

# SQL Interview Questions

answer all these interview question

### \*\*A. Basic SQL (1–20)\*\*

1. What is SQL?
2. What are the different types of SQL commands?
3. What is a primary key?
4. What is a foreign key?
5. What is the difference between WHERE and HAVING clauses?
6. What is the difference between UNION and UNION ALL?
7. How do you retrieve unique values from a column?
8. What are aggregate functions in SQL?
9. Explain the difference between CHAR and VARCHAR.
10. What is a NULL value in SQL?
11. How do you filter NULL values?
12. What does the DISTINCT keyword do?
13. How do you rename a column or table?
14. What is the ORDER BY clause used for?
15. What is the difference between DELETE, TRUNCATE, and DROP?
16. How can you change a column datatype?
17. How do you write a simple SELECT query?
18. How to retrieve records from two tables using JOIN?
19. What are constraints in SQL?
20. Explain the IN and BETWEEN operators.

Here are **clear, interview-ready answers** to all **Basic SQL (1–20)** questions. I'll keep them **simple, direct, and honest**, exactly how interviewers expect.

---

## 1. What is SQL?

SQL (Structured Query Language) is a standard language used to store, retrieve, update, and manage data in a relational database. It allows us to communicate with databases like MySQL, PostgreSQL, Oracle, or SQL Server to perform operations such as querying data, inserting records, updating values, and deleting data.

---

## 2. What are the different types of SQL commands?

SQL commands are mainly divided into five types:

DDL (Data Definition Language) is used to define database structures like tables using commands such as CREATE, ALTER, and DROP.

DML (Data Manipulation Language) is used to insert, update, and delete data using INSERT, UPDATE, and DELETE.

DQL (Data Query Language) is used to retrieve data using SELECT.

DCL (Data Control Language) is used to control access using GRANT and REVOKE.

TCL (Transaction Control Language) is used to manage transactions using COMMIT, ROLLBACK, and SAVEPOINT.

---

### 3. What is a primary key?

A primary key is a column or a set of columns that uniquely identifies each row in a table. It cannot contain NULL values and must always be unique. Each table can have only one primary key, which helps maintain data integrity.

---

### 4. What is a foreign key?

A foreign key is a column in one table that refers to the primary key of another table. It is used to establish a relationship between two tables and ensures referential integrity, meaning the value must exist in the referenced table.

---

### 5. What is the difference between WHERE and HAVING clauses?

The WHERE clause is used to filter rows before grouping happens, while the HAVING clause is used to filter groups after aggregation. WHERE cannot be used with aggregate functions, but HAVING can be used with aggregate functions like COUNT or SUM.

---

### 6. What is the difference between UNION and UNION ALL?

UNION combines results from multiple SELECT queries and removes duplicate records. UNION ALL also combines results but keeps duplicate records. UNION ALL is faster because it does not perform duplicate checking.

---

### 7. How do you retrieve unique values from a column?

Unique values are retrieved using the DISTINCT keyword. It removes duplicate values from the result set and returns only unique records.

---

### 8. What are aggregate functions in SQL?

Aggregate functions perform calculations on multiple rows and return a single value. Common aggregate functions include COUNT, SUM, AVG, MIN, and MAX. They are often used with GROUP BY.

---

### 9. Explain the difference between CHAR and VARCHAR.

CHAR is a fixed-length data type, meaning it always occupies the same space even if the value is shorter. VARCHAR is a variable-length data type and only uses the required space. VARCHAR is more memory-efficient than CHAR.

---

### 10. What is a NULL value in SQL?

A NULL value represents missing, unknown, or undefined data. It is not equal to zero or an empty string. NULL simply means no value is stored in that column.

---

## 11. How do you filter NULL values?

NULL values are filtered using IS NULL or IS NOT NULL because NULL cannot be compared using standard operators like = or !=.

---

## 12. What does the DISTINCT keyword do?

The DISTINCT keyword removes duplicate rows from the result set and ensures that only unique records are returned.

---

## 13. How do you rename a column or table?

A column is renamed using the AS keyword in a query or ALTER TABLE for permanent change. A table is renamed using the ALTER TABLE RENAME command.

---

## 14. What is the ORDER BY clause used for?

ORDER BY is used to sort the result set in ascending or descending order. By default, it sorts in ascending order, but DESC can be used for descending order.

---

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

DELETE removes specific rows and can be rolled back.

TRUNCATE removes all rows from a table quickly and cannot be rolled back.

