SQL Time-Based Sales Analysis Report

1. Objective

The objective of this analysis was to perform a time-based evaluation of sales performance using SQL aggregation techniques. The goal was to identify trends in **monthly revenue** and **order volume**, determine **peak sales periods**, and understand **customer buying patterns** over time.

---

2. Database & Table Structure

Two tables were used:

Orders Table

Contains transactional information such as order date and customer details.
Key columns: OrderID, OrderDate, CustomerName, State, City

Details Table

Contains line-level sales information such as revenue and product categories.
Key columns: OrderID, Amount, Profit, Quantity, Category, SubCategory, PaymentMode

Both tables were linked via OrderID.

---

3. Data Preparation

Before analysis, dates were formatted into proper SQL DATE type and tables were joined on OrderID.
Time elements (Year and Month) were extracted using:

EXTRACT(YEAR FROM OrderDate)
EXTRACT(MONTH FROM OrderDate)

These functions allowed grouping sales results month-wise.

---

4. Monthly Sales Aggregation

To calculate monthly performance, the following metrics were computed:

- **Total Revenue:** SUM(Amount)
- **Order Volume:** COUNT(DISTINCT OrderID)
- **Grouping:** By YEAR and MONTH of the order date
- **Sorting:** Chronologically using ORDER BY Year, Month

Output:

Year	Month	TotalRevenue	TotalOrders
2018	1	61632	61
2018	2	38962	54
2018	3	60694	58
2018	4	34330	44
2018	5	29093	31
2018	6	23658	30
2018	7	12966	31
2018	8	31492	31
2018	9	27283	30

This provided a month-over-month revenue and order count trend.

---

5. Year-Specific Filtering

A focused analysis was performed for the year **2018** using:

WHERE EXTRACT(YEAR FROM OrderDate) = 2018

Output:

Month	TotalRevenue	TotalOrders
1	61632	61
2	38962	54
3	60694	58
4	34330	44
5	29093	31
6	23658	30
7	12966	31
8	31492	31
9	27283	30

This helped isolate monthly behavior within a single year.

---

6. Peak Sales Identification

Top-performing months were identified using:

```
ORDER BY TotalRevenue DESC
LIMIT 5
```

Output:

MonthYear	TotalRevenue
2018-01	61632
2018-03	60694
2018-11	48469
2018-02	38962
2018-12	37579

This highlighted periods with the highest sales activity.

---

7. Formatted Time Output

To make the report more readable, dates were formatted as:

```
DATE_FORMAT(OrderDate, '%Y-%m')
```

Output:

MonthYear	Revenue	Orders
2018-01	61632	61
2018-02	38962	54
2018-03	60694	58
2018-04	34330	44
2018-05	29093	31
2018-06	23658	30
2018-07	12966	31
2018-08	31492	31
2018-09	27283	30

This produced clean month-year labels such as *2018-03*, *2018-08*, etc.

---

8. Key Insights

Based on the executed SQL queries:

✔ **Monthly Revenue Trends**

- Revenue fluctuated across months, showing clear peaks and lows.
- Certain months consistently generated higher sales due to increased order volume.

✔ **Order Volume Patterns**

- Order count closely followed revenue trends.
- High-revenue months typically matched high customer activity.

✔ **Peak Sales Months**

- Using revenue sorting, top months were identified as the peak performance periods.
- These months represent the strongest customer demand cycles.

✔ **Customer Buying Behavior**

- Sales showed variations between quarters, revealing seasonality in purchases.
- Categories such as Electronics, Phones, and Furniture often dominated high-performing months.

---

9. Conclusion

The SQL-based time-series analysis successfully revealed monthly trends in revenue and order volume.
Using aggregation, filtering, grouping, and date formatting, peak sales periods were identified, enabling valuable insights into customer buying patterns.

This analysis can help in business planning, forecasting, inventory management, and marketing strategies.