Fundamentals of Database

Short Notes based on ***Exit Exam***

Zemzem labs

1. Introduction to DBMS

What is a Database?

A structured collection of data stored electronically.

DBMS (Database Management System)

Software that manages:

- Data storage
- Data retrieval
- Security
- Concurrency
- Transactions

Advantages of DBMS

- Reduces redundancy
- Ensures consistency
- Supports concurrency
- Centralized control
- Data integrity
- Better security

2. Database Models

Hierarchical Model

- Tree-like structure
- Parent-child
- One-to-many

Network Model

- Graph-like
- Many-to-many

Relational Model (most used)

- Tables (relations)

- Rows (tuples)
- Columns (attributes)

Object-Oriented Model

- Supports objects, classes, inheritance

3. Relational Model Concepts

Keys

- **Primary Key:** uniquely identifies rows
- **Candidate Key:** potential primary keys
- **Super Key:** any key that can uniquely identify rows
- **Foreign Key:** references another table's primary key
- **Composite Key:** multiple attributes as key

Constraints

- NOT NULL
- UNIQUE
- CHECK
- PRIMARY KEY
- FOREIGN KEY

4. SQL – Structured Query Language

Basic SQL Commands

DDL (Data Definition Language)

- CREATE
- ALTER
- DROP
- TRUNCATE

DML (Data Manipulation Language)

- SELECT
- INSERT
- UPDATE

- DELETE

DCL (Data Control Language)

- GRANT
- REVOKE

TCL (Transaction Control Language)

- COMMIT
- ROLLBACK
- SAVEPOINT

5. Joins

Types of Joins

- **INNER JOIN** – matching rows only
- **LEFT JOIN** – all rows from left table
- **RIGHT JOIN** – all rows from right table
- **FULL JOIN** – all rows from both
- **CROSS JOIN** – cartesian product

6. Normalization

Process of organizing data to reduce redundancy.

Normal Forms

- **1NF:** atomic values
- **2NF:** no partial dependency
- **3NF:** no transitive dependency
- **BCNF:** every determinant is a candidate key
- **4NF:** no multi-valued dependency

Goals

- Reduce redundancy
- Improve data integrity
- Avoid anomalies (update, insert, delete)

7. Functional Dependencies (FD)

FD: $X \rightarrow Y$
Meaning: attribute Y depends on attribute X.

Closure of a Set (X^+)

All attributes that can be determined by X.

Keys from FD

- Compute closure
- Identify minimal super keys
- Pick smallest → candidate key

8. Indexing

Improves **search speed**.

Types

- Primary index
- Secondary index
- Clustered index
- Non-clustered index
- B+ Tree index (most widely used)

Hashing

- Direct access
- Good for exact match lookups
- Poor for range queries

9. Transactions

Transaction Properties (ACID)

1. **Atomicity** – all or nothing
2. **Consistency** – database remains valid
3. **Isolation** – concurrent transactions independent
4. **Durability** – changes persist after commit

Transactions States

- Active
- Partially committed
- Committed
- Failed
- Aborted

10. Concurrency Control

Needed because multiple transactions run simultaneously.

Problems

- Lost update
- Dirty read
- Unrepeatable read
- Phantom read

Techniques

- Locking (binary, shared, exclusive)
- Two-Phase Locking (2PL)
    - Growing phase
    - Shrinking phase
- Timestamp ordering
- Multiversion concurrency control (MVCC)

11. Deadlocks in DB

Deadlock Methods

- Deadlock prevention
- Deadlock detection
- Deadlock recovery
- Timeout-based
- Wait-die & Wound-wait schemes

12. Database Storage & Files

Data Storage Formats

- Heap files
- Sorted files
- Hash files

RAID Levels

- RAID 0: striping
- RAID 1: mirroring
- RAID 5: distributed parity
- RAID 10: mirroring + striping

13. Query Processing & Optimization

Steps

1. Parsing
2. Query planning
3. Query optimization
4. Execution

Optimization Techniques

- Use indexes
- Use join ordering
- Selection pushdown
- Projection pushdown

14. NoSQL Databases

Not only SQL — for big data, scalable systems.

Types

- Document-based (MongoDB)
- Key-value (Redis)
- Column-store (Cassandra)
- Graph databases (Neo4j)

Differences from SQL

- Schema-less
- Horizontal scaling
- High-performance reads/writes

⭐ Ultra-Condensed Exam Summary

- DBMS = manages data efficiently
- Keys = primary, foreign, composite
- SQL = DDL/DML/DCL/TCL
- Joins = inner, left, right, full
- Normalization = 1NF → BCNF
- ACID properties essential for transactions
- 2PL = most common concurrency control
- Indexes = speed up search
- NoSQL = non-relational scalable DBs