

SQL Database – Understanding Notes

1. What is a Database?

A database is a structured collection of data organized for efficient storage, retrieval, and management. It acts as a central repository for applications and users to store persistent data.

2. Components of a Database System

- Tables – Store data in rows (records) and columns (attributes).
- Schema – Defines database structure, tables, data types, and relationships.
- Queries – SQL statements used to retrieve, insert, update, and delete data.
- DBMS – Software that manages databases, security, backup, and integrity.
- Indexes – Improve query performance for faster data retrieval.
- Transactions – Ensure data consistency using commit and rollback operations.

3. Types of Databases

- Relational Databases – Structured tables, use SQL, enforce relationships with primary & foreign keys.
- NoSQL Databases – Flexible schema, handle unstructured/semi-structured data.
- Object-Oriented Databases – Store data as objects.
- Graph Databases – Store data as nodes and edges for complex relationships.

4. Common Database Operations

- CRUD – Create, Read, Update, Delete.
- Joins – Combine data from multiple tables.
- Aggregation – SUM, COUNT, AVG, MIN, MAX.
- Transactions – Group multiple operations into a single logical unit.

5. Importance of Databases

- Efficient data storage and organization.
- Data integrity through constraints.
- Scalability (vertical and horizontal).
- Security via authentication and encryption.

6. Key Properties of Databases

- Integrity – Ensures data accuracy and consistency.
- Security – Protects data from unauthorized access.
- Availability – Ensures continuous access to data.
- Concurrency – Manages multiple users simultaneously.
- Application Independence – Supports multiple applications accessing the same data.

7. Data Loss & Recovery

- Backup and recovery strategies restore lost data.
- Replication ensures redundancy.
- Disaster recovery plans minimize risk from failures or attacks.

8. SQL vs NoSQL

SQL: Structured tables, predefined schema, uses SQL language, supports relationships.

NoSQL: Flexible schema, handles big data, supports distributed systems, includes document, key-value, column-family, and graph databases.

Conclusion:

Databases are the backbone of modern applications. Understanding their structure, types,

operations, and properties is essential for building scalable, secure, and efficient systems.