New Wheels: Project

Introduction to SQL

Problem Statement

Business Context

A lot of people in the world share a common desire: to own a vehicle. A car or an automobile is seen as an object that gives the freedom of mobility. Many now prefer pre-owned vehicles because they come at an affordable cost, but at the same time, they are also concerned about whether the after-sales service provided by the resale vendors is as good as the care you may get from the actual manufacturers. New-Wheels, a vehicle resale company, has launched an app with an end-to-end service from listing the vehicle on the platform to shipping it to the customer's location. This app also captures the overall after-sales feedback given by the customer.

Objective

New-Wheels sales have been dipping steadily in the past year, and due to the critical customer feedback and ratings online, there has been a drop in new customers every quarter, which is concerning to the business. The CEO of the company now wants a quarterly report with all the key metrics sent to him so he can assess the health of the business and make the necessary decisions.

As a data analyst, you see that there is an array of questions that are being asked at the leadership level that need to be answered using data. Import the dump file that contains various tables that are present in the database. Use the data to answer the questions posed and create a quarterly business report for the CEO.

Business Questions

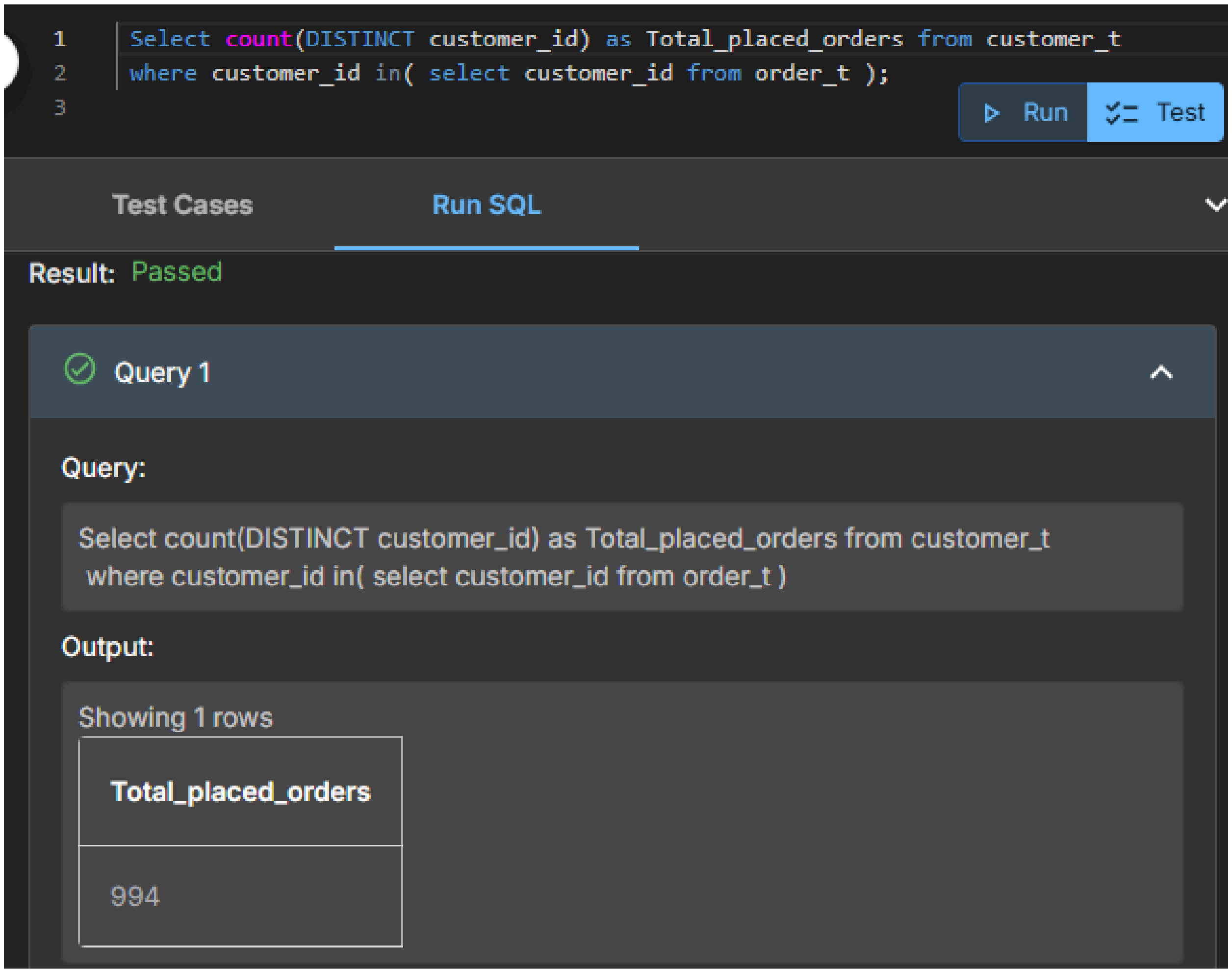
Question 1:

Q1.1:- Find the total number of customers who have placed orders

Solution Query:

Select count(DISTINCT customer_id) as Total_placed_order from customer_t

where customer_id in(select customer_id from order_t);



```
Select count(DISTINCT customer_id) as Total_placed_orders from customer_t
where customer_id in( select customer_id from order_t );
```

Result: Passed

Query 1

Query:

```
Select count(DISTINCT customer_id) as Total_placed_orders from customer_t
 where customer_id in( select customer_id from order_t )
```

Output:

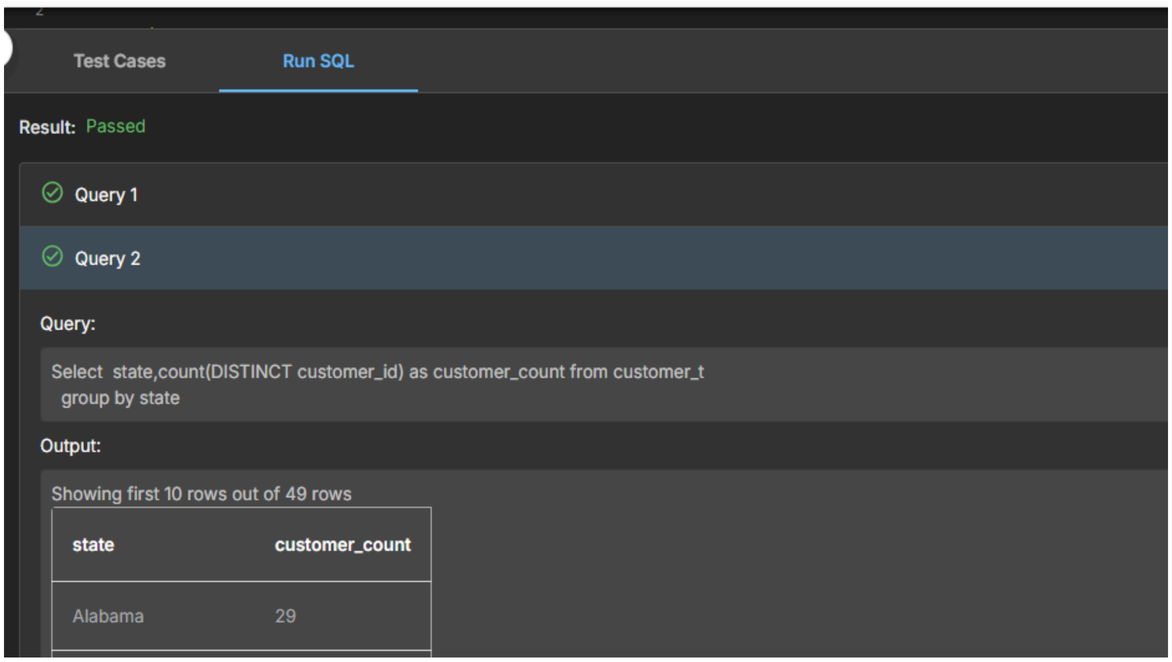
Showing 1 rows

Total_placed_orders
994

Q1.2:- What is the distribution of the customers across states?

Solution Query:

```
Select  state,count(DISTINCT customer_id) as customer_count from customer_t

 group by state;
```



Test Cases | Run SQL

Result: Passed

Query 1

Query 2

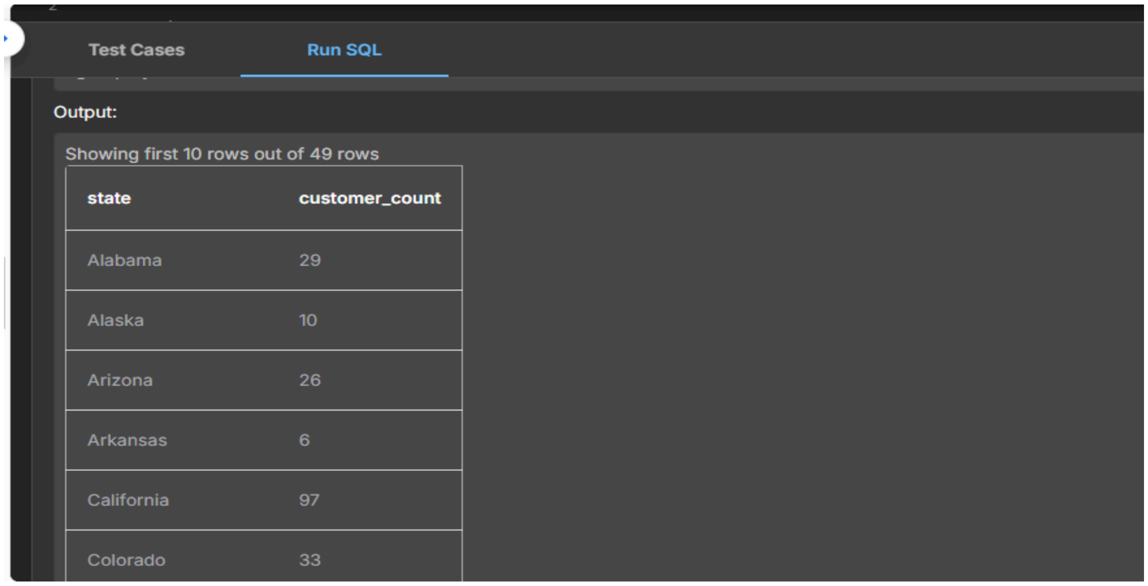
Query:

```
Select  state,count(DISTINCT customer_id) as customer_count from customer_t
  group by state
```

Output:

Showing first 10 rows out of 49 rows

state	customer_count
Alabama	29



Test Cases | Run SQL

Output:

Showing first 10 rows out of 49 rows

state	customer_count
Alabama	29
Alaska	10
Arizona	26
Arkansas	6
California	97
Colorado	33

Observations and Insights:

1. A total of 994 unique customers placed orders during the year.

2. Customer presence is uneven across states, with locations such as California and Florida contributing significantly higher volumes, while several other states register relatively low activity.

