

LAB 1: INTRODUCTION TO SQL

THEORY:

1. **SQL:** Structured Query Language (SQL) is the database language using which we can perform certain operations on the existing database and, we can use this language to create a database. It uses certain commands like create, drop, insert, etc. to carry out required tasks.
2. **DDL:** Data Definition Language (DDL) consists of SQL commands that can be used to define the database schema. It is used to create and modify the structure of database objects in the database.
3. **DQL:** Data Query Language (DQL) are used for performing queries on the data within schema objects. The purpose of DQL commands is to get some schema relation based on the query passed out.
4. **DML:** Data Manipulation Language (DML) is the SQL commands that deals with the manipulation of data present in the database belong to DML and this includes most of the SQL statements.
5. **DCL:** Data Control Language (DCL) includes commands such as GRANT and REVOKE which mainly deals with the rights, permissions, and other controls of the database system.
6. **Create:** This command is used to create the database or its objects.
7. **Drop:** This command is used to delete objects from the database.
8. **Insert:** This command is used to insert data into a table.
9. **Alter:** This command is used to alter the structure of the database.
10. **Delete:** This command is used to delete records from a database table.
11. **Use:** This command is used when there are multiple databases in the SQL and the programmer specifically wants to use a particular database.
12. **Domain Integrity Constraints:** Domain integrity constraints are a set of rules which is used to maintain the quality of information. It is used to guard against accidental damage to the database.
 - a. **Check:** This constraint ensures that only valid data is entered into the database. For example, if we have a column for “age > 18” then this constraint will ensure that only values greater than 18 can be entered into that column.
 - b. **Not null:** This constraint ensures that a column of a table contain no null values. This enforces a field to always contain a value, which means that we cannot insert a new record or update a record without adding a value to this field.
 - c. **Unique:** This constraint ensures that all values in a column are different.

SOURCE CODE:

```
SQLQuery3 Abindra...0UPIIQ\ACER (76))* X
drop table student076
create table student076(
  s_id int,
  s_name varchar(30),
  s_phone bigint,
  s_age int
);
select *from student076;
insert into student076 values(1, 'Amit', 9818024659, 25);
insert into student076 values(2, 'Siddhartha', 9818528623, 20);
insert into student076 values(3, 'Anita', 9815689511, 22);
insert into student076 values(4, 'Deepak', 9456928624, 21);
```

DISCUSSION:

Here we created a table named student076 using ‘create table’ command where we stored our data. Before creating the table, we deleted any table that had same name in this database using ‘drop’ command. While creating the table, we assigned some attributes such as s_id, s_name, s_phone and s_age which were assigned with datatypes such as int, varchar, bigint according to their types. We inserted the values into the table using ‘insert’ command. Finally, we selected everything from the table and displayed the entered values in the table.

OUTPUT:

Results		Messages		
	s_id	s_name	s_phone	s_age
1	1	Amit	9818024659	25
2	2	Siddhartha	9818528623	20
3	3	Anita	9815689511	22
4	4	Deepak	9456928624	21

```
SQLQuery3 Abindra...0UPII\ACER (76))* ✕
drop table student076;
create table student076(
    s_id int,
    s_name varchar(30),
    s_phone bigint,
    s_age int
);
select *from student076;
insert into student076 values(1, 'Amit', 9818024659, 25);
insert into student076 values(2, 'Siddhartha', 9818528623, 20);
alter table student076 add s_address varchar(30);
insert into student076 values(3, 'Anita', 9815689511, 26, 'Kathmandu');
insert into student076 (s_id, s_name, s_address) values(4, 'Deepika', 'Lalitpur');
```

DISCUSSION:

Here we updated the existing table with new attribute. In this experiment, we created a new attribute 's_address' in the existing table by using the 'add' command. We provided a new set of values into that table including the value for new attribute. We can also delete the attribute using 'drop' command. Here, we also learned a new method of providing value to the table. We provided a set of values to the table by leaving remaining places 'NULL' in the table. Finally, we displayed the table using 'select' command.

