1. What is the difference between WHERE and HAVING?

WHERE is used to filter rows before grouping or aggregation, while HAVING is used to filter groups after aggregation. Example: Use WHERE to filter records with sales > 1000, and HAVING to filter groups with SUM(sales) > 5000.

2. What are the different types of joins?

The main types of joins in SQL are: - INNER JOIN: Returns rows with matching values in both tables. - LEFT JOIN: Returns all rows from the left table and matching rows from the right. - RIGHT JOIN: Returns all rows from the right table and matching rows from the left. - FULL OUTER JOIN: Returns all rows when there is a match in one of the tables. - CROSS JOIN: Returns the Cartesian product of the two tables.

3. How do you calculate average revenue per user in SQL?

You can use the AVG or SUM/COUNT approach. Example: SELECT SUM(revenue)/COUNT(DISTINCT user_id) AS ARPU FROM sales;

4. What are subqueries?

A subquery is a query nested inside another SQL query. It can be used in SELECT, FROM, or WHERE clauses to return results for the main query. Example: SELECT name FROM customers WHERE id IN (SELECT customer_id FROM orders WHERE amount > 500);

5. How do you optimize a SQL query?

Ways to optimize SQL queries include: - Use proper indexing on frequently searched columns. - Avoid SELECT * and select only necessary columns. - Use JOINS instead of subqueries where possible. - Use EXISTS instead of IN for large datasets. - Analyze query execution plan to find bottlenecks.

6. What is a view in SQL?

A view is a virtual table based on the result of an SQL query. It stores a query definition, not actual data, and can simplify complex queries, provide security, and improve readability. Example: CREATE VIEW high_value_customers AS SELECT name, SUM(amount) FROM orders GROUP BY name HAVING SUM(amount) > 1000;

7. How would you handle null values in SQL?

You can handle NULL values using functions and conditions: - Use IS NULL / IS NOT NULL to check. - Use COALESCE(expr, replacement) to replace NULL with a default value. - Use IFNULL(expr, replacement) in MySQL. - In aggregates like COUNT, NULLs are ignored except with COUNT(*).