**INTRODUCTION TO DBMS, RDBMS**

**DATABASE**

A database is a structured collection of data that is organized in a way that makes it easy to manage, access, and update. It serves as a centralized repository for storing various types of information, ranging from simple text records to complex multimedia files.

Characteristics of a database include:

* Structured Format
* Data Independence
* Data Integrity
* Concurrency control

**DATABASE MANAGEMENT SYSTEM(DBMS)**

* Software system for managing databases.
* Facilitates creation, manipulation, and retrieval of data.
* Acts as an interface between users/applications and the database.
* Ensures data integrity, security, and concurrency control.
* Provides tools for data definition, manipulation, and control.
* Manages transactions, backups, and recovery processes.
* Optimizes query performance and database operations.

Key components and features of a DBMS include:

1. **Data Definition Language(DDL):** Allows users to define the database structure, including creating, modifying, and deleting database objects such as tables, indexes, and constraints.
2. **Data Manipulation Language(DML):** Enables users to retrieve, insert, update, and delete data from the database. Common DML commands include SELECT, INSERT, UPDATE, and DELETE.
3. **Data Control Language(DCL):** Controls access to the database by granting or revoking permissions to users and specifying access rights for different database objects.
4. **Transaction Management:** Ensures the integrity and consistency of data by grouping operations into transactions, which are atomic units of work that are either fully completed or fully rolled back in case of failure.
5. **Concurrency Control:** Manages simultaneous access to the database by multiple users or applications, preventing conflicts and ensuring data consistency.

**RELATIONAL** **DATABASE MANAGEMENT SYSTEM(RDBMS)**

* Type of DBMS that organizes data into tables (relations).
* Structures data in rows (tuples) and columns (attributes).
* Establishes relationships between tables using keys (primary, foreign).
* Enforces integrity constraints to maintain data accuracy and consistency.
* Supports SQL (Structured Query Language) for querying and manipulating data.
* Provides ACID properties (Atomicity, Consistency, Isolation, Durability) for transactions.

**ACID PROPERTIES**

The ACID properties are a set of four characteristics that ensure the reliability and consistency of transactions in a database system.

1. ATOMICITY: All operations within a transaction are treated as a single unit of work, ensuring that either all operations are completed successfully or none of them are.
2. CONSISTENCY: Ensures that the database remains in a valid state before and after the execution of a transaction, preserving integrity constraints and rules.
3. ISOLATION: Transactions operate independently of each other, ensuring that the execution of one transaction is isolated from other concurrent transactions.
4. DURABILITY: Once a transaction is committed, its changes are permanent and persist even in the event of system failures, crashes, or restarts.

**ADVANTAGE OF DBMS**

* Data Integration: DBMS allows for the integration of data from multiple sources into a single database, providing a unified view of the data.
* Data Security: DBMS provides mechanisms for access control, authentication, and encryption to protect sensitive data from unauthorized access.
* Data Consistency: DBMS enforces integrity constraints to ensure that data remains consistent and accurate throughout its lifecycle.
* Data Independence: DBMS separates the database structure from the application programs, allowing for changes to the database schema without affecting the applications.

**DISADVANTAGE OF DBMS**

* Complexity: Implementing and managing a DBMS can be complex and require specialized knowledge and skills.
* Cost: Licensing fees, hardware requirements, and maintenance costs associated with DBMS can be significant.
* Performance Overhead: DBMS adds overhead to data access and manipulation operations, impacting system performance.
* Single Point of Failure: Reliance on a single DBMS instance can pose a risk of system failure if it becomes unavailable.

**ADVANTAGE OF RDBMS**

* Structured Data Storage: RDBMS organizes data into tables with rows and columns, providing a structured format for data storage and retrieval.
* Data Integrity: RDBMS enforces integrity constraints such as primary keys, foreign keys, and unique constraints to maintain data integrity.
* Query Optimization: RDBMS optimizes query performance through indexing, query optimization techniques, and caching mechanisms.
* Data Independence: RDBMS separates the logical and physical aspects of data storage, allowing for changes to the physical storage without affecting the logical structure.

**DISADVANTAGE OF RDBMS**

* Scalability: Scaling an RDBMS to handle large volumes of data and high transaction rates can be challenging and may require additional hardware or software solutions.
* Complexity: Designing and managing relational databases can be complex, especially for databases with complex data models and relationships.
* Limited Flexibility: RDBMS is based on a fixed schema, which can limit flexibility when dealing with evolving data requirements.
* Performance Overhead: RDBMS is based on a fixed schema, which can limit flexibility when dealing with evolving data requirements.

*Mongo DB, Cassandra are some popular examples of DBMS while MySQL, Oracle Database, Microsoft SQL Server are the most common examples of RDBMS.*

**NORMALISATION**

Normalization is a database design technique used to minimize redundancy and dependency by organizing data into tables and defining relationships between them. Here are different types of normalization techniques commonly used in relational databases:

1. **First Normal Form (1NF):**

* Ensures that each column in a table contains atomic values, meaning each value is indivisible.
* Eliminates repeating groups within rows and ensures that each attribute has a single value.
* Example: Splitting a multi-valued attribute into separate columns.

1. **Second Normal Form (2NF):**

* Builds upon 1NF by removing partial dependencies, where non-key attributes depend on only a part of the primary key.
* Requires that every non-prime attribute is fully functionally dependent on the entire primary key.
* Example: Breaking a table into two tables to remove partial dependencies.

1. **Third Normal Form (3NF):**

* Builds upon 2NF by removing transitive dependencies, where non-key attributes depend on other non-key attributes.
* Ensures that every non-prime attribute is non-transitively dependent on the primary key.
* Example: Moving attributes that are dependent on other non-key attributes into a separate table.

1. **Boyce-Codd Normal Form (BCNF):**

* A stricter form of 3NF where every determinant is a candidate key.
* Ensures that there are no non-trivial functional dependencies where a determinant is not a superkey.
* Example: Decomposing tables further to remove dependencies on candidate keys.

1. **Fourth Normal Form (4NF):**

* Addresses multi-valued dependencies, where one or more non-key attributes depend on a multi-valued attribute.
* Ensures that there are no non-trivial multi-valued dependencies.
* Example: Splitting a table into two or more tables to remove multi-valued dependencies.

1. **Fifth Normal Form (5NF):**

* Also known as Project-Join Normal Form (PJ/NF).
* Addresses join dependencies, where some information is only available by joining multiple tables.
* Ensures that there are no non-trivial join dependencies.
* Example: Decomposing tables to remove join dependencies.

**CONCLUSION**

DBMS and RDBMS play a crucial role in organizing and managing data efficiently. While DBMS provides a platform for data management, RDBMS offers a structured approach to organizing data in tables with relational integrity. Understanding these concepts and their advantages and disadvantages is essential for effective database design and management.