.68
6	2023-06	166485.88	9758.12

Data Insights

Kap’s Café SQL Project

Q.3 What is the absolute difference in total sales (₹) between the selected month and its previous

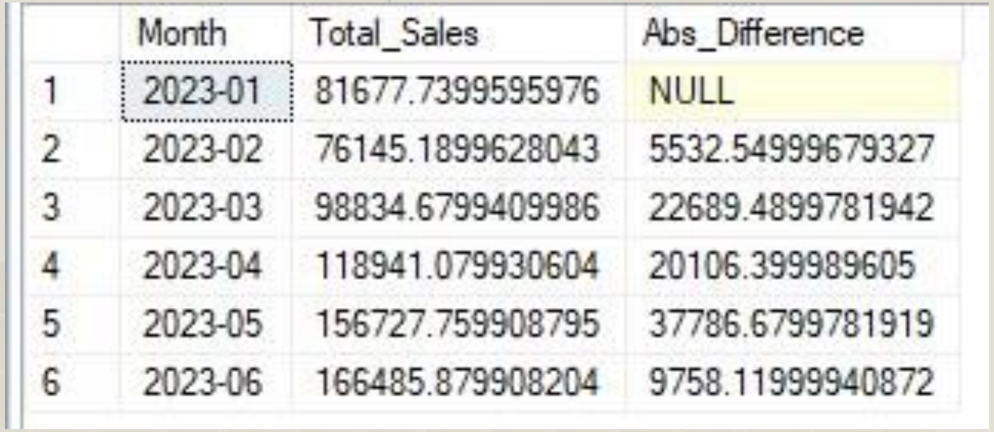


```
WITH MonthlySales AS (
    SELECT
        FORMAT(transaction_date, 'yyyy-MM') AS Month,
        SUM(unit_price * transaction_qty) AS Total_Sales
    FROM KAPS
    GROUP BY FORMAT(transaction_date, 'yyyy-MM')
)
SELECT
    Month,
    Total_Sales,
    ABS(Total_Sales - LAG(Total_Sales) OVER (ORDER BY Month)) AS Abs_Difference
FROM MonthlySales
ORDER BY Month;
```

Key Insights :

- The absolute variance helped spot volatility in sales performance.

- Large differences flagged anomalous months — requiring deeper investigation (e.g., supply chain issues, holidays, or external factors).

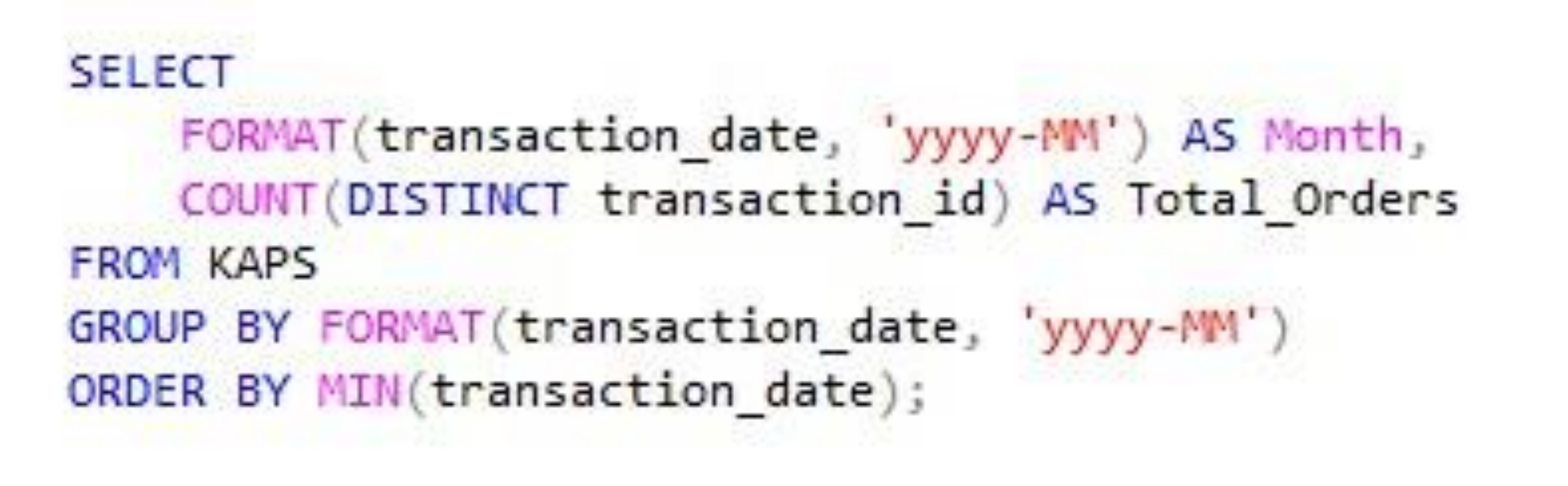


	Month	Total_Sales	Abs_Difference
1	2023-01	81677.7399595976	NULL
2	2023-02	76145.1899628043	5532.54999679327
3	2023-03	98834.6799409986	22689.4899781942
4	2023-04	118941.079930604	20106.399989605
5	2023-05	156727.759908795	37786.6799781919
6	2023-06	166485.879908204	9758.1199940872

