Top Flipkart SQL Interview Questions and Answers

Now lets check out some of the top SQL questions you may face in a Flipkart interview along with explanations and answers.

1.What is the difference between DELETE and TRUNCATE commands in SQL?

Answer:

- **DELETE:** The DELETE command removes rows from a table based on a condition. This DML operation, if necessary, can be rolled back. However, it doesn't free the occupied space and allows triggers to be fired.
- **TRUNCATE:** TRUNCATE is a DDL operation that removes all rows from a table without logging individual row deletions. It is faster than DELETE as it doesn't generate individual row delete triggers or logs. However, in most databases, the Delete command cannot be rolled back and resets any auto-increment counters.

2. Explain the concept of JOIN in SQL and its types.

Answer: JOIN combines rows from two or more tables based on a common column between them.. The main types of JOINs are:

- INNER JOIN: Returns records that have matching values in both tables.
- LEFT JOIN (or LEFT OUTER JOIN): Returns all records from the left table and the matching records from the right table. Unmatched records from the right table will be NULL.
- RIGHT JOIN (or RIGHT OUTER JOIN): Returns all records from the right table and the matching records from the left table. Unmatched records from the left table will be NULL.
- FULL JOIN (or FULL OUTER JOIN): It returns all records when there is a match in either the left or right table.

3. How to optimize a slow SQL query?

Answer: To optimize a slow SQL query you can follow these steps:

- **Leverage Indexing:** Optimize query performance with well-placed indexes on frequently searched columns.
- **Minimize Data Retrieval:** Select only essential data to reduce query overhead.
- **Analyze Query Execution:** Use EXPLAIN to identify performance bottlenecks and optimize your SQL.
- **Efficiently Join Tables:** Optimize JOIN operations by considering table order and types.
- **Replace Subqueries:** Improve query efficiency by converting subqueries to JOINs where possible.

- **Early Filtering:** Apply filters early in your queries to reduce the number of rows processed.

4. What is primary key and how is it different from unique key?

Answer:

- Primary Key: A primary key is a column (or set of columns) in a table that uniquely identifies each row in that table. It doesn't allow NULL values and must have unique values.
- Unique Key: A unique key also ensures all values in a column are unique but allows one NULL value per column. While there can be multiple unique keys in a table, there can be only one primary key.

[Join our online Data Science Course! Begin a high-paid career!](#)

5. How can you retrieve the second-highest salary from an employee table?

Answer: You can retrieve the second-highest salary using a subquery:

```
SELECT MAX(Salary)
FROM Employees WHERE Salary < (SELECT MAX(Salary) FROM Employees);
```

This query first finds the maximum salary in the Employees table, then the outer query retrieves the highest salary that is less than this maximum, effectively giving the second-highest salary.

6.What are window functions in SQL and how are they used?

Answer: Window functions perform calculations across a set of rows that are related to the current row. Unlike aggregate functions, window functions do not group rows into a single output row. Here are some common window functions:

- ROW_NUMBER(): Assigns a number to each row within a partition.
- RANK(): Gives a rank to each row within a partition, with gaps in the ranking for ties.
- DENSE_RANK(): Similar to RANK(), but no gaps in the ranking.
- OVER(): The window or set of rows to apply the window function.

For example, to assign row numbers based on salary:

```
SELECT Name, Salary, ROW_NUMBER() OVER (ORDER BY Salary DESC)

AS RowNum FROM Employees;
```

7. What are the ACID properties of a database?

Answer: ACID stands for Atomicity, Consistency, Isolation, and Durability. These are the properties that ensure transactions are reliable in a database:

- Atomicity: Atomicity requires the completion of all operations within a transaction. The transaction is rolled back if any operation fails, leaving the database unchanged.
- Consistency: A transaction ensures that the database transitions from one valid state to another, preserving database invariants.
- Isolation: Transactions operate independently, preventing interference from other transactions.
- Durability: Committed transactions are permanent, even in the event of system failures.

8. What is normalization? What are the normal forms?

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

- First Normal Form (1NF): All columns contain atomic values and the table has a primary key.
- Second Normal Form (2NF): The table is in 1NF and all non-key columns are fully dependent on the primary key.
- Third Normal Form (3NF): The table is in 2NF and all columns are non-transitively dependent on the primary key (i.e., no transitive dependencies).
- Boyce-Codd Normal Form (BCNF): A stricter version of 3NF where every determinant is a candidate key.

9. How do you handle NULLs in SQL?

Answer: NULLs can be handled using these SQL functions:

- IS NULL / IS NOT NULL: To check if a value is NULL or not.
- COALESCE(): Returns the first non-NULL value in the list of arguments.
- IFNULL()/NVL(): Replace NULL with a specified value.

For example, to replace NULL with 'Unknown':

```
SELECT Name, COALESCE(Department, 'Unknown') AS Department FROM
Employees;
```

14. What is a CASE statement, and how is it used in SQL?

Answer: A CASE statement is used to implement conditional logic in SQL. It allows you to return different values based on conditions. Here's an example:

```
SELECT employee_name, CASE
WHEN salary > 10000 THEN 'High'
WHEN salary > 5000 THEN 'Medium' ELSE 'Low'
END AS salary_category FROM employees;
```

[Join our online Data Science Course! Begin a high-paid career!](#)