Started on	Wednesday, 29 October 2025, 3:52 PM
State	Finished
Completed on	Wednesday, 29 October 2025, 3:58 PM
Time taken	6 mins 1 sec
Marks	10.00/10.00
Grade	**100.00** out of 100.00

Question 1

Complete

Mark 1.00 out of 1.00

In MySQL, what is the difference between RANK() and DENSE_RANK()?

- ○ a. DENSE_RANK() can't be used with PARTITION BY
- ◉ b. RANK() skips ranks for ties, while DENSE_RANK() doesn't
- ○ c. DENSE_RANK() skips ranks for ties, while RANK() doesn't
- ○ d. Both behave identically

Question 2

Complete

Mark 1.00 out of 1.00

Consider this query:

```
SELECT emp_name, department, salary,
CASE
  WHEN salary > AVG(salary) OVER (PARTITION BY department) THEN 'Above Avg'
  ELSE 'Below Avg'
END AS performance
FROM employees;
```

What does this query compute?

- ○ a. Calculates cumulative salary rank
- ◉ b. Compares each employee's salary with department average using a window function
- ○ c. Employees grouped by salary brackets
- ○ d. Finds total salary per department

Question 3

Complete

Mark 1.00 out of 1.00

What is the difference between ROW_NUMBER() and RANK() functions in MySQL?

- ◉ a. ROW_NUMBER() gives unique sequential numbers, RANK() gives same numbers for ties
- ○ b. ROW_NUMBER() cannot be used with ORDER BY
- ○ c. RANK() gives unique numbers, ROW_NUMBER() gives same numbers for ties
- ○ d. Both are identical

Question 4

Complete

Mark 1.00 out of 1.00

What is the purpose of the PARTITION BY clause in a window function?

- ○ a. To sort results in ascending order
- ○ b. To filter rows using a condition
- ● c. To divide result sets into groups before applying the window function
- ○ d. To limit the number of returned rows

Question 5

Complete

Mark 1.00 out of 1.00

What will the following query compute?

```
SELECT emp_id, department, salary,
SUM(salary) OVER (PARTITION BY department) AS dept_total
FROM employees;
```

- ○ a. The cumulative salary ordered by department
- ● b. The total salary per department
- ○ c. The average salary across all employees
- ○ d. The total salary of all employees

Question 6

Complete

Mark 1.00 out of 1.00

What will the following query return?

```
SELECT emp_name, salary,
RANK() OVER (ORDER BY salary DESC) AS position
FROM employees;
```

- ○ a. A running total of salaries
- ○ b. A unique rank even for equal salaries
- ● c. The same rank for equal salaries, with gaps in the sequence
- ○ d. The average salary per department

Question 7

Complete

Mark 1.00 out of 1.00

What will this CASE statement output when salary = 80000?

```
CASE
  WHEN salary > 100000 THEN 'High'
  WHEN salary >= 70000 THEN 'Medium'
  ELSE 'Low'
END
```

- ○ a.   Low
- ○ b.   High
- ○ c.   NULL
- ◉ d.   Medium

Question 8

Complete

Mark 1.00 out of 1.00

Which clause is mandatory when using a window function like RANK()?

- ○ a.   ORDER BY
- ○ b.   GROUP BY
- ○ c.   LIMIT
- ◉ d.   OVER()

Question 9

Complete

Mark 1.00 out of 1.00

Which of the following correctly defines a window function in MySQL?

- ○ a.   A function that aggregates rows into a single result for each group
- ◉ b.   A function that performs calculations across a set of rows related to the current row
- ○ c.   A function that creates temporary tables for joins
- ○ d.   A function that partitions data into multiple databases

Question 10

Complete

Mark 1.00 out of 1.00

Which of the following is TRUE about CASE statements in MySQL?

- ○ a.   CASE cannot be nested
- ○ b.   CASE can only compare numeric columns
- ○ c.   CASE must always end with an ELSE clause
- ◉ d.   CASE can be used both in SELECT and ORDER BY clauses