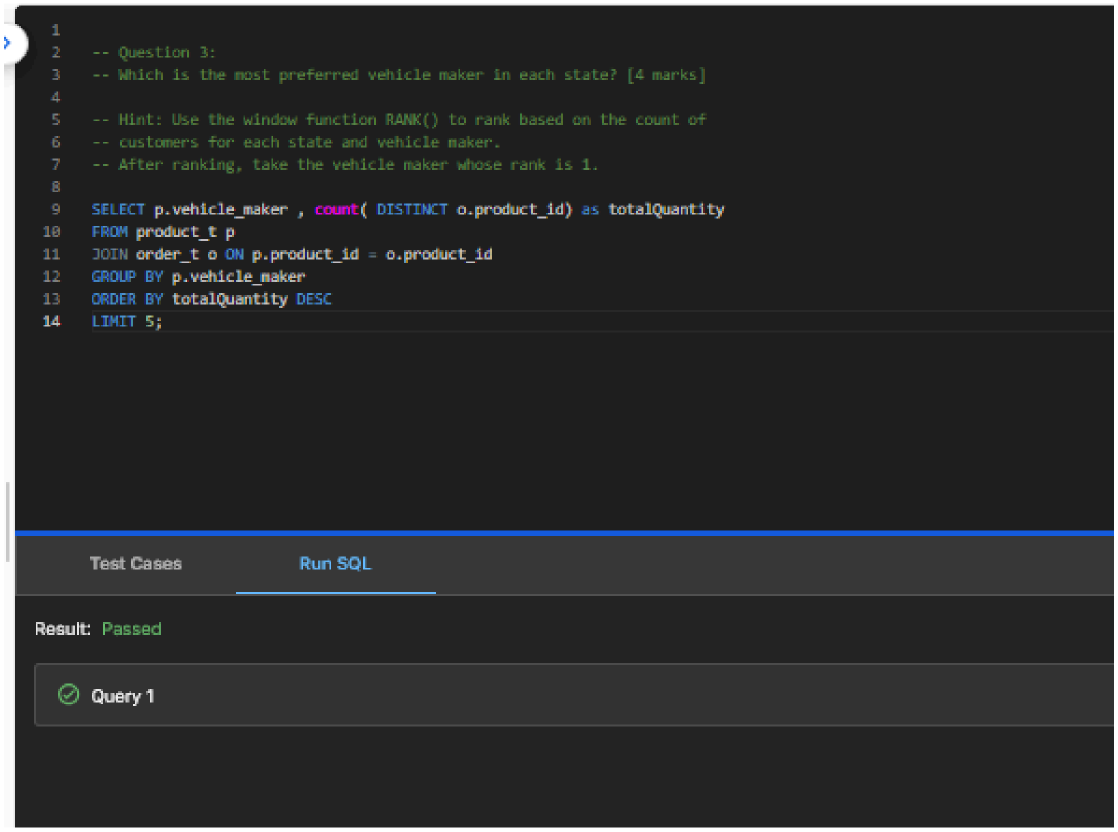
3. These disparities suggest differences in market penetration and growth opportunities across regions.

Question 2:

Which are the top 5 vehicle makers preferred by the customers?

Solution Query:

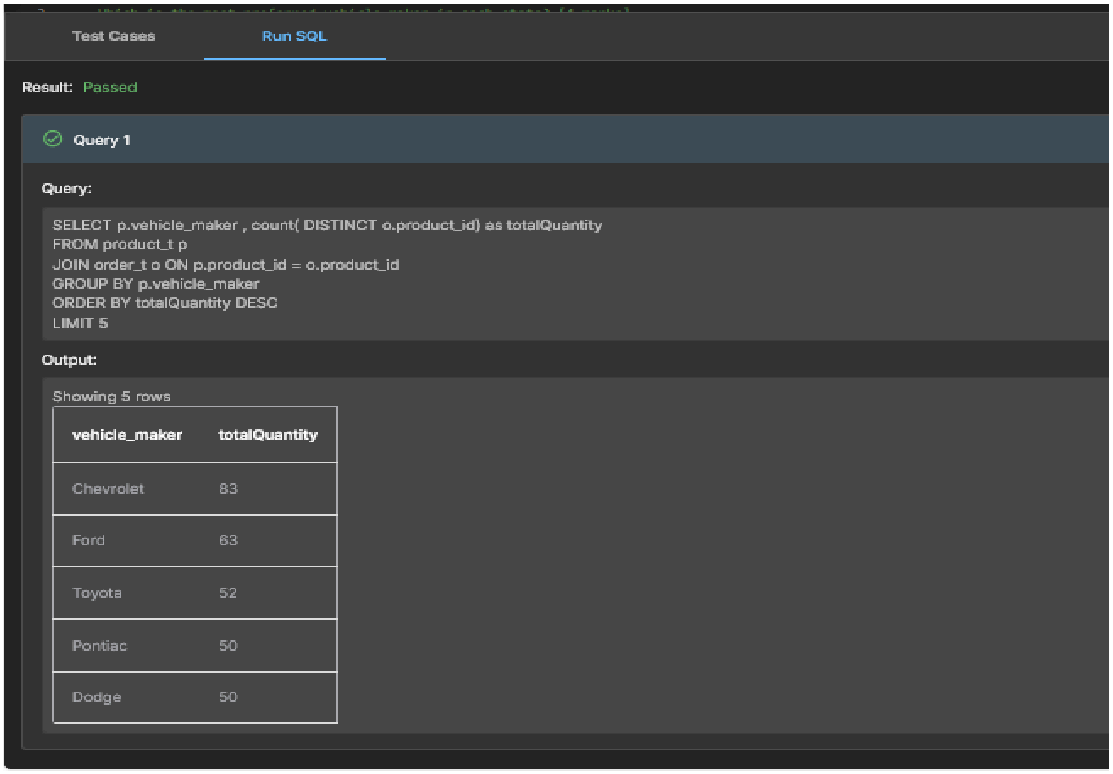
```
SELECT p.vehicle_maker , count( DISTINCT o.product_id) as totalQuantity

FROM product_t p

JOIN order_t o ON p.product_id = o.product_id

GROUP BY p.vehicle_maker

ORDER BY totalQuantity DESC

LIMIT 5;
```



```
-- Question 3:
-- Which is the most preferred vehicle maker in each state? [4 marks]

-- Hint: Use the window function RANK() to rank based on the count of
-- customers for each state and vehicle maker.
-- After ranking, take the vehicle maker whose rank is 1.

SELECT p.vehicle_maker , count( DISTINCT o.product_id) as totalQuantity
FROM product_t p
JOIN order_t o ON p.product_id = o.product_id
GROUP BY p.vehicle_maker
ORDER BY totalQuantity DESC
LIMIT 5;
```

Result: Passed

Query 1



Result: Passed

Query 1

Query:

```
SELECT p.vehicle_maker , count( DISTINCT o.product_id) as totalQuantity
FROM product_t p
JOIN order_t o ON p.product_id = o.product_id
GROUP BY p.vehicle_maker
ORDER BY totalQuantity DESC
LIMIT 5
```

Output:

Showing 5 rows

vehicle_maker	totalQuantity
Chevrolet	83
Ford	63
Toyota	52
Pontiac	50
Dodge	50

Observations and Insights:

1. Chevrolet leads the market by a clear margin, followed by Ford, Toyota, Pontiac, and Dodge.

2. The concentration at the top indicates strong brand affinity among customers.

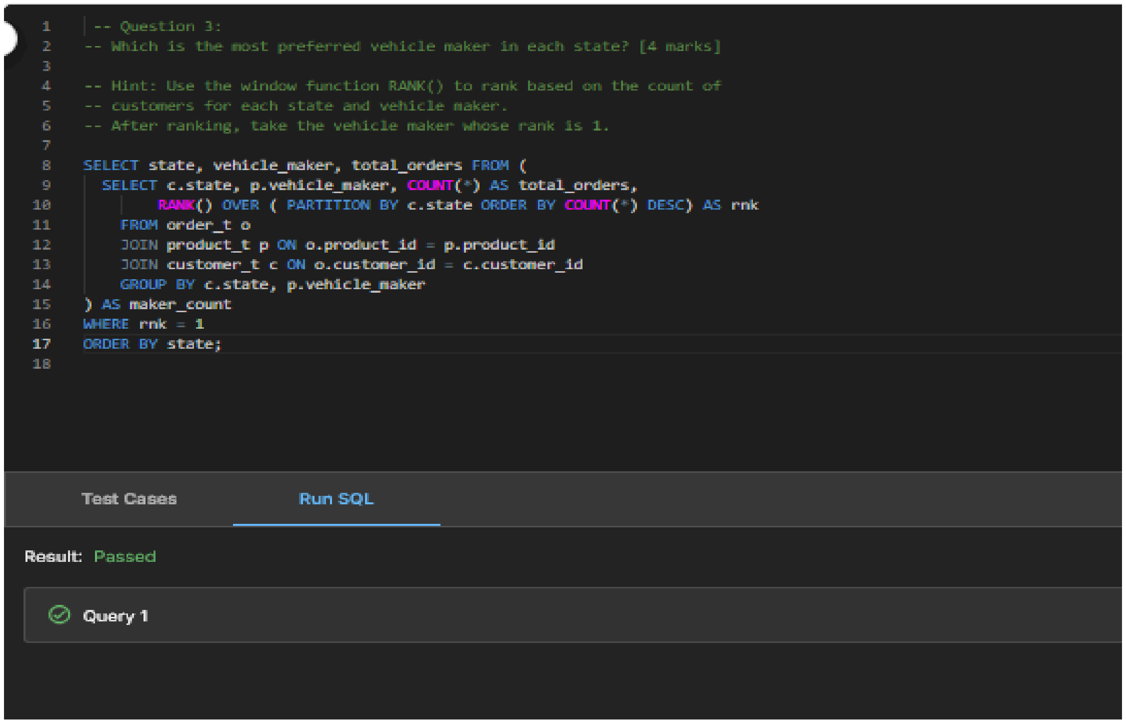
3. The spread among the remaining makers is relatively flat, implying highly fragmented demand outside the leading brands.

Question 3 :

Which is the most preferred vehicle maker in each state?

Solution Query::-

```
SELECT state, vehicle_maker, total_orders FROM (

 SELECT c.state, p.vehicle_maker, COUNT(*) AS total_orders,

      RANK() OVER ( PARTITION BY c.state ORDER BY COUNT(*) DESC) AS rnk

  FROM order_t o

  JOIN product_t p ON o.product_id = p.product_id

  JOIN customer_t c ON o.customer_id = c.customer_id

  GROUP BY c.state, p.vehicle_maker

) AS maker_count WHERE rnk = 1 ORDER BY state;
```

