

## MODULE 5.1: BASIC OF DATABASE

Que1 What do you understand By Database?

- Ans:
- Database is a collection of inter-related data and Management System is a set of programs to store and retrieve those data.
  - DBMS stands for Data Base Management System.
  - Database is a collection of inter-related data and Management System is a set of programs to store and retrieve those data.
  - DBMS is a collection of inter-related data and set of programs to store & access those data in an easy and effective manner.
  - For Example, university database organizes the data about students, faculty, and admin staff etc. which helps in efficient retrieval, insertion and deletion of data from it.

Que 2 What is Normalization?

- Ans:
- Normalization is the process of minimizing redundancy (duplicity) from a relation or set of relations. Redundancy in relation may cause insertion, deletion and updation anomalies. So, it helps to minimize the redundancy in relations.
  - Most Commonly used normal forms:
    - First normal form(1NF)
    - Second normal form(2NF)
    - Third normal form(3NF)
    - Boyce & Cod normal form(BCNF)

## MODULE 5.1: BASIC OF DATABASE

Que 3 What is Difference between DBMS and RDBMS?

Ans:

DBMS	RDBMS
<b>DBMS stores data as file.</b>	RDBMS stores data in tabular form.
<b>Data elements need to access individually.</b>	Multiple data elements can be accessed at the same time.
<b>No relationship between data.</b>	Data is stored in the form of tables which are related to each other.
<b>Normalization is not present.</b>	Normalization is present.
<b>DBMS does not support distributed database.</b>	RDBMS supports distributed database.
<b>It stores data in either a navigational or hierarchical form.</b>	It uses a tabular structure where the headers are the column names, and the rows contain corresponding values.
<b>It deals with small quantity of data.</b>	It deals with large amount of data.
<b>Data redundancy is common in this model.</b>	Keys and indexes do not allow Data redundancy.
<b>It is used for small organization and deal with small data.</b>	It is used to handle large amount of data.
<b>It supports single user.</b>	It supports multiple users.
<b>Data fetching is slower for the large amount of data.</b>	Data fetching is fast because of relational approach.
<b>The data in a DBMS is subject to low security levels with regards to data manipulation.</b>	There exist multiple levels of data security in a RDBMS.

## MODULE 5.1: BASIC OF DATABASE

Examples: XML, Window Registry, etc.

Examples: MySQL, PostgreSQL, SQL Server, Oracle, Microsoft Access etc.

Que 4 What is MF Cod Rule of RDBMS Systems?

- Ans:
- Codd's Rules, also known as Codd's Twelve Rules, were proposed by Dr. E.F. Codd, the inventor of the relational database model.
  - These rules were designed to define what characteristics a system must have to be considered a true relational database management system (RDBMS).
  - Here is a summary of Codd's rules:
    - 1) Information Rule:
      - All information in the database is to be stored in one and only place, the relational database.
    - 2) Guaranteed Access Rule:
      - Each unique piece of data (atomic value) is accessible by specifying the table name, primary key value, and column name.
    - 3) Systematic Treatment of Null Values:
      - The DBMS must allow each field to remain null (or empty).
    - 4) Dynamic Online Catalog Based on the Relational model:
      - The database schema, including metadata such as table definitions and constraints, is stored in a catalog that is accessible to users.
    - 5) Comprehensive Data Sublanguage Rule:
      - The DBMS must support a data sublanguage that is comprehensive, meaning it can express any complex query or operation.
    - 6) View Updating Rule:
      - All views that are theoretically updatable must be updatable by the system.

## MODULE 5.1: BASIC OF DATABASE

7) High-level insert, update and delete:

- The capacity of handling a base relation or a derived relation as a single operand applies not only to the retrieval of data but also to the insertion, update, and deletion of data.

8) Physical data independence:

- Changes in the physical storage structure or devices should not affect the access to the stored data.

9) Logical data independence:

- Changes to the logical schema (table structures, constraints) should not affect existing applications.

10) Integrity independence:

- Integrity constraints must be definable separately from application programs and stored in the catalog.

11) Distribution independence:

- A user's perception of the data should not be altered by the way the data is physically distributed or stored.

