

Assignment Module 3 (Testing on Live Application)

Q.1 What is RDBMS ?

Ans :-

A relational database is a type of database that stores and provides access to data points that are related to one another. Relational databases are based on the relational model, an intuitive, straightforward way of representing data in tables. In a relational database, each row in the table is a record with a unique ID called the key. The columns of the table hold attributes of the data, and each record usually has a value for each attribute, making it easy to establish the relationships among data points.

Q.2 What is SQL ?

Ans :-

Structured Query Language (SQL) refers to a standard programming language utilized to extract, organize, manage, and manipulate data stored in relational databases. SQL is thereby referred to as a database language that can execute activities on databases that consist of tables made up of rows and columns.

Q.3 Write SQL Commands.

Ans :-

SQL Commands

1. Data Definition Language (DDL)
2. Data Manipulation Language (DML)
3. Data Control Language(DCL)
4. Transaction Control Language(TCL)
5. Data Query Language(DQL)

Q.4 What is join?

Ans :-

A SQL join is an SQL clause that is used to join records from two or more tables in a database. The records from the tables are combined by a common attribute that the tables share, resulting in a single record set. This record set can be inserted into a temporary table to be stored and used.

Q.5 Write type of joins.

Ans :-

There are four types of SQL joins

1. Inner SQL Joins

Inner joins are the most commonly used. They only combine records from the two tables if they both match the join condition (share a common attribute). This joins work best when referential integrity is enforced in the database, especially on primary and foreign keys.

2. Outer SQL Joins

Outer joins combine everything from both tables together, even if both records do not share a matching value. They can be useful if you need to combine two tables together and want to keep all of the rows, or want a large result set.

Assignment Module 3 (Testing on Live Application)

3. Left SQL Joins

Left joins are a type of outer join that retains all of the records of the left table and include only records from the right table that match the join condition (share a common attribute). Left joins are similar to an inner join in that you are getting all of the related records that both tables share, but all of the left table's records are included in the result set whether they match the join condition or not. Left joins are particularly useful if you are joining two tables on a value that may be NULL (blank/empty) in the right table.

4. Right SQL Joins

Right joins are exactly the same as left joins except that all of the records from the right table are retained instead. Records from the left table that match the join condition (share a common attribute) will still be included in the result set as well. Right joins come in handy when the left table may contain NULLS (blanks) in the common field that is being used to join.

Q.6 How Many constraint and describes it self ?

Ans :-

There are six main constraints that are commonly used in SQL Server that we will describe deeply with examples within this article and the next one. These constraints are:

- NOT NULL
- UNIQUE
- PRIMARY KEY
- FOREIGN KEY
- CHECK
- DEFAULT

- **NOT NULL –**

If we specify a field in a table to be NOT NULL. Then the field will never accept null value. That is, you will be not allowed to insert a new row in the table without specifying any value to this field.

For example, the below query creates a table Student with the fields ID and NAME as NOT NULL. That is, we are bound to specify values for these two fields every time we wish to insert a new row.

Syntax: CREATE TABLE Student (ID int(6) NOT NULL, NAME varchar(10) NOT NULL, ADDRESS varchar(20));

- **UNIQUE –**

This constraint helps to uniquely identify each row in the table. i.e. for a particular column, all the rows should have unique values. We can have more than one UNIQUE column in a table.

For example, the below query creates a table Student where the field ID is specified as UNIQUE. i.e, no two students can have the same ID. Unique constraint in detail.

Syntax: CREATE TABLE Student(ID int(6) NOT NULL UNIQUE, NAME varchar(10), ADDRESS varchar(20));

Assignment Module 3 (Testing on Live Application)

- **PRIMARY KEY -**

Primary Key is a field which uniquely identifies each row in the table. If a field in a table as primary key, then the field will not be able to contain NULL values as well as all the rows should have unique values for this field. So, in other words we can say that this is combination of NOT NULL and UNIQUE constraints.

A table can have only one field as primary key. Below query will create a table named student and specifies the field ID as primary key.

Syntax: CREATE TABLE Student(ID int(6) NOT NULL UNIQUE,NAME varchar(10), ADDRESS varchar(20),PRIMARY KEY(ID));

- **FOREIGN KEY –**

Foreign Key is a field in a table which uniquely identifies each row of a another table. That is, this field points to primary key of another table. This usually creates a kind of link between the tables.

Consider the two tables as shown below:

Orders

O_ID	ORDER_NO	C_ID
1	2253	3
2	3325	3
3	4521	2
4	8532	1

Customers

C_ID	NAME	ADDRESS
1	RAMESH	DELHI
2	SURESH	NOIDA
3	DHARMESH	GURGAON

As we can see clearly that the field C_ID in Orders table is the primary key in Customers table,i.e. it uniquely identifies each row in the Customers table.