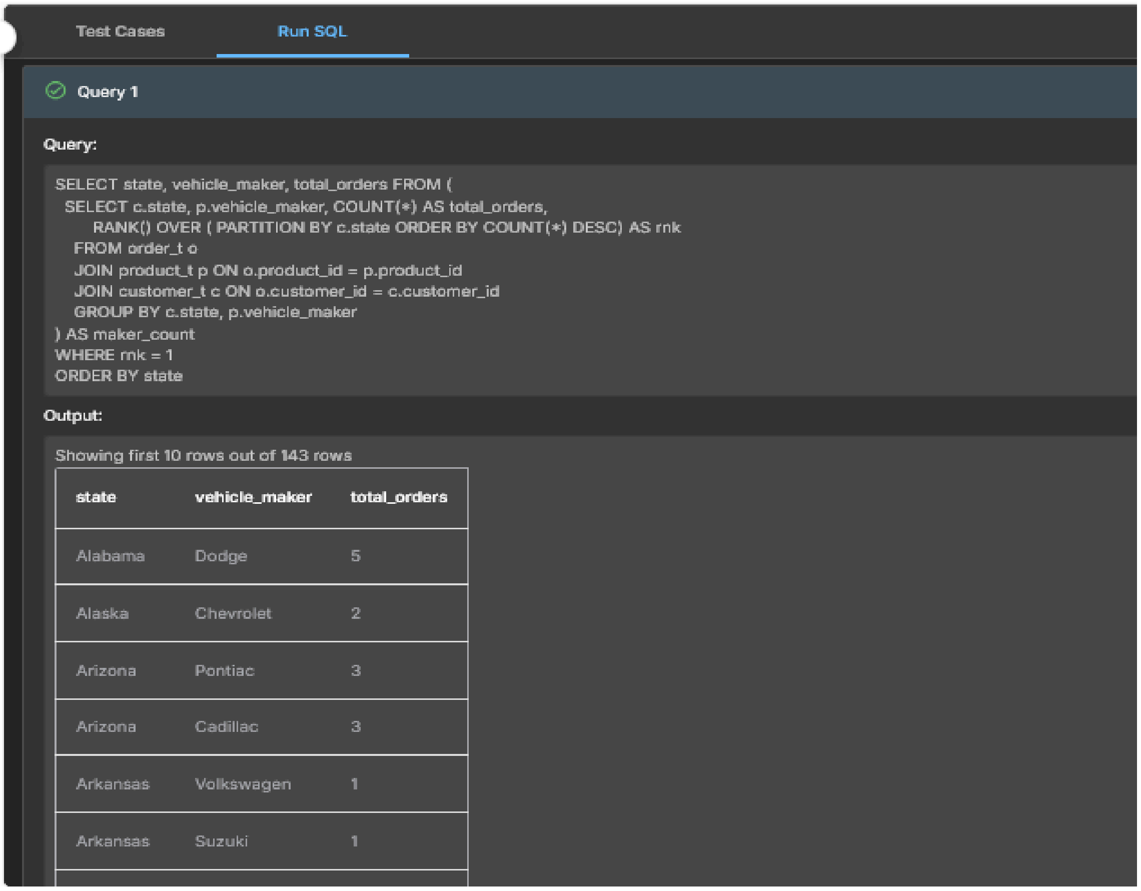


```sql
-- Question 3:
-- Which is the most preferred vehicle maker in each state? [4 marks]

-- Hint: Use the window function RANK() to rank based on the count of
-- customers for each state and vehicle maker.
-- After ranking, take the vehicle maker whose rank is 1.

SELECT state, vehicle_maker, total_orders FROM (
  SELECT c.state, p.vehicle_maker, COUNT(*) AS total_orders,
        RANK() OVER ( PARTITION BY c.state ORDER BY COUNT(*) DESC) AS rnk
    FROM order_t o
    JOIN product_t p ON o.product_id = p.product_id
    JOIN customer_t c ON o.customer_id = c.customer_id
    GROUP BY c.state, p.vehicle_maker
) AS maker_count
WHERE rnk = 1
ORDER BY state;
```

Test Cases Run SQL

Result: Passed

Query 1



Test Cases Run SQL

Query 1

Query:

```sql
SELECT state, vehicle_maker, total_orders FROM (
  SELECT c.state, p.vehicle_maker, COUNT(*) AS total_orders,
      RANK() OVER ( PARTITION BY c.state ORDER BY COUNT(*) DESC) AS rnk
    FROM order_t o
    JOIN product_t p ON o.product_id = p.product_id
    JOIN customer_t c ON o.customer_id = c.customer_id
    GROUP BY c.state, p.vehicle_maker
) AS maker_count
WHERE rnk = 1
ORDER BY state
```

Output:

Showing first 10 rows out of 143 rows

state	vehicle_maker	total_orders
Alabama	Dodge	5
Alaska	Chevrolet	2
Arizona	Pontiac	3
Arizona	Cadillac	3
Arkansas	Volkswagen	1
Arkansas	Suzuki	1

Observations and Insights:

1. Customer preferences vary substantially across states.
2. Several states display ties among manufacturers due to low order volumes.
3. States with higher order counts tend to show a clear tilt toward particular brands, which can be leveraged for more targeted marketing strategies.

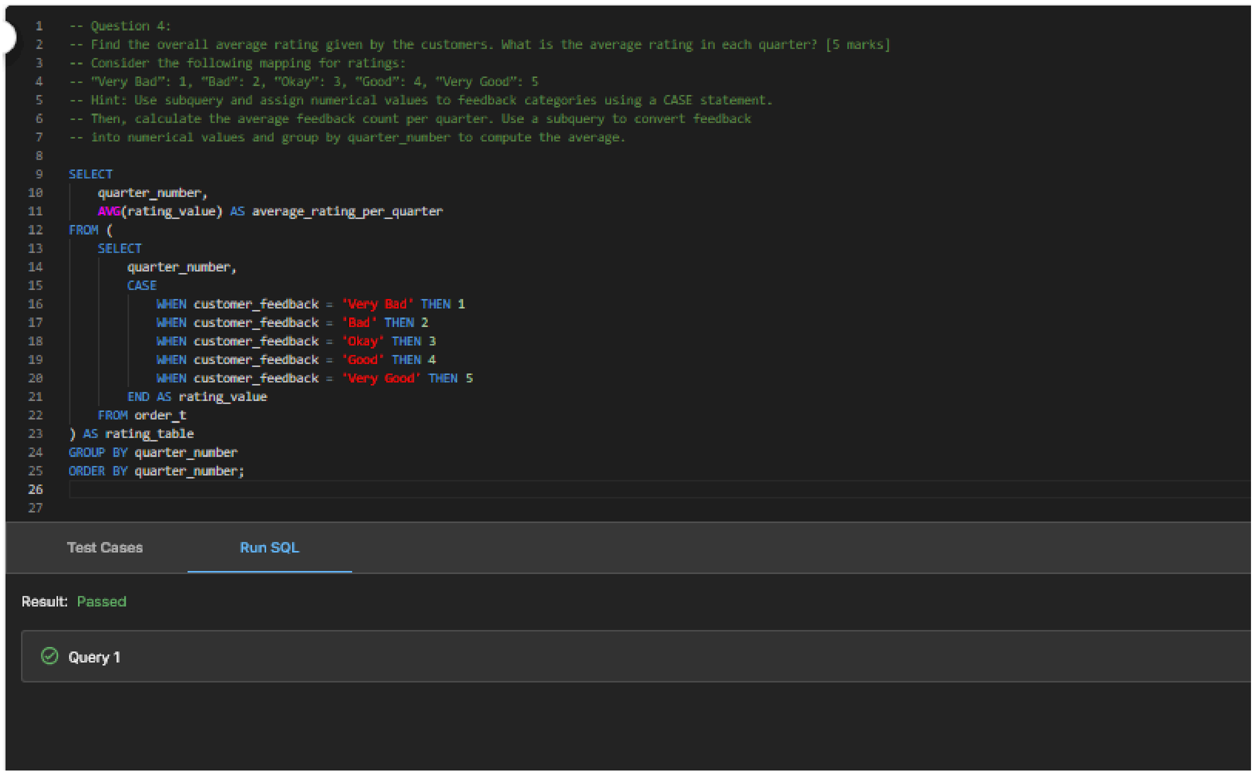
Question 4:

Find the overall average rating given by the customers. What is the average rating in each quarter

Solution Query :-

```
SELECT   quarter_number,
   AVG(rating_value) AS average_rating_per_quarter
FROM (
   SELECT  quarter_number,
      CASE
         WHEN customer_feedback = 'Very Bad' THEN 1
         WHEN customer_feedback = 'Bad' THEN 2
         WHEN customer_feedback = 'Okay' THEN 3
```

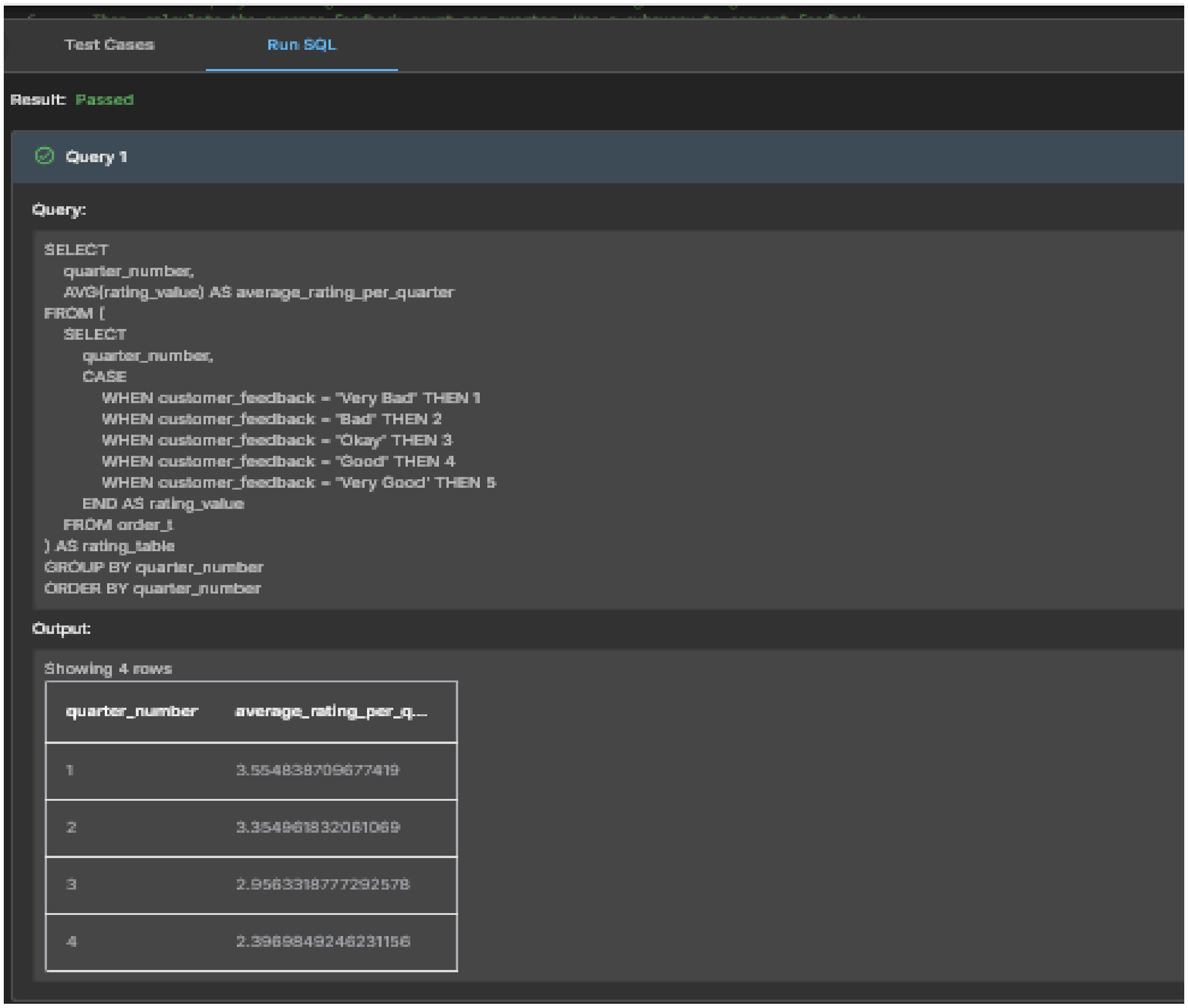
```
            WHEN customer_feedback = 'Good' THEN 4

            WHEN customer_feedback = 'Very Good' THEN 5

        END AS rating_value  FROM order_t

) AS rating_table

GROUP BY quarter_number

ORDER BY quarter_number;
```



```
-- Question 4:
-- Find the overall average rating given by the customers. What is the average rating in each quarter? [5 marks]
-- Consider the following mapping for ratings:
-- "Very Bad": 1, "Bad": 2, "Okay": 3, "Good": 4, "Very Good": 5
-- Hint: Use subquery and assign numerical values to feedback categories using a CASE statement.
-- Then, calculate the average feedback count per quarter. Use a subquery to convert feedback
-- into numerical values and group by quarter_number to compute the average.

SELECT
    quarter_number,
    AVG(rating_value) AS average_rating_per_quarter
FROM (
    SELECT
        quarter_number,
        CASE
            WHEN customer_feedback = 'Very Bad' THEN 1
            WHEN customer_feedback = 'Bad' THEN 2
            WHEN customer_feedback = 'Okay' THEN 3
            WHEN customer_feedback = 'Good' THEN 4
            WHEN customer_feedback = 'Very Good' THEN 5
        END AS rating_value
    FROM order_t
) AS rating_table
GROUP BY quarter_number
ORDER BY quarter_number;
```

