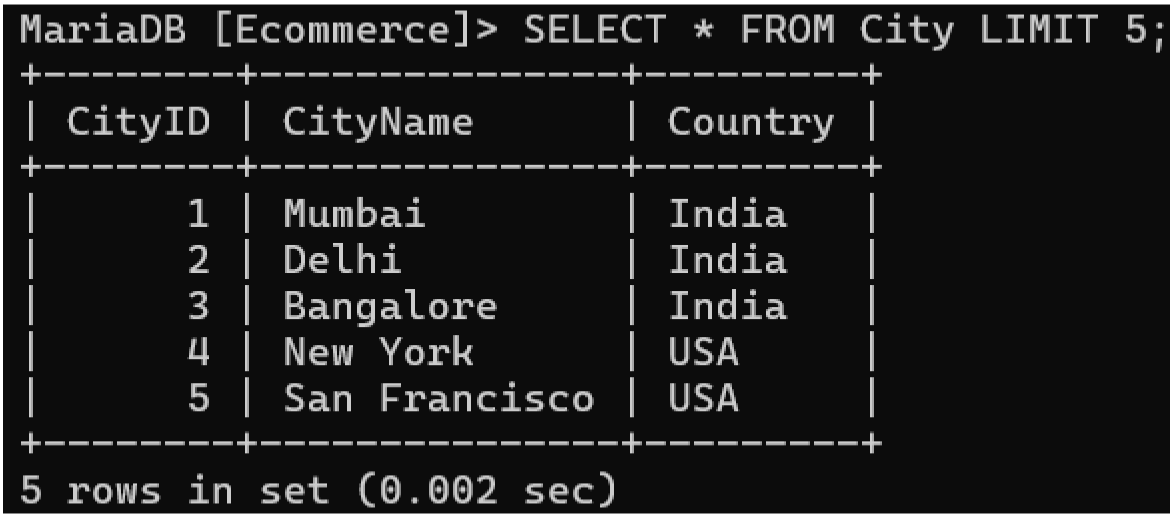
Task 4: SQL for Data Analysis

Step 1: Understand Your Database

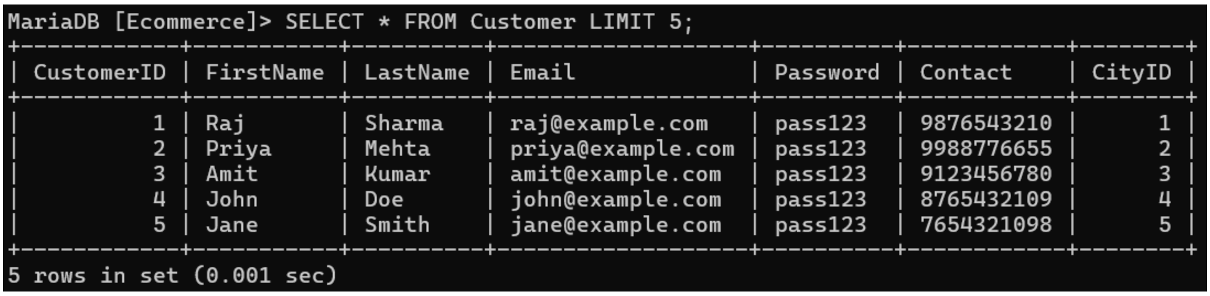
- **Preview your data** (first 5 rows from each table):

SELECT * FROM City LIMIT 5;



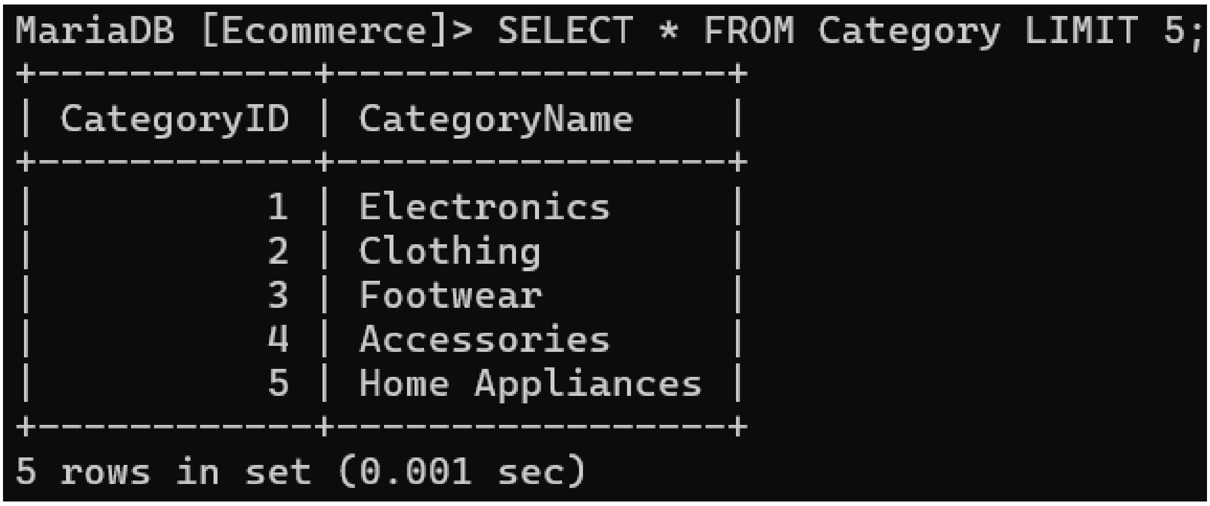
```
MariaDB [Ecommerce]> SELECT * FROM City LIMIT 5;
+--------+---------------+---------+
| CityID | CityName      | Country |
+--------+---------------+---------+
|      1 | Mumbai        | India   |
|      2 | Delhi         | India   |
|      3 | Bangalore     | India   |
|      4 | New York      | USA     |
|      5 | San Francisco | USA     |
+--------+---------------+---------+
5 rows in set (0.002 sec)
```

SELECT * FROM Customer LIMIT 5;



```
MariaDB [Ecommerce]> SELECT * FROM Customer LIMIT 5;
+------------+-----------+----------+-------------------+----------+------------+--------+
| CustomerID | FirstName | LastName | Email             | Password | Contact    | CityID |
+------------+-----------+----------+-------------------+----------+------------+--------+
|          1 | Raj       | Sharma   | raj@example.com   | pass123  | 9876543210 |      1 |
|          2 | Priya     | Mehta    | priya@example.com | pass123  | 9988776655 |      2 |
|          3 | Amit      | Kumar    | amit@example.com  | pass123  | 9123456780 |      3 |
|          4 | John      | Doe      | john@example.com  | pass123  | 8765432109 |      4 |
|          5 | Jane      | Smith    | jane@example.com  | pass123  | 7654321098 |      5 |
+------------+-----------+----------+-------------------+----------+------------+--------+
5 rows in set (0.001 sec)
```

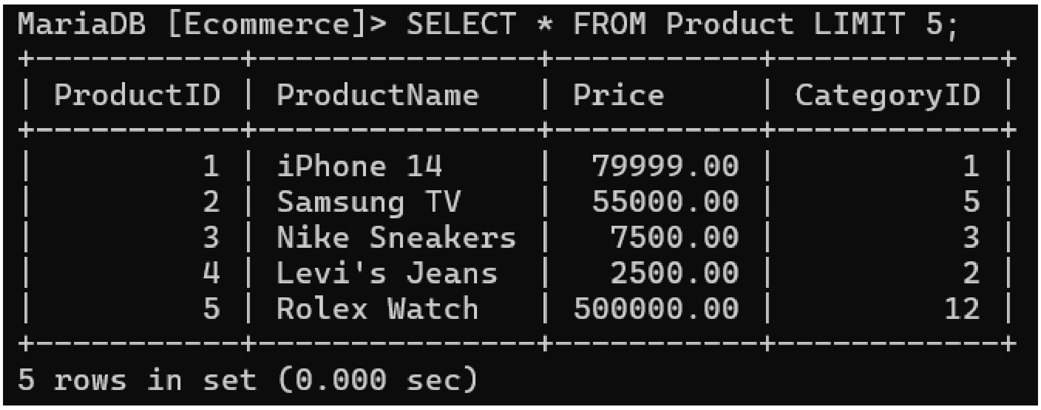
SELECT * FROM Category LIMIT 5;



```
MariaDB [Ecommerce]> SELECT * FROM Category LIMIT 5;
+------------+-----------------+
| CategoryID | CategoryName    |
+------------+-----------------+
|          1 | Electronics     |
|          2 | Clothing        |
|          3 | Footwear        |
|          4 | Accessories     |
|          5 | Home Appliances |
+------------+-----------------+
5 rows in set (0.001 sec)
```

