

# ELEVATE LABS

## DATA ANALYST INTERNSHIP

### TASK – 6

The database is customer which has a table customer

- Using 'select' we can view the customer table

```
mysql> use customer;
Database changed
mysql> select * from customer limit 5;
```

CustomerID	CustomerName	City	Product	Quantity	Price	TotalAmount	OrderDate	ProductID
1	Rati Thakkar	Mumbai	Smartphone	5	42949	214745	2024-12-10	P001
2	Miraan Dewan	Mumbai	Camera	3	15247	45741	2024-06-15	P002
3	Elakshi Sampath	Bangalore	Headphones	4	34454	137816	2025-01-23	P003
4	Vivaan Kara	Hyderabad	Headphones	4	21190	84760	2024-07-13	P003
5	Anahita Bhavsar	Mumbai	Headphones	3	64732	194196	2024-09-09	P003

5 rows in set (0.00 sec)

- Monthly sales trend

```
mysql> SELECT
->     EXTRACT(YEAR FROM OrderDate) AS order_year,
->     EXTRACT(MONTH FROM OrderDate) AS order_month,
->     SUM(TotalAmount) AS total_revenue,
->     COUNT(DISTINCT CustomerID) AS order_volume
-> FROM customer
-> WHERE OrderDate BETWEEN '2024-01-01' AND '2024-12-31'
-> GROUP BY order_year, order_month
-> ORDER BY order_year, order_month;
```

order_year	order_month	total_revenue	order_volume
2024	4	204351	3
2024	5	376451	6
2024	6	1155551	11
2024	7	471718	6
2024	8	387605	6
2024	9	1159122	9
2024	10	1352353	11
2024	11	1708192	13
2024	12	815130	8

9 rows in set (0.00 sec)

;

- Time Series & Trend Analysis (Monthly average order value)

```
mysql> SELECT
->   EXTRACT(MONTH FROM OrderDate) AS Month,
->   AVG(TotalAmount) AS AvgOrderValue
-> FROM customer
-> GROUP BY Month;
```

Month	AvgOrderValue
12	101891.2500
6	105050.0909
1	152038.7143
7	78619.6667
9	128791.3333
3	148513.6364
11	131399.3846
10	122941.1818
2	130369.3333
8	64600.8333
5	62741.8333
4	68117.0000

12 rows in set (0.01 sec)

- Product Performance (Best-selling products by revenue or quantity)

```
mysql> SELECT Product, SUM(Quantity) AS TotalSold, SUM(TotalAmount) AS Revenue
-> FROM customer
-> GROUP BY Product
-> ORDER BY Revenue DESC;
```

Product	TotalSold	Revenue
Smartphone	53	2296828
Laptop	63	2235220
Headphones	51	1938520
Smartwatch	56	1720940
Tablet	43	1696196
Camera	49	1614014

6 rows in set (0.01 sec)

## Location-Based Insights

- Top cities by sales:

```
mysql> SELECT City, SUM(TotalAmount) AS Revenue
-> FROM customer
-> GROUP BY City
-> ORDER BY Revenue DESC;
+-----+-----+
| City      | Revenue |
+-----+-----+
| Bangalore | 2095631 |
| Kolkata   | 1888761 |
| Chennai   | 1818675 |
| Mumbai    | 1438962 |
| Pune      | 1338937 |
| Ahmedabad | 1319074 |
| Delhi     | 928465  |
| Hyderabad | 673213  |
+-----+-----+
8 rows in set (0.00 sec)
```

## Customer Behaviour

- High-value customers:

```
mysql> SELECT CustomerID, CustomerName, SUM(TotalAmount) AS TotalSpent
-> FROM customer
-> GROUP BY CustomerID, CustomerName
-> ORDER BY TotalSpent DESC
-> LIMIT 10;
+-----+-----+-----+
| CustomerID | CustomerName | TotalSpent |
+-----+-----+-----+
| 66         | Mohanlal Borde | 345115     |
| 58         | Manjari Wagle  | 318805     |
| 90         | Priyansh Khare | 313840     |
| 76         | Jhanvi Goyal   | 301415     |
| 47         | Rania Varughese | 279850     |
| 32         | Samaira Kale   | 273990     |
| 15         | Jayan Goel     | 266948     |
| 54         | Nakul Dey      | 257295     |
| 67         | Mohanlal Warrior | 254275     |
| 41         | Miraya Talwar   | 237150     |
+-----+-----+-----+
10 rows in set (0.00 sec)

mysql> SELECT QUARTER(OrderDate) AS Quarter, SUM(TotalAmount) AS Revenue
```

## Seasonal Patterns

- Sales by quarter:

```
mysql> SELECT QUARTER(OrderDate) AS Quarter, SUM(TotalAmount) AS Revenue
-> FROM customer
-> GROUP BY Quarter;
+-----+-----+
| Quarter | Revenue |
+-----+-----+
|      4 | 3875675 |
|      2 | 1736353 |
|      1 | 3871245 |
|      3 | 2018445 |
+-----+-----+
4 rows in set (0.01 sec)
```