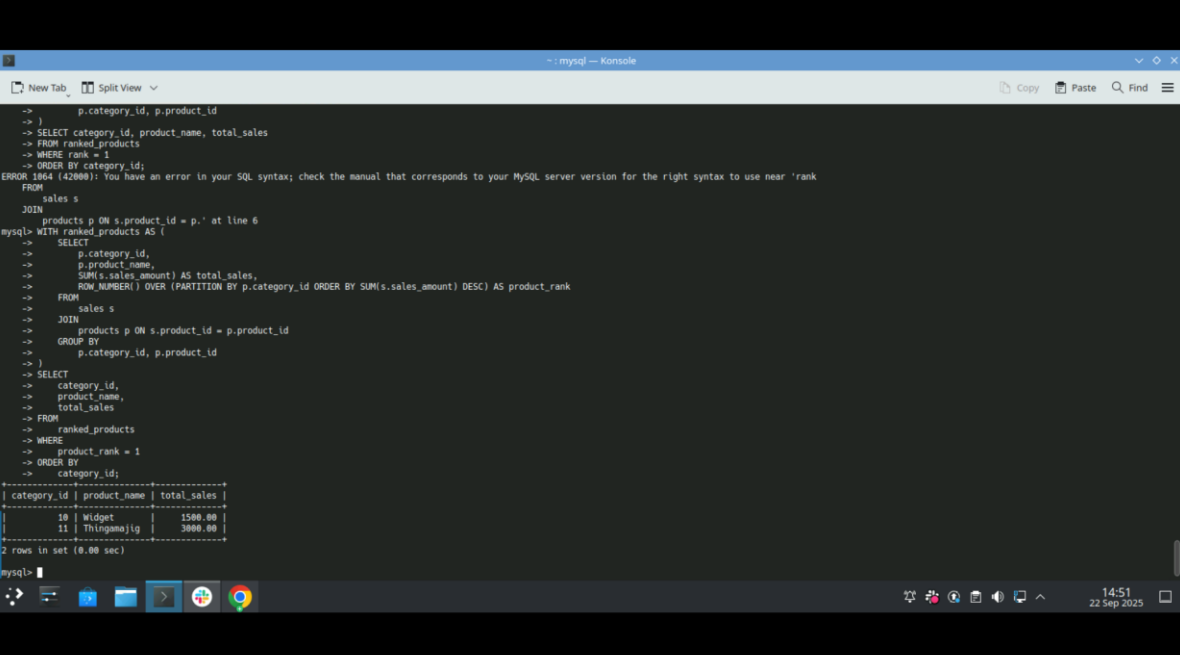


Module : SQL

Nithin K Shine
22/9/2025

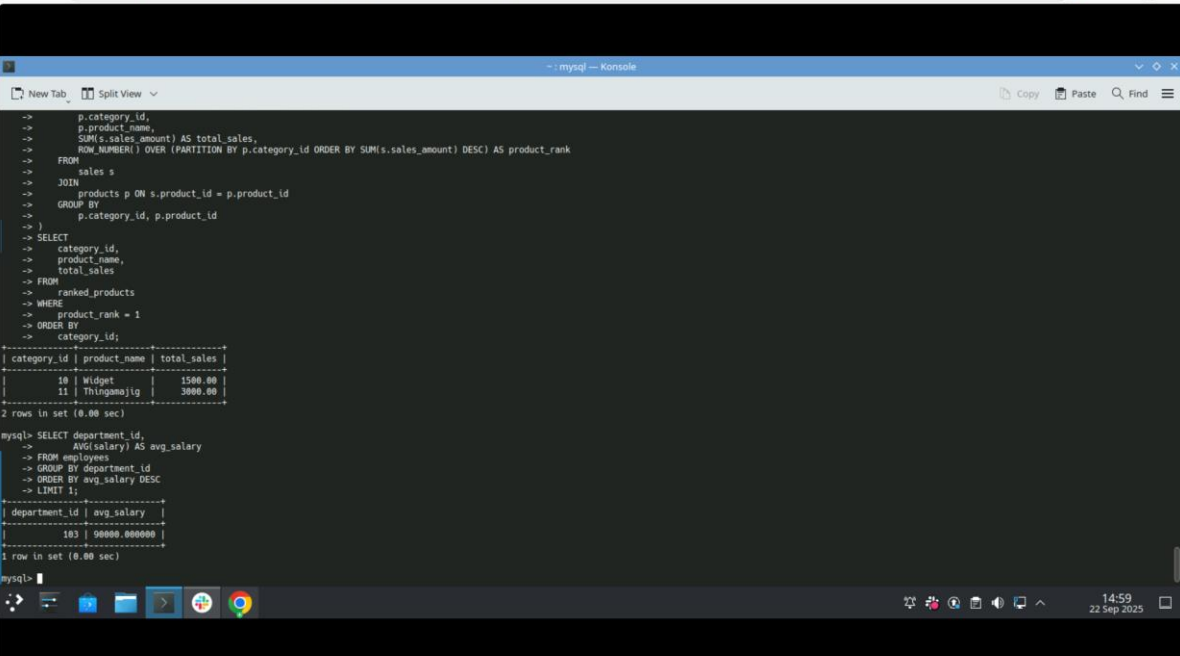
Q1: List Top-Selling Product in Each Category



```
-- p.category_id, p.product_id
-- SELECT category_id, product_name, total_sales
-- FROM ranked_products
-- WHERE rank = 1
-- ORDER BY category_id;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'rank
FROM
sales s
JOIN
  products p ON s.product_id = p.' at line 6
mysql> WITH ranked_products AS (
-- SELECT
--   p.category_id,
--   p.product_name,
--   SUM(s.sales_amount) AS total_sales,
--   ROW_NUMBER() OVER (PARTITION BY p.category_id ORDER BY SUM(s.sales_amount) DESC) AS product_rank
-- FROM
--   sales s
-- JOIN
--   products p ON s.product_id = p.product_id
-- GROUP BY
--   p.category_id, p.product_id
-- )
-- SELECT
--   category_id,
--   product_name,
--   total_sales
-- FROM
--   ranked_products
-- WHERE
--   product_rank = 1
-- ORDER BY
--   category_id;
+-----+-----+-----+
| category_id | product_name | total_sales |
+-----+-----+-----+
| 10 | Widget | 1500.00 |
| 11 | Thingamajig | 3000.00 |
+-----+-----+-----+
2 rows in set (0.00 sec)

mysql>
```