Result: Passed

Query 1



Test Cases Run SQL

Result: Passed

Query 1

Query:

```
SELECT
   quarter_number,
   AVG(rating_value) AS average_rating_per_quarter
FROM (
   SELECT
      quarter_number,
      CASE
         WHEN customer_feedback = 'Very Bad' THEN 1
         WHEN customer_feedback = 'Bad' THEN 2
         WHEN customer_feedback = 'Okay' THEN 3
         WHEN customer_feedback = 'Good' THEN 4
         WHEN customer_feedback = 'Very Good' THEN 5
      END AS rating_value
   FROM order_t
) AS rating_table
GROUP BY quarter_number
ORDER BY quarter_number
```

Output:

Showing 4 rows

quarter_number	average_rating_per_q...
1	3.554838709677419
2	3.354961832061069
3	2.9563318777292578
4	2.3969849246231156

Observations and Insights:

1. Customer satisfaction declines steadily across quarters.

2. Ratings initially indicate moderate satisfaction but progressively fall, with the final quarter showing a noticeably weaker performance.

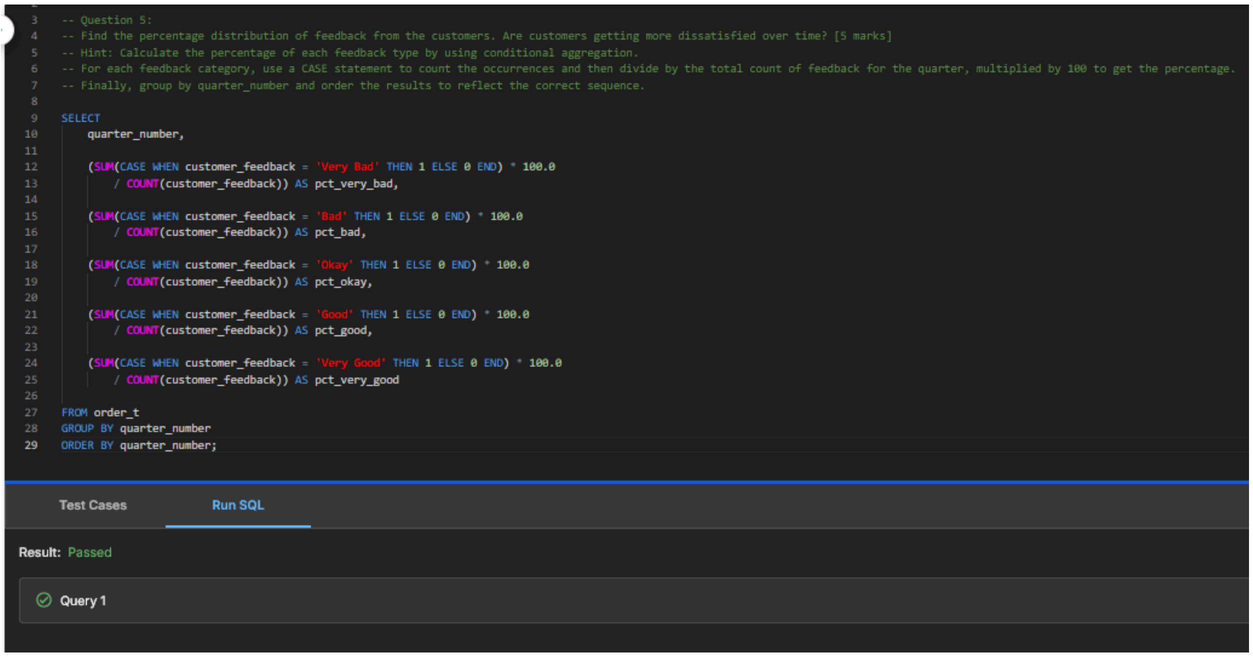
3. This consistent downward trend raises operational concerns and hints at service-level issues developing over time.

Question 5:

Find the percentage distribution of feedback from the customers. Are customers getting more dissatisfied over time?

Solution Query:-

```
SELECT quarter_number,

    (SUM(CASE WHEN customer_feedback = 'Very Bad' THEN 1 ELSE 0 END) * 100.0

        / COUNT(customer_feedback)) AS pct_very_bad,

    (SUM(CASE WHEN customer_feedback = 'Bad' THEN 1 ELSE 0 END) * 100.0

        / COUNT(customer_feedback)) AS pct_bad,

    (SUM(CASE WHEN customer_feedback = 'Okay' THEN 1 ELSE 0 END) * 100.0

        / COUNT(customer_feedback)) AS pct_okay,

    (SUM(CASE WHEN customer_feedback = 'Good' THEN 1 ELSE 0 END) * 100.0

        / COUNT(customer_feedback)) AS pct_good,

  (SUM(CASE WHEN customer_feedback = 'Very Good' THEN 1 ELSE 0 END) * 100.0
/ COUNT(customer_feedback)) AS pct_very_good

FROM order_t

GROUP BY quarter_number ORDER BY quarter_number;
```

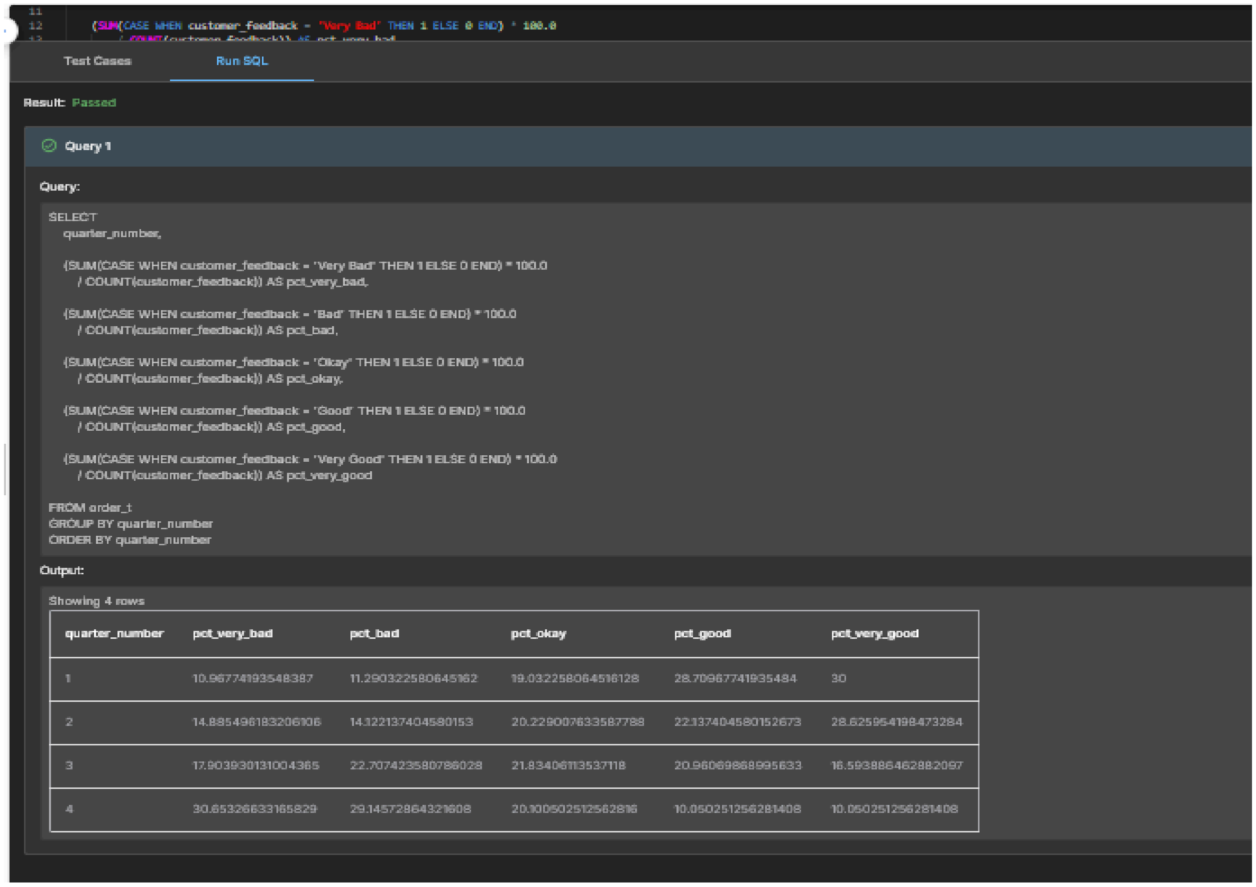


```sql
-- Question 5:
-- Find the percentage distribution of feedback from the customers. Are customers getting more dissatisfied over time? [5 marks]
-- Hint: Calculate the percentage of each feedback type by using conditional aggregation.
-- For each feedback category, use a CASE statement to count the occurrences and then divide by the total count of feedback for the quarter, multiplied by 100 to get the percentage.
-- Finally, group by quarter_number and order the results to reflect the correct sequence.

SELECT
    quarter_number,

    (SUM(CASE WHEN customer_feedback = 'Very Bad' THEN 1 ELSE 0 END) * 100.0
        / COUNT(customer_feedback)) AS pct_very_bad,

    (SUM(CASE WHEN customer_feedback = 'Bad' THEN 1 ELSE 0 END) * 100.0
        / COUNT(customer_feedback)) AS pct_bad,

    (SUM(CASE WHEN customer_feedback = 'Okay' THEN 1 ELSE 0 END) * 100.0
        / COUNT(customer_feedback)) AS pct_okay,

    (SUM(CASE WHEN customer_feedback = 'Good' THEN 1 ELSE 0 END) * 100.0
        / COUNT(customer_feedback)) AS pct_good,

    (SUM(CASE WHEN customer_feedback = 'Very Good' THEN 1 ELSE 0 END) * 100.0
        / COUNT(customer_feedback)) AS pct_very_good

FROM order_t
GROUP BY quarter_number
ORDER BY quarter_number;
```

Test Cases Run SQL

Result: Passed

Query 1



```sql
    (SUM(CASE WHEN customer_feedback = 'Very Bad' THEN 1 ELSE 0 END) * 100.0
```

