<https://www.javatpoint.com/dbms-interview-questions>

What is DBMS?

DBMS is a collection of programs that facilitates users to create and maintain a database. In other words, DBMS provides us an interface or tool for performing different operations such as the creation of a database, inserting data into it, deleting data from it, updating the data, etc. DBMS is a software in which data is stored in a more secure way as compared to the file-based system. Using DBMS, we can overcome many problems such as- data redundancy, data inconsistency, easy access, more organized and understandable, and so on. There is the name of some popular Database Management System- MySQL, Oracle, SQL Server, Amazon simple DB (Cloud-based), etc.

### What is a database?

A Database is a logical, consistent and organized collection of data that it can easily be accessed, managed and updated. Databases, also known as electronic databases are structured to provide the facility of creation, insertion, updating of the data efficiently and are stored in the form of a file or set of files, on the magnetic disk, tapes and another sort of secondary devices. Database mostly consists of the objects (tables), and tables include of the records and fields. Fields are the basic units of data storage, which contain the information about a particular aspect or attribute of the entity described by the database. DBMS is used for extraction of data from the database in the form of the queries.

### What is RDBMS?

RDBMS stands for Relational Database Management Systems. It is used to maintain the data records and indices in tables. RDBMS is the form of DBMS which uses the structure to identify and access data concerning the other piece of data in the database. RDBMS is the system that enables you to perform different operations such as- update, insert, delete, manipulate and administer a relational database with minimal difficulties. Most of the time RDBMS use SQL language because it is easily understandable and is used for often.

### Define a Relation Schema and a Relation.

A Relation Schema is specified as a set of attributes. It is also known as table schema. It defines what the name of the table is. Relation schema is known as the blueprint with the help of which we can explain that how the data is organized into tables. This blueprint contains no data.

A relation is specified as a set of tuples. A relation is the set of related attributes with identifying key attributes

### What are the disadvantages of file processing systems? FPS

* Inconsistent
* Not secure
* Data redundancy
* Difficult in accessing data
* Data isolation
* Data integrity
* Concurrent access is not possible
* Limited data sharing
* Atomicity problem

### What is Relational Algebra?

Relational Algebra is a Procedural Query Language which contains a set of operations that take one or two relations as input and produce a new relationship. Relational algebra is the basic set of operations for the relational model. The decisive point of relational algebra is that it is similar to the algebra which operates on the number.

There are few fundamental operations of relational algebra:

* select
* project
* set difference
* union
* rename,etc.

### What is normalization?

Normalization is a process of analysing the given relation schemas according to their functional dependencies. It is used to minimize redundancy and also used to minimize insertion, deletion and update distractions. Normalization is considered as an essential process as it is used to avoid data redundancy, insertion anomaly, updation anomaly, deletion anomaly.

There most commonly used normal forms are:

* First Normal Form(1NF)
* Second Normal Form(2NF)
* Third Normal Form(3NF)

### What is Denormalization?

Denormalization is the process of boosting up database performance and adding of redundant data which helps to get rid of complex data. Denormalization is a part of database optimization technique. This process is used to avoid the use of complex and costly joins. Denormalization doesn't refer to the thought of not to normalize instead of that denormalization takes place after normalization. In this process, firstly the redundancy of the data will be removed using normalization process than through denormalization process we will add redundant data as per the requirement so that we can easily avoid the costly joins.

### What is the E-R model?

E-R model is a short name for the Entity-Relationship model. This model is based on the real world. It contains necessary objects (known as entities) and the relationship among these objects. Here the primary objects are the entity, attribute of that entity, relationship set, an attribute of that relationship set can be mapped in the form of E-R diagram.

In E-R diagram, entities are represented by rectangles, relationships are represented by diamonds, attributes are the characteristics of entities and represented by ellipses, and data flow is represented through a straight line.

### What is an attribute?

An attribute refers to a database component. It is used to describe the property of an entity. An attribute can be defined as the characteristics of the entity. Entities can be uniquely identified using the attributes. Attributes represent the instances in the row of the database.

