MODULE: 5 (DATABASE)

VADHIYA NIRAJ

MODULE: 5_DBMS

1. What do you understand By Database

Ans: - A database is an organized collection of structured information or data, typically stored electronically in a computer system. It's like a digital filing cabinet where data can be efficiently accessed, analyzed, transformed, updated, and moved.

2. What is Normalization?

Ans: - In a DBMS,Normalization is the process of organizing data in a database. It includes creating tables and establishing relationships between those tables according to rules designed both to protect the data and to make the database more flexible by eliminating redundancy and inconsistent dependency.

3. What is Difference between DBMS and RDBMS?

Ans: - Difference between RDBMS and DBMS

RDBMS	DBMS
Data stored is in table format	Data stored is in the file format
Data in the form of a table are linked together	Individual access of data elements
Support distributed database	No support for distributed database

Data is stored in a large amount	Data stored is a small quantity
RDBMS supports multiple users	DBMS supports a single user
Oracle, SQL Server.	XML, Microsoft Access.

4. What is MF Cod Rule of RDBMS Systems?

Ans: - Codd’s rules are proposed by a computer scientist named Dr. Edgar F. Codd and he also invent the relational model for database management.

- **Codd’s Rules in DBMS**

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.

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 CatLog 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) 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.

5. What do you understand By Data Redundancy?

Ans: - Redundancy means having multiple copies of the same data in the database. This problem arises when a database is not normalized. Suppose a table of student details attributes is: student ID, student name, college name, college rank, and course opted.

6. What is DDL Interpreter?

Ans: - DDL Interpreter interprets the DDL statements and records the generated statements in the table containing metadata.

7. What is DML Compiler in SQL?

Ans: - DML Compiler: Translates DML statements in a query language within low level instructions understandable through the query evaluation engine.

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

Ans: - In a database table, we can add rules to a column known as constraints.

- constraints used in SQL are:

1. NOT NULL	values cannot be null
2. UNIQUE	values cannot match any older value
3. PRIMARY KEY	used to uniquely identify a row
4. FOREIGN KEY	references a row in another table
5. CHECK	validates condition for new value
6. DEFAULT	set default value if not passed
7. CREATE INDEX	used to speed up the read process

- **Example of SQL Key Constraints**

NOT NULL Constraint	CREATE TABLE Colleges (
	  college_id INT NOT NULL,
	  college_code VARCHAR(20) NOT NULL,
	  college_name VARCHAR(50)
	);

UNIQUE Constraint	CREATE TABLE Colleges (
	  college_id INT NOT NULL UNIQUE,
	  college_code VARCHAR(20) UNIQUE,
	  college_name VARCHAR(50)
	);

PRIMARY KEY Constraint	CREATE TABLE Colleges (
	  college_id INT PRIMARY KEY,
	  college_code VARCHAR(20) NOT NULL,
	  college_name VARCHAR(50)
	);

FOREIGN KEY Constraint	CREATE TABLE Orders (OrderID int NOT NULL PRIMARY KEY, OrderNumber int NOT NULL, PersonID int FOREIGN KEY REFERENCES Persons(PersonID));
CHECK Constraint	CREATE TABLE Persons (ID int NOT NULL, LastName varchar(255) NOT NULL, FirstName varchar(255), Age int CHECK (Age>=18));

DEFAULT Constraint	CREATE TABLE Orders (ID int NOT NULL, OrderNumber int NOT NULL, OrderDate date DEFAULT GETDATE());

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

Ans: - A save point is a logical rollback point within a transaction. When you set a save point, whenever an error occurs past a save point, you can undo the events you have done up to the save point using the rollback.

- Here is an example of how to create a savepoint in SQL

  SAVEPOINT my_savepoint;

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

Ans: - An SQL trigger is a database object that is associated with a table and automatically executes a set of SQL statements when a specific event occurs on that table.

- We can create a trigger in SQL Server by using the
- CREATE TRIGGER statement as follows: