AIMERZ.ai
Aim.Act.Achieve

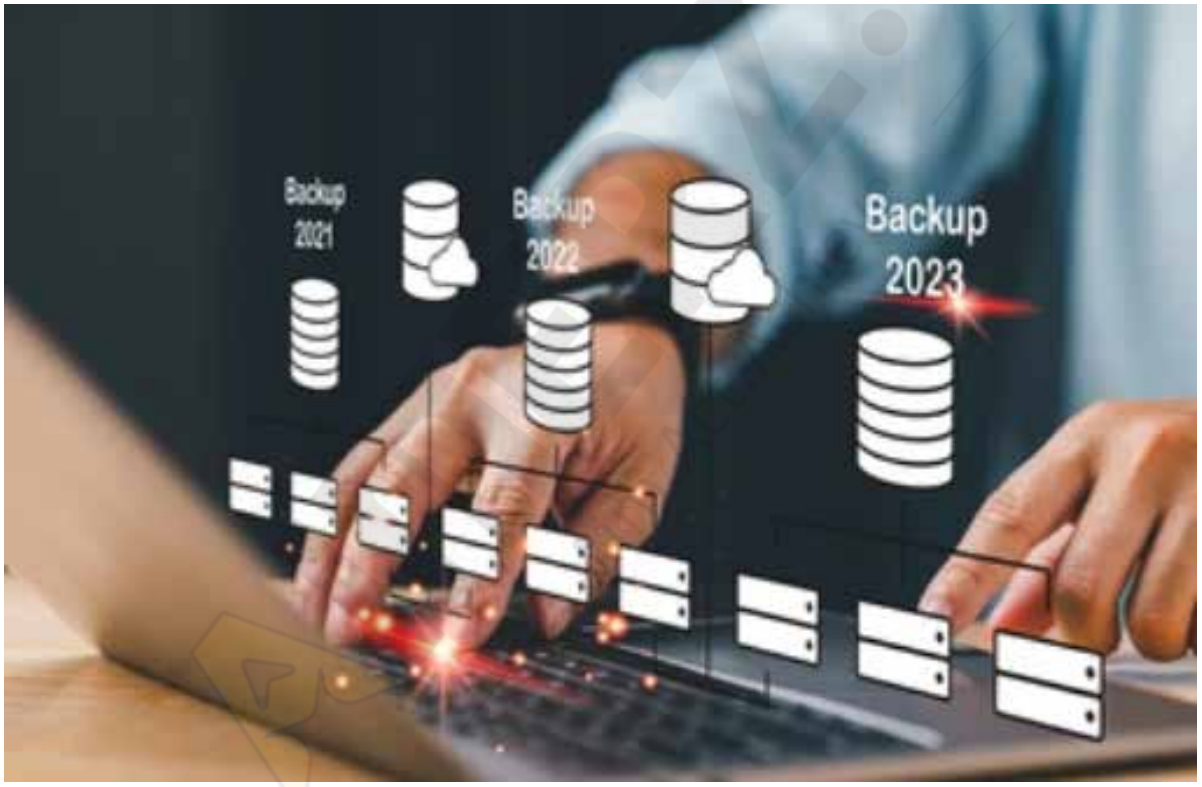
DBMS
INTERVIEW
Questions & Answers

DBMS (DATABASE MANAGEMENT SYSTEM)

DBMS is a software program that lets users handle, arrange, and work with data within databases. In case with the concept, a database is simply an electronic collection of data that can be accessed and managed through a database management system (DBMS).

1. When did DBMS was first introduced?

The Integrated Data Store (IDS) from IBM was one of the first database management systems (DBMS) to be released in the early 1960s. However, IBM researcher Edgar F. Codd originally presented the relational database model, the foundation of the majority of contemporary DBMSs, back in the 1970s.



2. Where can DBMS be used ?

DBMS has a wider usage application areas as follows:

a. Business: Businesses use it extensively to store and manage a variety of data, including financial, personnel, sales, inventory, and customer information.

b. Healthcare: Patient data, such as prescriptions, test results, medical histories, and other data, is managed by DBMS.

c. Education: Database management systems (DBMS) can be used by schools, universities, and other educational institutions to store student data, including attendance logs, grades, and academic achievement.

d. Government: To store and manage information about taxes, licences, permits, and other official records, governments use database management systems (DBMS).

e. E-commerce: Product catalogues, client orders, and payment details are stored and managed by online retailers using database management systems (DBMS).

f. Social Media Platforms: User data, such as profiles, posts, comments, and messages, are stored and managed by social media platforms using database management systems (DBMS).