For example: If a student is an entity in the table then age will be the attribute of that student.

### What is Join?

The Join operation is one of the most useful activities in relational algebra. It is most commonly used way to combine information from two or more relations. A Join is always performed on the basis of the same or related column. Most complex queries of SQL involve JOIN command.

There are following types of join:

* Inner joins: Inner join is of 3 categories. They are:
  + Theta join
  + Natural join
  + Equi join
* Outer joins: Outer join have three types. They are:
  + Left outer join
  + Right outer join
  + Full outer join

### What is 1NF?

**1NF** is the **First Normal Form**. It is the simplest type of normalization that you can implement in a database. The primary objectives of 1NF are to:

* Every column must have atomic (single value)
* To Remove duplicate columns from the same table
* Create separate tables for each group of related data and identify each row with a unique column

### What is 2NF?

**2NF** is the **Second Normal Form**. A table is said to be 2NF if it follows the following conditions:

* The table is in 1NF, i.e., firstly it is necessary that the table should follow the rules of 1NF.
* Every non-prime attribute is fully functionally dependent on the primary key, i.e., every non-key attribute should be dependent on the primary key in such a way that if any key element is deleted, then even the non\_key element will still be saved in the database.
* **What is 3NF?**

**3NF** stands for **Third Normal Form**. A database is called in 3NF if it satisfies the following conditions:

* It is in second normal form.
* There is no transitive functional dependency.
* For example: X->Z

**Where:**  
X->Y  
Y does not -> X  
Y->Z so, X->Z

### Explain ACID properties

ACID properties are some basic rules, which has to be satisfied by every transaction to preserve the integrity. These properties and rules are:

**ATOMICITY:** Atomicity is more generally known as ?all or nothing rule.' Which implies all are considered as one unit, and they either run to completion or not executed at all.

**CONSISTENCY:** This property refers to the uniformity of the data. Consistency implies that the database is consistent before and after the transaction.

**ISOLATION:** This property states that the number of the transaction can be executed concurrently without leading to the inconsistency of the database state.

**DURABILITY:** This property ensures that once the transaction is committed it will be stored in the non-volatile memory and system crash can also not affect it anymore.

### What is the difference between a shared lock and exclusive lock?

**Shared lock**: Shared lock is required for reading a data item. In the shared lock, many transactions may hold a lock on the same data item. When more than one transaction is allowed to read the data items then that is known as the shared lock.

**Exclusive lock**: When any transaction is about to perform the write operation, then the lock on the data item is an exclusive lock. Because, if we allow more than one transaction then that will lead to the inconsistency in the database

### Describe the types of keys?

**There are following types of keys:**

**Primary key**: The Primary key is an attribute in a table that can uniquely identify each record in a table. It is compulsory for every table.

**Candidate key**: The Candidate key is an attribute or set of an attribute which can uniquely identify a tuple. The Primary key can be selected from these attributes.

**Super key**: The Super key is a set of attributes which can uniquely identify a tuple. Super key is a superset of the candidate key.

**Foreign key**: The Foreign key is a primary key from one table, which has a relationship with another table. It acts as a cross-reference between tables.

**Question :: Composite Vs. Multi Valued Attributes**  
**Answer:**  
**Composite**:  
Composite attributes can be divided into subparts. For example, an attribute name  
could be structured as a composite attribute consisting of first-name, middle-initial,  
and last-name.  
  
**Multivalued:**  
There may be instances where an attribute has a set of values for a specific  
entity. Consider an employee entity set with the attribute phone-number. An  
employee may have zero, one, or several phone numbers, and different employees  
may have different numbers of phones. This type of attribute is said to be  
  
