# Database Management System (DBMS)

## 1. Define a Database Management System (DBMS). Describe any five advantages of using a DBMS.

A database management system (DBMS) is a collection of programs or software that manages the structure of database and organizes the data and data stored in appropriate way. This DBMS serves as the bridge between the user and the database. It receives all application requests from users and translates them into the complex operations required to fulfill those requests and it hides internal complexity from the GUI or application programs from users. (MyReadingRoom.co.in, 2025)

## Advantages of Database Management system (DBMS)

- Improved Data sharing
- Improved data security
- Data Integrity, Accuracy, and redundancy
- Data Independency
- Scalability

**Improved Data Sharing**: Improved data sharing is the main reason for creating DBMS. It helps to create an environment in which end users have better access to more and get more organized data. Using a DBMS, several users have access to use the database simultaneously without influencing one another.

**Improved Data Security:** Security is the backbone of every organization. DBMS provides strong security systems that help in the protection of data. This data can only be accessed or used by users who have permission to do so. It protects data from any external subjects or users to avoid any data breach.

**Data integrity, accuracy, and redundancy:** Another advantage of DBMS is data integrity and its accuracy. In database data redundancy and data mismatch are the biggest issues, but in DBMS data stored in one place can be accessed from anywhere as needed or need permissions. This saves storage space as well as ensures that the data remains consistent.

**Data Independency:** DBMS allows abstracting the data structure from the users. It means users can interact with data without knowing its complexity in the back end. It simplifies data management and improves the user's experience.

**Scalability:** DBMS allows users to store large amounts of data as well as scale up as requirements or businesses grow. Due to this, it is applicable in small and big companies. As data grows, DBMS can grow with the requirement.

## 2. What is meant by database architecture? Discuss its types with appropriate real-life examples.

The database architecture refers to the structural design and methodology of a database system, which is the form of a core Database Management System (DBMS). The word "architects" indicates how data is stored, organized, and retrieved and plays a crucial role in the efficiency and effectiveness of data management in a database.

### Types of database models are:

- One-tier architecture
- Two-tier architecture
- Three-tier architecture

**one-tier architecture**: In one-tier architecture, the database, user interface, and other application logics are stored in the same machine or computer. Simplify is used where simplicity, cost-effectiveness, and small-scale applications are priorities. One-tier architecture is faster than other architecture because there is no network involved. For example, a local desktop application where the database is embedded or stored the data in the same machine within the application itself or is running entirely on one compute**r.** (MongoDB, 2025)

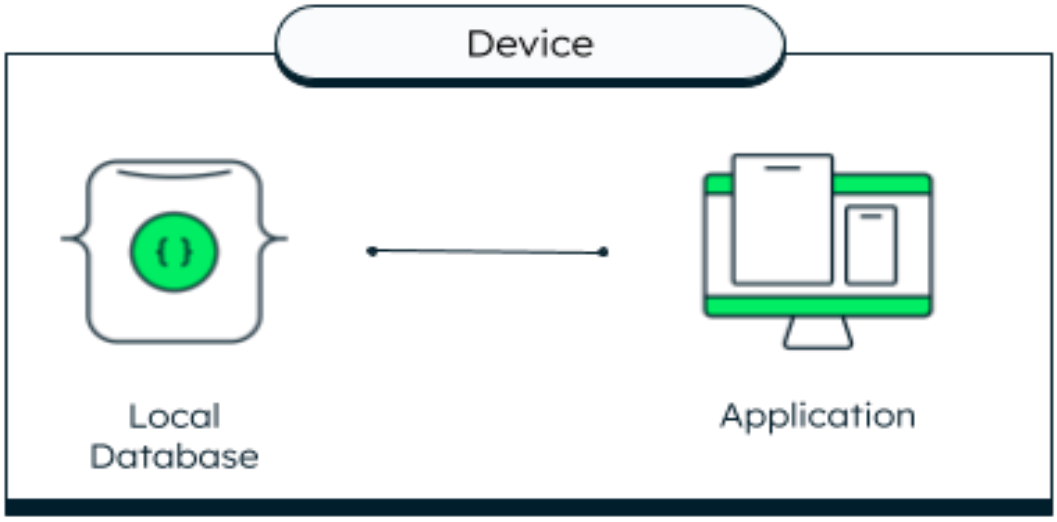


*Figure 1 one-tier architecture*

On a single-tier application, the application and database reside on the same device.