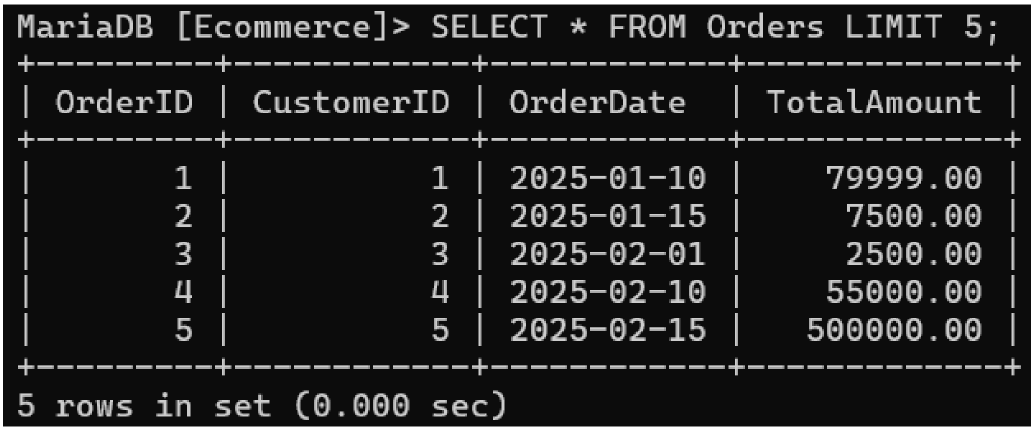
Task 4: SQL for Data Analysis

SELECT * FROM Product LIMIT 5;



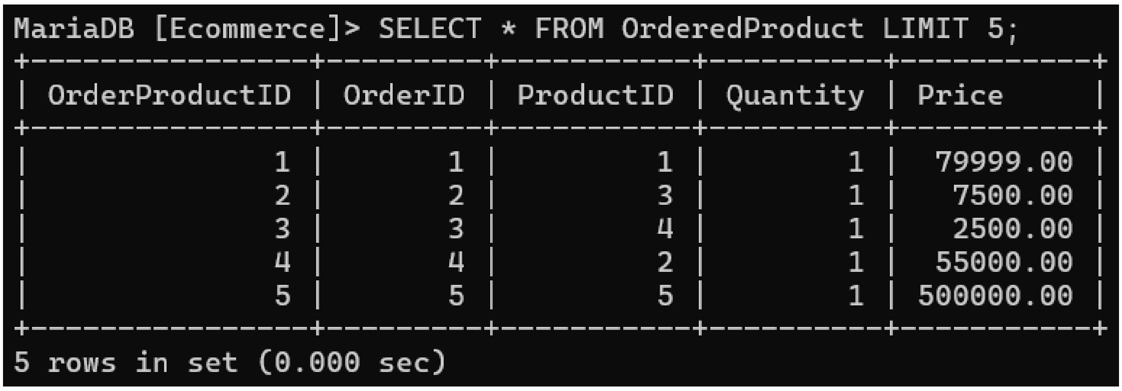
```
MariaDB [Ecommerce]> SELECT * FROM Product LIMIT 5;
+-----------+---------------+-----------+------------+
| ProductID | ProductName   | Price     | CategoryID |
+-----------+---------------+-----------+------------+
|         1 | iPhone 14     |  79999.00 |          1 |
|         2 | Samsung TV    |  55000.00 |          5 |
|         3 | Nike Sneakers |   7500.00 |          3 |
|         4 | Levi's Jeans  |   2500.00 |          2 |
|         5 | Rolex Watch   | 500000.00 |         12 |
+-----------+---------------+-----------+------------+
5 rows in set (0.000 sec)
```

SELECT * FROM Orders LIMIT 5;



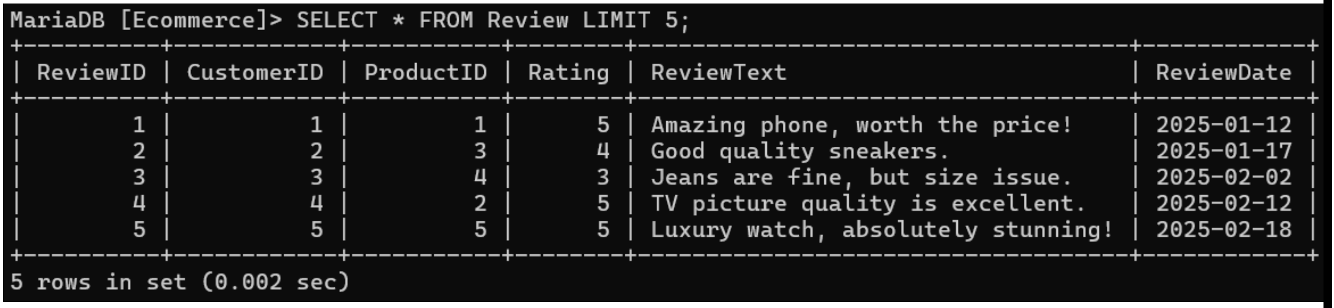
```
MariaDB [Ecommerce]> SELECT * FROM Orders LIMIT 5;
+---------+------------+------------+-------------+
| OrderID | CustomerID | OrderDate  | TotalAmount |
+---------+------------+------------+-------------+
|       1 |          1 | 2025-01-10 |    79999.00 |
|       2 |          2 | 2025-01-15 |     7500.00 |
|       3 |          3 | 2025-02-01 |     2500.00 |
|       4 |          4 | 2025-02-10 |    55000.00 |
|       5 |          5 | 2025-02-15 |   500000.00 |
+---------+------------+------------+-------------+
5 rows in set (0.000 sec)
```

SELECT * FROM OrderedProduct LIMIT 5;



```
MariaDB [Ecommerce]> SELECT * FROM OrderedProduct LIMIT 5;
+----------------+---------+-----------+----------+-----------+
| OrderProductID | OrderID | ProductID | Quantity | Price     |
+----------------+---------+-----------+----------+-----------+
|              1 |       1 |         1 |        1 |  79999.00 |
|              2 |       2 |         3 |        1 |   7500.00 |
|              3 |       3 |         4 |        1 |   2500.00 |
|              4 |       4 |         2 |        1 |  55000.00 |
|              5 |       5 |         5 |        1 | 500000.00 |
+----------------+---------+-----------+----------+-----------+
5 rows in set (0.000 sec)
```

SELECT * FROM Review LIMIT 5;

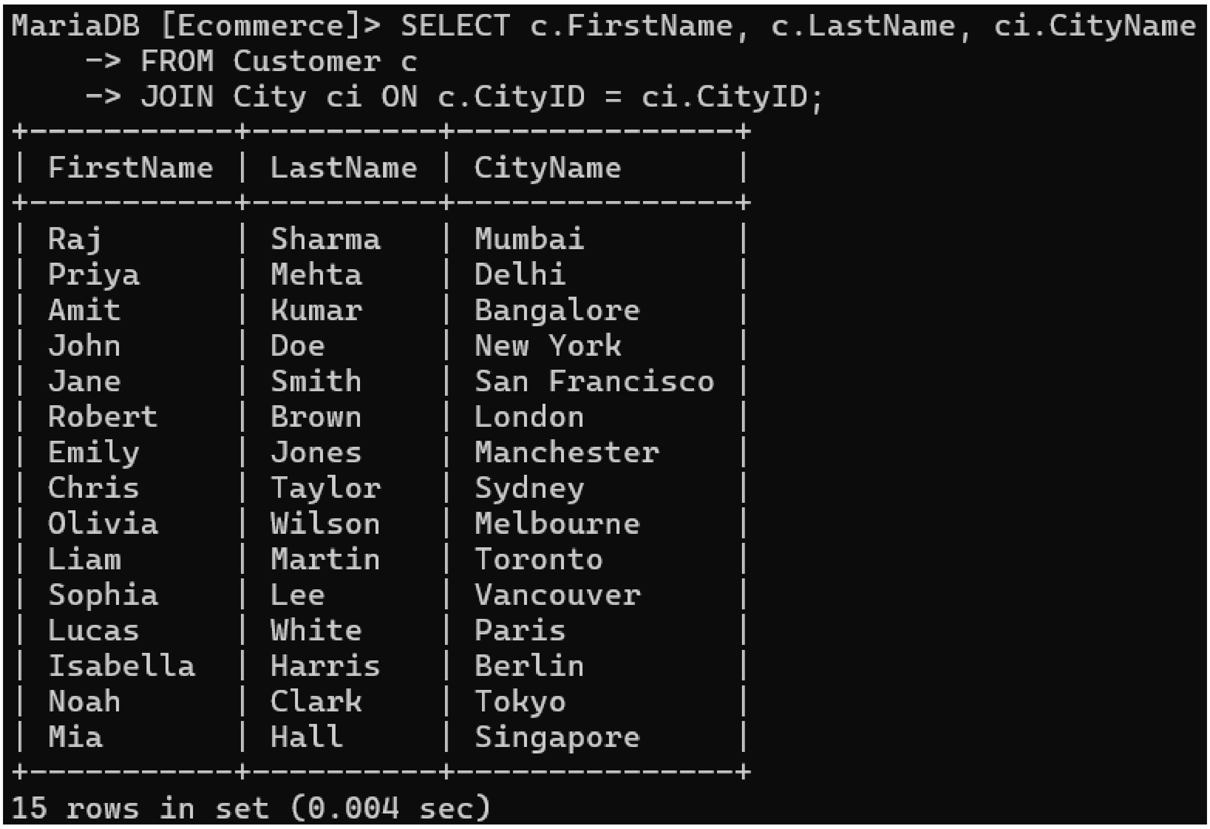


```
MariaDB [Ecommerce]> SELECT * FROM Review LIMIT 5;
+----------+------------+-----------+--------+------------------------------------+------------+
| ReviewID | CustomerID | ProductID | Rating | ReviewText                         | ReviewDate |
+----------+------------+-----------+--------+------------------------------------+------------+
|        1 |          1 |         1 |      5 | Amazing phone, worth the price!    | 2025-01-12 |
|        2 |          2 |         3 |      4 | Good quality sneakers.             | 2025-01-17 |
|        3 |          3 |         4 |      3 | Jeans are fine, but size issue.    | 2025-02-02 |
|        4 |          4 |         2 |      5 | TV picture quality is excellent.   | 2025-02-12 |
|        5 |          5 |         5 |      5 | Luxury watch, absolutely stunning! | 2025-02-18 |
+----------+------------+-----------+--------+------------------------------------+------------+
5 rows in set (0.002 sec)
```

