21 SQL Prep - Question Bank

1. Introduction to DBMS and Relational Databases

1. What is a Database Management System (DBMS)?
2. How is a DBMS different from a traditional file system?
3. Define a relational database.
4. What are tables, rows, and columns in a database?
5. Explain the concept of a primary key.
6. What is a foreign key and why is it important?
7. Define a candidate key and alternate key.
8. What is a composite key?
9. What is the difference between schema and instance in DBMS?
10. Explain data independence and its types.

---

2. DDL – Data Definition Language

11. What is DDL in SQL?
12. Name some DDL commands and their uses.
13. What happens internally when you execute a CREATE TABLE command?
14. Difference between TRUNCATE and DELETE commands.
15. How does ALTER TABLE differ from UPDATE?
16. What is the difference between DROP and TRUNCATE?
17. What are constraints in SQL?
18. Explain the NOT NULL and UNIQUE constraints.
19. How does CHECK constraint work?
20. What is DEFAULT constraint used for?

---

3. DML – Data Manipulation Language

21. What are the main DML commands?
22. How is UPDATE different from INSERT?
23. Can we use WHERE clause in INSERT? Why or why not?
24. How does DELETE differ from TRUNCATE in terms of rollback?
25. What is the purpose of MERGE statement?
26. How can you copy data from one table to another?
27. What are pseudo-columns like ROWID and ROWNUM?
28. Explain how you would update multiple columns in one query.
29. How can you delete duplicate rows from a table?

30. What is the significance of COMMIT and ROLLBACK with DML?

---

4. DQL – Data Query Language

31. What does the SELECT statement do?
32. What is the difference between SELECT * and SELECT column_list?
33. What is a subquery? Give an example.
34. What is the use of DISTINCT keyword?
35. How can you rename a column in a query result?
36. What is the difference between WHERE and HAVING clause?
37. Explain ORDER BY and its default sorting behavior.
38. What are aggregate functions in SQL?
39. Explain the difference between COUNT(*) and COUNT(column_name).
40. What are scalar functions? Give examples.

---

5. Grouping, Filtering, and Conditional Logic

41. What does GROUP BY do in SQL?
42. Can we use GROUP BY without aggregate functions?
43. What is the purpose of HAVING clause?
44. Difference between WHERE and HAVING in SQL execution order.
45. Explain the use of CASE statement in SQL.
46. How can you replace NULL values in a result set?
47. What are COALESCE() and NULLIF()?
48. Explain how filtering works with AND/OR/NOT operators.
49. What is the difference between IN, ANY, and ALL?
50. How can we perform conditional aggregation in SQL?

---

6. Table Relationships and Joins

51. What are the types of relationships in a database (1:1, 1:N, M:N)?
52. Explain how foreign keys enforce referential integrity.
53. What are INNER JOIN, LEFT JOIN, RIGHT JOIN, and FULL JOIN?
54. What is a CROSS JOIN?
55. How can we join more than two tables?
56. What is a SELF JOIN and when do you use it?
57. Difference between INNER JOIN and WHERE-based join.
58. What is an equi join vs non-equi join?
59. How can you find unmatched records between two tables?
60. What are performance implications of using multiple joins?

7. Normalization and Database Design

61. What is normalization? Why is it important?
62. Explain 1NF, 2NF, and 3NF with examples.
63. What are partial and transitive dependencies?
64. What is denormalization and when would you use it?
65. What are anomalies in database design?
66. How do surrogate keys differ from natural keys?
67. What is a schema diagram?
68. Explain referential integrity with an example.
69. How does normalization affect query performance?
70. What is a domain in database design?

8. Transactions and TCL

71. What are transactions in SQL?
72. Define the ACID properties.
73. Difference between COMMIT and ROLLBACK.
74. What is the SAVEPOINT command used for?
75. Explain what happens when a transaction fails mid-way.
76. How do you handle concurrent transactions in SQL?
77. What are dirty reads and how can they be avoided?
78. What are isolation levels in SQL?
79. How can you ensure atomicity in SQL operations?
80. What happens if you forget to COMMIT after DML operations?