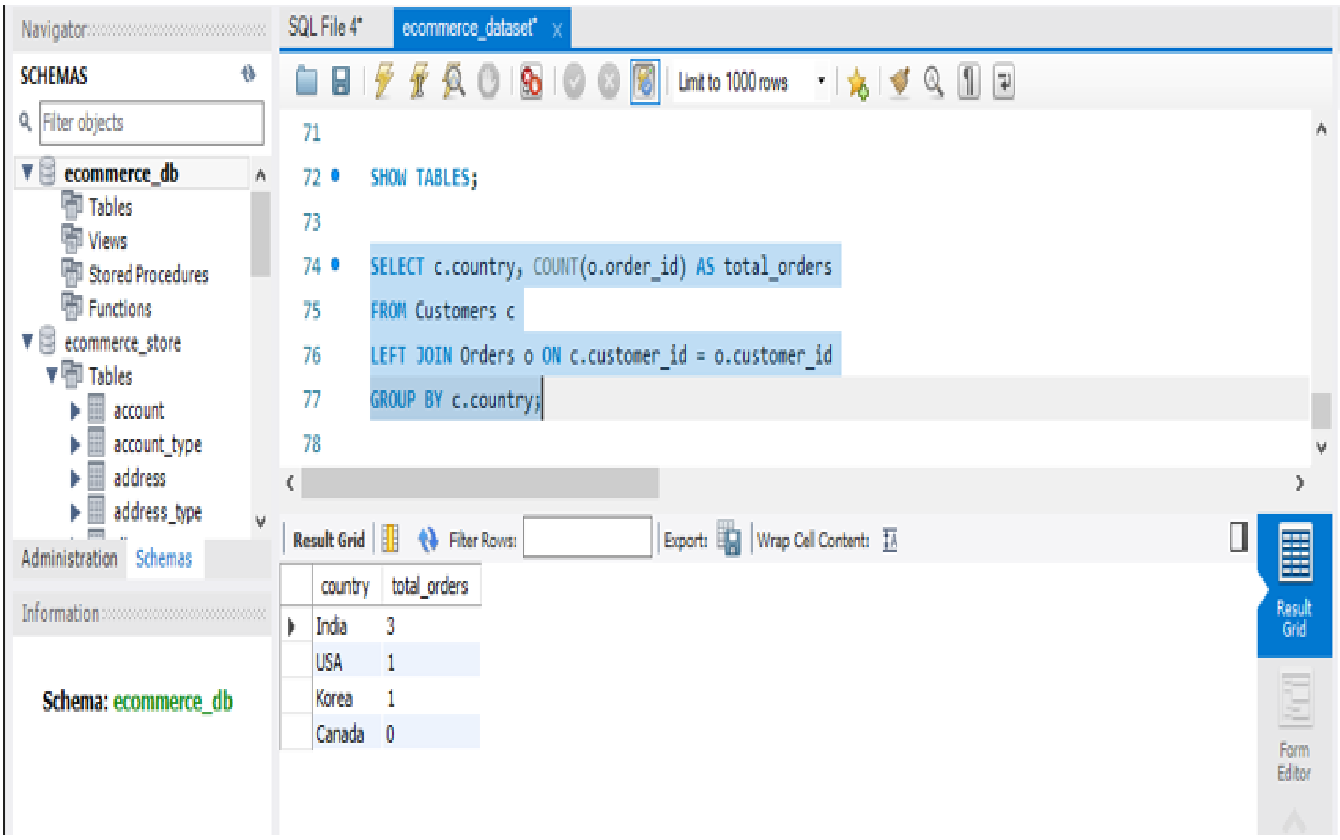
SQL SELECT QUERIES OUTPUT

1 Orders Count by Country

Query Summary

This query shows how many orders were placed from each country. It connects the Customers table with the Orders table and counts the number of orders for every country.



```
SHOW TABLES;

SELECT c.country, COUNT(o.order_id) AS total_orders
FROM Customers c
LEFT JOIN Orders o ON c.customer_id = o.customer_id
GROUP BY c.country;
```