Task 4: SQL for Data Analysis

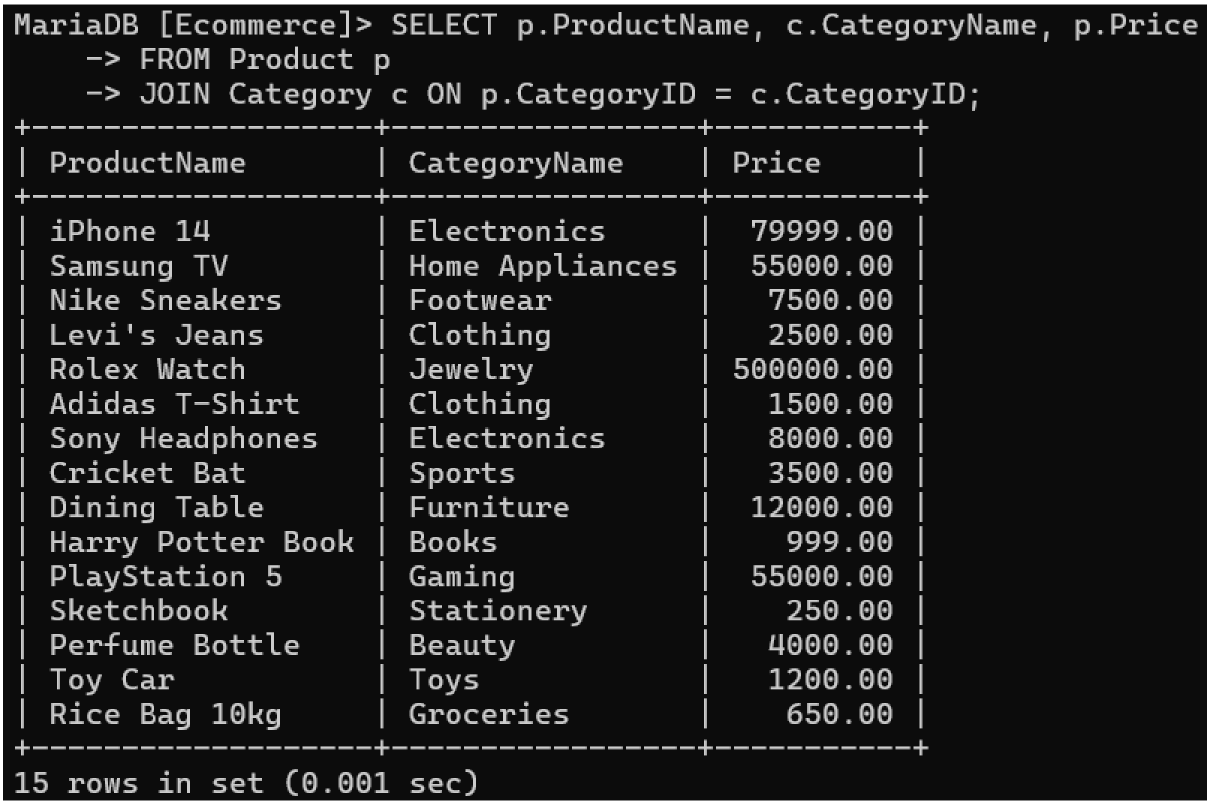
Step 2: Basic SELECT Queries

- **List all customers with their city:**



```
MariaDB [Ecommerce]> SELECT c.FirstName, c.LastName, ci.CityName
    -> FROM Customer c
    -> JOIN City ci ON c.CityID = ci.CityID;
+-----------+----------+---------------+
| FirstName | LastName | CityName      |
+-----------+----------+---------------+
| Raj       | Sharma   | Mumbai        |
| Priya     | Mehta    | Delhi         |
| Amit      | Kumar    | Bangalore     |
| John      | Doe      | New York      |
| Jane      | Smith    | San Francisco |
| Robert    | Brown    | London        |
| Emily     | Jones    | Manchester    |
| Chris     | Taylor   | Sydney        |
| Olivia    | Wilson   | Melbourne     |
| Liam      | Martin   | Toronto       |
| Sophia    | Lee      | Vancouver     |
| Lucas     | White    | Paris         |
| Isabella  | Harris   | Berlin        |
| Noah      | Clark    | Tokyo         |
| Mia       | Hall     | Singapore     |
+-----------+----------+---------------+
15 rows in set (0.004 sec)
```

- **List all products and their categories:**

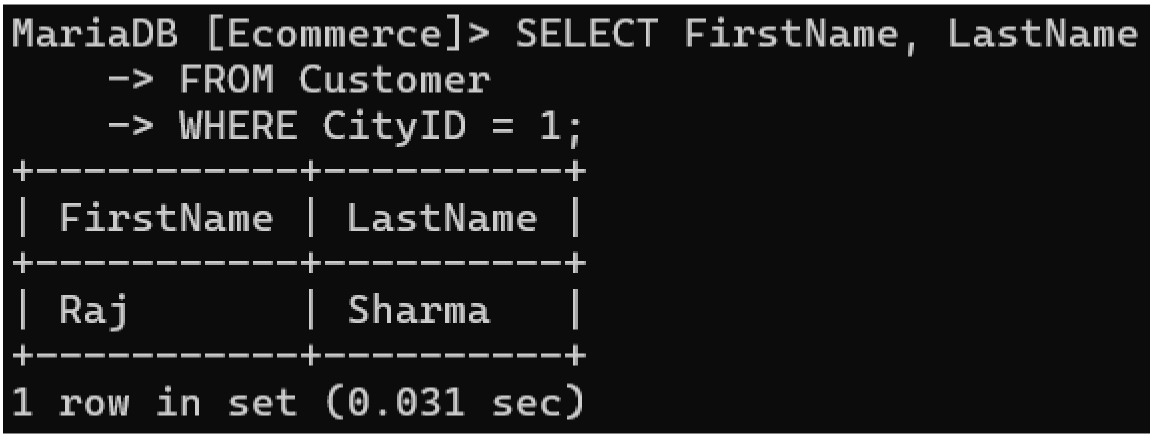


```
MariaDB [Ecommerce]> SELECT p.ProductName, c.CategoryName, p.Price
    -> FROM Product p
    -> JOIN Category c ON p.CategoryID = c.CategoryID;
+-------------------+-----------------+-----------+
| ProductName       | CategoryName    | Price     |
+-------------------+-----------------+-----------+
| iPhone 14         | Electronics     |  79999.00 |
| Samsung TV        | Home Appliances |  55000.00 |
| Nike Sneakers     | Footwear        |   7500.00 |
| Levi's Jeans      | Clothing        |   2500.00 |
| Rolex Watch       | Jewelry         | 500000.00 |
| Adidas T-Shirt    | Clothing        |   1500.00 |
| Sony Headphones   | Electronics     |   8000.00 |
| Cricket Bat       | Sports          |   3500.00 |
| Dining Table      | Furniture       |  12000.00 |
| Harry Potter Book | Books           |    999.00 |
| PlayStation 5     | Gaming          |  55000.00 |
| Sketchbook        | Stationery      |    250.00 |
| Perfume Bottle    | Beauty          |   4000.00 |
| Toy Car           | Toys            |   1200.00 |
| Rice Bag 10kg     | Groceries       |    650.00 |
+-------------------+-----------------+-----------+
15 rows in set (0.001 sec)
```

