DBMS + SQL Interview Bible (30 Q&A)

DBMS Theory Questions

Q1. What is DBMS? Explain advantages.

A DBMS (Database Management System) is software that manages storage, retrieval, and manipulation of data.

Advantages:

- Reduces data redundancy
- Ensures data consistency
- Provides security
- Allows concurrency control
- Ensures backup and recovery

Q2. Difference between DBMS and RDBMS.

DBMS: Stores data as files, no relations. Examples: XML DB.

RDBMS: Stores data in tables with relations. Uses SQL. Examples: MySQL, PostgreSQL.

Q3. Explain keys in DBMS.

- Primary Key: Unique identifier
- Candidate Key: All possible unique keys
- Alternate Key: Candidate keys not chosen as PK
- Foreign Key: References PK of another table
- Composite Key: Combination of attributes
- Super Key: Any set uniquely identifying tuples

Q4. What is Normalization?

Normalization is the process of organizing data to reduce redundancy.

1NF: Atomic values

2NF: No partial dependency

3NF: No transitive dependency

BCNF: Every determinant must be a candidate key

Q5. What is Denormalization?

Denormalization = introducing redundancy for faster queries. Used in OLAP systems.

Q6. What is a Transaction? Explain ACID properties.

Transaction = logical unit of work.

ACID: Atomicity, Consistency, Isolation, Durability

Q7. Explain Concurrency problems.

Problems: Dirty Reads, Non-repeatable Reads, Phantom Reads.

Q8. What are Isolation Levels?

Read Uncommitted, Read Committed, Repeatable Read, Serializable

Q9. What is Deadlock? How to handle?

Deadlock = circular wait for resources. Solutions: Prevention, Detection, Avoidance

Q10. Difference between OLTP and OLAP.

OLTP = short transactions (banking). OLAP = analytical queries (data warehouse).

SQL Query Questions

Q16. Find 2nd highest salary.

```
SELECT MAX(Salary) FROM Employee WHERE Salary < (SELECT MAX(Salary) FROM Employee);
```

Q17. Find Nth highest salary.

```
SELECT Salary FROM (SELECT Salary, DENSE_RANK() OVER (ORDER BY Salary DESC) AS rnk FROM Employee)
t WHERE rnk = N;
```

Q18. Find duplicate records.

```
SELECT Name, COUNT(*) FROM Student GROUP BY Name HAVING COUNT(*) > 1;
```

Q19. Employees who earn more than their manager.

```
SELECT e.Name FROM Employee e JOIN Employee m ON e.ManagerID = m.EmpID WHERE e.Salary > m.Salary;
```

Q20. Department with maximum employees.

```
SELECT Dept FROM Employee GROUP BY Dept ORDER BY COUNT(*) DESC LIMIT 1;
```

Q21. Employees with same salary.

```
SELECT Salary, COUNT(*) FROM Employee GROUP BY Salary HAVING COUNT(*) > 1;
```

Q22. Top 3 salaries.

```
SELECT DISTINCT Salary FROM Employee ORDER BY Salary DESC LIMIT 3;
```

Q23. Employees hired in last 30 days.

```
SELECT * FROM Employee WHERE HireDate >= CURDATE() - INTERVAL 30 DAY;
```

Q24. Average salary by department.

```
SELECT Dept, AVG(Salary) FROM Employee GROUP BY Dept;
```

Q25. Highest salary in each department.

```
SELECT Dept, MAX(Salary) FROM Employee GROUP BY Dept;
```

Q26. Employees with no manager.

```
SELECT Name FROM Employee WHERE ManagerID IS NULL;
```

Q27. Count students per department > 5.

```
SELECT Dept, COUNT(*) FROM Student GROUP BY Dept HAVING COUNT(*) > 5;
```

Q28. Employees whose name starts with A.

```
SELECT * FROM Employee WHERE Name LIKE 'A%';
```

Q29. Delete duplicate rows but keep one.

```
DELETE FROM Student WHERE RowID NOT IN (SELECT MIN(RowID) FROM Student GROUP BY Name);
```

Q30. Employees with highest salary.

```
SELECT * FROM Employee WHERE Salary = (SELECT MAX(Salary) FROM Employee);
```