Top 10 SQL Interview Questions for freshers and Experience

WRITTEN BY- ADITYA KALYAN JHA

1. What is a Common Table Expression (CTE), and how does it differ from a subquery?

A Common Table Expression (CTE) is a temporary result set defined within the execution scope of a single SELECT, INSERT, UPDATE, or DELETE statement. It is created using the WITH keyword. Unlike subqueries, CTEs enhance readability and can be referenced multiple times within the main query, simplifying complex queries. Additionally, CTEs support recursive queries, which are challenging to implement with traditional subqueries.

2. How can you retrieve the nth highest salary from an employee table?

To obtain the nth highest salary, you can use the DENSE_RANK() function, which assigns ranks to salaries in descending order, allowing you to filter for the desired rank efficiently.

```
SELECT salary
FROM (
    SELECT salary, DENSE_RANK() OVER (ORDER
BY salary DESC) AS rank
   FROM employees
) AS ranked_salaries
WHERE rank = n;
```

3. Explain the difference between INNER JOIN and LEFT JOIN.

An INNER JOIN retrieves only the rows that have matching values in both tables involved in the join. In contrast, a LEFT JOIN (or LEFT OUTER JOIN) returns all rows from the left table and the matched rows from the right table. If there's no match, the result is NULL on the side of the right table.

4. What is indexing, and what are the different types of indexes in SQL?

Indexing is a technique used to enhance database performance by speeding up data retrieval processes. Indexes are special lookup tables that the database search engine can use to speed up data retrieval. The different types of indexes include:

Clustered Index: Sorts and stores the data rows of the table or view in order based on the index key. There can be only one clustered index per table.

Non-Clustered Index: Contains a sorted list of references to the table data. A table can have multiple non-clustered indexes.

Unique Index: Ensures that all the values in the index key are unique.

5. Provide an example of a window function in SQL.

Window functions perform calculations across a set of table rows that are somehow related to the current row. They are often used for running totals, ranking, and moving averages. Here's an example:

```
SELECT employee_id, salary,
       AVG(salary) OVER (PARTITION BY department_id) AS department_avg_salary
FROM employees;
```

This query calculates the average salary within each department without collapsing the result set, meaning each row retains its identity while displaying the department's average salary.

6. What are recursive CTEs, and when are they useful?

Recursive CTEs are Common Table Expressions that reference themselves, allowing the output of one iteration to be used as the input for the next. They are particularly useful for querying hierarchical data structures,

such as organizational charts or bill-of-materials structures, where data is naturally recursive.

7. How can you find the second highest salary in a table?

To find the second highest salary, you can use the DENSE_RANK() function to assign ranks to each salary and then filter for the second rank:

```
SELECT salary
FROM (
    SELECT salary, DENSE_RANK() OVER (ORDER
BY salary DESC) AS rank
   FROM employees
) AS ranked_salaries
WHERE rank  = 2;
```

This query assigns a rank to each salary in descending order and retrieves the salary with a rank of 2, which corresponds to the second highest salary.

8. What is the difference between DELETE and TRUNCATE commands?

Both DELETE and TRUNCATE are used to remove data from a table, but they operate differently:

DELETE:

Removes rows one at a time and logs each deletion. It can include a WHERE clause to specify which rows to delete and can be rolled back if within a transaction.

TRUNCATE:

Removes all rows from a table by deallocating the data pages. It is faster than DELETE because it doesn’t log individual row deletions. However, it cannot be rolled back in some database systems and doesn’t fire triggers.

9. What is normalization, and what are its different forms?

Normalization is the process of organizing data in a database to reduce redundancy and improve data integrity. The common normal forms include:

First Normal Form (1NF):

Ensures that each column contains atomic (indivisible) values and each record is unique.

Second Normal Form (2NF):

Meets all requirements of 1NF and ensures that all non-key attributes are fully functional dependent on the primary key.

Third Normal Form (3NF):

Meets all requirements of 2NF and ensures that all non-key attributes are not transitively dependent on the primary key.

WRITTEN BY - ADITYA KALYAN JHA

10. What is denormalization, and when is it used?

Denormalization is the process of intentionally introducing redundancy into a database by combining tables or adding redundant data to optimize read performance. While normalization aims to reduce redundancy and improve data integrity, denormalization is employed to enhance query performance, especially in read-heavy operations.

It’s commonly used in data warehousing and reporting systems where complex queries require data from multiple tables, and the overhead of joining these tables can be significant. By denormalizing, you can reduce the number of joins, thereby speeding up data retrieval. However, this comes at the cost of potential data anomalies and increased storage requirements.

Thank You for Reading

We hope this compilation of the Top 10 Questions and Answers has provided you with valuable insights and enhanced your understanding of the subject. Whether you're preparing for interviews, revising concepts, or expanding your knowledge, this guide is designed to support your learning journey.

If you found this resource helpful, consider sharing it with peers and colleagues who may benefit as well.

Stay curious, keep learning, and continue striving for excellence.

Best regards,
ADITYA KALYAN JHA
Evoforge