**Composite Attributes :** Attribute Divided into sub parts. Eg. Name (First name,  
Middle Name, last name)  
  
**Multivalued Attributes :** Attribute having more than one values. Eg. Phone  
Number.

| DBMS | Spreadsheet |
| --- | --- |
| It is a collection of inter-related data. | It is an electronic graph sheets with rows and columns. |
| It stores and retrieve the data. | It stores and analyzes the data. |
| It provides support for data organizing and selecting. | It provides support for complex calculations. |
| DBMS capacity is only limited by the computer. | Spreadsheets hold limited amount of information. |
| It is more towards the linkage and organization of data. | It is more towards calculations. |
| It can do simple calculations. | It can do simple data organizing and selecting. |
| Efficient data handling. | Inefficient data handling. |
| Example: MySQL, PostgreSQL, Microsoft Access. | Example: Microsoft Excel, Lotus 1-2-3. |

## Cardinality in a Relation

- One to one

- One to many

- Many to one

high cardinality--> more unique data

less cardinality--> more duplicate data

What are the components of ER model?

There are three basic elements in an ER Diagram: entity, **attribute**, relationship. There are more elements which are based on the main elements. They are weak entity, multi valued **attribute**, derived **attribute**, weak relationship, and recursive relationship

### What is data abstraction in DBMS?

Data abstraction in DBMS is a process of hiding irrelevant details from users. Because database systems are made of complex data structures so, it makes accessible the user interaction with the database.

**For example**: We know that most of the users prefer those systems which have a simple GUI that means no complex processing. So, to keep the user tuned and for making the access to the data easy, it is necessary to do data abstraction. In addition to it, data abstraction divides the system in different layers to make the work specified and well defined.

### What are the three levels of data abstraction?

Following are three levels of data abstraction:

**Physical level**: It is the lowest level of abstraction. It describes how data are stored.

**Logical level**: It is the next higher level of abstraction. It describes what data are stored in the database and what the relationship among those data is.

**View level**: It is the highest level of data abstraction. It describes only part of the entire database.

**For example-** User interacts with the system using the GUI and fill the required details, but the user doesn't have any idea how the data is being used. So, the abstraction level is entirely high in VIEW LEVEL.

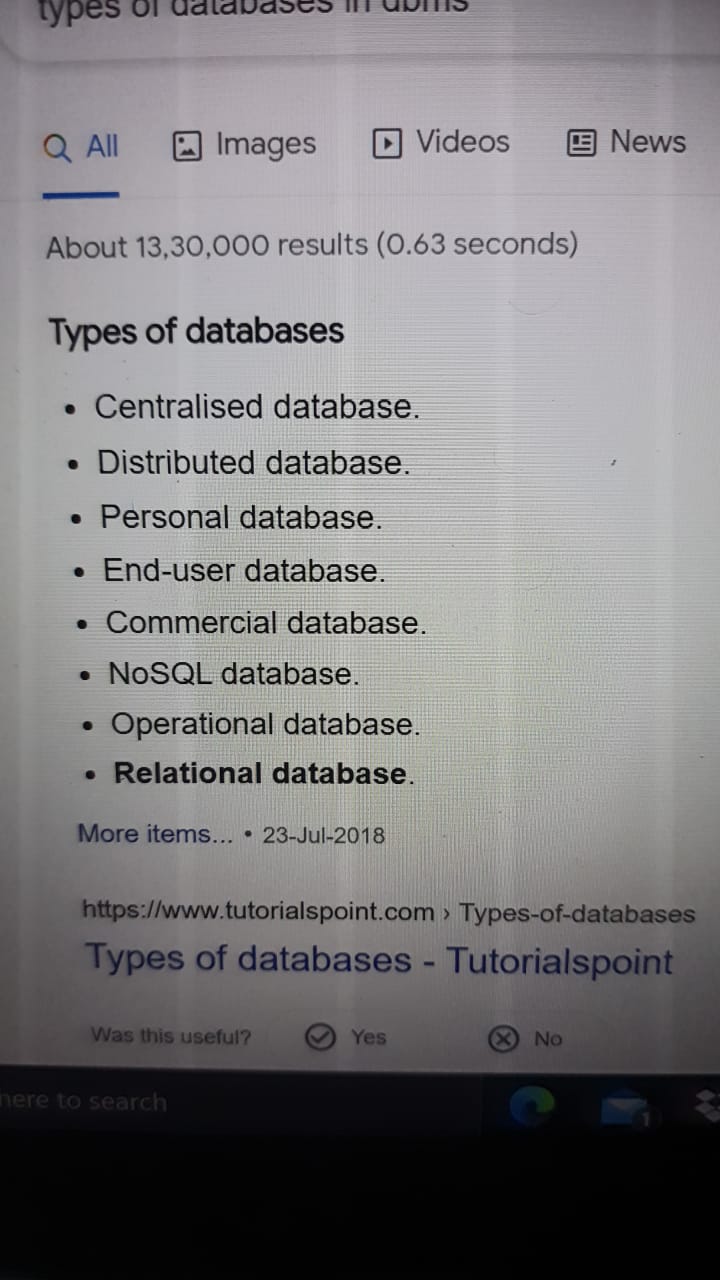
Then, the next level is for PROGRAMMERS as in this level the fields and records are visible and the programmers have the knowledge of this layer. So, the level of abstraction here is a little low in VIEW LEVEL.

