

1. What does this return?

```
SELECT LENGTH('SQL\nMCQ');
```

- A. 06
- B. 07
- C. 08
- D. Error

2. Consider this query: How many rows will be returned?

```
SELECT *  
  
FROM employees  
  
WHERE salary BETWEEN 50000 AND 40000;
```

- A. All rows where salary between 50000 to 40000
- B. 0 rows
- C. NULL
- D. Error

3. What will be the output of the following SQL query?

```
SELECT COUNT(NULL), COUNT(1), COUNT(*)  
  
FROM (SELECT NULL UNION ALL SELECT 1 UNION ALL SELECT NULL) AS temp;
```

- A. 0,2,3
- B. 1,2,3
- C. 0,1,3
- D. 0,1,2

4. What will be the output of the following SQL query?

```
SELECT COALESCE(NULLIF (10, 10), NULLIF(20, 20), 30);
```

- A. NULL
- B. 10
- C. 20
- D. 30

5. What is the output of this query?

```
SELECT LENGTH(NULL);
```

- A. 1
- B. 4
- C. NULL.
- D. ERROR

6. What happens when GROUP BY is applied to a column containing NULLs?

```
SELECT department, COUNT(*)
```

```
FROM employees
```

```
GROUP BY department;
```

- A. NULL are ignored in the result
- B. NULLs are grouped separately as one group
- C. The query returns an error
- D. Only non-NULL departments are counted

7. From the following statements which of them is/are true about SQL

- 1. All attributes used in the group by clause must appear in the select clause
- 2. An SQL query can contain a having clause only if it has a group by clause
- 3. An SQL query can contain a having clause even if it does not have a group by clause
- 4. Not all attributes used in the group by clause need to appear in the select clause

- A. 3 and 4
- B. 1 and 2
- C. 1,3 and 4
- D. All of the above

8.What does the SQL keyword "CUBE" do in a GROUP BY clause?

- A) It computes aggregate values across multiple dimensions.
- B) It orders the result set in a specified sequence.
- C) It excludes NULL values from the result set.
- D) It groups the result set by the specified columns and their combinations

9.Which of the following statements is correct regarding BETWEEN in SQL?

- A. BETWEEN includes the boundary values
- B. BETWEEN excludes the boundary values
- C. BETWEEN only works for dates
- D. BETWEEN works only for numbers

- 1. A
- 2. A and B
- 3. A, B.and C
- 4. A, B, C and D

10. What's the result of this?

**SELECT DENSE_RANK() OVER (ORDER BY NULL)
FROM employees;**

- A. 1 for all rows
- B. Ranks as usual
- C. Error
- D. NULL for all rows