DROP deletes the entire table structure along with the data permanently.

---

## 16. How can you change a column datatype?

A column datatype is changed using the ALTER TABLE command along with MODIFY or ALTER COLUMN, depending on the database.

---

## 17. How do you write a simple SELECT query?

A simple SELECT query retrieves data from a table by specifying the columns and the table name using the SELECT and FROM keywords.

---

## 18. How to retrieve records from two tables using JOIN?

Records from two tables are retrieved using JOIN by specifying a common column between them. The most common join is INNER JOIN, which returns matching records from both tables.

---

## 19. What are constraints in SQL?

Constraints are rules applied to table columns to enforce data integrity. Common constraints include NOT NULL, UNIQUE, PRIMARY KEY, FOREIGN KEY, CHECK, and DEFAULT.

## 20. Explain the IN and BETWEEN operators.

The IN operator is used to match a value against a list of values.

The BETWEEN operator is used to filter values within a specified range, including the start and end values.

If you want, I can **convert these into 1-2 line crisp answers, mock interview answers, or advanced SQL next (joins, indexes, normalization, subqueries).**

### \*\*Intermediate SQL (21-50)\*\*

1. What is a JOIN? List different types of JOINS.
2. What is a self JOIN?
3. Explain the difference between INNER JOIN and OUTER JOIN.
4. What is a subquery?
5. What is a correlated subquery?
6. How do you update multiple columns in SQL?
7. What is the GROUP BY clause used for?
8. How does SQL handle NULLs in GROUP BY and ORDER BY?
9. What is a CASE statement? Provide an example.
10. How can you fetch the first N rows from a table?
11. How do you find duplicate records in a table?
12. How to delete duplicate rows from a table?
13. What is normalization?
14. What are the normal forms in database design?
15. What is denormalization?
16. What is a view in SQL?
17. How to update a view in SQL?
18. What is the difference between a view and a table?
19. What are indexes in SQL?
20. How do indexes affect performance?
21. What is a clustered index?
22. What is a non-clustered index?
23. What is the difference between a unique constraint and a primary key?
24. What is the use of the COALESCE function?
25. What is the difference between RANK, DENSE\_RANK, and ROW\_NUMBER?
26. What is a CTE (Common Table Expression)?
27. How do you use recursive CTEs?
28. What is a window function?
29. How does the OVER clause work in SQL?
30. What are transactions in SQL?

Below are **clean, interview-ready, paragraph-style answers** for **Intermediate SQL (21–50)**. These are framed exactly the way you'd **speak in an interview**, not textbook fluff.

---

## **21. What is a JOIN? List different types of JOINS.**

A JOIN is used to combine rows from two or more tables based on a related column between them. The main types of joins are INNER JOIN, which returns matching records from both tables; LEFT JOIN, which returns all records from the left table and matching ones from the right; RIGHT JOIN, which does the opposite; and FULL OUTER JOIN, which returns records from both tables whether they match or not.

---

## **22. What is a self JOIN?**

A self JOIN is a join where a table is joined with itself. It is useful when comparing rows within the same table, such as finding employees and their managers when both are stored in the same table.

---

## **23. Explain the difference between INNER JOIN and OUTER JOIN.**

INNER JOIN returns only the rows that have matching values in both tables. OUTER JOIN returns matched rows as well as unmatched rows from one or both tables, depending on whether it is a LEFT, RIGHT, or FULL OUTER JOIN.

---

## **24. What is a subquery?**

A subquery is a query written inside another query. It is used to retrieve data that will be used by the main query, often inside WHERE, FROM, or SELECT clauses.

---

## **25. What is a correlated subquery?**

A correlated subquery is a subquery that depends on the outer query for its values. It is executed once for each row processed by the outer query, making it slower than a normal subquery in many cases.

---

## **26. How do you update multiple columns in SQL?**

Multiple columns can be updated in a single UPDATE statement by specifying each column with its new value in the SET clause, separated by commas.

---

## **27. What is the GROUP BY clause used for?**

GROUP BY is used to group rows that have the same values in specified columns. It is commonly used with aggregate functions like COUNT, SUM, and AVG to perform calculations on each group.

---

## **28. How does SQL handle NULLs in GROUP BY and ORDER BY?**

In GROUP BY, all NULL values are treated as a single group. In ORDER BY, NULL values are usually sorted either first or last depending on the database system and sorting order.