Data Insights

Kap's Café SQL Project

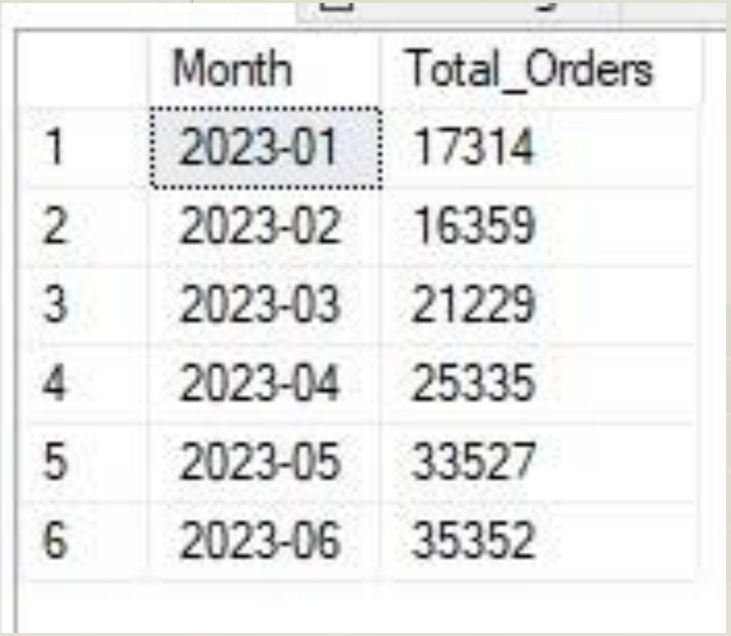
Q.4. What is the total number of orders for each month?



```
SELECT
    FORMAT(transaction_date, 'yyyy-MM') AS Month,
    COUNT(DISTINCT transaction_id) AS Total_Orders
FROM KAPS
GROUP BY FORMAT(transaction_date, 'yyyy-MM')
ORDER BY MIN(transaction_date);
```

Key Insights :

- Monthly order counts revealed customer engagement levels.
- Higher orders aligned with peak café traffic; lower orders suggested drop-offs in demand.
- Provided a baseline for understanding revenue drivers.

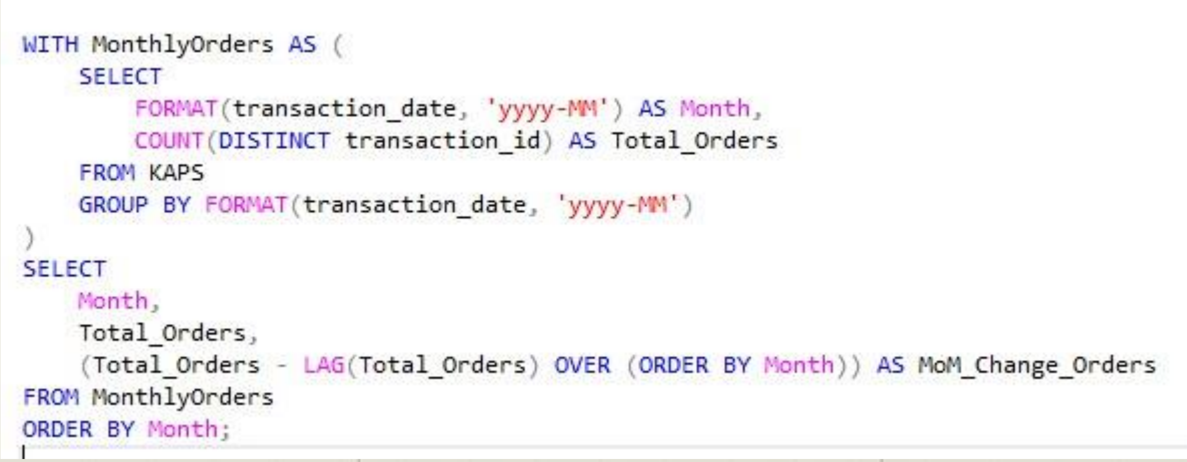


	Month	Total_Orders
1	2023-01	17314
2	2023-02	16359
3	2023-03	21229
4	2023-04	25335
5	2023-05	33527
6	2023-06	35352