And lastly, physical level in which storage blocks are described.

What are the different types of constraints?

**An informational constraint is an attribute of a certain type of constraint, but one that is not enforced by the database manager.**

* NOT NULL **constraints**.
* Unique **constraints**.
* Primary key **constraints**.
* (Table) Check **constraints**.
* Foreign key (referential) **constraints**.
* Informational **constraints**.



# **Deadlock in DBMS**

A deadlock is a condition where two or more transactions are waiting indefinitely for one another to give up locks. Deadlock is said to be one of the most feared complications in DBMS as no task ever gets finished and is in waiting state forever.

**For example:** In the student table, transaction T1 holds a lock on some rows and needs to update some rows in the grade table. Simultaneously, transaction T2 holds locks on some rows in the grade table and needs to update the rows in the Student table held by Transaction T1.

Now, the main problem arises. Now Transaction T1 is waiting for T2 to release its lock and similarly, transaction T2 is waiting for T1 to release its lock. All activities come to a halt state and remain at a standstill. It will remain in a standstill until the DBMS detects the deadlock and aborts one of the transactions.

# DBMS Serializability

When multiple transactions are running concurrently then there is a possibility that the database may be left in an inconsistent state. Serializability is a concept that helps us to check which [schedules](https://beginnersbook.com/2018/12/dbms-schedules/) are serializable. A serializable schedule is the one that always leaves the database in consistent state.

There are four types of database languages:

* **Data Definition Language (DDL)** e.g., CREATE, ALTER, DROP, TRUNCATE, RENAME, etc. All these commands are used for updating the data that?s why they are known as Data Definition Language.
* **Data Manipulation Language (DML)** e.g., SELECT, UPDATE, INSERT, DELETE, etc. These commands are used for the manipulation of already updated data that's why they are the part of Data Manipulation Language.
* **DATA Control Language (DCL)** e.g., GRANT and REVOKE. These commands are used for giving and removing the user access on the database. So, they are the part of Data Control Language.
* **Transaction Control Language (TCL)** e.g., COMMIT, ROLLBACK, and SAVEPOINT. These are the commands used for managing transactions in the database. TCL is used for managing the changes made by DML.

Database language implies the queries that are used for the update, modify and manipulate the data.

 Hashing is an effective technique to calculate the direct location of a data record on the disk without using index structure.

HASHING

Hashing uses hash functions with search keys as parameters to generate the address of a data record. Indexing is a data structure technique to efficiently retrieve records from the database files based on some attributes on which the indexing has been done. Indexing in database systems is similar to what we see in books.

INDEXING

Indexing is defined based on its indexing attributes. Indexing can be of the following types −

* **Primary Index** − Primary index is defined on an ordered data file. The data file is ordered on a **key field**. The key field is generally the primary key of the relation.
* **Secondary Index** − Secondary index may be generated from a field which is a candidate key and has a unique value in every record, or a non-key with duplicate values.
* **Clustering Index** − Clustering index is defined on an ordered data file. The data file is ordered on a non-key field.