3. What are the benefits of DBMS?

There are various advantages of using DBMS in any organization and applications:

Effective data management: It is made possible by database management systems (DBMS), which arrange data in a structured manner that facilitates easy searching, retrieval, and manipulation.

Data consistency and integrity: To prevent errors and inconsistencies in the data, database management systems enforce data constraints and rules, such as uniqueness and referential integrity.

Better data security: It is achieved by limiting access to data to authorized users and by putting in place a number of security measures, including auditing, access control, and encryption.

Better data accessibility: By enabling multiple users to view and modify data at once, it makes data easily accessible. This can be accomplished without losing on the data's integrity or consistency.

Productivity gains: By facilitating quicker access to data, minimizing manual data entry, and lowering the possibility of errors, database management systems increase productivity.

Improved decision-making: In addition, it makes better decisions easier by giving people access to precise, current, and thorough data. This facilitates the process of making data-driven, well-informed decisions.

4. What are the advantages of using a DBMS?

1. It keeps Redundancy controlled.
2. Providing multiple user interfaces.
3. Providing backup and recovery
4. Unauthorised access is restricted.
5. Enforcing integrity constraints.

5. What are the types of databases ?

Although there are many different kinds of databases, these three are the most common ones:

Databases with relationships:

The most prevalent kind of database is a relational database. They use tables to store data, with each table denoting a particular object or entity and each row denoting a record or instance of that object.

SQL (Structured Query Language): is used by relational databases for data management and manipulation. SQL is an ANSI (American National Standards Institute) standard computer language for accessing and manipulating database systems.

Databases without Relationships

> **Non-relational databases:** NoSQL databases store data in many different formats, including documents, graphs, and key-value pairs.
>
> They can scale horizontally to accommodate growing volumes of data and are made to handle sizable volumes of semi-structured or unstructured form of data.

6. Describe the Paradigm Shift from using traditional File System to utilizing DBMS.

One file or table at a time access is possible with a file management system (DBMS). Data is directly stored in a collection of files in a file system. It includes flat files, which are files that don't relate to other files at all (a flat file is one that contains just one table). The File System uses files on the hard drive to manage data.
According to their needs, users are free to add, remove, and update files.

Let's look at a file-based university management system as an example: -

> The departments, academic section, results section, accounts section, hostel office, etc. that are responsible for the students have access to their data. Some information, such as the student's roll number, name, father name, address, and phone number, is shared by all sections, but other information is exclusive to that section, such as the hostel allotment number, which is a component of the hostel office.

Let's examine the problems this system consists of:

Redundancy in data: When identical data is duplicated multiple times, it's considered redundant data. Students must update their phone numbers at multiple sections if they wish to change them. In a similar vein, all sections that pertain to that student must have their old records removed.

Data Inconsistency: When different copies of the same data do not match, the data is considered inconsistent.

It will be inconsistent if a phone number appears differently in the Academics and Accounts sections. Typing mistakes or failing to update all instances of the same data could be the cause of an inconsistency.

Unauthorised Access: Data may be accessed without authorization through the File System. A student can alter his marks in an unauthorized manner if he gains access to the file containing them.

No Concurrent Access: Concurrency is the ability for multiple users to access the same data at the same time. Concurrency is not possible with file systems because only one user can access data at a time.

Lack of Backup and Recovery: In the event that a file is lost or corrupted, the file system does not provide for data backup or recovery.

These are the primary causes of the switch from file systems to relational database management systems.

7. What is Relational DBMS ?

Relational database means the data is stored as well as retrieved in the form of relations (tables). The table below shows the relational database with only one relation called STUDENT which stores ROLL_NO, NAME, ADDRESS, PHONE and AGE of students.

STUDENT

Roll_no	Name	Address	Phone	Age
22	Kajal	Kanpur	8768655676	16
21	Ria	Bengaluru	6867878977	26
20	Shivani	Jaipur	7697858687	37
19	Sristi	Pune	8989786868	12
17	Raj	Mumbai	6878679687	24

Database language that is used to create, maintain and retrieve the relational database. Following are some interesting facts about SQL.

• SQL does not care about cases. However, it is advised to use capital letters for keywords (such as SELECT, UPDATE, CREATE, etc.) and small letters for user-defined items (such as table names, column names, etc.).
• The double hyphen "-" can be used at the start of any line in SQL comments.
• SQL is the programming language used by relational databases such as Postgre, MySQL, Oracle, Sybase, and others (explained below). SQL is not used by other non-relational databases, or NoSQL databases, such as MongoDB, DynamoDB, etc.
• Despite an ISO standard for SQL, the majority of implementations have slightly different syntax. As a result, there may be queries that execute properly in SQL Server but not in MySQL.

8. What does Non-Relational Databases do?

No-SQL

A NoSQL database, which stands for non-SQL or non-relational, is one that offers a method for storing and retrieving data. Different methods than tabular relations, which are employed in relational databases, are

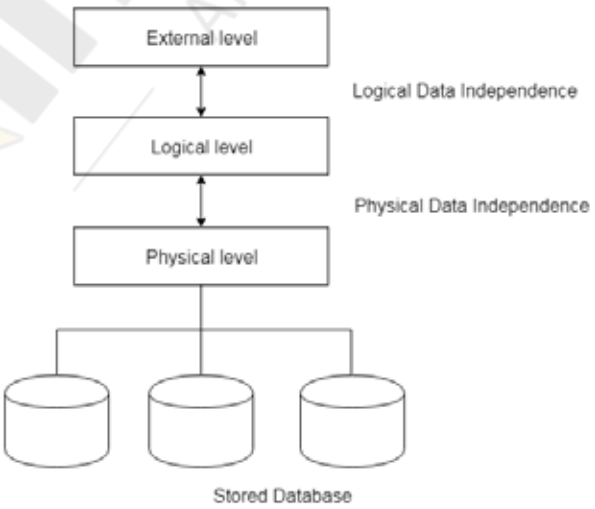
used to model this data. Big data and real-time web applications both use NoSQL databases, and their usage is growing over time.
Not only SQL is another term used to highlight the possibility that NoSQL systems support query languages similar to SQL.
Simpler design, easier for horizontal expansion to machine clusters, and more precise availability control are features of NoSQL databases.

Because NoSQL databases employ different data structures than relational databases do by default, some operations in NoSQL databases execute more quickly. A NoSQL database's suitability is determined by the issue it is intended to address. Relational database tables are sometimes thought to be less flexible than the data structures used by NoSQL databases.

NoSQL database types and the names of the databases systems that belong to that category are as follows:

1. MongoDB is classified as a NoSQL document-based database.
2. Essential value store: Redis, Coherence, and Memcached
3. Tabular: Accumulo, Hbase, and Big Table
4. Documentation-based: Cloudant, CouchDB, MongoDB

9. Explain the concept of Data Independence in Databases ?



A change in data at one level shouldn't have an impact on another level, according to data independence.

This architecture incorporates two different kinds of data independence:

Physical Data Independence: Modifications to the tables' and indexes' actual locations shouldn't have an impact on the external view or conceptual level of the data. The majority of DBMSs have implemented this data independence, which is simple to achieve.

Logical / Conceptual Data Independence: Both the external and conceptual level schemas must have independent data. Thus, modifications to conceptual schema shouldn't have an impact on external schema. For example, altering or removing a table's attributes shouldn't change how the user sees the table. However, because changes in conceptual schema are reflected in the user's perspective, this kind of independence is more challenging to achieve than physical data independence.

10. What is the Entity-Relationship Model in DBMS?

A high-level data model diagram is the entity-relationship model or ER Model. We depict the real-world problem in visual form in this model to make it easier for stakeholders to comprehend.

Features of an Entity-Relationship Model

a. Graphical Representation for Better Understanding: It is really straightforward and easy to comprehend, so developers can use it to interact with stakeholders.

b. Database Design: This approach is extensively used in database design and aids database designers in the creation of databases.

c. ER Diagram: The ER diagram is a visual representation of the model.

11. What is a Database Transaction?

A transaction comprises one or more database access (create/read/update/delete) operations and is a logical unit of database processing.

12. How do you write a SQL query to select all records from a table?

To select all records from a table, use the SELECT statement followed by an asterisk (*) symbol and the FROM clause with the table name. For example, to select all records from a table named employees, the SQL query would be:

Sql > SELECT * FROM employees;

13. What is DBMS transparency?

Transparencies in DBMS means a DBMS system should offer a transparent distribution to the user. In other words, it hides implementation details from the user.

There are 4 kinds of transparency: distribution transparency, transaction transparency, performance transparency, and the DBMS transparency itself.

14. What basic functions are present in every record management system?

Addition of a new Record, Subtraction of a new Record, and Alteration from the record.

15. Describe how structured and unstructured data differ from one another ?

Facts about items and events comprise structured data. Numerical, character, and date data are the most crucial types of structured data.

