SQL Complete Command Reference (Interview-Ready)

This document provides a **structured, detailed, and interview-oriented** breakdown of SQL commands, organized by **SQL language types** (DDL, DML, DQL, DCL, TCL) and by **skill level** (Basic → Intermediate → Advanced).

---

1. SQL OVERVIEW

SQL (Structured Query Language) is a standard language used to interact with **Relational Database Management Systems (RDBMS)** such as MySQL, PostgreSQL, SQL Server, and Oracle.

SQL is broadly divided into **five categories** based on functionality: - **DDL** – Data Definition Language - **DML** – Data Manipulation Language - **DQL** – Data Query Language - **DCL** – Data Control Language - **TCL** – Transaction Control Language

---

2. DATA DEFINITION LANGUAGE (DDL)

Purpose: Used to define, modify, and delete database structures (schema-level operations).

2.1 CREATE

Used to create database objects.

Create Database

```
CREATE DATABASE company_db;
```

Create Table

```
CREATE TABLE employees (
  emp_id INT PRIMARY KEY,
  name VARCHAR(50) NOT NULL,
  salary DECIMAL(10,2),
  department VARCHAR(30)
);
```

2.2 ALTER

Used to modify existing table structure.

Add Column

```
ALTER TABLE employees ADD age INT;
```

Modify Column

```
ALTER TABLE employees MODIFY salary DECIMAL(12,2);
```

Drop Column

```
ALTER TABLE employees DROP age;
```

2.3 DROP

Deletes database objects permanently.

```
DROP TABLE employees;
DROP DATABASE company_db;
```

⚠️**Interview Tip:** DROP removes both structure and data.

2.4 TRUNCATE

Removes all records but keeps table structure.

```
TRUNCATE TABLE employees;
```

Difference: - DELETE → row-level, rollback possible - TRUNCATE → table-level, rollback not possible

---

3. DATA MANIPULATION LANGUAGE (DML)

Purpose: Used to manipulate data inside tables.

3.1 INSERT

```sql
INSERT INTO employees VALUES (1, 'Amit', 60000, 'IT');
```

Insert specific columns:

```sql
INSERT INTO employees (emp_id, name) VALUES (2, 'Neha');
```

3.2 UPDATE

```sql
UPDATE employees
SET salary = 70000
WHERE emp_id = 1;
```

⚠️ Always use WHERE with UPDATE.

3.3 DELETE

```sql
DELETE FROM employees WHERE emp_id = 2;
```

Deletes selected rows only.

---

4. DATA QUERY LANGUAGE (DQL)

Purpose: Used to retrieve data from the database.

4.1 SELECT

```sql
SELECT * FROM employees;
SELECT name, salary FROM employees;
```

4.2 WHERE

Filters rows based on conditions.

```
SELECT * FROM employees WHERE salary > 50000;
```

Operators: - Comparison: `=, !=, >, <, >=, <=` - Logical: `AND, OR, NOT`

4.3 DISTINCT

Removes duplicate values.

```
SELECT DISTINCT department FROM employees;
```

Rule: DISTINCT comes immediately after SELECT.

4.4 ORDER BY

Sorts result set.

```
SELECT * FROM employees ORDER BY salary DESC;
```

4.5 LIMIT / OFFSET

```
SELECT * FROM employees ORDER BY salary DESC LIMIT 5 OFFSET 2;
```

4.6 AGGREGATE FUNCTIONS

```
SELECT COUNT(*) FROM employees;
SELECT AVG(salary) FROM employees;
SELECT MAX(salary) FROM employees;
```

4.7 GROUP BY

Used with aggregates.

```
SELECT department, AVG(salary)
FROM employees
GROUP BY department;
```

4.8 HAVING

Filters grouped data.

```
SELECT department, AVG(salary)
FROM employees
GROUP BY department
HAVING AVG(salary) > 60000;
```

5. DATA CONTROL LANGUAGE (DCL)

Purpose: Controls user access and permissions.

5.1 GRANT

```
GRANT SELECT, INSERT ON employees TO user1;
```

5.2 REVOKE

```
REVOKE INSERT ON employees FROM user1;
```

6. TRANSACTION CONTROL LANGUAGE (TCL)

Purpose: Manages transactions.

6.1 COMMIT

Saves changes permanently.

```
COMMIT;
```

6.2 ROLLBACK

Undo changes.

```
ROLLBACK;
```

6.3 SAVEPOINT

```
SAVEPOINT sp1;
ROLLBACK TO sp1;
```

7. SET OPERATORS

```
SELECT city FROM customers
UNION
SELECT city FROM suppliers;
```

- UNION → removes duplicates
- UNION ALL → keeps duplicates
- INTERSECT / EXCEPT → DB-specific

8. KEY INTERVIEW SUMMARY

- **DDL** → Structure
- **DML** → Data manipulation
- **DQL** → Data retrieval
- **DCL** → Security
- **TCL** → Transactions

This foundation is mandatory before moving to joins, subqueries, window functions, and performance tuning.

✅This document can be used directly for **revision, interviews, and real-world SQL work**.