**MODULE : 5 (DATABASE)**

**1 WHAT DO YOU UNDERSTAND BY DATABASE**

database is simply a structured and systematic way of storing information to be accessed, analyzed, transformed, updated and moved to other databases.

**2 WHAT IS NORMALIZATION?**

Database Normalization is a stepwise formal process that allows us to decompose database tables in such a way that both data dependency and update anomalies are minimized. It makes use of functional dependency that exists in the table and the primary key or candidate key in analyzing the tables. Normal forms were initially proposed called

* [First Normal Form (INF)](https://www.geeksforgeeks.org/first-normal-form-1nf/)
* [Second Normal Form (2NF)](https://www.geeksforgeeks.org/second-normal-form-2nf/)
* [Third Normal Form (3NF)](https://www.geeksforgeeks.org/third-normal-form-3nf/)

**3 WHAT IS DIFFRENCE BETWEEN DBMS AND RDBMS?**

|  |  |
| --- | --- |
| **DBMS** | **RDBMS** |
| **DBMS stores data as a file.** | **RDBMS stores data in tabular form** |
| Data elements need to access individually. | Multiple data elements can be accessed at the same time. |
| No relationship between data. | Data is stored in the form of tables which are related to each other. |
| Normalization is not present. | Normalization is present. |
| DBMS does not support distributed database. | RDBMS supports distributed database. |
| It stores data in either a navigational or hierarchical form. | It uses a tabular structure where the headers are the column names, and the rows contain corresponding values. |
| It deals with small quantity of data. | It deals with large amount of data. |

**4. What is MF Cod Rule of RDBMS Systems?**

Rule 1: The Information Rule

All information, whether it is user information or metadata, that is stored in a database must be entered as a value in a cell of a table. It is said that everything within the database is organized in a table layout.

Rule 2: The Guaranteed Access Rule

Each data element is guaranteed to be accessible logically with a combination of the table name, primary key (row value), and attribute name (column value).

Rule 3: Systematic Treatment of NULL Values

Every Null value in a database must be given a systematic and uniform treatment.

Rule 4: Active Online Catalog Rule

The database catalog, which contains metadata about the database, must be stored and accessed using the same relational database management system.

Rule 5: The Comprehensive Data Sublanguage Rule

A crucial component of any efficient database system is its ability to offer an easily understandable data manipulation language ([DML](https://www.geeksforgeeks.org/dml-full-form/)) that facilitates defining, querying, and modifying information within the database.

Rule 6: The View Updating Rule

All views that are theoretically updatable must also be updatable by the system.

Rule 7: High-level Insert, Update, and Delete

A successful database system must possess the feature of facilitating high-level insertions, updates, and deletions that can grant users the ability to conduct these operations with ease through a single query.

Rule 8: Physical Data Independence

Application programs and activities should remain unaffected when changes are made to the physical storage structures or methods.

Rule 9: Logical Data Independence

Application programs and activities should remain unaffected when changes are made to the logical structure of the data, such as adding or modifying tables.

Rule 10: Integrity Independence

Integrity constraints should be specified separately from application programs and stored in the catalog. They should be automatically enforced by the database system.

Rule 11: Distribution Independence

The distribution of data across multiple locations should be invisible to users, and the database system should handle the distribution transparently.

Rule 12: Non-Subversion Rule

If the interface of the system is providing access to low-level records, then the interface must not be able to damage the system and bypass security and integrity constraints.

"GeeksforGeeks helped me ace the GATE exam! Whenever I had any doubt regarding any topic, GFG always helped me and made my concepts quiet clear."

**5. What do you understand By Data Redundancy?**

Data redundancy refers to the practice of keeping data in two or more places within a database or data storage system. Data redundancy ensures an organization can provide continued operations or services in the event something happens to its data -- for example, in the case of data corruption or [data loss](https://www.techtarget.com/searchdatabackup/definition/Data-loss). The concept applies to areas such as databases, computer memory and file storage systems.

Data redundancy can occur within an organization intentionally or accidentally. If done intentionally, the same data is kept in different locations with the organization making a conscious effort to protect it and ensure its consistency. This data is often used for backups or disaster recovery.

If carried out by accident, duplicate data may cause data inconsistencies. Even though data redundancy can help minimize the chance of data loss, redundancy issues can affect larger data sets. For example, data that is stored in several places takes up valuable storage space and makes it difficult for the organization to identify which data they should access or update.

The word *redundant* can also be used as an independent technical term to refer to the following:

1. Computer or network system [components](https://www.techtarget.com/whatis/definition/component) that are installed to back up primary resources in case they fail.
2. Redundant information that is unneeded or duplicated.
3. Redundant bits or extra binary digits that are generated and moved with a data transfer to ensure that no bits were lost during the data transfer.
4. Redundant data that protects a storage [array](https://www.techtarget.com/searchstorage/definition/array) against data loss in the event of a hard disk failure.

**6. What is DDL Interpreter?**

DDL Interpreter: The DDL Interpreter is the **Data Definition Language Interpreter** which is used to build and modify the structure of your tables and other objects in the table.

**7. What is DML Compiler in SQL?**

DML is an abbreviation of **Data Manipulation Language**.

The DML commands in Structured Query Language change the data present in the SQL database. We can easily access, store, modify, update and delete the existing records from the database using DML commands.

**Following are the four main DML commands in SQL:**

1. SELECT Command
2. INSERT Command
3. UPDATE Command
4. DELETE Command

**8. What is SQL Key Constraints writing an Example of SQL Key Constraints**

SQL constraints are used to specify rules for data in a table.

## SQL Create Constraints

Constraints can be specified when the table is created with the CREATE TABLE statement, or after the table is created with the ALTER TABLE statement.

The following constraints are commonly used in SQL:

NOT NULL - Ensures that a column cannot have a NULL value

UNIQUE - Ensures that all values in a column are different

PRIMARY KEY - A combination of a NOT NULL and UNIQUE. Uniquely identifies each row in a table

FOREIGN KEY - Prevents actions that would destroy links between tables

CHECK - Ensures that the values in a column satisfies a specific condition

DEFAULT - Sets a default value for a column if no value is specified

CREATE INDEX - Used to create and retrieve data from the database very quickly

**9. What is save Point? How to create a save Point write a Query?**

* Savepoint is a command in SQL that is used with the rollback command.
* It is a command in Transaction Control Language that is used to mark the transaction in a table.
* Consider you are making a very long table, and you want to roll back only to a certain position in a table then; this can be achieved using the savepoint.
* If you made a transaction in a table, you could mark the transaction as a certain name, and later on, if you want to roll back to that point, you can do it easily by using the transaction's name.
* Savepoint is helpful when we want to roll back only a small part of a table and not the whole table. In simple words, we can say savepoint is a bookmark in SQL.

**10.What is trigger and how to create a Trigger in SQL ?**

**Trigger** is a statement that a system executes automatically when there is any modification to the database. In a trigger, we first specify when the trigger is to be executed and then the action to be performed when the trigger executes. Triggers are used to specify certain integrity constraints and referential constraints that cannot be specified using the constraint mechanism of SQL.

**Types of Triggers –**   
We can define 6 types of triggers for each table:

1. **AFTER INSERT** activated after data is inserted into the table.
2. **AFTER UPDATE:** activated after data in the table is modified.
3. **AFTER DELETE:** activated after data is deleted/removed from the table.
4. **BEFORE INSERT:** activated before data is inserted into the table.
5. **BEFORE UPDATE:** activated before data in the table is modified.
6. **BEFORE DELETE:** activated before data is deleted/removed from the table.