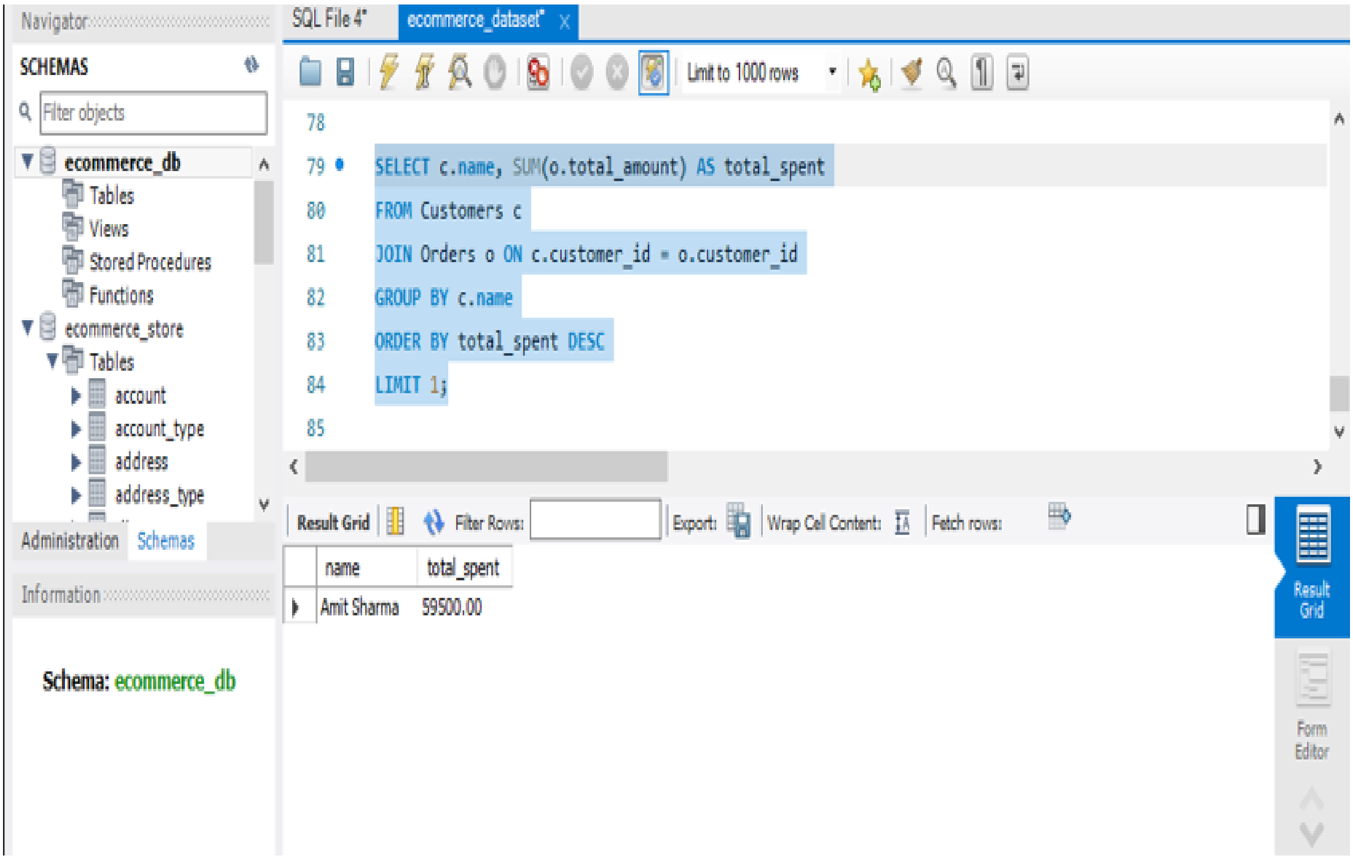
country	total_orders
India	3
USA	1
Korea	1
Canada	0

Schema: ecommerce_db

2 Highest Spending Customer

Query Summary

This query calculates how much each customer has spent in total and then identifies the customer who spent the most.
It uses SUM(total_amount) on the Orders table.



```sql
SELECT c.name, SUM(o.total_amount) AS total_spent
FROM Customers c
JOIN Orders o ON c.customer_id = o.customer_id
GROUP BY c.name
ORDER BY total_spent DESC
LIMIT 1;
```