OUTPUT:

Results		Messages			
	s_id	s_name	s_phone	s_age	s_address
1	1	Amit	9818024659	25	NULL
2	2	Siddhartha	9818528623	20	NULL
3	3	Anita	9815689511	26	Kathmandu
4	4	Deepika	NULL	NULL	Lalitpur

```
SQLQuery3 Abindra...0UPIIQ\ACER (76))* X
drop table student076
create table student076(
  s_id int,
  s_name varchar(30),
  s_phone bigint unique,
  s_age int
);
select *from student076;
insert into student076 values(1, 'Amit', 9818024659, 25);
insert into student076 values(2, 'Siddhartha', 9818528623, 20);
insert into student076 values(3, 'Ashok', 9818528623, 27);
```

DISCUSSION:

In this experiment, we tested the domain integrity constant: Unique. This constraint ensures that all values in the column are different. Here, we made the attribute s_phone as Unique so all the values in this column must be unique. We tried to provide same values twice which resulted in a violation of Unique Key constraint. After providing a new and unique value to the table, this violation was resolved. So, this constraint is used to the attributes where the values must be different.

OUTPUT:

```
Messages
(1 row affected)
(1 row affected)
Msg 2627, Level 14, State 1, Line 11
Violation of UNIQUE KEY constraint 'UQ__student0__67C822D1940D20E8'. Cannot insert duplicate key in object 'dbo.student076'. The duplicate key value is (9818528623).
The statement has been terminated.
Completion time: 2022-12-27T21:50:35.7323498+05:45
```

Results		Messages		
	s_id	s_name	s_phone	s_age
1	1	Amit	9818024659	25
2	2	Siddhartha	9818528623	20
3	3	Ashok	9818528633	27

SQLQuery3 Abindra...0UPIIQ\ACER (76))*

```
drop table student076
create table student076(
  s_id int,
  s_name varchar(30),
  s_phone bigint unique,
  s_age int check(s_age > 18)
);
select *from student076;
insert into student076 values(1, 'Amit', 9818024659, 25);
insert into student076 values(2, 'Siddhartha', 9818528623, 20);
insert into student076 values(3, 'Ashok', 9818528553, 27);
insert into student076 values(4, 'Bijay', 9818538633, 16);
```

DISCUSSION:

In this experiment, we tested the domain integrity constraint: Check. This constraint ensures that only valid data is entered into the database. Here, we made the attribute s_age with a constraint as (s_age > 18). So, this column will accept values of s_age that is greater than 18 only. We tried by providing a new value to the table with s_age as 16. This command resulted in violation of Check constraint. We again checked this command with s_age greater than 18, then this violation was resolved. So, this constraint is used when the value provided into the table is critical.

OUTPUT:

Messages

Msg 547, Level 16, State 0, Line 12

The INSERT statement conflicted with the CHECK constraint "CK__student07__s_age__2B5F6B28". The conflict occurred in database "master", table "dbo.student076", column 's_age'. The statement has been terminated.

Completion time: 2022-12-27T22:14:52.8661310+05:45

Results Messages

	s_id	s_name	s_phone	s_age
1	1	Amit	9818024659	25
2	2	Siddhartha	9818528623	20
3	3	Ashok	9818528553	27
4	4	Bijay	9818538633	21

```
SQLQuery3 Abindra...0UPIIQ\ACER (76))* X
drop table student076
create table student076(
s_id int NOT NULL,
s_name varchar(30),
s_phone bigint unique,
s_age int check(s_age > 18)
);
select *from student076;
insert into student076 values(1, 'Amit', 9818024659, 25);
insert into student076 (s_name, s_age) values('Deepika', 21);
```

DISCUSSION:

In this experiment, we tested the domain integrity constraint: Not Null. This constraint ensures that a column of a table contains no NULL values. Here, we made the attribute s_id as NOT NULL which means that the s_id attribute can have no null values. We tested this by providing a Null value to this column. This command resulted in violation of Null constraint. After providing all the required values, this violation was resolved. So, this constraint is used to the must have attributes of the table.

OUTPUT:

Messages
Msg 515, Level 16, State 2, Line 10
Cannot insert the value NULL into column 's_id', table 'master.dbo.student076'; column does not allow nulls. INSERT fails.
The statement has been terminated.
Completion time: 2022-12-27T22:40:22.0189201+05:45

Results Messages

	s_id	s_name	s_phone	s_age
1	1	Amit	9818024659	25
2	2	Deepika	NULL	21

CONCLUSION:

From this experiment, we learned the basic commands of SQL such as create, alter, drop, insert, and we used them to create a database with student's data.