**TASK 4:**

**QUESTION 1: Write an SQL query to find out which customers have not placed any orders.**

mysql> SELECT c.customerid, CONCAT(c.firstname, ' ', c.lastname) AS customername

-> FROM customers c

-> WHERE c.customerid NOT IN (SELECT DISTINCT o.customerid FROM orders o);

+------------+------------------+

| customerid | customername |

+------------+------------------+

| 6 | Evangelin Miller |

| 12 | Michael Scott |

+------------+------------------+

2 rows in set (0.01 sec)

**QUESTION 2: Write an SQL query to find the total number of products available for sale.**

mysql> SELECT COUNT(\*) AS total\_products FROM products;

+----------------+

| total\_products |

+----------------+

| 11 |

+----------------+

1 row in set (0.03 sec)

**QUESTION 3: Write an SQL query to calculate the total revenue generated by TechShop.**

mysql> SELECT SUM(totalamount) AS total\_revenue FROM orders;

+---------------+

| total\_revenue |

+---------------+

| 6972.86 |

+---------------+

1 row in set (0.00 sec)

**QUESTION 4: Write an SQL query to calculate the average quantity ordered for products in a specific category. Allow users to input the category name as a parameter.**

mysql> select \* from products;

+-----------+--------------+----------------------+---------+

| productid | productname | description | price |

+-----------+--------------+----------------------+---------+

| 1 | Laptop | Dell Inspiron 15 | 879.99 |

| 2 | Smartphone | Samsung Galaxy S22 | 1099.99 |

| 3 | Tablet | Apple iPad Pro | 1209.99 |

| 4 | Headphones | Sony WH-1000XM4 | 329.99 |

| 5 | Keyboard | Mechanical Keyboard | 89.99 |

| 6 | Mouse | Wireless Mouse | 49.99 |

| 7 | Monitor | 27-inch 4K Monitor | 384.99 |

| 8 | Printer | HP LaserJet Pro | 199.99 |

| 9 | External SSD | Samsung T7 1TB | 129.99 |

| 10 | Router | Netgear Nighthawk | 197.99 |

| 11 | Smartwatch | Apple Watch series 9 | 399.99 |

+-----------+--------------+----------------------+---------+

11 rows in set (0.00 sec)

**ADDING A CATEGORY COLUMN IN THE PRODUCTS TABLE:**

mysql> ALTER TABLE products ADD COLUMN category VARCHAR(50);

Query OK, 0 rows affected (0.16 sec)

Records: 0 Duplicates: 0 Warnings: 0

mysql> UPDATE products

-> SET category = CASE

-> WHEN productname IN ('Smartphone', 'Tablet', 'Smartwatch') THEN 'Mobile Devices'

-> WHEN productname IN ('Laptop') THEN 'Laptops'

-> WHEN productname IN ('Keyboard', 'Mouse', 'Monitor', 'External SSD') THEN 'Computer Accessories'

-> WHEN productname IN ('Headphones') THEN 'Audio & Accessories'

-> WHEN productname IN ('Router') THEN 'Networking & Storage'

-> WHEN productname IN ('Printer') THEN 'Office Equipment'

-> ELSE 'Other'

-> END;

Query OK, 11 rows affected (0.01 sec)

Rows matched: 11 Changed: 11 Warnings: 0

mysql> select productid, productname, category from products;

+-----------+--------------+----------------------+

| productid | productname | category |

+-----------+--------------+----------------------+

| 1 | Laptop | Laptops |

| 2 | Smartphone | Mobile Devices |

| 3 | Tablet | Mobile Devices |

| 4 | Headphones | Audio & Accessories |

| 5 | Keyboard | Computer Accessories |

| 6 | Mouse | Computer Accessories |

| 7 | Monitor | Computer Accessories |

| 8 | Printer | Office Equipment |

| 9 | External SSD | Computer Accessories |

| 10 | Router | Networking & Storage |

| 11 | Smartwatch | Mobile Devices |

+-----------+--------------+----------------------+

11 rows in set (0.00 sec)

mysql> select \* from products;

+-----------+--------------+----------------------+---------+----------------------+

| productid | productname | description | price | category |

+-----------+--------------+----------------------+---------+----------------------+

| 1 | Laptop | Dell Inspiron 15 | 879.99 | Laptops |

| 2 | Smartphone | Samsung Galaxy S22 | 1099.99 | Mobile Devices |

| 3 | Tablet | Apple iPad Pro | 1209.99 | Mobile Devices |

| 4 | Headphones | Sony WH-1000XM4 | 329.99 | Audio & Accessories |

| 5 | Keyboard | Mechanical Keyboard | 89.99 | Computer Accessories |

| 6 | Mouse | Wireless Mouse | 49.99 | Computer Accessories |

| 7 | Monitor | 27-inch 4K Monitor | 384.99 | Computer Accessories |

| 8 | Printer | HP LaserJet Pro | 199.99 | Office Equipment |

| 9 | External SSD | Samsung T7 1TB | 129.99 | Computer Accessories |

| 10 | Router | Netgear Nighthawk | 197.99 | Networking & Storage |

| 11 | Smartwatch | Apple Watch series 9 | 399.99 | Mobile Devices |

+-----------+--------------+----------------------+---------+----------------------+