Task 4: SQL for Data Analysis

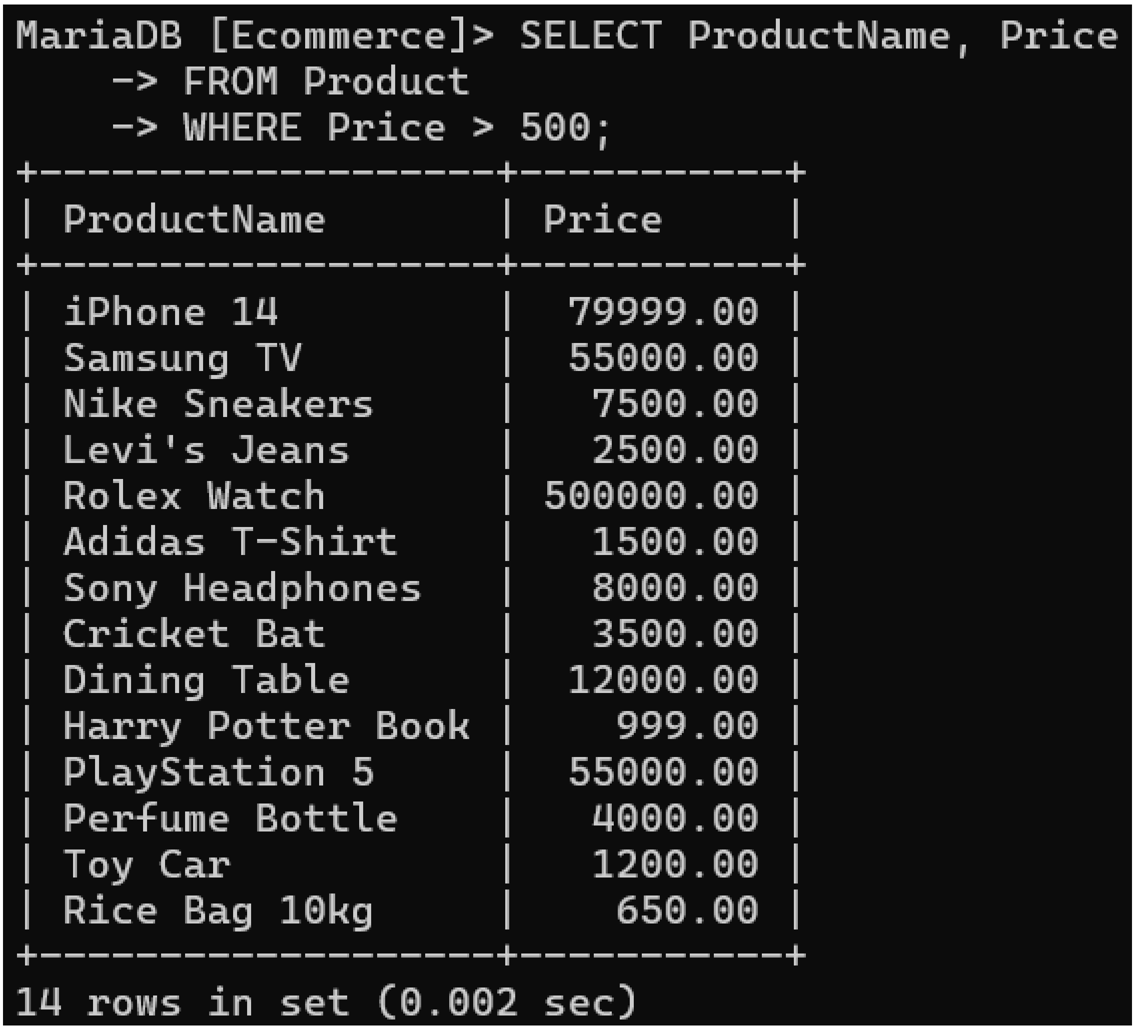
Step 3: Filtering Data (WHERE Clause)

- **Customers in a specific city:**



```
MariaDB [Ecommerce]> SELECT FirstName, LastName
    -> FROM Customer
    -> WHERE CityID = 1;
+-----------+----------+
| FirstName | LastName |
+-----------+----------+
| Raj       | Sharma   |
+-----------+----------+
1 row in set (0.031 sec)
```

- **Products above a certain price:**

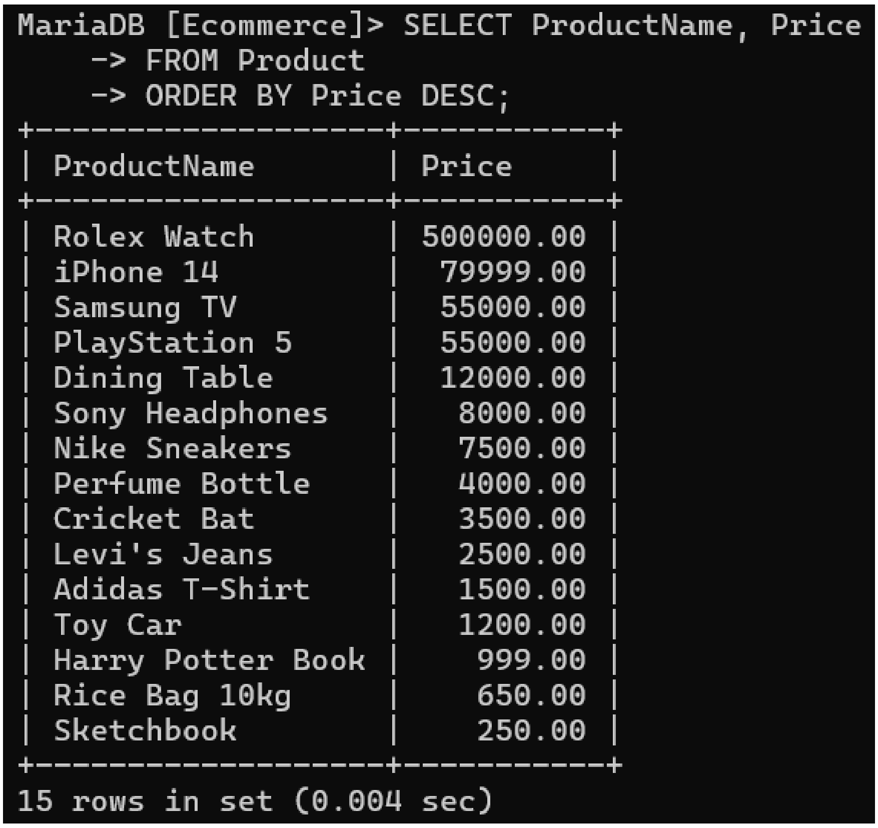


```
MariaDB [Ecommerce]> SELECT ProductName, Price
    -> FROM Product
    -> WHERE Price > 500;
+-------------------+-----------+
| ProductName       | Price     |
+-------------------+-----------+
| iPhone 14         |  79999.00 |
| Samsung TV        |  55000.00 |
| Nike Sneakers     |   7500.00 |
| Levi's Jeans      |   2500.00 |
| Rolex Watch       | 500000.00 |
| Adidas T-Shirt    |   1500.00 |
| Sony Headphones   |   8000.00 |
| Cricket Bat       |   3500.00 |
| Dining Table      |  12000.00 |
| Harry Potter Book |    999.00 |
| PlayStation 5     |  55000.00 |
| Perfume Bottle    |   4000.00 |
| Toy Car           |   1200.00 |
| Rice Bag 10kg     |    650.00 |
+-------------------+-----------+
14 rows in set (0.002 sec)
```

Task 4: SQL for Data Analysis

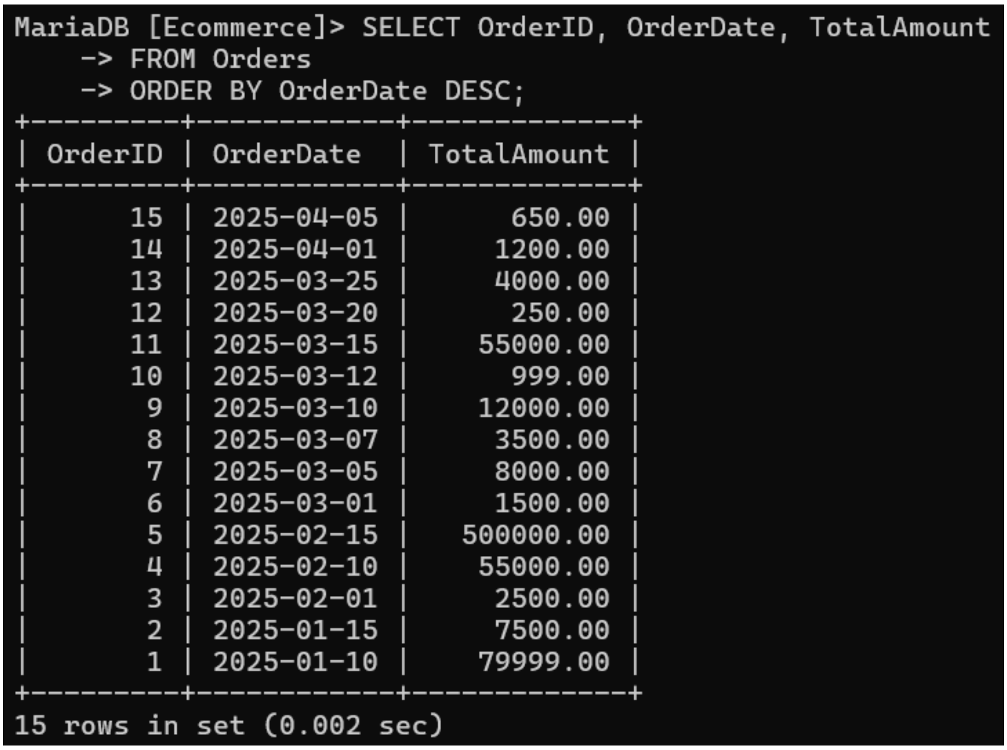
Step 4: Sorting Data (ORDER BY)

