Top 50 SQL Interview Questions and Answers

1. What is SQL? How is it different from MySQL or PostgreSQL?

SQL is a language for managing databases. MySQL and PostgreSQL are RDBMSs that use SQL. MySQL is fast and common, PostgreSQL supports advanced features.

2. What are the different types of SQL statements?

DDL (CREATE, DROP), DML (INSERT, UPDATE, DELETE), DQL (SELECT), DCL (GRANT, REVOKE), TCL (COMMIT, ROLLBACK).

3. Explain the difference between WHERE and HAVING.

WHERE filters rows before aggregation. HAVING filters after aggregation.

4. What are PRIMARY KEY, FOREIGN KEY, UNIQUE, and CHECK constraints?

PRIMARY KEY: Unique, no NULLs. FOREIGN KEY: Reference another table. UNIQUE: All values must be different. CHECK: Enforces a condition.

5. What is the difference between DELETE, TRUNCATE, and DROP?

DELETE removes rows (can rollback). TRUNCATE removes all rows (cannot rollback). DROP removes the entire table.

6. What is normalization? Explain different normal forms.

Normalization reduces redundancy and improves data integrity. 1NF: Atomic values, 2NF: No partial dependencies, 3NF: No transitive dependencies.

7. What is denormalization and when is it useful?

Denormalization adds redundancy to improve read performance, useful in reporting or heavy read systems.

8. Explain the difference between CHAR and VARCHAR.

CHAR is fixed-length and padded with spaces. VARCHAR is variable-length, saving space.

Top 50 SQL Interview Questions and Answers

9. What are ACID properties in databases?

Atomicity, Consistency, Isolation, Durability - ensure reliable transactions.

10. What is the difference between INNER JOIN, LEFT JOIN, RIGHT JOIN, and FULL JOIN?

INNER JOIN: matching rows only. LEFT: all from left + matched right. RIGHT: all from right. FULL: all rows from both sides.

11. Write a query to find the second highest salary from an Employee table.

SELECT MAX(salary) FROM Employee WHERE salary < (SELECT MAX(salary) FROM Employee);

12. Write a query to get the department-wise average salary.

SELECT department, AVG(salary) FROM Employee GROUP BY department;

13. How would you retrieve duplicate records from a table?

SELECT column, COUNT(*) FROM table GROUP BY column HAVING COUNT(*) > 1;

14. How do you update a column with a calculation (e.g., 10% tax added)?

UPDATE Products SET price = price * 1.10;

15. How would you delete only duplicate rows from a table?

DELETE FROM Employee WHERE id NOT IN (SELECT MIN(id) FROM Employee GROUP BY name, salary);

16. Write a query to list customers who have placed more than 5 orders.

SELECT customer_id FROM Orders GROUP BY customer_id HAVING COUNT(*) > 5;

17. Write a query to join three or more tables.

SELECT * FROM A JOIN B ON A.id = B.a_id JOIN C ON B.id = C.b_id;

Top 50 SQL Interview Questions and Answers

18. What is a subquery? How is it different from a JOIN?

Subquery is nested, JOIN combines rows. JOINs are generally faster and more readable.

19. What is a correlated subquery? Give an example.

Subquery that depends on outer query. Example: WHERE salary > (SELECT AVG(salary) FROM Employee WHERE dept = e.dept);

20. How do you filter data based on a date range?

Use BETWEEN: SELECT * FROM Orders WHERE order_date BETWEEN '2025-01-01' AND '2025-06-30';

21. What are WINDOW FUNCTIONS? Name a few.

Functions like ROW_NUMBER(), RANK(), LAG(), LEAD() used over partitions of result sets.

22. What is the use of RANK(), DENSE_RANK(), and ROW_NUMBER()?

RANK: skips ranks on ties. DENSE_RANK: no skips. ROW_NUMBER: unique numbering.

