DAY 5 – ADVANCED WINDOW FUNCTIONS

Dataset: Superstore Sales Dataset
Tool Used: MySQL
Analysis Type: Advanced Analytical SQL (Window Functions)

1.Objective of Day 5

The objective of Day 5 was to apply advanced SQL window functions to perform ranking, segmentation, running totals, and year-over-year comparisons. These techniques are widely used in real-world analytics and SQL interviews.

2.Why Window Functions Are Important

Window functions allow analysts to:

- Perform calculations across related rows without collapsing data
- Rank and compare records within groups
- Analyze trends and growth over time
- Replace complex subqueries with cleaner logic

Window functions are essential for **business reporting, dashboards, and advanced analytics**.

3.SQL Analysis Performed

3.1 Ranking Products by Profit Within Each Category

```
SELECT
  category,
  sub_category,
  ROUND(SUM(profit),2) AS total_profit,
  DENSE_RANK() OVER (
      PARTITION BY category
      ORDER BY SUM(profit) DESC
  ) AS category_rank
FROM orders
GROUP BY category, sub_category;
```

category	sub_category	total_profit	category_rank
Furniture	Chairs	26590.15	1
Furniture	Furnishings	13059.25	2
Furniture	Bookcases	-3472.56	3
Furniture	Tables	-17725.59	4
Office Supplies	Paper	34053.34	1
Office Supplies	Binders	30221.64	2
Office Supplies	Storage	21279.05	3
Office Supplies	Appliances	18138.07	4
Office Supplies	Envelopes	6964.10	5
Office Supplies	Art	6527.96	6
Office Supplies	Labels	5546.18	7
Office Supplies	Fasteners	949.53	8
Office Supplies	Supplies	-1188.99	9
Technology	Copiers	55617.90	1
Technology	Phones	44516.25	2
Technology	Accessories	41936.78	3
Technology	Machines	3384.73	4

3.2 Top 2 Products per Category (Top-N per Group)

```
SELECT *
FROM (
    SELECT
        category,
        sub_category,
        ROUND(SUM(profit),2) AS total_profit,
        DENSE_RANK() OVER (
            PARTITION BY category
            ORDER BY SUM(profit) DESC
        ) AS rnk
    FROM orders
    GROUP BY category, sub_category
) t
WHERE rnk <= 2;
```

category	sub_category	total_profit	rnk
Furniture	Chairs	26590.15	1
Furniture	Furnishings	13059.25	2
Office Supplies	Paper	34053.34	1
Office Supplies	Binders	30221.64	2
Technology	Copiers	55617.90	1
Technology	Phones	44516.25	2

3.3 Running Total of Sales by Year

```
SELECT
  YEAR(order_date) AS year,
  ROUND(SUM(sales),2) AS yearly_sales,
  ROUND(
      SUM(SUM(sales)) OVER (ORDER BY
YEAR(order_date)),
  2) AS running_sales
FROM orders
GROUP BY YEAR(order_date)
ORDER BY year;
```

year	yearly_sales	running_sales
2014	484247.56	484247.56
2015	470532.46	954780.02
2016	609205.86	1563985.88
2017	733215.19	2297201.07

3.4 Year-Over-Year (YoY) Sales Growth

```
SELECT
  year,
  yearly_sales,
  yearly_sales
    - LAG(yearly_sales) OVER (ORDER BY year) AS yoy_growth
FROM (
    SELECT
      YEAR(order_date) AS year,
      ROUND(SUM(sales),2) AS yearly_sales
    FROM orders
    GROUP BY YEAR(order_date)
) t;
```

year	yearly_sales	yoy_growth
2014	484247.56	NULL
2015	470532.46	-13715.10
2016	609205.86	138673.40
2017	733215.19	124009.33

3.5 Year-Over-Year Percentage Growth

```
SELECT
  year,
  yearly_sales,
  ROUND(
    (yearly_sales - LAG(yearly_sales) OVER (ORDER BY year))
    / LAG(yearly_sales) OVER (ORDER BY year) * 100, 2
  ) AS yoy_growth_percent
FROM (
    SELECT
      YEAR(order_date) AS year,
      SUM(sales) AS yearly_sales
    FROM orders
    GROUP BY YEAR(order_date)
) t;
```

year	yearly_sales	yoy_growth_percent
2014	484247.56	NULL
2015	470532.46	-2.83
2017	733215.19	20.36
2016	609205.86	29.47

3.6 Identifying Decline Years

```
SELECT *
FROM (
    SELECT
      year,
      yearly_sales,
      yearly_sales - LAG(yearly_sales) OVER (ORDER BY year) AS growth
    FROM (
        SELECT
          YEAR(order_date) AS year,
          SUM(sales) AS yearly_sales
        FROM orders
        GROUP BY YEAR(order_date)
    ) y
) g
WHERE growth < 0;
```

year	yearly_sales	growth
2015	470532.46	-13715.10

Key Insights from Day 5

- Window functions provide advanced analytical power.
- Ranking within groups reveals top and bottom performers.
- Running totals and YoY growth help track business performance.
- Window functions reduce the need for complex subqueries.

Business Recommendations

- Focus on top-ranked products within each category.
- Monitor year-over-year growth to identify decline early.
- Use running totals for forecasting and strategic planning.
- Apply window functions in dashboards and management reports.

Key Learnings from Day 5

- Use of `PARTITION BY` for grouped analysis
- Difference between `RANK`, `DENSE_RANK`, and `ROW_NUMBER`
- Application of `LAG()` for time-based comparisons
- Importance of window functions in real-world analytics