MODULE: 5 --> Database

1> What do you understand By Database?

Ans. Database is a collection of data in a format that can be easily accessed (Digital).

-> A software application used to manage our DB is called DBMS (DATABASE MANAGEMWENT SYSTEM).

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 redundancies in relations.

3> What is Difference between DBMS and RDBMS?

Ans.

DBMS	RDBMS
1.It stores data as a file.	**1.**It stores data in a tabular form.
2.Normalization is not present.	**2.**Normalization is present.
3.No relationship between data.	**3.**Data Stored in tables which are related to each other.
4.It supports single users.	**4.**It supports multiple users.
5.Data fetching is slower for the large amount of data.	**5.**Data fetching is fast because of relational approach
6.Ex: Microsoft access, Excel.	**6.**Ex: MySQL, Oracal.

4> What is MF Cod Rule of RDBMS Systems?

Ans. Edgar F. Codd wrote a paper in 1985 defining rules for Relational Database Management Systems (RDBMS), which revolutionized the IT industry. In 1993, Codd and colleagues worked up these 12 rules for defining OLAP (Online Analytical Processing), an industry of software and data processing which allows consolidation and analysis of data in a multidimensional space. Codd's 12 rules are:

1.Multidimensional conceptual view

User-analysts would view an enterprise as being multidimensional in nature – for example, profits could be viewed by region, product, time period, or scenario (such as actual, budget, or forecast). Multi-dimensional data models enable more straightforward and intuitive manipulation of data by users, including slicing and dicing.

2.Transparency

When OLAP forms part of the users' customary spreadsheet or graphics package, this should be transparent to the user. OLAP should be part of an open systems architecture which can be embedded in any place desired by the user without adversely affecting the functionality of the host tool. The user should not be exposed to the source of the data supplied to the OLAP tool, which may be homogeneous or heterogeneous.

3.Accessibility

The OLAP tool should be capable of applying its own logical structure to access heterogeneous sources of data and perform any conversions necessary to present a coherent view to the user. The tool (and not the user) should be concerned with where the physical data comes from.

4.Consistent reporting performance

The performance of the OLAP tool should not suffer significantly as the number of **dimensions** is increased.

5.Client/server architecture

The server component of OLAP tools should be sufficiently intelligent so that the various clients can be attached with minimum effort. The server should be capable of mapping and consolidating data between disparate databases.

6.Generic Dimensionality

Every data dimension should be equivalent in its structure and operational capabilities.

7.Dynamic sparse matrix handling

The OLAP server's physical structure should have optimal sparse matrix handling.

8.Multi-user support

OLAP tools must provide concurrent retrieval and update access, integrity and security.

9.Unrestricted cross-dimensional operations

Computational facilities must allow calculation and data manipulation across any number of **data dimensions**, and must not restrict any relationship between data cells.

10.Intuitive data manipulation

Data manipulation inherent in the consolidation path, such as drilling down or zooming out, should be accomplished via direct action on the analytical model's cells, and not require use of a menu or multiple trips across the user interface.

11.Flexible reporting

Reporting facilities should present information in any way the user wants to view it.

12.Unlimited Dimensions and aggregation levels.
The number of data dimensions supported should, to all intents and purposes, be unlimited. Each generic dimension should enable an essentially unlimited number of user-defined aggregation levels within any given consolidation path.

5> What do you understand By Data Redundancy?

Ans. Data redundancy refers to the duplication of data or storing multiple copies of the same data in multiple locations. This can be done for a variety of reasons, such as to ensure data availability in case of hardware failure, or to improve data security by having multiple backups.

6> What is DDL Interpreter?

Ans. The DDL interpreter interprets DDL statements and records the definition in the data dictionary.

7> What is DML Compiler in SQL?

Ans. The DML compiler translates DML statements in a query language into an evaluation plan consisting of low-level instructions that the query evaluation engine understands. The DML compiler also performs query optimization, which is it picks the lowest cost evaluation plan from among the alternatives. Query evaluation engine executes low level instructions generated by the DML compiler.

8> What are SQL Key Constraints? Writing an Example of SQL Key Constraints.

Ans. Constraints in SQL means we are applying certain conditions or restrictions on the database. This further means that before inserting data into the database, we are checking for some conditions. If the condition we have applied to the database holds true for the data which is to be inserted, then only the data will be inserted into the database tables.

Examples:

--> Every person has a unique email id. This is because while creating an email account for any user, the email providing services such as Gmail, Yahoo or any other email providing service will always check for the availability of the email id that the user wants for himself. If some other user already takes the email id that the user wants, then it cannot be assigned to another user. This simply means that no two users can have the same email ids on the same email providing service. So, here the email id is the constraint on the database of email providing services.

--> Whenever we set a password for any system, there are certain constraints that are to be followed. These constraints may include the following:

   -> There must be one uppercase character in the password.
   -> Password must be of at least eight characters in length.
   -> Password must contain at least one special symbol.

9> What is save Point? How to create a save Point write a Query?
Ans. Save point 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 Save point.

--> If you made a transaction on 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.

--> Save point 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 Save point is a bookmark in SQL.

--> We will use the MySQL database for writing all the queries.

MySQL> USE database;

MySQL>create table student (id int, name varchar (50), percentage int);

10> What is trigger and how to create a Trigger in SQL?
Ans. A trigger is a stored procedure in database which automatically invokes whenever a special event in the database occurs.

CREATE [OR REPLACE] TRIGGER trigger-name
{BEFORE | AFTER | INSTEAD OF}
{INSERT [OR] | UPDATE [OR] | DELETE}
[OF col-name]
ON table-name
[REFERENCING OLD AS o NEW AS n]
[FOR EACH ROW]
WHEN (condition)
DECLARE
        Declaration-statements
BEGIN
        Executable statements
EXCEPTION

Exception-handling-statements
END;