



# Introduction of DBMS (Database Management System)

Last Updated : 23 Jan, 2025

---

A **Database Management System (DBMS)** is a software solution designed to efficiently **manage, organize, and retrieve data** in a structured manner. It serves as a critical component in **modern computing, enabling** organizations to **store, manipulate, and secure** their data effectively. From small applications to enterprise systems, DBMS plays a vital role in supporting **data-driven decision-making** and operational efficiency.

In this article, we will explain the **key concepts, benefits**, and types of **Database Management Systems (DBMS)**. We'll also cover how DBMS solutions work, why they're important for modern applications, and what features they offer to ensure **data integrity, security**, and efficient retrieval.

## What is a DBMS?

A **DBMS** is a system that allows users to **create, modify**, and query databases while ensuring **data integrity, security**, and efficient data access. Unlike **traditional file systems**, DBMS minimizes **data redundancy**, prevents inconsistencies, and simplifies data management with features like concurrent access and **backup mechanisms**. It organizes data into **tables, views, schemas**, and reports, providing a structured approach to data management.

### Example:

A university database can store and manage student information, faculty records, and administrative data, allowing seamless retrieval, insertion, and deletion of information as required.

## Key Features of DBMS

1. **Data Modeling:** Tools to create and modify data models, defining the structure and relationships within the database.
2. **Data Storage and Retrieval:** Efficient mechanisms for storing data and executing queries to retrieve it quickly.

3. **Concurrency Control:** Ensures multiple users can access the database simultaneously without conflicts.
4. **Data Integrity and Security:** Enforces rules to maintain accurate and secure data, including access controls and encryption.
5. **Backup and Recovery:** Protects data with regular backups and enables recovery in case of system failures.

## Types of DBMS

There are several types of Database Management Systems (DBMS), each tailored to different data structures, scalability requirements, and application needs. The most common types are as follows:

### 1. Relational Database Management System (RDBMS)

RDBMS organizes data into tables (relations) composed of rows and columns. It uses primary keys to uniquely identify rows and foreign keys to establish relationships between tables. Queries are written in **SQL (Structured Query Language)**, which allows for efficient data manipulation and retrieval.

**Examples:** MySQL, Oracle, Microsoft SQL Server and PostgreSQL.

### 2. NoSQL DBMS

NoSQL systems are designed to handle large-scale data and provide high

Databases SQL MySQL PostgreSQL PL/SQL MongoDB SQL Cheat Sheet SQL Interview Questions My!

store data in various non-relational formats, such as **key-value pairs**, **documents**, **graphs**, or **columns**. These flexible data models enable rapid scaling and are well-suited for unstructured or semi-structured data.

**Examples:** MongoDB, Cassandra, DynamoDB and Redis.

### 3. Object-Oriented DBMS (OODBMS)

OODBMS integrates object-oriented programming concepts into the **database environment**, allowing data to be stored as objects. This approach supports complex data types and relationships, making it ideal for applications requiring advanced data modeling and **real-world simulations**.

**Examples:** ObjectDB, db4o.

## Database Languages

Database languages are specialized sets of **commands** and **instructions** used to define, manipulate, and control data within a database. Each language type plays a distinct role in database management, ensuring efficient **storage**, **retrieval**, and security of data. The primary database languages include:

### 1. Data Definition Language (DDL)

DDL is the short name for Data Definition Language, which deals with database schemas and descriptions, of how the data should reside in the database.

- **CREATE:** to create a database and its objects like (table, index, views, store procedure, function, and triggers)
- **ALTER:** alters the structure of the existing database
- **DROP:** delete objects from the database
- **TRUNCATE:** remove all records from a table, including all spaces allocated for the records are removed
- **COMMENT:** add comments to the data dictionary
- **RENAME:** rename an object

### 2. Data Manipulation Language (DML)

DML focuses on manipulating the data stored in the database, enabling users to retrieve, add, update, and delete data.

- **SELECT:** retrieve data from a database
- **INSERT:** insert data into a table
- **UPDATE:** updates existing data within a table
- **DELETE:** Delete all records from a database table
- **MERGE:** UPSERT operation (insert or update)
- **CALL:** call a PL/SQL or Java subprogram
- **EXPLAIN PLAN:** interpretation of the data access path
- **LOCK TABLE:** concurrency Control

### 3. Data Control Language (DCL)

DCL commands manage access permissions, ensuring data security by controlling who can perform certain actions on the database.

- **GRANT**: Provides specific privileges to a user (e.g., SELECT, INSERT).
- **REVOKE**: Removes previously granted permissions from a user.

### 4. Transaction Control Language (TCL)

TCL commands oversee transactional data to maintain consistency, reliability, and atomicity.

- **ROLLBACK**: Undoes changes made during a transaction.
- **COMMIT**: Saves all changes made during a transaction.
- **SAVEPOINT**: Sets a point within a transaction to which one can later roll back.

### 5. Data Query Language (DQL)

DQL is a subset of DML, specifically focused on data retrieval.

- **SELECT**: The primary DQL command, used to query data from the database without altering its structure or contents.

