

Class: SE Comp **Experiment No: 5**

Name : Ashish Jha

Roll No. 27

Batch : B

Topic:	Apply Integrity Constraints for the specified system.
Prerequisite:	Knowledge of SQL syntax.
Mapping With COs:	CSL402.2, CSL402.3
Objective:	To implement restrictions on your assigned database (CONSTRAINTS).

Outcome:	<p>After completion of this lab, the student should be able to:</p> <ul style="list-style-type: none"> - Create a simple table - Explain how constraints are created at the time of table creation - Explain the purpose of constraints in a table - Distinguish among PRIMARY KEY, FOREIGN KEY, UNIQUE, CHECK, and NOT NULL constraints and the appropriate use for each constraint - Explain how constraints can be created on an existing table - Create PRIMARY KEY constraints for a single column - Create a FOREIGN KEY constraint, UNIQUE constraint and CHECK constraint
Instruction	<p>1. This experiment is a compulsory experiments. All the students are required to perform this experiment individually.</p> <p>2. Implement all the types of constraints for the assigned</p>
s:	<p>system. For Submissions:</p>
Deliverable	<p>1. Implemented all types of constraints.</p> <ul style="list-style-type: none"> - Specify Rule in sentence - Mention suitable constraint type - Implement it - Add one record which should violate the rule - Add one successful record
s:	<p>ANS:</p> <p>Types of constraints</p> <ol style="list-style-type: none"> 1. Domain integrity constraints 2. Entity integrity constraints 3. Referential integrity constraints <p>1. Domain Integrity Constraints</p> <ul style="list-style-type: none"> ● Domain integrity constraints are used to test the value inserted into the table is correct or not according to the mapped domain. ● Domain integrity constraints <p>(a). Required data constraint / Not null constraint : Some attributes in database are not allowed to contain null values. Null values are the values which are unknown, unassigned or missing attribute values. Eg. in the student database every students must have a student id associated with it that cannot be repeated and can not be null. CREATE TABLE Student(Student_id int not null);</p> <p>(b). Check constraints: use of check constraints is to ensure that attribute value satisfies user defined condition Eg. student gender is either "M" or "F" it can't accept values other than it . CREATE TABLE Student(Gender char(1) not null, Check(Gender in ("M", "F")));</p>

(c).Default keyword : default keyword is used somevalue if no attribute value added for tuple

Eg. Table with student, having name S_id, name and gender in which S_id is primary key if student name is not given then that will be taken as UNKNOWN.

```
CREATE TABLE Student(  
S_id INT NOT NULL  
Name CHAR(30) DEFAULT "Unknown"  
);
```

2. Entity integrity constraints: Entity constraints are used to test the values inserted into the database are correct or not with respect to the tuple in the same table

- **To types** (a). Primary key constraints
(b). Unique constraints

(a) Primary key constraints : primary key constraints is same as that of unique key constraints with not null constraints.

Primary key attribute value needs to be unique as well as null value are not allowed in primary key attributes

Eg. CREATE TABLE Customer (
Name Varchar(20),
C_id Char(10) PRIMARY KEY,
Gender char(1),
CHECK (Gender("M","F"))
);

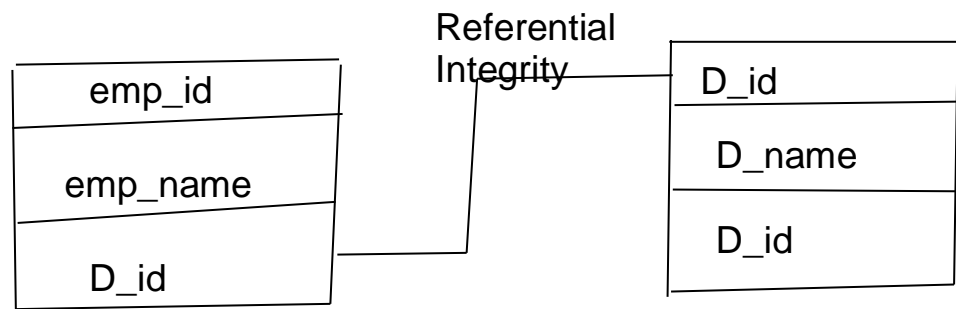
(b) Unique Constraint : In case of unique constraint no two tuples can have equal value for same attributes

This constraint says that attribute form candidate key, which allows one null value which is unique by itself