Tabular data is used to store structured data. Multimedia materials including documents, images, sounds, maps, photos, and video clips are examples of unstructured data. Web servers and databases with web access are the most common places to find unstructured data.

16.Describe some responsibilities of a database administrator.

Establishing database security, planning for database recovery, regulating processing rights and responsibilities, controlling concurrent processing, overseeing the DBMS, and keeping up with the data repository are all examples of managing database structure.

17. Explain the difference between attributes and identifiers.

Entities have attributes. Attributes are properties that describe the entity's characteristics. Entity instances have identifiers. Identifiers are attributes that name, or identify, entity instances.

18. What is the difference between SQL and SQL SERVER?

SQL is a language that provides an interface to RDBMS, developed by IBM. SQL SERVER is a RDBMS just like Oracle, DB2 .

19. What are the disadvantages of using a DBMS?

1) High initial investments in h/w, s/w, and training.

2) Generality that a DBMS provides for defining and processing data.

3) Overhead for providing security, concurrency control, recovery, and integrity functions.

20. What is Normalization in DBMS ?

Normalization is the process of minimizing redundancy from a relation or set of relations. Redundancy in relation may cause insertion, deletion and updating anomalies. So, it helps to minimize the redundancy in relations. Normal forms are used to eliminate or reduce redundancy in database tables.

FIRST NORMAL FORM (1NF): A relation is in first normal form if it does not contain a composite or multi-valued attribute, or it violates first normal form if it does.

Include any multi-valued or composite attribute. A relationship comes first in normal form if that relation's attributes are all single-valued quantities.

SECOND NORMAL FORM (2NF): If a table is in 1NF and all of its non-key attributes depend on the primary key in its entirety (i.e., no partial dependencies), then it is in 2NF.

THIRD NORMAL FORM (3NF): If a table is in 2NF and has no transitive dependencies, meaning every non-key attribute depends only on the primary key, then it is in 3NF.

Boyce-Codd Normal Form (BCNF): A relation R is in BCNF if R is in Third Normal Form and for every FD, LHS is super key. A relation is in BCNF if in every non-trivial functional dependency

X --> Y, X is a superkey.

Fifth Normal Form (5NF) : Project-join normal form is another name for the Fifth Normal Form, or 5NF. If a relation is in 4NF, it is in Fifth Normal Form (5NF) and will not decompose into smaller tables in a lossless manner.

If every join dependency in a relation is implied by the candidate key, you can also assume that the relation is in 5NF.

21. What is Deadlock in DBMS ?

A **deadlock** in DBMS is a situation where two or more transactions are waiting indefinitely for each other to release locks on resources they need. As a result, none of the transactions can proceed, leading to a standstill.

22. What is a DBMS tuple and what is its significance ?

In a database table, a tuple is a single row or record that has a group of associated fields or attributes. A tuple is the full set of information for a particular entity or object that the database is keeping track of it.

23. What is ACID property in DBMS ?

These properties are the fundamental characteristics that ensure the reliability and consistency in any Database Management System (DBMS).

Atomicity: Take all or nothing.

Consistency: Valid state of database is maintained.

Isolation:No interference between any transactions in the database.

Durability: Permanent changes are only done after commit.

24. Describe DDL and DML in brief.

Two significant SQL statement types used in database management are DDL and DML.

Data Definition Language, or DDL

Data Definition Language, or DDL for short, is a language used to specify how a database schema should be organized.

Tables, views, indexes, and stored procedures are examples of database objects that can be created, modified, and removed using DDL commands.

Typical DDL commands are as follows:

- CREATE
- ALTER
- DROP
- TRUNCATE

Data Manipulation Language, or DML

Data Manipulation Language, or DML for short, is used to work with data that is kept in databases.

Data can be added, changed, retrieved, and deleted from a database using DML commands.

Common DML commands are as follows:

- SELECT
- INSERT
- UPDATE
- DELETE

25. What are Database Constraints ?

A rule that is applied to data in a table to guarantee accuracy, consistency, and integrity is known as a database constraint. Limitations impose limitations on the categories of data that can be added, changed, or removed from a database. Constraints with the following common names are: PRIMARY KEY, FOREIGN KEY, UNIQUE, NOT NULL, and CHECK.

A short explanations of the common database constraints is given below:

1. **PRIMARY KEY**: Uniquely identifies each record; cannot be NULL.
2. **FOREIGN KEY**: Enforces a link between two tables; ensures referential integrity.
3. **UNIQUE**: Ensures all values in a column are unique.
4. **NOT NULL**: Ensures a column cannot have NULL values.
5. **CHECK**: Ensures that all values in a column satisfy a specific condition.