- **List products from highest to lowest price:**



```
MariaDB [Ecommerce]> SELECT ProductName, Price
    -> FROM Product
    -> ORDER BY Price DESC;
+-------------------+-----------+
| ProductName       | Price     |
+-------------------+-----------+
| Rolex Watch       | 500000.00 |
| iPhone 14         |  79999.00 |
| Samsung TV        |  55000.00 |
| PlayStation 5     |  55000.00 |
| Dining Table      |  12000.00 |
| Sony Headphones   |   8000.00 |
| Nike Sneakers     |   7500.00 |
| Perfume Bottle    |   4000.00 |
| Cricket Bat       |   3500.00 |
| Levi's Jeans      |   2500.00 |
| Adidas T-Shirt    |   1500.00 |
| Toy Car           |   1200.00 |
| Harry Potter Book |    999.00 |
| Rice Bag 10kg     |    650.00 |
| Sketchbook        |    250.00 |
+-------------------+-----------+
15 rows in set (0.004 sec)
```

- **Orders by most recent date:**

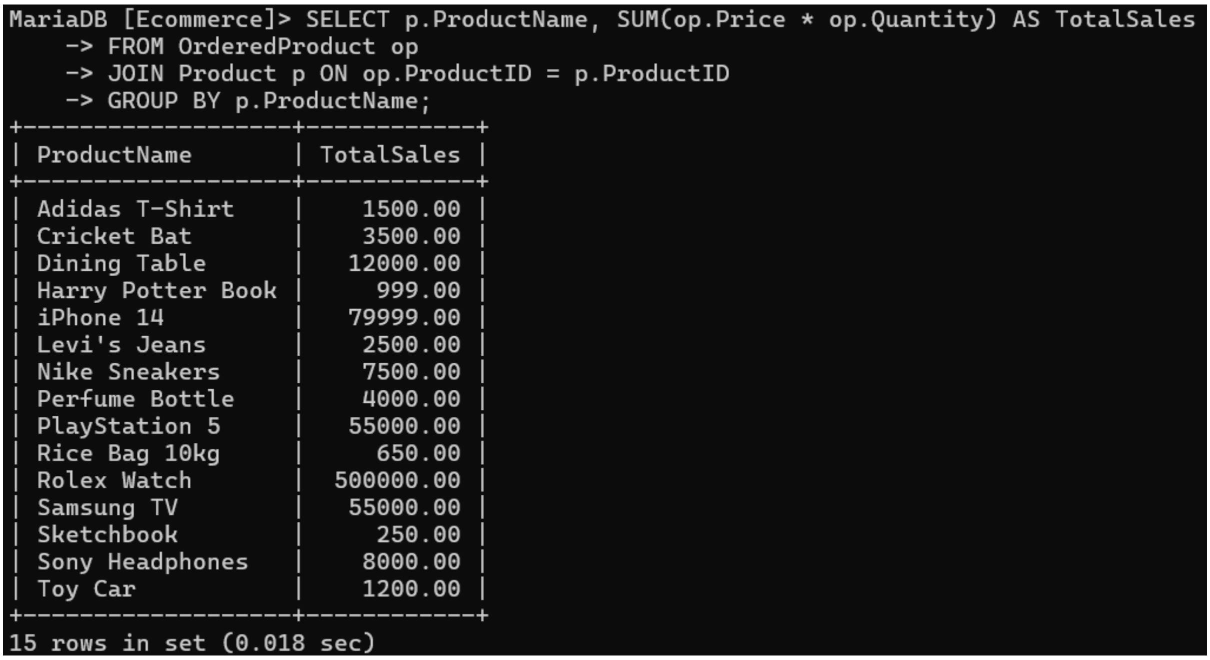


```
MariaDB [Ecommerce]> SELECT OrderID, OrderDate, TotalAmount
    -> FROM Orders
    -> ORDER BY OrderDate DESC;
+---------+------------+-------------+
| OrderID | OrderDate  | TotalAmount |
+---------+------------+-------------+
|      15 | 2025-04-05 |      650.00 |
|      14 | 2025-04-01 |     1200.00 |
|      13 | 2025-03-25 |     4000.00 |
|      12 | 2025-03-20 |      250.00 |
|      11 | 2025-03-15 |    55000.00 |
|      10 | 2025-03-12 |      999.00 |
|       9 | 2025-03-10 |    12000.00 |
|       8 | 2025-03-07 |     3500.00 |
|       7 | 2025-03-05 |     8000.00 |
|       6 | 2025-03-01 |     1500.00 |
|       5 | 2025-02-15 |   500000.00 |
|       4 | 2025-02-10 |    55000.00 |
|       3 | 2025-02-01 |     2500.00 |
|       2 | 2025-01-15 |     7500.00 |
|       1 | 2025-01-10 |    79999.00 |
+---------+------------+-------------+
15 rows in set (0.002 sec)
```

Task 4: SQL for Data Analysis

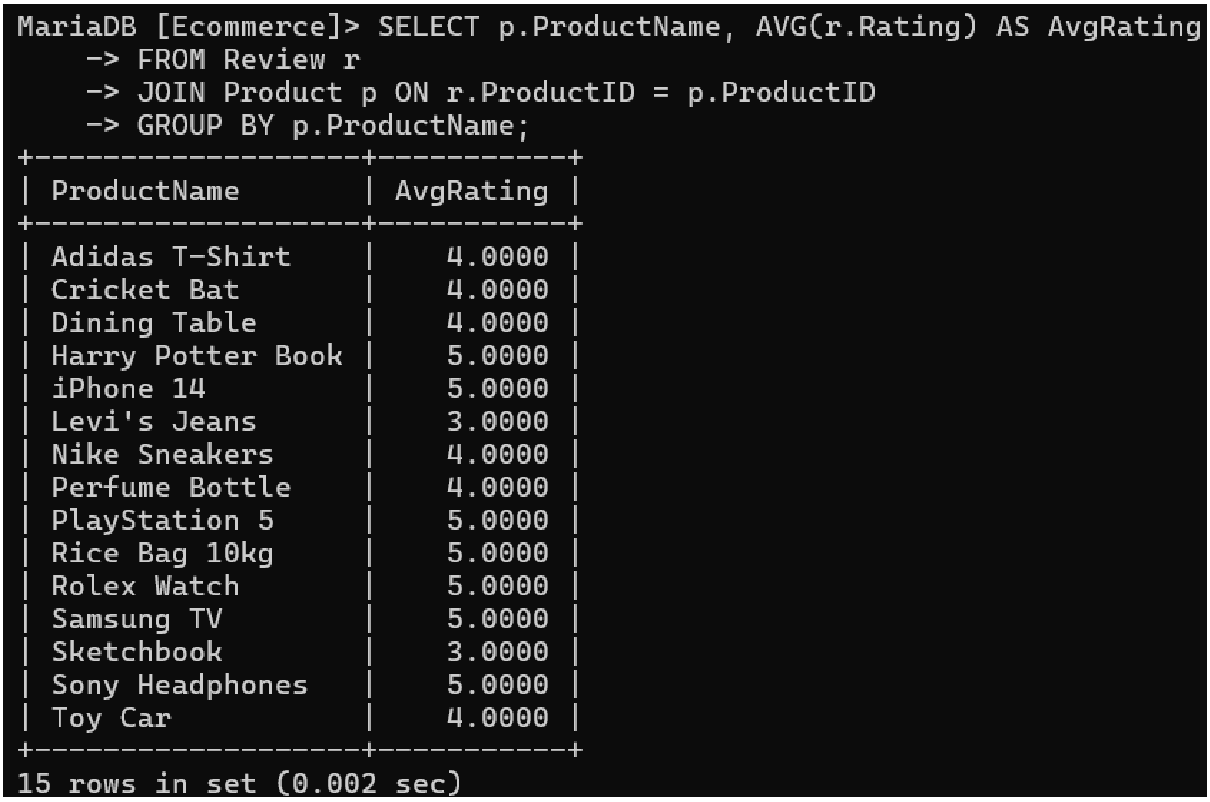
Step 5: Aggregation (GROUP BY & Aggregate Functions)

- **Total sales per product:**



```
MariaDB [Ecommerce]> SELECT p.ProductName, SUM(op.Price * op.Quantity) AS TotalSales
    -> FROM OrderedProduct op
    -> JOIN Product p ON op.ProductID = p.ProductID
    -> GROUP BY p.ProductName;
+-------------------+------------+
| ProductName       | TotalSales |
+-------------------+------------+
| Adidas T-Shirt    |    1500.00 |
| Cricket Bat       |    3500.00 |
| Dining Table      |   12000.00 |
| Harry Potter Book |     999.00 |
| iPhone 14         |   79999.00 |
| Levi's Jeans      |    2500.00 |
| Nike Sneakers     |    7500.00 |
| Perfume Bottle    |    4000.00 |
| PlayStation 5     |   55000.00 |
| Rice Bag 10kg     |     650.00 |
| Rolex Watch       |  500000.00 |
| Samsung TV        |   55000.00 |
| Sketchbook        |     250.00 |
| Sony Headphones   |    8000.00 |
| Toy Car           |    1200.00 |
+-------------------+------------+
15 rows in set (0.018 sec)
```