23. What is a Common Table Expression (CTE)? How is it different from a subquery?

CTE is a temp result set with a name (WITH clause). Easier to read and supports recursion.

24. What are stored procedures? When should they be used?

Stored procedures are predefined SQL logic blocks, used for repeatable or complex logic.

25. What is a trigger? Give a real-world example.

Trigger is auto-executed code on insert/update/delete. Example: log changes to audit table.

26. What is a VIEW? What are its pros and cons?

VIEW is a virtual table. Pros: Simplifies query, improves security. Cons: Can't always be updated, slower than table.

Top 50 SQL Interview Questions and Answers

27. What are indexes? How do they improve performance?

Indexes speed up data retrieval but may slow inserts/updates. Use on frequently queried columns.

28. What is a materialized view?

A stored view with physical data. Can be refreshed and is faster for large aggregations.

29. What are transactions? Explain COMMIT, ROLLBACK, and SAVEPOINT.

Transaction is a group of operations. COMMIT saves, ROLLBACK undoes, SAVEPOINT allows partial rollback.

30. What are aggregate functions? List a few with examples.

SUM(), AVG(), COUNT(), MAX(), MIN(). Used to summarize data, e.g., AVG(salary).

31. How can you optimize a slow-running SQL query?

Use indexes, avoid SELECT *, check EXPLAIN plan, reduce subqueries and large joins.

32. What is the EXPLAIN or EXPLAIN PLAN statement used for?

It shows the query execution plan. Helps understand how SQL will be executed.

33. How does indexing affect INSERT, UPDATE, and DELETE performance?

Indexes improve read performance but slow down writes because index must be updated.

34. What is a composite index and when should it be used?

An index on multiple columns. Use when queries filter on those columns together.

35. What is normalization overhead and how do you deal with it?

Too many joins causing slow queries. Fix with denormalization or materialized views.

36. How do you avoid Cartesian products in JOINs?

Always use ON clause in JOINs to specify joining conditions.

37. What is partitioning in SQL?

Splitting large tables into smaller parts to improve performance. Types: range, list, hash.

38. What causes a deadlock in SQL, and how can you prevent it?

When two transactions block each other. Prevent with lock ordering, short transactions.

39. What is the difference between clustered and non-clustered indexes?

Clustered index defines physical order of rows. Non-clustered is a separate structure.

40. What tools do you use to monitor SQL query performance?

Tools: EXPLAIN, pg_stat_statements, MySQL profiler, APM tools like New Relic or DataDog.

41. You are asked to design a student-course grading system. What tables and relationships would

Tables: Students, Courses, Enrollments (with student_id, course_id, grade).

42. How would you store and retrieve attendance for employees in a scalable way?

Attendance(emp_id, date, status). Use indexes on emp_id, date. Optimize reads with partitioning.

43. In a library system, how would you track overdue books and fines using SQL?

Use borrow_date, due_date, return_date. Fine = DATEDIFF(return_date, due_date) * daily_rate.

44. What would you do if a production database is missing some records due to a failed update?

Check logs, restore from backups, and reapply changes using staging environment.

45. How would you implement role-based access to sensitive information in SQL?

Create roles, grant/revoke permissions. Use views to restrict columns.

Top 50 SQL Interview Questions and Answers

46. You are given a raw CSV with dirty data. How would you load and clean it using SQL?

LOAD DATA INFILE, then clean with TRIM, REPLACE, ISNULL checks.

47. How would you calculate monthly retention from a user login dataset?

Compare users logged in this month vs previous. Use JOINs or CTEs to track cohorts.

48. What measures would you take to secure a database with sensitive data?

Encrypt data, use access control, audit logs, and backup strategies.

49. How do you create daily backup and restore plans for a SQL database?

Use cron jobs with mysqldump/pg_dump, store backups off-site, document restore process.

50. What are the best practices you follow while writing SQL in production?

Avoid SELECT *, use indexing, comment complex queries, test performance, use version control.