# 📘 Complete DBMS Interview Preparation Checklist

---

## 🔷 1. Introduction to DBMS

- What is DBMS? Advantages over File Systems

- Characteristics of DBMS

- Users of DBMS (DBA, End Users, etc.)

- Instances vs Schemas

- Data Abstraction (Physical, Logical, View Levels)

- Data Independence (Logical vs Physical)

- 3-Tier Architecture (Internal, Conceptual, External)

- Database Languages (DDL, DML, DCL, TCL)

- DBMS vs RDBMS vs NoSQL

- Popular DBMS software (MySQL, PostgreSQL, MongoDB, Oracle, SQLite, etc.)

---

## 🔷 2. Data Models

- Relational Model (most important)

- Hierarchical Model

- Network Model

- Object-Oriented Data Model

- Entity-Relationship Model (ER Model)

- Document and Key-Value Models (NoSQL context)

---

## 🔷 3. ER Model & Design

- Entity, Entity Set

- Attributes (Simple, Composite, Derived, Multivalued)

- Relationships (Unary, Binary, Ternary)

- Cardinality & Participation Constraints

- Keys in ER (Super Key, Candidate Key, Primary Key, Foreign Key)

- Weak vs Strong Entity

- Generalization, Specialization, Aggregation

- Converting ER to Relational Schema

---

## 🔷 4. Relational Model

- Structure of a Relational Table

- Tuple, Attribute, Relation

- Domain, Degree, Cardinality

- Integrity Constraints:

- Domain Constraint

- Entity Integrity

- Referential Integrity

- Schema Diagrams

---

## 🔷 5. Relational Algebra & Relational Calculus

- Relational Algebra Operators:

- Selection, Projection

- Cartesian Product

- Union, Intersection, Set Difference

- Rename (ρ)

- Join Operations (Theta Join, Natural Join, Outer Join, Semi Join)

- Division Operator

- Relational Calculus:

- Tuple Relational Calculus

- Domain Relational Calculus

- Expressing queries in Relational Algebra and Calculus

---

## 🔷 6. SQL (Structured Query Language)

### Basics:

- CREATE, INSERT, UPDATE, DELETE, SELECT

- WHERE, ORDER BY, GROUP BY, HAVING

- DISTINCT, LIKE, BETWEEN, IN, IS NULL

- LIMIT, OFFSET

### Joins:

- INNER JOIN

- LEFT, RIGHT, FULL OUTER JOIN

- SELF JOIN

- CROSS JOIN

- NATURAL JOIN

### Subqueries:

- Scalar, Correlated, Nested

- EXISTS vs IN vs ANY vs ALL

### Advanced SQL:

- Aggregation Functions: COUNT, SUM, AVG, MAX, MIN

- CASE statements

- COALESCE, NULLIF

- Window Functions (ROW\_NUMBER, RANK, DENSE\_RANK, LEAD, LAG, NTILE)

- Common Table Expressions (CTE), WITH clause

- Views (CREATE, DROP, Updateable Views)

- Indexing in SQL (Create, Drop)

- Triggers

- Stored Procedures and Functions

- Transactions (BEGIN, COMMIT, ROLLBACK)

- Constraints (PRIMARY, FOREIGN, UNIQUE, CHECK, DEFAULT, NOT NULL)

---

## 🔷 7. Keys and Constraints

- Super Key, Candidate Key

- Primary Key

- Alternate Key

- Foreign Key

- Composite Key

- Unique Key

- Nullability

- Check Constraints

---

## 🔷 8. Normalization

- Functional Dependency (FD)

- Armstrong's Axioms

- Closure of FDs

- Minimal Cover

- Types of Normal Forms:

- 1NF

- 2NF

- 3NF

- BCNF

- 4NF (Multivalued Dependencies)

- 5NF (Join Dependency)

- 6NF (very rare, temporal databases)

- Denormalization

---

## 🔷 9. Indexing

- Need for Indexing

- Types of Indexing:

- Single-Level Index

- Multi-Level Index

- Clustered vs Non-Clustered Index

- Primary, Secondary Index

- Dense vs Sparse Index

- B+ Tree Indexing (very important)

- Hash Indexing

- Composite Indexing

- Covering Index

- Bitmap Index (for OLAP)

---

## 🔷 10. Transactions and Concurrency Control

- What is a Transaction?

- ACID Properties (Atomicity, Consistency, Isolation, Durability)

- Transaction States

- Serializability:

- Conflict Serializable

- View Serializable

- Anomalies:

- Dirty Read

- Non-repeatable Read

- Phantom Read

- Concurrency Control Protocols:

- Lock-Based (Shared, Exclusive)

- 2-Phase Locking (2PL)

- Strict 2PL

- Time-Stamp Based Protocols

- Validation-based Protocols

- Deadlock:

- Deadlock Detection, Prevention, Avoidance

- Wait-for Graphs

- Starvation

---

## 🔷 11. Recovery and Logging

- Failure Types

- Recovery Techniques:

- Log-Based Recovery

- Deferred and Immediate Update

- Write-Ahead Logging (WAL)

- ARIES Protocol (Analysis, Redo, Undo)

- Checkpoints

- Shadow Paging

- Cascading Aborts

- Idempotency in Recovery

---

## 🔷 12. File Organization & Storage

- File Types (Heap, Sorted, Hashed)

- Pages, Blocks, Records

- Slotted Page Structure

- Fixed Length vs Variable Length Records

- Buffer Management

- Block Access and Cost Models

---

## 🔷 13. Query Processing & Optimization

- Query Parsing and Translation

- Query Execution Plan (QEP)

- Cost Estimation

- Heuristics for Query Optimization

- Join Order Optimization

- Materialized Views

- Query Caching

- Pipelining vs Materialization

---

## 🔷 14. Distributed Databases

- Characteristics & Advantages

- Fragmentation:

- Horizontal

- Vertical

- Replication

- CAP Theorem (Consistency, Availability, Partition Tolerance)

- BASE vs ACID

- Two-Phase Commit (2PC), Three-Phase Commit (3PC)

- Eventual Consistency

- Sharding and Partitioning

---

## 🔷 15. NoSQL Databases

- Characteristics of NoSQL

- Types:

- Key-Value Stores (Redis)

- Document Stores (MongoDB)

- Column-Family Stores (Cassandra, HBase)

- Graph Databases (Neo4j)

- When to use NoSQL

- Differences from RDBMS

---

## 🔷 16. Database Security

- Authentication and Authorization

- Roles and Privileges

- SQL Injection

- Data Masking and Encryption

- Audit Trails

- GRANT and REVOKE

---

## 🔷 17. Miscellaneous & Practical Topics

- Database Sharding

- Database Federation

- Event Sourcing and CQRS

- ORM (Object Relational Mapping) – Hibernate, JPA

- Backups and Snapshots

- Schema Migration

- Read Replicas, Failover, High Availability

- Real-Time vs Batch Processing

- OLAP vs OLTP

- ETL Process

- Data Warehousing

- Materialized Views

- Time-series Databases

---

## 🔷 18. Interview-Oriented Practice

- Normalize a given schema to 3NF / BCNF

- Write SQL queries (easy to advanced, subqueries, joins, window)

- ER to relational conversion

- Explain transaction anomalies and fix them

- Implement a mini relational DBMS (theoretical)

- Design a scalable DB (System Design context)

- How indexes help in query optimization

- ACID vs BASE with examples

---