

Relational Database Management System (RDBMS)

240110102

Definitions

- ▶ **Data: (Plural of Datum - Single piece of Information)**
 - ▶ Data is a known fact
 - ▶ Can be recorded and have implicit meaning
 - ▶ Raw or isolated facts from which the required information is produced
 - ▶ Data can exist in a variety of forms that have meaning in the user's environment
 - ▶ Data can be an object such as documents, images, video segments etc...
- ▶ **Example: A person's Name**
 - ▶ In an organization it is Employee Name
 - ▶ In an institute it is Student Name
 - ▶ In a Bill it is Customer Name
 - ▶ In a firm it is Client Name

Information

- ▶ Information is processed, organised or summarised data
- ▶ It is defined as a collection of related data, that when put together, communicate meaningful and useful message to a recipient who use it to make decision.



Database and Database System

- ▶ Database is a kind of electronic filing cabinet; It is a repository or a container for a collection of computerised data files.
- ▶ Database is a collection of persistent data that is used by the application systems of some given enterprise (eg. Manufacturing company, Bank, Hospital, university, Govt. Dept. etc...)
- ▶ Database system is just a computerised record keeping system
- ▶ Users of the database system can perform variety of operations as
 - ▶ Adding new files to the database
 - ▶ Inserting data into existing files
 - ▶ Retrieving data from the existing file
 - ▶ Deleting data from the existing file
 - ▶ Modifying data in the existing file
 - ▶ Removing existing files from the database

Disadvantages of File-Oriented System

- ▶ Data redundancy
- ▶ Data inconsistency
- ▶ Program - Data dependence
- ▶ Poor data control
- ▶ Limited data sharing
- ▶ Inadequate data manipulation capabilities
- ▶ Excessive programming efforts
- ▶ Security problems

Functions/ Advantages of DBMS

- ▶ **Controlling Redundancy:** Data redundancy leads to wasted storage space, duplication of effort (when multiple copies of a datum need to be updated), and a higher likelihood of the introduction of inconsistency
- ▶ **Restricting Unauthorized Access:** provide a security and authorization subsystem, which is used for specifying restrictions on user accounts. Common kinds of restrictions are to allow read-only access (no updating), or access only to a subset of the data
- ▶ **Providing Persistent Storage for Program Objects:**
- ▶ **Providing Storage Structures for Efficient Query Processing:** The DBMS maintains indexes (form of trees and/or hash tables) that are utilized to improve the execution time of queries and updates.
- ▶ **Providing Backup and Recovery:** The subsystem having this responsibility ensures that recovery is possible in the case of a system crash during execution of one or more transactions.
- ▶ **Providing Multiple User Interfaces:**
- ▶ **Representing Complex Relationships Among Data:**
- ▶ **Enforcing Integrity Constraints:**

Difference between File System and Database Management System (DBMS)

Feature	File System	Database Management System (DBMS)
1. Definition	A file system is a method for storing and organizing files on storage devices like HDDs or SSDs.	A DBMS is a software system that enables creation, manipulation, and management of databases.
2. Data Storage	Data is stored in separate files, usually in formats like .txt, .csv, etc.	Data is stored in structured tables in a database.
3. Data Access	Manual access or through basic programming logic (e.g., file handling in Python).	Accessed using structured queries (e.g., SQL).
4. Data Redundancy	High — same data may be stored in multiple files.	Controlled through normalization and relational design.
5. Data Integrity	Hard to enforce; handled manually in the application code.	Enforced through constraints, keys, and rules in DBMS.
6. Security	Basic file-level security (read/write permissions).	Advanced user-level and object-level security controls.
7. Concurrency	Difficult — simultaneous access may corrupt data.	Managed through transactions, locks, and concurrency control.
8. Backup and Recovery	Needs manual handling via system tools or scripts.	Built-in backup and recovery options in DBMS.
9. Querying	No support for complex queries — needs procedural code.	Supports powerful query languages like SQL.
10. Scalability	Not scalable for large or complex datasets.	Scalable and efficient for large enterprise applications.