12) Nonsubversion rule:

- If a relational system has a low-level language, that low-level language cannot be used to subvert or bypass the integrity rules and constraints expressed in the higher-level relational language.

Que 5 What do you understand By Data Redundancy?

Ans : ➤ Data redundancy occurs when the same piece of data is stored in two or more separate places and is a common occurrence in many businesses. As more companies are moving away from siloed data to using a central repository to store information, they are finding that their database is filled with inconsistent duplicates of the same entry. Although it can be challenging to reconcile — or even

## MODULE 5.1: BASIC OF DATABASE

benefit from — duplicate data entries, understanding how to reduce and track data redundancy efficiently can help mitigate long-term inconsistency issues for your business.

- Data redundancy can be found in a database, which is an organized collection of structured data that's stored by a computer system or the cloud. A retailer may have a database to track the products they stock. If the same product gets entered twice by mistake, data redundancy takes place.
- The same retailer may keep customer files in a file storage system. If a customer purchases from the company more than once, their name may be entered multiple times. Duplicate entries of the customer name is consider redundant data.

Que 6 What is DDL Interpreter?

- Ans:
- Data Definition Language (DDL) is used to create and modify the structure objects in a database using predefined commands and a specific syntax. These database objects include tables, sequences, locations ,aliases, schemas and indexes.
  - Command Description:
    - CREATE -Creates a new table, a view of table, or other object in database.
    - ALTER - Modifies an existing database object, such as a table.
    - DROP - Deletes an entire table, a view of a table or other object in the database.

Que 7 What is DML Compiler in SQL?

- Ans:
- DML stands for Data Manipulation Language. Tables and formulas are helpful when communicating with data stored up to a point in a database through SQL, but a time comes when we actually want to execute some fairly complicated data interactions. We will also need the Data Manipulation Language in that situation.
  - DML is a way to inform a database precisely what we want it to do by conversing in a manner that it has been built to comprehend from the scratch. When it comes to interacting

## MODULE 5.1: BASIC OF DATABASE

within existing data, whether adding, moving, or deleting data, it provides a convenient way to do so.

- Data manipulation includes introducing data into tables, altering the table's data and deleting the data from the table.

Que 8    What is SQL Key Constraints writing an Example of SQL Key Constraints?

- Ans:
- SQL constraints are used to specify rules for the data in a table.
  - Constraints are used to limit the type of data that can go into a table. This ensures the accuracy and reliability of the data in the table. If there is any violation between the constraint and the data action, the action is aborted.
  - The following constraints are commonly used in SQL:
    - **NOT NULL** - Ensures that a column cannot have a NULL value.
    - **UNIQUE** – Ensure 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.

## MODULE 5.1: BASIC OF DATABASE

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

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

Ans: ➤ A SAVEPOINT is a point in a transaction when you can roll the transaction back to a certain point without rolling back the entire transaction.  
➤ The syntax for a SAVEPOINT command is as shown below.

SAVEPOINT SAVEPOINT\_NAME;

➤ This command serves only in the creation of a SAVEPOINT among all the transactional statements. The ROLLBACK command is used to undo a group of transactions.

➤ The syntax for rolling back to a SAVEPOINT is as shown below.

ROLLBACK TO SAVEPOINT\_NAME;

➤ Query:

START TRANSACTION;

SAVEPOINT ini;

INSERT INTO student VALUES (10, "Saurabh Singh", 54, "Kashmir", "1989-01-06");

ROLLBACK TO ini ;

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

## MODULE 5.1: BASIC OF DATABASE

Ans:

- 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.
- Create trigger in SQL

delimiter \$\$

```
CREATE TRIGGER Check_age BEFORE INSERT ON employee
```

```
FOR EACH ROW
```

```
BEGIN
```

```
IF NEW.age < 25 THEN
```

```
SIGNAL SQLSTATE '45000'
```

```
SET MESSAGE_TEXT = 'ERROR:
```

```
    AGE MUST BE ATLEAST 25 YEARS!';
```

```
END IF;
```

```
END; $$
```

```
delimiter;
```