11 rows in set (0.00 sec)

mysql> SELECT AVG(od.quantity) AS avg\_quantity\_ordered

-> FROM orderdetails od

-> JOIN products p ON od.productid = p.productid

-> WHERE p.category = 'Mobile Devices';

+----------------------+

| avg\_quantity\_ordered |

+----------------------+

| 1.0000 |

+----------------------+

1 row in set (0.01 sec)

**QUESTION 5: Write an SQL query to calculate the total revenue generated by a specific customer. Allow users to input the customer ID as a parameter.**

mysql> SELECT c.customerid, CONCAT(c.firstname, ' ', c.lastname) AS customername, SUM(o.totalamount) AS total\_revenue

-> FROM customers c

-> JOIN orders o ON c.customerid = o.customerid

-> WHERE c.customerid = '4'

-> GROUP BY c.customerid, customername;

+------------+--------------+---------------+

| customerid | customername | total\_revenue |

+------------+--------------+---------------+

| 4 | Ally Brown | 1209.99 |

+------------+--------------+---------------+

1 row in set (0.01 sec)

**QUESTION 6: Write an SQL query to find the customers who have placed the most orders. List their names and the number of orders they've placed.**

mysql> SELECT c.customerid, CONCAT(c.firstname, ' ', c.lastname) AS customername, COUNT(o.orderid) AS total\_orders

-> FROM customers c

-> JOIN orders o ON c.customerid = o.customerid

-> GROUP BY c.customerid, customername

-> ORDER BY total\_orders DESC

-> LIMIT 1;

+------------+--------------+--------------+

| customerid | customername | total\_orders |

+------------+--------------+--------------+

| 1 | John Down | 2 |

+------------+--------------+--------------+

1 row in set (0.01 sec)

**QUESTION 7: Write an SQL query to find the most popular product category, which is the one with the highest total quantity ordered across all orders.**

mysql> SELECT p.category, SUM(od.quantity) AS total\_quantity\_ordered

-> FROM orderdetails od

-> JOIN products p ON od.productid = p.productid

-> GROUP BY p.category

-> ORDER BY total\_quantity\_ordered DESC

-> LIMIT 1;

+----------------------+------------------------+

| category | total\_quantity\_ordered |

+----------------------+------------------------+

| Computer Accessories | 5 |

+----------------------+------------------------+

1 row in set (0.00 sec)

**QUESTON 8: Write an SQL query to find the customer who has spent the most money (highest total revenue) on electronic gadgets. List their name and total spending.**

mysql> SELECT c.customerid, CONCAT(c.firstname, ' ', c.lastname) AS customername, SUM(od.quantity \* p.price) AS total\_spent

-> FROM customers c

-> JOIN orders o ON c.customerid = o.customerid

-> JOIN orderdetails od ON o.orderid = od.orderid

-> JOIN products p ON od.productid = p.productid

-> WHERE p.category IN ('Mobile Devices', 'Laptops', 'Audio & Accessories', 'Networking & Storage', 'Computer Accessories')

-> GROUP BY c.customerid, customername

-> ORDER BY total\_spent DESC

-> LIMIT 1;

+------------+--------------+-------------+

| customerid | customername | total\_spent |

+------------+--------------+-------------+

| 1 | John Down | 1889.97 |

+------------+--------------+-------------+

1 row in set (0.01 sec)

**QUESTON 9: Write an SQL query to calculate the average order value (total revenue divided by the number of orders) for all customers.**

mysql> SELECT AVG(o.totalamount) AS avg\_order\_value FROM orders o;

+-----------------+

| avg\_order\_value |

+-----------------+

| 536.373846 |

+-----------------+

1 row in set (0.00 sec)

**QUESTON 10: Write an SQL query to find the total number of orders placed by each customer and list their names along with the order count.**

mysql> SELECT c.customerid, CONCAT(c.firstname, ' ', c.lastname) AS customername, COUNT(o.orderid) AS total\_orders

-> FROM customers c

-> LEFT JOIN orders o ON c.customerid = o.customerid

-> GROUP BY c.customerid, customername

-> ORDER BY total\_orders DESC;

+------------+------------------+--------------+

| customerid | customername | total\_orders |

+------------+------------------+--------------+

| 1 | John Down | 2 |

| 3 | Bob Johnson | 2 |

| 5 | Charles David | 2 |

| 7 | Daniel Wilson | 2 |

| 2 | Persis Eloite | 1 |

| 4 | Ally Brown | 1 |

| 8 | Sophia Mordecai | 1 |

| 9 | Fanny Anderson | 1 |

| 10 | Olivia Harris | 1 |

| 6 | Evangelin Miller | 0 |

| 12 | Michael Scott | 0 |

+------------+------------------+--------------+

11 rows in set (0.00 sec)