# **What is data?**

**Data is a collection of raw bytes.**

In computer science or DBMS, data is an efficient model of information, which is stored in a format that is highly efficient for movement over various networks.

**Data is generally a raw format of any information**. It may be text information or any rich formatted information.

Consider you’re visiting a website, your computer downloads all the HTML, CSS and various java-script for that website and renders the page. All this is data download.

**Information**

Information is nothing but that data being **viewed in a structured format.**

Example, when you visited that website. The browser stores your cookies and the website based on your IP address can find the country you’re accessing the website from. Thus, your IP Address which is data becomes information when a meaningful context is applied to it.

## **Units of data storage**

#### **Bit:**

The smallest/fundamental unit of measurement used for measuring data is a single bit, it contains a binary value such as true/false(1 and 0’s)

#### **Byte :**

* + A byte corresponds to 8 bits of data which is the fundamental unit of measurement of data
  + A byte can Store 2n bits i.e 1 byte = 28 bits = 256 different values

#### 

#### **Kilobyte (KB) :**

Files require thousands of bytes to store, textual data files are often measured in **kilobytes(KB).**

#### **Megabytes(MB) :**

Larger files such as images, audio files, videos contain millions of bytes hence visual and audio data stored in **MB**.

#### **Gigabytes(GB) :**

Storage devices store thousands of files in a wide range of formats varying from a simple text file to high-end graphical videos, audio, images, etc to store data in the form of **gigabytes** or even **terabytes (TB).**

#### **Below is a list of standard measure of data storage in increasing order of size**

| **Unit** | **Size** |
| --- | --- |
| Byte(B) | 8 bits |
| KiloByte(KB) | 1000 1 bytes |
| MegaByte(MB) | 1000 2bytes |
| GigaByte(GB) | 1000 3 bytes |
| TeraByte(TB) | 1000 4 bytes |
| PetaByte(PB) | 1000 5 bytes |
| ExaByte(EB) | 1000 6 bytes |
| Zettabyte(ZB) | 1000 7bytes |
| yottabyte(YB) | 1000 8bytes |

#### **Data vs Information**

| **Aspect** | **Data** | **Information** |
| --- | --- | --- |
| **Definition** | Raw, unprocessed facts, numbers, symbols | Processed, organized, meaningful data |
| **Nature** | Can be unstructured or structured | Typically structured and organized |
| **Context** | Context-neutral; may lack meaning | Context-specific; conveys meaning |
| **Purpose** | Foundation for generating information | Used for decision-making, communication |
| **Examples** | List of numbers (e.g., 5, 10, 15, 20) | Average temperature over the past week |
| **Transformation** | Requires processing for meaning | Result of processing and structuring data |
| **Subjectivity** | Objective, no inherent bias | May involve subjectivity in interpretation |
| **Volume** | Can exist in large volumes | Tends to be more concise and focused |

# What is a database?

A database is an electronic place/system where data is stored in a way that can be easily **accessed, managed and updated.**

**To make real use of Data, we need Database management systems. (DBMS)**

The database is a collection of **related data, organized in a structured and meaningful way.** For computers, having the data stored in a way that is allowed faster –

* Storage(entry)
* Access
* Update
* Manipulation

Is also important than just having meaningfully sorted and stored. Consider this, there is a phonebook directory, with 1 million + business contacts in your city and the entry is done randomly.

To view this large data, you need to have the name of the business, phone numbers, addresses arranged together in an alphabetical format. Then only it will be called a database of numbers right.

The database is a container of multiple tables, views, procedures, functions, triggers, and other database objects. The data is stored in the form of tables and can be queried, updated, and manipulated using SQL (Structured Query Language) commands. The DBMS is responsible for managing and maintaining the integrity, security, and consistency of the data in the database.

This is why Database is a most efficient and structured way of storing information together so various operations like entry, addition, updation and manipulation are very easy, along with the view format as well.

#### 

#### **Databases in early systems**

The databases in early systems were stored on tapes, which were read only format, which means once you store data. It is impossible to delete or update or even add any new data on it easily.

Which is why the father of the database system, Founder of Oracle, Larry Ellison found the need for a fully managed database system and launched oracle in the market.

