

Name – Vikas Srivastava

Topic – SQL Assignment : 10

Batch - DATACOM+5G Dev

Batch Id - 25SUB4505

User Id – 55984

Email Id – vikas200027@gmail.com

Assignment 10: Utilize a subquery to find customers who have placed orders above the average order value, and write a UNION query to combine two SELECT statements with the same number of columns.

Solution :

```
mysql> SELECT c.customer_name, c.email, o.order_id, o.order_date, o.order_total
-> FROM customers c
-> LEFT JOIN orders o
->     ON c.customer_id = o.customer_id;
+-----+-----+-----+-----+
| customer_name | email           | order_id | order_date   | order_total |
+-----+-----+-----+-----+
| John Doe      | john.doe@example.com | 101      | 2023-11-01  | 150.00    |
| John Doe      | john.doe@example.com | 103      | 2023-11-10  | 180.00    |
| Jane Smith    | jane.smith@example.com | 102      | 2023-11-02  | 200.00    |
| Mike Johnson  | mike.johnson@example.com | 105      | 2023-11-15  | 130.00    |
| Emily Davis   | emily.davis@example.com | 104      | 2023-11-12  | 220.00    |
| David Brown   | david.brown@example.com | NULL     | NULL        | NULL      |
+-----+-----+-----+-----+
6 rows in set (0.00 sec)
```

1.

```
mysql> SELECT c.customer_name, c.email, o.order_id, o.order_date, o.order_total
-> FROM customers c
-> INNER JOIN orders o
->     ON c.customer_id = o.customer_id
-> WHERE o.order_total > (
->     SELECT AVG(order_total) FROM orders
-> );
+-----+-----+-----+-----+
| customer_name | email           | order_id | order_date   | order_total |
+-----+-----+-----+-----+
| Jane Smith    | jane.smith@example.com | 102      | 2023-11-02  | 200.00    |
| John Doe      | john.doe@example.com | 103      | 2023-11-10  | 180.00    |
| Emily Davis   | emily.davis@example.com | 104      | 2023-11-12  | 220.00    |
+-----+-----+-----+-----+
3 rows in set (0.02 sec)
```

2.

```
mysql> SELECT c.customer_name, c.email
-> FROM customers c
-> INNER JOIN orders o
->     ON c.customer_id = o.customer_id
->
-> UNION
->
-> -- Retrieve customers who have NOT placed orders
-> SELECT c.customer_name, c.email
-> FROM customers c
-> WHERE c.customer_id NOT IN (SELECT customer_id FROM orders);
+-----+-----+
| customer_name | email      |
+-----+-----+
| John Doe      | john.doe@example.com
| Jane Smith    | jane.smith@example.com
| Mike Johnson  | mike.johnson@example.com
| Emily Davis   | emily.davis@example.com
| David Brown   | david.brown@example.com
+-----+
5 rows in set (0.01 sec)
```

3.