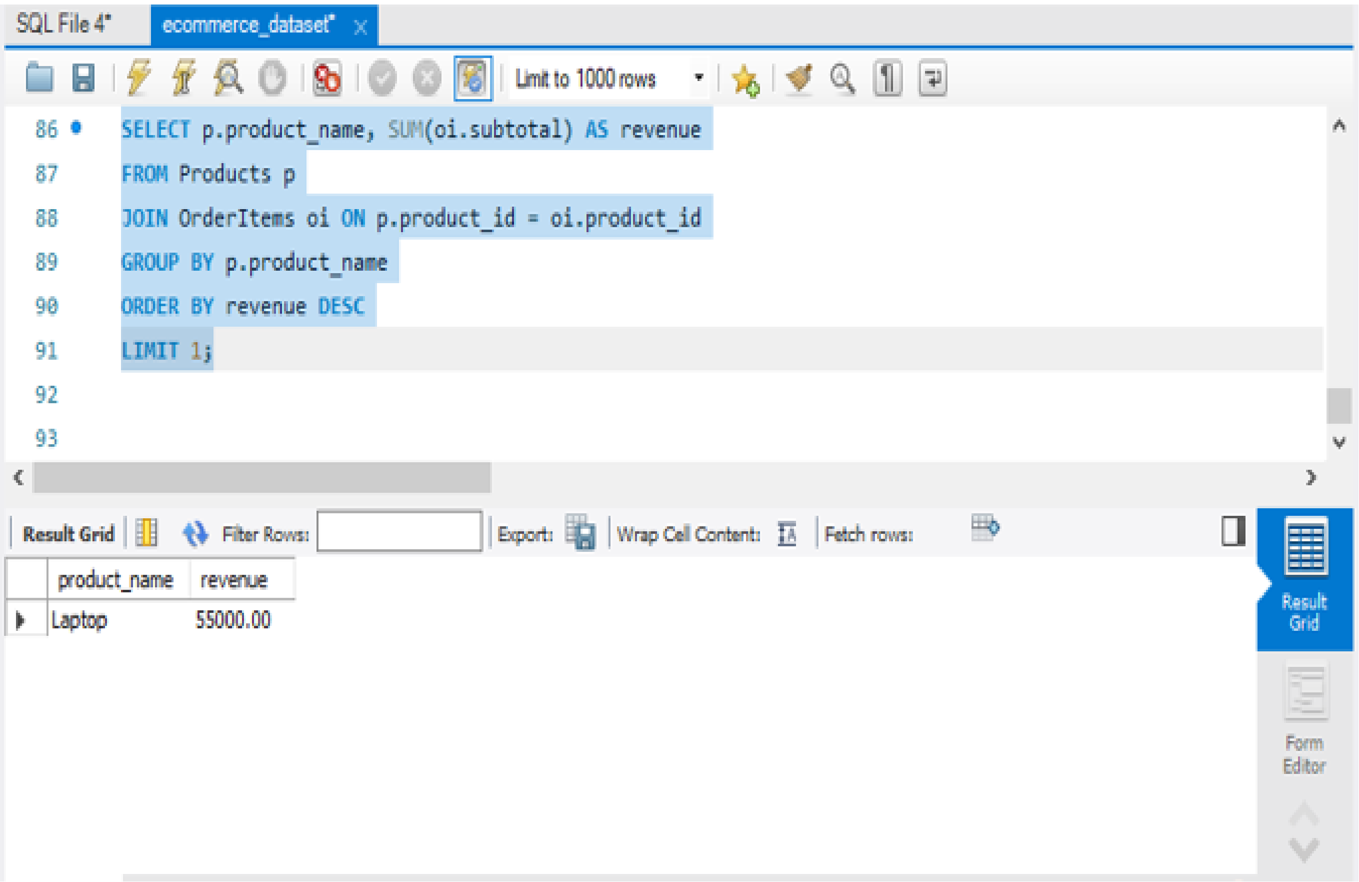
name	total_spent
Amit Sharma	59500.00

Schema: ecommerce_db

3 Product with the Highest Revenue

Query Summary

This query finds out which product generated the highest revenue. It sums the subtotal from the OrderItems table for each product.