**Two-tier architecture:** Two-tier architecture consists of multiple clients connecting directly to the single database. This type of architecture is known as client-server architecture. This type of architecture is more common when a desktop-based application connects to the single server database. For example, a library management system where staff use a desktop application to interact with clients directly with a database server that stores all library records. (MongoDB, 2025)

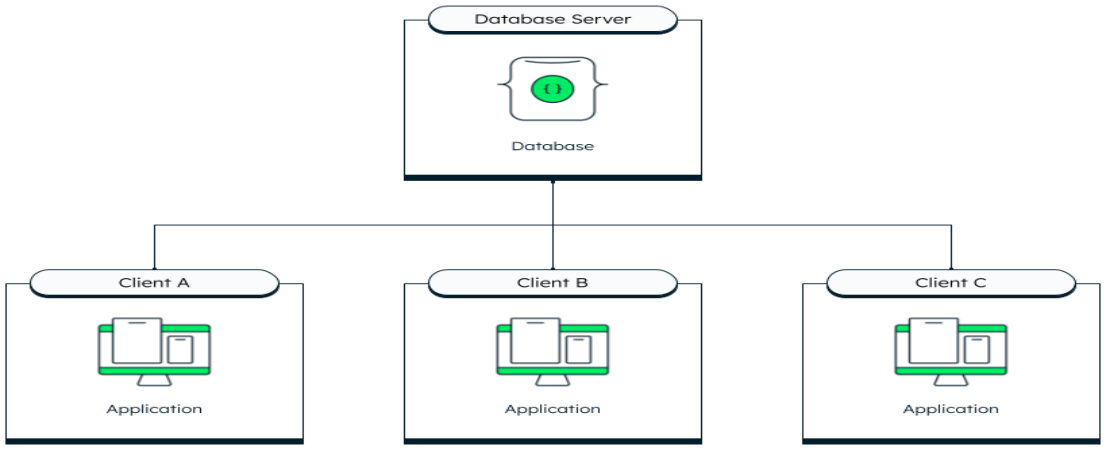


*Figure 2 Tier-two architecture*

In two-tier architecture clients are directly connected to the database.

**Three-tier architecture**: This type of architecture is widely used in modern web applications. In this architecture, clients connect to a backend, which in turn connects to the database. For example, an e-commerce website where the users interact with web browsers as clients and retrieve or store data in a separate database server. This means the user does not directly access the database. (MongoDB, 2025)

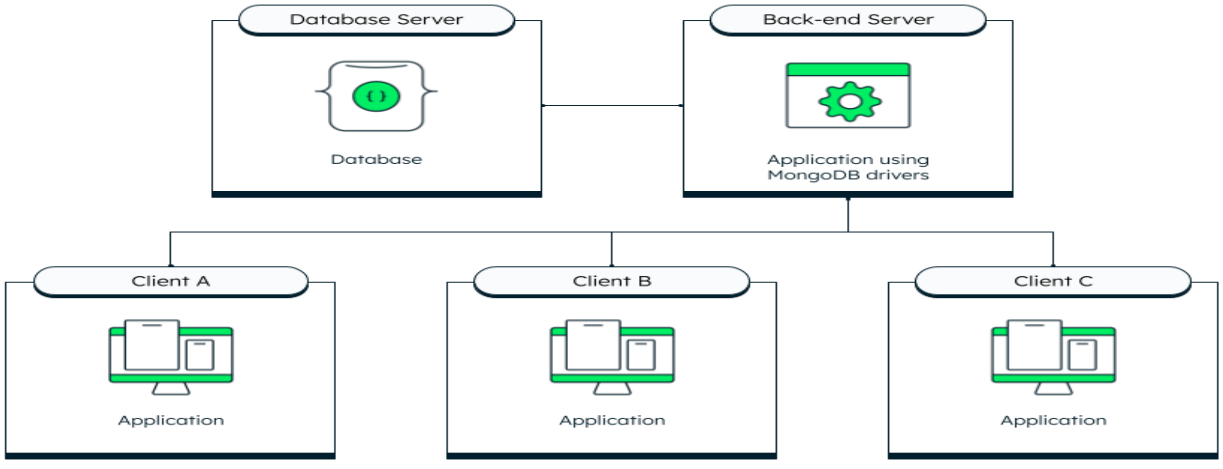


*Figure 3 Tier-three architecture*

Three-tier architecture comprises data, application, and presentation levels.

