

# Day 3 — Complete SQL Study Pack (Theory + Practice + Quiz + 50 Questions)

## PART 1 — Day 3 Theory

### 1. Types of Joins

- INNER JOIN – returns matching rows only
- LEFT JOIN – returns all left rows + matched right rows
- RIGHT JOIN – all right rows + matched left rows
- FULL JOIN – returns all rows from both tables
- CROSS JOIN – returns all combinations (Cartesian product)

### 2. Subqueries

- Single-level subquery
- Multi-level subquery
- Correlated subquery – depends on outer query

### 3. CTE (WITH Clause)

Used to simplify complex SQL queries.

### 4. Normalization

- 1NF – atomic values
- 2NF – partial dependency removed
- 3NF – transitive dependency removed

### 5. Keys

- Primary Key – unique identifier
- Foreign Key – reference to another table
- Composite Key – multiple columns combined
- Unique Key – unique but allows NULLs

### 6. Set Operators

- UNION – removes duplicates
- UNION ALL – keeps duplicates
- INTERSECT – common rows
- EXCEPT – rows in first query not in second

## PART 2 — Day 3 Practice (With Answers)

### Q1. Find customers who didn't place any orders.

```
SELECT c.* FROM customers c LEFT JOIN orders o  
ON c.customer_id = o.customer_id WHERE o.customer_id IS NULL;
```

### Q2. Total revenue per product.

```
SELECT product_id, SUM(amount) FROM orders GROUP BY product_id;
```

### Q3. Employees earning above average salary.

```
SELECT * FROM employees WHERE salary > (SELECT AVG(salary) FROM employees);
```

### Q4. Third-highest salary.

```
SELECT salary FROM (  
    SELECT salary, ROW_NUMBER() OVER(ORDER BY salary DESC) rn FROM employees  
) t WHERE rn = 3;
```

### **Q5. Count orders per customer using CTE.**

```
WITH c AS (SELECT customer_id, COUNT(*) AS total FROM orders GROUP BY customer_id)
SELECT * FROM c;
```

## **PART 3 — Day 3 Quiz (With Answers)**

### **Q1. What is a CROSS JOIN?**

**Answer:** Produces combinations of all rows.

### **Q2. What is 3NF?**

**Answer:** No non-key attribute depends on another non-key attribute.

### **Q3. UNION vs UNION ALL?**

**Answer:** UNION removes duplicates; UNION ALL keeps all rows.

### **Q4. Find duplicate emails.**

```
SELECT email FROM users GROUP BY email HAVING COUNT(*) > 1;
```

### **Q5. Composite key?**

Primary key made of multiple columns.

### **Q6. What is a CTE?**

Temporary result set defined using WITH.

### **Q7. Find unmatched rows.**

LEFT JOIN + WHERE right\_table.id IS NULL.

### **Q8. What is INTERSECT?**

Returns common rows.

### **Q9. Max salary per department.**

```
SELECT dept, MAX(salary) FROM employees GROUP BY dept;
```

### **Q10. Scalar subquery?**

Subquery returning exactly one value.

## **PART 4 — 50 SQL Interview Questions (With Answers)**

1. What is SQL?
2. Difference between DELETE and TRUNCATE.
3. What is ACID property?
4. What is normalization?
5. 1NF vs 2NF vs 3NF.
6. Foreign key definition.
7. INNER JOIN vs LEFT JOIN.
8. What is UNION?
9. What is a VIEW?
10. What is an INDEX?
11. Clustered vs Non-Clustered index.
12. What is a transaction?
13. COMMIT vs ROLLBACK.
14. GROUP BY vs ORDER BY.
15. WHERE vs HAVING.

16. DISTINCT usage.
17. COUNT(\*) vs COUNT(column).
18. Subquery types.
19. Correlated subquery example.
20. CTE usage.
21. Window functions.
22. ROW\_NUMBER vs RANK.
23. DENSE\_RANK usage.
24. Stored procedure definition.
25. Triggers usage.
26. JOIN types.
27. CROSS JOIN definition.
28. SELF JOIN usage.
29. Schema definition.
30. Primary key characteristics.
31. Unique key vs Primary key.
32. Composite key usage.
33. Set operators.
34. INTERSECT example.
35. EXCEPT example.
36. Index advantages.
37. Index disadvantages.
38. What is NULL?
39. Handling NULL in SQL.
40. COALESCE usage.
41. NVL usage (Oracle).
42. LIKE operator.
43. Wildcards in SQL.
44. BETWEEN operator.
45. IN vs NOT IN.
46. EXISTS vs IN.
47. DELETE vs DROP.
48. UPSERT (MERGE).
49. Temporary tables.
50. Best way to find Nth highest salary.