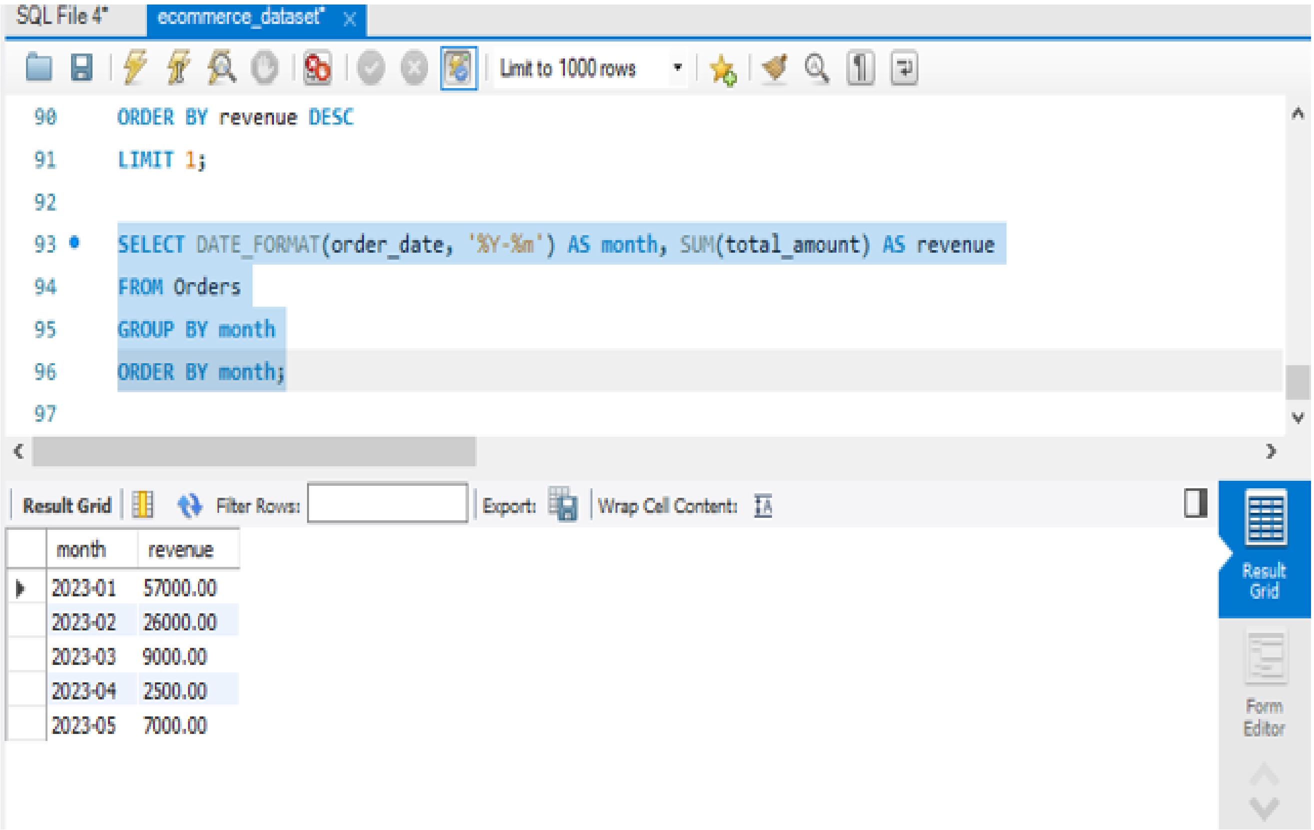
```
SELECT p.product_name, SUM(oi.subtotal) AS revenue
FROM Products p
JOIN OrderItems oi ON p.product_id = oi.product_id
GROUP BY p.product_name
ORDER BY revenue DESC
LIMIT 1;
```

```
product_name   revenue
Laptop         55000.00
```

4 Monthly Revenue Report

Query Summary

This query calculates total monthly revenue by formatting the order date into a YEAR-MONTH format.
Then it groups orders by month to get monthly earnings.



