1. SQL Basics (Must Know)

Q1. What is SQL?

SQL (Structured Query Language) is used to store, manipulate and retrieve data from relational databases.

Q2. DDL vs DML vs DCL vs TCL

- **DDL:** CREATE, ALTER, DROP

- **DML:** SELECT, INSERT, UPDATE, DELETE

- **DCL:** GRANT, REVOKE

- **TCL:** COMMIT, ROLLBACK, SAVEPOINT

---

2. Keys & Constraints

Primary Key

- Unique + Not Null

- One per table

Foreign Key

- Links two tables

-

Ensures referential integrity

Unique Key

- Unique values allowed

- Can allow NULL (DB dependent)

---

3. Joins (Top Interview Topic)

Join	Meaning
INNER JOIN	Matching rows only
LEFT JOIN	All left + matching right
RIGHT JOIN	All right + matching left
FULL OUTER JOIN	All rows from both
CROSS JOIN	Cartesian product

Example

```
SELECT e.name, d.dept_name FROM employee e INNER JOIN department d ON e.dept_id = d.dept_id;
```

---

4. WHERE vs HAVING

- WHERE filters rows before grouping

- HAVING filters groups after GROUP BY

```
SELECT dept_id, COUNT(*) FROM employee GROUP BY dept_id HAVING COUNT(*) > 5;
```

---

5. GROUP BY + Aggregations

Common aggregates:

- COUNT()

- SUM()

- AVG()

- MIN()

- MAX()

---

6. Window Functions (Very Important)

Q: What are window functions?

They perform calculations across a set of rows related to current row.

Common window functions:

- ROW_NUMBER()

- RANK()

- DENSE_RANK()

- LEAD()

- LAG()

Example: Top salary per department

```
SELECT * FROM ( SELECT emp_id, dept_id, salary, RANK() OVER(PARTITION BY dept_id ORDER BY salary DESC) AS rnk FROM employee ) t WHERE rnk = 1;
```

---

7. Normalization (Theory)

- **1NF:** atomic values

- **2NF:** no partial dependency

- **3NF:** no transitive dependency

Goal: reduce redundancy and anomalies.

---

8. Indexing

Q: What is an index?

An index improves SELECT query performance using a data structure like B-Tree.

Pros:

- Faster reads

Cons:

- Slower inserts/updates

- Uses extra memory

---

9. Common Query Patterns

Find duplicates

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

Second highest salary

```
SELECT MAX(salary) FROM employee WHERE salary < (SELECT MAX(salary) FROM employee);
```

Delete duplicates

```
DELETE FROM employee WHERE id NOT IN ( SELECT MIN(id) FROM employee GROUP BY email );
```

---

10. Rapid Fire SQL Interview Questions

- Difference between DELETE, TRUNCATE, DROP

- What is ACID?

- What is a view?

- What is a stored procedure?

- What is a trigger?

-

Clustered vs Non-clustered index

-

UNION vs UNION ALL