

## 1.What do you understand By Database

→A database is an organized collection of data. It stored and accessed data electronically.

## 2. What is Normalization?

→Normalization is the process of organizing data in a database to minimize redundancy and breaking down a table into smaller tables and establishing relationships between them.

## 3.What is Difference between DBMS and RDBMS?

→DBMS:

In DBMS data stored in file format. No connection between data. data stored in small quantity. Its supports single user.

→RDBMS:

In RDBMS data stored in table format. Data are linked together. Data is stored in large amount. RDBMS supports multiple users.

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

→Cod's Rules in RDBMS:

1. Data is stored in tables.
2. Data is accessible via primary key.
3. Null values are handled consistently.

4. Metadata is stored in tables.
5. A single language (like SQL) is used for all operations.
6. Views can be updated.
7. High-level operations (like SQL commands) can insert, update, or delete data.
8. Physical storage details don't affect data retrieval.
9. Logical structure changes don't affect applications.
10. Integrity constraints are independent of the application.
11. Distributed data is accessible like local data.
12. Subversion prevention ensures database integrity rules can't be bypassed.

5. What do you understand By Data Redundancy?

→ Data redundancy refers to the unnecessary duplication of data within a database.

6. What is DDL Interpreter?

→ A DDL Interpreter is used to define and modify the structure of database objects such as tables and indexes. DDL commands are Create, Alter, and Drop.

7. What is DML Compiler in SQL?

→ A DML Compiler are used to manipulate data within the database. DML commands include Select, Insert, Update, and Delete.

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

SQL Key Constraints are rules applied to table columns to ensure data integrity and accuracy. Common key constraints include:

1. **Primary Key:** Uniquely identifies each record in a table.
  2. **Foreign Key:** Ensures referential integrity by linking to a primary key in another table.
  3. **Unique Key:** Ensures all values in a column or a set of columns are unique across the table.
  4. **Check Constraint:** Ensures the values in a column satisfy a specific condition.
- Example:  
-- Create a table with a primary key  
CREATE TABLE Customers (  
    customer\_id INT PRIMARY KEY,  
    name VARCHAR(50),  
    city VARCHAR(50)  
);

```
-- Create another table with a foreign key
CREATE TABLE Orders (
    order_id INT PRIMARY KEY,
    order_date DATE,
    customer_id INT,
    FOREIGN KEY (customer_id) REFERENCES
Customers (customer_id)
);
```

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

→ A savepoint is like a bookmark within a transaction in SQL. If you make a mistake after setting the savepoint, you can roll back to this point without undoing the entire transaction.

Syntax:

1. Create Savepoint:

```
SAVEPOINT savepoint_name;
```

## 2. Rollback to Savepoint :-

ROLLBACK TO SAVEPOINT savepoint\_name;

## 3. Release Savepoint (Optional):

RELEASE SAVEPOINT savepoint\_name;

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

→ A trigger is like an automatic action set up in a database. When you add or change data in a table, the trigger jumps into action and does something you've predefined.

### **Syntax:**

CREATE TRIGGER [schema\_name.]trigger\_name

ON table\_name

AFTER {[INSERT], [UPDATE], [DELETE]}

AS

BEGIN

    -- SQL statements

END;