## 3. Explain the three levels of architecture in Database Management System.

There are three levels of architecture in Database Management system (DBMS):

I. External (view) Level

II. Conceptual Level (Logical Level)

III. Internal Level (Physical Level)

**External Level:** We can view data in terms of logical level tables when we have an external level. To view the data, each external level requires a certain user type. For instance, students are interested in viewing all information pertaining to academics, accounts, and courses, and university professors are interested in viewing student course data. As a result, various users may generate different views. Data abstraction is the primary objective of the external level. (Chaitanya Singh, November 13, 2018)

**Logical Level(Conceptual level):** At this stage, different database tables are used to represent data. Assume that the STUDENT database has STUDENT and COURSE tables that users can see but are not aware of where they are stored. It outlines the type of data that will be kept in the database and is also known as a logical schema. (Chaitanya Singh, November 13, 2018)

**Internal Level(Physical level):** This level stores information about where database objects are located within the data store. The majority of DBMS users don't know where the objects are located. To put it simply, a database's physical level explains how the data is kept on secondary storage devices such as disks, tapes, or servers and provides information on other storage specifics (GeeksForGeek, July 23, 2025)

## 4. Describe the concepts of data independence and data abstraction providing suitable examples.

### Data Abstraction

Data abstraction provides the user with simpler views while hiding the underlying complexity of the data structure. To put it simply, data abstraction wraps its complexity and only displays the outcome, leaving off the implementation and internal workings, only displaying the information that is required. Several levels of abstraction are used to achieve this. There are three categories of data abstraction based on levels: logical, internal, and external levels. A smartphone is an excellent example; all that is visible to us are its buttons, screens, and interface icons, which represent the data abstraction. Only a few clicks on the screen will make a call; it hides the complicated internal workings, such as how it operates, connects to the internet, or stores contacts. (GeeksForGeek, 15 February 2024)

#### Data Independence

The ability to change the schema at one database level without influencing the other levels in the DBMS architecture is known as data independence. This quality is essential for controlling and lowering maintenance expenses. Physical and logical data independence are the two categories of data independence. Physical data independence, for instance, allows a company to swap out its outdated hard drives with faster SSDs. Physical data independence would eliminate the need to alter the database's logical structure or application programs in order to switch storage hardware. A database administrator might add a new attribute to a student's record using logical data independence without modifying the current external views, which only show the student's name and ID. (GeeksForGeek, 15 February 2024)

## LAB QUESTIONS

### 5. Connect as user system.

```
SQL*Plus: Release 11.2.0.2.0 Production on Wed Sep 24 10:00:59 2025

Copyright (c) 1982, 2014, Oracle.  All rights reserved.

SQL> connect system;
Enter password:
Connected.
SQL>
```

*Figure 4 Connected to the system*

### 6.Unlock user hr and give the password as hr for it.

```
SQL> ALTER USER hr ACCOUNT UNLOCK;

User altered.

SQL> ALTER USER hr IDENTIFIED BY hr;

User altered.

SQL>
```

*Figure 5 Unlocked the user*

### 7. Connect as user hr.

```
SQL> CONNECT hr/hr;
Connected.
SQL>
```

*Figure 6 connected to the users*

## 8. Show all the tables in user hr.

```
SQL> SELECT table_name FROM user_tables;

TABLE_NAME
------------------------------
REGIONS
LOCATIONS
DEPARTMENTS
JOBS
EMPLOYEES
JOB_HISTORY
COUNTRIES

7 rows selected.
```

*Figure 7 Show table from the users*

## 9. Disconnect from user hr.

```
SQL> DISCONNECT;
Disconnected from Oracle Database 11g Express Edition Release 11.2.0.2.0 - 64bit Production
SQL>
```