26. Define Schema in the context of Databases.

The database schema is a description of a database that is specified during database design and is not likely to change frequently. A schema diagram is a displayed schema. Every item in the schema is referred to as a schema construct.

27. Define normalization and its benefits.

Normalization is the process of organizing data in database to reduce redundancy and improve data integrity. It involves breaking down large tables into smaller, related tables.

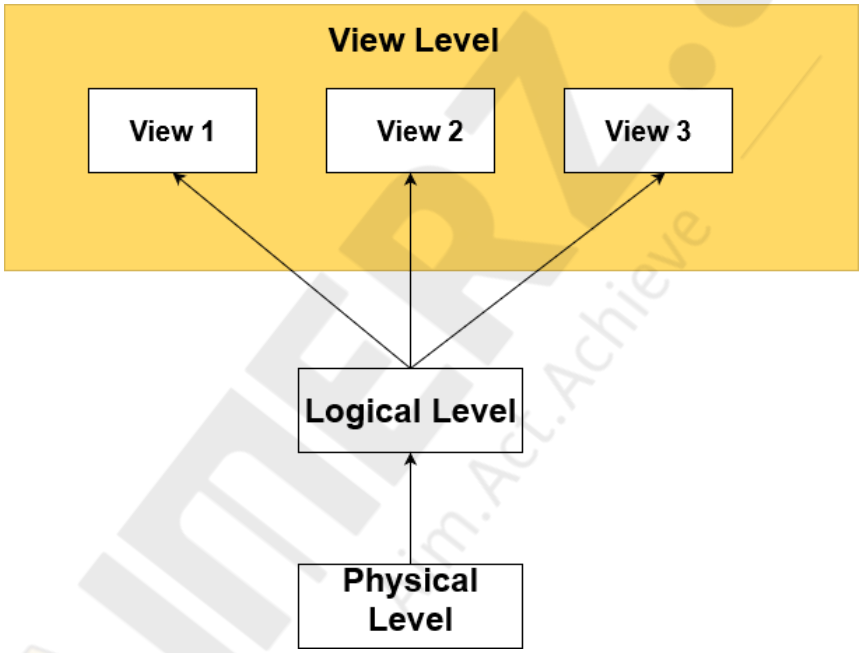
- Reducing data anomalies, such as update anomalies.
- Simplifying data maintenance and improving database performance.

28. Explain the concept of Denormalization.

Denormalization is the process of internationally introducing redundancy into a database to improve query performance. It may be used when read heavy operations are common, and the trade-off between storage and query performance is acceptable.

29. Describe the various DBMS data abstraction levels.

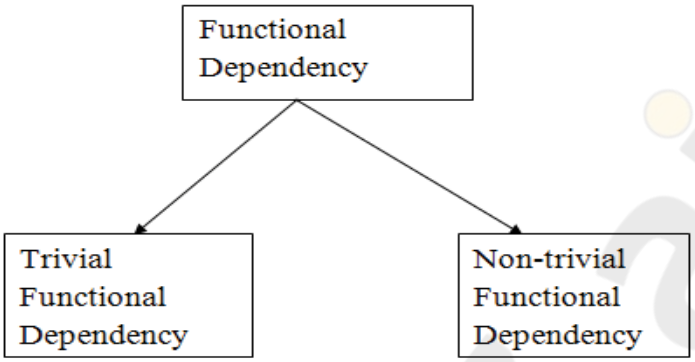
Three tiers of data abstraction exist in DBMSs:



- Physical Level: This level explains the data's placement on the storage medium.
- Logical Level: This level explains the relationships, constraints, tables, columns, and overall logical structure of the database.
- View Level: Views, which are virtual tables created from one or more base tables, give users access to a portion of the database and define how they see the data.

30. In DBMS terms, what is data dependency?

The relationship between various attributes or data elements within a database is referred to as data dependency in DBMS. It explains how modifications to one data element may have an effect on related data elements.



Data Dependencies Types

Two primary categories of data dependencies exist:

Functional Dependency:

This happens when an attribute in a table is determined by another attribute or set of attributes.

For instance, the customer name attribute in a table of customer orders may be determined by the customer ID attribute.

Transitive Dependency:

This happens when there is a functional dependency—that is, when one attribute influences another, which influences a third attribute—between three or more attributes.

For example, the manager attribute in a table of employee data may be determined by the department attribute, which may be determined by the employee ID attribute.

THANK YOU

AIMERZ.ai
Aim.Act.Achieve