---

## 29. What is a CASE statement? Provide an example.

A CASE statement is used to apply conditional logic in SQL, similar to if-else statements in programming. It allows different results based on different conditions.

Example:

```
sql

SELECT name,
CASE
  WHEN marks >= 60 THEN 'Pass'
  ELSE 'Fail'
END AS result
FROM students;
```

---

## 30. How can you fetch the first N rows from a table?

The first N rows can be fetched using LIMIT in MySQL and PostgreSQL, TOP in SQL Server, or FETCH FIRST in Oracle. The exact syntax depends on the database.

---

## 31. How do you find duplicate records in a table?

Duplicate records are found by grouping rows using GROUP BY and filtering groups having a count greater than one using the HAVING clause.

---

## 32. How to delete duplicate rows from a table?

Duplicate rows can be deleted using a subquery or a CTE while keeping one unique record, often by using ROW\_NUMBER to identify and remove extra duplicates.

---

## 33. What is normalization?

Normalization is the process of organizing database tables to reduce data redundancy and improve data integrity. It involves dividing large tables into smaller ones and defining relationships between them.

---

## 34. What are the normal forms in database design?

The common normal forms are First Normal Form (1NF), Second Normal Form (2NF), Third Normal Form (3NF), and Boyce-Codd Normal Form (BCNF). Each normal form addresses different types of data redundancy and dependency issues.

---

## 35. What is denormalization?

Denormalization is the process of intentionally adding redundancy to a database to improve read performance. It is often used in reporting systems where fast data retrieval is more important than storage efficiency.

---

### **36. What is a view in SQL?**

A view is a virtual table created using a SELECT query. It does not store data itself but displays data from one or more tables, helping simplify complex queries and improve security.

---

### **37. How to update a view in SQL?**

A view can be updated if it is based on a single table and does not contain complex clauses like JOIN, GROUP BY, or DISTINCT. Updates on the view are reflected in the underlying table.

---

### **38. What is the difference between a view and a table?**

A table stores data physically in the database, while a view is just a saved query that displays data. Views do not store data themselves, except in the case of materialized views.

---

### **39. What are indexes in SQL?**

Indexes are database objects used to speed up data retrieval operations. They work like an index in a book, allowing the database to find data quickly without scanning the entire table.

---

### **40. How do indexes affect performance?**

Indexes improve SELECT query performance but slow down INSERT, UPDATE, and DELETE operations because the index must also be updated. Too many indexes can negatively affect performance.

---

### **41. What is a clustered index?**

A clustered index defines the physical order of data in a table. A table can have only one clustered index because data can be stored in only one order.

---

### **42. What is a non-clustered index?**

A non-clustered index creates a separate structure that stores pointers to the actual data rows. A table can have multiple non-clustered indexes.

---

### **43. What is the difference between a unique constraint and a primary key?**

A primary key uniquely identifies records and does not allow NULL values. A unique constraint also enforces uniqueness but allows one NULL value and a table can have multiple unique constraints.

---

#### 44. What is the use of the COALESCE function?

COALESCE is used to return the first non-NULL value from a list of expressions. It is commonly used to handle NULL values in queries.

---

#### 45. What is the difference between RANK, DENSE\_RANK, and ROW\_NUMBER?

ROW\_NUMBER assigns a unique sequential number to rows. RANK gives the same rank to duplicate values but skips the next rank. DENSE\_RANK also gives the same rank to duplicates but does not skip numbers.

---

#### 46. What is a CTE (Common Table Expression)?

A CTE is a temporary result set defined using the WITH keyword. It improves query readability and is often used instead of subqueries for complex logic.

---

#### 47. How do you use recursive CTEs?

Recursive CTEs are used to query hierarchical data such as parent-child relationships. They consist of a base query and a recursive query that references the CTE itself.

---

#### 48. What is a window function?

A window function performs calculations across a set of rows related to the current row without collapsing the result into a single row. Examples include RANK, SUM, and AVG used with OVER.

---

#### 49. How does the OVER clause work in SQL?

The OVER clause defines the window of rows for a window function. It can specify partitioning and ordering of rows to control how calculations are applied.

---

#### 50. What are transactions in SQL?

A transaction is a sequence of SQL operations executed as a single unit of work. Transactions follow ACID properties to ensure data consistency, and they are controlled using COMMIT and ROLLBACK.

---

If you want, next I can give you **Advanced SQL (51–80)**, **real interview SQL queries**, or **one-line rapid-fire answers** for last-minute revision.