*Figure 8 Disconnected to the users*

## 10.Create a new user as follows: yourname yourLondonMetID (eg, RamaNathBthattrai_1234567).

```
SQL> CONNECT system;
Enter password:
Connected.
SQL> CREATE USER EijkeyalPakhrin_24058849 IDENTIFIED BY eijkeyal_oracle123;

User created.
```

*Figure 9 Created to the new users*

## 11. Grant permission connect and resource to that user.

```
SQL> CONNECT system;
Enter password:
Connected.
SQL> CREATE USER EijkeyalPakhrin_24058849 IDENTIFIED BY eijkeyal_oracle123;

User created.

SQL> GRANT CONNECT, RESOURCE TO EijkeyalPakhrin_24058849;

Grant succeeded.

SQL> CONNECT EijkeyalPakhrin_24058849/eijkeyal_oracle123;
Connected.
SQL>
```

*Figure 10 Granted permission to the user*

## 12. Connect as user you made in QNo 10.

```
SQL> CONNECT EijkeyalPakhrin_24058849/eijkeyal_oracle123;
Connected.
```

*Figure 11 Connected user to the database*

## 13. Create table named Book having columns bookID, bookName, bookAuthor and bookPrice.

```
Connected.
SQL> CREATE TABLE Book (
  2      bookID     NUMBER PRIMARY KEY,
  3      bookName   VARCHAR2(100),
  4      bookAuthor VARCHAR2(100),
  5      bookPrice  NUMBER(10,2)
  6  );

Table created.

SQL>
```

*Figure 12 Created a new table in users*

## 14. Describe table Book.

```
SQL> DESC Book;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 BOOKID                                    NOT NULL NUMBER
 BOOKNAME                                           VARCHAR2(100)
 BOOKAUTHOR                                         VARCHAR2(100)
 BOOKPRICE                                          NUMBER(10,2)

SQL>
```

*Figure 13 Described the created table*

## 15. Delete the table Book.

```
SQL> DROP TABLE Book;

Table dropped.

SQL>
```

*Figure 14 Deleted the existing table*

## References


Chaitanya Singh. (November 13, 2018). *DBMS-Three Level Architecture.* BeginnersBook. Retrieved september 25, 2025, from https://beginnersbook.com/2018/11/dbms-three-level-architecture/

Chaitanya Singh. (November 13, 2018). *DBMS-Three Level Architecture.* BeginnersBook.com. Retrieved September 25, 2025, from : https://beginnersbook.com/2018/11/dbms-three-level-architecture/

GeeksForGeek. (15 February 2024). *Data Abstraction.* GeeksforGeeks (Sanchhaya Education Private Limited). Retrieved september 25, 2025, from https://www.geeksforgeeks.org/dbms/data-abstraction-and-data-independence/

GeeksForGeek. (15 February 2024). *Data Independence.* GeeksforGeeks (Sanchhaya Education Private Limited). Retrieved september 25, 2025, from https://www.geeksforgeeks.org/dbms/data-abstraction-and-data-independence/

GeeksForGeek. (July 23, 2025). *DBMS Architecture 1-level, 2-Level, 3-Level.* GeeksforGeeks, specifically under Sanchhaya Education Private Limited a. Retrieved september 25, 2025, from https://www.geeksforgeeks.org/dbms/dbms-architecture-2-level-3-level/

MongoDB. (2025). *One-tier Architecture.* MongoDB, Inc. Retrieved September 25, 2025, from https://www.mongodb.com/

MongoDB. (2025). *Three-tier Architecture.* MongoDB. Retrieved September 25, 2025, from https://www.mongodb.com/

MongoDB. (2025). *Two-tier Architecture.* MongoDB,Inc. Retrieved september 25, 2025, from https://www.mongodb.com/

MyReadingRoom.co.in. (2025). *Define DBMS and its advantages.* MyReadingRoom.co.in. Retrieved September 25, 2025, from https://www.myreadingroom.co.in/notes-and-studymaterial/65-dbms/462-advantages-and-disadvantages-of-dbms.html