Data Insights

Kap's Café SQL Project

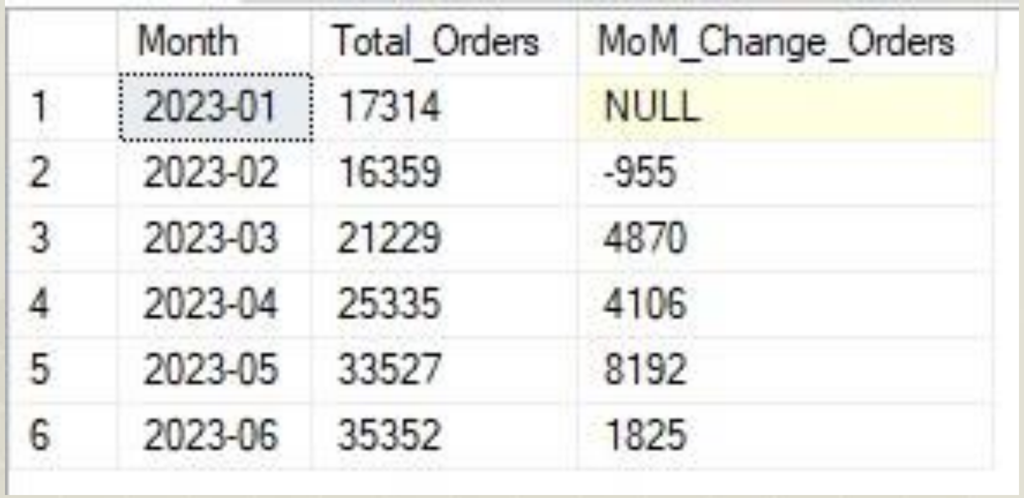
Q.5. What is the month-on-month (MoM) increase or decrease in the number of orders?



```
WITH MonthlyOrders AS (
    SELECT
        FORMAT(transaction_date, 'yyyy-MM') AS Month,
        COUNT(DISTINCT transaction_id) AS Total_Orders
    FROM KAPS
    GROUP BY FORMAT(transaction_date, 'yyyy-MM')
)
SELECT
    Month,
    Total_Orders,
    (Total_Orders - LAG(Total_Orders) OVER (ORDER BY Month)) AS MoM_Change_Orders
FROM MonthlyOrders
ORDER BY Month;
```

Key Insights :

- MoM changes showed fluctuations in customer activity.
- Declining orders during stable revenue months suggested higher spending per customer.
- Growth months confirmed success in attracting more customers.

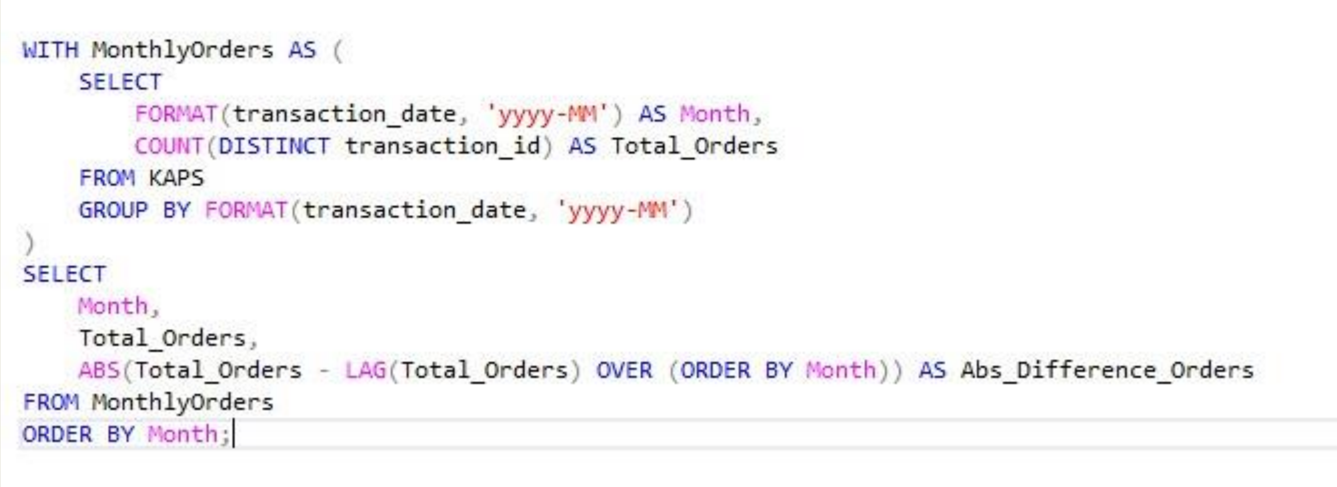


	Month	Total_Orders	MoM_Change_Orders
1	2023-01	17314	NULL
2	2023-02	16359	-955
3	2023-03	21229	4870
4	2023-04	25335	4106
5	2023-05	33527	8192
6	2023-06	35352	1825