# What is DBMS?

a. A database-management system (DBMS) is a collection of **interrelated data and a set of programs to access that data**. The collection of data, usually referred to as the **database**, contains information relevant to an enterprise. The primary goal of a DBMS is to provide a way to **store and retrieve database information** that is both convenient and efficient.

b. A DBMS is the database itself, along with all the software and functionality. It is used to perform different operations, like **addition, access, updating,** and **deletion** of the data.

**The database management system is software that controls all the different manipulation of stored or to be stored data in a database.** It allows the creation, update, manipulation, definition of a database.

As clear from the name it manages the whole database end to end for whatever operation that may be required. Some examples of DBMS are –

* SQL
* mySQL
* noSQL
* Oracle
* IBM DB2
* PostgreSQL
* MongoDB

A database management system perform the following –

* Database Definition
* Data Updation
* Data Retrieval
* Administration
* Security

#### **Database Definition**

This essentially defines how data is defined in the database. For example we have to create a database for students are PrepInsta users.

We will define the data as follows –

* Unique ID of signed up user
* Unique email ID of signed up user
* First Name
* Last Name
* Passwords etc

#### **Data Updation**

This majorly allows various operations like insertion of the new data, updating new data, deletion of data in the various database tables.

* Insertion – Inserting information of new user signed up on PrepInsta website
* Updation – Updating the password of the user who requested to change it
* Deletion – Deleting the complete or partial data of the user who has requested account deletion on PrepInsta.

#### **Data Retrieval**

The data is stored in various database servers and stored globally. For different instances like requesting login authentication details for a website. You need to communicate and retrieve a lot of data from database tables.

#### **Administration**

Database administration is important for a new employee who is working on a database system for your company. You don’t want to give him deleting capabilities. So you create different roles like –

* Database Administrator – Who calls perform all actions like modification, deletion, updation and creation etc.
* Database Manager – Who can define, update, modify but can’t delete
* Database Editor – Who can just retrieve and insert in the database

**Note** – The above are just for examples. Companies create their custom roles based upon exact requirements.

#### **Security**

The security of the system is also important. You don’t want hackers to steal your confidential database or maybe update your database with false data. Thus for any access to database nodes. There is an authentication that happens along.

**DBMS vs File System**

| Aspect | Database Management System (DBMS) | Filesystem |
| --- | --- | --- |
| Purpose | Manages and organizes structured data with an emphasis on data integrity and efficient retrieval. | Stores and manages files and data, often with a focus on basic file operations. |
| Data Structure | Supports structured data with tables, relationships, and data integrity constraints. | Typically stores unstructured or semi-structured data in files and directories. |
| Data Query and Retrieval | Offers a query language (e.g., SQL) for complex data retrieval and filtering capabilities. | Limited search and retrieval capabilities; primarily based on file attributes. |
| Data Integrity | Enforces data integrity through features like ACID transactions and foreign key constraints. | Limited data integrity checks; relies on external validation and error handling. |
| Scalability | Scales well for handling large datasets and complex queries. | Scaling can be challenging for very large numbers of files or directories. |
| Concurrency Control | Provides robust mechanisms for handling concurrent access and data consistency. | Limited or basic concurrency control features; file locking may be used. |
| Data Security | Offers granular access control and authentication mechanisms for securing data. | Limited access control, often relying on file permissions and access rights. |
| Backup and Recovery | Typically includes features for automated backups, point-in-time recovery, and data replication. | Backup and recovery processes may be less comprehensive and more manual. |
| Complex Data Types | Supports a wide range of complex data types (e.g., blobs, XML, JSON) natively. | Primarily deals with simple file types (text, binary, etc.). |
| Consistency | Ensures data consistency and enforces data integrity rules within the database. | Consistency depends on external processes and applications. |
| Transaction Management | Supports ACID (Atomicity, Consistency, Isolation, Durability) transactions. | Lacks built-in transaction management, typically relying on application logic. |
| Multi-user Support | Designed to handle multiple concurrent users and applications efficiently. | Supports multiple users but may have limitations in handling concurrent access. |
| Data Redundancy and Normalization | Promotes data normalization to minimize redundancy and improve data efficiency. | Data redundancy is common, and normalization is not a primary concern. |
| Example Software | MySQL, Oracle, PostgreSQL, Microsoft SQL Server, MongoDB, etc. | Windows File System, Unix/Linux File Systems, NTFS, ext4, etc. |