

Database Course Documentation

1. Flat File Systems vs. Relational Databases







◆ Flat File Systems:

- Data is stored in plain text files (e.g., CSV, Excel).
- High data redundancy due to lack of structured relationships.
- No direct links between data records.
- **Example:** Manual employee records saved in Excel.
- **Drawbacks:** Difficult to search, sort, update, or enforce security.

◆ Relational Databases:

- Data is organized into related tables using primary and foreign keys.
- Low redundancy due to normalization and table relations.
- Direct support for complex relationships (1:1, 1:M, M:M).
- **Example:** University student management system built using MySQL.
- **Drawbacks:** Requires careful schema design and knowledge of SQL.

2. Advantages of DBMS – Mind Map

-  **Security:** Controls access and permissions.
-  **Integrity:** Ensures accuracy and consistency of data.
-  **Backup:** Enables safe storage and recovery of data.
-  **Redundancy:** Reduces data repetition and inconsistency.
-  **Concurrency:** Supports multiple users accessing data simultaneously.
-  **Data Sharing:** Facilitates data access across teams and systems.



3. Roles in a Database System

| Role | Responsibility |
|--------------------------------------|---|
| System Analyst | Gathers and analyzes system requirements from users and stakeholders. |
| Database Designer | Designs the logical structure of the database using tools like ERD. |
| Database Developer | Builds the actual database, writes SQL queries and implements schema. |
| Database Administrator | Manages security, backups, performance, and database availability. |
| Application Developer | Develops applications that connect to and interact with the database. |
| BI (Business Intelligence) Developer | Works on data analysis, dashboards, reporting, and extracting insights. |

4. Types of Databases

◆ Relational vs. Non-Relational

- **Relational Databases:**

- Structured and organized in tables.
- Supports SQL for querying.
- Examples: MySQL, PostgreSQL, SQL Server.

- **Non-Relational Databases:**

- Store data in flexible formats like documents, key-value pairs, or graphs.
- Ideal for unstructured or semi-structured data.
- Examples: MongoDB, Cassandra.

◆ **Centralized vs. Distributed vs. Cloud Databases**





| Type | Description | Example Use Case |
|-------------|---|--|
| Centralized | All data stored in a single physical location. | Internal payroll system in one office. |
| Distributed | Data is distributed across multiple physical sites. | Global banking systems. |
| Cloud-Based | Hosted on cloud platforms with scalable access and tools. | Amazon RDS, Azure SQL, Google Spanner. |

5. Cloud Storage and Databases




What is Cloud Storage?

Cloud storage allows data to be stored remotely on internet-based servers, enabling accessibility, scalability, and integration with database solutions.

Benefits of Cloud-Based Databases:

-  Access from anywhere
-  Easy scalability (up/down)
-  Managed maintenance and updates
-  Pay-as-you-go pricing

Challenges:

-  Requires reliable internet connectivity
-  Data privacy and security concerns
-  Vendor lock-in or service dependency