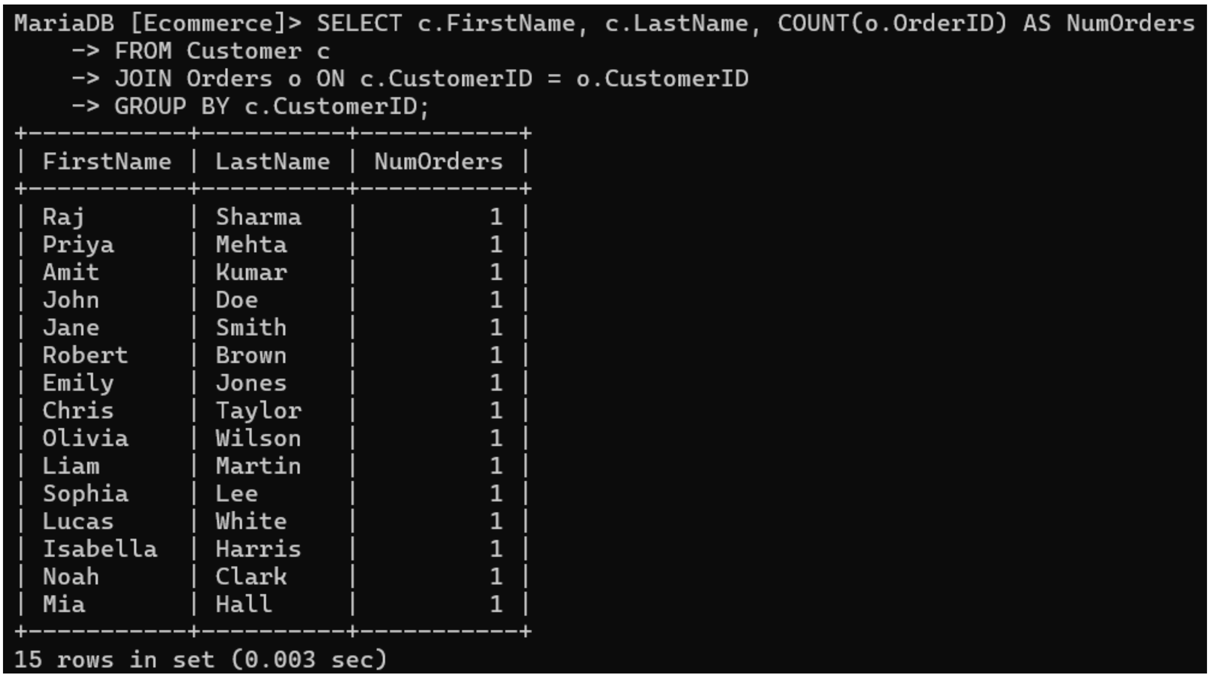
- **Average rating per product:**



```
MariaDB [Ecommerce]> SELECT p.ProductName, AVG(r.Rating) AS AvgRating
    -> FROM Review r
    -> JOIN Product p ON r.ProductID = p.ProductID
    -> GROUP BY p.ProductName;
+-------------------+-----------+
| ProductName       | AvgRating |
+-------------------+-----------+
| Adidas T-Shirt    |    4.0000 |
| Cricket Bat       |    4.0000 |
| Dining Table      |    4.0000 |
| Harry Potter Book |    5.0000 |
| iPhone 14         |    5.0000 |
| Levi's Jeans      |    3.0000 |
| Nike Sneakers     |    4.0000 |
| Perfume Bottle    |    4.0000 |
| PlayStation 5     |    5.0000 |
| Rice Bag 10kg     |    5.0000 |
| Rolex Watch       |    5.0000 |
| Samsung TV        |    5.0000 |
| Sketchbook        |    3.0000 |
| Sony Headphones   |    5.0000 |
| Toy Car           |    4.0000 |
+-------------------+-----------+
15 rows in set (0.002 sec)
```

Task 4: SQL for Data Analysis

- **Number of orders per customer:**

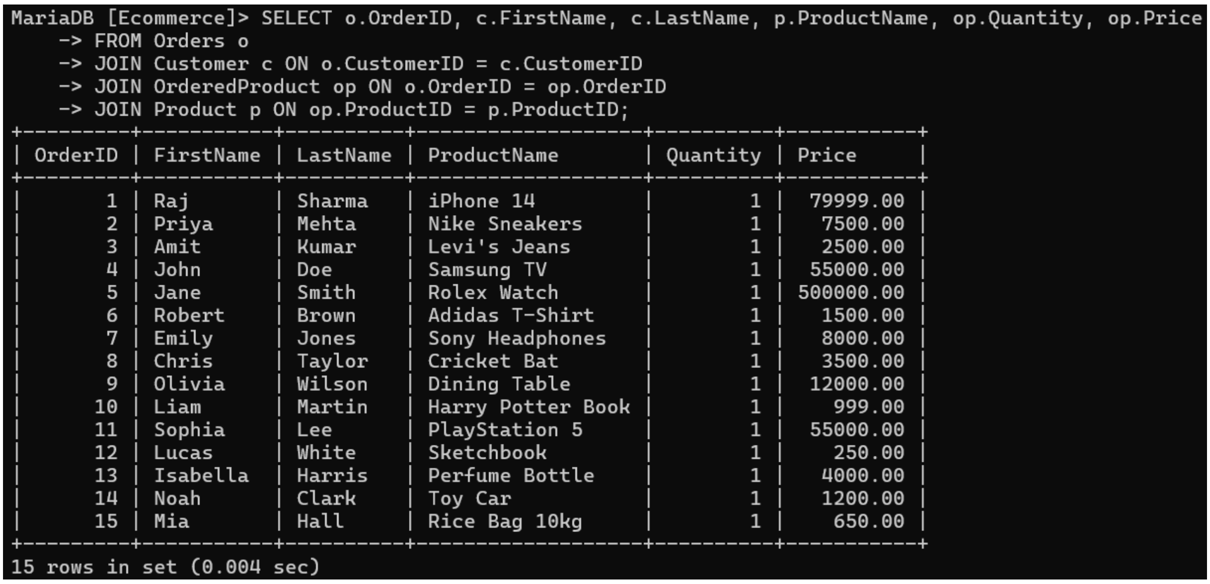


```
MariaDB [Ecommerce]> SELECT c.FirstName, c.LastName, COUNT(o.OrderID) AS NumOrders
    -> FROM Customer c
    -> JOIN Orders o ON c.CustomerID = o.CustomerID
    -> GROUP BY c.CustomerID;
+-----------+----------+-----------+
| FirstName | LastName | NumOrders |
+-----------+----------+-----------+
| Raj       | Sharma   |         1 |
| Priya     | Mehta    |         1 |
| Amit      | Kumar    |         1 |
| John      | Doe      |         1 |
| Jane      | Smith    |         1 |
| Robert    | Brown    |         1 |
| Emily     | Jones    |         1 |
| Chris     | Taylor   |         1 |
| Olivia    | Wilson   |         1 |
| Liam      | Martin   |         1 |
| Sophia    | Lee      |         1 |
| Lucas     | White    |         1 |
| Isabella  | Harris   |         1 |
| Noah      | Clark    |         1 |
| Mia       | Hall     |         1 |
+-----------+----------+-----------+
15 rows in set (0.003 sec)
```