## Paradigm Shift from File System to DBMS

Before the advent of **modern Database Management Systems (DBMS)**, data was managed using basic file systems on **hard drives**. While this approach allowed users to **store**, **retrieve**, and **update files** as needed, it came with numerous challenges.

A typical example can be seen in a file-based **university management system**, where data was stored in separate sections such as Departments, Academics, Results, Accounts, and Hostels. Certain information like student names and phone numbers was repeated across **multiple files**, leading to the following issues:

#### 1. Redundancy of data

When the same data exists in multiple places, any update must be **manually repeated everywhere**. For instance, if a student changes their phone number, it must be updated across all sections. Failure to do so leads to **unnecessary duplication** and wasted storage.

## 2. Inconsistency of Data

Data is said to be inconsistent if multiple copies of the same data do not match each other. If the Phone number is different in Accounts Section and Academics Section, it will be inconsistent. Inconsistency may be because of typing errors or not updating all copies of the same data.

## 3. Complex Data Access

A user should know the exact location of the file to access data, so the process is very cumbersome and tedious. If the user wants to search the student hostel allotment number of a student from 10000 unsorted students' records, how difficult it can be.

## 4. Lack of Security

File systems provided limited control over who could access certain data. A student who gained access to a file with grades might easily alter it without proper authorization, compromising data integrity.

## 5. No Concurrent Access

File systems were not designed for multiple users working at the same time. If one user was editing a file, others had to wait, which hindered collaboration and slowed down workflows.

## 6. No Backup and Recovery

File systems lacked built-in mechanisms for creating backups or recovering data after a loss. If a file was accidentally deleted or corrupted, there was no easy way to restore it, potentially causing permanent data loss.

## Advantages of DBMS

1. **Data organization:** A DBMS allows for the organization and storage of data in a structured manner, making it easy to retrieve and query the data as needed.
2. **Data integrity:** A DBMS provides mechanisms for enforcing data integrity constraints, such as constraints on the values of data and access controls that restrict who can access the data.
3. **Concurrent access:** A DBMS provides mechanisms for controlling concurrent access to the [database](#), to ensure that multiple users can access the data without conflicting with each other.
4. **Data security:** A DBMS provides tools for managing the security of the data, such as controlling access to the data and encrypting sensitive data.
5. **Backup and recovery:** A DBMS provides mechanisms for backing up and recovering the data in the event of a system failure.
6. **Data sharing:** A DBMS allows multiple users to access and share the same data, which can be useful in a collaborative work environment.

## Disadvantages of DBMS

1. **Complexity:** DBMS can be complex to set up and maintain, requiring specialized knowledge and skills.
2. **Performance overhead:** The use of a DBMS can add overhead to the performance of an application, especially in cases where high levels of concurrency are required.
3. **Scalability:** The use of a DBMS can limit the scalability of an application, since it requires the use of locking and other synchronization mechanisms to ensure data consistency.
4. **Cost:** The cost of purchasing, maintaining and upgrading a DBMS can be high, especially for large or complex systems.
5. **Limited Use Cases:** Not all use cases are suitable for a DBMS, some solutions don't need high reliability, consistency or security and may be better served by other types of data storage.

## Applications of DBMS

1. **Enterprise Information:** Sales, accounting, human resources, Manufacturing, online retailers.
2. **Banking and Finance Sector:** Banks maintaining the customer details, accounts, loans, banking transactions, credit card transactions. Finance: Storing the information about sales and holdings, purchasing of financial stocks and bonds.
3. **University:** Maintaining the information about student course enrolled information, student grades, staff roles.
4. **Airlines:** Reservations and schedules.
5. **Telecommunications:** Prepaid, postpaid bills maintenance.

## Conclusion

A **Database Management System (DBMS)** is an essential tool for efficiently **managing, organizing, and retrieving large volumes** of data across various industries. Its ability to handle data securely, ensure **integrity**, support **concurrent access**, and provide backup and recovery options makes it indispensable for modern **data-driven applications**. While DBMSs come with **complexities** and **costs**, their benefits in terms of data management and security far outweigh the **challenges**, making them a **crucial component** in any data-centric environment.

## FAQs

### What is the primary function of a DBMS?

*A DBMS is designed to store, retrieve, and manage data in a structured format, ensuring data integrity, security, and efficient access for multiple users and applications.*

### How does a DBMS improve data security?

*A DBMS enhances data security by implementing access controls, encryption, and authentication mechanisms that restrict unauthorized access and safeguard sensitive information.*



## What are the main types of DBMS?

*The main types of DBMS include Relational DBMS (RDBMS), NoSQL DBMS, and Object-Oriented DBMS. Each type caters to specific use cases, data structures, and performance requirements.*



Introdu  
of DBM



2.  
Intro  
to  
DBMS

### Introduction of DBMS

[Visit Course](#)[Comment](#)[More info](#)[Advertise with us](#)

### Next Article

[History of DBMS](#)

## Similar Reads

### DBMS Tutorial – Learn Database Management System

Database Management System (DBMS) is a software used to manage data from a database. A database is a structured collection of data that is stored in an electronic device. The data can be text, video, image or any...