Q2: List Department with the Highest Average Salary



```
-- p.category_id,
-- p.product_name,
-- SUM(s.sales_amount) AS total_sales,
-- ROW_NUMBER() OVER (PARTITION BY p.category_id ORDER BY SUM(s.sales_amount) DESC) AS product_rank
-- FROM
--   sales s
-- JOIN
--   products p ON s.product_id = p.product_id
-- GROUP BY
--   p.category_id, p.product_id
-- )
-- SELECT
--   category_id,
--   product_name,
--   total_sales
-- FROM
--   ranked_products
-- WHERE
--   product_rank = 1
-- ORDER BY
--   category_id;
+-----+-----+-----+
| category_id | product_name | total_sales |
+-----+-----+-----+
| 10 | Widget | 1500.00 |
| 11 | Thingamajig | 3000.00 |
+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> SELECT department_id,
-- AVG(salary) AS avg_salary
-- FROM employees
-- GROUP BY department_id
-- ORDER BY avg_salary DESC
-- LIMIT 1;
+-----+-----+
| department_id | avg_salary |
+-----+-----+
| 103 | 90000.000000 |
+-----+-----+
1 row in set (0.00 sec)

mysql>
```

Q3: List Employees Who Earn More Than Department Average

1 row in set (0.00 sec)

```
mysql> SELECT e.employee_id,
-> e.name,
-> e.department_id,
-> e.salary
-> FROM employees e
-> WHERE e.salary > (
-> SELECT AVG(salary)
-> FROM employees
-> WHERE department_id = e.department_id
-> );
```

employee_id	name	department_id	salary
2	Bob	101	80000.00
5	Ethan	102	70000.00

2 rows in set (0.00 sec)

```
mysql> SELECT e.employee_id,
-> e.name,
-> e.department_id,
-> e.salary,
-> d.avg_salary
-> FROM employees e
-> JOIN (
-> SELECT department_id, AVG(salary) AS avg_salary
-> FROM employees
-> GROUP BY department_id
-> ) d
-> ON e.department_id = d.department_id
-> WHERE e.salary > d.avg_salary;
```

employee_id	name	department_id	salary	avg_salary
2	Bob	101	80000.00	75000.000000
5	Ethan	102	70000.00	67500.000000

2 rows in set (0.01 sec)

mysql>

Q4: List Customer with Most Orders per Region

```
-> e.salary,
-> d.avg_salary
-> FROM employees e
-> JOIN (
-> SELECT department_id, AVG(salary) AS avg_salary
-> FROM employees
-> GROUP BY department_id
-> ) d
-> ON e.department_id = d.department_id
-> WHERE e.salary > d.avg_salary;
```

employee_id	name	department_id	salary	avg_salary
2	Bob	101	80000.00	75000.000000
5	Ethan	102	70000.00	67500.000000

2 rows in set (0.01 sec)

```
mysql> SELECT o.region,
-> o.customer_id,
-> COUNT(*) AS order_count
-> FROM orders o
-> GROUP BY o.region, o.customer_id
-> HAVING COUNT(*) = (
-> SELECT MAX(order_count)
-> FROM (
-> SELECT region, customer_id, COUNT(*) AS order_count
-> FROM orders
-> WHERE region = o.region
-> GROUP BY region, customer_id
-> ) sub
-> );
```

region	customer_id	order_count
North	301	2
North	302	2
South	303	1
South	304	1

4 rows in set (0.01 sec)

mysql>

Q5: List Categories Where Average Price > Overall Average Price

```
mysql> SELECT p.category_id,
->         AVG(p.price) AS avg_category_price,
->         t.overall_avg_price
-> FROM products p
-> CROSS JOIN (
->     SELECT AVG(price) AS overall_avg_price
->     FROM products
-> ) t
-> GROUP BY p.category_id, t.overall_avg_price
-> HAVING AVG(p.price) > t.overall_avg_price;
+-----+-----+-----+
| category_id | avg_category_price | overall_avg_price |
+-----+-----+-----+
| 11          | 100.000000        | 43.750000         |
+-----+-----+-----+
1 row in set (0.00 sec)
```

Q6: List Products Never Ordered

```
mysql> SELECT p.product_id,
->         p.product_name,
->         p.category_id,
->         p.price
-> FROM products p
-> LEFT JOIN orders o
-> ON p.product_id = o.product_id
-> WHERE o.order_id IS NULL;
+-----+-----+-----+-----+
| product_id | product_name | category_id | price |
+-----+-----+-----+-----+
| 204        | Doohickey    | 12          | 10.00 |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

=====TheEnd=====