Test Cases Run SQL

Result: Passed

Query 1

Query:

```sql
SELECT
    quarter_number,

    (SUM(CASE WHEN customer_feedback = 'Very Bad' THEN 1 ELSE 0 END) * 100.0
        / COUNT(customer_feedback)) AS pct_very_bad,

    (SUM(CASE WHEN customer_feedback = 'Bad' THEN 1 ELSE 0 END) * 100.0
        / COUNT(customer_feedback)) AS pct_bad,

    (SUM(CASE WHEN customer_feedback = 'Okay' THEN 1 ELSE 0 END) * 100.0
        / COUNT(customer_feedback)) AS pct_okay,

    (SUM(CASE WHEN customer_feedback = 'Good' THEN 1 ELSE 0 END) * 100.0
        / COUNT(customer_feedback)) AS pct_good,

    (SUM(CASE WHEN customer_feedback = 'Very Good' THEN 1 ELSE 0 END) * 100.0
        / COUNT(customer_feedback)) AS pct_very_good

FROM order_t
GROUP BY quarter_number
ORDER BY quarter_number
```

Output:

Showing 4 rows

quarter_number	pct_very_bad	pct_bad	pct_okay	pct_good	pct_very_good
1	10.967741935483870	11.290322580645162	19.032258064516128	28.70967741935484	30
2	14.885496183206106	14.122137404580153	20.229007633587788	22.137404580152673	28.625954198473284
3	17.903930131004365	22.707423580786028	21.83406113537118	20.96069868995633	16.593886462882097
4	30.65326633165829	29.14572864321608	20.100502512562816	10.050251256281408	10.050251256281408

Observations and Insights:

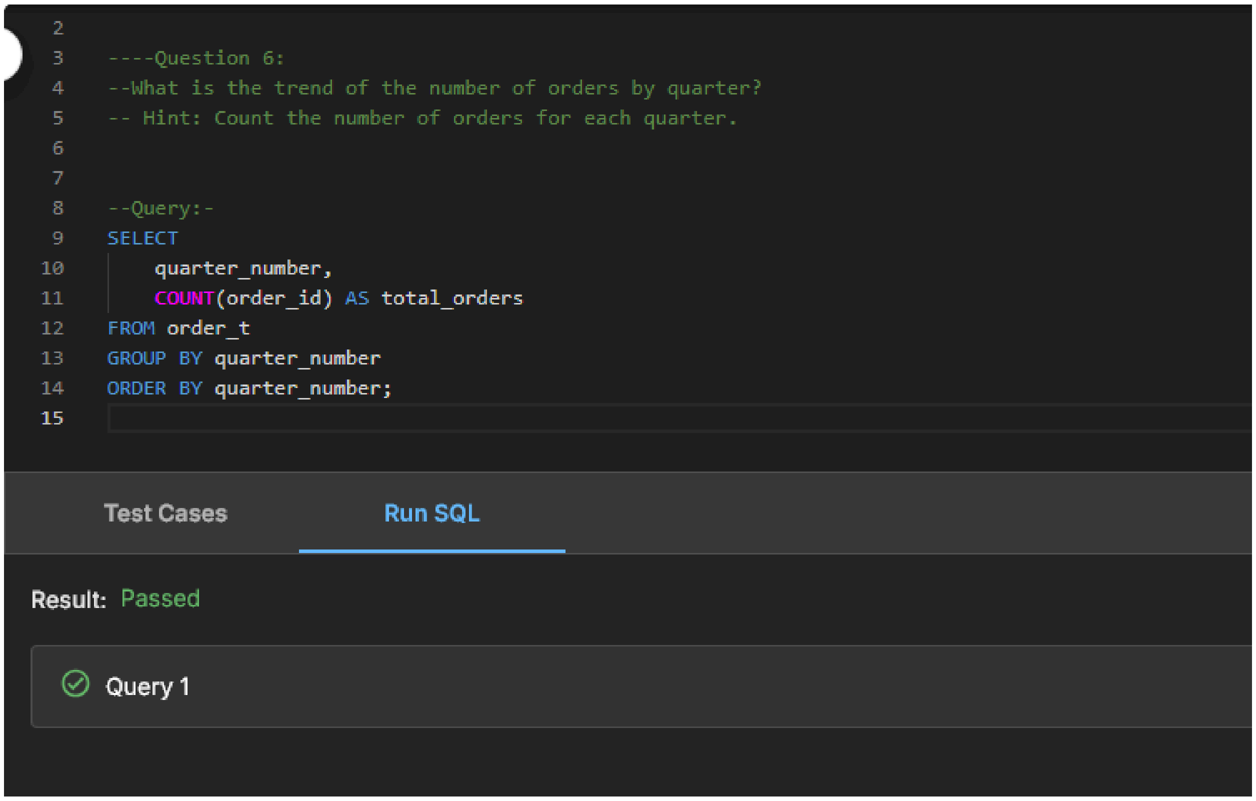
1. Negative sentiment increases sharply quarter over quarter.
2. Positive ratings such as “Very Good” and “Good” decline substantially by Q4.
3. Neutral feedback remains mostly stable.
4. Overall, the data points toward growing customer dissatisfaction, aligning with the weakening sales performance.

Question 6:

What is the trend of the number of orders by quarter?

Solution Query:-

```
SELECT
   quarter_number,
   COUNT(order_id) AS total_orders
FROM order_t
GROUP BY quarter_number
ORDER BY quarter_number;
```

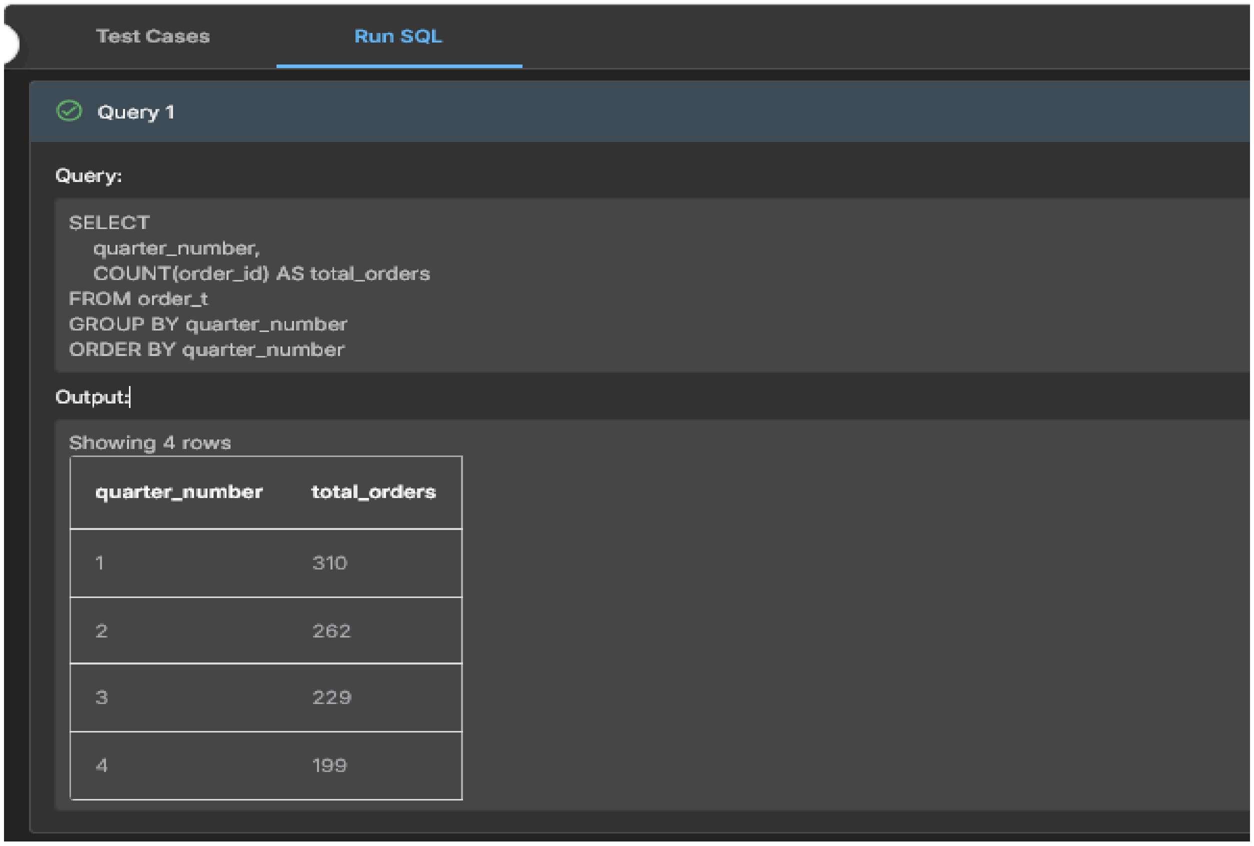


```
----Question 6:
--What is the trend of the number of orders by quarter?
-- Hint: Count the number of orders for each quarter.


--Query:-
SELECT
    quarter_number,
    COUNT(order_id) AS total_orders
FROM order_t
GROUP BY quarter_number
ORDER BY quarter_number;
```

Test Cases Run SQL

Result: Passed

Query 1



Test Cases Run SQL

Query 1

Query:

```
SELECT
  quarter_number,
  COUNT(order_id) AS total_orders
FROM order_t
GROUP BY quarter_number
ORDER BY quarter_number
```

Output:

Showing 4 rows

quarter_number	total_orders
1	310
2	262
3	229
4	199

Observations and Insights:

1. Order volume consistently declines from Q1 through Q4.

2. The initial quarter reflects relatively strong engagement, but the continued drop indicates weakening retention and conversion.

3. The decline in orders parallels falling customer satisfaction.

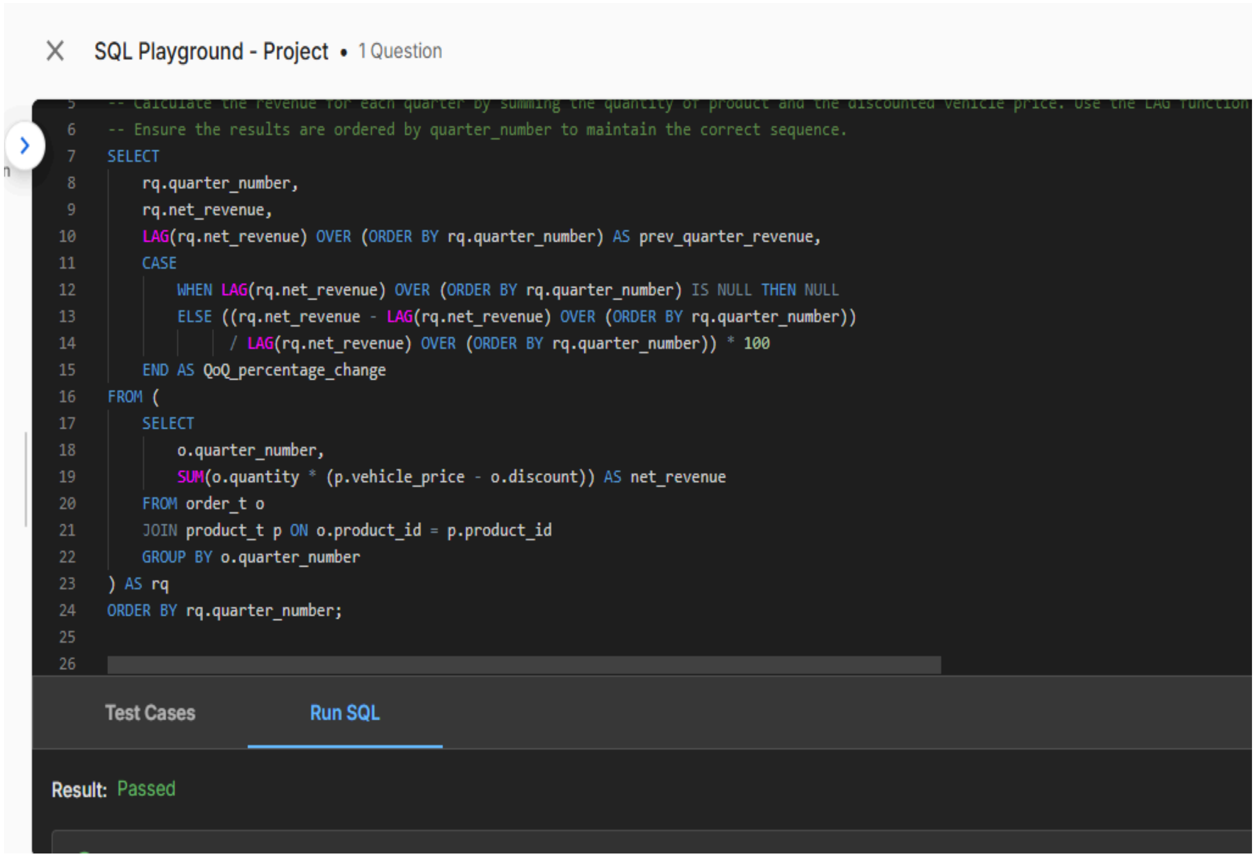
Question 7:

Calculate the net revenue generated by the company. What is the quarter-over-quarter % change in net revenue?