Therefore, it is a Foreign Key in Order stable.

Syntax: CREATE TABLE Orders (O_ID int NOT NULL,ORDER_NO int NOT NULL,C_ID int,PRIMARY KEY (O_ID),FOREIGN KEY (C_ID) REFERENCES Customers(C_ID))

- **CHECK –**

Using the CHECK constraint we can specify a condition for a field, which should be satisfied at the time of entering values for this field.

For example, the below query creates a table Student and specifies the condition for the field AGE as (AGE >= 18). That is, the user will not be allowed to enter any record in the table with AGE < 18.

Syntax: CREATE TABLE Student(ID int(6) NOT NULL,NAME varchar(10) NOT NULL,AGE int NOT NULL CHECK (AGE >= 18));

Assignment Module 3 (Testing on Live Application)

- **DEFAULT –**

This constraint is used to provide a default value for the fields. That is, if at the time of entering new records in the table if the user does.

For example, the below query will create a table named Student and specify the default value for the field AGE as 18.

Syntax: CREATE TABLE Student(ID int(6) NOT NULL,NAME varchar(10) NOT NULL,AGE int DEFAULT 18);

Q.7 Difference between RDBMS vs DBMS.

Ans:-

ASPECT	DBMS (DATABASE MANAGEMENT SYSTEM)	RDBMS (Relational Database Management System)
1. Data Organization	Stores data in files and directories.	Organizes data into tables with predefined relationships.
2. Data Relationships	Does not enforce relationships between data.	Enforces relationships between tables using primary and foreign keys.
3. Data Integrity	Offers limited support for data integrity.	Ensures data integrity through constraints and validations.
4. Data Model	Supports various data models, including NoSQL.	Follows a relational data model with tables and rows.
5. Schema Flexibility	Schema changes may require data migration.	Allows schema modifications without data migration.
6. ACID Properties	May not fully support ACID properties for transactions.	Fully supports ACID properties for reliable transactions.
7. Query Language	May have proprietary query languages.	Uses standard SQL (Structured Query Language) for data manipulation.
8. Performance Optimization	Performance optimization may be limited.	Utilizes advanced query optimization techniques for efficient queries.
9. Scalability	May have limitations in handling large datasets.	Offers scalability to manage large volumes of data effectively.
10. Examples of Systems	File systems, MongoDB, SQLite.	MySQL, PostgreSQL, Oracle, SQL Server, etc.

Assignment Module 3 (Testing on Live Application)

Q.8 What is API Testing ?

Ans:-

API testing involves verifying the API's functionality, reliability, performance, and security. This type of testing typically involves sending requests to the API and checking the responses to ensure that they match the expected results. API testing can be automated using specialized software tools, which can help to save time and reduce the risk of human error.

Q.9 Types of API Testing.

Ans:-

Types of API Testing

There are 3 different types of API testing:

1. **Open APIs** : These types of APIs are publicly available to use like OAuth APIs from google. It has also not given any restriction to use them. So, they are also known as Public APIs.
2. **Partner APIs** : Specific rights or licenses to access this type of API because they are not available to the public.
3. **Internal APIs** : Internal or Private. These APIs are developed by companies to use in their internal systems. It helps you to enhance the productivity of your teams.

Q.10 What is Responsive Testing?

Ans:-

Mobile Responsive Testing is simply a Testing Process performed to make sure that your website or an app is working appropriately on all types of devices. Whether it an app or website, they need to ensure that they are accessible from any device, anytime, anywhere.

Q.11 Which types of tools are available for Responsive Testing?

Ans:-

There are a lot of tools out there to help you test your website's responsiveness.

- LT Browser
- Lambda Testing
- Google Resizer
- Am i responsive
- Pixel tuner

Q.12 What is the full form of .ipa, .apk

Ans:-

.ipa = INTERNATIONAL PHONETIC ALPHABET

.apk = ANDROID APPLICATION PACKAGE

Assignment Module 3 (Testing on Live Application)

Q.13 How to create step for to open the developer option mode ON?

Ans:-

Step 1: Go to Settings > About phone.

Step 2: Scroll down to Build number.

Step 3: Tap Build number seven times. After the first few taps, you should see the steps counting down until you unlock the developer options. You may also have to tap in your PIN for verification.

Step 4: Once developer options are activated, you will see a message that reads, You are now a developer.

Step 5: Go back to the *Settings* pane and head to System, where you will now find Developer options as an entry.

Step 6: Tap it and toggle the switch on if it is not already, and from there, you can proceed to make adjustments to your phone.