7 min read



## Basic of DBMS

---

### Introduction of DBMS (Database Management System)

A Database Management System (DBMS) is a software solution designed to efficiently manage, organize, and retrieve data in a structured manner. It serves as a critical component in modern computing, enabling...

8 min read

### History of DBMS

The first database management systems (DBMS) were created to handle complex data for businesses in the 1960s. These systems included Charles Bachman's Integrated Data Store (IDS) and IBM's Information...

7 min read

### Advantages of Database Management System

Database Management System (DBMS) is a collection of interrelated data and a set of software tools/programs that access, process, and manipulate data. It allows access, retrieval, and use of that data by...

6 min read

### Disadvantages of DBMS

You might have encountered bulks of files/registers either at some office/school/university. The traditional file management system has been followed for managing the information or data at many organizations and by...

9 min read

### Application of DBMS

The efficient and safe management, saving and retrieval of data is made possible by the Database Management Systems. They provide strong solutions for the data management demands and are the...

5 min read

### Need for DBMS

In earlier times, data was stored and retrieved using files in a typical file system. For example: A company might keep separate files for employees' details, customer information, and daily sales. These files could be...

6 min read

### DBMS Architecture 1-level, 2-Level, 3-Level

A Database stores a lot of critical information to access data quickly and securely. Hence it is important to select the correct architecture for efficient data management. Database Management System (DBMS)...

6 min read

### Difference between File System and DBMS

A file system and a DBMS are two kinds of data management systems that are used in different capacities and possess different characteristics. A File System is a way of organizing files into groups and folders and...

6 min read

---

## Entity Relationship Model

---

## Relational Model

---

## Relational Algebra

---

## Functional Dependencies

---

## Normalisation

---

## Transactions and Concurrency Control

---

## Indexing, B and B+ trees

---

## File organization

---



Corporate & Communications Address:

A-143, 7th Floor, Sovereign Corporate  
Tower, Sector- 136, Noida, Uttar Pradesh  
(201305)

Registered Address:

K 061, Tower K, Gulshan Vivante  
Apartment, Sector 137, Noida, Gautam  
Buddh Nagar, Uttar Pradesh, 201305



[Advertise with us](#)

## Company

[About Us](#)  
[Legal](#)  
[Privacy Policy](#)  
[In Media](#)  
[Contact Us](#)  
[Advertise with us](#)  
[GFG Corporate Solution](#)  
[Placement Training Program](#)  
[GeeksforGeeks Community](#)

## DSA

[Data Structures](#)  
[Algorithms](#)  
[DSA for Beginners](#)  
[Basic DSA Problems](#)  
[DSA Roadmap](#)  
[Top 100 DSA Interview Problems](#)  
[DSA Roadmap by Sandeep Jain](#)  
[All Cheat Sheets](#)

## Web Technologies

[HTML](#)  
[CSS](#)  
[JavaScript](#)  
[TypeScript](#)  
[ReactJS](#)  
[NextJS](#)  
[Bootstrap](#)  
[Web Design](#)

## Computer Science

[Operating Systems](#)  
[Computer Network](#)  
[Database Management System](#)  
[Software Engineering](#)  
[Digital Logic Design](#)  
[Engineering Maths](#)  
[Software Development](#)  
[Software Testing](#)

## System Design

[High Level Design](#)  
[Low Level Design](#)

## Languages

[Python](#)  
[Java](#)  
[C++](#)  
[PHP](#)  
[GoLang](#)  
[SQL](#)  
[R Language](#)  
[Android Tutorial](#)  
[Tutorials Archive](#)

## Data Science & ML

[Data Science With Python](#)  
[Data Science For Beginner](#)  
[Machine Learning](#)  
[ML Maths](#)  
[Data Visualisation](#)  
[Pandas](#)  
[NumPy](#)  
[NLP](#)  
[Deep Learning](#)

## Python Tutorial

[Python Programming Examples](#)  
[Python Projects](#)  
[Python Tkinter](#)  
[Web Scraping](#)  
[OpenCV Tutorial](#)  
[Python Interview Question](#)  
[Django](#)

## DevOps

[Git](#)  
[Linux](#)  
[AWS](#)  
[Docker](#)  
[Kubernetes](#)  
[Azure](#)  
[GCP](#)  
[DevOps Roadmap](#)

## Interview Preparation

[Competitive Programming](#)  
[Top DS or Algo for CP](#)

UML Diagrams  
Interview Guide  
Design Patterns  
OOAD  
System Design Bootcamp  
Interview Questions

Company-Wise Recruitment Process  
Company-Wise Preparation  
Aptitude Preparation  
Puzzles

### School Subjects

Mathematics  
Physics  
Chemistry  
Biology  
Social Science  
English Grammar  
Commerce  
World GK

### GeeksforGeeks Videos

DSA  
Python  
Java  
C++  
Web Development  
Data Science  
CS Subjects

---

@GeeksforGeeks, Sanchhaya Education Private Limited, All rights reserved