Solution Query:-

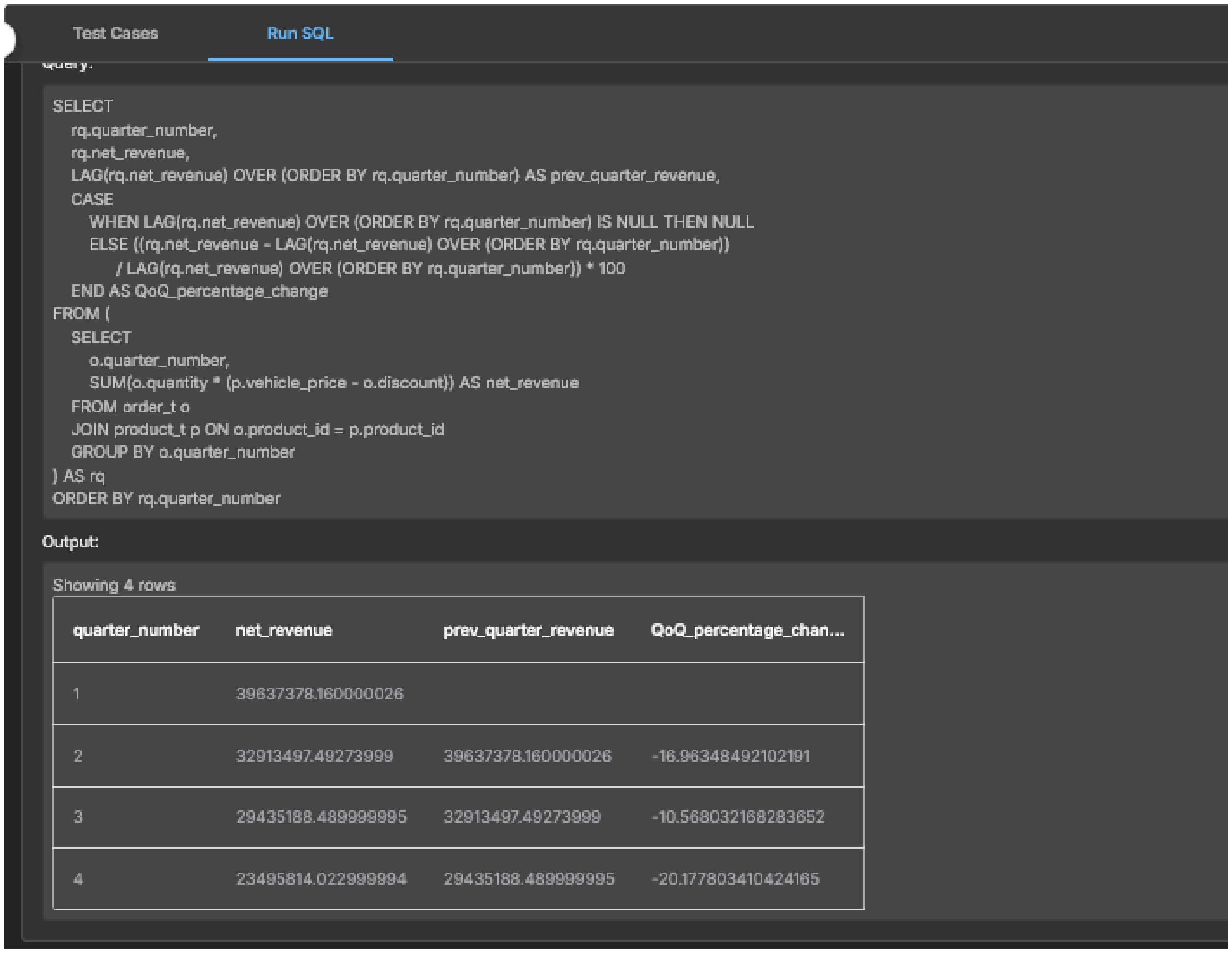
```
SELECT   rq.quarter_number,  rq.net_revenue,
   LAG(rq.net_revenue) OVER (ORDER BY rq.quarter_number) AS
prev_quarter_revenue,
   CASE
      WHEN LAG(rq.net_revenue) OVER (ORDER BY rq.quarter_number) IS NULL
THEN NULL
      ELSE ((rq.net_revenue - LAG(rq.net_revenue) OVER (ORDER BY
rq.quarter_number))
           / LAG(rq.net_revenue) OVER (ORDER BY rq.quarter_number)) * 100
```

```
    END AS QoQ_percentage_change
FROM (
    SELECT
        o.quarter_number,
        SUM(o.quantity * (p.vehicle_price - o.discount)) AS net_revenue
    FROM order_t o
    JOIN product_t p ON o.product_id = p.product_id
    GROUP BY o.quarter_number
) AS rq
ORDER BY rq.quarter_number;
```


SQL Playground - Project • 1 Question

```
-- Calculate the revenue for each quarter by summing the quantity of product and the discounted vehicle price. Use the LAG function
-- Ensure the results are ordered by quarter_number to maintain the correct sequence.
SELECT
    rq.quarter_number,
    rq.net_revenue,
    LAG(rq.net_revenue) OVER (ORDER BY rq.quarter_number) AS prev_quarter_revenue,
    CASE
        WHEN LAG(rq.net_revenue) OVER (ORDER BY rq.quarter_number) IS NULL THEN NULL
        ELSE ((rq.net_revenue - LAG(rq.net_revenue) OVER (ORDER BY rq.quarter_number))
              / LAG(rq.net_revenue) OVER (ORDER BY rq.quarter_number)) * 100
    END AS QoQ_percentage_change
FROM (
    SELECT
        o.quarter_number,
        SUM(o.quantity * (p.vehicle_price - o.discount)) AS net_revenue
    FROM order_t o
    JOIN product_t p ON o.product_id = p.product_id
    GROUP BY o.quarter_number
) AS rq
ORDER BY rq.quarter_number;
```

Result: Passed



```
SELECT
   rq.quarter_number,
   rq.net_revenue,
   LAG(rq.net_revenue) OVER (ORDER BY rq.quarter_number) AS prev_quarter_revenue,
   CASE
      WHEN LAG(rq.net_revenue) OVER (ORDER BY rq.quarter_number) IS NULL THEN NULL
      ELSE ((rq.net_revenue - LAG(rq.net_revenue) OVER (ORDER BY rq.quarter_number))
           / LAG(rq.net_revenue) OVER (ORDER BY rq.quarter_number)) * 100
   END AS QoQ_percentage_change
FROM (
   SELECT
      o.quarter_number,
      SUM(o.quantity * (p.vehicle_price - o.discount)) AS net_revenue
   FROM order_t o
   JOIN product_t p ON o.product_id = p.product_id
   GROUP BY o.quarter_number
) AS rq
ORDER BY rq.quarter_number
```

Output:

Showing 4 rows

quarter_number	net_revenue	prev_quarter_revenue	QoQ_percentage_chan...
1	39637378.160000026		
2	32913497.49273999	39637378.160000026	-16.96348492102191
3	29435188.489999995	32913497.49273999	-10.568032168283652
4	23495814.022999994	29435188.489999995	-20.177803410424165

Observations and Insights:

1. Net revenue mirrors the decline in order volume, with Q1 significantly outperforming subsequent quarters.

2. The quarter-over-quarter drop becomes steeper in the latter half of the year.

3. Revenue deterioration is directly aligned with customer experience issues and operational delays.

Question 8:

What is the trend of net revenue and orders by quarters?

Solution Query:-

SELECT o.quarter_number,

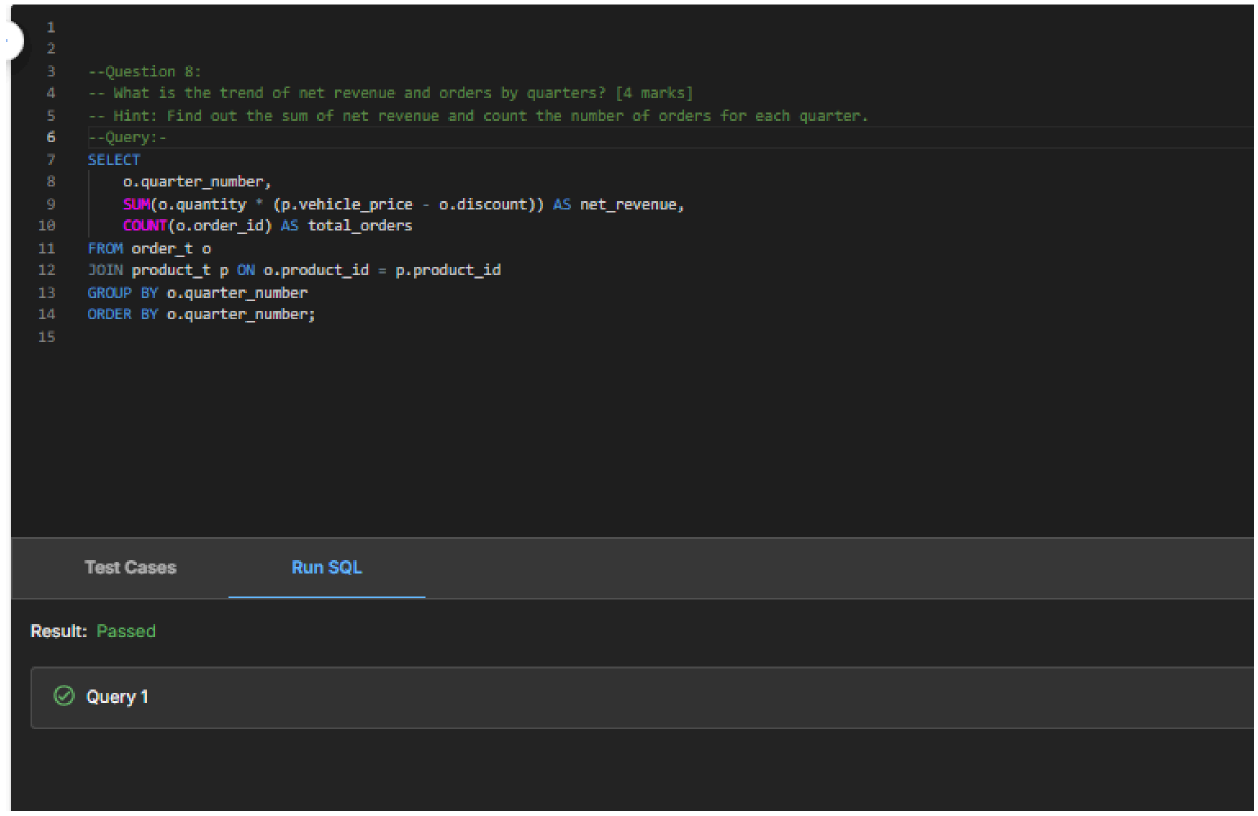
SUM(o.quantity * (p.vehicle_price - o.discount)) AS net_revenue,