Task 4: SQL for Data Analysis

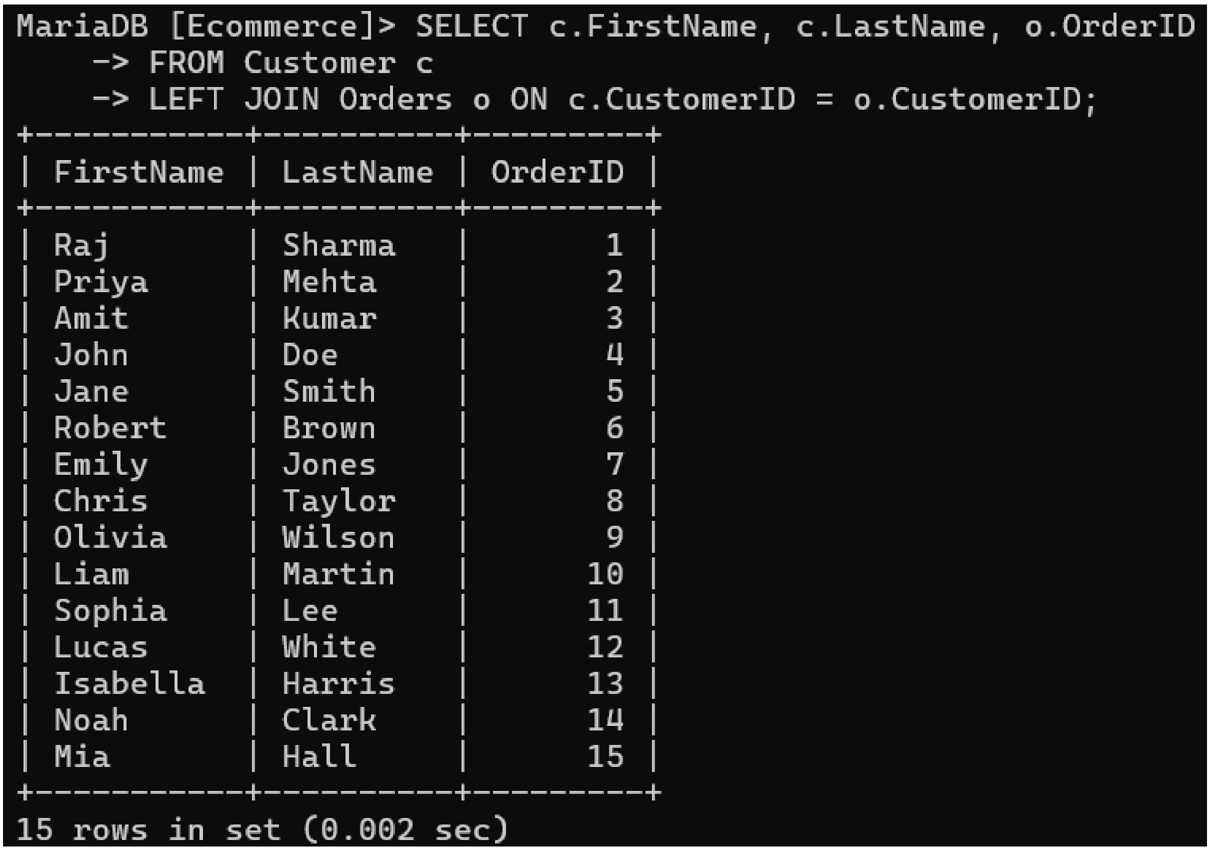
Step 6: Joins

- **List orders with product details and customer names:**



```
MariaDB [Ecommerce]> SELECT o.OrderID, c.FirstName, c.LastName, p.ProductName, op.Quantity, op.Price
    -> FROM Orders o
    -> JOIN Customer c ON o.CustomerID = c.CustomerID
    -> JOIN OrderedProduct op ON o.OrderID = op.OrderID
    -> JOIN Product p ON op.ProductID = p.ProductID;
+---------+-----------+----------+-------------------+----------+-----------+
| OrderID | FirstName | LastName | ProductName       | Quantity | Price     |
+---------+-----------+----------+-------------------+----------+-----------+
|       1 | Raj       | Sharma   | iPhone 14         |        1 |  79999.00 |
|       2 | Priya     | Mehta    | Nike Sneakers     |        1 |   7500.00 |
|       3 | Amit      | Kumar    | Levi's Jeans      |        1 |   2500.00 |
|       4 | John      | Doe      | Samsung TV        |        1 |  55000.00 |
|       5 | Jane      | Smith    | Rolex Watch       |        1 | 500000.00 |
|       6 | Robert    | Brown    | Adidas T-Shirt    |        1 |   1500.00 |
|       7 | Emily     | Jones    | Sony Headphones   |        1 |   8000.00 |
|       8 | Chris     | Taylor   | Cricket Bat       |        1 |   3500.00 |
|       9 | Olivia    | Wilson   | Dining Table      |        1 |  12000.00 |
|      10 | Liam      | Martin   | Harry Potter Book |        1 |    999.00 |
|      11 | Sophia    | Lee      | PlayStation 5     |        1 |  55000.00 |
|      12 | Lucas     | White    | Sketchbook        |        1 |    250.00 |
|      13 | Isabella  | Harris   | Perfume Bottle    |        1 |   4000.00 |
|      14 | Noah      | Clark    | Toy Car           |        1 |   1200.00 |
|      15 | Mia       | Hall     | Rice Bag 10kg     |        1 |    650.00 |
+---------+-----------+----------+-------------------+----------+-----------+
15 rows in set (0.004 sec)
```

- **Left join to see customers with or without orders:**

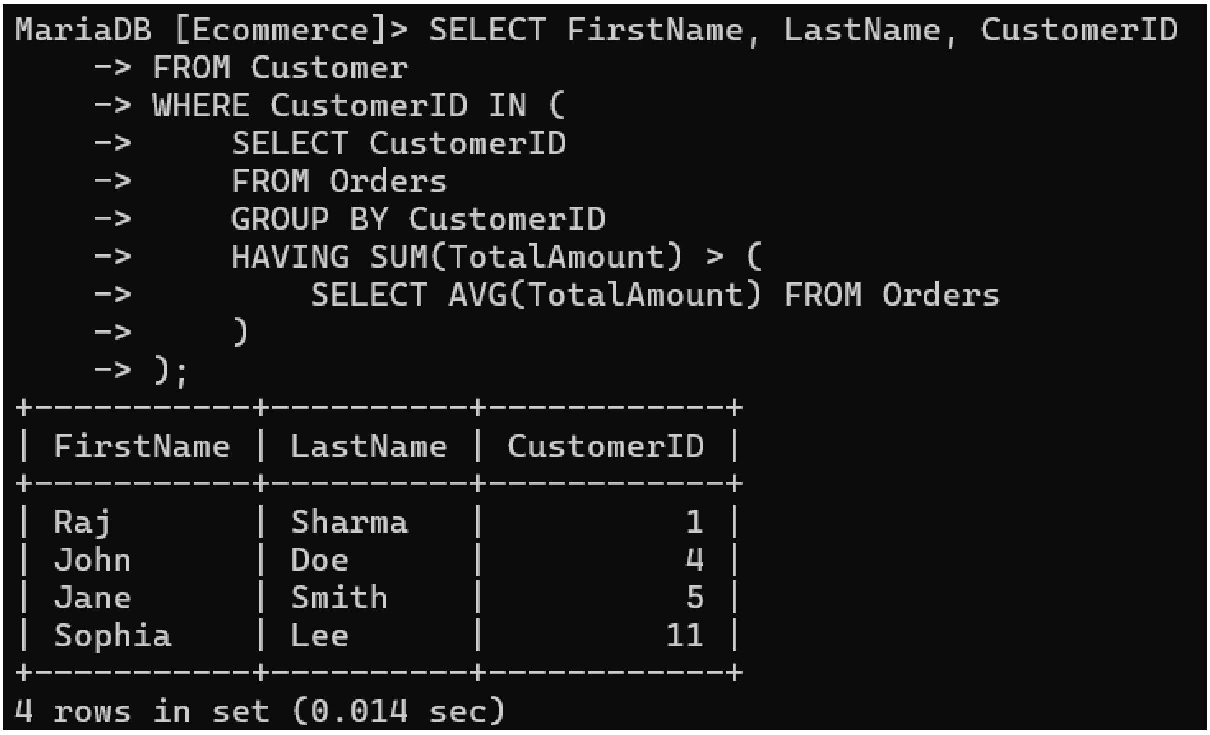


```
MariaDB [Ecommerce]> SELECT c.FirstName, c.LastName, o.OrderID
    -> FROM Customer c
    -> LEFT JOIN Orders o ON c.CustomerID = o.CustomerID;
+-----------+----------+---------+
| FirstName | LastName | OrderID |
+-----------+----------+---------+
| Raj       | Sharma   |       1 |
| Priya     | Mehta    |       2 |
| Amit      | Kumar    |       3 |
| John      | Doe      |       4 |
| Jane      | Smith    |       5 |
| Robert    | Brown    |       6 |
| Emily     | Jones    |       7 |
| Chris     | Taylor   |       8 |
| Olivia    | Wilson   |       9 |
| Liam      | Martin   |      10 |
| Sophia    | Lee      |      11 |
| Lucas     | White    |      12 |
| Isabella  | Harris   |      13 |
| Noah      | Clark    |      14 |
| Mia       | Hall     |      15 |
+-----------+----------+---------+
15 rows in set (0.002 sec)
```

Task 4: SQL for Data Analysis

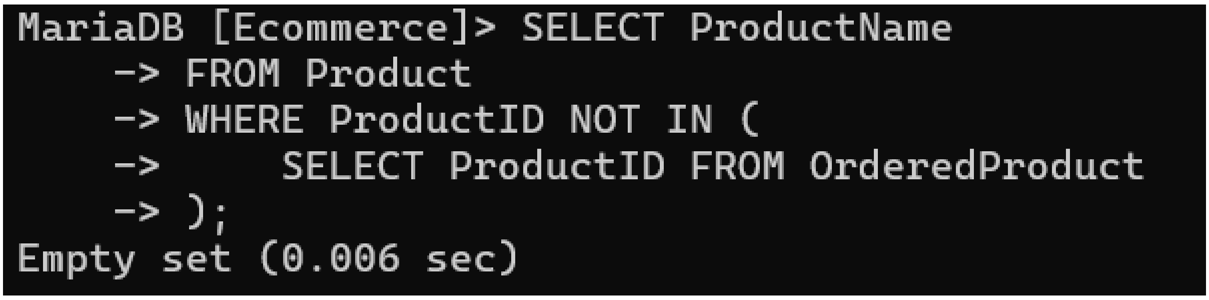
Step 7: Sub-queries

- **Find customers who spent more than average:**



```
MariaDB [Ecommerce]> SELECT FirstName, LastName, CustomerID
    -> FROM Customer
    -> WHERE CustomerID IN (
    ->     SELECT CustomerID
    ->     FROM Orders
    ->     GROUP BY CustomerID
    ->     HAVING SUM(TotalAmount) > (
    ->         SELECT AVG(TotalAmount) FROM Orders
    ->     )
    -> );
+-----------+----------+------------+
| FirstName | LastName | CustomerID |
+-----------+----------+------------+
| Raj       | Sharma   |          1 |
| John      | Doe      |          4 |
| Jane      | Smith    |          5 |
| Sophia    | Lee      |         11 |
+-----------+----------+------------+
4 rows in set (0.014 sec)
```