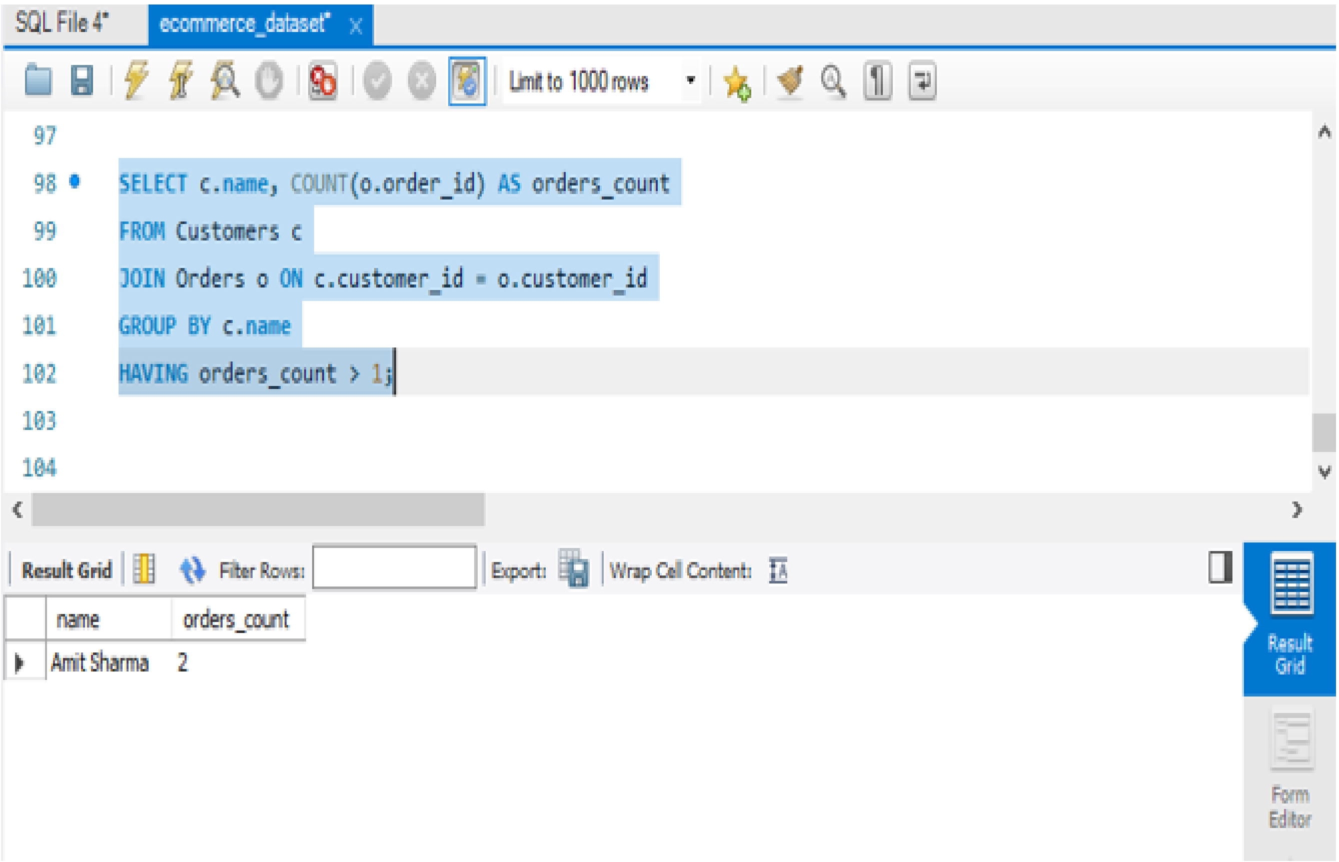
```sql
ORDER BY revenue DESC
LIMIT 1;

SELECT DATE_FORMAT(order_date, '%Y-%m') AS month, SUM(total_amount) AS revenue
FROM Orders
GROUP BY month
ORDER BY month;
```

month	revenue
2023-01	57000.00
2023-02	26000.00
2023-03	9000.00
2023-04	2500.00
2023-05	7000.00

5 Customers with More Than One Order

Query Summary

This query finds customers who placed more than one order.
It groups orders by customer and filters using the HAVING clause.