Data Insights

Kap's Café SQL Project

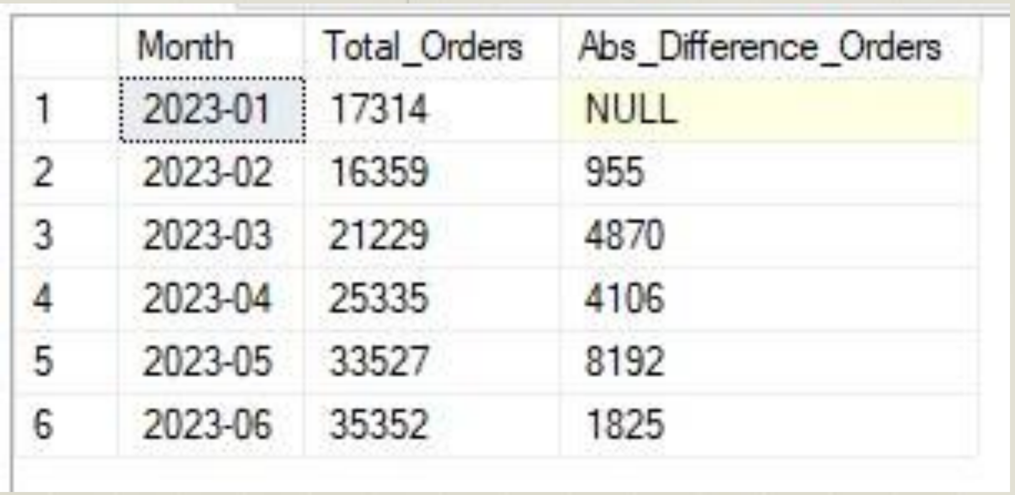
Q.6. What is the absolute difference in the number of orders between the selected month and its previous month?



```
WITH MonthlyOrders AS (
    SELECT
        FORMAT(transaction_date, 'yyyy-MM') AS Month,
        COUNT(DISTINCT transaction_id) AS Total_Orders
    FROM KAPS
    GROUP BY FORMAT(transaction_date, 'yyyy-MM')
)
SELECT
    Month,
    Total_Orders,
    ABS(Total_Orders - LAG(Total_Orders) OVER (ORDER BY Month)) AS Abs_Difference_Orders
FROM MonthlyOrders
ORDER BY Month;
```

Key Insights :

- Quantified the scale of monthly order shifts.
- Identified months with sharp customer drop-offs – potential signals of service gaps, seasonality, or lack of marketing.



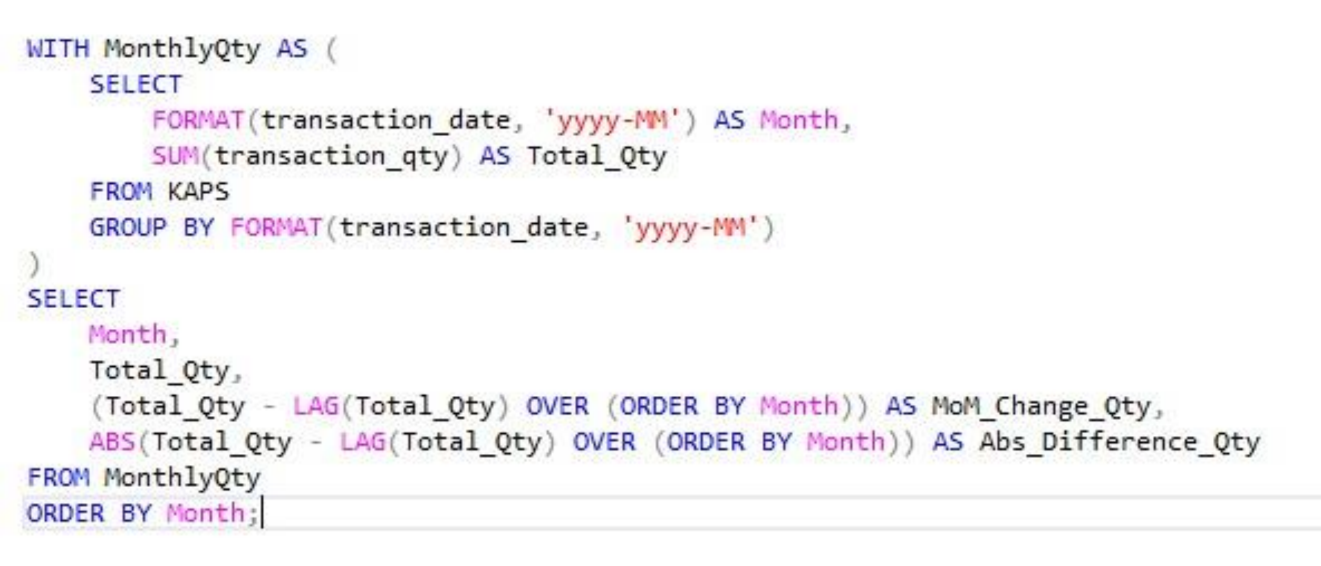
	Month	Total_Orders	Abs_Difference_Orders
1	2023-01	17314	NULL
2	2023-02	16359	955
3	2023-03	21229	4870
4	2023-04	25335	4106
5	2023-05	33527	8192
6	2023-06	35352	1825