- **Products never ordered:**

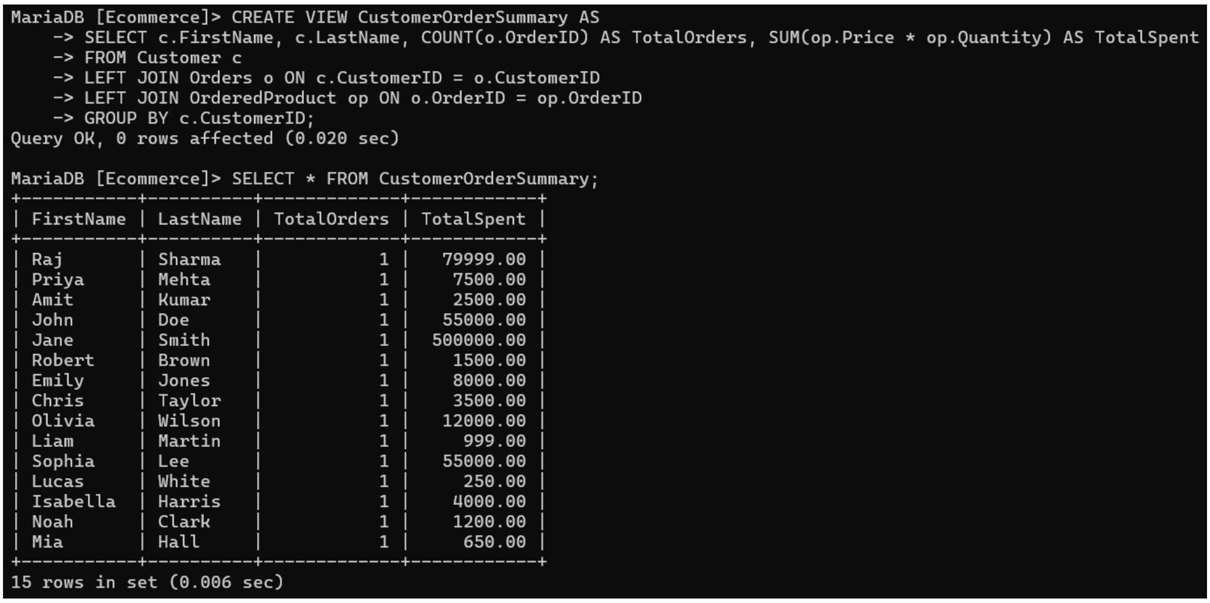


```
MariaDB [Ecommerce]> SELECT ProductName
    -> FROM Product
    -> WHERE ProductID NOT IN (
    ->     SELECT ProductID FROM OrderedProduct
    -> );
Empty set (0.006 sec)
```

Task 4: SQL for Data Analysis

Step 8: Views

- **Create a view for easier analysis:**

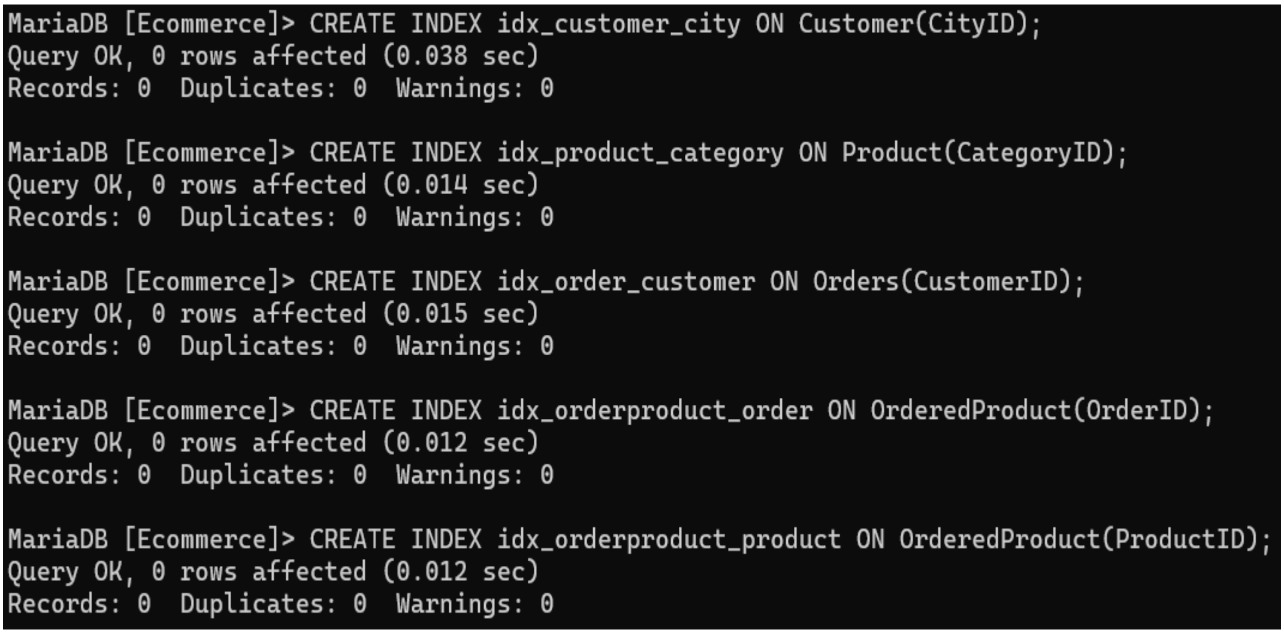


```
MariaDB [Ecommerce]> CREATE VIEW CustomerOrderSummary AS
    -> SELECT c.FirstName, c.LastName, COUNT(o.OrderID) AS TotalOrders, SUM(op.Price * op.Quantity) AS TotalSpent
    -> FROM Customer c
    -> LEFT JOIN Orders o ON c.CustomerID = o.CustomerID
    -> LEFT JOIN OrderedProduct op ON o.OrderID = op.OrderID
    -> GROUP BY c.CustomerID;
Query OK, 0 rows affected (0.020 sec)

MariaDB [Ecommerce]> SELECT * FROM CustomerOrderSummary;
+-----------+----------+-------------+------------+
| FirstName | LastName | TotalOrders | TotalSpent |
+-----------+----------+-------------+------------+
| Raj       | Sharma   |           1 |   79999.00 |
| Priya     | Mehta    |           1 |    7500.00 |
| Amit      | Kumar    |           1 |    2500.00 |
| John      | Doe      |           1 |   55000.00 |
| Jane      | Smith    |           1 |  500000.00 |
| Robert    | Brown    |           1 |    1500.00 |
| Emily     | Jones    |           1 |    8000.00 |
| Chris     | Taylor   |           1 |    3500.00 |
| Olivia    | Wilson   |           1 |   12000.00 |
| Liam      | Martin   |           1 |     999.00 |
| Sophia    | Lee      |           1 |   55000.00 |
| Lucas     | White    |           1 |     250.00 |
| Isabella  | Harris   |           1 |    4000.00 |
| Noah      | Clark    |           1 |    1200.00 |
| Mia       | Hall     |           1 |     650.00 |
+-----------+----------+-------------+------------+
15 rows in set (0.006 sec)
```

Step 9: Indexing for Optimization

- **Create indexes on frequently joined columns:**



```
MariaDB [Ecommerce]> CREATE INDEX idx_customer_city ON Customer(CityID);
Query OK, 0 rows affected (0.038 sec)
Records: 0  Duplicates: 0  Warnings: 0

MariaDB [Ecommerce]> CREATE INDEX idx_product_category ON Product(CategoryID);
Query OK, 0 rows affected (0.014 sec)
Records: 0  Duplicates: 0  Warnings: 0

MariaDB [Ecommerce]> CREATE INDEX idx_order_customer ON Orders(CustomerID);
Query OK, 0 rows affected (0.015 sec)
Records: 0  Duplicates: 0  Warnings: 0

MariaDB [Ecommerce]> CREATE INDEX idx_orderproduct_order ON OrderedProduct(OrderID);
Query OK, 0 rows affected (0.012 sec)
Records: 0  Duplicates: 0  Warnings: 0

MariaDB [Ecommerce]> CREATE INDEX idx_orderproduct_product ON OrderedProduct(ProductID);
Query OK, 0 rows affected (0.012 sec)
Records: 0  Duplicates: 0  Warnings: 0
```