COUNT(o.order_id) AS total_orders FROM order_t o

JOIN product_t p ON o.product_id = p.product_id

GROUP BY o.quarter_number

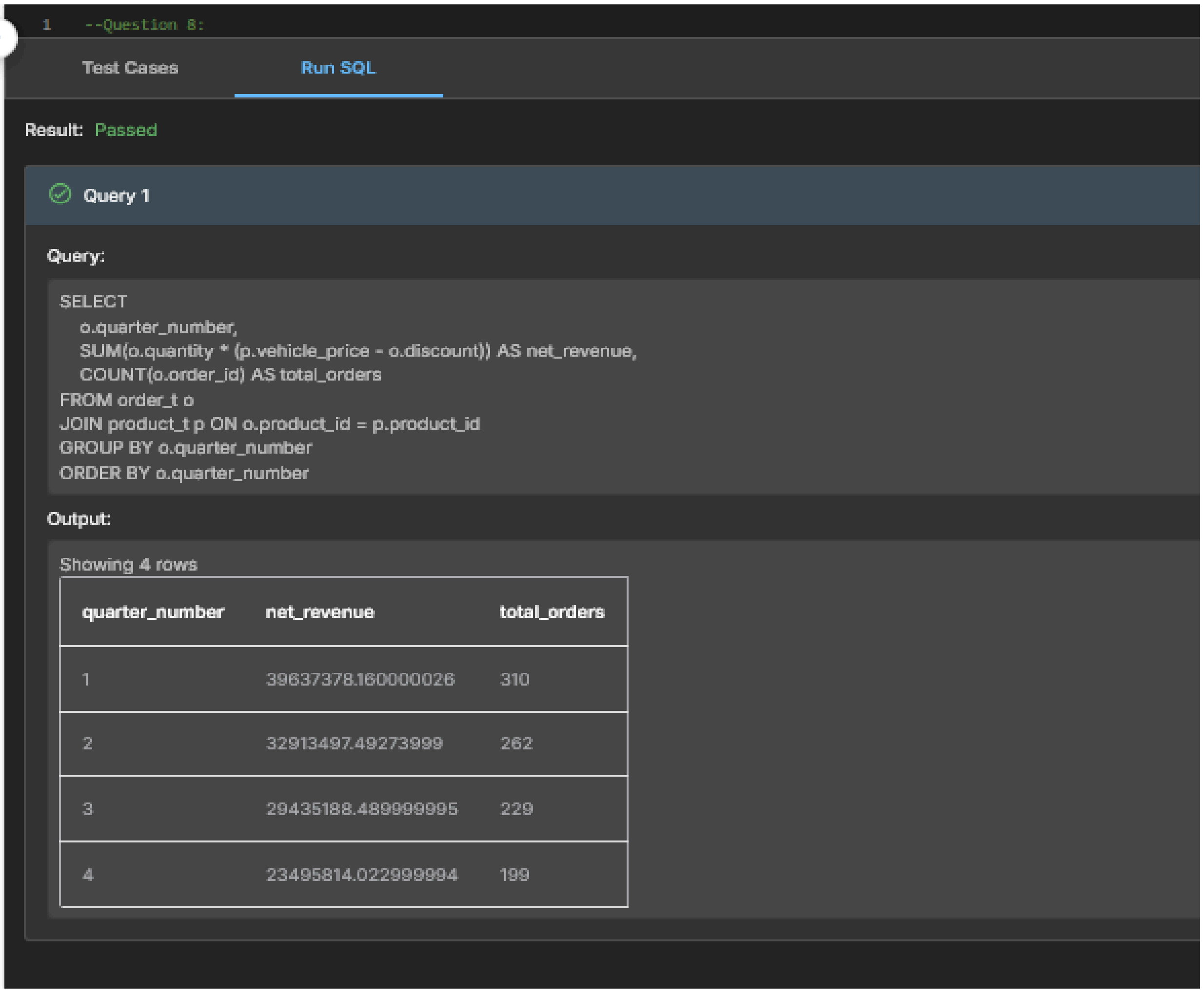
ORDER BY o.quarter_number;



```


--Question 8:
-- What is the trend of net revenue and orders by quarters? [4 marks]
-- Hint: Find out the sum of net revenue and count the number of orders for each quarter.
--Query:-
SELECT
    o.quarter_number,
    SUM(o.quantity * (p.vehicle_price - o.discount)) AS net_revenue,
    COUNT(o.order_id) AS total_orders
FROM order_t o
JOIN product_t p ON o.product_id = p.product_id
GROUP BY o.quarter_number
ORDER BY o.quarter_number;

```

Result: Passed

Query 1



Query:

```
SELECT
   o.quarter_number,
   SUM(o.quantity * (p.vehicle_price - o.discount)) AS net_revenue,
   COUNT(o.order_id) AS total_orders
FROM order_t o
JOIN product_t p ON o.product_id = p.product_id
GROUP BY o.quarter_number
ORDER BY o.quarter_number
```

Output:

Showing 4 rows

quarter_number	net_revenue	total_orders
1	39637378.160000026	310
2	32913497.49273999	262
3	29435188.489999995	229
4	23495814.022999994	199

Observations and Insights:

1. Both indicators orders and revenue follow the same downward trajectory.

2. The year begins strongly but concludes with notably weaker performance.

3. The synchronized decline strongly implies that customer dissatisfaction is translating into lost sales.

Question 9:

What is the average discount offered for different types of credit cards?

Solution Query:-

SELECT c.credit_card_type,

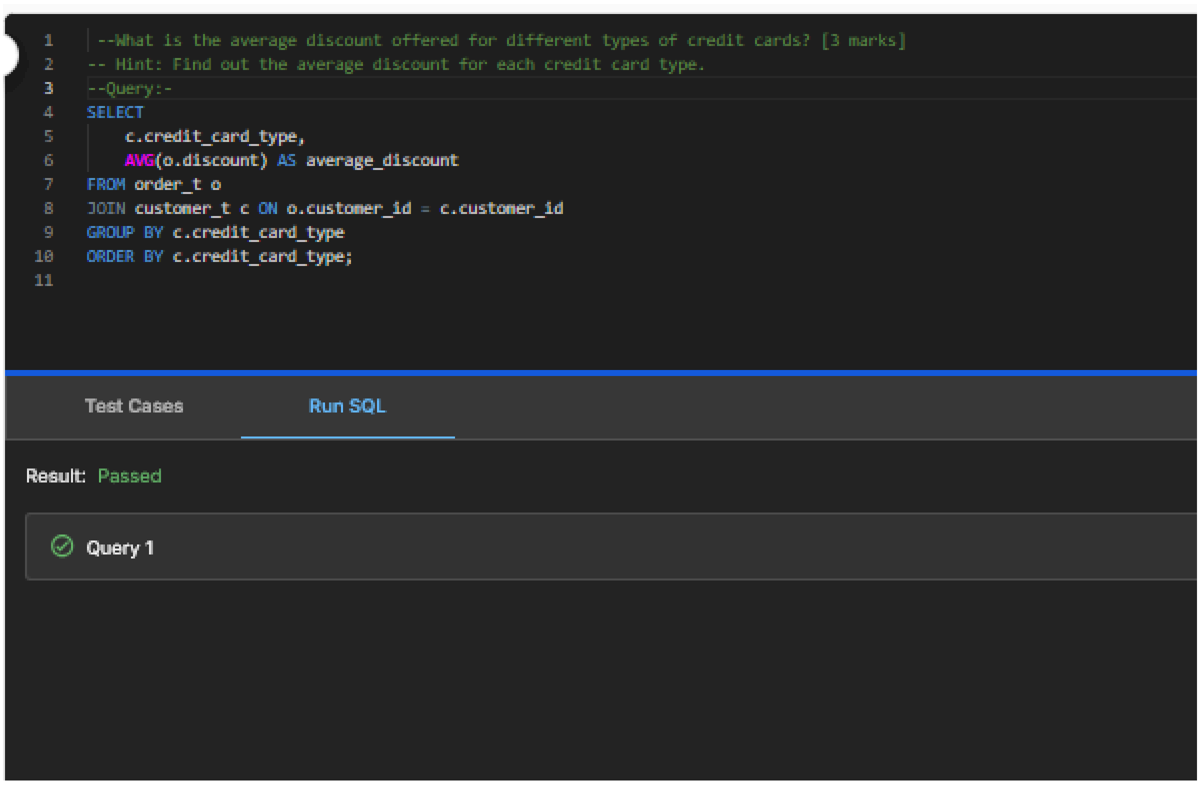
AVG(o.discount) AS average_discount

FROM order_t o

JOIN customer_t c ON o.customer_id = c.customer_id

GROUP BY c.credit_card_type

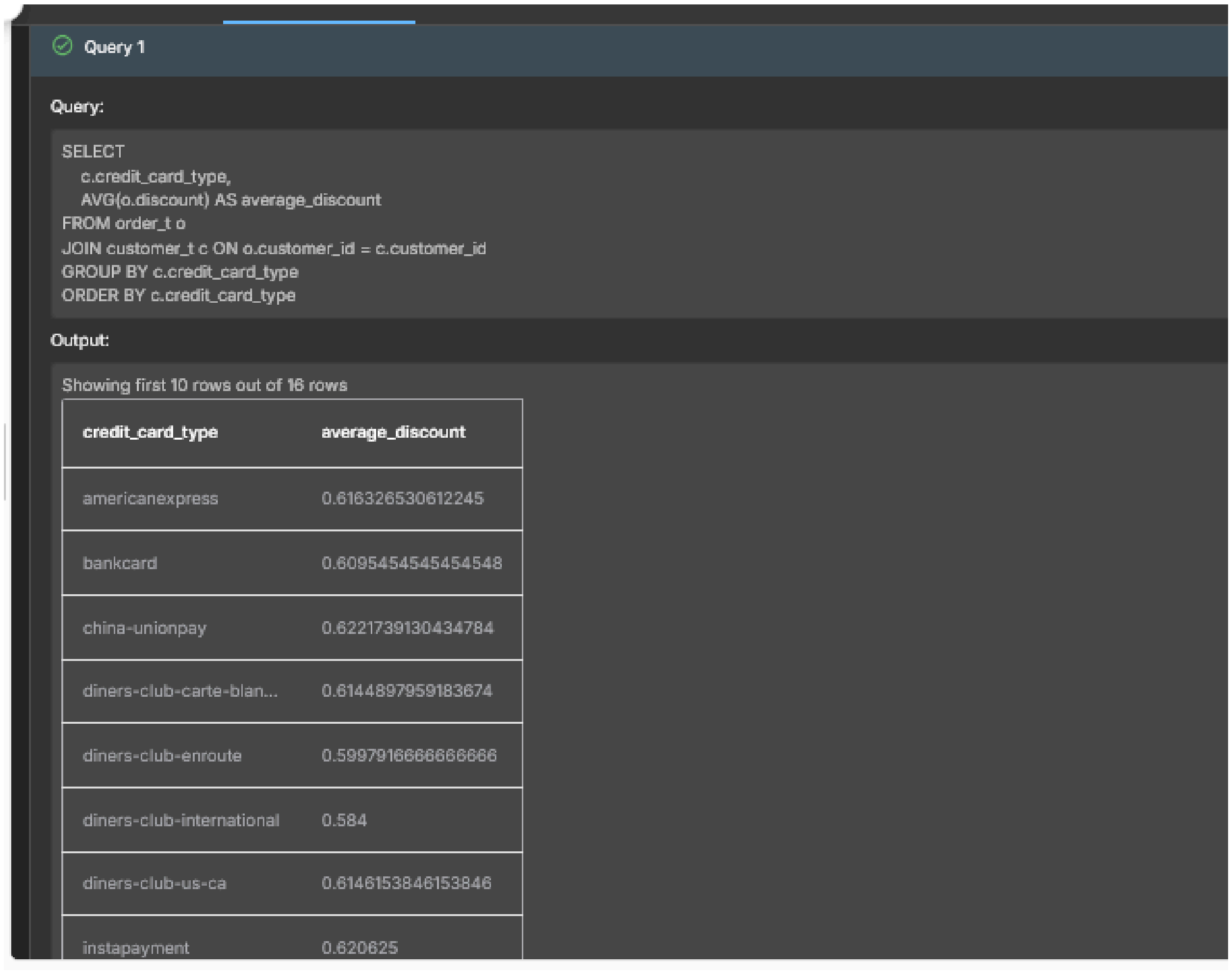
ORDER BY c.credit_card_type;



```
--What is the average discount offered for different types of credit cards? [3 marks]
-- Hint: Find out the average discount for each credit card type.
--Query:-
SELECT
    c.credit_card_type,
    AVG(o.discount) AS average_discount
FROM order_t o
JOIN customer_t c ON o.customer_id = c.customer_id
GROUP BY c.credit_card_type
ORDER BY c.credit_card_type;
```