Data

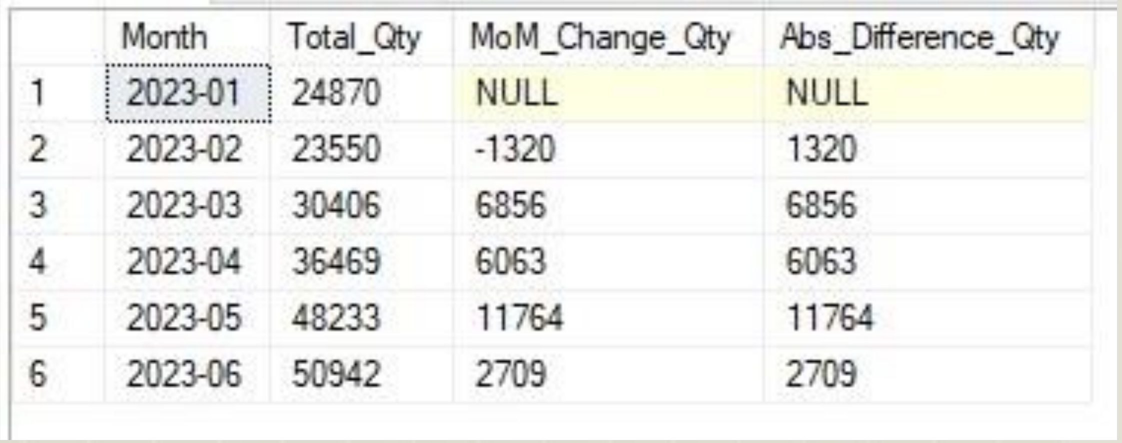
Insights

Kap's Café SQL Project

Q.7. . Total Quantity Sold Analysis



```
WITH MonthlyQty AS (
    SELECT
        FORMAT(transaction_date, 'yyyy-MM') AS Month,
        SUM(transaction_qty) AS Total_Qty
    FROM KAPS
    GROUP BY FORMAT(transaction_date, 'yyyy-MM')
)
SELECT
    Month,
    Total_Qty,
    (Total_Qty - LAG(Total_Qty) OVER (ORDER BY Month)) AS MoM_Change_Qty,
    ABS(Total_Qty - LAG(Total_Qty) OVER (ORDER BY Month)) AS Abs_Difference_Qty
FROM MonthlyQty
ORDER BY Month;
```



	Month	Total_Qty	MoM_Change_Qty	Abs_Difference_Qty
1	2023-01	24870	NULL	NULL
2	2023-02	23550	-1320	1320
3	2023-03	30406	6856	6856
4	2023-04	36469	6063	6063
5	2023-05	48233	11764	11764
6	2023-06	50942	2709	2709

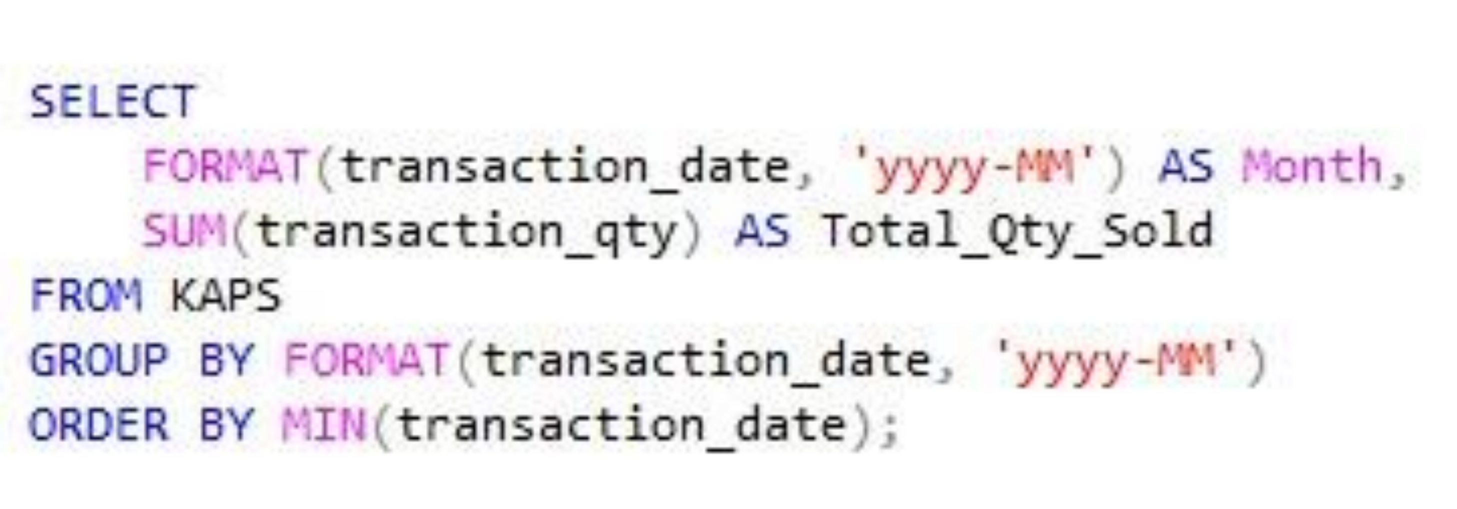
Key Insights :

- Quantity sold varied significantly by month, reflecting changes in demand intensity.
- In some cases, quantity dropped but revenue grew → indicating premium product sales.
- Helped link revenue to volume vs. pricing effects.