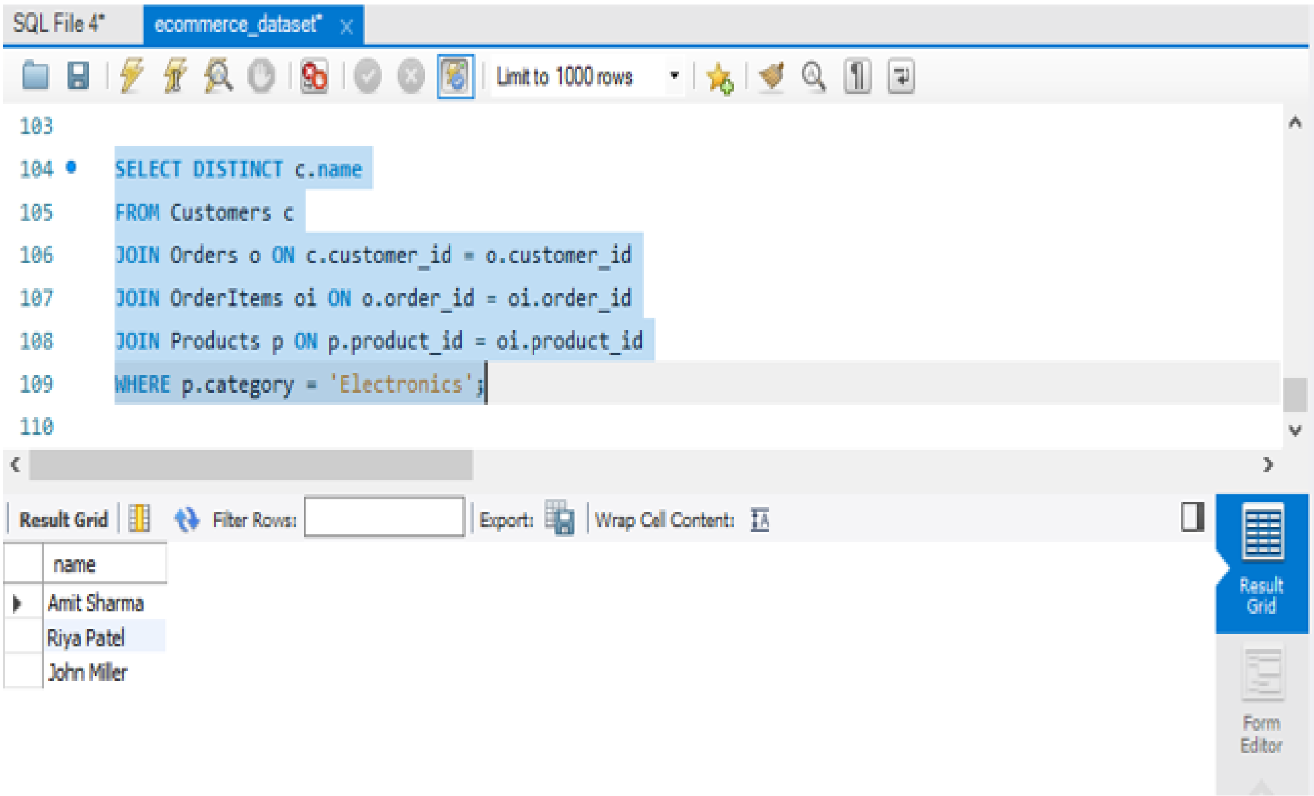
```sql
SELECT c.name, COUNT(o.order_id) AS orders_count
FROM Customers c
JOIN Orders o ON c.customer_id = o.customer_id
GROUP BY c.name
HAVING orders_count > 1;
```

name	orders_count
Amit Sharma	2

6 Customers Who Bought Electronics

Query Summary

This query finds all customers who purchased items from the Electronics category.
It joins Customers → Orders → OrderItems → Products to trace category-wise purchases.



```
SELECT DISTINCT c.name
FROM Customers c
JOIN Orders o ON c.customer_id = o.customer_id
JOIN OrderItems oi ON o.order_id = oi.order_id
JOIN Products p ON p.product_id = oi.product_id
WHERE p.category = 'Electronics';
```

name
Amit Sharma
Riya Patel
John Miller