Result: Passed

Query 1



Query:

```
SELECT
   c.credit_card_type,
   AVG(o.discount) AS average_discount
FROM order_t o
JOIN customer_t c ON o.customer_id = c.customer_id
GROUP BY c.credit_card_type
ORDER BY c.credit_card_type
```

Output:

Showing first 10 rows out of 16 rows

credit_card_type	average_discount
americanexpress	0.616326530612245
bankcard	0.6095454545454548
china-unionpay	0.6221739130434784
diners-club-carte-blan...	0.6144897959183674
diners-club-enroute	0.5997916666666666
diners-club-international	0.584
diners-club-us-ca	0.6146153846153846
instapayment	0.620625

Observations and Insights:

1. Discounts are almost the same for all card types.
2. Only small differences appear between card categories.
3. This shows discounts are not based on payment method.
4. Discount strategy is likely driven by products or general promotions.

Question 10:

What is the average time taken to ship the placed orders for each quarter?

Solution Query:

SELECT quarter_number,

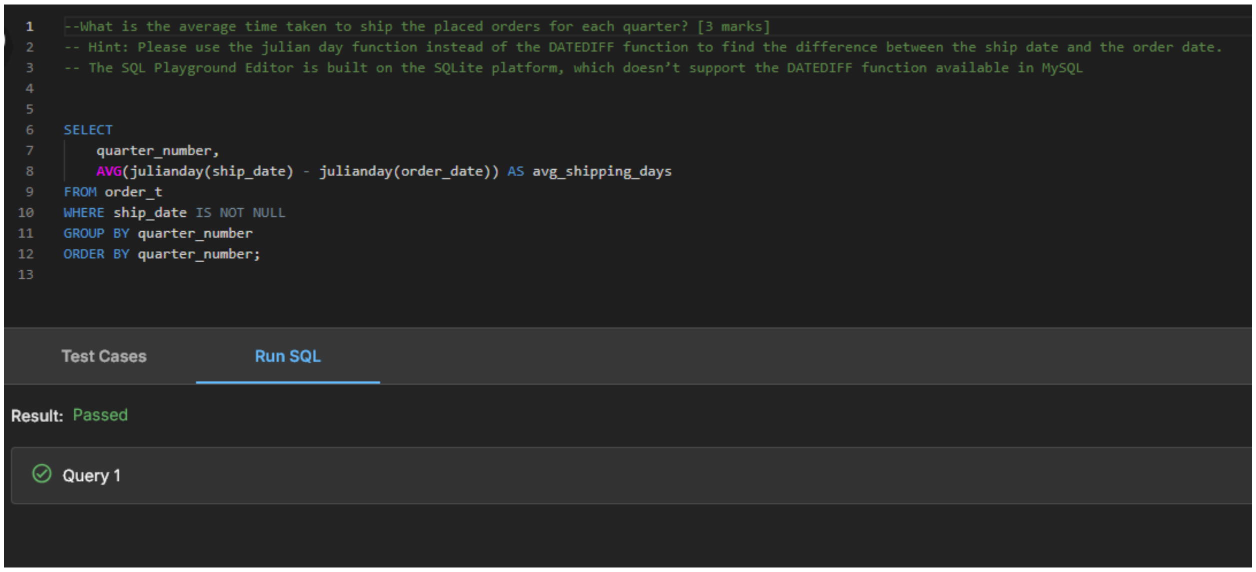
AVG(julianday(ship_date) - julianday(order_date)) AS avg_shipping_days

FROM order_t

WHERE ship_date IS NOT NULL

GROUP BY quarter_number

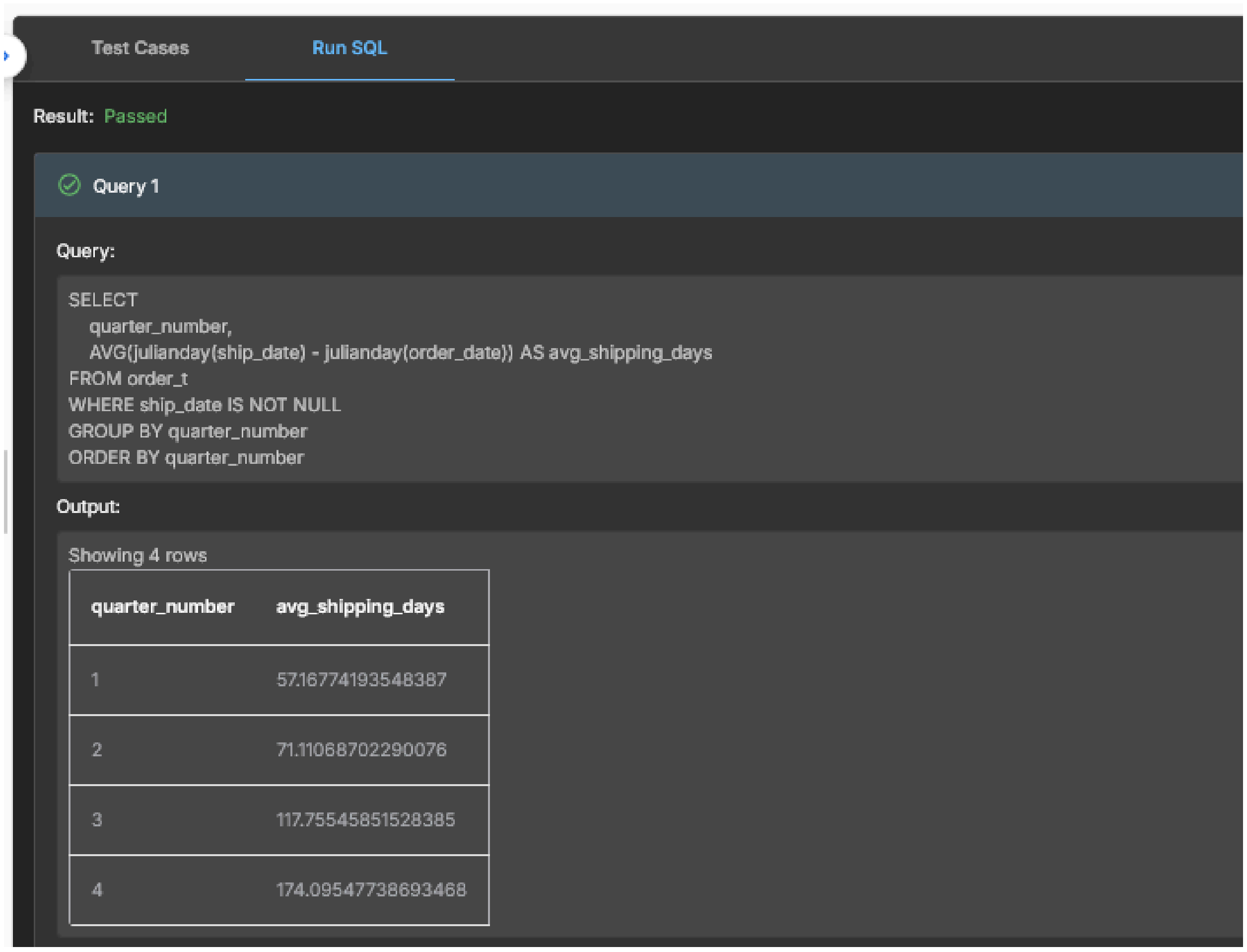
ORDER BY quarter_number;



```
--What is the average time taken to ship the placed orders for each quarter? [3 marks]
-- Hint: Please use the julian day function instead of the DATEDIFF function to find the difference between the ship date and the order date.
-- The SQL Playground Editor is built on the SQLite platform, which doesn’t support the DATEDIFF function available in MySQL


SELECT
    quarter_number,
    AVG(julianday(ship_date) - julianday(order_date)) AS avg_shipping_days
FROM order_t
WHERE ship_date IS NOT NULL
GROUP BY quarter_number
ORDER BY quarter_number;
```

Result: Passed

Query 1



Result: Passed

Query 1

Query:

```
SELECT
   quarter_number,
   AVG(julianday(ship_date) - julianday(order_date)) AS avg_shipping_days
FROM order_t
WHERE ship_date IS NOT NULL
GROUP BY quarter_number
ORDER BY quarter_number
```

Output:

Showing 4 rows

quarter_number	avg_shipping_days
1	57.16774193548387
2	71.11068702290076
3	117.75545851528385
4	174.09547738693468

Observations and Insights:

1. Shipping times get much longer with each quarter.

2. The average delivery time becomes extremely slow by Q4.

3. This delay likely frustrates customers and reduces satisfaction.

4. Longer shipping times appear strongly linked to rising complaints.

Business Metrics Overview

Total Revenue	Total Orders	Total Customers	Average Rating
125481878.16	1000	994	3.14

Last Quarter Revenue	Last quarter Orders	Average Days to Ship	% Good Feedback
23495814.02	199	98	44.1

Business Recommendations

Fix shipping delays

Delivery times rose from 57 to 174 days. Improve logistics, carriers, or inventory to restore customer satisfaction.

Improve customer experience first

Ratings dropped and negative feedback doubled. Focus on faster delivery, accurate orders, and better communication before pushing marketing.

Prioritize top-performing brands

Chevrolet and Ford drive most sales. Increase inventory and promotions for strong brands; reduce stock of weaker ones.

Target high-activity states

States like California and Florida show strong demand. Use targeted marketing, loyalty perks, and faster shipping to boost orders.

Thankyou

PGP DSA 16 NOV 2025

ANJU SAINI