Data Insights

Kap's Café SQL Project

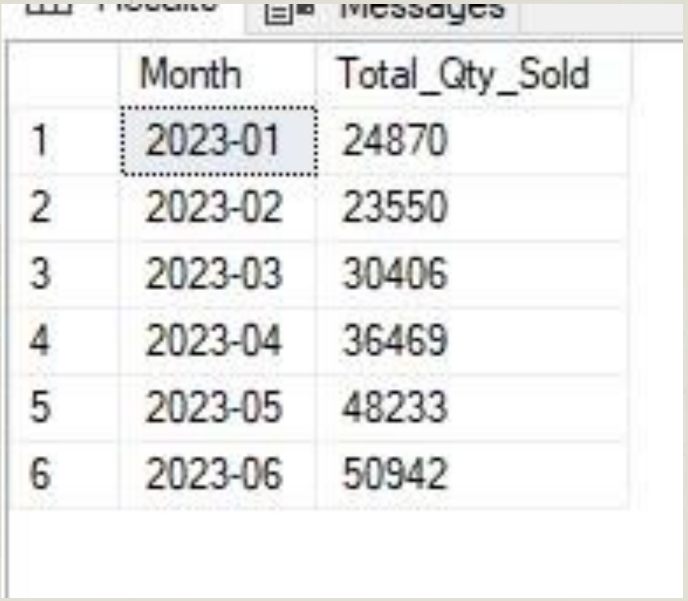
Q.8. What is the total quantity sold (units) for each month?



```
SELECT
    FORMAT(transaction_date, 'yyyy-MM') AS Month,
    SUM(transaction_qty) AS Total_Qty_Sold
FROM KAPS
GROUP BY FORMAT(transaction_date, 'yyyy-MM')
ORDER BY MIN(transaction_date);
```

Key Insights

- Monthly unit counts provided a direct measure of product movement.
- Peak months flagged high-demand periods, useful for inventory planning.
- Low months signaled possible menu/product underperformance.

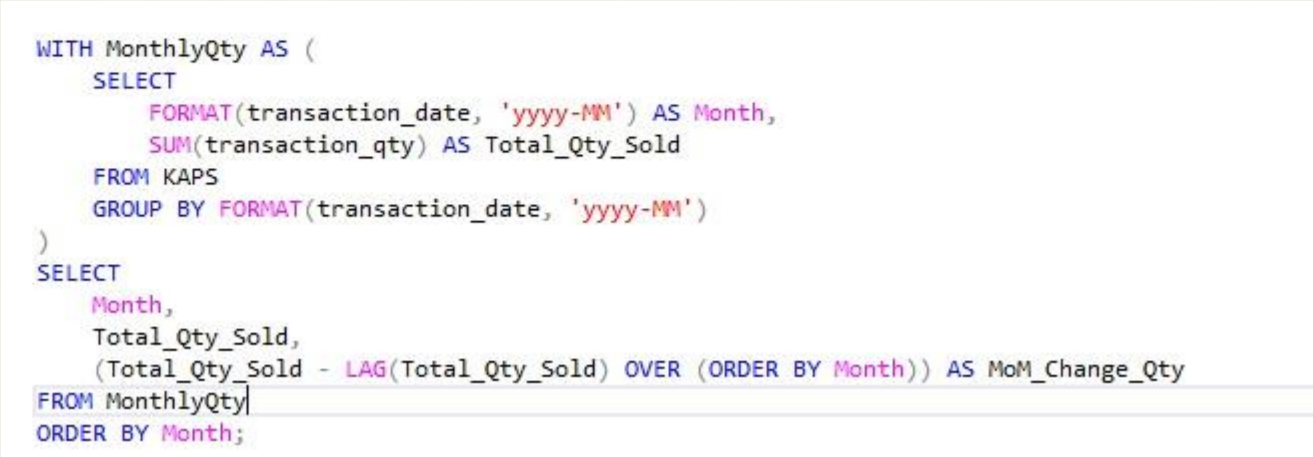


	Month	Total_Qty_Sold
1	2023-01	24870
2	2023-02	23550
3	2023-03	30406
4	2023-04	36469
5	2023-05	48233
6	2023-06	50942