Eg. CREATE TABLE Customer (
Name Varchar(20),
C_id Char(10) PRIMARY KEY,
Gender char(1),
CHECK (Gender("M","F"))
Email Char(20) UNIQUE
);

3. Referential integrity constraints :

A value appearing in a one relation for a given set of attributes also appear for another set of attributes in other relations this is called referential integrity constraints



```
CREATE TABLE EMPLOYEE(
```

```
Emp_id int ,
```

```
Emp_name varchar(20),
```

```
D_id int
```

```
);
```

```
CREATE TABLE DEPARTMENT (
```

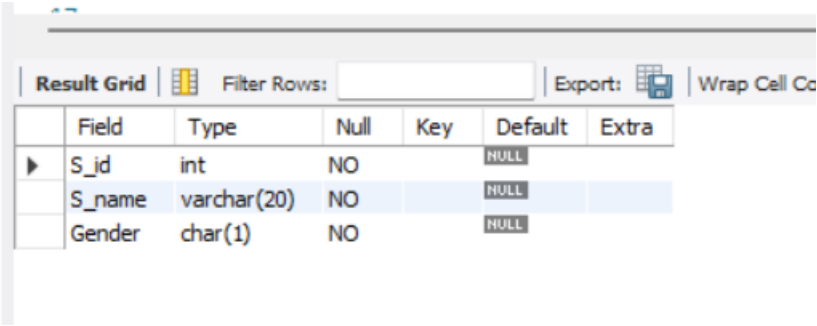
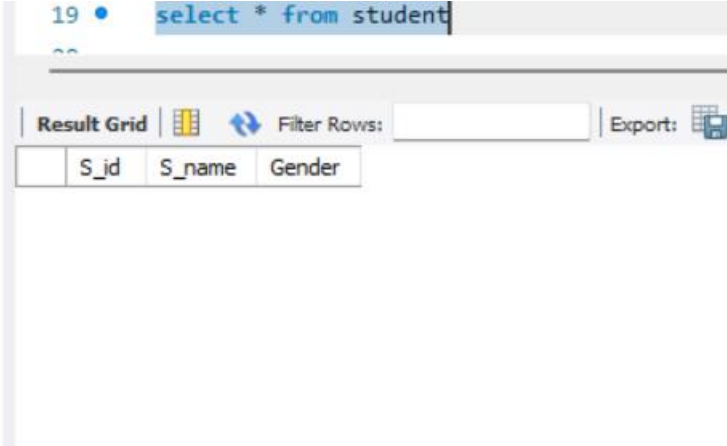
```
D_id int REFERENCE EMPLOYEE (D_id),
```

```
D_name Varchar(20) not null
```

```
);
```

3. For each query capture the output snapshot

ANS :

	 
Conclusion:	Able to create and apply various constraints on database.
References:	Moodle and class notes Database concepts by korth

Experiment No 5 Database Management System Lab 2021-22 Faculty: Sana Shaikh

Class: SE Comp

Don Bosco Institute of Technology

Department of Computer Engineering

Assessment Rubric for Experiment No. 5

Title of Experiment : Implement all types of constraints **Performance Date : Year and Semester :** 2nd Year and IVth Semester **Submission Date :**

Name: Batch :

Roll No. :

Sr. No.	Criteria	1 Marks	2 Marks	3 Marks	4 Marks	5 Marks
1	Execution	Executed 10-30% queries based on following constraints: -Not Null -Check (for particular value and range) -Primary key - Unique key - Foreign key	Executed 31-50% queries based on following constraints: -Not Null -Check (for particular value and range) -Primary key - Unique key - Foreign key	Executed 51-70% queries based on following constraints: -Not Null -Check (for particular value and range) -Primary key - Unique key - Foreign key	Executed 71-89% queries based on following constraints: -Not Null -Check (for particular value and range) -Primary key - Unique key - Foreign key	Executed 90-100% queries based on following constraints: -Not Null -Check (for particular value and range) -Primary key - Unique key - Foreign key
2	Documentati					
3	on Viva	20-39% of solution document ed properly.	40-59% of solution document ed properly.			
4	Submission on Time	Students hardly answered. Submitted after the given deadline	Student has problems while answering. Submitted before the given deadline			

				<div>answered fairly well. completely and correctly.</div>	
--	--	--	--	--	--