Data Insights

Kap’s Café SQL Project

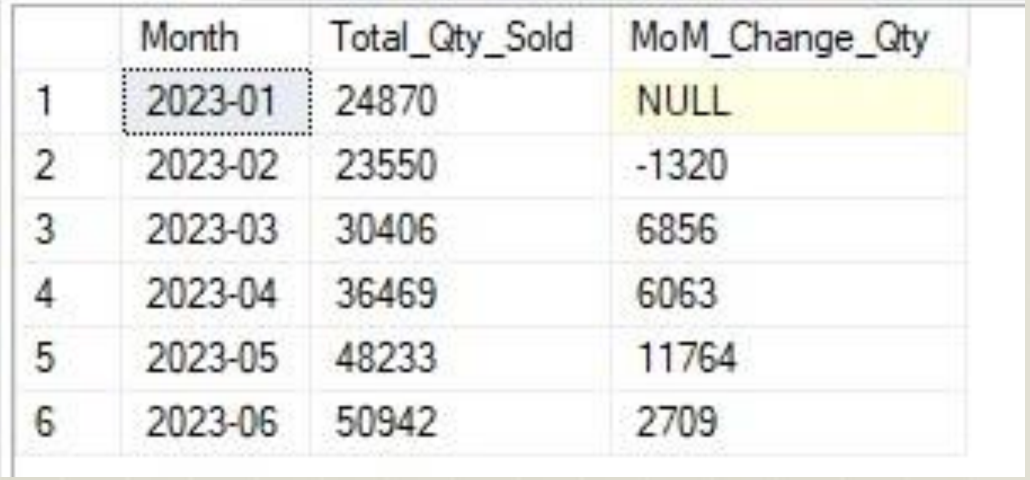
Q.9. What is the month-on-month (MoM) increase or decrease in total quantity sold?



```sql
WITH MonthlyQty AS (
    SELECT
        FORMAT(transaction_date, 'yyyy-MM') AS Month,
        SUM(transaction_qty) AS Total_Qty_Sold
    FROM KAPS
    GROUP BY FORMAT(transaction_date, 'yyyy-MM')
)
SELECT
    Month,
    Total_Qty_Sold,
    (Total_Qty_Sold - LAG(Total_Qty_Sold) OVER (ORDER BY Month)) AS MoM_Change_Qty
FROM MonthlyQty
ORDER BY Month;
```

Key Insights

- MoM analysis exposed demand spikes and drops.
- Growth aligned with successful promotions or seasonal boosts.
- Declines highlighted months requiring operational adjustments.



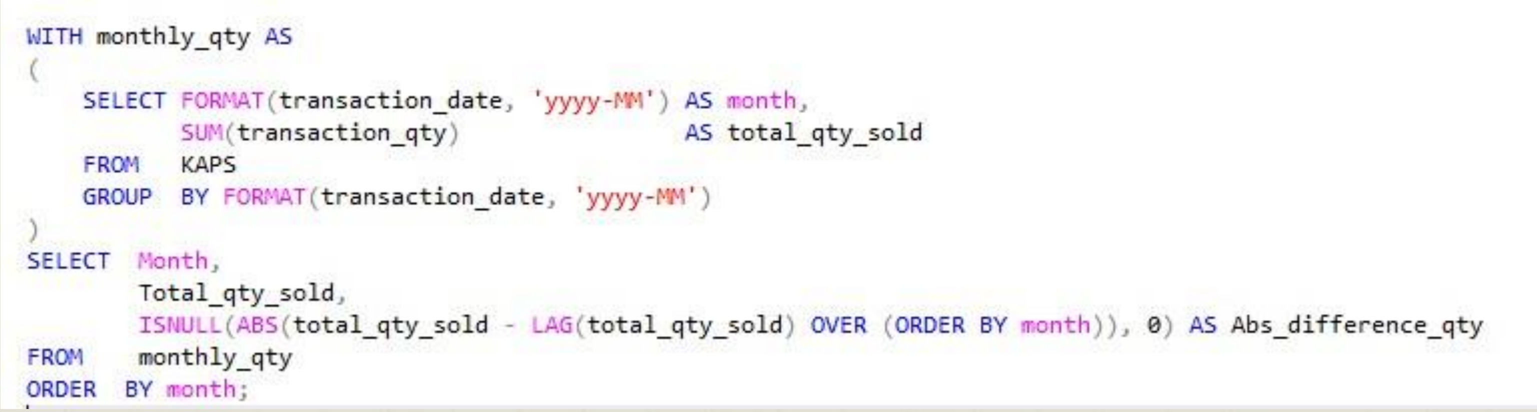
	Month	Total_Qty_Sold	MoM_Change_Qty
1	2023-01	24870	NULL
2	2023-02	23550	-1320
3	2023-03	30406	6856
4	2023-04	36469	6063
5	2023-05	48233	11764
6	2023-06	50942	2709

Data

Insights

Kap’s Café SQL Project

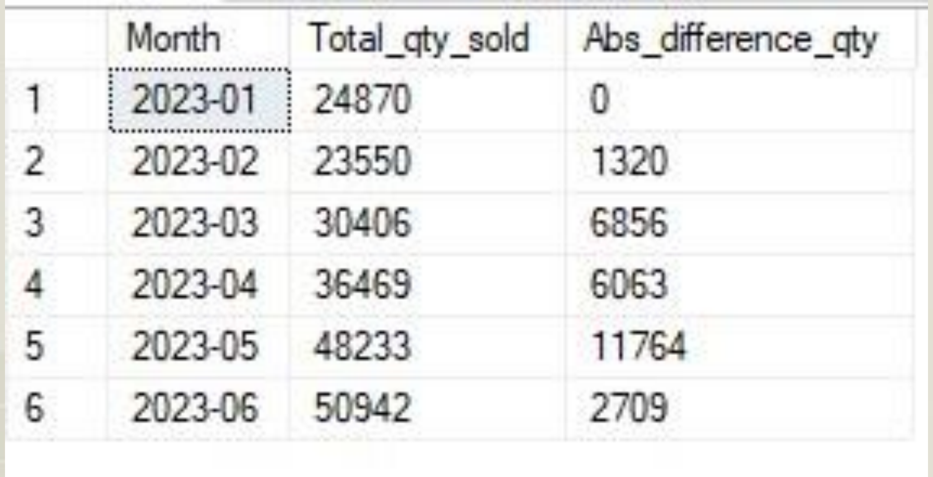
Q.10. What is the absolute difference in total quantity sold between the selected month and its previous month?



```
WITH monthly_qty AS
(
    SELECT FORMAT(transaction_date, 'yyyy-MM') AS month,
           SUM(transaction_qty)                AS total_qty_sold
    FROM   KAPS
    GROUP  BY FORMAT(transaction_date, 'yyyy-MM')
)
SELECT  Month,
        Total_qty_sold,
        ISNULL(ABS(total_qty_sold - LAG(total_qty_sold) OVER (ORDER BY month)), 0) AS Abs_difference_qty
FROM    monthly_qty
ORDER  BY month;
```

Key Insights

- **Demand Volatility Detected** → Some months showed sharp jumps/drops in quantity sold, signaling unstable customer demand.
- **Inventory Risks** → Large positive differences may risk stockouts, while negative swings could lead to overstocking.
- **Seasonal/Promotional Effects** → Spikes likely tied to promotions or seasonal demand, while stable months indicate consistent buying patterns.
- **Operational Impact** → Tracking absolute differences helps plan inventory and staffing to handle high-variance months.



	Month	Total_qty_sold	Abs_difference_qty
1	2023-01	24870	0
2	2023-02	23550	1320
3	2023-03	30406	6856
4	2023-04	36469	6063
5	2023-05	48233	11764
6	2023-06	50942	2709

Data Insights

Kap's Café SQL Project

Key Insights (Summary)

1. Revenue Trends 📈

- Monthly revenue fluctuated, with clear growth in certain periods and dips in others.
- MoM comparisons revealed seasonal effects and sharp deviations (high-variance months).
- In some months, revenue held steady despite falling order counts → indicates a premium-pricing effect (fewer orders but higher spend per order).

2. Orders Patterns 🛒

- Order volumes were not consistent month to month.
- Revenue growth largely tracked with order growth, but not always — highlighting changes in customer behavior.
- Declines in order counts during stable revenue months suggest fewer but higher-value customers drove sales.

3. Quantity Dynamics 📦

- Total product quantities sold showed peaks, hinting at promotions or seasonal demand.
- MoM quantity differences exposed volatility in demand — useful for inventory planning.
- Quantity drops in high-revenue months suggest higher sales from premium products rather than volume.

Data Insights

Kap’s Café SQL Project

Recommendations

1. Promotions in Weak Months 🎯

- Launch targeted promotions or seasonal campaigns in historically low-revenue months.
- Use insights from MoM trends to time campaigns strategically.

2. Inventory Optimization 📦

- Align purchasing and stocking decisions with monthly demand shifts.
- Prevent overstocking during slow months and stockouts during demand peaks

3. Loyalty Programs 🤝

- Introduce rewards or discounts for repeat customers to stabilize fluctuating order volumes.
- Leverage customer retention to reduce revenue volatility.

4. Pricing Strategy 💲

- When revenue grows despite fewer orders, explore premium pricing strategies.
- Segment products by value vs. volume to balance profitability and demand.

Data Insights

Kap's Café SQL Project

Conclusion

1. **SQL → Insights**

- SQL is not just a technical tool — it translated raw café transactions into clear, measurable business intelligence.

2. **Visibility Achieved**

- Kap's Café now has visibility into monthly sales, customer orders, and product demand, enabling informed decision-making.

3. From Data to Impact

- The analysis bridged the gap: Data → Decisions → Business Impact.
- Clear trends, patterns, and recommendations give management a roadmap to improve growth and efficiency.

Thank You

md.ahmedtanvirdev@gmail.com

01882060723